

RX14T Group

User's Manual: Hardware

RENESAS 32-Bit MCU
RX Family/RX100 Series

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (<http://www.renesas.com>).

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 claims 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, drawings, charts, programs, algorithms, and 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 be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer 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, copying or reverse engineering.
6. 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; 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.

Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or 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 damage (space system; 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 any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.

7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using 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 dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. 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. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a 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. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using 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.
11. 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 comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation.

All trademarks and registered trademarks are the property of their respective owners.

SuperFlash® is a registered trademark of Silicon Storage Technology, Inc. in several countries including the United States and Japan.

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:

www.renesas.com/contact/

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can 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 must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

Do not input signals or an I/O pull-up power supply while the device is powered off. 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. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. 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 V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, 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 V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

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, electrical characteristics, and usage notes.

Particular attention should be paid to the precautionary notes when using the manual. These notes can be found 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 RX14T Group. Make sure to refer to the latest versions of these documents. The latest versions of the listed documents are available from the Renesas Electronics website.

Document Type	Description	Document Title	Document No.
Datasheet	Hardware overview and electrical characteristics	RX14T Group Datasheet	R01DS0453EJ
User's Manual: Hardware	Hardware specifications (pin assignments, memory maps, peripheral specifications, electrical characteristics, and timing charts) and operation description Note: Refer to the application notes for details on using peripheral functions.	RX14T Group User's Manual: Hardware	This User's manual
User's Manual: Software	Description of CPU instruction set	RX Family RXv2 Instruction Set Architecture User's Manual: Software	R01US0071EJ
Application Note	Notes on Printed Circuit Board Patterns	RX Family Hardware Design Guide	R01AN1411EJ
	Examples of register initial setting	RX14T Group Initial Setting Examples	R01AN7885EJ
	Information on using peripheral functions and application examples Sample programs	Available from Renesas Electronics website.	
Renesas Technical Update	Product specifications, updates on documents, etc.		

2. Notation of Numbers and Symbols

Each register description includes a bit chart, illustrating the arrangement of bits, and a table of bits, describing the meanings of the bit settings. The standard format and notation for bit charts and tables are described below.

X.X.X ... Register

Address(es): xxxx xxxh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	...[1:0]	...4	—	—	—	—	...0
Value after reset:	x	0	0	0	0	0	0	0
x: Undefined								

Bit	Symbol	Bit Name	Description	R/W
b0	...0		0: 1: Setting prohibited (3)	R/W (1)
b3 to b1	—	Reserved (2)	These bits are read as 0. The write value should be 0.	R/W
b4	...4		0: 1:	R
b6, b5	...[1:0]		0 0: 0 1: Settings other than above are prohibited. (3)	R/(W)*1
b7	—	Reserved	The read value is undefined. Writing to this bit has no effect.	R

- (1) R/W: The bit or field is readable and writable.
R/(W): The bit or field is readable and writable. However, writing to this bit or field has some limitations. For details on the limitations, see the description or notes of respective registers.
R: The bit or field is readable. Writing to this bit or field has no effect.
- (2) Reserved.
Use the specified value when writing to this bit or field; otherwise, the correct operation is not guaranteed.
- (3) Setting prohibited. The correct operation is not guaranteed if such a setting is performed.

3. List of Abbreviations and Acronyms

Abbreviation	Full Form
ACIA	Asynchronous Communications Interface Adapter
bps	bits per second
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
GSM	Global System for Mobile Communications
Hi-Z	High Impedance
IEBus	Inter Equipment Bus
I/O	Input / Output
IrDA	Infrared Data Association
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Non-Connect
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
SFR	Special Function Registers
SIM	Subscriber Identity Module
UART	Universal Asynchronous Receiver / Transmitter
VCO	Voltage Controlled Oscillator

Caution: This product uses SuperFlash® technology licensed from Silicon Storage Technology, Inc.

Contents

Features	38
1. Overview	39
1.1 Outline of Specifications	39
1.2 List of Products	44
1.3 Block Diagram	46
1.4 Pin Functions	47
1.5 Pin Assignments	50
1.5.1 64-pin LFQFP, 64-pin LQFP	50
1.5.2 52-pin LQFP	51
1.5.3 48-pin LFQFP	52
1.5.4 48-pin HWQFN	52
1.5.5 44-pin LQFP	53
1.5.6 32-pin LQFP	54
1.5.7 32-pin HWQFN	54
1.5.8 24-pin HWQFN	55
1.6 List of Pins and Pin Functions	56
1.6.1 64-pin LFQFP, 64-pin LQFP	56
1.6.2 52-pin LQFP	59
1.6.3 48-pin LFQFP, 48-pin HWQFN	62
1.6.4 44-pin LQFP	64
1.6.5 32-pin LQFP, 32-pin HWQFN	66
1.6.6 24-pin HWQFN	68
2. CPU	70
2.1 Features	70
2.2 Register Set of the CPU	71
2.2.1 General-Purpose Registers (R0 to R15)	72
2.2.2 Control Registers	72
2.2.2.1 Interrupt Stack Pointer (ISP)/User Stack Pointer (USP)	73
2.2.2.2 Exception Table Register (EXTB)	73
2.2.2.3 Interrupt Table Register (INTB)	73
2.2.2.4 Program Counter (PC)	73
2.2.2.5 Processor Status Word (PSW)	74
2.2.2.6 Backup PC (BPC)	75
2.2.2.7 Backup PSW (BPSW)	76
2.2.2.8 Fast Interrupt Vector Register (FINTV)	76
2.2.2.9 Floating-Point Status Word (FPSW)	77
2.2.3 Accumulator	79
2.3 Processor Mode	80
2.3.1 Supervisor Mode	80
2.3.2 User Mode	80
2.3.3 Privileged Instruction	80

2.3.4	Switching Between Processor Modes	80
2.4	Data Types	81
2.4.1	Integer	81
2.4.2	Floating-Points	82
2.4.3	Bitwise Operations	82
2.4.4	Strings	83
2.5	Endian	84
2.5.1	Switching the Endian	84
2.5.2	Access to I/O Registers	87
2.5.3	Notes on Access to I/O Registers	87
2.5.4	Data Arrangement	88
2.5.4.1	Data Arrangement in Registers	88
2.5.4.2	Data Arrangement in Memory	88
2.6	Vector Table	89
2.6.1	Exception Vector Table	89
2.6.2	Interrupt Vector Table	90
2.7	Operation of Instructions	91
2.7.1	Restrictions on RMPA and String-Manipulation Instructions	91
2.7.1.1	Transfer Size and Data Prefetching	91
2.7.1.2	Access to I/O Registers	91
2.8	Number of Cycles	92
2.8.1	Instruction and Number of Cycle	92
2.8.2	Numbers of Cycles for Response to Interrupts	96
3.	Operating Modes	97
3.1	Operating Mode Types and Selection	97
3.2	Register Descriptions	98
3.2.1	Mode Monitor Register (MDMONR)	98
3.2.2	System Control Register 1 (SYSCR1)	99
3.3	Details of Operating Modes	100
3.3.1	Single-Chip Mode	100
3.3.2	Boot Mode (SCI Interface)	100
3.3.3	Boot Mode (FINE Interface)	100
3.4	Transitions of Operating Modes	100
3.4.1	Mode Setting Pin Levels and Operating Mode Transitions	100
4.	Address Space	101
4.1	Address Space	101
5.	I/O Registers	102
5.1	I/O Register Addresses (Address Order)	104
6.	Resets	121
6.1	Overview	121
6.2	Register Descriptions	123

6.2.1	Reset Status Register 0 (RSTSR0)	123
6.2.2	Reset Status Register 1 (RSTSR1)	124
6.2.3	Reset Status Register 2 (RSTSR2)	125
6.2.4	Software Reset Register (SWRR)	126
6.3	Operation	127
6.3.1	RES# Pin Reset	127
6.3.2	Power-On Reset and Voltage Monitoring 0 Reset	127
6.3.3	Voltage Monitoring 1 Reset and Voltage Monitoring 2 Reset	129
6.3.4	Independent Watchdog Timer Reset	130
6.3.5	Software Reset	130
6.3.6	Determination of Cold/Warm Start	131
6.3.7	Determination of Reset Generation Source	132
7.	Option-Setting Memory (OFSM)	133
7.1	Overview	133
7.2	Register Descriptions	134
7.2.1	Option Function Select Register 0 (OFS0)	134
7.2.2	Option Function Select Register 1 (OFS1)	136
7.2.3	Endian Select Register (MDE)	137
7.3	Usage Note	138
7.3.1	Setting Example of Option-Setting Memory	138
8.	Voltage Detection Circuit (LVDAb)	139
8.1	Overview	139
8.2	Register Descriptions	142
8.2.1	Voltage Monitoring 1 Circuit Control Register 1 (LVD1CR1)	142
8.2.2	Voltage Monitoring 1 Circuit Status Register (LVD1SR)	143
8.2.3	Voltage Monitoring 2 Circuit Control Register 1 (LVD2CR1)	144
8.2.4	Voltage Monitoring 2 Circuit Status Register (LVD2SR)	145
8.2.5	Voltage Monitoring Circuit Control Register (LVCMPCR)	146
8.2.6	Voltage Detection Level Select Register (LVDLVLR)	147
8.2.7	Voltage Monitoring 1 Circuit Control Register 0 (LVD1CR0)	148
8.2.8	Voltage Monitoring 2 Circuit Control Register 0 (LVD2CR0)	149
8.3	VCC Input Voltage Monitor	150
8.3.1	Monitoring Vdet0	150
8.3.2	Monitoring Vdet1	150
8.3.3	Monitoring Vdet2	150
8.4	Reset from Voltage Monitor 0	151
8.5	Interrupt and Reset from Voltage Monitoring 1	152
8.6	Interrupt and Reset from Voltage Monitoring 2	154
8.7	Usage Note	155
8.7.1	Note on Setting the Detection Level for LVD1 While LVD0 Is Enabled	155

9.	Clock Generation Circuit	156
9.1	Overview	156
9.2	Register Descriptions	158
9.2.1	System Clock Control Register (SCKCR)	158
9.2.2	System Clock Control Register 3 (SCKCR3)	160
9.2.3	PLL Control Register (PLLCR)	161
9.2.4	PLL Control Register 2 (PLLCR2)	163
9.2.5	Main Clock Oscillator Control Register (MOSCCR)	164
9.2.6	Low-Speed On-Chip Oscillator Control Register (LOCOCR)	165
9.2.7	IWDT-Dedicated On-Chip Oscillator Control Register (ILOCOCR)	166
9.2.8	High-Speed On-Chip Oscillator Control Register (HOCOOCR)	167
9.2.9	Oscillation Stabilization Flag Register (OSCOVFSR)	168
9.2.10	Oscillation Stop Detection Control Register (OSTDCR)	170
9.2.11	Oscillation Stop Detection Status Register (OSTDSR)	171
9.2.12	Main Clock Oscillator Wait Control Register (MOSCWTCR)	172
9.2.13	CLKOUT Output Control Register (CKOCR)	173
9.2.14	Main Clock Oscillator Forced Oscillation Control Register (MOFCR)	174
9.2.15	Low-Speed On-Chip Oscillator Trimming Register 2 (LOCOTRR2)	175
9.2.16	IWDT-Dedicated On-Chip Oscillator Trimming Register (ILOCOTRR)	175
9.2.17	High-Speed On-Chip Oscillator Trimming Register 0 (HOCOTRR0)	176
9.3	Main Clock Oscillator	177
9.3.1	Connecting a Crystal	177
9.3.2	External Clock Input	178
9.3.3	Handling of Pins When the Main Clock is Not Used	178
9.3.4	Notes on the External Clock Input	178
9.4	Oscillation Stop Detection Function	179
9.4.1	Oscillation Stop Detection and Operation after Detection	179
9.4.2	Oscillation Stop Detection Interrupts	180
9.5	PLL Circuit	181
9.6	Internal Clock	181
9.6.1	System Clock	181
9.6.2	Peripheral Module Clock	181
9.6.3	FlashIF Clock	181
9.6.4	CAC Clock	182
9.6.5	IWDT-Dedicated Clock	182
9.7	Usage Notes	183
9.7.1	Notes on Clock Generation Circuit	183
9.7.2	Note on Rewriting the SCKCR3 Register	183
9.7.3	Notes on Resonator	183
9.7.4	Notes on Board Design	184
9.7.5	Notes on Resonator Connection Pins	184

10.	Clock Frequency Accuracy Measurement Circuit (CAC)	185
10.1	Overview	185
10.2	Register Descriptions	187
10.2.1	CAC Control Register 0 (CACR0)	187
10.2.2	CAC Control Register 1 (CACR1)	188
10.2.3	CAC Control Register 2 (CACR2)	189
10.2.4	CAC Interrupt Request Enable Register (CAICR)	190
10.2.5	CAC Status Register (CASTR)	191
10.2.6	CAC Upper-Limit Value Setting Register (CAULVR)	192
10.2.7	CAC Lower-Limit Value Setting Register (CALLVR)	192
10.2.8	CAC Counter Buffer Register (CACNTBR)	192
10.3	Operation	193
10.3.1	Measuring Clock Frequency	193
10.3.2	Digital Filtering of Signals on the CACREF Pin	194
10.4	Interrupt Requests	194
10.5	Usage Notes	195
10.5.1	Module Stop Function Setting	195
11.	Low Power Consumption	196
11.1	Overview	196
11.2	Register Descriptions	200
11.2.1	Standby Control Register (SBYCR)	200
11.2.2	Module Stop Control Register A (MSTPCRA)	201
11.2.3	Module Stop Control Register B (MSTPCRB)	202
11.2.4	Module Stop Control Register C (MSTPCRC)	203
11.2.5	Operating Power Control Register (OPCCR)	204
11.3	Reducing Power Consumption by Switching Clock Signals	208
11.4	Module Stop Function	208
11.5	Function for Lower Operating Power Consumption	208
11.5.1	Setting Operating Power Control Mode	208
11.6	Low Power Consumption Modes	210
11.6.1	Sleep Mode	210
11.6.1.1	Entry to Sleep Mode	210
11.6.1.2	Exit from Sleep Mode	211
11.6.2	Deep Sleep Mode	212
11.6.2.1	Entry to Deep Sleep Mode	212
11.6.2.2	Exit from Deep Sleep Mode	213
11.6.3	Software Standby Mode	214
11.6.3.1	Entry to Software Standby Mode	214
11.6.3.2	Exit from Software Standby Mode	215
11.6.3.3	Example of Software Standby Mode Application	216
11.7	Usage Notes	217

11.7.1	I/O Port States	217
11.7.2	Module Stop State of DTC	217
11.7.3	On-Chip Peripheral Module Interrupts	217
11.7.4	Write Access to MSTPCRA, MSTPCRB, and MSTPCRC	217
11.7.5	Timing of WAIT Instructions	217
11.7.6	Rewrite the Register by DTC in Sleep Mode	217
12.	Register Write Protection Function	218
12.1	Register Descriptions	219
12.1.1	Protect Register (PRCR)	219
13.	Exception Handling	220
13.1	Exception Events	220
13.1.1	Undefined Instruction Exception	221
13.1.2	Privileged Instruction Exception	221
13.1.3	Floating-Point Exception	221
13.1.4	Reset	221
13.1.5	Non-Maskable Interrupt	221
13.1.6	Interrupt	221
13.1.7	Unconditional Trap	221
13.2	Exception Handling Procedure	222
13.3	Acceptance of Exception Events	224
13.3.1	Acceptance Timing and Saved PC Value	224
13.3.2	Vector and Site for Saving the Values in the PC and PSW	225
13.4	Hardware Processing for Accepting and Returning from Exceptions	226
13.5	Hardware Pre-Processing	227
13.5.1	Undefined Instruction Exception	227
13.5.2	Privileged Instruction Exception	227
13.5.3	Floating-Point Exception	227
13.5.4	Reset	227
13.5.5	Non-Maskable Interrupt	227
13.5.6	Interrupt	228
13.5.7	Unconditional Trap	228
13.6	Return from Exception Handling Routine	229
13.7	Priority of Exception Events	229
14.	Interrupt Controller (ICUb)	230
14.1	Overview	230
14.2	Register Descriptions	232
14.2.1	Interrupt Request Register n (IRn) (n = interrupt vector number)	232
14.2.2	Interrupt Request Enable Register m (IERm) (m = 02h to 1Fh)	233
14.2.3	Interrupt Source Priority Register n (IPRn) (n = interrupt vector number)	234
14.2.4	Fast Interrupt Set Register (FIR)	235
14.2.5	Software Interrupt Generation Register (SWINTR)	236

14.2.6	DTC Transfer Request Enable Register n (DTCERn) (n = interrupt vector number)	237
14.2.7	IRQ Control Register i (IRQCRi) (i = 0 to 7)	238
14.2.8	IRQ Pin Digital Filter Enable Register 0 (IRQFLTE0)	239
14.2.9	IRQ Pin Digital Filter Setting Register 0 (IRQFLTC0)	240
14.2.10	Non-Maskable Interrupt Status Register (NMISR)	241
14.2.11	Non-Maskable Interrupt Enable Register (NMIER)	243
14.2.12	Non-Maskable Interrupt Status Clear Register (NMICLR)	244
14.2.13	NMI Pin Interrupt Control Register (NMICR)	245
14.2.14	NMI Pin Digital Filter Enable Register (NMIFLTE)	245
14.2.15	NMI Pin Digital Filter Setting Register (NMIFLTC)	246
14.3	Vector Table	247
14.3.1	Interrupt Vector Table	247
14.3.2	Fast Interrupt Vector Table	253
14.3.3	Non-maskable Interrupt Vector Area	253
14.4	Interrupt Operation	254
14.4.1	Detecting Interrupts	254
14.4.1.1	Operation of Status Flags for Edge-Detected Interrupts	254
14.4.1.2	Operation of Status Flags for Level-Detected Interrupts	256
14.4.2	Enabling and Disabling Interrupt Sources	257
14.4.3	Selecting Interrupt Request Destinations	258
14.4.4	Determining Priority	259
14.4.5	Multiple Interrupts	259
14.4.6	Fast Interrupt	259
14.4.7	Digital Filter	260
14.4.8	External Pin Interrupts	260
14.5	Non-maskable Interrupt Operation	261
14.6	Return from Low Power Consumption States	262
14.6.1	Return from Sleep Mode or Deep Sleep Mode	262
14.6.2	Return from Software Standby Mode	262
14.7	Usage Note	263
14.7.1	Note on WAIT Instruction Used with Non-Maskable Interrupt	263
15.	Buses	264
15.1	Overview	264
15.2	Description of Buses	266
15.2.1	CPU Buses	266
15.2.2	Memory Buses	266
15.2.3	Internal Main Buses	266
15.2.4	Internal Peripheral Buses	267
15.2.5	Write Buffer Function (Internal Peripheral Bus)	268
15.2.6	Parallel Operation	269

15.2.7	Restrictions	269
15.3	Register Descriptions	270
15.3.1	Bus Error Status Clear Register (BERCLR)	270
15.3.2	Bus Error Monitoring Enable Register (BEREN)	270
15.3.3	Bus Error Status Register 1 (BERSR1)	271
15.3.4	Bus Error Status Register 2 (BERSR2)	271
15.3.5	Bus Priority Control Register (BUSPRI)	272
15.4	Bus Error Monitoring Section	274
15.4.1	Type of Bus Error	274
15.4.1.1	Illegal Address Access	274
15.4.1.2	Timeout	274
15.4.2	Operations When a Bus Error Occurs	274
15.4.3	Conditions Leading to Bus Errors	275
16.	Data Transfer Controller (DTCb)	276
16.1	Overview	276
16.2	Register Descriptions	278
16.2.1	DTC Mode Register A (MRA)	278
16.2.2	DTC Mode Register B (MRB)	280
16.2.3	DTC Mode Register C (MRC)	282
16.2.4	DTC Transfer Source Register (SAR)	283
16.2.5	DTC Transfer Destination Register (DAR)	283
16.2.6	DTC Transfer Count Register A (CRA)	284
16.2.7	DTC Transfer Count Register B (CRB)	285
16.2.8	DTC Control Register (DTCCR)	285
16.2.9	DTC Vector Base Register (DTCVBR)	286
16.2.10	DTC Address Mode Register (DTCADM0D)	286
16.2.11	DTC Module Start Register (DTCST)	287
16.2.12	DTC Status Register (DTCSTS)	288
16.2.13	DTC Index Table Base Register (DTCIBR)	289
16.2.14	DTC Operation Register (DTCOR)	290
16.2.15	DTC Sequence Transfer Enable Register (DTCSQE)	291
16.2.16	DTC Address Displacement Register (DTCDISP)	291
16.3	Request Sources	292
16.3.1	Allocating Transfer Information and DTC Vector Table	292
16.4	Operation	294
16.4.1	Transfer Information Read Skip Function	296
16.4.2	Transfer Information Write-Back Skip Function	297
16.4.2.1	Write-Back Skip by Fixing Addresses	297
16.4.2.2	Write-Back Skip by the MRA.WBDIS Bit	297
16.4.3	Normal Transfer Mode	298
16.4.4	Repeat Transfer Mode	299

16.4.5	Block Transfer Mode	300
16.4.6	Chain Transfer	301
16.4.7	Operation Timing	302
16.4.8	Execution Cycles of the DTC	305
16.4.9	DTC Bus Mastership Release Timing	305
16.4.10	Sequence Transfer	306
16.4.11	DTC Index Table	308
16.4.12	Example of Sequence Transfer	310
16.5	DTC Setting Procedure	316
16.6	Examples of DTC Usage	317
16.6.1	Normal Transfer	317
16.6.2	Chain Transfer When the Counter is 0	318
16.6.3	Sequence Transfer	319
16.7	Interrupt Source	320
16.8	Low Power Consumption Function	321
16.9	Usage Notes	322
16.9.1	Start Address of Transfer Information	322
16.9.2	Allocating Transfer Information	322
16.9.3	Notes on Using the Sequence Transfer	323
17.	I/O Ports	324
17.1	Overview	324
17.2	I/O Port Configuration	326
17.3	Register Descriptions	331
17.3.1	Port Direction Register (PDR)	331
17.3.2	Port Output Data Register (PODR)	332
17.3.3	Port Input Data Register (PIDR)	333
17.3.4	Port Mode Register (PMR)	334
17.3.5	Open Drain Control Register 0 (ODR0)	335
17.3.6	Open Drain Control Register 1 (ODR1)	336
17.3.7	Pull-Up Control Register (PCR)	337
17.4	Initialization of the Port Direction Register (PDR)	338
17.5	Handling of Unused Pins	341
18.	Multi-Function Pin Controller (MPC)	342
18.1	Overview	342
18.2	Register Descriptions	350
18.2.1	Write-Protect Register (PWPR)	350
18.2.2	P0n Pin Function Control Register (P0nPFS) (n = 0 to 4)	351
18.2.3	P1n Pin Function Control Register (P1nPFS) (n = 0 to 5)	352
18.2.4	P2n Pin Function Control Register (P2nPFS) (n = 2 to 6)	353
18.2.5	P3n Pin Function Control Register (P3nPFS) (n = 0 to 3, 6, 7)	354
18.2.6	P4n Pin Function Control Register (P4nPFS) (n = 0 to 7)	355

18.2.7	P7n Pin Function Control Register (P7nPFS) (n = 0 to 6)	356
18.2.8	P9n Pin Function Control Register (P9nPFS) (n = 0 to 7)	357
18.2.9	PBn Pin Function Control Register (PBnPFS) (n = 0 to 7)	358
18.2.10	PDn Pin Function Control Register (PDnPFS) (n = 3 to 7)	359
18.2.11	PEn Pin Function Control Register (PEnPFS) (n = 2)	360
18.2.12	PGn Pin Function Control Register (PGnPFS) (n = 7)	361
18.3	Usage Notes	362
18.3.1	Procedure for Specifying Input/Output Pin Function	362
18.3.2	Notes on Pmn Pin Function Control Register (PmnPFS) Setting	362
18.3.3	Note on Using Analog Functions	364
18.3.4	Note on P97 Pin Input Level	364
18.3.5	Notes on Controlling the Switch to General I/O Port Pin Operation by POE3	364
18.3.6	Notes on Inverting Input/Output Function for the MTU and GPTW Input/Output Pins	365
19.	Multi-Function Timer Pulse Unit 3 (MTU3c)	366
19.1	Overview	366
19.2	Register Descriptions	371
19.2.1	Timer Control Register (TCR)	371
19.2.2	Timer Control Register 2 (TCR2)	373
19.2.3	Timer Mode Register 1 (TMDR1)	377
19.2.4	Timer Mode Register 2A (TMDR2A)	379
19.2.5	Timer Mode Register 3 (TMDR3)	380
19.2.6	Timer I/O Control Register (TIOR)	381
19.2.7	Timer Compare Match Clear Register (TCNTCMPCLR)	392
19.2.8	Timer Interrupt Enable Register (TIER)	393
19.2.9	Timer Status Register (TSR)	396
19.2.10	Timer Buffer Operation Transfer Mode Register (TBTM)	397
19.2.11	Timer Input Capture Control Register (TICCR)	398
19.2.12	Timer Counter (TCNT)	399
19.2.13	Timer Longword Counter (TCNTLW)	399
19.2.14	Timer General Register m (TGRm) (m = A, B, C, D, E, F, U, V, W)	400
19.2.15	Timer Longword General Register m (TGRmLW) (m = A, B)	401
19.2.16	Timer Start Registers (TSTRA, TSTR)	402
19.2.17	Timer Synchronous Register A (TSYRA)	404
19.2.18	Timer Counter Synchronous Start Register (TCSYSTR)	405
19.2.19	Timer Read/Write Enable Register A (TRWERA)	406
19.2.20	Timer Output Master Enable Register A (TOERA)	407
19.2.21	Timer Output Control Register 1A (TOCR1A)	408
19.2.22	Timer Output Control Register 2A (TOCR2A)	410
19.2.23	Timer Output Level Buffer Register A (TOLBRA)	413
19.2.24	Timer Gate Control Register A (TGCRA)	414
19.2.25	Timer Subcounter A (TCNTSA)	415

19.2.26	Timer Period Data Register A (TCDRA)	415
19.2.27	Timer Period Buffer Register A (TCBRA)	416
19.2.28	Timer Dead Time Data Register A (TDDRA)	416
19.2.29	Timer Dead Time Enable Register A (TDERA)	417
19.2.30	Timer Buffer Transfer Set Register A (TBTERA)	418
19.2.31	Timer Waveform Control Register A (TWCRA)	419
19.2.32	Noise Filter Control Register n (NFCRn) (n = 0 to 4, C)	420
19.2.33	Noise Filter Control Register 5 (NFCR5)	422
19.2.34	Timer A/D Conversion Start Request Control Register (TADCR)	423
19.2.35	Timer A/D Conversion Start Request Cycle Set Register m (TADCORm) (m = A, B)	425
19.2.36	Timer A/D Conversion Start Request Cycle Set Buffer Register m (TADCOBRm) (m = A, B)	425
19.2.37	Timer Interrupt Skipping Mode Register A (TITMRA)	426
19.2.38	Timer Interrupt Skipping Set Register 1A (TITCR1A)	427
19.2.39	Timer Interrupt Skipping Counter 1A (TITCNT1A)	428
19.2.40	Timer Interrupt Skipping Set Register 2A (TITCR2A)	429
19.2.41	Timer Interrupt Skipping Counter 2A (TITCNT2A)	430
19.2.42	A/D Conversion Start Request Select Register 0 (TADSTRGR0)	431
19.2.43	A/D Conversion Start Request Select Register 1 (TADSTRGR1)	431
19.3	Operation	433
19.3.1	Basic Functions	433
19.3.2	Synchronous Operation	439
19.3.3	Buffer Operation	441
19.3.4	Cascaded Operation	445
19.3.5	PWM Modes	450
19.3.6	Phase Counting Mode	454
19.3.6.1	16-Bit Phase Counting Mode	454
19.3.6.2	Cascade Connection 32-Bit Phase Counting Mode	465
19.3.7	Reset-Synchronized PWM Mode	466
19.3.8	Complementary PWM Mode	469
19.3.9	A/D Conversion Start Request Delaying Function	504
19.3.10	Synchronous Operation of MTU0 to MTU4	510
19.3.11	External Pulse Width Measurement	511
19.3.12	Dead Time Compensation	512
19.3.13	TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode	514
19.3.14	Noise Filter Function	515
19.3.15	A/D Conversion Start Request Frame Synchronization Signal	516
19.4	Interrupt Sources	517
19.4.1	Interrupt Sources and Priorities	517
19.4.2	DTC Trigger Sources	518
19.4.3	A/D Converter Trigger Sources	519

19.5	Operation Timing	521
19.5.1	Input/Output Timing	521
19.5.2	Interrupt Signal Timing	527
19.6	Usage Notes	530
19.6.1	Module Stop Function Setting	530
19.6.2	Count Clock Restrictions	530
19.6.3	Note on Period Setting	530
19.6.4	Contention between TCNT Write and Clear Operations	531
19.6.5	Contention between TCNT Write and Increment Operations	531
19.6.6	Contention between TGR Write Operation and Compare Match	532
19.6.7	Contention between Buffer Register Write Operation and Compare Match	532
19.6.8	Contention between Buffer Register Write and TCNT Clear Operations	533
19.6.9	Contention between TGR Read Operation and Input Capture	533
19.6.10	Contention between TGR Write Operation and Input Capture	534
19.6.11	Contention between Buffer Register Write Operation and Input Capture	535
19.6.12	Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation	536
19.6.13	Counter Value When Count Operation is Stopped in Complementary PWM Mode	537
19.6.14	Buffer Operation Setting in Complementary PWM Mode	537
19.6.15	Buffer Operation and Compare Match in Reset-Synchronized PWM Mode	538
19.6.16	Overflow in Reset-Synchronized PWM Mode	539
19.6.17	Contention between Overflow/Underflow and Counter Clearing	540
19.6.18	Contention between TCNT Write Operation and Overflow/Underflow	540
19.6.19	Note on Transition from Normal Mode or PWM Mode 1 to Reset-Synchronized PWM Mode	541
19.6.20	Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode	541
19.6.21	Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection	541
19.6.22	Interrupt Skipping Function 2	542
19.6.23	Notes When Complementary PWM Mode Output Protection Function is Not Used	542
19.6.24	Notes Regarding Timer Counter (MTU5.TCNT) and Timer General Register (MTU5.TGR)	542
19.6.25	Notes to Prevent Malfunctions in Synchronous Clearing for Complementary PWM Mode	543
19.6.26	Continuous Output of Interrupt Signal in Response to a Compare Match	545
19.6.27	Usage Notes on A/D Conversion Start Request Delaying Function in Complementary PWM Mode	545
19.7	MTU Output Pin Initialization	547
19.7.1	Operating Modes	547
19.7.2	Operation in Case of Re-Setting Due to Error during Operation	547
19.7.3	Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation	548
20.	Port Output Enable 3 (POE3E)	574
20.1	Overview	574

20.2	Register Descriptions	578
20.2.1	Input Level Control/Status Register 1 (ICSR1)	578
20.2.2	Input Level Control/Status Register 3 (ICSR3)	580
20.2.3	Input Level Control/Status Register 4 (ICSR4)	582
20.2.4	Input Level Control/Status Register 5 (ICSR5)	584
20.2.5	Input Level Control/Status Register 6 (ICSR6)	586
20.2.6	Input Level Control/Status Register 7 (ICSR7)	587
20.2.7	Output Level Control/Status Register 1 (OCSR1)	589
20.2.8	Output Level Control/Status Register 3 (OCSR3)	590
20.2.9	Active Level Setting Register 1 (ALR1)	591
20.2.10	Active Level Setting Register 3 (ALR3)	593
20.2.11	Software Port Output Enable Register (SPOER)	594
20.2.12	Port Output Enable Control Register 1 (POECR1)	596
20.2.13	Port Output Enable Control Register 2 (POECR2)	597
20.2.14	Port Output Enable Control Register 3 (POECR3)	598
20.2.15	Port Output Enable Control Register 4 (POECR4)	599
20.2.16	Port Output Enable Control Register 5 (POECR5)	601
20.2.17	Port Output Enable Control Register 6 (POECR6)	603
20.2.18	Port Output Enable Control Register 6B (POECR6B)	605
20.2.19	Port Output Enable Control Register 9 (POECR9)	607
20.2.20	Port Mode Mask Control Register 0 (PMMCR0)	609
20.2.21	Port Mode Mask Control Register 1 (PMMCR1)	611
20.2.22	Port Mode Mask Control Register 2 (PMMCR2)	613
20.2.23	Port Output Enable Comparator Output Detection Flag Register (POECMPFR)	615
20.2.24	Port Output Enable Comparator Request Select Register (POECMPSEL)	616
20.2.25	Port Output Enable Comparator Request Extended Selection Register m (POECMPEXm) (m = 0, 2, 4 to 6)	617
20.2.26	MTU0 Pin Select Register 1 (M0SELR1)	618
20.2.27	MTU0 Pin Select Register 2 (M0SELR2)	619
20.2.28	MTU3 Pin Select Register (M3SELR)	620
20.2.29	MTU4 Pin Select Register 1 (M4SELR1)	621
20.2.30	MTU4 Pin Select Register 2 (M4SELR2)	622
20.2.31	GPTW0 Pin Select Register (G0SELR)	623
20.2.32	GPTW1 Pin Select Register (G1SELR)	624
20.2.33	GPTW2 Pin Select Register (G2SELR)	625
20.2.34	Input Signal Mask Control Register 0 (IMCR0)	626
20.2.35	Input Signal Mask Control Register 2 (IMCR2)	627
20.2.36	Input Signal Mask Control Register 3 (IMCR3)	628
20.2.37	Input Signal Mask Control Register 4 (IMCR4)	629
20.2.38	Input Signal Mask Control Register 5 (IMCR5)	630
20.2.39	Input Signal Mask Control Register 9 (IMCR9)	631

20.2.40	Input Signal Mask Control Register 10 (IMCR10)	632
20.2.41	Input Signal Mask Control Register 11 (IMCR11)	633
20.3	Operation	634
20.3.1	MTU/GPTW Pin Selection	649
20.3.2	Input-Level Detection Operation	650
20.3.3	Output-Level Compare Operation	651
20.3.4	Output Disabling Control Using Registers	652
20.3.5	Output Disabling Control through Detection of Oscillation Stop	652
20.3.6	Output Disabling Control through Detection of the Comparator Output	652
20.3.7	Additional Functions for Output Disabling Control	652
20.3.8	Masking of Output Disabling Control Signals	653
20.3.9	Control in Response to an Output Disabling Request	653
20.3.10	Recover from Output Disabled State	653
20.4	POE Setting Procedure	655
20.5	Interrupts	657
20.6	Usage Notes	658
20.6.1	Transition to Low Power Consumption Mode	658
20.6.2	Output Disabling Control When the MTU and GPTW Pins are Not Selected	658
20.6.3	When the POE is Not Used	658
20.6.4	Setting the Active Level When the Inverting Output Is Set for the MTU and GPTW	658
20.6.5	Reading the Pins When in High-Impedance State	658
20.6.6	Note for Using POE and POEG Together	658
20.6.7	Notes on Using POE with the INV Bit Set to 1	659
21.	General PWM Timer (GPTWd)	660
21.1	Overview	660
21.2	Register Descriptions	664
21.2.1	General PWM Timer Write-Protection Register (GTWP)	664
21.2.2	General PWM Timer Software Start Register (GTSTR)	667
21.2.3	General PWM Timer Software Stop Register (GTSTP)	668
21.2.4	General PWM Timer Software Clear Register (GTCLR)	669
21.2.5	General PWM Timer Start Source Select Register (GTSSR)	670
21.2.6	General PWM Timer Stop Source Select Register (GTPSR)	673
21.2.7	General PWM Timer Clear Source Select Register (GTCSR)	675
21.2.8	General PWM Timer Input Capture Source Select Register A (GTICASR)	678
21.2.9	General PWM Timer Input Capture Source Select Register B (GTICBSR)	681
21.2.10	General PWM Timer Control Register (GTCCR)	684
21.2.11	General PWM Timer Count Direction and Duty Setting Register (GTUDDTYC)	687
21.2.12	General PWM Timer I/O Control Register (GTIOR)	690
21.2.13	General PWM Timer Interrupt Output Setting Register (GTINTAD)	694
21.2.14	General PWM Timer Status Register (GTST)	698
21.2.15	General PWM Timer Buffer Enable Register (GTBER)	704

21.2.16	General PWM Timer Interrupt and A/D Conversion Start Request Skipping Setting Register (GTITC)	709
21.2.17	General PWM Timer Counter (GTCNT)	711
21.2.18	General PWM Timer Compare Capture Register m (GTCCRm) (m = A to F)	712
21.2.19	General PWM Timer Period Setting Register (GTPR)	713
21.2.20	General PWM Timer Period Setting Buffer Register (GTPBR)	713
21.2.21	General PWM Timer Period Setting Double-Buffer Register (GTPDBR)	714
21.2.22	A/D Conversion Start Request Timing Register m (GTADTRm) (m = A, B)	714
21.2.23	A/D Conversion Start Request Timing Buffer Register m (GTADTBRm) (m = A, B)	715
21.2.24	A/D Conversion Start Request Timing Double-Buffer Register m (GTADTDBRm) (m = A, B)	715
21.2.25	General PWM Timer Dead Time Control Register (GTDTCR)	716
21.2.26	General PWM Timer Dead Time Value Register m (GTDVm) (m = U, D)	718
21.2.27	General PWM Timer Dead Time Value Buffer Register m (GTDBm) (m = U, D)	719
21.2.28	General PWM Timer Output Protection Function Status Register (GTSOS)	720
21.2.29	General PWM Timer Output Protection Function Temporary Release Register (GTSOTR)	721
21.2.30	General PWM Timer A/D Conversion Start Request Signal Monitoring Register (GTADSMR)	722
21.2.31	General PWM Timer Operation Enable Bit Simultaneous Control Channel Select Register (GTSECSR)	724
21.2.32	General PWM Timer Operation Enable Bit Simultaneous Control Register (GTSECR)	725
21.2.33	General PWM Timer Buffer Enable Register 2 (GTBER2)	728
21.2.34	General PWM Timer Inter Channel Cooperation Input Capture Control Register (GTICCR)	732
21.3	Operation	736
21.3.1	Basic Operation	736
21.3.1.1	Counter Operation	736
21.3.1.2	Waveform Output by Compare Match	740
21.3.1.3	Input Capture Function	744
21.3.2	Buffer Operation	748
21.3.2.1	GTPR Register Buffer Operation	748
21.3.2.2	Buffer Operation for the GTCCRA and GTCCRB Registers	751
21.3.2.3	Buffer Operation for the GTADTRA and GTADTRB Registers	757
21.3.3	PWM Output Operating Mode	761
21.3.4	Automatic Dead Time Setting Function	774
21.3.5	Count Direction Changing Function	779
21.3.6	Duty Cycle 0%/100% Output Function	780
21.3.7	Hardware Count Start/Count Stop and Clear Operation	782
21.3.7.1	Hardware Start Operation	782
21.3.7.2	Hardware Stop Operation	785
21.3.7.3	Hardware Clear Operation	789

21.3.7.4	MTU Synchronous Count Start Operation	792
21.3.8	Synchronous Operation on Channels	793
21.3.8.1	Synchronous Operation by Software	793
21.3.8.2	Synchronous Operation by Hardware Source	795
21.3.8.3	Synchronous Clear Operation by Inter Channel Cooperation	797
21.3.8.4	Input Capture Operation by Inter Channel Cooperation	801
21.3.8.5	Synchronous Operation by Phase Shift Start	803
21.3.9	PWM Output Operation Examples	804
21.4	Interrupt Sources	810
21.4.1	Interrupt Sources and Priorities	810
21.4.2	DTC Activation	813
21.4.3	Interrupt and A/D Conversion Start Request Skipping Function	813
21.4.3.1	Interrupt Skipping Function by GTITC Register	813
21.5	A/D Conversion Start Request	817
21.6	Noise Filter Function	820
21.7	Protection Function	821
21.7.1	Write-Protection for Registers	821
21.7.2	Disabling of Buffer Operation	821
21.7.2.1	Simultaneous Control of Buffer Operations of Multiple Channels	823
21.7.2.2	Repeated Double-Buffered Operation When Disabling GTCCR Buffer Transfer	826
21.7.3	GTIOCNm Pin Output Negate Control (n = 0 to 2; m = A, B)	830
21.7.4	Output Protection Function for GTIOCNm Pin Output (n = 0 to 2; m = A, B)	831
21.8	Initialization Method of Output Pins	837
21.8.1	Pin Settings after Reset	837
21.8.2	Pin Initialization Due to Error during Operation	837
21.9	Usage Notes	838
21.9.1	Module Stop Function Setting	838
21.9.2	Settings of the GTCCRM Register during Compare Match Operation (m = A to F)	838
21.9.3	Setting Range of the GTCNT Counter	839
21.9.4	The GTCNT Counter Start/Stop	839
21.9.5	Order of Priority in Events	839
22.	GPTW Port Output Enable (POEGb)	841
22.1	Overview	841
22.2	Register Descriptions	844
22.2.1	POEG Group A Setting Register (POEGGA)	844
22.2.2	POEG Group A Input Control Register (POEGICRA)	846
22.2.3	GPTW Output Disabling Control Group A Write Protection Register (GTONCWA)	847
22.2.4	GPTW Output Disabling Control Group A Controlling Register (GTONCCA) (A = A to D)	848
22.3	Operation	849

22.3.1	Output Disabling Request by Detection of Input Level or Input Edge on the GTETRGA Pin	849
22.3.1.1	Digital Noise Filter	849
22.3.2	Output Disabling Request by Output Disabling Condition Detection Signal from GPTW	850
22.3.3	Output Disabling Request by Comparator Detection Signals	851
22.3.4	Output Disabling Request by Oscillation Stop Detection Signal	851
22.3.5	Output Disabling Request by a Register	851
22.3.6	Canceling Output Disabling Request	852
22.3.6.1	Cancellation of Output Disabling Request by Reset	852
22.3.6.2	Cancellation of Output Disabling Request by Clearing Flags in the POEGGA Register	852
22.3.6.3	Cancellation of Output Disabling Request by Negating Level Detection Signal	853
22.3.7	Direct Output Disabling Request by Input Signal and Canceling the Request	854
22.4	Interrupt Sources	856
22.5	External Trigger Output to GPTW	856
22.6	Usage Notes	857
22.6.1	Transitions to Low-Power Consumption Mode	857
22.6.2	Setting the Function for Stopping the Module	857
22.6.3	Duplication of Output Disabling Request	857
22.6.4	Masking of Output Disabling Request by COMPn Edge Detection Signal	857
23.	8-Bit Timer (TMR)	858
23.1	Overview	858
23.2	Register Descriptions	863
23.2.1	Timer Counter (TCNT)	863
23.2.2	Time Constant Register A (TCORA)	864
23.2.3	Time Constant Register B (TCORB)	864
23.2.4	Timer Control Register (TCR)	865
23.2.5	Timer Counter Control Register (TCCR)	866
23.2.6	Timer Control/Status Register (TCSR)	868
23.3	Operation	870
23.3.1	Pulse Output	870
23.3.2	External Counter Reset Input	871
23.4	Operation Timing	872
23.4.1	TCNT Count Timing	872
23.4.2	Timing of Interrupt Signal Output on a Compare Match	873
23.4.3	Timing of Timer Output Signal at Compare Match	873
23.4.4	Timing of Counter Clear by Compare Match	874
23.4.5	Timing of the External Reset for TCNT	874
23.4.6	Timing of Interrupt Signal Output on an Overflow	875
23.5	Operation with Cascaded Connection	876
23.5.1	16-Bit Count Mode	876
23.5.2	Compare Match Count Mode	876

23.6	Interrupt Sources	877
23.6.1	Interrupt Sources and DTC Activation	877
23.6.2	Startup of the A/D Converter	877
23.7	Usage Notes	878
23.7.1	Module Stop State Setting	878
23.7.2	Notes on Setting Cycle	878
23.7.3	Conflict between TCNT Write and Counter Clear	878
23.7.4	Conflict between TCNT Write and Increment	879
23.7.5	Conflict between TCORA or TCORB Write and Compare Match	879
23.7.6	Conflict between Compare Matches A and B	880
23.7.7	Switching of Internal Clocks and TCNT Operation	880
23.7.8	Clock Source Setting with Cascaded Connection	882
23.7.9	Continuous Output of Compare Match Interrupt Signal	882
24.	Compare Match Timer (CMT)	883
24.1	Overview	883
24.2	Register Descriptions	884
24.2.1	Compare Match Timer Start Register 0 (CMSTR0)	884
24.2.2	Compare Match Timer Control Register (CMCR)	884
24.2.3	Compare Match Counter (CMCNT)	885
24.2.4	Compare Match Constant Register (CMCOR)	885
24.3	Operation	886
24.3.1	Cyclic Count Operation	886
24.3.2	CMCNT Count Timing	886
24.4	Interrupts	887
24.4.1	Interrupt Sources	887
24.4.2	Timing of Compare Match Interrupt Generation	887
24.5	Usage Notes	888
24.5.1	Setting the Module Stop Function	888
24.5.2	Conflict between CMCNT Counter Writing and Compare Match	888
24.5.3	Conflict between CMCNT Counter Writing and Incrementing	888
25.	Independent Watchdog Timer (IWDTa)	889
25.1	Overview	889
25.2	Register Descriptions	891
25.2.1	IWDT Refresh Register (IWDTRR)	891
25.2.2	IWDT Control Register (IWDTCR)	892
25.2.3	IWDT Status Register (IWDTSR)	895
25.2.4	IWDT Reset Control Register (IWDTRCR)	896
25.2.5	IWDT Count Stop Control Register (IWDTCSTPR)	897
25.2.6	Option Function Select Register 0 (OFS0)	897
25.3	Operation	898
25.3.1	Count Operation in Each Start Mode	898

25.3.1.1	Register Start Mode	898
25.3.1.2	Auto-Start Mode	900
25.3.2	Refresh Operation	901
25.3.3	Status Flags	903
25.3.4	Reset Output	903
25.3.5	Interrupt Sources	903
25.3.6	Reading the Counter Value	904
25.3.7	Correspondence between Option Function Select Register 0 (OFS0) and IWDG Registers	905
25.4	Usage Notes	905
25.4.1	Refresh Operations	905
25.4.2	Clock Divide Ratio Setting	905
26.	Serial Communications Interface (SCIg, SCIf)	906
26.1	Overview	906
26.2	Register Descriptions	914
26.2.1	Receive Shift Register (RSR)	914
26.2.2	Receive Data Register (RDR)	914
26.2.3	Receive Data Register H, L, HL (RDRH, RDRL, RDRHL)	915
26.2.4	Transmit Data Register (TDR)	916
26.2.5	Transmit Data Register H, L, HL (TDRH, TDRL, TDRHL)	917
26.2.6	Transmit Shift Register (TSR)	917
26.2.7	Serial Mode Register (SMR)	918
26.2.8	Serial Control Register (SCR)	922
26.2.9	Serial Status Register (SSR)	926
26.2.10	Smart Card Mode Register (SCMR)	931
26.2.11	Bit Rate Register (BRR)	933
26.2.12	Modulation Duty Register (MDDR)	941
26.2.13	Serial Extended Mode Register (SEMR)	942
26.2.14	Noise Filter Setting Register (SNFR)	945
26.2.15	I ² C Mode Register 1 (SIMR1)	946
26.2.16	I ² C Mode Register 2 (SIMR2)	947
26.2.17	I ² C Mode Register 3 (SIMR3)	948
26.2.18	I ² C Status Register (SISR)	950
26.2.19	SPI Mode Register (SPMR)	951
26.2.20	Extended Serial Module Enable Register (ESMER)	952
26.2.21	Control Register 0 (CR0)	953
26.2.22	Control Register 1 (CR1)	953
26.2.23	Control Register 2 (CR2)	954
26.2.24	Control Register 3 (CR3)	955
26.2.25	Port Control Register (PCR)	955
26.2.26	Interrupt Control Register (ICR)	956

26.2.27	Status Register (STR)	957
26.2.28	Status Clear Register (STCR)	958
26.2.29	Control Field 0 Data Register (CF0DR)	958
26.2.30	Control Field 0 Compare Enable Register (CF0CR)	959
26.2.31	Control Field 0 Receive Data Register (CF0RR)	959
26.2.32	Primary Control Field 1 Data Register (PCF1DR)	959
26.2.33	Secondary Control Field 1 Data Register (SCF1DR)	960
26.2.34	Control Field 1 Compare Enable Register (CF1CR)	960
26.2.35	Control Field 1 Receive Data Register (CF1RR)	960
26.2.36	Timer Control Register (TCR)	961
26.2.37	Timer Mode Register (TMR)	961
26.2.38	Timer Prescaler Register (TPRE)	962
26.2.39	Timer Count Register (TCNT)	962
26.3	Operation in Asynchronous Mode	963
26.3.1	Serial Data Transfer Format	963
26.3.2	Receive Data Sampling Timing and Reception Margin in Asynchronous Mode	965
26.3.3	Clock	966
26.3.4	Double-Speed Mode	966
26.3.5	CTS and RTS Functions	967
26.3.6	SCI Initialization (Asynchronous Mode)	968
26.3.7	Serial Data Transmission (Asynchronous Mode)	970
26.3.8	Serial Data Reception (Asynchronous Mode)	974
26.4	Multi-Processor Communications Function	978
26.4.1	Multi-Processor Serial Data Transmission	979
26.4.2	Multi-Processor Serial Data Reception	980
26.5	Operation in Clock Synchronous Mode	983
26.5.1	Clock	983
26.5.2	CTS and RTS Functions	984
26.5.3	SCI Initialization (Clock Synchronous Mode)	985
26.5.4	Serial Data Transmission (Clock Synchronous Mode)	986
26.5.5	Serial Data Reception (Clock Synchronous Mode)	990
26.5.6	Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode)	993
26.6	Operation in Smart Card Interface Mode	994
26.6.1	Sample Connection	994
26.6.2	Data Format (Except in Block Transfer Mode)	995
26.6.3	Block Transfer Mode	996
26.6.4	Receive Data Sampling Timing and Reception Margin	997
26.6.5	SCI Initialization (Smart Card Interface Mode)	998
26.6.6	Serial Data Transmission (Except in Block Transfer Mode)	1000
26.6.7	Serial Data Reception (Except in Block Transfer Mode)	1003
26.6.8	Clock Output Control	1005

26.7	Operation in Simple I ² C Mode	1006
26.7.1	Generation of Start, Restart, and Stop Conditions	1007
26.7.2	Clock Synchronization	1009
26.7.3	SSDA Output Delay	1010
26.7.4	SCI Initialization (Simple I ² C Mode)	1011
26.7.5	Operation in Master Transmission (Simple I ² C Mode)	1012
26.7.6	Master Reception (Simple I ² C Mode)	1014
26.7.7	Recovery from Bus Hang-up	1016
26.8	Operation in Simple SPI Mode	1017
26.8.1	States of Pins in Master and Slave Modes	1018
26.8.2	SS Function in Master Mode	1018
26.8.3	SS Function in Slave Mode	1018
26.8.4	Relationship between Clock and Transmit/Receive Data	1019
26.8.5	SCI Initialization (Simple SPI Mode)	1020
26.8.6	Transmission and Reception of Serial Data (Simple SPI Mode)	1020
26.9	Bit Rate Modulation Function	1020
26.10	Extended Serial Mode Control Section: Description of Operation	1021
26.10.1	Serial Transfer Protocol	1021
26.10.2	Transmitting a Start Frame	1022
26.10.3	Receiving a Start Frame	1025
26.10.3.1	Priority Interrupt Bit	1030
26.10.4	Detection of Bus Collisions	1031
26.10.5	Digital Filter for Input on the RXDX12 Pin	1032
26.10.6	Bit Rate Measurement	1033
26.10.7	Selectable Timing for Sampling Data Received through RXDX12	1034
26.10.8	Timer	1035
26.11	Noise Cancellation Function	1037
26.12	Interrupt Sources	1038
26.12.1	Buffer Operations for TXI and RXI Interrupts	1038
26.12.2	Interrupts in Asynchronous Mode, Clock Synchronous Mode, and Simple SPI Mode	1038
26.12.3	Interrupts in Smart Card Interface Mode	1039
26.12.4	Interrupts in Simple I ² C Mode	1040
26.12.5	Interrupt Requests from the Extended Serial Mode Control Section	1041
26.13	Usage Notes	1042
26.13.1	Setting the Module Stop Function	1042
26.13.2	Break Detection and Processing	1042
26.13.3	Mark State and Sending Breaks	1042
26.13.4	Receive Error Flags and Transmit Operations (Clock Synchronous Mode and Simple SPI Mode)	1042
26.13.5	Writing Data to the TDR Register	1042
26.13.6	Restrictions on Clock Synchronous Transmission (Clock Synchronous Mode and Simple SPI Mode)	1043

26.13.7	Restrictions on Using DTC	1044
26.13.8	Notes on Starting Transfer	1044
26.13.9	SCI Operations during Low Power Consumption State	1044
26.13.10	External Clock Input in Clock Synchronous Mode and Simple SPI Mode	1047
26.13.11	Limitations on Simple SPI Mode	1048
26.13.12	Limitation 1 on Usage of the Extended Serial Mode Control Section	1049
26.13.13	Limitation 2 on Usage of the Extended Serial Mode Control Section	1049
26.13.14	Note on Transmit Enable Bit (TE Bit)	1050
26.13.15	Note on Stopping Reception When Using the RTS Function in Asynchronous Mode	1050
27.	I²C-bus Interface (R11Ca)	1051
27.1	Overview	1051
27.2	Register Descriptions	1053
27.2.1	I ² C-bus Control Register 1 (ICCR1)	1053
27.2.2	I ² C-bus Control Register 2 (ICCR2)	1055
27.2.3	I ² C-bus Mode Register 1 (ICMR1)	1059
27.2.4	I ² C-bus Mode Register 2 (ICMR2)	1060
27.2.5	I ² C-bus Mode Register 3 (ICMR3)	1062
27.2.6	I ² C-bus Function Enable Register (ICFER)	1064
27.2.7	I ² C-bus Status Enable Register (ICSER)	1066
27.2.8	I ² C-bus Interrupt Enable Register (ICIER)	1068
27.2.9	I ² C-bus Status Register 1 (ICSR1)	1070
27.2.10	I ² C-bus Status Register 2 (ICSR2)	1073
27.2.11	Slave Address Register Ly (SARLy) (y = 0 to 2)	1076
27.2.12	Slave Address Register Uy (SARUy) (y = 0 to 2)	1077
27.2.13	I ² C-bus Bit Rate Low-Level Register (ICBRL)	1078
27.2.14	I ² C-bus Bit Rate High-Level Register (ICBRH)	1079
27.2.15	I ² C-bus Transmit Data Register (ICDRT)	1081
27.2.16	I ² C-bus Receive Data Register (ICDRR)	1081
27.2.17	I ² C-bus Shift Register (ICDRS)	1081
27.3	Operation	1082
27.3.1	Communication Data Format	1082
27.3.2	Initial Settings	1083
27.3.3	Master Transmit Operation	1084
27.3.4	Master Receive Operation	1087
27.3.5	Slave Transmit Operation	1093
27.3.6	Slave Receive Operation	1096
27.4	SCL Synchronization Circuit	1098
27.5	SDA Output Delay Function	1099
27.6	Digital Noise Filters	1100
27.7	Address Match Detection	1101
27.7.1	Slave-Address Match Detection	1101

27.7.2	Detection of the General Call Address	1103
27.7.3	Device-ID Address Detection	1104
27.7.4	Host Address Detection	1106
27.8	Automatic Low-Hold Function for SCL	1107
27.8.1	Function to Prevent Wrong Transmission of Transmit Data	1107
27.8.2	NACK Reception Transfer Suspension Function	1108
27.8.3	Function to Prevent Failure to Receive Data	1109
27.9	Arbitration-Lost Detection Functions	1111
27.9.1	Master Arbitration-Lost Detection (MALE Bit)	1111
27.9.2	NACK Transmission Arbitration-Lost Detection Function (NALE Bit)	1113
27.9.3	Slave Arbitration-Lost Detection (SALE Bit)	1114
27.10	Start Condition/Restart Condition/Stop Condition Generating Function	1115
27.10.1	Generating a Start Condition	1115
27.10.2	Generating a Restart Condition	1115
27.10.3	Generating a Stop Condition	1116
27.11	Bus Hanging	1117
27.11.1	Timeout Function	1117
27.11.2	Additional SCL Output Function	1119
27.11.3	RIIC Reset and Internal Reset	1120
27.12	SMBus Operation	1121
27.12.1	SMBus Timeout Measurement	1121
27.12.2	Packet Error Code (PEC)	1122
27.12.3	SMBus Host Notification Protocol (Notify ARP Master Command)	1122
27.13	Interrupt Sources	1123
27.13.1	Buffer Operation for TXI and RXI Interrupts	1123
27.14	Initialization of Registers and Functions When a Reset is Applied or a Condition is Detected	1124
27.15	Usage Notes	1125
27.15.1	Setting Module Stop Function	1125
27.15.2	Notes on Starting Transfer	1125
28.	CRC Calculator (CRCA)	1126
28.1	Overview	1126
28.2	Register Descriptions	1127
28.2.1	CRC Control Register (CRCCR)	1127
28.2.2	CRC Data Input Register (CRCDIR)	1128
28.2.3	CRC Data Output Register (CRCDOR)	1129
28.3	Operation	1130
28.4	Usage Notes	1133
28.4.1	Module Stop Function Setting	1133
28.4.2	Note on Transmission	1133
29.	Arithmetic Unit for Trigonometric Functions (TFUV1)	1134
29.1	Overview	1134

30.	12-Bit A/D Converter (S12ADF)	1135
30.1	Overview	1135
30.2	Register Descriptions	1142
30.2.1	A/D Data Registers y (ADDRy) (y = 0 to 7), A/D Data Duplication Register (ADDBLDR), A/D Data Duplication Register A (ADDBLDRA), A/D Data Duplication Register B (ADDBLDRB), A/D Temperature Sensor Data Register (ADTSDR), A/D Internal Reference Voltage Data Register (ADOCDR)	1142
30.2.2	A/D Self-Diagnosis Data Register (ADRD)	1144
30.2.3	A/D Control Register (ADCSR)	1145
30.2.4	A/D Channel Select Register A0 (ADANSA0)	1149
30.2.5	A/D Channel Select Register B0 (ADANSB0)	1151
30.2.6	A/D Channel Select Register C0 (ADANSC0)	1153
30.2.7	A/D-Converted Value Addition/Average Function Channel Select Register 0 (ADADS0) ..	1155
30.2.8	A/D-Converted Value Addition/Average Count Select Register (ADADC)	1159
30.2.9	A/D Control Extended Register (ADCER)	1160
30.2.10	A/D Conversion Start Trigger Select Register (ADSTRGR)	1162
30.2.11	A/D Conversion Extended Input Control Register (ADEXICR)	1164
30.2.12	A/D Group C Trigger Select Register (ADGCTRGR)	1166
30.2.13	A/D Sampling State Register n (ADSSTRn) (n = 0 to 7, T, O)	1169
30.2.14	A/D Disconnection Detection Control Register (ADDISCR)	1171
30.2.15	A/D Group Scan Priority Control Register (ADGSPCR)	1172
30.2.16	A/D High-Potential/Low-Potential Reference Voltage Control Register (ADHVREFCNT)	1174
30.3	Operation	1175
30.3.1	A/D Converter Initialization Flowchart	1175
30.3.2	A/D Conversion Units and Analog Inputs	1176
30.3.3	Scanning Operation	1176
30.3.4	Single Scan Mode	1178
30.3.4.1	Basic Operation	1178
30.3.4.2	Channel Selection and Self-Diagnosis	1179
30.3.4.3	A/D Conversion of Temperature Sensor Output/Internal Reference Voltage	1180
30.3.4.4	A/D Conversion in Double Trigger Mode	1181
30.3.4.5	A/D Conversion in Extended Double Trigger Mode	1182
30.3.5	Continuous Scan Mode	1186
30.3.5.1	Basic Operation	1186
30.3.5.2	Channel Selection and Self-Diagnosis	1187
30.3.6	Group Scan Mode	1188
30.3.6.1	Basic Operation	1188
30.3.6.2	A/D Conversion in Double Trigger Mode	1189
30.3.6.3	Operation under Group Priority Control	1191
30.3.7	Time Interleaved Operation	1208

30.3.7.1	Trigger Input Timing	1208
30.3.7.2	Single Scan Mode	1210
30.3.7.3	Single Scan Mode with Extended Double Trigger Mode	1211
30.3.8	Analog Input Sampling Time and Scan Conversion Time	1213
30.3.8.1	Timing of Suspension and Starting of Scanning in Operation under Group Priority Control	1216
30.3.9	Usage Example of A/D Data Register Automatic Clearing Function	1217
30.3.10	A/D-Converted Value Addition/Average Mode	1217
30.3.11	Disconnection Detection Assist Function	1217
30.3.12	Starting A/D Conversion with Asynchronous Trigger	1219
30.3.13	Starting A/D Conversion with Synchronous Trigger from Peripheral Module	1219
30.3.14	Programmable Gain Amplifier	1219
30.4	Interrupt Sources and DTC Transfer Requests	1220
30.4.1	Interrupt Requests	1220
30.5	Selecting Reference Voltage	1221
30.6	Allowable Impedance of Signal Source	1221
30.7	Usage Notes	1222
30.7.1	Notes on Reading Data Registers	1222
30.7.2	Notes on Stopping A/D Conversion	1222
30.7.2.1	Procedure of Stopping A/D Conversion	1222
30.7.2.2	Notes on Modes and Status Flags	1223
30.7.3	A/D Conversion Restarting Timing and Termination Timing	1223
30.7.4	Notes on Scan End Interrupt Handling	1223
30.7.5	Module Stop Function Setting	1223
30.7.6	Notes on Entering Low Power Consumption States	1223
30.7.7	Notes on Canceling Software Standby Mode	1223
30.7.8	Pin Setting when the 12-bit A/D Converter is Used	1223
30.7.9	Error in Absolute Accuracy When Disconnection Detection Assistance is in Use	1224
30.7.10	Voltage Range of Analog Power Supply Pins	1224
30.7.11	Notes on Board Design	1225
30.7.12	Notes on Noise Prevention	1226
31.	Programmable Gain Amplifier (PGA)	1227
31.1	Overview	1227
31.2	Register Descriptions	1229
31.2.1	Programmable Gain Amplifier Control Register (PGACR)	1229
31.2.2	Programmable Gain Amplifier Gain Setting Register 0 (PGAGS0)	1230
31.2.3	Programmable Gain Amplifier Monitor Output Control Register (PGAMONCR)	1231
31.3	Operation	1232
31.4	Usage Notes	1234
31.4.1	Restrictions on Programmable Gain Amplifier Monitor Output Function	1234
31.4.2	Notes on Using the PGAGND Pin	1234

31.4.3	Module Stop Function Setting	1234
31.4.4	Programmable Gain Amplifier Operation in Module Stop State	1234
31.4.5	Programmable Gain Amplifier Operation in Software Standby Mode	1234
32.	D/A Converter (DAa)	1235
32.1	Overview	1235
32.2	Register Descriptions	1236
32.2.1	D/A Data Register m (DADRM) (m = 0, 1)	1236
32.2.2	D/A Control Register (DACR)	1237
32.2.3	Data Register Format Select Register (DADPR)	1237
32.2.4	D/A A/D Synchronous Start Control Register (DAADSCR)	1238
32.3	Operation	1239
32.3.1	Measure against Interference between D/A and A/D Conversion	1240
32.4	Usage Notes	1242
32.4.1	Module Stop Function Setting	1242
32.4.2	Operation of the D/A Converter in Module Stop State	1242
32.4.3	Operation of the D/A Converter in Software Standby Mode	1242
32.4.4	Setting the D/A Converter	1242
32.4.5	Note on Usage When Measure against Interference between D/A and A/D Conversion is Enabled	1242
32.4.6	Notes on Simultaneous Use of DA0 Pin Output and Comparator C Reference Voltage	1242
33.	Temperature Sensor (TEMPSA)	1243
33.1	Overview	1243
33.2	Register Descriptions	1244
33.2.1	Temperature Sensor Calibration Data Register (TSCDR)	1244
33.3	Using the Temperature Sensor	1245
33.3.1	Preparation for Using the Temperature Sensor	1245
33.3.2	Setting the 12-Bit A/D Converter	1246
34.	Comparator C (CMPCa)	1247
34.1	Overview	1247
34.2	Register Descriptions	1250
34.2.1	Comparator Control Register (CMPCTL)	1250
34.2.2	Comparator Input Select Register (CMPSEL0)	1251
34.2.3	Comparator Reference Voltage Select Register (CMPSEL1)	1252
34.2.4	Comparator Output Monitor Register (CMPMON)	1253
34.2.5	Comparator External Output Enable Register (CMPIOC)	1253
34.2.6	Comparator Control Register 2 (CMPCTL2)	1253
34.2.7	Noise Filter Setting Register (CMPNFS)	1254
34.3	Operation	1255
34.3.1	Comparator Operation Example	1255
34.3.2	Noise Filter	1256
34.3.3	Interrupts	1257

34.3.4	Comparator Pin Output	1257
34.3.5	Comparator Setting Flowchart	1258
34.4	Usage Notes	1260
34.4.1	Module Stop Function Setting	1260
34.4.2	Comparator C Operation in Module Stop State	1260
34.4.3	Comparator C Operation in Software Standby Mode	1260
34.4.4	Comparator Operation while the 12-Bit A/D Converter is in the Module-Stop State	1260
34.4.5	Setting the D/A Converter for Generating Reference Voltage	1260
34.4.6	Appearance of a Pulse in the Result of Comparison	1261
35.	Data Operation Circuit (DOC)	1262
35.1	Overview	1262
35.2	Register Descriptions	1263
35.2.1	DOC Control Register (DOCR)	1263
35.2.2	DOC Data Input Register (DODIR)	1264
35.2.3	DOC Data Setting Register (DODSR)	1264
35.3	Operation	1265
35.3.1	Data Comparison Mode	1265
35.3.2	Data Addition Mode	1266
35.3.3	Data Subtraction Mode	1267
35.4	Interrupt Requests	1267
35.5	Usage Note	1267
35.5.1	Module Stop Function Setting	1267
36.	RAM	1268
36.1	Overview	1268
36.2	Usage Notes	1269
36.2.1	Low Power Consumption Function	1269
36.2.2	Notes on Self-Diagnosis of the RAM	1269
37.	Flash Memory (FLASH)	1270
37.1	Overview	1270
37.2	ROM Area and Block Configuration	1271
37.3	E2 DataFlash Area and Block Configuration	1272
37.4	Register Descriptions	1273
37.4.1	E2 DataFlash Control Register (DFLCTL)	1273
37.4.2	Flash P/E Mode Entry Register (FENTRYR)	1274
37.4.3	Memory Wait Cycle Setting Register (MEMWAITR)	1275
37.4.4	Protection Unlock Register (FPR)	1276
37.4.5	Protection Unlock Status Register (FPSR)	1276
37.4.6	Flash P/E Mode Control Register (FPMCR)	1277
37.4.7	Flash Initial Setting Register (FISR)	1278
37.4.8	Flash Reset Register (FRESETR)	1280
37.4.9	Flash Area Select Register (FASR)	1280

37.4.10	Flash Control Register (FCR)	1281
37.4.11	Flash Extra Area Control Register (FEXCR)	1283
37.4.12	Flash Processing Start Address Register H (FSARH)	1284
37.4.13	Flash Processing Start Address Register L (FSARL)	1284
37.4.14	Flash Processing End Address Register H (FEARH)	1285
37.4.15	Flash Processing End Address Register L (FEARL)	1285
37.4.16	Flash Write Buffer Register n (FWBn) (n = 0 to 3)	1286
37.4.17	Flash Status Register 0 (FSTATR0)	1287
37.4.18	Flash Status Register 1 (FSTATR1)	1289
37.4.19	Flash Error Address Monitor Register H (FEAMH)	1290
37.4.20	Flash Error Address Monitor Register L (FEAML)	1290
37.4.21	Flash Start-Up Setting Monitor Register (FSCMR)	1291
37.4.22	Flash Access Window Start Address Monitor Register (FAWSMR)	1291
37.4.23	Flash Access Window End Address Monitor Register (FAWEMR)	1292
37.4.24	Unique ID Register n (UIDRn) (n = 0 to 3)	1292
37.5	Start-Up Program Protection	1293
37.6	Area Protection	1294
37.7	Programming and Erasure	1295
37.7.1	Sequencer Modes	1295
37.7.1.1	E2 DataFlash Access Disabled Mode	1295
37.7.1.2	Read Mode	1296
37.7.1.3	P/E Modes	1296
37.7.2	Mode Transitions	1296
37.7.2.1	Transition from E2 DataFlash Access Disable Mode to Read Mode	1296
37.7.2.2	Transition from Read Mode to P/E Mode	1297
37.7.2.3	Transition from P/E Mode to Read Mode	1299
37.7.3	Software Commands	1300
37.7.4	Software Command Usage	1301
37.7.4.1	Program	1301
37.7.4.2	Block Erase	1303
37.7.4.3	All-Block Erase	1305
37.7.4.4	Blank Check	1307
37.7.4.5	Start-Up Area Information Program/Access Window Protection/ Access Window Information Program	1309
37.7.4.6	Forced Stop of Software Commands	1310
37.7.5	Interrupt	1310
37.8	Boot Mode	1311
37.8.1	Boot Mode (SCI Interface)	1312
37.8.1.1	Operating Conditions in Boot Mode (SCI Interface)	1312
37.8.1.2	Starting Up in Boot Mode (SCI Interface)	1313
37.8.2	Boot Mode (FINE Interface)	1314

37.8.2.1	Operating Conditions in Boot Mode (FINE Interface)	1314
37.9	Flash Memory Protection	1315
37.9.1	ID Code Protection	1315
37.9.1.1	Boot Mode ID Code Protection	1316
37.9.1.2	On-Chip Debugging Emulator ID Code Protection	1317
37.10	Communication Protocol	1318
37.10.1	State Transition in Boot Mode (SCI Interface)	1318
37.10.2	Command and Response Configuration	1319
37.10.3	Response to Undefined Commands	1319
37.10.4	Boot Mode Status Inquiry	1320
37.10.5	Inquiry Commands	1321
37.10.5.1	Supported Device Inquiry	1321
37.10.5.2	Data Area Availability Inquiry	1322
37.10.5.3	User Area Information Inquiry	1322
37.10.5.4	Data Area Information Inquiry	1323
37.10.5.5	Block Information Inquiry	1323
37.10.6	Setting Commands	1324
37.10.6.1	Device Select	1324
37.10.6.2	Operating Frequency Select	1325
37.10.6.3	Program/Erase Host Command Wait State Transition	1326
37.10.7	ID Code Authentication Command	1327
37.10.7.1	ID Code Check	1327
37.10.8	Program/Erase Commands	1328
37.10.8.1	User/Data Area Program Preparation	1328
37.10.8.2	Program	1329
37.10.8.3	Data Area Program	1330
37.10.8.4	Erase Preparation	1331
37.10.8.5	Block Erase	1331
37.10.9	Read-Check Commands	1332
37.10.9.1	Memory Read	1332
37.10.9.2	User Area Checksum	1333
37.10.9.3	Data Area Checksum	1334
37.10.9.4	User Area Blank Check	1334
37.10.9.5	Data Area Blank Check	1335
37.10.9.6	Access Window Information Program	1335
37.10.9.7	Access Window Read	1336
37.10.9.8	Access Window Protection	1337
37.10.9.9	Access Window Protection Flag Read	1337
37.11	Serial Programmer Operation in Boot Mode (SCI Interface)	1338
37.11.1	Bit Rate Automatic Adjustment	1339
37.11.2	Receive the MCU Information	1340

37.11.3	Select the Device and Change the Bit Rate	1341
37.11.4	Enter the Program/Erase Host Command Wait State	1342
37.11.5	Unlock Boot Mode ID Code Protection	1343
37.11.6	Erase the User Area and Data Area	1344
37.11.7	Program the User Area and Data Area	1345
37.11.8	Check Data in the User Area	1346
37.11.9	Check Data in the Data Area	1347
37.11.10	Set the Access Window in the User Area	1348
37.11.11	Protect the Access Window Settings	1349
37.12	Rewriting by Self-Programming	1350
37.12.1	Overview	1350
37.13	Usage Notes	1351
37.14	Usage Notes in Boot Mode	1352
38.	Electrical Characteristics	1353
38.1	Absolute Maximum Ratings	1353
38.2	Recommended Operating Conditions	1354
38.3	Test Conditions	1355
38.4	DC Characteristics	1356
38.4.1	Characteristics of the Input and Output Pins	1356
38.4.2	Operating and Standby Currents	1359
38.4.3	VCC Rising Gradient and Ripple Characteristics	1367
38.4.4	Voltage of the VCL Pin	1368
38.4.5	RAM Data Standby Characteristics	1368
38.4.6	Thermal Resistance Value (Reference Values)	1368
38.5	AC Characteristics	1369
38.5.1	Clock Timing	1369
38.5.2	Reset Timing	1373
38.5.3	Timing of Recovery from Low Power Consumption Modes	1374
38.5.4	Control Signal Timing	1376
38.5.5	Timing of On-Chip Peripheral Modules	1377
38.5.5.1	I/O Port Input Timing	1377
38.5.5.2	MTU	1377
38.5.5.3	POE	1379
38.5.5.4	POEG	1381
38.5.5.5	GPTW	1384
38.5.5.6	TMR	1386
38.5.5.7	SCI	1387
38.5.5.8	RIIC	1393
38.5.5.9	A/D Converter Trigger	1395
38.5.5.10	CAC	1395
38.5.5.11	CLKOUT	1396

38.6	A/D Conversion Characteristics	1397
38.7	Programmable Gain Amplifier Characteristics	1403
38.8	Comparator Characteristics	1404
38.9	D/A Conversion Characteristics	1405
38.10	Temperature Sensor Characteristics	1406
38.11	Power-On Reset Circuit and Voltage Detection Circuit Characteristics	1406
38.12	Oscillation Stop Detection Timing	1410
38.13	ROM (Flash Memory for Code Storage) Characteristics	1411
38.14	E2 DataFlash Characteristics (Flash Memory for Data Storage)	1413
38.15	Usage Notes	1414
38.15.1	Connecting VCL Capacitor and Bypass Capacitors	1414
Appendix 1. Port States in Each Processing Mode		1415
Appendix 2. Package Dimensions		1417
REVISION HISTORY		1426

RX14T Group Renesas MCUs

R01UH1126EJ0102

Rev.1.02

Dec 12, 2025

48-MHz, 32-bit RX MCUs, on-chip FPU, 204 Coremark, supportive of 5 V power supply, 12-bit A/D converter (2 units, programmable gain amplifier × 3 units, and comparator), 48-MHz PWM (three-phase complementary output × 2 channels), on-chip data flash memory

Features

■ 32-bit RXv2 CPU core

- Maximum operating frequency: 48 MHz
- Capable of 204 Coremark in operation at 48 MHz
- Enhanced DSP instructions: 32-bit multiply-accumulate instructions, and 16-bit multiply-subtract instructions are supported.
- On-chip FPU: 32-bit single-precision floating point compliant with IEEE-754
- On-chip divider that operated at the fastest of two clock cycles
- Fast interrupt
- CISC Harvard architecture with 5-stage pipeline
- Variable-length instructions, ultra-compact code
- On-chip debugging circuit

■ Arithmetic unit for trigonometric functions (TFUV1)

■ Low power design and architecture

- Operation from a single 2.7-V to 5.5-V supply
- Three low power consumption modes

■ On-chip flash memory for code

- 128 Kbytes size capacities
- User code is programmable by on-board programming.
- For instructions and operands

■ On-chip data flash memory

- 4K bytes (1,000,000 program/erase cycles (typ.))
- BGO (Background Operation)

■ On-chip SRAM, no wait states

- 12 Kbytes size capacities

■ DTC

- Five transfer modes

■ Reset and supply management

- Seven types of reset, including the power-on reset (POR)
- Low voltage detection (LVD) with voltage settings

■ Clock functions

- External clock input frequency: Up to 20 MHz
- Main clock oscillator frequency: 1 to 20 MHz
- PLL circuit input: 4 MHz to 12.5 MHz
- Low-speed on-chip oscillator: 4 MHz
- High-speed on-chip oscillator: 24/32/48/64 MHz ± 1%
- IWDT-dedicated on-chip oscillator: 15 kHz
- On-chip clock frequency accuracy measurement circuit (CAC)

■ Independent watchdog timer

- 15-kHz on-chip oscillator produces a dedicated clock signal to drive IWDT operation.

■ Useful functions for IEC60730 compliance

- Self-diagnostic and disconnection-detection assistance functions for the A/D converter, clock frequency accuracy measurement circuit, independent watchdog timer, RAM test assistance functions using the DOC, CRCA, etc.

■ MPC

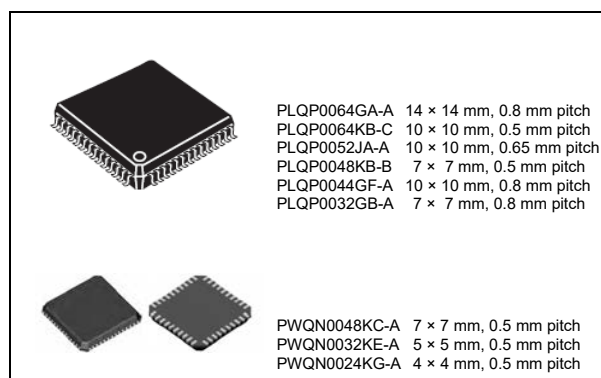
- Input/output functions selectable from multiple pins

■ Up to 5 communication functions

- SCI with multiple functionalities (4 channels)
- Choose from among asynchronous mode, clock-synchronous mode, smart-card interface mode, simplified SPI, simplified I²C, and extended serial mode.
- I²C bus interface: Transfer at up to 400 kbps, capable of SMBus operation (one channel)

■ Up to 15 extended-function timers

- 16-bit MTU3 (6 channels): 48 MHz operation, input capture, output compare, three-phase complementary PWM × 1 channel-output, CPU-efficient complementary PWM, phase counting mode (2 channels)



- 16-bit GPTW (3 channels): operation at 48 MHz, input capture, output compare, PWM waveforms: 6 output channels in single-phase complementary PWM mode/1 output channels in 3-phase complementary PWM mode, PWM waveforms: sawtooth-wave mode, triangle-wave mode (with dead time), linkage with comparator (counting operation, PWM negate control) Simultaneous starting of MTU3 and GPTW
- 8-bit TMR (4 channels)
- 16-bit compare-match timers (2 channels)

■ 12-bit A/D converter (2 units)

- Capable of conversion within 0.5 μs
- Unit 0: 8 channels unit 1: 8 channels
- Sampling time can be set for each channel
- Group scan priority control mode (3 levels)
- Self-diagnostic function and analog input disconnection detection assistance function
- Double trigger (data duplication) function for motor control
- Time-interleaved operation to achieve a high sample rate is possible.

■ Input signal amplification function by the programmable gain amplifier (3 units)

■ D/A converter

- 2 channels
- This can be used as reference voltage for a comparator

■ Analog Comparator: 3 channels

■ Register write protection function can protect values in important registers against overwriting.

■ Up to 60 pins for general I/O ports

- 5-V tolerant, open drain, input pull-up

■ Temperature sensor

■ Unique ID

- 32-byte ID code for the MCU

■ Operating temperature range

- −40 to +105°C
- −40 to +125°C

■ Applications

- General industrial and consumer equipment

1. Overview

1.1 Outline of Specifications

Table 1.1 lists the specifications, and Table 1.2 gives a comparison of the functions of the products in different packages.

Table 1.1 is for products with the greatest number of functions, so the number of peripheral modules and channels will differ in accordance with the package type. For details, see Table 1.2, Comparison of Functions for Different Packages.

Table 1.1 Outline of Specifications (1/4)

Classification	Module/Function	Description
CPU	CPU	- Maximum operating frequency: 48 MHz 32-bit RX CPU (RXv2) - Minimum instruction execution time: One instruction per clock cycle - Address space: 4-Gbyte linear - Register set - General purpose: Sixteen 32-bit registers - Control: Ten 32-bit registers - Accumulator: Two 72-bit registers - Basic instructions: 75 (variable-length instruction format) - Floating point instructions: 11 - DSP instructions: 23 - Addressing modes: 11 - Data arrangement - Instructions: Little endian - Data: Selectable as little endian or big endian - On-chip 32-bit multiplier: 32-bit × 32-bit → 64-bit - On-chip divider: 32-bit ÷ 32-bit → 32 bits - Barrel shifter: 32 bits
	FPU	- Single precision (32-bit) floating point - Data types and exceptions in conformance with the IEEE754 standard
Arithmetic unit for trigonometric functions (TFUV1)		- Sine, cosine, arctangent, $\sqrt{x^2 + y^2}$ - Simultaneous calculation of sine and cosine - Simultaneous calculation of arctangent and $\sqrt{x^2 + y^2}$
Memory	ROM	- Capacity: 128 Kbytes 32 MHz ≤: No-wait cycle access 32 MHz to 48 MHz: One-wait cycle access - Programming/erasing method: - Serial programming (asynchronous serial communication), self-programming
	RAM	- Capacity: 12 Kbytes - No-wait memory access
	E2 DataFlash	- Capacity: 4 Kbytes - Number of erase/write cycles: 1,000,000 (typ.)
MCU operating mode		Single-chip mode
Clock	Clock generation circuit	- Main clock oscillator, low-speed on-chip oscillator, high-speed on-chip oscillator, PLL frequency synthesizer, and IWDG-dedicated on-chip oscillator - Oscillation stop detection: Available - Clock frequency accuracy measurement circuit (CAC) - Independent settings for the system clock (ICLK), peripheral module clock (PCLK), and FlashIF clock (FCLK) - The CPU and system sections such as other bus masters run in synchronization with the system clock (ICLK): 48 MHz (at max.) - Peripheral modules run in synchronization with the PCLKB: 48 MHz (at max.) - The flash peripheral circuit runs in synchronization with the FCLK: 48 MHz (at max.) - ADCLK in the S12AD runs in synchronization with PCLKD: Up to 64 MHz
Resets		RES# pin reset, power-on reset, voltage monitoring reset, independent watchdog timer reset, and software reset
Voltage detection	Voltage detection circuit (LVDAb)	- When the voltage on VCC falls below the voltage detection level, an internal reset or internal interrupt is generated. - Voltage detection circuit 0 is capable of selecting the detection voltage from 3 levels - Voltage detection circuit 1 is capable of selecting the detection voltage from 10 levels - Voltage detection circuit 2 is capable of selecting the detection voltage from 4 levels

Table 1.1 Outline of Specifications (2/4)

Classification	Module/Function	Description
Low power consumption	Low power consumption functions	- Module stop function - Three low power consumption modes Sleep mode, deep sleep mode, software standby mode
	Function for lower operating power consumption	Operating power control modes High-speed operating mode, middle-speed operating mode
Interrupt	Interrupt controller (ICUb)	- Interrupt vectors: 256 - External interrupts: 9 (NMI, IRQ0 to IRQ7 pins) - Non-maskable interrupts: 5 (The NMI pin, oscillation stop detection interrupt, voltage monitoring 1 interrupt, voltage monitoring 2 interrupt, and IWDG interrupt) 16 levels specifiable for the order of priority
DMA	Data transfer controller (DTCb)	- Transfer modes: Normal transfer, repeat transfer, and block transfer - Activation sources: Interrupts - Sequence transfer
I/O ports	General I/O ports	64-pin/52-pin/48-pin/44-pin/32-pin/24-pin - I/O: 59/47/43/39/27/19 - Input: 1/1/1/1/1/1 - Pull-up resistors: 59/47/43/39/27/19 - Open-drain outputs: 47/37/33/30/20/14 5-V tolerant: 2/2/2/2/2/2
Multi-function pin controller (MPC)		Capable of selecting the input/output function from multiple pins
Timers	Multi-function timer pulse unit 3 (MTU3c)	6 units (16 bits × 6 channels) - Provides up to 16 pulse-input/output lines and three pulse-input lines - Select from among fourteen counter-input clock signals for each channel (PCLK/1, PCLK/2, PCLK/4, PCLK/8, PCLK/16, PCLK/32, PCLK/64, PCLK/256, PCLK/1024, MTCLKA, MTCLKB, MTCLKC, MTCLKD, MTIOC1A) other than channel 1/3/4, for which only eleven signals are available, channel 2 for 12, channel 5 for 10 26 output compare/input capture registers - Counter clear operation (with compare match- or input capture-sourced simultaneous counter clear capability) - Simultaneous writing to multiple timer counters (TCNT) - Simultaneous register input/output by synchronous counter operation - Buffer operation - Cascaded operation 28 interrupt sources - Automatic transfer of register data - Pulse output modes: Toggle/PWM/complementary PWM/reset-synchronized PWM - Complementary PWM output mode 3-phase non-overlapping waveform output for inverter control - Automatic dead time setting - Adjustable PWM duty cycle: from 0 to 100% - A/D conversion request delaying function - Interrupt at crest/trough can be skipped - Double buffer function - Reset-synchronized PWM mode - Outputs three phases each for positive and negative PWM waveforms in user-specified duty cycle - Phase counting modes: 16-bit mode (channel 1 and 2)/32-bit mode (channel 1 and 2) - Dead time compensation counter function - A/D converter start trigger can be generated - A/D converter start triggers can be skipped - Signals from the input capture and external counter clock pins are input via a digital filter
	Port output enable 3 (POE3E)	- Control of the high-impedance state of the MTU/GPTW's waveform output pins, and control of switching to the general I/O port pin 5 pins for input from signal sources: POE0, POE8, POE10, POE11, POE12 - Initiation by detection of short-circuited outputs (detection of PWM outputs that have become an active level simultaneously) - Initiation by comparator detection/oscillation stop detection/software - Additional programming of output control target pins is enabled

Table 1.1 Outline of Specifications (3/4)

Classification	Module/Function	Description
Timers	General PWM timer (GPTWd)	• 16 bits × 3 channels • Counting up or down (sawtooth-wave), counting up and down (triangle-wave) selectable for all channels • Clock sources independently selectable for each channel • 2 input/output pins per channel • 2 output compare/input capture registers per channel • For the 2 output compare/input capture registers of each channel, 4 registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use. • In output compare operation, buffer switching can be at peaks or troughs, enabling the generation of laterally asymmetrically PWM waveforms. • Registers for setting up frame intervals on each channel (with capability for generating interrupts on overflow or underflow) • Generation of dead times in PWM operation • Capable of synchronous start, stop, or clearing of counter for any channel • Capable of a start, stop, clearing, or up-/down-counting of the counter supporting input level comparison • Capable of a start, stop, clearing, or up-/down-counting of the counter supporting 1 external triggers • Output pin disabling function by a dead time error or a short circuit detection among output pins • Capable of generating conversion start triggers for the A/D converters as well as monitoring external pins for a start timing of conversion. • Capable of using noise filter of input capture • Simultaneous starting of the GPT by MTU0 or MTU4
	Port output enable for GPTW (POEG)	• Controlling the output disable for GPTW waveform output • Initiation by input level detection of GTETRGA pin • Initiation by output disable request from GPTW • Initiation by detection of comparator interrupt request • Initiation by detection of oscillation stop or by software
	8-bit timer (TMRb)	• (8 bits × 2 channels) × 2 units • Seven internal clocks (PCLK/1, PCLK/2, PCLK/8, PCLK/32, PCLK/64, PCLK/1024, and PCLK/8192) and an external clock can be selected • Pulse output and PWM output with any duty cycle are available • Two channels can be cascaded and used as a 16-bit timer
	Compare match timer (CMT)	• (16 bits × 2 channels) × 1 unit • Select from among four clock signals (PCLK/8, PCLK/32, PCLK/128, PCLK/512)
	Independent watchdog timer (IWDtA)	• 14 bits × 1 channel • Count clock: Dedicated low-speed clock for the IWDt • Frequency divided by 1, 16, 32, 64, 128, or 256
Communication functions	Serial communications interfaces (SCIg, SCIH)	• 4 channels (channel 1, 5, and 6: SCIg, channel 12: SCIH) • SCIg - Serial communications modes: Asynchronous, clock synchronous, and smart-card interface - Multi-processor function - On-chip baud rate generator allows selection of the desired bit rate - Choice of LSB-first or MSB-first transfer - Average transfer rate clock can be input from TMR timers (SCI5, SCI6, SCI12) - Start-bit detection: Level or edge detection is selectable. - Simple I²C - Simple SPI 9-bit transfer mode - Bit rate modulation • SCIH (The following functions are added to SCIg) - Supports the serial communications protocol, which contains the start frame and information frame - Supports the LIN format
	I ² C bus interface (RIICa)	• 1 channel • Communications formats: I²C bus format/SMBus format • Master mode or slave mode selectable • Supports fast mode

Table 1.1 Outline of Specifications (4/4)

Classification	Module/Function	Description
12-bit A/D converter (S12AD)		12 bits 2 units (8 channels × 2 units) - Minimum conversion time: 0.5 μs per channel when the ADCLK is operating at 64 MHz - Operating modes - Scan mode (single scan mode, continuous scan mode, and 3 group scan mode) - Starting and stopping of scanning according to group priority control is possible in the group scan mode. - The order of priority is group A (highest) > group B > group C (lowest). - Sampling variable - Sampling time can be set up for each channel - Self-diagnostic function - Double trigger mode (A/D conversion data duplicated) - Assist on analog input disconnection detection - A/D conversion start conditions - A software trigger, a trigger from a timer (MTU, GPTW, TMR), or an external trigger signal - Time-interleaved operation is possible, with two units of the AD converter alternately converting the voltage on the same channel to achieve a high sample rate. - The reference voltage is selectable as VREFH0/VCC or VREFL0/VSS.
Programmable Gain Amplifier (PGA)		- Input signal amplification function by the programmable gain amplifier (3 units) - Amplification rate: 4 times, 8 times, 16 times, 32 times (total of 4 steps)
Comparator C (CMPC)		3 channels - Function to compare the reference voltage and the analog input voltage - The reference voltage is selectable per comparator as either of the two outputs of the D/A converter or as an external input. - Analog input voltages: Four (for CMPC0 and CMPC1) or three (for CMPC2) - Digital filtering
D/A converter (DA)		2 channels 8-bit resolution - Output voltage: 0 V to VCC - External output is possible, but only from DA0. Both outputs can be used as comparator C reference voltages.
Temperature sensor (TEMPSA)		1 channel - The voltage output from the temperature sensor is converted into a digital value by the 12-bit A/D converter.
Safety	Register write protection function	Protects important registers from being overwritten in case a program runs out of control.
	CRC calculator (CRCA)	- Generation of CRC codes for 8-/32-bit data 8-bit data - Selectable from the following three polynomials $X^8 + X^2 + X + 1$, $X^{16} + X^{15} + X^2 + 1$, $X^{16} + X^{12} + X^5 + 1$ 32-bit data - Selectable from the following two polynomials $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$, $X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^8 + X^6 + 1$ - Generation of CRC codes for use with LSB-first or MSB-first communications is selectable
	Main clock oscillation stop detection function	Main clock oscillation stop detection: Available
	Clock frequency accuracy measurement circuit (CAC)	Monitors the clock output from the main clock oscillator, low- and high-speed on-chip oscillators, IWDI-dedicated on-chip oscillator, and PCLKB.
	Data operation circuit (DOC)	Comparison, addition, and subtraction of 16-bit data
Unique ID		32-byte ID code for the MCU
Power supply voltages/Operating frequencies		VCC = 2.7 to 5.5 V: 48 MHz
Operating temperature range		G version: -40 to +105°C, M version: -40 to +125°C
Packages		64-pin LFQFP (PLQP0064KB-C) 10 × 10 mm, 0.5 mm pitch 64-pin LQFP (PLQP0064GA-A) 14 × 14 mm, 0.8 mm pitch 52-pin LQFP (PLQP0052JA-A) 10 × 10 mm, 0.65 mm pitch 48-pin LFQFP (PLQP0048KB-B) 7 × 7 mm, 0.5 mm pitch 44-pin LQFP (PLQP0044GF-A) 10 × 10 mm, 0.8 mm pitch 32-pin LQFP (PLQP0032GB-A) 7 × 7 mm, 0.8 mm pitch 48-pin HWQFN (PWQN0048KC-A) 7 × 7 mm, 0.5 mm pitch 32-pin HWQFN (PWQN0032KE-A) 5 × 5 mm, 0.5 mm pitch 24-pin HWQFN (PWQN0024KG-A) 4 × 4 mm, 0.5 mm pitch
Debugging interfaces		FINE interface

Table 1.2 Comparison of Functions for Different Packages

Module/Functions		RX14T					
		64 Pins	52 Pins	48 Pins	44 Pins	32 Pins	24 Pins
Memory	ROM	128 Kbytes					
	RAM	12 Kbytes					
	E2 data flash	4 Kbytes					
External interrupts		NMI, IRQ0 to IRQ7					
DMA	Data transfer controller	Available					
Timers	Multi-function timer pulse unit 3	ch0 to ch5				ch0 to ch4	
	Port output enable 3	Available					
	General PWM timer	ch0 to ch2					
	Port output enable for GPTW	Available					
	8-bit timer	2 channels × 2 units					
	Compare match timer	2 channels × 1 unit					
	Independent watchdog timer	Available					
Communication functions	Serial communications interfaces (SCIg)	3 channels (SCI1, 5, 6)					
	Serial communications interfaces (SCIh)	1 channel (SCI12)					
	I ² C bus interface	1 channel					
12-bit A/D converter		Number of pins: 14*1 8 in unit 0 8 in unit 1	Number of pins: 12*1 8 in unit 0 6 in unit 1	Number of pins: 12*1 8 in unit 0 6 in unit 1	Number of pins: 11*1 7 in unit 0 6 in unit 1	Number of pins: 8*1 5 in unit 0 5 in unit 1	Number of pins: 6*1 3 in unit 0 5 in unit 1
Programmable Gain Amplifier		3 units					
Comparator C		3 channels					
D/A converter		2 channels					
Temperature sensor		Available					
CRC calculator		Available					
Data operation circuit		Available					
Clock frequency accuracy measurement circuit		Available					
Packages		64-pin LFQFP (0.5 mm) 64-pin LQFP (0.8 mm)	52-pin LQFP (0.65 mm)	48-pin LFQFP (0.5 mm) 48-pin HWQFN (0.5 mm)	44-pin LQFP (0.8 mm)	32-pin LQFP (0.8 mm) 32-pin HWQFN (0.5 mm)	24-pin HWQFN (0.5 mm)

Note 1. Two of these pins are exclusively shared by units 0 and 1.

1.2 List of Products

Table 1.3 is a lists of products, and Figure 1.1 shows how to read the product part no., memory capacity, and package type.

Table 1.3 List of Products

Group	Part No.	Package	ROM Capacity	RAM Capacity	E2 Data Flash	Operating Frequency	Operating Temperature
RX14T	R5F514T5AGFM	PLQP0064KB-C	128 Kbytes	12 Kbytes	4 Kbytes	48 MHz	–40 to +105°C
	R5F514T5AGFK	PLQP0064GA-A					
	R5F514T5AGFD	PLQP0052JA-A					
	R5F514T5AGFL	PLQP0048KB-B					
	R5F514T5AGFV	PLQP0044GF-A					
	R5F514T5AGFJ	PLQP0032GB-A					
	R5F514T5AGNE	PWQN0048KC-A					
	R5F514T5AGNH	PWQN0032KE-A					
	R5F514T5AGNK	PWQN0024KG-A					
	R5F514T5AMFM	PLQP0064KB-C	128 Kbytes	12 Kbytes	4 Kbytes	48 MHz	–40 to +125°C
	R5F514T5AMFK	PLQP0064GA-A					
	R5F514T5AMFD	PLQP0052JA-A					
	R5F514T5AMFL	PLQP0048KB-B					
	R5F514T5AMFJ	PLQP0032GB-A					
	R5F514T5AMNE	PWQN0048KC-A					
	R5F514T5AMNH	PWQN0032KE-A					
	R5F514T5AMNK	PWQN0024KG-A					

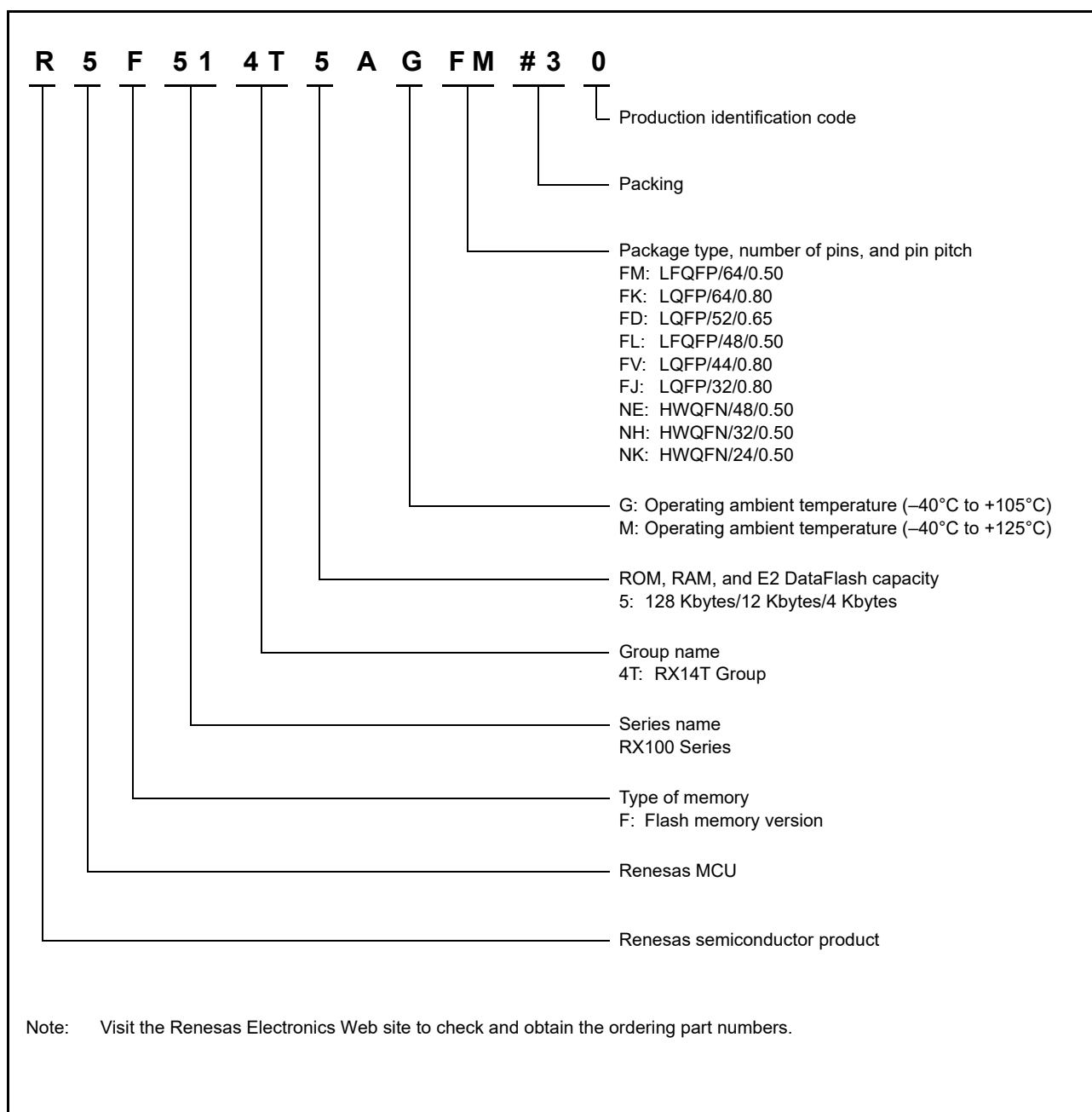


Figure 1.1 How to Read the Product Part Number

1.3 Block Diagram

Figure 1.2 shows a block diagram.

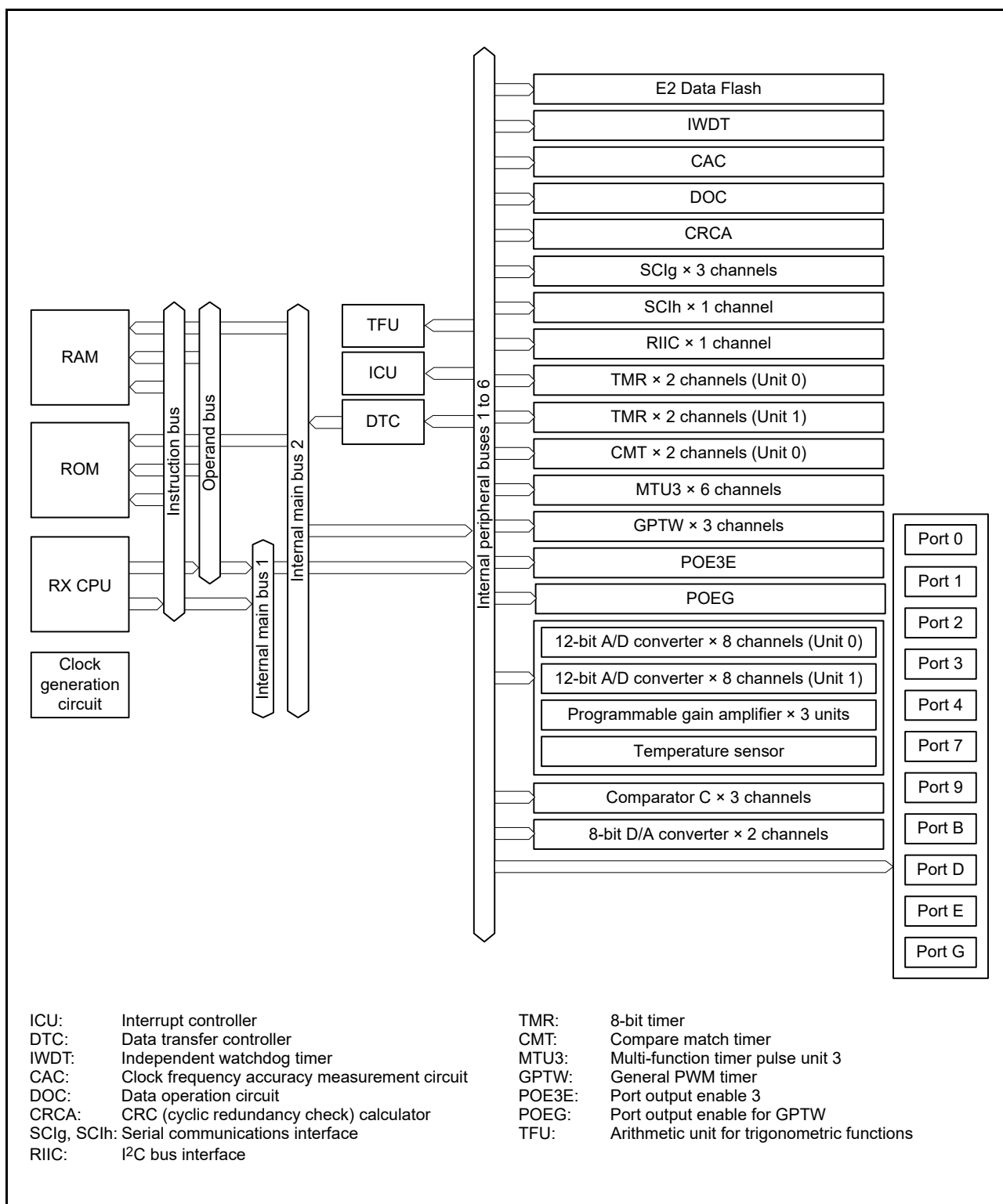


Figure 1.2 Block Diagram

1.4 Pin Functions

Table 1.4 lists the pin functions.

Table 1.4 Pin Functions (1/3)

Classifications	Pin Name	I/O	Description
Power supply	VCC	Input	Power supply pin. Connect this pin to the system power supply. Connect the pin to VSS via a 0.1-μF multilayer ceramic capacitor. The capacitor should be placed close to the pin.
	VCL	Input	Connect this pin to the VSS pin via the 4.7 μF smoothing capacitor used to stabilize the internal power supply. Place the capacitor close to the pin.
	VSS	Input	Ground pin. Connect it to the system power supply (0 V).
Clock	XTAL	Output	Pins for connecting a crystal. An external clock can be input through the EXTAL pin.
	EXTAL	Input	
	CLKOUT	Output	Clock output pin.
Operating mode control	MD	Input	Pin for setting the operating mode. For usage, refer to section 3.1, Operating Mode Types and Selection.
System control	RES#	Input	Reset pin. This MCU enters the reset state when this signal goes low.
Clock frequency accuracy measurement circuit	CACREF	Input	Input pin for the clock frequency accuracy measurement circuit.
On-chip emulator	FINED	I/O	FINE interface pin.
Interrupts	NMI	Input	Non-maskable interrupt request pin.
	IRQ0 to IRQ7	Input	Interrupt request pins.
Multi-function timer pulse unit 3	MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D	I/O	The TGRA0 to TGRD0 input capture input/output compare output/PWM output pins
	MTIOC0A#, MTIOC0B#, MTIOC0C#, MTIOC0D#	I/O	The TGRA0 to TGRD0 input capture inverted input/output compare inverted output/PWM inverted output pins.
	MTIOC1A, MTIOC1B	I/O	The TGRA1 and TGRB1 input capture input/output compare output/PWM output pins.
	MTIOC1A#, MTIOC1B#	I/O	The TGRA1 and TGRB1 input capture inverted input/output compare inverted output/PWM inverted output pins.
	MTIOC2A, MTIOC2B	I/O	The TGRA2 and TGRB2 input capture input/output compare output/PWM output pins.
	MTIOC2A#, MTIOC2B#	I/O	The TGRA2 and TGRB2 input capture inverted input/output compare inverted output/PWM inverted output pins.
	MTIOC3A, MTIOC3B, MTIOC3C, MTIOC3D	I/O	The TGRA3 to TGRD3 input capture input/output compare output/PWM output pins.
	MTIOC3A#, MTIOC3B#, MTIOC3C#, MTIOC3D#	I/O	The TGRA3 to TGRD3 input capture inverted input/output compare inverted output/PWM inverted output pins.
	MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D	I/O	The TGRA4 to TGRD4 input capture input/output compare output/PWM output pins
	MTIOC4A#, MTIOC4B#, MTIOC4C#, MTIOC4D#	I/O	The TGRA4 to TGRD4 input capture inverted input/output compare inverted output/PWM inverted output pins.
	MTIC5U, MTIC5V, MTIC5W	Input	The TGRU5, TGRV5, and TGRW5 input capture input/external pulse input pins
	MTIC5U#, MTIC5V#, MTIC5W#	Input	The TGRU5, TGRV5, and TGRW5 input capture inverted input/external pulse inverted input pins.
	MTCLKA, MTCLKB, MTCLKC, MTCLKD	Input	Input pins for the external clock.
	MTCLKA#, MTCLKB#, MTCLKC#, MTCLKD#	Input	Inverted input pins for the external clock.
	ADSM0, ADSM1	Output	A/D conversion start request frame synchronization signal output pins.

Table 1.4 Pin Functions (2/3)

Classifications	Pin Name	I/O	Description
Port output enable 3	POE0#, POE8#, POE10#, POE11#, POE12#	Input	Input pins for request signals to switch the MTU and GPTW pins between the high impedance state
General PWM timer	GTETRGA	Input	External trigger input pin
	GTIOC0A to GTIOC2A, GTIOC0B to GTIOC2B	I/O	Input capture input/output compare output/PWM output pins
	GTIOC0A# to GTIOC2A#, GTIOC0B# to GTIOC2B#	I/O	Input capture inverted input/output compare inverted output/PWM inverted output pins
	GTCPPO0	Output	Synchronized PWM output
	GTADSM0, GTADSM1	Output	A/D conversion start request monitoring output pins
8-bit timer	TMO0 to TMO3	Output	Compare match output pins.
	TMCIO to TMCIO3	Input	Input pins for the external clock to be input to the counter.
	TMRI0 to TMRI3	Input	Counter reset input pins.
Serial communications interface (SCIg)	• Asynchronous mode/clock synchronous mode		
	SCK1, SCK5, SCK6	I/O	Input/output pins for the clock.
	RXD1, RXD5, RXD6	Input	Input pins for received data.
	TXD1, TXD5, TXD6	Output	Output pins for transmitted data.
	CTS1#, CTS5#, CTS6#	Input	Input pins for controlling the start of transmission and reception.
	RTS1#, RTS5#, RTS6#	Output	Output pins for controlling the start of transmission and reception.
	• Simple I ² C mode		
	SSCL1, SSCL5, SSCL6	I/O	Input/output pins for the I ² C clock.
	SSDA1, SSDA5, SSDA6	I/O	Input/output pins for the I ² C data.
	• Simple SPI mode		
	SCK1, SCK5, SCK6	I/O	Input/output pins for the clock.
	SMISO1, SMISO5, SMISO6	I/O	Input/output pins for slave transmit data.
	SMOSI1, SMOSI5, SMOSI6	I/O	Input/output pins for master transmit data.
	SS1#, SS5#, SS6#	Input	Chip-select input pin.
Serial communications interface (SCIh)	• Asynchronous mode/clock synchronous mode		
	SCK12	I/O	Input/output pin for the clock.
	RXD12	Input	Input pin for receiving data.
	TXD12	Output	Output pin for transmitting data.
	CTS12#	Input	Input pin for controlling the start of transmission and reception.
	RTS12#	Output	Output pin for controlling the start of transmission and reception.
	• Simple I ² C mode		
	SSCL12	I/O	Input/output pin for the I ² C clock.
	SSDA12	I/O	Input/output pin for the I ² C data.
	• Simple SPI mode		
	SCK12	I/O	Input/output pin for the clock.
	SMISO12	I/O	Input/output pin for slave transmit data.
	SMOSI12	I/O	Input/output pin for master transmit data.
	SS12#	Input	Chip-select input pin.
	• Extended serial mode		
	RXD12	Input	Input pin for received data.
	TXD12	Output	Output pin for transmitted data.
	SIOX12	I/O	Input/output pin for received or transmitted data.

Table 1.4 Pin Functions (3/3)

Classifications	Pin Name	I/O	Description
I ² C bus interface	SCL0	I/O	Input/output pin for I ² C bus interface clocks. Bus can be directly driven by the N-channel open drain output.
	SDA0	I/O	Input/output pin for I ² C bus interface data. Bus can be directly driven by the N-channel open drain output.
12-bit A/D converter	AN000 to AN007, AN102 to AN107	Input	Input pins for the analog signals to be processed by the A/D converter.
	ADTRG0#, ADTRG1#	Input	Input pins for the external trigger signal that start the A/D conversion.
	ADST0, ADST1	Output	Output pins for A/D conversion status.
	ADSS0, ADSS1	Output	Output pins indicating results of AD sampling
PGA	PGAIN00 to PGAIN02	Input	PGA analog input pins
	PGAGND	Input	PGA ground pin
	PGAOUT0	Output	PGA monitoring pin
8-bit D/A converter	DA0	Output	Output pins for the analog signals to be processed by the D/A converter
Comparator C	COMP0 to COMP3	Output	Comparator detection result output pins.
	CVREFC0	Input	Analog reference voltage supply pins for comparator C.
	CMPCnm	Input	Analog input pin for CMPCnm (n = 0 to 2, m = 0 to 3)
Analog power supply	VREFH0	Input	Analog reference voltage supply pin for the 12-bit A/D converter.
	VREFL0	Input	Analog reference ground pin for the 12-bit A/D converter.
I/O ports	P00 to P04	I/O	5-bit input/output pins.
	P10 to P15	I/O	6-bit input/output pins.
	P22 to P26	I/O	5-bit input/output pins.
	P30 to P33, P36, P37	I/O	6-bit input/output pins.
	P40 to P47	I/O	8-bit input/output pins.
	P70 to P76	I/O	7-bit input/output pins.
	P90 to P97	I/O	8-bit input/output pins.
	PB0 to PB7	I/O	8-bit input/output pins.
	PD3 to PD7	I/O	5-bit input/output pins.
	PE2	Input	1-bit input/output pin (PE2 input pin).
	PG7	I/O	1-bit input/output pin.

1.5 Pin Assignments

1.5.1 64-pin LFQFP, 64-pin LQFP

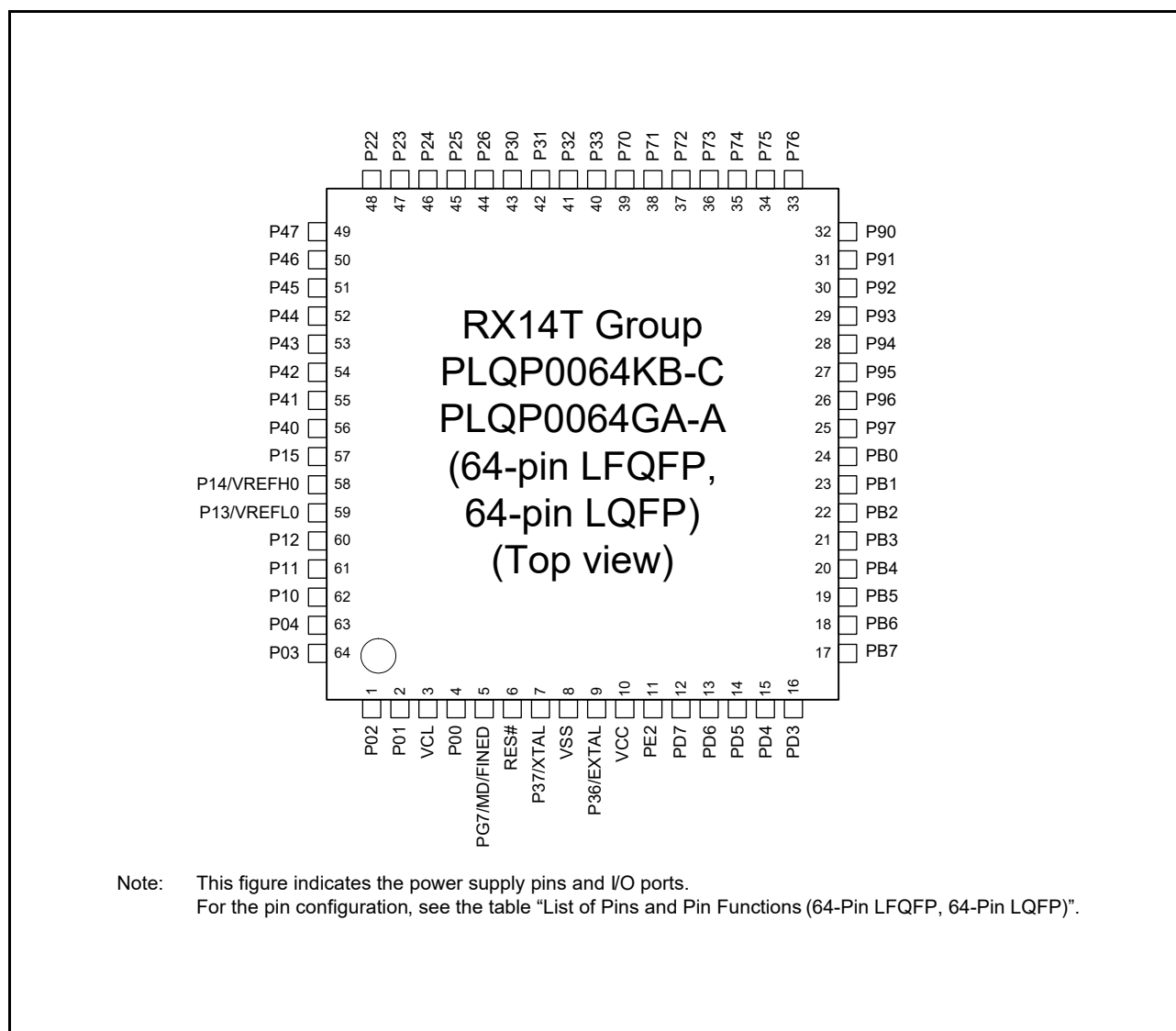
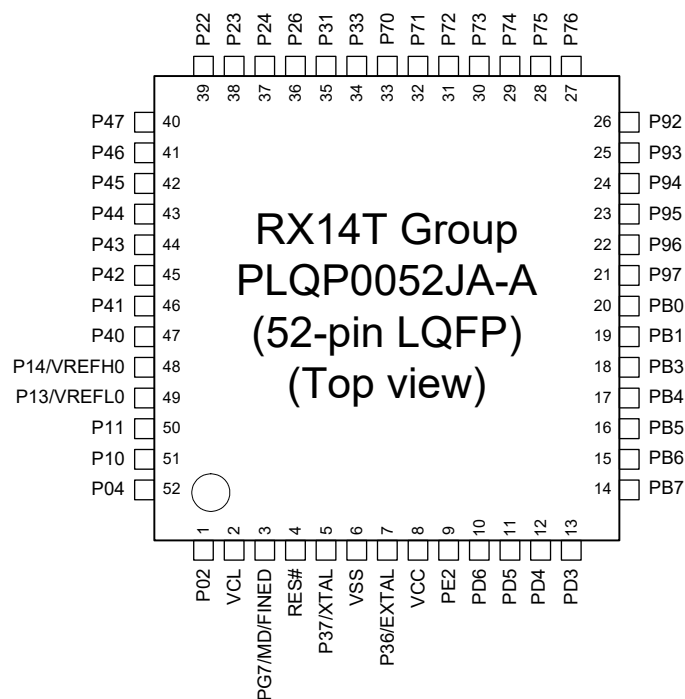


Figure 1.3 Pin Assignments of the 64-Pin LFQFP, 64-Pin LQFP

1.5.2 52-pin LQFP



Note: This figure indicates the power supply pins and I/O ports.
For the pin configuration, see the table "List of Pins and Pin Functions (52-Pin LQFP)".

Figure 1.4 Pin Assignments of the 52-Pin LQFP

1.5.3 48-pin LFQFP

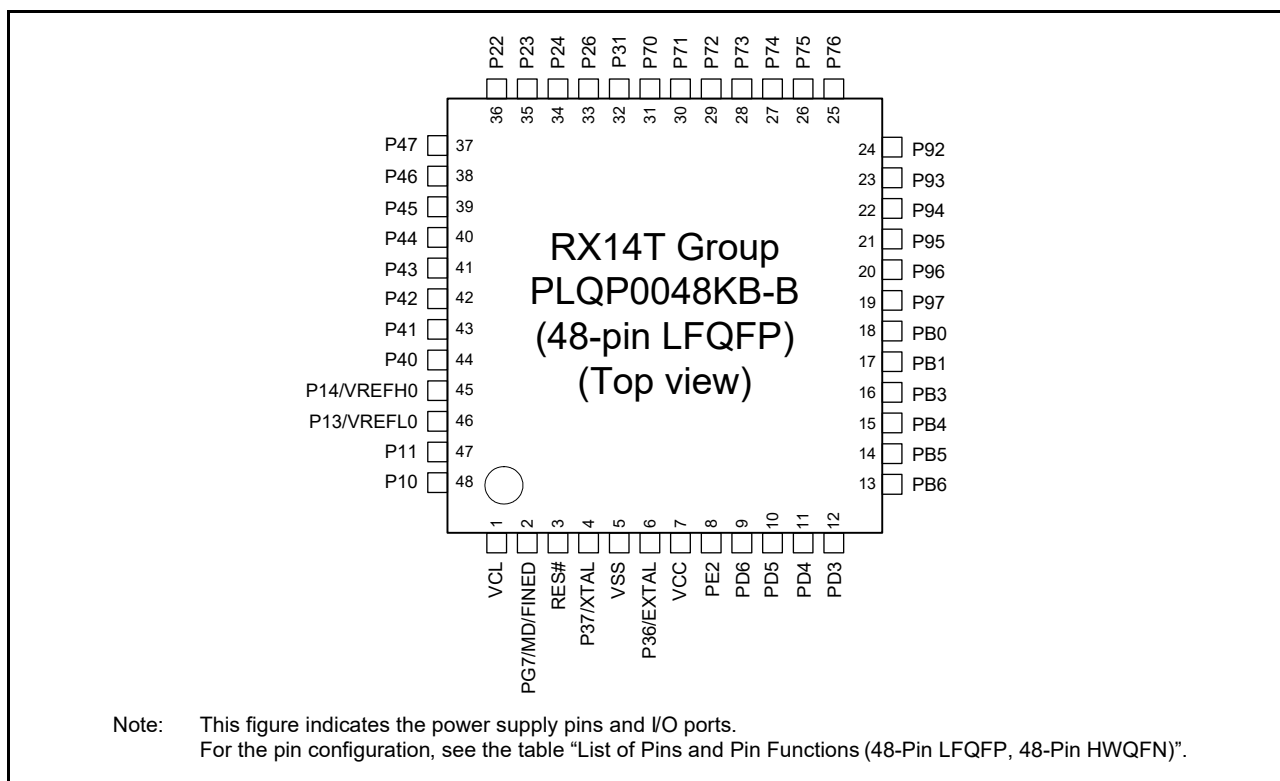


Figure 1.5 Pin Assignments of the 48-Pin LQFP

1.5.4 48-pin HWQFN

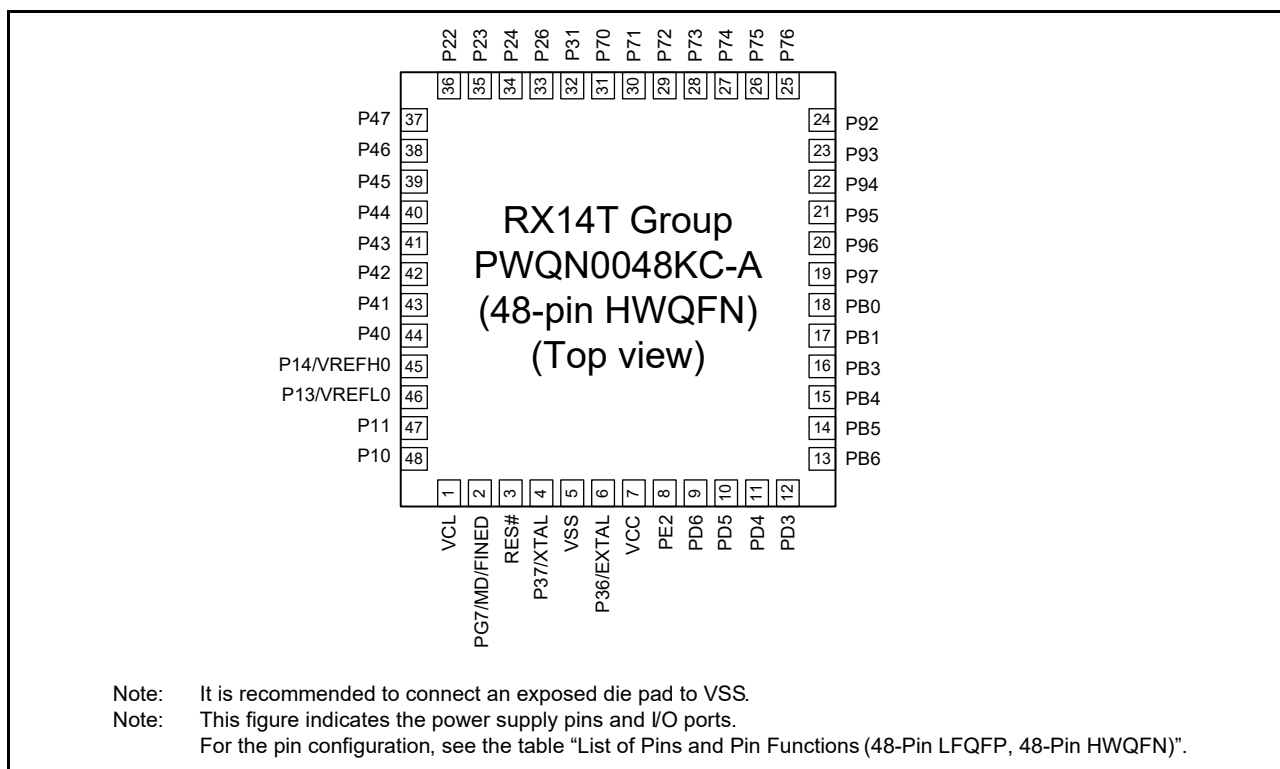


Figure 1.6 Pin Assignments of the 48-Pin HWQFN

1.5.5 44-pin LQFP

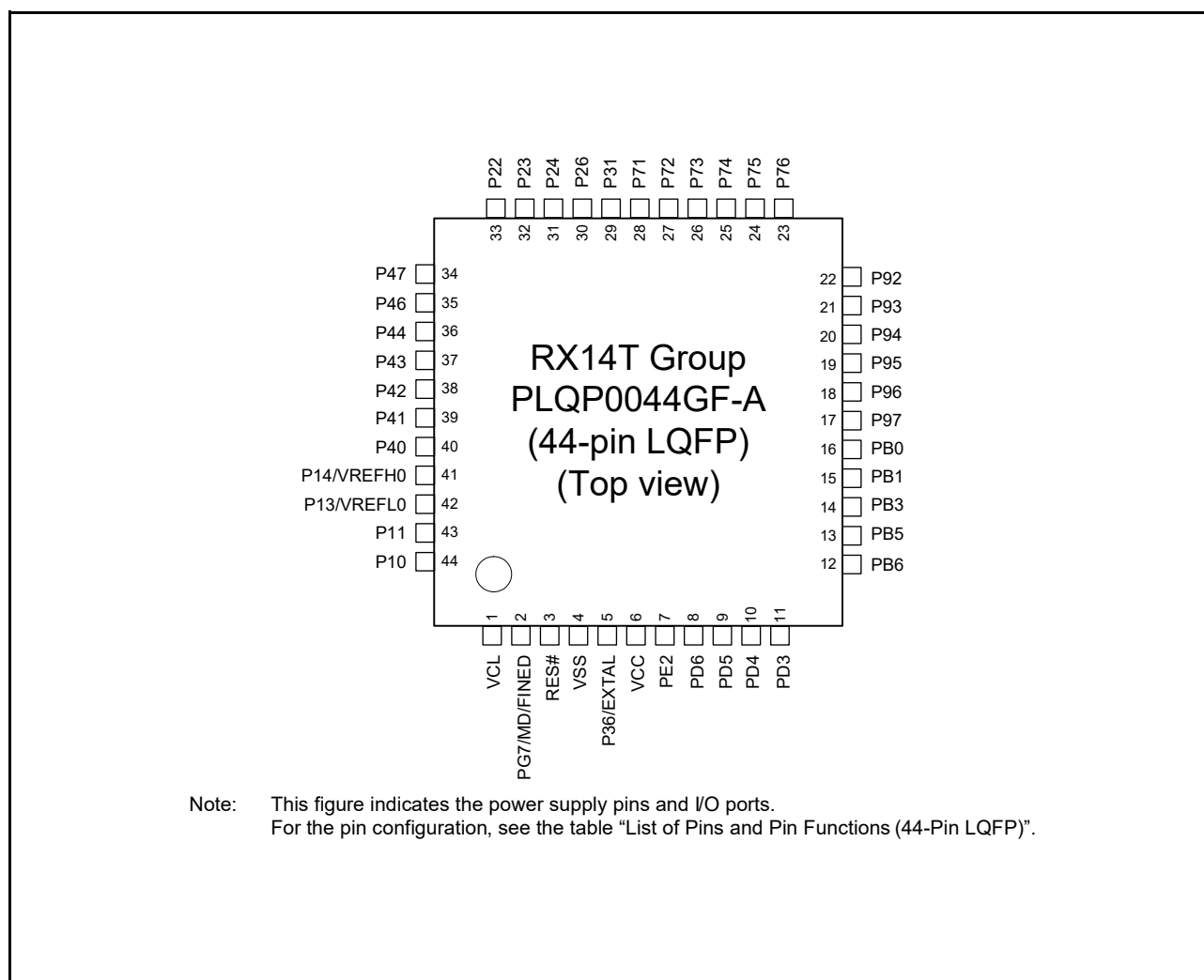


Figure 1.7 Pin Assignments of the 44-Pin LQFP

1.5.6 32-pin LQFP

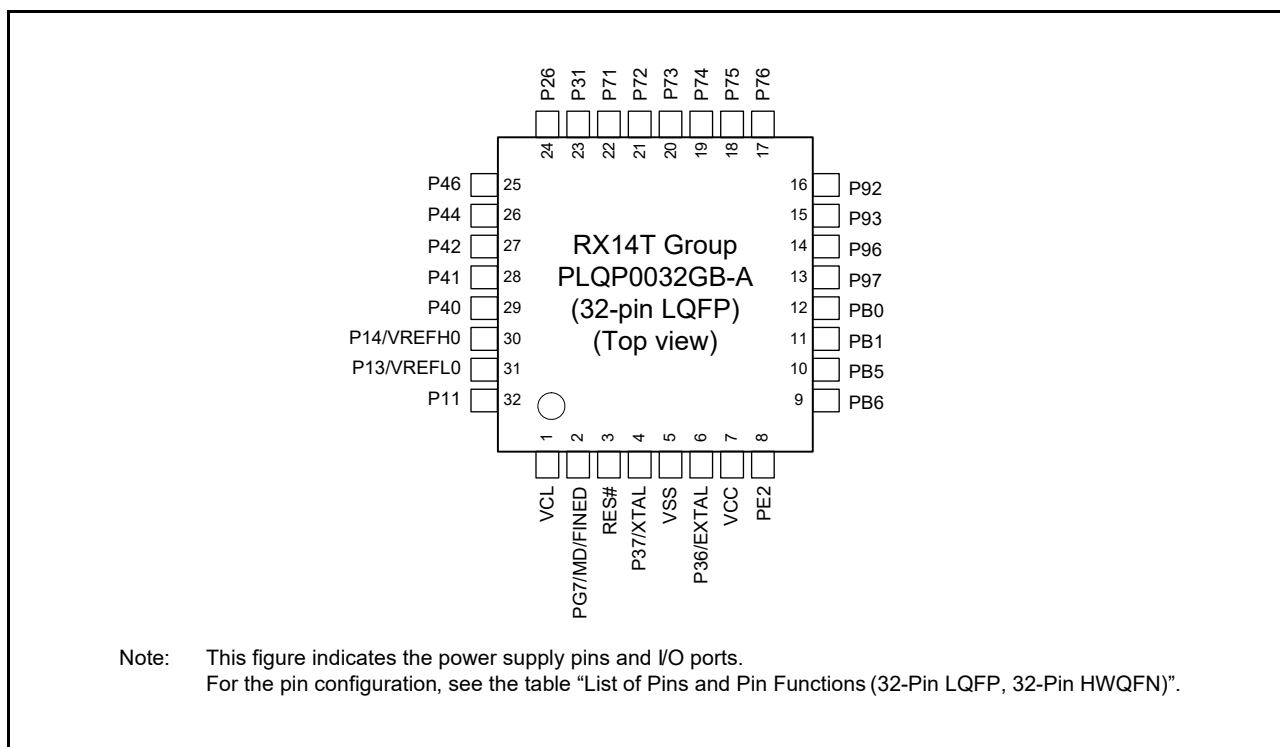


Figure 1.8 Pin Assignments of the 32-Pin LQFP

1.5.7 32-pin HWQFN

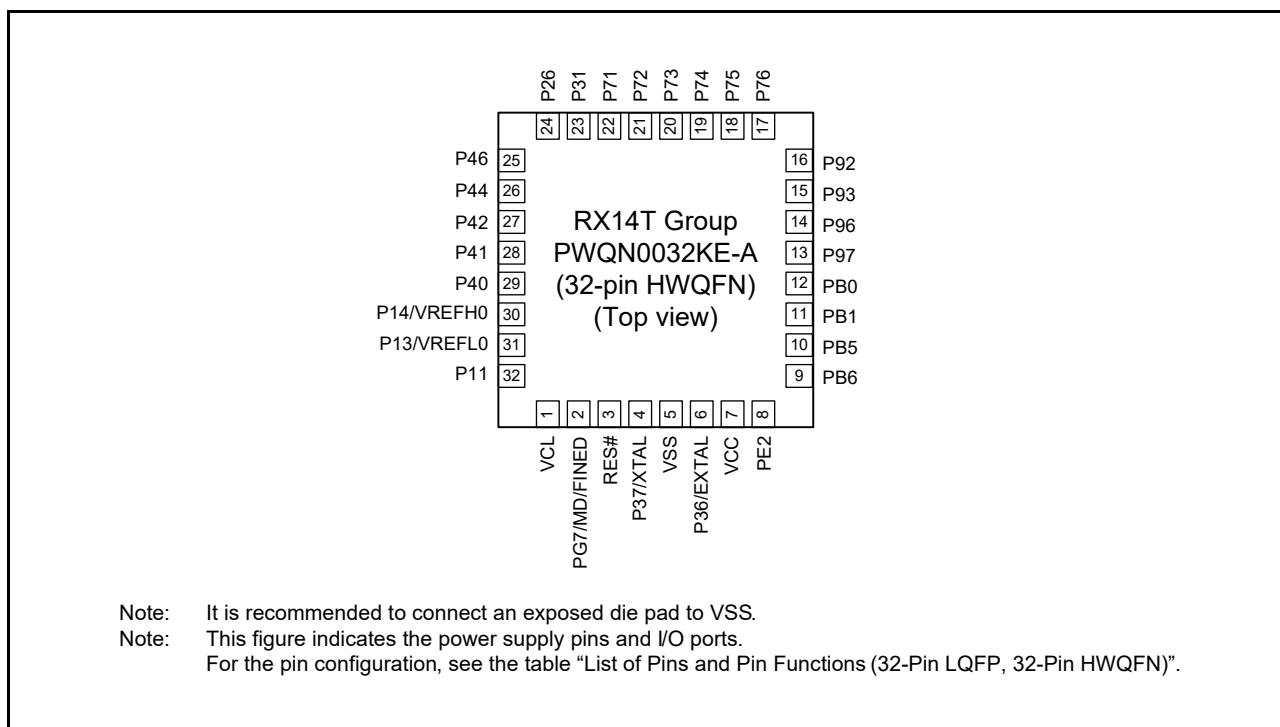


Figure 1.9 Pin Assignments of the 32-Pin HWQFN

1.5.8 24-pin HWQFN

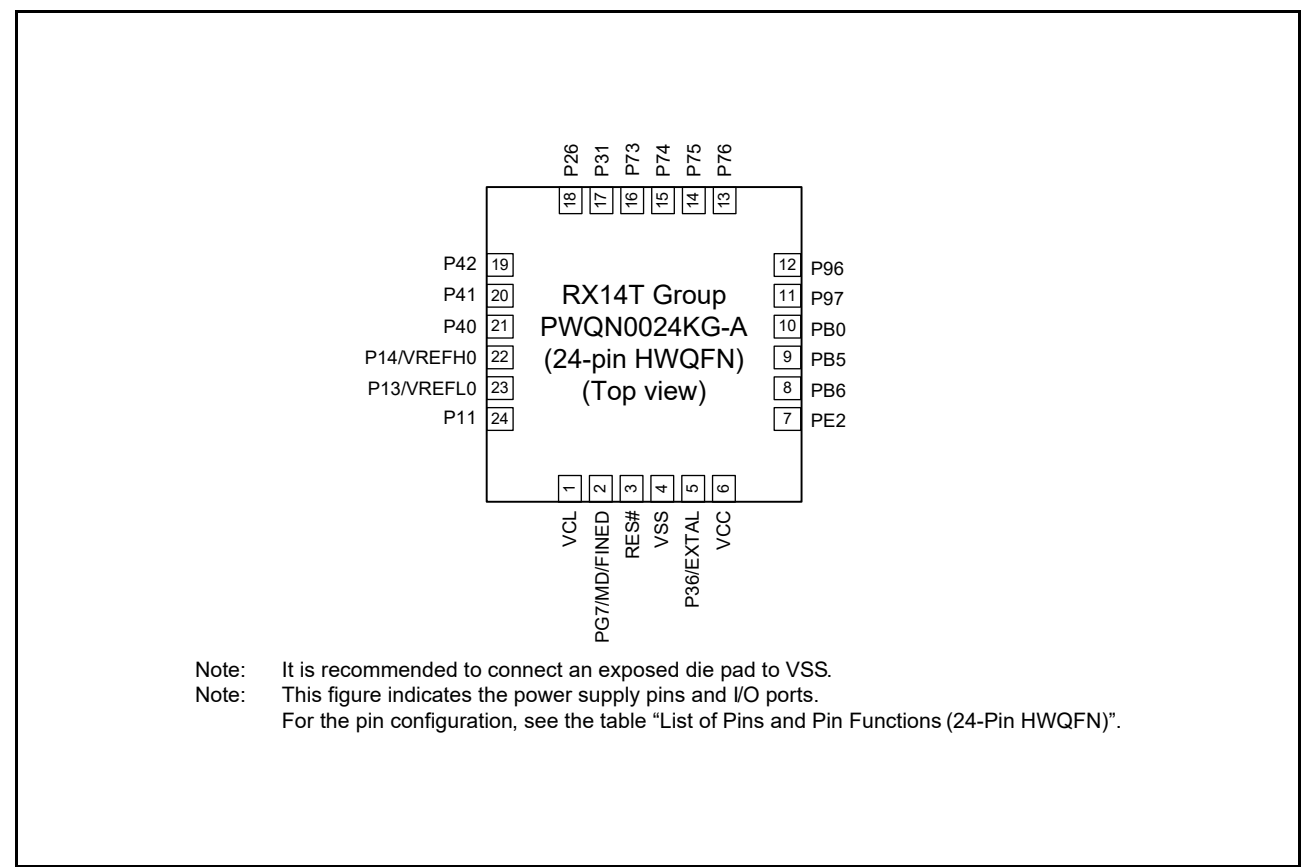


Figure 1.10 Pin Assignments of the 24-Pin HWQFN

1.6 List of Pins and Pin Functions

1.6.1 64-pin LFQFP, 64-pin LQFP

Table 1.5 List of Pins and Pin Functions (64-Pin LFQFP, 64-Pin LQFP) (1/3)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
1		P02		CTS1#/RTS1#/SS1#	IRQ5	ADST0
2		P01			IRQ2	ADST1
3	VCL					
4	CACREF	P00			IRQ4	ADSS0
5	MD/FINED	PG7		CTS5#/RTS5#/SS5#/SCK6/SCK12		ADST0
6	RES#					
7	XTAL	P37	POE11#	TXD6/SMOSI6/SSDA6/TXD12/SMOSI12/SSDA12/TXD12/SIOX12	IRQ0	ADSS1
8	VSS					
9	EXTAL	P36	MTIOC0A/MTIOC4A/MTIOC1A/MTIOC0A#/MTIOC4A#/MTIOC1A#/GTIOC0A/GTICPP00/GTIOC0A#/POE12#	RXD6/SMISO6/SSCL6/RXD12/SMISO12/SSCL12/RDX12	IRQ3	ADST1
10	VCC					
11		PE2	GTETRG/TMCI1/TMRI0/POE10#		IRQ0/NMI	
12		PD7	TMRI1			
13		PD6	MTIOC0D/MTIOC4D/MTIOC0D#/MTIOC4D#/GTIOC2B/GTIOC2B#/TMO1	CTS1#/RTS1#/SS1#	IRQ5	ADST0
14		PD5	MTIOC0C/MTIOC4C/MTIOC0C#/MTIOC4C#/GTIOC2A/GTIOC2A#/TMRI0	RXD1/SMISO1/SSCL1	IRQ3	ADSS0
15		PD4	MTIOC0B/MTIOC4B/MTIOC0B#/MTIOC4B#/GTIOC1B/GTIOC1B#/TMCI0	SCK1	IRQ2	
16		PD3	MTIOC0A/MTIOC4A/MTIOC0A#/MTIOC4A#/GTIOC1A/GTIOC1A#/TMO0	TXD1/SMOSI1/SSDA1		
17		PB7		SCK5/CTS6#/RTS6#/SS6#		
18		PB6	MTIOC3C/MTCLKD/MTIOC2B/MTIOC3C#/MTCLKD#/MTIOC2B#/GTIOC2B/GTICPP00/GTIOC2B#/TMCI0/POE11#	RXD5/SMISO5/SSCL5/RXD1/SMISO1/SSCL1	IRQ5	ADSS0
19		PB5	MTIOC1B/MTIOC3A/MTIOC2A/MTIOC1B#/MTIOC3A#/MTIOC2A#/GTIOC1A/GTIOC2A/GTIOC1A#/GTIOC2A#/TMO0/POE12#	TXD5/SMOSI5/SSDA5/TXD1/SMOSI1/SSDA1	IRQ7	ADSS1
20		PB4		RXD6/SMISO6/SSCL6		ADTRG0#
21		PB3	POE8#	TXD6/SMOSI6/SSDA6	IRQ3	ADTRG1#
22		PB2	MTIC5W/MTIC5W#	SCK6		
23	CACREF	PB1	MTIOC0A/MTIOC4D/MTIOC0A#/MTIOC4D#/GTIOC2B/GTIOC2B#	SCK5/SCK12	IRQ2	
24		PB0	MTIOC0B/MTCLKC/MTIOC4C/MTIOC0B#/MTCLKC#/MTIOC4C#/GTIOC1B/GTADSM0/GTIOC1B#/TMRI1	TXD5/SMOSI5/SSDA5/SCK1/SDA0		ADSM0
25		P97	MTIOC0C/MTCLKA/MTIOC3D/MTIOC0C#/MTCLKA#/MTIOC3D#/GTIOC0B/GTADSM1/GTIOC0B#/GTETRG/TMCI1	RXD5/SMISO5/SSCL5/SCL0	IRQ2	ADSM1

Table 1.5 List of Pins and Pin Functions (64-Pin LFQFP, 64-Pin LQFP) (2/3)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
26	CLKOUT	P96	MTIOC0D/MTCLKB/ MTIOC4B/MTIOC0D#/ MTCLKB#/MTIOC4B#/ GTIOC2A/GTIOC2A#/ TMO1/POE0#	SCK5/TXD12/SMOSI12/SSDA12/ TXDX12/SIOX12		
27		P95	MTIOC1B/MTIOC2A/ MTIOC4A/MTIOC1B#/ MTIOC2A#/MTIOC4A#/ GTIOC1A/GTIOC1A#	CTS12#/RTS12#/SS12#		
28		P94	MTIOC1A/MTIOC2B/ MTIOC3B/MTIOC1A#/ MTIOC2B#/MTIOC3B#/ GTIOC0A/GTIOC0A#	CTS5#/RTS5#/SS5#/TXD12/SMOSI12/ SSDA12/TXDX12/SIOX12	IRQ4	
29		P93	MTIOC0C/MTIOC2B/ MTCLKA/MTIOC0C#/ MTIOC2B#/MTCLKA#/ GTIOC1B/GTIOC1B#/TMO1	RXD12/SMISO12/SSCL12/RXDX12	IRQ1	ADTRG1#
30		P92	MTIOC0B/MTIOC1A/ MTIOC0B#/MTIOC1A#/ GTIOC1A/GTIOC1A#/TMR1	SCK5/SCK12	IRQ0	ADTRG0#
31		P91	MTIC5U/MTIC5U#/TMC1			
32		P90	MTIC5V/MTIC5V#		IRQ6	
33	CACREF	P76	MTIOC4D/MTIOC4D#/ GTIOC2B/GTIOC2B#/TMR2	RXD12/SMISO12/SSCL12/RXDX12		
34		P75	MTIOC4C/MTIOC4C#/ GTIOC1B/GTIOC1B#/ TMC2/POE8#	SCK12		
35		P74	MTIOC3D/MTIOC3D#/ GTIOC0B/GTIOC0B#/TMO2	CTS6#/RTS6#/SS6#/CTS12#/RTS12#/ SS12#		
36		P73	MTIOC4B/MTIOC4B#/ GTIOC2A/GTIOC2A#/TMR3	SCK6		
37		P72	MTIOC4A/MTIOC4A#/ GTIOC1A/GTIOC1A#	RXD1/SMISO1/SSCL1	IRQ7	
38		P71	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#	TXD1/SMOSI1/SSDA1	IRQ6	
39		P70	GTCPP00/POE0#	SCK1	IRQ5	
40		P33	MTIOC3A/MTCLKA/ MTIOC0C/MTIOC3A#/ MTCLKA#/MTIOC0C#			
41		P32	MTIOC3C/MTCLKB/ MTIOC0D/MTIOC3C#/ MTCLKB#/MTIOC0D#/ POE10#			
42		P31	MTIOC4A/MTIOC4A#/ GTIOC0B/GTIOC1A/ GTIOC0B#/GTIOC1A#/ TMC3	TXD6/SMOSI6/SSDA6/SDA0*1		
43		P30	MTIOC0A/MTCLKC/ MTIOC0A#/MTCLKC#/ POE11#	SCK6		
44		P26	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#/TMO3	RXD6/SMISO6/SSCL6/SCL0*1	IRQ1	
45		P25	MTIOC0B/MTCLKD/ MTIOC4B/MTIOC0B#/ MTCLKD#/MTIOC4B#/ GTETRG/TMR3/POE12#			
46	CLKOUT	P24	MTIC5U/MTIC5U#/ GTIOC2B/GTCPP00/ GTIOC2B#/TMC2	RXD5/SMISO5/SSCL5	IRQ3	COMP0
47	CACREF	P23	MTIC5V/MTCLKA/MTIC5V#/ MTCLKA#/GTIOC2A/ GTADSM0/GTIOC2A#/ TMO2	TXD5/SMOSI5/SSDA5	IRQ4	ADSS0/COMP1

Table 1.5 List of Pins and Pin Functions (64-Pin LFQFP, 64-Pin LQFP) (3/3)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
48		P22	MTIC5W/MTCLKB/ MTIC5W#/MTCLKB#/ GTADSM1/TMRI2	SCK5	IRQ2	ADSS1/COMP2/ PGAOUT0
49		P47				AN007/PGAGND
50		P46	GTADSM0			AN006/ADSS0/ CMPC13
51		P45				AN005
52		P44	GTADSM1			AN004/ADSS1/DA0/ CMPC03/CVREFC0
53		P43				AN003
54		P42				AN002/CMPC20/ CMPC21/PGAIN02
55		P41				AN001/CMPC10/ CMPC11/PGAIN01
56		P40				AN000/CMPC00/ CMPC01/PGAIN00
57		P15				AN107
58	VREFH0	P14			IRQ6	AN106/ADST0/CMPC22
59	VREFL0	P13			IRQ4	AN105/CMPC12
60		P12			IRQ7	AN104
61		P11	MTIOC3A/MTCLKC/ MTIOC4C/MTIOC3A#/ MTCLKC#/MTIOC4C#/ TMO3/TMO2/POE8#	CTS1#/RTS1#/SS1#/TXD6/SMOSI6/ SSDA6/TXD12/SMOSI12/SSDA12/ TXDX12/SIOX12	IRQ1	AN102/CMPC02
62		P10	MTCLKD/MTCLKD#/TMRI3/ TMO1		IRQ0	AN103
63		P04	MTIOC1A/MTIOC1A#/ GTIOC2A/GTIOC2A#/ TMC13/TMO0			ADSS1
64		P03	MTIOC1B/MTIOC1B#/ GTIOC2B/GTIOC2B#			ADTRG0#

Note 1. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

1.6.2 52-pin LQFP

Table 1.6 List of Pins and Pin Functions (52-Pin LQFP) (1/3)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
1		P02		CTS1#/RTS1#/SS1#	IRQ5	ADST0
2	VCL					
3	MD/FINED	PG7		CTS5#/RTS5#/SS5#/SCK6/SCK12		ADST0
4	RES#					
5	XTAL	P37	POE11#	TXD6/SMOSI6/SSDA6/TXD12/SMOSI12/SSDA12/TXD12/SIOX12	IRQ0	ADSS1
6	VSS					
7	EXTAL	P36	MTIOC0A/MTIOC4A/MTIOC1A/MTIOC0A#/MTIOC4A#/MTIOC1A#/GTIOC0A/GTICPP00/GTIOC0A#/POE12#	RXD6/SMISO6/SSCL6/RXD12/SMISO12/SSCL12/RXD12	IRQ3	ADST1
8	VCC					
9		PE2	GTETRG/TMCI1/TMRI0/POE10#		IRQ0/NMI	
10		PD6	MTIOC0D/MTIOC4D/MTIOC0D#/MTIOC4D#/GTIOC2B/GTIOC2B#/TMO1	CTS1#/RTS1#/SS1#	IRQ5	ADST0
11		PD5	MTIOC0C/MTIOC4C/MTIOC0C#/MTIOC4C#/GTIOC2A/GTIOC2A#/TMRI0	RXD1/SMISO1/SSCL1	IRQ3	ADSS0
12		PD4	MTIOC0B/MTIOC4B/MTIOC0B#/MTIOC4B#/GTIOC1B/GTIOC1B#/TMCI0	SCK1	IRQ2	
13		PD3	MTIOC0A/MTIOC4A/MTIOC0A#/MTIOC4A#/GTIOC1A/GTIOC1A#/TMO0	TXD1/SMOSI1/SSDA1		
14		PB7		SCK5/CTS6#/RTS6#/SS6#		
15		PB6	MTIOC3C/MTCLKD/MTIOC2B/MTIOC3C#/MTCLKD#/MTIOC2B#/GTIOC2B/GTICPP00/GTIOC2B#/TMCI0/POE11#	RXD5/SMISO5/SSCL5/RXD1/SMISO1/SSCL1	IRQ5	ADSS0
16		PB5	MTIOC1B/MTIOC3A/MTIOC2A/MTIOC1B#/MTIOC3A#/MTIOC2A#/GTIOC1A/GTIOC2A/GTIOC1A#/GTIOC2A#/TMO0/POE12#	TXD5/SMOSI5/SSDA5/TXD1/SMOSI1/SSDA1	IRQ7	ADSS1
17		PB4		RXD6/SMISO6/SSCL6		ADTRG0#
18		PB3	POE8#	TXD6/SMOSI6/SSDA6	IRQ3	ADTRG1#
19	CACREF	PB1	MTIOC0A/MTIOC4D/MTIOC0A#/MTIOC4D#/GTIOC2B/GTIOC2B#	SCK5/SCK12	IRQ2	
20		PB0	MTIOC0B/MTCLKC/MTIOC4C/MTIOC0B#/MTCLKC#/MTIOC4C#/GTIOC1B/GTADSM0/GTIOC1B#/TMRI1	TXD5/SMOSI5/SSDA5/SCK1/SDA0		ADSM0
21		P97	MTIOC0C/MTCLKA/MTIOC3D/MTIOC0C#/MTCLKA#/MTIOC3D#/GTIOC0B/GTADSM1/GTIOC0B#/GTETRG/TMCI1	RXD5/SMISO5/SSCL5/SCL0	IRQ2	ADSM1
22	CLKOUT	P96	MTIOC0D/MTCLKB/MTIOC4B/MTIOC0D#/MTCLKB#/MTIOC4B#/GTIOC2A/GTIOC2A#/TMO1/POE0#	SCK5/TXD12/SMOSI12/SSDA12/TXD12/SIOX12		

Table 1.6 List of Pins and Pin Functions (52-Pin LQFP) (2/3)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
23		P95	MTIOC1B/MTIOC2A/ MTIOC4A/MTIOC1B#/ MTIOC2A#/MTIOC4A#/ GTIOC1A/GTIOC1A#	CTS12#/RTS12#/SS12#		
24		P94	MTIOC1A/MTIOC2B/ MTIOC3B/MTIOC1A#/ MTIOC2B#/MTIOC3B#/ GTIOC0A/GTIOC0A#	CTS5#/RTS5#/SS5#/TXD12/SMOSI12/ SSDA12/TXD12/SIOX12	IRQ4	
25		P93	MTIOC0C/MTIOC2B/ MTCLKA/MTIOC0C#/ MTIOC2B#/MTCLKA#/ GTIOC1B/GTIOC1B#/TMO1	RXD12/SMISO12/SSCL12/RXD12	IRQ1	ADTRG1#
26		P92	MTIOC0B/MTIOC1A/ MTIOC0B#/MTIOC1A#/ GTIOC1A/GTIOC1A#/TMR1	SCK5/SCK12	IRQ0	ADTRG0#
27	CACREF	P76	MTIOC4D/MTIOC4D#/ GTIOC2B/GTIOC2B#/TMR12	RXD12/SMISO12/SSCL12/RXD12		
28		P75	MTIOC4C/MTIOC4C#/ GTIOC1B/GTIOC1B#/ TMC12/POE8#	SCK12		
29		P74	MTIOC3D/MTIOC3D#/ GTIOC0B/GTIOC0B#/TMO2	CTS6#/RTS6#/SS6#/CTS12#/RTS12#/ SS12#		
30		P73	MTIOC4B/MTIOC4B#/ GTIOC2A/GTIOC2A#/TMR13	SCK6		
31		P72	MTIOC4A/MTIOC4A#/ GTIOC1A/GTIOC1A#	RXD1/SMISO1/SSCL1	IRQ7	
32		P71	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#	TXD1/SMOSI1/SSDA1	IRQ6	
33		P70	GTCPP00/POE0#	SCK1	IRQ5	
34		P33	MTIOC3A/MTCLKA/ MTIOC0C/MTIOC3A#/ MTCLKA#/MTIOC0C#			
35		P31	MTIOC4A/MTIOC4A#/ GTIOC0B/GTIOC1A/ GTIOC0B#/GTIOC1A#/ TMC13	TXD6/SMOSI6/SSDA6/SDA0*1		
36		P26	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#/TMO3	RXD6/SMISO6/SSCL6/SCL0*1	IRQ1	
37	CLKOUT	P24	MTIC5U/MTIC5U#/ GTIOC2B/GTCPP00/ GTIOC2B#/TMC12	RXD5/SMISO5/SSCL5	IRQ3	COMP0
38	CACREF	P23	MTIC5V/MTCLKA/MTIC5V#/ MTCLKA#/GTIOC2A/ GTADSM0/GTIOC2A#/ TMO2	TXD5/SMOSI5/SSDA5	IRQ4	ADSS0/COMP1
39		P22	MTIC5W/MTCLKB/ MTIC5W#/MTCLKB#/ GTADSM1/TMR12	SCK5	IRQ2	ADSS1/COMP2/ PGAOUT0
40		P47				AN007/PGAGND
41		P46	GTADSM0			AN006/ADSS0/ CMPC13
42		P45				AN005
43		P44	GTADSM1			AN004/ADSS1/DA0/ CMPC03/CVREFC0
44		P43				AN003
45		P42				AN002/CMPC20/ CMPC21/PGAIN02
46		P41				AN001/CMPC10/ CMPC11/PGAIN01
47		P40				AN000/CMPC00/ CMPC01/PGAIN00
48	VREFH0	P14			IRQ6	AN106/ADST0/CMPC22
49	VREFL0	P13			IRQ4	AN105/CMPC12

Table 1.6 List of Pins and Pin Functions (52-Pin LQFP) (3/3)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
50		P11	MTIOC3A/MTCLKC/ MTIOC4C/MTIOC3A#/ MTCLKC#/MTIOC4C#/ TMO3/TMO2/POE8#	CTS1#/RTS1#/SS1#/TXD6/SMOSI6/ SSDA6/TXD12/SMOSI12/SSDA12/ TXDX12/SIOX12	IRQ1	AN102/CMPC02
51		P10	MTCLKD/MTCLKD#/TMR13/ TMO1		IRQ0	AN103
52		P04	MTIOC1A/MTIOC1A#/ GTIOC2A/GTIOC2A#/ TMC13/TMO0			ADSS1

Note 1. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

1.6.3 48-pin LFQFP, 48-pin HWQFN

Table 1.7 List of Pins and Pin Functions (48-Pin LFQFP, 48-Pin HWQFN) (1/2)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
1	VCL					
2	MD/FINED	PG7		CTS5#/RTS5#/SS5#/SCK6/SCK12		ADST0
3	RES#					
4	XTAL	P37	POE11#	TXD6/SMOSI6/SSDA6/TXD12/SMOSI12/SSDA12/TXDX12/SIOX12	IRQ0	ADSS1
5	VSS					
6	EXTAL	P36	MTIOC0A/MTIOC4A/MTIOC1A/MTIOC0A#/MTIOC4A#/MTIOC1A#/GTIOC0A/GTICPP00/GTIOC0A#/POE12#	RXD6/SMISO6/SSCL6/RXD12/SMISO12/SSCL12/RXDX12	IRQ3	ADST1
7	VCC					
8		PE2	GTETRGA/TMC11/TMRI0/POE10#		IRQ0/NMI	
9		PD6	MTIOC0D/MTIOC4D/MTIOC0D#/MTIOC4D#/GTIOC2B/GTIOC2B#/TMO1	CTS1#/RTS1#/SS1#	IRQ5	ADST0
10		PD5	MTIOC0C/MTIOC4C/MTIOC0C#/MTIOC4C#/GTIOC2A/GTIOC2A#/TMRI0	RXD1/SMISO1/SSCL1	IRQ3	ADSS0
11		PD4	MTIOC0B/MTIOC4B/MTIOC0B#/MTIOC4B#/GTIOC1B/GTIOC1B#/TMC10	SCK1	IRQ2	
12		PD3	MTIOC0A/MTIOC4A/MTIOC0A#/MTIOC4A#/GTIOC1A/GTIOC1A#/TMO0	TXD1/SMOSI1/SSDA1		
13		PB6	MTIOC3C/MTCLKD/MTIOC2B/MTIOC3C#/MTCLKD#/MTIOC2B#/GTIOC2B/GTICPP00/GTIOC2B#/TMC10/POE11#	RXD5/SMISO5/SSCL5/RXD1/SMISO1/SSCL1	IRQ5	ADSS0
14		PB5	MTIOC1B/MTIOC3A/MTIOC2A/MTIOC1B#/MTIOC3A#/MTIOC2A#/GTIOC1A/GTIOC2A/GTIOC1A#/GTIOC2A#/TMO0/POE12#	TXD5/SMOSI5/SSDA5/TXD1/SMOSI1/SSDA1	IRQ7	ADSS1
15		PB4		RXD6/SMISO6/SSCL6		ADTRG0#
16		PB3	POE8#	TXD6/SMOSI6/SSDA6	IRQ3	ADTRG1#
17	CACREF	PB1	MTIOC0A/MTIOC4D/MTIOC0A#/MTIOC4D#/GTIOC2B/GTIOC2B#	SCK5/SCK12	IRQ2	
18		PB0	MTIOC0B/MTCLKC/MTIOC4C/MTIOC0B#/MTCLKC#/MTIOC4C#/GTIOC1B/GTADSM0/GTIOC1B#/TMRI1	TXD5/SMOSI5/SSDA5/SCK1/SDA0		ADSM0
19		P97	MTIOC0C/MTCLKA/MTIOC3D/MTIOC0C#/MTCLKA#/MTIOC3D#/GTIOC0B/GTADSM1/GTIOC0B#/GTETRGA/TMC11	RXD5/SMISO5/SSCL5/SCL0	IRQ2	ADSM1
20	CLKOUT	P96	MTIOC0D/MTCLKB/MTIOC4B/MTIOC0D#/MTCLKB#/MTIOC4B#/GTIOC2A/GTIOC2A#/TMO1/POE0#	SCK5/TXD12/SMOSI12/SSDA12/TXDX12/SIOX12		
21		P95	MTIOC1B/MTIOC2A/MTIOC4A/MTIOC1B#/MTIOC2A#/MTIOC4A#/GTIOC1A/GTIOC1A#	CTS12#/RTS12#/SS12#		
22		P94	MTIOC1A/MTIOC2B/MTIOC3B/MTIOC1A#/MTIOC2B#/MTIOC3B#/GTIOC0A/GTIOC0A#	CTS5#/RTS5#/SS5#/TXD12/SMOSI12/SSDA12/TXDX12/SIOX12	IRQ4	
23		P93	MTIOC0C/MTIOC2B/MTCLKA/MTIOC0C#/MTIOC2B#/MTCLKA#/GTIOC1B/GTIOC1B#/TMO1	RXD12/SMISO12/SSCL12/RXDX12	IRQ1	ADTRG1#
24		P92	MTIOC0B/MTIOC1A/MTIOC0B#/MTIOC1A#/GTIOC1A/GTIOC1A#/TMRI1	SCK5/SCK12	IRQ0	ADTRG0#
25	CACREF	P76	MTIOC4D/MTIOC4D#/GTIOC2B/GTIOC2B#/TMRI2	RXD12/SMISO12/SSCL12/RXDX12		

Table 1.7 List of Pins and Pin Functions (48-Pin LFQFP, 48-Pin HWQFN) (2/2)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, IIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
26		P75	MTIOC4C/MTIOC4C#/GTIOC1B/ GTIOC1B#/TMC12/POE8#	SCK12		
27		P74	MTIOC3D/MTIOC3D#/GTIOC0B/ GTIOC0B#/TMO2	CTS6#/RTS6#/SS6#/CTS12#/ RTS12#/SS12#		
28		P73	MTIOC4B/MTIOC4B#/GTIOC2A/ GTIOC2A#/TMR13	SCK6		
29		P72	MTIOC4A/MTIOC4A#/GTIOC1A/ GTIOC1A#	RXD1/SMISO1/SSCL1	IRQ7	
30		P71	MTIOC3B/MTIOC3B#/GTIOC0A/ GTIOC0A#	TXD1/SMOSI1/SSDA1	IRQ6	
31		P70	GTCPPO0/POE0#	SCK1	IRQ5	
32		P31	MTIOC4A/MTIOC4A#/GTIOC0B/ GTIOC1A/GTIOC0B#/GTIOC1A#/ TMC13	TXD6/SMOSI6/SSDA6/SDA0*1		
33		P26	MTIOC3B/MTIOC3B#/GTIOC0A/ GTIOC0A#/TMO3	RXD6/SMISO6/SSCL6/SCL0*1	IRQ1	
34	CLKOUT	P24	MTIC5U/MTIC5U#/GTIOC2B/ GTCPPO0/GTIOC2B#/TMC12	RXD5/SMOSI5/SSCL5	IRQ3	COMP0
35	CACREF	P23	MTIC5V/MTCLKA/MTIC5V#/ MTCLKA#/GTIOC2A/GTADSM0/ GTIOC2A#/TMO2	TXD5/SMOSI5/SSDA5	IRQ4	ADSS0/ COMP1
36		P22	MTIC5W/MTCLKB/MTIC5W#/ MTCLKB#/GTADSM1/TMR12	SCK5	IRQ2	ADSS1/ COMP2/ PGAOUT0
37		P47				AN007/ PGAGND
38		P46	GTADSM0			AN006/ ADSS0/ CMPC13
39		P45				AN005
40		P44	GTADSM1			AN004/ ADSS1/DA0/ CMPC03/ CVREFC0
41		P43				AN003
42		P42				AN002/ CMPC20/ CMPC21/ PGAIN02
43		P41				AN001/ CMPC10/ CMPC11/ PGAIN01
44		P40				AN000/ CMPC00/ CMPC01/ PGAIN00
45	VREFH0	P14			IRQ6	AN106/ADST0/ CMPC22
46	VREFL0	P13			IRQ4	AN105/ CMPC12
47		P11	MTIOC3A/MTCLKC/MTIOC4C/ MTIOC3A#/MTCLKC#/MTIOC4C#/ TMO3/TMO2/POE8#	CTS1#/RTS1#/SS1#/TXD6/SMOSI6/ SSDA6/TXD12/SMOSI12/SSDA12/ TXDX12/SIOX12	IRQ1	AN102/ CMPC02
48		P10	MTCLKD/MTCLKD#/TMR13/TMO1		IRQ0	AN103

Note 1. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

1.6.4 44-pin LQFP

Table 1.8 List of Pins and Pin Functions (44-Pin LQFP) (1/2)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
1	VCL					
2	MD/FINED	PG7		CTS5#/RTS5#/SS5#/SCK6/SCK12		ADST0
3	RES#					
4	VSS					
5	EXTAL	P36	MTIOC0A/MTIOC4A/ MTIOC1A/MTIOC0A#/ MTIOC4A#/MTIOC1A#/ GTIOC0A/GTICPP00/ GTIOC0A#/POE12#	RXD6/SMISO6/SSCL6/RXD12/ SMISO12/SSCL12/RDX12	IRQ3	ADST1
6	VCC					
7		PE2	GTETRG/TMCI1/TMRI0/ POE10#		IRQ0/ NMI	
8		PD6	MTIOC0D/MTIOC4D/ MTIOC0D#/MTIOC4D#/ GTIOC2B/GTIOC2B#/TMO1	CTS1#/RTS1#/SS1#	IRQ5	ADST0
9		PD5	MTIOC0C/MTIOC4C/ MTIOC0C#/MTIOC4C#/ GTIOC2A/GTIOC2A#/TMRI0	RXD1/SMISO1/SSCL1	IRQ3	ADSS0
10		PD4	MTIOC0B/MTIOC4B/ MTIOC0B#/MTIOC4B#/ GTIOC1B/GTIOC1B#/TMCI0	SCK1	IRQ2	
11		PD3	MTIOC0A/MTIOC4A/ MTIOC0A#/MTIOC4A#/ GTIOC1A/GTIOC1A#/TMO0	TXD1/SMOSI1/SSDA1		
12		PB6	MTIOC3C/MTCLKD/ MTIOC2B/MTIOC3C#/ MTCLKD#/MTIOC2B#/ GTIOC2B/GTICPP00/ GTIOC2B#/TMCI0/POE11#	RXD5/SMISO5/SSCL5/RXD1/SMISO1/ SSCL1	IRQ5	ADSS0
13		PB5	MTIOC1B/MTIOC3A/ MTIOC2A/MTIOC1B#/ MTIOC3A#/MTIOC2A#/ GTIOC1A/GTIOC2A/ GTIOC1A#/GTIOC2A#/ TMO0/POE12#	TXD5/SMOSI5/SSDA5/TXD1/SMOSI1/ SSDA1	IRQ7	ADSS1
14		PB3	POE8#	TXD6/SMOSI6/SSDA6	IRQ3	ADTRG1#
15	CACREF	PB1	MTIOC0A/MTIOC4D/ MTIOC0A#/MTIOC4D#/ GTIOC2B/GTIOC2B#	SCK5/SCK12	IRQ2	
16		PB0	MTIOC0B/MTCLKC/ MTIOC4C/MTIOC0B#/ MTCLKC#/MTIOC4C#/ GTIOC1B/GTADSM0/ GTIOC1B#/TMRI1	TXD5/SMOSI5/SSDA5/SCK1/SDA0		ADSM0
17		P97	MTIOC0C/MTCLKA/ MTIOC3D/MTIOC0C#/ MTCLKA#/MTIOC3D#/ GTIOC0B/GTADSM1/ GTIOC0B#/GTETRG/ TMCI1	RXD5/SMISO5/SSCL5/SCL0	IRQ2	ADSM1
18	CLKOUT	P96	MTIOC0D/MTCLKB/ MTIOC4B/MTIOC0D#/ MTCLKB#/MTIOC4B#/ GTIOC2A/GTIOC2A#/ TMO1/POE0#	SCK5/TXD12/SMOSI12/SSDA12/ TXDX12/SIOX12		
19		P95	MTIOC1B/MTIOC2A/ MTIOC4A/MTIOC1B#/ MTIOC2A#/MTIOC4A#/ GTIOC1A/GTIOC1A#	CTS12#/RTS12#/SS12#		
20		P94	MTIOC1A/MTIOC2B/ MTIOC3B/MTIOC1A#/ MTIOC2B#/MTIOC3B#/ GTIOC0A/GTIOC0A#	CTS5#/RTS5#/SS5#/TXD12/SMOSI12/ SSDA12/TXDX12/SIOX12	IRQ4	

Table 1.8 List of Pins and Pin Functions (44-Pin LQFP) (2/2)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
21		P93	MTIOC0C/MTIOC2B/ MTCLKA/MTIOC0C#/ MTIOC2B#/MTCLKA#/ GTIOC1B/GTIOC1B#/TMO1	RXD12/SMISO12/SSCL12/RDX12	IRQ1	ADTRG1#
22		P92	MTIOC0B/MTIOC1A/ MTIOC0B#/MTIOC1A#/ GTIOC1A/GTIOC1A#/TMR1	SCK5/SCK12	IRQ0	ADTRG0#
23	CACREF	P76	MTIOC4D/MTIOC4D#/ GTIOC2B/GTIOC2B#/TMR12	RXD12/SMISO12/SSCL12/RDX12		
24		P75	MTIOC4C/MTIOC4C#/ GTIOC1B/GTIOC1B#/ TMC12/POE8#	SCK12		
25		P74	MTIOC3D/MTIOC3D#/ GTIOC0B/GTIOC0B#/TMO2	CTS6#/RTS6#/SS6#/CTS12#/RTS12#/ SS12#		
26		P73	MTIOC4B/MTIOC4B#/ GTIOC2A/GTIOC2A#/TMR13	SCK6		
27		P72	MTIOC4A/MTIOC4A#/ GTIOC1A/GTIOC1A#	RXD1/SMISO1/SSCL1	IRQ7	
28		P71	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#	TXD1/SMOSI1/SSDA1	IRQ6	
29		P31	MTIOC4A/MTIOC4A#/ GTIOC0B/GTIOC1A/ GTIOC0B#/GTIOC1A#/ TMC13	TXD6/SMOSI6/SSDA6/SDA0*1		
30		P26	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#/TMO3	RXD6/SMISO6/SSCL6/SCL0*1	IRQ1	
31	CLKOUT	P24	MTIC5U/MTIC5U#/ GTIOC2B/GTIOC2B#/ GTIOC2B#/TMC12	RXD5/SMISO5/SSCL5	IRQ3	COMP0
32	CACREF	P23	MTIC5V/MTCLKA/MTIC5V#/ MTCLKA#/GTIOC2A/ GTADSM0/GTIOC2A#/ TMO2	TXD5/SMOSI5/SSDA5	IRQ4	ADSS0/COMP1
33		P22	MTIC5W/MTCLKB/ MTIC5W#/MTCLKB#/ GTADSM1/TMR12	SCK5	IRQ2	ADSS1/COMP2/ PGAOUT0
34		P47				AN007/PGAGND
35		P46	GTADSM0			AN006/ADSS0/ CMPC13
36		P44	GTADSM1			AN004/ADSS1/DA0/ CMPC03/CVREFC0
37		P43				AN003
38		P42				AN002/CMPC20/ CMPC21/PGAIN02
39		P41				AN001/CMPC10/ CMPC11/PGAIN01
40		P40				AN000/CMPC00/ CMPC01/PGAIN00
41	VREFH0	P14			IRQ6	AN106/ADST0/CMPC22
42	VREFL0	P13			IRQ4	AN105/CMPC12
43		P11	MTIOC3A/MTCLKC/ MTIOC4C/MTIOC3A#/ MTCLKC#/MTIOC4C#/ TMO3/TMO2/POE8#	CTS1#/RTS1#/SS1#/TXD6/SMOSI6/ SSDA6/TXD12/SMOSI12/SSDA12/ TXDX12/SIOX12	IRQ1	AN102/CMPC02
44		P10	MTCLKD/MTCLKD#/TMR13/ TMO1		IRQ0	AN103

Note 1. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

1.6.5 32-pin LQFP, 32-pin HWQFN

Table 1.9 List of Pins and Pin Functions (32-Pin LQFP, 32-Pin HWQFN) (1/2)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
1	VCL					
2	MD/FINED	PG7		CTS5#/RTS5#/SS5#/SCK6/SCK12		ADST0
3	RES#					
4	XTAL	P37	POE11#	TXD6/SMOSI6/SSDA6/TXD12/ SMOSI12/SSDA12/TXDX12/ SIOX12	IRQ0	ADSS1
5	VSS					
6	EXTAL	P36	MTIOC0A/MTIOC4A/ MTIOC1A/MTIOC0A#/ MTIOC4A#/MTIOC1A#/ GTIOC0A/GTCCPO0/ GTIOC0A#/POE12#	RXD6/SMISO6/SSCL6/RXD12/ SMISO12/SSCL12/RXDX12	IRQ3	ADST1
7	VCC					
8		PE2	GTETRGA/TMCI1/TMRI0/ POE10#		IRQ0/ NMI	
9		PB6	MTIOC3C/MTCLKD/ MTIOC2B/MTIOC3C#/ MTCLKD#/MTIOC2B#/ GTIOC2B/GTCCPO0/ GTIOC2B#/TMCI0/POE11#	RXD5/SMISO5/SSCL5/RXD1/ SMISO1/SSCL1	IRQ5	ADSS0
10		PB5	MTIOC1B/MTIOC3A/ MTIOC2A/MTIOC1B#/ MTIOC3A#/MTIOC2A#/ GTIOC1A/GTIOC2A/ GTIOC1A#/GTIOC2A#/ TMO0/POE12#	TXD5/SMOSI5/SSDA5/TXD1/ SMOSI1/SSDA1	IRQ7	ADSS1
11	CACREF	PB1	MTIOC0A/MTIOC4D/ MTIOC0A#/MTIOC4D#/ GTIOC2B/GTIOC2B#	SCK5/SCK12	IRQ2	
12		PB0	MTIOC0B/MTCLKC/ MTIOC4C/MTIOC0B#/ MTCLKC#/MTIOC4C#/ GTIOC1B/GTADSM0/ GTIOC1B#/TMRI1	TXD5/SMOSI5/SSDA5/SCK1/SDA0		ADSM0
13		P97	MTIOC0C/MTCLKA/ MTIOC3D/MTIOC0C#/ MTCLKA#/MTIOC3D#/ GTIOC0B/GTADSM1/ GTIOC0B#/GTETRGA/TMCI1	RXD5/SMISO5/SSCL5/SCL0	IRQ2	ADSM1
14	CLKOUT	P96	MTIOC0D/MTCLKB/ MTIOC4B/MTIOC0D#/ MTCLKB#/MTIOC4B#/ GTIOC2A/GTIOC2A#/TMO1/ POE0#	SCK5/TXD12/SMOSI12/SSDA12/ TXDX12/SIOX12		
15		P93	MTIOC0C/MTIOC2B/ MTCLKA/MTIOC0C#/ MTIOC2B#/MTCLKA#/ GTIOC1B/GTIOC1B#/TMO1	RXD12/SMISO12/SSCL12/RXDX12	IRQ1	ADTRG1#
16		P92	MTIOC0B/MTIOC1A/ MTIOC0B#/MTIOC1A#/ GTIOC1A/GTIOC1A#/TMRI1	SCK5/SCK12	IRQ0	ADTRG0#
17	CACREF	P76	MTIOC4D/MTIOC4D#/ GTIOC2B/GTIOC2B#/TMRI2	RXD12/SMISO12/SSCL12/RXDX12		
18		P75	MTIOC4C/MTIOC4C#/ GTIOC1B/GTIOC1B#/TMCI2/ POE8#	SCK12		
19		P74	MTIOC3D/MTIOC3D#/ GTIOC0B/GTIOC0B#/TMO2	CTS6#/RTS6#/SS6#/CTS12#/ RTS12#/SS12#		
20		P73	MTIOC4B/MTIOC4B#/ GTIOC2A/GTIOC2A#/TMRI3	SCK6		
21		P72	MTIOC4A/MTIOC4A#/ GTIOC1A/GTIOC1A#	RXD1/SMISO1/SSCL1	IRQ7	
22		P71	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#	TXD1/SMOSI1/SSDA1	IRQ6	

Table 1.9 List of Pins and Pin Functions (32-Pin LQFP, 32-Pin HWQFN) (2/2)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
23		P31	MTIOC4A/MTIOC4A#/ GTIOC0B/GTIOC1A/ GTIOC0B#/GTIOC1A#/ TMC13	TXD6/SMOSI6/SSDA6/SDA0*1		
24		P26	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#/TMO3	RXD6/SMISO6/SSCL6/SCL0*1	IRQ1	
25		P46	GTADSM0			AN006/ADSS0/ CMPC13
26		P44	GTADSM1			AN004/ADSS1/DA0/ CMPC03/CVREFC0
27		P42				AN002/CMPC20/ CMPC21/PGAIN02
28		P41				AN001/CMPC10/ CMPC11/PGAIN01
29		P40				AN000/CMPC00/ CMPC01/PGAIN00
30	VREFH0	P14			IRQ6	AN106/ADST0/CMPC22
31	VREFL0	P13			IRQ4	AN105/CMPC12
32		P11	MTIOC3A/MTCLKC/ MTIOC4C/MTIOC3A#/ MTCLKC#/MTIOC4C#/TMO3/ TMO2/POE8#	CTS1#/RTS1#/SS1#/TXD6/ SMOSI6/SSDA6/TXD12/SMOSI12/ SSDA12/TXD12/SIOX12	IRQ1	AN102/CMPC02

Note 1. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

1.6.6 24-pin HWQFN

Table 1.10 List of Pins and Pin Functions (24-Pin HWQFN) (1/2)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
1	VCL					
2	MD/FINED	PG7		CTS5#/RTS5#/SS5#/SCK6/SCK12		ADST0
3	RES#					
4	VSS					
5	EXTAL	P36	MTIOC0A/MTIOC4A/ MTIOC1A/MTIOC0A#/ MTIOC4A#/MTIOC1A#/ GTIOC0A/GTCCPO0/ GTIOC0A#/POE12#	RXD6/SMISO6/SSCL6/RXD12/ SMISO12/SSCL12/RXD12	IRQ3	ADST1
6	VCC					
7		PE2	GTETRG/TMCI1/TMRI0/ POE10#		IRQ0/ NMI	
8		PB6	MTIOC3C/MTCLKD/ MTIOC2B/MTIOC3C#/ MTCLKD#/MTIOC2B#/ GTIOC2B/GTCCPO0/ GTIOC2B#/TMCI0/POE11#	RXD5/SMISO5/SSCL5/RXD1/ SMISO1/SSCL1	IRQ5	ADSS0
9		PB5	MTIOC1B/MTIOC3A/ MTIOC2A/MTIOC1B#/ MTIOC3A#/MTIOC2A#/ GTIOC1A/GTIOC2A/ GTIOC1A#/GTIOC2A#/ TMO0/POE12#	TXD5/SMOSI5/SSDA5/TXD1/ SMOSI1/SSDA1	IRQ7	ADSS1
10		PB0	MTIOC0B/MTCLKC/ MTIOC4C/MTIOC0B#/ MTCLKC#/MTIOC4C#/ GTIOC1B/GTADSM0/ GTIOC1B#/TMRI1	TXD5/SMOSI5/SSDA5/SCK1/SDA0		ADSM0
11		P97	MTIOC0C/MTCLKA/ MTIOC3D/MTIOC0C#/ MTCLKA#/MTIOC3D#/ GTIOC0B/GTADSM1/ GTIOC0B#/GTETRG/TMCI1	RXD5/SMISO5/SSCL5/SCL0	IRQ2	ADSM1
12	CLKOUT	P96	MTIOC0D/MTCLKB/ MTIOC4B/MTIOC0D#/ MTCLKB#/MTIOC4B#/ GTIOC2A/GTIOC2A#/TMO1/ POE0#	SCK5/TXD12/SMOSI12/SSDA12/ TXDX12/SIOX12		
13	CACREF	P76	MTIOC4D/MTIOC4D#/ GTIOC2B/GTIOC2B#/TMRI2	RXD12/SMISO12/SSCL12/RXD12		
14		P75	MTIOC4C/MTIOC4C#/ GTIOC1B/GTIOC1B#/TMCI2/ POE8#	SCK12		
15		P74	MTIOC3D/MTIOC3D#/ GTIOC0B/GTIOC0B#/TMO2	CTS6#/RTS6#/SS6#/CTS12#/ RTS12#/SS12#		
16		P73	MTIOC4B/MTIOC4B#/ GTIOC2A/GTIOC2A#/TMRI3	SCK6		
17		P31	MTIOC4A/MTIOC4A#/ GTIOC0B/GTIOC1A/ GTIOC0B#/GTIOC1A#/ TMCI3	TXD6/SMOSI6/SSDA6/SDA0*1		
18		P26	MTIOC3B/MTIOC3B#/ GTIOC0A/GTIOC0A#/TMO3	RXD6/SMISO6/SSCL6/SCL0*1	IRQ1	
19		P42				AN002/CMPC20/ CMPC21/PGAIN02
20		P41				AN001/CMPC10/ CMPC11/PGAIN01
21		P40				AN000/CMPC00/ CMPC01/PGAIN00
22	VREFH0	P14			IRQ6	AN106/ADST0/CMPC22
23	VREFL0	P13			IRQ4	AN105/CMPC12

Table 1.10 List of Pins and Pin Functions (24-Pin HWQFN) (2/2)

Pin No.	Power Supply, Clock, System Control, Others	I/O Port	Timers (MTU, GPTW, TMR, POEG)	Communications (SCI, RIIC)	Interrupt (IRQ, NMI)	Analog (A/D, D/A, CMPC, PGA)
24		P11	MTIOC3A/MTCLKC/ MTIOC4C/MTIOC3A#/ MTCLKC#/MTIOC4C#/TMO3/ TMO2/POE8#	CTS1#/RTS1#/SS1#/TXD6/ SMOSI6/SSDA6/TXD12/SMOSI12/ SSDA12/TXDX12/SIOX12	IRQ1	AN102/CMPC02

Note 1. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

2. CPU

The RXv2 instruction set architecture (RXv2) has upward compatibility with the RXv1 instruction set architecture (RXv1).

- Adoption of variable-length instruction format
As with RXv1, the RXv2 CPU has short formats for frequently used instructions, facilitating the development of efficient programs that take up less memory.
- Powerful instruction set
The RXv2 supports 109 selected instructions. Moreover, DSP instructions and floating-point operation instructions are added, thus realizing high-speed arithmetic processing.
- Versatile addressing modes
The RXv2 CPU has 11 versatile addressing modes, with register-register operations, register-memory operations, and bitwise operations included. Data transfer between memory locations is also possible.

2.1 Features

- Minimum instruction execution rate: One clock cycle
- Address space: 4-Gbyte linear addresses
- Register set of the CPU
General purpose: Sixteen 32-bit registers
Control: Ten 32-bit registers
Accumulator: Two 72-bit registers
- Variable-length instruction format (lengths from one to eight bytes)
- 109 instructions/11 addressing modes
Basic instructions: 75
Floating-point operation instructions: 11
DSP instructions: 23
- Processor modes
Supervisor mode and user mode
- Vector tables
Exception vector table and interrupt vector table
- Data arrangement
Selectable as little endian or big endian

2.2 Register Set of the CPU

The RXv2 CPU has sixteen general-purpose registers, ten control registers, and two accumulator used for DSP instructions.

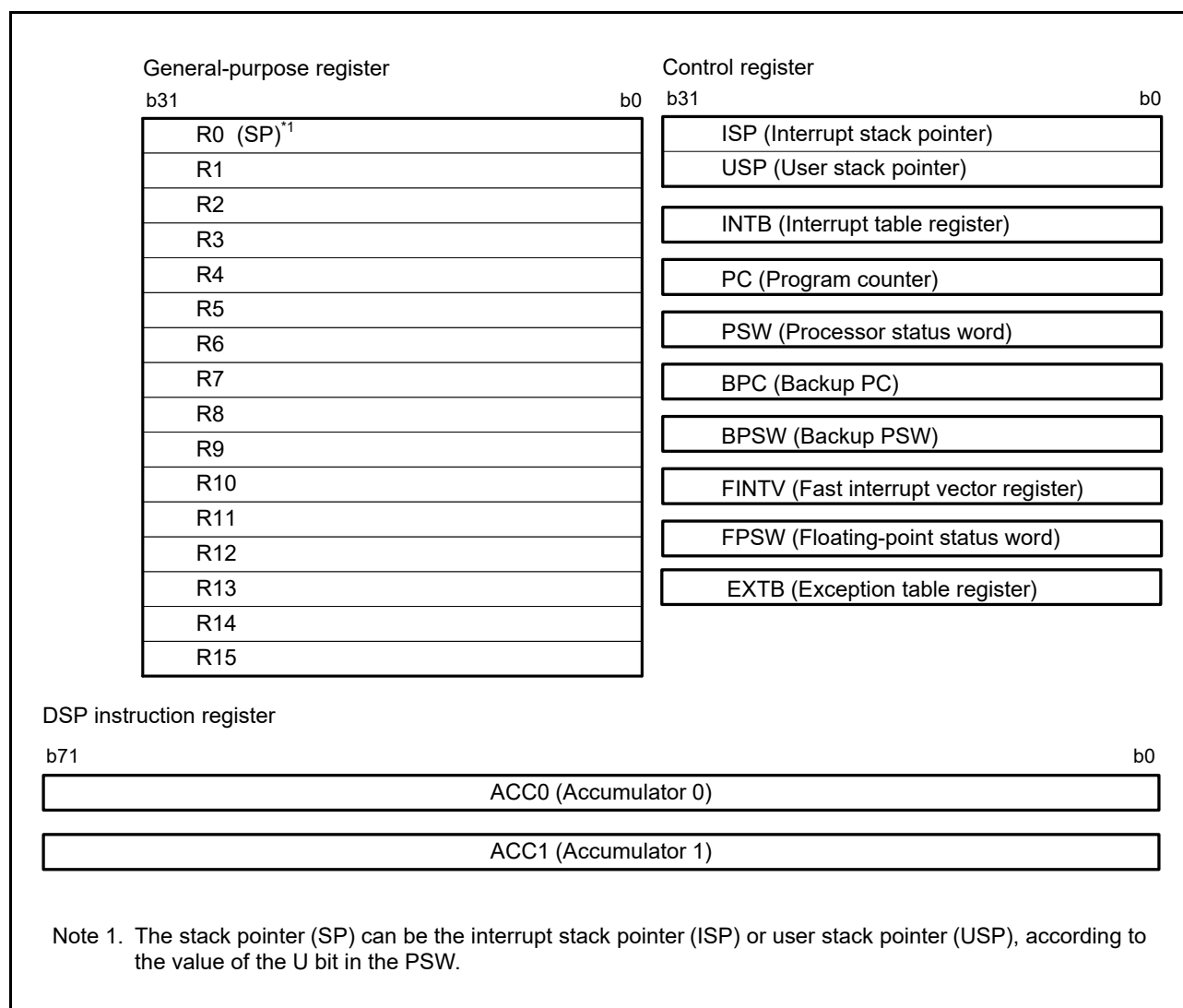


Figure 2.1 Register Set of the CPU

2.2.1 General-Purpose Registers (R0 to R15)

This CPU has sixteen 32-bit general-purpose registers (R0 to R15). R0 to R15 can be used as data registers or address registers.

R0, a general-purpose register, also functions as the stack pointer (SP).

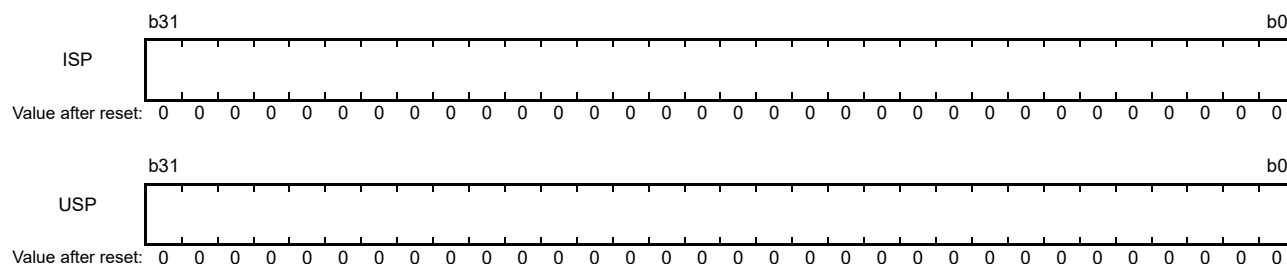
The stack pointer is switched to operate as the interrupt stack pointer (ISP) or user stack pointer (USP) by the value of the stack pointer select bit (U) in the processor status word (PSW).

2.2.2 Control Registers

This CPU has the following ten control registers.

- Interrupt stack pointer (ISP)
- User stack pointer (USP)
- Exception table register (EXTB)
- Interrupt table register (INTB)
- Program counter (PC)
- Processor status word (PSW)
- Backup PC (BPC)
- Backup PSW (BPSW)
- Fast interrupt vector register (FINTV)
- Floating-point status word (FPSW)

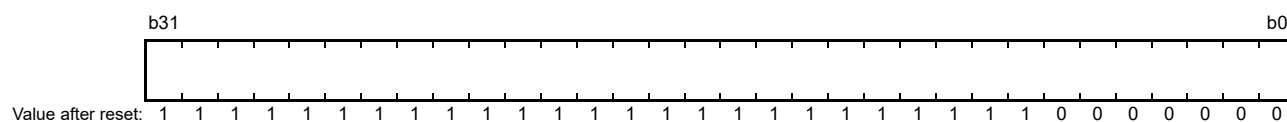
2.2.2.1 Interrupt Stack Pointer (ISP)/User Stack Pointer (USP)



The stack pointer (SP) can be either of two types, the interrupt stack pointer (ISP) or the user stack pointer (USP). Whether the stack pointer operates as the ISP or USP depends on the value of the stack pointer select bit (U) in the processor status word (PSW).

Set the ISP or USP to a multiple of 4 to reduce the number of cycles required to execute interrupt sequences and instructions entailing stack manipulation.

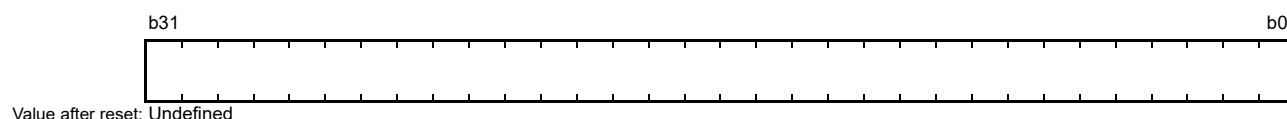
2.2.2.2 Exception Table Register (EXTB)



The exception table register (EXTB) specifies the address where the exception vector table starts.

Set the EXTB to a multiple of 4 to reduce the number of cycles required to execute interrupt sequences and instructions entailing stack manipulation.

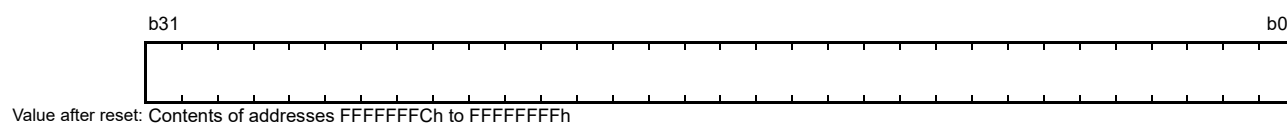
2.2.2.3 Interrupt Table Register (INTB)



The interrupt table register (INTB) specifies the address where the interrupt vector table starts.

Set the INTB to a multiple of 4 to reduce the number of cycles required to execute interrupt sequences and instructions entailing stack manipulation.

2.2.2.4 Program Counter (PC)



The program counter (PC) indicates the address of the instruction being executed.

2.2.2.5 Processor Status Word (PSW)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	—	IPL[3:0]				—	—	—	PM	—	—	U	I
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	—	—	—	—	—	O	S	Z	C
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	C	Carry Flag	0: No carry has occurred. 1: A carry has occurred.	R/W
b1	Z	Zero Flag	0: Result is non-zero. 1: Result is 0.	R/W
b2	S	Sign Flag	0: Result is a positive value or 0. 1: Result is a negative value.	R/W
b3	O	Overflow Flag	0: No overflow has occurred. 1: An overflow has occurred.	R/W
b15 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	I*1	Interrupt Enable	0: Interrupt disabled. 1: Interrupt enabled.	R/W
b17	U*1	Stack Pointer Select	0: Interrupt stack pointer (ISP) is selected. 1: User stack pointer (USP) is selected.	R/W
b19, b18	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b20	PM*1, *2, *3	Processor Mode Select	0: Supervisor mode is selected. 1: User mode is selected.	R/W
b23 to b21	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b27 to b24	IPL[3:0]*1	Processor Interrupt Priority Level	b27 b24 0 0 0 0: Priority level 0 (lowest) 0 0 0 1: Priority level 1 0 0 1 0: Priority level 2 0 0 1 1: Priority level 3 0 1 0 0: Priority level 4 0 1 0 1: Priority level 5 0 1 1 0: Priority level 6 0 1 1 1: Priority level 7 1 0 0 0: Priority level 8 1 0 0 1: Priority level 9 1 0 1 0: Priority level 10 1 0 1 1: Priority level 11 1 1 0 0: Priority level 12 1 1 0 1: Priority level 13 1 1 1 0: Priority level 14 1 1 1 1: Priority level 15 (highest)	R/W
b31 to b28	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. In user mode, writing to the IPL[3:0], PM, U, and I bits by an MVTC or a POPC instruction is ignored. Writing to the IPL[3:0] bits by an MVTIPL instruction generates a privileged instruction exception.

Note 2. In supervisor mode, writing to the PM bit by an MVTC or a POPC instruction is ignored, but writing to the other bits is possible.

Note 3. Switching from supervisor mode to user mode requires execution of an RTE instruction after having set the PSW.PM bit saved on the stack to 1 or executing an RTFI instruction after having set the BPSW.PM bit to 1.

The processor status word (PSW) indicates the results of instruction execution or the state of the CPU.

C Flag (Carry Flag)

This flag retains the state of the bit after a carry, borrow, or shift-out has occurred.

Z Flag (Zero Flag)

This flag is set to 1 if the result of an operation is 0; otherwise its value is cleared to 0.

S Flag (Sign Flag)

This flag is set to 1 if the result of an operation is negative; otherwise its value is cleared to 0.

O Flag (Overflow Flag)

This flag is set to 1 if the result of an operation overflows; otherwise its value is cleared to 0.

I Bit (Interrupt Enable)

This bit enables interrupt requests. When a WAIT instruction is executed, the value of this bit becomes 1. It becomes 0 when an exception is accepted.

U Bit (Stack Pointer Select)

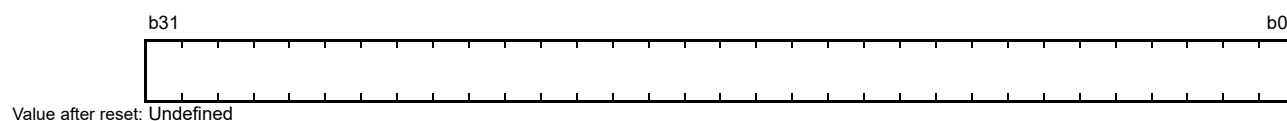
This bit specifies the stack pointer as either the ISP or USP. When an exception request is accepted, this bit is set to 0. When the processor mode is switched from supervisor mode to user mode, this bit is set to 1.

PM Bit (Processor Mode Select)

This bit specifies the processor mode. When an exception is accepted, the value of this bit becomes 0.

IPL[3:0] Bits (Processor Interrupt Priority Level)

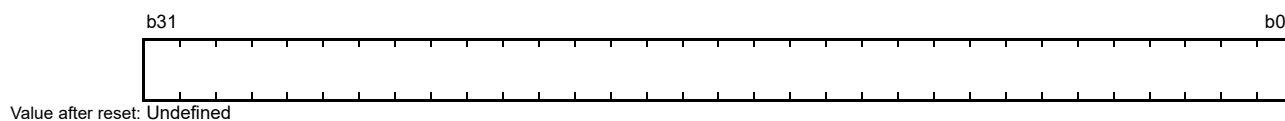
The IPL[3:0] bits specify the processor interrupt priority level as one of sixteen levels from zero to fifteen, wherein priority level zero is the lowest and priority level fifteen the highest. When the priority level of a requested interrupt is higher than the processor interrupt priority level, the interrupt is enabled. Setting the IPL[3:0] bits to level fifteen (Fh) disables all interrupt requests. The IPL[3:0] bits are set to level fifteen (Fh) when a non-maskable interrupt is generated. When interrupts in general are generated, the bits are set to the priority levels of accepted interrupts.

2.2.2.6 Backup PC (BPC)

The backup PC (BPC) is provided to speed up response to interrupts.

After a fast interrupt has been generated, the contents of the program counter (PC) are saved in the BPC register.

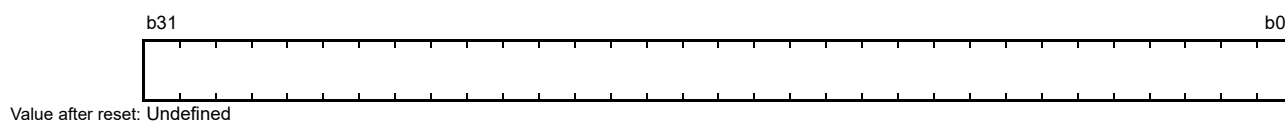
2.2.2.7 Backup PSW (BPSW)



The backup PSW (BPSW) is provided to speed up response to interrupts.

After a fast interrupt has been generated, the contents of the processor status word (PSW) are saved in the BPSW. The allocation of bits in the BPSW corresponds to that in the PSW.

2.2.2.8 Fast Interrupt Vector Register (FINTV)



The fast interrupt vector register (FINTV) is provided to speed up response to interrupts.

The FINTV register specifies a branch destination address when a fast interrupt has been generated.

2.2.2.9 Floating-Point Status Word (FPSW)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	FS	FX	FU	FZ	FO	FV	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	EX	EU	EZ	EO	EV	—	DN	CE	CX	CU	CZ	CO	CV	RM[1:0]	—
Value after reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	RM[1:0]	Floating-Point Rounding-Mode Setting	b1 b0 0 0: Rounding towards the nearest value 0 1: Rounding towards 0 1 0: Rounding towards $+\infty$ 1 1: Rounding towards $-\infty$	R/W
b2	CV	Invalid Operation Cause Flag	0: No invalid operation has been encountered. 1: Invalid operation has been encountered.	R/(W) *1
b3	CO	Overflow Cause Flag	0: No overflow has occurred. 1: Overflow has occurred.	R/(W) *1
b4	CZ	Division-by-Zero Cause Flag	0: No division-by-zero has occurred. 1: Division-by-zero has occurred.	R/(W) *1
b5	CU	Underflow Cause Flag	0: No underflow has occurred. 1: Underflow has occurred.	R/(W) *1
b6	CX	Inexact Cause Flag	0: No inexact exception has been generated. 1: Inexact exception has been generated.	R/(W) *1
b7	CE	Unimplemented Processing Cause Flag	0: No unimplemented processing has been encountered. 1: Unimplemented process has been encountered.	R/(W) *1
b8	DN	0 Flush Bit of Denormalized Number	0: A denormalized number is handled as a denormalized number. 1: A denormalized number is handled as 0.*2	R/W
b9	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b10	EV	Invalid Operation Exception Enable	0: Invalid operation exception is masked. 1: Invalid operation exception is enabled.	R/W
b11	EO	Overflow Exception Enable	0: Overflow exception is masked. 1: Overflow exception is enabled.	R/W
b12	EZ	Division-by-Zero Exception Enable	0: Division-by-zero exception is masked. 1: Division-by-zero exception is enabled.	R/W
b13	EU	Underflow Exception Enable	0: Underflow exception is masked. 1: Underflow exception is enabled.	R/W
b14	EX	Inexact Exception Enable	0: Inexact exception is masked. 1: Inexact exception is enabled.	R/W
b25 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b26	FV*3	Invalid Operation Flag	0: No invalid operation has been encountered. 1: Invalid operation has been encountered.*8	R/W
b27	FO*4	Overflow Flag	0: No overflow has occurred. 1: Overflow has occurred.*8	R/W
b28	FZ*5	Division-by-Zero Flag	0: No division-by-zero has occurred. 1: Division-by-zero has occurred.*8	R/W
b29	FU*6	Underflow Flag	0: No underflow has occurred. 1: Underflow has occurred.*8	R/W
b30	FX*7	Inexact Flag	0: No inexact exception has been generated. 1: Inexact exception has been generated.*8	R/W

Bit	Symbol	Bit Name	Description	R/W
b31	FS	Floating-Point Error Summary Flag	This bit reflects the logical OR of the FU, FZ, FO, and FV flags.	R

- Note 1. Writing 0 to the bit clears it. Writing 1 to the bit does not affect its value.
 Note 2. Positive denormalized numbers are treated as +0, negative denormalized numbers as -0.
 Note 3. When the EV bit is set to 0, the FV flag is enabled.
 Note 4. When the EO bit is set to 0, the FO flag is enabled.
 Note 5. When the EZ bit is set to 0, the FZ flag is enabled.
 Note 6. When the EU bit is set to 0, the FU flag is enabled.
 Note 7. When the EX bit is set to 0, the FX flag is enabled.
 Note 8. Once the bit has been set to 1, this value is retained until it is cleared to 0 by software.

The floating-point status word (FPSW) indicates the results of floating-point operations.

When an exception handling enable bit (Ej) enables the exception handling (Ej = 1), the exception cause can be identified by checking the corresponding Cj flag in the exception handling routine. If the exception handling is masked (Ej = 0), the occurrence of exception can be checked by reading the Fj flag at the end of a series of processing. Once the Fj flag has been set to 1, this value is retained until it is cleared to 0 by software (j = X, U, Z, O, or V).

RM[1:0] Bits (Floating-Point Rounding-Mode Setting)

These bits specify the floating-point rounding-mode.

Explanation of Floating-Point Rounding Modes

- Rounding towards the nearest value (the default behavior) : An inexact result is rounded to the available value that is closest to the result which would be obtained with an infinite number of digits. If two available values are equally close, rounding is to the even alternative.
- Rounding towards 0 : An inexact result is rounded to the smallest available absolute value, i.e. in the direction of zero (simple truncation).
- Rounding towards $+\infty$: An inexact result is rounded to the nearest available value in the direction of positive infinity.
- Rounding towards $-\infty$: An inexact result is rounded to the nearest available value in the direction of negative infinity.

- (1) Rounding to the nearest value is specified as the default mode and returns the most accurate value.
- (2) Modes such as rounding towards 0, rounding towards $+\infty$, and rounding towards $-\infty$ are used to ensure precision when interval arithmetic is employed.

CV Flag (Invalid Operation Cause Flag), CO Flag (Overflow Cause Flag), CZ Flag (Division-by-Zero Cause Flag), CU Flag (Underflow Cause Flag), CX Flag (Inexact Cause Flag), and CE Flag (Unimplemented Processing Cause Flag)

Floating-point exceptions include the five specified in the IEEE754 standard, namely overflow, underflow, inexact, division-by-zero, and invalid operation. For a further floating-point exception that is generated upon detection of unimplemented processing, the corresponding flag (CE) is set to 1.

- The bit that has been set to 1 is cleared to 0 when the FPU instruction is executed.
- When 0 is written to the bit by the MVTC and POPC instructions, the bit is set to 0; the bit retains the previous value when 1 is written by the instruction.

DN Flag (0 Flush Bit of Denormalized Number)

When this bit is set to 0, a denormalized number is handled as a denormalized number. When this bit is set to 1, a denormalized number is handled as 0.

EV Bit (Invalid Operation Exception Enable), EO Bit (Overflow Exception Enable), EZ Bit (Division-by-Zero Exception Enable), EU Bit (Underflow Exception Enable), and EX Bit (Inexact Exception Enable)

When any of five floating-point exceptions specified in the IEEE754 standard is generated by the floating-point

operation instruction, the bit decides whether the CPU will start handling the exception. When the bit is set to 0, the exception handling is masked; when the bit is set to 1, the exception handling is enabled.

FV Flag (Invalid Operation Flag), FO Flag (Overflow Flag), FZ Flag (Division-by-Zero Flag), FU Flag (Underflow Flag), and FX Flag (Inexact Flag)

While the exception handling enable bit (Ej) is 0 (exception handling is masked), if any of five floating-point exceptions specified in the IEEE754 standard is generated, the corresponding bit is set to 1.

- When Ej is 1 (exception handling is enabled), the value of the flag remains.
- When the corresponding flag is set to 1, it remains 1 until it is cleared to 0 by software. (accumulation flag)

FS Flag (Floating-Point Error Summary Flag)

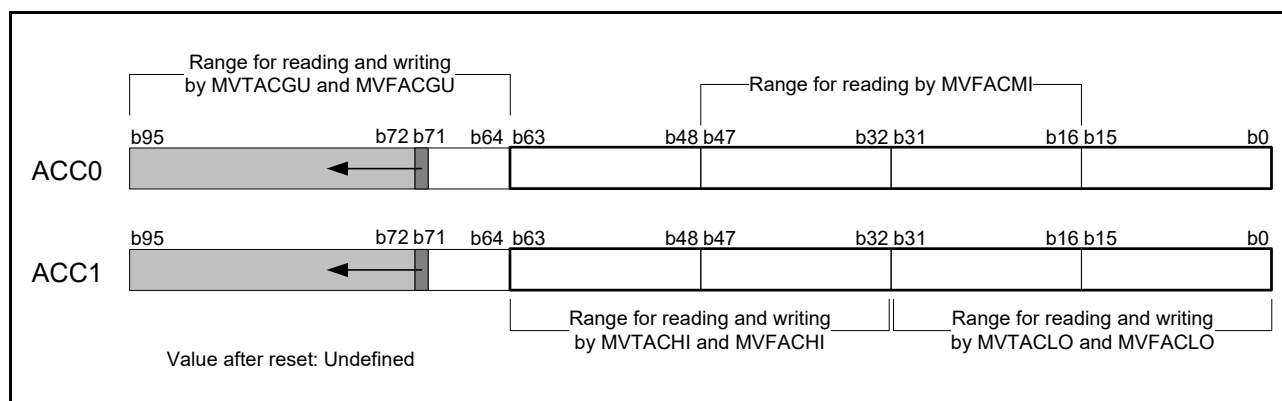
This bit reflects the logical OR of the FU, FZ, FO, and FV flags.

2.2.3 Accumulator

The accumulator (ACC0 or ACC1) is a 72-bit register used for DSP instructions. The accumulator is handled as a 96-bit register for reading and writing. At this time, when bits 95 to 72 of the accumulator are read, the value where the value of bit 71 is sign extended is read. Writing to bits 95 to 72 of the accumulator is ignored. ACC0 is also used for the multiply and multiply-and-accumulate instructions; EMUL, EMULU, FMUL, MUL, and RMPA, in which case the prior value in ACC0 is modified by execution of the instruction.

Use the MVTACGU, MVTACHI, and MVTACLO instructions for writing to the accumulator. The MVTACGU, MVTACHI, and MVTACLO instructions write data to bits 95 to 64, the higher-order 32 bits (bits 63 to 32), and the lower-order 32 bits (bits 31 to 0), respectively.

Use the MVFACGU, MVFACHI, MVFACMI, and MVFACLO instructions for reading data from the accumulator. The MVFACGU, MVFACHI, MVFACMI, and MVFACLO instructions read data from the guard bits (bits 95 to 64), higher-order 32 bits (bits 63 to 32), the middle 32 bits (bits 31 to 0), and the lower-order 32 bits (bits 31 to 0), respectively.



Note: The value of bit 71 is sign extended for bits 95 to 72 and the extended value is always read. Writing to this area is ignored.

2.3 Processor Mode

The RXv2 CPU supports two processor modes, supervisor and user. These processor modes enable the realization of a hierarchical CPU resource protection and memory protection mechanism. Each processor mode imposes a level on rights of access to memory and the instructions that can be executed. Supervisor mode carries greater rights than user mode. The initial state after a reset is supervisor mode.

2.3.1 Supervisor Mode

In supervisor mode, all CPU resources are accessible and all instructions are available. However, writing to the processor mode select bit (PM) in the processor status word (PSW) by executing an MVTC or a POPC instruction will be ignored. For details on how to write to the PM bit, refer to section 2.2.2.5, Processor Status Word (PSW).

2.3.2 User Mode

In user mode, write access to the CPU resources listed below is restricted. The restriction applies to any instruction capable of write access.

- Some bits (bits IPL[3:0], PM, U, and I) in the processor status word (PSW)
- Interrupt stack pointer (ISP)
- Exception table register (EXTB)
- Interrupt table register (INTB)
- Backup PSW (BPSW)
- Backup PC (BPC)
- Fast interrupt vector register (FINTV)

2.3.3 Privileged Instruction

Privileged instructions can only be executed in supervisor mode. Executing a privileged instruction in user mode produces a privileged instruction exception. Privileged instructions include the RTFI, MVTIPL, RTE, and WAIT instructions.

2.3.4 Switching Between Processor Modes

Manipulating the processor mode select bit (PM) in the processor status word (PSW) switches the processor mode. However, rewriting to the PM bit by executing an MVTC or a POPC instruction is prohibited. Switch the processor mode by following the procedures described below.

(1) Switching from user mode to supervisor mode

After an exception has been generated, the PSW.PM bit is set to 0 and the CPU switches to supervisor mode. The hardware pre-processing is executed in supervisor mode. The state of the processor mode before the exception was generated is retained in the copy of PSW.PM bit is saved on the stack.

(2) Switching from supervisor mode to user mode

Executing an RTE instruction when the value of the copy of the PSW.PM bit that has been preserved on the stack is 1 or an RTFI instruction when the value of the copy of the PSW.PM bit that has been preserved in the backup PSW (BPSW) is 1 causes a transition to user mode. In the transition to user mode, the value of the stack pointer designation bit (the U bit in the PSW) becomes 1.

2.4 Data Types

The RXv2 CPU can handle four types of data: integer, floating-point, bit, and string.

For details, refer to RX Family RXv2 Instruction Set Architecture User's Manual: Software.

2.4.1 Integer

An integer can be signed or unsigned. For signed integers, negative values are represented by two's complements.

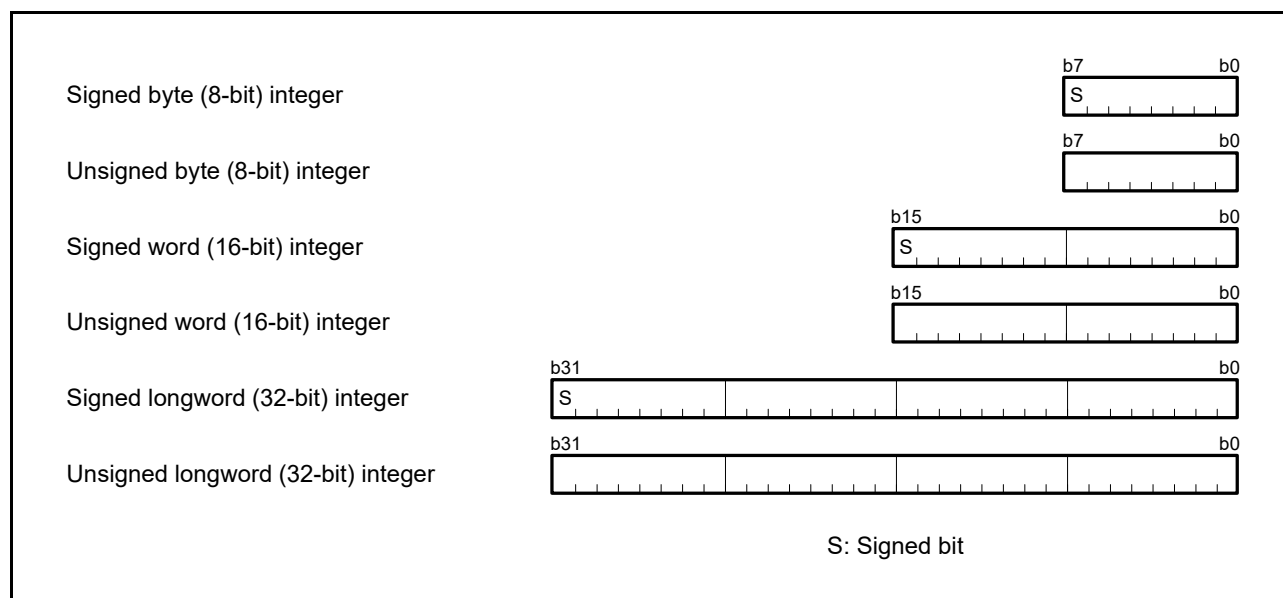


Figure 2.2 Integer

2.4.2 Floating-Points

Floating-point support is for the single-precision floating-point type specified in the IEEE754 standard; operands of this type can be used in eleven floating-point operation instructions: FADD, FCMP, FDIV, FMUL, FSQRT, FSUB, FTOI, FTOU, ITOF, ROUND, and UTOF.

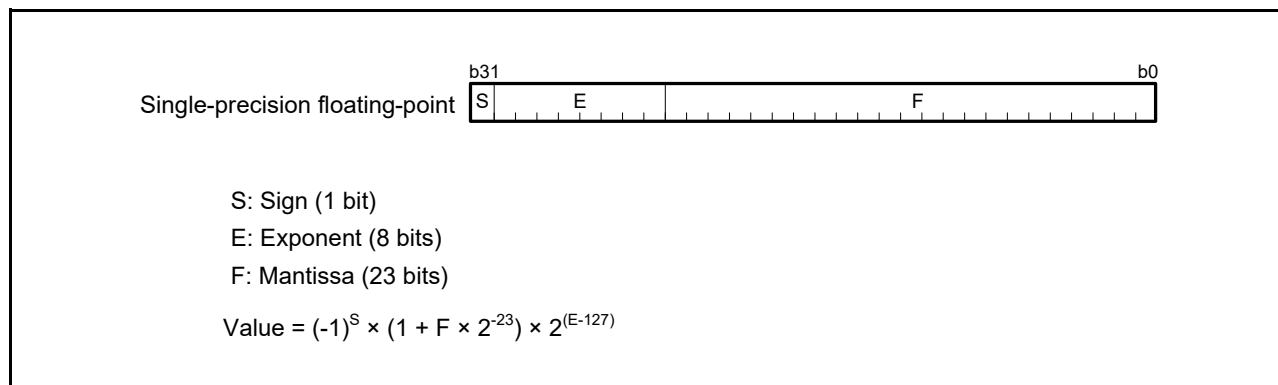


Figure 2.3 Floating-Point

The floating-point format supports the values listed below.

- 0 < E < 255 (normal numbers)
- E = 0 and F = 0 (signed zero)
- E = 0 and F > 0 (denormalized numbers)*1
- E = 255 and F = 0 (infinity)
- E = 255 and F > 0 (NaN: Not-a-Number)

Note 1. The number is treated as 0 when the FPSW.DN bit is 1. When the DN bit is 0, an unimplemented processing exception is generated.

2.4.3 Bitwise Operations

Five bit-manipulation instructions are provided for bitwise operations: BCLR, BMCnd, BNOT, BSET, and BTST.

A bit in a register is specified as the destination register and a bit number in the range from 31 to 0.

A bit in memory is specified as the destination address and a bit number from 7 to 0. The addressing modes available to specify addresses are register indirect and register relative.

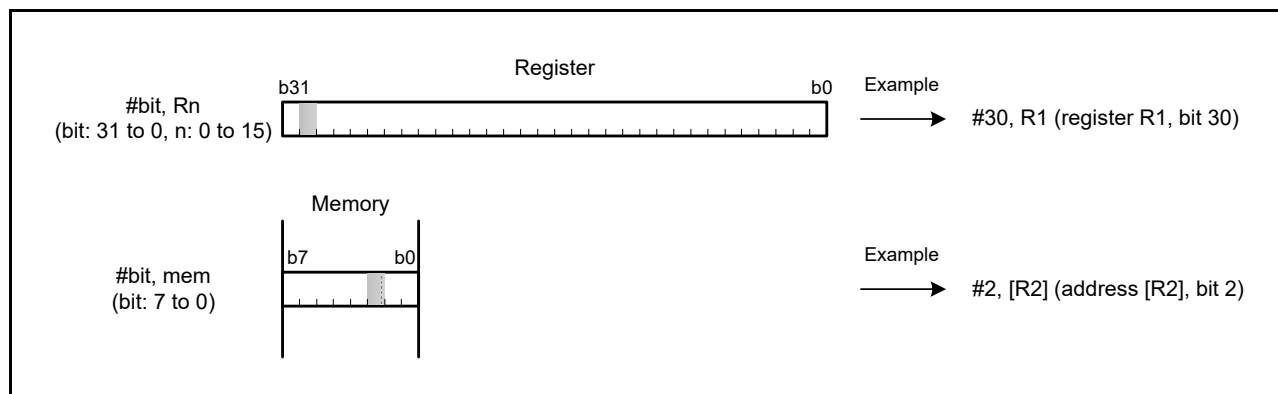


Figure 2.4 Bit

2.4.4 Strings

The string data type consists of an arbitrary number of consecutive byte (8-bit), word (16-bit), or longword (32-bit) units. Seven string manipulation instructions are provided for use with strings: SCMPU, SMOVB, SMOVF, SMOVU, SSTR, SUNTIL, and SWHILE.

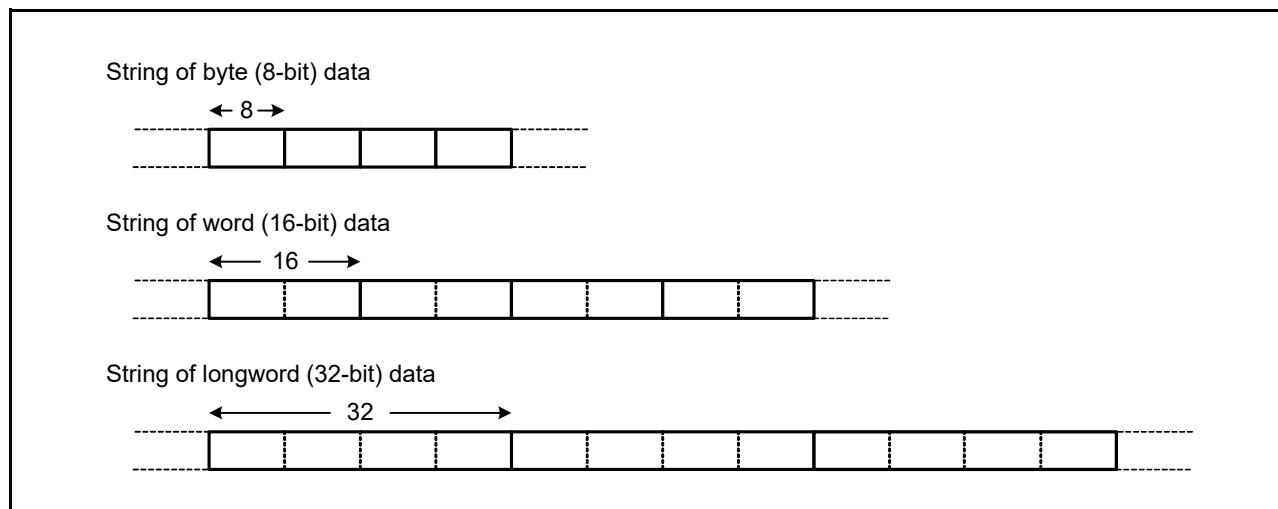


Figure 2.5 String

2.5 Endian

For the RXv2 CPU, instructions are little endian, but the treatment of data is selectable as little or big endian.

2.5.1 Switching the Endian

As arrangements of bytes, this MCU supports both big endian, where the higher-order byte (MSB) is at location 0, and little endian, where the lower-order byte (LSB) is at location 0.

For details on the endian setting, see [section 3, Operating Modes](#).

Operations for access differ according to the endian setting and, depending on the instruction, whether 8-, 16- or 32-bit access has been selected. Operations for access in the various possible cases are described in [Table 2.1](#) to [Table 2.12](#).

In the tables,

- LL indicates bits D7 to D0 of the general-purpose register,
- LH indicates bits D15 to D8 of the general-purpose register,
- HL indicates bits D23 to D16 of the general-purpose register, and
- HH indicates bits D31 to D24 of the general-purpose register.

	D31 to D24	D23 to D16	D15 to D8	D7 to D0
General purpose register: Rm	HH	HL	LH	LL

Table 2.1 32-Bit Read Operations when Little Endian has been Selected

Operation Address of src	Reading a 32-bit unit from address 0	Reading a 32-bit unit from address 1	Reading a 32-bit unit from address 2	Reading a 32-bit unit from address 3	Reading a 32-bit unit from address 4
Address 0	Transfer to LL	—	—	—	—
Address 1	Transfer to LH	Transfer to LL	—	—	—
Address 2	Transfer to HL	Transfer to LH	Transfer to LL	—	—
Address 3	Transfer to HH	Transfer to HL	Transfer to LH	Transfer to LL	—
Address 4	—	Transfer to HH	Transfer to HL	Transfer to LH	Transfer to LL
Address 5	—	—	Transfer to HH	Transfer to HL	Transfer to LH
Address 6	—	—	—	Transfer to HH	Transfer to HL
Address 7	—	—	—	—	Transfer to HH

Table 2.2 32-Bit Read Operations when Big Endian has been Selected

Operation Address of src	Reading a 32-bit unit from address 0	Reading a 32-bit unit from address 1	Reading a 32-bit unit from address 2	Reading a 32-bit unit from address 3	Reading a 32-bit unit from address 4
Address 0	Transfer to HH	—	—	—	—
Address 1	Transfer to HL	Transfer to HH	—	—	—
Address 2	Transfer to LH	Transfer to HL	Transfer to HH	—	—
Address 3	Transfer to LL	Transfer to LH	Transfer to HL	Transfer to HH	—
Address 4	—	Transfer to LL	Transfer to LH	Transfer to HL	Transfer to HH
Address 5	—	—	Transfer to LL	Transfer to LH	Transfer to HL
Address 6	—	—	—	Transfer to LL	Transfer to LH
Address 7	—	—	—	—	Transfer to LL

Table 2.3 32-Bit Write Operations when Little Endian has been Selected

Operation Address of dest	Writing a 32-bit unit to address 0	Writing a 32-bit unit to address 1	Writing a 32-bit unit to address 2	Writing a 32-bit unit to address 3	Writing a 32-bit unit to address 4
Address 0	Transfer from LL	—	—	—	—
Address 1	Transfer from LH	Transfer from LL	—	—	—
Address 2	Transfer from HL	Transfer from LH	Transfer from LL	—	—
Address 3	Transfer from HH	Transfer from HL	Transfer from LH	Transfer from LL	—
Address 4	—	Transfer from HH	Transfer from HL	Transfer from LH	Transfer from LL
Address 5	—	—	Transfer from HH	Transfer from HL	Transfer from LH
Address 6	—	—	—	Transfer from HH	Transfer from HL
Address 7	—	—	—	—	Transfer from HH

Table 2.4 32-Bit Write Operations when Big Endian has been Selected

Operation Address of dest	Writing a 32-bit unit to address 0	Writing a 32-bit unit to address 1	Writing a 32-bit unit to address 2	Writing a 32-bit unit to address 3	Writing a 32-bit unit to address 4
Address 0	Transfer from HH	—	—	—	—
Address 1	Transfer from HL	Transfer from HH	—	—	—
Address 2	Transfer from LH	Transfer from HL	Transfer from HH	—	—
Address 3	Transfer from LL	Transfer from LH	Transfer from HL	Transfer from HH	—
Address 4	—	Transfer from LL	Transfer from LH	Transfer from HL	Transfer from HH
Address 5	—	—	Transfer from LL	Transfer from LH	Transfer from HL
Address 6	—	—	—	Transfer from LL	Transfer from LH
Address 7	—	—	—	—	Transfer from LL

Table 2.5 16-Bit Read Operations when Little Endian has been Selected

Operation Address of src	Reading a 16-bit unit from address 0	Reading a 16-bit unit from address 1	Reading a 16-bit unit from address 2	Reading a 16-bit unit from address 3	Reading a 16-bit unit from address 4	Reading a 16-bit unit from address 5	Reading a 16-bit unit from address 6
Address 0	Transfer to LL	—	—	—	—	—	—
Address 1	Transfer to LH	Transfer to LL	—	—	—	—	—
Address 2	—	Transfer to LH	Transfer to LL	—	—	—	—
Address 3	—	—	Transfer to LH	Transfer to LL	—	—	—
Address 4	—	—	—	Transfer to LH	Transfer to LL	—	—
Address 5	—	—	—	—	Transfer to LH	Transfer to LL	—
Address 6	—	—	—	—	—	Transfer to LH	Transfer to LL
Address 7	—	—	—	—	—	—	Transfer to LH

Table 2.6 16-Bit Read Operations when Big Endian has been Selected

Operation Address of src	Reading a 16-bit unit from address 0	Reading a 16-bit unit from address 1	Reading a 16-bit unit from address 2	Reading a 16-bit unit from address 3	Reading a 16-bit unit from address 4	Reading a 16-bit unit from address 5	Reading a 16-bit unit from address 6
Address 0	Transfer to LH	—	—	—	—	—	—
Address 1	Transfer to LL	Transfer to LH	—	—	—	—	—
Address 2	—	Transfer to LL	Transfer to LH	—	—	—	—
Address 3	—	—	Transfer to LL	Transfer to LH	—	—	—
Address 4	—	—	—	Transfer to LL	Transfer to LH	—	—
Address 5	—	—	—	—	Transfer to LL	Transfer to LH	—
Address 6	—	—	—	—	—	Transfer to LL	Transfer to LH
Address 7	—	—	—	—	—	—	Transfer to LL

Table 2.7 16-Bit Write Operations when Little Endian has been Selected

Operation Address of dest	Writing a 16-bit unit to address 0	Writing a 16-bit unit to address 1	Writing a 16-bit unit to address 2	Writing a 16-bit unit to address 3	Writing a 16-bit unit to address 4	Writing a 16-bit unit to address 5	Writing a 16-bit unit to address 6
Address 0	Transfer from LL	—	—	—	—	—	—
Address 1	Transfer from LH	Transfer from LL	—	—	—	—	—
Address 2	—	Transfer from LH	Transfer from LL	—	—	—	—
Address 3	—	—	Transfer from LH	Transfer from LL	—	—	—
Address 4	—	—	—	Transfer from LH	Transfer from LL	—	—
Address 5	—	—	—	—	Transfer from LH	Transfer from LL	—
Address 6	—	—	—	—	—	Transfer from LH	Transfer from LL
Address 7	—	—	—	—	—	—	Transfer from LH

Table 2.8 16-Bit Write Operations when Big Endian has been Selected

Operation Address of dest	Writing a 16-bit unit to address 0	Writing a 16-bit unit to address 1	Writing a 16-bit unit to address 2	Writing a 16-bit unit to address 3	Writing a 16-bit unit to address 4	Writing a 16-bit unit to address 5	Writing a 16-bit unit to address 6
Address 0	Transfer from LH	—	—	—	—	—	—
Address 1	Transfer from LL	Transfer from LH	—	—	—	—	—
Address 2	—	Transfer from LL	Transfer from LH	—	—	—	—
Address 3	—	—	Transfer from LL	Transfer from LH	—	—	—
Address 4	—	—	—	Transfer from LL	Transfer from LH	—	—
Address 5	—	—	—	—	Transfer from LL	Transfer from LH	—
Address 6	—	—	—	—	—	Transfer from LL	Transfer from LH
Address 7	—	—	—	—	—	—	Transfer from LL

Table 2.9 8-Bit Read Operations when Little Endian has been Selected

Operation Address of src	Reading an 8-bit unit from address 0	Reading an 8-bit unit from address 1	Reading an 8-bit unit from address 2	Reading an 8-bit unit from address 3
Address 0	Transfer to LL	—	—	—
Address 1	—	Transfer to LL	—	—
Address 2	—	—	Transfer to LL	—
Address 3	—	—	—	Transfer to LL

Table 2.10 8-Bit Read Operations when Big Endian has been Selected

Operation Address of src	Reading an 8-bit unit from address 0	Reading an 8-bit unit from address 1	Reading an 8-bit unit from address 2	Reading an 8-bit unit from address 3
Address 0	Transfer to LL	—	—	—
Address 1	—	Transfer to LL	—	—
Address 2	—	—	Transfer to LL	—
Address 3	—	—	—	Transfer to LL

Table 2.11 8-Bit Write Operations when Little Endian has been Selected

Operation Address of dest	Writing an 8-bit unit to address 0	Writing an 8-bit unit to address 1	Writing an 8-bit unit to address 2	Writing an 8-bit unit to address 3
Address 0	Transfer from LL	—	—	—
Address 1	—	Transfer from LL	—	—
Address 2	—	—	Transfer from LL	—
Address 3	—	—	—	Transfer from LL

Table 2.12 8-Bit Write Operations when Big Endian has been Selected

Operation Address of dest	Writing an 8-bit unit to address 0	Writing an 8-bit unit to address 1	Writing an 8-bit unit to address 2	Writing an 8-bit unit to address 3
Address 0	Transfer from LL	—	—	—
Address 1	—	Transfer from LL	—	—
Address 2	—	—	Transfer from LL	—
Address 3	—	—	—	Transfer from LL

2.5.2 Access to I/O Registers

The addresses of I/O registers are fixed, and this is regardless of whether the setting is for little endian or big endian. Accordingly, changes to the endian do not affect access to I/O registers. For the arrangements of I/O registers, refer to the descriptions of registers in the relevant sections.

2.5.3 Notes on Access to I/O Registers

Ensure that access to I/O registers is in accord with the following rules.

- With I/O registers for which a bus width of eight bits is indicated, use instructions having operands of the same width (eight bits). That is, access these registers by using instructions with .B as the size specifier (.size), or with .B or .UB as the size-extension specifier (.memex).
- With I/O registers for which a bus width of 16 bits is indicated, use instructions having operands of the same width (16 bits). That is, access these registers by using instructions with .W as the size specifier (.size), or with .W or .UW as the size-extension specifier (.memex).
- With I/O registers for which a bus width of 32 bits is indicated, use instructions having operands of the same width (32 bits). That is, access these registers by using instructions with .L as the size specifier (.size), or with .L size-extension specifier (.memex).

2.5.4 Data Arrangement

2.5.4.1 Data Arrangement in Registers

Figure 2.6 shows the relation between the sizes of registers and bit numbers.

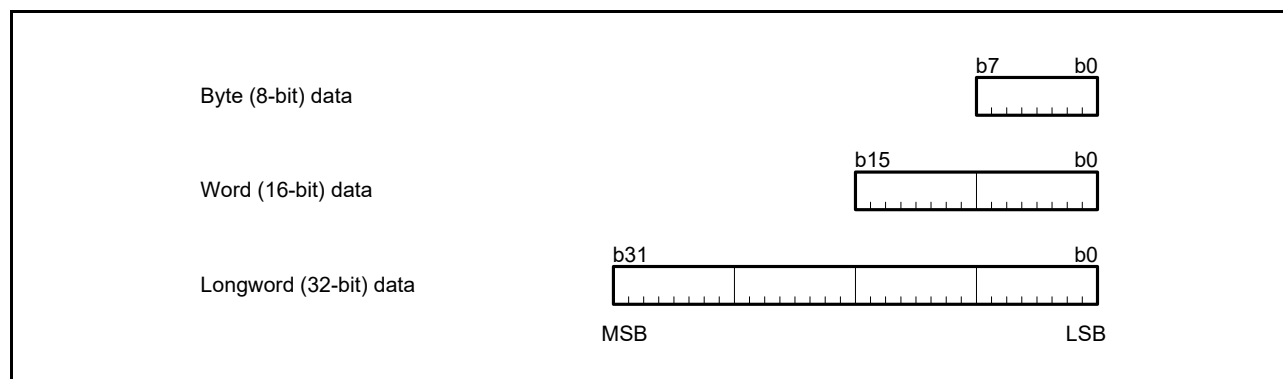


Figure 2.6 Data Arrangement in Registers

2.5.4.2 Data Arrangement in Memory

Data in memory have three sizes: byte (8-bit), word (16-bit), and longword (32-bit). The data arrangement is selectable as little endian or big endian. Figure 2.7 shows the arrangement of data in memory.

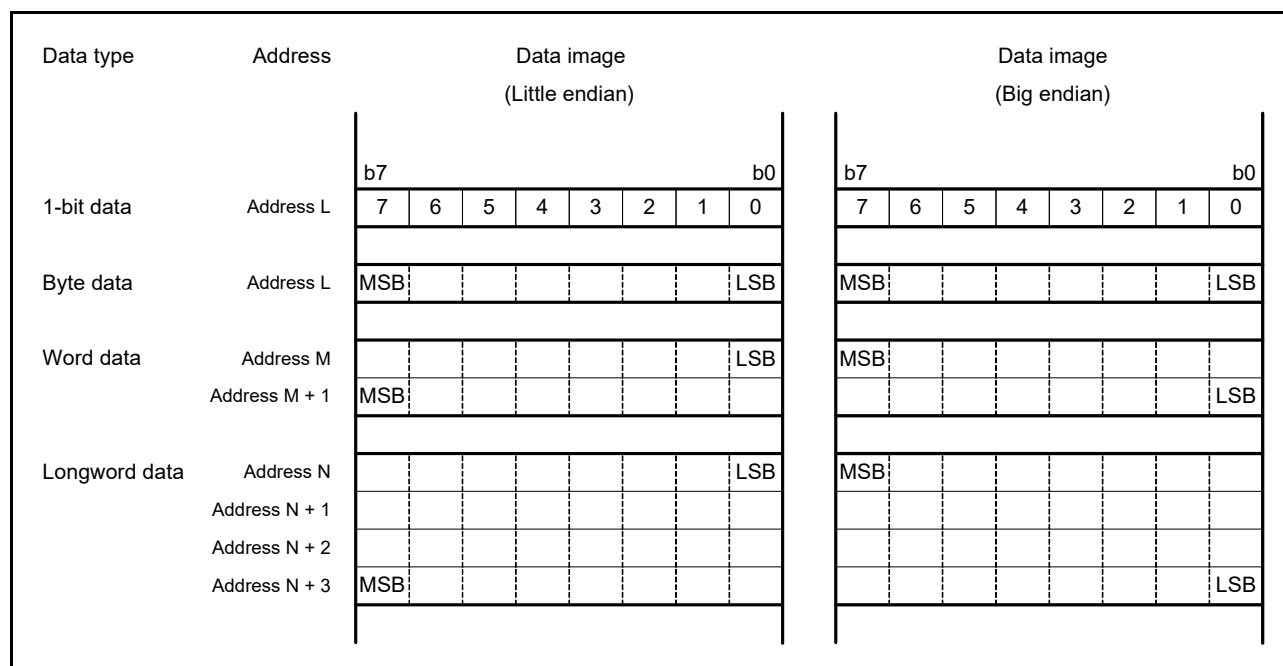


Figure 2.7 Data Arrangement in Memory

2.6 Vector Table

There are two types of vector table: exception and interrupt. Each vector in the vector table consists of four bytes and specifies the address where the corresponding exception handling routine starts.

2.6.1 Exception Vector Table

In the exception vector table, the individual vectors for the privileged instruction exception, undefined instruction exception, floating-point exception, and non-maskable interrupt are allocated to the 124-byte area where the value indicated by the exception table register (EXTB) is used as the starting address (ExtBase). The reset vector is always allocated to FFFFFFFCh, regardless of the value of the exception vector table.

Figure 2.8 shows the exception vector table.

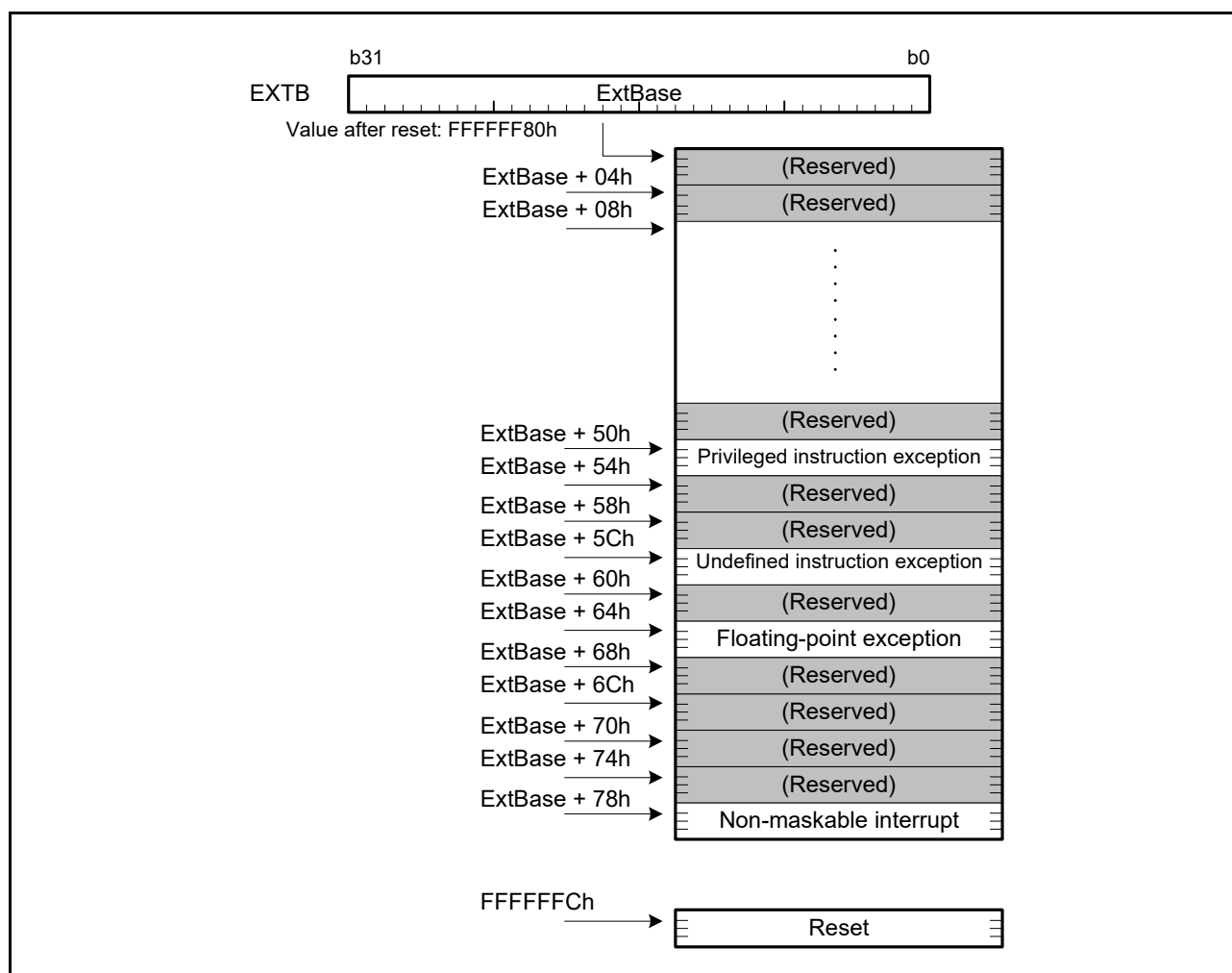


Figure 2.8 Exception Vector Table

2.6.2 Interrupt Vector Table

The address where the interrupt vector table is placed can be adjusted. The table is a 1,024-byte region that contains all vectors for unconditional traps and interrupts and starts at the address (IntBase) specified in the interrupt table register (INTB). Figure 2.9 shows the interrupt vector table.

Each vector in the interrupt vector table has a vector number from 0 to 255. Each of the INT instructions, which act as the sources of unconditional traps, is allocated to the vector that has the same number as is specified as the operand of the instruction itself (from 0 to 255). The BRK instruction is allocated to the vector with number 0. Furthermore, vector numbers (from 0 to 255) are allocated to interrupt requests in a fixed way for each product. For more on interrupt vector numbers, see section 14.3.1, Interrupt Vector Table.

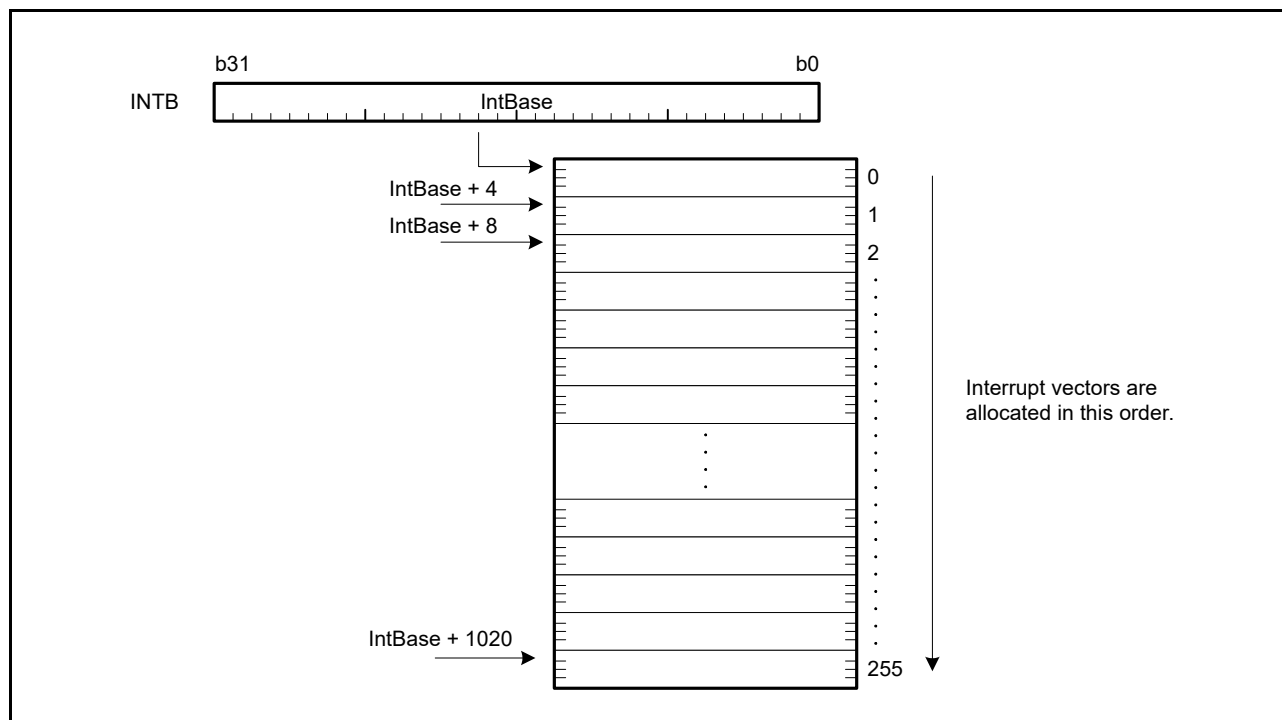


Figure 2.9 Interrupt Vector Table

2.7 Operation of Instructions

2.7.1 Restrictions on RMPA and String-Manipulation Instructions

2.7.1.1 Transfer Size and Data Prefetching

The RMPA instruction and the string-manipulation instructions (SCMPU, SMOVB, SMOVF, SMOVU, SSTR, SUNTIL, and SWHILE instructions) transfer data in longword units to speed up the reading of data from and writing of data to the memory. If the last of the data to be processed is less than a longword, data transfer proceeds with the sizes described below:

- RMPA, SSTR, SUNTIL, and SWHILE instructions: Size specified by the size specifier
- SCMPU, SMOVB, SMOVF, and SMOVU instructions: Byte

Additionally, in the above processing, the RMPA instruction and the string-manipulation instructions other than the SSTR instruction (that is, the SCMPU, SMOVB, SMOVF, SMOVU, SUNTIL, and SWHILE instructions) prefetch data when reading data from the memory. Data is prefetched from the prefetching start position with three bytes as the upper limit. The prefetching start positions of each operation are shown below.

- RMPA instruction: The multiplicand address specified by R1, and the multiplier address specified by R2
- SCMPU instruction: The source address specified by R1 for comparison, and the destination address specified by R2 for comparison
- SUNTIL and SWHILE instructions: The destination address specified by R1 for comparison
- SMOVB, SMOVF, and SMOVU instructions: The source address specified by R2 for transfer

2.7.1.2 Access to I/O Registers

The allocation of data to be handled by RMPA or string-manipulation instructions (SCMPU, SMOVB, SMOVF, SMOVU, SSTR, SUNTIL, and SWHILE instructions) to I/O registers is prohibited, and operation is not guaranteed if this restriction is not observed.

2.8 Number of Cycles

2.8.1 Instruction and Number of Cycle

Table 2.13 to Table 2.20 show the number of cycles in operation of each instruction. The listed numbers of cycles for access to memory are the numbers of cycles during no-wait access. The operands in the table below indicate the following meanings.

#IMM: Immediate

flag: bit, flag

Rs, Rs2, Rd, Rd2, Ri, Rb: General-purpose register

As, Ad: Accumulator

CR: Control register

dsp: displacement

pcdsp: displacement

Table 2.13 Number of Cycles for Arithmetic/logic Instructions

Instruction	Mnemonic (indicates the common operation when the size is omitted)	Number of Cycles
Arithmetic/logic instructions (register-register, immediate- register)	• {ABS, NEG, NOT} "Rd"/"Rs, Rd" • {ADC, MAX, MIN, ROTL, ROTR, XOR} "#IMM, Rd"/"Rs, Rd" • ADD "#IMM, Rd"/"Rs, Rd"/"#IMM, Rs, Rd"/"Rs, Rs2, Rd" • {AND, MUL, OR, SUB} "#IMM, Rd"/"Rs, Rd"/"Rs, Rs2, Rd" • {CMP, TST} "#IMM, Rs"/"Rs, Rs2" • NOP • {ROL, ROR, SAT} "Rd" • SBB "Rs, Rd" • {SHAR, SHLL, SHLR} "#IMM, Rd"/"Rs, Rd"/"#IMM, Rs, Rd"	1
	• DIV "#IMM, Rd"/"Rs, Rd"	3 to 20*1
	• DIVU "#IMM, Rd"/"Rs, Rd"	2 to 18*1
	• {EMUL, EMULU} "#IMM, Rd"/"Rs, Rd"	2
	• SATR	3
	Arithmetic/logic instructions (memory source operand)	3
	• {ADC, ADD, AND, MAX, MIN, MUL, OR, SBB, SUB, XOR} "[Rs], Rd"/"dsp[Rs], Rd" • {CMP, TST} "[Rs], Rs2"/"dsp[Rs], Rs2"	
	• DIV "[Rs], Rd / dsp[Rs], Rd"	5 to 22
	• DIVU "[Rs], Rd / dsp[Rs], Rd"	4 to 20
	• {EMUL, EMULU} "[Rs], Rd"/"dsp[Rs], Rd"	4
	• RMPA.B	6+7×floor(n/4)+4×(n%4) n: Number of processing bytes*2
	• RMPA.W	6+5×floor(n/2)+4×(n%2) n: Number of processing words*2
	• RMPA.L	6+4n n: Number of processing longwords*2

Note 1. The number of cycles for the dividing instruction varies according to the divisor and dividend.

Note 2. floor (x): Max. integer that is smaller than x.

Table 2.14 Number of Cycles for Transfer Instructions

Instruction	Mnemonic (indicates the common operation when the size is omitted)	Number of Cycles
Transfer instructions (register-register, immediate-register)	• MOV "#IMM, Rd"/"Rs, Rd" • {MOVU, REVL, REVW} "Rs, Rd" • SCCnd "Rd" • {STNZ, STZ} "#IMM, Rd"/"Rs, Rd"	1
	• XCHG "Rs, Rd"	2
Transfer instructions (load operation)	• {MOV, MOVU} "[Rs], Rd"/"dsp[Rs], Rd"/"[Rs+], Rd"/"[-Rs], Rd"/"[Ri, Rb], Rd" • MOVLi "[Rs], Rd" • POP "Rd"	Throughput: 1 Latency: 2* ¹
	• POPC "CR"	Throughput: 3 Latency: 4* ¹
	• POPM "Rd-Rd2"	Throughput: n Latency: n+1 n: Number of registers* ¹ , * ²
Transfer instructions (store operation)	• MOV "Rs, [Rd]"/"Rs, dsp[Rd]"/"Rs, [Rd+]/"Rs, [-Rd]"/"Rs, [Ri, Rb]"/"IMM, dsp[Rd]"/"IMM, [Rd]" • PUSH "Rs" • PUSHC "CR" • SCCnd "[Rd]"/"dsp[Rd]" • MOVCO "Rs, [Rd]"	1
	• PUSHM "Rs-Rs2"	n n: Number of registers* ³
Transfer instructions (memory-register)	• XCHG "[Rs], Rd"/"dsp[Rs], Rd"	2
Transfer instructions (memory-memory)	• MOV "[Rs], [Rd]"/"dsp[Rs], [Rd]"/"[Rs], dsp[Rd]"/"dsp[Rs], dsp[Rd]" • PUSH "[Rs]"/"dsp[Rs]"	3

Note 1. When the load data is used by the subsequent instruction, the number of cycles described as "latency" is counted as the number of cycles for the memory load instruction. For the cycles other than the memory load instruction, the number of cycles described as "throughput" is counted.

Note 2. The POPM instruction is converted into multiple load operations. The processing is the same as the one for the load operations of the MOV instruction, where the operation is repeated for the number of specified registers.

Note 3. The PUSHM instruction is converted into multiple store operations. The processing is the same as the one for the store operations of the MOV instruction, where the operation is repeated for the number of specified registers.

Table 2.15 Number of Cycles for Bit Manipulation Instructions

Instruction	Mnemonic (indicates the common operation when the size is omitted)	Number of Cycles
Bit manipulation instructions (register)	• {BCLR, BNOT, BSET} "#IMM, Rd"/"Rs, Rd" • BMCnd "#IMM, Rd" • BTST "#IMM, Rs"/"Rs, Rs2"	1
Bit manipulation instructions (memory source operand)	• {BCLR, BNOT, BSET} "#IMM, [Rd]"/"IMM, dsp[Rd]"/"Rs, [Rd]"/"Rs, dsp[Rd]" • BMCnd "#IMM, [Rd]"/"IMM, dsp[Rd]" • BTST "#IMM, [Rs]"/"IMM, dsp[Rs]"/"Rs, [Rs2]"/"Rs, dsp[Rs2]"	3

Table 2.16 Number of Cycles for Branch Instructions

Instruction	Mnemonic (indicates the common operation when the size is omitted)	Number of Cycles
Branch instructions	• BCnd "pcdsp" • {BRA, BSR} "pcdsp"/"Rs" • {JMP, JSR} "Rs"	Branch taken: 3 Branch not taken: 1
	• RTE	6
	• RTFI	3
	• RTS	5
	• RTSD "#IMM"	5
	• RTSD "#IMM, Rd-Rd2"	Throughput: $n < 5?5:1+n$ Latency: $n < 4?5:2+n$ n: Number of registers*1

?: Conditional operator

Note 1. When the load data is used by the subsequent instruction, the number of cycles described as "latency" is counted as the number of cycles for the memory load instruction. For the cycles other than the memory load instruction, the number of cycles described as "throughput" is counted.

Table 2.17 Number of Cycles for Floating-Point Operation Instructions

Instruction	Mnemonic (indicates the common operation when the size is omitted)	Number of Cycles
Floating-point operation instructions (register-register, immediate-register)	• {FADD, FSUB} "#IMM, Rd"/"Rs, Rd"/"Rs, Rs2, Rd"	2
	• FCMP "#IMM, Rs"/"Rs, Rs2"	1
	• FDIV "#IMM, Rd"/"Rs, Rd"	16
	• FMUL "#IMM, Rd"/"Rs, Rd"/"Rs, Rs2, Rd"	2
	• FSQRT "Rs, Rd"	16
	• {FTOI, ROUND, ITOF} "Rs, Rd"	2
	• {FTOU, UTOF} "Rs, Rd"	2
Floating-point operation instructions (memory source operand)	• {FADD, FSUB} "[Rs], Rd"/"dsp[Rs], Rd"	4
	• FCMP "[Rs], Rs2"/"dsp[Rs], Rs2"	3
	• FDIV "[Rs], Rd"/"dsp[Rs], Rd"	18
	• FMUL "[Rs], Rd"/"dsp[Rs], Rd"	4
	• FSQRT "[Rs], Rd"/"dsp[Rs], Rd"	18
	• {FTOI, ROUND, ITOF} "[Rs], Rd"/"dsp[Rs], Rd"	4
	• {FTOU, UTOF} "[Rs], Rd"/"dsp[Rs], Rd"	4

Table 2.18 Number of Cycles for DSP Instructions

Instruction	Mnemonic (indicates the common operation when the size is omitted)	Number of Cycles
DSP instructions	• {EMULA, EMACA, EMSBA, MULLH, MULHI, MULLO, MACLH, MACHI, MACLO, MSBLH, MSBHI, MSBLO} "Rs, Rs2, Ad" • {MVFACHI, MVFACMI, MVFACLO, MVFACGU} "#IMM, As, Rd" • {MVTACHI, MVTACLO, MVTACGU} "As, Rd" • {RDACW, RDACL, RACW, RACL} "#IMM, Ad"	1

Table 2.19 Number of Cycles for String Manipulation Instructions

Instruction	Mnemonic (indicates the common operation when the size is omitted)	Number of Cycles
String manipulation instructions* ¹	• SCMPU	$2+4\times\text{floor}(n/4)+4\times(n\%4)$ n: Number of comparison bytes* ²
	• SMOVB	$n>3?6+3\times\text{floor}(n/4)+3\times(n\%4):2+3n$ n: Number of transfer bytes* ²
	• SMOVF, SMOVU	$2+3\times\text{floor}(n/4)+3\times(n\%4)$ n: Number of transfer bytes* ²
	• SSTR.B	$2+\text{floor}(n/4)+n\%4$ n: Number of transfer bytes* ²
	• SSTR.W	$2+\text{floor}(n/2)+n\%2$ n: Number of transfer words* ²
	• SSTR.L	$2+n$ n: Number of transfer longwords
	• SUNTIL.B, SWHILE.B	$3+3\times\text{floor}(n/4)+3\times(n\%4)$ n: Number of comparison bytes* ²
	• SUNTIL.W, SWHILE.W	$3+3\times\text{floor}(n/2)+3\times(n\%2)$ n: Number of comparison words* ²
	• SUNTIL.L, SWHILE.L	$3+3\times n$ n: Number of comparison longwords

?: Conditional operator

Note 1. Each of the SCMPU, SMOVU, SWHILE, and SUNTIL instructions ends the execution regardless of the specified cycles, if the end condition is satisfied during execution.

Note 2. floor (x): Max. integer that is smaller than x.

Table 2.20 Number of Cycles for System Manipulation Instructions

Instruction	Mnemonic (indicates the common operation when the size is omitted)	Number of Cycles
System manipulation instructions	• {CLRPSW, SETPSW}“flag” • MVTC “#IMM, CR”/“Rs, CR” • MVFC “CR, Rd” • MVTIPL “#IMM”	1
	• RTE	6
	• RTFI	3

2.8.2 Numbers of Cycles for Response to Interrupts

Table 2.21 lists numbers of cycles taken by processing for response to interrupts.

Table 2.21 Numbers of Cycles for Response to Interrupts

Type of Interrupt Request/Details of Processing	Fast Interrupt	Other Interrupts
ICU Judgment of priority order	2 cycles	
CPU Number of cycles from notification to acceptance of the interrupt request	N cycles (varies with the instruction being executed at the time the interrupt was received)	
CPU Pre-processing by hardware Saving the current PC and PSW values in RAM (or in control registers in the case of the fast interrupt) Reading of the vector Branching to the start of the exception handling routine	4 cycles	6 cycles

Times calculated from the values in Table 2.21 will be applicable when access to memory from the CPU is processed with no waiting. The RAM and code flash memory in this MCU allow such access. Numbers of cycles for response to interrupts can be minimized by placing program code (and vectors) in code flash memory and the stack in RAM.

Furthermore, place the addresses where the exception handling routine start on 8-byte boundaries.

For information on the number of cycles from notification to acceptance of the interrupt request, indicated by N in the table above, see Table 2.13 to Table 2.20.

The timing of interrupt acceptance depends on the execution state of the instruction. For more information on this, see section 13.3.1, Acceptance Timing and Saved PC Value.

3. Operating Modes

3.1 Operating Mode Types and Selection

The operating mode is selectable in either of the following ways.

- Level on the mode-setting pin (MD) on release from a reset (an RES# pin reset or power-on reset)
- Control by software following release from the reset state

Table 3.1 shows the relationships between the level on the mode-setting pin and the operating mode to be selected. For details on each of the operating modes, refer to section 3.3, Details of Operating Modes.

Table 3.1 Selection of Operating Modes by the Mode-Setting Pin

Mode-Setting Pin	Operating Mode
MD	
High*1, *2	Single-chip mode
Low*1	Boot mode (SCI interface)
Low→High*3	Boot mode (FINE interface)

Note 1. Since the period between the release from the reset state and the wait time after the release or the wait time after the release from the power-on reset state is the period that the transition of the operating mode is in progress, do not change the input level on the MD pin. For details of the wait time after the release from the reset state and the wait time after the release from the power-on reset state, refer to section 38, Electrical Characteristics. In boot mode, an input pull-up resistor is in the enabled state during the period from the resetting in progress to the completion of the operating mode transition. In single-chip mode, an input pull-up resistor is in the enabled state during the period from the resetting in progress to the setting of disabling the input pull-up resistor in the pull-up control register.

Note 2. The PG7 pin function and several input and output functions for certain on-chip peripheral modules are multiplexed on the same pin as the MD pin function. After the chip has started up in the single-chip mode, the pin can be used for one of these other pin functions.

Note 3. Controlled by a serial programmer or emulator.

The endian is selectable in single-chip mode. Endian is set by the MDE.MDE[2:0] bits in the option-setting memory. For the correspondence between the setting and endian, refer to Table 3.2.

Table 3.2 Endian Setting in Single-Chip Mode

MDE.MDE[2:0] Bits	Endian
000b	Big endian
111b	Little endian

3.2 Register Descriptions

3.2.1 Mode Monitor Register (MDMONR)

Address(es): SYSTEM.MDMONR 0008 0000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MD
Value after reset:	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0	0/1 *1

Bit	Symbol	Bit Name	Description	R/W
b0	MD	MD Pin Status Flag	0: The MD pin is low. 1: The MD pin is high.	R
b7 to b1	—	Reserved	These bits are read as 0.	R
b8	—	Reserved	The read value is undefined.	R
b15 to b9	—	Reserved	These bits are read as 0.	R

Note 1. The initial value of the MD bit depends on the setting of the mode pin (MD). For the setting PORTG.PMR.b7, if the PG7/MD pin is set to the general purpose I/O port pin, this bit is read as 0 regardless of the state of the PG7/MD pin.

3.2.2 System Control Register 1 (SYSCR1)

Address(es): SYSTEM.SYSCR1 0008 0008h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RAME
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	RAME	RAM Enable	0: The RAM is disabled. 1: The RAM is enabled.	R/W
b15 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

RAME Bit (RAM Enable)

The RAME bit enables or disables the RAM.

A 0 should not be written to this bit during access to the RAM. When accessing the RAM immediately after changing the RAME bit from 0 (RAM disabled) to 1 (RAM enabled), make sure that the RAME bit is 1 before the access.

Even when the RAME bit is set to 0, the RAM retains its value. To retain the value in the RAM, keep the specified RAM standby voltage (VRAM). For details, refer to section 38, Electrical Characteristics.

3.3 Details of Operating Modes

3.3.1 Single-Chip Mode

In this mode, all I/O ports can be used as general input/output ports, peripheral function input/output, or interrupt input pins.

The chip starts up in single-chip mode if the high level is on the MD pin on release from the reset state.

3.3.2 Boot Mode (SCI Interface)

In this mode, the on-chip flash memory modifying program (boot program) stored in a dedicated area within the MCU operates. The on-chip flash memory (ROM and E2 DataFlash) can be modified from outside the MCU by using a universal asynchronous receiver/transmitter (SCI1). For details, refer to **section 37, Flash Memory (FLASH)**.

The chip starts up in boot mode if the low level is on the MD pin on release from the reset state.

When a reset is released while the MD pin is low, boot mode (SCI interface) is selected. For details on boot mode (SCI interface), refer to **section 37.8.1, Boot Mode (SCI Interface)**.

3.3.3 Boot Mode (FINE Interface)

In this mode, the flash memory modifying program (boot program) stored in a dedicated area within the MCU operates. The on-chip flash memory (ROM and E2 DataFlash) can be modified from outside the MCU by using the FINE. For details, refer to **section 37, Flash Memory (FLASH)**.

The chip starts up in boot mode (FINE interface) when the MD pin is set to the low level at the time of release from the reset state and then is then switched to the high level within 20 to 100 msec.

3.4 Transitions of Operating Modes

3.4.1 Mode Setting Pin Levels and Operating Mode Transitions

Figure 3.1 shows operating mode transitions according to the setting of the MD pin.

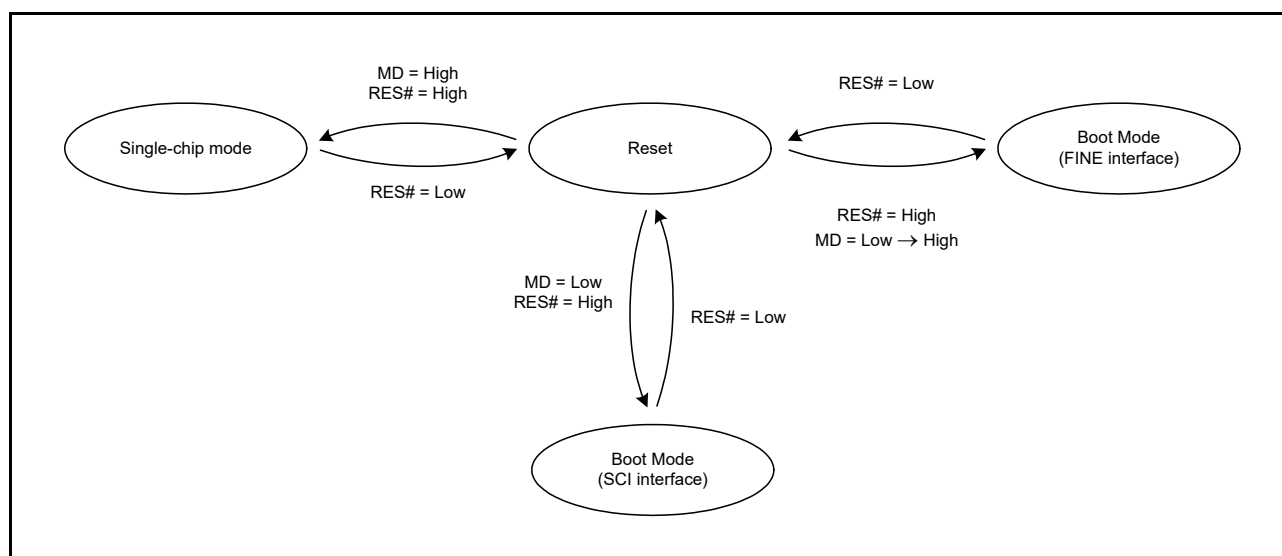


Figure 3.1 Mode Setting Pin Levels and Operating Modes

4. Address Space

4.1 Address Space

This MCU has a 4-Gbyte address space, consisting of the range of addresses from 0000 0000h to FFFF FFFFh. That is, linear access to an address space of up to 4 Gbytes is possible, and this contains both program and data areas. Figure 4.1 shows the memory maps.

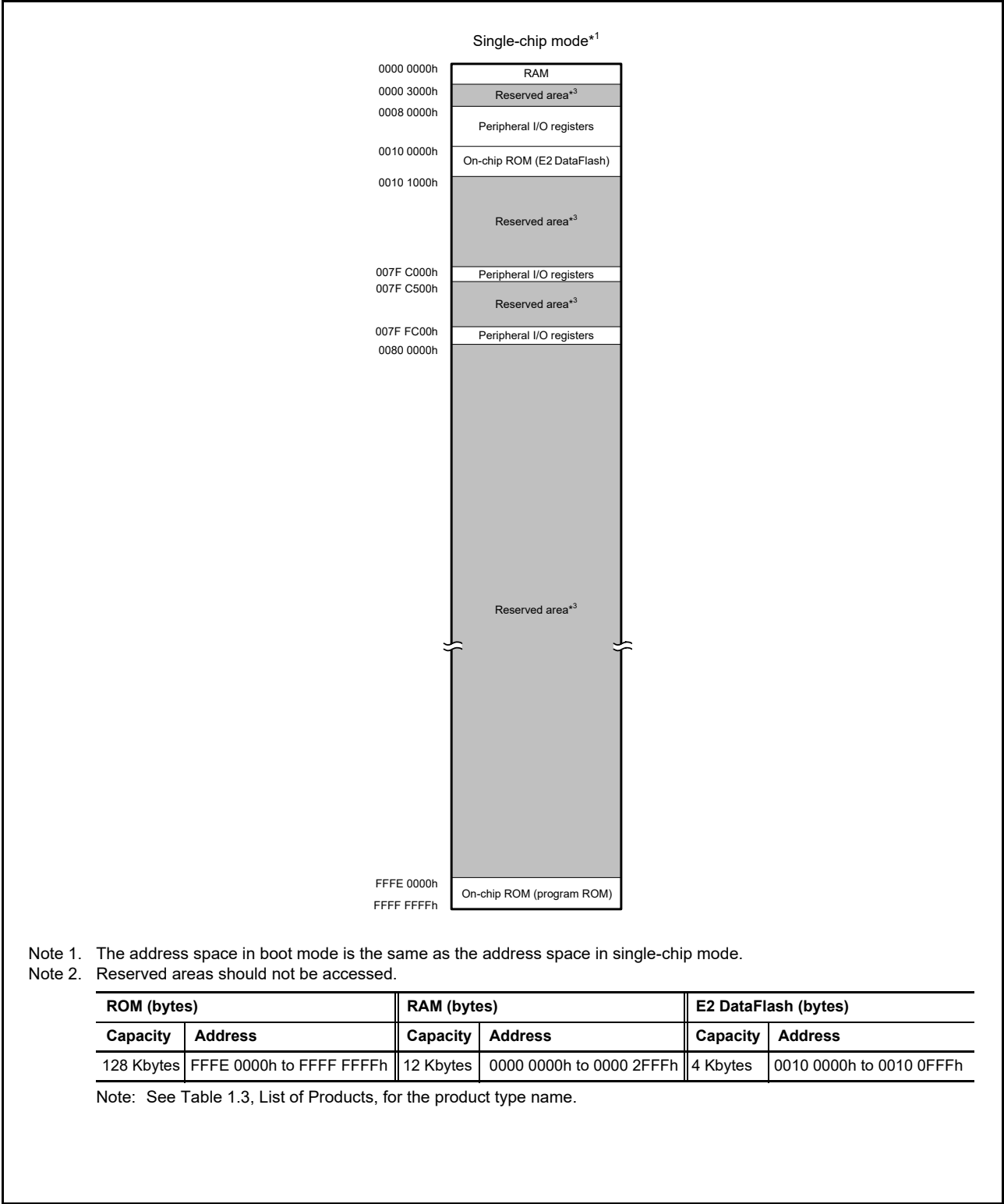


Figure 4.1 Memory Map in Each Operating Mode

5. I/O Registers

This section provides information on the on-chip I/O register addresses and bit configuration. The information is given as shown below. Notes on writing to registers are also given below.

(1) I/O register addresses (address order)

- Registers are listed from the lower allocation addresses.
- Registers are classified according to module symbols.
- Numbers of cycles for access indicate numbers of cycles of the given base clock.
- Among the internal I/O register area, addresses not listed in the list of registers are reserved. Reserved addresses must not be accessed. Do not access these addresses; otherwise, the operation when accessing these bits and subsequent operations cannot be guaranteed.

(2) Notes on writing to I/O registers

When writing to an I/O register, the CPU starts executing the subsequent instruction before completing I/O register write. This may cause the subsequent instruction to be executed before the post-update I/O register value is reflected on the operation.

As described in the following examples, special care is required for the cases in which the subsequent instruction must be executed after the post-update I/O register value is actually reflected.

[Examples of cases requiring special care]

- The subsequent instruction must be executed while an interrupt request is disabled with the IENj bit in IERN of the ICU (interrupt request enable bit) cleared to 0.
- A WAIT instruction is executed immediately after the preprocessing for causing a transition to the low power consumption state.

In the above cases, after writing to an I/O register, wait until the write operation is completed using the following procedure and then execute the subsequent instruction.

- Write to an I/O register.
- Read the value from the I/O register to a general register.
- Execute the operation using the value read.
- Execute the subsequent instruction.

[Instruction examples]

- Byte-size I/O registers

```
MOV.L #SFR_ADDR, R1
MOV.B #SFR_DATA, [R1]
CMP [R1].UB, R1
;; Next process
```

- Word-size I/O registers

```
MOV.L #SFR_ADDR, R1
MOV.W #SFR_DATA, [R1]
CMP [R1].W, R1
;; Next process
```

- Longword-size I/O registers

```
MOV.L #SFR_ADDR, R1
MOV.L #SFR_DATA, [R1]
CMP [R1].L, R1
;; Next process
```

If multiple registers are written to and a subsequent instruction should be executed after the write operations are entirely completed, only read the I/O register that was last written to and execute the operation using the value; it is not necessary to read or execute operation for all the registers that were written to.

(3) Number of Access Cycles to I/O Registers

For numbers of clock cycles for access to I/O registers, see Table 5.1, List of I/O Registers (Address Order). The number of access cycles to I/O registers is obtained by following equation.*1

$$\begin{aligned} \text{Number of access cycles to I/O registers} = & \text{Number of bus cycles for internal main bus 1} + \\ & \text{Number of divided clock synchronization cycles} + \\ & \text{Number of bus cycles for internal peripheral buses 1 to 3, and 6} \end{aligned}$$

The number of bus cycles of internal peripheral buses 1 to 3, and 6 differs according to the register to be accessed.

When the registers for peripheral functions connected to internal peripheral buses 2, 3, and 6 (except for bus error related registers) are accessed, the number of divided clock synchronization cycles is added.

The number of divided clock synchronization cycles differs depending on the frequency ratio between ICLK and PCLK (or FCLK) or bus access timing.

In the peripheral function unit, when the frequency ratio of ICLK is equal to or greater than that of PCLK (or FCLK), the sum of the number of bus cycles for internal main bus 1 and the number of the divided clock synchronization cycles will be one cycle of PCLK (or FCLK) at a maximum. Therefore, one PCLK (or FCLK) has been added to the number of access cycles shown in Table 5.1.

When the frequency ratio of ICLK is lower than that of PCLK (or FCLK), the subsequent bus access is started from the ICLK cycle following the completion of the access to the peripheral functions. Therefore, the access cycles are described on an ICLK basis.

Note 1. This applies to the number of cycles when the access from the CPU does not conflict with the instruction fetching to the external memory or bus access from the different bus master (DTC).

(4) Restrictions in Relation to RMPA and String-Manipulation Instructions

The allocation of data to be handled by RMPA or string-manipulation instructions to I/O registers is prohibited, and operation is not guaranteed if this restriction is not observed.

(5) Notes on Sleep Mode and Mode Transitions

During sleep mode or mode transitions, do not write to the system control related registers (indicated by 'SYSTEM' in the Module Symbol column in Table 5.1, List of I/O Registers (Address Order)).

5.1 I/O Register Addresses (Address Order)

Table 5.1 List of I/O Registers (Address Order) (1 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 0000h	SYSTEM	Mode Monitor Register	MDMONR	16	16	3 ICLK	section 3.
0008 0008h	SYSTEM	System Control Register 1	SYSCR1	16	16		
0008 000Ch	SYSTEM	Standby Control Register	SBYCR	16	16		
0008 0010h	SYSTEM	Module Stop Control Register A	MSTPCRA	32	32	3 ICLK	section 11.
0008 0014h	SYSTEM	Module Stop Control Register B	MSTPCRB	32	32		
0008 0018h	SYSTEM	Module Stop Control Register C	MSTPCRC	32	32		
0008 0020h	SYSTEM	System Clock Control Register	SCKCR	32	32	3 ICLK	section 9.
0008 0026h	SYSTEM	System Clock Control Register 3	SCKCR3	16	16		
0008 0028h	SYSTEM	PLL Control Register	PLLCR	16	16		
0008 002Ah	SYSTEM	PLL Control Register 2	PLLCR2	8	8	3 ICLK	section 9.
0008 0032h	SYSTEM	Main Clock Oscillator Control Register	MOSCCR	8	8		
0008 0034h	SYSTEM	Low-Speed On-Chip Oscillator Control Register	LOCOCR	8	8		
0008 0035h	SYSTEM	IWDT-Dedicated On-Chip Oscillator Control Register	ILOCOCR	8	8	3 ICLK	section 9.
0008 0036h	SYSTEM	High-Speed On-Chip Oscillator Control Register	HOCOCR	8	8		
0008 003Ch	SYSTEM	Oscillation Stabilization Flag Register	OSCOVFSR	8	8		
0008 003Eh	SYSTEM	CLKOUT Output Control Register	CKOCR	16	16	3 ICLK	section 9.
0008 0040h	SYSTEM	Oscillation Stop Detection Control Register	OSTDCR	8	8		
0008 0041h	SYSTEM	Oscillation Stop Detection Status Register	OSTDSR	8	8		
0008 0061h	SYSTEM	Low-Speed On-Chip Oscillator Trimming Register 2	LOCOTRR2	8	8	3 ICLK	section 9.
0008 0064h	SYSTEM	IWDT-Dedicated On-Chip Oscillator Trimming Register	ILOCOTRR	8	8		
0008 0068h	SYSTEM	High-Speed On-Chip Oscillator Trimming Register 0	HOCOTRR0	8	8		
0008 00A0h	SYSTEM	Operating Power Control Register	OPCCR	8	8	3 ICLK	section 11.
0008 00A2h	SYSTEM	Main Clock Oscillator Wait Control Register	MOSCWTCT	8	8		
0008 00C0h	SYSTEM	Reset Status Register 2	RSTS2	8	8		
0008 00C2h	SYSTEM	Software Reset Register	SWRR	16	16	3 ICLK	section 8.
0008 00E0h	SYSTEM	Voltage Monitoring 1 Circuit Control Register 1	LVD1CR1	8	8		
0008 00E1h	SYSTEM	Voltage Monitoring 1 Circuit Status Register	LVD1SR	8	8		
0008 00E2h	SYSTEM	Voltage Monitoring 2 Circuit Control Register 1	LVD2CR1	8	8	3 ICLK	section 12.
0008 00E3h	SYSTEM	Voltage Monitoring 2 Circuit Status Register	LVD2SR	8	8		
0008 03FEh	SYSTEM	Protect Register	PRCR	16	16		
0008 1300h	BSC	Bus Error Status Clear Register	BERCLR	8	8	2 ICLK	section 15.
0008 1304h	BSC	Bus Error Monitoring Enable Register	BEREN	8	8		
0008 1308h	BSC	Bus Error Status Register 1	BERSR1	8	8		
0008 130Ah	BSC	Bus Error Status Register 2	BERSR2	16	16	2 ICLK	section 16.
0008 1310h	BSC	Bus Priority Control Register	BUSPRI	16	16		
0008 2400h	DTC	DTC Control Register	DTCCR	8	8		
0008 2404h	DTC	DTC Vector Base Register	DTCVBR	32	32	2 ICLK	section 16.
0008 2408h	DTC	DTC Address Mode Register	DTCADMOD	8	8		
0008 240Ch	DTC	DTC Module Start Register	DTCTST	8	8		
0008 240Eh	DTC	DTC Status Register	DTCTST	16	16	2 ICLK	section 16.
0008 2410h	DTC	DTC Index Table Base Register	DTCIBR	32	32		
0008 2414h	DTC	DTC Operation Register	DTCOR	8	8		
0008 2416h	DTC	DTC Sequence Transfer Enable Register	DTCSQE	16	16	2 ICLK	section 16.
0008 2418h	DTC	DTC Address Displacement Register	DTCDISP	32	32		
0008 7010h to 0008 70FFh	ICU	Interrupt Request Register 016 to Interrupt Request Register 255	IR016 to IR255	8	8		
0008 711Bh to 0008 71FFh	ICU	DTC Transfer Request Enable Register 027 to DTC Transfer Request Enable Register 255	DTCTER027 to DTCTER255	8	8	2 ICLK	section 16.
0008 7202h to 0008 721Fh	ICU	Interrupt Request Enable Register 02 to Interrupt Request Enable Register 1F	IER02 to IER1F	8	8		

Table 5.1 List of I/O Registers (Address Order) (2 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 72E0h	ICU	Software Interrupt Generation Register	SWINTR	8	8	2 ICLK	section 14.
0008 72F0h	ICU	Fast Interrupt Set Register	FIR	16	16		
0008 7300h to 0008 73FFh	ICU	Interrupt Source Priority Register 000 to Interrupt Source Priority Register 255	IPR000 to IPR255	8	8		
0008 7500h to 0008 7507h	ICU	IRQ Control Register 0 to IRQ Control Register 7	IRQCR0 to IRQCR7	8	8		
0008 7510h	ICU	IRQ Pin Digital Filter Enable Register 0	IRQFLTE0	8	8		
0008 7514h	ICU	IRQ Pin Digital Filter Setting Register 0	IRQFLTC0	16	16		
0008 7580h	ICU	Non-Maskable Interrupt Status Register	NMISR	8	8		
0008 7581h	ICU	Non-Maskable Interrupt Enable Register	NMIER	8	8		
0008 7582h	ICU	Non-Maskable Interrupt Status Clear Register	NMICLR	8	8		
0008 7583h	ICU	NMI Pin Interrupt Control Register	NMICR	8	8		
0008 7590h	ICU	NMI Pin Digital Filter Enable Register	NMIFLTE	8	8		
0008 7594h	ICU	NMI Pin Digital Filter Setting Register	NMIFLTC	8	8		
0008 8000h	CMT	Compare Match Timer Start Register 0	CMSTR0	16	16	2 or 3 PCLKB	section 24.
0008 8002h	CMT0	Compare Match Timer Control Register	CMCR	16	16		
0008 8004h	CMT0	Compare Match Counter	CMCNT	16	16		
0008 8006h	CMT0	Compare Match Constant Register	CMCOR	16	16		
0008 8008h	CMT1	Compare Match Timer Control Register	CMCR	16	16		
0008 800Ah	CMT1	Compare Match Counter	CMCNT	16	16		
0008 800Ch	CMT1	Compare Match Constant Register	CMCOR	16	16	2 or 3 PCLKB	section 25.
0008 8030h	IWDT	IWDT Refresh Register	IWDTRR	8	8		
0008 8032h	IWDT	IWDT Control Register	IWDTCR	16	16		
0008 8034h	IWDT	IWDT Status Register	IWDTSR	16	16		
0008 8036h	IWDT	IWDT Reset Control Register	IWDTRCR	8	8		
0008 8038h	IWDT	IWDT Count Stop Control Register	IWDTCSTPR	8	8		
0008 80C0h	DA	D/A Data Register 0	DADR0	16	16	2 or 3 PCLKB	section 32.
0008 80C2h	DA	D/A Data Register 1	DADR1	16	16		
0008 80C4h	DA	D/A Control Register	DACR	8	8		
0008 80C5h	DA	Data Register Format Select Register	DADPR	8	8		
0008 80C6h	DA	D/A/A/D Synchronous Start Control Register	DAADSCR	8	8		
0008 8200h	TMR0	Timer Control Register	TCR	8	8	2 or 3 PCLKB	section 23.
0008 8201h	TMR1	Timer Control Register	TCR	8	8		
0008 8202h	TMR0	Timer Control/Status Register	TCSR	8	8		
0008 8203h	TMR1	Timer Control/Status Register	TCSR	8	8		
0008 8204h	TMR0	Time Constant Register A	TCORA	8	8		
0008 8204h	TMR01	Time Constant Register A	TCORA	16	16		
0008 8205h	TMR1	Time Constant Register A	TCORA	8	8		
0008 8206h	TMR0	Time Constant Register B	TCORB	8	8		
0008 8206h	TMR01	Time Constant Register B	TCORB	16	16		
0008 8207h	TMR1	Time Constant Register B	TCORB	8	8		
0008 8208h	TMR0	Timer Counter	TCNT	8	8		
0008 8208h	TMR01	Timer Counter	TCNT	16	16		
0008 8209h	TMR1	Timer Counter	TCNT	8	8		
0008 820Ah	TMR0	Timer Counter Control Register	TCCR	8	8		
0008 820Ah	TMR01	Timer Counter Control Register	TCCR	16	16		
0008 820Bh	TMR1	Timer Counter Control Register	TCCR	8	8		
0008 8210h	TMR2	Timer Control Register	TCR	8	8		
0008 8211h	TMR3	Timer Control Register	TCR	8	8		
0008 8212h	TMR2	Timer Control/Status Register	TCSR	8	8		
0008 8213h	TMR3	Timer Control/Status Register	TCSR	8	8		
0008 8214h	TMR2	Time Constant Register A	TCORA	8	8		

Table 5.1 List of I/O Registers (Address Order) (3 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 8214h	TMR23	Time Constant Register A	TCORA	16	16	2 or 3 PCLKB	section 23.
0008 8215h	TMR3	Time Constant Register A	TCORA	8	8		
0008 8216h	TMR2	Time Constant Register B	TCORB	8	8		
0008 8216h	TMR23	Time Constant Register B	TCORB	16	16		
0008 8217h	TMR3	Time Constant Register B	TCORB	8	8		
0008 8218h	TMR2	Timer Counter	TCNT	8	8		
0008 8218h	TMR23	Timer Counter	TCNT	16	16		
0008 8219h	TMR3	Timer Counter	TCNT	8	8		
0008 821Ah	TMR2	Timer Counter Control Register	TCCR	8	8		
0008 821Ah	TMR23	Timer Counter Control Register	TCCR	16	16		
0008 821Bh	TMR3	Timer Counter Control Register	TCCR	8	8		
0008 8280h	CRC	CRC Control Register	CRCCR	8	8	2 or 3 PCLKB	section 28.
0008 8284h	CRC	CRC Data Input Register	CRCDIR	32	8, 32		
0008 8288h	CRC	CRC Data Output Register	CRCDOR	32	8, 16, 32		
0008 8300h	RIIC0	I ² C-bus Control Register 1	ICCR1	8	8	2 or 3 PCLKB	section 27.
0008 8301h	RIIC0	I ² C-bus Control Register 2	ICCR2	8	8		
0008 8302h	RIIC0	I ² C-bus Mode Register 1	ICMR1	8	8		
0008 8303h	RIIC0	I ² C-bus Mode Register 2	ICMR2	8	8		
0008 8304h	RIIC0	I ² C-bus Mode Register 3	ICMR3	8	8		
0008 8305h	RIIC0	I ² C-bus Function Enable Register	ICFER	8	8		
0008 8306h	RIIC0	I ² C-bus Status Enable Register	ICSER	8	8		
0008 8307h	RIIC0	I ² C-bus Interrupt Enable Register	ICIER	8	8		
0008 8308h	RIIC0	I ² C-bus Status Register 1	ICSR1	8	8		
0008 8309h	RIIC0	I ² C-bus Status Register 2	ICSR2	8	8		
0008 830Ah	RIIC0	Slave Address Register L0	SARL0	8	8		
0008 830Bh	RIIC0	Slave Address Register U0	SARU0	8	8		
0008 830Ch	RIIC0	Slave Address Register L1	SARL1	8	8		
0008 830Dh	RIIC0	Slave Address Register U1	SARU1	8	8		
0008 830Eh	RIIC0	Slave Address Register L2	SARL2	8	8		
0008 830Fh	RIIC0	Slave Address Register U2	SARU2	8	8		
0008 8310h	RIIC0	I ² C-bus Bit Rate Low-Level Register	ICBRL	8	8	2 or 3 PCLKB	section 30.
0008 8311h	RIIC0	I ² C-bus Bit Rate High-Level Register	ICBRH	8	8		
0008 8312h	RIIC0	I ² C-bus Transmit Data Register	ICDRT	8	8		
0008 8313h	RIIC0	I ² C-bus Receive Data Register	ICDRR	8	8		
0008 9000h	S12AD	A/D Control Register	ADCSR	16	16		
0008 9004h	S12AD	A/D Channel Select Register A0	ADANSA0	16	16		
0008 9008h	S12AD	A/D-Converted Value Addition/Average Function Channel Select Register 0	ADADS0	16	16		
0008 900Ch	S12AD	A/D-Converted Value Addition/Average Count Select Register	ADADC	8	8		
0008 900Eh	S12AD	A/D Control Extended Register	ADCER	16	16		
0008 9010h	S12AD	A/D Conversion Start Trigger Select Register	ADSTRGR	16	16		
0008 9014h	S12AD	A/D Channel Select Register B0	ADANSB0	16	16		
0008 9018h	S12AD	A/D Data Duplication Register	ADDBLDR	16	16		
0008 901Eh	S12AD	A/D Self-Diagnosis Data Register	ADRD	16	16		
0008 9020h	S12AD	A/D Data Register 0	ADDR0	16	16		
0008 9022h	S12AD	A/D Data Register 1	ADDR1	16	16		
0008 9024h	S12AD	A/D Data Register 2	ADDR2	16	16		
0008 9026h	S12AD	A/D Data Register 3	ADDR3	16	16		
0008 9028h	S12AD	A/D Data Register 4	ADDR4	16	16		
0008 902Ah	S12AD	A/D Data Register 5	ADDR5	16	16		
0008 902Ch	S12AD	A/D Data Register 6	ADDR6	16	16		
0008 902Eh	S12AD	A/D Data Register 7	ADDR7	16	16		

Table 5.1 List of I/O Registers (Address Order) (4 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 907Ah	S12AD	A/D Disconnection Detection Control Register	ADDISCR	8	8	2 or 3 PCLKB	section 30.
0008 9080h	S12AD	A/D Group Scan Priority Control Register	ADGSPCR	16	16		
0008 9084h	S12AD	A/D Data Duplication Register A	ADDBLDRA	16	16		
0008 9086h	S12AD	A/D Data Duplication Register B	ADDBLDRB	16	16		
0008 908Ah	S12AD	A/D High-Potential/Low-Potential Reference Voltage Control Register	ADHVREFCNT	8	8		
0008 90D4h	S12AD	A/D Channel Select Register C0	ADANSC0	16	16		
0008 90D9h	S12AD	A/D Group C Trigger Select Register	ADGCTRGR	8	8		
0008 90E0h	S12AD	A/D Sampling State Register 0	ADSSTR0	8	8		
0008 90E1h	S12AD	A/D Sampling State Register 1	ADSSTR1	8	8		
0008 90E2h	S12AD	A/D Sampling State Register 2	ADSSTR2	8	8		
0008 90E3h	S12AD	A/D Sampling State Register 3	ADSSTR3	8	8		
0008 90E4h	S12AD	A/D Sampling State Register 4	ADSSTR4	8	8		
0008 90E5h	S12AD	A/D Sampling State Register 5	ADSSTR5	8	8		
0008 90E6h	S12AD	A/D Sampling State Register 6	ADSSTR6	8	8		
0008 90E7h	S12AD	A/D Sampling State Register 7	ADSSTR7	8	8		
0008 91A0h	PGA0	Programmable Gain Amplifier Control Register	PGACR	16	16	2 or 3 PCLKB	section 31.
0008 91A2h	PGA0	Programmable Gain Amplifier Gain Setting Register 0	PGAGS0	16	16		
0008 91E0h	PGA0	Programmable Gain Amplifier Monitor Output Control Register	PGAMONCR	8	8		
0008 9200h	S12AD1	A/D Control Register	ADCSR	16	16	2 or 3 PCLKB	section 30.
0008 9204h	S12AD1	A/D Channel Select Register A0	ADANSA0	16	16		
0008 9208h	S12AD1	A/D-Converted Value Addition/Average Function Channel Select Register 0	ADADS0	16	16		
0008 920Ch	S12AD1	A/D-Converted Value Addition/Average Count Select Register	ADADC	8	8		
0008 920Eh	S12AD1	A/D Control Extended Register	ADCER	16	16		
0008 9210h	S12AD1	A/D Conversion Start Trigger Select Register	ADSTRGR	16	16		
0008 9212h	S12AD1	A/D Conversion Extended Input Control Register	ADEXICR	16	16		
0008 9214h	S12AD1	A/D Channel Select Register B0	ADANSB0	16	16		
0008 9218h	S12AD1	A/D Data Duplication Register	ADDBLDR	16	16		
0008 921Ah	S12AD1	A/D Temperature Sensor Data Register	ADTSDR	16	16		
0008 921Ch	S12AD1	A/D Internal Reference Voltage Data Register	ADOCDR	16	16		
0008 921Eh	S12AD1	A/D Self-Diagnosis Data Register	ADRD	16	16		
0008 9220h	S12AD1	A/D Data Register 0	ADDR0	16	16		
0008 9222h	S12AD1	A/D Data Register 1	ADDR1	16	16		
0008 9224h	S12AD1	A/D Data Register 2	ADDR2	16	16		
0008 9226h	S12AD1	A/D Data Register 3	ADDR3	16	16		
0008 9228h	S12AD1	A/D Data Register 4	ADDR4	16	16		
0008 922Ah	S12AD1	A/D Data Register 5	ADDR5	16	16		
0008 922Ch	S12AD1	A/D Data Register 6	ADDR6	16	16		
0008 922Eh	S12AD1	A/D Data Register 7	ADDR7	16	16		
0008 927Ah	S12AD1	A/D Disconnection Detection Control Register	ADDISCR	8	8		
0008 9280h	S12AD1	A/D Group Scan Priority Control Register	ADGSPCR	16	16		
0008 9284h	S12AD1	A/D Data Duplication Register A	ADDBLDRA	16	16		
0008 9286h	S12AD1	A/D Data Duplication Register B	ADDBLDRB	16	16		
0008 928Ah	S12AD1	A/D High-Potential/Low-Potential Reference Voltage Control Register	ADHVREFCNT	8	8		
0008 92D4h	S12AD1	A/D Channel Select Register C0	ADANSC0	16	16		
0008 92D9h	S12AD1	A/D Group C Trigger Select Register	ADGCTRGR	8	8		
0008 92DEh	S12AD1	A/D Sampling State Register T	ADSSTRT	8	8		
0008 92DFh	S12AD1	A/D Sampling State Register O	ADSSTRO	8	8		
0008 92E0h	S12AD1	A/D Sampling State Register 0	ADSSTR0	8	8		
0008 92E1h	S12AD1	A/D Sampling State Register 1	ADSSTR1	8	8		
0008 92E2h	S12AD1	A/D Sampling State Register 2	ADSSTR2	8	8		
0008 92E3h	S12AD1	A/D Sampling State Register 3	ADSSTR3	8	8		

Table 5.1 List of I/O Registers (Address Order) (5 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section	
0008 92E4h	S12AD1	A/D Sampling State Register 4	ADSSTR4	8	8	2 or 3 PCLKB	section 30.	
0008 92E5h	S12AD1	A/D Sampling State Register 5	ADSSTR5	8	8			
0008 92E6h	S12AD1	A/D Sampling State Register 6	ADSSTR6	8	8			
0008 92E7h	S12AD1	A/D Sampling State Register 7	ADSSTR7	8	8			
0008 A020h	SCI1	Serial Mode Register	SMR	8	8	2 or 3 PCLKB	section 26.	
0008 A020h	SMCI1	Serial Mode Register	SMR	8	8			
0008 A021h	SCI1	Bit Rate Register	BRR	8	8			
0008 A022h	SCI1	Serial Control Register	SCR	8	8			
0008 A022h	SMCI1	Serial Control Register	SCR	8	8			
0008 A023h	SCI1	Transmit Data Register	TDR	8	8			
0008 A024h	SCI1	Serial Status Register	SSR	8	8			
0008 A024h	SMCI1	Serial Status Register	SSR	8	8			
0008 A025h	SCI1	Receive Data Register	RDR	8	8			
0008 A026h	SCI1	Smart Card Mode Register	SCMR	8	8			
0008 A026h	SMCI1	Smart Card Mode Register	SCMR	8	8			
0008 A027h	SCI1	Serial Extended Mode Register	SEMR	8	8			
0008 A028h	SCI1	Noise Filter Setting Register	SNFR	8	8			
0008 A029h	SCI1	I ² C Mode Register 1	SIMR1	8	8			
0008 A02Ah	SCI1	I ² C Mode Register 2	SIMR2	8	8			
0008 A02Bh	SCI1	I ² C Mode Register 3	SIMR3	8	8			
0008 A02Ch	SCI1	I ² C Status Register	SISR	8	8			
0008 A02Dh	SCI1	SPI Mode Register	SPMR	8	8			
0008 A02Eh	SCI1	Transmit Data Register HL	TDRHL	16	16	4 or 5 PCLKB		
0008 A02Eh	SCI1	Transmit Data Register H	TDRH	8	8	2 or 3 PCLKB		
0008 A02Fh	SCI1	Transmit Data Register L	TDRL	8	8			
0008 A030h	SCI1	Receive Data Register HL	RDRHL	16	16	4 or 5 PCLKB	2 or 3 PCLKB	
0008 A030h	SCI1	Receive Data Register H	RDRH	8	8			
0008 A031h	SCI1	Receive Data Register L	RDRL	8	8			
0008 A032h	SCI1	Modulation Duty Register	MDDR	8	8			
0008 A0A0h	SCI5	Serial Mode Register	SMR	8	8			
0008 A0A0h	SMCI5	Serial Mode Register	SMR	8	8			
0008 A0A1h	SCI5	Bit Rate Register	BRR	8	8			
0008 A0A2h	SCI5	Serial Control Register	SCR	8	8			
0008 A0A2h	SMCI5	Serial Control Register	SCR	8	8			
0008 A0A3h	SCI5	Transmit Data Register	TDR	8	8			
0008 A0A4h	SCI5	Serial Status Register	SSR	8	8			
0008 A0A4h	SMCI5	Serial Status Register	SSR	8	8			
0008 A0A5h	SCI5	Receive Data Register	RDR	8	8			
0008 A0A6h	SCI5	Smart Card Mode Register	SCMR	8	8			
0008 A0A6h	SMCI5	Smart Card Mode Register	SCMR	8	8			
0008 A0A7h	SCI5	Serial Extended Mode Register	SEMR	8	8			
0008 A0A8h	SCI5	Noise Filter Setting Register	SNFR	8	8			
0008 A0A9h	SCI5	I ² C Mode Register 1	SIMR1	8	8			
0008 A0AAh	SCI5	I ² C Mode Register 2	SIMR2	8	8			
0008 A0ABh	SCI5	I ² C Mode Register 3	SIMR3	8	8			
0008 A0ACh	SCI5	I ² C Status Register	SISR	8	8			
0008 A0ADh	SCI5	SPI Mode Register	SPMR	8	8			
0008 A0AEh	SCI5	Transmit Data Register HL	TDRHL	16	16	4 or 5 PCLKB		
0008 A0AEh	SCI5	Transmit Data Register H	TDRH	8	8	2 or 3 PCLKB		
0008 A0AFh	SCI5	Transmit Data Register L	TDRL	8	8			
0008 A0B0h	SCI5	Receive Data Register HL	RDRHL	16	16	4 or 5 PCLKB	2 or 3 PCLKB	
0008 A0B0h	SCI5	Receive Data Register H	RDRH	8	8			

Table 5.1 List of I/O Registers (Address Order) (6 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 A0B1h	SCI5	Receive Data Register L	RDRL	8	8	2 or 3 PCLKB	section 26.
0008 A0B2h	SCI5	Modulation Duty Register	MDDR	8	8		
0008 A0C0h	SCI6	Serial Mode Register	SMR	8	8	2 or 3 PCLKB	section 26.
0008 A0C0h	SMCI6	Serial Mode Register	SMR	8	8		
0008 A0C1h	SCI6	Bit Rate Register	BRR	8	8	2 or 3 PCLKB	section 26.
0008 A0C2h	SCI6	Serial Control Register	SCR	8	8		
0008 A0C2h	SMCI6	Serial Control Register	SCR	8	8	2 or 3 PCLKB	section 26.
0008 A0C3h	SCI6	Transmit Data Register	TDR	8	8		
0008 A0C4h	SCI6	Serial Status Register	SSR	8	8	2 or 3 PCLKB	section 26.
0008 A0C4h	SMCI6	Serial Status Register	SSR	8	8		
0008 A0C5h	SCI6	Receive Data Register	RDR	8	8	2 or 3 PCLKB	section 26.
0008 A0C6h	SCI6	Smart Card Mode Register	SCMR	8	8		
0008 A0C6h	SMCI6	Smart Card Mode Register	SCMR	8	8	2 or 3 PCLKB	section 26.
0008 A0C7h	SCI6	Serial Extended Mode Register	SEMR	8	8		
0008 A0C8h	SCI6	Noise Filter Setting Register	SNFR	8	8	2 or 3 PCLKB	section 26.
0008 A0C9h	SCI6	I ² C Mode Register 1	SIMR1	8	8		
0008 A0CAh	SCI6	I ² C Mode Register 2	SIMR2	8	8	2 or 3 PCLKB	section 26.
0008 A0CBh	SCI6	I ² C Mode Register 3	SIMR3	8	8		
0008 A0CCh	SCI6	I ² C Status Register	SISR	8	8	2 or 3 PCLKB	section 26.
0008 A0CDh	SCI6	SPI Mode Register	SPMR	8	8		
0008 A0CEh	SCI6	Transmit Data Register HL	TDRHL	16	16	4 or 5 PCLKB	section 26.
0008 A0CEh	SCI6	Transmit Data Register H	TDRH	8	8	2 or 3 PCLKB	
0008 A0CFh	SCI6	Transmit Data Register L	TDRL	8	8	2 or 3 PCLKB	section 26.
0008 A0D0h	SCI6	Receive Data Register HL	RDRHL	16	16	4 or 5 PCLKB	
0008 A0D0h	SCI6	Receive Data Register H	RDRH	8	8	2 or 3 PCLKB	section 26.
0008 A0D1h	SCI6	Receive Data Register L	RDRL	8	8		
0008 A0D2h	SCI6	Modulation Duty Register	MDDR	8	8	2 or 3 PCLKB	section 26.
0008 B000h	CAC	CAC Control Register 0	CACR0	8	8		
0008 B001h	CAC	CAC Control Register 1	CACR1	8	8	2 or 3 PCLKB	section 10.
0008 B002h	CAC	CAC Control Register 2	CACR2	8	8		
0008 B003h	CAC	CAC Interrupt Request Enable Register	CAICR	8	8	2 or 3 PCLKB	section 10.
0008 B004h	CAC	CAC Status Register	CASTR	8	8		
0008 B006h	CAC	CAC Upper-Limit Value Setting Register	CAULVR	16	16	2 or 3 PCLKB	section 35.
0008 B008h	CAC	CAC Lower-Limit Value Setting Register	CALLVR	16	16		
0008 B00Ah	CAC	CAC Counter Buffer Register	CACNTBR	16	16	2 or 3 PCLKB	section 35.
0008 B080h	DOC	DOC Control Register	DOCR	8	8		
0008 B082h	DOC	DOC Data Input Register	DODIR	16	16	2 or 3 PCLKB	section 35.
0008 B084h	DOC	DOC Data Setting Register	DODSR	16	16		
0008 B300h	SCI12	Serial Mode Register	SMR	8	8	2 or 3 PCLKB	section 26.
0008 B300h	SMCI12	Serial Mode Register	SMR	8	8		
0008 B301h	SCI12	Bit Rate Register	BRR	8	8	2 or 3 PCLKB	section 26.
0008 B302h	SCI12	Serial Control Register	SCR	8	8		
0008 B302h	SMCI12	Serial Control Register	SCR	8	8	2 or 3 PCLKB	section 26.
0008 B303h	SCI12	Transmit Data Register	TDR	8	8		
0008 B304h	SCI12	Serial Status Register	SSR	8	8	2 or 3 PCLKB	section 26.
0008 B304h	SMCI12	Serial Status Register	SSR	8	8		
0008 B305h	SCI12	Receive Data Register	RDR	8	8	2 or 3 PCLKB	section 26.
0008 B306h	SCI12	Smart Card Mode Register	SCMR	8	8		
0008 B306h	SMCI12	Smart Card Mode Register	SCMR	8	8	2 or 3 PCLKB	section 26.
0008 B307h	SCI12	Serial Extended Mode Register	SEMR	8	8		
0008 B308h	SCI12	Noise Filter Setting Register	SNFR	8	8	2 or 3 PCLKB	section 26.
0008 B309h	SCI12	I ² C Mode Register 1	SIMR1	8	8		

Table 5.1 List of I/O Registers (Address Order) (7 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 B30Ah	SCI12	I ² C Mode Register 2	SIMR2	8	8	2 or 3 PCLKB	section 26.
0008 B30Bh	SCI12	I ² C Mode Register 3	SIMR3	8	8		
0008 B30Ch	SCI12	I ² C Status Register	SISR	8	8		
0008 B30Dh	SCI12	SPI Mode Register	SPMR	8	8	4 or 5 PCLKB	
0008 B30Eh	SCI12	Transmit Data Register HL	TDRHL	16	16		
0008 B30Eh	SCI12	Transmit Data Register H	TDRH	8	8		
0008 B30Fh	SCI12	Transmit Data Register L	TDRL	8	8	2 or 3 PCLKB	
0008 B310h	SCI12	Receive Data Register HL	RDRHL	16	16		
0008 B310h	SCI12	Receive Data Register H	RDRH	8	8		
0008 B311h	SCI12	Receive Data Register L	RDRL	8	8	2 or 3 PCLKB	
0008 B312h	SCI12	Modulation Duty Register	MDDR	8	8		
0008 B320h	SCI12	Extended Serial Module Enable Register	ESMER	8	8		
0008 B321h	SCI12	Control Register 0	CR0	8	8		
0008 B322h	SCI12	Control Register 1	CR1	8	8		
0008 B323h	SCI12	Control Register 2	CR2	8	8		
0008 B324h	SCI12	Control Register 3	CR3	8	8		
0008 B325h	SCI12	Port Control Register	PCR	8	8		
0008 B326h	SCI12	Interrupt Control Register	ICR	8	8		
0008 B327h	SCI12	Status Register	STR	8	8		
0008 B328h	SCI12	Status Clear Register	STCR	8	8		
0008 B329h	SCI12	Control Field 0 Data Register	CF0DR	8	8		
0008 B32Ah	SCI12	Control Field 0 Compare Enable Register	CF0CR	8	8		
0008 B32Bh	SCI12	Control Field 0 Receive Data Register	CF0RR	8	8		
0008 B32Ch	SCI12	Primary Control Field 1 Data Register	PCF1DR	8	8		
0008 B32Dh	SCI12	Secondary Control Field 1 Data Register	SCF1DR	8	8		
0008 B32Eh	SCI12	Control Field 1 Compare Enable Register	CF1CR	8	8		
0008 B32Fh	SCI12	Control Field 1 Receive Data Register	CF1RR	8	8		
0008 B330h	SCI12	Timer Control Register	TCR	8	8		
0008 B331h	SCI12	Timer Mode Register	TMR	8	8		
0008 B332h	SCI12	Timer Prescaler Register	TPRE	8	8		
0008 B333h	SCI12	Timer Counter Register	TCNT	8	8		
0008 C000h	PORT0	Port Direction Register	PDR	8	8		
0008 C001h	PORT1	Port Direction Register	PDR	8	8		
0008 C002h	PORT2	Port Direction Register	PDR	8	8	2 or 3 PCLKB	section 17.
0008 C003h	PORT3	Port Direction Register	PDR	8	8		
0008 C004h	PORT4	Port Direction Register	PDR	8	8		
0008 C007h	PORT7	Port Direction Register	PDR	8	8		
0008 C009h	PORT9	Port Direction Register	PDR	8	8		
0008 C00Bh	PORTB	Port Direction Register	PDR	8	8		
0008 C00Dh	PORTD	Port Direction Register	PDR	8	8		
0008 C010h	PORTG	Port Direction Register	PDR	8	8		
0008 C020h	PORT0	Port Output Data Register	PODR	8	8		
0008 C021h	PORT1	Port Output Data Register	PODR	8	8		
0008 C022h	PORT2	Port Output Data Register	PODR	8	8		
0008 C023h	PORT3	Port Output Data Register	PODR	8	8		
0008 C024h	PORT4	Port Output Data Register	PODR	8	8		
0008 C027h	PORT7	Port Output Data Register	PODR	8	8		
0008 C029h	PORT9	Port Output Data Register	PODR	8	8		
0008 C02Bh	PORTB	Port Output Data Register	PODR	8	8		
0008 C02Dh	PORTD	Port Output Data Register	PODR	8	8		
0008 C030h	PORTG	Port Output Data Register	PODR	8	8		

Table 5.1 List of I/O Registers (Address Order) (8 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 C040h	PORT0	Port Input Data Register	PIDR	8	8	3 or 4 PCLKB	section 17.
0008 C041h	PORT1	Port Input Data Register	PIDR	8	8		
0008 C042h	PORT2	Port Input Data Register	PIDR	8	8		
0008 C043h	PORT3	Port Input Data Register	PIDR	8	8		
0008 C044h	PORT4	Port Input Data Register	PIDR	8	8		
0008 C047h	PORT7	Port Input Data Register	PIDR	8	8		
0008 C049h	PORT9	Port Input Data Register	PIDR	8	8		
0008 C04Bh	PORTB	Port Input Data Register	PIDR	8	8		
0008 C04Dh	PORTD	Port Input Data Register	PIDR	8	8		
0008 C04Eh	PORTE	Port Input Data Register	PIDR	8	8		
0008 C050h	PORTG	Port Input Data Register	PIDR	8	8	2 or 3 PCLKB	
0008 C060h	PORT0	Port Mode Register	PMR	8	8		
0008 C061h	PORT1	Port Mode Register	PMR	8	8		
0008 C062h	PORT2	Port Mode Register	PMR	8	8		
0008 C063h	PORT3	Port Mode Register	PMR	8	8		
0008 C064h	PORT4	Port Mode Register	PMR	8	8		
0008 C067h	PORT7	Port Mode Register	PMR	8	8		
0008 C069h	PORT9	Port Mode Register	PMR	8	8		
0008 C06Bh	PORTB	Port Mode Register	PMR	8	8		
0008 C06Dh	PORTD	Port Mode Register	PMR	8	8		
0008 C06Eh	PORTE	Port Mode Register	PMR	8	8		
0008 C070h	PORTG	Port Mode Register	PMR	8	8		
0008 C080h	PORT0	Open Drain Control Register 0	ODR0	8	8		
0008 C081h	PORT0	Open Drain Control Register 1	ODR1	8	8		
0008 C082h	PORT1	Open Drain Control Register 0	ODR0	8	8		
0008 C084h	PORT2	Open Drain Control Register 0	ODR0	8	8		
0008 C085h	PORT2	Open Drain Control Register 1	ODR1	8	8		
0008 C086h	PORT3	Open Drain Control Register 0	ODR0	8	8		
0008 C087h	PORT3	Open Drain Control Register 1	ODR1	8	8		
0008 C08Eh	PORT7	Open Drain Control Register 0	ODR0	8	8		
0008 C08Fh	PORT7	Open Drain Control Register 1	ODR1	8	8		
0008 C092h	PORT9	Open Drain Control Register 0	ODR0	8	8		
0008 C093h	PORT9	Open Drain Control Register 1	ODR1	8	8		
0008 C096h	PORTB	Open Drain Control Register 0	ODR0	8	8		
0008 C097h	PORTB	Open Drain Control Register 1	ODR1	8	8		
0008 C09Ah	PORTD	Open Drain Control Register 0	ODR0	8	8		
0008 C09Bh	PORTD	Open Drain Control Register 1	ODR1	8	8		
0008 C0A1h	PORTG	Open Drain Control Register 1	ODR1	8	8		
0008 C0C0h	PORT0	Pull-Up Control Register	PCR	8	8		
0008 C0C1h	PORT1	Pull-Up Control Register	PCR	8	8		
0008 C0C2h	PORT2	Pull-Up Control Register	PCR	8	8		
0008 C0C3h	PORT3	Pull-Up Control Register	PCR	8	8		
0008 C0C4h	PORT4	Pull-Up Control Register	PCR	8	8		
0008 C0C7h	PORT7	Pull-Up Control Register	PCR	8	8		
0008 C0C9h	PORT9	Pull-Up Control Register	PCR	8	8		
0008 C0CBh	PORTB	Pull-Up Control Register	PCR	8	8		
0008 C0CDh	PORTD	Pull-Up Control Register	PCR	8	8		
0008 C0D0h	PORTG	Pull-Up Control Register	PCR	8	8		
0008 C11Fh	MPC	Write-Protect Register	PWPR	8	8	2 or 3 PCLKB	section 18.
0008 C140h	MPC	P00 Pin Function Control Register	P00PFS	8	8		
0008 C141h	MPC	P01 Pin Function Control Register	P01PFS	8	8		
0008 C142h	MPC	P02 Pin Function Control Register	P02PFS	8	8		

Table 5.1 List of I/O Registers (Address Order) (9 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 C143h	MPC	P03 Pin Function Control Register	P03PFS	8	8	2 or 3 PCLKB	section 18.
0008 C144h	MPC	P04 Pin Function Control Register	P04PFS	8	8		
0008 C148h	MPC	P10 Pin Function Control Register	P10PFS	8	8		
0008 C149h	MPC	P11 Pin Function Control Register	P11PFS	8	8		
0008 C14Ah	MPC	P12 Pin Function Control Register	P12PFS	8	8		
0008 C14Bh	MPC	P13 Pin Function Control Register	P13PFS	8	8		
0008 C14Ch	MPC	P14 Pin Function Control Register	P14PFS	8	8		
0008 C14Dh	MPC	P15 Pin Function Control Register	P15PFS	8	8		
0008 C152h	MPC	P22 Pin Function Control Register	P22PFS	8	8		
0008 C153h	MPC	P23 Pin Function Control Register	P23PFS	8	8		
0008 C154h	MPC	P24 Pin Function Control Register	P24PFS	8	8		
0008 C155h	MPC	P25 Pin Function Control Register	P25PFS	8	8		
0008 C156h	MPC	P26 Pin Function Control Register	P26PFS	8	8		
0008 C158h	MPC	P30 Pin Function Control Register	P30PFS	8	8		
0008 C159h	MPC	P31 Pin Function Control Register	P31PFS	8	8		
0008 C15Ah	MPC	P32 Pin Function Control Register	P32PFS	8	8		
0008 C15Bh	MPC	P33 Pin Function Control Register	P33PFS	8	8		
0008 C15Eh	MPC	P36 Pin Function Control Register	P36PFS	8	8		
0008 C15Fh	MPC	P37 Pin Function Control Register	P37PFS	8	8		
0008 C160h	MPC	P40 Pin Function Control Register	P40PFS	8	8		
0008 C161h	MPC	P41 Pin Function Control Register	P41PFS	8	8		
0008 C162h	MPC	P42 Pin Function Control Register	P42PFS	8	8		
0008 C163h	MPC	P43 Pin Function Control Register	P43PFS	8	8		
0008 C164h	MPC	P44 Pin Function Control Register	P44PFS	8	8		
0008 C165h	MPC	P45 Pin Function Control Register	P45PFS	8	8		
0008 C166h	MPC	P46 Pin Function Control Register	P46PFS	8	8		
0008 C167h	MPC	P47 Pin Function Control Register	P47PFS	8	8		
0008 C178h	MPC	P70 Pin Function Control Register	P70PFS	8	8		
0008 C179h	MPC	P71 Pin Function Control Register	P71PFS	8	8		
0008 C17Ah	MPC	P72 Pin Function Control Register	P72PFS	8	8		
0008 C17Bh	MPC	P73 Pin Function Control Register	P73PFS	8	8		
0008 C17Ch	MPC	P74 Pin Function Control Register	P74PFS	8	8		
0008 C17Dh	MPC	P75 Pin Function Control Register	P75PFS	8	8		
0008 C17Eh	MPC	P76 Pin Function Control Register	P76PFS	8	8		
0008 C188h	MPC	P90 Pin Function Control Register	P90PFS	8	8		
0008 C189h	MPC	P91 Pin Function Control Register	P91PFS	8	8		
0008 C18Ah	MPC	P92 Pin Function Control Register	P92PFS	8	8		
0008 C18Bh	MPC	P93 Pin Function Control Register	P93PFS	8	8		
0008 C18Ch	MPC	P94 Pin Function Control Register	P94PFS	8	8		
0008 C18Dh	MPC	P95 Pin Function Control Register	P95PFS	8	8		
0008 C18Eh	MPC	P96 Pin Function Control Register	P96PFS	8	8		
0008 C18Fh	MPC	P97 Pin Function Control Register	P97PFS	8	8		
0008 C198h	MPC	PB0 Pin Function Control Register	PB0PFS	8	8		
0008 C199h	MPC	PB1 Pin Function Control Register	PB1PFS	8	8		
0008 C19Ah	MPC	PB2 Pin Function Control Register	PB2PFS	8	8		
0008 C19Bh	MPC	PB3 Pin Function Control Register	PB3PFS	8	8		
0008 C19Ch	MPC	PB4 Pin Function Control Register	PB4PFS	8	8		
0008 C19Dh	MPC	PB5 Pin Function Control Register	PB5PFS	8	8		
0008 C19Eh	MPC	PB6 Pin Function Control Register	PB6PFS	8	8		
0008 C19Fh	MPC	PB7 Pin Function Control Register	PB7PFS	8	8		
0008 C1ABh	MPC	PD3 Pin Function Control Register	PD3PFS	8	8		
0008 C1ACh	MPC	PD4 Pin Function Control Register	PD4PFS	8	8		

Table 5.1 List of I/O Registers (Address Order) (10 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0008 C1ADh	MPC	PD5 Pin Function Control Register	PD5PFS	8	8	2 or 3 PCLKB	section 18.
0008 C1AEh	MPC	PD6 Pin Function Control Register	PD6PFS	8	8		
0008 C1AFh	MPC	PD7 Pin Function Control Register	PD7PFS	8	8		
0008 C1B2h	MPC	PE2 Pin Function Control Register	PE2PFS	8	8		
0008 C1C7h	MPC	PG7 Pin Function Control Register	PG7PFS	8	8		
0008 C290h	SYSTEM	Reset Status Register 0	RSTSR0	8	8	4 or 5 PCLKB	section 6.
0008 C291h	SYSTEM	Reset Status Register 1	RSTSR1	8	8		
0008 C293h	SYSTEM	Main Clock Oscillator Forced Oscillation Control Register	MOFCR	8	8	4 or 5 PCLKB	section 9.
0008 C297h	SYSTEM	Voltage Monitoring Circuit Control Register	LVCMPCCR	8	8	4 or 5 PCLKB	section 8.
0008 C298h	SYSTEM	Voltage Detection Level Select Register	LVDLVL	8	8		
0008 C29Ah	SYSTEM	Voltage Monitoring 1 Circuit Control Register 0	LVD1CR0	8	8		
0008 C29Bh	SYSTEM	Voltage Monitoring 2 Circuit Control Register 0	LVD2CR0	8	8		
0009 5200h	MTU3	Timer Control Register	TCR	8	8, 16, 32	4 or 5 PCLKB	section 19.
0009 5201h	MTU4	Timer Control Register	TCR	8	8		
0009 5202h	MTU3	Timer Mode Register 1	TMDR1	8	8, 16		
0009 5203h	MTU4	Timer Mode Register 1	TMDR1	8	8		
0009 5204h	MTU3	Timer I/O Control Register H	TIORH	8	8, 16, 32		
0009 5205h	MTU3	Timer I/O Control Register L	TIORL	8	8		
0009 5206h	MTU4	Timer I/O Control Register H	TIORH	8	8, 16		
0009 5207h	MTU4	Timer I/O Control Register L	TIORL	8	8		
0009 5208h	MTU3	Timer Interrupt Enable Register	TIER	8	8, 16		
0009 5209h	MTU4	Timer Interrupt Enable Register	TIER	8	8		
0009 520Ah	MTU	Timer Output Master Enable Register A	TOERA	8	8		
0009 520Dh	MTU	Timer Gate Control Register A	TGCRA	8	8		
0009 520Eh	MTU	Timer Output Control Register 1A	TOCR1A	8	8, 16		
0009 520Fh	MTU	Timer Output Control Register 2A	TOCR2A	8	8		
0009 5210h	MTU3	Timer Counter	TCNT	16	16, 32		
0009 5212h	MTU4	Timer Counter	TCNT	16	16		
0009 5214h	MTU	Timer Period Data Register A	TCORA	16	16, 32		
0009 5216h	MTU	Timer Dead Time Data Register A	TDDRA	16	16		
0009 5218h	MTU3	Timer General Register A	TGRA	16	16, 32		
0009 521Ah	MTU3	Timer General Register B	TGRB	16	16		
0009 521Ch	MTU4	Timer General Register A	TGRA	16	16, 32		
0009 521Eh	MTU4	Timer General Register B	TGRB	16	16		
0009 5220h	MTU	Timer Subcounter A	TCNTSA	16	16, 32		
0009 5222h	MTU	Timer Period Buffer Register A	TCBRA	16	16		
0009 5224h	MTU3	Timer General Register C	TGRC	16	16, 32		
0009 5226h	MTU3	Timer General Register D	TGRD	16	16		
0009 5228h	MTU4	Timer General Register C	TGRC	16	16, 32		
0009 522Ah	MTU4	Timer General Register D	TGRD	16	16		
0009 522Ch	MTU3	Timer Status Register	TSR	8	8, 16		
0009 522Dh	MTU4	Timer Status Register	TSR	8	8		
0009 5230h	MTU	Timer Interrupt Skipping Set Register 1A	TITCR1A	8	8, 16		
0009 5231h	MTU	Timer Interrupt Skipping Counter 1A	TITCNT1A	8	8		
0009 5232h	MTU	Timer Buffer Transfer Set Register A	TBTERA	8	8		
0009 5234h	MTU	Timer Dead Time Enable Register A	TDERA	8	8		
0009 5236h	MTU	Timer Output Level Buffer Register A	TOLBRA	8	8		
0009 5238h	MTU3	Timer Buffer Operation Transfer Mode Register	TBTM	8	8, 16		
0009 5239h	MTU4	Timer Buffer Operation Transfer Mode Register	TBTM	8	8		
0009 523Ah	MTU	Timer Interrupt Skipping Mode Register A	TITMRA	8	8		
0009 523Bh	MTU	Timer Interrupt Skipping Set Register 2A	TITCR2A	8	8		
0009 523Ch	MTU	Timer Interrupt Skipping Counter 2A	TITCNT2A	8	8		

Table 5.1 List of I/O Registers (Address Order) (11 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0009 5240h	MTU4	Timer A/D Conversion Start Request Control Register	TADCR	16	16	4 or 5 PCLKB	section 19.
0009 5244h	MTU4	Timer A/D Conversion Start Request Cycle Set Register A	TADCORA	16	16, 32		
0009 5246h	MTU4	Timer A/D Conversion Start Request Cycle Set Register B	TADCORB	16	16		
0009 5248h	MTU4	Timer A/D Conversion Start Request Cycle Set Buffer Register A	TADCOBRA	16	16, 32		
0009 524Ah	MTU4	Timer A/D Conversion Start Request Cycle Set Buffer Register B	TADCOBRB	16	16		
0009 524Ch	MTU3	Timer Control Register 2	TCR2	8	8		
0009 524Dh	MTU4	Timer Control Register 2	TCR2	8	8		
0009 5260h	MTU	Timer Waveform Control Register A	TWCRA	8	8		
0009 5270h	MTU	Timer Mode Register 2A	TMDR2A	8	8		
0009 5272h	MTU3	Timer General Register E	TGRE	16	16		
0009 5274h	MTU4	Timer General Register E	TGRE	16	16		
0009 5276h	MTU4	Timer General Register F	TGRF	16	16		
0009 5280h	MTU	Timer Start Register A	TSTRA	8	8, 16		
0009 5281h	MTU	Timer Synchronous Register A	TSYRA	8	8		
0009 5282h	MTU	Timer Counter Synchronous Start Register	TCSYSTR	8	8		
0009 5284h	MTU	Timer Read/Write Enable Register A	TRWERA	8	8		
0009 5290h	MTU0	Noise Filter Control Register 0	NFCR0	8	8		
0009 5291h	MTU1	Noise Filter Control Register 1	NFCR1	8	8		
0009 5292h	MTU2	Noise Filter Control Register 2	NFCR2	8	8		
0009 5293h	MTU3	Noise Filter Control Register 3	NFCR3	8	8		
0009 5294h	MTU4	Noise Filter Control Register 4	NFCR4	8	8		
0009 5295h	MTU5	Noise Filter Control Register 5	NFCR5	8	8		
0009 5299h	MTU0	Noise Filter Control Register C	NFCRC	8	8		
0009 5300h	MTU0	Timer Control Register	TCR	8	8, 16, 32		
0009 5301h	MTU0	Timer Mode Register 1	TMDR1	8	8		
0009 5302h	MTU0	Timer I/O Control Register H	TIORH	8	8, 16		
0009 5303h	MTU0	Timer I/O Control Register L	TIORL	8	8		
0009 5304h	MTU0	Timer Interrupt Enable Register	TIER	8	8, 16, 32		
0009 5306h	MTU0	Timer Counter	TCNT	16	16		
0009 5308h	MTU0	Timer General Register A	TGRA	16	16, 32		
0009 530Ah	MTU0	Timer General Register B	TGRB	16	16		
0009 530Ch	MTU0	Timer General Register C	TGRC	16	16, 32		
0009 530Eh	MTU0	Timer General Register D	TGRD	16	16		
0009 5320h	MTU0	Timer General Register E	TGRE	16	16, 32		
0009 5322h	MTU0	Timer General Register F	TGRF	16	16		
0009 5324h	MTU0	Timer Interrupt Enable Register 2	TIER2	8	8, 16		
0009 5326h	MTU0	Timer Buffer Operation Transfer Mode Register	TBTM	8	8		
0009 5328h	MTU0	Timer Control Register 2	TCR2	8	8		
0009 5380h	MTU1	Timer Control Register	TCR	8	8, 16		
0009 5381h	MTU1	Timer Mode Register 1	TMDR1	8	8		
0009 5382h	MTU1	Timer I/O Control Register	TIOR	8	8		
0009 5384h	MTU1	Timer Interrupt Enable Register	TIER	8	8, 16, 32		
0009 5385h	MTU1	Timer Status Register	TSR	8	8		
0009 5386h	MTU1	Timer Counter	TCNT	16	16		
0009 5388h	MTU1	Timer General Register A	TGRA	16	16, 32		
0009 538Ah	MTU1	Timer General Register B	TGRB	16	16		
0009 5390h	MTU1	Timer Input Capture Control Register	TICCR	8	8		
0009 5391h	MTU1	Timer Mode Register 3	TMDR3	8	8		
0009 5394h	MTU1	Timer Control Register 2	TCR2	8	8		
0009 53A0h	MTU1	Timer Longword Counter	TCNTLW	32	32		
0009 53A4h	MTU1	Timer Longword General Register A	TGRALW	32	32		
0009 53A8h	MTU1	Timer Longword General Register B	TGRBLW	32	32		

Table 5.1 List of I/O Registers (Address Order) (12 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0009 5400h	MTU2	Timer Control Register	TCR	8	8, 16	4 or 5 PCLKB	section 19.
0009 5401h	MTU2	Timer Mode Register 1	TMDR1	8	8		
0009 5402h	MTU2	Timer I/O Control Register	TIOR	8	8		
0009 5404h	MTU2	Timer Interrupt Enable Register	TIER	8	8, 16, 32		
0009 5405h	MTU2	Timer Status Register	TSR	8	8		
0009 5406h	MTU2	Timer Counter	TCNT	16	16		
0009 5408h	MTU2	Timer General Register A	TGRA	16	16, 32		
0009 540Ah	MTU2	Timer General Register B	TGRB	16	16		
0009 540Ch	MTU2	Timer Control Register 2	TCR2	8	8		
0009 5480h	MTU5	Timer Counter U	TCNTU	16	16, 32		
0009 5482h	MTU5	Timer General Register U	TGRU	16	16		
0009 5484h	MTU5	Timer Control Register U	TCRU	8	8		
0009 5485h	MTU5	Timer Control Register 2U	TCR2U	8	8		
0009 5486h	MTU5	Timer I/O Control Register U	TIORU	8	8		
0009 5490h	MTU5	Timer Counter V	TCNTV	16	16, 32		
0009 5492h	MTU5	Timer General Register V	TGRV	16	16		
0009 5494h	MTU5	Timer Control Register V	TCRV	8	8		
0009 5495h	MTU5	Timer Control Register 2V	TCR2V	8	8		
0009 5496h	MTU5	Timer I/O Control Register V	TIORV	8	8		
0009 54A0h	MTU5	Timer Counter W	TCNTW	16	16, 32		
0009 54A2h	MTU5	Timer General Register W	TGRW	16	16		
0009 54A4h	MTU5	Timer Control Register W	TCRW	8	8		
0009 54A5h	MTU5	Timer Control Register 2W	TCR2W	8	8		
0009 54A6h	MTU5	Timer I/O Control Register W	TIORW	8	8		
0009 54B2h	MTU5	Timer Interrupt Enable Register	TIER	8	8		
0009 54B4h	MTU5	Timer Start Register	TSTR	8	8		
0009 54B6h	MTU5	Timer Compare Match Clear Register	TCNTCMPCLR	8	8		
0009 5D30h	MTU	A/D Conversion Start Request Select Register 0	TADSTRGR0	8	8	4 or 5 PCLKB	section 21.
0009 5D32h	MTU	A/D Conversion Start Request Select Register 1	TADSTRGR1	8	8		
0009 6000h	GPTW0	General PWM Timer Write-Protection Register	GTWP	32	32		
0009 6004h	GPTW0	General PWM Timer Software Start Register	GTSTR	32	32		
0009 6008h	GPTW0	General PWM Timer Software Stop Register	GTSTP	32	32		
0009 600Ch	GPTW0	General PWM Timer Software Clear Register	GTCLR	32	32		
0009 6010h	GPTW0	General PWM Timer Start Source Select Register	GTSSR	32	32		
0009 6014h	GPTW0	General PWM Timer Stop Source Select Register	GTCSR	32	32		
0009 6018h	GPTW0	General PWM Timer Clear Source Select Register	GTCSR	32	32		
0009 6024h	GPTW0	General PWM Timer Input Capture Source Select Register A	GTICASR	32	32		
0009 6028h	GPTW0	General PWM Timer Input Capture Source Select Register B	GTICBSR	32	32		
0009 602Ch	GPTW0	General PWM Timer Control Register	GTCCR	32	32		
0009 6030h	GPTW0	General PWM Timer Count Direction and Duty Setting Register	GTUDDTYC	32	32		
0009 6034h	GPTW0	General PWM Timer I/O Control Register	GTIOR	32	32		
0009 6038h	GPTW0	General PWM Timer Interrupt Output Setting Register	GTINTAD	32	32		
0009 603Ch	GPTW0	General PWM Timer Status Register	GTST	32	32		
0009 6040h	GPTW0	General PWM Timer Buffer Enable Register	GTBER	32	32		
0009 6044h	GPTW0	General PWM Timer Interrupt and A/D Conversion Start Request Skipping Setting Register	GTITC	32	32		
0009 6048h	GPTW0	General PWM Timer Counter	GTCNT	32	32		
0009 604Ch	GPTW0	General PWM Timer Compare Capture Register A	GTCCRA	32	32		
0009 6050h	GPTW0	General PWM Timer Compare Capture Register B	GTCCRB	32	32		
0009 6054h	GPTW0	General PWM Timer Compare Capture Register C	GTCCRC	32	32		
0009 6058h	GPTW0	General PWM Timer Compare Capture Register E	GTCCRE	32	32		
0009 605Ch	GPTW0	General PWM Timer Compare Capture Register D	GTCCRD	32	32		
0009 6060h	GPTW0	General PWM Timer Compare Capture Register F	GTCCRF	32	32		

Table 5.1 List of I/O Registers (Address Order) (13 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0009 6064h	GPTW0	General PWM Timer Period Setting Register	GTPR	32	32	4 or 5 PCLKB	section 21.
0009 6068h	GPTW0	General PWM Timer Period Setting Buffer Register	GTPBR	32	32		
0009 606Ch	GPTW0	General PWM Timer Period Setting Double-Buffer Register	GTPDBR	32	32		
0009 6070h	GPTW0	A/D Conversion Start Request Timing Register A	GTADTRA	32	32		
0009 6074h	GPTW0	A/D Conversion Start Request Timing Buffer Register A	GTADTBRA	32	32		
0009 6078h	GPTW0	A/D Conversion Start Request Timing Double-Buffer Register A	GTADTDBRA	32	32		
0009 607Ch	GPTW0	A/D Conversion Start Request Timing Register B	GTADTRB	32	32		
0009 6080h	GPTW0	A/D Conversion Start Request Timing Buffer Register B	GTADTBRB	32	32		
0009 6084h	GPTW0	A/D Conversion Start Request Timing Double-Buffer Register B	GTADTDBRB	32	32		
0009 6088h	GPTW0	General PWM Timer Dead Time Control Register	GTDTCR	32	32		
0009 608Ch	GPTW0	General PWM Timer Dead Time Value Register U	GTDVU	32	32		
0009 6090h	GPTW0	General PWM Timer Dead Time Value Register D	GTDVD	32	32		
0009 6094h	GPTW0	General PWM Timer Dead Time Value Buffer Register U	GTDBU	32	32		
0009 6098h	GPTW0	General PWM Timer Dead Time Value Buffer Register D	GTDBD	32	32		
0009 609Ch	GPTW0	General PWM Timer Output Protection Function Status Register	GTSOS	32	32		
0009 60A0h	GPTW0	General PWM Timer Output Protection Function Temporary Release Register	GTSOTR	32	32		
0009 60A4h	GPTW0	General PWM Timer A/D Conversion Start Request Signal Monitoring Register	GTADSMR	32	32		
0009 60D0h	GPTW0	General PWM Timer Operation Enable Bit Simultaneous Control Channel Select Register	GTSECSR	32	32		
0009 60D4h	GPTW0	General PWM Timer Operation Enable Bit Simultaneous Control Register	GTSECR	32	32		
0009 60E0h	GPTW0	General PWM Timer Buffer Enable Register 2	GTBER2	32	32		
0009 60ECh	GPTW0	General PWM Timer Inter Channel Cooperation Input Capture Control Register	GTICCR	32	32		
0009 6100h	GPTW1	General PWM Timer Write-Protection Register	GTWP	32	32		
0009 6104h	GPTW1	General PWM Timer Software Start Register	GTSTR	32	32		
0009 6108h	GPTW1	General PWM Timer Software Stop Register	GTSTP	32	32		
0009 610Ch	GPTW1	General PWM Timer Software Clear Register	GTCLR	32	32		
0009 6110h	GPTW1	General PWM Timer Start Source Select Register	GTSSR	32	32		
0009 6114h	GPTW1	General PWM Timer Stop Source Select Register	GTSPSR	32	32		
0009 6118h	GPTW1	General PWM Timer Clear Source Select Register	GTCSR	32	32		
0009 6124h	GPTW1	General PWM Timer Input Capture Source Select Register A	GTICASR	32	32		
0009 6128h	GPTW1	General PWM Timer Input Capture Source Select Register B	GTICBSR	32	32		
0009 612Ch	GPTW1	General PWM Timer Control Register	GTCR	32	32		
0009 6130h	GPTW1	General PWM Timer Count Direction and Duty Setting Register	GTUDDTYC	32	32		
0009 6134h	GPTW1	General PWM Timer I/O Control Register	GTIOR	32	32		
0009 6138h	GPTW1	General PWM Timer Interrupt Output Setting Register	GTINTAD	32	32		
0009 613Ch	GPTW1	General PWM Timer Status Register	GTST	32	32		
0009 6140h	GPTW1	General PWM Timer Buffer Enable Register	GTBER	32	32		
0009 6144h	GPTW1	General PWM Timer Interrupt and A/D Conversion Start Request Skipping Setting Register	GTITC	32	32		
0009 6148h	GPTW1	General PWM Timer Counter	GTCNT	32	32		
0009 614Ch	GPTW1	General PWM Timer Compare Capture Register A	GTCCRA	32	32		
0009 6150h	GPTW1	General PWM Timer Compare Capture Register B	GTCCRB	32	32		
0009 6154h	GPTW1	General PWM Timer Compare Capture Register C	GTCCRC	32	32		
0009 6158h	GPTW1	General PWM Timer Compare Capture Register E	GTCCRE	32	32		
0009 615Ch	GPTW1	General PWM Timer Compare Capture Register D	GTCCRD	32	32		
0009 6160h	GPTW1	General PWM Timer Compare Capture Register F	GTCCRF	32	32		
0009 6164h	GPTW1	General PWM Timer Period Setting Register	GTPR	32	32		
0009 6168h	GPTW1	General PWM Timer Period Setting Buffer Register	GTPBR	32	32		
0009 616Ch	GPTW1	General PWM Timer Period Setting Double-Buffer Register	GTPDBR	32	32		
0009 6170h	GPTW1	A/D Conversion Start Request Timing Register A	GTADTRA	32	32		
0009 6174h	GPTW1	A/D Conversion Start Request Timing Buffer Register A	GTADTBRA	32	32		

Table 5.1 List of I/O Registers (Address Order) (14 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0009 6178h	GPTW1	A/D Conversion Start Request Timing Double-Buffer Register A	GTADTDBRA	32	32	4 or 5 PCLKB	section 21.
0009 617Ch	GPTW1	A/D Conversion Start Request Timing Register B	GTADTRB	32	32		
0009 6180h	GPTW1	A/D Conversion Start Request Timing Buffer Register B	GTADTBRB	32	32		
0009 6184h	GPTW1	A/D Conversion Start Request Timing Double-Buffer Register B	GTADTDBRB	32	32		
0009 6188h	GPTW1	General PWM Timer Dead Time Control Register	GTDTCCR	32	32		
0009 618Ch	GPTW1	General PWM Timer Dead Time Value Register U	GTDTVU	32	32		
0009 6190h	GPTW1	General PWM Timer Dead Time Value Register D	GTDVD	32	32		
0009 6194h	GPTW1	General PWM Timer Dead Time Value Buffer Register U	GTDBU	32	32		
0009 6198h	GPTW1	General PWM Timer Dead Time Value Buffer Register D	GTDBD	32	32		
0009 619Ch	GPTW1	General PWM Timer Output Protection Function Status Register	GTSOS	32	32		
0009 61A0h	GPTW1	General PWM Timer Output Protection Function Temporary Release Register	GTSOTR	32	32		
0009 61A4h	GPTW1	General PWM Timer A/D Conversion Start Request Signal Monitoring Register	GTADSMR	32	32		
0009 61D0h	GPTW1	General PWM Timer Operation Enable Bit Simultaneous Control Channel Select Register	GTSECSR	32	32		
0009 61D4h	GPTW1	General PWM Timer Operation Enable Bit Simultaneous Control Register	GTSECR	32	32		
0009 61E0h	GPTW1	General PWM Timer Buffer Enable Register 2	GTBER2	32	32		
0009 61ECh	GPTW1	General PWM Timer Inter Channel Cooperation Input Capture Control Register	GTICCR	32	32		
0009 6200h	GPTW2	General PWM Timer Write-Protection Register	GTWP	32	32		
0009 6204h	GPTW2	General PWM Timer Software Start Register	GTSTR	32	32		
0009 6208h	GPTW2	General PWM Timer Software Stop Register	GTSTP	32	32		
0009 620Ch	GPTW2	General PWM Timer Software Clear Register	GTCLR	32	32		
0009 6210h	GPTW2	General PWM Timer Start Source Select Register	GTSSR	32	32		
0009 6214h	GPTW2	General PWM Timer Stop Source Select Register	GTPSR	32	32		
0009 6218h	GPTW2	General PWM Timer Clear Source Select Register	GTCSR	32	32		
0009 6224h	GPTW2	General PWM Timer Input Capture Source Select Register A	GTICASR	32	32		
0009 6228h	GPTW2	General PWM Timer Input Capture Source Select Register B	GTICBSR	32	32		
0009 622Ch	GPTW2	General PWM Timer Control Register	GTCR	32	32		
0009 6230h	GPTW2	General PWM Timer Count Direction and Duty Setting Register	GTUDDTYC	32	32		
0009 6234h	GPTW2	General PWM Timer I/O Control Register	GTIOR	32	32		
0009 6238h	GPTW2	General PWM Timer Interrupt Output Setting Register	GTINTAD	32	32		
0009 623Ch	GPTW2	General PWM Timer Status Register	GTST	32	32		
0009 6240h	GPTW2	General PWM Timer Buffer Enable Register	GTBER	32	32		
0009 6244h	GPTW2	General PWM Timer Interrupt and A/D Conversion Start Request Skipping Setting Register	GTITC	32	32		
0009 6248h	GPTW2	General PWM Timer Counter	GTCNT	32	32		
0009 624Ch	GPTW2	General PWM Timer Compare Capture Register A	GTCCRA	32	32		
0009 6250h	GPTW2	General PWM Timer Compare Capture Register B	GTCCRB	32	32		
0009 6254h	GPTW2	General PWM Timer Compare Capture Register C	GTCCRC	32	32		
0009 6258h	GPTW2	General PWM Timer Compare Capture Register E	GTCCRE	32	32		
0009 625Ch	GPTW2	General PWM Timer Compare Capture Register D	GTCCRD	32	32		
0009 6260h	GPTW2	General PWM Timer Compare Capture Register F	GTCCRF	32	32		
0009 6264h	GPTW2	General PWM Timer Period Setting Register	GTPR	32	32		
0009 6268h	GPTW2	General PWM Timer Period Setting Buffer Register	GTPBR	32	32		
0009 626Ch	GPTW2	General PWM Timer Period Setting Double-Buffer Register	GTPDBR	32	32		
0009 6270h	GPTW2	A/D Conversion Start Request Timing Register A	GTADTRA	32	32		
0009 6274h	GPTW2	A/D Conversion Start Request Timing Buffer Register A	GTADTBRA	32	32		
0009 6278h	GPTW2	A/D Conversion Start Request Timing Double-Buffer Register A	GTADTDBRA	32	32		
0009 627Ch	GPTW2	A/D Conversion Start Request Timing Register B	GTADTRB	32	32		
0009 6280h	GPTW2	A/D Conversion Start Request Timing Buffer Register B	GTADTBRB	32	32		
0009 6284h	GPTW2	A/D Conversion Start Request Timing Double-Buffer Register B	GTADTDBRB	32	32		
0009 6288h	GPTW2	General PWM Timer Dead Time Control Register	GTDTCCR	32	32		

Table 5.1 List of I/O Registers (Address Order) (15 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0009 628Ch	GPTW2	General PWM Timer Dead Time Value Register U	GTDVU	32	32	4 or 5 PCLKB	section 21.
0009 6290h	GPTW2	General PWM Timer Dead Time Value Register D	GTDVD	32	32		
0009 6294h	GPTW2	General PWM Timer Dead Time Value Buffer Register U	GTDBU	32	32		
0009 6298h	GPTW2	General PWM Timer Dead Time Value Buffer Register D	GTDBD	32	32		
0009 629Ch	GPTW2	General PWM Timer Output Protection Function Status Register	GTSOS	32	32		
0009 62A0h	GPTW2	General PWM Timer Output Protection Function Temporary Release Register	GTSOTR	32	32		
0009 62A4h	GPTW2	General PWM Timer A/D Conversion Start Request Signal Monitoring Register	GTADSMR	32	32		
0009 62D0h	GPTW2	General PWM Timer Operation Enable Bit Simultaneous Control Channel Select Register	GTSECSR	32	32		
0009 62D4h	GPTW2	General PWM Timer Operation Enable Bit Simultaneous Control Register	GTSECR	32	32		
0009 62E0h	GPTW2	General PWM Timer Buffer Enable Register 2	GTBER2	32	32		
0009 62ECh	GPTW2	General PWM Timer Inter Channel Cooperation Input Capture Control Register	GTICCR	32	32		
0009 E000h	POEG	POEG Group A Setting Register	POEGGA	32	32	2 or 3 PCLKB	section 22.
0009 E004h	POEG	POEG Group A Input Control Register	POEGICRA	32	32		
0009 E040h	POEG	GPTW Output Disabling Control Group A Write Protection Register	GTONCWPA	16	16		
0009 E044h	POEG	GPTW Output Disabling Control Group A Controlling Register	GTONCCRA	16	16		
0009 E400h	POE	Input Level Control/Status Register 1	ICSR1	16	16	2 or 3 PCLKB	section 20.
0009 E402h	POE	Output Level Control/Status Register 1	OCSR1	16	16		
0009 E408h	POE	Input Level Control/Status Register 3	ICSR3	16	16		
0009 E40Bh	POE	Port Output Enable Control Register 1	POECR1	8	8		
0009 E40Ch	POE	Port Output Enable Control Register 2	POECR2	16	16		
0009 E40Eh	POE	Port Output Enable Control Register 3	POECR3	16	16		
0009 E410h	POE	Port Output Enable Control Register 4	POECR4	16	16		
0009 E412h	POE	Port Output Enable Control Register 5	POECR5	16	16		
0009 E414h	POE	Port Output Enable Control Register 6	POECR6	16	16		
0009 E416h	POE	Input Level Control/Status Register 4	ICSR4	16	16		
0009 E418h	POE	Input Level Control/Status Register 5	ICSR5	16	16		
0009 E41Ah	POE	Active Level Setting Register 1	ALR1	16	16	8	
0009 E41Ch	POE	Input Level Control/Status Register 6	ICSR6	16	16		
0009 E420h	POE	Input Level Control/Status Register 7	ICSR7	16	16		
0009 E426h	POE	Port Output Enable Comparator Output Detection Flag Register	POECMPFR	16	16		
0009 E428h	POE	Port Output Enable Comparator Request Select Register	POECMPSEL	16	16		
0009 E42Ah	POE	Output Level Control/Status Register 3	OCSR3	16	16		
0009 E42Ch	POE	Active Level Setting Register 3	ALR3	16	16		
0009 E42Eh	POE	Software Port Output Enable Register	SPOER	16	16		
0009 E430h	POE	Port Mode Mask Control Register 0	PMMCR0	16	16		
0009 E432h	POE	Port Mode Mask Control Register 1	PMMCR1	16	16		
0009 E434h	POE	Port Mode Mask Control Register 2	PMMCR2	16	16		
0009 E438h	POE	Port Output Enable Comparator Request Extended Selection Register 0	POECMPX0	8	8	8	
0009 E43Ah	POE	Port Output Enable Comparator Request Extended Selection Register 2	POECMPX2	8	8		
0009 E43Ch	POE	Port Output Enable Comparator Request Extended Selection Register 4	POECMPX4	8	8		
0009 E43Dh	POE	Port Output Enable Comparator Request Extended Selection Register 5	POECMPX5	8	8		
0009 E450h	POE	Port Output Enable Control Register 6B	POECR6B	16	16		
0009 E452h	POE	Port Output Enable Control Register 9	POECR9	16	16		
0009 E458h	POE	Port Output Enable Comparator Request Extended Selection Register 6	POECMPX6	8	8		
0009 E45Ch	POE	Input Signal Mask Control Register 0	IMCR0	8	8		
0009 E45Eh	POE	Input Signal Mask Control Register 2	IMCR2	8	8		
0009 E45Fh	POE	Input Signal Mask Control Register 3	IMCR3	8	8		

Table 5.1 List of I/O Registers (Address Order) (16 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
0009 E460h	POE	MTU0 Pin Select Register 1	M0SELR1	8	8	2 or 3 PCLKB	section 20.
0009 E461h	POE	MTU0 Pin Select Register 2	M0SELR2	8	8		
0009 E462h	POE	MTU3 Pin Select Register	M3SELR	8	8		
0009 E463h	POE	MTU4 Pin Select Register 1	M4SELR1	8	8		
0009 E464h	POE	MTU4 Pin Select Register 2	M4SELR2	8	8		
0009 E46Ah	POE	GPTW0 Pin Select Register	G0SELR	8	8		
0009 E46Bh	POE	GPTW1 Pin Select Register	G1SELR	8	8		
0009 E46Ch	POE	GPTW2 Pin Select Register	G2SELR	8	8		
0009 E474h	POE	Input Signal Mask Control Register 4	IMCR4	8	8		
0009 E475h	POE	Input Signal Mask Control Register 5	IMCR5	8	8		
0009 E479h	POE	Input Signal Mask Control Register 9	IMCR9	8	8		
0009 E47Ah	POE	Input Signal Mask Control Register 10	IMCR10	8	8	1 to 2 PCLKB	section 34.
0009 E47Bh	POE	Input Signal Mask Control Register 11	IMCR11	8	8		
000A 0C80h	CMPC0	Comparator Control Register	CMPCTL	8	8		
000A 0C84h	CMPC0	Comparator Input Select Register	CMPSEL0	8	8		
000A 0C88h	CMPC0	Comparator Reference Voltage Select Register	CMPSEL1	8	8		
000A 0C8Ch	CMPC0	Comparator Output Monitor Register	CMPMON	8	8		
000A 0C90h	CMPC0	Comparator External Output Enable Register	CMPIOC	8	8		
000A 0C98h	CMPC0	Comparator Control Register 2	CMPCTL2	8	8		
000A 0C9Ch	CMPC0	Noise Filter Setting Register	CMPNFS	8	8		
000A 0CA0h	CMPC1	Comparator Control Register	CMPCTL	8	8		
000A 0CA4h	CMPC1	Comparator Input Select Register	CMPSEL0	8	8		
000A 0CA8h	CMPC1	Comparator Reference Voltage Select Register	CMPSEL1	8	8		
000A 0CACH	CMPC1	Comparator Output Monitor Register	CMPMON	8	8		
000A 0CB0h	CMPC1	Comparator External Output Enable Register	CMPIOC	8	8		
000A 0CB8h	CMPC1	Comparator Control Register 2	CMPCTL2	8	8		
000A 0CBCCh	CMPC1	Noise Filter Setting Register	CMPNFS	8	8		
000A 0CC0h	CMPC2	Comparator Control Register	CMPCTL	8	8		
000A 0CC4h	CMPC2	Comparator Input Select Register	CMPSEL0	8	8		
000A 0CC8h	CMPC2	Comparator Reference Voltage Select Register	CMPSEL1	8	8		
000A 0CCCh	CMPC2	Comparator Output Monitor Register	CMPMON	8	8		
000A 0CD0h	CMPC2	Comparator External Output Enable Register	CMPIOC	8	8		
000A 0CD8h	CMPC2	Comparator Control Register 2	CMPCTL2	8	8		
000A 0CDCh	CMPC2	Noise Filter Setting Register	CMPNFS	8	8		
007F C090h	FLASH	E2 DataFlash Control Register	DFLCTL	8	8	2 or 3 FCLK	section 37.
007F C228h	TEMPS	Temperature Sensor Calibration Data Register	TSCDR	16	16		
007F C100h	FLASH	Flash P/E Mode Control Register	FPMCR	8	8	2 or 3 FCLK	section 37.
007F C104h	FLASH	Flash Area Select Register	FASR	8	8		
007F C108h	FLASH	Flash Processing Start Address Register L	FSARL	16	16		
007F C110h	FLASH	Flash Processing Start Address Register H	FSARH	16	16		
007F C114h	FLASH	Flash Control Register	FCR	8	8		
007F C118h	FLASH	Flash Processing End Address Register L	FEARL	16	16		
007F C120h	FLASH	Flash Processing End Address Register H	FEARH	16	16		
007F C124h	FLASH	Flash Reset Register	FRESETR	8	8		
007F C128h	FLASH	Flash Status Register 0	FSTATR0	8	8		
007F C12Ch	FLASH	Flash Status Register 1	FSTATR1	8	8		
007F C130h	FLASH	Flash Write Buffer Register 0	FWB0	16	16		
007F C138h	FLASH	Flash Write Buffer Register 1	FWB1	16	16		
007F C140h	FLASH	Flash Write Buffer Register 2	FWB2	16	16		
007F C144h	FLASH	Flash Write Buffer Register 3	FWB3	16	16		
007F C180h	FLASH	Protection Unlock Register	FPR	8	8		
007F C184h	FLASH	Protection Unlock Status Register	FPSR	8	8		

Table 5.1 List of I/O Registers (Address Order) (17 / 17)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size	Number of Access Cycles	Reference Section
007F C1C0h	FLASH	Flash Start-Up Setting Monitor Register	FSCMR	16	16	2 or 3 FCLK	section 37.
007F C1C8h	FLASH	Flash Access Window Start Address Monitor Register	FAWSMR	16	16		
007F C1D0h	FLASH	Flash Access Window End Address Monitor Register	FAWEMR	16	16		
007F C1D8h	FLASH	Flash Initial Setting Register	FISR	8	8		
007F C1DCh	FLASH	Flash Extra Area Control Register	FEXCR	8	8		
007F C1E0h	FLASH	Flash Error Address Monitor Register L	FEAML	16	16		
007F C1E8h	FLASH	Flash Error Address Monitor Register H	FEAMH	16	16		
007F C350h	FLASH	Unique ID Register 0	UIDR0	32	32		
007F C354h	FLASH	Unique ID Register 1	UIDR1	32	32		
007F C358h	FLASH	Unique ID Register 2	UIDR2	32	32		
007F C35Ch	FLASH	Unique ID Register 3	UIDR3	32	32		
007F FFB0h	FLASH	Flash P/E Mode Entry Register	FENTRYR	16	16	1 ICLK	section 7.
007F FFC0h	FLASH	Memory Wait Cycle Setting Register	MEMWAITR	16	16		
FFFF FF80h	OFSM	Endian Select Register	MDE	32	32		
FFFF FF88h	OFSM	Option Function Select Register 1	OFS1	32	32		
FFFF FF8Ch	OFSM	Option Function Select Register 0	OFS0	32	32		

6. Resets

6.1 Overview

There are seven types of resets: RES# pin reset, power-on reset, voltage monitoring 0 reset, voltage monitoring 1 reset, voltage monitoring 2 reset, independent watchdog timer reset, and software reset.

Table 6.1 lists the reset names and sources.

Table 6.1 Reset Names and Sources

Reset Name	Source
RES# pin reset	Voltage input to the RES# pin is driven low.
Power-on reset	VCC rises (voltage monitored: VPOR)* ¹
Voltage monitoring 0 reset	VCC falls (voltage monitored: Vdet0)* ¹
Voltage monitoring 1 reset	VCC falls (voltage monitored: Vdet1)* ¹
Voltage monitoring 2 reset	VCC falls (voltage monitored: Vdet2)* ¹
Independent watchdog timer reset	The independent watchdog timer underflows, or a refresh error occurs.
Software reset	Register setting

Note 1. For the voltages to be monitored (VPOR, Vdet0, Vdet1, and Vdet2), see section 8, Voltage Detection Circuit (LVDAb) and section 38, Electrical Characteristics.

The internal state and pins are initialized by a reset.

Table 6.2 lists the reset targets to be initialized.

Table 6.2 Targets for Initialization by Each Reset Source

Target to be Initialized	Reset Source						
	RES# Pin Reset	Power-On Reset	Voltage Monitoring 0 Reset	Independent Watchdog Timer Reset	Voltage Monitoring 1 Reset	Voltage Monitoring 2 Reset	Software Reset
The power-on reset detect flag (RSTSR0.PORF)	✓	—	—	—	—	—	—
Register related to the cold start/warm start determination flag (RSTSR1.CWSF)	—*1	✓	—	—	—	—	—
Voltage monitoring 0 reset detect flag (RSTSR0.LVD0RF)	✓	✓	—	—	—	—	—
The independent watchdog timer reset detect flag (RSTSR2.IWDTRF)	✓	✓	✓	—	—	—	—
Registers related to the independent watchdog timer (IWDTRR, IWDTCR, IWDTSR, IWDTRCR, IWDTCSTPR, ILOCOCR)	✓	✓	✓	—	—	—	—
The voltage monitoring 1 reset detect flag (RSTSR0.LVD1RF)	✓	✓	✓	✓	—	—	—
Registers related to voltage monitor function 1 (LVD1CR0, LVCMPCR.LVD1E, LVDLVL.LVD1LVL[3:0])	✓	✓	✓	✓	—	—	—
(LVD1CR1, LVD1SR)	✓	✓	✓	✓	—	—	—
The voltage monitoring 2 reset detect flag (RSTSR0.LVD2RF)	✓	✓	✓	✓	✓	—	—
Registers related to voltage monitor function 2 (LVD2CR0, LVCMPCR.LVD2E, LVDLVL.LVD2LVL[1:0])	✓	✓	✓	✓	✓	—	—
(LVD2CR1, LVD2SR)	✓	✓	✓	✓	✓	—	—
The software reset detect flag (RSTSR2.SWRF)	✓	✓	✓	✓	✓	✓	—
Operating mode*2	✓	✓	—	—	—	—	—
Registers other than the above, CPU, and internal state	✓	✓	✓	✓	✓	✓	✓

✓: Targets to be initialized, —: No change occurs.

Note 1. Initialized at a power-on.

Note 2. The operating mode is determined by the level of the mode setting pin when the reset is released. For details, refer to section 3, Operating Modes.

When a reset is canceled, the reset exception handling starts. For the reset exception handling, see section 13, Exception Handling.

Table 6.3 lists the pin related to the reset.

Table 6.3 Pin Related to Reset

Pin Name	I/O	Function
RES#	Input	Reset pin

6.2 Register Descriptions

6.2.1 Reset Status Register 0 (RSTSR0)

Address(es): SYSTEM.RSTSR0 0008 C290h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	LVD2R F	LVD1R F	LVD0R F	PORF
Value after reset:	0	0	0	0	0*1	0*1	0*1

Bit	Symbol	Bit Name	Description	R/W
b0	PORF	Power-On Reset Detect Flag	0: Power-on reset not detected. 1: Power-on reset detected.	R/(W) *2
b1	LVD0RF	Voltage Monitoring 0 Reset Detect Flag	0: Voltage monitoring 0 reset not detected. 1: Voltage monitoring 0 reset detected.	R/(W) *2
b2	LVD1RF	Voltage Monitoring 1 Reset Detect Flag	0: Voltage monitoring 1 reset not detected. 1: Voltage monitoring 1 reset detected.	R/(W) *2
b3	LVD2RF	Voltage Monitoring 2 Reset Detect Flag	0: Voltage monitoring 2 reset not detected. 1: Voltage monitoring 2 reset detected.	R/(W) *2
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The value after reset depends on the reset source.

Note 2. Only 0 can be written to clear the flag.

PORF Flag (Power-On Reset Detect Flag)

The PORF flag indicates that a power-on reset has occurred.

[Setting condition]

- When a power-on reset occurs.

[Clearing conditions]

- When resets shown in Table 6.2 occur.
- When PORF is read as 1 and then 0 is written to PORF.

LVD0RF Flag (Voltage Monitoring 0 Reset Detect Flag)

The LVD0RF flag indicates that VCC voltage has fallen below Vdet0.

[Setting condition]

- When Vdet0-level VCC voltage is detected.

[Clearing conditions]

- When resets listed in Table 6.2 occur.
- When LVD0RF is read as 1 and then 0 is written to LVD0RF.

LVD1RF Flag (Voltage Monitoring 1 Reset Detect Flag)

The LVD1RF flag indicates that VCC voltage has fallen below Vdet1.

[Setting condition]

- When Vdet1-level VCC voltage is detected.

[Clearing conditions]

- When a reset listed in Table 6.2 occurs.
- When LVD1RF is read as 1 and then 0 is written to LVD1RF.

LVD2RF Flag (Voltage Monitoring 2 Reset Detect Flag)

The LVD2RF flag indicates that VCC voltage has fallen below Vdet2.

[Setting condition]

- When Vdet2-level VCC voltage is detected.

[Clearing conditions]

- When a reset listed in Table 6.2 occurs.
- When LVD2RF is read as 1 and then 0 is written to LVD2RF.

6.2.2 Reset Status Register 1 (RSTSR1)

Address(es): SYSTEM.RSTSR1 0008 C291h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	CWSF
Value after reset:	0	0	0	0	0	0	0/1*1

Bit	Symbol	Bit Name	Description	R/W
b0	CWSF	Cold/Warm Start Determination Flag	0: Cold start 1: Warm start	R/(W) *2
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The value after reset depends on the reset source.

Note 2. Only 1 can be written to set the flag.

RSTSR1 determines whether a power-on reset has caused the reset processing (cold start) or a reset signal input during operation has caused the reset processing (warm start).

CWSF Flag (Cold/Warm Start Determination Flag)

The CWSF flag indicates the type of reset processing: cold start or warm start.

The CWSF flag is initialized at a power-on.

[Setting condition]

- When 1 is written through programming; it is not set to 0 even when 0 is written.

[Clearing condition]

- When a reset listed in Table 6.2 occurs.

6.2.3 Reset Status Register 2 (RSTSR2)

Address(es): SYSTEM.RSTSR2 0008 00C0h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SWRF	—	IWDTR F
Value after reset:	0	0	0	0	0	0*1	0	0*1

Bit	Symbol	Bit Name	Description	R/W
b0	IWDTRF	Independent Watchdog Timer Reset Detect Flag	0: Independent watchdog timer reset not detected. 1: Independent watchdog timer reset detected.	R/(W) *2
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	SWRF	Software Reset Detect Flag	0: Software reset not detected. 1: Software reset detected.	R/(W) *2
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The value after reset depends on the reset source.

Note 2. Only 0 can be written to clear the flag.

IWDTRF Flag (Independent Watchdog Timer Reset Detect Flag)

The IWDTRF flag indicates that an independent watchdog timer reset has occurred.

[Setting condition]

- When an independent watchdog timer reset occurs.

[Clearing conditions]

- When a reset listed in Table 6.2 occurs.
- When IWDTRF is read as 1 and then 0 is written to IWDTRF.

SWRF Flag (Software Reset Detect Flag)

The SWRF flag indicates that a software reset has occurred.

[Setting condition]

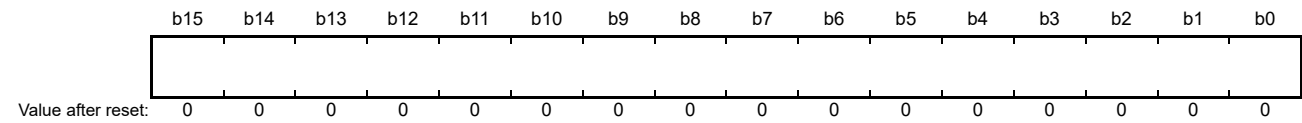
- When a software reset occurs.

[Clearing conditions]

- When a reset listed in Table 6.2 occurs.
- When SWRF is read as 1 and then 0 is written to SWRF.

6.2.4 Software Reset Register (SWRR)

Address(es): SYSTEM.SWRR 0008 00C2h



Writing A501h in the SWRR register resets the MCU. This register is read as 0000h. Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

6.3 Operation

6.3.1 RES# Pin Reset

This is a reset generated by the RES# pin.

When the RES# pin is driven low, all the processing in progress is aborted and the LSI enters a reset state.

In order to unfailingly reset the LSI, the RES# pin should be held low for the specified power supply stabilization time at a power-on.

When the RES# pin is driven high from low, the internal reset is canceled after the post-RES# cancellation wait time (tRESWT) has elapsed, and then the CPU starts the reset exception handling.

For details, see section 38, Electrical Characteristics.

6.3.2 Power-On Reset and Voltage Monitoring 0 Reset

The power-on reset is an internal reset generated by the power-on reset circuit. A power-on reset is generated when power is supplied to the RES# pin while it is connected to VCC via a resistor. When connecting a capacitor to the RES# pin, also ensure that the voltage on the RES# pin is always at least VIH. For details on VIH, refer to section 38, Electrical Characteristics. After VCC has exceeded VPOR and the specified period (power-on reset time) has elapsed, the internal reset is canceled and the CPU starts the reset exception handling. The power-on reset time is a stabilization period for the external power supply and the MCU circuit. After a power-on reset has been generated, the PORF flag in RSTSR0 is set to 1. The PORF flag is initialized by RES# pin reset.

The voltage monitoring 0 reset is an internal reset generated by the voltage monitoring circuit. If the voltage detection circuit 0 start bit (LVDAS) in option function select register 1 (OFS1) is 0 (voltage monitoring 0 reset is enabled after a reset) and VCC falls below Vdet0, the RSTSR0.LVD0RF flag becomes 1 and the voltage detection circuit generates voltage monitoring 0 reset. Clear the OFS1.LVDAS bit to 0 if the voltage monitoring 0 reset is to be used.

Release from the voltage monitoring 0 reset state occurs when VCC rises above Vdet0 and the LVD0 reset time (tLVD0) elapses, and then the CPU starts the reset exception handling.

Figure 6.1 shows operations during a power-on reset and voltage monitoring 0 reset.

For details on voltage monitoring 0 reset, refer to section 8, Voltage Detection Circuit (LVDAb).

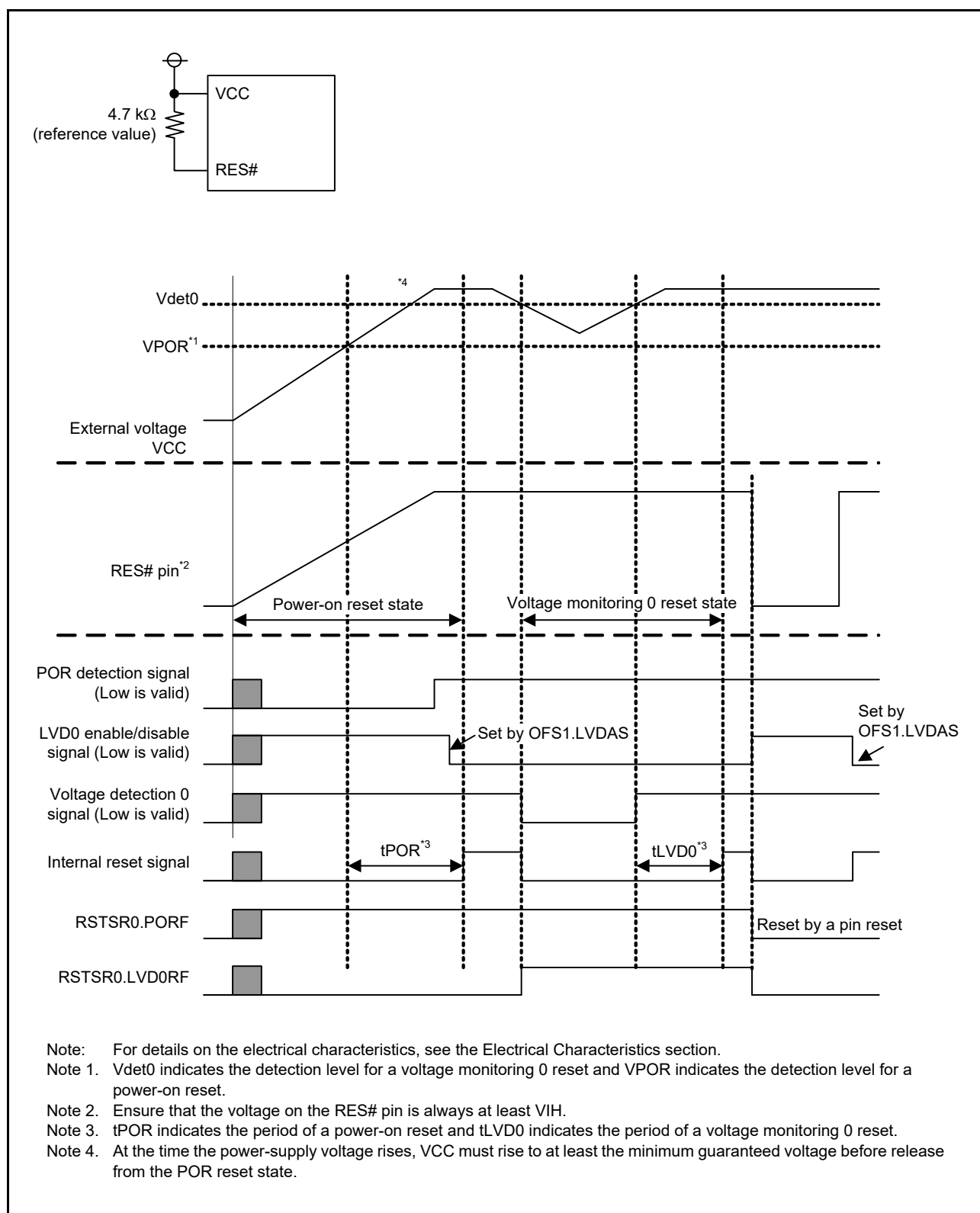


Figure 6.1 Operation Examples During a Power-On Reset and Voltage Monitoring 0 Reset

6.3.3 Voltage Monitoring 1 Reset and Voltage Monitoring 2 Reset

The voltage monitoring 1 reset and voltage monitoring 2 reset are internal resets generated by the voltage monitoring circuit.

When the voltage monitoring 1 interrupt/reset enable bit (LVD1RIE) is set to 1 (enabling generation of a reset or interrupt by the voltage detection circuit) and the voltage monitoring 1 circuit mode select bit (LVD1RI) is set to 1 (selecting generation of a reset in response to detection of a low voltage) in the voltage monitoring 1 circuit control register 0 (LVD1CR0), the RSTSR0.LVD1RF flag is set to 1 and the voltage-detection circuit generates a voltage monitoring 1 reset if VCC falls to or below Vdet1.

Likewise, when the voltage monitoring 2 interrupt/reset enable bit (LVD2RIE) is set to 1 (enabling generation of a reset or interrupt by the voltage detection circuit) and the voltage monitoring 2 circuit mode select bit (LVD2RI) is set to 1 (selecting generation of a reset in response to detection of a low voltage) in voltage monitoring 2 circuit control register 0 (LVD2CR0), the RSTSR0.LVD2RF flag is set to 1 and the voltage detection circuit generates a voltage monitoring 2 reset if VCC falls to or below Vdet2.

Timing for release from the voltage monitoring 1 reset state is selectable with the voltage monitoring 1 reset negation select bit (LVD1RN) in the LVD1CR0 register. When the LVD1CR0.LVD1RN bit is 0 and VCC has fallen to or below Vdet1, the CPU is released from the internal reset state and starts reset exception handling once the voltage monitoring 1 reset time (tLVD1) has elapsed after VCC has risen above Vdet1. When the LVD1CR0.LVD1RN bit is 1 and VCC has fallen to or below Vdet1, the CPU is released from the internal reset state and starts reset exception handling once the voltage monitoring 1 reset time (tLVD1) has elapsed.

Likewise, timing for release from the voltage monitoring 2 reset state is selectable by setting the voltage monitoring 2/comparator A2 reset negation select bit (LVD2RN) in the LVD2CR0 register. Detection levels Vdet1 and Vdet2 can be changed by settings in the voltage detection level select register (LVDLVLR).

Figure 6.2 shows examples of operations during voltage monitoring 1 and 2 resets.

For details on the voltage monitoring 1 reset and voltage monitoring 2 reset, refer to section 8, Voltage Detection Circuit (LVDAb).

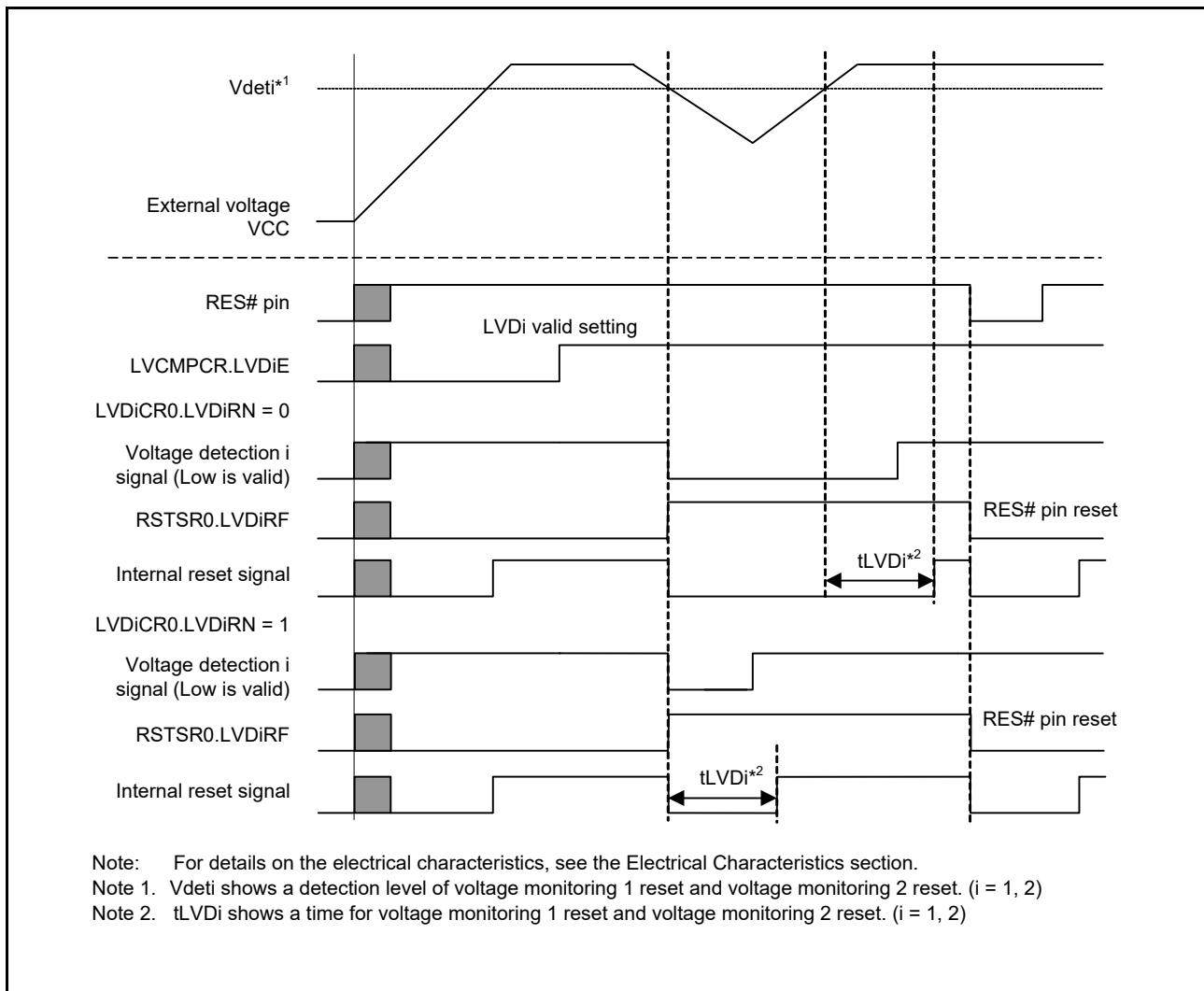


Figure 6.2 Operation Examples During Voltage Monitoring 1 Reset and Voltage Monitoring 2 Reset

6.3.4 Independent Watchdog Timer Reset

Independent watchdog timer reset is an internal reset generated by the independent watchdog timer.

Output of the independent watchdog timer reset from the independent watchdog timer can be selected by setting the IWDTR reset control register (IWDTRCR) and option function select register 0 (OFS0).

When output of the independent watchdog timer reset is selected, an independent watchdog timer reset is generated if the independent watchdog timer underflows, or if data is written outside the refresh-permitted period. When the internal reset time has elapsed after the independent watchdog timer reset has been generated, the internal reset is canceled and the CPU starts the reset exception handling.

For details on the independent watchdog timer reset, see section 25, Independent Watchdog Timer (IWDtA).

6.3.5 Software Reset

The software reset is an internal reset generated by the software reset circuit.

When A501h is written to SWRR, a software reset is generated. When the internal reset time has elapsed after the software reset is generated, the internal reset is canceled and the CPU starts the reset exception handling.

6.3.6 Determination of Cold/Warm Start

By reading the CWSF flag in RSTSR1, the type of reset processing caused can be identified; that is, whether a power-on reset has caused the reset processing (cold start) or a reset signal input during operation has caused the reset processing (warm start).

The CWSF flag in RSTSR1 is set to 0 when a power-on reset occurs (cold start); otherwise the flag is not set to 0. The flag is set to 1 when 1 is written to it through programming; it is not set to 0 even when 0 is written.

Figure 6.3 shows an example of cold/warm start determination operation.

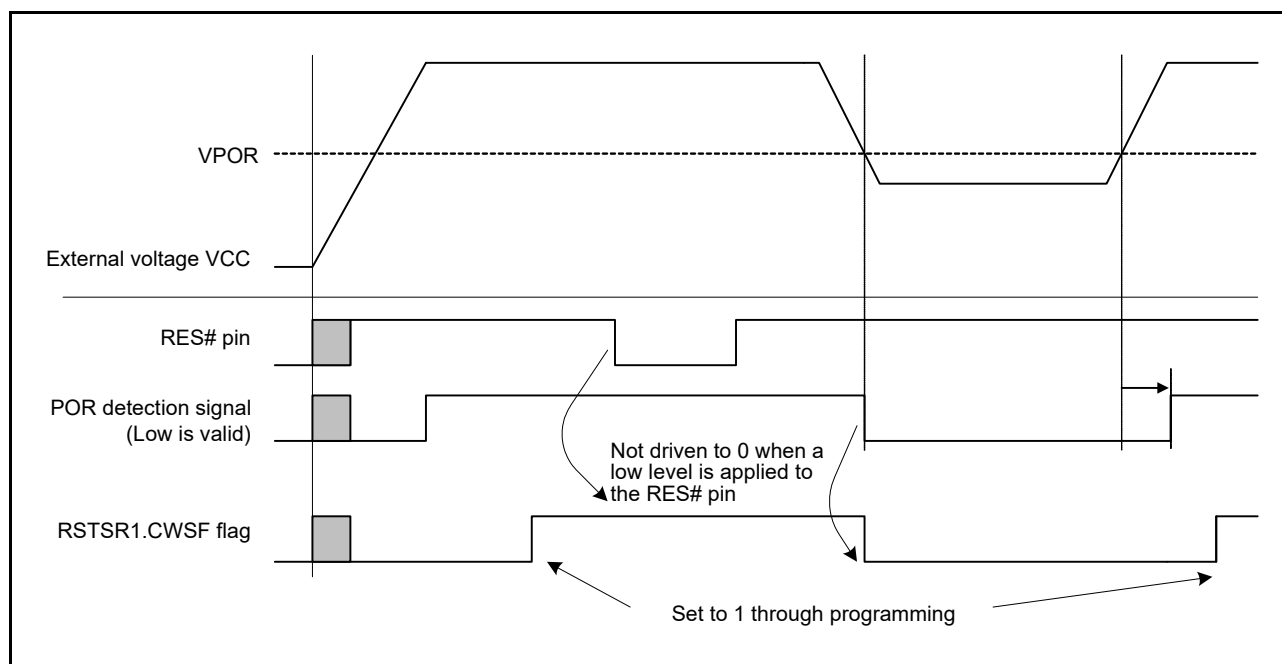


Figure 6.3 Example of Cold/Warm Start Determination Operation

6.3.7 Determination of Reset Generation Source

Reading RSTSR0 and RSTSR2 determines which reset was used to execute the reset exception handling.

Figure 6.4 shows an example of the flow to identify a reset generation source.

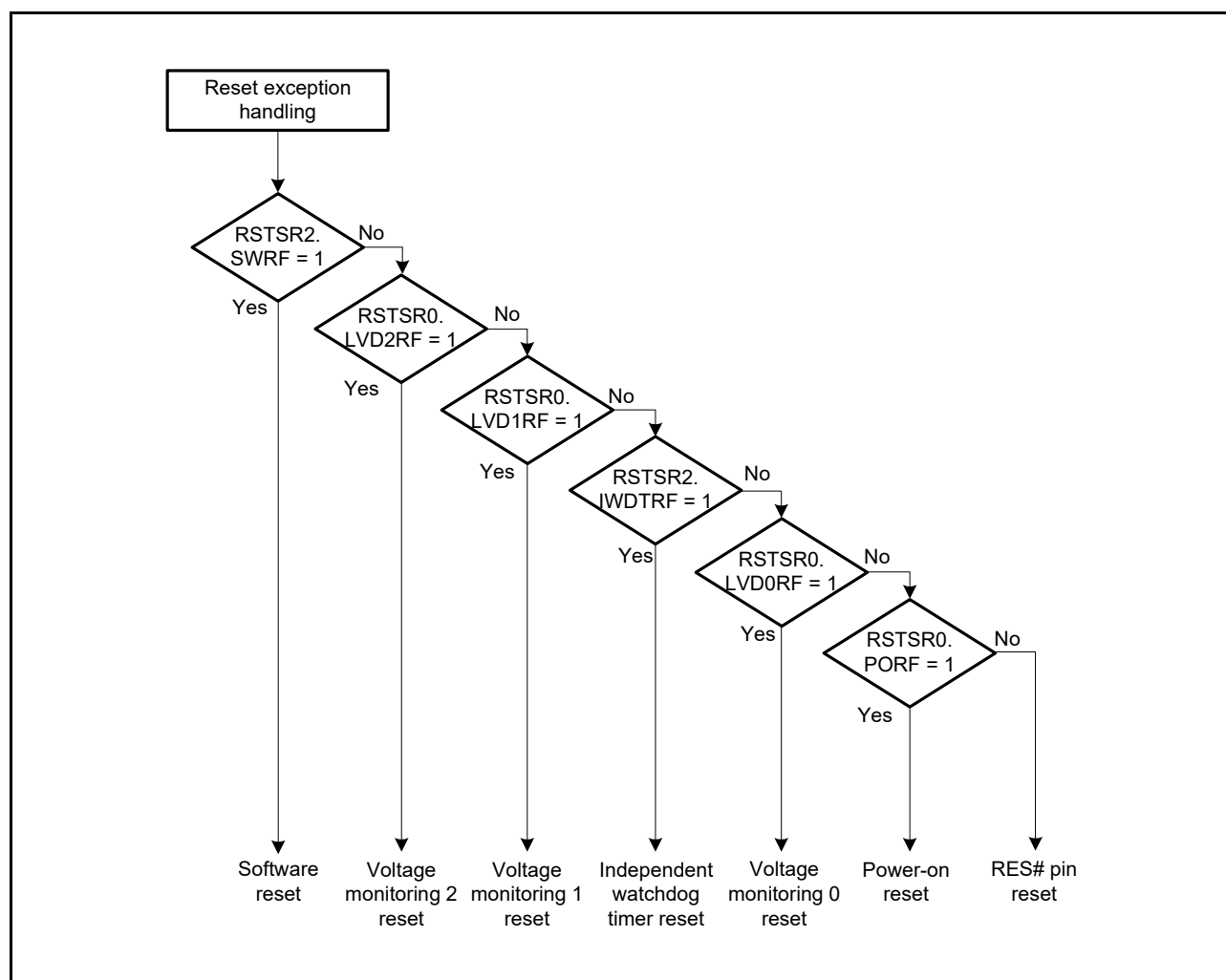


Figure 6.4 Example of Reset Generation Source Determination Flow

7. Option-Setting Memory (OFSM)

7.1 Overview

Option-setting memory (OFSM) refers to a set of registers that are provided for selecting the state of the microcontroller after a reset. The option-setting memory is allocated in the ROM.

Figure 7.1 shows the option-setting memory area.

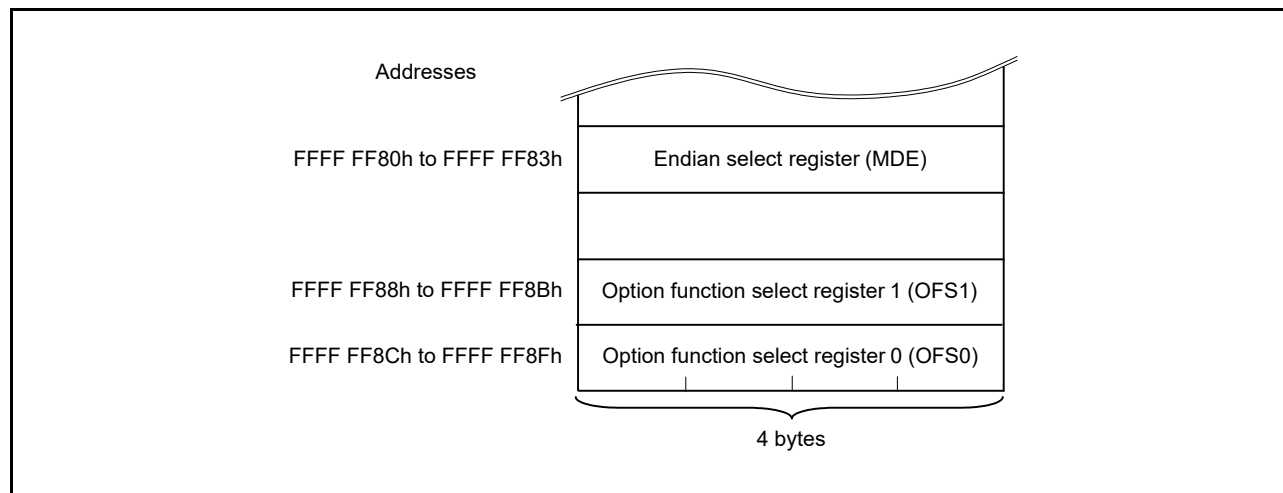


Figure 7.1 Option-Setting Memory Area

7.2 Register Descriptions

7.2.1 Option Function Select Register 0 (OFS0)

Address(es): OFSM.OFS0 FFFF FF8Ch

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Value after reset: The value set by the user*1

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	IWDTSLCSTP	—	IWDTRSTIRQS	IWDTRPSS[1:0]	IWDTRPES[1:0]	IWDTCKS[3:0]			IWDTTOS[1:0]			IWDTSTR	—		

Value after reset: The value set by the user*1

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	When reading, this bit returns the value written by the user. The write value should be 1.	R
b1	IWDTSTRT	IWDT Start Mode Select	0: IWDT is automatically activated in auto-start mode after a reset 1: IWDT is halted after a reset	R
b3, b2	IWDTTOS[1:0]	IWDT Timeout Period Select	b3 b2 0 0: 128 cycles (007Fh) 0 1: 512 cycles (01FFh) 1 0: 1024 cycles (03FFh) 1 1: 2048 cycles (07FFh)	R
b7 to b4	IWDTCKS[3:0]	IWDT Clock Frequency Division Ratio Select	b7 b4 0 0 0 0: No division 0 0 1 0: Divide-by-16 0 0 1 1: Divide-by-32 0 1 0 0: Divide-by-64 1 1 1 1: Divide-by-128 0 1 0 1: Divide-by-256 Settings other than above are prohibited.	R
b9, b8	IWDTRPES[1:0]	IWDT Window End Position Select	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (No window end position setting)	R
b11, b10	IWDTRPSS[1:0]	IWDT Window Start Position Select	b11 b10 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (No window start position setting)	R
b12	IWDTRSTIRQS	IWDT Reset Interrupt Request Select	0: Non-maskable interrupt request is enabled 1: Reset is enabled	R
b13	—	Reserved	When reading, this bit returns the value written by the user. The write value should be 1.	R
b14	IWDTSLCSTP	IWDT Sleep Mode Count Stop Control	0: Counting stop is disabled 1: Counting stop is enabled when entering sleep, software standby, or deep sleep mode	R
b31 to b15	—	Reserved	When reading, these bits return the value written by the user. The write value should be 1.	R

Note 1. The value of the blank product is FFFF FFFFh. This register is set to a specified value after programming of the flash memory with the user program.

The OFS0 register is allocated in the ROM. Set this register at the same time as writing the program. After writing to the OFS0 register once, do not write to it again.

When erasing the block including the OFS0 register, the OFS0 register value becomes FFFF FFFFh.

The setting in the OFS0 register is ignored in boot mode, and this register functions similarly when it is set to FFFF FFFFh.

IWDTSTRT Bit (IWDT Start Mode Select)

This bit selects the mode in which the IWDT is activated after a reset (stopped state or activated in auto-start mode). When activated in auto-start mode, the OFS0 register setting for the IWDT is effective.

IWDTTOPS[1:0] Bits (IWDT Timeout Period Select)

These bits select the timeout period, i.e. the time it takes for the down-counter to underflow, as 128, 512, 1024, or 2048 cycles of the frequency-divided clock set by the IWDTCKS[3:0] bits. The time (number of IWDT-dedicated clock cycles) it takes to underflow after a refresh operation is determined by the combination of the IWDTCKS[3:0] bits and IWDTTOPS[1:0] bits.

For details, see section 25, Independent Watchdog Timer (IWDTa).

IWDTCKS[3:0] Bits (IWDT Clock Frequency Division Ratio Select)

These bits select, from 1/1, 1/16, 1/32, 1/64, 1/128, and 1/256, the division ratio of the prescaler to divide the frequency of the IWDT-dedicated clock. Using the setting of these bits together with the IWDTTOPS[1:0] bit setting, the IWDT counting period can be set from 128 to 524288 IWDT-dedicated clock cycles.

For details, see section 25, Independent Watchdog Timer (IWDTa).

IWDRPES[1:0] Bits (IWDT Window End Position Select)

These bits select the position of the end of the window for the down-counter as 0%, 25%, 50%, or 75% of the value being counted by the counter. The value of the window end position must be smaller than the value of the window start position (window start position > window end position). If the value for the window end position is greater than the value for the window start position, only the value for the window start position is effective.

The counter values corresponding to the settings for the start and end positions of the window in the IWDRPSS[1:0] and IWDRPES[1:0] bits vary with the setting of the IWDTTOPS[1:0] bits.

For details, refer to section 25, Independent Watchdog Timer (IWDTa).

IWDRPSS[1:0] Bits (IWDT Window Start Position Select)

These bits select the position where the window for the down-counter starts as 25%, 50%, 75%, or 100% of the value being counted (the point at which counting starts is 100% and the point at which an underflow occurs is 0%). The interval between the positions where the window starts and ends becomes the period in which refreshing is possible, and refreshing is not possible outside this period.

For details, refer to section 25, Independent Watchdog Timer (IWDTa).

IWDRSTIRQS Bit (IWDT Reset Interrupt Request Select)

The setting of this bit selects the operation on an underflow of the down-counter or generation of a refresh error. Either an independent watchdog timer reset or a non-maskable interrupt request is selectable.

For details, refer to section 25, Independent Watchdog Timer (IWDTa).

IWDTSLCSTP Bit (IWDT Sleep Mode Count Stop Control)

This bit selects whether to stop counting when entering sleep, software standby, or deep sleep mode.

For details, see section 25, Independent Watchdog Timer (IWDTa).

7.2.2 Option Function Select Register 1 (OFS1)

Address(es): OFSM.OFS1 FFFF FF88h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Value after reset: The value set by the user*1

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	HOCOFRQ[1:0]	—	—	—	—	HOCOEN	—	—	—	—	FASTSTUP	LVDAS	VDSEL[1:0]	—

Value after reset: The value set by the user*1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	VDSEL[1:0]	Voltage Detection 0 Level Select	b1 b0 0 0: 3.85 V is selected 0 1: 2.85 V is selected 1 0: 2.53 V is selected Settings other than above are prohibited when the voltage detection 0 circuit is used.	R
b2	LVDAS	Voltage Detection 0 Circuit Start	0: Voltage monitoring 0 reset is enabled after a reset 1: Voltage monitoring 0 reset is disabled after a reset	R
b3	FASTSTUP	Power-On Fast Startup Time	0: Fast startup time at power on 1: Normal startup	R
b7 to b4	—	Reserved	When reading, these bits return the value written by the user. The write value should be 1.	R
b8	HOCOEN	HOCO Oscillation Enable	0: HOCO oscillation is enabled after a reset 1: HOCO oscillation is disabled after a reset	R
b11 to b9	—	Reserved	When reading, these bits return the value written by the user. The write value should be 1.	R
b13 to b12	HOCOFRQ[1:0]	HOCO Frequency Select	b13 b12 0 0: 64 MHz is selected 0 1: 48 MHz is selected 1 0: 24 MHz is selected 1 1: 32 MHz is selected	R
b31 to b14	—	Reserved	When reading, these bits return the value written by the user. The write value should be 1.	R

Note 1. The value of the blank product is FFFF FFFFh. This register is set to a specified value after programming of the flash memory with the user program.

The OFS1 register is allocated in the ROM. Set this register at the same time as writing the program. After writing, do not write additions to this register.

When erasing the block including the OFS1 register, the OFS1 register value becomes FFFF FFFFh.

The setting in the OFS1 register is ignored in boot mode, and this register functions similarly when it is set to FFFF FFFFh.

VDSEL[1:0] Bits (Voltage Detection 0 Level Select)

These bits select the voltage detection level to be monitored by the voltage detection 0 circuit.

LVDAS Bit (Voltage Detection 0 Circuit Start)

This bit selects whether the voltage monitoring 0 reset is enabled or disabled after a reset.

The Vdet0 voltage to be monitored by the voltage detection 0 circuit is selected by the VDSEL[1:0] bits.

FASTSTUP Bit (Power-On Fast Startup Time)

The startup time can be reduced by setting this bit to 0 (fast startup time at power on) when it is possible to meet the power-on VCC rising gradient (during fast startup time) shown in Electrical Characteristics. Do not set this bit to 0 when it is not possible to meet the power-on VCC rising gradient (during fast startup time).

HOCOEN Bit (HOCO Oscillation Enable)

This bit selects whether the HOCO oscillation is effective or not after a reset.

Setting the HOCOEN bit to 0 allows the HOCO oscillation to be started before the CPU starts operation, and therefore reduces the wait time for oscillation stabilization.

Note that even if the HOCOEN bit is set to 0, the system clock source is not switched to HOCO. The system clock source is switched to HOCO only by modifying the clock source select bits (SCKCR3.CKSEL[2:0]) from the CPU.

Also, when the HOCOEN bit is set to 0, the HOCO oscillation stabilization time (tHOCO) is secured by hardware, so the clock with the accuracy of the HOCO oscillation frequency (fHOCO) shown in Electrical Characteristics is supplied after release from the CPU reset state.

7.2.3 Endian Select Register (MDE)

Address(es): OFSM.MDE FFFF FF80h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset: The value set by the user*1															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	MDE[2:0]	—
Value after reset: The value set by the user*1															

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	MDE[2:0]	Endian Select	b2 b0 0 0 0: Big endian 1 1 1: Little endian Settings other than above are prohibited.	R
b31 to b3	—	Reserved	When reading, these bits return the value written by the user. The write value should be 1.	R

Note 1. The value of the blank product is FFFF FFFFh. This register is set to a specified value after programming of the flash memory with the user program.

The MDE register selects the endian for the CPU.

MDE is allocated in the ROM. Set the register at the same time as writing the program. After writing to the register once, do not write to it again.

When erasing the block including the MDE register, the MDE register value becomes FFFF FFFFh.

MDE[2:0] Bits (Endian Select)

These bits select little endian or big endian for the CPU.

7.3 Usage Note

7.3.1 Setting Example of Option-Setting Memory

Since the option-setting memory is allocated in the ROM, values cannot be written by executing instructions. Write appropriate values when writing the program. An example of the settings is shown below.

- To set FFFF FFF8h in the OFS0 register

```
.ORG    0FFFFFF8CH
.LWORD  0FFFFFFF8H
```

Note: Programming formats vary depending on the compiler. Refer to the compiler manual for details.

8. Voltage Detection Circuit (LVDAb)

The voltage detection circuit (LVD) monitors the voltage level input to the VCC pin using a program.

8.1 Overview

In voltage detection 0, the detection voltage can be selected from three levels using option function select register 1 (OFS1).

In voltage detection 1, the detection voltage can be selected from 10 levels using the voltage detection level select register (LVDLVLR).

In voltage detection 2, the detection voltage can be selected from four levels using the LVDLVLR register.

Voltage monitoring 0 reset, voltage monitoring 1 reset/interrupt, and voltage monitoring 2 reset/interrupt can be used.

Table 8.1 lists the specifications of the voltage detection circuit. Figure 8.1 is a block diagram of the voltage detection circuit. Figure 8.2 is a block diagram of the voltage monitoring 1 interrupt/reset circuit. Figure 8.3 is a block diagram of the voltage monitoring 2 interrupt/reset circuit.

Table 8.1 LVD Specifications

Item		Voltage Monitoring 0	Voltage Monitoring 1	Voltage Monitoring 2
VCC monitoring	Monitored voltage	Vdet0	Vdet1	Vdet2
	Detection voltage	Voltage selectable from three levels using OFS1	Voltage selectable from 10 levels using the LVDLVLR.LVD1LVL[3:0] bits	Voltage selectable from four levels using the LVDLVLR.LVD2LVL[1:0] bits
	Monitoring flag	Not available	LVD1SR.LVD1MON flag: Monitors whether voltage is higher or lower than Vdet1 LVD1SR.LVD1DET flag: Vdet1 passage detection	LVD2SR.LVD2MON flag: Monitors whether voltage is higher or lower than Vdet2 LVD2SR.LVD2DET flag: Vdet2 passage detection
Process upon voltage detection	Reset	Voltage monitoring 0 reset Reset when Vdet0 > VCC CPU restart after specified time with VCC > Vdet0	Voltage monitoring 1 reset Reset when Vdet1 > VCC CPU restart timing selectable: after specified time with VCC > Vdet1 or Vdet1 > VCC	Voltage monitoring 2 reset Reset when Vdet2 > VCC CPU restart timing selectable: after specified time with VCC > Vdet2 or after specified time with Vdet2 > VCC
	Interrupt	Not available	Voltage monitoring 1 interrupt Non-maskable or maskable interrupt is selectable	Voltage monitoring 2 interrupt Non-maskable or maskable interrupt is selectable
			Interrupt request issued when Vdet1 > VCC and VCC > Vdet1 or either	Interrupt request issued when Vdet2 > VCC and VCC > Vdet2 or either

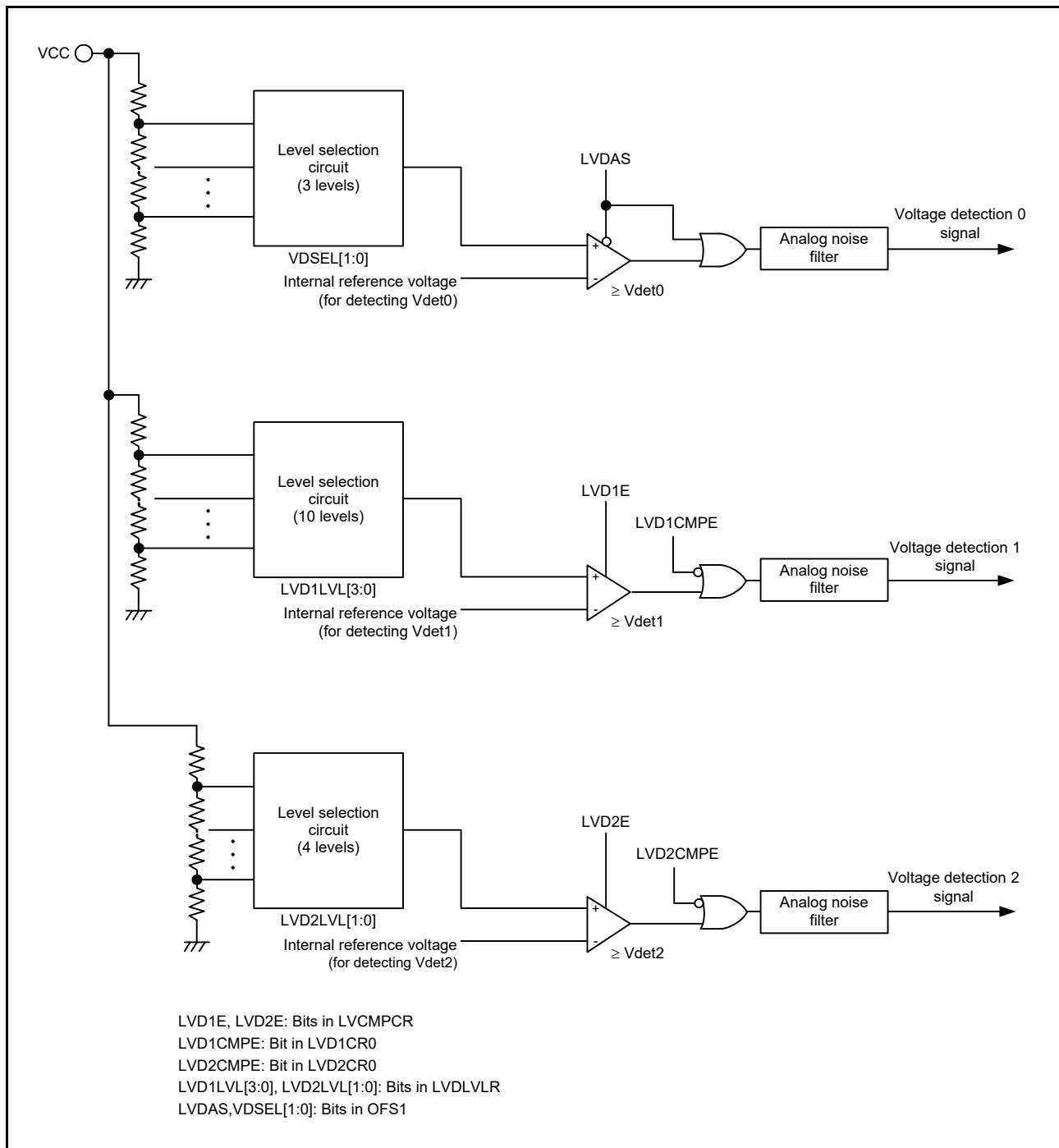


Figure 8.1 Block Diagram of the LVD



8.2 Register Descriptions

8.2.1 Voltage Monitoring 1 Circuit Control Register 1 (LVD1CR1)

Address(es): SYSTEM.LVD1CR1 0008 00E0h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	LVD1IRQSEL	LVD1IDTSEL[1:0]	—
Value after reset:	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	LVD1IDTSEL [1:0]	Voltage Monitoring 1 Interrupt Generation Condition Select	b1 b0 0 0: When $VCC \geq V_{det1}$ (rise) is detected 0 1: When $VCC < V_{det1}$ (drop) is detected 1 0: When drop and rise are detected 1 1: Setting prohibited	R/W
b2	LVD1IRQSEL	Voltage Monitoring 1 Interrupt Type Select	0: Non-maskable interrupt 1: Maskable interrupt	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

8.2.2 Voltage Monitoring 1 Circuit Status Register (LVD1SR)

Address(es): SYSTEM.LVD1SR 0008 00E1h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	LVD1MON	LVD1DET
0	0	0	0	0	0	1	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	LVD1DET	Voltage Monitoring 1 Voltage Change Detection Flag	0: Not detected 1: Vdet1 passage detection	R/(W) *1
b1	LVD1MON	Voltage Monitoring 1 Signal Monitor Flag	0: VCC < Vdet1 1: VCC ≥ Vdet1 or LVD1MON circuit is disabled	R
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

Note 1. Only 0 can be written to this bit. After writing 0 to this bit, it takes two system clock cycles for the bit to be read as 0.

LVD1DET Flag (Voltage Monitoring 1 Voltage Change Detection Flag)

The LVD1DET flag is enabled when the LVCMPCR.LVD1E bit is 1 (voltage detection 1 circuit enabled) and the LVD1CR0.LVD1CMPE bit is 1 (voltage monitoring 1 circuit comparison result output enabled).

The LVD1DET flag should be set to 0 after LVD1CR0.LVD1RIE is set to 0 (disabled). LVD1CR0.LVD1RIE can be set to 1 (enabled) again after a period of two or more cycles of PCLKB has elapsed.

Depending on the number of cycles of PCLKB defined for access to read an I/O register, two or more cycles of PCLKB may have to be secured as wait time.

LVD1MON Flag (Voltage Monitoring 1 Signal Monitor Flag)

The LVD1MON flag is enabled when the LVCMPCR.LVD1E bit is 1 (voltage detection 1 circuit enabled) and the LVD1CR0.LVD1CMPE bit is 1 (voltage monitoring 1 circuit comparison result output enabled).

8.2.3 Voltage Monitoring 2 Circuit Control Register 1 (LVD2CR1)

Address(es): SYSTEM.LVD2CR1 0008 00E2h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	LVD2IRQSEL	LVD2IDTSEL[1:0]	
0	0	0	0	0	0	0	1

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b1, b0	LVD2IDTSEL [1:0]	Voltage Monitoring 2 Interrupt Generation Condition Select	b1 b0 0 0: When VCC $\geq$ Vdet2 (rise) is detected 0 1: When VCC < Vdet2 (drop) is detected 1 0: When drop and rise are detected 1 1: Setting prohibited	R/W
b2	LVD2IRQSEL	Voltage Monitoring 2 Interrupt Type Select	0: Non-maskable interrupt 1: Maskable interrupt	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

8.2.4 Voltage Monitoring 2 Circuit Status Register (LVD2SR)

Address(es): SYSTEM.LVD2SR 0008 00E3h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	LVD2MON	LVD2DET
0	0	0	0	0	0	1	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	LVD2DET	Voltage Monitoring 2 Voltage Change Detection Flag	0: Not detected 1: Vdet2 passage detection	R/(W) *1
b1	LVD2MON	Voltage Monitoring 2 Signal Monitor Flag	0: VCC < Vdet2 1: VCC ≥ Vdet2 or LVD2MON is disabled	R
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

Note 1. Only 0 can be written to this bit. After writing 0 to this bit, it takes two system clock cycles for the bit to be read as 0.

LVD2DET Flag (Voltage Monitoring 2 Voltage Change Detection Flag)

The LVD2DET flag is enabled when the LVCMPCR.LVD2E bit is 1 (voltage detection 2 circuit enabled) and the LVD2CR0.LVD2CMPE bit is 1 (voltage monitoring 2 circuit comparison result output enabled).

The LVD2DET flag should be set to 0 after LVD2CR0.LVD2RIE is set to 0 (disabled). LVD2CR0.LVD2RIE can be set to 1 (enabled) again after a period of two or more cycles of PCLKB has elapsed.

Depending on the number of cycles of PCLKB defined for access to read an I/O register, two or more cycles of PCLKB may have to be secured as wait time.

LVD2MON Flag (Voltage Monitoring 2 Signal Monitor Flag)

The LVD2MON flag is enabled when the LVCMPCR.LVD2E bit is 1 (voltage detection 2 circuit enabled) and the LVD2CR0.LVD2CMPE bit is 1 (voltage monitoring 2 circuit comparison result output enabled).

8.2.5 Voltage Monitoring Circuit Control Register (LVCMPPCR)

Address(es): SYSTEM.LVCMPPCR 0008 C297h

b7	b6	b5	b4	b3	b2	b1	b0
—	LVD2E	LVD1E	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b5	LVD1E	Voltage Detection 1 Enable	0: Voltage detection 1 circuit disabled 1: Voltage detection 1 circuit enabled	R/W
b6	LVD2E	Voltage Detection 2 Enable	0: Voltage detection 2 circuit disabled 1: Voltage detection 2 circuit enabled	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

LVD1E Bit (Voltage Detection 1 Enable)

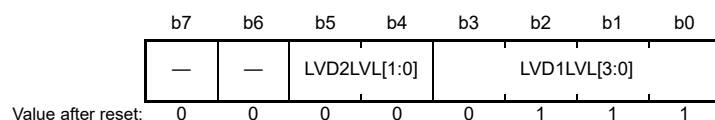
When using voltage detection 1 interrupt/reset or the LVD1SR.LVD1MON flag, set the LVD1E bit to 1. The voltage detection 1 circuit starts once $td(E-A)$ passes after the LVD1E bit value is changed from 0 to 1.

LVD2E Bit (Voltage Detection 2 Enable)

When using voltage detection 2 interrupt/reset or the LVD2SR.LVD2MON flag, set the LVD2E bit to 1. The voltage detection 2 circuit starts once $td(E-A)$ passes after the LVD2E bit value is changed from 0 to 1.

8.2.6 Voltage Detection Level Select Register (LVDLVLR)

Address(es): SYSTEM.LVDLVLR 0008 C298h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	LVD1LVL[3:0]	Voltage Detection 1 Level Select (Standard voltage during drop in voltage)	b3b0 0 0 0 0: 4.29 V 0 0 0 1: 4.16 V 0 0 1 0: 4.03 V 0 0 1 1: 3.86 V 0 1 0 0: 3.10 V 0 1 0 1: 3.00 V 0 1 1 0: 2.90 V 0 1 1 1: 2.80 V 1 0 0 0: 2.68 V 1 0 0 1: 2.59 V Settings other than those listed above are prohibited.	R/W
b5, b4	LVD2LVL[1:0]	Voltage Detection 2 Level Select (Standard voltage during drop in voltage)	b5b4 0 0: 4.32 V 0 1: 4.17 V 1 0: 4.03 V 1 1: 3.84 V	R/W
b7, b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

When changing the LVDLVLR register, first set the LVCMPCR.LVD1E and LVCMPCR.LVD2E bits to 0 (voltage detection n circuit disabled) (n = 1, 2).

When a setting is made so that the voltage detection level range set by the LVD1LVL[3:0] bit overlaps with the range set by the LVD2LVL[1:0] bit register, it cannot be specified which of LVD1 and LVD2 is used for voltage detection. For details on the voltage detection level range, refer to **section 38, Electrical Characteristics**.

When LVD0 is enabled, set the detection level for LVD1 higher than that of LVD0.

While LVD0 is enabled, only execute changing of the voltage detection level using the LVD1LVL[3:0] bits one time after release from the reset state.

8.2.7 Voltage Monitoring 1 Circuit Control Register 0 (LVD1CR0)

Address(es): SYSTEM.LVD1CR0 0008 C29Ah

b7	b6	b5	b4	b3	b2	b1	b0
LVD1RN	LVD1RI	—	—	—	LVD1CMPE	—	LVD1RIE
Value after reset:	1	0	0	0	X	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	LVD1RIE	Voltage Monitoring 1 Interrupt/Reset Enable	0: Disabled 1: Enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	LVD1CMPE	Voltage Monitoring 1 Circuit Comparison Result Output Enable	0: Voltage monitoring 1 circuit comparison results output disabled 1: Voltage monitoring 1 circuit comparison results output enabled	R/W
b3	—	Reserved	The read value is undefined. The write value should be 0.	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	LVD1RI	Voltage Monitoring 1 Circuit Mode Select	0: Voltage monitoring 1 interrupt occurs when the voltage passes Vdet1 1: Voltage monitoring 1 reset occurs when the voltage falls below Vdet1	R/W
b7	LVD1RN	Voltage Monitoring 1 Reset Negation Select	0: Negation follows a stabilization time (tLVD1) after VCC > Vdet1 is detected. 1: Negation follows a stabilization time (tLVD1) after assertion of the voltage monitoring 1 reset.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

LVD1RN Bit (Voltage Monitoring 1 Reset Negation Select)

If the LVD1RN bit is to be set to 1 (negation follows a stabilization time after assertion of the voltage monitoring 1 reset), set the LOCOCR.LCSTP bit to 0 (LOCO is operating). Furthermore, if a transition to software standby mode, the only possible value for the LVD1RN bit is 0 (negation follows a stabilization time after VCC > Vdet1 is detected). Do not set the LVD1RN bit to 1 (negation follows a stabilization time after assertion of the voltage monitoring 1 reset).

8.2.8 Voltage Monitoring 2 Circuit Control Register 0 (LVD2CR0)

Address(es): SYSTEM.LVD2CR0 0008 C29Bh

b7	b6	b5	b4	b3	b2	b1	b0
LVD2RN	LVD2RI	—	—	—	LVD2CMPE	—	LVD2RIE
Value after reset:	1	0	0	0	X	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	LVD2RIE	Voltage Monitoring 2 Interrupt/Reset Enable	0: Disabled 1: Enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	LVD2CMPE	Voltage Monitoring 2 Circuit Comparison Result Output Enable	0: Voltage monitoring 2 circuit comparison results output disabled 1: Voltage monitoring 2 circuit comparison results output enabled	R/W
b3	—	Reserved	The read value is undefined. The write value should be 0.	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	LVD2RI	Voltage Monitoring 2 Circuit Mode Select	0: Voltage monitoring 2 interrupt during Vdet2 passage 1: Voltage monitoring 2 reset enabled when the voltage falls to and below Vdet2	R/W
b7	LVD2RN	Voltage Monitoring 2 Reset Negation Select	0: Negation follows a stabilization time (tLVD2) after VCC > Vdet2 is detected. 1: Negation follows a stabilization time (tLVD2) after assertion of the voltage monitoring 2 reset.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

LVD2RN Bit (Voltage Monitoring 2 Reset Negation Select)

If the LVD2RN bit is to be set to 1 (negation follows a stabilization time after assertion of the voltage monitoring 2 reset), set the LOCOCR.LCSTP bit to 0 (LOCO is operating). Furthermore, if a transition to software standby mode, the only possible value for the LVD2RN bit is 0 (negation follows a stabilization time after VCC > Vdet2 is detected). Do not set the LVD2RN bit to 1 (negation follows a stabilization time after assertion of the voltage monitoring 2 reset).

8.3 VCC Input Voltage Monitor

8.3.1 Monitoring Vdet0

Monitoring Vdet0 is not possible.

8.3.2 Monitoring Vdet1

After making the following settings, the LVD1SR.LVD1MON flag can be used to monitor the results of comparison by voltage monitor 1.

- (1) Specify the detection voltage by setting the LVDLVLR.LVD1LVL[3:0] bits (voltage detection 1 level select).
- (2) Set the LVCMPCR.LVD1E bit to 1 (voltage detection 1 circuit enabled).
- (3) After waiting for $t_d(E-A)$, set the LVD1CR0.LVD1CMPE bit to 1 (voltage monitoring 1 circuit comparison results output enabled).

8.3.3 Monitoring Vdet2

After making the following settings, the LVD2SR.LVD2MON flag can be used to monitor the results of comparison by voltage monitor 2.

- (1) Specify the detection voltage by setting the LVDLVLR.LVD2LVL[1:0] bits (voltage detection 2 level).
- (2) Set the LVCMPCR.LVD2E bit to 1 (voltage detection 2 circuit enabled).
- (3) After waiting for $t_d(E-A)$, set the LVD2CR0.LVD2CMPE bit to 1 (voltage monitoring 2 circuit comparison results output enabled).

8.4 Reset from Voltage Monitor 0

When using the reset from voltage monitor 0, clear the voltage detection 0 circuit start bit (OFS1.LVDAS) to 0 (enabling the voltage monitor 0 reset after a reset).

Figure 8.4 shows an example of operations for a voltage monitoring 0 reset.

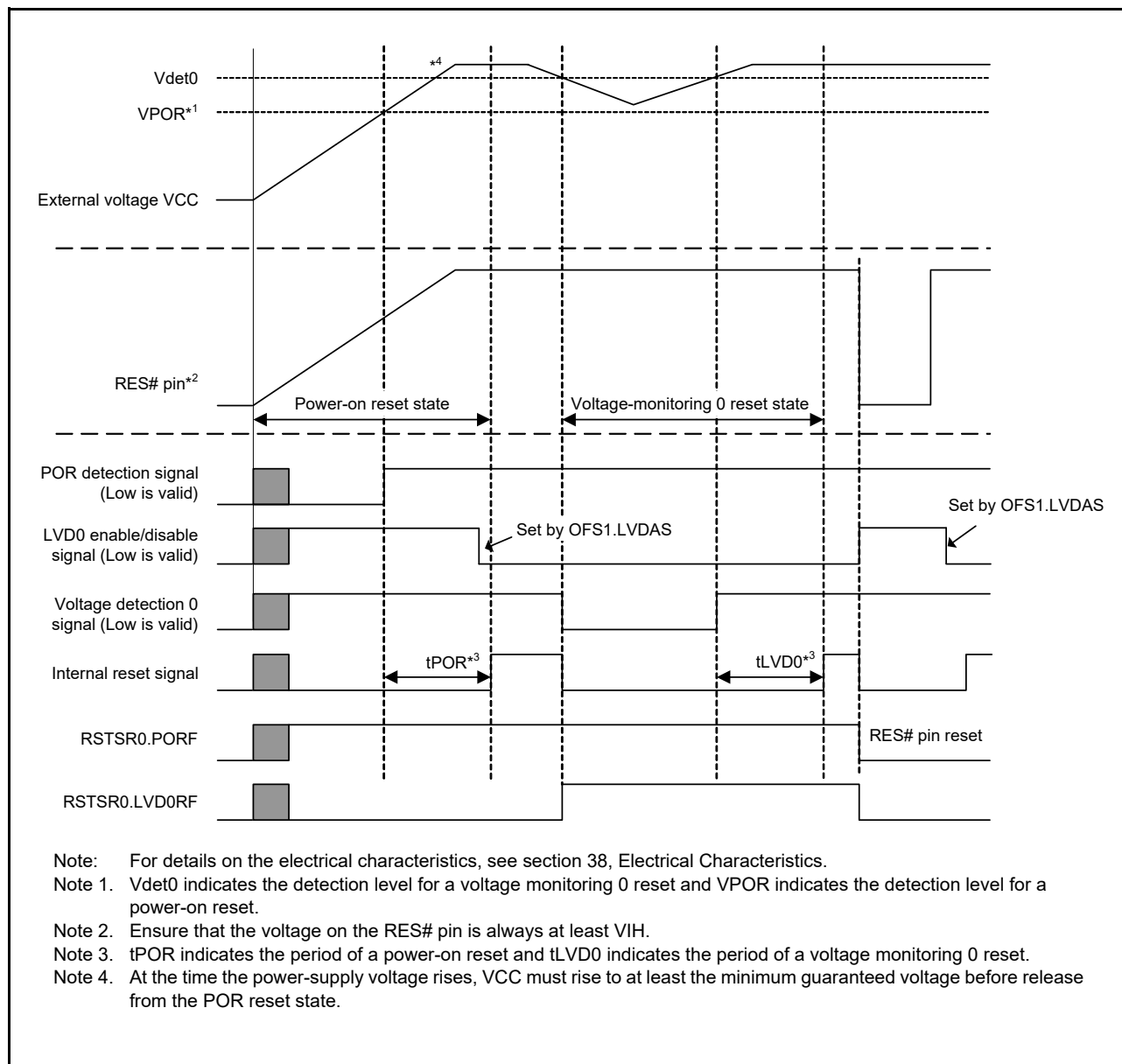


Figure 8.4 Example of Voltage Monitoring 0 Reset Operation

8.5 Interrupt and Reset from Voltage Monitoring 1

Table 8.2 shows the procedures for setting bits related to the voltage monitoring 1 interrupt and voltage monitoring 1 reset. Table 8.3 shows the procedures for stopping bits related to the voltage monitoring 1 interrupt and voltage monitoring 1 reset. Figure 8.5 shows an example of operations for a voltage monitoring 1 interrupt. For the operation of the voltage monitoring 1 reset, see Figure 6.2 in section 6, Resets.

Table 8.2 Procedures for Setting Bits Related to the Voltage Monitoring 1 Interrupt and Voltage Monitoring 1 Reset

Step	Voltage Monitoring 1 Interrupt	Voltage Monitoring 1 Reset
1*1	Select the detection voltage by setting the LVD1LVL[3:0] bits.	
2*1	Set the LVD1CR0.LVD1RI bit to 0 (voltage monitoring 1 interrupt).	Set the LVD1CR0.LVD1RI bit to 1 (voltage monitoring 1 reset). Select the type of reset negation by setting the LVD1CR0.LVD1RN bit.
3	Select the timing of interrupt requests by setting the LVD1CR1.LVD1IDTSEL[1:0] bits. Select the type of interrupt by setting the LVD1CR1.LVD1IRQSEL bit.	—
4	—	Set the LVD1CR0.LVD1RIE bit to 1 (voltage monitoring 1 interrupt/reset enabled).
5*1	Set the LVCMPCR.LVD1E bit to 1 (voltage detection 1 circuit enabled).	
6*1	Wait for at least $t_d(E-A)$.	
7	Set the LVD1CR0.LVD1CMPE bit to 1 (voltage monitoring 1 circuit comparison results output enabled).	
8	Wait for at least 2 μs .	—
9	Set the LVD1SR.LVD1DET bit to 0.	—
10	Set the LVD1CR0.LVD1RIE bit to 1 (voltage monitoring 1 interrupt/reset enabled).	—

Note 1. Steps 1, 2, 5, and 6 are not required if operation is with the setting to select the voltage monitoring 1 interrupt (LVD1CR0.LVD1RI = 0) and operation can be restarted by simply changing the settings of the LVD1CR1.LVD1IRQSEL and LVD1IDTSEL[1:0] bits after monitoring is stopped or if restarting is in a case where the settings related to the voltage-detection circuit were not changed after monitoring was stopped. When changes are to be made and operation is with the setting to select the voltage monitoring 1 reset (LVD1CR0.LVD1RI = 1), proceed through all steps from 1 to 10.

Table 8.3 Procedures for Stopping Bits Related to the Voltage Monitoring 1 Interrupt and Voltage Monitoring 1 Reset

Step	Voltage Monitoring 1 Interrupt	Voltage Monitoring 1 Reset
1	Set the LVD1CR0.LVD1RIE bit to 0 (voltage monitoring 1 interrupt/reset disabled).	—
2	Set the LVD1CR0.LVD1CMPE bit to 0 (voltage monitoring 1 circuit comparison results output disabled).	
3*1	Set the LVCMPCR.LVD1E bit to 0 (voltage detection 1 circuit disabled).	
4	—	Set the LVD1CR0.LVD1RIE bit to 0 (voltage monitoring 1 interrupt/reset disabled).
5	Modify settings of bits related to the voltage detection circuit registers other than LVCMPCR.LVD1E, LVD1CR0.LVD1RIE, and LVD1CR0.LVD1CMPE.	

Note 1. Step 3 is not required if operation is with the setting to select the voltage monitoring 1 interrupt (LVD1CR0.LVD1RI = 0) and operation can be restarted by simply changing the settings of the LVD1CR1.LVD1IRQSEL and LVD1IDTSEL[1:0] bits after monitoring is stopped or if restarting is in a case where the settings related to the voltage-detection circuit were not changed after monitoring was stopped. When changes are to be made and operation is with the setting to select the voltage monitoring 1 reset (LVD1CR0.LVD1RI = 1), proceed through all steps from 1 to 5.

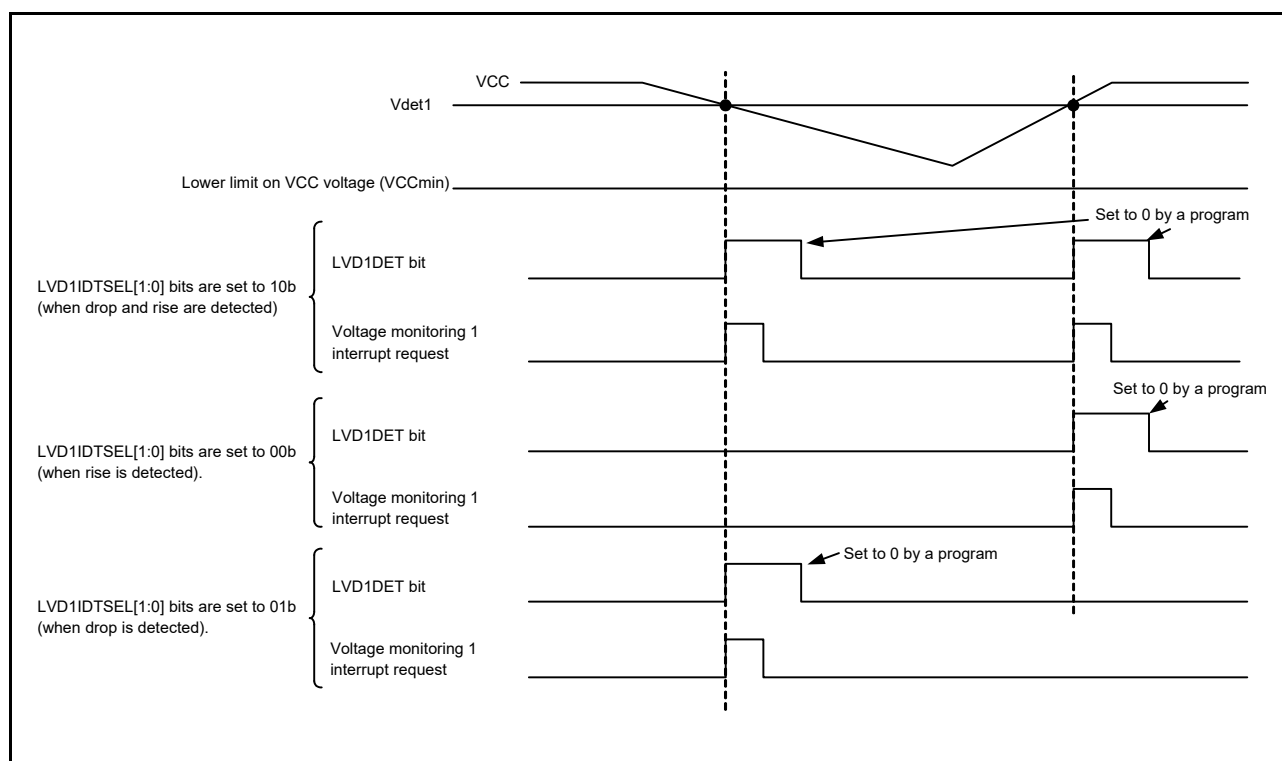


Figure 8.5 Example of Voltage Monitoring 1 Interrupt Operation

8.6 Interrupt and Reset from Voltage Monitoring 2

Table 8.4 shows the procedures for setting bits related to the voltage monitoring 2 interrupt and voltage monitoring 2 reset. Table 8.5 shows the procedure for stopping bits related to the voltage monitoring 2 interrupt and voltage monitoring 2 reset. Figure 8.6 shows an example of operations for a voltage monitoring 2 interrupt. For the operation of the voltage monitoring 2 reset, see Figure 6.2 in section 6, Resets.

Table 8.4 Procedures for Setting Bits Related to the Voltage Monitoring 2 Interrupt and Voltage Monitoring 2 Reset

Step	Voltage Monitoring 2 Interrupt	Voltage Monitoring 2 Reset
1*1	Select the detection voltage by setting the LVDLVL.R.LVD2LVL[1:0] bits.	
2*1	Set the LVD2CR0.LVD2RI bit to 0 (voltage monitoring 2 interrupt).	Set the LVD2CR0.LVD2RI bit to 1 (voltage monitoring 2 reset). Select the type of reset negation by setting the LVD2CR0.LVD2RN bit.
3	Select the timing of interrupt requests by setting the LVD2CR1.LVD2IDTSEL[1:0] bits. Select the type of interrupt by setting the LVD2CR1.LVD2IRQSEL bit.	—
4	—	Set the LVD2CR0.LVD2RIE bit to 1 (voltage monitoring 2 interrupt/reset enabled).
5*1	Set the LVCMPCR.LVD2E bit to 1 (voltage detection 2 circuit enabled).	
6*1	Wait for at least $t_d(E-A)$.	
7	Set the LVD2CR0.LVD2CMPE bit to 1 (voltage monitoring 2 circuit comparison results output enabled).	
8	Wait for at least 2 μs .	—
9	Set the LVD2SR.LVD2DET bit to 0.	—
10	Set the LVD2CR0.LVD2RIE bit to 1 (voltage monitoring 2 interrupt/reset enabled)	—

Note 1. Steps 1, 2, 5, and 6 are not required if operation is with the setting to select the voltage monitoring 2 interrupt (LVD2CR0.LVD2RI = 0) and operation can be restarted by simply changing the settings of the LVD2CR1.LVD2IRQSEL and LVD2IDTSEL[1:0] bits after monitoring is stopped or if restarting is in a case where the settings related to the voltage-detection circuit were not changed after monitoring was stopped. When changes are to be made and operation is with the setting to select the voltage monitoring 2 reset (LVD2CR0.LVD2RI = 1), proceed through all steps from 1 to 10.

Table 8.5 Procedures for Stopping Bits Related to the Voltage Monitoring 2 Interrupt and Voltage Monitoring 2 Reset

Step	Voltage Monitoring 2 Interrupt	Voltage Monitoring 2 Reset
1	Set the LVD2CR0.LVD2RIE bit to 0 (voltage monitoring 2 interrupt/reset disabled).	—
2	Set the LVD2CR0.LVD2CMPE bit to 0 (voltage monitoring 2 circuit comparison results output disabled).	
3*1	Set the LVCMPCR.LVD2E bit to 0 (voltage monitoring 2 circuit disabled).	
4	—	Set the LVD2CR0.LVD2RIE bit to 0 (voltage monitoring 2 interrupt/reset disabled).
5	Modify settings of bits related to the voltage detection circuit registers other than LVCMPCR.LVD2E, LVD2CR0.LVD2RIE, and LVD2CR0.LVD2CMPE.	

Note 1. Step 3 is not required if operation is with the setting to select the voltage monitoring 2 interrupt (LVD2CR0.LVD2RI = 0) and operation can be restarted by simply changing the settings of the LVD2CR1.LVD2IRQSEL and LVD2IDTSEL[1:0] bits after monitoring is stopped or if restarting is in a case where the settings related to the voltage-detection circuit were not changed after monitoring was stopped. When changes are to be made and operation is with the setting to select the voltage monitoring 2 reset (LVD2CR0.LVD2RI = 1), proceed through all steps from 1 to 5.

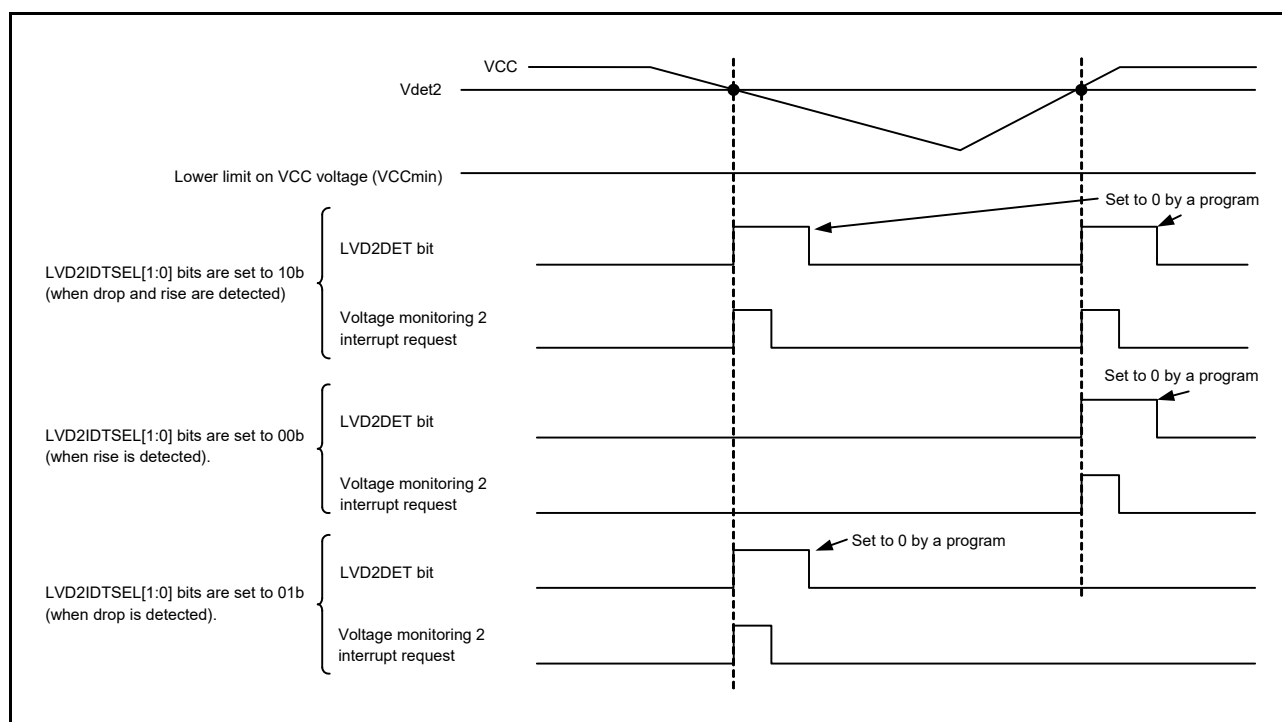


Figure 8.6 Example of Voltage Monitoring 2 Interrupt Operation

8.7 Usage Note

8.7.1 Note on Setting the Detection Level for LVD1 While LVD0 Is Enabled

While LVD0 is to be enabled, set the detection level for LVD1 to a value higher than that for LVD0. Moreover, once LVD0 has been enabled after release from the reset state, using the LVD1LVL[3:0] bits to change the voltage detection level for LVD1 is only allowed to proceed once.

9. Clock Generation Circuit

9.1 Overview

This MCU incorporates a clock generation circuit.

Table 9.1 lists the specifications of the clock generation circuit. Figure 9.1 shows a block diagram of the clock generation circuit.

Table 9.1 Specifications of Clock Generation Circuit

Item	Specification
Uses	- Generates the system clock (ICLK) to be supplied to the CPU, DTC, ROM, and RAM. - Generates the peripheral module clocks (PCLKB and PCLKD) to be supplied to peripheral modules. The peripheral module clock used as the operating clock is PCLKD for S12AD and PCLKB are for other modules. - Generates the FlashIF clock (FCLK) to be supplied to the FlashIF. - Generates the CAC clock (CACCLK) to be supplied to the CAC. - Generates the IWDTC-dedicated low-speed clock (IWDTCCLK) to be supplied to the IWDTC.
Operating frequencies*1	- ICLK: 48 MHz (max) - PCLKB: 48 MHz (max) - PCLKD: 64 MHz (max) - FCLK: 1 to 48 MHz (for programming and erasing the ROM and E2 DataFlash) 48 MHz (max) (for reading from the E2 DataFlash) - CACCLK: The same frequency as that of the selected measurement clock - IWDTCCLK: 15 kHz
Main clock oscillator	- Resonator frequency: 1 to 20 MHz - External clock input frequency: 20 MHz (max) - Connectable resonator or additional circuit: ceramic resonator, crystal - Connection pins: EXTAL, XTAL - Oscillation stop detection function: When a main clock oscillation stop is detected, the system clock source is switched to LOCO, and MTU and GPTW output can be forcedly driven to high-impedance. - Drive capacity switching function
PLL circuit	- Input clock source: Main clock, HOCO*2 - Input pulse frequency division ratio: Selectable from 1, 2, and 4 - Input frequency: 4 to 12.5 MHz - Frequency multiplication ratio: Selectable from 4 to 15.5 (increments of 0.5) - VCO oscillation frequency: 24 to 64 MHz
High-speed on-chip oscillator (HOCO)	Oscillation frequency: 24 MHz, 32 MHz, 48 MHz, 64 MHz
Low-speed on-chip oscillator (LOCO)	Oscillation frequency: 4 MHz
IWDTC-dedicated on-chip oscillator	Oscillation frequency: 15 kHz

Note 1. The maximum operating frequency in high-speed operating mode. For the maximum operating frequency in the other operating modes, refer to section 11.2.5, Operating Power Control Register (OPCCR).

Note 2. The specification for the PLL input frequency only allows the following frequencies for the HOCO:
12 MHz (48 MHz divided by 4), 8 MHz (32 MHz divided by 4), 12 or 6 MHz (24 MHz divided by 2 or 4).

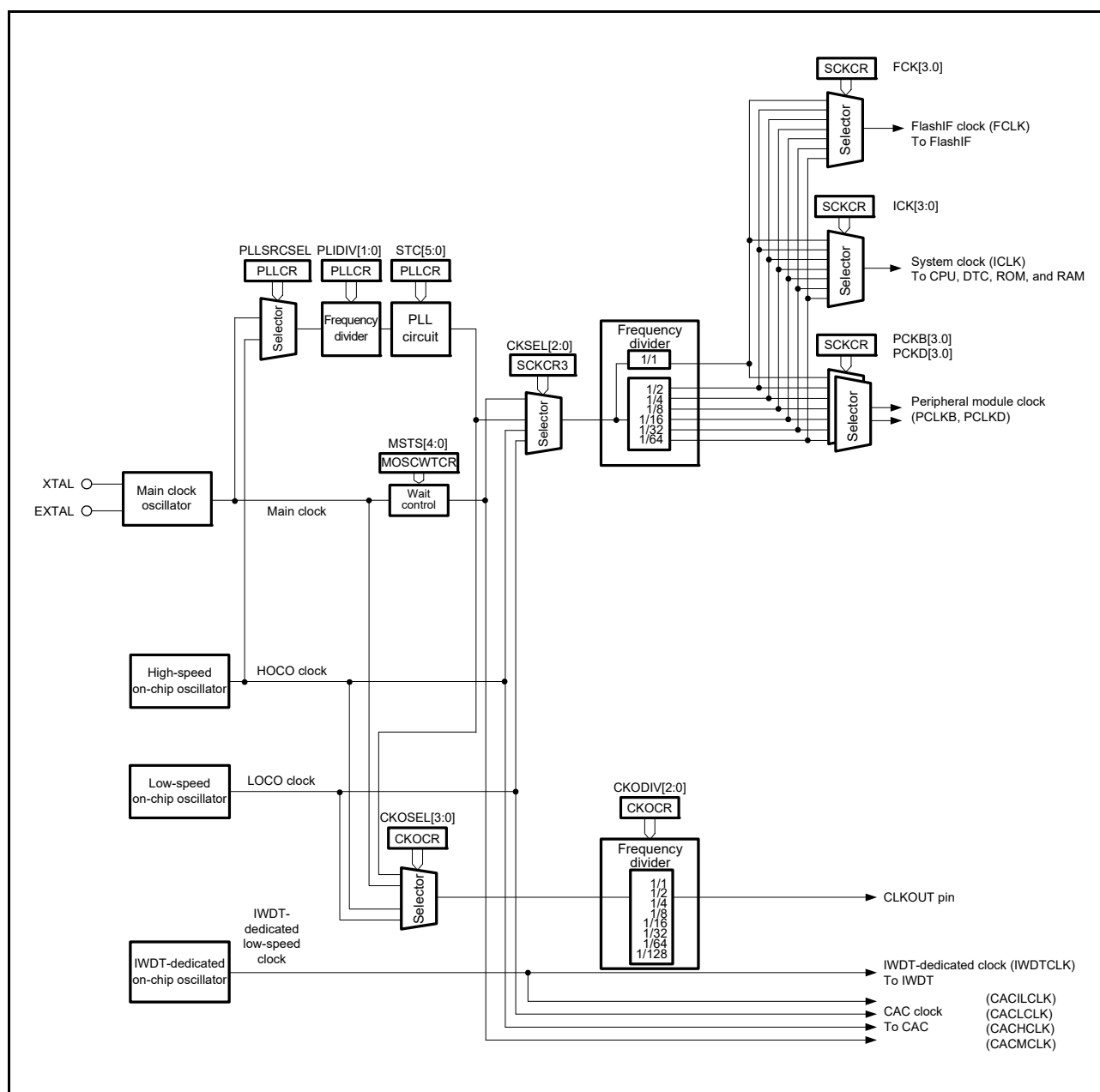


Figure 9.1 Block Diagram of Clock Generation Circuit

Table 9.2 lists the I/O pins of the clock generation circuit.

Table 9.2 I/O Pins of Clock Generation Circuit

Pin Name	I/O	Description
XTAL	Output	These pins are used to connect a crystal. The EXTAL pin can also be used to input an external clock. For details, refer to section 9.3.2, External Clock Input.
EXTAL	Input	
CLKOUT	Output	Clock output pin

9.2 Register Descriptions

9.2.1 System Clock Control Register (SCKCR)

Address(es): SYSTEM.SCKCR 0008 0020h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	FCK[3:0]				ICK[3:0]				—	—	—	—	—	—	—	—
Value after reset:	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PCKB[3:0]				—	—	—	—	PCKD[3:0]			
Value after reset:	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	PCKD[3:0]	Peripheral Module Clock D (PCLKD) Select*1	b3 b0 0 0 0 0: ×1 0 0 0 1: ×1/2 0 0 1 0: ×1/4 0 0 1 1: ×1/8 0 1 0 0: ×1/16 0 1 0 1: ×1/32 0 1 1 0: ×1/64 Settings other than above are prohibited.	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b11 to b8	PCKB[3:0]	Peripheral Module Clock B (PCLKB) Select*1, *2	b11 b8 0 0 0 0: ×1 0 0 0 1: ×1/2 0 0 1 0: ×1/4 0 0 1 1: ×1/8 0 1 0 0: ×1/16 0 1 0 1: ×1/32 0 1 1 0: ×1/64 Settings other than above are prohibited.	R/W
b23 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b27 to b24	ICK[3:0]	System Clock (ICKL) Select*2	b27 b24 0 0 0 0: ×1 0 0 0 1: ×1/2 0 0 1 0: ×1/4 0 0 1 1: ×1/8 0 1 0 0: ×1/16 0 1 0 1: ×1/32 0 1 1 0: ×1/64 Settings other than above are prohibited.	R/W
b31 to b28	FCK[3:0]	FlashIF Clock (FCLK) Select*2	b31 b28 0 0 0 0: ×1 0 0 0 1: ×1/2 0 0 1 0: ×1/4 0 0 1 1: ×1/8 0 1 0 0: ×1/16 0 1 0 1: ×1/32 0 1 1 0: ×1/64 Settings other than above are prohibited.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. The relationship of frequencies of the peripheral module clocks B and D (PCLKB and PCLKD) must be set as follows.

PCLKB:PCLKD = 1:1, 1:2, 2:1, 4:1, 8:1

Note 2. The relationship of frequencies of the system clock (ICKL), peripheral module clock B (PCLKB), and FlashIF clock (FCLK) must be set as follows.

ICKL ≥ FCLK, ICKL ≥ PCLKB

Setting examples 1: ICKL = FCLK = PCLKB = PCLKD = 48 MHz (FLASH.MEMWAITR.MEMWAIT = 1)

Setting examples 2: ICKL = FCLK = PCLKB = 32 MHz, PCLKD = 64 MHz (FLASH.MEMWAITR.MEMWAIT = 0)

For details on the MEMWAIT bit, refer to section 37, Flash Memory (FLASH).

This register cannot be rewritten while the flash memory is being programmed or erased.

When an instruction for writing to SCKCR or SCKCR3 is to follow writing to the SCKCR register, do so in accord with the procedure below.

1. Write to the SCKCR register.
2. Confirm that the value has actually been written to the SCKCR register.
3. Proceed to the next step.

PCKD[3:0] Bits (Peripheral Module Clock D (PCLKD) Select)

These bits select the frequency of peripheral module clock D (PCLKD).

PCKB[3:0] Bits (Peripheral Module Clock B (PCLKB) Select)

These bits select the frequency of peripheral module clock B (PCLKB).

ICK[3:0] Bits (System Clock (ICLK) Select)

These bits select the frequency of the system clock (ICLK).

FCK[3:0] Bits (FlashIF Clock (FCLK) Select)

These bits select the frequency of the FlashIF clock (FCLK).

9.2.2 System Clock Control Register 3 (SCKCR3)

Address(es): SYSTEM.SCKCR3 0008 0026h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	CKSEL[2:0]			—	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b10 to b8	CKSEL[2:0]	Clock Source Select	b10 b8 0 0 0: LOCO 0 0 1: HOCO 0 1 0: Main clock oscillator 1 0 0: PLL circuit Settings other than above are prohibited.	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

This register cannot be rewritten while the flash memory is being programmed or erased.

CKSEL[2:0] Bits (Clock Source Select)

These bits select the source of the system clock (ICLK), peripheral module clocks (PCLKB and PCLKD), and FlashIF clock (FCLK) from low-speed on-chip oscillator (LOCO), high-speed on-chip oscillator (HOCO), the main clock oscillator, and the PLL circuit.

Transitions to clock sources which are not in operation are prohibited.

9.2.3 PLL Control Register (PLLCR)

Address(es): SYSTEM.PLLCR 0008 0028h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	STC[5:0]					—	—	—	—	—	—	PLLSR CSEL	PLIDIV[1:0]	
Value after reset:	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	PLIDIV[1:0]	PLL Input Frequency Division Ratio Select	b1 b0 0 0: ×1 0 1: ×1/2 1 0: ×1/4 1 1: Setting prohibited	R/W
b2	PLLSRCSEL	PLL Clock Source Select	0: Main clock oscillator 1: HOCO	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13 to b8	STC[5:0]	Frequency Multiplication Factor Select	b13 b8 0 0 0 1 1 1: ×4 0 0 1 0 0 0: ×4.5 0 0 1 0 0 1: ×5 0 0 1 0 1 0: ×5.5 0 0 1 0 1 1: ×6 0 0 1 1 0 0: ×6.5 0 0 1 1 0 1: ×7 0 0 1 1 1 0: ×7.5 0 0 1 1 1 1: ×8 0 1 0 0 0 0: ×8.5 0 1 0 0 0 1: ×9 0 1 0 0 1 0: ×9.5 0 1 0 0 1 1: ×10 0 1 0 1 0 0: ×10.5 0 1 0 1 0 1: ×11 0 1 0 1 1 0: ×11.5 0 1 0 1 1 1: ×12 0 1 1 0 0 0: ×12.5 0 1 1 0 0 1: ×13 0 1 1 0 1 0: ×13.5 0 1 1 0 1 1: ×14 0 1 1 1 0 0: ×14.5 0 1 1 1 0 1: ×15 0 1 1 1 1 0: ×15.5 Settings other than above are prohibited.	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Writing to the PLLCR is prohibited when the PLLCR2.PLEN bit is 0 (PLL is operating).

PLIDIV[1:0] Bits (PLL Input Frequency Division Ratio Select)

These bits select the frequency division ratio of the PLL clock source.

Set these bits so that the frequency of PLL input signal is within the range of 4 MHz to 12.5 MHz.

PLLSRCSEL Bit (PLL Clock Source Select)

This bit selects the clock source for the PLL.

When the HOCO is selected for a clock source, do not set the HOCO frequency select bit (OFS1.HOCOFREQ[1:0]) in the option function select register 1 to 00b (the frequency is selected to 64 MHz).

STC[5:0] Bits (Frequency Multiplication Factor Select)

These bits select the frequency multiplication factor of the PLL circuit.

Set these bits so that the PLL oscillation frequency is within the range of 24 MHz to 64 MHz.

9.2.4 PLL Control Register 2 (PLLCR2)

Address(es): SYSTEM.PLLCR2 0008 002Ah

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	PLLEN
Value after reset:	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	PLLEN	PLL Stop Control	0: PLL is operating. 1: PLL is stopped.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

PLLEN Bit (PLL Stop Control)

This bit runs or stops the PLL circuit.

The PLL clock source is selectable as the main clock oscillator and HOCO.

Selecting the main clock oscillator as the PLL clock source with the PLLCR.PLLSRCSEL bit requires setting the main clock oscillator wait control register (MOSCWTCR).

After setting the PLLEN bit to 0 (PLL is operating), confirm that the OSCOVFSR.PLOVF bit is 1 before switching the system clock to the PLL clock.

That is, a fixed time for stabilization is required after the setting for PLL operation. A fixed time is also required for oscillation to stop after the setting to stop PLL operation. Accordingly, take note of the following limitations when starting and stopping PLL operation. The following notes apply when selecting the main clock oscillator as the PLL clock source.

- After stopping the PLL, confirm that the OSCOVFSR.PLOVF bit is 0 before restarting the PLL.
- Confirm that the PLL is operating and that the OSCOVFSR.PLOVF bit is 1 before stopping the PLL.
- Regardless of whether or not it is selected as the system clock, confirm that the OSCOVFSR.PLOVF bit is 1 before executing a WAIT instruction to place the MCU in software standby mode.
- After stopping the PLL, confirm that the OSCOVFSR.PLOVF bit is 0 and execute a WAIT instruction before entering software standby mode.

When the PLL clock is selected by the SCKCR3.CKSEL[2:0] bits, do not set the PLLEN bit (PLL is stopped) to 1.

9.2.5 Main Clock Oscillator Control Register (MOSCCR)

Address(es): SYSTEM.MOSCCR 0008 0032h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	MOSTP
0	0	0	0	0	0	0	1

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	MOSTP	Main Clock Oscillator Stop	0: Main clock oscillator is operating. 1: Main clock oscillator is stopped.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Set this register after setting up the main clock oscillator wait control register.

MOSTP Bit (Main Clock Oscillator Stop)

This bit runs or stops the main clock oscillator.

After setting the MOSTP bit to 0 (main clock oscillator is operating), read the OSCOVFSR.MOOVF bit to confirm that it has become 1, and then use the main clock.

For the main clock oscillator, a fixed time is required for oscillation to become stable after the settings for operation have been made. Furthermore, a fixed time is required for oscillation to actually stop after the settings to stop oscillation have been made. Accordingly, take note of the following limitations when starting and stopping operation.

- After stopping the main clock oscillator, confirm that the OSCOVFSR.MOOVF bit is 0 before restarting the main clock oscillator.
- Confirm that the main clock oscillator is operating and that the OSCOVFSR.MOOVF bit is 1 before stopping the main clock oscillator.
- Regardless of whether or not it is selected as the system clock, confirm that the OSCOVFSR.MOOVF bit is 1 and execute a WAIT instruction in order to operate the main clock oscillator and place the MCU in software standby mode.
- After stopping the main clock oscillator, confirm that the OSCOVFSR.MOOVF bit is 0 and execute a WAIT instruction before entering software standby mode.

Do not set the MOSTP bit to 1 when one of the following condition is met.

- When the main clock is selected as the clock source for the system clock (SCKCR3.CKSEL[2:0] = 010b)
- When MOSC is selected as the clock source for PLL (the PLLCR.PLLSRCSEL bit is 0) and the PLL clock is selected as the clock source for the system clock (the SCKCR3.CKSEL[2:0] bits are 100b)
- When MOSC is selected as the clock source for PLL (the PLLCR.PLLSRCSEL bit is 0) and PLL is operating (the PLLCR2.PLEN bit is 0)

9.2.6 Low-Speed On-Chip Oscillator Control Register (LOCOCR)

Address(es): SYSTEM.LOCOCR 0008 0034h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	LCSTP
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	LCSTP	LOCO Stop	0: LOCO is operating. 1: LOCO is stopped.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

LCSTP Bit (LOCO Stop)

This bit runs or stops the LOCO.

After the setting of the LCSTP bit has been changed so that the LOCO operates, only start using the LOCO clock after the LOCO clock oscillation stabilization time (t_{LOCO}) has elapsed.

That is, a fixed time for stabilization of oscillation is required after the setting for LOCO operation. A fixed time is also required for oscillation to stop after the setting to stop the oscillator. Accordingly, take note of the following limitations when starting and stopping the oscillator.

- When restarting the LOCO after it has been stopped, allow at least five cycles of the LOCO as an interval over which it is still stopped.
- Ensure that oscillation by the LOCO is stable when making the setting to stop the LOCO.
- Regardless of whether or not it is selected as the system clock, ensure that oscillation by the LOCO is stable before executing a WAIT instruction to place the chip on software standby.
- When a transition to software standby mode is to follow the setting to stop the LOCO, wait for at least three cycles of the LOCO after the setting to stop the LOCO and before executing the WAIT instruction.
- To switch between starting and stopping of the LOCO on the LCSTP bit, do so while the oscillators other than LOCO are stable or stopped.

While the LOCO is selected by the SCKCR3.CKSEL[2:0] bits, do not set the LCSTP bit to 1 (LOCO is stopped).

9.2.7 IWDT-Dedicated On-Chip Oscillator Control Register (ILOCOCR)

Address(es): SYSTEM.ILOCOCR 0008 0035h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	ILCSTP
0	0	0	0	0	0	0	1

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	ILCSTP	IWDT-Dedicated On-Chip Oscillator Stop	0: IWDT-dedicated on-chip oscillator is operating. 1: IWDT-dedicated on-chip oscillator is stopped.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

When the IWDT start mode select bit in option function select register 0 (OFS0.IWDTSTRT) is 0 (IWDT is operating), the setting of this register is invalid; it is valid only when the OFS0.IWDTSTRT bit is set to 1 (IWDT is stopped). The ILCSTP bit cannot be changed from 0 (IWDT-dedicated on-chip oscillator is operating) to 1 (IWDT-dedicated on-chip oscillator is stopped) while ILOCOCR is valid.

ILCSTP Bit (IWDT-Dedicated On-Chip Oscillator Stop)

This bit runs or stops the IWDT-dedicated on-chip oscillator.

After the setting of the ILCSTP bit has been changed so that the IWDT-dedicated on-chip oscillator operates, supply of the clock is started the MCU internally after a fixed time corresponding to the IWDT-dedicated clock oscillation stabilization time (t_{ILOCO}) has elapsed.

If the IWDT-dedicated clock is to be used, only start using the oscillator after this wait time has elapsed.

Ensure that oscillation by the IWDT-dedicated on-chip oscillator is stable before executing a WAIT instruction to place the chip on software standby mode.

9.2.8 High-Speed On-Chip Oscillator Control Register (HOCOOCR)

Address(es): SYSTEM.HOCOOCR 0008 0036h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	HCSTP
0	0	0	0	0	0	0	0/1**

Bit	Symbol	Bit Name	Description	R/W
b0	HCSTP	HOCO Stop	0: HOCO is operating. 1: HOCO is stopped.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Use the OFS1.HOCOFREQ[1:0] bits to set the frequency of the HOCO.

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. The HCSTP bit value after a reset is 0 when the HOCO oscillation enable bit in option function select register 1 (OFS1.HOCOEN) is 0. The HCSTP bit value after a reset is 1 when the OFS1.HOCOEN bit is 1.

HCSTP Bit (HOCO Stop)

This bit runs or stops the HOCO.

When changing the HCSTP bit from 1 to 0 (i.e. changing the HOCO clock from stopped to operating), confirm that the OSCOVFSR.HCOVF flag is 1 before switching the system clock to the HOCO clock.

That is, a fixed time for stabilization of oscillation is required after the setting for HOCO operation. A fixed time is also required for oscillation to stop after the setting to stop the oscillator. Accordingly, take note of the following limitations when starting and stopping the oscillator.

- After stopping the HOCO, confirm that the OSCOVFSR.HCOVF flag is 0 before restarting the HOCO.
- Confirm that the HOCO is operating and that the OSCOVFSR.HCOVF flag is 1 before stopping the HOCO.
- Regardless of whether or not it is selected as the system clock, confirm that the OSCOVFSR.HCOVF flag is 1 before executing a WAIT instruction to place the MCU in software standby mode.
- After stopping the HOCO, confirm that the OSCOVFSR.HCOVF flag is 0 and execute a WAIT instruction before entering software standby mode.
- Make sure that the ROM and E2 DataFlash are placed in the programming and erasing mode while the high-speed on-chip oscillator is in operation.

While the HOCO is selected by the SCKCR3.CKSEL[2:0] bits, do not set the HCSTP bit to 1 (HOCO is stopped).

Writing 1 to the HCSTP bit (the HOCO is stopped) is prohibited when the HOCO is selected as the clock source for PLL (the PLLCR.PLLSRCSEL bit is 1) and the PLL clock is selected as the clock source for the system clock (the SCKCR3.CKSEL[2:0] bits are 100b), and the HOCO is selected as the clock source for PLL (the PLLCR.PLLSRCSEL bit is 1) and while the PLL is operating (the PLLCR2.PLEN bit is 0).

9.2.9 Oscillation Stabilization Flag Register (OSCOVFSR)

Address(es): SYSTEM.OSCOVFSR 0008 003Ch

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	HCOVF	PLOVF	—	MOOV F
Value after reset:	0	0	0	0	0/1*1	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MOOVF	Main Clock Oscillation Stabilization Flag	0: Main clock is stopped 1: Oscillation is stable and the clock can be used as the system clock*2	R
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	PLOVF	PLL Clock Oscillation Stabilization Flag	0: PLL is stopped or not stabilized 1: Oscillation is stable and the clock can be used as the system clock	R
b3	HCOVF	HOCO Clock Oscillation Stabilization Flag	0: HOCO is stopped or not stabilized 1: Oscillation is stable and the clock can be used as the system clock	R
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The HCOVF flag value after a reset is 1 when the HOCO oscillation enable bit in option function selection register 1 (OFS1.HOCOEN) is 0. The HCOVF flag value after a reset is 1 when the OFS1.HOCOEN bit is 0.

Note 2. When an appropriate value is set in the wait control register for each oscillator. If a set value (wait time) is not adequate, clock supply starts before oscillation becomes stable.

The OSCOVFSR register monitors whether oscillation of each oscillator has become stable.

If a wait control register is provided for each oscillator, specify a wait time that is longer than or equal to the stabilization time of the corresponding oscillation circuit.

MOOVF Flag (Main Clock Oscillation Stabilization Flag)

This flag indicates whether oscillation of the main clock is stable.

[Setting condition]

- After the MOSCCR.MOSTP bit is set to 0 (main clock oscillator is operating) when the MOSTP bit is 1 (main clock oscillator is stopped), the corresponding time set in the MOSCWTCR register has elapsed and supply of the main clock is started to the MCU internally.

[Clearing condition]

- After the MOSCCR.MOSTP bit is set to 1, the processing to stop the oscillation of the main clock oscillator is completed.

PLOVF Flag (PLL Clock Oscillation Stabilization Flag)

This flag indicates whether oscillation of the PLL clock is stable.

[Setting condition]

After the PLLCR2.PLEN is set to 0 (PLL is operating) when the PLEN bit is 1 (PLL is stopped), the MOOVF flag becomes 1, the PLL clock oscillation stabilization time (t_{PLL}) has elapsed, and supply of the PLL clock is started to the MCU internally.

However, if oscillation by the PLL clock source selected in the PLLCR.PLLSRCSEL bit is not stable when the PLEN bit is set to 0, counting by the LOCO clock counter proceeds after the oscillation of the PLL clock source has become stable.

[Clearing condition]

After the PLLCR2.PLEN bit is set to 1, the processing to stop the oscillation of the PLL is completed.

HCOVF Flag (HOCO Clock Oscillation Stabilization Flag)

This flag indicates whether oscillation of the HOCO clock is stable.

[Setting condition]

- After the HOCOCR.HCSTP bit is set to 0 (HOCO is operating) when the HCSTP bit is 1 (HOCO is stopped), supply of the HOCO clock is started to the MCU internally.

[Clearing condition]

- After the HOCOCR.HCSTP bit is set to 1, the processing to stop the oscillation of the HOCO is completed.

9.2.10 Oscillation Stop Detection Control Register (OSTDCR)

Address(es): SYSTEM.OSTDCR 0008 0040h

	b7	b6	b5	b4	b3	b2	b1	b0
	OSTDE	—	—	—	—	—	—	OSTDIE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OSTDIE	Oscillation Stop Detection Interrupt Enable	0: The oscillation stop detection interrupt is disabled. Oscillation stop detection is not notified to the POE and POEG. 1: The oscillation stop detection interrupt is enabled. Oscillation stop detection is notified to the POE and POEG.	R/W
b6 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	OSTDE	Oscillation Stop Detection Function Enable	0: Oscillation stop detection function is disabled. 1: Oscillation stop detection function is enabled.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

OSTDIE Bit (Oscillation Stop Detection Interrupt Enable)

If the oscillation stop detection flag in the oscillation stop detection status register (OSTDSR.OSTDF) requires clearing, do this after setting the OSTDIE bit to 0. Wait for at least two cycles of PCLKB before again setting the OSTDIE bit to 1. According to the number of cycles for access to read a given I/O register, wait time longer than two cycles of PCLKB may have to be secured.

OSTDE Bit (Oscillation Stop Detection Function Enable)

This bit enables or disables the oscillation stop detection function.

When the OSTDE bit is 1 (oscillation stop detection function enabled), the LOCO stop bit (LOCOCR.LCSTP) is set to 0 and the LOCO operation is started. The LOCO cannot be stopped while the oscillation stop detection function is enabled; writing 1 (LOCO is stopped) to the LOCOCR.LCSTP bit is invalid.

When the oscillation stop detection flag in the oscillation stop detection status register (OSTDSR.OSTDF) is 1 (main clock oscillation stop has been detected), writing 0 to the OSTDE bit is invalid.

When the OSTDE bit is 1, a transition cannot be made to software standby mode. To make a transition to software standby mode, execute the WAIT instruction with the OSTDE bit being 0.

9.2.11 Oscillation Stop Detection Status Register (OSTDSR)

Address(es): SYSTEM.OSTDSR 0008 0041h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	OSTDF
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	OSTDF	Oscillation Stop Detection Flag	0: The main clock oscillation stop has not been detected. 1: The main clock oscillation stop has been detected.	R/(W) *1
b7 to b1	—	Reserved	These bits are read as 0 and cannot be modified.	R

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. This bit can only be set to 0.

OSTDF Flag (Oscillation Stop Detection Flag)

This bit is a flag to indicate the main clock status. When the OSTDF flag is 1, it indicates that the main clock oscillation stop has been detected.

Once the main clock oscillation stop is detected, the OSTDF flag is not set to 0 even though the main clock oscillation is restarted. The OSTDF flag is set to 0 by reading 1 from the bit and then writing 0. At least three ICLK cycles of wait time is necessary between writing 0 to the OSTDF flag and reading the OSTDF flag as 0. If the OSTDF flag is set to 0 while the main clock oscillation is stopped, the OSTDF flag becomes 0 and then returns to 1.

When the main clock oscillator (010b) or PLL (100b) is selected by the clock source select bits in system clock control register 3 (SCKCR3.CKSEL[2:0]), the OSTDF flag cannot be modified to 0. The OSTDF flag should be set to 0 after switching the clock source to a source other than the main clock oscillator and the PLL.

[Setting condition]

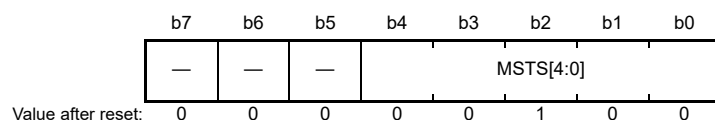
- The main clock oscillation is stopped with the OSTDCR.OSTDE bit being 1 (oscillation stop detection function enabled).

[Clearing condition]

- 1 is read and then 0 is written when the SCKCR3.CKSEL[2:0] bits are neither 010b nor 100b.

9.2.12 Main Clock Oscillator Wait Control Register (MOSCWTCR)

Address(es): SYSTEM.MOSCWTCR 0008 00A2h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	MSTS[4:0]	Main Clock Oscillator Wait Time	b4 b0 0 0 0 0: Wait time = 0 cycles (0 μ s) 0 0 0 1: Wait time = 1024 cycles (256 μ s) 0 0 1 0: Wait time = 2048 cycles (512 μ s) 0 0 1 1: Wait time = 4096 cycles (1.024 ms) 0 1 0 0: Wait time = 8192 cycles (2.048 ms) 0 1 0 1: Wait time = 16384 cycles (4.096 ms) 0 1 1 0: Wait time = 32768 cycles (8.192 ms) 0 1 1 1: Wait time = 65536 cycles (16.384 ms) 0 1 0 0: Wait time = 131072 cycles (32.768 ms) Settings other than above are prohibited. Wait time when LOCO = 4.0 MHz (0.25 μ s, TYP.)	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

MSTS[4:0] Bits (Main Clock Oscillator Wait Time)

Set these bits to select the oscillation stabilization wait time of the main clock oscillator.

Set the main clock oscillation stabilization time to longer than or equal to the stabilization time recommended by the oscillator manufacturer. When the main clock is externally input, set these bits to 00000b because the oscillation stabilization time is not required.

The wait time set by the MSTS[4:0] bits is counted using the LOCO clock. The LOCO automatically oscillates when necessary, regardless of the value of the LOCOCR.LCSTP bit.

After the set wait time has elapsed, supply of the main clock is started to the MCU internally and the OSCOVFSR.MOOVF flag becomes 1. If the set wait time is short, supply of the main clock is started before oscillation of the clock becomes stable.

Only rewrite the MOSCWTCR register when the MOSCCR.MOSTP bit is 1 and the OSCOVFSR.MOOVF flag is 0. Do not rewrite this register under any other conditions.

9.2.13 CLKOUT Output Control Register (CKOCR)

Address(es): SYSTEM.CKOCR 0008 003Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CKOSTP	CKODIV[2:0]			CKOSEL[3:0]				—	—	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b11 to b8	CKOSEL[3:0]	CLKOUT Output Source Select	b11 b8 0 0 0 0: LOCO clock 0 0 0 1: HOCO clock 0 0 1 0: Main clock oscillator 0 1 0 0: PLL Settings other than above are prohibited.	R/W
b14 to b12	CKODIV[2:0]	CLKOUT Output Division Ratio Select	b14 b12 0 0 0: No division 0 0 1: ×1/2 0 1 0: ×1/4 0 1 1: ×1/8 1 0 0: ×1/16 1 0 1: ×1/32 1 1 0: ×1/64 1 1 1: ×1/128	R/W
b15	CKOSTP	CLKOUT Output Stop Control	0: CLKOUT pin output enabled*1 1: CLKOUT pin output disabled	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. It is also necessary to set the pin function control register and port mode register for the corresponding pin.

CKOSEL[3:0] Bits (CLKOUT Output Source Select)

Set these bits to select the LOCO clock, HOCO clock, main clock, or PLL as the source of the clock to be output from the CLKOUT pin.

CKODIV[2:0] Bits (CLKOUT Output Division Ratio Select)

Set these bits to select the clock division ratio.

Set the CKOSTP bit to 1 when changing the division ratio.

For details on the characteristics of the clock output from the CLKOUT pin, see section 38.5.5.11, CLKOUT.

The division ratio of the output clock frequency should be set in accord with the specification for the CLKOUT pin output cycle.

CKOSTP Bit (CLKOUT Output Stop Control)

Set this bit to enable or disable output from the CLKOUT pin.

When this bit is set to 1, the selected clock is output. When this bit is set to 0, a low level is output.

If the CKOSTP bit is rewritten while the clock is still oscillating, a glitch may be generated in the output.

9.2.14 Main Clock Oscillator Forced Oscillation Control Register (MOFCR)

Address(es): SYSTEM.MOFCR 0008 C293h

b7	b6	b5	b4	b3	b2	b1	b0
—	MOSEL	MODRV21	—	—	—	—	—
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b5	MODRV21	Main Clock Oscillator Drive Capability Switch	0: 1 MHz to 10 MHz 1: 10 MHz to 20 MHz	R/W
b6	MOSEL	Main Clock Oscillator Switch	0: Resonator 1: External oscillator input	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

The EXTAL/XTAL pin is also used as a port. In the initial setting state, the pin is set as a port.

Only write new values to the MOFCR register when the setting of the main clock oscillator stop bit (MOSCCR.MOSTP) is 1 (stopping the main clock oscillator).

MODRV21 Bit (Main Clock Oscillator Drive Capability Switch)

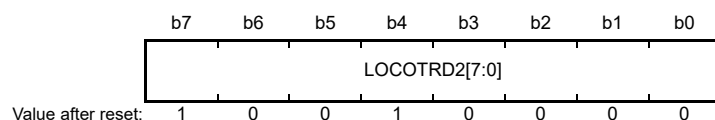
These bits select the drive capability of the main clock oscillator.

MOSEL Bit (Main Clock Oscillator Switch)

This bit selects the oscillation source of the main clock oscillator.

9.2.15 Low-Speed On-Chip Oscillator Trimming Register 2 (LOCOTRR2)

Address(es): SYSTEM.LOCOTRR2 0008 0061h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	LOCOTRD2[7:0]	Low-Speed On-Chip Oscillator Frequency Adjustment 2	b7 b0 0 0 0 0 0 0 0 0: 0 (Frequency: Low) 0 0 0 0 0 0 0 1: 1 : 1 1 1 1 1 1 1 0: 254 1 1 1 1 1 1 1 1: 255 (Frequency: High)	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

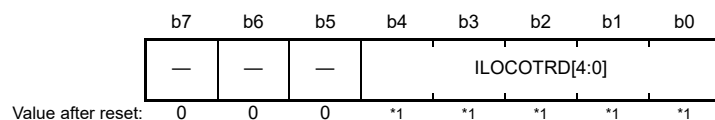
LOCOTRD2[7:0] Bits (Low-Speed On-Chip Oscillator Frequency Adjustment 2)

Set the frequency adjustment value for the low-speed on-chip oscillator.

The setting range is from 0 (00h) to 255 (FFh) by binary numbers. The greater the set value is, the higher the frequency is.

9.2.16 IWDT-Dedicated On-Chip Oscillator Trimming Register (ILOCOTRR)

Address(es): SYSTEM.ILOCOTRR 0008 0064h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	ILOCOTRD [4:0]	IWDT-Dedicated On-Chip Oscillator Frequency Adjustment	b4 b0 0 0 0 0 0: 0 (Frequency: Low) 0 0 0 0 1: 1 : 1 1 1 1 0: 30 1 1 1 1 1: 31 (Frequency: High)	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. Chip-specific value

ILOCOTRD[4:0] Bits (IWDT-Dedicated On-Chip Oscillator Frequency Adjustment)

Set the frequency adjustment value for the IWDT-dedicated on-chip oscillator.

The setting range is from 0 (00h) to 31 (1Fh) by binary numbers. The greater the set value is, the higher the frequency is. The frequency is adjusted under certain conditions before shipment, so the value after reset varies with the chip. After a reset, the oscillation frequency returns to the factory default.

9.2.17 High-Speed On-Chip Oscillator Trimming Register 0 (HOCOTRR0)

Address(es): SYSTEM.HOCOTRR0 0008 0068h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	HOCOTRD[5:0]	High-Speed On-Chip Oscillator Frequency Adjustment	b5 b0 0 0 0 0 0: 0 (Frequency: Low) 0 0 0 0 1: 1 : : 1 1 1 1 0: 62 1 1 1 1 1: 63 (Frequency: High)	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

Note 1. Chip-specific value

HOCOTRD[5:0] Bits (High-Speed On-Chip Oscillator Frequency Adjustment)

Set the frequency adjustment value for the high-speed on-chip oscillator.

The setting range is from 0 (00h) to 63 (3Fh) by binary numbers. The greater the set value is, the higher the frequency is.

The frequency is adjusted under certain conditions before shipment, so the value after reset varies with the chip. After a reset, the oscillation frequency returns to the factory default.

9.3 Main Clock Oscillator

There are two ways of supplying the clock signal from the main clock oscillator: connecting an oscillator or the input of an external clock signal.

9.3.1 Connecting a Crystal

Figure 9.2 shows an example of connecting a crystal.

A damping resistor (R_d) should be added, if necessary. Since the resistor values vary depending on the resonator and the oscillation drive capability, use values recommended by the resonator manufacturer. If use of an external feedback resistor (R_f) is directed by the resonator manufacturer, insert an R_f between EXTAL and XTAL by following the instruction.

When connecting a resonator to supply the clock, the frequency of the resonator should be in the frequency range of the resonator for the main clock oscillator described in Table 9.1.

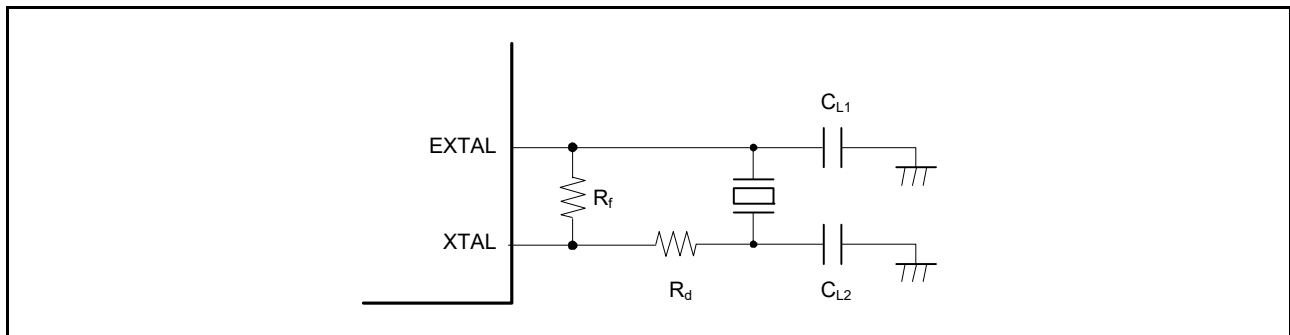


Figure 9.2 Example of Crystal Connection

9.3.2 External Clock Input

Figure 9.3 shows connection of an external clock. Set the MOFCR.MOSEL bit to 1 to operate the oscillator by inputting an external clock signal. The XTAL pin becomes high impedance and can be used as an input port.

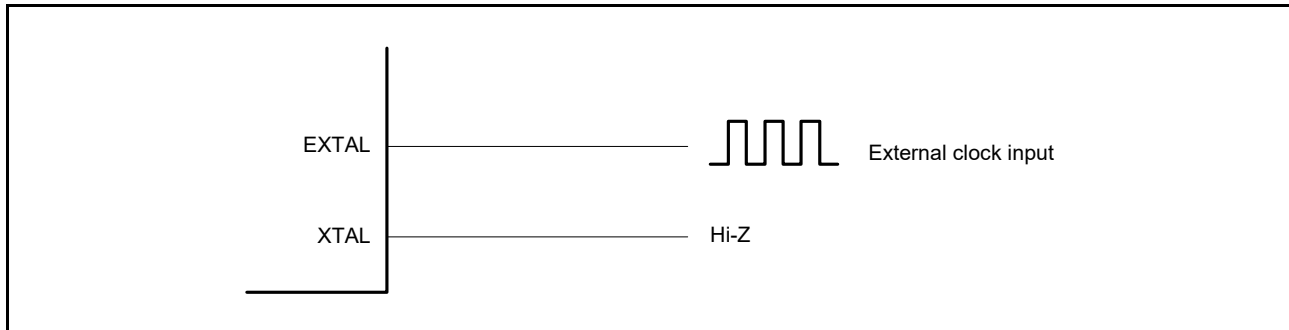


Figure 9.3 Connection Example of External Clock

9.3.3 Handling of Pins When the Main Clock is Not Used

For details on pin handling when the main clock is not used, refer to section 17.5, Handling of Unused Pins.

9.3.4 Notes on the External Clock Input

The frequency of the external clock input can only be changed while the main clock oscillator is stopped. Do not change the frequency of the external clock input while the setting of the main clock oscillator stop bit (MOSCCR.MOSTP) is 0 (main clock oscillator is operating).

9.4 Oscillation Stop Detection Function

9.4.1 Oscillation Stop Detection and Operation after Detection

The oscillation stop detection function is used to detect the main clock oscillator stop and to supply LOCO clock pulses from the low-speed on-chip oscillator as the system clock source instead of the main clock.

When the HOCO is selected as the clock source for the PLL and the PLL clock is selected as the system clock, the system clock is not switched to the LOCO even if stopping of the main clock is detected.

An oscillation stop detection interrupt request can be generated when an oscillation stop is detected. In addition, the MTU and GPTW output can be forcedly driven to the high-impedance on the detection. For details, refer to [section 19, Multi-Function Timer Pulse Unit 3 \(MTU3c\)](#), [section 20, Port Output Enable 3 \(POE3E\)](#), and [section 22, GPTW Port Output Enable \(POEGb\)](#).

In the MCU, the main clock oscillation stop is detected when the input clock remains to be 0 or 1 for a certain period, for example, due to a malfunction of the main clock oscillator (refer to [section 38, Electrical Characteristics](#)).

When an oscillation stop is detected, the main clock selected by the clock source select bits (SCKCR3.CKSEL[2:0]) is switched to the LOCO clock by the corresponding selectors in the former stage. Therefore, if an oscillation stop is detected with the main clock selected as the system clock source, the system clock source is switched to the LOCO clock without a change of CKSEL[2:0].

If an oscillation stop is detected while the PLL clock is selected by the clock source select bits (SCKCR3.CKSEL[2:0]) in system clock control register 3, the SCKCR3.CKSEL[2:0] bit value does not change and the PLL clock remains the system clock source. However, the frequency becomes a free-running oscillation frequency.

Switching between the main clock and LOCO clock is controlled by the oscillation stop detection flag (OSTDSR.OSTDF). The clock source is switched to the LOCO clock when the OSTDF flag is 1, and is switched to the main clock again when the OSTDF flag is set to 0. At this time, if the main clock or PLL clock is selected with the CKSEL[2:0] bits, the OSTDF flag cannot be set to 0. To switch the clock source to the main clock or PLL clock again after the oscillation stop detection, set the CKSEL[2:0] bits to a clock source other than the main clock or PLL clock and set the OSTDF flag to 0. After that, check that the OSTDF flag is not 1, and then set the CKSEL[2:0] bits to the main clock or PLL clock after the specified oscillation stabilization time has elapsed.

After a reset is released, the main clock oscillator is stopped and the oscillation stop detection function is disabled. To enable the oscillation stop detection function, activate the main clock oscillator and write 1 to the oscillation stop detection function enable bit (OSTDCR.OSTDE) after a specified oscillation stabilization time has elapsed.

The oscillation stop detection function is provided against the main clock stop by an external cause. Therefore, the oscillation stop detection function should be disabled before the main clock oscillator is stopped by the software or a transition is made to software standby mode.

When the system clock with the main clock selected as the system clock source and CAC main clock (CACMCLK) are selected, these clocks are switched to the LOCO clock by the oscillation stop detection.

The system clock (ICLK) frequency during the LOCO clock operation is specified by the LOCO oscillation frequency and the division ratio set by the system clock (ICLK) select bits (SCKCR.ICK[3:0]).

When the PLL clock is selected as the system clock source, this clock operate at the PLL free-running frequency by the oscillation stop detection.

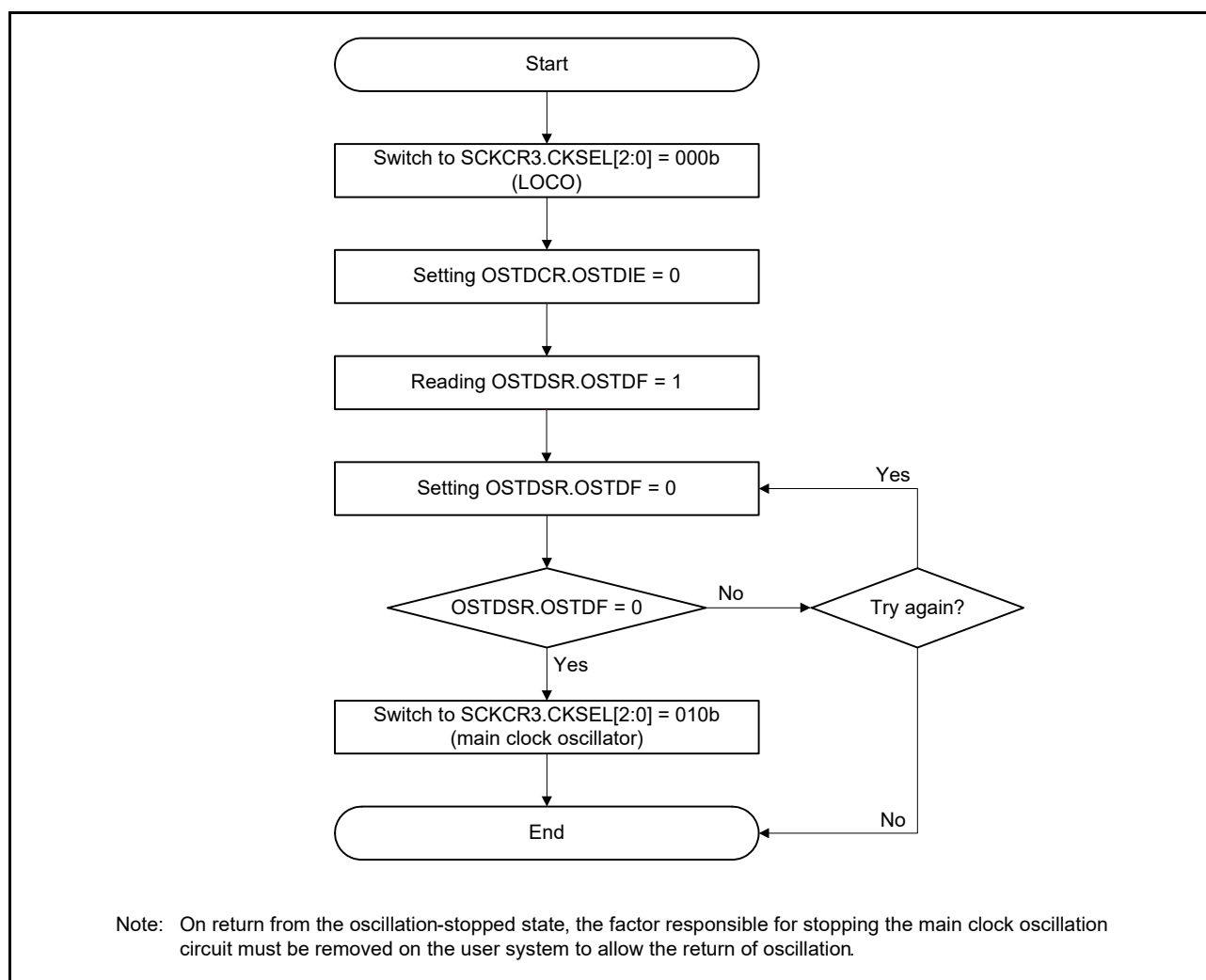


Figure 9.4 Flow of Recovery from Detection of Oscillator Stop

9.4.2 Oscillation Stop Detection Interrupts

An oscillation-stop detection interrupt will be generated if the oscillation-stop detection flag (OSTDSR.OSTDF) becomes 1 while the oscillation-stop detection interrupt enable bit (OSTDCR.OSTDIE) is 1 (oscillation stop detection interrupt enabled). At this time, the main clock oscillator stop is notified to the port output enable (POE) and GPTW port output enable (POEG). On accepting the notification of the oscillation stop, the POE sets the OSTST high-impedance flag in input level control/status register 6 (ICSR6.OSTSTF) to 1. After the oscillation stop is detected, wait for at least 10 cycles of PCLKB before writing to this ICSR6.OSTSTF flag.

Wait at least three cycles of PCLKB before clearing the oscillation stop detection interrupt enable bit (OSTDCR.OSTDIE) to 0 after the oscillation stop detection flag (OSTDSR.OSTDF) has been set to 1.

When the OSTDSR.OSTDF flag requires clearing, do so setting the oscillation stop detection interrupt enable bit (OSTDCR.OSTDIE) to 0. Wait for at least two cycles of PCLKB clock before again setting the OSTDCR.OSTDIE bit to 1. According to the number of cycles for access to read a given I/O register, wait time longer than two cycles of PCLKB may have to be secured.

While at the same time, on accepting the notification of the oscillation stop, the POEG sets the oscillation stop detection flag (POEGGA.OSTPF) in the POEG group A setting register to 1.

The oscillation stop detection interrupt is a non-maskable interrupt. Since non-maskable interrupts are disabled in the initial state after a reset release, enable the non-maskable interrupts by the software before using oscillation stop detection interrupts. For details, refer to section 14, Interrupt Controller (ICUb).

9.5 PLL Circuit

The PLL circuit has a function to multiply the frequency from the oscillator.

9.6 Internal Clock

Clock sources of internal clock signals are the main clock, HOCO clock, LOCO clock, PLL clock, and dedicated low-speed clock for the IWDI. The internal clocks listed below are produced from these sources.

- (1) Operating clock of the CPU, DTC, ROM, and RAM: System clock (ICLK)
- (2) Operating clocks of peripheral modules: Peripheral module clocks (PCLKB and PCLKD)
- (3) Operating clock of the FlashIF: FlashIF clock (FCLK)
- (4) Operating clock for the CAC: CAC clock (CACCLK)
- (5) Operating clock for the IWDI: IWDI-dedicated low-speed clock (IWDICLK)

Frequencies of the internal clocks are set by the combination of the divisors selected by the SCKCR.FCK[3:0], ICK[3:0], PCKB[3:0], and PCKD[3:0] bits, the clock source selected by the SCKCR3.CKSEL[2:0] bits, the bits that select the frequency of the PLL circuit (the PLLCR.STC[5:0] and PLIDIV[1:0] bits), and the bits that select the frequency of the HOCO circuit (the OFS1.HOCOFREQ[1:0] bits). If the value of any of these bits is changed, subsequent operation will be at the frequency determined by the new value.

If the clock with frequency higher than 32 MHz is to be selected as the system clock (ICLK) and operating clock of the FlashIF (FCLK), the number of wait cycles to insert must be set for the ROM access. [section 37, Flash Memory \(FLASH\)](#).

9.6.1 System Clock

The system clock (ICLK) is used as the operating clock of the CPU, DTC, ROM, and RAM.

The ICLK frequency is specified by the SCKCR.ICK[3:0] bits, and the SCKCR3.CKSEL[2:0] bits, the PLLCR.STC[5:0] and PLIDIV[1:0] bits, the OFS1.HOCOFREQ[1:0] bits.

9.6.2 Peripheral Module Clock

The peripheral module clocks (PCLKB and PCLKD) are the operating clocks for use by peripheral modules.

The PCLKB and PCLKD frequencies are specified by the SCKCR.PCKB[3:0] and PCKD[3:0] bits, the SCKCR3.CKSEL[2:0] bits, the PLLCR.STC[5:0] and PLIDIV[1:0] bits, the OFS1.HOCOFREQ[1:0] bits.

The peripheral module clock used as the operating clock is PCLKD for S12AD, and PCLKB are for other modules.

9.6.3 FlashIF Clock

The FlashIF clock (FCLK) is used as the operating clock of the FlashIF.

The FCLK frequency is specified by the SCKCR.FCK[3:0] bits, and the SCKCR3.CKSEL[2:0] bits, the PLLCR.STC[5:0] and PLIDIV[1:0] bits, the OFS1.HOCOFREQ[1:0] bits.

9.6.4 CAC Clock

The CAC clock (CACCLK) is an operating clock for the CAC module.

The CACCLK clocks include CACMCLK which is generated by the main clock oscillator, CACHCLK which is generated by the high-speed on-chip oscillator, CACLCLK which is generated by the low-speed on-chip oscillator, and CACILCLK which is generated by the IWDt-dedicated on-chip oscillator.

9.6.5 IWDt-Dedicated Clock

The IWDt-dedicated clock (IWDtCLK) is the operating clock for the IWDt.

IWDtCLK is internally generated by the IWDt-dedicated on-chip oscillator.

9.7 Usage Notes

9.7.1 Notes on Clock Generation Circuit

- (1) The frequencies of the system clock (ICLK), peripheral module clocks (PCLKB and PCLKD), and FlashIF clock (FCLK) supplied to each module can be selected by the SCKCR register. Each frequency should meet the following:
 Select each frequency that is within the operation guaranteed range of clock cycle time (tcyc) specified in AC characteristics of electrical characteristics.
 The frequencies must not exceed the ranges listed in Table 9.1.
 The peripheral modules operate on the PCLKB and PCLKD. Note therefore that the operating speed of modules such as the timer and SCI varies before and after the frequency is changed.
- (2) The relationship of frequencies of the system clock (ICLK), peripheral module clocks B and D (PCLKB and PCLKD), and FlashIF clock (FCLK) must be set as follows.

$$\text{ICLK} \geq \text{FCLK},$$

$$\text{ICLK} \geq \text{PCLKB},$$

$$\text{PCLKB}:\text{PCLKD} = 1:1, 1:2, 2:1, 4:1, 8:1$$
 Setting examples 1: ICLK = FCLK = PCLKB = PCLKD = 48 MHz (FLASH.MEMWAITR.MEMWAIT = 1)
 Setting examples 2: ICLK = FCLK = PCLKB = 32 MHz, PCLKD = 64 MHz (FLASH.MEMWAITR.MEMWAIT = 0)
 For details on the MEMWAIT bit, refer to section 37, Flash Memory (FLASH).
- (3) To secure the processing after the clock frequency is changed, modify the pertinent clock control register to change the frequency, and then read the value from the register, and then perform the subsequent processing.

9.7.2 Note on Rewriting the SCKCR3 Register

When the SCKCR3.CKSEL[2:0] bits have been rewritten, the clock output is temporarily stopped to prevent the switch of the clock source from generating a clock pulse of short duration (glitch). The interrupt controller may not detect the following signals that were input during this period.

- (1) An external pin interrupt or NMI pin interrupt with a pulse width shorter than 4 cycles of the PCLKB after the switch while the PCLKB frequency is the 1/1 of the clock source (the SCKCR.PCKB[3:0] bits are 0000b).
- (2) An external pin interrupt or NMI pin interrupt with a pulse width shorter than 2.5 cycles of the PCLKB after the switch while the PCLKB frequency is the 1/2 of the clock source (the SCKCR.PCKB[3:0] bits are 0001b).

When the external pin interrupt or NMI pin interrupt is in use, input the signal with enough pulse width to exceed the time condition described in (1) and (2).

9.7.3 Notes on Resonator

Since various resonator characteristics relate closely to the user's board design, adequate evaluation is required on the user side before use, referencing the resonator connection example shown in this section. The circuit constants for the resonator depend on the resonator to be used and the stray capacitance of the mounting circuit. Therefore, the circuit constants should be determined in full consultation with the resonator manufacturer. The voltage to be applied between the resonator pins must be within the absolute maximum rating.

9.7.4 Notes on Board Design

When using a crystal, place the resonator and its load capacitors as close to the XTAL and EXTAL pins as possible. Other signal lines should be routed away from the oscillation circuit as shown in Figure 9.5 to prevent electromagnetic induction from interfering with correct oscillation.

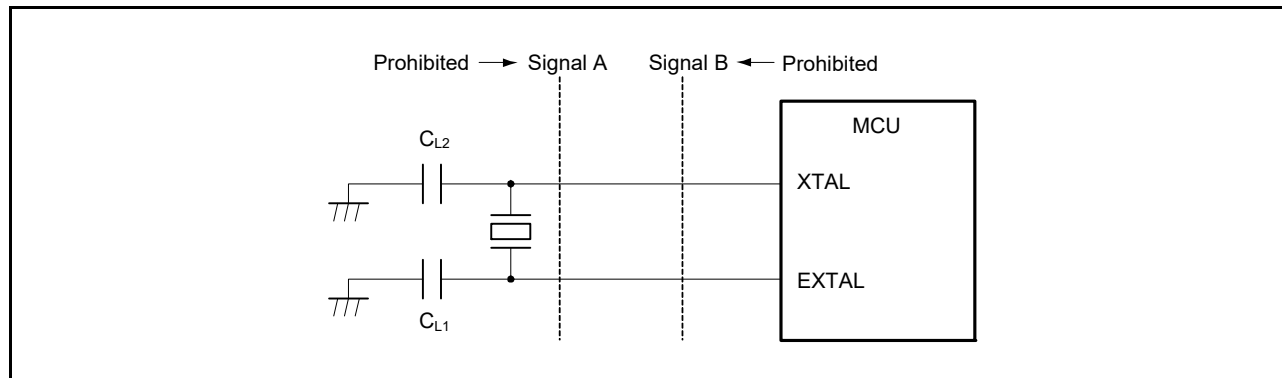


Figure 9.5 Notes on Board Design for Oscillation Circuit

9.7.5 Notes on Resonator Connection Pins

When the main clock is not used, the EXTAL and XTAL pins can be used as general input/output ports P36 and P37 or peripheral function input/output pins. When using these pins as general input/output ports or peripheral function input/output pins, be sure to stop the main clock (MOSCCR.MOSTP = 1). However, do not use the EXTAL and XTAL pins as general input/output ports P36 and P37 or peripheral function input/output pins in a system that uses the main clock. When the main clock is used, do not set P36 and P37 pins to output ports.

Note that the XTAL pin can be used as the P37 input port pin or peripheral function input pin when the settings of the MOFCR.MOSEL bit and MOSCCR.MOSTP bit are respectively 1 and 0, selecting operation with the external clock being input to the EXTAL pin.

10. Clock Frequency Accuracy Measurement Circuit (CAC)

This MCU incorporates a clock frequency accuracy measurement circuit (CAC).

The CAC counts pulses of the clock to be measured (measurement target clock) within the time generated by the clock to be used as a measurement reference (measurement reference clock), and determines the accuracy depending on whether the number of pulses is within the allowable range.

When measurement is completed or the number of pulses within the time generated by the measurement reference clock is not within the allowable range, an interrupt request is generated.

10.1 Overview

Table 10.1 lists the specifications of the CAC and Figure 10.1 shows a block diagram of the CAC.

Table 10.1 CAC Specifications

Item	Description
Measurement target clocks	The frequency of the following clocks can be measured. • Main clock • HOCO clock • LOCO clock • IWDT-dedicated clock (IWDTCCLK) • Peripheral module clock B (PCLKB)
Measurement reference clocks	• External clock input to the CACREF pin • Main clock • HOCO clock • LOCO clock • IWDT-dedicated clock (IWDTCCLK) • Peripheral module clock B (PCLKB)
Selectable function	Digital filter function
Interrupt sources	• Measurement end interrupt • Frequency error interrupt • Overflow interrupt
Low power consumption function	Module stop state can be set.

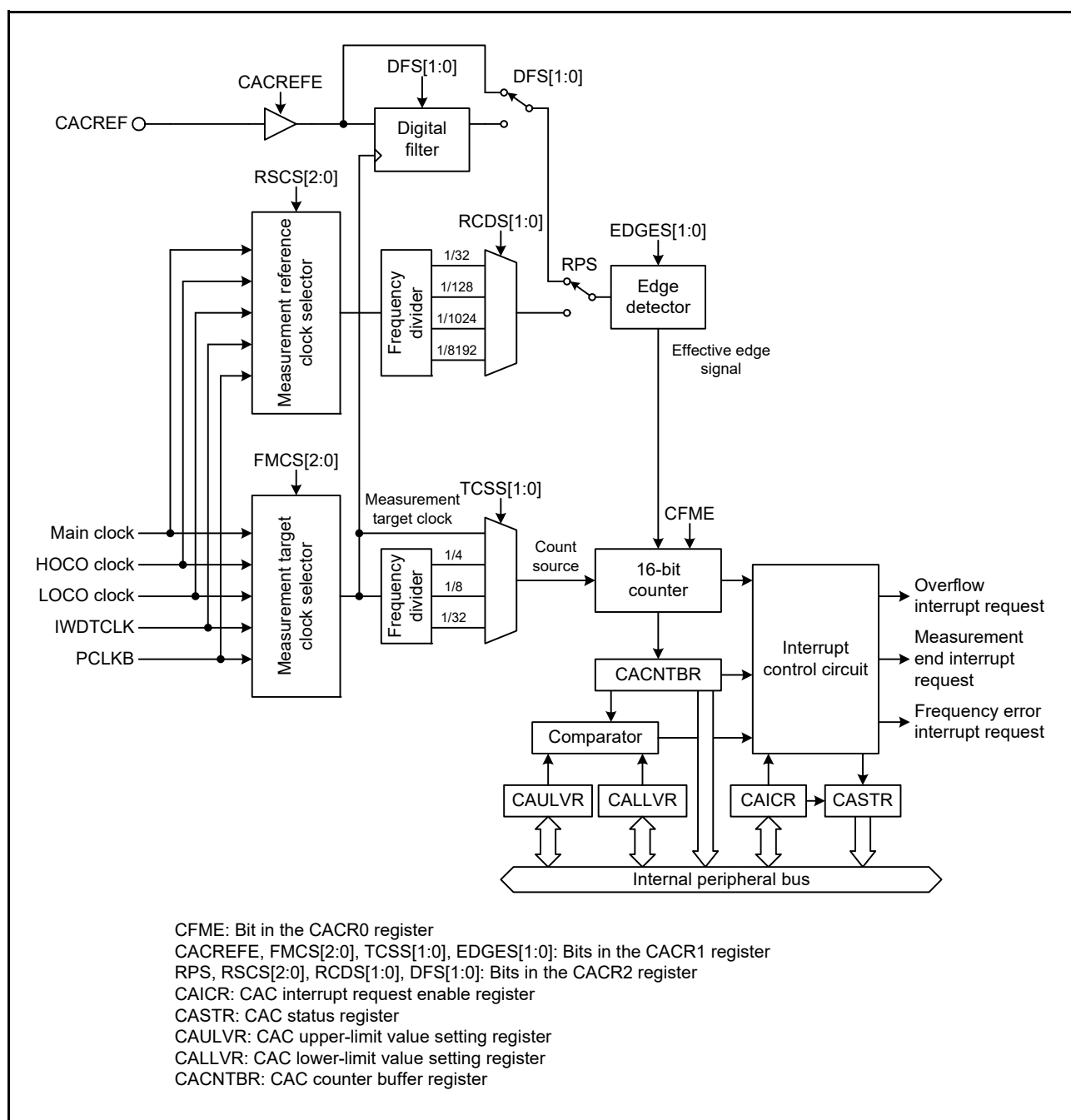


Figure 10.1 CAC Block Diagram

Table 10.2 shows the pin configuration of the CAC.

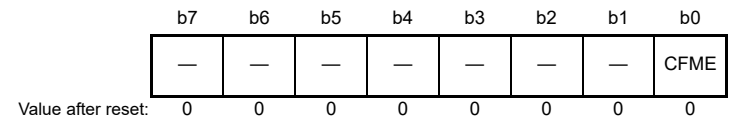
Table 10.2 Pin Configuration of CAC

Pin Name	I/O	Function
CACREF	Input	Measurement reference clock input pin

10.2 Register Descriptions

10.2.1 CAC Control Register 0 (CACR0)

Address(es): CAC.CACR0 0008 B000h



Bit	Symbol	Bit Name	Description	R/W
b0	CFME	Clock Frequency Measurement Enable	0: Clock frequency measurement is disabled. 1: Clock frequency measurement is enabled.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CFME Bit (Clock Frequency Measurement Enable)

This bit specifies whether clock frequency measurement is enabled or disabled.

When rewriting this bit, more time is required than other bits for the new value to be reflected in the register. Further write access to this bit are ignored until the current write access is reflected in the register. Read the bit to confirm that the rewrite has been reflected in the register.

10.2.2 CAC Control Register 1 (CACR1)

Address(es): CAC.CACR1 0008 B001h

b7	b6	b5	b4	b3	b2	b1	b0
EDGES[1:0]		TCSS[1:0]		FMCS[2:0]		CACREFE	
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	CACREFE	CACREF Pin Input Enable	0: CACREF pin input is disabled. 1: CACREF pin input is enabled.	R/W
b3 to b1	FMCS[2:0]	Measurement Target Clock Select	b3 b1 0 0 0: Main clock 0 1 0: HOCO clock 0 1 1: LOCO clock 1 0 0: IWDTC-dedicated clock (IWDTCCLK) 1 0 1: Peripheral module clock B (PCLKB) Settings other than above are prohibited.	R/W
b5, b4	TCSS[1:0]	Timer Count Source Select	b5 b4 0 0: No division 0 1: $\times 1/4$ clock 1 0: $\times 1/8$ clock 1 1: $\times 1/32$ clock	R/W
b7, b6	EDGES[1:0]	Valid Edge Select	b7 b6 0 0: Rising edge 0 1: Falling edge 1 0: Both rising and falling edges 1 1: Setting prohibited	R/W

Note: Set the CACR1 register when the CACR0.CFME bit is 0.

CACREFE Bit (CACREF Pin Input Enable)

This bit specifies whether the CACREF pin input is enabled or disabled.

FMCS[2:0]Bits (Measurement Target Clock Select)

These bits select the measurement target clock whose frequency is to be measured.

TCSS[1:0] Bits (Timer Count Source Select)

These bits select the count source for the clock frequency accuracy measurement circuit.

EDGES[1:0]Bits (Valid Edge Select)

These bits select the valid edge for the reference signal.

10.2.3 CAC Control Register 2 (CACR2)

Address(es): CAC.CACR2 0008 B002h

b7	b6	b5	b4	b3	b2	b1	b0
DFS[1:0]		RCDS[1:0]		RSCS[2:0]		RPS	
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	RPS	Reference Signal Select	0: CACREF pin input 1: Internal clock (internally generated signal)	R/W
b3 to b1	RSCS[2:0]	Measurement Reference Clock Select	b3 b1 0 0 0: Main clock 0 1 0: HOCO clock 0 1 1: LOCO clock 1 0 0: IWDT-dedicated clock (IWDTCCLK) 1 0 1: Peripheral module clock B (PCLKB) Settings other than above are prohibited.	R/W
b5, b4	RCDS[1:0]	Measurement Reference Clock Frequency Division Ration Select	b5 b4 0 0: $\times 1/32$ clock 0 1: $\times 1/128$ clock 1 0: $\times 1/1024$ clock 1 1: $\times 1/8192$ clock	R/W
b7, b6	DFS[1:0]	Digital Filter Select	b7 b6 0 0: Digital filtering is disabled. 0 1: The sampling clock for the digital filter is the measurement target clock. 1 0: The sampling clock for the digital filter is the measurement target clock divided by 4. 1 1: The sampling clock for the digital filter is the measurement target clock divided by 16.	R/W

Note: Set the CACR2 register when the CACR0.CFME bit is 0.

RPS Bit (Reference Signal Select)

This bit selects whether to use the CACREF pin input or an internal clock (internally generated signal) as the reference signal.

RSCS[2:0]Bits (Measurement Reference Clock Select)

These bits select the clock source for generating the measurement reference clock.

RCDS[1:0]Bits (Measurement Reference Clock Frequency Division Ration Select)

These bits select the frequency division ratio of the measurement reference clock.

DFS[1:0]Bits (Digital Filter Select)

The setting of these bits enables or disables the digital filter and selects its sampling clock.

10.2.4 CAC Interrupt Request Enable Register (CAICR)

Address(es): CAC.CAICR 0008 B003h

b7	b6	b5	b4	b3	b2	b1	b0
—	OVFFCL	MENDFCL	FERRFCL	—	OVFIE	MENDIE	FERRIE
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FERRIE	Frequency Error Interrupt Request Enable	0: Frequency error interrupt request is disabled. 1: Frequency error interrupt request is enabled.	R/W
b1	MENDIE	Measurement End Interrupt Request Enable	0: Measurement end interrupt request is disabled. 1: Measurement end interrupt request is enabled.	R/W
b2	OVFIE	Overflow Interrupt Request Enable	0: Overflow interrupt request is disabled. 1: Overflow interrupt request is enabled.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	FERRFCL	FERRF Clear	When 1 is written to this bit, the CASTR.FERRF flag is cleared. This bit is read as 0.	R/W
b5	MENDFCL	MENDF Clear	When 1 is written to this bit, the CASTR.MENDF flag is cleared. This bit is read as 0.	R/W
b6	OVFFCL	OVFF Clear	When 1 is written to this bit, the CASTR.OVFF flag is cleared. This bit is read as 0.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

FERRIE Bit (Frequency Error Interrupt Request Enable)

This bit specifies whether the frequency error interrupt request is enabled or disabled.

MENDIE Bit (Measurement End Interrupt Request Enable)

This bit specifies whether the measurement end interrupt request is enabled or disabled.

OVFIE Bit (Overflow Interrupt Request Enable)

This bit specifies whether the overflow interrupt request is enabled or disabled.

FERRFCL Bit (FERRF Clear)

Setting this bit to 1 clears the CASTR.FERRF flag.

MENDFCL Bit (MENDF Clear)

Setting this bit to 1 clears the CASTR.MENDF flag.

OVFFCL Bit (OVFF Clear)

Setting this bit to 1 clears the CASTR.OVFF flag.

10.2.5 CAC Status Register (CASTR)

Address(es): CAC.CASTR 0008 B004h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	OVFF	MENDF	FERRF
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	FERRF	Frequency Error Flag	0: The clock frequency is within the range corresponding to the settings. 1: The clock frequency has deviated beyond the range corresponding to the settings (frequency error).	R
b1	MENDF	Measurement End Flag	0: Measurement is in progress. 1: Measurement has ended.	R
b2	OVFF	Overflow Flag	0: The counter has not overflowed. 1: The counter has overflowed.	R
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

FERRF Flag (Frequency Error Flag)

This flag indicates deviation of the clock frequency from the set value (frequency error).

[Setting condition]

- The clock frequency is outside of the setting range.

[Clearing condition]

- 1 is written to the CAICR.FERRFCL bit.

MENDF Flag (Measurement End Flag)

This flag indicates the end of measurement.

[Setting condition]

- Measurement has finished.

[Clearing condition]

- 1 is written to the CAICR.MENDFCL bit.

OVFF Flag (Overflow Flag)

This flag indicates that the counter has overflowed.

[Setting condition]

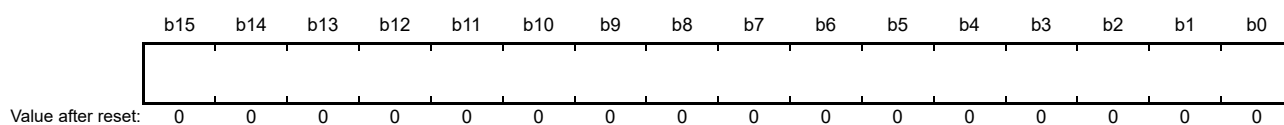
- The counter has overflowed.

[Clearing condition]

- 1 is written to the CAICR.OVFFCL bit.

10.2.6 CAC Upper-Limit Value Setting Register (CAULVR)

Address(es): CAC.CAULVR 0008 B006h



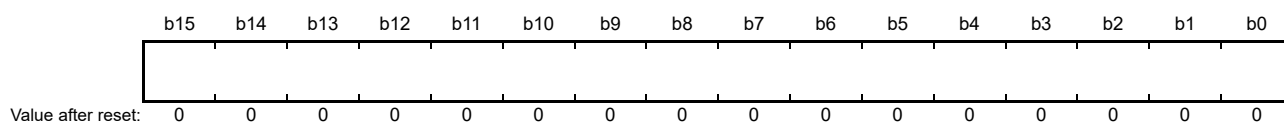
The CAULVR register is a 16-bit readable/writable register that specifies the upper-limit value of the counter used for measuring the frequency. When the frequency rises above the value specified in this register, a frequency error is detected.

Write to this register when the CACR0.CFME bit is 0.

The counter value held in the CACNTBR register can vary with the difference between the phases of the digital filter and edge-detection circuit on the one hand and the signal on the CACREF pin on the other, so ensure that this setting allows an adequate margin.

10.2.7 CAC Lower-Limit Value Setting Register (CALLVR)

Address(es): CAC.CALLVR 0008 B008h



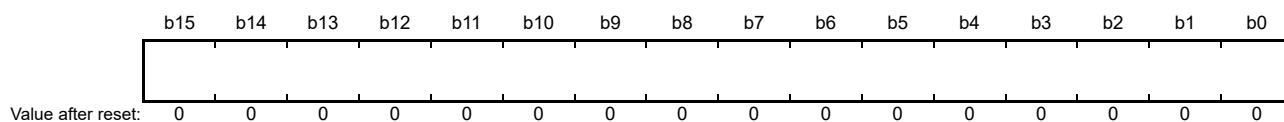
The CALLVR register is a 16-bit readable/writable register that specifies the lower-limit value of the counter used for measuring the frequency. When the frequency falls below the value specified in this register, a frequency error is detected.

Write to this register when the CACR0.CFME bit is 0.

The counter value held in the CACNTBR register can vary with the difference between the phases of the digital filter and edge-detection circuit on the one hand and the signal on the CACREF pin on the other, so ensure that this setting allows an adequate margin.

10.2.8 CAC Counter Buffer Register (CACNTBR)

Address(es): CAC.CACNTBR 0008 B00Ah



The CACNTBR register is a 16-bit read-only register that retains the counter value at the time a valid reference signal edge is input.

10.3 Operation

10.3.1 Measuring Clock Frequency

The clock frequency accuracy measurement circuit measures the clock frequency using the CACREF pin input or the internal clock as a reference. Figure 10.2 shows an operating example of the clock frequency accuracy measurement circuit.

The clock frequency accuracy measurement circuit operates as shown below when measuring the clock frequency.

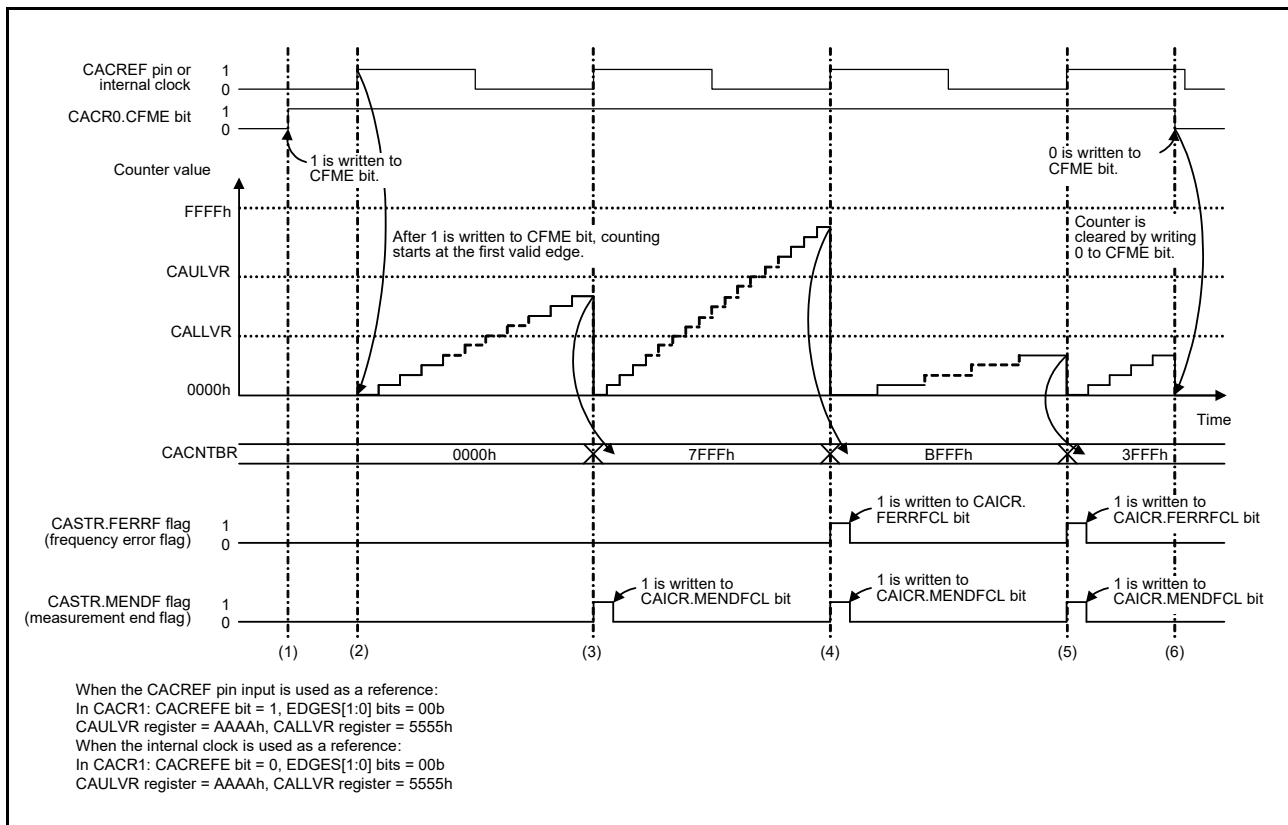


Figure 10.2 Operating Example of Clock Frequency Accuracy Measurement Circuit

- (1) When the CACREF pin input is used as a reference (the CACR1.CACREFE bit = 1), clock frequency measurement is enabled by writing 1 to the CACR0.CFME bit while the CACR2.RPS bit is 0 and the CACR1.CACREFE bit is 1. On the other hand, when the internal clock is used as a reference (the CACR1.CACREFE bit = 0), clock frequency measurement is enabled by writing 1 to the CACR0.CFME bit while the CACR2.RPS bit is 1.
- (2) When the CACREF pin input is used as a reference, the timer starts up-counting if the valid edge selected by the CACR1.EDGES[1:0] bits is input from the CACREF pin after 1 is written to the CFME bit. The valid edge is a rising edge (the CACR1.EDGES[1:0] bits = 00b) in Figure 10.2.
When the internal clock is used as a reference, the timer starts up-counting if the valid edge selected by the CACR1.EDGES[1:0] bits is input based on the clock source selected by the CACR2.RSCS[2:0] bits after 1 is written to the CFME bit. The valid edge is a rising edge (the CACR1.EDGES[1:0] bits = 00b) in Figure 10.2.
- (3) When the next valid edge is input, the counter value is transferred in the CACNTBR register and compared with the values of the CAULVR and CALLVR registers. If the formula $CALLVR \leq CACNTBR \leq CAULVR$ is satisfied, only the CASTER.MENDF flag is set to 1 because the clock frequency is correct. If the CAICR.MENDIE bit is 1, a measurement end interrupt is generated.
- (4) When the next valid edge is input, the counter value is transferred in the CACNTBR register and compared with the values of the CAULVR and CALLVR registers. In the case of $CACNTBR > CAULVR$, the CASTER.FERRF flag is set to 1 because the clock frequency is erroneous. If the CAICR.FERRIE bit is 1, a frequency error interrupt is

generated. Also, the CASTR.MENDF flag is set to 1. If the CAICR.MENDIE bit is 1, a measurement end interrupt is generated.

- (5) When the next valid edge is input, the counter value is transferred in the CACNTBR register and compared with the values of the CAULVR and CALLVR registers. In the case of $CACNTBR < CALLVR$, the CASTR.FERRF flag is set to 1 because the clock frequency is erroneous. If the CAICR.FERRIE bit is 1, a frequency error interrupt is generated. Also, the CASTR.MENDF flag is set to 1. If the CAICR.MENDIE bit is 1, a measurement end interrupt is generated.
- (6) While the CACR0.CFME bit is 1, the counter value is transferred in the CACNTBR register and compared with the values of the CAULVR and CALLVR registers every time a valid edge is input. Writing 0 to the CACR0.CFME bit clears the counter and stops up-counting.

10.3.2 Digital Filtering of Signals on the CACREF Pin

The CACREF pin has a digital filter. Levels on the target pin for sampling are conveyed to the internal circuitry after matching three consecutive times at the selected sampling interval and the same level continues to be conveyed internally until the level on the pin again matches three consecutive times.

Enabling and disabling of the digital filter and its sampling clock are selectable.

The counter value transferred in the CACNTBR register may be in error by up to one cycle of the sampling clock due to the difference between the phases of the digital filter and the signal input to the CACREF pin.

When a frequency dividing clock is selected as a count source, the counter value error is obtained by the following formula:

$$\text{Counter value error} = (\text{One cycle of the count source}) / (\text{One cycle of the sampling clock})$$

10.4 Interrupt Requests

The CAC generates three types of interrupt request: frequency error interrupt, measurement end interrupt, and overflow interrupt. When an interrupt source is generated, the corresponding status flag becomes 1. Table 10.3 lists details on the interrupt requests of the clock frequency accuracy measurement circuit.

Table 10.3 Interrupt Requests of Clock Frequency Accuracy Measurement Circuit

Interrupt Request	Interrupt Enable Bit	Status Flag	Interrupt Source
Frequency error interrupt	CAICR.FERRIE	CASTR.FERRF	The result of comparing CACNTBR to CAULVR and CALLVR is either $CACNTBR > CAULVR$ or $CACNTBR < CALLVR$.
Measurement end interrupt	CAICR.MENDIE	CASTR.MENDF	A valid edge is input from the CACREF pin. Note however that a measurement end interrupt does not occur at the first valid edge after writing 1 to the CACR0.CFME bit.
Overflow interrupt	CAICR.OVFIE	CASTR.OVFF	The counter has overflowed.

10.5 Usage Notes

10.5.1 Module Stop Function Setting

CAC operation can be disabled or enabled using module stop control register C (MSTPCRC). The initial setting is for the CAC to be halted. Register access is enabled by releasing the module stop state. For details, refer to **section 11, Low Power Consumption**.

11. Low Power Consumption

11.1 Overview

This MCU has several functions for reducing power consumption, by setting clock dividers, stopping modules, changing to low power consumption mode in normal operation, and changing to operating power control mode.

Table 11.1 lists the specifications of low power consumption functions, and Table 11.2 lists the conditions to change to low power consumption modes, states of the CPU and peripheral modules, and the method for exiting each mode.

After a reset, this MCU returns to normal mode, but modules except the DTC and RAM are stopped.

Table 11.1 Specifications of Low Power Consumption Functions

Item	Specification
Clock divider functions	The frequency division ratio can be set independently for the system clock (ICLK), peripheral module clock (PCLKB), S12AD clock (PCLKD), and FlashIF clock (FCLK). ^{*1}
Module stop function	Each peripheral module can be stopped independently by the module stop control register.
Function for transition to low power consumption mode	Transition to a low power consumption mode in which the CPU, peripheral modules, or oscillators are stopped is enabled.
Low power consumption modes	• Sleep mode • Deep sleep mode • Software standby mode
Operating power control modes	• Power consumption can be reduced in normal operation, sleep mode, and deep sleep mode by selecting an appropriate operating power consumption control mode according to the operating frequency and operating voltage range. • Two operating power control modes are available High-speed operating mode Middle-speed operating mode

Note 1. For details, refer to section 9, Clock Generation Circuit.

Table 11.2 Operating Conditions of Each Power Consumption Mode

Operating Conditions	Sleep Mode	Deep Sleep Mode	Software Standby Mode
Entry trigger	Control register + instruction	Control register + instruction	Control register + instruction
Triggers for exit other than a reset	Interrupt	Interrupt	Interrupt ^{*1}
State after exit from the given mode ^{*2}	Program in progress (interrupt handling)	Program in progress (interrupt handling)	Program in progress (interrupt handling)
Main clock oscillator	Operating possible	Operating possible	Stopped
High-speed on-chip oscillator (HOCO)	Operating possible	Operating possible	Stopped
Low-speed on-chip oscillator (LOCO)	Operating possible	Operating possible	Stopped
IWDT-dedicated on-chip oscillator (ILOCO)	Operating possible ^{*3}	Operating possible ^{*3}	Operating possible ^{*3}
PLL	Operating possible	Operating possible	Stopped
CPU	Stopped (Retained)	Stopped (Retained)	Stopped (Retained)
RAM0 (0000 0000h to 0000 2FFFh)	Operating possible (Retained)	Stopped (Retained)	Stopped (Retained)
DTC	Operating possible ^{*5}	Stopped (Retained)	Stopped (Retained)
Flash memory	Operating	Stopped (Retained)	Stopped (Retained)
Independent watchdog timer (IWDT)	Operating possible ^{*3}	Operating possible ^{*3}	Operating possible ^{*3}
Voltage detection circuit (LVD)	Operating possible	Operating possible	Operating possible
Power-on reset circuit	Operating	Operating	Operating
Peripheral modules	Operating possible	Operating possible	Stopped (Retained) ^{*4}
I/O ports	Operating	Operating	Retained
CLKOUT	Operating possible	Operating possible	Stopped
Comparator C	Operating possible	Operating possible	Operating possible ^{*6}

"Operating possible" means that operating or stopped can be controlled by the register setting.

"Stopped (Retained)" means that internal register values are retained and internal operations are suspended.

- Note 1. "Interrupts" here indicates an external pin interrupt (the NMI or IRQ0 to IRQ7) or any of peripheral interrupts (interrupts on the IWDG, voltage monitoring).
- Note 2. This does not include a RES# pin reset, power-on reset, voltage monitoring reset, or independent watchdog-timer reset. One of these reset sources initiate transition to reset state.
- Note 3. Operating or stopping is selected by setting the IWDG sleep mode count stop control bit (IWDGSLCSTP) in option function select register 0 (OFS0) in IWDG auto-start mode. In any mode other than IWDG auto-start mode, operating or stopping is selected by the setting of the sleep mode count stop control bit (SLCSTP) in the IWDG count stop control register (IWDGCSCTR).
- Note 4. The peripheral logic states are retained.
- Note 5. Writing to the system control registers in the sleep mode is prohibited. For the system control registers, see Table 5.1, List of I/O Registers (Address Order), in which they are indicated as "SYSTEM" in the Module System column.
- Note 6. Using the digital filter function is prohibited. Operation for outputting the comparison result to the COMPn pin is possible.

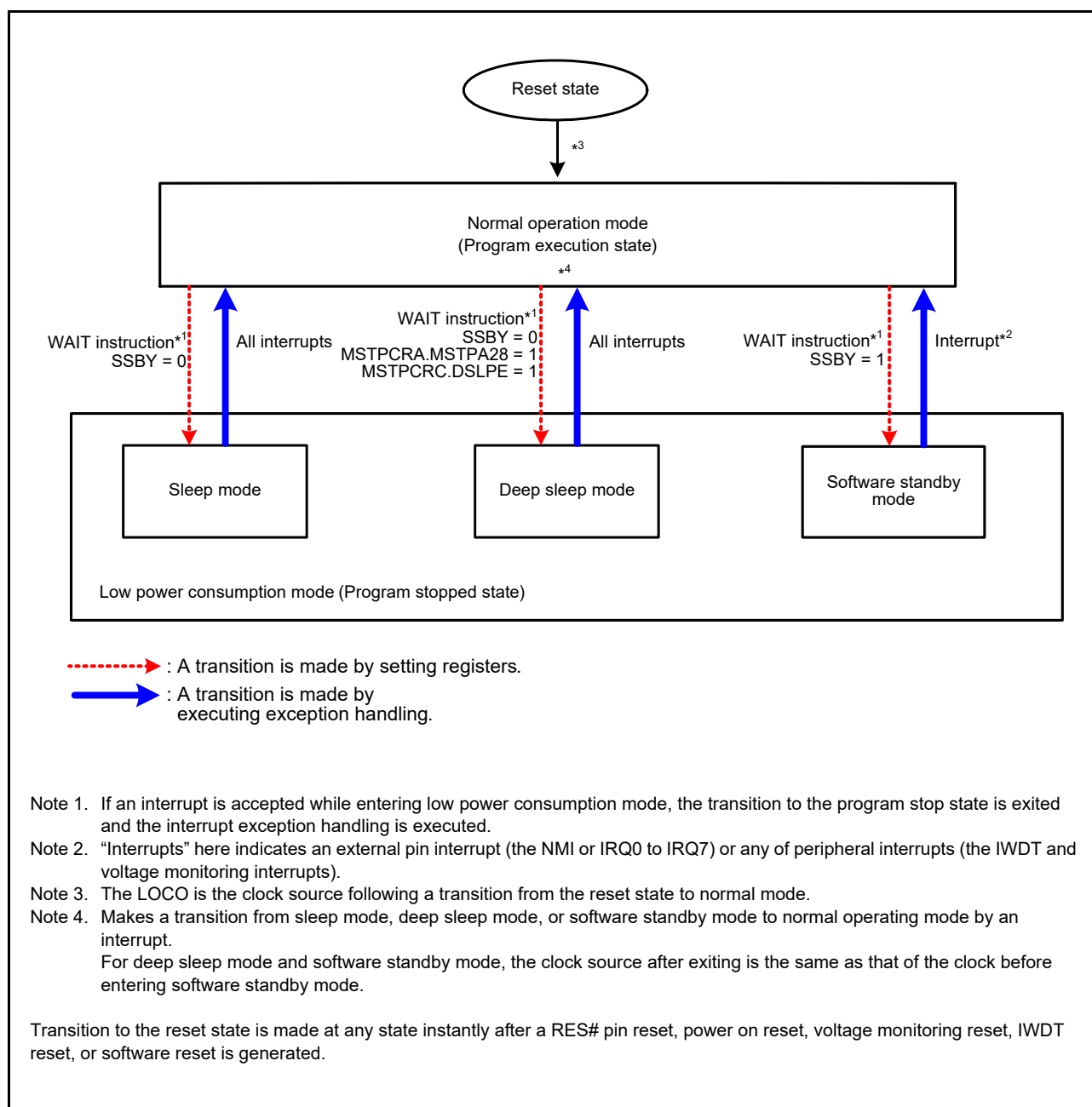


Figure 11.1 Mode Transitions

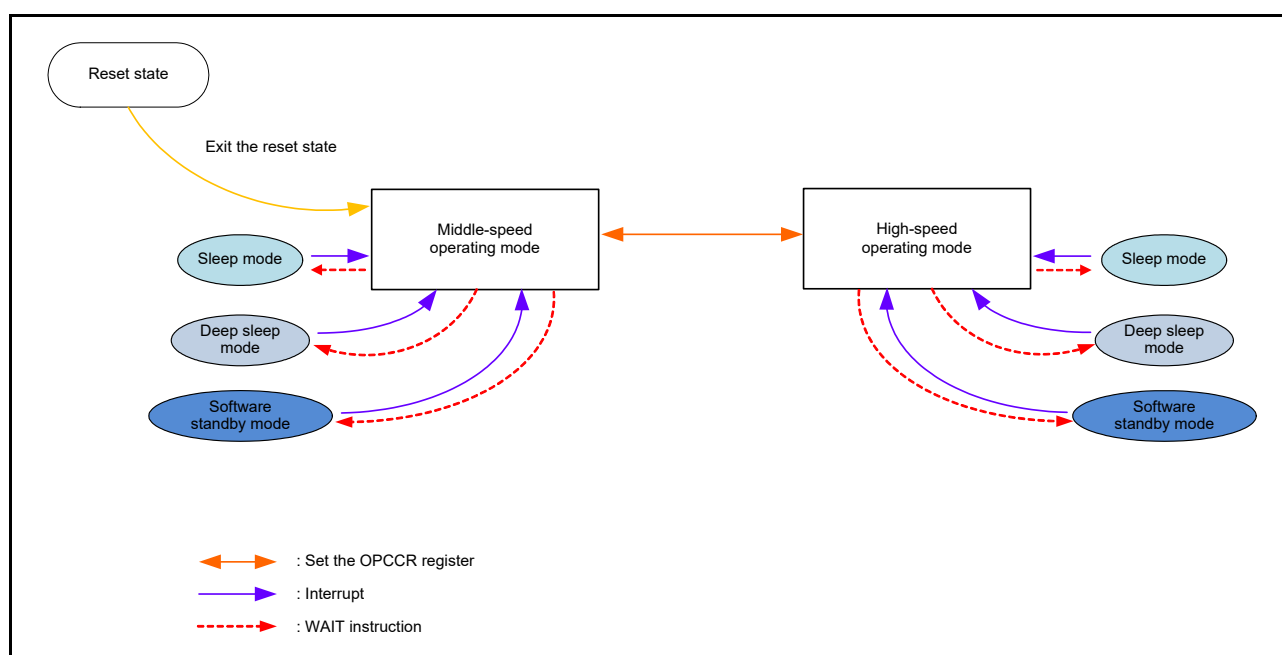


Figure 11.2 Operating Modes

- It is possible to return from sleep mode to the previous operating state used before entering sleep mode.
- After releasing from the reset state, the MCU starts operating in the middle-speed operating mode.
- In sleep mode or when the mode transition is in progress, writing to the registers related to the system controlling (registers for which SYSTEM is shown in the module symbol column in the I/O register list) is prohibited.

Table 11.3 Operations of the Oscillators in Each Operating Mode

	PLL	HOCO	LOCO	IWDT- dedicated On-chip Oscillator	Main Clock Oscillator
High-speed operating mode	Possible	Possible	Possible	Possible	Possible
Middle-speed operating mode	Possible	Possible	Possible	Possible	Possible

11.2 Register Descriptions

11.2.1 Standby Control Register (SBYCR)

Address(es): SYSTEM.SBYCR 0008 000Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SSBY	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b14 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	SSBY	Software Standby	0: Set entry to sleep mode or deep sleep mode after the WAIT instruction is executed 1: Set entry to software standby mode after the WAIT instruction is executed	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

SSBY Bit (Software Standby)

The SSBY bit specifies the transition destination after the WAIT instruction is executed.

When the SSBY bit is set to 1, the MCU enters software standby mode after execution of the WAIT instruction. When the MCU returns to normal mode after an interrupt has triggered and exits from software standby mode, the SSBY bit remains 1. The SSBY bit can be cleared by writing 0 to the SSBY bit.

When the oscillation stop detection function enable bit (OSTDCR.OSTDE) in the oscillation stop detection control register is 1, the set value of the SSBY bit is invalid. Even if the SSBY bit is 1, the MCU will enter sleep mode or deep sleep mode after execution of the WAIT instruction.

11.2.2 Module Stop Control Register A (MSTPCRA)

Address(es): SYSTEM.MSTPCRA 0008 0010h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	MSTPA 28	—	—	—	—	—	—	—	—	MSTPA 19	—	MSTPA 17	MSTPA 16
Value after reset:	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	MSTPA 15	—	—	—	—	—	MSTPA 9	—	MSTPA 7	—	MSTPA 5	MSTPA 4	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b4	MSTPA4	8-bit Timer 3 and 2 (Unit 1) Module Stop	Target module: TMR3, TMR2 0: This module clock is enabled 1: This module clock is disabled	R/W
b5	MSTPA5	8-bit Timer 1 and 0 (Unit 0) Module Stop	Target module: TMR1, TMR0 0: This module clock is enabled 1: This module clock is disabled	R/W
b6	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b7	MSTPA7	General PWM Timer/ GPTW Port Output Enable Module Stop	Target module: GPTW, POEG 0: This module clock is enabled 1: This module clock is disabled	R/W
b8	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b9	MSTPA9	Multifunction Timer Pulse Unit 3 Module Stop	Target module: MTU 0: This module clock is enabled 1: This module clock is disabled	R/W
b14 to b10	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b15	MSTPA15	Compare Match Timer (Unit 0) Module Stop	Target module: CMT unit 0 (CMT0, CMT1) 0: This module clock is enabled 1: This module clock is disabled	R/W
b16	MSTPA16	12-bit A/D Converter (Unit 1) Module Stop	Target module: S12AD unit 1 0: This module clock is enabled 1: This module clock is disabled	R/W
b17	MSTPA17	12-Bit A/D Converter (Unit 0)/Programmable Gain Amplifier Module Stop	Target module: S12AD unit 0, PGA 0: This module clock is enabled 1: This module clock is disabled	R/W
b18	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b19	MSTPA19	D/A Converter Module Stop	Target module: DA 0: This module clock is enabled 1: This module clock is disabled	R/W
b27 to b20	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b28	MSTPA28	Data Transfer Controller Module Stop	Target module: DTC 0: This module clock is enabled 1: This module clock is disabled	R/W
b31 to b29	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

11.2.3 Module Stop Control Register B (MSTPCRB)

Address(es): SYSTEM.MSTPCRB 0008 0014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	MSTPB30	—	—	—	MSTPB26	MSTPB25	—	MSTPB23	—	MSTPB21	—	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MSTPB10	—	—	—	MSTPB6	—	MSTPB4	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b4	MSTPB4	Serial Communication Interface 12 Module Stop	Target module: SCI12 0: This module clock is enabled 1: This module clock is disabled	R/W
b5	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b6	MSTPB6	DOC Module Stop	Target module: DOC 0: This module clock is enabled 1: This module clock is disabled	R/W
b9 to b7	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b10	MSTPB10	Comparator C Module Stop	Target module: CMPC 0: This module clock is enabled 1: This module clock is disabled	R/W
b20 to b11	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b21	MSTPB21	I ² C Bus Interface 0 Module Stop	Target module: RIIC0 0: This module clock is enabled 1: This module clock is disabled	R/W
b22	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b23	MSTPB23	CRC Calculator Module Stop	Target module: CRC 0: This module clock is enabled 1: This module clock is disabled	R/W
b24	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b25	MSTPB25	Serial Communication Interface 6 Module Stop	Target module: SCI6 0: This module clock is enabled 1: This module clock is disabled	R/W
b26	MSTPB26	Serial Communication Interface 5 Module Stop	Target module: SCI5 0: This module clock is enabled 1: This module clock is disabled	R/W
b29 to b27	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b30	MSTPB30	Serial Communication Interface 1 Module Stop	Target module: SCI1 0: This module clock is enabled 1: This module clock is disabled	R/W
b31	—	Reserved	This bit is read as 1. The write value should be 1.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

11.2.4 Module Stop Control Register C (MSTPCRC)

Address(es): SYSTEM.MSTPCRC 0008 0018h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DSLPE	—	—	—	—	—	—	—	—	—	—	—	MSTPC19	—	—	—
Value after reset:	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTPC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MSTPC0	RAM0 Module Stop*1	Target module: RAM0 (0000 0000h to 0000 2FFFh) 0: RAM0 operating 1: RAM0 stopped	R/W
b15 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b18 to b16	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b19	MSTPC19	Clock Frequency Accuracy Measurement Circuit Module Stop*2	Target module: CAC 0: This module clock is enabled 1: This module clock is disabled	R/W
b30 to b20	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b31	DSLPE	Deep Sleep Mode Enable	0: Deep sleep mode is disabled 1: Deep sleep mode is enabled	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

Note 1. The corresponding MSTPC0 bit should not be set to 1 during access to the RAM. The corresponding RAM should not be accessed while the MSTPC0 bit is 1.

Note 2. The MSTPC19 bit should be rewritten while the oscillation of the clock to be controlled by this bit is stable. For entering software standby mode after rewriting this bit, wait for two cycles of the slowest clock among the clocks output by the oscillators actually oscillating and execute the WAIT instruction.

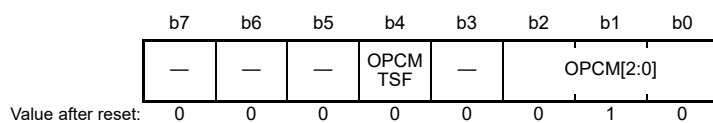
DSLPE Bit (Deep Sleep Mode Enable)

The DSLPE bit enables or disables a transition to deep sleep mode.

When the CPU executes the WAIT instruction with the DSLPE bit set to 1 and the SBYCR.SSBY and MSTPCRA.MSTPA28 bits meet specified conditions, the MCU enters deep sleep mode. For details, refer to section 11.6.2, Deep Sleep Mode.

11.2.5 Operating Power Control Register (OPCCR)

Address(es): SYSTEM.OPCCR 0008 00A0h



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	OPCM[2:0]	Operating Power Control Mode Select	b2 b0 0 0 0: High-speed operating mode 0 1 0: Middle-speed operating mode Settings other than above are prohibited.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	OPCMTSF	Operating Power Control Mode Transition Status Flag	0: Transition completed 1: During transition	R
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

The OPCCR register is used to reduce the power consumption in the normal operating mode, sleep mode, and deep sleep mode.

Power consumption can be reduced according to the operating frequency to be used by the OPCCR setting.

The OPCCR register cannot be rewritten under the following conditions:

- When the OPCCR.OPCMTSF flag is 1 (during transition)
- Time period from WAIT instruction execution for a sleep mode transition, until exit from sleep mode to normal operation
- Time period from WAIT instruction execution for a deep sleep mode transition, until exit from deep sleep mode to normal operation

The OPCCR register cannot be rewritten while the flash memory is being programmed or erased (P/E).

For the procedures of changing operating power control modes, refer to Function in section 11.5, Function for Lower Operating Power Consumption.

During a transition to an operating power control mode (while the OPCCR.OPCMTSF flag is 1), a correct value cannot be read from the E2 DataFlash. If a setting is made so that the E2 DataFlash is read using a DTC transfer, stop the DTC module before rewriting the OPCCR.OPCM[2:0] bits.

During sleep mode or mode transitions, do not write to the registers related to system control (indicated by 'SYSTEM' in the Module Symbol column in Table 5.1, List of I/O Registers (Address Order)).

OPCM[2:0] Bits (Operating Power Control Mode Select)

The operating power control mode in the normal operating mode, sleep mode, and deep sleep mode is selectable.

Table 11.4 shows the relationship between operating power control modes, the OPCM[2:0] bit settings, and the operating frequency and voltage ranges.

OPCMTSF Flag (Operating Power Control Mode Transition Status Flag)

This flag indicates the switching control state during and after operating power mode transition.

This flag becomes 1 when the value of the OPCM[2:0] bits is rewritten, and 0 when mode transition is completed. Read this flag and confirm that it is 0 before proceeding to the next processing. Only rewrite the OPCM[2:0] bits when this flag is 0.

Table 11.4 Operating Frequency and Voltage Ranges in Operating Power Control Modes

Operating Power Control Mode	OPCM[2:0] Bits	Operating Voltage Range	Operating Frequency Range				
			Flash Memory Read Frequency				Flash Memory Programming/Erase Frequency
			ICLK	FCLK	PCLKD	PCLKB	
High-speed operating mode	000b	2.7 to 5.5 V	Up to 48 MHz*1	Up to 48 MHz	Up to 64 MHz	Up to 48 MHz	1 to 48 MHz*1
Middle-speed operating mode	010b		Up to 24 MHz	Up to 24 MHz	Up to 24 MHz	Up to 24 MHz	1 to 24 MHz

Note: When using the FCLK at lower than 4 MHz during programming or erasing the flash memory, the frequency can be set to 1, 2, or 3 MHz.

Note 1. When setting a clock with a frequency higher than 32 MHz, it is necessary to set the wait insertion when accessing the ROM. For details, refer to section 37, Flash Memory (FLASH).

Each operating power control mode is described below.

- High-Speed Operating Mode

The maximum operating frequency during FLASH read is 48 MHz for ICLK, PCLKB, FCLK; 64 MHz for PCLKD. The operating voltage range is 2.7 to 5.5 V during FLASH read. During FLASH programming/erasure, the operating frequency range is 1 to 48 MHz and the operating voltage range is 2.7 to 5.5 V.

Figure 11.3 shows the relation between the operating voltage and operating frequency in the high-speed operating mode.

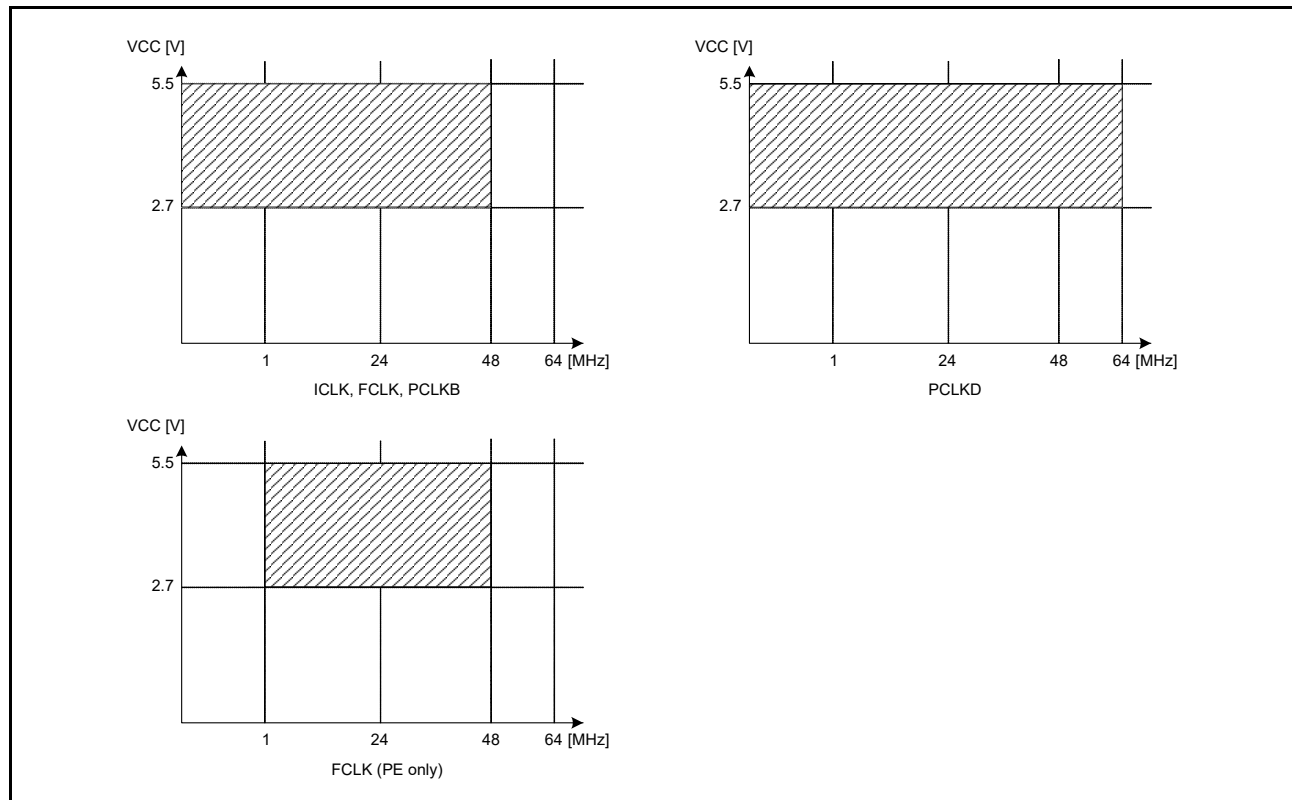


Figure 11.3 Operating Voltages and Frequencies in High-Speed Operating Mode

Note: When using the FCLK at lower than 4 MHz during programming or erasing the flash memory, the frequency can be set to 1, 2, or 3 MHz.

- Middle-Speed Operating Mode

As compared to high-speed operating mode, this mode reduces power consumption for low-speed operation. The maximum operating frequency during FLASH read is 24 MHz for ICLK, FCLK, PCLKB, and PCLKD. The operating voltage range is 2.7 to 5.5 V during FLASH read.

During FLASH programming/erasure, the operating frequency range is 1 to 24 MHz and the operating voltage range is 2.7 to 5.5 V.

The power consumption of this mode is lower than that of high speed mode under the same conditions.

After a reset is canceled, operation is started from this mode.

Figure 11.4 shows the relation between the operating voltage and operating frequency in the middle-speed operating mode.

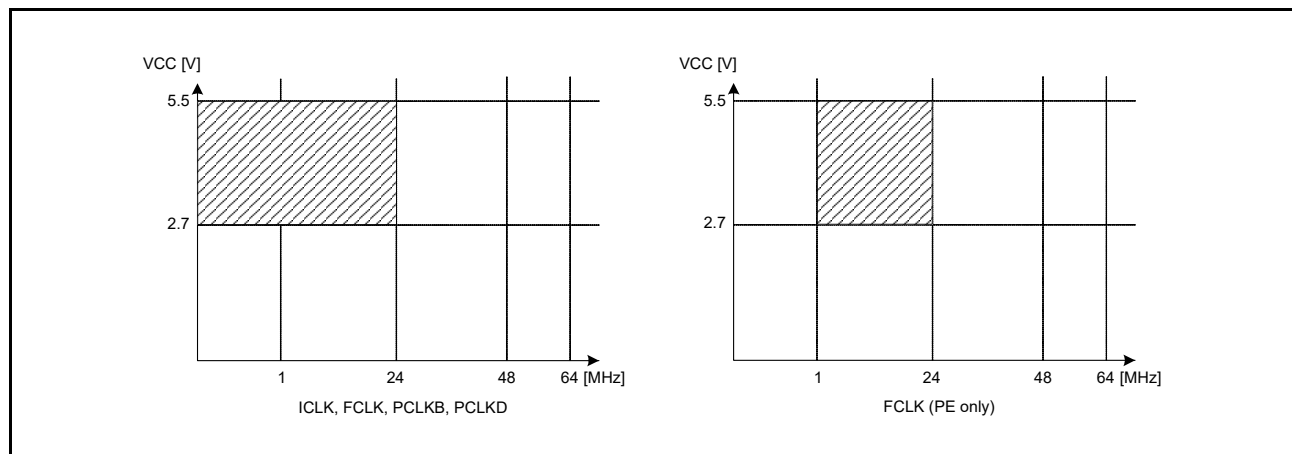


Figure 11.4 Operating Voltages and Frequencies in Middle-Speed Operating Mode

Note: When using the FCLK at lower than 4 MHz during programming or erasing the flash memory, the frequency can be set to 1, 2, or 3 MHz.

11.3 Reducing Power Consumption by Switching Clock Signals

The clock frequency can change by setting the SCKCR.FCK[3:0], ICK[3:0], PCKB[3:0], and PCKD[3:0] bits. The CPU, DTC, ROM, and RAM clocks can be set by the ICK[3:0] bits. The peripheral module clocks can be set by the PCKB[3:0] and PCKD[3:0] bits.

The flash memory clock can be set by the FCK[3:0] bits.

For details, refer to section 9, Clock Generation Circuit.

11.4 Module Stop Function

The module stop function can be used to stop the operation of on-chip peripheral modules, and can be individually set for each peripheral module. The CPU continues its operation.

Setting the MSTPmi bit (m = A to C; i = 0 to 31) in MSTPCRA to MSTPCRC registers to 1 stops the operation of the corresponding peripheral module and places it in the module stop state. The internal states of peripheral modules are retained in the module stop state. Setting the MSTPmi bit to 0 releases the module stop state for the corresponding peripheral module, and operation resumes at the end of the bus cycle.

After a reset is canceled, all peripheral modules other than the DTC, and on-chip RAM are in the module stop state.

Basically the registers in the module stop state cannot be read or written. However, note that data may be written to these registers if write access is made immediately after the setting of the module stop state. To avoid this, always write to the module stop registers after confirming that the last register setting is done. Note that, for some on-chip peripheral modules, access to registers while in the module stop state may cause a bus timeout error. Accordingly, be sure to release the module stop state before accessing peripheral module registers.

For details on bus timeout, see section 15.4.1.2, Timeout.

11.5 Function for Lower Operating Power Consumption

By selecting an appropriate operating power control mode according to the operating frequency and operating voltage, power consumption can be reduced in normal mode, sleep mode, and deep sleep mode.

11.5.1 Setting Operating Power Control Mode

Examples of the procedures for switching operating power control modes are shown below:

(1) Switching from Normal Power Consumption Mode to Low Power Consumption Mode

- Example: From high-speed operating mode to middle-speed operating mode

(High-speed operation in high-speed operating mode)



Set the frequency of each clock to lower than the maximum operating frequency for middle-speed operating mode



Confirm that the OPCCR.OPCMTSF flag is 0 (transition completed)



Set the OPCCR.OPCM[2:0] bits to 010b (middle-speed operating mode)



Confirm that the OPCCR.OPCMTSF flag is 0 (transition completed)



(Middle-speed operation in middle-speed operating mode)

(2) Switching from Low Power Consumption Mode to Normal Power Consumption Mode

- Example: From middle-speed operating mode to high-speed operating mode

Middle-speed operation in middle-speed operating mode



Confirm that the OPCCR.OPCMTSF flag is 0 (transition completed)



Set the OPCCR.OPCM[2:0] bit to 0 (high-speed operating mode)



Confirm that the OPCCR.OPCMTSF flag is 0 (transition completed)



Set the frequency of each clock to lower than the maximum operating frequency for high-speed operating mode



High-speed operation in high-speed operating mode

11.6 Low Power Consumption Modes

11.6.1 Sleep Mode

11.6.1.1 Entry to Sleep Mode

When the WAIT instruction is executed while the SBYCR.SSBY bit is 0, the CPU enters sleep mode. In sleep mode, the CPU stops operating but the contents of its internal registers are retained. Other peripheral functions do not stop.

Counting by the IWDT stops if a transition to sleep mode is made while the IWDT is being used in auto-start mode and the OFS0.IWDTSLCSTP bit is 1. In the same way, counting by the IWDT stops if a transition to sleep mode is made while the IWDT is being used in register start mode and the IWDTCSLSTP bit is 1.

Furthermore, counting by the IWDT continues if a transition to sleep mode is made while the IWDT is being used in auto-start mode and the OFS0.IWDTSLCSTP bit is 0 (counting by the IWDT continues through transitions to low power consumption modes). In the same way, counting by the IWDT continues if a transition to sleep mode is made while the IWDT is being used in register start mode and the IWDTCSLTPR.SLCSTP bit is 0.

To use sleep mode, make the following settings and then execute a WAIT instruction.

- (1) Set the PSW.I bit^{*1} of the CPU to 0.
- (2) Set the interrupt request destination^{*2} to be used for exit from sleep mode.
- (3) Set the priority^{*3} of the interrupt to be used for exit from sleep mode to a level higher than the setting of the PSW.IPL[3:0] bits^{*1} of the CPU.
- (4) Set the IERm.IENj bit^{*3} to 1 for the interrupt.
- (5) Read the I/O register that is written last and confirm that the written value has been reflected.
- (6) Execute the WAIT instruction (this automatically sets the I bit^{*1} in the PSW of the CPU to 1).

Note 1. For details, refer to section 2, CPU.

Note 2. For details, refer to section 14.4.3, *Selecting Interrupt Request Destinations*.

Note 3. For details, refer to section 14, Interrupt Controller (ICUb).

11.6.1.2 Exit from Sleep Mode

Exit from sleep mode is initiated by any interrupt, a RES# pin reset, a power-on reset, a voltage monitoring reset, or a reset caused by an IWDT underflow.

- Initiated by an interrupt
An interrupt initiates exit from sleep mode and the interrupt exception handling starts. If a maskable interrupt has been masked by the CPU (the priority level*¹ of the interrupt has been set to a value lower than that of the PSW.IPL[3:0] bits*² of the CPU), sleep mode is not exited.
- Initiated by a RES# pin reset
When the RES# pin is driven low, the MCU enters the reset state. When the RES# pin is driven high after the reset signal is input for a predetermined time period, the CPU starts the reset exception handling.
- Initiated by a power-on reset
A power-on reset asserts a reset to the MCU.
When a power-on reset is negated by a rise in the supply voltage, the CPU starts the reset exception handling.
- Initiated by a voltage monitoring reset
A voltage monitoring reset asserts a reset to the MCU.
When a voltage monitoring reset is negated by a rise in the supply voltage, the CPU starts the reset exception handling.
- Initiated by an independent watchdog timer reset
An internal reset generated by an IWDT underflow asserts a reset to the MCU. However, when IWDT counting is stopped in sleep mode by setting OFS0.IWDTSTRT = 0 and OFS0.IWDTSLCSTP = 1, or OFS0.IWDTSTRT = 1 and IWDTCSTPR.SLCSTP = 1, the IWDT is stopped in sleep mode and sleep mode is not exited by the independent watchdog timer reset.

Note 1. For details, refer to section 14, Interrupt Controller (ICUb).

Note 2. For details, refer to section 2, CPU.

11.6.2 Deep Sleep Mode

11.6.2.1 Entry to Deep Sleep Mode

When a WAIT instruction is executed with the MSTPCRC.DSLPE bit set to 1, the MSTPCRA.MSTPA28 bit set to 1, and the SBYCR.SSBY bit set to 0, a transition to deep sleep mode is made.

In deep sleep mode, the CPU and the DTC, ROM, and RAM clocks stop. Peripheral functions do not stop.

Counting by the IWDTC stops if a transition to deep sleep mode is made while the IWDTC is being used in auto-start mode and the OFS0.IWDTCSTP bit is 1. In the same way, counting by the IWDTC stops if a transition to deep sleep mode is made while the IWDTC is being used in register start mode and the IWDTCSTPR.SLCSTP bit is 1.

Furthermore, counting by the IWDTC continues if a transition to deep sleep mode is made while the IWDTC is being used in auto-start mode and the OFS0.IWDTCSTP bit is 0 (counting by the IWDTC continues through transitions to low power consumption modes). In the same way, counting by the IWDTC continues if a transition to deep sleep mode is made while the IWDTC is being used in register start mode and the IWDTCSTPR.SLCSTP bit is 0.

To use deep sleep mode, make the following settings and then execute a WAIT instruction.

- (1) Set the PSW.I bit*¹ of the CPU to 0.
- (2) Set the interrupt request destination*² to be used for exit from deep sleep mode.
- (3) Set the priority*³ of the interrupt to be used for exit from deep sleep mode to a level higher than the setting of the PSW.IPL[3:0] bits*¹ of the CPU.
- (4) Set the IERm.IENj bit*³ to 1 for the interrupt.
- (5) Read the I/O register that is written last and confirm that the written value has been reflected.
- (6) Execute a WAIT instruction (executing a WAIT instruction causes automatic setting of the PSW.I bit*¹ of the CPU to 1).

Note 1. For details, refer to section 2, CPU.

Note 2. For details, refer to section 14.4.3, Selecting Interrupt Request Destinations.

Note 3. For details, refer to section 14, Interrupt Controller (ICUb).

11.6.2.2 Exit from Deep Sleep Mode

Exit from deep sleep mode is initiated by any interrupt, a RES# pin reset, a power-on reset, a voltage monitoring reset, or a reset caused by an IWDT underflow.

- Initiated by an interrupt
An interrupt initiates exit from deep sleep mode and the interrupt exception handling starts. If a maskable interrupt has been masked by the CPU (the priority level*¹ of the interrupt has been set to a value lower than that of the PSW.IPL[3:0] bits*² of the CPU), deep sleep mode is not exited.
- Initiated by the RES# pin reset
When the RES# pin is driven low, the MCU enters the reset state. When the RES# pin is driven high after the reset signal is input for a predetermined time period, the CPU starts the reset exception handling.
- Initiated by a power-on reset
A power-on reset asserts a reset to the MCU.
When a power-on reset is negated by a rise in the supply voltage, the CPU starts the reset exception handling.
- Initiated by a voltage monitoring reset
A voltage monitoring reset asserts a reset to the MCU.
When a voltage monitoring reset is negated by a rise in the supply voltage, the CPU starts the reset exception handling.
- Initiated by the independent watchdog timer
An internal reset generated by an IWDT underflow asserts a reset to the MCU. However, when IWDT counting is stopped in deep sleep mode by setting OFS0.IWDTSTRT = 0 and OFS0.IWDTSLCSTP = 1, or OFS0.IWDTSTRT = 1 and IWDTCSTPR.SLCSTP = 1, the IWDT is stopped in deep sleep mode and deep sleep mode is not exited by the independent watchdog timer reset.

Note 1. For details, refer to section 14, Interrupt Controller (ICUb).

Note 2. For details, refer to section 2, CPU.

11.6.3 Software Standby Mode

11.6.3.1 Entry to Software Standby Mode

When a WAIT instruction is executed with the SBYCR.SSBY bit set to 1, a transition to software standby mode is made. In this mode, all the other functions except the CPU and on-chip peripheral functions stop. However, the contents of the CPU internal registers, RAM data, the states of on-chip peripheral functions, and the I/O ports are retained. Software standby mode allows significant reduction in power consumption because the oscillator stops in this mode. Counting by the IWDT stops if a transition to software standby mode is made while the IWDT is being used in auto-start mode and the OFS0.IWDTSLCSTP bit is 1. In the same way, counting by the IWDT stops if a transition to software standby mode is made while the IWDT is being used in register start mode and the IWDTCSLTPR.SLCSTP bit is 1.

Furthermore, counting by the IWDT continues if a transition to software standby mode is made while the IWDT is being used in auto-start mode and the `OFS0.IWDTSLCSTP` bit is 0 (counting by the IWDT continues through transitions to low power consumption modes). In the same way, counting by the IWDT continues if a transition to software standby mode is made while the IWDT is being used in register start mode and the `IWDTCSSTPR.SLCSTP` bit is 0.

To use software standby mode, make the following settings and then execute a `WAIT` instruction.

- (1) Set the PSW.I bit^{*1} of the CPU to 0.
- (2) Set the interrupt request destination^{*2} to be used for recovery from software standby mode to the CPU.
- (3) Set the priority^{*3} of the interrupt to be used for recovery from software standby mode to a level higher than the setting of the PSW.IPL[3:0] bits^{*1} of the CPU.
- (4) Set the IERm.IENj bit^{*3} to 1 for the interrupt.
- (5) Read the I/O register that is written last and confirm that the written value has been reflected.
- (6) Execute a WAIT instruction (executing a WAIT instruction causes automatic setting of the PSW.I bit^{*1} of the CPU to 1).

Note 1. For details, refer to section 2, CPU.

Note 2. For details, refer to section 14.4.3, *Selecting Interrupt Request Destinations*.

Note 3. For details, refer to section 14, Interrupt Controller (ICUb).

11.6.3.2 Exit from Software Standby Mode

Release from the software standby mode is initiated by an external pin interrupt (the NMI or IRQ0 to IRQ7), peripheral interrupts (the IWDT and voltage monitoring), an RES# pin reset, a power-on reset, a voltage monitoring reset, or an independent watchdog timer reset. When any trigger which initiates exit from software standby mode is asserted, the oscillators which were operating before entry to software standby mode restart operation. After the oscillation of all these oscillators has been stabilized, operation returns from software standby mode.

- Initiated by an interrupt
When an interrupt request from the NMI, IRQ0 to IRQ7, IWDT, voltage monitoring, each oscillator which was operating before a transition to software standby mode resumes oscillation. After the oscillation stabilization wait time of each oscillator set by the MOSCWTCR.MSTS[4:0] bits has elapsed, the MCU exits software standby mode and interrupt exception processing starts.
- Initiated by a RES# pin reset
Clock oscillation starts when the low level is applied to the RES# pin. Clock supply for the MCU starts at the same time. Keep the level on the RES# pin low over the time required for oscillation of the clocks to become stable. Reset exception processing starts when the high level is applied to the RES# pin.
- Initiated by a power-on reset
A power-on reset asserts a reset to the MCU.
When a power-on reset is negated by a rise in the supply voltage, the CPU starts the reset exception handling.
- Initiated by a voltage monitoring reset
A voltage monitoring reset asserts a reset to the MCU.
When a voltage monitoring reset is negated by a rise in the supply voltage, the CPU starts the reset exception handling.
- Initiated by an independent watchdog timer reset
An internal reset generated by an IWDT underflow asserts a reset to the MCU.
Note that the independent watchdog timer is stopped in software standby mode due to the register settings (OFS0.IWDTSTRT = 0 and OFS0.IWDTSLCSTP = 1, or OFS0.IWDTSTRT = 1 and IWDTCSTPR.SLCSTP = 1) in software standby mode. In that case, exit from software standby mode by the independent watchdog timer reset cannot be done.

11.6.3.3 Example of Software Standby Mode Application

Figure 11.5 shows an example of entry to software standby mode by the falling edge of the IRQn pin, and exit from software standby mode by the rising edge of the IRQn pin.

In this example, an IRQn interrupt is accepted with the IRQCRi.IRQMD[1:0] bits of the ICU set to 01b (falling edge), and then the IRQCRi.IRQMD[1:0] bits are set to 10b (rising edge). After that, the SBYCR.SSBY bit is set to 1 and the WAIT instruction is executed. Thus entry to software standby mode is completed. After that, exit from software standby mode is initiated by the rising edge of the IRQn pin.

To exit software standby mode, settings of the interrupt controller (ICU) are also necessary. For details, refer to [section 14, Interrupt Controller \(ICUb\)](#).

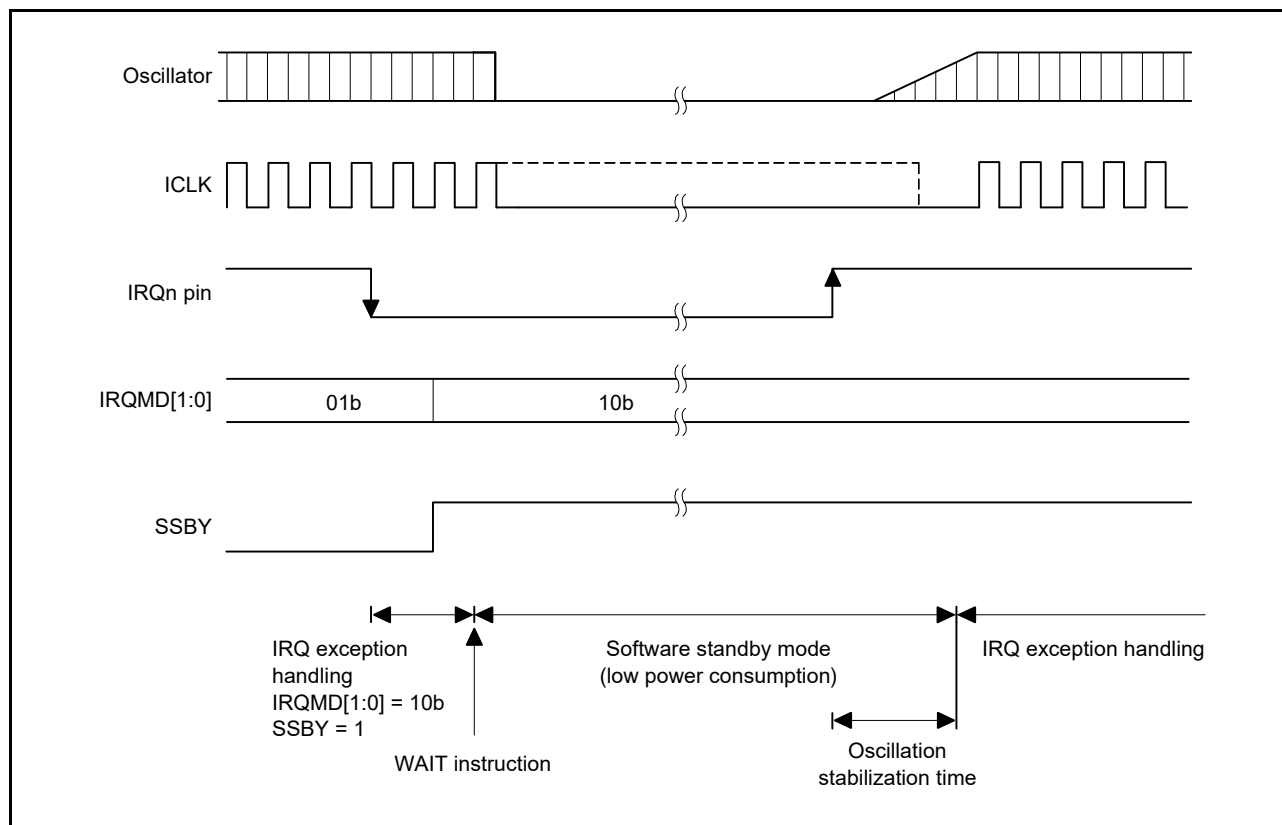


Figure 11.5 Example of Software Standby Mode Application

11.7 Usage Notes

11.7.1 I/O Port States

I/O port states are retained in software standby mode.

11.7.2 Module Stop State of DTC

Before setting the MSTPCRA.MSTPA28 bit to 1, set the DTCST.DTCST bit of the DTC to 0 to avoid activating the DTC.

For details, refer to section 16, Data Transfer Controller (DTCb).

11.7.3 On-Chip Peripheral Module Interrupts

Interrupts do not operate in the module stop state. Therefore, if the module stop state is made after an interrupt request is generated, a CPU interrupt source or a DTC startup source cannot be cleared. For this reason, disable interrupts before entering the module stop state.

11.7.4 Write Access to MSTPCRA, MSTPCRB, and MSTPCRC

Write accesses to MSTPCRA, MSTPCRB, and MSTPCRC should be made only by the CPU.

11.7.5 Timing of WAIT Instructions

The WAIT instruction is executed before completion of the preceding register write. The WAIT instruction being executed before the register setting is modified may cause unintended operation. To avoid this, always execute the WAIT instruction after confirming that the last register setting is done.

11.7.6 Rewrite the Register by DTC in Sleep Mode

Depending on the settings of the OFS0.IWDTSLCSTP bit and IWDTCTSTPR.SLCSTP bit, the IWDT may also stop in sleep mode. To avoid this, do not set up the DTC to rewrite any registers related to the IWDT in sleep mode.

12. Register Write Protection Function

The register write protection function protects important registers from being overwritten for in case a program runs out of control. The registers to be protected are set with the protect register (PRCR).

Table 12.1 lists the association between the PRCR bits and the registers to be protected.

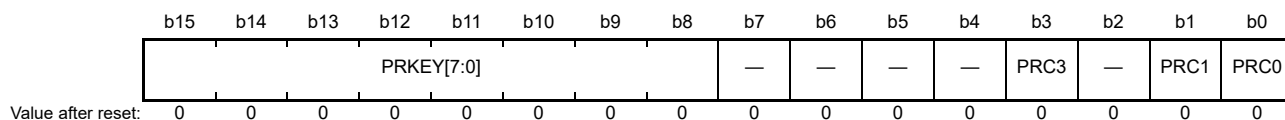
Table 12.1 Association between PRCR Bits and Registers to be Protected

PRCR Bit	Register to be Protected
PRC0	Registers related to the clock generation circuit: SCKCR, SCKCR3, PLLCR, PLLCR2, MOSCCR, LOCOCR, ILOCOCR, HOCOCC, OSTDCR, OSTDSR, CKOCR, LOCOTRR2, ILOCOTRR, HOCOTRR0
PRC1	- Register related to the operating modes: SYSCR1 - Registers related to low power consumption functions: SBYCR, MSTPCRA, MSTPCRB, MSTPCRC, OPCCR - Registers related to the clock generation circuit: MOFCR, MOSCWTCR - Software reset register: SWRR
PRC3	Registers related to the LVD: LVCMPCR, LVDLVLR, LVD1CR0, LVD1CR1, LVD1SR, LVD2CR0, LVD2CR1, LVD2SR

12.1 Register Descriptions

12.1.1 Protect Register (PRCR)

Address(es): SYSTEM.PRCR 0008 03FEh



Bit	Symbol	Bit Name	Description	R/W
b0	PRC0	Protect Bit 0	Enables writing to the register related to the clock generation circuit. 0: Write disabled 1: Write enabled	R/W
b1	PRC1	Protect Bit 1	Enables writing to the registers related to operating modes, low power consumption functions, the clock generation circuit, and software reset. 0: Write disabled 1: Write enabled	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	PRC3	Protect Bit 3	Enables writing to the registers related to the LVD. 0: Write disabled 1: Write enabled	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	PRKEY[7:0]	PRC Key Code	These bits control permission and prohibition of writing to the PRCR register. To modify the PRCR register, write A5h to the 8 higher-order bits and the desired value to the 8 lower-order bits as a 16-bit unit.	R/W*1

Note 1. Write data is not retained.

PRCi Bits (Protect Bit i) (i = 0, 1, 3)

These bits enable or disable writing to the corresponding registers to be protected.

Setting the PRCi bits to 1 and 0 enable and disable writing to the corresponding registers to be protected, respectively.

13. Exception Handling

13.1 Exception Events

During execution of a program by the CPU, the occurrence of a certain event may cause execution of that program to be suspended and execution of another program to be started. Such kinds of events are called exception events.

The RXv2 CPU supports seven types of exceptions. The types of exception events are shown in Figure 13.1.

The occurrence of an exception causes the processor mode to shift to supervisor mode.

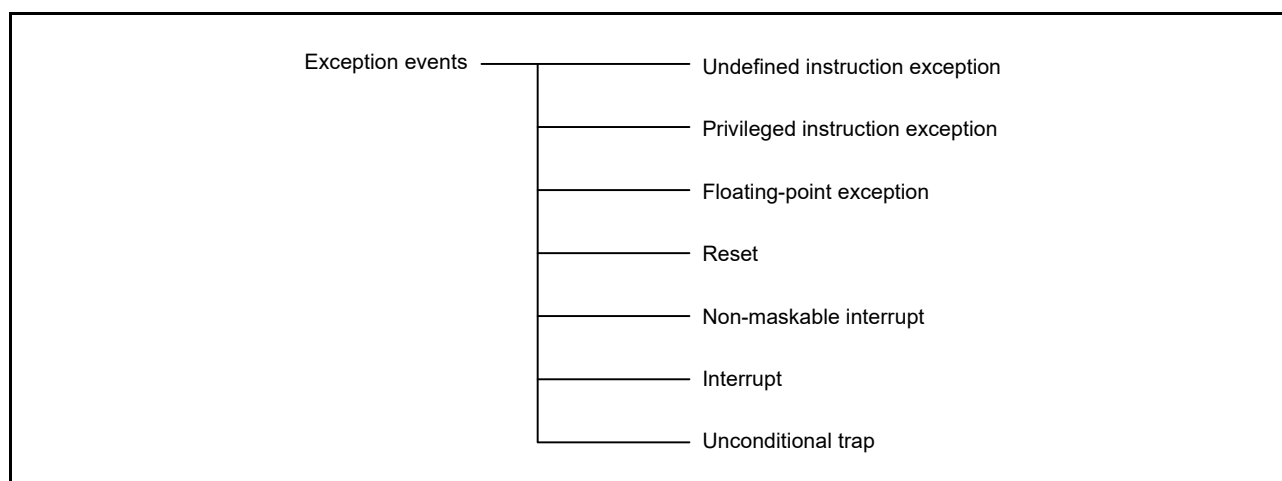


Figure 13.1 Types of Exception Events

13.1.1 Undefined Instruction Exception

An undefined instruction exception occurs when execution of an undefined instruction (an instruction not implemented) is detected.

13.1.2 Privileged Instruction Exception

A privileged instruction exception occurs when execution of a privileged instruction is detected in user mode. Privileged instructions can be executed only in supervisor mode.

13.1.3 Floating-Point Exception

Floating-point exceptions include the five exception events (overflow, underflow, inexact, division-by-zero, and invalid operation) specified in the IEEE754 standard and another floating-point exception that is generated on detection of unimplemented processing. The exception handling of floating-point exceptions is prohibited when the EX, EU, EZ, EO, or EV bit in FPSW is 0.

13.1.4 Reset

A reset is generated by input of a reset signal to the CPU. This has the highest priority of any exception and is always accepted.

13.1.5 Non-Maskable Interrupt

The non-maskable interrupt is generated by input of a non-maskable interrupt signal to the CPU and is only used when a fatal fault is considered to have occurred in the system. Never use the non-maskable interrupt with an attempt to return to the program that was being executed at the time of interrupt generation after the exception handling routine is ended.

13.1.6 Interrupt

Interrupts are generated by the input of interrupt signals to the CPU. A fast interrupt can be selected as the interrupt with the highest priority. In the case of the fast interrupt, hardware pre-processing and hardware post-processing are handled fast. The priority level of the fast interrupt is 15 (the highest). The exception handling of interrupts is masked when the I bit in PSW is 0.

13.1.7 Unconditional Trap

An unconditional trap is generated when the INT or BRK instruction is executed.

13.2 Exception Handling Procedure

In the exception handling, part of the processing is handled automatically by hardware and part of it is handled by a program (exception handling routine) that has been written by the user. Figure 13.2 shows the processing procedure when an exception other than a reset is accepted.

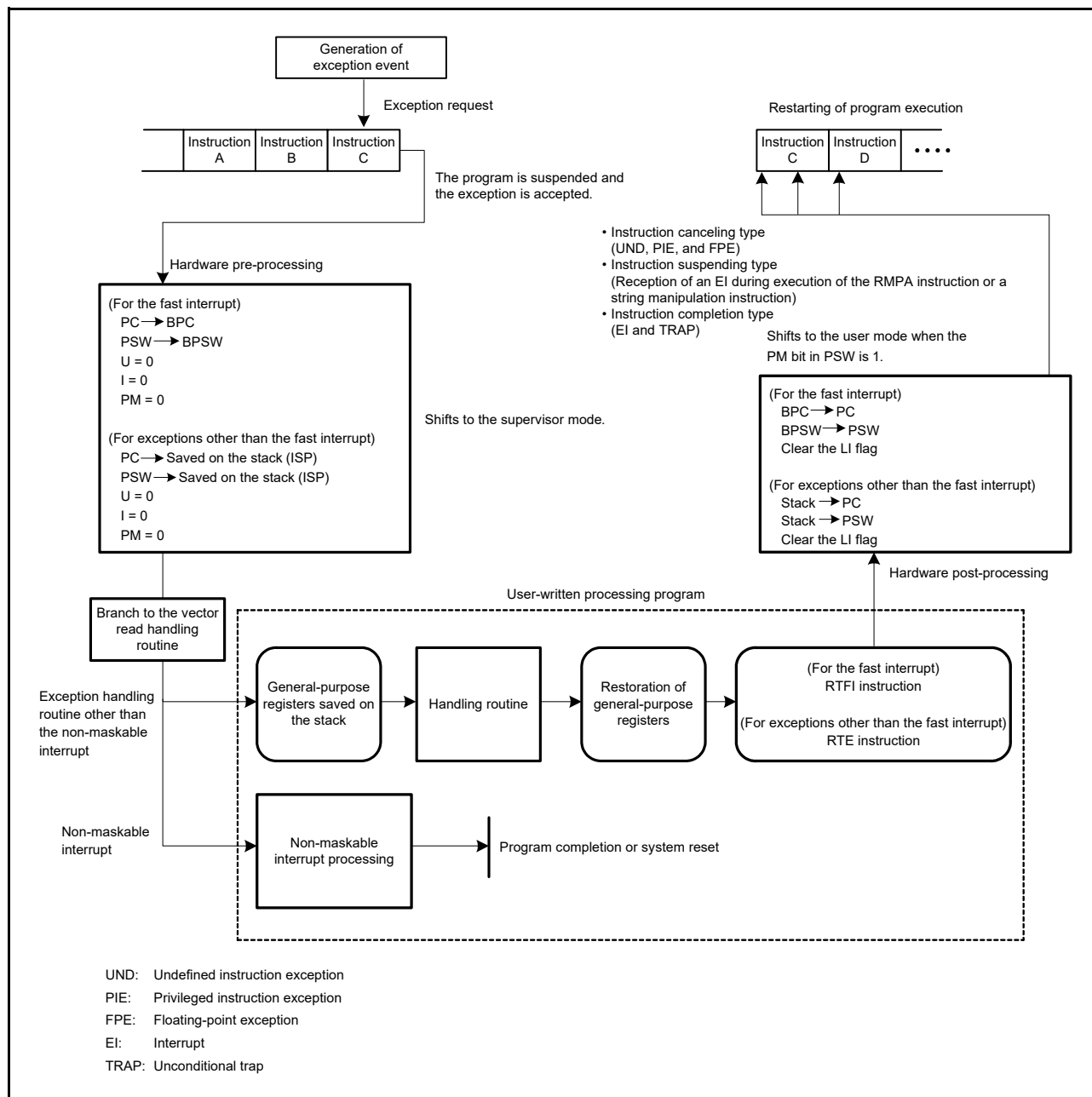


Figure 13.2 Outline of Exception Handling Procedure

When an exception is accepted, hardware processing by the RXv2 CPU is followed by access to the vector to acquire the address of the branch destination. In the vector, a vector address is allocated to each exception, and the branch destination address of the exception handling routine is written to each vector address.

Hardware pre-processing by the RXv2 CPU handles saving of the contents of the program counter (PC) and processor status word (PSW). In the case of a fast interrupt, the contents are saved in the backup PC (BPC) and the backup PSW (BPSW), respectively. In the case of exceptions other than a fast interrupt, the contents are saved in the stack area.

General purpose registers and control registers other than the PC and PSW that are to be used within the exception handling routine must be saved on the stack by a user program at the start of the exception handling routine.

On completion of processing by an exception handling routine, registers saved on the stack are restored and the RTE instruction is executed to restore execution from the exception handling routine to the original program. For return from a fast interrupt, the RTFI instruction is used instead. In the case of a non-maskable interrupt, however, finish the program or reset the system without returning to the original program.

Hardware post-processing by the RXv2 CPU handles restoration of the contents of PC and PSW. In the case of a fast interrupt, the values of BPC and BPSW are restored to PC and PSW, respectively. In the case of exceptions other than a fast interrupt, the values are restored from the stack to PC and PSW.

13.3 Acceptance of Exception Events

When an exception occurs, the CPU suspends the execution of the program and processing branches to the exception handling routine.

13.3.1 Acceptance Timing and Saved PC Value

Table 13.1 lists the timing of acceptance and the program counter (PC) value to be saved for each exception event.

Table 13.1 Acceptance Timing and Saved PC Value

Exception Event		Type of Handling	Acceptance Timing	Value Saved in BPC or on the Stack
Undefined instruction exception		Instruction canceling type	During instruction execution	PC value of the instruction that generated the exception
Privileged instruction exception		Instruction canceling type	During instruction execution	PC value of the instruction that generated the exception
Floating-point exception		Instruction canceling type	During instruction execution	PC value of the instruction that generated the exception
Reset		Instruction abandonment type	Any machine cycle	None
Non-maskable interrupt	During execution of the RMPA, SCMPU, SMOVB, SMOVF, SMOVU, SSTR, SUNTIL, and SWHILE instructions	Instruction suspending type	During instruction execution	PC value of the instruction being executed
	Other than above	Instruction completion type	At the next break between instructions	PC value of the next instruction
Interrupt	During execution of the RMPA, SCMPU, SMOVB, SMOVF, SMOVU, SSTR, SUNTIL, and SWHILE instructions	Instruction suspending type	During instruction execution	PC value of the instruction being executed
	Other than above	Instruction completion type	At the next break between instructions	PC value of the next instruction
Unconditional trap		Instruction completion type	At the next break between instructions	PC value of the next instruction

13.3.2 Vector and Site for Saving the Values in the PC and PSW

The vector for each type of exception and the site for saving the values of the program counter (PC) and processor status word (PSW) are listed in Table 13.2. The addresses where the exception vector table and interrupt vector table start must be set. For details, see section 2.6, Vector Table.

Table 13.2 Vector and Site for Saving the Values in the PC and PSW

Exception		Vector	Site for Saving the Values in the PC and PSW
Undefined instruction exception		Exception vector table (EXTB)	Stack
Privileged instruction exception		Exception vector table (EXTB)	Stack
Floating-point exception		Exception vector table (EXTB)	Stack
Reset		Exception vector table (EXTB)	Nowhere
Non-maskable interrupt		Exception vector table (EXTB)	Stack
Interrupt	Fast interrupt	FINTV	BPC and BPSW
	Other than above	Interrupt vector table (INTB)	Stack
Unconditional trap		Interrupt vector table (INTB)	Stack

13.4 Hardware Processing for Accepting and Returning from Exceptions

This section describes the hardware processing for accepting and returning from exceptions other than a reset.

(1) Hardware Pre-Processing for Accepting an Exception

(a) Saving PSW

- For a fast interrupt
PSW → BPSW
- For exceptions other than a fast interrupt
PSW → Stack

Note: The values in FPSW are not saved by hardware pre-processing. Therefore, if floating-point instructions are to be used within an exception handling routine, the user must save these values on the stack within the exception handling routine.

(b) Updating PM, U, and I Bits in PSW

I: Set to 0

U: Set to 0

PM: Set to 0

(c) Saving PC

- For a fast interrupt
PC → BPC
- For exceptions other than a fast interrupt
PC → Stack

(d) Setting Branch Destination Address of Exception Handling Routine in PC

Processing is shifted to the exception handling routine by acquiring the vector corresponding to the exception and then branching accordingly.

(2) Hardware Post-Processing for Execution of RTE and RTFI Instructions

(a) Restoring PSW

- For a fast interrupt
BPSW → PSW
- For exceptions other than a fast interrupt
Stack → PSW

(b) Restoring PC

- For a fast interrupt
BPC → PC
- For exceptions other than a fast interrupt
Stack → PC

(c) Clearing the LI flag

13.5 Hardware Pre-Processing

The hardware pre-processing from reception of each exception request to execution of the associated exception handling routine are explained below.

13.5.1 Undefined Instruction Exception

1. The value of the processor status word (PSW) is saved on the stack (ISP).
2. The processor mode select bit (PM), the stack pointer select bit (U), and the interrupt enable bit (I) in PSW are cleared to 0.
3. The value of the program counter (PC) is saved on the stack (ISP).
4. The vector is fetched from the value of EXTB + address 0000 005Ch.
5. The fetched vector is set to the PC and processing branches to the exception handling routine.

13.5.2 Privileged Instruction Exception

1. The value in the processor status word (PSW) is saved on the stack (ISP).
2. The processor mode select bit (PM), the stack pointer select bit (U), and the interrupt enable bit (I) in PSW are cleared to 0.
3. The value of the program counter (PC) is saved on the stack (ISP).
4. The vector is fetched from the value of EXTB + address 0000 0050h.
5. The fetched vector is set to the PC and processing branches to the exception handling routine.

13.5.3 Floating-Point Exception

1. The value in the processor status word (PSW) is saved on the stack (ISP).
2. The processor mode select bit (PM), the stack pointer select bit (U), and the interrupt enable bit (I) in PSW are cleared to 0.
3. The value of the program counter (PC) is saved on the stack (ISP).
4. The vector is fetched from the value of EXTB + address 0000 0064h.
5. The fetched vector is set to the PC and processing branches to the exception handling routine.

13.5.4 Reset

1. The control registers are initialized.
2. The vector is fetched from address FFFF FFFCh.
3. The fetched vector is set to the PC.

13.5.5 Non-Maskable Interrupt

1. The value of the processor status word (PSW) is saved on the stack (ISP).
2. The processor mode select bit (PM), the stack pointer select bit (U), and the interrupt enable bit (I) in PSW are cleared to 0.
3. If the interrupt was generated during the execution of an RMPA, SCMPU, SMOVB, SMOVF, SMOVU, SSTR, SUNTIL, or SWHILE instruction, the value of the program counter (PC) for that instruction is saved on the stack (ISP). For other instructions, the PC value of the next instruction is saved.
4. The processor interrupt priority level bits (IPL[3:0]) in PSW are set to Fh.
5. The vector is fetched from the value of EXTB + address 0000 0078h.
6. The fetched vector is set to the PC and processing branches to the exception handling routine.

13.5.6 Interrupt

1. The value of the processor status word (PSW) is saved on the stack (ISP) or, for the fast interrupt, in the backup PSW (BPSW).
2. The processor mode select bit (PM), the stack pointer select bit (U), and the interrupt enable bit (I) in PSW are cleared to 0.
3. If the interrupt was generated during the execution of an RMPA, SCMPU, SMOVB, SMOVF, SMOVU, SSTR, SUNTIL, or SWHILE instruction, the value of the program counter (PC) for that instruction is saved. For other instructions, the PC value of the next instruction is saved. Saving of the PC is in the backup PC (BPC) for fast interrupts.
4. The processor interrupt priority level bits (IPL[3:0]) in PSW indicate the interrupt priority level of the interrupt.
5. The vector for an interrupt source other than the fast interrupt is fetched from the interrupt vector table. For the fast interrupt, the address is fetched from the fast interrupt vector register (FINTV).
6. The fetched vector is set to the PC and processing branches to the exception handling routine.

13.5.7 Unconditional Trap

1. The value in the processor status word (PSW) is saved on the stack (ISP).
2. The processor mode select bit (PM), the stack pointer select bit (U), and the interrupt enable bit (I) in PSW are cleared to 0.
3. The value of the program counter (PC) for the next instruction is saved on the stack (ISP).
4. For the INT instruction, the value at the vector corresponding to the INT instruction number is fetched from the interrupt vector table.
For the BRK instruction, the value at the vector from the start address is fetched from the interrupt vector table.
5. The fetched vector is set to the PC and processing branches to the exception handling routine.

13.6 Return from Exception Handling Routine

Executing the instruction listed in Table 13.3 at the end of the corresponding exception handling routine restores the values of the program counter (PC) and processor status word (PSW) that were saved on the stack or in the control registers (BPC and BPSW) immediately before the exception handling sequence.

Table 13.3 Return from Exception Handling Routine

Exception		Instruction for Return
Undefined instruction exception		RTE
Privileged instruction exception		RTE
Floating-point exception		RTE
Reset		Return is impossible
Non-maskable interrupt		Prohibited
Interrupt	Fast interrupt	RTFI
	Other than above	RTE
Unconditional trap		RTE

13.7 Priority of Exception Events

The priority of exception events is listed in Table 13.4. When multiple exceptions are generated at the same time, the exception with the highest priority is accepted first.

Table 13.4 Priority of Exception Events

Priority		Exception Event
↑ High Low	1	Reset
	2	Non-maskable interrupt
	3	Interrupt
	4	Undefined instruction exception Privileged instruction exception
	5	Unconditional trap
	6	Floating-point exception

14. Interrupt Controller (ICUb)

14.1 Overview

The interrupt controller receives interrupt requests from peripheral modules and external pins, and generates an interrupt request to the CPU and a transfer request to the DTC.

Table 14.1 lists the specifications of the interrupt controller, and Figure 14.1 shows a block diagram of the interrupt controller.

Table 14.1 Specifications of Interrupt Controller

Item	Description	
Interrupts	Peripheral function interrupts	- Interrupts from peripheral modules - Interrupt detection: Edge detection/level detection Edge detection or level detection is fixed for each source of connected peripheral modules.
	External pin interrupts	- Interrupts from pins IRQ0 to IRQ7 - Number of sources: 8 - Interrupt detection: Low level/falling edge/rising edge/rising and falling edges One of these detection methods can be set for each source. - Digital filter function: Supported
	Software interrupt	- Interrupt generated by writing to a register - One interrupt source
	Interrupt priority	Specified by registers.
	Fast interrupt function	Faster interrupt processing of the CPU can be set only for a single interrupt source.
	DTC control	Interrupt sources can be used to start the DTC.*1
Non-maskable interrupts	NMI pin interrupt	- Interrupt from the NMI pin - Interrupt detection: Falling edge/rising edge - Digital filter function: Supported
	Oscillation stop detection interrupt	Interrupt on detection of oscillation having stopped
	IWDT underflow/refresh error	Interrupt on an underflow of the down counter or occurrence of a refresh error
	Voltage monitoring 1 interrupt	Voltage monitoring interrupt of voltage monitoring circuit 1 (LVD1)
	Voltage monitoring 2 interrupt	Voltage monitoring interrupt of voltage monitoring circuit 2 (LVD2)
Return from low power consumption states	Sleep mode Deep sleep mode	Return is initiated by any non-maskable interrupt or any interrupt.
	Software standby mode	Return is initiated by the non-maskable interrupt, external pin interrupts (IRQ0 to IRQ7), or peripheral interrupts (voltage monitoring 1, voltage monitoring 2).

Note 1. For the DTC trigger, refer to Table 14.3, Interrupt Vector Table.

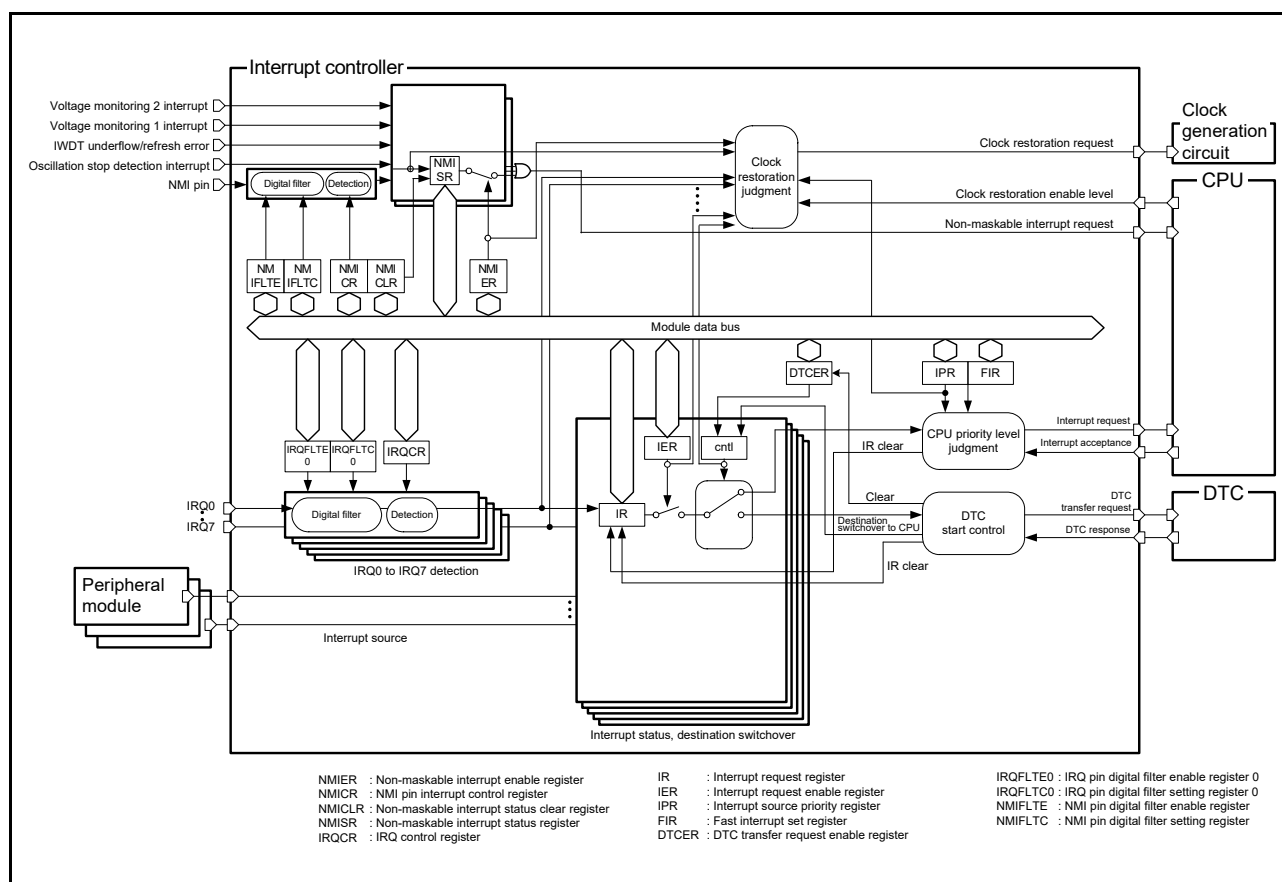


Figure 14.1 Block Diagram of Interrupt Controller

Table 14.2 lists the input/output pins of the interrupt controller.

Table 14.2 Pin Configuration of Interrupt Controller

Pin Name	I/O	Description
NMI	Input	Non-maskable interrupt request pin
IRQ0 to IRQ7	Input	External interrupt request pins

14.2 Register Descriptions

14.2.1 Interrupt Request Register n (IRn) (n = interrupt vector number)

Address(es): ICU.IR016 0008 7010h to ICU.IR255 0008 70FFh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	IR
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	IR	Interrupt Status Flag	0: No interrupt request is generated 1: An interrupt request is generated	R/(W) *1
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. For an edge detection interrupt, only 0 can be written to this bit; do not write 1.
For a level detection interrupt, neither 0 nor 1 can be written.

IRn is provided for each interrupt source, where “n” indicates the interrupt vector number.

For the correspondence between interrupt sources and interrupt vector numbers, see Table 14.3, Interrupt Vector Table.

IR Flag (Interrupt Status Flag)

This flag is the status flag of an individual interrupt request. This flag is set to 1 when the corresponding interrupt request is generated. To detect an interrupt request, the interrupt request output should be enabled by the corresponding peripheral module interrupt enable bit.

There are two interrupt request detection methods: edge detection and level detection. For interrupts from peripheral modules, either edge detection or level detection is determined per interrupt source. For interrupts from IRQi (i = 0 to 7) pins, edge detection or level detection is selected by setting the corresponding IRQCRi.IRQMD[1:0] bits. For detection of the various interrupt sources, see Table 14.3, Interrupt Vector Table.

(1) Edge detection

[Setting condition]

- The flag is set to 1 in response to the generation of an interrupt request from the corresponding peripheral module or IRQi pin. For interrupt generation by the various peripheral modules, refer to the sections describing the modules.

[Clearing conditions]

- The flag is cleared to 0 when the interrupt request destination accepts the interrupt request.
- The IR flag is cleared to 0 by writing 0 to it. Note, however, that writing 0 to the IR flag is prohibited if the destination of the interrupt request is the DTC.

(2) Level detection

[Setting condition]

- The flag remains set to 1 while an interrupt request is being sent from the corresponding peripheral module or IRQi pin. For interrupt generation by the various peripheral modules, refer to the sections describing the modules.

[Clearing condition]

- The flag is cleared to 0 when the source of the interrupt request is cleared (it is not cleared when the interrupt request destination accepts the interrupt request). For clearing interrupts from the various peripheral modules, refer to the sections describing the modules.

When level detection has been selected for an IRQi pin, the interrupt request is withdrawn by driving the IRQi pin high. Do not write 0 or 1 to the IR flag while level detection is selected.

14.2.2 Interrupt Request Enable Register m (IERm) (m = 02h to 1Fh)

Address(es): ICU.IER02 0008 7202h to ICU.IER1F 0008 721Fh

b7	b6	b5	b4	b3	b2	b1	b0
IEN7	IEN6	IEN5	IEN4	IEN3	IEN2	IEN1	IEN0
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	IEN0	Interrupt Request Enable 0	0: Interrupt request is disabled 1: Interrupt request is enabled	R/W
b1	IEN1	Interrupt Request Enable 1		R/W
b2	IEN2	Interrupt Request Enable 2		R/W
b3	IEN3	Interrupt Request Enable 3		R/W
b4	IEN4	Interrupt Request Enable 4		R/W
b5	IEN5	Interrupt Request Enable 5		R/W
b6	IEN6	Interrupt Request Enable 6		R/W
b7	IEN7	Interrupt Request Enable 7		R/W

Note: Write 0 to the bit that corresponds to the vector number for reservation. These bits are read as 0.

IENj Bit (Interrupt Request Enable j) (j = 0 to 7)

When an IENj bit is 1, the corresponding interrupt request will be output to the destination selected for the request.

When an IENj bit is 0, the corresponding interrupt request will not be output to the destination selected for the request.

The setting of an IENj bit does not affect the IRn.IR flag (n = interrupt vector number). Even if the corresponding IENj bit is 0, the IR flag value changes according to the descriptions in section 14.2.1, Interrupt Request Register n (IRn) (n = interrupt vector number).

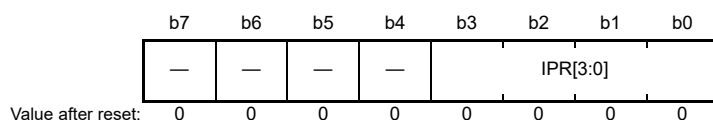
The IERm.IENj bit is set for each request source (vector number).

For the correspondence between interrupt sources and IERm.IENj bits, see Table 14.3, Interrupt Vector Table.

For the procedure for setting IERm.IENj bits during the selection of destinations for interrupt requests, refer to section 14.4.3, Selecting Interrupt Request Destinations.

14.2.3 Interrupt Source Priority Register n (IPRn) (n = interrupt vector number)

Address(es): ICU.IPR000 0008 7300h to ICU.IPR255 0008 73FFh



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	IPR[3:0]	Interrupt Priority Level Select	b3 b0 0 0 0 0: Level 0 (interrupt disabled)*1 0 0 0 1: Level 1 0 0 1 0: Level 2 0 0 1 1: Level 3 0 1 0 0: Level 4 0 1 0 1: Level 5 0 1 1 0: Level 6 0 1 1 1: Level 7 1 0 0 0: Level 8 1 0 0 1: Level 9 1 0 1 0: Level 10 1 0 1 1: Level 11 1 1 0 0: Level 12 1 1 0 1: Level 13 1 1 1 0: Level 14 1 1 1 1: Level 15 (highest)	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. When the interrupt is specified as a fast interrupt, it can be issued even if the priority level is level 0.

For the correspondence between interrupt sources and IPRn registers, see Table 14.3, Interrupt Vector Table.

IPR[3:0] Bits (Interrupt Priority Level Select)

These bits specify the priority level of the corresponding interrupt source.

Priority levels specified by the IPR[3:0] bits are used only to determine the priority of interrupt requests to be transferred to the CPU, and do not affect transfer requests to the DTC.

The CPU accepts only interrupt requests higher than the priority level specified by the IPL[3:0] bits in PSW, and handles accepted interrupts.

If two or more interrupt requests are generated at the same time, their priority levels are compared with the value of the IPR[3:0] bits. If interrupt requests of the same priority level are generated at the same time, an interrupt source with a smaller vector number takes precedence.

These bits should be written to while an interrupt request is disabled (IERm.IENj bit = 0 (m = 02h to 1Fh; j = 0 to 7)).

14.2.4 Fast Interrupt Set Register (FIR)

Address(es): ICU.FIR 0008 72F0h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FIEN	—	—	—	—	—	—	—	FVCT[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	FVCT[7:0]	Fast Interrupt Vector Number	Specify the vector number of an interrupt source to be a fast interrupt.	R/W
b14 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	FIEN	Fast Interrupt Enable	0: Fast interrupt is disabled 1: Fast interrupt is enabled	R/W

The fast interrupt function based on the FIR register setting is applicable only to interrupts to the CPU. It will not affect any transfer request to the DTC.

Before writing to this register, be sure to disable interrupt requests (IERm.IENj bit = 0 (m = 02h to 1Fh; j = 0 to 7)).

FVCT[7:0] Bits (Fast Interrupt Vector Number)

The FVCT[7:0] bits specify the vector number of an interrupt source that uses the fast interrupt function.

FIEN Bit (Fast Interrupt Enable)

This bit enables the fast interrupt.

Setting this bit to 1 makes the interrupt request of the vector number specified by the FVCT[7:0] bits a fast interrupt.

When an interrupt request of the vector number specified by the FVCT[7:0] bits is generated and the interrupt request destination is the CPU while the FIEN bit is 1, the interrupt request is output to the CPU as a fast interrupt regardless of the setting of the IPRn register (n = interrupt vector number). When using the fast interrupt for returning from the software standby mode, see section 14.6.2, Return from Software Standby Mode.

If the setting of the IERm.IENj bit has disabled interrupt requests from the interrupt source with the vector number in this register, fast interrupt requests are not output to the CPU.

For settable vector numbers, see Table 14.3, Interrupt Vector Table.

Do not write any reserved vector numbers to the FVCT[7:0] bits.

For details on the fast interrupt, see section 13, Exception Handling, and section 14.4.6, Fast Interrupt.

14.2.5 Software Interrupt Generation Register (SWINTR)

Address(es): ICU.SWINTR 0008 72E0h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	SWINT
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SWINT	Software Interrupt Generation	This bit is read as 0. Writing 1 issues a software interrupt request. Writing 0 to this bit has no effect.	R/(W) *1
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only 1 can be written.

SWINT Bit (Software Interrupt Generation)

When 1 is written to the SWINT bit, the interrupt request register 027 (IR027) is set to 1.

If 1 is written to the SWINT bit when the DTC transfer request enable register 027 (DTCER027) is set to 0, an interrupt to the CPU is generated.

If 1 is written to the SWINT bit when the DTC transfer request enable register 027 (DTCER027) is set to 1, a DTC transfer request is issued.

14.2.6 DTC Transfer Request Enable Register n (DTCERn) (n = interrupt vector number)

Address(es): ICU.DTCER027 0008 711Bh to ICU.DTCER255 0008 71FFh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	DTCE
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DTCE	DTC Transfer Request Enable	0: The corresponding interrupt source is not selected as the DTC trigger. 1: The corresponding interrupt source is selected as the DTC trigger.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

See Table 14.3, Interrupt Vector Table, for the interrupt sources that are selectable as the DTC trigger.

DTCE Bit (DTC Transfer Request Enable)

When the DTCE bit is set to 1, the corresponding interrupt source is selected as the DTC trigger.

[Setting condition]

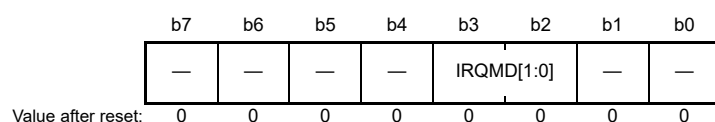
- When 1 is written to the DTCE bit

[Clearing conditions]

- When the specified number of transfers is completed (for the chain transfer, the number of transfers for the last chain transfer is completed)
- When 0 is written to the DTCE bit

14.2.7 IRQ Control Register i (IRQCRi) (i = 0 to 7)

Address(es): ICU.IRQCR0 0008 7500h to ICU.IRQCR7 0008 7507h



Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b3, b2	IRQMD[1:0]	IRQ Detection Sense Select	b3 b2 0 0: Low level 0 1: Falling edge 1 0: Rising edge 1 1: Rising and falling edges	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Only change the settings of this register while the corresponding interrupt request enable bit is prohibiting the interrupt request (IEN_j bit in IER_m (m = 02h to 1Fh; j = 0 to 7) is 0). After changing the setting, clear the IR flag in IR_n before setting the interrupt enable bit. However, when the change is to the low level, the IR flag does not require clearing.

IRQMD[1:0] Bits (IRQ Detection Sense Select)

These bits select the interrupt detection sensing method of IRQ_i pin.

For the external pin interrupt detection setting, see section 14.4.8, External Pin Interrupts.

14.2.8 IRQ Pin Digital Filter Enable Register 0 (IRQFLTE0)

Address(es): ICU.IRQFLTE0 0008 7510h

	b7	b6	b5	b4	b3	b2	b1	b0
	FLTEN 7	FLTEN 6	FLTEN 5	FLTEN 4	FLTEN 3	FLTEN 2	FLTEN 1	FLTEN 0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FLTEN0	IRQ0 Digital Filter Enable	0: Digital filter is disabled 1: Digital filter is enabled	R/W
b1	FLTEN1	IRQ1 Digital Filter Enable		R/W
b2	FLTEN2	IRQ2 Digital Filter Enable		R/W
b3	FLTEN3	IRQ3 Digital Filter Enable		R/W
b4	FLTEN4	IRQ4 Digital Filter Enable		R/W
b5	FLTEN5	IRQ5 Digital Filter Enable		R/W
b6	FLTEN6	IRQ6 Digital Filter Enable		R/W
b7	FLTEN7	IRQ7 Digital Filter Enable		R/W

FLTENi Bit (IRQi Digital Filter Enable) (i = 0 to 7)

This bit enables the digital filter used for the IRQi pin.

The digital filter is enabled when the FLTENi bit is 1, and disabled when the FLTENi bit is 0.

The IRQi pin level is sampled at the sampling clock cycle specified with the IRQFLTC0.FCLKSELi[1:0] bits. When the sampled level matches three times, the output level from the digital filter changes.

For details of the digital filter, see section 14.4.7, Digital Filter.

14.2.9 IRQ Pin Digital Filter Setting Register 0 (IRQFLTC0)

Address(es): ICU.IRQFLTC0 0008 7514h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
FCLKSEL7[1:0]		FCLKSEL6[1:0]		FCLKSEL5[1:0]		FCLKSEL4[1:0]		FCLKSEL3[1:0]		FCLKSEL2[1:0]		FCLKSEL1[1:0]		FCLKSEL0[1:0]	
0		0		0		0		0		0		0		0	

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b1, b0	FCLKSEL0[1:0]	IRQ0 Digital Filter Sampling Clock	0 0: PCLK 0 1: PCLK/8	R/W
b3, b2	FCLKSEL1[1:0]	IRQ1 Digital Filter Sampling Clock	1 0: PCLK/32	R/W
b5, b4	FCLKSEL2[1:0]	IRQ2 Digital Filter Sampling Clock	1 1: PCLK/64	R/W
b7, b6	FCLKSEL3[1:0]	IRQ3 Digital Filter Sampling Clock		R/W
b9, b8	FCLKSEL4[1:0]	IRQ4 Digital Filter Sampling Clock		R/W
b11, b10	FCLKSEL5[1:0]	IRQ5 Digital Filter Sampling Clock		R/W
b13, b12	FCLKSEL6[1:0]	IRQ6 Digital Filter Sampling Clock		R/W
b15, b14	FCLKSEL7[1:0]	IRQ7 Digital Filter Sampling Clock		R/W

FCLKSELi[1:0] Bits (IRQi Digital Filter Sampling Clock) (i = 0 to 7)

These bits select the cycle of the digital filter sampling clock for the IRQi pin.

The sampling clock cycle can be selected from among the PCLK (every cycle), PCLK/8 (once every eight cycles), PCLK/32 (once every 32 cycles), and PCLK/64 (once every 64 cycles).

For details of the digital filter, see section 14.4.7, Digital Filter.

14.2.10 Non-Maskable Interrupt Status Register (NMISR)

Address(es): ICU.NMISR 0008 7580h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	LVD2S T	LVD1S T	IWDTS T	—	OSTST	NMIST
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NMIST	NMI Status Flag	0: NMI pin interrupt is not requested 1: NMI pin interrupt is requested	R
b1	OSTST	Oscillation Stop Detection Interrupt Status Flag	0: Oscillation stop detection interrupt is not requested 1: Oscillation stop detection interrupt is requested	R
b2	—	Reserved	This bit is read as 0. Writing to this bit has no effect.	R
b3	IWDTS	IWDT Underflow/Refresh Error Status Flag	0: IWDT underflow/refresh error interrupt is not requested 1: IWDT underflow/refresh error interrupt is requested	R
b4	LVD1ST	Voltage Monitoring 1 Interrupt Status Flag	0: Voltage monitoring 1 interrupt is not requested 1: Voltage monitoring 1 interrupt is requested	R
b5	LVD2ST	Voltage Monitoring 2 Interrupt Status Flag	0: Voltage monitoring 2 interrupt is not requested 1: Voltage monitoring 2 interrupt is requested	R
b7, b6	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R

The NMISR register monitors the status of a non-maskable interrupt source. Writing to the NMISR register is ignored. The setting in the non-maskable interrupt enable register (NMIER) does not affect the status flags in NMISR.

Before the end of the non-maskable interrupt handler, read the NMISR register and confirm the generation status of other non-maskable interrupts. Be sure to confirm that all of the bits in the NMISR register are set to 0 before the end of the handler.

NMIST Flag (NMI Status Flag)

This flag indicates the NMI pin interrupt request.

The NMIST flag is read-only, and cleared by the NMICLR.NMICLR bit.

[Setting condition]

- When an edge specified by the NMICR.NMIMD bit is input to the NMI pin

[Clearing condition]

- When 1 is written to the NMICLR.NMICLR bit

OSTST Flag (Oscillation Stop Detection Interrupt Status Flag)

This flag indicates the oscillation stop detection interrupt request.

The OSTST flag is read-only, and cleared by the NMICLR.OSTCLR bit.

[Setting condition]

- When the oscillation stop detection interrupt is generated

[Clearing condition]

- When 1 is written to the NMICLR.OSTCLR bit

IWDTS Flag (IWDT Underflow/Refresh Error Status Flag)

This flag indicates the IWDT underflow/refresh error interrupt request.

The IWDTS flag is read-only, and cleared by the NMICLR.IWDTCLR bit.

[Setting condition]

- When the IWDT underflow/refresh error interrupt is generated while this interrupt is enabled at its source.

[Clearing condition]

- When 1 is written to the NMICLR.IWDTCLR bit

LVD1ST Flag (Voltage Monitoring 1 Interrupt Status Flag)

This flag indicates the request for voltage monitoring 1 interrupt.

The LVD1ST flag is read-only, and cleared by the NMICLR.LVD1CLR bit.

[Setting condition]

- When the voltage monitoring 1 interrupt is generated while this interrupt is enabled at its source.

[Clearing condition]

- When 1 is written to the NMICLR.LVD1CLR bit

LVD2ST Flag (Voltage Monitoring 2 Interrupt Status Flag)

This flag indicates the request for voltage monitoring 2 interrupt.

The LVD2ST flag is read-only, and cleared by the NMICLR.LVD2CLR bit.

[Setting condition]

- When the voltage monitoring 2 interrupt is generated while this interrupt is enabled at its source.

[Clearing condition]

- When 1 is written to the NMICLR.LVD2CLR bit

14.2.11 Non-Maskable Interrupt Enable Register (NMIER)

Address(es): ICU.NMIER 0008 7581h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	LVD2E N	LVD1E N	IWDTE N	—	OSTEN	NMIEN
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NMIEN	NMI Pin Interrupt Enable	0: NMI pin interrupt is disabled 1: NMI pin interrupt is enabled	R/(W) *1
b1	OSTEN	Oscillation Stop Detection Interrupt Enable	0: Oscillation stop detection interrupt is disabled 1: Oscillation stop detection interrupt is enabled	R/(W) *1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	IWDTEN	IWDT Underflow/Refresh Error Enable	0: IWDT underflow/refresh error interrupt is disabled 1: IWDT underflow/refresh error interrupt is enabled	R/(W) *1
b4	LVD1EN	Voltage Monitoring 1 Interrupt Enable	0: Voltage monitoring 1 interrupt is disabled 1: Voltage monitoring 1 interrupt is enabled	R/(W) *1
b5	LVD2EN	Voltage Monitoring 2 Interrupt Enable	0: Voltage monitoring 2 interrupt is disabled 1: Voltage monitoring 2 interrupt is enabled	R/(W) *1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. A 1 can be written to this bit only once, and subsequent write accesses are no longer enabled.

NMIEN Bit (NMI Pin Interrupt Enable)

This bit enables the NMI pin interrupt.

A 1 can be written to this bit only once, and subsequent write accesses are no longer enabled.

Writing 0 to this bit is disabled.

OSTEN Bit (Oscillation Stop Detection Interrupt Enable)

This bit enables the oscillation stop detection interrupt.

A 1 can be written to this bit only once, and subsequent write accesses are no longer enabled.

Writing 0 to this bit is disabled.

IWDTEN Bit (IWDT Underflow/Refresh Error Enable)

This bit enables the IWDT underflow/refresh error interrupt.

A 1 can be written to this bit only once, and subsequent write accesses are no longer enabled.

Writing 0 to this bit is disabled.

LVD1EN Bit (Voltage Monitoring 1 Interrupt Enable)

This bit enables the voltage monitoring 1 interrupt.

A 1 can be written to this bit only once, and subsequent write accesses are no longer enabled.

Writing 0 to this bit is disabled.

LVD2EN Bit (Voltage Monitoring 2 Interrupt Enable)

This bit enables the voltage monitoring 2 interrupt.

A 1 can be written to this bit only once, and subsequent write accesses are no longer enabled.

Writing 0 to this bit is disabled.

14.2.12 Non-Maskable Interrupt Status Clear Register (NMICLR)

Address(es): ICU.NMICLR 0008 7582h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	LVD2C LR	LVD1C LR	IWDTC LR	—	OSTCL R	NMICL R
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NMICLR	NMI Clear	This bit is read as 0. Writing 1 to this bit clears the NMISR.NMIST flag. Writing 0 to this bit has no effect.	R/(W) *1
b1	OSTCLR	OST Clear	This bit is read as 0. Writing 1 to this bit clears the NMISR.OSTST flag. Writing 0 to this bit has no effect.	R/(W) *1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	IWDTCCLR	IWDT Clear	This bit is read as 0. Writing 1 to this bit clears the NMISR.IWDTST flag. Writing 0 to this bit has no effect.	R/(W) *1
b4	LVD1CLR	LVD1 Clear	This bit is read as 0. Writing 1 to this bit clears the NMISR.LVD1ST flag. Writing 0 to this bit has no effect.	R/(W) *1
b5	LVD2CLR	LVD2 Clear	This bit is read as 0. Writing 1 to this bit clears the NMISR.LVD2ST flag. Writing 0 to this bit has no effect.	R/(W) *1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only 1 can be written to this bit.

NMICLR Bit (NMI Clear)

Writing 1 to the NMICLR bit clears the NMISR.NMIST flag. This bit is read as 0.

OSTCLR Bit (OST Clear)

Writing 1 to the OSTCLR bit clears the NMISR.OSTST flag. This bit is read as 0.

IWDTCCLR Bit (IWDT Clear)

Writing 1 to the IWDTCCLR bit clears the NMISR.IWDTST flag. This bit is read as 0.

LVD1CLR Bit (LVD1 Clear)

Writing 1 to the LVD1CLR bit clears the NMISR.LVD1ST flag. This bit is read as 0.

LVD2CLR Bit (LVD2 Clear)

Writing 1 to the LVD2CLR bit clears the NMISR.LVD2ST flag. This bit is read as 0.

14.2.13 NMI Pin Interrupt Control Register (NMICR)

Address(es): ICU.NMICR 0008 7583h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	NMIMD	—	—	—
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b3	NMIMD	NMI Detection Set	0: Falling edge 1: Rising edge	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Change the setting of the NMICR register before the NMI pin interrupt is enabled (before setting the NMIER.NMIEN bit to 1).

NMIMD Bit (NMI Detection Set)

This bit specifies the detection edge of the NMI pin interrupt.

14.2.14 NMI Pin Digital Filter Enable Register (NMIFLTE)

Address(es): ICU.NMIFLTE 0008 7590h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	NFLTE N
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	NFLTEN	NMI Digital Filter Enable	0: Digital filter is disabled 1: Digital filter is enabled	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NFLTEN Bit (NMI Digital Filter Enable)

This bit enables the digital filter used for the NMI pin interrupt.

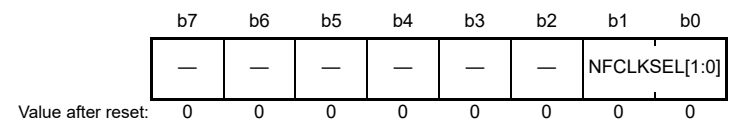
The digital filter is enabled when the NFLTEN bit is 1, and disabled when the NFLTEN bit is 0.

The NMI pin level is sampled at the sampling clock cycle specified with the NMIFLTC.NFCLKSEL[1:0] bits. When the sampled level matches three times, the output level from the digital filter changes.

For details of the digital filter, see section 14.4.7, Digital Filter.

14.2.15 NMI Pin Digital Filter Setting Register (NMIFLTC)

Address(es): ICU.NMIFLTC 0008 7594h



Bit	Symbol	Bit Name	Description	R/W
b1, b0	NFCLKSEL[1:0]	NMI Digital Filter Sampling Clock	b1 b0 0 0: PCLK 0 1: PCLK/8 1 0: PCLK/32 1 1: PCLK/64	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NFCLKSEL[1:0] Bits (NMI Digital Filter Sampling Clock)

These bits select the cycle of the digital filter sampling clock for the NMI pin interrupt. The sampling clock cycle can be selected from among the PCLK (every cycle), PCLK/8 (once every eight cycles), PCLK/32 (once every 32 cycles), and PCLK/64 (once every 64 cycles). For details of the digital filter, see section 14.4.7, Digital Filter.

14.3 Vector Table

There are two types of interrupts detected by the interrupt controller: maskable interrupts and non-maskable interrupts. When the CPU accepts an interrupt or non-maskable interrupt, it acquires a 4-byte vector address from the vector table.

14.3.1 Interrupt Vector Table

The interrupt vector table is placed in the 1024-byte range (4 bytes × 256 sources) beginning at the address specified in the interrupt table register (INTB) of the CPU. Write a value to the INTB register before enabling interrupts. The value written to the INTB register should be a multiple of 4.

Executing an INT instruction or BRK instruction leads to the generation of an unconditional trap. The same range of memory as shown in Table 14.3, Interrupt Vector Table, is used for the vectors for unconditional traps. The vector for BRK instructions is vector 0 while the vector numbers for INT instructions are specifiable as numbers in the range from 0 to 255.

Table 14.3 lists details of the interrupt vectors. Details of the headings in Table 14.3 are listed below.

Item	Description
Source of interrupt request generation	Name of the source for generation of the interrupt request
Name	Name of the interrupt
Vector no.	Vector number for the interrupt
Vector address offset	Value of the offset from the base address for the vector table
Interrupt detection method	"Edge" or "level" as the method for detection of the interrupt
CPU interrupt	"✓" in this column indicates usability as a CPU interrupt.
Start DTC	"✓" in this column indicates usability as a request for DTC transfer.
SSBY return	"✓" in this column indicates usability as a request for return from software-standby mode.
IER	Name of the IER register and bit corresponding to the vector number
IPR	Name of the IPR register corresponding to the interrupt source
DTCER	Name of the DTCER register corresponding to the DTC trigger

Table 14.3 Interrupt Vector Table (1/6)

Source of Interrupt Request Generation	Name	Vector No.*1	Vector Address Offset	Interrupt Detection Method	CPU Interrupt	Start DTC	SSBY Return	IER	IPR	DTCER
—	For an unconditional trap	0	0000h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	1	0004h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	2	0008h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	3	000Ch	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	4	0010h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	5	0014h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	6	0018h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	7	001Ch	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	8	0020h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	9	0024h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	10	0028h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	11	002Ch	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	12	0030h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	13	0034h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	14	0038h	—	N/A	N/A	N/A	—	—	—
—	For an unconditional trap	15	003Ch	—	N/A	N/A	N/A	—	—	—
BSC	BUSERR	16	0040h	Level	✓	N/A	N/A	IER02.IEN0	IPR000	—
—	Reserved	17	0044h	—	N/A	N/A	N/A	—	—	—
—	Reserved	18	0048h	—	N/A	N/A	N/A	—	—	—
—	Reserved	19	004Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	20	0050h	—	N/A	N/A	N/A	—	—	—
—	Reserved	21	0054h	—	N/A	N/A	N/A	—	—	—
—	Reserved	22	0058h	—	N/A	N/A	N/A	—	—	—
FCU	FRDYI	23	005Ch	Edge	✓	N/A	N/A	IER02.IEN7	IPR002	—
—	Reserved	24	0060h	—	N/A	N/A	N/A	—	—	—
—	Reserved	25	0064h	—	N/A	N/A	N/A	—	—	—
—	Reserved	26	0068h	—	N/A	N/A	N/A	—	—	—
ICU	SWINT	27	006Ch	Edge	✓	✓	N/A	IER03.IEN3	IPR003	DTCER027
CMT0	CMI0	28	0070h	Edge	✓	✓	N/A	IER03.IEN4	IPR004	DTCER028
CMT1	CMI1	29	0074h	Edge	✓	✓	N/A	IER03.IEN5	IPR005	DTCER029
—	Reserved	30	0078h	—	N/A	N/A	N/A	—	—	—
—	Reserved	31	007Ch	—	N/A	N/A	N/A	—	—	—
CAC	FERRF	32	0080h	Level	✓	N/A	N/A	IER04.IEN0	IPR032	—
	MENDF	33	0084h	Level	✓	N/A	N/A	IER04.IEN1	IPR033	—
	OVFF	34	0088h	Level	✓	N/A	N/A	IER04.IEN2	IPR034	—
—	Reserved	35	008Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	36	0090h	—	N/A	N/A	N/A	—	—	—
—	Reserved	37	0094h	—	N/A	N/A	N/A	—	—	—
—	Reserved	38	0098h	—	N/A	N/A	N/A	—	—	—
—	Reserved	39	009Ch	—	N/A	N/A	N/A	—	—	—
POEG	POEGGAI	40	00A0h	Level	✓	N/A	N/A	IER05.IEN0	IPR040	—
—	Reserved	41	00A4h	—	N/A	N/A	N/A	—	—	—
—	Reserved	42	00A8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	43	00ACh	—	N/A	N/A	N/A	—	—	—
—	Reserved	44	00B0h	—	N/A	N/A	N/A	—	—	—
—	Reserved	45	00B4h	—	N/A	N/A	N/A	—	—	—

Table 14.3 Interrupt Vector Table (2/6)

Source of Interrupt Request Generation	Name	Vector No.*1	Vector Address Offset	Interrupt Detection Method	CPU Interrupt	Start DTC	SSBY Return	IER	IPR	DT CER
—	Reserved	46	00B8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	47	00BCh	—	N/A	N/A	N/A	—	—	—
GPTW0	GTCIA0	48	00C0h	Edge	✓	✓	N/A	IER06.IEN0	IPR048	DT CER048
	GTCIB0	49	00C4h	Edge	✓	✓	N/A	IER06.IEN1	IPR049	DT CER049
	GTCIC0	50	00C8h	Edge	✓	✓	N/A	IER06.IEN2	IPR050	DT CER050
	GTCID0	51	00CCh	Edge	✓	✓	N/A	IER06.IEN3	IPR051	DT CER051
	GDTE0	52	00D0h	Edge	✓	N/A	N/A	IER06.IEN4	IPR052	—
	GTCIE0	53	00D4h	Edge	✓	✓	N/A	IER06.IEN5	IPR053	DT CER053
	GTCIF0	54	00D8h	Edge	✓	✓	N/A	IER06.IEN6	IPR054	DT CER054
	GTCIV0	55	00DCh	Edge	✓	✓	N/A	IER06.IEN7	IPR055	DT CER055
	GTCIU0	56	00E0h	Edge	✓	✓	N/A	IER07.IEN0	IPR056	DT CER056
DOC	DOPCF	57	00E4h	Level	✓	N/A	N/A	IER07.IEN1	IPR057	—
—	Reserved	58	00E8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	59	00ECh	—	N/A	N/A	N/A	—	—	—
—	Reserved	60	00F0h	—	N/A	N/A	N/A	—	—	—
—	Reserved	61	00F4h	—	N/A	N/A	N/A	—	—	—
—	Reserved	62	00F8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	63	00FCh	—	N/A	N/A	N/A	—	—	—
ICU	IRQ0	64	0100h	Edge/Level	✓	✓	✓	IER08.IEN0	IPR064	DT CER064
	IRQ1	65	0104h	Edge/Level	✓	✓	✓	IER08.IEN1	IPR065	DT CER065
	IRQ2	66	0108h	Edge/Level	✓	✓	✓	IER08.IEN2	IPR066	DT CER066
	IRQ3	67	010Ch	Edge/Level	✓	✓	✓	IER08.IEN3	IPR067	DT CER067
	IRQ4	68	0110h	Edge/Level	✓	✓	✓	IER08.IEN4	IPR068	DT CER068
	IRQ5	69	0114h	Edge/Level	✓	✓	✓	IER08.IEN5	IPR069	DT CER069
	IRQ6	70	0118h	Edge/Level	✓	✓	✓	IER08.IEN6	IPR070	DT CER070
	IRQ7	71	011Ch	Edge/Level	✓	✓	✓	IER08.IEN7	IPR071	DT CER071
—	Reserved	72	0120h	—	N/A	N/A	N/A	—	—	—
—	Reserved	73	0124h	—	N/A	N/A	N/A	—	—	—
—	Reserved	74	0128h	—	N/A	N/A	N/A	—	—	—
—	Reserved	75	012Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	76	0130h	—	N/A	N/A	N/A	—	—	—
—	Reserved	77	0134h	—	N/A	N/A	N/A	—	—	—
—	Reserved	78	0138h	—	N/A	N/A	N/A	—	—	—
—	Reserved	79	013Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	80	0140h	—	N/A	N/A	N/A	—	—	—
—	Reserved	81	0144h	—	N/A	N/A	N/A	—	—	—
—	Reserved	82	0148h	—	N/A	N/A	N/A	—	—	—
—	Reserved	83	014Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	84	0150h	—	N/A	N/A	N/A	—	—	—
—	Reserved	85	0154h	—	N/A	N/A	N/A	—	—	—
—	Reserved	86	0158h	—	N/A	N/A	N/A	—	—	—
—	Reserved	87	015Ch	—	N/A	N/A	N/A	—	—	—
LVD	LVD1	88	0160h	Edge	✓	N/A	✓	IER0B.IEN0	IPR088	—
	LVD2	89	0164h	Edge	✓	N/A	✓	IER0B.IEN1	IPR089	—
—	Reserved	90	0168h	—	N/A	N/A	N/A	—	—	—
—	Reserved	91	016Ch	—	N/A	N/A	N/A	—	—	—

Table 14.3 Interrupt Vector Table (3/6)

Source of Interrupt Request Generation	Name	Vector No.*1	Vector Address Offset	Interrupt Detection Method	CPU Interrupt	Start DTC	SSBY Return	IER	IPR	DT CER
—	Reserved	92	0170h	—	N/A	N/A	N/A	—	—	—
—	Reserved	93	0174h	—	N/A	N/A	N/A	—	—	—
—	Reserved	94	0178h	—	N/A	N/A	N/A	—	—	—
—	Reserved	95	017Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	96	0180h	—	N/A	N/A	N/A	—	—	—
—	Reserved	97	0184h	—	N/A	N/A	N/A	—	—	—
GPTW1	GTCIA1	98	0188h	Edge	✓	✓	N/A	IER0C.IEN2	IPR098	DT CER098
	GTCIB1	99	018Ch	Edge	✓	✓	N/A	IER0C.IEN3	IPR099	DT CER099
	GTCIC1	100	0190h	Edge	✓	✓	N/A	IER0C.IEN4	IPR100	DT CER100
	GTCID1	101	0194h	Edge	✓	✓	N/A	IER0C.IEN5	IPR101	DT CER101
S12AD	S12ADI	102	0198h	Edge	✓	✓	N/A	IER0C.IEN6	IPR102	DT CER102
	GBADI	103	019Ch	Edge	✓	✓	N/A	IER0C.IEN7	IPR103	DT CER103
	GCADI	104	01A0h	Edge	✓	✓	N/A	IER0D.IEN0	IPR104	DT CER104
S12AD1	S12ADI1	105	01A4h	Edge	✓	✓	N/A	IER0D.IEN1	IPR105	DT CER105
	GBADI1	106	01A8h	Edge	✓	✓	N/A	IER0D.IEN2	IPR106	DT CER106
	GCADI1	107	01ACh	Edge	✓	✓	N/A	IER0D.IEN3	IPR107	DT CER107
CMPC0	CMPC0	108	01B0h	Edge	✓	✓	N/A	IER0D.IEN4	IPR108	DT CER108
CMPC1	CMPC1	109	01B4h	Edge	✓	✓	N/A	IER0D.IEN5	IPR109	DT CER109
CMPC2	CMPC2	110	01B8h	Edge	✓	✓	N/A	IER0D.IEN6	IPR110	DT CER110
—	Reserved	111	01BCh	—	N/A	N/A	N/A	—	—	—
—	Reserved	112	01C0h	—	N/A	N/A	N/A	—	—	—
—	Reserved	113	01C4h	—	N/A	N/A	N/A	—	—	—
MTU0	TGIA0	114	01C8h	Edge	✓	✓	N/A	IER0E.IEN2	IPR114	DT CER114
	TGIB0	115	01CCh	Edge	✓	✓	N/A	IER0E.IEN3		DT CER115
	TGIC0	116	01D0h	Edge	✓	✓	N/A	IER0E.IEN4		DT CER116
	TGID0	117	01D4h	Edge	✓	✓	N/A	IER0E.IEN5		DT CER117
	TCIV0	118	01D8h	Edge	✓	N/A	N/A	IER0E.IEN6	IPR118	—
	TGIE0	119	01DCh	Edge	✓	N/A	N/A	IER0E.IEN7		—
	TGIF0	120	01E0h	Edge	✓	N/A	N/A	IER0F.IEN0		—
MTU1	TGIA1	121	01E4h	Edge	✓	✓	N/A	IER0F.IEN1	IPR121	DT CER121
	TGIB1	122	01E8h	Edge	✓	✓	N/A	IER0F.IEN2		DT CER122
	TCIV1	123	01ECh	Edge	✓	N/A	N/A	IER0F.IEN3	IPR123	—
	TCIU1	124	01F0h	Edge	✓	N/A	N/A	IER0F.IEN4		—
MTU2	TGIA2	125	01F4h	Edge	✓	✓	N/A	IER0F.IEN5	IPR125	DT CER125
	TGIB2	126	01F8h	Edge	✓	✓	N/A	IER0F.IEN6		DT CER126
	TCIV2	127	01FCh	Edge	✓	N/A	N/A	IER0F.IEN7	IPR127	—
	TCIU2	128	0200h	Edge	✓	N/A	N/A	IER10.IEN0		—
MTU3	TGIA3	129	0204h	Edge	✓	✓	N/A	IER10.IEN1	IPR129	DT CER129
	TGIB3	130	0208h	Edge	✓	✓	N/A	IER10.IEN2		DT CER130
	TGIC3	131	020Ch	Edge	✓	✓	N/A	IER10.IEN3		DT CER131
	TGID3	132	0210h	Edge	✓	✓	N/A	IER10.IEN4		DT CER132
	TCIV3	133	0214h	Edge	✓	N/A	N/A	IER10.IEN5	IPR133	—

Table 14.3 Interrupt Vector Table (4/6)

Source of Interrupt Request Generation	Name	Vector No.*1	Vector Address Offset	Interrupt Detection Method	CPU Interrupt	Start DTC	SSBY Return	IER	IPR	DT CER
MTU4	TGIA4	134	0218h	Edge	✓	✓	N/A	IER10.IEN6	IPR134	DT CER134
	TGIB4	135	021Ch	Edge	✓	✓	N/A	IER10.IEN7		DT CER135
	TGIC4	136	0220h	Edge	✓	✓	N/A	IER11.IEN0		DT CER136
	TGID4	137	0224h	Edge	✓	✓	N/A	IER11.IEN1		DT CER137
	TCIV4	138	0228h	Edge	✓	✓	N/A	IER11.IEN2	IPR138	DT CER138
MTU5	TGIU5	139	022Ch	Edge	✓	✓	N/A	IER11.IEN3	IPR139	DT CER139
	TGIV5	140	0230h	Edge	✓	✓	N/A	IER11.IEN4		DT CER140
	TGIW5	141	0234h	Edge	✓	✓	N/A	IER11.IEN5		DT CER141
—	Reserved	142	0238h	—	N/A	N/A	N/A	—	—	—
—	Reserved	143	023Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	144	0240h	—	N/A	N/A	N/A	—	—	—
—	Reserved	145	0244h	—	N/A	N/A	N/A	—	—	—
—	Reserved	146	0248h	—	N/A	N/A	N/A	—	—	—
—	Reserved	147	024Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	148	0250h	—	N/A	N/A	N/A	—	—	—
—	Reserved	149	0254h	—	N/A	N/A	N/A	—	—	—
—	Reserved	150	0258h	—	N/A	N/A	N/A	—	—	—
—	Reserved	151	025Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	152	0260h	—	N/A	N/A	N/A	—	—	—
—	Reserved	153	0264h	—	N/A	N/A	N/A	—	—	—
—	Reserved	154	0268h	—	N/A	N/A	N/A	—	—	—
—	Reserved	155	026Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	156	0270h	—	N/A	N/A	N/A	—	—	—
—	Reserved	157	0274h	—	N/A	N/A	N/A	—	—	—
—	Reserved	158	0278h	—	N/A	N/A	N/A	—	—	—
—	Reserved	159	027Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	160	0280h	—	N/A	N/A	N/A	—	—	—
—	Reserved	161	0284h	—	N/A	N/A	N/A	—	—	—
—	Reserved	162	0288h	—	N/A	N/A	N/A	—	—	—
—	Reserved	163	028Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	164	0290h	—	N/A	N/A	N/A	—	—	—
—	Reserved	165	0294h	—	N/A	N/A	N/A	—	—	—
—	Reserved	166	0298h	—	N/A	N/A	N/A	—	—	—
—	Reserved	167	029Ch	—	N/A	N/A	N/A	—	—	—
POE	OEI1	168	02A0h	Level	✓	N/A	N/A	IER15.IEN0	IPR168	—
—	Reserved	169	02A4h	—	N/A	N/A	N/A	—		—
POE	OEI3	170	02A8h	Level	✓	N/A	N/A	IER15.IEN2	IPR168	—
	OEI4	171	02ACh	Level	✓	N/A	N/A	IER15.IEN3		—
	OEI5	172	02B0h	Level	✓	N/A	N/A	IER15.IEN4		—
—	Reserved	173	02B4h	—	N/A	N/A	N/A	—	—	—
TMR0	CMIA0	174	02B8h	Edge	✓	✓	N/A	IER15.IEN6	IPR174	DT CER174
	CMIB0	175	02BCh	Edge	✓	✓	N/A	IER15.IEN7		DT CER175
	OVI0	176	02C0h	Edge	✓	N/A	N/A	IER16.IEN0		—
TMR1	CMIA1	177	02C4h	Edge	✓	✓	N/A	IER16.IEN1	IPR177	DT CER177
	CMIB1	178	02C8h	Edge	✓	✓	N/A	IER16.IEN2		DT CER178
	OVI1	179	02CCh	Edge	✓	N/A	N/A	IER16.IEN3		—

Table 14.3 Interrupt Vector Table (5/6)

Source of Interrupt Request Generation	Name	Vector No.*1	Vector Address Offset	Interrupt Detection Method	CPU Interrupt	Start DTC	SSBY Return	IER	IPR	DT CER
TMR2	CMIA2	180	02D0h	Edge	✓	✓	N/A	IER16.IEN4	IPR180	DT CER180
	CMIB2	181	02D4h	Edge	✓	✓	N/A	IER16.IEN5		DT CER181
	OVI2	182	02D8h	Edge	✓	N/A	N/A	IER16.IEN6		—
TMR3	CMIA3	183	02DCh	Edge	✓	✓	N/A	IER16.IEN7	IPR183	DT CER183
	CMIB3	184	02E0h	Edge	✓	✓	N/A	IER17.IEN0		DT CER184
	OVI3	185	02E4h	Edge	✓	N/A	N/A	IER17.IEN1		—
—	Reserved	186	02E8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	187	02ECh	—	N/A	N/A	N/A	—	—	—
—	Reserved	188	02F0h	—	N/A	N/A	N/A	—	—	—
—	Reserved	189	02F4h	—	N/A	N/A	N/A	—	—	—
—	Reserved	190	02F8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	191	02FCh	—	N/A	N/A	N/A	—	—	—
—	Reserved	192	0300h	—	N/A	N/A	N/A	—	—	—
—	Reserved	193	0304h	—	N/A	N/A	N/A	—	—	—
—	Reserved	194	0308h	—	N/A	N/A	N/A	—	—	—
—	Reserved	195	030Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	196	0310h	—	N/A	N/A	N/A	—	—	—
—	Reserved	197	0314h	—	N/A	N/A	N/A	—	—	—
—	Reserved	198	0318h	—	N/A	N/A	N/A	—	—	—
—	Reserved	199	031Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	200	0320h	—	N/A	N/A	N/A	—	—	—
—	Reserved	201	0324h	—	N/A	N/A	N/A	—	—	—
GPTW1	GDTE1	202	0328h	Edge	✓	N/A	N/A	IER19.IEN2	IPR202	—
	GTCIE1	203	032Ch	Edge	✓	✓	N/A	IER19.IEN3	IPR203	DT CER203
	GTCIF1	204	0330h	Edge	✓	✓	N/A	IER19.IEN4	IPR204	DT CER204
	GTCIV1	205	0334h	Edge	✓	✓	N/A	IER19.IEN5	IPR205	DT CER205
	GTCIU1	206	0338h	Edge	✓	✓	N/A	IER19.IEN6	IPR206	DT CER206
GPTW2	GTCIA2	207	033Ch	Edge	✓	✓	N/A	IER19.IEN7	IPR207	DT CER207
	GTCIB2	208	0340h	Edge	✓	✓	N/A	IER1A.IEN0	IPR208	DT CER208
	GTCIC2	209	0344h	Edge	✓	✓	N/A	IER1A.IEN1	IPR209	DT CER209
	GTCID2	210	0348h	Edge	✓	✓	N/A	IER1A.IEN2	IPR210	DT CER210
	GDTE2	211	034Ch	Edge	✓	N/A	N/A	IER1A.IEN3	IPR211	—
	GTCIE2	212	0350h	Edge	✓	✓	N/A	IER1A.IEN4	IPR212	DT CER212
	GTCIF2	213	0354h	Edge	✓	✓	N/A	IER1A.IEN5	IPR213	DT CER213
	GTCIV2	214	0358h	Edge	✓	✓	N/A	IER1A.IEN6	IPR214	DT CER214
	GTCIU2	215	035Ch	Edge	✓	✓	N/A	IER1A.IEN7	IPR215	DT CER215
—	Reserved	216	0360h	—	N/A	N/A	N/A	—	—	—
—	Reserved	217	0364h	—	N/A	N/A	N/A	—	—	—
SCI1	ER11	218	0368h	Level	✓	N/A	N/A	IER1B.IEN2	IPR218	—
	RXI1	219	036Ch	Edge	✓	✓	N/A	IER1B.IEN3		DT CER219
	TXI1	220	0370h	Edge	✓	✓	N/A	IER1B.IEN4		DT CER220
	TEI1	221	0374h	Level	✓	N/A	N/A	IER1B.IEN5		—
SCI5	ER15	222	0378h	Level	✓	N/A	N/A	IER1B.IEN6	IPR222	—
	RXI5	223	037Ch	Edge	✓	✓	N/A	IER1B.IEN7		DT CER223
	TXI5	224	0380h	Edge	✓	✓	N/A	IER1C.IEN0		DT CER224
	TEI5	225	0384h	Level	✓	N/A	N/A	IER1C.IEN1		—

Table 14.3 Interrupt Vector Table (6/6)

Source of Interrupt Request Generation	Name	Vector No.*1	Vector Address Offset	Interrupt Detection Method	CPU Interrupt	Start DTC	SSBY Return	IER	IPR	DT CER
SCI6	ERI6	226	0388h	Level	✓	N/A	N/A	IER1C.IEN2	IPR226	—
	RXI6	227	038Ch	Edge	✓	✓	N/A	IER1C.IEN3		DT CER227
	TXI6	228	0390h	Edge	✓	✓	N/A	IER1C.IEN4		DT CER228
	TEI6	229	0394h	Level	✓	N/A	N/A	IER1C.IEN5		—
—	Reserved	230	0398h	—	N/A	N/A	N/A	—	—	—
—	Reserved	231	039Ch	—	N/A	N/A	N/A	—	—	—
—	Reserved	232	03A0h	—	N/A	N/A	N/A	—	—	—
—	Reserved	233	03A4h	—	N/A	N/A	N/A	—	—	—
—	Reserved	234	03A8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	235	03ACh	—	N/A	N/A	N/A	—	—	—
—	Reserved	236	03B0h	—	N/A	N/A	N/A	—	—	—
—	Reserved	237	03B4h	—	N/A	N/A	N/A	—	—	—
SCI12	ERI12	238	03B8h	Level	✓	N/A	N/A	IER1D.IEN6	IPR238	—
	RXI12	239	03BCh	Edge	✓	✓	N/A	IER1D.IEN7		DT CER239
	TXI12	240	03C0h	Edge	✓	✓	N/A	IER1E.IEN0		DT CER240
	TEI12	241	03C4h	Level	✓	N/A	N/A	IER1E.IEN1		—
	SCIX0	242	03C8h	Level	✓	N/A	N/A	IER1E.IEN2	IPR242	—
	SCIX1	243	03CCh	Level	✓	N/A	N/A	IER1E.IEN3	IPR243	—
	SCIX2	244	03D0h	Level	✓	N/A	N/A	IER1E.IEN4	IPR244	—
	SCIX3	245	03D4h	Level	✓	N/A	N/A	IER1E.IEN5	IPR245	—
RIIC0	EEI0	246	03D8h	Level	✓	N/A	N/A	IER1E.IEN6	IPR246	—
	RXI0	247	03DCh	Edge	✓	✓	N/A	IER1E.IEN7	IPR247	DT CER247
	TXI0	248	03E0h	Edge	✓	✓	N/A	IER1F.IEN0	IPR248	DT CER248
	TEI0	249	03E4h	Level	✓	N/A	N/A	IER1F.IEN1	IPR249	—
—	Reserved	250	03E8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	251	03ECh	—	N/A	N/A	N/A	—	—	—
—	Reserved	252	03F0h	—	N/A	N/A	N/A	—	—	—
—	Reserved	253	03F4h	—	N/A	N/A	N/A	—	—	—
—	Reserved	254	03F8h	—	N/A	N/A	N/A	—	—	—
—	Reserved	255	03FCh	—	N/A	N/A	N/A	—	—	—

Note 1. An interrupt source with a smaller vector number takes precedence.

14.3.2 Fast Interrupt Vector Table

The address of the entry in the interrupt vector table that corresponds to the vector number of the fast interrupt is placed in the fast interrupt vector register (FINTV) of the CPU.

14.3.3 Non-maskable Interrupt Vector Area

Non-maskable interrupts use the vector area in the exception vector table.

The exception vector table is allocated to the 128-byte area (4 bytes × 32 sources) beginning with the address set in the EXTB register in the CPU. Set a multiple of 4 in the EXTB register.

14.4 Interrupt Operation

The interrupt controller performs the following processing.

- Detecting interrupts
- Enabling and disabling interrupts
- Selecting interrupt request destinations (CPU interrupt or DTC trigger)
- Determining priority

14.4.1 Detecting Interrupts

Interrupt requests are detected in either of two ways: the detection of edges of the interrupt signal or the detection of a level of the interrupt signal.

Edge detection or level detection is selected for the IRQ_i pins ($i = 0$ to 7) as external interrupt requests by the setting of the IRQMD[1:0] bits in IRQCR_i.

For interrupts from peripheral modules, either edge detection or level detection is determined per interrupt source.

For the correspondence between interrupt sources and methods of detection, see Table 14.3, Interrupt Vector Table.

14.4.1.1 Operation of Status Flags for Edge-Detected Interrupts

Figure 14.2 shows the operation of the IR flag in IR_n ($n =$ interrupt vector number) in the case of edge detection of an interrupt from a peripheral module or on an external pin.

The IR flag in IR_n is set to 1 immediately after the transition of the interrupt signal due to generation of the interrupt. If the CPU is the request destination for the interrupt, the IR flag is automatically cleared to 0 on acceptance of the interrupt. If the DTC is the request destination for the interrupt, the IR_n.IR flag operation differs according to the DTC transfer settings and transfer count. For details, see Table 14.4, Operation When Starting the DTC.

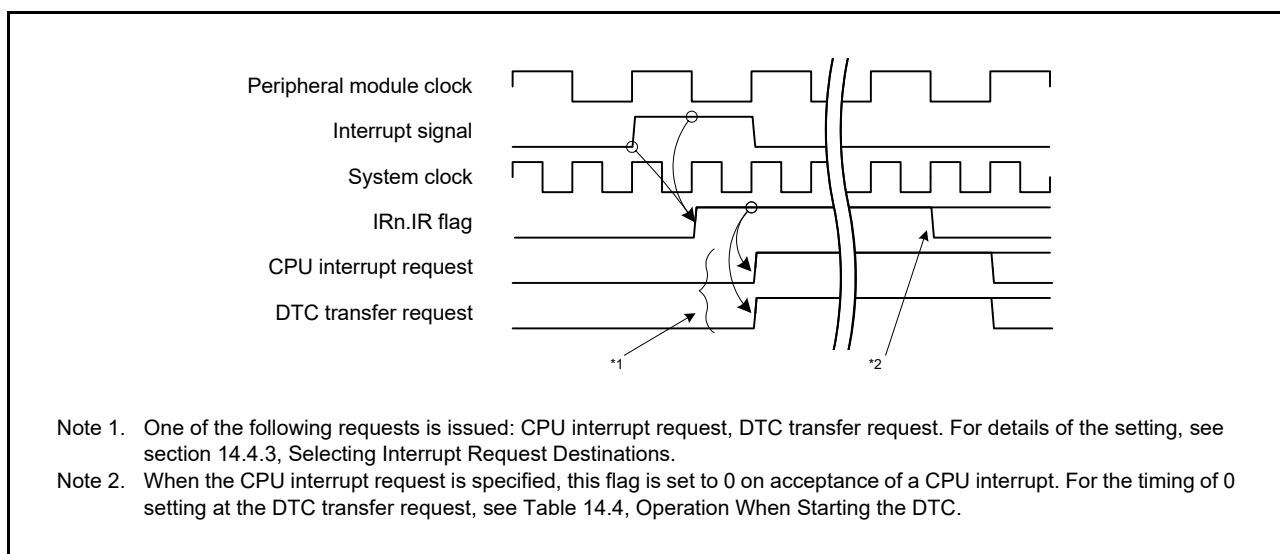


Figure 14.2 IR_n.IR Flag Operation for Edge Detection Interrupts

Figure 14.3 to Figure 14.5 show the interrupt signals of the interrupt controller. Note that the timings of the interrupts with interrupt vector numbers 64 to 95 are different from those of other interrupts. For the IRQ pin interrupts with interrupt vector numbers 64 to 79, “internal delay + 2 PCLK cycles” of delay is added after the IRQ pin input. For the interrupts with interrupt vector numbers 80 to 95, “2 PCLK cycles” of delay is added.

If an interrupt signal is generated every clock cycle, the subsequent interrupts cannot be detected; secure two or more clock cycles of the system clock between issuance of continuous interrupt requests.

While the IRn.IR flag is 1 after an interrupt request is generated, the interrupt request that is generated again will be ignored.*1

Figure 14.3 shows the timing for IRn.IR flag re-setting.

Note 1. When the transmission or reception interrupt of the SCI or RIIC is generated with the IRn.IR flag being 1, the interrupt request is retained. After the IRn.IR flag is cleared to 0, the IRn.IR flag is set to 1 again by the retained request. For details, see descriptions of the interrupts in section 26, Serial Communications Interface (SCIg, SCIfh) and section 27, I²C-bus Interface (RIICa).

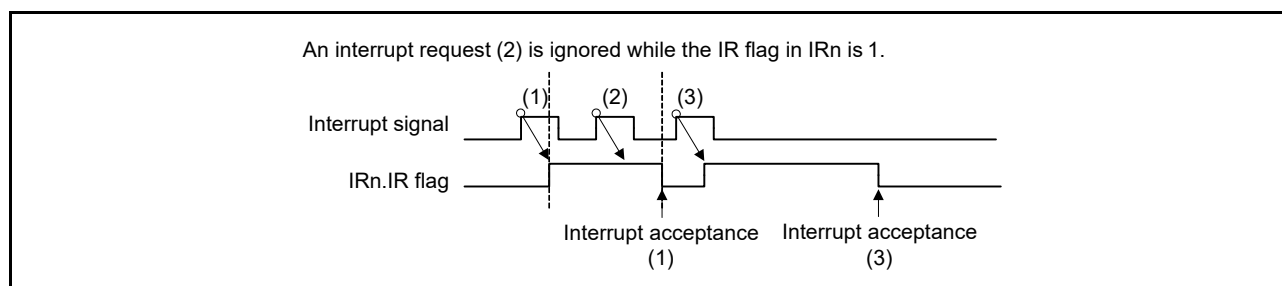


Figure 14.3 Timing for IRn.IR Flag Re-Setting

If an interrupt is disabled after the IRn.IR flag is set to 1 (output of the interrupt request is disabled by the interrupt enable bit of the relevant peripheral module), the IRn.IR flag is not affected but retains its state. Figure 14.4 shows operation when the interrupt is disabled.

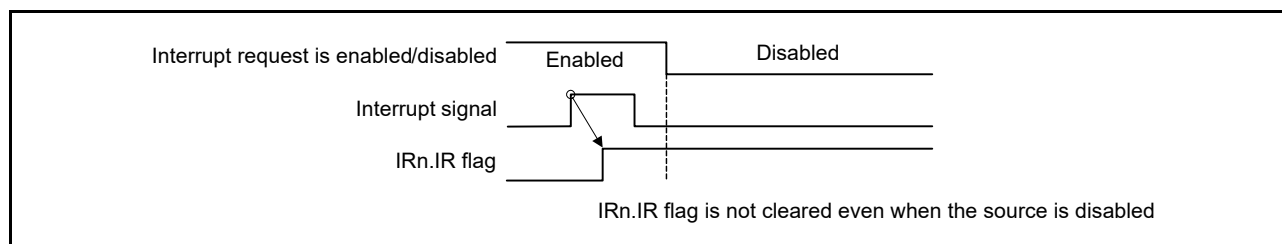


Figure 14.4 Relationship between IRn.IR Flag Operation and Disabling of Interrupt Request

14.4.1.2 Operation of Status Flags for Level-Detected Interrupts

Figure 14.5 shows the operation of the interrupt status flag (IR flag) in IR_n (n = interrupt vector number) in the case of level detection of an interrupt from a peripheral module or an external pin.

The IR flag in IR_n remains set to 1 as long as the interrupt signal is asserted. To clear the IR_n .IR flag to 0, clear the interrupt request in the source generating the interrupt. Confirm that the interrupt request flag in the source generating the interrupt has been cleared to 0 and that the IR_n .IR flag has been cleared to 0, and then complete the interrupt handling.

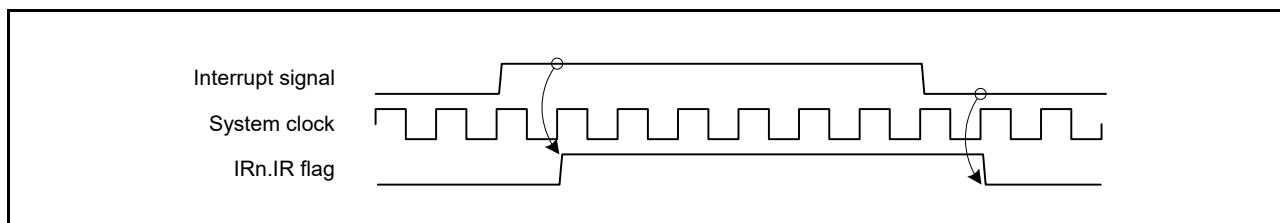


Figure 14.5 IR_n .IR Flag Operation for Level Detection Interrupts

Figure 14.6 shows the procedure for handling level detection interrupts.

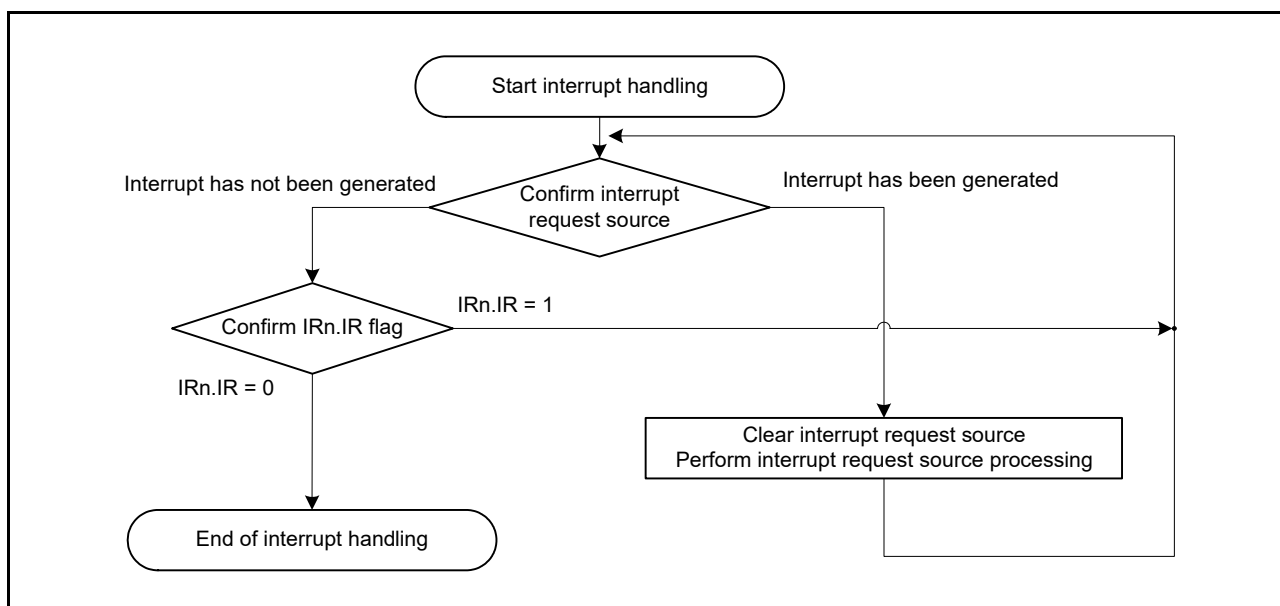


Figure 14.6 Procedure for Handling Level Detection Interrupts

14.4.2 Enabling and Disabling Interrupt Sources

Enabling requests from a given interrupt source requires the following settings.

1. In the case of interrupt requests from peripheral modules, setting the interrupt enable bit for the peripheral module to permit the output of interrupt requests from the source
2. Enabling of the interrupt by the IERm.IENj bit (m = 02h to 1Fh; j = 0 to 7)

When an interrupt request that is enabled at the corresponding source is generated, the corresponding IRn.IR flag (n = interrupt vector number) is set to 1.

Setting the IERm.IENj bit to enable an interrupt request allows the interrupt request for which the corresponding IRn.IR is 1 to be output to the interrupt request destination. Setting the IERm.IENj bit to disable an interrupt request suspends the output of the interrupt request for which the corresponding IRn.IR is 1.

The IRn.IR flag is not affected by the IERm.IENj bit.

Use the following procedure to disable interrupt requests.

1. Set the IERm.IENj bit to disable interrupt requests.
2. Set the peripheral module interrupt output enable bit to disable the output. Read the last written register and confirm that writing is completed.
3. Check the IRn.IR flag, and clear the IRn.IR flag if necessary.*1

Note 1. To disable the transmission or reception interrupt of the SCI or RIIC from the enabled state, clear the IRn.IR flag to 0 using the above procedure. For details, see descriptions of the interrupts in section 26, Serial Communications Interface (SCIg, SCIH) and section 27, I²C-bus Interface (RIICa).

14.4.3 Selecting Interrupt Request Destinations

Possible settings for the request destination of each interrupt are fixed. That is, settings for request destination other than those indicated in Table 14.3, Interrupt Vector Table, are not possible. Do not make an interrupt request destination setting that is not indicated by a “✓” in Table 14.3.

If the DTC is selected as the destination for requests from an IRQi pin (i = 0 to 7), be sure to set the IRQMD[1:0] bits in IRQCRi for that interrupt to select edge detection.

The following describes how to specify the destinations of interrupt requests.

(1) DTC Trigger

Make the following settings for each source while the IERm.IENj bit (m = 02h to 1Fh; j = 0 to 7) is 0.

- Set the DTC transfer request enable bit in the DTC transfer request enable register (DTCERn.DTCE (n = interrupt vector number)) for the pertinent source to 1.

After making the above settings, set the IERm.IENj bit to 1.

In addition, set the DTC module start bit (DTCST.DTCST) to 1. The order of making settings for each interrupt and enabling the DTC module start bit does not matter.

For the DTC setting procedure, refer to section 16.5, DTC Setting Procedure, in section 16, Data Transfer Controller (DTCb).

(2) CPU Interrupt Request

If the interrupt request destination is the DTC, the interrupt request is sent to the CPU. Set the IERm.IENj bit (m = 02h to 1Fh; j = 0 to 7) to 1 while the DTC trigger settings described above are in place.

Table 14.4 shows operation when the DTC is the request destination.

Table 14.4 Operation When Starting the DTC

Interrupt Request Destination	DISEL *1	Remaining Number of Transfer Operations	Operation per Request	IR*2	Interrupt Request Destination after Transfer
DTC*3	1	≠ 0	DTC transfer → CPU interrupt	Cleared on interrupt acceptance by the CPU	DTC
		= 0	DTC transfer → CPU interrupt	Cleared on interrupt acceptance by the CPU	The DTCERn.DTCE bit is cleared and the CPU becomes the destination.
	0	≠ 0	DTC transfer	Cleared at the start of DTC data transfer after reading DTC transfer information	DTC
		= 0	DTC transfer → CPU interrupt	Cleared on interrupt acceptance by the CPU	The DTCERn.DTCE bit is cleared and the CPU becomes the destination.

Note 1. DISEL for the DTC is set by the DTC.MRB.DISEL bit.

Note 2. When the IRn.IR flag is 1, an interrupt request (DTC transfer request) that is generated again will be ignored.

Note 3. For chain transfer, DTC transfer continues until the last chain transfer ends. Whether a CPU interrupt is generated at the end of chain transfer, the IRn.IR flag clear timing, and the interrupt request destination after transfer are determined by the state of DISEL and the remaining transfer count at the end of chain transfer. For the chain transfer, see Table 16.4, Chain Transfer Conditions in section 16, Data Transfer Controller (DTCb).

The request destination for an interrupt should be changed while the IERm.IENj bit is 0.

When a source is to be changed to an interrupt request or the DTC transfer information is to be changed while a transfer is not complete (i.e. while the DTCERn.DTCE bit (n = interrupt vector number) has not been cleared) after the settings described under (1) DTC Trigger have been made, follow the procedure below.

- For both the source to be withdrawn and the source that will have a new trigger, clear the IENj bits in IERm to 0.
- Check the state of transfer by the DTC. If transfer is in progress, wait for its completion.
- Make the settings described under (1) DTC Trigger.

14.4.4 Determining Priority

Interrupt priority is determined for each interrupt request destination.

The priority for each interrupt request destination is determined as follows.

(1) Determining Priority when the CPU is the Request Destination of the Interrupt

A source selected for the fast interrupt has the highest priority. After that, an interrupt source with a larger value of the interrupt priority level select bits (IPR[3:0]) in IPR_n takes priority. If interrupts with the same priority level are generated by multiple sources, the source with the smallest vector number takes precedence.

(2) Determining Priority when the DTC is the Request Destination of the Interrupt

The IPR[3:0] bits in IPR_n (n = interrupt vector number) have no effect. An interrupt source with a smaller vector number takes precedence.

14.4.5 Multiple Interrupts

To enable multiple interrupts of the CPU, set the PSW.I bit to 1 (interrupt enabled) in the handling routine of accepted interrupts.

The PSW.IPL[3:0] bits immediately after processing branches to the interrupt handling routine are set to the same value as the interrupt priority level of the accepted interrupt request. If an interrupt request which has an interrupt level higher than that of the PSW.IPL[3:0] bits is generated at this time, this interrupt request (for multiple interrupts) is accepted.

If the interrupt priority level of the accepted interrupt request is 15 (fast interrupt or interrupt when IPR[3:0] are set to 1111b), multiple interrupts are not generated.

14.4.6 Fast Interrupt

The fast interrupt is an interrupt for executing a faster interrupt response by the CPU, so only one of the interrupt sources can be assigned.

The interrupt priority level of the fast interrupt is 15 (highest) regardless of the setting of the IPR[3:0] bits in IPR_n (n = interrupt vector number). In addition, the fast interrupt is accepted with precedence over other interrupt sources with level 15. However, when the value of the PSW.IPL[3:0] bits are 1111b (priority level 15), even the fast interrupt cannot be accepted.

To assign an interrupt source to the fast interrupt, specify the vector number of the source in the FIR.FVCT[7:0] bits, and set the FIR.FIEN bit to 1 (fast interrupt is enabled).

For details on the fast interrupt, see section 2, CPU and section 13, Exception Handling.

14.4.7 Digital Filter

The digital filter function is provided for the external interrupt request IRQ_i pins ($i = 0$ to 7) and NMI pin interrupt. The digital filter samples input signals at the filter sampling clock (PCLK) and removes the pulses of which length is less than three sampling cycles.

To use the digital filter for the IRQ_i pin, set the sampling clock cycle (PCLK, PCLK/8, PCLK/32, or PCLK/64) with the IRQFLTC0.FCLKSEL_i[1:0] bits and set the IRQFLTE0.FLTEN_i bit to 1 (digital filter enabled).

To use the digital filter for the NMI pin, set the sampling clock cycle (PCLK, PCLK/8, PCLK/32, or PCLK/64) with the NMIFLTC.NFCLKSEL[1:0] bits and set the NMIFLTE.NFLTEN bit to 1 (digital filter enabled).

Figure 14.7 shows an example of digital filter operation.

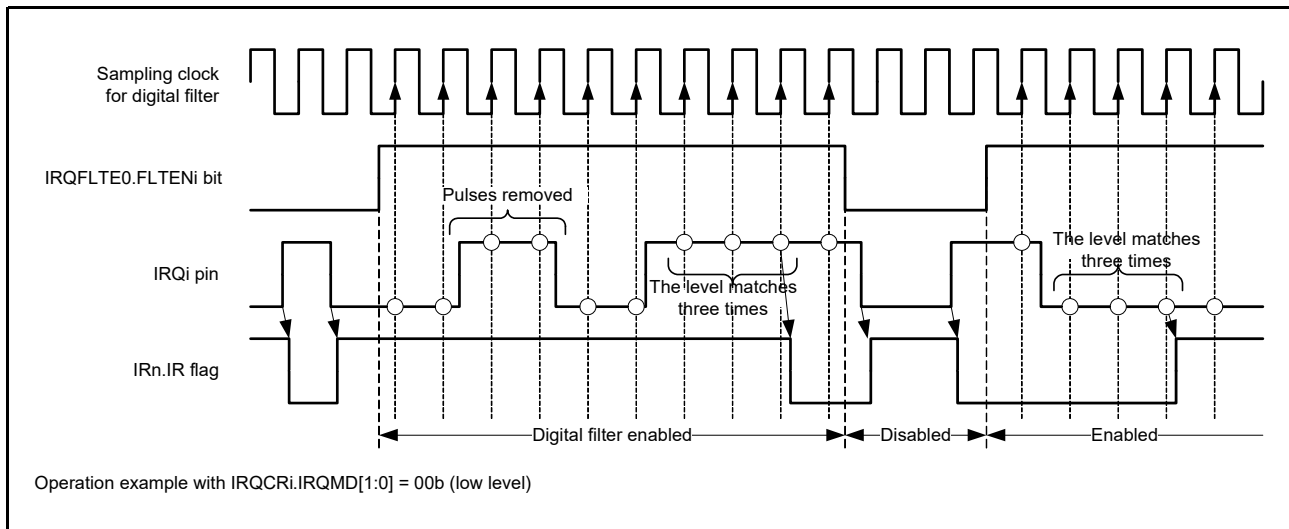


Figure 14.7 Digital Filter Operation Example

Before software standby mode is entered, set the IRQFLTE0.FLTEN_i and NMIFLTE.NFLTEN bits to 0 (digital filter disabled). To use the digital filter again after return from software standby mode, set the IRQFLTE0.FLTEN_i or NMIFLTE.NFLTEN bit to 1 (digital filter enabled).

14.4.8 External Pin Interrupts

The procedure for using the signal on an external pin as an interrupt is as follows.

1. Clear the IER_m.IEN_j bit ($m = 02h$ to $1Fh$; $j = 0$ to 7) to 0 (interrupt request disabled).
2. Clear the IRQFLTE0.FLTEN_i bit ($i = 0$ to 7) to 0 (digital filter disabled).^{*1}
3. Set the digital filter sampling clock with the IRQFLTC0.FCLKSEL_i[1:0] bits.^{*1}
4. Make or confirm the I/O port settings.
5. Set the method of detection for the interrupt in the IRQCR_i.IRQMD[1:0] bits.
6. Clear the corresponding IR_n.IR flag ($n =$ interrupt vector number) to 0 (if edge detection is in use).
7. Set the IRQFLTE0.FLTEN_i bit to 1 (digital filter enabled).^{*1}
8. If the interrupt is to be used for DTC trigger, set the DTCER_n.DTCE bit. The interrupt will be a CPU interrupt if the setting not is made.
9. Set the IER_m.IEN_j bit to 1 (interrupt request enabled).

Note 1. To use the digital filter function, settings must be made beforehand.

14.5 Non-maskable Interrupt Operation

There are six types of non-maskable interrupt: the NMI pin interrupt, oscillation stop detection interrupt, IWDTC underflow/refresh error, voltage monitoring 1 interrupt, and voltage monitoring 2 interrupt. Non-maskable interrupts are only usable as interrupts for the CPU; that is, they are not capable of DTC trigger. Non-maskable interrupts take precedence over all interrupts, including the fast interrupt.

Non-maskable interrupt requests are accepted regardless of the states of the I (interrupt enable) bit and IPL[3:0] (processor interrupt priority level) bits in the PSW of the CPU. The current states of the non-maskable interrupts can be checked in the non-maskable interrupt status register (NMISR).

Confirm that all bits in the NMISR have returned to 0 from within the handler for the non-maskable interrupt, before ending the handler.

Non-maskable interrupts are disabled by default. If a system is to use non-maskable interrupts, the following procedure must be followed at the beginning of program processing.

Non-maskable interrupt usage procedure:

1. Set the stack pointer (SP).
2. To use the NMI pin, clear the NMIFLTE.NFLTEN bit to 0 (digital filter disabled).^{*1}
3. To use the NMI pin, set the digital filter sampling clock with the NMIFLTC.NFCLKSEL[1:0] bits.^{*1}
4. To use the NMI pin, set the NMI pin detection sense with the NMICR.NMIMD bit.
5. To use the NMI pin, write 1 to the NMICLR.NMICLR bit to clear the NMISR.NMIST flag to 0.
6. To use the NMI pin, set the NMIFLTE.NFLTEN bit to 1 (digital filter enabled).^{*1}
7. Enable the non-maskable interrupt by writing 1 to the corresponding bit in the non-maskable interrupt enable register (NMIER).

Note 1. To use the digital filter function, settings must be made beforehand.

After 1 is written to the NMIER register, subsequent write access to the NMIE bit in NMIER is ignored. The NMI interrupt cannot be disabled. It can be disabled only by a reset.

For the flow of non-maskable interrupt processing, see section 13, Exception Handling.

Writing 1 to the NMICLR.NMICLR bit sets the NMI status flag (NMISR.NMIST) to 0.

Writing 1 to the NMICLR.OSTCLR bit sets the oscillation stop detection interrupt status flag (NMISR.OSTST) to 0.

Writing 1 to the NMICLR.IWDTCCLR bit sets the IWDTC underflow/refresh error status flag (NMISR.IWDTCST) to 0.

Writing 1 to the NMICLR.LVD1CLR bit sets the voltage monitoring 1 interrupt status flag (NMISR.LVD1ST) to 0.

Writing 1 to the NMICLR.LVD2CLR bit sets the voltage monitoring 2 interrupt status flag (NMISR.LVD2ST) to 0.

14.6 Return from Low Power Consumption States

Interrupts can be used to return from sleep mode, deep sleep mode, or software standby mode.

Refer to section 11, Low Power Consumption for details. This section describes the procedure to set an interrupt source for returning from each low power consumption mode.

14.6.1 Return from Sleep Mode or Deep Sleep Mode

All non-maskable interrupts and all interrupts can be used to return from sleep mode and deep sleep mode. The following conditions must be satisfied.

(1) Non-maskable interrupts

- Generation of the interrupt that is used for return is enabled in the NMIER register.

(2) Interrupts

- The CPU is selected as the interrupt request destination.
- The interrupt request is enabled by the IERm.IENj bit (m = 02h to 1Fh; j = 0 to 7).
- The priority level is higher than the PSW.IPL[3:0] bit value in the CPU.

14.6.2 Return from Software Standby Mode

Non-maskable interrupts and interrupts that have a “✓” in the SSBY return column in Table 14.3, Interrupt Vector Table can be used to return from software standby mode. The conditions below must be satisfied.

(1) Non-maskable interrupts

- Generation of the interrupt that is used for return is enabled in the NMIER register.
- When using the NMI pin interrupt, the digital filter is disabled.

(2) Interrupts

- The interrupt source can be used to return from software standby mode.
- The CPU is selected as the interrupt request destination.
- The interrupt request is enabled by the IERm.IENj bit (m = 02h to 1Fh; j = 0 to 7).
- The priority level is higher than the PSW.IPL[3:0] bit value in the CPU.
When using the fast interrupt, set not only the FIR register but also the corresponding IPRn.IPR[3:0] bits (n = interrupt vector number). Set the IPRn.IPR[3:0] bits to a higher level than the PSW.IPL[3:0] bit value in the CPU.
- When using the external pin interrupt, the digital filter of the IRQi pin used is disabled.

Refer to section 14.4.7, Digital Filter for details on the procedure to set the digital filter.

14.7 Usage Note

14.7.1 Note on WAIT Instruction Used with Non-Maskable Interrupt

Before executing the WAIT instruction, check to see that all the status flags in NMISR are 0.

15. Buses

15.1 Overview

Table 15.1 lists the bus specifications, Figure 15.1 shows the bus configuration, and Table 15.2 lists the addresses assigned to each bus.

Table 15.1 Bus Specifications

Bus Type		Description
CPU bus	Instruction bus	- Connected to the CPU for instructions - Connected to on-chip memory (RAM, ROM) - Operates in synchronization with the system clock (ICLK)
	Operand bus	- Connected to the CPU (for operands) - Connected to on-chip memory (RAM, ROM) - Operates in synchronization with the system clock (ICLK)
Memory bus	Memory bus 1	Connected to RAM
	Memory bus 2	Connected to ROM
Internal main buses	Internal main bus 1	- Connected to the CPU - Operates in synchronization with the system clock (ICLK)
	Internal main bus 2	- Connected to the DTC - Connected to on-chip memory (RAM, ROM) - Operates in synchronization with the system clock (ICLK)
Internal peripheral buses	Internal peripheral bus 1	- Connected to peripheral modules (DTC, interrupt controller, and bus error monitoring section) - Operates in synchronization with the system clock (ICLK)
	Internal peripheral bus 2	- Connected to peripheral modules - Operates in synchronization with the peripheral module clock (PCLKB, PCLKD)
	Internal peripheral bus 3	- Connected to peripheral modules (CMPC) - Operates in synchronization with the peripheral module clock (PCLKB)
	Internal peripheral bus 6	- Connected to ROM (P/E) and E2 DataFlash - Operates in synchronization with the FlashIF clock (FCLK)

P/E: Programming/Erase

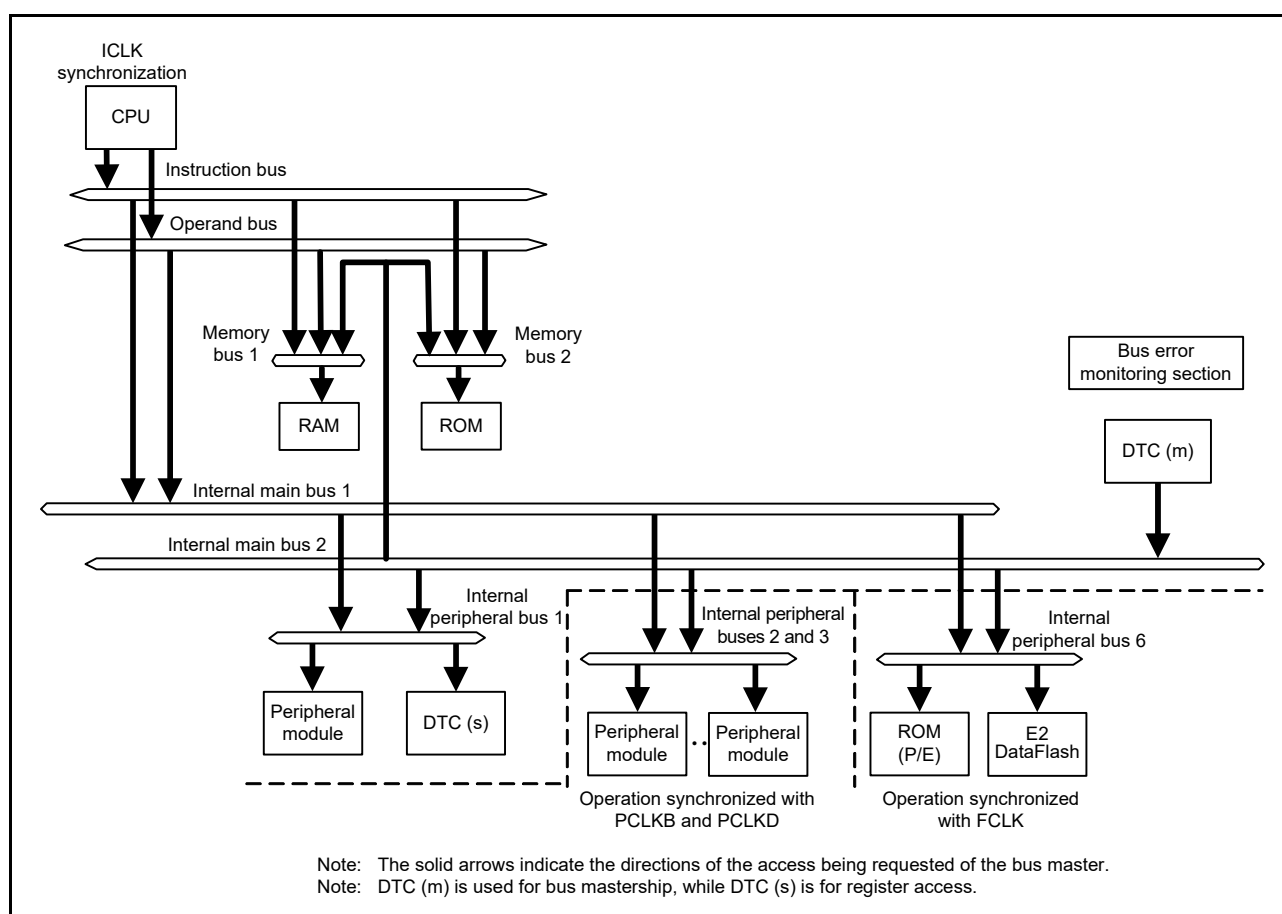


Figure 15.1 Bus Configuration

Table 15.2 Addresses Assigned for Each Bus

Address	Bus	Area
0000 0000h to 0000 FFFFh	Memory bus 1	RAM
0008 0000h to 0008 7FFFh	Internal peripheral bus 1	Peripheral I/O registers
0008 8000h to 0009 FFFFh	Internal peripheral bus 2	
000A 0000h to 000B FFFFh	Internal peripheral bus 3	
0010 0000h to 00FF FFFFh	Internal peripheral bus 6	E2 DataFlash memory and ROM (for programming/erasure)
8000 0000h to FEFF FFFFh	Memory bus 2	ROM (for reading only)
FF00 0000h to FFFF FFFFh		

15.2 Description of Buses

15.2.1 CPU Buses

The CPU buses consist of the instruction and operand buses, which are connected to internal main bus 1. As the names suggest, the instruction bus is used to fetch instructions for the CPU, while the operand bus is used for operand access. The instruction bus is 64 bits while the operand bus is 32 bits.

Connection of the instruction and operand buses to RAM and ROM provides the CPU with direct access to these areas, i.e. access is not via internal main bus 1. However, only reading is possible in direct access to ROM by the CPU; programming and erasure are handled via an internal peripheral bus.

Bus requests for instruction fetching and operand access are arbitrated through internal main bus 1. The order of priority is operand access then instruction fetching.

If instruction fetching and operand access are requested for different buses (memory bus 1, memory bus 2, and internal main bus 1), the bus-access operations can proceed simultaneously. For example, parallel access to ROM and RAM is possible.

15.2.2 Memory Buses

The memory buses consist of memory bus 1 and memory bus 2. RAM is connected to memory bus 1 and ROM is connected to memory bus 2. The memory buses are 64 bits. Requests for bus mastership from the CPU buses (instruction fetching and operand) and internal main bus 2 are arbitrated through memory buses 1 and 2.

The priority order of CPU bus and internal main bus 2 can be set using the memory bus 1 (RAM) priority control bits (BPRA[1:0]) and memory bus 2 (ROM) priority control bits (BPRO[1:0]) in the bus priority control register (BUSPRI) for the corresponding memory buses. When the priority order is fixed, internal main bus 2 has priority over the CPU bus (operand over instruction fetching). When the priority order is toggled, the bus for which a request has been accepted has lower priority.

15.2.3 Internal Main Buses

The internal main buses consist of a bus for use by the CPU (internal main bus 1) and a bus for use by the other bus-master modules, i.e. the DTC (internal main bus 2).

Bus requests for instruction fetching and operand access are arbitrated through internal main bus 1. The order of priority is operand access then instruction fetching.

Requests for bus mastership from the DTC is arbitrated by internal main bus 2. The order of priority is as shown in Table 15.3.

If the CPU and another bus master are requesting access to different buses (on-chip memory, internal peripheral buses 1 to 3 and 6), the respective bus-access operations can proceed simultaneously.

However, when the CPU executes the XCHG instruction, requests for bus access from masters other than the CPU are not accepted until data transfer for the XCHG instruction is completed regardless of the bus priority control register (BUSPRI) setting. Furthermore, requests for bus access from masters other than the DTC are not accepted during reading and writing-back of transfer control information for the DTC.

Table 15.3 Order of Priority for Bus Masters

Priority	Bus Master
High	DTC
↑	
Low	CPU

15.2.4 Internal Peripheral Buses

Connection of peripheral modules to the internal peripheral buses is as described in Table 15.4.

Table 15.4 Connection of Peripheral Modules to the Internal Peripheral Buses

Type of Bus	Peripheral Modules
Internal peripheral bus 1	DTC, interrupt controller, and bus error monitoring section
Internal peripheral bus 2	Peripheral modules other than those connected to internal peripheral buses 1 and 3
Internal peripheral bus 3	CMPC
Internal peripheral bus 6	ROM (P/E)/E2 DataFlash

Requests for bus mastership from the CPU (internal main bus 1) and other bus masters (internal main bus 2) are arbitrated through internal peripheral buses 1 to 3 and 6.

The priority order of the two internal main buses can be set using the bus priority control register (BUSPRI). The priority order can be set with the internal peripheral bus 1 priority control bits (BUSPRI.BPIB[1:0]), internal peripheral buses 2 and 3 priority control bits (BUSPRI.BPGB[1:0]), and internal peripheral bus 6 priority control bits (BUSPRI.BPFB[1:0]) for the corresponding internal peripheral buses. When the priority order is fixed, internal main bus 2 has priority over internal main bus 1. When the priority order is toggled, a bus has a lower priority when the request of that bus is accepted.

The order of accepting requests may change depending on the BUSPRI setting (see Figure 15.2).

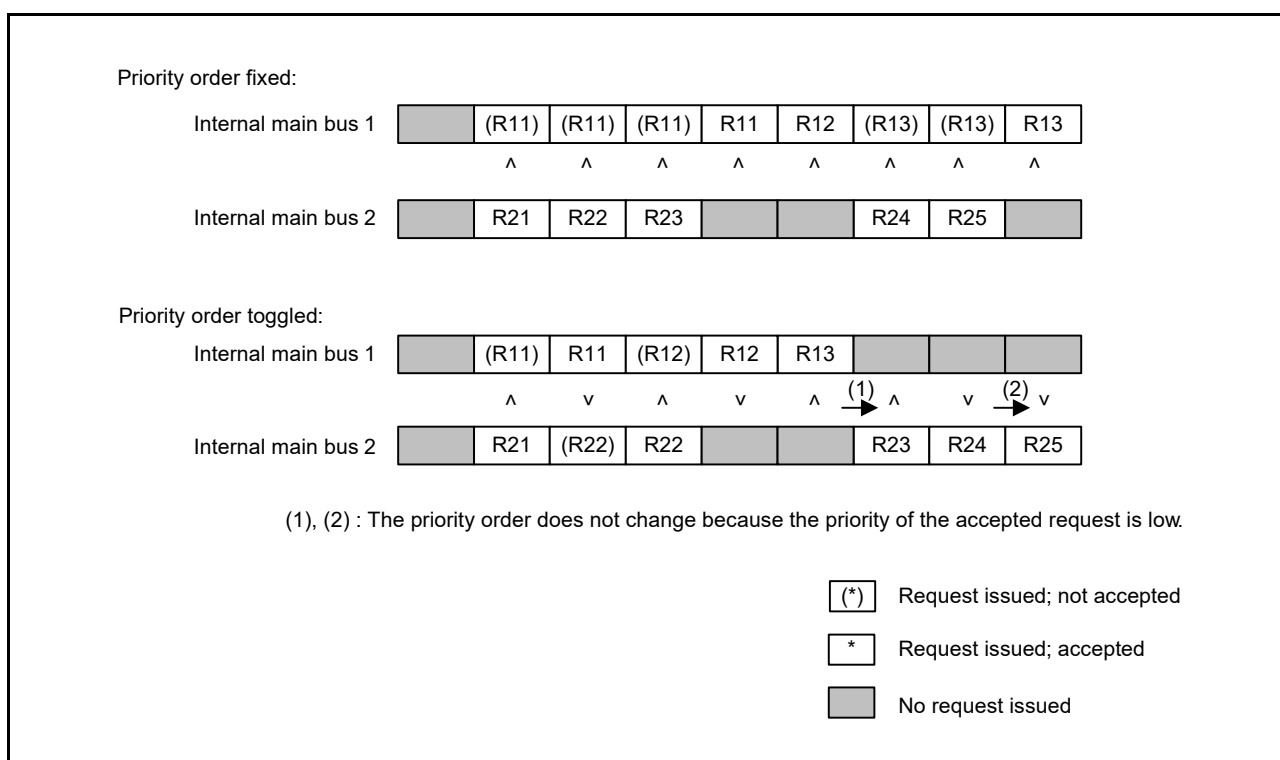


Figure 15.2 Priority Order Between Internal Peripheral Bus Accesses

15.2.5 Write Buffer Function (Internal Peripheral Bus)

The internal peripheral bus has the write buffer function, which allows the next round of bus access to start, before the current write access is completed, in write access. However, if the following round of bus access is from the same bus master but to the different internal peripheral bus, it is suspended until the bus operations already in progress are completed. When read access to the internal memory is scheduled after the write access to the internal peripheral bus from the CPU, the following round of bus access can be started before the current bus operation is completed and thus the order of accesses may be changed (see Figure 15.3).

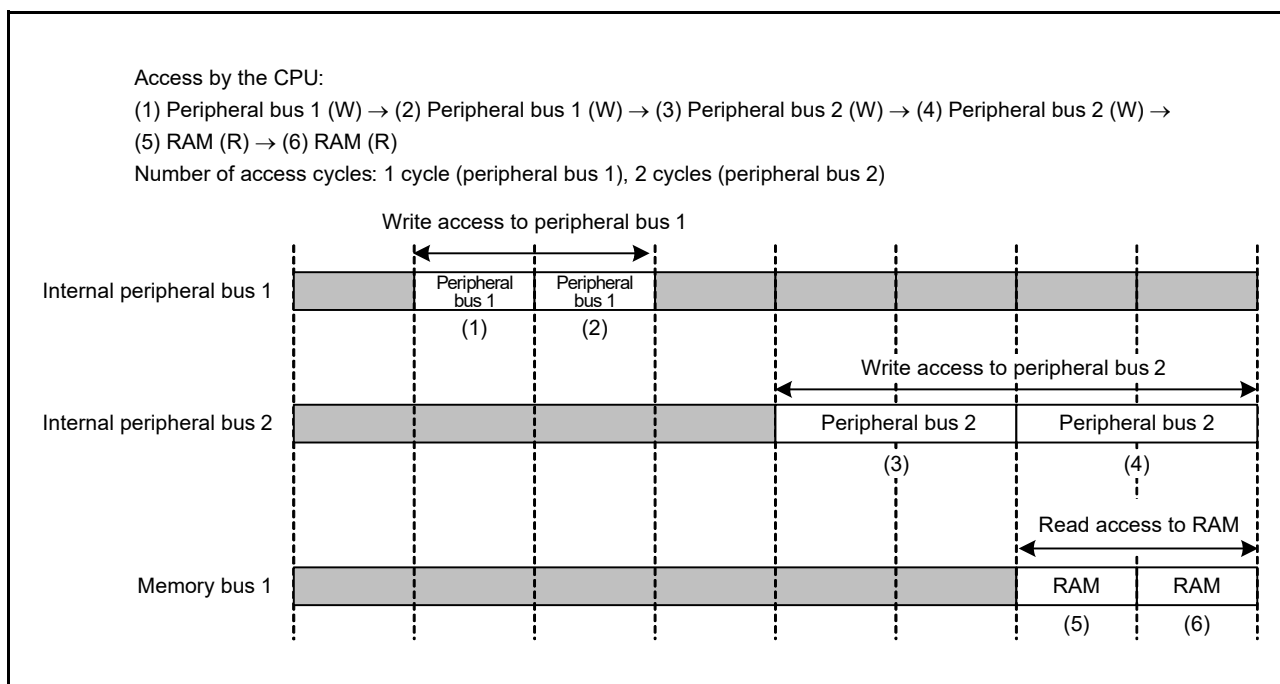


Figure 15.3 Write Buffer Function

15.2.6 Parallel Operation

Parallel operation is possible when different bus-master modules are requesting access to different slave modules. For example, if the CPU is fetching an instruction from ROM and an operand from RAM, the DTC is able to handle transfer between a peripheral bus and peripheral bus at the same time.

An example of parallel operations is shown in Figure 15.4. In this example, the CPU is able to employ the instruction and operand buses for simultaneous access to ROM and RAM, respectively. Furthermore, the DTC simultaneously employs internal main bus 2 for access to a peripheral bus during access to RAM and ROM by the CPU.

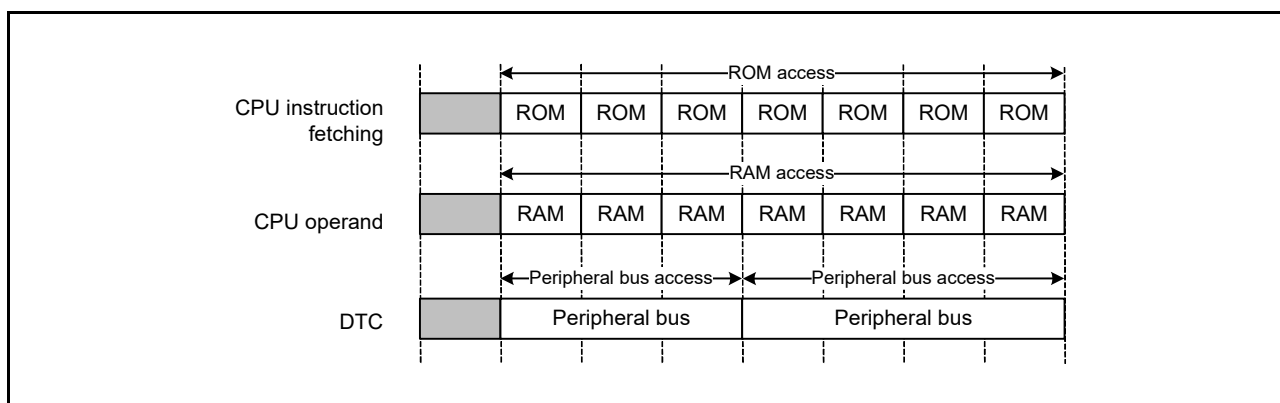


Figure 15.4 Example of Parallel Operations

15.2.7 Restrictions

(1) Prohibition of Access that Spans Multiple Areas of Address Space

Single access that spans two areas of the address space is prohibited, and operation of such an access is not guaranteed. Ensure that a single word or longword access does not span across two areas by crossing address space area boundaries.

(2) Restrictions on RMPA and String-Manipulation Instructions

- (a) The allocation of data to be handled by RMPA or string-manipulation instructions to I/O registers is prohibited, and operation is not guaranteed if this restriction is not observed.

15.3 Register Descriptions

15.3.1 Bus Error Status Clear Register (BERCLR)

Address(es): BSC.BERCLR 0008 1300h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	STSCLR
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	STSCLR	Status Clear	0: Invalid 1: Bus error status register cleared	(W)*1
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only writing 1 is effective; i.e. writing 0 has no effect.

STSCLR Bit (Status Clear)

Writing 1 to this bit clears the bus error status registers 1 and 2 (BERSR1 and BERSR2).

Writing 0 has no effect. It is read as 0.

15.3.2 Bus Error Monitoring Enable Register (BEREN)

Address(es): BSC.BEREN 0008 1304h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	TOEN	IGAEN
0	0	0	0	0	0	0	0

Value after reset:

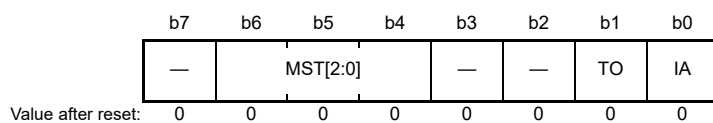
Bit	Symbol	Bit Name	Description	R/W
b0	IGAEN	Illegal Address Access Detection Enable	0: Illegal address access detection is disabled. 1: Illegal address access detection is enabled.	R/W
b1	TOEN	Timeout Detection Enable*1, *2	0: Bus timeout detection is disabled. 1: Bus timeout detection is enabled.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. When detection is disabled (the TOEN bit is cleared to 0), bus access can cause the bus to freeze.

Note 2. Do not set the TOEN bit to 0 (bus timeout detection disabled) while timeout errors are being detected.

15.3.3 Bus Error Status Register 1 (BERSR1)

Address(es): BSC.BERSR1 0008 1308h



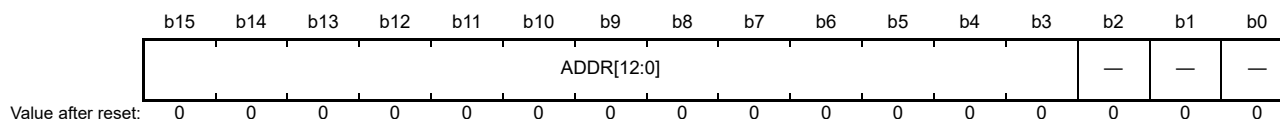
Bit	Symbol	Bit Name	Description	R/W
b0	IA	Illegal Address Access	0: Illegal address access not made 1: Illegal address access made	R
b1	TO	Timeout	0: Timeout not generated 1: Timeout generated	R
b3, b2	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R
b6 to b4	MST[2:0]	Bus Master Code	b6 b4 0 0 0: CPU 0 0 1: Reserved 0 1 0: Reserved 0 1 1: DTC 1 0 0: Reserved 1 0 1: Reserved 1 1 0: Reserved 1 1 1: Reserved	R
b7	—	Reserved	This bit is read as 0. Writing to this bit has no effect.	R

MST[2:0] Bits (Bus Master Code)

These bits indicate the bus master that accessed a bus when a bus error occurred.

15.3.4 Bus Error Status Register 2 (BERSR2)

Address(es): BSC.BERSR2 0008 130Ah



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R
b15 to b3	ADDR[12:0]	Bus Error Occurrence Address	The upper 13 bits of an address that was accessed when a bus error occurred (in units of 512 Kbytes).	R

15.3.5 Bus Priority Control Register (BUSPRI)

Address(es): BSC.BUSPRI 0008 1310h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	BPFB[1:0]	—	—	—	BPGB[1:0]	—	BPIB[1:0]	—	BPRO[1:0]	—	BPRA[1:0]	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b1, b0	BPRA[1:0]	Memory Bus 1 (RAM) Priority Control	b1 b0 0 0: The order of priority is fixed. 0 1: The order of priority is toggled. 1 0: Setting prohibited 1 1: Setting prohibited	R/(W) *1
b3, b2	BPRO[1:0]	Memory Bus 2 (ROM) Priority Control	b3 b2 0 0: The order of priority is fixed. 0 1: The order of priority is toggled. 1 0: Setting prohibited 1 1: Setting prohibited	R/(W) *1
b5, b4	BPIB[1:0]	Internal Peripheral Bus 1 Priority Control	b5 b4 0 0: The order of priority is fixed. 0 1: The order of priority is toggled. 1 0: Setting prohibited 1 1: Setting prohibited	R/(W) *1
b7, b6	BPGB[1:0]	Internal Peripheral Buses 2 and 3 Priority Control	b7 b6 0 0: The order of priority is fixed. 0 1: The order of priority is toggled. 1 0: Setting prohibited 1 1: Setting prohibited	R/(W) *1
b9, b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b11, b10	BPFB[1:0]	Internal Peripheral Bus 6 Priority Control	b11 b10 0 0: The order of priority is fixed. 0 1: The order of priority is toggled. 1 0: Setting prohibited 1 1: Setting prohibited	R/(W) *1
b15 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. These bits can be written to only once while the DTC is stopped. When they are written to more than one time, the operation is not guaranteed.

BPRA[1:0] Bits (Memory Bus 1 (RAM) Priority Control)

These bits specify the priority order for memory bus 1 (RAM).

When the priority order is fixed, internal main bus 2 has priority over the CPU bus.

When the priority order is toggled, a bus has a lower priority when the request of that bus is accepted.

BPRO[1:0] Bits (Memory Bus 2 (ROM) Priority Control)

These bits specify the priority order for memory bus 2 (ROM).

When the priority order is fixed, internal main bus 2 has priority over the CPU bus.

When the priority order is toggled, a bus has a lower priority when the request of that bus is accepted.

BPIB[1:0] Bits (Internal Peripheral Bus 1 Priority Control)

These bits specify the priority order for internal peripheral bus 1.

When the priority order is fixed, internal main bus 2 has priority over internal main bus 1.

When the priority order is toggled, a bus has a lower priority when the request of that bus is accepted.

BPGB[1:0] Bits (Internal Peripheral Buses 2 and 3 Priority Control)

These bits specify the priority order for internal peripheral buses 2 and 3.

When the priority order is fixed, internal main bus 2 has priority over internal main bus 1.

When the priority order is toggled, a bus has a lower priority when the request of that bus is accepted.

BPFB[1:0] Bits (Internal Peripheral Bus 6 Priority Control)

These bits specify the priority order for internal peripheral bus 6.

When the priority order is fixed, internal main bus 2 has priority over internal main bus 1.

When the priority order is toggled, a bus has a lower priority when the request of that bus is accepted.

15.4 Bus Error Monitoring Section

The bus error monitoring section monitors the individual areas for bus errors, and when a bus error occurs, the error is indicated to the bus master.

15.4.1 Type of Bus Error

There is a illegal address access bus error.

Illegal address access is the detection of access to an illegal area.

15.4.1.1 Illegal Address Access

When the illegal address access detection enable bit (IGAEN) in the bus error monitoring enable register (BEREN) is set to 1, access to an illegal address area leads to illegal address access errors.

The address ranges where access will lead to illegal address access errors are listed in Table 15.5.

15.4.1.2 Timeout

When the timeout detection enable bit in the bus error monitoring enable register is enabled (BEREN.TOEN = 1), bus access that is not completed within 768 cycles leads to a timeout error.

- Internal peripheral buses (2 and 3): Bus access is not completed within 768 peripheral module clock (PCLKB) cycles from the start of the access.
If a timeout error occurs, accesses from the bus master are not accepted for 256 PCLKB cycles.
- Internal peripheral bus (6): Bus access is not completed within 768 FlashIF clock (FCLK) cycles from the start of the access.
If a timeout error occurs, accesses from the bus master are not accepted for 256 FCLK cycles.

15.4.2 Operations When a Bus Error Occurs

When a bus error occurs, the error is indicated to the CPU. Operation is not guaranteed when a bus error occurs.

- Bus error indication to the CPU
An interrupt is generated. The ICU.IERn register can specify whether to generate an interrupt in the case of a bus error.

15.4.3 Conditions Leading to Bus Errors

Table 15.5 lists the type of bus errors for each area in the respective address space.

If an illegal address access error is detected when no bus error has occurred (bus error status register n (BERSRn; n = 1, 2) is cleared), the detected error is reflected in the BERSRn register. Once a bus error occurs, no subsequent bus errors are reflected in the register unless the register is cleared.

If bus errors are simultaneously caused by two or more bus masters, error information of only one bus master is reflected. Once a bus error occurs, the status is retained until the BERSRn register is cleared.

Table 15.5 Type of Bus Errors

Address	Type of Area	Type of Error	
		Illegal Address Access	Timeout
0000 0000h to 0007 FFFFh	Memory bus 1	—	—
0008 0000h to 0008 7FFFh	Internal peripheral bus 1	—	—
0008 8000h to 0009 FFFFh	Internal peripheral bus 2	Δ	—
000A 0000h to 000B FFFFh	Internal peripheral bus 3	Δ	—
000C 0000h to 000F FFFFh	Reserved area	✓	—
0010 0000h to 00FF FFFFh	Internal peripheral bus 6	Δ	—
0100 0000h to 07FF FFFFh	Reserved area	✓	—
0800 0000h to 0FFF FFFFh	Reserved area	—	—
1000 0000h to 7FFF FFFFh	Reserved area	✓	—
8000 0000h to FFFF FFFFh	Memory bus 2	—	—

—: A bus error does not result.

Δ: A bus error may or may not result.

✓: A bus error results.

Note: The capacity of the RAM, data flash, and ROM differs depending on the product. For details, refer to section 36, RAM, and section 37, Flash Memory (FLASH).

16. Data Transfer Controller (DTCb)

This MCU incorporates a data transfer controller (DTC).

The DTC is triggered by an interrupt request to perform data transfers.

In addition to the conventional methods of DTC transfer (normal, repeat, block, and chain), DTCb supports sequential transfer, in which it handles a series of transfers made up of a combination of the other methods. In sequential transfer, the data that is initially transferred selects one from possible 256 sequences for execution. The DTCb can divide one sequence into several transfers depending on how the parts of the sequence are combined.

16.1 Overview

Table 16.1 lists the specifications of the DTC, and Figure 16.1 shows a block diagram of the DTC.

Table 16.1 DTC Specifications

Item	Description
Number of transfer channels	The same number as all interrupt sources that can start the DTC transfer.
Transfer modes	- Normal transfer mode A single transfer request leads to a single data transfer. - Repeat transfer mode A single transfer request leads to a single data transfer. The transfer address is returned to the transfer start address after the number of data transfers corresponding to "repeat size". The maximum number of repeat transfers is 256, and the maximum data transfer size is 256×32 bits, 1024 bytes. - Block transfer mode A single transfer request leads to the transfer of a single block. The maximum block size is 256×32 bits = 1024 bytes.
Chain transfer	- Multiple types of data transfers can sequentially be executed in response to a single request. - Either "performed only when the transfer counter becomes 0" or "every time" can be selected.
Sequence transfer	A series of complicated transfers can be registered as a sequence. Any sequence can be selected by the transfer data and executed. - Only one trigger source can be set at a time. - Up to 256 sequences for a single trigger source - The data that is initially transferred in response to a transfer request determines a sequence - The whole sequence can be executed on a single request, or be suspended in the middle of the sequence and resumed on the next transfer request (division of sequence).
Transfer space	- In short-address mode: 16 Mbytes (Areas from 0000 0000h to 007F FFFFh and FF80 0000h to FFFF FFFFh except reserved areas) - In full-address mode: 4 Gbytes (Area from 0000 0000h to FFFF FFFFh except reserved areas)
Data transfer units	- Single data: 1 byte (8 bits), 1 word (16 bits), 1 longword (32 bits) - Single block size: 1 to 256 data
CPU interrupt source	- An interrupt request can be generated to the CPU on a request source for a data transfer. - An interrupt request can be generated to the CPU after a single data transfer. - An interrupt request can be generated to the CPU after data transfer of specified volume.
Read skip	Reading of the transfer information can be skipped when the same transfer is repeated.
Write-back skip	Write-back of the transferred data that is not updated can be skipped when the address of the transfer source or destination is fixed.
Write-back disable	Allows disabling the write-back of transfer information.
Displacement addition	The displacement value can be added to the transfer source address (for each transfer information)
Low power consumption function	Module stop state can be set.

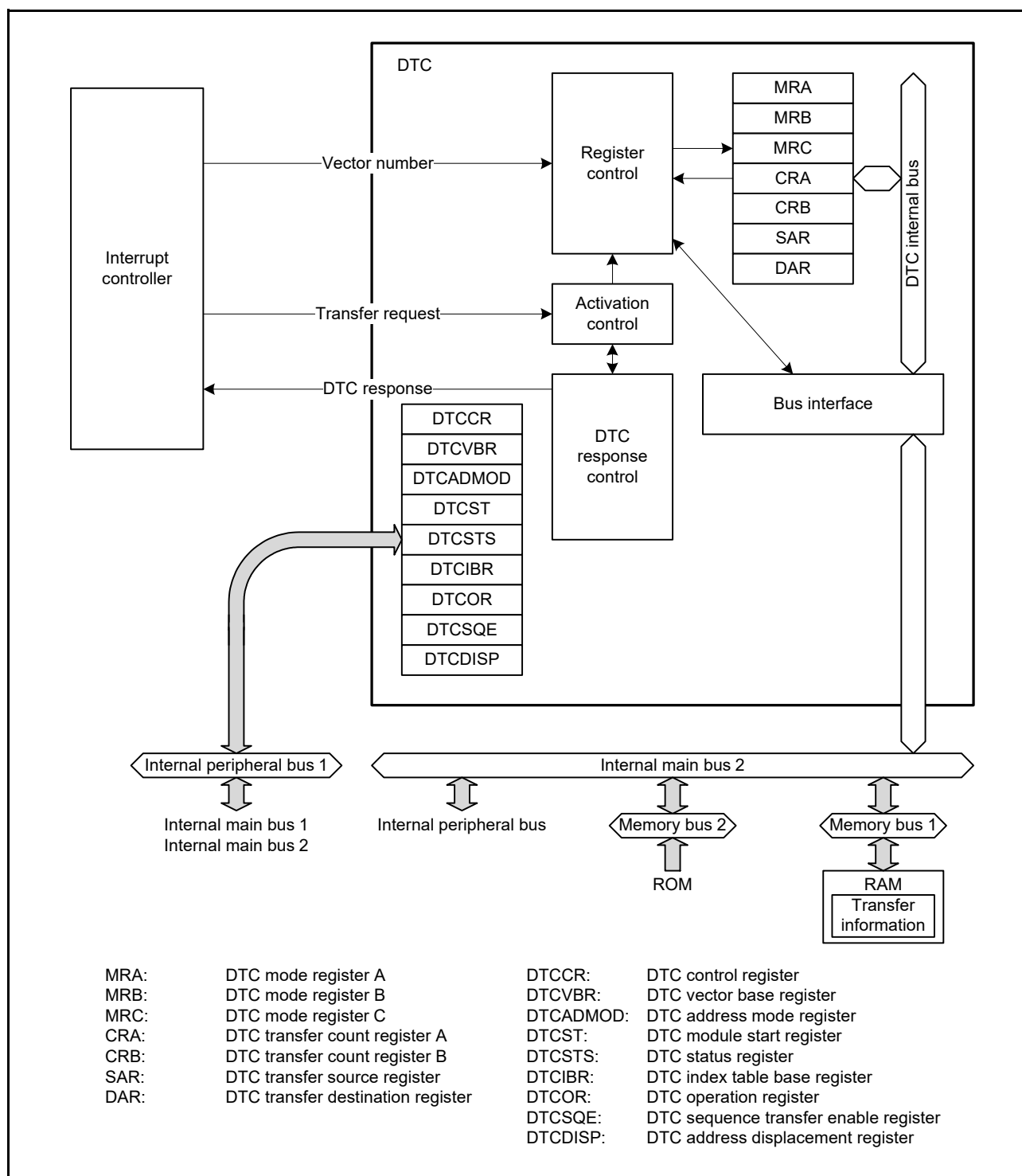


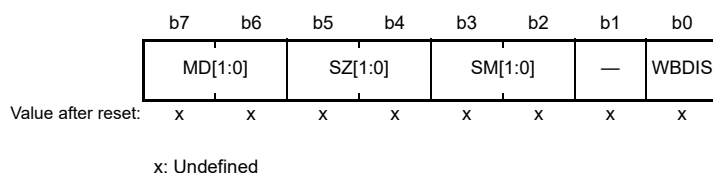
Figure 16.1 DTC Block Diagram

16.2 Register Descriptions

Registers MRA, MRB, MRC, SAR, DAR, CRA, and CRB are DTC internal registers, which cannot be directly accessed from the CPU. Values to be set in these DTC internal registers are placed in the RAM area as transfer information. When accepting a transfer request, the DTC reads the transfer information from the RAM area and sets it in the internal registers. After the data transfer ends, the values of the updated internal register are written back to the RAM area as transfer information.

16.2.1 DTC Mode Register A (MRA)

Address(es): (inaccessible directly from the CPU)



Bit	Symbol	Bit Name	Description	R/W
b0	WBDIS	Write-Back Disable	0: Writes back the transfer information on completion of the data transfer 1: Does not write back the transfer information on completion of the data transfer	—
b1	—	Reserved	Set this bit to 0.	—
b3, b2	SM[1:0]	Transfer Source Address Addressing Mode	b3 b2 0 0: The address in the SAR register is fixed. (write-back to SAR is skipped.) 0 1: The address in the SAR register is fixed. (write-back to SAR is skipped.) 1 0: The SAR value is incremented after a data transfer. (+1 when the SZ[1:0] bits are 00b, +2 when 01b, +4 when 10b) 1 1: The SAR value is decremented after a data transfer. (–1 when the SZ[1:0] bits are 00b, –2 when 01b, –4 when 10b)	—
b5, b4	SZ[1:0]	DTC Data Transfer Size	b5 b4 0 0: Byte (8-bit) transfer 0 1: Word (16-bit) transfer 1 0: Longword (32-bit) transfer 1 1: Setting prohibited	—
b7, b6	MD[1:0]	DTC Transfer Mode Select	b7 b6 0 0: Normal transfer mode 0 1: Repeat transfer mode 1 0: Block transfer mode 1 1: Setting prohibited	—

MRA register is used to select the DTC operating mode and cannot be accessed directly from the CPU.

WBDIS Bit (Write-Back Disable)

The WBDIS bit selects whether to write back the transfer information.

When the bit is 0, updated transfer information is written back.

When the bit is 1, updated transfer information is not written back even with the setting of that address is incremented after a transfer, and the same data transfer is executed every time for each transfer request. The transfer information can be stored in ROM because the transfer information is not written back.

While the WBDIS bit is 1, operation for each transfer mode is as follows:

(1) Normal transfer and repeat transfer modes

1-byte, 1-word, or 1-longword of data is transferred on a single transfer request. The transfer address and transfer count are not updated so that the same transfer is repeated on each transfer request. When the transfer count is 1, the ICU.DTCERn.DTCE bit is not set to 0, and data transfer continues in response to the next transfer request.

(2) Block transfer mode

1-block of data is transferred on a single transfer request. The transfer address and transfer count are not updated so that the same block transfer is repeated on each transfer request. When the block transfer count is 1, the ICU.DTCERn.DTCE bit is not set to 0, and data transfer continues in response to the next transfer request.

When setting the DISPE bit to 1, set the MRA.WBDIS bit to 1 (does not write back the transfer information). If the value of the WBDIS bit in any transfer information is 1, set the DTCCR.RRS bit to 0 (so that reading of the transfer information is not skipped).

16.2.2 DTC Mode Register B (MRB)

Address(es): (inaccessible directly from the CPU)

	b7	b6	b5	b4	b3	b2	b1	b0
	CHNE	CHNS	DISEL	DTS	DM[1:0]		INDX	SQEND
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	SQEND	Sequence Transfer End	0: Continue the sequence transfer 1: End the sequence transfer	—
b1	INDX	Index Table Reference	0: Does not refer to the index table 1: Refers the index table based on the transferred data*1	—
b3, b2	DM[1:0]	Transfer Destination Address Addressing Mode	b3 b2 0 0: The address in the DAR register is fixed. (Write-back to DAR is skipped.) 0 1: The address in the DAR register is fixed. (Write-back to DAR is skipped.) 1 0: The DAR value is incremented after data transfer. (+1 when the MRA.SZ[1:0] bits are 00b, +2 when 01b, +4 when 10b) 1 1: The DAR value is decremented after data transfer. (-1 when the MRA.SZ[1:0] bits are 00b, -2 when 01b, -4 when 10b)	—
b4	DTS	DTC Transfer Mode Select	0: Transfer destination side is repeat area or block area. 1: Transfer source side is repeat area or block area.	—
b5	DISEL	DTC Interrupt Select	0: An interrupt request to the CPU is generated on completion of the specified number of data transfers. 1: An interrupt request to the CPU is generated for each data transfer.	—
b6	CHNS	DTC Chain Transfer Select	0: Chain transfer is performed on completion of each transfer. 1: Chain transfer is performed only when the transfer counter is changed from 1 to 0 or 1 to CRAH.	—
b7	CHNE	DTC Chain Transfer Enable	0: Chain transfer is disabled. 1: Chain transfer is enabled.	—

Note 1. Set the MRA.MD[1:0] bits to 00b (normal transfer mode) when setting the INDX bit to 1.

MRB register is used to select the DTC operating mode and cannot be accessed directly from the CPU.

SQEND Bit (Sequence Transfer End)

The SQEND bit selects whether to continue or end sequence transfer. Refer to Table 16.2 for details.

This bit can only be set to 1 for transfer information referred to by the DTC index table. Set this bit to 0 for transfer information referred to by the DTC vector table.

INDX Bit (Index Table Reference)

When the value of the INDX bit in transfer information that is read is 1, a sequence transfer proceeds. Refer to Table 16.2 for details.

Set this bit to 0 for transfer information which is not associated with sequence transfer or is not intended to start sequence transfer. Do not allow transfer requests to be generated by the sources different from that specified in the DTCSQE register but having the INDX bit set to 1.

Table 16.2 Values of Bits CHNE, SQEND, and INDX in the Sequence Transfer and DTC Operation

CHNE Bit	SQEND Bit	INDX Bit	Operation	Usage
0	0	1	Start sequence transfer	Use this setting for the transfer information that is first read in response to a transfer request from the source specified in the DTCSQE register.
1	0	0	Continue sequence transfer	Use this setting for the first or intermediate transfer information in a sequence.
0	0	0	Suspend sequence transfer	Use this setting for the first or intermediate transfer information in a sequence.
0	1	0	End sequence transfer	Use this setting with the last transfer information in a sequence.
0	1	1	End current sequence transfer and start new sequence transfer	Use this setting with the last transfer information in a sequence.

Note: Do not set the values other than listed above.

DTS Bit (DTC Transfer Mode Select)

The DTS bit specifies the side (transfer source or destination) to be a repeat area or block area in repeat transfer mode or block transfer mode.

CHNS Bit (DTC Chain Transfer Select)

The CHNS bit selects the chain transfer condition.

When the CHNE bit is 0, setting of the CHNS bit is ignored. For details on the conditions to select the chain transfer, refer to Table 16.4, Chain Transfer Conditions.

When the next transfer is chain transfer, completion of the specified number of transfers is not determined, the interrupt status flag for the request source is not cleared, and an interrupt request to the CPU is not generated.

CHNE Bit (DTC Chain Transfer Enable)

The CHNE bit enables or disables chain transfer.

The chain transfer condition is selected by the CHNS bit.

For details of chain transfer, refer to section 16.4.6, Chain Transfer.

Refer to Table 16.2 for the setting value to be used in the sequence transfer.

16.2.3 DTC Mode Register C (MRC)

Address(es): (inaccessible directly from the CPU)

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	DISPE
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	DISPE	Displacement Addition	0: The displacement value is not added to the transfer source address. 1: The displacement value is added to the transfer source address.	—
b7 to b1	—	Reserved	Set these bits to 0.	—

The MRC register is used to select DTC operating mode and cannot be accessed directly from the CPU.

This register can only be used in full-address mode, but not in short-address mode. Therefore, set the DTCADMOD.SHORT bit to 0 (full-address mode) when using the displacement addition function.

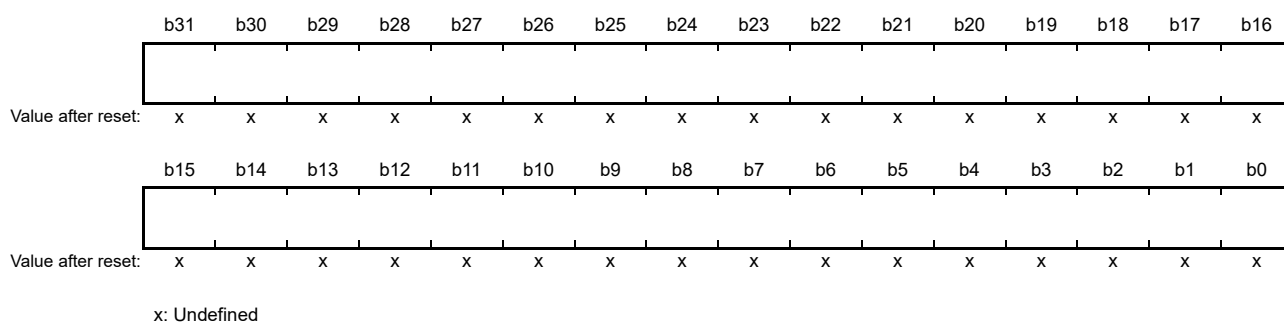
DISPE Bit (Displacement Addition)

This bit specifies whether to use the SAR + DTCDISP value as the transfer source address.

When setting the DISPE bit to 1, set the MRA.WBDIS bit to 1 (does not write back the transfer information) and set the DTCCR.RRS bit to 0 (transfer information read is not skipped).

16.2.4 DTC Transfer Source Register (SAR)

Address(es): (inaccessible directly from the CPU)



SAR register is used to set the transfer source start address.

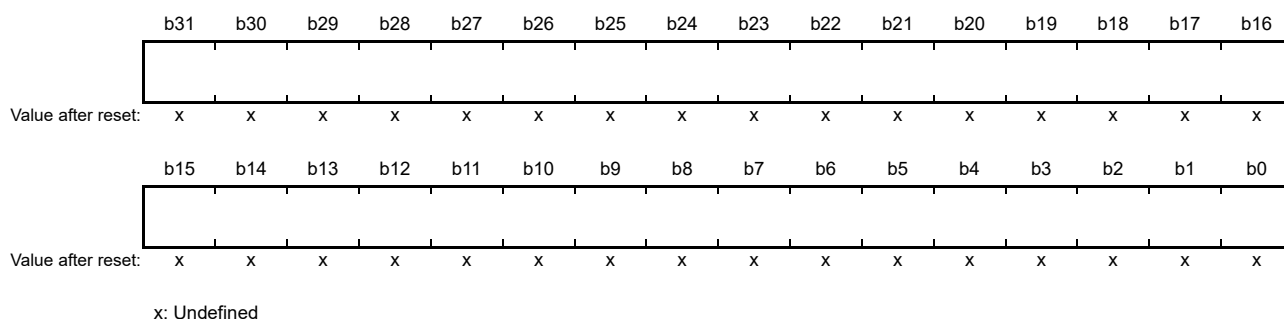
In full-address mode, 32 bits are valid.

In short-address mode, lower 24 bits are valid and upper 8 bits (b31 to b24) are ignored. The address of this register is extended by the value specified by b23.

SAR register cannot be accessed directly from the CPU.

16.2.5 DTC Transfer Destination Register (DAR)

Address(es): (inaccessible directly from the CPU)



DAR register is used to set the transfer destination start address.

In full-address mode, 32 bits are valid.

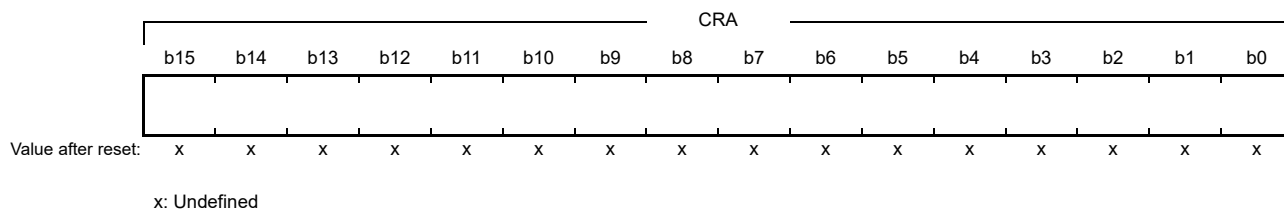
In short-address mode, lower 24 bits are valid and upper 8 bits (b31 to b24) are ignored. The address of this register is extended by the value specified by b23.

DAR register cannot be accessed directly from the CPU.

16.2.6 DTC Transfer Count Register A (CRA)

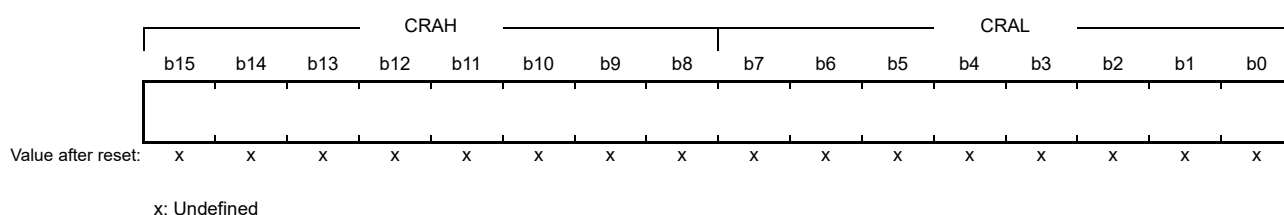
- Normal transfer mode

Address(es): (inaccessible directly from the CPU)



- Repeat transfer mode/block transfer mode

Address(es): (inaccessible directly from the CPU)



Symbol	Register Name	Description	R/W
CRAL	Transfer Counter A Lower Register	Set transfer count. This register functions as a transfer counter during data transfer.	—
CRAH	Transfer Counter A Upper Register	Set transfer count. This register functions as a reload register during data transfer.	—

Note: The function depends on transfer mode.

Note: Set CRAH and CRAL to the same value in repeat transfer mode and block transfer mode.

This register is for counting the number of transfers and cannot be accessed directly from the CPU.

(1) Normal transfer mode (MRA.MD[1:0] bits = 00b)

CRA register functions as a 16-bit transfer counter in normal transfer mode.

The transfer count is 1, 65535, and 65536 when the set value is 0001h, FFFFh, and 0000h, respectively.

The CRA value is decremented (−1) at each data transfer.

(2) Repeat transfer mode (MRA.MD[1:0] bits = 01b)

The CRAH register retains the transfer count and the CRAL register functions as an 8-bit transfer counter.

The transfer count is 1, 255, and 256 when the set value is 01h, FFh, and 00h, respectively.

The CRAL value is decremented (−1) at each data transfer. When it reaches 00h, the CRAH value is reloaded to the CRAL register.

(3) Block transfer mode (MRA.MD[1:0] bits = 10b)

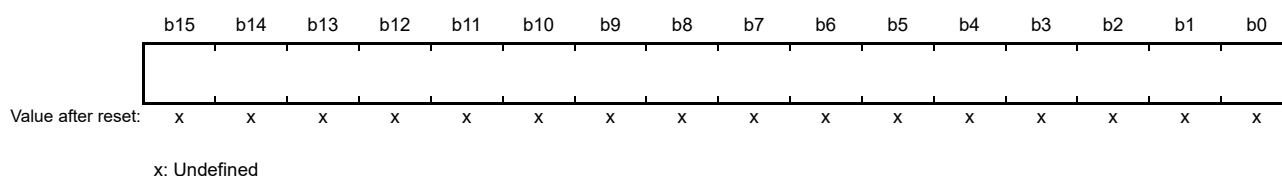
The CRAH register retains the block size and the CRAL register functions as an 8-bit block size counter.

The transfer count is 1, 255, and 256 when the set value is 01h, FFh, and 00h, respectively.

The CRAL value is decremented (−1) at each data transfer. When it reaches 00h, the CRAH value is reloaded to the CRAL register.

16.2.7 DTC Transfer Count Register B (CRB)

Address(es): (inaccessible directly from the CPU)



CRB register is used to set the block transfer count for block transfer mode and cannot be accessed directly from the CPU.

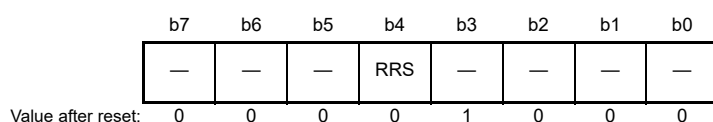
The transfer count is 1, 65535, and 65536 when the set value is 0001h, FFFFh, and 0000h, respectively.

The CRB value is decremented (–1) when the final data of a single block size is transferred.

When normal transfer mode or repeat transfer mode is selected, this register is not used and the set value is ignored.

16.2.8 DTC Control Register (DTCCR)

Address(es): DTC.DTCCR 0008 2400h



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b3	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b4	RRS	DTC Transfer Information Read Skip Enable*1	0: Transfer information read is not skipped. 1: Transfer information read is skipped when vector numbers match.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Set this bit to 0 when using the sequence transfer.

DTCCR register is used to control the DTC operation.

RRS Bit (DTC Transfer Information Read Skip Enable)

The DTC vector number is compared with the vector number in the previous data transfer.

When these vector numbers match and the RRS bit is set to 1, DTC data transfer is performed without reading the transferred information. However, when the previous transfer was chain transfer, the transferred information is read regardless of the value of the RRS bit.

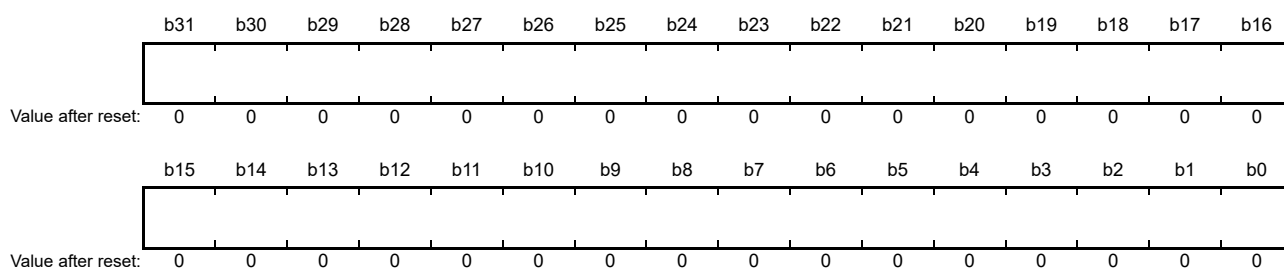
Furthermore, when the transfer counter (CRA register) became 0 during the previous normal transfer and when the transfer counter (CRB register) became 0 during the previous block transfer, the transferred information is read regardless of the RRS bit value.

If the value of the MRA.WBDIS bit in any transfer information is 1, set the RRS bit to 0. Note that the MRA.WBDIS bit should be set to 1 when the MRC.DISPE bit is set to 1.

Like chain transfer, sequence transfer handles sequences of multiple types of data transfer. When sequence transfer is to be used, set the RRS bit to 0 so that the previous data transfer will not be repeated.

16.2.9 DTC Vector Base Register (DTCVBR)

Address(es): DTC.DTCVBR 0008 2404h

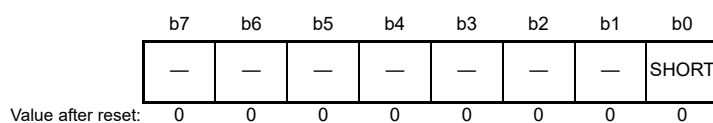


The DTCVBR register is used to set the base address for calculating the address to which the DTC vector is allocated. Values for the upper 4 bits (b31 to b28) cannot be written but reflect the value written to b27. The lower 10 bits are reserved and the values are fixed to 0. Write 0 to the lower 10 bits if necessary.

It can be set in the range of 0000 0000h to 07FF FC00h and F800 0000h to FFFF FC00h in 1-Kbyte units.

16.2.10 DTC Address Mode Register (DTCADMOD)

Address(es): DTC.DTCADMOD 0008 2408h



Bit	Symbol	Bit Name	Description	R/W
b0	SHORT	Short-Address Mode Set*1	0: Full-address mode 1: Short-address mode	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Set this bit to 0 (full-address mode) when using the sequence transfer.

DTCADMOD register is used to specify the area accessible by the DTC.

SHORT Bit (Short-Address Mode Set)

This bit is used to select address mode of registers SAR and DAR.

Full-address mode allows the DTC to access to a 4-Gbyte space (0000 0000h to FFFF FFFFh).

Short-address mode allows the DTC to access to a 16-Mbyte space (0000 0000h to 007F FFFFh and FF80 0000h to FFFF FFFFh).

16.2.11 DTC Module Start Register (DTCST)

Address(es): DTC.DTCST 0008 240Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	DTCST
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DTCST	DTC Module Start	0: DTC module stop 1: DTC module start	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

DTCST Bit (DTC Module Start)

Set the DTCST bit to 1 to enable the DTC to accept transfer requests. When this bit is set to 0, transfer requests are no longer accepted.

If this bit is set to 0 during data transfer, the accepted transfer request is active until the processing is completed.

Set the DTCST bit to 0 before making a transition to the module stop state, deep sleep mode, or software standby mode.

Set the DTCST bit to 1 to resume the data transfer after returning from the module stop state, deep sleep mode, or software standby mode.

For details on transitions to the module stop state, deep sleep mode, and software standby mode, refer to section 16.8, Low Power Consumption Function, and section 11, Low Power Consumption.

16.2.12 DTC Status Register (DTCSTS)

Address(es): DTC.DTCSTS 0008 240Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ACT	—	—	—	—	—	—	—	VECN[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	VECN[7:0]	DTC Active Vector Number Monitoring Flag	These bits indicate the vector number for the request source when data transfer is in progress. The value is only valid if data transfer is in progress (the value of the ACT flag is 1).	R
b14 to b8	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R
b15	ACT	DTC Active Flag	0: Data transfer is not in progress. 1: Data transfer is in progress.	R

VECN[7:0] Flags (DTC Active Vector Number Monitoring Flag)

While data transfer is in progress, these bits indicate the vector number corresponding to the request source for the transfer.

When the DTCSTS register is read, the value of the VECN[7:0] flags is valid if the value of the ACT flag was 1 (data transfer is in progress) and invalid if the value of the ACT flag was 0 (data transfer is not in progress).

For the correspondence between the DTC request sources and the vector addresses, refer to [section 14.3.1, Interrupt Vector Table](#) in [section 14, Interrupt Controller \(ICUb\)](#).

ACT Flag (DTC Active Flag)

This flag indicates the state of data transfer operation.

[Setting condition]

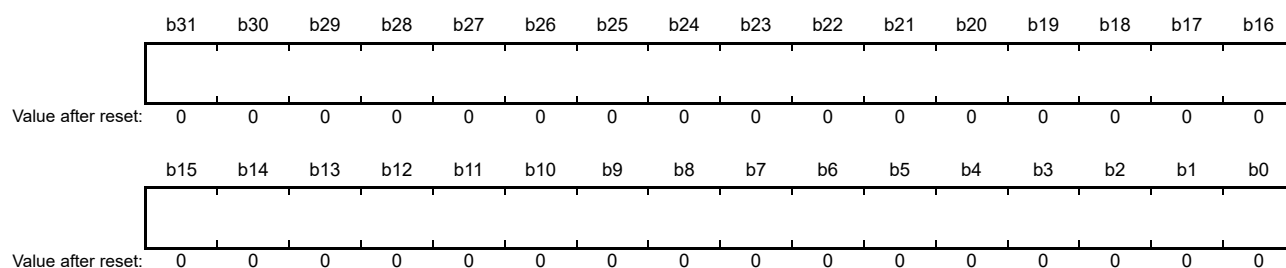
- When the data transfer is started by a transfer request.
- When the sequence transfer is resumed.

[Clearing condition]

- When the data transfer is completed in response to a transfer request.
- When the sequence transfer is suspended.

16.2.13 DTC Index Table Base Register (DTCIBR)

Address(es): DTC.DTCIBR 0008 2410h



The DTCIBR register is used to set the base address for calculating the address to which the DTC index is allocated.

Values for the upper 4 bits (b31 to b28) cannot be written but reflect the value written to b27.

The lower 10 bits (b9 to b0) are reserved bits and fixed to 0. When writing this register, set these bits to 0.

It can be set in the range of 0000 0000h to 07FF FC00h and F800 0000h to FFFF FC00h in 1-Kbyte units.

16.2.14 DTC Operation Register (DTCOR)

Address(es): DTC.DTCOR 0008 2414h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	SQTFRL
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SQTFRL	Sequence Transfer Terminate	Writing 1 to this bit terminates the sequence transfer in progress. This bit is read as 0.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R

The DTCOR register sets the operation of the DTC module.

SQTFRL Bit (Sequence Transfer Terminate)

Setting the SQTFRL bit to 1 terminates the sequence transfer in progress.

When the DTCSQE.ESPSEL bit is 1 (Sequence transfer is enabled), follow the procedure shown in Figure 16.2 to terminate the sequence transfer.

Writing 1 to the bit, while no sequence transfer is performed, have no effect.

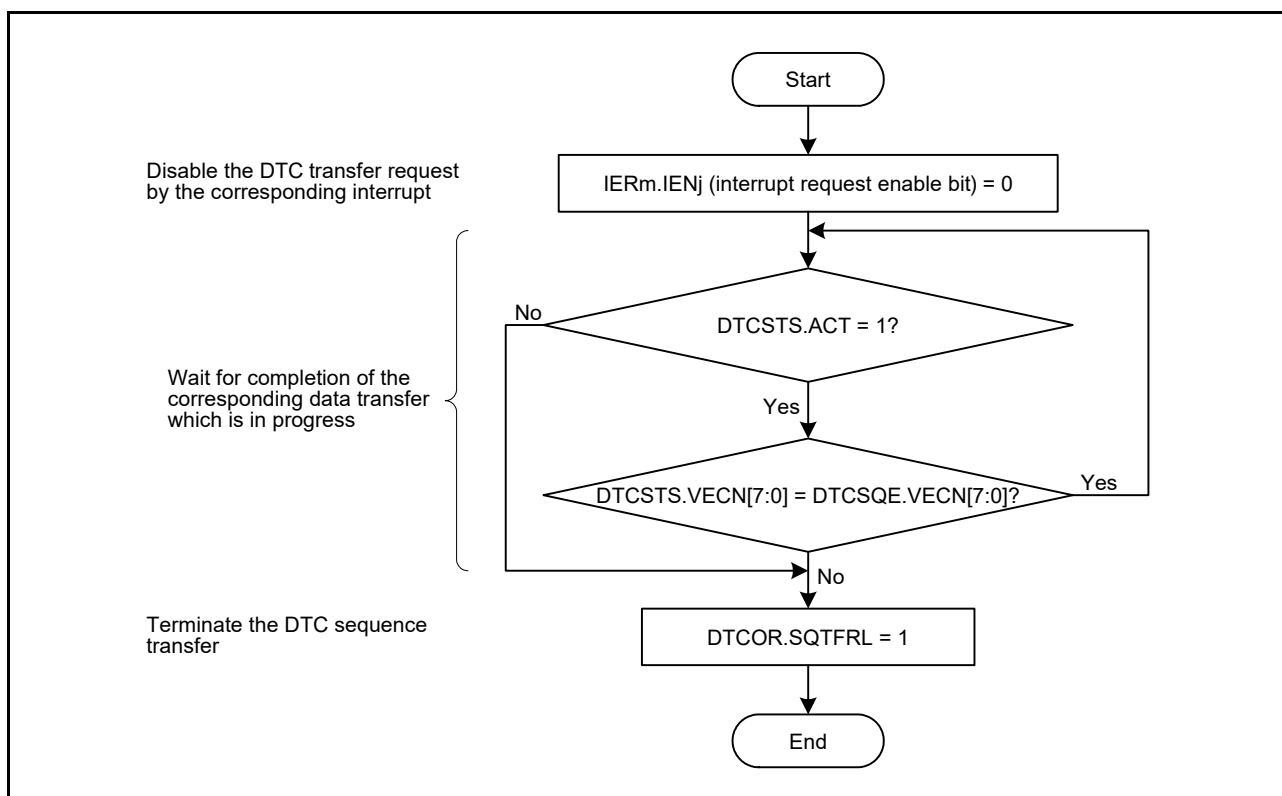


Figure 16.2 Procedure to Terminate Sequence Transfer

16.2.15 DTC Sequence Transfer Enable Register (DTCSQE)

Address(es): DTC.DTCSQE 0008 2416h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ESPSEL	—	—	—	—	—	—	—	VECN[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	VECN[7:0]	Sequence Transfer Vector Number Setting	Specify the vector number by which a sequence transfer is enabled. The value is only valid when the ESPSEL bit is 1.	R/W
b14 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b15	ESPSEL	Sequence Transfer Enable	0: Sequence transfer is disabled. 1: Sequence transfer is enabled.	R/W

The DTCSQE register is used to specify sequence transfer. Follow Figure 16.24 for details on the setting procedure.

VECN[7:0] Bit (Sequence Transfer Vector Number Setting)

This bit is used to specify for which vector number to perform sequence transfer. Sequence transfer can occur only for this trigger source.

section 14.3.1, Interrupt Vector Table in section 14, Interrupt Controller (ICUb) shows the relationship between the trigger source and the vector number.

ESPSEL Bit (Sequence Transfer Enable)

The ESPSEL bit specifies whether sequence transfer is used.

Set the DTCADMOD.SHORT bit to 0 (full address mode), when setting the ESPSEL bit to 1.

16.2.16 DTC Address Displacement Register (DTCDISP)

Address(es): DTC.DTCDISP 0008 2418h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The DTCDISP register is used to specify the displacement value to add to the DTC transfer source address.

If MRC.DISPE bit is 1, the value SAR + DTCDISP is used as the transfer source address.

16.3 Request Sources

The DTC data transfer is triggered by an interrupt request. Setting the ICU.DTCERn.DTCE bit (n = interrupt vector number) to 1 selects the corresponding interrupt request as a request source for the DTC.

For the correspondence between the DTC request sources and the vector addresses, refer to section 14.3.1, Interrupt Vector Table in section 14, Interrupt Controller (ICUb). For request by software, refer to section 14.2.5, Software Interrupt Generation Register (SWINTR) in section 14, Interrupt Controller (ICUb).

Once the DTC has accepted a transfer request, it does not accept another transfer request until transfer for that single request is completed, regardless of the priority of the requests.

When multiple transfer requests are generated during data transfer by the DTC, the request with the highest priority on completion of the current transfer is accepted. When multiple transfer requests are generated while the DTCST.DTCST bit is 0 (DTC module stop), the request with the highest priority at the moment when the bit is subsequently set to 1 (DTC module start) is accepted.

The DTC performs the following operations at the start of a single data transfer (or the last of the consecutive transfers in the case of a chain transfer).

- On completion of a specified number of data transfer, the ICU.DTCERn.DTCE bit is set to 0 and an interrupt is requested to the CPU.
- If the MRB.DISEL bit is 1, an interrupt is requested to the CPU on completion of data transfer.
- For the other transfers, the interrupt status flag of the request source is set to 0 at the start of data transfer.

16.3.1 Allocating Transfer Information and DTC Vector Table

The DTC reads the start address of the transfer information corresponding to each request source from the vector table and reads the transfer information starting at that address.

The vector table should be located so that the lower 10 bits of the base address (start address) are 0. Use the DTC vector base register (DTCVBR) to set the base address of the DTC vector table.

Transfer information is allocated in the RAM area. Transfer information can be allocated in the ROM area when the MRA.WBDIS bit is set to 1. The start address of the transfer information n with vector number n should be allocated at DTCVBR + 4n.

Transfer information should be aligned on a 4-byte boundary. The size of a transfer information is 12 bytes in short-address mode or 16 bytes in full-address mode. Use the DTCADMOD.SHORT bit to select short-address mode (SHORT bit = 1) or full-address mode (SHORT bit = 0).

Figure 16.3 shows the relationship between the DTC vector table and transfer information.

Figure 16.4 shows the allocation of transfer information in the RAM area. The lower addresses vary according to the endian of the corresponding allocation area. For details, refer to section 16.9.2, Allocating Transfer Information.

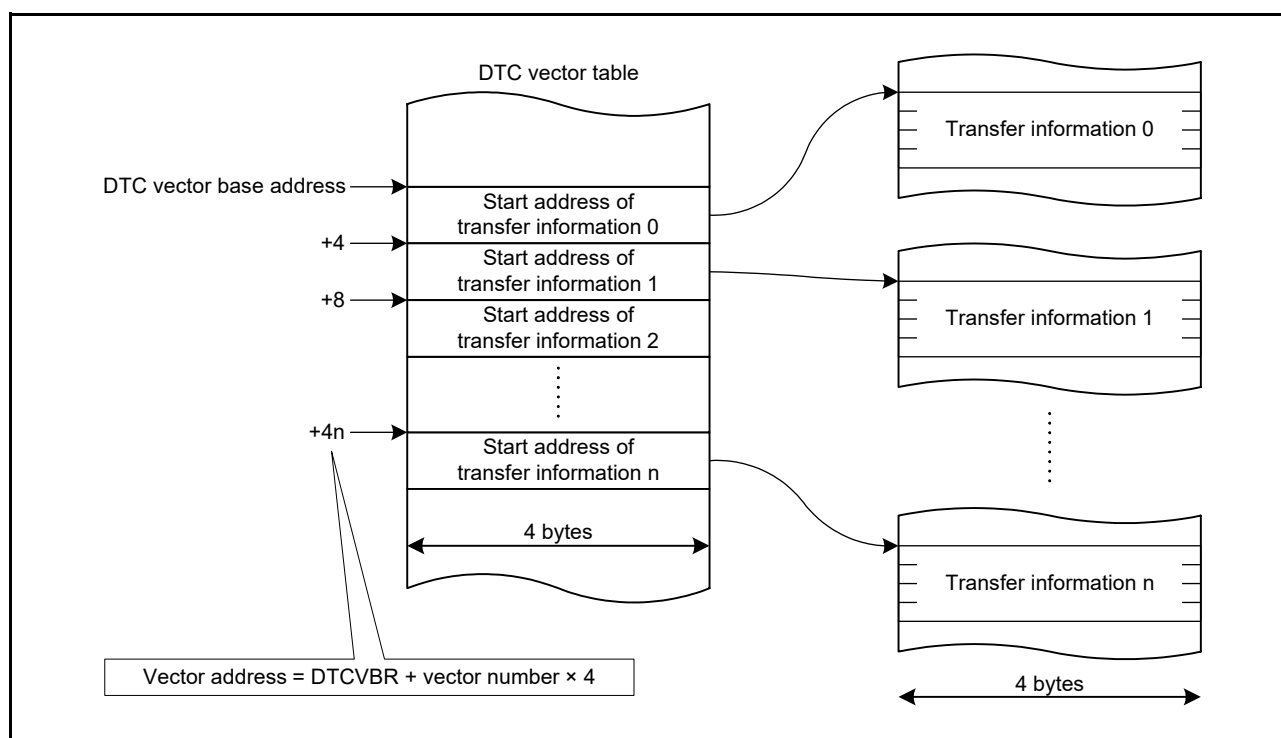


Figure 16.3 DTC Vector Table and Transfer Information

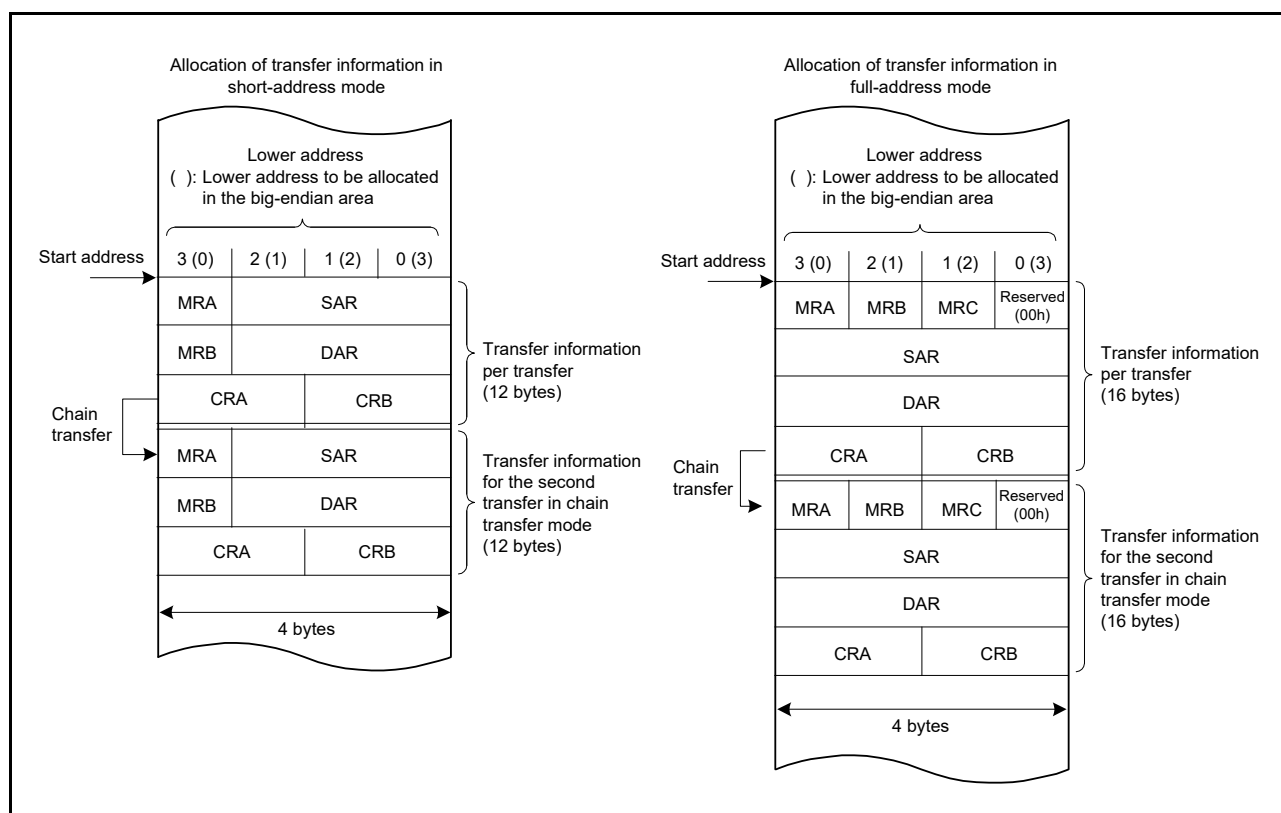


Figure 16.4 Allocation of Transfer Information in the RAM Area

16.4 Operation

The DTC transfers data in accordance with the transfer information. Storage of the transfer information in the RAM area is required before DTC operation.

When the DTC accepts a transfer request, it reads the DTC vector corresponding to the vector number. Next, the DTC reads transfer information from the address pointed by the DTC vector, transfers data, and then writes back the transfer information after the data transfer. Allocating transfer information in the RAM area allows data transfer of arbitrary number of channels.

There are three transfer modes: normal transfer mode, repeat transfer mode, and block transfer mode.

Set a transfer source address in the SAR register and a transfer destination address in the DAR register. The SAR and DAR registers are updated after the transfer according to the respective settings (increment, decrement, or fixed).

Table 16.3 lists transfer modes of the DTC.

Table 16.3 Transfer Modes of the DTC

Transfer Mode	Data Size Transferred on Single Transfer Request	Increment/Decrement of Memory Address	Settable Transfer Count
Normal transfer mode	1 byte/1 word/1 longword	Incremented/decremented by 1, 2, or 4 or address fixed	1 to 65536
Repeat transfer mode*1	1 byte/1 word/1 longword	Incremented/decremented by 1, 2, or 4 or address fixed	1 to 256*3
Block transfer mode*2	Block size specified in CRAH (1 to 256 bytes/1 to 256 words/1 to 256 longwords)	Incremented/decremented by 1, 2, or 4 or address fixed	1 to 65536

Note 1. Set transfer source or transfer destination in the repeat area.

Note 2. Set transfer source or transfer destination in the block area.

Note 3. After data transfer of the specified count, the initial state is restored and the operation is continued (repeated).

Setting the MRB.CHNE bit to 1 allows multiple transfers (chain transfer) on a single transfer request. The setting in combination with the MRB.CHNS bit enables a chain transfer when the specified number of data transfers is completed. Figure 16.5 shows the operation flowchart of the DTC. Table 16.4 lists chain transfer conditions.

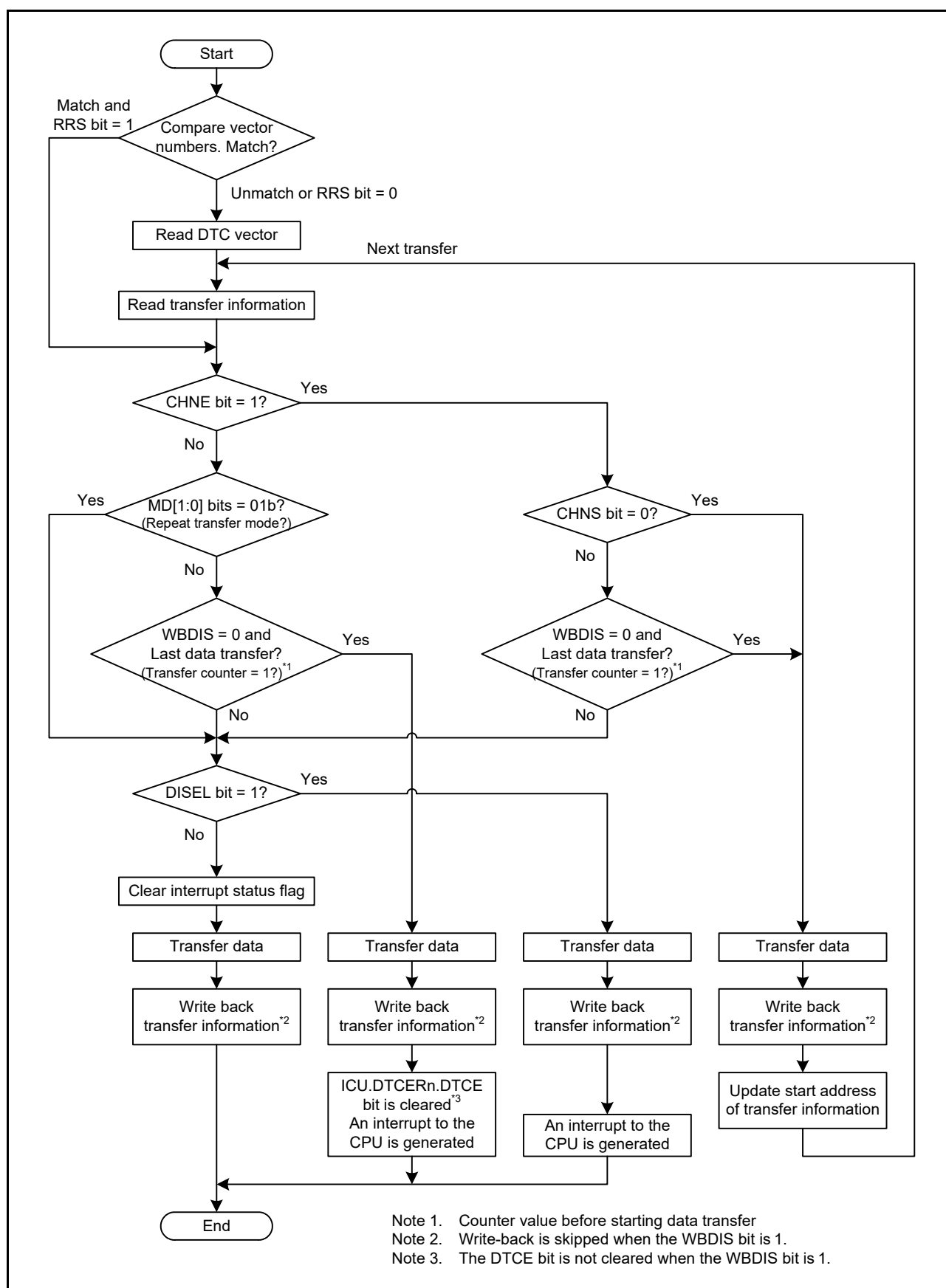


Figure 16.5 Operation Flowchart of the DTC

Table 16.4 Chain Transfer Conditions

First Transfer				Second Transfer*3				Data Transfer
CHNE Bit	CHNS Bit	DISEL Bit	Transfer Counter*1,*2	CHNE Bit	CHNS Bit	DISEL Bit	Transfer Counter*1,*2	
0	—	0	Other than (1 → 0)	—	—	—	—	Ends after the first transfer
0	—	0	(1 → 0)	—	—	—	—	Ends after the first transfer with an interrupt request to the CPU
0	—	1	—	—	—	—	—	
1	0	—	—	0	—	0	Other than (1 → 0)	Ends after the second transfer
				0	—	0	(1 → 0)	Ends after the second transfer with an interrupt request to the CPU
				0	—	1	—	
1	1	0	Other than (1 → *)	—	—	—	—	Ends after the first transfer
1	1	—	(1 → *)	0	—	0	Other than (1 → 0)	Ends after the second transfer
				0	—	0	(1 → 0)	Ends after the second transfer with an interrupt request to the CPU
				0	—	1	—	
1	1	1	Other than (1 → *)	—	—	—	—	Ends after the first transfer with an interrupt request to the CPU

Note 1. The transfer counters used depend on transfer modes as follows:

Normal transfer mode: CRA register

Repeat transfer mode: CRAL register

Block transfer mode: CRB register

Note 2. On completion of data transfer, the counters operate as follows:

1 → 0 in normal and block transfer modes

1 → CRAH in repeat transfer mode

(1 → *) in the table indicates both of the two operations above.

Note 3. Chain transfer can be selected for the second or subsequent transfers. The condition combination of “second transfer and the CHNE bit is 1” is omitted.

16.4.1 Transfer Information Read Skip Function

Reading of DTC vector and transfer information can be skipped by the setting of the DTCCR.RRS bit.

When a DTC transfer request is accepted, the current DTC vector number is compared with the DTC vector number in the previous data transfer. When these vector numbers match and the RRS bit is 1, the DTC does not read the DTC vector and transfer information, and transfers data according to the transfer information remained in the DTC.

However, when the previous transfer was chain transfer, the DTC vector and transfer information are read. Furthermore, when the transfer counter (CRA register) became 0 during the previous normal transfer and when the transfer counter (CRB register) became 0 during the previous block transfer, transfer information is read regardless of the value of the RRS bit. Figure 16.14 shows an example of transfer information read skip.

When updating the vector table and transfer information, set the RRS bit to 0, update the vector table and transfer information, and then set the RRS bit to 1. Setting the RRS bit to 0 discards the vector numbers retained in the DTC. The updated DTC vector table and transfer information are read in the next data transfer.

16.4.2 Transfer Information Write-Back Skip Function

16.4.2.1 Write-Back Skip by Fixing Addresses

When the MRA.SM[1:0] bits or the MRB.DM[1:0] bits are set to “address is fixed” (00b or 01b), a part of transfer information is not written back. This function is performed independently of the setting of short-address mode or full-address mode.

Table 16.5 lists transfer information write-back skip conditions and applicable registers. The CRA and CRB registers are written back independently of the setting of short-address mode or full-address mode.

Furthermore, in full-address mode, write-back of registers MRA, MRB, and MRC is skipped.

Table 16.5 Transfer Information Write-Back Skip Conditions and Applicable Registers

MRA.SM[1:0] Bits		MRB.DM[1:0] Bits		SAR Register	DAR Register
b3	b2	b3	b2		
0	0	0	0	Skip	Skip
0	0	0	1		
0	1	0	0		
0	1	0	1		
0	0	1	0	Skip	Write-back
0	0	1	1		
0	1	1	0		
0	1	1	1		
1	0	0	0	Write-back	Skip
1	0	0	1		
1	1	0	0		
1	1	0	1		
1	0	1	0	Write-back	Write-back
1	0	1	1		
1	1	1	0		
1	1	1	1		

16.4.2.2 Write-Back Skip by the MRA.WBDIS Bit

When the MRA.WBDIS bit is 1, the transfer information (SAR, DAR, CRA, and CRB) is not written back regardless of the settings of the transfer information.

The transfer information on the memory is not updated, data can be transferred by the DTC without copying the transfer information from ROM to RAM. Skipping a write-back reduces time for post-processing of the data transfer.

16.4.3 Normal Transfer Mode

This mode allows 1-byte, 1-word, or 1-longword data transfer on a single transfer request. The transfer count can be set to 1 to 65536.

Transfer source addresses and transfer destination addresses can be set to increment, decrement, or fixed independently. This mode enables an interrupt request to the CPU to be generated at the end of specified-count transfer.

Table 16.6 lists register functions in normal transfer mode, and Figure 16.6 shows the memory map of normal transfer mode.

Table 16.6 Register Functions in Normal Transfer Mode

Register	Description	Value Written Back by Writing Transfer Information*1
SAR	Transfer source address	Increment/decrement/fixed*2
DAR	Transfer destination address	Increment/decrement/fixed*2
CRA	Transfer counter A	CRA – 1
CRB	Transfer counter B	Not updated

Note 1. Write-back operation is skipped when the MRA.WBDIS bit is 1.

Note 2. Write-back operation is skipped when address is fixed.

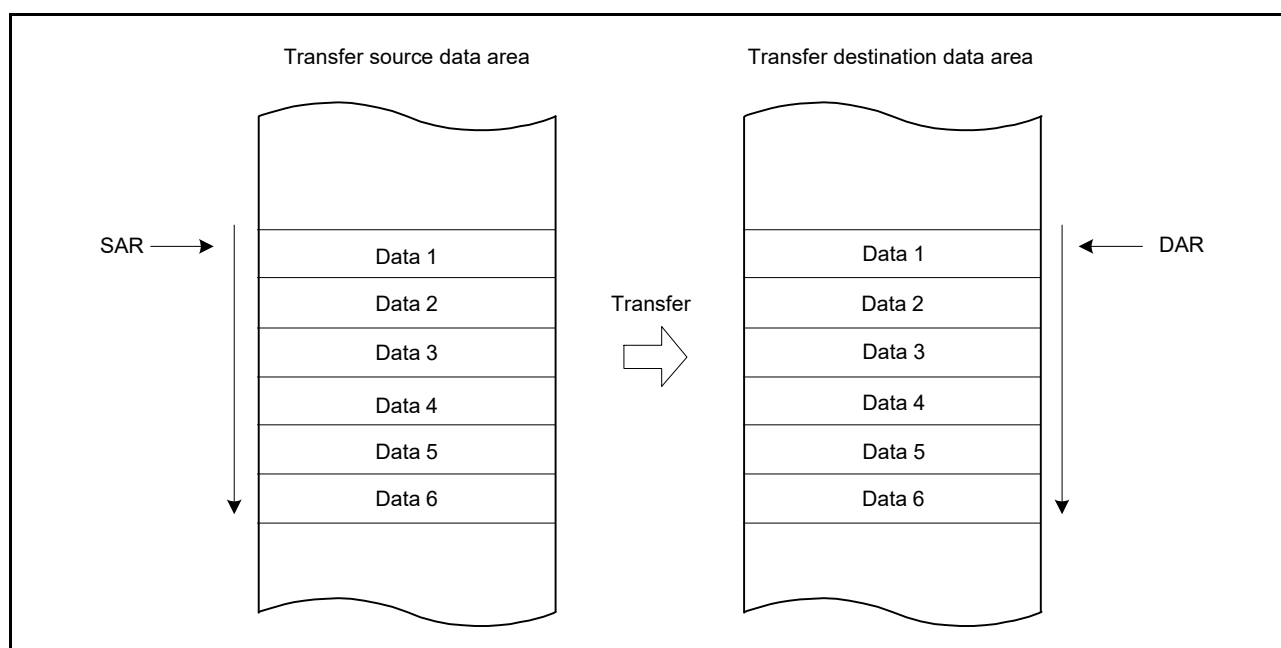


Figure 16.6 Memory Map of Normal Transfer Mode

16.4.4 Repeat Transfer Mode

This mode allows 1-byte, 1-word, or 1-longword data transfer on a single transfer request.

Specify either transfer source or transfer destination for the repeat area by the MRB.DTS bit. The transfer count can be set to 1 to 256. When the specified-count transfer is completed, the initial value of the address register specified in the transfer counter and the repeat area is restored and transfer is repeated. The other address register is incremented or decremented continuously or remains unchanged.

When the transfer counter CRAL is decreased to 00h in repeat transfer mode, the CRAL value is updated to the value set in the CRAH register. Thus the transfer counter does not become 00h, which disables an interrupt request to be generated to the CPU when the MRB.DISEL bit is set to 0 (an interrupt request to the CPU is generated on completion of the specified number of data transfers).

Table 16.7 lists the register functions in repeat transfer mode, and Figure 16.7 shows the memory map of repeat transfer mode.

Table 16.7 Register Functions in Repeat Transfer Mode

Register	Description	Value Written Back by Writing Transfer Information*1		
		When CRAL ≠ 1	When CRAL = 1	
			When the MRB.DTS Bit is 0	When the MRB.DTS Bit is 1
SAR	Transfer source address	Increment/decrement/fixed*2	Increment/decrement/fixed*2	SAR register initial value
DAR	Transfer destination address	Increment/decrement/fixed*2	DAR register initial value	Increment/decrement/fixed*2
CRAH	Retains initial value of transfer counter	CRAH	CRAH	
CRAL	Transfer counter A	CRAL – 1	CRAH	
CRB	Transfer counter B	Not updated	Not updated	

Note 1. Write-back operation is skipped when the MRA.WBDIS bit is 1.

Note 2. Write-back operation is skipped when address is fixed.

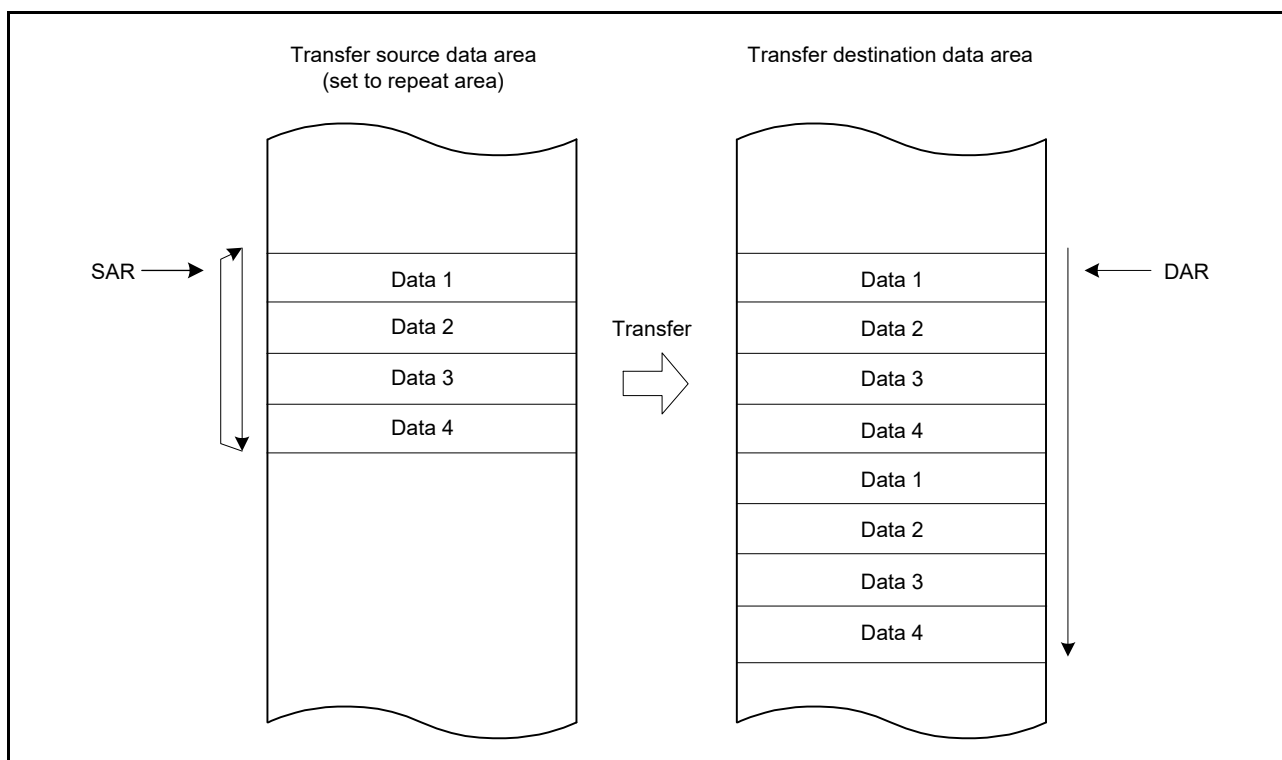


Figure 16.7 Memory Map of Repeat Transfer Mode (Transfer Source: Repeat Area)

16.4.5 Block Transfer Mode

This mode allows single-block data transfer on a single transfer request.

Specify either transfer source or transfer destination for the block area by the MRB.DTS bit. The block size can be set to 1 to 256 bytes, 1 to 256 words, or 1 to 256 longwords.

When transfer of the specified one block is completed, the initial values of the block size counter CRAH and the address register (the SAR register when the MRB.DTS bit is 1 or the DAR register when the DTS bit is 0) specified in the block area are restored. The other address register is incremented or decremented continuously or remains unchanged.

The transfer count (block count) can be set to 1 to 65536. This mode enables an interrupt request to the CPU to be generated at the end of specified-count block transfer.

Table 16.8 lists register functions in block transfer mode, and Figure 16.8 shows the memory map of block transfer mode.

Table 16.8 Register Functions in Block Transfer Mode

Register	Description	Value Written Back by Writing Transfer Information*1	
		When MRB.DTS Bit is 0	When MRB.DTS Bit is 1
SAR	Transfer source address	Increment/decrement/fixed*2	SAR register initial value
DAR	Transfer destination address	DAR register initial value	Increment/decrement/fixed*2
CRAH	Retains initial value of block size	CRAH	
CRAL	Block size counter	CRAH	
CRB	Block transfer counter	CRB – 1	

Note 1. Write-back operation is skipped when the MRA.WBDIS bit is 1.

Note 2. Write-back operation is skipped when address is fixed.

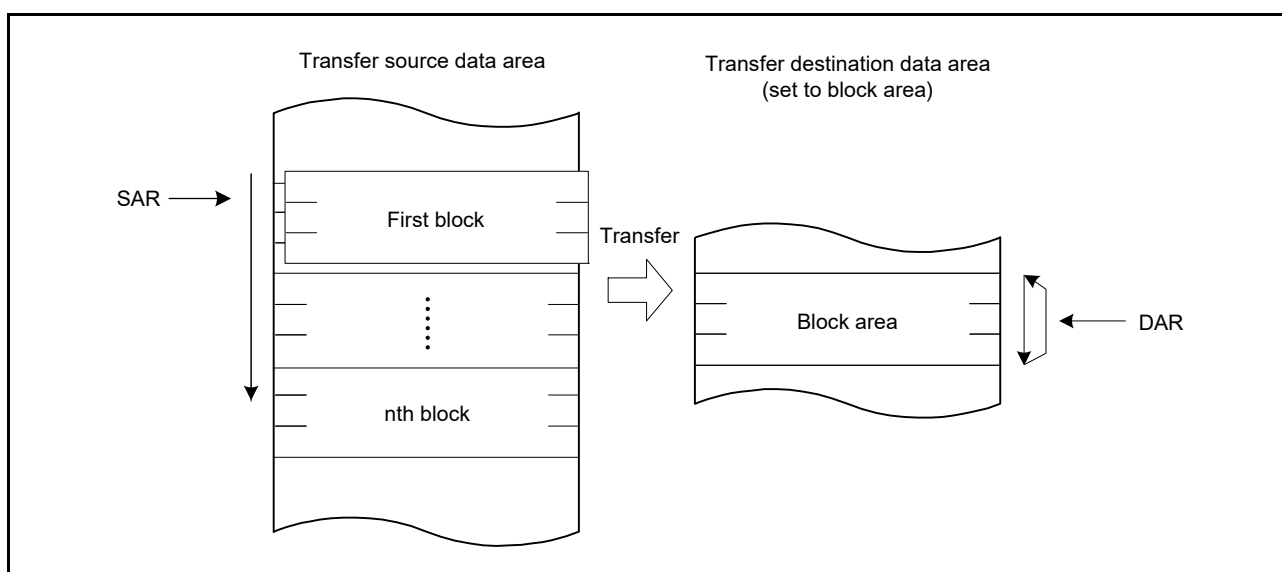


Figure 16.8 Memory Map of Block Transfer Mode (Transfer Destination: Block Area)

16.4.6 Chain Transfer

Setting the MRB.CHNE bit to 1 allows chain transfer to be performed continuously on a single transfer request.

If the MRB.CHNE bit is 1 and the MRB.CHNS bit is 0, an interrupt request to the CPU is not generated when the specified number of data transfers is completed, or while the MRB.DISEL bit is 1 (an interrupt request to the CPU is generated for every data transfer). Data transfer has no effect on the interrupt status flag, which is the request source.

The transfer information (SAR, DAR, CRA, CRB, MRA, MRB, and MRC) that define a data transfer can be specified independently of each other. Figure 16.9 shows chain transfer operation.

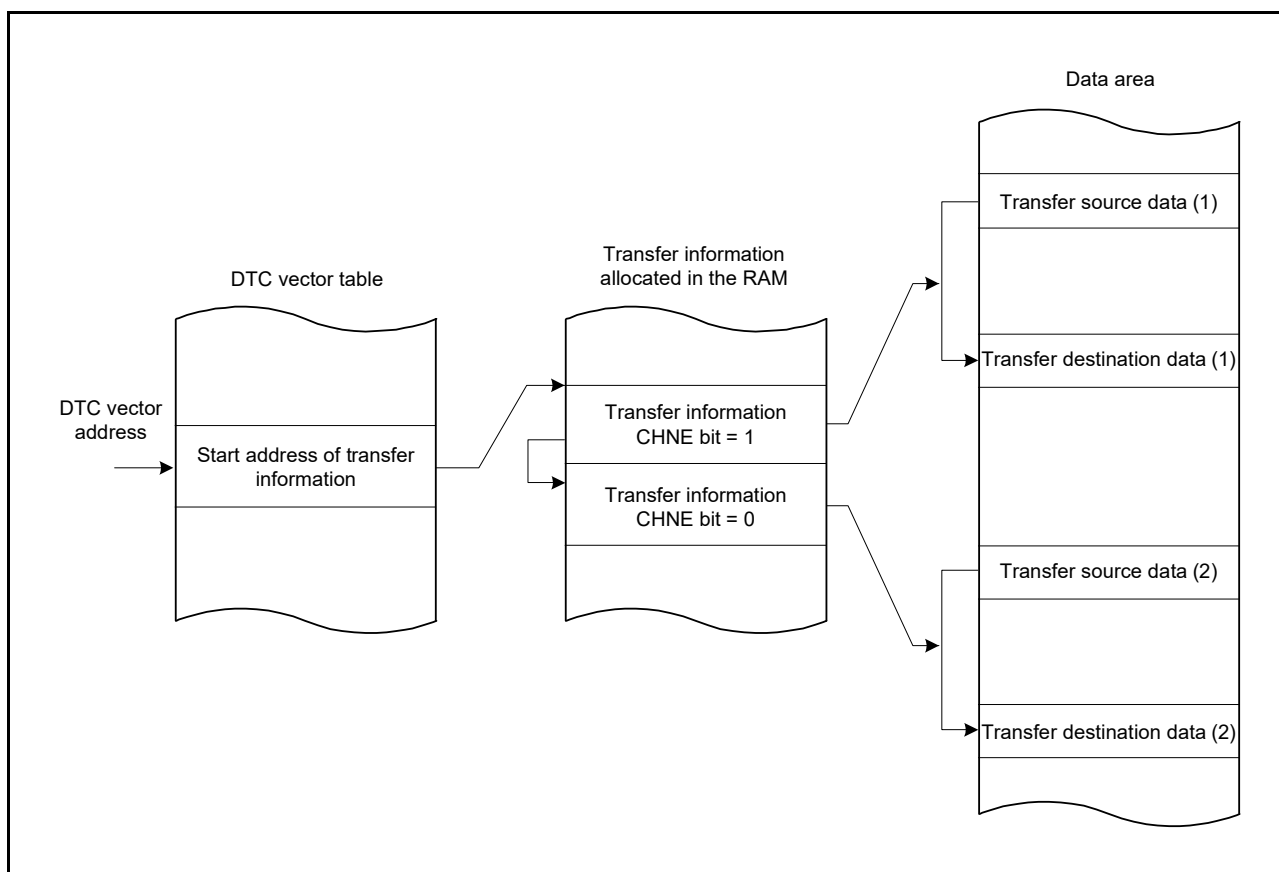


Figure 16.9 Chain Transfer Operation

If the MRB.CHNE bit is 1 and the CHNS bit is 1, chain transfer is performed only after completion of specified number of data transfers. In repeat transfer mode, chain transfer is performed after completion of specified number of data transfers.

For details on chain transfer conditions, refer to Table 16.4, Chain Transfer Conditions.

16.4.7 Operation Timing

Figure 16.10 to Figure 16.14 show examples of DTC operation timing.

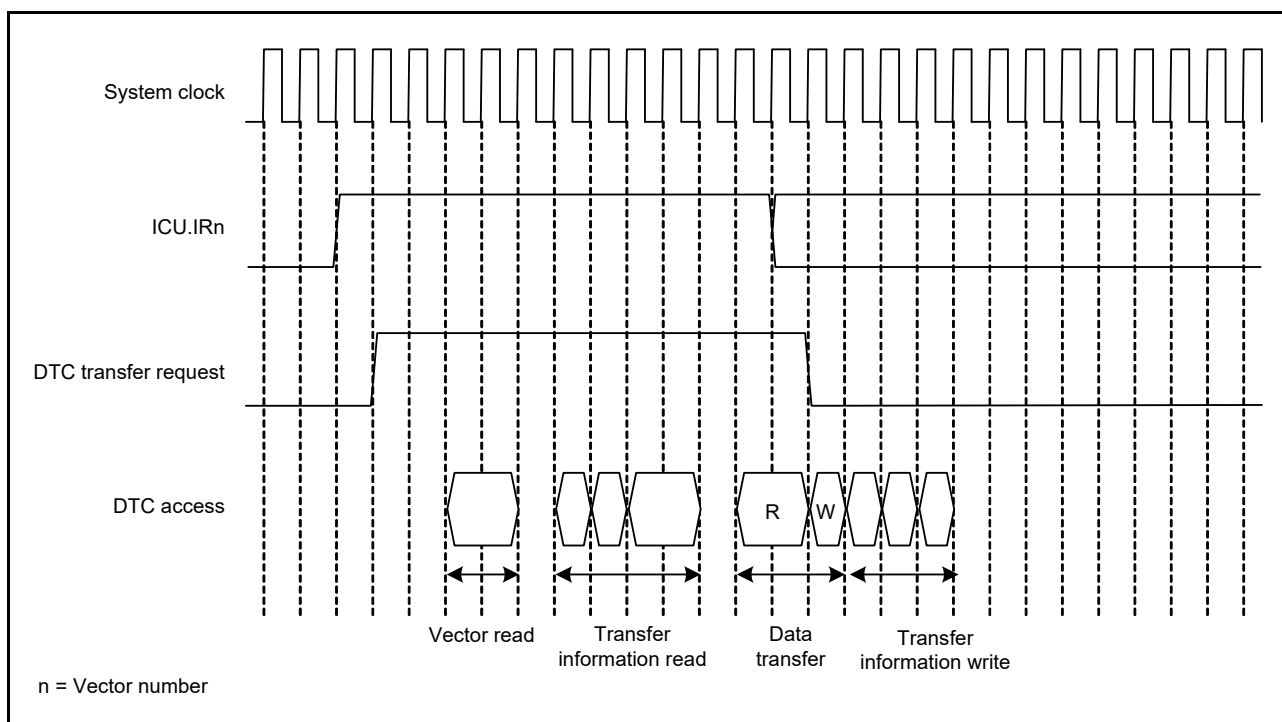


Figure 16.10 Example (1) of DTC Operation Timing
(Short-Address Mode, Normal Transfer Mode, Repeat Transfer Mode)

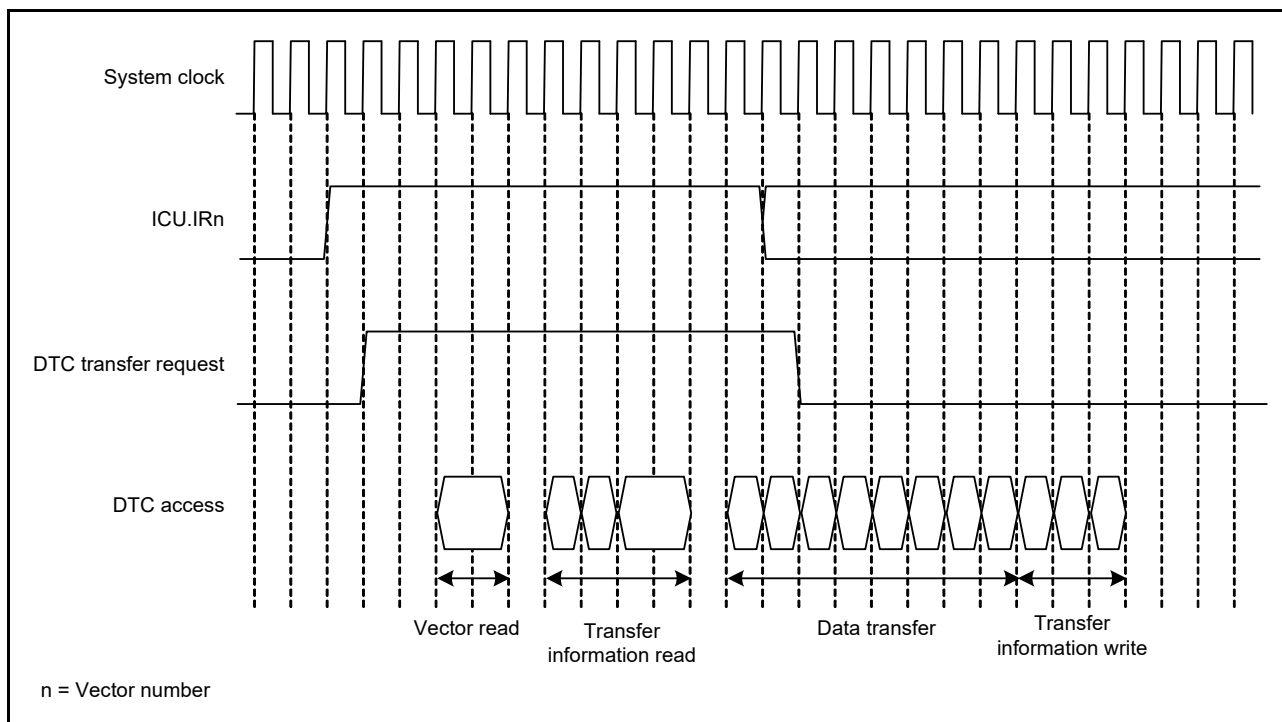


Figure 16.11 Example (2) of DTC Operation Timing
(Short-Address Mode, Block Transfer Mode, Block Size = 4)

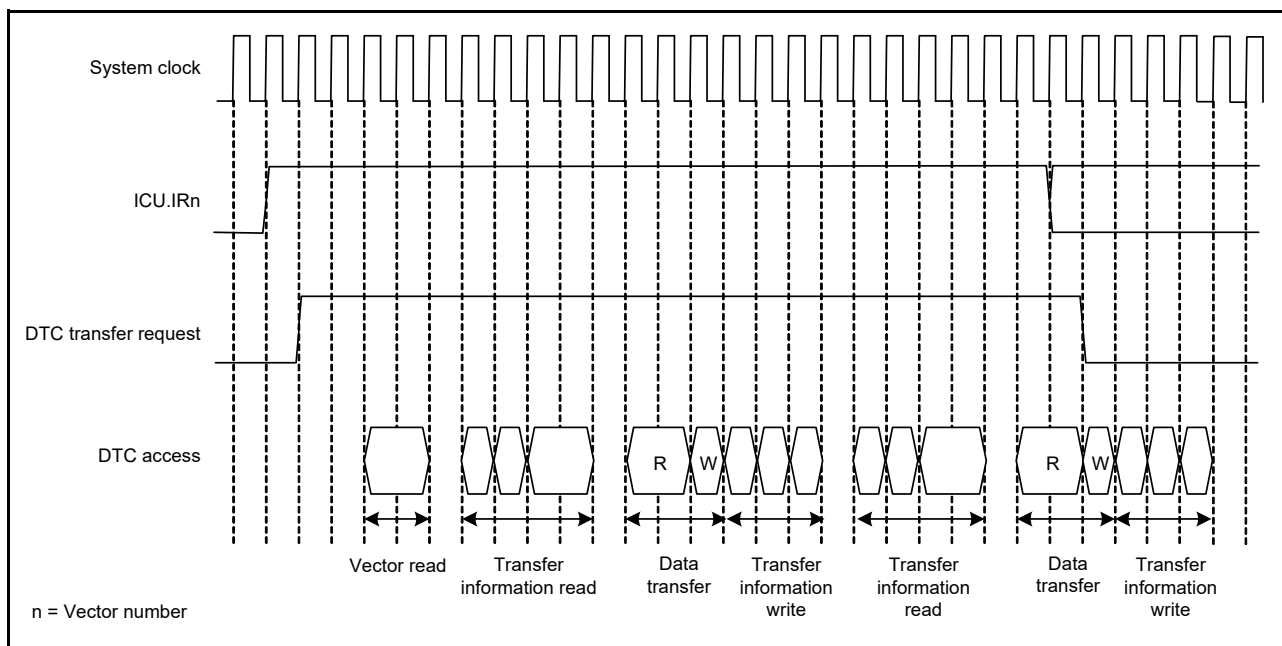


Figure 16.12 Example (3) of DTC Operation Timing (Short-Address Mode, Chain Transfer)

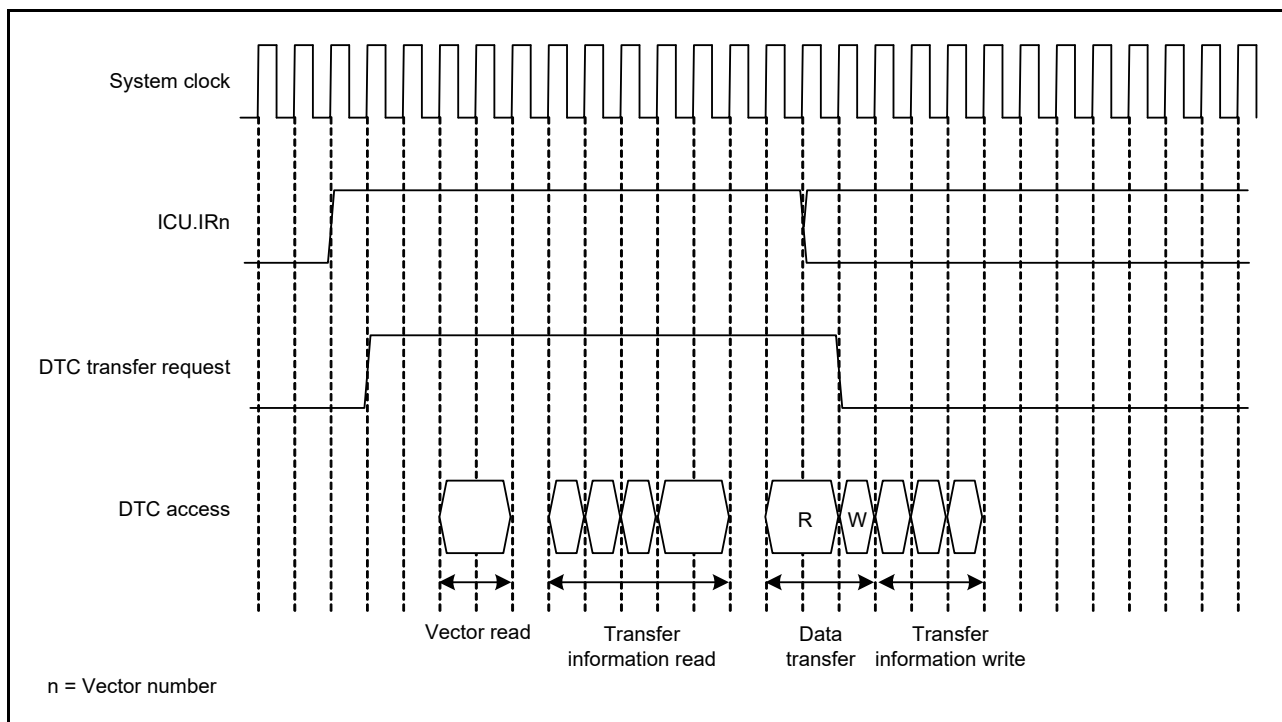


Figure 16.13 Example (4) of DTC Operation Timing (Full-Address Mode, Normal Transfer Mode, Repeat Transfer Mode)

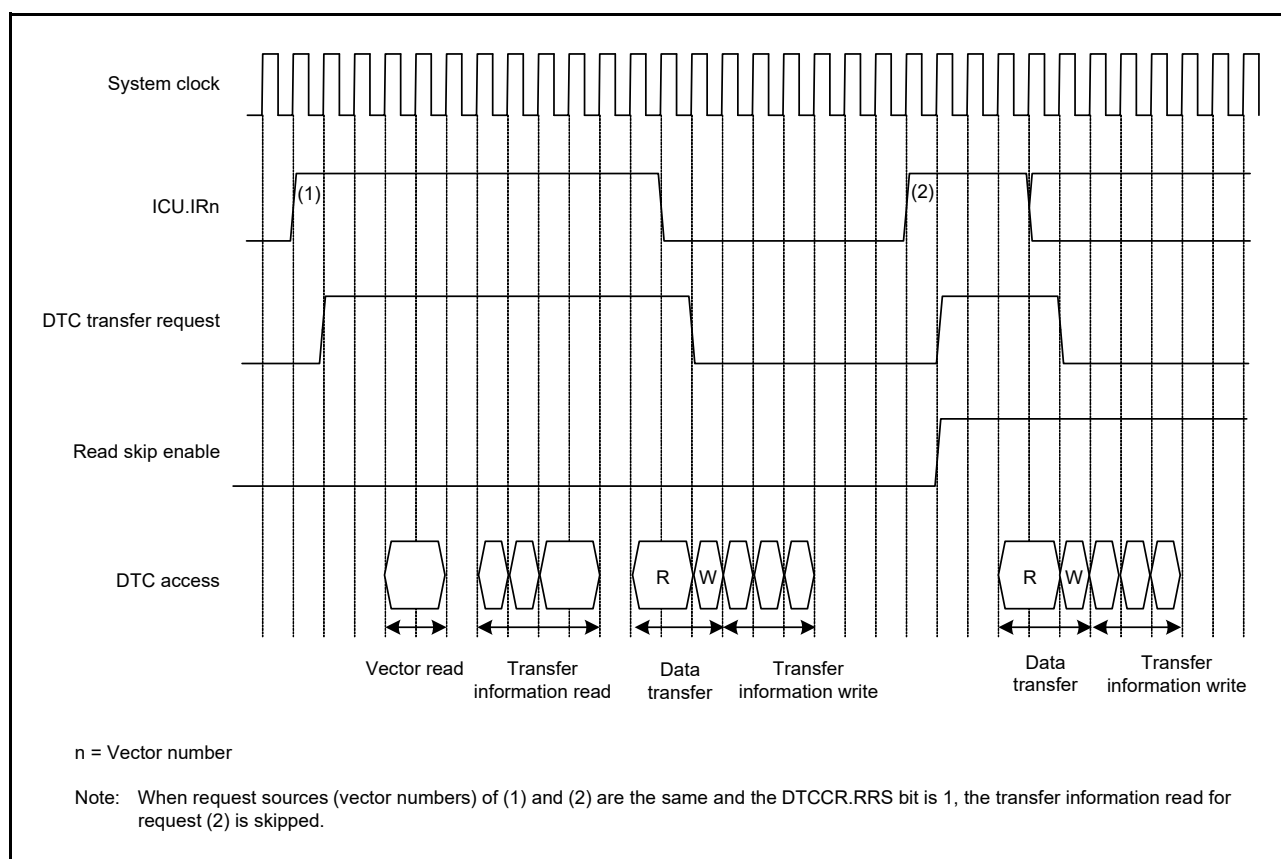


Figure 16.14 Example of Operation When Transfer Information Read Skip is Executed (Vector, Transfer Information, and Transfer Destination Data on the RAM, and Transfer Source Data on the Peripheral Module)

16.4.8 Execution Cycles of the DTC

Table 16.9 lists the execution cycles of single data transfer of the DTC.

For the order of the execution states, refer to section 16.4.7, Operation Timing.

Table 16.9 Execution Cycles of the DTC

Transfer Mode	Vector Read		Transfer Information Read			Transfer Information Write			Data Transfer		Internal Operation	
									Read	Write		
Normal	$C_v + 1$	0^{*1}	$4 \times C_i + 1^{*2}$	$3 \times C_i + 1^{*3}$	0^{*1}	$3 \times C_i^{*4}$	$2 \times C_i^{*5}$	C_i^{*6}	$C_r + 1$	C_w	2	0^{*1}
Repeat									$C_r + 1$	C_w		
Block ^{*7}									$P \times C_r$	$P \times C_w$		

Note 1. When transfer information read is skipped

Note 2. In full-address mode

Note 3. In short-address mode

Note 4. When neither SAR nor DAR is set to address-fixed

Note 5. When SAR or DAR is set to address-fixed

Note 6. When SAR and DAR are set to address-fixed

Note 7. When the block size is 2 or more. If the block size is 1, the cycle number for normal transfer is applied.

P: Block size (initial settings of CRAH and CRAL)

C_v : Cycles for access to vector transfer information storage destination

C_i : Cycles for access to transfer information storage destination address

C_r : Cycles for access to data read destination

C_w : Cycles for access to data write destination

(The unit is system clocks (ICLK) for "+ 1" in the Vector Read, Transfer Information Read, and Data Transfer Read columns and "2" in the Internal Operation column.)

(C_v , C_i , C_r , and C_w vary depending on the corresponding access destination. For the number of cycles for respective access destinations, refer to section 36, RAM, section 37, Flash Memory (FLASH), and section 5, I/O Registers.)

16.4.9 DTC Bus Mastership Release Timing

The DTC does not release the bus mastership during transfer information read and transfer information write. While transfer information is not read or written, bus arbitration is made according to the priority determined by the bus master arbitrator.

For bus arbitration, refer to section 15, Buses.

16.4.10 Sequence Transfer

A sequence transfer can be executed on the request source that is specified in the DTCSQE register. A sequence transfer is started by setting the MRB.INDX bit to 1 and ended by setting the MRB.SQEND bit to 1. Setting the DTCOR.SQTFRL bit to 1, even during a sequence transfer, forcibly ends the transfer and the next DTC transfer request starts new sequence transfer with reference to the index table.

A sequence transfer includes the following processing.

- (1) The first data transfer is executed by referring to the DTC vector table in response to a DTC transfer request from the source specified in the DTCSQE register.
- (2) The DTC index table is referred based on the value of the lower 8 bits of the first data transferred data in (1) (sequence number).
- (3) The transfer information is read from the address obtained from the DTC index table.
- (4) A data transfer is executed in accordance with the transfer information. On completion of the data transfer, either one of the following operations is performed by using the values of bits MRB.CHNE and MRB.SQEND.
 - A chain transfer is executed when the CHNE bit is 1. The next transfer information is read. Go to (4).
 - The sequence transfer is suspended when the CHNE bit is 0 and the SQEND bit is 0. Go to (5).
 - The sequence transfer ends when the CHNE bit is 0 and the SQEND bit is 1.
- (5) When a DTC transfer request from the source specified in the DTCSQE register is accepted, the suspended sequence transfer is resumed and the next transfer information is read. Go to (4).

Note 1. When the ICU.DTCERn.DTCE bit becomes 0 based on the result of the data transfer, a DTC transfer request is not generated. Set the DTCE bit to 1 to resume sequence transfer. Refer to Figure 16.5 or section 14, Interrupt Controller (ICUb) for the conditions where the DTCE bit becomes 1.

Figure 16.15 and Figure 16.16 shows a basic sequence transfer operation.

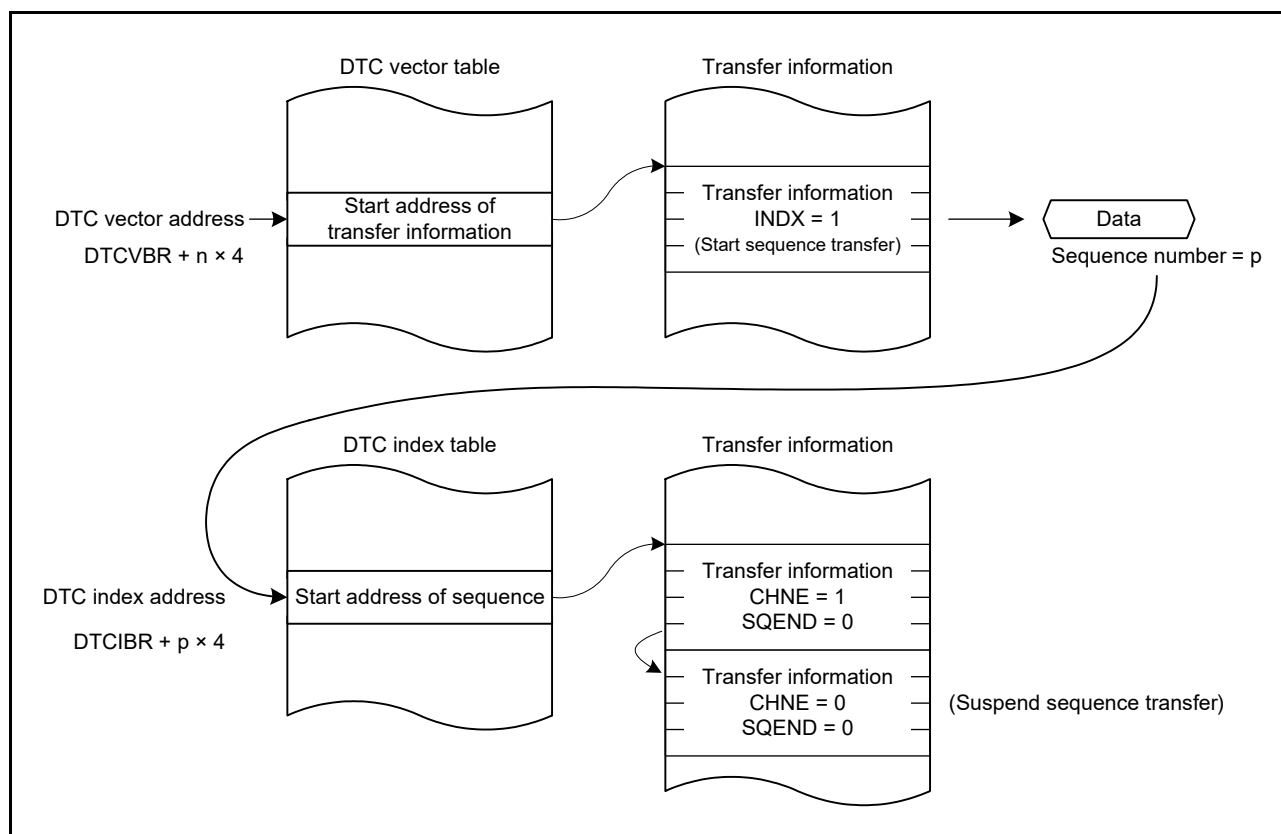


Figure 16.15 Start and Suspension of Sequence Transfer

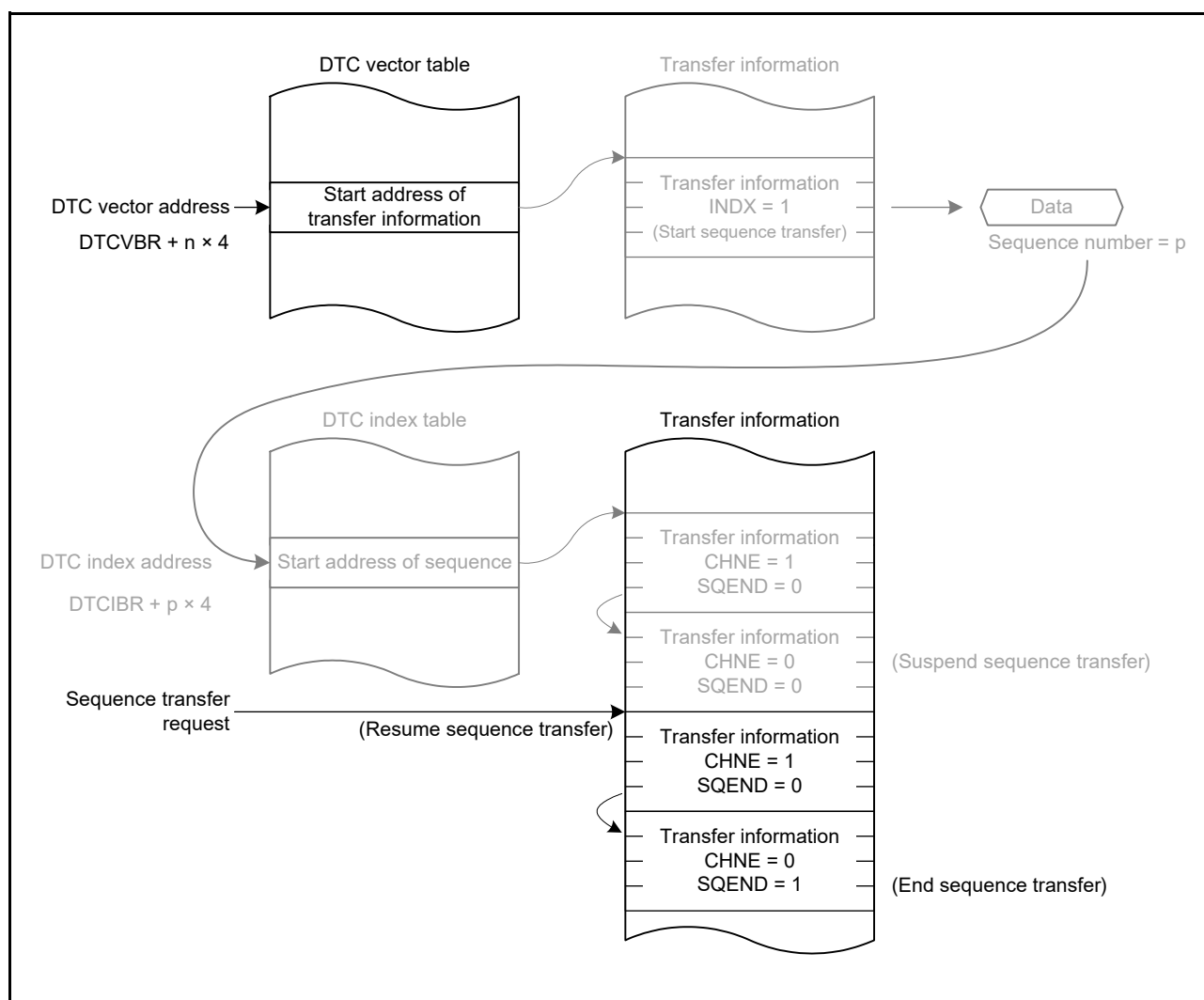


Figure 16.16 Resumption and End of Sequence Transfer

Table 16.10 lists the settings of bits CHNE, SQEND, and INDX during the sequence transfer.

Table 16.10 Sequence Transfer Process and Values of Bits CHNE, SQEND, and INDX

DTC Operations	CHNE Bit	SQEND Bit	INDX Bit
Start sequence transfer	0	0	1*1
Continue sequence transfer	1	0	0
Suspend sequence transfer*2	0	0	0
End sequence transfer	0	1	0
End current sequence transfer and Obtain new sequence number	0	1	1*1
Some other transfer (not sequence transfer)	—	0	0

Note: Do not set the values other than listed above.

Note 1. Set MRA.MD[1:0] bits to 00b (normal transfer mode) when setting the INDX bit to 1.

Note 2. When a sequence transfer is suspended, the ICU.DTCERn.DTCE bit may become 0. Set the DTCE bit to 1 to resume sequence transfer.

Even when a sequence transfer is suspended, a new sequence transfer cannot start until the suspended sequence transfer is eventually completed. When a sequence transfer request is received during suspension of the sequence transfer, the suspended sequence transfer is resumed.

16.4.11 DTC Index Table

The DTC index table is allocated to the area where its start address is configured in the DTCIBR register.

Store the start address of transfer information table p for sequence number p in the address of $\text{DTCIBR} + p \times 4$.

The upper 30 bits of the start address is set to the upper 30 bits of the DTC index. Set the CPUSEL bit to select either of reading the transfer information and starting the sequence, or output an interrupt request to the CPU without starting the sequence. For a complicated sequence that the DTC cannot handle, set the CPUSEL bit to 1 to allow the CPU to handle such a sequence.

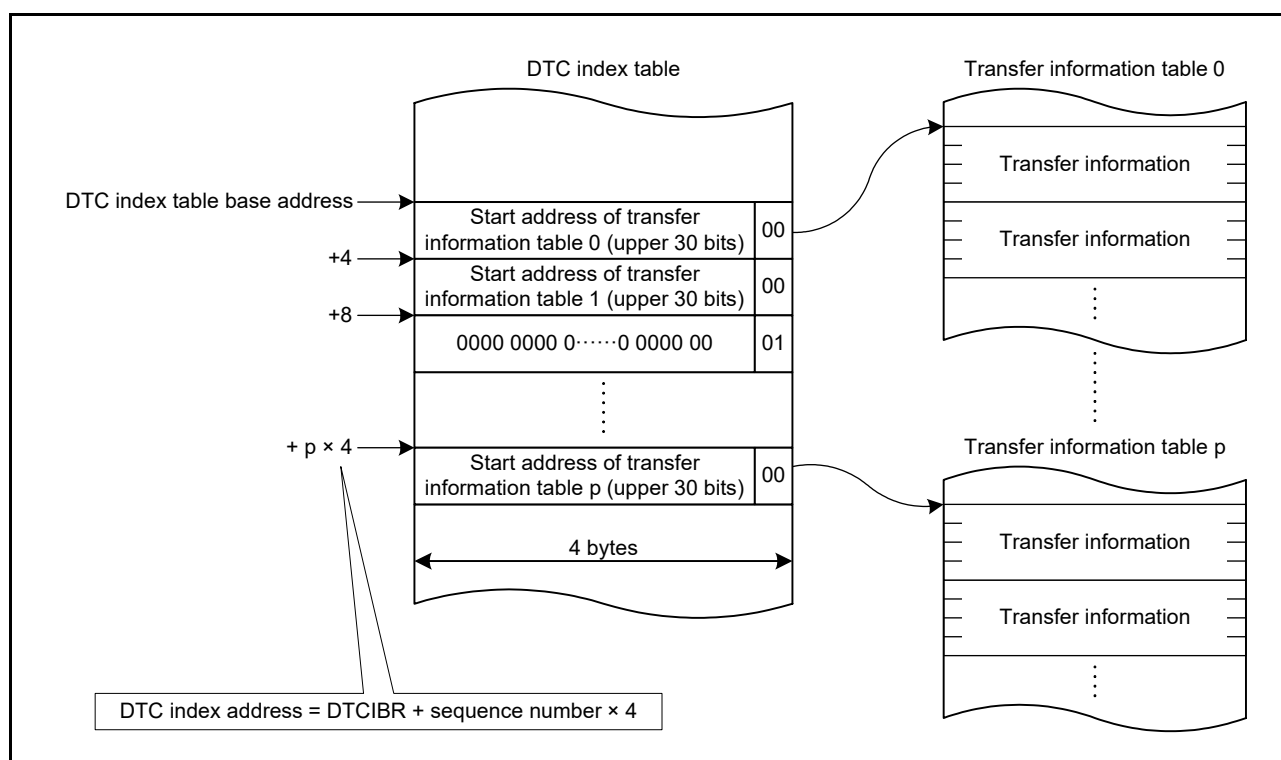
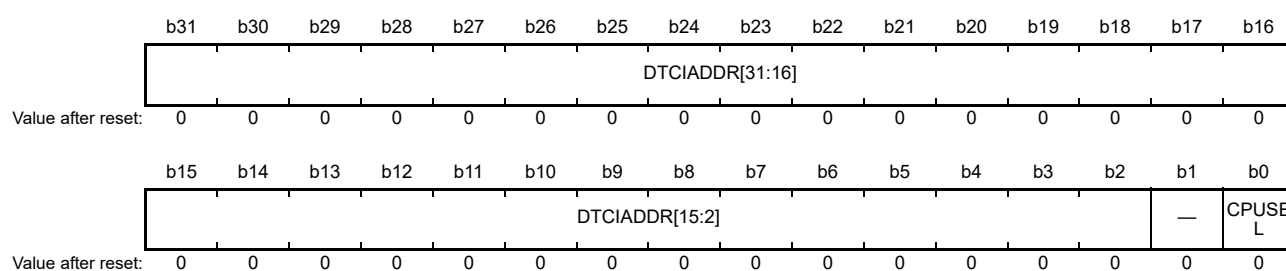


Figure 16.17 DTC Index Table

- DTC Index

Address(es): DTCIBR + p × 4



Bit	Symbol	Bit Name	Description	R/W
b0	CPUSEL	Sequence Transfer/CPU Interrupt Select	0: Continues the sequence transfer (starts the sequence) 1: Ends the sequence transfer and outputs an interrupt request to the CPU	—
b1	—	Reserved	Set this bit to 0.	—
b31 to b2	DTCIADDR[31:2]	Transfer Information Table Address	Set the upper 30 bits of the start address of the transfer information table to these bits. Writing to the upper 4 bits (b31 to b28) is ignored and the values in b31 to b28 become the same value as b27.	—

When the CPUSEL bit in the DTC index that the obtained sequence number indicates is 1, an interrupt request to the CPU is generated. At this time, the ICU.DTCERn.DTCE bit becomes 0. From this point, the interrupt request signal from the request source that is specified in the DTCSQE register is sent to the CPU, but not DTC. After completion of CPU interrupt processing, set the ICU.DTCERn.DTCE bit to 1 to enable DTC transfer request for starting the next sequence transfer.

16.4.12 Example of Sequence Transfer

Figure 16.18 shows a typical examples of a sequence transfer and Figure 16.19 to Figure 16.23 show configurations of the transfer information for the examples of the transfers in the figure.

In these examples, the interrupt source of vector number n is set as the source of the sequence transfer (DTCSQE.VECN[7:0] = n).

Once the DTC transfer request due to the interrupt source of vector number n (hereinafter referred to as “transfer request n ”) is input, the DTC refers to the DTC vector table and reads the corresponding transfer information. The lower 8 bits that have been transferred based on the transfer information become a sequence number, selecting one sequence among possible 256 sequences.

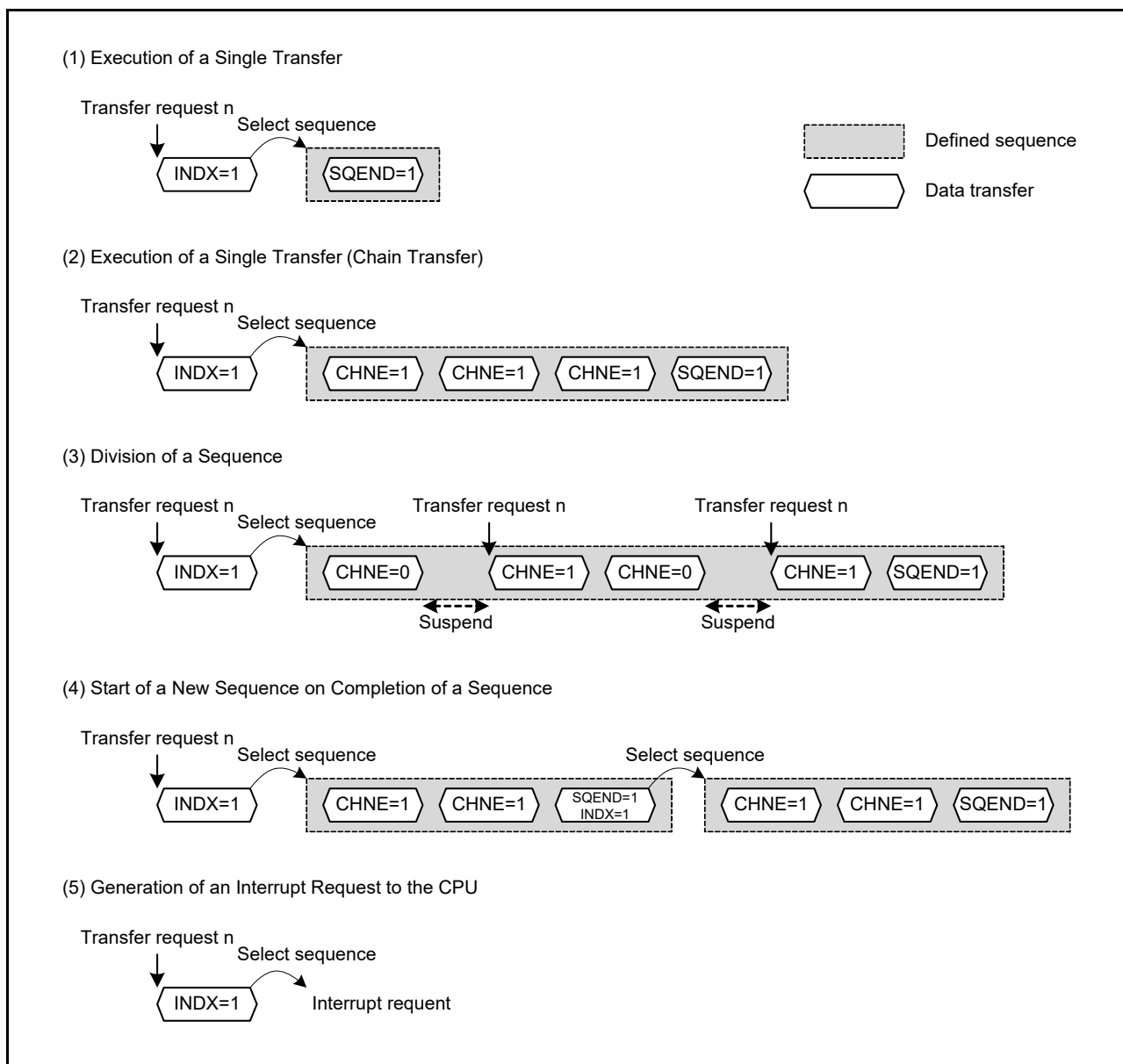


Figure 16.18 Examples of Sequence Transfers

(1) When Executing a Single Transfer

Figure 16.19 shows an example of a single transfer (normal, repeat, or block).

The DTC refers to the DTC index table, and reads the transfer information corresponding to the obtained sequence number p.

Since the values of the CHNE, INDX, and SQEND bits are 0, 0, and 1 respectively, the sequence ends after the specified transfer is executed.

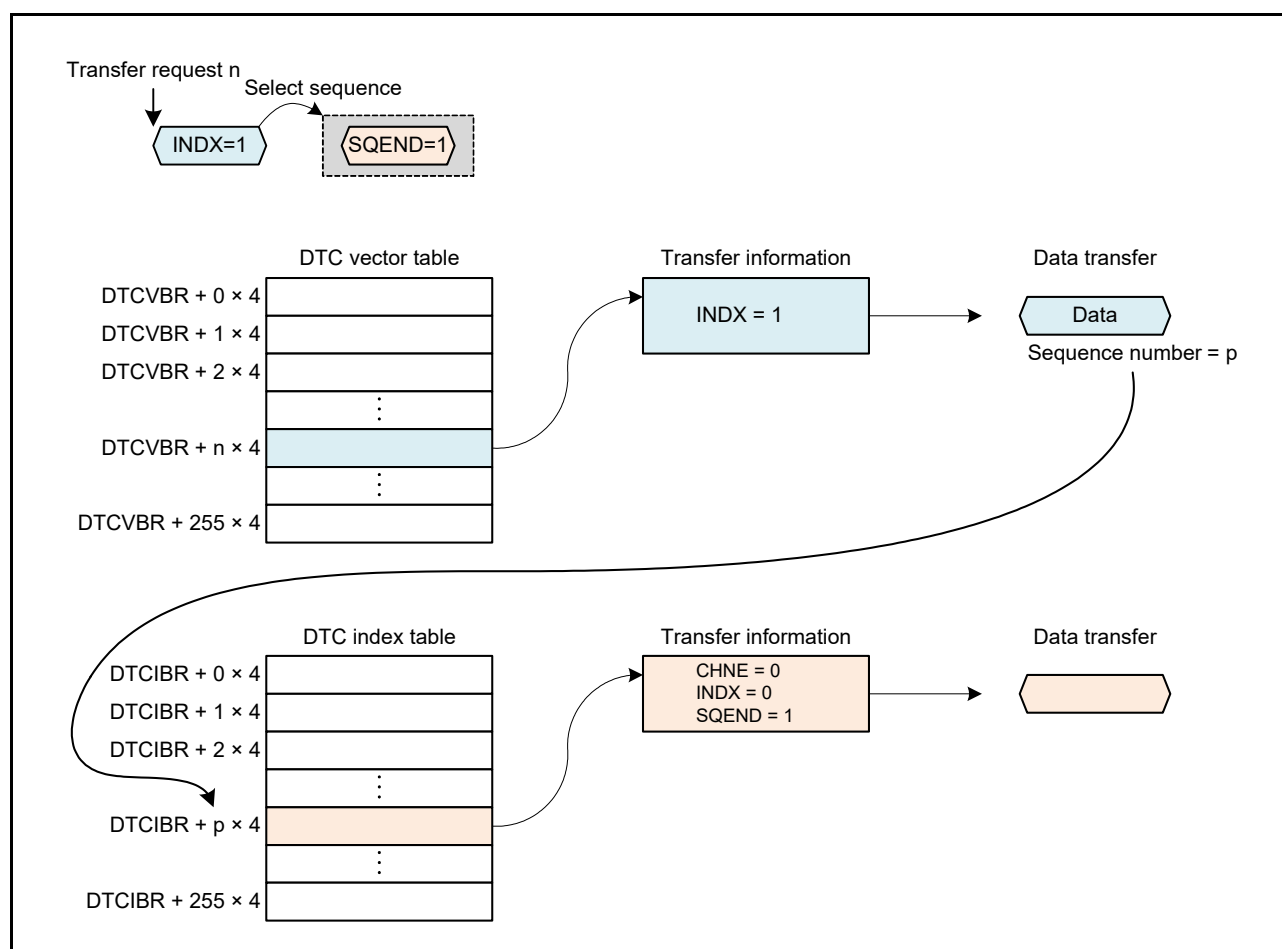


Figure 16.19 Example of a Sequence of Single Transfer

(2) When Executing a Single Chain Transfer

Figure 16.20 is an example of a sequence for a single chain transfer.

The DTC refers to the DTC index table, and reads the transfer information corresponding to the obtained sequence number q .

While the values of the CHNE, INDX, and SQEND bits are 1, 0, and 0 respectively, the specified chain transfer is executed. When the transfer information in which the CHNE, INDX, and SQEND bits are 0, 0, and 1 respectively is read, the sequence ends after the specified transfer is executed.

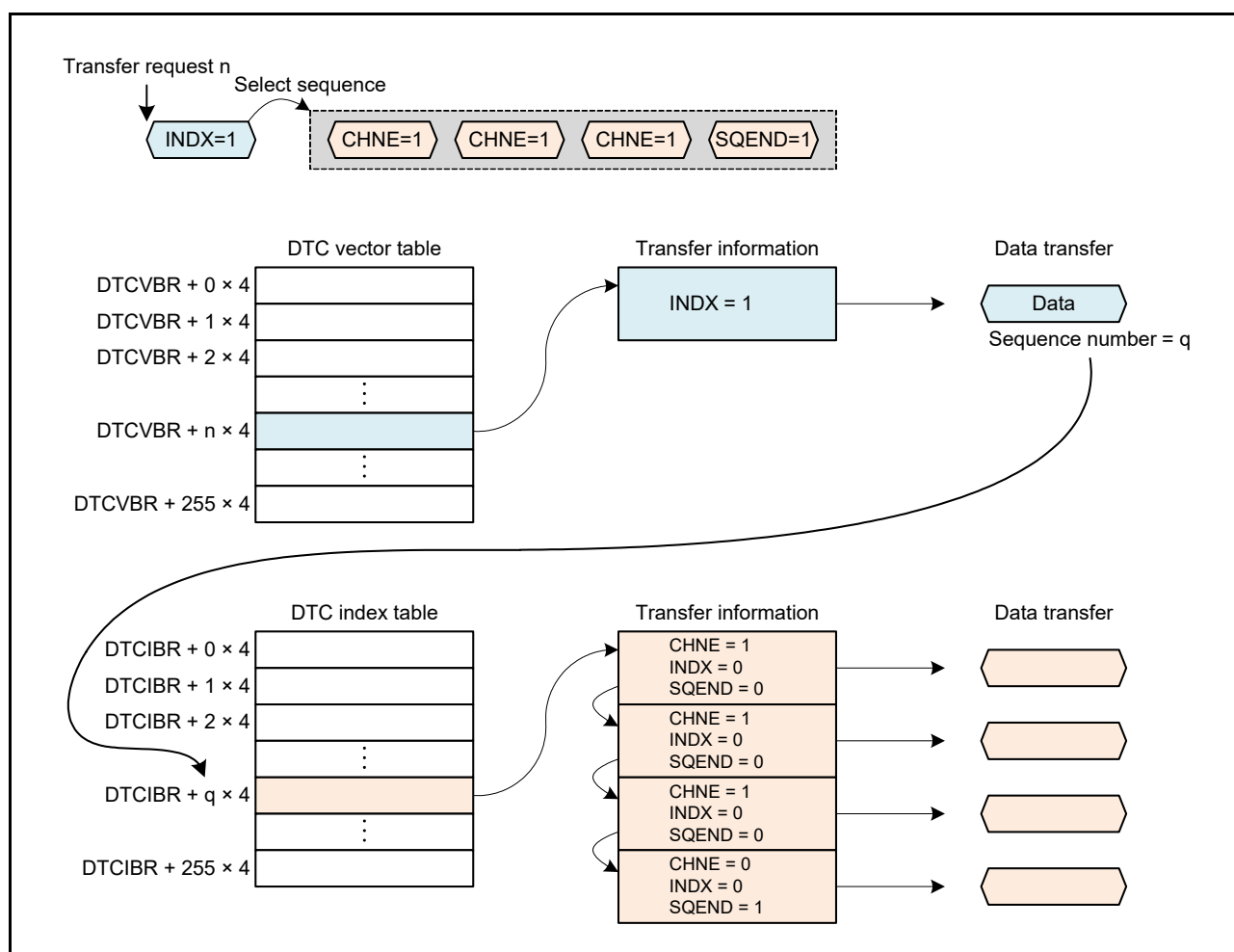


Figure 16.20 Example of Sequence of a Single Chain Transfer

(3) When Dividing a Sequence

Figure 16.21 is an example of the sequence that is divided into 3 parts.

The DTC refers to the DTC index table, and reads the transfer information corresponding to the obtained sequence number r .

Since the values of the CHNE, INDX, and SQEND bits in the transfer information are 0, 0, and 0 respectively, the sequence is suspended after the specified transfer is executed and the DTC waits for the next transfer request n .

When the transfer request n is input during a sequence transfer, the DTC vector table is not referred and the suspended sequence is resumed.

When the transfer information in which the CHNE, INDX, and SQEND bits are 0, 0, and 1 respectively is read, the sequence ends after the specified transfer is executed.

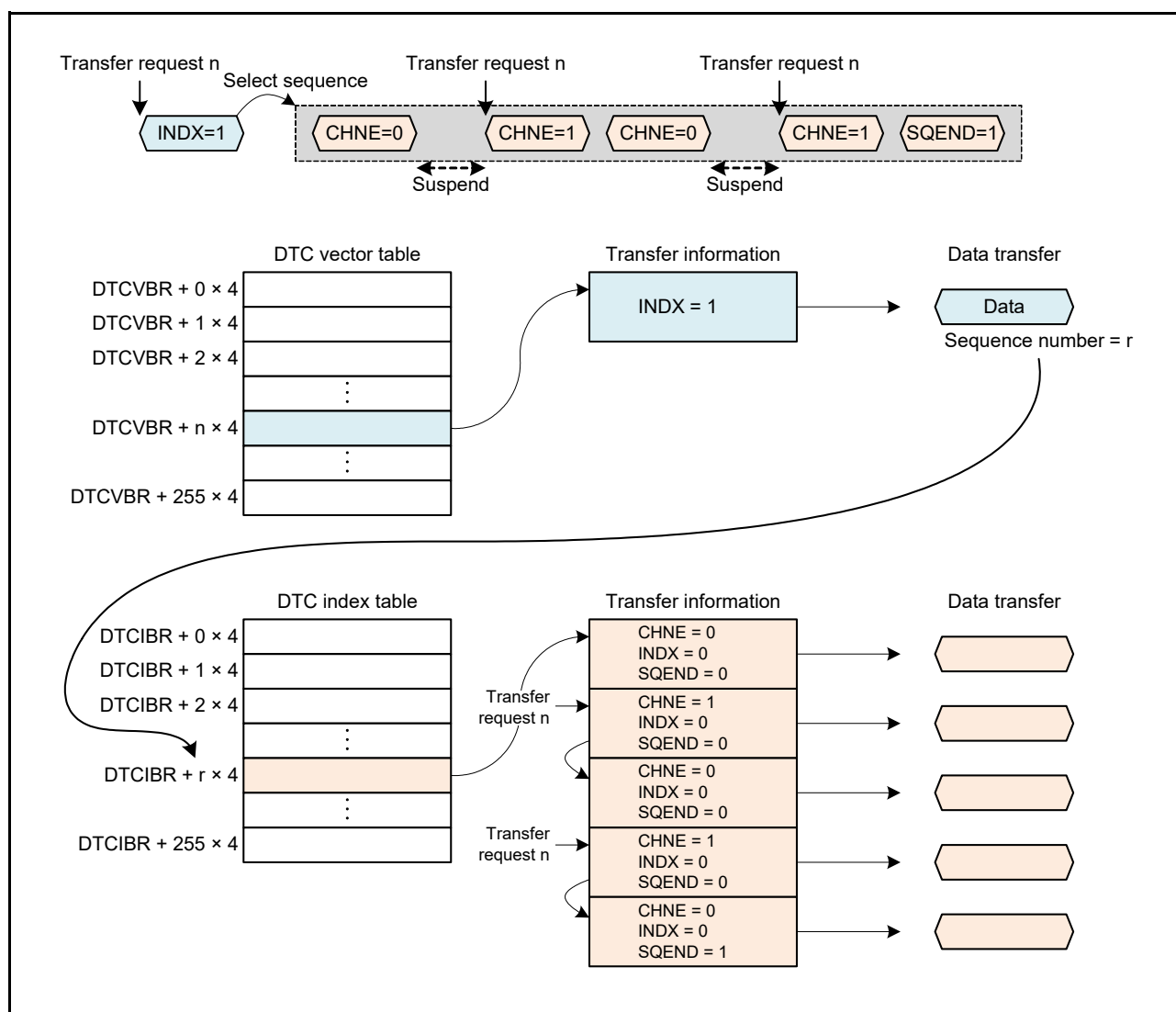


Figure 16.21 Example of Divided Sequence

(4) When Starting a New Sequence on completion of a Sequence

Figure 16.22 is an example for starting the next and new sequence on completion of the first sequence.

The DTC refers to the DTC index table, and reads the transfer information corresponding to the obtained sequence number s .

When the transfer information in which the CHNE, INDX, and SQEND bits are 0, 1, and 1 respectively is read, the specified transfer is executed, then a new sequence number is obtained from the lower 8 bits of the transferred data.

The DTC again refers to the DTC index table, and reads the transfer information corresponding to the obtained sequence number k and then starts a new sequence.

When the transfer information in which the CHNE, INDX, and SQEND bits are 0, 0, and 1 respectively is read, the sequence ends after the specified transfer is executed.

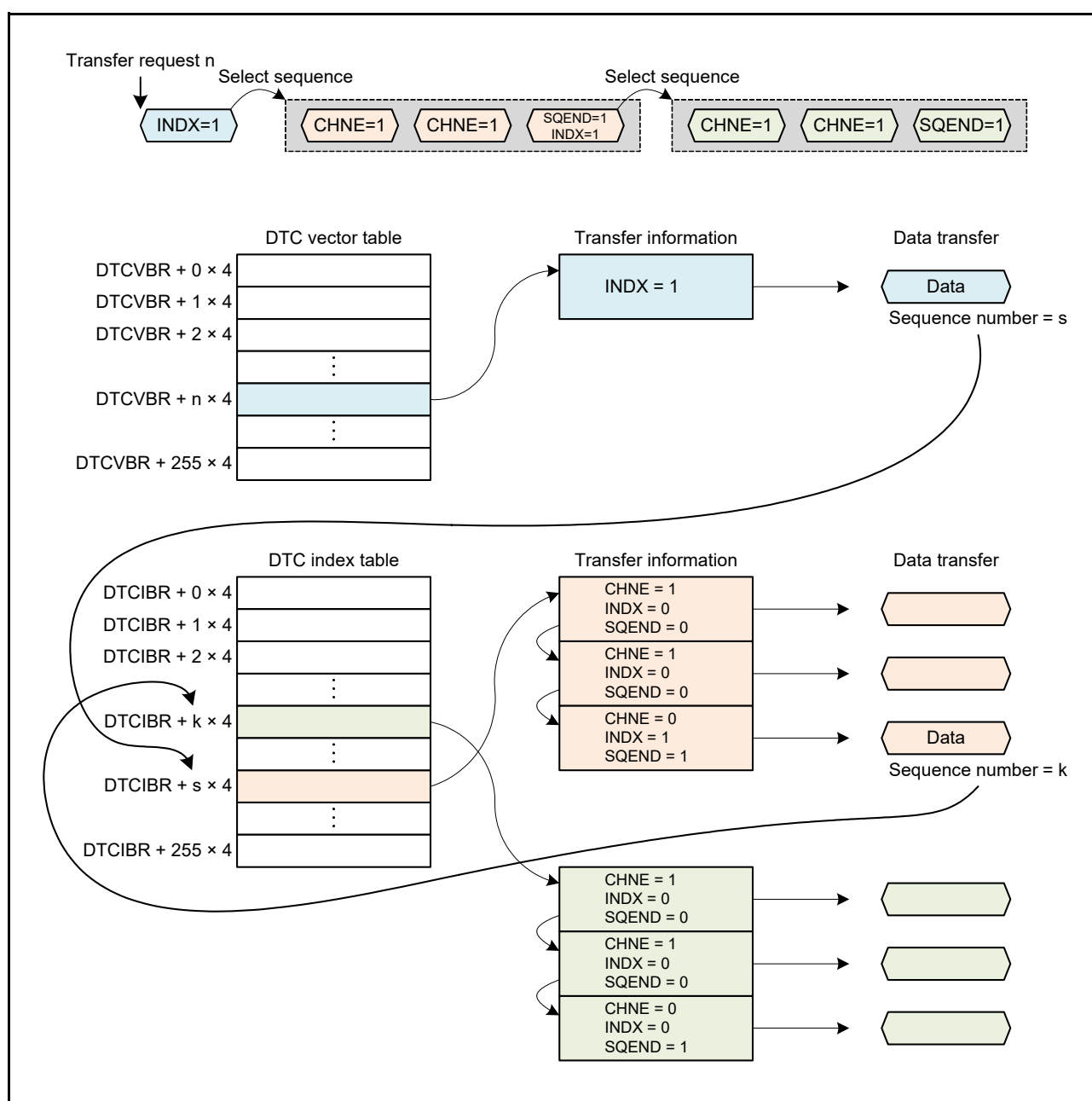


Figure 16.22 Example When Starting a New Sequence on Completion of a Sequence

(5) When Generating an Interrupt Request to the CPU

Figure 16.23 is an example of that an interrupt request is output to the CPU without starting of sequence.

The DTC obtains a DTC index that corresponds to the obtained sequence number t .

When the CPUSEL bit of the obtained DTC index is 1, the DTC ends the sequence transfer without starting the sequence, and then outputs an interrupt request to the CPU.

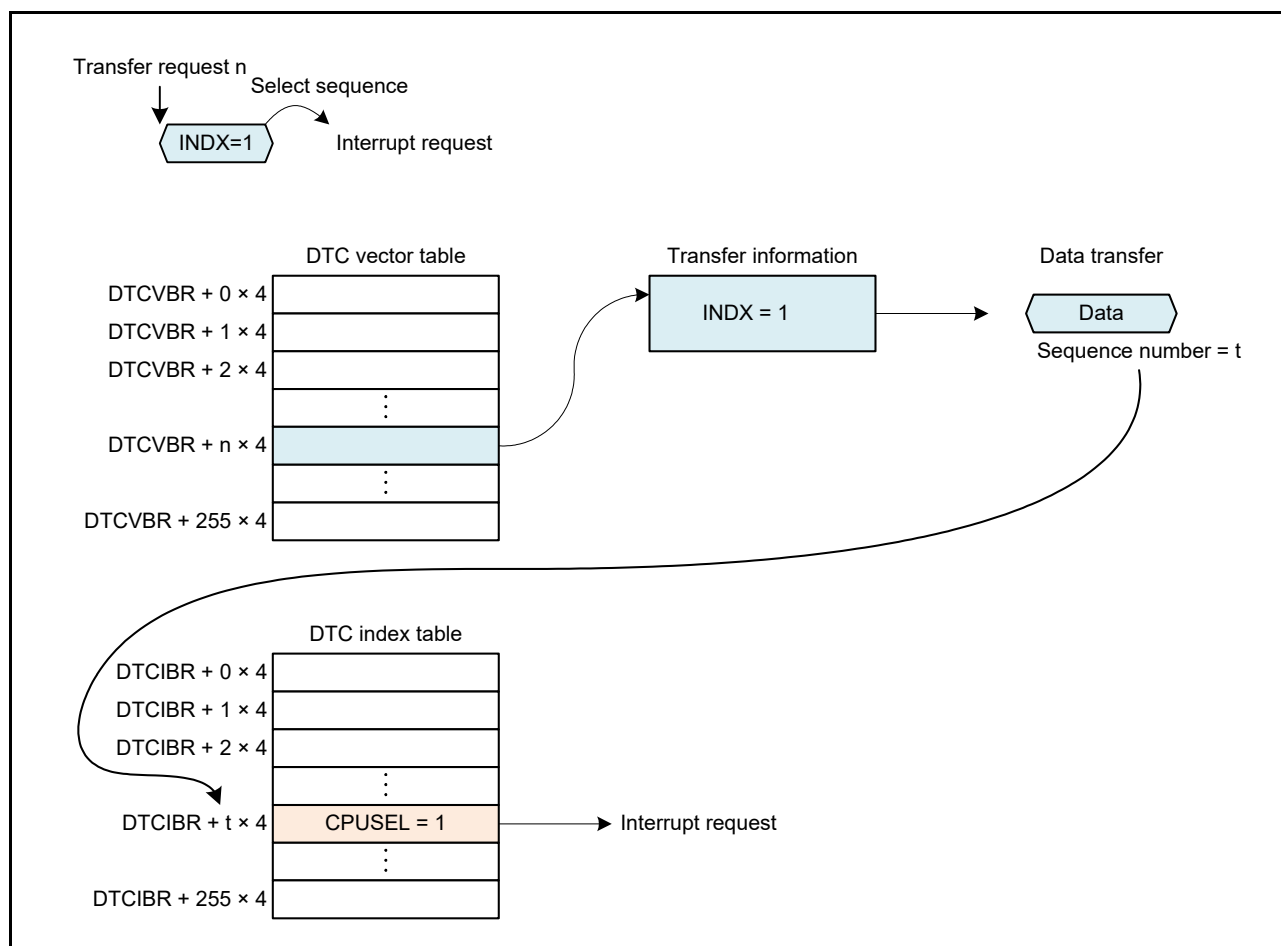


Figure 16.23 Example of Output of an Interrupt Request to the CPU

16.5 DTC Setting Procedure

Before using the DTC, set the DTC vector base register (DTCVBR). When using sequence transfer, also set the DTC index table base register (DTCIBR).

Figure 16.24 shows the procedure to set the DTC.

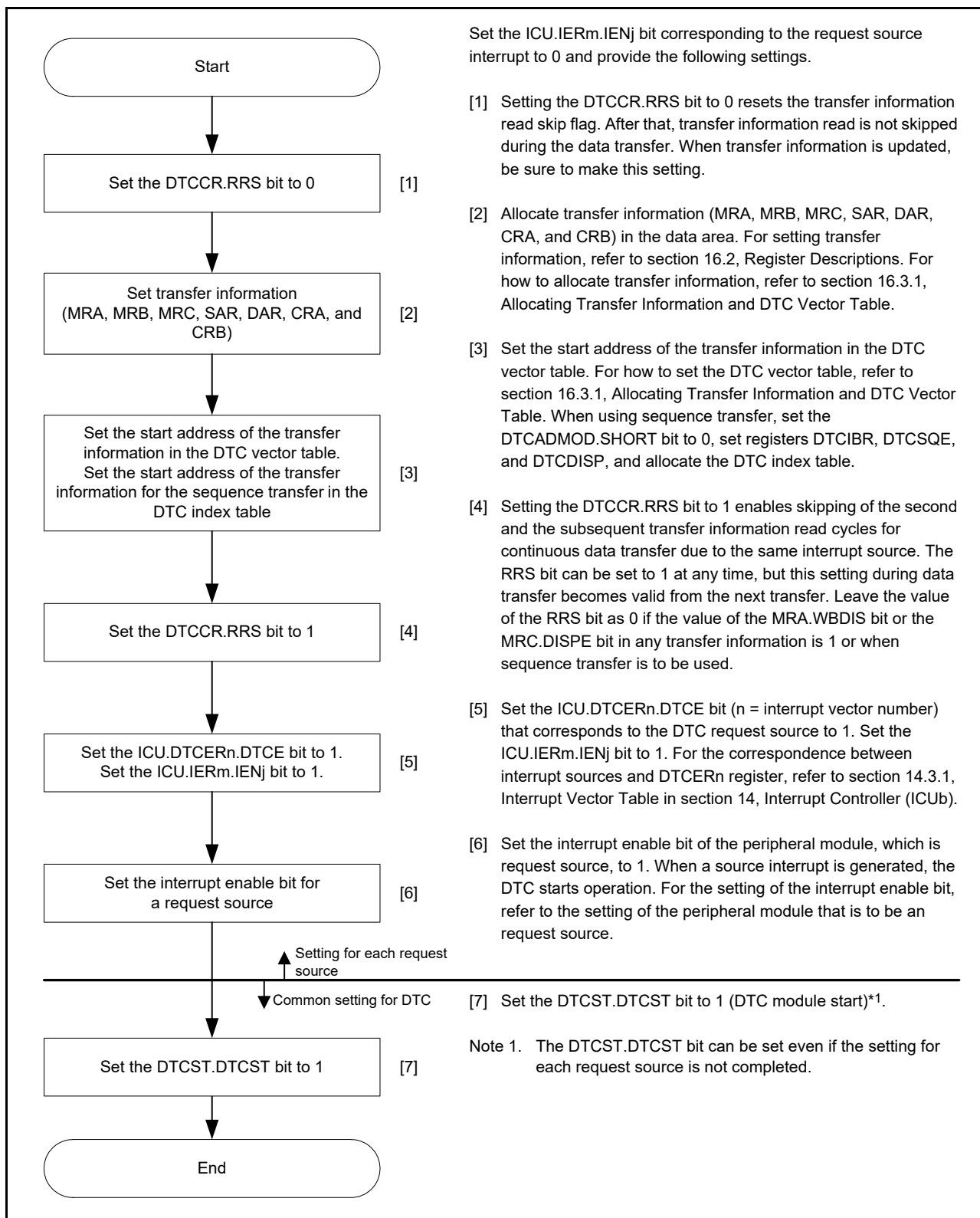


Figure 16.24 Procedure to Set the DTC

16.6 Examples of DTC Usage

16.6.1 Normal Transfer

As an example of DTC usage, its employment in the reception of 128 bytes of data by an SCI is described below.

(1) Transfer Information Setting

Set the MRA.MD[1:0] bits to 00b (normal transfer mode), the MRA.SZ[1:0] bits to 00b (byte transfer), and the MRA.SM[1:0] bits to 00b (source address is fixed). Set the MRB.CHNE bit to 0 (chain transfer is disabled), the MRB.DISEL bit to 0 (an interrupt request to the CPU is generated on completion of the specified number of data transfers), and the MRB.DM[1:0] bits to 10b (DAR is incremented after data transfer). The MRB.DTS bit can be set to any value. Set the RDR register address of the SCI in the SAR register, the start address of the RAM area for data storage in the DAR register, and 128 (0080h) in the CRA register. The CRB register can be set to any value.

(2) DTC Vector Table Setting

The start address of the transfer information for the RXI interrupt is set in the vector table for the DTC.

(3) ICU Setting and DTC Module Activation

Set the corresponding ICU.DTCERn.DTCE bit to 1 and the ICU.IERm.IENj bit to 1.
Set the DTCST.DTCST bit to 1.

(4) SCI Setting

Enable the RXI interrupt by setting the SCR.RIE bit in the SCI to 1. If a reception error occurs during the SCI receive operation, further reception is not performed. Accordingly, make settings so that the CPU can accept receive error interrupts.

(5) DTC Transfer

Every time the reception of 1 byte by the SCI is completed, an RXI interrupt is generated to start the data transfer. The DTC transfers the received byte from the RDR of the SCI to RAM, after which the DAR register is incremented and the CRA register is decremented.

(6) Interrupt Handling

After 128 times of data transfers have been completed and the value in the CRA register becomes 0, an RXI interrupt request is output to the CPU. Complete the process in the handling routine for this interrupt.

16.6.2 Chain Transfer When the Counter is 0

The second data transfer is performed only when the transfer counter is set to 0 in the first data transfer, and the first data transfer information is repeatedly changed in the second data transfer. Repeating this chain transfer enables transfers to be repeated more than 256 times.

The following shows an example of configuring a 128-Kbyte input buffer to addresses 0020 0000h to 0021 FFFFh (where the input buffer is set so that its lower address starts with 0000h). Figure 16.25 shows a chain transfer when the counter is 0.

- (1) Set normal transfer mode for input data for the first data transfer. Set the following:
Transfer source address: Fixed, the CRA register is 0000h (65,536 times), the MRB.CHNE bit is 1 (chain transfer is enabled), the MRB.CHNS bit is 1 (chain transfer is performed only when the transfer counter becomes 0), and the MRB.DISEL bit is 0 (an interrupt request to the CPU is generated on completion of the specified number of data transfers).
- (2) Prepare the upper 8 bits (in this case, 21h and 20h) of the start address at every 65,536 times of the transfer destination address for the first data transfer in another area (such as ROM).
- (3) For the second data transfer, set repeat transfer mode (source is repeat area) for rewriting the transfer destination address of the first data transfer. The transfer destination is the address where the upper 8 bits of the DAR register in the first transfer information is allocated. In this case, set the MRB.CHNE bit to 0 (chain transfer is disabled) and the MRB.DISEL bit to 0 (an interrupt request to the CPU is generated on completion of the specified number of data transfers). In this case, set the transfer counter to 2.
- (4) When a transfer request is accepted, the first data transfer is executed. When transfer is executed 65,536 times and the transfer counter of the first data transfer becomes 0, the second data transfer is started and the upper 8 bits of the transfer destination address of the first data transfer is set to 21h. At this time, the lower 16 bits of the transfer destination address and the transfer counter of the first data transfer have become 0000h.
- (5) In succession, when another transfer request is accepted, the first data transfer is executed. When transfer is executed 65,536 times and the transfer counter of the first data transfer becomes 0, the second data transfer is started and the upper 8 bits of the transfer destination address of the first data transfer is set to 20h. At this time, the lower 16 bits of the transfer destination address and the transfer counter of the first data transfer have become 0000h.
- (6) Steps (4) and (5) above are repeated infinitely. Because the second data transfer is in repeat transfer mode, no interrupt request to the CPU is generated.

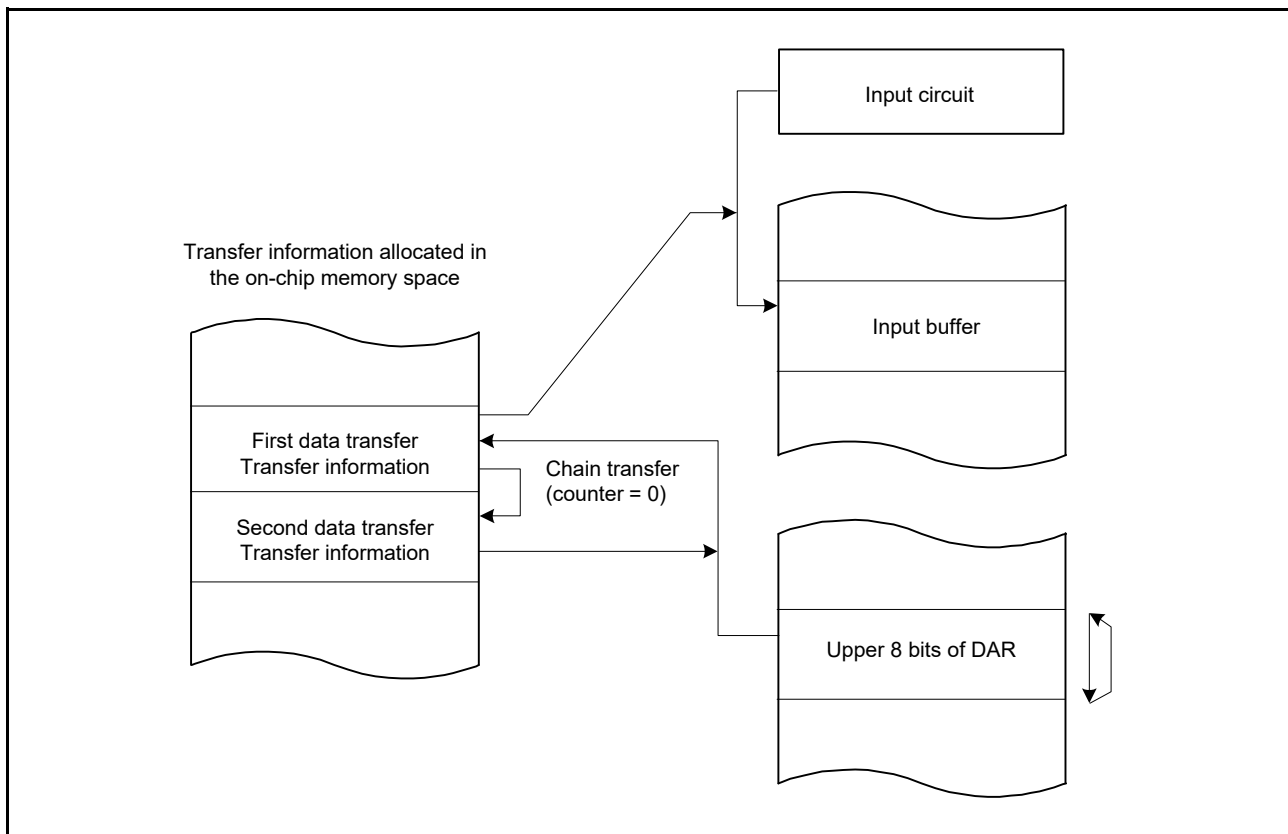


Figure 16.25 Chain Transfer When the Counter is 0

16.6.3 Sequence Transfer

The following is an example of using the SCI receive interrupt as a request source of sequence transfer.

(1) Transfer Information Settings

Set the MRA.MD[1:0] bits to 00b (normal transfer mode), the MRA.SZ[1:0] bits to 00b (byte transfer), and the MRA.SM[1:0] bits to 00b (source address is fixed). Set the MRB.CHNE bit to 0 (chain transfer is disabled), the MRB.DISEL bit to 0 (an interrupt request to the CPU is generated on completion of the specified number of data transfers), the MRB.DM[1:0] bits to 10b (DAR is incremented after data transfer), the MRB.INDX bit to 1 (start sequence transfer), and the MRB.SQEND bit to 0 (continue the sequence transfer). The MRB.DTS bit can be set to any value. Set the address of the SCIk.RDR register in the SAR register and set the start address of the RAM area which stores the data in the DAR register.

When the MRA.WBDIS bit is set to 1 (Does not write back the transfer information), the values of registers CRA and CRB are ignored.

(2) DTC Vector Table Setting

Set the start address of the transfer information for the corresponding receive data full interrupt (RXI) in the DTC vector table.

(3) DTC Index Table Setting

Set the start address of the transfer information for each sequence in the DTC index table.

(4) ICU Setting and DTC Module Activation

Set the corresponding ICU.DTCERn.DTCE bit to 1 and the ICU.IERm.IENj bit to 1. Set the DTCST.DTCST bit to 1.

(5) SCI Setting

Set the SCIk.SCR.RIE bit to 1 to enable the RXI interrupt. If a reception error occurs during the SCI receive operation, subsequent receptions are not performed. Accordingly, make settings so that the CPU can accept receive error interrupts.

(6) Start of the Sequence Transfer

On completion of reception of 1-byte data by the SCI, an RXI interrupt is generated to start the DTC. The DTC transfers the received data from the SCIk.RDR register to the RAM. The DTC looks up the DTC index table by using the value from the received data (sequence number) and continues to transfer data corresponding to the that number.

When the CPUSEL bit in the DTC index is 1, the DTC does not read the transfer information and sets the ICU.DTCERn.DTCE bit to 0. Then the DTC outputs an interrupt request to the CPU and ends the sequence transfer.

(7) During Suspension of the Sequence Transfer

Set the ICU.DTCERn.DTCE bit to 1 if the bit is 0. The DTC continues to transfer the data for every generation of the DTC transfer request in response to the corresponding RXI interrupt.

(8) End of the Sequence Transfer

Set the MRB.SQEND bit in the last transfer information of the sequence transfer to 1. After execution of this data transfer, the DTC ends the sequence transfer. The DTC starts to refer to the DTC vector table when a DTC transfer request is generated due to the next corresponding RXI interrupt.

16.7 Interrupt Source

When the DTC has finished data transfer of specified count or when data transfer with the MRB.DISEL bit set to 1 (an interrupt request to the CPU is generated each time the data transfer is performed) has been completed, an interrupt to the CPU is generated by the DTC trigger source. Such interrupts to the CPU are controlled according to the PSW.I bit (interrupt enable) of the CPU, the PSW.IPL[3:0] bits (processor interrupt priority level), and the priority level of the interrupt controller.

16.8 Low Power Consumption Function

Before making a transition to the module stop state, deep sleep mode, or software standby mode, set the DTCST.DTCST bit to 0 (DTC module stop), and then perform the following.

(1) Module Stop Function

Writing 1 (transition to the module-stop state is made) to the MSTPCRA.MSTPA28 bit enables the module stop function of the DTC. If data transfer is in progress at the time 1 is written to the MSTPCRA.MSTPA28 bit, the transition to the module stop state proceeds after data transfer has ended. While the MSTPCRA.MSTPA28 bit is 1, accessing the DTC registers is prohibited.

Writing 0 (release from the module-stop state) to the MSTPCRA.MSTPA28 bit releases the DTC from the module stop state.

(2) Deep Sleep Mode

Make settings according to the procedure under section 11.6.2.1, Entry to Deep Sleep Mode, in section 11, Low Power Consumption.

If any data transfer is in progress at the time the WAIT instruction is executed, the transition to deep sleep mode follows the completion of the data transfer.

The DTC is released from the module stop state by writing 0 to the MSTPCRA.MSTPA28 bit following recovery from deep sleep mode.

(3) Software Standby Mode

Make settings according to the procedure under section 11.6.3.1, Entry to Software Standby Mode in section 11, Low Power Consumption.

If any data transfer is in progress at the time the WAIT instruction is executed, the transition to software standby mode follows the completion of the data transfer.

(4) Notes on Low Power Consumption Function

For the WAIT instruction and the register setting procedure, refer to section 11.7.5, Timing of WAIT Instructions in section 11, Low Power Consumption.

To perform data transfer after returning from a low power consumption mode, set the DTCST.DTCST bit to 1 again.

To use a request that is generated in deep sleep mode or software standby mode as an interrupt request to the CPU but not as a DTC transfer request, specify the CPU as the interrupt request destination according to the description in section 14.4.3, Selecting Interrupt Request Destinations in section 14, Interrupt Controller (ICUb), and then execute the WAIT instruction.

16.9 Usage Notes

16.9.1 Start Address of Transfer Information

Set multiples of 4 for the start addresses of the transfer information to be specified in the DTC vector table. If any value other than a multiple of 4 is specified, access still proceeds with the lower 2 bits of the address regarded as 00b.

16.9.2 Allocating Transfer Information

Allocate transfer information in the memory area according to the endian of the area as shown in Figure 16.26.

For example, when writing CRA and CRB settings in 16-bit units in big endian, write the CRA setting to the address plus 8h (Ch) and the CRB setting to the address plus Ah (Eh). In little endian, write the CRB setting to the address plus 8h (Ch) and the CRA setting to the address plus Ah (Eh). When writing CRA and CRB settings in 32-bit units, allocate the CRA setting at the MSB side of the 32 bits and the CRB setting at the LSB side, and write the settings to the address plus 8h (Ch), regardless of endian.

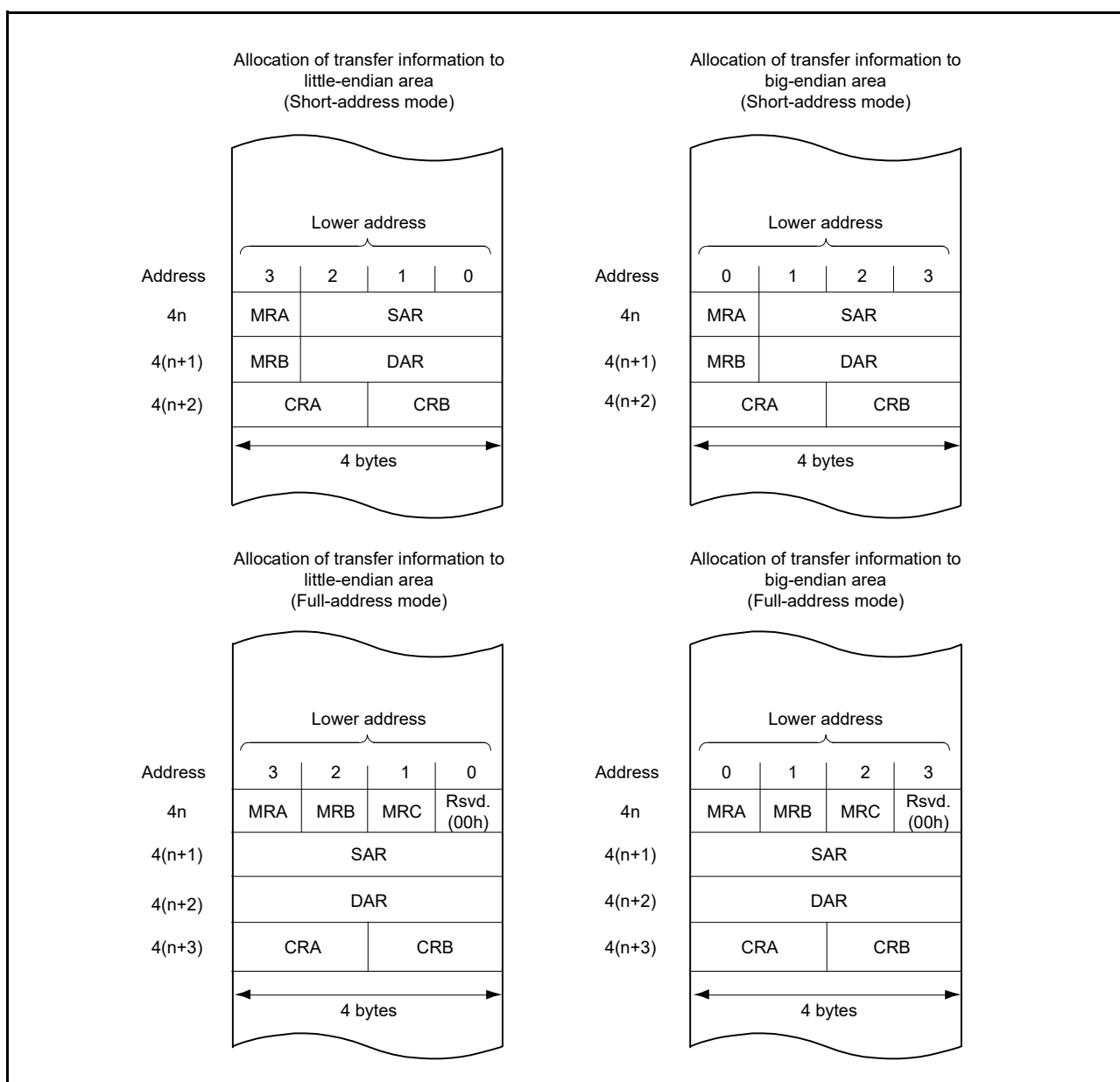


Figure 16.26 Allocation of Transfer Information

16.9.3 Notes on Using the Sequence Transfer

When sequence transfer is to be used, make sure that the DTCADMOD.SHORT bit is 0 (full-address mode) and the DTCCR.RRS bit is also 0 (transfer information read is not skipped).

In addition, set the MRB.CHNE bit to 0 (chain transfer is disabled) when setting the MRB.INDX bit to 1 (start sequence transfer and refer the index table) or the MRB.SQEND bit to 1 (end the sequence transfer).

17. I/O Ports

17.1 Overview

The I/O ports function as a general I/O port, an I/O pin of a peripheral module, an input pin for an interrupt.

Some of the pins are also configurable as an I/O pin of a peripheral module or an input pin for an interrupt. All pins function as input pins immediately after a reset, and pin functions are switched by register settings. The setting of each pin is specified by the registers for the corresponding I/O port and on-chip peripheral modules.

Each port has the port direction register (PDR) that selects input or output direction, the port output data register (PODR) that holds data for output, the port input data register (PIDR) that indicates the pin states, the open drain control register y (ODRy, y = 0, 1) that selects the output type of each pin, the pull-up control register (PCR) that controls on/off of the input pull-up MOS, and the port mode register (PMR) that specifies the pin function of each port.

For details on the PMR register, see section 18, Multi-Function Pin Controller (MPC).

The configuration of the I/O ports differs depending on the package. Table 17.1 lists the specifications of I/O ports, and Table 17.2 list the port functions.

Table 17.1 Specifications of I/O Ports

Port	Package		Package		Package		Package		Package		Package	
	64 Pins	Number of Pin	52 Pins	Number of Pin	48 Pins	Number of Pin	44 Pins	Number of Pin	32 Pins	Number of Pin	24 Pins	Number of Pin
PORT0	P00 to P04	5	P02, P04	2	Not provided	0	Not provided	0	Not provided	0	Not provided	0
PORT1	P10 to P15	6	P10, P11, P13, P14	4	P10, P11, P13, P14	4	P10, P11, P13, P14	4	P11, P13, P14	3	P11, P13, P14	3
PORT2	P22 to P26	5	P22 to P24, P26	4	P22 to P24, P26	4	P22 to P24, P26	4	P26	1	P26	1
PORT3	P30 to P33, P36, P37	6	P31, P33, P36, P37	4	P31, P36, P37	3	P31, P36	2	P31, P36, P37	3	P31, P36	2
PORT4	P40 to P47	8	P40 to P47	8	P40 to P47	8	P40 to P44, P46, P47	7	P40 to P42, P44, P46	5	P40 to P42	3
PORT7	P70 to P76	7	P70 to P76	7	P70 to P76	7	P71 to P76	6	P71 to P76	6	P73 to P76	4
PORT9	P90 to P97	8	P92 to P97	6	P92 to P97	6	P92 to P97	6	P92, P93, P96, P97	4	P96, P97	2
PORTB	PB0 to PB7	8	PB0, PB1, PB3 to PB7	7	PB0, PB1, PB3 to PB6	6	PB0, PB1, PB3, PB5, PB6	5	PB0, PB1, PB5, PB6	4	PB0, PB5, PB6	3
PORTD	PD3 to PD7	5	PD3 to PD6	4	PD3 to PD6	4	PD3 to PD6	4	Not provided	0	Not provided	0
PORTE	PE2	1	PE2	1	PE2	1	PE2	1	PE2	1	PE2	1
PORTG	PG7	1	PG7	1	PG7	1	PG7	1	PG7	1	PG7	1
Total of Pins		60	Total of Pins	48	Total of Pins	44	Total of Pins	40	Total of Pins	28	Total of Pins	20

Table 17.2 Port Functions

Port	Pin	I/O	Input Pull-up	Open Drain Output	5-V Tolerant	I/O Level
PORT0	P00 to P04	I/O	✓	✓	—	VCC
PORT1	P10, P11	I/O	✓	✓	—	
	P12 to P15	I/O	✓	—	—	
PORT2	P22 to P26	I/O	✓	✓	—	
PORT3	P30 to P33, P36, P37	I/O	✓	✓	—	
PORT4	P40 to P47	I/O	✓	—	—	
PORT7	P70 to P76	I/O	✓	✓	—	
PORT9	P90 to P96	I/O	✓	✓	—	
	P97	I/O	✓	✓	✓	
PORTB	PB0	I/O	✓	✓	✓	
	PB1 to PB7	I/O	✓	✓	—	
PORTD	PD3 to PD7	I/O	✓	✓	—	
PORTE	PE2	Input	—	—	—	
PORTG	PG7	I/O	✓	✓	—	

✓: Supported

—: Unsupported

Specifying input pull-up, open-drain output, or 5-V tolerant is available for other signals on pins that also function as general I/O pins.

17.2 I/O Port Configuration

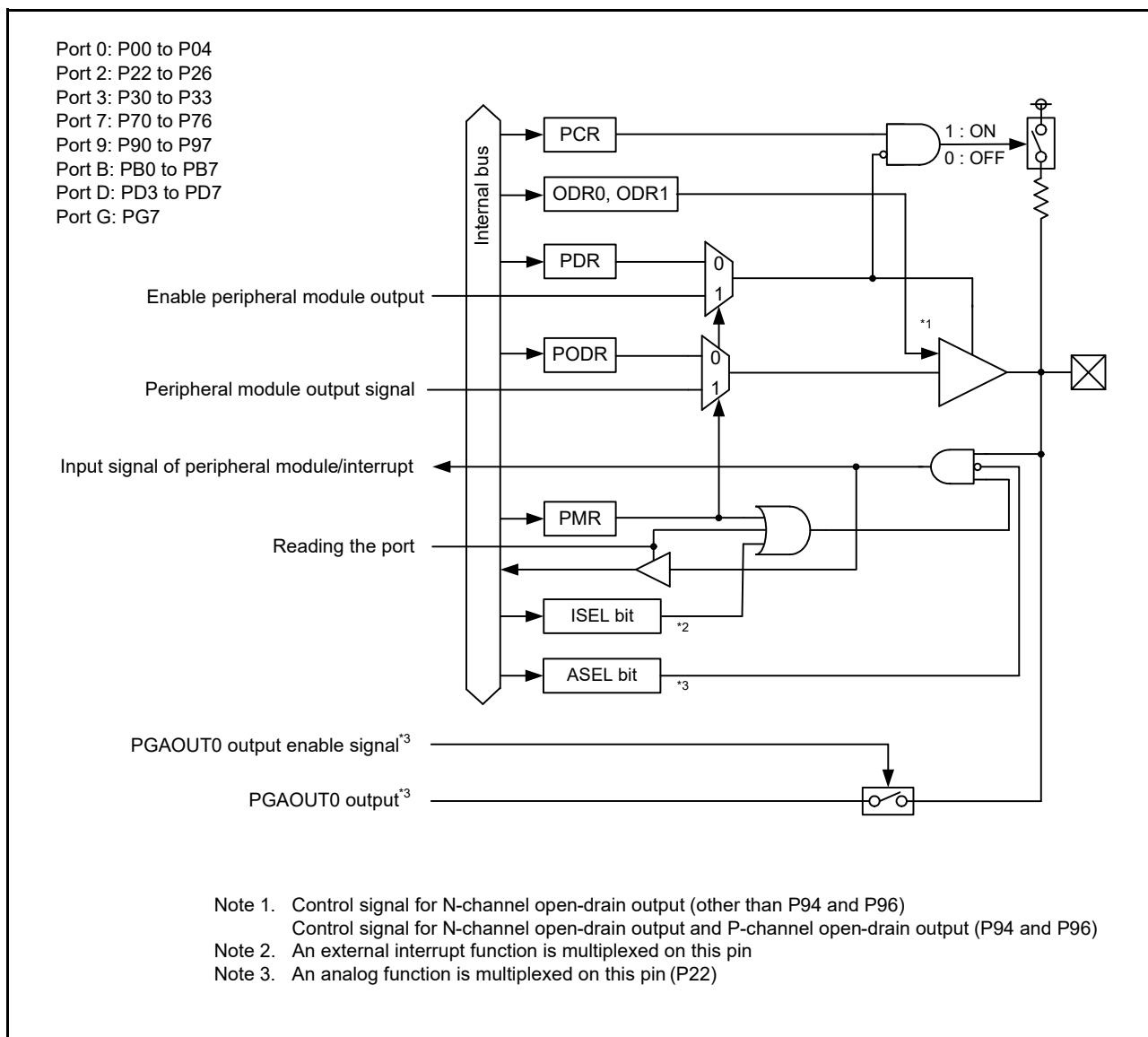


Figure 17.1 I/O Port Configuration (1)

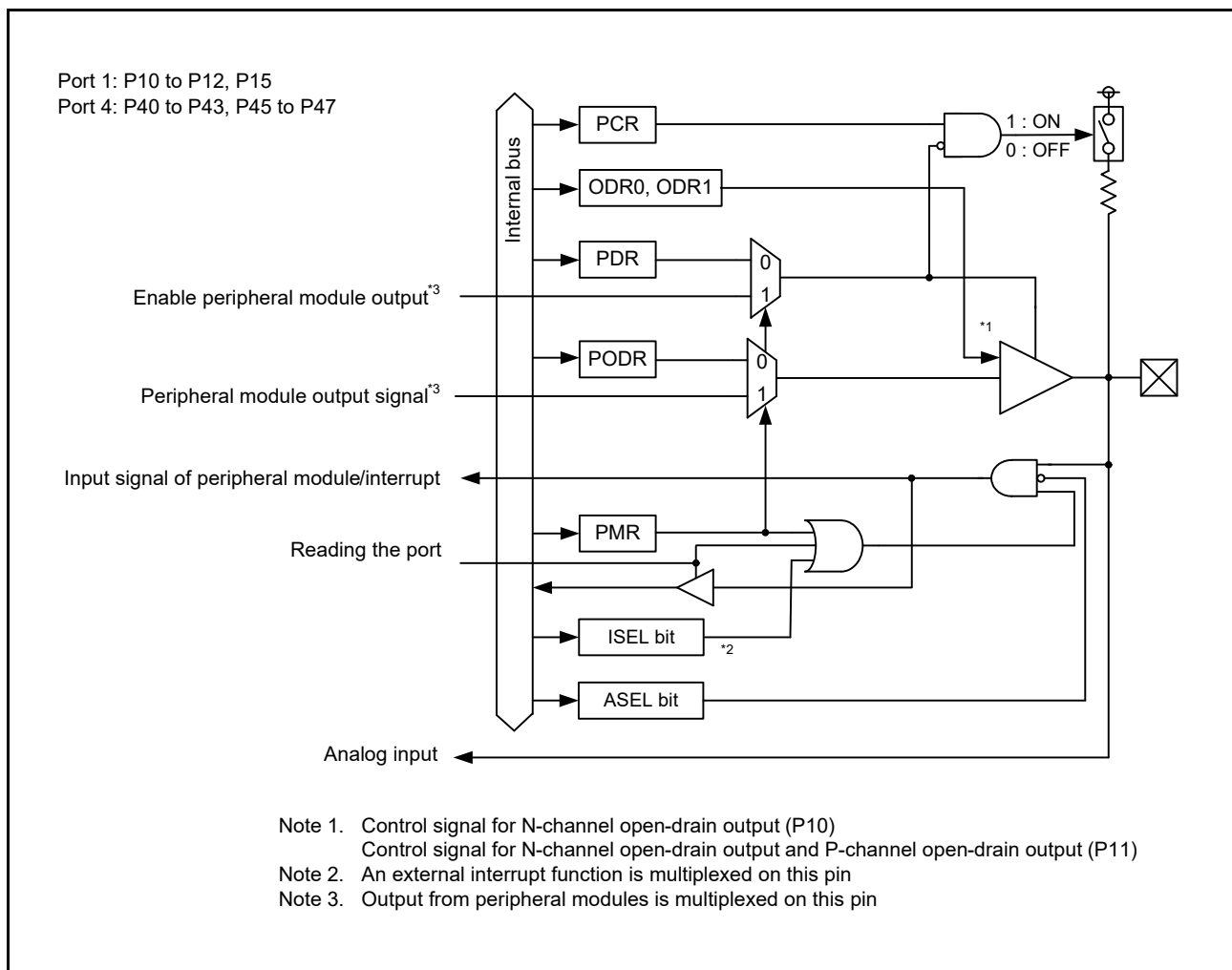


Figure 17.2 I/O Port Configuration (2)

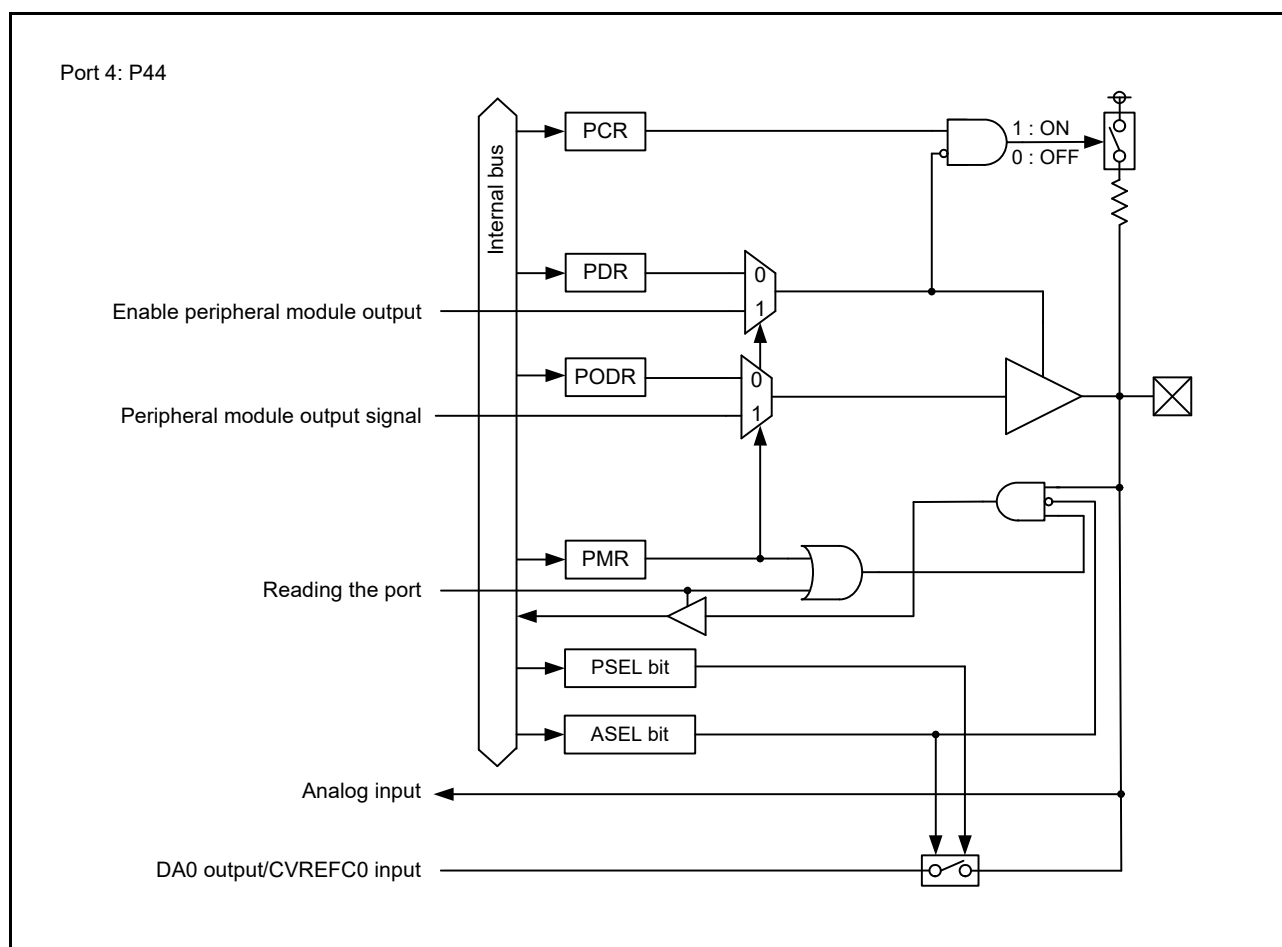


Figure 17.3 I/O Port Configuration (3)

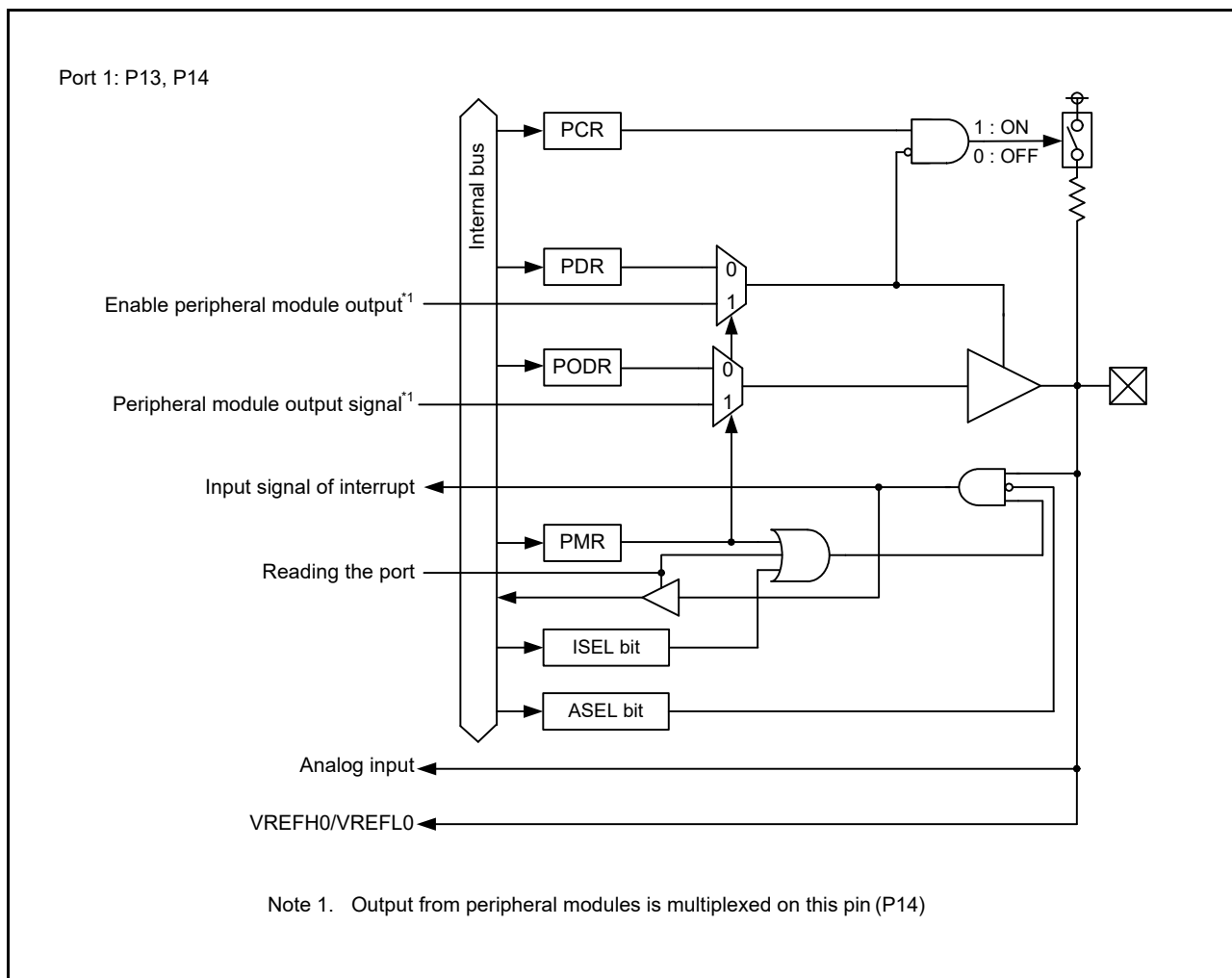


Figure 17.4 I/O Port Configuration (4)

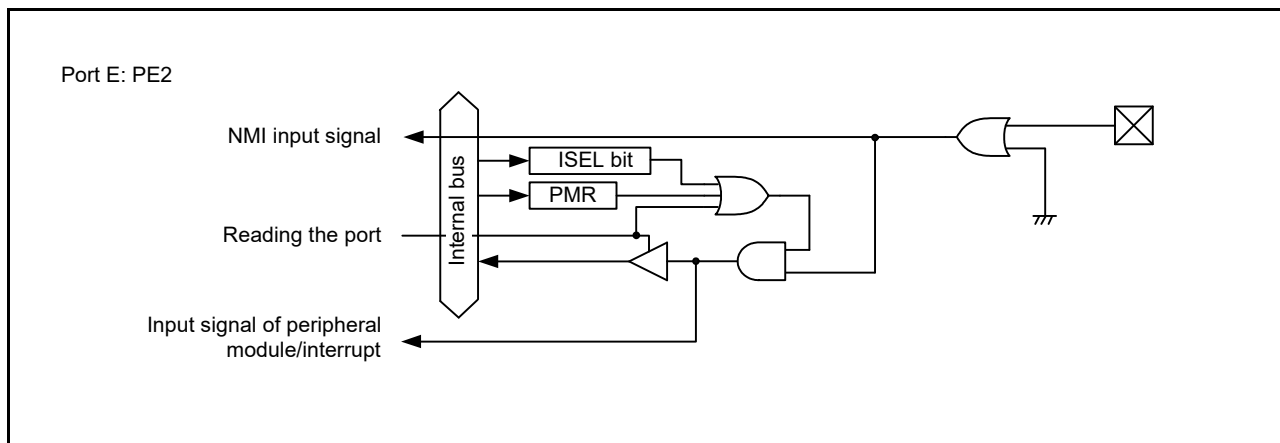


Figure 17.5 I/O Port Configuration (5)

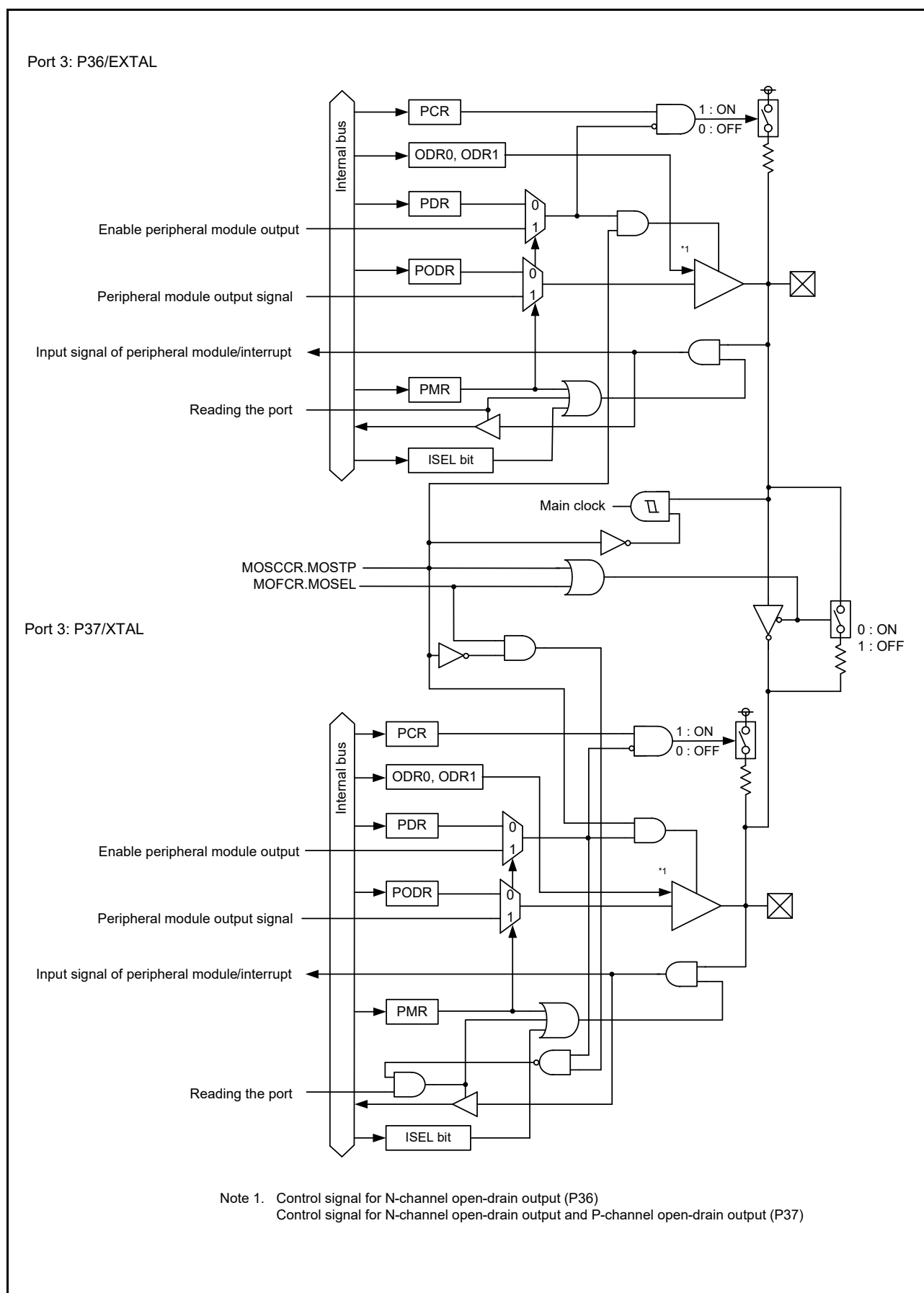


Figure 17.6 I/O Port Configuration (6)

17.3 Register Descriptions

17.3.1 Port Direction Register (PDR)

Address(es): PORT0.PDR 0008 C000h, PORT1.PDR 0008 C001h, PORT2.PDR 0008 C002h, PORT3.PDR 0008 C003h, PORT4.PDR 0008 C004h, PORT7.PDR 0008 C007h, PORT9.PDR 0008 C009h, PORTB.PDR 0008 C00Bh, PORTD.PDR 0008 C00Dh, PORTG.PDR 0008 C010h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

Value after reset: 0 0 0 0 0 0 0 0

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 I/O Select	0: Input (Functions as an input pin.)	R/W
b1	B1	Pm1 I/O Select	1: Output (Functions as an output pin.)	R/W
b2	B2	Pm2 I/O Select		R/W
b3	B3	Pm3 I/O Select		R/W
b4	B4	Pm4 I/O Select		R/W
b5	B5	Pm5 I/O Select		R/W
b6	B6	Pm6 I/O Select		R/W
b7	B7	Pm7 I/O Select		R/W

m = 0 to 4, 7, 9, B, D, G

PDR is used to select the input or output direction for individual pins of the corresponding port m when the pins are configured as the general I/O pins.

Each bit of PORTm.PDR corresponds to each pin of port m; I/O direction can be specified in 1-bit units.

Each bit of PDR corresponding to port m that does not exist is reserved. Make settings according to the description in section 17.4, Initialization of the Port Direction Register (PDR).

17.3.2 Port Output Data Register (PODR)

Address(es): PORT0.PODR 0008 C020h, PORT1.PODR 0008 C021h, PORT2.PODR 0008 C022h, PORT3.PODR 0008 C023h, PORT4.PODR 0008 C024h, PORT7.PODR 0008 C027h, PORT9.PODR 0008 C029h, PORTB.PODR 0008 C02Bh, PORTD.PODR 0008 C02Dh, PORTG.PODR 0008 C030h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

Value after reset: 0 0 0 0 0 0 0 0

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 Output Data Store	Holds output data.	R/W
b1	B1	Pm1 Output Data Store		R/W
b2	B2	Pm2 Output Data Store		R/W
b3	B3	Pm3 Output Data Store		R/W
b4	B4	Pm4 Output Data Store		R/W
b5	B5	Pm5 Output Data Store		R/W
b6	B6	Pm6 Output Data Store		R/W
b7	B7	Pm7 Output Data Store		R/W

m = 0 to 4, 7, 9, B, D, G

PODR holds the data to be output from the pins used for general output ports.

Bits corresponding to port m but which do not exist are reserved. Write 0 to these bits.

The bit corresponding to a pin that does not exist is reserved. A reserved bit is read as 0. The write value should be 0.

17.3.3 Port Input Data Register (PIDR)

Address(es): PORT0.PIDR 0008 C040h, PORT1.PIDR 0008 C041h, PORT2.PIDR 0008 C042h, PORT3.PIDR 0008 C043h, PORT4.PIDR 0008 C044h, PORT7.PIDR 0008 C047h, PORT9.PIDR 0008 C049h, PORTB.PIDR 0008 C04Bh, PORTD.PIDR 0008 C04Dh, PORTE.PIDR 0008 C04Eh, PORTG.PIDR 0008 C050h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

Value after reset: x x x x x x x x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0	Indicates individual pin states of the corresponding port.	R
b1	B1	Pm1		R
b2	B2	Pm2		R
b3	B3	Pm3		R
b4	B4	Pm4		R
b5	B5	Pm5		R
b6	B6	Pm6		R
b7	B7	Pm7		R

m = 0 to 4, 7, 9, B, D, E, G

PIDR indicates individual pin states of port m.

The pin states of port m can be read with the PORTm.PIDR, regardless of the values of PORTm.PDR and PORTm.PMR.

The NMI pin state is reflected in the PE2 bit.

The bit corresponding to a pin that does not exist is reserved. A reserved bit is read as undefined, and cannot be modified.

Note: The state of the P37 pin is not reflected in this register if the PORT3.PDR.B7 bit is set to 1 while the settings of the MOSCCR.MOSTP and MOFCR.MOSEL bits are respectively 0 (enabling the main clock oscillation) and 1 (external oscillator input). Writing to the PORT3.PDR.B7 bit has no effect in this situation.

17.3.4 Port Mode Register (PMR)

Address(es): PORT0.PMR 0008 C060h, PORT1.PMR 0008 C061h, PORT2.PMR 0008 C062h, PORT3.PMR 0008 C063h, PORT4.PMR 0008 C064h, PORT7.PMR 0008 C067h, PORT9.PMR 0008 C069h, PORTB.PMR 0008 C06Bh, PORTD.PMR 0008 C06Dh, PORTE.PMR 0008 C06Eh, PORTG.PMR 0008 C070h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

Value after reset: 0*1 0 0 0 0 0 0 0

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 Pin Mode Control	0: Use pin as general I/O port.	R/W
b1	B1	Pm1 Pin Mode Control	1: Use pin as I/O port for peripheral functions.	R/W
b2	B2	Pm2 Pin Mode Control		R/W
b3	B3	Pm3 Pin Mode Control		R/W
b4	B4	Pm4 Pin Mode Control		R/W
b5	B5	Pm5 Pin Mode Control		R/W
b6	B6	Pm6 Pin Mode Control		R/W
b7	B7	Pm7 Pin Mode Control	- PG7 0: Use pin as general I/O port. 1: Use the pin as the MD function (initial value). - P37, P47, PB7, P97, PD7 0: Use pin as general I/O port (initial value). 1: Use pin as I/O port for peripheral functions.	R/W

m = 0 to 4, 7, 9, B, D, E, G

Note 1. The initial value of the PORTG.PMR.B7 bit is 1.

PMR is a register which specifies the function of the pins of the port.

Each bit of PORTm.PMR corresponds to each pin of port m; pin function can be specified in 1-bit units.

Bits corresponding to port m on the 64 pin-product but which do not exist on a product with fewer than 64 pins are reserved. Write 0 to these bits.

The bit corresponding to a pin that does not exist is reserved. A reserved bit is read as 0. The write value should be 0.

17.3.5 Open Drain Control Register 0 (ODR0)

Address(es): PORT0.ODR0 0008 C080h, PORT1.ODR0 0008 C082h, PORT2.ODR0 0008 C084h, PORT3.ODR0 0008 C086h, PORT7.ODR0 0008 C08Eh, PORT9.ODR0 0008 C092h, PORTB.ODR0 0008 C096h, PORTD.ODR0 0008 C09Ah

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

Value after reset: 0 0 0 0 0 0 0 0

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 Output Type Select	0: CMOS output 1: N-channel open-drain	R/W
b1	B1	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	B2	Pm1 Output Type Select	P01, P31, P71, P91, PB1	R/W
b3	B3		b2 0: CMOS output 1: N-channel open-drain b3 This bit is read as 0. The write value should be 0. P11 b3 b2 0 0: CMOS output 0 1: N-channel open-drain 1 0: P-channel open-drain 1 1: Setting prohibited	R/W
b4	B4	Pm2 Output Type Select	0: CMOS output 1: N-channel open-drain	R/W
b5	B5	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	B6	Pm3 Output Type Select	0: CMOS output 1: N-channel open-drain	R/W
b7	B7	Reserved	This bit is read as 0. The write value should be 0.	R/W

m = 0 to 3, 7, 9, B, D

ODR0 is a register which is used to select an output type for the pins of the port.

Bits corresponding to port m on the 64 pin-product but which do not exist on a product with fewer than 64 pins are reserved. Write 0 to these bits.

The bits corresponding to a pin that does not exist or pins with no open-drain output allocation are reserved. A reserved bit is read as 0. The write value should be 0.

17.3.6 Open Drain Control Register 1 (ODR1)

Address(es): PORT0.ODR1 0008 C081h, PORT2.ODR1 0008 C085h, PORT3.ODR1 0008 C087h, PORT7.ODR1 0008 C08Fh, PORT9.ODR1 0008 C093h, PORTB.ODR1 0008 C097h, PORTD.ODR1 0008 C09Bh, PORTG.ODR1 0008 C0A1h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

Value after reset: 0 0 0 0 0 0 0 0

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm4 Output Type Select	P04, P24, P74, PB4, PD4	R/W
b1	B1		b0 0: CMOS output 1: N-channel open-drain b1 This bit is read as 0. The write value should be 0. P94 b1 b0 0 0: CMOS output 0 1: N-channel open-drain 1 0: P-channel open-drain 1 1: Setting prohibited	R/W
b2	B2	Pm5 Output Type Select	0: CMOS output 1: N-channel open-drain	R/W
b3	B3	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	B4	Pm6 Output Type Select	P26, P36, P76, PB6, PD6	R/W
b5	B5		b4 0: CMOS output 1: N-channel open-drain b5 This bit is read as 0. The write value should be 0. P96 b5 b4 0 0: CMOS output 0 1: N-channel open-drain 1 0: P-channel open-drain 1 1: Setting prohibited	R/W
b6	B6	Pm7 Output Type Select	P97, PB7, PD7, PG7	R/W
b7	B7		b6 0: CMOS output 1: N-channel open-drain b7 This bit is read as 0. The write value should be 0. P37 b7 b6 0 0: CMOS output 0 1: N-channel open-drain 1 0: P-channel open-drain 1 1: Setting prohibited	R/W

m = 0 to 3, 7, 9, B, D, G

ODR1 is used to select an output type for each pin of the port.

Bits corresponding to port m on the 64 pin-product but which do not exist on a product with fewer than 64 pins are reserved. Write 0 to these bits.

The bits corresponding to a pin that does not exist or pins with no open-drain output allocation are reserved. A reserved bit is read as 0. The write value should be 0.

17.3.7 Pull-Up Control Register (PCR)

Address(es): PORT0.PCR 0008 C0C0h, PORT1.PCR 0008 C0C1h, PORT2.PCR 0008 C0C2h, PORT3.PCR 0008 C0C3h, PORT4.PCR 0008 C0C4h, PORT7.PCR 0008 C0C7h, PORT9.PCR 0008 C0C9h, PORTB.PCR 0008 C0CBh, PORTD.PCR 0008 C0CDh, PORTG.PCR 0008 C0D0h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

Value after reset: 0*1 0 0 0 0 0 0 0

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 Input Pull-Up Resistor Control	0: Disables an input pull-up resistor.	R/W
b1	B1	Pm1 Input Pull-Up Resistor Control	1: Enables an input pull-up resistor.	R/W
b2	B2	Pm2 Input Pull-Up Resistor Control		R/W
b3	B3	Pm3 Input Pull-Up Resistor Control		R/W
b4	B4	Pm4 Input Pull-Up Resistor Control		R/W
b5	B5	Pm5 Input Pull-Up Resistor Control		R/W
b6	B6	Pm6 Input Pull-Up Resistor Control		R/W
b7	B7	Pm7 Input Pull-Up Resistor Control		R/W

m = 0 to 4, 7, 9, B, D, G

Note 1. The initial value of the PORTG.PCR.B7 bit is 1.

PCR is a register which enables or disables an input pull-up resistor for individual pins of the port.

While a pin is in the input state with the corresponding bit in PORTm.PCR set to 1, the pull-up resistor connected to the pin is enabled.

When a pin is used as a general port output pin, or a peripheral function output pin, the pull-up resistor for the pin is disabled regardless of the settings of the PCR register.

The pull-up resistor is also disabled in the reset state.

The bit corresponding to a pin that does not exist is also reserved. A reserved bit is read as 0. The write value should be 0.

17.4 Initialization of the Port Direction Register (PDR)

Initialize reserved bits in the PDR register according to Table 17.3 to Table 17.8.

- The blank columns in Table 17.3 to Table 17.8 indicate the bits corresponding to the pins listed in Table 17.1, Specifications of I/O Ports.

The corresponding bits should be set to 1 (output) or 0 (input) depending on the user system.

- The columns other than the blank columns in Table 17.3 to Table 17.8 indicate reserved bits.

A reserved bit should be set to 0 (input) or 1 (output) according to Table 17.3 to Table 17.8.

When setting a value to a reserved bit, access in byte units.

Table 17.3 PDR Register Settings in 64-Pin Packages

Port Symbol	PDR Register							
	b7	b6	b5	b4	b3	b2	b1	b0
PORT0	1	1	1					
PORT1	1	1						
PORT2	1						1	1
PORT3			1	1				
PORT4								
PORT7	1							
PORT9								
PORTB								
PORTD						1	1	1
PORTG		1	1	1	1	1	1	1

Table 17.4 PDR Register Settings in 52-Pin Packages

Port Symbol	PDR Register							
	b7	b6	b5	b4	b3	b2	b1	b0
PORT0	1	1	1		1		1	1
PORT1	1	1	1			1		
PORT2	1		1				1	1
PORT3			1	1		1		1
PORT4								
PORT7	1							
PORT9							1	1
PORTB						1		
PORTD	1					1	1	1
PORTG		1	1	1	1	1	1	1

Table 17.5 PDR Register Settings in 48-Pin Packages

Port Symbol	PDR Register							
	b7	b6	b5	b4	b3	b2	b1	b0
PORT0	1	1	1	1	1	1	1	1
PORT1	1	1	1			1		
PORT2	1		1				1	1
PORT3			1	1	1	1		1
PORT4								
PORT7	1							
PORT9							1	1
PORTB	1					1		
PORTD	1					1	1	1
PORTG		1	1	1	1	1	1	1

Table 17.6 PDR Register Settings in 44-Pin Packages

Port Symbol	PDR Register							
	b7	b6	b5	b4	b3	b2	b1	b0
PORT0	1	1	1	1	1	1	1	1
PORT1	1	1	1			1		
PORT2	1		1				1	1
PORT3	1		1	1	1	1		1
PORT4			1					
PORT7	1							1
PORT9							1	1
PORTB	1			1		1		
PORTD	1					1	1	1
PORTG		1	1	1	1	1	1	1

Table 17.7 PDR Register Settings in 32-Pin Packages

Port Symbol	PDR Register							
	b7	b6	b5	b4	b3	b2	b1	b0
PORT0	1	1	1	1	1	1	1	1
PORT1	1	1	1			1		1
PORT2	1		1	1	1	1	1	1
PORT3			1	1	1	1		1
PORT4	1		1		1			
PORT7	1							1
PORT9			1	1			1	1
PORTB	1			1	1	1		
PORTD	1	1	1	1	1	1	1	1
PORTG		1	1	1	1	1	1	1

Table 17.8 PDR Register Settings in 24-Pin Packages

Port Symbol	PDR Register							
	b7	b6	b5	b4	b3	b2	b1	b0
PORT0	1	1	1	1	1	1	1	1
PORT1	1	1	1			1		1
PORT2	1		1	1	1	1	1	1
PORT3	1		1	1	1	1		1
PORT4	1	1	1	1	1			
PORT7	1					1	1	1
PORT9			1	1	1	1	1	1
PORTB	1			1	1	1	1	
PORTD	1	1	1	1	1	1	1	1
PORTG		1	1	1	1	1	1	1

17.5 Handling of Unused Pins

The configuration of unused pins is listed in Table 17.9.

Table 17.9 Unused Pin Configuration

Pin Name	Description
PG7/MD	(Always used as mode pin)
RES#	Connect this pin to VCC via a pull-up resistor.
PE2/NMI	Connect this pin to VCC via a pull-up resistor.
Ports 0 to 4, 7, 9 Ports B, D, G	• If the direction setting is for input (PORTn.PDR = 0), the corresponding pin is connected to VCC (pulled up) via a resistor or to VSS (pulled down) via a resistor.*1 • If the direction setting is for output (PORTn.PDR = 1), the pin is released.*1, *2

Note 1. Clear the PORTn.PMR bit, the PmnPFS.ISEL bit and the PmnPFS.ASEL bit to 0.

Note 2. In the case of release when the setting is for output, the port is an input over the period from release from the reset state to the pin becoming an output. Since the voltage on the pin is undefined while it is an input, this may lead to an increase in the current drawn.

18. Multi-Function Pin Controller (MPC)

18.1 Overview

The multi-function pin controller (MPC) is used to allocate input and output signals for peripheral modules and input interrupt signals to pins from among multiple ports.

Table 18.1 shows the allocation of pin functions to multiple pins. The symbols ✓ and × in the table indicate whether the pins are or are not present on the given package. Allocating the same function to more than one pin is prohibited.

Table 18.1 Allocation of Pin Functions to Multiple Pins (1 / 8)

Module/Function	Channel	Pin Functions	Allocation Port	Package					
				64-pin	52-pin	48-pin	44-pin	32-pin	24-pin
Interrupt		NMI (input)	PE2	✓	✓	✓	✓	✓	✓
Interrupt	IRQ0	IRQ0 (input)	P10	✓	✓	✓	✓	×	×
			P37	✓	✓	✓	×	✓	×
			P92	✓	✓	✓	✓	✓	×
			PE2	✓	✓	✓	✓	✓	✓
	IRQ1	IRQ1 (input)	P11	✓	✓	✓	✓	✓	✓
			P26	✓	✓	✓	✓	✓	✓
			P93	✓	✓	✓	✓	✓	×
	IRQ2	IRQ2 (input)	P01	✓	×	×	×	×	×
			P22	✓	✓	✓	✓	×	×
			P97	✓	✓	✓	✓	✓	✓
			PB1	✓	✓	✓	✓	✓	×
			PD4	✓	✓	✓	✓	×	×
	IRQ3	IRQ3 (input)	P24	✓	✓	✓	✓	×	×
			P36	✓	✓	✓	✓	✓	✓
			PB3	✓	✓	✓	✓	×	×
			PD5	✓	✓	✓	✓	×	×
	IRQ4	IRQ4 (input)	P00	✓	×	×	×	×	×
			P23	✓	✓	✓	✓	×	×
			P13	✓	✓	✓	✓	✓	✓
			P94	✓	✓	✓	✓	×	×
	IRQ5	IRQ5 (input)	P02	✓	✓	×	×	×	×
			P70	✓	✓	✓	×	×	×
			PB6	✓	✓	✓	✓	✓	✓
			PD6	✓	✓	✓	✓	×	×
	IRQ6	IRQ6 (input)	P14	✓	✓	✓	✓	✓	✓
			P71	✓	✓	✓	✓	✓	×
			P90	✓	×	×	×	×	×
	IRQ7	IRQ7 (input)	P12	✓	×	×	×	×	×
			P72	✓	✓	✓	✓	✓	×
			PB5	✓	✓	✓	✓	✓	✓
Clock generation circuit		CLKOUT (output)	P96	✓	✓	✓	✓	✓	✓
			P24	✓	✓	✓	✓	×	×

Table 18.1 Allocation of Pin Functions to Multiple Pins (2 / 8)

Module/Function	Channel	Pin Functions	Allocation Port	Package					
				64-pin	52-pin	48-pin	44-pin	32-pin	24-pin
Multi-function timer unit 3	MTU0	MTIOC0A (input/output)/MTIOC0A# (input/output)	P30	✓	x	x	x	x	x
			P36	✓	✓	✓	✓	✓	✓
			PB1	✓	✓	✓	✓	✓	x
			PD3	✓	✓	✓	✓	x	x
		MTIOC0B (input/output)/MTIOC0B# (input/output)	P25	✓	x	x	x	x	x
			P92	✓	✓	✓	✓	✓	x
			PB0	✓	✓	✓	✓	✓	✓
			PD4	✓	✓	✓	✓	x	x
		MTIOC0C (input/output)/MTIOC0C# (input/output)	P33	✓	✓	x	x	x	x
			P93	✓	✓	✓	✓	✓	x
			P97	✓	✓	✓	✓	✓	✓
			PD5	✓	✓	✓	✓	x	x
		MTIOC0D (input/output)/MTIOC0D# (input/output)	P32	✓	x	x	x	x	x
			P96	✓	✓	✓	✓	✓	✓
			PD6	✓	✓	✓	✓	x	x
	MTU1	MTIOC1A (input/output)/MTIOC1A# (input/output)	P04	✓	✓	x	x	x	x
			P36	✓	✓	✓	✓	✓	✓
			P92	✓	✓	✓	✓	✓	x
			P94	✓	✓	✓	✓	x	x
		MTIOC1B (input/output)/MTIOC1B# (input/output)	P03	✓	x	x	x	x	x
			P95	✓	✓	✓	✓	x	x
			PB5	✓	✓	✓	✓	✓	✓
	MTU2	MTIOC2A (input/output)/MTIOC2A# (input/output)	P95	✓	✓	✓	✓	x	x
			PB5	✓	✓	✓	✓	✓	✓
		MTIOC2B (input/output)/MTIOC2B# (input/output)	P93	✓	✓	✓	✓	✓	x
			P94	✓	✓	✓	✓	x	x
			PB6	✓	✓	✓	✓	✓	✓
	MTU3	MTIOC3A (input/output)/MTIOC3A# (input/output)	P11	✓	✓	✓	✓	✓	✓
			P33	✓	✓	x	x	x	x
			PB5	✓	✓	✓	✓	✓	✓
		MTIOC3B (input/output)/MTIOC3B# (input/output)	P26	✓	✓	✓	✓	✓	✓
			P71	✓	✓	✓	✓	✓	x
			P94	✓	✓	✓	✓	x	x
		MTIOC3C (input/output)/MTIOC3C# (input/output)	P32	✓	x	x	x	x	x
			PB6	✓	✓	✓	✓	✓	✓
		MTIOC3D (input/output)/MTIOC3D# (input/output)	P74	✓	✓	✓	✓	✓	✓
			P97	✓	✓	✓	✓	✓	✓
	MTU4	MTIOC4A (input/output)/MTIOC4A# (input/output)	P31	✓	✓	✓	✓	✓	✓
			P36	✓	✓	✓	✓	✓	✓
			P72	✓	✓	✓	✓	✓	x
			P95	✓	✓	✓	✓	x	x
			PD3	✓	✓	✓	✓	x	x

Table 18.1 Allocation of Pin Functions to Multiple Pins (3 / 8)

Module/Function	Channel	Pin Functions	Allocation Port	Package					
				64-pin	52-pin	48-pin	44-pin	32-pin	24-pin
Multi-function timer unit 3	MTU4	MTIOC4B (input/output)/MTIOC4B# (input/output)	P25	✓	x	x	x	x	x
			P73	✓	✓	✓	✓	✓	✓
			P96	✓	✓	✓	✓	✓	✓
			PD4	✓	✓	✓	✓	x	x
		MTIOC4C (input/output)/MTIOC4C# (input/output)	P11	✓	✓	✓	✓	✓	✓
			P75	✓	✓	✓	✓	✓	✓
			PB0	✓	✓	✓	✓	✓	✓
			PD5	✓	✓	✓	✓	x	x
		MTIOC4D (input/output)/MTIOC4D# (input/output)	P76	✓	✓	✓	✓	✓	✓
			PB1	✓	✓	✓	✓	✓	x
			PD6	✓	✓	✓	✓	x	x
	MTU5	MTIC5U (input)/MTIC5U# (input)	P24	✓	✓	✓	✓	x	x
			P91	✓	x	x	x	x	x
		MTIC5V (input)/MTIC5V# (input)	P23	✓	✓	✓	✓	x	x
			P90	✓	x	x	x	x	x
		MTIC5W (input)/MTIC5W# (input)	P22	✓	✓	✓	✓	x	x
			PB2	✓	x	x	x	x	x
	MTU	MTCLKA (input)/MTCLKA# (input)	P23	✓	✓	✓	✓	x	x
			P33	✓	✓	x	x	x	x
			P93	✓	✓	✓	✓	✓	x
			P97	✓	✓	✓	✓	✓	✓
		MTCLKB (input)/MTCLKB# (input)	P22	✓	✓	✓	✓	x	x
			P32	✓	x	x	x	x	x
			P96	✓	✓	✓	✓	✓	✓
		MTCLKC (input)/MTCLKC# (input)	P11	✓	✓	✓	✓	✓	✓
			P30	✓	x	x	x	x	x
			PB0	✓	✓	✓	✓	✓	✓
		MTCLKD (input)/MTCLKD# (input)	P10	✓	✓	✓	✓	x	x
			P25	✓	x	x	x	x	x
			PB6	✓	✓	✓	✓	✓	✓
		ADSM0 (output)	PB0	✓	✓	✓	✓	✓	✓
		ADSM1 (output)	P97	✓	✓	✓	✓	✓	✓
General PWM timer	GPTW0	GTIOC0A (input/output)/GTIOC0A# (input/output)	P26	✓	✓	✓	✓	✓	✓
			P36	✓	✓	✓	✓	✓	✓
			P71	✓	✓	✓	✓	✓	x
			P94	✓	✓	✓	✓	x	x
		GTIOC0B (input/output)/GTIOC0B# (input/output)	P31	✓	✓	✓	✓	✓	✓
			P74	✓	✓	✓	✓	✓	✓
			P97	✓	✓	✓	✓	✓	✓

Table 18.1 Allocation of Pin Functions to Multiple Pins (4 / 8)

Module/Function	Channel	Pin Functions	Allocation Port	Package					
				64-pin	52-pin	48-pin	44-pin	32-pin	24-pin
General PWM timer	GPTW1	GTIOC1A (input/output)/GTIOC1A# (input/output)	P31	✓	✓	✓	✓	✓	✓
			P72	✓	✓	✓	✓	✓	x
			P92	✓	✓	✓	✓	✓	x
			P95	✓	✓	✓	✓	x	x
			PB5	✓	✓	✓	✓	✓	✓
			PD3	✓	✓	✓	✓	x	x
		GTIOC1B (input/output)/GTIOC1B# (input/output)	P75	✓	✓	✓	✓	✓	✓
			P93	✓	✓	✓	✓	✓	x
			PB0	✓	✓	✓	✓	✓	✓
			PD4	✓	✓	✓	✓	x	x
	GPTW2	GTIOC2A (input/output)/GTIOC2A# (input/output)	P04	✓	✓	x	x	x	x
			P23	✓	✓	✓	✓	x	x
			P73	✓	✓	✓	✓	✓	✓
			P96	✓	✓	✓	✓	✓	✓
			PB5	✓	✓	✓	✓	✓	✓
			PD5	✓	✓	✓	✓	x	x
		GTIOC2B (input/output)/GTIOC2B# (input/output)	P03	✓	x	x	x	x	x
			P24	✓	✓	✓	✓	x	x
			P76	✓	✓	✓	✓	✓	✓
			PB1	✓	✓	✓	✓	✓	x
			PB6	✓	✓	✓	✓	✓	✓
			PD6	✓	✓	✓	✓	x	x
	GPTW	GTETRGA (input)	P25	✓	x	x	x	x	x
			P97	✓	✓	✓	✓	✓	✓
			PE2	✓	✓	✓	✓	✓	✓
		GTADSM0 (output)	P23	✓	✓	✓	✓	x	x
			P46	✓	✓	✓	✓	✓	x
			PB0	✓	✓	✓	✓	✓	✓
		GTADSM1 (output)	P22	✓	✓	✓	✓	x	x
			P44	✓	✓	✓	✓	✓	x
			P97	✓	✓	✓	✓	✓	✓
		GTCPP00 (output)	P24	✓	✓	✓	✓	x	x
			P36	✓	✓	✓	✓	✓	✓
			P70	✓	✓	✓	x	x	x
			PB6	✓	✓	✓	✓	✓	✓
Port output enable 3	POE0	POE0# (input)	P70	✓	✓	✓	x	x	x
			P96	✓	✓	✓	✓	✓	✓
	POE8	POE8# (input)	P11	✓	✓	✓	✓	✓	✓
			P75	✓	✓	✓	✓	✓	✓
			PB3	✓	✓	✓	✓	x	x
	POE10	POE10# (input)	P32	✓	x	x	x	x	x
			PE2	✓	✓	✓	✓	✓	✓

Table 18.1 Allocation of Pin Functions to Multiple Pins (5 / 8)

Module/Function	Channel	Pin Functions	Allocation Port	Package					
				64-pin	52-pin	48-pin	44-pin	32-pin	24-pin
Port output enable 3	POE11	POE11# (input)	P30	✓	x	x	x	x	x
			P37	✓	✓	✓	x	✓	x
			PB6	✓	✓	✓	✓	✓	✓
	POE12	POE12# (input)	P25	✓	x	x	x	x	x
			P36	✓	✓	✓	✓	✓	✓
			PB5	✓	✓	✓	✓	✓	✓
	TMR0	TMO0 (output)	P04	✓	✓	x	x	x	x
			PB5	✓	✓	✓	✓	✓	✓
			PD3	✓	✓	✓	✓	x	x
		TMCI0 (input)	PB6	✓	✓	✓	✓	✓	✓
			PD4	✓	✓	✓	✓	x	x
		TMRI0 (input)	PE2	✓	✓	✓	✓	✓	✓
			PD5	✓	✓	✓	✓	x	x
	TMR1	TMO1 (output)	P10	✓	✓	✓	✓	x	x
			P93	✓	✓	✓	✓	✓	x
			P96	✓	✓	✓	✓	✓	✓
			PD6	✓	✓	✓	✓	x	x
		TMCI1 (input)	P91	✓	x	x	x	x	x
			P97	✓	✓	✓	✓	✓	✓
			PE2	✓	✓	✓	✓	✓	✓
		TMRI1 (input)	P92	✓	✓	✓	✓	✓	x
			PB0	✓	✓	✓	✓	✓	✓
8-bit timer	TMR2	TMO2 (output)	PD7	✓	x	x	x	x	x
			P11	✓	✓	✓	✓	✓	✓
			P23	✓	✓	✓	✓	x	x
		TMCI2 (input)	P74	✓	✓	✓	✓	✓	✓
			P24	✓	✓	✓	✓	x	x
		TMRI2 (input)	P75	✓	✓	✓	✓	✓	✓
			P22	✓	✓	✓	✓	x	x
		TMR3	P76	✓	✓	✓	✓	✓	✓
			P11	✓	✓	✓	✓	✓	✓
	TMR3	TMO3 (output)	P26	✓	✓	✓	✓	✓	✓
			P04	✓	✓	x	x	x	x
		TMCI3 (input)	P31	✓	✓	✓	✓	✓	✓
			P10	✓	✓	✓	✓	x	x
		TMRI3 (input)	P25	✓	x	x	x	x	x
			P73	✓	✓	✓	✓	✓	✓
		TMR3	P72	✓	✓	✓	✓	✓	x
			PB6	✓	✓	✓	✓	✓	✓
		TMR3	PD5	✓	✓	✓	✓	x	x
			P71	✓	✓	✓	✓	✓	x
		TMR3	PB5	✓	✓	✓	✓	✓	✓
			PD3	✓	✓	✓	✓	x	x
Serial communications interface	SCI1	RXD1 (input)/ SMISO1 (input/output)/SSCL1 (input/output)	P72	✓	✓	✓	✓	✓	x
			PB6	✓	✓	✓	✓	✓	✓
			PD5	✓	✓	✓	✓	x	x
		TXD1 (output)/ SMOSI1 (input/output)/SSDA1 (input/output)	P71	✓	✓	✓	✓	✓	x
			PB5	✓	✓	✓	✓	✓	✓
			PD3	✓	✓	✓	✓	x	x

Table 18.1 Allocation of Pin Functions to Multiple Pins (6 / 8)

Module/Function	Channel	Pin Functions	Allocation Port	Package					
				64-pin	52-pin	48-pin	44-pin	32-pin	24-pin
Serial communications interface	SCI1	SCK1 (input/output)	P70	✓	✓	✓	x	x	x
			PB0	✓	✓	✓	✓	✓	✓
			PD4	✓	✓	✓	✓	x	x
		CTS1# (input)/RTS1# (output)/SS1# (input)	P02	✓	✓	x	x	x	x
			P11	✓	✓	✓	✓	✓	✓
			PD6	✓	✓	✓	✓	x	x
	SCI5	RXD5 (input)/SMISO5 (input/output)/SSCL5 (input/output)	P24	✓	✓	✓	✓	x	x
			P97	✓	✓	✓	✓	✓	✓
			PB6	✓	✓	✓	✓	✓	✓
		TXD5 (output)/SMOSI5 (input/output)/SSDA5 (input/output)	P23	✓	✓	✓	✓	x	x
			PB0	✓	✓	✓	✓	✓	✓
			PB5	✓	✓	✓	✓	✓	✓
		SCK5 (input/output)	P22	✓	✓	✓	✓	x	x
			P92	✓	✓	✓	✓	✓	x
			P96	✓	✓	✓	✓	✓	✓
			PB1	✓	✓	✓	✓	✓	x
			PB7	✓	✓	x	x	x	x
		CTS5# (input)/RTS5# (output)/SS5# (input)	P94	✓	✓	✓	✓	x	x
			PG7	✓	✓	✓	✓	✓	✓
	SCI6	RXD6 (input)/SMISO6 (input/output)/SSCL6 (input/output)	P26	✓	✓	✓	✓	✓	✓
			P36	✓	✓	✓	✓	✓	✓
			PB4	✓	✓	✓	x	x	x
		TXD6 (output)/SMOSI6 (input/output)/SSDA6 (input/output)	P11	✓	✓	✓	✓	✓	✓
			P31	✓	✓	✓	✓	✓	✓
			P37	✓	✓	✓	x	✓	x
			PB3	✓	✓	✓	✓	x	x
		SCK6 (input/output)	P30	✓	x	x	x	x	x
			P73	✓	✓	✓	✓	✓	✓
			PB2	✓	x	x	x	x	x
			PG7	✓	✓	✓	✓	✓	✓
		CTS6# (input)/RTS6# (output)/SS6# (input)	P74	✓	✓	✓	✓	✓	✓
			PB7	✓	✓	x	x	x	x
	SCI12	RXD12 (input)/SMISO12 (input/output)/SSCL12 (input/output)/RXDX12 (input)	P36	✓	✓	✓	✓	✓	✓
			P76	✓	✓	✓	✓	✓	✓
			P93	✓	✓	✓	✓	✓	x
		TXD12 (output)/SMOSI12 (input/output)/SSDA12 (input/output)/SIOX12 (input/output)/TXDX12 (output)	P11	✓	✓	✓	✓	✓	✓
			P37	✓	✓	✓	x	✓	x
			P94	✓	✓	✓	✓	x	x
			P96	✓	✓	✓	✓	✓	✓

Table 18.1 Allocation of Pin Functions to Multiple Pins (7 / 8)

Module/Function	Channel	Pin Functions	Allocation Port	Package					
				64-pin	52-pin	48-pin	44-pin	32-pin	24-pin
Serial communications interface	SCI12	SCK12 (input/output)	P75	✓	✓	✓	✓	✓	✓
			P92	✓	✓	✓	✓	✓	x
			PB1	✓	✓	✓	✓	✓	x
			PG7	✓	✓	✓	✓	✓	✓
		CTS12# (input)/ RTS12# (output)/ SS12# (input)	P74	✓	✓	✓	✓	✓	✓
			P95	✓	✓	✓	✓	x	x
I ² C bus interface	RIIC0	SCL0 (input/output)	P97	✓	✓	✓	✓	✓	✓
			P26*2	✓	✓	✓	✓	✓	✓
		SDA0 (input/output)	PB0	✓	✓	✓	✓	✓	✓
			P31*2	✓	✓	✓	✓	✓	✓
12-bit A/D converter		AN000 (input)*1	P40	✓	✓	✓	✓	✓	✓
		AN001 (input)*1	P41	✓	✓	✓	✓	✓	✓
		AN002 (input)*1	P42	✓	✓	✓	✓	✓	✓
		AN003 (input)*1	P43	✓	✓	✓	✓	x	x
		AN004 (input)*1	P44	✓	✓	✓	✓	✓	x
		AN005 (input)*1	P45	✓	✓	✓	x	x	x
		AN006 (input)*1	P46	✓	✓	✓	✓	✓	x
		AN007 (input)*1	P47	✓	✓	✓	✓	x	x
		AN102 (input)*1	P11	✓	✓	✓	✓	✓	✓
		AN103 (input)*1	P10	✓	✓	✓	✓	x	x
		AN104 (input)*1	P12	✓	x	x	x	x	x
		AN105 (input)*1	P13	✓	✓	✓	✓	✓	✓
		AN106 (input)*1	P14	✓	✓	✓	✓	✓	✓
		AN107 (input)*1	P15	✓	x	x	x	x	x
		ADTRG0# (input)	P03	✓	x	x	x	x	x
			P92	✓	✓	✓	✓	✓	x
			PB4	✓	✓	✓	x	x	x
		ADTRG1# (input)	P93	✓	✓	✓	✓	✓	x
			PB3	✓	✓	✓	✓	x	x
		ADST0 (output)	P02	✓	✓	x	x	x	x
			P14	✓	✓	✓	✓	✓	✓
			PD6	✓	✓	✓	✓	x	x
			PG7	✓	✓	✓	✓	✓	✓
		ADST1 (output)	P01	✓	x	x	x	x	x
			P36	✓	✓	✓	✓	✓	✓
		ADSS0 (output)	P00	✓	x	x	x	x	x
			P23	✓	✓	✓	✓	x	x
			P46	✓	✓	✓	✓	✓	x
			PB6	✓	✓	✓	✓	✓	✓
			PD5	✓	✓	✓	✓	x	x

Table 18.1 Allocation of Pin Functions to Multiple Pins (8 / 8)

Module/Function	Channel	Pin Functions	Allocation Port	Package					
				64-pin	52-pin	48-pin	44-pin	32-pin	24-pin
12-bit A/D converter	ADSS1 (output)		P04	✓	✓	×	×	×	×
			P22	✓	✓	✓	✓	×	×
			P37	✓	✓	✓	×	✓	×
			P44	✓	✓	✓	✓	✓	×
			PB5	✓	✓	✓	✓	✓	✓
D/A converter	DA0 (output)*1		P44	✓	✓	✓	✓	✓	×
Clock frequency accuracy measurement circuit	CACREF (input)		P00	✓	×	×	×	×	×
			P23	✓	✓	✓	✓	×	×
			P76	✓	✓	✓	✓	✓	✓
			PB1	✓	✓	✓	✓	✓	×
PGA	PGAIN00 (input)*1		P40	✓	✓	✓	✓	✓	✓
	PGAIN01 (input)*1		P41	✓	✓	✓	✓	✓	✓
	PGAIN02 (input)*1		P42	✓	✓	✓	✓	✓	✓
	PGAOUT0 (output)*1		P22	✓	✓	✓	✓	×	×
	PGAGND (input)*1		P47	✓	✓	✓	✓	×	×
Comparator	COMP0 (output)		P24	✓	✓	✓	✓	×	×
	COMP1 (output)		P23	✓	✓	✓	✓	×	×
	COMP2 (output)		P22	✓	✓	✓	✓	×	×
	CVREFC0 (input)*1		P44	✓	✓	✓	✓	✓	×
	CMPC00 (input)*1		P40	✓	✓	✓	✓	✓	✓
	CMPC01 (input)*1		P40	✓	✓	✓	✓	✓	✓
	CMPC02 (input)*1		P11	✓	✓	✓	✓	✓	✓
	CMPC03 (input)*1		P44	✓	✓	✓	✓	✓	×
	CMPC10 (input)*1		P41	✓	✓	✓	✓	✓	✓
	CMPC11 (input)*1		P41	✓	✓	✓	✓	✓	✓
	CMPC12 (input)*1		P13	✓	✓	✓	✓	✓	✓
	CMPC13 (input)*1		P46	✓	✓	✓	✓	✓	×
	CMPC20 (input)*1		P42	✓	✓	✓	✓	✓	✓
	CMPC21 (input)*1		P42	✓	✓	✓	✓	✓	✓
	CMPC22 (input)*1		P14	✓	✓	✓	✓	✓	✓

Note 1. Select general input (by setting the Bm bits for the given pin in the PDR and PMR for the given port to 0) for the pin if this pin function is to be used.

Note 2. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

18.2 Register Descriptions

Registers and bits for pins that are not present due to differences according to the package are reserved. Write the value after a reset when writing to such bits.

18.2.1 Write-Protect Register (PWPR)

Address(es): MPC.PWPR 0008 C11Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	B0WI	PFSWE	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	PFSWE	PFS Register Write Enable	0: Writing to the PFS register is disabled 1: Writing to the PFS register is enabled	R/W
b7	B0WI	PFSWE Bit Write Disable	0: Writing to the PFSWE bit is enabled 1: Writing to the PFSWE bit is disabled	R/W

PFSWE Bit (PFS Register Write Enable)

Writing to PmnPFS register is enabled only when the PFSWE bit is set to 1.

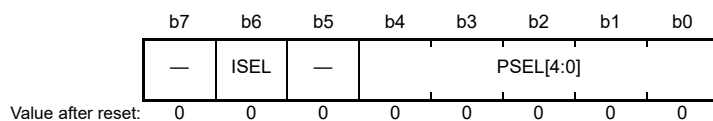
To write to the PFSWE bit after writing 0 to the B0WI bit.

B0WI Bit (PFSWE Bit Write Disable)

Writing to the PFSWE bit is enabled only when the B0WI bit is set to 0.

18.2.2 P0n Pin Function Control Register (P0nPFS) (n = 0 to 4)

Address(es): MPC.P00PFS 0008 C140h, MPC.P01PFS 0008 C141h, MPC.P02PFS 0008 C142h, MPC.P03PFS 0008 C143h, MPC.P04PFS 0008 C144h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.2.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Input Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin P00: IRQ4 (64 pin) P01: IRQ2 (64 pin) P02: IRQ5 (64/52 pin)	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

The Pmn pin function control register (PmnPFS) selects the pin function.

Bits PSEL[4:0] select the peripheral function assigned to each port pin.

The ISEL bit is set when a pin is used as an IRQ input pin. This setting can be used with the combination of the peripheral function, though IRQn (external pin interrupt) of the same number should not be enabled by two or more pins.

The ASEL bit is set when a pin is used as an analog pin. When switching a pin to analog using the ASEL bit, set the corresponding port mode register bit (PORTm.PMR) to “general I/O port” and the port direction register bit (PORTm.PDR) to “input”. The pin state cannot be read at this point. The PmnPFS register is protected by the write-protect register (PWPR). Modify the register after releasing the protection.

The ISEL bit to which IRQn is not specified is reserved. The ASEL bit to which analog input/output is not specified is reserved.

Table 18.2 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin				
	P00	P01	P02	P03	P04
00000b (initial value)	Hi-Z				
00001b	—	—	—	MTIOC1B	MTIOC1A
00011b	—	—	—	MTIOC1B#	MTIOC1A#
00101b	—	—	—	—	TMCI3
00110b	—	—	—	—	TMO0
00111b	CACREF	—	—	—	—
01001b	ADSS0	ADST1	ADST0	ADTRG0#	ADSS1
01010b	—	—	CTS1# RTS1# SS1#	—	—
10100b	—	—	—	GTIOC2B	GTIOC2A
10110b	—	—	—	GTIOC2B#	GTIOC2A#

—: Do not specify this value.

18.2.3 P1n Pin Function Control Register (P1nPFS) (n = 0 to 5)

Address(es): MPC.P10PFS 0008 C148h, MPC.P11PFS 0008 C149h, MPC.P12PFS 0008 C14Ah, MPC.P13PFS 0008 C14Bh, MPC.P14PFS 0008 C14Ch, MPC.P15PFS 0008 C14Dh



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.3.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin P10: IRQ0 input switch (64/52/48/44 pins) P11: IRQ1 input switch (64/52/48/44/32/24 pins) P12: IRQ7 input switch (64 pins) P13: IRQ4 input switch (64/52/48/44/32/24 pins) P14: IRQ6 input switch (64/52/48/44/32/24 pins)	R/W
b7	ASEL	Analog Function Select*1	0: Used other than as analog pin. 1: Used as analog pin. P10: AN103 (64/52/48/44 pins) P11: AN102/CMPC02 (64/52/48/44/32/24 pins) P12: AN104 (64 pins) P13: AN105/VREFL0/CMPC12 (64/52/48/44/32/24 pins) P14: AN106/VREFH0/CMPC22 (64/52/48/44/32/24 pins) P15: AN107 (64 pins)	R/W

Note 1. Using P13 and P14 for VREFL0 and VREFH0, respectively, requires setting of the ADHVREFCNT register in addition to setting this bit. For details, see section 30, 12-Bit A/D Converter (S12ADF).

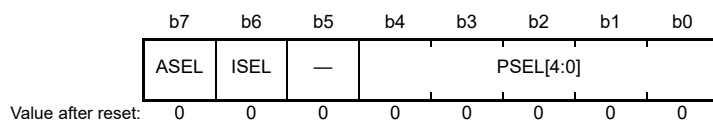
Table 18.3 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin					
	P10	P11	P12	P13	P14	P15
00000b (initial value)	Hi-Z					
00001b	—	MTIOC3A	—	—	—	—
00010b	MTCLKD	MTCLKC	—	—	—	—
00011b	—	MTIOC3A#	—	—	—	—
00100b	MTCLKD#	MTCLKC#	—	—	—	—
00101b	TMR13	TMO3	—	—	—	—
00110b	TMO1	TMO2	—	—	—	—
00111b	—	POE8#	—	—	—	—
01000b	—	MTIOC4C	—	—	—	—
01001b	—	—	—	—	ADST0	—
01010b	—	CTS1# RTS1# SS1#	—	—	—	—
01011b	—	TXD6 SMOSI6 SSDA6	—	—	—	—
01100b	—	TXD12 SMOSI12 SSDA12 TXDX12 SIOX12	—	—	—	—
01110b	—	MTIOC4C#	—	—	—	—

—: Do not specify this value.

18.2.4 P2n Pin Function Control Register (P2nPFS) (n = 2 to 6)

Address(es): MPC.P22PFS 0008 C152h, MPC.P23PFS 0008 C153h, MPC.P24PFS 0008 C154h, MPC.P25PFS 0008 C155h, MPC.P26PFS 0008 C156h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.4.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin P22: IRQ2 input switch (64/52/48/44 pins) P23: IRQ4 input switch (64/52/48/44 pins) P24: IRQ3 input switch (64/52/48/44 pins) P26: IRQ1 input switch (64/52/48/44/32/24 pins)	R/W
b7	ASEL	Analog Function Select	0: Used other than as analog pin. 1: Used as analog pin. P22: PGAOUT0 (64/52/48/44 pins)	R/W

Table 18.4 Register Settings for Input/Output Pin Function

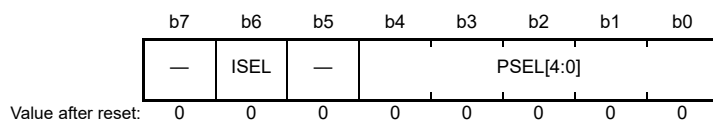
PSEL[4:0] Settings	Pin				
	P22	P23	P24	P25	P26
00000b (initial value)	Hi-Z				
00001b	MTIC5W	MTIC5V	MTIC5U	MTIOC0B	MTIOC3B
00010b	MTCLKB	MTCLKA	—	MTCLKD	—
00011b	MTIC5W#	MTIC5V#	MTIC5U#	MTIOC0B#	MTIOC3B#
00100b	MTCLKB#	MTCLKA#	—	MTCLKD#	—
00101b	TMRI2	TMO2	TMCI2	TMRI3	TMO3
00111b	—	CACREF	—	POE12#	—
01000b	—	—	—	MTIOC4B	—
01001b	ADSS1	ADSS0	CLKOUT	—	—
01010b	SCK5	TXD5 SMOSI5 SSDA5	RXD5 SMISO5 SSCL5	—	RXD6 SMISO6 SSCL6
01110b	—	—	—	MTIOC4B#	—
01111b	—	—	—	—	SCL0*1
10100b	—	GTIOC2A	GTIOC2B	—	GTIOC0A
10101b	GTADSM1	GTADSM0	GTCPPO0	—	—
10110b	—	GTIOC2A#	GTIOC2B#	—	GTIOC0A#
10111b	—	—	—	GTETRGA	—
11110b	COMP2	COMP1	COMP0	—	—

—: Do not specify this value.

Note 1. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

18.2.5 P3n Pin Function Control Register (P3nPFS) (n = 0 to 3, 6, 7)

Address(es): MPC.P30PFS 0008 C158h, MPC.P31PFS 0008 C159h, MPC.P32PFS 0008 C15Ah, MPC.P33PFS 0008 C15Bh, MPC.P36PFS 0008 C15Eh, MPC.P37PFS 0008 C15Fh



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.5.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Input Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin P36: IRQ3 input switch (64/52/48/44/32/24 pins) P37: IRQ0 input switch (64/52/48/32 pins)	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Table 18.5 Register Settings for Input/Output Pin Function

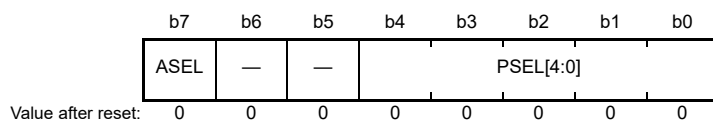
PSEL[4:0] Settings	Pin					
	P30	P31	P32	P33	P36	P37
00000b (initial value)	Hi-Z					
00001b	MTIOC0A	MTIOC4A	MTIOC3C	MTIOC3A	MTIOC0A	—
00010b	MTCLKC	—	MTCLKB	MTCLKA	MTIOC4A	—
00011b	MTIOC0A#	MTIOC4A#	MTIOC3C#	MTIOC3A#	MTIOC0A#	—
00100b	MTCLKC#	—	MTCLKB#	MTCLKA#	MTIOC4A#	—
00101b	—	TMCI3	—	—	—	—
00111b	POE11#	—	POE10#	—	POE12#	POE11#
01000b	—	—	MTIOC0D	MTIOC0C	MTIOC1A	—
01001b	—	—	—	—	ADST1	ADSS1
01010b	SCK6	TXD6 SMOSI6 SSDA6	—	—	—	—
01011b	—	—	—	—	RXD6 SMISO6 SSCL6	TXD6 SMOSI6 SSDA6
01100b	—	—	—	—	RXD12 SMISO12 SSCL12 RXDX12	TXD12 SMOSI12 SSDA12 TXDX12 SIOX12
01110b	—	—	MTIOC0D#	MTIOC0C#	MTIOC1A#	—
01111b	—	SDA0*1	—	—	—	—
10100b	—	GTIOC0B	—	—	GTIOC0A	—
10101b	—	GTIOC1A	—	—	GTCPP00	—
10110b	—	GTIOC0B#	—	—	GTIOC0A#	—
10111b	—	GTIOC1A#	—	—	—	—

—: Do not specify this value.

Note 1. This pin is not 5-V tolerant. For the input voltages of the 5-V tolerant and non-5-V tolerant pins, see the input voltage column in Recommended Operating Conditions (1) under section 38, Electrical Characteristics.

18.2.6 P4n Pin Function Control Register (P4nPFS) (n = 0 to 7)

Address(es): MPC.P40PFS 0008 C160h, MPC.P41PFS 0008 C161h, MPC.P42PFS 0008 C162h, MPC.P43PFS 0008 C163h, MPC.P44PFS 0008 C164h, MPC.P45PFS 0008 C165h, MPC.P46PFS 0008 C166h, MPC.P47PFS 0008 C167h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.6.	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	ASEL	Analog Function Select	0: Not used as an analog pin 1: Used as an analog pin*1 P40: AN000/PGAIN00/CMPC00/CMPC01 (64/52/48/44/32/24 pins) P41: AN001/PGAIN01/CMPC10/CMPC11 (64/52/48/44/32/24 pins) P42: AN002/PGAIN02/CMPC20/CMPC21 (64/52/48/44/32/24 pins) P43: AN003 (64/52/48/44 pins) P44: AN004/DA0/CMPC03/CVREFC0 (64/52/48/44/32 pins) P45: AN005 (64/52/48 pins) P46: AN006/CMPC13 (64/52/48/44/32 pins) P47: AN007/PGAGND (64/52/48/44 pins)	R/W

Note 1. When the pin is to be used for the DA0 or CVREFC0 analog function, set the P44PFS.PSEL[4:0] bits to 11101b and the P44PFS.ASEL bit to 1 to enable the analog switch. When the pin is to be used for CVREFC0, do not use DA0 as an output. When the pin is to be used for AN004 or CMPC03, set the P44PFS.PSEL[4:0] bits to a value other than 11101b.

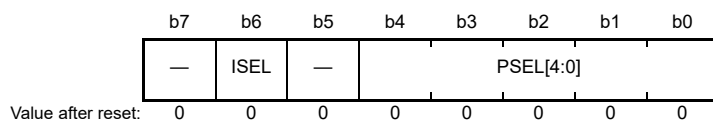
Table 18.6 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin							
	P40	P41	P42	P43	P44	P45	P46	P47
00000b (initial value)	Hi-Z							
01001b	—	—	—	—	ADSS1	—	ADSS0	—
10101b	—	—	—	—	GTADSM1	—	GTADSM0	—
11101b	—	—	—	—	DA0 CVREFC0	—	—	—

—: Do not specify this value.

18.2.7 P7n Pin Function Control Register (P7nPFS) (n = 0 to 6)

Address(es): MPC.P70PFS 0008 C178h, MPC.P71PFS 0008 C179h, MPC.P72PFS 0008 C17Ah, MPC.P73PFS 0008 C17Bh, MPC.P74PFS 0008 C17Ch, MPC.P75PFS 0008 C17Dh, MPC.P76PFS 0008 C17Eh



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.7.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Input Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin P70: IRQ5 (64/52/48 pins) P71: IRQ6 (64/52/48/44/32 pins) P72: IRQ7 (64/52/48/44/32 pins)	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

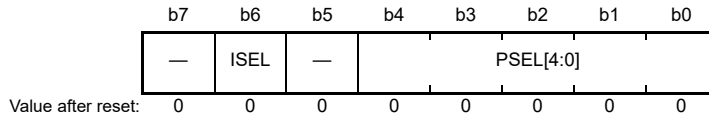
Table 18.7 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin						
	P70	P71	P72	P73	P74	P75	P76
00000b (initial value)	Hi-Z						
00001b	—	MTIOC3B	MTIOC4A	MTIOC4B	MTIOC3D	MTIOC4C	MTIOC4D
00011b	—	MTIOC3B#	MTIOC4A#	MTIOC4B#	MTIOC3D#	MTIOC4C#	MTIOC4D#
00101b	—	—	—	TMRI3	TMO2	TMCI2	TMRI2
00111b	POE0#	—	—	—	—	POE8#	CACREF
01010b	SCK1	TXD1 SMOSI1 SSDA1	RXD1 SMISO1 SSCL1	SCK6	CTS6# RTS6# SS6#	—	—
01100b	—	—	—	—	CTS12# RTS12# SS12#	SCK12	RXD12 SMISO12 SSCL12 RXDX12
10100b	—	GTIOC0A	GTIOC1A	GTIOC2A	GTIOC0B	GTIOC1B	GTIOC2B
10101b	GTCPP00	—	—	—	—	—	—
10110b	—	GTIOC0A#	GTIOC1A#	GTIOC2A#	GTIOC0B#	GTIOC1B#	GTIOC2B#

—: Do not specify this value.

18.2.8 P9n Pin Function Control Register (P9nPFS) (n = 0 to 7)

Address(es): MPC.P90PFS 0008 C188h, MPC.P91PFS 0008 C189h, MPC.P92PFS 0008 C18Ah, MPC.P93PFS 0008 C18Bh, MPC.P94PFS 0008 C18Ch, MPC.P95PFS 0008 C18Dh, MPC.P96PFS 0008 C18Eh, MPC.P97PFS 0008 C18Fh



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.8.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Input Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin P90: IRQ6 (64 pins) P92: IRQ0 (64/52/48/44/32 pins) P93: IRQ1 (64/52/48/44/32 pins) P94: IRQ4 (64/52/48/44 pins) P97: IRQ2 (64/52/48/44/32/24 pins)	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Table 18.8 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin							
	P90	P91	P92	P93	P94	P95	P96	P97
00000b (initial value)	Hi-Z							
00001b	MTIC5V	MTIC5U	MTIOC0B	MTIOC0C	MTIOC1A	MTIOC1B	MTIOC0D	MTIOC0C
00010b	—	—	MTIOC1A	MTIOC2B	MTIOC2B	MTIOC2A	MTCLKB	MTCLKA
00011b	MTIC5V#	MTIC5U#	MTIOC0B#	MTIOC0C#	MTIOC1A#	MTIOC1B#	MTIOC0D#	MTIOC0C#
00100b	—	—	MTIOC1A#	MTIOC2B#	MTIOC2B#	MTIOC2A#	MTCLKB#	MTCLKA#
00101b	—	TMC11	TMR11	TMO1	—	—	TMO1	TMC11
00111b	—	—	—	—	—	—	POE0#	—
01000b	—	—	—	MTCLKA	MTIOC3B	MTIOC4A	MTIOC4B	MTIOC3D
01001b	—	—	ADTRG0#	ADTRG1#	—	—	CLKOUT	ADSM1
01010b	—	—	SCK5	—	CTS5# RTS5# SS5#	—	SCK5	RXD5 SMISO5 SSCL5
01100b	—	—	SCK12	RXD12 SMISO12 SSCL12 RXDX12	TXD12 SMOSI12 SSDA12 TXDX12 SIOX12	CTS12# RTS12# SS12#	TXD12 SMOSI12 SSDA12 TXDX12 SIOX12	—
01110b	—	—	—	MTCLKA#	MTIOC3B#	MTIOC4A#	MTIOC4B#	MTIOC3D#
01111b	—	—	—	—	—	—	—	SCL0
10100b	—	—	GTIOC1A	GTIOC1B	GTIOC0A	GTIOC1A	GTIOC2A	GTIOC0B
10101b	—	—	—	—	—	—	—	GTADSM1
10110b	—	—	GTIOC1A#	GTIOC1B#	GTIOC0A#	GTIOC1A#	GTIOC2A#	GTIOC0B#
10111b	—	—	—	—	—	—	—	GTETRGA

—: Do not specify this value.

18.2.9 PB_n Pin Function Control Register (PB_nPFS) (n = 0 to 7)

Address(es): MPC.PB0PFS 0008 C198h, MPC.PB1PFS 0008 C199h, MPC.PB2PFS 0008 C19Ah, MPC.PB3PFS 0008 C19Bh, MPC.PB4PFS 0008 C19Ch, MPC.PB5PFS 0008 C19Dh, MPC.PB6PFS 0008 C19Eh, MPC.PB7PFS 0008 C19Fh



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.9.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Input Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin PB1: IRQ2 (64/52/48/44/32 pins) PB3: IRQ3 (64/52/48/44 pins) PB5: IRQ7 (64/52/48/44/32/24 pins) PB6: IRQ5 (64/52/48/44/32/24 pins)	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

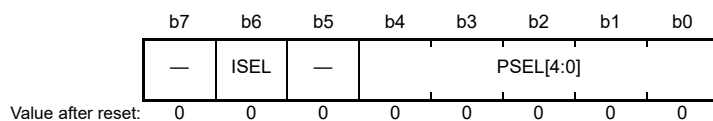
Table 18.9 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin							
	PB0	PB1	PB2	PB3	PB4	PB5	PB6	PB7
00000b (initial value)	Hi-Z							
00001b	MTIOC0B	MTIOC0A	MTIC5W	—	—	MTIOC1B	MTIOC3C	—
00010b	MTCLKC	—	—	—	—	MTIOC3A	MTCLKD	—
00011b	MTIOC0B#	MTIOC0A#	MTIC5W#	—	—	MTIOC1B#	MTIOC3C#	—
00100b	MTCLKC#	—	—	—	—	MTIOC3A#	MTCLKD#	—
00101b	TMR1	—	—	—	—	TMO0	TMCIO	—
00111b	—	CACREF	—	POE8#	—	POE12#	POE11#	—
01000b	MTIOC4C	MTIOC4D	—	—	—	MTIOC2A	MTIOC2B	—
01001b	ADSM0	—	—	ADTRG1#	ADTRG0#	ADSS1	ADSS0	—
01010b	TXD5 SMOSI5 SSDA5	SCK5	SCK6	TXD6 SMOSI6 SSDA6	RXD6 SMISO6 SSCL6	TXD5 SMOSI5 SSDA5	RXD5 SMISO5 SSCL5	SCK5
01011b	SCK1	—	—	—	—	TXD1 SMOSI1 SSDA1	RXD1 SMISO1 SSCL1	CTS6# RTS6# SS6#
01100b	—	SCK12	—	—	—	—	—	—
01110b	MTIOC4C#	MTIOC4D#	—	—	—	MTIOC2A#	MTIOC2B#	—
01111b	SDA0	—	—	—	—	—	—	—
10100b	GTIOC1B	GTIOC2B	—	—	—	GTIOC1A	GTIOC2B	—
10101b	GTADSM0	—	—	—	—	GTIOC2A	GTCPP00	—
10110b	GTIOC1B#	GTIOC2B#	—	—	—	GTIOC1A#	GTIOC2B#	—
10111b	—	—	—	—	—	GTIOC2A#	—	—

—: Do not specify this value.

18.2.10 PDn Pin Function Control Register (PDnPFS) (n = 3 to 7)

Address(es): MPC.PD3PFS 0008 C1ABh, MPC.PD4PFS 0008 C1ACh, MPC.PD5PFS 0008 C1ADh, MPC.PD6PFS 0008 C1AEh, MPC.PD7PFS 0008 C1AFh



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.10.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Input Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin PD4: IRQ2 (64/52/48/44 pins) PD5: IRQ3 (64/52/48/44 pins) PD6: IRQ5 (64/52/48/44 pins)	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

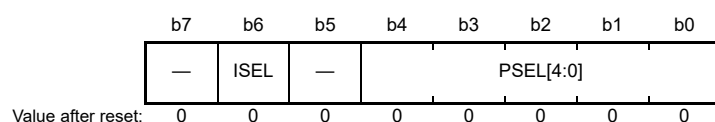
Table 18.10 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin				
	PD3	PD4	PD5	PD6	PD7
00000b (initial value)	Hi-Z				
00001b	MTIOC0A	MTIOC0B	MTIOC0C	MTIOC0D	—
00010b	MTIOC4A	MTIOC4B	MTIOC4C	MTIOC4D	—
00011b	MTIOC0A#	MTIOC0B#	MTIOC0C#	MTIOC0D#	—
00100b	MTIOC4A#	MTIOC4B#	MTIOC4C#	MTIOC4D#	—
00101b	TMO0	TMCIO	TMRI0	TMO1	TMRI1
01001b	—	—	ADSS0	ADST0	—
01010b	TXD1 SMOSI1 SSDA1	SCK1	RXD1 SMISO1 SSCL1	CTS1# RTS1# SS1#	—
10100b	GTIOC1A	GTIOC1B	GTIOC2A	GTIOC2B	—
10110b	GTIOC1A#	GTIOC1B#	GTIOC2A#	GTIOC2B#	—

—: Do not specify this value.

18.2.11 PEn Pin Function Control Register (PEnPFS) (n = 2)

Address(es): MPC.PE2PFS 0008 C1B2h



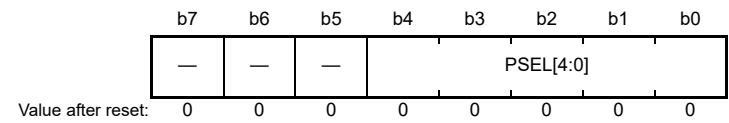
Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.11.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	ISEL	Interrupt Input Function Select	0: Not used as IRQn input pin 1: Used as IRQn input pin PE2: IRQ0 (64/52/48/44/32/24 pins)	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Table 18.11 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin
	PE2
00000b (initial value)	Hi-Z
00101b	TMCI1
00110b	TMRI0
00111b	POE10#
10111b	GTETRG

18.2.12 P_G_n Pin Function Control Register (P_G_nPFS) (n = 7)

Address(es): MPC.PG7PFS 0008 C1C7h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	PSEL[4:0]	Pin Function Select	These bits select the peripheral function. For individual pin functions, see Table 18.12.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Table 18.12 Register Settings for Input/Output Pin Function

PSEL[4:0] Settings	Pin
	PG7
00000b (initial value)	Hi-Z
01001b	ADST0
01010b	CTS5# RTS5# SS5#
01011b	SCK6
01100b	SCK12

18.3 Usage Notes

18.3.1 Procedure for Specifying Input/Output Pin Function

Use the following procedure to specify the input/output pin functions.

- (1) Clear the port mode register (PMR) to 0 to select the general I/O port function.
- (2) Specify the assignments of input/output signals for peripheral functions to the desired pins.
- (3) Enable writing to the Pmn pin function control register (PmnPFS) through the write-protect register (PWPR) setting. (m = 0 to 4, 7, 9, B, D, E, G; n = 0 to 7)
- (4) Specify the input/output function for the pin through the PSEL[4:0] bit settings in the PmnPFS register.
- (5) Clear the PFSWE bit in the PWPR register to 0 to disable writing to the PmnPFS register.
- (6) Set the PMR to 1 as necessary to switch to the selected input/output function for the pin.

18.3.2 Notes on Pmn Pin Function Control Register (PmnPFS) Setting

- (1) Settings of the PmnPFS register should be made only while the port mode register (PMR) for the target pin is cleared to 0. If a Pmn pin function control register is set while the PMR register is 1, unexpected edges may be input through the input pin or unexpected pulses are output through the output pin.
- (2) Only the allowed values (functions) should be specified in the PmnPFS registers. If a value that is not allowed for the register is specified, correct operation is not guaranteed.
- (3) Do not assign a single function to multiple pins through the multi-function controller settings.
- (4) Pins of ports 1 and 4 are multiplexed with the analog input/output functions for the A/D converter, PGA, comparator, and D/A converter, and the reference voltage function for the A/D converter. When a pin is to be used as an analog input or output, or as the reference voltage for the A/D converter, avoid loss of precision by setting the corresponding bits of the PMR register and of the port direction register (PDR) to 0, i.e. configuring the pin as a general-purpose input, and setting the PmnPFS.ASEL bit to 1. When a pin is to be used as the reference voltage, in addition to the above settings, respectively set the ADHVREFCNT.LVSEL and ADHVREFCNT.HVSEL[1:0] bits to 1 and 01b.
- (5) Pins P36 and P37 are respectively multiplexed with the EXTAL and XTAL functions. When P36 pin is to be used as EXTAL, set the MOSCCR.MOSTP bit to 0. When P37 pin is to be used as XTAL, set the MOSCCR.MOSTP and MOFCR.MOSEL bits to 0. When the settings of the MOSCCR.MOSTP and MOFCR.MOSEL bits are respectively 0 and 1, the P37 pin can be used as an input port.
- (6) Points to note regarding the PMR register, PDR register, and PmnPFS register settings for pins that have multiplexed pin functions are listed in Table 18.13.

Table 18.13 Register Settings

Item			PmnPFS			Point to Note		
			PMR.Bn	PDR.Bn	ASEL		ISEL	PSEL[4:0]
After a reset			0	0	0	0	00000b	Pins function as general input port pins after release from the reset state.
General input ports			0	0	0	0/1	x	Set the ISEL bit to 1 if these are multiplexed with interrupt inputs.
General output ports			0	1	0	0	x	
Peripheral functions			1	x	0	0/1	Peripheral functions (see Table 18.2 to Table 18.12)	Set the ISEL bit to 1 if these are multiplexed with interrupt inputs.
Interrupt inputs			0	0	0	1	x	
NMI			x	x	x	x	x	Register settings are not required.
Analog inputs and outputs	P44	DA0/ CVREFC0	0	0	1	x	11101b	Set the PDR.Bn bit to 0 to select the general-purpose input port pin function (de-selecting the output buffer for use with general-purpose input/output). When the pin is to be used for the DA0 or CVREFC0 analog function, set the P44PFS.PSEL[4:0] bits to 11101b and the P44PFS.ASEL bit to 1 to enable the analog switch. When the pin is to be used for CVREFC0, do not use DA0 as an output.
		AN004/ CMPC03	0	0	1	x	Other than 11101b	Set the PDR.Bn bit to 0 to select the general-purpose input port pin function (de-selecting the output buffer for use with general-purpose input/output). When the pin is to be used for AN004 or CMPC03, set the P44PFS.PSEL[4:0] bits to a value other than 11101b.
	Other than P44	0	0	1	x*1	x	Set the PDR.Bn bit to 0 to select the general-purpose input port pin function (de-selecting the output buffer for use with general-purpose input/output).	
P13/VREFL0, P14/VREFH0			0	0	1	x*1	x	Set the PDR.Bn bit to 0 to select the general-purpose input port pin function (de-selecting the output buffer for use with general-purpose input/output). When the pin is to be used as VREFL0 or VREFH0, respectively set the ADHVREFCNT.LVSEL and ADHVREFCNT.HVSEL[1:0] bits to 1 and 01b.
P36/EXTAL			0	0	x	0	x	Set the PDR.Bn bit to 0 to select the general-purpose input port pin function (de-selecting the output buffer for use with general-purpose input/output). When the pin is to be used as EXTAL, set the MOSCCR.MOSTP bit to 0.
P37/XTAL			0	0	x	0	x	Set the PDR.Bn bit to 0 to select the general-purpose input port pin function (de-selecting the output buffer for use with general-purpose input/output). When the pin is to be used as XTAL, set the MOSCCR.MOSTP and MOFCR.MOSEL bits to 0.

x: Setting not required.

0/1: Setting the PmnPFS.ISEL bit to 0 makes the pin incapable of functioning as an IRQ pin.

Setting the PmnPFS.ISEL bit to 1 makes the pin capable of functioning as an IRQ pin (if the IRQ is selected from the multiplexed functions).

Note 1. The pin does not function as the IRQn input pin even if the PmnPFS.ISEL bit is set to 1.

Note: The pin state is readable when the PmnPFS.ASEL bit is 0.

- If the value of the PmnPFS.PSEL[4:0] bits is to be changed, do so while the PMR.Bn bit is 0.
- If an RIIC function is assigned to a port pin, clear the PCR.Bn (to 0); pulling up is automatically turned off for outputs from peripheral modules other than the RIIC.

18.3.3 Note on Using Analog Functions

When an analog function is in use, configure the pin as a general-purpose input by setting the given bits of the port mode register (PMR), of the port direction register (PDR), and the pull-up control register (PCR) to 0, and then set the ASEL bit in the Pmn pin function control register (PmnPFS) to 1.

18.3.4 Note on P97 Pin Input Level

P97 input level is specified to TTL when SCL is selected in the P97PFS.PSEL[4:0] bits and SMBus is selected in the ICMR3.SMBS bit in RIIC. At this time, input levels of the P97 port read and the IRQ2 also become TTL.

18.3.5 Notes on Controlling the Switch to General I/O Port Pin Operation by POE3

On the generation of requests for disabling of output specified by the POE3, the pins for which the setting is 1 in the corresponding PMMCRn register (n = 0 to 2) of the POE3 are switched to general I/O port pins. Set the bits of the corresponding POECRn register (n = 1 to 3) to 0 beforehand.

18.3.6 Notes on Inverting Input/Output Function for the MTU and GPTW Input/Output Pins

The MTU and GPTW input/output pins shown in Table 18.14 can invert and output the signal captured by inverting the input signal when the PmnPFS.PSEL[4:0] bit for the target pins is set. Setting for switching the state between the normal input/output and the inverting input/output must be done while the PMR register for the target pins is set as 0.

Table 18.14 MTU and GPTW Input/Output Pins

Module/Function	Channel	Normal Input/Output	Inverting Input/Output
Multi-function timer unit 3	MTU0	MTIOC0A	MTIOC0A#
		MTIOC0B	MTIOC0B#
		MTIOC0C	MTIOC0C#
		MTIOC0D	MTIOC0D#
	MTU1	MTIOC1A	MTIOC1A#
		MTIOC1B	MTIOC1B#
	MTU2	MTIOC2A	MTIOC2A#
		MTIOC2B	MTIOC2B#
	MTU3	MTIOC3A	MTIOC3A#
		MTIOC3B	MTIOC3B#
		MTIOC3C	MTIOC3C#
		MTIOC3D	MTIOC3D#
	MTU4	MTIOC4A	MTIOC4A#
		MTIOC4B	MTIOC4B#
		MTIOC4C	MTIOC4C#
		MTIOC4D	MTIOC4D#
	MTU5	MTIC5U	MTIC5U#
		MTIC5V	MTIC5V#
		MTIC5W	MTIC5W#
General PWM timer	GPTW0	GTIOC0A	GTIOC0A#
		GTIOC0B	GTIOC0B#
	GPTW1	GTIOC1A	GTIOC1A#
		GTIOC1B	GTIOC1B#
	GPTW2	GTIOC2A	GTIOC2A#
		GTIOC2B	GTIOC2B#

19. Multi-Function Timer Pulse Unit 3 (MTU3c)

19.1 Overview

This MCU has an on-chip multi-function timer pulse unit 3 (MTU3c), consisting of six 16-bit timer channels.

Table 19.1 shows the specifications of the MTU and Table 19.2 lists the functions of the MTU. Figure 19.1 shows a block diagram of the MTU.

Table 19.1 MTU Specifications

Item	Description
Pulse input/output	16 lines max.
Pulse input	3 lines
Count clock	11 clocks for each channel (14 clocks for MTU0, 12 clocks for MTU2, 10 clocks for MTU5, and four clocks for MTU1 & MTU2 (LWA = 1))
Available operations	[MTU0 to MTU4] • Waveform output on compare match • Input capture function (noise filter setting available) • Counter-clearing operation • Simultaneous writing to multiple timer counters (TCNT) • Simultaneous clearing on compare match or input capture • Simultaneous input and output to registers in synchronization with counter operations • Up to 12-phase PWM output in combination with synchronous operation
	[MTU0, MTU3, MTU4] • Buffer operation specifiable
	[MTU1, MTU2] • Phase counting mode can be specified independently • 32-bit phase counting mode can be specified for interlocked operation of MTU1 and MTU2 (when TMDR3.LWA = 1) • Cascade connection operation available
	[MTU3, MTU4] • Through interlocked operation of MTU3/4, the positive and negative signals in six phases can be output in complementary PWM and reset-synchronized PWM operation. • In complementary PWM mode, transfer of values from buffer registers to temporary registers on crests or troughs of the timer-counter values or writing to the buffer registers (MTU4.TGRD) • Double-buffering selectable in complementary PWM mode
	[MTU3, MTU4] • Through interlocking with MTU0, a mode for driving AC synchronous motors (brushless DC motors) by using complementary PWM output and reset-synchronized PWM output is settable and allows the selection of two types of waveform output (chopping or level)
	[MTU5] • Capable of operation as a dead-time compensation counter
	[MTU0, MTU4] GPTW can start counting in synchronization with the counting start of MTU0 or MTU4.
Interrupt skipping function	• In complementary PWM mode, interrupts on crests and troughs of counter values and triggers to start conversion by the A/D converter can be skipped
Interrupt sources	28 sources
Buffer operation	Automatic transfer of register data (transfer from the buffer register to the timer register)
Trigger generation	A/D conversion start triggers can be generated
	A/D conversion start request delaying function enables A/D converter to be started with any desired timing and to be synchronized with PWM output
Low power consumption function	Module stop mode can be set

Table 19.2 MTU Functions (1/2)

Item	MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU5
Count clock	PCLKB/1 PCLKB/2 PCLKB/4 PCLKB/8 PCLKB/16 PCLKB/32 PCLKB/64 PCLKB/256 PCLKB/1024 MTCLKA MTCLKB MTCLKC MTCLKD MTIOC1A	PCLKB/1 PCLKB/2 PCLKB/4 PCLKB/8 PCLKB/16 PCLKB/32 PCLKB/64 PCLKB/256 PCLKB/1024 MTCLKA MTCLKB	PCLKB/1 PCLKB/2 PCLKB/4 PCLKB/8 PCLKB/16 PCLKB/32 PCLKB/64 PCLKB/256 PCLKB/1024 MTCLKA MTCLKB MTCLKC	MTCLKA MTCLKB MTCLKC MTCLKD	PCLKB/1 PCLKB/2 PCLKB/4 PCLKB/8 PCLKB/16 PCLKB/32 PCLKB/64 PCLKB/256 PCLKB/1024 MTCLKA MTCLKB	PCLKB/1 PCLKB/2 PCLKB/4 PCLKB/8 PCLKB/16 PCLKB/32 PCLKB/64 PCLKB/256 PCLKB/1024 MTCLKA MTCLKB	PCLKB/1 PCLKB/2 PCLKB/4 PCLKB/8 PCLKB/16 PCLKB/32 PCLKB/64 PCLKB/256 PCLKB/1024 MTIOC1A
External clock for phase counting mode	—	MTCLKA MTCLKB	MTCLKA MTCLKB MTCLKC MTCLKD	MTCLKA MTCLKB MTCLKC MTCLKD	—	—	—
General registers (TGR)	TGRA TGRB TGRE	TGRA TGRB	TGRA TGRB	TGRALW TGRBLW	TGRA TGRB	TGRA TGRB	TGRU TGRV TGRW
General registers/ buffer registers	TGRC TGRD TGRF	—	—	—	TGRC TGRD TGRE	TGRC TGRD TGRE TGRF	—
I/O pins	MTIOC0A MTIOC0B MTIOC0C MTIOC0D	MTIOC1A MTIOC1B	MTIOC2A MTIOC2B	MTIOC1A MTIOC1B	MTIOC3A MTIOC3B MTIOC3C MTIOC3D	MTIOC4A MTIOC4B MTIOC4C MTIOC4D	MTIC5U MTIC5V MTIC5W
Counter clear function	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGRALW/TGRBLW compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	✓	✓	✓	✓	✓	—
	1 output	✓	✓	✓	✓	✓	—
	Toggle output	✓	✓	✓	✓	✓	—
Input capture function	✓	✓	✓	✓*1	✓	✓	✓
Synchronous operation	✓	✓	✓	—	✓	✓	—
PWM mode 1	✓	✓	✓	—	✓	✓	—
PWM mode 2	✓	✓	✓	—	—	—	—
Complementary PWM mode	—	—	—	—	✓	✓	—
Reset-synchronized PWM mode	—	—	—	—	✓	✓	—
AC synchronous motor drive mode	✓	—	—	—	✓	✓	—
Phase counting mode	—	✓	✓	✓	—	—	—
Buffer operation	✓	—	—	—	✓	✓	—
Dead time compensation counter function	—	—	—	—	—	—	✓
DTC trigger sources	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGRALW/TGRBLW compare match or input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow/underflow*2	TGR compare match or input capture
A/D conversion start trigger	TGRA compare match or input capture TGRE compare match	TGRA compare match or input capture	TGRA compare match or input capture	TGRALW compare match or input capture	TGRA compare match or input capture	TGRA compare match or input capture, or TCNT underflow (trough) in complementary PWM mode	—

Table 19.2 MTU Functions (2/2)

Item	MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU5
Interrupt sources	Seven sources • Compare match or input capture 0A • Compare match or input capture 0B • Compare match or input capture 0C • Compare match or input capture 0D • Compare match 0E • Compare match 0F • Overflow	Four sources • Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow	Four sources • Compare match or input capture 2A • Compare match or input capture 2B • Overflow • Underflow	Four sources • Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow	Five sources • Compare match or input capture 3A • Compare match or input capture 3B • Compare match or input capture 3C • Compare match or input capture 3D • Overflow	Five sources • Compare match or input capture 4A • Compare match or input capture 4B • Compare match or input capture 4C • Compare match or input capture 4D • Overflow or underflow*2	Three sources • Compare match or input capture 5U • Compare match or input capture 5V • Compare match or input capture 5W
A/D conversion start request delaying function	—	—	—	—	—	A/D conversion start request at a match between TADCORA and TCNT or A/D conversion start request at a match between TADCORB and TCNT	—
Interrupt skipping 1	—	—	—	—	Skips TGRA compare match interrupts	Skips TCIV interrupts	—
Interrupt skipping 2	—	—	—	—	—	Skipping in compare count between TADCORA and TCNT, and TADCORB and TCNT	—
Module stop function	MSTPCRA.MSTPA9*3						

✓: Possible —: Not possible

Note 1. When LWA is 1, the TGRALW capture source can be selected from either of the following: an input from MTIOC1A or MTU0.TGRA compare match/input capture event.

The TGRBLW capture source can be selected from any of the following: an input from MTIOC1B, MTU0.TGRC compare match/input capture event.

Note 2. Underflow is available only in complementary PWM mode.

Note 3. For details on the module stop function, refer to section 11, Low Power Consumption.

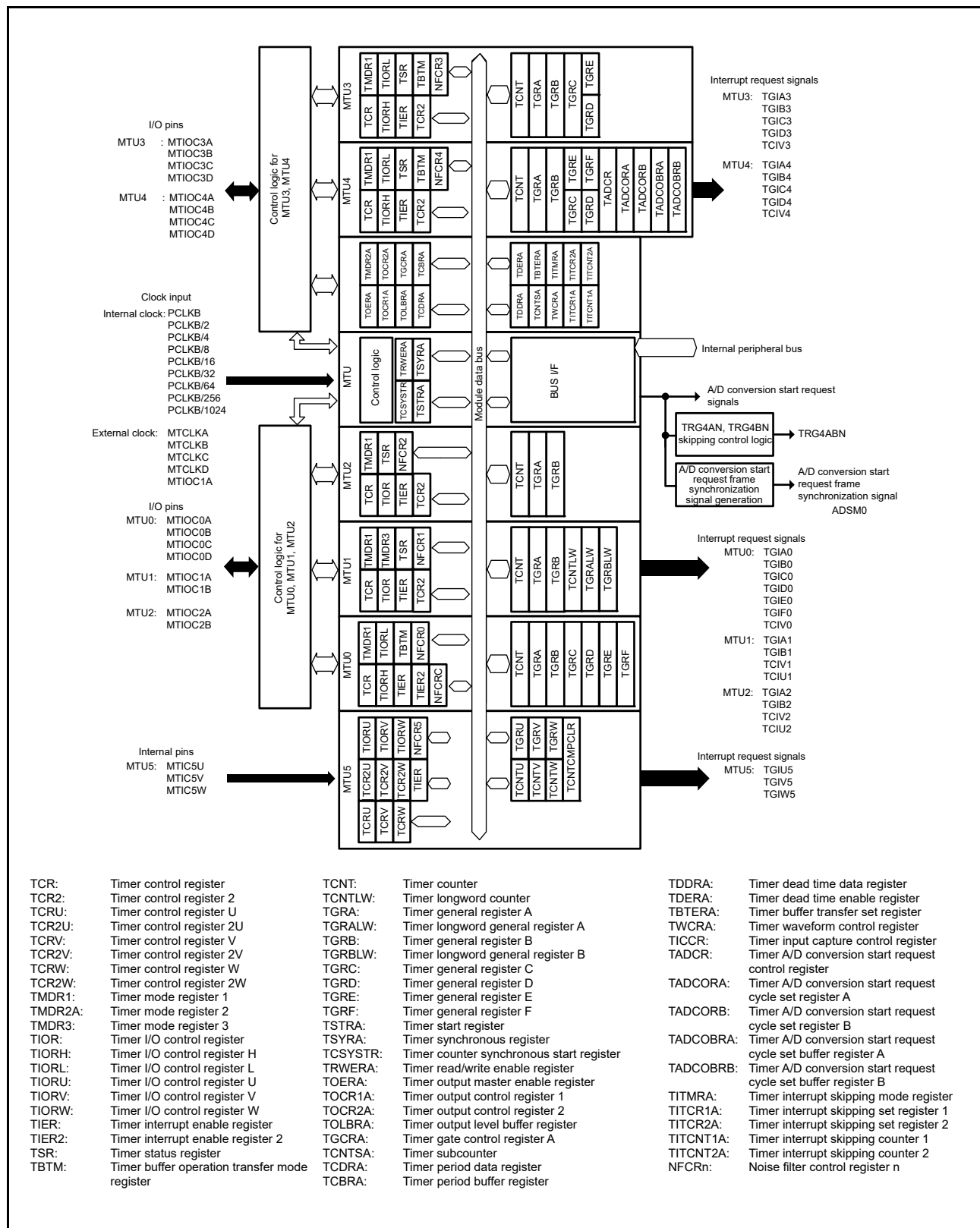


Figure 19.1 Block Diagram of MTU (MTU0 to MTU5)

Table 19.3 shows the configuration of pins for the MTU.

Table 19.3 Pin Configuration of the MTU

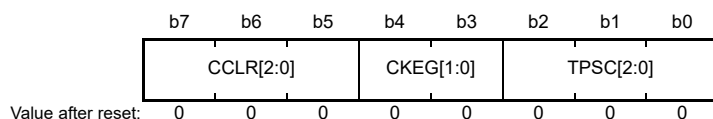
Channel	Pin Name	I/O	Function
MTU	MTCLKA	Input	External clock A input pin (MTU1 and MTU2 phase counting mode A phase input)
	MTCLKB	Input	External clock B input pin (MTU1 and MTU2 phase counting mode B phase input)
	MTCLKC	Input	External clock C input pin (MTU2 phase counting mode A phase input)
	MTCLKD	Input	External clock D input pin (MTU2 phase counting mode B phase input)
	ADSM0	Output	A/D conversion start request frame synchronization signal 0 output pin
	ADSM1	Output	A/D conversion start request frame synchronization signal 1 output pin
MTU0	MTIOC0A	I/O	MTU0 TGRA input capture input/output compare output/PWM output pin
	MTIOC0B	I/O	MTU0 TGRB input capture input/output compare output/PWM output pin
	MTIOC0C	I/O	MTU0 TGRC input capture input/output compare output/PWM output pin
	MTIOC0D	I/O	MTU0 TGRD input capture input/output compare output/PWM output pin
MTU1	MTIOC1A	I/O	MTU1 TGRA input capture input/output compare output/PWM output pin
	MTIOC1B	I/O	MTU1 TGRB input capture input/output compare output/PWM output pin
MTU2	MTIOC2A	I/O	MTU2 TGRA input capture input/output compare output/PWM output pin
	MTIOC2B	I/O	MTU2 TGRB input capture input/output compare output/PWM output pin
MTU3	MTIOC3A	I/O	MTU3 TGRA input capture input/output compare output/PWM output pin
	MTIOC3B	I/O	MTU3 TGRB input capture input/output compare output/PWM output pin
	MTIOC3C	I/O	MTU3 TGRC input capture input/output compare output/PWM output pin
	MTIOC3D	I/O	MTU3 TGRD input capture input/output compare output/PWM output pin
MTU4	MTIOC4A	I/O	MTU4 TGRA input capture input/output compare output/PWM output pin
	MTIOC4B	I/O	MTU4 TGRB input capture input/output compare output/PWM output pin
	MTIOC4C	I/O	MTU4 TGRC input capture input/output compare output/PWM output pin
	MTIOC4D	I/O	MTU4 TGRD input capture input/output compare output/PWM output pin
MTU5	MTIC5U	Input	MTU5 TGRU input capture input/external pulse input pin
	MTIC5V	Input	MTU5 TGRV input capture input/external pulse input pin
	MTIC5W	Input	MTU5 TGRW input capture input/external pulse input pin

19.2 Register Descriptions

19.2.1 Timer Control Register (TCR)

- MTU0.TCR, MTU1.TCR, MTU2.TCR, MTU3.TCR, MTU4.TCR

Address(es): MTU0.TCR 0009 5300h, MTU1.TCR 0009 5380h, MTU2.TCR 0009 5400h, MTU3.TCR 0009 5200h, MTU4.TCR 0009 5201h



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TPSC[2:0]	Time Prescaler Select	Refer to Table 19.6 to Table 19.9.	R/W
b4, b3	CKEG[1:0]	Clock Edge Select	b4 b3 0 0: Count at rising edge 0 1: Count at falling edge 1 x: Count at both edges	R/W
b7 to b5	CCLR[2:0]	Counter Clear Source Select	Refer to Table 19.4 and Table 19.5.	R/W

x: Don't care

The TCR register controls the TCNT operation for each channel in combination with the TCR2 register. The MTU has a total of eight TCR registers, one each for MTU0 to MTU4 and three (TCRU, TCRV, and TCRW) for MTU5. TCR values should be specified only while TCNT operation is stopped.

TPSC[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. The count clock source can be selected independently for each channel. Refer to Table 19.6 to Table 19.9 for details.

CKEG[1:0] Bits (Clock Edge Select)

These bits select the clock edge, including the MTIOC1A pin. When the internal clock is counted at both edges, the count clock period is halved (e.g. PCLKB/4 at both edges = PCLKB/2 at rising edge). If phase counting mode is used on MTU1 and MTU2, the setting of these bits is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the count clock source is PCLKB/2 or slower. When PCLKB/1 or the overflow/underflow in another channel is selected for the count clock source, a value can be written to these bits but counter operation compiles with the initial value.

CCLR[2:0] Bits (Counter Clear Source Select)

These bits select the TCNT counter clearing source. Refer to Table 19.4 and Table 19.5 for details.

Table 19.4 CCLR[2:0] (MTU0, MTU3, MTU4)

	Bit 7	Bit 6	Bit 5	
Channel	CCLR[2]	CCLR[1]	CCLR[0]	Description
MTU0	0	0	0	TCNT clearing disabled
MTU3	0	0	1	TCNT cleared by TGRA compare match/input capture
MTU4	0	1	0	TCNT cleared by TGRB compare match/input capture
	0	1	1	TCNT cleared by counter clearing in another channel performing synchronous clearing/synchronous operation*1
	1	0	0	TCNT clearing disabled
	1	0	1	TCNT cleared by TGRD compare match/input capture*2
	1	1	0	TCNT cleared by TGRD compare match/input capture*2
	1	1	1	TCNT cleared by counter clearing in another channel performing synchronous clearing/synchronous operation*1

Note 1. Synchronous operation is selected by setting the TSYRA.SYNCn bit to 1

Note 2. When TGRD or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority and compare match/input capture does not occur.

Table 19.5 CCLR[2:0] (MTU1 and MTU2)

	Bit 7	Bit 6	Bit 5	
Channel	Reserved*2	CCLR[1]	CCLR[0]	Description
MTU1	0	0	0	TCNT clearing disabled
MTU2	0	0	1	TCNT cleared by TGRA compare match/input capture (when LWA = 0) TCNTLW cleared by TGRALW compare match/input capture (when LWA = 1)
	0	1	0	TCNT cleared by TGRB compare match/input capture (when LWA = 0) TCNTLW cleared by TGRBLW compare match/input capture (when LWA = 1)
	0	1	1	TCNT cleared by counter clearing in another channel performing synchronous clearing/synchronous operation*1

Note 1. Synchronous operation is selected by setting the TSYRA.SYNCn bit to 1.

Note 2. Bit 7 is reserved in MTU1 and MTU2. It is read as 0. The write value is ignored.

- MTU5.TCRU, MTU5.TCRV, MTU5.TCRW

Address(es): MTU5.TCRU 0009 5484h, MTU5.TCRV 0009 5494h, MTU5.TCRW 0009 54A4h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TPSC[1:0]	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TPSC[1:0]	Time Prescaler Select	Refer to Table 19.10.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

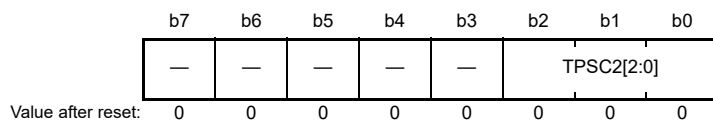
TPSC[1:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. Refer to Table 19.10 for details.

19.2.2 Timer Control Register 2 (TCR2)

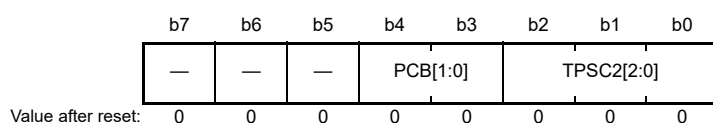
- MTU0.TCR2, MTU3.TCR2, MTU4.TCR2

Address(es): MTU0.TCR2 0009 5328h, MTU3.TCR2 0009 524Ch, MTU4.TCR2 0009 524Dh



- MTU1.TCR2, MTU2.TCR2

Address(es): MTU1.TCR2 0009 5394h, MTU2.TCR2 0009 540Ch



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TPSC2[2:0]	Time Prescaler Select	Refer to Table 19.6 to Table 19.9.	R/W
b4, b3	PCB[1:0]	Phase Counting Mode Function Expansion Control	Functional Expansion Control for Phase Counting Modes 2, 3, and 5	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The TCR2 register controls the TCNT operation for each channel in combination with the TCR register. The MTU has a total of eight TCR2 registers, one each for MTU0 to MTU4 and three (TCR2U, TCR2V, and TCR2W) for MTU5. TCR2 values should be specified only while TCNT operation is stopped.

TPSC2[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. The count clock source can be selected independently for each channel. Refer to Table 19.6 to Table 19.9 for details.

PCB[1:0] Bits (Phase Counting Mode Function Expansion Control)

These bits control extended functions for phase counting mode 2, 3, and 5 in MTU1 and MTU2. Refer to section 19.3.6, Phase Counting Mode.

- MTU5.TCR2U, MTU5.TCR2V, MTU5.TCR2W

Address(es): MTU5.TCR2U 0009 5485h, MTU5.TCR2V 0009 5495h, MTU5.TCR2W 0009 54A5h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	CKEG[1:0]		TPSC2[2:0]		
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TPSC2[2:0]	Time Prescaler Select	Refer to Table 19.10.	R/W
b4, b3	CKEG[1:0]	Clock Edge Select	b4 b3 0 0: Counts at the rising edge. 0 1: Counts at the falling edge. 1 x: Counts at both edges.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

x: Don't care

TPSC2[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. Refer to Table 19.10 for details.

CKEG[1:0] Bits (Clock Edge Select)

These bits select the edge of the count clock signal input from the MTIOC1A pin.

Table 19.6 TPSC[2:0], TPSC2[2:0] (MTU0)

Channel	TCR2 register			TCR register			Description
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
	TPSC2[2]	TPSC2[1]	TPSC2[0]	TPSC[2]	TPSC[1]	TPSC[0]	
MTU0	0	0	0	0	0	0	Internal clock: counts on PCLKB/1
	0	0	0	0	0	1	Internal clock: counts on PCLKB/4
	0	0	0	0	1	0	Internal clock: counts on PCLKB/16
	0	0	0	0	1	1	Internal clock: counts on PCLKB/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	External clock: counts on MTCLKC pin input
	0	0	0	1	1	1	External clock: counts on MTCLKD pin input
	0	0	1	x	x	x	Internal clock: counts on PCLKB/2
	0	1	0	x	x	x	Internal clock: counts on PCLKB/8
	0	1	1	x	x	x	Internal clock: counts on PCLKB/32
	1	0	0	x	x	x	Internal clock: counts on PCLKB/256
	1	0	1	x	x	x	Internal clock: counts on PCLKB/1024
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	External clock: counts on MTIOC1A pin input

x: Don't care

Table 19.7 TPSC[2:0], TPSC2[2:0] (MTU1)

Channel	TCR2 register			TCR register			Description
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
	TPSC2[2]	TPSC2[1]	TPSC2[0]	TPSC[2]	TPSC[1]	TPSC[0]	
MTU1	0	0	0	0	0	0	Internal clock: counts on PCLKB/1
	0	0	0	0	0	1	Internal clock: counts on PCLKB/4
	0	0	0	0	1	0	Internal clock: counts on PCLKB/16
	0	0	0	0	1	1	Internal clock: counts on PCLKB/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	Internal clock: counts on PCLKB/256
	0	0	0	1	1	1	Overflow/underflow of MTU2.TCNT
	0	0	1	x	x	x	Internal clock: counts on PCLKB/2
	0	1	0	x	x	x	Internal clock: counts on PCLKB/8
	0	1	1	x	x	x	Internal clock: counts on PCLKB/32
	1	0	0	x	x	x	Internal clock: counts on PCLKB/1024
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

x: Don't care

Note: This setting has no effect when MTU1 is in phase counting mode.

Table 19.8 TPSC[2:0], TPSC2[2:0] (MTU2)

Channel	TCR2 register			TCR register			Description
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
	TPSC2[2]	TPSC2[1]	TPSC2[0]	TPSC[2]	TPSC[1]	TPSC[0]	
MTU2	0	0	0	0	0	0	Internal clock: counts on PCLKB/1
	0	0	0	0	0	1	Internal clock: counts on PCLKB/4
	0	0	0	0	1	0	Internal clock: counts on PCLKB/16
	0	0	0	0	1	1	Internal clock: counts on PCLKB/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	External clock: counts on MTCLKC pin input
	0	0	0	1	1	1	Internal clock: counts on PCLKB/1024
	0	0	1	x	x	x	Internal clock: counts on PCLKB/2
	0	1	0	x	x	x	Internal clock: counts on PCLKB/8
	0	1	1	x	x	x	Internal clock: counts on PCLKB/32
	1	0	0	x	x	x	Internal clock: counts on PCLKB/256
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

x: Don't care

Note: This setting has no effect when the MTU2 is in phase counting mode.

Table 19.9 TPSC[2:0], TPSC2[2:0] (MTU3, MTU4)

Channel	TCR2 register			TCR register			Description
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
	TPSC2[2]	TPSC2[1]	TPSC2[0]	TPSC[2]	TPSC[1]	TPSC[0]	
MTU3	0	0	0	0	0	0	Internal clock: counts on PCLKB/1
MTU4	0	0	0	0	0	1	Internal clock: counts on PCLKB/4
	0	0	0	0	1	0	Internal clock: counts on PCLKB/16
	0	0	0	0	1	1	Internal clock: counts on PCLKB/64
	0	0	0	1	0	0	Internal clock: counts on PCLKB/256
	0	0	0	1	0	1	Internal clock: counts on PCLKB/1024
	0	0	0	1	1	0	External clock: counts on MTCLKA pin input
	0	0	0	1	1	1	External clock: counts on MTCLKB pin input
	0	0	1	x	x	x	Internal clock: counts on PCLKB/2
	0	1	0	x	x	x	Internal clock: counts on PCLKB/8
	0	1	1	x	x	x	Internal clock: counts on PCLKB/32
	1	0	0	x	x	x	Setting prohibited
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

x: Don't care

Table 19.10 TPSC[1:0], TPSC2[2:0] (MTU5)

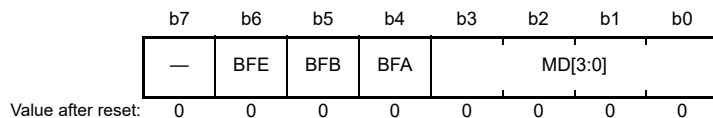
Channel	TCR2 register			TCR register		Description
	Bit 2	Bit 1	Bit 0	Bit 1	Bit 0	
	TPSC2[2]	TPSC2[1]	TPSC2[0]	TPSC[1]	TPSC[0]	
MTU5	0	0	0	0	0	Internal clock: counts on PCLKB/1
	0	0	0	0	1	Internal clock: counts on PCLKB/4
	0	0	0	1	0	Internal clock: counts on PCLKB/16
	0	0	0	1	1	Internal clock: counts on PCLKB/64
	0	0	1	x	x	Internal clock: counts on PCLKB/2
	0	1	0	x	x	Internal clock: counts on PCLKB/8
	0	1	1	x	x	Internal clock: counts on PCLKB/32
	1	0	0	x	x	Internal clock: counts on PCLKB/256
	1	0	1	x	x	Internal clock: counts on PCLKB/1024
	1	1	0	x	x	Setting prohibited
	1	1	1	x	x	External clock: counts on MTIOC1A pin input

x: Don't care

19.2.3 Timer Mode Register 1 (TMDR1)

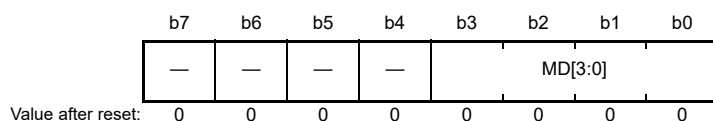
- MTU0.TMDR1

Address(es): MTU0.TMDR1 0009 5301h



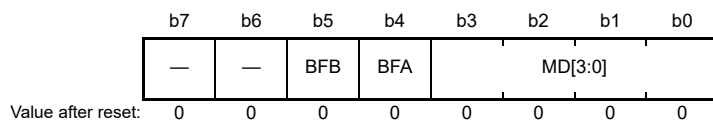
- MTU1.TMDR1, MTU2.TMDR1

Address(es): MTU1.TMDR1 0009 5381h, MTU2.TMDR1 0009 5401h



- MTU3.TMDR1, MTU4.TMDR1

Address(es): MTU3.TMDR1 0009 5202h, MTU4.TMDR1 0009 5203h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	MD[3:0]	Mode Select	These bits specify the timer operating mode. Refer to Table 19.11 for details.	R/W
b4	BFA	Buffer Operation A	0: TGRA and TGRC operate normally 1: TGRA and TGRC used together for buffer operation	R/W
b5	BFB	Buffer Operation B	0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation	R/W
b6	BFE	Buffer Operation E	0: MTU0.TGRE and MTU0.TGRF operate normally 1: MTU0.TGRE and MTU0.TGRF used together for buffer operation	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

The TMDR1 register specifies the operating mode of each channel. The MTU has a total of five TMDR1 registers, one each for MTU0 to MTU4. TMDR1 register values should be specified only while TCNT operation is stopped.

Table 19.11 Operating Mode Setting by MD[3:0] Bits (MTU0 to MTU4)

Bit 3	Bit 2	Bit 1	Bit 0		MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4
MD[3]	MD[2]	MD[1]	MD[0]	Description						
0	0	0	0	Normal mode	✓	✓	✓		✓	✓
0	0	0	1	Setting prohibited						
0	0	1	0	PWM mode 1	✓	✓	✓		✓	✓
0	0	1	1	PWM mode 2	✓	✓	✓			
0	1	0	0	Phase counting mode 1		✓	✓	✓		
0	1	0	1	Phase counting mode 2		✓	✓	✓		
0	1	1	0	Phase counting mode 3		✓	✓	✓		
0	1	1	1	Phase counting mode 4		✓	✓	✓		
1	0	0	0	Reset-synchronized PWM mode*1					✓	
1	0	0	1	Phase counting mode 5		✓	✓	✓		
1	0	1	x	Setting prohibited						
1	1	0	0	Setting prohibited						
1	1	0	1	Complementary PWM mode 1 (transfer at crest)*1					✓	
1	1	1	0	Complementary PWM mode 2 (transfer at trough)*1					✓	
1	1	1	1	Complementary PWM mode 3 (transfer at crest and trough)*1					✓	

x: Don't care

Note: Only set the corresponding operating mode listed above for each channel.

Note 1. Reset-synchronized PWM mode and complementary PWM mode can only be set for MTU3.

When MTU3 is set to reset-synchronized PWM mode or complementary PWM mode, the MTU4 settings become ineffective and automatically conform to the MTU3 setting, respectively. MTU4 should be set to the initial values (normal mode).

BFA Bit (Buffer Operation A)

This bit specifies whether to operate TGRA in the normal way or to use TGRA and TGRC together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare does not take place in modes other than complementary PWM mode, but compare match with TGRC occurs in complementary PWM mode.

If a compare match occurs on MTU4 in the Tb interval in complementary PWM mode, the TGIEC bit in timer interrupt enable register (MTU4.TIER) should be set to 0.

In reset-synchronized PWM mode or complementary PWM mode, buffer operation in MTU3 and MTU4 depends on the settings in the BFA bit of MTU3.TMDR1. The BFA bit of MTU4.TMDR1 should be set to 0.

In MTU1 and MTU2, which have no TGRC, this bit is reserved. It is read as 0. The write value should be 0.

Refer to Figure 19.47 for an illustration of the Tb interval in complementary PWM mode.

BFB Bit (Buffer Operation B)

This bit specifies whether to operate TGRB in the normal way or to use TGRB and TGRD together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare does not take place in modes other than complementary PWM mode, but compare match with TGRD occurs in complementary PWM mode.

If a compare match occurs on MTU4 in the Tb interval in complementary PWM mode, the TGIED bit in timer interrupt enable register (MTU4.TIER) should be set to 0.

In reset-synchronized PWM mode or complementary PWM mode, buffer operation in MTU3 and MTU4 depends on the settings in the BFB bit of MTU3.TMDR1. The BFB bit of MTU4.TMDR1 should be set to 0.

In MTU1 and MTU2, which have no TGRD, this bit is reserved. It is read as 0. The write value should be 0.

Refer to Figure 19.47 for an illustration of the Tb interval in complementary PWM mode.

BFE Bit (Buffer Operation E)

This bit specifies whether to operate MTU0.TGRE and MTU0.TGRF in the normal way or to use them together for buffer operation. Compare match with TGRF occurs even when TGRF is used as a buffer register.

In MTU0 to MTU4, this bit is reserved. It is read as 0. The write value should be 0.

19.2.4 Timer Mode Register 2A (TMDR2A)

Address(es): MTU.TMDR2A 0009 5270h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	DRS
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	DRS	Double Buffer Select	0: Double buffer function is disabled 1: Double buffer function is enabled	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TMDR2A specifies the double buffer function in complementary PWM mode 3 (transfer at the crest and trough of the counter value). TMDR2A value should be specified only while TCNT operation is stopped.

DRS Bit (Double Buffer Select)

This bit enables or disables the double buffer function in complementary PWM mode.

19.2.5 Timer Mode Register 3 (TMDR3)

Address(es): MTU1.TMDR3 0009 5391h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PHCKSEL	LWA
Value after reset:	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	LWA	MTU1/MTU2 Combination Longword Access Control	0: 16-bit access is enabled. 1: 32-bit access is enabled.	R/W
b1	PHCKSEL	External Input Phase Clock Select	0: MTCLKA and MTCLKB are selected for the external phase clock. 1: MTCLKC and MTCLKD are selected for the external phase clock.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The TMDR3 register controls longword access to a 32-bit register or counter in a combination of MTU1 and MTU2. There is only one TMDR3 register in MTU1. The counter (TCNTLW), general register A (TGRALW), and general register B (TGRBLW) of MTU1 and MTU2 are accessed in the combinations listed in Table 19.12.

LWA Bit (MTU1/MTU2 Combination Longword Access Control)

This bit selects a 32-bit access in a combination of MTU1 and MTU2.

When LWA is set to 0, the MTU1 and MTU2 independently operate as a 16-bit timer. therefore registers TCNTLW, TGRALW, and TGRBLW cannot be accessed.

When LWA is set to 1, MTU1 and MTU2 operate as a 32-bit cascaded timer and the timer is controlled by registers MTU1.TCR, MTU1.TCR2, MTU1.TIOR, and MTU1.TMDR1. The settings of registers MTU2.TCR, MTU2.TCR2, MTU2.TIOR, and MTU2.TMDR1 are disabled and the 16-bit registers (TCNT, TGRA, and TGRB) in MTU1 and MTU2 cannot be accessed. Furthermore, MTU2 input capture and compare match are also disabled.

The cascaded connection of MTU1 and MTU2 with the LWA bit set to 1 can only be used in phase counting mode, but not in normal mode, PWM1 mode, or PWM2 mode. Select phase counting mode when setting the LWA bit to 1.

Initialize the registers TCNT, TGRA, and TGRB in MTU1 and MTU2 in advance before setting the LWA bit to 1.

PHCKSEL Bit (External Input Phase Clock Select)

When the MTU1 and MTU2 registers are combined for 32-bit phase counting mode or MTU2 phase counting mode, this bit selects either the A- or B-phase signal from the external clock. Refer to Table 19.50, Clock Input Pins in Phase Counting Mode for details.

Table 19.12 Setting and Combination of the TMDR3 Register

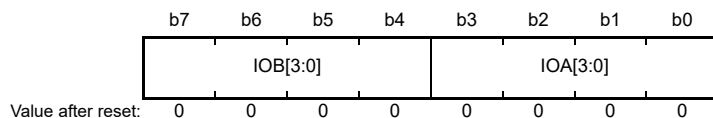
Register	TMDR3.LWA = 0		TMDR3.LWA = 1	
	Symbol	Access mode	Symbol	Access mode
Counter in MTU1*1	MTU1.TCNT	Word	MTU1.TCNTLW	Longword
Counter in MTU2	MTU2.TCNT	Word		
General register A in MTU1	MTU1.TGRA	Word	MTU1.TGRALW	Longword
General register A in MTU2	MTU2.TGRA	Word		
General register B in MTU1	MTU1.TGRB	Word	MTU1.TGRBLW	Longword
General register B in MTU2	MTU2.TGRB	Word		

Note 1. When the LWA bit is set to 1, setting the count clock for MTU1 as MTU2.TCNT overflow/underflow is not required.

19.2.6 Timer I/O Control Register (TIOR)

- MTU0.TIORH, MTU1.TIOR, MTU2.TIOR, MTU3.TIORH, MTU4.TIORH

Address(es): MTU0.TIORH 0009 5302h, MTU1.TIOR 0009 5382h, MTU2.TIOR 0009 5402h, MTU3.TIORH 0009 5204h, MTU4.TIORH 0009 5206h

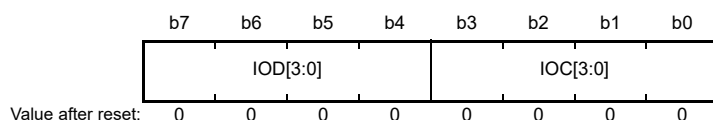


Bit	Symbol	Bit Name	Description	R/W
b3 to b0	IOA[3:0]	I/O Control A* ¹	Refer to the following tables. MTU0.TIORH: Table 19.21 MTU1.TIOR: Table 19.23 MTU2.TIOR: Table 19.24 MTU3.TIORH: Table 19.25 MTU4.TIORH: Table 19.27	R/W
b7 to b4	IOB[3:0]	I/O Control B* ¹	Refer to the following tables. MTU0.TIORH: Table 19.13 MTU1.TIOR: Table 19.15 MTU2.TIOR: Table 19.16 MTU3.TIORH: Table 19.17 MTU4.TIORH: Table 19.19	R/W

Note 1. When the value of IOm[3:0] (m = A, B) is changed to the output-prohibited state (0000b or 0100b) during low, high, or toggle output at compare match, this register is in Hi-Z.

- MTU0.TIORL, MTU3.TIORL, MTU4.TIORL

Address(es): MTU0.TIORL 0009 5303h, MTU3.TIORL 0009 5205h, MTU4.TIORL 0009 5207h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	IOC[3:0]	I/O Control C* ¹	Refer to the following tables. MTU0.TIORL: Table 19.22 MTU3.TIORL: Table 19.26 MTU4.TIORL: Table 19.28	R/W
b7 to b4	IOD[3:0]	I/O Control D* ¹	Refer to the following tables. MTU0.TIORL: Table 19.14 MTU3.TIORL: Table 19.18 MTU4.TIORL: Table 19.20	R/W

Note 1. When the value of IOm[3:0] (m = C, D) is changed to the output-prohibited state (0000b or 0100b) during low, high, or toggle output at compare match, this register is in Hi-Z.

- MTU5.TIORU, MTU5.TIORV, MTU5.TIORW

Address(es): MTU5.TIORU 0009 5486h, MTU5.TIORV 0009 5496h, MTU5.TIORW 0009 54A6h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	IOC[4:0]	I/O Control C	Refer to the following table. MTU5.TIORU, MTU5.TIORV, MTU5.TIORW: Table 19.29	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The TIOR register controls the TGRm register. The MTU has a total of 11 TIOR registers, two each for MTU0, MTU3, and MTU4, one each for MTU1 and MTU2, and three (MTU5.TIORU/TIORV/TIORW) for MTU5. The TIOR register should be set when the TMDR register setting is normal mode, PWM mode, or phase counting mode.

Note that TIOR is affected by the TMDR1 setting.

The initial output specified by TIOR is valid when the counter is stopped (the CSTn bit in TSTRA is set to 0). Note also that, in PWM mode 2, the output at the point at which the counter becomes 0000h is specified.

When TGRC or TGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

Table 19.13 TIORH (MTU0)

Bit 7	Bit 6	Bit 5	Bit 4	Description
IOB[3]	IOB[2]	IOB[1]	IOB[0]	TGRB Register Function MTIOC0B Pin Function
0	0	0	0	Output compare register Output prohibited
0	0	0	1	Initial output is low. Low output at compare match.
0	0	1	0	Initial output is low. High output at compare match.
0	0	1	1	Initial output is low. Toggle output at compare match.
0	1	0	0	Output prohibited
0	1	0	1	Initial output is high. Low output at compare match.
0	1	1	0	Initial output is high. High output at compare match.
0	1	1	1	Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register Input capture at rising edge.
1	0	0	1	Input capture at falling edge.
1	0	1	x	Input capture at both edges.
1	1	x	x	Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by MTU1.TCNT (LWA = 0) or MTU1.TCNTLW (LWA = 1).*1

x: Don't care

Note 1. When PCLKB/1 is selected as the count clock for MTU1, MTU0 input capture is not generated. Do not select PCLKB/1 as the count clock for MTU1.

Table 19.14 TIORL (MTU0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD[3]	IOD[2]	IOD[1]	IOD[0]	TGRD Register Function	MTIOC0D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register*1	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by MTU1.TCNT (LWA = 0) or MTU1.TCNTLW (LWA = 1).*2

x: Don't care

Note 1. When the MTU0.TMDR1.BFB bit is set to 1 and MTU0.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Note 2. When PCLKB/1 is selected as the count clock for MTU1, MTU0 input capture is not generated. Do not select PCLKB/1 as the count clock for MTU1.

Table 19.15 TIOR (MTU1)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB[3]	IOB[2]	IOB[1]	IOB[0]	TGRB/TGRBLW Register Function	MTIOC1B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Input capture at occurrence of compare match or input capture in the MTU0.TGRC register

x: Don't care

Table 19.16 TIOR (MTU2)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB[3]	IOB[2]	IOB[1]	IOB[0]	TGRB Register Function	MTIOC2B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 19.17 TIORH (MTU3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB[3]	IOB[2]	IOB[1]	IOB[0]	TGRB Register Function	MTIOC3B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 19.18 TIORL (MTU3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD[3]	IOD[2]	IOD[1]	IOD[0]	TGRD Register Function	MTIOC3D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the MTU3.TMDR1.BFB bit is set to 1 and MTU3.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 19.19 TIORH (MTU4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB[3]	IOB[2]	IOB[1]	IOB[0]	TGRB Register Function	MTIOC4B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 19.20 TIORL (MTU4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD[3]	IOD[2]	IOD[1]	IOD[0]	TGRD Register Function	MTIOC4D Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the MTU4.TMDR1.BFB bit is set to 1 and MTU4.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 19.21 TIORH (MTU0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA[3]	IOA[2]	IOA[1]	IOA[0]	TGRA Register Function	MTIOC0A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by MTU1.TCNT (LWA = 0) or MTU1.TCNTLW (LWA = 1).*1

x: Don't care

Note 1. When PCLKB/1 is selected as the count clock for MTU1, MTU0 input capture is not generated. Do not select PCLKB/1 as the count clock for MTU1.

Table 19.22 TIORL (MTU0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC[3]	IOC[2]	IOC[1]	IOC[0]	TGRC Register Function	MTIOC0C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register*1	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by MTU1.TCNT (LWA = 0) or MTU1.TCNTLW (LWA = 1).*2

x: Don't care

Note 1. When the MTU0.TMDR1.BFB bit is set to 1 and MTU0.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Note 2. When PCLKB/1 is selected as the count clock for MTU1, MTU0 input capture is not generated. Do not select PCLKB/1 as the count clock for MTU1.

Table 19.23 TIOR (MTU1)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA[3]	IOA[2]	IOA[1]	IOA[0]	TGRA/TGRALW Register Function	MTIOC1A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Input capture at generation of MTU0.TGRA compare match/input capture.

x: Don't care

Table 19.24 TIOR (MTU2)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA[3]	IOA[2]	IOA[1]	IOA[0]	TGRA Register Function	MTIOC2A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 19.25 TIORH (MTU3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA[3]	IOA[2]	IOA[1]	IOA[0]	TGRA Register Function	MTIOC3A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 19.26 TIORL (MTU3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC[3]	IOC[2]	IOC[1]	IOC[0]	TGRC Register Function	MTIOC3C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the MTU3.TMDR1.BFB bit is set to 1 and MTU3.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 19.27 TIORH (MTU4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA[3]	IOA[2]	IOA[1]	IOA[0]	TGRA Register Function	MTIOC4A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Table 19.28 TIORL (MTU4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC[3]	IOC[2]	IOC[1]	IOC[0]	TGRC Register Function	MTIOC4C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

x: Don't care

Note 1. When the MTU4.TMDR1.BFB bit is set to 1 and MTU4.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 19.29 TIORU, TIORV, and TIORW (MTU5)

Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC[4]	IOC[3]	IOC[2]	IOC[1]	IOC[0]	TGRU, TGRV, TGRW Registers Function	MTIC5U, MTIC5V, MTIC5W Pin Function
0	0	0	0	0	Output compare register	No function
0	0	0	0	1		Setting prohibited
0	0	0	1	x		Setting prohibited
0	0	1	x	x		Setting prohibited
0	1	x	x	x		Setting prohibited
1	0	0	0	0	Input capture register*1	Setting prohibited
1	0	0	0	1		Input capture at rising edge.
1	0	0	1	0		Input capture at falling edge.
1	0	0	1	1		Input capture at both edges.
1	0	1	x	x		Setting prohibited
1	1	0	0	0		Setting prohibited
1	1	0	0	1		Measurement of low pulse width of external input signal. Capture at trough in complementary PWM mode.
1	1	0	1	0		Measurement of low pulse width of external input signal. Capture at crest of complementary PWM mode.
1	1	0	1	1		Measurement of low pulse width of external input signal. Capture at crest and trough of complementary PWM mode.
1	1	1	0	0		Setting prohibited
1	1	1	0	1		Measurement of high pulse width of external input signal. Capture at trough in complementary PWM mode.
1	1	1	1	0		Measurement of high pulse width of external input signal. Capture at crest of complementary PWM mode.
1	1	1	1	1		Measurement of high pulse width of external input signal. Capture at crest and trough of complementary PWM mode.

x: Don't care

Note 1. Set the IOC[4:0] bits to 19h, 1Ah, 1Bh, 1Dh, 1Eh, or 1Fh only when using external pulse width measurement or only when using dead time compensation linked with MTU3 and MTU4. For details, refer to section 19.3.11, External Pulse Width Measurement and section 19.3.12, Dead Time Compensation.

19.2.7 Timer Compare Match Clear Register (TCNTCMPCLR)

Address(es): MTU5.TCNTCMPCLR 0009 54B6h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CMPCLR5U	CMPCLR5V	CMPCLR5W
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMPCLR5W	TCNT Compare Clear 5W	0: Disables MTU5.TCNTW to be cleared to 0000h at MTU5.TCNTW and MTU5.TGRW compare match or input capture 1: Enables MTU5.TCNTW to be cleared to 0000h at MTU5.TCNTW and MTU5.TGRW compare match or input capture	R/W
b1	CMPCLR5V	TCNT Compare Clear 5V	0: Disables MTU5.TCNTV to be cleared to 0000h at MTU5.TCNTV and MTU5.TGRV compare match or input capture 1: Enables MTU5.TCNTV to be cleared to 0000h at MTU5.TCNTV and MTU5.TGRV compare match or input capture	R/W
b2	CMPCLR5U	TCNT Compare Clear 5U	0: Disables MTU5.TCNTU to be cleared to 0000h at MTU5.TCNTU and MTU5.TGRU compare match or input capture 1: Enables MTU5.TCNTU to be cleared to 0000h at MTU5.TCNTU and MTU5.TGRU compare match or input capture	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TCNTCMPCLR specifies requests to clear MTU5.TCNTU, MTU5.TCNTV, and MTU5.TCNTW. The MTU has one TCNTCMPCLR (on MTU5).

19.2.8 Timer Interrupt Enable Register (TIER)

- MTU1.TIER, MTU2.TIER

Address(es): MTU1.TIER 0009 5384h, MTU2.TIER 0009 5404h

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA
Value after reset:	0	0	0	0	0	0	0	0

- MTU0.TIER, MTU3.TIER

Address(es): MTU0.TIER 0009 5304h, MTU3.TIER 0009 5208h

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Value after reset:	0	0	0	0	0	0	0	0

- MTU4.TIER

Address(es): MTU4.TIER 0009 5209h

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TGIEA	TGR Interrupt Enable A	0: Interrupt requests (TGIA) disabled 1: Interrupt requests (TGIA) enabled	R/W
b1	TGIEB	TGR Interrupt Enable B	0: Interrupt requests (TGIB) disabled 1: Interrupt requests (TGIB) enabled	R/W
b2	TGIEC	TGR Interrupt Enable C	0: Interrupt requests (TGIC) disabled 1: Interrupt requests (TGIC) enabled	R/W
b3	TGIED	TGR Interrupt Enable D	0: Interrupt requests (TGID) disabled 1: Interrupt requests (TGID) enabled	R/W
b4	TCIEV	Overflow Interrupt Enable	0: Interrupt requests (TCIV) disabled 1: Interrupt requests (TCIV) enabled	R/W
b5	TCIEU	Underflow Interrupt Enable	0: Interrupt requests (TCIU) disabled 1: Interrupt requests (TCIU) enabled	R/W
b6	TTGE2	A/D Conversion Start Request Enable 2	0: A/D conversion start request generation by MTUn.TCNT underflow (trough) disabled 1: A/D conversion start request generation by MTUn.TCNT underflow (trough) enabled	R/W
b7	TTGE	A/D Conversion Start Request Enable	0: A/D conversion start request generation disabled 1: A/D conversion start request generation enabled	R/W

n = 4

The TIER register enables or disables interrupt requests from each channel. The MTU has a total of seven TIER registers, two for MTU0 and one each for MTU1 to MTU5.

TGIEA and TGIEB Bits (TGR Interrupt Enable A and B)

Each bit enables or disables interrupt requests (TGIm) (m = A, B).

TGIEC and TGIED Bits (TGR Interrupt Enable C and D)

Each bit enables or disables an interrupt request (TGIm) (m = C, D).

In MTU1 and MTU2, these bits are reserved. They are read as 0. The write value should be 0.

TCIEV Bit (Overflow Interrupt Enable)

This bit enables or disables interrupt requests (TCIV).

TCIEU Bit (Underflow Interrupt Enable)

This bit enables or disables interrupt requests (TCIU).

In MTU0, MTU3, and MTU4, this bit is reserved. It is read as 0. The write value should be 0.

TTGE2 Bit (A/D Conversion Start Request Enable 2)

This bit enables or disables generation of A/D conversion start requests by MTUn.TCNT underflow (trough) in complementary PWM mode (n = 4).

In MTU0 to MTU3, this bit is reserved. It is read as 0. The write value should be 0.

TTGE Bit (A/D Conversion Start Request Enable)

This bit enables or disables generation of A/D conversion start requests by TGRA input capture/compare match.

- MTU0.TIER2

Address(es): MTU0.TIER2 0009 5324h

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE2	—	—	—	—	—	TGIEF	TGIEE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TGIEE	TGR Interrupt Enable E	0: Interrupt requests (TGIE) disabled 1: Interrupt requests (TGIE) enabled	R/W
b1	TGIEF	TGR Interrupt Enable F	0: Interrupt requests (TGIF) disabled 1: Interrupt requests (TGIF) enabled	R/W
b6 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TTGE2	A/D Conversion Start Request Enable 2	0: A/D conversion start request generation by compare match between MTU0.TCNT and MTU0.TGRE disabled 1: A/D conversion start request generation by compare match between MTU0.TCNT and MTU0.TGRE enabled	R/W

TGIEE and TGIEF Bits (TGR Interrupt Enable E and F)

Each bit enables or disables interrupt requests by compare match between MTU0.TCNT and MTU0.TGRm (m = E, F).

TTGE2 Bit (A/D Conversion Start Request Enable 2)

Each bit enables or disables A/D conversion start requests by compare match between MTU0.TCNT and MTU0.TGRE.

- MTU5.TIER

Address(es): MTU5.TIER 0009 54B2h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TGIE5 U	TGIE5V	TGIE5 W
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TGIE5W	TGR Interrupt Enable 5W	0: Interrupt requests TGIW5 disabled 1: Interrupt requests TGIW5 enabled	R/W
b1	TGIE5V	TGR Interrupt Enable 5V	0: Interrupt requests TGIV5 disabled 1: Interrupt requests TGIV5 enabled	R/W
b2	TGIE5U	TGR Interrupt Enable 5U	0: Interrupt requests TGIU5 disabled 1: Interrupt requests TGIU5 enabled	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TGIE5m Bits (TGR Interrupt Enable 5m)

Each bit enables or disables interrupt requests (TGIm5) (m = U, V, W).

19.2.9 Timer Status Register (TSR)

- MTU1.TSR, MTU2.TSR

Address(es): MTU1.TSR 0009 5385h, MTU2.TSR 0009 5405h

	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	—	—	—	—	—
Value after reset:	1	1	0	0	0	0	0	0

- MTU3.TSR, MTU4.TSR

Address(es): MTU3.TSR 0009 522Ch, MTU4.TSR 0009 522Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	—	—	—	—	—
Value after reset:	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	The read value is undefined. The write value should be 1.	R/W
b6	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b7	TCFD	Count Direction Flag	0: TCNT counts down 1: TCNT counts up	R

TSR indicates the states of each of the channels. The MTU has a total of four TSR registers, one each for MTU1 to MTU4.

TCFD Flag (Count Direction Flag)

Status flag that indicates the direction in which TCNT is counting in MTU1 to MTU4.

19.2.10 Timer Buffer Operation Transfer Mode Register (TBTM)

- MTU0.TBTM

Address(es): MTU0.TBTM 0009 5326h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	TTSE	TTSB	TTSA
0	0	0	0	0	0	0	0

Value after reset:

- MTU3.TBTM, MTU4.TBTM

Address(es): MTU3.TBTM 0009 5238h, MTU4.TBTM 0009 5239h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	TTSB	TTSA
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	TTSA	Timing Select A	0: When compare match A occurs in each channel, data is transferred from TGRC to TGRA 1: When TCNT is cleared in each channel, data is transferred from TGRC to TGRA	R/W
b1	TTSB	Timing Select B	0: When compare match B occurs in each channel, data is transferred from TGRD to TGRB 1: When TCNT is cleared in each channel, data is transferred from TGRD to TGRB	R/W
b2	TTSE	Timing Select E	0: When compare match E occurs in MTU0, data is transferred from MTU0.TGRF to MTU0.TGRE 1: When MTU0.TCNT is cleared, data is transferred from MTU0.TGRF to MTU0.TGRE	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TBTM specifies the timing for transferring data from the buffer register to the timer general register in PWM mode. The MTU has a total of three TBTM registers, one each for MTU0, MTU3, and MTU4.

TTSA Bit (Timing Select A)

This bit specifies the timing for transferring data from TGRC to TGRA in each channel when they are used together for buffer operation. When a channel is not set to PWM mode, do not set the TTSA bit in the channel to 1.

TTSB Bit (Timing Select B)

This bit specifies the timing for transferring data from TGRD to TGRB in each channel when they are used together for buffer operation. When a channel is not set to PWM mode, do not set the TTSB bit in the channel to 1.

TTSE Bit (Timing Select E)

This bit specifies the timing for transferring data from MTU0.TGRF to MTU0.TGRE when they are used together for buffer operation.

In MTU3 and MTU4, this bit is reserved. It is read as 0 and the write value should be 0. When a channel is not set to PWM mode, do not set the TTSE bit in the channel to 1.

19.2.11 Timer Input Capture Control Register (TICCR)

Address(es): MTU1.TICCR 0009 5390h

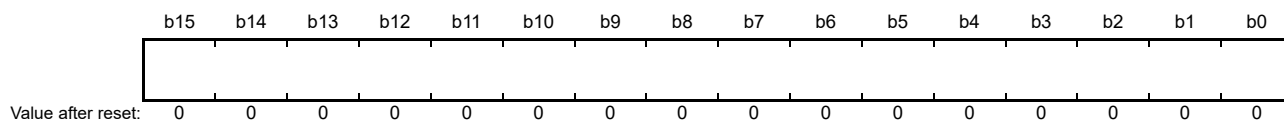
	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	I2BE	I2AE	I1BE	I1AE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	I1AE	Input Capture Enable	0: Does not include the MTIOC1A pin in the MTU2.TGRA input capture conditions 1: Includes the MTIOC1A pin in the MTU2.TGRA input capture conditions	R/W
b1	I1BE	Input Capture Enable	0: Does not include the MTIOC1B pin in the MTU2.TGRB input capture conditions 1: Includes the MTIOC1B pin in the MTU2.TGRB input capture conditions	R/W
b2	I2AE	Input Capture Enable	0: Does not include the MTIOC2A pin in the MTU1.TGRA input capture conditions 1: Includes the MTIOC2A pin in the MTU1.TGRA input capture conditions	R/W
b3	I2BE	Input Capture Enable	0: Does not include the MTIOC2B pin in the MTU1.TGRB input capture conditions 1: Includes the MTIOC2B pin in the MTU1.TGRB input capture conditions	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TICCR specifies input capture conditions when MTU1.TCNT and MTU2.TCNT are cascaded. The MTU has one TICCR for MTU1.

19.2.12 Timer Counter (TCNT)

Address(es): MTU0.TCNT 0009 5306h, MTU1.TCNT 0009 5386h, MTU2.TCNT 0009 5406h, MTU3.TCNT 0009 5210h, MTU4.TCNT 0009 5212h, MTU5.TCNTU 0009 5480h, MTU5.TCNTV 0009 5490h, MTU5.TCNTW 0009 54A0h



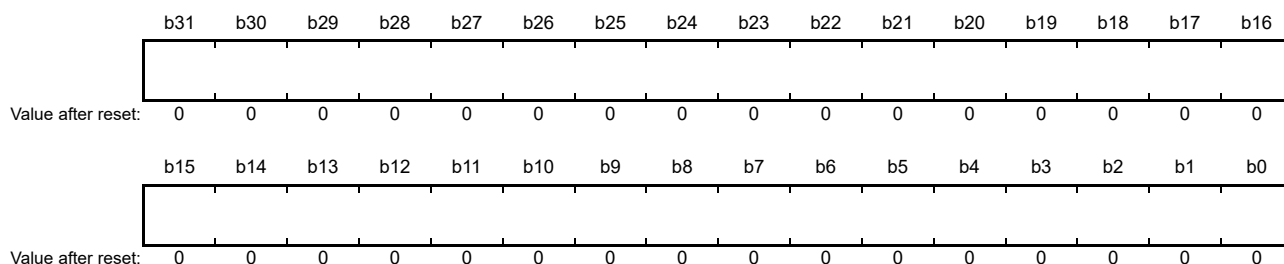
Note: TCNT must not be accessed in 8 bits; it should be accessed in 16 bits.

TCNT is a 16-bit readable/writable counter. The MTU has a total of eight TCNT counters, one each for MTU0 to MTU4 and three (MTU5.TCNTU, TCNTV, and TCNTW) for MTU5. The TCNT counters in MTU0 to MTU4 are initialized to 0000h by a reset. MTU5.TCNTU, MTU5.TCNTV, and MTU5.TCNTW are initialized to 0000h by a reset.

In MTU0 to MTU4, the TCNT counters must not be accessed in 8-bit units; they should be accessed in 16-bit units. The MTU1.TCNT and MTU2.TCNT counters are read as 0000h when TMDR3.LWA is 1. Refer to section 19.2.5, Timer Mode Register 3 (TMDR3) for details.

19.2.13 Timer Longword Counter (TCNTLW)

Address(es): MTU1.TCNTLW 0009 53A0h



Note: TCNTLW must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

The TCNTLW counter is a 32-bit readable/writable counter. Only one counter of this type is provided, and is formed by combining MTU1.TCNT and MTU2.TCNT. Such operation is only effective when TMDR3.LWA is 1. The TCNTLW counter is initialized to 0000 0000h by a reset. This counter is read as 0000 0000h when TMDR3.LWA is 0. Refer to section 19.2.5, Timer Mode Register 3 (TMDR3) for details. This register can only be used in 32-bit phase counting mode.

19.2.14 Timer General Register m (TGRm) (m = A, B, C, D, E, F, U, V, W)

Address(es): MTU0.TGRA 0009 5308h, MTU0.TGRB 0009 530Ah, MTU0.TGRC 0009 530Ch, MTU0.TGRD 0009 530Eh, MTU0.TGRE 0009 5320h, MTU0.TGRF 0009 5322h, MTU1.TGRA 0009 5388h, MTU1.TGRB 0009 538Ah, MTU2.TGRA 0009 5408h, MTU2.TGRB 0009 540Ah, MTU3.TGRA 0009 5218h, MTU3.TGRB 0009 521Ah, MTU3.TGRC 0009 5224h, MTU3.TGRD 0009 5226h, MTU3.TGRE 0009 5272h, MTU4.TGRA 0009 521Ch, MTU4.TGRB 0009 521Eh, MTU4.TGRC 0009 5228h, MTU4.TGRD 0009 522Ah, MTU4.TGRE 0009 5274h, MTU4.TGRF 0009 5276h, MTU5.TGRU 0009 5482h, MTU5.TGRV 0009 5492h, MTU5.TGRW 0009 54A2h



Note: The TGRm registers must not be accessed in 8 bits; They should be accessed in 16 bits. The initial value of the TGRm register is FFFFh.

The TGRm register is 16-bit readable/writable register. The MTU has a total of 24 TGRm registers, six for MTU0, two each for MTU1 and MTU2, five for MTU3, six for MTU4, and three for MTU5.

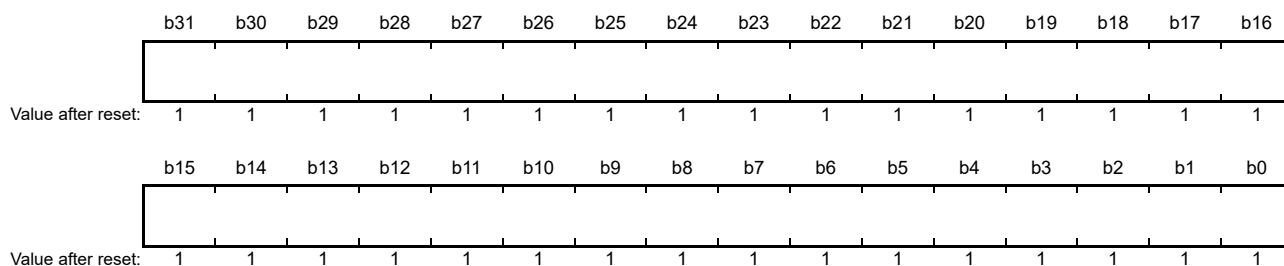
The TGRA, TGRB, TGRC, and TGRD registers function as either output compare or input capture registers. The TGRC and TGRD registers for MTU0, MTU3, and MTU4 can also be designated for operation as buffer registers. TGRm buffer register combinations are TGRA and TGRC, and TGRB and TGRD.

MTU0.TGRE and MTU0.TGRF function as compare registers. When the MTU0.TCNT count matches the MTU0.TGRE value, an A/D conversion start request can be issued. The TGRF register can also be designated for operation as a buffer register. TGRm buffer register combination is TGRE and TGRF. MTU5.TGRU, MTU5.TGRV, and MTU5.TGRW function as compare match, input capture, or external pulse width measurement registers.

The MTU1.TGRA, MTU2.TGRA, MTU1.TGRB, and MTU2.TGRB registers are read as 0000h when TMDR3.LWA is 1. Refer to section 19.2.5, Timer Mode Register 3 (TMDR3) for details.

19.2.15 Timer Longword General Register m (TGRmLW) (m = A, B)

Address(es): MTU1.TGRALW 0009 53A4h, MTU1.TGRBLW 0009 53A8h



Note: TGRALW and TGRBLW must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

The TGRmLW register is a 32-bit readable/writable register. Two general registers of this type are provided, and are formed by combining MTU1.TGRm and MTU2.TGRm. Such operation is only effective when TMDR3.LWA is 1. The TGRmLW register is initialized to FFFF FFFFh by a reset, but it is read as 0000 0000h when TMDR3.LWA is 0. Refer to section 19.2.5, Timer Mode Register 3 (TMDR3) for details.

The TGRALW and TGRBLW registers function as compare match or input capture registers which can only be used in 32-bit phase counting mode.

19.2.16 Timer Start Registers (TSTRA, TSTR)

- MTU.TSTRA (for MTU0, MTU1, MTU2, MTU3, and MTU4)

Address(es): MTU.TSTRA 0009 5280h

b7	b6	b5	b4	b3	b2	b1	b0
CST4	CST3	—	—	—	CST2	CST1	CST0

Value after reset: 0 0 0 0 0 0 0 0

Bit	Symbol	Bit Name	Description	R/W
b0	CST0	Counter Start 0	0: MTU0.TCNT counting is stopped 1: MTU0.TCNT performs count operation	R/W
b1	CST1	Counter Start 1	0: MTU1.TCNT counting is stopped 1: MTU1.TCNT performs count operation	R/W
b2	CST2	Counter Start 2	0: MTU2.TCNT counting is stopped 1: MTU2.TCNT performs count operation	R/W
b5 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	CST3	Counter Start 3	0: MTU3.TCNT counting is stopped 1: MTU3.TCNT performs count operation	R/W
b7	CST4	Counter Start 4	0: MTU4.TCNT counting is stopped 1: MTU4.TCNT performs count operation	R/W

Note: When 1 is written to a bit in TCSYSTR, the corresponding bit in TSTRA is also set to 1 automatically.

The TSTRA register starts or stops TCNT operation in MTU0 to MTU4.

TSTR starts or stops TCNT operation in MTU5.

Before setting the operating mode in TMDR1 or setting the TCNT count clock in TCR, be sure to stop the TCNT counter.

CSTn Bits (Counter Start n) (n = 0, 1, 2, 3, 4)

Each bit starts or stops TCNT in the corresponding channel.

If 0 is written to the CSTn bit during operation with the MTIOC pin designated for output, the counter stops. At this time, initial output level specified in the TOCR1A or TOCR2A register is output from the MTIOC pin in complementary PWM mode or reset-synchronized PWM mode. In any mode other than complementary PWM mode and reset synchronous PWM mode, the output compare signal level from the MTIOC pin is retained. If TIOR is written to while the CSTn bit is 0, the pin output level will be changed to the specified initial output value.

- MTU5.TSTR

Address(es): MTU5.TSTR 0009 54B4h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CSTU5	CSTV5	CSTW5
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CSTW5	Counter Start W5	0: MTU5.TCNTW counting is stopped 1: MTU5.TCNTW performs count operation	R/W
b1	CSTV5	Counter Start V5	0: MTU5.TCNTV counting is stopped 1: MTU5.TCNTV performs count operation	R/W
b2	CSTU5	Counter Start U5	0: MTU5.TCNTU counting is stopped 1: MTU5.TCNTU performs count operation	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

19.2.17 Timer Synchronous Register A (TSYRA)

Address(es): MTU.TSYRA 0009 5281h

b7	b6	b5	b4	b3	b2	b1	b0
SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	SYNC0	Timer Synchronous Operation 0	0: MTU0.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU0.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b1	SYNC1	Timer Synchronous Operation 1	0: MTU1.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU1.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b2	SYNC2	Timer Synchronous Operation 2	0: MTU2.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU2.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b5 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	SYNC3	Timer Synchronous Operation 3	0: MTU3.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU3.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W
b7	SYNC4	Timer Synchronous Operation 4	0: MTU4.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU4.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)	R/W

TSYRA selects independent operation or synchronous operation of TCNT in MTU0 to MTU4.

A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

SYNCn Bits (Timer Synchronous Operation n) (n = 0, 1, 2, 3, 4)

Each bit selects whether operation is independent of or synchronized with other channels.

When synchronous operation is selected, the TCNT synchronous setting of multiple channels and synchronous clearing by counter clearing on another channel are possible.

To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of the TCR.CCLR[2:0] bits.

19.2.18 Timer Counter Synchronous Start Register (TCSYSTR)

Address(es): MTU.TCSYSTR 0009 5282h

b7	b6	b5	b4	b3	b2	b1	b0
SCH0	SCH1	SCH2	SCH3	SCH4	—	—	—
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	SCH4	Synchronous Start 4	0: Does not specify synchronous start for MTU4.TCNT 1: Specifies synchronous start for MTU4.TCNT	R/(W)*1
b4	SCH3	Synchronous Start 3	0: Does not specify synchronous start for MTU3.TCNT 1: Specifies synchronous start for MTU3.TCNT	R/(W)*1
b5	SCH2	Synchronous Start 2	0: Does not specify synchronous start for MTU2.TCNT 1: Specifies synchronous start for MTU2.TCNT	R/(W)*1
b6	SCH1	Synchronous Start 1	0: Does not specify synchronous start for MTU1.TCNT 1: Specifies synchronous start for MTU1.TCNT	R/(W)*1
b7	SCH0	Synchronous Start 0	0: Does not specify synchronous start for MTU0.TCNT 1: Specifies synchronous start for MTU0.TCNT	R/(W)*1

Note 1. Only 1 can be written to this bit. This bit is automatically cleared when the corresponding counter starts.

TCSYSTR specifies synchronous start of the counters.

SCH4 Bit (Synchronous Start 4)

This bit controls synchronous start of MTU4.TCNT.

[Clearing condition]

- When 1 is set to the TSTRA.CST4 bit while SCH4 = 1

SCH3 Bit (Synchronous Start 3)

This bit controls synchronous start of MTU3.TCNT.

[Clearing condition]

- When 1 is set to the TSTRA.CST3 bit while SCH3 = 1

SCH2 Bit (Synchronous Start 2)

This bit controls synchronous start of MTU2.TCNT.

[Clearing condition]

- When 1 is set to the TSTRA.CST2 bit while SCH2 = 1

SCH1 Bit (Synchronous Start 1)

This bit controls synchronous start of MTU1.TCNT.

[Clearing condition]

- When 1 is set to the TSTRA.CST1 bit while SCH1 = 1

SCH0 Bit (Synchronous Start 0)

This bit controls synchronous start of MTU0.TCNT.

[Clearing condition]

- When 1 is set to the TSTRA.CST0 bit while SCH0 = 1

19.2.19 Timer Read/Write Enable Register A (TRWERA)

Address(es): MTU.TRWERA 0009 5284h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	RWE
Value after reset:	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	RWE	Read/Write Enable	0: Read/write access to the registers is disabled 1: Read/write access to the registers is enabled	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TRWERA enables or disables access to the registers and counters that have write-protection capability against accidental modification in MTU3 and MTU4.

RWE Bit (Read/Write Enable)

This bit enables or disables access to the registers that have write-protection capability against accidental modification.
[Clearing condition]

- When 0 is written to the RWE bit after reading RWE = 1
- Registers and Counters having Write-Protection Capability against Accidental Modification (TRWERA)
24 registers: MTUn.TCR, MTUn.TCR2, MTUn.TMDR1, MTUn.TIORH, MTUn.TIORL, MTUn.TIER, MTUn.TGRA, MTUn.TGRB, MTU.TOERA, MTU.TOCR1A, MTU.TOCR2A, MTU.TGCRA, MTU.TCDRA, MTU.TDDRA, and MTUn.TCNT (n = 3, 4)

19.2.20 Timer Output Master Enable Register A (TOERA)

Address(es): MTU.TOERA 0009 520Ah

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
Value after reset:	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OE3B	Master Enable MTIOC3B	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b1	OE4A	Master Enable MTIOC4A	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b2	OE4B	Master Enable MTIOC4B	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b3	OE3D	Master Enable MTIOC3D	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b4	OE4C	Master Enable MTIOC4C	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b5	OE4D	Master Enable MTIOC4D	0: MTU output is disabled*1 1: MTU output is enabled	R/W
b7, b6	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

Note 1. To output the inactive level from each pin when the MTU output is set to disabled, first set the data direction register (PDR) and port output data register (PODR) of I/O ports to output the inactive level from general I/O ports, and then set the port mode register (PMR) to use general I/O ports. For details, refer to section 17, I/O Ports.

TOERA enables or disables output settings for output pins MTIOC4D, MTIOC4C, MTIOC3D, MTIOC4B, MTIOC4A, and MTIOC3B.

These pins do not output correctly if the bits in the TOERA register have not been set. In MTU3 and MTU4, set TOERA prior to setting TIOR.

Set MTU.TOERA after setting the CST3 and CST4 bits in MTU.TSTRA to 0 (refer to Figure 19.42 and Figure 19.45).

19.2.21 Timer Output Control Register 1A (TOCR1A)

Address(es): MTU.TOCR1A 0009 520Eh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OLSP	Output Level Select P*1, *3	Refer to Table 19.30.	R/W
b1	OLSN	Output Level Select N*1, *3	Refer to Table 19.31.	R/W
b2	TOCS	TOC Select	0: TOCR1m setting is selected (m = A) 1: TOCR2m setting is selected	R/W
b3	TOCL	TOC Register Write Protection*2, *4	0: Write access to the TOCS, OLSN, and OLSP bits is enabled 1: Write access to the TOCS, OLSN, and OLSP bits is disabled	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	PSYE	PWM Synchronous Output Enable	0: Toggle output is disabled 1: Toggle output is enabled	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note 1. Setting the TOCR1m.TOCS bit to 0 makes this bit setting valid.

Note 2. Setting the TOCR1m.TOCL bit to 1 prevents accidental modification when the CPU goes out of control.

Note 3. If dead-time is not generated, the negative-phase output is the exact inverse of the positive-phase output. In this case, only the OLSP bit is valid.

Note 4. This bit can be set to 1 only once after a reset. After 1 is written, 0 cannot be written to the bit.

TOCR1A enables or disables PWM-synchronized toggle output in complementary PWM mode and reset-synchronized PWM mode, and control inversion of PWM output level.

OLSP Bit (Output Level Select P)

This bit selects the positive-phase output level in reset-synchronized PWM mode and complementary PWM mode. The initial output is selected while the counter is stopped.

OLSN Bit (Output Level Select N)

This bit selects the negative-phase output level in reset-synchronized PWM mode and complementary PWM mode. The initial output is selected while the counter is stopped.

TOCS Bit (TOC Select)

This bit selects either the TOCR1m or TOCR2m (m = A) setting to be used for the output level in complementary PWM mode and reset-synchronized PWM mode.

TOCL Bit (TOC Register Write Protection)

This bit enables or disables write access to the TOCS, OLSN, and OLSP bits in TOCR1m (m = A).

PSYE Bit (PWM Synchronous Output Enable)

This bit enables or disables toggle output synchronized with the PWM period from the MTIOC3A pin.

Table 19.30 Output Level Select Function

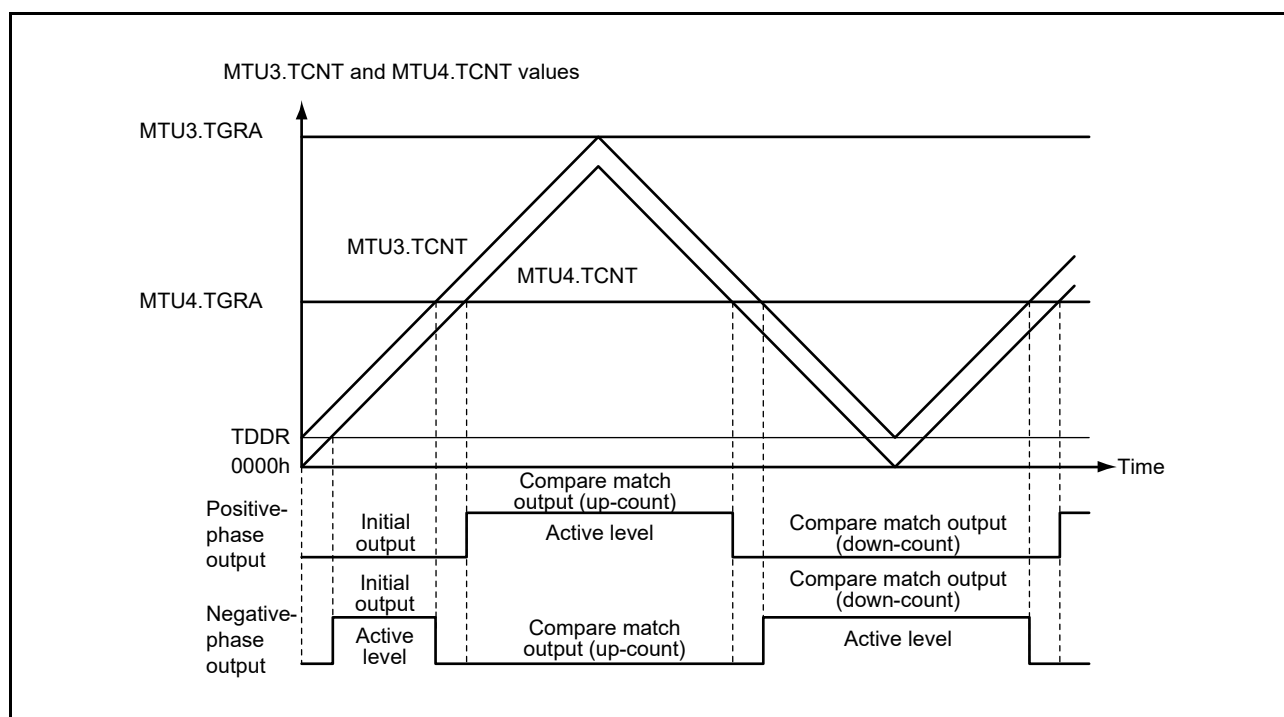
Bit 0	Function			
OLSP	Initial Output	Active Level	Compare Match Output	
			Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 19.31 Output Level Select Function

Bit 1	Function			
OLSN	Initial Output	Active Level	Compare Match Output	
			Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Figure 19.2 shows an example of output in complementary PWM mode (one phase) when OLSN = 1 and OLSP = 1.

**Figure 19.2 Example of Output in Complementary PWM Mode**

19.2.22 Timer Output Control Register 2A (TOCR2A)

Address(es): MTU.TOCR2A 0009 520Fh

b7	b6	b5	b4	b3	b2	b1	b0
BF[1:0]	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OLS1P	Output Level Select 1P*1, *2	This bit selects the output level on MTIOC3B in reset-synchronized PWM mode and complementary PWM mode. Refer to Table 19.32.	R/W
b1	OLS1N	Output Level Select 1N*1, *2	This bit selects the output level on MTIOC3D in reset-synchronized PWM mode and complementary PWM mode. Refer to Table 19.33.	R/W
b2	OLS2P	Output Level Select 2P*1, *2	This bit selects the output level on MTIOC4A in reset-synchronized PWM mode and complementary PWM mode. Refer to Table 19.34.	R/W
b3	OLS2N	Output Level Select 2N*1, *2	This bit selects the output level on MTIOC4C in reset-synchronized PWM mode and complementary PWM mode. Refer to Table 19.35.	R/W
b4	OLS3P	Output Level Select 3P*1, *2	This bit selects the output level on MTIOC4B in reset-synchronized PWM mode and complementary PWM mode. Refer to Table 19.36.	R/W
b5	OLS3N	Output Level Select 3N*1, *2	This bit selects the output level on MTIOC4D in reset-synchronized PWM mode and complementary PWM mode. Refer to Table 19.37.	R/W
b7, b6	BF[1:0]	TOLBR Buffer Transfer Timing Select	These bits select the timing for transferring data from TOLBRm to TOCR2m. Refer to Table 19.38 for details.	R/W

m = A

Note 1. Setting the TOCR1m.TOCS bit to 1 makes this bit setting valid.

Note 2. If dead-time is not generated, the negative-phase output is the exact inverse of the positive-phase output. In this case, only the OLSiP bits are valid (i = 1 to 3).

TOCR2A controls inversion of PWM output level in complementary PWM mode and reset-synchronized PWM mode. The initial output is selected while the counter is stopped.

Table 19.32 MTIOCnB Output Level Select Function

Bit 0	Function			
OLS1P	Initial Output	Active Level	Compare Match Output	
			Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

n = 3

Table 19.33 MTIOCnD Output Level Select Function

Bit 1	Function			
OLS1N	Initial Output	Active Level	Compare Match Output	
			Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

n = 3

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 19.34 MTIOCnA Output Level Select Function

Bit 2	Function			
OLS2P	Initial Output	Active Level	Compare Match Output	
			Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

n = 4

Table 19.35 MTIOCnC Output Level Select Function

Bit 3	Function			
OLS2N	Initial Output	Active Level	Compare Match Output	
			Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

n = 4

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 19.36 MTIOCnB Output Level Select Function

Bit 4	Function			
OLS3P	Initial Output	Active Level	Compare Match Output	
			Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

n = 4

Table 19.37 MTIOCnD Output Level Select Function

Bit 5	Function			
OLS3N	Initial Output	Active Level	Compare Match Output	
			Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

n = 4

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 19.38 Setting of TOCR2m.BF[1:0] Bits

Bit 7	Bit 6	Description	
BF[1]	BF[0]	Complementary PWM Mode	Reset-Synchronized PWM Mode
0	0	Does not transfer data from the buffer register (TOLBRm) to TOCR2m.	Does not transfer data from the buffer register (TOLBRm) to TOCR2m.
0	1	Transfers data from the buffer register (TOLBRm) to TOCR2m at the crest of the MTUn.TCNT count.	Transfers data from the buffer register (TOLBRm) to TOCR2m when MTUk.TCNT or MTUn.TCNT is cleared.
1	0	Transfers data from the buffer register (TOLBRm) to TOCR2m at the trough of the MTUn.TCNT count.	Setting prohibited
1	1	Transfers data from the buffer register (TOLBRm) to TOCR2m at the crest and trough of the MTUn.TCNT count.	Setting prohibited

n = 4; k = 3; m = A

19.2.23 Timer Output Level Buffer Register A (TOLBRA)

Address(es): MTU.TOLBRA 0009 5236h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OLS1P	Output Level Select 1P	Specify the buffer value to be transferred to the OLS1P bit in TOCR2m.	R/W
b1	OLS1N	Output Level Select 1N	Specify the buffer value to be transferred to the OLS1N bit in TOCR2m.	R/W
b2	OLS2P	Output Level Select 2P	Specify the buffer value to be transferred to the OLS2P bit in TOCR2m.	R/W
b3	OLS2N	Output Level Select 2N	Specify the buffer value to be transferred to the OLS2N bit in TOCR2m.	R/W
b4	OLS3P	Output Level Select 3P	Specify the buffer value to be transferred to the OLS3P bit in TOCR2m.	R/W
b5	OLS3N	Output Level Select 3N	Specify the buffer value to be transferred to the OLS3N bit in TOCR2m.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

m = A

TOLBRA is buffer register for TOCR2A and specify the PWM output level in complementary PWM mode and reset-synchronized PWM mode.

Figure 19.3 shows an example of the PWM output level setting procedure in buffer operation.

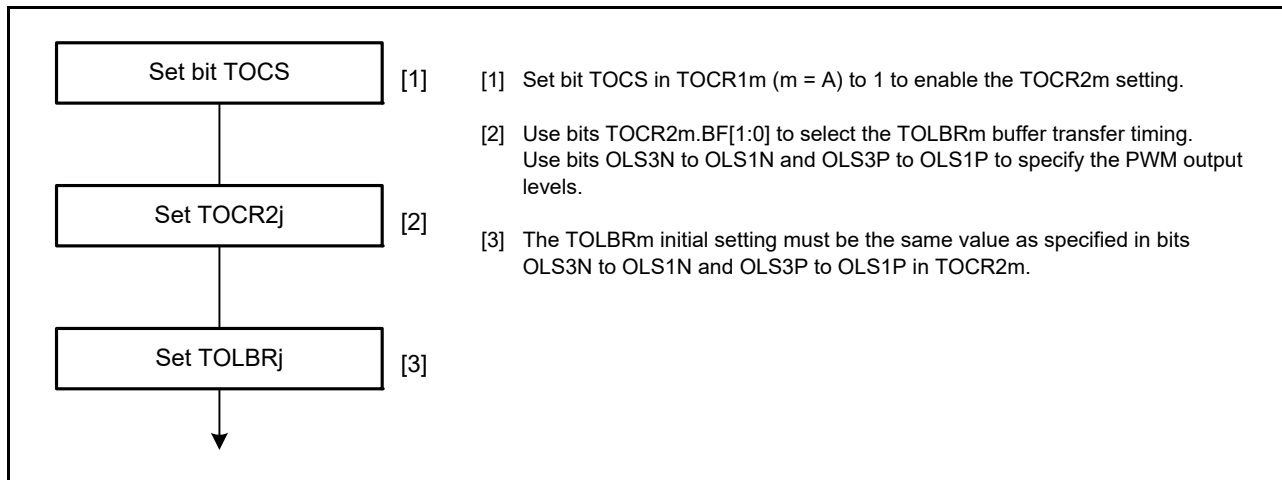


Figure 19.3 Example of PWM Output Level Setting Procedure in Buffer Operation

19.2.24 Timer Gate Control Register A (TGCRA)

Address(es): MTU.TGCRA 0009 520Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	BDC	N	P	FB	WF	VF	UF
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	UF	Output Phase Switch	These bits turn on or off the positive-phase/negative-phase output. The setting of these bits is valid only when the FB bit is set to 1. In this case, the setting of b0 to b2 is used instead of the external input. Refer to Table 19.39.	R/W
b1	VF			R/W
b2	WF			R/W
b3	FB	External Feedback Signal Enable	0: Output is switched by external input (input sources are TGRA, TGRB, and TGRC input capture signals in MTU0) 1: Output is switched by software (TGCRA's UF, VF, and WF settings)	R/W
b4	P	Positive-Phase Output (P) Control	0: Level output 1: Reset-synchronized PWM or complementary PWM output	R/W
b5	N	Negative-Phase Output (N) Control	0: Level output 1: Reset-synchronized PWM or complementary PWM output	R/W
b6	BDC	Brushless DC Motor	0: Ordinary output 1: Functions of this register are made effective	R/W
b7	—	Reserved	This bit is read as 1. The write value should be 1.	R/W

TGCRA controls the output waveform necessary for brushless DC motor control in reset-synchronized PWM mode and complementary PWM mode. TGCRA register settings are ineffective for anything other than complementary PWM mode and reset-synchronized PWM mode.

UF, VF, and WF Bits (Output Phase Switch)

The setting of these bits is valid only when the FB bit is set to 1. In this case, the setting of b0 to b2 is used instead of the external input. Refer to Table 19.39 for details.

FB Bit (External Feedback Signal Enable)

This bit selects whether the positive-/negative-phase output is switched automatically with the TGRA, TGRB, and TGRC input capture signals in MTU0 or by writing 0 or 1 to bits 2 to 0 in TGCRA.

When the TGCRA.FB bit is 0, output of MTU3 and MTU4 can be switched with the TGRA, TGRB, and TGRC input capture signals in MTU0.

P Bit (Positive-Phase Output (P) Control)

This bit selects the level output or the reset-synchronized PWM/complementary PWM output for the positive-phase output pins (MTIOC3B, MTIOC4A, and MTIOC4B pins).

N Bit (Negative-Phase Output (N) Control)

This bit selects the level output or the reset-synchronized PWM/complementary PWM output for the negative-phase output pins (MTIOC3D, MTIOC4C, and MTIOC4D pins).

BDC Bit (Brushless DC Motor)

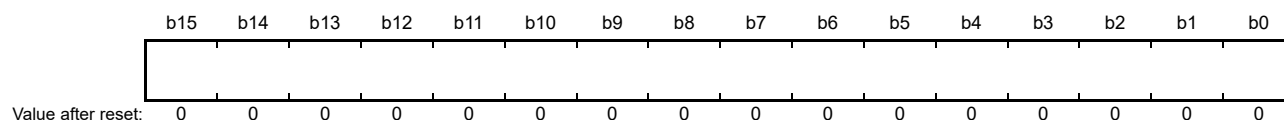
This bit selects whether to make the functions of TGCRA effective or ineffective.

Table 19.39 Output Level Select Function

Bit 2	Bit 1	Bit 0	Function					
WF	VF	UF	MTIOC3B	MTIOC4A	MTIOC4B	MTIOC3D	MTIOC4C	MTIOC4D
			U Phase	V Phase	W Phase	U Phase	V Phase	W Phase
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
0	0	1	ON	OFF	OFF	OFF	OFF	ON
0	1	0	OFF	ON	OFF	ON	OFF	OFF
0	1	1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
1	0	1	ON	OFF	OFF	OFF	ON	OFF
1	1	0	OFF	OFF	ON	ON	OFF	OFF
1	1	1	OFF	OFF	OFF	OFF	OFF	OFF

19.2.25 Timer Subcounter A (TCNTSA)

Address(es): MTU.TCNTSA 0009 5220h



Note: TCNTSA must not be accessed in 8 bits; it should be accessed in 16 bits.

TCNTSA is a 16-bit read-only counter used only in complementary PWM mode.

The initial value of TCNTSA after a reset is 0000h.

19.2.26 Timer Period Data Register A (TCDRA)

Address(es): MTU.TCDRA 0009 5214h



Note: TCDRA must not be accessed in 8 bits; it should be accessed in 16 bits.

TCDRA is a 16-bit readable/writable register used only in complementary PWM mode. Set half the PWM carrier period as the TCDRA value. The TCDRA register is constantly compared with the TCNTSA counter in complementary PWM mode, respectively. When a match occurs, the TCNTSA counter switches the count direction (down-count to up-count). The initial value of TCDRA after a reset is FFFFh.

19.2.27 Timer Period Buffer Register A (TCBRA)

Address(es): MTU.TCBRA 0009 5222h



Note: TCBRA must not be accessed in 8 bits; it should be accessed in 16 bits.

TCBRA is a 16-bit readable/writable register, used only in complementary PWM mode, that function as buffer register for TCDRA. The TCBRA value is transferred to TCDRA with the transfer timing set in TMDR1. The initial value of TCBRA after a reset is FFFFh.

19.2.28 Timer Dead Time Data Register A (TDDRA)

Address(es): MTU.TDDRA 0009 5216h



Note: TDDRA must not be accessed in 8 bits; it should be accessed in 16 bits.

TDDRA is a 16-bit readable/writable register, used only in complementary PWM mode, that specify the MTU3.TCNT and MTU4.TCNT counter offset value. In complementary PWM mode, when the MTU3.TCNT and MTU4.TCNT counters are cleared and then restarted, the TDDRA value is loaded into the MTU3.TCNT counter and the count operation starts. The initial value of TDDRA after a reset is FFFFh.

19.2.29 Timer Dead Time Enable Register A (TDERA)

Address(es): MTU.TDERA 0009 5234h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDER
Value after reset:	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	TDER	Dead Time Enable	0: No dead time is generated 1: Dead time is generated* ¹	R/(W)
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. TDDRA must be set to 1 or a larger value.

TDERA controls dead time generation in complementary PWM mode. The MTU has one TDER for MTU3. TDERA should be modified only while TCNT stops.

TDER Bit (Dead Time Enable)

This bit specifies whether to generate dead time.

[Clearing condition]

- When 0 is written to TDER after reading TDER = 1

19.2.30 Timer Buffer Transfer Set Register A (TBTERA)

Address(es): MTU.TBTERA 0009 5232h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	BTE[1:0]	
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b1, b0	BTE[1:0]	Buffer Transfer Disable and Interrupt Skipping Link Setting	These bits enable or disable transfer from the buffer registers* ¹ used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping function 1. For details, refer to Table 19.40.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Applicable buffer registers (TBTERA):
MTU3.TGRC, MTU3.TGRD, MTU4.TGRC, MTU4.TGRD, and TCBRA

TBTERA enables or disables transfer from the buffer registers used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping 1 operation.

Table 19.40 Setting of TBTERA.BTE[1:0] Bits

Bit 1	Bit 0	
BTE[1]	BTE[0]	Description
0	0	Enables transfer from the buffer registers to the temporary registers* ¹ and does not link the transfer with interrupt skipping function 1.
0	1	Disables transfer from the buffer registers to the temporary registers.
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping function 1.* ²
1	1	Setting prohibited

Note 1. Data is transferred according to the MD3 to MD0 bit setting in TMDR1. For details, refer to section 19.3.8, Complementary PWM Mode.

Note 2. When interrupt skipping is disabled the T3AEN and T4VEN bits are set to 0 in the timer interrupt skipping set register (TITCR1A) or the skipping count set bits (T3ACOR and T4VCOR) in TITCR1A are set to 0), be sure to disable link of buffer transfer with interrupt skipping (set the BTE1 bit in the timer buffer transfer set register (TBTERA) to 0). If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

19.2.31 Timer Waveform Control Register A (TWCRA)

Address(es): MTU.TWCRA 0009 5260h

b7	b6	b5	b4	b3	b2	b1	b0
CCE	—	—	—	—	—	—	WRE
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	WRE	Waveform Retain Enable	0: Initial values specified in TOCR1A and TOCR2A are output 1: Initial output is inhibited	R/(W)
b6 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	CCE	Compare Match Clear Enable ^{*1}	0: Counters are not cleared at MTU3.TGRA compare match 1: Counters are cleared at MTU3.TGRA compare match	R/(W)

Note 1. Do not set to 1 when complementary PWM mode 1 is not selected.

TWCRA controls the output waveform when synchronous counter clearing occurs in MTU3.TCNT and MTU4.TCNT in complementary PWM mode and specifies whether to clear the counters at MTU3.TGRA compare match.

The CCE bit and WRE bit in TWCRA should be modified only while TCNT stops.

WRE Bit (Waveform Retain Enable)

This bit selects the waveform output when synchronous counter clearing occurs in complementary PWM mode.

The initial output is inhibited with this function only when synchronous clearing occurs within the Tb interval at the trough in complementary PWM mode. When synchronous clearing occurs outside this interval, the initial values specified in TOCR1A and TOCR2A are output regardless of the WRE bit setting. The initial values specified in TOCR1A and TOCR2A are also output when synchronous clearing occurs in the Tb interval at the trough immediately after MTU3.TCNT and MTU4.TCNT start operation.

For the Tb interval at the trough in complementary PWM mode, refer to Figure 19.47.

[Setting condition]

- When 1 is written to the WRE bit after reading WRE = 0

CCE Bit (Compare Match Clear Enable)

This bit specifies whether to clear counters at MTU3.TGRA compare match in complementary PWM mode.

[Setting condition]

- When 1 is written to CCE after reading CCE = 0

19.2.32 Noise Filter Control Register n (NFCRn) (n = 0 to 4, C)

- MTU0.NFCR0, MTU1.NFCR1, MTU2.NFCR2, MTU3.NFCR3, MTU4.NFCR4

Address(es): MTU0.NFCR0 0009 5290h, MTU1.NFCR1 0009 5291h, MTU2.NFCR2 0009 5292h, MTU3.NFCR3 0009 5293h, MTU4.NFCR4 0009 5294h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	NFCS[1:0]		NFDEN	NFCEN	NFBEN	NFAEN
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFAEN	Noise Filter A Enable	0: The noise filter for the MTIOCnA pin is disabled. 1: The noise filter for the MTIOCnA pin is enabled.	R/W
b1	NFBEN	Noise Filter B Enable	0: The noise filter for the MTIOCnB pin is disabled. 1: The noise filter for the MTIOCnB pin is enabled.	R/W
b2	NFCEN	Noise Filter C Enable*1	0: The noise filter for the MTIOCnC pin is disabled. 1: The noise filter for the MTIOCnC pin is enabled.	R/W
b3	NFDEN	Noise Filter D Enable*1	0: The noise filter for the MTIOCnD pin is disabled. 1: The noise filter for the MTIOCnD pin is enabled.	R/W
b5, b4	NFCS[1:0]	Noise Filter Clock Select	b5 b4 0 0: PCLKB/1 0 1: PCLKB/8 1 0: PCLKB/32 1 1: Clock source for counting	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. These bits are reserved in MTU1 and MTU2. These bits are read as 0 and writing to them has no effect.

The NFCRn register (n = 0 to 4) sets the noise filter function of input capture pins for the corresponding channel.

NFAEN Bit (Noise Filter A Enable)

This bit disables or enables the noise filter for input from the MTIOCnA pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFBEN Bit (Noise Filter B Enable)

This bit disables or enables the noise filter for input from the MTIOCnB pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFCEN Bit (Noise Filter C Enable)

This bit disables or enables the noise filter for input from the MTIOCnC pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFDEN Bit (Noise Filter D Enable)

This bit disables or enables the noise filter for input from the MTIOCnD pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. When setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval before setting the input-capture function. When the NFCS[1:0] bits are set to 11b, i.e. selecting the external clock as the source to drive counting, wait for two cycles of the external clock before setting the input capture function.

- MTU0.NFCRC

Address(es): MTU0.NFCRC 0009 5299h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	NFCS[1:0]	NFDEN	NFCEN	NFBEN	NFAEN	
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFAEN	Noise Filter A Enable	0: The noise filter for the MTCLKA pin is disabled. 1: The noise filter for the MTCLKA pin is enabled.	R/W
b1	NFBEN	Noise Filter B Enable	0: The noise filter for the MTCLKB pin is disabled. 1: The noise filter for the MTCLKB pin is enabled.	R/W
b2	NFCEN	Noise Filter C Enable	0: The noise filter for the MTCLKC pin is disabled. 1: The noise filter for the MTCLKC pin is enabled.	R/W
b3	NFDEN	Noise Filter D Enable	0: The noise filter for the MTCLKD pin is disabled. 1: The noise filter for the MTCLKD pin is enabled.	R/W
b5, b4	NFCS[1:0]	Noise Filter Clock Select	b5 b4 0 0: PCLKB/1 0 1: PCLKB/2 1 0: PCLKB/8 1 1: PCLKB/32	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The NFCRC register sets the noise filter function of external clock pins common to each channel.

NFAEN Bit (Noise Filter A Enable)

This bit disables or enables the noise filter for input from the MTCLKA pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFBEN Bit (Noise Filter B Enable)

This bit disables or enables the noise filter for input from the MTCLKB pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFCEN Bit (Noise Filter C Enable)

This bit disables or enables the noise filter for input from the MTCLKC pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFDEN Bit (Noise Filter D Enable)

This bit disables or enables the noise filter for input from the MTCLKD pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. After setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval to set the input capture function.

19.2.33 Noise Filter Control Register 5 (NFCR5)

Address(es): MTU5.NFCR5 0009 5295h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	NFCS[1:0]		—	NFWEN	NFVEN	NFUEN
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFUEN	Noise Filter U Enable	0: The noise filter for the MTIC5U pin is disabled. 1: The noise filter for the MTIC5U pin is enabled.	R/W
b1	NFVEN	Noise Filter V Enable	0: The noise filter for the MTIC5V pin is disabled. 1: The noise filter for the MTIC5V pin is enabled.	R/W
b2	NFWEN	Noise Filter W Enable	0: The noise filter for the MTIC5W pin is disabled. 1: The noise filter for the MTIC5W pin is enabled.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5, b4	NFCS[1:0]	Noise Filter Clock Select	b5 b4 0 0: PCLKB/1 0 1: PCLKB/8 1 0: PCLKB/32 1 1: Clock source for counting	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NFUEN Bit (Noise Filter U Enable)

This bit disables or enables the noise filter for input from the MTIC5U pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFVEN Bit (Noise Filter V Enable)

This bit disables or enables the noise filter for input from the MTIC5V pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFWEN Bit (Noise Filter W Enable)

This bit disables or enables the noise filter for input from the MTIC5W pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. When setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

19.2.34 Timer A/D Conversion Start Request Control Register (TADCR)

Address(es): MTU4.TADCR 0009 5240h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]	—	—	—	—	—	—	—	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ITB4VE	TCIV4 Interrupt Skipping Link Enable*1, *2, *3	0: A/D conversion start request TRG4BN and TCIV4 interrupt skipping 1 are not linked 1: A/D conversion start request TRG4BN and TCIV4 interrupt skipping 1 are linked	R/W
b1	ITB3AE	TGIA3 Interrupt Skipping Link Enable*1, *2, *3	0: A/D conversion start request TRG4BN and TGIA3 interrupt skipping 1 are not linked 1: A/D conversion start request TRG4BN and TGIA3 interrupt skipping 1 are linked	R/W
b2	ITA4VE	TCIV4 Interrupt Skipping Link Enable*1, *2, *3	0: A/D conversion start request TRG4AN and TCIV4 interrupt skipping 1 are not linked 1: A/D conversion start request TRG4AN and TCIV4 interrupt skipping 1 are linked	R/W
b3	ITA3AE	TGIA3 Interrupt Skipping Link Enable*1, *2, *3	0: A/D conversion start request TRG4AN and TGIA3 interrupt skipping 1 are not linked 1: A/D conversion start request TRG4AN and TGIA3 interrupt skipping 1 are linked	R/W
b4	DT4BE	Down-Count TRG4BN Enable*3	0: A/D conversion start requests (TRG4BN) disabled during MTU4.TCNT down-count operation 1: A/D conversion start requests (TRG4BN) enabled during MTU4.TCNT down-count operation	R/W
b5	UT4BE	Up-Count TRG4BN Enable	0: A/D conversion start requests (TRG4BN) disabled during MTU4.TCNT up-count operation 1: A/D conversion start requests (TRG4BN) enabled during MTU4.TCNT up-count operation	R/W
b6	DT4AE	Down-Count TRG4AN Enable*3	0: A/D conversion start requests (TRG4AN) disabled during MTU4.TCNT down-count operation 1: A/D conversion start requests (TRG4AN) enabled during MTU4.TCNT down-count operation	R/W
b7	UT4AE	Up-Count TRG4AN Enable	0: A/D conversion start requests (TRG4AN) disabled during MTU4.TCNT up-count operation 1: A/D conversion start requests (TRG4AN) enabled during MTU4.TCNT up-count operation	R/W
b13 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15, b14	BF[1:0]	MTU4.TADCOBRA/TADCOBRB Transfer Timing Select	Refer to Table 19.41 for details. These bits specify the transfer timing from MTU4.TADCOBRA and MTU4.TADCOBRB to MTU4.TADCORA and MTU4.TADCORB.	R/W

Note: MTU4.TADCR must not be accessed in 8 bits; it should be accessed in 16 bits.

Note 1. Set to 0 when interrupt skipping is disabled (the T3AEN and T4VEN bits in TITCR1A are set to 0 or the T3ACOR and T4VCOR bits in TITCR1A are set to 0).

Note 2. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D conversion start requests will not be issued.

Note 3. Set to 0 when complementary PWM mode is not selected.

TADCR enables or disables A/D conversion start requests and specifies whether to link A/D conversion start requests with interrupt skipping function.

Table 19.41 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU4)

Bit 15	Bit 14	Description			
BF[1]	BF[0]	In Complementary PWM Mode	In Reset-Synchronized PWM Mode	In PWM Mode 1	In Normal Mode
0	0	Data is not transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB).	Data is not transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB).	Data is not transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB).	Data is not transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB).
0	1	Data is transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB) at the crest of the MTU4.TCNT.	Data is transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB) when a compare match occurs between MTU3.TCNT and MTU3.TGRA.	Data is transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB) when a compare match occurs between MTU4.TCNT and MTU4.TGRA.	Data is transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB) when a compare match occurs between MTU4.TCNT and MTU4.TGRA.
1	0	Data is transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB) at the trough of the MTU4.TCNT.	Setting prohibited	Setting prohibited	Setting prohibited
1	1	Data is transferred from the cycle set buffer register (MTU4.TADCOBRA, MTU4.TADCOBRB) to the cycle set register (MTU4.TADCORA, MTU4.TADCORB) at the crest and trough of the MTU4.TCNT.	Setting prohibited	Setting prohibited	Setting prohibited

19.2.35 Timer A/D Conversion Start Request Cycle Set Register m (TADCORm) (m = A, B)

Address(es): MTU4.TADCORA 0009 5244h, MTU4.TADCORB 0009 5246h



Note: TADCORA and TADCORB must not be accessed in 8 bits; it should be accessed in 16 bits.

Note 1. When the A/D conversion start request delaying function linked with skipping function 1 (for details, refer to section 19.3.9 (5), A/D Conversion Start Request Delaying Function Linked with Interrupt Skipping Function 1) is used, the value of this register should be 0002h to TCDRA setting – 2 in MTU4.

Note 2. When interrupt skipping function 2 is used and the difference between the TADCORA value and the TADCORB value is small, the skipping count may not be counted correctly and the A/D conversion start request may not be generated with the expected timing in some cases. The TADCORA and TADCORB values should satisfy the following conditions.

(1) When skipping function 2 is specified with the skipping count set to 0

- The difference between the TADCORA and TADCORB values should be equal to or greater than 4.
- The TADCORA compare interval should be equal to or greater than 4 PCLKB cycles (the TADCORA update value should be the previous value + 4 or greater, or previous value – 4 or smaller).
- The TADCORB compare interval should be equal to or greater than 4 PCLKB cycles (the TADCORB update value should be the previous value + 4 or greater, or previous value – 4 or smaller).

(2) When skipping function 2 is specified with the skipping count set to 1 or greater

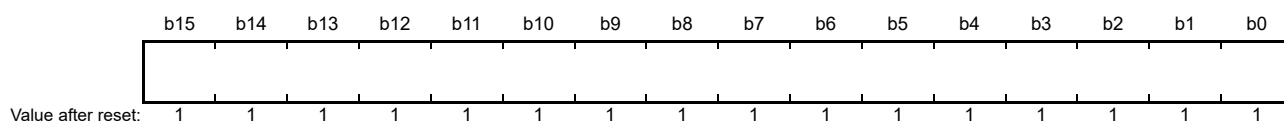
- The difference between the TADCORA and TADCORB values should be equal to or greater than 2.
- The TADCORB compare interval should be equal to or greater than 2 PCLKB cycles (the TADCORB update value should be the previous value + 2 or greater, or previous value – 2 or smaller)

TADCORA and TADCORB are 16-bit readable/writable registers that issue a corresponding A/D conversion start request when the MTUn.TCNT (n = 4) count reaches the value in TADCORA or TADCORB.

MTUn.TADCORA and TADCORB are initialized to FFFFh by a reset.

19.2.36 Timer A/D Conversion Start Request Cycle Set Buffer Register m (TADCOBRm) (m = A, B)

Address(es): MTU4.TADCOBRA 0009 5248h, MTU4.TADCOBRB 0009 524Ah



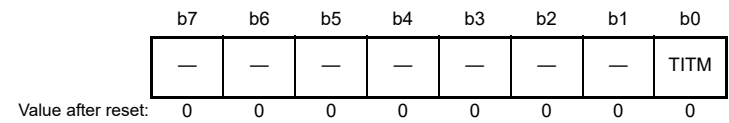
Note: TADCOBRA and TADCOBRB must not be accessed in 8 bits; it should be accessed in 16 bits.

TADCOBRA and TADCOBRB are 16-bit readable/writable registers whose values are transferred to TADCORA and TADCORB, respectively, when the crest or trough of the MTUn.TCNT count is reached.

TADCOBRA and TADCOBRB are initialized to FFFFh by a reset.

19.2.37 Timer Interrupt Skipping Mode Register A (TITMRA)

Address(es): MTU.TITMRA 0009 523Ah



Bit	Symbol	Bit Name	Description	R/W
b0	TITM	Interrupt Skipping Function Select	Selects one of the two types of interrupt skipping functions. 0: Selects interrupt skipping function 1*1 1: Selects interrupt skipping function 2*2	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Setting the TITCR1A register enables interrupt skipping function 1.
Note 2. Setting the TITCR2A register enables interrupt skipping function 2.

TITMRA is used to select either of two skipping functions for the TITMRA register.

19.2.38 Timer Interrupt Skipping Set Register 1A (TITCR1A)

Address(es): MTU.TITCR1A 0009 5230h

b7	b6	b5	b4	b3	b2	b1	b0
T3AEN	T3ACOR[2:0]			T4VEN	T4VCOR[2:0]		
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	T4VCOR[2:0]	TCIV4 Interrupt Skipping Count Setting	These bits specify the TCIV4 interrupt skipping count within the range from 0 to 7.* ¹ For details, refer to Table 19.42.	R/W
b3	T4VEN	T4VEN	0: TCIV4 interrupt skipping disabled 1: TCIV4 interrupt skipping enabled	R/W
b6 to b4	T3ACOR[2:0]	TGIA3 Interrupt Skipping Count Setting	These bits specify the TGIA3 interrupt skipping count within the range from 0 to 7.* ¹ For details, refer to Table 19.43.	R/W
b7	T3AEN	T3AEN	0: TGIA3 interrupt skipping disabled 1: TGIA3 interrupt skipping enabled	R/W

Note 1. When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed.
Before changing the interrupt skipping count, be sure to set the TITCR1A.T3AEN and TITCR1A.T4VEN bits to 0 to clear the skipping counter (TITCNT1A).

The TITCR1A register enables or disables interrupt skipping and specify the interrupt skipping count. This setting is valid only while the TITMRA.TITM bit is set to 0; when the TITMRA.TITM bit is set to 1, the setting in the TITCR1A register is cleared.

Table 19.42 Setting of Interrupt Skipping Count by T4VCOR[2:0] Bits

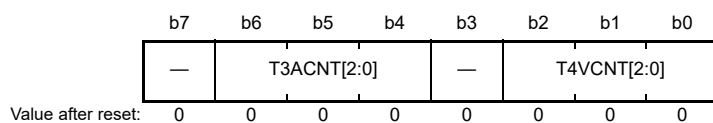
Bit 2	Bit 1	Bit 0	Description
T4VCOR[2]	T4VCOR[1]	T4VCOR[0]	
0	0	0	Does not skip TCIV4 interrupts.
0	0	1	Sets the TCIV4 interrupt skipping count to 1.
0	1	0	Sets the TCIV4 interrupt skipping count to 2.
0	1	1	Sets the TCIV4 interrupt skipping count to 3.
1	0	0	Sets the TCIV4 interrupt skipping count to 4.
1	0	1	Sets the TCIV4 interrupt skipping count to 5.
1	1	0	Sets the TCIV4 interrupt skipping count to 6.
1	1	1	Sets the TCIV4 interrupt skipping count to 7.

Table 19.43 Setting of Interrupt Skipping Count by T3ACOR[2:0] Bits

Bit 6	Bit 5	Bit 4	Description
T3ACOR[2]	T3ACOR[1]	T3ACOR[0]	
0	0	0	Does not skip TGIA3 interrupts.
0	0	1	Sets the TGIA3 interrupt skipping count to 1.
0	1	0	Sets the TGIA3 interrupt skipping count to 2.
0	1	1	Sets the TGIA3 interrupt skipping count to 3.
1	0	0	Sets the TGIA3 interrupt skipping count to 4.
1	0	1	Sets the TGIA3 interrupt skipping count to 5.
1	1	0	Sets the TGIA3 interrupt skipping count to 6.
1	1	1	Sets the TGIA3 interrupt skipping count to 7.

19.2.39 Timer Interrupt Skipping Counter 1A (TITCNT1A)

Address(es): MTU.TITCNT1A 0009 5231h



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	T4VCNT[2:0]	TCIV4 Interrupt Counter	While the T4VEN bit in TITCR1A is set to 1, the count in these bits is incremented every time a TCIV4 interrupt occurs.	R
b3	—	Reserved	This bit is read as 0.	R
b6 to b4	T3ACNT[2:0]	TGIA3 Interrupt Counter	While the T3AEN bit in TITCR1A is set to 1, the count in these bits is incremented every time a TGIA3 interrupt occurs.	R
b7	—	Reserved	This bit is read as 0.	R

Note: To clear the TITCNT1A, set the TITCR1A.T3AEN and TITCR1A.T4VEN bits to 0.

TITCNT1A is 8-bit readable/writable counters. TITCNT1A retains their values even after stopping the count operation of MTU3.TCNT and MTU4.TCNT.

T4VCNT[2:0] Bits (TCIV4 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRA is 1
- When the T4VEN bit in TITCR1A is set to 0
- When the T4VCOR[2:0] bits in TITCR1A are set to 000b
- When the T4VCNT[2:0] bits in TITCNT1A match the T4VCOR[2:0] bits in TITCR1A

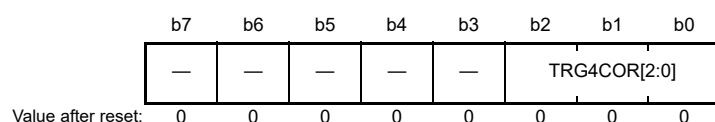
T3ACNT[2:0] Bits (TGIA3 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRA is 1
- When the T3AEN bit in TITCR1A is set to 0
- When the T3ACOR[2:0] bits in TITCR1A are set to 000b
- When the T3ACNT[2:0] bits in TITCNT1A match the T3ACOR[2:0] bits in TITCR1A

19.2.40 Timer Interrupt Skipping Set Register 2A (TITCR2A)

Address(es): MTU.TITCR2A 0009 523Bh



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TRG4COR[2:0]	TRG4AN/TRG4BN Interrupt Skipping Count Setting	These bits specify the TRG4AN/TRG4BN interrupt skipping count within the range from 0 to 7. For details, refer to Table 19.44.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

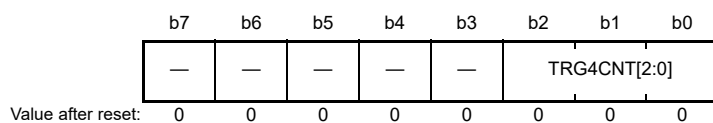
TITCR2A specifies the interrupt skipping count for TRG4AN and TRG4BN. This setting is valid only while TITMRA is set to 1.

Table 19.44 Setting of Interrupt Skipping Count by TRG4COR[2:0] Bits

Bit 2	Bit 1	Bit 0	Description
TRG4COR[2]	TRG4COR[1]	TRG4COR[0]	
0	0	0	Does not skip TRG4AN and TRG4BN interrupts.
0	0	1	Sets the TRG4AN and TRG4BN interrupt skipping count to 1.
0	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 2.
0	1	1	Sets the TRG4AN and TRG4BN interrupt skipping count to 3.
1	0	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 4.
1	0	1	Sets the TRG4AN and TRG4BN interrupt skipping count to 5.
1	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 6.
1	1	1	Sets the TRG4AN and TRG4BN interrupt skipping count to 7.

19.2.41 Timer Interrupt Skipping Counter 2A (TITCNT2A)

Address(es): MTU.TITCNT2A 0009 523Ch



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TRG4CNT[2:0]	TRG4AN/TRG4BN Interrupt Counter	These bits start counting from the value set in TRG4COR[2:0] and the count decrements every time TRG4AN or TRG4BN is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.	R
b7 to b3	—	Reserved	These bits are read as 0.	R

TITCNT2A starts counting from the values set in the TRG4COR[2:0] bits and the count decrements every time TRG4AN or TRG4BN (TITCNT2A) is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.

TRG4CNT[2:0] Bits (TRG4AN/TRG4BN Interrupt Counter)

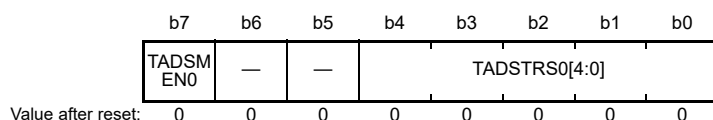
These bits start counting from the value set in the TRG4COR[2:0] bits and the count decrements every time a TRG4AN or TRG4BN interrupt is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.

[Clearing conditions]

- When the TITM bit in TITMRA is 0
- When the TRG4COR[2:0] bits in TITCR2A are set to 000b
- When the count of TRG4AN and TRG4BN occurrence matches the TRG4COR[2:0] value in TITCR2A

19.2.42 A/D Conversion Start Request Select Register 0 (TADSTRGR0)

Address(es): MTU.TADSTRGR0 0009 5D30h

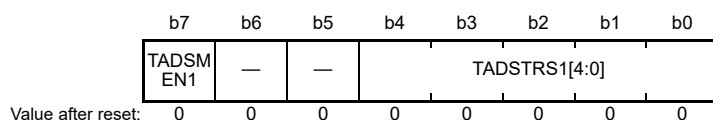


Bit	Symbol	Bit Name	Description	R/W
b4 to b0	TADSTRS0[4:0]	A/D Conversion Start Request Select for ADSM0 Pin Output Frame Synchronization Signal Generation	These bits select the A/D conversion start request for generating the frame synchronization signal to be output from the ADSM0 pin. Refer to Table 19.45 for the relationship between the A/D conversion start request and settings. Settings other than those listed in Table 19.45 are prohibited.	R/W
b6 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TADSMEN0	ADSM0 Pin Output Enable	0: ADSM0 pin output disabled 1: ADSM0 pin output enabled	R/W

The TADSTRGR0 register selects the A/D conversion start request for generating the A/D conversion start request frame synchronization signal to be output from the ADSM0 pin.

19.2.43 A/D Conversion Start Request Select Register 1 (TADSTRGR1)

Address(es): MTU.TADSTRGR1 0009 5D32h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	TADSTRS1[4:0]	A/D Conversion Start Request Select for ADSM1 Pin Output Frame Synchronization Signal Generation	These bits select the A/D conversion start request for generating the frame synchronization signal to be output from the ADSM1 pin. Refer to Table 19.45 for the relationship between the A/D conversion start request and settings. Settings other than those listed in Table 19.45 are prohibited.	R/W
b6 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TADSMEN1	ADSM1 Pin Output Enable	0: ADSM1 pin output disabled 1: ADSM1 pin output enabled	R/W

The TADSTRGR1 register selects the A/D conversion start request for generating the A/D conversion start request frame synchronization signal to be output from the ADSM1 pin.

Table 19.45 Settings of A/D Conversion Start Request for Generating Frame Synchronization Signal (n = 0, 1)

TADSTRSn[4:0]					Source	Descriptions
[4]	[3]	[2]	[1]	[0]		
0	0	0	0	0	—	Source not selected
0	0	0	0	1	TRGA0N	Compare match/input capture in MTU0.TGRA
0	0	0	1	0	TRGA1N	Compare match/input capture in MTU1.TGRA
0	0	0	1	1	TRGA2N	Compare match/input capture in MTU2.TGRA
0	0	1	0	0	TRGA3N	Compare match/input capture in MTU3.TGRA
0	0	1	0	1	TRGA4N	Compare match/input capture in MTU4.TGRA or MTU4.TCNT underflow (trough) in complementary PWM mode
0	1	0	0	0	TRG0N	Compare match in MTU0.TGRE
0	1	0	0	1	TRG4AN	Compare match between MTU4.TADCORA and MTU4.TCNT
0	1	0	1	0	TRG4BN	Compare match between MTU4.TADCORB and MTU4.TCNT
0	1	1	0	0	TRG4ABN	Compare match between MTU4.TADCORA and MTU4.TCNT or compare match between MTU4.TADCORB and MTU4.TCNT (Interrupt skipping function 2 used)

19.3 Operation

19.3.1 Basic Functions

Each channel has TCNT and TGR register. TCNT performs up-counting, and is also capable of free-running operation, periodic counting, and external event counting.

Each TGR register can be used as an input capture register or an output compare register.

(1) Counter Operation

When one of bits CST0 to CST4 in the TSTRA register, and bits CSTU5, CSTV5, and CSTW5 in the MTU5.TSTR register is set to 1, TCNT for the corresponding channel begins counting. TCNT can operate as a free-running counter, periodic counter, for example.

(a) Example of Count Operation Setting Procedure

Figure 19.4 shows an example of the count operation setting procedure.

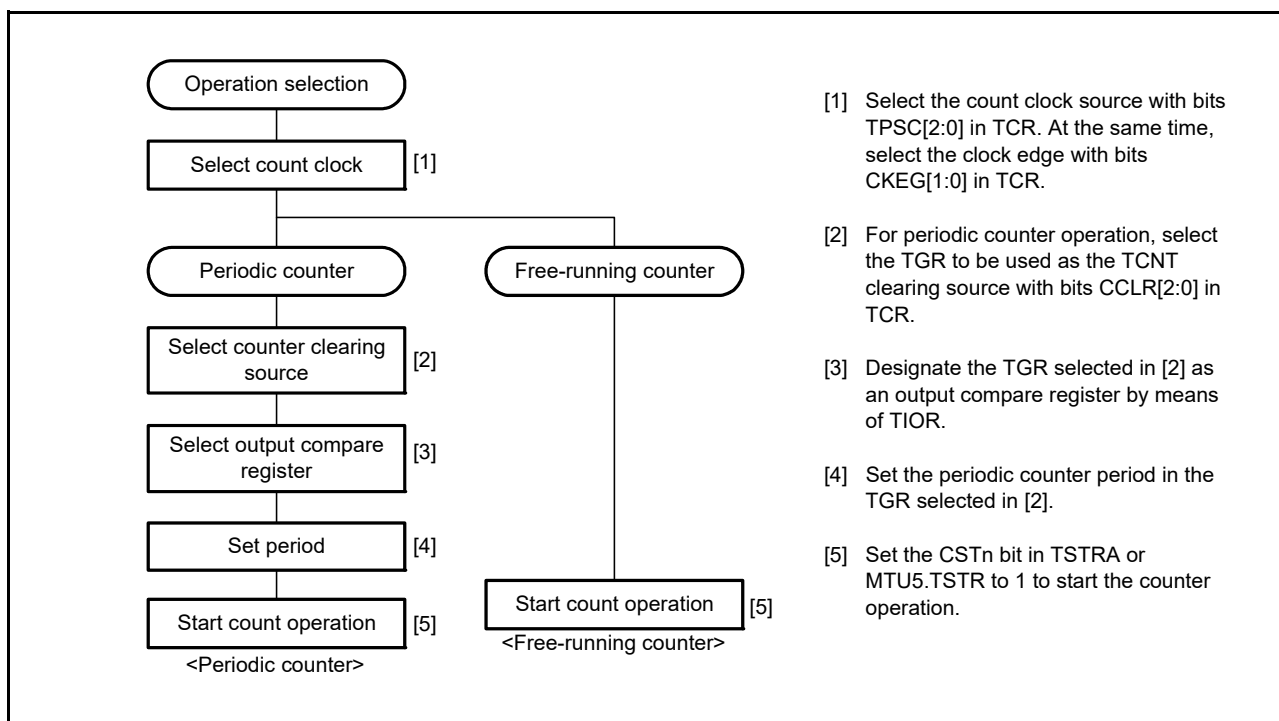


Figure 19.4 Example of Count Operation Setting Procedure

(b) Free-Running Count Operation and Periodic Count Operation

Immediately after a reset, the TCNT counters are all designated as free-running counters. When the CSTn bit in TSTRA or MTU5.TSTR is set to 1, the corresponding TCNT counter starts up-count operation as a free-running counter. When TCNT overflows (from FFFFh to 0000h), an interrupt request is issued to the CPU if the corresponding TIER.TCIEV bit is 1. After an overflow, TCNT starts counting up again from 0000h.

Figure 19.5 illustrates free-running counter operation.

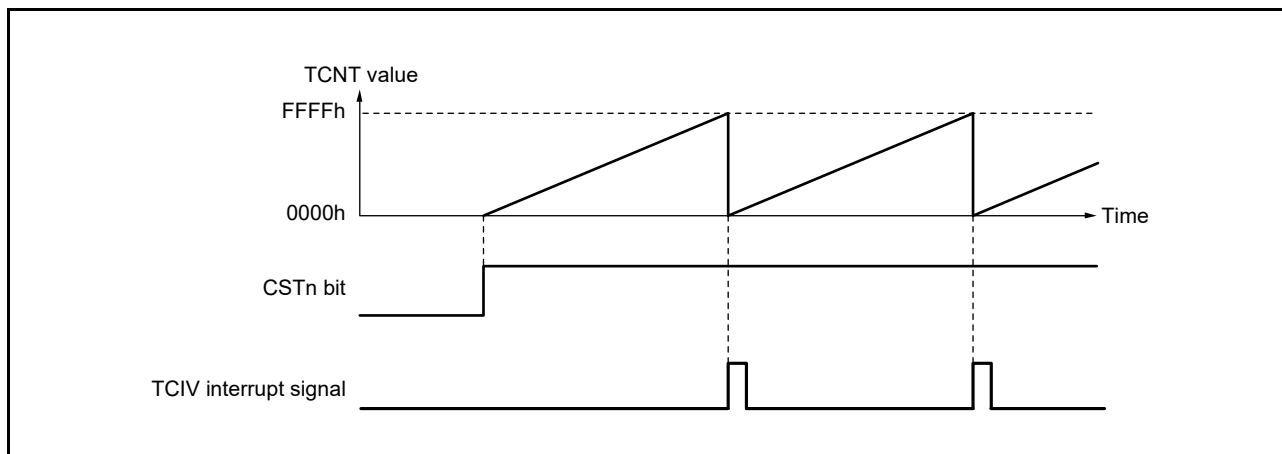


Figure 19.5 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, TCNT for the relevant channel performs periodic count operation. TGR for setting the period is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR[2:0] in TCR. After the settings have been made, TCNT starts up-count operation as a periodic counter when the CSTn bit in TSTRA or MTU5.TSTR is set to 1. When the count matches the value in TGR, TCNT becomes 0000h.

If the value of the corresponding TIER.TGIE bit is 1 at this point, an interrupt request is issued to the CPU. After a compare match, TCNT starts counting up again from 0000h.

Figure 19.6 illustrates periodic counter operation.

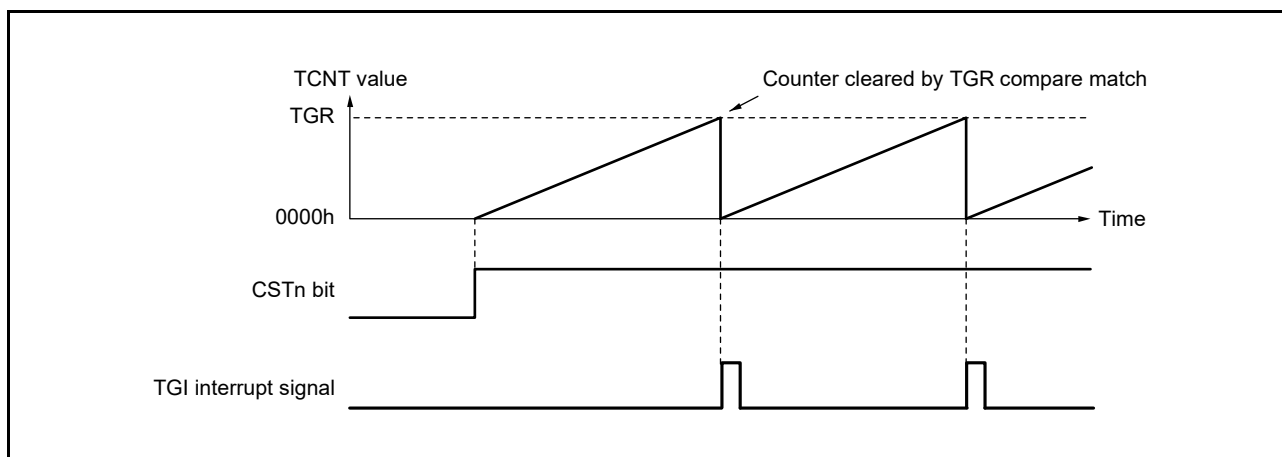


Figure 19.6 Periodic Counter Operation

(2) Waveform Output by Compare Match

Upon compare match, low, high, or toggle output from the corresponding pin can be performed.

(a) Example of Procedure for Setting Waveform Output by Compare Match

Figure 19.7 shows an example of the procedure for setting waveform output by compare match

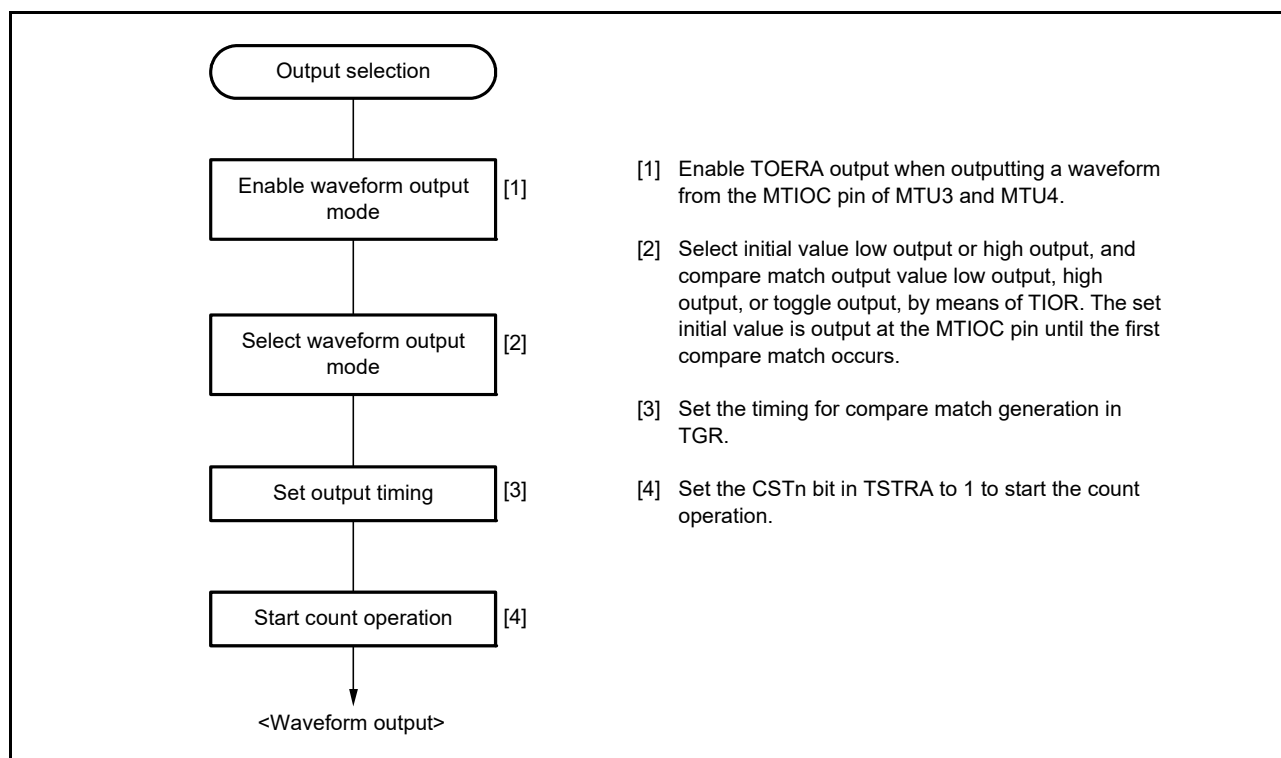


Figure 19.7 Example of Procedure for Setting Waveform Output by Compare Match

(b) Examples of Waveform Output Operation

Figure 19.8 shows an example of low output and high output.

In this example, TCNT has been designated as a free-running counter, and settings have been made so that high is output by compare match A and low is output by compare match B. When the pin level is the same as the specified level, the pin level does not change.

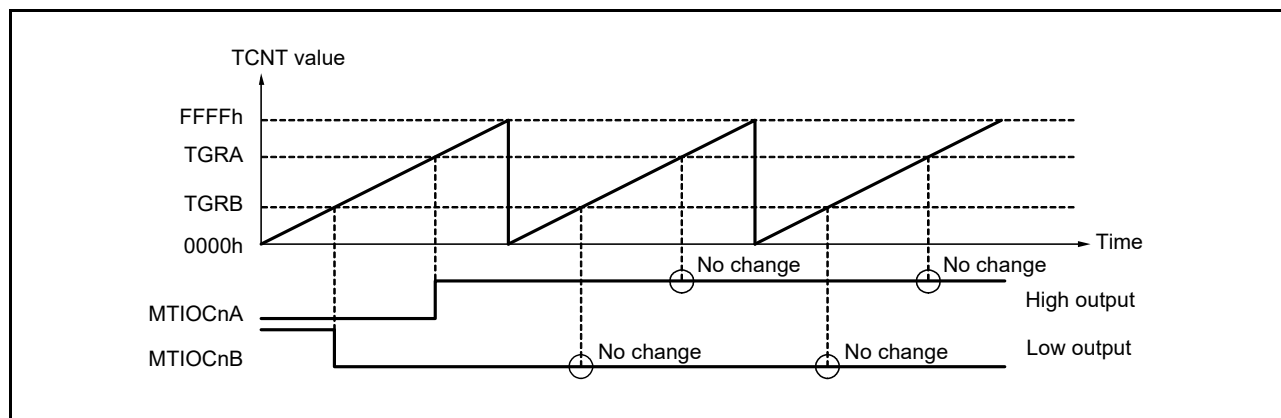


Figure 19.8 Example of low output and high output Operation (n = 0 to 4)

Figure 19.9 shows an example of toggle output.

In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made so that the output is toggled by both compare match A and compare match B.

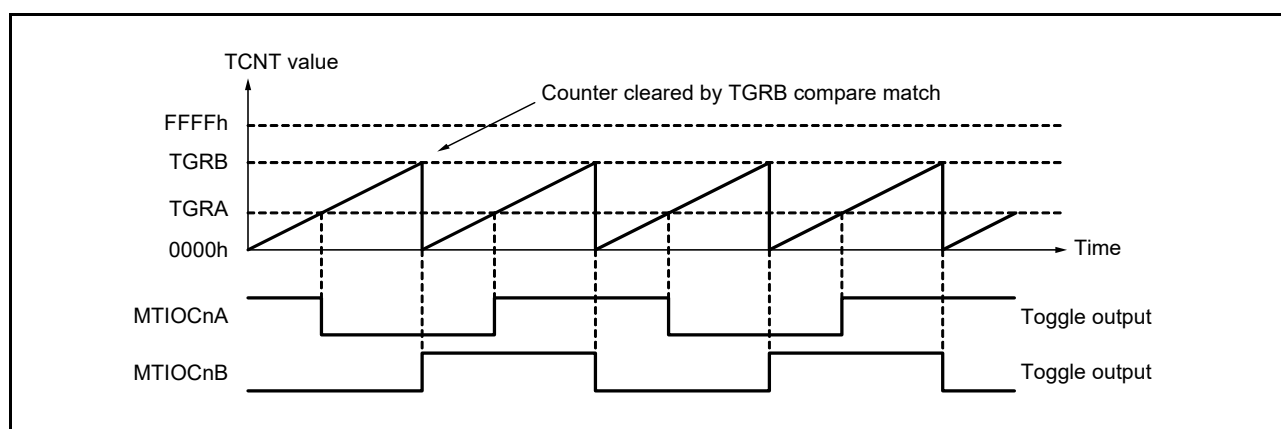


Figure 19.9 Example of Toggle Output Operation (n = 0 to 4)

(3) Input Capture Function

The TCNT value can be transferred to TGR on detection of the MTIOCnm pin (n = 0 to 4; m = A to D) or MTIC5m pin (m = U, V, W) input edge.

The rising edge, falling edge, or both edges can be selected as the detection edge. For MTU0 and MTU1, another channel's count clock or compare match signal can also be specified as the input capture source.

Note: When another channel's count clock is used as the input capture input for MTU0 and MTU1, PCLKB/1 should not be selected as the count clock used for input capture input. Input capture will not be generated if PCLKB/1 is selected.

(a) Example of Input Capture Operation Setting Procedure

Figure 19.10 shows an example of the input capture operation setting procedure.

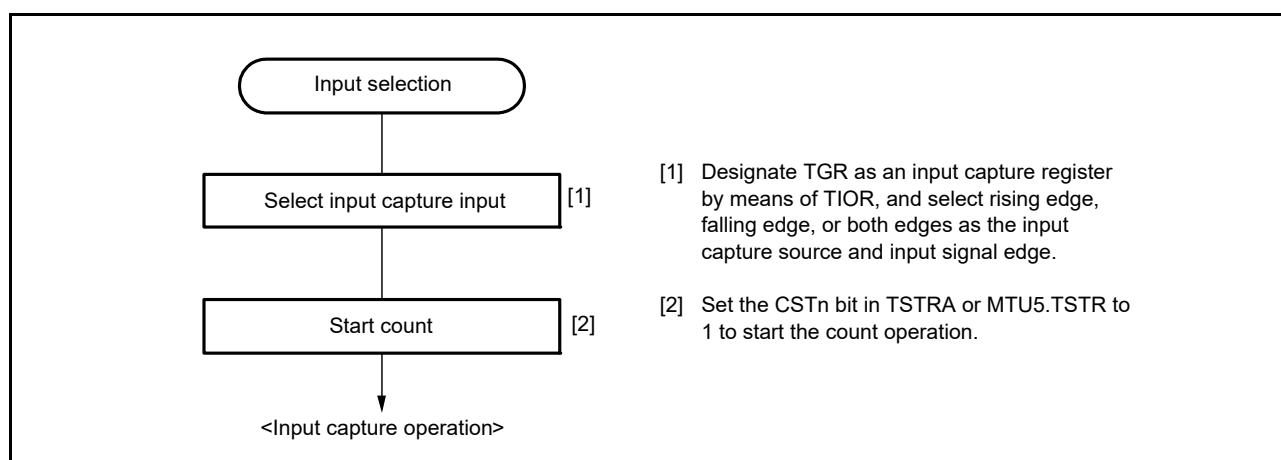


Figure 19.10 Example of Input Capture Operation Setting Procedure

(b) Example of Input Capture Operation

Figure 19.11 shows an example of input capture operation.

In this example, both rising and falling edges have been selected as the MTIOCN_A pin input capture input edge, the falling edge has been selected as the MTIOCN_B pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT. (n = 0 to 4)

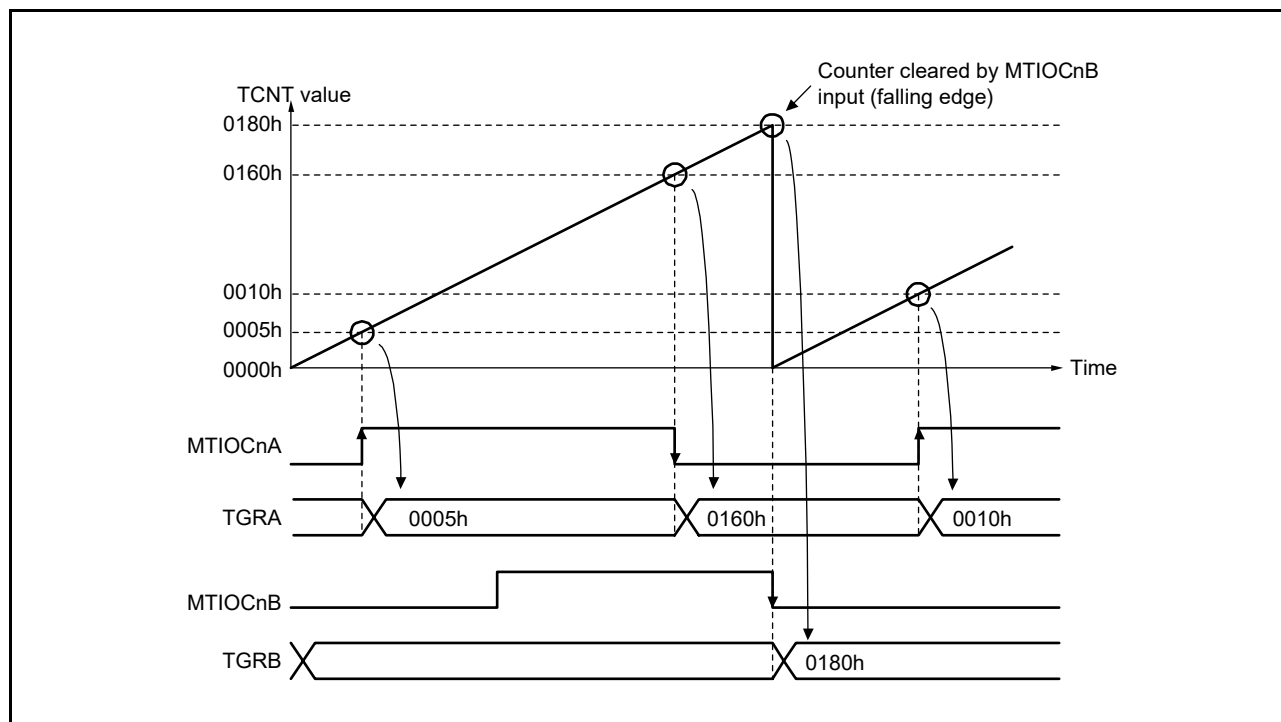


Figure 19.11 Example of Input Capture Operation (n = 0 to 4)

19.3.2 Synchronous Operation

In synchronous operation, the values in multiple TCNT counters can be modified simultaneously (synchronous setting). In addition, multiple TCNT counters can be cleared simultaneously (synchronous clearing) by making the appropriate setting in TCR.

Synchronous operation can increase the number of TGR registers assigned to the same time base.

MTU0 to MTU4 can all be designated for synchronous operation. MTU5 cannot be used for synchronous operation.

(1) Example of Synchronous Operation Setting Procedure

Figure 19.12 shows an example of the synchronous operation setting procedure.

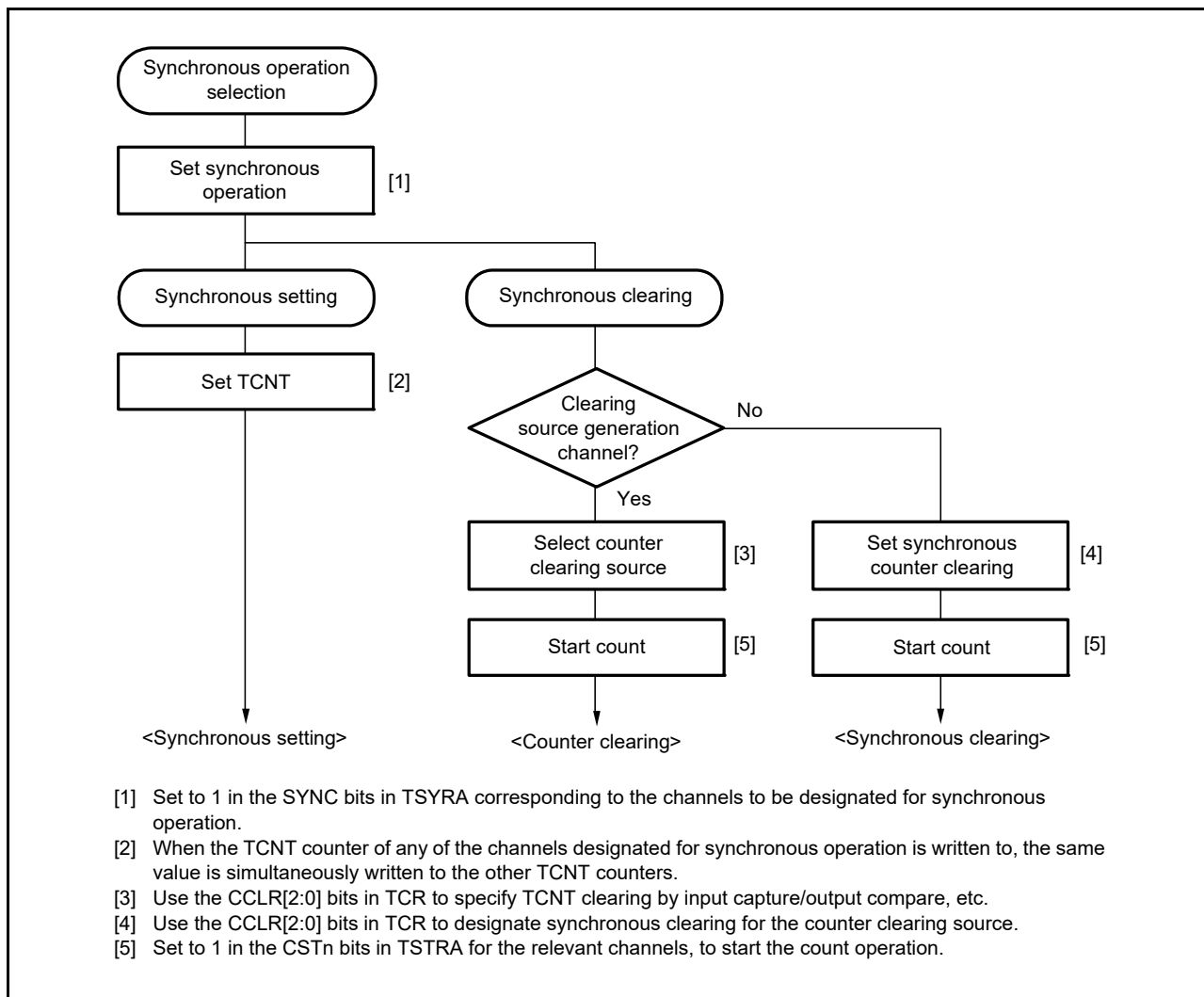


Figure 19.12 Example of Synchronous Operation Setting Procedure

(2) Example of Synchronous Operation

Figure 19.13 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for MTU0 to MTU 2, MTU0.TGRB compare match has been set as the counter clearing source in MTU0, and synchronous clearing has been set for the counter clearing source in MTU1 and MTU2.

Three-phase PWM waveforms are output from pins MTIOC0A, MTIOC1A, and MTIOC2A. At this time, synchronous setting and synchronous clearing by MTU0.TGRB compare match are performed for the TCNT counters in MTU0 to MTU2, and the data set in MTU0.TGRB is used as the PWM period.

For details of PWM modes, refer to section 19.3.5, PWM Modes.

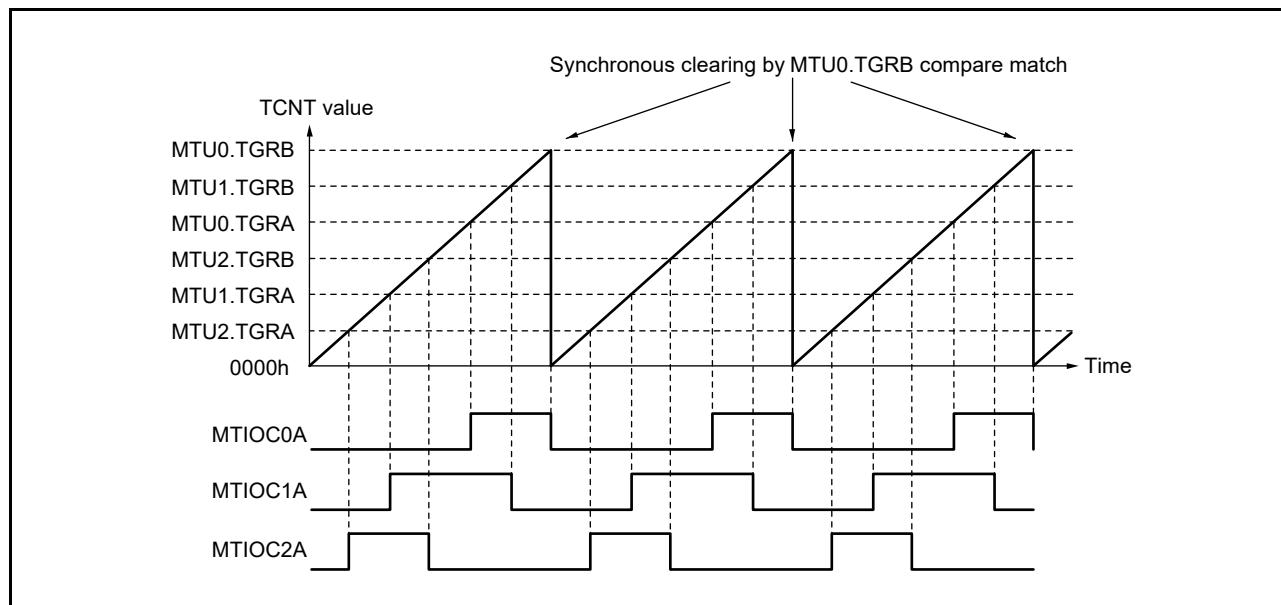


Figure 19.13 Example of Synchronous Operation

19.3.3 Buffer Operation

Buffer operation, provided for MTU0, MTU3, and MTU4, enables TGRC and TGRD to be used as buffer registers. In MTU0, TGRF can also be used as a buffer register.

Buffer operation differs depending on whether TGR has been designated as an input capture register or as a compare match register.

Note: MTU0.TGRE cannot be designated as an input capture register and can only operate as a compare match register.

Table 19.46 shows the register combinations used in buffer operation.

Table 19.46 Register Combinations in Buffer Operation

Channel	Timer General Register	Buffer Register
MTU0	TGRA	TGRC
	TGRB	TGRD
	TGRE	TGRF
MTU3	TGRA	TGRC
	TGRB	TGRD
MTU4	TGRA	TGRC
	TGRB	TGRD

- When TGR is an output compare register

When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register.

This operation is illustrated in Figure 19.14.

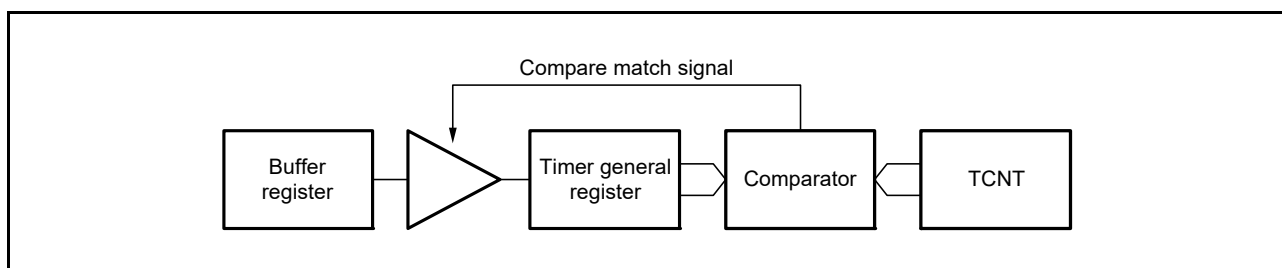


Figure 19.14 Compare Match Buffer Operation

- When TGR is an input capture register

When an input capture occurs, the value in TCNT is transferred to TGR and the value previously held in TGR is transferred to the buffer register.

This operation is illustrated in Figure 19.15.

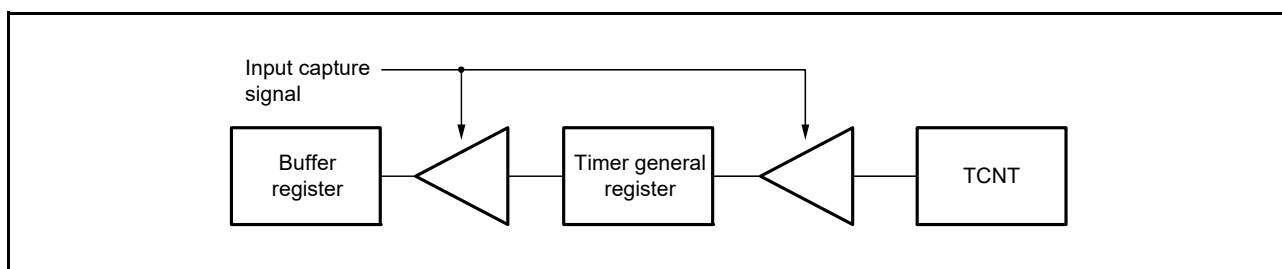


Figure 19.15 Input Capture Buffer Operation

(1) Example of Buffer Operation Setting Procedure

Figure 19.16 shows an example of the buffer operation setting procedure.

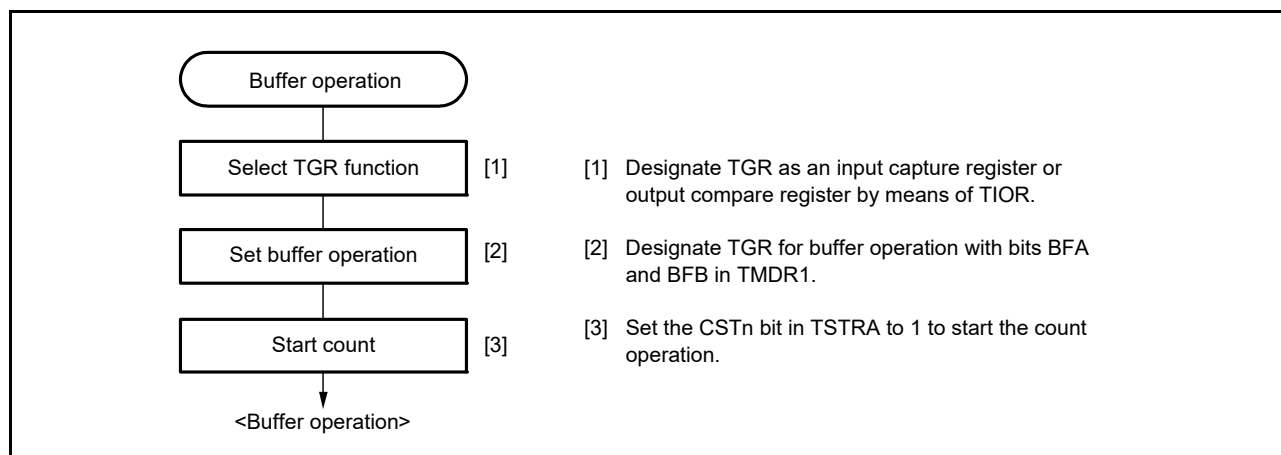


Figure 19.16 Example of Buffer Operation Setting Procedure

(2) Examples of Buffer Operation

(a) When TGR is an Output Compare Register

Figure 19.17 shows an operation example in which PWM mode 1 has been designated for MTU0, and buffer operation has been designated for TGRA and TGRC. The settings used in this example are TCNT clearing by compare match B, high output at compare match A, and low output at compare match B. In this example, the TTSA bit in TBTM is set to 0. As buffer operation has been set, when compare match A occurs, the output changes and the value in buffer register TGRC is simultaneously transferred to timer general register TGRA. This operation is repeated each time compare match A occurs.

For details of PWM modes, refer to section 19.3.5, PWM Modes.

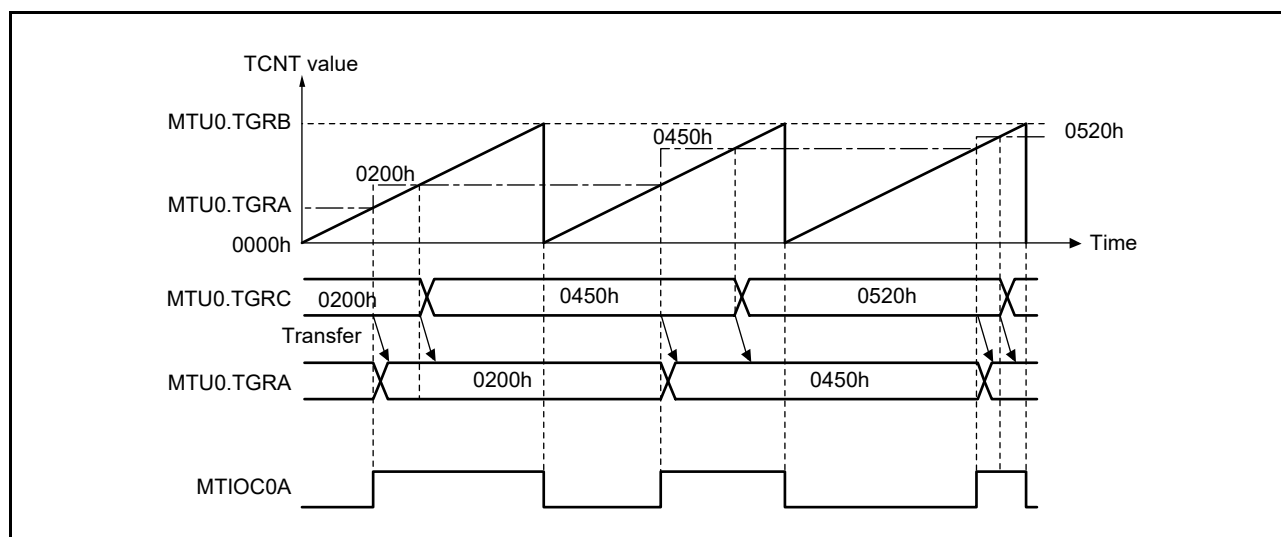


Figure 19.17 Example of Buffer Operation (1)

(b) When TGR is an Input Capture Register

Figure 19.18 shows an operation example in which TGRA has been designated as an input capture register, and buffer operation has been designated for TGRA and TGRC.

Counter clearing by TGRA input capture has been set for TCNT, and both rising and falling edges have been selected as the MTIOCnA pin input capture input edge. (n = 0 to 4)

As buffer operation has been set, when the TCNT value is transferred to TGRA upon occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.

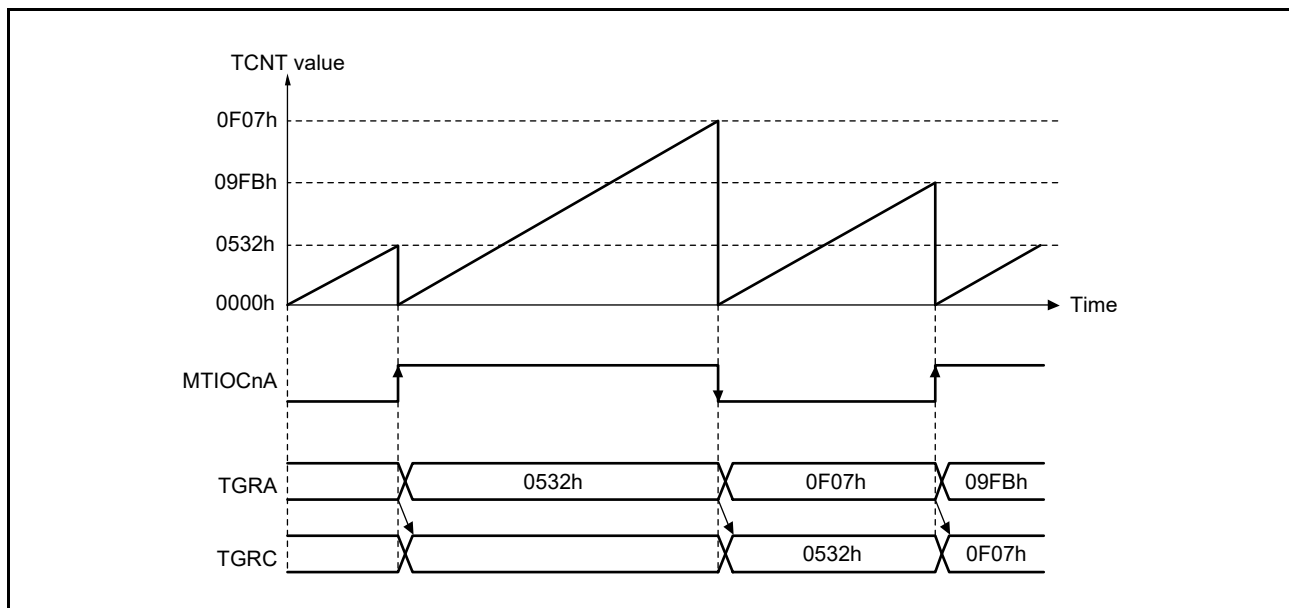


Figure 19.18 Example of Buffer Operation (2) (n = 0 to 4,)

(3) Selecting Timing for Transfer from Buffer Registers to Timer General Registers in Buffer Operation

The timing for transfer from buffer registers to timer general registers can be selected in PWM mode 1 or 2 for MTU0 or in PWM mode 1 for MTU3 and MTU4 by setting the buffer operation transfer mode registers (MTUn.TBTM (n = 0, 3, 4)). Either compare match (value after reset) or TCNT clearing can be selected for the transfer timing. TCNT clearing as transfer timing is one of the following cases.

- When TCNT overflows (FFFFh to 0000h)
- When 0000h is written to TCNT during counting
- When TCNT becomes 0000h under the condition specified in the CCLR[2:0] bits in TCR

Note: TBTM must be modified only while TCNT stops.

Figure 19.19 shows an operation example in which PWM mode 1 is designated for MTU0 and buffer operation is designated for MTU0.TGRA and MTU0.TGRC. The settings used in this example are MTU0.TCNT clearing by compare match B, high output at compare match A, and low output at compare match B. The TTSA bit in MTU0.TBTM is set to 1.

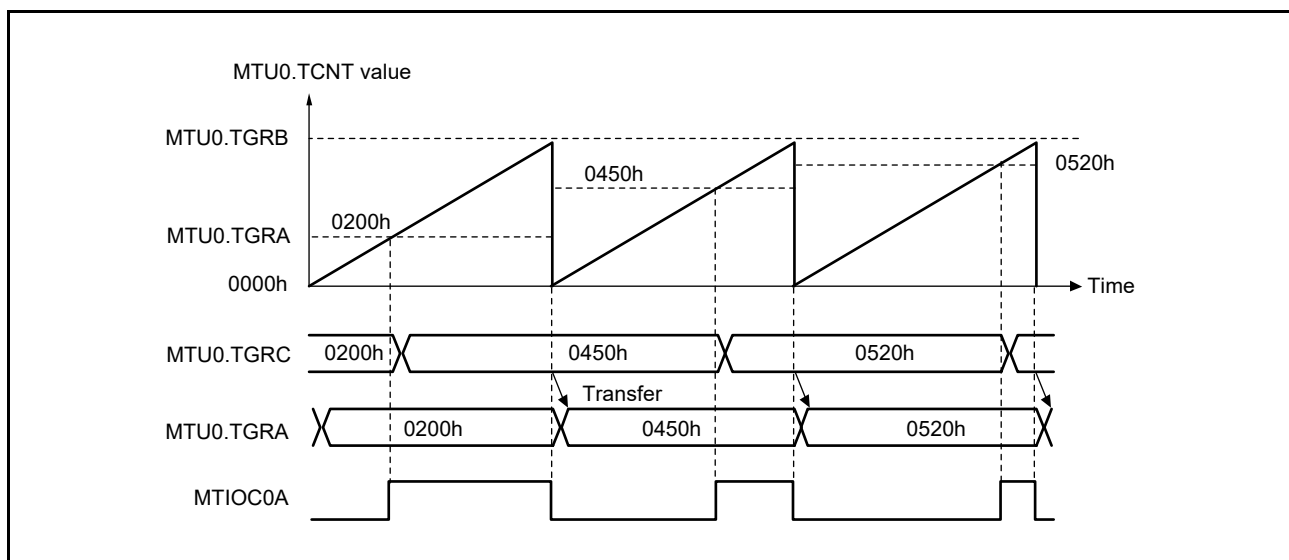


Figure 19.19 Example of Buffer Operation When MTU0.TCNT Clearing is Selected for MTU0.TGRC to MTU0.TGRA Transfer Timing

19.3.4 Cascaded Operation

In cascaded operation, two 16-bit counters in different channels are used together as a 32-bit counter.

There are two functions for connecting MTU1 and MTU2 to use as a 32-bit counter: cascade connection to be set when the MTU1.TMDR3.LWA bit is 0, and cascade connection 32-bit phase counting mode to be set when the MTU1.TMDR3.LWA bit is 1. For details on cascade connection 32-bit phase counting mode, refer to [section 19.3.6.2, Cascade Connection 32-Bit Phase Counting Mode](#). This section describes the cascade connection function to be set when the MTU1.TMDR3.LWA bit is 0.

This function operates when the MTU1.TMDR3.LWA bit is set to 0 and the MTU1.TCR.TPSC[2:0] bits are set so that MTU1.TCNT counts at an overflow/underflow of MTU2.TCNT. Underflow occurs only when the MTU2 to which the lower 16 bits allocated is in phase counting mode.

Table 19.47 shows the register combinations used in cascaded operation.

Note: When phase counting mode is set for MTU1, the count clock setting is invalid and the counters operate independently in phase counting mode.

Table 19.47 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
MTU1 and MTU2	MTU1.TCNT	MTU2.TCNT

For simultaneous input capture of MTU1.TCNT and MTU2.TCNT during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). The input-capture condition is of edges in the signal produced by taking the logical OR of the input level on the main input pin and the input level on the added input pin. Accordingly, if either is at the high level, a change in the level of the other will not produce an edge for detection. For details, refer to (4), Cascaded Operation Example (c). For input capture in cascade connection, refer to [section 19.6.21, Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection](#).

Table 19.48 shows the TICCR setting and input capture input pins.

Table 19.48 TICCR Setting and Input Capture Input Pins

Target Input Capture	TICCR Setting	Input Capture Input Pin
Input capture from MTU1.TCNT to MTU1.TGRA	I2AE bit = 0 (Initial value)	MTIOC1A
	I2AE bit = 1	MTIOC1A, MTIOC2A
Input capture from MTU1.TCNT to MTU1.TGRB	I2BE bit = 0 (Initial value)	MTIOC1B
	I2BE bit = 1	MTIOC1B, MTIOC2B
Input capture from MTU2.TCNT to MTU2.TGRA	I1AE bit = 0 (Initial value)	MTIOC2A
	I1AE bit = 1	MTIOC2A, MTIOC1A
Input capture from MTU2.TCNT to MTU2.TGRB	I1BE bit = 0 (Initial value)	MTIOC2B
	I1BE bit = 1	MTIOC2B, MTIOC1B

(1) Example of Cascaded Operation Setting Procedure

Figure 19.20 shows an example of the cascaded operation setting procedure.

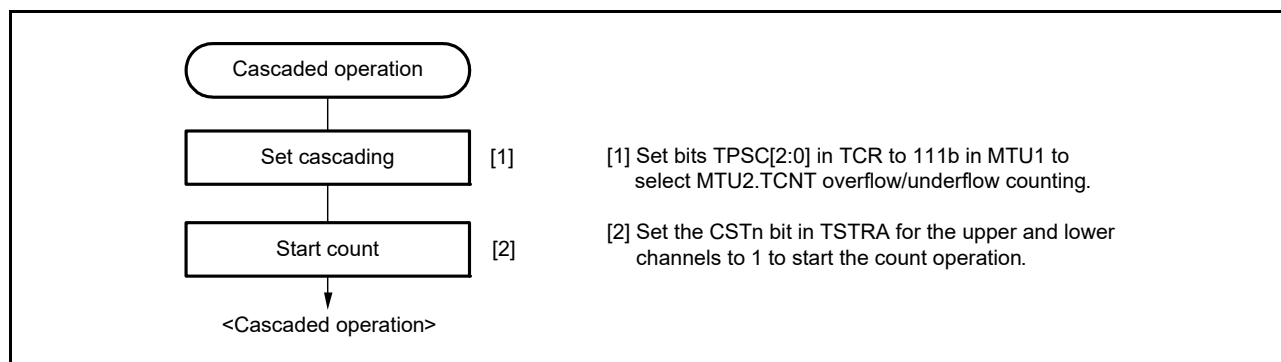


Figure 19.20 Cascaded Operation Setting Procedure

(2) Cascaded Operation Example (a)

Figure 19.21 shows the operation when MTU1.TCNT is set for counting at MTU2.TCNT overflow/underflow and MTU2 is set for phase counting mode 1 while MTU1.TCNT and MTU2.TCNT are cascaded. MTU1.TCNT is incremented by MTU2.TCNT overflow and decremented by MTU2.TCNT underflow.

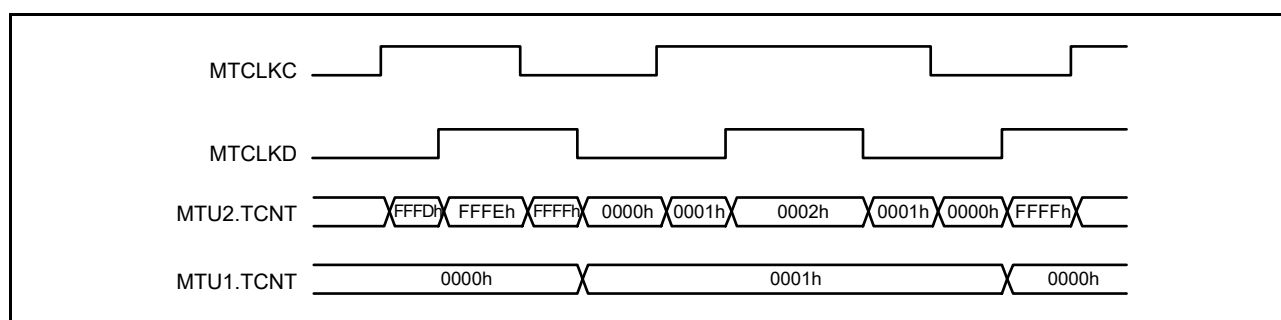


Figure 19.21 Cascaded Operation Example (a)

(3) Cascaded Operation Example (b)

Figure 19.22 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the I2AE bit in TICCRR has been set to 1 to include the MTIOC2A pin in the MTU1.TGRA input capture conditions. In this example, the MTU1.TIOR.IOA[3:0] bits have selected the MTIOC1A rising edge for the input capture timing while the MTU2.TIOR.IOA[3:0] bits have selected the MTIOC2A rising edge for the input capture timing.

Under these conditions, the rising edge of both MTIOC1A and MTIOC2A is used for the MTU1.TGRA input capture condition. For the MTU2.TGRA input capture condition, the MTIOC2A rising edge is used.

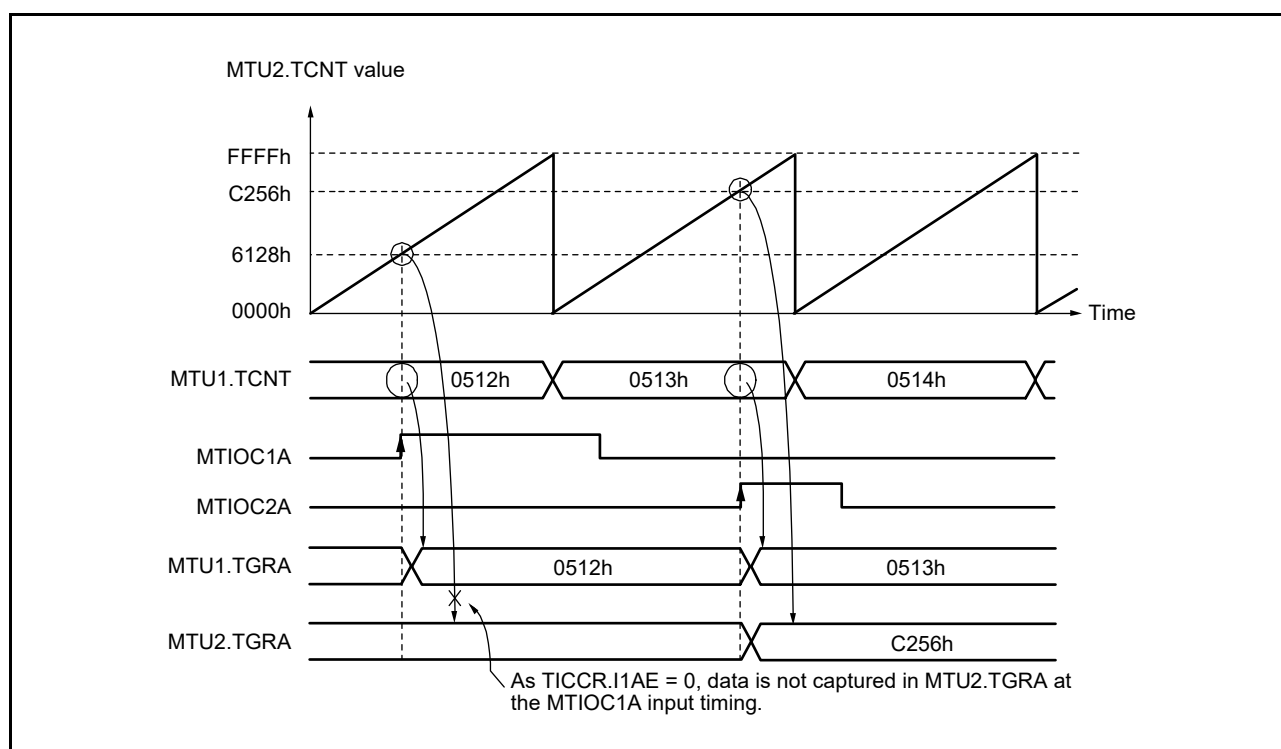


Figure 19.22 Cascaded Operation Example (b)

(4) Cascaded Operation Example (c)

Figure 19.23 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the TICC.R.I2AE and I1AE bits have been set to 1 to include the MTIOC2A and MTIOC1A pins in the MTU1.TGRA and MTU2.TGRA input capture conditions, respectively. In this example, the IOA[3:0] bits in both MTU1.TIOR and MTU2.TIOR have selected both the rising and falling edges for the input capture timing. Under these conditions, the ORed result of MTIOC1A and MTIOC2A input is used for the MTU1.TGRA and MTU2.TGRA input capture conditions.

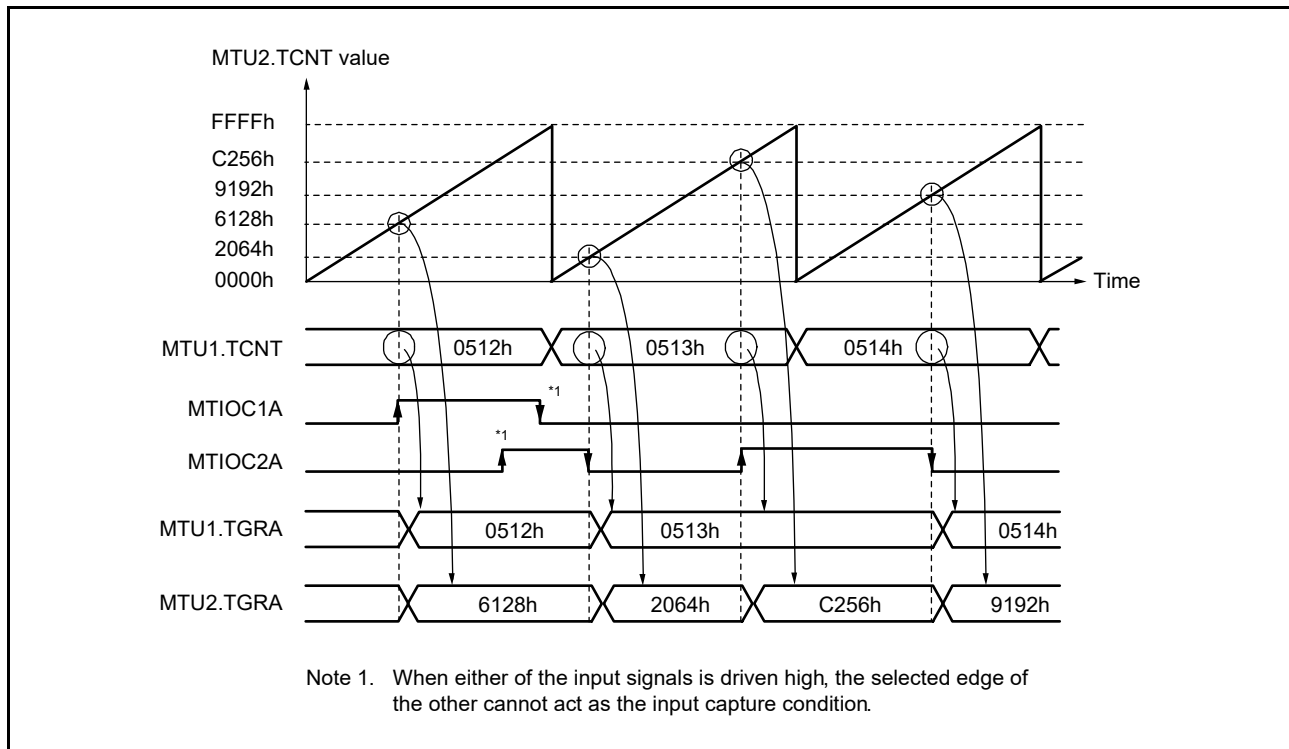


Figure 19.23 Cascaded Operation Example (c)

(5) Cascaded Operation Example (d)

Figure 19.24 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the TICC.R.I2AE bit has been set to 1 to include the MTIOC2A pin in the MTU1.TGRA input capture conditions. In this example, the IOA[3:0] bits in MTU1.TIOR have selected occurrence of MTU0.TGRA compare match or input capture for the input capture timing while the IOA[3:0] bits in MTU2.TIOR have selected the MTIOC2A rising edge for the input capture timing.

Under these conditions, as MTU1.TIOR has selected occurrence of MTU0.TGRA compare match or input capture for the input capture timing, the MTIOC2A edge is not used for MTU1.TGRA input capture condition although the I2AE bit in TICC.R has been set to 1.

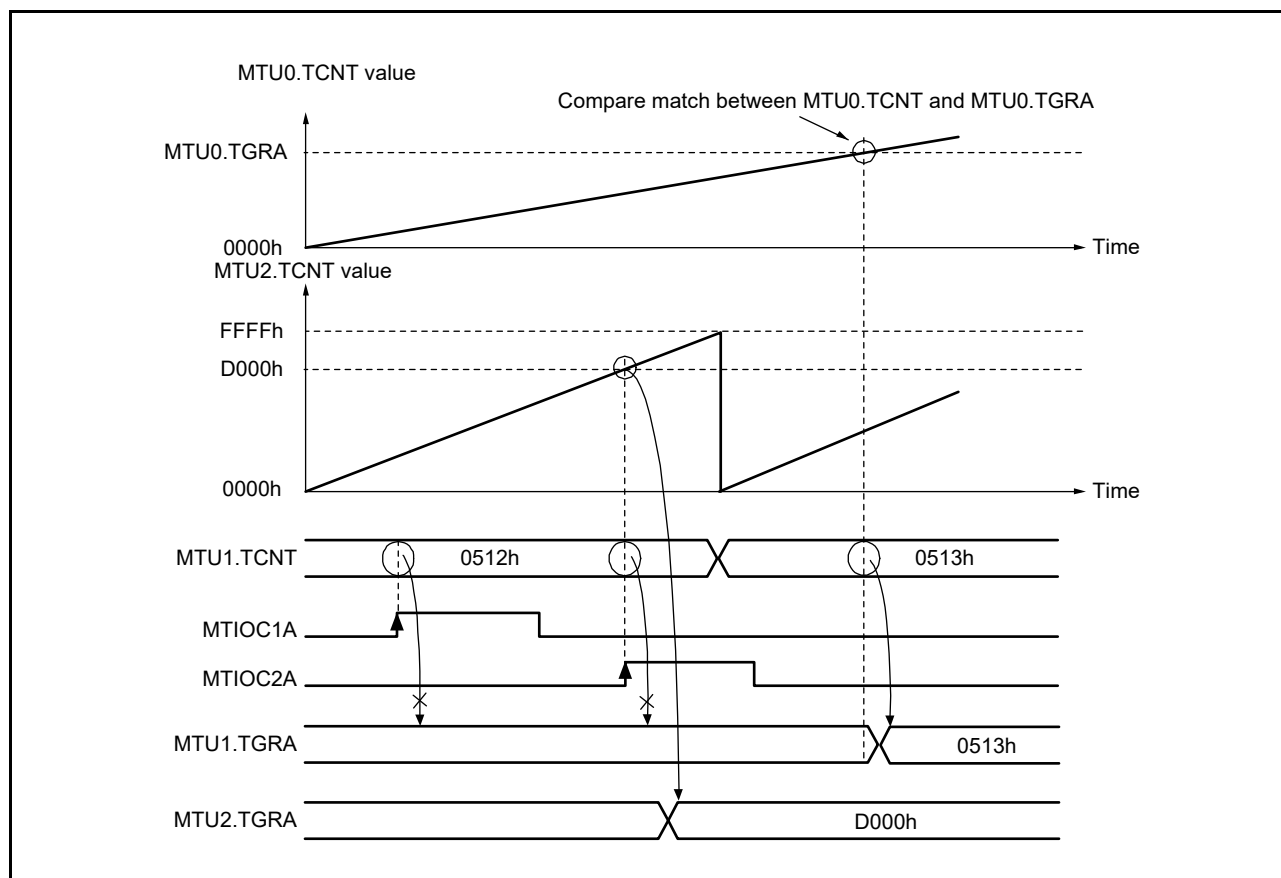


Figure 19.24 Cascaded Operation Example (d)

19.3.5 PWM Modes

PWM modes are provided to output PWM waveforms from the external pins. The output level can be selected as low, high, or toggle output in response to a compare match of each TGR.

PWM waveforms in the range of 0% to 100% duty cycle can be output according to the TGR settings.

By designating TGR compare match as the counter clearing source, the PWM period can be specified in that register.

Every channel except MTU5 can be set to PWM mode independently. Channels set to PWM mode can perform synchronous operation with each other or other channels set to any other mode.

There are two PWM modes as described below.

(a) PWM Mode 1

PWM waveforms are output from the MTIOCnA and MTIOCnC pins by pairing TGRA with TGRB and TGRC with TGRD. The levels specified by the TIOR.IOA[3:0] and IOC[3:0] bits are output from the MTIOCnA and MTIOCnC pins at compare matches A and C, and the level specified by the TIOR.IOB[3:0] and IOD[3:0] bits are output at compare matches B and D ($n = 0$ to 4). The initial output value is set in the TIOR.IOA[3:0] and IOC[3:0] bits. If the values set in paired TGRs are identical, the output value does not change even when a compare match occurs.

In PWM mode 1, PWM waveforms in up to eight phases can be output.

(b) PWM Mode 2

PWM waveform output is generated using one TGR as the period register and the others as duty registers. The level specified in TIOR is output at compare matches. Upon counter clearing by a period register compare match, the initial value set in TIOR is output from each pin. If the values set in the period and duty registers are identical, the output value does not change even when a compare match occurs.

Up to eight phases of PWM waveforms can be output by combining synchronous clearing of channels that cannot be set to PWM mode 2 as synchronous operation.

The correspondence between PWM output pins and registers is shown in Table 19.49.

Table 19.49 PWM Output Registers and Output Pins

Channel	Register	Output Pins	
		PWM Mode 1	PWM Mode 2
MTU0	TGRA	MTIOC0A	MTIOC0A
	TGRB		MTIOC0B
	TGRC	MTIOC0C	MTIOC0C
	TGRD		MTIOC0D
MTU1	TGRA	MTIOC1A	MTIOC1A
	TGRB		MTIOC1B
MTU2	TGRA	MTIOC2A	MTIOC2A
	TGRB		MTIOC2B
MTU3	TGRA	MTIOC3A	Setting prohibited
	TGRB		
	TGRC	MTIOC3C	
	TGRD		
MTU4	TGRA	MTIOC4A	
	TGRB		
	TGRC	MTIOC4C	
	TGRD		

Note: In PWM mode 2, PWM waveform output is not possible for the TGR register in which the PWM period is set.

(1) Example of PWM Mode Setting Procedure

Figure 19.25 shows an example of the PWM mode setting procedure.

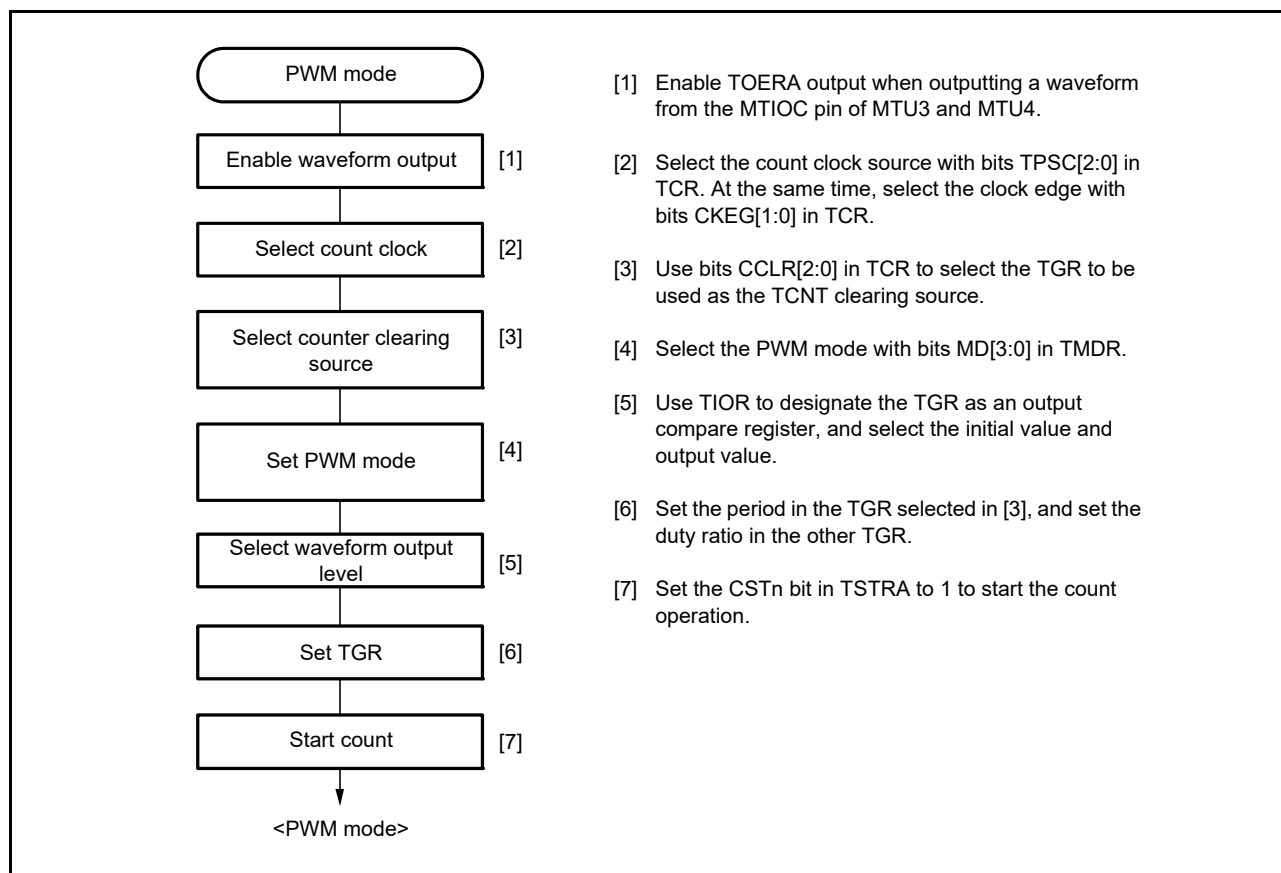


Figure 19.25 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 19.26 shows an example of operation in PWM mode 1.

In this example, TGRA compare match is set as the TCNT clearing source, 0 is set as the initial output value and output value for TGRA, and 1 is set as the output value for TGRB.

In this case, the value set in TGRA is used as the period, and the value set in TGRB is used as the duty ratio.

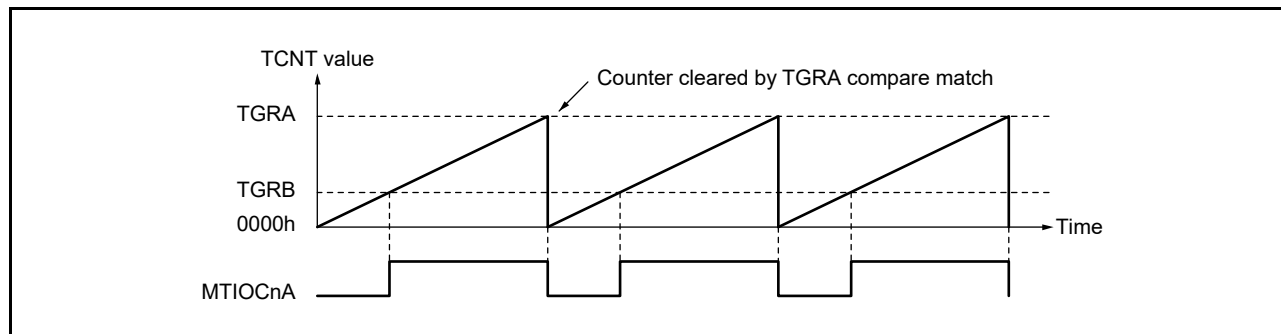


Figure 19.26 Example of PWM Mode 1 Operation (n = 0 to 4)

Figure 19.27 shows an example of operation in PWM mode 2.

In this example, synchronous operation is designated for MTU0 and MTU1, MTU1.TGRB compare match is set as the TCNT clearing source, and low is set as the initial output value and high as the output value for the other TGR registers (MTU0.TGRA to MTU0.TGRD and MTU1.TGRA), outputting 5-phase PWM waveforms.

In this case, the value set in MTU1.TGRB is used as the period, and the values set in the other TGRs are used as the duty ratio.

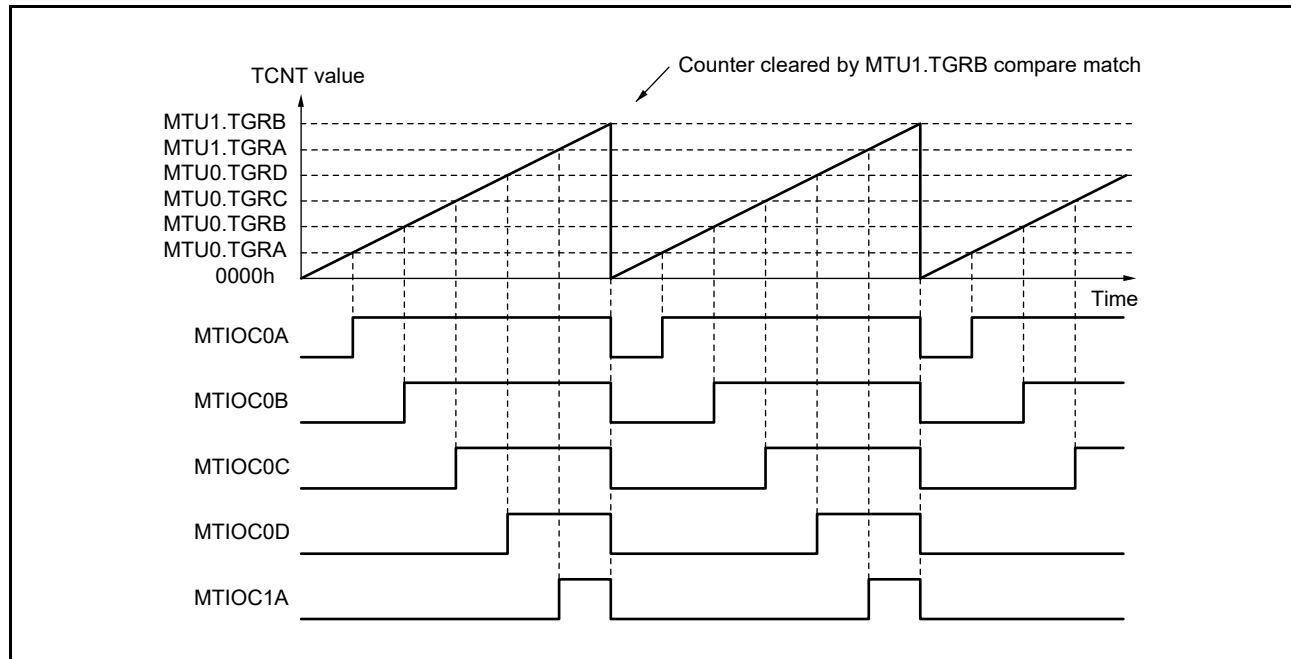


Figure 19.27 Example of PWM Mode 2 Operation

Figure 19.28 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode 1. In this example, TGRA compare match is set as the TCNT clearing source, a low level is set as the initial output value for TGRA, and a high level is set as the output value for TGRB.

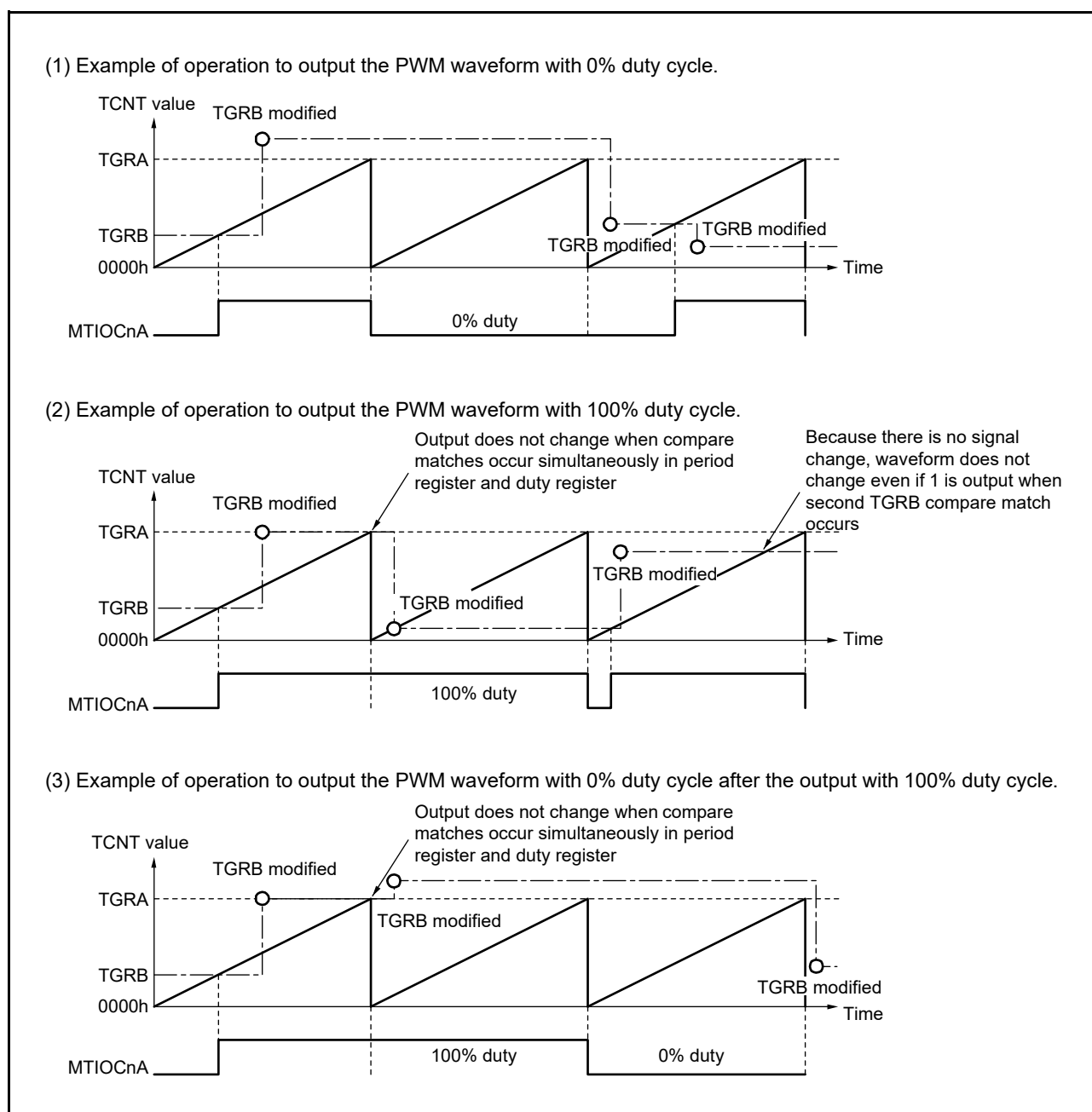


Figure 19.28 Examples of PWM Mode Operation (PWM Waveform Output with 0% Duty and 100% Duty) (n = 0 to 4)

19.3.6 Phase Counting Mode

There are two phase counting modes: 16-bit phase counting mode in which MTU1 and MTU2 operate independently, and cascade connection 32-bit phase counting mode in which MTU1 and MTU2 are cascaded.

In phase counting mode, the phase difference between two external input clocks is detected and the corresponding TCNT is incremented or decremented.

Two external clock input pins for each phase counting mode are not affected by the settings of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. The two external clock input pins used in 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2 can be selected by MTU1.TMDR3.PHCKSEL. In a phase counting mode other than 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2, MTCLKA and MTCLKB are selected for A-phase and B-phase, respectively. In phase counting mode, the external clock pins MTCLKA, MTCLKB, MTCLKC, and MTCLKD are used for two-phase encoder pulse input.

Table 19.50 lists the external clock input pins to be connected in each phase counting mode.

Table 19.50 Clock Input Pins in Phase Counting Mode

Phase Counting Mode	TMDR3.PHCKSEL bit	External Clock Input Pins	
		A-Phase	B-Phase
MTU1 16-bit phase counting mode	x (Don't care)	MTCLKA	MTCLKB
MTU2 16-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD
Cascade connection 32-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD

19.3.6.1 16-Bit Phase Counting Mode

When the MTU1.TMDR3.LWA is 0, 16-bit phase counting mode can be set individually for MTU1 and MTU2.

In 16-bit phase counting mode, the phase difference between two external input clocks is detected and the 16-bit counter TCNT of the corresponding channel is incremented or decremented.

When 16-bit phase counting mode is specified, an external clock is selected as the count clock and TCNT operates as an up-counter/down-counter regardless of the setting of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. However, the functions of TCR.CCLR[1:0], TIOR, TIER, and TGR are enabled, and input capture/compare match and interrupt functions can be used.

These external input pins can be used for two-phase encoder pulse input.

When an overflow occurs while TCNT is counting up and the corresponding TIER.TCIEV bit is 1, a TCIV interrupt is generated. When an underflow occurs while TCNT is counting down and the corresponding TIER.TCIEU bit is 1, a TCIU interrupt is generated.

The TSR.TCFD flag is the count direction flag. Read the TCFD flag to check whether TCNT is counting up and down.

(1) Example of 16-Bit Phase Counting Mode Setting Procedure

Figure 19.29 shows an example of the phase counting mode setting procedure.

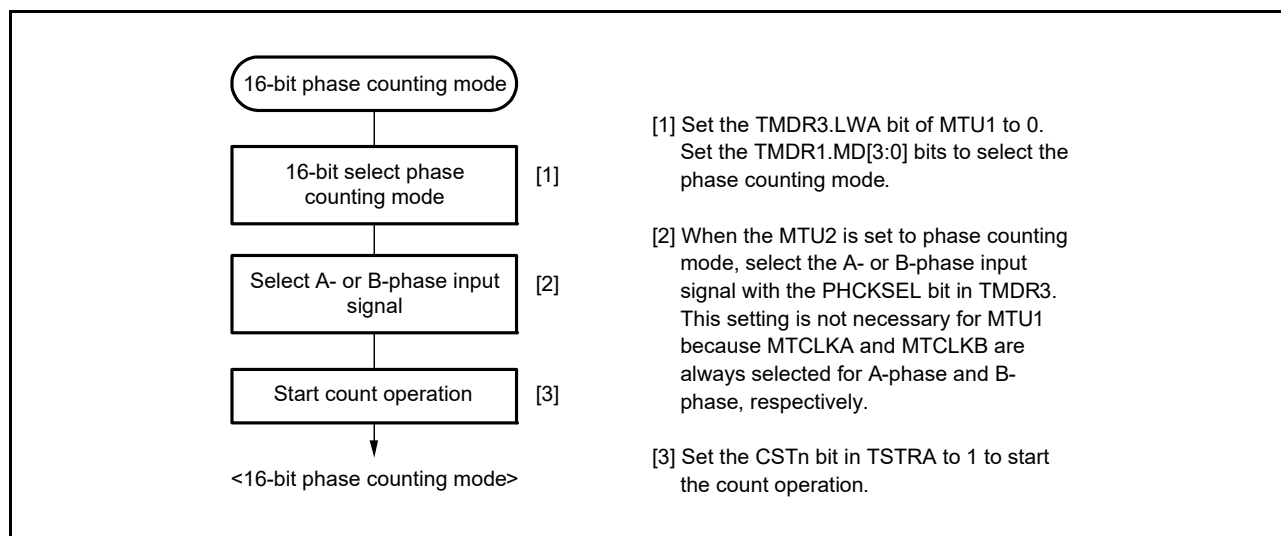


Figure 19.29 Example of 16-Bit Phase Counting Mode Setting Procedure

(2) Examples of 16-Bit Phase Counting Mode Operation

In phase counting mode, TCNT is incremented or decremented according to the phase difference between two external clocks. There are five modes according to the count conditions. Each mode operates under the condition PHCKSEL = 1, which means the phase clock for MTU1 is input from MTCLKA or MTCLKB and that for MTU2 is input from MTCLKC or MTCLKD.

(a) Phase Counting Mode 1

Figure 19.30 shows an example of operation in phase counting mode 1, and Table 19.51 summarizes the TCNT up-counting and down-counting conditions.

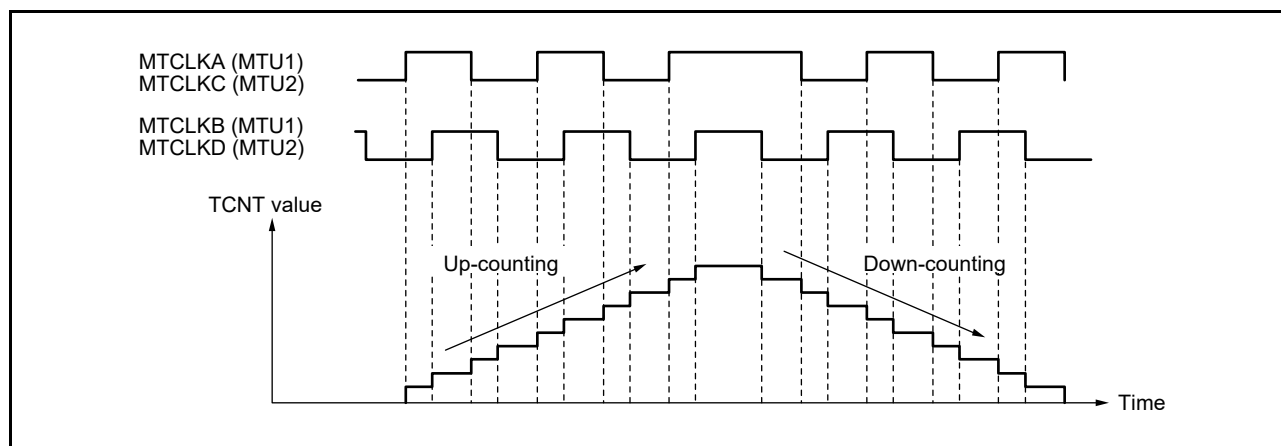


Figure 19.30 Example of Operation in Phase Counting Mode 1

Table 19.51 Up-Counting and Down-Counting Conditions in Phase Counting Mode 1

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
High		Up-counting
Low		
	Low	
	High	
High		Down-counting
Low		
	High	
	Low	

 : Rising edge
 : Falling edge

(b) Phase Counting Mode 2

Figure 19.31 to Figure 19.33 show the examples of operation in phase counting mode 2 and Table 19.52 summarizes the TCNT up-counting and down-counting conditions.

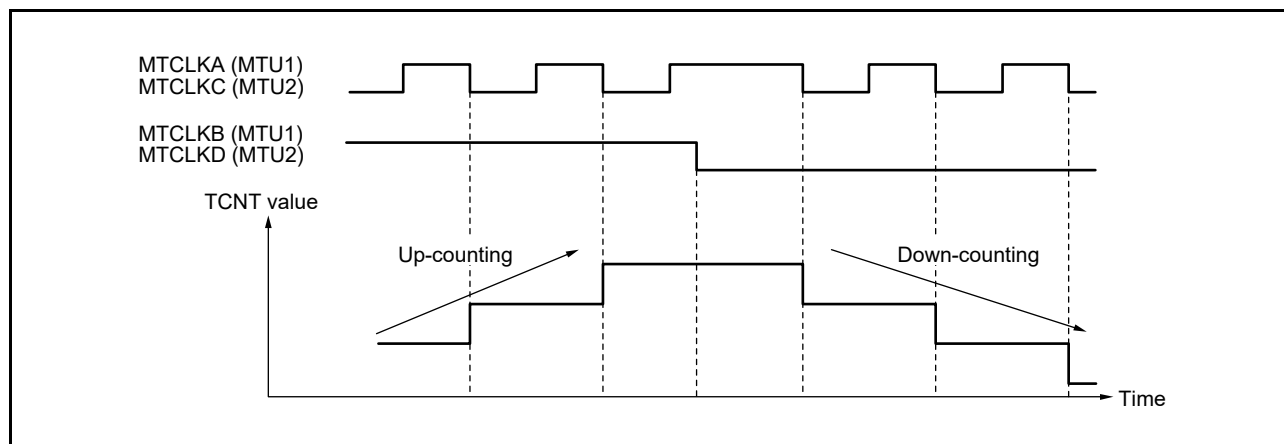


Figure 19.31 Example of Operation in Phase Counting Mode 2 (When $MTUn.TCR2.PCB[1:0] = 00b$ ($n = 1, 2$))

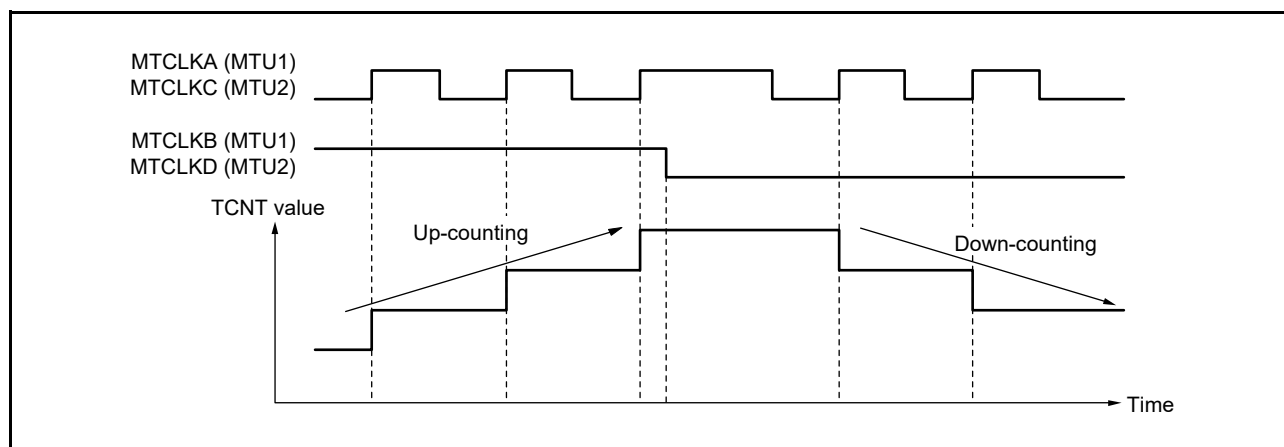


Figure 19.32 Example of Operation in Phase Counting Mode 2 (When $MTUn.TCR2.PCB[1:0] = 01b$ ($n = 1, 2$))

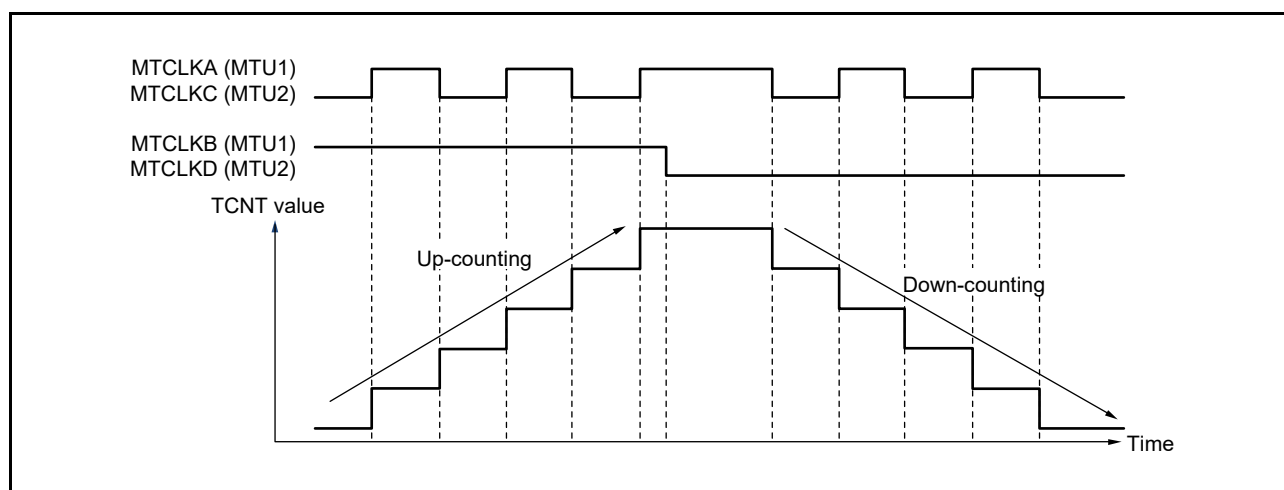


Figure 19.33 Example of Operation in Phase Counting Mode 2 (When $MTUn.TCR2.PCB[1:0] = 1xb$ ($n = 1, 2$))

Table 19.52 Up-Counting and Down-Counting Conditions in Phase Counting Mode 2

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
00b	High		Not counted (Don't care)
	Low		
		Low	Up-counting
		High	
	High		Not counted (Don't care)
	Low		
		High	Down-counting
		Low	
01b	High		Not counted (Don't care)
	Low		
		Low	Down-counting
		High	
	High		Not counted (Don't care)
	Low		
		High	Up-counting
		Low	
1xb	High		Not counted (Don't care)
	Low		
		Low	Down-counting
		High	
	High		Not counted (Don't care)
	Low		
		High	Up-counting
		Low	

 : Rising edge
 : Falling edge

(c) Phase Counting Mode 3

Figure 19.34 to Figure 19.36 show the examples of operation in phase counting mode 3 and Table 19.53 summarizes the TCNT up-counting and down-counting conditions.

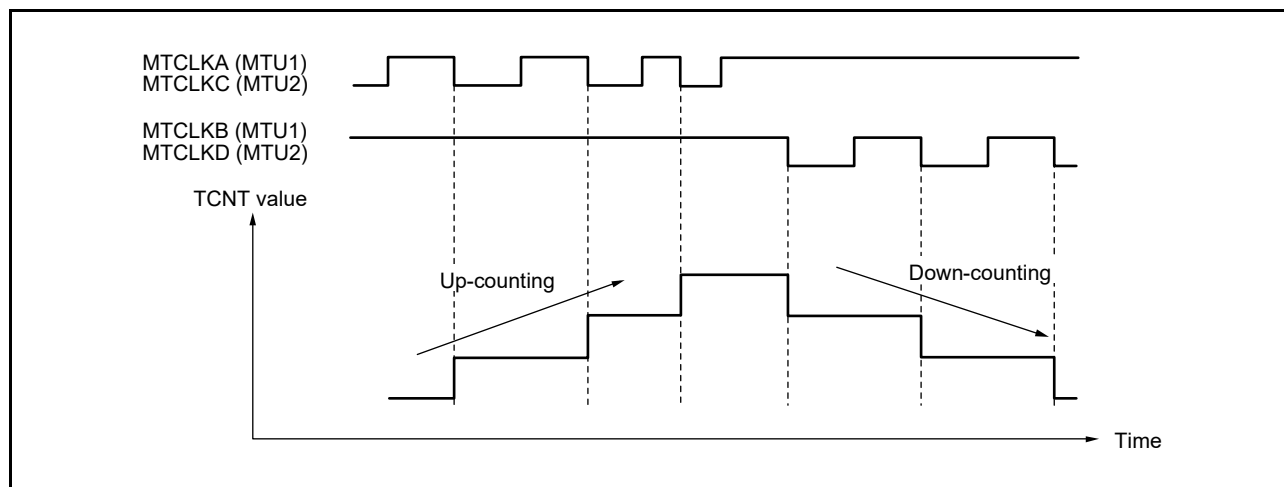


Figure 19.34 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] = 00b (n = 1, 2))

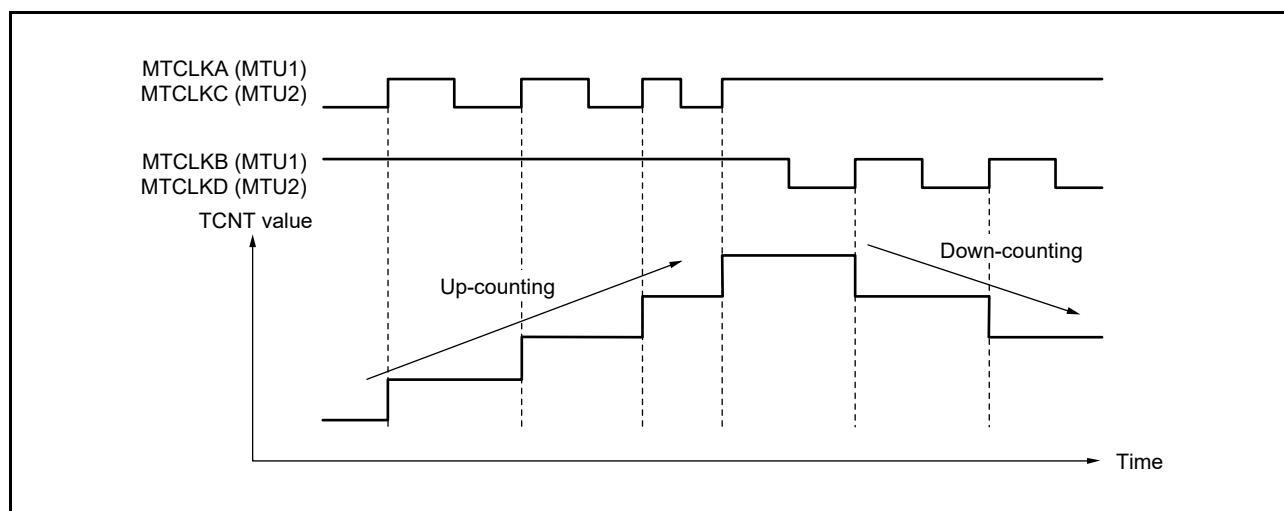


Figure 19.35 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] = 01b (n = 1, 2))

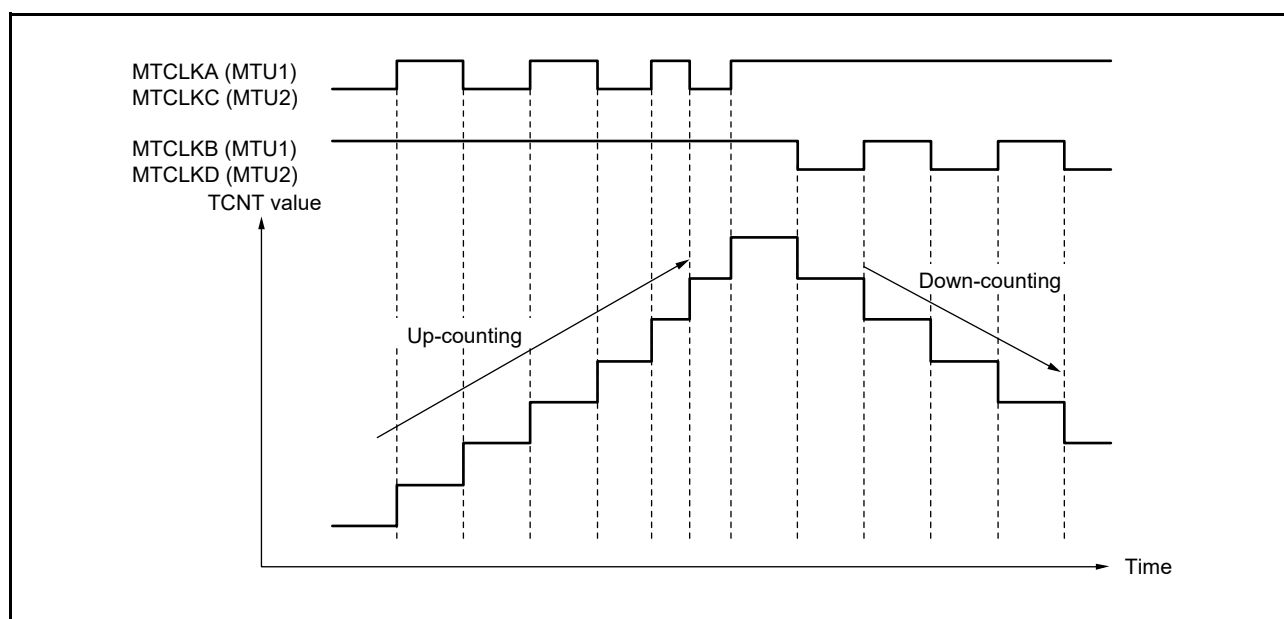


Figure 19.36 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] = 1xb (n = 1, 2))

Table 19.53 Up-Counting and Down-Counting Conditions in Phase Counting Mode 3

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
00b	High		Not counted (Don't care)
	Low		
		Low	Up-counting
		High	
	High		Down-counting
	Low		
		High	Not counted (Don't care)
		Low	
01b	High		Down-counting
	Low		
		Low	Not counted (Don't care)
		High	
	High		Up-counting
	Low		
		High	Not counted (Don't care)
		Low	
1xb	High		Down-counting
	Low		
		Low	Not counted (Don't care)
		High	
	High		Up-counting
	Low		
		High	Not counted (Don't care)
		Low	

: Rising edge
 : Falling edge

(d) Phase Counting Mode 4

Figure 19.37 shows an example of operation in phase counting mode 4, and Table 19.54 summarizes the TCNT up-counting and down-counting conditions.

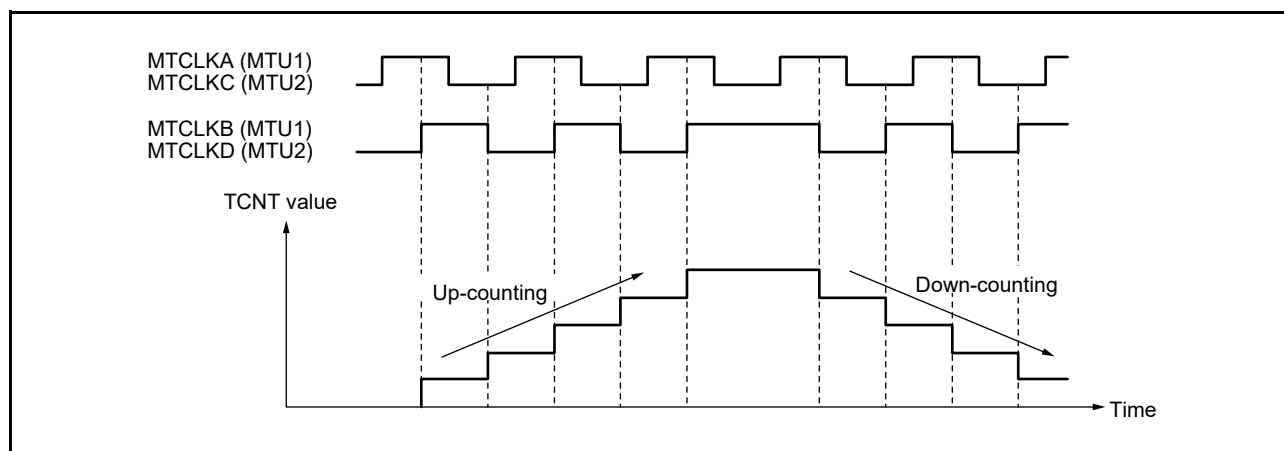


Figure 19.37 Example of Operation in Phase Counting Mode 4

Table 19.54 Up-Counting and Down-Counting Conditions in Phase Counting Mode 4

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
High		Up-counting
Low		Up-counting
	Low	Not counted (Don't care)
	High	Not counted (Don't care)
High		Down-counting
Low		Down-counting
	High	Not counted (Don't care)
	Low	Not counted (Don't care)

 : Rising edge

 : Falling edge

(e) Phase Counting Mode 5

Figure 19.38 and Figure 19.39 show the examples of operation in phase counting mode 5 and Table 19.55 summarizes the TCNT up-counting and down-counting conditions.

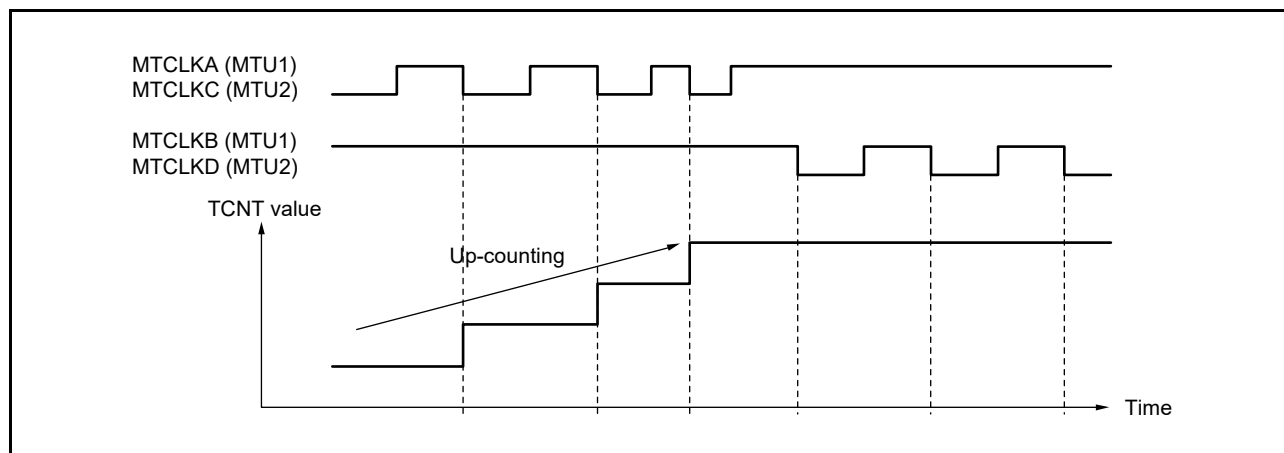


Figure 19.38 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 0xb (n = 1, 2))

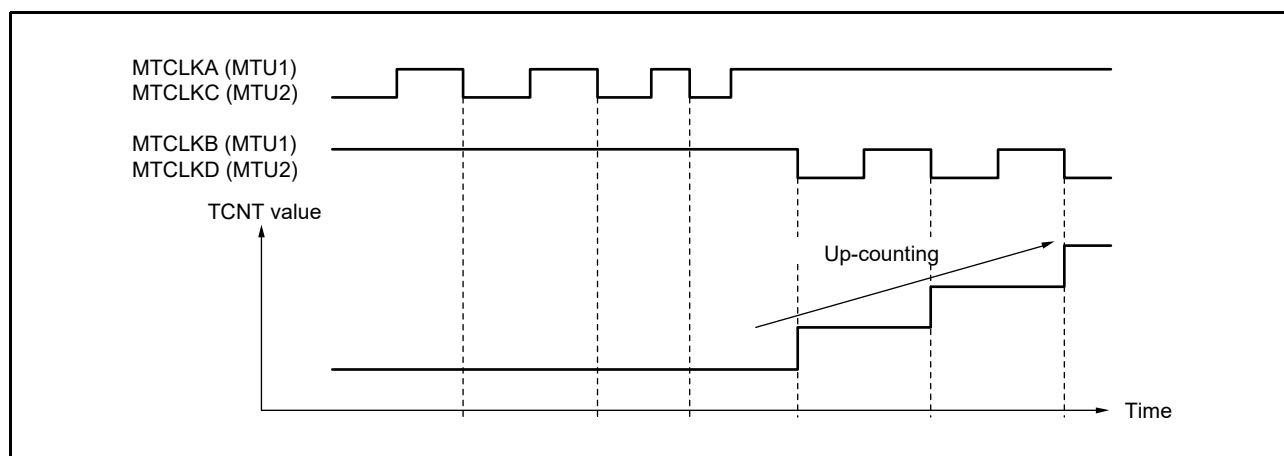


Figure 19.39 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 1xb (n = 1, 2))

Table 19.55 Up-Counting and Down-Counting Conditions in Phase Counting Mode 5

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
0xb	High		Not counted (Don't care)
	Low		
		Low	
		High	Up-counting
	High		Not counted (Don't care)
	Low		
		High	
		Low	Up-counting
1xb	High		Not counted (Don't care)
	Low		Up-counting
		Low	Not counted (Don't care)
		High	
	High		Up-counting
	Low		Not counted (Don't care)
		High	
		Low	

 : Rising edge
 : Falling edge

(3) 16-Bit Phase Counting Mode Application Example

Figure 19.40 shows an example in which MTU1 is in phase counting mode, and MTU1 is coupled with MTU0 to input 2-phase encoder pulses of a servo motor in order to detect position or speed.

MTU1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to MTCLKA and MTCLKB.

In MTU0, MTU0.TGRC compare match is specified as the TCNT clearing source and MTU0.TGRA and MTU0.TGRC are used for the compare match function and are set with the speed control period and position control period.

MTU0.TGRB is used for input capture, with MTU0.TGRB and MTU0.TGRD operating in buffer mode. The MTU1 count clock is designated as the MTU0.TGRB input capture source, and the widths of 2-phase encoder 4-multiplication pulses are detected.

MTU1.TGRA and MTU1.TGRB for MTU1 are designated for the input capture function and MTU0.TGRA and MTU0.TGRC compare matches in MTU0 are selected as the input capture sources to store the up/down-counter values for the control periods.

This procedure enables the accurate detection of position and speed.

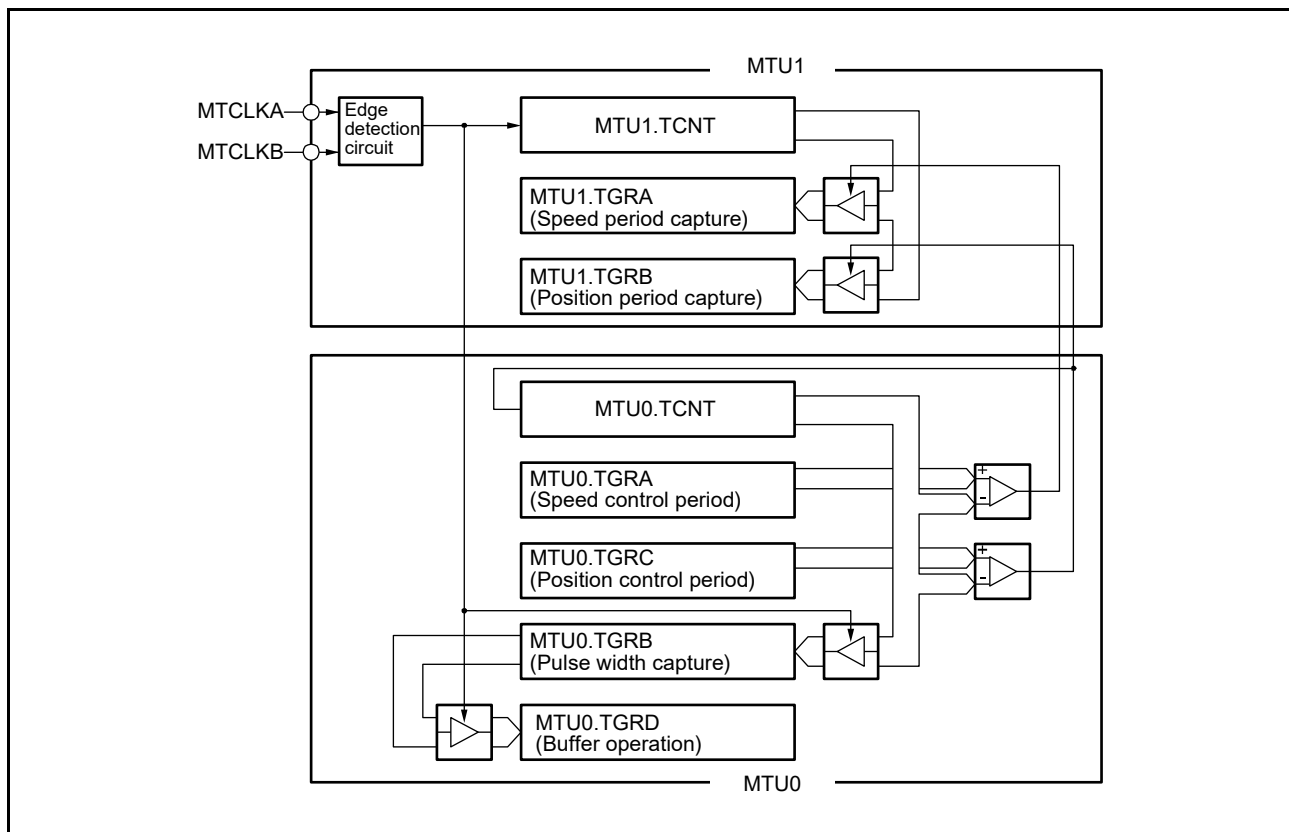


Figure 19.40 16-Bit Phase Counting Mode Application Example

19.3.6.2 Cascade Connection 32-Bit Phase Counting Mode

When MTU1 is set to phase counting mode by setting MTU1.TMDR3.LWA = 1, MTU1 and MTU2 are connected to operate in cascade connection 32-bit phase counting mode. When this mode is used, the TCR, TCR2, TIOR, TIER, TGR, and TSR registers are controlled by MTU1 and the settings of MTU2 are disabled. Refer to Figure 19.41 for the procedure for setting cascade connection 32-bit phase counting mode.

Refer to section 19.3.4, **Cascaded Operation**, for details on the cascade connection function for connecting MTU1 and MTU2 in a mode other than cascade connection 32-bit phase counting mode.

(1) Example of Setting Cascade Connection 32-Bit Phase Counting Mode

Figure 19.41 shows an example of the procedure for setting cascade connection 32-bit phase counting mode.

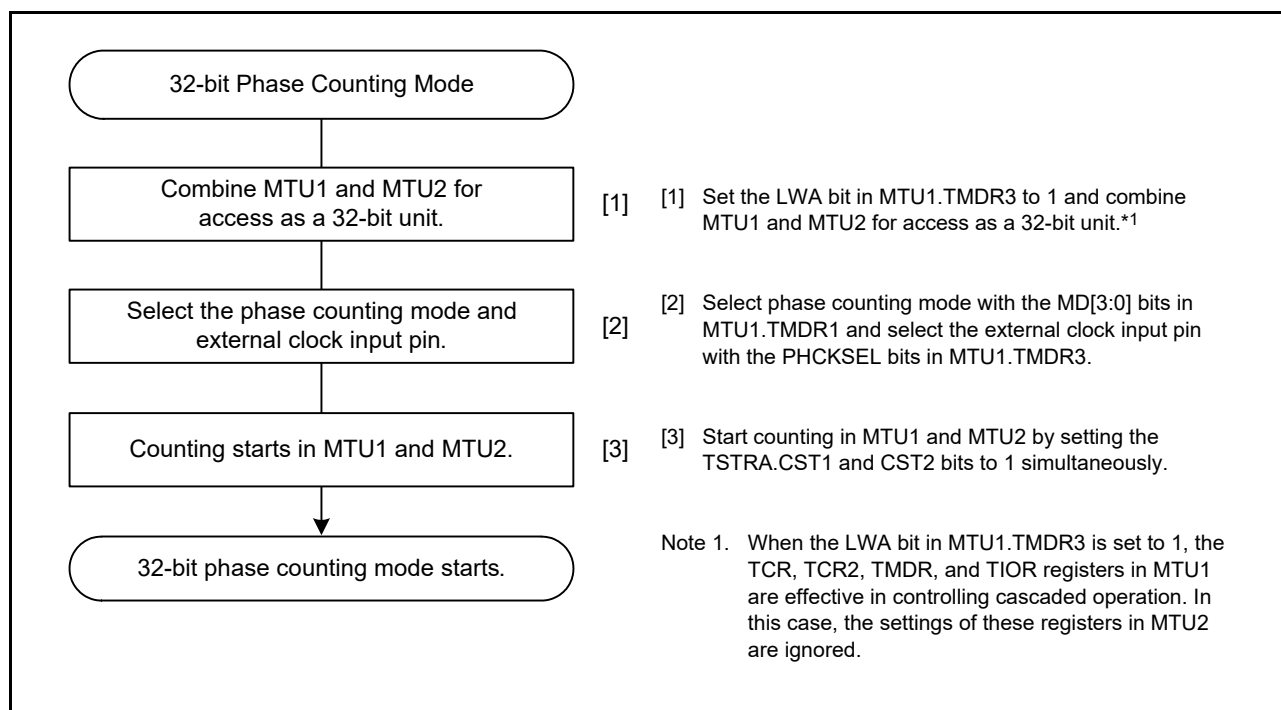


Figure 19.41 Procedure for Setting Cascade Connection 32-Bit Phase Counting Mode

19.3.7 Reset-Synchronized PWM Mode

In the reset-synchronized PWM mode, three phases of positive and negative PWM waveforms (six phases in total) that share a common wave transition point can be output by combining MTU3 and MTU4.

When set for reset-synchronized PWM mode, the MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, and MTIOC4D pins function as PWM output pins and timer counter 3 (MTU3.TCNT) functions as an up-counter.

Table 19.56 shows the PWM output pins used. Table 19.57 shows the settings of the registers.

Table 19.56 Output Pins for Reset-Synchronized PWM Mode

Channel	Output Pin	Description
MTU3	MTIOC3A	Toggle output synchronized with PWM period (or I/O port)
	MTIOC3B	PWM output pin 1
	MTIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
MTU4	MTIOC4A	PWM output pin 2
	MTIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	MTIOC4B	PWM output pin 3
	MTIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)

Table 19.57 Register Settings for Reset-Synchronized PWM Mode

Register	Setting
MTU3.TCNT	Initial setting (0000h)
MTU4.TCNT	Initial setting (0000h)
MTU3.TGRA	Set the count period for MTU3.TCNT
MTU3.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC3B and MTIOC3D pins
MTU4.TGRA	Set the transition point of the PWM waveform to be output from the MTIOC4A and MTIOC4C pins
MTU4.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC4B and MTIOC4D pins

(1) Example of Procedure for Setting Reset-Synchronized PWM Mode

Figure 19.42 shows an example of procedure for setting the reset-synchronized PWM mode.

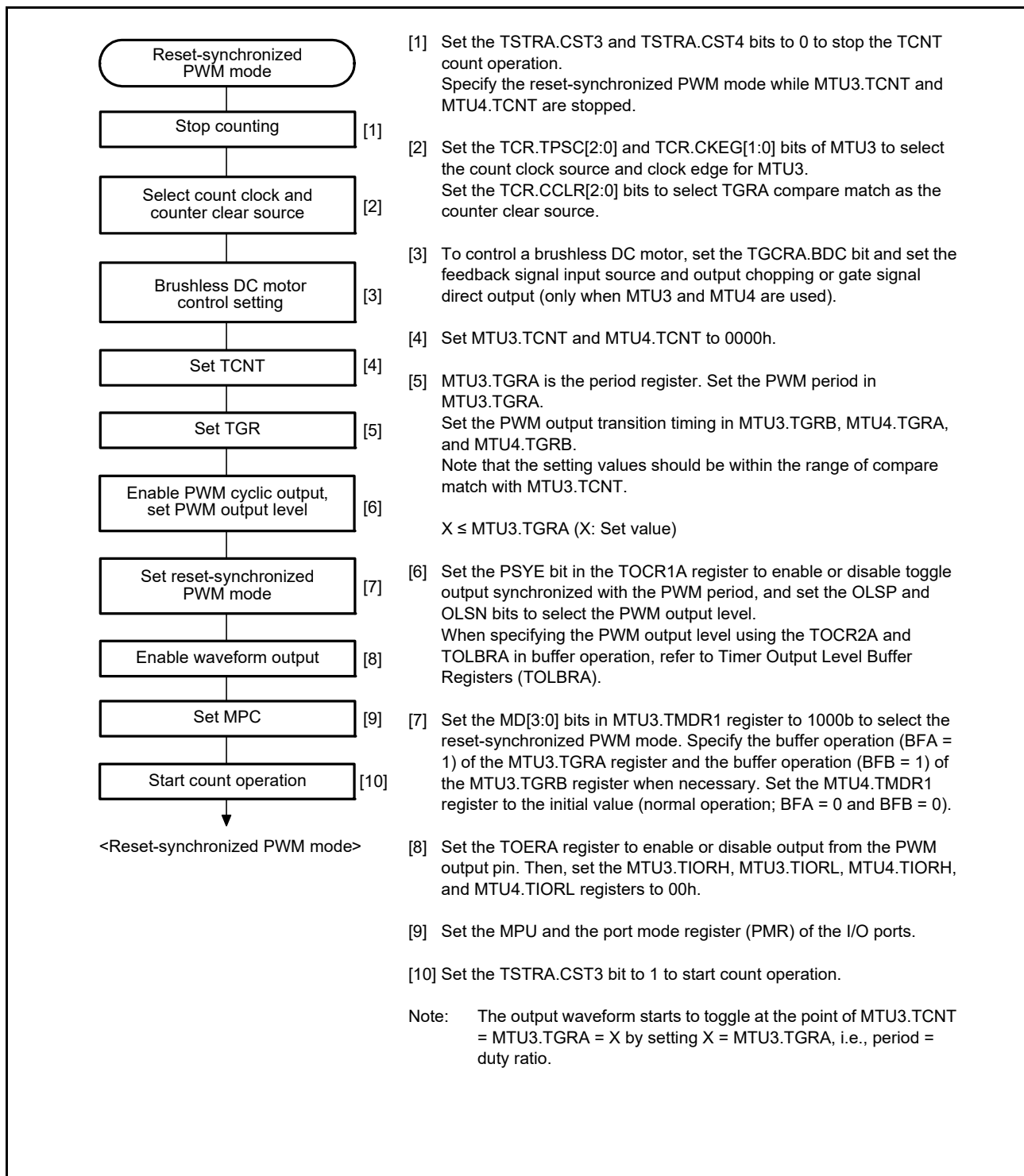


Figure 19.42 Procedure for Selecting Reset-Synchronized PWM Mode

(2) Example of Reset-Synchronized PWM Mode Operation

Figure 19.43 shows an example of operation in the reset-synchronized PWM mode.

MTU3.TCNT and MTU4.TCNT operate as up-counters. The counters are cleared when a compare match occurs between MTU3.TCNT and MTU3.TGRA, and then begin incrementing from 0000h. The output from the PWM pins toggles every time a compare match occurs in MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB and the counters are cleared.

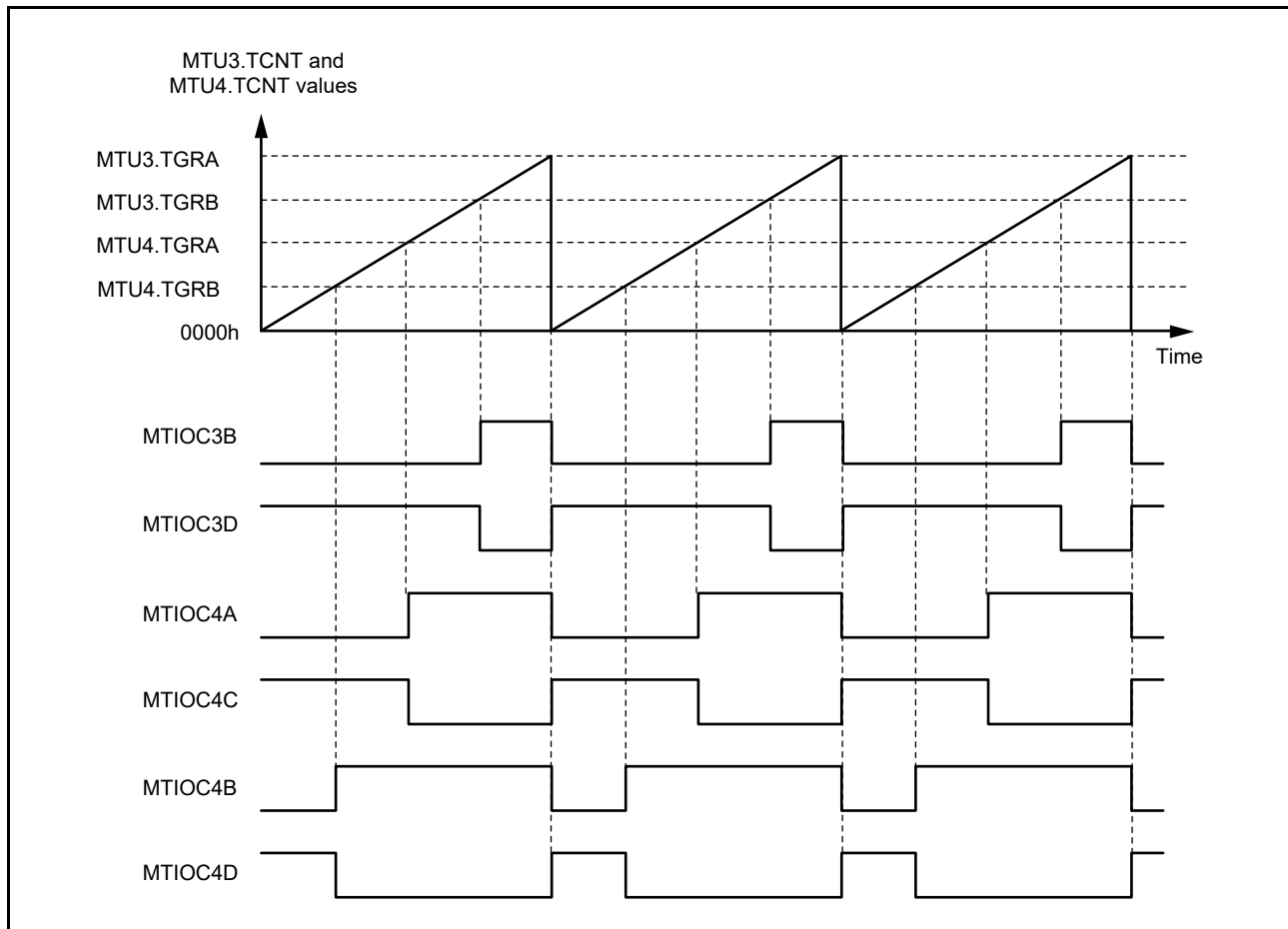


Figure 19.43 Example of Reset-Synchronized PWM Mode Operation
(When OLSN = 1 and OLSP = 1 in MTU3.TOCR1 and MTU4.TOCR1)

19.3.8 Complementary PWM Mode

In complementary PWM mode, dead time can be set for PWM waveforms to be output. The dead time is the period during which the upper and lower arm transistors are set to the inactive level in order to prevent short-circuiting of the arms.

Six positive-phase and three negative-phase PWM waveforms (six phases in total) with dead time can be output by combining MTU3/ MTU4. PWM waveforms without dead time can also be output.

In complementary PWM mode, MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, and MTIOC4D pins function as PWM output pins, and the MTIOC3A pin can be set for toggle output synchronized with the PWM period. MTU3.TCNT and MTU4.TCNT function as up/down-counters.

Table 19.58 shows the PWM output pins used. Table 19.59 shows the settings of the registers used.

A function to directly cut off the PWM output by using an external signal is supported as a port function.

Table 19.58 Output Pins for Complementary PWM Mode

Channel	Output Pin	Description
MTU3	MTIOC3A	Toggle output synchronized with PWM period (or I/O port)
	MTIOC3B	PWM output pin 1
	MTIOC3C	I/O port*1
	MTIOC3D	PWM output pin 1' (negative-phase waveform output of PWM output 1)
MTU4	MTIOC4A	PWM output pin 2
	MTIOC4C	PWM output pin 2' (negative-phase waveform output of PWM output 1)
	MTIOC4B	PWM output pin 3
	MTIOC4D	PWM output pin 3' (negative-phase waveform output of PWM output 1)

Note 1. Avoid setting the MTIOC3C pin as timer I/O pins in complementary PWM mode.

Table 19.59 Register Settings for Complementary PWM Mode (1/2)

Channel	Counter/ Register	Description	Read/Write from CPU
MTU3	TCNT	Starts up-counting from the value set in the dead time register	Maskable by TRWERA setting*1
	TGRA	Set MTU3.TCNT upper limit value (1/2 carrier period + dead time)	Maskable by TRWERA setting*1
	TGRB	PWM output 1 compare register	Maskable by TRWERA setting*1
	TGRC	MTU3.TGRA buffer register	Readable/writable
	TGRD	PWM output 1/MTU3.TGRB buffer register	Readable/writable
	TGRE	MTU3.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU4	TCNT	Starts up-counting after being initialized to 0000h	Maskable by TRWERA setting*1
	TGRA	PWM output 2 compare register	Maskable by TRWERA setting*1
	TGRB	PWM output 3 compare register	Maskable by TRWERA setting*1
	TGRC	PWM output 2/MTU4.TGRA buffer register	Readable/writable
	TGRD	PWM output 3/MTU4.TGRB buffer register	Readable/writable
	TGRE	MTU4.TGRA buffer register B (when double buffer function is used)	Readable/writable
	TGRF	MTU4.TGRB buffer register B (when double buffer function is used)	Readable/writable

Note 1. Access can be enabled or disabled according to the setting in TRWERA (timer read/write enable register A).

Table 19.60 Register Settings for Complementary PWM Mode (2/2)

Channel	Counter/ Register	Description	Read/Write from CPU
Timer dead time data register A (TDDRA)		Set MTU4.TCNT and MTU3.TCNT offset value (dead time value)	Maskable by TRWERA setting*1
Timer period data register A (TCDRA)		Set MTU4.TCNT upper limit value (1/2 carrier period)	Maskable by TRWERA setting*1
Timer period buffer register A (TCBRA)		TCDRA buffer register	Readable/writable
Subcounter A (TCNTSA)		Subcounter A for dead time generation	Read-only
Temporary register 1A (TEMP1A)		PWM output 1/MTU3.TGRB temporary register A	Not readable/writable
Temporary register 1B (TEMP1B)		PWM output 1/MTU3.TGRB temporary register B (when double buffer function is used)	Not readable/writable
Temporary register 2A (TEMP2A)		PWM output 2/MTU4.TGRA temporary register A	Not readable/writable
Temporary register 2B (TEMP2B)		PWM output 2/MTU4.TGRA temporary register B (when double buffer function is used)	Not readable/writable

Note 1. Access can be enabled or disabled according to the setting in TRWERA (timer read/write enable register A).

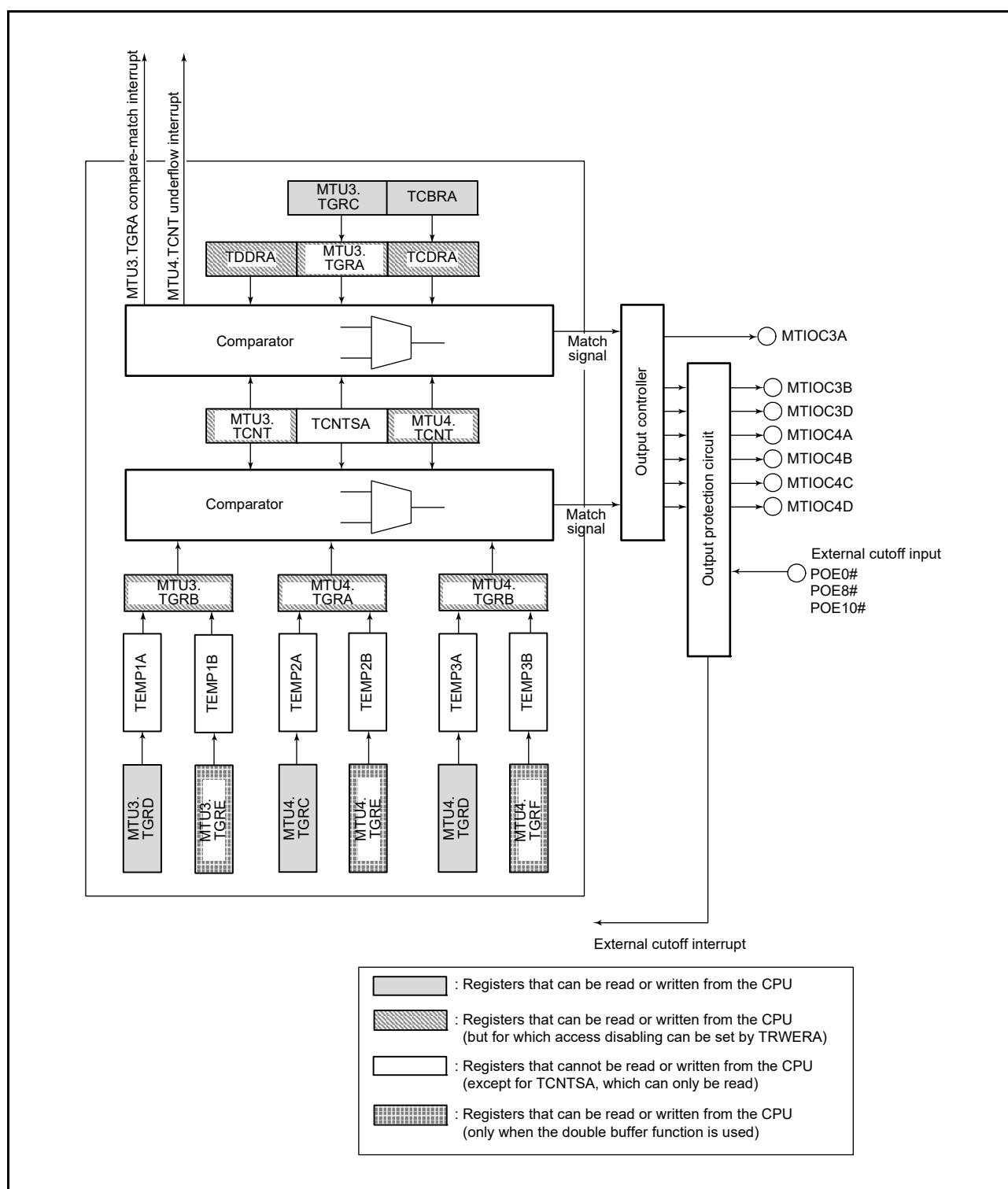


Figure 19.44 Block Diagram of MTU3 and MTU4 in Complementary PWM Mode

(1) Example of Complementary PWM Mode Setting Procedure

Figure 19.45 shows an example of the complementary PWM mode setting procedure.

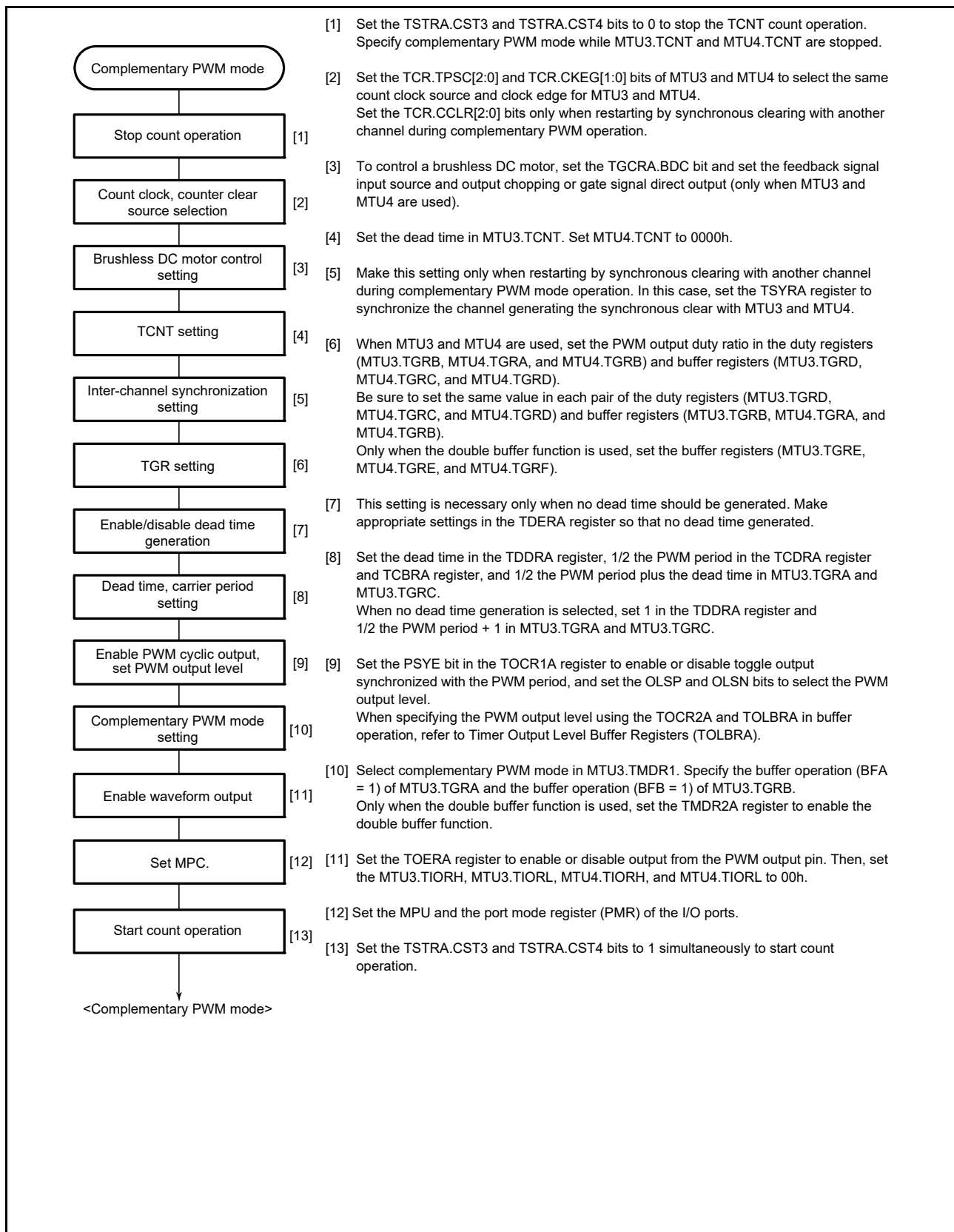


Figure 19.45 Example of Complementary PWM Mode Setting Procedure

(2) Outline of Complementary PWM Mode Operation

In complementary PWM mode, six phases (three positive and three negative) PWM waveforms can be output. Figure 19.46 illustrates counter operation in complementary PWM mode (MTU3 and MTU4), and Figure 19.47 shows an example of operation in complementary PWM mode.

(a) Counter Operation

In complementary PWM mode, three counters—MTU3.TCNT, MTU4.TCNT, and TCNTSA—in each unit perform up-/down-count operations.

MTU3.TCNT is automatically initialized to the value set in TDDRA when complementary PWM mode is selected and the CST3 bit in TSTRA is 0. When the CST3 bit is set to 1, MTU3.TCNT counts up to the value set in MTU3.TGRA, then switches to down-counting when it matches MTU3.TGRA. When the MTU4.TCNT value matches 0000h, MTU3.TCNT switches to up-counting, and the operation is repeated in this way.

MTU4.TCNT should be initialized to 0000h after a reset. When the CST4 bit is set to 1, MTU4.TCNT counts up in synchronization with MTU3.TCNT, and switches to down-counting when MTU3.TCNT matches MTU3.TGRA. On reaching 0000h, MTU4.TCNT switches to up-counting, and the operation is repeated in this way. TCNTSA is a read-only counter. It does not need to be initialized after a reset.

In counting up by MTU3.TCNT and MTU4.TCNT, MTU3.TCNT starts counting up when it matches TCDRA and switches to counting down when it matches MTU3.TGRA. Furthermore, when MTU4.TCNT matches TDDRA, TCNTSA is set to the value in MTU3.TGRA and counting is stopped.

When MTU4.TCNT matches TDDRA during down-counting of MTU3.TCNT and MTU4.TCNT, TCNTSA starts up-counting, and when MTU4.TCNT matches 0000h, the operation switches to down-counting. When MTU3.TCNT matches TCDRA, TCNTSA becomes 0000h and stops counting.

TCNTSA is compared with the compare register and temporary register, in which the PWM duty is specified, only during the count operation.

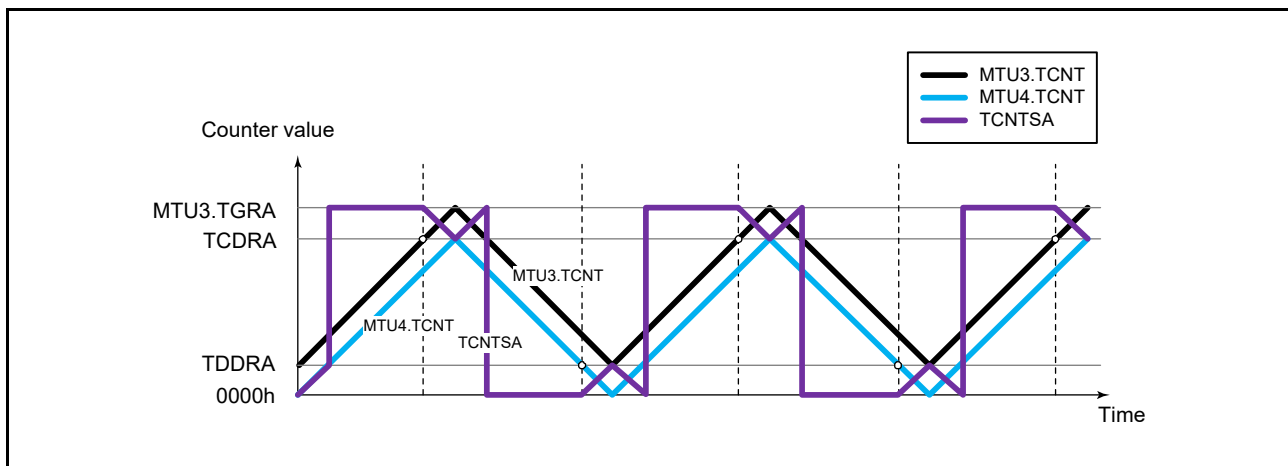


Figure 19.46 Count Operation in Complementary PWM Mode (MTU3 and MTU4)

(b) Register Operation

In complementary PWM mode, nine registers (compare registers, buffer registers, and temporary registers) are used to control the duty ratio for the PWM output. Figure 19.47 shows an example of operation in complementary PWM mode (MTU3 and MTU4).

MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB are constantly compared with the counters to generate PWM waveforms. When these registers match the counter, the value set in the OLSN and OLSP bits in the timer output control register (TOCR1A) is output from the PWM output pin.

MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD are buffer registers for these compare registers.

When the double buffer function is used, MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF are also used as buffer registers B. For details of double buffer operation, refer to section 19.3.8 (2) (r), Double Buffer Function in Complementary PWM Mode.

Data in a compare register can be changed by writing new data to the corresponding buffer register. The buffer registers can be read or written at any time.

When modifying data in a buffer register, write to MTU4.TGRD last and enable data transfer from the buffer register to a temporary register. At this time, transfer from the TCBRA register and MTU3.TGRC register, which operate as buffer registers for the timer period registers, to temporary registers is also enabled. Data is transferred to all five temporary registers at the same time.

When transfer is enabled in the Ta interval, data written to a buffer register is transferred to the temporary register. The data is not transferred to the temporary register in the Tb1 and Tb2 intervals. Data enabled for transfer in this interval is transferred to the temporary register at the end of this interval.

The value transferred to a temporary register is transferred to the compare register at the end of the Tb1 interval (when matches MTU3.TGRA while TCNTSA is counting up), or at the end of the Tb2 interval (when matches 0000h while TCNTSA is counting down). The timing for transfer from the temporary register to the compare register can be selected with bits MD[3:0] in the timer mode register 1 (TMDR1). Figure 19.47 shows an example in which the trough is selected for the transfer timing.

In the Tb interval in which data is not transferred to the temporary register (Tb1 in Figure 19.47), the temporary register has the same function as the compare register and is compared with the counter. In this interval, therefore, there are two compare match registers for one output phase; the compare register contains the pre-change data and the temporary register contains new data. In this interval, three counters MTU3.TCNT, MTU4.TCNT, and TCNTSA and two registers (compare register and temporary register) are compared, and PWM output is controlled accordingly.

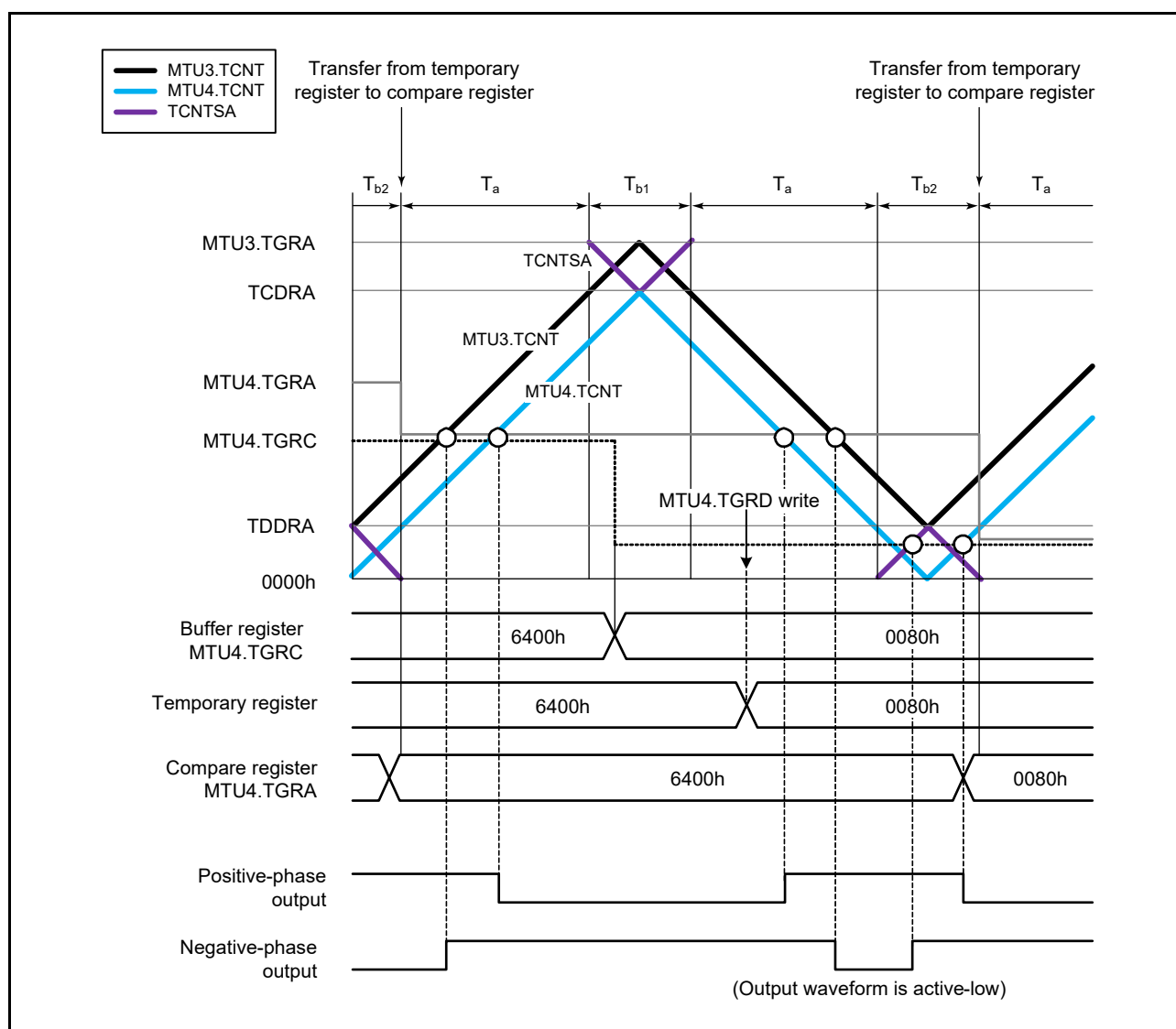


Figure 19.47 Example of Operation in Complementary PWM Mode (MTU3 and MTU4)

(c) Initial Setting

In complementary PWM mode, there are nine registers that require initial setting. In addition, there is a register that specifies whether to generate dead time (it should be used only when dead time generation should be disabled). Before setting complementary PWM mode with MTU3.TMDR1.MD[3:0] bits, initial values should be set in the following registers.

The TOCR1A and TOCR2A registers are used to set the PWM output level. MTU3.TGRC operates as the buffer register for MTU3.TGRA, and should be set with 1/2 the PWM period + dead time Td. The timer period buffer register (TCBRA) operates as the buffer register for the timer period data register (TCDRA), and should be set with 1/2 the PWM period. Set dead time Td in the timer dead time data register (TDDRA).

When dead time is not needed, the TDER bit in the timer dead time enable register (TDERA) should be set to 0, MTU3.TGRC and MTU3.TGRA should be set to 1/2 the PWM carrier period + 1, and TDDRA should be set to 1. Set the respective initial PWM duty values in three buffer registers A (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD). Set three buffer registers B (MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF) only when the double buffer function is used.

The values set in the five buffer registers excluding TDDRA are transferred to the corresponding compare registers as soon as complementary PWM mode is set.

Set MTU4.TCNT to 0000h before setting complementary PWM mode.

Table 19.61 Registers and Counters Requiring Initial Setting

Register and Counter	Setting
TOCR1A, TOCR2A	PWM output level
MTU3.TGRC	1/2 PWM period + dead time Td (1/2 PWM period + 1 when dead time generation is disabled by TDERA)
TDDRA	Dead time Td (1 when dead time generation is disabled by TDERA)
TCBRA	1/2 PWM period
MTU3.TGRD, MTU4.TGRC, MTU4.TGRD	Initial PWM duty ratio value for each phase
MTU3.TGRE, MTU4.TGRE, MTU4.TGRF	Initial PWM duty ratio value for each phase (only when double buffer function is used)
MTU4.TCNT	0000h

Note: The value set in MTU3.TGRC should be the sum of 1/2 the PWM period set in TCBRA and dead time Td set in TDDRA. When dead time generation is disabled by TDERA, TGRC should be set to 1/2 the PWM period + 1.

(d) PWM Output Level Setting

In complementary PWM mode, the PWM output level is set with bits OLSN and OLSP in timer output control register 1 (TOCR1A) or bits OLS1P to OLS3P and OLS1N to OLS3N in timer output control register 2 (TOCR2A).

The output level can be set for each of the three positive phases and three negative phases of 6-phase output.

Complementary PWM mode should be cleared before setting or changing output levels.

(e) Dead Time Setting

In complementary PWM mode, dead time can be set for PWM output.

The dead time is set in the timer dead time data register (TDDRA). The value set in TDDRA is used as the MTU3.TCNT counter start value and creates a non-overlapping interval between MTU3.TCNT and MTU4.TCNT. Complementary PWM mode should be cleared before changing the contents of TDDRA.

(f) Dead Time Suppressing

Dead time generation is suppressed by setting the TDER bit in the timer dead time enable register (TDERA) to 0. TDERA can be set to 0 only when 0 is written to it after reading TDER = 1.

MTU3.TGRA and MTU3.TGRC should be set to $1/2$ PWM period + 1 and the timer dead time data register (TDDRA) should be set to 1.

By the above settings, PWM waveforms without dead time can be obtained. Figure 19.48 shows an example of operation without dead time (MTU3 and MTU4).

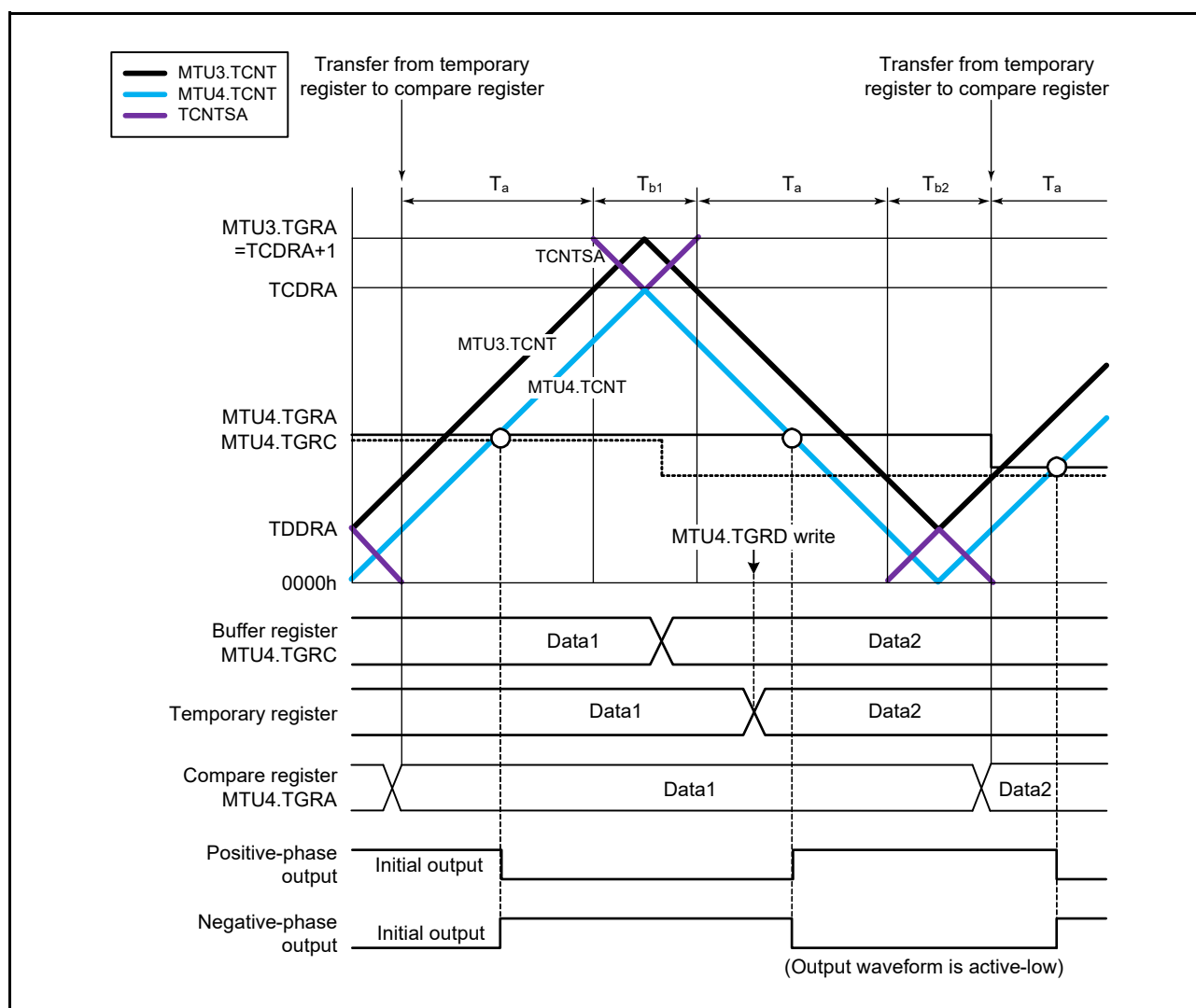


Figure 19.48 Example of Operation without Dead Time (MTU3 and MTU4)

(g) PWM Period Setting

In complementary PWM mode, the PWM period is set in two registers—MTU3.TGRA, in which the MTU3.TCNT upper limit value is set, and TCDRA, in which the MTU4.TCNT upper limit value is set. The settings should be made so as to achieve the following relationship between these two registers:

With dead time: MTU3.TGRA setting = TCDRA setting + TDDRA setting

Without dead time: MTU3.TGRA setting = TCDRA setting + 1

In addition, the settings should be made so as to achieve the following relationship between the TCDRA register and the TDDRA register:

$TCDRA \text{ setting} > TDDRA \text{ setting} \times 2 + 2$

The MTU3.TGRA and TCDRA settings are made by setting values in buffer registers MTU3.TGRC and TCBRA. When data is written to MTU4.TGRD to enable transfers, the values set in MTU3.TGRC and TCBRA are transferred simultaneously to the MTU3.TGRA and TCDRA with the transfer timing selected with the MTU3.TMDR1.MD[3:0] bits.

The new PWM period is reflected from the next cycle when data is updated at the crest, or from the current cycle when updated in the trough. Figure 19.49 illustrates the operation when the PWM period is updated at the crest.

Refer to the following section, (h), Register Data Updating, for the method of updating the data in each buffer register.

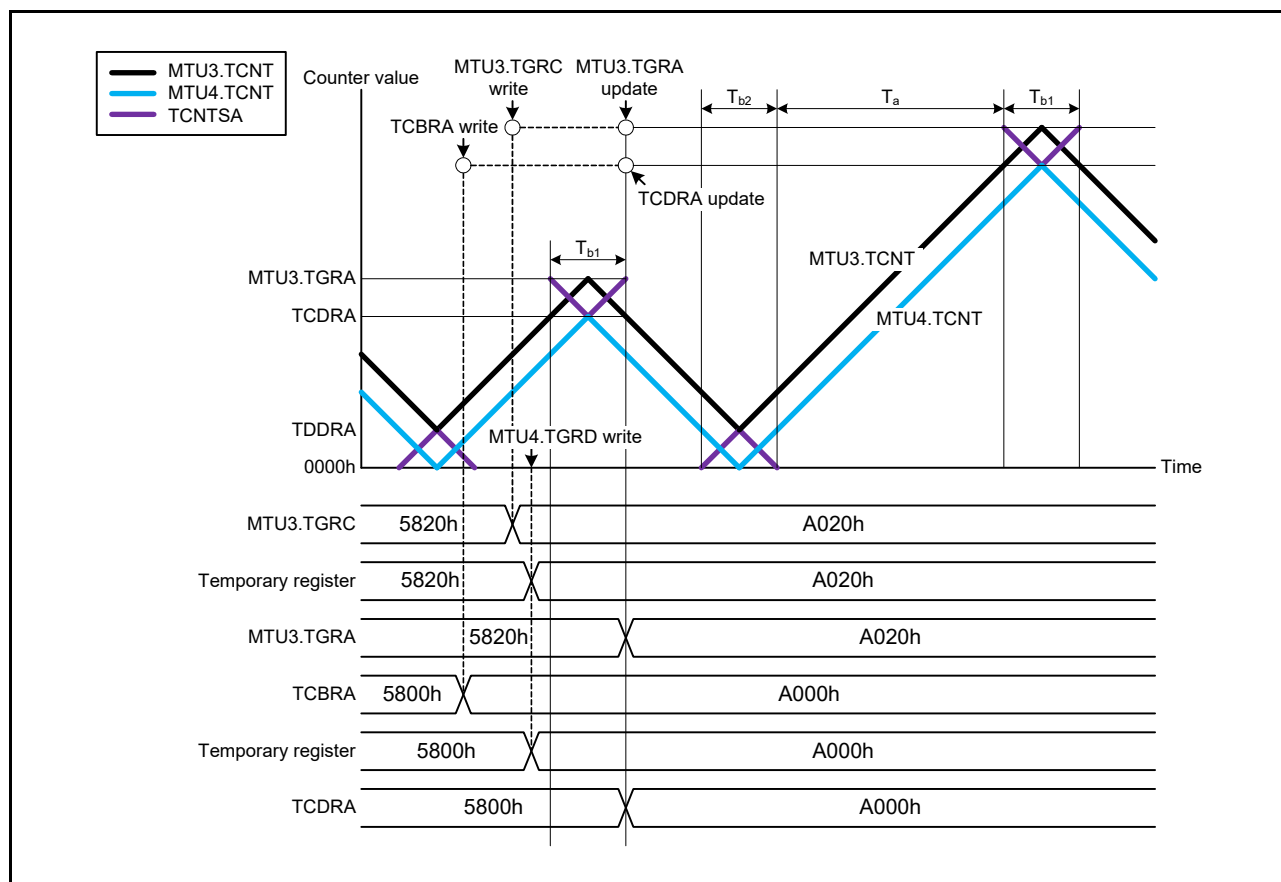


Figure 19.49 Example of PWM Period Updating (MTU3 and MTU4)

(h) Register Data Updating

The buffer registers are used to update the data in five compare registers for the PWM duty and PWM period in complementary PWM mode. The update data can be written to the buffer register at any time.

There is a temporary register between each of these registers and its buffer register. While subcounter TCNTSA is not counting, if buffer register data is updated, the temporary register value also changes. Data is not transferred from buffer registers to temporary registers while TCNTSA is counting; in this case, the value written to a buffer register is transferred after TCNTSA halts.

The temporary register value is transferred to the compare register at the data update timing set with MTU3.TMDR1.MD[3:0] bits. **Figure 19.50** shows an example of data updating in complementary PWM mode (MTU3 and MTU4). This example shows the mode in which data is updated at both the counter crest and trough.

When updating buffer register data, be sure to write to MTU4.TGRD at the end of the update. Data is transferred from buffer registers to the temporary registers simultaneously for all five registers after the write to MTU4.TGRD.

Even when not updating all five registers or when not updating the MTU4.TGRD data, be sure to write to MTU4.TGRD after writing data to the registers to be updated. In this case, the data written to MTU4.TGRD should be the same as the data prior to the write operation.

Refer to section 19.3.8 (2) (r), Double Buffer Function in Complementary PWM Mode, for data updating when the double buffer function is used.

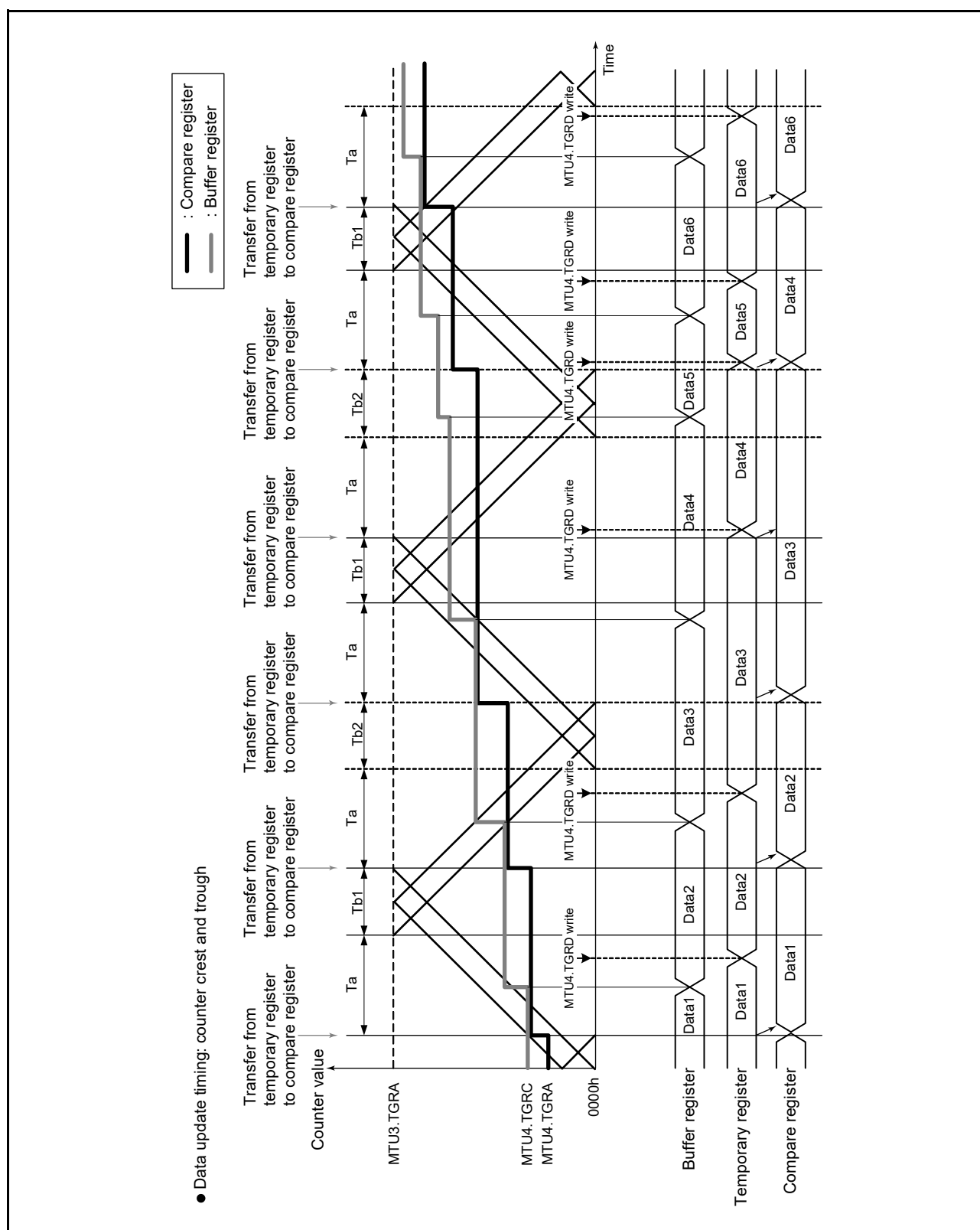


Figure 19.50 Example of Data Updating in Complementary PWM Mode (MTU3 and MTU4)

(i) Initial Output in Complementary PWM Mode

In complementary PWM mode, the initial output is determined by the setting of the OLSN and OLSP bits in the TOCR1A register or the OLS1N to OLS3N and OLS1P to OLS3P bits in the TOCR2A register.

This initial output is the non-active level of the PWM output and continues from when complementary PWM mode is set with the MTU3.TMDR1 until MTU4.TCNT exceeds the value set in the TDDRA register. Figure 19.51 shows an example of the initial output in complementary PWM mode.

An example of the waveform when the initial PWM duty ratio value is smaller than the TDDRA value is shown in Figure 19.52.

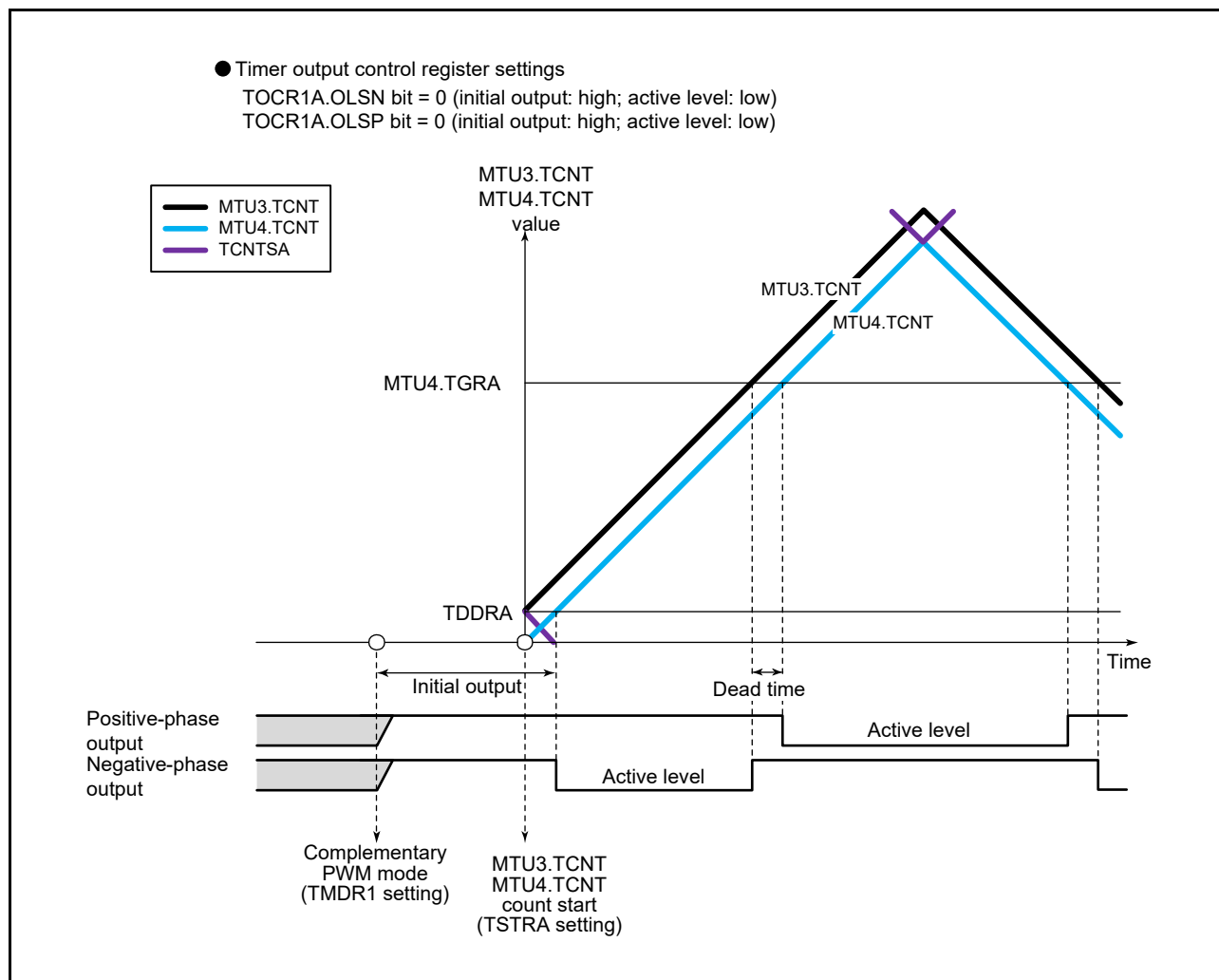


Figure 19.51 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (1))

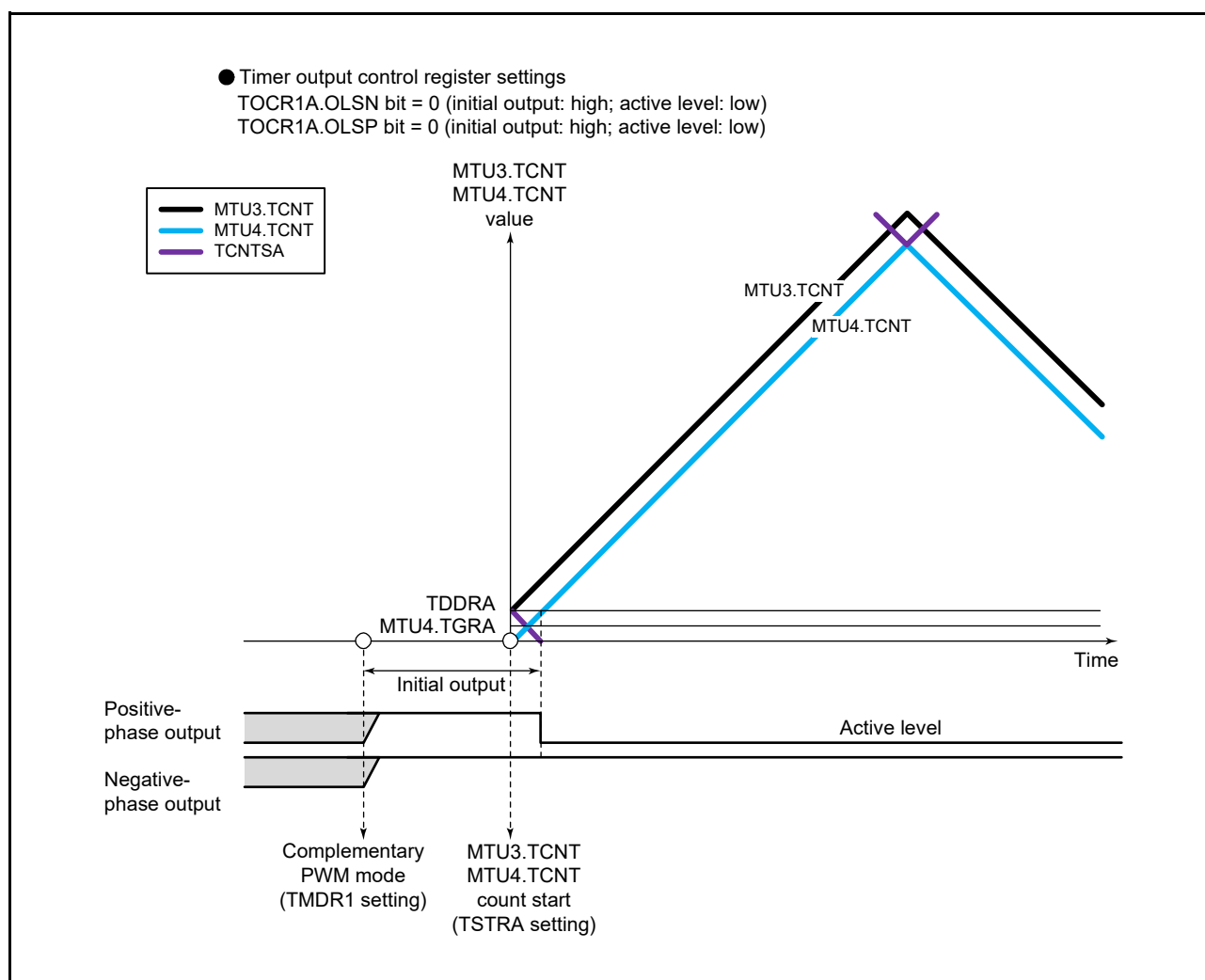


Figure 19.52 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (2)

(j) Method for Generating PWM Output in Complementary PWM Mode

In complementary PWM mode, six phases (three positive and three negative) PWM waveforms can be output. Dead time can be set for PWM waveforms to be output.

A PWM waveform is generated by output of the level selected in the timer output control register in the event of a compare match between a counter and a compare register. While TCNTSA is counting, the compare register and temporary register values are simultaneously compared to create consecutive PWM output from 0 to 100% duty ratio. The relative timing of turn-on and turn-off compare match occurrence may vary, but the compare match that turns off each phase takes precedence to secure the dead time and ensure that the positive-phase and negative-phase turn-on times do not overlap. Figure 19.53 to Figure 19.55 show examples of waveform generation in complementary PWM mode. The positive-phase and negative-phase turn-off timing is generated by a compare match with the counter indicated by a solid line, and the turn-on timing is generated by a compare match with the counter indicated by a dotted line, which operates with a delay equal to the dead time behind the counter indicated by a solid line. In the T1 period, compare match a that turns off the negative phase has the highest priority, and compare matches before a are ignored. In the T2 period, compare match c that turns off the positive phase has the highest priority, and compare matches before c are ignored. In most cases, compare matches occur in the order $a \rightarrow b \rightarrow c \rightarrow d$ (or $c \rightarrow d \rightarrow a' \rightarrow b'$) as shown in Figure 19.53. If compare matches deviate from the $a \rightarrow b \rightarrow c \rightarrow d$ order, since the time for which the negative phase is off is shorter than twice the dead time, the positive phase is not turned on. If compare matches deviate from the $c \rightarrow d \rightarrow a' \rightarrow b'$ order, since the time for which the positive phase is off is shorter than twice the dead time, the negative phase is not turned on. As shown in Figure 19.54, if compare match c follows compare match a before compare match b, compare match b is ignored and the negative phase is turned on by compare match d. This is because turning off the positive phase has priority due to the occurrence of compare match c (positive-phase off timing) before compare match b (positive-phase on timing) (consequently, the waveform does not change because the positive phase goes from off to off). Similarly, in the example in Figure 19.55, turning off the negative phase has priority due to the occurrence of compare match a' (negative-phase off timing) before compare match d (negative-phase on timing). As a result, the negative phase is not turned on.

Thus, in complementary PWM mode, compare matches at turn-off timings take precedence, and turn-on timing compare matches that occur before a turn-off timing compare match are ignored.

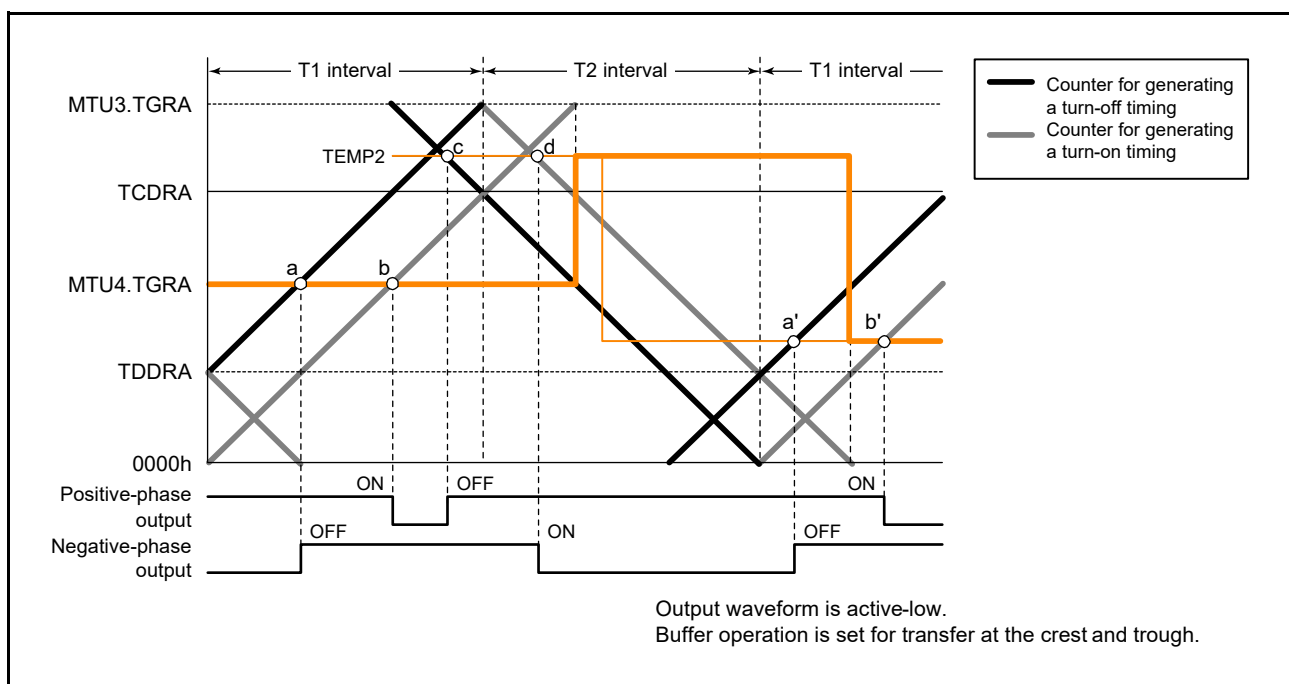


Figure 19.53 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1)

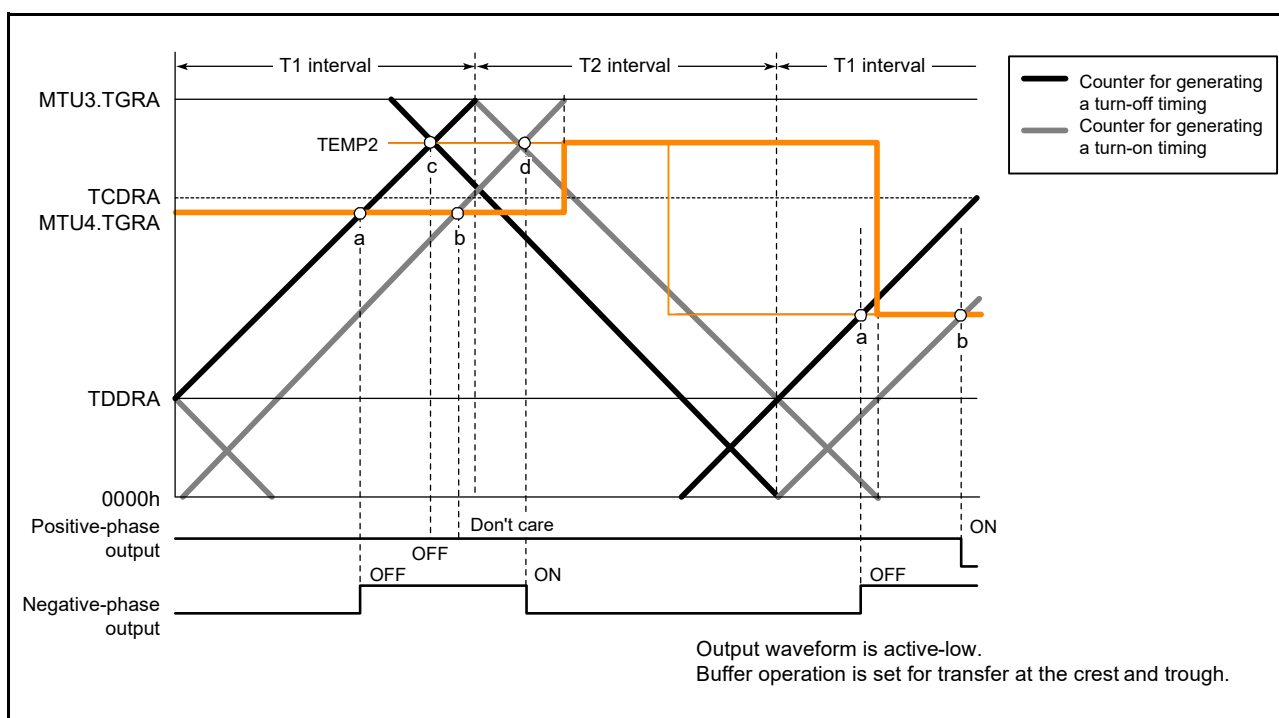


Figure 19.54 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2)

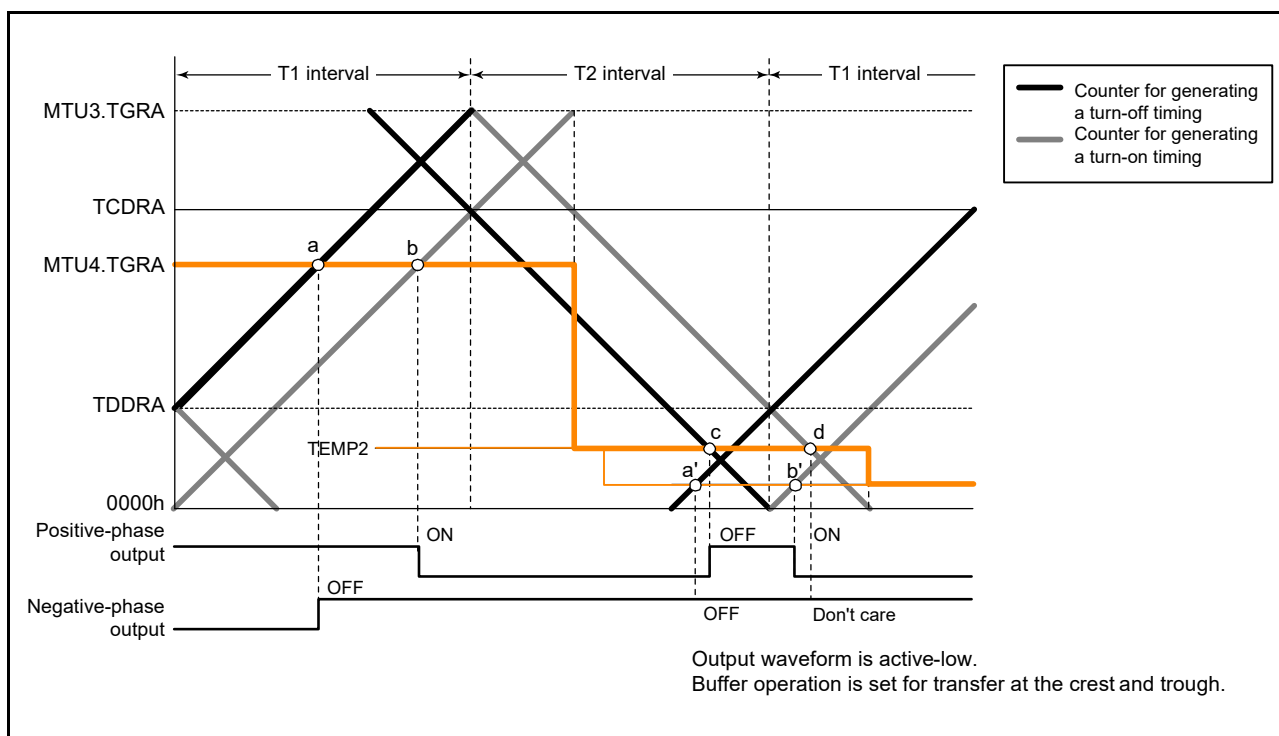


Figure 19.55 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3)

(k) 0% and 100% Duty Ratio Output in Complementary PWM Mode

In complementary PWM mode, 0% and 100% duty PWM output can be output as required. Figure 19.56 to Figure 19.60 show output examples.

A 100% duty waveform is output when the compare register value is set to 0000h. The waveform in this case has a positive phase with a 100% on-state. A 0% duty waveform is output when the compare register value is set to the same value as MTU3.TGRA. The waveform in this case has a positive phase with a 100% off-state.

Turn-on and turn-off compare matches occur simultaneously, but if a turn-on compare match and turn-off compare match for the same phase occur simultaneously, both compare matches are ignored and the waveform does not change.

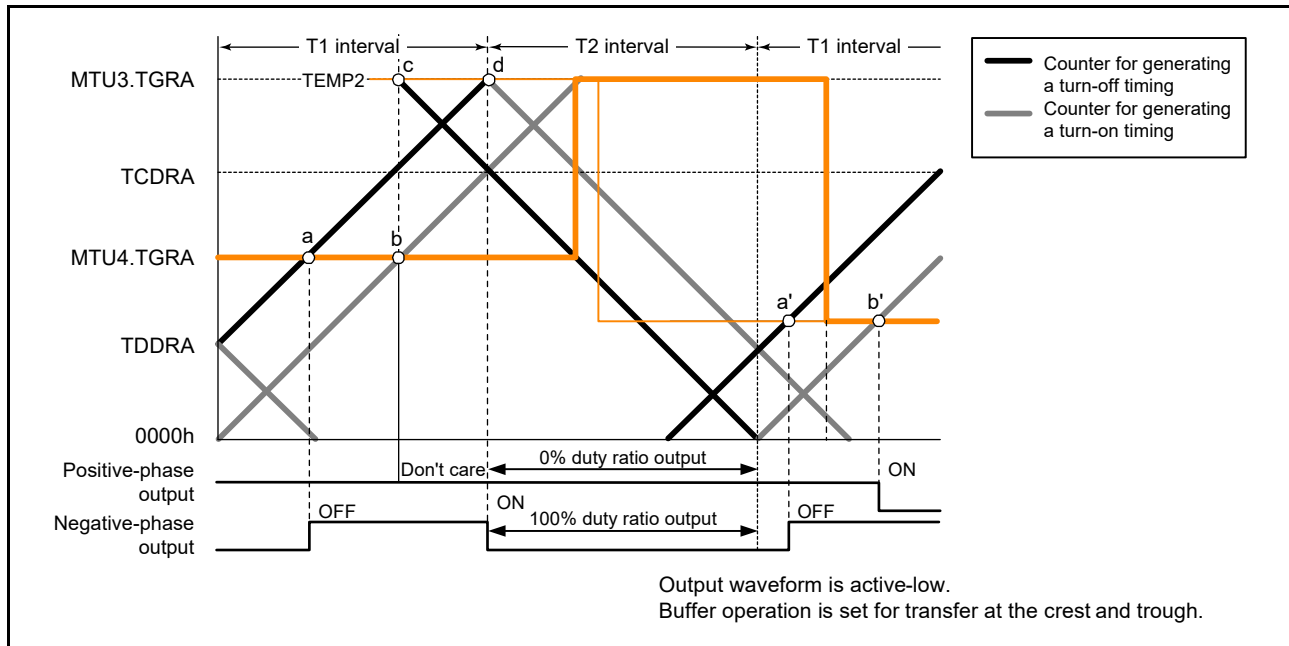


Figure 19.56 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1)

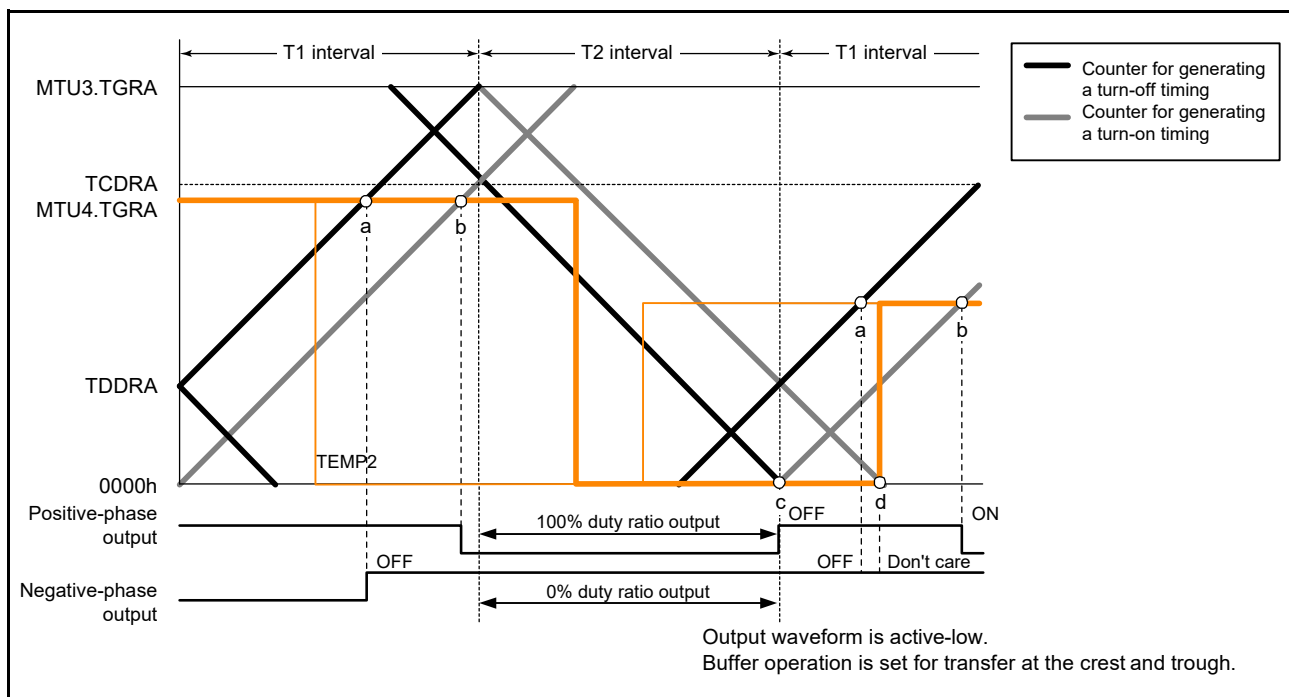


Figure 19.57 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2)

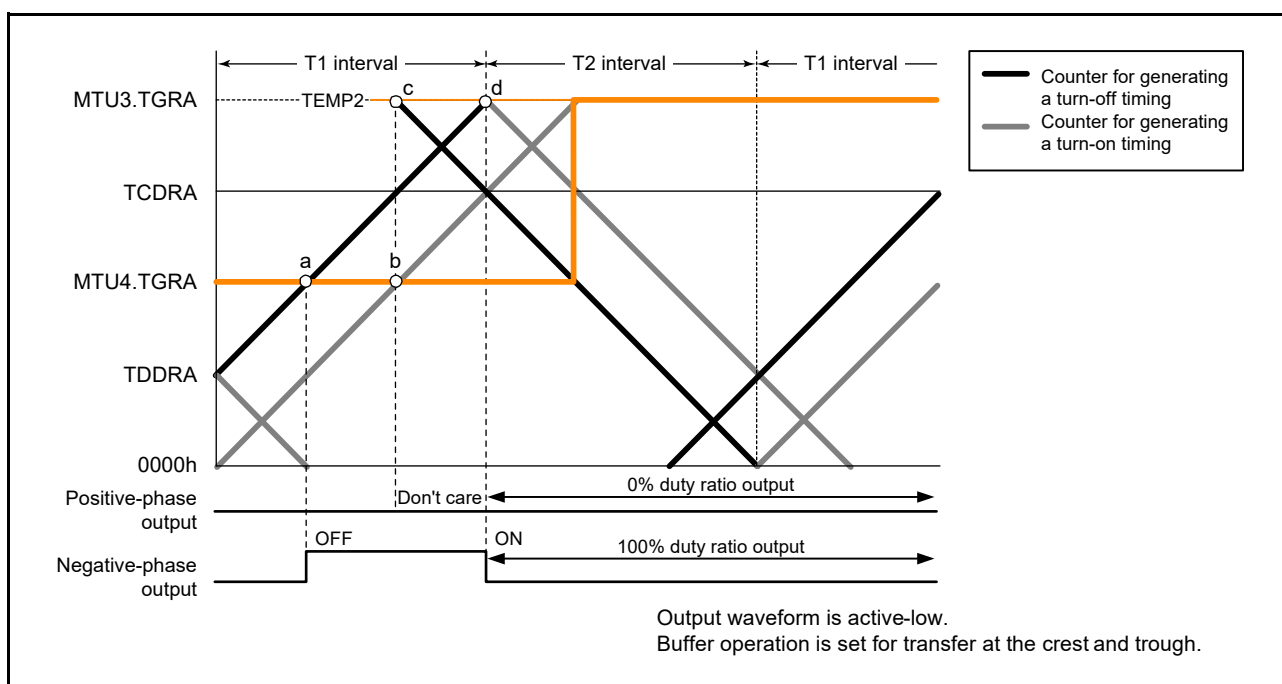


Figure 19.58 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3)

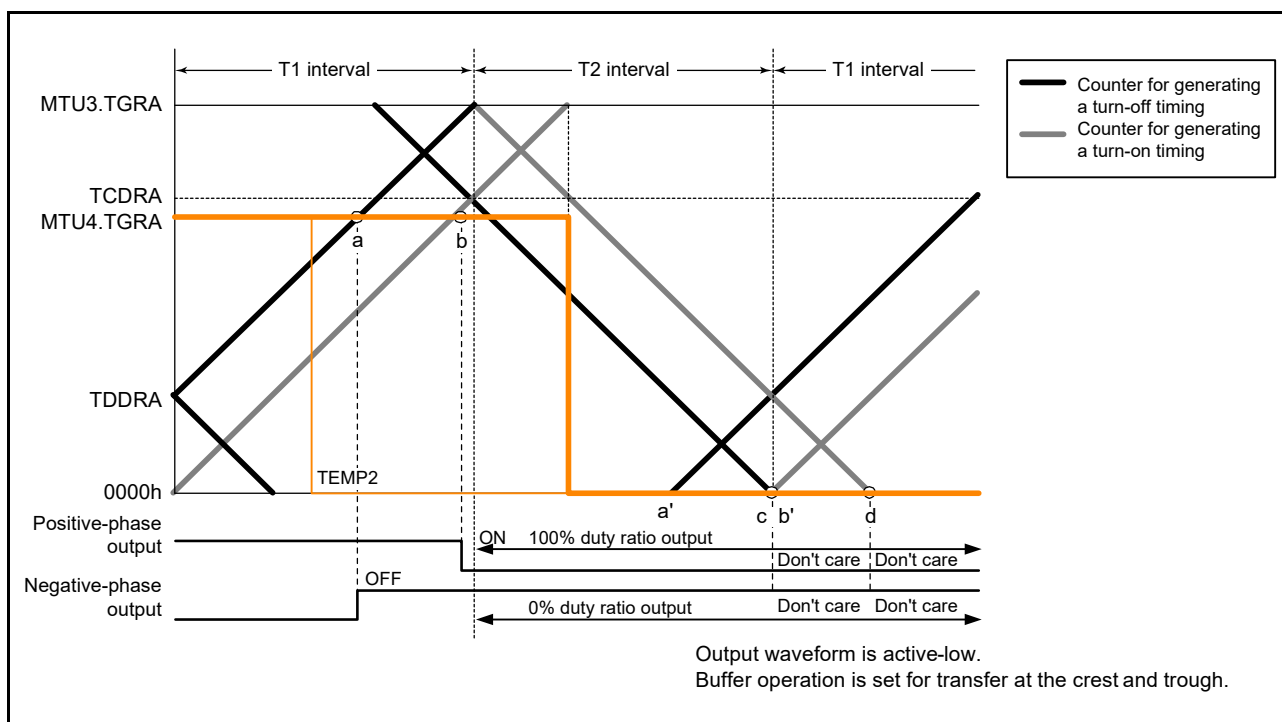


Figure 19.59 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (4)

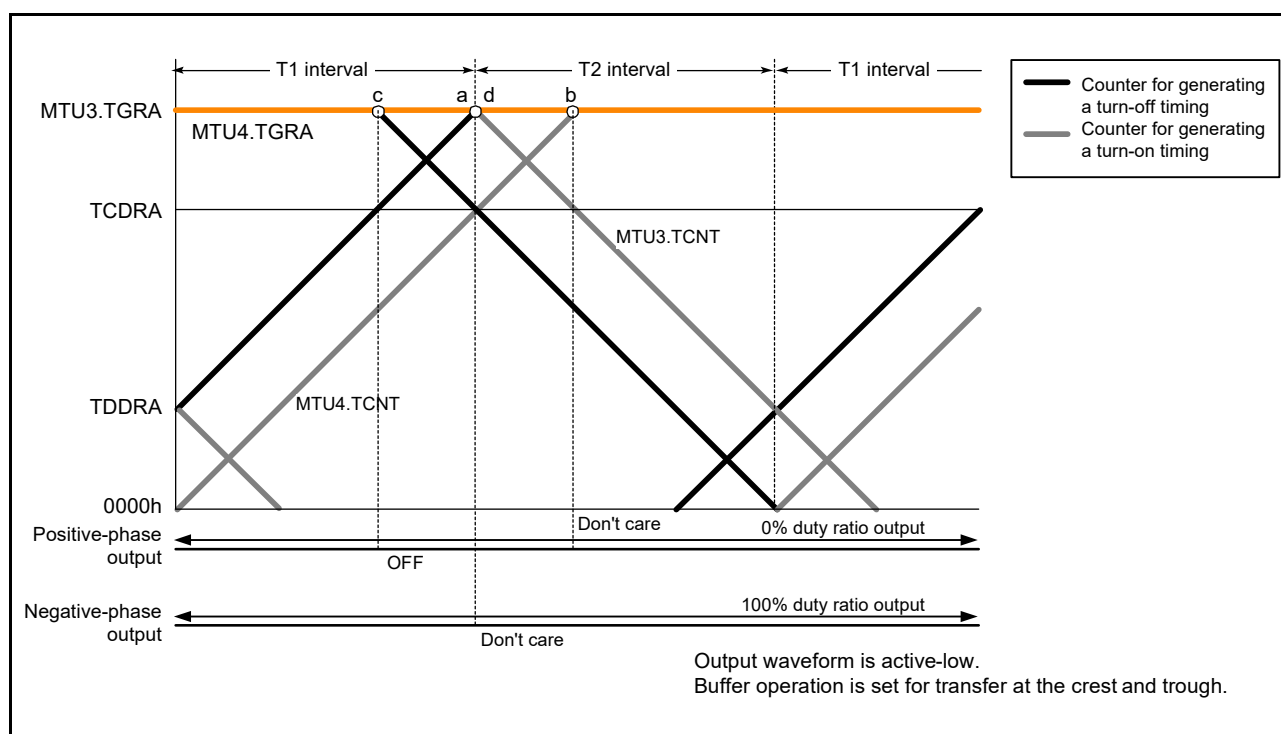


Figure 19.60 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (5)

(I) Toggle Output Synchronized with PWM Period

In complementary PWM mode, toggle output from the PWM output pin in synchronization with the PWM period can be enabled by setting the PSYE bit in the TOCR1A register to 1. An example of a toggle output waveform is shown in Figure 19.61.

This output is toggled by a compare match between MTU3.TCNT and MTU3.TGRA and a compare match between MTU4.TCNT and 0000h.

pin is assigned for this toggle output. The initial output is high-level output.

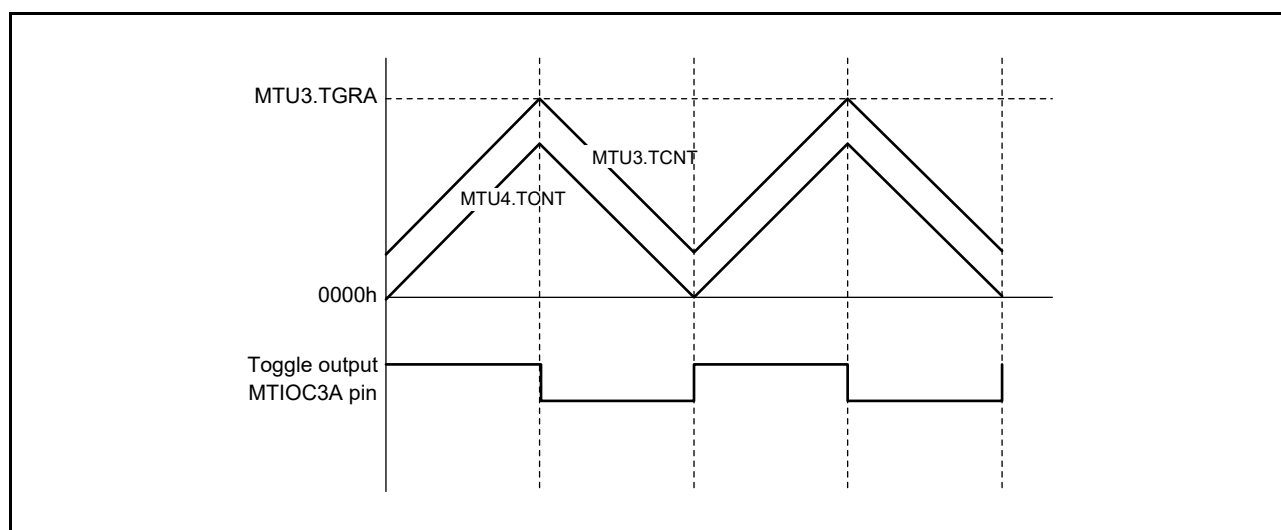


Figure 19.61 Example of Toggle Output Waveform Synchronized with PWM Output (MTU3 and MTU4)

(m) Counter Clearing by Another Channel

In complementary PWM mode, MTU3.TCNT, MTU4.TCNT, and TCNTSA can be cleared by another channel source when a mode for synchronization with another channel is specified through the TSYRA register and synchronous clearing is selected with MTU3.TCR.CCLR[2:0] bits.

Figure 19.62 illustrates an example of this operation.

Use of this function enables a counter to be cleared and restarted through an external signal.

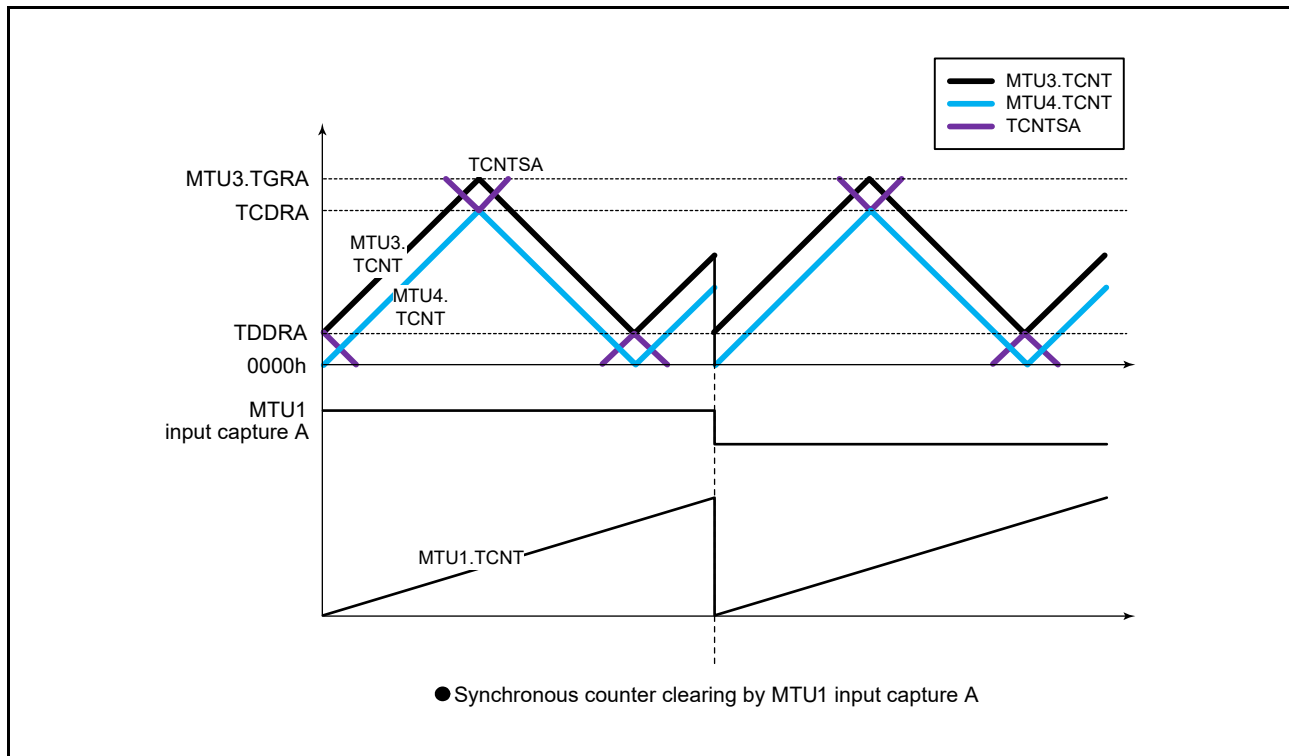


Figure 19.62 Counter Clearing Synchronized with Another Channel (MTU3 and MTU4)

(n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Setting the WRE bit in TWCRA to 1 suppresses initial output when synchronous counter clearing occurs in the Tb interval (Tb2 interval) at the trough in complementary PWM mode and controls abrupt change in duty cycle at synchronous counter clearing.

Initial output suppression through the WRE bit = 1 is applicable only when synchronous clearing occurs in the Tb2 interval as indicated by (10) or (11) in Figure 19.63. When synchronous clearing occurs outside that interval, the initial value specified by the OLSN and OLSP bits in TOCR1A is output. Even in the Tb2 interval, if synchronous clearing occurs in the initial value output period (indicated by (1) in Figure 19.63) immediately after the counters start operation, initial value output is not suppressed.

This function can be used in both channel combinations of MTU3 and MTU4. In MTU3 and MTU4, synchronous clearing in any of MTU0 to MTU2 can cause counter clearing.

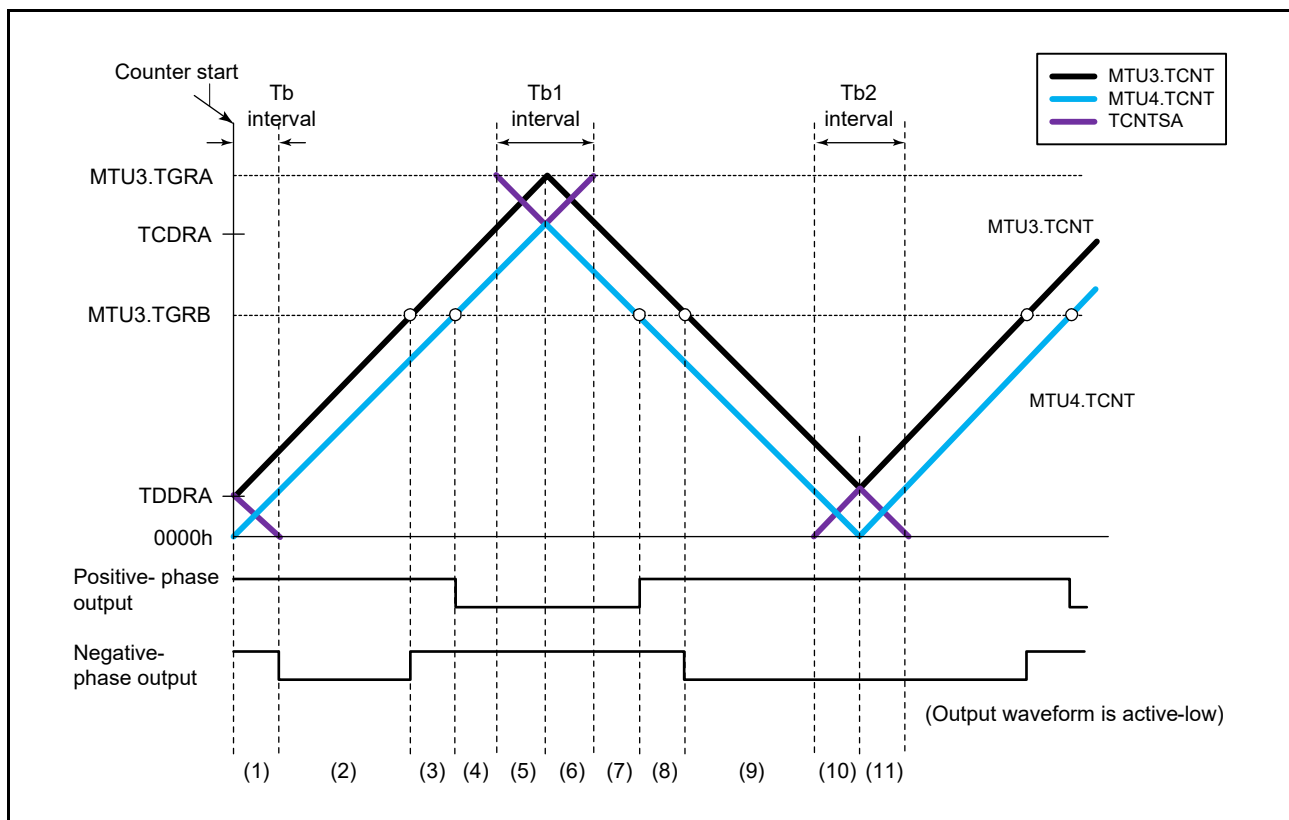


Figure 19.63 Timing for Synchronous Counter Clearing (MTU3 and MTU4)

- Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode.

An example of the procedure for setting output waveform control at synchronous counter clearing in complementary PWM mode is shown in Figure 19.64.

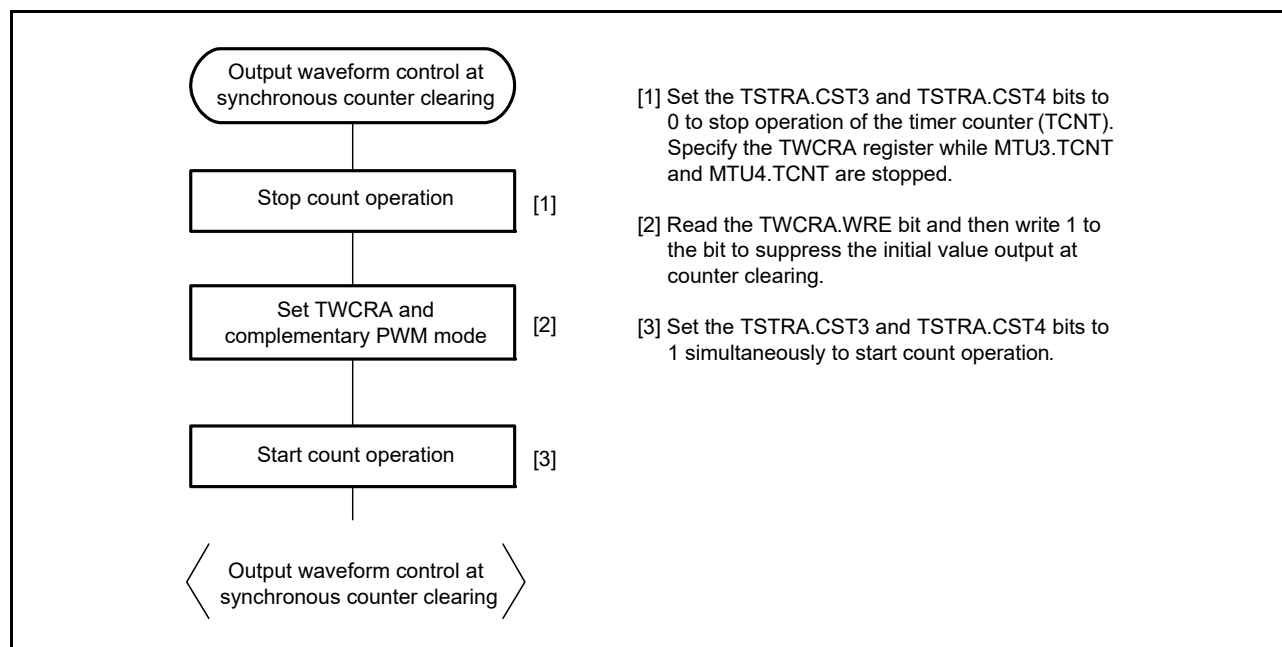


Figure 19.64 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode (MTU3 and MTU4)

- Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Figure 19.65 to Figure 19.68 show examples of output waveform control in which MTU3 and MTU4 operate in complementary PWM mode and synchronous counter clearing is generated while the WRE bit in TWCRA is set to 1. In the examples shown in Figure 19.65 to Figure 19.68, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in Figure 19.63, respectively.

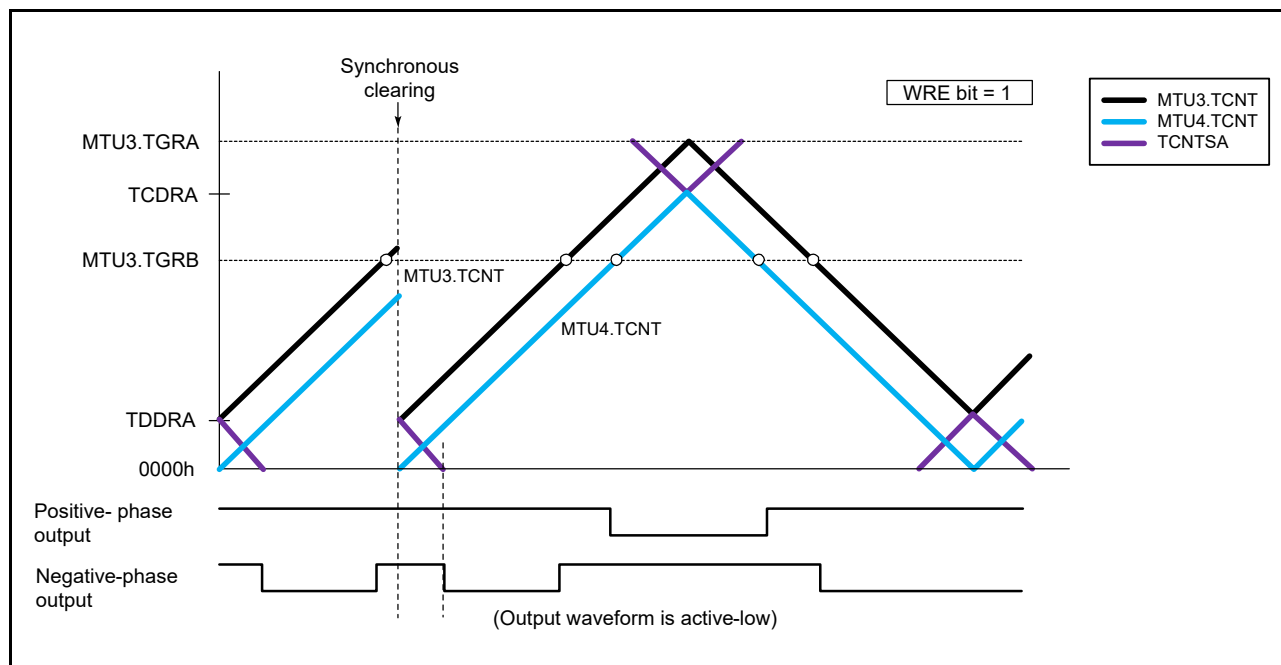


Figure 19.65 Example of Synchronous Clearing in Dead Time during Up-Counting
(Timing (3) in Figure 19.63; TWCRA.WRE Bit is 1)

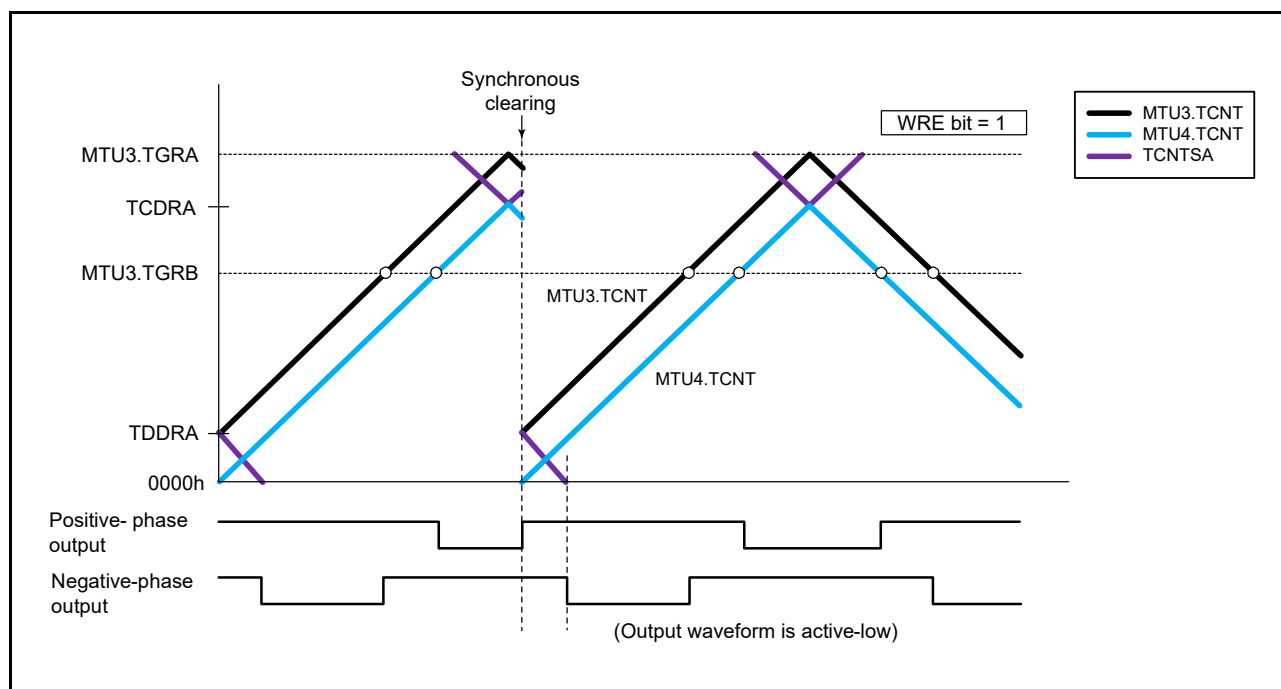


Figure 19.66 Example of Synchronous Clearing in Tb1 interval
(Timing (6) in Figure 19.63; TWCRA.WRE Bit is 1)

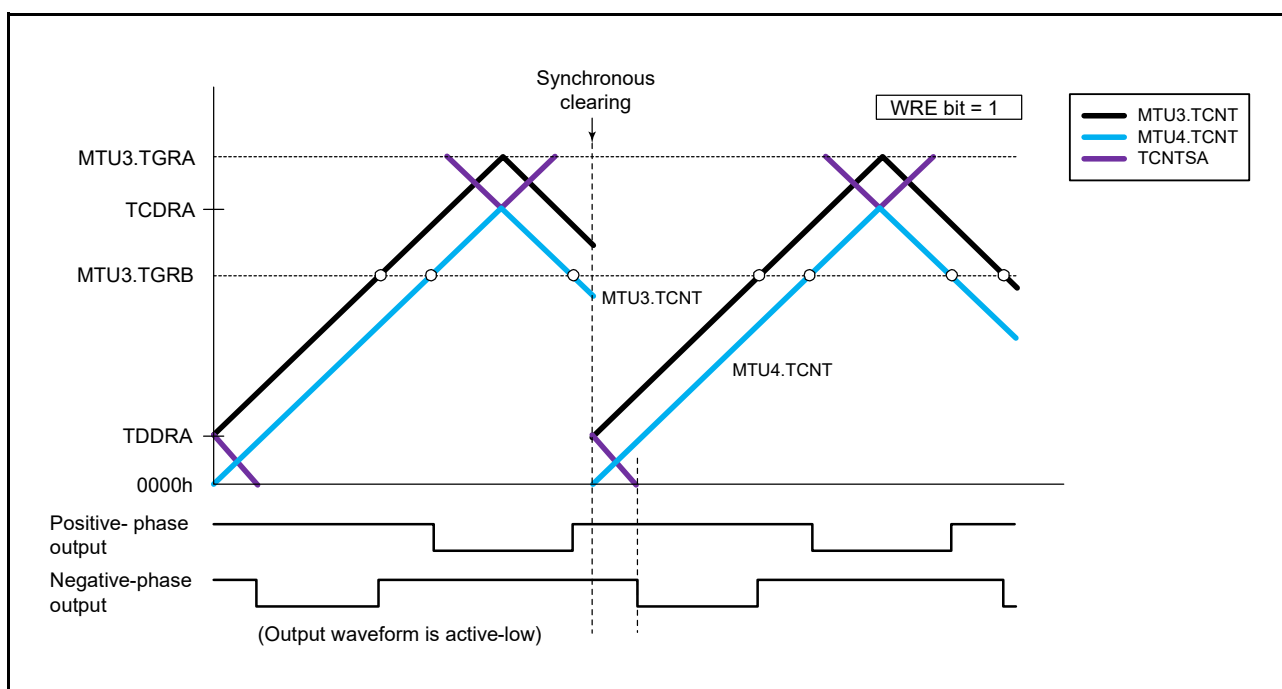


Figure 19.67 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 19.63; TWCRA.WRE Bit is 1)

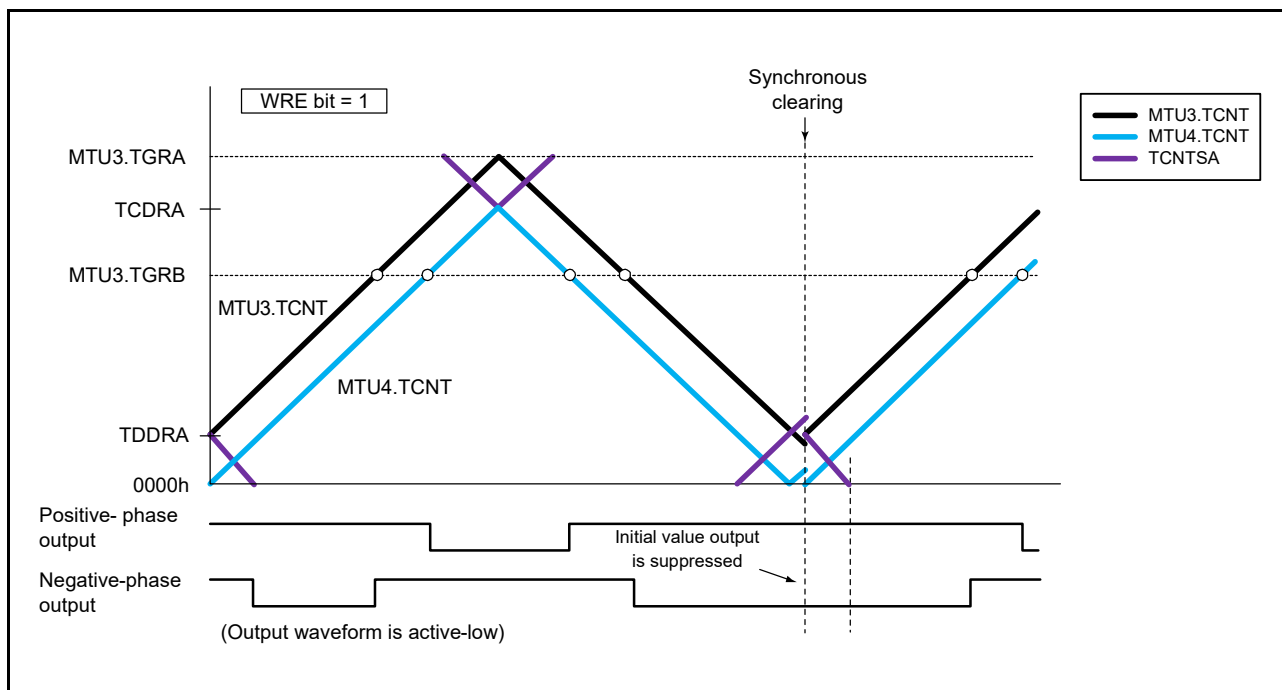


Figure 19.68 Example of Synchronous Clearing in Tb2 interval (Timing (11) in Figure 19.63; TWCRA.WRE Bit is 1)

(o) Counter Clearing by MTU3.TGRA Compare Match

In complementary PWM mode, MTU3.TCNT, MTU4.TCNT, and TCNTSA can be cleared by MTU3.TGRA compare match when the TWCRA.CCE bit.

Figure 19.69 illustrates an operation example.

Note 1. Use this function only in complementary PWM mode 1 (transfer at crest).

Note 2. Do not specify synchronous clearing by another channel (do not set 1 in the SYNC0 to SYNC4 bits in the timer synchronous register (TSYRA)).

Note 3. Do not set the PWM duty value to 0000h.

Note 4. Do not set the PSYE bit in timer output control register 1 (TOCR1A) to 1.

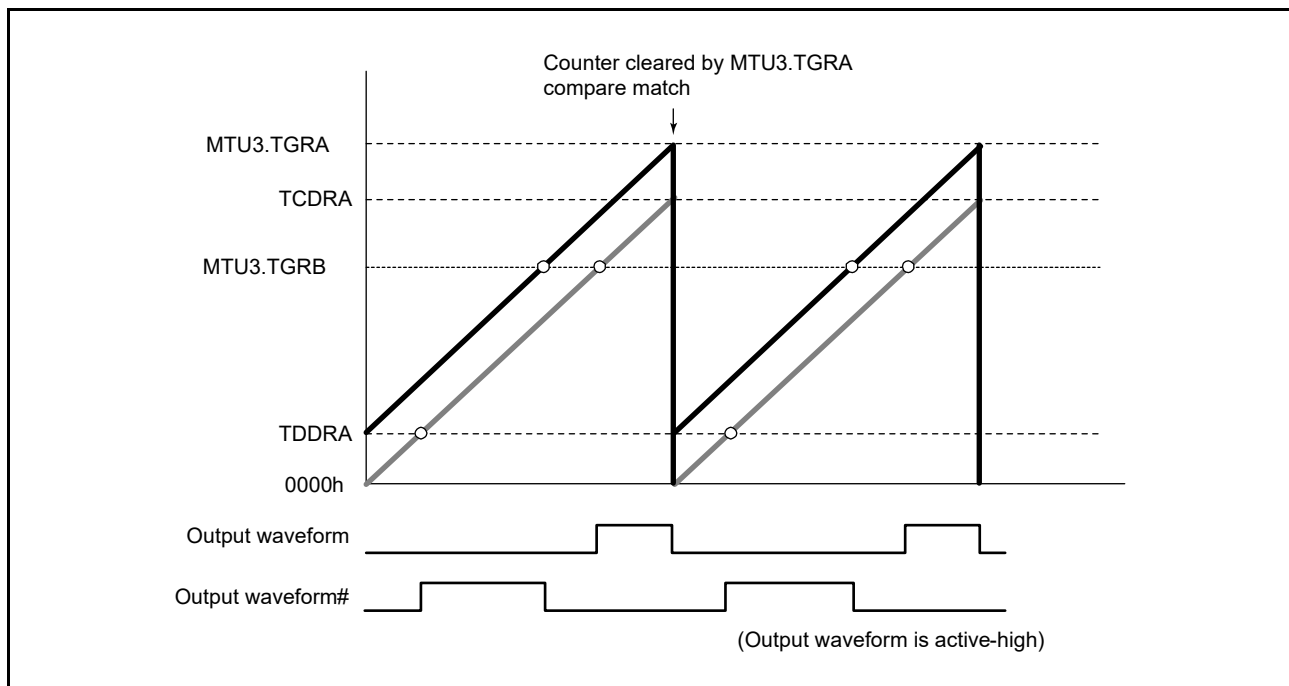


Figure 19.69 Example of Counter Clearing Operation by MTU3.TGRA Compare Match

(p) Example of Waveform Output for Driving AC Synchronous Motor (Brushless DC Motor)

In complementary PWM mode when MTU3 and MTU4 are used, a brushless DC motor can easily be controlled using the TGCRA register. Figure 19.70 to Figure 19.73 show examples of brushless DC motor driving waveforms created using TGCRA.

To switch the output phases for a 3-phase brushless DC motor by means of external signals detected with a Hall element, etc., set the TGCRA.FB bit to 0. In this case, the external signals indicating the magnetic pole position should be input to timer input pins MTIOC0A, MTIOC0B, and MTIOC0C in MTU0 (make appropriate settings with the MPC and port mode registers (PMR) of the I/O ports). When an edge is detected at pin MTIOC0A, MTIOC0B, or MTIOC0C, the output on/off state is switched automatically.

When the TGCRA.FB bit is 1, the output on/off state is switched when the UF, VF, or WF bit in TGCRA is set to 0 or 1. The driving waveforms are output from the 6-phase PWM output pins for complementary PWM mode.

With this 6-phase output, while the output is turned on, chopping output is available through complementary PWM mode output function by setting the N bit or P bit in TGCRA to 1. When the N bit or P bit is 0, the level output is selected. The active level of the 6-phase output (on output level) can be set with the TOCR1A.OLSN and TOCR1A.OLSP bits regardless of the setting of the N and P bits.

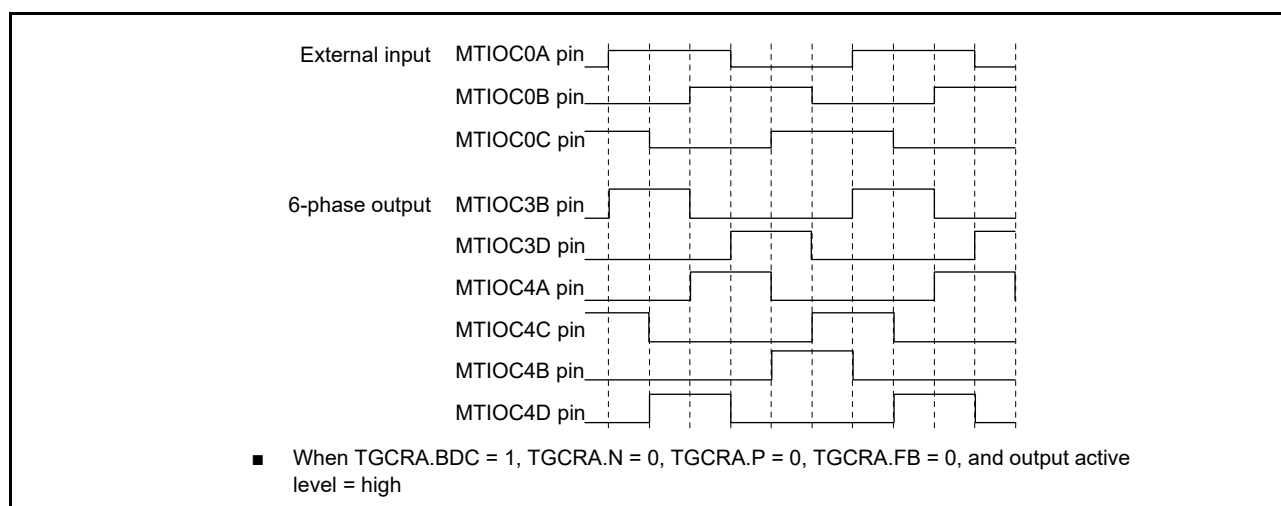


Figure 19.70 Example of Output Phase Switching by External Input (1)

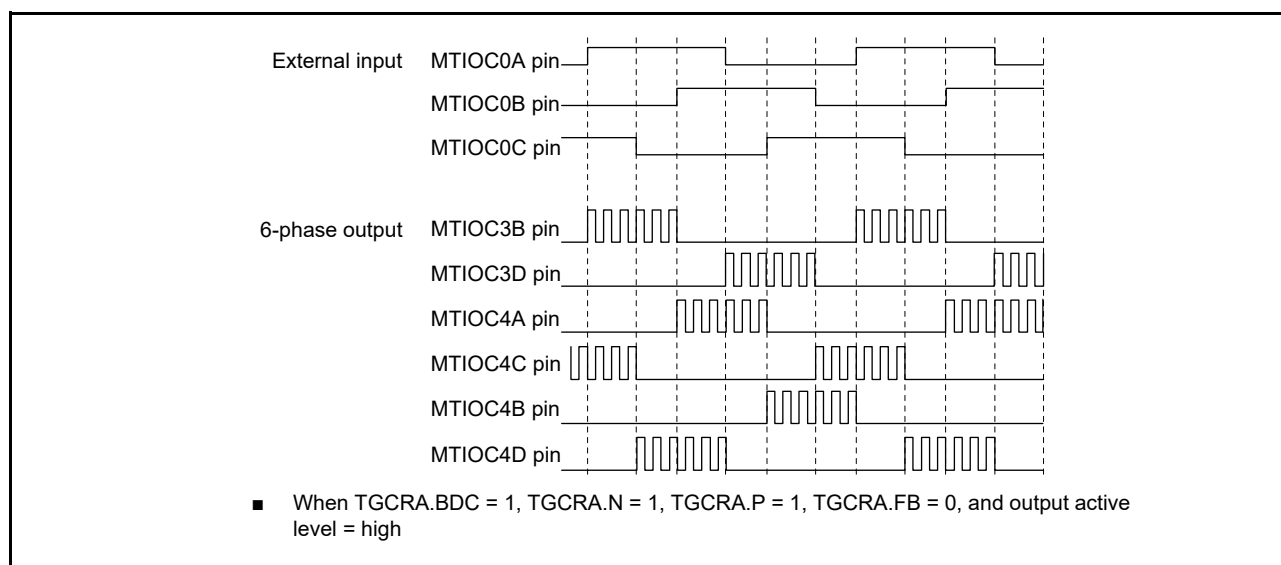


Figure 19.71 Example of Output Phase Switching by External Input (2)

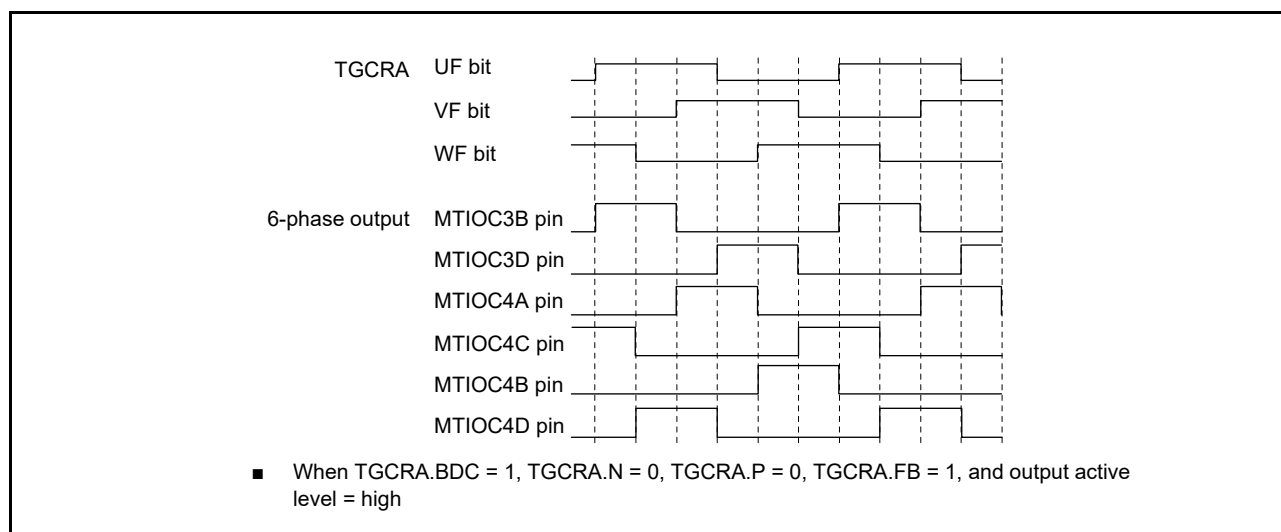


Figure 19.72 Example of Output Phase Switching through UF, VF, and WF Bit Settings (1)

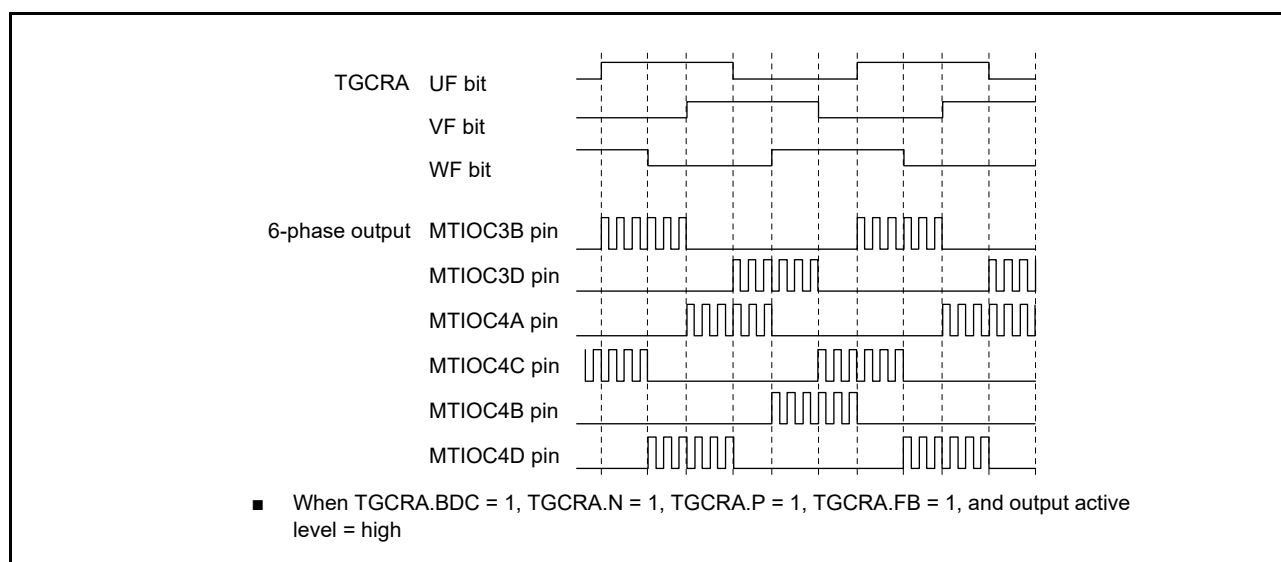


Figure 19.73 Example of Output Phase Switching through UF, VF, and WF Bit Settings (2)

(q) A/D Conversion Start Request Setting

In complementary PWM mode, an A/D conversion start request can be issued using MTU3.TGRA compare match, MTU4.TCNT underflow (trough), or compare match on a channel other than MTU3 and MTU4.

When start requests using MTU3.TGRA compare match are specified, A/D conversion can be started at the crest of the MTU3.TCNT count.

A/D conversion start requests can be specified by setting the TIER.TTGE bit. To issue an A/D conversion start request at an MTU4.TCNT underflow (trough), set the MTU4.TIER.TTGE2 bit to 1.

(r) Double Buffer Function in Complementary PWM Mode

In complementary PWM mode 3 (transfer at the crest and trough), the PWM output setting resolution can be improved from ± 2 to ± 1 by setting the TMDR2A.DRS bit to 1.

When setting buffer registers A (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD), set also buffer registers B (MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF) at the same time. For details of the setting procedure, refer to section 19.3.8 (1), Example of Complementary PWM Mode Setting Procedure

Note: When a buffer register B is set to the buffer register A value, symmetric PWM waveforms are output. When a buffer register B is not set to the buffer register A value, asymmetric PWM waveforms are output.

Figure 19.74 shows an example of double buffer operation.

Each register data is transferred as follows.

- After MTU4.TGRD (buffer A) is written to, data is transferred from MTU4.TGRD (buffer A) to TEMP3A (temporary A) and from MTU4.TGRF (buffer B) to TEMP3B (temporary B).
- With timing (1) in the figure, data is transferred from TEMP3A (temporary A) to MTU4.TGRB (compare).
- With timing (2) in the figure, data is transferred from TEMP3B (temporary B) to MTU4.TGRB (compare).

In the crest interval (Tb1 interval), the compare register and temporary register A are valid; in the trough interval (Tb2 interval), the compare register and temporary register B are valid.

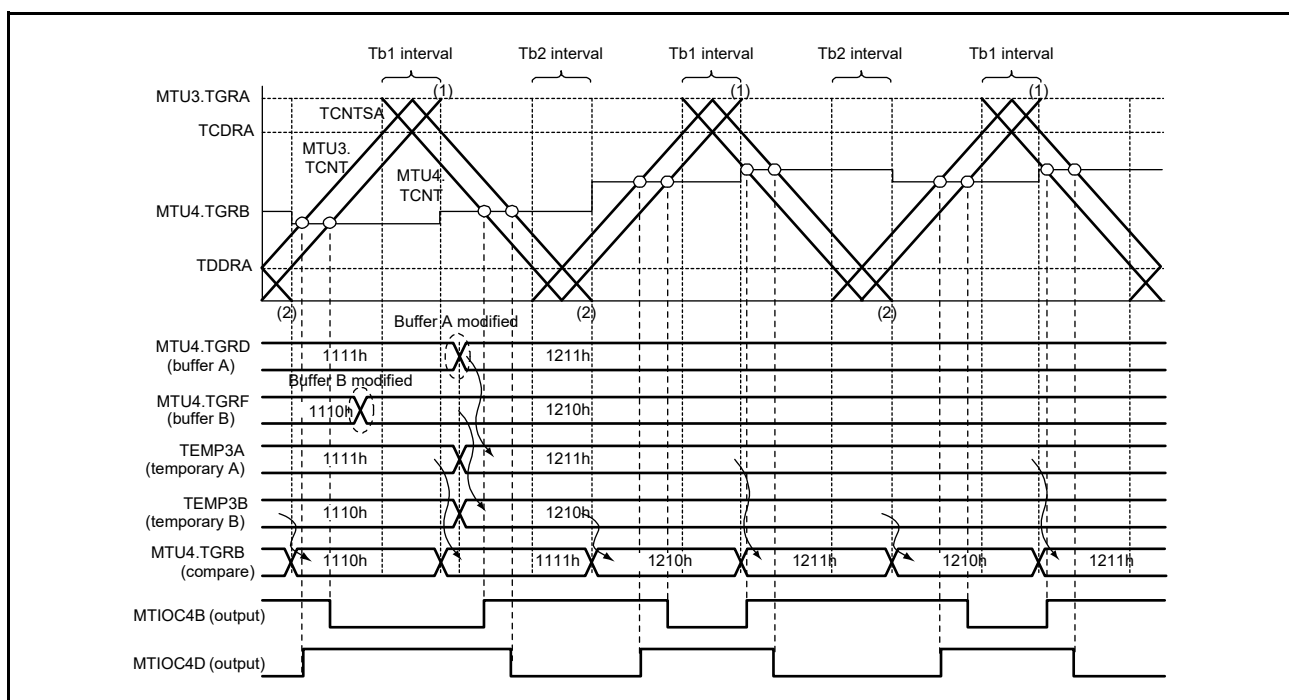


Figure 19.74 Example of Double Buffer Operation

Figure 19.75 shows an example when the buffer write value is smaller than the TDDRA value, and Figure 19.76 shows an example when the write value is greater than TCDRA.

In the crest interval, the output is controlled according to the compare match with the compare register or temporary register A; in the trough interval, the output is controlled according to the compare match with the compare register or temporary register B.

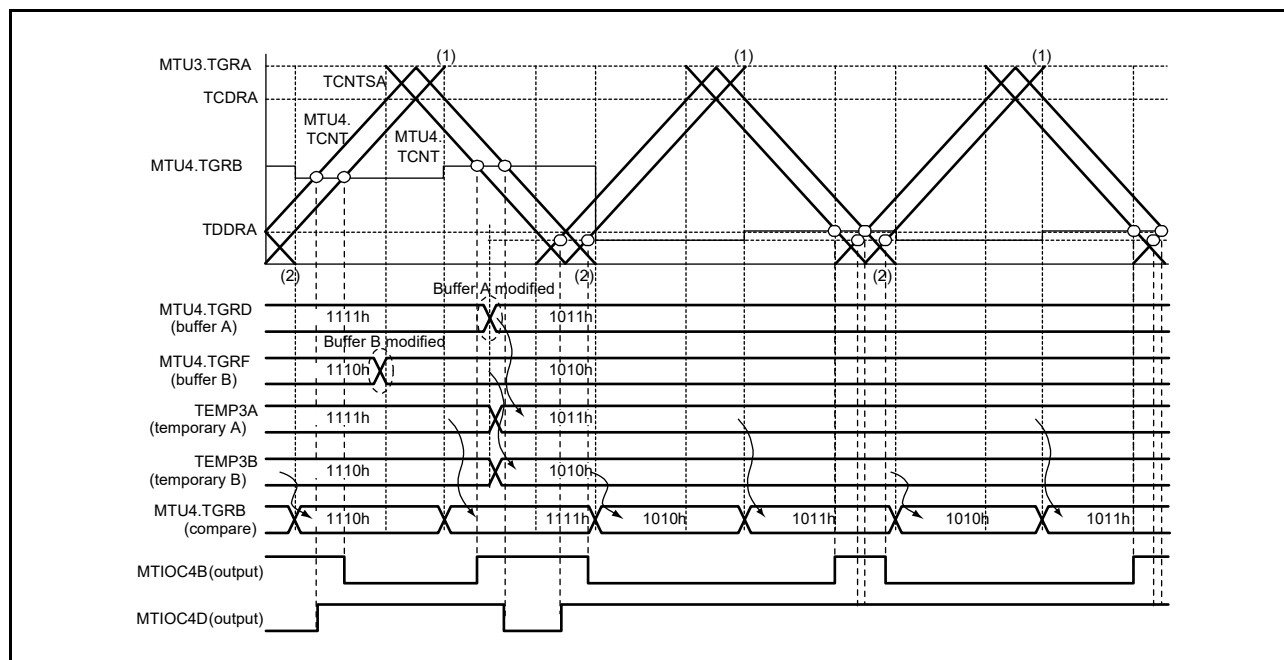


Figure 19.75 Example of Double Buffer Operation (Buffer Write Value is Smaller than TDDRA)

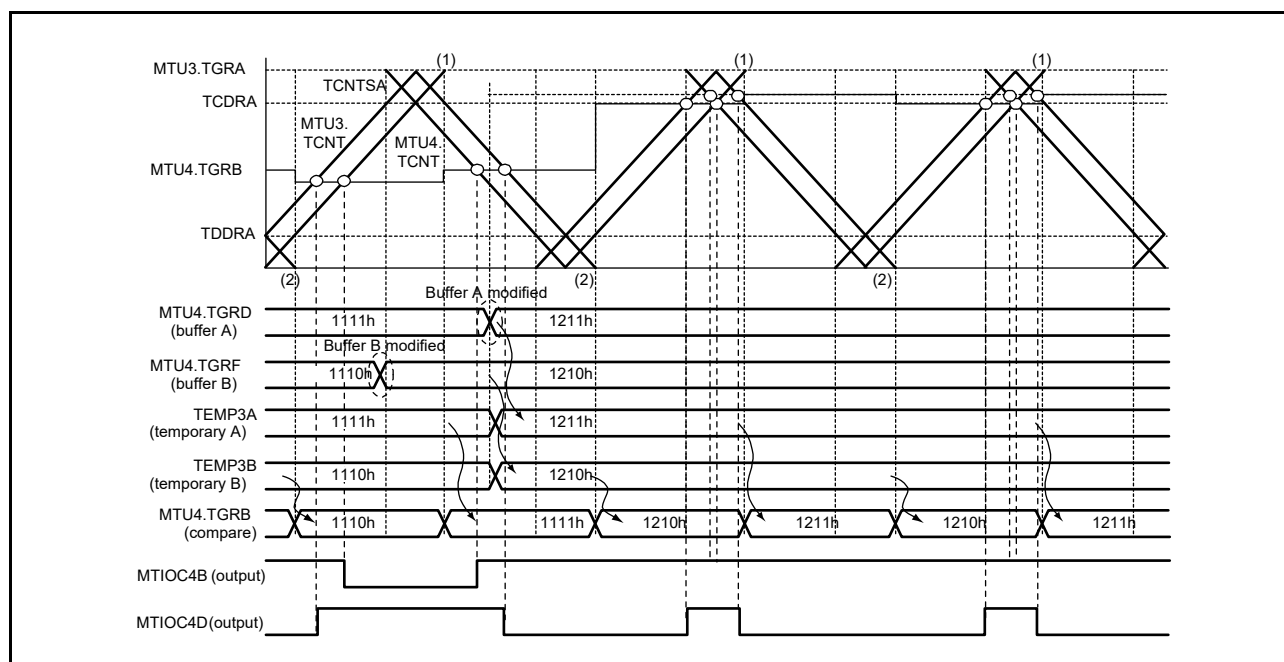


Figure 19.76 Example of Double Buffer Operation (Buffer Write Value is Greater than TCDRA)

(3) Interrupt Skipping Function 1 in Complementary PWM Mode

Interrupts TGIA3 (at the crest) and TCIV4 (at the trough) in MTU3 and MTU4 can be skipped up to seven times by making settings in the TITCR1A register.

Transfers from a buffer register to a temporary register or a compare register can be skipped in coordination with interrupt skipping by making settings in the TBTERA register. For the linkage with buffer registers, refer to description (c), Buffer Transfer Control Linked with Interrupt Skipping, below.

A/D conversion start requests generated by the A/D conversion start request delaying function can also be skipped in coordination with interrupt skipping by making settings in the MTU4.TADCR register. For the linkage with the A/D conversion start request delaying function, refer to section 19.3.9, A/D Conversion Start Request Delaying Function.

The TITCR1A register should be set while interrupt skipping function 1 is selected by setting the TITM bit in the timer interrupt skipping mode register (TITMRA) to 0, TGIA3 interrupt requests are disabled by setting the MTU3.TIER register, TCIV4 interrupt requests are disabled by setting the MTU4.TIER register, and a compare match is not generated. Before changing the skipping count, be sure to set the T3AEN and T4VEN bits to 0 to clear the skipping counter.

(a) Example of Interrupt Skipping Function 1 Setting Procedure

Figure 19.77 shows an example of the interrupt skipping function 1 setting procedure. Figure 19.78 shows the periods during which interrupt skipping count can be changed.

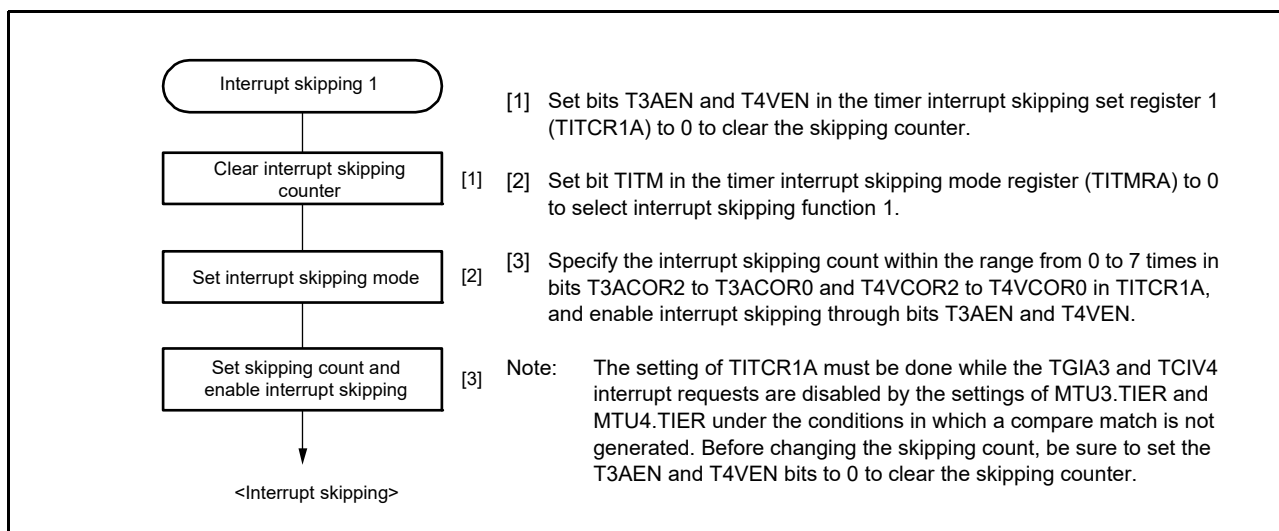


Figure 19.77 Example of Interrupt Skipping Function 1 Setting Procedure

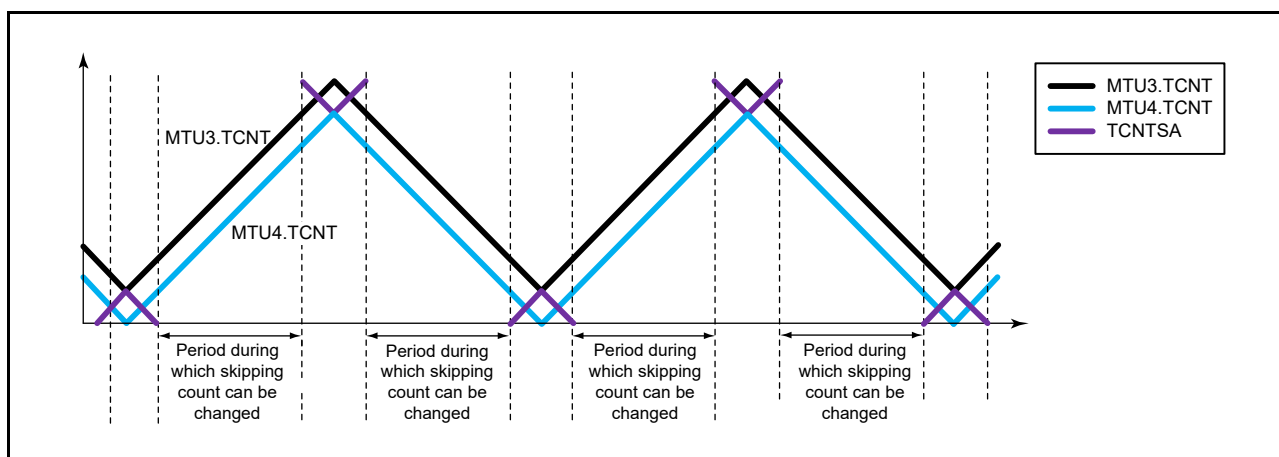


Figure 19.78 Periods during which Interrupt Skipping Count can be Changed

(b) Example of Interrupt Skipping Function 1

Figure 19.79 shows an example of TGIA3 interrupt skipping in which the interrupt skipping count is set to three by the T3ACOR bits and the T3AEN bit is set to 1 in the TITCR1A register.

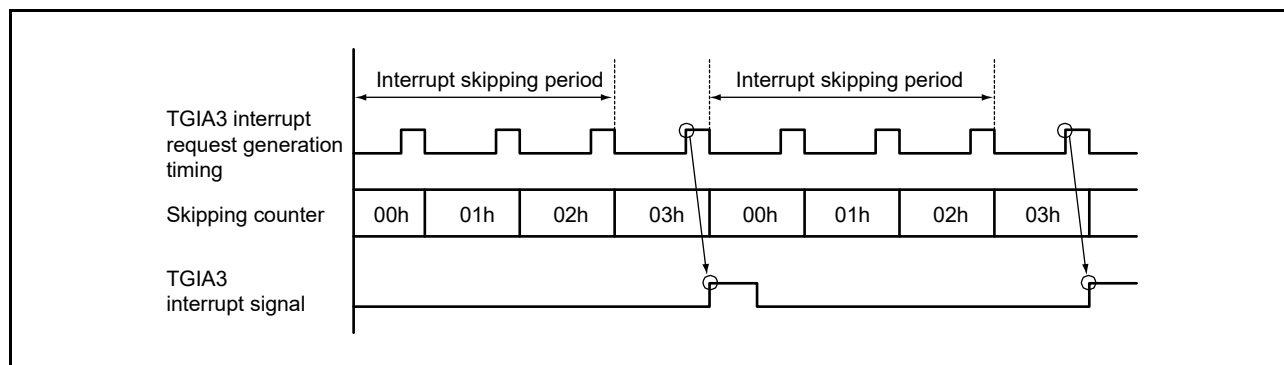


Figure 19.79 Example of Interrupt Skipping Function 1

(c) Buffer Transfer Control Linked with Interrupt Skipping

In complementary PWM mode, whether to transfer data from a buffer register to a temporary register and whether to link the transfer with interrupt skipping can be specified with the BTE[1:0] bits in the TBTERA register.

Figure 19.80 shows an example of operation when buffer transfer is disabled (BTE[1:0] = 01b). While this setting is valid, data is not transferred from the buffer register to the temporary register.

Figure 19.81 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE[1:0] = 10b).

While this setting is valid, data is not transferred from the buffer register outside the buffer transfer-enabled period.

Note that the buffer transfer-enabled period differs depending on whether only the T3AEN bit in the TITCR1A register is set to 1, only the T4VEN bit in the TITCR1A register is set to 1, or both the T3AEN and T4VEN bits are set to 1. Figure 19.82 shows the relationship between the T3AEN and T4VEN bit settings in TITCR1A and buffer transfer-enabled period.

Note: This function must be used in combination with interrupt skipping function 1.

When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register 1 (TITCR1A) are set to 0 or the skipping count set bits (T3ACOR and T4VCOR) in TITCR1A are set to 0), make sure that buffer transfer is not linked with interrupt skipping (set the BTE1 bit in TBTERA to 0).

If buffer transfer is linked with interrupt skipping while interrupt skipping is disabled, buffer transfer is never performed.

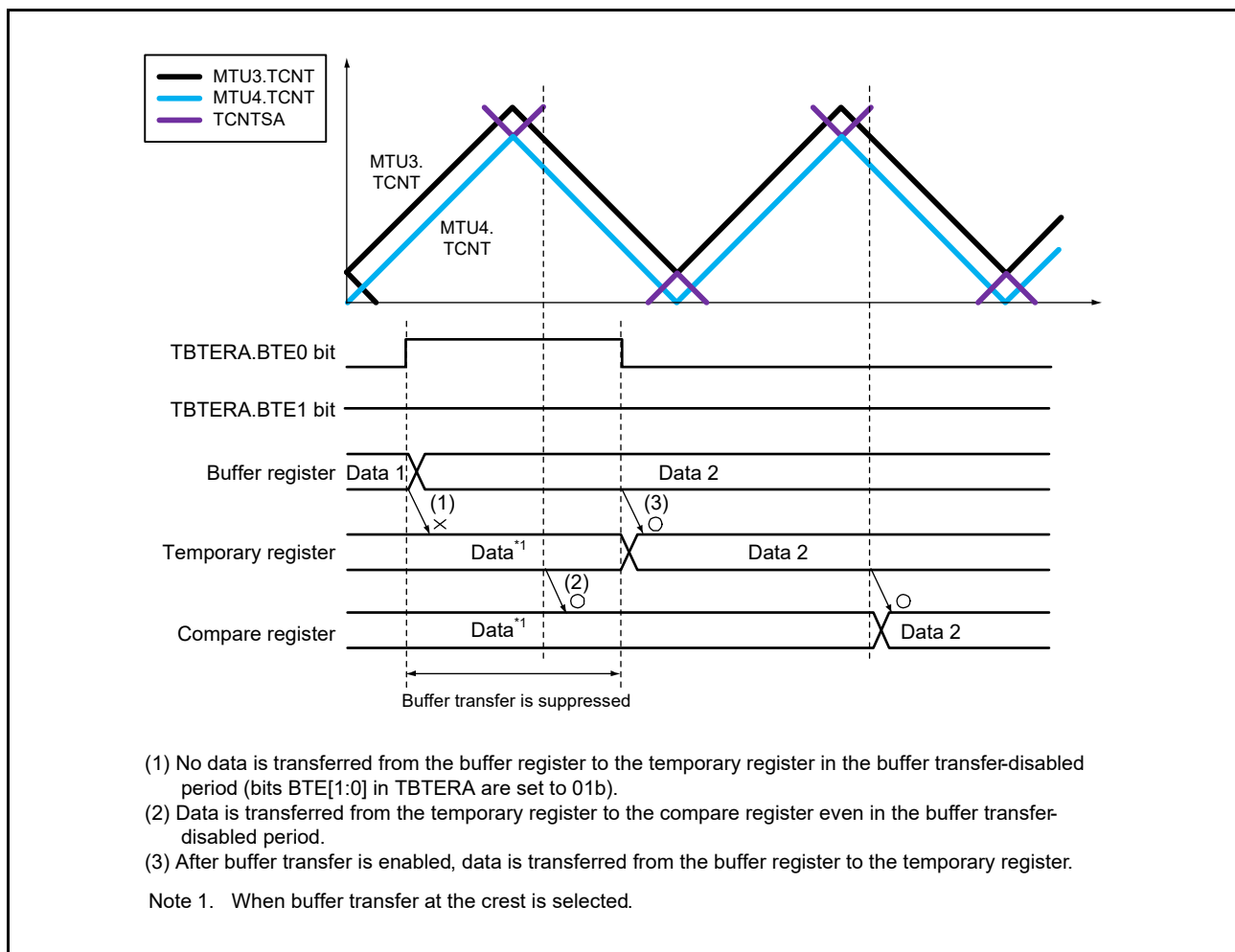


Figure 19.80 Example of Operation When Buffer Transfer is Disabled (BTE[1:0] = 01b)

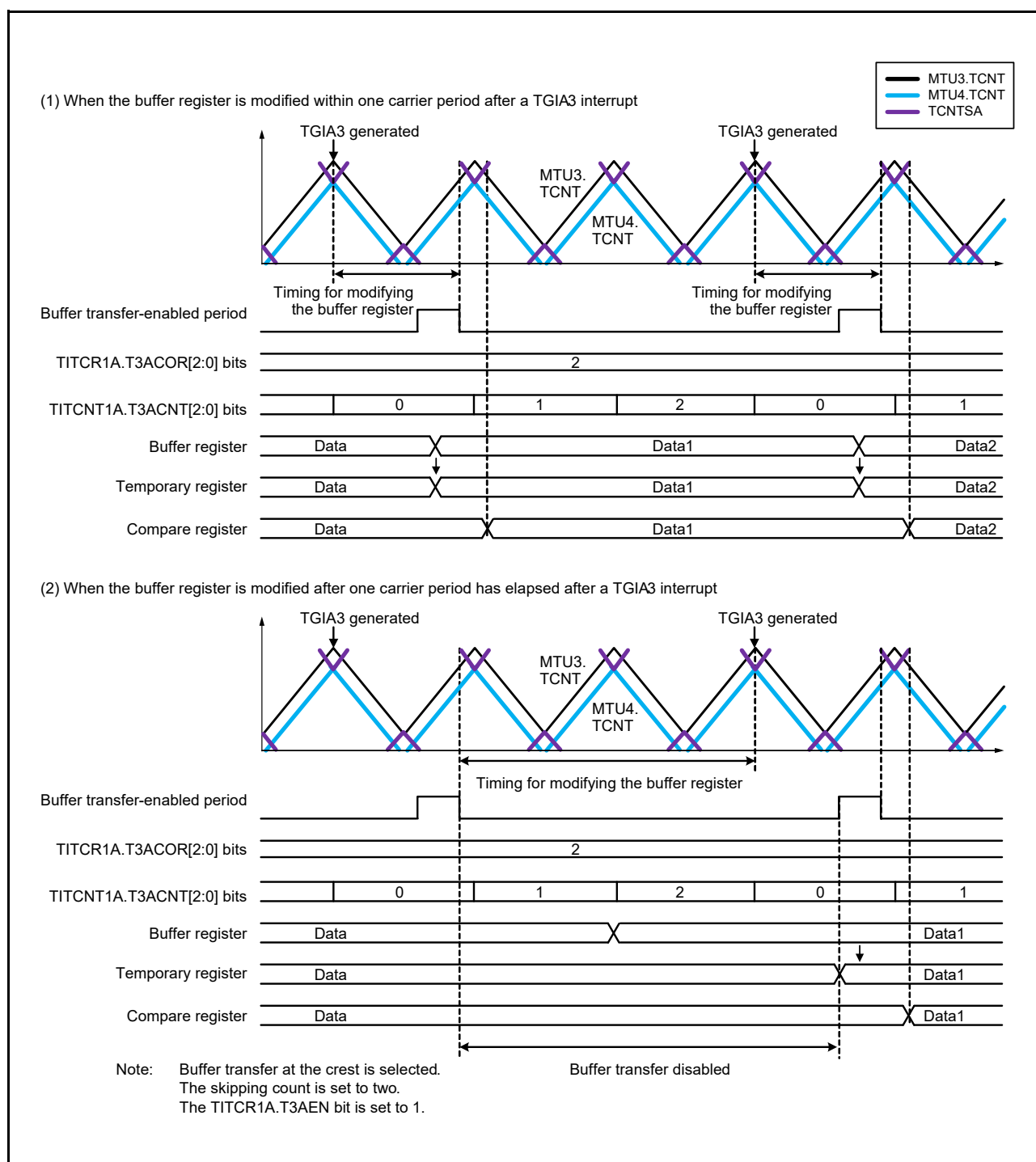


Figure 19.81 Example of Operation When Buffer Transfer is Linked with Interrupt Skipping (BTE[1:0] = 10b)

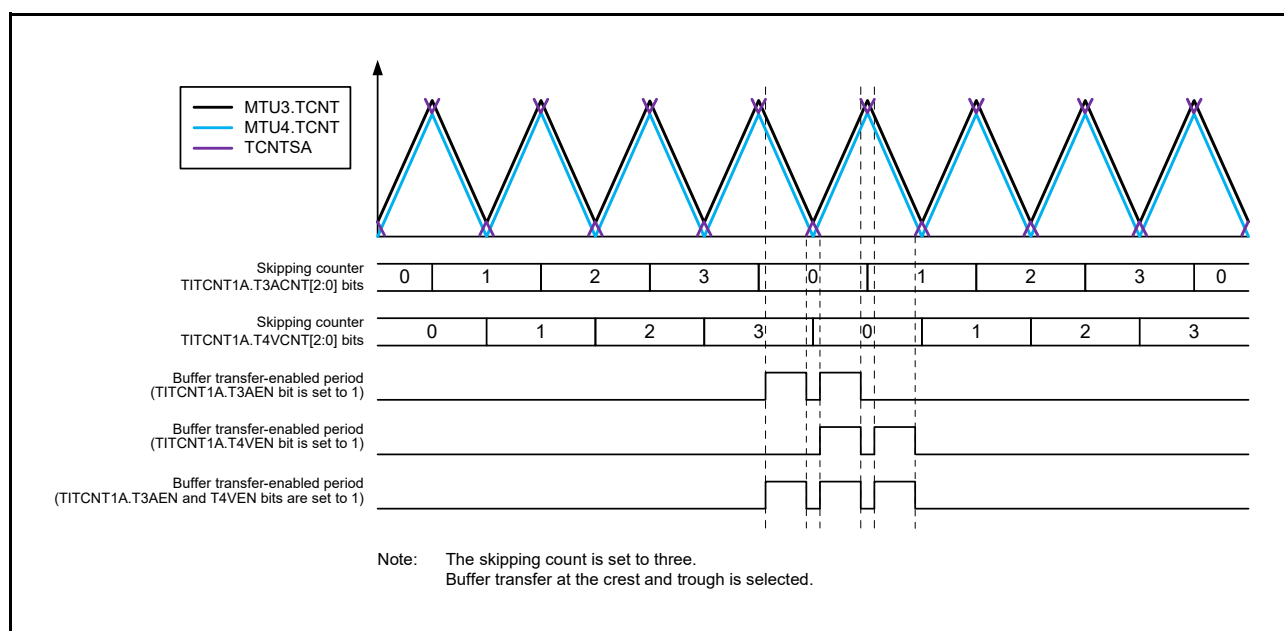


Figure 19.82 Relationship between Bits T3AEN and T4VEN in TITCR1A and Buffer Transfer-Enabled Period

(4) Complementary PWM Mode Output Protection Functions

The following output protection functions are provided for complementary PWM mode.

(a) Register and Counter Miswrite Prevention Function

Access from the CPU to the mode registers, control registers, compare registers, and counters can be enabled or disabled by setting the RWE bit in the TRWERA register. The applicable registers are some of the registers in MTU3 and MTU4 shown below:

24 registers in total

MTU3.TCR, MTU4.TCR, MTU3.TCR2, MTU4.TCR2, MTU3.TMDR1, MTU4.TMDR1, MTU3.TIORH, MTU4.TIORH, MTU3.TIORL, MTU4.TIORL, MTU3.TIER, MTU4.TIER, MTU3.TCNT, MTU4.TCNT, MTU3.TGRA, MTU4.TGRA, MTU3.TGRB, MTU4.TGRB, MTU.TOERA, MTU.TOCR1A, MTU.TOCR2A, MTU.TGCRA, MTU.TCDRA, and MTU.TDDRA

This function can disable CPU access to the mode registers, control registers, and counters to prevent miswriting due to CPU runaway. In the access-disabled state, the applicable registers are read as undefined and writing to these registers is ignored.

(b) Halting of PWM Output by External Signal

The PWM output pins of MTU0, MTU3, and MTU4 can be set to the high-impedance state automatically. Refer to section 20, Port Output Enable 3 (POE3E), for details.

19.3.9 A/D Conversion Start Request Delaying Function

A/D conversion start requests can be issued in MTU4 by making settings in the timer A/D conversion start request control register (MTU4.TADCR), timer A/D conversion start request cycle set registers (MTU4.TADCORA and MTU4.TADCORB), and timer A/D conversion start request cycle set buffer registers (MTU4.TADCOBRA and MTU4.TADCOBRB).

The A/D conversion start request delaying function compares MTU4.TCNT with MTU4.TADCORA or MTU4.TADCORB, and when their values match, the function issues a respective A/D conversion start request (TRG4AN or TRG4BN).

A/D conversion start requests (TRG4AN and TRG4BN) can be skipped in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in MTU4.TADCR.

(1) Example of Procedure for Specifying A/D Conversion Start Request Delaying Function

Figure 19.83 shows an example of procedure for specifying the A/D conversion start request delaying function.

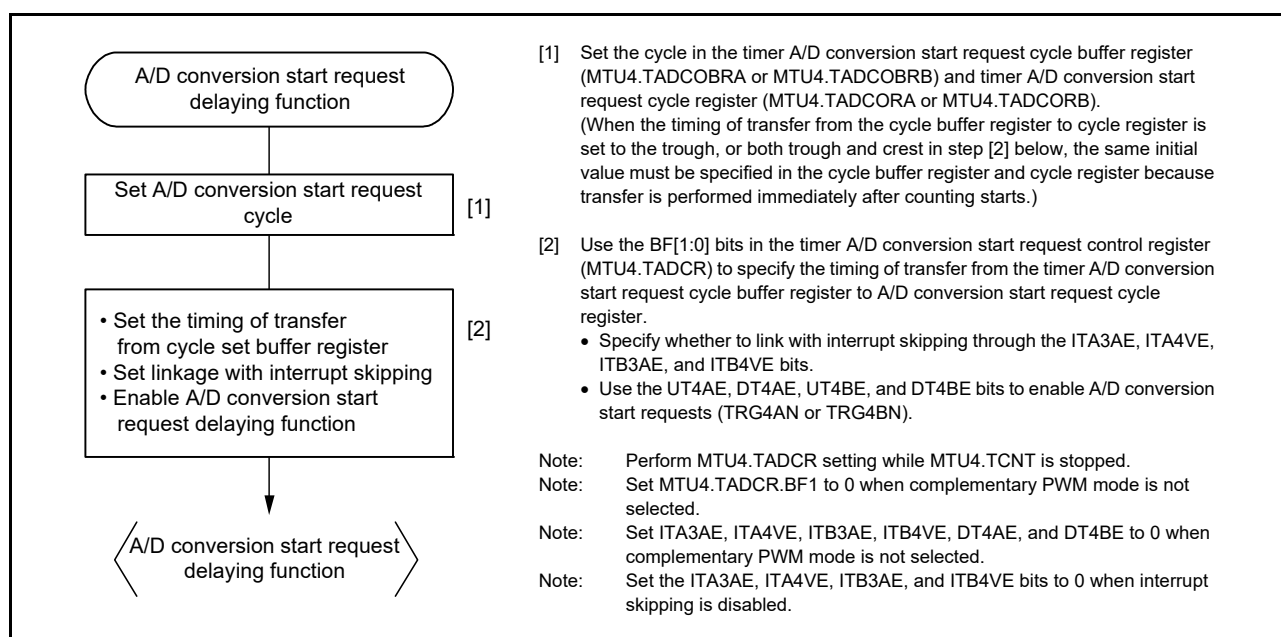


Figure 19.83 Example of Procedure for Specifying A/D Conversion Start Request Delaying Function (MTU3 and MTU4)

(2) Basic Example of A/D Conversion Start Request Delaying Function Operation

Figure 19.84 shows a basic example of A/D conversion start request signal (TRG4AN) operation when the trough of MTU4.TCNT is specified for the buffer transfer timing and an A/D conversion start request is output during MTU4.TCNT down-counting.

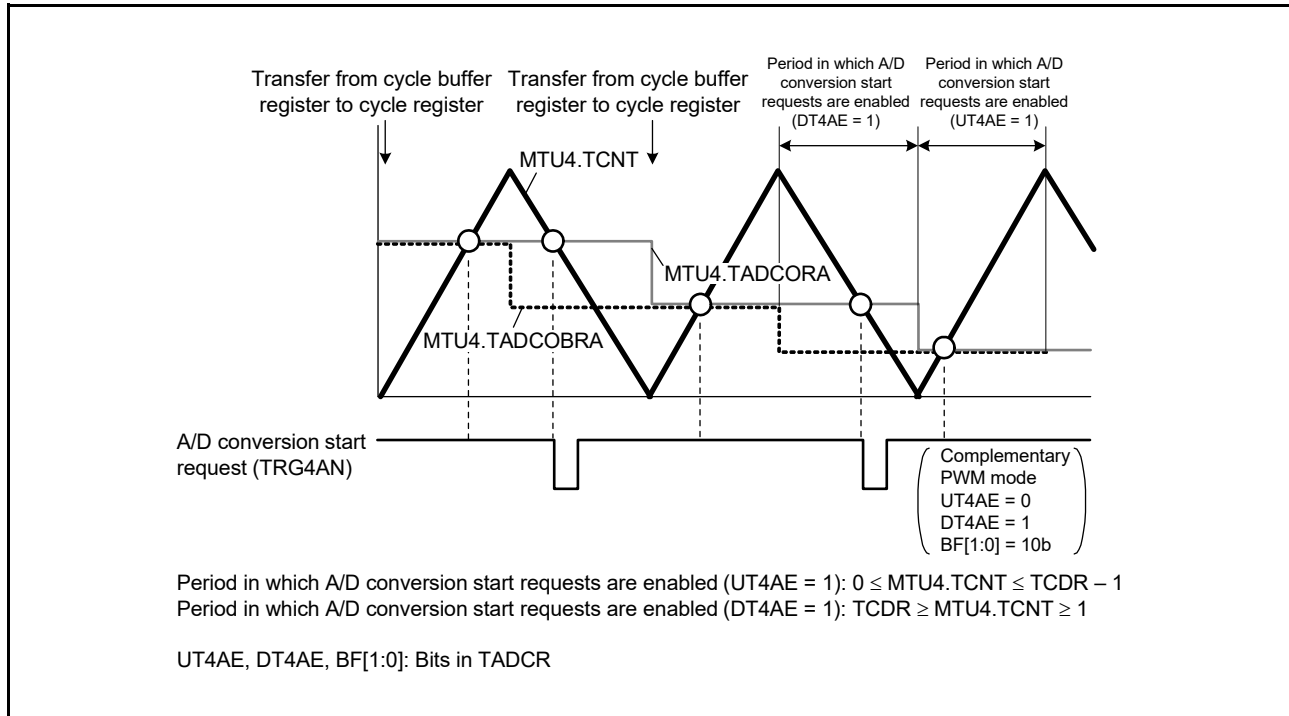


Figure 19.84 Basic Example of A/D Conversion Start Request Signal (TRG4AN) Operation

(3) Period in Which A/D Conversion Start Requests are Enabled

When the MTU4.TCNT counter and the MTU4.TADCORA or MTU4.TADCORB register matches within the period enabled by the UT4AE and UT4BE bits, the corresponding A/D conversion start request (TRG4AN or TRG4BN) is issued.

When the UT4AE and UT4BE bits in the MTU4.TADCR register are set to 1 in complementary PWM mode, A/D conversion start requests are enabled during the MTU4.TCNT up-counting ($0 \leq \text{MTU4.TCNT} \leq \text{TCDR} - 1$). When the DT4AE and DT4BE bits in the MTU4.TADCR register are set to 1, A/D conversion start requests are enabled during MTU4.TCNT down-counting ($\text{TCDR} \geq \text{MTU4.TCNT} \geq 1$). Refer to Figure 19.84.

(4) Buffer Transfer

The data in the timer A/D conversion start request cycle set registers (MTU4.TADCORA and MTU4.TADCORB) is updated by writing data to the timer A/D conversion start request cycle set buffer registers (MTU4.TADCOBRA and MTU4.TADCOBRB). Data is transferred from the buffer registers to the respective cycle set registers at the timing selected with the BF[1:0] bits in the MTU4.TADCR register.

In complementary PWM mode, data is also transferred from the timer A/D conversion start request cycle set buffer registers to the timer A/D conversion start request cycle set registers when MTU4.TGRD register is updated.

There are notes on the timing for transferring data when using buffer transfer in complementary PWM mode.

For details, section 19.6.27, Usage Notes on A/D Conversion Start Request Delaying Function in Complementary PWM Mode.

In modes other than complementary PWM mode, set the BF1 bit in the MTU4.TADCR register to 0.

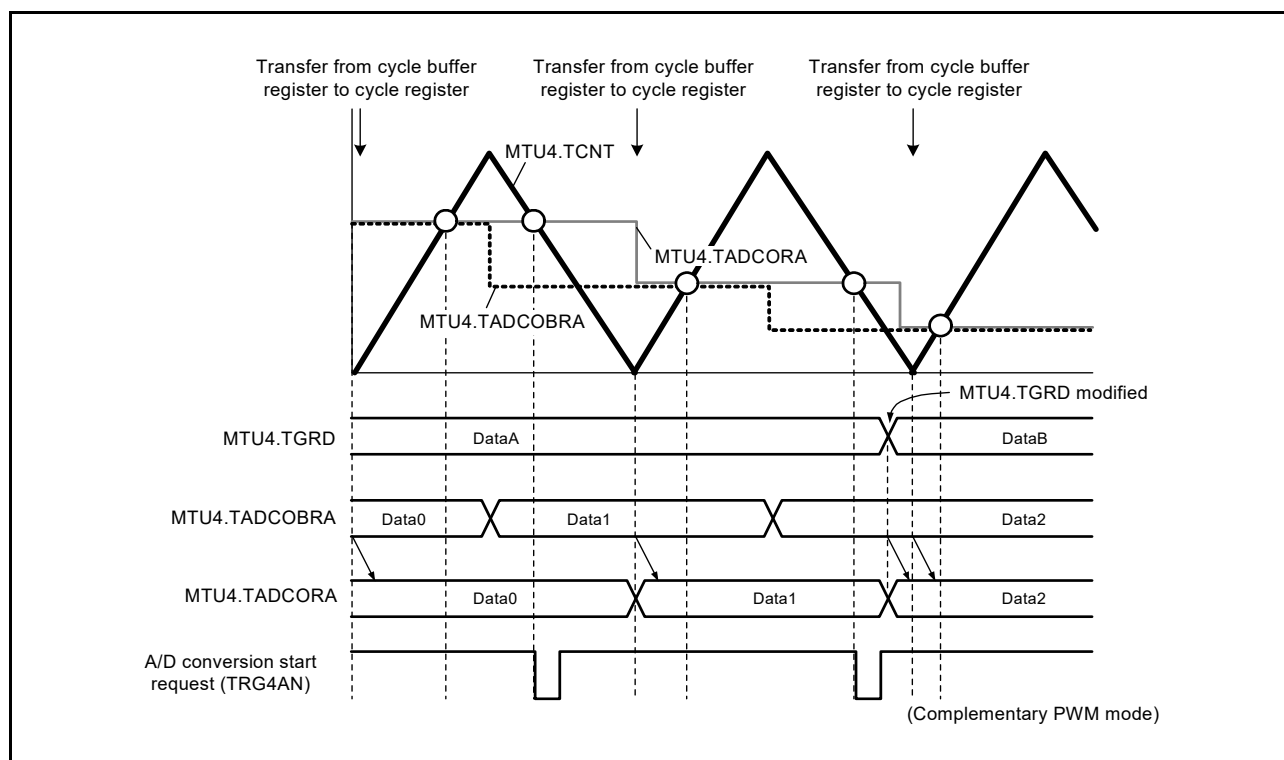


Figure 19.85 Example of A/D Conversion Start Request Signal (TRG4AN) and Buffer Transfer Operation

(5) A/D Conversion Start Request Delaying Function Linked with Interrupt Skipping Function 1

In complementary PWM mode, A/D conversion start requests (TRG4AN and TRG4BN) can be issued in coordination with interrupt skipping 1 by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the MTU4.TADCR register.

Figure 19.86 shows an example of A/D conversion start request signal (TRG4AN) operation when TRG4AN output is enabled during MTU4.TCNT up-counting and down-counting and A/D conversion start requests are linked with interrupt skipping 1.

Figure 19.87 shows another example of A/D conversion start request signal (TRG4AN) operation when TRG4AN output is enabled during MTU4.TCNT up-counting and A/D conversion start requests are linked with interrupt skipping 1.

In modes other than complementary PWM mode, do not use the A/D conversion start request delaying function linked with the interrupt skipping function 1.

Set the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the MTU4.TADCR register to 0.

Note: This function should be used in combination with interrupt skipping 1.

When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR1A) are set to 0 or the skipping count set bits (T3ACOR and T4VCOR) in TITCR1A are set to 0), make sure that A/D conversion start requests are not linked with interrupt skipping 1 (set the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D conversion start request control register (MTU4.TADCR) to 0).

When this function is used, MTU4.TADCORA and MTU4.TADCORB should be set with the value ranging 0002h to the value set in TCDRA minus 2.

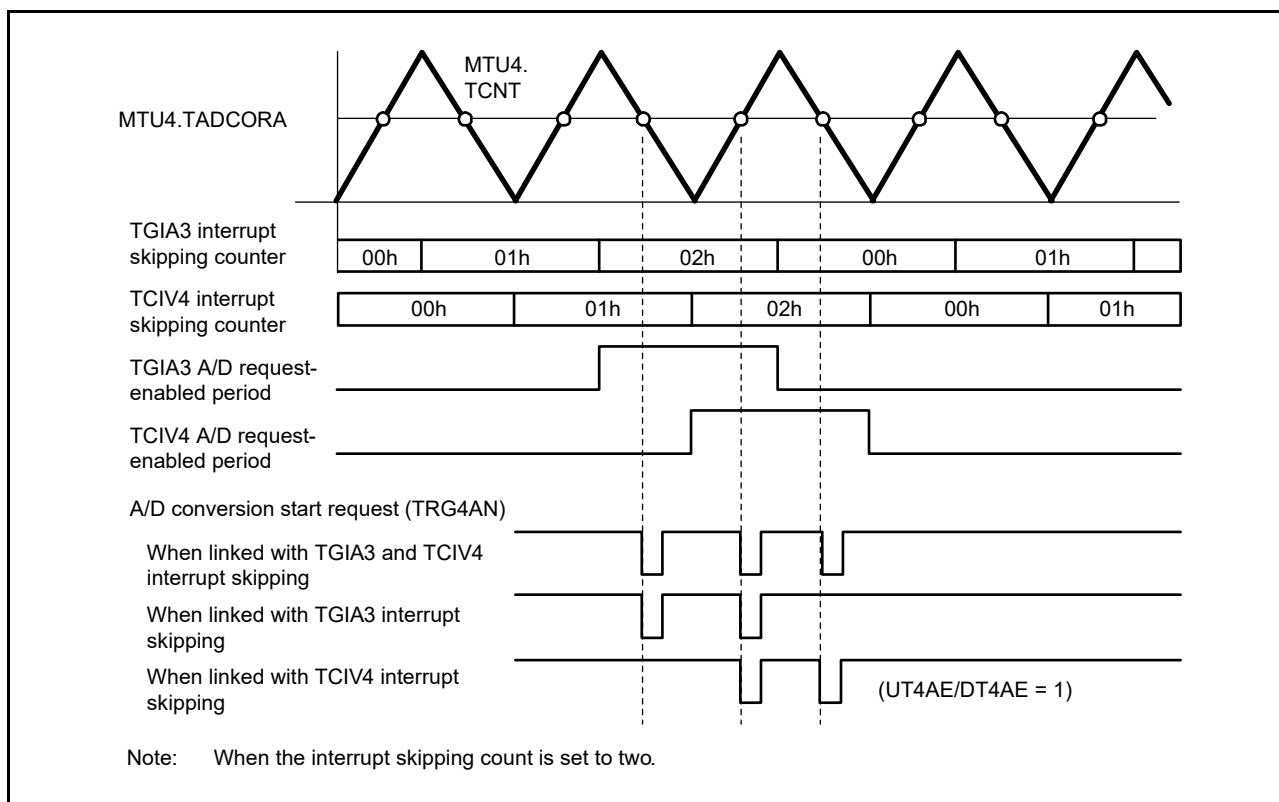


Figure 19.86 Example of A/D Conversion Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE and DT4AE = 1)

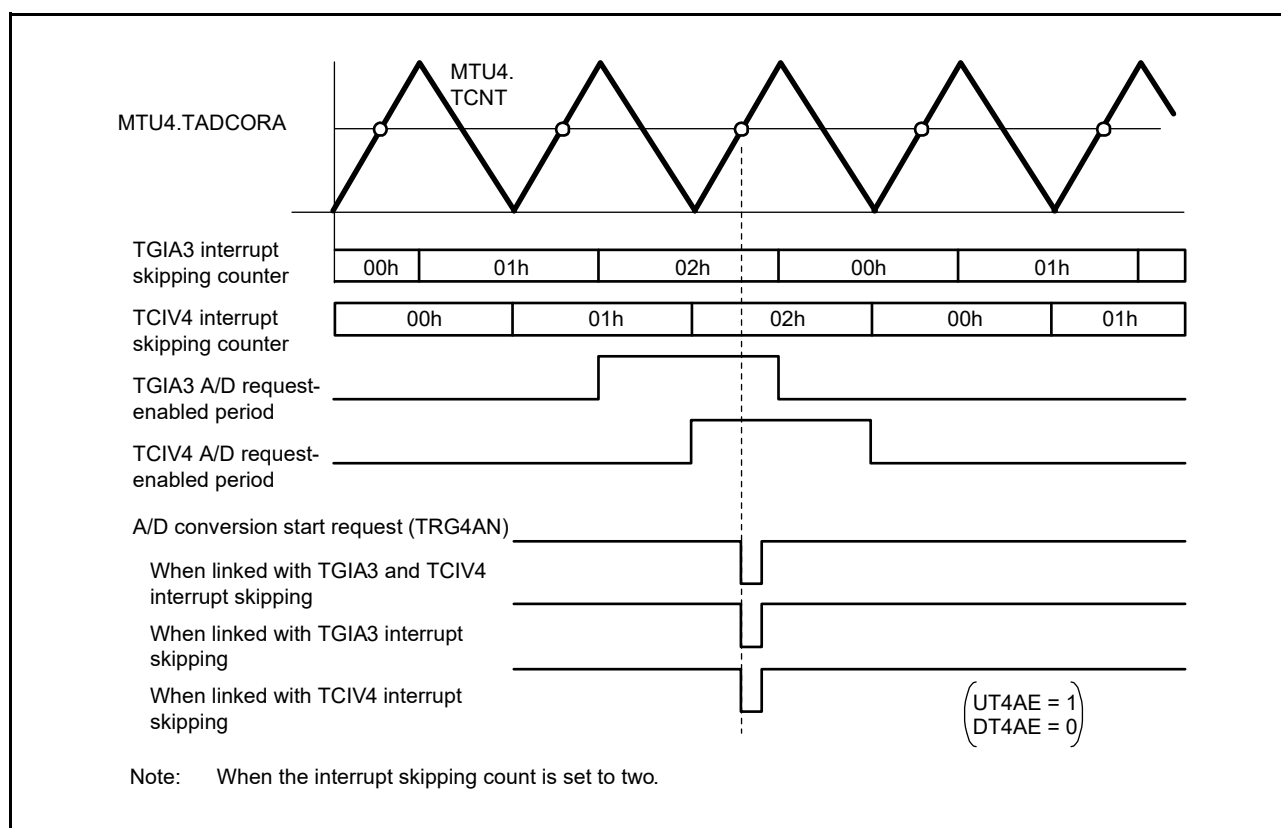


Figure 19.87 Example of A/D Conversion Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE = 1, DT4AE = 0)

(6) A/D Conversion Start Request Delaying Function Linked with Interrupt Skipping Function 2

By setting the TITM bit to 1 in the TITMRA register, the counter starts down-counting from the value (0 to 7) set in the TRG4COR[2:0] bits in TITCR2A register every time an A/D conversion start trigger (TRG4AN or TRG4BN) is generated. When the counter value reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid and an A/D conversion start request signal (TRG4ABN) is output.

This function is valid only when the A/D conversion start request delaying function is enabled.

(a) Example of Procedure for Setting Interrupt Skipping Function 2

Figure 19.88 shows an example of procedure for setting interrupt skipping function 2.

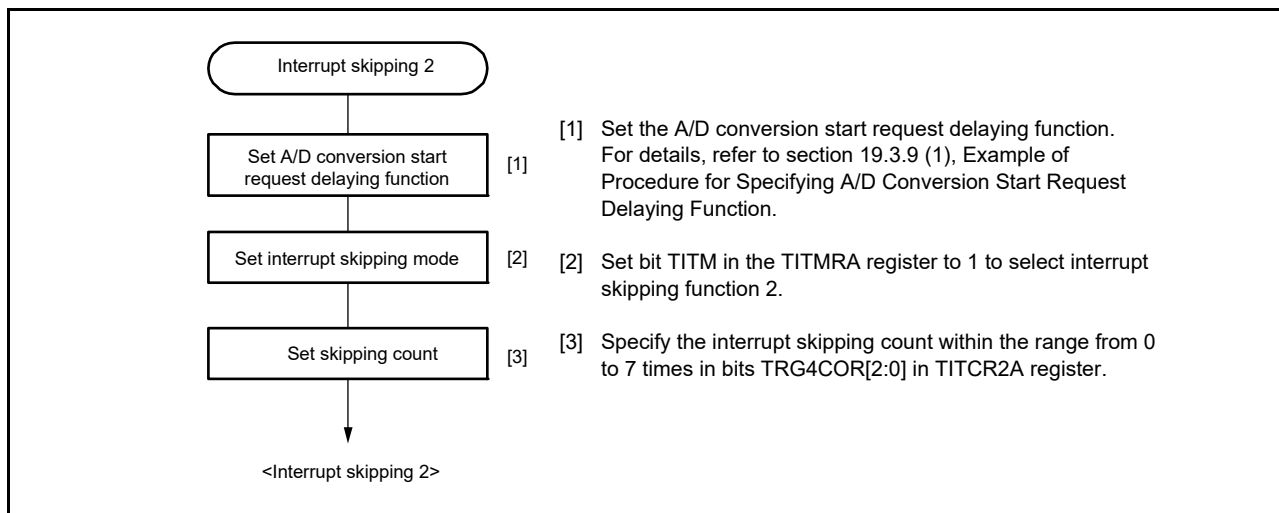


Figure 19.88 Example of Procedure for Setting Interrupt Skipping Function 2

(b) Example of Interrupt Skipping Function 2 Operation

Figure 19.89 shows an example of interrupt skipping function 2 operation.

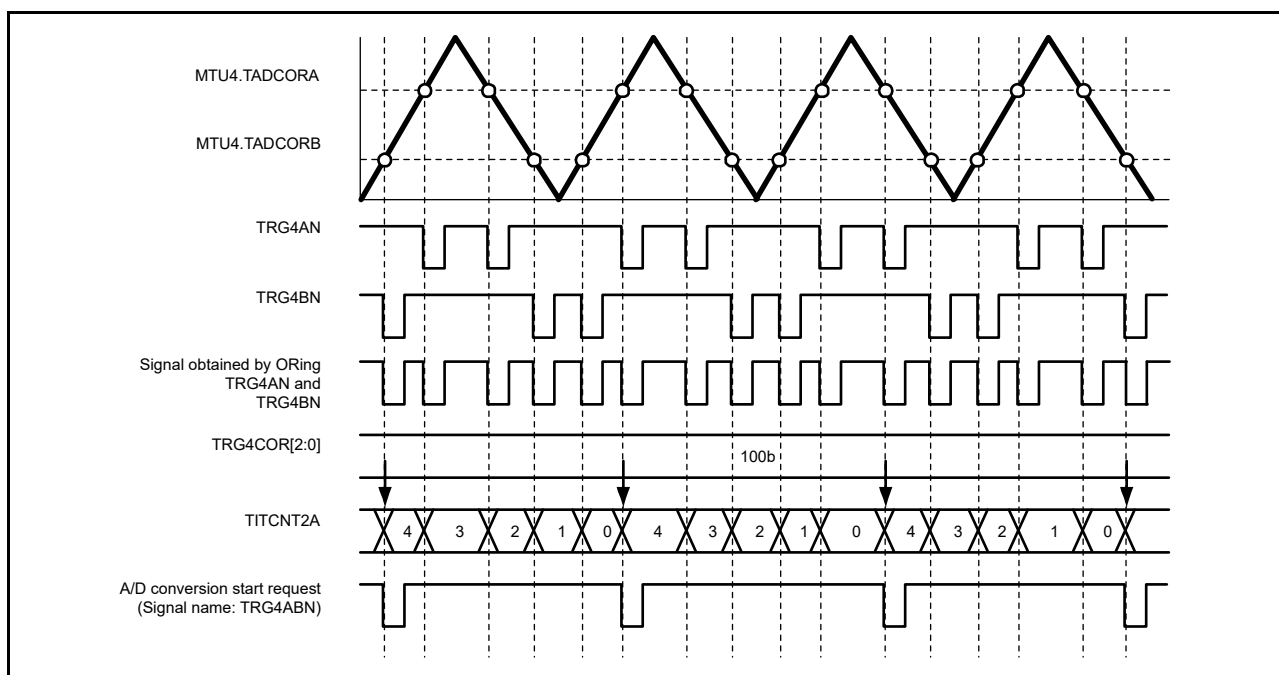


Figure 19.89 Example of Interrupt Skipping Function 2 Operation (Skipping Count is Set to Four)

19.3.10 Synchronous Operation of MTU0 to MTU4

(1) Synchronous Counter Start for MTU0 to MTU4

The counters in MTU0 to MTU4 can be started synchronously by making the TCSYSTR settings.

(a) Example of Procedure for Setting Synchronous Counter Start for MTU0 to MTU4

Figure 19.90 shows an example of procedure for setting synchronous counter start for MTU0 to MTU4.

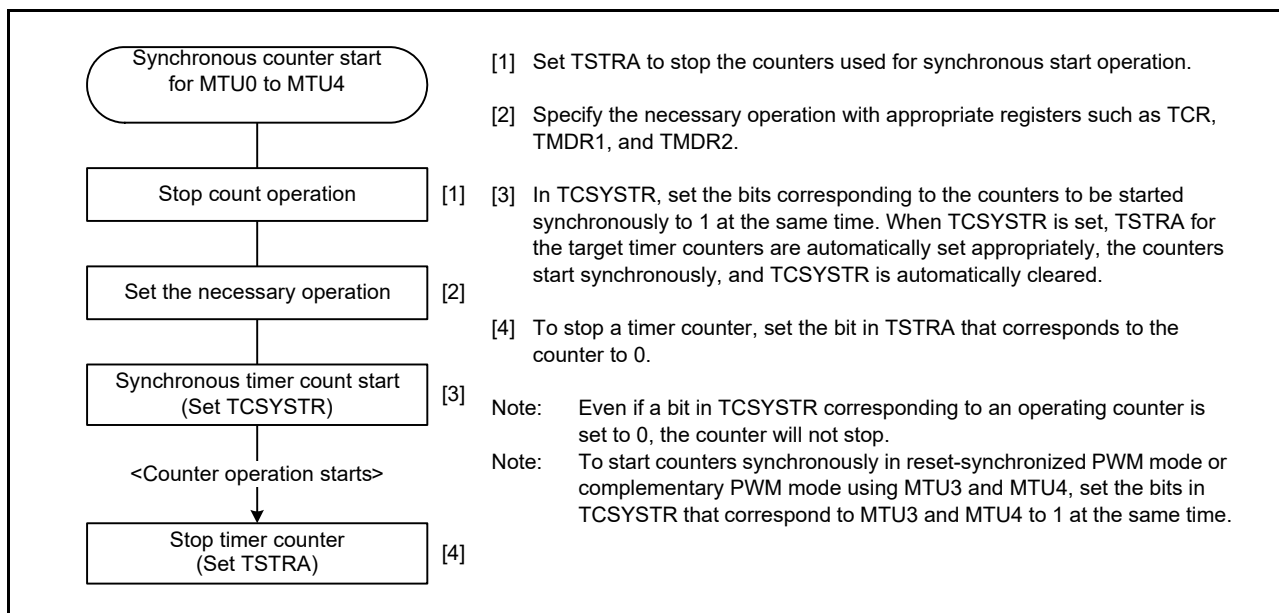


Figure 19.90 Example of Procedure for Setting Synchronous Counter Start for MTU0 to MTU4

(b) Examples of Synchronous Counter Start Operation

Figure 19.91 shows an examples of synchronous counter start operation for MTU0 to MTU4.

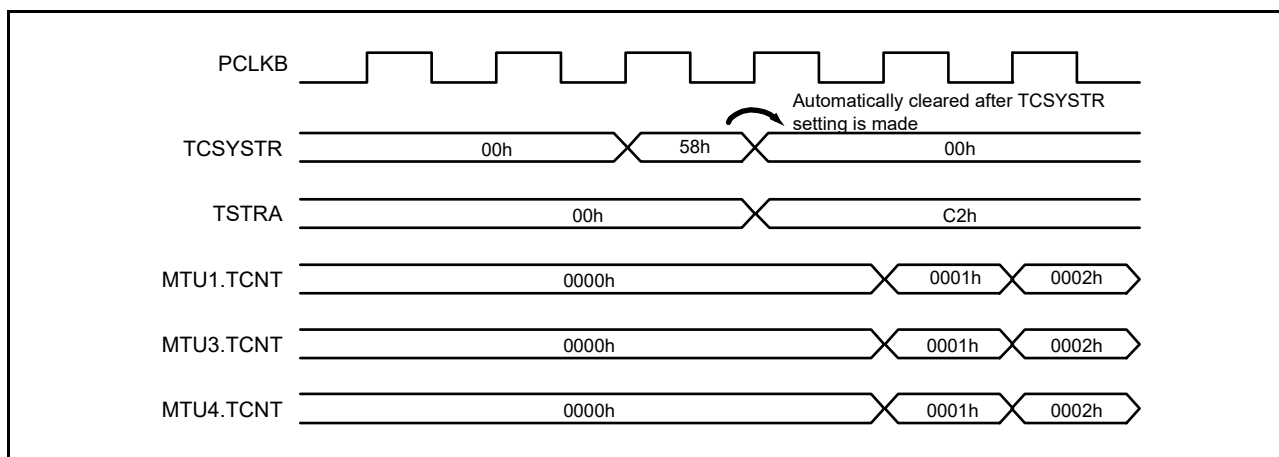


Figure 19.91 Examples of Synchronous Counter Start Operation for MTU0 to MTU4

19.3.11 External Pulse Width Measurement

The pulse widths of up to three external input lines can be measured in MTU5.

When the IOC[4:0] bits in MTU5.TIORU, MTU5.TIORV, MTU5.TIORW are set for pulse width measurement, the pulse width of the signal input to the MTIC5U, MTIC5V, and MTIC5W pins are measured. TCNTU, TCNTV, and TCNTW count up while the level specified by the IOC[4:0] bits is input.

Figure 19.92 shows an example of setting external pulse width measurement, and Figure 19.93 an example of external pulse width measurement.

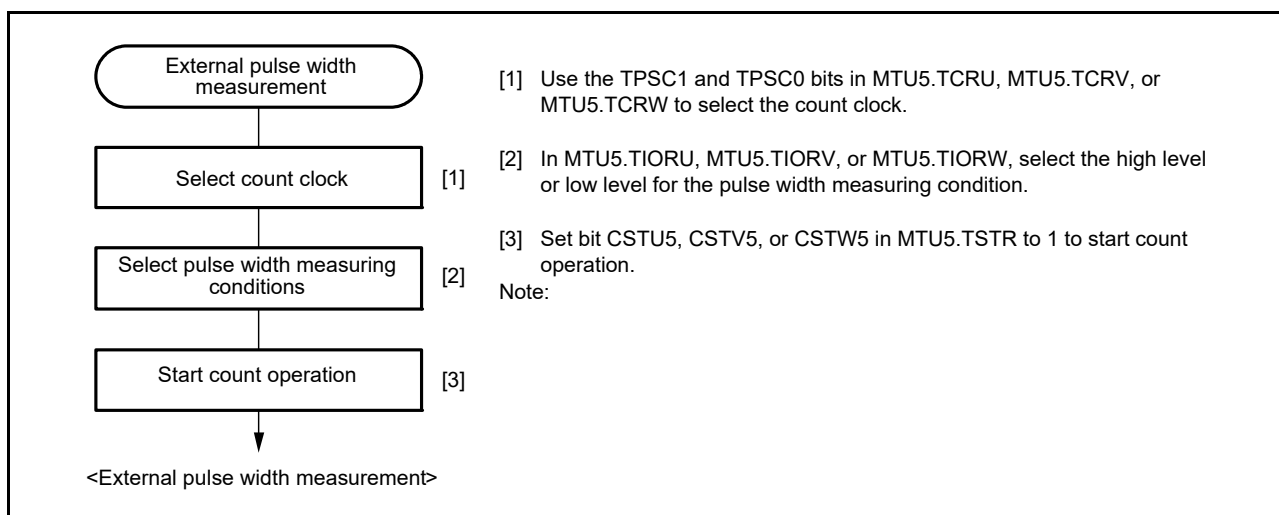


Figure 19.92 Example of External Pulse Width Measurement Setting Procedure

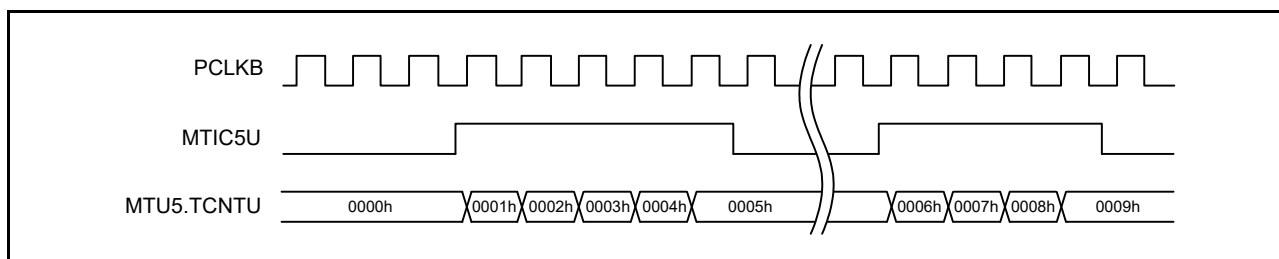


Figure 19.93 Example of External Pulse Width Measurement (Measuring High Pulse Width)

19.3.12 Dead Time Compensation

A dead time delay (a propagation delay of the inverter output from the complementary PWM output) can be compensated by combining MTU5 with MTU3 and MTU4. Figure 19.94 shows an example of the motor control circuit compensating a dead time delay by combining MTU5 with MTU3 and MTU4. A dead time for the PWM output waveform during complementary PWM operation using MTU3 and MTU4 can be compensated by adjusting a duty ratio set in a compare register for the PWM output after measuring a delay of the inverter output from the complementary PWM output by an external pulse measurement function for MTU5 (Figure 19.95). Figure 19.96 shows the procedure for setting dead time compensation using MTU3 to MTU5. For details on MTU5 operation at this time, refer to section 19.3.13, TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode.

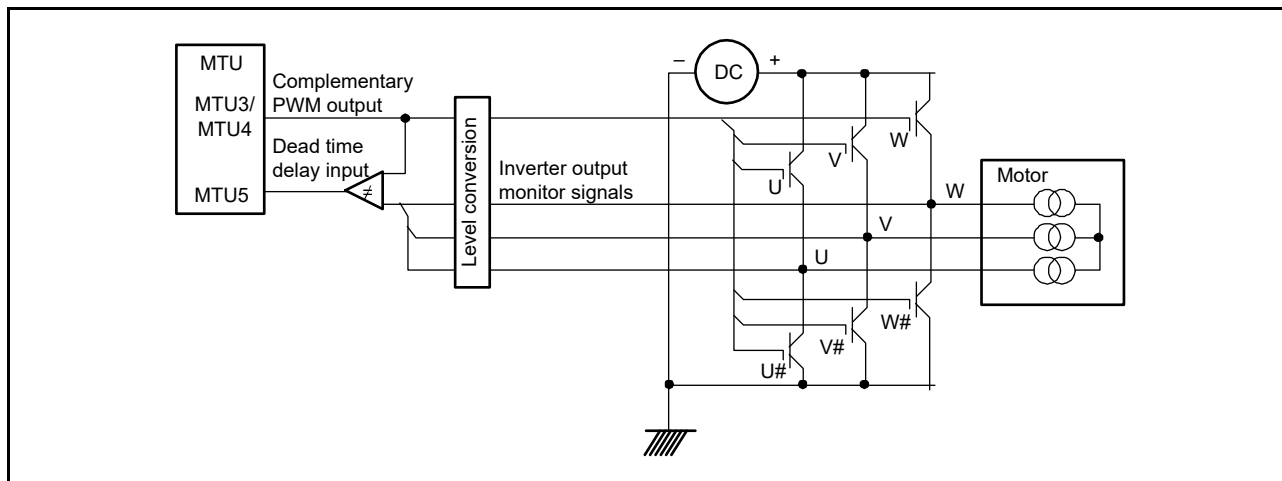


Figure 19.94 Motor Control Circuit Example

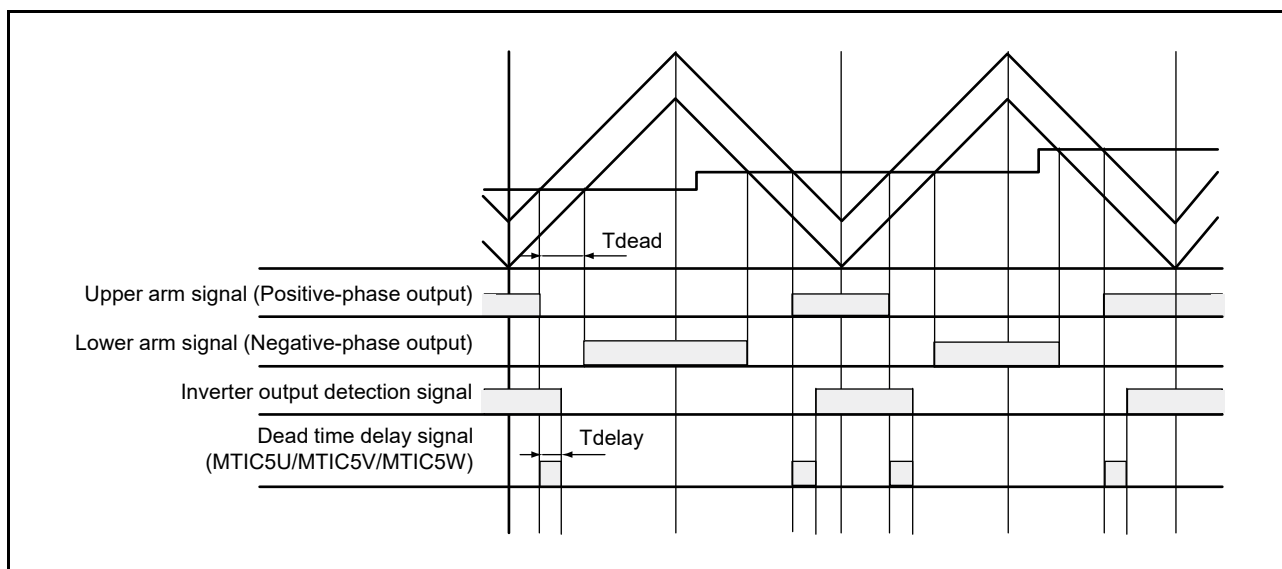


Figure 19.95 Delay in Dead Time in Complementary PWM Operation

(1) Example of Dead Time Compensation Setting Procedure

Figure 19.96 shows an example of dead time compensation setting procedure by using three counters in MTU5.

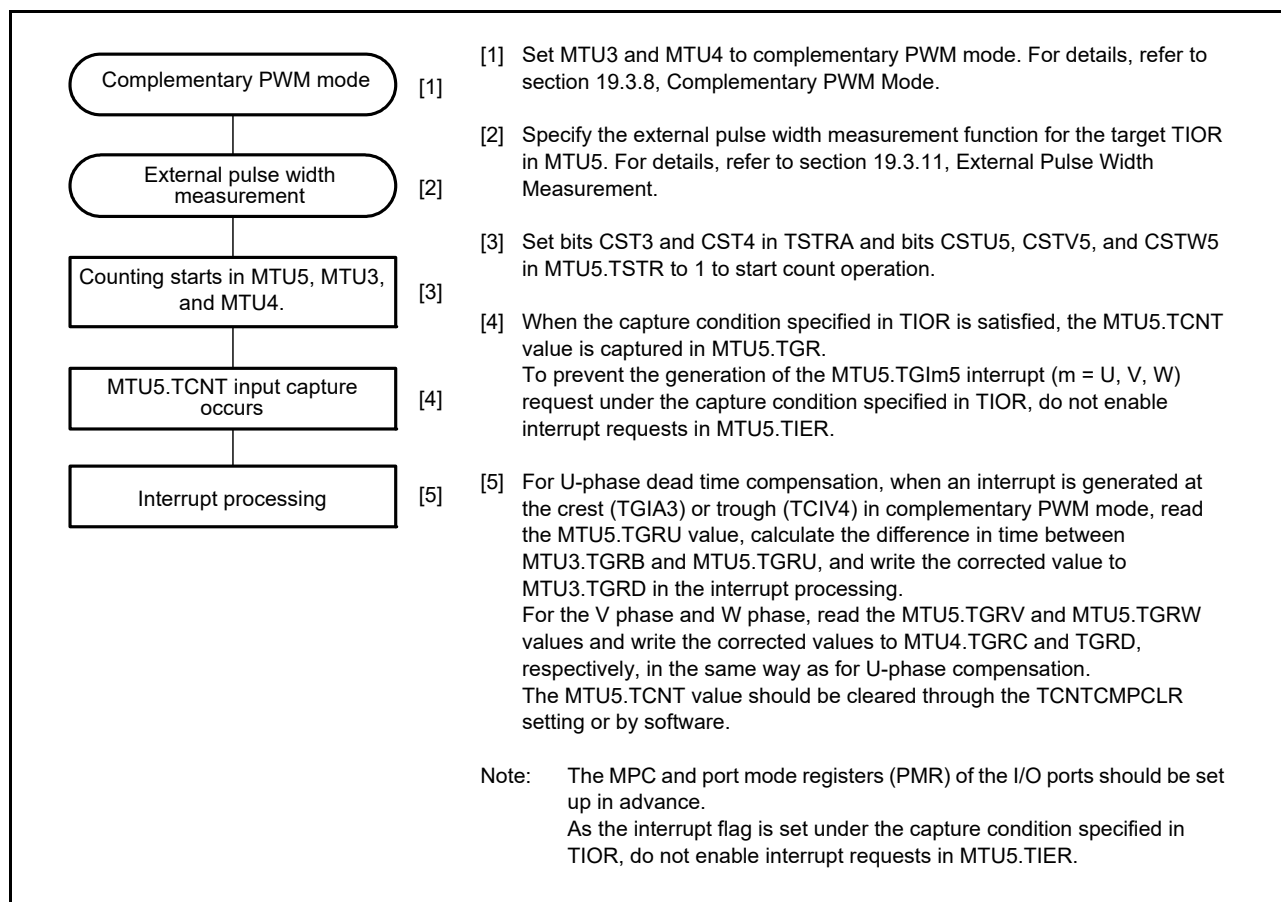


Figure 19.96 Example of Dead Time Compensation Setting Procedure

19.3.13 TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode

The MTU5 external pulse width measurement function allows to transfer the value in TCNTU, TCNTV, and TCNTW to TGRU, TGRV, and TGRW at the crest, or trough, or crest and trough when MTU3 and MTU4 operate in complementary PWM mode. The transfer timing is set in TIORU, TIORV, and TIORW. When the CMPCLR5U, CMPCLR5V, and CMPCLR5W bits in the TCNTCMPCLR register are set to 1, TCNTU, TCNTV, and TCNTW become 0000h at the transfer timing for TGRU, TGRV, and TGRW.

Figure 19.97 shows an operation example in which TCNTU is used as a free-running counter without being cleared, and the value is captured in TGRU at the crest and trough in complementary PWM mode.

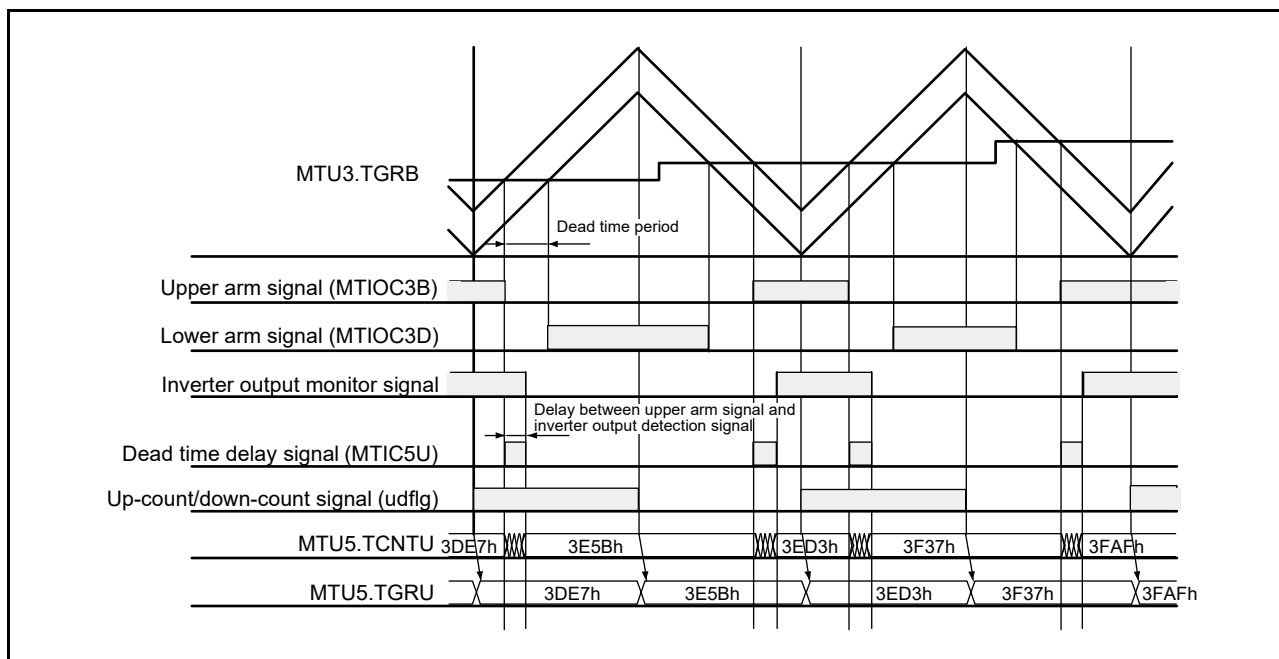


Figure 19.97 TCNTU Capture at Crest and Trough in Complementary PWM Operation

19.3.14 Noise Filter Function

The input capture input pins and external pulse input pins have a noise filter function.

When the noise filter function is enabled, input pulses that are less than three sampling clock periods can be removed.

The noise filter for each pin can be enabled or disabled individually, and the sampling clock can be set for each channel.

Figure 19.98 shows the timing of noise filtering.

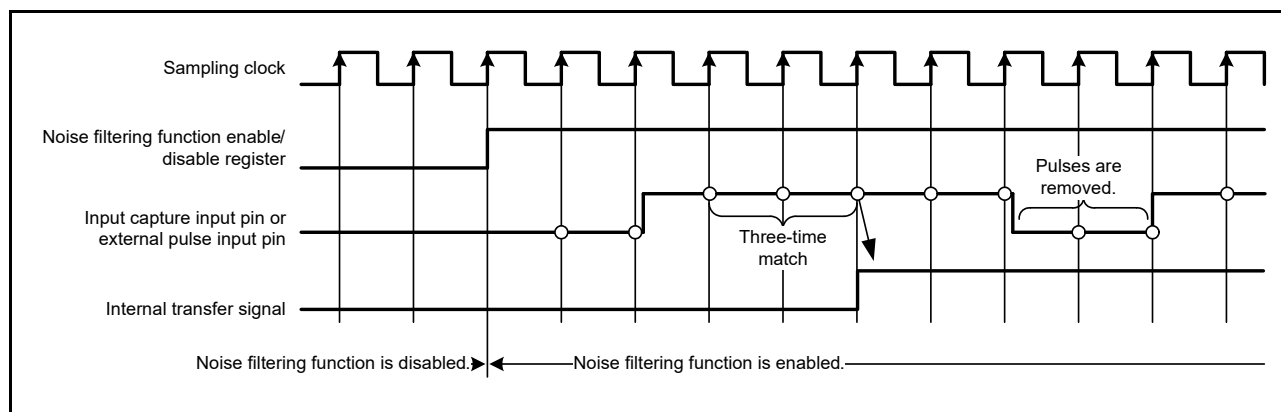


Figure 19.98 Timing of Noise Filtering

Note that when the noise filter function is enabled, the input signal is delayed at least by two sampling clock cycles plus one PCLKB cycle.

19.3.15 A/D Conversion Start Request Frame Synchronization Signal

This function can be used to monitor the generation timing of the A/D conversion start request signal using an external pin.

When the A/D conversion start request signal to be monitored is selected by the TADSTRGRn register ($n = 0, 1$) and the ADSTMn pin output is enabled by TADSTRGRn.TADSMENn, a pulse signal is output from the ADSTMn pin that is at the high level when the A/D conversion start request signal is generated, and at the low level in the timer cycle used to generate the A/D conversion start request signal.

Figure 19.99 shows an example of outputting the A/D conversion start request frame synchronization signal.

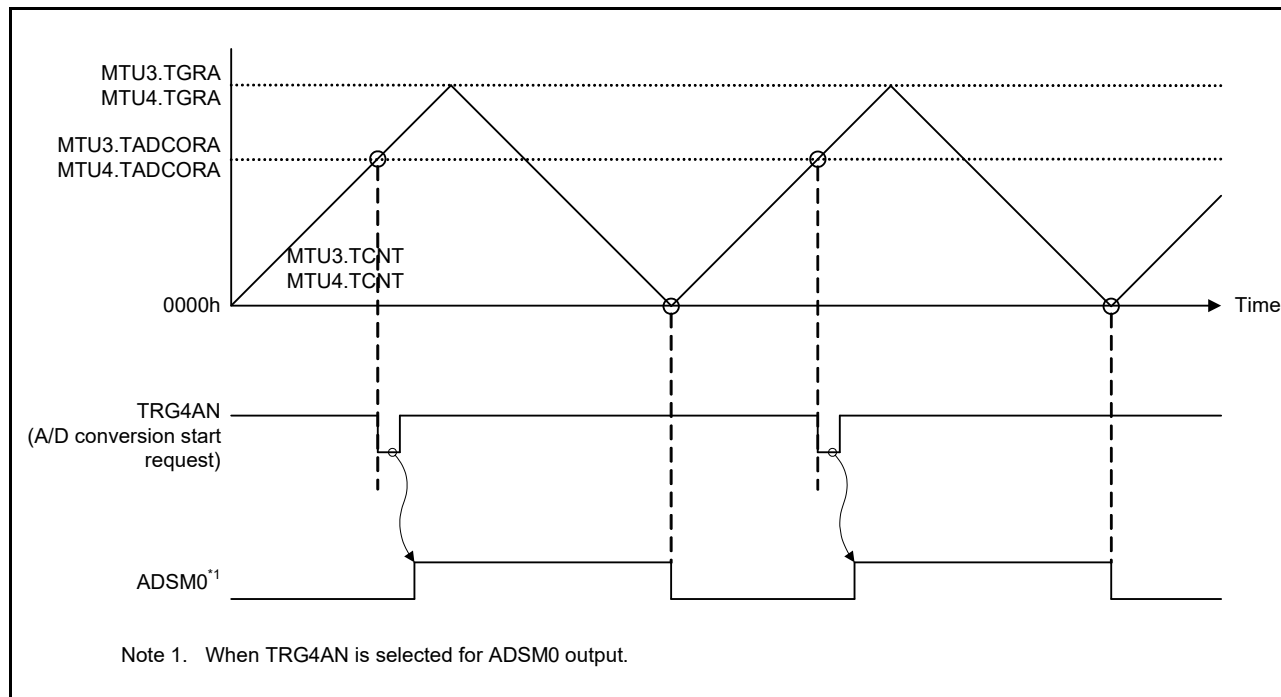


Figure 19.99 Example of Outputting A/D Conversion Start Request Frame Synchronization Signal

19.4 Interrupt Sources

19.4.1 Interrupt Sources and Priorities

There are three kinds of interrupt source; TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own enable/disable bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt source is generated, if the corresponding enable/disable bit in TIER is set to 1, an interrupt is requested.

Relative channel priorities can be changed by the interrupt controller; however the priority within a channel is fixed. For details, refer to section 14, Interrupt Controller (ICUb). Table 19.62 lists the MTU interrupt sources.

Table 19.62 MTU Interrupt Sources

Channel	Name	Interrupt Source	DTC Activation	Priority
MTU0	TGIA0	MTU0.TGRA input capture/compare match	Possible	High ↑
	TGIB0	MTU0.TGRB input capture/compare match	Possible	
	TGIC0	MTU0.TGRC input capture/compare match	Possible	
	TGID0	MTU0.TGRD input capture/compare match	Possible	
	TCIV0	MTU0.TCNT overflow	Not possible	
	TGIE0	MTU0.TGRE compare match	Not possible	
	TGIF0	MTU0.TGRF compare match	Not possible	
MTU1	TGIA1	MTU1.TGRA input capture/compare match	Possible	↑
	TGIB1	MTU1.TGRB input capture/compare match	Possible	
	TCIV1	MTU1.TCNT overflow	Not possible	
	TCIU1	MTU1.TCNT underflow	Not possible	
MTU2	TGIA2	MTU2.TGRA input capture/compare match	Possible	
	TGIB2	MTU2.TGRB input capture/compare match	Possible	
	TCIV2	MTU2.TCNT overflow	Not possible	
	TCIU2	MTU2.TCNT underflow	Not possible	
MTU3	TGIA3	MTU3.TGRA input capture/compare match	Possible	
	TGIB3	MTU3.TGRB input capture/compare match	Possible	
	TGIC3	MTU3.TGRC input capture/compare match	Possible	
	TGID3	MTU3.TGRD input capture/compare match	Possible	
	TCIV3	MTU3.TCNT overflow	Not possible	
MTU4	TGIA4	MTU4.TGRA input capture/compare match	Possible	
	TGIB4	MTU4.TGRB input capture/compare match	Possible	
	TGIC4	MTU4.TGRC input capture/compare match	Possible	
	TGID4	MTU4.TGRD input capture/compare match	Possible	
	TCIV4	MTU4.TCNT overflow/underflow*1	Possible	
MTU5	TGIU5	MTU5.TGRU input capture/compare match	Possible	
	TGIV5	MTU5.TGRV input capture/compare match	Possible	
	TGIW5	MTU5.TGRW input capture/compare match	Possible	Low

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

Note 1. Underflow is available only in complementary PWM mode.

(1) Input Capture/Compare Match Interrupt

If the TIER.TGIE bit is set to 1 when a TGR input capture/compare match occurs on a channel, an interrupt is requested. The MTU has 21 input capture/compare match interrupts (six for MTU0, four each for MTU3 and MTU4, two each for MTU1 and MTU2, and three for MTU5).

(2) Overflow Interrupt

If the TIER.TCIEV bit is set to 1 when a TCNT overflow occurs on a channel, an interrupt is requested. The MTU has five overflow interrupts (one for each channel except MTU5).

Note that an overflow interrupt is generated also when an underflow of the MTU4.TCNT occurs while operating in complementary PWM mode.

(3) Underflow Interrupt

If the TIER.TCIEU bit is set to 1 when a TCNT underflow occurs on a channel, an interrupt is requested. The MTU has two underflow interrupts (one each for MTU1, and MTU2).

19.4.2 DTC Trigger Sources

(1) DTC Trigger Sources

The DTC can be triggered by the TGR input capture/compare match interrupt in each channel or the overflow interrupt in MTU4. For details, refer to [section 16, Data Transfer Controller \(DTCb\)](#).

The MTU provides a total of 20 input capture/compare match interrupts and overflow interrupts that can be used as DTC trigger sources: four each for MTU0 and MTU3, two each for MTU1 and MTU2, five for MTU4, and three for MTU5.

19.4.3 A/D Converter Trigger Sources

The A/D converter can be triggered by one of the following three methods in the MTU. Table 19.63 shows the relationship between interrupt sources and A/D conversion start request signals.

(1) A/D Conversion Start by TGRA Input Capture/Compare Match or at Trough of MTU4.TCNT in Complementary PWM Mode

The A/D converter can be triggered by the occurrence of a TGRA input capture/compare match in each channel. In addition, if complementary PWM operation is performed while the TTGE2 bit in MTU4.TIER is set to 1, the A/D converter can be triggered at the trough of MTU4.TCNT count (MTU4.TCNT = 0000h).

A/D conversion start request TRGAnN is issued to the A/D converter under either of the following conditions (n = 0 to 4).

- When a TGRA input capture/compare match occurs on a channel while the TIER.TTGE bit is set to 1
- When the MTU4.TCNT count reaches the trough (MTU4.TCNT = 0000h) during complementary PWM operation while the TTGE2 bit in MTU4.TIER is set to 1

When either condition is satisfied, if A/D conversion start request signal TRGAnN from the MTU is selected as the trigger in the A/D converter, A/D conversion will start.

(2) A/D Conversion Start by Compare Match between MTU0.TCNT and MTU0.TGRE

A/D conversion start request TRG0N is issued to the A/D converter when a compare match occurs between MTU0.TCNT and MTU0.TGRE.

When a compare match occurs between MTU0.TCNT and MTU0.TGRE while the TTGE2 bit in MTU0.TIER2 is set to 1, A/D conversion start request TRG0N is issued to the A/D converter. If A/D conversion start request signal TRG0N from the MTU is selected as the trigger in the A/D converter, A/D conversion will start.

(3) A/D Conversion Start by A/D Conversion Start Request Delaying Function

The A/D converter can be triggered by generating A/D conversion start request signal TRG4AN or TRG4BN when the MTU4.TCNT count matches the MTU4.TADCORA or MTU4.TADCORB value if the UT4AE, DT4AE, UT4BE, or DT4BE bit in the A/D conversion start request control register (MTU4.TADCR) is set to 1. For details, refer to section 19.3.9, A/D Conversion Start Request Delaying Function.

A/D conversion will start when TRG4AN is generated if A/D conversion start request signal TRG4AN from the MTU is selected as the trigger in the A/D converter, when TRG4BN is generated if TRG4BN from the MTU is selected as the trigger in the A/D converter, or when TRG4ABN is generated if TRG4ABN from the MTU is selected as the trigger in the A/D converter.

Table 19.63 Interrupt Sources and A/D Conversion Start Request Signals

Target Registers	Interrupt Source	A/D Conversion Start Request Signal
MTU0.TGRA and MTU0.TCNT	Input capture/compare match	TRGA0N
MTU1.TGRA and MTU1.TCNT		TRGA1N
MTU2.TGRA and MTU2.TCNT		TRGA2N
MTU3.TGRA and MTU3.TCNT		TRGA3N
MTU4.TGRA and MTU4.TCNT*1		TRGA4N
MTU4.TCNT	MTU4.TCNT trough in complementary PWM mode	
MTU0.TGRE and MTU0.TCNT	Compare match	TRG0N
MTU4.TADCORA and MTU4.TCNT		TRG4AN
MTU4.TADCORB and MTU4.TCNT		TRG4BN
MTU4.TADCORA and MTU4.TCNT, MTU4.TADCORB and MTU4.TCNT	Compare match (interrupt skipping function 2)	TRG4ABN

Note 1. Since PWM waveforms are generated in complementary PWM mode, MTU4.TGRA compare match not only with MTU4.TCNT but also with MTU3.TCNT and TCNTSA is detected. Accordingly, when compare match with MTU3.TCNT and TCNTSA occurs, TRGA4N is also generated.

When MTU3 and MTU 4 are made to operate in complementary PWM mode for generating an A/D conversion start request, use the A/D conversion start request by compare match between MTU4.TCNT and MTU4.TADCORA/TADCORB.

19.5 Operation Timing

19.5.1 Input/Output Timing

(1) TCNT Count Timing

Figure 19.100 and Figure 19.101 show the TCNT count timing in internal clock operation, Figure 19.102 shows the TCNT count timing in external clock operation (normal mode), and Figure 19.103 shows the TCNT count timing in external clock operation (phase counting mode).

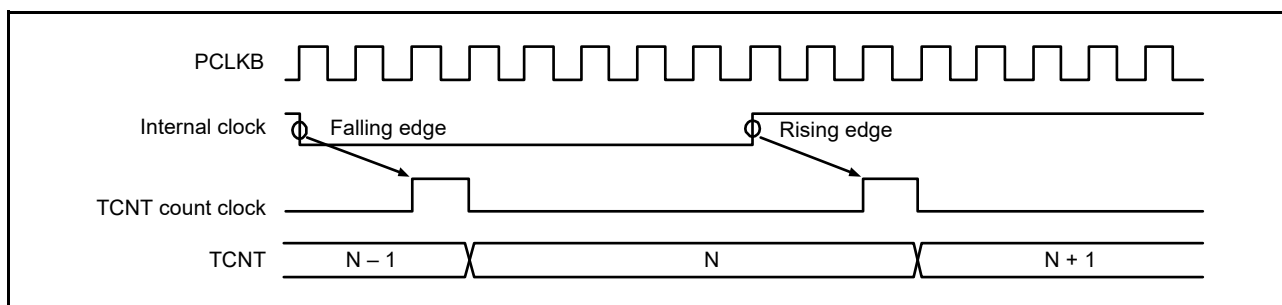


Figure 19.100 Count Timing in Internal Clock Operation (MTU0 to MTU4)

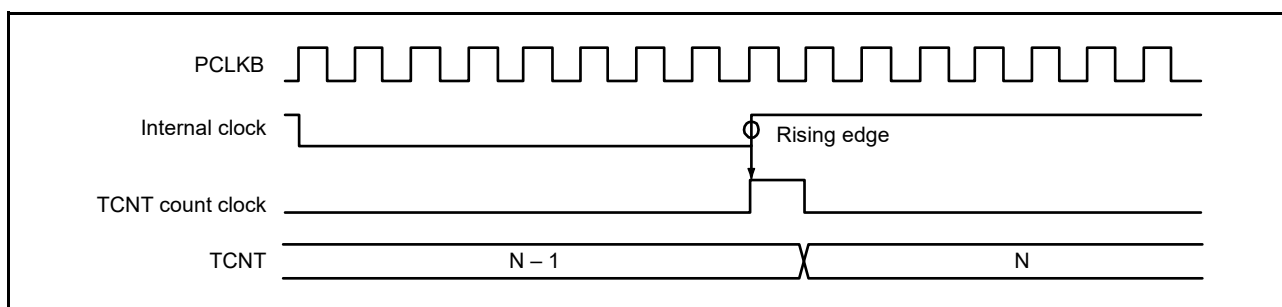


Figure 19.101 Count Timing in Internal Clock Operation (MTU5)

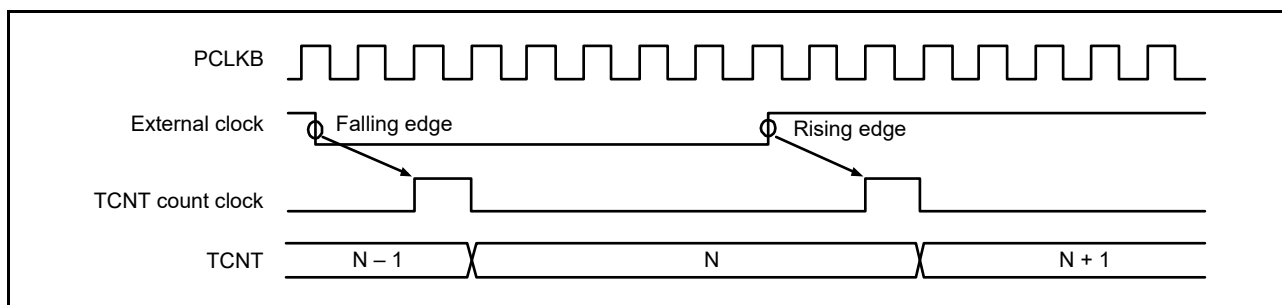


Figure 19.102 Count Timing in External Clock Operation (MTU0 to MTU4)

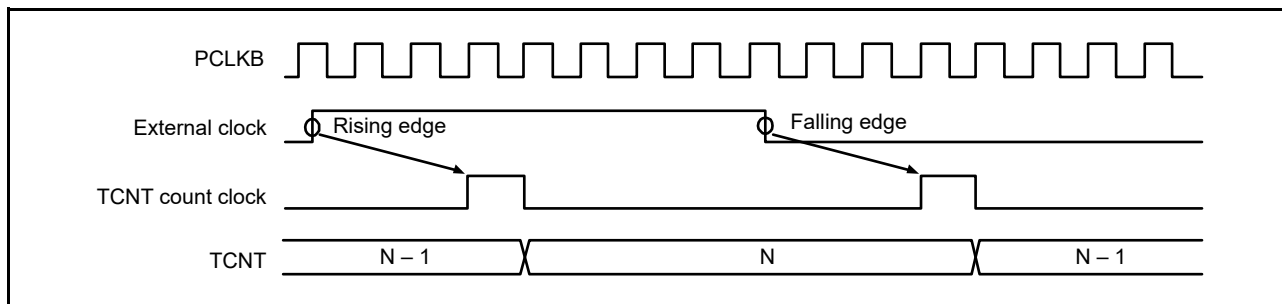


Figure 19.103 Count Timing in External Clock Operation (Phase Counting Mode)

(2) Output Compare Output Timing

A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the value set in TIOR is output from MTIOCNm pin ($n = 0$ to 4 ; $m = A$ to D). After a match between TCNT and TGR, the compare match signal is not generated until the TCNT count clock is generated.

Figure 19.104 shows the output compare output timing (normal mode or PWM mode) and Figure 19.105 shows the output compare output timing (complementary PWM mode or reset-synchronized PWM mode).

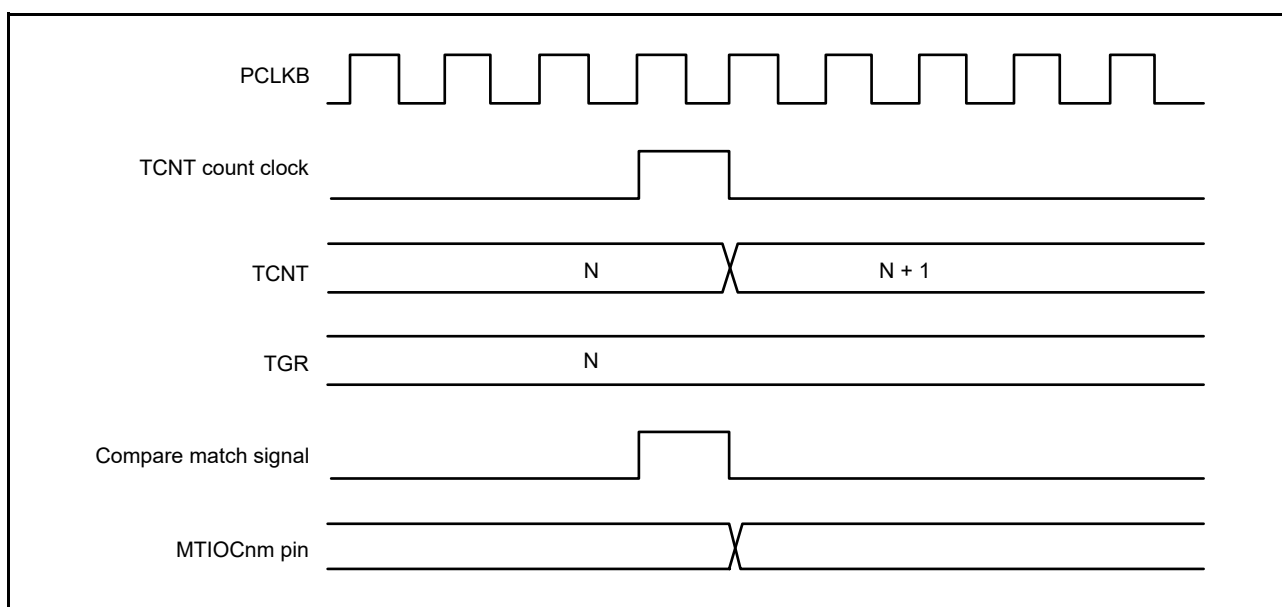


Figure 19.104 Output Compare Output Timing (Normal Mode or PWM Mode) ($n = 0$ to 4 ; $m = A$ to D)

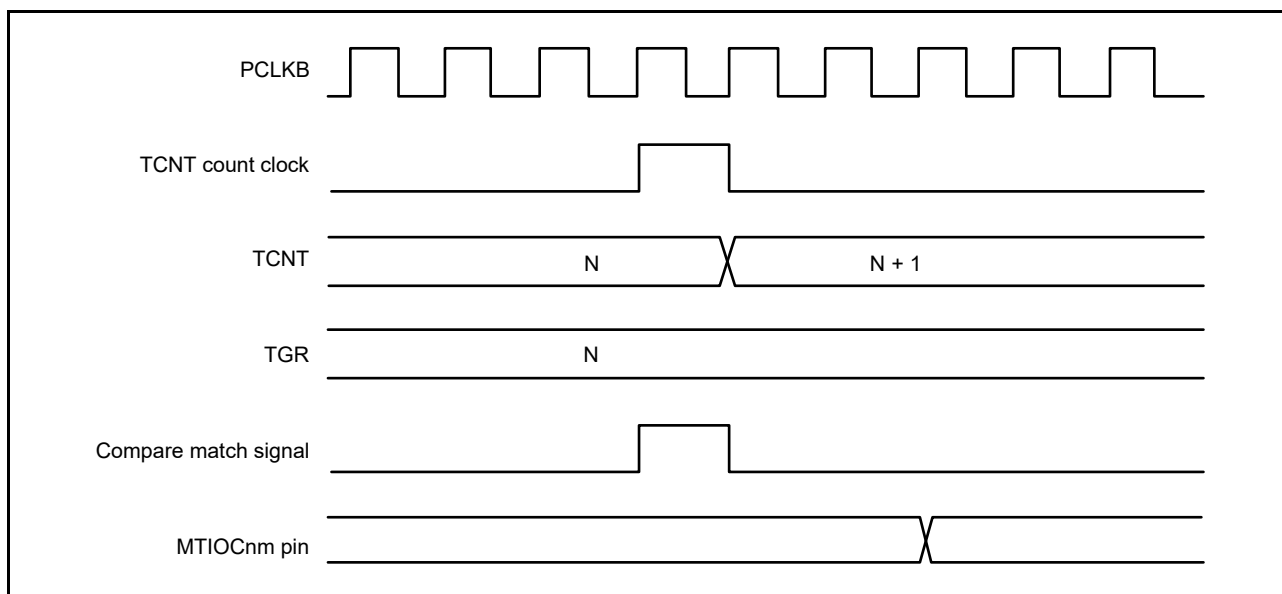


Figure 19.105 Output Compare Output Timing (Complementary PWM Mode or Reset-Synchronized PWM Mode) ($n = 0$ to 4 ; $m = A$ to D)

(3) Input Capture Signal Timing

Figure 19.106 shows the input capture signal timing.

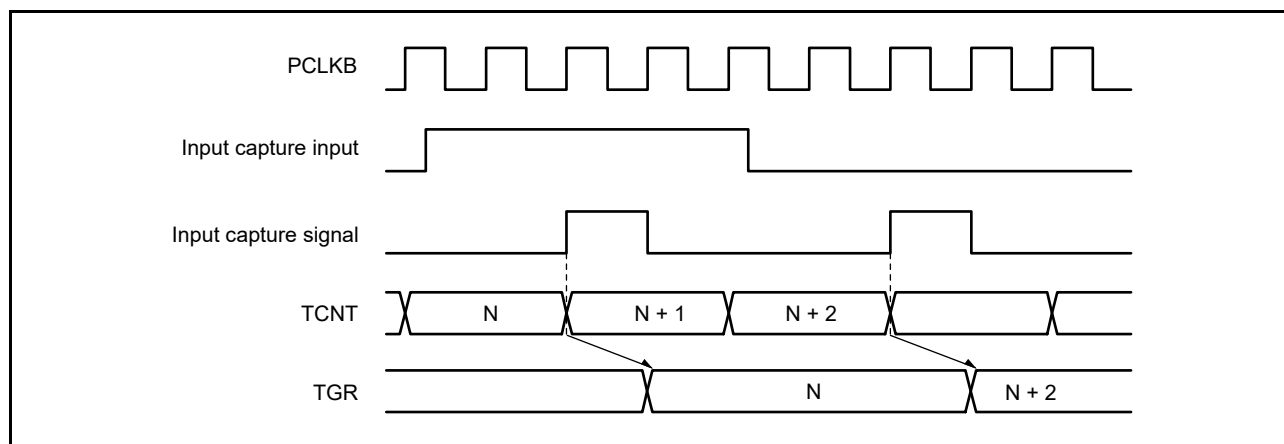


Figure 19.106 Input Capture Input Signal Timing

(4) Timing for Counter Clearing by Compare Match/Input Capture

Figure 19.107 and Figure 19.108 show the timing when counter clearing on compare match is specified, and Figure 19.109 shows the timing when counter clearing on input capture is specified.

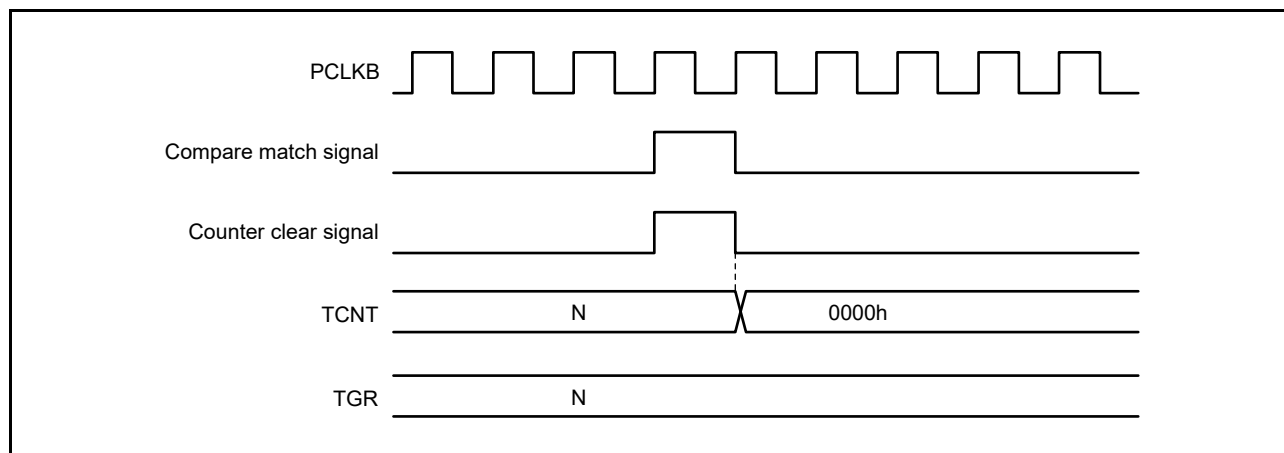


Figure 19.107 Counter Clear Timing (Compare Match) (MTU0 to MTU4)

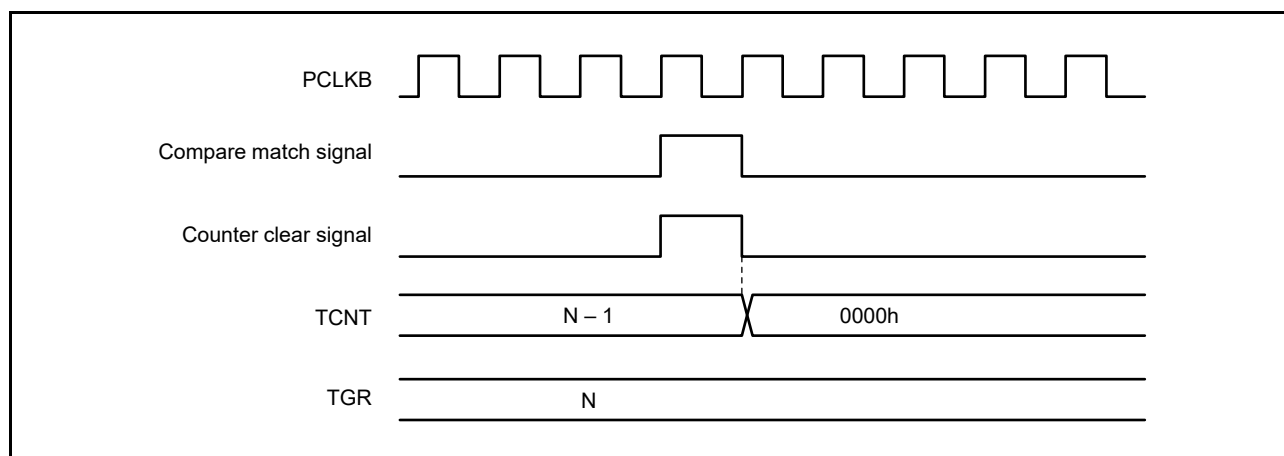


Figure 19.108 Counter Clear Timing (Compare Match) (MTU5)

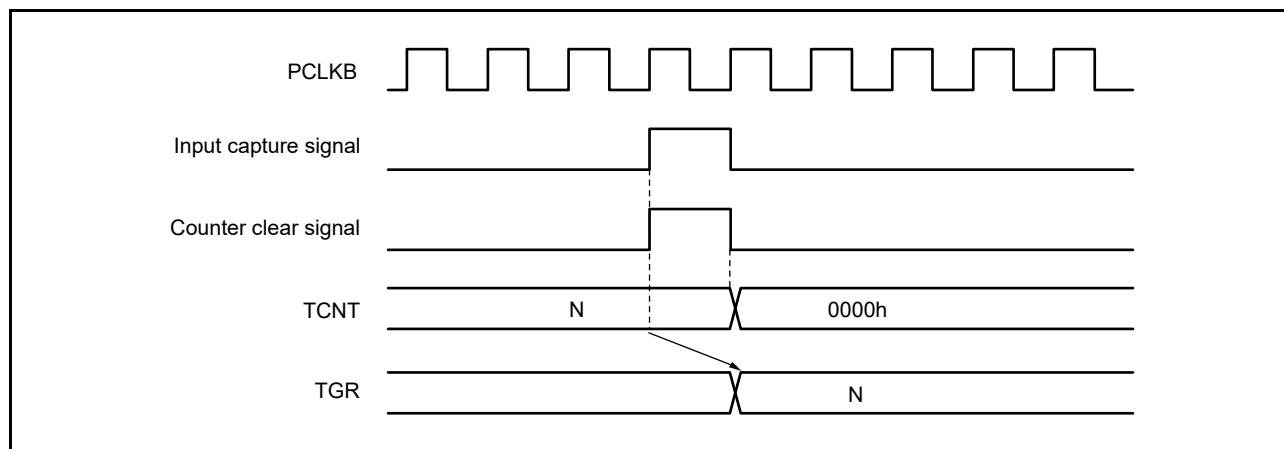


Figure 19.109 Counter Clear Timing (Input Capture) (MTU0 to MTU5)

(5) Buffer Operation Timing

Figure 19.110 to Figure 19.112 show the timing in buffer operation.

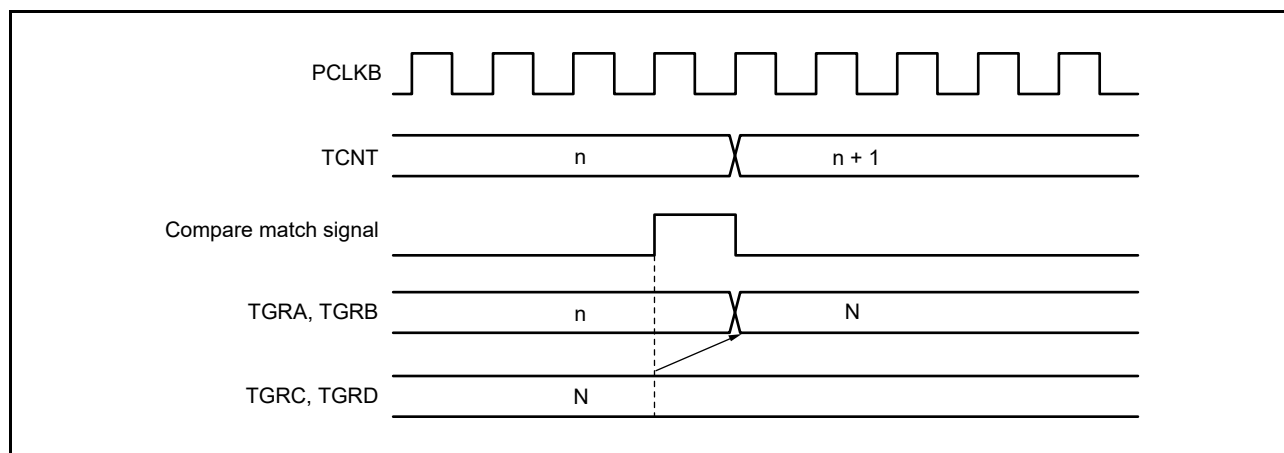


Figure 19.110 Buffer Operation Timing (Compare Match)

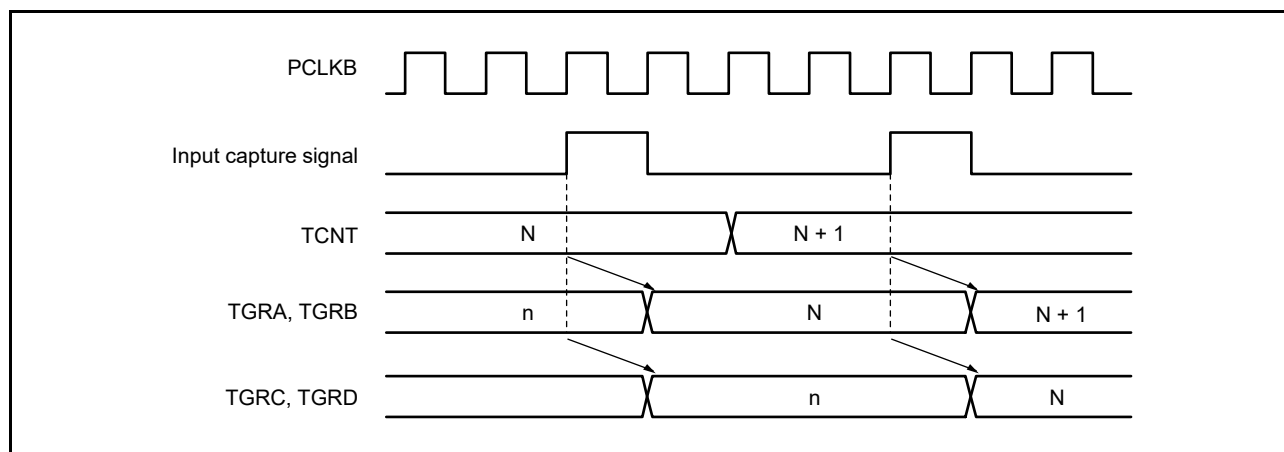


Figure 19.111 Buffer Operation Timing (Input Capture)

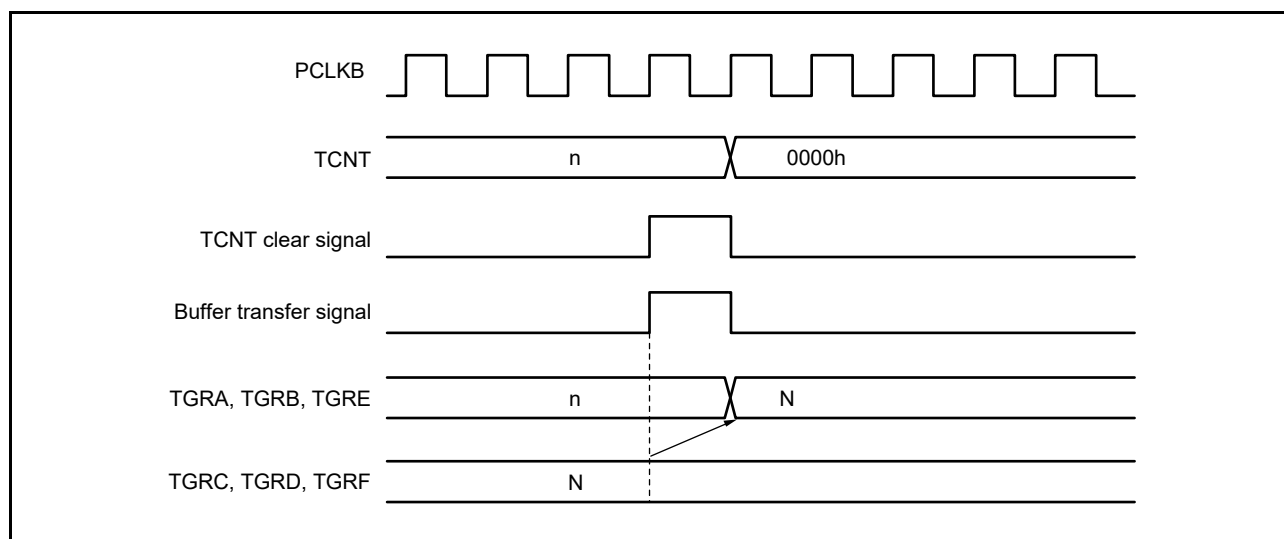


Figure 19.112 Buffer Operation Timing (When TCNT Cleared)

(6) Buffer Transfer Timing (Complementary PWM Mode)

Figure 19.113 to Figure 19.115 show the buffer transfer timing in complementary PWM mode.

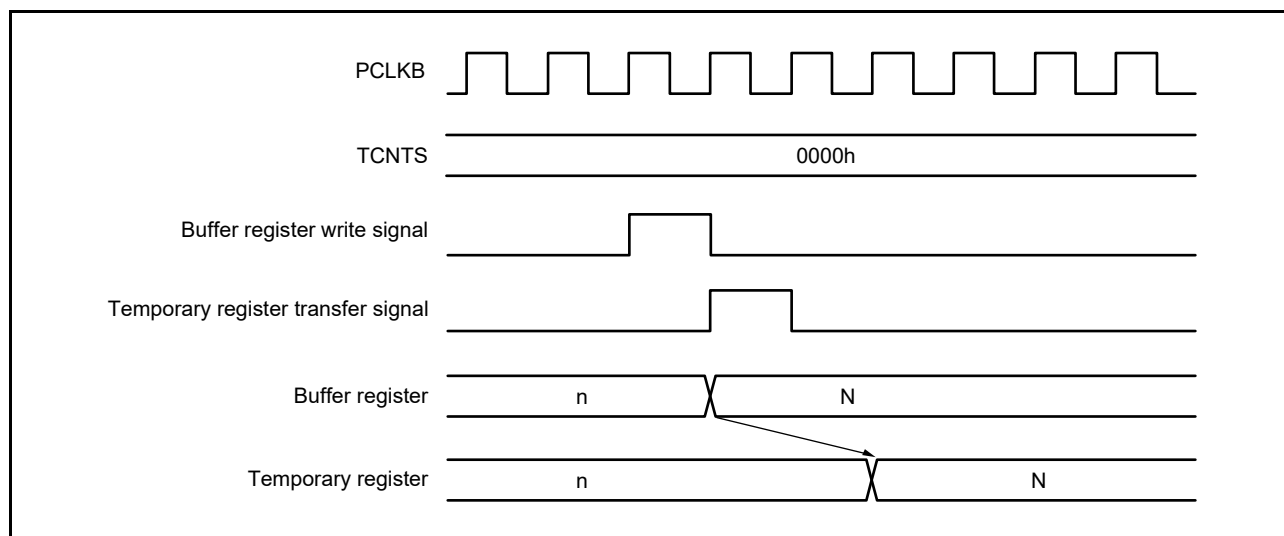


Figure 19.113 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Stopped)

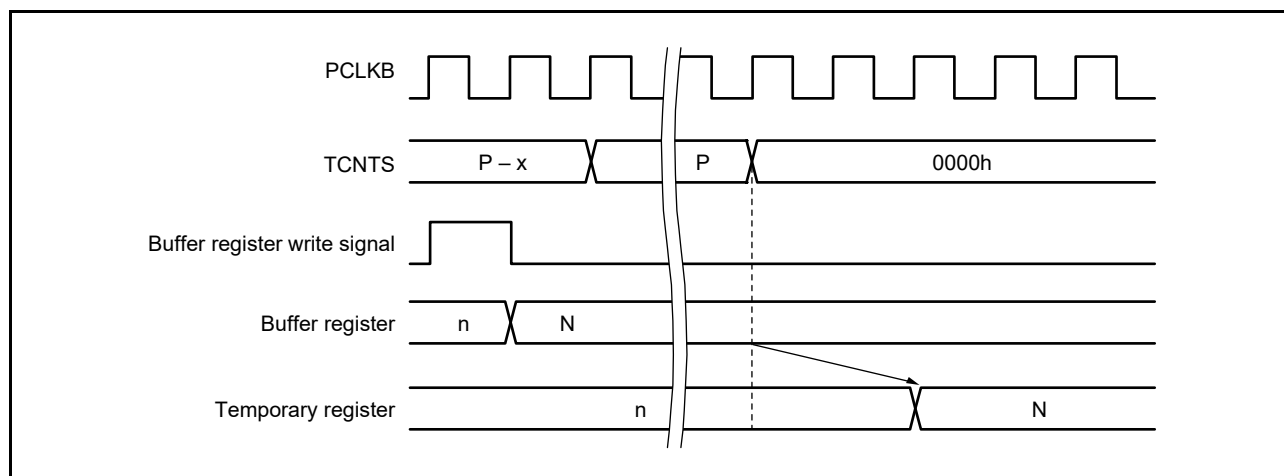


Figure 19.114 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Operating)

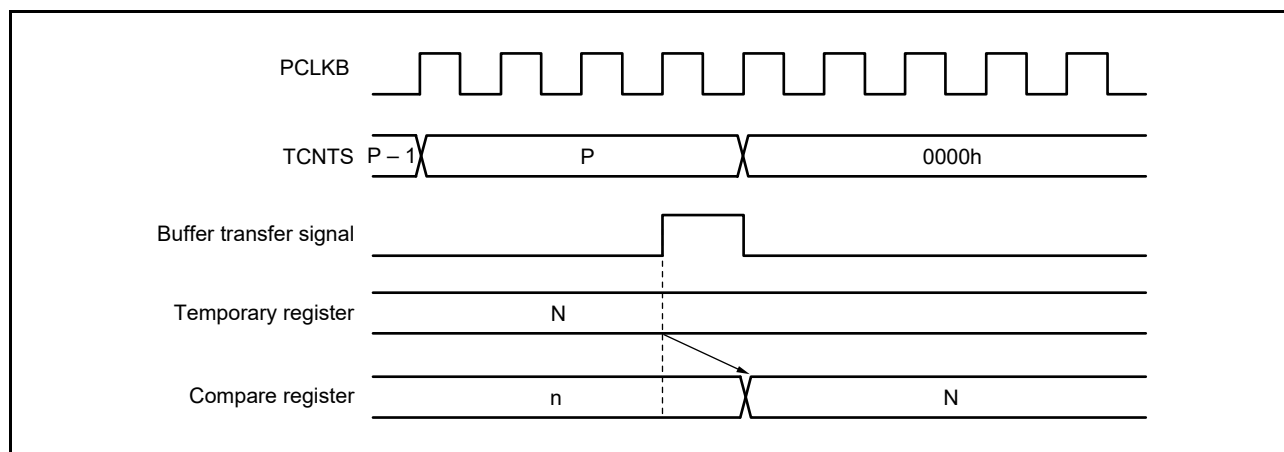


Figure 19.115 Transfer Timing from Temporary Register to Compare Register

19.5.2 Interrupt Signal Timing

(1) TGI Interrupt Timing by Compare Match

Figure 19.116 and Figure 19.117 show the TGI interrupt request signal timing when a compare match occurs.

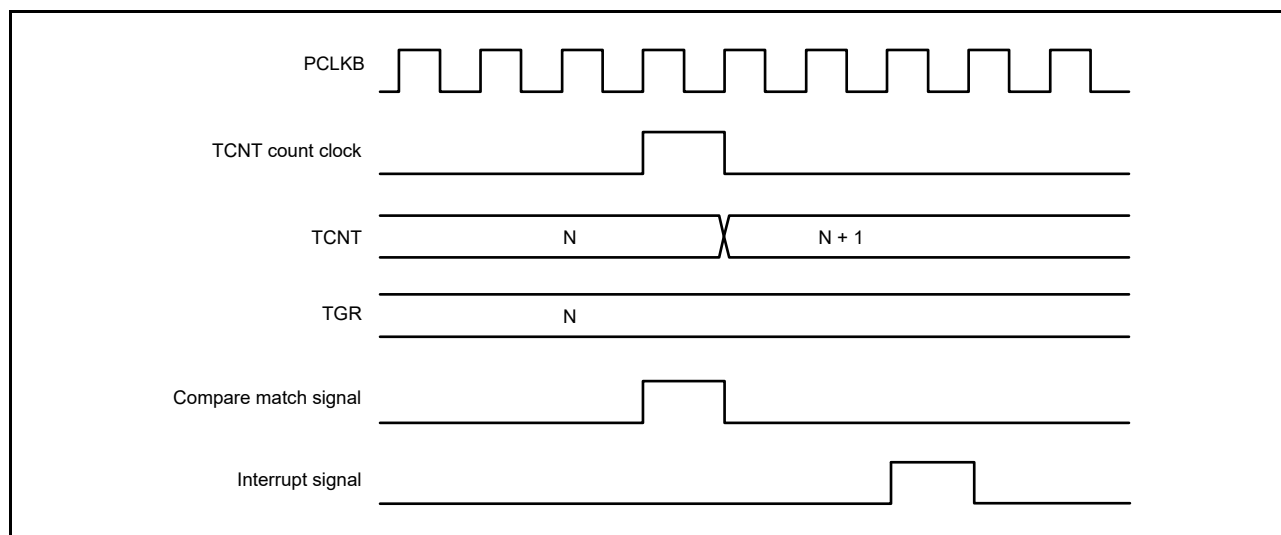


Figure 19.116 TGI Interrupt Timing (Compare Match) (MTU0 to MTU4)

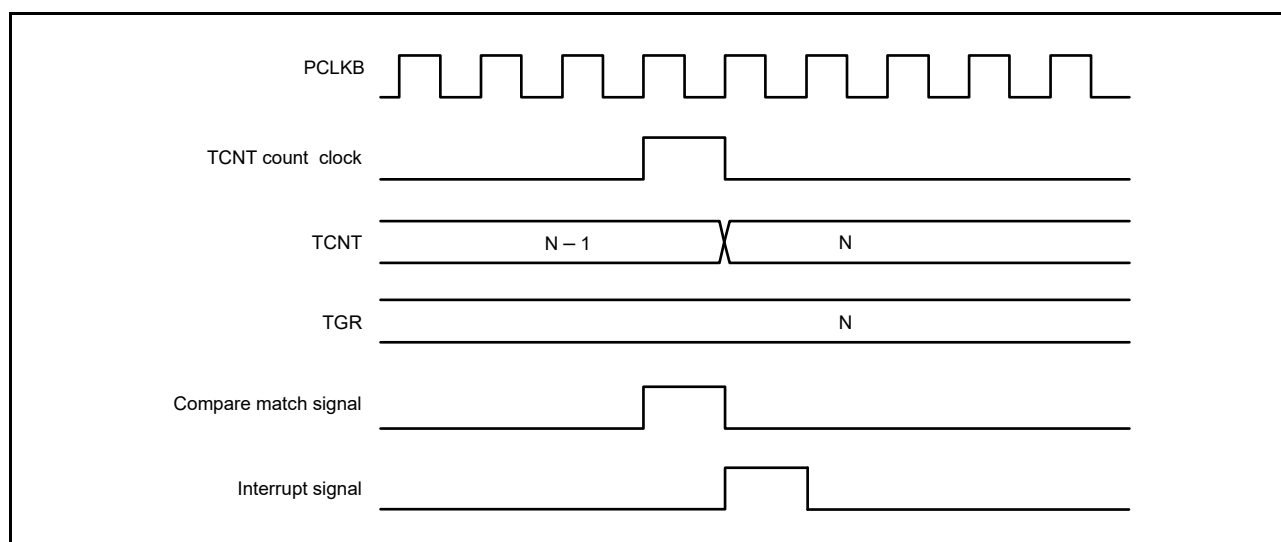


Figure 19.117 TGI Interrupt Timing (Compare Match) (MTU5)

(2) TGI Interrupt Timing by Input Capture

Figure 19.118 and Figure 19.119 show the TGI interrupt request signal timing when an input capture occurs.

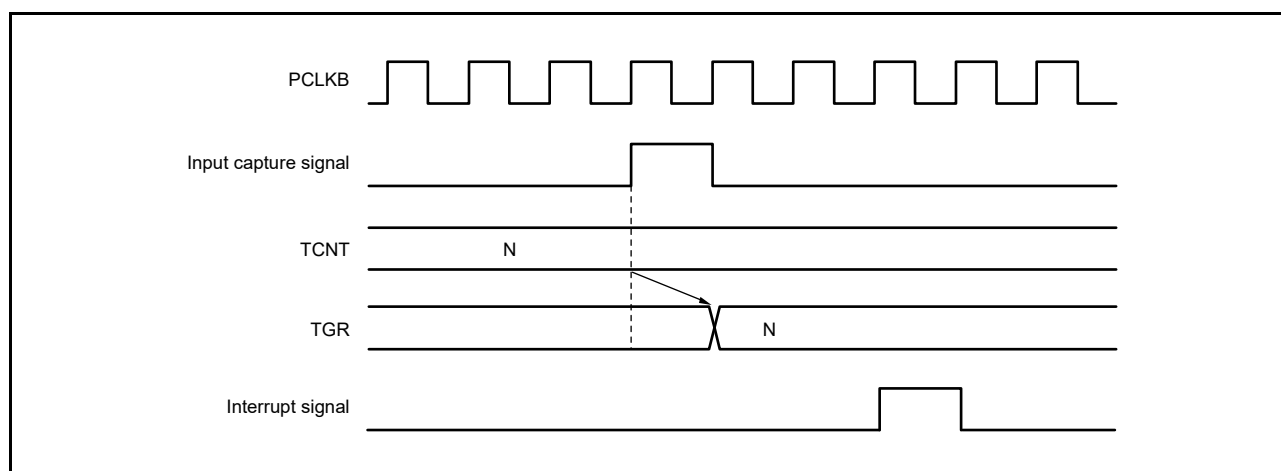


Figure 19.118 TGI Interrupt Timing (Input Capture) (MTU0 to MTU4)

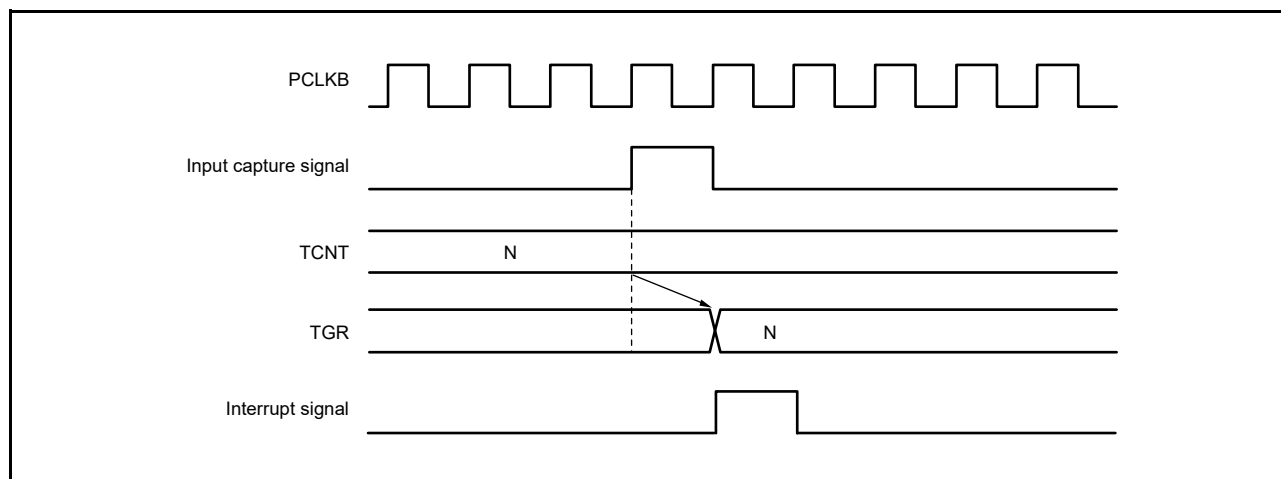


Figure 19.119 TGI Interrupt Timing (Input Capture) (MTU5)

(3) TCIV and TCIU Interrupt Timing

Figure 19.120 shows the TCIV interrupt request signal timing when an overflow is generated.

Figure 19.121 shows the TCIU interrupt request signal timing when an underflow is generated.

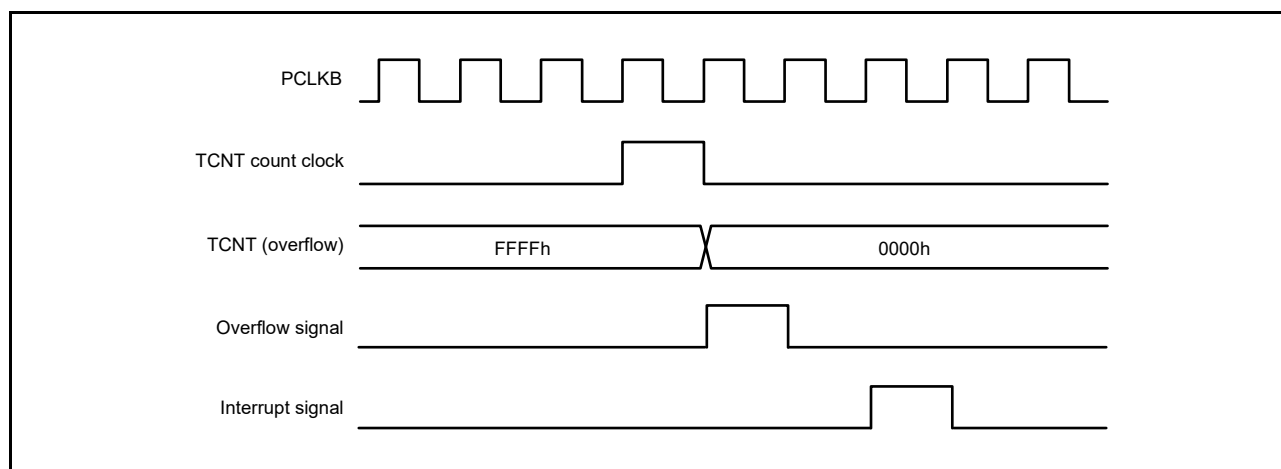


Figure 19.120 TCIV Interrupt Timing

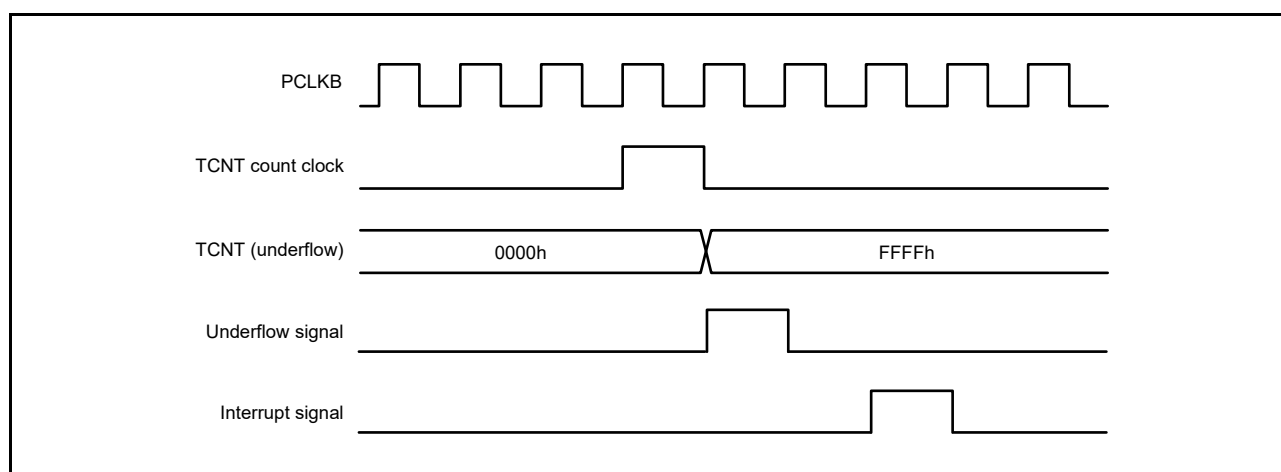


Figure 19.121 TCIU Interrupt Timing

19.6 Usage Notes

19.6.1 Module Stop Function Setting

MTU operation can be disabled or enabled using the module stop control register. MTU operation is stopped with the initial setting. Register access is enabled by releasing the module clock stop state. For details, refer to [section 11, Low Power Consumption](#).

19.6.2 Count Clock Restrictions

The count clock source pulse width must be at least 1.5 PCLKB cycles for single-edge detection, and at least 2.5 PCLKB cycles for both-edge detection. The MTU will not operate properly at narrower pulse widths.

In phase counting mode, the phase difference and overlap between the two input clocks must be at least 1.5 PCLKB cycles, and the pulse width must be at least 2.5 PCLKB cycles. [Figure 19.122](#) shows the input clock conditions in phase counting mode.

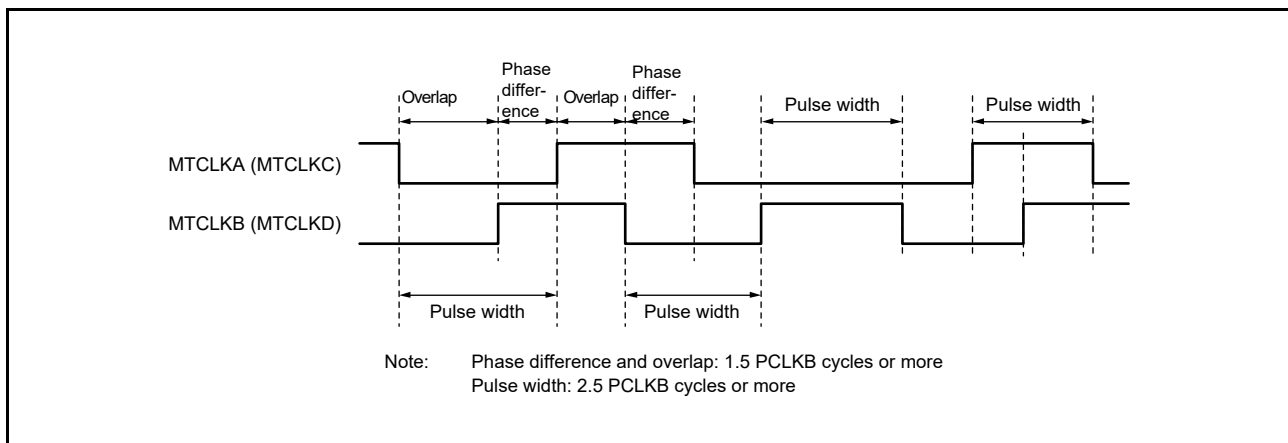


Figure 19.122 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

19.6.3 Note on Period Setting

When counter clearing on compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which TCNT updates the matched count value). Consequently, the actual counter frequency is given by the following formula:

- MTU0 to MTU4

$$f = \frac{\text{CNTCLK}}{N + 1}$$

- MTU5

$$f = \frac{\text{CNTCLK}}{N}$$

f: Counter frequency

CNTCLK: The count clock frequency set by TCR.TPSC[2:0] and TCR2.TPSC2[2:0]

N: TGR setting

19.6.4 Contention between TCNT Write and Clear Operations

If the counter clear signal is generated in the TCNT write cycle, TCNT clearing takes precedence and TCNT write operation is not performed.

Figure 19.123 shows the timing in this case.

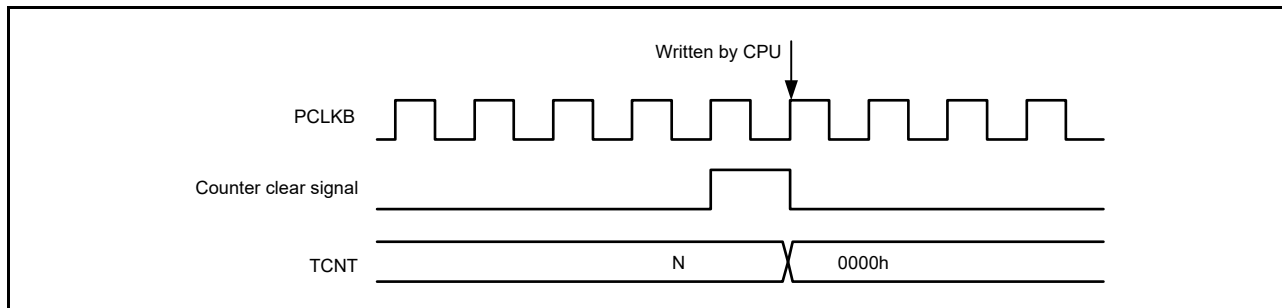


Figure 19.123 Contention between TCNT Write and Clear Operations

19.6.5 Contention between TCNT Write and Increment Operations

If incrementing occurs in a TCNT write cycle, TCNT write operation takes precedence and TCNT is not incremented.

Figure 19.124 shows the timing in this case.

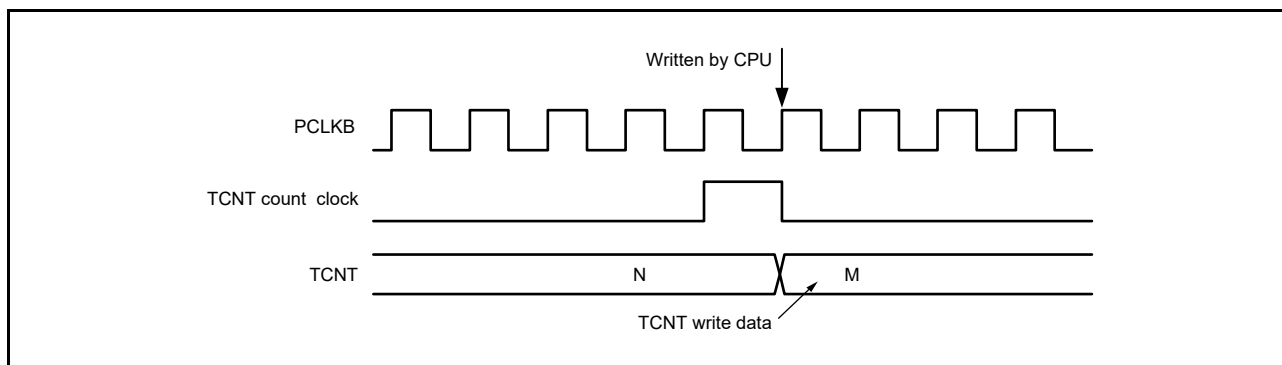


Figure 19.124 Contention between TCNT Write and Increment Operations

19.6.6 Contention between TGR Write Operation and Compare Match

If a compare match occurs in a TGR write cycle, TGR write operation is executed and the compare match signal is also generated.

Figure 19.125 shows the timing in this case.

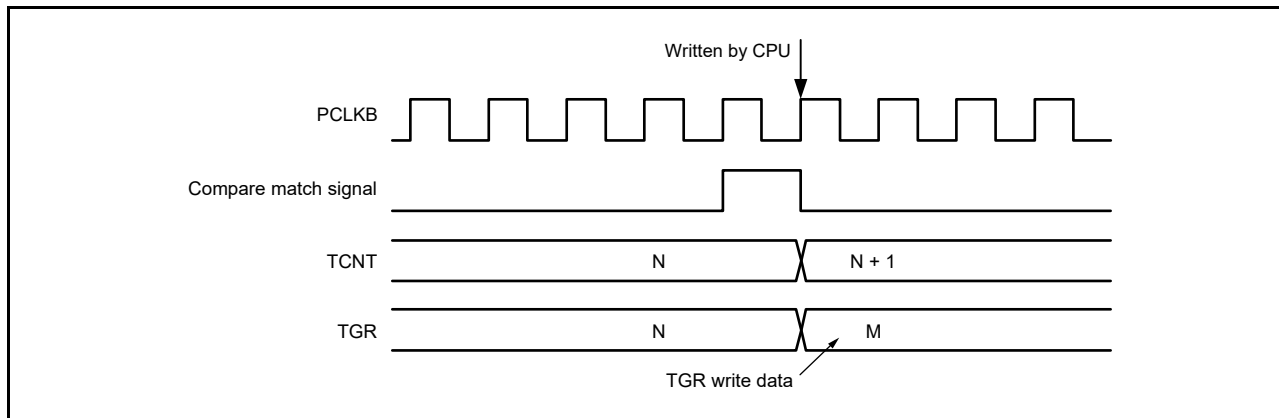


Figure 19.125 Contention between TGR Write Operation and Compare Match

19.6.7 Contention between Buffer Register Write Operation and Compare Match

If a compare match occurs in a buffer register write cycle, the data before write operation is transferred to TGR by the buffer operation.

Figure 19.126 shows the timing in this case.

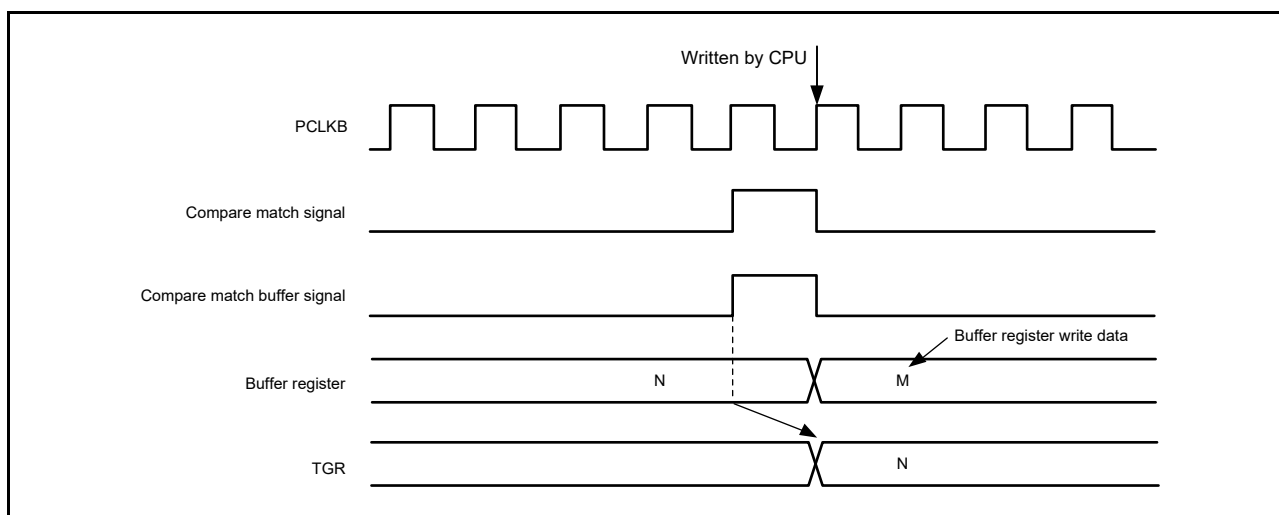


Figure 19.126 Contention between Buffer Register Write Operation and Compare Match

19.6.8 Contention between Buffer Register Write and TCNT Clear Operations

When the buffer transfer timing is set at the TCNT clear timing by the timer buffer transfer mode register (TBTM), if TCNT clearing occurs in the buffer register write cycle, the data before write operation is transferred to TGR by the buffer operation.

Figure 19.127 shows the timing in this case.

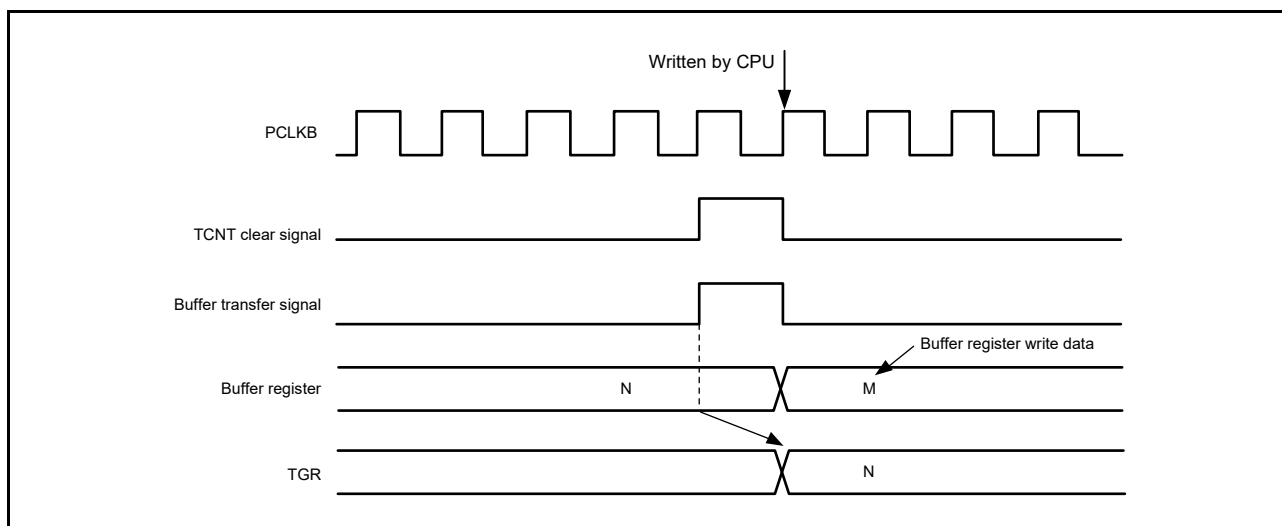


Figure 19.127 Contention between Buffer Register Write and TCNT Clear Operations

19.6.9 Contention between TGR Read Operation and Input Capture

If an input capture signal is generated in a TGR read cycle, the data before input capture transfer is read.

Figure 19.128 shows the timing in this case.

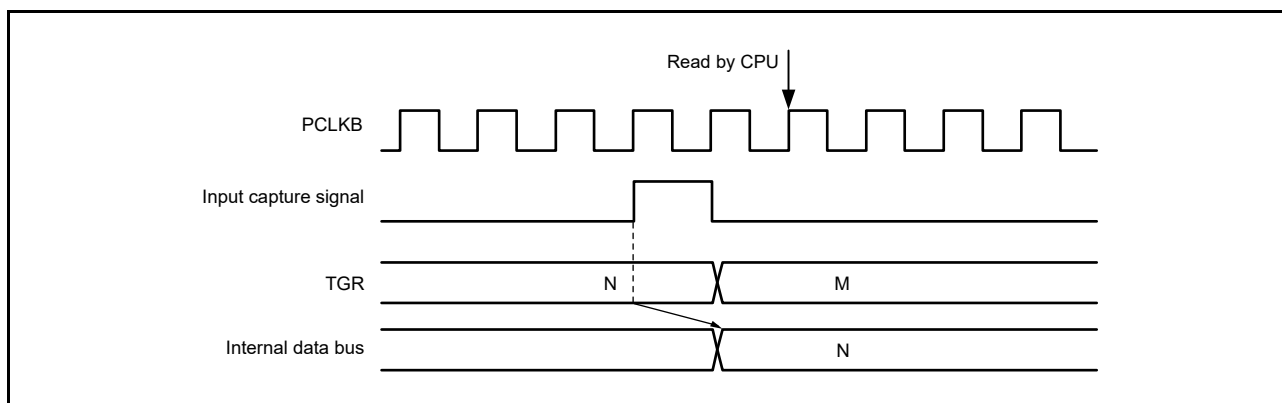


Figure 19.128 Contention between TGR Read Operation and Input Capture (MTU0 to MTU5)

19.6.10 Contention between TGR Write Operation and Input Capture

If an input capture signal is generated in the TGR write cycle, the input capture operation takes precedence and the TGR write operation is not performed in MTU0 to MTU4. In MTU5, the TGR write operation is performed and the input capture signal is generated.

Figure 19.129 and Figure 19.130 show the timing in this case.

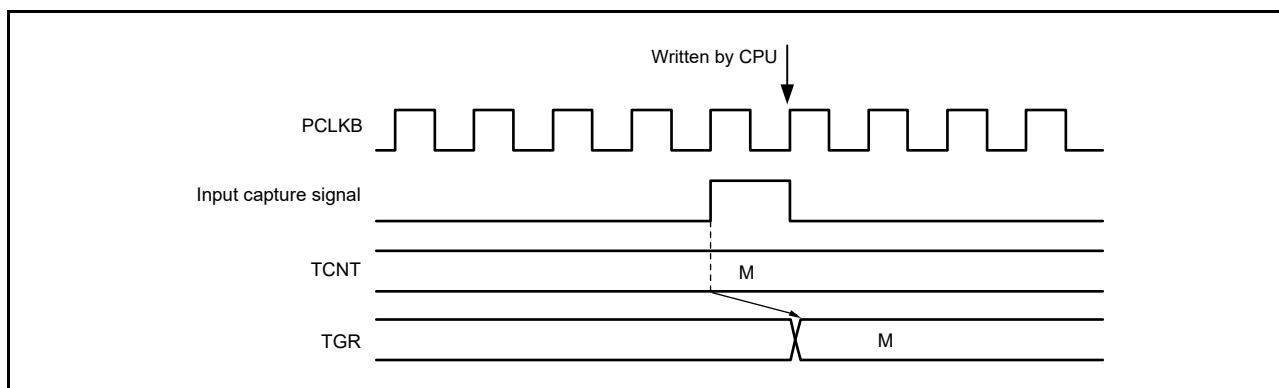


Figure 19.129 Contention between TGR Write Operation and Input Capture (MTU0 to MTU4)

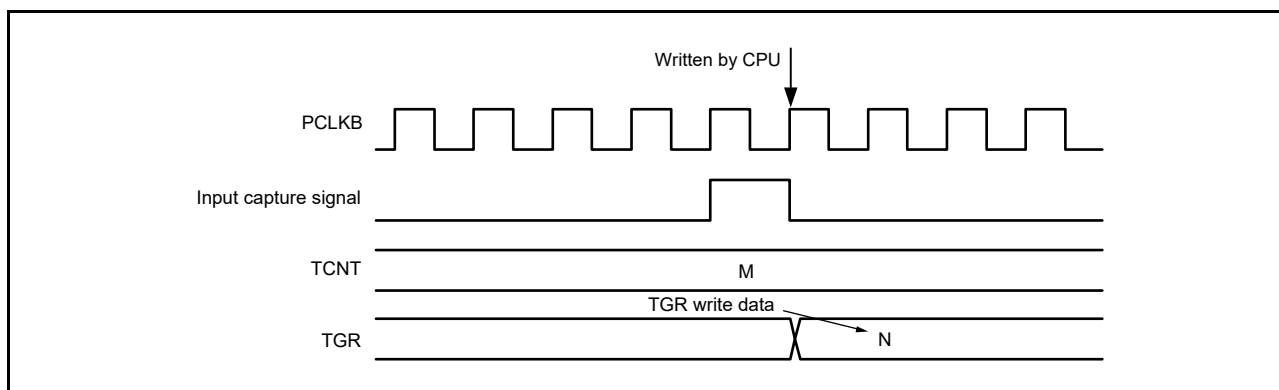


Figure 19.130 Contention between TGR Write Operation and Input Capture (MTU5)

19.6.11 Contention between Buffer Register Write Operation and Input Capture

If an input capture signal is generated in the buffer register write cycle, the buffer operation takes precedence and the buffer register write operation is not performed.

Figure 19.131 shows the timing in this case.

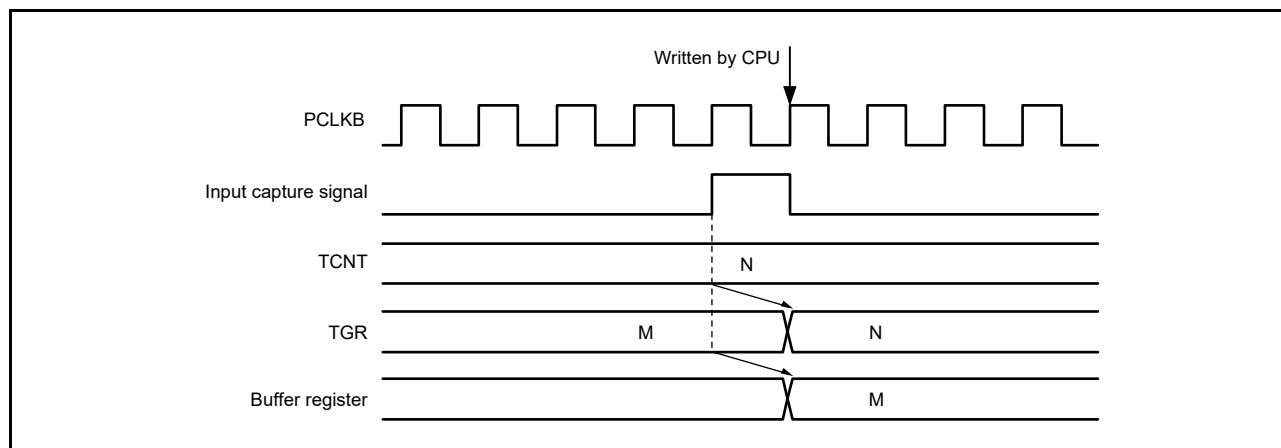


Figure 19.131 Contention between Buffer Register Write Operation and Input Capture

19.6.12 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation

With timer counters MTU1.TCNT and MTU2.TCNT in a cascade, when a contention occurs between MTU1.TCNT counting (an MTU2.TCNT overflow/underflow) and the MTU2.TCNT write operation, the MTU2.TCNT write operation is performed and the MTU1.TCNT count signal is disabled. In this case, if MTU1.TGRA works as a compare match register and there is a match between the MTU1.TGRA and MTU1.TCNT values, a compare match signal is issued.

Furthermore, when the MTU1.TCNT count clock is selected as the input capture source of MTU0, MTU0.TGRA to MTU0.TGRD work in input capture mode. In addition, when the MTU0.TGRC compare match/input capture is selected as the input capture source of MTU1.TGRB, MTU1.TGRB works in input capture mode.

Figure 19.132 shows the timing in this case.

When setting the TCNT clearing function in cascaded operation, be sure to synchronize MTU1 and MTU2.

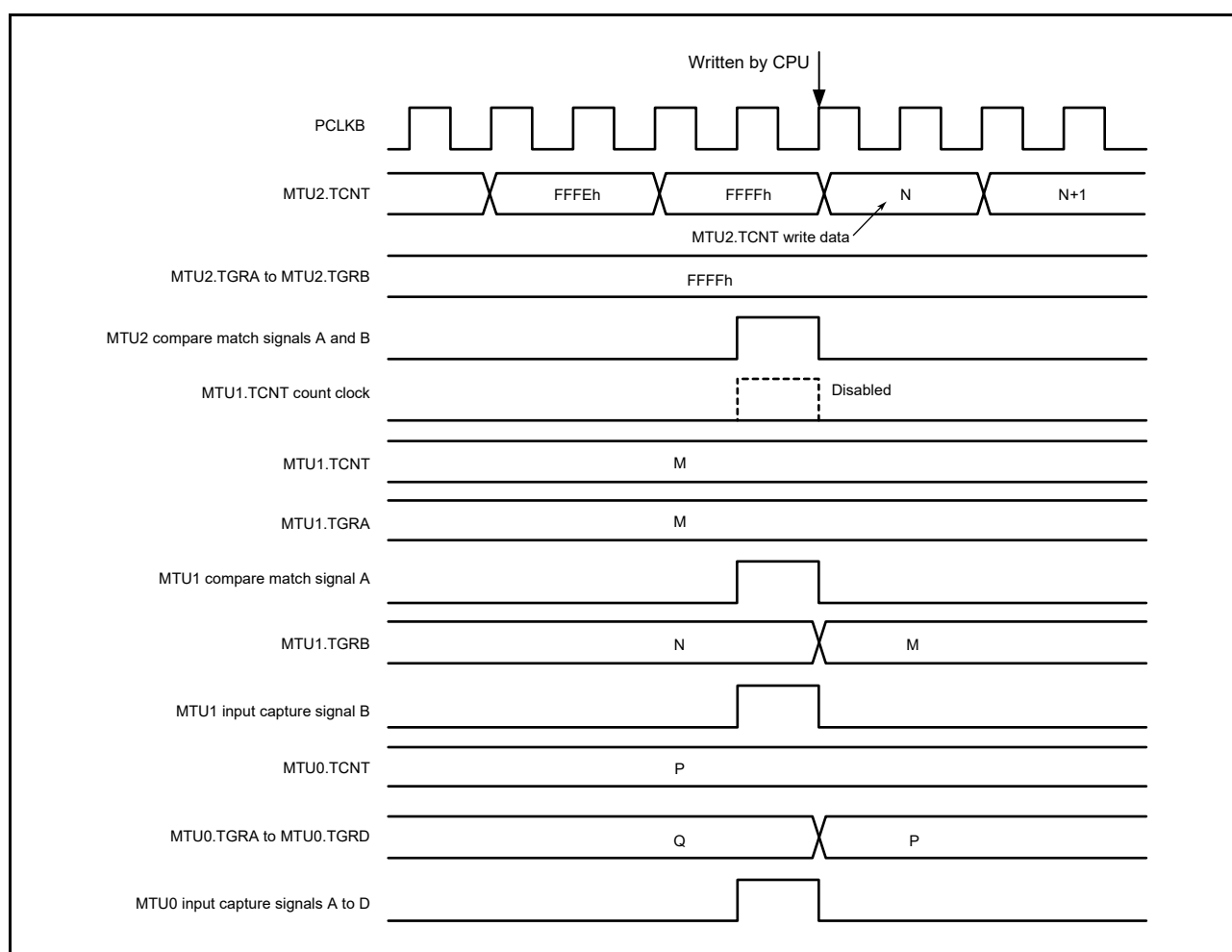


Figure 19.132 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation

19.6.13 Counter Value When Count Operation is Stopped in Complementary PWM Mode

When counting operation in MTU3.TCNT and MTU4.TCNT is stopped in complementary PWM mode, the MTU3.TCNT value is set to the timer dead time register (TDDRA) value and MTU4.TCNT is set to 0000h. When operation is restarted in complementary PWM mode, counting begins automatically from the initial setting state. Figure 19.133 shows this operation.

When counting begins in another operating mode, be sure to make initial settings in MTU3.TCNT and MTU4.TCNT.

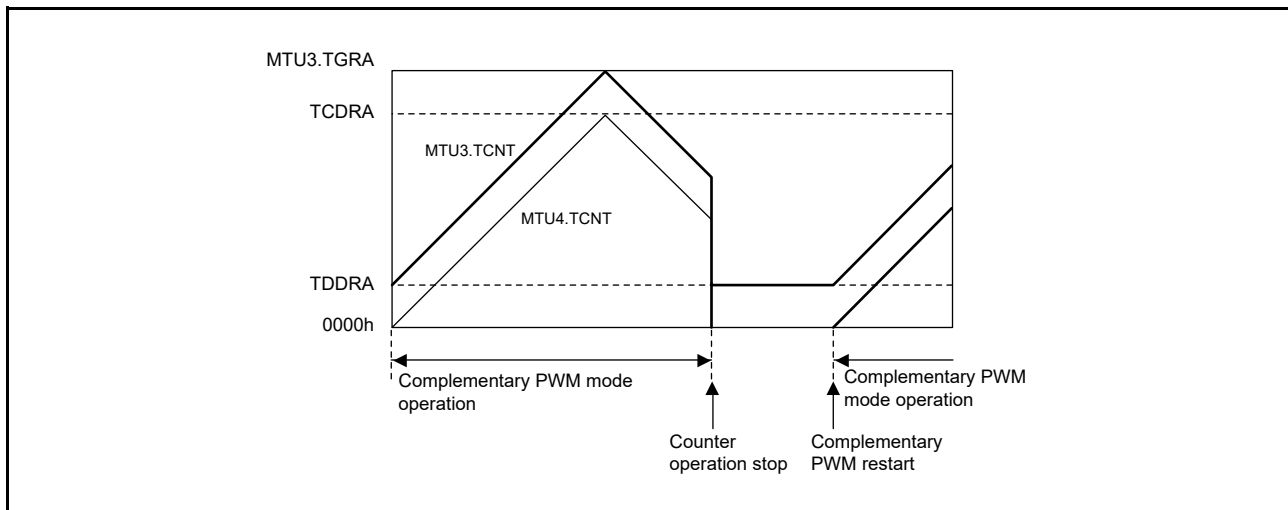


Figure 19.133 Counter Value When Stopped in Complementary PWM Mode

19.6.14 Buffer Operation Setting in Complementary PWM Mode

When modifying the PWM period set register (MTU3.TGRA), timer period data register (TCDRA), and duty set registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB) in complementary PWM mode, be sure to use buffer operation. In addition, set the BFA and BFB bits in MTU4.TMDR1 to 0. The MTIOC4C pin cannot output waveforms if the BFA bit in MTU4.TMDR1 is set to 1. Likewise, the MTIOC4D pin cannot output waveforms if the BFB bit in MTU4.TMDR1 is set to 1.

In complementary PWM mode, buffer operation in MTU3 and MTU4 depends on the settings in bits BFA and BFB of MTU3.TMDR1. When the BFA bit in MTU3.TMDR1 is set to 1, MTU3.TGRC functions as a buffer register for MTU3.TGRA. At the same time, MTU4.TGRC functions as a buffer register for MTU4.TGRA, and TCBRA functions as a buffer register for TCDRA.

19.6.15 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode

When setting buffer operation in reset-synchronized PWM mode, set the BFA and BFB bits in MTU4.TMDR1 to 0. The MTIOC4C pin cannot output waveforms if the BFA bit in MTU4.TMDR1 is set to 1. Likewise, the MTIOC4D pin cannot output waveforms if the BFB bit in MTU4.TMDR1 is set to 1.

In reset-synchronized PWM mode, buffer operation in MTU3 and MTU4 depends on the settings in the BFA and BFB bits of MTU3.TMDR1. For example, if the BFA bit in MTU3.TMDR1 is set to 1, MTU3.TGRC functions as a buffer register for MTU3.TGRA. At the same time, MTU4.TGRC functions as a buffer register for MTU4.TGRA.

While the MTU3.TGRC and MTU3.TGRD are operating as buffer registers, a TGI_mn interrupt (m = C, D; n = 3, 4) is not generated.

Figure 19.134 shows an example of MTU3.TGR, MTU4.TGR, MTIOC3, and MTIOC4 operation with the BFA and BFB bits in MTU3.TMDR1 set to 1 and the BFA and BFB bits in MTU4.TMDR1 set to 0.

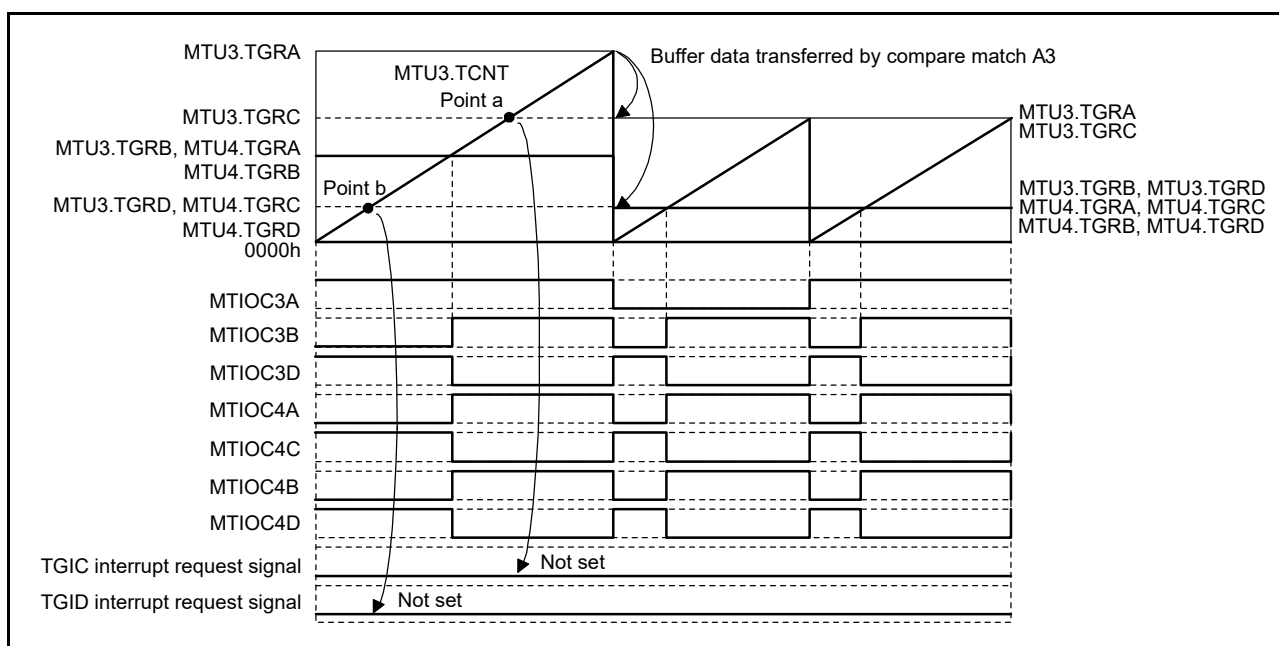


Figure 19.134 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode

19.6.16 Overflow in Reset-Synchronized PWM Mode

After reset-synchronized PWM mode is selected, MTU3.TCNT and MTU4.TCNT start counting when the CST3 bit of TSTRA is set to 1. In this state, the MTU4.TCNT count clock source and count edge are determined by the MTU3.TCR setting.

In reset-synchronized PWM mode, with period register MTU3.TGRA set to FFFFh and the MTU3.TGRA compare match selected as the counter clearing source, MTU3.TCNT and MTU4.TCNT count up to FFFFh, then a compare match occurs with MTU3.TGRA, and MTU3.TCNT and MTU4.TCNT are both cleared. In this case, a TCIVn interrupt (n = 3, 4) is not generated.

Figure 19.135 shows an example of operation in reset-synchronized PWM mode with period register MTU3.TGRA set to FFFFh and the MTU3.TGRA compare match specified for the counter clearing source.

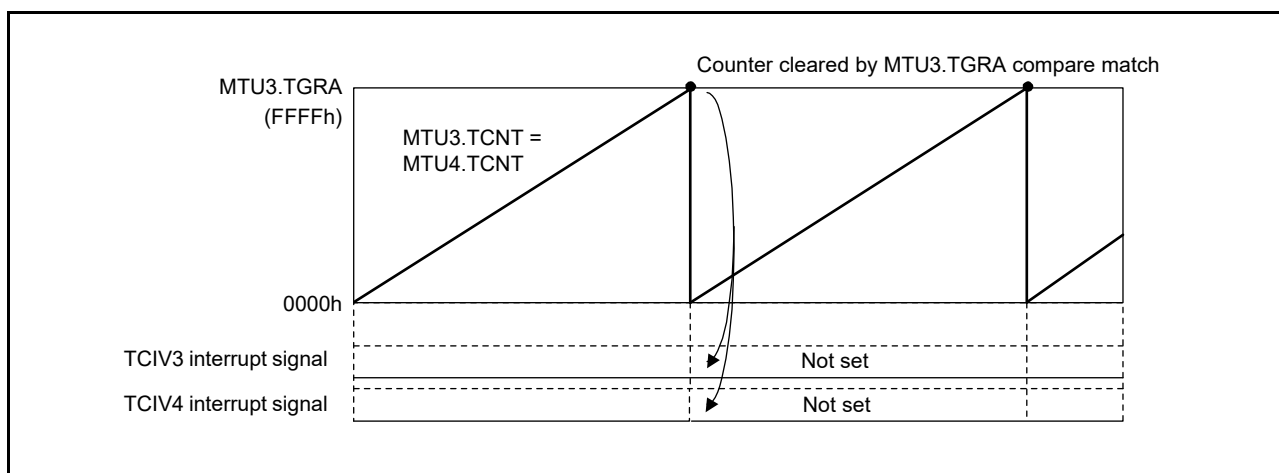


Figure 19.135 Overflow in Reset-Synchronized PWM Mode

19.6.17 Contention between Overflow/Underflow and Counter Clearing

If an overflow/underflow and counter clearing occur simultaneously, a TCIV_n interrupt ($n = 0$ to 4) nor a TCIUn interrupt ($n = 1, 2$) is not generated and TCNT clearing takes precedence.

Figure 19.136 shows the operation timing when a TGR compare match is specified as the clearing source and TGR is set to FFFFh.

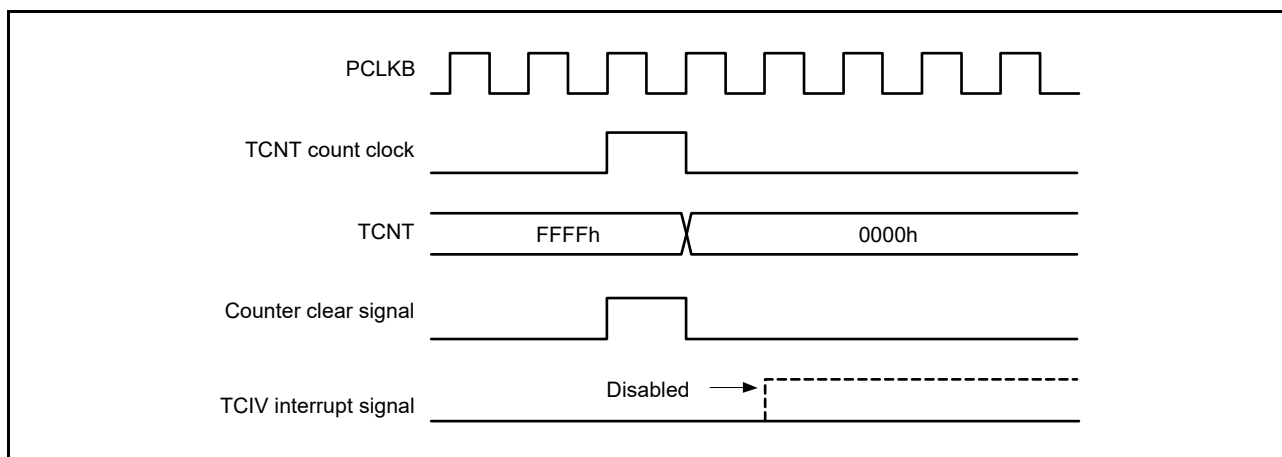


Figure 19.136 Contention between Overflow and Counter Clearing

19.6.18 Contention between TCNT Write Operation and Overflow/Underflow

If TCNT counts up or down in a TCNT write cycle and an overflow or an underflow occurs, the TCNT write operation takes precedence. A TCIV_n interrupt ($n = 0$ to 4) nor a TCIUn interrupt ($n = 1, 2$) is not generated.

Figure 19.137 shows the operation timing when there is contention between TCNT write operation and overflow.

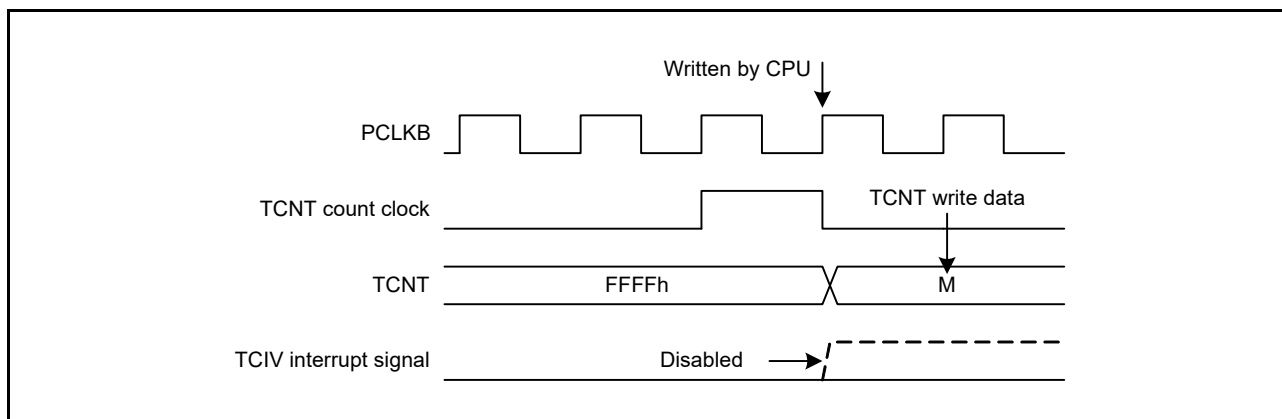


Figure 19.137 Contention between TCNT Write Operation and Overflow

19.6.19 Note on Transition from Normal Mode or PWM Mode 1 to Reset-Synchronized PWM Mode

When making a transition from normal mode or PWM mode 1 to reset-synchronized PWM mode in MTU3 and MTU4, if the counter is stopped while the output pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B and MTIOC4D) are held at a high level and then operation is started after a transition to reset-synchronized PWM mode, the initial pin output will not be correct.

When making a transition from normal mode to reset-synchronized PWM mode, write 11h to MTU3.TIORH, MTU3.TIORL, MTU4.TIORH, and MTU4.TIORL to initialize the output pin state to a low level, then set the registers to the initial value (00h) before making the mode transition.

When making a transition from PWM mode 1 to reset-synchronized PWM mode, switch to normal mode, initialize the output pin state to a low level, and then set the registers to the initial value (00h) before making the transition to reset-synchronized PWM mode.

19.6.20 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode

When MTU3 and MTU4 are in complementary PWM mode or reset-synchronized PWM mode, the PWM waveform output level is determined by the OLSP and OLSN bits in the timer output control register (TOCR1A). In complementary PWM mode or reset-synchronized PWM mode, TIOR should be set to 00h.

The output level in negative phase when the TDERA.TDER bit is set to 0 in complementary PWM mode (the dead time is not generated) does not depend on the setting of the TOCR1A.OLSN bit. It is equivalent to the inverted level of positive phase output based on the setting of the TOCR1A.OLSP bit.

19.6.21 Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection

When timer counters 1 and 2 (MTU1.TCNT and MTU2.TCNT) operate as a 32-bit counter in cascade connection, the cascaded counter value cannot be captured successfully in some cases even if input-capture input is simultaneously done to MTIOC1A and MTIOC2A or to MTIOC1B and MTIOC2B. This is because the input timing of MTIOC1A and MTIOC2A or to MTIOC1B and MTIOC2B may not be the same when external input-capture signals input into MTU1.TCNT and MTU2.TCNT are taken in synchronization with the internal clock.

For example, MTU1.TCNT (the counter for upper 16 bits) does not capture the count-up value by an overflow from MTU2.TCNT (the counter for lower 16 bits) but captures the count value before the up-counting. In this case, the values of MTU1.TCNT = FFF1h and MTU2.TCNT = 0000h should be transferred to MTU1.TGRA and MTU2.TGRA or to MTU1.TGRB and MTU2.TGRB, but the values of MTU1.TCNT = FFF0h and MTU2.TCNT = 0000h are erroneously transferred.

The MTU has a function that allows simultaneous capture of MTU1.TCNT and MTU2.TCNT with a single input capture input. This function can be used to read the 32-bit counter such that MTU1.TCNT and MTU2.TCNT are captured at the same time. For details, refer to section 19.2.11, Timer Input Capture Control Register (TICCR).

19.6.22 Interrupt Skipping Function 2

When interrupt skipping function 2 is in use and the difference between the values in MTU4.TADCORA and MTU4.TADCORB is small, correct counting of the number skipped may not be possible, in which case requests for A/D conversion will not be generated with the expected timing. The conditions listed below thus apply to these settings.

- (1) When the number skipped is zero for skipping function 2
 - The difference between the values in MTU4.TADCORA and MTU4.TADCORB must be at least four.
 - The interval of comparison for MTU4.TADCORA must be at least four cycles of PCLKB (the updated value of MTU4.TADCORA is set to the previous value plus or minus at least four).
 - The interval of comparison for MTU4.TADCORB must be at least four cycles of PCLKB (the updated value of MTU4.TADCORB is set to the previous value plus or minus at least four).
- (2) When the number skipped is one or more for skipping function 2
 - The difference between the values in MTU4.TADCORA and MTU4.TADCORB must be at least two.
 - The interval of comparison for MTU4.TADCORB must be at least two cycles of PCLKB (the updated value of MTU4.TADCORB is set to the previous value plus or minus at least two).

19.6.23 Notes When Complementary PWM Mode Output Protection Function is Not Used

The complementary PWM mode output protection function is initially enabled. For details, refer to section 20, Port Output Enable 3 (POE3E).

19.6.24 Notes Regarding Timer Counter (MTU5.TCNT) and Timer General Register (MTU5.TGR)

Do not set an MTU5.TGR_m ($m = U, V, W$) bit to the value of the corresponding MTU5.TCNT_m ($m = U, V, W$) plus one while counting by the MTU5.TCNT_m ($m = U, V, W$) register is stopped. If an MTU5.TGR_m ($m = U, V, W$) bit is set to the value of the corresponding MTU5.TCNT_m ($m = U, V, W$) plus one while counting by the MTU5.TCNT_m ($m = U, V, W$) is stopped, a compare-match will be generated even though counting is stopped.

In this case, if the compare match enable bit (MTU5.TIER.TGIE5_m ($m = U, V, W$) bit is set to 1 (enabling interrupts), a compare-match interrupt will also be generated. If the value of the timer compare match clear register is 1 (enabled), the timer is automatically cleared to 0000h when the compare-match is generated, regardless of whether interrupts from the MTU5.TCNT_m ($m = U, V, W$) are enabled or disabled.

19.6.25 Notes to Prevent Malfunctions in Synchronous Clearing for Complementary PWM Mode

If control of the output waveform is enabled (TWCRA.WRE bit = 1) at the time of synchronous counter clearing in complementary PWM mode, satisfaction of either condition 1 or 2 below has the following effects.

- Dead time on the PWM output pins is shortened (or disappears).
- The active level is output on the negative phase PWM output pins beyond the period for active-level output.

Condition 1: In portion (10) of the initial output inhibition period in Figure 19.138, synchronous clearing occurs within the dead-time period for PWM output.

Condition 2: In portions (10) and (11) of the initial output inhibition period in Figure 19.139, synchronous clearing occurs when any condition from among $MTU3.TGRB \leq TDDRA$, $MTU4.TGRA \leq TDDRA$, or $MTU4.TGRB \leq TDDRA$ is satisfied.

The following method avoids the above phenomena.

Ensure that synchronous clearing proceeds with the value of each comparison register ($MTU3.TGRB$, $MTU4.TGRA$, and $MTU4.TGRB$) set to at least double the value of the $TDDRA$ register.

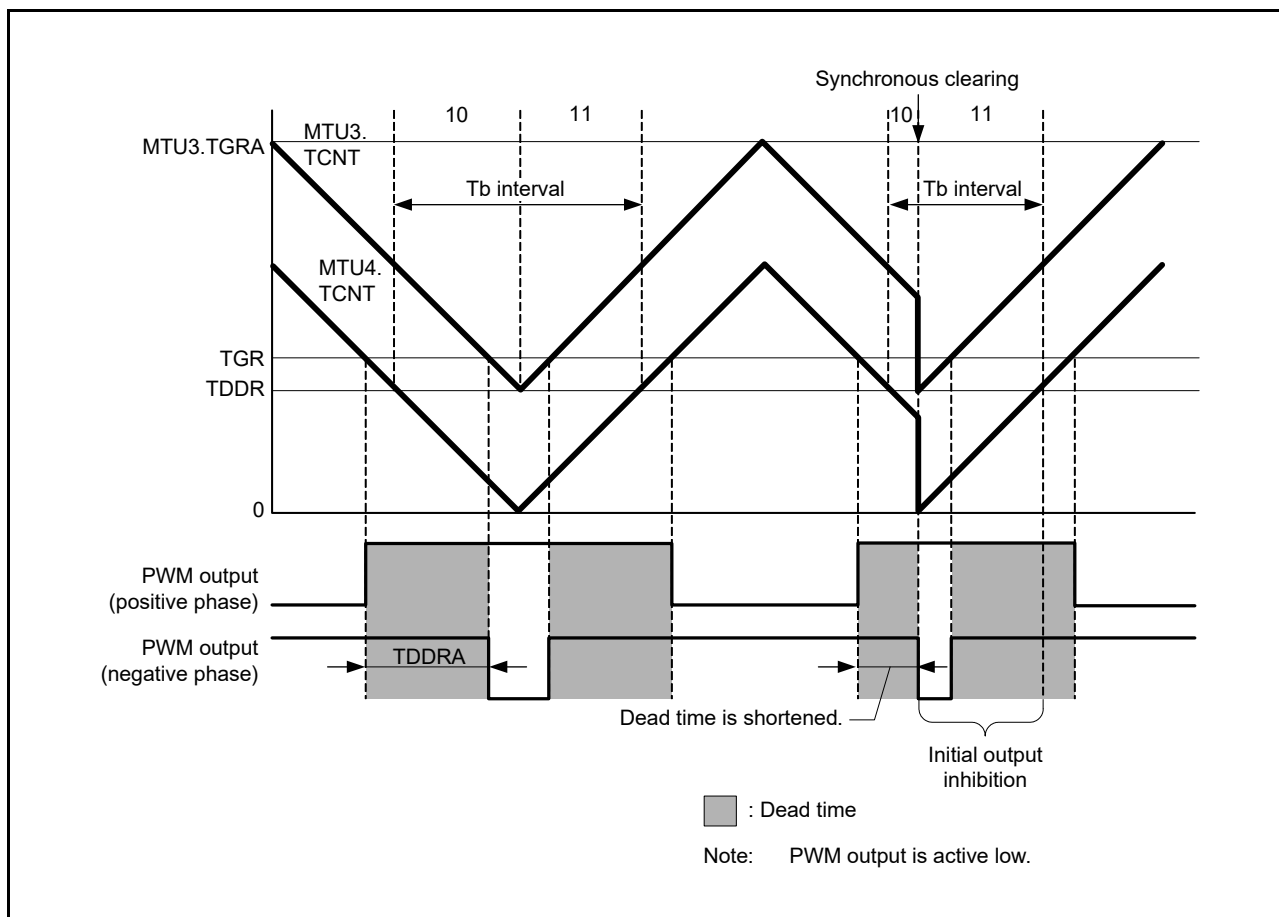


Figure 19.138 Example of Synchronous Clearing (When Condition 1 Applies)

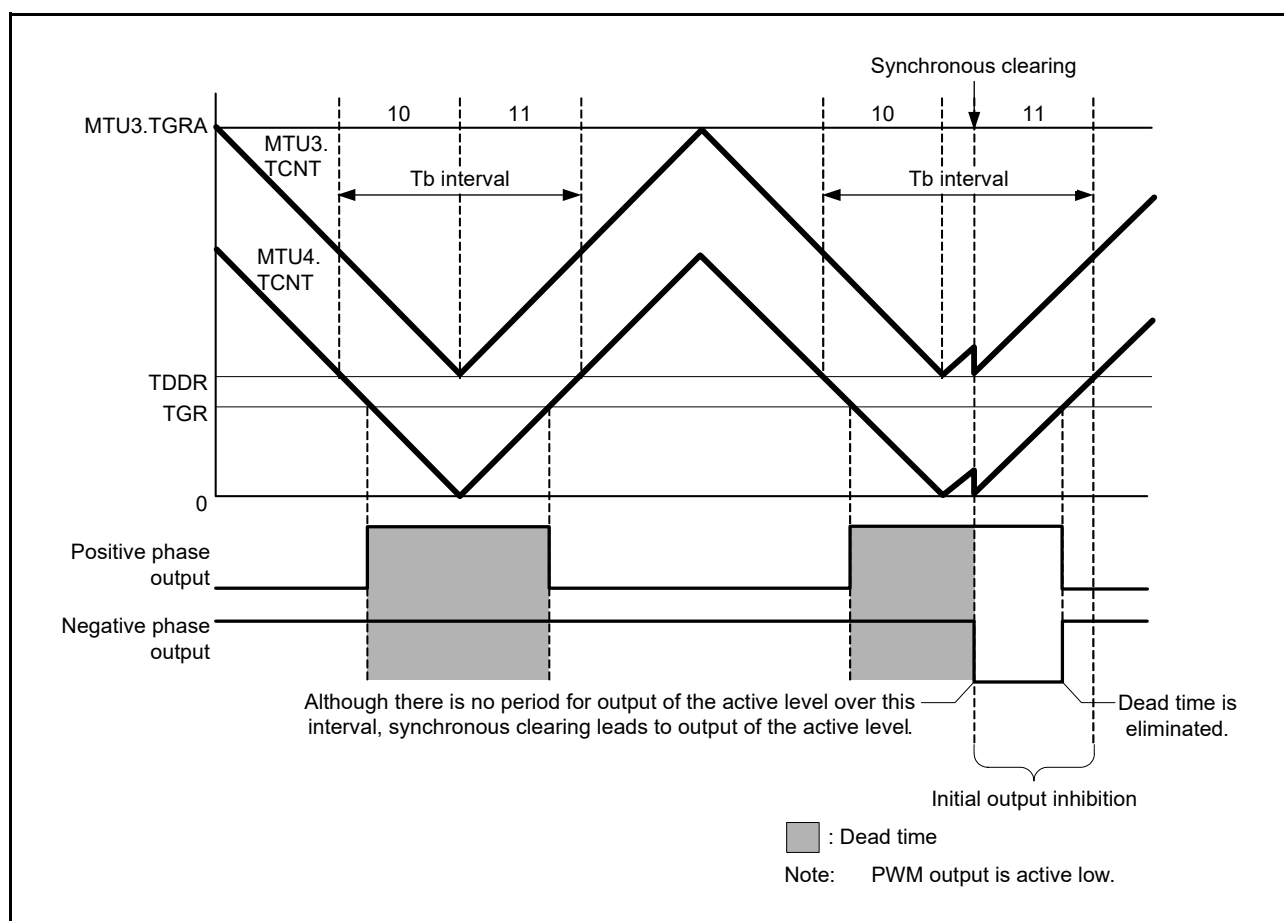


Figure 19.139 Example of Synchronous Clearing (When Condition 2 Applies)

19.6.26 Continuous Output of Interrupt Signal in Response to a Compare Match

When the TGR register is set to 0000h, the PCLKB/1 is set as the count clock, and compare match is set as the trigger for clearing of the count clock, the value of the TCNT counter remains 0000h, and the interrupt signal will be output continuously (i.e. its level will be flat) rather than output over a single cycle. Consequently, interrupts will not be detected in response to second and subsequent compare matches.

Figure 19.140 shows the timing for continuous output of the interrupt signal in response to a compare match.

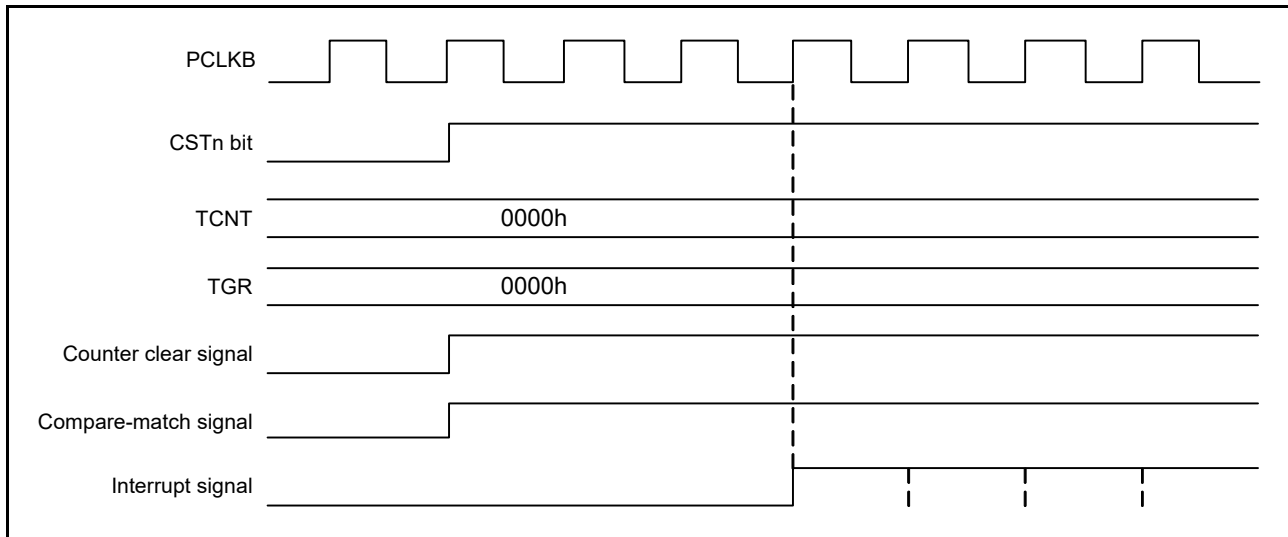


Figure 19.140 Continuous Output of Interrupt Signal in Response to a Compare Match

19.6.27 Usage Notes on A/D Conversion Start Request Delaying Function in Complementary PWM Mode

- When data is transferred from a buffer register at the trough of the MTU4.TCNT counter while the MTU4.TADCOBRA and MTU4.TADCOBRB registers are set to 0 and the UT4AE and UT4BE bits in the MTU4.TADCR register are set to 1, no A/D conversion start request is issued during up-counting immediately after transfer. Refer to Figure 19.141.
- When data is transferred from a buffer register at the crest of the MTU4.TCNT counter while the MTU4.TADCOBRA and MTU4.TADCOBRB registers are set to the same value as the TCDR and the UT4AE and UT4BE bits in the MTU4.TADCR register are set to 1, no A/D conversion start request is issued during down-counting immediately after transfer. Refer to Figure 19.142.
- To issue an A/D conversion start request linked with the interrupt skipping function, set the MTU4.TADCORA and MTU4.TADCORB registers so that $2 \leq \text{MTUn.TADCORA/TADCORB} \leq \text{TCDR} - 2$ is satisfied ($n = 4$).

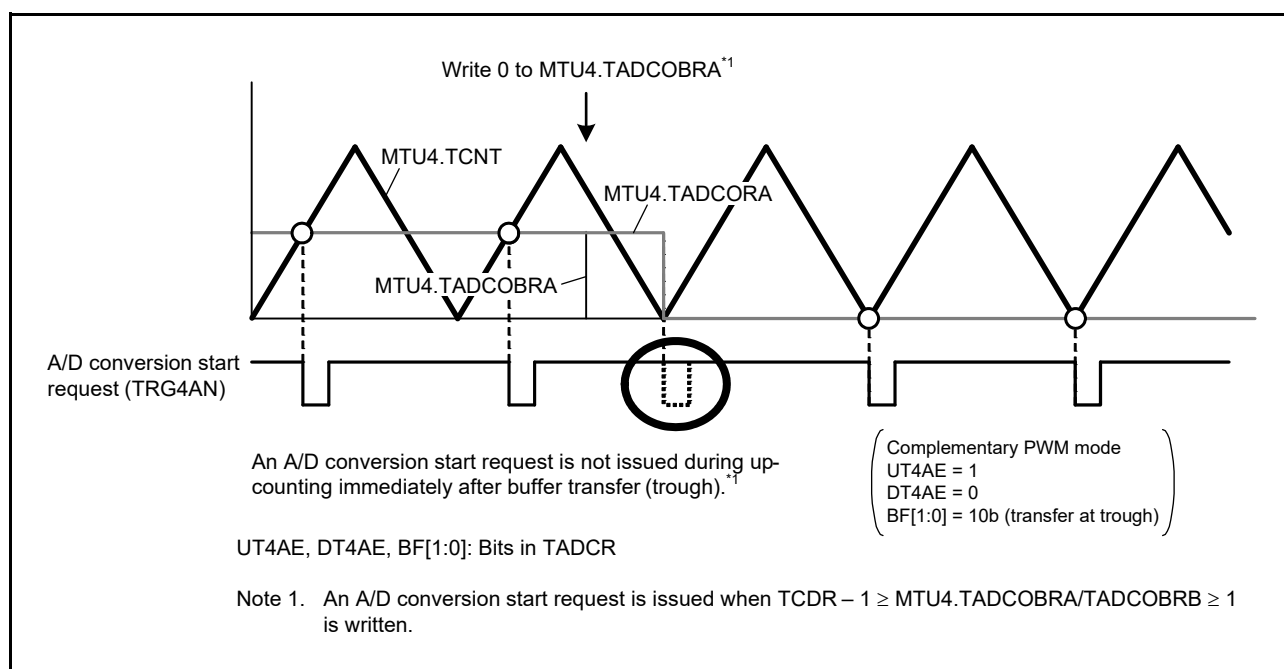


Figure 19.141 A/D Conversion Start Request When 0 is Written to MTU4.TADCOBRA

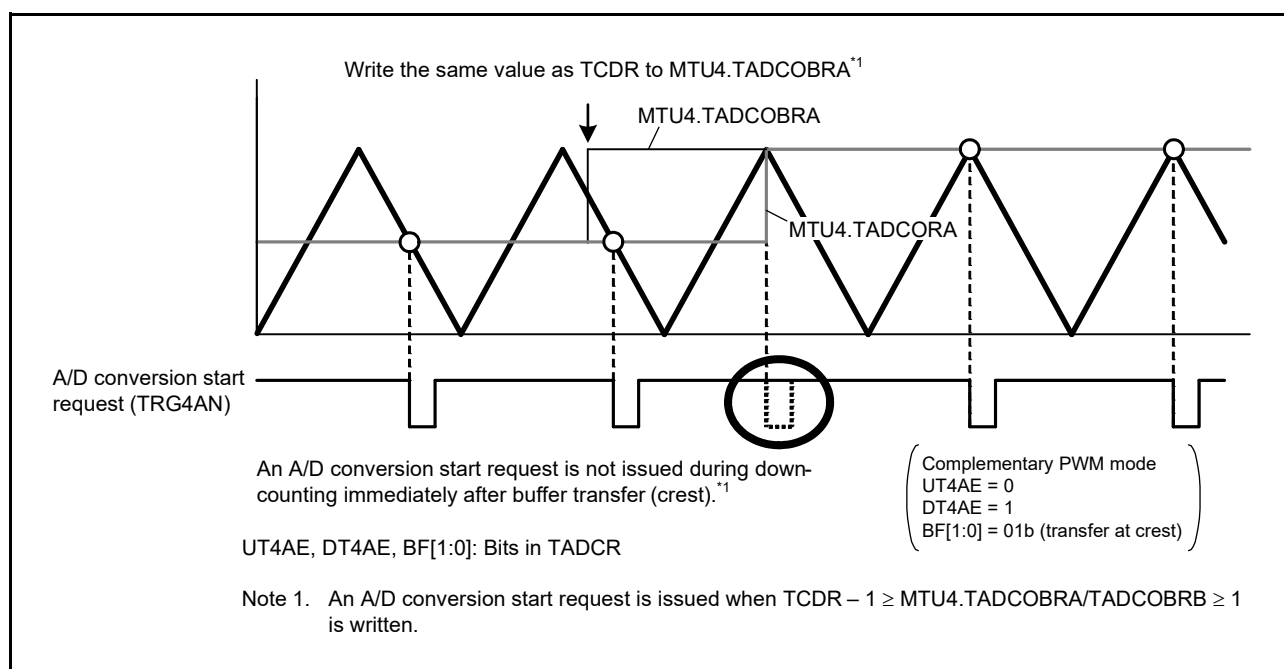


Figure 19.142 A/D Conversion Start Request When the Same Value as TCDR is Written to MTU4.TADCOBRA

19.7 MTU Output Pin Initialization

19.7.1 Operating Modes

The MTU has the following six operating modes. Waveforms can be output in any of these modes.

- Normal mode (MTU0 to MTU4)
- PWM mode 1 (MTU0 to MTU4)
- PWM mode 2 (MTU0 to MTU2)
- Phase counting modes 1 to 5 (MTU1 and MTU2)
- Complementary PWM mode (MTU3 and MTU4)
- Reset-synchronized PWM mode (MTU3 and MTU4)

This section describes how to initialize the MTU output pins in each of these modes.

19.7.2 Operation in Case of Re-Setting Due to Error during Operation

If an error occurs during MTU operation, MTU output should be cut off by the system. The output can be cut off by allowing non-active level output from the pins by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports. MTU output can be disabled through TIOR settings. Complementary PWM output (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C and MTIOC4D) should be specified through TOERA setting. For PWM output pins, output can also be cut by hardware, using port output enable 3 (POE3). The pin initialization procedures for re-setting due to an error during operation and the procedures for restarting in a different mode after re-setting are described below.

The MTU has six operating modes, as stated above. There are thus 36 mode transition combinations, but some transitions are not available with certain channel and mode combinations. Available mode transition combinations are shown in Table 19.64.

Table 19.64 Mode Transition Combinations

	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	Not available	Not available
PCM	(17)	(18)	(19)	(20)	Not available	Not available
CPWM	(21)	(22)	Not available	Not available	(23) (24)	(25)
RPWM	(26)	(27)	Not available	Not available	(28)	(29)

Normal: Normal mode
 PWM1: PWM mode 1
 PWM2: PWM mode 2
 PCM: Phase counting modes 1 to 5
 CPWM: Complementary PWM mode
 RPWM: Reset-synchronized PWM mode

19.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation

- When making a transition to a mode (Normal, PWM1, PWM2, or PCM) in which the pin output level is selected by the timer I/O control register (TIOR) setting, initialize the pins by means of TIOR setting.
- In PWM mode 1, waveforms are not output to the MTIOCnB and MTIOCnD ($n = 3, 4$) pins. When a pin is configured for MTIOCnB or MTIOCnD, it enters high-impedance state. To output a specified level, set the pin to general output port.
- In PWM mode 2, a waveform is not output to the pin that corresponds to the TGR register used as a period register. When a pin is configured for MTIOCnm ($n = 0$ to 2 ; $m = A$ to D), it enters high-impedance state. To output a specified level, set the pin to general output port.
- In normal mode or PWM mode 2, if TGRC and TGRD operate as buffer registers, waveforms are not output to the corresponding pins (MTIOCnC or MTIOCnD ($n = 0, 3, 4$)). When a pin is configured for MTIOCnC or MTIOCnD, it enters high-impedance state. To output a specified level, set the pin to general output port.
- In PWM mode 1, if either TGRC or TGRD operates as a buffer register, waveforms are not output to the corresponding pins (MTIOCnC or MTIOCnD ($n = 0, 3, 4$)). When a pin is configured for MTIOCnC or MTIOCnD, it enters high-impedance state. To output a specified level, set the pin to general output port.
- When making a transition to a mode (CPWM or RPWM) in which the pin output level is selected by the timer output control register (TOCR1A or TOCR2A) setting, temporarily disable output in MTU3 and MTU4 with the timer output master enable register (TOERA). At this time, when a pin is configured for MTIOCnm ($n = 3, 4$; $m = A$ to D), it enters high-impedance state. To output a specified level, set the pin to general output port. Switch to normal mode, perform initialization with TIOR, restore TIOR to its initial value, then operate the MTU in accordance with the mode setting procedure (TOCR1A setting, TOCR2A setting, TMDR1 setting, and TOERA setting).

Note: Channel number is substituted for “n” indicated in this section.

Pin initialization procedures are described below for the numbered combinations in Table 19.64. The active level is assumed to be low.

(1) Operation When Error Occurs in Normal Mode and Operation is Restarted in Normal Mode

Figure 19.143 shows a case in which an error occurs in normal mode and operation is restarted in normal mode after re-setting.

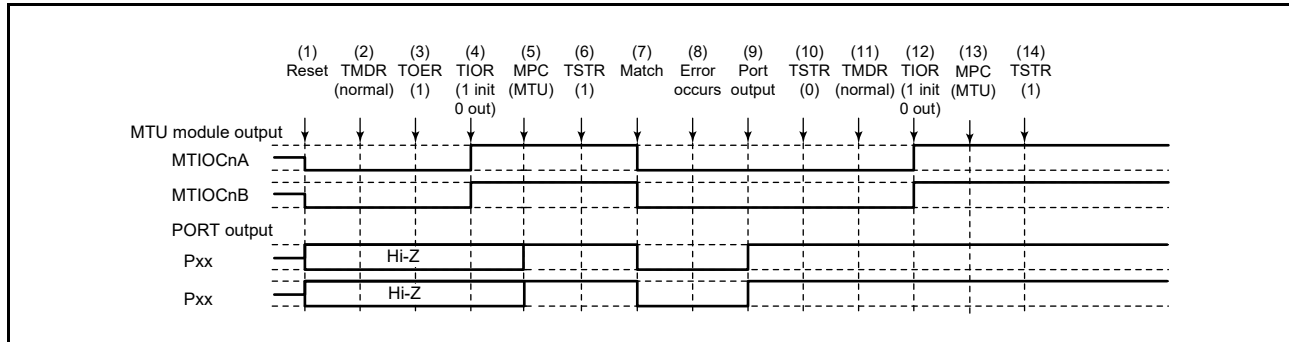


Figure 19.143 Error Occurrence in Normal Mode, Recovery in Normal Mode

- (1) After a reset, the MTU output goes low and the ports enter high-impedance state.
- (2) After a reset, the TMDR1 setting is for normal mode.
- (3) For MTU3 and MTU4, enable output with the TOERA register before initializing the pins with the TIOR register.
- (4) Initialize the pins with the TIOR register. (In the example, the initial output is a high level, and a low level is output on compare match occurrence.)
- (5) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (6) Start count operation by setting the TSTR register.
- (7) Output goes low on compare match occurrence.
- (8) An error occurs.
- (9) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- (10) Stop count operation by setting the TSTR register.
- (11) This step is not necessary when restarting in normal mode.
- (12) Initialize the pins with the TIOR register.
- (13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (14) Restart operation by setting the TSTR register.

(2) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM Mode 1

Figure 19.144 shows a case in which an error occurs in normal mode and operation is restarted in PWM mode 1 after re-setting.

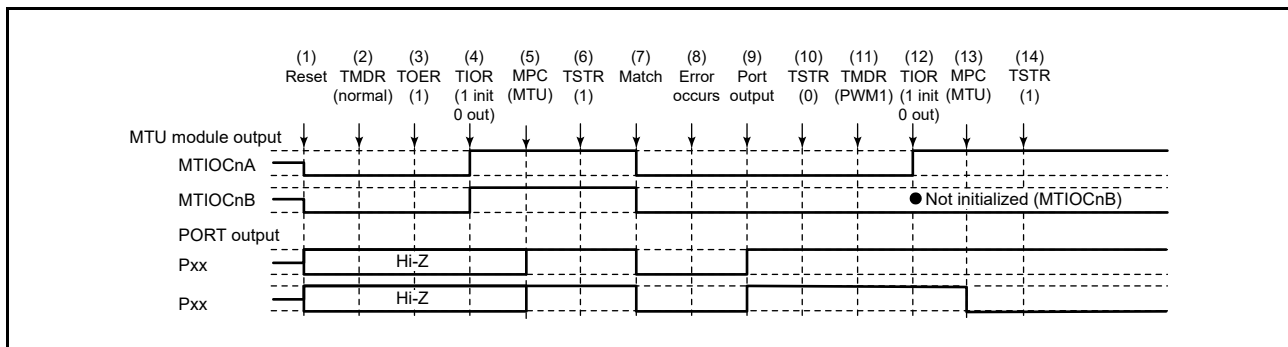


Figure 19.144 Error Occurrence in Normal Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in Figure 19.143.

(11) Set PWM mode 1.

(12) Initialize the pins with the TIOR register. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(14) Restart operation by setting the TSTRA register.

(3) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM mode 2

Figure 19.145 shows a case in which an error occurs in normal mode and operation is restarted in PWM mode 2 after re-setting.

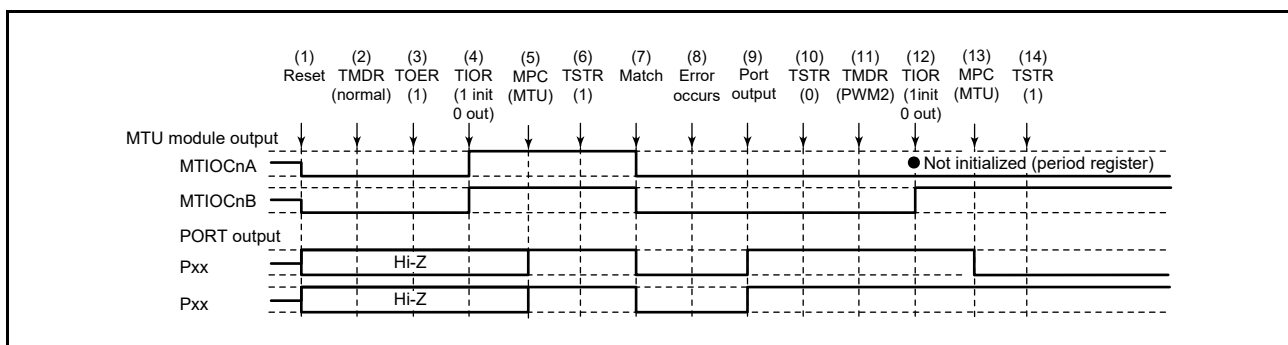


Figure 19.145 Error Occurrence in Normal Mode, Recovery in PWM Mode 2

(1) to (10) are the same as in Figure 19.143.

(11) Set PWM mode 2.

(12) Initialize the pins with the TIOR register. (In PWM mode 2, a waveform is not output to the pin that corresponds to the TGR register used as a period register. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(14) Restart operation by setting the TSTRA register.

Note: PWM mode 2 can only be selected for MTU0 to MTU2, and therefore the TOERA register setting is not necessary.

(4) Operation When Error Occurs in Normal Mode and Operation is Restarted in Phase Counting Mode

Figure 19.146 shows a case in which an error occurs in normal mode and operation is restarted in phase counting mode after re-setting.

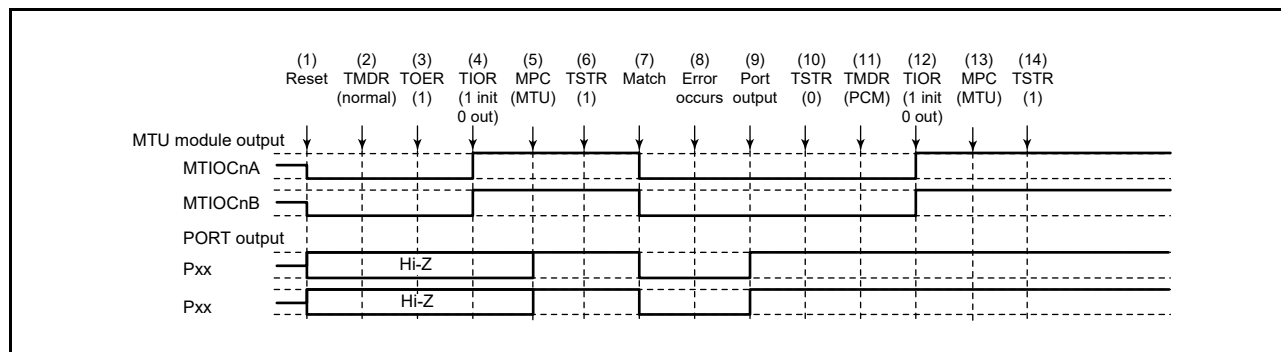


Figure 19.146 Error Occurrence in Normal Mode, Recovery in Phase Counting Mode

(1) to (10) are the same as in Figure 19.143.

(11) Set the phase counting mode.

(12) Initialize the pins with the TIOR register.

(13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(14) Restart operation by setting the TSTR register.

Note: The phase counting mode can only be selected for MTU1 and MTU2, and therefore the TOERA register setting is not necessary.

(5) Operation When Error Occurs in Normal Mode and Operation is Restarted in Complementary PWM Mode

Figure 19.147 shows a case in which an error occurs in normal mode and operation is restarted in complementary PWM mode after re-setting.

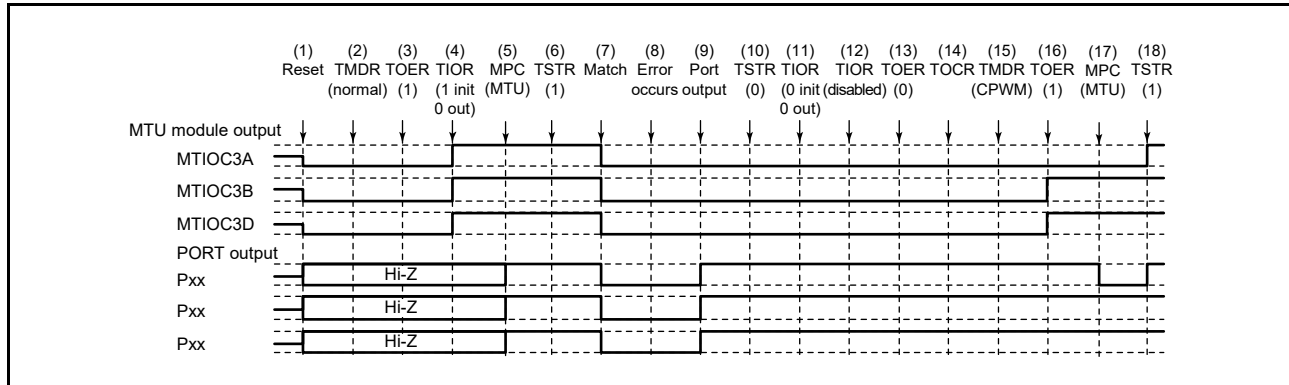


Figure 19.147 Error Occurrence in Normal Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in Figure 19.143.

(11) Initialize the normal mode waveform generation block with the TIOR register.

(12) Disable operation of the normal mode waveform generation block with the TIOR register.

(13) Disable output in MTU3 and MTU4 with the TOERA register.

(14) Select the complementary PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.

(15) Set complementary PWM mode.

(16) Enable output in MTU3 and MTU4 with the TOERA register.

(17) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(18) Restart operation by setting the TSTR register.

(6) Operation When Error Occurs in Normal Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 19.148 shows a case in which an error occurs in normal mode and operation is restarted in reset-synchronized PWM mode after re-setting.

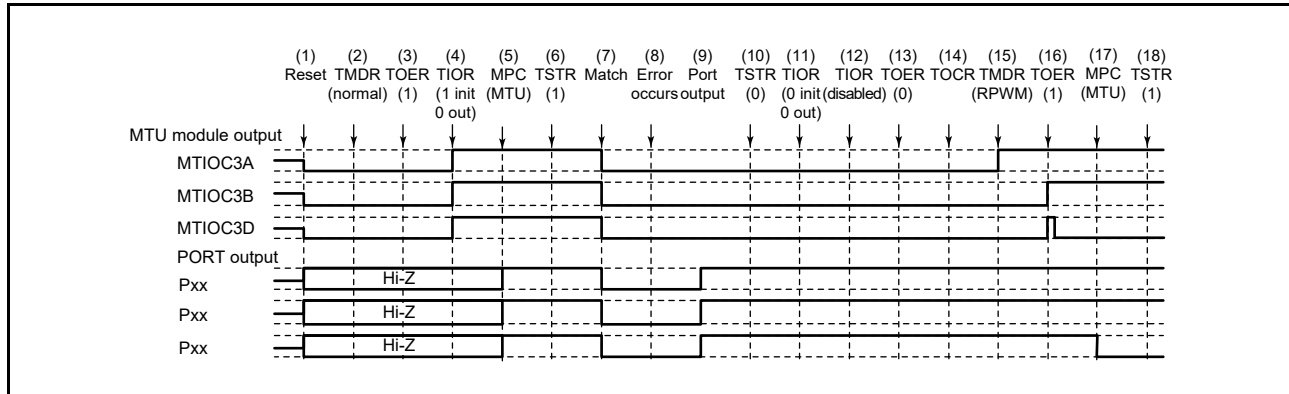


Figure 19.148 Error Occurrence in Normal Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (13) are the same as in Figure 19.147.

(14) Select the reset-synchronized PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.

(15) Set reset-synchronized PWM mode.

(16) Enable output in MTU3 and MTU4 with the TOERA register.

(17) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(18) Restart operation by setting the TSTR register.

(7) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Normal Mode

Figure 19.149 shows a case in which an error occurs in PWM mode 1 and operation is restarted in normal mode after re-setting.

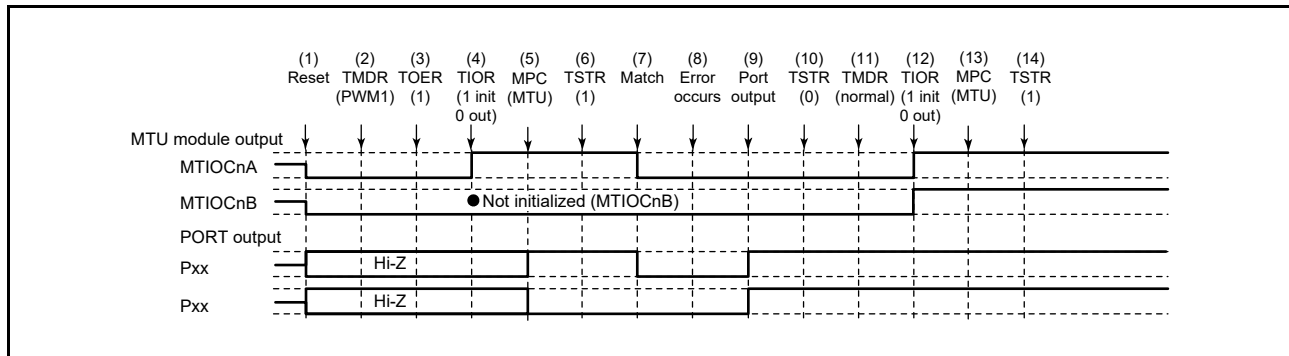


Figure 19.149 Error Occurrence in PWM Mode 1, Recovery in Normal Mode

- (1) After a reset, the MTU output goes low and the ports enter high-impedance state.
- (2) Set PWM mode 1.
- (3) For MTU3 and MTU4, enable output with the TOERA register before initializing the pins with the TIOR register.
- (4) Initialize the pins with the TIOR register. (In the example, the initial output is a high level, and a low level is output on compare match occurrence. In PWM mode 1, the MTIOcNB side is not initialized.)
- (5) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (6) Start count operation by setting the TSTR register.
- (7) Output goes low on compare match occurrence.
- (8) An error occurs.
- (9) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- (10) Stop count operation by setting the TSTR register.
- (11) Set normal mode.
- (12) Initialize the pins with the TIOR register.
- (13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (14) Restart operation by setting the TSTR register.

(8) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 1

Figure 19.150 shows a case in which an error occurs in PWM mode 1 and operation is restarted in PWM mode 1 after re-setting.

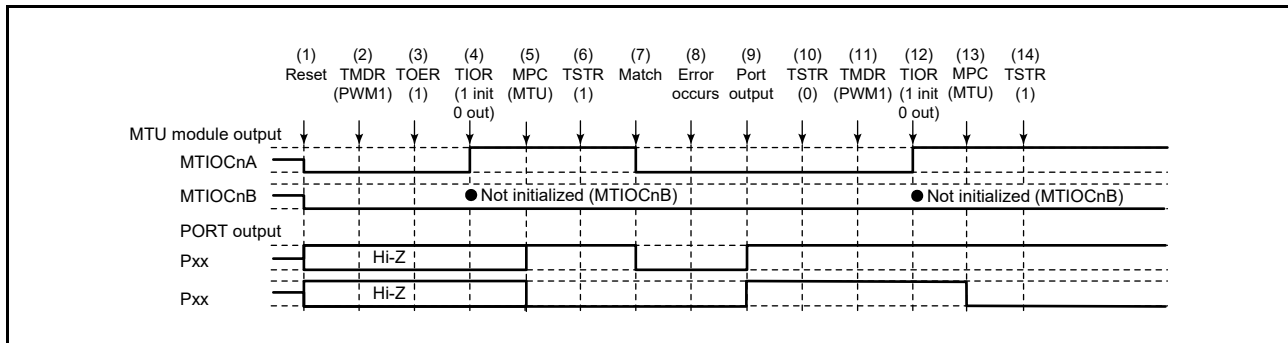


Figure 19.150 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1

(1) to (10) are the same as in Figure 19.149.

(11) This step is not necessary when restarting in PWM mode 1.

(12) Initialize the pins with the TIOR register. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(14) Restart operation by setting the TSTRA register.

(9) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 2

Figure 19.151 shows a case in which an error occurs in PWM mode 1 and operation is restarted in PWM mode 2 after re-setting.

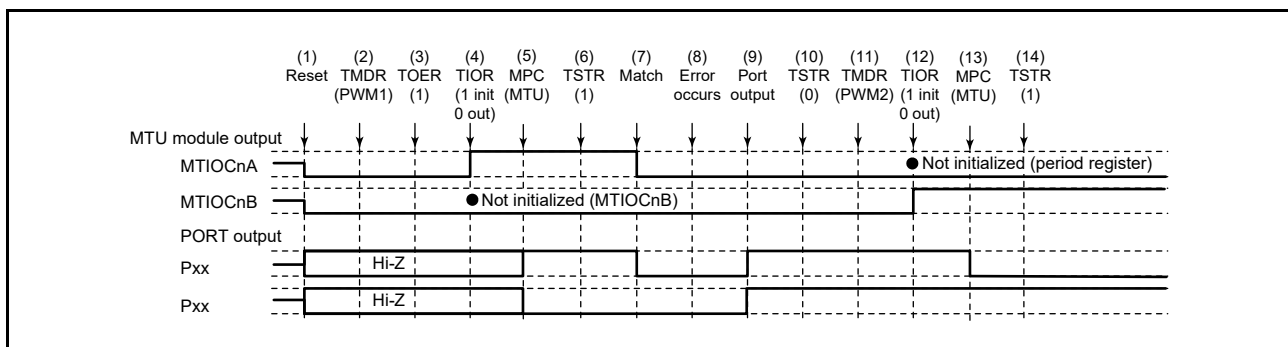


Figure 19.151 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

(1) to (10) are the same as in Figure 19.149.

(11) Set PWM mode 2.

(12) Initialize the pins with the TIOR register. (In PWM mode 2, a waveform is not output to the pin that corresponds to the TGR register used as a period register. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(14) Restart operation by setting the TSTRA register.

Note: PWM mode 2 can only be selected for MTU0 to MTU2, and therefore the TOERA register setting is not necessary.

(10) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Phase Counting Mode

Figure 19.152 shows a case in which an error occurs in PWM mode 1 and operation is restarted in phase counting mode after re-setting.

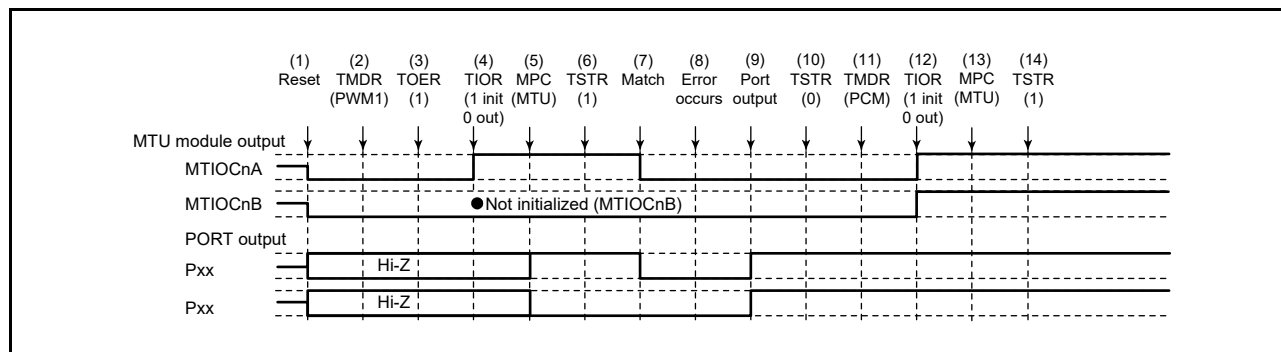


Figure 19.152 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode

(1) to (10) are the same as in Figure 19.149.

(11) Set the phase counting mode.

(12) Initialize the pins with the TIOR register.

(13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(14) Restart operation by setting the TSTR register.

Note: The phase counting mode can only be selected for MTU1 and MTU2, and therefore the TOERA register setting is not necessary.

(11) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Complementary PWM Mode

Figure 19.153 shows a case in which an error occurs in PWM mode 1 and operation is restarted in complementary PWM mode after re-setting.

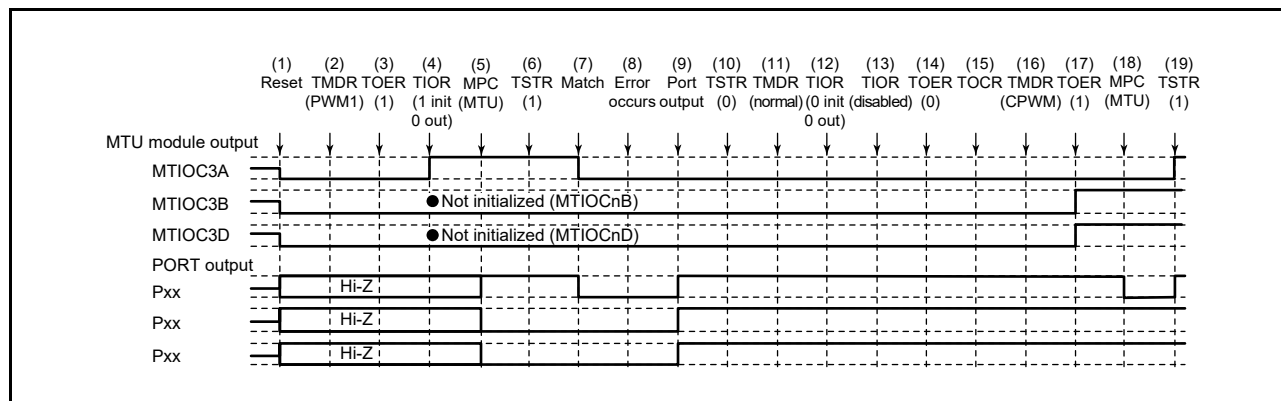


Figure 19.153 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode

(1) to (10) are the same as in Figure 19.149.

(11) Set normal mode to initialize the normal mode waveform generation block.

(12) Initialize the PWM mode 1 waveform generation block with the TIOR register.

(13) Disable operation of the PWM mode 1 waveform generation block with the TIOR register.

(14) Disable output in MTU3 and MTU4 with the TOERA register.

(15) Select the complementary PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.

(16) Set complementary PWM mode.

(17) Enable output in MTU3 and MTU4 with the TOERA register.

(18) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(19) Restart operation by setting the TSTR register.

(12) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 19.154 shows a case in which an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.

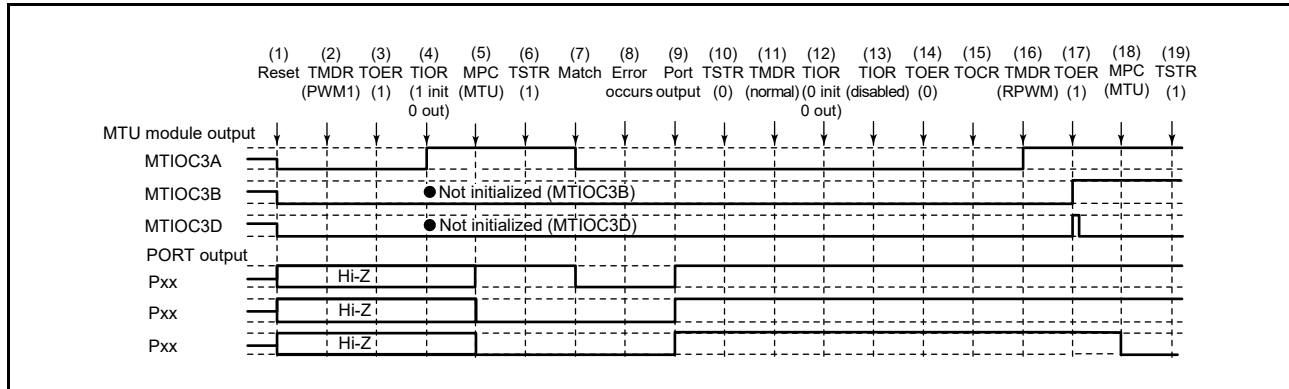


Figure 19.154 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

(1) to (14) are the same as in Figure 19.153.

(15) Select the reset-synchronized PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.

(16) Set reset-synchronized PWM mode.

(17) Enable output in MTU3 and MTU4 with the TOERA register.

(18) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(19) Restart operation by setting the TSTRA register.

(13) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Normal Mode

Figure 19.155 shows a case in which an error occurs in PWM mode 2 and operation is restarted in normal mode after re-setting.

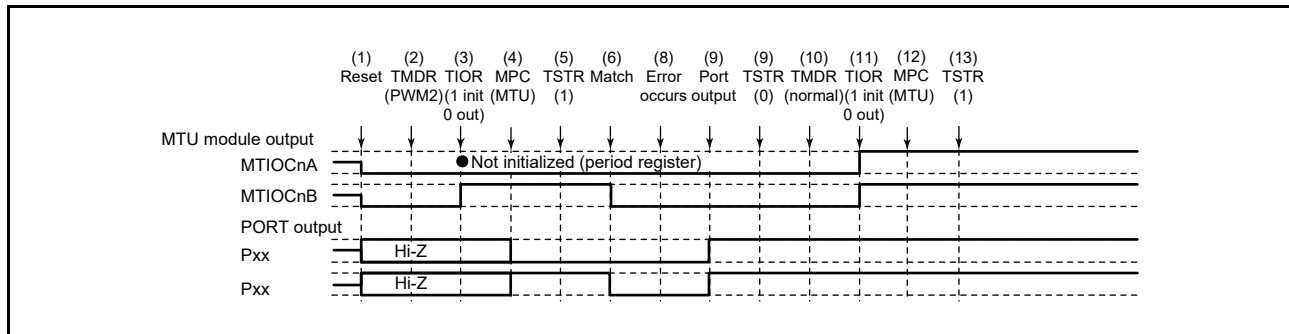


Figure 19.155 Error Occurrence in PWM Mode 2, Recovery in Normal Mode

- (1) After a reset, the MTU output goes low and the ports enter high-impedance state.
- (2) Set PWM mode 2.
- (3) Initialize the pins with the TIOR register. (In the example, the initial output is a high level, and a low level is output on compare match occurrence. In PWM mode 2, the pin that corresponds to the TGR register used as a period register is not initialized. In the example, the MTUUn.TGRA register is used as a period register.)
- (4) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (5) Start count operation by setting the TSTRA register.
- (6) Output goes low on compare match occurrence.
- (7) An error occurs.
- (8) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- (9) Stop count operation by setting the TSTRA register.
- (10) Set normal mode.
- (11) Initialize the pins with the TIOR register.
- (12) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (13) Restart operation by setting the TSTRA register.

(14) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 1

Figure 19.156 shows a case in which an error occurs in PWM mode 2 and operation is restarted in PWM mode 1 after re-setting.

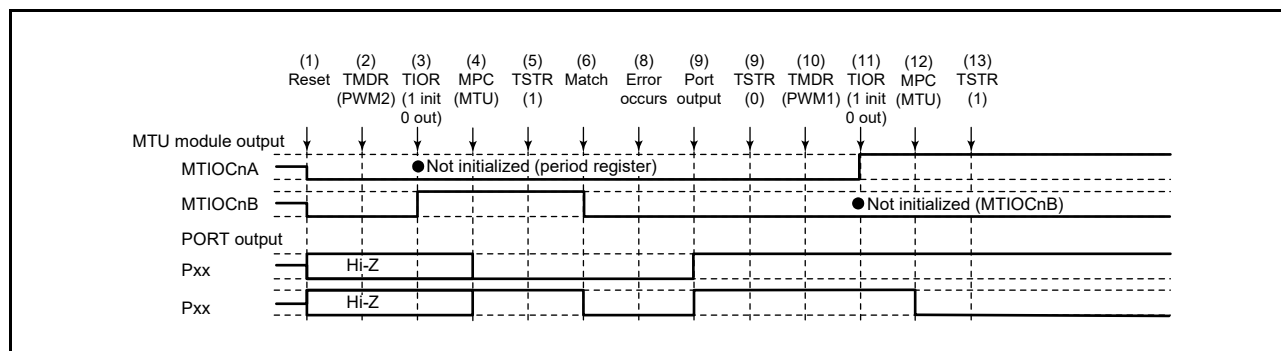


Figure 19.156 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1

(1) to (9) are the same as in Figure 19.155.

(10) Set PWM mode 1.

(11) Initialize the pins with the TIOR register. (In PWM mode 1, waveforms are not output to the MTIOcNB (MTIOcND) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(12) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(13) Restart operation by setting the TSTRA register.

(15) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 2

Figure 19.157 shows a case in which an error occurs in PWM mode 2 and operation is restarted in PWM mode 2 after re-setting.

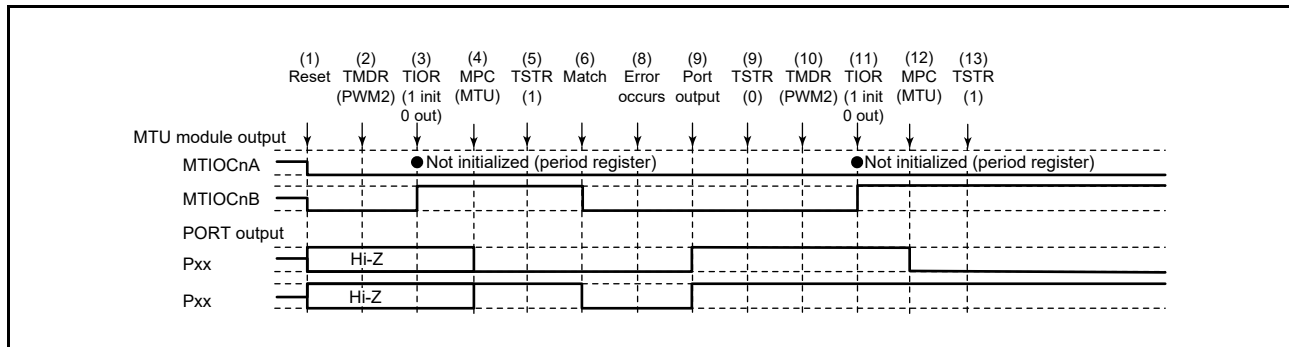


Figure 19.157 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2

(1) to (9) are the same as in Figure 19.155.

(10) This step is not necessary when restarting in PWM mode 2.

(11) Initialize the pins with the TIOR register. (In PWM mode 2, a waveform is not output to the pin that corresponds to the TGR register used as a period register. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(12) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(13) Restart operation by setting the TSTR register.

(16) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Phase Counting Mode

Figure 19.158 shows a case in which an error occurs in PWM mode 2 and operation is restarted in phase counting mode after re-setting.

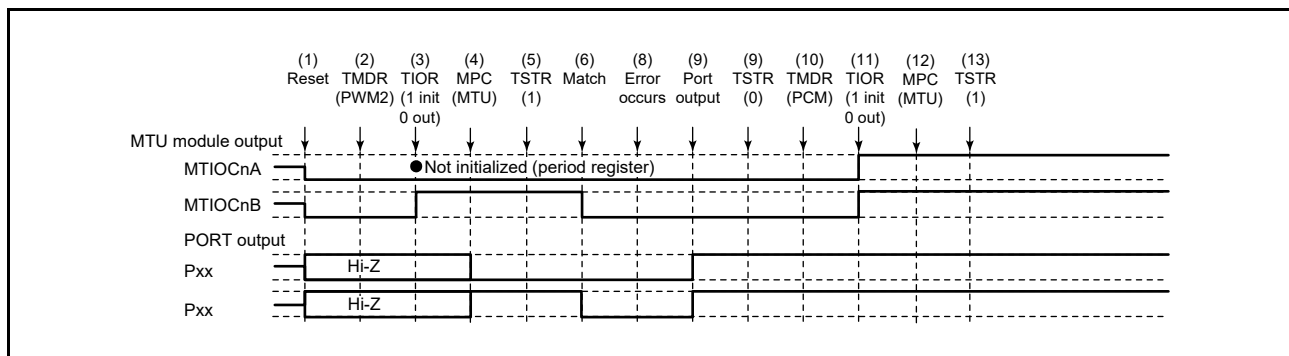


Figure 19.158 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

(1) to (9) are the same as in Figure 19.155.

(10) Set the phase counting mode.

(11) Initialize the pins with the TIOR register.

(12) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(13) Restart operation by setting the TSTR register.

(17) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in Normal Mode

Figure 19.159 shows a case in which an error occurs in phase counting mode and operation is restarted in normal mode after re-setting.

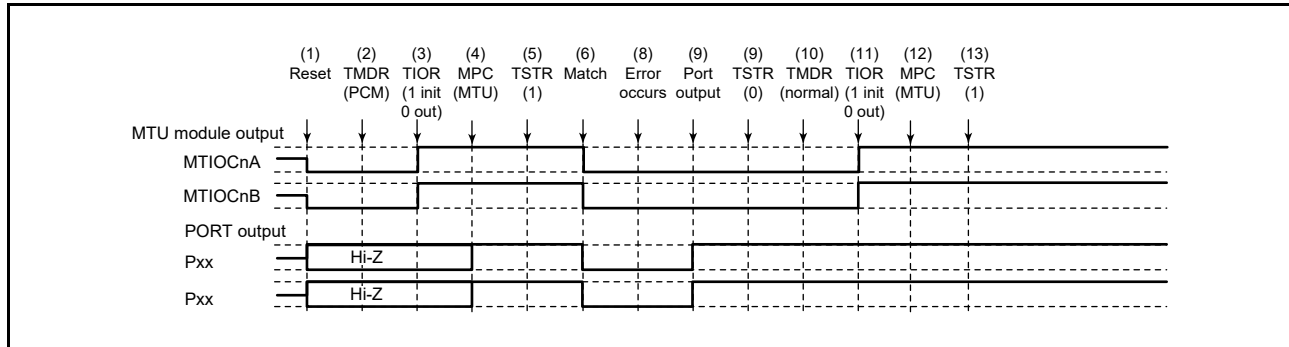


Figure 19.159 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode

- (1) After a reset, the MTU output goes low and the ports enter high-impedance state.
- (2) Set phase counting mode.
- (3) Initialize the pins with the TIOR register. (In the example, the initial output is a high level, and a low level is output on compare match occurrence.)
- (4) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (5) Start count operation by setting the TSTR register.
- (6) Output goes low on compare match occurrence.
- (7) An error occurs.
- (8) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- (9) Stop count operation by setting the TSTR register.
- (10) Set normal mode.
- (11) Initialize the pins with the TIOR register.
- (12) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (13) Restart operation by setting the TSTR register.

(18) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 1

Figure 19.160 shows a case in which an error occurs in phase counting mode and operation is restarted in PWM mode 1 after re-setting.

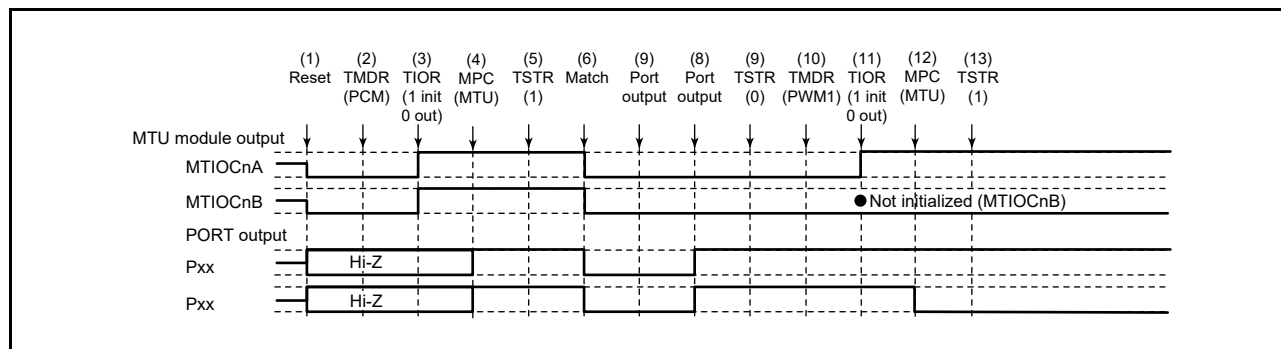


Figure 19.160 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1

(1) to (9) are the same as in Figure 19.159.

(10) Set PWM mode 1.

(11) Initialize the pins with the TIOR register. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(12) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(13) Restart operation by setting the TSTR register.

(19) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 2

Figure 19.161 shows a case in which an error occurs in phase counting mode and operation is restarted in PWM mode 2 after re-setting.

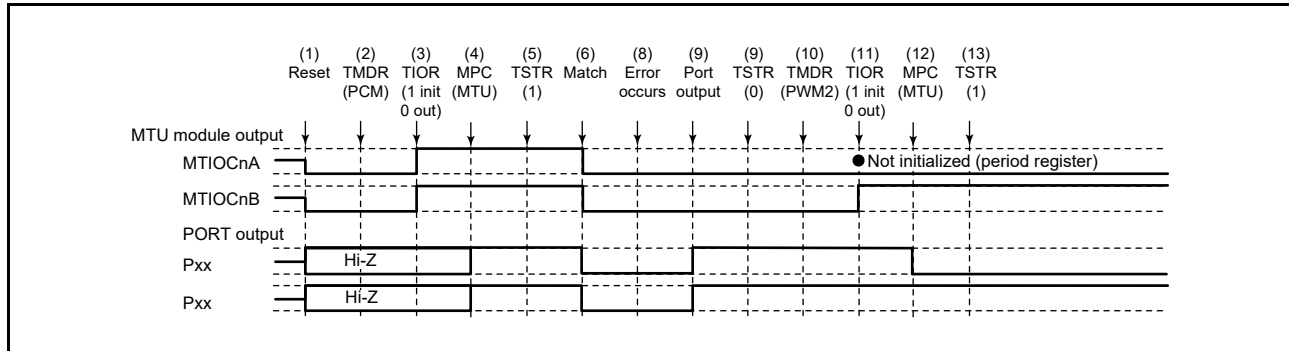


Figure 19.161 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2

(1) to (9) are the same as in Figure 19.159.

(10) Set PWM mode 2.

(11) Initialize the pins with the TIOR register. (In PWM mode 2, a waveform is not output to the pin that corresponds to the TGR register used as a period register. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(12) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(13) Restart operation by setting the TSTR register.

(20) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in Phase Counting Mode

Figure 19.162 shows a case in which an error occurs in phase counting mode and operation is restarted in phase counting mode after re-setting.

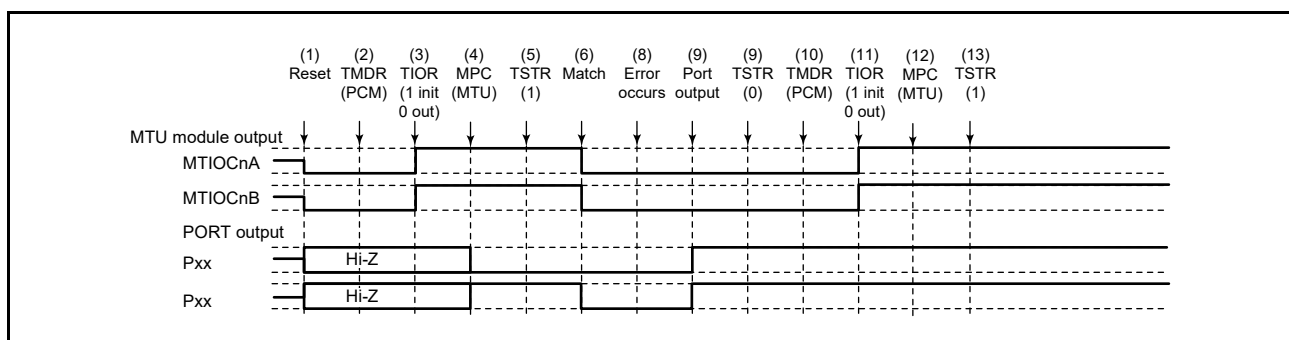


Figure 19.162 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode

(1) to (9) are the same as in Figure 19.159.

(10) This step is not necessary when restarting in phase counting mode.

(11) Initialize the pins with the TIOR register.

(12) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(13) Restart operation by setting the TSTR register.

(21) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Normal Mode

Figure 19.163 shows a case in which an error occurs in complementary PWM mode and operation is restarted in normal mode after re-setting.

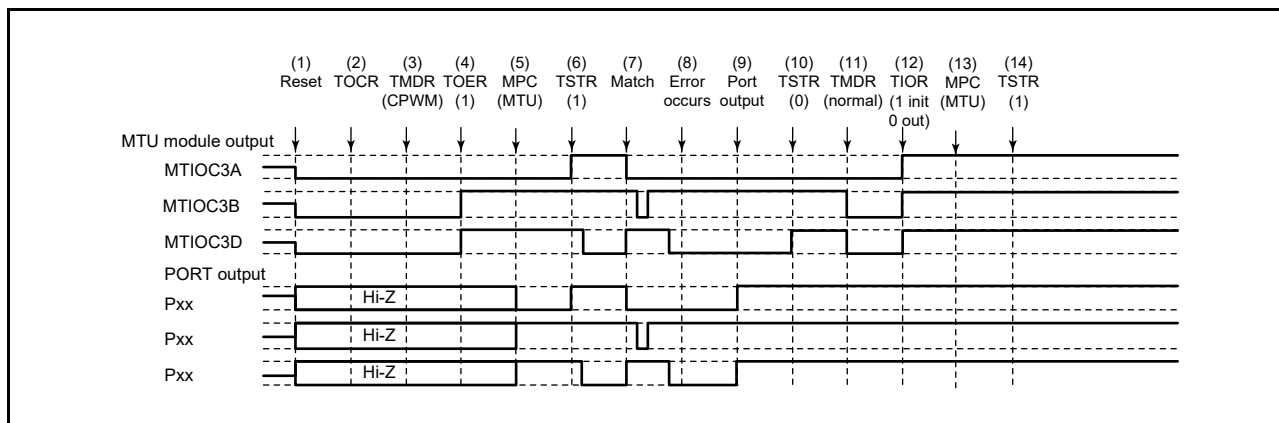


Figure 19.163 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode

- (1) After a reset, the MTU3 output goes low and the ports enter high-impedance state.
- (2) Select the complementary PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.
- (3) Set complementary PWM mode.
- (4) Enable output in MTU3 and MTU4 with the TOERA register.
- (5) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (6) Start count operation by setting the TSTRA register.
- (7) The complementary PWM waveform is output on compare match occurrence.
- (8) An error occurs.
- (9) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- (10) Stop count operation by setting the TSTRA register. (MTU output becomes the initial complementary PWM output value).
- (11) Set normal mode (MTU output goes low).
- (12) Initialize the pins with the TIOR register.
- (13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (14) Restart operation by setting the TSTRA register.

(22) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in PWM Mode 1

Figure 19.164 shows a case in which an error occurs in complementary PWM mode and operation is restarted in PWM mode 1 after re-setting.

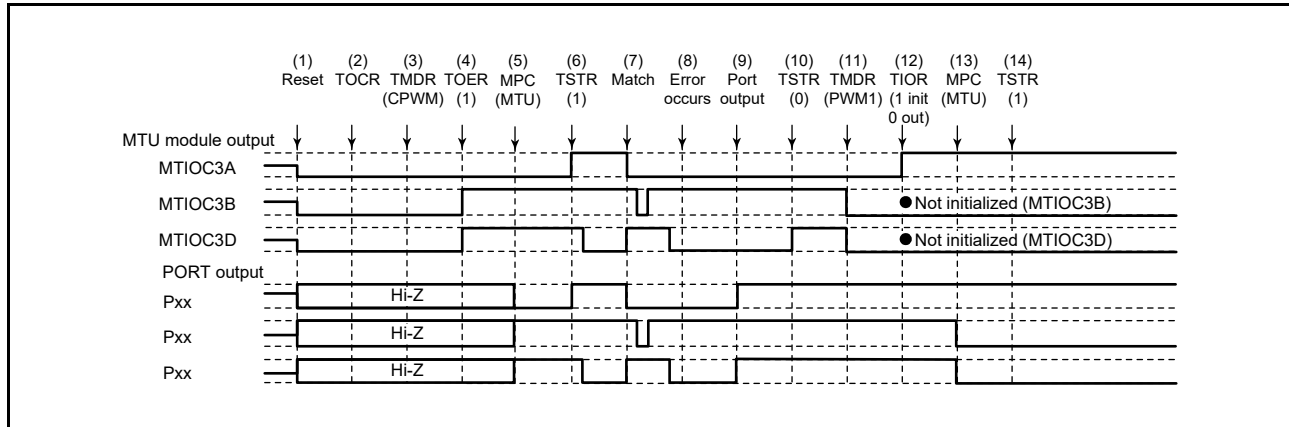


Figure 19.164 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1

- (1) to (10) are the same as in Figure 19.163.
- (11) Set PWM mode 1 (MTU output goes low).
- (12) Initialize the pins with the TIOR register. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)
- (13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (14) Restart operation by setting the TSTR register.

(23) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode

Figure 19.165 shows a case in which an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using the period and duty settings at the time of stopping the counter).

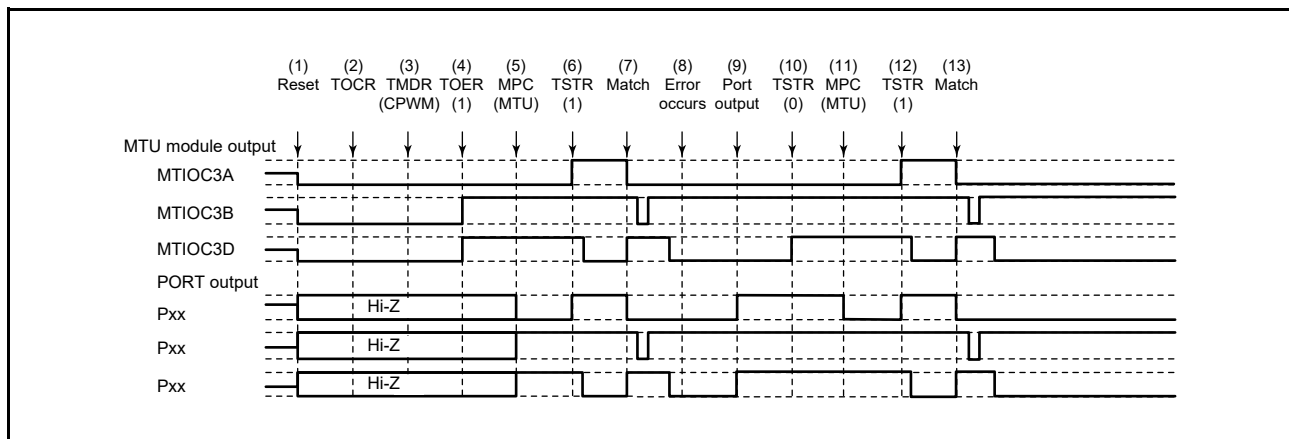


Figure 19.165 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode (1)

(1) to (10) are the same as in Figure 19.163.

(11) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(12) Restart operation by setting the TSTR register.

(13) The complementary PWM waveform is output on compare match occurrence.

(24) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode with New Settings

Figure 19.166 shows a case in which an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (operation is restarted using new period and duty ratio settings).

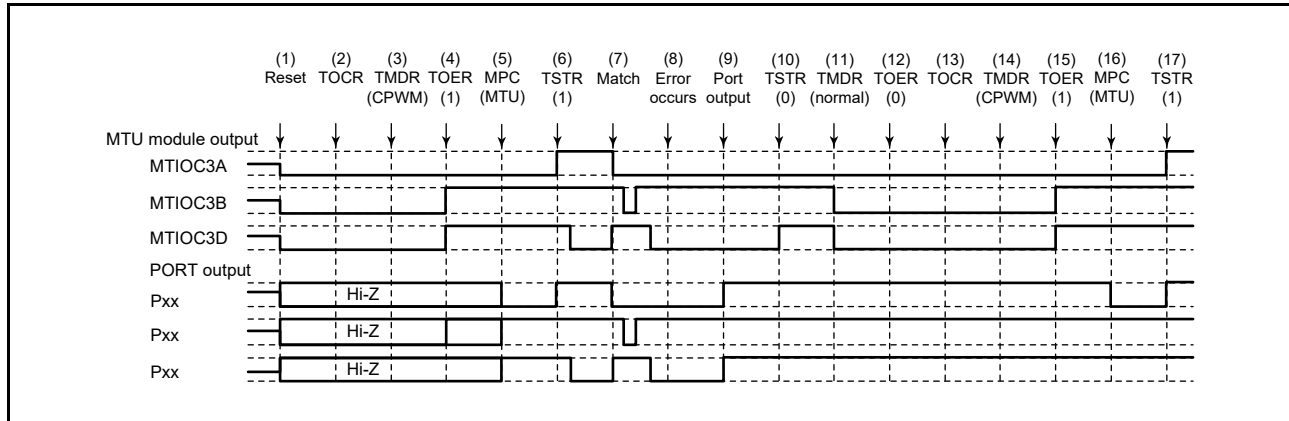


Figure 19.166 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode (2)

- (1) to (10) are the same as in Figure 19.163.
- (11) Set normal mode and make new settings (MTU output goes low).
- (12) Disable output in MTU3 and MTU4 with the TOERA register.
- (13) Select the complementary PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.
- (14) Set complementary PWM mode.
- (15) Enable output in MTU3 and MTU4 with the TOERA register.
- (16) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (17) Restart operation by setting the TSTR register.

(25) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 19.167 shows a case in which an error occurs in complementary PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

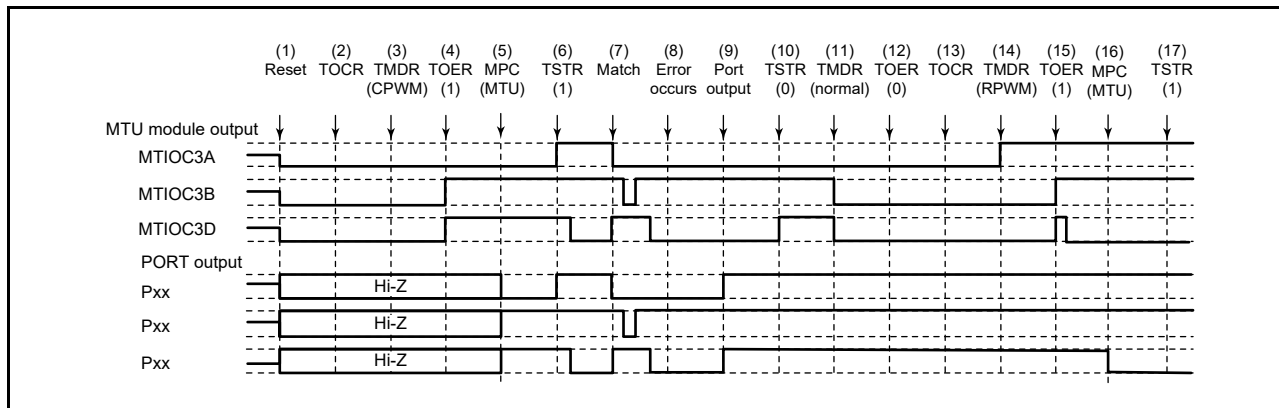


Figure 19.167 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode

- (1) to (10) are the same as in Figure 19.163.
- (11) Set normal mode (MTU output goes low).
- (12) Disable output in MTU3 and MTU4 with the TOERA register.
- (13) Select the reset-synchronized PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.
- (14) Set reset-synchronized PWM mode.
- (15) Enable output in MTU3 and MTU4 with the TOERA register.
- (16) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (17) Restart operation by setting the TSTR register.

(26) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Normal Mode

Figure 19.168 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in normal mode after re-setting.

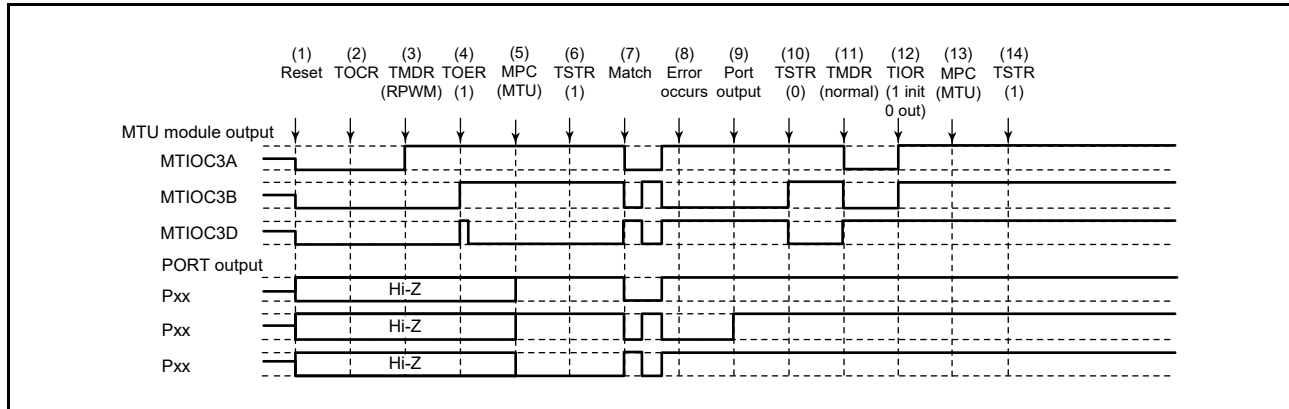


Figure 19.168 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Normal Mode

- (1) After a reset, the MTU output goes low and the ports enter high-impedance state.
- (2) Select the reset-synchronized PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.
- (3) Set reset-synchronized PWM mode.
- (4) Enable output in MTU3 and MTU4 with the TOERA register.
- (5) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (6) Start count operation by setting the TSTR register.
- (7) The reset-synchronized PWM waveform is output on compare match occurrence.
- (8) An error occurs.
- (9) Allow non-active level output by setting the pins as general output ports using the port direction registers (PDR), port output data registers (PODR), and port mode registers (PMR) of the I/O ports.
- (10) Stop count operation by setting the TSTR register. (MTU output becomes the initial reset-synchronized PWM output value.)
- (11) Set normal mode (positive-phase MTU output goes low, and negative-phase output goes high).
- (12) Initialize the pins with the TIOR register.
- (13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.
- (14) Restart operation by setting the TSTR register.

(27) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in PWM Mode 1

Figure 19.169 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in PWM mode 1 after re-setting.

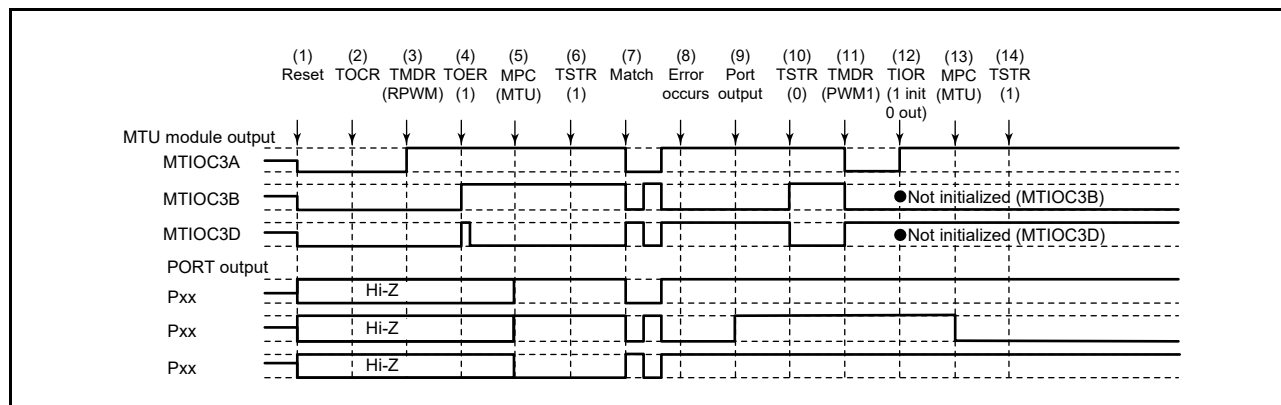


Figure 19.169 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in Figure 19.168.

(11) Set PWM mode 1 (positive-phase MTU output goes low, and negative-phase output goes high).

(12) Initialize the pins with the TIOR register. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port direction registers (PDR) and port output data registers (PODR) of the I/O ports.)

(13) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(14) Restart operation by setting the TSTRA register.

(28) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Complementary PWM Mode

Figure 19.170 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in complementary PWM mode after re-setting.

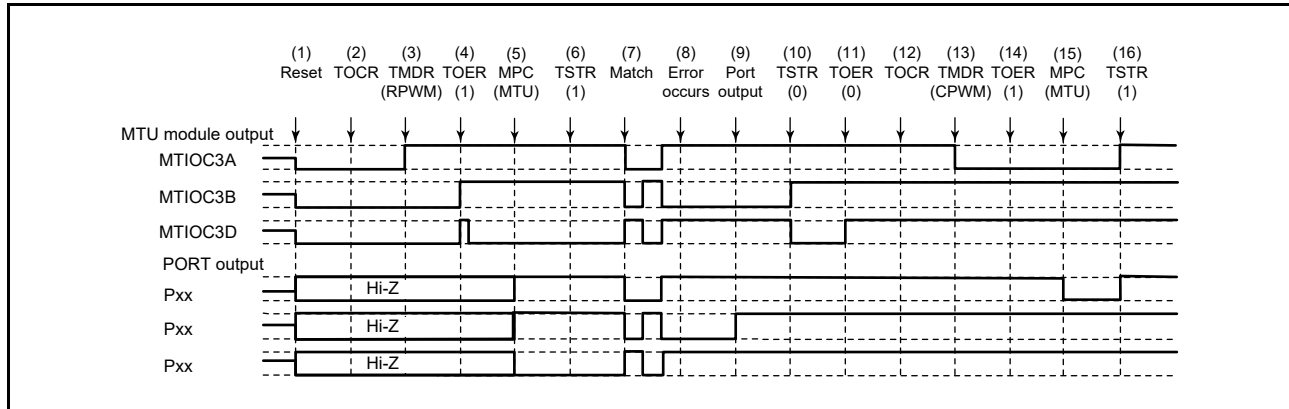


Figure 19.170 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in Figure 19.168.

(11) Disable output in MTU3 and MTU4 with the TOERA register.

(12) Select the complementary PWM output level and enable or disable cyclic output with registers TOCR1A and TOCR2A.

(13) Set complementary PWM mode (MTU cyclic output pin goes low).

(14) Enable output in MTU3 and MTU4 with the TOERA register.

(15) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(16) Restart operation by setting the TSTR register.

(29) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 19.171 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

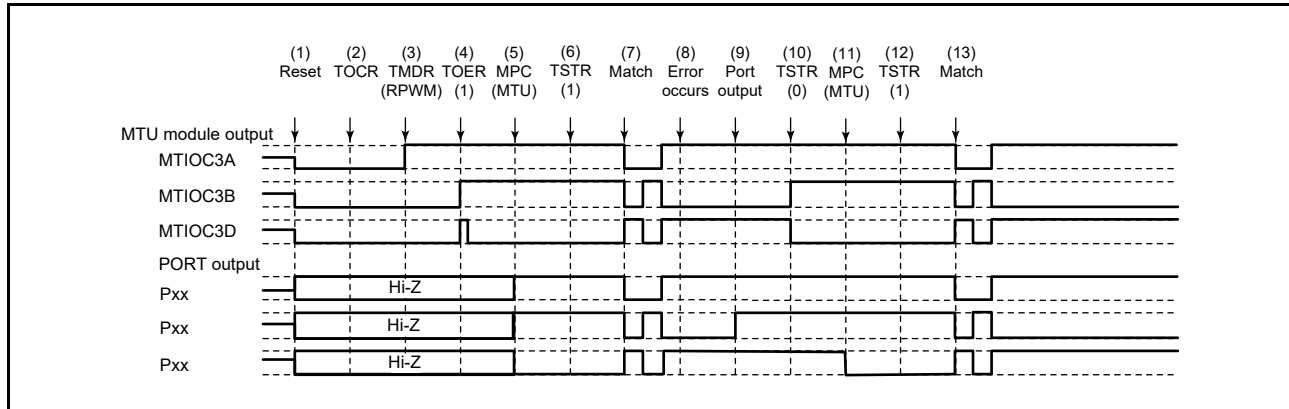


Figure 19.171 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in Figure 19.168.

(11) Set MTU output using the MPC and port mode registers (PMR) corresponding to the I/O ports.

(12) Restart operation by setting the TSTR register.

(13) The reset-synchronized PWM waveform is output on compare match occurrence.

20. Port Output Enable 3 (POE3E)

This MCU incorporates a port output enable 3 (POE3E) which can be used to, under various conditions, disable output signals for the MTU and the GPTW. The output status can be selected from high-impedance or general I/O port while the output is disabled.

In this section, “PCLK” is used to refer to PCLKB.

20.1 Overview

Table 20.1 lists the specifications of the POE. Figure 20.1 and Figure 20.2 show a system block diagram around POE and a block diagram of POE, respectively.

Table 20.1 POE Specifications

Item	Description																
Pin status while output is disabled	• High-impedance • General I/O port																
Target pins for switching to disabling of signal output	• MTU output pins - MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D) - MTU3 pins (MTIOC3B, MTIOC3D) - MTU4 pins (MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D) • GPTW output pins - GPTW0 pins (GTIOC0A, GTIOC0B) - GPTW1 pins (GTIOC1A, GTIOC1B) - GPTW2 pins (GTIOC2A, GTIOC2B)																
Generating conditions of request for switching to disable output	• Input signal detection: Detection of the POE0#, POE8#, POE10#, POE11#, and POE12# signal level. • Simultaneous conduction between output signals: When the output signal levels (active levels) of the following combinations match for one cycle or more (simultaneous conduction). <table><tr><td></td><td>MTU Complementary PWM Output Signals</td><td></td><td>GPTW Complementary PWM Output Signals</td></tr><tr><td>1</td><td>MTIOC3B and MTIOC3D</td><td>1</td><td>GTIOC0A and GTIOC0B</td></tr><tr><td>2</td><td>MTIOC4A and MTIOC4C</td><td>2</td><td>GTIOC1A and GTIOC1B</td></tr><tr><td>3</td><td>MTIOC4B and MTIOC4D</td><td>3</td><td>GTIOC2A and GTIOC2B</td></tr></table> • SPOER register setting being made • Detection that the main clock oscillator had stopped oscillating • Detection of the comparator C (CMPC) outputs		MTU Complementary PWM Output Signals		GPTW Complementary PWM Output Signals	1	MTIOC3B and MTIOC3D	1	GTIOC0A and GTIOC0B	2	MTIOC4A and MTIOC4C	2	GTIOC1A and GTIOC1B	3	MTIOC4B and MTIOC4D	3	GTIOC2A and GTIOC2B
	MTU Complementary PWM Output Signals		GPTW Complementary PWM Output Signals														
1	MTIOC3B and MTIOC3D	1	GTIOC0A and GTIOC0B														
2	MTIOC4A and MTIOC4C	2	GTIOC1A and GTIOC1B														
3	MTIOC4B and MTIOC4D	3	GTIOC2A and GTIOC2B														
Function	• Each of the POE0#, POE8#, POE10#, POE11#, and POE12# pins can be set for falling-edge, rising-edge, low-level, or high-level detection. When setting a level detection, a sampling clock can be selected from PCLK/1, PCLK/2, PCLK/4, PCLK/8, PCLK/16, and PCLK/128, while the number of samples can be selected from 4 times, or 8 to 16 times. • The outputs of the target pins can be disabled by detecting edge or level of input to the POE0#, POE8#, POE10#, POE11#, and POE12# pins. • The outputs of the target pins can be disabled when oscillation stop is detected by the oscillation stop detection function of the clock generator. • The MTU complementary PWM outputs can be disabled when output levels of the MTU complementary PWM output signals are compared and simultaneous active-level output continues for one cycle or more. • The GPTW outputs can be disabled when output levels of the GPTW complementary PWM output signals (GPTW0 to GPTW2) are compared and simultaneous active-level output continues for one cycle or more. • The outputs of the target pins can be disabled in response to comparator C (CMPC) output detection. • The outputs of the target pins can be disabled by modifying the settings of the POE registers. • Interrupts can be generated by input-level sampling or output-level comparison results. • Output disabling requests triggered by the POE0#, POE8#, POE10#, POE11#, and POE12# pins and the COMP0 to COMP2 level detection signals can be masked by the MTU (MTU0 to MTU4) and GPTW (GPTW0 to GPTW2) output signals.																

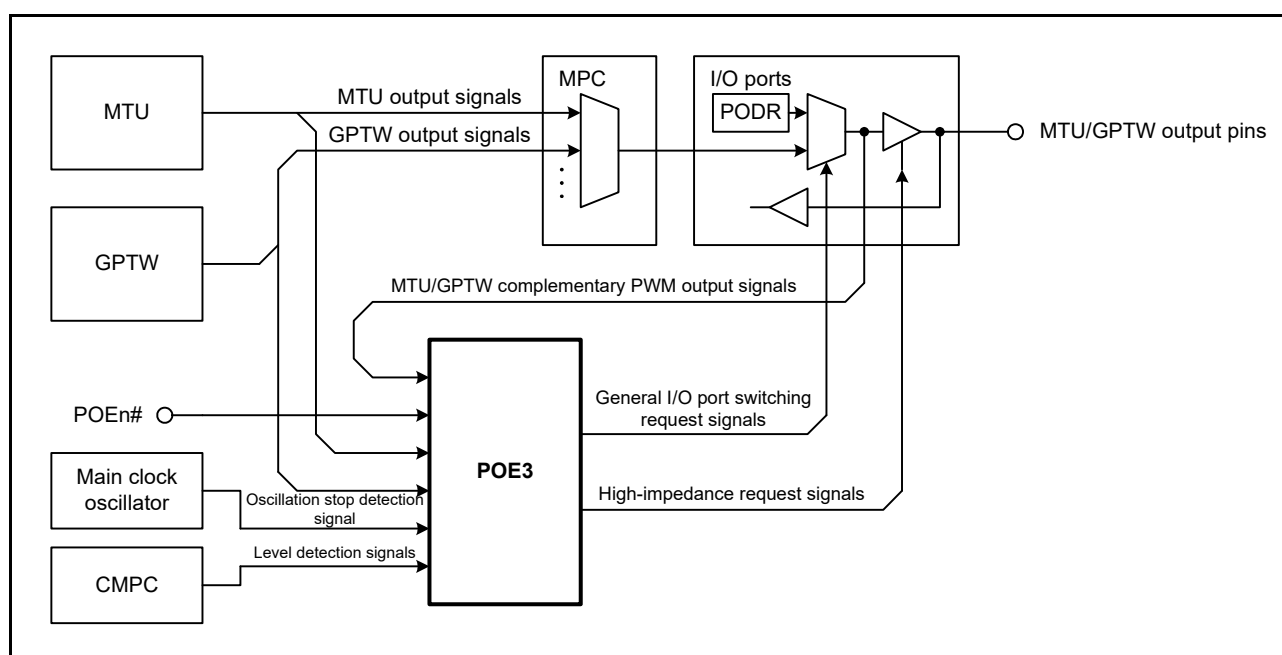


Figure 20.1 POE System Block Diagram

The POE has input-level detector, pin selector, output-level comparator, and a high-impedance request/port switching request/interrupt request generator as shown in Figure 20.2.

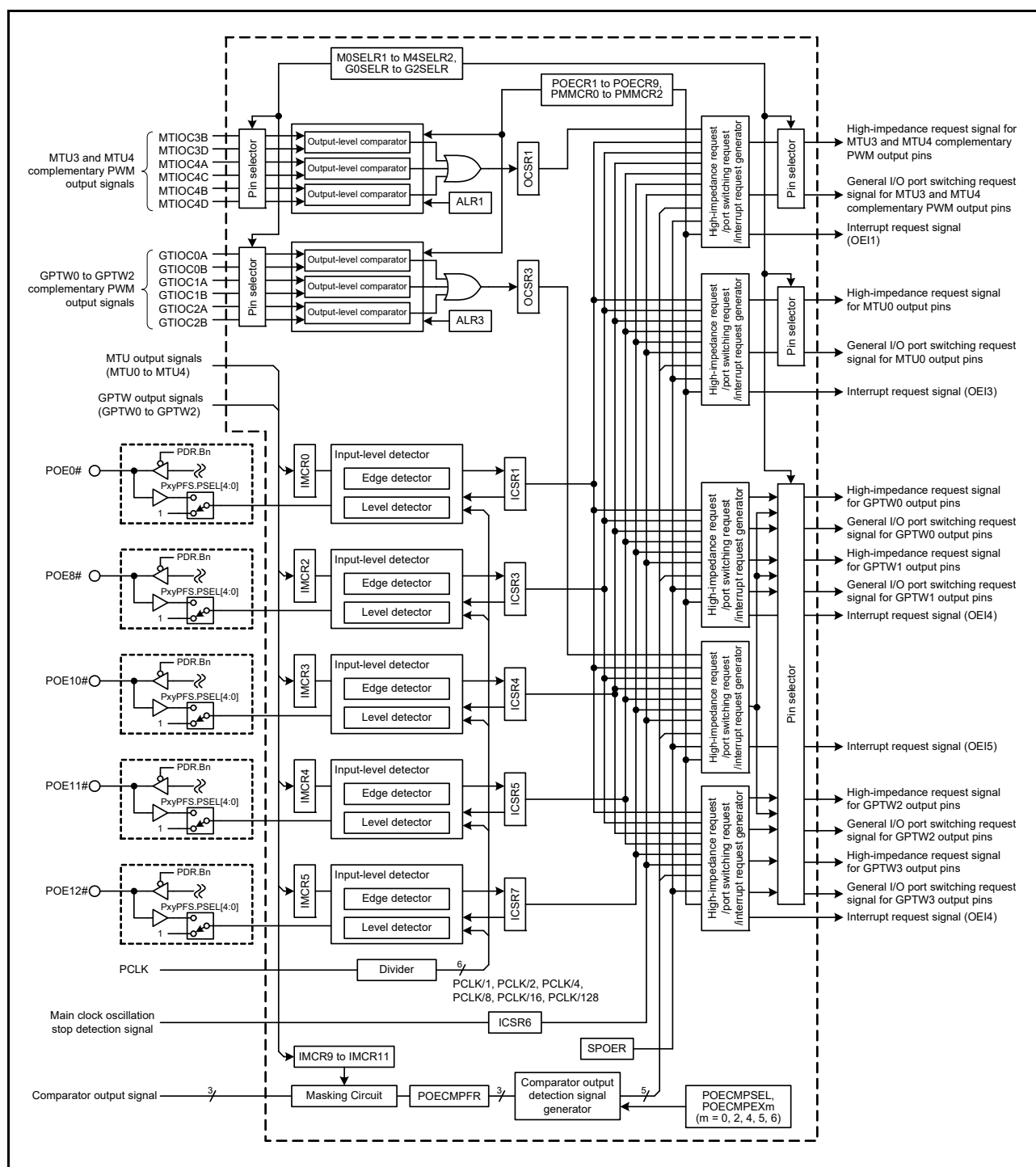


Figure 20.2 POE Block Diagram

Table 20.2 shows I/O pins to be used by the POE.

Table 20.2 POE I/O Pins

Pin Name	I/O	Description
POE0#	Input	Controls all the target pins by register settings. Assigned as a request signal to disable outputs of the MTU complementary PWM output pins (MTU3 and MTU4 pins) after reset.
POE8#	Input	Controls all the target pins by register settings. Assigned as a request signal to disable outputs of the MTU0 pins after reset.
POE10#	Input	Controls all the target pins by register settings. Assigned as a request signal to disable outputs of the GPTW0 and GPTW1 pins after reset.
POE11#	Input	Controls all the target pins by register settings. Assigned as a request signal to disable outputs of the GPTW2 pins after reset.
POE12#	Input	Controls all the target pins by register settings. Assigned as a request signal to disable outputs of the GPTW0 to GPTW2 pins after reset.

Table 20.3 shows output-level comparisons with signal combinations.

Table 20.3 Output Signal Combinations

Signal Combination	I/O	Description
MTIOC3B and MTIOC3D	Output	The MTU complementary PWM output pins (MTU3 and MTU4 pins) are disabled when two signals in a pair simultaneously output the active level (low level when the MTU.TOCR1A.OLSP bit is 0 or high level when the OLSP bit is 1 while the ALR1.OLSEN bit is 0 and the MTU.TOCR1A.TOCS bit is 0, low level when the OLS1P, OLS1N, OLS2P, OLS2N, OLS3P, and OLS3N bits in the MTU.TOCR2A register are 0 or high level when these bits are 1 while the ALR1.OLSEN bit is 0 and the MTU.TOCR1A.TOCS bit is 1, or low level when the OLSG0A, OLSG0B, OLSG1A, OLSG1B, OLSG2A, and OLSG2B bits in the ALR1 register are 0 and high level when these bits are 1 while the ALR1.OLSEN bit is 1) for one or more cycles of the peripheral module clock (PCLK). Pin combinations for output comparison can be selected by registers of POE.
MTIOC4A and MTIOC4C	Output	
MTIOC4B and MTIOC4D	Output	
GTIOC0A and GTIOC0B	Output	The GPTW output pins (GPTW0 to GPTW2 pins) are disabled when two signals in a pair simultaneously output the active level (low level when the OLSG0A, OLSG0B, OLSG1A, OLSG1B, OLSG2A, and OLSG2B bits in the ALR3 register are 0, or high level when these bits are 1) for one or more cycles of the peripheral module clock (PCLK). Pin combinations for output comparison can be selected by registers of POE.
GTIOC1A and GTIOC1B	Output	
GTIOC2A and GTIOC2B	Output	

20.2 Register Descriptions

The POE registers are initialized by a reset.

20.2.1 Input Level Control/Status Register 1 (ICSR1)

Address(es): POE.ICSR1 0009 E400h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
INV	—	—	POE0F	—	—	—	PIE1		POE0M2[3:0]			POE0M[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	POE0M[3:0]	POE0 Mode Select	b3 b0 0 0 0 0: Accepts the request on the falling or rising edge of the POE0# pin input. 0 0 0 1: Samples the input from the POE0# pin with PCLK/8 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 0: Samples the input from the POE0# pin with PCLK/16 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 1: Samples the input from the POE0# pin with PCLK/128 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 0: Samples the input from the POE0# pin with PCLK and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 1: Samples the input from the POE0# pin with PCLK/2 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 1 0: Samples the input from the POE0# pin with PCLK/4 and accepts the request when the low or high level is detected consecutively a specified number of times. Settings other than above are prohibited.	R/W*1
b7 to b4	POE0M2[3:0]	POE0 Sampling Count Select	b7 b4 0 0 0 0: 16 times 0 0 0 1: 4 times 0 0 1 0: 8 times 0 0 1 1: 9 times 0 1 0 0: 10 times 0 1 0 1: 11 times 0 1 1 0: 12 times 0 1 1 1: 13 times 1 0 0 0: 14 times 1 0 0 1: 15 times Settings other than above are prohibited.	R/W*1
b8	PIE1	Port Interrupt Enable 1	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b11 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE0F	POE0 Flag	0: Indicates that an output disabling request has not been input to the POE0# pin. 1: Indicates that an output disabling request has been input to the POE0# pin.	R/(W)*2
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	INV	POE0# Pin Input Invert	0: Does not invert the input signal from the POE0# pin (falling edge or low-level detection) 1: Inverts the input signal from the POE0# pin (rising edge or high-level detection)	R/W

Note 1. Can be modified only once after a reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

The ICSR1 register selects the POE0# pin input modes, controls the enable/disable of interrupts, and indicates status.

POE0M[3:0] Bits (POE0 Mode Select)

These bits select detection mode for the POE0# pin input. Either of edge detection or level detection can be selected. The INV bit specifies which edge to use or which level to use.

When level detection is selected, a sampling interval can be specified.

POE0M2[3:0] Bits (POE0 Sampling Count Select)

These bits specify sampling count when level detection is selected.

When edge detection is selected, setting value for these bits is ignored.

PIE1 Bit (Port Interrupt Enable 1)

This bit enables or disables interrupt requests when the POE0F flag is set to 1.

POE0F Flag (POE0 Flag)

This flag indicates that an output disabling request has been input to the POE0# pin.

[Setting condition]

- When the input set by the POE0M[3:0] and POE0M2[3:0] bits occurs at the POE0# pin

[Clearing condition]

- By writing 0 to the POE0F flag after reading POE0F = 1
When low-level detection is set by the POE0M[3:0] bits, the high level needs to be input to the POE0# pin to write 0 to this flag. Similarly, when high-level detection is set, the low level needs to be input to the POE0# pin to write 0 to this flag. For details, refer to section 20.3.10, Recover from Output Disabled State.

INV Bit (POE0# Pin Input Invert)

This bit logically inverts the input signal from the POE0# pin.

When this bit is 0, falling-edge detection or low-level detection can be selected. When it is 1, rising-edge detection or high-level detection can be selected.

20.2.2 Input Level Control/Status Register 3 (ICSR3)

Address(es): POE.ICSR3 0009 E408h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
INV	—	—	POE8F	—	—	POE8E	PIE3	POE8M2[3:0]				POE8M[3:0]			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	POE8M[3:0]	POE8 Mode Select	b3 b0 0 0 0 0: Accepts the request on the falling or rising edge of the POE8# pin input. 0 0 0 1: Samples the input from the POE8# pin with PCLK/8 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 0: Samples the input from the POE8# pin with PCLK/16 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 1: Samples the input from the POE8# pin with PCLK/128 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 0: Samples the input from the POE8# pin with PCLK and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 1: Samples the input from the POE8# pin with PCLK/2 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 1 0: Samples the input from the POE8# pin with PCLK/4 and accepts the request when the low or high level is detected consecutively a specified number of times. Settings other than above are prohibited.	R/W*1
b7 to b4	POE8M2[3:0]	POE8 Sampling Count Select	b7 b4 0 0 0 0: 16 times 0 0 0 1: 4 times 0 0 1 0: 8 times 0 0 1 1: 9 times 0 1 0 0: 10 times 0 1 0 1: 11 times 0 1 1 0: 12 times 0 1 1 1: 13 times 1 0 0 0: 14 times 1 0 0 1: 15 times Settings other than above are prohibited.	R/W*1
b8	PIE3	Port Interrupt Enable 3	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	POE8E	POE8 Output Disable	0: Does not disable the output by POE8# signal. 1: Disable the output by POE8# signal.	R/W*1
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE8F	POE8 Flag	0: Indicates that an output disabling request has not been input to the POE8# pin. 1: Indicates that an output disabling request has been input to the POE8# pin.	R/(W)*2
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	INV	POE8# Pin Input Invert	0: Does not invert the input signal from the POE8# pin (falling edge or low-level detection) 1: Inverts the input signal from the POE8# pin (rising edge or high-level detection)	R/W

Note 1. Can be modified only once after a reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

The ICSR3 register selects the POE8# pin input mode, controls the enable/disable of interrupts, and indicates status.

POE8M[3:0] Bits (POE8 Mode Select)

These bits select detection mode for the POE8# pin input. Either of edge detection or level detection can be selected. The INV bit specifies which edge to use or which level to use.

When level detection is selected, a sampling interval can be specified.

POE8M2[3:0] Bits (POE8 Sampling Count Select)

These bits specify sampling count when level detection is selected.

When edge detection is selected, setting value for these bits is ignored.

PIE3 Bit (Port Interrupt Enable 3)

This bit enables or disables interrupt requests when the POE8F flag is set to 1.

POE8E Bit (POE8 Output Disable)

This bit specifies whether to disable the output of the corresponding pin when the POE8F flag is set to 1.

POE8F Flag (POE8 Flag)

This flag indicates that an output disabling request has been input to the POE8# pin.

[Setting condition]

- When the input set by the POE8M[3:0] and POE8M2[3:0] bits occurs at the POE8# pin

[Clearing condition]

- By writing 0 to the POE8F flag after reading POE8F = 1
When low-level detection is set by the POE8M[3:0] bits, the high level needs to be input to the POE8# pin to write 0 to this flag. Similarly, when high-level detection is set, the low level needs to be input to the POE8# pin to write 0 to this flag. For details, refer to section 20.3.10, Recover from Output Disabled State.

INV Bit (POE8# Pin Input Invert)

This bit logically inverts the input signal from the POE8# pin.

When this bit is 0, falling-edge detection or low-level detection can be selected. When it is 1, rising-edge detection or high-level detection can be selected.

20.2.3 Input Level Control/Status Register 4 (ICSR4)

Address(es): POE.ICSR4 0009 E416h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
INV	—	—	POE10F	—	—	POE10E	PIE4	POE10M2[3:0]				POE10M[3:0]			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	POE10M[3:0]	POE10 Mode Select	b3 b0 0 0 0 0: Accepts the request on the falling or rising edge of the POE10# pin input. 0 0 0 1: Samples the input from the POE10# pin with PCLK/8 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 0: Samples the input from the POE10# pin with PCLK/16 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 1: Samples the input from the POE10# pin with PCLK/128 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 0: Samples the input from the POE10# pin with PCLK and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 1: Samples the input from the POE10# pin with PCLK/2 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 1 0: Samples the input from the POE10# pin with PCLK/4 and accepts the request when the low or high level is detected consecutively a specified number of times. Settings other than above are prohibited.	R/W*1
b7 to b4	POE10M2[3:0]	POE10 Sampling Count Select	b7 b4 0 0 0 0: 16 times 0 0 0 1: 4 times 0 0 1 0: 8 times 0 0 1 1: 9 times 0 1 0 0: 10 times 0 1 0 1: 11 times 0 1 1 0: 12 times 0 1 1 1: 13 times 1 0 0 0: 14 times 1 0 0 1: 15 times Settings other than above are prohibited.	R/W*1
b8	PIE4	Port Interrupt Enable 4	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	POE10E	POE10 Output Disable	0: Does not disable the output by POE10# signal. 1: Disable the output by POE10# signal.	R/W*1
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE10F	POE10 Flag	0: Indicates that an output disabling request has not been input to the POE10# pin. 1: Indicates that an output disabling request has been input to the POE10# pin.	R/(W)*2
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	INV	POE10# Pin Input Invert	0: Does not invert the input signal from the POE10# pin (falling edge or low-level detection) 1: Inverts the input signal from the POE10# pin (rising edge or high-level detection)	R/W

Note 1. Can be modified only once after a reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

The ICSR4 register selects the POE10# pin input mode, controls the enable/disable of interrupts, and indicates status.

POE10M[3:0] Bits (POE10 Mode Select)

These bits select detection mode for the POE10# pin input. Either of edge detection or level detection can be selected.

The INV bit specifies which edge to use or which level to use.

When level detection is selected, a sampling interval can be specified.

POE10M2[3:0] Bits (POE10 Sampling Count Select)

These bits specify sampling count when level detection is selected.

When edge detection is selected, setting value for these bits is ignored.

PIE4 Bit (Port Interrupt Enable 4)

This bit enables or disables interrupt requests when the POE10F flag is set to 1.

POE10E Bit (POE10 Output Disable)

This bit specifies whether to disable the output of the corresponding pin when the POE10F flag is set to 1.

POE10F Flag (POE10 Flag)

This flag indicates that an output disabling request has been input to the POE10# pin.

[Setting condition]

- When the input set by the POE10M[3:0] and POE10M2[3:0] bits occurs at the POE10# pin

[Clearing condition]

- By writing 0 to the POE10F flag after reading POE10F = 1
When low-level detection is set by the POE10M[3:0] bits, the high level needs to be input to the POE10# pin to write 0 to this flag. Similarly, when high-level detection is set, the low level needs to be input to the POE10# pin to write 0 to this flag. For details, refer to section 20.3.10, Recover from Output Disabled State.

INV Bit (POE10# Pin Input Invert)

This bit logically inverts the input signal from the POE10# pin.

When this bit is 0, falling-edge detection or low-level detection can be selected. When it is 1, rising-edge detection or high-level detection can be selected.

20.2.4 Input Level Control/Status Register 5 (ICSR5)

Address(es): POE.ICSR5 0009 E418h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
INV	—	—	POE11F	—	—	POE11E	PIE5	POE11M2[3:0]				POE11M[3:0]			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	POE11M[3:0]	POE11 Mode Select	b3 b0 0 0 0 0: Accepts the request on the falling or rising edge of the POE11# pin input. 0 0 0 1: Samples the input from the POE11# pin with PCLK/8 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 0: Samples the input from the POE11# pin with PCLK/16 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 1: Samples the input from the POE11# pin with PCLK/128 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 0: Samples the input from the POE11# pin with PCLK and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 1: Samples the input from the POE11# pin with PCLK/2 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 1 0: Samples the input from the POE11# pin with PCLK/4 and accepts the request when the low or high level is detected consecutively a specified number of times. Settings other than above are prohibited.	R/W*1
b7 to b4	POE11M2[3:0]	POE11 Sampling Count Select	b7 b4 0 0 0 0: 16 times 0 0 0 1: 4 times 0 0 1 0: 8 times 0 0 1 1: 9 times 0 1 0 0: 10 times 0 1 0 1: 11 times 0 1 1 0: 12 times 0 1 1 1: 13 times 1 0 0 0: 14 times 1 0 0 1: 15 times Settings other than above are prohibited.	R/W*1
b8	PIE5	Port Interrupt Enable 5	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	POE11E	POE11 Output Disable	0: Does not disable the output by POE11# signal. 1: Disable the output by POE11# signal.	R/W*1
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE11F	POE11 Flag	0: Indicates that an output disabling request has not been input to the POE11# pin. 1: Indicates that an output disabling request has been input to the POE11# pin.	R/(W)*2
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	INV	POE11# Pin Input Invert	0: Does not invert the input signal from the POE11# pin (falling edge or low-level detection) 1: Inverts the input signal from the POE11# pin (rising edge or high-level detection)	R/W

Note 1. Can be modified only once after a reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

The ICSR5 register selects the POE11# pin input mode, controls the enable/disable of interrupts, and indicates status.

POE11M[3:0] Bits (POE11 Mode Select)

These bits select detection mode for the POE11# pin input. Either of edge detection or level detection can be selected.

The INV bit specifies which edge to use or which level to use.

When level detection is selected, a sampling interval can be specified.

POE11M2[3:0] Bits (POE11 Sampling Count Select)

These bits specify sampling count when level detection is selected.

When edge detection is selected, setting value for these bits is ignored.

PIE5 Bit (Port Interrupt Enable 5)

This bit enables or disables interrupt requests when the POE11F flag is set to 1.

POE11E Bit (POE11 Output Disable)

This bit specifies whether to disable the output of the corresponding pin when the POE11F flag is set to 1.

POE11F Flag (POE11 Flag)

This flag indicates that an output disabling request has been input to the POE11# pin.

[Setting condition]

- When the input set by the POE11M[3:0] and POE11M2[3:0] bits occurs at the POE11# pin

[Clearing condition]

- By writing 0 to the POE11F flag after reading POE11F = 1
When low-level detection is set by the POE11M[3:0] bits, the high level needs to be input to the POE11# pin to write 0 to this flag. Similarly, when high-level detection is set, the low level needs to be input to the POE11# pin to write 0 to this flag. For details, refer to section 20.3.10, Recover from Output Disabled State.

INV Bit (POE11# Pin Input Invert)

This bit logically inverts the input signal from the POE11# pin.

When this bit is 0, falling-edge detection or low-level detection can be selected. When it is 1, rising-edge detection or high-level detection can be selected.

20.2.5 Input Level Control/Status Register 6 (ICSR6)

Address(es): POE.ICSR6 0009 E41Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	OSTST F	—	—	OSTST E	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b8 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b9	OSTSTE	Oscillation Stop Output Disable	0: Does not disable the output when the oscillation stop is detected. 1: Disable the output when the oscillation stop is detected.	R/W*1
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	OSTSTF	Oscillation Stop Detection Flag	0: Indicates that an output disabling request by oscillation stop has not been generated. 1: Indicates that an output disabling request by oscillation stop has been generated.	R/W*2
b15 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

The ICSR6 register controls the oscillation stop output disabling and indicates status.

OSTSTE Bit (Oscillation Stop Output Disable)

This bit specifies whether to disable the output of the target pin when oscillation stop is detected.

OSTSTF Flag (Oscillation Stop Detection Flag)

This flag indicates that an output disabling request by the oscillation stop has been generated.

When the main clock oscillation stops, this flag is set to 1. To clear this flag, wait for at least 10 cycles of PCLK after this flag becomes 1 and write 0 to this flag while the OSTDSR.OSTDF flag is 0. Writing 0 to this flag while the OSTDSR.OSTDF flag is 1 cannot clear this flag. After clearing this flag, confirm that the flag has actually been modified to 0.

[Setting condition]

- When oscillation stop is detected

[Clearing condition]

- By writing 0 to the OSTSTF flag after reading OSTSTF = 1

20.2.6 Input Level Control/Status Register 7 (ICSR7)

Address(es): POE.ICSR7 0009 E420h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
INV	—	—	POE12F	—	—	POE12E	PIE7	POE12M2[3:0]				POE12M[3:0]			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	POE12M[3:0]	POE12 Mode Select	b3 b0 0 0 0 0: Accepts the request on the falling or rising edge of the POE12# pin input. 0 0 0 1: Samples the input from the POE12# pin with PCLK/8 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 0: Samples the input from the POE12# pin with PCLK/16 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 0 1 1: Samples the input from the POE12# pin with PCLK/128 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 0: Samples the input from the POE12# pin with PCLK and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 0 1: Samples the input from the POE12# pin with PCLK/2 and accepts the request when the low or high level is detected consecutively a specified number of times. 0 1 1 0: Samples the input from the POE12# pin with PCLK/4 and accepts the request when the low or high level is detected consecutively a specified number of times. Settings other than above are prohibited.	R/W*1
b7 to b4	POE12M2[3:0]	POE12 Sampling Count Select	b7 b4 0 0 0 0: 16 times 0 0 0 1: 4 times 0 0 1 0: 8 times 0 0 1 1: 9 times 0 1 0 0: 10 times 0 1 0 1: 11 times 0 1 1 0: 12 times 0 1 1 1: 13 times 1 0 0 0: 14 times 1 0 0 1: 15 times Settings other than above are prohibited.	R/W*1
b8	PIE7	Port Interrupt Enable 7	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	POE12E	POE12 Output Disable	0: Does not disable the output by POE12# signal. 1: Disable the output by POE12# signal.	R/W*1
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	POE12F	POE12 Flag	0: Indicates that an output disabling request has not been input to the POE12# pin. 1: Indicates that an output disabling request has been input to the POE12# pin.	R/(W)*2
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	INV	POE12# Pin Input Invert	0: Does not invert the input signal from the POE12# pin (falling edge or low-level detection) 1: Inverts the input signal from the POE12# pin (rising edge or high-level detection)	R/W

Note 1. Can be modified only once after a reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

The ICSR7 register selects the POE12# pin input mode, controls the enable/disable of interrupts, and indicates status.

POE12M[3:0] Bits (POE12 Mode Select)

These bits select detection mode for the POE12# pin input. Either of edge detection or level detection can be selected.

The INV bit specifies which edge to use or which level to use.

When level detection is selected, a sampling interval can be specified.

POE12M2[3:0] Bits (POE12 Sampling Count Select)

These bits specify sampling count when level detection is selected.

When edge detection is selected, setting value for these bits is ignored.

PIE7 Bit (Port Interrupt Enable 7)

This bit enables or disables interrupt requests when the POE12F flag is set to 1.

POE12E Bit (POE12 Output Disable)

This bit specifies whether to disable the output of the corresponding pin when the POE12F flag is set to 1.

POE12F Flag (POE12 Flag)

This flag indicates that an output disabling request has been input to the POE12# pin.

[Setting condition]

- When the input set by the POE12M[3:0] and POE12M2[3:0] bits occurs at the POE12# pin

[Clearing condition]

- By writing 0 to the POE12F flag after reading POE12F = 1
When low-level detection is set by the POE12M[3:0] bits, the high level needs to be input to the POE12# pin to write 0 to this flag. Similarly, when high-level detection is set, the low level needs to be input to the POE12# pin to write 0 to this flag. For details, refer to section 20.3.10, Recover from Output Disabled State.

INV Bit (POE12# Pin Input Invert)

This bit logically inverts the input signal from the POE12# pin.

When this bit is 0, falling-edge detection or low-level detection can be selected. When it is 1, rising-edge detection or high-level detection can be selected.

20.2.7 Output Level Control/Status Register 1 (OCSR1)

Address(es): POE.OCSR1 0009 E402h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OSF1	—	—	—	—	—	OCE1	OIE1	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	OIE1	Simultaneous Conduction Interrupt Enable 1	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	OCE1	Simultaneous Conduction Output Disable 1	0: Does not disable the outputs when they simultaneously go to an active level. 1: Disable the outputs when they simultaneously go to an active level.	R/W ^{*1}
b14 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	OSF1	Simultaneous Conduction Flag 1	0: Indicates that outputs have not simultaneously become an active level. 1: Indicates that outputs have simultaneously become an active level.	R/(W) ^{*2}

Note 1. Can be modified only once after a reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

The OCSR1 register enables/disables comparison of output levels, enables/disables a simultaneous conduction interrupt, and indicates the status for the MTU3 and MTU4 pins.

OIE1 Bit (Simultaneous Conduction Interrupt Enable 1)

This bit enables or disables interrupt requests when the OSF1 flag is set to 1.

OCE1 Bit (Simultaneous Conduction Output Disable 1)

This bit specifies whether to disable the output of the corresponding pin when the OSF1 flag is set to 1.

OSF1 Flag (Simultaneous Conduction Flag 1)

This flag indicates that at least one of the three pairs of two-phase output signals for MTU complementary PWM output (MTU3 and MTU4) has simultaneously become at the active level. If the output disabling control for the corresponding pins is not enabled, this flag does not become 1.

For setting the active level, refer to section 20.2.9, Active Level Setting Register 1 (ALR1).

[Setting condition]

- When the MTIOC3B and MTIOC3D signals simultaneously go to the active level^{*1} for at least one cycle of PCLK while the value of the POE2R2.MTU3BDZE bit, or either or both of the PMMCR1.MTU3BME and PMMCR1.MTU3DME bits, is 1.
- When the MTIOC4A and MTIOC4C signals simultaneously go to the active level^{*1} for at least one cycle of PCLK while the value of the POE2R2.MTU4ACZE bit, or either or both of the PMMCR1.MTU4AME and PMMCR1.MTU4CME bits, is 1.
- When the MTIOC4B and MTIOC4D signals simultaneously go to the active level^{*1} for at least one cycle of PCLK while the value of the POE2R2.MTU4BDZE bit, or either or both of the PMMCR1.MTU4BME and PMMCR1.MTU4DME bits, is 1.

Note 1. The setting condition is judged only by the level of the signal regardless the setting of the MPC.PxyPFS register.

[Clearing condition]

- By writing 0 to the OSF1 flag after reading OSF1 = 1

To write 0 to this flag, the MTU complementary PWM output signals should be set to the inactive level. For details, refer to section 20.3.10, Recover from Output Disabled State.

20.2.8 Output Level Control/Status Register 3 (OCSR3)

Address(es): POE.OCSR3 0009 E42Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OSF3	—	—	—	—	—	OCE3	OIE3	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	OIE3	Simultaneous Conduction Interrupt Enable 3	0: Interrupt requests disabled 1: Interrupt requests enabled	R/W
b9	OCE3	Simultaneous Conduction Output Disable 3	0: Does not disable the outputs when they simultaneously go to an active level. 1: Disable the outputs when they simultaneously go to an active level.	R/W*1
b14 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	OSF3	Simultaneous Conduction Flag 3	0: Indicates that outputs have not simultaneously become an active level. 1: Indicates that outputs have simultaneously become an active level.	R/(W)*2

Note 1. Can be modified only once after a reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

The OCSR3 register enables/disables comparison of output levels, enables/disables a simultaneous conduction interrupt, and indicates the status for the GPTW0 to GPTW2 pins.

OIE3 Bit (Simultaneous Conduction Interrupt Enable 3)

This bit enables or disables interrupt requests when the OSF3 flag is set to 1.

OCE3 Bit (Simultaneous Conduction Output Disable 3)

This bit specifies whether to disable the output of the corresponding pin when the OSF3 flag is set to 1.

OSF3 Flag (Simultaneous Conduction Flag 3)

This flag indicates that at least one of the three pairs of two-phase output signals for GPTW complementary PWM output (GPTW0 to GPTW2) has simultaneously become at the active level. If the output disabling control for the corresponding pins is not enabled, this flag does not become 1.

For setting the active level, refer to section 20.2.10, Active Level Setting Register 3 (ALR3).

[Setting condition]

- When the GTIOC0A and GTIOC0B signals simultaneously go to the active level*1 for at least one cycle of PCLK while the value of the POE3R3.GPT0ABZE bit, or either or both of the PMMCR2.GPT0AME and PMMCR2.GPT0BME bits, is 1.
- When the GTIOC1A and GTIOC1B signals simultaneously go to the active level*1 for at least one cycle of PCLK

while the value of the POECR3.GPT1ABZE bit, or either or both of the PMMCR2.GPT1AME and PMMCR2.GPT1BME bits, is 1.

- When the GTIOC2A and GTIOC2B signals simultaneously go to the active level*1 for at least one cycle of PCLK while the value of the POECR3.GPT2ABZE bit, or either or both of the PMMCR2.GPT2AME and PMMCR2.GPT2BME bits, is 1.

Note 1. The setting condition is judged only by the level of the signal regardless the setting of the MPC.PxyPFS register.

[Clearing condition]

- By writing 0 to the OSF3 flag after reading OSF3 = 1

To write 0 to this flag, the GPTW complementary PWM output signals should be set to the inactive level. For details, refer to section 20.3.10, Recover from Output Disabled State.

20.2.9 Active Level Setting Register 1 (ALR1)

Address(es): POE.AL1 0009 E41Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OLSEN	—	OLSG2B	OLSG2A	OLSG1B	OLSG1A	OLSG0B	OLSG0A
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OLSG0A	MTIOC3B Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b1	OLSG0B	MTIOC3D Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b2	OLSG1A	MTIOC4A Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b3	OLSG1B	MTIOC4C Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b4	OLSG2A	MTIOC4B Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b5	OLSG2B	MTIOC4D Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	OLSEN	Active Level Setting Enable	0: Disabled 1: Enabled	R/W*1
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The ALR1 register specifies active levels for when detecting a simultaneous conduction for the MTU3 and MTU4 by the OCSR1 register.

OLSG0A Bit (MTIOC3B Signal Active Level Setting)

This bit sets the active level of the MTIOC3B output. Specifically, setting the OLSG0A bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG0B Bit (MTIOC3D Signal Active Level Setting)

This bit sets the active level of the MTIOC3D output. Specifically, setting the OLSG0B bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG1A Bit (MTIOC4A Signal Active Level Setting)

This bit sets the active level of the MTIOC4A output. Specifically, setting the OLSG1A bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG1B Bit (MTIOC4C Signal Active Level Setting)

This bit sets the active level of the MTIOC4C output. Specifically, setting the OLSG1B bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG2A Bit (MTIOC4B Signal Active Level Setting)

This bit sets the active level of the MTIOC4B output. Specifically, setting the OLSG2A bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG2B Bit (MTIOC4D Signal Active Level Setting)

This bit sets the active level of the MTIOC4D output. Specifically, setting the OLSG2B bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSEN Bit (Active Level Setting Enable)

This bit determines enabling/disabling the active level setting by the OLSGnm bit ($n = 0$ to 2 ; $m = A, B$). When the OLSEN bit is 0, the OLSGnm bit setting becomes disabled, and the active level for the MTU3 and MTU4 outputs follows the settings for the MTU.TOCR1A and MTU.TOCR2A registers. When the OLSEN bit is 1, the active level for the MTU3 and MTU4 outputs follows the OLSGnm bit setting.

20.2.10 Active Level Setting Register 3 (ALR3)

Address(es): POE.AL3 0009 E42Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OLSEN	—	OLSG2B	OLSG2A	OLSG1B	OLSG1A	OLSG0B	OLSG0A
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OLSG0A	GTIOC0A Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b1	OLSG0B	GTIOC0B Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b2	OLSG1A	GTIOC1A Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b3	OLSG1B	GTIOC1B Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b4	OLSG2A	GTIOC2A Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b5	OLSG2B	GTIOC2B Signal Active Level Setting	0: Active low 1: Active high	R/W*1
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	OLSEN	Active Level Setting Enable	0: Disabled 1: Enabled	R/W*1
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The ALR3 register specifies active levels for when detecting a simultaneous conduction for the GPTW0 to GPTW2 by the OCSR3 register.

OLSG0A Bit (GTIOC0A Signal Active Level Setting)

This bit sets the active level of the GTIOC0A output. Specifically, setting the OLSG0A bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG0B Bit (GTIOC0B Signal Active Level Setting)

This bit sets the active level of the GTIOC0B output. Specifically, setting the OLSG0B bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG1A Bit (GTIOC1A Signal Active Level Setting)

This bit sets the active level of the GTIOC1A output. Specifically, setting the OLSG1A bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG1B Bit (GTIOC1B Signal Active Level Setting)

This bit sets the active level of the GTIOC1B output. Specifically, setting the OLSG1B bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG2A Bit (GTIOC2A Signal Active Level Setting)

This bit sets the active level of the GTIOC2A output. Specifically, setting the OLSG2A bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSG2B Bit (GTIOC2B Signal Active Level Setting)

This bit sets the active level of the GTIOC2B output. Specifically, setting the OLSG2B bit to 0 sets the low level and to 1 sets the high level as the active level for detection of simultaneous conduction.

OLSEN Bit (Active Level Setting Enable)

This bit determines enabling/disabling the active level setting by the OLSGnm bit (n = 0 to 2; m = A, B). When the OLSEN bit is 0, the OLSGnm bit setting becomes disabled.

The active level for the GPTW output can be specified only when the OLSEN bit is 1. When detecting a simultaneous conduction for the GPTW, set the OLSEN bit to 1, and specify the active level for the GPTW output by the OLSGnm bit.

20.2.11 Software Port Output Enable Register (SPOER)

Address(es): POE.SPOER 0009 E42Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	GPT02 HIZ	—	—	—	GPT23 HIZ	GPT01 HIZ	MTUC H0HIZ	—	MTUC H34HIZ
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MTUCH34HIZ	MTU3 and MTU4 Pin Output Disable	0: Does not disable the outputs 1: Disable the outputs.	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	MTUCH0HIZ	MTU0 Pin Output Disable	0: Does not disable the outputs 1: Disable the outputs.	R/W
b3	GPT01HIZ	GPTW0 and GPTW1 Pin Output Disable	0: Does not disable the outputs 1: Disable the outputs.	R/W
b4	GPT23HIZ	GPTW2 Pin Output Disable	0: Does not disable the outputs 1: Disable the outputs.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	GPT02HIZ	GPTW0 to GPTW2 Pin Output Disable	0: Does not disable the outputs 1: Disable the outputs.	R/W
b15 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The SPOER register is used to disable the outputs of the corresponding pins.

MTUCH34HIZ Bit (MTU3 and MTU4 Pin Output Disable)

This bit specifies whether to disable the outputs of the MTU complementary PWM output pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D). To set this bit to 0, confirm that the bit is 1 and then write 0 to the bit.

MTUCH0HIZ Bit (MTU0 Pin Output Disable)

This bit specifies whether to disable the outputs of the MTU0 pins. To set this bit to 0, confirm that the bit is 1 and then write 0 to the bit.

GPT01HIZ Bit (GPTW0 and GPTW1 Pin Output Disable)

This bit specifies whether to disable the outputs of the GPTW0 and GPTW1 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B). To set this bit to 0, confirm that the bit is 1 and then write 0 to the bit.

GPT23HIZ Bit (GPTW2 Pin Output Disable)

This bit specifies whether to disable the outputs of the GPTW2 pins (GTIOC2A, GTIOC2B). To set this bit to 0, confirm that the bit is 1 and then write 0 to the bit.

GPT02HIZ Bit (GPTW0 to GPTW2 Pin Output Disable)

This bit specifies whether to disable the outputs of the GPTW0 to GPTW2 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B, GTIOC2A, GTIOC2B). To set this bit to 0, confirm that the bit is 1 and then write 0 to the bit.

20.2.12 Port Output Enable Control Register 1 (POECR1)

Address(es): POE.POECR1 0009 E40Bh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	MTU0DZE	MTU0CZE	MTU0BZE	MTU0AZE
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	MTU0AZE	MTIOC0A Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b1	MTU0BZE	MTIOC0B Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b2	MTU0CZE	MTIOC0C Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b3	MTU0DZE	MTIOC0D Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECR1 register controls high-impedance state of the MTU0 pins.

MTU0AZE Bit (MTIOC0A Pin High-Impedance Enable)

This bit specifies whether to switch the MTIOC0A output to the high-impedance state when any of the SPOER.MTUCH0HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag (n = 1, 3 to 5, 7; m = 0, 8, 10, 11, 12) additionally selected in the POECR5 register, or the POECMPFR.CjFLAG flag (j = 0 to 2), is set to 1.

Set the PMMCR0.MTU0AME bit to 0 when setting this bit to 1.

MTU0BZE Bit (MTIOC0B Pin High-Impedance Enable)

This bit specifies whether to switch the MTIOC0B output to the high-impedance state when any of the SPOER.MTUCH0HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR5 register, or the POECMPFR.CjFLAG flag, is set to 1.

Set the PMMCR0.MTU0BME bit to 0 when setting this bit to 1.

MTU0CZE Bit (MTIOC0C Pin High-Impedance Enable)

This bit specifies whether to switch the MTIOC0C output to the high-impedance state when any of the SPOER.MTUCH0HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR5 register, or the POECMPFR.CjFLAG flag, is set to 1.

Set the PMMCR0.MTU0CME bit to 0 when setting this bit to 1.

MTU0DZE Bit (MTIOC0D Pin High-Impedance Enable)

This bit specifies whether to switch the MTIOC0D output to the high-impedance state when any of the SPOER.MTUCH0HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR5 register, or the POECMPFR.CjFLAG flag, is set to 1.

Set the PMMCR0.MTU0DME bit to 0 when setting this bit to 1.

20.2.13 Port Output Enable Control Register 2 (POECR2)

Address(es): POE.POECR2 0009 E40Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MTU3B DZE	MTU4A CZE	MTU4B DZE	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	MTU4BDZE	MTIOC4B/MTIOC4D Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b9	MTU4ACZE	MTIOC4A/MTIOC4C Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b10	MTU3BDZE	MTIOC3B/MTIOC3D Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECR2 register controls high-impedance state of the MTU complementary PWM output pins (MTU3 and MTU4 pins).

MTU4BDZE Bit (MTIOC4B/MTIOC4D Pin High-Impedance Enable)

This bit specifies whether to switch the MTIOC4B output and MTIOC4D output to the high-impedance state when at least one of the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, or the POECMPFR.CjFLAG flag is set to 1.

Set the PMMCR1.MTU4BME and PMMCR1.MTU4DME bits to 0 when setting this bit to 1.

MTU4ACZE Bit (MTIOC4A/MTIOC4C Pin High-Impedance Enable)

This bit specifies whether to switch the MTIOC4A output and MTIOC4C output to the high-impedance state when at least one of the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, or the POECMPFR.CjFLAG flag is set to 1.

Set the PMMCR1.MTU4AME and PMMCR1.MTU4CME bits to 0 when setting this bit to 1.

MTU3BDZE Bit (MTIOC3B/MTIOC3D Pin High-Impedance Enable)

This bit specifies whether to switch the MTIOC3B output and MTIOC3D output to the high-impedance state when at least one of the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, or the POECMPFR.CjFLAG flag is set to 1.

Set the PMMCR1.MTU3BME and PMMCR1.MTU3DME bits to 0 when setting this bit to 1.

20.2.14 Port Output Enable Control Register 3 (POECR3)

Address(es): POE.POECR3 0009 E40Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	GPT2A BZE	GPT1A BZE	GPT0A BZE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	GPT0ABZE	GTIOC0A/GTIOC0B Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b1	GPT1ABZE	GTIOC1A/GTIOC1B Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b2	GPT2ABZE	GTIOC2A/GTIOC2B Pin High-Impedance Enable	0: Does not switch the pin to high-impedance state. 1: Switch the pin to high-impedance state.	R/W*1
b15 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECR3 register controls high-impedance state of the GPTW pins (GPTW0 to GPTW2).

GPT0ABZE Bit (GTIOC0A/GTIOC0B Pin High-Impedance Enable)

This bit determines whether to switch the GTIOC0A and GTIOC0B outputs to the high-impedance state when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit is 1), the SPOER.GPT01HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag (n = 1, 3 to 5, 7; m = 0, 8, 10, 11, 12) additionally selected in the POECR6 register, POECMPFR.CjFLAG flag (j = 0 to 2), the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the POECMPFR.CjFLAG flag.

Set the PMMCR2.GPT0AME and PMMCR2.GPT0BME bits to 0 when setting this bit to 1.

GPT1ABZE Bit (GTIOC1A/GTIOC1B Pin High-Impedance Enable)

This bit determines whether to switch the GTIOC1A and GTIOC1B outputs to the high-impedance state when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit is 1), the SPOER.GPT01HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag additionally selected in the POECR6 register, POECMPFR.CjFLAG flag, the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the POECMPFR.CjFLAG flag.

Set the PMMCR2.GPT1AME and PMMCR2.GPT1BME bits to 0 when setting this bit to 1.

GPT2ABZE Bit (GTIOC2A/GTIOC2B Pin High-Impedance Enable)

This bit determines whether to switch the GTIOC2A and GTIOC2B outputs to the high-impedance state when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit is 1), the SPOER.GPT23HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag additionally selected in the POECR6B register, POECMPFR.CjFLAG flag, the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the POECMPFR.CjFLAG flag.

Set the PMMCR2.GPT2AME and PMMCR2.GPT2BME bits to 0 when setting this bit to 1.

20.2.15 Port Output Enable Control Register 4 (POECR4)

Address(es): POE.POECR4 0009 E410h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	IC6ADDM T34ZE	IC5ADDM T34ZE	IC4ADDM T34ZE	IC3ADDM T34ZE	—	IC1ADDM T34ZE	CMADDM T34ZE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMADDMT34ZE	MTU3 and MTU4 Output Disabling Condition CFLAG Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b1	IC1ADDMT34ZE	MTU3 and MTU4 Output Disabling Condition POE0F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	IC3ADDMT34ZE	MTU3 and MTU4 Output Disabling Condition POE8F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b4	IC4ADDMT34ZE	MTU3 and MTU4 Output Disabling Condition POE10F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b5	IC5ADDMT34ZE	MTU3 and MTU4 Output Disabling Condition POE11F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b6	IC6ADDMT34ZE	MTU3 and MTU4 Output Disabling Condition POE12F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b15 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECR4 register is used to set the control conditions to disable the output of the MTU complementary PWM output pins (MTU3 and MTU4 pins).

CMADDMT34ZE Bit (MTU3 and MTU4 Output Disabling Condition CFLAG Add)

Adds the POECMPFR.CjFLAG flag (j = 0 to 2) to the output disabling control conditions for the MTU3 and MTU4 pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D).

However, when the pins are disabled by the flag, an OEIk interrupt (k = 1, 3 to 5) will not be generated.

IC1ADDMT34ZE Bit (MTU3 and MTU4 Output Disabling Condition POE0F Add)

Adds the ICSR1.POE0F flag to the output disabling control conditions for the MTU3 and MTU4 pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D).

IC3ADDMT34ZE Bit (MTU3 and MTU4 Output Disabling Condition POE8F Add)

Adds the ICSR3.POE8F flag to the output disabling control conditions for the MTU3 and MTU4 pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D).

IC4ADDMT34ZE Bit (MTU3 and MTU4 Output Disabling Condition POE10F Add)

Adds the ICSR4.POE10F flag to the output disabling control conditions for the MTU3 and MTU4 pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D).

IC5ADDMT34ZE Bit (MTU3 and MTU4 Output Disabling Condition POE11F Add)

Adds the ICSR5.POE11F flag to the output disabling control conditions for the MTU3 and MTU4 pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D).

IC6ADDMT34ZE Bit (MTU3 and MTU4 Output Disabling Condition POE12F Add)

Adds the ICSR7.POE12F flag to the output disabling control conditions for the MTU3 and MTU4 pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D).

20.2.16 Port Output Enable Control Register 5 (POECR5)

Address(es): POE.POECR5 0009 E412h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	IC6ADD MT0ZE	IC5ADD MT0ZE	IC4ADD MT0ZE	IC3ADD MT0ZE	—	IC1ADD MT0ZE	CMADD MT0ZE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMADDMT0ZE	MTU0 Output Disabling Condition CFLAG Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b1	IC1ADDMT0ZE	MTU0 Output Disabling Condition POE0F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	IC3ADDMT0ZE	MTU0 Output Disabling Condition POE8F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b4	IC4ADDMT0ZE	MTU0 Output Disabling Condition POE10F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b5	IC5ADDMT0ZE	MTU0 Output Disabling Condition POE11F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b6	IC6ADDMT0ZE	MTU0 Output Disabling Condition POE12F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b15 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECR5 register is used to set the control conditions to disable the output of the MTU0 pins.

CMADDMT0ZE Bit (MTU0 Output Disabling Condition CFLAG Add)

Adds the POECMPFR.CjFLAG flag (j = 0 to 2) to the output disabling control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

However, when the pins are disabled by the flag, an OEIk interrupt (k = 1, 3 to 5) will not be generated.

IC1ADDMT0ZE Bit (MTU0 Output Disabling Condition POE0F Add)

Adds the ICSR1.POE0F flag to the output disabling control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC3ADDMT0ZE Bit (MTU0 Output Disabling Condition POE8F Add)

Adds the ICSR3.POE8F flag to the output disabling control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC4ADDMT0ZE Bit (MTU0 Output Disabling Condition POE10F Add)

Adds the ICSR4.POE10F flag to the output disabling control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC5ADDMT0ZE Bit (MTU0 Output Disabling Condition POE11F Add)

Adds the ICSR5.POE11F flag to the output disabling control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC6ADDMT0ZE Bit (MTU0 Output Disabling Condition POE12F Add)

Adds the ICSR7.POE12F flag to the output disabling control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

20.2.17 Port Output Enable Control Register 6 (POECR6)

Address(es): POE.POECR6 0009 E414h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	IC6ADDGPT01ZE	IC5ADDGPT01ZE	IC4ADDGPT01ZE	IC3ADDGPT01ZE	—	IC1ADDGPT01ZE	CMADDGPT01ZE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMADDGPT01ZE	GPTW0 and GPTW1 Output Disabling Condition CFLAG Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b1	IC1ADDGPT01ZE	GPTW0 and GPTW1 Output Disabling Condition POE0F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	IC3ADDGPT01ZE	GPTW0 and GPTW1 Output Disabling Condition POE8F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b4	IC4ADDGPT01ZE	GPTW0 and GPTW1 Output Disabling Condition POE10F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b5	IC5ADDGPT01ZE	GPTW0 and GPTW1 Output Disabling Condition POE11F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b6	IC6ADDGPT01ZE	GPTW0 and GPTW1 Output Disabling Condition POE12F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b15 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECR6 register is used to set the control conditions to disable the output of the GPTW0 and GPTW1 pins.

CMADDGPT01ZE Bit (GPTW0 and GPTW1 Output Disabling Condition CFLAG Add)

Adds the POECMPFR.CjFLAG flag (j = 0 to 2) to the output disabling control conditions for the GPTW0 and GPTW1 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B).

However, when the pins are disabled by the flag, an OEIk interrupt (k = 1, 3 to 5) will not be generated.

IC1ADDGPT01ZE Bit (GPTW0 and GPTW1 Output Disabling Condition POE0F Add)

Adds the ICSR1.POE0F flag to the output disabling control conditions for the GPTW0 and GPTW1 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B).

IC3ADDGPT01ZE Bit (GPTW0 and GPTW1 Output Disabling Condition POE8F Add)

Adds the ICSR3.POE8F flag to the output disabling control conditions for the GPTW0 and GPTW1 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B).

IC4ADDGPT01ZE Bit (GPTW0 and GPTW1 Output Disabling Condition POE10F Add)

Adds the ICSR4.POE10F flag to the output disabling control conditions for the GPTW0 and GPTW1 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B).

IC5ADDGPT01ZE Bit (GPTW0 and GPTW1 Output Disabling Condition POE11F Add)

Adds the ICSR5.POE11F flag to the output disabling control conditions for the GPTW0 and GPTW1 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B).

IC6ADDGPT01ZE Bit (GPTW0 and GPTW1 Output Disabling Condition POE12F Add)

Adds the ICSR7.POE12F flag to the output disabling control conditions for the GPTW0 and GPTW1 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B).

20.2.18 Port Output Enable Control Register 6B (POECR6B)

Address(es): POE.POECR6B 0009 E450h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	IC6ADDGPT23ZE	IC5ADDGPT23ZE	IC4ADDGPT23ZE	IC3ADDGPT23ZE	—	IC1ADDGPT23ZE	CMADDGPT23ZE
Value after reset:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMADDGPT23ZE	GPTW2 Output Disabling Condition CFLAG Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b1	IC1ADDGPT23ZE	GPTW2 Output Disabling Condition POE0F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	IC3ADDGPT23ZE	GPTW2 Output Disabling Condition POE8F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b4	IC4ADDGPT23ZE	GPTW2 Output Disabling Condition POE10F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b5	IC5ADDGPT23ZE	GPTW2 Output Disabling Condition POE11F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b6	IC6ADDGPT23ZE	GPTW2 Output Disabling Condition POE12F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b15 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECR6B register is used to set the control conditions to disable the output of the GPTW2 pins.

CMADDGPT23ZE Bit (GPTW2 Output Disabling Condition CFLAG Add)

Adds the POECMPFR.CjFLAG flag (j = 0 to 2) to the output disabling control conditions for the GPTW2 pins (GTIOC2A, GTIOC2B).

However, when the pins are disabled by the flag, an OEIk interrupt (k = 1, 3 to 5) will not be generated.

IC1ADDGPT23ZE Bit (GPTW2 Output Disabling Condition POE0F Add)

Adds the ICSR1.POE0F flag to the output disabling control conditions for the GPTW2 pins (GTIOC2A, GTIOC2B).

IC3ADDGPT23ZE Bit (GPTW2 Output Disabling Condition POE8F Add)

Adds the ICSR3.POE8F flag to the output disabling control conditions for the GPTW2 pins (GTIOC2A, GTIOC2B).

IC4ADDGPT23ZE Bit (GPTW2 Output Disabling Condition POE10F Add)

Adds the ICSR4.POE10F flag to the output disabling control conditions for the GPTW2 pins (GTIOC2A, GTIOC2B).

IC5ADDGPT23ZE Bit (GPTW2 Output Disabling Condition POE11F Add)

Adds the ICSR5.POE11F flag to the output disabling control conditions for the GPTW2 pins (GTIOC2A, GTIOC2B).

IC6ADDGPT23ZE Bit (GPTW2 Output Disabling Condition POE12F Add)

Adds the ICSR7.POE12F flag to the output disabling control conditions for the GPTW2 pins (GTIOC2A, GTIOC2B).

20.2.19 Port Output Enable Control Register 9 (POECR9)

Address(es): POE.POECR9 0009 E452h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	IC6ADDGPT02ZE	IC5ADDGPT02ZE	IC4ADDGPT02ZE	IC3ADDGPT02ZE	—	IC1ADDGPT02ZE	CMADDGPT02ZE
Value after reset:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMADDGPT02ZE	GPTW0 to GPTW2 Output Disabling Condition CFLAG Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b1	IC1ADDGPT02ZE	GPTW0 to GPTW2 Output Disabling Condition POE0F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	IC3ADDGPT02ZE	GPTW0 to GPTW2 Output Disabling Condition POE8F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b4	IC4ADDGPT02ZE	GPTW0 to GPTW2 Output Disabling Condition POE10F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b5	IC5ADDGPT02ZE	GPTW0 to GPTW2 Output Disabling Condition POE11F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b6	IC6ADDGPT02ZE	GPTW0 to GPTW2 Output Disabling Condition POE12F Add	0: Does not add the flags to the conditions to disable the output. 1: Adds the flags to the conditions to disable the output.	R/W*1
b15 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECR9 register is used to set the control conditions to disable the output of the GPTW0 to GPTW2 pins.

CMADDGPT02ZE Bit (GPTW0 to GPTW2 Output Disabling Condition CFLAG Add)

Adds the POECMPFR.CjFLAG flag (j = 0 to 2) to the output disabling control conditions for the GPTW0 to GPTW2 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B, GTIOC2A, GTIOC2B).

However, when the pins are disabled by the flag, an OEIk interrupt (k = 1, 3 to 5) will not be generated.

IC1ADDGPT02ZE Bit (GPTW0 to GPTW2 Output Disabling Condition POE0F Add)

Adds the ICSR1.POE0F flag to the output disabling control conditions for the GPTW0 to GPTW2 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B, GTIOC2A, GTIOC2B).

IC3ADDGPT02ZE Bit (GPTW0 to GPTW2 Output Disabling Condition POE8F Add)

Adds the ICSR3.POE8F flag to the output disabling control conditions for the GPTW0 to GPTW2 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B, GTIOC2A, GTIOC2B).

IC4ADDGPT02ZE Bit (GPTW0 to GPTW2 Output Disabling Condition POE10F Add)

Adds the ICSR4.POE10F flag to the output disabling control conditions for the GPTW0 to GPTW2 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B, GTIOC2A, GTIOC2B).

IC5ADDGPT02ZE Bit (GPTW0 to GPTW2 Output Disabling Condition POE11F Add)

Adds the ICSR5.POE11F flag to the output disabling control conditions for the GPTW0 to GPTW2 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B, GTIOC2A, GTIOC2B).

IC6ADDGPT02ZE Bit (GPTW0 to GPTW2 Output Disabling Condition POE12F Add)

Adds the ICSR7.POE12F flag to the output disabling control conditions for the GPTW0 to GPTW2 pins (GTIOC0A, GTIOC0B, GTIOC1A, GTIOC1B, GTIOC2A, GTIOC2B).

20.2.20 Port Mode Mask Control Register 0 (PMMCR0)

Address(es): POE.PMMCR0 0009 E430h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	MTU0DME	MTU0CME	MTU0BME	MTU0AME
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MTU0AME	MTIOC0A Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b1	MTU0BME	MTIOC0B Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b2	MTU0CME	MTIOC0C Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b3	MTU0DME	MTIOC0D Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b15 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The PMMCR0 register is used to mask the settings of the PMR register related to the MTU0 pins.

MTU0AME Bit (MTIOC0A Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC0A pin to the general I/O port when one of the following flags and bit is set to 1: the SPOER.MTUCH0HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag (n = 1, 3 to 5, 7; m = 0, 8, 10, 11, 12) additionally selected in the POECR5 register, and the POECMPFR.CjFLAG flag (j = 0 to 2).

When setting this bit to 1, set the POECR1.MTU0AZE bit to 0. When the POECR1.MTU0AZE bit is 1, the setting of the bit is ignored.

MTU0BME Bit (MTIOC0B Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC0B pin to the general I/O port when one of the following flags and bit is set to 1: the SPOER.MTUCH0HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR5 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR1.MTU0BZE bit to 0. When the POECR1.MTU0BZE bit is 1, the setting of the bit is ignored.

MTU0CME Bit (MTIOC0C Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC0C pin to the general I/O port when one of the following flags and bit is set to 1: the SPOER.MTUCH0HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR5 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR1.MTU0CZE bit to 0. When the POECR1.MTU0CZE bit is 1, the setting of the bit is ignored.

MTU0DME Bit (MTIOC0D Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC0D pin to the general I/O port when one of the following flags and bit is set to 1: the SPOER.MTUCH0HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR5 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR1.MTU0DZE bit to 0. When the POECR1.MTU0DZE bit is 1, the setting of the

bit is ignored.

20.2.21 Port Mode Mask Control Register 1 (PMMCR1)

Address(es): POE.PMMCR1 0009 E432h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	MTU3DME	MTU4CME	MTU4DME	MTU3BME	MTU4AME	MTU4BME	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	MTU4BME	MTIOC4B Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b9	MTU4AME	MTIOC4A Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b10	MTU3BME	MTIOC3B Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b11	MTU4DME	MTIOC4D Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b12	MTU4CME	MTIOC4C Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b13	MTU3DME	MTIOC3D Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The PMMCR1 register is used to mask the settings of the PMR register related to the MTU complementary PWM output pins (MTU3, MTU4).

MTU4BME Bit (MTIOC4B Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC4B pin to the general I/O port when one of the following flags and bit is set to 1: the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR2.MTU4BDZE bit to 0. When the POECR2.MTU4BDZE bit is 1, the setting of the bit is ignored.

MTU4AME Bit (MTIOC4A Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC4A pin to the general I/O port when one of the following flags and bit is set to 1: the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR2.MTU4ACZE bit to 0. When the POECR2.MTU4ACZE bit is 1, the setting of the bit is ignored.

MTU3BME Bit (MTIOC3B Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC3B pin to the general I/O port when one of the following flags and bit is set to 1: the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR2.MTU3BDZE bit to 0. When the POECR2.MTU3BDZE bit is 1, the setting of

the bit is ignored.

MTU4DME Bit (MTIOC4D Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC4D pin to the general I/O port when one of the following flags and bit is set to 1: the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR2.MTU4BDZE bit to 0. When the POECR2.MTU4BDZE bit is 1, the setting of the bit is ignored.

MTU4CME Bit (MTIOC4C Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC4C pin to the general I/O port when one of the following flags and bit is set to 1: the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR2.MTU4ACZE bit to 0. When the POECR2.MTU4ACZE bit is 1, the setting of the bit is ignored.

MTU3DME Bit (MTIOC3D Pin Port Mode Mask Enable)

This bit specifies whether switching the MTIOC3D pin to the general I/O port when one of the following flags and bit is set to 1: the OCSR1.OSF1 flag (when the OCSR1.OCE1 bit is 1), the SPOER.MTUCH34HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit is 1), the ICSRn.POE_mF flag additionally selected in the POECR4 register, and the POECMPFR.CjFLAG flag.

When setting this bit to 1, set the POECR2.MTU3BDZE bit to 0. When the POECR2.MTU3BDZE bit is 1, the setting of the bit is ignored.

20.2.22 Port Mode Mask Control Register 2 (PMMCR2)

Address(es): POE.PMMCR2 0009 E434h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	GPT2B ME	GPT2A ME	GPT1B ME	GPT1A ME	GPT0B ME	GPT0A ME
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	GPT0AME	GTIOC0A Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b1	GPT0BME	GTIOC0B Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b2	GPT1AME	GTIOC1A Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b3	GPT1BME	GTIOC1B Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b4	GPT2AME	GTIOC2A Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b5	GPT2BME	GTIOC2B Pin Port Mode Mask Enable	0: Switching to the general I/O port pin is disabled. 1: Switching to the general I/O port pin is enabled.	R/W*1
b15 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The PMMCR2 register is used to mask the settings of the PMR register related to the GPTW output pins (GPTW0 to GPTW2).

GPT0AME Bit (GTIOC0A Pin Port Mode Mask Enable)

This bit determines whether to switch the GTIOC0A pin to the general I/O port when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit = 1), the SPOER.GPT01HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag (n = 1, 3 to 5, 7; m = 0, 8, 10, 11, 12) additionally selected in the POECR6 register, POECMPFR.CjFLAG flag (j = 0 to 2), the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the POECMPFR.CjFLAG flag.

To set this bit to 1, set the POECR3.GPT0ABZE bit to 0. If the POECR3.GPT0ABZE bit is 1, setting of this bit is ignored.

GPT0BME Bit (GTIOC0B Pin Port Mode Mask Enable)

This bit determines whether to switch the GTIOC0B pin to the general I/O port when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit = 1), the SPOER.GPT01HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag additionally selected in the POECR6 register, POECMPFR.CjFLAG flag, the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the POECMPFR.CjFLAG flag.

To make this bit 1, set the POECR3.GPT0ABZE bit to 0. If the POECR3.GPT0ABZE bit is 1, setting of this bit is ignored.

GPT1AME Bit (GTIOC1A Pin Port Mode Mask Enable)

This bit determines whether to switch the GTIOC1A pin to the general I/O port when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit = 1), the SPOER.GPT01HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag additionally selected in the POECR6 register, POECMPFR.CjFLAG flag, the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the

POECMPFR.CjFLAG flag.

To make this bit 1, set the POECR3.GPT1ABZE bit to 0. If the POECR3.GPT1ABZE bit is 1, setting of this bit is ignored.

GPT1BME Bit (GTIOC1B Pin Port Mode Mask Enable)

This bit determines whether to switch the GTIOC1B pin to the general I/O port when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit = 1), the SPOER.GPT01HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag additionally selected in the POECR6 register, POECMPFR.CjFLAG flag, the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the POECMPFR.CjFLAG flag.

To make this bit 1, set the POECR3.GPT1ABZE bit to 0. If the POECR3.GPT1ABZE bit is 1, setting of this bit is ignored.

GPT2AME Bit (GTIOC2A Pin Port Mode Mask Enable)

This bit determines whether to switch the GTIOC2A pin to the general I/O port when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit = 1), the SPOER.GPT23HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag additionally selected in the POECR6B register, POECMPFR.CjFLAG flag, the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the POECMPFR.CjFLAG flag.

To make this bit 1, set the POECR3.GPT2ABZE bit to 0. If the POECR3.GPT2ABZE bit is 1, setting of this bit is ignored.

GPT2BME Bit (GTIOC2B Pin Port Mode Mask Enable)

This bit determines whether to switch the GTIOC2B pin to the general I/O port when at least one of the followings is set to 1: the OCSR3.OSF3 flag (when the OCSR3.OCE3 bit = 1), the SPOER.GPT23HIZ bit, the SPOER.GPT02HIZ bit, the ICSR6.OSTSTF flag (when the ICSR6.OSTSTE bit = 1), the ICSRn.POE_mF flag additionally selected in the POECR6B register, POECMPFR.CjFLAG flag, the ICSRn.POE_mF flag additionally selected in the POECR9 register, and the POECMPFR.CjFLAG flag.

To make this bit 1, set the POECR3.GPT2ABZE bit to 0. If the POECR3.GPT2ABZE bit is 1, setting of this bit is ignored.

20.2.23 Port Output Enable Comparator Output Detection Flag Register (POECMPFR)

Address(es): POE.POECMPFR 0009 E426h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	C2FLAG	C1FLAG	C0FLAG
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	C0FLAG	Comparator Channel 0 Output Detection Flag	0: Comparator output not detected 1: Comparator output detected	R/(W) *1
b1	C1FLAG	Comparator Channel 1 Output Detection Flag	0: Comparator output not detected 1: Comparator output detected	R/(W) *1
b2	C2FLAG	Comparator Channel 2 Output Detection Flag	0: Comparator output not detected 1: Comparator output detected	R/(W) *1
b15 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The flag can only be set to 0 by writing 0 after reading 1.

CjFLAG Flag (Comparator Channel j Output Detection Flag) (j = 0 to 2)

This flag indicates whether each comparator output is detected or not detected.

[Setting condition]

- A change from low level to high level in the comparator output is detected.
 - When the comparator is set to non-inverted output, the input voltage changes from lower to higher than the reference voltage
 - When the comparator is set to inverted output, the input voltage changes from higher to lower than the reference voltage

[Clearing condition]

- By writing 0 to the CjFLAG flag after reading CjFLAG = 1

20.2.24 Port Output Enable Comparator Request Select Register (POECMPSEL)

Address(es): POE.POECMPSEL 0009 E428h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	POERE Q2	POERE Q1	POERE Q0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	POEREQ0	Comparator Channel 0 Output Disabling Request Enable	0: Disables output disabling request generation upon comparator output detection. 1: Enables output disabling request generation upon comparator output detection.	R/W*1
b1	POEREQ1	Comparator Channel 1 Output Disabling Request Enable	0: Disables output disabling request generation upon comparator output detection. 1: Enables output disabling request generation upon comparator output detection.	R/W*1
b2	POEREQ2	Comparator Channel 2 Output Disabling Request Enable	0: Disables output disabling request generation upon comparator output detection. 1: Enables output disabling request generation upon comparator output detection.	R/W*1
b15 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECMPSEL register sets a comparator output detection flag to use as a control condition to disable the outputs.

POEREQj Bit (Comparator Channel j Output Disabling Request Enable) (j = 0 to 2)

This bit disables or enables output disabling request generation in response to each comparator output detection. An output disabling request is generated when one of the comparator outputs is detected.

20.2.25 Port Output Enable Comparator Request Extended Selection Register m (POECMPEXm) (m = 0, 2, 4 to 6)

Address(es): POE.POECMPEX0 0009 E438h, POE.POECMPEX2 0009 E43Ah, POE.POECMPEX4 0009 E43Ch, POE.POECMPEX5 0009 E43Dh, POE.POECMPEX6 0009 E458h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	POERE Q2	POERE Q1	POERE Q0
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	POEREQ0	Comparator Channel 0 Output Disabling Request Enable	0: Disables output disabling request generation upon comparator output detection. 1: Enables output disabling request generation upon comparator output detection.	R/W*1
b1	POEREQ1	Comparator Channel 1 Output Disabling Request Enable	0: Disables output disabling request generation upon comparator output detection. 1: Enables output disabling request generation upon comparator output detection.	R/W*1
b2	POEREQ2	Comparator Channel 2 Output Disabling Request Enable	0: Disables output disabling request generation upon comparator output detection. 1: Enables output disabling request generation upon comparator output detection.	R/W*1
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The POECMPEXm register specifies the comparator output detection flag to use as a control condition to disable the outputs.

The register can select a source for each output pin group where as the POECMPSEL register selects the source for all pins. Table 20.4 lists the output pin groups and the corresponding POECMPEXm registers.

Table 20.4 Output Pin Groups and Corresponding POECMPEXm Registers

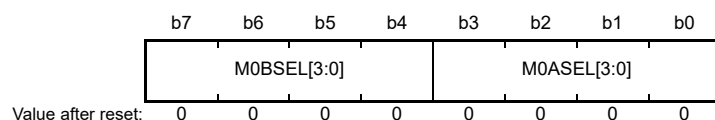
Output Pin Group	Corresponding POECMPEXm Register	Corresponding CFLAG Add Bit
MTU3, MTU4	POECMPEX0	POECR4.CMADDMT34ZE
MTU0	POECMPEX2	POECR5.CMADDMT0ZE
GPTW0, GPTW1	POECMPEX4	POECR6.CMADDGPT01ZE
GPTW2	POECMPEX5	POECR6B.CMADDGPT23ZE
GPTW0 to GPTW2	POECMPEX6	POECR9.CMADDGPT02ZE

POEREQj Bit (Comparator Channel j Output Disabling Request Enable) (j = 0 to 2)

This bit disables or enables output disabling request generation in response to each comparator output detection. An output disabling request is generated when one of the comparator outputs is detected.

20.2.26 MTU0 Pin Select Register 1 (M0SELR1)

Address(es): POE.M0SELR1 0009 E460h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	M0ASEL[3:0]	MTU0-A (MTIOC0A) Pin Select	b3 b0 0 0 0 0: Does not perform the output disabling control for any MTIOC0A pin. 0 0 0 1: Perform the output disabling control for the P30 as the MTIOC0A pin. 0 0 1 0: Perform the output disabling control for the P36 as the MTIOC0A pin. 0 0 1 1: Perform the output disabling control for the PB1 as the MTIOC0A pin. 0 1 0 0: Perform the output disabling control for the PD3 as the MTIOC0A pin. Settings other than above are prohibited.	R/W*1
b7 to b4	M0BSEL[3:0]	MTU0-B (MTIOC0B) Pin Select	b7 b4 0 0 0 0: Does not perform the output disabling control for any MTIOC0B pin. 0 0 0 1: Perform the output disabling control for the P25 as the MTIOC0B pin. 0 0 1 0: Perform the output disabling control for the P92 as the MTIOC0B pin. 0 0 1 1: Perform the output disabling control for the PB0 as the MTIOC0B pin. 0 1 0 0: Perform the output disabling control for the PD4 as the MTIOC0B pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset.

The M0SELR1 register is an 8-bit readable/writable register, and selects relevant pins to the output disabling control from the MTU0-A and MTU0-B pins respectively.

M0ASEL[3:0] Bits (MTU0-A (MTIOC0A) Pin Select)

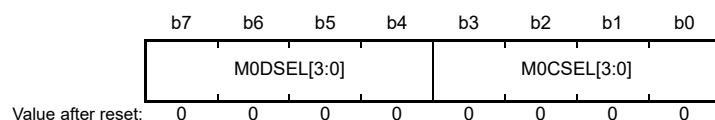
Selects a relevant MTIOC0A pin to the output disabling control.

M0BSEL[3:0] Bits (MTU0-B (MTIOC0B) Pin Select)

Selects a relevant MTIOC0B pin to the output disabling control.

20.2.27 MTU0 Pin Select Register 2 (M0SELR2)

Address(es): POE.M0SELR2 0009 E461h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	M0CSEL[3:0]	MTU0-C (MTIOC0C) Pin Select	b3b0 0 0 0 0: Does not perform the output disabling control for any MTIOC0C pin. 0 0 0 1: Perform the output disabling control for the P33 as the MTIOC0C pin. 0 0 1 0: Perform the output disabling control for the P93 as the MTIOC0C pin. 0 0 1 1: Perform the output disabling control for the P97 as the MTIOC0C pin. 0 1 0 0: Perform the output disabling control for the PD5 as the MTIOC0C pin. Settings other than above are prohibited.	R/W*1
b7 to b4	M0DSEL[3:0]	MTU0-D (MTIOC0D) Pin Select	b7b4 0 0 0 0: Does not perform the output disabling control for any MTIOC0D pin. 0 0 0 1: Perform the output disabling control for the P32 as the MTIOC0D pin. 0 0 1 0: Perform the output disabling control for the P96 as the MTIOC0D pin. 0 0 1 1: Perform the output disabling control for the PD6 as the MTIOC0D pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset.

The M0SELR2 register is an 8-bit readable/writable register, and selects relevant pins to the output disabling control from the MTU0-C and MTU0-D pins respectively.

M0CSEL[3:0] Bits (MTU0-C (MTIOC0C) Pin Select)

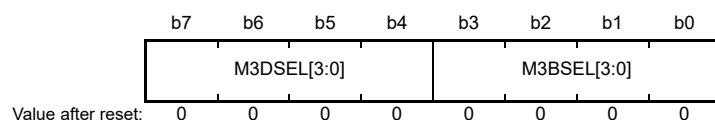
Selects a relevant MTIOC0C pin to the output disabling control.

M0DSEL[3:0] Bits (MTU0-D (MTIOC0D) Pin Select)

Selects a relevant MTIOC0D pin to the output disabling control.

20.2.28 MTU3 Pin Select Register (M3SELR)

Address(es): POE.M3SELR 0009 E462h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	M3BSEL[3:0]	MTU3-B (MTIOC3B) Pin Select	b3 b0 0 0 0 0: Does not perform the output disabling control for any MTIOC3B pin. 0 0 0 1: Perform the output disabling control for the P26 as the MTIOC3B pin. 0 0 1 0: Perform the output disabling control for the P71 as the MTIOC3B pin. 0 0 1 1: Perform the output disabling control for the P94 as the MTIOC3B pin. Settings other than above are prohibited.	R/W*1
b7 to b4	M3DSEL[3:0]	MTU3-D (MTIOC3D) Pin Select	b7 b4 0 0 0 0: Does not perform the output disabling control for any MTIOC3D pin. 0 0 0 1: Perform the output disabling control for the P74 as the MTIOC3D pin. 0 0 1 0: Perform the output disabling control for the P97 as the MTIOC3D pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset.

The M3SELR register is an 8-bit readable/writable register, and selects relevant pins to the output disabling control from the MTU3-B and MTU3-D pins respectively.

M3BSEL[3:0] Bits (MTU3-B (MTIOC3B) Pin Select)

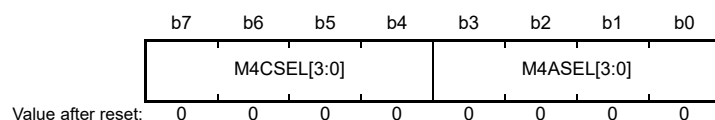
Selects a relevant MTIOC3B pin to the output disabling control.

M3DSEL[3:0] Bits (MTU3-D (MTIOC3D) Pin Select)

Selects a relevant MTIOC3D pin to the output disabling control.

20.2.29 MTU4 Pin Select Register 1 (M4SELR1)

Address(es): POE.M4SELR1 0009 E463h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	M4ASEL[3:0]	MTU4-A (MTIOC4A) Pin Select	b3 b0 0 0 0 0: Does not perform the output disabling control for any MTIOC4A pin. 0 0 0 1: Perform the output disabling control for the P31 as the MTIOC4A pin. 0 0 1 0: Perform the output disabling control for the P36 as the MTIOC4A pin. 0 0 1 1: Perform the output disabling control for the P72 as the MTIOC4A pin. 0 1 0 0: Perform the output disabling control for the P95 as the MTIOC4A pin. 0 1 0 1: Perform the output disabling control for the PD3 as the MTIOC4A pin. Settings other than above are prohibited.	R/W*1
b7 to b4	M4CSEL[3:0]	MTU4-C (MTIOC4C) Pin Select	b7 b4 0 0 0 0: Does not perform the output disabling control for any MTIOC4C pin. 0 0 0 1: Perform the output disabling control for the P11 as the MTIOC4C pin. 0 0 1 0: Perform the output disabling control for the P75 as the MTIOC4C pin. 0 0 1 1: Perform the output disabling control for the PB0 as the MTIOC4C pin. 0 1 0 0: Perform the output disabling control for the PD5 as the MTIOC4C pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset.

The M4SELR1 register is an 8-bit readable/writable register, and selects relevant pins to the output disabling control from the MTU4-A and MTU4-C pins respectively.

M4ASEL[3:0] Bits (MTU4-A (MTIOC4A) Pin Select)

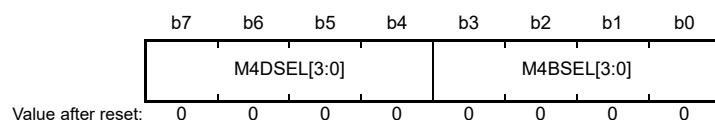
Selects a relevant MTIOC4A pin to the output disabling control.

M4CSEL[3:0] Bits (MTU4-C (MTIOC4C) Pin Select)

Selects a relevant MTIOC4C pin to the output disabling control.

20.2.30 MTU4 Pin Select Register 2 (M4SELR2)

Address(es): POE.M4SELR2 0009 E464h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	M4BSEL[3:0]	MTU4-B (MTIOC4B) Pin Select	b3 b0 0 0 0 0: Does not perform the output disabling control for any MTIOC4B pin. 0 0 0 1: Perform the output disabling control for the P25 as the MTIOC4B pin. 0 0 1 0: Perform the output disabling control for the P73 as the MTIOC4B pin. 0 0 1 1: Perform the output disabling control for the P96 as the MTIOC4B pin. 0 1 0 0: Perform the output disabling control for the PD4 as the MTIOC4B pin. Settings other than above are prohibited.	R/W*1
b7 to b4	M4DSEL[3:0]	MTU4-D (MTIOC4D) Pin Select	b7 b4 0 0 0 0: Does not perform the output disabling control for any MTIOC4D pin. 0 0 0 1: Perform the output disabling control for the P76 as the MTIOC4D pin. 0 0 1 0: Perform the output disabling control for the PB1 as the MTIOC4D pin. 0 0 1 1: Perform the output disabling control for the PD6 as the MTIOC4D pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset.

The M4SELR2 register is an 8-bit readable/writable register, and selects relevant pins to the output disabling control from the MTU4-B and MTU4-D pins respectively.

M4BSEL[3:0] Bits (MTU4-B (MTIOC4B) Pin Select)

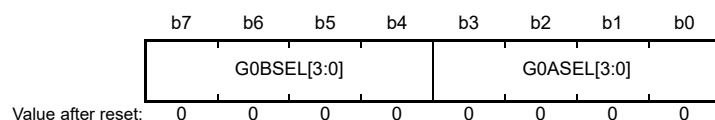
Selects a relevant MTIOC4B pin to the output disabling control.

M4DSEL[3:0] Bits (MTU4-D (MTIOC4D) Pin Select)

Selects a relevant MTIOC4D pin to the output disabling control.

20.2.31 GPTW0 Pin Select Register (G0SELR)

Address(es): POE.G0SELR 0009 E46Ah



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	G0ASEL[3:0]	GPTW0-A (GTIOC0A) Pin Select	b3b0 0 0 0 0: Does not perform the output disabling control for any GTIOC0A pin. 0 0 0 1: Perform the output disabling control for the P26 as the GTIOC0A pin. 0 0 1 0: Perform the output disabling control for the P36 as the GTIOC0A pin. 0 0 1 1: Perform the output disabling control for the P71 as the GTIOC0A pin. 0 1 0 0: Perform the output disabling control for the P94 as the GTIOC0A pin. Settings other than above are prohibited.	R/W*1
b7 to b4	G0BSEL[3:0]	GPTW0-B (GTIOC0B) Pin Select	b7b4 0 0 0 0: Does not perform the output disabling control for any GTIOC0B pin. 0 0 0 1: Perform the output disabling control for the P31 as the GTIOC0B pin. 0 0 1 0: Perform the output disabling control for the P74 as the GTIOC0B pin. 0 0 1 1: Perform the output disabling control for the P97 as the GTIOC0B pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset.

The G0SELR register is an 8-bit readable/writable register, and selects relevant pins to the output disabling control from the GPTW0-A and GPTW0-B pins respectively.

G0ASEL[3:0] Bits (GPTW0-A (GTIOC0A) Pin Select)

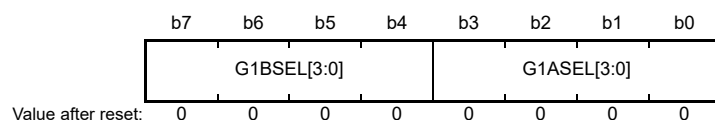
Selects a relevant GTIOC0A pin to the output disabling control.

G0BSEL[3:0] Bits (GPTW0-B (GTIOC0B) Pin Select)

Selects a relevant GTIOC0B pin to the output disabling control.

20.2.32 GPTW1 Pin Select Register (G1SELR)

Address(es): POE.G1SELR 0009 E46Bh



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	G1ASEL[3:0]	GPTW1-A (GTIOC1A) Pin Select	b3 b0 0 0 0 0: Does not perform the output disabling control for any GTIOC1A pin. 0 0 0 1: Perform the output disabling control for the P31 as the GTIOC1A pin. 0 0 1 0: Perform the output disabling control for the P72 as the GTIOC1A pin. 0 0 1 1: Perform the output disabling control for the P92 as the GTIOC1A pin. 0 1 0 0: Perform the output disabling control for the P95 as the GTIOC1A pin. 0 1 0 1: Perform the output disabling control for the PB5 as the GTIOC1A pin. 0 1 1 0: Perform the output disabling control for the PD3 as the GTIOC1A pin. Settings other than above are prohibited.	R/W*1
b7 to b4	G1BSEL[3:0]	GPTW1-B (GTIOC1B) Pin Select	b7 b4 0 0 0 0: Does not perform the output disabling control for any GTIOC1B pin. 0 0 0 1: Perform the output disabling control for the P75 as the GTIOC1B pin. 0 0 1 0: Perform the output disabling control for the P93 as the GTIOC1B pin. 0 0 1 1: Perform the output disabling control for the PB0 as the GTIOC1B pin. 0 1 0 0: Perform the output disabling control for the PD4 as the GTIOC1B pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset.

The G1SELR register is an 8-bit readable/writable register, and selects relevant pins to the output disabling control from the GPTW1-A and GPTW1-B pins respectively.

G1ASEL[3:0] Bits (GPTW1-A (GTIOC1A) Pin Select)

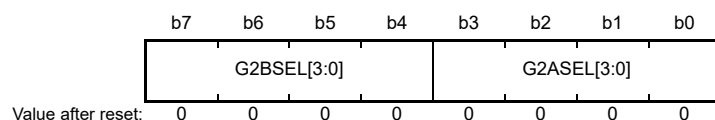
Selects a relevant GTIOC1A pin to the output disabling control.

G1BSEL[3:0] Bits (GPTW1-B (GTIOC1B) Pin Select)

Selects a relevant GTIOC1B pin to the output disabling control.

20.2.33 GPTW2 Pin Select Register (G2SELR)

Address(es): POE.G2SELR 0009 E46Ch



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	G2ASEL[3:0]	GPTW2-A (GTIOC2A) Pin Select	b3 b0 0 0 0 0: Does not perform the output disabling control for any GTIOC2A pin. 0 0 0 1: Perform the output disabling control for the P04 as the GTIOC2A pin. 0 0 1 0: Perform the output disabling control for the P23 as the GTIOC2A pin. 0 0 1 1: Perform the output disabling control for the P73 as the GTIOC2A pin. 0 1 0 0: Perform the output disabling control for the P96 as the GTIOC2A pin. 0 1 0 1: Perform the output disabling control for the PB5 as the GTIOC2A pin. 0 1 1 0: Perform the output disabling control for the PD5 as the GTIOC2A pin. Settings other than above are prohibited.	R/W*1
b7 to b4	G2BSEL[3:0]	GPTW2-B (GTIOC2B) Pin Select	b7 b4 0 0 0 0: Does not perform the output disabling control for any GTIOC2B pin. 0 0 0 1: Perform the output disabling control for the P03 as the GTIOC2B pin. 0 0 1 0: Perform the output disabling control for the P24 as the GTIOC2B pin. 0 0 1 1: Perform the output disabling control for the P76 as the GTIOC2B pin. 0 1 0 0: Perform the output disabling control for the PB1 as the GTIOC2B pin. 0 1 0 1: Perform the output disabling control for the PB6 as the GTIOC2B pin. 0 1 1 0: Perform the output disabling control for the PD6 as the GTIOC2B pin. Settings other than above are prohibited.	R/W*1

Note 1. Can be modified only once after a reset.

The G2SELR register is an 8-bit readable/writable register, and selects relevant pins to the output disabling control from the GPTW2-A and GPTW2-B pins respectively.

G2ASEL[3:0] Bits (GPTW2-A (GTIOC2A) Pin Select)

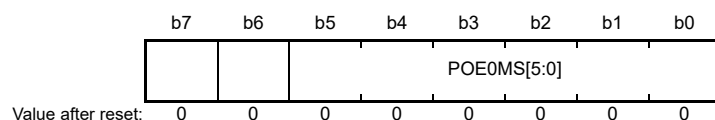
Selects a relevant GTIOC2A pin to the output disabling control.

G2BSEL[3:0] Bits (GPTW2-B (GTIOC2B) Pin Select)

Selects a relevant GTIOC2B pin to the output disabling control.

20.2.34 Input Signal Mask Control Register 0 (IMCR0)

Address(es): POE.IMCR0 0009 E45Ch



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	POE0MS[5:0]	POE0 Mask Signal Select	b5 b0 000000: Does not mask the POE0# signal 000001: Mask the POE0# signal with the MTIOC0A signal 000010: Mask the POE0# signal with the MTIOC0B signal 000011: Mask the POE0# signal with the MTIOC0C signal 000100: Mask the POE0# signal with the MTIOC0D signal 000101: Mask the POE0# signal with the MTIOC1A signal 000110: Mask the POE0# signal with the MTIOC1B signal 000111: Mask the POE0# signal with the MTIOC2A signal 001000: Mask the POE0# signal with the MTIOC2B signal 001001: Mask the POE0# signal with the MTIOC3A signal 001010: Mask the POE0# signal with the MTIOC3B signal 001011: Mask the POE0# signal with the MTIOC3C signal 001100: Mask the POE0# signal with the MTIOC3D signal 001101: Mask the POE0# signal with the MTIOC4A signal 001110: Mask the POE0# signal with the MTIOC4B signal 001111: Mask the POE0# signal with the MTIOC4C signal 010000: Mask the POE0# signal with the MTIOC4D signal 011101: Mask the POE0# signal with the GTIOC0A signal 011110: Mask the POE0# signal with the GTIOC0B signal 011111: Mask the POE0# signal with the GTIOC1A signal 100000: Mask the POE0# signal with the GTIOC1B signal 100001: Mask the POE0# signal with the GTIOC2A signal 100010: Mask the POE0# signal with the GTIOC2B signal Settings other than above are prohibited.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

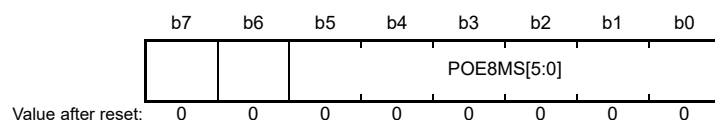
The IMCR0 register is an 8-bit readable/writable register used to mask the output disabling requests by the POE0# pin.

POE0MS[5:0] Bits (POE0 Mask Signal Select)

These bits are used to select a signal that masks the POE0# signal. When the selected pin drives high, the POE0# signal is masked.

20.2.35 Input Signal Mask Control Register 2 (IMCR2)

Address(es): POE.IMCR2 0009 E45Eh



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	POE8MS[5:0]	POE8 Mask Signal Select	b5 b0 000000: Does not mask the POE8# signal 000001: Mask the POE8# signal with the MTIOC0A signal 000010: Mask the POE8# signal with the MTIOC0B signal 000011: Mask the POE8# signal with the MTIOC0C signal 000100: Mask the POE8# signal with the MTIOC0D signal 000101: Mask the POE8# signal with the MTIOC1A signal 000110: Mask the POE8# signal with the MTIOC1B signal 000111: Mask the POE8# signal with the MTIOC2A signal 001000: Mask the POE8# signal with the MTIOC2B signal 001001: Mask the POE8# signal with the MTIOC3A signal 001010: Mask the POE8# signal with the MTIOC3B signal 001011: Mask the POE8# signal with the MTIOC3C signal 001100: Mask the POE8# signal with the MTIOC3D signal 001101: Mask the POE8# signal with the MTIOC4A signal 001110: Mask the POE8# signal with the MTIOC4B signal 001111: Mask the POE8# signal with the MTIOC4C signal 010000: Mask the POE8# signal with the MTIOC4D signal 011101: Mask the POE8# signal with the GTIOC0A signal 011110: Mask the POE8# signal with the GTIOC0B signal 011111: Mask the POE8# signal with the GTIOC1A signal 100000: Mask the POE8# signal with the GTIOC1B signal 100001: Mask the POE8# signal with the GTIOC2A signal 100010: Mask the POE8# signal with the GTIOC2B signal Settings other than above are prohibited.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

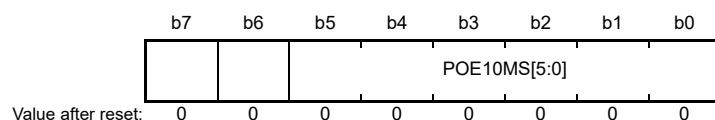
The IMCR2 register is an 8-bit readable/writable register used to mask the output disabling requests by the POE8# pin.

POE8MS[5:0] Bits (POE8 Mask Signal Select)

These bits are used to select a signal that masks the POE8# signal. When the selected pin drives high, the POE8# signal is masked.

20.2.36 Input Signal Mask Control Register 3 (IMCR3)

Address(es): POE.IMCR3 0009 E45Fh



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	POE10MS[5:0]	POE10 Mask Signal Select	b5 b0 000000: Does not mask the POE10# signal 000001: Mask the POE10# signal with the MTIOC0A signal 000010: Mask the POE10# signal with the MTIOC0B signal 000011: Mask the POE10# signal with the MTIOC0C signal 000100: Mask the POE10# signal with the MTIOC0D signal 000101: Mask the POE10# signal with the MTIOC1A signal 000110: Mask the POE10# signal with the MTIOC1B signal 000111: Mask the POE10# signal with the MTIOC2A signal 001000: Mask the POE10# signal with the MTIOC2B signal 001001: Mask the POE10# signal with the MTIOC3A signal 001010: Mask the POE10# signal with the MTIOC3B signal 001011: Mask the POE10# signal with the MTIOC3C signal 001100: Mask the POE10# signal with the MTIOC3D signal 001101: Mask the POE10# signal with the MTIOC4A signal 001110: Mask the POE10# signal with the MTIOC4B signal 001111: Mask the POE10# signal with the MTIOC4C signal 010000: Mask the POE10# signal with the MTIOC4D signal 011101: Mask the POE10# signal with the GTIOC0A signal 011110: Mask the POE10# signal with the GTIOC0B signal 011111: Mask the POE10# signal with the GTIOC1A signal 100000: Mask the POE10# signal with the GTIOC1B signal 100001: Mask the POE10# signal with the GTIOC2A signal 100010: Mask the POE10# signal with the GTIOC2B signal Settings other than above are prohibited.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

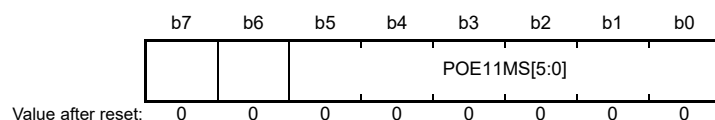
The IMCR3 register is an 8-bit readable/writable register used to mask the output disabling requests by the POE10# pin.

POE10MS[5:0] Bits (POE10 Mask Signal Select)

These bits are used to select a signal that masks the POE10# signal. When the selected pin drives high, the POE10# signal is masked.

20.2.37 Input Signal Mask Control Register 4 (IMCR4)

Address(es): POE.IMCR4 0009 E474h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	POE11MS[5:0]	POE11 Mask Signal Select	b5 b0 000000: Does not mask the POE11# signal 000001: Mask the POE11# signal with the MTIOC0A signal 000010: Mask the POE11# signal with the MTIOC0B signal 000011: Mask the POE11# signal with the MTIOC0C signal 000100: Mask the POE11# signal with the MTIOC0D signal 000101: Mask the POE11# signal with the MTIOC1A signal 000110: Mask the POE11# signal with the MTIOC1B signal 000111: Mask the POE11# signal with the MTIOC2A signal 001000: Mask the POE11# signal with the MTIOC2B signal 001001: Mask the POE11# signal with the MTIOC3A signal 001010: Mask the POE11# signal with the MTIOC3B signal 001011: Mask the POE11# signal with the MTIOC3C signal 001100: Mask the POE11# signal with the MTIOC3D signal 001101: Mask the POE11# signal with the MTIOC4A signal 001110: Mask the POE11# signal with the MTIOC4B signal 001111: Mask the POE11# signal with the MTIOC4C signal 010000: Mask the POE11# signal with the MTIOC4D signal 011101: Mask the POE11# signal with the GTIOC0A signal 011110: Mask the POE11# signal with the GTIOC0B signal 011111: Mask the POE11# signal with the GTIOC1A signal 100000: Mask the POE11# signal with the GTIOC1B signal 100001: Mask the POE11# signal with the GTIOC2A signal 100010: Mask the POE11# signal with the GTIOC2B signal Settings other than above are prohibited.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

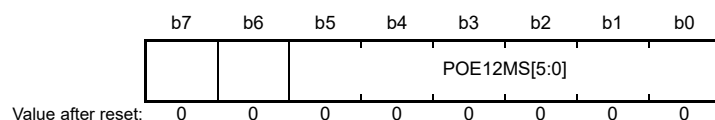
The IMCR4 register is an 8-bit readable/writable register used to mask the output disabling requests by the POE11# pin.

POE11MS[5:0] Bits (POE11 Mask Signal Select)

These bits are used to select a signal that masks the POE11# signal. When the selected pin drives high, the POE11# signal is masked.

20.2.38 Input Signal Mask Control Register 5 (IMCR5)

Address(es): POE.IMCR5 0009 E475h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	POE12MS[5:0]	POE12 Mask Signal Select	b5 b0 000000: Does not mask the POE12# signal 000001: Mask the POE12# signal with the MTIOC0A signal 000010: Mask the POE12# signal with the MTIOC0B signal 000011: Mask the POE12# signal with the MTIOC0C signal 000100: Mask the POE12# signal with the MTIOC0D signal 000101: Mask the POE12# signal with the MTIOC1A signal 000110: Mask the POE12# signal with the MTIOC1B signal 000111: Mask the POE12# signal with the MTIOC2A signal 001000: Mask the POE12# signal with the MTIOC2B signal 001001: Mask the POE12# signal with the MTIOC3A signal 001010: Mask the POE12# signal with the MTIOC3B signal 001011: Mask the POE12# signal with the MTIOC3C signal 001100: Mask the POE12# signal with the MTIOC3D signal 001101: Mask the POE12# signal with the MTIOC4A signal 001110: Mask the POE12# signal with the MTIOC4B signal 001111: Mask the POE12# signal with the MTIOC4C signal 010000: Mask the POE12# signal with the MTIOC4D signal 011101: Mask the POE12# signal with the GTIOC0A signal 011110: Mask the POE12# signal with the GTIOC0B signal 011111: Mask the POE12# signal with the GTIOC1A signal 100000: Mask the POE12# signal with the GTIOC1B signal 100001: Mask the POE12# signal with the GTIOC2A signal 100010: Mask the POE12# signal with the GTIOC2B signal Settings other than above are prohibited.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

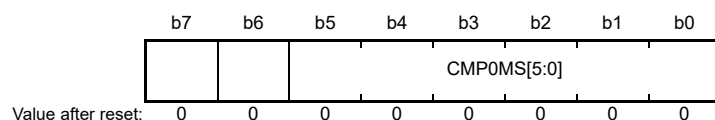
The IMCR5 register is an 8-bit readable/writable register used to mask the output disabling requests by the POE12# pin.

POE12MS[5:0] Bits (POE12 Mask Signal Select)

These bits are used to select a signal that masks the POE12# signal. When the selected pin drives high, the POE12# signal is masked.

20.2.39 Input Signal Mask Control Register 9 (IMCR9)

Address(es): POE.IMCR9 0009 E479h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	CMP0MS[5:0]	COMP0 Mask Signal Select	b5 b0 000000: Does not mask the COMP0 level detection signal 000001: Mask COMP0 level detection signal with MTIOC0A signal 000010: Mask COMP0 level detection signal with MTIOC0B signal 000011: Mask COMP0 level detection signal with MTIOC0C signal 000100: Mask COMP0 level detection signal with MTIOC0D signal 000101: Mask COMP0 level detection signal with MTIOC1A signal 000110: Mask COMP0 level detection signal with MTIOC1B signal 000111: Mask COMP0 level detection signal with MTIOC2A signal 001000: Mask COMP0 level detection signal with MTIOC2B signal 001001: Mask COMP0 level detection signal with MTIOC3A signal 001010: Mask COMP0 level detection signal with MTIOC3B signal 001011: Mask COMP0 level detection signal with MTIOC3C signal 001100: Mask COMP0 level detection signal with MTIOC3D signal 001101: Mask COMP0 level detection signal with MTIOC4A signal 001110: Mask COMP0 level detection signal with MTIOC4B signal 001111: Mask COMP0 level detection signal with MTIOC4C signal 010000: Mask COMP0 level detection signal with MTIOC4D signal 011101: Mask COMP0 level detection signal with GTIOC0A signal 011110: Mask COMP0 level detection signal with GTIOC0B signal 011111: Mask COMP0 level detection signal with GTIOC1A signal 100000: Mask COMP0 level detection signal with GTIOC1B signal 100001: Mask COMP0 level detection signal with GTIOC2A signal 100010: Mask COMP0 level detection signal with GTIOC2B signal Settings other than above are prohibited.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

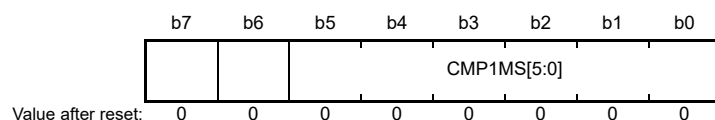
The IMCR9 register is an 8-bit readable/writable register used to mask the output disabling requests by the COMP0 level detection signal.

CMP0MS[5:0] Bits (COMP0 Mask Signal Select)

These bits are used to select a signal that masks the COMP0 level detection signal. When the selected pin drives high, the COMP0 level detection signal is masked.

20.2.40 Input Signal Mask Control Register 10 (IMCR10)

Address(es): POE.IMCR10 0009 E47Ah



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	CMP1MS[5:0]	COMP1 Mask Signal Select	b5 b0 000000: Does not mask the COMP1 level detection signal 000001: Mask COMP1 level detection signal with MTIOC0A signal 000010: Mask COMP1 level detection signal with MTIOC0B signal 000011: Mask COMP1 level detection signal with MTIOC0C signal 000100: Mask COMP1 level detection signal with MTIOC0D signal 000101: Mask COMP1 level detection signal with MTIOC1A signal 000110: Mask COMP1 level detection signal with MTIOC1B signal 000111: Mask COMP1 level detection signal with MTIOC2A signal 001000: Mask COMP1 level detection signal with MTIOC2B signal 001001: Mask COMP1 level detection signal with MTIOC3A signal 001010: Mask COMP1 level detection signal with MTIOC3B signal 001011: Mask COMP1 level detection signal with MTIOC3C signal 001100: Mask COMP1 level detection signal with MTIOC3D signal 001101: Mask COMP1 level detection signal with MTIOC4A signal 001110: Mask COMP1 level detection signal with MTIOC4B signal 001111: Mask COMP1 level detection signal with MTIOC4C signal 010000: Mask COMP1 level detection signal with MTIOC4D signal 011101: Mask COMP1 level detection signal with GTIOC0A signal 011110: Mask COMP1 level detection signal with GTIOC0B signal 011111: Mask COMP1 level detection signal with GTIOC1A signal 100000: Mask COMP1 level detection signal with GTIOC1B signal 100001: Mask COMP1 level detection signal with GTIOC2A signal 100010: Mask COMP1 level detection signal with GTIOC2B signal Settings other than above are prohibited.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

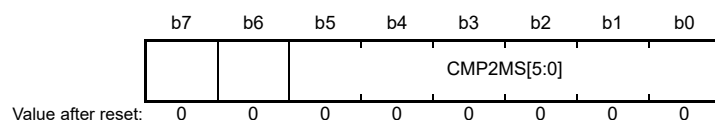
The IMCR10 register is an 8-bit readable/writable register used to mask the output disabling requests by the COMP1 level detection signal.

CMP1MS[5:0] Bits (COMP1 Mask Signal Select)

These bits are used to select a signal that masks the COMP1 level detection signal. When the selected pin drives high, the COMP1 level detection signal is masked.

20.2.41 Input Signal Mask Control Register 11 (IMCR11)

Address(es): POE.IMCR11 0009 E47Bh



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	CMP2MS[5:0]	COMP2 Mask Signal Select	b5 b0 000000: Does not mask the COMP2 level detection signal 000001: Mask COMP2 level detection signal with MTIOC0A signal 000010: Mask COMP2 level detection signal with MTIOC0B signal 000011: Mask COMP2 level detection signal with MTIOC0C signal 000100: Mask COMP2 level detection signal with MTIOC0D signal 000101: Mask COMP2 level detection signal with MTIOC1A signal 000110: Mask COMP2 level detection signal with MTIOC1B signal 000111: Mask COMP2 level detection signal with MTIOC2A signal 001000: Mask COMP2 level detection signal with MTIOC2B signal 001001: Mask COMP2 level detection signal with MTIOC3A signal 001010: Mask COMP2 level detection signal with MTIOC3B signal 001011: Mask COMP2 level detection signal with MTIOC3C signal 001100: Mask COMP2 level detection signal with MTIOC3D signal 001101: Mask COMP2 level detection signal with MTIOC4A signal 001110: Mask COMP2 level detection signal with MTIOC4B signal 001111: Mask COMP2 level detection signal with MTIOC4C signal 010000: Mask COMP2 level detection signal with MTIOC4D signal 011101: Mask COMP2 level detection signal with GTIOC0A signal 011110: Mask COMP2 level detection signal with GTIOC0B signal 011111: Mask COMP2 level detection signal with GTIOC1A signal 100000: Mask COMP2 level detection signal with GTIOC1B signal 100001: Mask COMP2 level detection signal with GTIOC2A signal 100010: Mask COMP2 level detection signal with GTIOC2B signal Settings other than above are prohibited.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Can be modified only once after a reset.

The IMCR11 register is an 8-bit readable/writable register used to mask the output disabling requests by the COMP2 level detection signal.

CMP2MS[5:0] Bits (COMP2 Mask Signal Select)

These bits are used to select a signal that masks the COMP2 level detection signal. When the selected pin drives high, the COMP2 level detection signal is masked.

20.3 Operation

The following shows the target pins and conditions for output disabling control.

(1) MTU3 pins (MTIOC3B, MTIOC3D)

When one of the following conditions is satisfied while the POECR2.MTU3BDZE bit is 1, the pins become high-impedance. Furthermore, when the POECR2.MTU3BDZE bit is 0 and the PMMCR1.MTU3BME and MTU3DME bits are 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- Operation for comparison of the output levels on the MTIOC3B and MTIOC3D signals
When the OCSR1.OSF1 flag becomes 1 while the OCSR1.OCE1 bit is 1.
- SPOER setting
When the SPOER.MTUCH34HIZ bit is set to 1.
- Conditions added by POECR4
When the ICSR1.POE0F flag becomes 1 while the POECR4.IC1ADDMT34ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR4.IC3ADDMT34ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR4.IC4ADDMT34ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR4.IC5ADDMT34ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR4.IC6ADDMT34ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(2) MTU4 pins (MTIOC4A, MTIOC4C)

When one of the following conditions is satisfied while the POECR2.MTU4ACZE bit is 1, the pins become high-impedance. Furthermore, when the POECR2.MTU4ACZE bit is 0 and the PMMCR1.MTU4AME and MTU4CME bits are 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- Operation for comparison of the output levels on the MTIOC4A and MTIOC4C signals
When the OCSR1.OSF1 flag becomes 1 while the OCSR1.OCE1 bit is 1.
- SPOER setting
When the SPOER.MTUCH34HIZ bit is set to 1.
- Conditions added by POECR4
When the ICSR1.POE0F flag becomes 1 while the POECR4.IC1ADDMT34ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR4.IC3ADDMT34ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR4.IC4ADDMT34ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR4.IC5ADDMT34ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR4.IC6ADDMT34ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(3) MTU4 pins (MTIOC4B, MTIOC4D)

When one of the following conditions is satisfied while the POECR2.MTU4BDZE bit is 1, the pins become high-impedance. Furthermore, when the POECR2.MTU4BDZE bit is 0 and the PMMCR1.MTU4BME and MTU4DME bits are 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- Operation for comparison of the output levels on the MTIOC4B and MTIOC4D signals
When the OCSR1.OSF1 flag becomes 1 while the OCSR1.OCE1 bit is 1.
- SPOER setting
When the SPOER.MTUCH34HIZ bit is set to 1.
- Conditions added by POECR4
When the ICSR1.POE0F flag becomes 1 while the POECR4.IC1ADDMT34ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR4.IC3ADDMT34ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR4.IC4ADDMT34ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR4.IC5ADDMT34ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR4.IC6ADDMT34ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR4.CMADDMT34ZE bit is 1 and the POECMPPEX0.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(4) MTU0 pin (MTIOC0A)

When one of the following conditions is satisfied while the POECR1.MTU0AZE bit is 1, the pins become high-impedance. Furthermore, when the POECR1.MTU0AZE bit is 0 and the PMMCR0.MTU0AME bit is 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- SPOER setting
When the SPOER.MTUCH0HIZ bit is set to 1.
- Conditions added by POECR5
When the ICSR1.POE0F flag becomes 1 while the POECR5.IC1ADDMT0ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR5.IC3ADDMT0ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR5.IC4ADDMT0ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR5.IC5ADDMT0ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR5.IC6ADDMT0ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(5) MTU0 pin (MTIOC0B)

When one of the following conditions is satisfied while the POECR1.MTU0BZE bit is 1, the pins become high-impedance. Furthermore, when the POECR1.MTU0BZE bit is 0 and the PMMCR0.MTU0BME bit is 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- SPOER setting
When the SPOER.MTUCH0HIZ bit is set to 1.
- Conditions added by POECR5
When the ICSR1.POE0F flag becomes 1 while the POECR5.IC1ADDMT0ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR5.IC3ADDMT0ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR5.IC4ADDMT0ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR5.IC5ADDMT0ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR5.IC6ADDMT0ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPEX2.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPEX2.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPEX2.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(6) MTU0 pin (MTIOC0C)

When one of the following conditions is satisfied while the POECR1.MTU0CZE bit is 1, the pins become high-impedance. Furthermore, when the POECR1.MTU0CZE bit is 0 and the PMMCR0.MTU0CME bit is 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- SPOER setting
When the SPOER.MTUCH0HIZ bit is set to 1.
- Conditions added by POECR5
When the ICSR1.POE0F flag becomes 1 while the POECR5.IC1ADDMT0ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR5.IC3ADDMT0ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR5.IC4ADDMT0ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR5.IC5ADDMT0ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR5.IC6ADDMT0ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(7) MTU0 pin (MTIOC0D)

When one of the following conditions is satisfied while the POECR1.MTU0DZE bit is 1, the pins become high-impedance. Furthermore, when the POECR1.MTU0DZE bit is 0 and the PMMCR0.MTU0DME bit is 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- SPOER setting
When the SPOER.MTUCH0HIZ bit is set to 1.
- Conditions added by POECR5
When the ICSR1.POE0F flag becomes 1 while the POECR5.IC1ADDMT0ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR5.IC3ADDMT0ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR5.IC4ADDMT0ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR5.IC5ADDMT0ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR5.IC6ADDMT0ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR5.CMADDMT0ZE bit is 1 and the POECMPPEX2.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(8) GPTW0 pins (GTIOC0A, GTIOC0B)

When one of the following conditions is satisfied while the POECR3.GPT0ABZE bit is 1, the pins become high-impedance. Furthermore, when the POECR3.GPT0ABZE bit is 0 and the PMMCR2.GPT0AME and GPT0BME bits are 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- Operation for comparison of the output levels on the GTIOC0A and GTIOC0B signals
When the OCSR3.OSF3 flag becomes 1 while the OCSR3.OCE3 bit is 1.
- SPOER setting
When the SPOER.GPT01HIZ bit is set to 1.
When the SPOER.GPT02HIZ bit is set to 1.
- Conditions added by POECR6
When the ICSR1.POE0F flag becomes 1 while the POECR6.IC1ADDGPT01ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR6.IC3ADDGPT01ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR6.IC4ADDGPT01ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR6.IC5ADDGPT01ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR6.IC6ADDGPT01ZE bit and the ICSR7.POE12E bit are 1.
- Conditions added by POECR9
When the ICSR1.POE0F flag becomes 1 while the POECR9.IC1ADDGPT02ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR9.IC3ADDGPT02ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR9.IC4ADDGPT02ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR9.IC5ADDGPT02ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR9.IC6ADDGPT02ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR6.CMADDGPT01ZE bit is 1 and the POECMPEX4.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR6.CMADDGPT01ZE bit is 1 and the POECMPEX4.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR6.CMADDGPT01ZE bit is 1 and the POECMPEX4.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(9) GPTW1 pins (GTIOC1A, GTIOC1B)

When one of the following conditions is satisfied while the POECR3.GPT1ABZE bit is 1, the pins become high-impedance. Furthermore, when the POECR3.GPT1ABZE bit is 0 and the PMMCR2.GPT1AME and GPT1BME bits are 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- Operation for comparison of the output levels on the GTIOC1A and GTIOC1B signals
When the OCSR3.OSF3 flag becomes 1 while the OCSR3.OCE3 bit is 1.
- SPOER setting
When the SPOER.GPT01HIZ bit is set to 1.
When the SPOER.GPT02HIZ bit is set to 1.
- Conditions added by POECR6
When the ICSR1.POE0F flag becomes 1 while the POECR6.IC1ADDGPT01ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR6.IC3ADDGPT01ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR6.IC4ADDGPT01ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR6.IC5ADDGPT01ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR6.IC6ADDGPT01ZE bit and the ICSR7.POE12E bit are 1.
- Conditions added by POECR9
When the ICSR1.POE0F flag becomes 1 while the POECR9.IC1ADDGPT02ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR9.IC3ADDGPT02ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR9.IC4ADDGPT02ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR9.IC5ADDGPT02ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR9.IC6ADDGPT02ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR6.CMADDGPT01ZE bit is 1 and the POECMPEX4.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR6.CMADDGPT01ZE bit is 1 and the POECMPEX4.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR6.CMADDGPT01ZE bit is 1 and the POECMPEX4.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

(10) GPTW2 pins (GTIOC2A, GTIOC2B)

When one of the following conditions is satisfied while the POECR3.GPT2ABZE bit is 1, the pins become high-impedance. Furthermore, when the POECR3.GPT2ABZE bit is 0 and the PMMCR2.GPT2AME and GPT2BME bits are 1 and when one of the following conditions is satisfied, the pins are switched to general I/O port pins.

- Operation for comparison of the output levels on the GTIOC2A and GTIOC2B signals
When the OCSR3.OSF3 flag becomes 1 while the OCSR3.OCE3 bit is 1.
- SPOER setting
When the SPOER.GPT23HIZ bit is set to 1.
When the SPOER.GPT02HIZ bit is set to 1.
- Conditions added by POECR6B
When the ICSR1.POE0F flag becomes 1 while the POECR6B.IC1ADDGPT23ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR6B.IC3ADDGPT23ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR6B.IC4ADDGPT23ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR6B.IC5ADDGPT23ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR6B.IC6ADDGPT23ZE bit and the ICSR7.POE12E bit are 1.
- Conditions added by POECR9
When the ICSR1.POE0F flag becomes 1 while the POECR9.IC1ADDGPT02ZE bit is 1.
When the ICSR3.POE8F flag becomes 1 while the POECR9.IC3ADDGPT02ZE bit and the ICSR3.POE8E bit are 1.
When the ICSR4.POE10F flag becomes 1 while the POECR9.IC4ADDGPT02ZE bit and the ICSR4.POE10E bit are 1.
When the ICSR5.POE11F flag becomes 1 while the POECR9.IC5ADDGPT02ZE bit and the ICSR5.POE11E bit are 1.
When the ICSR7.POE12F flag becomes 1 while the POECR9.IC6ADDGPT02ZE bit and the ICSR7.POE12E bit are 1.
- Comparator output detection
When the POECMPFR.C0FLAG flag becomes 1 while the POECR6B.CMADDGPT23ZE bit is 1 and the POECMPEX5.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ0 bit or the POECMPSEL.POEREQ0 bit is 1.
When the POECMPFR.C1FLAG flag becomes 1 while the POECR6B.CMADDGPT23ZE bit is 1 and the POECMPEX5.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ1 bit or the POECMPSEL.POEREQ1 bit is 1.
When the POECMPFR.C2FLAG flag becomes 1 while the POECR6B.CMADDGPT23ZE bit is 1 and the POECMPEX5.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1, or while the POECR9.CMADDGPT02ZE bit is 1 and the POECMPEX6.POEREQ2 bit or the POECMPSEL.POEREQ2 bit is 1.
- Detection of oscillation stop
When the ICSR6.OSTSTF flag becomes 1 while the ICSR6.OSTSTE bit is 1.

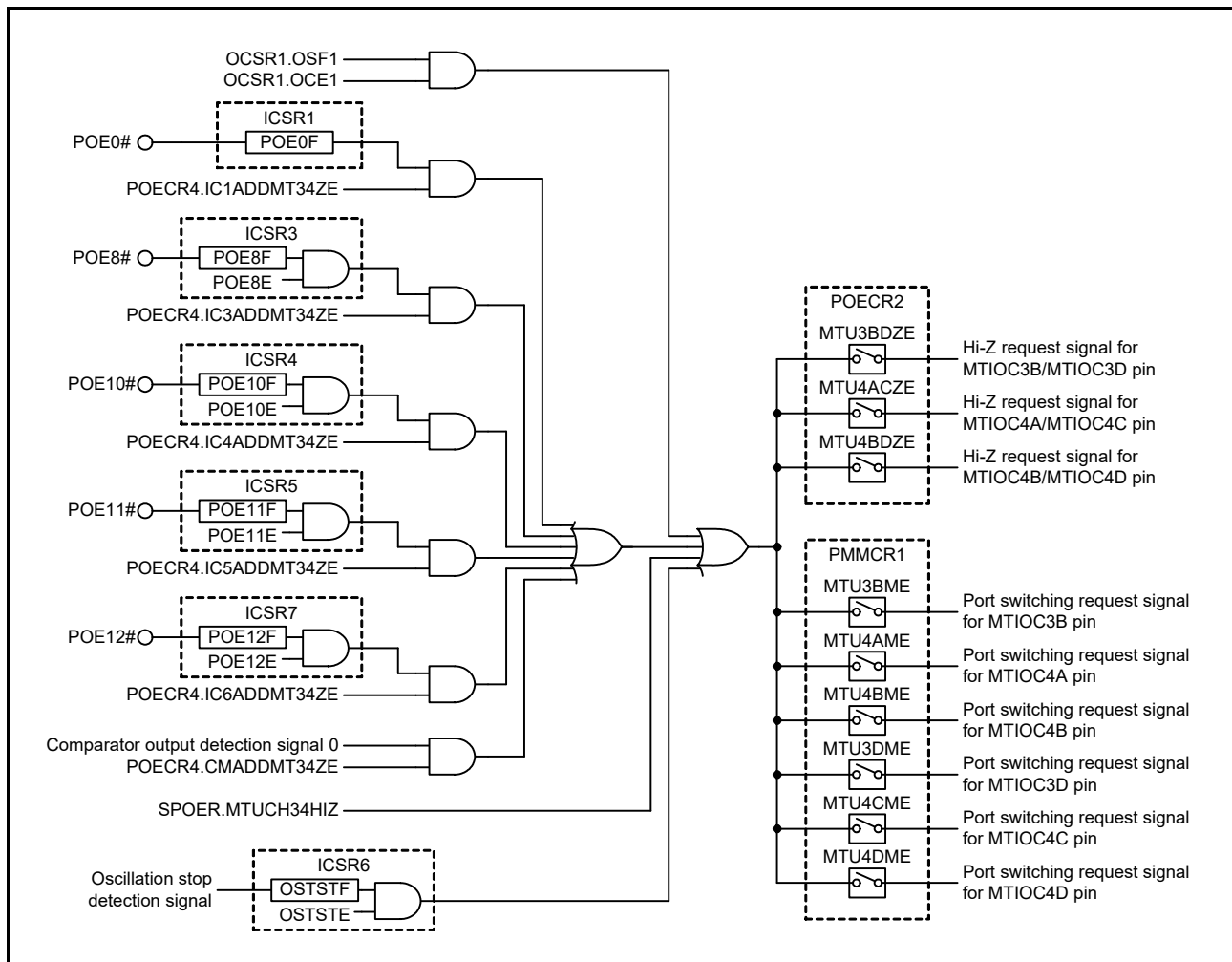


Figure 20.3 Target Pins and Conditions for High-Impedance Control (1)

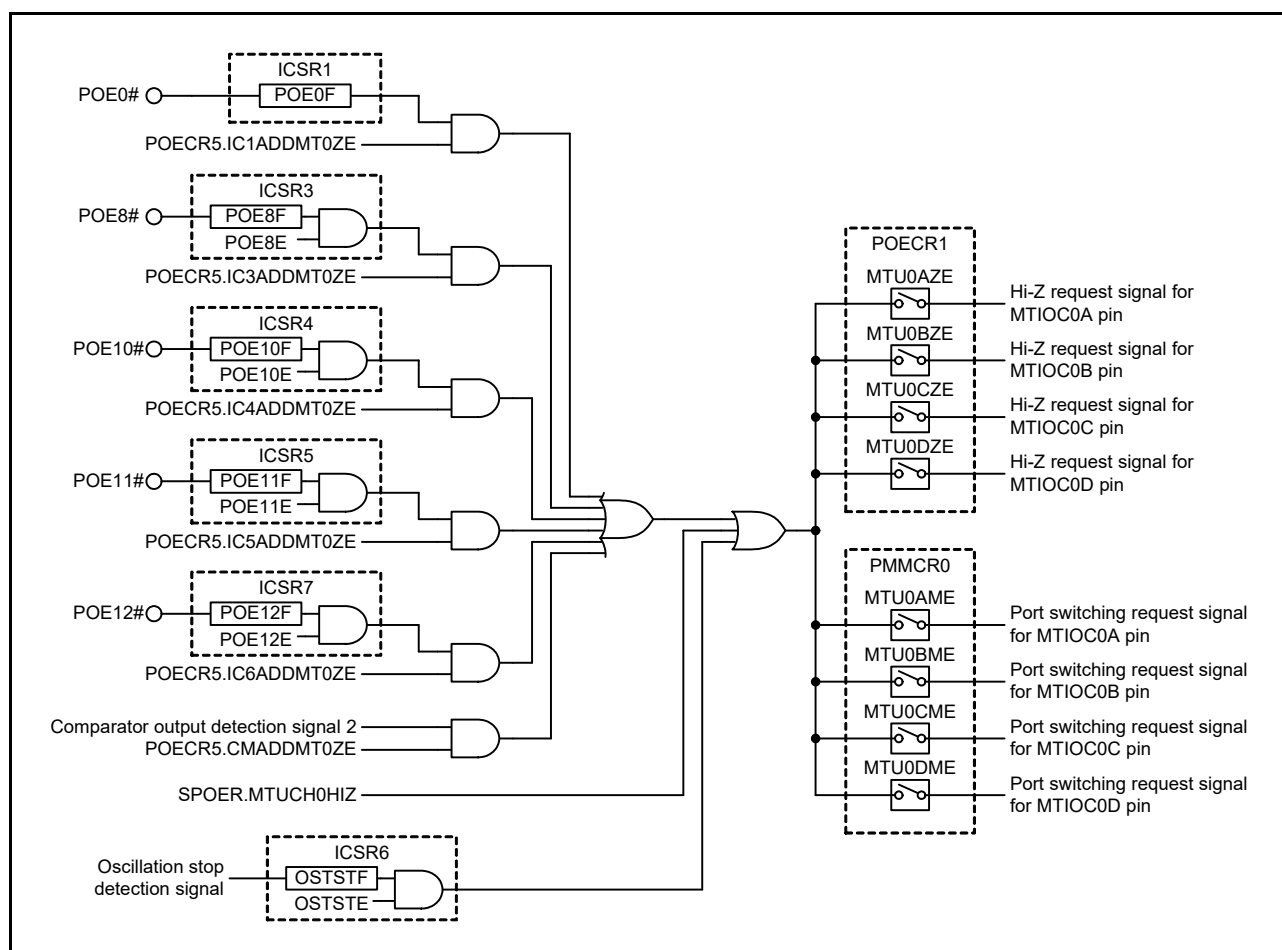


Figure 20.4 Target Pins and Conditions for High-Impedance Control (2)

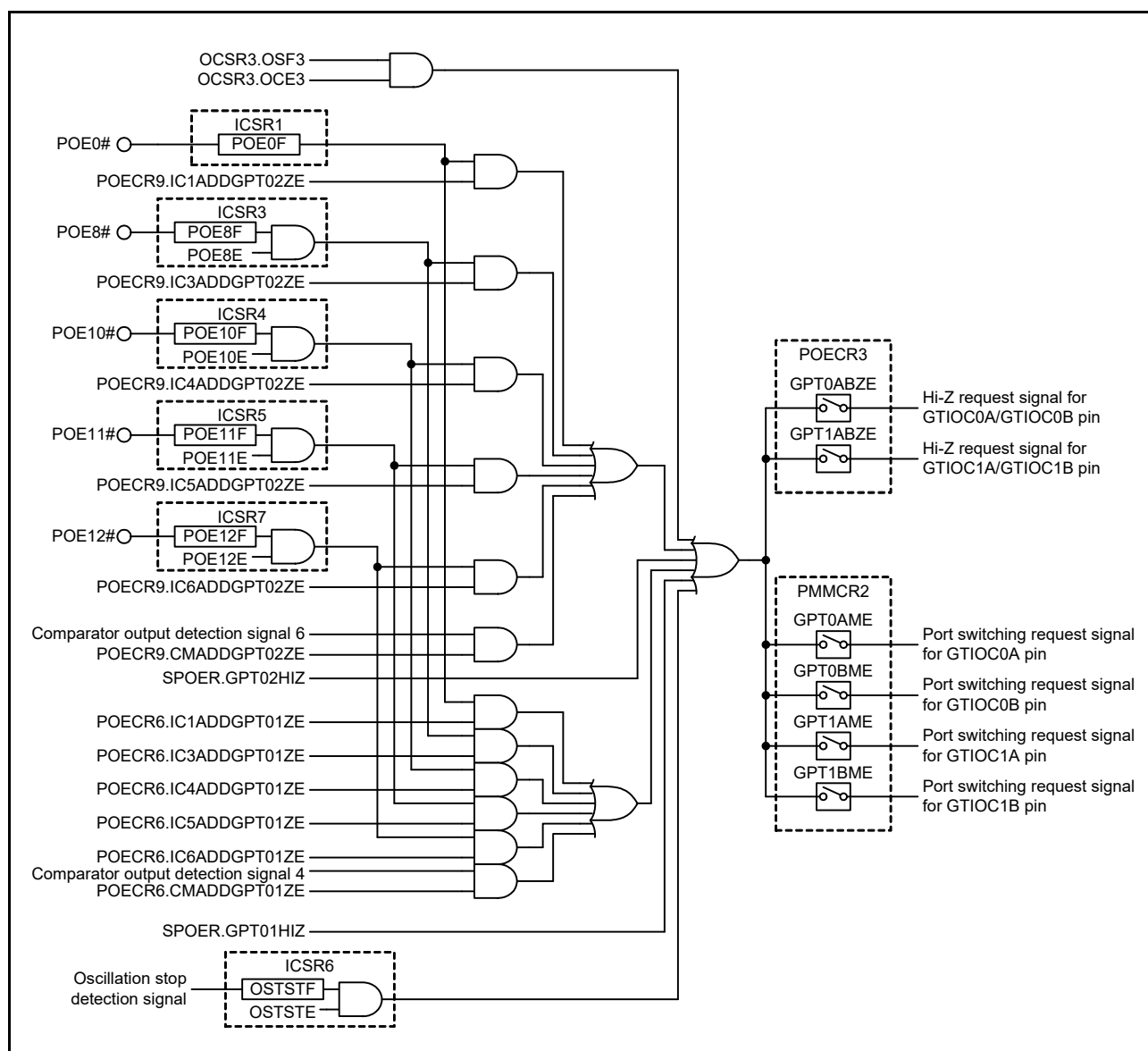


Figure 20.5 Target Pins and Conditions for High-Impedance Control (3)

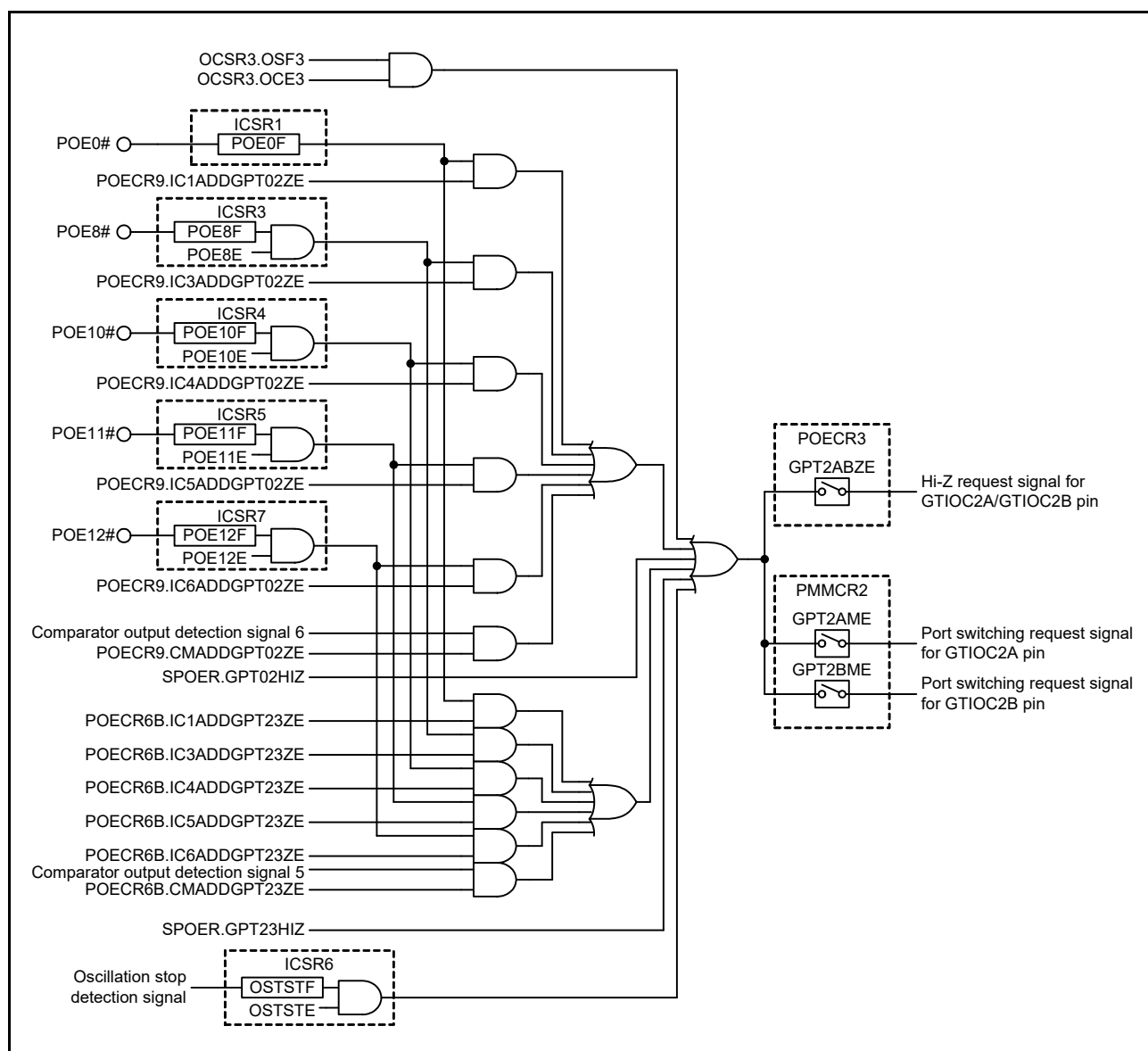


Figure 20.6 Target Pins and Conditions for High-Impedance Control (4)

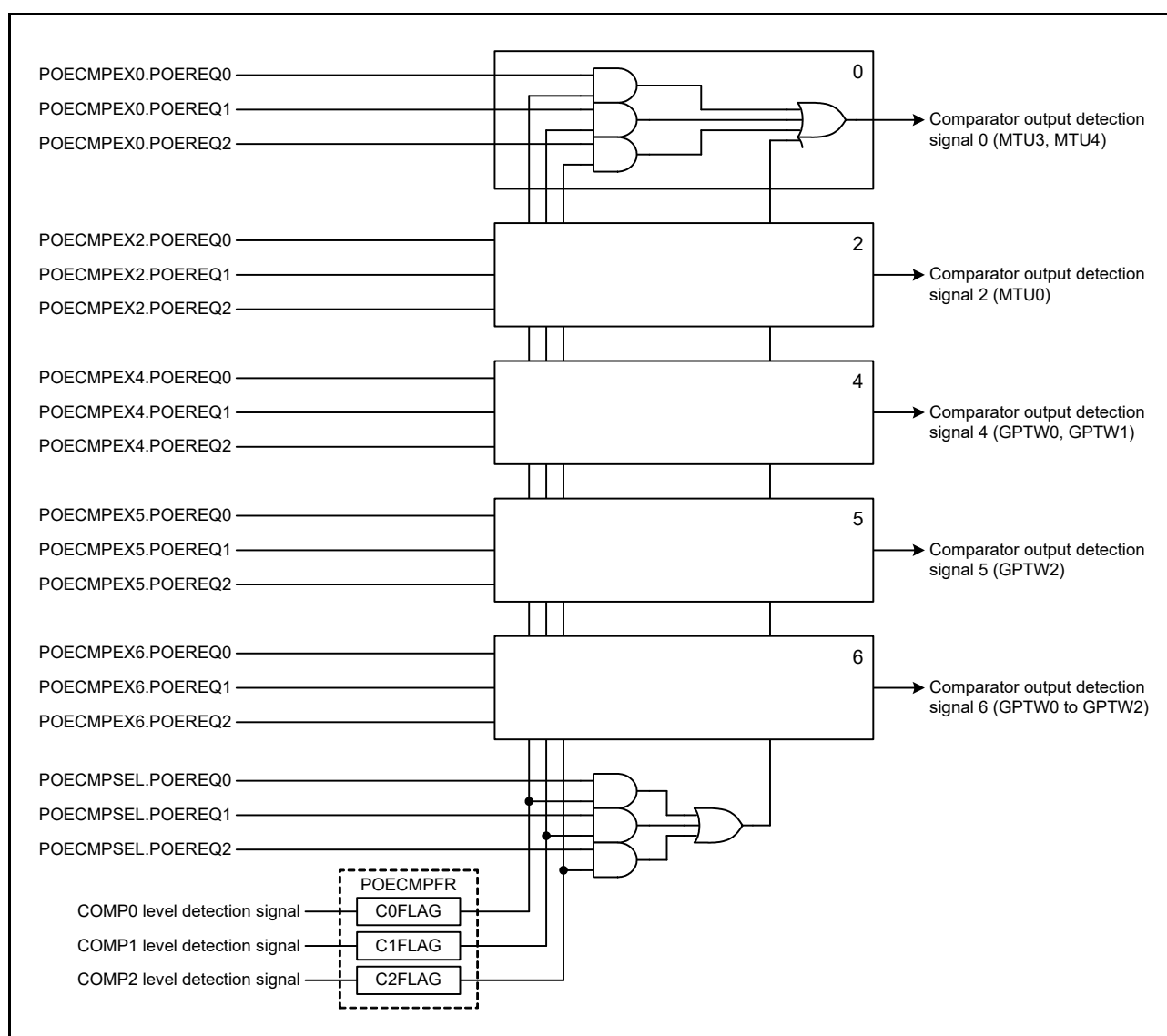


Figure 20.7 Comparator Output Detection Signal Generator Block Diagram

20.3.1 MTU/GPTW Pin Selection

In this MCU, the pin functions for MTU and GPTW are respectively multiplexed with multiple sets of port pins. The target pins for output disabling control can be selected by the pin select register in POE (M0SELR1, M0SELR2, M3SELR, M4SELR1, M4SELR2, G0SELR, G1SELR, or G2SELR register). Table 20.5 shows the correspondence between MTU pins and select registers. Table 20.6 shows the correspondence between GPTW pins and select registers. Note that settings for pins to be used as MTU or GPTW must be separately made in the registers of the multi-function pin controller (MPC). Take care so that there are no differences between the pins selected in the POE registers and the pins selected in the MPC registers.

Table 20.5 Correspondence between MTU Pins

MTU Pin Functions	Corresponding Ports	Select Registers
MTIOC0A	P30	M0SELR1
	P36	
	PB1	
	PD3	
MTIOC0B	P25	
	P92	
	PB0	
	PD4	
MTIOC0C	P33	M0SELR2
	P93	
	P97	
	PD5	
MTIOC0D	P32	
	P96	
	PD6	
MTIOC3B	P26	M3SELR
	P71	
	P94	
MTIOC3D	P74	
	P97	

MTU Pin Functions	Corresponding Ports	Select Registers
MTIOC4A	P31	M4SELR1
	P36	
	P72	
	P95	
	PD3	
MTIOC4C	P11	
	P75	
	PB0	
	PD5	
MTIOC4B	P25	M4SELR2
	P73	
	P96	
	PD4	
MTIOC4D	P76	
	PB1	
	PD6	

Table 20.6 Correspondence between GPTW Pins

GPTW Pin Functions	Corresponding Ports	Select Registers
GTIOC0A	P26	G0SELR
	P36	
	P71	
	P94	
GTIOC0B	P31	
	P74	
	P97	
GTIOC1A	P31	G1SELR
	P72	
	P92	
	P95	
	PB5	
	PD3	
GTIOC1B	P75	
	P93	
	PB0	
	PD4	

GPTW Pin Functions	Corresponding Ports	Select Registers
GTIOC2A	P04	G2SELR
	P23	
	P73	
	P96	
	PB5	
	PD5	
GTIOC2B	P03	
	P24	
	P76	
	PB1	
	PB6	
	PD6	

20.3.2 Input-Level Detection Operation

If the input conditions set by ICSR1, ICSR3 to ICSR5, and ICSR7 occur on the POE0#, POE8#, POE10#, POE11#, and POE12# pins, the outputs of the MTU complementary PWM output pins (MTU3 and MTU4), MTU0 pins, and GPTW pins are disabled.

(1) Edge Detection

When the input levels to the POE0#, POE8#, POE10#, POE11#, and POE12# pins change from high to low (when the INV bit is 0) or low to high (when the INV bit is 1), outputs of the MTU complementary PWM output pins, MTU0 pins, and GPTW pins are disabled.

The edge is detected after the level is sampled with PCLK. Input a low or high level for at least 1.5 PCLK clocks to the POE0#, POE8#, POE10#, POE11#, and POE12# pins.

Figure 20.8 shows a sample timing after the level changes in input to the POE0#, POE8#, POE10#, POE11#, and POE12# pins until the respective pins become high-impedance.

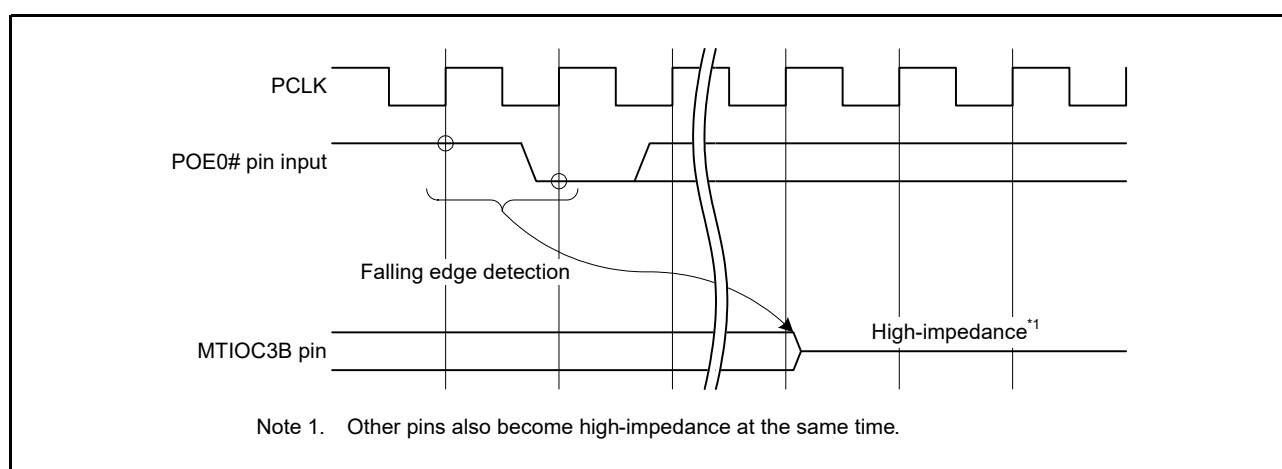


Figure 20.8 Operation when A Falling Edge Detection and High-Impedance are Selected

(2) Level Detection

Figure 20.9 shows an example of changing pins to high-impedance state while a low-level detection is selected (the INV bit is 0). When low is consecutively detected for the specified number of times with the sampling clock set in the ICSR1, ICSR3 to ICSR5, and ICSR7 registers, it is considered that low has been input, and the outputs of the MTU complementary PWM output pins, MTU0 pins, and GPTW pins are disabled. At this time, if a high is detected even once, it is not considered that low has been input.

The timing when the outputs of the MTU complementary PWM output pins, MTU0 pins, and GPTW pins are disabled after the sampling clock is input is the same in both edge detection and in level detection.

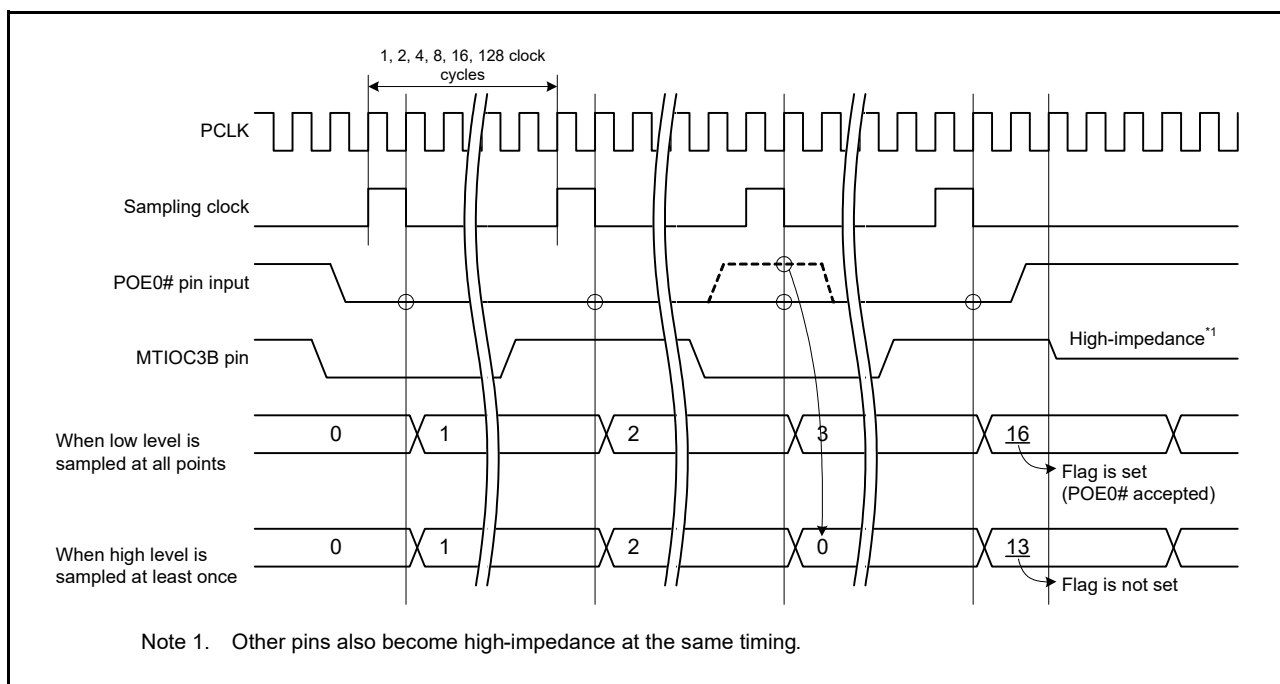


Figure 20.9 Operation when A Low-Level Detection and High-Impedance are Selected (16 samplings)

20.3.3 Output-Level Compare Operation

Figure 20.10 shows an example of the output-level compare operation for the combination of MTIOC3B and MTIOC3D. When the MTIOC3B and MTIOC3D signals simultaneously go to the active level for at least one cycle of PCLK, the outputs are disabled.

The operation is the same for the other signal combinations and switching to general I/O ports.

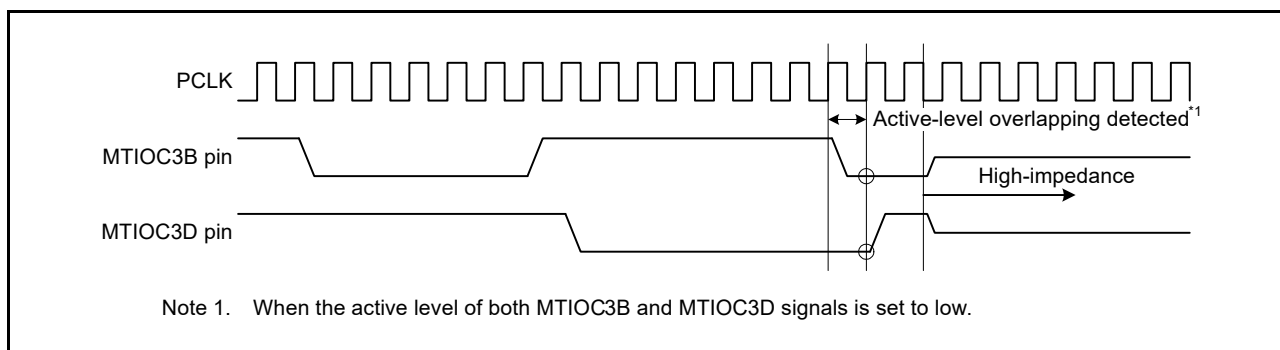


Figure 20.10 Output-Level Compare Operation

20.3.4 Output Disabling Control Using Registers

The output disabling request of the MTU pins (MTU0, MTU3, and MTU4) and the GPTW pins (GPTW0 to GPTW2) can be directly controlled by using the SPOER register.

For instance, setting the SPOER.MTUCH34HIZ bit to 1 switches the MTU3 and MTU4 pins specified by the POECR2 register to the high-impedance state.

The output disabling request of other pins can also be controlled by setting the appropriate bits in the SPOER register.

20.3.5 Output Disabling Control through Detection of Oscillation Stop

When the oscillation stop detection function for the clock generation circuit detects the oscillation stop while the ICSR6.OSTSTE bit is 1, the MTU complementary PWM output pins, MTU0 pins, and GPTW pins specified by the registers POECR1 to POECR3 can be switched to high-impedance, and the MTU complementary PWM output pins, MTU0 pins, and GPTW pins specified by the registers PMMCR0 to PMMCR2 can be switched to a general I/O port.

20.3.6 Output Disabling Control through Detection of the Comparator Output

The outputs of the MTU complementary PWM output pins, MTU0 pins, and GPTW pins can be disabled in response to detection of the output of the comparator.

For instance, when the POECMPFR.CjFLAG flag ($j = 0$ to 2) is added to the output disabling control conditions for the MTU3 and MTU4 pins by setting the POECR4.CMADDMT34ZE bit to 1, the MTU3 and MTU4 pins specified by the POECR2 register become high-impedance and the MTU3 and MTU4 pins specified by the PMMCR1 register are switched to general I/O port pins on comparator output detection.

The output disabling control of other pins can be controlled by the POECR1 to POECR6B registers and the PMMCR0 to PMMCR2 registers.

When using the CjFLAG flag ($j = 0$ to 2) as a source for the output disabling control conditions, set the POECMPSEL or POECMPExm register ($m = 0, 2, 4$ to 6). When the POECMPSEL.POEREQj bit is set to 1, the CjFLAG flag is used as a source for all target pins. When the POECMPExm.POEREQj bit is set to 1, the CjFLAG flag is used as a source for the corresponding pins.

20.3.7 Additional Functions for Output Disabling Control

Output disabling control conditions for the MTU complementary PWM output pins, MTU0 pins, and GPTW pins can be added by setting the POECR4 to POECR6, POECR6B, and POECR9 registers.

For instance, the settings listed below can be added as output disabling control conditions for the MTU3 and MTU4 pins.

- Setting the POECR4.CMADDMT34ZE bit to 1 adds comparator output detection
- Setting the POECR4.IC1ADDMT34ZE bit to 1 and adds the input-level detection by the POE0# pin
- Setting the POECR4.IC3ADDMT34ZE bit to 1 and adds the input-level detection by the POE8# pin
- Setting the POECR4.IC4ADDMT34ZE bit to 1 and adds the input-level detection by the POE10# pin
- Setting the POECR4.IC5ADDMT34ZE bit to 1 and adds the input-level detection by the POE11# pin
- Setting the POECR4.IC6ADDMT34ZE bit to 1 and adds the input-level detection by the POE12# pin

The output disabling control of other pins can also be controlled by setting the appropriate bits in the POECR4 to POECR6, POECR6B, and POECR9 registers.

20.3.8 Masking of Output Disabling Control Signals

Output disabling requests from the POE0#, POE8#, POE10#, POE11#, and POE12# signals and the comparator level detection signals can be masked by the IMCRn registers ($n = 0, 2$ to 5, 9 to 11).

When the signal selected by the IMCRn.POE_mMS[5:0] bits ($n = 0, 2$ to 5; $m = 0, 8, 10, 11, 12$) is high, the input signal from the POE_m# pin is masked. When the signal selected by the IMCRn.CMP_jMS[5:0] bits ($n = 9$ to 11; $j = 0$ to 2) is high, the comparator level detection signal is masked.

Figure 20.11 shows sample timing for input signal masking when falling-edge detection is selected.

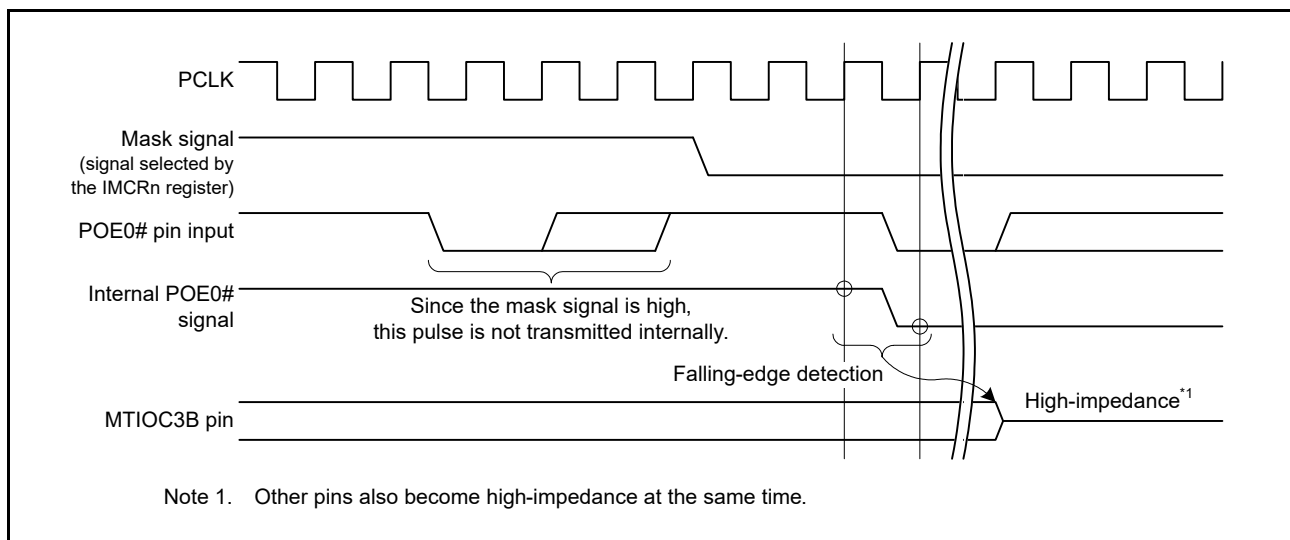


Figure 20.11 POE_m# Signal Masking Operation (When Falling-Edge Detection is Selected)

20.3.9 Control in Response to an Output Disabling Request

When a request to disable outputs is generated, those pins for which the corresponding bits of the POECR1 to POECR3 registers is set to 1 become high-impedance and those pins for which the corresponding bits of the PMMCR0 to PMMCR2 registers are set to 1 are switched to general I/O port pins.

When both bits are set to 1 for the same pin, the settings of the POECR1 to POECR3 registers take priority and the pins become high-impedance.

After a pin is switched to a general I/O port pin, the settings of the corresponding bits in the PDR and PODR registers determine the state of the pin.

20.3.10 Recover from Output Disabled State

The outputs which have been disabled due to input-level detection can be recovered from the state either by returning them to their initial state with a reset, or by clearing all of the ICSR1.POE0F, ICSR3.POE8F, ICSR4.POE10F, ICSR5.POE11F, and ICSR7.POE12F flags. However, note that when low-level detection is selected with the ICSR1.POE0M[3:0], ICSR3.POE8M[3:0], ICSR4.POE10M[3:0], ICSR5.POE11M[3:0], and ICSR7.POE12M[3:0] bits, just writing 0 to a flag is ignored (the flag is not set to 0); flags can be cleared by writing 0 to it only after a high level is input to the POE0#, POE8#, POE10#, POE11#, and POE12# pins and is detected. Similarly, when high-level detection is selected, the flag can be cleared by writing 0 to it only after a low level is input to the pin and is detected.

Pins whose outputs are disabled by detecting a simultaneous conduction can be released by resetting to return to the initial state or by writing 0 to the OCSR1.OSF1 flag and OCSR3.OSF3 flag. However, note that just writing 0 to a flag is ignored (the flag is not set to 0); the flags can be cleared by writing 0 to it only after setting the signals to the inactive level. In the MTU, the inactive level (initial output level) can be output by stopping the count operation. In the GPTW, the inactive level can be output in accordance with the procedure described in section 21.8.2, Pin Initialization Due to

Error during Operation.

The outputs which have been disabled due to comparator output detection can be recovered from the state either by returning them to their initial state with a reset or by setting the POECMPFR.CjFLAG flag (j = 0 to 2) to 0.

When setting the POECMPFR.CjFLAG flag to 0, be sure to confirm that the analog input signal that triggered comparator output detection has returned to a normal value by performing A/D conversion and so on.

Note that the above POECMPFR.CjFLAG flag is not set to 1 again in the following cases:

- This flag is cleared without confirmation that the analog input signal has returned to a normal value, and
 1. the analog input signal remains above the reference voltage when the comparator is set to non-inverted output, or
 2. the analog input signal remains below the reference voltage when the comparator is set to inverted output.

The outputs which have been disabled due to oscillation stop detection can be recovered from the state either by returning them to their initial state with a reset or by setting the SYSTEM.OSTDSR.OSTDF flag to 0 to set the ICSR6.OSTSTF flag to 0.

20.4 POE Setting Procedure

Figure 20.12 and Figure 20.13 show the procedures for setting the POE. Figure 20.12 shows an example of high-impedance control in response to comparison of the output levels on the MTU3 signals (MTIOC3B/MTIOC3D) when the MTIOC3B signal is assigned to the P71 pin and the MTIOC3D signal to the P74 pin. Figure 20.13 shows an example of high-impedance control in response to detection of the falling edge of the POE0# signal when the POE0# signal is assigned to the P71 pin, the MTIOC3B signal to the P71 pin, and the MTIOC3D signal to the P74 pin.

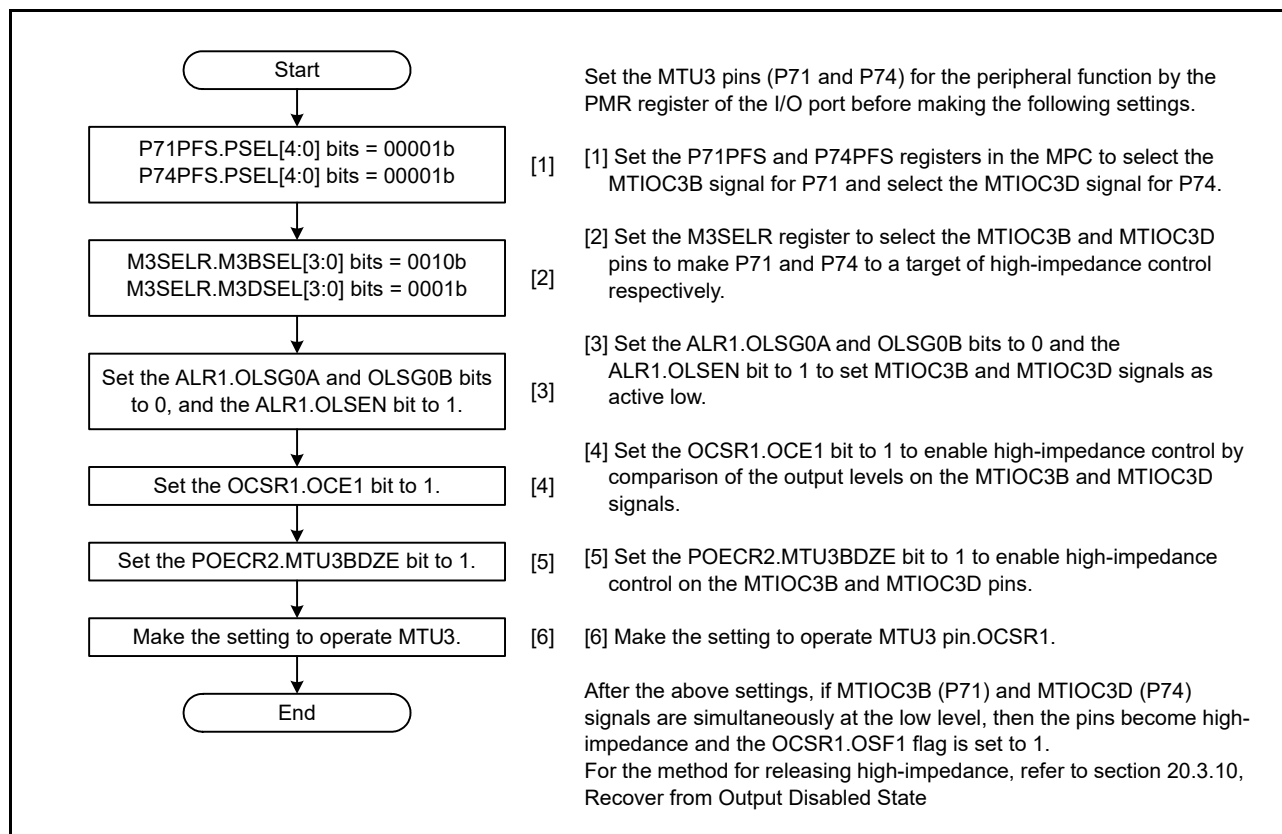


Figure 20.12 Example of Procedure for Setting the POE (Output-Level Compare Operation)

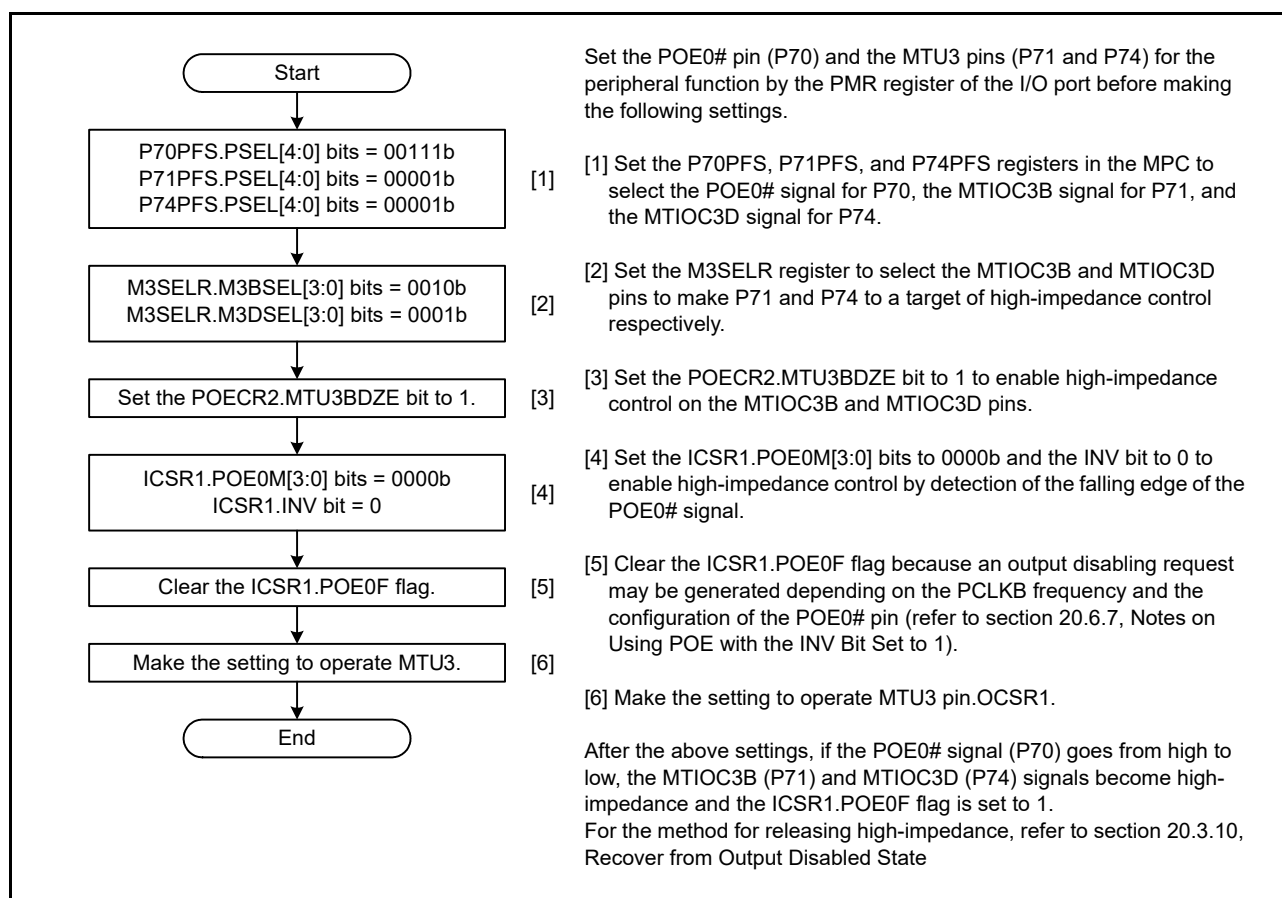


Figure 20.13 Example of Procedure for Setting the POE (Input-Level Detection Operation)

20.5 Interrupts

The POE issues a request to generate an interrupt when the specified condition is satisfied during input-level detection or output-level comparison. Table 20.7 shows the interrupt sources and their conditions.

Table 20.7 Interrupt Sources and Conditions

Name	Interrupt Source	Interrupt Flag	Condition
OEI1	Output enable interrupt 1	POE0F, OSF1	When the ICSR1.POE0F flag becomes 1 while the ICSR1.PIE1 bit is 1 or when the OCSR1.OSF1 flag becomes 1 while the OCSR1.OIE1 bit is 1
OEI3	Output enable interrupt 3	POE8F	When the ICSR3.POE8F flag becomes 1 while the ICSR3.PIE3 bit is 1
OEI4	Output enable interrupt 4	POE10F, POE11F	When the ICSR4.POE10F flag becomes 1 while the ICSR4.PIE4 bit is 1 or when the ICSR5.POE11F flag becomes 1 while the ICSR5.PIE5 bit is 1
OEI5	Output enable interrupt 5	POE12F, OSF3	When the ICSR7.POE12F flag becomes 1 while the ICSR7.PIE7 bit is 1 or when the OCSR3.OSF3 flag becomes 1 while the OCSR3.OIE3 bit is 1

20.6 Usage Notes

20.6.1 Transition to Low Power Consumption Mode

When using the POE, do not transition to software standby mode. Output disabling control for pins cannot be performed as the POE operation is disabled in software standby mode.

20.6.2 Output Disabling Control When the MTU and GPTW Pins are Not Selected

When the control conditions are satisfied while the output disabling control for the MTU and GPTW pins are enabled by the POECR1 to POECR3 registers, and the PMMCR0 to PMMCR2 registers, outputs are disabled for the pins which have become controlled by the pin select register settings even when the MTU and GPTW functions are not selected by the PxyPFS register in the MPC.

To avoid unintended output disabling, ensure that there are no differences between the settings for MTU and GPTW pin selection in the PxyPFS registers of the MPC and for MTU and GPTW pin selection in the pin select register of the POE.

20.6.3 When the POE is Not Used

The output disabling control for the pins by the POE becomes enabled for some pins after resetting. Even when the POE is not used, write 0 to a relevant bit in the POECR2 register. Also write 0000b to relevant bits in the following pin select registers: M0SELR1, M0SELR2, M3SELR, M4SELR1, M4SELR2, G0SELR, G1SELR, and G2SELR.

20.6.4 Setting the Active Level When the Inverting Output Is Set for the MTU and GPTW

In this MCU, either of normal output or inverting output can be selected for an output of the MTU and GPTW by the MPC.PxyPFS register.

When inverting output of the MTU is selected, the active level specified by the MTU.TOCR1j and MTU.TOCR2j registers (j = A, B) and the active level of the signal which is output to the pins become inverted. At this time, to use detection of simultaneous conduction, specify, by the ALR1 register, the active level with reference to signals which are output to the pins.

When the inverting output of the GPTW is selected, the active level of the signal which is output to the pins become inverted. At this time, to use detection of simultaneous conduction, specify, by the ALR3 register, the active level with reference to signals which are output to the pins.

20.6.5 Reading the Pins When in High-Impedance State

When the pins are in high-impedance state by the POE, the level of the relevant pins cannot be read. The value when read is indeterminable. To read the level of the pins, release from the high-impedance state.

When the port switching control is selected instead of high-impedance control, this is not applied.

20.6.6 Note for Using POE and POEG Together

When using the POE and POEG together, do not control the output disabling for the same GPTW output pin by both the POE and POEG.

20.6.7 Notes on Using POE with the INV Bit Set to 1

When using the POE with the ICSRn.INV bit (n = 1, 3 to 5, 7) set to 1 (rising edge or high-level detection), it is expected that low is input to the POEm# pin (m = 0, 8, 10, 11, 12) when the reset is released. When the POEm# signal is selected by the MPC.PxyPFS register in this state, the internal signal changes from high to low, and an output disabling request is generated. Set the ICSRn register first and then the PxyPFS register.

Depending on the settings of the ICSRn.POEmM[3:0] and POEmM2[3:0] bits, an output disabling request may be generated between the settings of the ICSRn register and the PxyPFS register. In this case, clear the corresponding ICSRn.POEmF flag according to section 20.3.10, Recover from Output Disabled State.

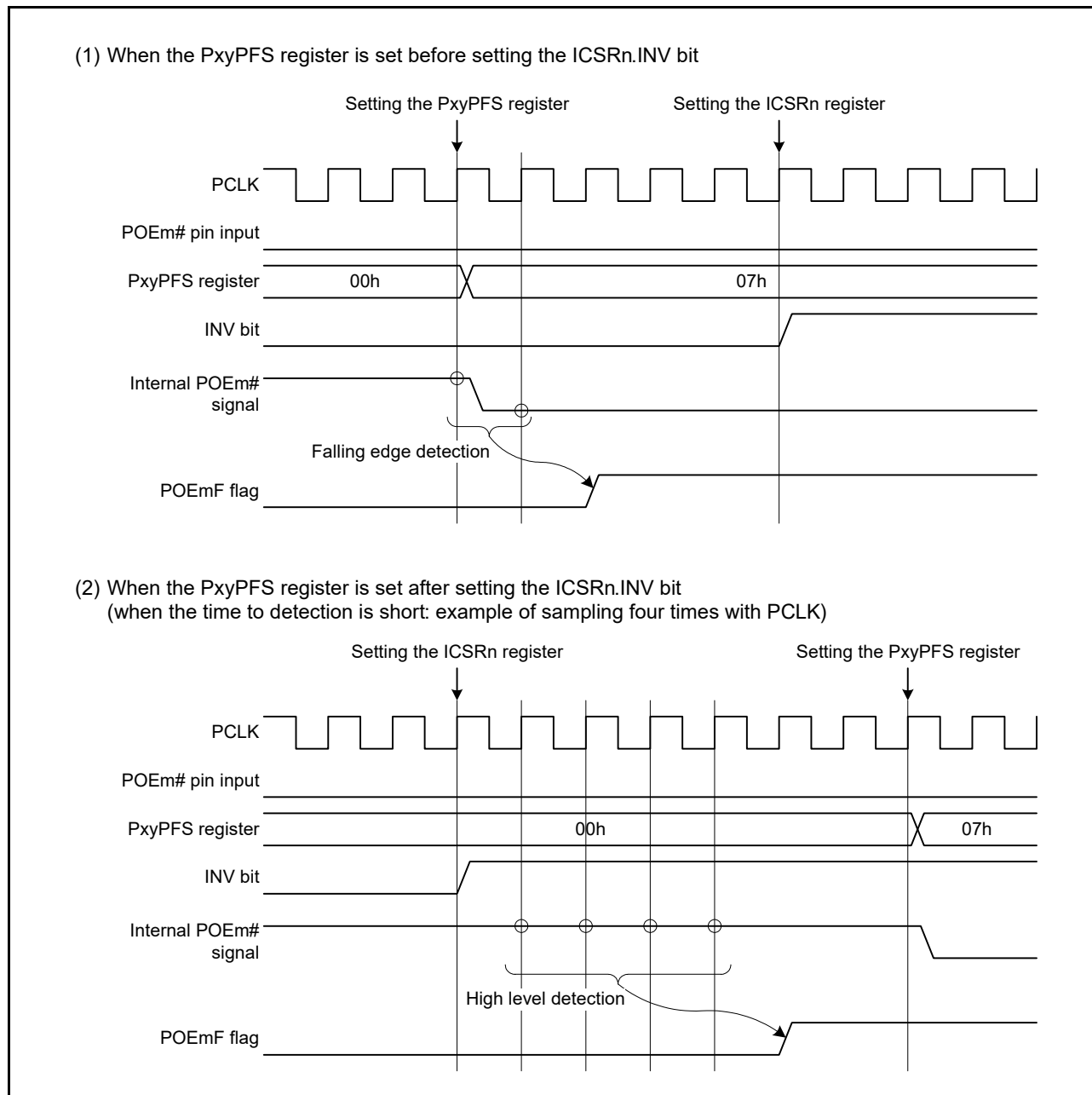


Figure 20.14 Timing to Set ICSRn.INV Bit and PxyPFS Register

21. General PWM Timer (GPTWd)

This MCU has a general PWM timer (GPTW) consisting of three 16-bit timer channels.

21.1 Overview

Table 21.1 lists the specifications for the GPTW, and Table 21.2 shows the functions of the GPTW. Figure 21.1 shows block diagrams of the GPTW.

Table 21.1 GPTW Specifications

Item	Description
Functions	• 16 bits × 3 channels • Up-counting or down-counting (sawtooth waves) or up/down-counting (triangle waves) for each counter. • Clock sources independently selectable for each channel • Two I/O pins per channel • Two output compare/input capture registers per channel • For the two output compare/input capture registers of each channel, four registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use. • In output compare operation, buffer switching can take place at crests or troughs, enabling the generation of laterally asymmetrical PWM waveforms. • Registers for setting up frame cycles in each channel (with capability for generating interrupts at overflow or underflow) • Generation of dead times in PWM operation • Simultaneous start/stop/clearing of desired channel counters • Operation of count start/count stop/counter clearing/up-counting/down-counting/input capture by detecting two input signal conditions • Operation of count start/count stop/counter clearing/up-counting/down-counting/input capture by one external trigger • MTU synchronous count start • Count start operation is available with MTU0/MTU4 count start operation • Duty 0%/100% Output operation • Function to control output negation by dead time error and requests for disabling of output from the POEG • A/D conversion start trigger generation function • Input capture input can select noise filter function • Synchronous counter clearing/counter setting/input capture among channels • Bus clock, GPTW count reference clock: PCLKB

Table 21.2 GPTW Functions

Item		GPTW0	GPTW1, GPTW2
Timer counter		16-bit	
Count clocks		PCLKB, PCLKB/2, PCLKB/4, PCLKB/8, PCLKB/16, PCLKB/32, PCLKB/64, PCLKB/256, PCLKB/1024, GTETRGA	
Output compare/input capture registers (GTCCR)		GTCCRA, GTCCRB	
Compare/buffer registers		GTCCRC, GTCCRD, GTCCRE, GTCCRF	
PWM period setting register		GTPR	
PWM period setting buffer registers		GTPBR, GTPDBR	
Input capture input/ compare match output/ PWM output pin		GTIOCnA, GTIOCnB	
External trigger input pin (via the POEG)		GTETRGA	
Counter clear sources		Compare match for the GTPR register, input capture, the state of input pins, external trigger input, GTCCR register compare match, and Other channel's counter clear sources	
Compare match output	Low output	✓	
	High output	✓	
	Toggle output	✓	
Input capture function		✓	
Automatic addition of dead time		✓	
PWM mode	Sawtooth-wave PWM mode	✓	
	Sawtooth-wave one-shot pulse mode	✓	
	Triangle-wave PWM mode 1/2/3	✓	
PWM synchronous output		✓	—
Buffer operation	Double buffer	✓	
	Simultaneous operation disable control for multiple channels	✓	
	Buffer transfer by counter clear / compare match	✓	
One-shot operation		✓	
DTC activation		All interrupt sources	
A/D conversion start trigger		Compare match of GTADTRA or GTADTRB register	
Interrupt sources		9 sources • GTCCRA register compare match/input capture (GTCIA_n) • GTCCRB register compare match/input capture (GTCIB_n) • GTCCRC register compare match (GTCIC_n) • GTCCRD register compare match (GTCID_n) • Dead time error (GDTEn) • GTCCRE register compare match (GTCIE_n) • GTCCRF register compare match (GTCIF_n) • GTCNT counter overflow (GTPR register compare match) (GTCIV_n) • GTCNT counter underflow (GTCIU_n)	
Interrupt skipping function		• Skipping of interrupts of GTCNT counter overflow (GTPR register compare match) (GTCIV_n) and GTCNT counter underflow (GTCIU_n) (interlocked with other interrupts and A/D conversion start requests)	
Noise filter function		✓	
Synchronous counter clearing/counter setting/input capture		✓	
MTU Synchronous count start		✓	

✓: Possible

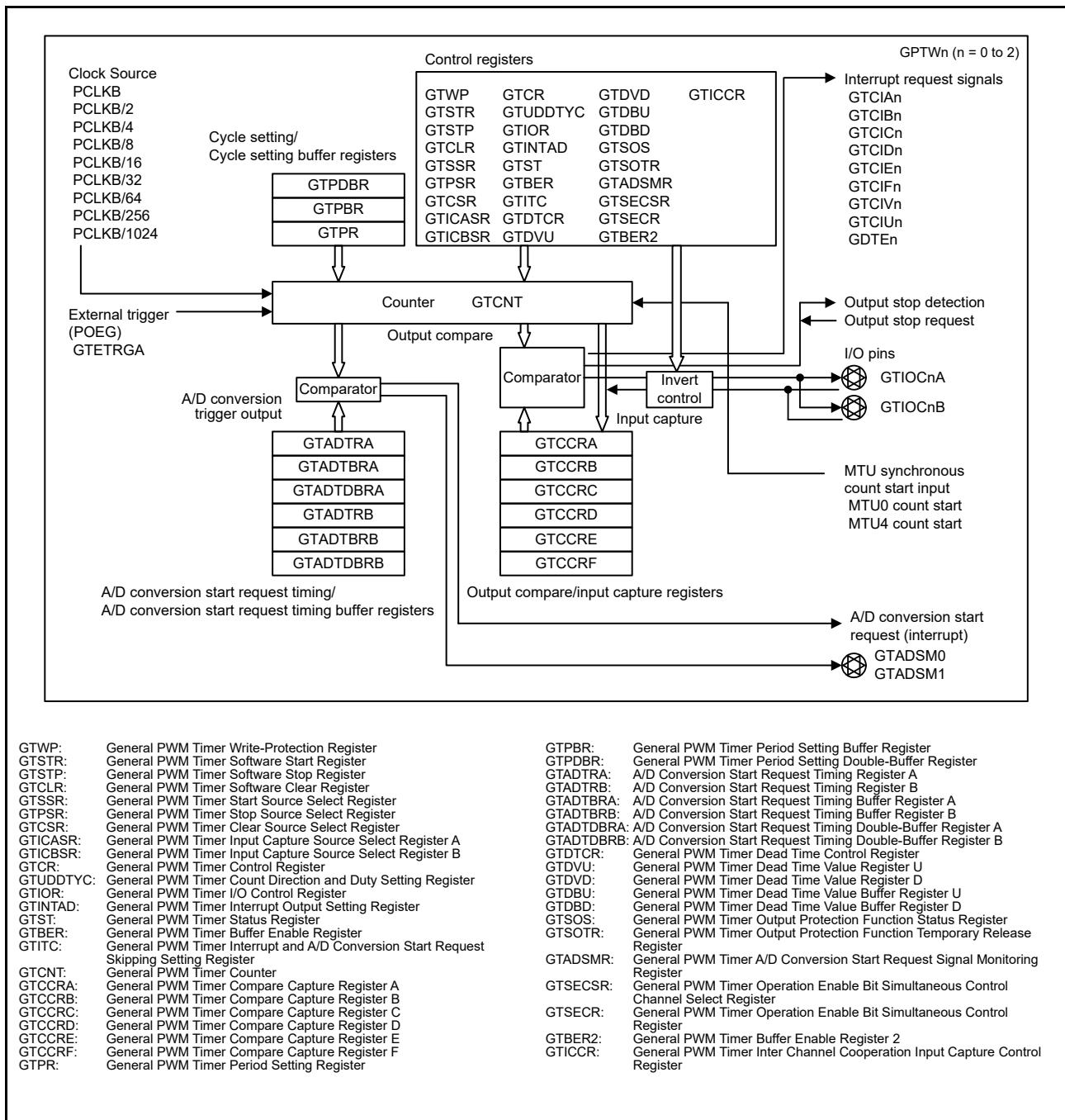


Figure 21.1 GPTW Block Diagram (Sawtooth-wave PWM mode, Sawtooth-wave one-shot pulse mode, and Triangle-wave PWM mode 1/2/3)

Table 21.3 lists the I/O pins used in the GPTW.

Table 21.3 GPTW I/O Pins (n = 0 to 2)

Channel	Pin Name	I/O	Function
GPTW	GTETRG	Input	External trigger input pin A (input via the POEG)
	GTADSM0	Output	A/D conversion start request monitor 0 output pin
	GTADSM1	Output	A/D conversion start request monitor 1 output pin
GPTWn	GTIOCN	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOCNB	I/O	GTCCRB register input capture input/output compare output/PWM output pin
	GTCPP0	Output	Toggle output synchronized with PWM period

21.2 Register Descriptions

21.2.1 General PWM Timer Write-Protection Register (GTWP)

Address(es): GPTW0.GTWP 0009 6000h, GPTW1.GTWP 0009 6100h, GPTW2.GTWP 0009 6200h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRKEY[7:0]							—	—	—	CMNWP	CLRWP	STPWP	STRWP	WP	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	WP	Register Write Disabled	0: Write to the register is enabled 1: Write to the register is disabled	R/W
b1	STRWP	GTSTR.CSTRT Bit Write Disabled	0: Write to the bit is enabled 1: Write to the bit is disabled	R/W
b2	STPWP	GTSTP.CSTOP Bit Write Disabled	0: Write to the bit is enabled 1: Write to the bit is disabled	R/W
b3	CLRWP	GTCLR.CCLR Bit Write Disabled	0: Write to the bit is enabled 1: Write to the bit is disabled	R/W
b4	CMNWP	Common Register Write Disabled	0: Write to the register is enabled 1: Write to the register is disabled	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	PRKEY[7:0]	GTWP Key Code	These bits are read as 0. To modify the WP, STRWP, STPWP, CLRWP, and CMNWP bits, write A5h.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTWP enables or disables writing to registers to prevent accidental modification.

Protection by the GTWP register only covers the CPU write operation.

Such protection does not cover a register updating generated in coordination with the CPU write.

For registers that are write enabled or disabled depending on the setting of the GTWP register, see section 21.7.1, Write-Protection for Registers.

WP Bit (Register Write Disabled)

Write enable/disable to the GPTW can be selected with this bit.

Registers which are targets of write enable/disable are as follows:

GTSSR, GTPSR, GTCSSR, GTICASSR, GTICBSR, GTCR, GTUDDTYC, GTIOR, GTINTAD, GTST, GTBER, GTITC, GTCNT, GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, GTCCRF, GTPR, GTPBR, GTPDBR, GTADTRA, GTADTBRA, GTADTDBRA, GTADTRB, GTADTBRB, GTADTDBRB, GTDTCR, GTDVU, GTDVD, GTDBU, GTDBD, GTSOS, GTSOTR, GTADSMR, GTBER2, GTICCR

STRWP Bit (GTSTR.CSTRT Bit Write Disabled)

The STRWP bits enable or disable starting the updating of counter values by writing to the CSTRTn bit corresponding to a channel number in the GTSTR register.

The bit position of each CSTRTn bit in the GTSTR registers is allocated to the channel with the corresponding number, and writing to the GTSTR register for any channel results in writing to the registers of all channels. The STRWP bit for

each channel does not control writing but only controls updating of the CSTRT bit for the corresponding channel in the case of simultaneous writing to all channels.

Thus, in the case of writing to the CSTRT bits of a channel for which the setting of the STRWP bit is 1 (disabling writing), the CSTRT bit for the given channel will not be updated, but the CSTRT bits corresponding to channel for which the setting of the STRWP bit is 0 (enabling writing) will be updated. For example, when the setting of the GPTW0.GTWP.STRWP bit is 0 (enabling writing), writing 1 to the GPTW1.GTSTR.CSTRT0 bit when its current setting is 0 will lead to updating of the value, and the GPTW0.GTCNT counter starts running. When the setting of the GPTW0.GTWP.STRWP bit is 1 (disabling writing), writing 1 to the GPTW1.GTSTR.CSTRT0 bit when its current setting is 0 will leave the bit with the value 0, and the GPTW0.GTCNT counter will not start running.

If you want to protect all bits in the GTSTR register from being updated, set the STRWP bits of all channels to 1.

STPWP Bit (GTSTP.CSTOP Bit Write Disabled)

The STPWP bits enable or disable starting the updating of counter values by writing to the CSTOPn bit corresponding to a channel number in the GTSTP register.

The bit position of each CSTOPn bit in the GTSTP registers is allocated to the channel with the corresponding number, and writing to the GTSTP register for any channel results in writing to the registers of all channels. The STPWP bit for each channel does not control writing but only controls updating of the CSTOP bit for the corresponding channel in the case of simultaneous writing to all channels.

Thus, in the case of writing to the CSTOP bits of a channel for which the setting of the STPWP bit is 1 (disabling writing), the CSTOP bit for the given channel will not be updated, but the CSTOP bits corresponding to channel for which the setting of the STPWP bit is 0 (enabling writing) will be updated. For example, when the setting of the GPTW0.GTWP.STPWP bit is 0 (enabling writing), writing 1 to the GPTW1.GTSTP.CSTOP0 bit when its current setting is 0 will lead to updating of the value, and the GPTW0.GTCNT counter is stopped. When the setting of the GPTW0.GTWP.STPWP bit is 1 (disabling writing), writing 1 to the GPTW1.GTSTP.CSTOP0 bit when its current setting is 0 will leave the bit with the value 0, and the GPTW0.GTCNT counter will not be stopped.

If you want to protect all bits in the GTSTP register from being updated, set the STPWP bits of all channels to 1.

CLRWP Bit (GTCLR.CCLR Bit Write Disabled)

The CLRWP bits enable or disable starting the updating of counter values by writing to the CCLRn bit corresponding to a channel number in the GTCLR register.

The bit position of each CCLRn bit in the GTCLR registers is allocated to the channel with the corresponding number, and writing to the GTCLR register for any channel results in writing to the registers of all channels. The CLRWP bit for each channel does not control writing but only controls updating of the CCLR bit for the corresponding channel in the case of simultaneous writing to all channels.

Thus, in the case of writing to the CCLR bits of a channel for which the setting of the CLRWP bit is 1 (disabling writing), the CCLR bit for the given channel will not be updated, but the CCLR bits corresponding to channel for which the setting of the CLRWP bit is 0 (enabling writing) will be updated. For example, when the setting of the GPTW0.GTWP.CLRWP bit is 0 (enabling writing), writing 1 to the GPTW1.GTCLR.CCLR0 bit when its current setting is 0 will lead to updating of the value, and the GPTW0.GTCNT counter is cleared.

If you want to protect all bits in the GTCLR register from being updated, set the CLRWP bits of all channels to 1.

CMNWP Bit (Common Register Write Disabled)

The CMNWP bit enables or disables starting the updating of counter values by writing to the SECSELn bit (n = 0 to 2) corresponding to a channel number in the GTSECSR register or to the GTSECR register.

The bit position of each SECSEL bit in the GTSECSR registers is allocated to the channel with the corresponding number, and writing to the GTSECSR register for any channel results in writing to the registers of all channels. Writing to the GTSECR register of any channel leads to writing to the registers of all channels. The CMNWP bit for each channel does not control writing but only controls updating of the SECSEL bit and the GTSECR register value for the corresponding channel in the case of simultaneous writing to all channels.

Thus, in the case of writing to the SECSEL bit and the GTSECR register value of a channel for which the setting of the CMNWP bit is 1 (disabling writing), the SECSEL bit and the GTSECR register value for the given channel will not be updated, but the SECSEL bit and the GTSECR register value corresponding to channel for which the setting of the CMNWP bit is 0 (enabling writing) will be updated.

For example, when the setting of the GPTW0.GTWP.CMNWP bit is 0 (enabling writing), writing to the GPTW1.GTSECSR.SECSEL0 bit will lead to updating of the value of the GPTW0.GTSECSR.SECSEL0 bit. In the same way, writing to the GPTW1.GTSECR register will update the value of the GPTW0.GTSECR register. When the setting of the GPTW0.GTWP.CMNWP bit is 1 (disabling writing), writing to the GPTW1.GTSECSR.SECSEL0 bit will not lead to updating of the value of the GPTW0.GTSECSR.SECSEL0 bit. In the same way, writing to the GPTW1.GTSECR register will not lead to updating of the value of the GPTW0.GTSECR register.

If you want to protect all bits in the GTSECSR and GTSECR registers from being updated, set the CMNWP bits of all channels to 1.

PRKEY[7:0] Bits (GTWP Key Code)

This bit controls whether the WP, STRWP, STPWP, CLRWP, and CMNWP bits can be overwritten.

21.2.2 General PWM Timer Software Start Register (GTSTR)

Address(es): GPTW0.GTSTR 0009 6004h, GPTW1.GTSTR 0009 6104h, GPTW2.GTSTR 0009 6204h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CSTRT 2	CSTRT 1	CSTRT 0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CSTRT0	Channel 0 Count Start	When reading 0: The counter is stopped. 1: The counter is running. When writing 0: Ignored. 1: The counter starts running.	R/W
b1	CSTRT1	Channel 1 Count Start		R/W
b2	CSTRT2	Channel 2 Count Start		R/W
b31 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTSTR register starts the GTCNT counter operation.

A bit position of the GTSTR register indicates a channel number. The GTSTR register of each channel is a common register, and writing 1 to the GTSTR register in any channel and updating it can start operation of the GTCNT counter in all the channels related to the position of the bit written with 1. A change in counter operation and the register value is not generated by writing 0 to the bit.

CSTRTn Bit (Channel n Count Start) (n = 0 to 2)

This bit starts the GTCNT counter for channel n.

A read value indicates the state of the counter operation in a corresponding channel (GTCR.CST bit). The bit with 0 indicates that the counter is stopped, while the bit with 1 indicates that the counter is in operation.

21.2.3 General PWM Timer Software Stop Register (GTSTP)

Address(es): GPTW0.GTSTP 0009 6008h, GPTW1.GTSTP 0009 6108h, GPTW2.GTSTP 0009 6208h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CSTOP ₂	CSTOP ₁	CSTOP ₀
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	CSTOP0	Channel 0 Count Stop	When reading 0: The counter is operating. 1: The counter is stopped. When writing 0: Ignored. 1: The counter is stopped.	R/W
b1	CSTOP1	Channel 1 Count Stop		R/W
b2	CSTOP2	Channel 2 Count Stop		R/W
b31 to b3	—	Reserved	These bits are read as 1. The write value should be 0.	R/W

The GTSTP register stops the GTCNT counter operation.

A bit position of the GTSTP register indicates a channel number. The GTSTP register of each channel is a common register, and writing 1 to the GTSTP register in any channel and updating it can stop operation of the GTCNT counter in all the channels related to the position of the bit written with 1. A change in counter operation and the register value is not generated by writing 0 to the bit.

CSTOPn Bit (Channel n Count Stop) (n = 0 to 2)

This bit stops the GTCNT counter for channel n.

A read value indicates the state of the counter operation in a corresponding channel (inverted GTCR.CST bit). The bit with 0 indicates that the counter is in operation, while the bit with 1 indicates that the counter is stopped.

21.2.4 General PWM Timer Software Clear Register (GTCLR)

Address(es): GPTW0.GTCLR 0009 600Ch, GPTW1.GTCLR 0009 610Ch, GPTW2.GTCLR 0009 620Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CCLR2	CCLR1	CCLR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CCLR0	Channel 0 Counter Clear	0: Ignored. 1: The counter is to be cleared.	W
b1	CCLR1	Channel 1 Counter Clear		W
b2	CCLR2	Channel 2 Counter Clear		W
b31 to b3	—	Reserved	The write value should be 0.	W

The GTCLR register is a write-only register which sets clearing of the GTCNT counter.

A bit position of the GTCLR register indicates a channel number. The GTCLR register of each channel is a common register, and writing 1 to the GTCLR register in any channel and updating it changes to 0 of the GTCNT counter in all the channels related to the position of the bit written with 1. A change in counter operation and the register value is not generated by writing 0 to the bit.

CCLRn Bit (Channel n Counter Clear) (n = 0 to 2)

When the counting direction flag is set for decrementation (GTST.TUCF flag = 0) with sawtooth-wave mode selected in the GTCR.MD[2:0] bits, the value of the GTCNT counter becomes that of the corresponding GTPR register in response to writing 1 to the CCLRn bit. The value of the counter becomes 0000 0000h with other settings.

21.2.5 General PWM Timer Start Source Select Register (GTSSR)

Address(es): GPTW0.GTSSR 0009 6010h, GPTW1.GTSSR 0009 6110h, GPTW2.GTSSR 0009 6210h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
CSTRT	—	—	—	—	—	—	—	—	—	—	—	—	—	SSMTU 4	SSMTU 0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
SSCBF AH	SSCBF AL	SSCBR AH	SSCBR AL	SSCAF BH	SSCAF BL	SSCAR BH	SSCAR BL	—	—	—	—	—	—	SSGTR GAF	SSGTR GAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SSGTRGAR	GTETRGA Signal Edge Select	b1 b0 0 0: The GTETRGA signal is not used as a trigger to start counting. 0 1: The counter starts at a rising edge of the GTETRGA signal. 1 0: The counter starts at a falling edge of the GTETRGA signal. 1 1: The counter starts at both edges of the GTETRGA signal.	R/W
b1	SSGTRGAF			R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	SSCARBL	GTIOCnA Signal Rising Edge Applying Condition Select	b9 b8 0 0: Rising edge of the GTIOCnA signal is not used as a trigger to start counting. 0 1: The counter starts at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter starts at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter starts at a rising edge of the GTIOCnA signal.	R/W
b9	SSCARBH			R/W
b10	SSCAFBL	GTIOCnA Signal Falling Edge Applying Condition Select	b11 b10 0 0: Falling edge of the GTIOCnA signal is not used as a trigger to start counting. 0 1: The counter starts at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter starts at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter starts at a falling edge of the GTIOCnA signal.	R/W
b11	SSCAFBH			R/W
b12	SSCBRAL	GTIOCnB Signal Rising Edge Applying Condition Select	b13 b12 0 0: Rising edge of the GTIOCnB signal is not used as a trigger to start counting. 0 1: The counter starts at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven low. 1 0: The counter starts at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven high. 1 1: The counter starts at a rising edge of the GTIOCnB signal.	R/W
b13	SSCBRAH			R/W
b14	SSCBFAL	GTIOCnB Signal Falling Edge Applying Condition Select	b15 b14 0 0: Falling edge of the GTIOCnB signal is not used as a trigger to start counting. 0 1: The counter starts at a falling edge of the GTIOCnB signal while the GTIOCnA pin is driven low. 1 0: The counter starts at a falling edge of the GTIOCnB signal while the GTIOCnA pin is driven high. 1 1: The counter starts at a falling edge of the GTIOCnB signal.	R/W
b15	SSCBFAH			R/W
b16	SSMTU0	MTU0 Count Start Source Count Start Enable	0: Disables count start by the MTU0 count start 1: Enables count start by the MTU0 count start	R/W

Bit	Symbol	Bit Name	Description	R/W
b17	SSMTU4	MTU4 Count Start Source Count Start Enable	0: Disables count start by the MTU4 count start 1: Enables count start by the MTU4 count start	R/W
b30 to b18	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	CSTRT	Software Source Count Start Enable	0: Disables count start by the GTSTR register 1: Enables count start by the GTSTR register	R/W

n = 0 to 2

The GTSSR register sets a count start source for the GTCNT counter.

Inputs from the GTETRGA pin is input to the GPTW via the POEG. Set the polarity of this signal with the POEG.

SSGTRGAR Bit (GTETRGA Rising Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at rising of the GTETRGA input.

SSGTRGAF Bit (GTETRGA Falling Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at falling of the GTETRGA input.

SSCARBL Bit (GTIOCnA Pin at Rising and GTIOCnB Pin 0 Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at rising of the GTIOCnA pin input while the GTIOCnB pin input is 0.

SSCARBH Bit (GTIOCnA Pin at Rising and GTIOCnB Pin 1 Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at rising of the GTIOCnA pin input while the GTIOCnB pin input is 1.

SSCAFBL Bit (GTIOCnA Pin at Falling and GTIOCnB Pin 0 Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at falling of the GTIOCnA pin input while the GTIOCnB pin input is 0.

SSCAFBL Bit (GTIOCnA Pin at Falling and GTIOCnB Pin 0 Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at falling of the GTIOCnA pin input while the GTIOCnB pin input is 1.

SSCBRAL Bit (GTIOCnB Pin at Rising and GTIOCnA Pin 0 Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at rising of the GTIOCnB pin input while the GTIOCnA pin input is 0.

SSCBRAH Bit (GTIOCnB Pin at Rising and GTIOCnA Pin 1 Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at rising of the GTIOCnB pin input while the GTIOCnA pin input is 1.

SSCBFAL Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 0 Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at falling of the GTIOCnB pin input while the GTIOCnA pin input is 0.

SSCBFAH Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 1 Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter at falling of the GTIOCnB pin input while the GTIOCnA pin input is 1.

SSMTU0 Bit (MTU0 Count Start Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter by MTU count start from the MTU0.

SSMTU4 Bit (MTU4 Count Start Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter by MTU count start from the MTU4.

CSTRT Bit (Software Source Count Start Enable)

Select enable or disable for the count start of the GTCNT counter by the GTSTR register.

21.2.6 General PWM Timer Stop Source Select Register (GTPSR)

Address(es): GPTW0.GTPSR 0009 6014h, GPTW1.GTPSR 0009 6114h, GPTW2.GTPSR 0009 6214h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CSTOP	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PSCBF AH	PSCBF AL	PSCBR AH	PSCBR AL	PSCAF BH	PSCAF BL	PSCAR BH	PSCAR BL	—	—	—	—	—	—	PSGTR GAF	PSGTR GAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PSGTRGAR	GTETRGA Signal Edge Select	b1 b0 0 0: The GTETRGA signal is not used as a trigger to stop counting. 0 1: The counter stops at a rising edge of the GTETRGA signal. 1 0: The counter stops at a falling edge of the GTETRGA signal. 1 1: The counter stops at both edges of the GTETRGA signal.	R/W
b1	PSGTRGAF			R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	PSCARBL	GTIOCnA Signal Rising Edge Applying Condition Select	b9 b8 0 0: Rising edge of the GTIOCnA signal is not used as a trigger to stop counting. 0 1: The counter stops at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter stops at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter stops at a rising edge of the GTIOCnA signal.	R/W
b9	PSCARBH			R/W
b10	PSCAFBL	GTIOCnA Signal Falling Edge Applying Condition Select	b11 b10 0 0: Falling edge of the GTIOCnA signal is not used as a trigger to stop counting. 0 1: The counter stops at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter stops at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter stops at a falling edge of the GTIOCnA signal.	R/W
b11	PSCAFBH			R/W
b12	PSCBRAL	GTIOCnB Signal Rising Edge Applying Condition Select	b13 b12 0 0: Rising edge of the GTIOCnB signal is not used as a trigger to stop counting. 0 1: The counter stops at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven low. 1 0: The counter stops at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven high. 1 1: The counter stops at a rising edge of the GTIOCnB signal.	R/W
b13	PSCBRAH			R/W
b14	PSCBFAL	GTIOCnB Signal Falling Edge Applying Condition Select	b15 b14 0 0: Falling edge of the GTIOCnB signal is not used as a trigger to stop counting. 0 1: The counter stops at a falling edge of the GTIOCnB signal while the GTIOCnA pin is driven low. 1 0: The counter stops at a falling edge of the GTIOCnB signal while the GTIOCnA pin is driven high. 1 1: The counter stops at a falling edge of the GTIOCnB signal.	R/W
b15	PSCBFAH			R/W
b30 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b31	CSTOP	Software Source Count Stop Enable	0: Disables count stop by the GTSTP register 1: Enables count stop by the GTSTP register	R/W

n = 0 to 2

The GTPSR register sets a count stop source for the GTCNT counter.

Inputs from the GTETRGA pin is input to the GPTW via the POEG. Set the polarity of this signal with the POEG.

PSGTRGAR Bit (GTETRGA Rising Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at rising of the GTETRGA input.

PSGTRGAF Bit (GTETRGA Falling Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at falling of the GTETRGA input.

PSCARBL Bit (GTIOCnA Pin at Rising and GTIOCnB Pin 0 Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at rising of the GTIOCnA pin input while the GTIOCnB input (n = 0 to 2) is 0.

PSCARBH Bit (GTIOCnA Pin at Rising and GTIOCnB Pin 1 Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at rising of the GTIOCnA pin input while the GTIOCnB input is 1.

PSCAFBL Bit (GTIOCnA Pin at Falling and GTIOCnB Pin 0 Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at falling of the GTIOCnA pin input while the GTIOCnB input is 0.

PSCAFBH Bit (GTIOCnA Pin at Falling and GTIOCnB Pin 1 Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at falling of the GTIOCnA pin input while the GTIOCnB input is 1.

PSCBRAL Bit (GTIOCnB Pin at Rising and GTIOCnA Pin 0 Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at rising of the GTIOCnB pin input while the GTIOCnA input is 0.

PSCBRAH Bit (GTIOCnB Pin at Rising and GTIOCnA Pin 1 Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at rising of the GTIOCnB pin input while the GTIOCnA input is 1.

PSCBFAL Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 0 Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at falling of the GTIOCnB pin input while the GTIOCnA input is 0.

PSCBFAH Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 1 Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter at falling of the GTIOCnB pin input while the GTIOCnA input is 1.

CSTOP Bit (Software Source Count Stop Enable)

Select enable or disable for the count stop of the GTCNT counter by the GTSTP register.

21.2.7 General PWM Timer Clear Source Select Register (GTCSR)

Address(es): GPTW0.GTCSR 0009 6018h, GPTW1.GTCSR 0009 6118h, GPTW2.GTCSR 0009 6218h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CCLR	—	—	—	—	CSCMSC[2:0]		—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CSCBF AH	CSCBF AL	CSCBR AH	CSCBR AL	CSCAF BH	CSCAF BL	CSCAR BH	CSCAR BL	—	—	—	—	—	—	CSGTR GAF	CSGTR GAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CSGTRGAR	GTETRGA Signal Edge Select	b1 b0 0 0: The GTETRGA signal is not used as a trigger to clear the counter. 0 1: The counter is cleared at a rising edge of the GTETRGA signal. 1 0: The counter is cleared at a falling edge of the GTETRGA signal. 1 1: The counter is cleared at both edges of the GTETRGA signal.	R/W
b1	CSGTRGAF			R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	CSCARBL	GTIOCnA Signal Rising Edge Applying Condition Select	b9 b8 0 0: Rising edge of the GTIOCnA signal is not used as a trigger to clear the counter. 0 1: The counter is cleared at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter is cleared at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter is cleared at a rising edge of the GTIOCnA signal.	R/W
b9	CSCARBH			R/W
b10	CSCAFBL	GTIOCnA Signal Falling Edge Applying Condition Select	b11 b10 0 0: Falling edge of the GTIOCnA signal is not used as a trigger to clear the counter. 0 1: The counter is cleared at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter is cleared at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter is cleared at a falling edge of the GTIOCnA signal.	R/W
b11	CSCAFBH			R/W
b12	CSCBRAL	GTIOCnB Signal Rising Edge Applying Condition Select	b13 b12 0 0: Rising edge of the GTIOCnB signal is not used as a trigger to clear the counter. 0 1: The counter is cleared at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven low. 1 0: The counter is cleared at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven high. 1 1: The counter is cleared at a rising edge of the GTIOCnB signal.	R/W
b13	CSCBRAH			R/W

Bit	Symbol	Bit Name	Description	R/W
b14	CSCBFAL	GTIOCnB Signal Falling Edge Applying Condition Select	b15 b14 0 0: Falling edge of the GTIOCnB signal is not used as a trigger to clear the counter. 0 1: The counter is cleared at a falling edge of the GTIOCnB signal while the GTIOCnA pin is driven low. 1 0: The counter is cleared at a falling edge of the GTIOCnB signal while the GTIOCnA pin is driven high. 1 1: The counter is cleared at a falling edge of the GTIOCnB signal.	R/W
b15	CSCBFAH			R/W
b23 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b26 to b24	CSCMSC[2:0]	Compare Match/Input Capture/Synchronous Counter Clearing Source Counter Clear Enable	b26 b24 0 0 0: Counter clear disabled by Compare match/Input capture/Synchronous counter clearing group. 0 0 1: Counter clear enabled at the GTCCRA register compare match/Input capture. 0 1 0: Counter clear enabled at the GTCCRB register compare match/Input capture. 0 1 1: Counter clear enabled at the GTCCRC register compare match. 1 0 0: Counter clear enabled at the GTCCRD register compare match. 1 0 1: Counter clear enabled at the GTCCRE register compare match. 1 1 0: Counter clear enabled at the GTCCRF register compare match. 1 1 1: Counter clear enabled at the synchronous counter clearing group.	R/W
b30 to b27	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	CCLR	Software Source Counter Clear Enable	0: Disables counter clear by the GTCLR register 1: Enables counter clear by the GTCLR register	R/W

n = 0 to 2

The GTCSR register sets a counter clear source for the GTCNT counter.

Counter clearing can be executed whether the counter is running (the GTCR.CST bit = 1) or stopped (the GTCR.CST bit = 0).

Inputs from the GTETRGA pin is input to the GPTW via the POEG. Set the polarity of this signal with the POEG.

CSGTRGAR Bit (GTETRGA Rising Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at rising of the GTETRGA input.

CSGTRGAF Bit (GTETRGA Falling Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at falling of the GTETRGA input.

CSCARBL Bit (GTIOCnA Pin at Rising and GTIOCnB Pin 0 Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at rising of the GTIOCnA pin input while the GTIOCnB pin input (n = 0 to 2) is 0.

CSCARBH Bit (GTIOCnA Pin at Rising and GTIOCnB Pin 1 Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at rising of the GTIOCnA pin input while the GTIOCnB pin input is 1.

CSCAFBL Bit (GTIOCnA Pin at Falling and GTIOCnB Pin 0 Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at falling of the GTIOCnA pin input while the GTIOCnB pin input is 0.

CSCAFBH Bit (GTIOCnA Pin at Falling and GTIOCnB Pin 1 Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at falling of the GTIOCnA pin input while the GTIOCnB pin input is 1.

CSCBRAL Bit (GTIOCnB Pin at Rising and GTIOCnA Pin 0 Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at rising of the GTIOCnB pin input while the GTIOCnA pin input is 0.

CSCBRAH Bit (GTIOCnB Pin at Rising and GTIOCnA Pin 1 Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at rising of the GTIOCnB pin input while the GTIOCnA pin input is 1.

CSCBFAL Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 0 Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at falling of the GTIOCnB pin input while the GTIOCnA pin input is 0.

CSCBFAH Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 1 Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter at falling of the GTIOCnB pin input while the GTIOCnA pin input is 1.

CSCMSC[2:0] Bits (Compare Match/Input Capture/Synchronous Counter Clearing Source Counter Clear Enable)

Select whether to enable or disable for the counter clear of the GTCNT counter by compare match/input capture/synchronous counter clearing group. When counter clearing in response to a match in comparison or input capture is enabled, either can be used as a source for synchronous clearing by inter channel cooperation as described in section 21.3.8.3, Synchronous Clear Operation by Inter Channel Cooperation.

In addition, an input capture triggered by a source from another channel can be used as a source for counter clearing (by setting the GTICASR.ASOC bit or GTICBSR.BSOC bit to 1). Input capture triggered by a source from another channel cannot be used as the synchronous clearing source in interlinked channel operation but can be used as a counter-clearing source for the channel itself.

Since the compare match by the register that is performing the buffer operation (including the wave mode specific case) does not occur, the counter clear enable setting that makes the target register of the buffer operation the compare match factor is invalid.

CCLR Bit (Software Source Counter Clear Enable)

Select enable or disable for the counter clear of the GTCNT counter by GTCLR register.

21.2.8 General PWM Timer Input Capture Source Select Register A (GTICASR)

Address(es): GPTW0.GTICASR 0009 6024h, GPTW1.GTICASR 0009 6124h, GPTW2.GTICASR 0009 6224h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	ASOC	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ASCBF AH	ASCBF AL	ASCBR AH	ASCBR AL	ASCAF BH	ASCAF BL	ASCAR BH	ASCAR BL	—	—	—	—	—	—	ASGTR GAF	ASGTR GAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ASGTRGAR	GTETRGA Signal Edge Select	b1 b0 0 0: The GTETRGA signal is not used as a trigger to capture the counter value in the GTCCRA register. 0 1: The counter value is captured in the GTCCRA register at a rising edge of the GTETRGA signal. 1 0: The counter value is captured in the GTCCRA register at a falling edge of the GTETRGA signal. 1 1: The counter value is captured in the GTCCRA register at both edges of the GTETRGA signal.	R/W
b1	ASGTRGAF			R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	ASCARBL	GTIOCnA Signal Rising Edge Applying Condition Select	b9 b8 0 0: Rising edge of the GTIOCnA signal is not used as a trigger to capture the counter value in the GTCCRA register. 0 1: The counter value is captured in the GTCCRA register at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter value is captured in the GTCCRA register at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter value is captured in the GTCCRA register at a rising edge of the GTIOCnA signal.	R/W
b9	ASCARBH			R/W
b10	ASCAFBL	GTIOCnA Signal Falling Edge Applying Condition Select	b11 b10 0 0: Falling edge of the GTIOCnA signal is not used as a trigger to capture the counter value in the GTCCRA register. 0 1: The counter value is captured in the GTCCRA register at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter value is captured in the GTCCRA register at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter value is captured in the GTCCRA register at a falling edge of the GTIOCnA signal.	R/W
b11	ASCAFBH			R/W
b12	ASCBRAL	GTIOCnB Signal Rising Edge Applying Condition Select	b13 b12 0 0: Rising edge of the GTIOCnB signal is not used as a trigger to capture the counter value in the GTCCRA register. 0 1: The counter value is captured in the GTCCRA register at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven low. 1 0: The counter value is captured in the GTCCRA register at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven high. 1 1: The counter value is captured in the GTCCRA register at a rising edge of the GTIOCnB signal.	R/W
b13	ASCBRAH			R/W

Bit	Symbol	Bit Name	Description	R/W
b14	ASCBFAL	GTIOcNB Signal Falling Edge Applying Condition Select	b15 b14 0 0: Falling edge of the GTIOcNB signal is not used as a trigger to capture the counter value in the GTCCRA register. 0 1: The counter value is captured in the GTCCRA register at a falling edge of the GTIOcNB signal while the GTIOcNA pin is driven low. 1 0: The counter value is captured in the GTCCRA register at a falling edge of the GTIOcNB signal while the GTIOcNA pin is driven high. 1 1: The counter value is captured in the GTCCRA register at a falling edge of the GTIOcNB signal.	R/W
b15	ASCBFAH			R/W
b23 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b24	ASOC	Other Channel Source GTCCRA Input Capture Enable	0: Disables GTCCRA input capture by the other channel factor 1: Enables GTCCRA input capture by the other channel factor	R/W
b31 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

n = 0 to 2

The GTICASR register sets an input capture source for the GTCCRA register.

When at least one bit among bits in the GTICASR register is set to 1, input capture operation making the GTCCRA register as an input capture register is performed.

Inputs from the GTETRGA pin is input to the GPTW via the POEG. Set the polarity of this signal with the POEG.

ASGTRGAR Bit (GTETRGA Rising Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at rising of GTETRGA input.

ASGTRGAF Bit (GTETRGA Falling Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at falling of GTETRGA input.

ASCARBL Bit (GTIOcNA Pin at Rising and GTIOcNB Pin 0 Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at rising of GTIOcNA pin input while the GTIOcNB pin input (n = 0 to 2) is 0.

ASCARBH Bit (GTIOcNA Pin at Rising and GTIOcNB Pin 1 Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at rising of GTIOcNA pin input while the GTIOcNB pin input is 1.

ASCAFBL Bit (GTIOcNA Pin at Falling and GTIOcNB Pin 0 Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at falling of GTIOcNA pin input while the GTIOcNB pin input is 0.

ASCAFBL Bit (GTIOcNA Pin at Falling and GTIOcNB Pin 1 Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at falling of GTIOcNA pin input while the GTIOcNB pin input is 1.

ASCBRAL Bit (GTIOcNB Pin at Rising and GTIOcNA Pin 0 Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at rising of GTIOcNB pin input while the GTIOcNA pin input is 0.

ASCBRAH Bit (GTIOCnB Pin at Rising and GTIOCnA Pin 1 Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at rising of GTIOCnB pin input while the GTIOCnA pin input is 1.

ASCBFAL Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 0 Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at falling of GTIOCnB pin input while the GTIOCnA pin input is 0.

ASCBFAH Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 1 Source GTCCRA Input Capture Enable)

Select enable or disable for an input capture to the GTCCRA register at falling of GTIOCnB pin input while the GTIOCnA pin input is 1.

ASOC Bit (Other Channel Source GTCCRA Input Capture Enable)

This bit enables or disables triggering of input capture of a timer value by the GTCCRA register through interlinked channel operation with the trigger source specified in the GTICCR register in another channel. Input capture triggered by a source in another channel enabled by this bit cannot be used as a source to trigger input capture in another channel through specification by the ICAFA and ICBFA bits in the GTICCR register in its own channel.

21.2.9 General PWM Timer Input Capture Source Select Register B (GTICBSR)

Address(es): GPTW0.GTICBSR 0009 6028h, GPTW1.GTICBSR 0009 6128h, GPTW2.GTICBSR 0009 6228h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	BSOC	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BSCBF AH	BSCBF AL	BSCBR AH	BSCBR AL	BSCAF BH	BSCAF BL	BSCAR BH	BSCAR BL	—	—	—	—	—	—	BSGTR GAF	BSGTR GAR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	BSGTRGAR	GTETRGA Signal Edge Select	b1 b0 0 0: The GTETRGA signal is not used as a trigger to capture the counter value in the GTCCRB register. 0 1: The counter value is captured in the GTCCRB register at a rising edge of the GTETRGA signal. 1 0: The counter value is captured in the GTCCRB register at a falling edge of the GTETRGA signal. 1 1: The counter value is captured in the GTCCRB register at both edges of the GTETRGA signal.	R/W
b1	BSGTRGAF			R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	BSCARBL	GTIOCnA Signal Rising Edge Applying Condition Select	b9 b8 0 0: Rising edge of the GTIOCnA signal is not used as a trigger to capture the counter value in the GTCCRB register. 0 1: The counter value is captured in the GTCCRB register at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter value is captured in the GTCCRB register at a rising edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter value is captured in the GTCCRB register at a rising edge of the GTIOCnA signal.	R/W
b9	BSCARBH			R/W
b10	BSCAFBL	GTIOCnA Signal Falling Edge Applying Condition Select	b11 b10 0 0: Falling edge of the GTIOCnA signal is not used as a trigger to capture the counter value in the GTCCRB register. 0 1: The counter value is captured in the GTCCRB register at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven low. 1 0: The counter value is captured in the GTCCRB register at a falling edge of the GTIOCnA signal while the GTIOCnB pin is driven high. 1 1: The counter value is captured in the GTCCRB register at a falling edge of the GTIOCnA signal.	R/W
b11	BSCAFBH			R/W
b12	BSCBRAL	GTIOCnB Signal Rising Edge Applying Condition Select	b13 b12 0 0: Rising edge of the GTIOCnB signal is not used as a trigger to capture the counter value in the GTCCRB register. 0 1: The counter value is captured in the GTCCRB register at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven low. 1 0: The counter value is captured in the GTCCRB register at a rising edge of the GTIOCnB signal while the GTIOCnA pin is driven high. 1 1: The counter value is captured in the GTCCRB register at a rising edge of the GTIOCnB signal.	R/W
b13	BSCBRAH			R/W

Bit	Symbol	Bit Name	Description	R/W
b14	BSCBFAL	GTIOcNB Signal Falling Edge Applying Condition Select	b15 b14 0 0: Falling edge of the GTIOcNB signal is not used as a trigger to capture the counter value in the GTCCRB register. 0 1: The counter value is captured in the GTCCRB register at a falling edge of the GTIOcNB signal while the GTIOcNA pin is driven low. 1 0: The counter value is captured in the GTCCRB register at a falling edge of the GTIOcNB signal while the GTIOcNA pin is driven high. 1 1: The counter value is captured in the GTCCRB register at a falling edge of the GTIOcNB signal.	R/W
b15	BSCBFAH			R/W
b23 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b24	BSOC	Other Channel Source GTCCRB Input Capture Enable	0: Disables GTCCRB input capture by the other channel factor 1: Enables GTCCRB input capture by the other channel factor	R/W
b31 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

n = 0 to 2

The GTICBSR register sets an input capture source for the GTCCRB register.

When at least one bit among bits in the GTICBSR register is set to 1, input capture operation making the GTCCRB register as an input capture register is performed.

Inputs from the GTETRGA pin is input to the GPTW via the POEG. Set the polarity of this signal with the POEG.

BSGTRGAR Bit (GTETRGA Rising Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at rising of GTETRGA input.

BSGTRGAF Bit (GTETRGA Falling Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at falling of GTETRGA input.

BSCARBL Bit (GTIOcNA Pin at Rising and GTIOcNB Pin 0 Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at rising of GTIOcNA pin input while the GTIOcNB pin input (n = 0 to 2) is 0.

BSCARBH Bit (GTIOcNA Pin at Rising and GTIOcNB Pin 1 Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at rising of GTIOcNA pin input while the GTIOcNB pin input is 1.

BSCAFBL Bit (GTIOcNA Pin at Falling and GTIOcNB Pin 0 Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at falling of GTIOcNA pin input while the GTIOcNB pin input is 0.

BSCAFBH Bit (GTIOcNA Pin at Falling and GTIOcNB Pin 1 Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at falling of GTIOcNA pin input while the GTIOcNB pin input is 1.

BSCBRAL Bit (GTIOcNB Pin at Rising and GTIOcNA Pin 0 Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at rising of GTIOcNB pin input while the GTIOcNA pin input is 0.

BSCBRAH Bit (GTIOCnB Pin at Rising and GTIOCnA Pin 1 Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at rising of GTIOCnB pin input while the GTIOCnA pin input is 1.

BSCBFAL Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 0 Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at falling of GTIOCnB pin input while the GTIOCnA pin input is 0.

BSCBFAH Bit (GTIOCnB Pin at Falling and GTIOCnA Pin 1 Source GTCCRB Input Capture Enable)

Select enable or disable for an input capture to the GTCCRB register at falling of GTIOCnB pin input while the GTIOCnA pin input is 1.

BSOC Bit (Other Channel Source GTCCRB Input Capture Enable)

This bit enables or disables triggering of input capture of a timer value by the GTCCRB register through interlinked channel operation with the trigger source specified in the GTICCR register in another channel. Input capture triggered by a source in another channel enabled by this bit cannot be used as a source to trigger input capture in another channel through specification by the ICAFB and ICBFB bits in the GTICCR register in its own channel.

21.2.10 General PWM Timer Control Register (GTCR)

Address(es): GPTW0.GTCR 0009 602Ch, GPTW1.GTCR 0009 612Ch, GPTW2.GTCR 0009 622Ch

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	—	—	—	TPCS[3:0]	—	—	—	—	—	—	MD[2:0]	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
SSCEN	—	—	—	SSCGRP[1:0]	SCGTIOC	ICDS	—	—	—	—	AINV	—	—	—	CST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CST	Count Start	0: Count operation is stopped 1: Count operation is started	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	AINV	GTIOCnA input/output pin polarity reversal control	0: Not invert 1: Invert	R/W
b5	BINV	GTIOCnB input/output pin polarity reversal control	0: Not invert 1: Invert	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	ICDS	Input Capture Operation Select at Count Stop	0: Input capture is operated at count stop 1: Input capture is not operated at count stop	R/W
b9	SCGTIOC	GTIOC Input Source Synchronous Clear Enable	0: Disables to use the counter clear by GTIOC input as the clear factor for other channels 1: Enables to use the counter clear by GTIOC input as the clear factor for other channels	R/W
b11, b10	SSCGRP[1:0]	Synchronous Set/Clear Group Select	b11 b10 0 0: Select synchronous set/clear group A 0 1: Select synchronous set/clear group B 1 0: Select synchronous set/clear group C 1 1: Select synchronous set/clear group D	R/W
b14 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	SSCEN	Synchronous Set/Clear Enable	0: Disable Synchronous set/clear of the GTCNT counter 1: Enable Synchronous set/clear of the GTCNT counter	R/W
b18 to b16	MD[2:0]	Mode Select	b18 b16 0 0 0: Sawtooth-wave PWM mode (single buffer or double buffer possible) 0 0 1: Sawtooth-wave one-shot pulse mode (fixed buffer operation) 0 1 0: Setting prohibited 0 1 1: Setting prohibited 1 0 0: Triangle-wave PWM mode 1 (32-bit transfer at trough) (single buffer or double buffer possible) 1 0 1: Triangle-wave PWM mode 2 (32-bit transfer at crest and trough) (single buffer or double buffer possible) 1 1 0: Triangle-wave PWM mode 3 (64-bit transfer at trough) (fixed buffer operation) 1 1 1: Setting prohibited	R/W
b22 to b19	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b26 to b23	TPCS[3:0]	Timer Prescaler Select	b26 b23 0 0 0 0: PCLKB 0 0 0 1: PCLKB/2 0 0 1 0: PCLKB/4 0 0 1 1: PCLKB/8 0 1 0 0: PCLKB/16 0 1 0 1: PCLKB/32 0 1 1 0: PCLKB/64 0 1 1 1: Setting prohibited 1 0 0 0: PCLKB/256 1 0 0 1: Setting prohibited 1 0 1 0: PCLKB/1024 1 0 1 1: Setting prohibited 1 1 0 0: GTETRGA (via the POEG) 1 1 0 1: Setting prohibited 1 1 1 0: Setting prohibited 1 1 1 1: Setting prohibited	R/W
b31 to b27	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTCR register controls the GTCNT counter.

CST Bit (Count Start)

This bit controls start or stop for the GTCNT counter.

[Setting conditions]

- When 1 is written to a bit related to a channel number for the GTSTR register while the GTSSR.CSTRT bit is set to 1
- When an external trigger, or a condition for the GTIOCnA pin input and GTIOCnB pin input (n = 0 to 2), which is enabled as a count start source by the GTSSR register, is generated
- 1 is written by software

[Clearing conditions]

- When 1 is written to a bit related to a channel number for the GTSTP register while the GTPSR.CSTOP bit is set to 1
- When an external trigger, or a condition for the GTIOCnA pin input and GTIOCnB pin input, which is enabled as a count stop source by the GTPSR register, is generated
- 0 is written by software

AINV Bit (GTIOCnA input/output pin polarity reversal control)

This bit selects GTIOCnA input/output pin polarity reversal control.

BINV Bit (GTIOCnB input/output pin polarity reversal control)

This bit selects GTIOCnB input/output pin polarity reversal control.

ICDS Bit (Input Capture Operation Select at Count Stop)

This bit selects Input capture operation at count stop when the input capture function is selected.

SCGTIOC Bit (GTIOC Input Source Synchronous Clear Enable)

This bit selects enables or disables the use of counter clearing at the GTIOCnA/GTIOCnB input pin selected by the GTCSR register as a counter clear factor for other channels.

SSGRP[1:0] Bits (Synchronous Set/Clear Group Select)

This bit selects the synchronization set/clear channel group.

SSCEN Bit (Synchronous Set/Clear Enable)

This bit selects disable or enable of Synchronous set/clear.

MD[2:0] Bits (Mode Select)

These bits select the GPTW operating mode.

Only the MD[2] bit is valid at input capture.

Counting in sawtooth-wave mode is performed with 0b for the MD[2] bit, counting in triangle-wave mode is performed with 1b for the MD[2] bit.

Set the MD[2:0] bits while the GTCNT counter operation is stopped.

TPCS[3:0] Bits (Timer Prescaler Select)

These bits select a clock for the GTCNT counter. The clock source can be selected independently for each channel.

Set the TPCS[3:0] bits while the GTCNT counter operation is stopped.

21.2.11 General PWM Timer Count Direction and Duty Setting Register (GTUDDTYC)

Address(es): GPTW0.GTUDDTYC 0009 6030h, GPTW1.GTUDDTYC 0009 6130h, GPTW2.GTUDDTYC 0009 6230h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	OABDTYT	OBDR	ODTYF	ODTY[1:0]	—	—	—	—	—	OADTYR	OADTYF	OADTY[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UDF	UD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	UD	Count Direction Setting	0: GTCNT counts down. 1: GTCNT counts up.	R/W
b1	UDF	Forcible Count Direction Setting	0: Count direction is not forcibly set. 1: Count direction is forcibly set.	R/W
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b17, b16	OADTY[1:0]	GTIOcNA Pin Output Duty Setting	b17 b16 0 x: Compare matches determine the duty cycle of the output on the GTIOcNA pin. 1 0: The duty cycle of the output on the GTIOcNA pin is 0%. 1 1: The duty cycle of the output on the GTIOcNA pin is 100%.	R/W
b18	OADTYF	GTIOcNA Pin Output Duty Forced Setting	0: Duty of the GTIOcNA pin output is not forcibly set. 1: Duty of the GTIOcNA pin output is forcibly set.	R/W
b19	OADTYR	Output after Release of GTIOcNA Pin Output 0%/100% Duty Cycle Settings	0: The function selected by the GTIOA[3:2] bits is applied to the output value when the duty cycle is set after release from the 0 or 100% duty-cycle setting. 1: The function selected by the GTIOA[3:2] bits is applied to the compare match output value which is masked after release from the 0 or 100% duty-cycle setting.	R/W
b23 to b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b25, b24	ODTY[1:0]	GTIOcNB Pin Output Duty Setting	b25 b24 0 x: Compare matches determine the duty cycle of the output on the GTIOcNB pin. 1 0: The duty cycle of the output on the GTIOcNB pin is 0%. 1 1: The duty cycle of the output on the GTIOcNB pin is 100%.	R/W
b26	ODTYF	GTIOcNB Pin Output Duty Forced Setting	0: Duty of the GTIOcNB pin output is not forcibly set. 1: Duty of the GTIOcNB pin output is forcibly set.	R/W
b27	ODTYR	Output after Release of GTIOcNB Pin Output 0%/100% Duty Cycle Settings	0: The function selected by the GTIOB[3:2] bits is applied to the output value when the duty cycle is set after release from the 0 or 100% duty-cycle setting. 1: The function selected by the GTIOB[3:2] bits is applied to the compare match output value which is masked after release from the 0 or 100% duty-cycle setting.	R/W
b28	OABDTYT	GTIOcNA, B pin output 0%/100% duty setting reflection timing setting	0: The output duty-cycle setting changed at an underflow is reflected. 1: The output duty-cycle setting changed at an underflow and at the top of triangle-wave is reflected. The setting value 1 is invalid except the triangle-wave PWM	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

n = 0 to 2

The GTUDDTYC register sets the count direction (up-counting/down-counting) for the GTCNT counter and configures duty setting for output from the GTIOcNA and GTIOcNB pins.

(1) Setting of Count Direction

- In sawtooth-wave mode

When the UD bit value is set to 0 during up-counting, the count direction is changed at an overflow (count clock when the GTCNT counter value is equal to the GTPR register value). When the UD bit value is set to 1 during down-counting, the count direction is changed at an underflow (count clock when the GTCNT counter value is equal to 0000 0000h). If the UD bit value is changed from 1 to 0 with the UDF bit being 0 and while counting is stopped, the counter starts up-counting and the count direction is changed at an overflow (count clock when the GTCNT counter value is equal to the GTPR register value).

If the UD bit value is changed from 0 to 1 with the UDF bit being 0 and while counting is stopped, the counter starts down-counting and the count direction is changed at an underflow (count clock when the GTCNT counter value is equal to 0000 0000h).

When the UDF bit is set to 1 while counting is stopped, the UD bit value at that time is reflected in the count direction when counting starts.

- In triangle-wave mode

Even if the UD bit value is changed during counting, the change will not be reflected in the count direction.

If the UD bit value is modified while the UDF bit is 0 and counting is stopped, the change will not be reflected in the count direction when counting starts. If the UDF bit is set to 1 while counting is stopped, the UD bit value at that time is reflected in the count direction when counting starts.

UD Bit (Count Direction Setting)

This bit sets the count direction (up-counting or down-counting) for the GTCNT counter.

UDF Bit (Forcible Count Direction Setting)

This bit forcibly sets the count direction when the GTCNT counter starts operation as the UD bit value.

Only 0 should be written to this bit during count operation.

When 1 has been written to this bit while counting is stopped, this bit should be returned to 0 before counting starts.

(2) Setting of Output Duty

- In sawtooth-wave mode

When the OADTY[1:0]/OBDTY[1:0] bits are changed during up-count operation, the duty setting changed at an overflow is reflected.

When the OADTY[1:0]/OBDTY[1:0] bits are changed during down-count operation, the duty setting changed at an underflow is reflected.

When the OADTYF/OBDTYF bit is 0 while count operation is stopped and the OADTY[1:0]/OBDTY[1:0] bits are changed, the change in the duty-cycle setting is not reflected in the first count operation, but the change is reflected on an overflow in the case of counting up and on an underflow in the case of counting down.

When the OADTYF/OBDTYF bit is 1 while count operation is stopped and the OADTY[1:0]/OBDTY[1:0] bits are changed, the change in the duty-cycle setting is immediately reflected during the first count operation.

- In triangle-wave mode

When the OADTY[1:0]/OBDTY[1:0] bits are changed during count operation, the duty-cycle setting changed at an underflow is reflected.

When the OADTYF/OBDTYF bit is 0 during count operation and the OADTY[1:0]/OBDTY[1:0] bits are changed, the duty-cycle setting changed during the first count operation is not reflected, the duty-cycle setting changed at an underflow is reflected.

When the OADTYF/OBDTYF bit is 1 while count operation is stopped and the OADTY[1:0]/OBDTY[1:0] bits are changed, the change in the duty-cycle setting is immediately reflected during the first count operation.

Whether it's a sawtooth wave or a triangle wave, When the OADTYF/OBDTYF bit is set to 1 during the count stop and

the duty of the first cycle after the count start is set to the OADTY[1:0]/OBDTY[1:0] bits, Even if the OADTYF/OBDTYF bit is set to 0 and the OADTY[1:0]/OBDTY[1:0] bits are set again, When the count starts, the duty of the first cycle and the duty of the next cycle will be the duty setting of the OADTY[1:0]/OBDTY[1:0] bits set during the count stop.

OmDTY[1:0] Bits (GTIOCnm Pin Output Duty Setting) (n = 0 to 2; m = A, B)

These bits set the output of the duty cycle (0%, 100% or compare match control) on the GTIOCnm pin.

OmDTYF Bit (GTIOCnm Pin Output Duty Forced Setting) (n = 0 to 2; m = A, B)

This bit forcibly specifies the duty cycle at the start of the GTCNT counter in the OmDTY[1:0] bits.

Always write 0 to the bit during count operation.

When writing 1 to the bit while count operation is stopped, set the bit back to 0 by the end of the first cycle after the count operation started and specify the next cycle.

OmDTYR Bit (Output after Release of GTIOCnm Pin Output 0%/100% Duty Cycle Settings) (n = 0 to 2; m = A, B)

When the setting of a GTIOCnm pin for a 0% or 100% duty cycle is forcibly changed due to a compare match, the output level on the pin is maintained after the period has elapsed if the value of the GTIOR.GTIOM[3:2] bits is 00b, but maintenance and a value to control toggling are selected after the period has elapsed if the output has become toggled and the setting is 11b.

The GPTW internally continues to perform compare match operation during duty-cycle 0% or 100% operation. When the OmDTYR bit is 1, the value after the period has elapsed due this compare match operation is target for the GTIOM[3:2] bits.

OABDYYT Bit (GTIOCnm pin output 0%/100% duty setting reflection timing setting) (n = 0 to 2; m = A, B)

Select the reflection timing of the 0%/100% duty setting for the GTIOCnm pin in triangle-wave mode.

If the OABDYYT bit is 0, the output duty setting set in the GTUDDTYC.OADTY[1:0] and OBDTY[1:0] bits is reflected when an underflow occurs.

If the OABDYYT bit is 1, the output duty setting set in the GTUDDTYC.OADTY[1:0] and OBDTY[1:0] bits is reflected when underflow occurs and at the top of the triangle-wave.

21.2.12 General PWM Timer I/O Control Register (GTIOR)

Address(es): GPTW0.GTIOR 0009 6034h, GPTW1.GTIOR 0009 6134h, GPTW2.GTIOR 0009 6234h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
NFC SB[1:0]	NFBEN	—	—	OBDF[1:0]	OBE	OBHLD	OBDFLT	—	GTIOB[4:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
NFC SA[1:0]	NFAEN	PSYE	—	OADF[1:0]	OAE	OAHL D	OADFLT	—	GTIOA[4:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	GTIOA[4:0]	GTIOCnA Pin Function Select	See Table 21.4.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	OADFLT	GTIOCnA Pin Output Value Setting at the Count Stop	0: The GTIOCnA pin outputs low when counting is stopped. 1: The GTIOCnA pin outputs high when counting is stopped.	R/W
b7	OAHL D	GTIOCnA Pin Output Retention at the Start/Stop Count	0: The GTIOCnA pin output level at start/stop of counting depends on the register setting. 1: The GTIOCnA pin output level is retained at start/stop of counting.	R/W
b8	OAE	GTIOCnA Pin Output Enable	0: Pin output is disabled 1: Pin output is enabled	R/W
b10, b9	OADF[1:0]	GTIOCnA Pin Negate Value Setting	b10 b9 0 0: None of the following sources is specified 0 1: GTIOCnA pin is placed in the Hi-Z state in response to control for output negation. 1 0: GTIOCnA pin is set to 0 in response to control for output negation. 1 1: GTIOCnA pin is set to 1 in response to control for output negation.	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b12	PSYE	PWM Synchronous Output Enable*1	0: Disable GTCPP00 pin output 1: Enable GTCPP00 pin output	R/W
b13	NFAEN	GTIOCnA Pin Input Noise Filter Enable	0: The noise filter for GTIOCnA pin input is disabled 1: The noise filter for GTIOCnA pin input is enabled	R/W
b15, b14	NFC SA[1:0]	GTIOCnA Pin Input Noise Filter Sampling Clock Select	b15 b14 0 0: PCLKB 0 1: PCLKB/4 1 0: PCLKB/16 1 1: PCLKB/64	R/W
b20 to b16	GTIOB[4:0]	GTIOCnB Pin Function Select	See Table 21.4.	R/W
b21	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b22	OBDFLT	GTIOCnB Pin Output Value Setting at the Count Stop	0: The GTIOCnB pin outputs low when counting is stopped. 1: The GTIOCnB pin outputs high when counting is stopped.	R/W
b23	OBHL D	GTIOCnB Pin Output Retention at the Start/Stop Count	0: The GTIOCnB pin output level at start/stop of counting depends on the register setting. 1: The GTIOCnB pin output level is retained at start/stop of counting.	R/W
b24	OBE	GTIOCnB Pin Output Enable	0: Pin output is disabled 1: Pin output is enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b26, b25	OBDf[1:0]	GTIOCnB Pin Negate Value Setting	b26 b25 0 0: None of the following sources is specified 0 1: GTIOCnB pin is placed in the Hi-Z state in response to control for output negation. 1 0: GTIOCnB pin is set to 0 in response to control for output negation. 1 1: GTIOCnB pin is set to 1 in response to control for output negation.	R/W
b28, b27	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b29	NFBEN	GTIOCnB Pin Input Noise Filter Enable	0: The noise filter for GTIOCnB pin input is disabled 1: The noise filter for GTIOCnB pin input is enabled	R/W
b31, b30	NFCsB[1:0]	GTIOCnB Pin Input Noise Filter Sampling Clock Select	b31 b30 0 0: PCLKB 0 1: PCLKB/4 1 0: PCLKB/16 1 1: PCLKB/64	R/W

n = 0 to 2

Note 1. PWM synchronous output bit is available only for GPTW0.

This bit in each of GPTW1 and GPTW2 is reserved, and is read as 0. When writing, write 0 to the bit.

GTIOR sets the functions of the GTIOCnA, GTIOCnB, and GTCPP0 pins.

GTIOA[4:0] Bits (GTIOCnA Pin Function Select) (n = 0 to 2)

These bits select the GTIOCnA pin function. For details, see Table 21.4.

OADFLT Bit (GTIOCnA Pin Output Value Setting at the Count Stop) (n = 0 to 2)

This bit sets whether the GTIOCnA pin outputs high or low when counting is stopped.

OAHLd Bit (GTIOCnA Pin Output Retention at the Start/Stop Count) (n = 0 to 2)

This bit specifies whether the GTIOCnA pin output level is retained or the level depends on the register setting when counting is started or stopped.

[Setting conditions]

- The output is retained when counting starts or stops.

[Clearing conditions]

- The value specified by the GTIOA[4] bit is output when counting starts.
- The value specified by the OADFLT bit is output when counting stops.
- If the OADFLT bit is modified while counting is stopped, it is immediately reflected in the output.

OAE Bit (GTIOCnA Pin Output Enable) (n = 0 to 2)

This bit selects whether the GTIOCnA pin output is performed or not.

When the GTCCRA register is used as an input capture register (at least one bit among the bits for the GTICASR register is set to 1), the GTIOCnA pin output is not performed regardless of the setting of the OAE bit.

OADF[1:0] Bits (GTIOCnA Pin Negate Value Setting) (n = 0 to 2)

Select the value for output from the GTIOCnA pin in response to a request to negate output from the POEG.

PSYE Bit (PWM Synchronous Output Enable)

This bit sets enable or disable of output signal from GTCPP0 pin synchronized with the PWM cycle that toggles, crest/trough/GTCNT counter clear at Triangle-wave mode, or the end of the cycle of the Sawtooth-wave mode. The initial output of the GTCPP0 output pins is at the low level, and the output goes to the high level when counting starts.

NFAEN Bit (GTIOCnA Pin Input Noise Filter Enable) (n = 0 to 2)

This bit sets enable or disable of the noise filter function for the GTIOCnA pin input. Switch these bits while the corresponding pin function for the timer I/O control register is set as the output compare function, because unintended internal edge may be generated when these bits are switched.

NFCSA[1:0] Bits (GTIOCnA Pin Input Noise Filter Sampling Clock Select) (n = 0 to 2)

These bits set a sampling clock for the noise filter to the GTIOCnA pin input. Set to the input capture function after waiting two cycles for the sampling clock cycle following the setting of these bits.

GTIOB[4:0] Bits (GTIOCnB Pin Function Select) (n = 0 to 2)

These bits select the GTIOCnB pin function. For details, see Table 21.4.

OBDFLT Bit (GTIOCnB Pin Output Value Setting at the Count Stop) (n = 0 to 2)

This bit sets whether the GTIOCnB pin outputs high or low when counting is stopped.

OBHLD Bit (GTIOCnB Pin Output Retention at the Start/Stop Count) (n = 0 to 2)

This bit specifies whether the GTIOCnB pin output level is retained or the level depends on the register setting when counting is started or stopped.

[Setting conditions]

- The output is retained when counting starts or stops.

[Clearing conditions]

- The value specified by the GTIOB[4] bit is output when counting starts.
- The value specified by the OBDFLT bit is output when counting stops.
- If the OBDFLT bit is modified while counting is stopped, it is immediately reflected in the output.

OBE Bit (GTIOCnB Pin Output Enable) (n = 0 to 2)

This bit selects whether the GTIOCnB pin output is performed or not.

When the GTCCRB register is used as an input capture register (at least one bit among the bits for the GTICBSR register is set to 1), the GTIOCnB pin output is not performed regardless of the setting of the OBE bit.

OBDF[1:0] Bits (GTIOCnB Pin Negate Value Setting) (n = 0 to 2)

Select the value for output from the GTIOCnB pin in response to a request to negate output from the POEG.

NFBEN Bit (GTIOCnB Pin Input Noise Filter Enable) (n = 0 to 2)

This bit sets enable or disable of the noise filter function for the GTIOCnB pin input. Switch these bits while the corresponding pin function for the timer I/O control register is set as the output compare function, because unintended internal edge may be generated when these bits are switched.

NFCSB[1:0] Bits (GTIOCnB Pin Input Noise Filter Sampling Clock Select) (n = 0 to 2)

These bits set a sampling clock for the noise filter to the GTIOCnB pin input. Set to the input capture function after waiting two cycles for the sampling clock cycle following the setting of these bits.

Table 21.4 Settings of GTIOA[4:0] Bits (GTIOB[4:0] Bits) (Sawtooth-wave mode, Triangle-wave mode)

GTIOA[4:0] (GTIOB[4:0]) Bits					Function		
b4	b3	b2	b1	b0	b4	b3, b2	b1, b0
0	0	0	0	0	Initial output is Low.	Output retained at the end of a cycle	Output retained at GTCCRA/GTCCRB compare match
0	0	0	0	1			Low output at GTCCRA/GTCCRB compare match
0	0	0	1	0			High output at GTCCRA/GTCCRB compare match
0	0	0	1	1			Output toggled at GTCCRA/GTCCRB compare match
0	0	1	0	0		Low output at the end of a cycle	Output retained at GTCCRA/GTCCRB compare match
0	0	1	0	1			Low output at GTCCRA/GTCCRB compare match
0	0	1	1	0			High output at GTCCRA/GTCCRB compare match
0	0	1	1	1			Output toggled at GTCCRA/GTCCRB compare match
0	1	0	0	0		High output at the end of a cycle	Output retained at GTCCRA/GTCCRB compare match
0	1	0	0	1			Low output at GTCCRA/GTCCRB compare match
0	1	0	1	0			High output at GTCCRA/GTCCRB compare match
0	1	0	1	1			Output toggled at GTCCRA/GTCCRB compare match
0	1	1	0	0		Output toggled at the end of a cycle	Output retained at GTCCRA/GTCCRB compare match
0	1	1	0	1			Low output at GTCCRA/GTCCRB compare match
0	1	1	1	0			High output at GTCCRA/GTCCRB compare match
0	1	1	1	1			Output toggled at GTCCRA/GTCCRB compare match
1	0	0	0	0	Initial output is High.	Output retained at the end of a cycle	Output retained at GTCCRA/GTCCRB compare match
1	0	0	0	1			Low output at GTCCRA/GTCCRB compare match
1	0	0	1	0			High output at GTCCRA/GTCCRB compare match
1	0	0	1	1			Output toggled at GTCCRA/GTCCRB compare match
1	0	1	0	0		Low output at the end of a cycle	Output retained at GTCCRA/GTCCRB compare match
1	0	1	0	1			Low output at GTCCRA/GTCCRB compare match
1	0	1	1	0			High output at GTCCRA/GTCCRB compare match
1	0	1	1	1			Output toggled at GTCCRA/GTCCRB compare match
1	1	0	0	0		High output at the end of a cycle	Output retained at GTCCRA/GTCCRB compare match
1	1	0	0	1			Low output at GTCCRA/GTCCRB compare match
1	1	0	1	0			High output at GTCCRA/GTCCRB compare match
1	1	0	1	1			Output toggled at GTCCRA/GTCCRB compare match
1	1	1	0	0		Output toggled at the end of a cycle	Output retained at GTCCRA/GTCCRB compare match
1	1	1	0	1			Low output at GTCCRA/GTCCRB compare match
1	1	1	1	0			High output at GTCCRA/GTCCRB compare match
1	1	1	1	1			Output toggled at GTCCRA/GTCCRB compare match

Note: In sawtooth-wave mode, "end of a cycle" refers to an overflow (the value of the GTCNT counter changing from that of the GTPR register to 0000 0000h in up-counting), an underflow (the value of the GTCNT counter changing from 0000 0000h to that of the GTPR register in down-counting), or counter clearing. It refers to a trough in triangle-wave mode (the value of the GTCNT counter changing from 0000 0000h to 0000 0001h).

Note: When the timing of an end of a cycle and the timing of a GTCCRA/GTCCRB register compare match are the same in a compare-match operation, the b3 and b2 settings are given priority in sawtooth-wave PWM mode, and the b1 and b0 settings are given priority in any other mode.

21.2.13 General PWM Timer Interrupt Output Setting Register (GTINTAD)

Address(es): GPTW0.GTINTAD 0009 6038h, GPTW1.GTINTAD 0009 6138h, GPTW2.GTINTAD 0009 6238h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	GRPAB L	GRPAB H	GRPD T E	—	—	—	—	—	—	—	—	ADTRB DEN	ADTRB UEN	ADTRA DEN	ADTRA UEN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SCFPU	SCFPO	SCFF	SCFE	SCFD	SCFC	SCFB	SCFA	GTINTPR[1:0]	GTINT F	GTINT E	GTINT D	GTINT C	GTINT B	GTINT A	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	GTINTA	GTCCRA Register Compare Match/ Input Capture Interrupt Enable	0: Interrupt request is disabled. 1: Interrupt request is enabled.	R/W
b1	GTINTB	GTCCRB Register Compare Match/ Input Capture Interrupt Enable	0: Interrupt request is disabled. 1: Interrupt request is enabled.	R/W
b2	GTINTC	GTCCRC Register Compare Match Interrupt Enable	0: Interrupt request is disabled. 1: Interrupt request is enabled.	R/W
b3	GTINTD	GTCCRD Register Compare Match Interrupt Enable	0: Interrupt request is disabled. 1: Interrupt request is enabled.	R/W
b4	GTINTE	GTCCRE Register Compare Match Interrupt Enable	0: Interrupt request is disabled. 1: Interrupt request is enabled.	R/W
b5	GTINTF	GTCCRF Register Compare Match Interrupt Enable	0: Interrupt request is disabled. 1: Interrupt request is enabled.	R/W
b7, b6	GTINTPR[1:0]	GTPR Register Compare Match Interrupt Enable	b7 b6 0 0: Interrupt request is disabled. 0 1: In sawtooth-wave mode, interrupt requests are enabled at overflows. In triangle-wave mode, interrupt requests are enabled at crests. 1 0: In sawtooth-wave mode, interrupt requests are enabled at underflows. In triangle-wave mode, interrupt requests are enabled at troughs. 1 1: In sawtooth-wave mode, interrupt requests are enabled at both overflows and underflows. In triangle-wave mode, interrupt requests are enabled at both crests and troughs.	R/W
b8	SCFA	GTCCRA Register Compare Match/ Input Capture Source Synchronous Clear Enable	0: Disables the GTCCRA register comparison match/ input capture as a source for other channel clears. 1: Enables the GTCCRA register comparison match/ input capture as a source for other channel clears.	R/W
b9	SCFB	GTCCRB Register Compare Match Source Synchronous Clear Enable	0: Disables the GTCCRB register comparison match/ input capture as a source for other channel clears. 1: Enables the GTCCRB register comparison match/ input capture as a source for other channel clears.	R/W
b10	SCFC	GTCCRC Register Compare Match Source Synchronous Clear Enable	0: Disables the GTCCRC register comparison match/ input capture as a source for other channel clears. 1: Enables the GTCCRC register comparison match/ input capture as a source for other channel clears.	R/W
b11	SCFD	GTCCRD Register Compare Match Source Synchronous Clear Enable	0: Disables the GTCCRD register comparison match/ input capture as a source for other channel clears. 1: Enables the GTCCRD register comparison match/ input capture as a source for other channel clears.	R/W
b12	SCFE	GTCCRE Register Compare Match Source Synchronous Clear Enable	0: Disables the GTCCRE register comparison match/ input capture as a source for other channel clears. 1: Enables the GTCCRE register comparison match/ input capture as a source for other channel clears.	R/W

Bit	Symbol	Bit Name	Description	R/W
b13	SCFF	GTCCRF Register Compare Match Source Synchronous Clear Enable	0: Disables the GTCCRF register comparison match/ input capture as a source for other channel clears. 1: Enables the GTCCRF register comparison match/ input capture as a source for other channel clears.	R/W
b14	SCFPO	Overflow Source Synchronous Clear Enable	0: Disables Overflow as a source for other channel clears. 1: Enables Overflow as a source for other channel clears.	R/W
b15	SCFPU	Underflow Source Synchronous Clear Enable	0: Disables Underflow as a source for other channel clears. 1: Enables Underflow as a source for other channel clears.	R/W
b16	ADTRAUEN	GTADTRA Register Compare Match (Up-Counting) A/D Conversion Start Request Enable	0: A/D conversion start request is disabled. 1: A/D conversion start request is enabled.	R/W
b17	ADTRADEN	GTADTRA Register Compare Match (Down-Counting) A/D Conversion Start Request Enable	0: A/D conversion start request is disabled. 1: A/D conversion start request is enabled.	R/W
b18	ADTRBUEN	GTADTRB Register Compare Match (Up-Counting) A/D Conversion Start Request Enable	0: A/D conversion start request is disabled. 1: A/D conversion start request is enabled.	R/W
b19	ADTRBDEN	GTADTRB Register Compare Match (Down-Counting) A/D Conversion Start Request Enable	0: A/D conversion start request is disabled. 1: A/D conversion start request is enabled.	R/W
b27 to b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	GRPDTE	Dead Time Error Output Stop Detection Enable	0: Dead time error output stop detection is disabled 1: Dead time error output stop detection is enabled	R/W
b29	GRPABH	Simultaneous High Output Stop Detection Enable	0: Simultaneous high output stop detection is disabled 1: Simultaneous high output stop detection is enabled	R/W
b30	GRPABL	Simultaneous Low Output Stop Detection Enable	0: Simultaneous low output stop detection is disabled 1: Simultaneous low output stop detection is enabled	R/W
b31	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTINTAD register sets enabling or disabling for interrupt request, A/D conversion start request, and output stop detection.

GTINTA Bit (GTCCRA Register Compare Match/Input Capture Interrupt Enable)

This bit enables or disables interrupt requests by the GTCCRA register compare match/input capture (GTCIA).

GTINTB Bit (GTCCRB Register Compare Match/Input Capture Interrupt Enable)

This bit enables or disables interrupt requests by the GTCCRB register compare match/input capture (GTCIB).

GTINTC Bit (GTCCRC Register Compare Match Interrupt Enable)

This bit enables or disables interrupt requests by the GTCCRC register compare match (GTCIC).

GTINTD Bit (GTCCRD Register Compare Match Interrupt Enable)

This bit enables or disables interrupt requests by the GTCCRD register compare match (GTCID).

GTINTE Bit (GTCCRE Register Compare Match Interrupt Enable)

This bit enables or disables interrupt requests by the GTCCRE register compare match (GTCIE).

GTINTF Bit (GTCCRF Register Compare Match Interrupt Enable)

This bit enables or disables interrupt requests by the GTCCRF register compare match (GTCIF).

GTINTPR[1:0] Bits (GTPR Register Compare Match Interrupt Enable)

These bits enable or disable interrupt requests by a GTPR register compare match (GTCNT counter overflow) and those by a GTCNT counter underflow (GTCIV/GTCIU).

SCFA Bit (GTCCRA Register Compare Match/Input Capture Source Synchronous Clear Enable)

This bit enables or disables the GTCCRA register compare match/input capture as a clearing factor for other channels.

SCFB Bit (GTCCRB Register Compare Match Source Synchronous Clear Enable)

This bit enables or disables the GTCCRB register compare match/input capture as a clearing factor for other channels.

SCFC Bit (GTCCRC Register Compare Match Source Synchronous Clear Enable)

This bit enables or disables the GTCCRC register compare match as a clearing factor for other channels.

SCFD Bit (GTCCRD Register Compare Match Source Synchronous Clear Enable)

This bit enables or disables the GTCCRD register compare match as a clearing factor for other channels.

SCFE Bit (GTCCRE Register Compare Match Source Synchronous Clear Enable)

This bit enables or disables the GTCCRE register compare match as a clearing factor for other channels.

SCFF Bit (GTCCRF Register Compare Match Source Synchronous Clear Enable)

This bit enables or disables the GTCCRF register compare match as a clearing factor for other channels.

SCFPO Bit (Overflow Source Synchronous Clear Enable)

This bit enables or prohibits overflow as a clearing factor for other channels.

SCFPU Bit (Underflow Source Synchronous Clear Enable)

This bit enables or prohibits underflow as a clearing factor for other channels.

ADTRAUEN Bit (GTADTRA Register Compare Match (Up-Counting) A/D Conversion Start Request Enable)

This bit enables or disables A/D conversion start requests generated by the GTADTRA register compare matches during GTCNT counter up-counting.

ADTRADEN Bit (GTADTRA Register Compare Match (Down-Counting) A/D Conversion Start Request Enable)

This bit enables or disables A/D conversion start requests generated by the GTADTRA register compare matches during GTCNT counter down-counting.

ADTRBUEN Bit (GTADTRB Register Compare Match (Up-Counting) A/D Conversion Start Request Enable)

This bit enables or disables A/D conversion start requests generated by the GTADTRB register compare matches during GTCNT counter up-counting.

ADTRBDEN Bit (GTADTRB Register Compare Match (Down-Counting) A/D Conversion Start Request Enable)

This bit enables or disables A/D conversion start requests generated by the GTADTRB register compare matches during GTCNT counter down-counting.

GRPDTE Bit (Dead Time Error Output Stop Detection Enable)

This bit enables or disables the output stop detection by a dead time error.

GRPABH Bit (Simultaneous High Output Stop Detection Enable)

This bit enables or disables the output stop detection when the GTIOCnA and GTIOCnB pins have become high simultaneously.

GRPABL Bit (Simultaneous Low Output Stop Detection Enable)

This bit enables or disables the output stop detection when the GTIOCnA and GTIOCnB pins have become low simultaneously.

21.2.14 General PWM Timer Status Register (GTST)

Address(es): GPTW0.GTST 0009 603Ch, GPTW1.GTST 0009 613Ch, GPTW2.GTST 0009 623Ch

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	OABLF	OABHF	DTEF	—	—	—	ODF	—	—	—	—	ADTRB DF	ADTRB UF	ADTRA DF	ADTRA UF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
TUCF	—	—	—	—	ITCNT[2:0]		TCFPU	TCFPO	TCFF	TCFE	TCFD	TCFC	TCFB	TCFA	
Value after reset:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TCFA	Input Capture/Compare Match Flag A	0: No input capture/compare match of GTCCRA is generated. 1: An input capture/compare match of GTCCRA is generated.	R/(W) *1
b1	TCFB	Input Capture/Compare Match Flag B	0: No input capture/compare match of GTCCRB is generated. 1: An input capture/compare match of GTCCRB is generated.	R/(W) *1
b2	TCFC	Compare Match Flag C	0: No compare match of GTCCRC is generated. 1: An compare match of GTCCRC is generated.	R/(W) *1
b3	TCFD	Compare Match Flag D	0: No compare match of GTCCRD is generated. 1: An compare match of GTCCRD is generated.	R/(W) *1
b4	TCFE	Compare Match Flag E	0: No compare match of GTCCRE is generated. 1: An compare match of GTCCRE is generated.	R/(W) *1
b5	TCFF	Compare Match Flag F	0: No compare match of GTCCRF is generated. 1: An compare match of GTCCRF is generated.	R/(W) *1
b6	TCFPO	Overflow Flag	0: No overflow (crest) occurred. 1: An overflow (crest) occurred.	R/(W) *1
b7	TCFPU	Underflow Flag	0: No underflow (trough) occurred. 1: An underflow (trough) occurred.	R/(W) *1
b10 to b8	ITCNT[2:0]	GTCIV/GTCIU Interrupt Skipping Count Counter	Counter for counting the number of times a timer interrupt has been skipped.	R
b14 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	TUCF	Count Direction Flag	0: The GTCNT counter counts downward. 1: The GTCNT counter counts upward.	R
b16	ADTRAUF	GTADTRA Register Compare Match (Up-Counting) A/D Conversion Start Request Flag	0: No GTADTRA register compare match has occurred in up-counting. 1: A GTADTRA register compare match has occurred in up-counting.	R/(W) *1
b17	ADTRADF	GTADTRA Register Compare Match (Down-Counting) A/D Conversion Start Request Flag	0: No GTADTRA register compare match has occurred in down-counting. 1: A GTADTRA register compare match has occurred in down-counting.	R/(W) *1
b18	ADTRBUF	GTADTRB Register Compare Match (Up-Counting) A/D Conversion Start Request Flag	0: No GTADTRB register compare match has occurred in up-counting. 1: A GTADTRB register compare match has occurred in up-counting.	R/(W) *1
b19	ADTRBDF	GTADTRB Register Compare Match (Down-Counting) A/D Conversion Start Request Flag	0: No GTADTRB register compare match has occurred in down-counting. 1: A GTADTRB register compare match has occurred in down-counting.	R/(W) *1
b23 to b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b24	ODF	Output Stop Request Flag	0: No output stop request has occurred. 1: An output stop request has occurred.	R
b27 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b28	DTEF	Dead Time Error Flag	0: No dead time error has occurred. 1: A dead time error has occurred.	R
b29	OABHF	Simultaneous High Output Flag	0: No simultaneous generation of 1 both for the GTIOCnA and GTIOCnB pins has occurred. 1: A simultaneous generation of 1 both for the GTIOCnA and GTIOCnB pins has occurred.	R
b30	OABLF	Simultaneous Low Output Flag	0: No simultaneous generation of 0 both for the GTIOCnA and GTIOCnB pins has occurred. 1: A simultaneous generation of 0 both for the GTIOCnA and GTIOCnB pins has occurred.	R
b31	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

n = 0 to 2

Note 1. Only 0 can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, be sure to write 0 only to the target flag or flags for clearing and to write 1 to the other flags not for clearing.

The GTST register indicates the status of the GPTW.

TCFA Flag (Input Capture/Compare Match Flag A)

The TCFA flag indicates the status for the input capture or compare match of the GTCCRA register.

[Setting conditions]

- GTCNT = GTCCRA, when the GTCCRA register functions as a compare match register
- The GTCNT counter value is transferred to the GTCCRA register by the input capture signal when the GTCCRA register functions as an input capture register.

[Clearing condition]

- 0 is written to this flag.

TCFB Flag (Input Capture/Compare Match Flag B)

The TCFB flag indicates the status for the input capture or compare match of the GTCCRB register.

[Setting conditions]

- GTCNT = GTCCRB, when the GTCCRB register functions as a compare match register
- The GTCNT counter value is transferred to the GTCCRB register by the input capture signal when the GTCCRB register functions as an input capture register.

[Clearing condition]

- 0 is written to this flag.

TCFC Flag (Compare Match Flag C)

The TCFC flag indicates the status for the compare match of the GTCCRC register.

When the GTCCRC register performs buffer operation, the GTCCRC register doesn't perform compare match.

[Setting conditions]

- GTCNT = GTCCRC

[Clearing condition]

- 0 is written to this flag.

[Not comparing condition]

- GTCR.MD[2:0] = 001b (sawtooth-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (the GTCCRC register performs buffer operation).

TCFD Flag (Compare Match Flag D)

The TCFD flag indicates the status for the compare match of the GTCCRD register.

When the GTCCRD register performs buffer operation, the GTCCRD register doesn't perform compare match.

[Setting conditions]

- GTCNT = GTCCRD

[Clearing condition]

- 0 is written to this flag.

[Not comparing condition]

- GTCR.MD[2:0] = 001b (sawtooth-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (the GTCCRD register performs buffer operation).

TCFE Flag (Compare Match Flag E)

The TCFE flag indicates the status for the compare match of the GTCCRE register.

When the GTCCRE register performs buffer operation, the GTCCRE register doesn't perform compare match.

[Setting conditions]

- GTCNT = GTCCRE

[Clearing condition]

- 0 is written to this flag.

[Not comparing condition]

- GTCR.MD[2:0] = 001b (sawtooth-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (the GTCCRE register performs buffer operation).

TCFF Flag (Compare Match Flag F)

The TCFF flag indicates the status for the compare match of the GTCCRF register.

When the GTCCRF register performs buffer operation, the GTCCRF register doesn't perform compare match.

[Setting conditions]

- GTCNT = GTCCRF

[Clearing condition]

- 0 is written to this flag.

[Not comparing condition]

- GTCR.MD[2:0] = 001b (sawtooth-wave one-shot pulse mode).
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (the GTCCRF register performs buffer operation).

TCFPO Flag (Overflow Flag)

The TCFPO flag indicates when an overflow or crest has occurred.

[Setting conditions]

- In sawtooth-wave mode, an overflow (the GTCNT counter changes from the GTPR register value to 0 in up-counting) has occurred
- In triangle-wave mode, a crest (the GTCNT counter changes from the GTPR register value to GTPR – 1) has occurred
- In counting by hardware sources, an overflow (the GTCNT counter changes from the GTPR register value to 0 in up-counting) has occurred.

[Clearing condition]

- 0 is written to this flag.

TCFPU Flag (Underflow Flag)

The TCFPU flag indicates when an underflow or trough has occurred.

[Setting conditions]

- In sawtooth-wave mode, an underflow (the GTCNT counter changes from 0 to the GTPR register value in down-counting) has occurred
- In triangle-wave mode, a trough (the GTCNT counter changes from 0 to 1) has occurred
- In counting by hardware sources, an underflow (the GTCNT counter changes from 0 to the GTPR register value in down-counting) has occurred.

[Clearing condition]

- 0 is written to this flag.

ITCNT[2:0] Bits (GTCIV/GTCIU Interrupt Skipping Count Counter)

When the GTCIV/GTCIU interrupt skipping function is used (the GTITC.IVTC[1:0] bits are set to a value other than 00b), the counter is incremented by 1 every time the GTCIV/GTCIU interrupt source is generated.

These bits are operated independently from the extended interrupt skipping by the GTEITC register.

[Clearing conditions]

- The GTCIV/GTCIU interrupt skipping function is not used (the GTITC.IVTT[2:0] bits are 000b when the IVTC[1:0] bits are 00b).
- The GTCIV/GTCIU interrupt skipping count matches the specified count (the ITCNT[2:0] bits match the skipping count specified by the IVTT[2:0] bits).
- When the count operation is stopped.

TUCF Flag (Count Direction Flag)

This flag indicates the count direction of the GTCNT counter.

ADTRAUF Flag (GTADTRA Register Compare Match (Up-Counting) A/D Conversion Start Request Flag)

This status flag indicates generation of a GTADTRA register compare match in up-counting.

[Setting condition]

- The GTCNT counter matches the GTADTRA register in up-counting

[Clearing conditions]

- 0 is written to the ADTRAUF flag

ADTRADF Flag (GTADTRA Register Compare Match (Down-Counting) A/D Conversion Start Request Flag)

This status flag indicates generation of a GTADTRA register compare match in down-counting.

[Setting condition]

- The GTCNT counter matches the GTADTRA register in down-counting

[Clearing conditions]

- 0 is written to the ADTRADF flag

ADTRBUF Flag (GTADTRB Register Compare Match (Up-Counting) A/D Conversion Start Request Flag)

This status flag indicates generation of a GTADTRB register compare match in up-counting.

[Setting condition]

- The GTCNT counter matches the GTADTRB register in up-counting

[Clearing conditions]

- 0 is written to the ADTRBUF flag

ADTRBDF Flag (GTADTRB Register Compare Match (Down-Counting) A/D Conversion Start Request Flag)

This status flag indicates generation of a GTADTRB register compare match in down-counting.

[Setting condition]

- The GTCNT counter matches the GTADTRB register in down-counting

[Clearing conditions]

- 0 is written to the ADTRBDF flag

ODF Flag (Output Stop Request Flag)

This flag monitors the output stop request for POEG.

Even if a request to disable output is de-asserted after the request is made, releasing the control of negation on the PWM pins has to wait until the end of the PWM cycle.

DTEF Flag (Dead Time Error Flag)

This flag indicates that the timer output toggle point after the automatic addition of dead time has exceeded the count period.

This flag returns to 0 when the timer output toggle point after the automatic addition of dead time is back within the period.

This flag can only be read from. (Writing 0 to clear the flag is not allowed.)

When an output stop detection and an interrupt by the DTEF flag is enabled (when the GTINTAD.GRPDTE bit is 1), the DTEF flag is output as detection of disabling of output to the POEG, and an GDTE interrupt is generated whenever the DTEF flag changes from 0 to 1.

[Setting condition]

- When a change point of the waveform after the automatic setting of dead time has exceeded the count period (in the following cases)

Up-counting in triangle-wave mode: $GTCCRA \text{ register} - GTDVU \text{ register} \leq 0$

Down-counting in triangle-wave mode: $GTCCRA \text{ register} - GTDVD \text{ register} < 0$

Up-counting in sawtooth-wave one-shot pulse mode:

When $GTCCRA \text{ register} - GTDVU \text{ register} < 0$, or $GTCCRA \text{ register} + GTDVD \text{ register} > GTPR \text{ register}$

Down-counting in sawtooth-wave one-shot pulse mode:

When $GTCCRA \text{ register} + GTDVU \text{ register} > GTPR \text{ register}$, or $GTCCRA \text{ register} - GTDVD \text{ register} < 0$

[Clearing condition]

- The timer output toggle point after the automatic addition of dead time is within the count period.

OABHF Flag (Simultaneous High Output Flag)

This flag indicates that the simultaneous output of 1 both from the GTIOCnA and GTIOCnB pins.

When one of the GTIOCnA and GTIOCnB pins changes to 0, the flag changes to 0.

This flag can only be read from. (Writing 0 to clear the flag is not allowed.)

When the output stop detection by the OABHF flag is enabled (the GTINTAD.GRPABH bit is 1), the OABHF flag is output as the output stop detection to the POEG. The GPTW does not have an interrupt to indicate that outputs have been simultaneous driven to the high level. Use the interrupt function in the POEG if this is necessary.

[Setting condition]

- The GTIOCnA and GTIOCnB pins output 1 simultaneously while both the OAE and OBE bits in the GTIOR

register are set to 1.

[Clearing conditions]

- The GTIOCnA and GTIOCnB pins output different values while both the OAE and OBE bits are set to 1.
- The GTIOCnA and GTIOCnB pins output 0 simultaneously while both the OAE and OBE bits are set to 1.
- At least one of the OAE and OBE bits is set to 0.

OABLF Flag (Simultaneous Low Output Flag)

This flag indicates that the simultaneous output of 0 both from the GTIOCnA and GTIOCnB pins.

When one of the GTIOCnA and GTIOCnB pins changes to 1, the flag changes to 0.

This flag can only be read from. (Writing 0 to clear the flag is not allowed.)

When the output stop detection by the OABLF flag is enabled (the GTINTAD.GRPABL bit is 1), the OABLF flag is output as the output stop detection to the POEG. The GPTW does not have an interrupt to indicate that outputs have been simultaneous driven to the high level. Use the interrupt function in the POEG if this is necessary.

[Setting condition]

- The GTIOCnA and GTIOCnB pins output 0 simultaneously while both the OAE and OBE bits in the GTIOR register are set to 1.

[Clearing conditions]

- The GTIOCnA and GTIOCnB pins output different values while both the OAE and OBE bits are set to 1.
- The GTIOCnA and GTIOCnB pins output 1 simultaneously while both the OAE and OBE bits are set to 1.
- At least one of the OAE and OBE bits is set to 0.

Comparison for the output value to generate the OABHF or OABLF flag uses the value before the compare match output (PWM output) is masked by the output negate function. Even during the output negate condition, compare match operation continues internally, where the OABHF or OABLF flag is updated based on the operation results.

21.2.15 General PWM Timer Buffer Enable Register (GTBER)

Address(es): GPTW0.GTBER 0009 6040h, GPTW1.GTBER 0009 6140h, GPTW2.GTBER 0009 6240h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	ADTDB	ADTTB[1:0]	—	ADTTA	ADTTA[1:0]	—	CCRS	WT	PR[1:0]	CCRB[1:0]	CCRA[1:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
DBRTS	DBRTE	DBRTS	DBRTE	DBRTS	DBRTE	DBRTS	DBRTE	—	—	—	—	BD[3]	BD[2]	BD[1]	BD[0]
ADB	ADB	ADA	ADA	CB	CB	CA	CA								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	BD[0]	GTCCRA/GTCCRB Registers Buffer Operation Disable	0: Buffer operation is enabled 1: Buffer operation is disabled	R/W
b1	BD[1]	GTPR Register Buffer Operation Disable		R/W
b2	BD[2]	GTADTRA/GTADTRB Registers Buffer Operation Disable		R/W
b3	BD[3]	GTDVU/GTDVD Registers Buffer Operation Disable		R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	DBRTECA	GTCCRA Register Double Buffer Repeat Operation Enable	0: GTCCRA register double buffer repeat operation is disabled 1: GTCCRA register double buffer repeat operation is enabled	R/W
b9	DBRTSCA	GTCCRA Register Double Buffer Repeat Operation Period Select	0: Repeats the transfer from the intermediate buffer to the GTCCRA register with trough as the cycle. 1: Repeats the transfer from the intermediate buffer to the GTCCRA register with crest as the cycle.	R/W
b10	DBRTECB	GTCCRB Register Double Buffer Repeat Operation Enable	0: GTCCRB register double buffer repeat operation is disabled 1: GTCCRB register double buffer repeat operation is enabled	R/W
b11	DBRTSCB	GTCCRB Register Double Buffer Repeat Operation Period Select	0: Repeats the transfer from the intermediate buffer to the GTCCRB register with trough as the cycle. 1: Repeats the transfer from the intermediate buffer to the GTCCRB register with crest as the cycle.	R/W
b12	DBRTEADA	GTADTRA Register Double Buffer Repeat Operation Enable	0: GTADTRA register double buffer repeat operation is disabled 1: GTADTRA register double buffer repeat operation is enabled	R/W
b13	DBRTSADA	GTADTRA Register Double Buffer Repeat Operation Period Select	0: Repeats the transfer from the intermediate buffer to the GTADTRA register with trough as the cycle. 1: Repeats the transfer from the intermediate buffer to the GTADTRA register with crest as the cycle.	R/W
b14	DBRTEADB	GTADTRB Register Double Buffer Repeat Operation Enable	0: GTADTRB register double buffer repeat operation is disabled 1: GTADTRB register double buffer repeat operation is enabled	R/W
b15	DBRTSADB	GTADTRB Register Double Buffer Repeat Operation Period Select	0: Repeats the transfer from the intermediate buffer to the GTADTRB register with trough as the cycle. 1: Repeats the transfer from the intermediate buffer to the GTADTRB register with crest as the cycle.	R/W

Bit	Symbol	Bit Name	Description	R/W
b17, b16	CCRA[1:0]	GTCCRA Register Buffer Operation	b17 b16 0 0: Buffer operation is not performed 0 1: Single buffer operation (GTCCRA register ⇔ GTCCRC register) 1 x: Double buffer operation (GTCCRA register ⇔ GTCCRC register ⇔ GTCCRD register)	R/W
b19, b18	CCRB[1:0]	GTCCRB Register Buffer Operation	b19 b18 0 0: Buffer operation is not performed 0 1: Single buffer operation (GTCCRB register ⇔ GTCCRE register) 1 x: Double buffer operation (GTCCRB register ⇔ GTCCRE register ⇔ GTCCRF register)	R/W
b21, b20	PR[1:0]	GTPR Register Buffer Operation	b21 b20 0 0: Buffer operation is not performed 0 1: Single buffer operation (GTPBR register ⇒ GTPR register) 1 x: Double buffer operation (GTPDBR register ⇒ GTPBR register ⇒ GTPR register)	R/W
b22	CCRSWT	GTCCRA and GTCCRB Registers Forcible Buffer Operation	Writing 1 to this bit forcibly performs buffer transfer of GTCCRA and GTCCRB registers. This bit automatically returns to 0 after the writing of 1. This bit is read as 0.	R/W
b23	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b25, b24	ADTTA[1:0]	GTADTRA Register Buffer Transfer Timing Select	- Triangle waves b25 b24 0 0: No transfer 0 1: Transfer at crest 1 0: Transfer at trough 1 1: Transfer at both crest and trough - Sawtooth waves b25 b24 0 0: No transfer Values other than 0 0: Transfer at underflow (during down-counting), overflow (during up-counting), or counter clearing	R/W
b26	ADTDA	GTADTRA Register Double Buffer Operation	0: Single buffer operation (GTADTBRA register ⇒ GTADTRA register) 1: Double buffer operation (GTADTDBRA register ⇒ GTADTBRA register ⇒ GTADTRA register)	R/W
b27	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b29, b28	ADTTB[1:0]	GTADTRB Register Buffer Transfer Timing Select	- Triangle waves b29 b28 0 0: No transfer 0 1: Transfer at crest 1 0: Transfer at trough 1 1: Transfer at both crest and trough - Sawtooth waves b29 b28 0 0: No transfer Values other than 0 0: Transfer at underflow (during down-counting), overflow (during up-counting), or counter clearing	R/W
b30	ADTDB	GTADTRB Register Double Buffer Operation	0: Single buffer operation (GTADTBRB register ⇒ GTADTRB register) 1: Double buffer operation (GTADTDBRB register ⇒ GTADTBRB register ⇒ GTADTRB register)	R/W
b31	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

The GTBER register makes settings for buffer operation.

Set the GTBER register except the BD[3:0] bits while the GTCNT counter is stopped.

BD[0] Bit (GTCCRA/GTCCRB Registers Buffer Operation Disable)

This bit disables buffer operation using the GTCCRA, GTCCRC, and GTCCRD registers together and buffer operation using the GTCCRB, GTCCRE, and GTCCRF registers together.

Even though the BD[0] bit is set to 0, buffer operation in the GTCCRB register is not performed if the GTDTCR.TDE bit is set to 1 in sawtooth-wave one-shot pulse mode or triangle-wave PWM mode. Instead, the compare match value for negative-phase waveform with dead time is set automatically.

A value for the BD[0] bit in the channel related to the position of the bit written with 1 by the GTSECSR register can be set when 1 is written to the SBDCE or SBDCE bit in the GTSECR register.

When the DBRTEC_m ($m = A, B$) bit is 1, setting the BD[0] bit to 1 while the mode of operation is sawtooth-wave one-shot pulse mode, triangle-wave PWM mode 3, or double buffer operation with triangle-wave PWM mode 2 results in transfer from the intermediate buffer to the GTCCR_m register.

BD[1] Bit (GTPR Register Buffer Operation Disable)

This bit disables buffer operation using the GTPR, GTPBR, and GTPDBR registers together.

A value for the BD[1] bit in the channel related to the position of the bit written with 1 by the GTSECSR register can be set when 1 is written to the SBDPE or SBDPE bit in the GTSECR register.

BD[2] Bit (GTADTRA/GTADTRB Registers Buffer Operation Disable)

This bit disables buffer operation using the GTADTRA, GTADTBRA, and GTADTDBRA registers together and buffer operation using the GTADTRB, GTADTBRB, and GTADTDBRB registers together.

A value for the BD[2] bit in the channel related to the position of the bit written with 1 by the GTSECSR register can be set when 1 is written to the SBDAB or SBDAB bit in the GTSECR register.

When the DBRTEC_m ($m = A, B$) bit is 1, triangle-wave is used, the GTADTR_m register is operated in a double buffer, and the buffer transfer timing is set to both crest and trough (GTBER.ADTT_m[1:0] bits are 11b). Even if the BD[2] bit is set to 1, the buffer operation from the intermediate buffer to the GTADTR_m register is performed.

BD[3] Bit (GTDVU/GTDVD Registers Buffer Operation Disable)

In sawtooth-wave PWM mode, sawtooth-wave one-shot pulse mode or triangle-wave PWM mode, this bit disables buffer operation using the GTDVU and GTDBU registers together and buffer operation using the GTDVD and GTDBD registers together.

Even though the BD[3] bit is set to 0, buffer operation in the GTDVD register is not performed if the GTDTCR.TDFER bit is set to 1. Instead, the value in the GTDVU register is set automatically.

A value for the BD[3] bit in the channel related to the position of the bit written with 1 by the GTSECSR register can be set when 1 is written to the SBDDE or SBDDE bit in the GTSECR register.

DBRTEC_m Bit (GTCCR_m Register Double Buffer Repeat Operation Enable) ($m = A, B$)

This setting enables the operation to repeat a transfer to the GTCCR_m register from the intermediate buffer by cycle during the buffer transfer disable period when performing the double buffer operation using the GTCCR_m register. It is valid in sawtooth-wave one-shot pulse mode, triangle-wave PWM mode 3, or triangle wave PWM mode 2 that double buffers the GTCCR_m register (GTBER.CCR_m[1] bit is 1).

The disabling period of buffer transfer indicates the period that buffer transfer is stopped by setting of the BD[0] bit (CPU writing or simultaneous buffer operation control by the GTSECSR register).

When the DBRTEC_m bit is 1, writing by the CPU to the GTCCR_m register sets the same value in temporary register x ($x = C, E$). The value of the GTCCRx ($x = C, E$) register is also transferred to the temporary register x ($x = C, E$) by forcible buffer transfer.

DBRTSC_m Bit (GTCCR_m Register Double Buffer Repeat Operation Period Select) ($m = A, B$)

When GTCCR_m double buffer repeat operation is permitted, select the period for which the repeat operation is to be performed. This is valid for triangular wave PWM mode 2.

When changing the prohibition/permission of buffer transfer during the counting operation of the GTCNT counter, change it in the same period as the period set by the DBRTSCm bit.

DBRTEADm Bit (GTADTRm Register Double Buffer Repeat Operation Enable) (m = A, B)

When operating the GTADTRm register in a double buffer operation, the operation of repeating the transfer from the intermediate buffer to the GTADTRm register in periodic units is permitted during the period when buffer transfer is prohibited.

Using the triangular wave, the GTADTRm register is double-buffered (GTBER.ADTDm bit is 1), and the buffer transfer timing is both crest and trough (GTBER.ADTTm[1:0] bits are 11b). It is effective in the case.

The buffer transfer prohibition period is set for the BD[2] bits (CPU write or buffer operation control by the GTSECSR register).

When the ADTDm bit is 1 and the DBRTEADm bit is 1, when CPU writing is performed to the GTADTRm register, the same value is also written to the temporary register ADm (m = A, B).

DBRTSADm Bit (GTADTRA Register Double Buffer Repeat Operation Period Select) (m = A, B)

When GTADTRm double buffer repeat operation is permitted, select the period for which repeat operation is to be performed.

When changing disable or enable during the counting operation of the GTCNT counter, change it in the same cycle as the cycle set in the DBRTSADm bit.

CCRA[1:0] Bits (GTCCRA Register Buffer Operation)

These bits set buffer operation with the GTCCRA, GTCCRC, and GTCCRD registers combined. When buffer operation is restricted by the operating mode set in the GTCR register, the GTCR register setting is given priority.*¹

CCRB[1:0] Bits (GTCCRB Register Buffer Operation)

These bits set buffer operation with the GTCCRB, GTCCRE, and GTCCRF registers combined. When buffer operation is restricted by the operating mode set in the GTCR register, the GTCR register setting is given priority.*¹

PR[1:0] Bits (GTPR Register Buffer Operation)

These bits set buffer operation with the GTPR, GTPBR, and GTPDBR registers combined.

CCRSWT Bit (GTCCRA and GTCCRB Registers Forcible Buffer Operation)

Writing 1 to the CCRSWT bit forcibly performs buffer transfer of the GTCCRA and GTCCRB registers. This bit automatically returns to 0 after the writing of 1. This bit is read as 0.

This bit is valid only when counting is stopped with compare match operation specified.

ADTTA[1:0] Bits (GTADTRA Register Buffer Transfer Timing Select)

These bits set the transfer timing for buffer operation of the GTADTRA, GTADTBRA, and GTADTDBRA registers.

ADTDA Bit (GTADTRA Register Double Buffer Operation)

These bits set buffer operation with the GTADTRA, GTADTBRA, and GTADTDBRA registers combined.

ADTTB[1:0] Bits (GTADTRB Register Buffer Transfer Timing Select)

These bits set the transfer timing for buffer operation of the GTADTRB, GTADTBRB, and GTADTDBRB registers.

ADTDB Bit (GTADTRB Register Double Buffer Operation)

These bits set buffer operation with the GTADTRA, GTADTBRA, and GTADTDBRA registers combined.

Note 1. The buffer operation mode is fixed in sawtooth-wave one-shot pulse mode, triangle-wave PWM mode 3 (64-bit transfer at trough).

21.2.16 General PWM Timer Interrupt and A/D Conversion Start Request Skipping Setting Register (GTITC)

Address(es): GPTW0.GTITC 0009 6044h, GPTW1.GTITC 0009 6144h, GPTW2.GTITC 0009 6244h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	ADTBL	—	ADTAL	—	IVTT[2:0]		IVTC[1:0]		ITLF	ITLE	ITLD	ITLC	ITLB	ITLA	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ITLA	GTCCRA Register Compare Match/Input Capture Interrupt Link	0: Not linked with GTCIV/GTCIU interrupt skipping function. 1: Linked with GTCIV/GTCIU interrupt skipping function.	R/W
b1	ITLB	GTCCRB Register Compare Match/Input Capture Interrupt Link	0: Not linked with GTCIV/GTCIU interrupt skipping function. 1: Linked with GTCIV/GTCIU interrupt skipping function.	R/W
b2	ITLC	GTCCRC Register Compare Match Interrupt Link	0: Not linked with GTCIV/GTCIU interrupt skipping function. 1: Linked with GTCIV/GTCIU interrupt skipping function.	R/W
b3	ITLD	GTCCRD Register Compare Match Interrupt Link	0: Not linked with GTCIV/GTCIU interrupt skipping function. 1: Linked with GTCIV/GTCIU interrupt skipping function.	R/W
b4	ITLE	GTCCRE Register Compare Match Interrupt Link	0: Not linked with GTCIV/GTCIU interrupt skipping function. 1: Linked with GTCIV/GTCIU interrupt skipping function.	R/W
b5	ITLF	GTCCRF Register Compare Match Interrupt Link	0: Not linked with GTCIV/GTCIU interrupt skipping function. 1: Linked with GTCIV/GTCIU interrupt skipping function.	R/W
b7, b6	IVTC[1:0]	GTCIV/GTCIU Interrupt Skipping Function Select	b7 b6 0 0: Skipping is not performed 0 1: Both overflow and underflow for sawtooth waves and crest for triangle waves are counted and skipped. 1 0: Both overflow and underflow for sawtooth waves and trough for triangle waves are counted and skipped. 1 1: Both overflow and underflow for sawtooth waves and both crest and trough for triangle waves are counted and skipped	R/W
b10 to b8	IVTT[2:0]	GTCIV/GTCIU Interrupt Skipping Count Select	b10 b8 0 0 0: Skipping is not performed 0 0 1: Skipping count of 1 0 1 0: Skipping count of 2 0 1 1: Skipping count of 3 1 0 0: Skipping count of 4 1 0 1: Skipping count of 5 1 1 0: Skipping count of 6 1 1 1: Skipping count of 7	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b12	ADTAL	GTADTRA Register A/D Conversion Start Request Link	0: Not linked with GTCIV/GTCIU interrupt skipping function. 1: Linked with GTCIV/GTCIU interrupt skipping function.	R/W
b13	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b14	ADTBL	GTADTRB Register A/D Conversion Start Request Link	0: Not linked with GTCIV/GTCIU interrupt skipping function. 1: Linked with GTCIV/GTCIU interrupt skipping function.	R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GTITC sets the skipping function for the GTCNT counter overflow (GTPR compare match) interrupt (GTCIV)/ underflow interrupt (GTCIU) and also sets whether to link the other interrupts and A/D conversion start requests with the GTCIV/GTCIU interrupt skipping function. Note that detection of disabling of output to the POEG and dead-time error interrupts cannot be linked with the GTCIV or GTCIU interrupt skipping function. Additionally, if the interrupt skipping function is performed, the change in the status flag is also skipped.

ITLA Bit (GTCCRA Register Compare Match/Input Capture Interrupt Link)

This bit specifies whether to link the GTCCRA compare match/input capture interrupt (GTCIA) with the GTCIV/GTCIU interrupt skipping function.

ITLB Bit (GTCCRB Register Compare Match/Input Capture Interrupt Link)

This bit specifies whether to link the GTCCRB compare match/input capture interrupt (GTCIB) with the GTCIV/GTCIU interrupt skipping function.

ITLC Bit (GTCCRC Register Compare Match Interrupt Link)

This bit specifies whether to link the GTCCRC compare match interrupt (GTCIC) with the GTCIV/GTCIU interrupt skipping function.

ITLD Bit (GTCCRD Register Compare Match Interrupt Link)

This bit specifies whether to link the GTCCRD compare match interrupt (GTCID) with the GTCIV/GTCIU interrupt skipping function.

ITLE Bit (GTCCRE Register Compare Match Interrupt Link)

This bit specifies whether to link the GTCCRE compare match interrupt (GTCIE) with the GTCIV/GTCIU interrupt skipping function.

ITLF Bit (GTCCRF Register Compare Match Interrupt Link)

This bit specifies whether to link the GTCCRF compare match interrupt (GTCIF) with the GTCIV/GTCIU interrupt skipping function.

IVTC[1:0] Bits (GTCIV/GTCIU Interrupt Skipping Function Select)

These bits set the skipping function for the GTPR compare match (GTCNT counter overflow) interrupt (GTCIV)/ GTCNT counter underflow interrupt (GTCIU).

IVTT[2:0] Bits (GTCIV/GTCIU Interrupt Skipping Count Select)

These bits set the skipping count for the GTPR compare match (GTCNT counter overflow) interrupt (GTCIV)/GTCNT counter underflow interrupt (GTCIU).

When modifying the IVTT[2:0] bits, first set the IVTC[1:0] bits to 00b.

ADTAL Bit (GTADTRA Register A/D Conversion Start Request Link)

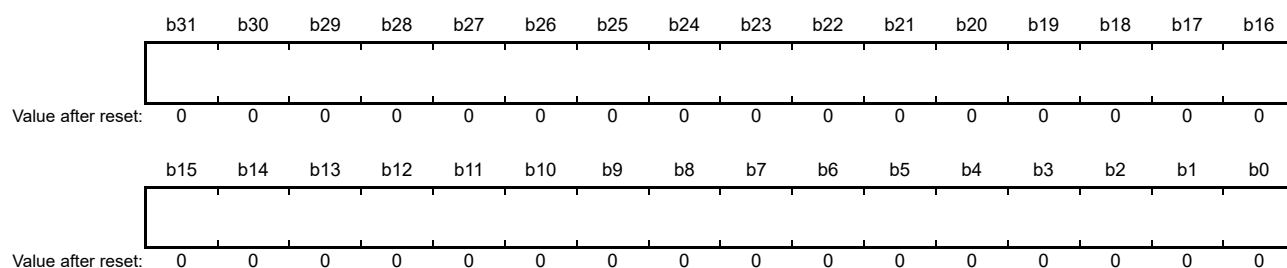
This bit specifies whether to link the A/D conversion start request, which is generated in response to a compare match with the GTCNT counter and the GTADTRA register, with the GTCIV/GTCIU interrupt skipping function.

ADTBL Bit (GTADTRB Register A/D Conversion Start Request Link)

This bit specifies whether to link the A/D conversion start request, which is generated in response to a compare match with the GTCNT counter and the GTADTRB register, with the GTCIV/GTCIU interrupt skipping function.

21.2.17 General PWM Timer Counter (GTCNT)

Address(es): GPTW0.GTCNT 0009 6048h, GPTW1.GTCNT 0009 6148h, GPTW2.GTCNT 0009 6248h



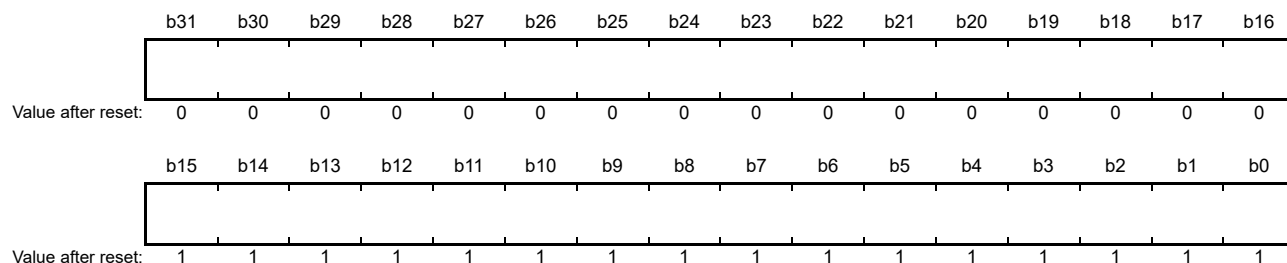
The GTCNT counter is a 32-bit readable/writable counter. There is one GTCNT counter for each channel. Writing is available only when counting is stopped, and thus writing is disabled while counting is in progress (when CST = 1). The upper 16 bits are reserved, and these bits are read as 0.

Access in 8-bit or 16-bit units to the GTCNT counter is prohibited, and it should be accessed in 32-bit units.

Set the range of the GTCNT counter within the range of $0 \leq \text{GTCNT counter} \leq \text{GTPR register}$.

21.2.18 General PWM Timer Compare Capture Register m (GTCCRm) (m = A to F)

Address(es): GPTW0.GTCCRA 0009 604Ch, GPTW1.GTCCRA 0009 614Ch, GPTW2.GTCCRA 0009 624Ch,
 GPTW0.GTCCRB 0009 6050h, GPTW1.GTCCRB 0009 6150h, GPTW2.GTCCRB 0009 6250h,
 GPTW0.GTCCRC 0009 6054h, GPTW1.GTCCRC 0009 6154h, GPTW2.GTCCRC 0009 6254h,
 GPTW0.GTCCRE 0009 6058h, GPTW1.GTCCRE 0009 6158h, GPTW2.GTCCRE 0009 6258h,
 GPTW0.GTCCRD 0009 605Ch, GPTW1.GTCCRD 0009 615Ch, GPTW2.GTCCRD 0009 625Ch,
 GPTW0.GTCCRF 0009 6060h, GPTW1.GTCCRF 0009 6160h, GPTW2.GTCCRF 0009 6260h



The GTCCRm register is a 32-bit readable/writable register. The upper 16 bits are reserved, and these bits are read as 0. Access in 8-bit or 16-bit units to the GTCCRm register is prohibited, and it should be accessed in 32-bit units.

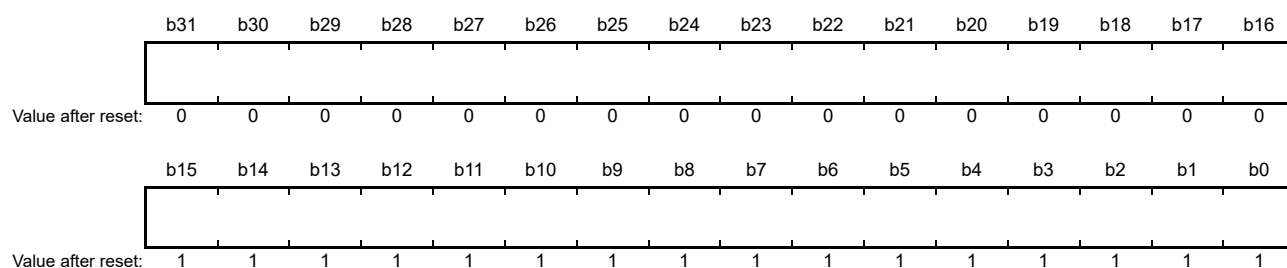
The GTCCRA and GTCCRB registers are registers used for both output compare and input capture.

The GTCCRC and GTCCRE registers are compare match registers, and can also function as buffer registers for the GTCCRA and GTCCRB registers.

The GTCCRD and GTCCRF registers are compare match registers, and can also function as buffer registers for the GTCCRC and GTCCRE registers (double buffer registers for the GTCCRA and GTCCRB registers).

21.2.19 General PWM Timer Period Setting Register (GTPR)

Address(es): GPTW0.GTPR 0009 6064h, GPTW1.GTPR 0009 6164h, GPTW2.GTPR 0009 6264h



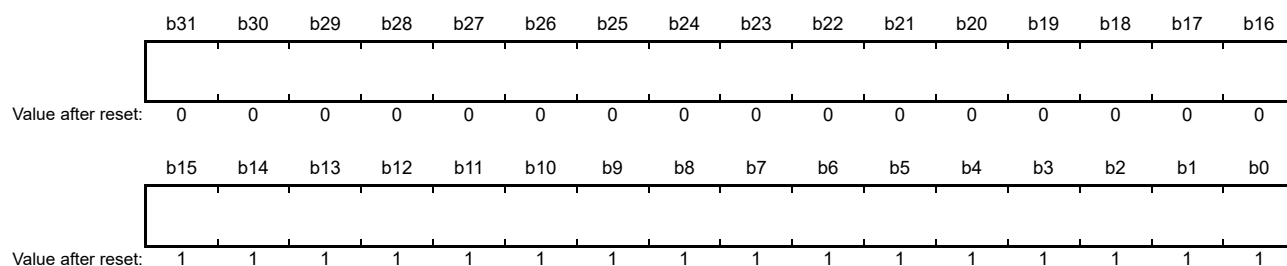
The GTPR register is a 32-bit readable/writable register that sets the maximum count value of the GTCNT counter. The upper 16 bits are reserved, and these bits are read as 0.

Access in 8-bit or 16-bit units to the GTPR register is prohibited, and it should be accessed in 32-bit units.

In sawtooth-wave mode, the value of (GTPR register + 1) is the count period. In triangle-wave mode, the value of (GTPR register value × 2) is the count period.

21.2.20 General PWM Timer Period Setting Buffer Register (GTPBR)

Address(es): GPTW0.GTPBR 0009 6068h, GPTW1.GTPBR 0009 6168h, GPTW2.GTPBR 0009 6268h

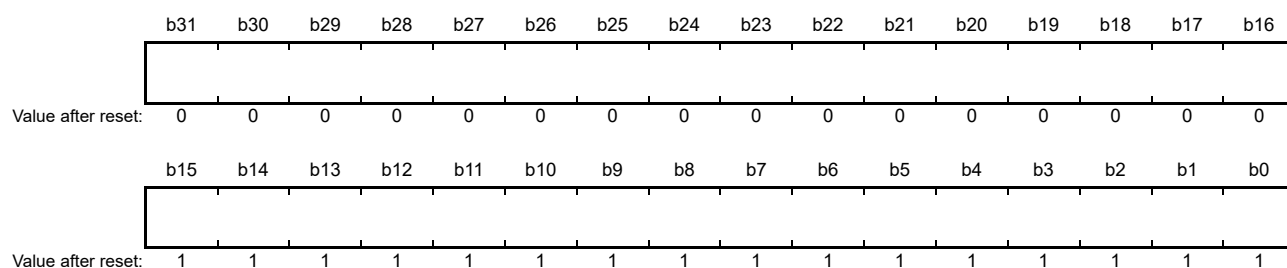


The GTPBR register is a 32-bit readable/writable register that functions as a buffer register for the GTPR register. The upper 16 bits are reserved, and these bits are read as 0.

Access in 8-bit or 16-bit units to the GTPBR register is prohibited, and it should be accessed in 32-bit units.

21.2.21 General PWM Timer Period Setting Double-Buffer Register (GTPDBR)

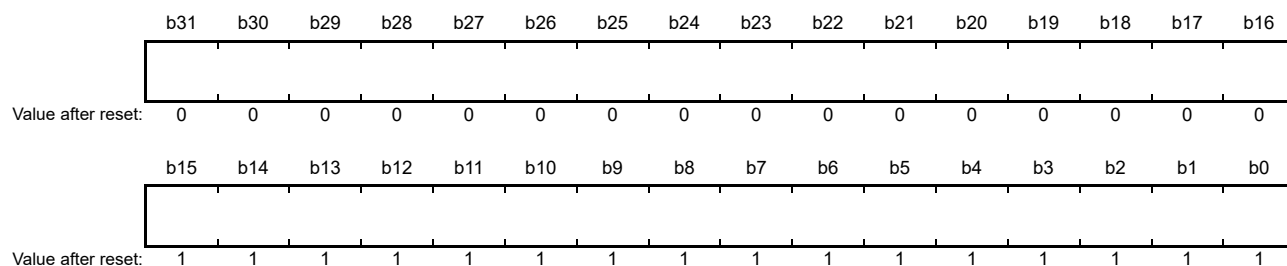
Address(es): GPTW0.GTPDBR 0009 606Ch, GPTW1.GTPDBR 0009 616Ch, GPTW2.GTPDBR 0009 626Ch



The GTPDBR register is a 32-bit readable/writable register that functions as a buffer register for the GTPBR register (a double buffer register for the GTPR register). The upper 16 bits are reserved, and these bits are read as 0. Access in 8-bit or 16-bit units to the GTPDBR register is prohibited, and it should be accessed in 32-bit units.

21.2.22 A/D Conversion Start Request Timing Register m (GTADTRm) (m = A, B)

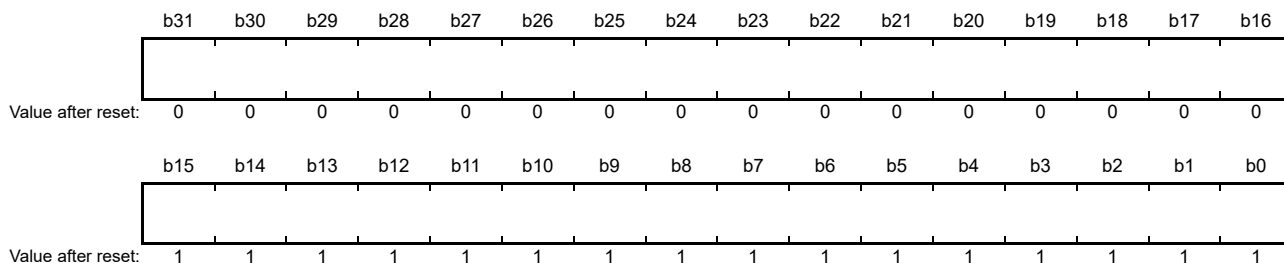
Address(es): GPTW0.GTADTRA 0009 6070h, GPTW1.GTADTRA 0009 6170h, GPTW2.GTADTRA 0009 6270h, GPTW0.GTADTRB 0009 607Ch, GPTW1.GTADTRB 0009 617Ch, GPTW2.GTADTRB 0009 627Ch



The GTADTRm register is 32-bit readable/writable register that set the timing of A/D conversion start request generation. The upper 16 bits are reserved, and these bits are read as 0. Access in 8-bit or 16-bit units to the GTADTRm register is prohibited, and it should be accessed in 32-bit units. When the GTADTRm register value matches the GTCNT counter value, an A/D conversion start request is generated.

21.2.23 A/D Conversion Start Request Timing Buffer Register m (GTADTBm) (m = A, B)

Address(es): GPTW0.GTADTBRA 0009 6074h, GPTW1.GTADTBRA 0009 6174h, GPTW2.GTADTBRA 0009 6274h,
GPTW0.GTADTBRB 0009 6080h, GPTW1.GTADTBRB 0009 6180h, GPTW2.GTADTBRB 0009 6280h

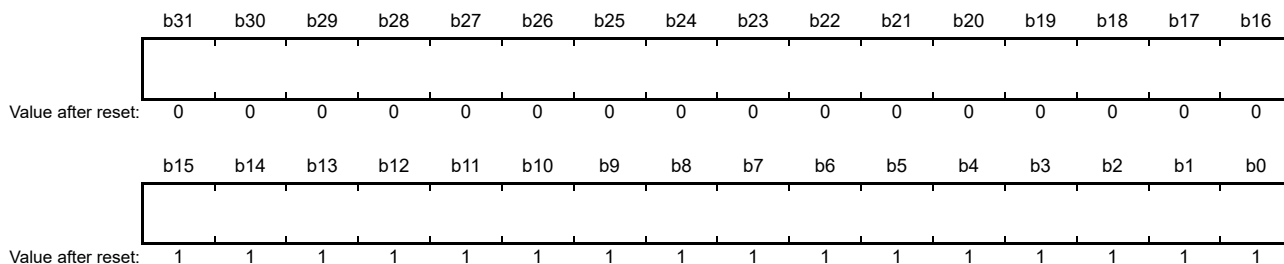


The GTADTBm register is 32-bit readable/writable register that function as buffer register for the GTADTRm register. The upper 16 bits are reserved, and these bits are read as 0.

Access in 8-bit or 16-bit units to the GTADTBm register is prohibited, and it should be accessed in 32-bit units.

21.2.24 A/D Conversion Start Request Timing Double-Buffer Register m (GTADTDBRm) (m = A, B)

Address(es): GPTW0.GTADTDBRA 0009 6078h, GPTW1.GTADTDBRA 0009 6178h, GPTW2.GTADTDBRA 0009 6278h,
GPTW0.GTADTDBRB 0009 6084h, GPTW1.GTADTDBRB 0009 6184h, GPTW2.GTADTDBRB 0009 6284h



The GTADTDBRm register is 32-bit readable/writable register that function as buffer register for the GTADTBm register (double buffer register for the GTADTRm register). The upper 16 bits are reserved, and these bits are read as 0. Access in 8-bit or 16-bit units to the GTADTDBRm register is prohibited, and it should be accessed in 32-bit units.

21.2.25 General PWM Timer Dead Time Control Register (GTDTCR)

Address(es): GPTW0.GTDTCR 0009 6088h, GPTW1.GTDTCR 0009 6188h, GPTW2.GTDTCR 0009 6288h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDFER	—	—	TDBDE	TDBUE	—	—	—	TDE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TDE	Negative-Phase Waveform Setting	0: GTCCRB register is set without using GTDVU and GTDVD registers. 1: GTDVU and GTDVD registers are used to set the compare match value for negative-phase waveform with dead time automatically in GTCCRB register.	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	TDBUE	GTDVU Register Buffer Operation Enable	0: GTDVU register buffer operation is disabled 1: GTDVU register buffer operation is enabled	R/W
b5	TDBDE	GTDVD Register Buffer Operation Enable	0: GTDVD register buffer operation is disabled 1: GTDVD register buffer operation is enabled	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	TDFER	GTDVD Register Setting	0: GTDVU and GTDVD registers are set separately. 1: The value written to GTDVU register is automatically set to GTDVD register.	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTDTCR register enables automatic setting of a compare match value for negative-phase waveform with dead time.

TDE Bit (Negative-Phase Waveform Setting)

This bit sets whether to use the GTDVU and GTDVD registers. When the GTDVU and GTDVD registers are used, the compare match value for a negative-phase waveform with dead time that was obtained by the compare match value of a positive-phase waveform (GTCCRA register) and the dead time value (GTDVU and GTDVD registers) is automatically set in the GTCCRB register.

The TDE bit setting is ignored in sawtooth-wave PWM mode, and the GTCCRB register is not automatic setting.

The automatically set GTCCRB register value has the following upper and lower limit values.

- Triangle wave PWM mode
Upper limit value: the value set in the GTPR register – 1
Lower limit value: 0000 0001h in up-counting, 0000 0000h in down-counting
- Sawtooth-wave one-shot pulse mode
Upper limit value: the value set in the GTPR register
Lower limit value: 0000 0000h

If the obtained GTCCRB register value is not within the range between the upper and lower limits, the upper or lower limit is set in the GTCCRB register, and the GTST.DTEF flag becomes 1. However, if the obtained GTCCRB register value exceeds the upper limit in triangle-wave PWM mode, the DTEF flag becomes 0.

TDBUE Bit (GTDVU Register Buffer Operation Enable)

This bit enables buffer operation with the GTDVU and GTDBU registers combined.

The timing of buffer transfer is at troughs in triangle-wave mode, and at overflows or underflows in sawtooth-wave mode.

TDBDE Bit (GTDVD Register Buffer Operation Enable)

This bit enables buffer operation with the GTDVD and GTDBD registers combined.

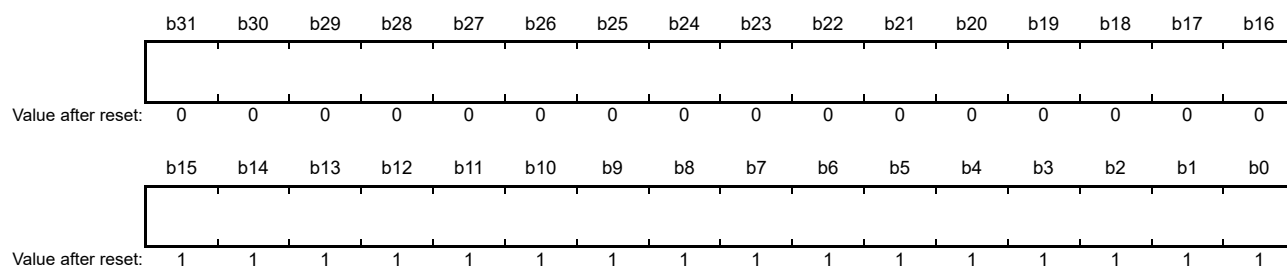
The buffer transfer timing is the trough in triangle-wave mode, and at an overflow or underflow in sawtooth-wave mode. When this bit and the TDFER bit are set to 1 simultaneously, the TDFER bit setting is given priority.

TDFER Bit (GTDVD Register Setting)

This bit sets whether or not the value written to the GTDVU register is also set to the GTDVD register automatically.

21.2.26 General PWM Timer Dead Time Value Register m (GTDVm) (m = U, D)

Address(es): GPTW0.GTDVU 0009 608Ch, GPTW1.GTDVU 0009 618Ch, GPTW2.GTDVU 0009 628Ch,
GPTW0.GTDVD 0009 6090h, GPTW1.GTDVD 0009 6190h, GPTW2.GTDVD 0009 6290h



The GTDm register is 32-bit readable/writable register that set the dead time for generating PWM waveforms with dead time. The upper 16 bits are reserved, and these bits are read as 0.

Access in 8-bit or 16-bit units to the GTDm register is prohibited, and it should be accessed in 32-bit units.

The GTDVU register is used for up-counting in triangle-wave mode. The GTDVD register is used for down-counting.

In the case of a sawtooth-waves, the GTDVU register controls the front dead time and the GTDVD register controls the rear dead time, regardless of whether the count is up or down.

Do not set this register to a value equal to or larger than that of the GTPR register.

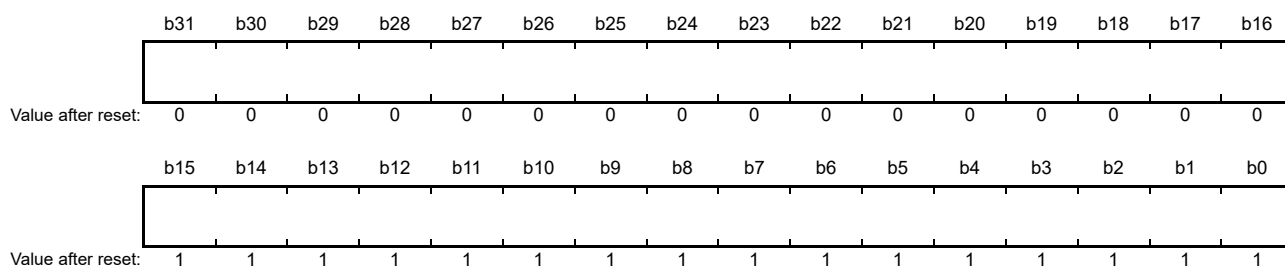
- $\text{GTDVU} > 0$
- $\text{GTDVU} < \text{GTPR}/2$
- $\text{GTDVU} + \text{GTPR} \leq \text{FFFF FFFFh}$

In addition, when using the automatic dead time setting function, do not set a value that makes a change point of the waveform exceeding the count period. The change point of the negative-phase waveform, which is automatically calculated, is obtained by reading the GTCCRB register. When using the GTDm register, writing to the GTCCRB register is prohibited. Setting this register to 0 leads to the output of waveforms without dead time.

When the GTDTCR.TDFER bit is 1, the writing to the GTDVD register has no effect. At this time, when the GTDVD register is read, the value for the GTDVU register is read.

21.2.27 General PWM Timer Dead Time Value Buffer Register m (GTDBm) (m = U, D)

Address(es): GPTW0.GTDBU 0009 6094h, GPTW1.GTDBU 0009 6194h, GPTW2.GTDBU 0009 6294h,
GPTW0.GTDBD 0009 6098h, GPTW1.GTDBD 0009 6198h, GPTW2.GTDBD 0009 6298h



The GTDBm register is 32-bit readable/writable register that function as buffer register for the GTDVM register. The upper 16 bits are reserved, and these bits are read as 0.

Access in 8-bit or 16-bit units to the GTDBm register is prohibited, and it should be accessed in 32-bit units.

21.2.28 General PWM Timer Output Protection Function Status Register (GTSOS)

Address(es): GPTW0.GTSOS 0009 609Ch, GPTW1.GTSOS 0009 619Ch, GPTW2.GTSOS 0009 629Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOS[1:0]	—
Value after reset:	0	0	0	0	0	0	x	x	0	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b1, b0	SOS[1:0]	Output Protection Function Status	b1 b0 0 0: Normal operation 0 1: Protected state (GTCCRA = 0 is set during transfer at trough or crest) 1 0: Protected state (GTCCRA ≥ GTPR is set during transfer at trough) 1 1: Protected state (GTCCRA ≥ GTPR is set during transfer at crest)	R
b7 to b2	—	Reserved	These bits are read as 0.	R
b9, b8	—	Reserved	The read value is undefined.	R
b31 to b10	—	Reserved	These bits are read as 0.	R

The GTSOS register is a status register that indicates the status of the output protection function. The output protection function is enabled only when the dead time is automatically set (the GTDTCR.TDE bit is 1) in triangle-wave mode.

SOS[1:0] Bits (Output Protection Function Status)

This bit indicates the status of the output protection function in triangle-wave PWM mode. For details of the output protection function, see section 21.7.4, Output Protection Function for GTIOChm Pin Output (n = 0 to 2; m = A, B).

21.2.29 General PWM Timer Output Protection Function Temporary Release Register (GTSOTR)

Address(es): GPTW0.GTSOTR 0009 60A0h, GPTW1.GTSOTR 0009 61A0h, GPTW2.GTSOTR 0009 62A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOTR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SOTR	Output Protection Function Temporary Release	0: Protected state is not released 1: Protected state is released	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTSOTR register temporarily releases the protected state of GTIOCnB pin output (n = 0 to 2) when output protection has been set.

The protected state can be released only for the case of the GTSOS.SOS[1:0] bits are 10b (protected state in which GTCCRA register $\geq$ GTPR register has occurred during transfer at trough). The protected state cannot be released for any other case.

SOTR Bit (Output Protection Function Temporary Release)

This bit sets whether to temporarily release the protected state of the GTIOCnB pin output in an output protected state. After the SOTR bit has been set to 1, the output protection function is canceled from the first trough. After the SOTR bit has been set to 0, output protection is resumed from the first trough.

21.2.30 General PWM Timer A/D Conversion Start Request Signal Monitoring Register (GTADSMR)

Address(es): GPTW0.GTADSMR 0009 60A4h, GPTW1.GTADSMR 0009 61A4h, GPTW2.GTADSMR 0009 62A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	ADSMEN1	—	—	—	—	—	—	ADSMS1[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ADSMEN0	—	—	—	—	—	—	ADSMS0[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	ADSMS0[1:0]	A/D Conversion Start Request Signal Monitor 0 Selection	b1 b0 0 0: A/D conversion start request signal generated by the GTADTRA register during up-counting 0 1: A/D conversion start request signal generated by the GTADTRA register during down-counting 1 0: A/D conversion start request signal generated by the GTADTRB register during up-counting 1 1: A/D conversion start request signal generated by the GTADTRB register during down-counting	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	ADSMEN0	A/D Conversion Start Request Signal Monitor 0 Output Enabling	0: Output of A/D conversion start request signal monitor 0 is disabled. 1: Output of A/D conversion start request signal monitor 0 is enabled.	R/W
b15 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b17, b16	ADSMS1[1:0]	A/D Conversion Start Request Signal Monitor 1 Selection	b17 b16 0 0: A/D conversion start request signal generated by the GTADTRA register during up-counting 0 1: A/D conversion start request signal generated by the GTADTRA register during down-counting 1 0: A/D conversion start request signal generated by the GTADTRB register during up-counting 1 1: A/D conversion start request signal generated by the GTADTRB register during down-counting	R/W
b23 to b18	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b24	ADSMEN1	A/D Conversion Start Request Signal Monitor 1 Output Enabling	0: Output of A/D conversion start request signal monitor 1 is disabled. 1: Output of A/D conversion start request signal monitor 1 is enabled.	R/W
b31 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

This register is used to control monitors for the A/D conversion start request signal that is synchronized with a frame.

ADSMsk[1:0] Bits (A/D Conversion Start Request Signal Monitor k Selection) (k = 0, 1)

These bits are used to select A/D conversion start request signal synchronized with a frame which is monitored by the GTADSMk pin.

In triangle-wave PWM mode, the following settings are prohibited:

- Set the ADSMSk[1:0] bits to 00b (A/D conversion start request during up-counting) when GTADTRA = 0
- Set the ADSMSk[1:0] bits to 10b (A/D conversion start request during up-counting) when GTADTRB = 0
- Set the ADSMSk[1:0] bits to 01b (A/D conversion start request during down-counting) when GTADTRA = GTPR
- Set the ADSMSk[1:0] bits to 11b (A/D conversion start request during down-counting) when GTADTRB = GTPR

ADSMENk Bit (A/D Conversion Start Request Signal Monitor k Output Enabling) (k = 0, 1)

This bit enables or disables the monitor output to the GTADSMk pin.

When the output is disabled, the GTADSMk pin goes to the low level.

When the bit is 1, the signal on the GTADSMk pin goes to the high level on assertion of the signal to request to the start of A/D conversion selected by the ADSMSk[1:0] bits and returns to the low level at the end of the current cycle of the timer for the channel that generated the given signal to request the start of A/D conversion. When the counter stops, the value when the counter stopped is retained for output. Set the ADSMENk bit to 0 to output the low level.

When a signal to request the start of A/D conversion is generated at the end of a timer period, the generation of this signal has priority in terms of monitoring output and the output remains at the high level till the end of the next period.

When the output of the same A/D conversion start request signal monitoring output is enabled for multiple channels, ORed signals will be output from the GPTW.

21.2.31 General PWM Timer Operation Enable Bit Simultaneous Control Channel Select Register (GTSECSR)

Address(es): GPTW0.GTSECSR 0009 60D0h, GPTW1.GTSECSR 0009 61D0h, GPTW2.GTSECSR 0009 62D0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	SECSEL2	SECSEL1	SECSEL0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SECSEL0	Channel 0 Operation Enable Bit Simultaneous Control Channel Select	0: Disable simultaneous control. 1: Enable simultaneous control.	R/W
b1	SECSEL1	Channel 1 Operation Enable Bit Simultaneous Control Channel Select		R/W
b2	SECSEL2	Channel 2 Operation Enable Bit Simultaneous Control Channel Select		R/W
b31 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTSECSR register selects an intended channel for updating an operation enable bit by the GTSECR register.

A bit position for the GTSECSR register indicates a channel number. The GTSECSR register of each channel is a common register, and writing 1 to a bit in the GTSECSR register in any channel and updating it changes a channel, related to the position of the bit written with 1 by the GTSECSR register, to be simultaneously controlled of the operation enable bit by the GTSECR register.

Access in 8-bit or 16-bit units to the GTSECSR register is prohibited, and it should be accessed in 32-bit units.

SECSELn Bit (Operation Enable Bit Simultaneous Control Channel Select) (n = 0 to 2)

This bit enables or disables the simultaneous control of operation enable in channel n.

When the bit is set to 1, the simultaneous control is enabled, and disabled when the bit is 0.

21.2.32 General PWM Timer Operation Enable Bit Simultaneous Control Register (GTSECR)

Address(es): GPTW0.GTSECR 0009 60D4h, GPTW1.GTSECR 0009 61D4h, GPTW2.GTSECR 0009 62D4h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	—	—	—	SSCD	—	—	—	—	—	—	—	SSCE	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	SBDDD	SBDAD	SBDPD	SBDCE	—	—	—	—	SBDDE	SBDPE	SBDCE	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SBDCE	GTCCR Register Buffer Operation Simultaneous Enable	0: Disable simultaneous enabling GTCCR buffer operations 1: Enable GTCCR register buffer operations simultaneously	R/W
b1	SBDPE	GTPR Register Buffer Operation Simultaneous Enable	0: Disable simultaneous enabling GTPR buffer operations 1: Enable GTPR register buffer operations simultaneously	R/W
b2	SBDPE	GTADTR Register Buffer Operation Simultaneous Enable	0: Disable simultaneous enabling GTADTR buffer operations 1: Enable GTADTR register buffer operations simultaneously	R/W
b3	SBDDE	GTDV Register Buffer Operation Simultaneous Enable	0: Disable simultaneous enabling GTDV buffer operations 1: Enable GTDV register buffer operations simultaneously	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	SBDCE	GTCCR Register Buffer Operation Simultaneous Disable	0: Disable simultaneous disabling GTCCR buffer operations 1: Disable GTCCR register buffer operations simultaneously	R/W
b9	SBDPD	GTPR Register Buffer Operation Simultaneous Disable	0: Disable simultaneous disabling GTPR buffer operations 1: Disable GTPR register buffer operations simultaneously	R/W
b10	SBDAD	GTADTR Register Buffer Operation Simultaneous Disable	0: Disable simultaneous disabling GTADTR buffer operations 1: Disable GTADTR register buffer operations simultaneously	R/W
b11	SBDDD	GTDV Register Buffer Operation Simultaneous Disable	0: Disable simultaneous disabling GTDV buffer operations 1: Disable GTDV register buffer operations simultaneously	R/W
b16 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b17	SSCE	Synchronous Set/Clear Simultaneous Enable	0: Disable simultaneous enabling synchronous set/clear 1: Enable synchronous set/clear simultaneously	R/W
b24 to b18	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b25	SSCD	Synchronous Set/Clear Simultaneous Disable	0: Disable simultaneous disabling synchronous set/clear 1: Disable synchronous set/clear simultaneously	R/W
b31 to b26	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTSECR register simultaneously updates the value for operation enable bits of a channel set by the GTSECSR register.

Writing 1 to a bit in the GTSECR register in any channel and updating it updates an operation enable bit for all channels, related to the position of the bit written with 1 by the all GTSECR registers. Setting enable and disable bits for the same operation enable bit to 1 in the GTSECR is prohibited.

A bit written to 1 is automatically cleared. When the GTSECR is read, 0 is read.

Access in 8-bit or 16-bit units to the GTSECR register is prohibited, and it should be accessed in 32-bit units.

SBDCE Bit (GTCCR Register Buffer Operation Simultaneous Enable)

When 1 is written to this bit, 0 is simultaneously set to a GTBER.BD[0] bit in the channels set to 1 by the GTSECSR register, and buffer operations using the GTCCRA, GTCCRC, and GTCCRD registers and using the GTCCRB, GTCCRE, and GTCCRF registers are enabled.

Simultaneous setting of SBDCE and SBDCD bits to 1 is prohibited.

SBDPE Bit (GTPR Register Buffer Operation Simultaneous Enable)

When 1 is written to this bit, 0 is simultaneously set to a GTBER.BD[1] bit in the channels set to 1 by the GTSECSR register, and buffer operations using the GTPR, GTPBR, and GTPDBR registers are enabled.

Simultaneous setting of SBDPE and SBDPD bits to 1 is prohibited.

SBD AE Bit (GTADTR Register Buffer Operation Simultaneous Enable)

When 1 is written to this bit, 0 is simultaneously set to a GTBER.BD[2] bit in the channels set to 1 by the GTSECSR register, and buffer operations using the GTADTRA, GTADTBRA, and GTADTDBRA registers and using the GTADTRB, GTADTBRB, and GTADTDBRB registers are enabled.

Simultaneous setting of SBD AE and SBDAD bits to 1 is prohibited.

SBDDE Bit (GTDV Register Buffer Operation Simultaneous Enable)

When 1 is written to this bit, 0 is simultaneously set to a GTBER.BD[3] bit in the channels set to 1 by the GTSECSR register, and buffer operations using the GTDVU and GTDBU registers and using the GTDVD and GTDBD registers are enabled.

Simultaneous setting of SBDDE and SBDDD bits to 1 is prohibited.

SBDCD Bit (GTCCR Register Buffer Operation Simultaneous Disable)

When 1 is written to this bit, 1 is simultaneously set to a GTBER.BD[0] bit in the channels set to 1 by the GTSECSR register, and buffer operations using the GTCCRA, GTCCRC, and GTCCRD registers and using the GTCCRB, GTCCRE, and GTCCRF registers are disabled.

Simultaneous setting of SBDCE and SBDCD bits to 1 is prohibited.

SBDPD Bit (GTPR Register Buffer Operation Simultaneous Disable)

When 1 is written to this bit, 1 is simultaneously set to a GTBER.BD[1] bit in the channels set to 1 by the GTSECSR register, and buffer operations using the GTPR, GTPBR, and GTPDBR registers are disabled.

Simultaneous setting of SBDPE and SBDPD bits to 1 is prohibited.

SBDAD Bit (GTADTR Register Buffer Operation Simultaneous Disable)

When 1 is written to this bit, 1 is simultaneously set to a GTBER.BD[2] bit in the channels set to 1 by the GTSECSR register, and buffer operations using the GTADTRA, GTADTBRA, and GTADTDBRA registers and using the GTADTRB, GTADTBRB, and GTADTDBRB registers are disabled.

Simultaneous setting of SBD AE and SBDAD bits to 1 is prohibited.

SBDDD Bit (GTDV Register Buffer Operation Simultaneous Disable)

When 1 is written to this bit, 1 is simultaneously set to a GTBER.BD[3] bit in the channels set to 1 by the GTSECSR register, and buffer operations using the GTDVU and GTDBU registers and using the GTDVD and GTDBD registers are disabled.

Simultaneous setting of SBDDE and SBDDD bits to 1 is prohibited.

SSCE Bit (Synchronous Set/Clear Simultaneous Enable)

Writing 1 to this bit simultaneously sets any GTCR.SSCEN bits to 1 for channels with simultaneous control enabled by the setting 1 in the GTSECSR register and enables the synchronous set/clear function.

Simultaneous setting of SSCE and SSCD bits to 1 is prohibited.

SSCD Bit (Synchronous Set/Clear Simultaneous Disable)

Writing 1 to this bit simultaneously sets any GTCR.SSCEN bits to 0 for channels with simultaneous control enabled by the setting 1 in the GTSECSR register and disables the synchronous set/clear function.

Simultaneous setting of SSCE and SSCD bits to 1 is prohibited.

21.2.33 General PWM Timer Buffer Enable Register 2 (GTBER2)

Address(es): GPTW0.GTBER2 0009 60E0h, GPTW1.GTBER2 0009 61E0h, GPTW2.GTBER2 0009 62E0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	CPTDV	CPTAD B	CPTAD A	CPTPR	CPTCB	CPTCA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	CMTAD B	CMTAD A	—	CMTCB[1:0]	CMTCA[1:0]	—	—	—	CCTDV	CCTAD B	CCTAD A	CCTPR	CCTCB	CCTCA	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CCTCA	Counter Clear Source GTCCRA Register Buffer Transfer Disable	0: Enable GTCCRA register buffer transfer by counter clear 1: Disable GTCCRA register buffer transfer by counter clear	R/W
b1	CCTCB	Counter Clear Source GTCCRB Register Buffer Transfer Disable	0: Enable GTCCRB register buffer transfer by counter clear 1: Disable GTCCRB register buffer transfer by counter clear	R/W
b2	CCTPR	Counter Clear Source GTPR Register Buffer Transfer Disable	0: Enable GTPR register buffer transfer by counter clear 1: Disable GTPR register buffer transfer by counter clear	R/W
b3	CCTADA	Counter Clear Source GTADTRA Register Buffer Transfer Disable	0: Enable GTADTRA register buffer transfer by counter clear 1: Disable GTADTRA register buffer transfer by counter clear	R/W
b4	CCTADB	Counter Clear Source GTADTRB Register Buffer Transfer Disable	0: Enable GTADTRB register buffer transfer by counter clear 1: Disable GTADTRB register buffer transfer by counter clear	R/W
b5	CCTDV	Counter Clear Source GTDVU/GTDVD Register Buffer Transfer Disable	0: Enable GTDVU/GTDVD register buffer transfer by counter clear 1: Disable GTDVU/GTDVD register buffer transfer by counter clear	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b9, b8	CMTCA[1:0]	Compare Match Source GTCCRA Register Buffer Transfer Enable	b9 b8 0 0: Disable GTCCRA register Buffer Transfer by compare match of GTCCRA register and GTCCRB register 0 1: Enable GTCCRA register Buffer Transfer by compare match of GTCCRA register 1 0: Enable GTCCRA register Buffer Transfer by compare match of GTCCRB register 1 1: Enable GTCCRA register Buffer Transfer by compare match of GTCCRA register and GTCCRB register	R/W
b11, b10	CMTCB[1:0]	Compare Match Source GTCCRB Register Buffer Transfer Enable	b11 b10 0 0: Disable GTCCRB register Buffer Transfer by compare match of GTCCRA register and GTCCRB register 0 1: Enable GTCCRB register Buffer Transfer by compare match of GTCCRA register 1 0: Enable GTCCRB register Buffer Transfer by compare match of GTCCRB register 1 1: Enable GTCCRB register Buffer Transfer by compare match of GTCCRA register and GTCCRB register	R/W
b12	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b13	CMTADA	Compare Match Source GTADTRA Register Buffer Transfer Enable	0: Disable GTADTRA register buffer transfer by compare match of GTADTRA register 1: Enable GTADTRA register buffer transfer by compare match of GTADTRA register	R/W

Bit	Symbol	Bit Name	Description	R/W
b14	CMTADB	Compare Match Source GTADTRB Register Buffer Transfer Enable	0: Disable GTADTRB register buffer transfer by compare match of GTADTRB register 1: Enable GTADTRB register buffer transfer by compare match of GTADTRB register	R/W
b15	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b16	CPTCA	Overflow/Underflow Source GTCCRA Register Buffer Transfer Disable	0: Enable GTCCRA register buffer transfer by overflow/underflow 1: Disable GTCCRA register buffer transfer by overflow/underflow	R/W
b17	CPTCB	Overflow/Underflow Source GTCCRB Register Buffer Transfer Disable	0: Enable GTCCRB register buffer transfer by overflow/underflow 1: Disable GTCCRB register buffer transfer by overflow/underflow	R/W
b18	CPTPR	Overflow/Underflow Source GTPR Register Buffer Transfer Disable	0: Enable GTPR register buffer transfer by overflow/underflow 1: Disable GTPR register buffer transfer by overflow/underflow	R/W
b19	CPTADA	Overflow/Underflow Source GTADTRA Register Buffer Transfer Disable	0: Enable GTADTRA register buffer transfer by overflow/underflow 1: Disable GTADTRA register buffer transfer by overflow/underflow	R/W
b20	CPTADB	Overflow/Underflow Source GTADTRB Register Buffer Transfer Disable	0: Enable GTADTRB register buffer transfer by overflow/underflow 1: Disable GTADTRB register buffer transfer by overflow/underflow	R/W
b21	CPTDV	Overflow/Underflow Source GTDVU/GTDVD Register Buffer Transfer Disable	0: Enable GTDVU/GTDVD register buffer transfer by overflow/underflow 1: Disable GTDVU/GTDVD register buffer transfer by overflow/underflow	R/W
b31 to b22	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The GTBER2 register makes settings for buffer operation.

CCTCm Bit (Counter Clear Source GTCCRm Register Buffer Transfer Disable) (m = A, B)

This bit disables buffer transfer by counter clear using the GTCCRm, GTCCRN, and GTCCRx registers together (m = A, B; n = C, E; x = D, F).

This bit is effective when the GTBER.BD[0] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.CCRm[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves. Even if the CCTCm bit is 0, buffer transfer is not performed by counter clear.

If there is a conflict with the CMTCm bit setting, the CCTCm bit setting has priority.

CCTPR Bit (Counter Clear Source GTPR Register Buffer Transfer Disable)

This bit disables buffer transfer by counter clear using the GTPR, GTPBR, and GTPDBR registers together.

This bit is effective when the GTBER.BD[1] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.PR[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves. Even if the CCTPR bit is 0, buffer transfer is not performed by counter clear.

CCTADm Bit (Counter Clear Source GTADTRm Register Buffer Transfer Disable) (m = A, B)

This bit disables buffer transfer by counter clear using the GTADTRm, GTADTBm and GTADTDBRm registers together.

This bit is effective when the GTBER.BD[2] bit is 0 (buffer operation enabled) and the buffer operation is selected by the ADTTm[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves. Even if the CCTADm bit is 0, buffer transfer is not performed by counter clear.

If there is a conflict with the CMTADm bit setting, the CCTADm bit setting has priority.

CCTDV Bit (Counter Clear Source GTDVU/GTDVD Register Buffer Transfer Disable)

This bit disables buffer transfer by counter clear using the GTDVU and GTDBU or GTDVD and GTDBD registers together.

This bit is effective when the GTBER.BD[3] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTDTCR.TDBUE or GTDTCR.TDBDE bit with sawtooth-waves.

The setting is invalid in triangle waves. Even if the CCTDV bit is 0, buffer transfer is not performed by counter clear.

CMTCA[1:0] Bits (Compare Match Source GTCCRA Register Buffer Transfer Enable)

These bits enable buffer transfer by compare match of GTCCRA using the GTCCRA, GTCCRC and GTCCRD registers together.

These bits are effective when the GTBER.BD[0] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.CCRA[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves.

If there is a conflict with the CCTCA bit setting, the CCTCA bit setting has priority.

CMTCB[1:0] Bits (Compare Match Source GTCCRB Register Buffer Transfer Enable)

These bits enable buffer transfer by compare match of GTCCRB using the GTCCRB, GTCCRE and GTCCRF registers together.

These bits are effective when the GTBER.BD[0] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.CCRB[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves.

If there is a conflict with the CCTCB bit setting, the CCTCB bit setting has priority.

CMTADm Bit (Compare Match Source GTADTRm Register Buffer Transfer Enable) (m = A, B)

This bit enables buffer transfer by compare match of the GTADTRm using the GTADTRm, GTADTBRm, and GTADTDBRm registers together.

This bit is effective when the GTBER.BD[2] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.ADTTm[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves.

If there is a conflict with the CCTADm bit setting, the CCTADm bit setting has priority.

CPTCA Bit (Overflow/Underflow Source GTCCRA Register Buffer Transfer Disable)

This bit disables buffer transfer by overflow/underflow in sawtooth-waves using the GTCCRA, GTCCRC and GTCCRD registers together.

This bit is effective when the CCTCA bit is 0 (GTCCRA register buffer transfer by counter clear is enabled), the GTBER.BD[0] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.CCRA[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves.

CPTCB Bit (Overflow/Underflow Source GTCCRB Register Buffer Transfer Disable)

This bit disables buffer transfer by overflow/underflow in sawtooth-waves using the GTCCRB, GTCCRE and GTCCRF registers together.

This bit is effective when the CCTCB bit is 0 (GTCCRB register buffer transfer by counter clear is enabled), the GTBER.BD[0] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.CCRB[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves.

CPTPR Bit (Overflow/Underflow Source GTPR Register Buffer Transfer Disable)

This bit disables buffer transfer by overflow/underflow in sawtooth-waves using the GTPR, GTPBR and GTPDBR

registers together.

This bit is effective when the CCTPR bit is 0 (GTPR register buffer transfer by counter clear is enabled), the GTBER.BD[1] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.PR[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves.

CPTADm Bit (Overflow/Underflow Source GTADTRm Register Buffer Transfer Disable) (m = A, B)

This bit disables buffer transfer by overflow/underflow in sawtooth-waves using the GTADTRm, GTADTBm, and GTADTDBRm registers together.

This bit is effective when the CCTADm bit is 0 (GTADTRm register buffer transfer by counter clear is enabled), the GTBER.BD[2] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTBER.ADTTm[1:0] bits with sawtooth-waves.

The setting is invalid in triangle waves.

CPTDV Bit (Overflow/Underflow Source GTDVU/GTDVD Register Buffer Transfer Disable)

This bit disables buffer transfer by overflow/underflow in sawtooth-waves using the GTDVU and GTDBU or GTDVD and GTDBD registers together.

This bit is effective when the CCTDV bit is 0 (GTDVU/GTDVD register buffer transfer by counter clear is enabled), the GTBER.BD[3] bit is 0 (buffer operation enabled) and the buffer operation is selected by the GTDTCR.TDBUE or GTDTCR.TDBDE bit with sawtooth-waves.

The setting is invalid in triangle waves.

21.2.34 General PWM Timer Inter Channel Cooperation Input Capture Control Register (GTICCR)

Address(es): GPTW0.GTICCR 0009 60ECh, GPTW1.GTICCR 0009 61ECh, GPTW2.GTICCR 0009 62ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ICBGRP[1:0]	—	—	—	—	—	—	ICBCLK	ICBFPU	ICBFPO	ICBFF	ICBFEE	ICBFED	ICBFEC	ICBFEB	ICBFEA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ICAGRP[1:0]	—	—	—	—	—	—	ICACK	ICAFFPU	ICAFFPO	ICAFF	ICAFFEE	ICAFFED	ICAFFEC	ICAFFEB	ICAFFEA
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ICAFA	Enable GTCCRA Register Compare Match/Input Capture as Source of Other Channel GTCCRA Input Capture	0: Disable GTCCRA register comparison match/input capture transfer, which is the GTCCRA input capture source of other channels 1: Enable GTCCRA register comparison match/input capture transfer, which is the GTCCRA input capture source of other channels	R/W
b1	ICAFB	Enable GTCCRB Register Compare Match/Input Capture as Source of Other Channel GTCCRA Input Capture	0: Disable GTCCRB register comparison match/input capture transfer, which is the GTCCRA input capture source of other channels 1: Enable GTCCRB register comparison match/input capture transfer, which is the GTCCRA input capture source of other channels	R/W
b2	ICAFC	Enable GTCCRC Register Compare Match as Source of Other Channel GTCCRA Input Capture	0: Disable GTCCRC register comparison match, which is the GTCCRA input capture source of other channels 1: Enable GTCCRC register comparison match, which is the GTCCRA input capture source of other channels	R/W
b3	ICAFD	Enable GTCCRD Register Compare Match as Source of Other Channel GTCCRA Input Capture	0: Disable GTCCRD register comparison match, which is the GTCCRA input capture source of other channels 1: Enable GTCCRD register comparison match, which is the GTCCRA input capture source of other channels	R/W
b4	ICAFF	Enable GTCCRE Register Compare Match as Source of Other Channel GTCCRA Input Capture	0: Disable GTCCRE register comparison match, which is the GTCCRA input capture source of other channels 1: Enable GTCCRE register comparison match, which is the GTCCRA input capture source of other channels	R/W
b5	ICAFF	Enable GTCCRF Register Compare Match as Source of Other Channel GTCCRA Input Capture	0: Disable GTCCRF register comparison match, which is the GTCCRA input capture source of other channels 1: Enable GTCCRF register comparison match, which is the GTCCRA input capture source of other channels	R/W
b6	ICAFFPO	Enable Overflow as Source of Other Channel GTCCRA Input Capture	0: Disable overflow for sawtooth wave and trough for triangular wave, which are the GTCCRA input capture source of other channels 1: Enable overflow for sawtooth wave and trough for triangular wave, which are the GTCCRA input capture source of other channels	R/W
b7	ICAFFPU	Enable Underflow as Source of Other Channel GTCCRA Input Capture	0: Disable underflow for sawtooth wave and trough for triangular wave, which are the GTCCRA input capture source of other channels 1: Enable underflow for sawtooth wave and trough for triangular wave, which are the GTCCRA input capture source of other channels	R/W
b8	ICACK	Enable Count Clock as Source of Other Channel GTCCRA Input Capture	0: Disable count clock, which are the GTCCRA input capture source of other channels 1: Enable count clock, which are the GTCCRA input capture source of other channels	R/W

Bit	Symbol	Bit Name	Description	R/W
b13 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15, b14	ICAGRP[1:0]	GTCCRA Input Capture Group Select	b15 b14 0 0: Group A 0 1: Group B 1 0: Group C 1 1: Group D	R/W
b16	ICBFA	Enable GTCCRA Register Compare Match/Input Capture as Source of Other Channel GTCCRB Input Capture	0: Disable GTCCRA register comparison match/input capture transfer, which is the GTCCRB input capture source of other channels 1: Enable GTCCRA register comparison match/input capture transfer, which is the GTCCRB input capture source of other channels	R/W
b17	ICBFB	Enable GTCCRB Register Compare Match/Input Capture as Source of Other Channel GTCCRB Input Capture	0: Disable GTCCRB register comparison match/input capture transfer, which is the GTCCRB input capture source of other channels 1: Enable GTCCRB register comparison match/input capture transfer, which is the GTCCRB input capture source of other channels	R/W
b18	ICBFC	Enable GTCCRC Register Compare Match as Source of Other Channel GTCCRB Input Capture	0: Disable GTCCRC register comparison match, which is the GTCCRB input capture source of other channels 1: Enable GTCCRC register comparison match, which is the GTCCRB input capture source of other channels	R/W
b19	ICBFD	Enable GTCCRD Register Compare Match as Source of Other Channel GTCCRB Input Capture	0: Disable GTCCRD register comparison match, which is the GTCCRB input capture source of other channels 1: Enable GTCCRD register comparison match, which is the GTCCRB input capture source of other channels	R/W
b20	ICBFE	Enable GTCCRE Register Compare Match as Source of Other Channel GTCCRB Input Capture	0: Disable GTCCRE register comparison match, which is the GTCCRB input capture source of other channels 1: Enable GTCCRE register comparison match, which is the GTCCRB input capture source of other channels	R/W
b21	ICBFF	Enable GTCCRF Register Compare Match as Source of Other Channel GTCCRB Input Capture	0: Disable GTCCRF register comparison match, which is the GTCCRB input capture source of other channels 1: Enable GTCCRF register comparison match, which is the GTCCRB input capture source of other channels	R/W
b22	ICBFPO	Enable Overflow as Source of Other Channel GTCCRB Input Capture	0: Disable overflow for sawtooth wave and trough for triangular wave, which are the GTCCRB input capture source of other channels 1: Enable overflow for sawtooth wave and trough for triangular wave, which are the GTCCRB input capture source of other channels	R/W
b23	ICBFPU	Enable Underflow as Source of Other Channel GTCCRB Input Capture	0: Disable underflow for sawtooth wave and trough for triangular wave, which are the GTCCRB input capture source of other channels 1: Enable underflow for sawtooth wave and trough for triangular wave, which are the GTCCRB input capture source of other channels	R/W
b24	ICBCLK	Enable Count Clock as Source of Other Channel GTCCRB Input Capture	0: Disable count clock, which are the GTCCRB input capture source of other channels 1: Enable count clock, which are the GTCCRB input capture source of other channels	R/W
b29 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31, b30	ICBGRP[1:0]	GTCCRB Input Capture Group Select	b31 b30 0 0: Group A 0 1: Group B 1 0: Group C 1 1: Group D	R/W

The GTICCR register is a register that controls input capture by inter channel link.

For channels that perform input capture in cooperation, the input capture factor set in the GTICASR and GTICASR registers is invalid.

ICAFm Bit (Enable GTCCRm Register Compare Match/Input Capture as Source of Other Channel GTCCRA Input Capture) (m = A, B)

Enables/disables the use of compare match/input capture of GTCCRm register as the input capture source of other channel's GTCCRA register.

ICAFn Bit (Enable GTCCRn Register Compare Match as Source of Other Channel GTCCRA Input Capture) (n = C, D, E, F)

Enables/disables the use of compare match of GTCCRn register as the input capture source of other channel's GTCCRA register.

ICAFPO Bit (Enable Overflow as Source of Other Channel GTCCRA Input Capture)

Enable/disable to use the overflow of sawtooth-waves, the crest of triangle-waves as the input capture source of other channel's GTCCRA register.

ICAFPU Bit (Enable Underflow as Source of Other Channel GTCCRA Input Capture)

Enable/disable to use the underflow of sawtooth-waves, the trough of triangle-waves as the input capture source of other channel's GTCCRA register.

ICACLK Bit (Enable Count Clock as Source of Other Channel GTCCRA Input Capture)

Enable/disable to use count clock as the input capture source of other channel's GTCCRA register.

ICAGRP[1:0] Bits (GTCCRA Input Capture Group Select)

Select the group of input capture by inter channel cooperation for GTCCRA register.

For channels that accept input capture of the GTCCRA register due to input capture sources from other channels, set the GTICASR.ASOC bit to 1 and select the group of inter channel cooperation with the ICAGRP[1:0] bits.

ICBFm Bit (Enable GTCCRm Register Compare Match/Input Capture as Source of Other Channel GTCCRB Input Capture) (m = A, B)

Enables/disables the use of compare match/input capture of GTCCRm register as the input capture source of other channel's GTCCRB register.

ICBFn Bit (Enable GTCCRn Register Compare Match as Source of Other Channel GTCCRB Input Capture) (n = C, D, E, F)

Enables/disables the use of compare match of GTCCRn register as the input capture source of other channel's GTCCRB register.

ICBFPO Bit (Enable Overflow as Source of Other Channel GTCCRB Input Capture)

Enable/disable to use the overflow of sawtooth-waves, the crest of triangle-waves as the input capture source of other channel's GTCCRB register.

ICBFPU Bit (Enable Underflow as Source of Other Channel GTCCRB Input Capture)

Enable/disable to use the underflow of sawtooth-waves, the trough of triangle-waves as the input capture source of other channel's GTCCRB register.

ICBCLK Bit (Enable Count Clock as Source of Other Channel GTCCRB Input Capture)

Enable/disable to use count clock as the input capture source of other channel's GTCCRB register.

ICBGRP[1:0] Bits (GTCCRB Input Capture Group Select)

Select the group of input capture by inter channel cooperation for GTCCRB register.

For channels that accept input capture of the GTCCRB register due to input capture sources from other channels, set the GTICBSR.BSOC bit to 1 and select the group of inter channel cooperation with the ICBGRP[1:0] bits.

21.3 Operation

21.3.1 Basic Operation

The timer in each channel performs cycle count operation by count clock. The GTCNT counter performs up-counting, or down-counting. The timer period is controlled by the GTPR or GTCCRM ($m = A$ to F) registers. When the GTCNT counter value matches the values for GTCCRA or GTCCRB registers, the GTIOCnA or GTIOCnB pin outputs ($n = 0$ to 2) can be changed respectively. Also, the GTCCRA or GTCCRB registers can be used as the input capture registers by hardware source. The GTCCRC and GTCCRD registers can be performed as buffer registers for GTCCRA register, and the GTCCRE and GTCCRF registers can be performed as buffer registers for GTCCRB register.

21.3.1.1 Counter Operation

(1) Count Start/Count Stop

The counter in each channel starts counting operation when the corresponding GTCR.CST bit is set to 1, and stops counting when the bit is set to 0.

CST bit value can be changed by the following sources:

- Writing to the GTCR register
- Writing 1 to the bit corresponding to the channel number for the GTSTR register while the GTSSR.CSTRT bit is 1
- Writing 1 to the bit corresponding to the channel number for the GTSTP register while the GTPSR.CSTOP bit is 1
- Hardware source specified by the GTSSR register
- Hardware source specified by the GTPSR register

(2) Cycle Count Operation (In Up-Counting by the Count Clock)

The counter in each channel starts up-counting when the corresponding CST bit is set to 1. When the value of the GTCNT counter changes from the GTPR register value to 0000 0000h (an overflow), the GTST.TCFPO flag is set to 1, and if the setting of the GTINTAD.GTINTPR[0] bit is 1, a GTCIV interrupt request is also issued. After the GTCNT counter has overflowed, counting up is resumed from 0000 0000h.

Figure 21.2 shows an example of cycle count operation in up-counting by count clock.

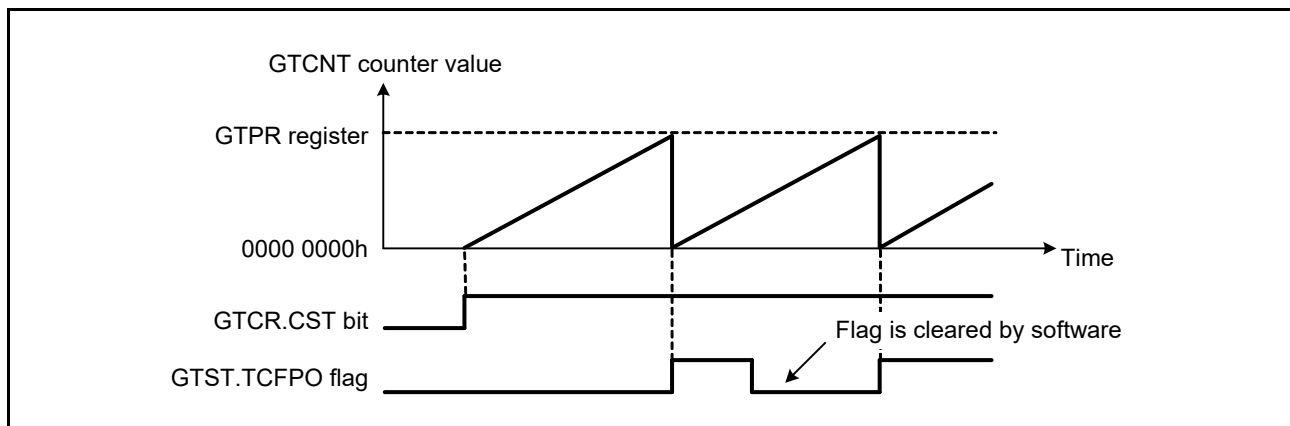


Figure 21.2 Example of Cycle Count Operation (In Up-Counting by the Count Clock)

Figure 21.3 shows an example for setting cycle count operation in up-counting.

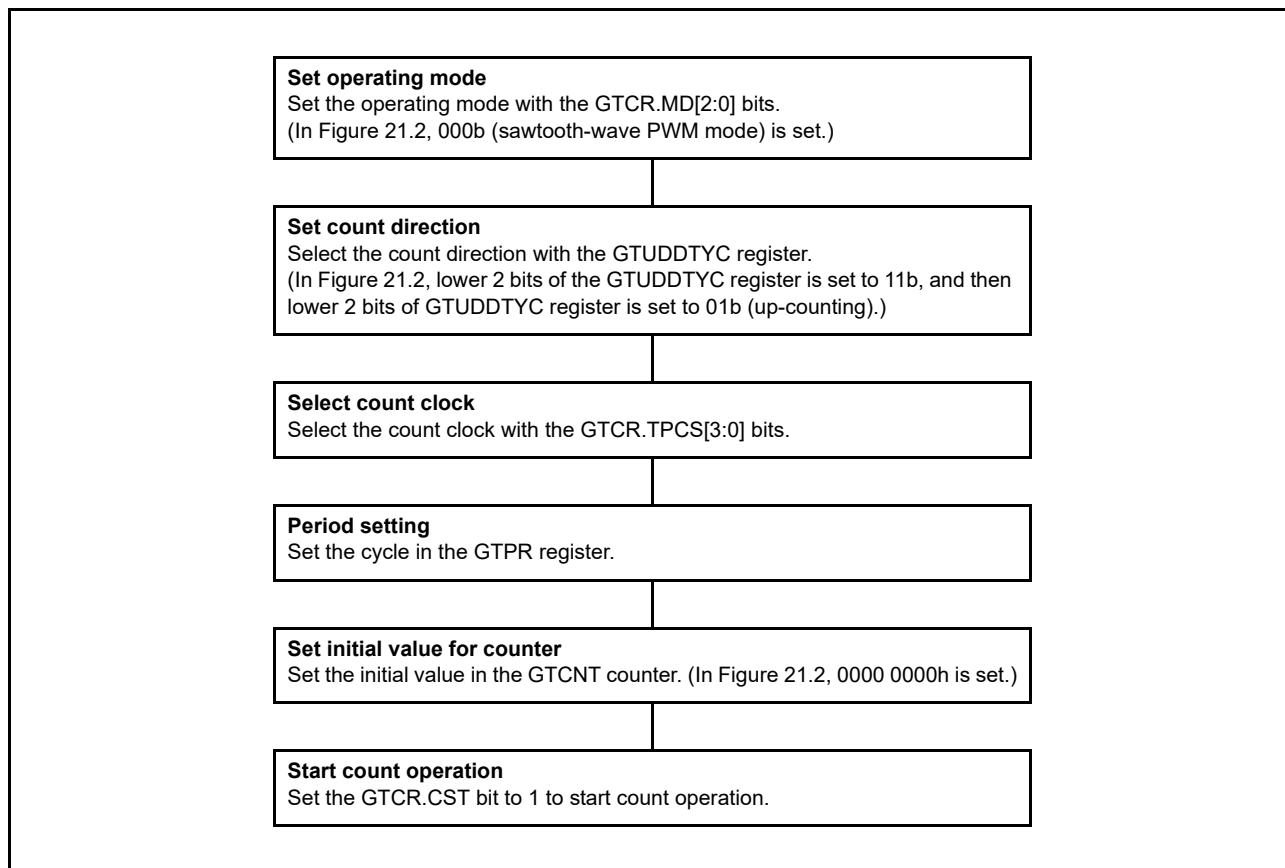


Figure 21.3 Example for Setting Cycle Count Operation (In Up-Counting by the Count Clock)

(3) Cycle Count Operation (In Down-Counting by the Count Clock)

The counter in each channel can start down-counting when the corresponding GTUDDTYC.UD bit is set. When the value of the GTCNT counter changes from that of the GTPR register to 0000 0000h (an underflow), the GTST.TCFPU flag is set to 1, and if the setting of the GTINTAD.GTINTPR[1] bit is 1, a GTCIU interrupt request is also issued. After the GTCNT counter underflows, down-counting is resumed from the GTPR register value.

Figure 21.4 shows an example of cycle count operation in down-counting by count clock.

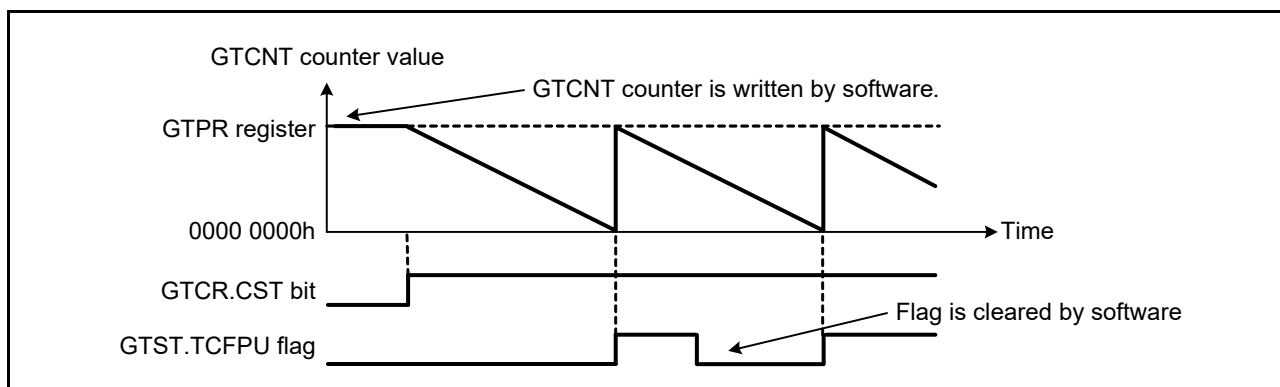


Figure 21.4 Example of Cycle Count Operation (In Down-Counting by the Count Clock)

Figure 21.5 shows an example for setting cycle count operation in down-counting by count clock.

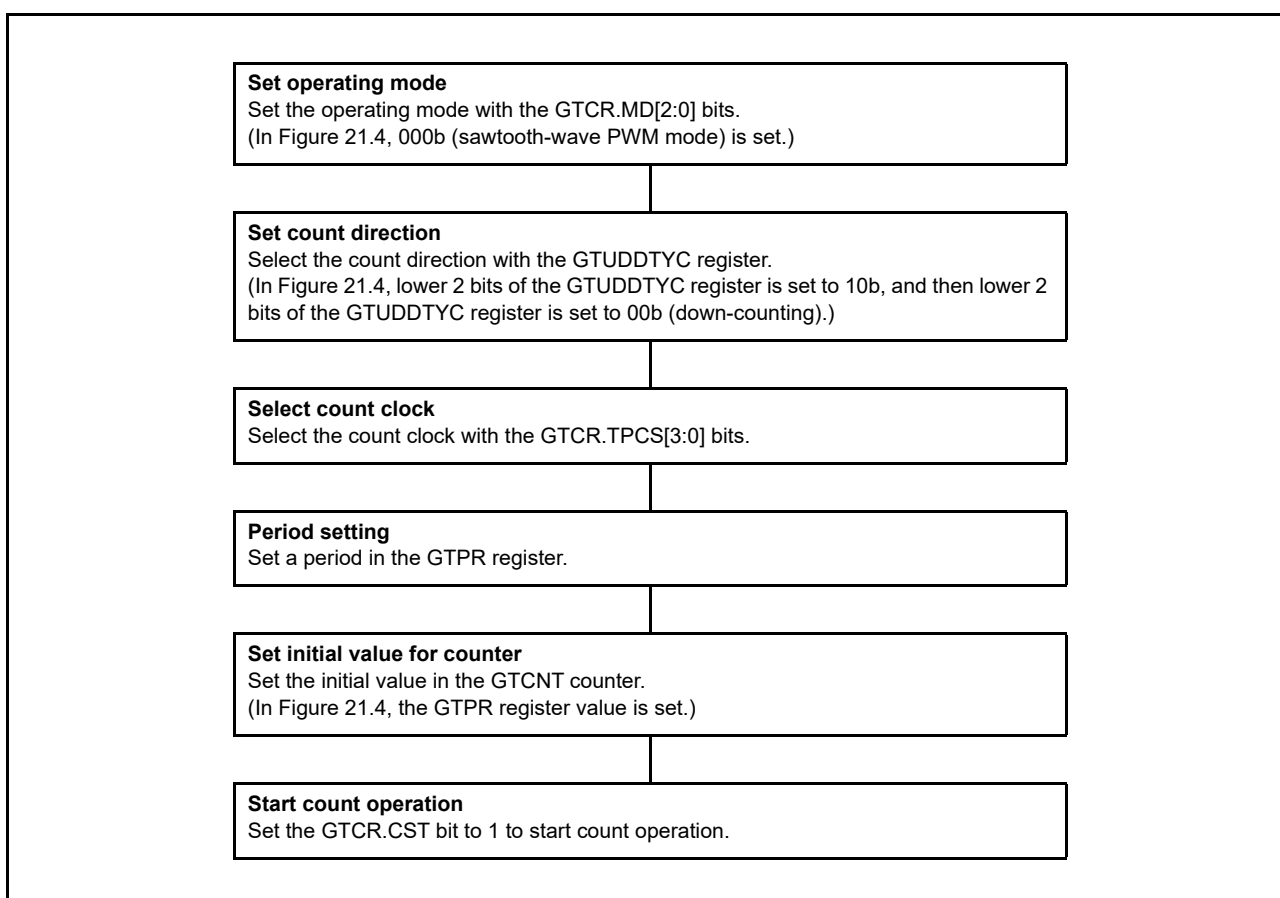


Figure 21.5 Example for Setting Cycle Count Operation (In Down-Counting by the Count Clock)

(4) Counter Clearing Operation

The counter in each channel can be cleared by the following sources:

- Writing to the GTCNT counter
- Writing 1 to the bit corresponding to the channel number for the GTCLR register while the GTCSR.CCLR bit is 1
- Hardware source specified by the GTCSR register

Write access during counting (when CST = 1) is disabled.

GTCLR register can be written to and cleared by hardware source whether the count operation is performed (GTCR.CST bit = 1) or is stopped (CST bit = 0). When the count direction flag is set as decrement (GTST.TUCF flag = 0) in sawtooth-wave mode selected with GTCR.MD[2:0] bits, the GTCNT counter value for writing to the GTCLR register and for clearing by hardware source becomes the GTPR register value whether the counter is in operation or is stopped. The GTCNT counter value becomes 0000 0000h for other settings.

21.3.1.2 Waveform Output by Compare Match

Compare match refers to when the GTCNT counter value matches the GTCCRA or GTCCRB register value. The output of the corresponding GTIOCnA or GTIOCnB pin (n = 0 to 2) can be set to be driven low, high, or toggled in synchronization with the counter clock after the match.

In addition, the GTIOCnA or GTIOCnB pin output can be driven low, high, or toggled at the “end of the cycle” which is determined by the GTPR register. The end of the cycle is as follows:

- For sawtooth waves in up-counting: When the GTCNT counter value changes from the GTPR register value to 0000 0000h (overflow)
- For sawtooth waves in down-counting: When the GTCNT counter value changes from 0000 0000h to the GTPR register value (underflow)
- For sawtooth waves with the GTCNT counter clear
- For triangle waves: When the GTCNT counter value changes from 0000 0000h to 0000 0001h (trough)

(1) Low Output and High Output

Figure 21.6 shows an example of low output and high output operation by a compare match of the GTCNT counter and GTCCRA register and of the GTCNT counter and GTCCRB register.

In this example, the GPTW0.GTCNT counter performs up-counting, and settings have been made so that high is output from the GTIOC0A pin by a GPTW0.GTCCRA register compare match, and low is output from the GTIOC0B pin by a GPTW0.GTCCRB register compare match. The pin level does not change when the specified level and pin level match.

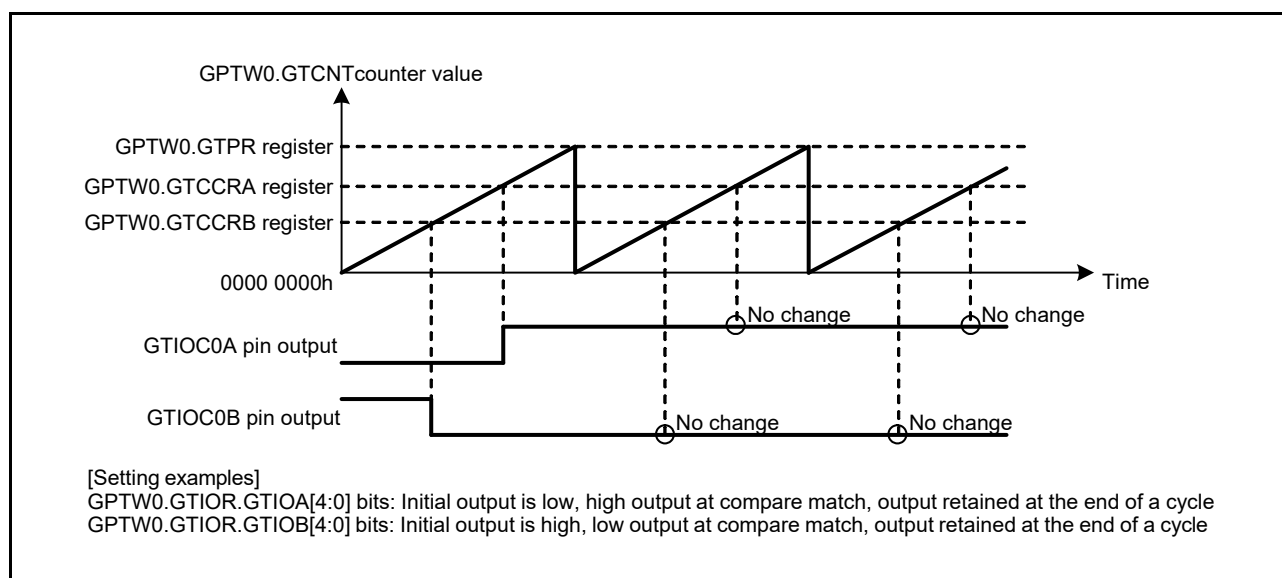


Figure 21.6 Example of Low Output and High Output Operation

Figure 21.7 shows an example for setting Low/High output operation.

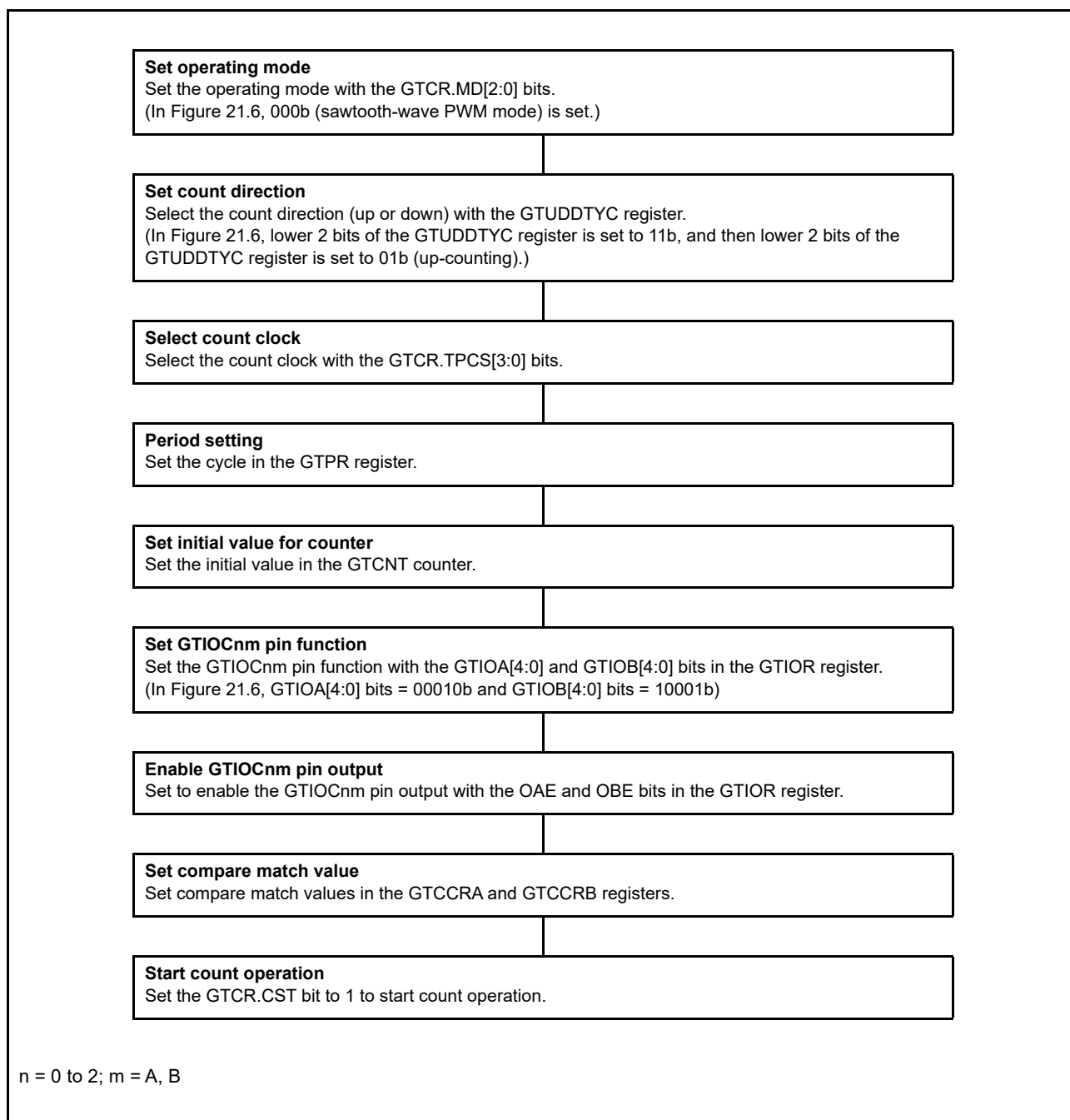


Figure 21.7 Example for Setting Low/High Output Operation

(2) Toggled Output

Figure 21.8 and Figure 21.9 show examples of toggle output operations by compare match between the GPTW0.GTCNT counter and the GTCCRA and GTCCRB registers.

In Figure 21.8, the GPTW0.GTCNT counter performs up-counting, and settings have been made so that the GTIOC0A pin output by a GPTW0.GTCCRA register compare match and GTIOC0B pin output by a GPTW0.GTCCRB register compare match are toggled.

In Figure 21.9, the GPTW0.GTCNT counter performs up-counting, and settings have been made so that the GTIOC0A pin output is toggled by a compare match of GPTW0.GTCCRA register and the GTIOC0B pin output is toggled at the end of the cycle.

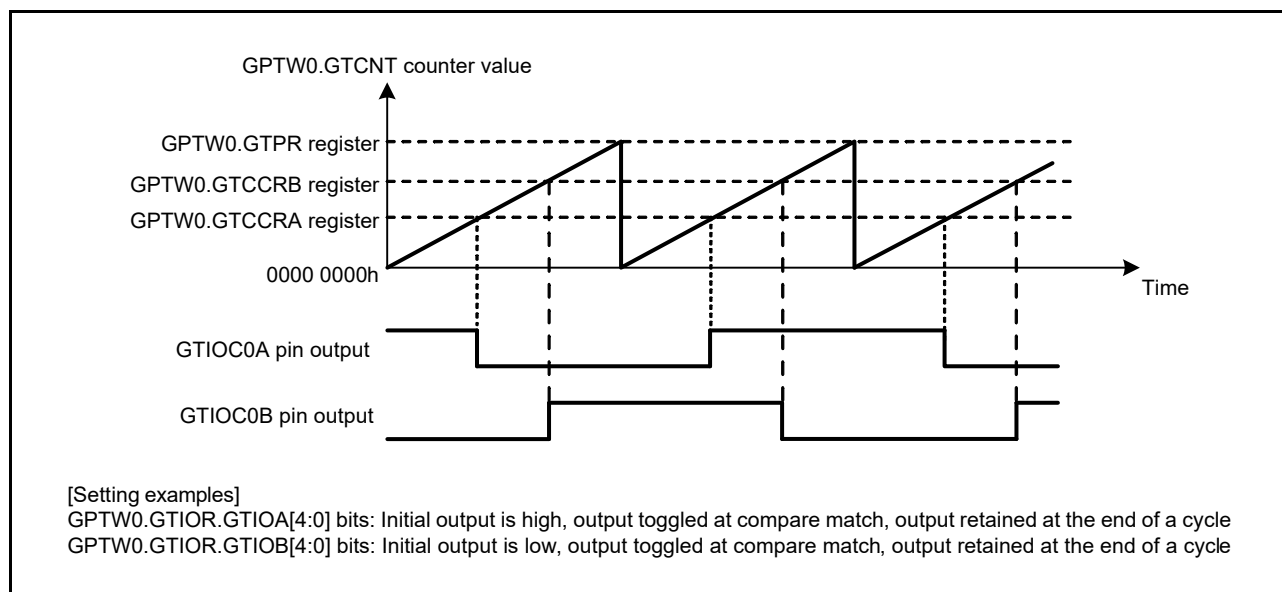


Figure 21.8 Example of Toggled Output Operation (1)

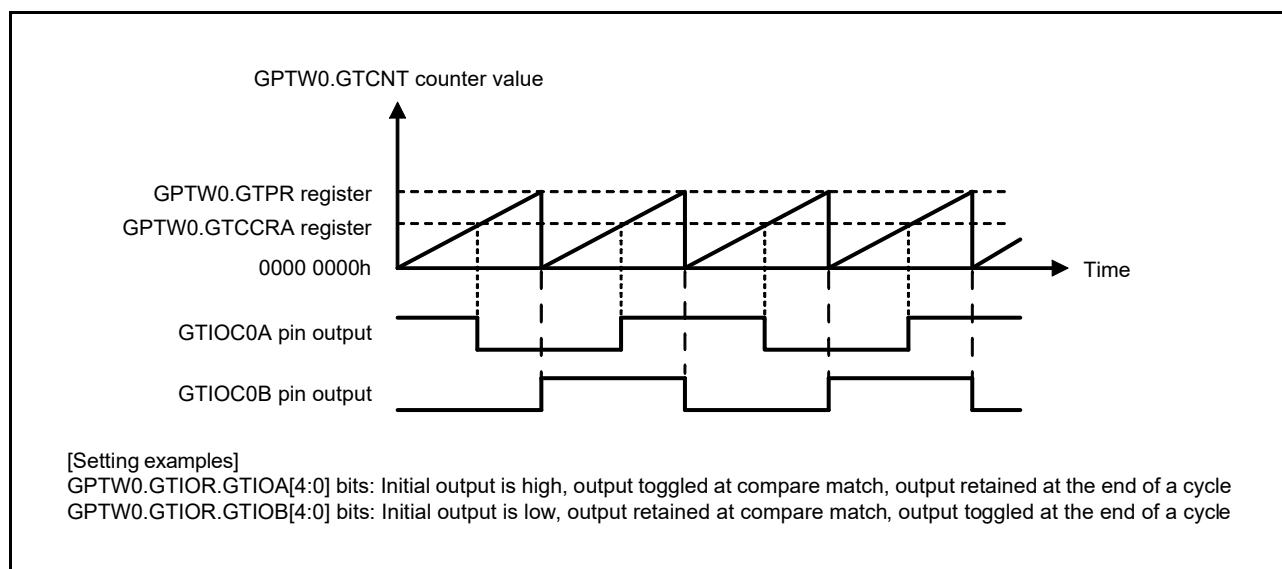


Figure 21.9 Example of Toggled Output Operation (2)

Figure 21.10 shows an example for setting toggled output operation.

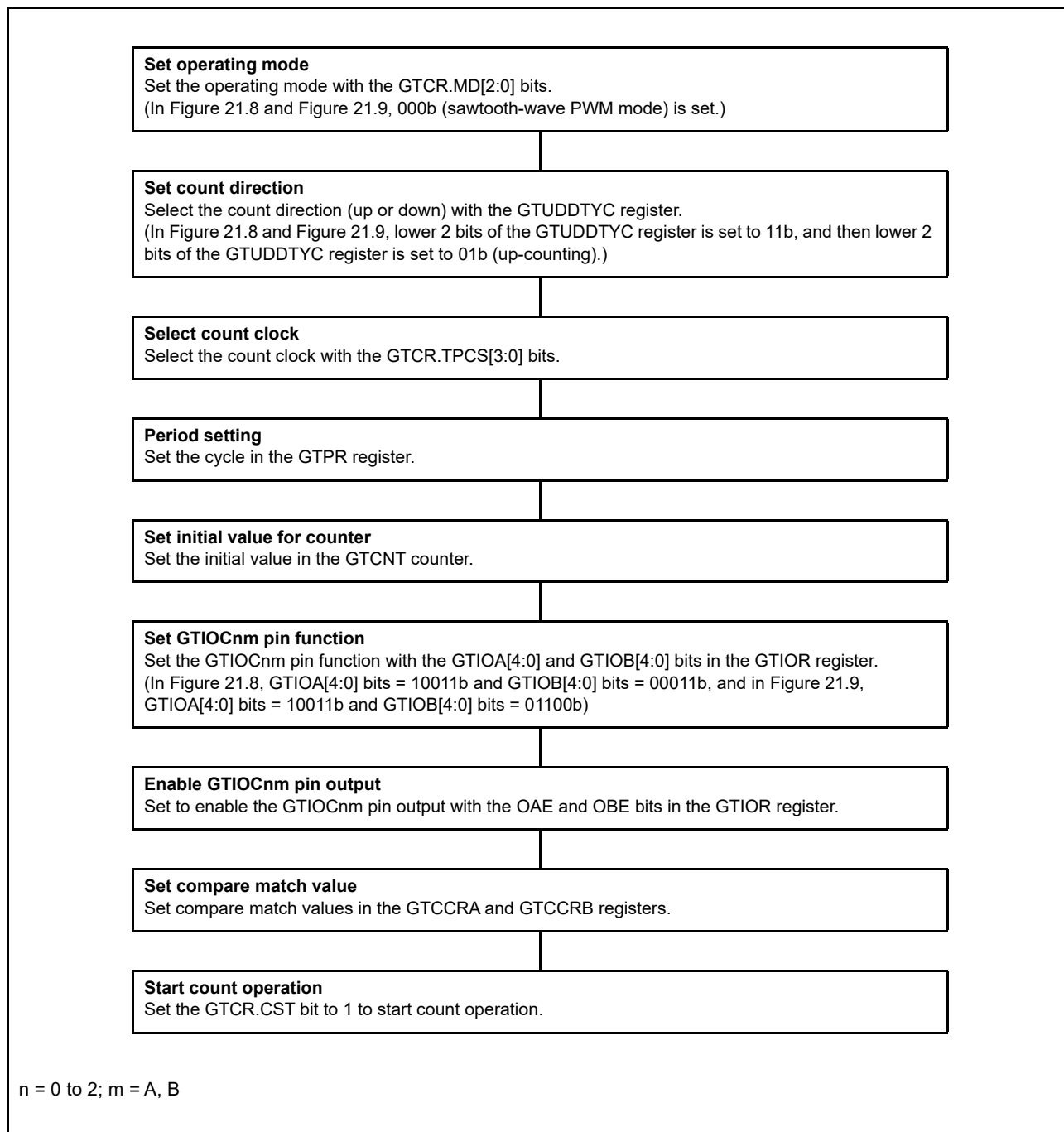


Figure 21.10 Example for Setting Toggled Output Operation

21.3.1.3 Input Capture Function

Through detecting hardware sources selected by the GTICASR and GTICBSR registers, the GTCNT counter values can be transferred to the GTCCRA and GTCCRB registers respectively.

Figure 21.11 shows an example of input capture function operation.

In this example, the GPTW0.GTCNT counter performs up-counting, and settings have been made so that an input capture is performed at both edges of the GTIOC0A pin input and at the rising edge of the GTIOC0B pin input.

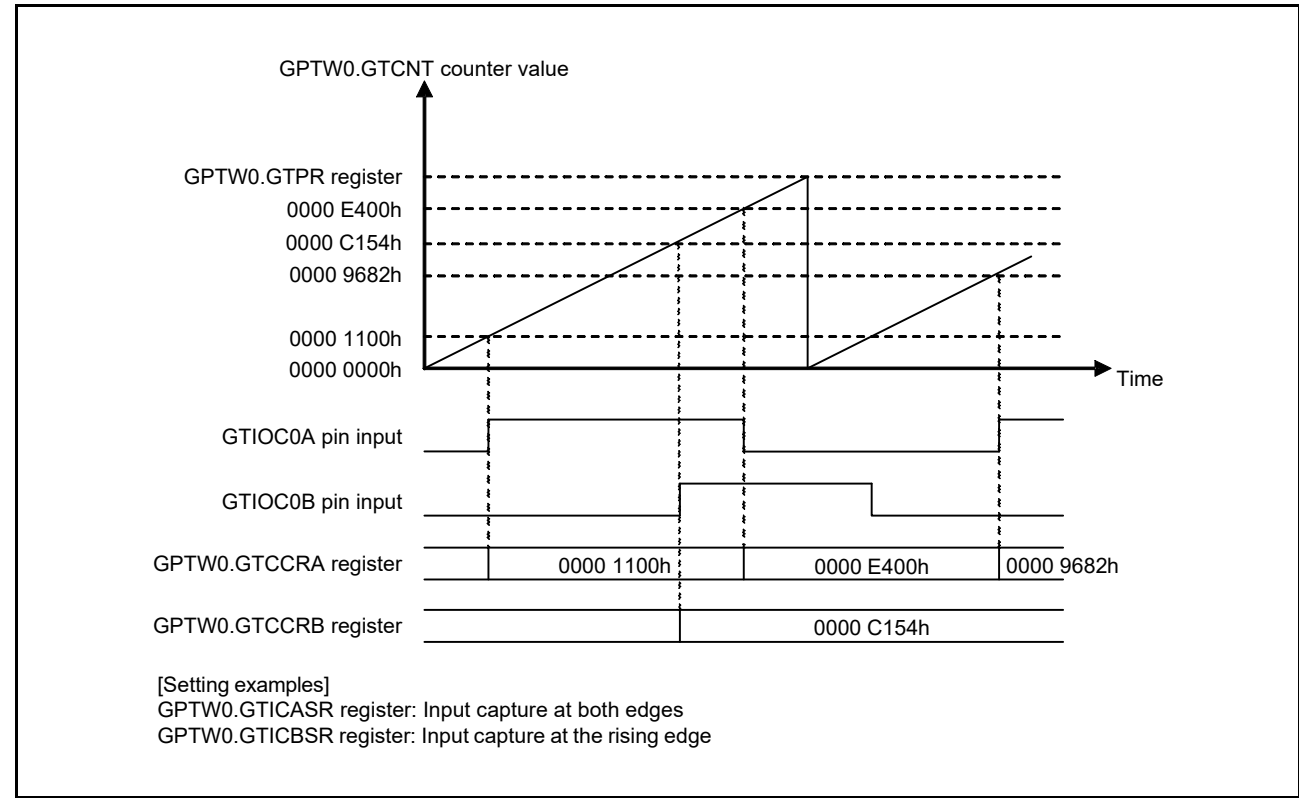


Figure 21.11 Example of Input Capture Operation

Figure 21.12 shows an example for setting input capture operation.

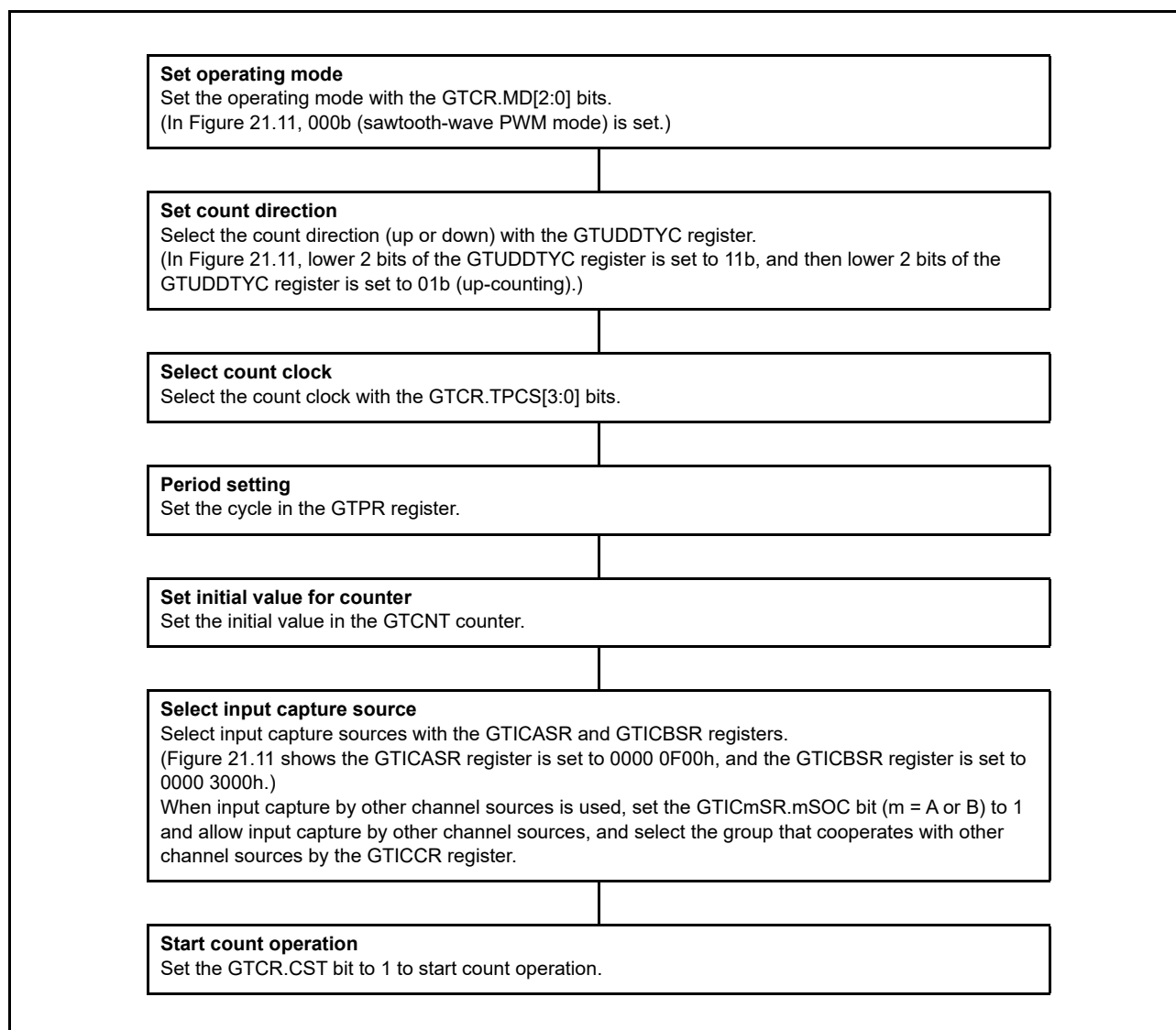


Figure 21.12 Example for Setting Input Capture Operation

Figure 21.13 shows an example of the timing of input capture operation in response to a rising edge of the input on the GTETRGA pin.

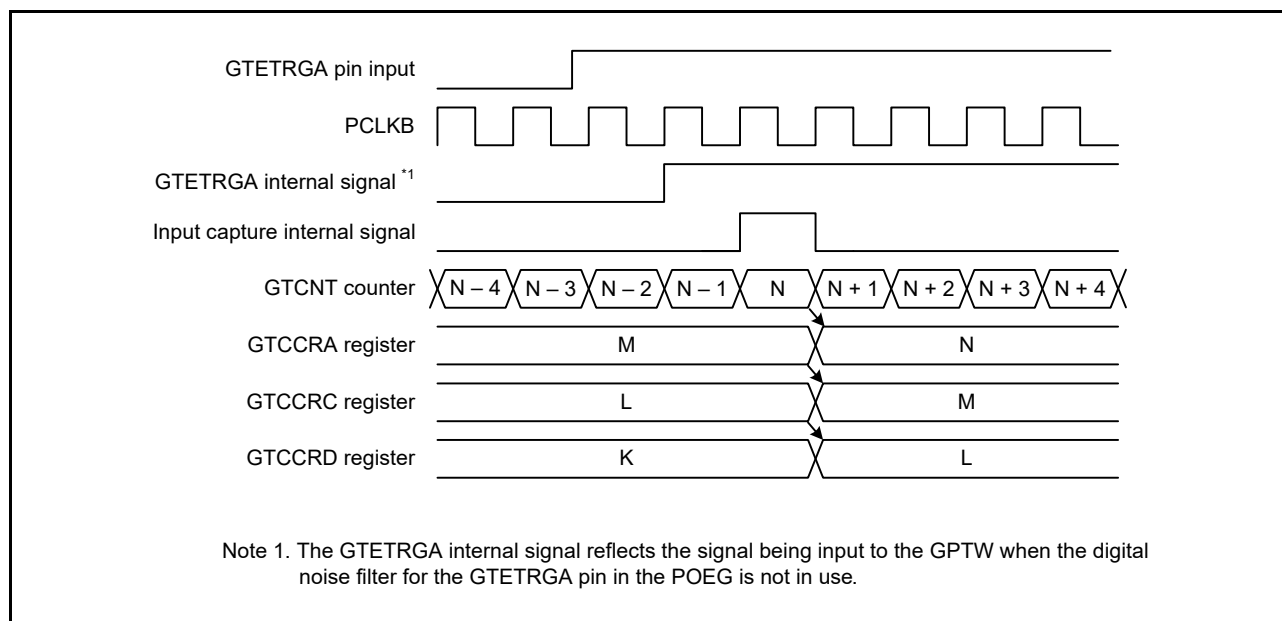


Figure 21.13 Example of the Timing of Input Capture Operation in Response to a Rising Edge of the Input on the GTETRGA Pin

Figure 21.14 shows an example of the timing of input capture operation in response to a rising edge of the input on the GTIOCnA pin.

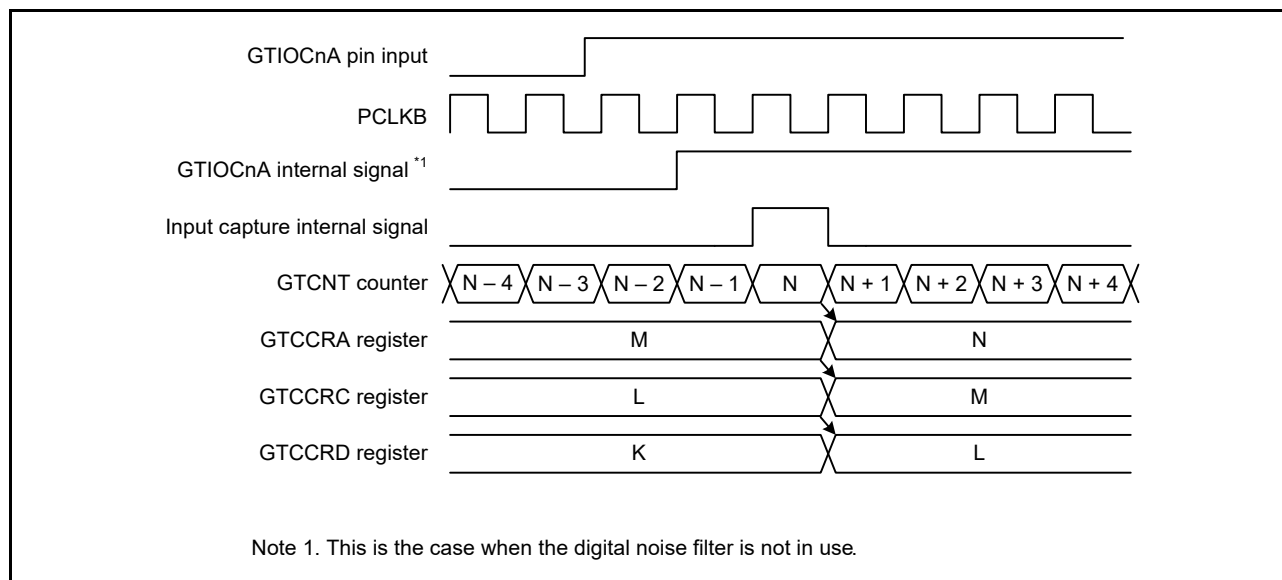


Figure 21.14 Example of the Timing of Input Capture Operation in Response to a Rising Edge of the Input on the GTIOCnA Pin

Figure 21.15 shows the input capture operation timing by the counter clock of other channels.

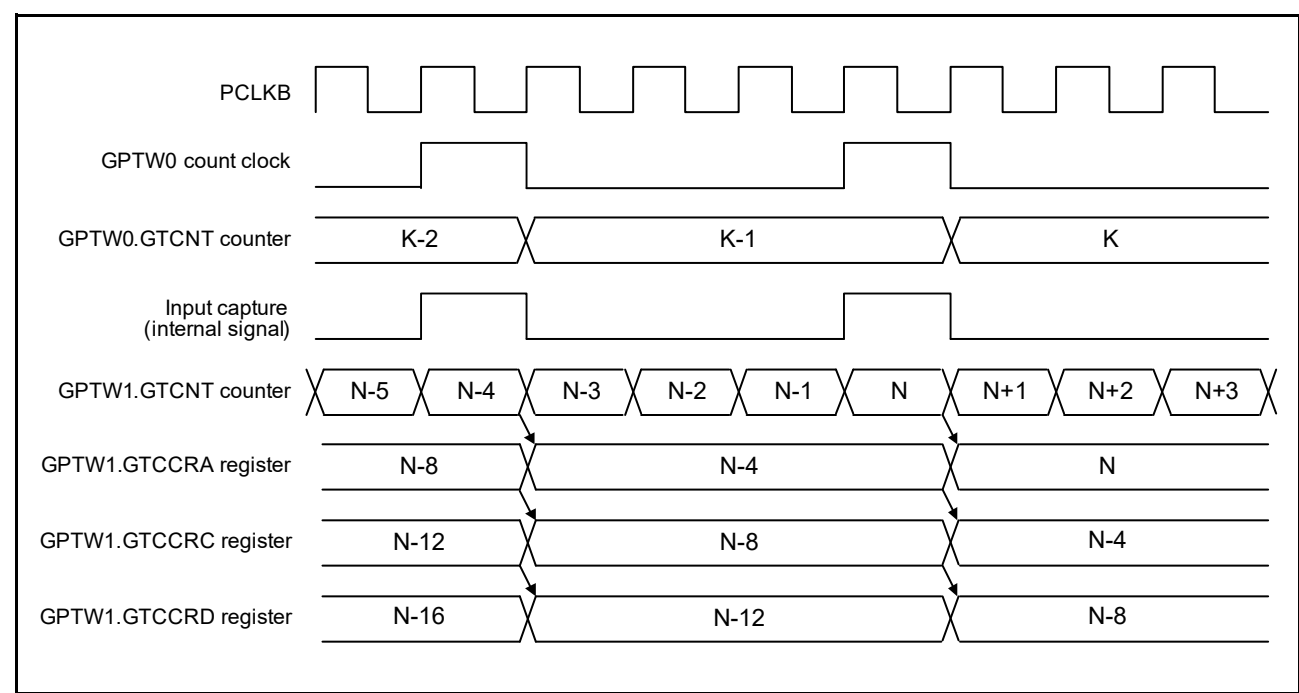


Figure 21.15 Example of the Timing of Input Capture Operation in Response to Count Clock from Other Channel

21.3.2 Buffer Operation

The following buffer operation can be set with the GTBER register.

- Buffer operation with the GTPR, GTPBR, and GTPDBR registers used together
- Buffer operation with the GTCCRA, GTCCRC, and GTCCRD registers used together
- Buffer operation with the GTCCRB, GTCCRE, and GTCCRF registers used together
- Buffer operation with the GTADTRA, GTADTBRA, and GTADTDBRA registers used together
- Buffer operation with the GTADTRB, GTADTBRB, and GTADTDBRB registers used together

The following buffer operation can be set with the GTDTCR register.

- Buffer operation with the GTDVU and GTDBU registers used together
- Buffer operation with the GTDVD and GTDBD registers used together

21.3.2.1 GTPR Register Buffer Operation

The GTPBR register can function as a buffer register for the GTPR register, and the GTPDBR register can function as a buffer register for the GTPBR register (double buffer register for the GTPR register).

The buffer transfer is performed at an overflow (in up-counting) or underflow (in down-counting) in sawtooth-wave mode, or at a trough in triangle-wave mode.

When in sawtooth-wave mode, the buffer transfer is performed even when the counter clearing occurs during the count is in operation.

- Hardware source clearing

(Clear source selected with GTCSR.CSGTRGAR, CSGTRGAF, CSCARBL, CSCARBH, CSCAFBL, CSCAFBH, CSCBRAL, CSCBRAH, CSCBFAL, CSCBFAH, or CSCMSC[2:0] bit)

- Clearing by software

(When the GTCLR.CCLRn bit is written to 1 during the GTCSR.CCLR bit is set to 1) (n = 0 to 2)

In the case of sawtooth waves, the buffer transfer by clearing counter can be prohibited by the GTBER2.CCTPR bit.

To set the GTPR register to function as double buffer, set the GTBER.PR[1:0] bits to 10b or 11b. For single buffer operation, set 01b. Not to function as buffer, set 00b.

Figure 21.16 to Figure 21.18 show examples of the GTPR register buffer operation and Figure 21.19 shows an example for setting the GTPR register buffer operation.

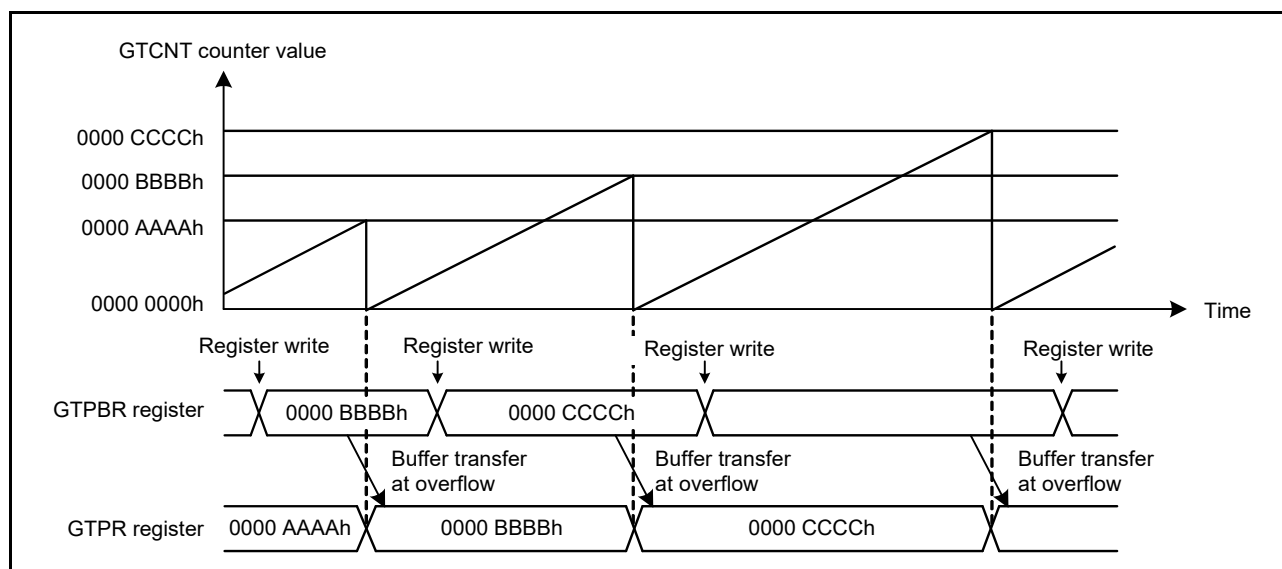


Figure 21.16 Example of the GTPR Register Buffer Operation (Sawtooth Waves in Up-Counting)

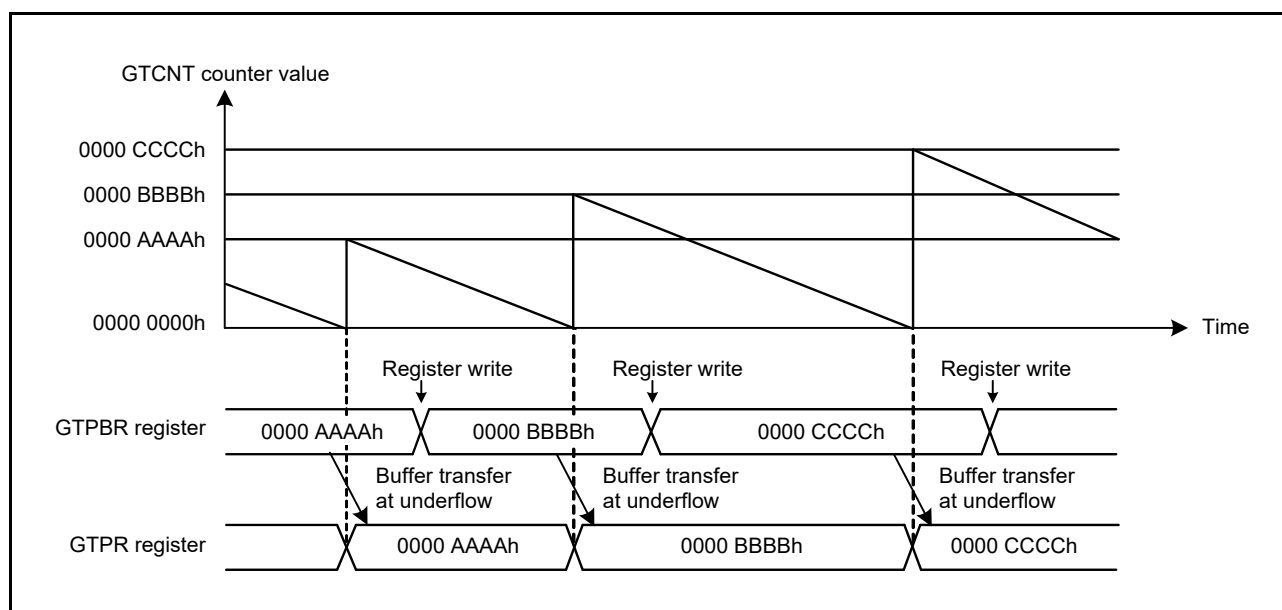


Figure 21.17 Example of the GTPR Register Buffer Operation (Sawtooth Waves in Down-Counting)

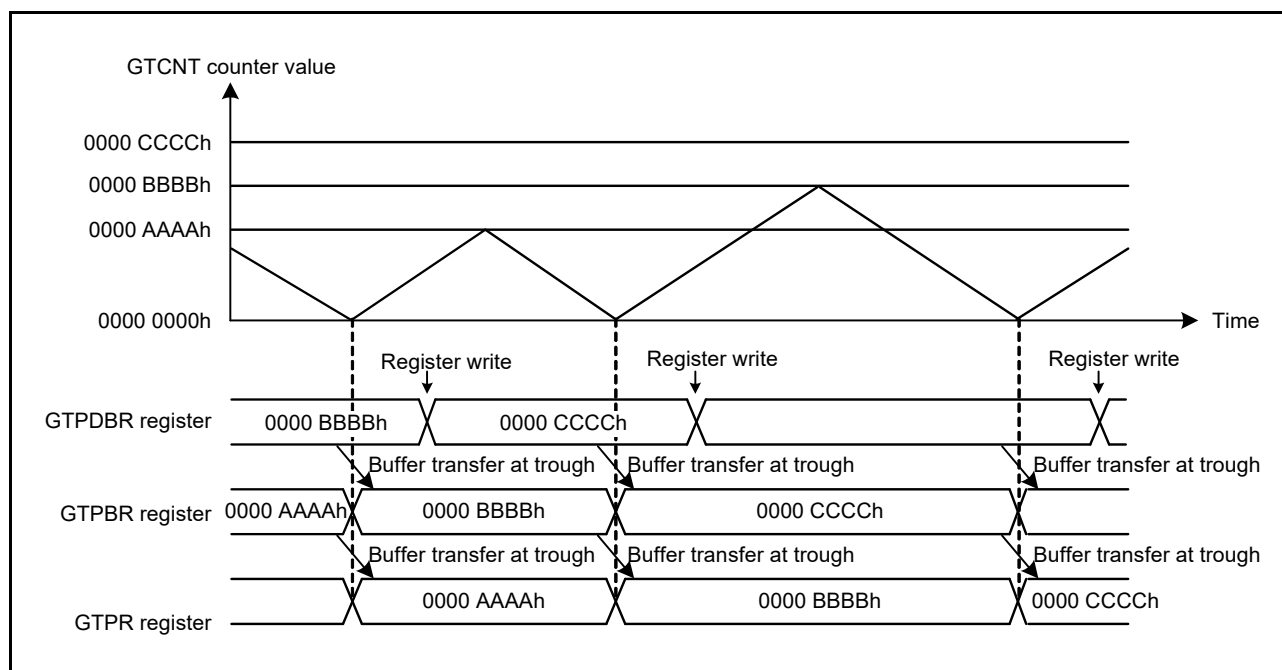


Figure 21.18 Example of the GTPR Register Double Buffer Operation (Triangle Waves)

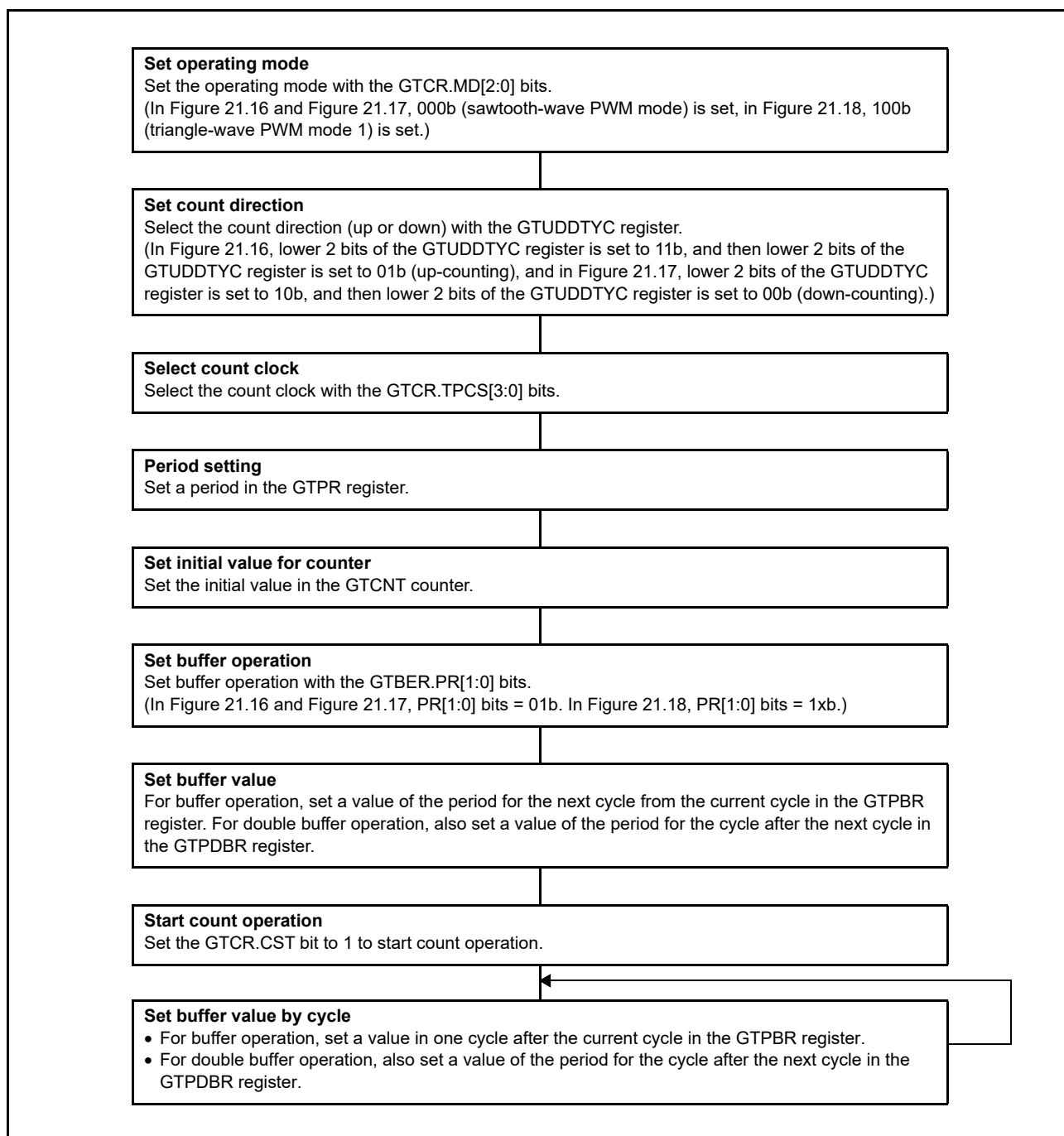


Figure 21.19 Example for Setting the GTPR Register Buffer Operation

21.3.2.2 Buffer Operation for the GTCCRA and GTCCRB Registers

The GTCCRC register can function as the GTCCRA buffer register and the GTCCRD register can function as the GTCCRC buffer register (double buffer register for the GTCCRA register). Similarly, the GTCCRE register can function as the GTCCRB buffer register and the GTCCRF register can function as the GTCCRE buffer register (double buffer register for the GTCCRB register).

To set the GTCCRA or GTCCRB register to function as a double buffer, set the GTBER.CCRA[1:0] or GTBER.CCRB[1:0] bits to 10b or 11b. For single buffer operation, set 01b. Not to function as buffer, set 00b.

The following describes buffer operation during output compare and input capture operation.

In sawtooth-wave one-shot pulse mode, triangle-wave PWM mode 3, the buffer operations that specific each PWM output operation mode are performed regardless of the setting of GTBER.CCRA[1:0] bits and GTBER.CCRB[1:0] bits.

(1) When the GTCCRA or GTCCRB Register Functions as Output Compare Register

In sawtooth-wave one-shot pulse mode, triangle-wave PWM mode 3, the buffer operations that specific each PWM output operation mode are performed regardless of the setting of GTBER.CCRA[1:0] bits and GTBER.CCRB[1:0] bits. For details, refer section 21.3.3, PWM Output Operating Mode. Other than above PWM output operation mode, buffer transfer is performed for the following four cases:

- Buffer transfer at an overflow (in up-counting) and underflow (in down-counting)
Buffer transfer is performed at an overflow (in up-counting) or an underflow (in down-counting) in sawtooth-wave mode, and at a trough (triangle-wave PWM mode 1) or at a crest/trough (triangle-wave PWM mode 2) in triangle-wave mode.
- Buffer transfer by counter clearing
In sawtooth-wave mode, the buffer transfer by similar counter clearing sources in counting operation and in section 21.3.2.1, GTPR Register Buffer Operation is performed in the same way at an overflow (in up-counting) or an underflow (in down-counting).
In sawtooth-wave, the buffer transfer of GTCCRm register by counter clearing is possible to be disabled with GTBER2.CCTCm bit (m = A, B).
In triangle-wave mode, buffer transfer by counter clearing is not performed.
- Buffer transfer by compare match
In sawtooth-wave, the buffer transfer by the compare match of GTCCRm register enabled by GTBER2.CMTCm (m = A, B) bit is performed.
- Forcible buffer transfer
In sawtooth-wave mode and triangle-wave mode, buffer transfer of the GTCCRA and GTCCRB registers is forcibly performed by writing 1 to the GTBER.CCRSWT bit while the counting is stopped.
In sawtooth-wave one-shot pulse mode and triangle-wave PWM mode 3, buffer transfer from the GTCCRD register to the temporary register A and from the GTCCRF register to the temporary register B is also performed using the forcible buffer operation.

When the setting of the GTBER.DBRTCEm (m = A, B) bit is 1 in sawtooth-wave one-shot pulse mode or triangle-wave PWM mode 3, transfer from the intermediate buffers to the GTCCRm (m = A, B) registers is repeated on a cyclic basis even while buffer transfer is disabled by the setting of the GTBER.BD[0] bit or buffer transfer extended skipping function (function for repeated double-buffer operation while buffer transfer is disabled). For details, refer to section 21.7.2.2, Repeated Double-Buffered Operation When Disabling GTCCR Buffer Transfer.

Figure 21.20 to Figure 21.22 show examples of the GTCCRA and GTCCRB registers buffer operations, and Figure 21.23 shows an example for setting the GTCCRA and GTCCRB registers buffer operations.

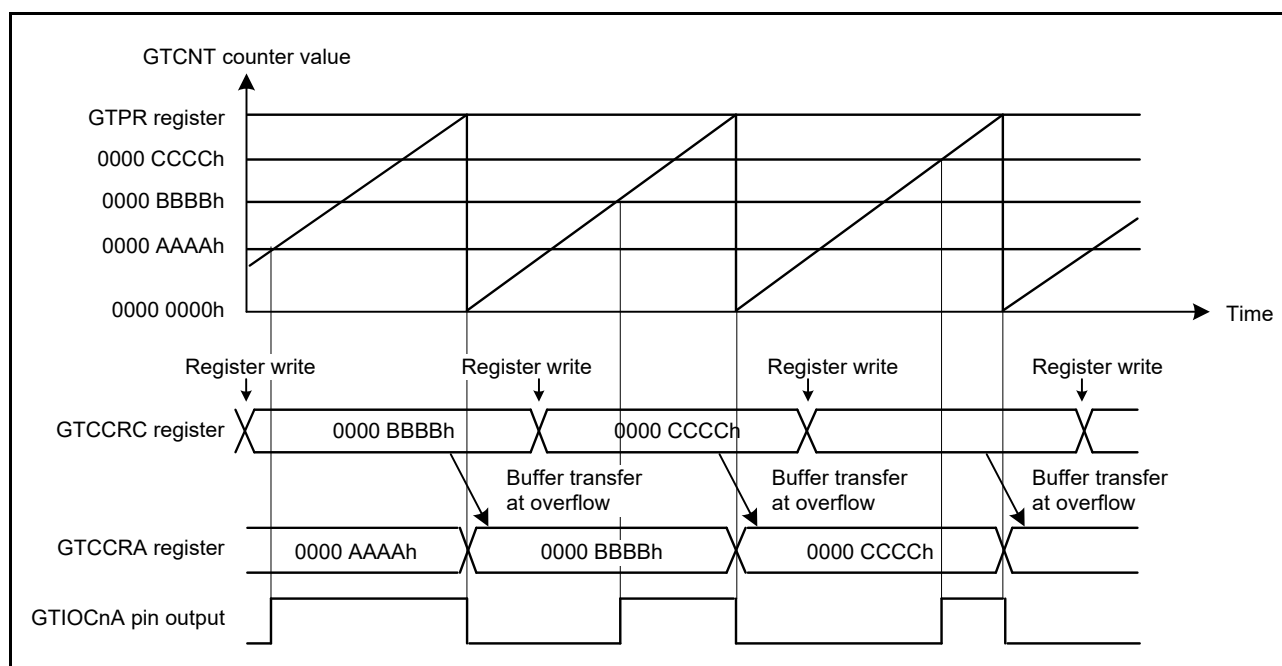


Figure 21.20 Example of GTCCRA and GTCCRB Registers Buffer Operation (Output Compare, Sawtooth-Waves in Up-Counting, High Output at GTCCRA Register Compare Match, Low Output at the End of the Cycle) ($n = 0$ to 2)

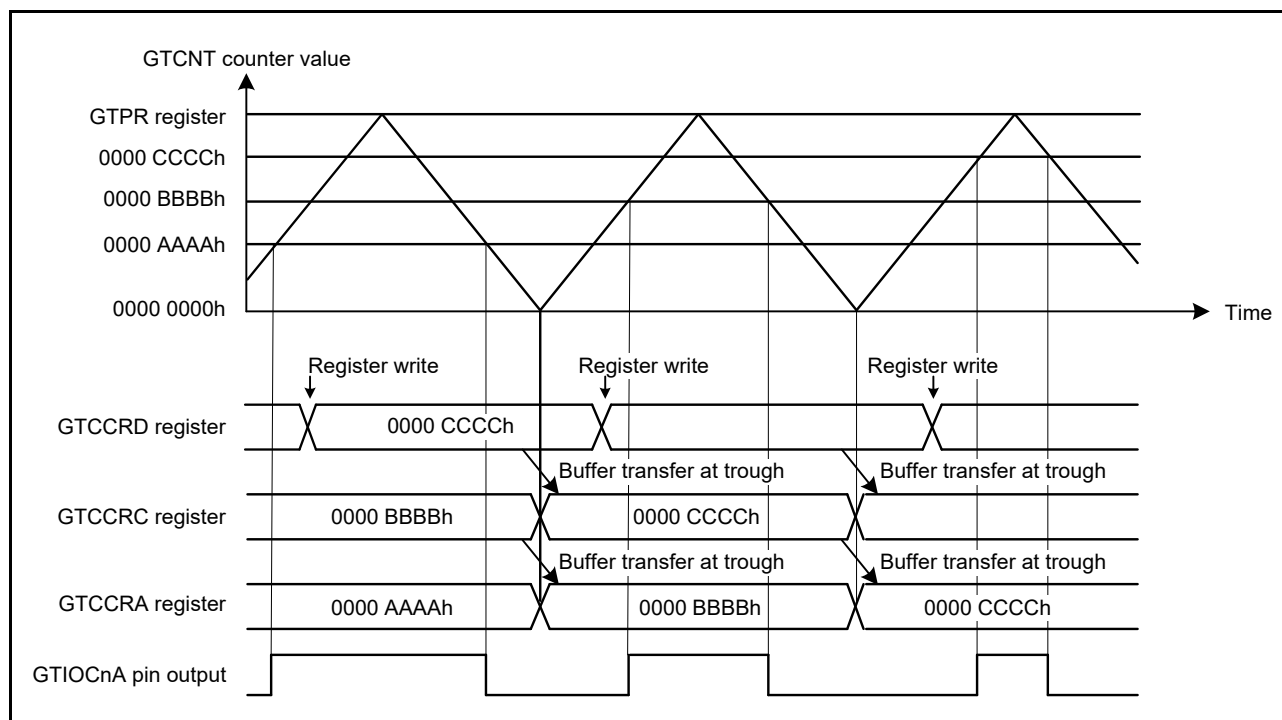


Figure 21.21 Example of GTCCRA and GTCCRB Registers Double Buffer Operation (Output Compare, Triangle Waves, Buffer Operation at Trough, Output Toggled at GTCCRA Register Compare Match, Output Retained at the End of the Cycle) ($n = 0$ to 2)

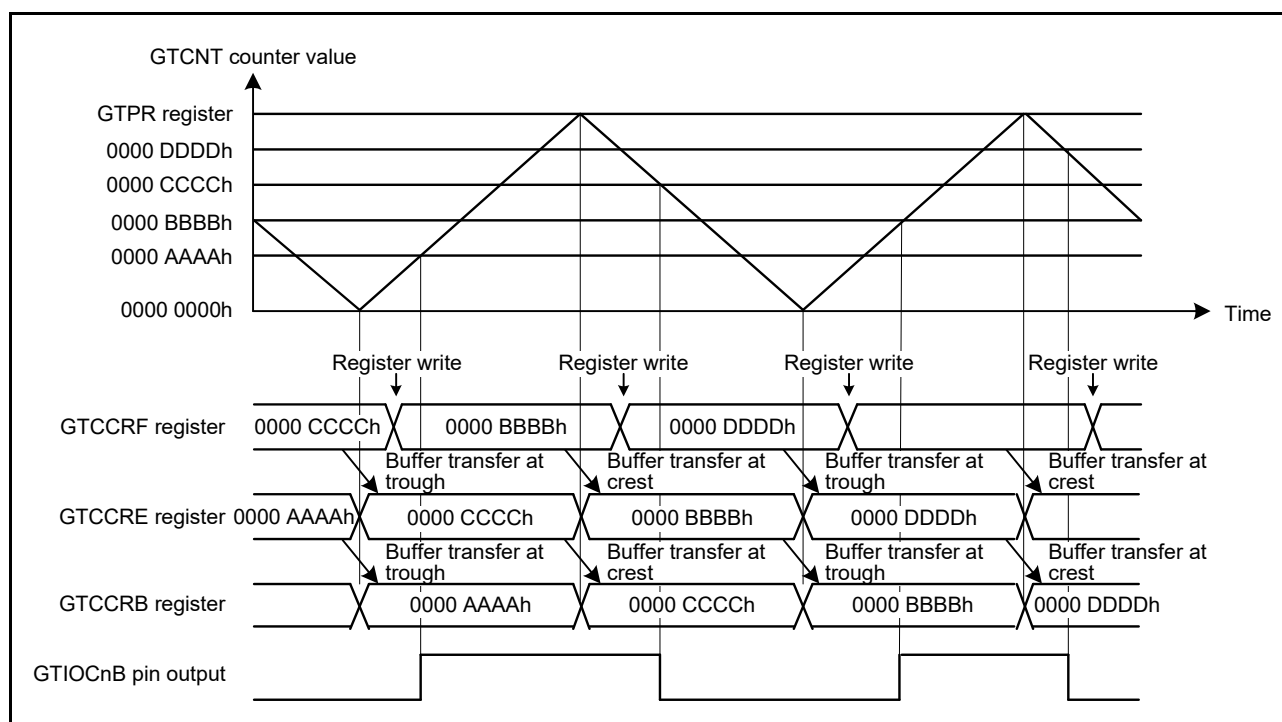


Figure 21.22 Example of GTCCRf and GTCCRB Registers Double Buffer Operation (Output Compare, Triangle Waves, Buffer Operation at Both Troughs and Crests, Output Toggled at GTCCRB Register Compare Match, Output Retained at the End of the Cycle) ($n = 0$ to 2)

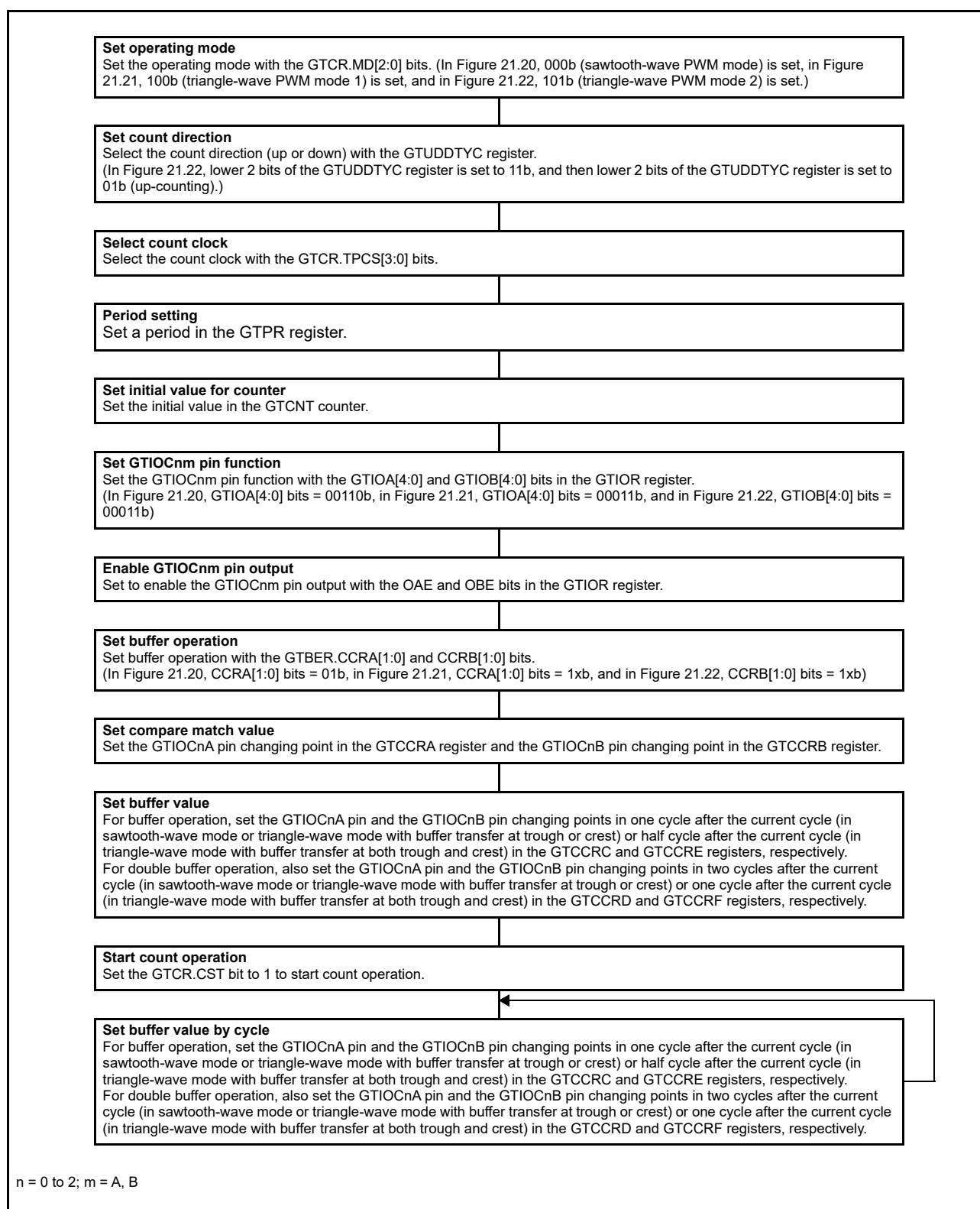


Figure 21.23 Example for Setting Buffer Operation of the GTCCRA and GTCCRB Registers (for Output Compare)

(2) When the GTCCRA or GTCCRB Register Functions as Input Capture Register

The timing of transfer to the buffers is when an input capture is generated. When an input capture is generated, the GTCNT counter value is transferred to GTCCRA and GTCCRB registers and the stored GTCCRA and GTCCRB registers values are transferred to buffer registers. Buffer transfer by counter clearing is not performed at input capture.

Figure 21.24 and Figure 21.25 show examples of buffer operation of the GTCCRA and GTCCRB registers, and Figure 21.26 shows an example for setting buffer operation of the GTCCRA and GTCCRB registers.

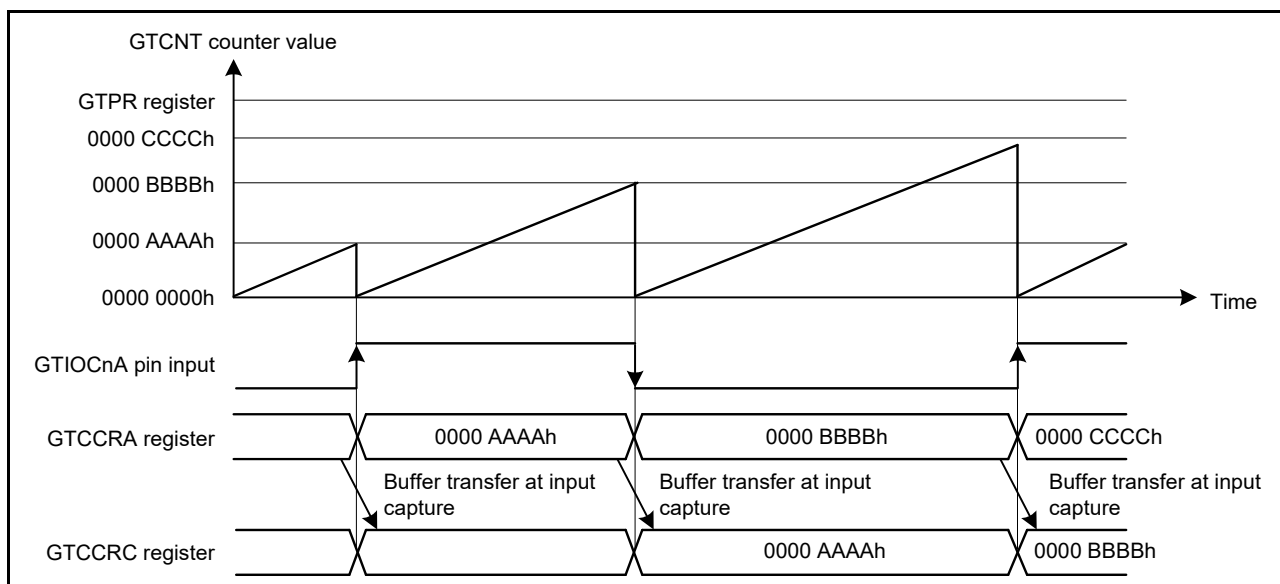


Figure 21.24 Example of Buffer Operation of the GTCCRA and GTCCRB Registers (Counting up to Produce a Sawtooth Wave, and Clearing the GTCNT Counter and Input Capture on Both Edges of the Input on the GTIOcNA Pin) (n = 0 to 2)

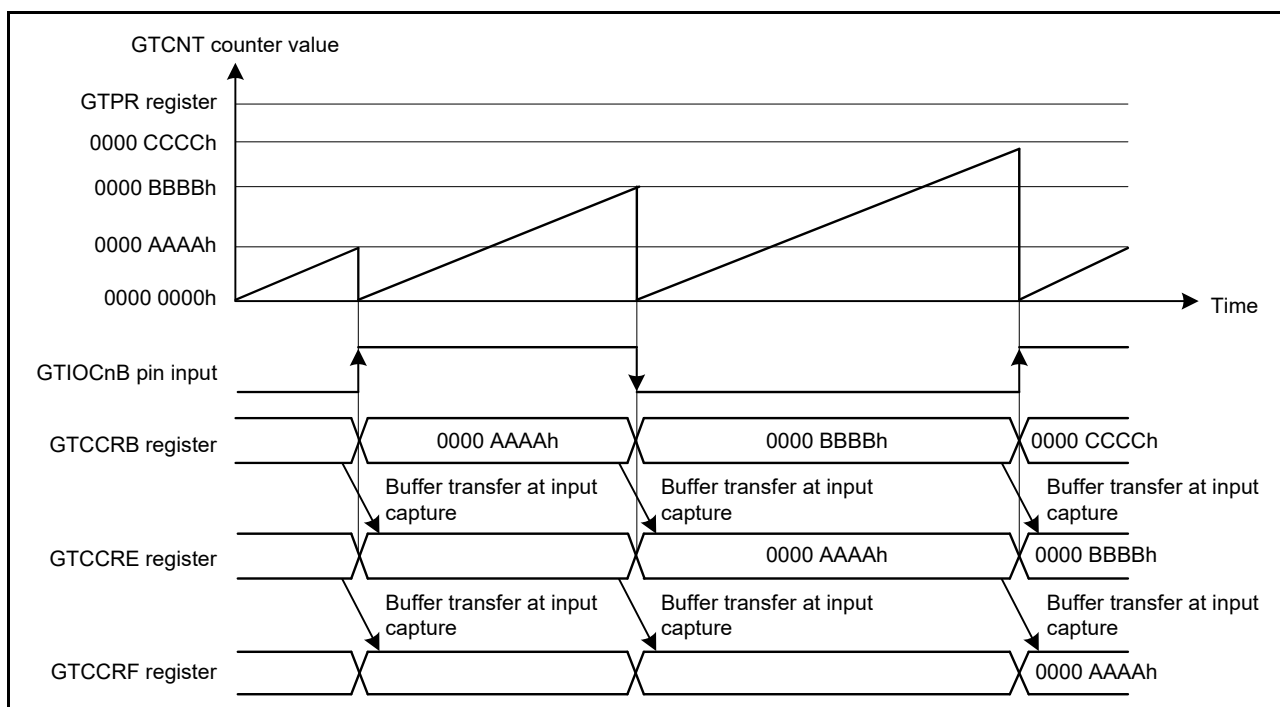


Figure 21.25 Example of Double Buffer Operation of the GTCCRA and GTCCRB Registers (Counting up to Produce a Sawtooth Wave, and Clearing the GTCNT Counter and Input Capture on Both Edges of the Input on the GTIOcNB Pin) (n = 0 to 2)

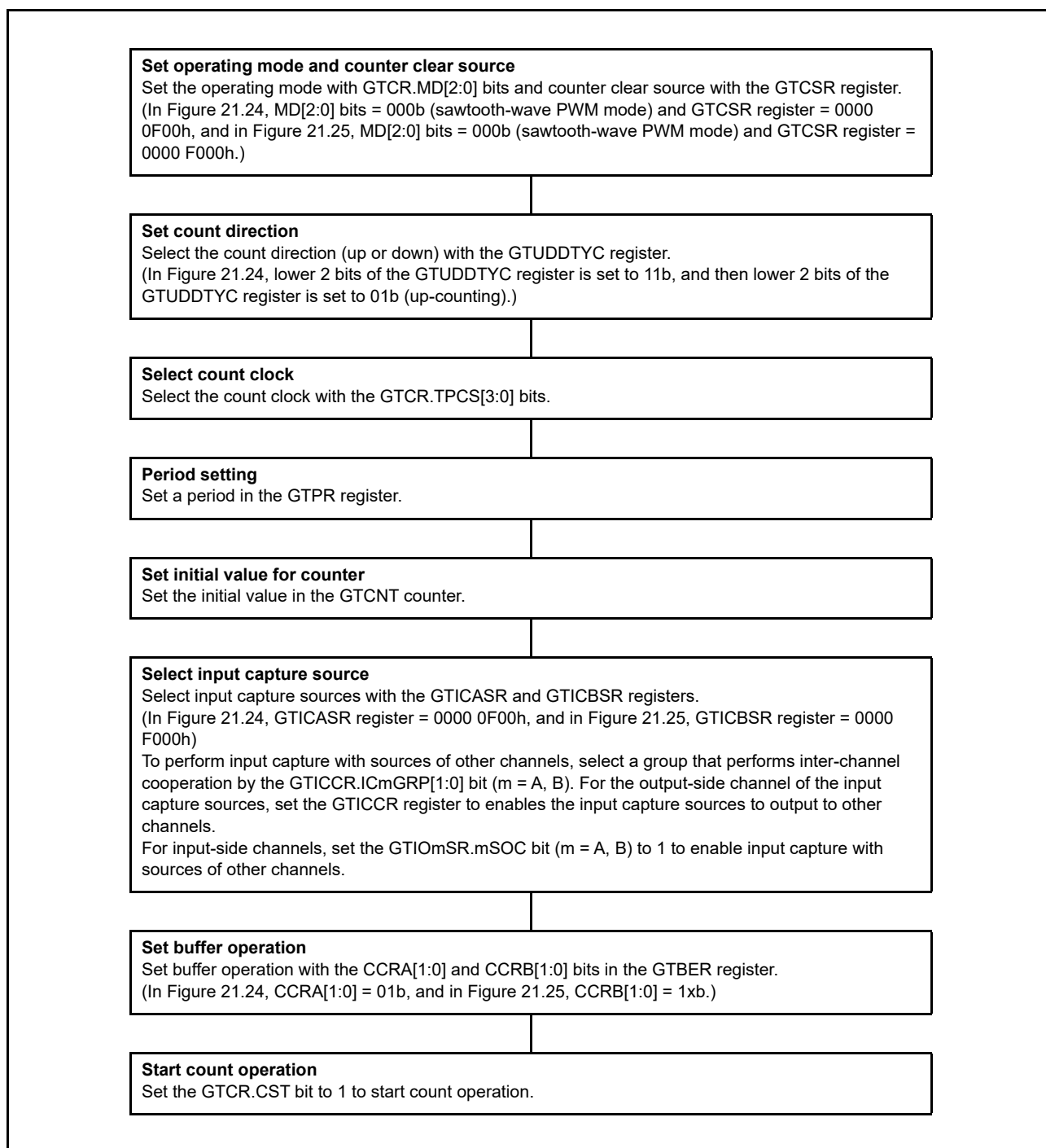


Figure 21.26 Example for Setting Buffer Operation of the GTCCRA and GTCCRB Registers (for Input Capture)

21.3.2.3 Buffer Operation for the GTADTRA and GTADTRB Registers

The GTADTBRA register can function as the GTADTRA buffer register and the GTADTDBRA register can function as the GTADTBRA buffer register (double buffer register for the GTADTRA register). Similarly, the GTADTBRB register can function as the GTADTRB buffer register and the GTADTDBRB register can function as the GTADTBRB buffer register (double buffer register for the GTADTRB register).

To set the GTADTRA or GTADTRB register to function as a double buffer, set the GTBER.ADTDA or ADTDB bit to 1. For single buffer operation, set 0. Not to function as buffer, set the GTBER.ADTTA[1:0] or ADTTB[1:0] bits to 00b. The buffer transfer timing can be set with the ADTTA[1:0] and ADTTB[1:0] bits to an overflow (in up-counting) or an underflow (in down-counting) in sawtooth-wave mode, with ADTTA[1:0] and ADTTB[1:0] bits to 01b for a crest, to 10b for a trough, or to 11b for both crest and trough in triangle-wave mode.

In sawtooth-wave mode, when the ADTTA[1:0] and ADTTB[1:0] bits are set to value other than 00b and in count operation, the buffer transfer, by similar counter clearing sources in section 21.3.2.1, GTPR Register Buffer Operation, is performed in the same way at an overflow (in up-counting) or an underflow (in down-counting).

In sawtooth-wave mode, the buffer transfer of GTADTRm register by counter clearing can be disabled by GTBER2.CCTADm (m = A, B) bit setting.

In sawtooth-wave mode, the buffer transfer of GTADTRm register by its own compare match can be enabled by the GTBER2.CMTADm (m = A, B) bit setting.

Figure 21.27 to Figure 21.30 show examples of buffer operation of the GTADTRA and GTADTRB registers, and Figure 21.31 shows an example for setting buffer operation of the GTADTRA and GTADTRB registers.

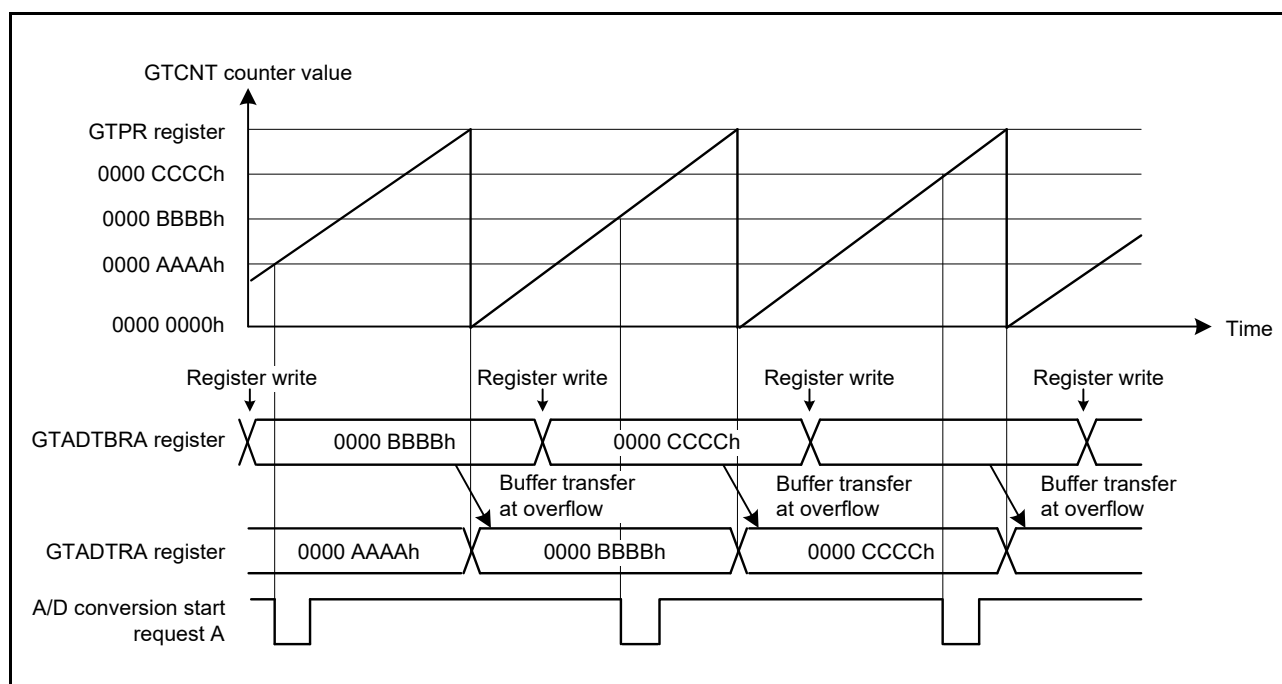


Figure 21.27 Example of Buffer Operation of the GTADTRA and GTADTRB Registers (Sawtooth Waves in Up-Counting, A/D Conversion Start Request Generated by Up-Counting)

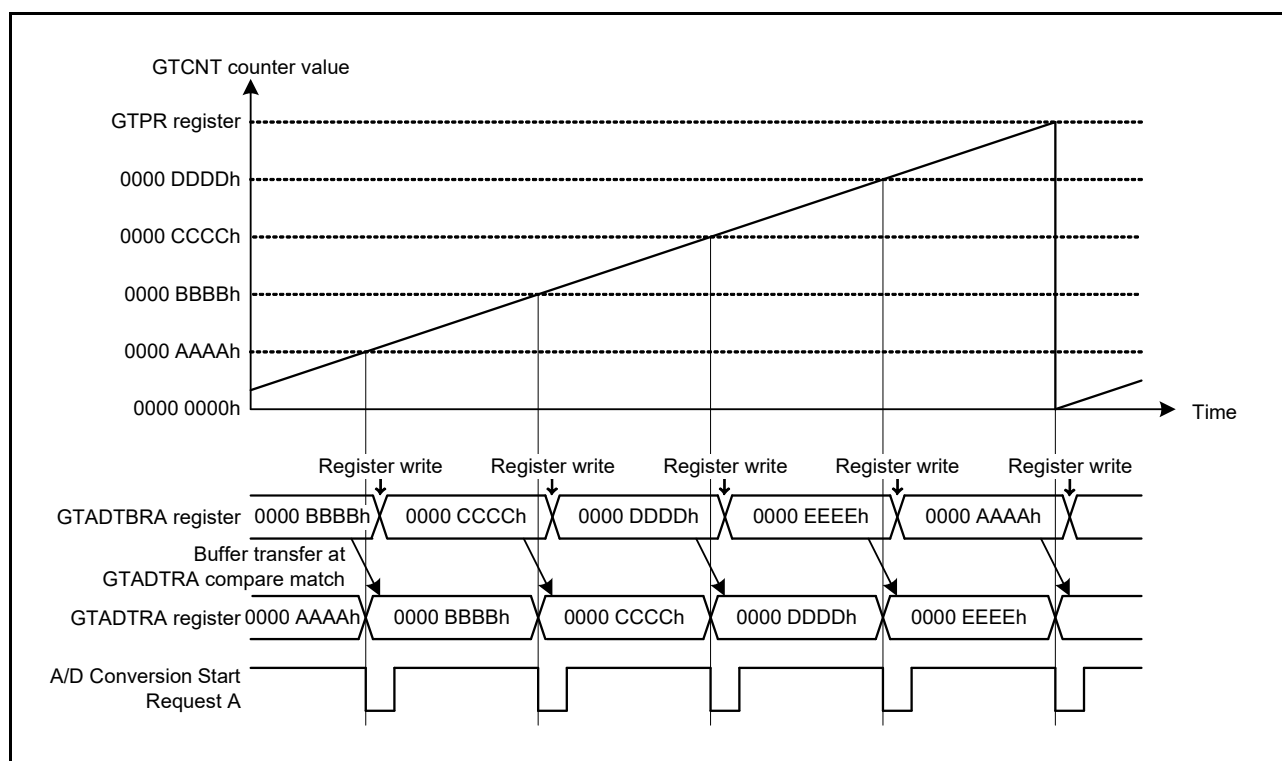


Figure 21.28 Example of Buffer Operation of the GTADTRA and GTADTRB Registers (Sawtooth Waves in Up-Counting, A/D Conversion Start Request Generated by Up-Counting)

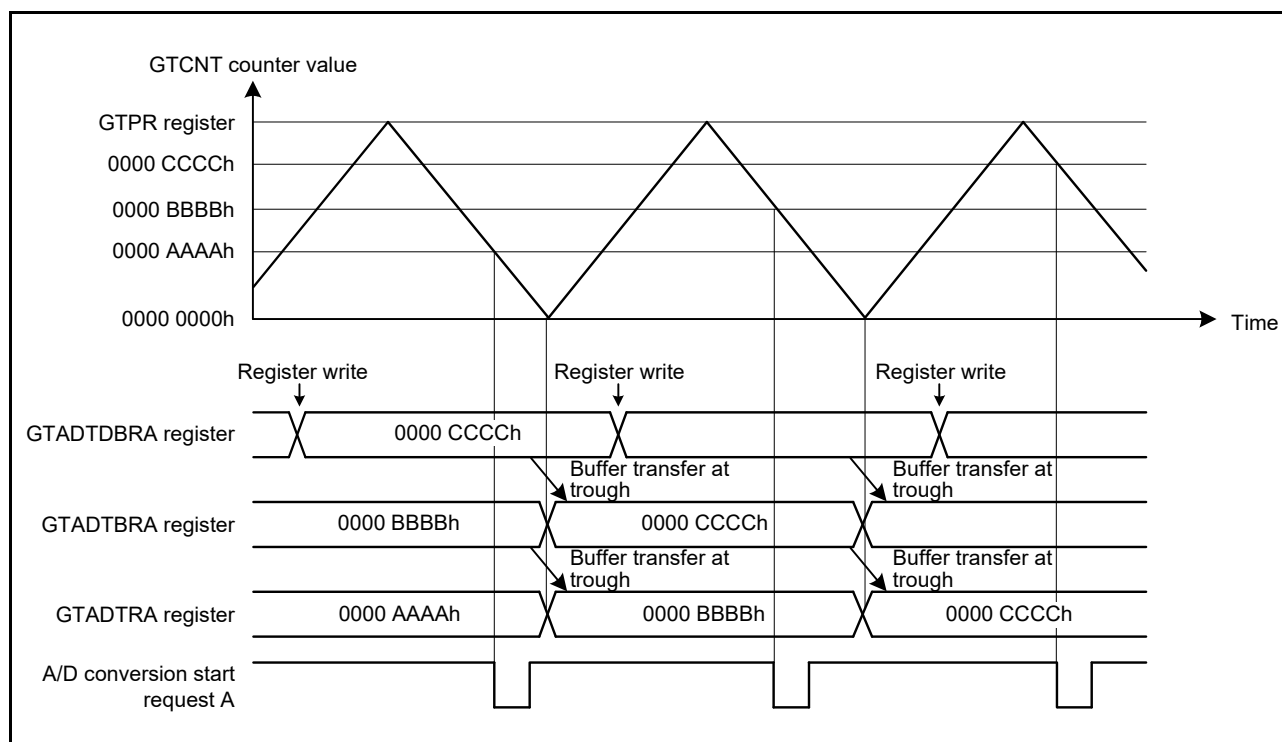


Figure 21.29 Example of Buffer Operation of the GTADTRA and GTADTRB Registers (Triangle Waves, Buffer Operation at Trough, A/D Conversion Start Request Generated by Down-Counting)

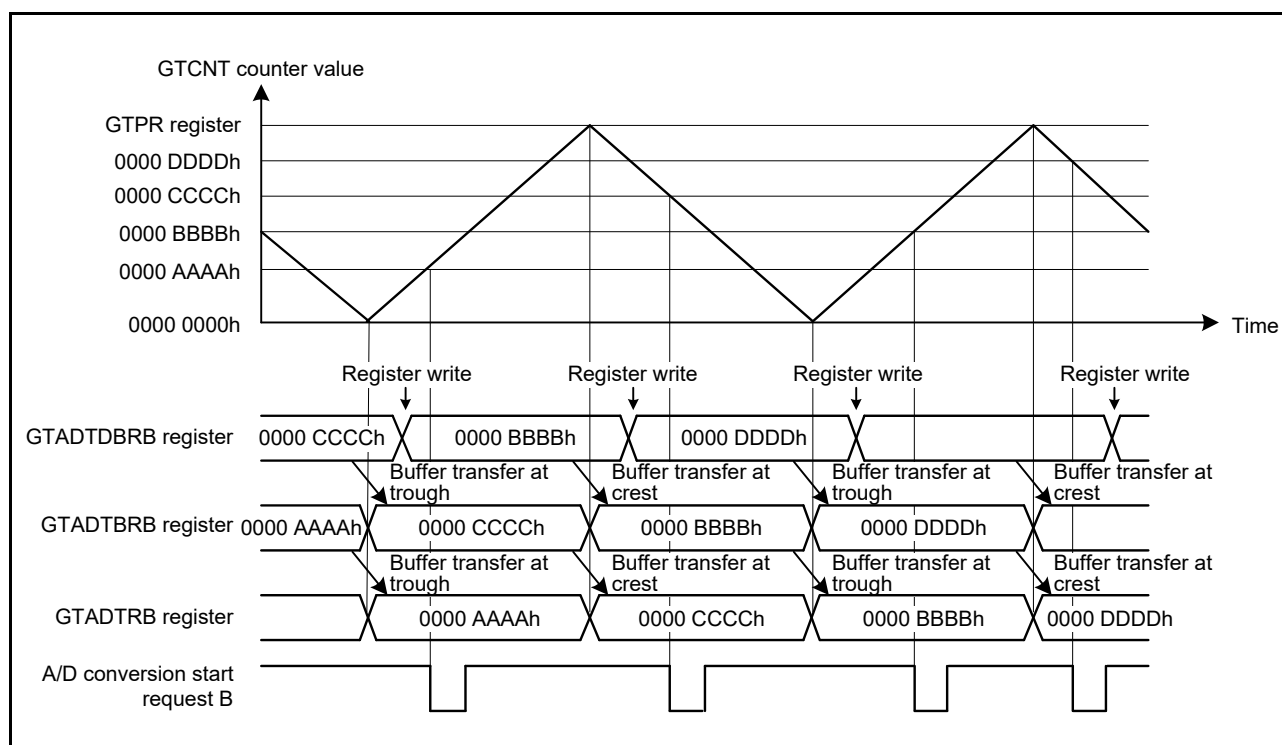


Figure 21.30 Example of Double Buffer Operation of the GTADTRA and GTADTRB Registers (Triangle Waves, Buffer Transfer at Both Troughs and Crests, A/D Conversion Start Request Generated by Both Up- and Down-Counting)

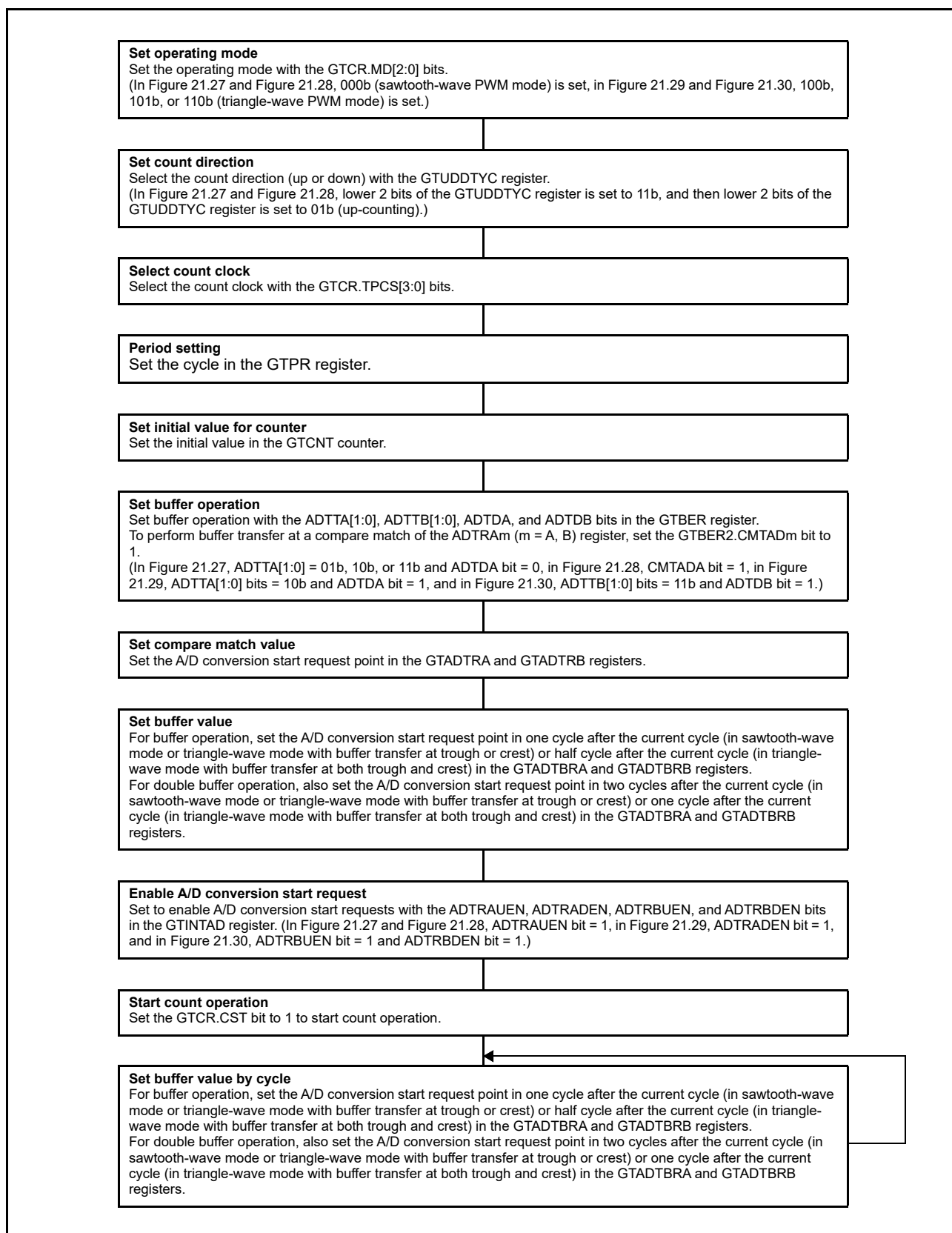


Figure 21.31 Example for Setting Buffer Operation of the GTADTRA and GTADTRB Registers

21.3.3 PWM Output Operating Mode

The GPTW can output PWM waveforms to the GTIOCnA pin or GTIOCnB pin ($n = 0$ to 2) by a compare match between the GTCNT counter and the GTCCRA or GTCCRB register.

By setting the GTDTCR, GTDVU, and GTDVD registers, the compare match value for a negative-phase waveform with dead time can automatically be set to the GTCCRB register.

Sawtooth-wave mode, or triangle-wave mode, the signal synchronized with the PWM cycle can be output from the GTCPP00 output terminal by setting the GTIOR.PSYE bit to 1. The GTCPP00 output is toggled at the end of cycle in sawtooth-wave mode or at the timing of crest/trough/GTCNT counter clearing in triangle-wave mode. The initial output of GTCPP00 is low, and it becomes high when the count starts.

(1) Sawtooth-Wave PWM Mode

In sawtooth-wave PWM mode, the GTCNT counter performs sawtooth-wave (half-wave) operation by setting the period in the GTPR register and a PWM waveform is output to the GTIOCnA or GTIOCnB pin when a GTCCRA or GTCCRB register compare match occurs. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the end of the cycle according to the GTIOR register setting.

The timing of the end of cycle and the timing of the GTCCRm register compare match are the same time, the output pin performs along the PWM output setting for the end of cycle set by the GTIOR.GTIOM[3:2] bits.

Figure 21.32 shows an example of sawtooth-wave PWM mode operation, and Figure 21.33 shows an example for setting sawtooth-wave PWM mode.

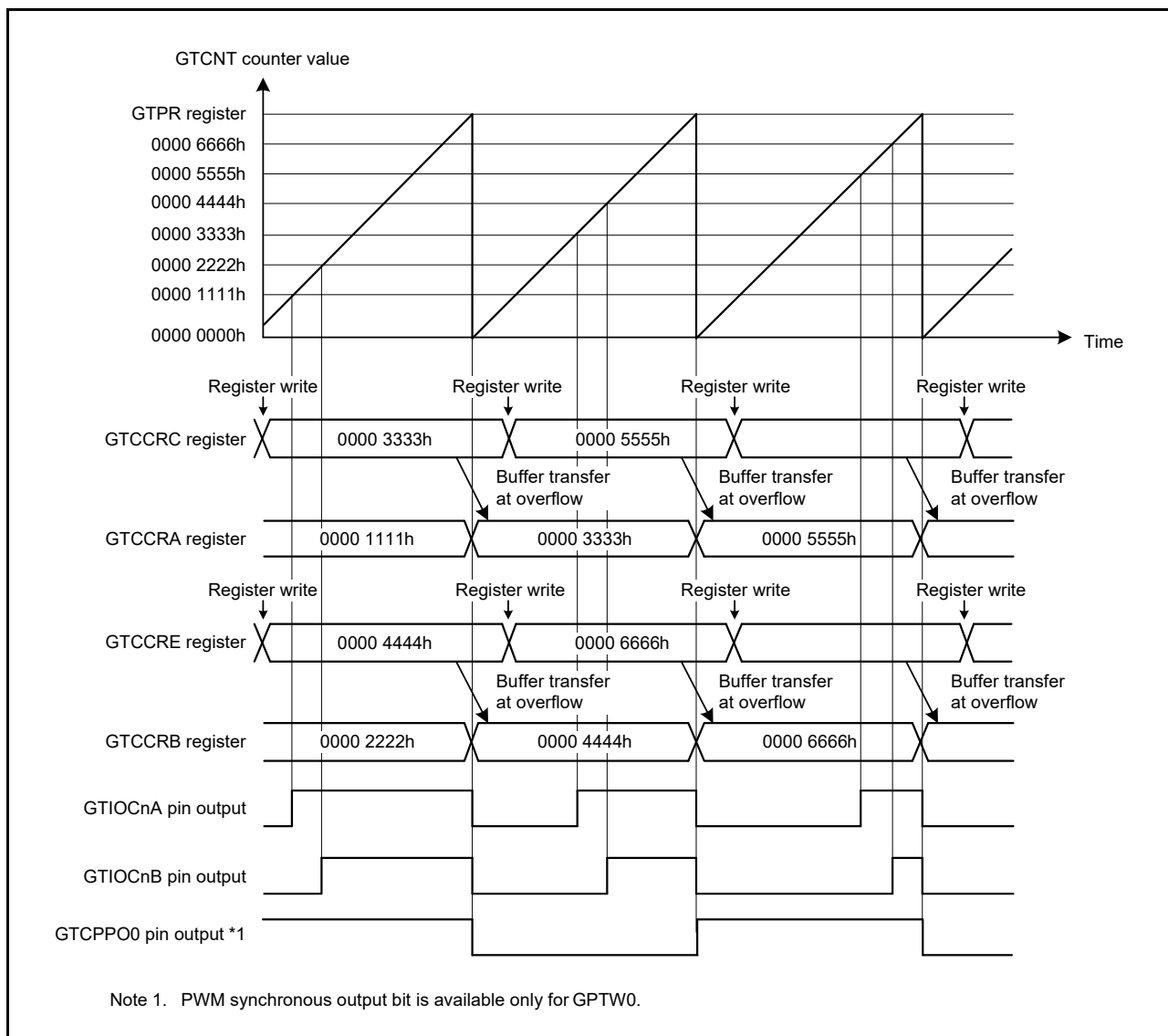


Figure 21.32 Example of Sawtooth-Wave PWM Mode Operation (Up-Counting, Buffer Operation, High Output on Compare Match between the GTCCRA and GTCCRB Registers, Low Output at the End of the Cycle ($n = 0$ to 2), GTIOR.PSYE bit = 1 in GPTW0)

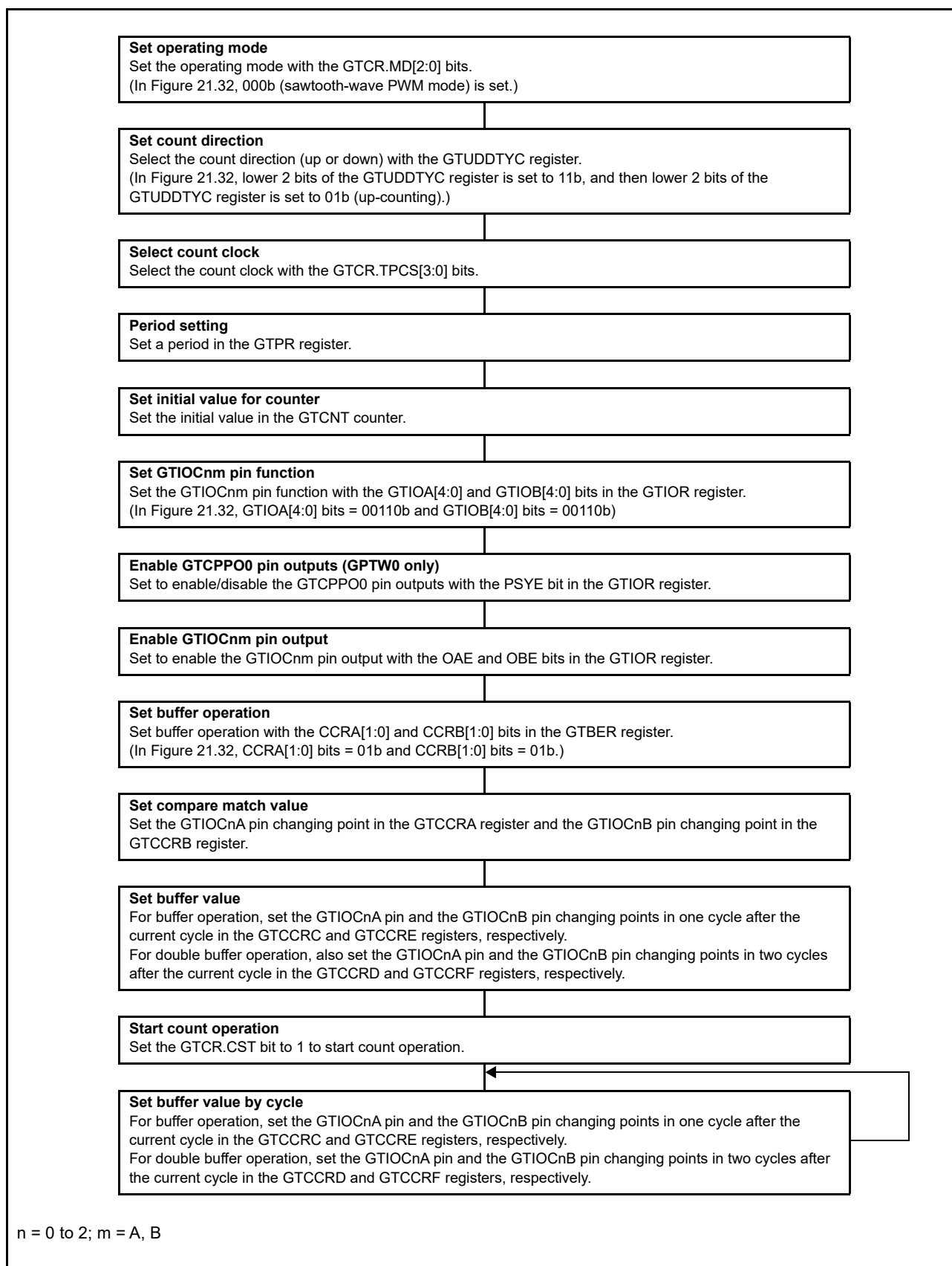


Figure 21.33 Example for Setting Sawtooth-Wave PWM Mode

(2) Sawtooth-Wave One-Shot Pulse Mode

The sawtooth-wave one-shot pulse mode is a mode in which the period is set in the GTPR register, the GTCNT counter performs sawtooth-wave (half-wave) operation and a PWM waveform is output to the GTIOCnA or GTIOCnB pin at a compare match of the GTCCRA or GTCCRB register with buffer operation fixed ($n = 0$ to 2).

Buffer operation in sawtooth-wave one-shot pulse mode is different from the usual buffer operation. Buffer transfer is performed from the GTCCRC register to the GTCCRA register, from the GTCCRE register to the GTCCRB register, from the GTCCRD register to the temporary register A, and from the GTCCRF register to the temporary register B at the end of the cycle, and from the temporary register A to the GTCCRA register at a GTCCRA register compare match and from the temporary register B to the GTCCRB register at a GTCCRB register compare match. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and the end of the cycle according to the GTIOR register setting.

The temporary register A and the temporary register B can perform forcible buffer transfers from the GTCCRD register to the temporary register A and from the GTCCRF register to the temporary register B respectively by writing 1 to the GTBER.CCRSWT bit while the count is stopped.

By setting the GTDTCR, GTDVU, and GTDVD registers, a compare match value for a negative-phase waveform with dead time can automatically be set to the GTCCRB register.

When the GTBER.DBRTCEm ($m = A, B$) bit is set to 1, transfer from an intermediate buffer to the GTCCRm ($m = A, B$) register is repeated on a cyclic basis with using the temporary register x ($x = C, E$) and temporary register m ($m = A, B$) which operate as intermediate buffers for the GTCCRx ($x = C, E$) and GTCCRm ($m = A, B$) registers, respectively, even while buffer transfer is disabled (repeated double buffer operation function during disabling of buffer transfer). For details, refer to section 21.7.2.2, Repeated Double-Buffered Operation When Disabling GTCCR Buffer Transfer.

Figure 21.34 shows an example of sawtooth-wave one-shot pulse mode operation, and Figure 21.35 shows an example for setting sawtooth-wave one-shot pulse mode.

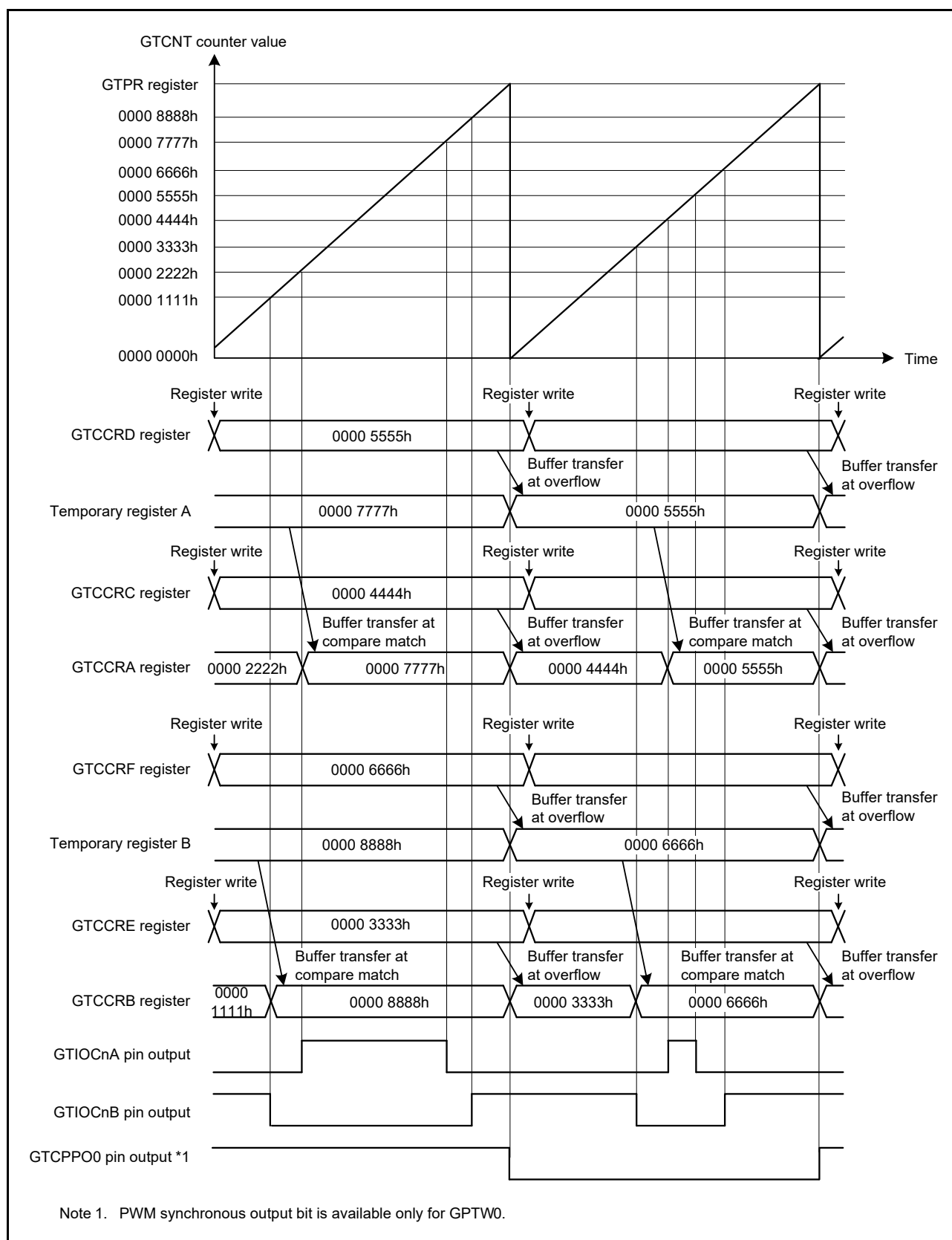


Figure 21.34 Example of Sawtooth-Wave One-Shot Pulse Mode Operation
 (Up-Counting, Low Output from the GTIOcNA Pin and High Output from the GTIOcNB Pin at Initial Output, Output Toggled at GTCCRA/GTCCRB Register Compare Match, Output Retained at the End of the Cycle ($n = 0$ to 2), GTIOR.PSYE bit = 1 in GPTW0)

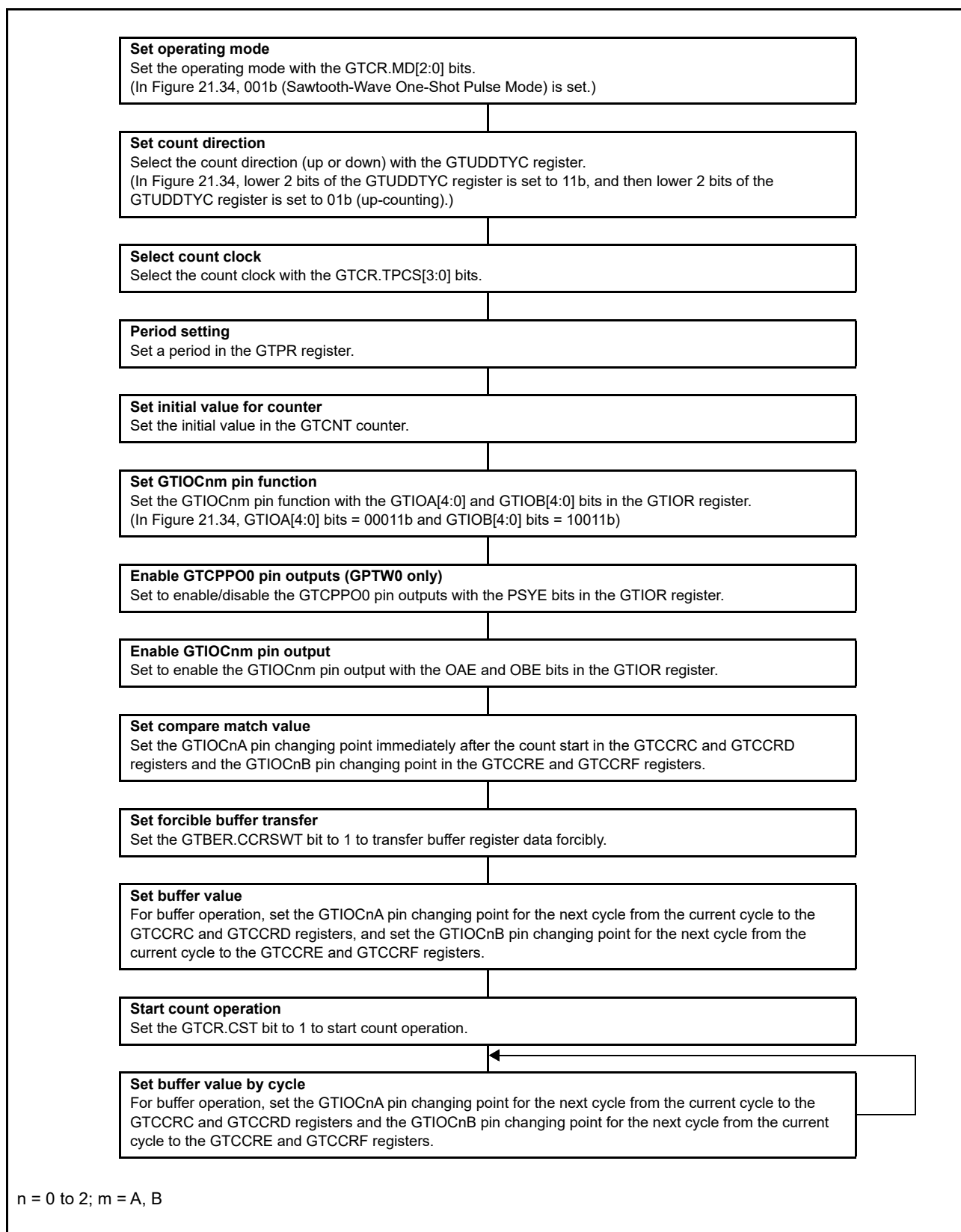


Figure 21.35 Example for Setting Sawtooth-Wave One-Shot Pulse Mode

(3) Triangle-Wave PWM Mode 1 (32-Bit Transfer at Trough)

The triangle-wave PWM mode 1 is a mode in which the period is set in the GTPR register, the GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCnA or GTIOCnB pin when a GTCCRA or GTCCRB register compare match occurs ($n = 0$ to 2).

Buffer transfer is performed at the trough. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the end of the cycle according to the GTIOR register setting.

By setting the GTDTCR, GTDVU, and GTDVD registers, a compare match value for a negative-phase waveform with dead time can automatically be set to the GTCCRB register.

Figure 21.36 shows an example of triangle-wave PWM mode 1 operation, and Figure 21.37 shows an example for setting triangle-wave PWM mode 1.

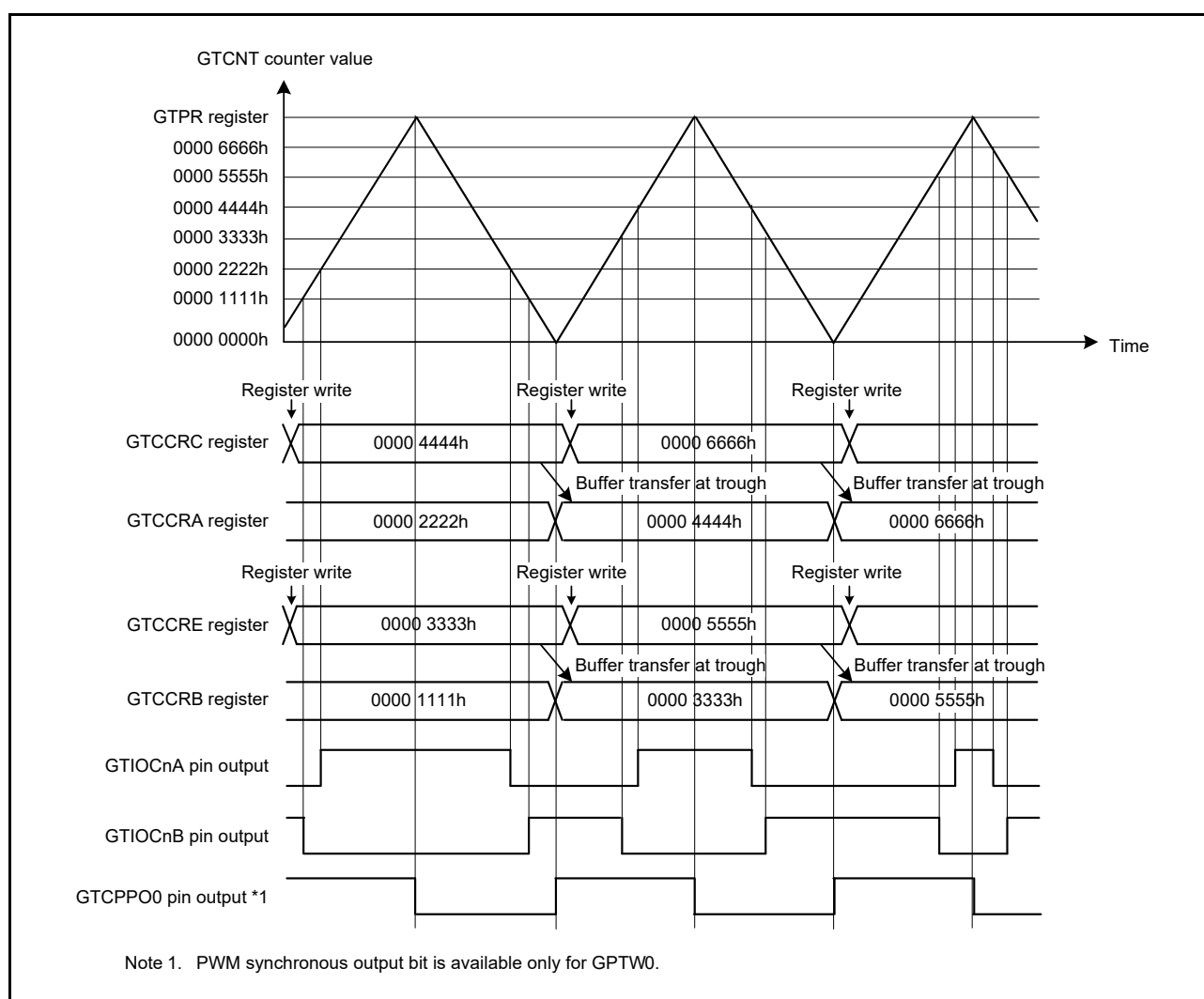


Figure 21.36 Example of Triangle-Wave PWM Mode 1 Operation
 (Buffer Operation, Low Output from the GTIOCnA Pin and High Output from the GTIOCnB Pin at Initial Output, Output Toggled at GTCCRA/GTCCRB Register Compare Match, Output Retained at the End of the Cycle ($n = 0$ to 2), GTIOR.PSYE bit = 1 in GPTW0)

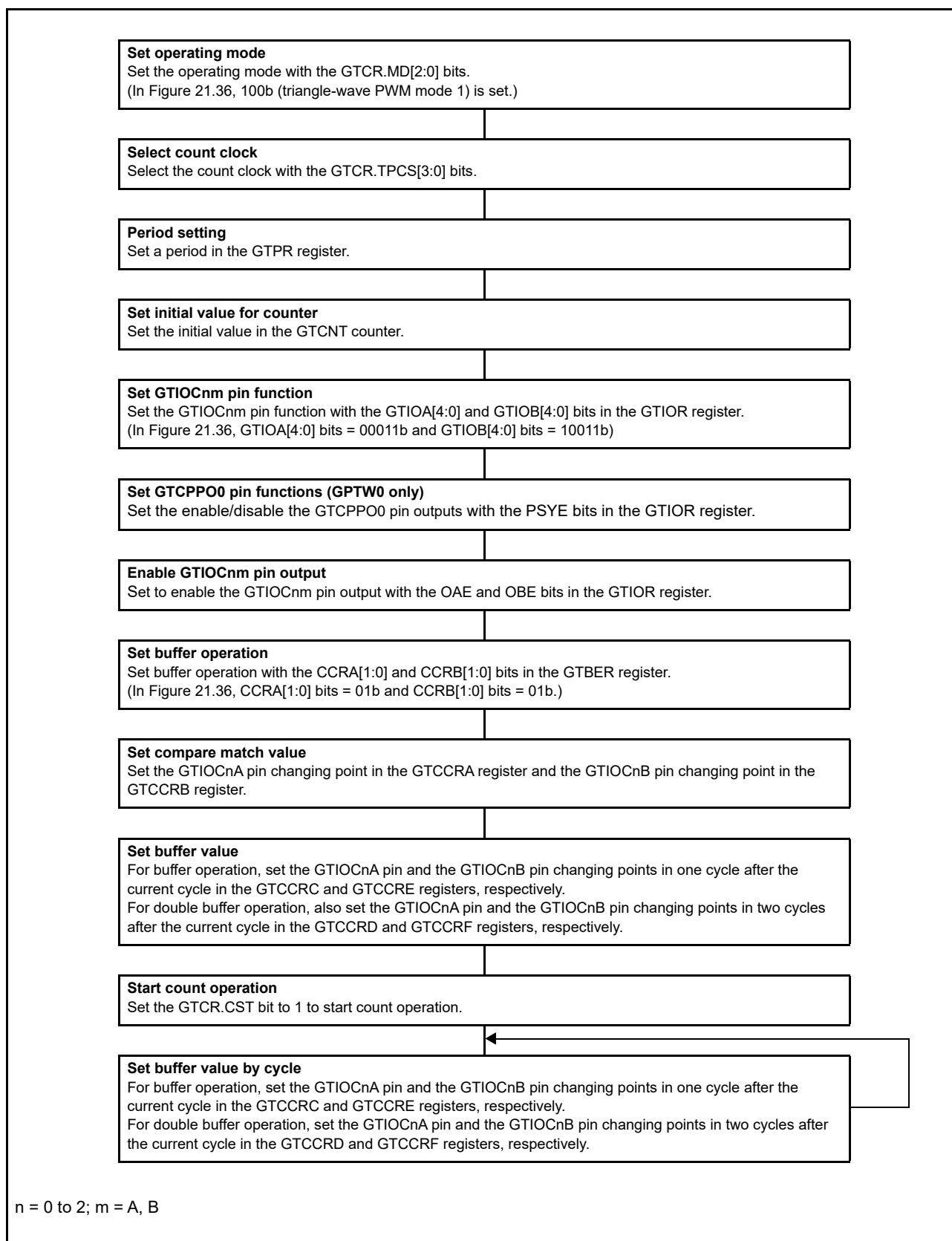


Figure 21.37 Example for Setting Triangle-Wave PWM Mode 1

(4) Triangle-Wave PWM Mode 2 (32-Bit Transfer at Crest and Trough)

Similarly to triangle-wave PWM mode 1, in triangle-wave PWM mode 2 the period is set in the GTPR register, the GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCnA or GTIOCnB pin when a GTCCRA or GTCCRB register compare match occurs ($n = 0$ to 2). The buffer transfer is performed at both crests and troughs. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the end of the cycle according to the GTIOR register setting.

By setting the GTDTCR, GTDVU, and GTDVD registers, a compare match value for a negative-phase waveform with dead time can automatically be set to the GTCCRB register.

Figure 21.38 shows an example of triangle-wave PWM mode 2 operation, and Figure 21.39 shows an example for setting triangle-wave PWM mode 2.

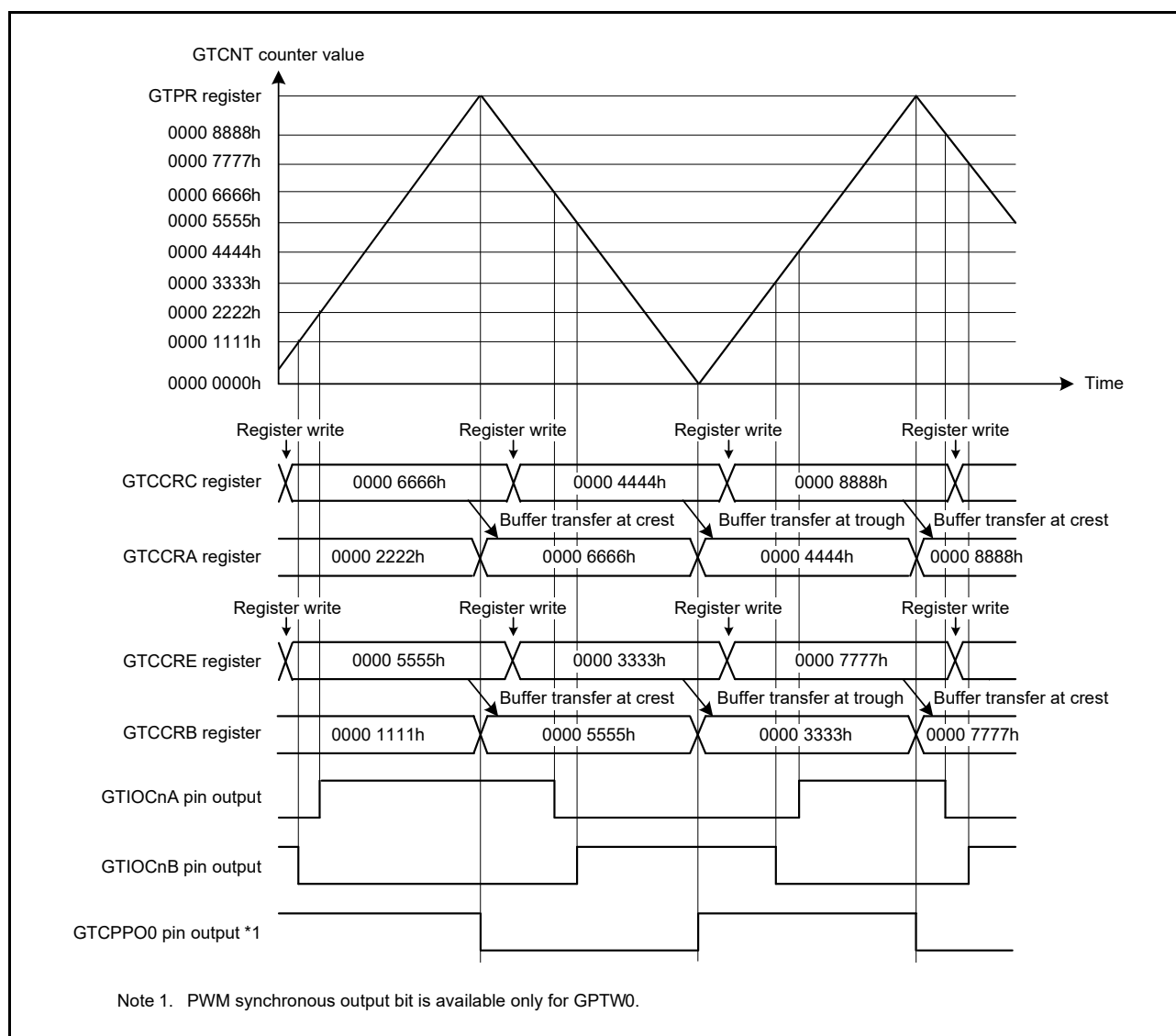


Figure 21.38 Example of Triangle-Wave PWM Mode 2 Operation
(Buffer Operation, Low Output from the GTIOCnA Pin and High Output from the GTIOCnB Pin at Initial Output, Output Toggled at GTCCRA/GTCCRB Register Compare Match, Output Retained at the End of the Cycle ($n = 0$ to 2), GTIOR.PSYE bit = 1 in GPTW0)

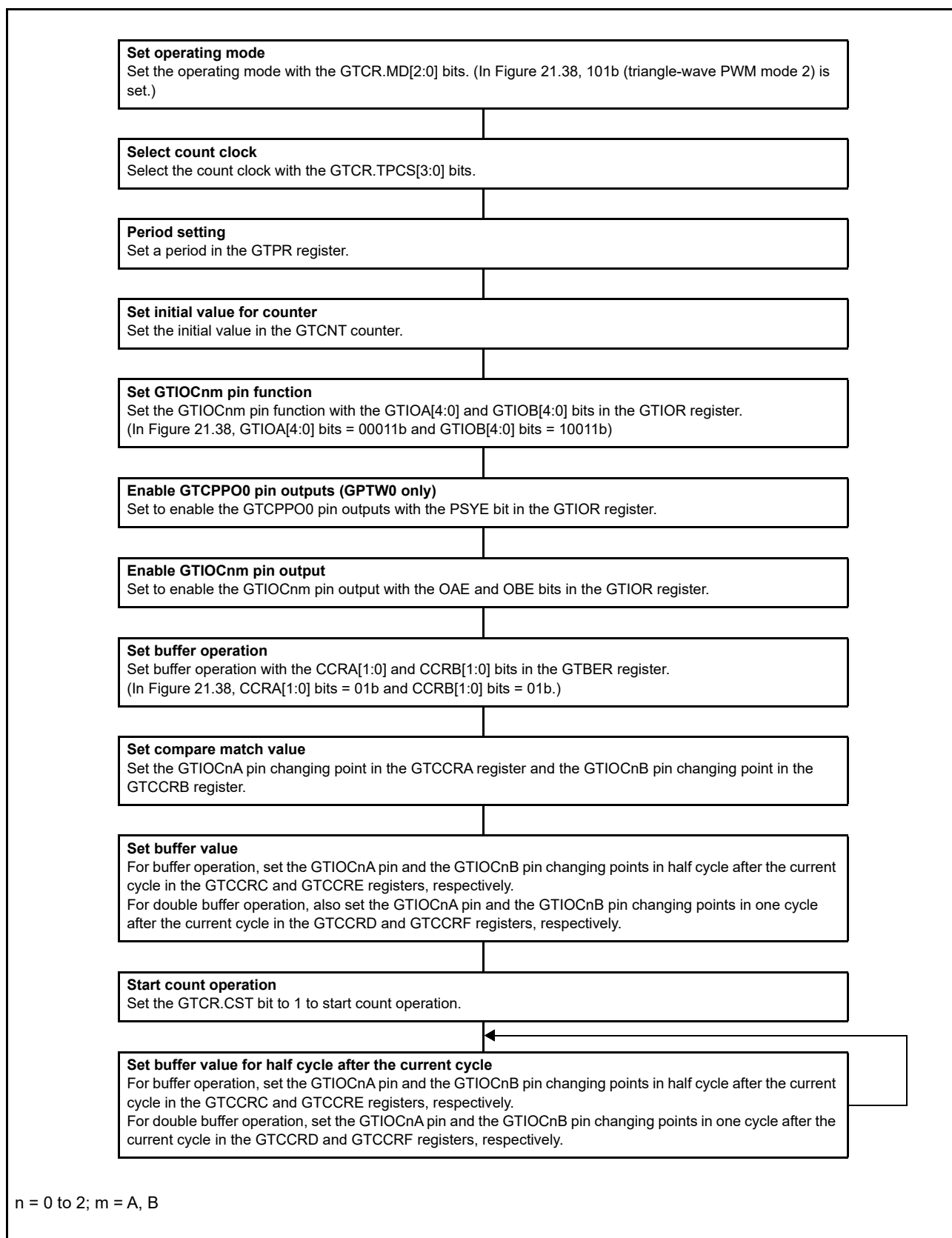


Figure 21.39 Example for Setting Triangle-Wave PWM Mode 2

(5) Triangle-Wave PWM Mode 3 (64-Bit Transfer at Trough)

The triangle-wave PWM mode 3 is a mode in which the period is set in the GTPR register, the GTCNT counter performs triangle-wave (full-wave) operation and a PWM waveform is output to the GTIOCnA or GTIOCnB pin at a compare match of the GTCCRA or GTCCRB register with buffer operation fixed ($n = 0$ to 2). Buffer operation in triangle-wave PWM mode 3 is different from the usual buffer operation. Buffer transfer is performed from the GTCCRC register to the GTCCRA register, from the GTCCRE register to the GTCCRB register, from the GTCCRD register to the temporary register A, and from the GTCCRF register to the temporary register B at the trough, and from the temporary register A to the GTCCRA register and from the temporary register B to the GTCCRB register at the crest. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the end of the cycle according to the GTIOR register setting.

By setting the GTDTCR, GTDVU, and GTDVD registers, a compare match value for a negative-phase waveform with dead time can automatically be set to the GTCCRB register.

When the GTBER.DBRTEC m ($m = A, B$) bit is set to 1, transfer from an intermediate buffer to the GTCCR m ($m = A, B$) register is repeated on a cyclic basis with using the temporary register x ($x = C, E$) and temporary register m ($m = A, B$) which operate as intermediate buffers for the GTCCR x ($x = C, E$) and GTCCR m ($m = A, B$) registers, respectively, even while buffer transfer is disabled (repeated double buffer operation function during disabling of buffer transfer). For details, refer to section 21.7.2.2, Repeated Double-Buffered Operation When Disabling GTCCR Buffer Transfer.

Figure 21.40 shows an example of triangle-wave PWM mode 3 operation, and Figure 21.41 shows an example for setting triangle-wave PWM mode 3.

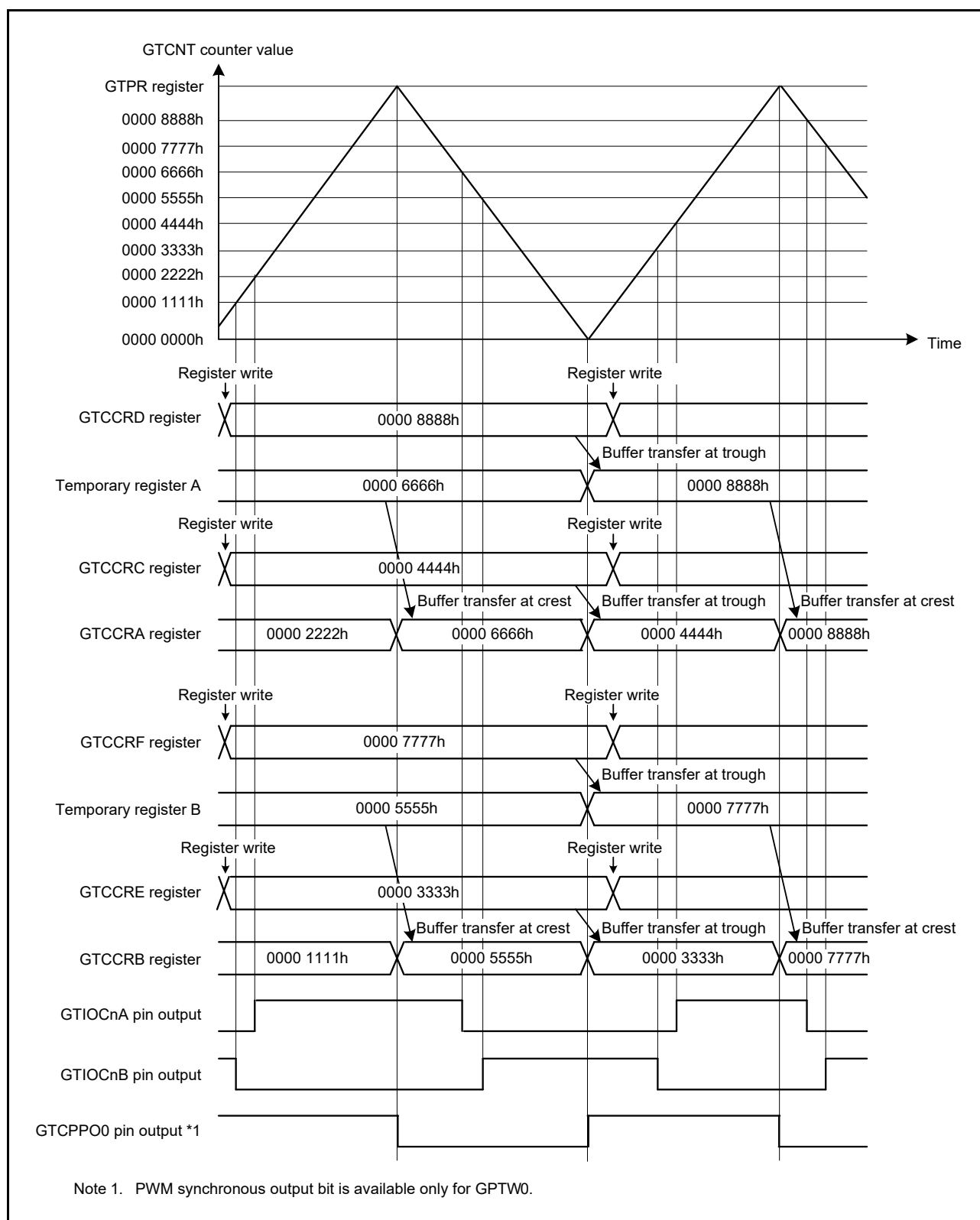


Figure 21.40 Example of Triangle-Wave PWM Mode 3 Operation (Low Output from the GTIOCnA Pin and High Output from the GTIOCnB Pin at Initial Output, Output Toggled at GTCCRA/GTCCRB Register Compare Match, Output Retained at the End of the Cycle (n = 0 to 2), GTIOR.PSYE bit = 1 in GPTW0)

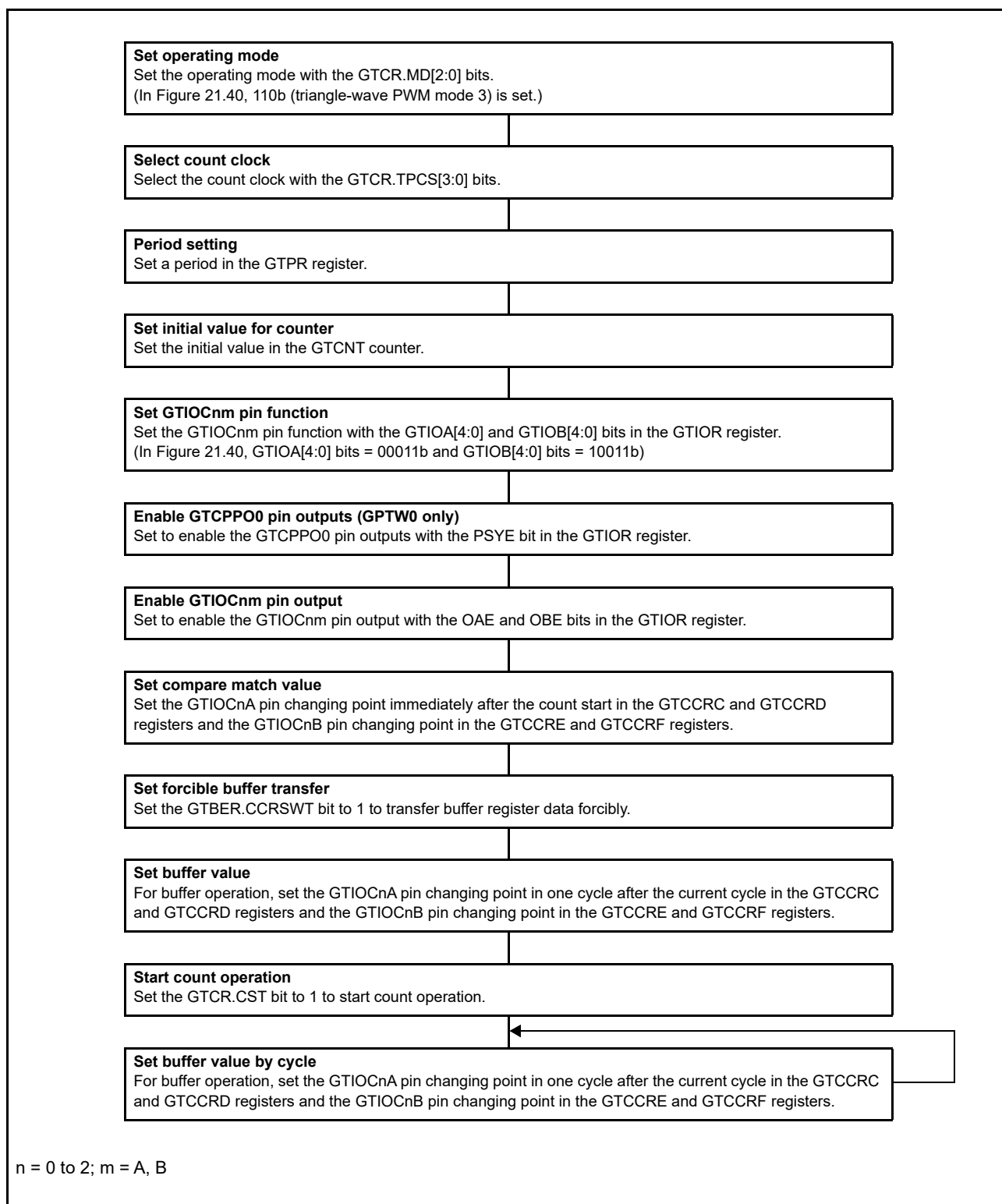


Figure 21.41 Example for Setting Triangle-Wave PWM Mode 3

21.3.4 Automatic Dead Time Setting Function

By setting the GTDTCR register, a compare match value for a negative waveform with dead time obtained by a compare match value for a positive waveform (the GTCCRA register value) and specified dead time values (the GTDVU and GTDVD registers values) can automatically be set to the GTCCRB register.

The automatic dead time setting function can be used in sawtooth-wave one-shot pulse mode and all the triangle PWM modes.

Dead time can be separately set for the first half and second half of a waveform. Dead time for the changing point in the first half of a negative waveform is set in the GTDVU register and that in the second half is set in the GTDVD register. The same dead time can also be set for the first and second halves by setting the GTDTCR.TDFER bit to 1.

The GTDBU register can be used as a buffer register of the GTDVU register, and the GTDBD register can be used as a buffer register of the GTDVD register. Buffer transfer is performed at the end of the cycle (in sawtooth-wave mode: either of an overflow of the GTCNT counter (up-counting), an underflow (down-counting), or the GTCNT counter clearing; in triangle-wave mode: a trough).

The change point of the negative-phase waveform, which is automatically calculated, is obtained by reading the GTCCRB register. Writing to the GTCCRB register is prohibited when the automatic dead time setting function is used.

Do not set the dead-time that makes the change point of the waveform exceeding the count period. When any dead-time setting which would generate a dead-time error is made, adjust the change points of the positive- and negative-phase waveforms to generate waveforms with secured dead-time as shown in Table 21.5. The adjusted change point of the negative-phase waveform is automatically set in the GTCCRB register. An internal signal is used to judge the change point of the positive-phase waveform, thus the value of the GTCCRA register is not updated by the adjusted value.

If the order of the change point becomes inconsistent by adjustment of the waveform change point due to occurrence of dead time errors, or if the change point exceeds the count period even after the adjustment, the complementary relation between the positive- and negative-phases cannot be guaranteed.

If dead-time exceeds the count period by setting 0000 0000h or a value greater than or equal to the setting value of the GTPR register is set in the GTCCRA register in triangle-wave PWM mode, output change is controlled by the output protection function (refer to section 21.7.4, Output Protection Function for GTIOcnm Pin Output (n = 0 to 2; m = A, B)). When GTCCRA register is greater than or equal to [GTPR register + GTDVm register], [GTPR register – 1] is set in the GTCCRB register as the upper limit. (m = U, D)

Automatic setting for a dead time value to the GTCCRB register is performed at the next count clock after the register value for calculating the automatic setting value is updated. In triangle-wave mode, it also can be done at the next count clock from the current crest.

Table 21.5 Adjustment of the Waveform Change Point When a Dead-Time Error Occurs

PWM Output Operating Mode	Count Direction	Period	Condition for Dead-Time Error	Change Point of the Positive-Phase Waveform after Adjustment	Change Point of the Negative-Phase Waveform after Adjustment
Sawtooth-wave one-shot pulse mode	Up-counting	First half	$GTCCRA - GTDVU < 0$	GTDVU	0
		Second half	$GTCCRA + GTDVD > GTPR$ ($GTCCRA + GTDVU > GTPR$)*1	$GTPR - GTDVD$ ($GTPR - GTDVU$)*1	GTPR
	Down-counting	First half	$GTCCRA + GTDVU > GTPR$	$GTPR - GTDVU$	GTPR
		Second half	$GTCCRA - GTDVD < 0$ ($GTCCRA - GTDVU < 0$)*1	GTDVD (GTDVU)*1	0
Triangle-wave PWM mode 1/2/3	Up-counting	(First half)	$GTCCRA - GTDVU \leq 0$	$GTDVU + 1$	1
	Down-counting	(Second half)	$GTCCRA - GTDVD < 0$ ($GTCCRA - GTDVU < 0$)*1	GTDVD (GTDVU)*1	0

Note 1. In the case of GTDTCR.TDFER = 1.

Figure 21.42 to Figure 21.45 show examples of automatic dead time setting function operation. Figure 21.46 and Figure 21.47 show the setting examples.

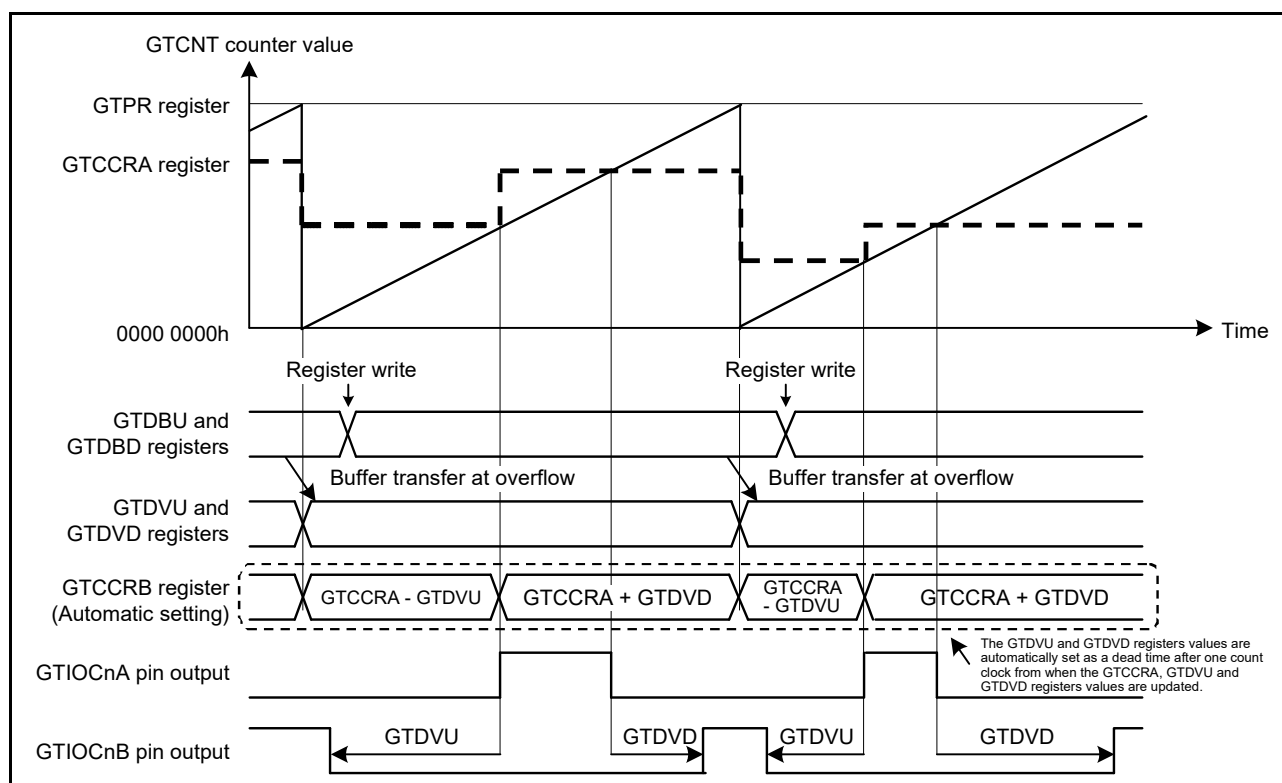


Figure 21.42 Example of Automatic Dead Time Setting Function Operation (Sawtooth-Wave One-Shot Pulse Mode, Up-Counting, GTDVU and GTDVD Registers Set to Buffer Operation, Active Level: High) ($n = 0$ to 2)

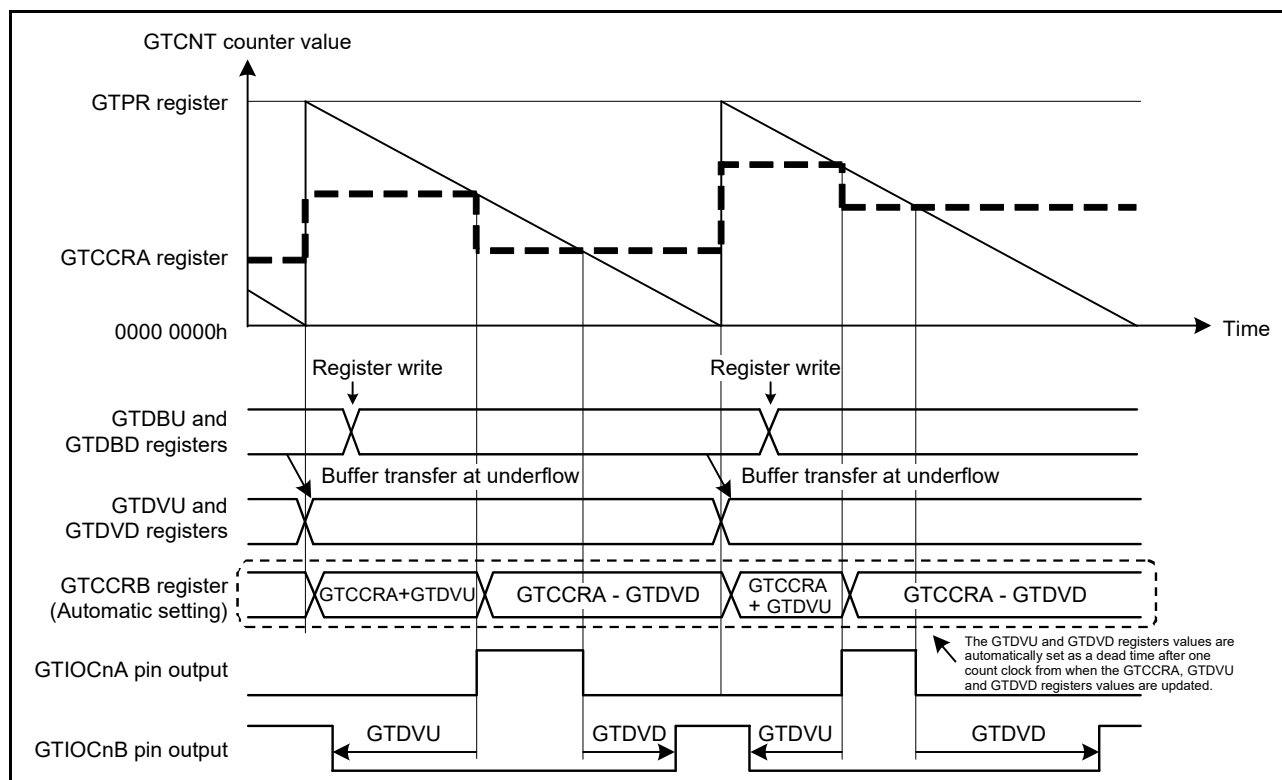


Figure 21.43 Example of Automatic Dead Time Setting Function Operation (Sawtooth-Wave One-Shot Pulse Mode, Down-Counting, GTDVU and GTDVD Registers Set to Buffer Operation, Active Level: High) ($n = 0$ to 2)

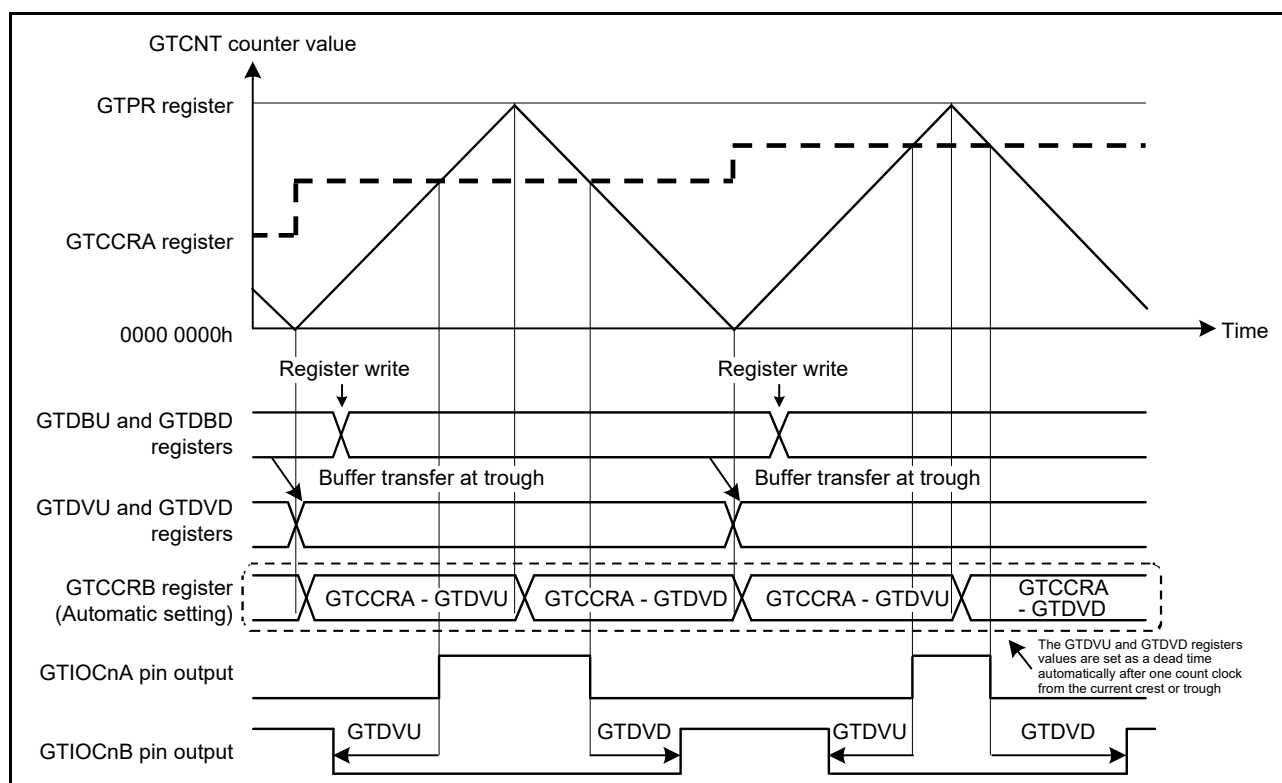


Figure 21.44 Example of Automatic Compare-Match Value Setting Function with Dead Time (Triangle-Wave PWM Mode 1, GTDVU and GTDVD Registers Set to Buffer Operation, Active Level: High) (n = 0 to 2)

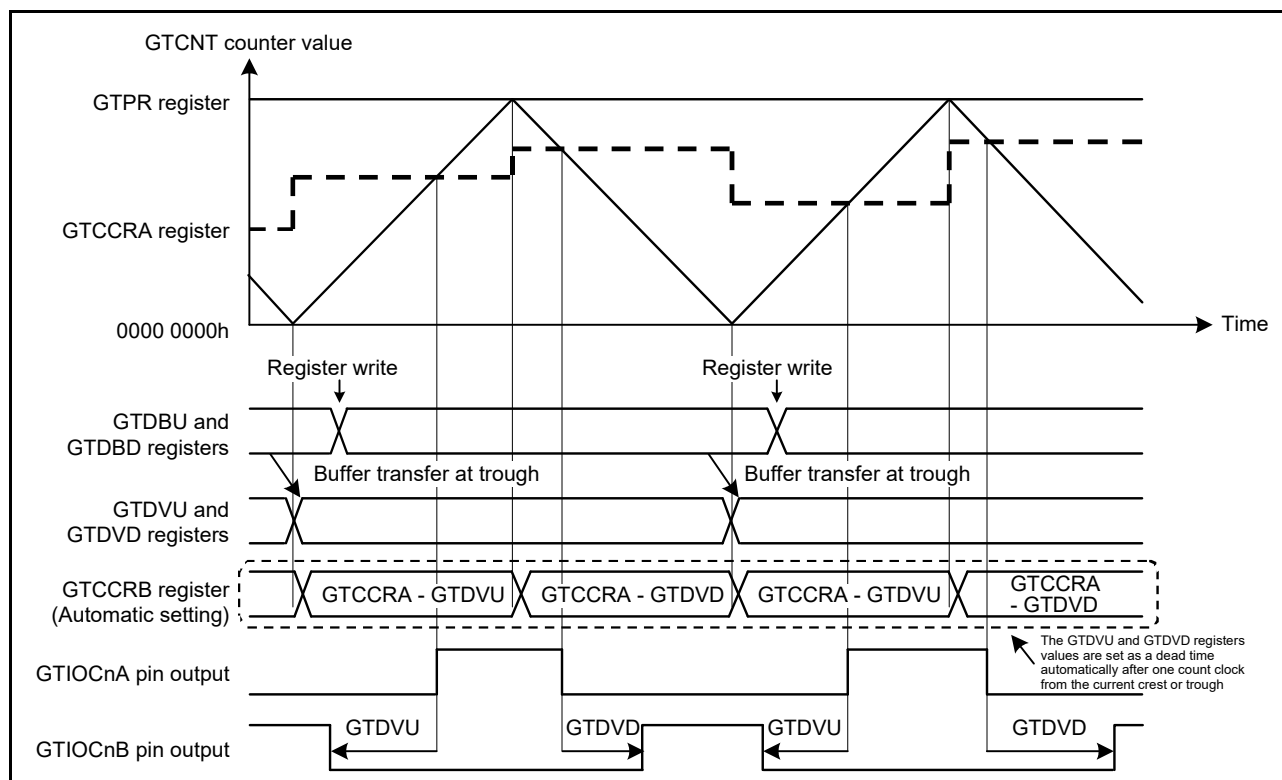


Figure 21.45 Example of Automatic Compare-Match Value Setting Function with Dead Time (Triangle-Wave PWM Mode 2 or 3, GTDVU and GTDVD Registers Set to Buffer Operation, Active Level: High) (n = 0 to 2)

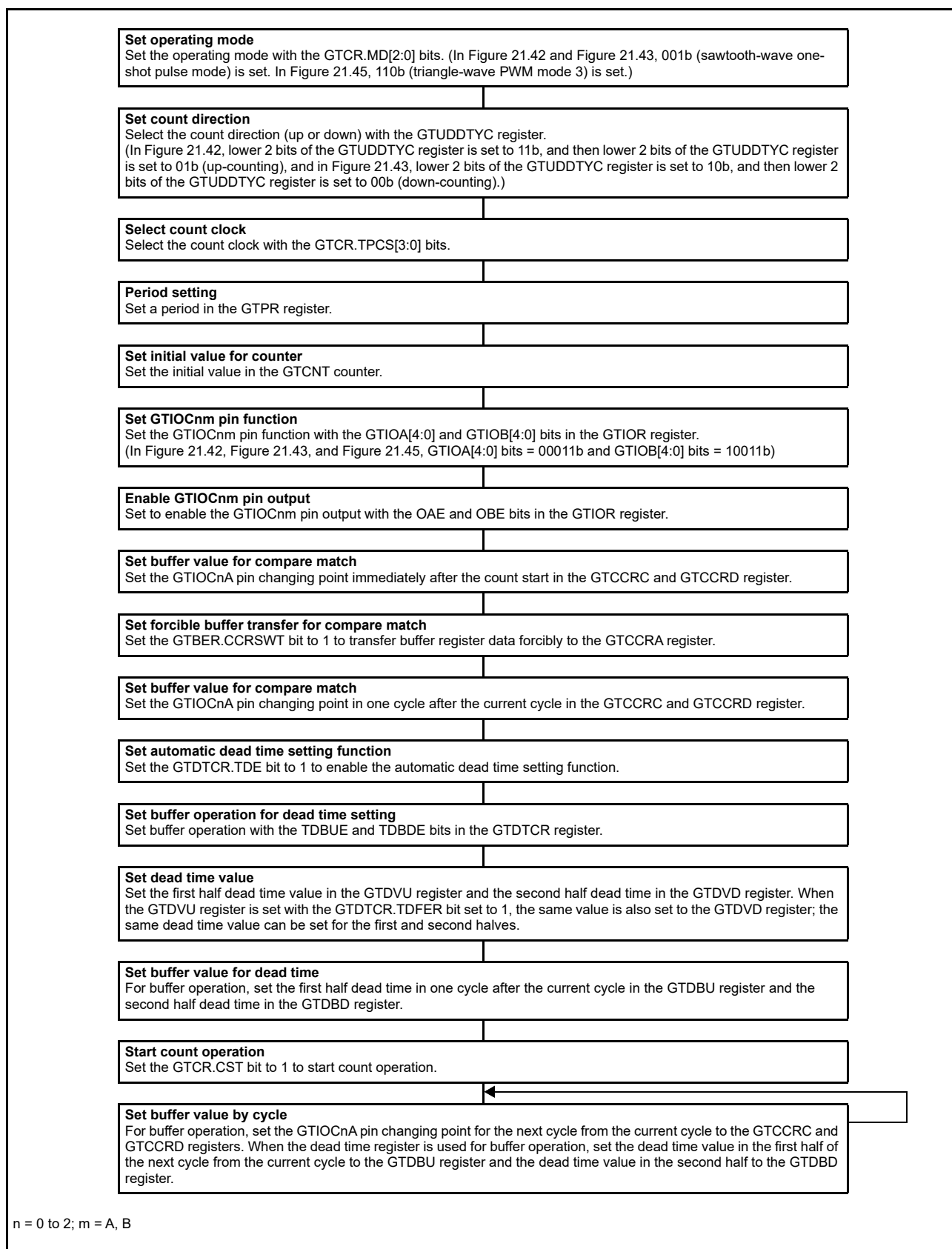


Figure 21.46 Example for Setting Automatic Dead Time Setting Function (Sawtooth-Wave One-Shot Pulse Mode, Triangle-Wave PWM Mode 3)

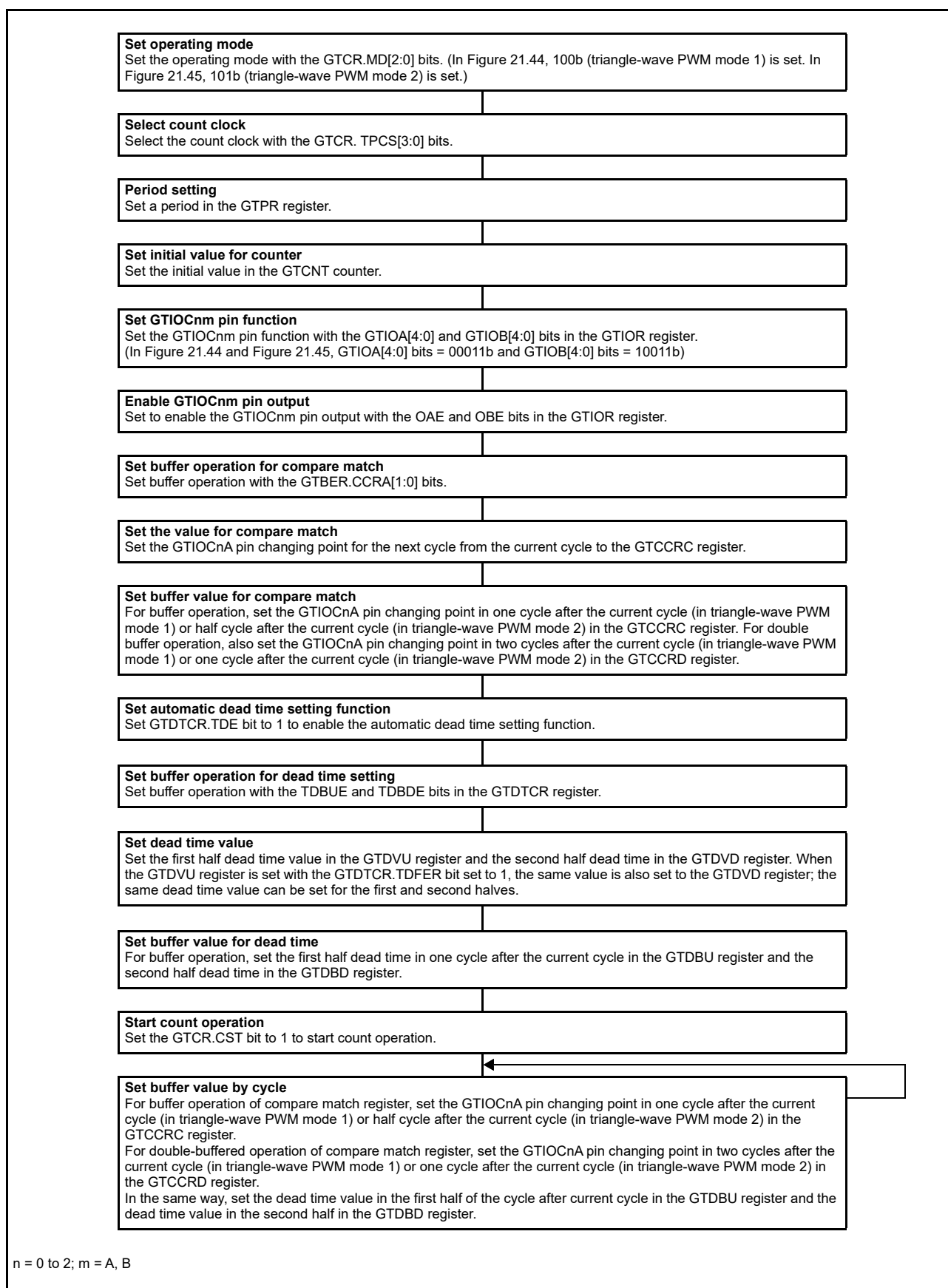


Figure 21.47 Example for Setting Automatic Dead Time Setting Function (Triangle-Wave PWM Mode 1 or 2)

21.3.5 Count Direction Changing Function

The count direction of the GTCNT counter can be changed by modifying the GTUDDTYC.UD bit.

In sawtooth-wave mode, if the UD bit is modified during count operation, the count direction is changed at an overflow (when modified during up-counting) or an underflow (when modified during down-counting). If the UD bit is modified while count operation is stopped and the GTUDDTYC.UDF bit is 0, the UD bit modification is not reflected at the start of counting and the count direction is changed at an overflow or an underflow. If the UDF bit is set to 1 while count operation is stopped, the UD bit value at that time is reflected at the start of counting.

In triangle-wave mode, changing the value of the GTUDDTYC.UD bit during counting does not switch the direction of counting. Likewise, changing the value of the GTUDDTYC.UD bit while the GTUDDTYC.UDF bit is 0 and counting is stopped does not actually update the value of the bit. If 1 is written to the GTUDDTYC.UDF bit while counting is stopped, the value of the GTUDDTYC.UD bit at that time is reflected from the time counting is started.

When counting direction is switched during count operation in sawtooth-wave mode, the value of the GTPR register after start of up-counting is reflected to the count period in up-counting operation, and the value of the GTPR register after the start of down-counting is reflected to the count period in down-counting operation.

Figure 21.48 shows an example of count direction changing function operation.

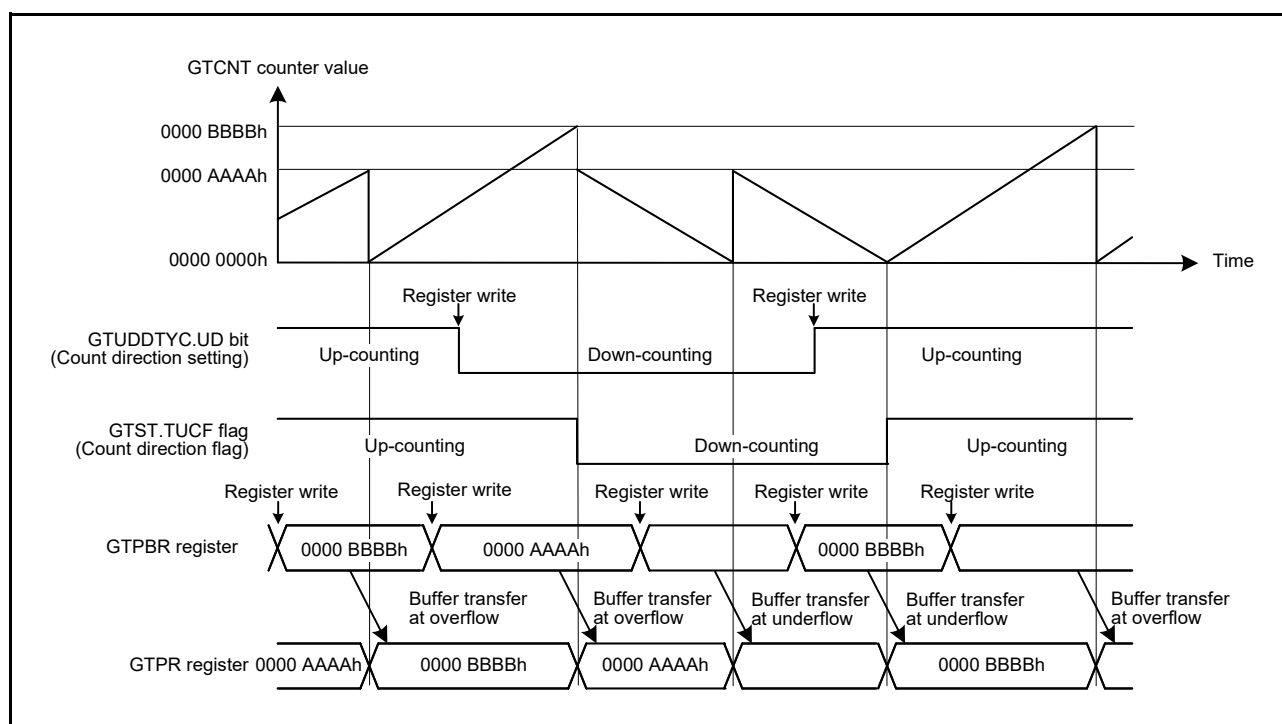


Figure 21.48 Example of Count Direction Changing Function Operation (during Buffer Operation)

21.3.6 Duty Cycle 0%/100% Output Function

Changing the value of the GTUDDTYC.OADTY[1:0] bits and GTUDDTYC.OBDTY[1:0] bits specifies the output duty setting on the GTIOCnA and GTIOCnB pins to 0% or 100%. (n = 0 to 2)

In sawtooth-wave mode, if the value of the OADTY[1:0] bits or OBDTY[1:0] bits is changed during the count operation, the output duty-cycle setting is reflected at an overflow (when changed during up-counting) or an underflow (when changed during down-counting). When the GTUDDTYC.OADTYF bit or GTUDDTYC.OBDTYF bit is 0 while count operation is stopped and the value of the OADTY[1:0] bits or OBDTY[1:0] bits is changed, the output duty-cycle setting changed at the start of counting is not reflected, but the output duty-cycle setting changed at an overflow or underflow is reflected. When the OADTYF bit or OBDTYF bit is 1 while count operation is stopped and the value of the OADTY[1:0] bits or OBDTY[1:0] bits is changed, the value of the OADTY[1:0] bits or OBDTY[1:0] bits at that time is reflected at the start of counting.

In triangle-wave mode, if the value of the OADTY[1:0] bits or OBDTY[1:0] bits is changed during the count operation, When the GTUDDTYC.OABDITYT bit is 0, the output duty setting changed when underflow occurs is reflected, and when the GTUDDTYC.OABDITYT bit is 1, the output duty setting changed when underflow occurs and at the top of the triangle-wave is reflected.

When the OADTYF bit or OBDTYF bit is 0 while count operation is stopped and the value of the OADTY[1:0] bits or OBDTY[1:0] bits is changed, the output duty-cycle setting changed at the start of counting is not reflected, but the output duty-cycle setting changed at an underflow is reflected. When the OADTYF bit or OBDTYF bit is 1 while count operation is stopped and the value of the OADTY[1:0] bits or OBDTY[1:0] bits is changed, the value of the OADTY[1:0] bits or OBDTY[1:0] bits at that time is reflected at the start of counting.

During the operation under the settings of duty cycle for 0%/100%, compare match operation inside the GPTW continues to perform interrupt outputs and buffer operations.

When the output duty setting is changed from 0% or 100% due to a compare match, the GTIOR.GTIOA[3:2] bits and the OADTYR bits determine the level output on the GTIOCnA pin and GTIOR.GTIOB[3:2] bits and OBDTYR bits determine the level output on the GTIOCnB pin at the end of the cycle. When the GTIOA[3:2], GTIOB[3:2] bits are set as 01b, output is driven Low at the end of the cycle, and when set as 10b, output is driven High at the end of the cycle. The output is retained at the end of the cycles when the GTIOA[3:2] or GTIOB[3:2] bits are 00b, and toggled at the end of the cycles when the GTIOA[3:2] or GTIOB[3:2] bits are 11b. A value to be output is selectable in the OADTYR or OBDTYR bit when the corresponding GTIOA[3:2] or GTIOB[3:2] bits are either 00b or 11b. Table 21.6 lists the output values at the end of the cycle when the output setting is changed from duty 0% or 100% to compare match.

Table 21.6 Output Value after Release of Duty 0%/100%

GTIOR.GTIOm[3:2]	Value at Compare Match Output at the End of the Cycle in the Case of Masking by a Duty Cycle 0% or 100%	The GTUDDTYC.OmDITYR Bit at Duty 0% Setting		The GTUDDTYC.OmDITYR Bit at Duty 100% Setting	
		0	1	0	1
00b (Output retained at the end of the cycle)	0	0	0	1	0
	1	0	1	1	1
01b (Low output at the end of the cycle)	—	0	0	0	0
10b (High output at the end of the cycle)	—	1	1	1	1
11b (Toggle output at the end of the cycle)	0	1	1	0	1
	1	1	0	0	0

m = A, B

Figure 21.49 shows an example of operation of the output of the duty cycle 0% or 100% ($n = 0$ to 2).

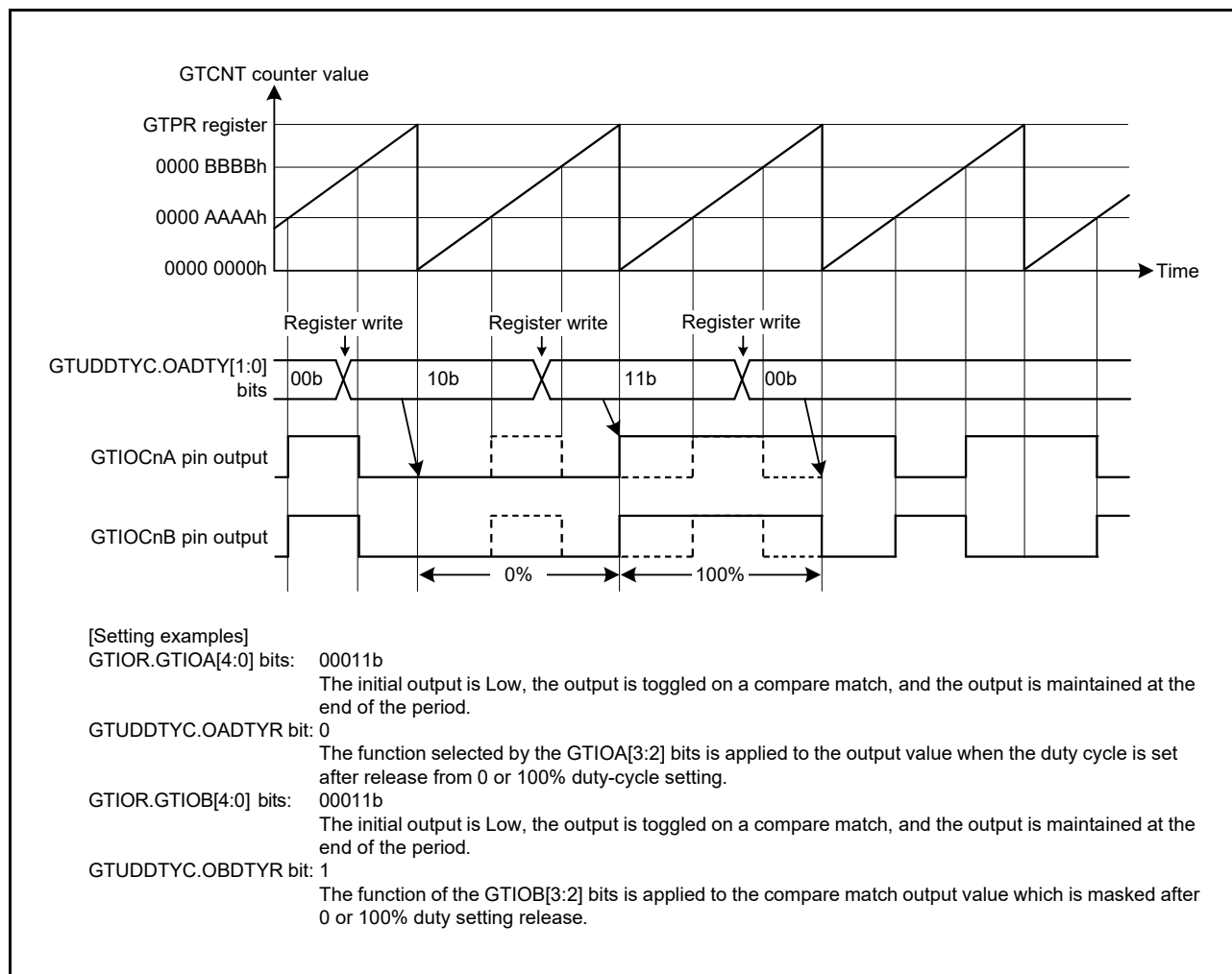


Figure 21.49 Example of Operation of Output of Duty Cycle 0% or 100% ($n = 0$ to 2)

21.3.7 Hardware Count Start/Count Stop and Clear Operation

The GTCNT counter can be started, stopped, or cleared by hardware sources in this MCU. There are 3 types of hardware sources, including external trigger input and GTIOCnm pin input ($n = 0$ to 2 ; $m = A, B$)

There are 2 types of hardware sources, including external trigger input and GTIOCnm pin input ($n = 0$ to 2 ; $m = A, B$)

It is also possible to control the count start of the GTCNT counter synchronized with the MTU count start.

21.3.7.1 Hardware Start Operation

The GTCNT counter can be started by a hardware source. Select a hardware source to start counting using the GTSSR register, and enable to start counting.

Figure 21.50 shows an example of count start operation by a hardware source. Figure 21.51 shows the setting example.

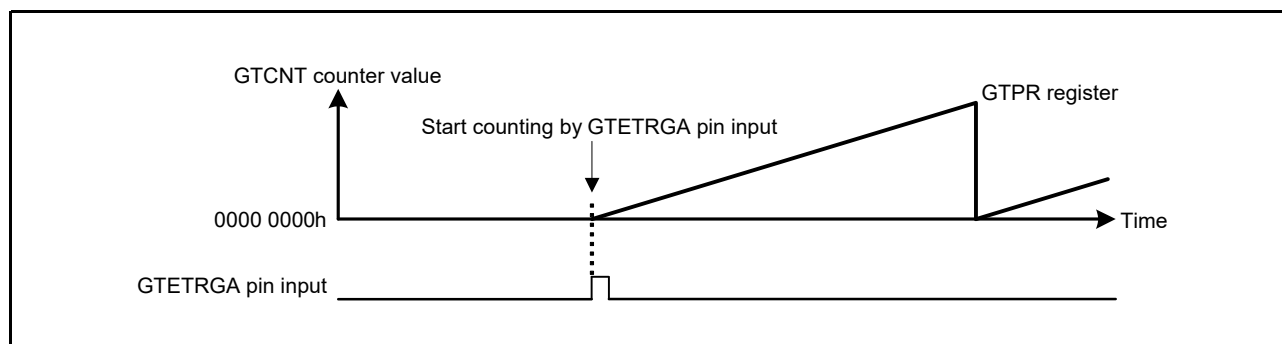


Figure 21.50 Example of Count Start Operation by Hardware Source (At starting by GTETRGA pin)

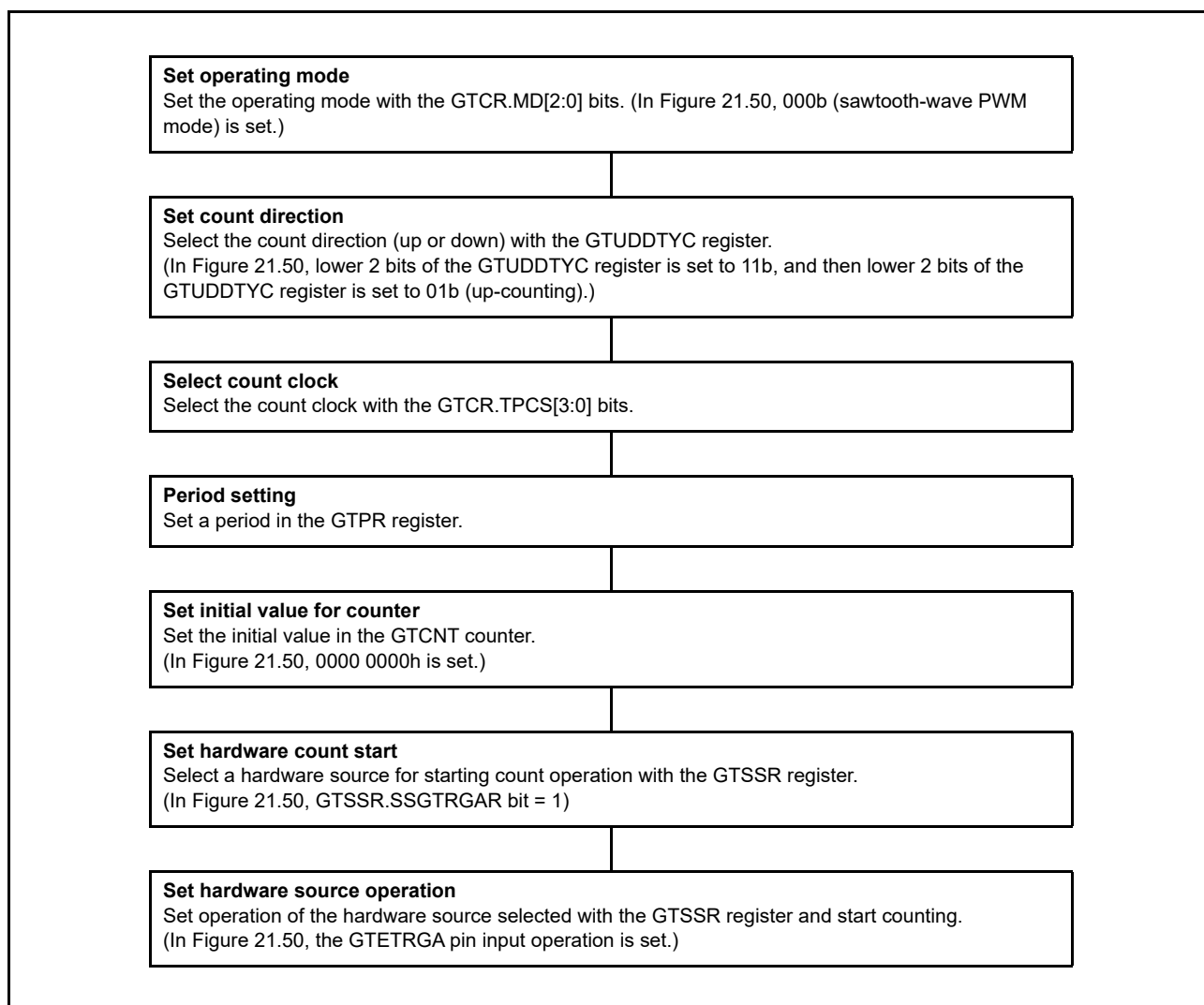


Figure 21.51 Example for Setting Count Start Operation by Hardware Source

Figure 21.52 shows an example of timing of operations to start counting in response to a rising edge of the input on the GTETRGA pin.

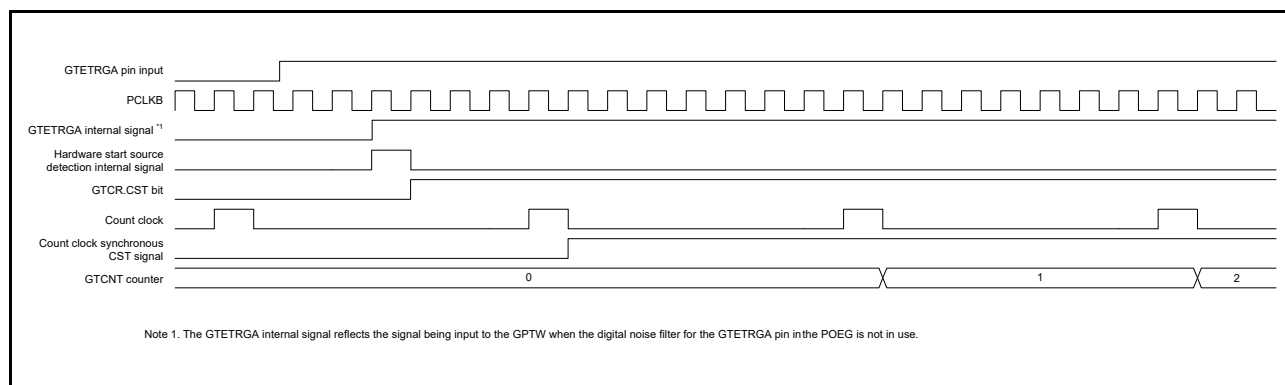


Figure 21.52 Example of Timing of Operations to Start Counting in Response to a Rising Edge of the Input on the GTETRGA Pin

Figure 21.53 shows an example of timing of operations to start counting in response to a rising edge of the input on the GTIOCnA pin.

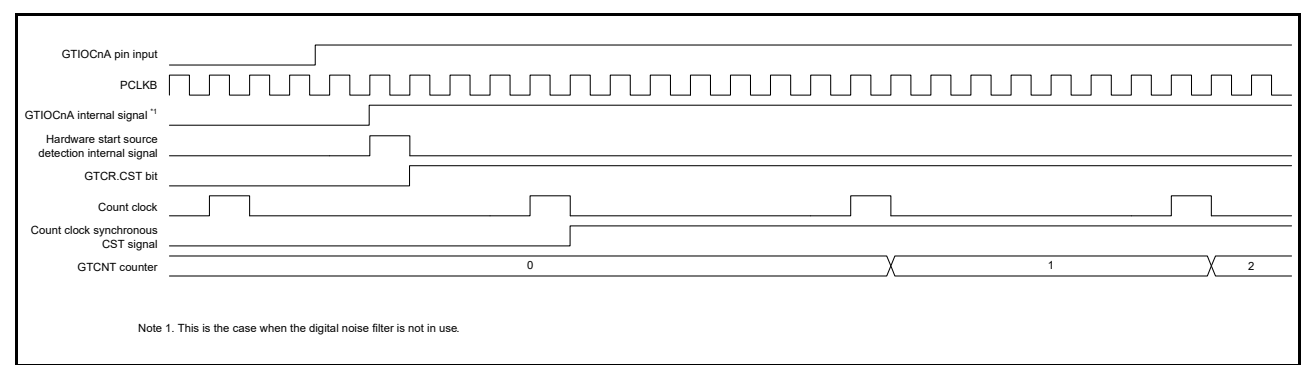


Figure 21.53 Example of Timing of Operations to Start Counting in Response to a Rising Edge of the Input on the GTIOCnA Pin

21.3.7.2 Hardware Stop Operation

The GTCNT counter can be stopped by a hardware source. Select a hardware source to stop counting with the GTPSR register, and enable to stop counting.

Figure 21.54 shows an example of count stop operation by a hardware source. Figure 21.55 shows the setting example. In this example, the count operation is stopped by the GTETRGA pin input, and is restarted by the GTIOCnA pin input.

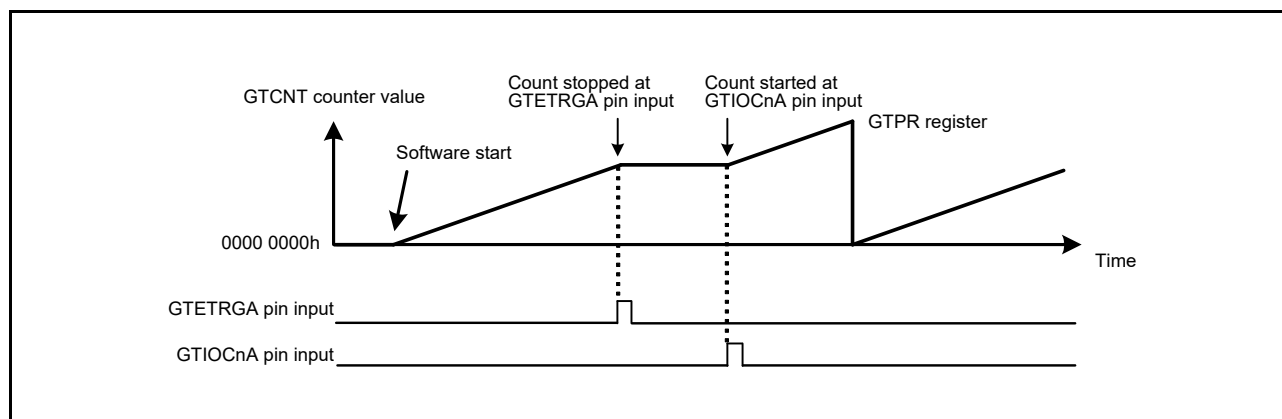


Figure 21.54 Example of Count Stop Operation by Hardware Source
(Started by Software, Stopped at GTETRGA pin input, and Restarted at GTIOCnA pin input)

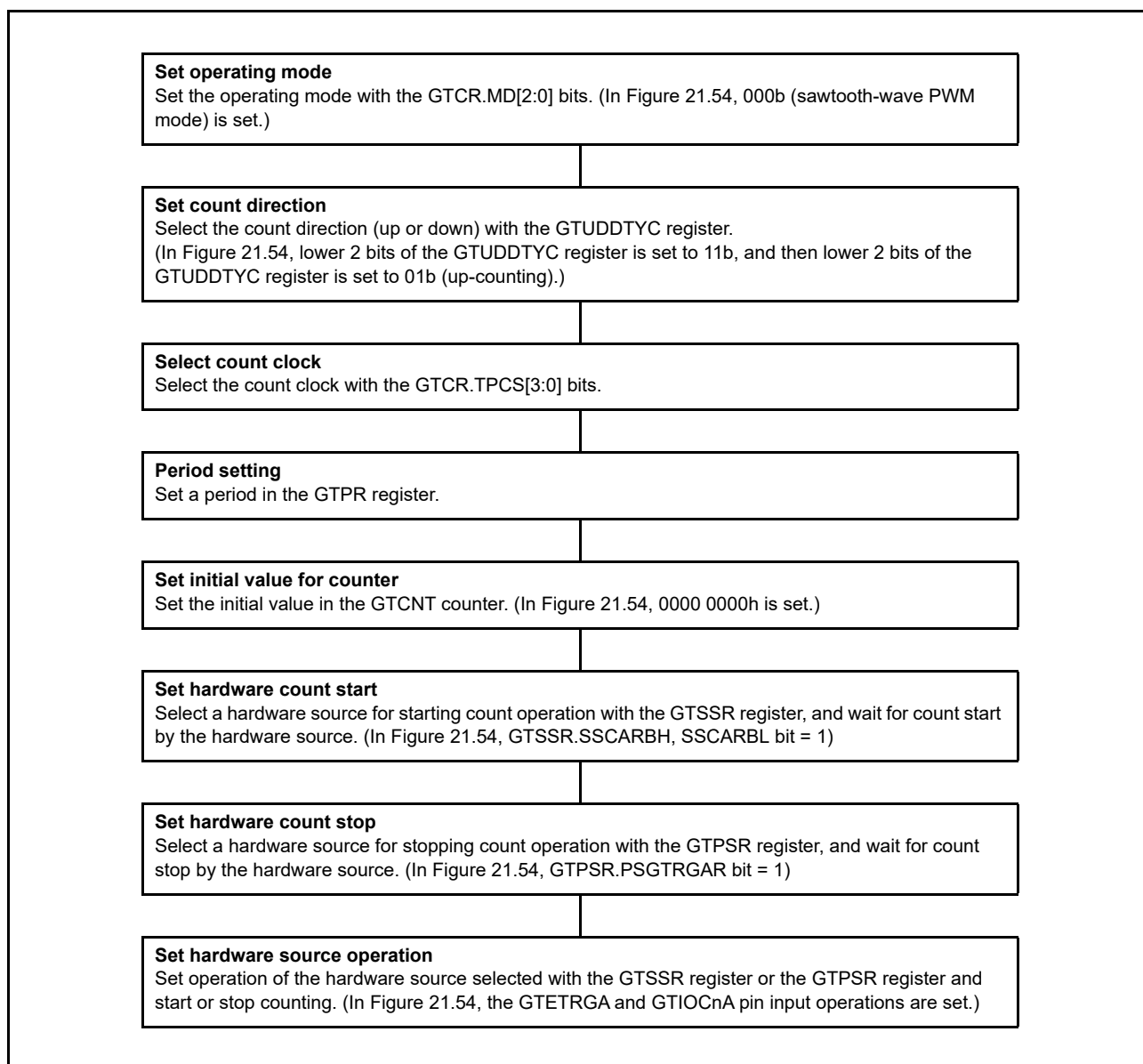


Figure 21.55 Example for Setting Count Stop Operation by Hardware Source

Figure 21.56 shows an example of count start/stop operation by a hardware source. Figure 21.57 shows the setting example. In this example, the counter operates during the high-level periods of the external trigger input GTETRGA pin.

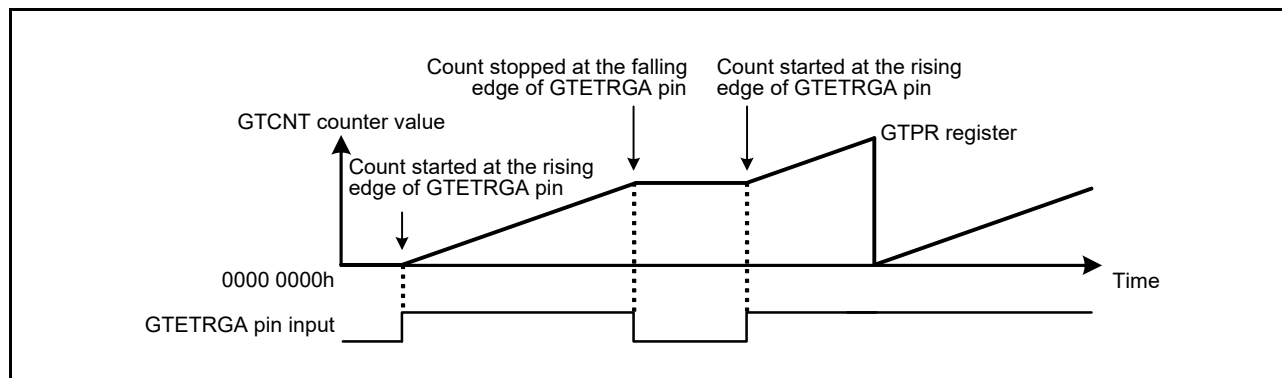


Figure 21.56 Example of Count Start/Stop Operation by Hardware Source (Started at Rising Edge of GTETRGA Pin Input, Stopped at Falling Edge of GTETRGA Pin Input)

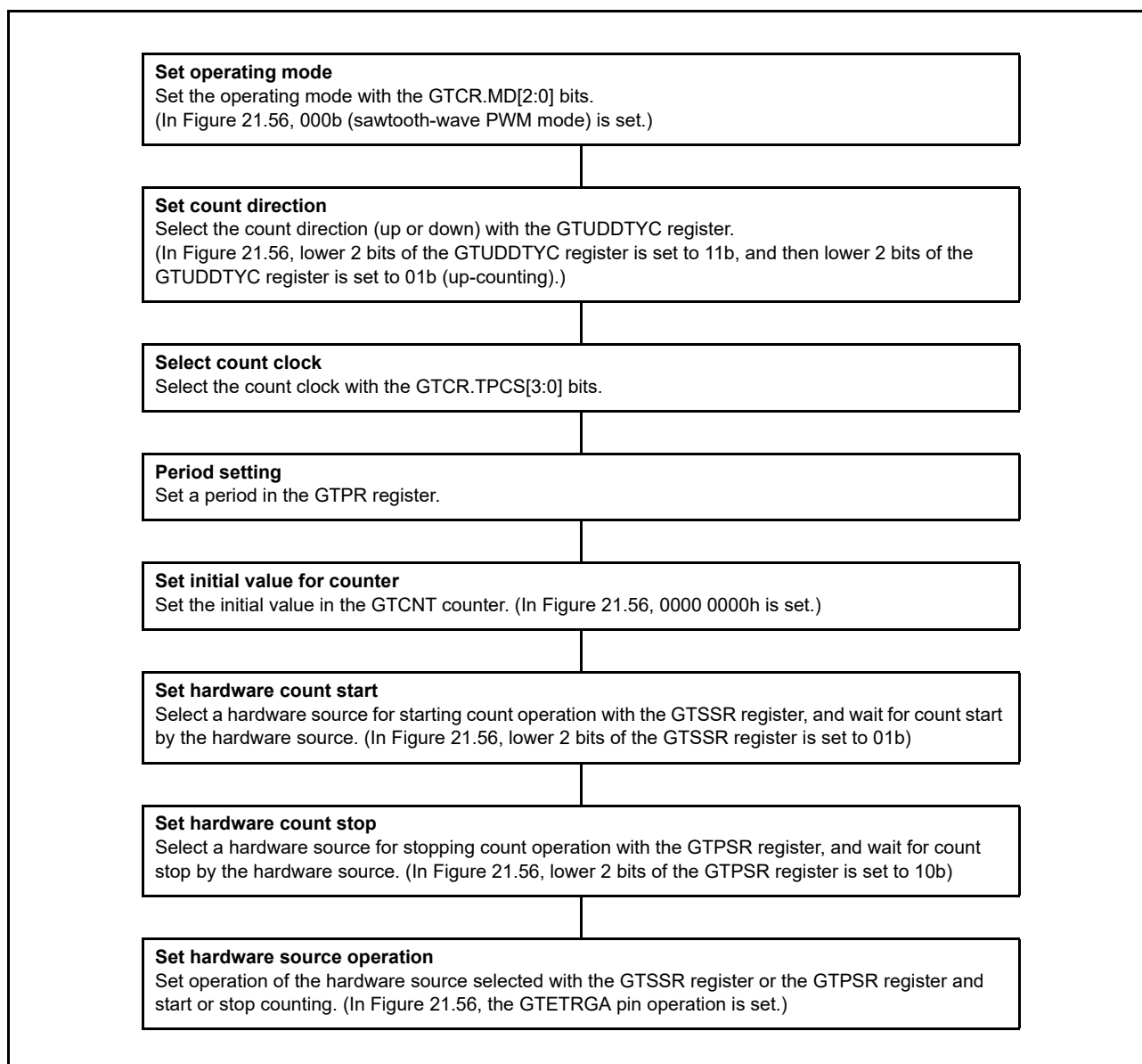


Figure 21.57 Example for Setting Count Start/Stop Operation by Hardware Source

Figure 21.58 shows an example of timing of operations to stop counting in response to a rising edge of the input on the GTETRGA pin.

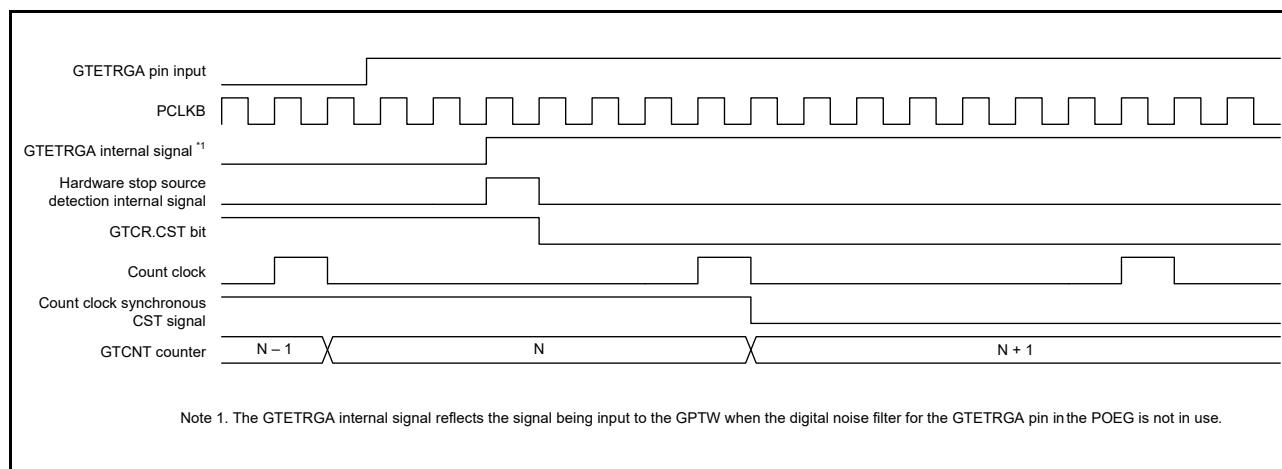


Figure 21.58 Example of Timing of Operations to Stop Counting in Response to a Rising Edge of the Input on the GTETRGA Pin

Figure 21.59 shows an example of timing of operations to stop counting in response to a rising edge of the input on the GTIOCnA pin.

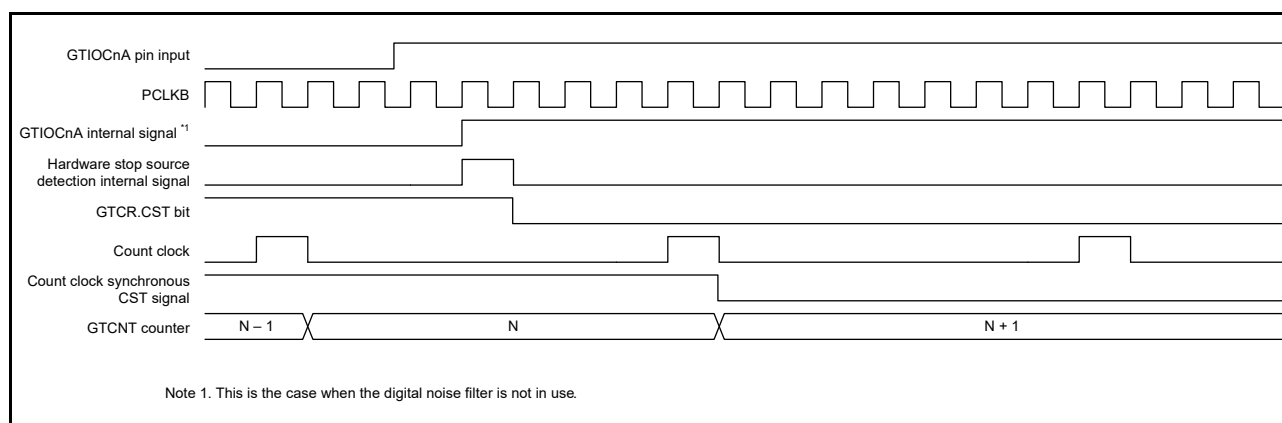


Figure 21.59 Example of Timing of Operations to Stop Counting in Response to a Rising Edge of the Input on the GTIOCnA Pin

21.3.7.3 Hardware Clear Operation

The GTCNT counter can be cleared by a hardware source. Select a hardware source to clear the counter with the GTCSCR register, and enable clearing the counter.

Note that the GTCIV/GTCIU interrupt (overflow/underflow interrupt) is not generated when the GTCNT counter is cleared by a hardware source or by software.

Figure 21.60 and Figure 21.61 show examples of the GTCNT counter clearing operation by a hardware source. Figure 21.62 shows the setting example. In this example, the GTCNT counter is started at the GTIOCnA pin input, and is stopped/cleared at the GTETRGA pin input.

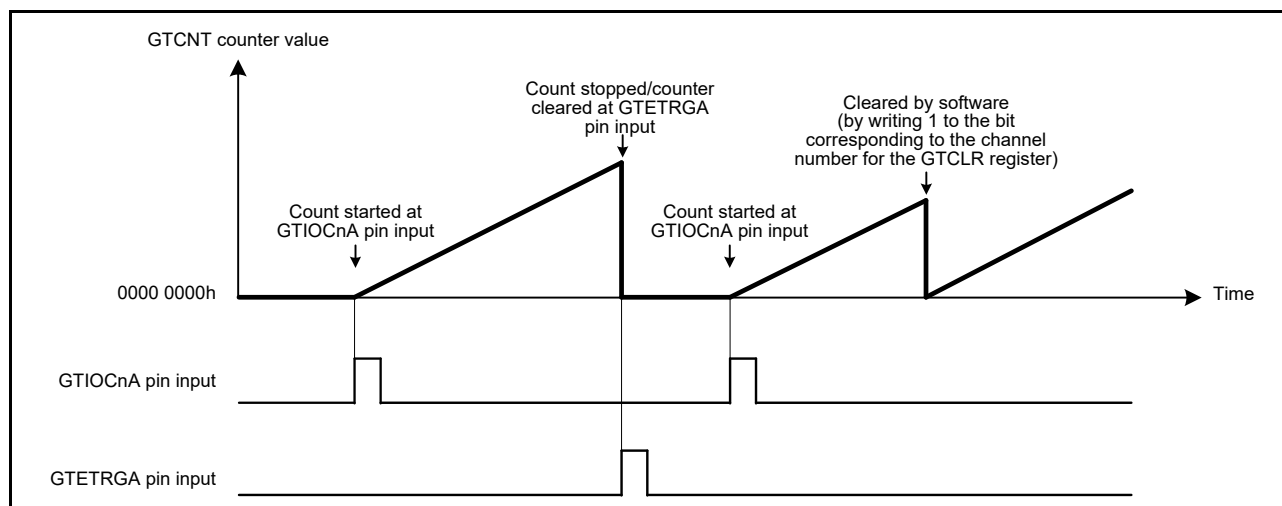


Figure 21.60 Examples of Counter Clearing Operation by Hardware Source (Sawtooth-Wave Up-Counting, Started at GTIOCnA pin input, and Count Stopped/Counter Cleared at GTETRGA pin input)

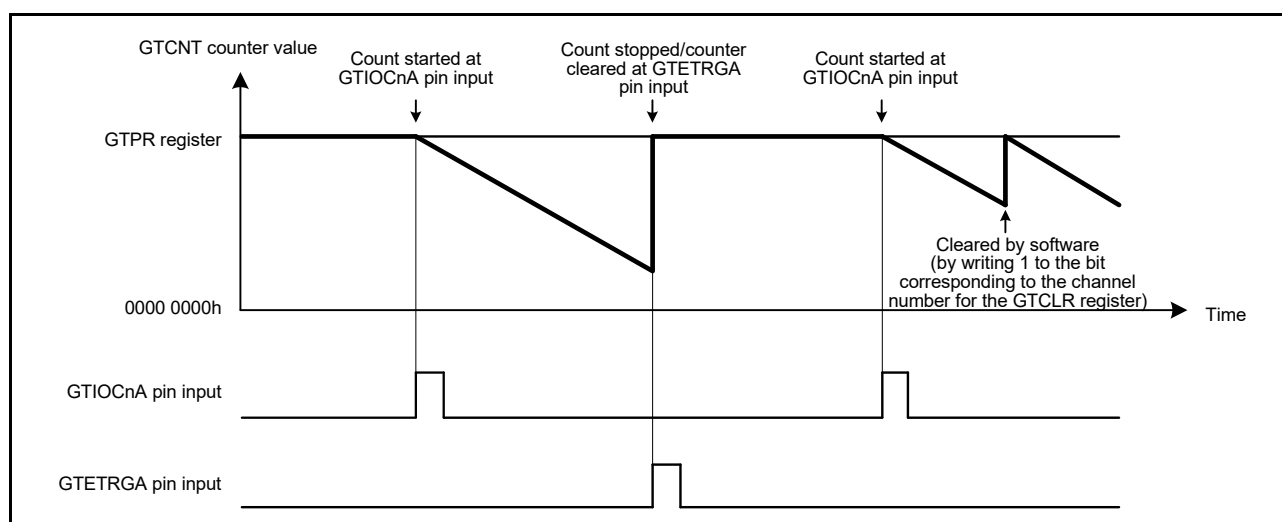


Figure 21.61 Examples of Counter Clearing Operation by Hardware Source (Sawtooth-Wave Down-Counting, Started at GTIOCnA pin input, and Count Stopped/Counter Cleared at GTETRGA pin input)

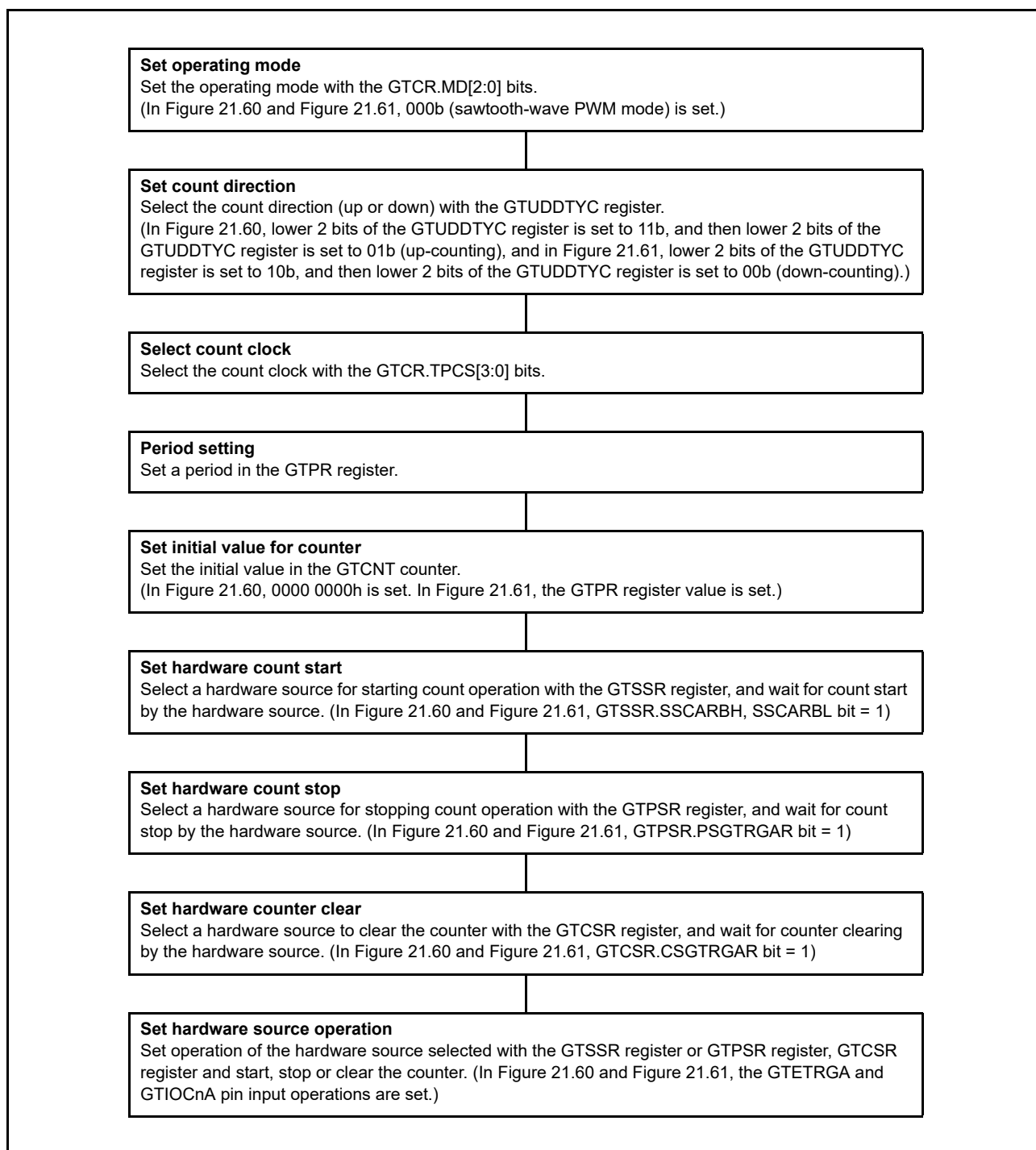


Figure 21.62 Example for Setting Counter Clearing Operation by Hardware Source

The GTCIV/GTCIU interrupt (overflow/underflow interrupt) does not occur when the counter is cleared by a hardware source. In the same way, the GTCIV/GTCIU interrupt does not occur when the counter is cleared by software. Figure 21.63 shows the relationship between counter clearing by a hardware source and the GTCIV interrupt.

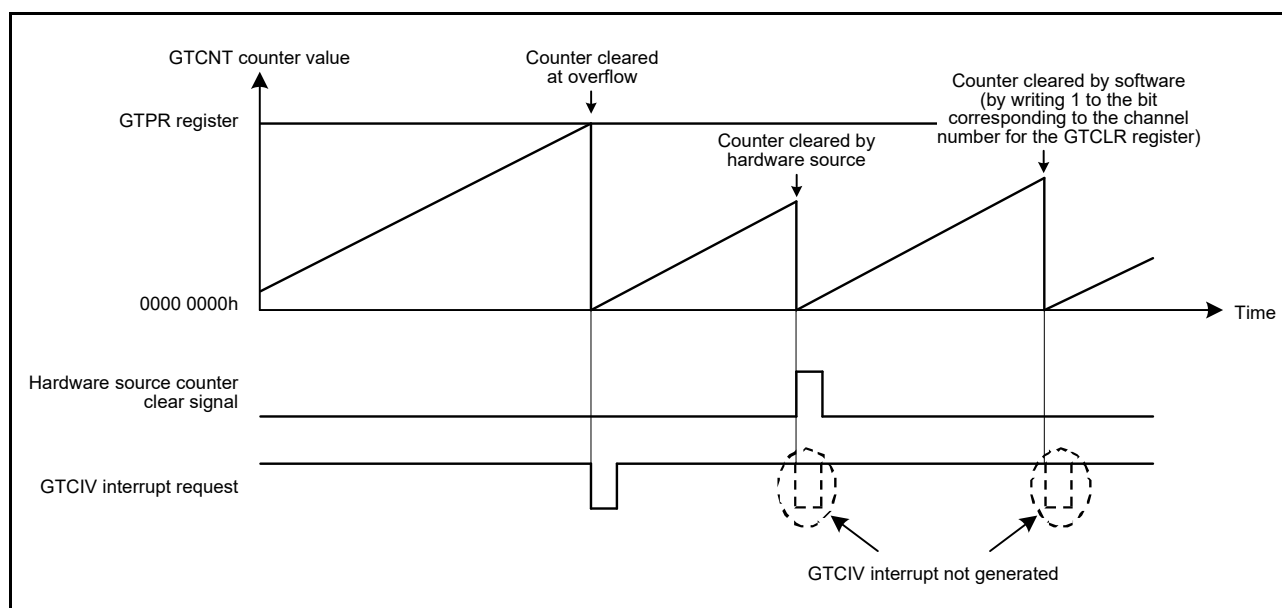


Figure 21.63 Relationship between Counter Clearing by Hardware Source and GTCIV Interrupt

Figure 21.64 shows an example of the timing of operations to clear the counter in response to a rising edge of the input on the GTETRGA pin when a clock signal produced by frequency-dividing the PCLKB signal is used as the counter clock for the GTCNT counter.

The GTCNT counter is cleared when counting is in progress after the GPTW has detected the internal clearing signal.

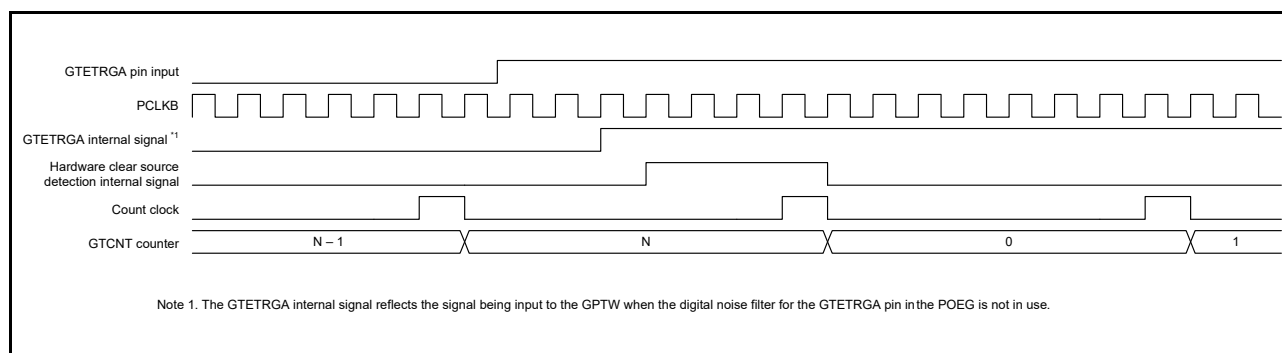


Figure 21.64 Example of the Timing of Operations for Counter Clearing in Response to a Rising Edge of the Input on the GTETRGA Pin (During the Counting of Cycles of Clock Signal Produced by Dividing the PCLKB Frequency)

Figure 21.65 shows an example of the timing of operations to clear the counter in response to a rising edge of the input on the GTIOCnA pin when a clock signal produced by frequency-dividing the PCLKB signal is used as the counter clock for the GTCNT counter.

The GTCNT counter is cleared when counting is in progress after the GPTW has detected the internal clearing signal.

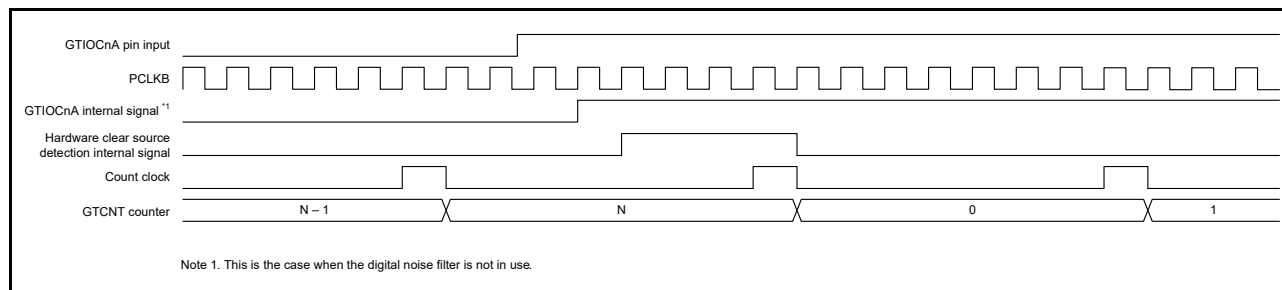


Figure 21.65 Example of the Timing of Operations for Counter Clearing in Response to a Rising Edge of the Input on the GTIOCnA Pin (During the Counting of Cycles of Clock Signal Produced by Dividing the PCLKB Frequency)

21.3.7.4 MTU Synchronous Count Start Operation

GTCNT counter count start control synchronized with MTU count start input is available.

Figure 21.50 shows an example of count start operation by a hardware source. Figure 21.51 shows the setting example.

Figure 21.66 shows an example of GPTW0 count start operation timing with MTU0 count start. This is an example of a count start operation of the GPTW0.GTCNT counter by a count start operation of the MTU.TSTRA.CST0 bit. The MTU and GPTW count clocks are counted using the PCLKB/2 prescaler divider ratio setting, and GPTW starts counting 4 PCLKB cycles after the MTU starts counting.

It takes “2 PCLKB clock cycles × GPTW prescaler divider ratio” cycles from MTU count start to GPTW count start. The division ratio of the prescaler can be set with the GTCR.TPCS[3:0] bits.

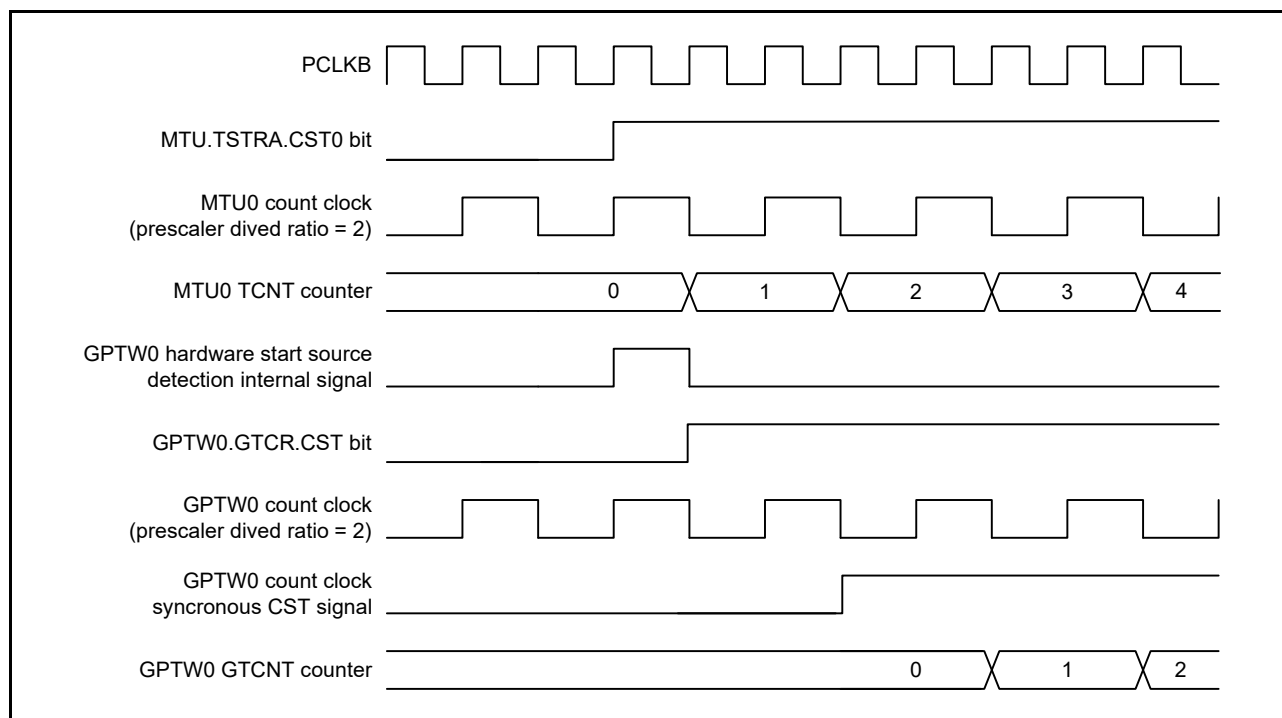


Figure 21.66 Example of count start operation timing by MTU count start

21.3.8 Synchronous Operation on Channels

Synchronous operation on channels (synchronous start/stop/clear operation) can be performed.

21.3.8.1 Synchronous Operation by Software

The count operations for respective channels can be started, stopped and cleared simultaneously by setting multiple bits of the GTSTR, GTSTP, and GTCLR registers to 1 at the same time.

The GTCNT counter value settings are synchronized by setting the GTCR.SSCEN bit to 1, and can be written simultaneously to multiple channels set in the same group by the GTCR.SSCGRP[1:0] bits.

When either the SSCE bit or the SSCD bit of the GTSECR register is set to 1, the GTCR.SSCEN bits of the channels that selected on the GTSECSR register are set to value and the GTCNT synchronous set/clear function of multiple channels are enabled or disabled at the same time.

Because the clock of count operation is selected by GTCR.TPCS[3:0] bits in respective channels, if the clock period of each channel that performs synchronous operation (count start/stop/clear) is different from others, the synchronous operation timings of every channels are not exact same.

Figure 21.67 shows an example of simultaneous start/stop/clearing of three channels by software. Figure 21.68 to Figure 21.70 show examples of simultaneous start/stop/clearing with different count period.

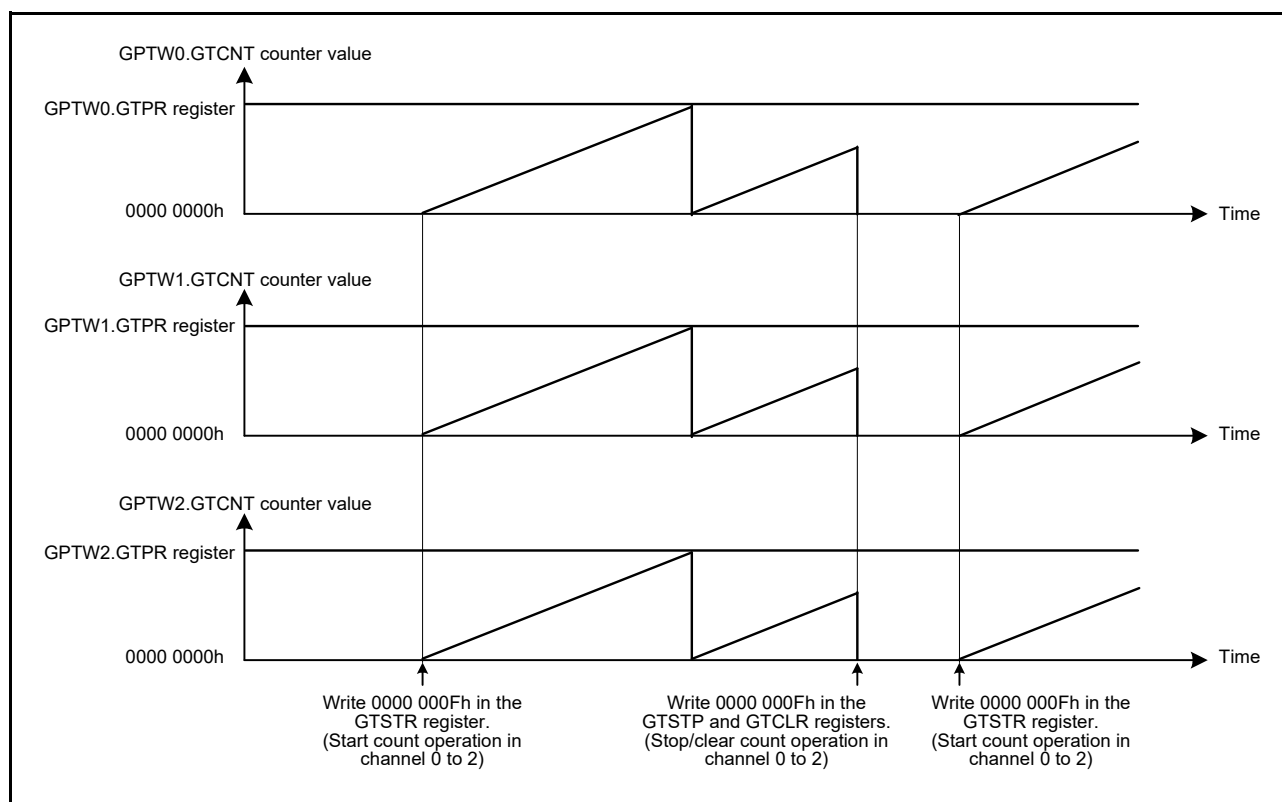


Figure 21.67 Example of Simultaneous Start/Stop/Clearing Operation by Software (with the Same Count Period (GTPR Register Value))

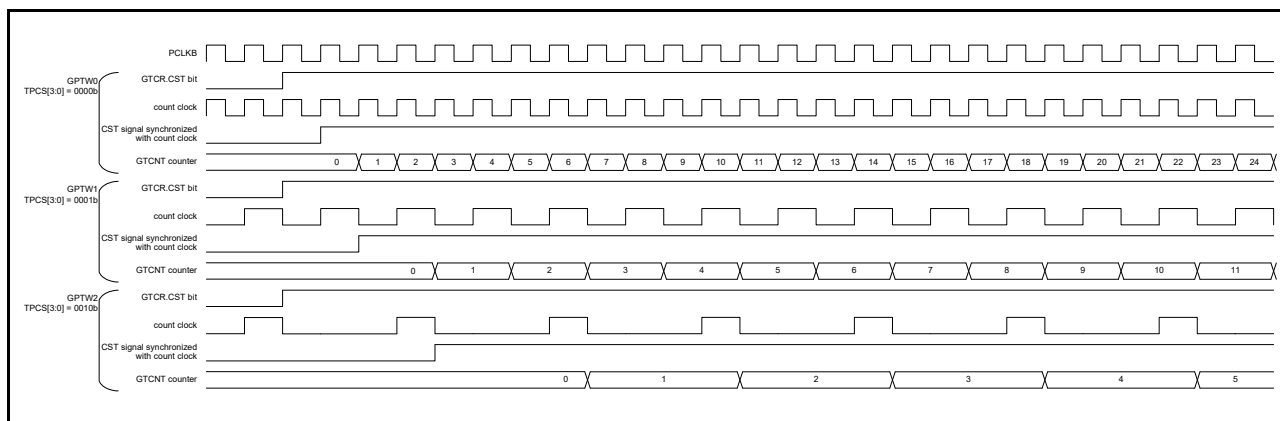


Figure 21.68 Example of Simultaneous Start Operation by Software (with Different Count Period)

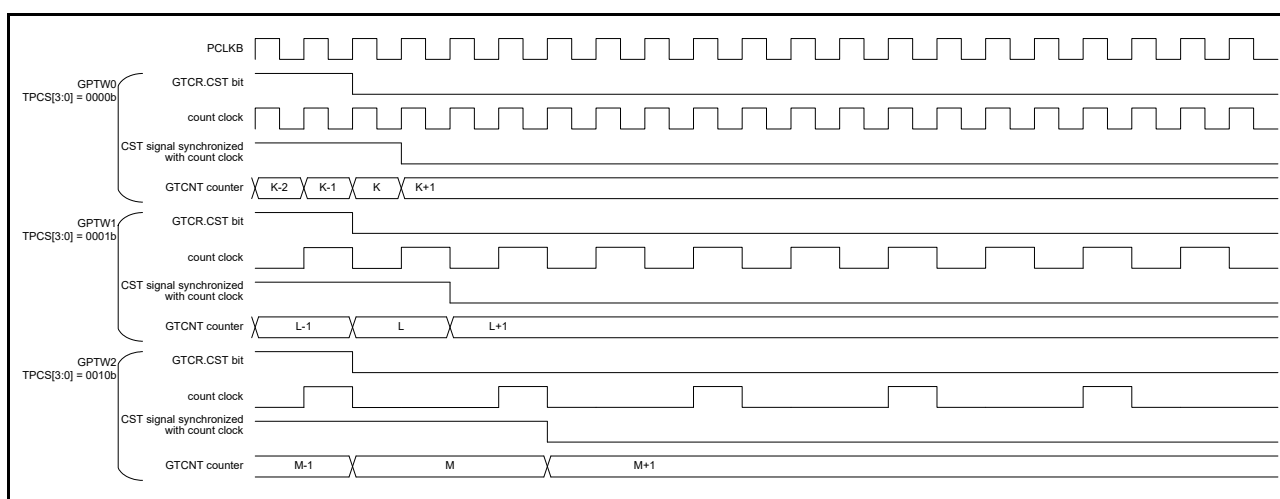


Figure 21.69 Example of Simultaneous Stop Operation by Software (with Different Count Period)

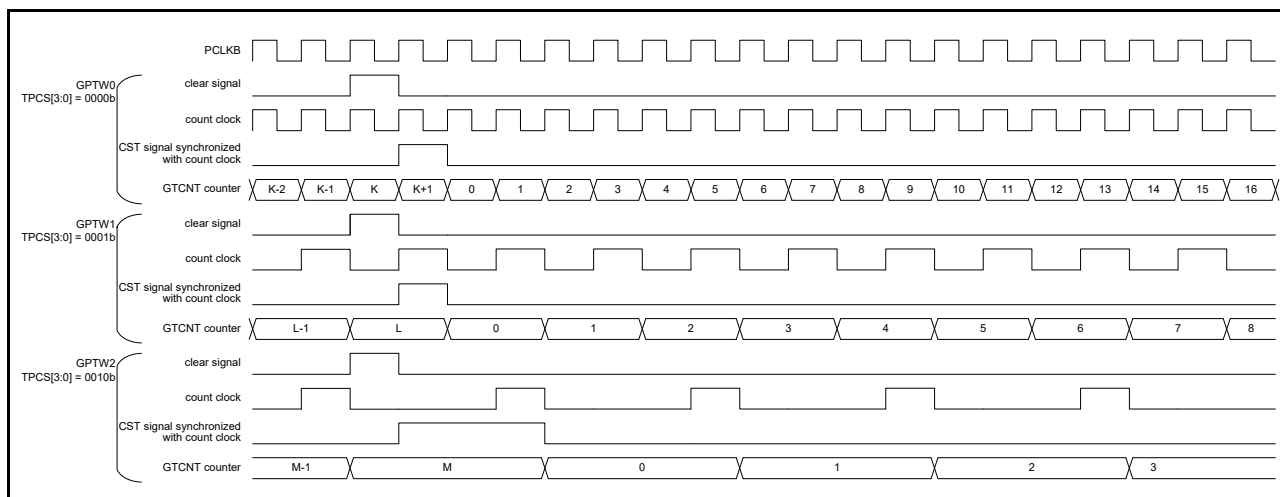


Figure 21.70 Example of Simultaneous Clearing Operation by Software (with Different Count Period)

21.3.8.2 Synchronous Operation by Hardware Source

The count operations for respective channels can be started/stopped/cleared simultaneously by following hardware sources: external trigger input, and MTU count start input.

Figure 21.71 shows an example of simultaneous start/stop/clearing operation of three channels by hardware source.

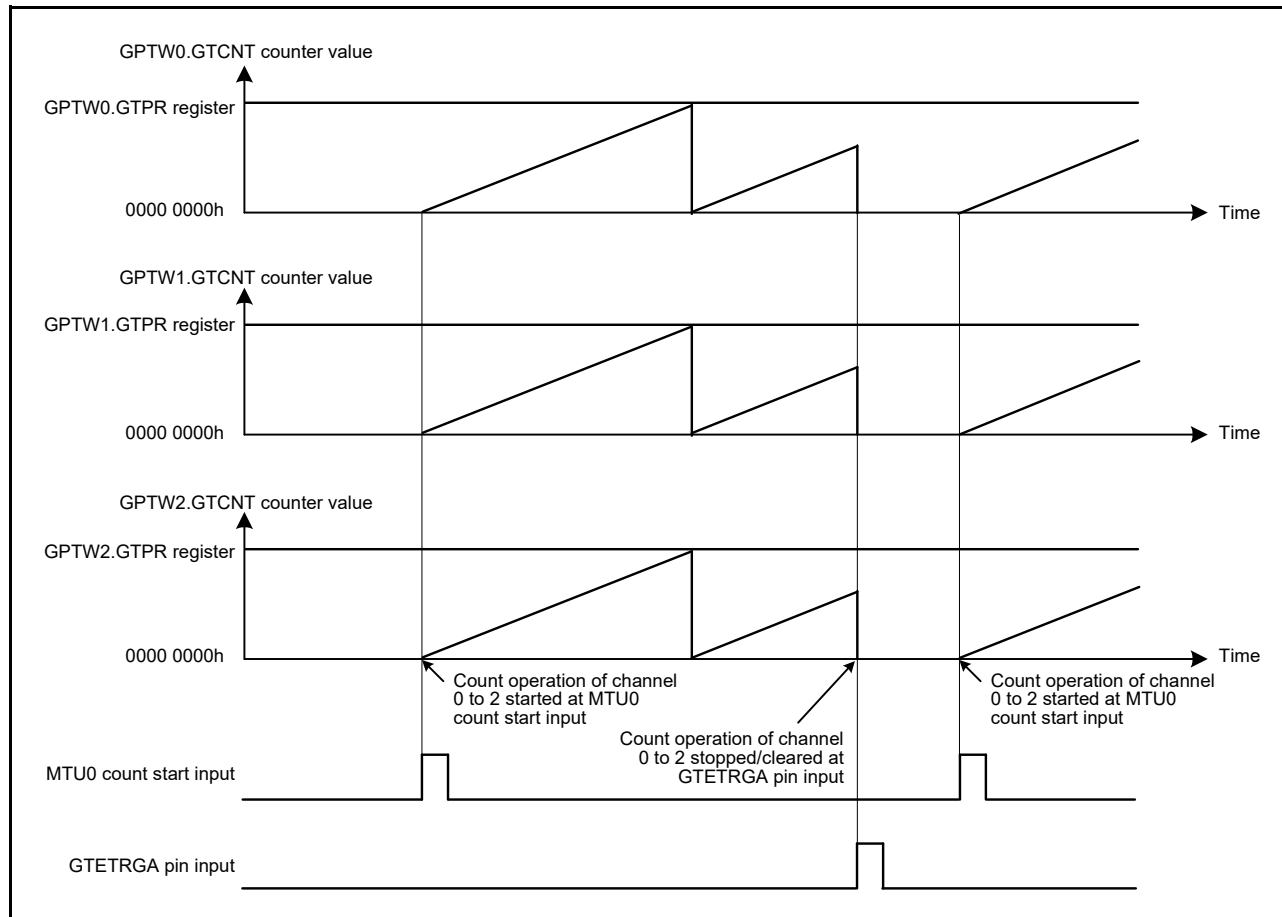


Figure 21.71 Example of Simultaneous Start/Stop/Clearing Operation by Hardware Source (with Same Count Period (GTPR Register Value))

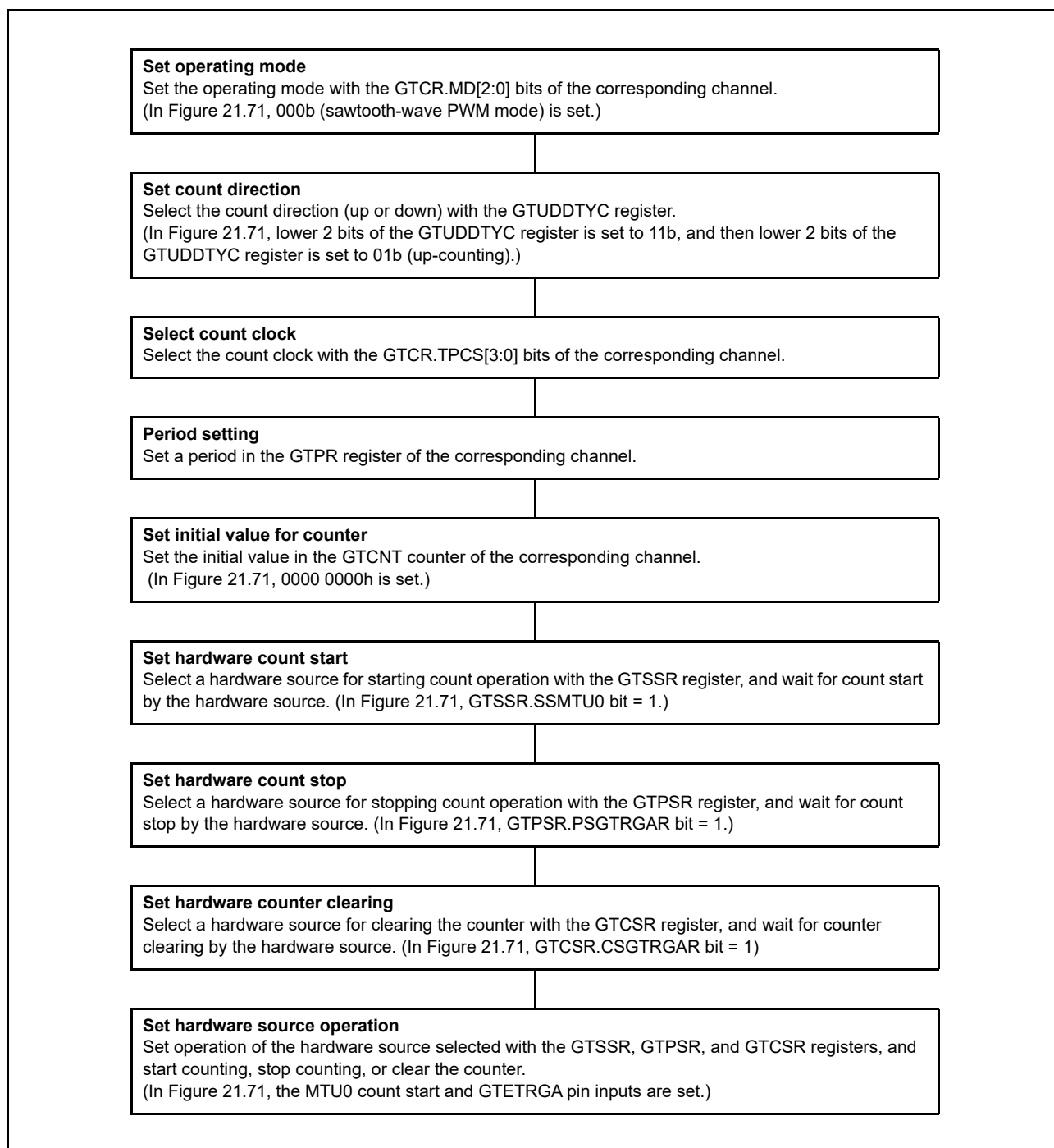


Figure 21.72 Example for Setting Simultaneous Start by Hardware Source

21.3.8.3 Synchronous Clear Operation by Inter Channel Cooperation

The counters of other channels can be cleared at the same time with the counter clear of some channel caused by the compare match, input capture, sawtooth wave up-count overflow, sawtooth wave down-count underflow, and the GTIOCnA/GTIOCnB pin input selected in the GTCSR register.

The counter clear caused by GTCLR register, external trigger input can perform as synchronous clear when the same counter clear factor is set to target channels of synchronous clear. Therefore, these factors are not prepared for the synchronous clear operation by inter channel cooperation.

Set the channel that generates the synchronization clear source and the channel that is synchronized clear in the same synchronous set/clear group with the GTCR.SSCGRP[1:0] bits.

Similar to the synchronous operation in section 21.3.8.1, Synchronous Operation by Software, if the count clock selected by the GTCR.TPCS[3:0] bits is different for each channel, the synchronous clear operation cannot be performed at exact same timing. Similar to the example of simultaneous clearing operation by software in Figure 21.70, if the count clock is different for each channel, the synchronous clear factor is synchronized with the count clock of each channel before counter clearing.

When either the SSCE bit or the SSCD bit of the GTSECR register is set to 1, the GTCR.SSCEN bits of the channels that selected on the GTSECSR register are set to 0 or 1 and the GTCNT synchronous set/clear function of multiple channels are enabled or disabled at the same time.

Table 21.7 lists the settings of the registers used to set sources for synchronous clearing. Figure 21.73 shows an example of synchronous clear operation by inter channel cooperation and Figure 21.74 shows an example for setting synchronous clear operation by inter channel cooperation.

Table 21.7 Settings of the Registers Used to Set Sources for Synchronous Clearing

Sources for Synchronous Clearing by Inter Channel Cooperation	Register.Bit(s)	Value
GTCCRA register compare match/input capture	GTINTAD.SCFA	1b
	GTCSR.CSCMSC[2:0]	001b
GTCCRB register compare match/input capture	GTINTAD.SCFB	1b
	GTCSR.CSCMSC[2:0]	010b
GTCCRC register compare match	GTINTAD.SCFC	1b
	GTCSR.CSCMSC[2:0]	011b
GTCCRD register compare match	GTINTAD.SCFD	1b
	GTCSR.CSCMSC[2:0]	100b
GTCCRE register compare match	GTINTAD.SCFE	1b
	GTCSR.CSCMSC[2:0]	101b
GTCCRF register compare match	GTINTAD.SCFF	1b
	GTCSR.CSCMSC[2:0]	110b
Sawtooth wave up-count overflow	GTINTAD.SCFPO	1b
Sawtooth wave down-count underflow	GTINTAD.SCFPU	1b
Clearing due to input on the GTIOCnA or GTIOCnB pin	GTCR.SCGTIOC	1b

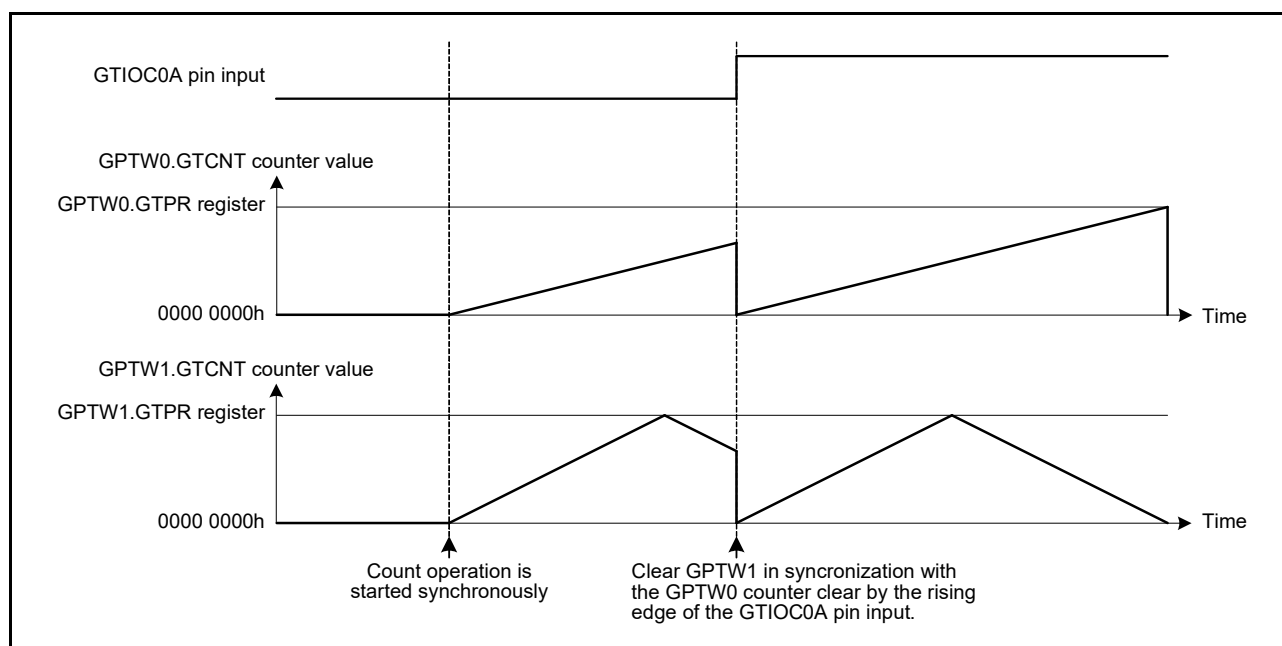


Figure 21.73 Example of Synchronous Clear Operation by Inter Channel Cooperation (Clear Caused by GTIOCnA/GTIOCnB Pin)
 (GPTW0 is sawtooth wave and counter is cleared by the rising edge of the GTIOC0A, GPTW1 is triangle wave, GPTW0, GPTW1 are the same synchronous set/clear group)

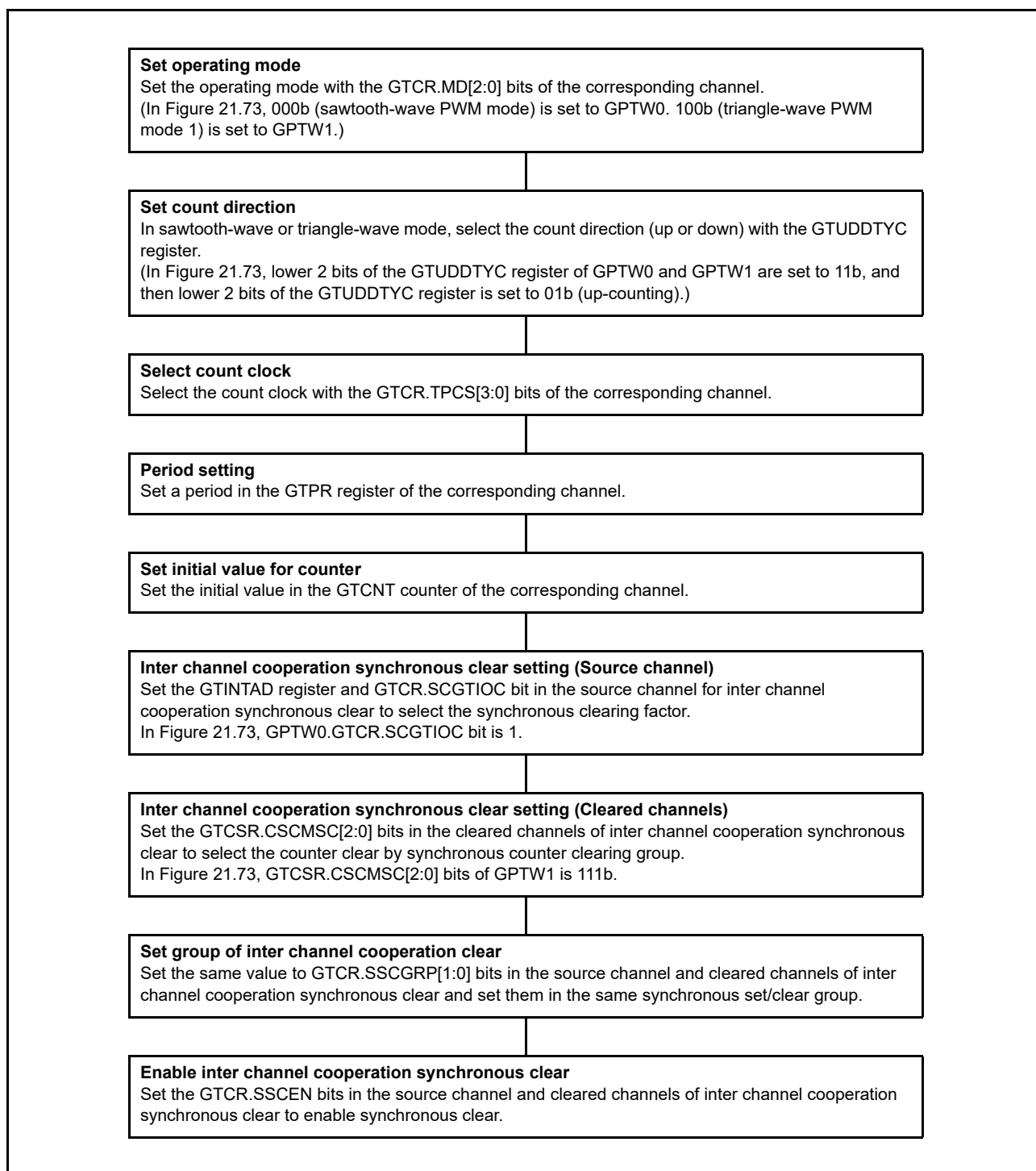


Figure 21.74 Example for Setting Synchronous Clear Operation by Inter Channel Cooperation (Clear Caused by GTIOCnA/GTIOCnB Pin)

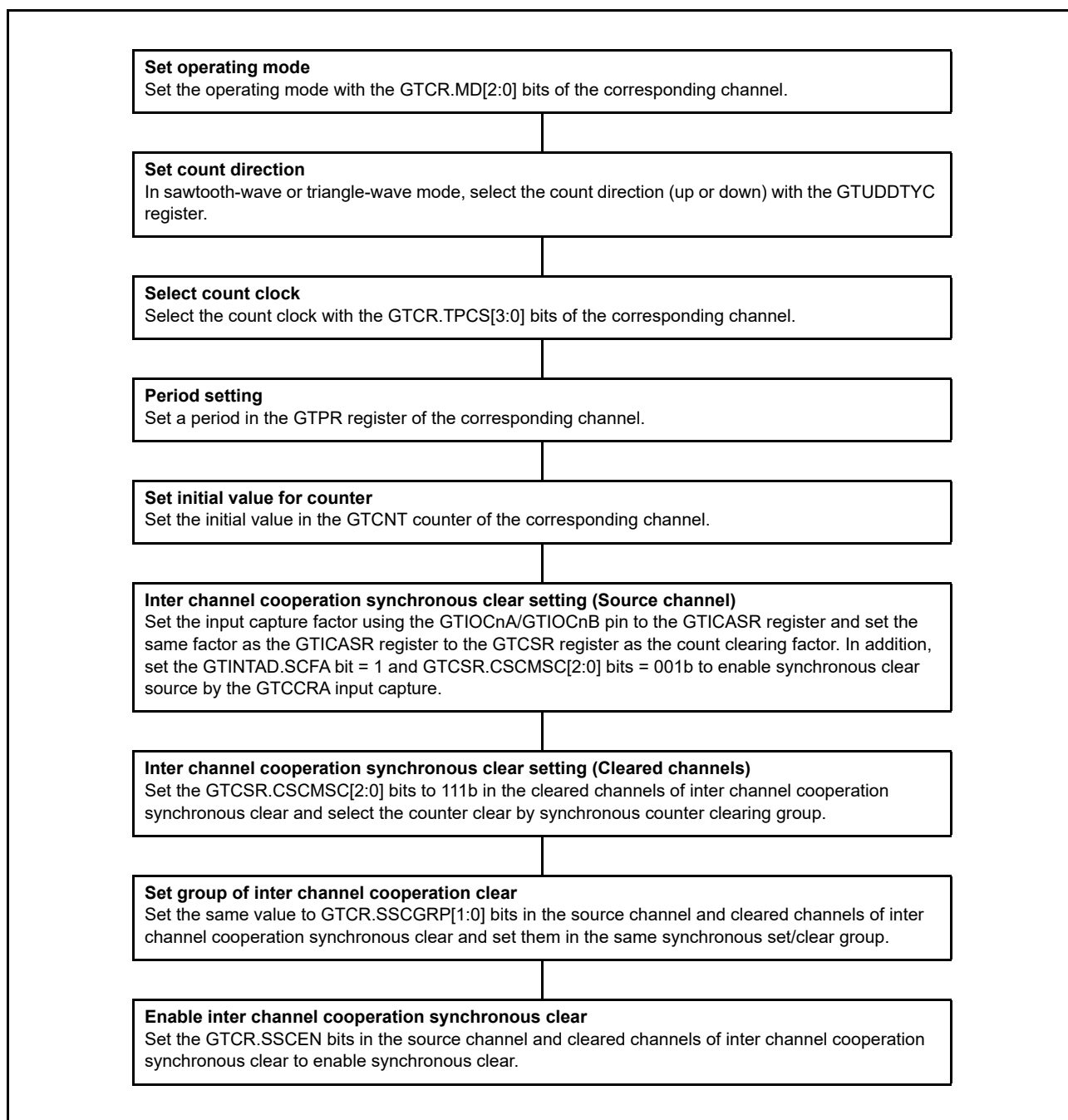


Figure 21.75 Example for Setting Synchronous Clear Operation by Inter Channel Cooperation (Clear Caused by the GTCCRA input capture using the GTIOCnA/GTIOCnB Pin)

21.3.8.4 Input Capture Operation by Inter Channel Cooperation

The events of compare match, input capture, sawtooth wave up-count overflow, sawtooth wave down-count underflow, the crest and trough of triangle wave and the count clock can be used as the input capture factor for the GTCCRm register ($m = A$ or B) of other channels.

The input capture factor by inter channel cooperation can be set by the GTICCR register of the channel that generates the input capture factor. And input capture by other channel factors can be enable by GTICmSR.mSOC bit ($m = A$ or B) in the captured channels. The channels that generate input capture factors and the channels that are captured are set to the same input capture group by GTICCR.ICmGRP[1:0] bit ($m = A$ or B).

Figure 21.76 shows an example of input capture operation by inter channel cooperation and Figure 21.77 shows an example for setting input capture operation by inter channel cooperation.

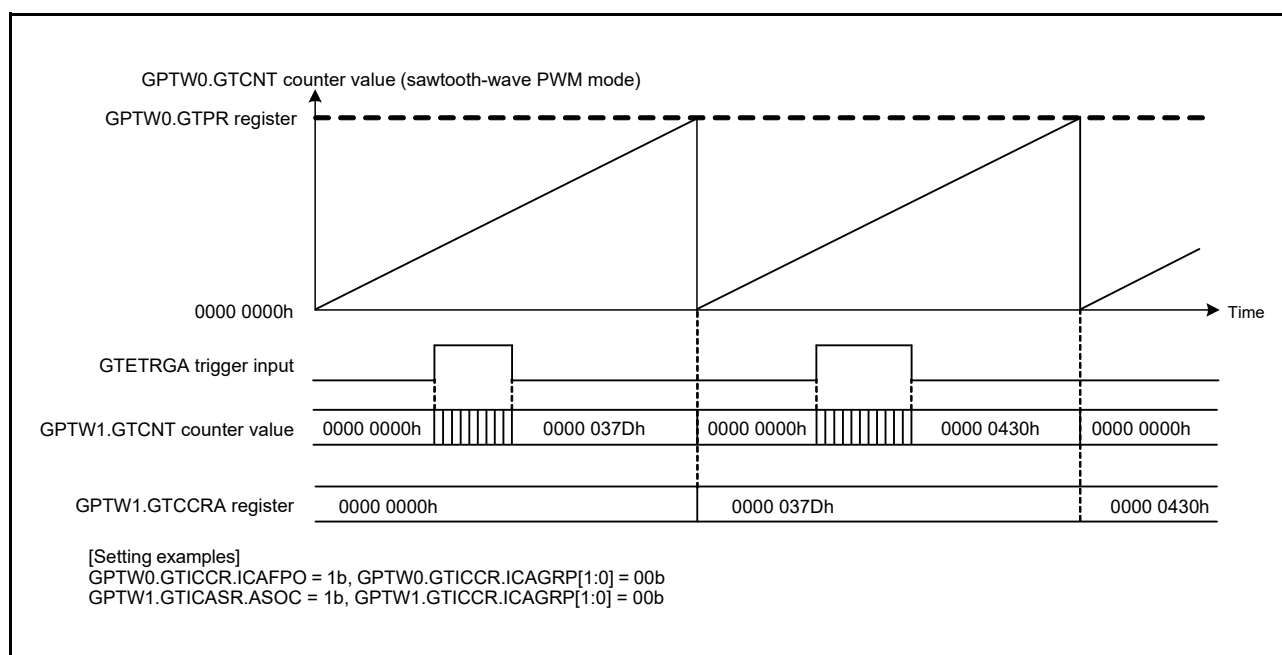


Figure 21.76 Example of Input Capture Operation by Inter Channel Cooperation (GPTW1 is captured by the overflow of GPTW0)

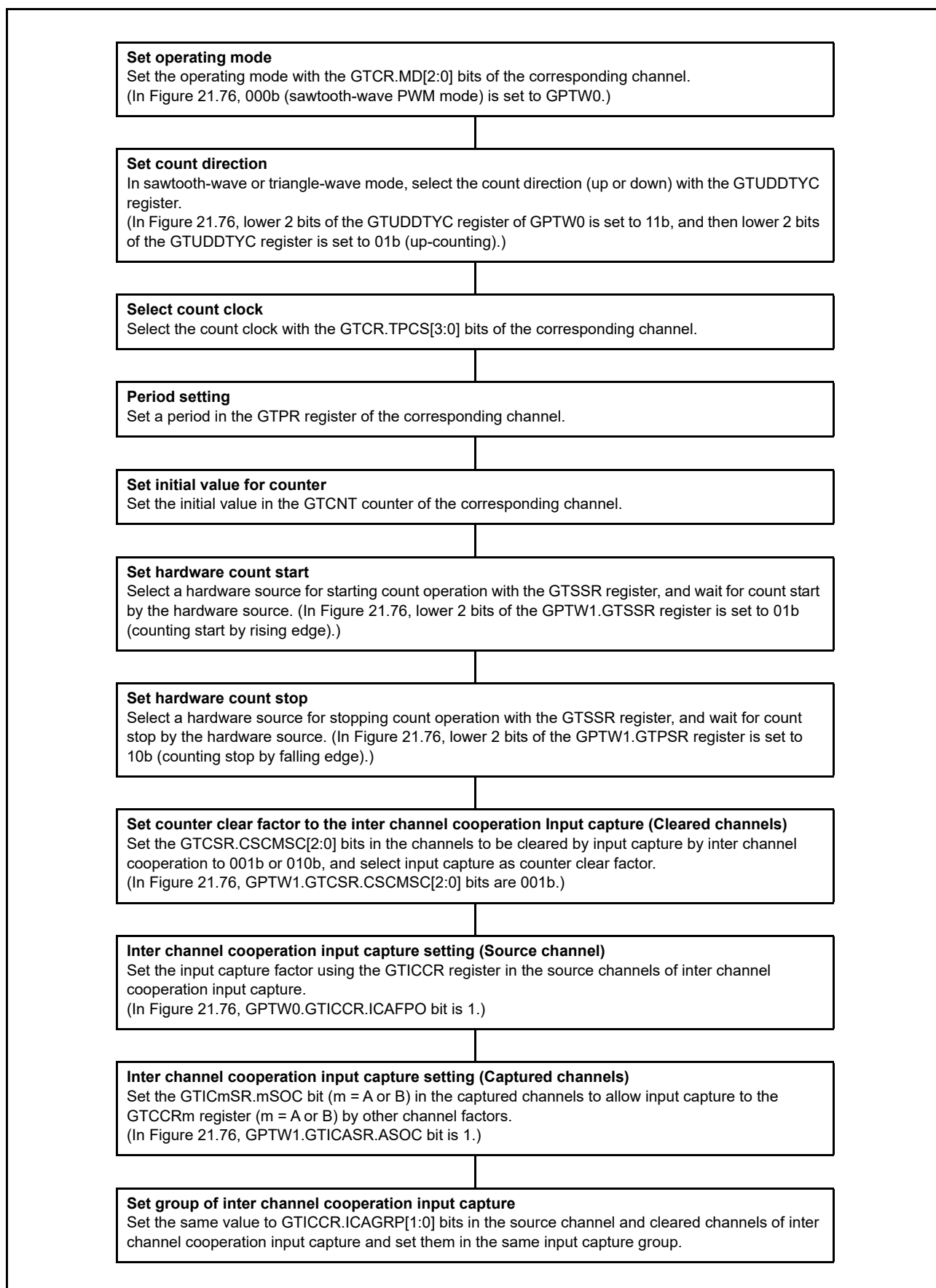


Figure 21.77 Example for Setting Input Capture Operation by Inter Channel Cooperation

21.3.8.5 Synchronous Operation by Phase Shift Start

Count start with phase differences among channels is possible firstly by setting the GTCNT counter value before counting starts, and then, by setting simultaneously multiple bits in the GTSTR register to 1.

Figure 21.78 shows an example of phase shift start among three channels by software.

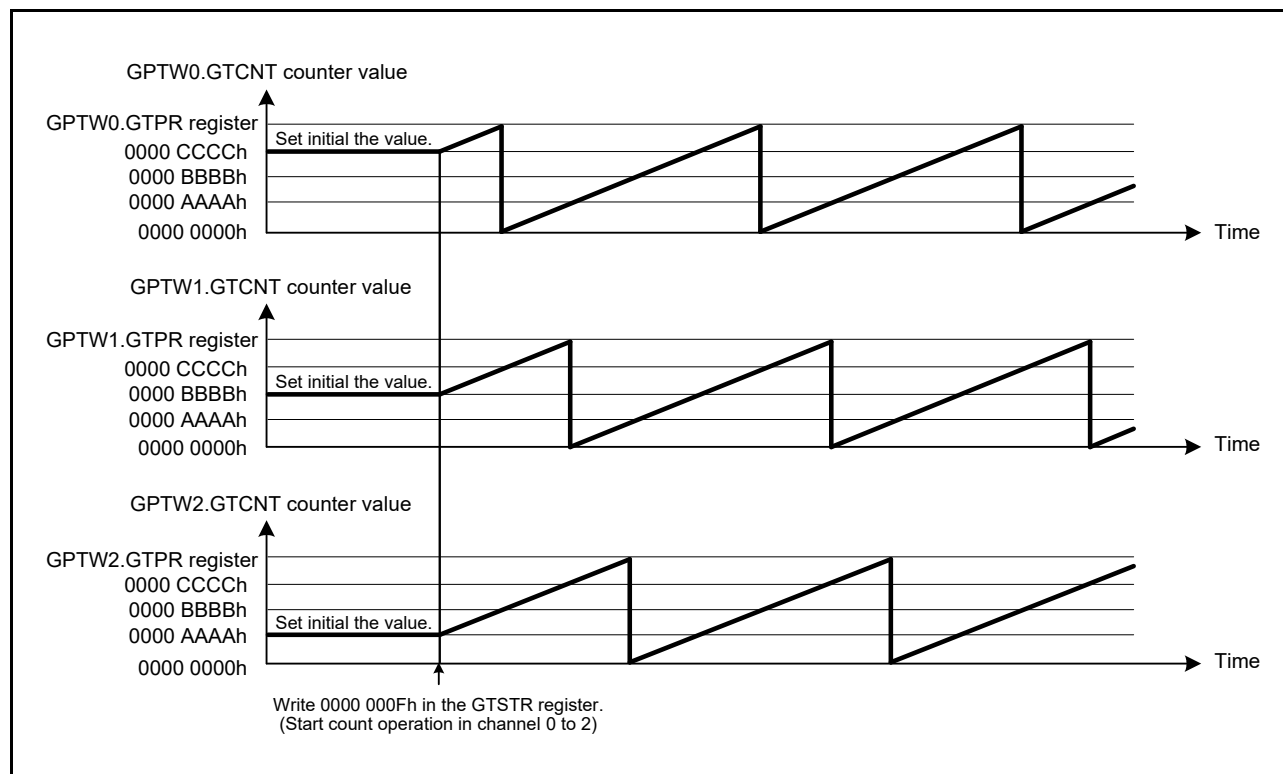


Figure 21.78 Example of Software Phase Shift Start (with Same Count Period (GTPR Register Value))

21.3.9 PWM Output Operation Examples

(1) Synchronous PWM Output

The GPTW can output two phases of linked PWM waveforms for one channel, or 6 phases of linked PWM waveforms for a maximum of 3 channels by synchronizing operation of the channels.

Figure 21.79 shows an example in which three channels perform synchronous operation in sawtooth-wave PWM mode and six phases of PWM waveforms are output. The GTIOCnA pin ($n = 0$ to 2) is set so that it will output low as the initial output, high at a GTCCRA register compare match, and low at the end of the cycle. The GTIOCnB pin is set so that it will output low as the initial output, high at a GTCCRB register compare match, and low at the end of the cycle.

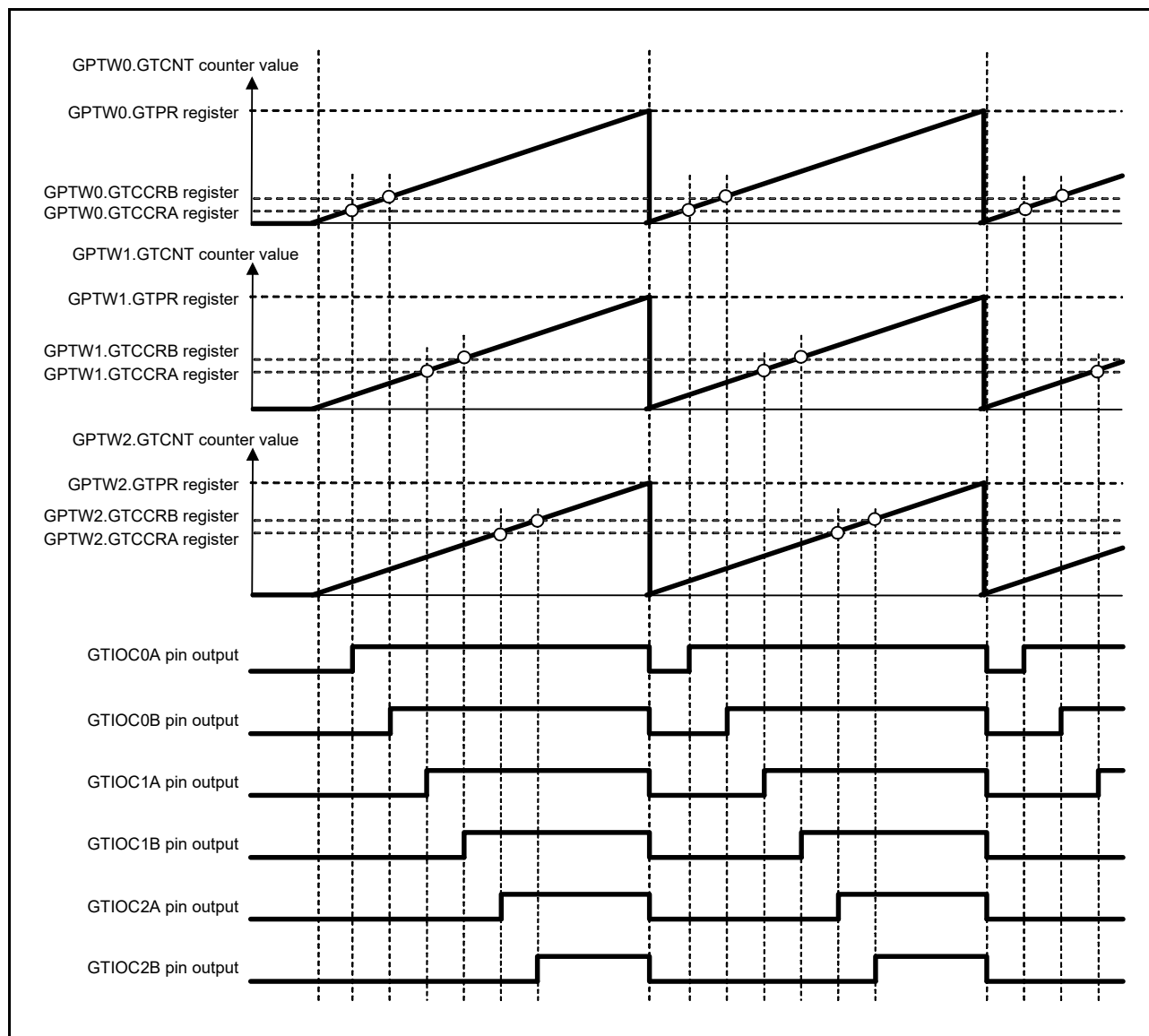


Figure 21.79 Example of Synchronous PWM Output

(2) Three-Phase Sawtooth-Wave Complementary PWM Output

Figure 21.80 shows an example in which three channels perform synchronous operation in sawtooth-wave PWM mode and three-phase complementary PWM waveforms are output. The GTIOCnA pin ($n = 0$ to 2) is set so that it will output low as the initial output, high at a GTCCRA register compare match, and low at the end of the cycle. The GTIOCnB pin is set so that it will output high as the initial output, low at a GTCCRB register compare match, and high at the end of the cycle.

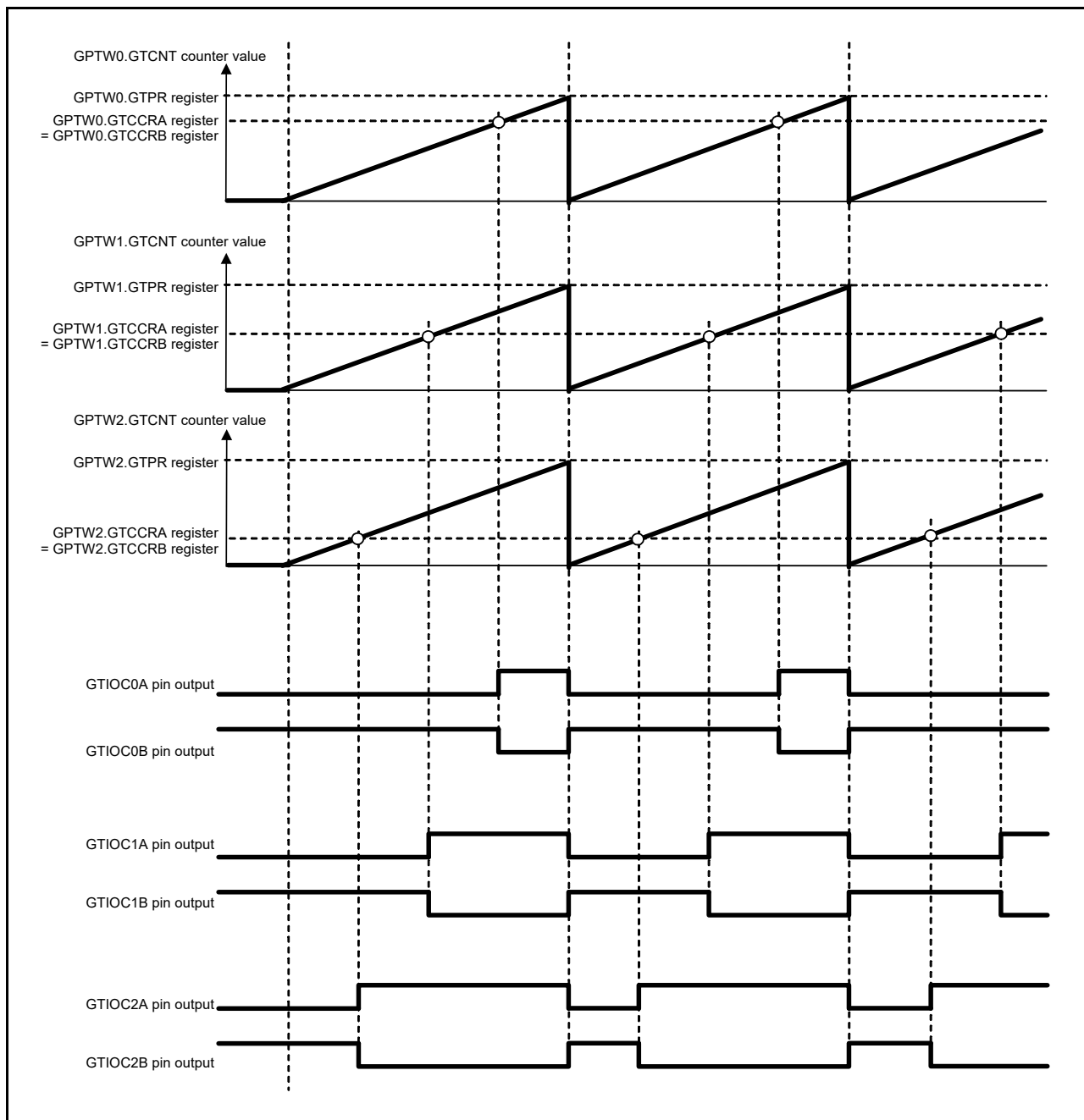


Figure 21.80 Example of Three-Phase Sawtooth-Wave Complementary PWM Output

(3) Three-Phase Sawtooth-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 21.81 shows an example in which three channels perform synchronous operation in sawtooth-wave one-shot pulse mode with automatic dead time setting and three-phase complementary PWM waveforms are output. The GTIOCnA pin ($n = 0$ to 2) is set so that it will output low as the initial output, toggle the output at a GTCCRA register compare match, and retain the output at the end of the cycle. The GTIOCnB pin is set so that it will output high as the initial output, toggle the output at a GTCCRB register compare match, and retain the output at the end of the cycle.

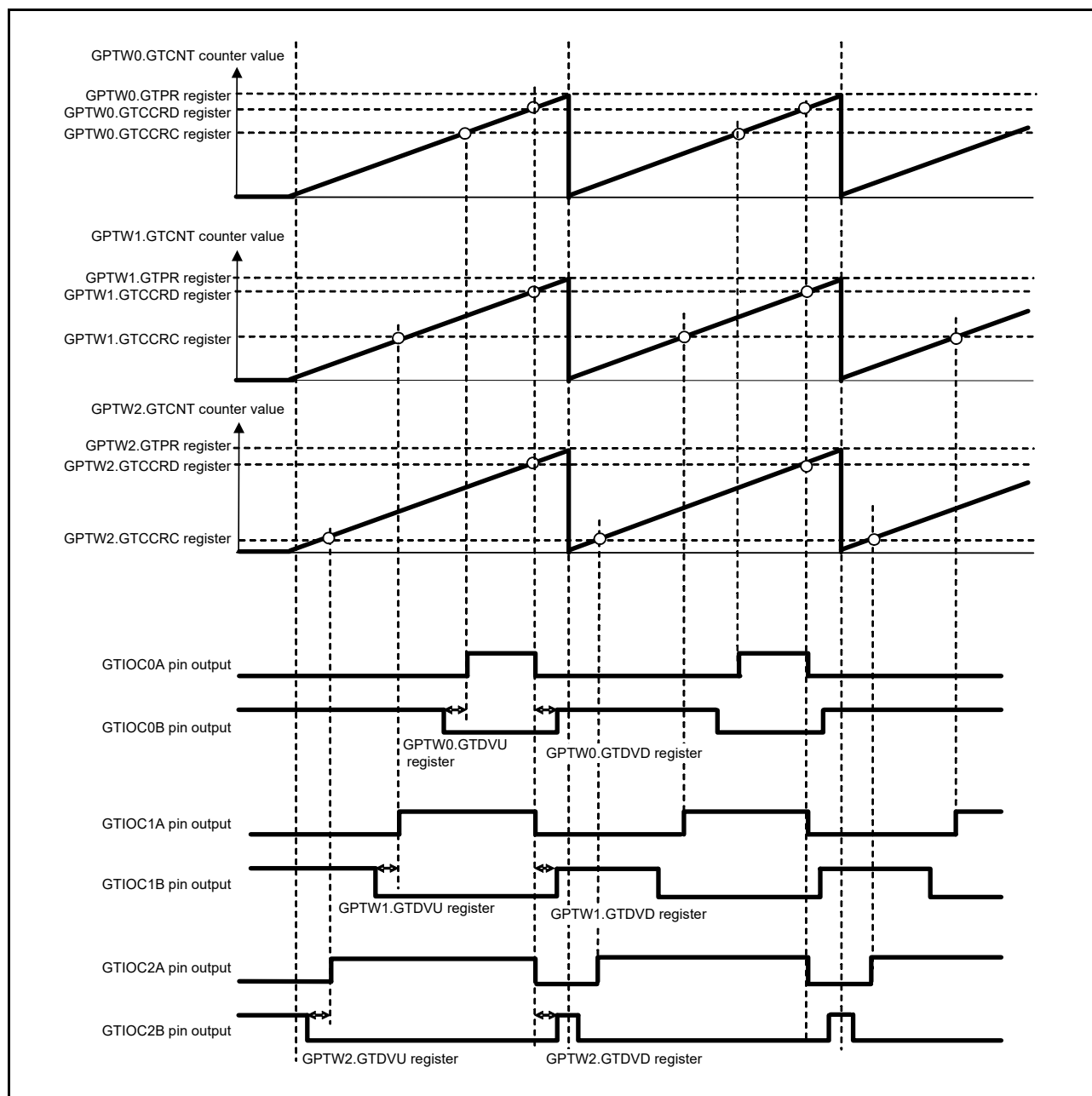


Figure 21.81 Example of Three-Phase Sawtooth-Wave Complementary PWM Output with Automatic Dead Time Setting

(4) Three-Phase Triangle-Wave Complementary PWM Output

Figure 21.82 shows an example in which three channels perform synchronous operation in triangle-wave PWM mode 1 and three-phase complementary PWM waveforms are output. The GTIOCnA pin ($n = 0$ to 2) is set so that it will output low as the initial output, toggle the output at a GTCCRA register compare match, and retain the output at the end of the cycle. The GTIOCnB pin is set so that it will output high as the initial output, toggle the output at a GTCCRB register compare match, and retain the output at the end of the cycle.

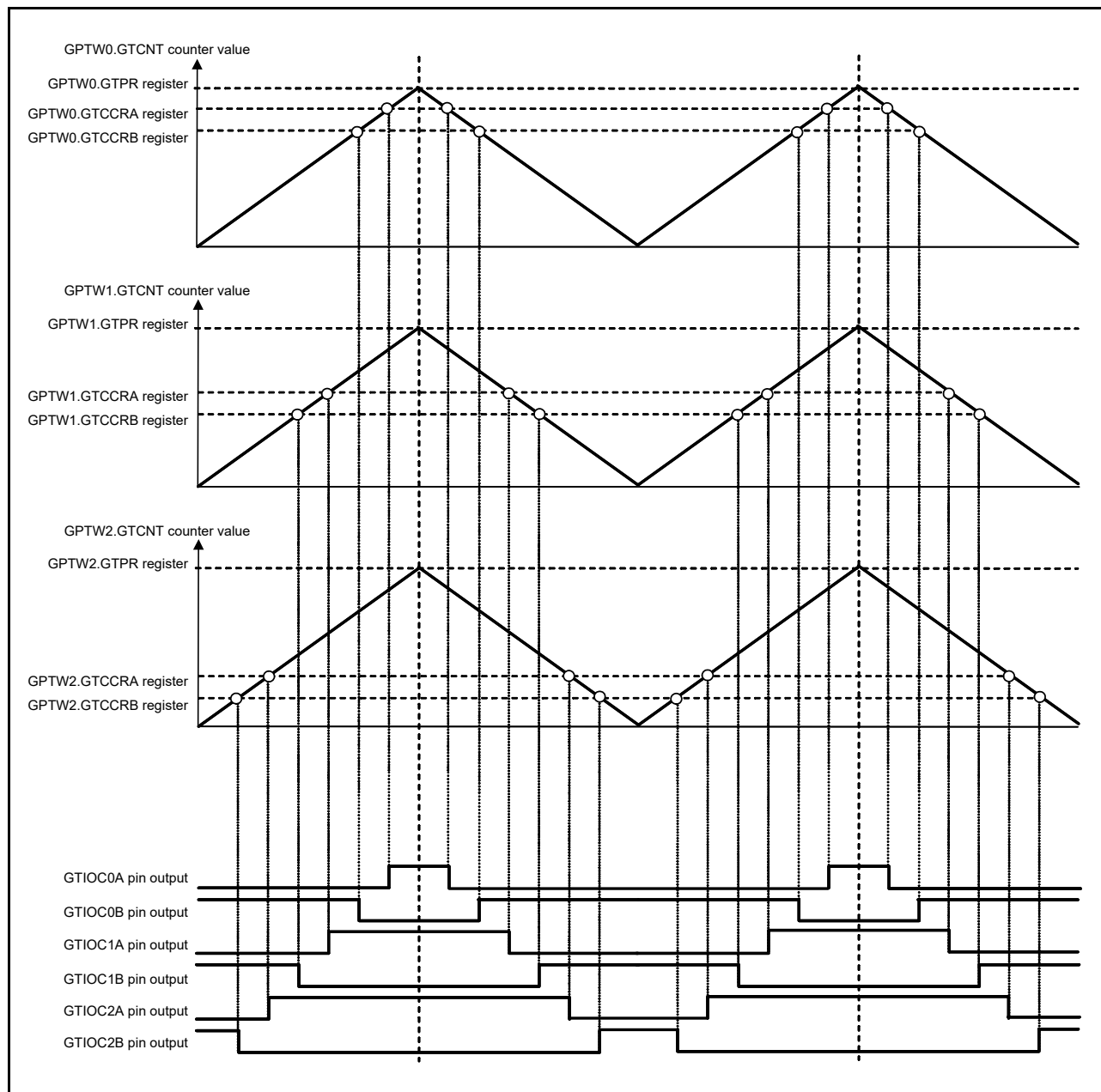


Figure 21.82 Example of Three-Phase Triangle-Wave Complementary PWM Output

(5) Three-Phase Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 21.83 shows an example in which three channels perform synchronous operation in triangle-wave PWM mode 1 with automatic dead time setting and three-phase complementary PWM waveforms are output. The GTIOCnA pin ($n = 0$ to 2) is set so that it will output low as the initial output, toggle the output at a GTCCRA register compare match, and retain the output at the end of the cycle. The GTIOCnB pin is set so that it will output high as the initial output, toggle the output at a GTCCRB register compare match, and retain the output at the end of the cycle.

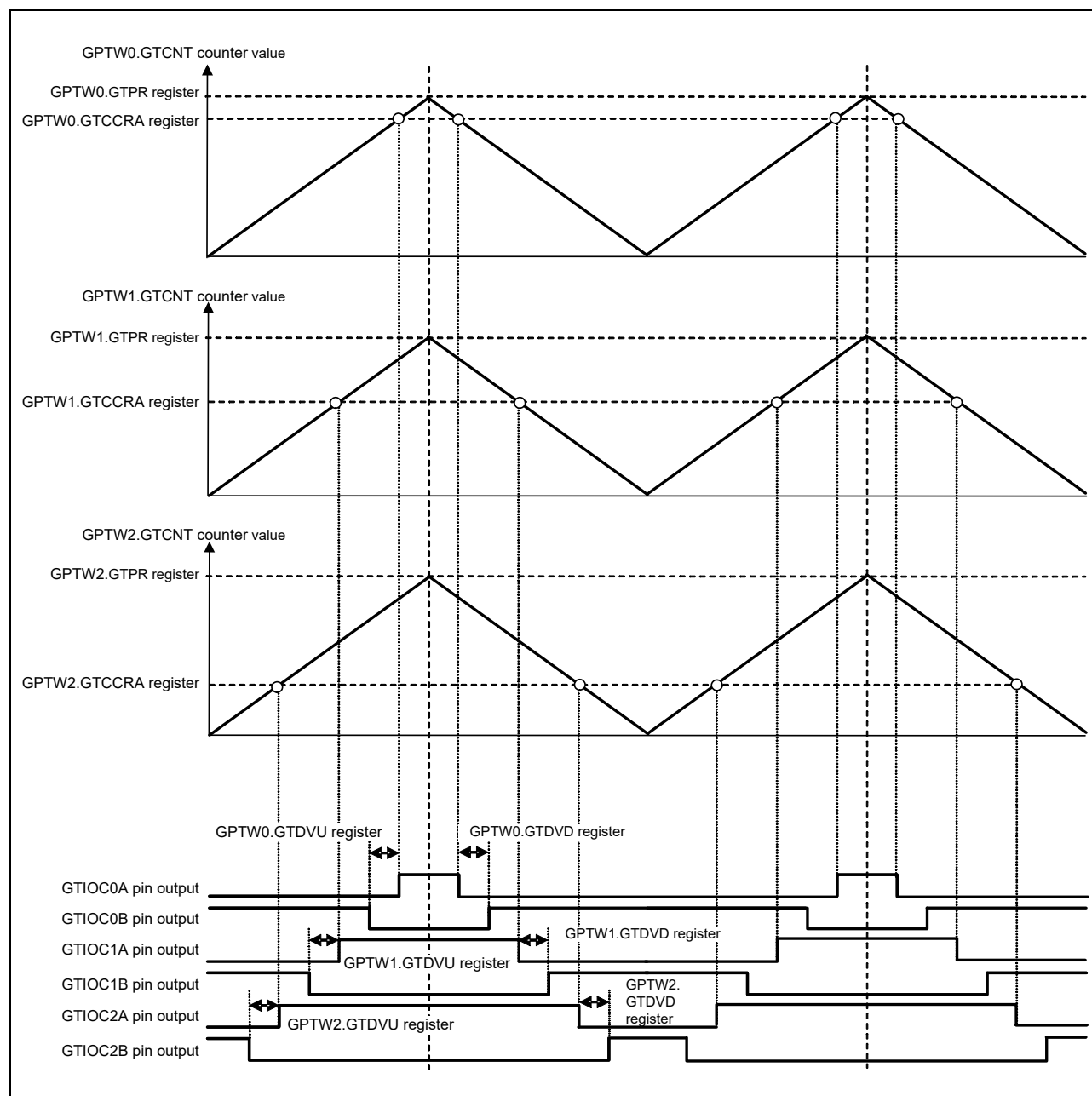


Figure 21.83 Example of Three-Phase Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

(6) Three-Phase Asymmetric Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 21.84 shows an example in which three channels perform synchronous operation in triangle-wave PWM mode 3 with automatic dead time setting and three-phase complementary PWM waveforms are output. The GTIOCnA pin ($n = 0$ to 2) is set so that it will output low as the initial output, toggle the output at a GTCCRA register compare match, and retain the output at the end of the cycle. The GTIOCnB pin is set so that it will output high as the initial output, toggle the output at a GTCCRB register compare match, and retain the output at the end of the cycle.

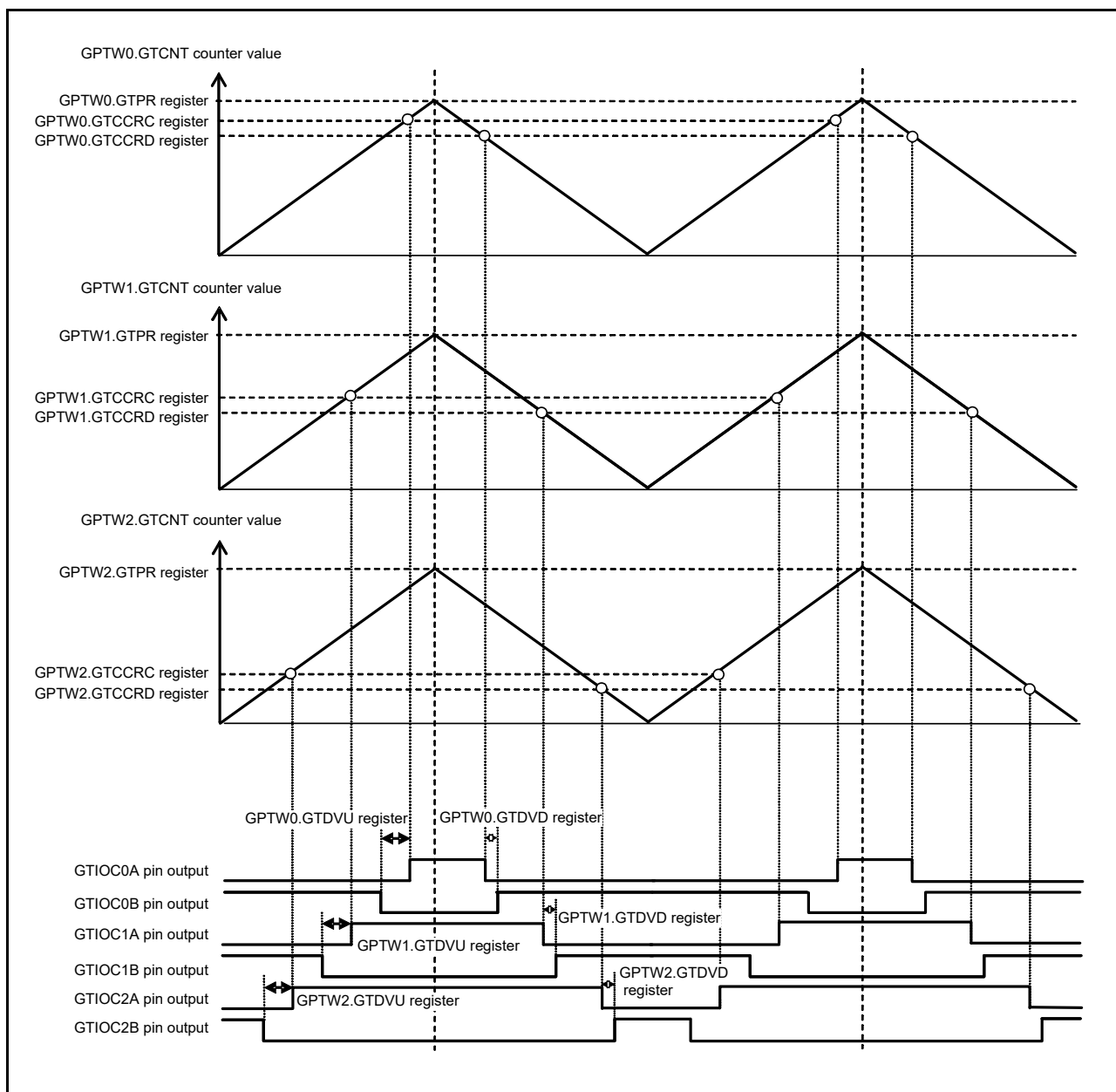


Figure 21.84 Example of Three-Phase Asymmetric Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

21.4 Interrupt Sources

21.4.1 Interrupt Sources and Priorities

There are three kinds of interrupt sources; GTCCRm register (m = A to F) input capture/compare match, GTCNT counter overflow (GTPR register compare match)/underflow, and dead time error.

Each interrupt source has its own status flag and control bit for generating interrupt request signals, where the generation of interrupt request signals can be enabled or disabled individually.

When an interrupt source condition is satisfied, the corresponding status flag in GTST is set to 1, and an interrupt request is generated if the corresponding interrupt request enable or disable bit in the GTINTAD register is 1.

The corresponding status flag in the GTST register is writable to clear. If flag set and flag clear happens at the same time, flag clear have a priority to flag set.

For details, refer to section 14, Interrupt Controller (ICUb). Table 21.8 shows a list of GPTW interrupt sources.

Table 21.8 GPTW Interrupt Sources (n = 0 to 2)

Channel	Name	Interrupt Source	Interrupt Flag
GPTWn	GTClAn	GTCCRA register input capture/compare match	GPTWn.GTST.TCFA
	GTClBn	GTCCRB register input capture/compare match	GPTWn.GTST.TCFB
	GTClCn	GTCCRC register compare match	GPTWn.GTST.TCFC
	GTClDn	GTCCRD register compare match	GPTWn.GTST.TCFD
	GDTEn	Dead time error	GPTWn.GTST.DTEF
	GTClEn	GTCCRE register compare match	GPTWn.GTST.TCFE
	GTClFn	GTCCRF register compare match	GPTWn.GTST.TCFF
	GTClVn	GTCNT counter overflow (GTPR register compare match)	GPTWn.GTST.TCFPO
	GTClUn	GTCNT counter underflow	GPTWn.GTST.TCFPU

(1) GTClAn Interrupt (n = 0 to 2)

When the GTINTAD.GTINTA bit is 1, an interrupt request is generated under the following conditions.

- When the GTCCRA register functions as a compare match register, the GTCNT counter value matches with the GTCCRA register.
- When the GTCCRA register functions as an input capture register, the input-capture signal has caused transfer of the GTCNT counter value to the GTCCRA register.

(2) GTClBn Interrupt (n = 0 to 2)

When the GTINTAD.GTINTB bit is 1, an interrupt request is generated under the following conditions.

- When the GTCCRB register functions as a compare match register, the GTCNT counter value matches with the GTCCRB register.
- When the GTCCRB register functions as an input capture register, the input-capture signal has caused transfer of the GTCNT counter value to the GTCCRB register.

(3) GTClCn Interrupt (n = 0 to 2)

When the GTINTAD.GTINTC bit is 1, an interrupt request is generated under the following condition.

- When the GTCCRC register functions as a compare match register, the GTCNT counter value matches with the GTCCRC register.

A compare match is not performed and thus interrupt is not requested under the following conditions.

- GTCR.MD[2:0] bits = 001b (sawtooth-wave one-shot pulse mode)

- GTCR.MD[2:0] bits = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] bits = 01b, 10b, 11b (buffer operation with the GTCCRC register)

(4) GTCIDn Interrupt (n = 0 to 2)

When the GTINTAD.GTINTD bit is 1, an interrupt request is generated under the following condition.

- When the GTCCRD register functions as a compare match register, the GTCNT counter value matches with the GTCCRD register.

A compare match is not performed and thus interrupt is not requested under the following conditions.

- GTCR.MD[2:0] bits = 001b (sawtooth-wave one-shot pulse mode)
- GTCR.MD[2:0] bits = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] bits = 10b, 11b (buffer operation with the GTCCRD register)

(5) GTCIE n Interrupt (n = 0 to 2)

When the GTINTAD.GTINTE bit is 1, an interrupt request is generated under the following condition.

- When the GTCCRE register functions as a compare match register, the GTCNT counter value matches with the GTCCRE register.

A compare match is not performed and thus interrupt is not requested under the following conditions.

- GTCR.MD[2:0] bits = 001b (sawtooth-wave one-shot pulse mode)
- GTCR.MD[2:0] bits = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] bits = 01b, 10b, 11b (buffer operation with the GTCCRE register)

(6) GTCIFn Interrupt (n = 0 to 2)

When the GTINTAD.GTINTF bit is 1, an interrupt request is generated under the following condition.

- When the GTCCRF register functions as a compare match register, the GTCNT counter value matches with the GTCCRF register.

A compare match is not performed and thus interrupt is not requested under the following conditions.

- GTCR.MD[2:0] bits = 001b (sawtooth-wave one-shot pulse mode)
- GTCR.MD[2:0] bits = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] bits = 10b, 11b (buffer operation with the GTCCRF register)

(7) GTCIVn Interrupt (n = 0 to 2)

When the GTINTAD.GTINTPR[0] bit is 1, an interrupt request is generated under the following conditions.

- In sawtooth-wave mode, interrupt requests are enabled at overflows (the GTCNT counter value changes from the GTPR register value to 0 during up-counting).
- In triangle-wave mode, interrupt requests are enabled at crests (the GTCNT counter value changes from the GTPR register value to the GTPR register value minus 1).
- In counting driven by a hardware source, interrupt requests are enabled on overflows (that is, when the GTCNT counter value changes from the GTPR register value to 0 during up-counting).

(8) GTCIU n Interrupt (n = 0 to 2)

When the GTINTAD.GTINTPR[1] bit is 1, an interrupt request is generated under the following conditions.

- In sawtooth-wave mode, interrupt requests are enabled at underflows (the GTCNT counter value changes from 0 to the GTPR register value during down-counting).

- In triangle-wave mode, interrupt requests are enabled at troughs (the GTCNT counter value changes from 0 to 1).
- In count operation by a hardware source, interrupt requests are enabled at underflows (the GTCNT counter value changes from 0 to the GTPR register value during down-counting).

(9) GDTE_n Interrupt (n = 0 to 2)

When automatic dead time setting has been made, the GTST.DTEF flag becomes 1 when the timer output toggle point with dead time added exceeds the count period. If the GTINTAD.GRPDTE bit is 1 at this time, a dead time error interrupt request (GDTE) is generated.

In addition, when the timer output toggle point with dead time added is back within the count period, the GTST.DTEF flag changes from 1 to 0.

Table 21.9 Relationship between Interrupt Signals and Interrupt Enable Bits (n = 0 to 2)

Interrupt Signal	Interrupt Enable Bit	Status Flag
GTCIA _n	GTINTAD.GTINTA bit	GTST.TCFA Flag
GTCIB _n	GTINTAD.GTINTB bit	GTST.TCFB Flag
GTCIC _n	GTINTAD.GTINTC bit	GTST.TCFC Flag
GTCID _n	GTINTAD.GTINTD bit	GTST.TCFD Flag
GDTE _n	GTINTAD.GRPDTE bit	GTST.DTEF Flag
GTCIE _n	GTINTAD.GTINTE bit	GTST.TCFE Flag
GTCIF _n	GTINTAD.GTINTF bit	GTST.TCFF Flag
GTCIV _n	GTINTAD.GTINTPR[1:0] bits	GTST.TCFPO Flag
GTCIU _n		GTST.TCFPU Flag

21.4.2 DTC Activation

The DTC can be triggered by the interrupt request in each channel. For details, see section 14, Interrupt Controller (ICUb), section 16, Data Transfer Controller (DTCb).

21.4.3 Interrupt and A/D Conversion Start Request Skipping Function

21.4.3.1 Interrupt Skipping Function by GTITC Register

By setting the GTITC register, the GTCNT counter overflow (GTPR register compare match) interrupt (GTCIV) and underflow interrupt (GTCIU) can be skipped. Other interrupts and A/D conversion start request signals can be skipped in coordination with the GTCIV/GTCIU skipping function. However, the dead time error interrupts cannot be linked with the GTCIV/GTCIU skipping function.

When the interrupt is skipped, the updating of relevant status flag is also skipped. The interrupt skipping is continued even if the status flag is set to 1.

The interrupt skipping function is related only to the GTITC register setting, and is not related to setting of the GTINTAD register interrupt enable bit. The interrupt enable bit is used only to control the output of the interrupt signal after thinning out.

When both troughs and crests are counted and skipped in triangle-wave mode, if the number of times of skipping is odd, GTCIV/GTCIU interrupt requests cannot be generated at troughs only or at crests only depending on the skipping counter start timing. Therefore, in order to count both troughs and crests and generate the GTCIV/GTCIU interrupts at troughs only or crests only in triangle-wave mode, the number of times of skipping should be even.

Similarly, in sawtooth-wave mode, when both overflows and underflows are counted and skipped with the count direction changed, GTCIV/GTCIU interrupt requests cannot be generated at overflows only or at underflows only. Therefore, in order to count both overflows and underflows with the count direction changed and generate the GTCIV/GTCIU interrupts at overflows only or underflows only in sawtooth-wave mode, the skipping state should be carefully checked before using.

When changing the skipping count, be sure to release the skipping count setting (GTITC.IVTC[1:0] bits = 00b).

Figure 21.85 to Figure 21.90 show examples of skipping function operation.

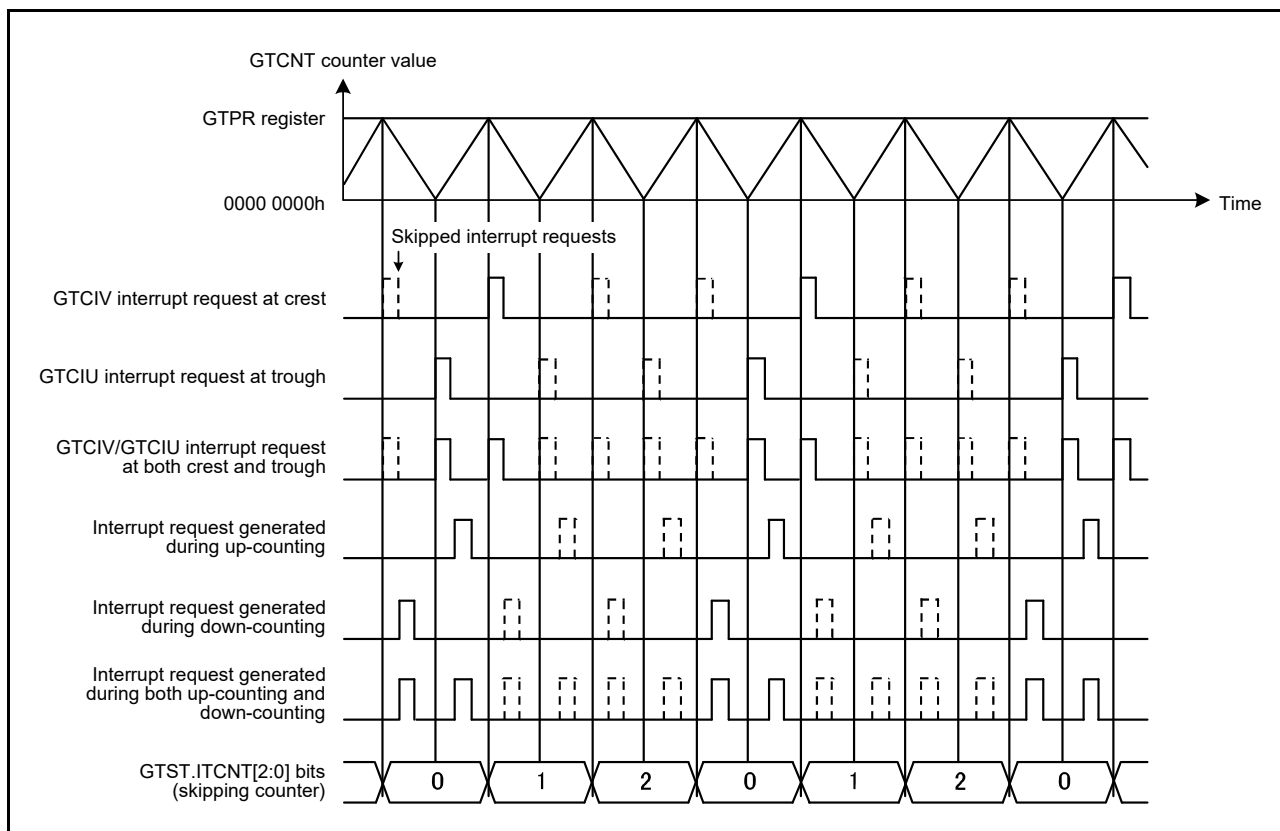


Figure 21.85 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Crests, Skipping Count: 2)

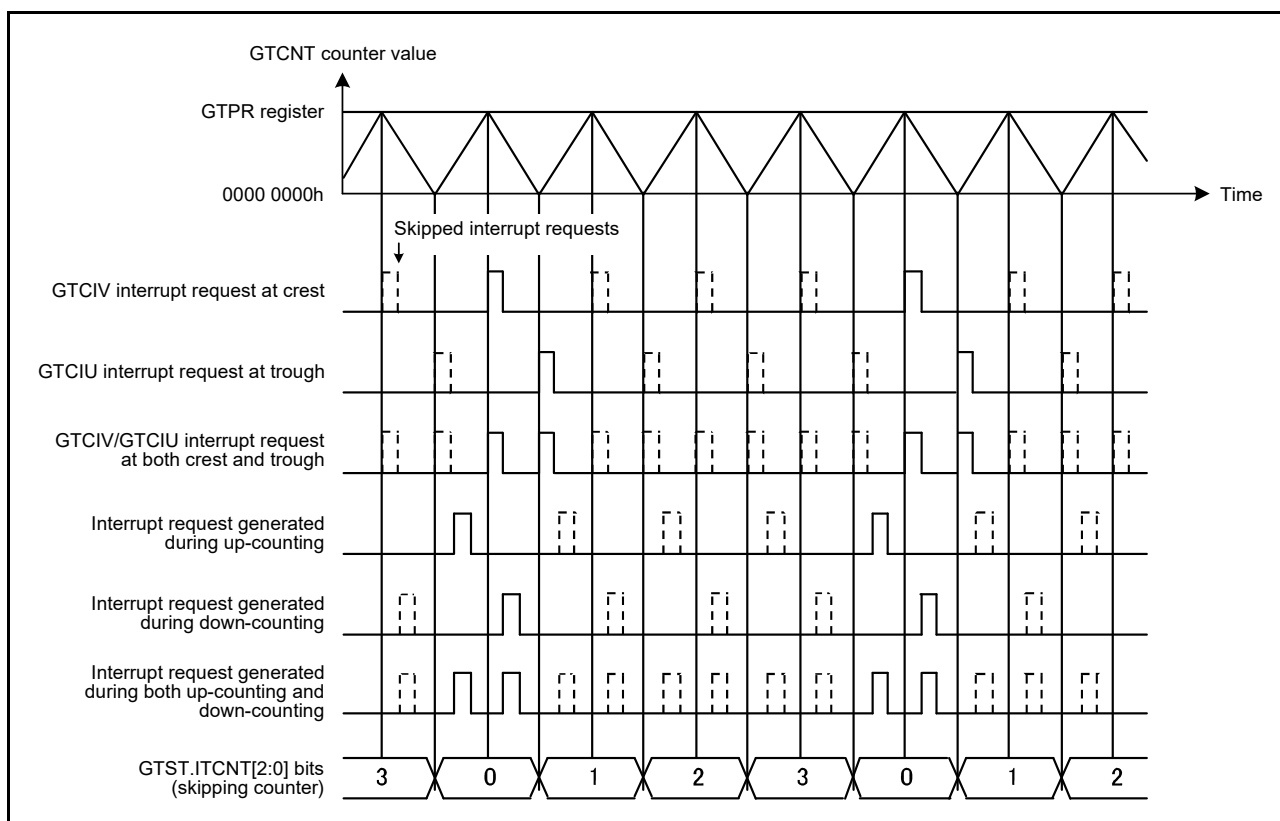


Figure 21.86 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Troughs, Skipping Count: 3)

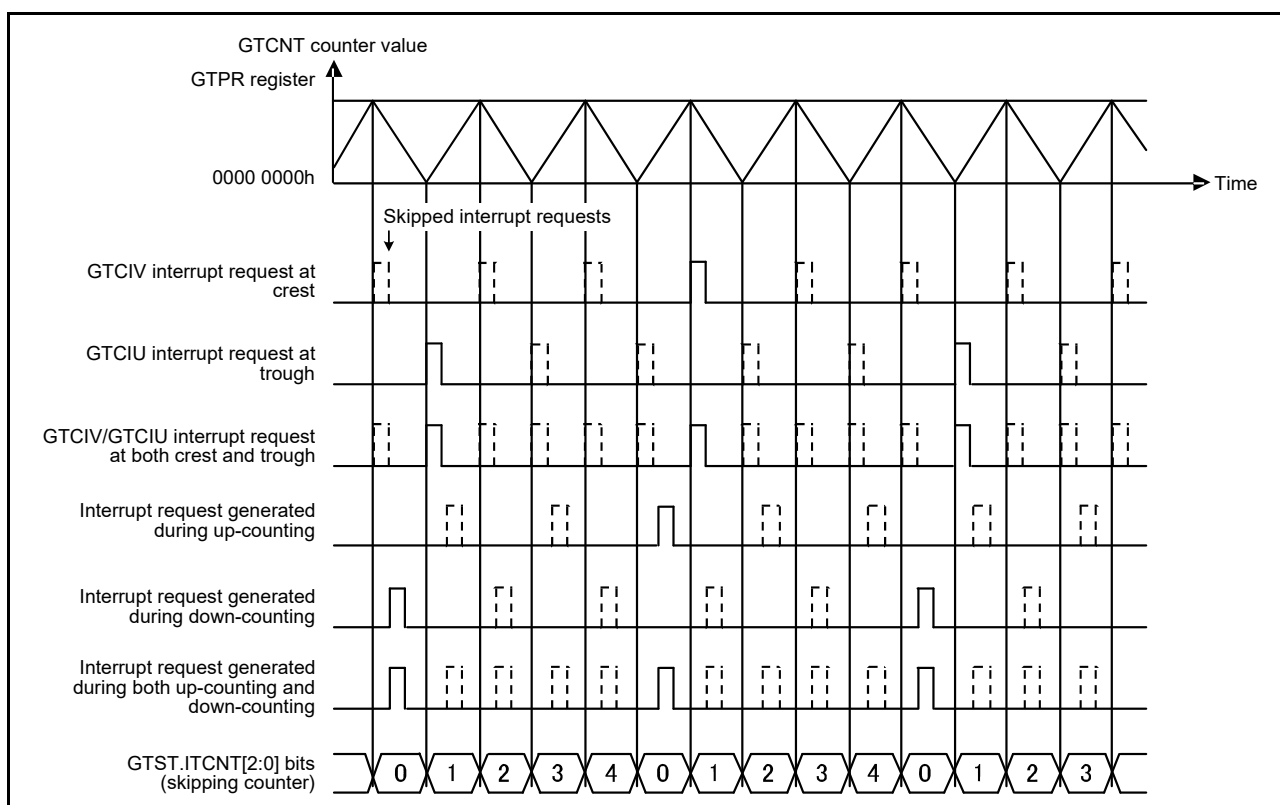


Figure 21.87 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Both Troughs and Crests, Skipping Count: 4)

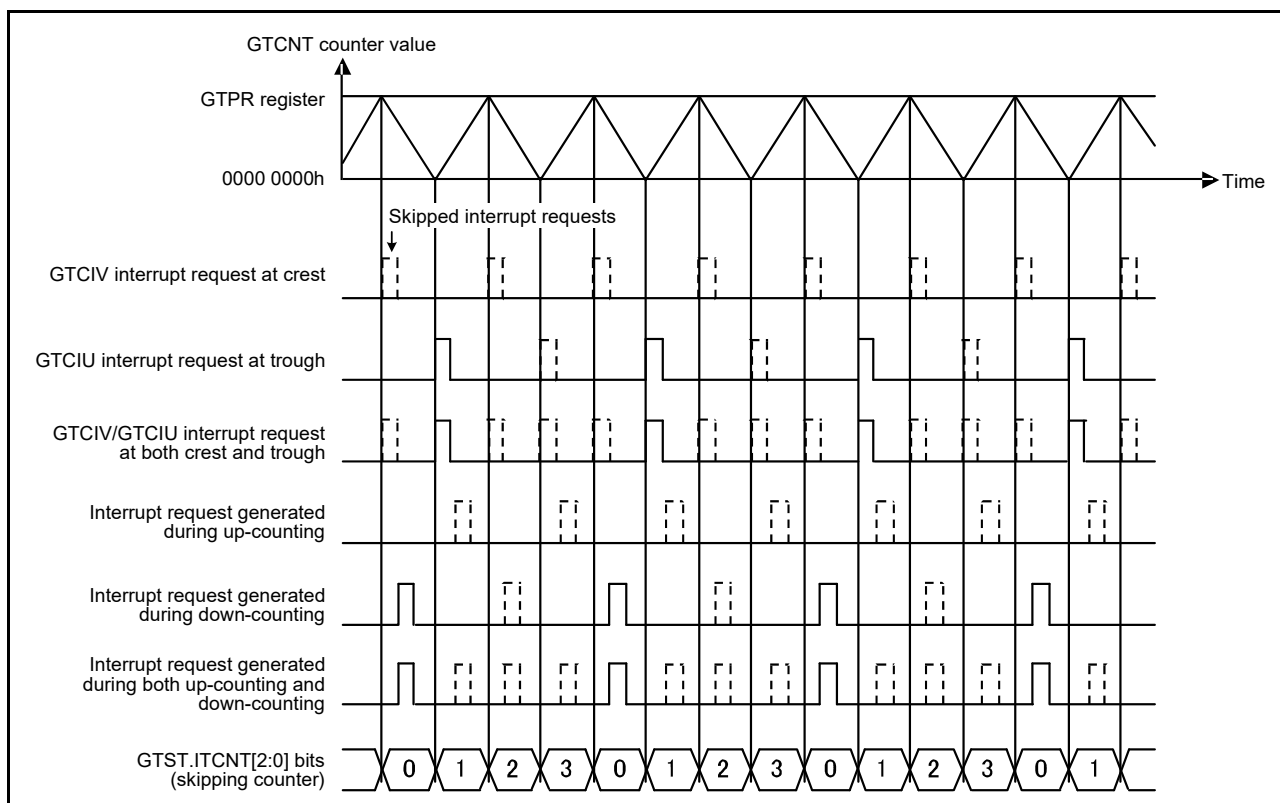


Figure 21.88 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Both Troughs and Crests, Skipping Count: 3, Skipping Started at Up-Counting)

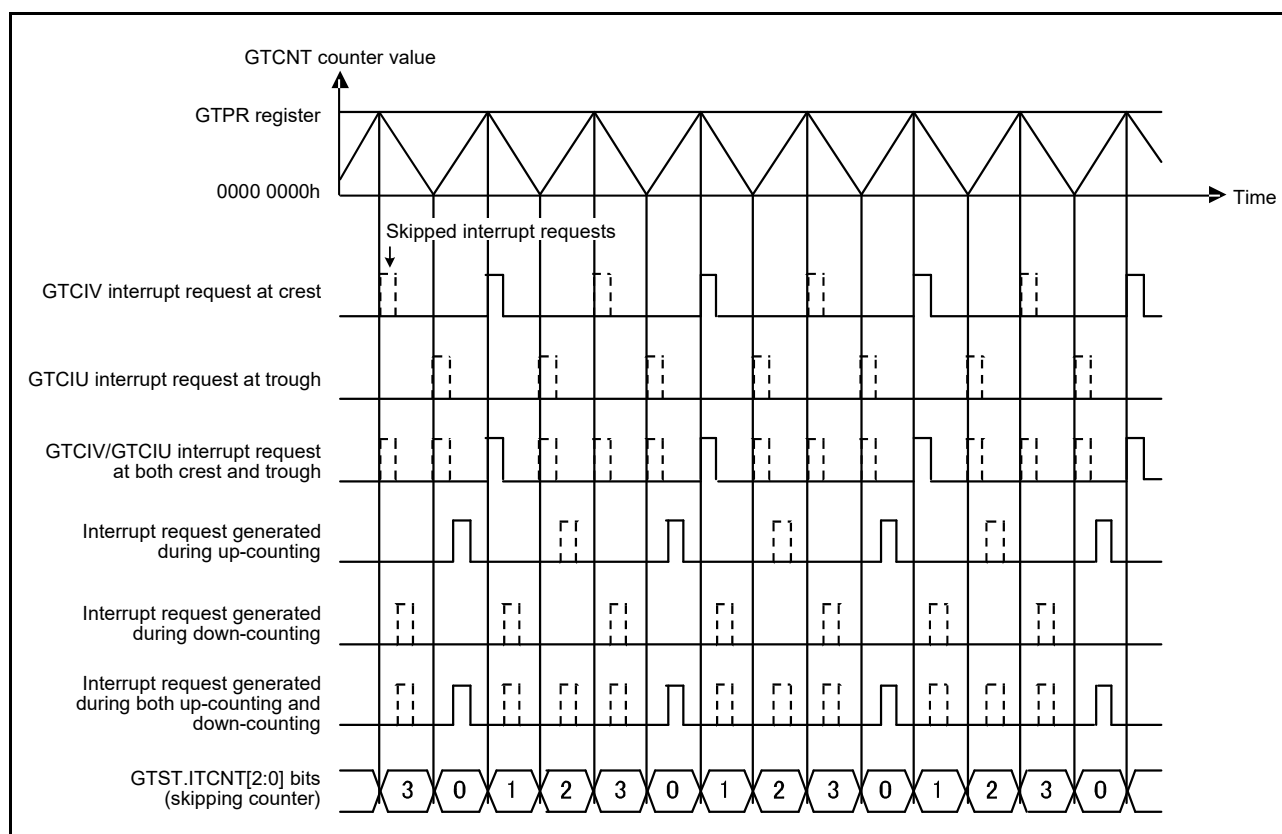


Figure 21.89 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Both Troughs and Crests, Skipping Count: 3, Skipping Started at Down-Counting)

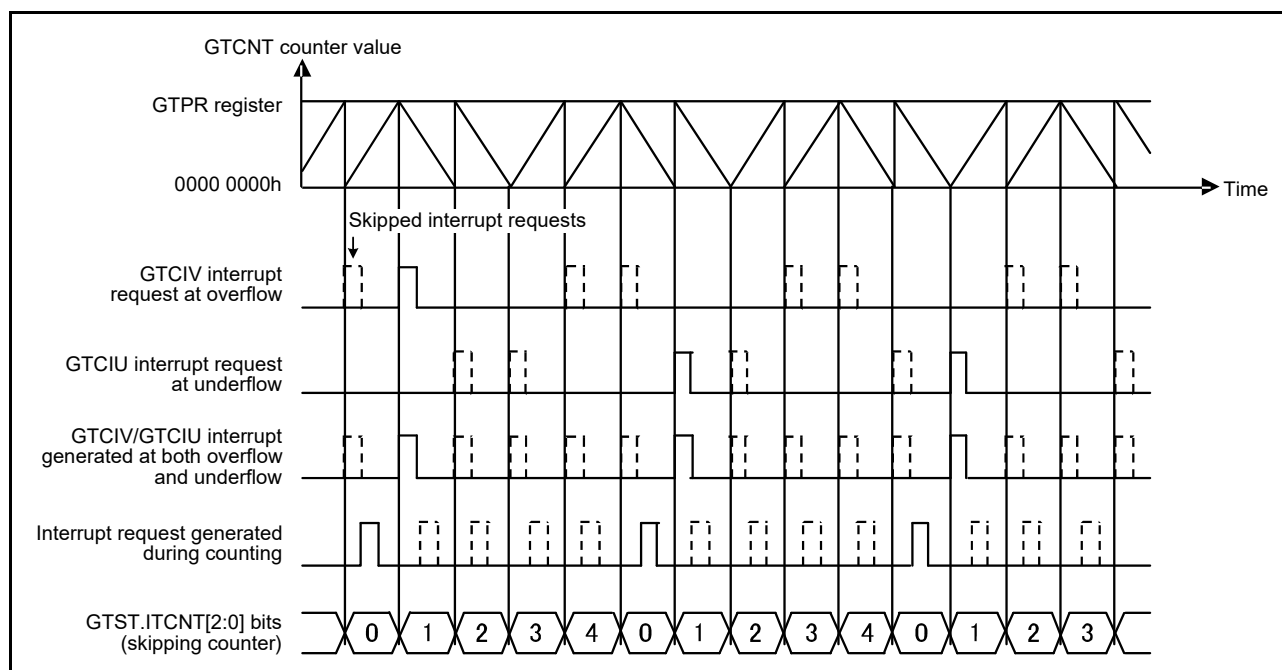


Figure 21.90 Example of Interrupt Skipping Function Operation (Sawtooth Waves, Operation with Count Direction Changed, Counting and Skipping Both Overflows and Underflows, Skipping Count: 4)

21.5 A/D Conversion Start Request

An A/D conversion start request can be issued at a compare match between the GTCNT counter and the GTADTRA or GTADTRB register, up-counting only, down-counting only, or both up-counting and down-counting can be specified by setting the GTINTAD register.

The GTADTRA and GTADTRB registers each has two buffer registers. Buffer operation with the GTADTRA register used together with the GTADTBRA and GTADTDBRA registers, and buffer operation with the GTADTRB register used together with the GTADTBRB and GTADTDBRB registers can be performed.

The timing of the generation of requests to start A/D conversion can be monitored by an external pin. When the A/D conversion start request signal to be monitored is selected in the GTADSMR.ADSMSk bit ($k = 0, 1$) and when the output is enabled in the ADSMENk bit, a signal is output synchronized with a cycle frame of the timer used to generate the A/D conversion start request signal, of which the output is driven high at the generation of the A/D conversion start request signal by the GTADSMk pin, or at the end of the cycle of which the output is driven low. When a signal to request the start of A/D conversion is generated at the end of the cycle, the generation of this signal has priority in terms of monitoring output and the output remains at the high level till the end of the next cycle. The registers (GTADTRA and GTADTRB) that are sources of generating the A/D conversion start request signals and their counting directions can be checked by the A/D conversion start request flags (ADTRAUF, ADTRAUF, ADTRBUF, and ADTRBDF) in the GTST register. When the output of the same A/D conversion start request signal monitoring output is enabled for multiple channels, ORed signals will be output from the GPTW.

Figure 21.91 shows an example of A/D conversion start request operation, Figure 21.92 shows an example for setting A/D conversion start request operation.

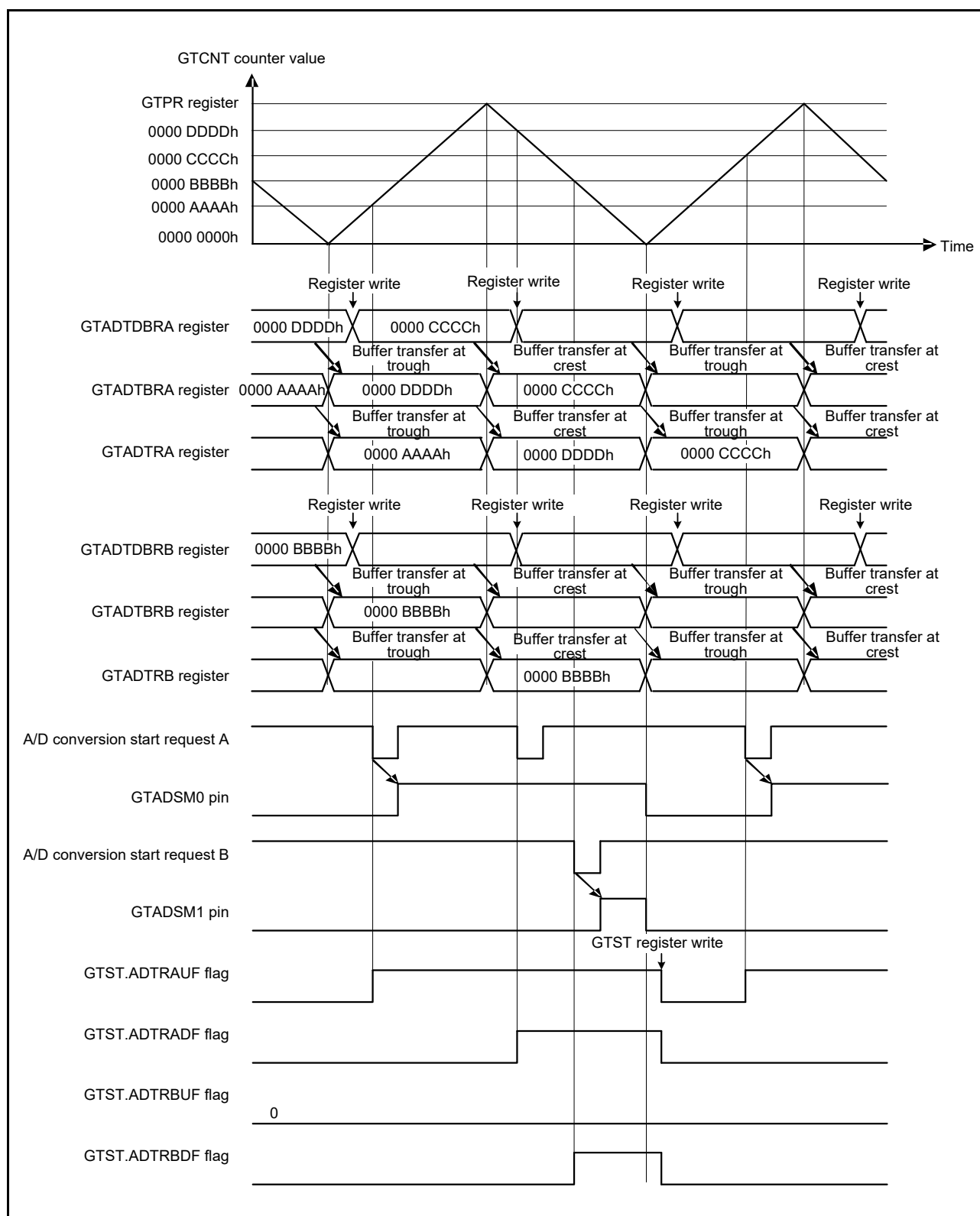


Figure 21.91 Example of A/D Conversion Start Request Timing Operation
 (Triangle Waves, Double Buffer Operation, Buffer Transfer at Both Troughs and Crests, A/D Conversion Start Requested by GTADTRA Register at Both Up-Counting and Down-Counting, A/D Conversion Start Requested by GTADTRB Register at Down-Counting, Monitoring of the GTADTRA Register Up-Counting by the GTADSM0 Pin, Monitoring of the GTADTRB Register Down-Counting by the GTADSM1 Pin)

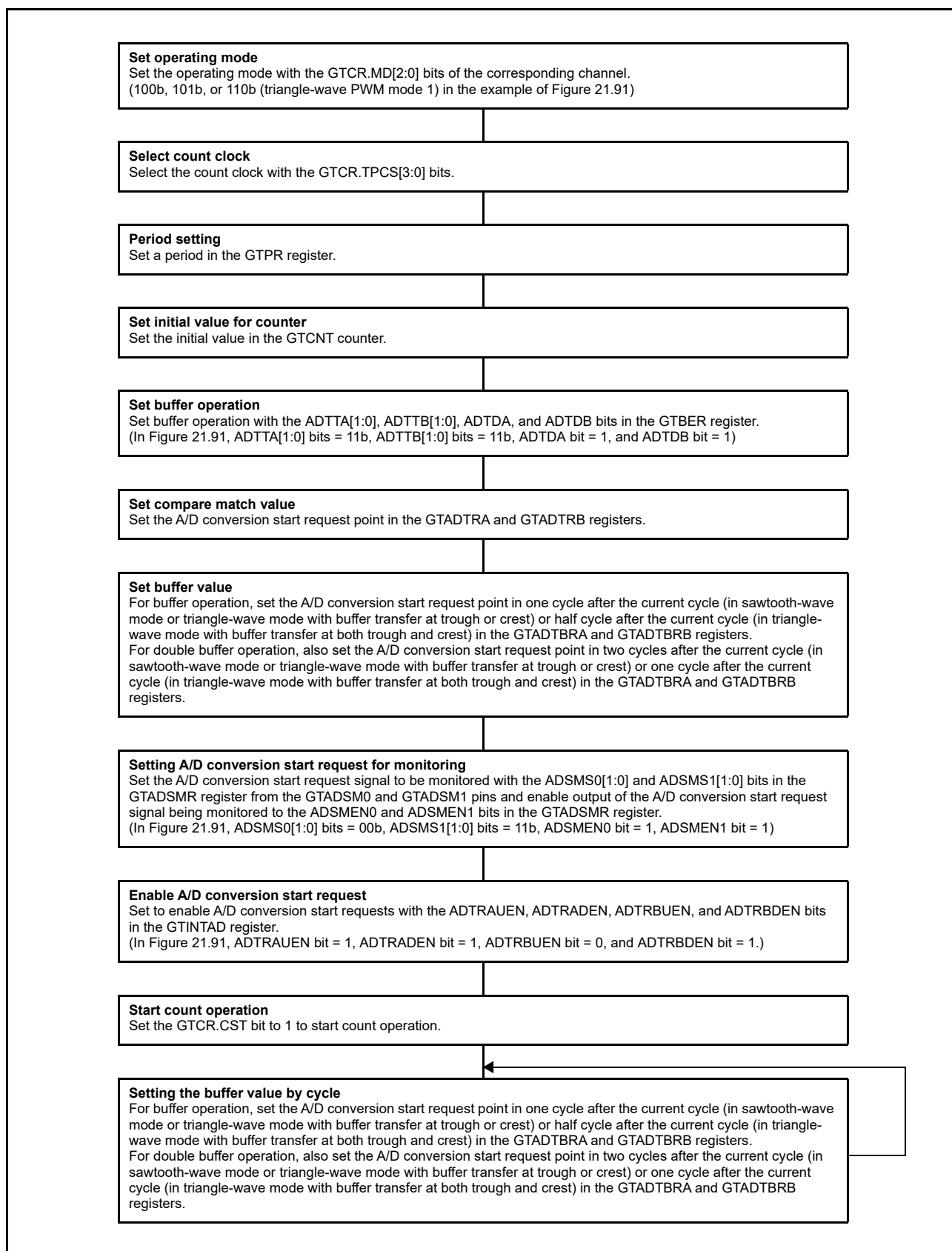


Figure 21.92 Example for Setting A/D Conversion Start Request Operation

21.6 Noise Filter Function

Each pin for use in input capture to the GPTW is equipped with a noise filter. The noise filter samples input signals at the sampling clock and removes the pulses of which length is less than three sampling periods.

The noise filter functionality includes enabling and disabling of the noise filter for each pin and setting of the sampling clock for each channel.

Figure 21.93 shows the timing of noise filtering.

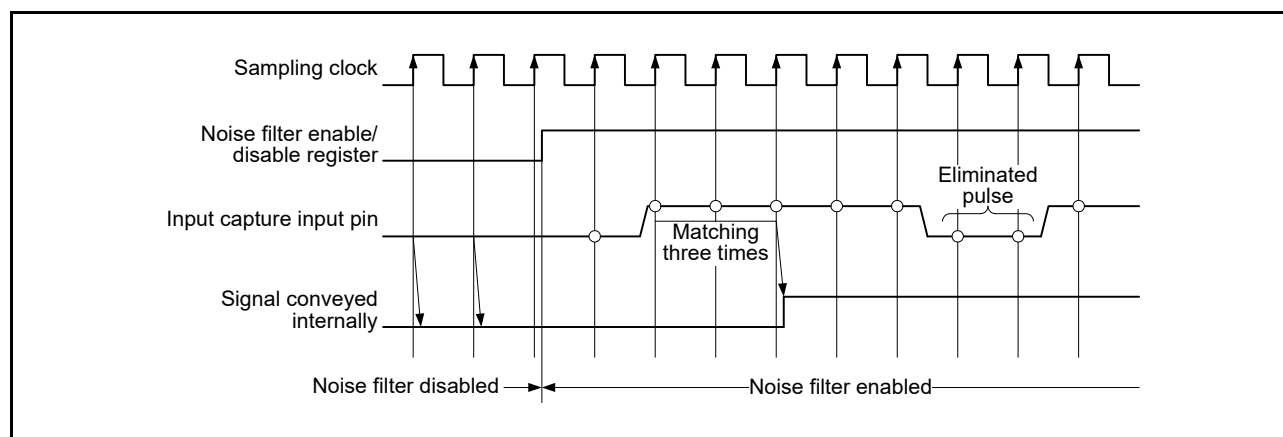


Figure 21.93 Timing of Noise Filtering

If noise filtering is enabled, input capture operation is performed on the edges of noise-filtered signal after a delay of $(\text{minimum sampling interval} \times 2 + \text{PCLKB})$ due to noise filtering for the input capture input.

21.7 Protection Function

21.7.1 Write-Protection for Registers

In order to avoid incorrect writing to registers, write access to registers can be enabled or disabled per channel by setting the GTWP.WP bit for the given channel.

The WP bits enable or disable writing to the registers listed below.

GTSSR, GTPSR, GTCSR, GTICASR, GTICBSR, GTCR, GTUDDTYC, GTIOR, GTINTAD, GTST, GTBER, GTITC, GTCNT, GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, GTCCRF, GTPR, GTPBR, GTPDBR, GTADTRA, GTADTBRA, GTADTDBRA, GTADTRB, GTADTBRB, GTADTDBRB, GTDTCR, GTDVU, GTDVD, GTDBU, GTDBD, GTSOS, GTSOTR, GTADSMR, GTBER2, GTICCR

Every bit in registers GTSTR, GTSTP, and GTCLR, which can update the corresponding registers in other channels and can be updated by any of the corresponding registers in other channels conversely, can be protected by setting the GTWP.STRWP, STWP, and CLRWP bits, respectively, per channel.

Likewise, writing to the GTSECSR and GTSECR registers, which can control all channels by writing to the GTSECSR and GTSECR registers of a given channel, can be enabled or disabled by the setting of the GTWP.CMNWP bit.

Protection by the GTWP register only targets writing operation by the CPU. Updating the register generated in related to the CPU writing is not protected.

21.7.2 Disabling of Buffer Operation

If the timing of buffer register write is too late for the buffer transfer timing, buffer operation can be suspended with setting the BD[0], BD[1], BD[2], and BD[3] bits in the GTBER register.

Specifically, buffer transfer can be temporarily disabled, even though a buffer transfer condition is generated during buffer register write, by setting the BD[0], BD[1], BD[2], and BD[3] bits to 1 (buffer operation disabled) before buffer register write, and setting the bits to 0 (buffer operation enabled) after completion of writing to all the buffer registers.

The BD[0], BD[1], BD[2], and BD[3] bits can be set on channel basis by writing directly to the GTBER register or can be set simultaneously by setting the GTSECR register for multiple channels which were set by the GTSECSR register.

Figure 21.94 shows an example of operation for disabling buffer operation.

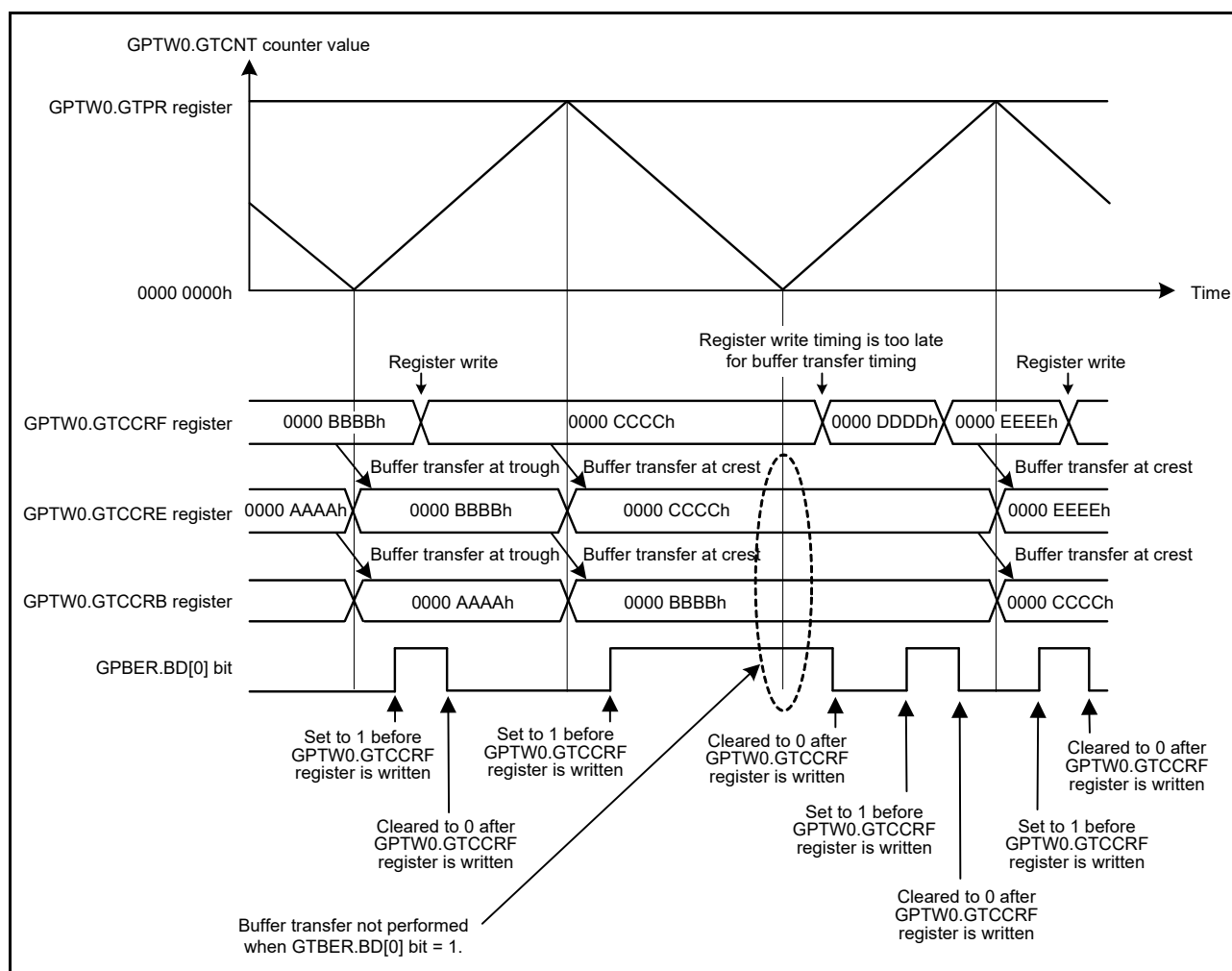


Figure 21.94 Example of Operation for Disabling Buffer Operation (Triangle Waves, Double Buffer Operation, Buffer Transfer at Both Troughs and Crests)

21.7.2.1 Simultaneous Control of Buffer Operations of Multiple Channels

The GTBER.BD bit can be set by writing directly to the GTBER register per channel or by making settings in the GTSECR register for multiple channels that have already set in the GTSECSR register.

Follow the procedure below to simultaneously set the GTBER.BD bits of multiple channels.

(1) Select the channels for simultaneously setting of the GTSECSR register

Set the GTSECSR register so that the values at the bit positions for the corresponding channels for simultaneously setting of the GTBER.BD bits become 1. All GTSECSR registers can be updated by writing to the GTSECSR register of any channel.

(2) Simultaneously set the GTBER.BD bits by updating the GTSECR register

In the GTSECR register, set the operation of the GTBER.BD bits (enabling or disabling of buffer operation) which are to be simultaneously set. Writing to a GTSECR register from any channel updates the GTBER.BD bits in all channels corresponding to the bits set as 1 in the GTSECSR register, in accordance with the value of the GTSECR register. Figure 21.95 and Figure 21.96 show examples of simultaneously controlling the enabling or disabling of buffer operation for multiple channels.

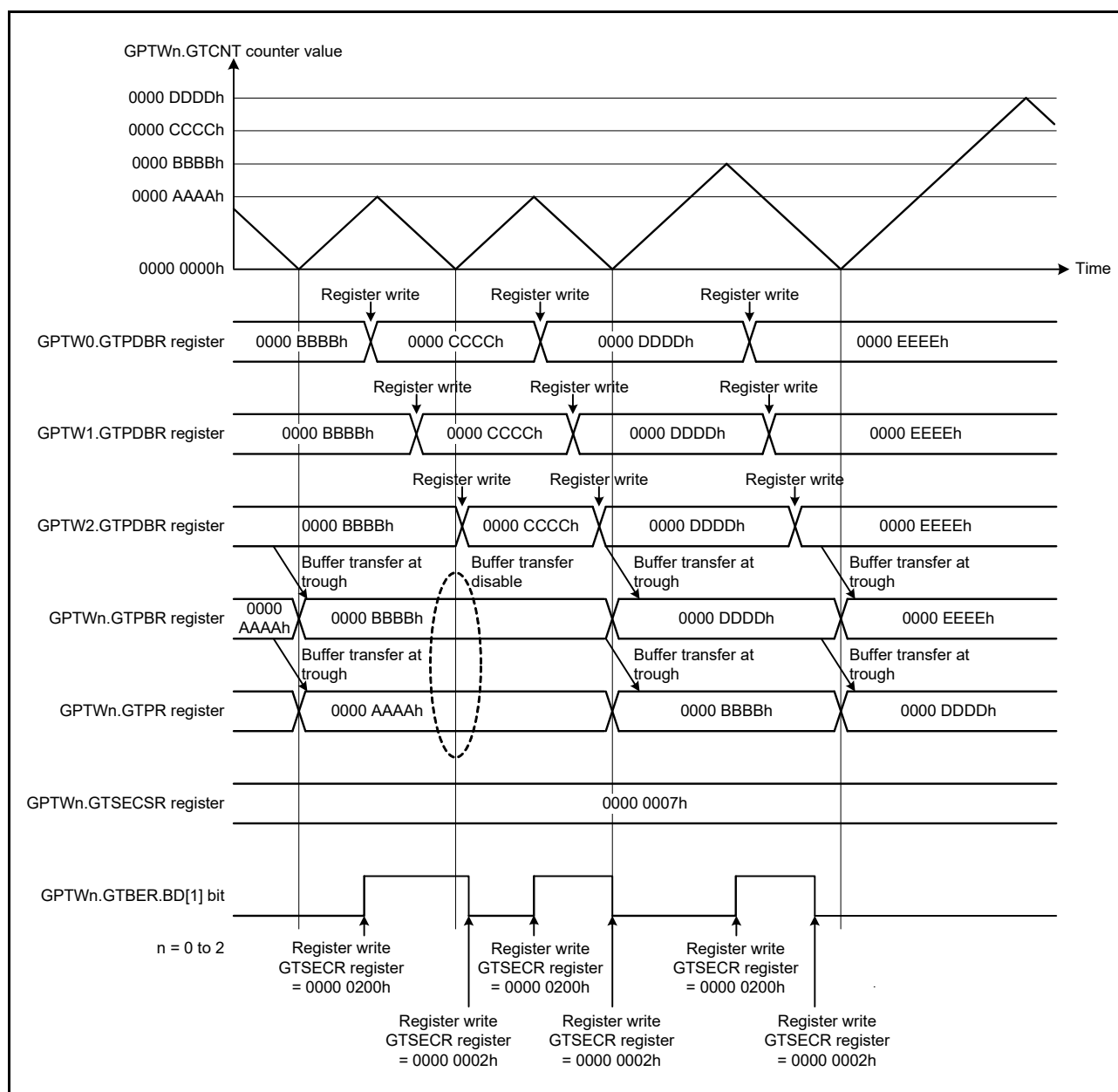


Figure 21.95 Example of Multiple Channel Operation for Simultaneous Control of Buffer Operation (Triangle Waves, Double Buffer Operation)

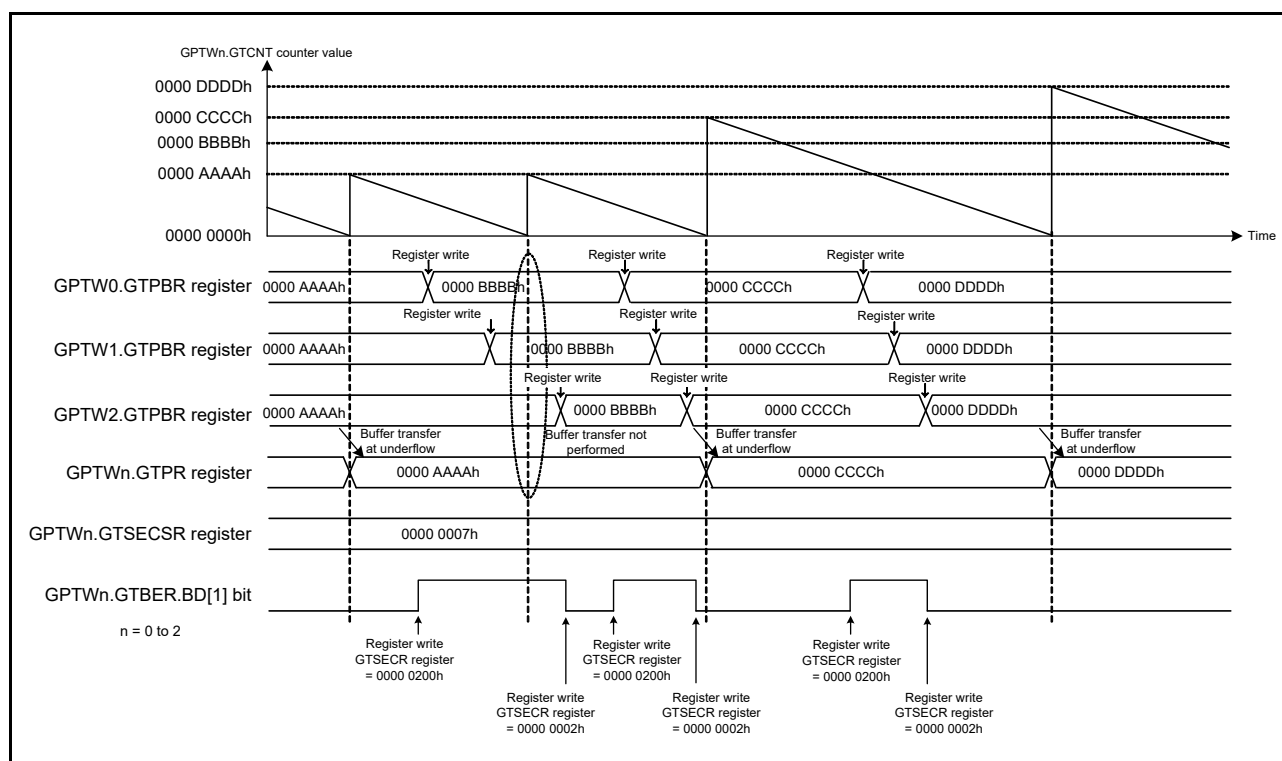


Figure 21.96 Example of Multiple Channel Operation for Simultaneous Control of Buffer Operation (Sawtooth Waves, Single Buffer Operation)

21.7.2.2 Repeated Double-Buffered Operation When Disabling GTCCR Buffer Transfer

When a GTBER.DBRTEC_m ($m = A, B$) bit is set to 1 in sawtooth-wave one-shot pulse mode or triangle-wave PWM mode 3, transfer from the intermediate buffer to the GTCCR_m ($m = A, B$) register is repeated on a cyclic basis even while buffer transfer is disabled by the setting of the GTBER.BD[0] bit.

(1) In sawtooth-wave one-shot pulse mode

In sawtooth-wave one-shot pulse mode, the compare match value for the first half of a waveform is stored in temporary register x ($x = C, E$) as the intermediate buffer for the GTCCR_x ($x = C, E$) register and the compare match value for the second half of a waveform is stored in temporary register m ($m = A, B$) as the intermediate buffer for the GTCCR_y ($y = D, F$) register, respectively, for compare match values during repeated buffer operation, and the given values are alternately transferred to the GTCCR_m ($m = A, B$) registers.

Table 21.10 lists the types of buffer transfer of the GTCCR register during counting in sawtooth-wave one-shot pulse mode. While counting is stopped, the setting of the value in the temporary register is transferred through forcible buffer transfer.

In forcible buffer transfer, the values of the GTCCR_y ($y = D, F$) registers are transferred to temporary registers m ($m = A, B$), and the values of the GTCCR_x ($x = C, E$) are transferred to temporary registers x ($x = C, E$), when the setting of the corresponding GTBER.DBRTEC_m ($m = A, B$) bit is 1, respectively.

When the setting of the GTBER.DBRTEC_m ($m = A, B$) bit is 1, values written by the CPU to the GTCCR_m ($m = A, B$) registers are reflected as the values of temporary registers x ($x = C, E$).

Table 21.10 GTCCR Buffer Operation in Sawtooth-Wave One-Shot Pulse Mode

GTBER.DBRTEC _m	Buffer transfer	Timing of transfer				
		GTCCR _x ↓ GTCCR _m	GTCCR _x ↓ Temporary register x	Temporary register x ↓ GTCCR _m	GTCCR _y ↓ Temporary register m	Temporary register m ↓ GTCCR _m
0	Transfer-enabled period	Overflow or underflow	No transfer	No transfer	Overflow or underflow	GTCCR _m compare match
	Transfer-disabled period	No transfer	No transfer	No transfer	No transfer	No transfer
1	Transfer-enabled period	Overflow or underflow	Overflow or underflow	No transfer	Overflow or underflow	GTCCR _m compare match
	Transfer-disabled period	No transfer	No transfer	Overflow or underflow	No transfer	GTCCR _m compare match

$m = A, B$

$x = C, E$

$y = D, F$

Figure 21.97 shows the operation of generating transfer-disabled period by updating the GTBER.BD[0] bit as an example of repeated double-buffer operations when the GTCCR buffer transfer is disabled in sawtooth-wave one-shot pulse mode.

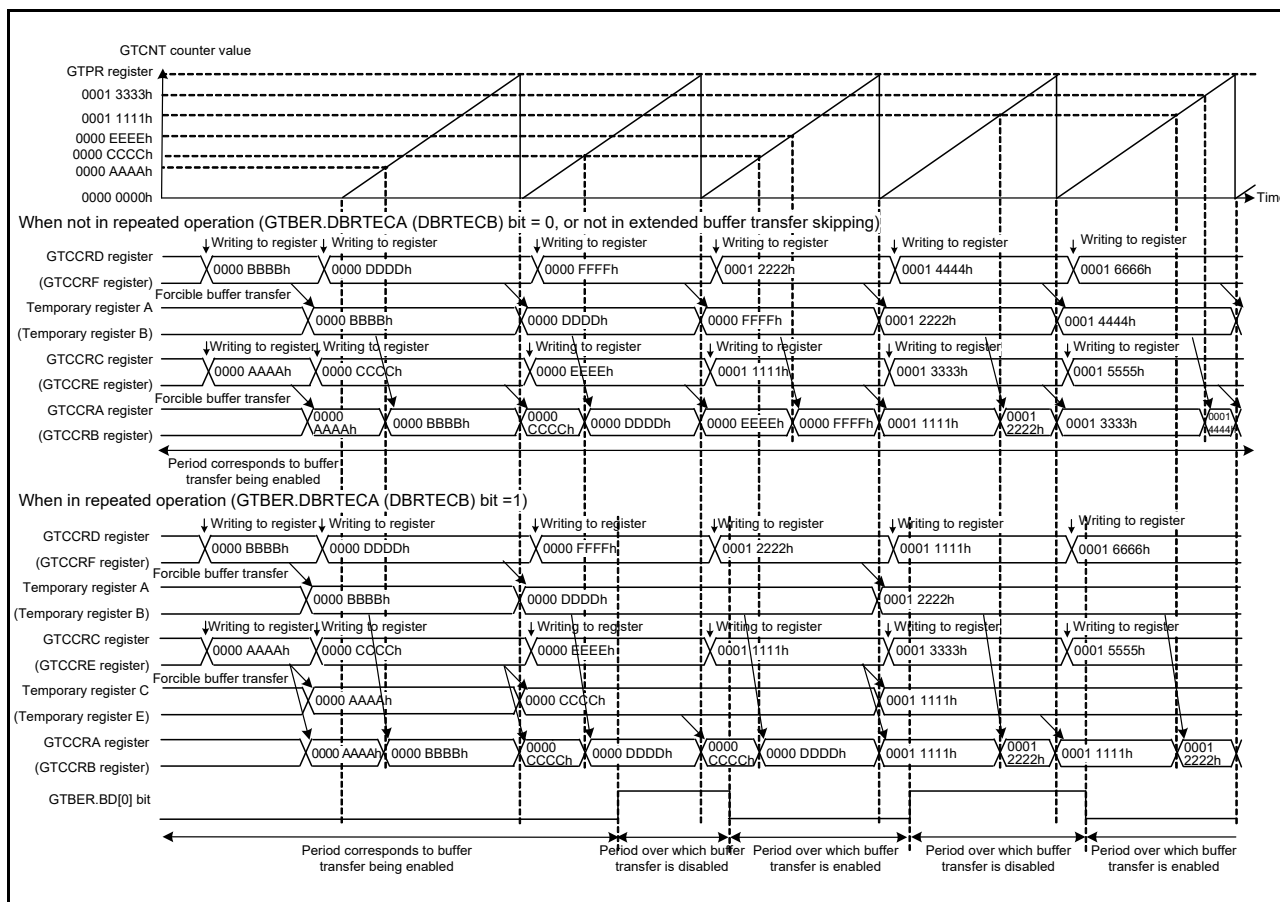


Figure 21.97 Example of Repeated Double-Buffer Operation When GTCCR Buffer Transfer is Disabled (Sawtooth-Wave One-Shot Pulse Mode, Updating the GTBER.BD[0] Bit)

(2) In triangle-wave PWM mode 3

In triangle-wave PWM mode 3, the compare match value for the first half of a waveform is stored in the temporary register x (x = C, E) as the intermediate buffer for the GTCCR_x (x = C, E) register and the compare match value for the second half of a waveform is stored in temporary register m (m = A, B) as the intermediate buffer for the GTCCR_y (y = D, F) register, respectively, for compare match values during repeated buffer operation, and the given values are alternately transferred to the GTCCR_m (m = A, B) register.

Table 21.11 lists the types of buffer transfer of the GTCCR register during counting operation in triangle-wave PWM mode 3.

While counting is stopped, the setting of the value in the temporary register is transferred through forcible buffer transfer. In forcible buffer transfer, the values of the GTCCR_y (y = D, F) registers are transferred to temporary registers m (m = A, B), and the values of the GTCCR_x (x = C, E) are transferred to temporary registers x (x = C, E), when the setting of the corresponding GTBER.DBRTEC_m (m = A, B) bit is 1, respectively.

When the setting of the GTBER.DBRTEC_m (m = A, B) bit is set to 1, values written by the CPU to the GTCCR_m (m = A, B) registers are reflected as the values of temporary registers x (x = C, E).

Table 21.11 GTCCR Buffer Operation in Triangle-Wave PWM Mode 3

GTBER.DBRTEC _m	Buffer transfer	Timing of transfer				
		GTCCR _x ↓ GTCCR _m	GTCCR _x ↓ Temporary register x	Temporary register x ↓ GTCCR _m	GTCCR _y ↓ Temporary register m	Temporary register m ↓ GTCCR _m
0	Transfer-enabled period	Trough	No transfer	No transfer	Trough	Crest
	Transfer-disabled period	No transfer	No transfer	No transfer	No transfer	No transfer
1	Transfer-enabled period	Trough	Trough	No transfer	Trough	Crest
	Transfer-disabled period	No transfer	No transfer	Trough	No transfer	Crest

m = A, B

x = C, E

y = D, F

Figure 21.98 shows the operation of generating transfer-disabled period by updating the GTBER.BD[0] bit as an example of repeated double-buffer operations when the GTCCR buffer transfer is disabled in triangle-wave PWM mode 3.

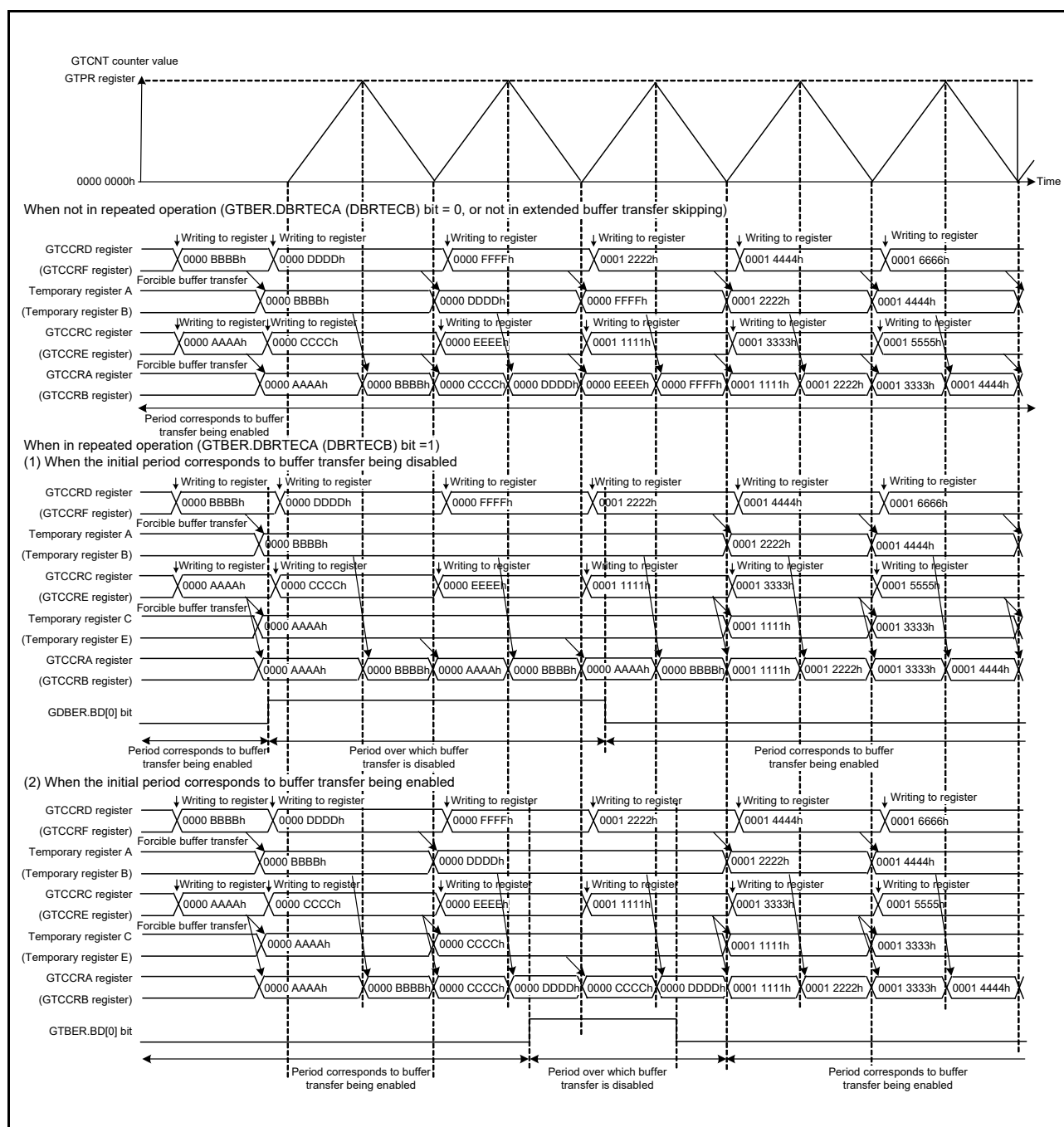


Figure 21.98 Example of Repeated Double-Buffer Operation When GTCCR Buffer Transfer is Disabled (Triangle-Wave PWM Mode 3, Updating the GTBER.BD[0] Bit)

21.7.3 GTIOCnm Pin Output Negate Control (n = 0 to 2; m = A, B)

For protection from system failure, the output negate control which changes the GTIOCnm pin output forcibly by the output stop request from the POEG is provided.

Output protection is required when a dead-time error or the same output level being on the GTIOCnA and GTIOCnB pins is detected. At that time, the detection of disabling of output is output to the group of the POEG set in the GTINTAD.GRP[1:0] bits based on the setting of the output disable detection enable bits, i.e. the GRPDTE, GRPABH, and GRPABL bits in the GTINTAD register. The POEG takes the logical OR of detection of the types described above and other forms of detection of the disabling of output, and outputs requests for the disabling of output to the GPTW. By setting the GTINTAD.GRP[1:0] bits, one output stop request can be selected out of four output stop requests which are input from the POEG as output stop request signal common in the GTIOCnA and GTIOCnB pins. Selected output stop request can be checked by reading the GTST.ODF flag.

The states of outputs at the time of control for their negation can be set with the GTIOR.OADF[1:0] bits for the GTIOCnA pin and with the GTIOR.OBDF[1:0] bits for the GTIOCnB pin.

Transition to the output negate condition by generating the output stop request from the POEG is performed asynchronously, while a release from the output negate condition by making the output stop request no longer satisfied is performed at the end of the cycle. It is after 3 PCLKB at shortest when the output negate condition is released after the output stop request becomes no longer satisfied. To reliably control output negation, clear the flag of POEG for which the condition for the request to disable the output is no longer satisfied after 4 cycles of PCLKB.

To release the output stop condition during without waiting the end of the cycle, set the OADF[1:0] bits to 00b for the GTIOCnA pin, and set the OBDF[1:0] bits to 00b for the GTIOCnB pin.

Figure 21.99 shows the output negate control operation for the GTIOCnm pin output.

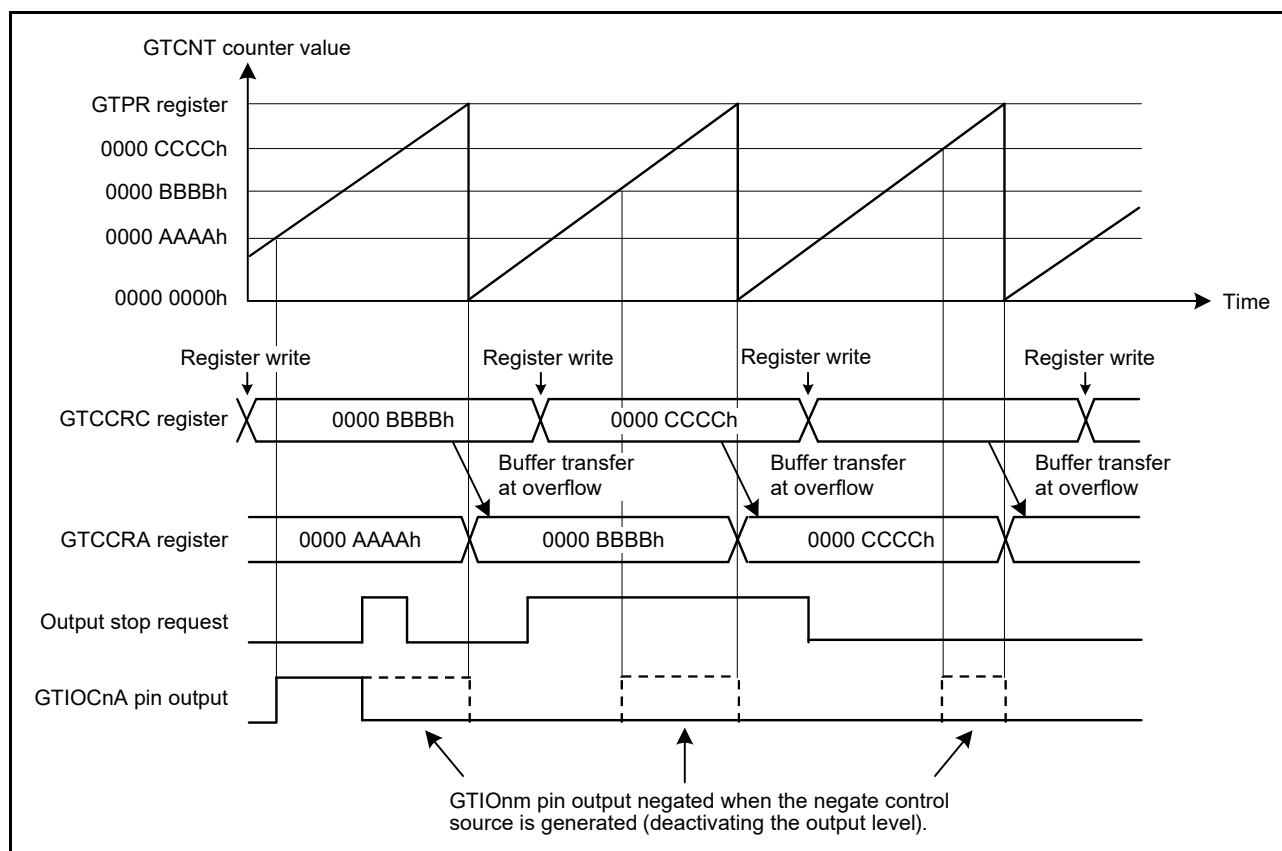


Figure 21.99 Example of Operation for GTIOCnm Pin Output Negate Control
(Sawtooth-Wave Up-Counting, Buffer Operation, Active Level: High Output at GTCCRA Register Compare Match, Low Output at the End of the Cycle) and Low Output at Output Negate)
(n = 0 to 2; m = A, B)

21.7.4 Output Protection Function for GTIOCnm Pin Output (n = 0 to 2; m = A, B)

To prepare for a case when an incorrect value (0000 0000h or a value greater than or equal to the GTPR register value) is set in the GTCCRA register, the output protection function for the GTIOCnm pin output (disabling function) is activated when the automatic dead time is set (GTDTCR.TDE bit = 1) in triangle-wave PWM mode.

The status of the output protection function can be read from the GTSOS.SOS[1:0] bits.

Figure 21.100 shows the output protection function state transition.

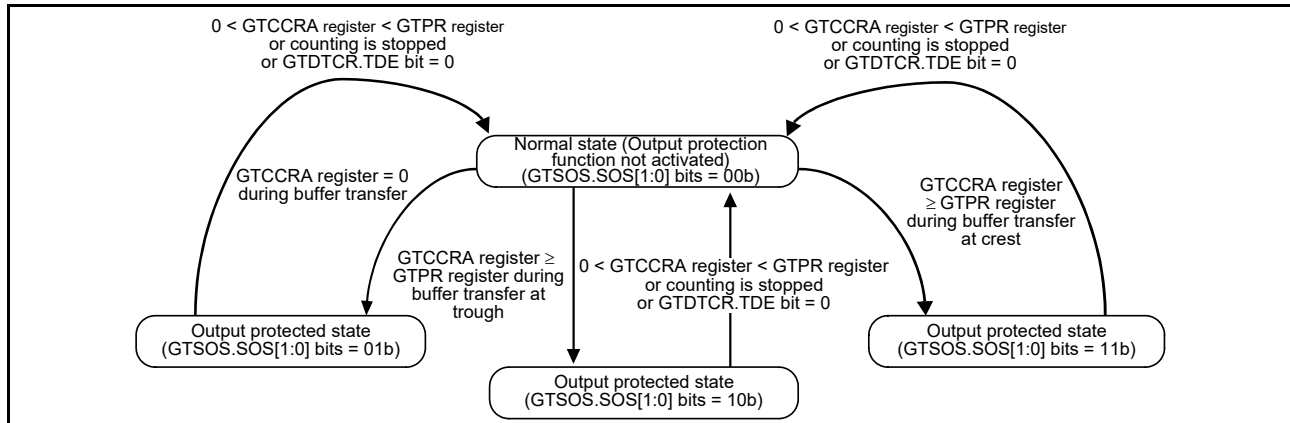


Figure 21.100 Output Protection Function

(1) Output Protection Function when the GTCCRA Register is Set to 0000 0000h during Buffer Transfer

Figure 21.101 and Figure 21.102 show examples of output protection function operation when the GTCCRA register is set to 0000 0000h during buffer transfer at troughs, and Figure 21.103 and Figure 21.104 show examples when the GTCCRA register is set to 0000 0000h during buffer transfer at crests.

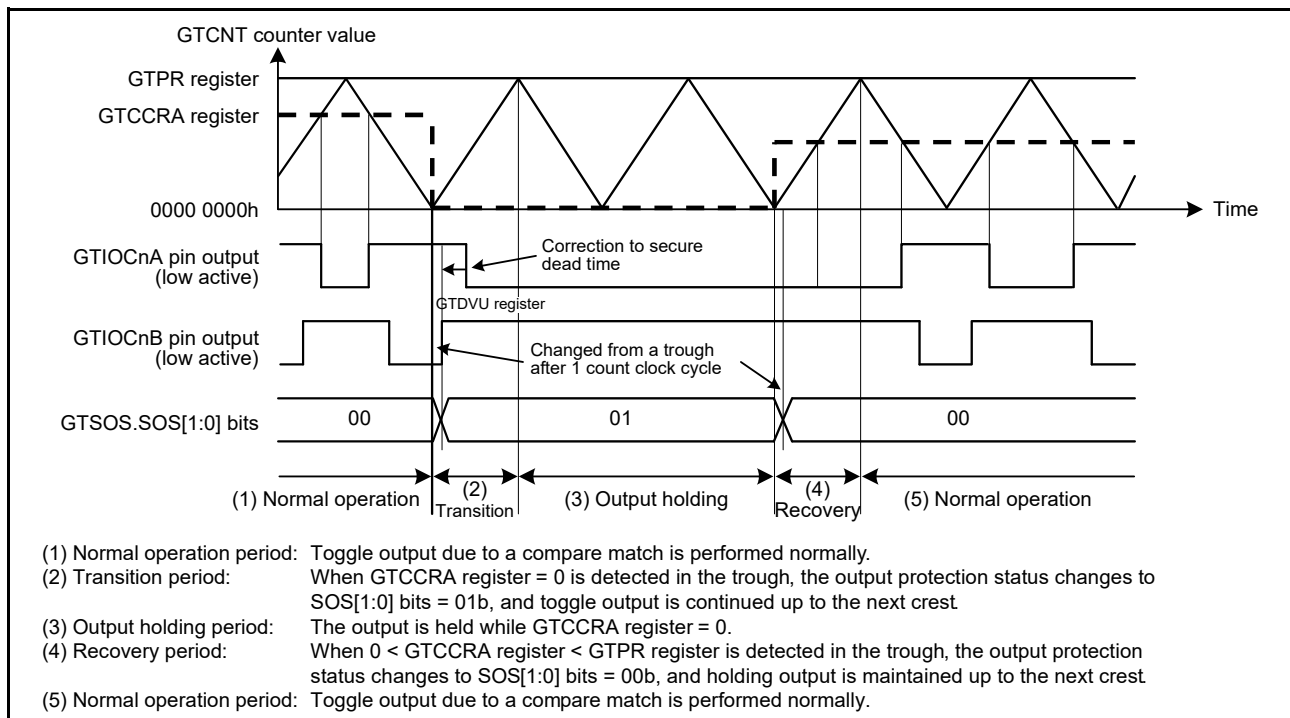


Figure 21.101 Example of Output Protection Function Operation When the GTCCRA Register is Set to 0000 0000h during Buffer Transfer at Troughs (Restored to $0 < \text{GTCCRA Register} < \text{GTPR Register}$ during Buffer Transfer at Troughs, Active Level: Low) (n = 0 to 2)

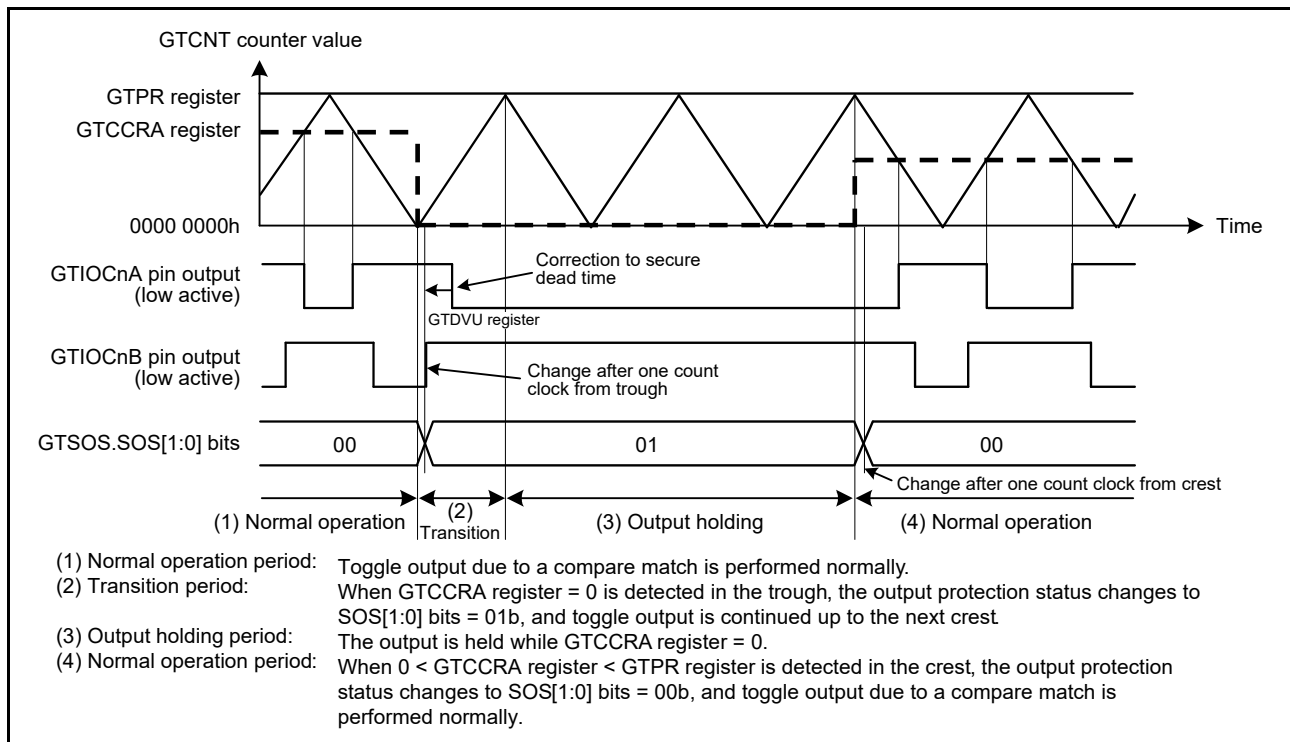


Figure 21.102 Example of Output Protection Function Operation When the GTCCRA Register is Set to 0000 0000h during Buffer Transfer at Troughs (Restored to $0 < \text{GTCCRA Register} < \text{GTPR Register}$ during Buffer Transfer at Crests, Active Level: Low) ($n = 0$ to 2)

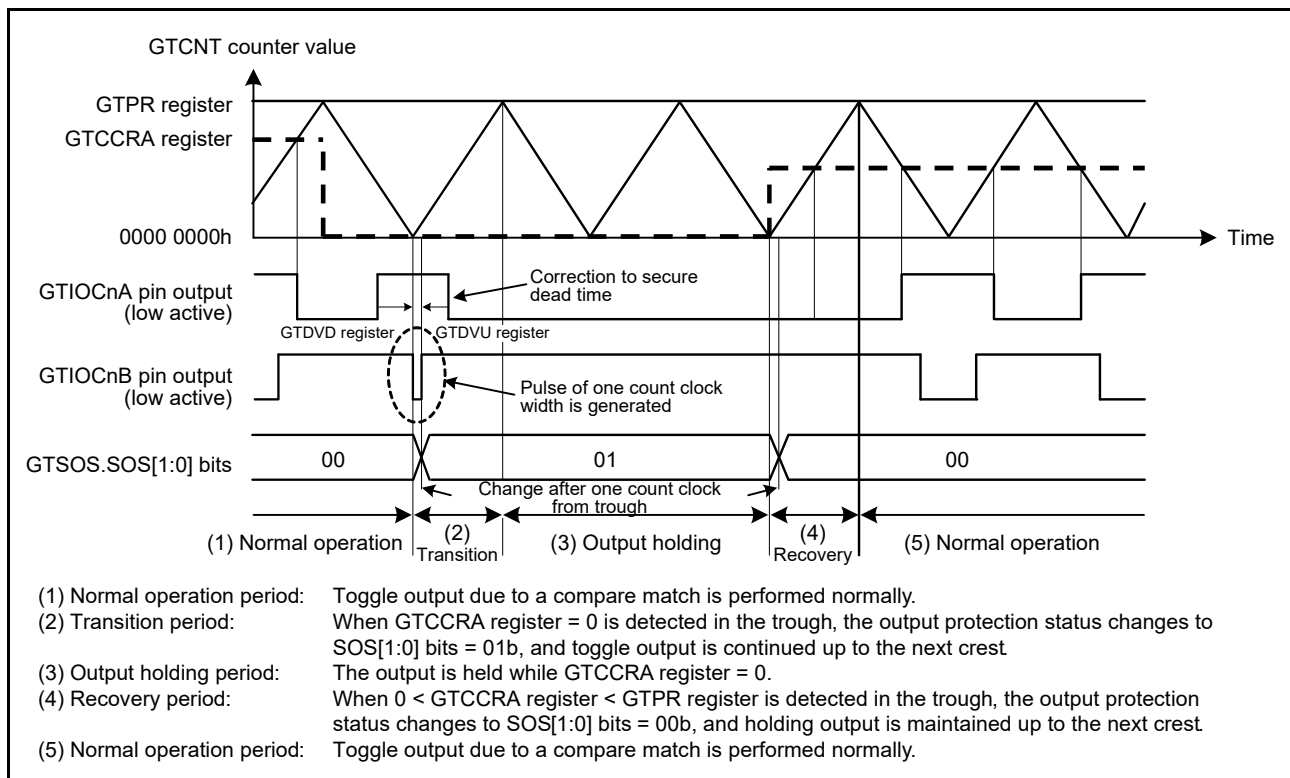


Figure 21.103 Example of Output Protection Function Operation When the GTCCRA Register is Set to 0000 0000h during Buffer Transfer at Crests (Restored to $0 < \text{GTCCRA Register} < \text{GTPR Register}$ during Buffer Transfer at Troughs, Active Level: Low) ($n = 0$ to 2)

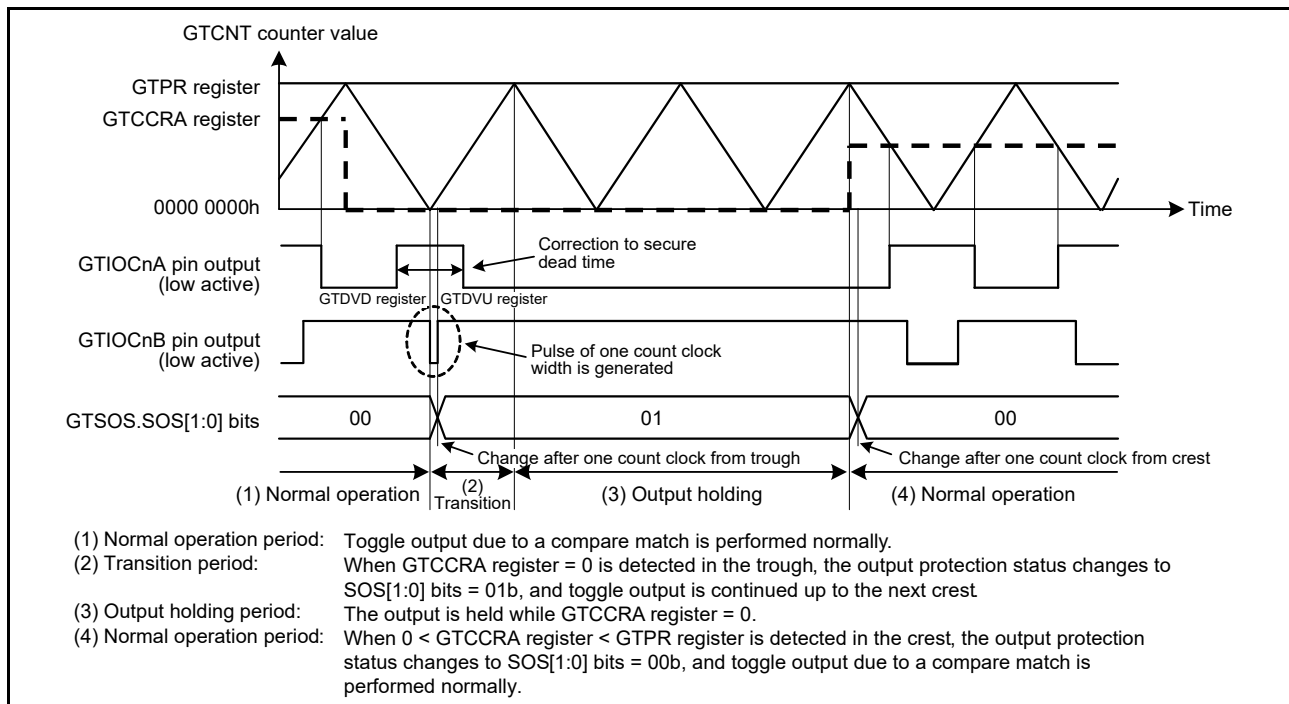


Figure 21.104 Example of Output Protection Function Operation When the GTCCRA Register is Set to 0000 0000h during Buffer Transfer at Crests (Restored to $0 < \text{GTCCRA Register} < \text{GTPR Register}$ during Buffer Transfer at Crests, Active Level: Low) ($n = 0$ to 2)

(2) Output Protection Function when GTCCRA Register $\geq$ GTPR Register is Set during Buffer Transfer at Troughs

Figure 21.105 and Figure 21.106 show examples of output protection function operation when GTCCRA register $\geq$ GTPR register is set during buffer transfer at troughs.

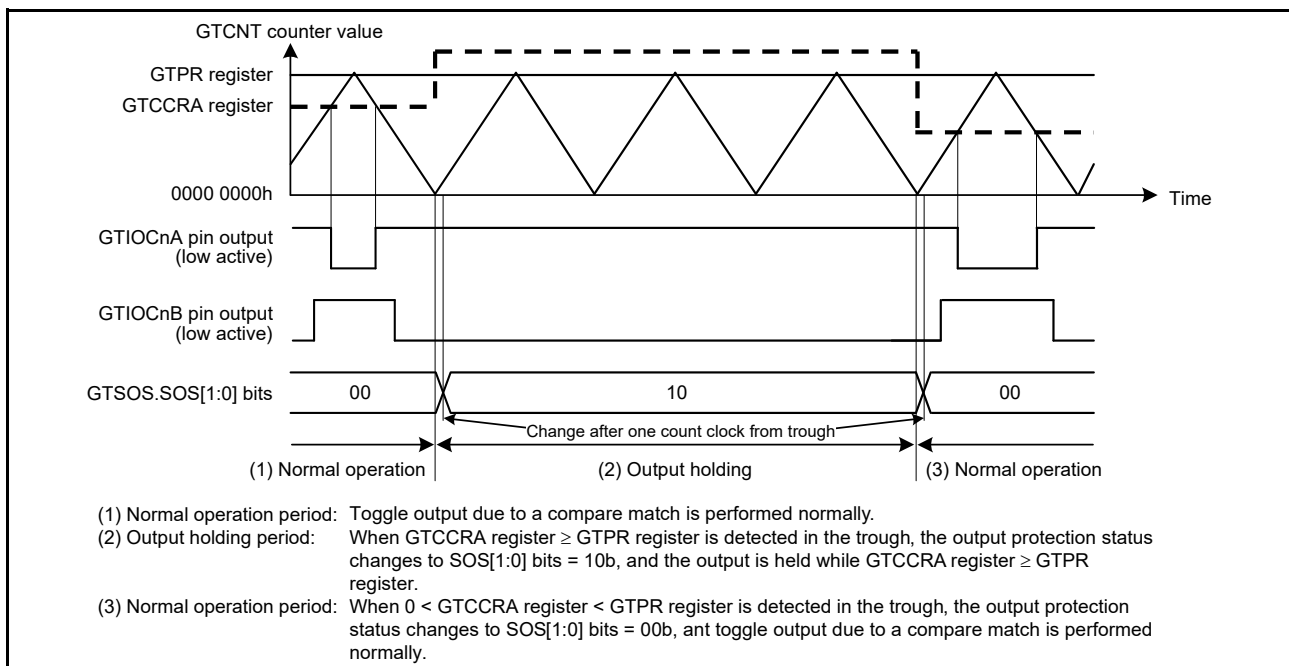


Figure 21.105 Example of Output Protection Function Operation When GTCCRA Register $\geq$ GTPR Register is set during Buffer Transfer at Troughs (Restored to $0 < \text{GTCCRA Register} < \text{GTPR Register}$ during Buffer Transfer at Troughs, Active Level: Low) ($n = 0$ to 2)

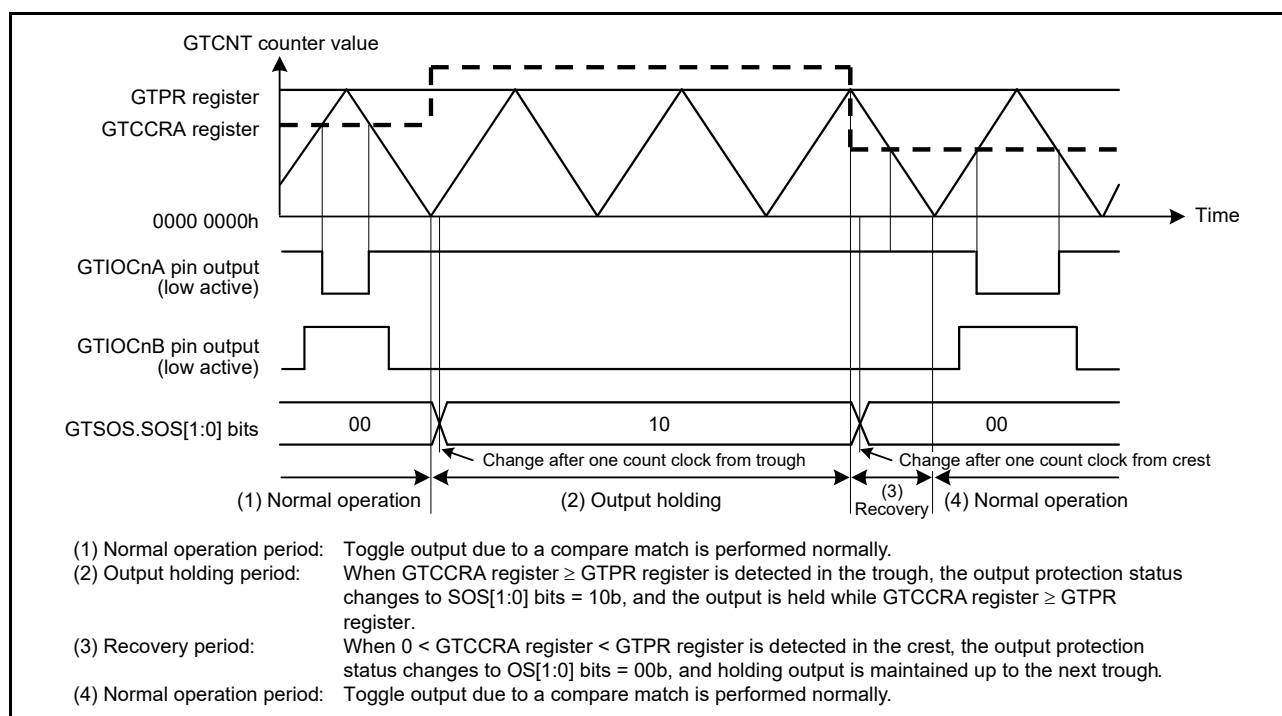


Figure 21.106 Example of Output Protection Function Operation When GTCCRA Register $\geq$ GTPR Register is Set during Buffer Transfer at Troughs (Restored to $0 < GTCCRA < GTPR$ Register during Buffer Transfer at Crests, Active Level: Low) (n = 0 to 2)

(3) Output Protection Function when GTCCRA Register $\geq$ GTPR Register is Set during Buffer Transfer at Crests

Figure 21.107 and Figure 21.108 show examples of output protection function operation when GTCCRA register $\geq$ GTPR register is set during buffer transfer at crests.

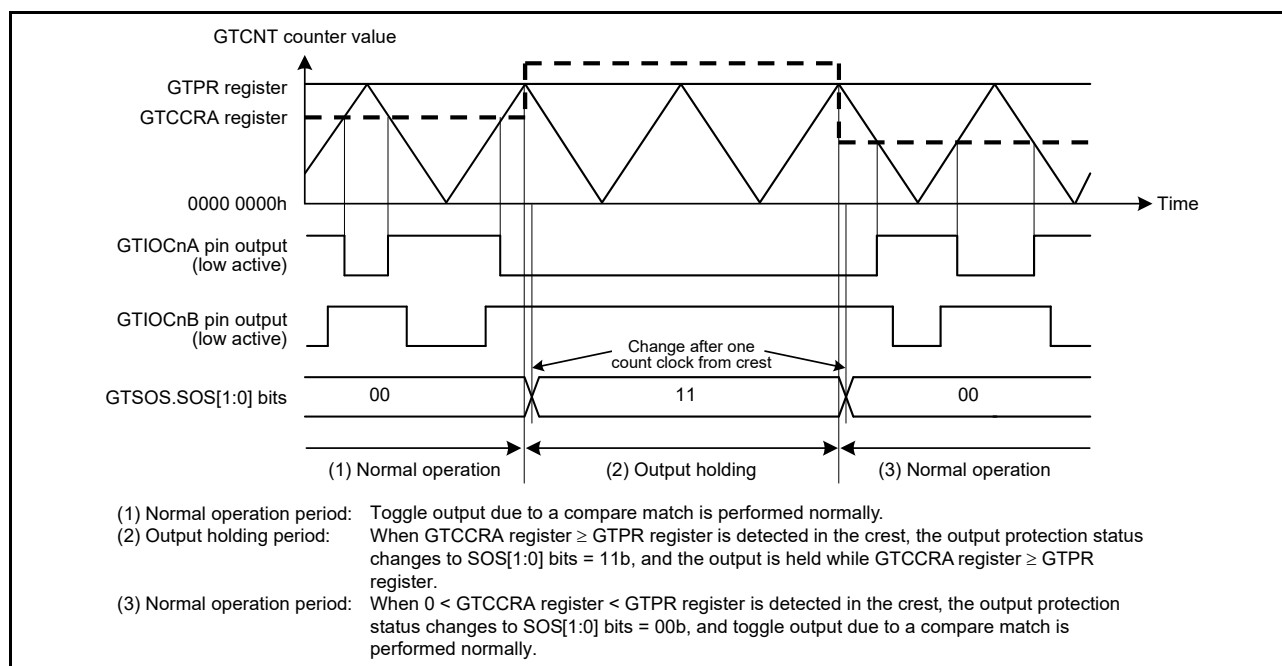


Figure 21.107 Example of Output Protection Function Operation When GTCCRA Register $\geq$ GTPR Register is Set during Buffer Transfer at Crests (Restored to $0 < GTCCRA < GTPR$ Register during Buffer Transfer at Crests, Active Level: Low) (n = 0 to 2)

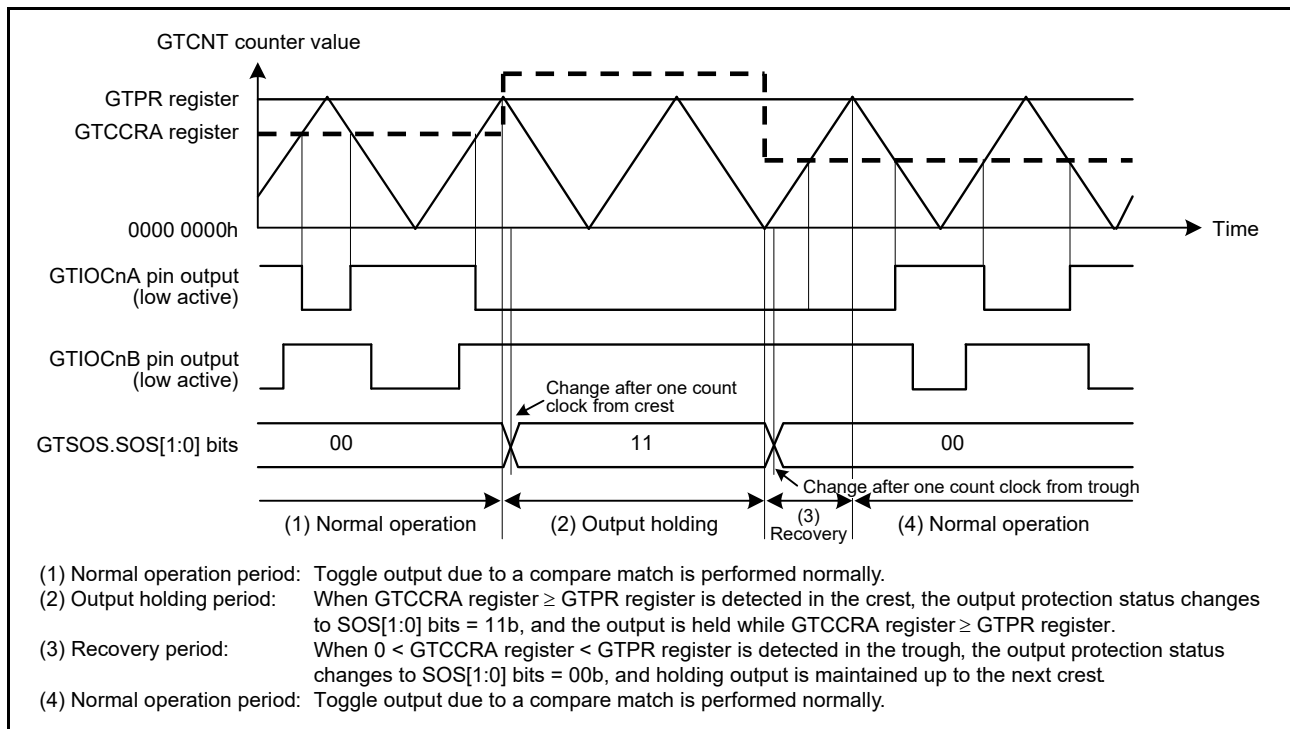


Figure 21.108 Example of Output Protection Function Operation When GTCCRA Register $\geq$ GTPR Register is Set during Buffer Transfer at Crests (Restored to $0 < \text{GTCCRA Register} < \text{GTPR Register}$ during Buffer Transfer at Troughs, Active Level: Low) (n = 0 to 2)

(4) Restricted Specification of Output Protection Function

Even if an incorrect value (0000 0000h or a value greater than or equal to the GTPR register value) is set in the GTCCRA register during count operation, the output protection functions in a specific way such that one of the positive- and negative-phase outputs becomes non-active. However, the output protection does not operate normally.

- When the GTCCRA register value at the start of count operation is greater than 0000 0000h, and less than the setting value of the GTPR register

(5) Temporary Release of Output Protection Function

When the GTSOS.SOS[1:0] bits = 10b (protected state in which GTCCRA register $\geq$ GTPR register has occurred during transfer at trough), the protected state of the GTIOcNB pin output can be temporarily released by setting the GTSOTR.SOTR bit to 1. The SOS[1:0] bits retain 10b even if the output protection function is released.

When the SOTR bit is set to 0, the GTIOcNB pin output protection can be restarted.

Figure 21.109 shows an example of the operation of temporary release of output protection when the setting of the GTCCRA register $\geq$ GTPR register during buffer transfer at troughs.

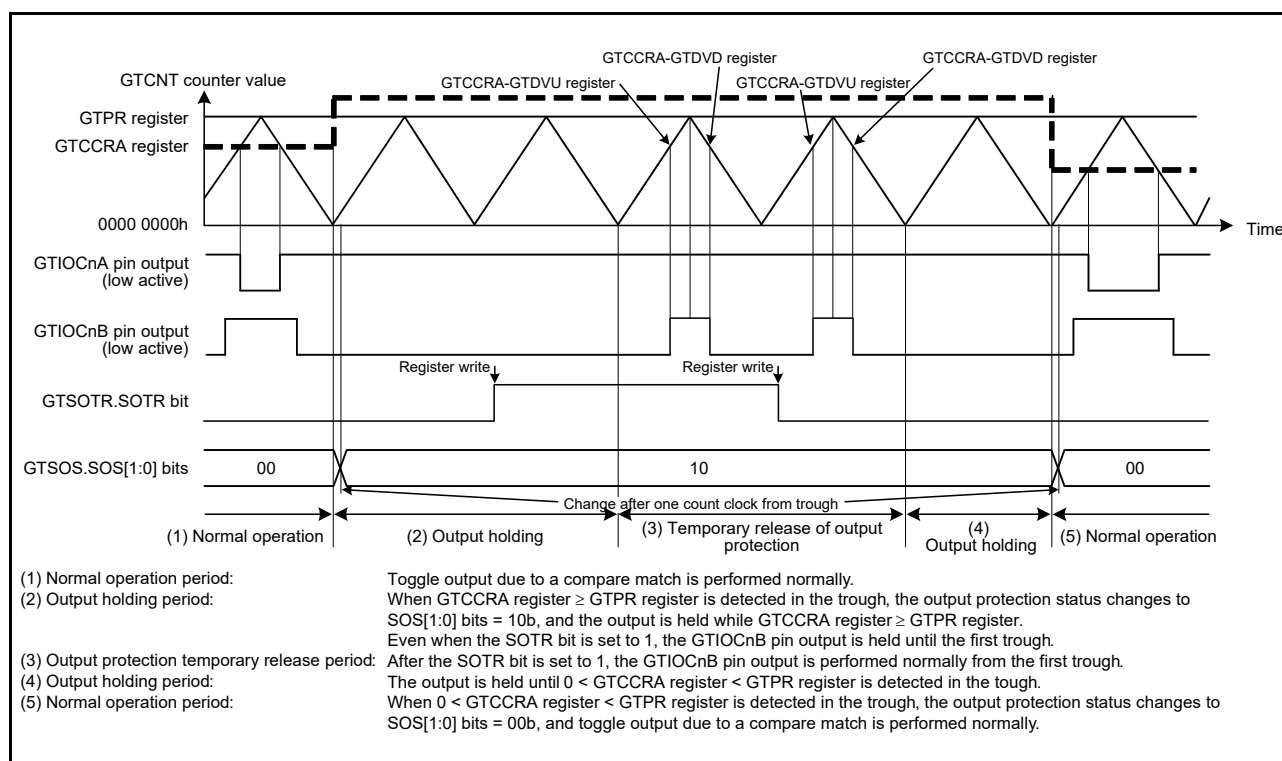


Figure 21.109 Example of the Operation of Temporary Release of Output Protection When the Setting of the GTCCRA Register $\geq$ GTPR Register during Buffer Transfer at Troughs (Restored to $0 < \text{GTCCRA Register} < \text{GTPR Register}$ during Buffer Transfer at Troughs, Active Level: Low) ($n = 0$ to 2)

21.8 Initialization Method of Output Pins

21.8.1 Pin Settings after Reset

The GPTW registers are initialized at a reset. Initialize the GPTW for output to the external pins by making the port mode settings and setting the OAE and OBE bits in the GTIOR register, and then start the counter counting.

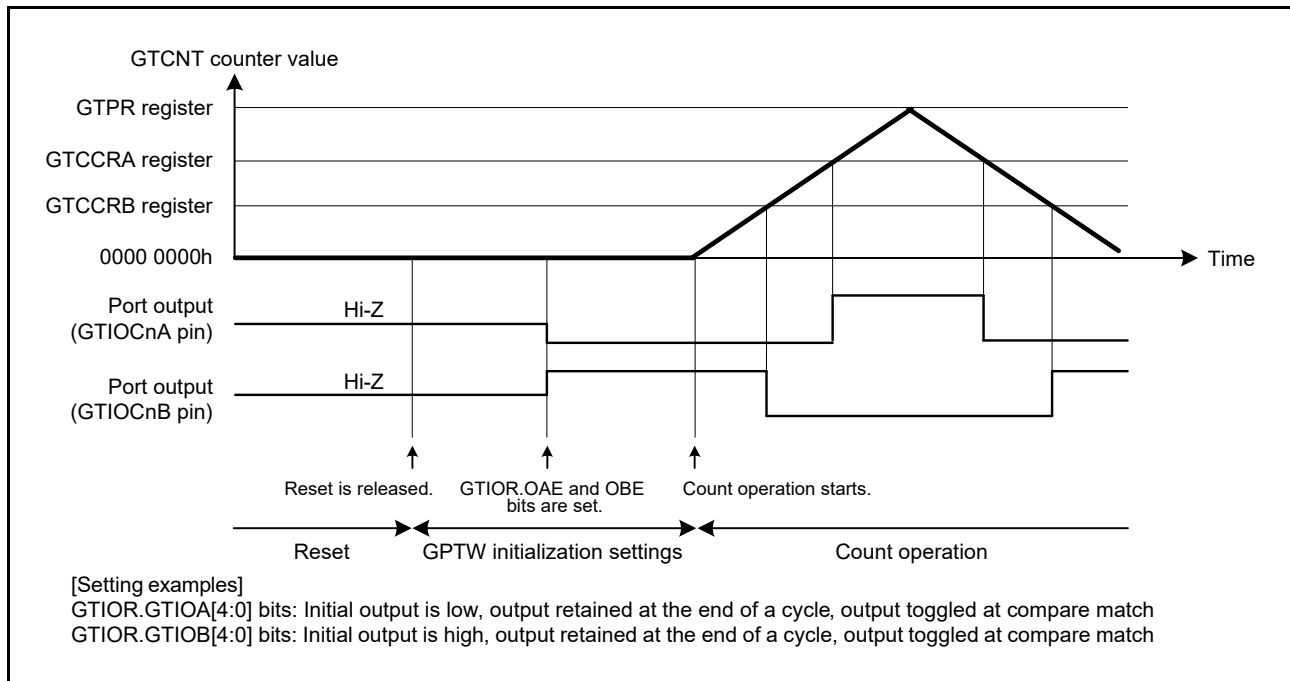


Figure 21.110 Example of Pin Settings after Reset (n = 0 to 2)

21.8.2 Pin Initialization Due to Error during Operation

If an error occurs during GPTW operation, the following four types of pin processing can be performed before pin initialization.

- (1) Set OAHLD and OBHLD bits in the GTIOR register to 1 and retain the outputs at count stop.
- (2) Set OAHLD and OBHLD bits to 0, specify arbitrary output values at OADFLT and OBDFLT bits in the GTIOR register, and output the arbitrary values at count stop.
- (3) Set the pin to output an arbitrary value as a general output port by setting the PDR, PODR, and PMR registers of the I/O port in advance. Set the OAE and OBE bits in the GTIOR register to 0 and the control bit in PMR register that corresponds to the pin to 0 to allow the arbitrary values to be output from the pin set as a general output port when an error occurs.
- (4) Drive the output to a high impedance state using the POEG function.

When automatic dead time setting has been made, set the GTDTCR.TDE bit to 0 once after counting is stopped.

When counting is stopped, only the values of registers that are changed by a GPTW external source will change. If counting is resumed, operation will carry on from where it was stopped.

If counting was stopped, registers should be initialized before counting is started.

21.9 Usage Notes

21.9.1 Module Stop Function Setting

Operation of the GPTW can be disabled or enabled by the module stop control register. The initial setting is for operation of the GPTW to be halted. Register access is enabled by clearing module stop state. For details, see section 11, Low Power Consumption.

21.9.2 Settings of the GTCCRm Register during Compare Match Operation (m = A to F)

(1) When Automatic Dead Time Setting has been Made in Triangle-Wave PWM Mode

The GTCCRA register should satisfy the following conditions:

- GTCCRA register > GTDVU register,
- GTCCRA register > GTDVD register, and
- GTCCRA register < GTPR register.

When the GTCCRA register is set to 0000 0000h or a value greater than or equal to the GTPR register value during count operation, the output protection function is activated.

However, if the following conditions are not satisfied, the output protection does not operate normally.

- When the GTCCRA register value at the start of count operation is 0000 0001h or greater, and less than the setting value of the GTPR register

For details, refer to section 21.7.4, Output Protection Function for GTIOcnm Pin Output (n = 0 to 2; m = A, B).

(2) When Automatic Dead Time Setting has not been Made in Triangle-Wave PWM Mode

Set a value greater than 0000 0000h, and less than the setting value of the GTPR register in the GTCCRA register. When 0000 0000h or the same value as that of the GTPR register is set in the GTCCRA register, compare match is generated in one cycle only when [GTCCRA register = 0000 0000h] or [GTCCRA register = GTPR register] is met. Furthermore, a value exceeding the setting value of the GTPR register is set in the GTCCRA register, compare match does not occur. Similarly, set a value greater than 0000 0000h, and less than the setting value of the GTPR register in the GTCCRB register. When 0000 0000h or the same value as that of the GTPR register is set in the GTCCRB register, compare match is generated in one cycle only when [GTCCRB register = 0000 0000h] or [GTCCRB register = GTPR register] is met. Furthermore, a value exceeding the setting value of the GTPR register is set in the GTCCRB register, compare match does not occur.

(3) When Automatic Dead Time Setting has been Made in Sawtooth-Wave One-Shot Pulse Mode

The GTCCRC and GTCCRD registers should be set to satisfy the following restrictions. If the restrictions are not satisfied, correct output waveforms with secured dead time may not be obtained.

- In up-counting:
 - GTCCRC register < GTCCRD register
 - GTCCRC register > GTDVU register
 - GTCCRD register < GTPR register – GTDVD register
- In down-counting:
 - GTCCRC register > GTCCRD register
 - GTCCRC register < GTPR register – GTDVU register
 - GTCCRD register > GTDVD register

(4) When Automatic Dead Time Setting has not been Made in Sawtooth-Wave One-Shot Pulse Mode

The GTCCRC and GTCCRD registers should be set to satisfy the following restrictions. If the restrictions are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: $0 < \text{GTCCRC register} < \text{GTCCRD register} < \text{GTPR register}$
- In down-counting: $\text{GTPR register} > \text{GTCCRC register} > \text{GTCCRD register} > 0$

Similarly, GTCCRE and GTCCRF registers should be set to satisfy the following restrictions. If the restrictions are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: $0 < \text{GTCCRE register} < \text{GTCCRF register} < \text{GTPR register}$
- In down-counting: $\text{GTPR register} > \text{GTCCRE register} > \text{GTCCRF register} > 0$

(5) In Sawtooth-Wave PWM Mode

The GTCCRA register should be set with the range of $0000\ 0000h < \text{GTCCRA register} < \text{GTPR register}$. If GTCCRA register = $0000\ 0000h$ or GTCCRA register = GTPR register is set, a compare match occurs within the cycle only when GTCCRA register = $0000\ 0000h$ or GTCCRA register = GTPR register is satisfied. If GTCCRA register > GTPR register is set, no compare match occurs.

Similarly, GTCCRB register should be set with the range of $0000\ 0000h < \text{GTCCRB register} < \text{GTPR register}$. If GTCCRB register = $0000\ 0000h$ or GTCCRB register = GTPR register is set, a compare match occurs within the cycle only when GTCCRB register = $0000\ 0000h$ or GTCCRB register = GTPR register is satisfied. If GTCCRB register > GTPR register is set, no compare match occurs.

21.9.3 Setting Range of the GTCNT Counter

Set the range of the GTCNT counter within the range of $0 \leq \text{GTCNT counter} \leq \text{GTPR register}$.

21.9.4 The GTCNT Counter Start/Stop

The GTCNT counter start/stop control by the GTCR.CST bit is synchronized with the count clock selected by the GTCR.TPCS[3:0] bits. Since the GTCNT counter starts or stops after one count clock cycle selected by the TPCS[3:0] bits following the CST bit updating, events until the GTCNT counter actually starts are ignored, where events may be accepted and the interrupt may be generated after the CST bit becomes 0.

21.9.5 Order of Priority in Events

(1) GTCNT Counter

Order of priority in events to update the GTCNT counter is shown below.

Table 21.12 Order of Priority in Updating GTCNT Counter

The GTCNT counter updating source	Order of priority
CPU writing (GTCNT counter writing/GTCLR register writing)	High ↑ Low
Clearing by a hardware source set by the GTCSR register	
Count operation	

When updating of the GTCNT counter conflicts with reading by the CPU, the value before updating is reflected.

(2) The GTCR.CST Bit

In case that start/stop by a hardware source set by the GTSSR and GTPSR registers conflicts with the CPU writing (GTCR register writing/GTSTR register writing /GTSTP register writing), the CPU writing takes priority.

In case of contention between start by a hardware source set by the GTSSR register and stop by a hardware source set by the GTPSR register, the state of the CST bit does not change.

In case of contention between the CST bit updating and the CPU reading (GTCR register reading/GTSTR register reading /GTSTP register reading), data before updating is read.

(3) GTCCRm Register (m = A to F)

In case of contention between the GTCCRm register writing and an input capture/buffer transfer, an input capture/buffer transfer takes priority over the GTCCRm register writing.

In case that an input capture conflicts with the CPU writing or the counter updating by a hardware source, the counter value before updating is captured.

In case of contention between the GTCCRm register updating and the CPU reading, data before updating is read.

(4) GTPR Register

In case of contention between a buffer transfer and the GTPR register writing, the GTPR register writing takes priority over a buffer transfer.

In case of contention between the GTPR register updating and the CPU reading, data before updating is read.

(5) GTADTRm Register (m = A, B)

In case of contention between a buffer transfer and the GTADTRm register writing, the GTADTRm register writing takes priority over a buffer transfer.

In case of contention between the GTADTRm register updating and the CPU reading, data before updating is read.

(6) GTDVM Register (m = U, D)

In case of contention between a buffer transfer and the GTDVM register writing, the GTDVM register writing takes priority over a buffer transfer.

In case of contention between the GTDVM register updating and the CPU reading, data before updating is read.

(7) GTIOR.GTIOm bit (m = A, B)

When there is a conflict between buffer transfer operation and writing to GTIOR.GTIOm bit, writing to GTIOR.GTIOm bit has priority over buffer transfer operation.

In case of contention between the GTIOR.GTIOm bit updating and the CPU reading, data before updating is read.

22. GPTW Port Output Enable (POEGb)

22.1 Overview

The POEG issues requests to disable output from output pins of the general PWM timer (GPTW). The combination of output pins of the POEG to be disabled can be specified from any channel of the GPTW, which is a different form of operation from that of the POE3. Use the POE3 to disable output from pins of the MTU3 and GPTW at the same time. Select the method of detection for disabling the output from the list below.

- Detection of input level or edge on the GTETRGA pin
- Output disabling condition detection by the GPTW
- Detection by comparator (edge detection or level detection)
- Detection of stopping of oscillation by the oscillation stop detection circuit for the main clock
- Register setting

The GTETRGA pins can also be used as the external trigger input pins for the GPTW.

Table 22.1 shows specifications, Figure 22.1 shows a system block diagram around the POEG, Figure 22.2 shows a block diagram of the POEG, and Table 22.2 shows input pins.

Table 22.1 POEG Specifications

Item	Description
Output disabling request by detection of input level or edge on the GTETRGA pin	• The output disabling request is issued to the GPTW when a POEGGA.PIDF flag is set to 1 by the detection of either input of the selected level or the selected edge on the GTETRGA pin. • The output disabling request is issued to the GPTW directly from the input signal on the GTETRGA pin.
Output disabling request by output disabling condition detection signal from the GPTW	• The output disabling request is issued to the GPTW when the GPTW detects a simultaneous active level (high or low) on the GTIOCA and GTIOCB pins and the POEGGA.IOCF flag is set to 1. • The output disabling request is issued to the GPTW when the GPTW detects a dead-time error and the POEGGA.IOCF flag is set to 1.
Output disabling request by detection signal from the comparator	• The output disabling request is issued to the GPTW when the POEGGA.IOCF flag is set to 1 by the COMPn edge detection signals. • The output disabling request is issued to the GPTW directly from the COMPn level detection signals. • The PWM signal output from the GPTW can mask the output disabling request.
Output disabling request in response to detecting the stopping of oscillation	The output disabling request is issued to the GPTW when the oscillation stop detection circuit for the main clock detects the oscillation stop and the POEGGA.OSTPF flag is set to 1.
Output disabling request by software	The output disabling request is issued to the GPTW when the software sets the POEGGA.SSF bit to 1.
Interrupt	• An interrupt is generated by an output disabling request via the POEGGA.PIDF flag. • An interrupt is generated by an output disabling request via the POEGGA.IOCF flag.
External trigger output to the GPTW	The input signal from the GTETRGA pin is output to the GPTW as an external trigger.
Noise removal	Each GTETRGA pin has a digital noise filter. • Eight types of sampling clock are selectable for the filter. • Match detection conditions can be selected from three to six consecutive samples with the same level.

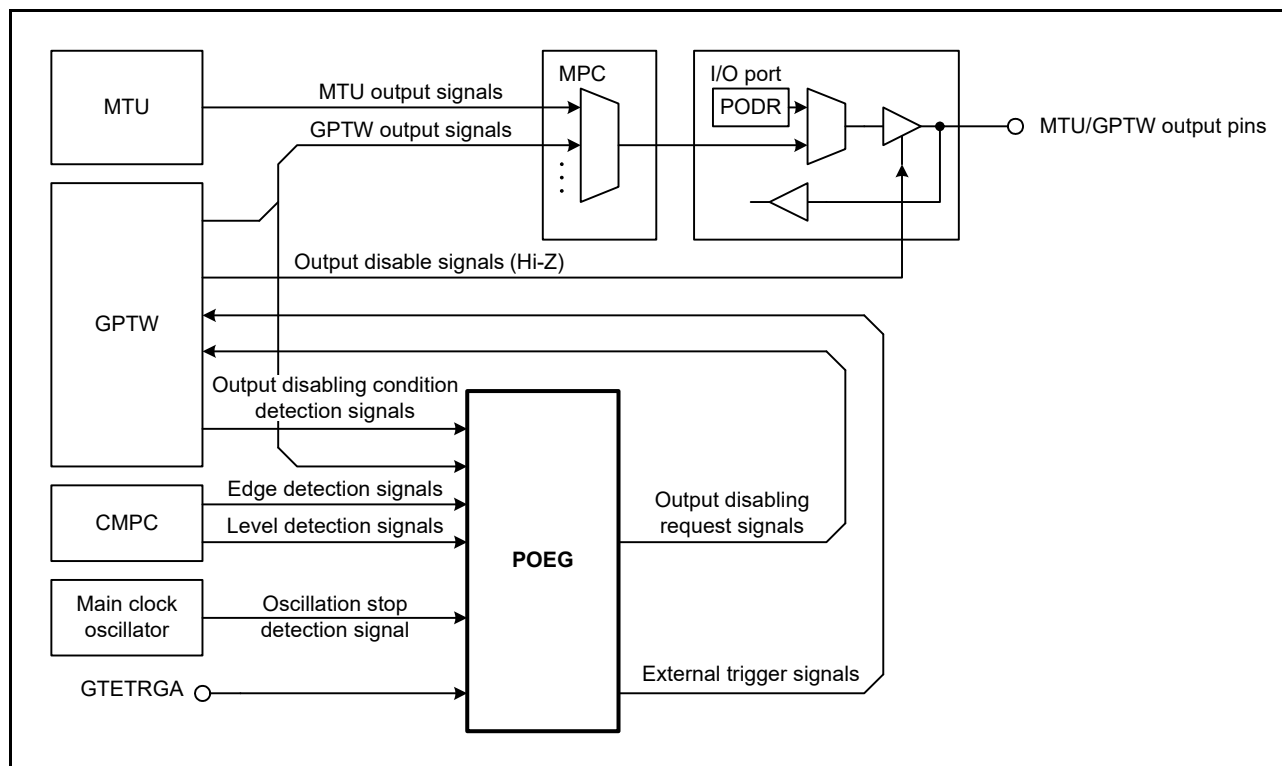


Figure 22.1 POEG System Block Diagram

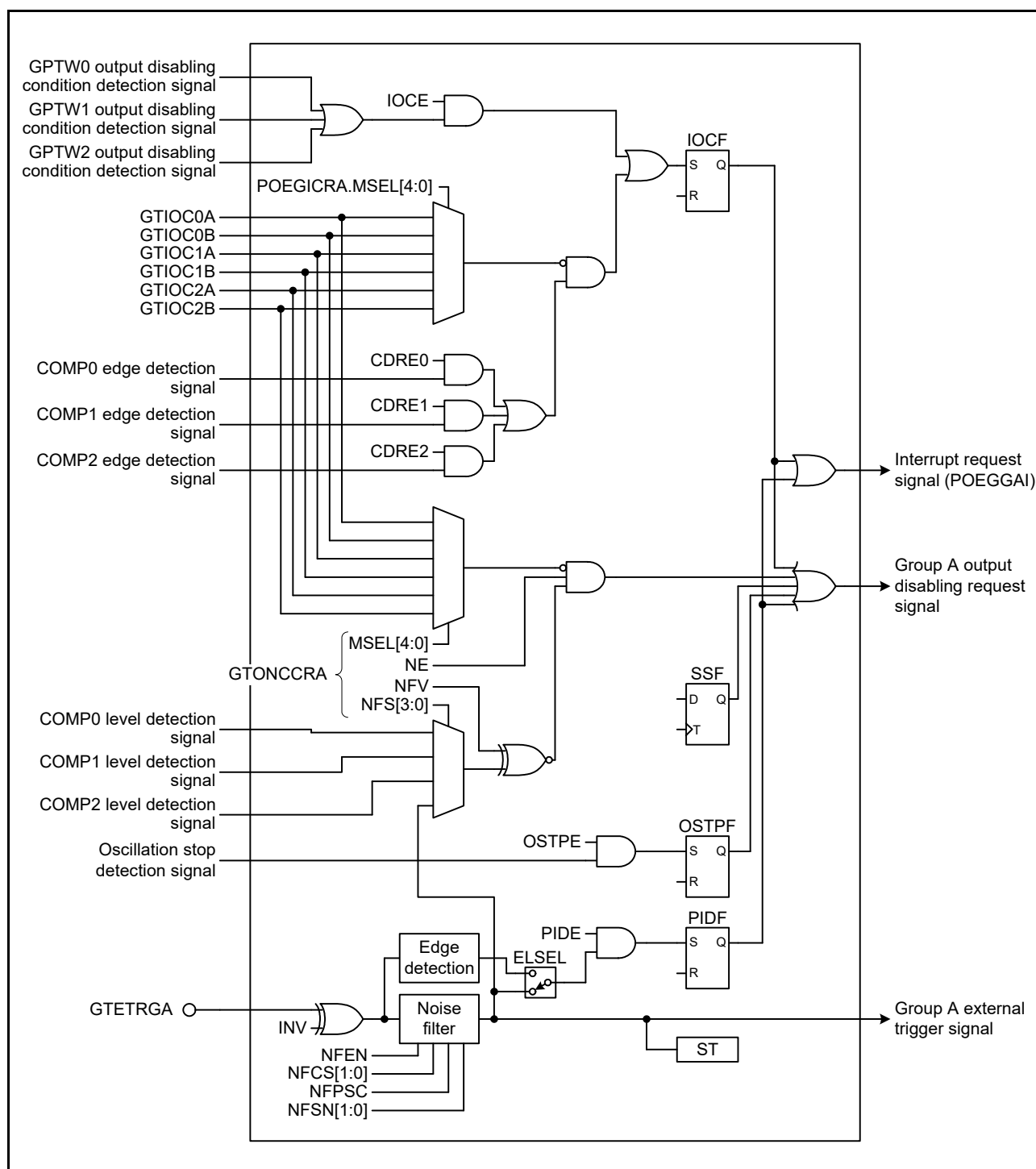


Figure 22.2 POEG Block Diagram

Table 22.2 POEG I/O Pins

Pin Name	I/O	Function
GTETRGA	Input	Input pin A for output disable request signal for GPTW output pin and GPTW external trigger

22.2 Register Descriptions

22.2.1 POEG Group A Setting Register (POEGGA)

Address(es): POEG.POEGGA 0009 E000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NFCS[1:0]		NFEN	INV	NFSN[1:0]		ELSEL	NFPSC	—	—	—	—	—	—	—	ST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CDRE2	CDRE1	CDRE0	—	OSTPE	IOCE	PIDE	SSF	OSTPF	IOCF	PIDF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PIDF	Port Input Detection Flag	0: Indicates that an output disabling request by the GTETRGA signal has not been issued. 1: Indicates that an output disabling request by the GTETRGA signal has been issued.	R/(W) *1
b1	IOCF	GPTW/CMPC Output Disabling Condition Detection Flag	0: Indicates that an output disabling request by the GPTW output disabling condition detection signal or the COMPn edge detection signal has not been issued. 1: Indicates that an output disabling request by the GPTW output disabling condition detection signal or the COMPn edge detection signal has been issued.	R/(W) *1
b2	OSTPF	Oscillation Stop Detection Flag	0: Indicates that an output disabling request by oscillation stop has not been issued. 1: Indicates that an output disabling request by oscillation stop has been issued.	R/(W) *1
b3	SSF	Software Output Disabling Request	0: Does not issue output disabling request by software. 1: Issue an output disabling request by software.	R/W
b4	PIDE	Port Input Signal Enable	0: Output disabling request by the GTETRGA signal is disabled. 1: Output disabling request by the GTETRGA signal is enabled.	R/W*2
b5	IOCE	GPTW Output Disabling Condition Detection Signal Enable	0: Output disabling request by the GPTW output disabling condition detection signal is disabled. 1: Output disabling request by the GPTW output disabling condition detection signal is enabled.	R/W*2
b6	OSTPE	Oscillation Stop Detection Signal Enable	0: Output disabling request by the oscillation stop detection signal is disabled. 1: Output disabling request by the oscillation stop detection signal is enabled.	R/W*2
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b8	CDRE0	COMP0 Edge Detection Signal Enable	0: Output disabling request by the COMP0 edge detection signal is disabled. 1: Output disabling request by the COMP0 edge detection signal is enabled.	R/W*2
b9	CDRE1	COMP1 Edge Detection Signal Enable	0: Output disabling request by the COMP1 edge detection signal is disabled. 1: Output disabling request by the COMP1 edge detection signal is enabled.	R/W*2
b10	CDRE2	COMP2 Edge Detection Signal Enable	0: Output disabling request by the COMP2 edge detection signal is disabled. 1: Output disabling request by the COMP2 edge detection signal is enabled.	R/W*2
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b16	ST	GTETRGA Signal Status Flag	0: The GTETRGA signal output to the GPTW is low. 1: The GTETRGA signal output to the GPTW is high.	R
b23 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b24	NFPSC	Noise Filter Clock Select	These bits are used in combination with the NFCS[1:0] bits to set the sampling clock for the noise filter.	R/W
b25	ELSEL	GTETRGA Input Edge or Level Detection Select	0: Level detection 1: Edge detection	R/W
b27, b26	NFSN[1:0]	Noise Filter Subsequent Match Count Select	b27 b26 0 0: 3 times 0 1: 4 times 1 0: 5 times 1 1: 6 times	R/W
b28	INV	GTETRGA Input Inverting	0: Does not invert the input signal from the GTETRGA pin. 1: Invert the input signal from the GTETRGA pin.	R/W
b29	NFEN	Noise Filter Enable	0: Digital noise filter on the GTETRGA pin is disabled. 1: Digital noise filter on the GTETRGA pin is enabled.	R/W
b31, b30	NFCS[1:0]	Noise Filter Clock Select	- When the setting of the NFPSC bit is 0 b31 b30 0 0: The input signal from the GTETRGA pin is sampled every PCLKB/1 clock. 0 1: The input signal from the GTETRGA pin is sampled every PCLKB/8 clock. 1 0: The input signal from the GTETRGA pin is sampled every PCLKB/32 clock. 1 1: The input signal from the GTETRGA pin is sampled every PCLKB/128 clock. - When the setting of the NFPSC bit is 1 b31 b30 0 0: The input signal from the GTETRGA pin is sampled every PCLKB/2 clock. 0 1: The input signal from the GTETRGA pin is sampled every PCLKB/4 clock. 1 0: The input signal from the GTETRGA pin is sampled every PCLKB/16 clock. 1 1: The input signal from the GTETRGA pin is sampled every PCLKB/64 clock.	R/W

Note 1. Only 0 can be written to clear the flag.

Note 2. This bit can be written only once after reset.

The POEGGA register controls output disabling requests and an external trigger for the GPTW in response to the various detection signals.

SSF Bit (Software Output Disabling Request)

Writing 1 to this bit leads to an output disabling request being issued to the GPTW, and writing 0 to the bit releases the GPTW from the output disabling request.

22.2.2 POEG Group A Input Control Register (POEGICRA)

Address(es): POEG.POEGICRA 0009 E004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

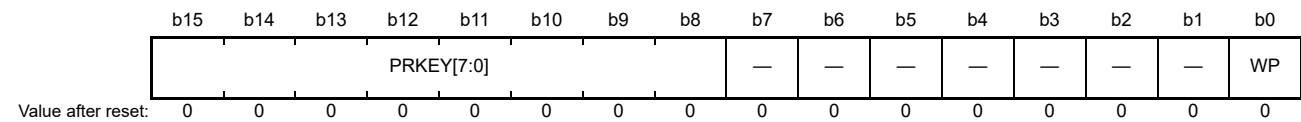
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	MSEL[4:0]				—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	MSEL[4:0]	Output Disabling Request Mask Signal Select	When the selected signal is high, the output disabling requests by the COMPn edge detection signals enabled by the POEGGA.CDREn bits are masked (n = 0 to 2). b4 b0 0 0 0 0: Output disabling requests are not masked. 0 0 0 1: The GTIOC0A signal is used to mask output disabling requests. 0 0 1 0: The GTIOC0B signal is used to mask output disabling requests. 0 0 1 1: The GTIOC1A signal is used to mask output disabling requests. 0 1 0 0: The GTIOC1B signal is used to mask output disabling requests. 0 1 0 1: The GTIOC2A signal is used to mask output disabling requests. 0 1 1 0: The GTIOC2B signal is used to mask output disabling requests. Settings other than above are prohibited.	R/W *1
b31 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. These bits can be written only once after reset.

22.2.3 GPTW Output Disabling Control Group A Write Protection Register (GTONCWPA)

Address(es): POEG.GTONCWPA 0009 E040h



Bit	Symbol	Bit Name	Description	R/W
b0	WP	Register Writing Disable	0: Writing to the GTONCCRA register is enabled. 1: Writing to the GTONCCRA register is disabled.	R/W*1
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	PRKEY[7:0]	Key Code	These bits control whether or not the WP bit can be rewritten.	R/W*2

Note 1. Writing to this bit is valid only when this register is written in 16-bit units with the PRKEY[7:0] bits set to A5h.
Note 2. Written values are not retained by these bits. These bits are read as 00h.

The GTONCWPA register enables or disables writing to the GTONCCRA register in order to avoid incorrect writing to the register.

22.2.4 GPTW Output Disabling Control Group A Controlling Register (GTONCCRA) (A = A to D)

Address(es): POEG.GTONCCRA 0009 E044h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	MSEL[4:0]					—	—	NFV	NFS[3:0]			—	—	—	NE	
Value after reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NE	Direct Output Disabling Request Enable	0: Direct output disabling request is disabled. 1: Direct output disabling request is enabled.	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7 to b4	NFS[3:0]	Direct Output Disabling Request Source Select	b7 b4 0 0 0 0: COMP0 level detection signal 0 0 0 1: COMP1 level detection signal 0 0 1 0: COMP2 level detection signal 0 1 1 1: GTETRGA pin input signal (high or low) 1 0 0 0: GTETRGA pin input signal, COMP0 level detection signal, COMP1 level detection signal, or COMP2 level detection signal 1 0 0 1: GTETRGA pin input signal, COMP0 level detection signal, or COMP1 level detection signal 1 0 1 0: GTETRGA pin input signal, COMP1 level detection signal, or COMP2 level detection signal 1 0 1 1: GTETRGA pin input signal or COMP0 level detection signal 1 1 0 0: GTETRGA pin input signal or COMP1 level detection signal Other settings are prohibited.	R/W
b8	NFV	Direct Output Disabling Request Polarity	0: Requests output disabling when the detection signal is 0. 1: Requests output disabling when the detection signal is 1.	R/W
b10 to b9	—	Reserve	These bits are read as 0. The write value should be 0.	R/W
b15 to b11	MSEL[4:0]	Output Disabling Request Mask Signal Select	When the selected signal is high, the output disabling requests from the source selected by the NFS[3:0] bits are masked. b15 b11 0 0 0 0 0: Output disabling requests are not masked. 0 0 0 0 1: The GTIOC0A signal is used to mask output disabling requests. 0 0 0 1 0: The GTIOC0B signal is used to mask output disabling requests. 0 0 0 1 1: The GTIOC1A signal is used to mask output disabling requests. 0 0 1 0 0: The GTIOC1B signal is used to mask output disabling requests. 0 0 1 0 1: The GTIOC2A signal is used to mask output disabling requests. 0 0 1 1 0: The GTIOC2B signal is used to mask output disabling requests. Settings other than above are prohibited.	R/W

The GTONCCRA register is used to configure the direct output disabling request.

NE Bit (Direct Output Disabling Request Enable)

Writing 1 to the NE bit leads to the detection signal selected by the NFS[3:0] bits being output directly to the GPTW as an output disabling request signal.

22.3 Operation

22.3.1 Output Disabling Request by Detection of Input Level or Input Edge on the GTETRGA Pin

There are two types of output disabling requests using the GTETRGA pin: requests issued via the POEGGA.PIDF flag and requests directly from the detection signal.

(1) Output Disabling Requests via the PIDF Flag

When the input level or edge set by the POEGGA.NFCS[1:0], NFEN, NFSN[1:0], NFPSC, ELSEL, and INV bits is detected while the POEGGA.PIDE bit is 1, the PIDF flag is set to 1 and an output disabling request is issued to all GPTW channels.

To cancel the output disabling request, clear the PIDF flag. For details, refer to [section 22.3.6.2, Cancellation of Output Disabling Request by Clearing Flags in the POEGGA Register](#).

(2) Direct Output Disabling Requests from the Input Signal

When the input level set by the POEGGA.NFCS[1:0], NFEN, NFSN[1:0], NFPSC, INV, and GTONCCRA.NFV bits is detected while the GTONCCRA.NFS[3:0] bits are 0111b (GTETRGA pin input signal) and the NE bit is 1, an output disabling request is directly issued to all GPTW channels.

The request is canceled when the level on the GTETRGA pin no longer matches the set condition. For details, refer to [section 22.3.7, Direct Output Disabling Request by Input Signal and Canceling the Request](#).

22.3.1.1 Digital Noise Filter

Each GTETRGA pin has a digital noise filter. [Figure 22.3](#) shows an example of the noise filter operation. This example assumes that the POEGGA.INV bit is 0 (no inversion, i.e., high level detection), NFSN[1:0] bits are 00b (three times), and NFEN bit is 1 (noise filter enabled).

The signal level on the GTETRGA pin is sampled with the sampling clock selected by the POEGGA.NFPSC and NFCS[1:0] bits, and when last three consecutive samples are high, the POEG determines that high is input and issues an output disabling request to the GPTW.

At this time, if low is detected even once, the POEG does not determine that high is input. Note that changes in the GTETRGA pin level that occur between sampling timings are ignored.

The output of the digital noise filter is used for three purposes: output disabling request via the POEGGA.PIDF flags, output disabling request directly issued by a detection signal, and an external trigger output to the GPTW.

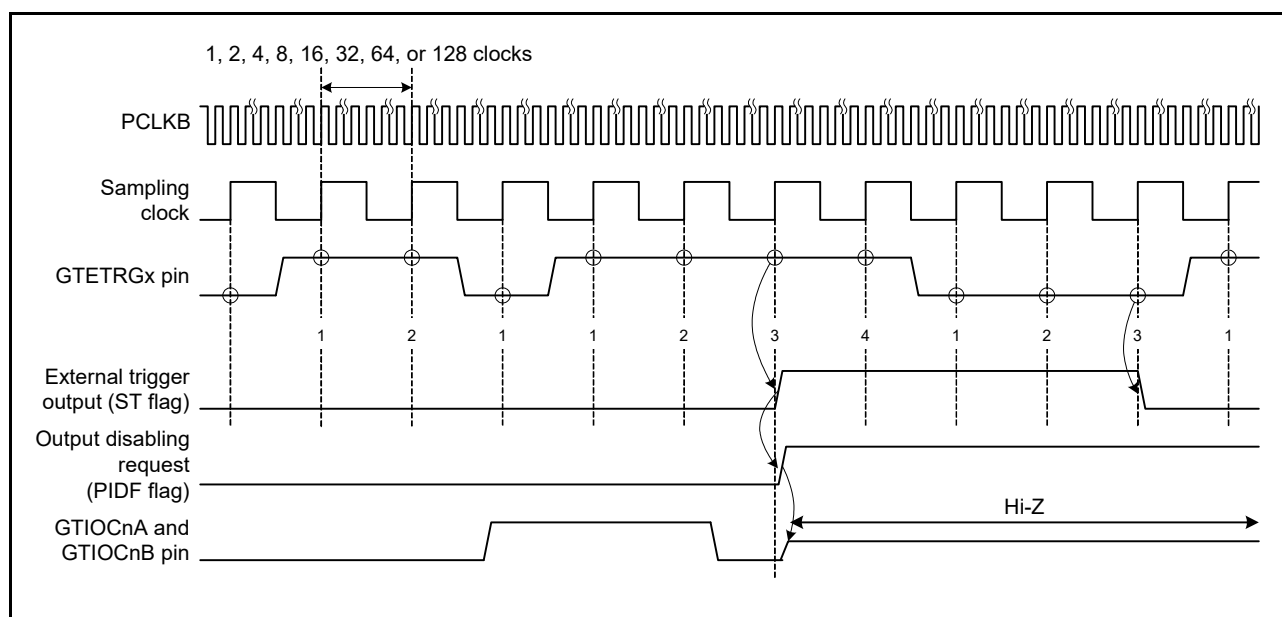


Figure 22.3 Example for Operation of Digital Noise Filter

22.3.2 Output Disabling Request by Output Disabling Condition Detection Signal from GPTW

When the GPTW detects a dead-time error or simultaneous high or low output while the POEGGA.IOCE bit is 1, the IOCF flag is set to 1 and an output disabling request is issued to all GPTW channels. The IOCF flag is also used to indicate the output disabling request by COMPn edge detection signals.

To cancel the output disabling request, clear the IOCF flag. For details, refer to [section 22.3.6.2, Cancellation of Output Disabling Request by Clearing Flags in the POEGGA Register](#).

To detect a dead-time error, simultaneous high output, or simultaneous low output in the GPTW, set the GPTWn.GTINTAD.GRPDTE, GRPABH, or GRPABL bit to 1, respectively. The output disabling condition detection signal generated by each GPTW channel is output only to the POEG group selected by the GPTWn.GTINTAD.GRP[1:0] bits. For details, refer to [section 21.2.13, General PWM Timer Interrupt Output Setting Register \(GTINTAD\)](#).

22.3.3 Output Disabling Request by Comparator Detection Signals

There are two types of output disabling requests by the comparator: requests issued via the POEGGA.IOCF flag using the COMPn edge detection signals and requests issued directly from the COMPn level detection signal.

The output disabling requests by the comparator can be masked while the GPTW output signal selected by the POEGICRA.MSEL[4:0] bits or the GTONCCRA.MSEL[4:0] bits is high.

(1) Output Disabling Requests via the IOCF Flag

When the COMPn edge detection signal becomes 1 while the POEGGA.CDREn bit is 1, the IOCF flag is set to 1 and an output disabling request is issued to all GPTW channels. The IOCF flag is also used to indicate an output disabling request by GPTW.

To cancel the output disabling request, clear the IOCF flag. For details, refer to section 22.3.6.2, Cancellation of Output Disabling Request by Clearing Flags in the POEGGA Register.

(2) Direct Output Disabling Requests from the COMPn Level Detection Signal

When the COMPn level detection signal selected by the GTONCCRA.NFS[3:0] bits matches the value set in the NFV bit while the NE bit is 1, an output disabling request is directly issued to all GPTW channels.

The request is canceled when the COMPn level detection signal no longer matches the value of the NFV bit. For details, refer to section 22.3.7, Direct Output Disabling Request by Input Signal and Canceling the Request.

22.3.4 Output Disabling Request by Oscillation Stop Detection Signal

When the oscillation stop detection circuit in the clock generation circuit detects the oscillation stop while the POEGGA.OSTPE bit is 1, the OSTPF flag is set to 1 and an output disabling request is issued to all GPTW channels.

To cancel the output disabling request, clear the OSTPF flag. For details, refer to section 22.3.6.2, Cancellation of Output Disabling Request by Clearing Flags in the POEGGA Register.

22.3.5 Output Disabling Request by a Register

When the POEGGA.SSF bit is set to 1, an output disabling request is issued to all GPTW channels.

To cancel the output disabling request, set the SSF bit to 0. For details, refer to section 22.3.6, Canceling Output Disabling Request.

22.3.6 Canceling Output Disabling Request

There are three ways to cancel an output disabling request as follows.

- Reset (return to the initial state)
- Clearing the flags in the POEGGA register
- Negating the level detection signal

22.3.6.1 Cancellation of Output Disabling Request by Reset

Any type of reset can cancel an output disabling request. For details on resets, refer to [section 6, Resets](#).

22.3.6.2 Cancellation of Output Disabling Request by Clearing Flags in the POEGGA Register

An output disabling request is canceled by setting all of the following flags and bit that are set to 1 to 0.

- POEGGA.PIDF flag
- POEGGA.IOCF flag
- POEGGA.OSTPF flag
- POEGGA.SSF bit

When the output disabling request is canceled, the GPTW captures this signal at the end of the next PWM cycle and releases the PWM outputs from the disabled state after three PCLKB cycles at the earliest. [Figure 22.4](#) shows the timing for releasing the output disabled state.

To clear the flag, make sure that the detection signals that set the flag to 1 is 0 before writing 0 to the flag. Writing 0 to the flag when the detection signal is 1 does not clear the flag. The COMPn edge detection signals and the edge detection signal of the GTETRGA signal are set to 1 immediately after the cause is detected, and then return to 0. Therefore, writing 0 at any time clears the flag.

The status of the level detection signals can be monitored using the following status flags.

- GTETRGA pin input signal: POEGGA.ST flag (GTETRGA signal status flag)
- Output disabling condition detection signals:
 - GPTWn.GTST.DTEF flag (dead time error flag)
 - GPTWn.GTST.OABLF flag (simultaneous low output flag)
 - GPTWn.GTST.OABHF flag (simultaneous high output flag)
- Oscillation stop detection signal: OSTDSR.OSTDF flag (oscillation stop detection flag)

The status of the comparator comparison results (COMPn level detection signals) before edge detection can be monitored using the following status flags.

- COMPn level detection signals: CMPCn.CMPMON.CMPMON0 flag (comparator output monitor flag)

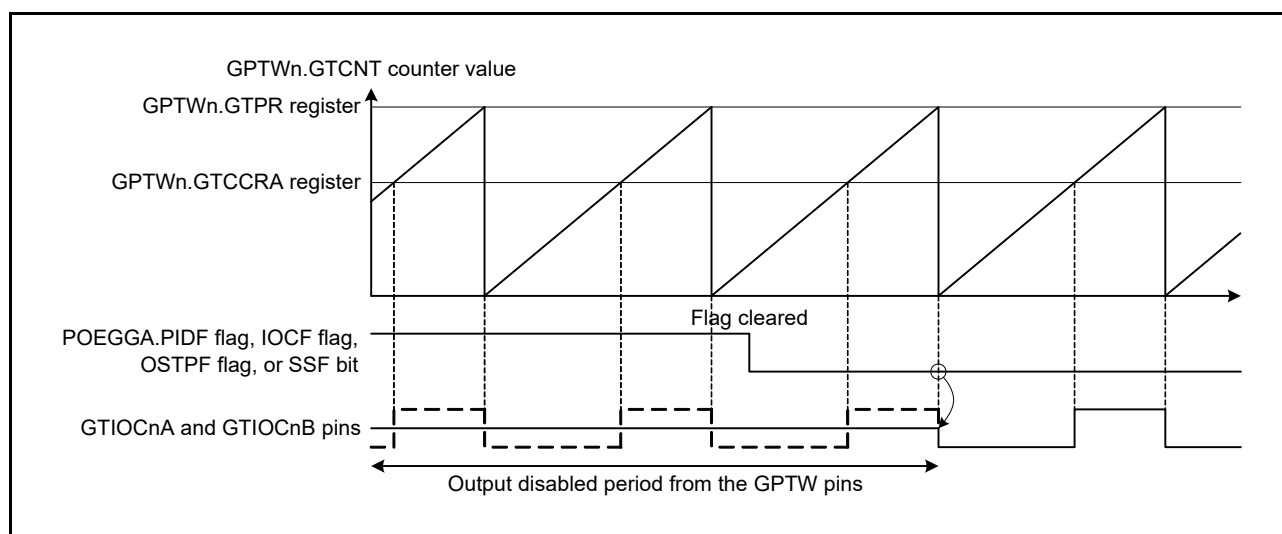


Figure 22.4 Timing for Releasing the Output Disabled State of the GPTW Pins by Canceling the Output Disabling Request

22.3.6.3 Cancellation of Output Disabling Request by Negating Level Detection Signal

Direct output disabling request from the GTETRGA pin input signal or the COMPn level detection signal is canceled when the input level of the selected signal no longer matches the value of the NFV bit.

For details, refer to section 22.3.7, Direct Output Disabling Request by Input Signal and Canceling the Request.

22.3.7 Direct Output Disabling Request by Input Signal and Canceling the Request

Direct output disabling request by input signals can be used with the input signal from the GTETRGA pin and the COMPn level detection signals ($n = 0$ to 2). The signal to be used as the source is selected by the GTONCCRA.NFS[3:0] bits. When the NE bit is 1 and the level of the selected signal matches the value set by the INV and NFV bits, an output disabling request is issued to the GPTW.

The output disabling request is canceled when the level of the input signal from the GTETRGA pin or the COMPn level detection signals no longer matches the issuing conditions. When canceling a request, make sure that the output disabling request has not been issued for any other reason (both the POEGGA.PIDF and IOCF flags are 0).

Figure 22.5 shows an example of the output disabling operation by the COMP0 level detection signal. In this example, when the analog input voltage on CMPC0 becomes higher than the reference input voltage while the GPTW0 is outputting a PWM waveform from the GTIOC0A pin, the COMP0 level detection signal goes high. The POEG issues an output disabling request to the GPTW0 based on this signal, and the GPTW0 immediately disables the output. The GPTW0 maintains the output disabled state until the end of the PWM cycle, even if the output disabling request is canceled before the end of the PWM cycle. If the output disabling request is not canceled at the end of a PWM cycle, GPTW0 continues in the output disabled state until the end of the next PWM cycle.

The status of the output disabling request can be checked with the GTST.ODF flag of the GPTW. For details, refer to the description of the ODF flag in section 21.2.14, General PWM Timer Status Register (GTST).

Figure 22.6 shows an example of the procedure for setting a direct output disabling request by an input signal.

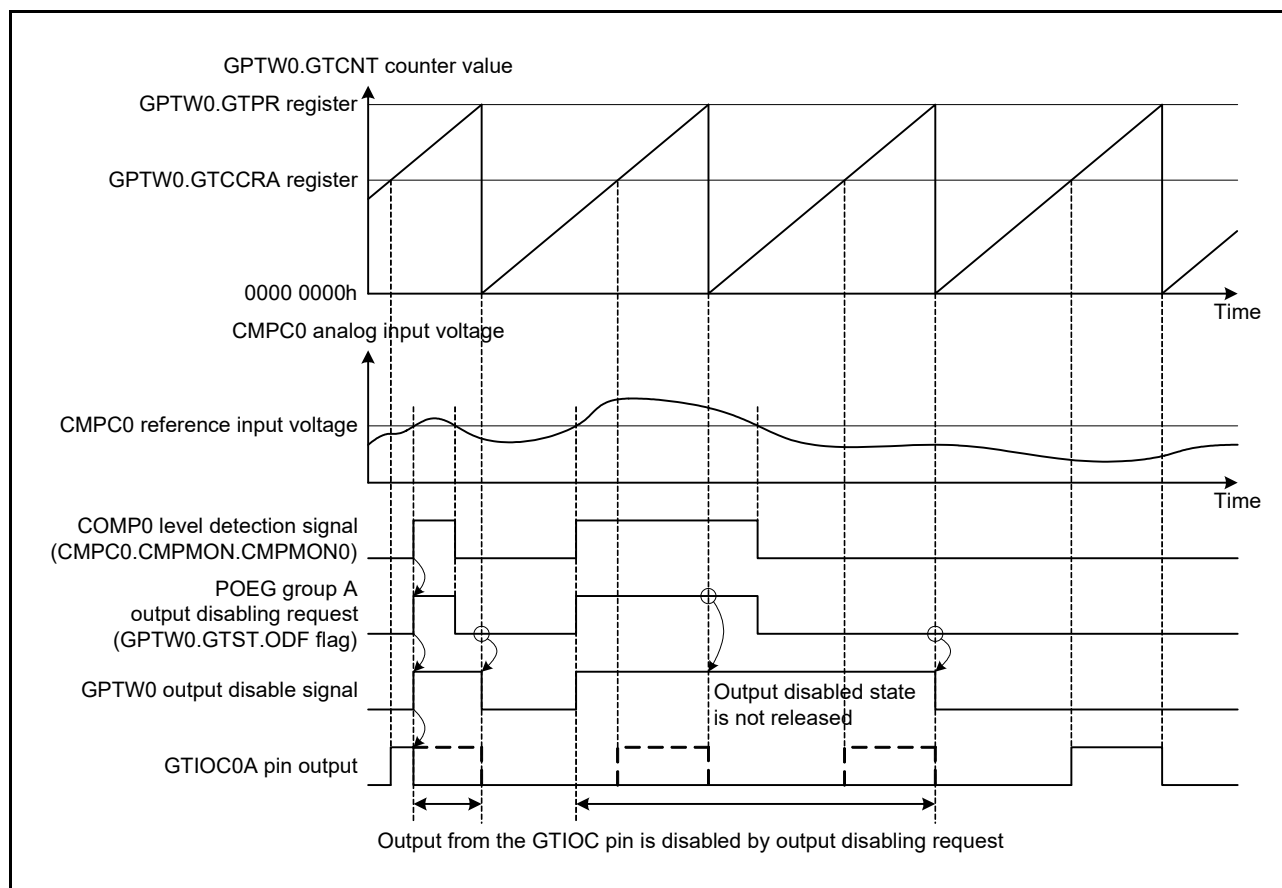


Figure 22.5 Example of Operation to Disable Output from the GTIOC Pin by COMP0 Level Detection Signal

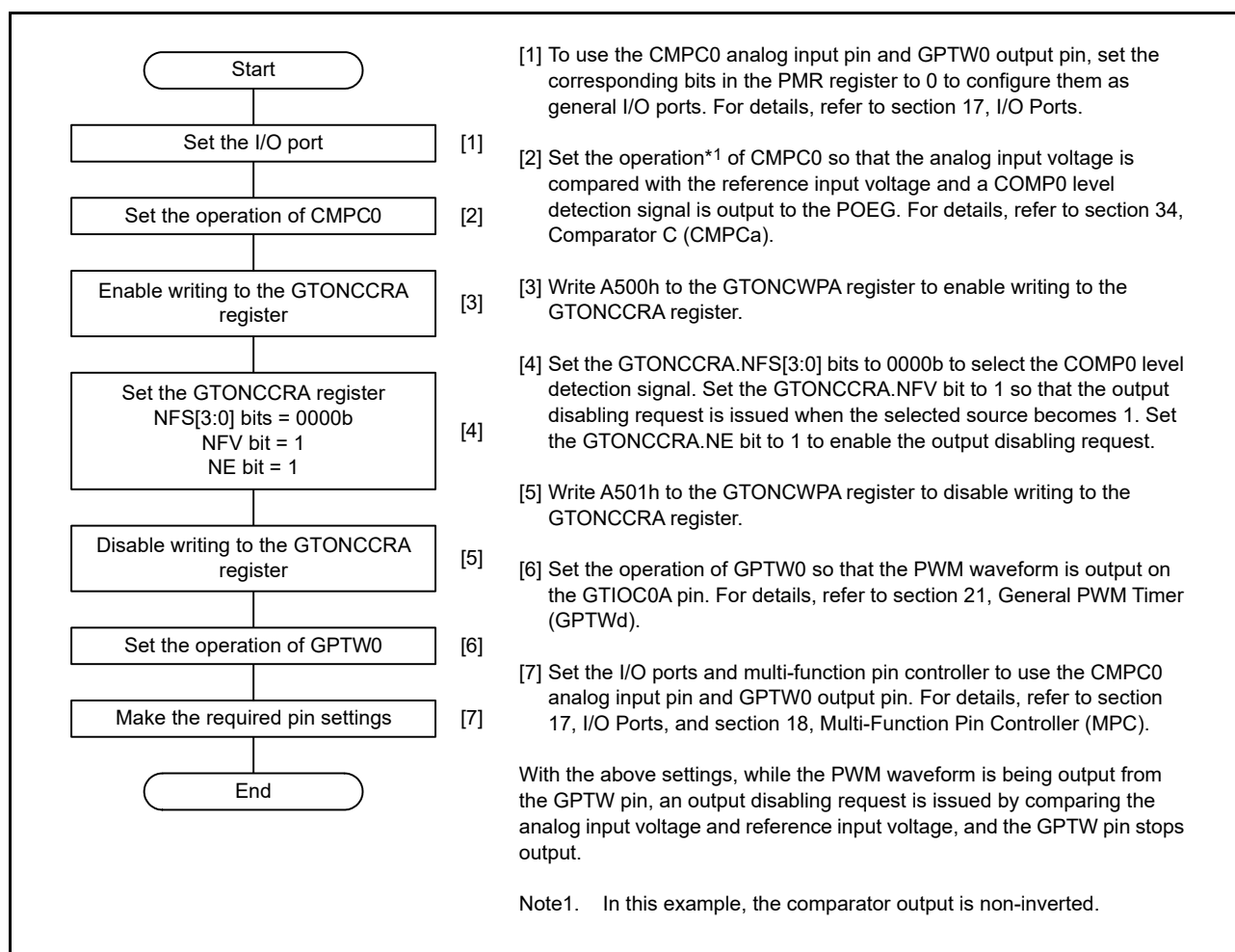


Figure 22.6 Example of Settings to Disable the GTIOC Pin Output by COMP0 Level Detection Signal

22.4 Interrupt Sources

The POEG has the following interrupt sources for each group.

- Detection of input level or input edge on the GTETRGA pin
- Detection of output disabling condition detection signal from the GPTW
- Detection of COMPn edge detection signal (n = 0 to 2)

Table 22.3 lists the interrupt sources.

Table 22.3 Interrupt sources

Interrupt Name	Symbol	Corresponding Flag	Interrupt Source
Group A Interrupt	POEGGAI	POEGGA.PIDF	Input signal from the GTETRGA pin
		POEGGA.IOCF	Output disabling condition detection signal from the GPTW
			COMPn edge detection signal

22.5 External Trigger Output to GPTW

The POEG outputs the input signals on the GTETRGA pin to the GPTW as external trigger signals via polarity conversion and a digital noise filter. The external trigger signal can be monitored by the POEGGA.ST flag. The GPTW can perform the following operations with the external trigger signal.

- Count start
- Count stop
- Counter clear
- Count up
- Count down
- Input capture

For details, refer to section 21, General PWM Timer (GPTWd).

22.6 Usage Notes

22.6.1 Transitions to Low-Power Consumption Mode

When using the POEG, do not enter software standby mode. In this mode, the POEG is stopped and output disabling request cannot be issued.

22.6.2 Setting the Function for Stopping the Module

Operation of the POEG can be disabled or enabled using the module stop control register A (MSTPCRA). After reset is released, the POEG function is disabled. Register access is enabled by releasing the module stop state. For details, refer to section 11, Low Power Consumption. Note that the module stop bit for the POEG is also used for the GPTW.

22.6.3 Duplication of Output Disabling Request

The output disabling request via flags is a logical OR of the POEGGA.PIDF, IOCF, OSTPF flags, and the SSF bit.

Therefore, note that the output disabling request will not be canceled if any one of the flags/bit is set to 1.

Also, regarding the direct output disabling request by an input signal selected by the GTONCCRA register, if any of the above flags/bit is set to 1, the output disabling request will not be canceled even if the input signal is negated.

22.6.4 Masking of Output Disabling Request by COMPn Edge Detection Signal

If the POEGICRA.MSEL[4:0] bits are set after the POEGGA.CDREn bits are set to 1, the POEGGA.IOCF flag may unintentionally be set to 1. Set the POEGICRA.MSEL[4:0] bits before setting the POEGGA.CDREn bits to 1.

23. 8-Bit Timer (TMR)

This MCU has two units (unit 0, unit 1) of an on-chip 8-bit timer (TMR) module that comprise two 8-bit counter channels, totaling four channels. The 8-bit timer module can be used to count external events and also be used as a multi-function timer in a variety of applications, such as generation of counter reset signal, interrupt requests, and pulse output with a desired duty cycle using a compare-match signal with two registers.

Unit 0 and unit 1 have the same functions, and can generate a base clock for the SCI.

In this section, “PCLK” is used to refer to PCLKB.

23.1 Overview

Table 23.1 lists the specifications of the TMR. Table 23.2 lists the TMR functions.

Figure 23.1 shows a block diagram of the 8-bit timer module (unit 0), and Figure 23.2 shows that of the 8-bit timer module (unit 1).

Table 23.1 Specifications of TMR

Item	Description
Count clock	- Internal clock: PCLK/1, PCLK/2, PCLK/8, PCLK/32, PCLK/64, PCLK/1024, PCLK/8192 - External clock: external count clock
Number of channels	(8 bits × 2 channels) × 2 units
Compare match	8-bit mode (compare match A, compare match B) 16-bit mode (compare match A, compare match B)
Counter clear	Selected by compare match A or B, or an external counter reset signal.
Timer output	Output pulses with a desired duty cycle or PWM output
Cascading of two channels	16-bit count mode 16-bit timer using TMR0 for the upper 8 bits and TMR1 for the lower 8 bits (TMR2 for the upper 8 bits and TMR3 for the lower 8 bits) - Compare match count mode TMR1 can be used to count TMR0 compare matches (TMR3 can be used to count TMR2 compare matches).
Interrupt sources	Compare match A, compare match B, and overflow
DTC activation	DTC can be activated by compare match A interrupts or compare match B interrupts.
A/D conversion start trigger of the A/D converter	Compare match A of TMR0 and TMR2
Capable of generating base clock for SCI	Generates base clock for SCI.*1
Low power consumption function	Each unit can be placed in a module stop state

Note 1. For details, refer to section 26, Serial Communications Interface (SCIg, SCIH).

Table 23.2 TMR Functions

Item		Unit 0			Unit 1		
Counter mode		8 Bits		16 Bits	8 Bits		16 Bits
Channel		TMR0	TMR1	TMR0 + TMR1	TMR2	TMR3	TMR2 + TMR3
Count clock		PCLK/1 PCLK/2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCi0	PCLK/1 PCLK/2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCi1	PCLK/1 PCLK/2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCi1	PCLK/1 PCLK/2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCi2	PCLK/1 PCLK/2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCi3	PCLK/1 PCLK/2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCi3
Counter clear		TMR0.TCORA TMR0.TCORB TMRi0	TMR1.TCORA TMR1.TCORB TMRi1	TMR0.TCORA + TMR1.TCORA TMR0.TCORB + TMR1.TCORB TMRi0	TMR2.TCORA TMR2.TCORB TMRi2	TMR3.TCORA TMR3.TCORB TMRi3	TMR2.TCORA + TMR3.TCORA TMR2.TCORB + TMR3.TCORB TMRi2
Compare match	Compare match A	✓	✓	✓	✓	✓	✓
	Compare match B	✓	✓	✓	✓	✓	✓
Timer output	Low output	✓	✓	✓	✓	✓	✓
	High output	✓	✓	✓	✓	✓	✓
	Toggle output	✓	✓	✓	✓	✓	✓
DTC activation	Compare match A	✓	✓	✓	✓	✓	✓
	Compare match B	✓	✓	✓	✓	✓	✓
	TCNT overflow	—	—	—	—	—	—
Interrupt	Compare match A	CMIA0	CMIA1	CMIA0	CMIA2	CMIA3	CMIA2
	Compare match B	CMIB0	CMIB1	CMIB0	CMIB2	CMIB3	CMIB2
	TCNT overflow	OVI0	OVI1	OVI0	OVI2	OVI3	OVI2
Cascaded connection		TMR1 overflow	TMR0 compare match A	—	TMR3 overflow	TMR2 compare match A	—
A/D conversion start trigger of the A/D converter*1		✓	—	✓	✓	—	✓
SCI base clock generation*2		✓		—	✓		—
Module stop setting*3		MSTPCRA.MSTPA5 bit (unit 0), MSTPCRA.MSTPA4 bit (unit 1)					

✓: Possible

—: Impossible

Note 1. For details, refer to section 30, 12-Bit A/D Converter (S12ADF).

Note 2. For details, refer to section 26, Serial Communications Interface (SCIg, SCIf).

Note 3. For details, refer to section 11, Low Power Consumption.

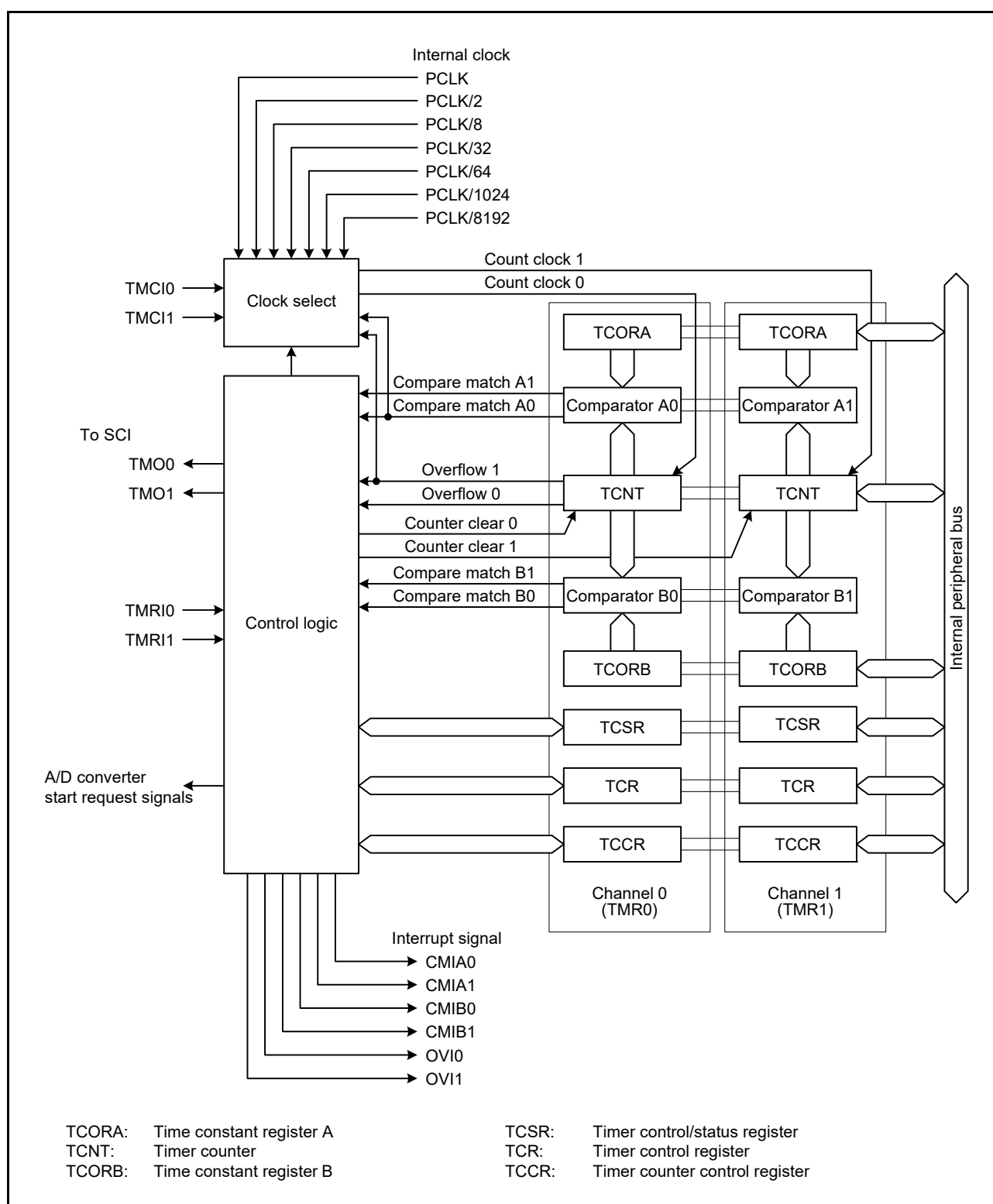


Figure 23.1 Block Diagram of TMR (Unit 0)



Table 23.3 lists the I/O pins of the TMR.

Table 23.3 Pin Configuration of TMR

Unit	Channel	Pin Name	I/O	Description
0	TMR0	TMO0	Output	Outputs compare match
		TMCi0	Input	Inputs external count clock
		TMRI0	Input	Inputs external counter reset
	TMR1	TMO1	Output	Outputs compare match
		TMCi1	Input	Inputs external count clock
		TMRI1	Input	Inputs external counter reset
1	TMR2	TMO2	Output	Outputs compare match
		TMCi2	Input	Inputs external count clock
		TMRI2	Input	Inputs external counter reset
	TMR3	TMO3	Output	Outputs compare match
		TMCi3	Input	Inputs external count clock
		TMRI3	Input	Inputs external counter reset

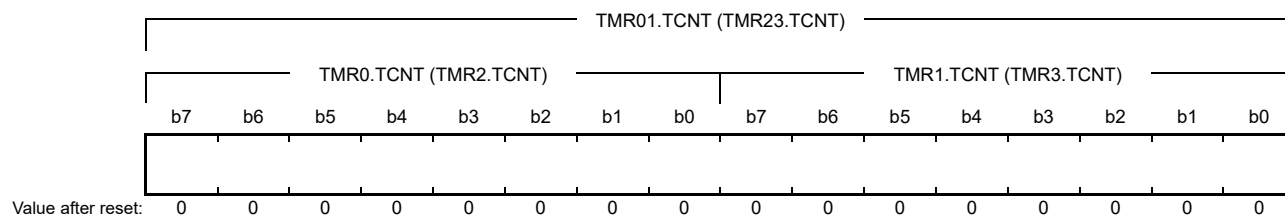
23.2 Register Descriptions

Table 23.4 Register Allocation for 16-Bit Access

Address	Register	Upper 8 Bits	Lower 8 Bits
0008 8208h	TMR01.TCNT	TMR0.TCNT	TMR1.TCNT
0008 8204h	TMR01.TCORA	TMR0.TCORA	TMR1.TCORA
0008 8206h	TMR01.TCORB	TMR0.TCORB	TMR1.TCORB
0008 820Ah	TMR01.TCCR	TMR0.TCCR	TMR1.TCCR
0008 8218h	TMR23.TCNT	TMR2.TCNT	TMR3.TCNT
0008 8214h	TMR23.TCORA	TMR2.TCORA	TMR3.TCORA
0008 8216h	TMR23.TCORB	TMR2.TCORB	TMR3.TCORB
0008 821Ah	TMR23.TCCR	TMR2.TCCR	TMR3.TCCR

23.2.1 Timer Counter (TCNT)

Address(es): TMR0.TCNT 0008 8208h, TMR1.TCNT 0008 8209h, TMR2.TCNT 0008 8218h, TMR3.TCNT 0008 8219h,
TMR01.TCNT 0008 8208h, TMR23.TCNT 0008 8218h



TCNT is an 8-bit readable/writable up-counter.

TMR0.TCNT and TMR1.TCNT (TMR2.TCNT and TMR3.TCNT) comprise a single 16-bit counter (TMR01.TCNT, TMR23.TCNT) so they can be accessed together in 16-bit units.

The TCCR.CSS[1:0] and CKS[2:0] bits are used to select a count clock.

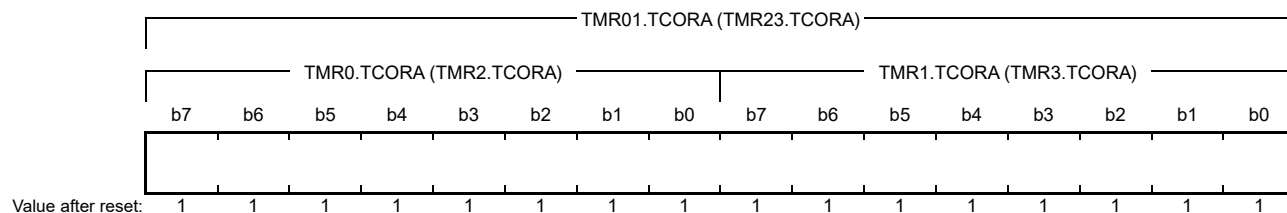
TCNT can be cleared by an external counter reset signal, compare match A, or compare match B. Which compare match to be used for clearing is selected by the TCR.CCLR[1:0] bits.

When TCNT overflows (its value changes from FFh to 00h), an overflow interrupt is output provided the interrupt request is enabled by the TCR.OVIE bit.

For details on the corresponding interrupt vector number, refer to section 14, Interrupt Controller (ICUb), and Table 23.6, TMR Interrupt Sources.

23.2.2 Time Constant Register A (TCORA)

Address(es): TMR0.TCORA 0008 8204h, TMR1.TCORA 0008 8205h, TMR2.TCORA 0008 8214h, TMR3.TCORA 0008 8215h,
TMR01.TCORA 0008 8204h, TMR23.TCORA 0008 8214h



TCORA is an 8-bit readable/writable register.

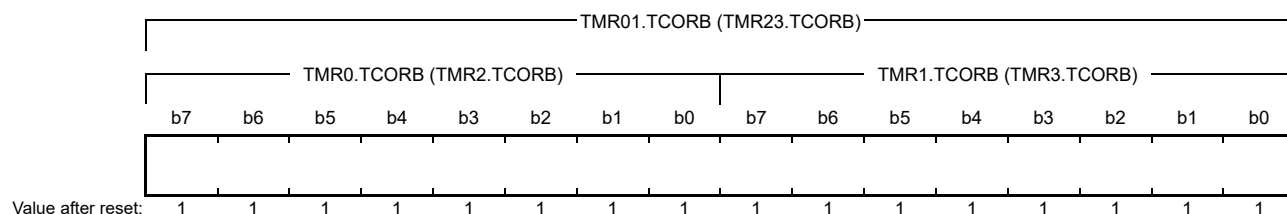
TMR0.TCORA and TMR1.TCORA (TMR2.TCORA and TMR3.TCORA) comprise a single 16-bit register (TMR01.TCORA, TMR23.TCORA) so they can be accessed together in 16-bit units.

The value in TCORA is continually compared with the value in TCNT. When a match is detected, the corresponding compare match A is generated, and a compare match A interrupt is output provided the interrupt request is enabled by the TCR.CMIEA bit.

However, comparison is not performed during writing to TCORA. The timer output from the TMO pin can be freely controlled by this compare match A and the settings of the TCSR.OSA[1:0] bits.

23.2.3 Time Constant Register B (TCORB)

Address(es): TMR0.TCORB 0008 8206h, TMR1.TCORB 0008 8207h, TMR2.TCORB 0008 8216h, TMR3.TCORB 0008 8217h,
TMR01.TCORB 0008 8206h, TMR23.TCORB 0008 8216h



TCORB is an 8-bit readable/writable register.

TMR0.TCORB and TMR1.TCORB (TMR2.TCORB and TMR3.TCORB) comprise a single 16-bit register (TMR01.TCORB, TMR23.TCORB) so they can be accessed together in 16-bit units.

The value in TCORB is continually compared with the value in TCNT. When a match is detected, the corresponding compare match B is generated, and a compare match B interrupt is output provided the interrupt request is enabled by the TCR.CMIEB bit.

However, comparison is not performed during writing to TCORB. The timer output from the TMO pin can be freely controlled by this compare match B and the settings of the TCSR.OSB[1:0] bits.

23.2.4 Timer Control Register (TCR)

Address(es): TMR0.TCR 0008 8200h, TMR1.TCR 0008 8201h, TMR2.TCR 0008 8210h, TMR3.TCR 0008 8211h

b7	b6	b5	b4	b3	b2	b1	b0
CMIEB	CMIEA	OVIE	CCLR[1:0]	—	—	—	—
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4, b3	CCLR[1:0]	Counter Clear	b4 b3 0 0: Clearing is disabled 0 1: Cleared by compare match A 1 0: Cleared by compare match B 1 1: Cleared by the external counter reset signal*1 (Select edge or level by the TMRIS bit in TCCR.)	R/W
b5	OVIE	Overflow Interrupt Enable	0: Overflow interrupt requests (OVIn) are disabled 1: Overflow interrupt requests (OVIn) are enabled	R/W
b6	CMIEA	Compare Match A Interrupt Enable	0: Compare match A interrupt requests (CMIAIn) are disabled 1: Compare match A interrupt requests (CMIAIn) are enabled	R/W
b7	CMIEB	Compare Match B Interrupt Enable	0: Compare match B interrupt requests (CMIBIn) are disabled 1: Compare match B interrupt requests (CMIBIn) are enabled	R/W

Note 1. To use an external counter reset signal, set the corresponding pin function. For details, refer to section 17, I/O Ports and section 18, Multi-Function Pin Controller (MPC).

CCLR[1:0] Bits (Counter Clear)

Select the condition by which TCNT is cleared.

OVIE Bit (Overflow Interrupt Enable)

Selects whether overflow interrupt requests (OVIn) issued by TCNT are enabled or disabled.

CMIEA Bit (Compare Match A Interrupt Enable)

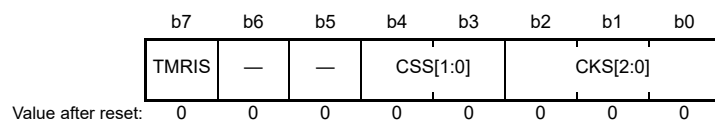
Selects whether compare match A interrupt requests (CMIAIn) that are issued when the value of TCORA corresponds to that of TCNT are enabled or disabled.

CMIEB Bit (Compare Match B Interrupt Enable)

Selects whether compare match B interrupt requests (CMIBIn) that are issued when the value of TCORB corresponds to that of TCNT are enabled or disabled.

23.2.5 Timer Counter Control Register (TCCR)

Address(es): TMR0.TCCR 0008 820Ah, TMR1.TCCR 0008 820Bh, TMR2.TCCR 0008 821Ah, TMR3.TCCR 0008 821Bh,
TMR01.TCCR 0008 820Ah, TMR23.TCCR 0008 821Ah



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	CKS[2:0]	Clock Select* ¹	See Table 23.5.	R/W
b4, b3	CSS[1:0]	Clock Source Select	See Table 23.5.	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TMRIS	Timer Reset Detection Condition Select	0: Cleared at rising edge of the external counter reset signal 1: Cleared when the external counter reset signal is high	R/W

Note 1. To use an external count clock, set the corresponding pin function. For details, refer to section 17, I/O Ports and section 18, Multi-Function Pin Controller (MPC).

TCCR register is a 8-bit register used to configure the basic operation of the counter. Two TCCR registers can be accessed simultaneously by accessing the address of the even channel TCCR register in 16-bit units.

CKS[2:0] Bits (Clock Select)

CSS[1:0] Bits (Clock Source Select)

The CKS[2:0] and CSS[1:0] bits select a clock. For details, see Table 23.5.

TMRIS Bit (Timer Reset Detection Condition Select)

This bit is enabled when the TCR.CCLR[1:0] bits are 11b (cleared by external counter reset signal) and selects the condition for detecting counter reset (level or edge).

Table 23.5 Clock Input to TCNT and Count Condition

Channel	TCCR Register					Description
	CSS[1:0]		CKS[2:0]			
	b4	b3	b2	b1	b0	
TMR0 (TMR2)	0	0	—	0	0	Clock input prohibited
					1	Uses external count clock. Counts at rising edge*1.
				1	0	Uses external count clock. Counts at falling edge*1.
					1	Uses external count clock. Counts at both rising and falling edges*1.
	0	1	0	0	0	Uses internal clock. Counts at PCLK.
					1	Uses internal clock. Counts at PCLK/2.
				1	0	Uses internal clock. Counts at PCLK/8.
					1	Uses internal clock. Counts at PCLK/32.
			1	0	0	Uses internal clock. Counts at PCLK/64.
					1	Uses internal clock. Counts at PCLK/1024.
				1	0	Uses internal clock. Counts at PCLK/8192.
					1	Clock input prohibited
	1	0	—	—	—	Setting prohibited
	1	1	—	—	—	Counts at TMR1.TCNT (TMR3.TCNT) overflow signal*2.
TMR1 (TMR3)	0	0	—	0	0	Clock input prohibited
					1	Uses external count clock. Counts at rising edge*1.
				1	0	Uses external count clock. Counts at falling edge*1.
					1	Uses external count clock. Counts at both rising and falling edges*1.
	0	1	0	0	0	Uses internal clock. Counts at PCLK.
					1	Uses internal clock. Counts at PCLK/2.
				1	0	Uses internal clock. Counts at PCLK/8.
					1	Uses internal clock. Counts at PCLK/32.
			1	0	0	Uses internal clock. Counts at PCLK/64.
					1	Uses internal clock. Counts at PCLK/1024.
				1	0	Uses internal clock. Counts at PCLK/8192.
					1	Clock input prohibited
	1	0	—	—	—	Setting prohibited
	1	1	—	—	—	Counts at TMR0.TCNT (TMR2.TCNT) compare match A*2.

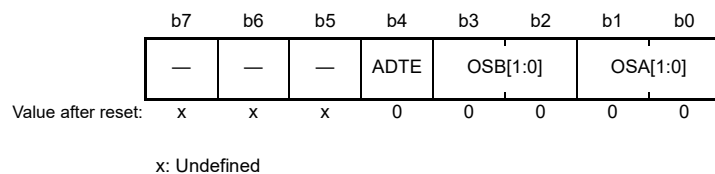
Note 1. To use an external count clock, set the corresponding pin function. For details, refer to section 17, I/O Ports and section 18, Multi-Function Pin Controller (MPC).

Note 2. If the clock input of TMR0 (TMR2) is the overflow signal of the TMR1.TCNT (TMR3.TCNT) counter and that of TMR1 (TMR3) is the compare match signal of the TMR0.TCNT (TMR2.TCNT) counter, no TCNT count clock is generated. Do not use this setting.

23.2.6 Timer Control/Status Register (TCSR)

- TMR0.TCSR, TMR2.TCSR

Address(es): TMR0.TCSR 0008 8202h, TMR2.TCSR 0008 8212h



Bit	Symbol	Bit Name	Description	R/W
b1, b0	OSA[1:0]	Output Select A* ¹	b1 b0 0 0: No change 0 1: Low is output 1 0: High is output 1 1: Output is inverted (toggle output)	R/W
b3, b2	OSB[1:0]	Output Select B* ¹	b3 b2 0 0: No change 0 1: Low is output 1 0: High is output 1 1: Output is inverted (toggle output)	R/W
b4	ADTE	A/D Trigger Enable	0: A/D conversion start request in response to compare match A is disabled. 1: A/D conversion start request in response to compare match A is enabled.	R/W
b7 to b5	—	Reserved	These bits are read as an undefined value. The write value should be 1.	R/W

Note 1. When all OSA[1:0] and OSB[1:0] bits are 0, the output enable signal corresponding to the TMO_n pin is negated and a request for high-impedance output is issued to the I/O port. Timer output pin is driven low until the first compare-match occurs after a reset when at least one of the OSA[1:0] and OSB[1:0] bits is 1.

OSA[1:0] Bits (Output Select A)

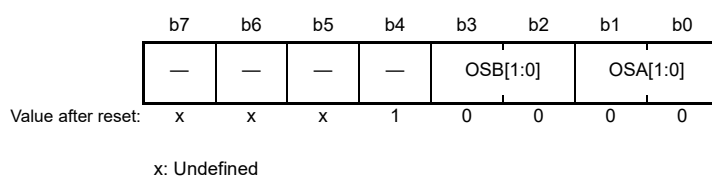
These bits select a method of TMO_n pin output when compare match A of TCORA and TCNT occurs.

OSB[1:0] Bits (Output Select B)

These bits select a method of TMO_n pin output when compare match B of TCORB and TCNT occurs.

- TMR1.TCSR, TMR3.TCSR

Address(es): TMR1.TCSR 0008 8203h, TMR3.TCSR 0008 8213h



Bit	Symbol	Bit Name	Description	R/W
b1, b0	OSA[1:0]	Output Select A*1	b1 b0 0 0: No change 0 1: Low is output 1 0: High is output 1 1: Output is inverted (toggle output)	R/W
b3, b2	OSB[1:0]	Output Select B*1	b3 b2 0 0: No change 0 1: Low is output 1 0: High is output 1 1: Output is inverted (toggle output)	R/W
b4	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b7 to b5	—	Reserved	These bits are read as an undefined value. The write value should be 1.	R/W

Note 1. When all OSA[1:0] and OSB[1:0] bits are 0, the output enable signal corresponding to the TMO pin is negated and a request for high-impedance output is issued to the I/O port. Timer output pin is driven low until the first compare match occurs after a reset when at least one of the OSA[1:0] and OSB[1:0] bits is 1.

OSA[1:0] Bits (Output Select A)

These bits select a method of TMO pin output when compare match A of TCORA and TCNT occurs.

OSB[1:0] Bits (Output Select B)

These bits select a method of TMO pin output when compare match B of TCORB and TCNT occurs.

23.3 Operation

23.3.1 Pulse Output

Figure 23.3 shows an example of the 8-bit timer being used to generate a pulse output with a desired duty cycle.

1. Set the TCR.CCLR[1:0] bits to 01b (cleared by compare match A) so that TCNT is cleared at a compare match of TCORA.
2. Set the TCSR.OSA[1:0] bits to 10b (high is output) and TCSR.OSB[1:0] bits to 01b (low is output), causing the output to change to high at a compare match of TCORA and to low at a compare match of TCORB.

With these settings, the 8-bit timer provides pulses output at a cycle determined by TCORA with a pulse width determined by TCORB. No software intervention is required.

The timer output pin is low after the TCSR.OSA[1:0] or TCSR.OSB[1:0] bits are set until the first compare match occurs after a reset.

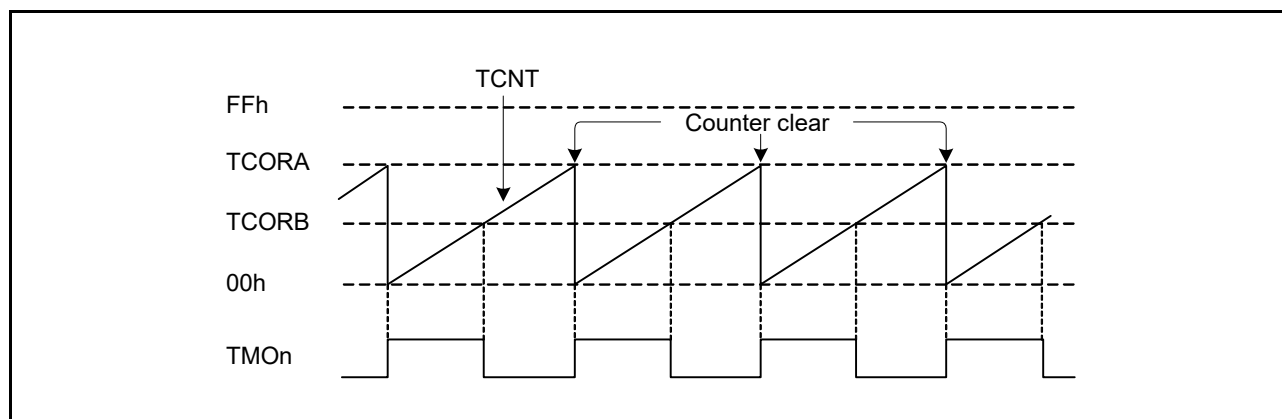


Figure 23.3 Example of Pulse Output (n = 0 to 3)

23.3.2 External Counter Reset Input

Figure 23.4 shows an example of the 8-bit timer being used to generate a pulse which is output after a desired delay time from a TMRIn input.

1. Set the TCR.CCLR[1:0] bits to 11b (cleared by external counter reset signal) and set the TMRIS bit in TCCR to 1 (cleared when the external counter reset signal is high) so that TCNT is cleared at the high level input of the TMRIn signal.
2. Set the TCSR.OSA[1:0] bits to 10b (high output) and the TCSR.OSB[1:0] bits to 01b (low output), causing the output to change to high at a compare match of TCORA and to low at a compare match of TCORB.

With these settings, the 8-bit timer provides pulses output at a desired delay time from a TMRIn input determined by TCORA and with a pulse width determined by TCORB and TCORA.

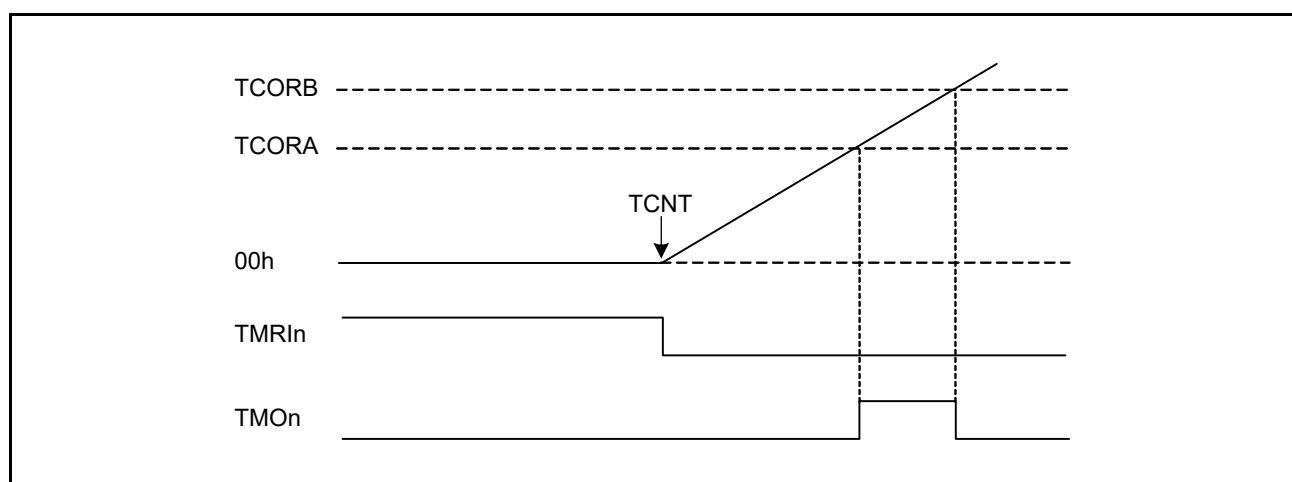


Figure 23.4 Example of External Counter Reset Signal Input (n = 0 to 3)

23.4 Operation Timing

23.4.1 TCNT Count Timing

Figure 23.5 shows the count timing of TCNT for internal clock. Figure 23.6 shows the count timing of TCNT for external clock.

Note that the external clock pulse width must be at least 1.5 PCLK cycles for increment at a single edge, and at least 2.5 PCLK cycles for increment at both edges. The counter will not increment correctly if the pulse width is less than these values.

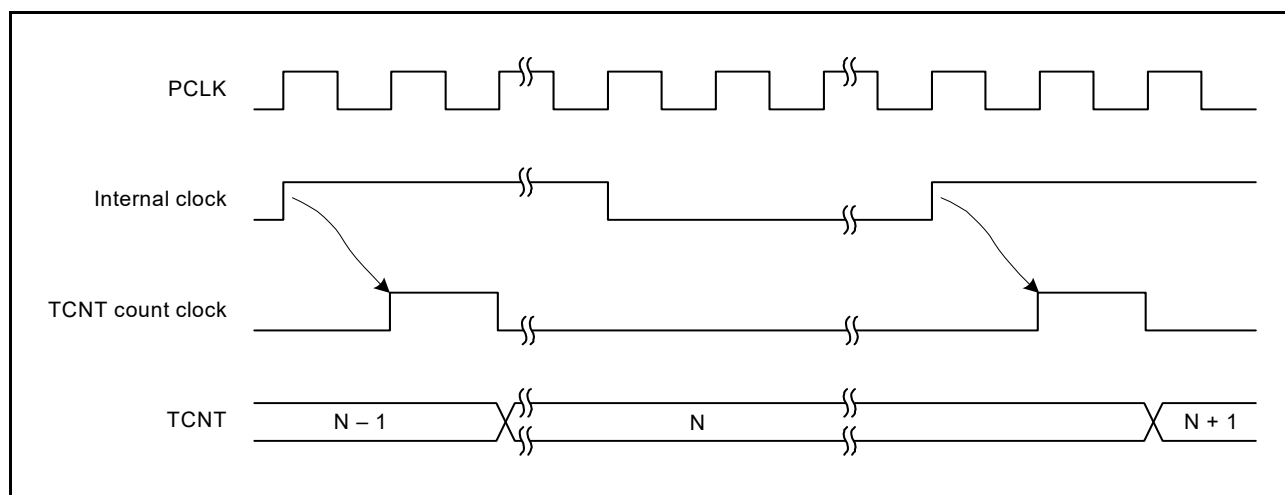


Figure 23.5 Count Timing for Internal Clock

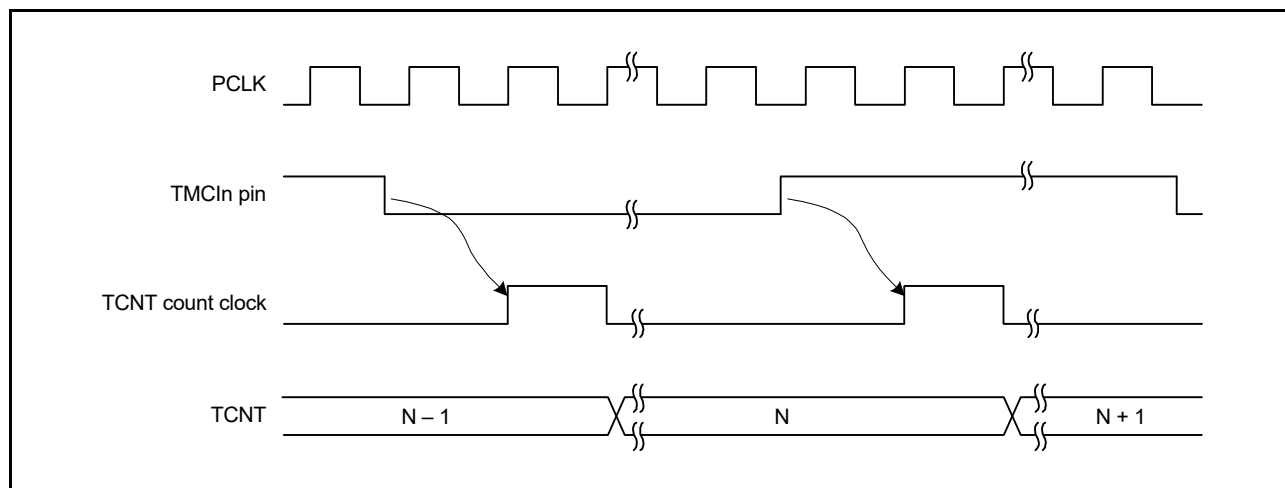


Figure 23.6 Count Timing for External Clock (at Both Edges)

23.4.2 Timing of Interrupt Signal Output on a Compare Match

A compare match refers to a match between the value of the TCORA or TCORB register and the TCNT, and a compare match interrupt signal is output at this time if the interrupt request is enabled. The compare match is generated in the last cycle in which the values match (at the time at which the value counted by TCNT to produce the match is updated). Accordingly, after a match between TCNT and the TCORA or TCORB register is detected, the compare match is not actually generated until the next cycle of the TCNT count clock. Figure 23.7 shows the timing of output of the interrupt signal.

For the corresponding interrupt vector number, refer to section 14, Interrupt Controller (ICUb) and Table 23.6.

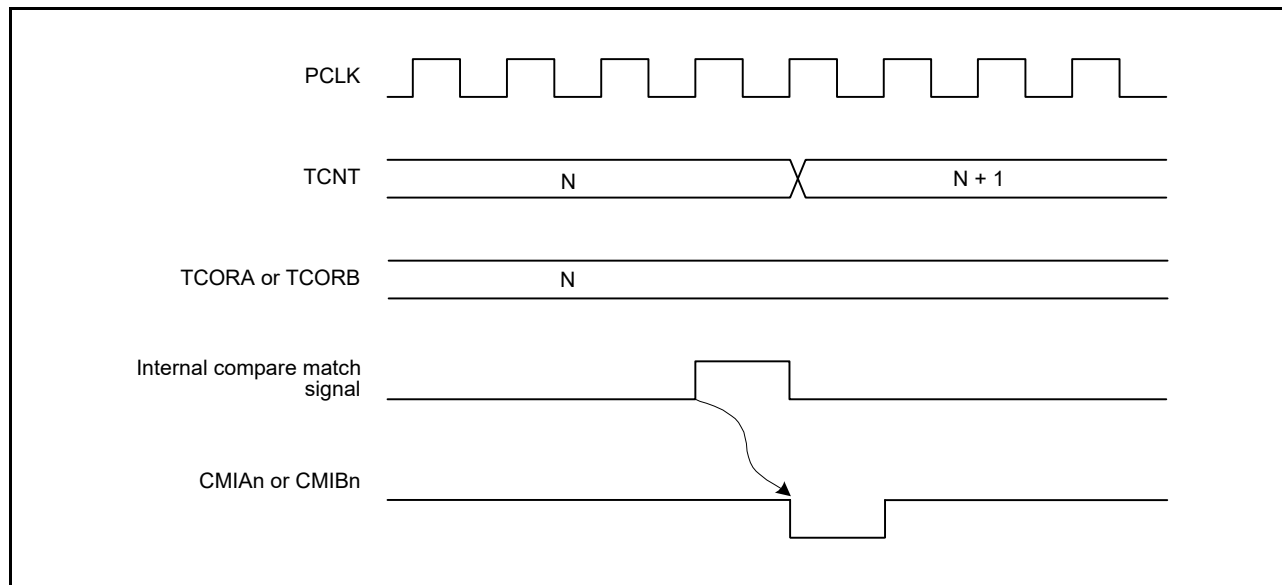


Figure 23.7 Timing of Interrupt Flag Setting to 1 at Compare Match ($n = 0$ to 3)

23.4.3 Timing of Timer Output Signal at Compare Match

When a compare match signal is generated, the output value specified by the TCSR.OSA[1:0] and OSB[1:0] bits is output on the timer output pin (TMO_n).

Figure 23.8 shows the timing when the timer output is toggled by the compare match A signal.

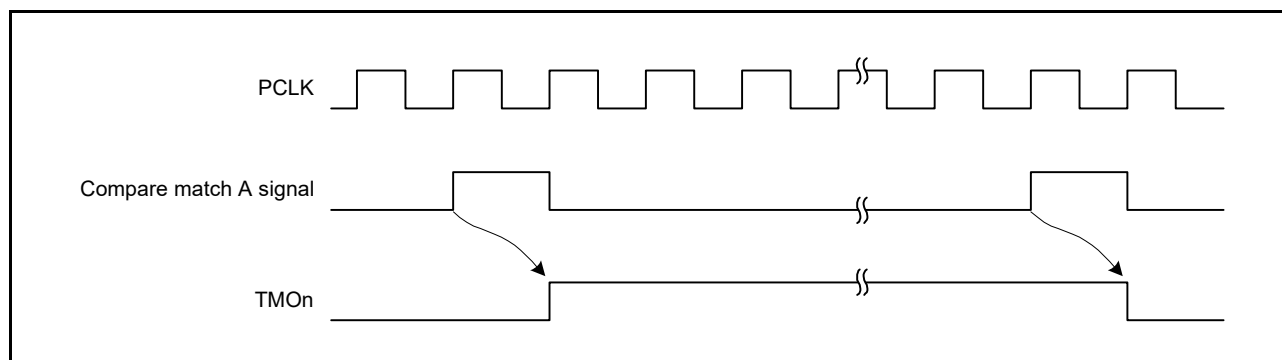


Figure 23.8 Timing of Timer Output Signal at Compare Match A Signal ($n = 0$ to 3)

23.4.4 Timing of Counter Clear by Compare Match

TCNT is cleared when compare match A or B occurs, depending on the settings of the TCR.CCLR[1:0] bits. Figure 23.9 shows the timing of this operation.

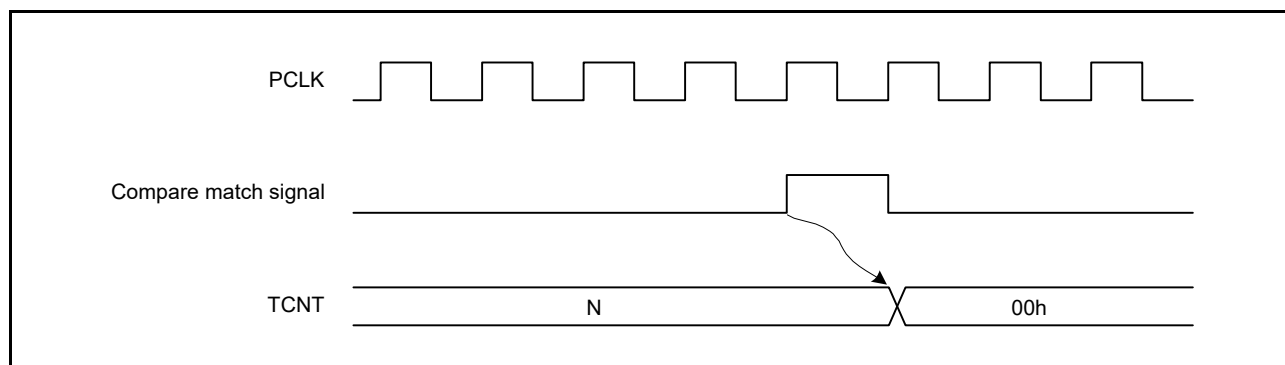


Figure 23.9 Timing of Counter Clear by Compare Match

23.4.5 Timing of the External Reset for TCNT

TCNT is cleared at the rising edge or high level of an external counter reset signal, depending on the settings of the TCR.CCLR[1:0] bits. At least 2 PCLK cycles are required from a reset input to clearing of TCNT. Figure 23.10 and Figure 23.11 show the timing of this operation.

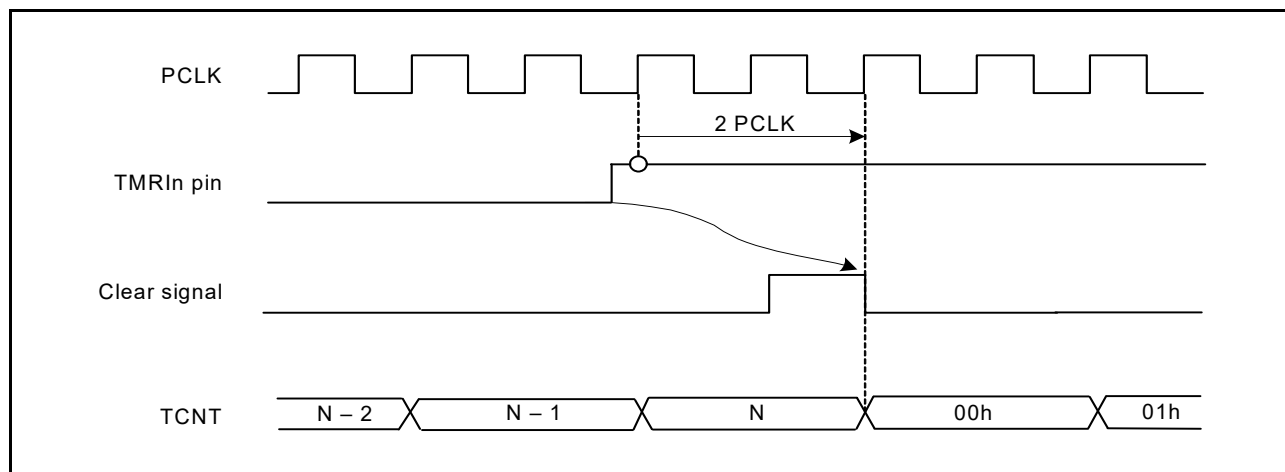


Figure 23.10 Clear Timing by External Counter Reset Signal (Rising Edge)

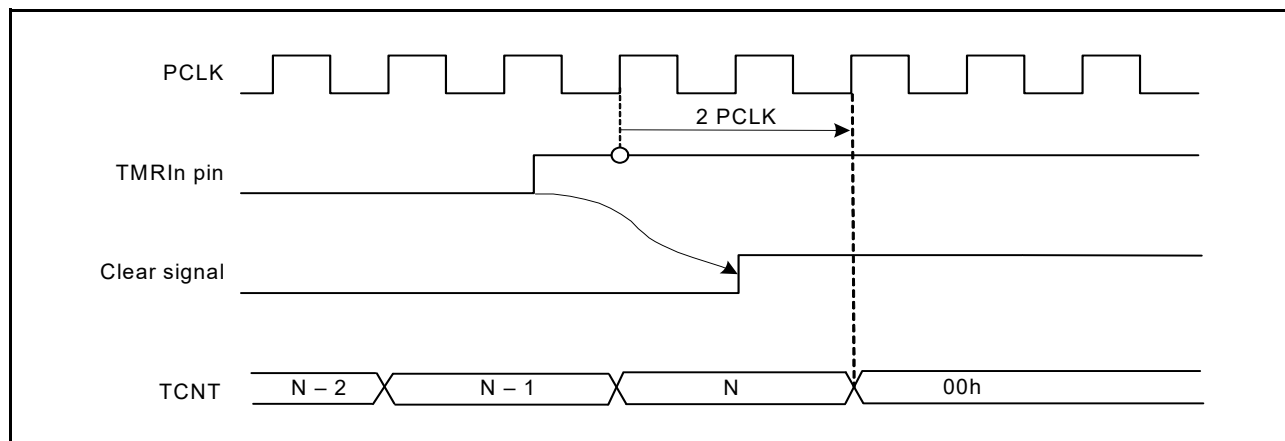


Figure 23.11 Clear Timing by External Counter Reset Signal (High Level)

23.4.6 Timing of Interrupt Signal Output on an Overflow

When TCNT overflows (changes from FFh to 00h), an overflow interrupt signal is output if this interrupt request is enabled.

Figure 23.12 shows the timing of output of the interrupt signal.

For the corresponding interrupt vector number, refer to section 14, Interrupt Controller (ICUb) and Table 23.6.

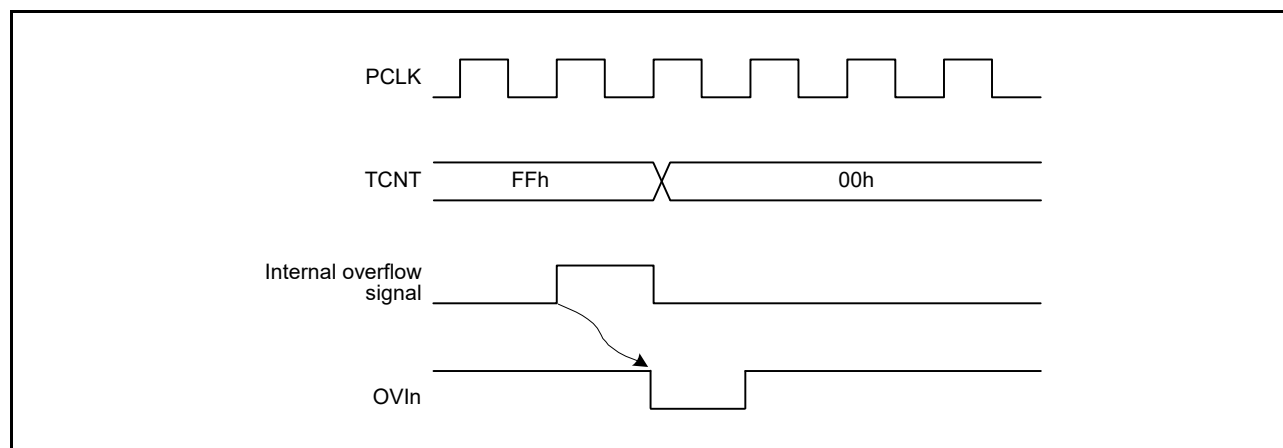


Figure 23.12 Timing of Overflow Interrupt Flag Setting to 1 (n = 0 to 3)

23.5 Operation with Cascaded Connection

If the CSS[1:0] bits in either TMR0.TCCR or TMR1.TCCR are set to 11b, the TMR of the two channels are cascaded. With this configuration, a single 16-bit timer could be used (16-bit counter mode) or compare matches of TMR0 could be counted by TMR1 (compare match count mode).

This section describes unit 0. The operation of unit 1 with cascaded connection is the same as unit 0.

23.5.1 16-Bit Count Mode

When the TMR0.TCCR.CSS[1:0] bits are set to 11b, the timer functions as a single 16-bit timer with TMR0 occupying the upper 8 bits and TMR1 occupying the lower 8 bits.

(1) Counter Clear Specification

- The settings of the TMR0.TCR.CCLR[1:0] bits become effective for the 16-bit counter. If the TMR0.TCR.CCLR[1:0] bits have been set for counter clear at compare match, the 16-bit counter (TMR0.TCNT and TMR1.TCNT together) is cleared when a 16-bit compare match event occurs. The 16-bit counter (TMR0.TCNT and TMR1.TCNT together) is cleared even if counter clear by the TMRI0 pin has been set.
- The settings of the TMR1.TCR.CCLR[1:0] bits are ignored.

(2) Pin Output

- Control of output from the TMO0 pin by the TMR0.TCSR.OSA[1:0] and OSB[1:0] bits is in accordance with the 16-bit compare match conditions.
- Control of output from the TMO1 pin by the TMR1.TCSR.OSA[1:0] and OSB[1:0] bits is in accordance with the lower 8-bit compare match conditions.

23.5.2 Compare Match Count Mode

When the TMR1.TCCR.CSS[1:0] bits are set to 11b, TMR1.TCNT counts the number of occurrences of compare match A for TMR0. TMR0 and TMR1 are controlled independently. Conditions such as generation of interrupts, output from the TMO_n pin (n = 0, 1), and counter clear are in accordance with the settings for each channel.

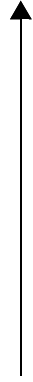
23.6 Interrupt Sources

23.6.1 Interrupt Sources and DTC Activation

There are three interrupt sources for TMRn: CMIA_n, CMIB_n, and OVIn. Their interrupt sources and priorities are listed in Table 23.6.

It is also possible to activate the DTC by means of CMIA_n and CMIB_n interrupts.

Table 23.6 TMR Interrupt Sources

Name	Interrupt Sources	DTC Activation	Priority
CMIA0	TMR0.TCORA compare match	Possible	 High Low
CMIB0	TMR0.TCORB compare match	Possible	
OVI0	TMR0.TCNT overflow	Not possible	
CMIA1	TMR1.TCORA compare match	Possible	
CMIB1	TMR1.TCORB compare match	Possible	
OVI1	TMR1.TCNT overflow	Not possible	
CMIA2	TMR2.TCORA compare match	Possible	
CMIB2	TMR2.TCORB compare match	Possible	
OVI2	TMR2.TCNT overflow	Not possible	
CMIA3	TMR3.TCORA compare match	Possible	
CMIB3	TMR3.TCORB compare match	Possible	
OVI3	TMR3.TCNT overflow	Not possible	

23.6.2 Startup of the A/D Converter

The compare match A of TMR0 and TMR2 allows the A/D converter to be started.

An A/D conversion start request is issued to the A/D converter in response to a generation of compare match A when the TMR_n.TCSR.ADTE bit is 1 (i.e., when an A/D conversion request in response to compare match A is enabled). In this case, the conversion trigger for the 8-bit timer should be selected in the A/D converter to start A/D conversion.

Table 23.7 Startup of A/D Converter

A/D Converter	TMR Unit No.	Target	A/D Conversion Start Request
S12AD, S12AD1 (12-bit A/D converter)	0	Compare match between TMR0.TCORA and TMR0.TCNT	TMTRG0AN_0
	1	Compare match between TMR2.TCORA and TMR2.TCNT	TMTRG0AN_1

23.7 Usage Notes

23.7.1 Module Stop State Setting

Operation of the TMR can be disabled or enabled by using the module stop control registers. The initial setting is for halting of TMR operation. Register access becomes possible after release from the module stop state. For details, refer to section 11, Low Power Consumption.

23.7.2 Notes on Setting Cycle

If the compare match is selected for counter clear, TCNT is cleared at the last PCLK in the cycle in which the value of TCNT matches with that of TCORA or TCORB. TCNT updates the counter value at this last state. Therefore, the counter frequency is obtained by the following formula (f: Counter frequency, PCLK: Operating frequency, N: TCORA and TCORB register setting value).

$$f = \text{PCLK} / (N + 1)$$

23.7.3 Conflict between TCNT Write and Counter Clear

If a counter clear signal is generated concurrently with CPU write to TCNT, the clear takes priority and the write is not performed as shown in Figure 23.13.

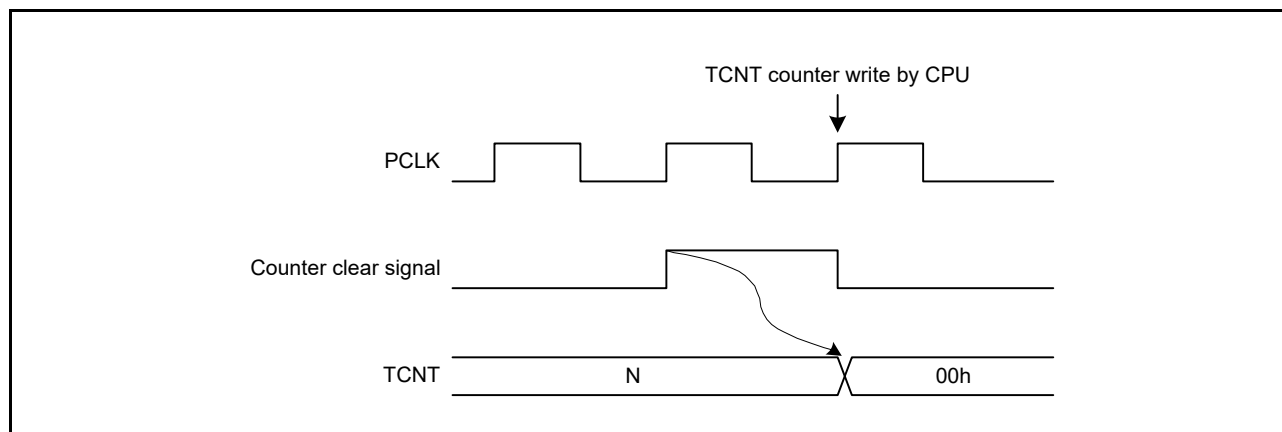


Figure 23.13 Conflict between TCNT Write and Counter Clear

23.7.4 Conflict between TCNT Write and Increment

Even if a counting-up signal is generated concurrently with CPU write to TCNT, the counting-up is not performed and the write takes priority as shown in Figure 23.14.

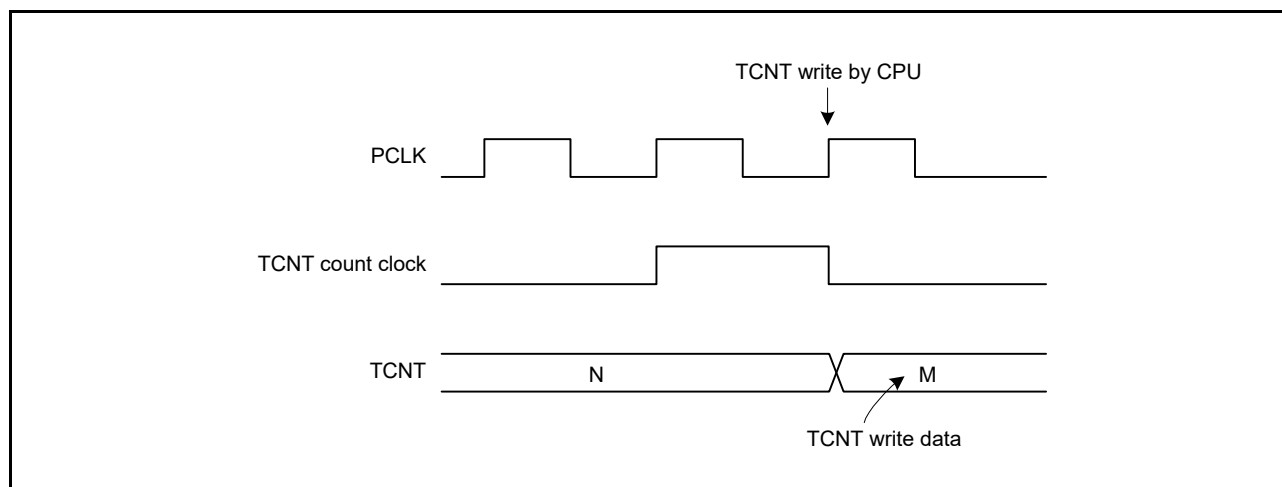


Figure 23.14 Conflict between TCNT Write and Increment

23.7.5 Conflict between TCORA or TCORB Write and Compare Match

Even if a compare match signal is generated simultaneously with CPU write to TCORA or TCORB as shown in Figure 23.15, the write takes priority and the compare match signal does not reach High level.

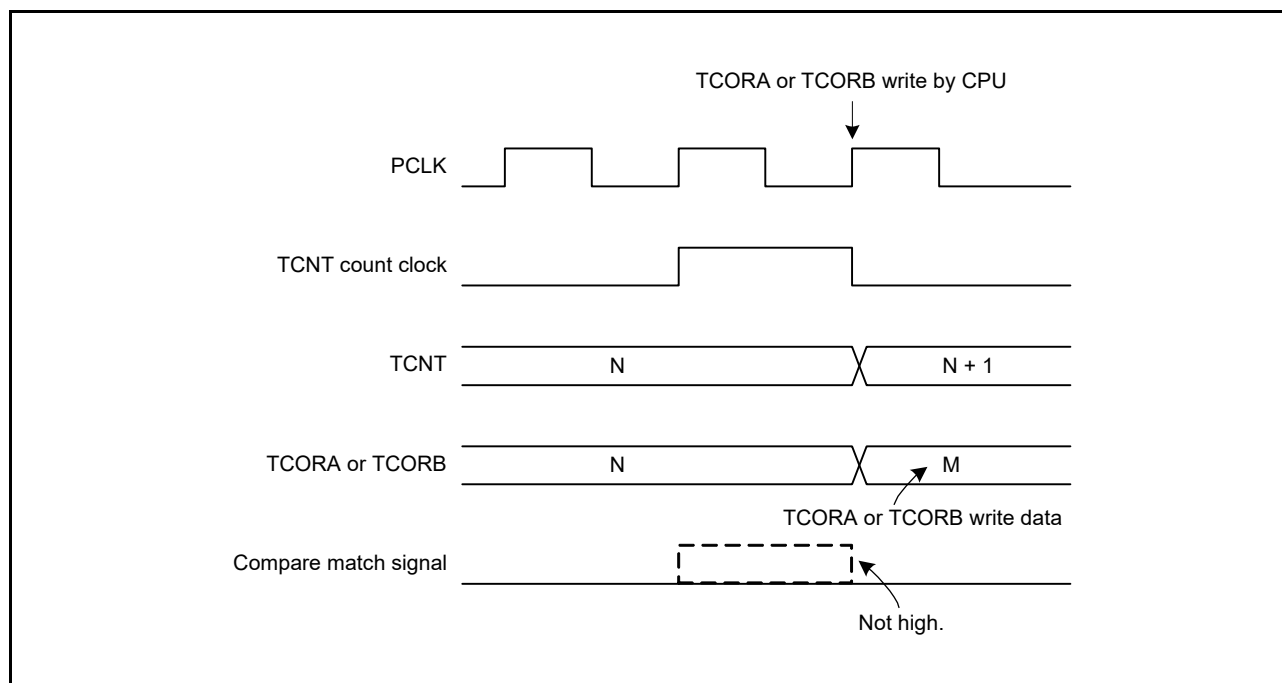


Figure 23.15 Conflict between TCORA or TCORB Write and Compare Match

23.7.6 Conflict between Compare Matches A and B

If compare match events A and B occur at the same time, the 8-bit timer operates in accordance with the priorities for the output methods high for compare match A and compare match B, as listed in Table 23.8.

Table 23.8 Timer Output Priorities

Output Setting	Priority
Toggle output	High
High output	↑
Low output	↑
No change	Low

23.7.7 Switching of Internal Clocks and TCNT Operation

TCNT may be incremented erroneously depending on when the internal clock is switched. Table 23.9 lists the relationship between the timing at which the internal clock is switched (by writing to the TCCR.CKS[2:0] bits) and the operation of TCNT.

When TCNT count clock is generated from an internal clock, the rising edge of the internal clock pulse are always monitored. If the signal levels of the clocks before and after switching change from low to high as shown in No. 2 in Table 23.9, the change is considered as an edge. Therefore, a TCNT count clock is generated and TCNT is incremented. The erroneous increment of TCNT can also happen when switching between internal and internal clocks.

Table 23.9 Switching of Internal Clocks and TCNT Operation (1/2)

No.	Timing to Change the TCCR.CKS[2:0] Bits	TCNT Counter Operation
1	Switching from low to low*1	TCNT TCCR.CKS[2:0] bits changed
2	Switching from low to high*2	TCNT TCCR.CKS[2:0] bits changed

Table 23.9 Switching of Internal Clocks and TCNT Operation (2/2)

No.	Timing to Change the TCCR.CKS[2:0] Bits	TCNT Counter Operation
3	Switching from high to low*4	TCNT TCCR.CKS[2:0] bits changed
4	Switching from high to high	TCNT TCCR.CKS[2:0] bits changed

Note 1. Includes switching from low to stop, and from stop to low.

Note 2. Includes switching from stop to high.

Note 3. Generated because the change of the signal levels is considered as an edge; TCNT counter is incremented.

Note 4. Includes switching from high to stop.

23.7.8 Clock Source Setting with Cascaded Connection

If 16-bit counter mode and compare match count mode are specified at the same time, count clocks for TMR0.TCNT and TMR1.TCNT (TMR2.TCNT and TMR3.TCNT) are not generated, and the counter stops. Do not specify 16-bit counter mode and compare match count mode simultaneously.

23.7.9 Continuous Output of Compare Match Interrupt Signal

When TCORA or TCORB is set to 00h, PCLK/1 is set as the internal clock, and compare match is set as the counter clear source, the TCNT counter remains 00h and is not updated, and a compare match interrupt signal is output continuously to form a flat signal level.

At this time, the interrupt controller cannot detect the second and subsequent interrupts.

Figure 23.16 shows operation timing when the compare match interrupt signal is continuously output.

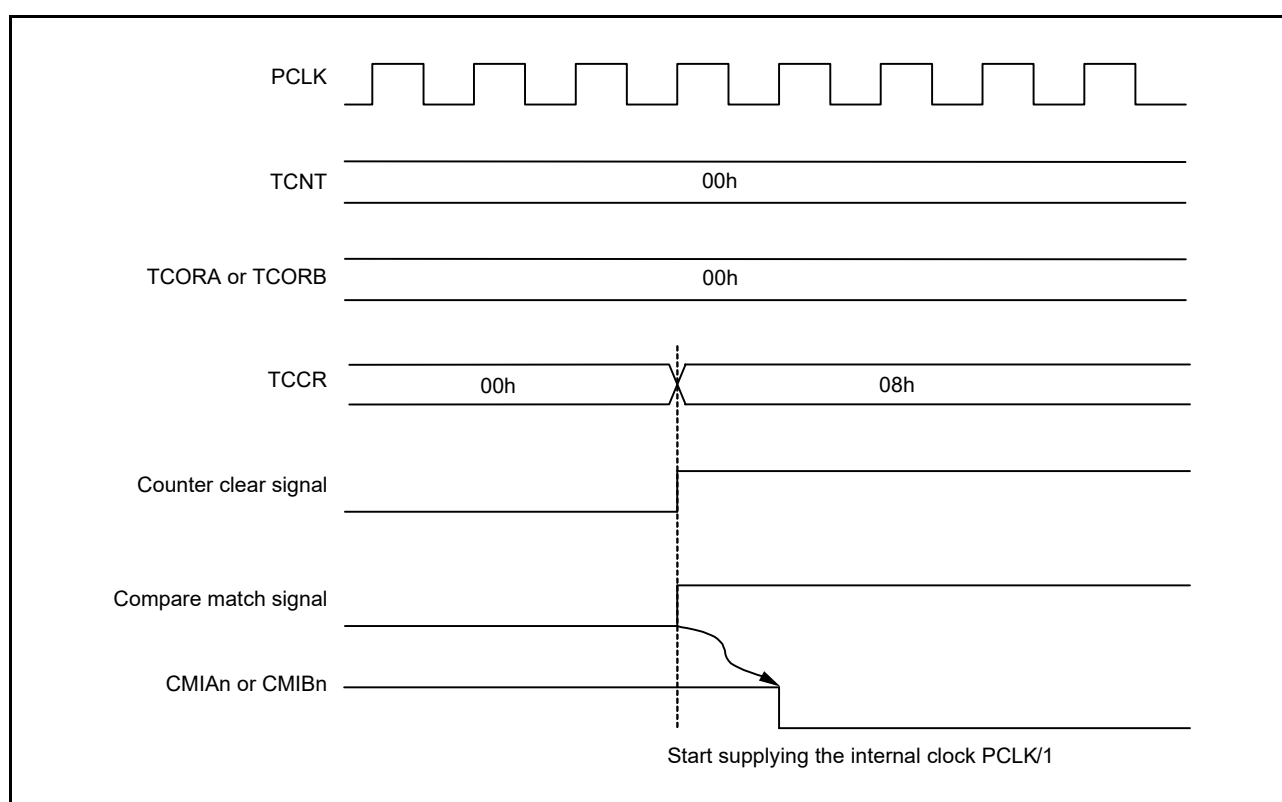


Figure 23.16 Continuous Output of Compare Match Interrupt Signal (n = 0 to 3)

24. Compare Match Timer (CMT)

This MCU has an on-chip compare match timer (CMT) unit (unit 0) consisting of a two-channel 16-bit timer (i.e., a total of two channels). The CMT has a 16-bit counter, and can generate interrupts at set intervals.

In this section, “PCLK” is used to refer to PCLKB.

24.1 Overview

Table 24.1 lists the specifications for the CMT.

Figure 24.1 shows a block diagram of the CMT (unit 0). A two-channel CMT constitutes a unit.

Table 24.1 CMT Specifications

Item	Description
Count clocks	- Four frequency dividing clocks - One clock from PCLK/8, PCLK/32, PCLK/128, and PCLK/512 can be selected for each channel.
Interrupt	A compare match interrupt can be requested for each channel.
Low power consumption function	Module stop state can be set.

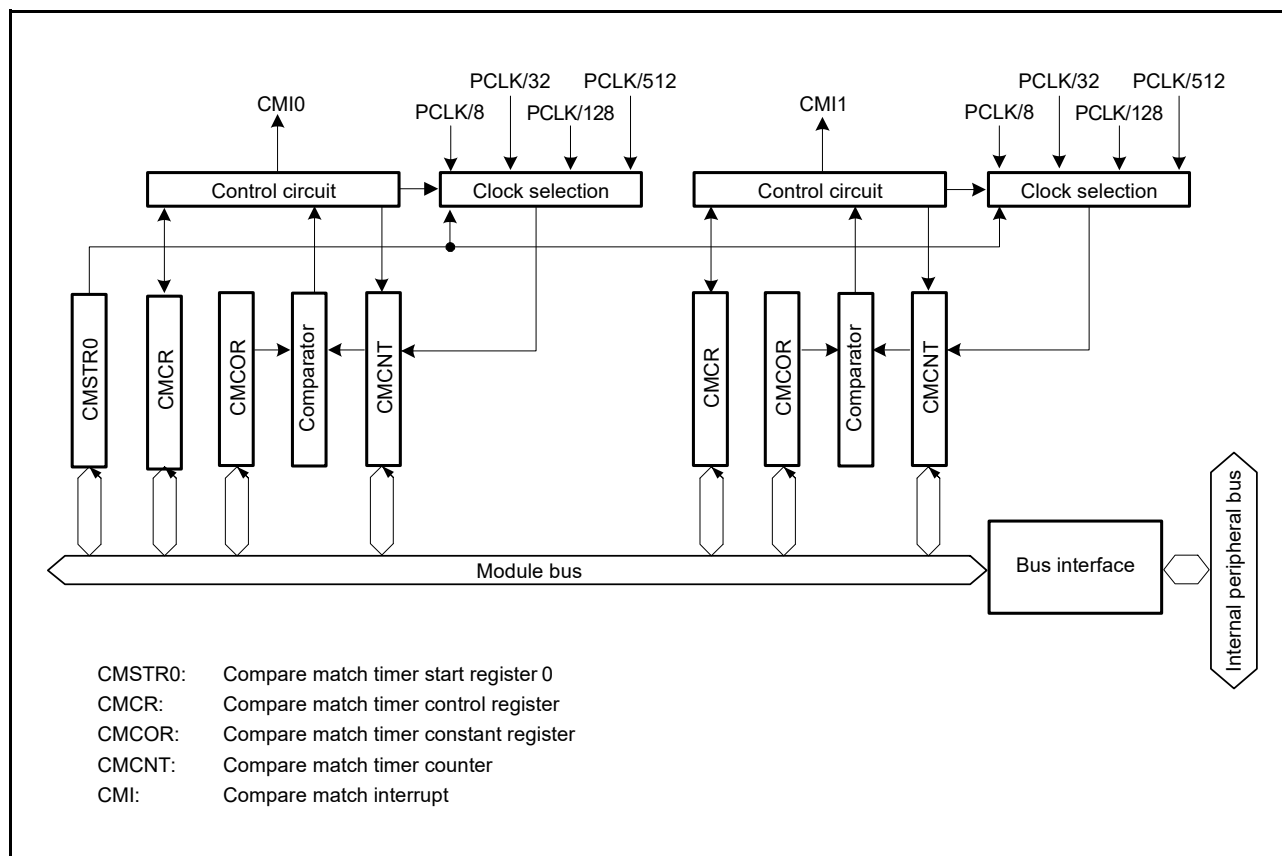


Figure 24.1 CMT (Unit 0) Block Diagram

24.2 Register Descriptions

24.2.1 Compare Match Timer Start Register 0 (CMSTR0)

Address(es): CMT.CMSTR0 0008 8000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR1	STR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR0	Count Start 0	0: CMT0.CMCNT count is stopped. 1: CMT0.CMCNT count is started.	R/W
b1	STR1	Count Start 1	0: CMT1.CMCNT count is stopped. 1: CMT1.CMCNT count is started.	R/W
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

24.2.2 Compare Match Timer Control Register (CMCR)

Address(es): CMT0.CMCR 0008 8002h, CMT1.CMCR 0008 8008h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	CMIE	—	—	—	—	CKS[1:0]	—
Value after reset:	0	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: PCLK/8 0 1: PCLK/32 1 0: PCLK/128 1 1: PCLK/512	R/W
b5 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	CMIE	Compare Match Interrupt Enable	0: Compare match interrupt (CMIn) disabled 1: Compare match interrupt (CMIn) enabled	R/W
b7	—	Reserved	This bit is read as undefined. The write value should be 1.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CKS[1:0] Bits (Clock Select)

These bits select the count source from four frequency dividing clocks obtained by dividing the peripheral module clock (PCLK).

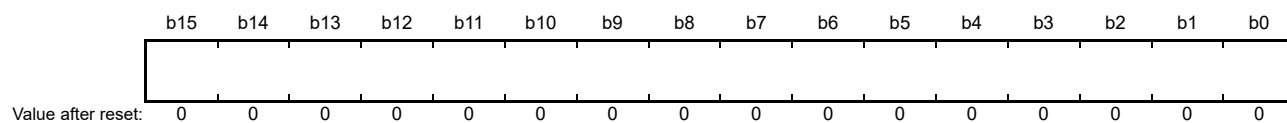
When the CMSTR0.STRn (n = 0, 1) bit is set to 1, the CMCNT counter starts counting up on the clock selected with the CKS[1:0] bits.

CMIE Bit (Compare Match Interrupt Enable)

The CMIE bit enables or disables compare match interrupt (CMIn) (n = 0, 1) generation when the CMCNT counter and the CMCOR register values match.

24.2.3 Compare Match Counter (CMCNT)

Address(es): CMT0.CMCNT 0008 8004h, CMT1.CMCNT 0008 800Ah



The CMCNT counter is a readable/writable up-counter.

When an frequency dividing clock is selected by the CMCR.CKS[1:0] bits and the CMSTR0.STRn (n = 0, 1) bit is set to 1, the CMCNT counter starts counting up using the selected clock.

When the value in the CMCNT counter and the value in the CMCOR register match, the CMCNT counter is set to 0000h. At the same time, a compare match interrupt (CMIn) (n = 0, 1) is generated.

24.2.4 Compare Match Constant Register (CMCOR)

Address(es): CMT0.CMCOR 0008 8006h, CMT1.CMCOR 0008 800Ch



The CMCOR register is a readable/writable register to set a value for compare match with the CMCNT counter.

24.3 Operation

24.3.1 Cyclic Count Operation

When an frequency dividing clock is selected by the CMCR.CKS[1:0] bits and the CMSTR0.STRn (n = 0, 1) bit is set to 1, the CMCNT counter starts counting up using the selected clock.

When the value in the counter and the value in the register match, a compare match interrupt (CMIn) (n = 0,1) is generated. The CMCNT counter then starts counting up again from 0000h. Figure 24.2 shows the operation of the CMCNT counter.

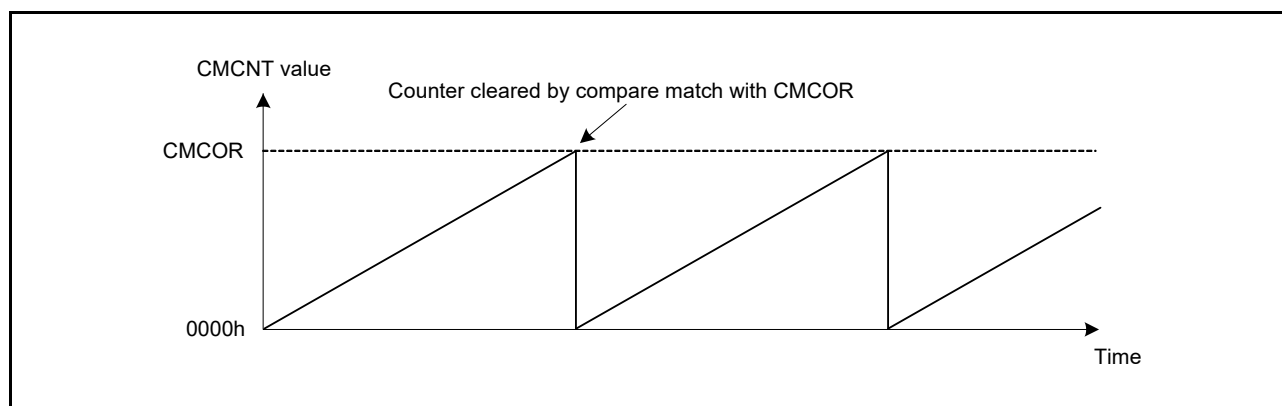


Figure 24.2 CMCNT Counter Operation

24.3.2 CMCNT Count Timing

As the count clock to be input to the CMCNT counter, one of four frequency dividing clocks (PCLK/8, PCLK/32, PCLK/128, and PCLK/512) obtained by dividing the peripheral module clock (PCLK) can be selected with the CMCR.CKS[1:0] bits. Figure 24.3 shows the timing of the CMCNT counter.

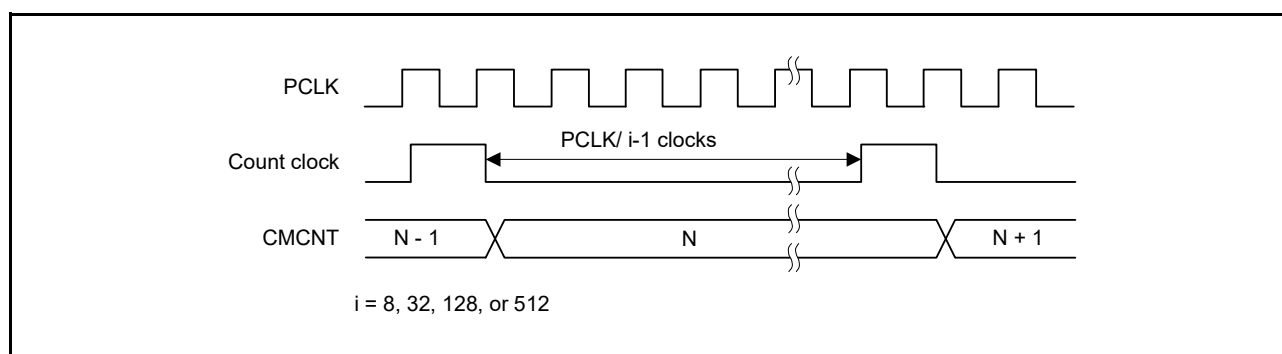


Figure 24.3 CMCNT Count Timing

24.4 Interrupts

24.4.1 Interrupt Sources

The CMT has channels and each of them to which a different vector address is allocated has a compare match interrupt (CMIn) ($n = 0, 1$). When a compare match interrupt occurs, the corresponding interrupt request is output.

When the interrupt request is used to generate a CPU interrupt, the priority of channels can be changed by the interrupt controller settings. For details, see section 14, Interrupt Controller (ICUb).

Table 24.2 CMT Interrupt Sources

Name	Interrupt Sources	DTC Activation
CMIO	Compare match in CMT0	Possible
CM11	Compare match in CMT1	Possible

24.4.2 Timing of Compare Match Interrupt Generation

When the CMCNT counter and the CMCOR register match, a compare match interrupt (CMIn) ($n = 0, 1$) is generated. A compare match signal is generated at the last state in which the values match (the timing when the CMCNT counter updates the matched count value). That is, after a match between the CMCOR register and the CMCNT counter, the compare match signal is not generated until the next the CMCNT counter input clock.

Figure 24.4 shows the timing of a compare match interrupt.

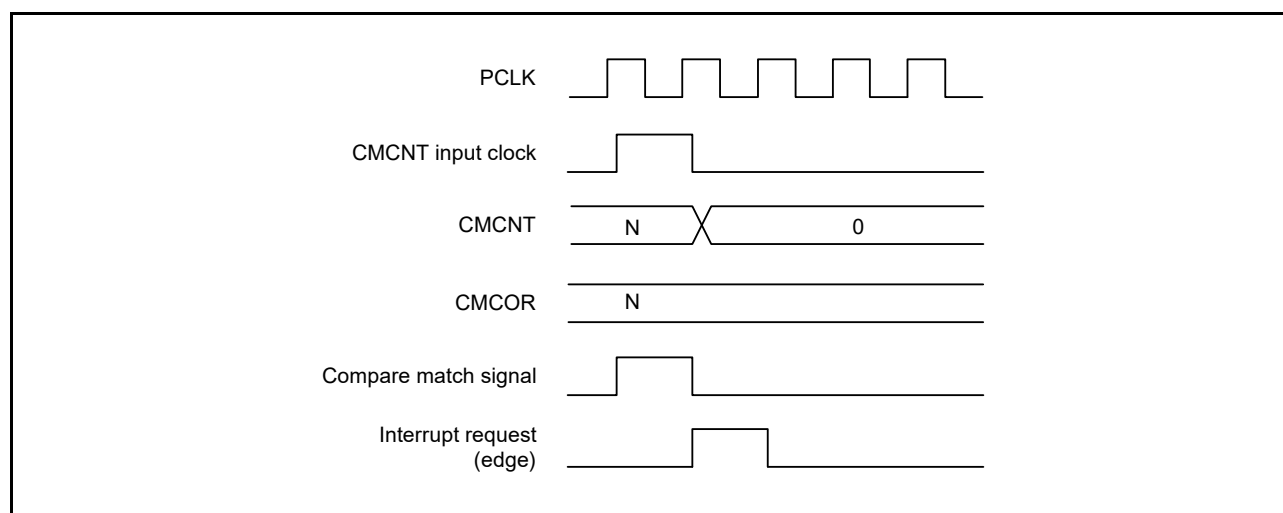


Figure 24.4 Timing of a Compare Match Interrupt

24.5 Usage Notes

24.5.1 Setting the Module Stop Function

The CMT can be enabled or disabled using the module stop control register. After a reset, the CMT is in the module stop state. The registers can be accessed by releasing the module stop state. For details, refer to [section 11, Low Power Consumption](#).

24.5.2 Conflict between CMCNT Counter Writing and Compare Match

When the compare match signal is generated while writing to the CMCNT counter, clearing the CMCNT counter has priority over writing to it. In this case, the CMCNT counter is not written to. [Figure 24.5](#) shows the timing to clear the CMCNT counter.

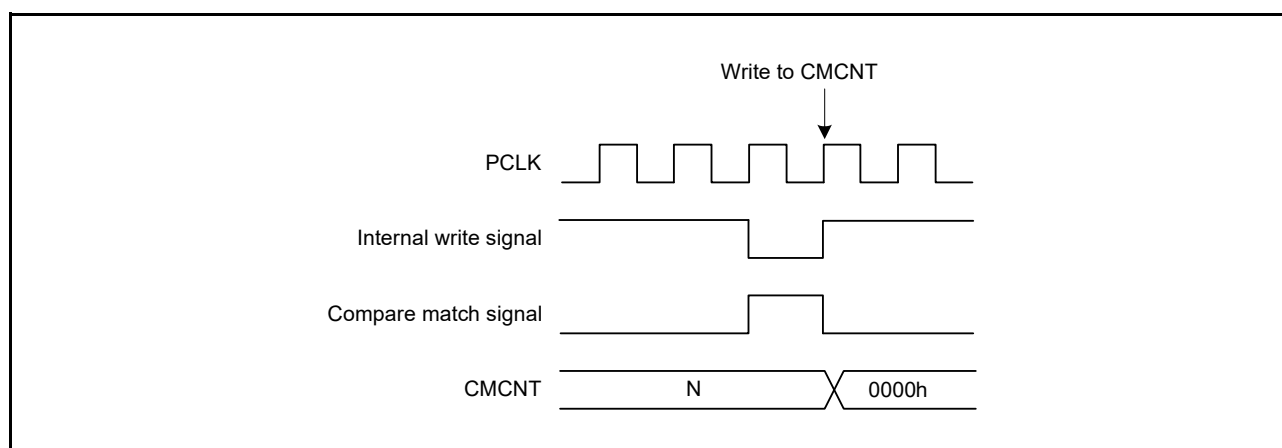


Figure 24.5 Conflict between CMCNT Counter Writing and Compare Match

24.5.3 Conflict between CMCNT Counter Writing and Incrementing

If writing to the counter and the incrementing conflict, the writing has priority over the incrementing. [Figure 24.6](#) shows the timing to write the CMCNT counter.

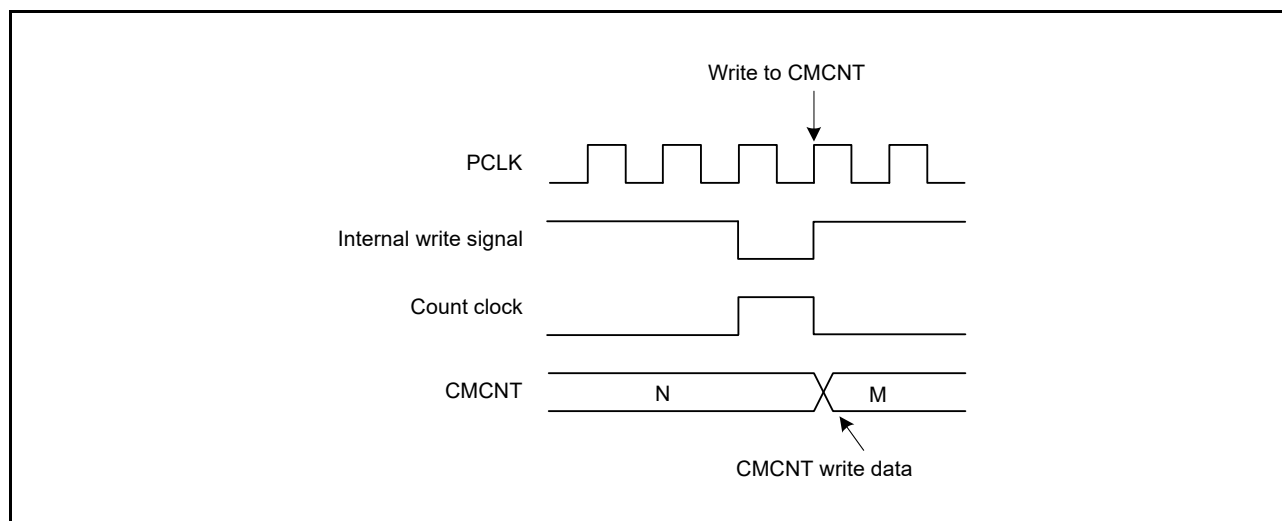


Figure 24.6 Conflict between CMCNT Counter Writing and Incrementing

25. Independent Watchdog Timer (IWDTa)

The independent watchdog timer (IWDT) can be used to detect programs being out of control.

The user can detect when a program runs out of control if an underflow occurs, by creating a program that refreshes the IWDT counter before it underflows.

The functions of the IWDT are different from those of the WDT in the following respects.

- The divided IWDT-dedicated clock (IWDTCLK) is used as the count source (not affected by the peripheral module clock).
- The IWDTCSLTPR.SLCSTP bit or the OFS0.IWDTSLCSTP bit can be used to select not to stop the counter in low power consumption mode.

25.1 Overview

Table 25.1 lists the specifications of the IWDT.

Table 25.1 IWDT Specifications

Item	Description
Count source*1	IWDT-dedicated clock (IWDTCLK)
Clock divide ratio	Divide by 1, 16, 32, 64, 128, or 256
Counter operation	Counting down using a 14-bit counter
Conditions for starting the counter	• Auto-start mode: Counting automatically starts after a reset is released • Register start mode: Counting is started by refresh operation (writing 00h and then FFh to the IWDTRR register).
Conditions for stopping the counter	• Reset • In low power consumption modes (depends on the register setting*2) • A counter underflows or a refresh error occurs (only in register start mode)
Window function	Window start and end positions can be specified (refresh-permitted and refresh-prohibited periods)
Reset output sources	• Counter underflows • Refreshing outside the refresh-permitted period (refresh error)
Non-maskable interrupt sources	• Counter underflows • Refreshing outside the refresh-permitted period (refresh error)
Reading the counter value	The counter value can be read by the IWDTSR register.
Auto-start mode (controlled by option function select register 0 (OFS0))	• Selecting the clock frequency divide ratio after a reset (OFS0.IWDTCKS[3:0] bits) • Selecting the timeout period of the independent watchdog timer (OFS0.IWDTTOPS[1:0] bits) • Selecting the window start position in the independent watchdog timer (OFS0.IWDRPSS[1:0] bits) • Selecting the window end position in the independent watchdog timer (OFS0.IWDRPES[1:0] bits) • Selecting the reset output or interrupt request output (OFS0.IWDRSTIRQS bit) • Selecting the counter operation/stop in low power consumption mode (OFS0.IWDTSLCSTP bit)
Register start mode (controlled by the IWDT registers)	• Selecting the clock frequency divide ratio after refreshing (IWDTCR.CKS[3:0] bits) • Selecting the timeout period of the independent watchdog timer (IWDTCR.TOPS[1:0] bits) • Selecting the window start position in the independent watchdog timer (IWDTCR.RPSS[1:0] bits) • Selecting the window end position in the independent watchdog timer (IWDTCR.RPES[1:0] bits) • Selecting the reset output or interrupt request output (IWDTCCR.RSTIRQS bit) • Selecting the counter operation/stop in low power consumption mode (IWDTCSLTPR.SLCSTP bit)

Note 1. Satisfy the frequency of the peripheral module clock (PCLKB) $\geq 4 \times$ (the frequency of the count source after divide).

Note 2. When the OFS0.IWDTSLCSTP bit is 1 in auto-start mode, and when the IWDTCSLTPR.SLCSTP bit is 1 in register start mode.

Figure 25.1 is a block diagram of the IWDT.

In IWDT, the bus interface and registers operate with the peripheral module clock (PCLKB), while the 14-bit down-counter and control circuits operate with IWDTCLK.

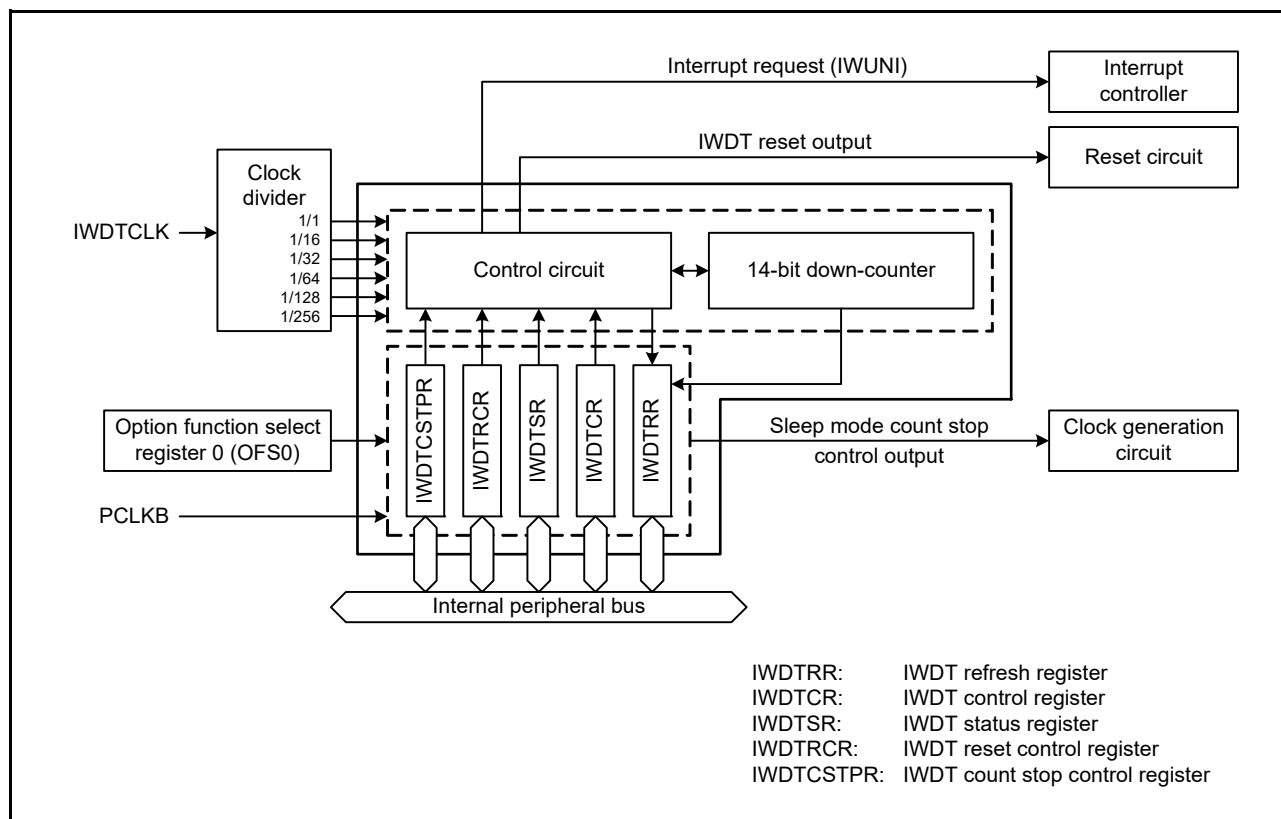


Figure 25.1 IWDT Block Diagram

25.2 Register Descriptions

25.2.1 IWDT Refresh Register (IWDTRR)

Address(es): IWDT.IWDTRR 0008 8030h



Bit	Description	R/W
b7 to b0	The counter is refreshed by writing 00h and then writing FFh to this register.	R/W

The IWDTRR register is used to refresh the counter of the IWDT.

The counter of the IWDT is refreshed by writing 00h and then writing FFh to the IWDTRR register (refresh operation) within the refresh-permitted period.

After the counter has been refreshed, it starts counting down from the value selected by the IWDTTOPS[1:0] bits in option function select register 0 (OFS0) in auto-start mode. In register start mode, counting down starts from the value selected by setting the IWDTCR.TOPS[1:0] bits in the first refresh operation after a reset is released.

When 00h is written, the read value is 00h. When a value other than 00h is written, the read value is FFh.

For details of the refresh operation, refer to [section 25.3.2, Refresh Operation](#).

25.2.2 IWDt Control Register (IWDTCR)

Address(es): IWDt.IWDTCR 0008 8032h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	RPSS[1:0]	—	—	RPES[1:0]			CKS[3:0]				—	—	TOPS[1:0]	
Value after reset:	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TOPS[1:0]	Timeout Period Select	b1 b0 0 0: 128 cycles (007Fh) 0 1: 512 cycles (01FFh) 1 0: 1024 cycles (03FFh) 1 1: 2048 cycles (07FFh)	R/W
b3, b2	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R
b7 to b4	CKS[3:0]	Clock Divide Ratio Select	b7 b4 0 0 0 0: No division 0 0 1 0: Divide-by-16 0 0 1 1: Divide-by-32 0 1 0 0: Divide-by-64 1 1 1 1: Divide-by-128 0 1 0 1: Divide-by-256 Settings other than above are prohibited.	R/W
b9, b8	RPES[1:0]	Window End Position Select	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (window end position is not specified.)	R/W
b11, b10	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R
b13, b12	RPSS[1:0]	Window Start Position Select	b13 b12 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (window start position is not specified.)	R/W
b15, b14	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R

Note: This register can be written only once after a reset, and cannot be written after the counter starts counting.

This register is used to specify the time until a timeout occurs and the refresh-permitted period when using register start mode.

In auto-start mode, the settings in the IWDTCR register are disabled, and the settings in option function select register 0 (OFS0) are enabled. The bit setting made to the IWDTCR register can also be made in the OFS0 register. For details, refer to section 25.3.7, Correspondence between Option Function Select Register 0 (OFS0) and IWDt Registers.

TOPS[1:0] Bits (Timeout Period Select)

These bits select the timeout period (period until the counter underflows) from among 128, 512, 1024, or 2048 cycles, taking the divided clock specified by the CKS[3:0] bits as one cycle.

After the counter is refreshed, the combination of the CKS[3:0] and TOPS[1:0] bits determines the time (number of IWDTCCLK cycles) until the counter underflows.

Relations between the CKS[3:0] and TOPS[1:0] bit setting, the timeout period, and the number of IWDTCCLK cycles are listed in Table 25.2.

Table 25.2 Settings and Timeout Periods

CKS[3:0] Bits				TOPS[1:0] Bits		Clock Divide Ratio	Timeout Period (Number of Cycles)	Cycles of IWDTCCLK
b7	b6	b5	b4	b1	b0			
0	0	0	0	0	0	No division	128	128
				0	1		512	512
				1	0		1024	1024
				1	1		2048	2048
0	0	1	0	0	0	Divide-by-16	128	2048
				0	1		512	8192
				1	0		1024	16384
				1	1		2048	32768
0	0	1	1	0	0	Divide-by-32	128	4096
				0	1		512	16384
				1	0		1024	32768
				1	1		2048	65536
0	1	0	0	0	0	Divide-by-64	128	8192
				0	1		512	32768
				1	0		1024	65536
				1	1		2048	131072
1	1	1	1	0	0	Divide-by-128	128	16384
				0	1		512	65536
				1	0		1024	131072
				1	1		2048	262144
0	1	0	1	0	0	Divide-by-256	128	32768
				0	1		512	131072
				1	0		1024	262144
				1	1		2048	524288

CKS[3:0] Bits (Clock Divide Ratio Select)

These bits select the IWDTCCLK divide ratio from among divide-by 1, 16, 32, 64, 128, and 256. Combination with the TOPS[1:0] bit setting, a count period between 128 and 524288 cycles of the IWDTCCLK can be selected for the IWDTC.

RPES[1:0] Bits (Window End Position Select)

These bits select 75%, 50%, 25% or 0% of the count period for the window end position of the counter. The window end position should be a value smaller than the window start position (window start position > window end position). If the window end position is greater than the window start position, only the window start position setting is enabled.

The counter values for the window start and end positions selected by setting the RPSS[1:0] and RPES[1:0] bits change depending on the TOPS[1:0] bit setting.

Table 25.3 lists the counter values for the window start and end positions corresponding to TOPS[1:0] bit values.

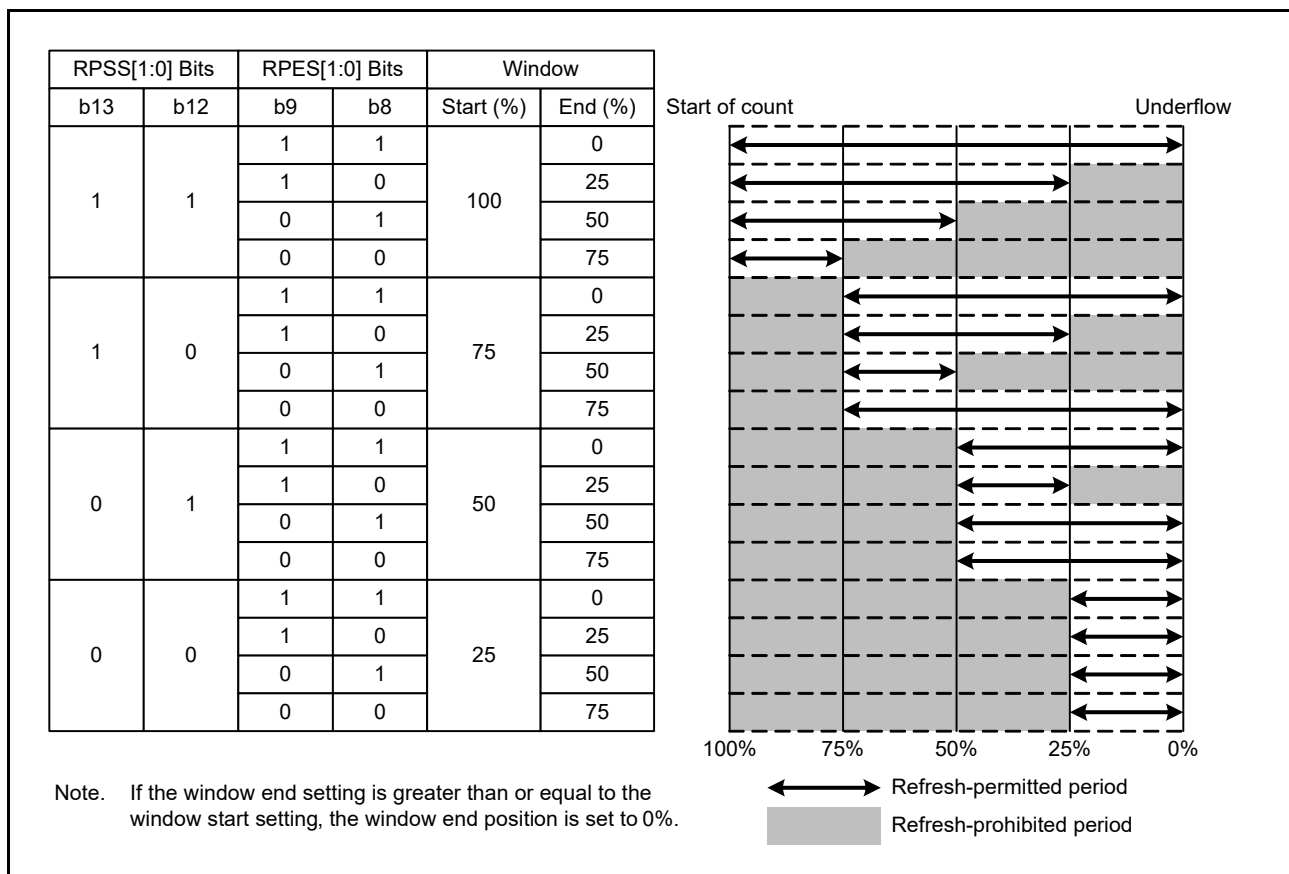
Table 25.3 Relationship between Timeout Period and Window Start and End Counter Values

TOPS[1:0] Bits		Timeout Period		Window Start and End Counter Value			
b1	b0	Cycles	Counter Value	100%	75%	50%	25%
0	0	128	007Fh	007Fh	005Fh	003Fh	001Fh
0	1	512	01FFh	01FFh	017Fh	00FFh	007Fh
1	0	1024	03FFh	03FFh	02FFh	01FFh	00FFh
1	1	2048	07FFh	07FFh	05FFh	03FFh	01FFh

RPSS[1:0] Bits (Window Start Position Select)

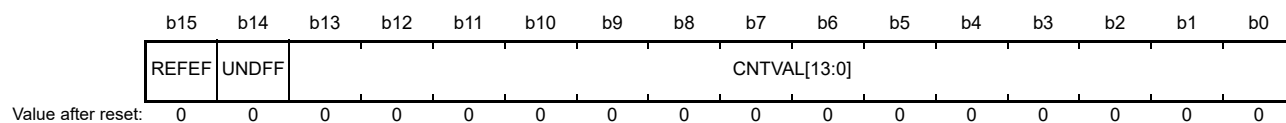
These bits select a counter window start position from 100%, 75%, 50%, or 25% of the count period (100% when the count starts and 0% when the counter underflows). The interval between the window start position and window end position is the refresh-permitted period and the other periods are refresh-prohibited periods.

Figure 25.2 shows the relationship between of the RPSS[1:0] and RPES[1:0] bit setting and the refresh-permitted and refresh-prohibited periods.

**Figure 25.2 RPSS[1:0] and RPES[1:0] Bit Settings and the Refresh-Permitted Period**

25.2.3 IWDT Status Register (IWDTSR)

Address(es): IWDT.IWDTSR 0008 8034h



Bit	Symbol	Bit Name	Description	R/W
b13 to b0	CNTVAL[13:0]	Counter Value	Value counted by the counter	R
b14	UNDF	Underflow Flag	0: No underflow occurred 1: Underflow occurred	R/(W) *1
b15	REFEF	Refresh Error Flag	0: No refresh error occurred 1: Refresh error occurred	R/(W) *1

Note 1. Only 0 can be written to clear the flag.

CNTVAL[13:0] Bits (Counter Value)

These bits are used to confirm the value of the counter, but note that the read value may differ from the actual count by a value of one count.

UNDF Flag (Underflow Flag)

This flag is used to confirm whether or not an underflow has occurred in the counter.

The value 1 indicates that the counter has underflowed. The value 0 indicates that the counter has not underflowed.

Write 0 to the UNDF flag to set the value to 0. Writing 1 has no effect.

REFEF Flag (Refresh Error Flag)

This flag is used to confirm whether or not a refresh error (performing a refresh operation during a refresh-prohibited period) has occurred.

The value 1 indicates that a refresh error has occurred. The value 0 indicates that no refresh error has occurred.

Write 0 to the REFEF flag to set the value to 0. Writing 1 has no effect.

25.2.4 IWDTRCR Reset Control Register (IWDTRCR)

Address(es): IWDTRCR 0008 8036h

	b7	b6	b5	b4	b3	b2	b1	b0
	RSTIR QS	—	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R
b7	RSTIRQS	Reset Interrupt Request Select	0: Non-maskable interrupt request is output. 1: Reset signal is output.	R/W

Note: This register can be written only once after a reset, and cannot be written after the counter starts counting.

This register is used to specify the behavior when a timeout or refresh error occurs when using register start mode.

In auto-start mode, the IWDTRCR register setting is disabled, and the settings in option function select register 0 (OFS0) are enabled. The bit setting made to the IWDTRCR register can also be made in the OFS0 register. For details, refer to section 25.3.7, Correspondence between Option Function Select Register 0 (OFS0) and IWDTR Registers.

25.2.5 IWDTCSTPR Register (IWDTCSTPR)

Address(es): IWDTCSTPR 0008 8038h

b7	b6	b5	b4	b3	b2	b1	b0
SLCSTP	—	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R
b7	SLCSTP	Sleep Mode Count Stop Control	0: Counter continues counting even in low power consumption mode.*1 1: Counter stops in low power consumption mode.	R/W

Note: This register can be written only once after a reset, and cannot be written after the counter starts counting.

Note 1. The counter continues counting in sleep mode, deep sleep mode, or software standby mode.

This register is used to specify whether the counter operates in low power consumption mode when using register start mode.

In auto-start mode, the IWDTCSTPR register setting is disabled, and the settings in option function select register 0 (OFS0) are enabled. The bit setting mode to the IWDTCSTPR register can also be made in the OFS0 register. For details, refer to section 25.3.7, Correspondence between Option Function Select Register 0 (OFS0) and IWDTCSTPR Registers.

SLCSTP Bit (Sleep Mode Count Stop Control)

This bit selects whether the counter continues or stops counting in low power consumption mode.

25.2.6 Option Function Select Register 0 (OFS0)

For option function select register 0 (OFS0), refer to section 25.3.7, Correspondence between Option Function Select Register 0 (OFS0) and IWDTCSTPR Registers.

25.3 Operation

25.3.1 Count Operation in Each Start Mode

Select the IWDt start mode by setting the IWDtSTRT bit in option function select register 0 (OFS0).

When the OFS0.IWDtSTRT bit is 1 (register start mode), the IWDtCR, IWDtRCR, and IWDtCSTPR registers are enabled, and counting is started by refresh operation (writing) to the IWDtRR register. When the OFS0.IWDtSTRT bit is 0 (auto-start mode), the setting of the OFS0 register is enabled, and counting automatically starts after reset.

25.3.1.1 Register Start Mode

When the OFS0.IWDtSTRT bit in option function select register 0 is 1, register start mode is selected, and the IWDtCR, IWDtRCR, and IWDtCSTPR registers are enabled.

After a reset is released, set the clock divide ratio, window start and end positions, and timeout period in the IWDtCR register, the reset output or interrupt request output in the IWDtRCR register, and counter operation/stop in low power consumption mode in the IWDtCSTPR register. Then, refresh the counter to start counting down from the value selected by setting the IWDtCR.TOPS[1:0] bits.

Thereafter, as long as the program continues normal operation and the counter is refreshed in the refresh-permitted period, the value in the counter is re-set each time the counter is refreshed and counting down continues. The IWDt does not output the reset signal as long as this continues. However, if the counter underflows because the counter cannot be refreshed due to a program runaway, or if a refresh error occurs because the counter was refreshed outside the refresh-permitted period, the IWDt outputs a reset signal or an interrupt request (IWUNI). Set the IWDtRCR.RSTIRQS bit to select either reset signal output or interrupt request output.

Figure 25.3 shows an example of operation under the following conditions.

- Register start mode (OFS0.IWDtSTRT = 1)
- Reset output is enabled (IWDtRCR.RSTIRQS = 1)
- The window start position is 75% (IWDtCR.RPSS[1:0] = 10b)
- The window end position is 25% (IWDtCR.RPES[1:0] = 10b)

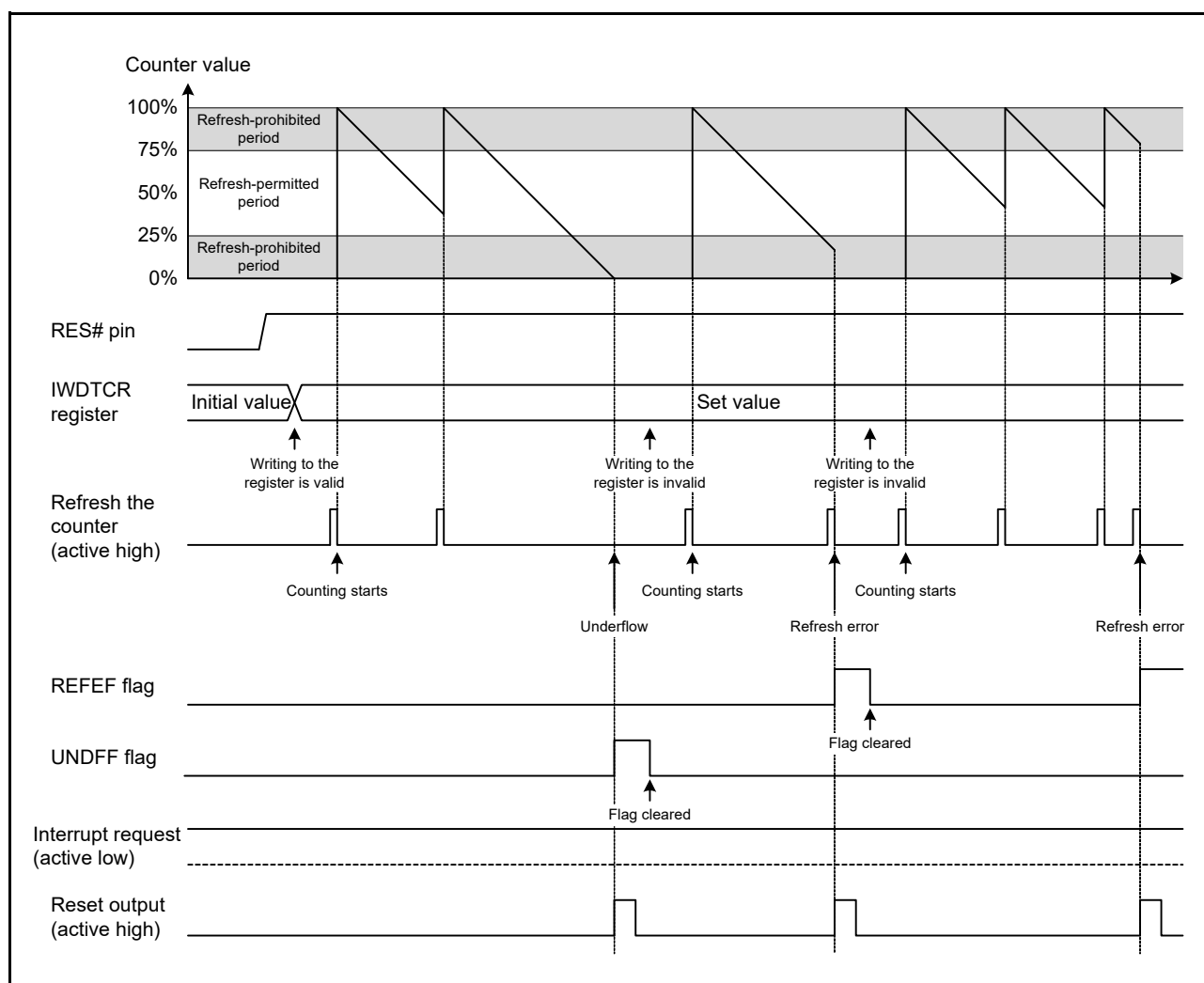


Figure 25.3 Operation Example in Register Start Mode

25.3.1.2 Auto-Start Mode

When the IWDTSTRT bit in option function select register 0 (OFS0) is 0, auto-start mode is selected, and the IWDTCR, IWDTRCR, and IWDTCSTPR registers are disabled.

During the reset state, the clock divide ratio, window start and end positions, timeout period, reset output or interrupt request output, and counter operation/stop in low power consumption mode are set using the values specified in the OFS0 register. When the reset is released, the counter automatically starts counting down from the value selected by the OFS0.IWDTTPOS[1:0] bits.

After that, as long as the program continues normal operation and the counter is refreshed in the refresh-permitted period, the value in the counter is re-set each time the counter is refreshed and counting down continues. The IWDT does not output the reset signal as long as this continues. However, if the counter underflows because refreshing of the counter is not possible due to the program having entered crashed execution or if a refresh error occurs due to refreshing outside the refresh-permitted period, the IWDT outputs the reset signal or an interrupt request (IWUNI). After the reset signal or interrupt request (IWUNI) is generated, the counter reloads the timeout period after counting for one cycle, and restarts counting. Set the OFS0.IWDTRSTIRQS bit to select either reset signal output or interrupt request output.

Figure 25.4 shows an example of operation under the following conditions.

- Auto-start mode (OFS0.IWDTSTRT = 0)
- Interrupt request output is enabled (OFS0.IWDTRSTIRQS = 0)
- The window start position is 75% (OFS0.IWDRPSS[1:0] = 10b)
- The window end position is 25% (OFS0.IWDRPES[1:0] = 10b)

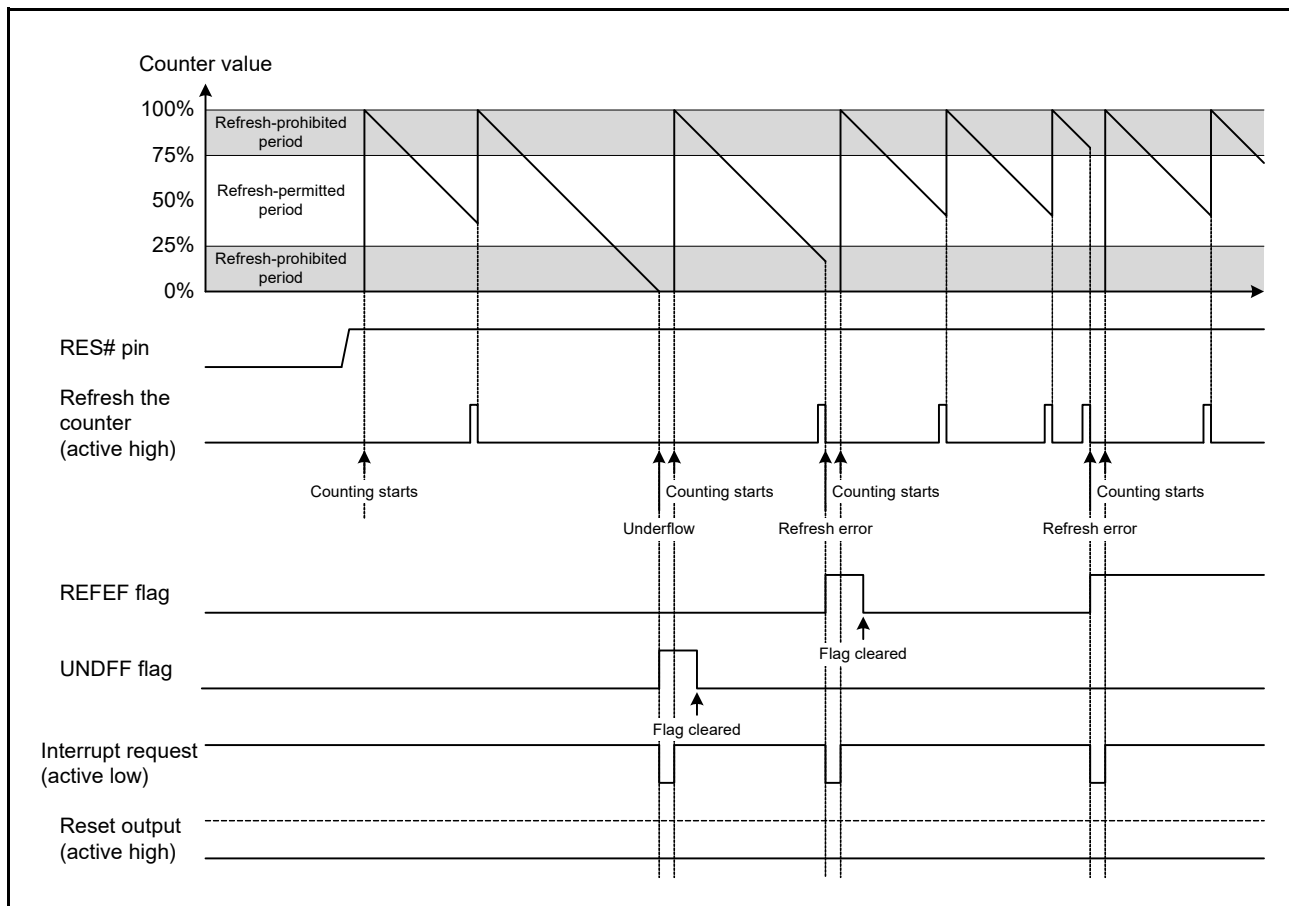


Figure 25.4 Operation Example in Auto-Start Mode

25.3.2 Refresh Operation

The counter is refreshed and starts operation (counting is started by refreshing) by writing the values 00h and then FFh to the IWDTRR register. If a value other than FFh is written after 00h, the counter is not refreshed. After such invalid writing, correct refreshing is performed by again writing 00h and then FFh to the IWDTRR register.

When writing is done in the order of 00h (first time) → 00h (second time), and if FFh is written after that, the writing order 00h → FFh is satisfied; writing 00h (n-1-th time) → 00h (nth time) → FFh is valid and correct refreshing will be done. Even when the first value written before 00h is not 00h, correct refreshing will be done if the operation contains the set of writing 00h → FFh. Moreover, even if a register other than the IWDTRR register is accessed or the IWDTRR register is read between writing 00h and writing FFh to the IWDTRR register, correct refreshing will be done.

[Sample sequences of writing that are valid for refreshing the counter]

- 00h → FFh
- 00h (n-1-th time) → 00h (nth time) → FFh
- 00h → access to another register or read from the IWDTRR register → FFh

[Sample sequences of writing that are not valid for refreshing the counter]

- 23h (a value other than 00h) → FFh
- 00h → 54h (a value other than FFh)
- 00h → AAh (00h and a value other than FFh) → FFh

Even when 00h is written to the IWDTRR register outside the refresh-permitted period, if FFh is written to the IWDTRR register in the refresh-permitted period, the writing sequence is valid and refreshing will be done.

After FFh is written to the IWDTRR register, refreshing the counter requires up to four cycles of the signal for counting (the IWDTCR.CKS[3:0] bits determine how many cycles of the IWDCLK make up one cycle for counting). Therefore, writing FFh to the IWDTRR register should be completed four-count cycles before the end position of the refresh-permitted period or a counter underflow. The value of the counter can be checked by the IWDTSR.CNTVAL[13:0] bits.

[Sample refreshing timings]

- When the window start position is set to 01FFh, even if 00h is written to the IWDTRR register before 01FFh is reached (0202h, for example), refreshing is done if FFh is written to the IWDTRR register after the value of the IWDTSR.CNTVAL[13:0] bits has reached 01FFh.
- When the window end position is set to 01FFh, refreshing is done if 0203h (four-count cycles before 01FFh) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 00h → FFh to the IWDTRR register.
- When the refresh-permitted period continues until count 0000h, refreshing can be done immediately before an underflow. In this case, if 0003h (four-count cycles before an underflow) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 00h → FFh to the IWDTRR register, no underflow occurs and refreshing is done.

Figure 25.5 shows the IWDt refresh operation waveforms when $PCLKB > IWDtCLK$ and the count clock is $IWDtCLK/1$.

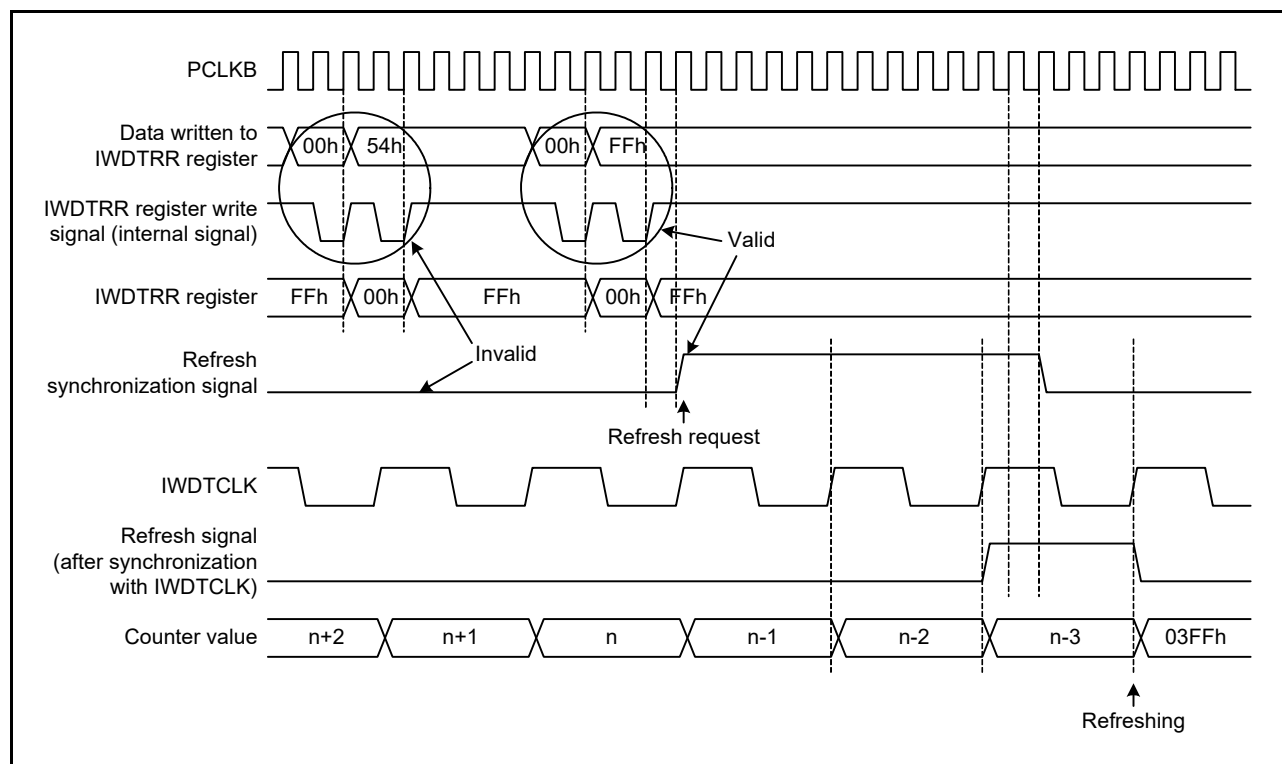


Figure 25.5 IWDt Refresh Operation Waveforms (IWDTCR.CKS[3:0] = 0000b, IWDTCR.TOPS[1:0] = 10b)

25.3.3 Status Flags

The IWDTSR.REFEF and IWDTSR.UNDF flags retain the source of the reset signal output from the IWDt or the source of the interrupt request from the IWDt.

Thus, after release from the reset state or interrupt request generation, read the IWDTSR.REFEF and IWDTSR.UNDF flags to check for the reset or interrupt source.

For each flag, writing 0 clears the bit and writing 1 has no effect.

Leaving the status flags unchanged does not affect operation. If the flags are not cleared, at the time of the next reset or interrupt request from the IWDt, the earlier reset or interrupt source is cleared and the new reset or interrupt source is written.

After 0 is written to each flag, up to three IWDtCLK cycles and two PCLKB cycles are required before the value is reflected.

25.3.4 Reset Output

When the IWDTRCR.RSTIRQS bit is set to 1 in register start mode or when the IWDTRSTIRQS bit in option function select register 0 (OFS0) is set to 1 in auto-start mode, a reset signal is output when an underflow in the counter or a refresh error occurs.

In register start mode, the counter is initialized (0000h) and stopped in that state after output of the reset signal. After the reset is released, the counter value is restored and down-counting is started by performing a refresh operation.

In auto-start mode, counting down automatically starts after the reset output.

25.3.5 Interrupt Sources

When the IWDTRCR.RSTIRQS bit is set to 0 in register start mode or when the OFS0.IWDTRSTIRQS bit is set to 0 in auto-start mode, an interrupt (IWUNI) signal is output when an underflow in the counter or a refresh error occurs. This interrupt can be used as a non-maskable interrupt. For details, refer to section 14, Interrupt Controller (ICUb).

Table 25.4 IWDt Interrupt Source

Name	Interrupt Source	DTC Activation
IWUNI	Counter underflow Refresh error	Not possible

25.3.6 Reading the Counter Value

As the counter in IWDTCCLK, the counter value cannot be read directly. The IWDt synchronizes the counter value with the PCLKB and stores it in the IWDTSR.CNTVAL[13:0] bits. Thus, the counter value can be checked indirectly through the IWDTSR.CNTVAL[13:0] bits.

Reading the counter value requires multiple PCLKB clock cycles (up to four clock cycles), and the read counter value may differ from the actual counter value by a value of one count.

Figure 25.6 shows the processing for reading the IWDt counter value when $PCLKB > IWDTCCLK$ and the count clock is $IWDTCCLK/1$.

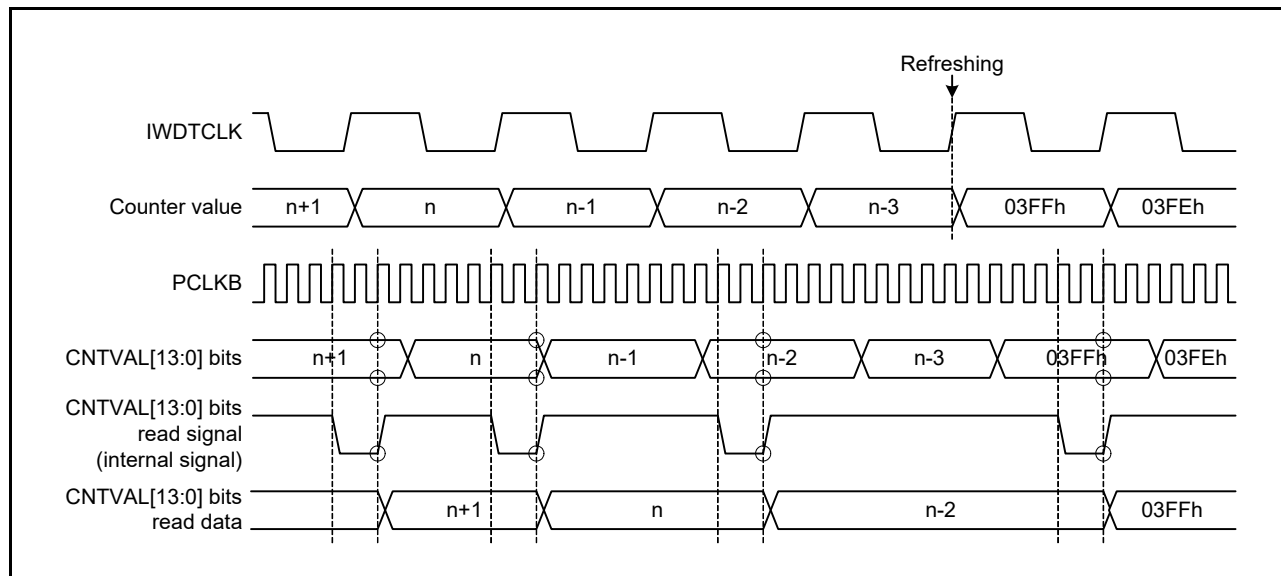


Figure 25.6 Processing for Reading IWDt Counter Value
(IWDTCR.CKS[3:0] = 0000b, IWDTCR.TOPS[1:0] = 10b)

25.3.7 Correspondence between Option Function Select Register 0 (OFS0) and IWDt Registers

Table 25.5 lists the correspondence between option function select register 0 (OFS0) used in auto-start mode and the registers used in register start mode.

For details on the OFS0 register, refer to section 7.2.1, Option Function Select Register 0 (OFS0).

Table 25.5 Correspondence between Option Function Select Register 0 (OFS0) and IWDt Registers

Target of Control	Function	OFS0 Register (Enabled in Auto-Start Mode) OFS0.IWDTSTRT = 0	IWDt Registers (Enabled in Register Start Mode) OFS0.IWDTSTRT = 1
Counter	Timeout period selection	OFS0.IWDTTOPS[1:0]	IWDTCR.TOPS[1:0]
	Clock divide ratio selection	OFS0.IWDTCKS[3:0]	IWDTCR.CKS[3:0]
	Window start position selection	OFS0.IWDTRPSS[1:0]	IWDTCR.RPSS[1:0]
	Window end position selection	OFS0.IWDTRPES[1:0]	IWDTCR.RPES[1:0]
Reset output or interrupt request output	Reset output or interrupt request output selection	OFS0.IWDTIRSTIRQS	IWDTCR.RSTIRQS
Count stop	Sleep mode count stop control	OFS0.IWDTSLCSTP	IWDTCSTPR.SLCSTP

25.4 Usage Notes

25.4.1 Refresh Operations

When making the settings to control the timing of refreshing, consider variations in the range of errors due to the accuracy of the PCLKB and IWDtCLK and set values which ensure that refreshing is possible.

25.4.2 Clock Divide Ratio Setting

Satisfy the frequency of the PCLKB $\geq 4 \times$ (the frequency of the count source after divide).

26. Serial Communications Interface (SCIg, SCIH)

This MCU has four independent serial communications interface (SCI) channels. The SCI consists of the SCIg module (SCI1, SCI5, and SCI6) and the SCIH module (SCI12).

The SCIg module (SCI1, SCI5, and SCI6) can handle both asynchronous and clock synchronous serial communications. Asynchronous serial data communications can be carried out with standard asynchronous communications chips such as a Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communications Interface Adapter (ACIA). As an extended function in asynchronous communications mode, the SCI also supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards). The SCI also supports simple SPI interfaces, and simple I²C-bus interfaces when configured for single-master systems.

The SCIH module includes the functions of the SCIg module, and supports an extended serial communication protocol formed of Start Frames and Information Frames.

In this section, "PCLK" is used to refer to PCLKB.

26.1 Overview

Table 26.1 lists the specifications of the SCIg module, Table 26.2 lists the specifications of the SCIH module, and Table 26.3 lists the specifications of the individual SCI channels.

Figure 26.1 shows the block diagram of SCI1, Figure 26.2 shows the block diagram of SCI5 and SCI6, and Figure 26.3 shows the block diagram of SCI12 (SCIH).

Table 26.1 SCIg Specifications (1/2)

Item		Description
Serial communication modes		- Asynchronous - Clock synchronous - Smart card interface - Simple I²C-bus - Simple SPI bus
Transfer speed		Bit rate specifiable with the on-chip baud rate generator.
Full-duplex communications		Transmitter: Continuous transmission possible using double-buffer structure. Receiver: Continuous reception possible using double-buffer structure.
I/O pins		Refer to Table 26.4 to Table 26.6.
Data transfer		Selectable as LSB first or MSB first transfer*1
Interrupt sources		Transmit end, transmit data empty, receive data full, and receive error Completion of generation of a start condition, restart condition, or stop condition (for simple I ² C mode)
Low power consumption function		Module stop state can be set for each channel.
Asynchronous mode	Data length	7, 8, or 9 bits
	Transmission stop bit	1 or 2 bits
	Parity	Even parity, odd parity, or no parity
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	CTS _n # and RTS _n # pins can be used in controlling transmission/reception.
	Start-bit detection	Low level or falling edge is selectable.
	Break detection	When a framing error occurs, a break can be detected by reading the RXD _n pin level directly.
	Clock source	An internal or external clock can be selected. Transfer rate clock input from the TMR can be used. (SCI5 and SCI6)
	Double-speed mode	Baud rate generator double-speed mode is selectable.
	Multi-processor communications function	Serial communication among multiple processors
Noise cancellation		The signal paths from input on the RXD _n pins incorporate digital noise filters.

Table 26.1 SCIg Specifications (2/2)

Item	Description	
Clock synchronous mode	Data length	8 bits
	Receive error detection	Overrun error
	Hardware flow control	CTS# and RTS# pins can be used in controlling transmission/reception.
Smart card interface mode	Error processing	An error signal can be automatically transmitted when detecting a parity error during reception
		Data can be automatically retransmitted when receiving an error signal during transmission
	Data type	Both direct convention and inverse convention are supported.
Simple I ² C mode	Transfer format	I ² C-bus format
	Operating mode	Master (single-master operation only)
	Transfer rate	Fast mode is supported (refer to section 26.2.11, Bit Rate Register (BRR) to set the transfer rate).
	Noise cancellation	The signal paths from input on the SSCLn and SSDAn pins incorporate digital noise filters, and the interval for noise cancellation is adjustable.
Simple SPI bus	Data length	8 bits
	Detection of errors	Overrun error
	SS input pin function	Applying the high level to the SSn# pin can cause the output pins to enter the high-impedance state.
	Clock settings	Four kinds of settings for clock phase and clock polarity are selectable.
Bit rate modulation function		Correction of outputs from the on-chip baud rate generator can reduce errors.

Note 1. In simple I²C mode, only MSB first is available.

Table 26.2 SCIH Specifications (1/2)

Item	Description
Serial communication modes	• Asynchronous • Clock synchronous • Smart card interface • Simple I²C-bus • Simple SPI bus
Transfer speed	Bit rate specifiable with the on-chip baud rate generator.
Full-duplex communications	Transmitter: Continuous transmission possible using double-buffer structure. Receiver: Continuous reception possible using double-buffer structure.
I/O pins	Refer to Table 26.4 to Table 26.7.
Data transfer	Selectable as LSB first or MSB first transfer*1
Interrupt sources	Transmit end, transmit data empty, receive data full, and receive error Completion of generation of a start condition, restart condition, or stop condition (for simple I ² C mode)
Low power consumption function	Module stop state can be set.

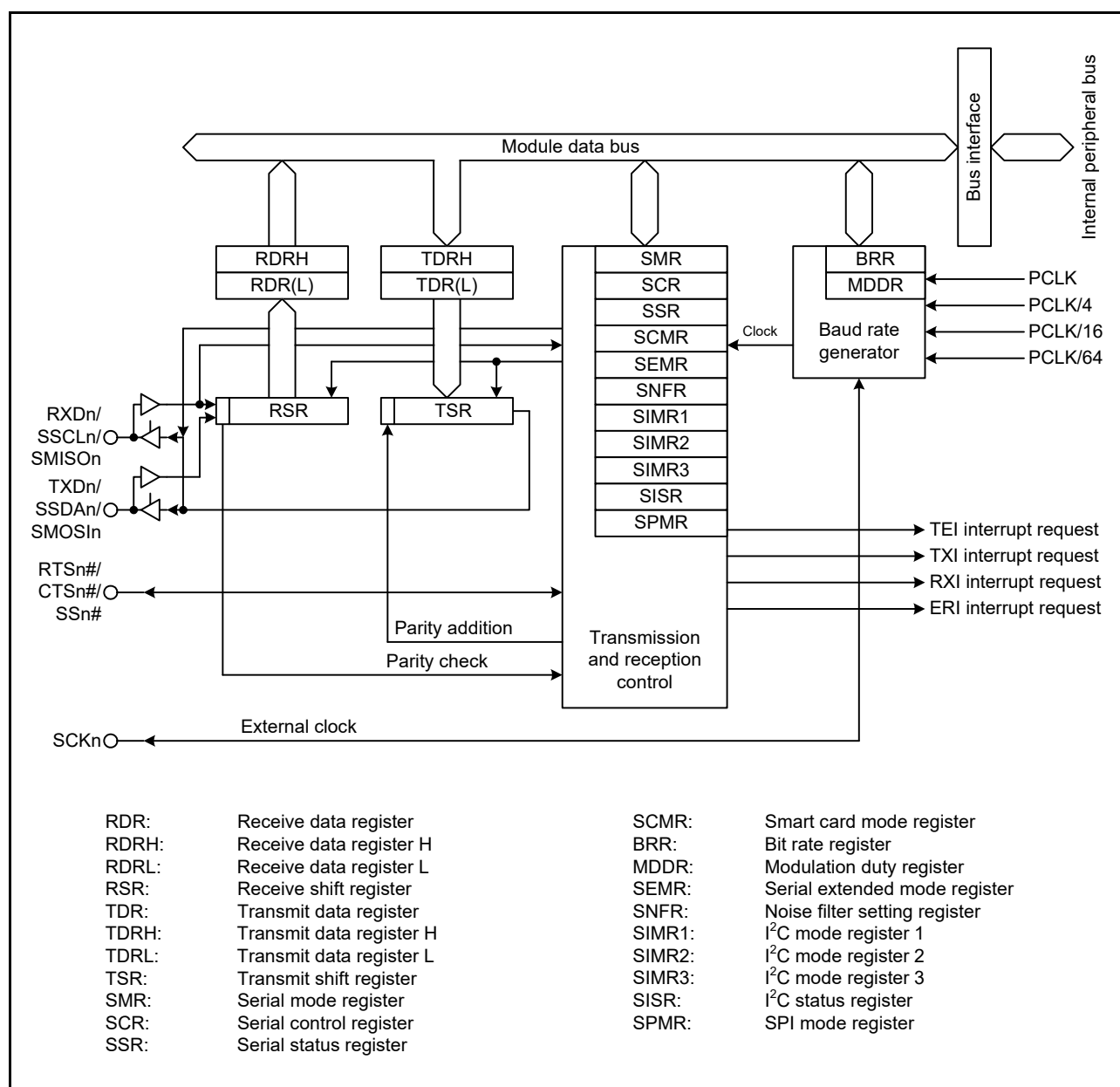
Table 26.2 SCIf Specifications (2/2)

Item	Description	
Asynchronous mode	Data length	7, 8, or 9 bits
	Transmission stop bit	1 or 2 bits
	Parity	Even parity, odd parity, or no parity
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	CTSn# and RTSn# pins can be used in controlling transmission/reception.
	Start-bit detection	Low level or falling edge is selectable.
	Break detection	When a framing error occurs, a break can be detected by reading the RXDn pin level directly.
	Clock source	An internal or external clock can be selected. Transfer rate clock input from the TMR can be used.
	Double-speed mode	Baud rate generator double-speed mode is selectable.
	Multi-processor communications function	Serial communication among multiple processors
	Noise cancellation	The signal paths from input on the RXDn pins incorporate digital noise filters.
Clock synchronous mode	Data length	8 bits
	Receive error detection	Overrun error
	Hardware flow control	CTSn# and RTSn# pins can be used in controlling transmission/reception.
Smart card interface mode	Error processing	An error signal can be automatically transmitted when detecting a parity error during reception
		Data can be automatically retransmitted when receiving an error signal during transmission
	Data type	Both direct convention and inverse convention are supported.
Simple I ² C mode	Transfer format	I ² C-bus format
	Operating mode	Master (single-master operation only)
	Transfer rate	Fast mode is supported (refer to section 26.2.11, Bit Rate Register (BRR) to set the transfer rate).
	Noise cancellation	The signal paths from input on the SSCLn and SSDAn pins incorporate digital noise filters, and the interval for noise cancellation is adjustable.
Simple SPI bus	Data length	8 bits
	Detection of errors	Overrun error
	SS input pin function	Applying the high level to the SSn# pin can cause the output pins to enter the high-impedance state.
	Clock settings	Four kinds of settings for clock phase and clock polarity are selectable.
Extended serial mode	Start Frame transmission	- Output of a low level as the Break Field over a specified width and generation of interrupts on completion - Detection of bus collisions and the generation of interrupts on detection
	Start Frame reception	- Detection of the Break Field low width and generation of an interrupt on detection - Comparison of Control Fields 0 and 1 and generation of an interrupt when the two match - Two kinds of data for comparison (primary and secondary) can be set in Control Field 1. - A priority interrupt bit can be set in Control Field 1. - Handling of Start Frames that do not include a Break Field - Handling of Start Frames that do not include a Control Field 0 - Function for measuring bit rates
	I/O control function	- Selectable polarity for TXDX12 and RXDX12 signals - Selection of a digital filter for the RXDX12 signal - Half-duplex operation employing RXDX12 and TXDX12 signals multiplexed on the same pin - Selectable timing for the sampling of data received through RXDX12
	Timer function	Usable as a reload timer
	Bit rate modulation function	Correction of outputs from the on-chip baud rate generator can reduce errors.

Note 1. In simple I²C mode, only MSB first is available.

Table 26.3 Functions of SCI Channels

Item	SCI1	SCI5	SCI6	SCI12
Asynchronous mode	Available	Available	Available	Available
Clock synchronous mode	Available	Available	Available	Available
Smart card interface mode	Available	Available	Available	Available
Simple I ² C mode	Available	Available	Available	Available
Simple SPI mode	Available	Available	Available	Available
Extended serial mode	Not available	Not available	Not available	Available
TMR clock input	Not available	Available	Available	Available

**Figure 26.1 Block Diagram of SCIg (SCI1)**

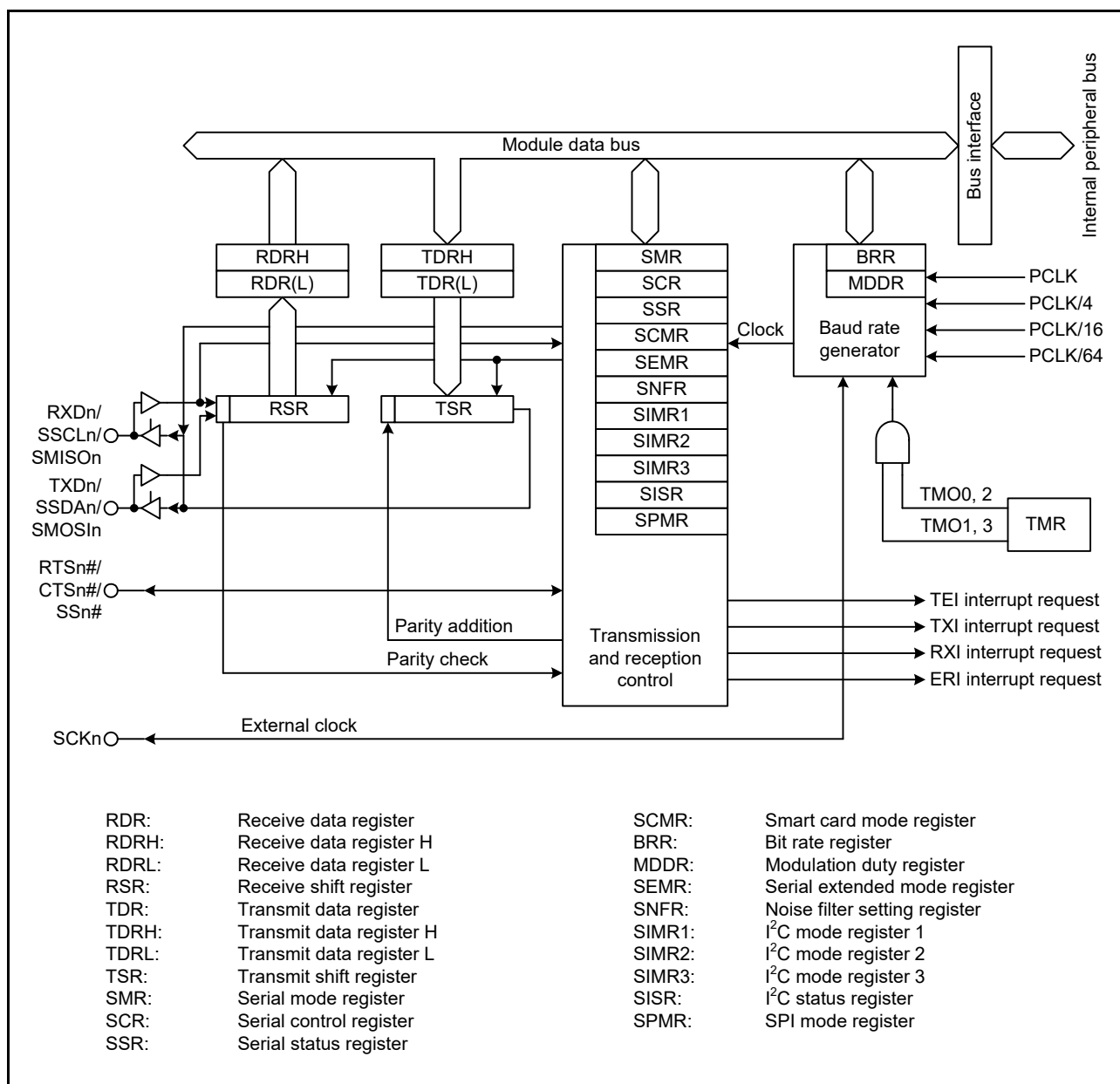


Figure 26.2 Block Diagram of SCIg (SCI5 and SCI6)

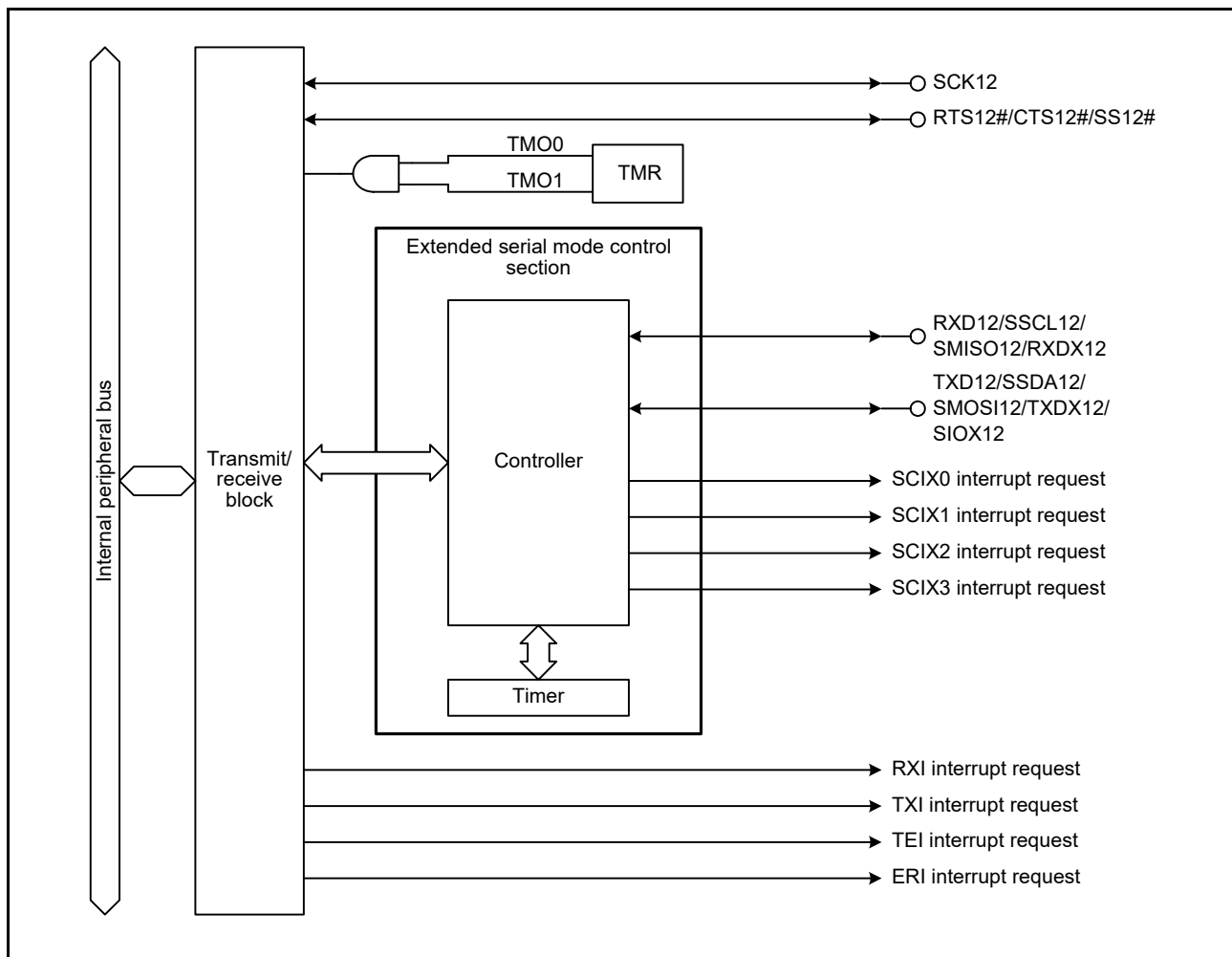


Figure 26.3 Block Diagram of SC1h (SCI12)

Table 26.4 to Table 26.7 list the pin configuration of the SCIs for the individual modes.

Table 26.4 SCI Pin Configuration in Asynchronous Mode and Clock Synchronous Mode

Channel	Pin Name	I/O	Function
SCI1	SCK1	I/O	SCI1 clock input/output
	RXD1	Input	SCI1 receive data input
	TXD1	Output	SCI1 transmit data output
	CTS1#/RTS1#	I/O	SCI1 transfer start control input/output
SCI5	SCK5	I/O	SCI5 clock input/output
	RXD5	Input	SCI5 receive data input
	TXD5	Output	SCI5 transmit data output
	CTS5#/RTS5#	I/O	SCI5 transfer start control input/output
SCI6	SCK6	I/O	SCI6 clock input/output
	RXD6	Input	SCI6 receive data input
	TXD6	Output	SCI6 transmit data output
	CTS6#/RTS6#	I/O	SCI6 transfer start control input/output
SCI12	SCK12	I/O	SCI12 clock input/output
	RXD12	Input	SCI12 receive data input
	TXD12	Output	SCI12 transmit data output
	CTS12#/RTS12#	I/O	SCI12 transfer start control input/output

Table 26.5 SCI Pin Configuration in Simple I²C Mode

Channel	Pin Name	I/O	Function
SCI1	SSCL1	I/O	SCI1 I ² C clock input/output
	SSDA1	I/O	SCI1 I ² C data input/output
SCI5	SSCL5	I/O	SCI5 I ² C clock input/output
	SSDA5	I/O	SCI5 I ² C data input/output
SCI6	SSCL6	I/O	SCI6 I ² C clock input/output
	SSDA6	I/O	SCI6 I ² C data input/output
SCI12	SSCL12	I/O	SCI12 I ² C clock input/output
	SSDA12	I/O	SCI12 I ² C data input/output

Table 26.6 SCI Pin Configuration in Simple SPI Mode (1/2)

Channel	Pin Name	I/O	Function
SCI1	SCK1	I/O	SCI1 clock input/output
	SMISO1	I/O	SCI1 slave transmit data input/output
	SMOSI1	I/O	SCI1 master transmit data input/output
	SS1#	Input	SCI1 chip select input
SCI5	SCK5	I/O	SCI5 clock input/output
	SMISO5	I/O	SCI5 slave transmit data input/output
	SMOSI5	I/O	SCI5 master transmit data input/output
	SS5#	Input	SCI5 chip select input
SCI6	SCK6	I/O	SCI6 clock input/output
	SMISO6	I/O	SCI6 slave transmit data input/output
	SMOSI6	I/O	SCI6 master transmit data input/output
	SS6#	Input	SCI6 chip select input

Table 26.6 SCI Pin Configuration in Simple SPI Mode (2/2)

Channel	Pin Name	I/O	Function
SCI12	SCK12	I/O	SCI12 clock input/output
	SMISO12	I/O	SCI12 slave transmit data input/output
	SMOSI12	I/O	SCI12 master transmit data input/output
	SS12#	Input	SCI12 chip select input

Table 26.7 SCI Pin Configuration in Extended Serial Mode

Channel	Pin Name	I/O	Function
SCI12	RXDX12	Input	SCI12 receive data input
	TXDX12	Output	SCI12 transmit data output
	SIOX12	I/O	SCI12 transfer data input/output

26.2 Register Descriptions

26.2.1 Receive Shift Register (RSR)

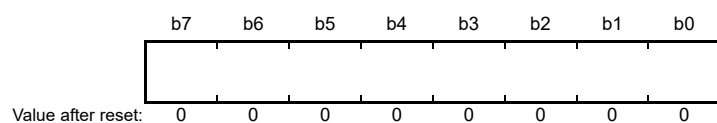
The RSR register is a shift register which is used to receive serial data input from the RXDn pin and converts it into parallel data.

When one frame of data has been received, it is automatically transferred to the RDR register.

The RSR register cannot be directly accessed by the CPU.

26.2.2 Receive Data Register (RDR)

Address(es): SCI1.RDR 0008 A025h, SCI5.RDR 0008 A0A5h, SCI6.RDR 0008 A0C5h, SCI12.RDR 0008 B305h



The RDR register is an 8-bit register that stores receive data.

When one frame of serial data has been received, the received serial data is transferred from the RSR register to the RDR register. Then the RSR register can receive the next data.

Since the RSR and RDR registers function as a double buffer in this way, continuous receive operations can be performed.

Read the RDR register only once after a receive data full interrupt (RXI) has occurred. Note that if next one frame of data is received before reading receive data from the RDR register, an overrun error occurs.

The RDR register cannot be written to by the CPU.

26.2.3 Receive Data Register H, L, HL (RDRH, RDRL, RDRHL)

- Receive Data Register H (RDRH)

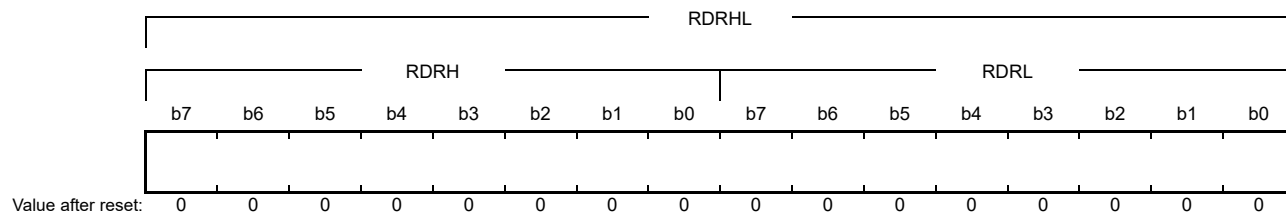
Address(es): SCI1.RDRH 0008 A030h, SCI5.RDRH 0008 A0B0h, SCI6.RDRH 0008 A0D0h, SCI12.RDRH 0008 B310h

- Receive Data Register L (RDRL)

Address(es): SCI1.RDRL 0008 A031h, SCI5.RDRL 0008 A0B1h, SCI6.RDRL 0008 A0D1h, SCI12.RDRL 0008 B311h

- Receive Data Register HL (RDRHL)

Address(es): SCI1.RDRHL 0008 A030h, SCI5.RDRHL 0008 A0B0h, SCI6.RDRHL 0008 A0D0h, SCI12.RDRHL 0008 B310h



The RDRH and RDRL registers are 8-bit registers that store receive data. Use these registers when asynchronous mode and 9-bit data length are selected.

The RDRL register is the shadow register of the RDR register; i.e. access to the RDRL register is equivalent to access to the RDR register.

After one frame of data is received, the received data is transferred from the RSR register to these registers, thus allowing the RSR register to receive the next data.

The RSR, RDRH and RDRL registers have a double-buffered construction to enable continuous reception.

Read the RDRH and RDRL registers should be performed only once in the order from the RDRH register to the RDRL register when a receive data full interrupt (RXI) request is issued. Note that an overrun error occurs when the next frame of data is received before the received data has been read from the RDRL register.

The CPU cannot write to the RDRH and RDRL registers. Bits 0 to 7 in the RDRH register are fixed to 0. These bits are read as 0.

The RDRHL register can be accessed in 16-bit units.

26.2.4 Transmit Data Register (TDR)

Address(es): SCI1.TDR 0008 A023h, SCI5.TDR 0008 A0A3h, SCI6.TDR 0008 A0C3h, SCI12.TDR 0008 B303h



The TDR register is an 8-bit register that stores transmit data.

When the SCI detects that the TSR register is empty, it transfers the transmit data written in the TDR register to the TSR register and starts transmission.

The double-buffered structures of the TDR register and the TSR register enable continuous serial transmission. If the next transmit data has already been written to the TDR register when one frame of data is transmitted, the SCI transfers the written data to the TSR register to continue transmission.

The CPU is able to read from or write to the TDR register at any time. Only write transmit data to the TDR register once after each instance of the transmit data empty interrupt (TXI).

26.2.5 Transmit Data Register H, L, HL (TDRH, TDRL, TDRHL)

- Transmit Data Register H (TDRH)

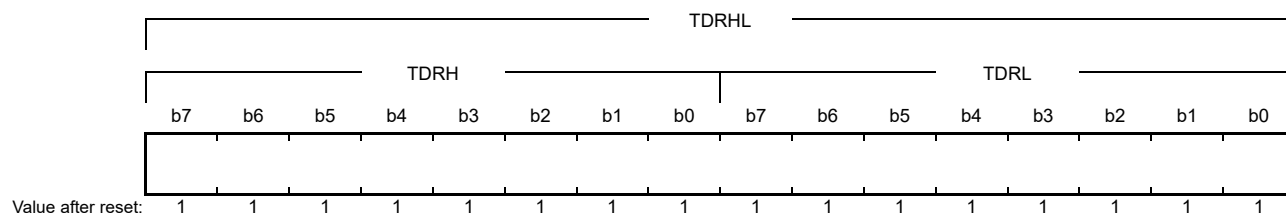
Address(es): SCI1.TDRH 0008 A02Eh, SCI5.TDRH 0008 A0AEh, SCI6.TDRH 0008 A0CEh, SCI12.TDRH 0008 B30Eh

- Transmit Data Register L (TDRL)

Address(es): SCI1.TDRL 0008 A02Fh, SCI5.TDRL 0008 A0AFh, SCI6.TDRL 0008 A0CFh, SCI12.TDRL 0008 B30Fh

- Transmit Data Register HL (TDRHL)

Address(es): SCI1.TDRHL 0008 A02Eh, SCI5.TDRHL 0008 A0AEh, SCI6.TDRHL 0008 A0CEh, SCI12.TDRHL 0008 B30Eh



The TDRH and TDRL registers are 8-bit registers that store transmit data. Use these registers when asynchronous mode and 9-bit data length are selected.

The TDRL register is the shadow register of the TDR register; i.e. access to the TDRL register is equivalent to access to the TDR register.

When empty space is detected in the TSR register, the transmit data stored in the TDRH and TDRL registers is transferred to the TSR register; i.e., transmitting is started.

The TSR, TDRH and TDRL registers have a double-buffered construction to realize continuous reception. When the next data to be transmitted is stored in the TDRL register after one frame of data has been transmitted, the transmitting operation is continued by transfer to the TSR register.

The CPU can read and write to the TDRH and TDRL registers. Bits 0 to 7 in the RDRH register are fixed to 1. These bits are read as 1. The write value should be 1.

Writing transmit data to the TDRH and TDRL registers should be performed only once in the order from the TDRH register to the TDRL register when a transmit data empty interrupt (TXI) request is issued.

The TDRHL register can be accessed in 16-bit units.

26.2.6 Transmit Shift Register (TSR)

The TSR register is a shift register that transmits serial data.

To perform serial data transmission, the SCI first automatically transfers transmit data from the TDR register to the TSR register, and then sends the data to the TXDn pin.

The TSR register cannot be directly accessed by the CPU.

26.2.7 Serial Mode Register (SMR)

Some bits in the SMR register have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Address(es): SCI1.SMR 0008 A020h, SCI5.SMR 0008 A0A0h, SCI6.SMR 0008 A0C0h, SCI12.SMR 0008 B300h

b7	b6	b5	b4	b3	b2	b1	b0
CM	CHR	PE	PM	STOP	MP	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: PCLK (n = 0)*1 0 1: PCLK/4 (n = 1)*1 1 0: PCLK/16 (n = 2)*1 1 1: PCLK/64 (n = 3)*1	R/W*4
b2	MP	Multi-Processor Mode	(Valid only in asynchronous mode) 0: Multi-processor communications function is disabled 1: Multi-processor communications function is enabled	R/W*4
b3	STOP	Stop Bit Length	(Valid only in asynchronous mode) 0: 1 stop bit 1: 2 stop bits	R/W*4
b4	PM	Parity Mode	(Valid only when the PE bit is 1) 0: Selects even parity 1: Selects odd parity	R/W*4
b5	PE	Parity Enable	(Valid only in asynchronous mode) - When transmitting 0: Parity bit addition is not performed 1: The parity bit is added - When receiving 0: Parity bit checking is not performed 1: The parity bit is checked	R/W*4
b6	CHR	Character Length	(Valid only in asynchronous mode*2) Selects in combination with the SCMR.CHR1 bit. CHR1 CHR 0 0: Transmit/receive in 9-bit data length 0 1: Transmit/receive in 9-bit data length 1 0: Transmit/receive in 8-bit data length (initial value) 1 1: Transmit/receive in 7-bit data length*3	R/W*4
b7	CM	Communications Mode	0: Asynchronous mode or simple I ² C mode 1: Clock synchronous mode or simple SPI mode	R/W*4

Note 1. n is the decimal notation of the value of n in the BRR register (refer to section 26.2.11, Bit Rate Register (BRR)).

Note 2. In other than asynchronous mode, this bit setting is invalid and a fixed data length of 8 bits is used.

Note 3. LSB first is fixed and the MSB (bit 7) in the TDR register is not transmitted in transmission.

Note 4. Writable only when the SCR.TE bit is 0 and the SCR.RE bit is 0 (both serial transmission and reception are disabled).

CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.

For the relation between the settings of these bits and the baud rate, refer to section 26.2.11, Bit Rate Register (BRR).

MP Bit (Multi-Processor Mode)

Disables/enables the multi-processor communications function. The settings of the PE bit and PM bit are invalid in multi-processor mode.

STOP Bit (Stop Bit Length)

Selects the stop bit length in transmission.

In reception, only the first stop bit is checked regardless of this bit setting. If the second stop bit is 0, it is treated as the start bit of the next transmit frame.

PM Bit (Parity Mode)

Selects the parity mode (even or odd) for transmission and reception.

The setting of the PM bit is invalid in multi-processor mode.

PE Bit (Parity Enable)

When this bit is set to 1, the parity bit is added to transmit data, and the parity bit is checked in reception.

Irrespective of the setting of the PE bit, the parity bit is not added or checked in multi-processor format.

CHR Bit (Character Length)

Selects the data length for transmission and reception.

Selects in combination with the SCMR.CHR1 bit.

In other than asynchronous mode, a fixed data length of 8 bits is used.

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

Address(es): SMC11.SMR 0008 A020h, SMC15.SMR 0008 A0A0h, SMC16.SMR 0008 A0C0h, SMC112.SMR 0008 B300h

b7	b6	b5	b4	b3	b2	b1	b0
GM	BLK	PE	PM	BCP[1:0]		CKS[1:0]	
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: PCLK (n = 0)*1 0 1: PCLK/4 (n = 1)*1 1 0: PCLK/16 (n = 2)*1 1 1: PCLK/64 (n = 3)*1	R/W*2
b3, b2	BCP[1:0]	Base Clock Pulse	Selects the number of base clock cycles in combination with the SCMR.BCP2 bit. Table 26.8 lists the combinations of the SCMR.BCP2 bit and SMR.BCP[1:0] bits.	R/W*2
b4	PM	Parity Mode	(Valid only when the PE bit is 1) 0: Selects even parity 1: Selects odd parity	R/W*2
b5	PE	Parity Enable	When this bit is set to 1, a parity bit is added to transmit data, and the parity of received data is checked. Set this bit to 1 in smart card interface mode.	R/W*2
b6	BLK	Block Transfer Mode	0: Non-block transfer mode operation 1: Block transfer mode operation	R/W*2
b7	GM	GSM Mode	0: Non-GSM mode operation 1: GSM mode operation	R/W*2

Note 1. n is the decimal notation of the value of n in the BRR register (refer to section 26.2.11, Bit Rate Register (BRR)).

Note 2. Writable only when the SCR.TE bit is 0 and the SCR.RE bit is 0 (both serial transmission and reception are disabled).

CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.

For the relationship between the settings of these bits and the baud rate, refer to section 26.2.11, Bit Rate Register (BRR).

BCP[1:0] Bits (Base Clock Pulse)

These bits select the number of base clock cycles in a 1-bit data transfer time in smart card interface mode.

Set these bits in combination with the SCMR.BCP2 bit.

For details, refer to section 26.6.4, Receive Data Sampling Timing and Reception Margin.

Table 26.8 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits

SCMR.BCP2 Bit	SMR.BCP[1:0] Bits		Number of Base Clock Cycles for 1-Bit Transfer Period
0	0	0	93 clock cycles (S = 93)*1
0	0	1	128 clock cycles (S = 128)*1
0	1	0	186 clock cycles (S = 186)*1
0	1	1	512 clock cycles (S = 512)*1
1	0	0	32 clock cycles (S = 32)*1 (Initial Value)
1	0	1	64 clock cycles (S = 64)*1
1	1	0	372 clock cycles (S = 372)*1
1	1	1	256 clock cycles (S = 256)*1

Note 1. S is the value of S in the BRR register (refer to section 26.2.11, Bit Rate Register (BRR)).

PM Bit (Parity Mode)

Selects the parity mode for transmission and reception (even or odd).

For details on the usage of this bit in smart card interface mode, refer to [section 26.6.2, Data Format \(Except in Block Transfer Mode\)](#).

PE Bit (Parity Enable)

Set the PE bit to 1.

The parity bit is added to transmit data before transmission, and the parity bit is checked in reception.

BLK Bit (Block Transfer Mode)

Setting this bit to 1 allows block transfer mode operation.

For details, refer to [section 26.6.3, Block Transfer Mode](#).

GM Bit (GSM Mode)

Setting this bit to 1 allows GSM mode operation.

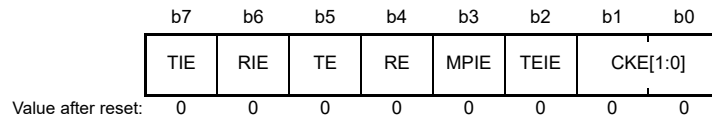
In GSM mode, the SSR.TEND flag set timing is put forward to 11.0 etu (elementary time unit = 1-bit transfer time) from the start and the clock output control function is appended. For details, refer to [section 26.6.6, Serial Data Transmission \(Except in Block Transfer Mode\)](#) and [section 26.6.8, Clock Output Control](#).

26.2.8 Serial Control Register (SCR)

Some bits in the SCR register have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Address(es): SCI1.SCR 0008 A022h, SCI5.SCR 0008 A0A2h, SCI6.SCR 0008 A0C2h, SCI12.SCR 0008 B302h



Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	(Asynchronous mode) b1 b0 0 0: On-chip baud rate generator The SCKn pin becomes high-impedance. 0 1: On-chip baud rate generator The clock with the same frequency as the bit rate is output from the SCKn pin. 1 x: External clock or TMR clock*2 • The clock with a frequency 16 times the bit rate should be input from the SCKn pin when an external clock is used. Input a clock signal with a frequency eight times the bit rate when the SEMR.ABCS bit is 1. • The SCKn pin becomes high-impedance when the TMR clock*2 is used.	R/W*1
			(Clock synchronous mode) b1 b0 0 x: Internal clock The SCKn pin functions as the clock output pin. 1 x: External clock The SCKn pin functions as the clock input pin.	
b2	TEIE	Transmit End Interrupt Enable	0: A TEI interrupt request is disabled 1: A TEI interrupt request is enabled	R/W
b3	MPIE	Multi-Processor Interrupt Enable	(Valid in asynchronous mode when the SMR.MP bit is 1) 0: Normal reception 1: When the data with the multi-processor bit set to 0 is received, the data is not read, and setting the status flags RDRF, ORER, and FER in the SSR register to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE bit is automatically cleared to 0, and normal reception is resumed.	R/W
b4	RE	Receive Enable	0: Serial reception is disabled 1: Serial reception is enabled	R/W*3
b5	TE	Transmit Enable	0: Serial transmission is disabled 1: Serial transmission is enabled	R/W*3
b6	RIE	Receive Interrupt Enable	0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled	R/W
b7	TIE	Transmit Interrupt Enable	0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled	R/W

x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. TMR clock is selectable for SCI5, SCI6, and SCI12.

Note 3. 1 can be written only when TE = 0 and RE = 0, while the SMR.CM bit is 1. After setting TE or RE to 1, only 0 can be written to TE and RE. While the SMR.CM bit is 0 and the SIMR1.IICM bit is 0, writing is enabled under any condition.

CKE[1:0] Bits (Clock Enable)

These bits select the clock source and SCKn pin function.

The combination of the settings of these bits and of the SEMR.ACS0 bit sets the internal TMR clock.

TEIE Bit (Transmit End Interrupt Enable)

Enables or disables a TEI interrupt request.

A TEI interrupt request is disabled by setting the TEIE bit to 0.

In simple I²C mode, the TEI is allocated to the interrupt on completion of issuing a start, restart, or stop condition (STI).

In this case, the TEIE bit can be used to enable or disable the STI.

MPIE Bit (Multi-Processor Interrupt Enable)

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 status flags RDRF, ORER, and FER in the SSR register to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE bit is automatically cleared to 0, and normal reception is resumed. For details, refer to **section 26.4, Multi-Processor Communications Function**.

When the data with the multi-processor bit set to 0 is received, the receive data is not transferred from the RSR register to the RDR register, a receive error is not detected, and setting the flags RDRF, ORER, and FER to 1 is disabled.

When the data with the multi-processor bit set to 1 is received, the MPB bit is set to 1, the MPPIE bit is automatically cleared to 0, the RXI and ERI interrupt requests are enabled (if the SCR.RIE bit is set to 1), and setting the flags RDRF, ORER, and FER to 1 is enabled.

Set the MPPIE bit to 0 if multi-processor communications function is not to be used.

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit in asynchronous mode or the synchronous clock input in clock synchronous mode. Note that the SMR register should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, PER, and RDRF flags in the SSR register are not affected and the previous value is retained.

TE Bit (Transmit Enable)

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to the TDR register. Note that the SMR register should be set prior to setting the TE bit to 1 in order to designate the transmission format.

RIE Bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by setting the RIE bit to 0.

An ERI interrupt request can be canceled by reading 1 from the ORER, FER, or PER flag in the SSR register and then setting the flag to 0, or setting the RIE bit to 0.

TIE Bit (Transmit Interrupt Enable)

Enables or disables TXI interrupt request.

A TXI interrupt request is disabled by setting the TIE bit to 0.

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

Address(es): SMCI1.SCR 0008 A022h, SMCI5.SCR 0008 A0A2h, SMCI6.SCR 0008 A0C2h, SMCI12.SCR 0008 B302h

b7	b6	b5	b4	b3	b2	b1	b0
TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	- When SMR.GM = 0 - b1 b0 0 0: Output disabled - The SCKn pin becomes high-impedance. 0 1: Clock output 1 x: Setting prohibited - When SMR.GM = 1 - b1 b0 0 0: Output fixed low - x 1: Clock output 1 0: Output fixed high	R/W*1
b2	TEIE	Transmit End Interrupt Enable	This bit should be 0 in smart card interface mode.	R/W
b3	MPIE	Multi-Processor Interrupt Enable	This bit should be 0 in smart card interface mode.	R/W
b4	RE	Receive Enable	0: Serial reception is disabled 1: Serial reception is enabled	R/W*2
b5	TE	Transmit Enable	0: Serial transmission is disabled 1: Serial transmission is enabled	R/W*2
b6	RIE	Receive Interrupt Enable	0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled	R/W
b7	TIE	Transmit Interrupt Enable	0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled	R/W

x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. 1 can be written only when TE = 0 and RE = 0. After setting TE or RE to 1, only 0 can be written in TE and RE.

For details on interrupt requests, refer to section 26.12, Interrupt Sources.

CKE[1:0] Bits (Clock Enable)

These bits control the clock output from the SCKn pin.

In GSM mode, clock output can be dynamically switched. For details, refer to section 26.6.8, Clock Output Control.

TEIE Bit (Transmit End Interrupt Enable)

This bit should be 0 in smart card interface mode.

MPIE Bit (Multi-Processor Interrupt Enable)

This bit should be 0 in smart card interface mode.

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit. Note that the SMR register should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, and PER flags in the SSR register are not affected and the previous value is retained.

TE Bit (Transmit Enable)

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to the TDR register. Note that the SMR register should be set prior to setting the TE bit to 1 in order to designate the transmission format.

RIE Bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by setting the RIE bit to 0.

An ERI interrupt request can be canceled by reading 1 from the ORER, FER, or PER flag in the SSR register and then setting the flag to 0, or setting the RIE bit to 0.

TIE Bit (Transmit Interrupt Enable)

Enables or disables TXI interrupt request.

A TXI interrupt request is disabled by setting the TIE bit to 0.

26.2.9 Serial Status Register (SSR)

Some bits in the SSR register have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Address(es): SCIf1.SSR 0008 A024h, SCIf5.SSR 0008 A0A4h, SCIf6.SSR 0008 A0C4h, SCIf12.SSR 0008 B304h

	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Value after reset:	1	0	0	0	0	1	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MPBT	Multi-Processor Bit Transfer	Sets the multi-processor bit for adding to the transmission frame 0: Data transmission cycles 1: ID transmission cycles	R/W
b1	MPB	Multi-Processor	Value of the multi-processor bit in the reception frame 0: Data transmission cycles 1: ID transmission cycles	R
b2	TEND	Transmit End Flag	0: A character is being transmitted. 1: Character transfer has been completed.	R
b3	PER	Parity Error Flag	0: No parity error occurred 1: A parity error has occurred	R/(W) *1
b4	FER	Framing Error Flag	0: No framing error occurred 1: A framing error has occurred	R/(W) *1
b5	ORER	Overrun Error Flag	0: No overrun error occurred 1: An overrun error has occurred	R/(W) *1
b6	RDRF	Receive Data Full Flag	0: No valid data is held in the RDR register 1: Received data is held in the RDR register	R/(W) *2
b7	TDRE	Transmit Data Empty Flag	0: Data to be transmitted is held in the TDR register 1: No data is held in the TDR register	R/(W) *2

Note 1. Only 0 can be written to this bit, to clear the flag. To clear this flag, confirm that the flag is 1 and then set it to 0.

Note 2. Write 1 when writing is necessary.

MPB Bit (Multi-Processor)

Holds the value of the multi-processor bit in the reception frame. This bit does not change when the SCR.RE bit is 0.

TEND Flag (Transmit End Flag)

Indicates completion of transmission.

[Setting conditions]

- When the SCR.TE bit is set to 0 (serial transmission is disabled)
When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- When the TDR register is not updated at the time of transmission of the tail-end bit of a character being transmitted

[Clearing condition]

- When transmit data are written to the TDR register while the SCR.TE bit is 1
When setting the TEND flag to 0 to complete the interrupt handling, refer to section 14.4.1.2, Operation of Status Flags for Level-Detected Interrupts.

PER Flag (Parity Error Flag)

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception
Although receive data when the parity error occurs is transferred to the RDR register, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to the RDR register.

[Clearing condition]

- When 0 is written to the PER flag after reading PER = 1
When setting the PER flag to 0 to complete the interrupt handling, refer to **section 14.4.1.2, Operation of Status Flags for Level-Detected Interrupts**.
Even when the SCR.RE bit is set to 0 (serial reception is disabled), the PER flag is not affected and retains its previous value.

FER Flag (Framing Error Flag)

Indicates that a framing error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When the stop bit is 0
In 2-stop-bit mode, only the first stop bit is checked whether it is 1 but the second stop bit is not checked. Note that although receive data when the framing error occurs is transferred to the RDR register, no RXI interrupt request occurs. In addition, when the FER flag is being set to 1, the subsequent receive data is not transferred to the RDR register.

[Clearing condition]

- When 0 is written to the FER flag after reading FER = 1
When setting the FER flag to 0 to complete the interrupt handling, refer to **section 14.4.1.2, Operation of Status Flags for Level-Detected Interrupts**.
Even when the SCR.RE bit is set to 0, the FER flag is not affected and retains its previous value.

ORER Flag (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from the RDR register
In the RDR register, receive data prior to an overrun error occurrence 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 performed. Note that, in clock synchronous mode, serial transmission also cannot continue.

[Clearing condition]

- When 0 is written to the ORER flag after reading ORER = 1
When setting the ORER flag to 0 to complete the interrupt handling, refer to **section 14.4.1.2, Operation of Status Flags for Level-Detected Interrupts**.
Even when the SCR.RE bit is set to 0, the ORER flag is not affected and retains its previous value.

RDRF Flag (Receive Data Full Flag)

Indicates whether the RDR register has received data.

[Setting condition]

- When data has been received normally, and transferred from the RSR register to the RDR register

[Clearing condition]

- When data is read from the RDR register

TDRE Flag (Transmit Data Empty Flag)

Indicates whether the TDR register has data to be transmitted.

[Setting condition]

- When data is transferred from the TDR register to the TSR register

[Clearing condition]

- When data is written to the TDR register

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

Address(es): SMC11.SSR 0008 A024h, SMC15.SSR 0008 A0A4h, SMC16.SSR 0008 A0C4h, SMC112.SSR 0008 B304h

b7	b6	b5	b4	b3	b2	b1	b0
TDRE	RDRF	ORER	ERS	PER	TEND	MPB	MPBT
Value after reset:	1	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	MPBT	Multi-Processor Bit Transfer	This bit should be set to 0 in smart card interface mode.	R/W
b1	MPB	Multi-Processor	This bit is not used in smart card interface mode. It should be set to 0.	R
b2	TEND	Transmit End Flag	0: A character is being transmitted. 1: Character transfer has been completed.	R
b3	PER	Parity Error Flag	0: No parity error occurred 1: A parity error has occurred	R/(W) *1
b4	ERS	Error Signal Status Flag	0: Low error signal not responded 1: Low error signal responded	R/(W) *1
b5	ORER	Overflow Error Flag	0: No overflow error occurred 1: An overflow error has occurred	R/(W) *1
b6	RDRF	Receive Data Full Flag	0: No valid data is held in the RDR register 1: Received data is held in the RDR register	R/(W) *2
b7	TDRE	Transmit Data Empty Flag	0: Data to be transmitted is held in the TDR register 1: No data is held in the TDR register	R/(W) *2

Note 1. Only 0 can be written to this bit, to clear the flag. To clear this flag, confirm that the flag is 1 and then set it to 0.

Note 2. Write 1 when writing is necessary.

TEND Flag (Transmit End Flag)

With no error signal from the receiving side, this bit is set to 1 when further data for transfer is ready to be transferred to the TDR register.

[Setting conditions]

- When the SCR.TE bit is 0 (serial transmission is disabled)
When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- When a specified period has elapsed after the latest transmission of 1 byte, the ERS flag is 0, and the TDR register is not updated
The set timing is determined by register settings as listed below.
When SMR.GM = 0 and SMR.BLK = 0, 12.5 etu after the start of transmission
When SMR.GM = 0 and SMR.BLK = 1, 11.5 etu after the start of transmission
When SMR.GM = 1 and SMR.BLK = 0, 11.0 etu after the start of transmission
When SMR.GM = 1 and SMR.BLK = 1, 11.0 etu after the start of transmission

[Clearing condition]

- When transmit data are written to the TDR register while the SCR.TE bit is 1
When setting the TEND flag to 0 to complete the interrupt handling, refer to **section 14.4.1.2, Operation of Status Flags for Level-Detected Interrupts**.

PER Flag (Parity Error Flag)

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception
Although receive data when the parity error occurs is transferred to the RDR register, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to the RDR register.

[Clearing condition]

- When 0 is written to the PER flag after reading PER = 1
When setting the PER flag to 0 to complete the interrupt handling, refer to **section 14.4.1.2, Operation of Status Flags for Level-Detected Interrupts**.
Even when the SCR.RE bit is set to 0 (serial reception is disabled), the PER flag is not affected and retains its previous value.

ERS Flag (Error Signal Status Flag)

[Setting condition]

- When a low error signal is sampled

[Clearing condition]

- When 0 is written to the ERS flag after reading ERS = 1
When setting the ERS flag to 0 to complete the interrupt handling, refer to **section 14.4.1.2, Operation of Status Flags for Level-Detected Interrupts**.
Even when the SCR.RE bit is set to 0, the ERS flag is not affected and retains its previous value.

ORER Flag (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from the RDR register
In the RDR register, the receive data prior to an overrun error occurrence is retained, but data received following the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed.

[Clearing condition]

- When 0 is written to the ORER flag after reading ORER = 1
When setting the ORER flag to 0 to complete the interrupt handling, refer to section 14.4.1.2, Operation of Status Flags for Level-Detected Interrupts.
Even when the SCR.RE bit is set to 0, the ORER flag is not affected and retains its previous value.

RDRF Flag (Receive Data Full Flag)

Indicates whether the RDR register has received data.

[Setting condition]

- When data has been received normally, and transferred from the RSR register to the RDR register

[Clearing condition]

- When data is read from the RDR register

TDRE Flag (Transmit Data Empty Flag)

Indicates whether the TDR register has data to be transmitted.

[Setting condition]

- When data is transferred from the TDR register to the TSR register

[Clearing condition]

- When data is written to the TDR register

26.2.10 Smart Card Mode Register (SCMR)

Address(es): SCI1.SCMR 0008 A026h, SCI5.SCMR 0008 A0A6h, SCI6.SCMR 0008 A0C6h, SCI12.SCMR 0008 B306h, SMC11.SCMR 0008 A026h, SMC15.SCMR 0008 A0A6h, SMC16.SCMR 0008 A0C6h, SMC112.SCMR 0008 B306h

	b7	b6	b5	b4	b3	b2	b1	b0
	BCP2	—	—	CHR1	SDIR	SINV	—	SMIF
Value after reset:	1	1	1	1	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	SMIF	Smart Card Interface Mode Select	0: Non-smart card interface mode (Asynchronous mode, clock synchronous mode, simple SPI mode, or simple I ² C mode) 1: Smart card interface mode	R/W*1
b1	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b2	SINV	Transmitted/Received Data Invert*2, *3	0: Data bits in the TDR register are transferred to the TSR register as they are. Data bits in the RSR register are transferred to the RDR register as they are. 1: Data bits in the TDR register are transferred to the TSR register with inverting. Data bits in the RSR register are transferred to the RDR register with inverting.	R/W*1
b3	SDIR	Transmitted/Received Data Transfer Direction*2, *4	0: Transfer with LSB first 1: Transfer with MSB first	R/W*1
b4	CHR1	Character Length 1*5	Selects in combination with the SMR.CHR bit. CHR1 CHR 0 0: Transmit/receive in 9-bit data length 0 1: Transmit/receive in 9-bit data length 1 0: Transmit/receive in 8-bit data length (initial value) 1 1: Transmit/receive in 7-bit data length*6	R/W*1
b6, b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b7	BCP2	Base Clock Pulse 2	Selects the number of base clock cycles in combination with the SMR.BCP[1:0] bits. Table 26.9 lists the combinations of the SCMR.BCP2 bit and SMR.BCP[1:0] bits.	R/W*1

Note 1. Writable only when the SCR.TE bit is 0 and the SCR.RE bit is 0 (both serial transmission and reception are disabled).

Note 2. This bit can be used in the smart card interface mode, asynchronous mode (multi-processor mode), clock synchronous mode, and simple SPI mode.

Note 3. Set this bit to 0 if operation is to be in simple I²C mode.

Note 4. Set this bit to 1 if operation is to be in simple I²C mode.

Note 5. This bit is only valid in asynchronous mode. The setting is invalid and a fixed data length of 8 bits is used in modes other than asynchronous mode.

Note 6. LSB first should be selected and the value of MSB (b7) in the TDR register cannot be transmitted.

SMIF Bit (Smart Card Interface Mode Select)

When this bit is set to 1, smart card interface mode is selected.

When this bit is set to 0, non-smart card interface mode, i.e., asynchronous mode (including multi-processor mode), clock synchronous mode, simple SPI mode, or simple I²C mode is selected.

SINV Bit (Transmitted/Received Data Invert)

This bit is used to invert the logic level of the data bits when the data is transferred between data register and shift register. This bit does not affect the logic level of the parity bit. To invert the parity bit, invert the SMR.PM bit.

CHR1 Bit (Character Length 1)

Selects the data length of transmit/receive data.

Selects in combination with the SMR.CHR bit.

A fixed data length of 8 bits is used in modes other than asynchronous mode.

BCP2 Bit (Base Clock Pulse 2)

Selects the number of base clock cycles in a 1-bit data transfer time in smart card interface mode. Set this bit in combination with the SMR.BCP[1:0] bits.

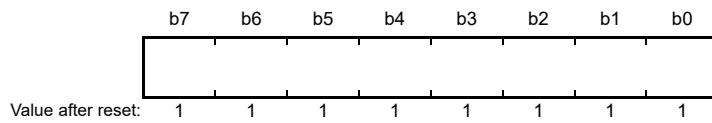
Table 26.9 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits

SCMR.BCP2 Bit	SMR.BCP[1:0] Bits		Number of Base Clock Cycles for 1-Bit Transfer Period
0	0	0	93 clock cycles (S = 93)* ¹
0	0	1	128 clock cycles (S = 128)* ¹
0	1	0	186 clock cycles (S = 186)* ¹
0	1	1	512 clock cycles (S = 512)* ¹
1	0	0	32 clock cycles (S = 32)* ¹ (Initial Value)
1	0	1	64 clock cycles (S = 64)* ¹
1	1	0	372 clock cycles (S = 372)* ¹
1	1	1	256 clock cycles (S = 256)* ¹

Note 1. S is the value of S in the BRR register (refer to section 26.2.11, Bit Rate Register (BRR)).

26.2.11 Bit Rate Register (BRR)

Address(es): SCI1.BRR 0008 A021h, SCI5.BRR 0008 A0A1h, SCI6.BRR 0008 A0C1h, SCI12.BRR 0008 B301h



The BRR register is an 8-bit register that adjusts the bit rate.

As each SCI channel has independent baud rate generator control, different bit rates can be set for each. Table 26.10 shows the relationship between the setting (N) in the BRR register and the bit rate (B) for normal asynchronous mode, multi-processor communication, clock synchronous mode, smart card interface mode, simple SPI mode, and simple I²C mode.

The BRR register is writable only when the TE and RE bits in the SCR register are 0.

Table 26.10 Relationship between N Setting in the BRR Register and Bit Rate B

Mode	SEMR Settings		BRR Setting	Error (%)
	BGDM bit	ABCS bit		
Asynchronous, multi-processor communication	0	0	$N = \frac{PCLK \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	$Error = \left\{ \frac{PCLK \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
	0	1	$N = \frac{PCLK \times 10^6}{32 \times 2^{2n-1} \times B} - 1$	$Error = \left\{ \frac{PCLK \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
	1	0		
	1	1	$N = \frac{PCLK \times 10^6}{16 \times 2^{2n-1} \times B} - 1$	$Error = \left\{ \frac{PCLK \times 10^6}{B \times 16 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
Clock synchronous, simple SPI			$N = \frac{PCLK \times 10^6}{8 \times 2^{2n-1} \times B} - 1$	
Smart card interface			$N = \frac{PCLK \times 10^6}{S \times 2^{2n+1} \times B} - 1$	$Error = \left\{ \frac{PCLK \times 10^6}{B \times S \times 2^{2n+1} \times (N+1)} - 1 \right\} \times 100$
Simple I ² C*1			$N = \frac{PCLK \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	

B: Bit rate (bps)

N: BRR setting for on-chip baud rate generator ($0 \leq N \leq 255$)

PCLK: Operating frequency (MHz)

n and S: Determined by the settings of the SMR and SCMR registers as listed in Table 26.12 and Table 26.13.

Note 1. Adjust the bit rate so that the widths at high and low level of the SCL output in simple I²C mode satisfy the I²C-bus standard.

Table 26.11 Calculating Widths at High and Low Level for SCL

Mode	SCL	Formula (Result in Seconds)
I ² C	High period (minimum value)	$(N+1) \times 4 \times 2^{2n-1} \times 7 \times \frac{1}{PCLK \times 10^6}$
	Low period (minimum value)	$(N+1) \times 4 \times 2^{2n-1} \times 8 \times \frac{1}{PCLK \times 10^6}$

Table 26.12 Clock Source Settings

SMR.CKS[1:0] Bit Setting	Clock Source	n
0 0	PCLK	0
0 1	PCLK/4	1
1 0	PCLK/16	2
1 1	PCLK/64	3

Table 26.13 Base Clock Settings in Smart Card Interface Mode

SCMR.BCP2 Bit Setting	SMR.BCP[1:0] Bit Setting	Base Clock Cycles for 1-bit Period	S
0	0 0	93 clock cycles	93
0	0 1	128 clock cycles	128
0	1 0	186 clock cycles	186
0	1 1	512 clock cycles	512
1	0 0	32 clock cycles	32
1	0 1	64 clock cycles	64
1	1 0	372 clock cycles	372
1	1 1	256 clock cycles	256

Table 26.14 lists examples of N settings in the BRR register in normal asynchronous mode. Table 26.15 lists the maximum bit rate settable for each operating frequency. Examples of BRR (N) settings in clock synchronous mode and simple SPI mode are listed in Table 26.18. Examples of BRR (N) settings in smart card interface mode are listed in Table 26.20. Examples of BRR (N) settings in simple I²C mode are listed in Table 26.22. In smart card interface mode, the number of base clock cycles S in a 1-bit data transfer time can be selected. For details, refer to section 26.6.4, Receive Data Sampling Timing and Reception Margin. Table 26.16 and Table 26.19 list the maximum bit rates with external clock input.

When either the SEMR.ABCS or BGDM bit is set to 1 in asynchronous mode, the bit rate becomes twice that listed in Table 26.14. When both of those bits are set to 1, the bit rate becomes four times the listed value.

Table 26.14 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode)

Bit Rate (bps)	Operating Frequency PCLK (MHz)														
	8			9.8304			10			12			12.288		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	141	0.03	2	174	-0.26	2	177	-0.25	2	212	0.03	2	217	0.08
150	2	103	0.16	2	127	0.00	2	129	0.16	2	155	0.16	2	159	0.00
300	1	207	0.16	1	255	0.00	2	64	0.16	2	77	0.16	2	79	0.00
600	1	103	0.16	1	127	0.00	1	129	0.16	1	155	0.16	1	159	0.00
1200	0	207	0.16	0	255	0.00	1	64	0.16	1	77	0.16	1	79	0.00
2400	0	103	0.16	0	127	0.00	0	129	0.16	0	155	0.16	0	159	0.00
4800	0	51	0.16	0	63	0.00	0	64	0.16	0	77	0.16	0	79	0.00
9600	0	25	0.16	0	31	0.00	0	32	-1.36	0	38	0.16	0	39	0.00
19200	0	12	0.16	0	15	0.00	0	15	1.73	0	19	-2.34	0	19	0.00
31250	0	7	0.00	0	9	-1.70	0	9	0.00	0	11	0.00	0	11	2.40
38400	—	—	—	0	7	0.00	0	7	1.73	0	9	-2.34	0	9	0.00

Bit Rate (bps)	Operating Frequency PCLK (MHz)														
	14			16			17.2032			18			19.6608		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	248	-0.17	3	70	0.03	3	75	0.48	3	79	-0.12	3	86	0.31
150	2	181	0.16	2	207	0.16	2	223	0.00	2	233	0.16	2	255	0.00
300	2	90	0.16	2	103	0.16	2	111	0.00	2	116	0.16	2	127	0.00
600	1	181	0.16	1	207	0.16	1	223	0.00	1	233	0.16	1	255	0.00
1200	1	90	0.16	1	103	0.16	1	111	0.00	1	116	0.16	1	127	0.00
2400	0	181	0.16	0	207	0.16	0	223	0.00	0	233	0.16	0	255	0.00
4800	0	90	0.16	0	103	0.16	0	111	0.00	0	116	0.16	0	127	0.00
9600	0	45	-0.93	0	51	0.16	0	55	0.00	0	58	-0.69	0	63	0.00
19200	0	22	-0.93	0	25	0.16	0	27	0.00	0	28	1.02	0	31	0.00
31250	0	13	0.00	0	15	0.00	0	16	1.20	0	17	0.00	0	19	-1.70
38400	—	—	—	0	12	0.16	0	13	0.00	0	14	-2.34	0	15	0.00

Bit Rate (bps)	Operating Frequency PCLK (MHz)														
	20			25			30			33			40		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	88	-0.25	3	110	-0.02	3	132	0.13	3	145	0.33	3	177	-0.25
150	3	64	0.16	3	80	0.47	3	97	-0.35	3	106	0.39	3	129	0.16
300	2	129	0.16	2	162	-0.15	2	194	0.16	2	214	-0.07	3	64	0.16
600	2	64	0.16	2	80	0.47	2	97	-0.35	2	106	0.39	2	129	0.16
1200	1	129	0.16	1	162	-0.15	1	194	0.16	1	214	-0.07	2	64	0.16
2400	1	64	0.16	1	80	0.47	1	97	-0.35	1	106	0.39	1	129	0.16
4800	0	129	0.16	0	162	-0.15	0	194	0.16	0	214	-0.07	1	64	0.16
9600	0	64	0.16	0	80	0.47	0	97	-0.35	0	106	0.39	0	129	0.16
19200	0	32	-1.36	0	40	-0.76	0	48	-0.35	0	53	-0.54	0	64	0.16
31250	0	19	0.00	0	24	0.00	0	29	0.00	0	32	0.00	0	39	0.00
38400	0	15	1.73	0	19	1.73	0	23	1.73	0	26	-0.54	0	32	-1.36

Note: This is an example when the ABCS and BGDM bits in the SEMR register are 0.
When either the ABCS bit or BGDM bit is set to 1, the bit rate doubles.
When both ABCS and BGDM bits in the SEMR register are set to 1, the bit rate increases four times.

Table 26.15 Maximum Bit Rate for Each Operating Frequency (Asynchronous Mode)

PCLK (MHz)	SEMR Settings				Maximum Bit Rate (bps)	PCLK (MHz)	SEMR Settings				Maximum Bit Rate (bps)
	BGDM Bit	ABCS Bit	n	N			BGDM Bit	ABCS Bit	n	N	
8	0	0	0	0	250000	9.8304	0	0	0	0	307200
		1	0	0	500000			1	0	0	614400
	1	0	0	0			1	0	0	0	
		1	0	0	1000000			1	0	0	1228800
10	0	0	0	0	312500	12	0	0	0	0	375000
		1	0	0	625000			1	0	0	750000
	1	0	0	0			1	0	0	0	
		1	0	0	1250000			1	0	0	1500000
12.288	0	0	0	0	384000	14	0	0	0	0	437500
		1	0	0	768000			1	0	0	875000
	1	0	0	0			1	0	0	0	
		1	0	0	1536000			1	0	0	1750000
16	0	0	0	0	500000	17.2032	0	0	0	0	537600
		1	0	0	1000000			1	0	0	1075200
	1	0	0	0			1	0	0	0	
		1	0	0	2000000			1	0	0	2150400
18	0	0	0	0	562500	19.6608	0	0	0	0	614400
		1	0	0	1125000			1	0	0	1228800
	1	0	0	0			1	0	0	0	
		1	0	0	2250000			1	0	0	2457600
20	0	0	0	0	625000	25	0	0	0	0	781250
		1	0	0	1250000			1	0	0	1562500
	1	0	0	0			1	0	0	0	
		1	0	0	2500000			1	0	0	3125000
30	0	0	0	0	937500	33	0	0	0	0	1031250
		1	0	0	1875000			1	0	0	2062500
	1	0	0	0			1	0	0	0	
		1	0	0	3750000			1	0	0	4125000
40	0	0	0	0	1250000						
		1	0	0	2500000						
	1	0	0	0							
		1	0	0	5000000						

Table 26.16 Maximum Bit Rate with External Clock Input (Asynchronous Mode)

PCLK (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bps)	
		SEMR.ABCS Bit = 0	SEMR.ABCS Bit = 1
8	2.0000	125000	250000
9.8304	2.4576	153600	307200
10	2.5000	156250	312500
12	3.0000	187500	375000
12.288	3.0720	192000	384000
14	3.5000	218750	437500
16	4.0000	250000	500000
17.2032	4.3008	268800	537600
18	4.5000	281250	562500
19.6608	4.9152	307200	614400
20	5.0000	312500	625000
25	6.2500	390625	781250
30	7.5000	468750	937500
33	8.2500	515625	1031250
40	10.0000	625000	1250000

Table 26.17 Maximum Bit Rate with TMR Clock Input (Asynchronous Mode)

PCLK (MHz)	TMR Clock (MHz)	Maximum Bit Rate (bps)	
		SEMR.ABCS Bit = 0	SEMR.ABCS Bit = 1
8	4	250000	500000
9.8304	4.9152	307200	614400
10	5	312500	625000
12	6	375000	750000
12.288	6.144	384000	768000
14	7	437500	875000
16	8	500000	1000000
17.2032	8.6016	537600	1075200
18	9	562500	1125000
19.6608	9.8304	614400	1228800
20	10	625000	1250000
25	12.5	781250	1562500
30	15	937500	1875000
33	16.5	1031250	2062500
40	20	1250000	2500000

Table 26.18 Examples of BRR Settings for Various Bit Rates (Clock Synchronous Mode, Simple SPI Mode)

Bit Rate (bps)	Operating Frequency PCLK (MHz)															
	8		10		16		20		25		30		33		40	
	n	N	n	N	n	N	n	N	n	N	n	N	n	N	n	N
110																
250	3	124	3	155	3	249										
500	2	249	3	77	3	124	3	155	3	194	3	233	3	255		
1 k	2	124	2	155	2	249	3	77	3	97	3	116	3	128	3	155
2.5 k	1	199	1	249	2	99	2	124	2	155	2	187	2	205	2	249
5 k	1	99	1	124	1	199	1	249	2	77	2	93	2	102	2	124
10 k	0	199	0	249	1	99	1	124	1	155	1	187	1	205	1	249
25 k	0	79	0	99	0	159	0	199	0	249	1	74	1	82	1	99
50 k	0	39	0	49	0	79	0	99	0	124	0	149	0	164	1	49
100 k	0	19	0	24	0	39	0	49	0	62	0	74	0	82	0	99
250 k	0	7	0	9	0	15	0	19	0	24	0	29	0	32	0	39
500 k	0	3	0	4	0	7	0	9	—	—	0	14	—	—	0	19
1 M	0	1			0	3	0	4	—	—			—	—	0	9
2.5 M			0	0*1			0	1			0	2			0	3
5 M							0	0*1							0	1
7.5 M											0	0*1				

Blank cell: Cannot be set since the bit rate error exceeds 5%.

—: Can be set, but a bit rate error of 1 to 5% will occur.

Note 1. Continuous transmission or reception is not possible. After transmitting/receiving one frame of data, there is an interval of a 1-bit period before starting transmitting/receiving the next frame of data. The output of the synchronization clock is stopped for a 1-bit period. For this reason, it takes 9 bits worth of time to transfer one frame (8 bits) of data, and the average transfer rate is 8/9 times the bit rate.

Table 26.19 Maximum Bit Rate with External Clock Input (Clock Synchronous Mode, Simple SPI Mode)

PCLK (MHz)	External Input Clock (MHz)	Maximum Bit Rate (Mbps)
8	1.3333	1.3333
10	1.6667	1.6667
12	2.0000	2.0000
14	2.3333	2.3333
16	2.6667	2.6667
18	3.0000	3.0000
20	3.3333	3.3333
25	4.1667	4.1667
30	5.0000	5.0000
33	5.5000	5.5000
40	6.6667	6.6667

Table 26.20 Examples of BRR Settings for Various Bit Rates (Smart Card Interface Mode, S = 372)

Bit Rate (bps)	PCLK (MHz)	n	N	Error (%)
9600	7.1424	0	0	0.00
	10.00	0	1	-30.00
	10.7136	0	1	-25.00
	13.00	0	1	-8.99
	14.2848	0	1	0.00
	16.00	0	1	12.01
	18.00	0	2	-15.99
	20.00	0	2	-6.66
	25.00	0	3	-12.49
	30.00	0	3	5.01
	33.00	0	4	-7.59
	40.00	0	5	-6.66

Table 26.21 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 32)

PCLK (MHz)	Maximum Bit Rate (bps)	n	N
10.00	156250	0	0
10.7136	167400	0	0
13.00	203125	0	0
16.00	250000	0	0
18.00	281250	0	0
20.00	312500	0	0
25.00	390625	0	0
30.00	468750	0	0
33.00	515625	0	0
40.00	625000	0	0

Table 26.22 Examples of BRR Settings for Various Bit Rates (Simple I²C Mode)

Bit Rate (bps)	Operating Frequency PCLK (MHz)														
	8			10			16			20			25		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
10 k	0	24	0.0	0	30	0.8	0	49	0.0	0	62	−0.8	0	77	0.2
25 k	0	9	0.0	0	12	−3.8	1	4	0.0	0	24	0.0	0	30	0.8
50 k	0	4	0.0	0	5	4.2	0	9	0.0	0	12	−3.8	2	0	−2.3
100 k	0	2	−16.7	1	0	−21.9	0	4	0.0	0	6	−10.7	1	1	−2.3
250 k	0	0	0.0	0	0	25.0	0	1	0.0	0	2	−16.7	0	2	4.2
350 k										0	1	−10.7	0	2	−25.6

Bit Rate (bps)	Operating Frequency PCLK (MHz)								
	30			33			40		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
10 k	0	93	−0.3	0	102	0.1	0	124	0.0
25 k	0	37	−1.3	0	40	0.6	0	49	0.0
50 k	0	18	−1.3	0	20	−1.8	0	24	0.0
100 k	0	9	−6.3	0	10	−6.3	0	12	−3.8
250 k	1	0	−6.3	1	0	3.1	0	4	0.0
350 k	0	2	−10.7	0	2	−1.8	1	0	−10.7

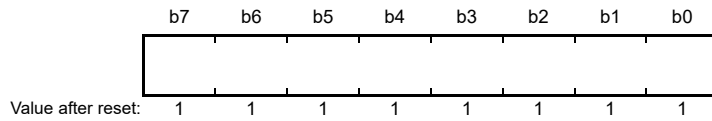
Table 26.23 Minimum Widths at High and Low Level for SCL at Various Bit Rates (Simple I²C Mode)

Bit Rate (bps)	Operating Frequency PCLK (MHz)											
	8			10			16			20		
	n	N	Min. Widths at High/Low Level for SCL (μs)	n	N	Min. Widths at High/Low Level for SCL (μs)	n	N	Min. Widths at High/Low Level for SCL (μs)	n	N	Min. Widths at High/Low Level for SCL (μs)
10 k	0	24	43.75/50.00	0	30	43.40/49.60	0	49	43.75/50.00	0	62	44.10/50.40
25 k	0	9	17.50/20.00	0	12	18.20/20.80	1	4	17.50/20.00	0	24	17.50/20.00
50 k	0	4	8.75/10.00	0	5	8.40/9.60	0	9	8.75/10.00	0	12	9.10/10.40
100 k	0	2	5.25/6.00	1	0	5.60/6.40	0	4	4.38/5.00	0	6	4.90/5.60
250 k	0	0	1.75/2.00	0	0	1.40/1.60	0	1	1.75/2.00	0	2	2.10/2.40
350 k										0	1	1.40/1.60

Bit Rate (bps)	Operating Frequency PCLK (MHz)											
	25			30			33			40		
	n	N	Min. Widths at High/Low Level for SCL (μs)	n	N	Min. Widths at High/Low Level for SCL (μs)	n	N	Min. Widths at High/Low Level for SCL (μs)	n	N	Min. Widths at High/Low Level for SCL (μs)
10 k	0	77	43.68/49.92	0	93	43.87/50.13	0	102	43.70/49.94	0	124	43.75/50.00
25 k	0	30	17.36/19.84	0	37	17.73/20.27	0	40	17.39/19.88	0	49	17.50/20.00
50 k	2	0	8.96/10.24	0	18	8.87/10.13	0	20	8.91/10.18	0	24	8.75/10.00
100 k	1	1	4.48/5.12	0	9	4.67/5.33	0	10	4.67/5.33	0	12	4.55/5.20
250 k	0	2	1.68/1.92	1	0	1.87/2.13	1	0	1.70/1.94	0	4	1.75/2.00
350 k	0	2	1.68/1.92	0	2	1.40/1.60	0	2	1.27/1.45	1	0	1.40/1.60

26.2.12 Modulation Duty Register (MDDR)

Address(es): SCI1.MDDR 0008 A032h, SCI5.MDDR 0008 A0B2h, SCI6.MDDR 0008 A0D2h, SCI12.MDDR 0008 B312h



The MDDR register corrects the bit rate adjusted by the BRR register.

When the SEMR.BRME bit is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of the MDDR register ($M/256$). The relationship between the MDDR register setting (M) and the bit rate (B) is given in Table 26.24.

The range of the value that can be set in the MDDR register is from 80h to FFh. A value other than these cannot be set. The MDDR register is writable only when the TE and RE bits in the SCR register are 0.

Table 26.24 Relationship between MDDR Setting (M) and Bit Rate (B) When Bit Rate Modulation Function is Used

Mode	SEMR Settings		BRR Setting	Error (%)
	BGDM Bit	ABCS Bit		
Asynchronous, multi-processor communication	0	0	$N = \frac{PCLK \times 10^6}{64 \times 2^{2n-1} \times \frac{256}{M} \times B} - 1$	$\text{Error} = \left\{ \frac{PCLK \times 10^6}{B \times 64 \times 2^{2n-1} \times \frac{256}{M} \times (N+1)} - 1 \right\} \times 100$
	1	0	$N = \frac{PCLK \times 10^6}{32 \times 2^{2n-1} \times \frac{256}{M} \times B} - 1$	$\text{Error} = \left\{ \frac{PCLK \times 10^6}{B \times 32 \times 2^{2n-1} \times \frac{256}{M} \times (N+1)} - 1 \right\} \times 100$
	0	1		
	1	1	$N = \frac{PCLK \times 10^6}{16 \times 2^{2n-1} \times \frac{256}{M} \times B} - 1$	$\text{Error} = \left\{ \frac{PCLK \times 10^6}{B \times 16 \times 2^{2n-1} \times \frac{256}{M} \times (N+1)} - 1 \right\} \times 100$
Clock synchronous, simple SPI*1			$N = \frac{PCLK \times 10^6}{8 \times 2^{2n-1} \times \frac{256}{M} \times B} - 1$	
Smart card interface			$N = \frac{PCLK \times 10^6}{S \times 2^{2n+1} \times \frac{256}{M} \times B} - 1$	$\text{Error} = \left\{ \frac{PCLK \times 10^6}{B \times S \times 2^{2n+1} \times \frac{256}{M} \times (N+1)} - 1 \right\} \times 100$
Simple I ² C*2			$N = \frac{PCLK \times 10^6}{64 \times 2^{2n-1} \times \frac{256}{M} \times B} - 1$	

B: Bit rate (bps)

M: MDDR setting ($128 \leq MDDR \leq 256$)

N: BRR setting for baud rate generator ($0 \leq N \leq 255$)

PCLK: Operating frequency (MHz)

n and S: Determined by the settings of the SMR and SCMR registers as listed in Table 26.12 and Table 26.13, section 26.2.11, Bit Rate Register (BRR).

Note 1. Do not use this function in clock synchronous mode and in the highest speed settings in simple SPI mode (SMR.CKS[1:0] = 00b, SCR.CKE[1] = 0, and BRR = 0).

Note 2. Adjust the bit rate so that the widths at high and low level of the SCL output in simple I²C mode satisfy the I²C-bus standard.

Smaller settings of the SMR.CKS[1:0] bits and larger settings of the BRR register reduce difference in the length of the 1-bit period.

26.2.13 Serial Extended Mode Register (SEMR)

Address(es): SCI1.SEMR 0008 A027h, SCI5.SEMR 0008 A0A7h, SCI6.SEMR 0008 A0C7h, SCI12.SEMR 0008 B307h

b7	b6	b5	b4	b3	b2	b1	b0
RXDESEL	BGDM	NFEN	ABCS	—	BRME	—	ACS0
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ACS0	Asynchronous Mode Clock Source Select	(Valid only in asynchronous mode) 0: External clock input 1: Logical AND of two compare matches output from TMR (valid for SCI5, SCI6, and SCI12 only) Available compare match output varies per SCI channel.	R/W*1
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	BRME	Bit Rate Modulation Enable	0: Bit rate modulation function is disabled. 1: Bit rate modulation function is enabled.	R/W*1
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ABCS	Asynchronous Mode Base Clock Select	(Valid only in asynchronous mode) 0: Selects 16 base clock cycles for 1-bit period. 1: Selects 8 base clock cycles for 1-bit period.	R/W*1
b5	NFEN	Digital Noise Filter Function Enable	(In asynchronous mode) 0: Noise cancellation function for the RXDn input signal is disabled. 1: Noise cancellation function for the RXDn input signal is enabled. (in simple I ² C mode) 0: Noise cancellation function for the SSCLn and SSDAn input signals is disabled. 1: Noise cancellation function for the SSCLn and SSDAn input signals is enabled. The NFEN bit should be 0 in any mode other than above.	R/W*1
b6	BGDM	Baud Rate Generator Double-Speed Mode Select	(Only valid the SCR.CKE[1] bit is 0 in asynchronous mode) 0: Baud rate generator outputs the clock with normal frequency. 1: Baud rate generator outputs the clock with doubled frequency.	R/W*1
b7	RXDESEL	Asynchronous Start Bit Edge Detection Select	(Valid only in asynchronous mode) 0: The low level on the RXDn pin is detected as the start bit. 1: A falling edge on the RXDn pin is detected as the start bit.	R/W*1

Note 1. Writable only when the SCR.TE bit is 0 and the SCR.RE bit is 0 (both serial transmission and reception are disabled).

The SEMR register is used to select a clock source for 1-bit period in asynchronous mode or a detection method of the start bit.

ACS0 Bit (Asynchronous Mode Clock Source Select)

Selects the clock source in the asynchronous mode.

The ACS0 bit is valid in asynchronous mode (SMR.CM bit = 0) and when an external clock input is selected (SCR.CKE[1:0] bits = 10b or 11b). This bit is used to select an external clock input or the logical AND of compare matches output from the internal TMR.

Set the ACS0 bit to 0 in other than asynchronous mode.

For SCI5, SCI6, and SCI12, the TMO_n output (n = 0 to 3) of TMR units 0 and 1 can be set as the base clock source. Refer to Table 26.25 for details.

The ACS0 bit for SCI1 is reserved. The write value to this bit for SCI1 should be 0.

Table 26.25 Correspondence between SCI Channels and Compare Match Outputs

SCI	TMR	Compare Match Output
SCI5	Unit 0	TMO0, TMO1
SCI6	Unit 1	TMO2, TMO3
SCI12	Unit 0	TMO0, TMO1

Figure 26.4 shows a setting example of when TMO0 and TMO1 in the TMR unit 0 are selected for output.

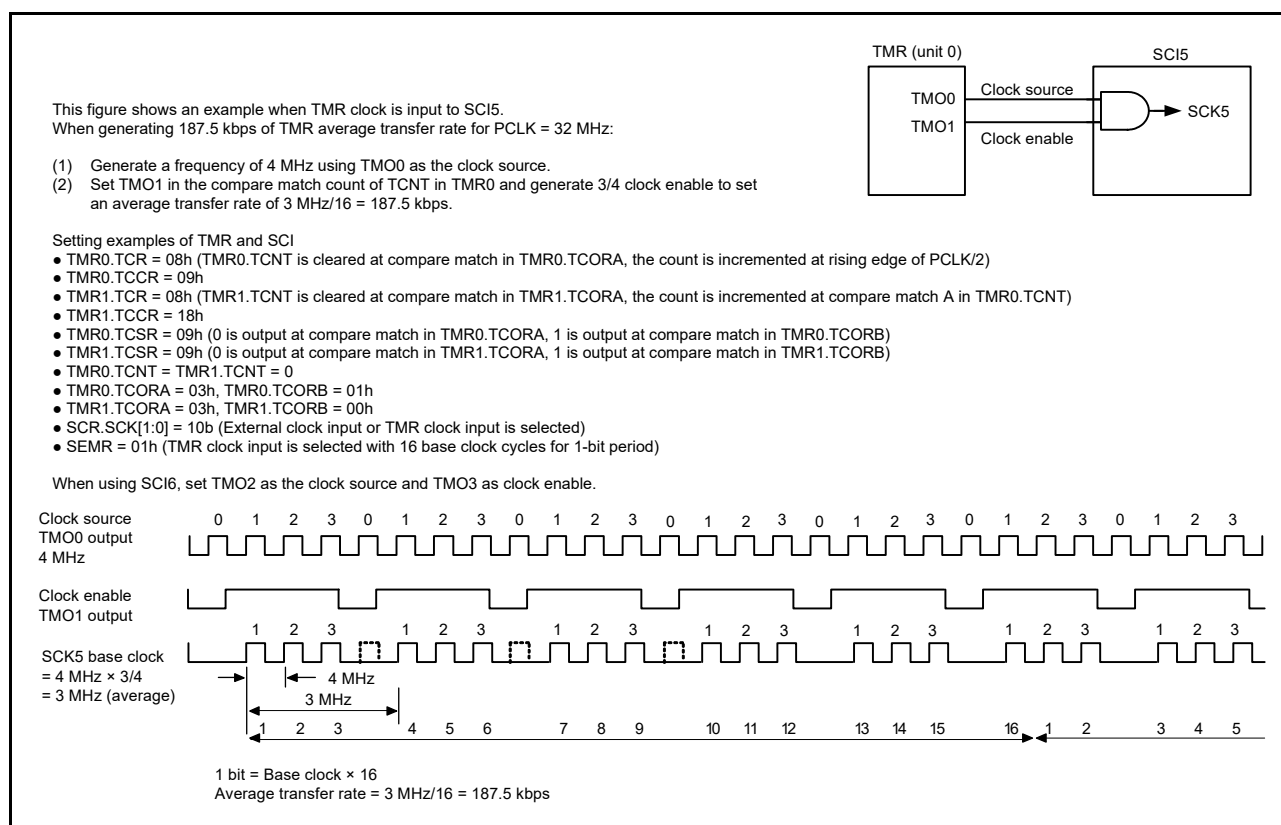


Figure 26.4 Example of Average Transfer Rate Setting When TMR Clock is Input

BRME Bit (Bit Rate Modulation Enable)

This bit enables and disables the bit rate modulation function. The bit rate generated by on-chip baud rate generator is evenly corrected when this function is enabled.

NFEN Bit (Digital Noise Filter Function Enable)

This bit enables or disables the digital noise filter function.

When the function is enabled, noise cancellation is applied to the RXDn input signal in asynchronous mode, and noise cancellation is applied to the SSDAn and SSCLn input signals in simple I²C mode.

In any mode other than above, set the NFEN bit to 0 to disable the digital noise filter function.

When the function is disabled, input signals are transferred as is, as internal signals.

BGDM Bit (Baud Rate Generator Double-Speed Mode Select)

Selects the cycle of output clock for the baud rate generator.

This bit is valid when the on-chip baud rate generator is selected as the clock source (SCR.CKE[1] = 0) in asynchronous mode (SMR.CM = 0). For the clock output from the baud rate generator, either normal or doubled frequency can be selected. The base clock is generated by the clock output from the baud rate generator. When the BGDM bit is set to 1, the base clock cycle is halved and the bit rate is doubled.

Set this bit to 0 in modes other than asynchronous mode.

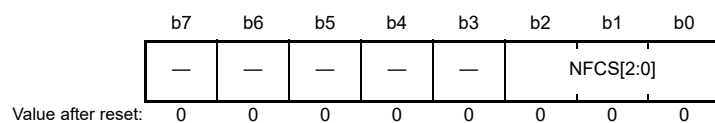
RXDESEL Bit (Asynchronous Start Bit Edge Detection Select)

Selects the detection method of the start bit for reception in asynchronous mode. When a break occurs, data receiving operation depends on the settings of this bit. Set this bit to 1 when reception should be stopped while a break occurs or when reception should be started without retaining the RXDn pin input at high level for the period of one data frame or longer after completion of the break.

Set this bit to 0 in modes other than asynchronous mode.

26.2.14 Noise Filter Setting Register (SNFR)

Address(es): SCI1.SNFR 0008 A028h, SCI5.SNFR 0008 A0A8h, SCI6.SNFR 0008 A0C8h, SCI12.SNFR 0008 B308h



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	NFCS[2:0]	Noise Filter Clock Select	In asynchronous mode, the standard setting for the base clock is as follows. b2 b0 0 0 0: The clock signal divided by 1 is used with the noise filter. In simple I²C mode, the standard settings for the clock source of the on-chip baud rate generator selected by the SMR.CKS[1:0] bits are given below. b2 b0 0 0 1: The clock signal divided by 1 is used with the noise filter. 0 1 0: The clock signal divided by 2 is used with the noise filter. 0 1 1: The clock signal divided by 4 is used with the noise filter. 1 0 0: The clock signal divided by 8 is used with the noise filter. Settings other than above are prohibited.	R/W*1
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

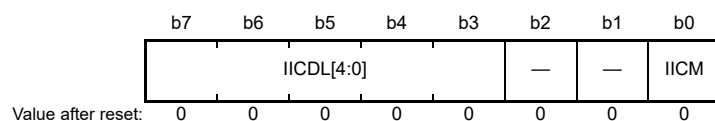
Note 1. Writing to these bits is only possible when the RE and TE bits in the SCR register are 0 (serial reception and transmission disabled).

NFCS[2:0] Bits (Noise Filter Clock Select)

These bits select the sampling clock for the digital noise filter. To use the noise filter in asynchronous mode, set these bits to 000b. In simple I²C mode, set the bits to a value in the range from 001b to 100b.

26.2.15 I²C Mode Register 1 (SIMR1)

Address(es): SCI1.SIMR1 0008 A029h, SCI5.SIMR1 0008 A0A9h, SCI6.SIMR1 0008 A0C9h, SCI12.SIMR1 0008 B309h



Bit	Symbol	Bit Name	Description	R/W
b0	IICM	Simple I ² C Mode Select	SMIF IICM 0 0: Asynchronous mode, Multi-processor mode, Clock synchronous mode (in asynchronous mode, synchronous, or simple SPI mode) 0 1: Simple I ² C mode 1 0: Smart card interface mode 1 1: Setting prohibited.	R/W*1
b2, b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7 to b3	IICDL[4:0]	SSDA Output Delay Select	(Cycles below are of the clock signal from the on-chip baud rate generator.) b7 b3 0 0 0 0 0: No output delay 0 0 0 0 1: 0 to 1 cycle 0 0 0 1 0: 1 to 2 cycles 0 0 0 1 1: 2 to 3 cycles 0 0 1 0 0: 3 to 4 cycles 0 0 1 0 1: 4 to 5 cycles : : 1 1 1 1 0: 29 to 30 cycles 1 1 1 1 1: 30 to 31 cycles	R/W*1

Note 1. Writing to these bits is only possible when the RE and TE bits in the SCR register are 0 (both serial transmission and reception are disabled).

SIMR1 is used to select simple I²C mode and the number of delay stages for the SSDA output.

IICM Bit (Simple I²C Mode Select)

In conjunction with the SCMR.SMIF bit, this bit selects the operating mode.

IICDL[4:0] Bits (SSDA Output Delay Select)

These bits are used to set a delay for output on the SSDAn pin relative to the falling edge of the output on the SSCLn pin. The available delay settings range from no delay to 31 cycles, with the clock signal from the on-chip baud rate generator as the base. The signal obtained by frequency-dividing PCLK by the divisor set in the SMR.CKS[1:0] bits is supplied as the clock signal from the on-chip baud rate generator. Set these bits to 00000b unless operation is in simple I²C mode. In simple I²C mode, set the bits to a value in the range from 00001b to 11111b.

26.2.16 I²C Mode Register 2 (SIMR2)

Address(es): SCI1.SIMR2 0008 A02Ah, SCI5.SIMR2 0008 A0AAh, SCI6.SIMR2 0008 A0CAh, SCI12.SIMR2 0008 B30Ah

b7	b6	b5	b4	b3	b2	b1	b0
—	—	IICACK T	—	—	—	IICCS C	IICINT M
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	IICINTM	I ² C Interrupt Mode Select	0: Use ACK/NACK interrupts. 1: Use reception and transmission interrupts.	R/W*1
b1	IICCS	Clock Synchronization	0: No synchronization with the clock signal 1: Synchronization with the clock signal	R/W*1
b4 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b5	IICACKT	ACK Transmission Data	0: ACK transmission 1: NACK transmission and reception of ACK/NACK	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Writing to these bits is only possible when the RE and TE bits in the SCR register are 0 (serial reception and transmission disabled).

SIMR2 is used to select how reception and transmission are controlled in simple I²C mode.

IICINTM Bit (I²C Interrupt Mode Select)

This bit selects the sources of interrupt requests in simple I²C mode.

IICCS Bit (Clock Synchronization)

Set the IICCS bit to 1 if the internally generated SSCLn clock signal is to be synchronized when the SSCLn pin has been placed at the low level in the case of a wait inserted by the other device, etc.

The SSCLn clock signal is not synchronized if the IICCS bit is 0. The SSCLn clock signal is generated in accord with the rate selected in the BRR regardless of the level being input on the SSCLn pin.

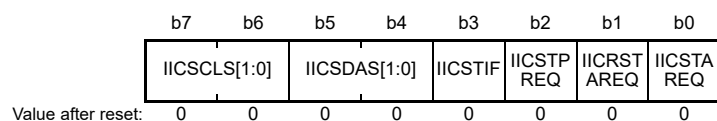
Set the IICCS bit to 1 except during debugging.

IICACKT Bit (ACK Transmission Data)

Transmitted data contains ACK bits. Set this bit to 1 when ACK and NACK bits are received.

26.2.17 I²C Mode Register 3 (SIMR3)

Address(es): SCI1.SIMR3 0008 A02Bh, SCI5.SIMR3 0008 A0ABh, SCI6.SIMR3 0008 A0CBh, SCI12.SIMR3 0008 B30Bh



Bit	Symbol	Bit Name	Description	R/W
b0	IICSTAREQ	Start Condition Generation	0: A start condition is not generated. 1: A start condition is generated.*1, *3, *4, *5	R/W
b1	IICRSTAREQ	Restart Condition Generation	0: A restart condition is not generated. 1: A restart condition is generated.*2, *3, *4, *5	R/W
b2	IICSTPREQ	Stop Condition Generation	0: A stop condition is not generated. 1: A stop condition is generated.*2, *3, *4, *5	R/W
b3	IICSTIF	Issuing of Start, Restart, or Stop Condition Completed Flag	0: There are no requests for generating conditions or a condition is being generated. 1: A start, restart, or stop condition is completely generated.	R/W
b5, b4	IICSDAS[1:0]	SSDA Output Select	b5 b4 0 0: Serial data output 0 1: Generate a start, restart, or stop condition. 1 0: Output the low level on the SSDAn pin. 1 1: Place the SSDAn pin in the high-impedance state.	R/W
b7, b6	IICSCLS[1:0]	SSCL Output Select	b7 b6 0 0: Serial clock output 0 1: Generate a start, restart, or stop condition. 1 0: Output the low level on the SSCLn pin. 1 1: Place the SSCLn pin in the high-impedance state.	R/W

Note 1. Generate a start condition only when the SSCLn and SSDAn pins are both high (the corresponding bits in the corresponding PIDR registers are 1).

Note 2. Generate a restart or stop condition only when the SSCLn pin is low (the corresponding bit in the PIDR register is 0).

Note 3. Do not set more than one from among the IICSTAREQ, IICRSTAREQ, and IICSTPREQ bits to 1 at a given time.

Note 4. Execute the generation of a condition after the value of the IICSTIF flag is 0.

Note 5. Do not write 0 to this bit while it is 1. Generation of a condition is suspended by writing 0 to this bit while it is 1.

SIMR3 is used to control the simple I²C mode start and stop conditions, and to hold the SSDAn and SSCLn pins at fixed levels.

IICSTAREQ Bit (Start Condition Generation)

When a start condition is to be generated, set both the IICSDAS[1:0] and IICSCLS[1:0] bits to 01b as well as setting the IICSTAREQ bit to 1.

[Setting condition]

- Writing 1 to the bit

[Clearing condition]

- Completion of generation of the start condition

IICRSTAREQ Bit (Restart Condition Generation)

When a restart condition is to be generated, set both the IICSDAS[1:0] and IICSCLS[1:0] bits to 01b as well as setting the IICRSTAREQ bit to 1.

[Setting condition]

- Writing 1 to the bit

[Clearing condition]

- Completion of generation of the restart condition

IICSTPREQ Bit (Stop Condition Generation)

When a stop condition is to be generated, set both the IICSDAS[1:0] and IICSCLS[1:0] bits to 01b as well as setting the IICSTPREQ bit to 1.

[Setting condition]

- Writing 1 to the bit

[Clearing condition]

- Completion of generation of the stop condition

IICSTIF Flag (Issuing of Start, Restart, or Stop Condition Completed Flag)

After generating a condition, this bit indicates that the generation is completed. When using the IICSTAREQ, IICRSTAREQ, or IICSTPREQ bit to cause generation of a condition, do so after setting the IICSTIF flag to 0.

When the IICSTIF flag is 1 while an interrupt request is enabled by setting the SCR.TEIE bit, an STI request is output.

[Setting condition]

- Completion of the generation of a start, restart, or stop condition (however, in cases where this conflicts with any of the conditions for the flag becoming 0 listed below, the other condition takes precedence)

[Clearing conditions]

- Writing 0 to the bit (confirm that the IICSTIF flag is 0 before doing so)
- Writing 0 to the SIMR1.IICM bit (when operation is not in simple I²C mode)
- Writing 0 to the SCR.TE bit

IICSDAS[1:0] Bits (SSDA Output Select)

These bits control output from the SSDAn pin.

Set the IICSDAS[1:0] and IICSCLS[1:0] bits to the same value during normal operations.

IICSCLS[1:0] Bits (SSCL Output Select)

These bits control output from the SSCLn pin.

Set the IICSCLS[1:0] and IICSDAS[1:0] bits to the same value during normal operations.

26.2.18 I²C Status Register (SISR)

Address(es): SCI1.SISR 0008 A02Ch, SCI5.SISR 0008 A0ACh, SCI6.SISR 0008 A0CCh, SCI12.SISR 0008 B30Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	IICACK R
Value after reset:	0	0	x	x	0	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	IICACKR	ACK Reception Data Flag	0: ACK received 1: NACK received	R/W*1
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	—	Reserved	The read value is undefined.	R
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5, b4	—	Reserved	The read value is undefined.	R
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only 0 can be written to this bit, to clear the flag.

SISR is used to monitor state in relation to simple I²C mode.

IICACKR Flag (ACK Reception Data Flag)

Received ACK and NACK bits can be read from this bit.

The IICACKR flag is updated at the rising of SSCLn clock for the ACK/NACK receiving bit.

26.2.19 SPI Mode Register (SPMR)

Address(es): SCI1.SPMR 0008 A02Dh, SCI5.SPMR 0008 A0ADh, SCI6.SPMR 0008 A0CDh, SCI12.SPMR 0008 B30Dh

b7	b6	b5	b4	b3	b2	b1	b0
CKPH	CKPOL	—	MFF	—	MSS	CTSE	SSE
Value after reset: 0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SSE	SSn# Pin Function Enable	0: SSn# pin function is disabled. 1: SSn# pin function is enabled.	R/W*1
b1	CTSE	CTS Enable	0: CTS function is disabled (RTS output function is enabled). 1: CTS function is enabled.	R/W*1
b2	MSS	Master Slave Select	0: Transmission is through the SMOSIn pin and reception is through the SMISOn pin (master mode). 1: Reception is through the SMOSIn pin and transmission is through the SMISOn pin (slave mode).	R/W*1
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	MFF	Mode Fault Flag	0: No mode fault error 1: Mode fault error	R/W*2
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	CKPOL	Clock Polarity Select	0: Clock polarity is not inverted. 1: Clock polarity is inverted.	R/W*1
b7	CKPH	Clock Phase Select	0: Clock is not delayed. 1: Clock is delayed.	R/W*1

Note 1. Writing to these bits is only possible when the RE and TE bits in the SCR register are 0 (both serial transmission and reception are disabled).

Note 2. Only 0 can be written to these bits, which clears the flag.

SPMR is used to select the extension settings in asynchronous and clock synchronous modes.

SSE Bit (SSn# Pin Function Enable)

Set this bit to 1 if the SSn# pin is to be used in control of transmission and reception (in simple SPI mode). Set this bit to 0 in any other mode. Furthermore, even for usage in simple SPI mode, the SSn# pin on the master side is not required to control reception and transmission when master mode (SCR.CKE[1:0] = 00b and MSS = 0) is selected and there is a single master, so the setting for the SSE bit is 0. Do not set both the SSE and CTSE bits to enabled (even if this setting is made, operation is the same as that when these bits are set to 0).

CTSE Bit (CTS Enable)

Set this bit to 1 if the SSn# pin is to be used for inputting of the CTS control signal to control of transmission and reception. The RTS signal is output when this bit is set to 0. Set this bit to 0 in smart card interface mode, simple SPI mode, and simple I²C mode. Do not set both the CTSE and SSE bits to enabled (even if this setting is made, operation is the same as that when these bits are set to 0).

MSS Bit (Master Slave Select)

This bit selects between master and slave operation in simple SPI mode. When the MSS bit is set to 1, data is received through the SMOSIn pin and transmitted through the SMISOn pin.

Set this bit to 0 in modes other than simple SPI mode.

MFF Flag (Mode Fault Flag)

This bit indicates mode fault errors.

In a multi-master configuration, determine the mode fault error occurrence by reading the MFF flag.

[Setting condition]

- Input on the SSn# pin being at the low level during master operation in simple SPI mode (SSE bit = 1 and MSS bit = 0)

[Clearing condition]

- Writing 0 to the bit after it was read as 1

CKPOL Bit (Clock Polarity Select)

This bit selects the polarity of the clock signal output through the SCKn pin. Refer to Figure 26.58 for details.

Set the bit to 0 in other than simple SPI mode and clock synchronous mode.

CKPH Bit (Clock Phase Select)

This bit selects the phase of the clock signal output through the SCKn pin. Refer to Figure 26.58 for details.

Set the bit to 0 in other than simple SPI mode and clock synchronous mode.

26.2.20 Extended Serial Module Enable Register (ESMER)

Address(es): SCI12.ESMER 0008 B320h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	ESME
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ESME	Extended Serial Mode Enable	0: The extended serial mode is disabled. 1: The extended serial mode is enabled.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

ESME Bit (Extended Serial Mode Enable)

When the ESME bit is 1, the facilities of the extended serial mode control section are enabled.

When the ESME bit is 0, the extended serial mode control section is initialized.

Table 26.26 Settings of the ESME Bit and Timer Operation Mode

ESME Bit	Timer Mode	Break Field Low Width Determination Mode	Break Field Low Width Output Mode
0	Available*1	Not available	Not available
1	Available	Available	Available

Note 1. Operation is only possible with PCLK selected.

26.2.21 Control Register 0 (CR0)

Address(es): SCI12.CR0 0008 B321h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	BRME	RXDSF	SFSF	—
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	SFSF	Start Frame Status Flag	0: Start Frame detection function is disabled. 1: Start Frame detection function is enabled.	R
b2	RXDSF	RXDX12 Input Status Flag	0: RXDX12 input is enabled. 1: RXDX12 input is disabled.	R
b3	BRME	Bit Rate Measurement Enable	0: Measurement of bit rate is disabled. 1: Measurement of bit rate is enabled.	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

26.2.22 Control Register 1 (CR1)

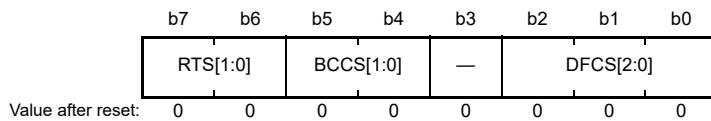
Address(es): SCI12.CR1 0008 B322h

b7	b6	b5	b4	b3	b2	b1	b0
PIBS[2:0]			PIBE	CF1DS[1:0]		CF0RE	BFE
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	BFE	Break Field Enable	0: Break Field detection is disabled. 1: Break Field detection is enabled.	R/W
b1	CF0RE	Control Field 0 Reception Enable	0: Reception of Control Field 0 is disabled. 1: Reception of Control Field 0 is enabled.	R/W
b3, b2	CF1DS[1:0]	Control Field 1 Data Register Select	b3 b2 0 0: Selects comparison with the value in the PCF1DR register. 0 1: Selects comparison with the value in the SCF1DR register. 1 0: Selects comparison with the values in the PCF1DR and SCF1DR registers. 1 1: Setting prohibited.	R/W
b4	PIBE	Priority Interrupt Bit Enable	0: The priority interrupt bit is disabled. 1: The priority interrupt bit is enabled.	R/W
b7 to b5	PIBS[2:0]	Priority Interrupt Bit Select	b7 b5 0 0 0: 0th bit of Control Field 1 0 0 1: 1st bit of Control Field 1 0 1 0: 2nd bit of Control Field 1 0 1 1: 3rd bit of Control Field 1 1 0 0: 4th bit of Control Field 1 1 0 1: 5th bit of Control Field 1 1 1 0: 6th bit of Control Field 1 1 1 1: 7th bit of Control Field 1	R/W

26.2.23 Control Register 2 (CR2)

Address(es): SCI12.CR2 0008 B323h



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	DFCS[2:0]	RXDX12 Signal Digital Filter Clock Select	b2 b0 0 0 0: Filter is disabled. 0 0 1: Filter clock is base clock*1, *2 0 1 0: Filter clock is PCLK/8 0 1 1: Filter clock is PCLK/16 1 0 0: Filter clock is PCLK/32 1 0 1: Filter clock is PCLK/64 1 1 0: Filter clock is PCLK/128 1 1 1: Setting prohibited	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5, b4	BCCS[1:0]	Bus Collision Detection Clock Select	• When SEMR.BGDM = 0 or SEMR.BGDM = 1 and SMR.CKS[1:0] = a value other than 00b b5 b4 0 0: Base clock 0 1: Base clock frequency divided by 2 1 0: Base clock frequency divided by 4 1 1: Setting prohibited • When SEMR.BGDM = 1 and SMR.CKS[1:0] = 00b b5 b4 0 0: Base clock frequency divided by 2 0 1: Base clock frequency divided by 4 1 0: Setting prohibited 1 1: Setting prohibited	R/W
b7, b6	RTS[1:0]	RXDX12 Reception Sampling Timing Select	• When SCI12.SEMR.ABCS = 0 b7 b6 0 0: Rising edge of the 8th cycle of base clock 0 1: Rising edge of the 10th cycle of base clock 1 0: Rising edge of the 12th cycle of base clock 1 1: Rising edge of the 14th cycle of base clock • When SCI12.SEMR.ABCS = 1 b7 b6 0 0: Rising edge of the 4th cycle of base clock 0 1: Rising edge of the 5th cycle of base clock 1 0: Rising edge of the 6th cycle of base clock 1 1: Rising edge of the 7th cycle of base clock	R/W

Note: The period of the base clock is 1/16 of a single bit period when the SCI12.SEMR.ABCS is 0, and 1/8 of a single bit period when the SCI12.SEMR.ABCS is 1.

Note 1. To use the base clock, set the SCI12.SCR.TE bit to 1.

Note 2. The base clock divided by 2 is the filter clock when the SEMR.BGDM bit is 1 and the SMR.CKS[1:0] bits are 00b.

26.2.24 Control Register 3 (CR3)

Address(es): SCI12.CR3 0008 B324h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	SDST
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SDST	Start Frame Detection Start	0: Detection of Start Frame is not performed. 1: Detection of Start Frame is performed.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SDST Bit (Start Frame Detection Start)

Detection of a Start Frame begins when this bit is set to 1. The bit is read as 0.

26.2.25 Port Control Register (PCR)

Address(es): SCI12.PCR 0008 B325h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	SHARPS	—	—	RDXPS	TXDXPS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TXDXPS	TXDX12 Signal Polarity Select	0: The polarity of TXDX12 signal is not inverted for output. 1: The polarity of TXDX12 signal is inverted for output.	R/W
b1	RDXPS	RDX12 Signal Polarity Select	0: The polarity of RDX12 signal is not inverted for input. 1: The polarity of RDX12 signal is inverted for input.	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	SHARPS	TXDX12/RDX12 Pin Multiplexing Select	0: The TXDX12 and RDX12 pins are independent. 1: The TXDX12 and RDX12 signals are multiplexed on the same pin.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SHARPS Bit (TXDX12/RDX12 Pin Multiplexing Select)

When this bit is set to 1, the TXDX12 and RDX12 signals are multiplexed on the same pin so that half-duplex communications become possible.

26.2.26 Interrupt Control Register (ICR)

Address(es): SCI12.ICR 0008 B326h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	AEDIE	BCDIE	PIBDIE	CF1MIE	CF0MIE	BFDIE
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	BFDIE	Break Field Low Width Detected Interrupt Enable	0: Interrupts on detection of the low width for a Break Field are disabled. 1: Interrupts on detection of the low width for a Break Field are enabled.	R/W
b1	CF0MIE	Control Field 0 Match Detected Interrupt Enable	0: Interrupts on detection of a match with Control Field 0 are disabled. 1: Interrupts on detection of a match with Control Field 0 are enabled.	R/W
b2	CF1MIE	Control Field 1 Match Detected Interrupt Enable	0: Interrupts on detection of a match with Control Field 1 are disabled. 1: Interrupts on detection of a match with Control Field 1 are enabled.	R/W
b3	PIBDIE	Priority Interrupt Bit Detected Interrupt Enable	0: Interrupts on detection of the priority interrupt bit are disabled. 1: Interrupts on detection of the priority interrupt bit are enabled.	R/W
b4	BCDIE	Bus Collision Detected Interrupt Enable	0: Interrupts on detection of a bus collision are disabled. 1: Interrupts on detection of a bus collision are enabled.	R/W
b5	AEDIE	Valid Edge Detected Interrupt Enable	0: Interrupts on detection of a valid edge are disabled. 1: Interrupts on detection of a valid edge are enabled.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

26.2.27 Status Register (STR)

Address(es): SCI12.STR 0008 B327h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	AEDF	BCDF	PIBDF	CF1MF	CF0MF	BFDF
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	BFDF	Break Field Low Width Detection Flag	[Setting conditions] • Detection of the low width for a Break Field • Completion of the output of the low width for a Break Field • Underflow of the timer [Clearing condition] • Writing 1 to the STCR.BFDCL bit	R
b1	CF0MF	Control Field 0 Match Flag	[Setting condition] • A match between the value received in Control Field 0 and the set value. [Clearing condition] • Writing 1 to the STCR.CF0MCL bit	R
b2	CF1MF	Control Field 1 Match Flag	[Setting condition] • A match between the data received in Control Field 1 and the set values. [Clearing condition] • Writing 1 to the STCR.CF1MCL bit	R
b3	PIBDF	Priority Interrupt Bit Detection Flag	[Setting condition] • Detection of the priority interrupt bit [Clearing condition] • Writing 1 to the STCR.PIBDCL bit	R
b4	BCDF	Bus Collision Detected Flag	[Setting condition] • Detection of the bus collision [Clearing condition] • Writing 1 to the STCR.BCDCL bit	R
b5	AEDF	Valid Edge Detection Flag	[Setting condition] • Detection of a valid edge [Clearing condition] • Writing 1 to the STCR.AEDCL bit	R
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R

26.2.28 Status Clear Register (STCR)

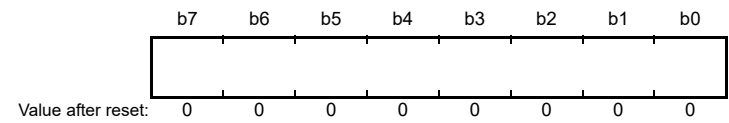
Address(es): SCI12.STCR 0008 B328h



Bit	Symbol	Bit Name	Description	R/W
b0	BFDCL	BFDF Clear	Setting this bit to 1 clears the STR.BFDF flag. This bit is read as 0.	R/W
b1	CF0MCL	CF0MF Clear	Setting this bit to 1 clears the STR.CF0MF flag. This bit is read as 0.	R/W
b2	CF1MCL	CF1MF Clear	Setting this bit to 1 clears the STR.CF1MF flag. This bit is read as 0.	R/W
b3	PIBDCL	PIBDF Clear	Setting this bit to 1 clears the STR.PIBDF flag. This bit is read as 0.	R/W
b4	BCDCL	BCDF Clear	Setting this bit to 1 clears the STR.BCDF flag. This bit is read as 0.	R/W
b5	AEDCL	AEDF Clear	Setting this bit to 1 clears the STR.AEDF flag. This bit is read as 0.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

26.2.29 Control Field 0 Data Register (CF0DR)

Address(es): SCI12.CF0DR 0008 B329h



The CF0DR register is an 8-bit readable and writable register that holds a value for comparison with Control Field 0.

26.2.30 Control Field 0 Compare Enable Register (CF0CR)

Address(es): SCI12.CF0CR 0008 B32Ah

	b7	b6	b5	b4	b3	b2	b1	b0
	CF0CE7	CF0CE6	CF0CE5	CF0CE4	CF0CE3	CF0CE2	CF0CE1	CF0CE0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CF0CE0	Control Field 0 Bit 0 Compare Enable	0: Comparison with bit 0 of Control Field 0 is disabled. 1: Comparison with bit 0 of Control Field 0 is enabled.	R/W
b1	CF0CE1	Control Field 0 Bit 1 Compare Enable	0: Comparison with bit 1 of Control Field 0 is disabled. 1: Comparison with bit 1 of Control Field 0 is enabled.	R/W
b2	CF0CE2	Control Field 0 Bit 2 Compare Enable	0: Comparison with bit 2 of Control Field 0 is disabled. 1: Comparison with bit 2 of Control Field 0 is enabled.	R/W
b3	CF0CE3	Control Field 0 Bit 3 Compare Enable	0: Comparison with bit 3 of Control Field 0 is disabled. 1: Comparison with bit 3 of Control Field 0 is enabled.	R/W
b4	CF0CE4	Control Field 0 Bit 4 Compare Enable	0: Comparison with bit 4 of Control Field 0 is disabled. 1: Comparison with bit 4 of Control Field 0 is enabled.	R/W
b5	CF0CE5	Control Field 0 Bit 5 Compare Enable	0: Comparison with bit 5 of Control Field 0 is disabled. 1: Comparison with bit 5 of Control Field 0 is enabled.	R/W
b6	CF0CE6	Control Field 0 Bit 6 Compare Enable	0: Comparison with bit 6 of Control Field 0 is disabled. 1: Comparison with bit 6 of Control Field 0 is enabled.	R/W
b7	CF0CE7	Control Field 0 Bit 7 Compare Enable	0: Comparison with bit 7 of Control Field 0 is disabled. 1: Comparison with bit 7 of Control Field 0 is enabled.	R/W

26.2.31 Control Field 0 Receive Data Register (CF0RR)

Address(es): SCI12.CF0RR 0008 B32Bh

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

CF0RR is a readable register that holds the value received in Control Field 0.

26.2.32 Primary Control Field 1 Data Register (PCF1DR)

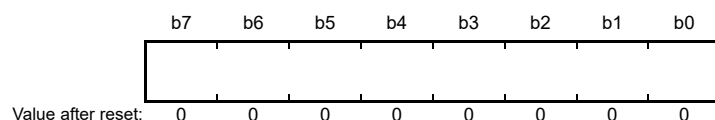
Address(es): SCI12.PCF1DR 0008 B32Ch

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

PCF1DR is an 8-bit readable and writable register that holds the 8-bit primary value for comparison with Control Field 1.

26.2.33 Secondary Control Field 1 Data Register (SCF1DR)

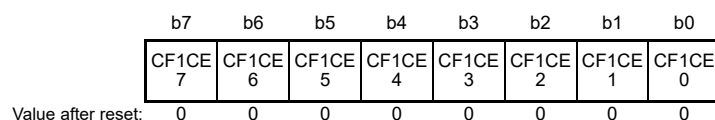
Address(es): SCI12.SCF1DR 0008 B32Dh



PCF1DR is an 8-bit readable and writable register that holds the 8-bit secondary value for comparison with Control Field 1.

26.2.34 Control Field 1 Compare Enable Register (CF1CR)

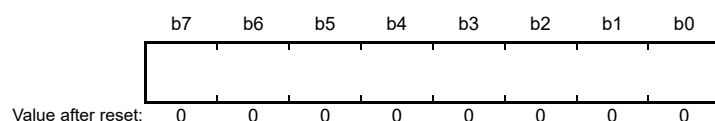
Address(es): SCI12.CF1CR 0008 B32Eh



Bit	Symbol	Bit Name	Description	R/W
b0	CF1CE0	Control Field 1 Bit 0 Compare Enable	0: Comparison with bit 0 of Control Field 1 is disabled. 1: Comparison with bit 0 of Control Field 1 is enabled.	R/W
b1	CF1CE1	Control Field 1 Bit 1 Compare Enable	0: Comparison with bit 1 of Control Field 1 is disabled. 1: Comparison with bit 1 of Control Field 1 is enabled.	R/W
b2	CF1CE2	Control Field 1 Bit 2 Compare Enable	0: Comparison with bit 2 of Control Field 1 is disabled. 1: Comparison with bit 2 of Control Field 1 is enabled.	R/W
b3	CF1CE3	Control Field 1 Bit 3 Compare Enable	0: Comparison with bit 3 of Control Field 1 is disabled. 1: Comparison with bit 3 of Control Field 1 is enabled.	R/W
b4	CF1CE4	Control Field 1 Bit 4 Compare Enable	0: Comparison with bit 4 of Control Field 1 is disabled. 1: Comparison with bit 4 of Control Field 1 is enabled.	R/W
b5	CF1CE5	Control Field 1 Bit 5 Compare Enable	0: Comparison with bit 5 of Control Field 1 is disabled. 1: Comparison with bit 5 of Control Field 1 is enabled.	R/W
b6	CF1CE6	Control Field 1 Bit 6 Compare Enable	0: Comparison with bit 6 of Control Field 1 is disabled. 1: Comparison with bit 6 of Control Field 1 is enabled.	R/W
b7	CF1CE7	Control Field 1 Bit 7 Compare Enable	0: Comparison with bit 7 of Control Field 1 is disabled. 1: Comparison with bit 7 of Control Field 1 is enabled.	R/W

26.2.35 Control Field 1 Receive Data Register (CF1RR)

Address(es): SCI12.CF1RR 0008 B32Fh



CF1RR is a readable register that holds the value received in Control Field 1.

26.2.36 Timer Control Register (TCR)

Address(es): SCI12.TCR 0008 B330h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	TCST
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	TCST	Timer Count Start	0: Stops the timer counting 1: Starts the timer counting	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

26.2.37 Timer Mode Register (TMR)

Address(es): SCI12.TMR 0008 B331h

b7	b6	b5	b4	b3	b2	b1	b0
—	TCSS[2:0]			TWRC	—	TOMS[1:0]	
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TOMS[1:0]	Timer Operating Mode Select*1	b1 b0 0 0: Timer mode 0 1: Break Field low width determination mode 1 0: Break Field low width output mode 1 1: Setting prohibited	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	TWRC	Counter Write Control	0: Data is written to the reload register and counter 1: Data is written to the reload register only	R/W
b6 to b4	TCSS[2:0]	Timer Count Clock Source Select*1	b6 b4 0 0 0: PCLK 0 0 1: PCLK/2 0 1 0: PCLK/4 0 1 1: PCLK/8 1 0 0: PCLK/16 1 0 1: PCLK/32 1 1 0: PCLK/64 1 1 1: PCLK/128	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

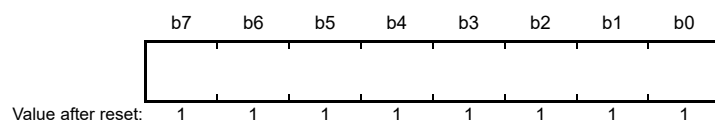
Note 1. Rewrite the TOMS[1:0] and TCSS[2:0] bits only when the timer is stopped (TCST = 0).

TWRC Bit (Counter Write Control)

This bit determines whether a value written to the TPRES or TCNT register is written to the reload register only or is written to both the reload register and the counter.

26.2.38 Timer Prescaler Register (TPRE)

Address(es): SCI12.TPRE 0008 B332h

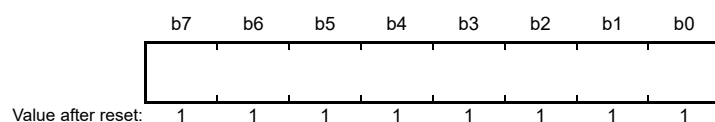


TPRE consists of an 8-bit reload register, a read buffer, and a counter, each of which has FFh as its initial value. The counter counts down in synchronization with the counter clock selected by the TMR.TCSS[2:0] bits, and is reloaded with the value in the reload register when it underflows. Underflows of this register provide the clock source to drive counting by the TCNT register. The reload register and read buffer are allocated to the same address. Data is written to the reload register in writing, and the counter value that has been transferred to the read buffer is returned in reading.

It takes one PCLK cycle to load a value from the reload register to the counter.

26.2.39 Timer Count Register (TCNT)

Address(es): SCI12.TCNT 0008 B333h



TCNT consists of an 8-bit reload register, a read buffer, and a counter, each of which has FFh as its initial value. This down-counter counts underflows of the TPRE register until the TCNT register underflows, and is then reloaded with the value from the reload register. The reload register and read buffer are allocated to the same address. Data is written to the reload register in writing, and the counter value that has been transferred to the read buffer is returned in reading.

It takes one PCLK cycle to load a value from the reload register to the counter.

26.3 Operation in Asynchronous Mode

Figure 26.5 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 communications line is usually held in the mark state (high level).

The SCI monitors the communications line. When the SCI detects a low, it regards that as a start bit and starts serial communication.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.

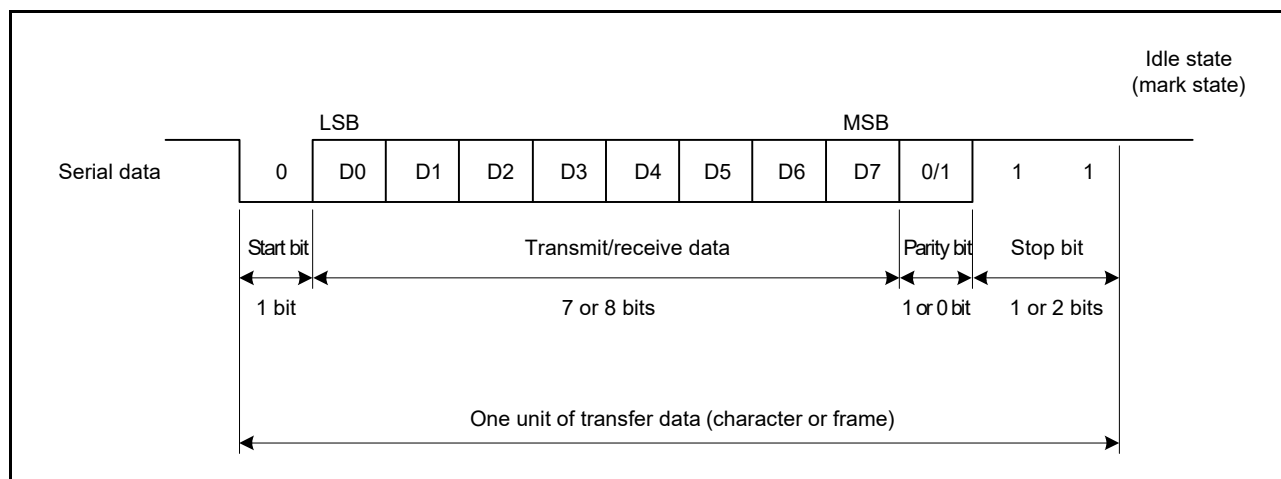


Figure 26.5 Data Format in Asynchronous Serial Communications
(Example with 8-Bit Data, Parity, 2 Stop Bits)

26.3.1 Serial Data Transfer Format

Table 26.27 lists the serial data transfer formats that can be used in asynchronous mode.

Any of 18 transfer formats can be selected according to the SMR and SCMR setting. For details of multi-processor function, refer to section 26.4, Multi-Processor Communications Function.

Table 26.27 Serial Transfer Formats (Asynchronous Mode)

SCMR Setting SMR Setting					Serial Transfer Format and Frame Length												
CHR1	CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12	13
0	0	0	0	0	S	9-bit data									STOP		
0	0	0	0	1	S	9-bit data									STOP	STOP	
0	0	1	0	0	S	9-bit data									P	STOP	
0	0	1	0	1	S	9-bit data									P	STOP	STOP
1	0	0	0	0	S	8-bit data								STOP			
1	0	0	0	1	S	8-bit data								STOP	STOP		
1	0	1	0	0	S	8-bit data								P	STOP		
1	0	1	0	1	S	8-bit data								P	STOP	STOP	
1	1	0	0	0	S	7-bit data							STOP				
1	1	0	0	1	S	7-bit data							STOP	STOP			
1	1	1	0	0	S	7-bit data							P	STOP			
1	1	1	0	1	S	7-bit data							P	STOP	STOP		
0	0	—	1	0	S	9-bit data									MPB	STOP	
0	0	—	1	1	S	9-bit data									MPB	STOP	STOP
1	0	—	1	0	S	8-bit data								MPB	STOP		
1	0	—	1	1	S	8-bit data								MPB	STOP	STOP	
1	1	—	1	0	S	7-bit data							MPB	STOP			
1	1	—	1	1	S	7-bit data							MPB	STOP	STOP		

S: Start bit
 STOP: Stop bit
 P: Parity bit
 MPB: Multi-processor bit

26.3.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a base clock with a frequency of 16 times*1 the bit rate.

In reception, the SCI samples the falling edge of the start bit using the base clock, and performs internal synchronization. Since receive data is sampled at the rising edge of the 8th pulse*1 of the base clock, data is latched at the middle of each bit, as shown in Figure 26.6. Thus the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100 (\%) \quad \cdots \text{Formula (1)}$$

M: Reception margin

N: Ratio of bit rate to clock

(N = 16 when SEMR.ABCS = 0, N = 8 when SEMR.ABCS = 1)

D: Duty cycle of clock (D = 0.5 to 1.0)

L: Frame length (L = 9 to 13)

F: Absolute value of clock frequency deviation

Assuming values of F = 0 and D = 0.5 in formula (1), the reception margin is determined by the formula below.

$$M = \{0.5 - 1/(2 \times 16)\} \times 100 (\%) = 46.875 (\%)$$

However, this is only the computed value, and a margin of 20% to 30% should be allowed in system design.

Note 1. These are values when the SEMR.ABCS bit is 0. When the ABCS bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock.

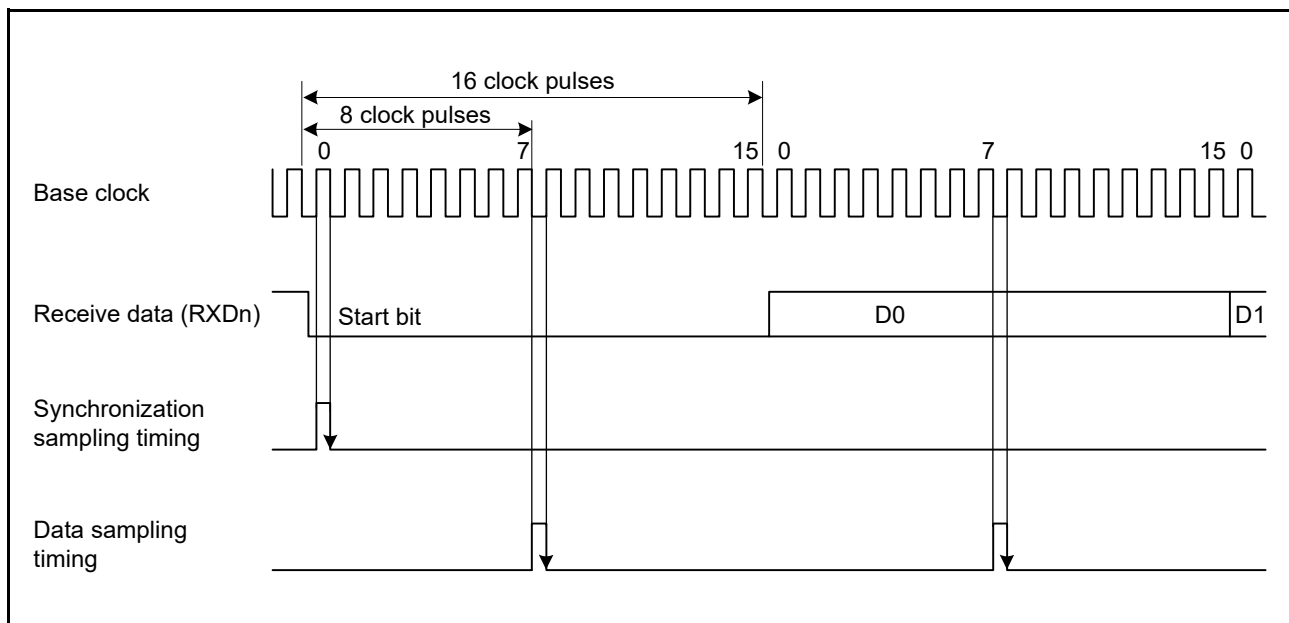


Figure 26.6 Receive Data Sampling Timing in Asynchronous Mode

26.3.3 Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input to the SCKn pin can be selected as the SCI's transfer clock, according to the setting of the SMR.CM bit and the SCR.CKE[1:0] bits.

When an external clock is input to the SCKn pin, the clock frequency should be 16 times the bit rate (when SEMR.ABCS bit = 0) and 8 times the bit rate (when SEMR.ABCS bit = 1). In addition, when an external clock is specified, the base clock of TMR0 and TMR1 can be selected by the SCIn.SEMR.ACS0 bit (n = 5, 6, 12).

When the SCI is operated on an internal clock, the clock can be output from the SCKn pin. The frequency of the clock output in this case is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in Figure 26.7.

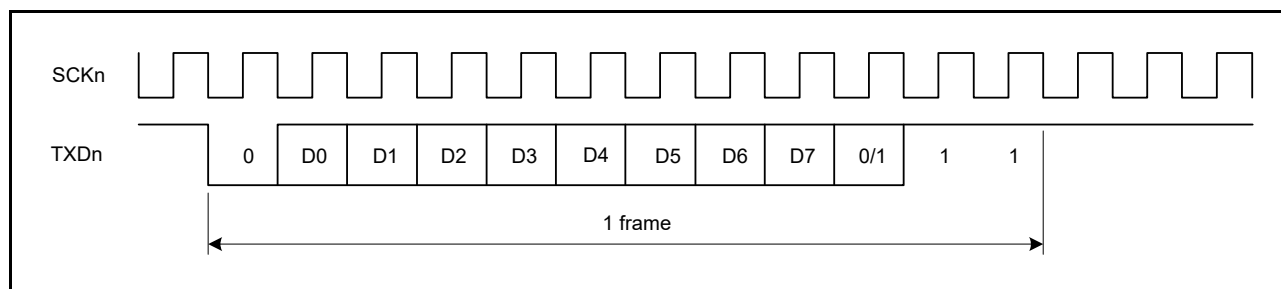


Figure 26.7 Phase Relationship between Output Clock and Transmit Data
(Asynchronous Mode: SMR.CHR = 0, PE = 1, MP = 0, STOP = 1)

26.3.4 Double-Speed Mode

The output clock frequency of the on-chip baud rate generator is doubled by setting the SEMR.BGDM bit to 1, enabling high-speed communication at a doubled bit rate. If the SEMR.ABCS bit is set to 1 under the above condition, the number of base clock cycles changes from 16 to 8, so the bit rate becomes four times faster than the initial state.

As shown by Formula (1) in section 26.3.2, Receive Data Sampling Timing and Reception Margin in Asynchronous Mode, setting the SEMR.ABCS bit to 1 changes the number of cycles to 8, and the sampling interval becomes longer. This causes the reception margin to decrease. Therefore, setting the SEMR.BGDM bit to 1 and the SEMR.ABCS bit to 0 is recommended instead of setting the SEMR.BGDM bit to 0 and the SEMR.ABCS bit to 1 for high-speed operation at a doubled bit rate.

26.3.5 CTS and RTS Functions

The CTS function is the use of input on the CTSn# pin in transmission control. Setting the SPMR.CTSE bit to 1 enables the CTS function. When the CTS function is enabled, placing the low level on the CTSn# pin causes transmission to start.

Applying the high level to the CTS# pin while transmission is in progress does not affect transmission of the current frame, which continues.

In the RTS function, by using the function of output on the RTSn# pin, a low level is output when reception becomes possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output]

When the following conditions are all satisfied.

- The SCR.RE bit is 1.
- Reception is not in progress.
- There are no received data yet to be read.
- The ORER, FER, and PER flags in the SSR register are all 0.

[Condition for high-level output]

When the conditions for low-level output are not satisfied.

Note that either one of CTS and RTS can be selected.

26.3.6 SCI Initialization (Asynchronous Mode)

Before transmitting and receiving data, start by writing the initial value 00h to the SCR register and then continue through the procedure for SCI given in Figure 26.8. Whenever the operating mode or transfer format is changed, the SCR register must be initialized before the change is made.

When the external clock is used in asynchronous mode, ensure that the clock signal is supplied even during initialization. Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, PER, and RDRF flags in the SSR register nor registers RDR, RDRH, and RDRL.

In addition, note that setting bits TIE, TE, and TEIE in the SCR register to 1 simultaneously leads to the generation of a transmit end interrupt (TEI) request before the generation of a transmit data empty interrupt (TXI) request.

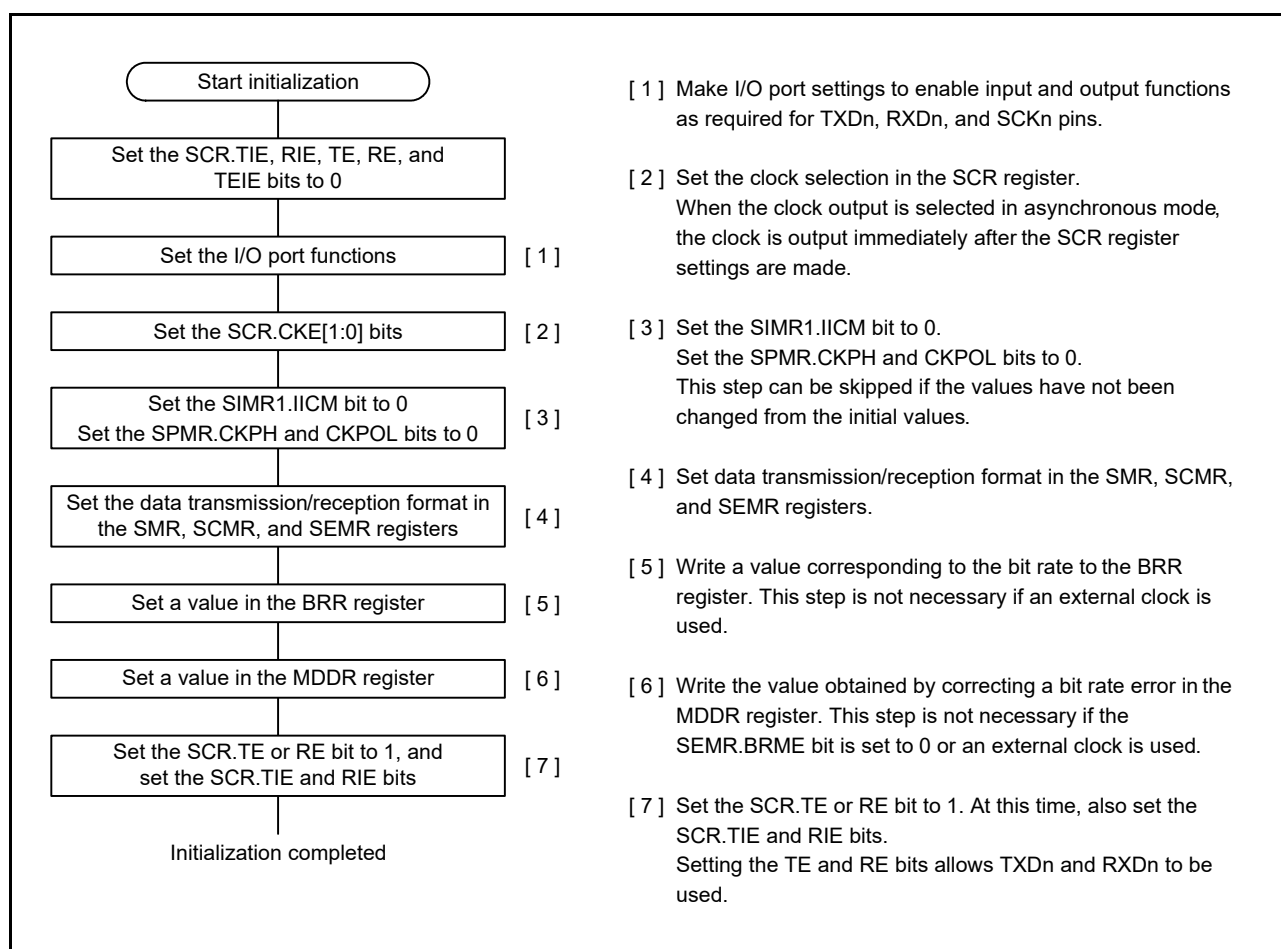


Figure 26.8 Sample SCI Initialization Flowchart (Asynchronous Mode)

Figure 26.9 shows an example of data transmission when the SCI is set to asynchronous mode according to the flow described in Figure 26.8 after a reset. When the pin function is set to the TXD pin, it is still high-impedance because the SCR.TE bit is 0. When the transmit data is written after setting the TE bit to 1, a data transmission starts. After the TE bit is set to 1, one frame of high is output from TXD pin (internal wait time) and then the data transmission starts.

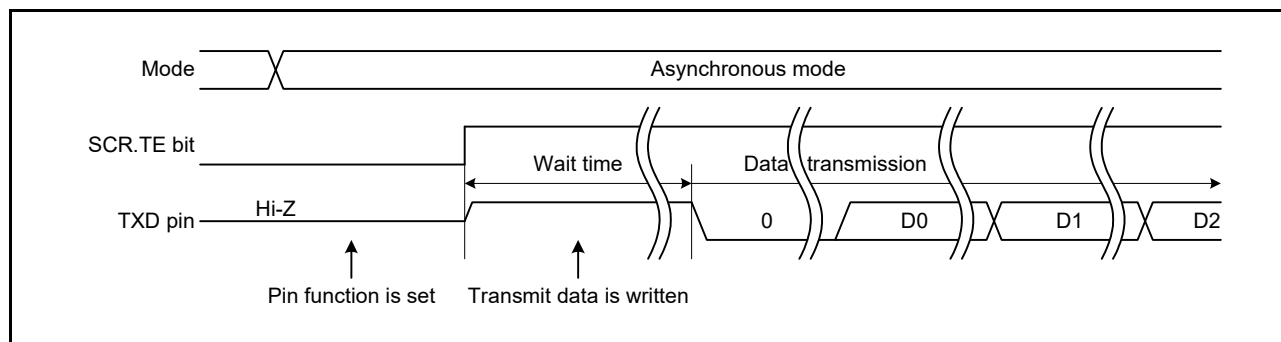


Figure 26.9 Example of Data Transmission Timing in Asynchronous Mode

26.3.7 Serial Data Transmission (Asynchronous Mode)

Figure 26.10 to Figure 26.12 show an example of the operation for serial transmission in asynchronous mode. In serial transmission, the SCI operates as described below.

1. The SCI transfers data from the TDR register*¹ to the TSR register when data is written to the TDR register*¹ in the TXI interrupt handling routine. The TXI interrupt request at the beginning of transmission is generated when the SCR.TE bit is set to 1 after the SCR.TIE bit is set to 1 or when these 2 bits are set to 1 simultaneously by a single instruction.
2. Transmission starts after the SPMR.CTSE bit is set to 0 (CTS function is disabled) and a low level on the CTSn# pin causes data transfer from the TDR register*¹ to the TSR register. If the SCR.TIE bit is 1 at this time, a TXI interrupt request is generated. Continuous transmission is obtainable by writing the next transmit data to the TDR register*¹ in the TXI interrupt handling routine before transmission of the current transmit data is completed. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (a TXI interrupt request is disabled) and the SCR.TEIE bit to 1 (a TEI interrupt request is enabled) after the last of the data to be transmitted are written to the TDR register*¹, *² from the handling routine for TXI requests.
3. Data is sent from the TXDn 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 SCI checks for updating of (writing to) the TDR register*³ at the time of stop bit output.
5. When the TDR register*³ is updated, setting of the SPMR.CTSE bit to 0 (CTS function is disabled) or a low level input on the CTSn# pin cause the next transfer of the next transmit data from the TDR register*¹ to the TSR register and sending of the stop bit, after which serial transmission of the next frame starts.
6. If the TDR register*³ is not updated, the SCI sets the SSR.TEND flag to 1, sends the stop bit, and then outputs high to put the line in the mark state. If the SCR.TEIE bit is 1 at this time, the SSR.TEND flag is set to 1 and a TEI interrupt request is generated.

Note 1. Write data not to the TDR register but to the TDRH and TDRL registers when 9-bit data length is selected.

Note 2. Write data in the order from the TDRH register to the TDRL register when 9-bit data length is selected.

Note 3. The SCI checks for updating of the TDRL register only and does not check for updating of the TDRH register when 9-bit data length is selected.

Figure 26.13 shows a sample flowchart for serial transmission in asynchronous mode.

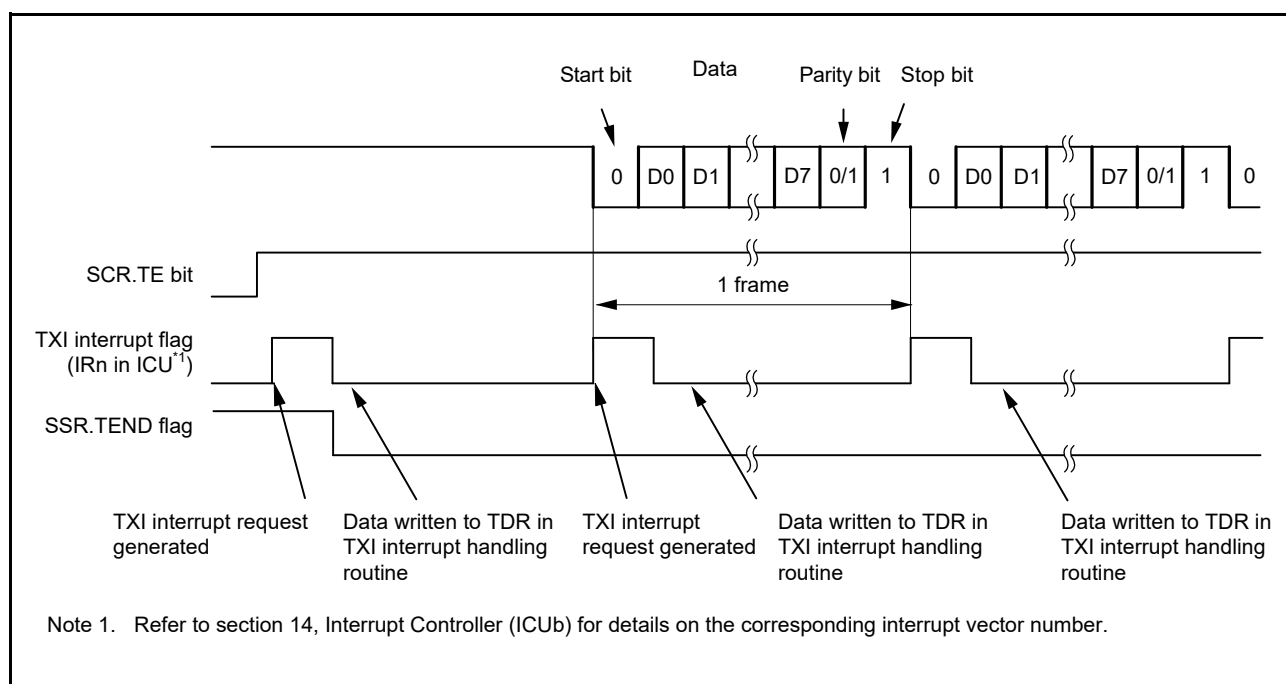


Figure 26.10 Example of Operation for Serial Transmission in Asynchronous Mode (1)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, at the Beginning of Transmission)

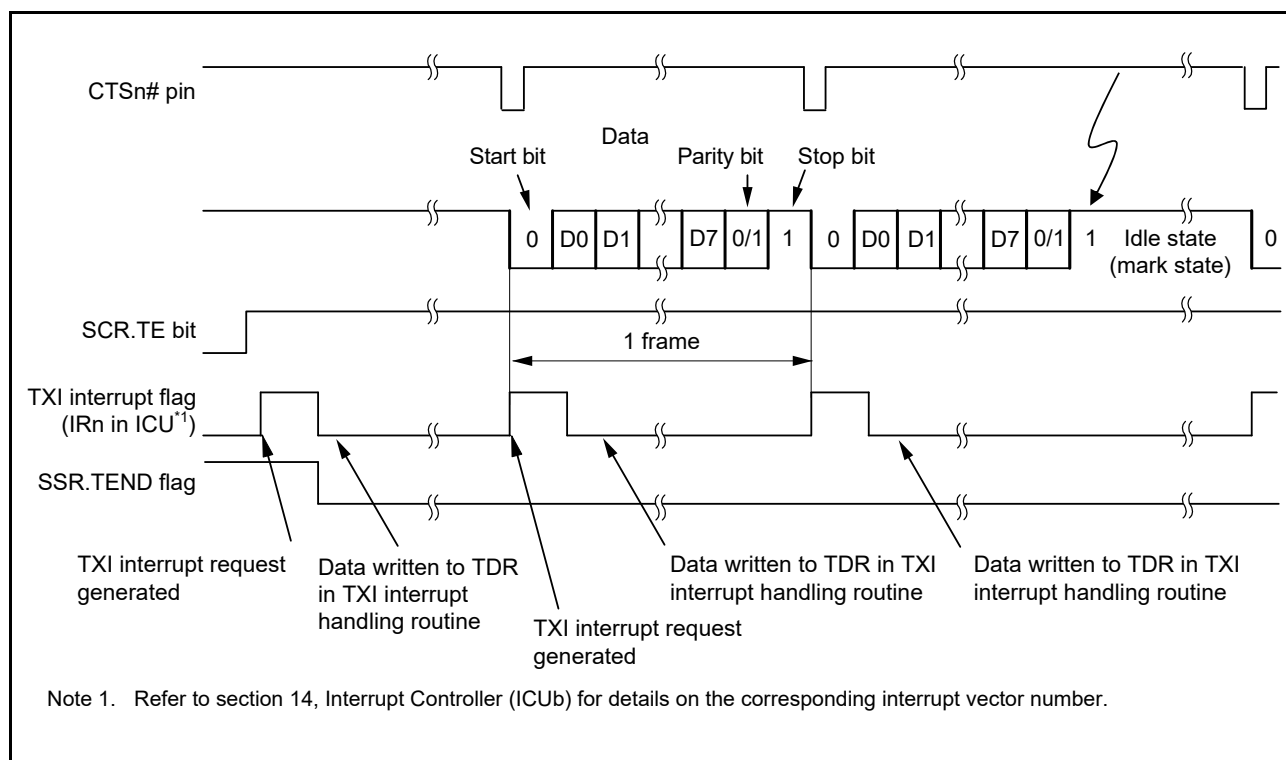


Figure 26.11 Example of Operation for Serial Transmission in Asynchronous Mode (2)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Used, at the Beginning of Transmission)

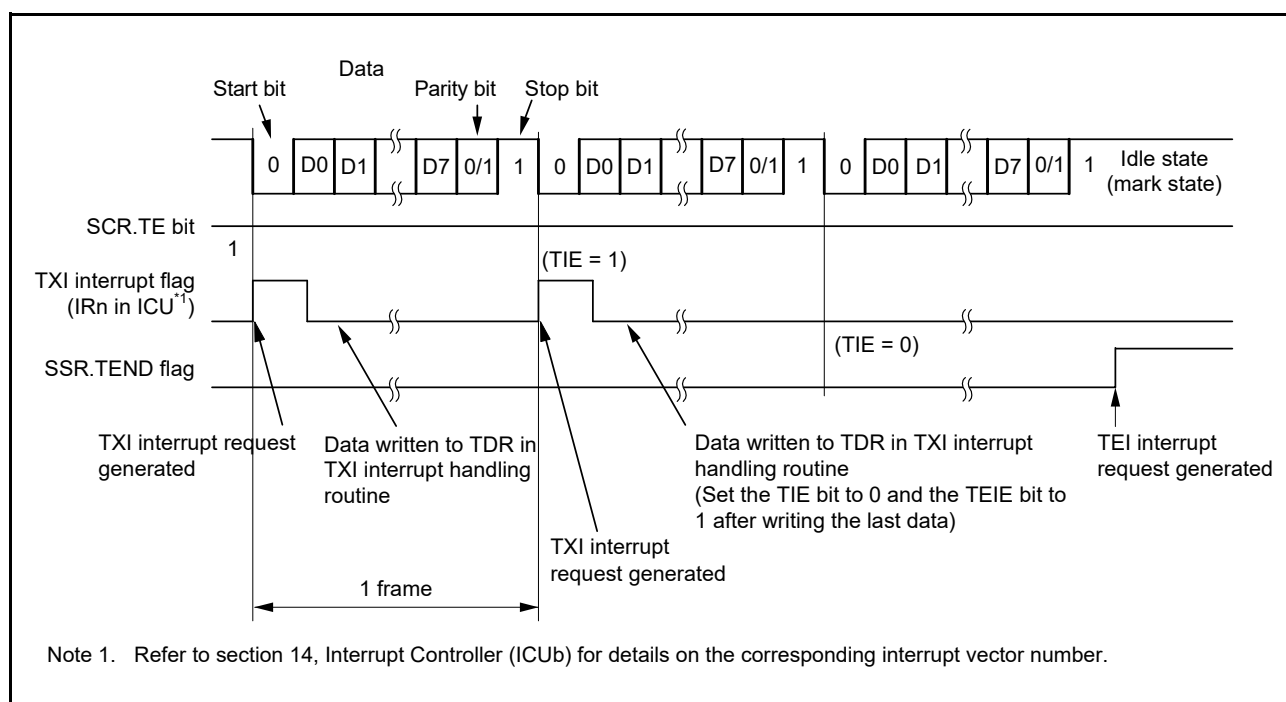


Figure 26.12 Example of Operation for Serial Transmission in Asynchronous Mode (3)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, from the Middle of Transmission until Transmission Completion)

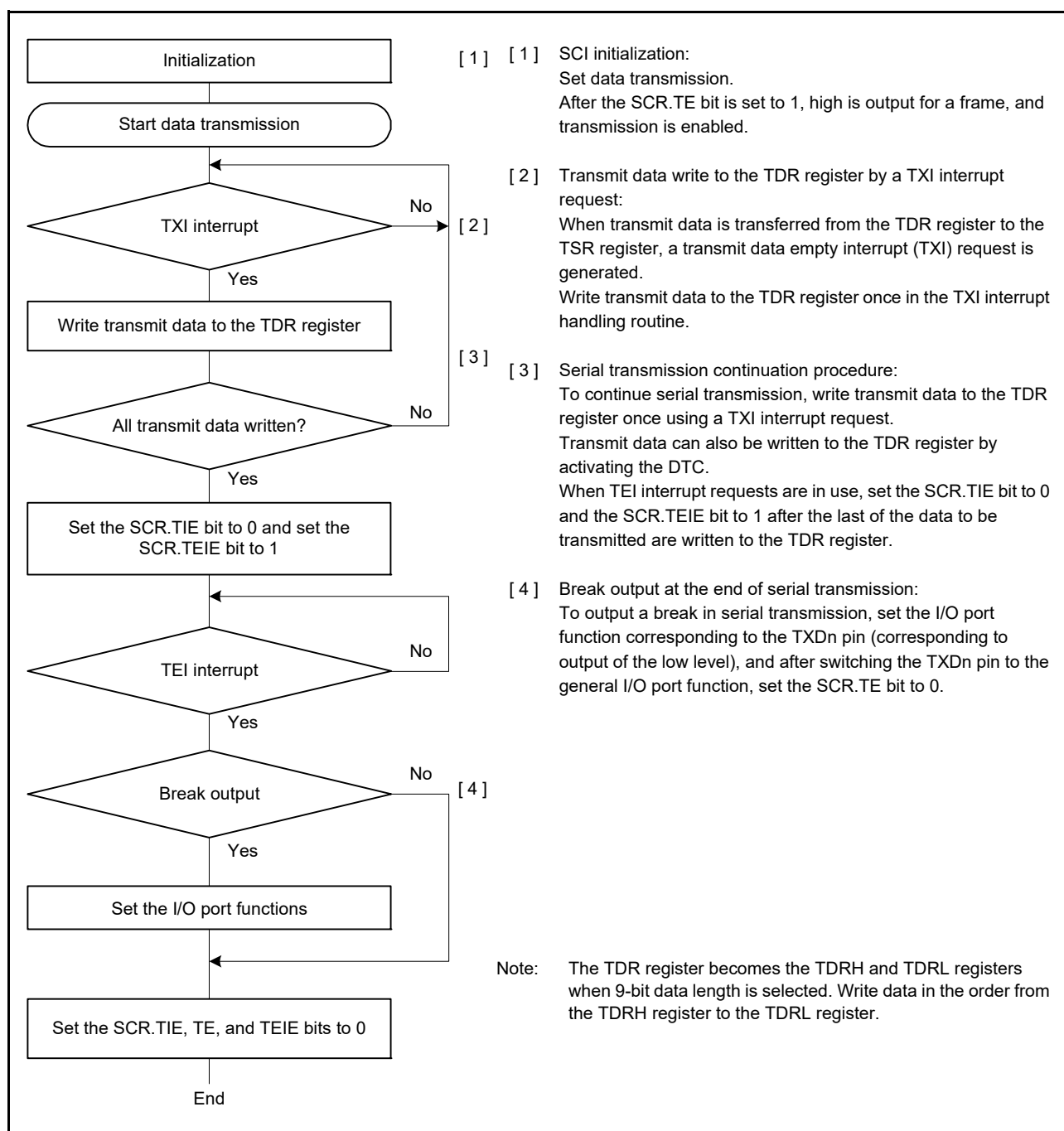


Figure 26.13 Example of Serial Transmission Flowchart in Asynchronous Mode

26.3.8 Serial Data Reception (Asynchronous Mode)

Figure 26.14 and Figure 26.15 show an example of the operation for serial data reception in asynchronous mode. In serial data reception, the SCI operates as described below.

1. When the value of the SCR.RE bit becomes 1, the output signal on the RTSn# pin goes to the low level.
2. When the SCI monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in the RSR register, and checks the parity bit and stop bit.
3. If an overrun error occurs, the SSR.ORER flag is set to 1. If the SCR.RIE bit is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to the RDR register*¹.
4. If a parity error is detected, the SSR.PER flag is set to 1 and receive data is transferred to the RDR register*¹. If the SCR.RIE bit is 1 at this time, an ERI interrupt request is generated.
5. If a framing error (when the stop bit is 0) is detected, the SSR.FER flag is set to 1 and receive data is transferred to the RDR register*¹. If the SCR.RIE bit is 1 at this time, an ERI interrupt request is generated.
6. When reception finishes successfully, receive data is transferred to the RDR register*¹. If the SCR.RIE bit is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to the RDR register*¹ in this RXI interrupt handling routine before reception of the next receive data is completed. Reading the received data that have been transferred to the RDR register*¹ causes the RTSn# pin to output the low level.

Note 1. Read data not in the RDR register but in the RDRH and RDRL registers when 9-bit data length is selected.

Note 2. The SCI checks for reading of the RDRL register only and does not check for reading of the RDRH register when 9-bit data length is selected.

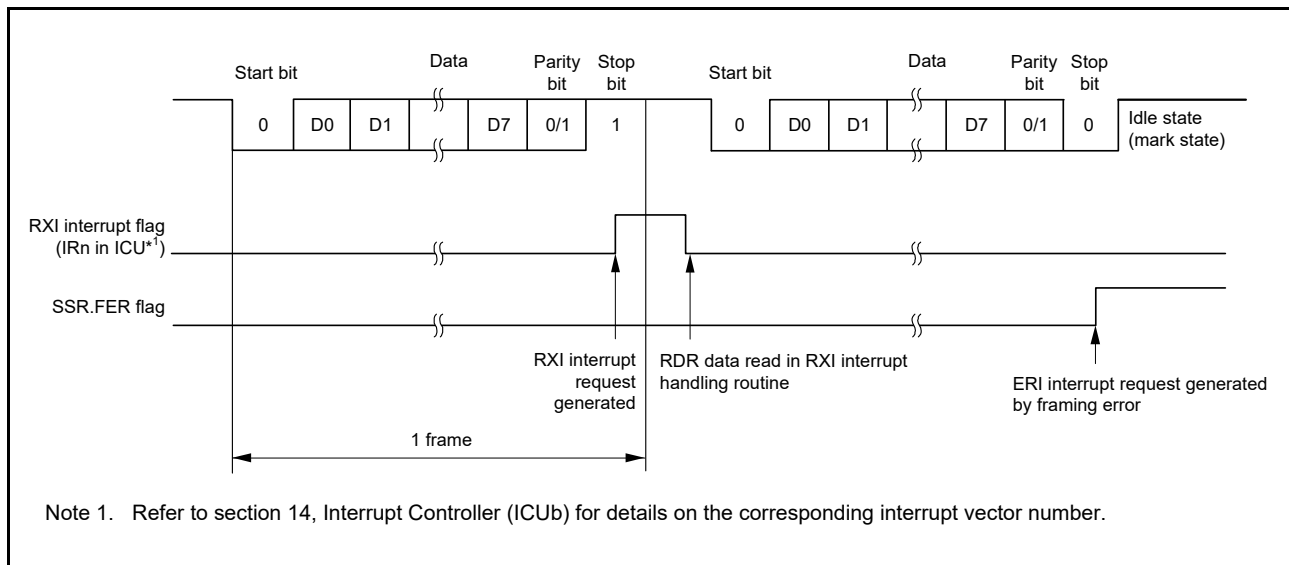


Figure 26.14 Example of SCI Operation for Serial Reception in Asynchronous Mode (1) (When RTS Function is Not Used) (Example with 8-Bit Data, Parity, 1 Stop Bit)

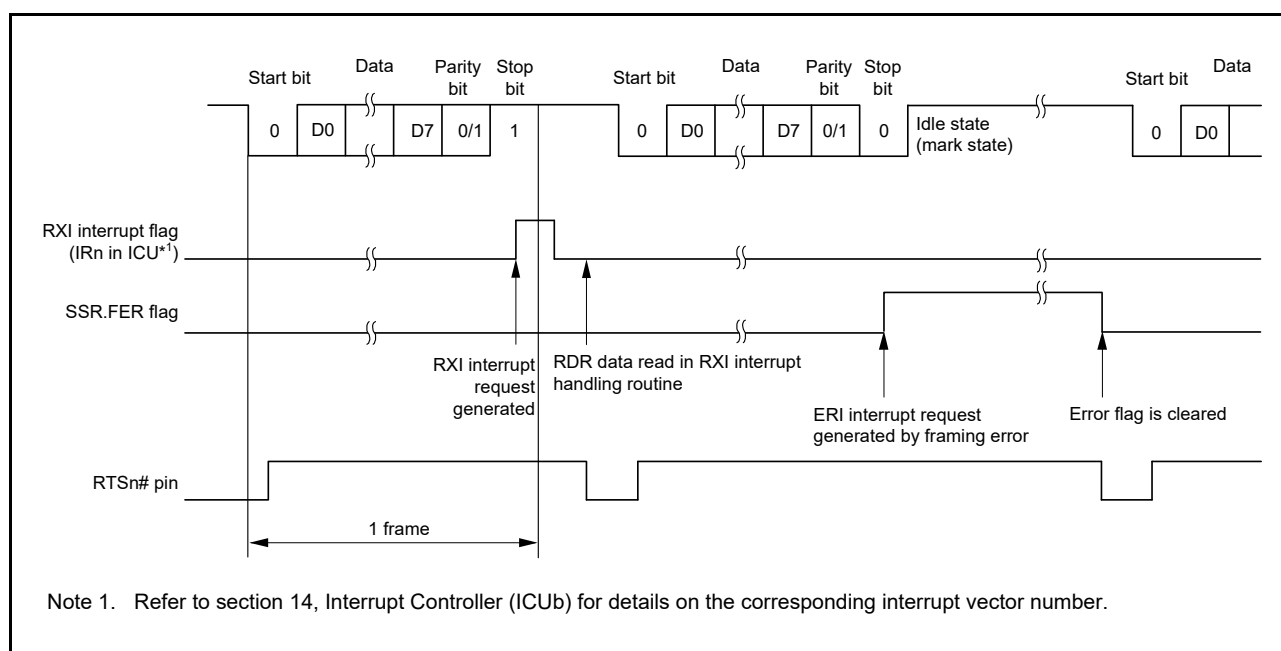


Figure 26.15 Example of SCI Operation for Serial Reception in Asynchronous Mode (2)
(When RTS Function is Used) (Example with 8-Bit Data, Parity, 1 Stop Bit)

Table 26.28 lists the states of the status flags in the SSR register and receive data handling when a receive error is detected.

If a receive error is detected, an ERI interrupt request is generated but an RXI interrupt request is not generated. Data reception cannot be resumed while the receive error flag is 1. Accordingly, set the ORER, FER, and PER flags to 0 before resuming reception. Moreover, be sure to read the RDR (or the RDRL) register during overrun error processing. When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read the RDR (or the RDRL) register because received data which has not yet been read may be left in the RDR (or the RDRL) register.

Figure 26.16 and Figure 26.17 show samples of flowcharts for serial data reception.

Table 26.28 Status Flags in the SSR Register and Receive Data Handling

Status Flags in the SSR Register				
ORER	FER	PER	Receive Data	Receive Error Type
1	0	0	Lost	Overflow error
0	1	0	Transferred to the RDR register*1	Framing error
0	0	1	Transferred to the RDR register*1	Parity error
1	1	0	Lost	Overflow error + framing error
1	0	1	Lost	Overflow error + parity error
0	1	1	Transferred to the RDR register*1	Framing error + parity error
1	1	1	Lost	Overflow error + framing error + parity error

Note 1. Read data not in the RDR register but in the RDRH and RDRL registers when 9-bit data length is selected.

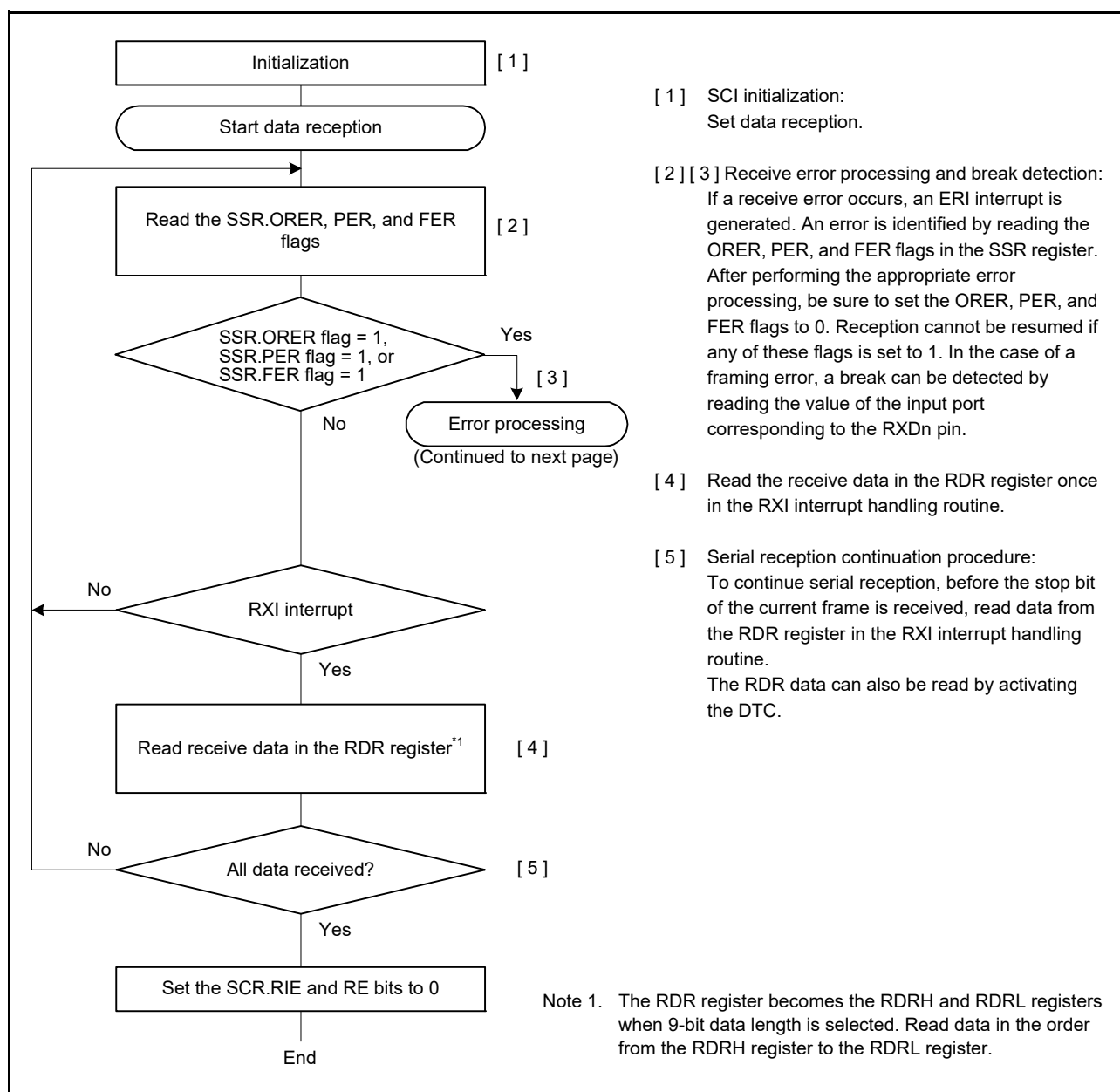


Figure 26.16 Example Flowchart of Serial Reception in Asynchronous Mode (1)

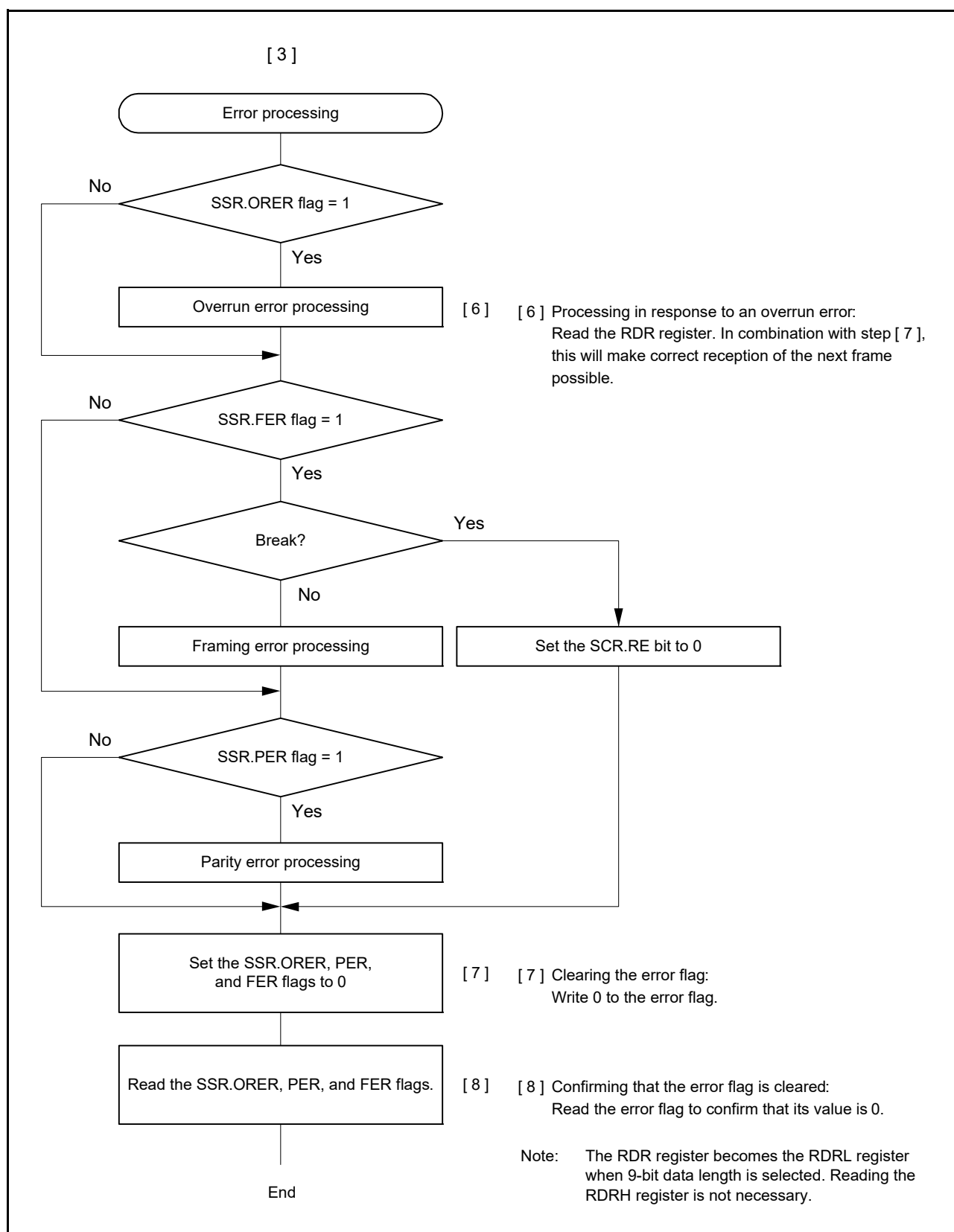


Figure 26.17 Example Flowchart of Serial Reception in Asynchronous Mode (2)

26.4 Multi-Processor Communications Function

Using the multi-processor communication functions enables to transmit and receive data 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 between the ID transmission cycle and the data transmission cycle. When the multi-processor bit is set to 1, it indicates the ID transmission cycle and when the multi-processor bit is set to 0, it indicates the data transmission cycle. Figure 26.18 shows an example of communication between processors by using a multi-processor format. First, a transmitting station transmits communication data in which the multi-processor bit set to 1 is added to the ID code of the receiving station. Next, the transmitting station transmits the communication data in which the multi-processor bit set to 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 and if the two match, receives the communication data that is subsequently transmitted. If the received ID does not match with the ID of the receiving station, the receiving station skips the communication data until again receiving the communication data in which the multi-processor bit is set to 1. For supporting this function, the SCI provides the SCR.MPIE bit. When the MPIE bit is set to 1, transfer of receive data from the RSR register to the RDR register (the RDRH and RDRL registers when 9-bit data length is selected), detection of a receive error, and setting the respective status flags RDRF, ORER, and FER in the SSR register are disabled until reception of data in which the multi-processor bit is set to 1. Upon receiving a reception character in which the multi-processor bit is set to 1, the SSR.MPB bit is set to 1 and the SCR.MPIE bit is automatically cleared, thus returning to a normal reception operation. During this time, an RXI interrupt is generated if the SCR.RIE bit is 1. 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 which is used for the multi-processor communication is also the same as the clock used in the normal asynchronous mode.

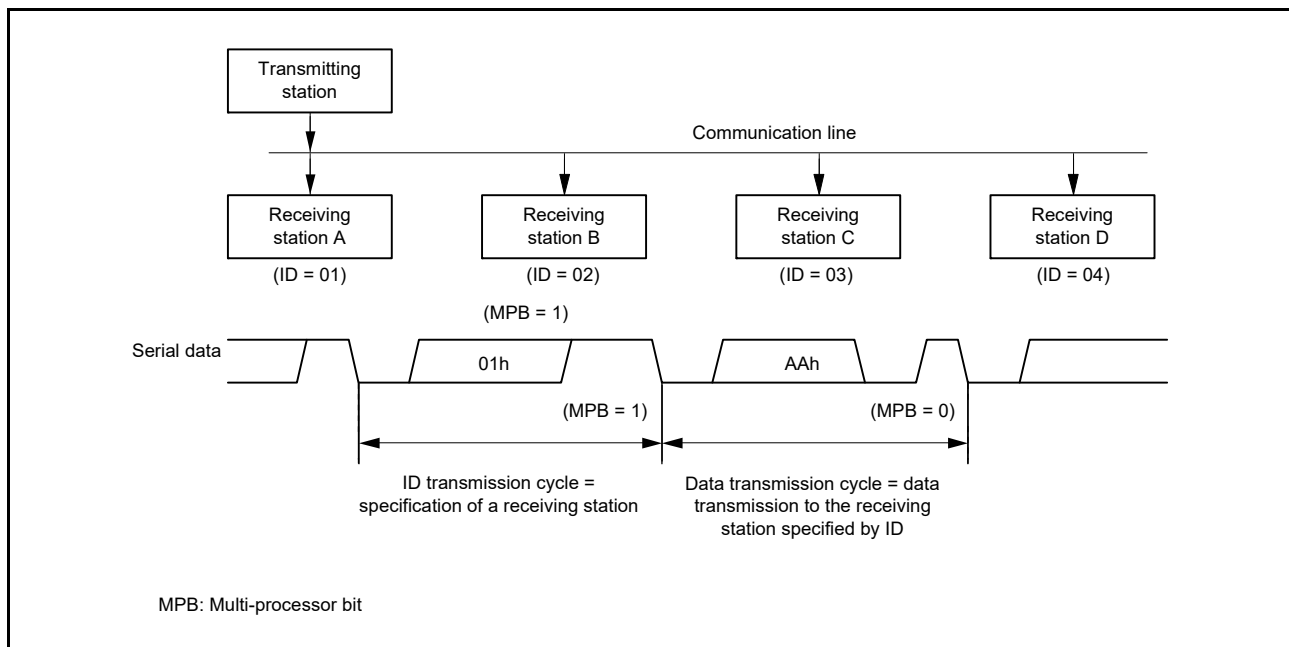


Figure 26.18 An Example of Communication using the Multi-Processor Format
(Example of Transmission of Data AAh to Receiving Station A)

26.4.1 Multi-Processor Serial Data Transmission

Figure 26.19 is a sample flowchart of multi-processor data transmission. In the ID transmission cycle, the ID should be transmitted with the SSR.MPBT bit set to 1. In the data transmission cycle, the data should be transmitted with the MPBT bit set to 0. The other operations are the same as the operations in asynchronous mode.

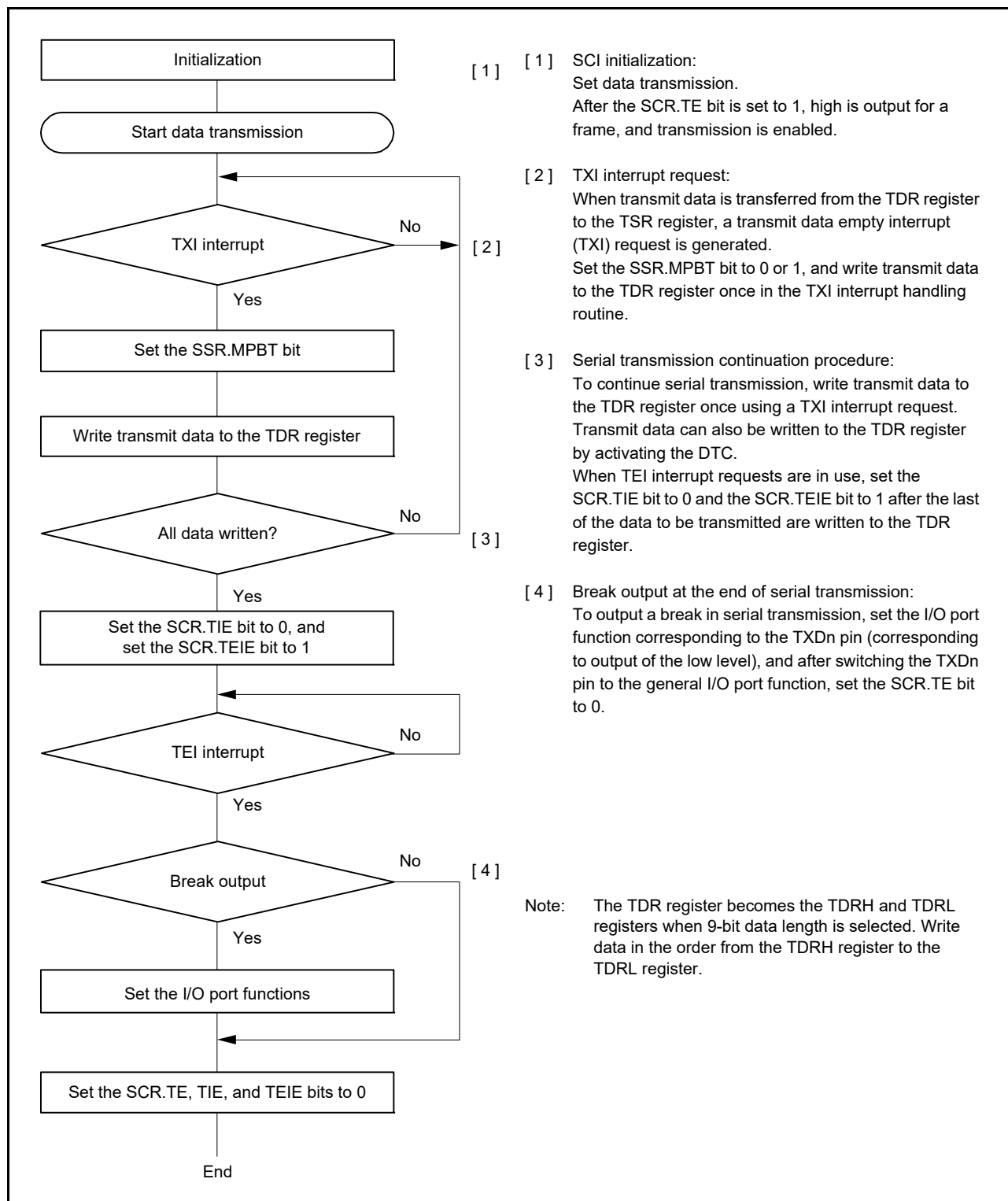


Figure 26.19 Example of Multi-Processor Serial Transmission Flowchart

26.4.2 Multi-Processor Serial Data Reception

Figure 26.21 and Figure 26.22 are sample flowcharts of multi-processor data reception. When the SCR.MPIE bit is set to 1, reading the communication data is skipped until reception of the communication data in which the multi-processor bit is set to 1. When the communication data in which the multi-processor bit is set to 1 is received, the received data is transferred to the RDR register (the RDRH and RDRL registers when 9-bit data length is selected). During this time, the RXI interrupt request is generated. The other operations are the same as the operations in asynchronous mode. Figure 26.20 is the example of operation for reception.

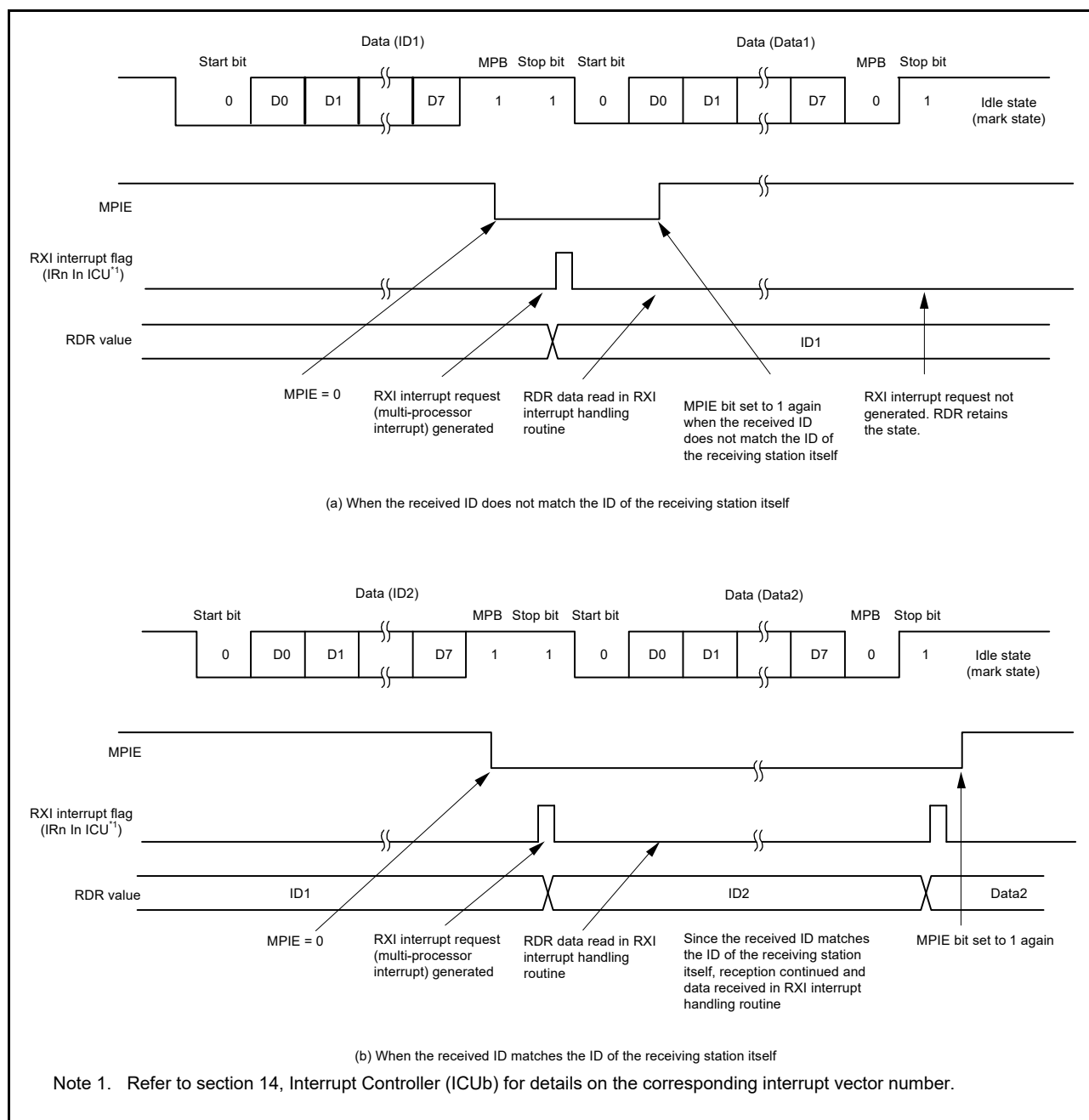


Figure 26.20 Example of SCI Reception (8-Bit Data/Multi-Processor Bit/1 Stop Bit)

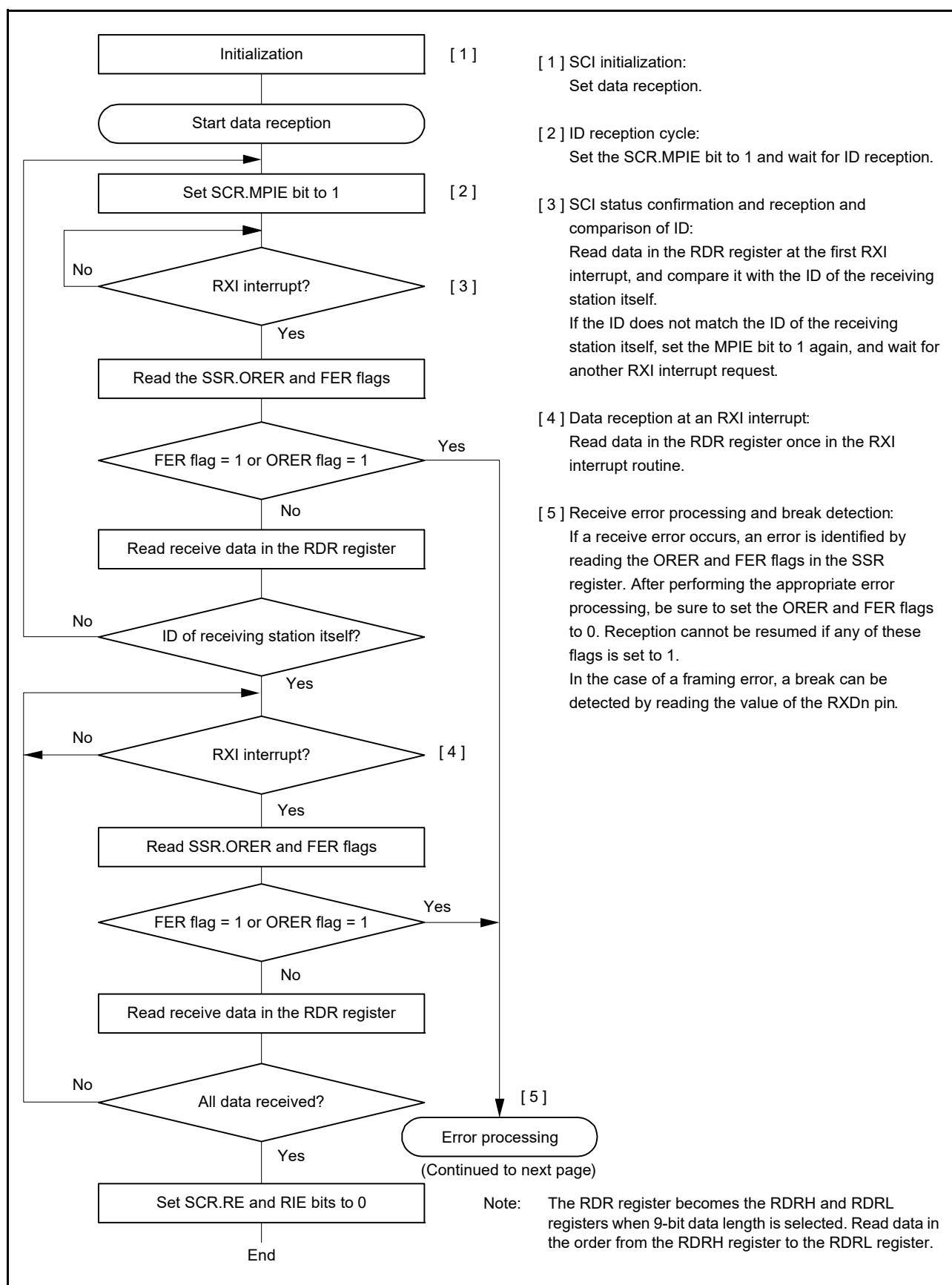


Figure 26.21 Example of Multi-Processor Serial Reception Flowchart (1)

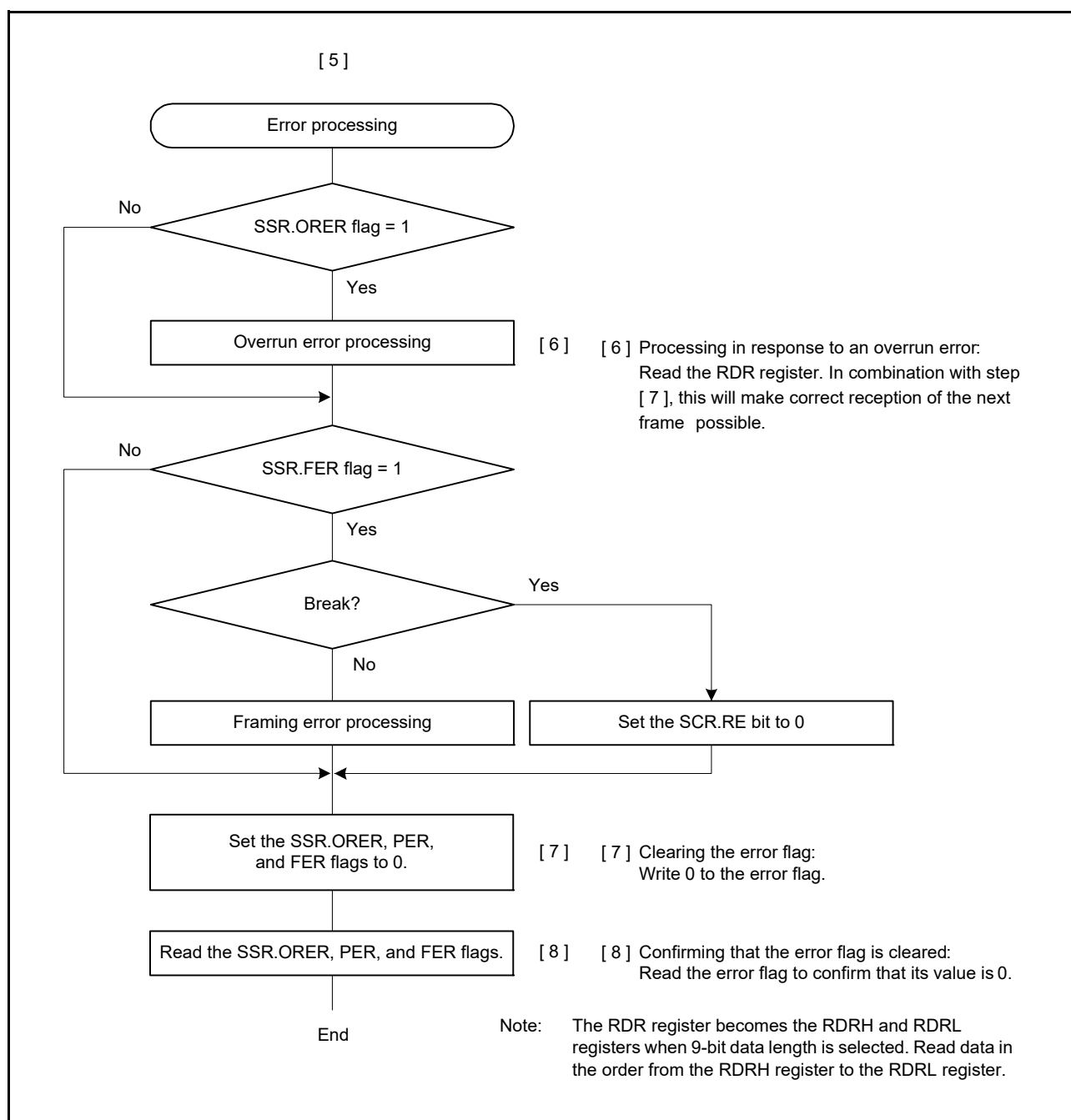


Figure 26.22 Example of Multi-Processor Serial Reception Flowchart (2)

26.5 Operation in Clock Synchronous Mode

Figure 26.23 shows the data format for clock synchronous serial data communications.

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 clock synchronous mode, no parity bit can be added.

In data transmission, the SCI outputs data from one falling edge of the synchronization clock to the next. In data reception, the SCI 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.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications by use of a common clock. Both the transmitter and the receiver also 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.

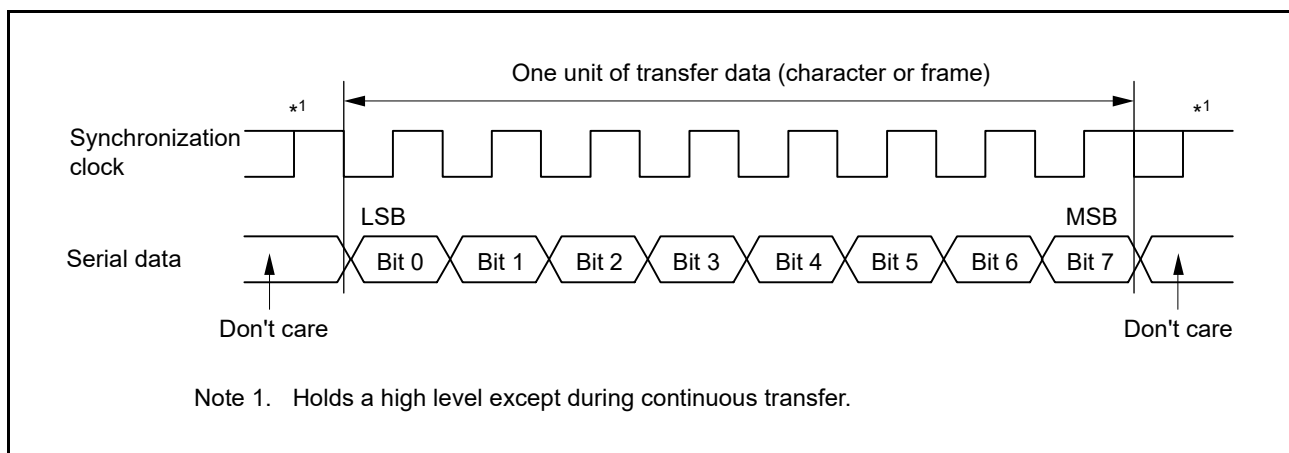


Figure 26.23 Data Format in Clock Synchronous Serial Communications (LSB First)

26.5.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCKn pin can be selected, according to the setting of the SCR.CKE[1:0] bits.

When the SCI is operated on an internal clock, the synchronization clock is output from the SCKn pin. Eight synchronization clock pulses are output in the transfer of one character, and when no transfer is performed the clock is held high. However, when only data reception is performed while the CTS function is disabled, the synchronization clock output is started at the same time when the SCR.RE bit set to 1. The synchronization clock is stopped at the high level when an overrun error occurs or the SCR.RE bit is set to 0.

When only data reception is performed and the CTS function is enabled, the clock output is not started even when the SCR.RE bit set to 1 if the CTSn# pin input is high when the SCR.RE bit is 0. The synchronization clock output is started when the SCR.RE bit is set to 1 and the CTSn# pin input is low. After that, if the CTSn# pin input is high on completion of the frame reception, the synchronization clock output is stopped at the high level. If the CTSn# pin input continues to be low, the synchronization clock is stopped at the high level when an overrun error occurs or the SCR.RE bit is set to 0.

26.5.2 CTS and RTS Functions

In the CTS function, CTSn# pin input is used to control reception/transmission start when the clock source is the internal clock. Setting the SPMR.CTSE bit to 1 enables the CTS function.

When the CTS function is enabled, placing the low level on the CTSn# pin causes reception/transmission to start.

Applying the high level to the CTS# pin while reception/transmission are in progress does not affect reception/transmission of the current frame, which continues.

In the RTS function, RTSn# pin output is used to request reception/transmission start when the clock source is an external synchronizing clock. A low level is output when serial communications become possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output]

When the following conditions are all satisfied.

- The SCR.RE or SCR.TE bit is 1.
- Transmission or reception is not in progress.
- There are no received data yet to be read (when the SCR.RE bit is 1).
- Untransmitted data is present (when the SCR.TE bit is 1).
- The SSR.ORER flag is 0.

[Condition for high-level output]

The conditions for low-level output have not been satisfied.

26.5.3 SCI Initialization (Clock Synchronous Mode)

Before transmitting and receiving data, start by writing the initial value 00h to the SCR register and then continue through the procedure for SCI given in Figure 26.24. Whenever the operating mode or transfer format is changed, the SCR register must be initialized before the change is made.

Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in the SSR register nor the RDR register.

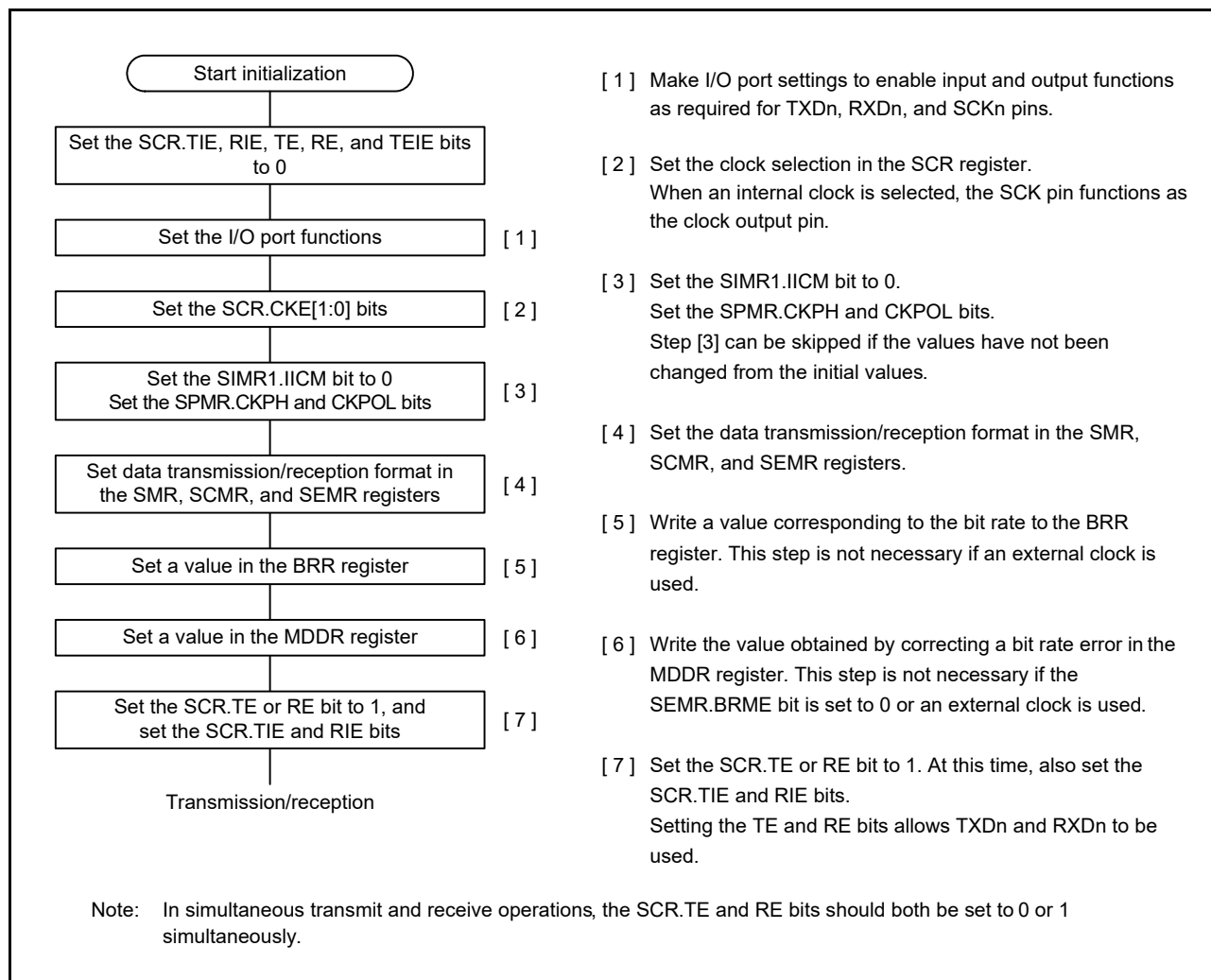


Figure 26.24 Example of SCI Initialization Flowchart (Clock Synchronous Mode)

26.5.4 Serial Data Transmission (Clock Synchronous Mode)

Figure 26.25, Figure 26.26, and Figure 26.27 show an example of the operation for serial transmission in clock synchronous mode.

In serial data transmission, the SCI operates as described below.

1. The SCI transfers data from the TDR register to the TSR register when data is written to the TDR register in the TXI interrupt handling routine. The TXI interrupt request at the beginning of transmission is generated when the SCR.TE bit is set to 1 after the SCR.TIE bit is set to 1 or when these 2 bits are set to 1 simultaneously by a single instruction.
2. After transferring data from the TDR register to the TSR register, the SCI starts transmission. When the SCR.TIE bit is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to the TDR register in this TXI interrupt handling routine before transmission of the current transmit data has finished. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (a TXI interrupt request is disabled) and the SCR.TEIE bit to 1 (a TEI interrupt request is enabled) after the last of the data to be transmitted are written to the TDR register from the handling routine for TXI requests.
3. 8-bit data is sent from the TXDn pin in synchronization with the output clock when clock output mode has been specified and in synchronization with the input clock when use of an external clock has been specified. Output of the clock signal is suspended until the input CTS signal is at the low level while the SPMR.CTSE bit is 1 (CTS function is enabled).
4. The SCI checks for updating of (writing to) the TDR register at the time of the last bit output.
5. When TDR is updated, the next transmit data is transferred from the TDR register to the TSR register, and serial transmission of the next frame is started.
6. If the TDR register is not updated, set the SSR.TEND flag to 1 and the TXDn pin retains the output state of the last bit. If the SCR.TEIE bit is 1 at this time, a TEI interrupt request is generated. The SCKn pin is held high.

Figure 26.28 shows a sample flowchart of serial data transmission.

Transmission will not start while a receive error flag (ORER, FER, or PER in the SSR register) is set to 1. Be sure to set the receive error flags to 0 before starting transmission. Note that setting the SCR.RE bit to 0 does not clear the receive error flags.

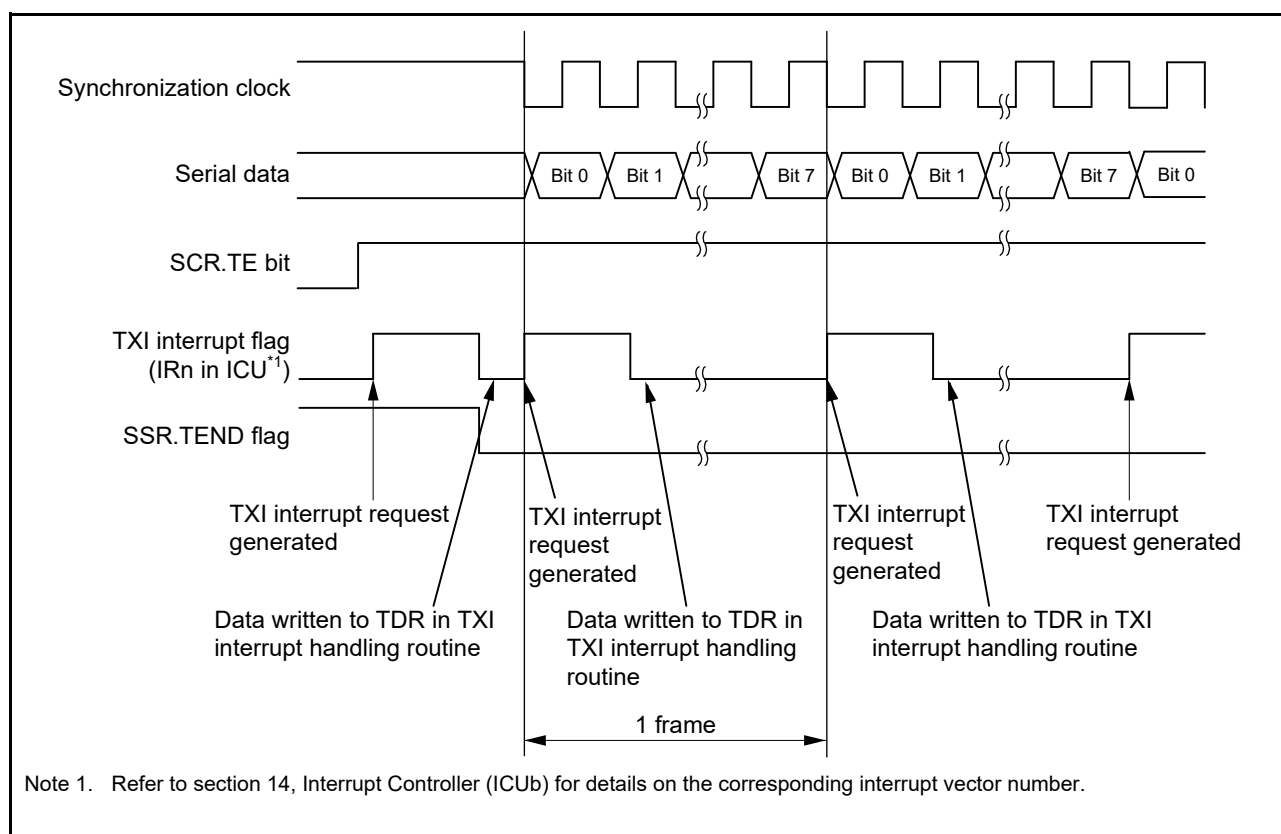


Figure 26.25 Example of Serial Data Transmission in Clock Synchronous Mode When the CTS Function is Not Used at the Beginning of Transmission

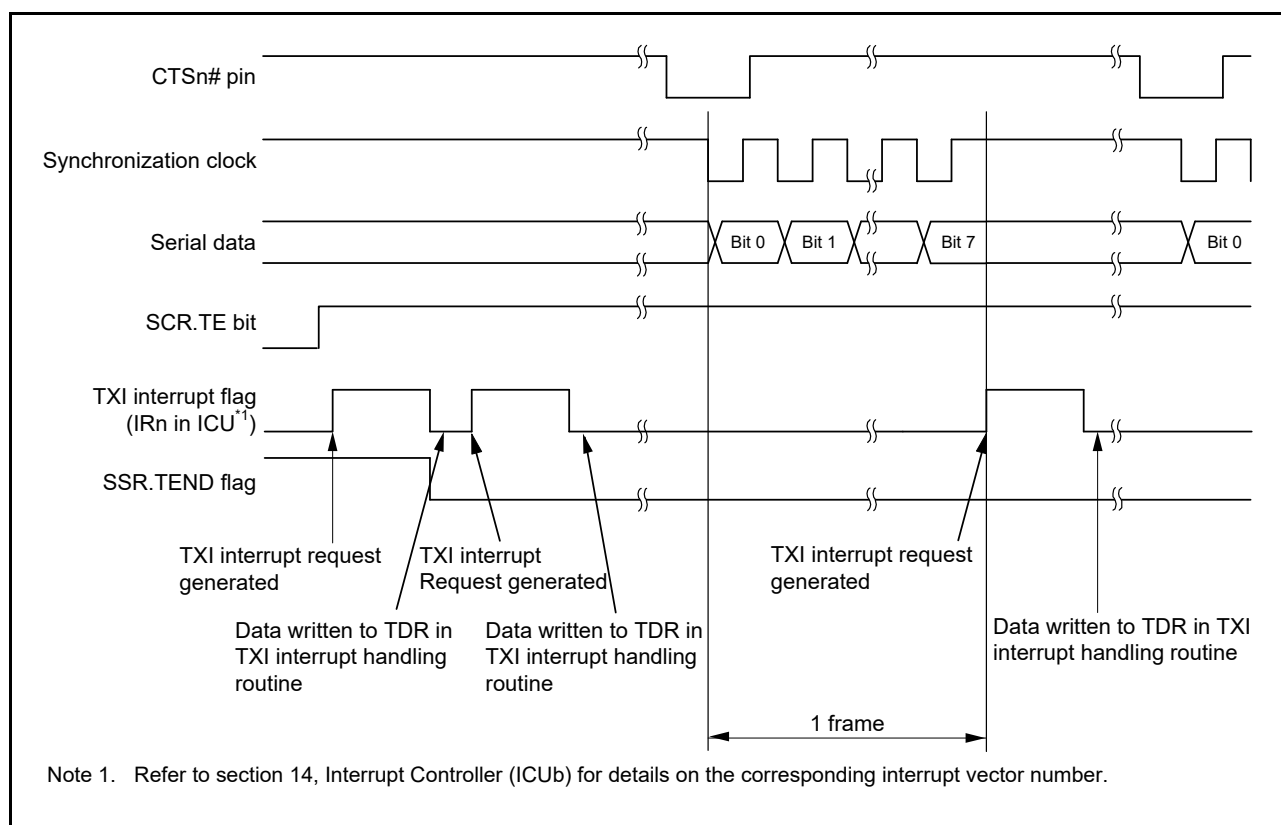


Figure 26.26 Example of Serial Data Transmission in Clock Synchronous Mode When the CTS Function is Used at the Beginning of Transmission

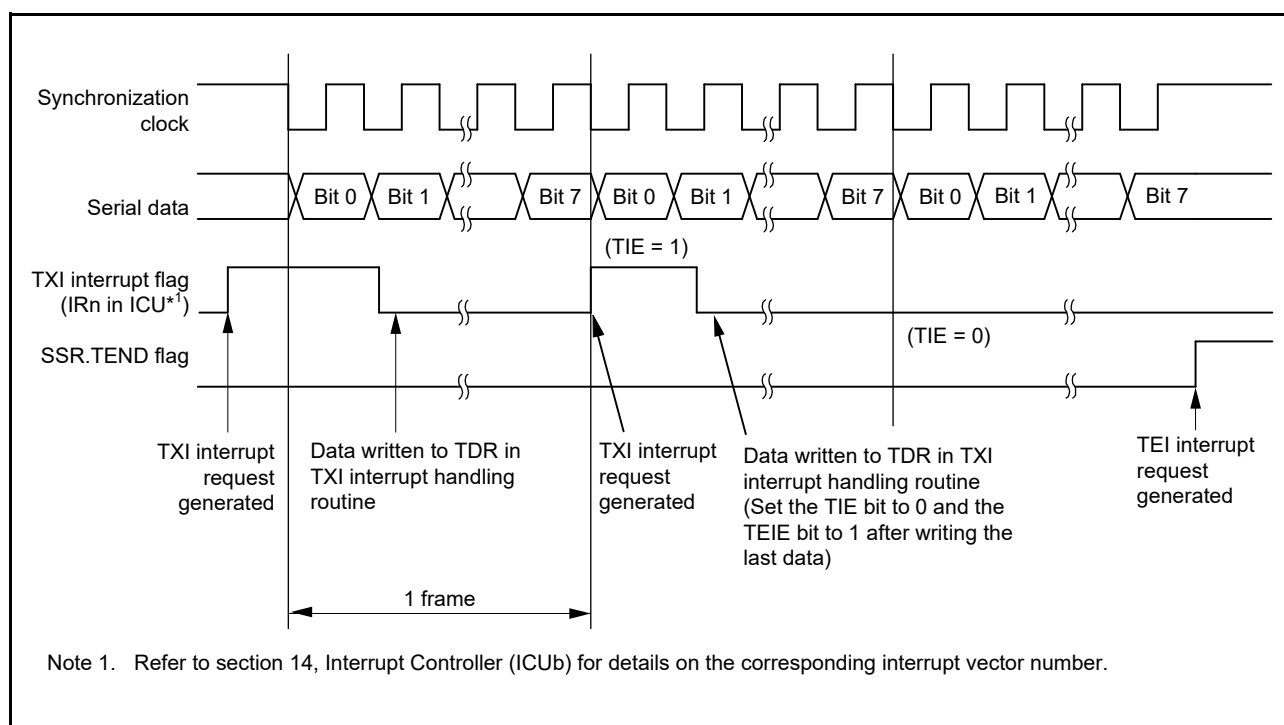


Figure 26.27 Example of Serial Data Transmission in Clock Synchronous Mode from the Middle of Transmission until Transmission Completion

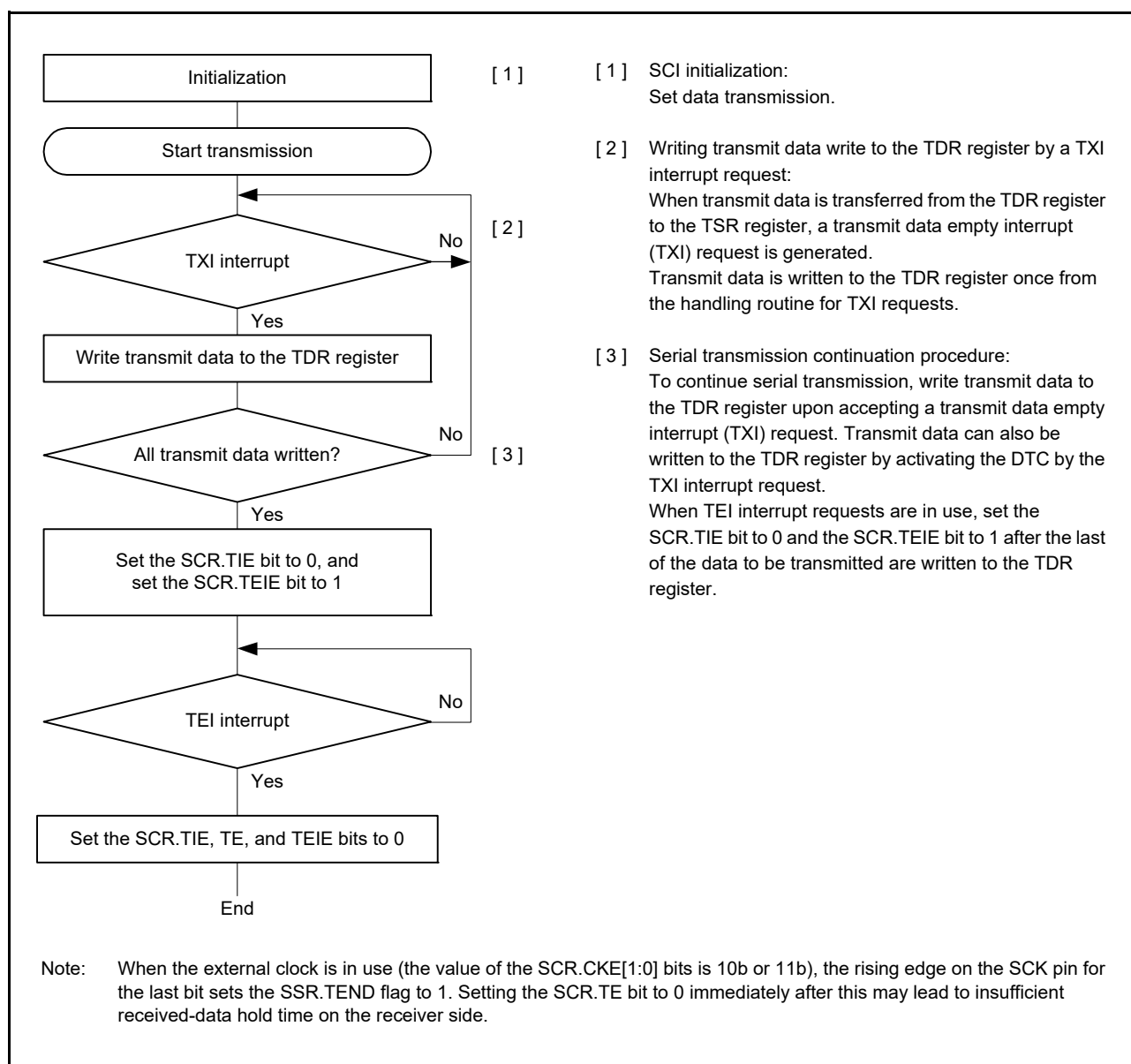


Figure 26.28 Example Flowchart of Serial Transmission in Clock Synchronous Mode

26.5.5 Serial Data Reception (Clock Synchronous Mode)

Figure 26.29 and Figure 26.30 show an example of SCI operation for serial reception in clock synchronous mode. In serial data reception, the SCI operates as described below.

1. The value of the SCR.RE bit becoming 1 places the signal output on the RTSn# pin at the low level (when the RTS function is in use).
2. The SCI performs internal initialization and starts receiving data in synchronization with a synchronization clock input or output, and stores the receive data in the RSR register.
3. If an overrun error occurs, the SSR.ORER flag is set to 1. If the SCR.RIE bit is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to the RDR register.
4. When reception finishes successfully, receive data is transferred to the RDR register. If the SCR.RIE bit is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to the RDR register in this RXI interrupt handling routine before reception of the next receive data is completed. Reading out the received data that have been transferred to the RDR register causes the RTSn# pin to output the low level (when the RTS function is in use).

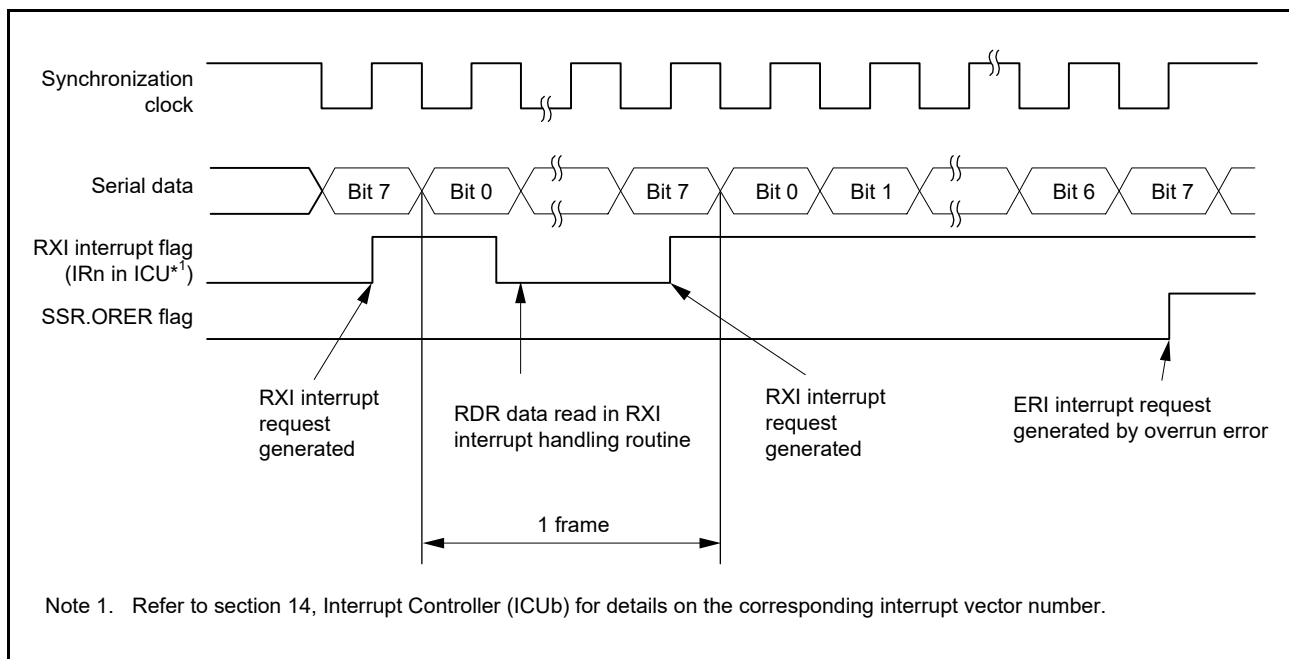


**Figure 26.29 Example of Operation for Serial Reception in Clock Synchronous Mode (1)
(When RTS Function is Not Used)**

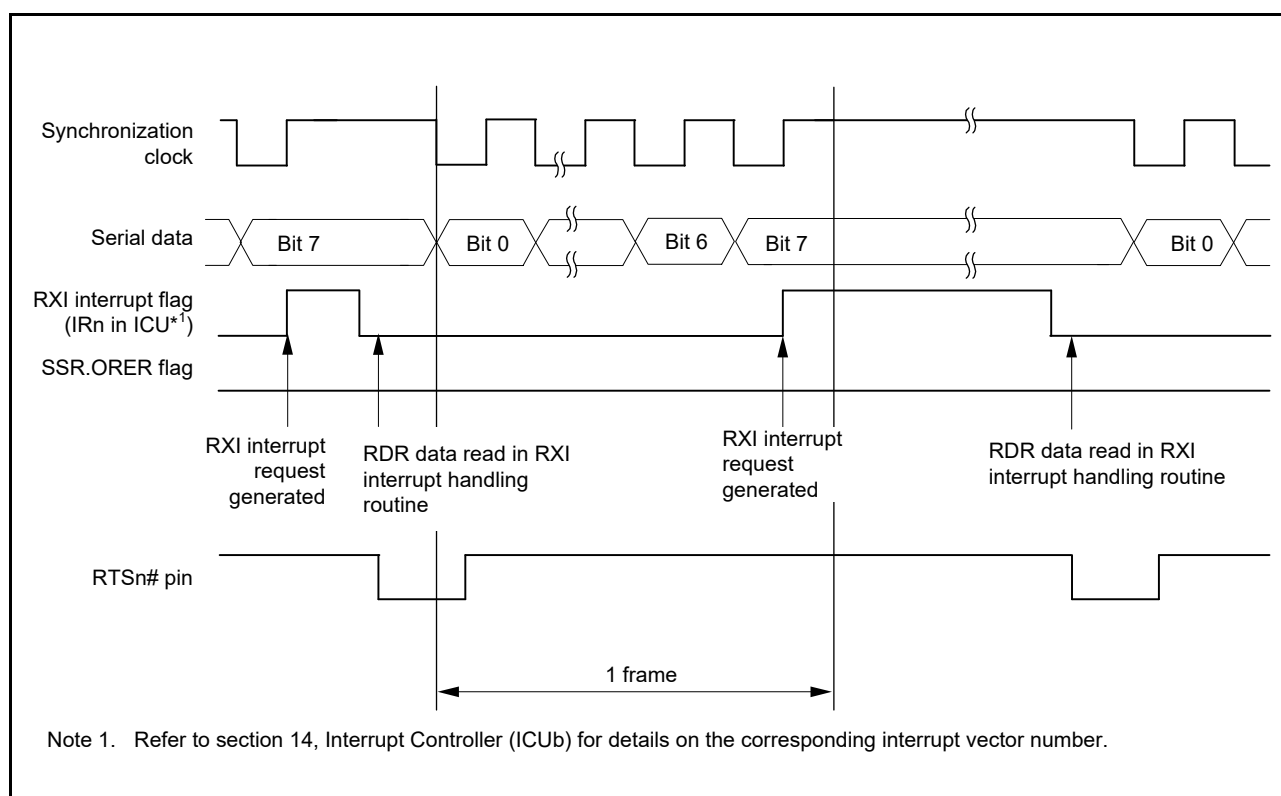


Figure 26.30 Example of Operation for Serial Reception in Clock Synchronous Mode (2) (When RTS Function is Used)

Data transfer cannot be resumed while a receive error flag is 1. Accordingly, clear the ORER, FER, and PER flags in the SSR register before resuming reception. Moreover, be sure to read the RDR register during overrun error processing. When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read the RDR register because received data which has not yet been read may be left in the RDR register.

Figure 26.31 shows a sample flowchart for serial data reception.

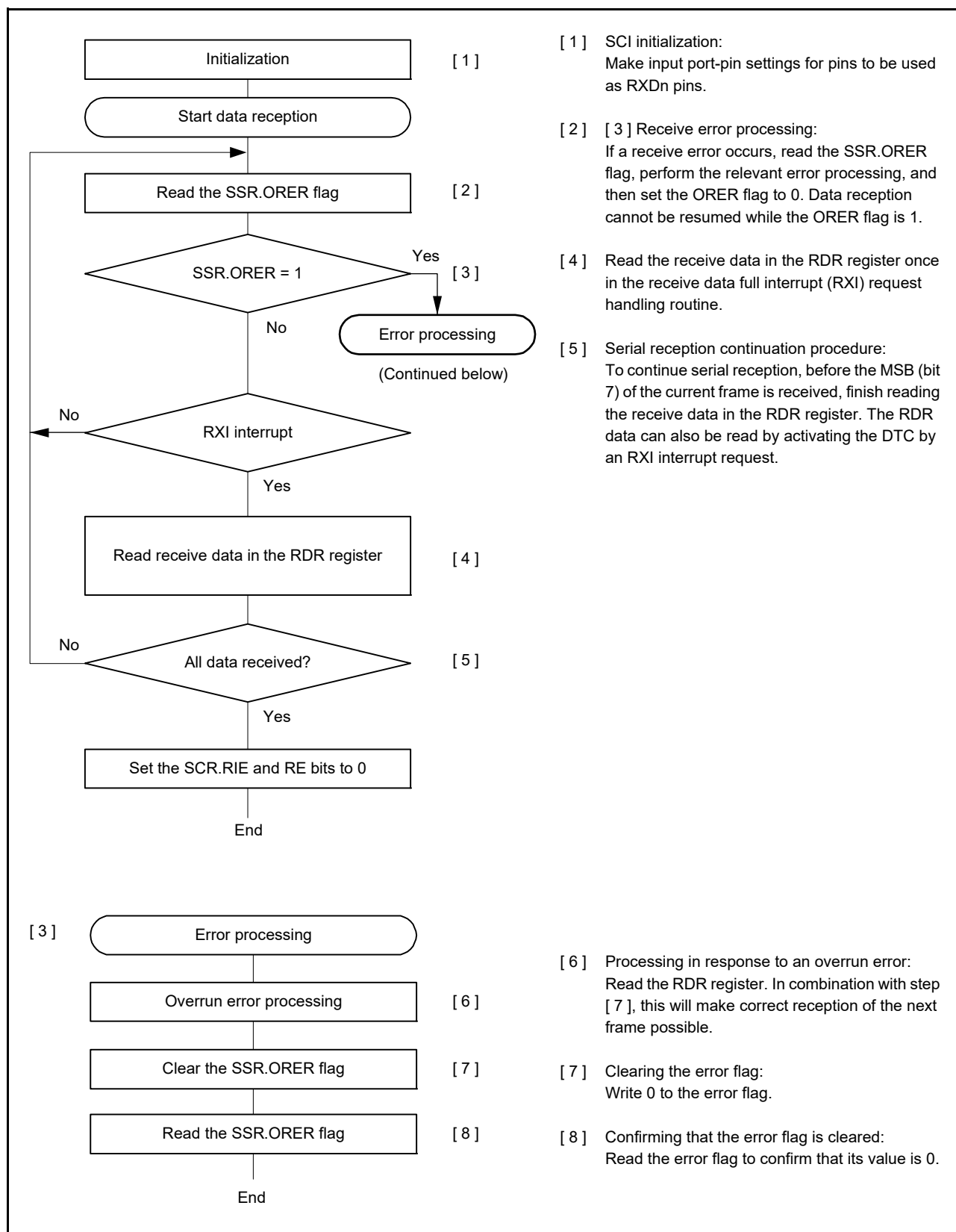


Figure 26.31 Example Flowchart of Serial Reception in Clock Synchronous Mode

26.5.6 Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode)

Figure 26.32 shows a sample flowchart for simultaneous serial transmit and receive operations in clock synchronous mode.

After initializing the SCI, the following procedure should be used for simultaneous serial data transmit and receive operations.

To switch from transmit mode to simultaneous transmit and receive mode, check that the SCI has finished transmission by reading that the SSR.TEND flag is 1, and then initialize the SCR register. Then set the TIE, RIE, TE, and RE bits in the SCR register to 1 simultaneously by a single instruction.

To switch from receive mode to simultaneous transmit and receive mode, check that the SCI has finished reception, and then set the RIE and RE bits to 0. Then check that the receive error flags (ORER, FER, and PER in the SSR register) are 0, and then set the TIE, RIE, TE, and RE bits in the SCR register to 1 simultaneously by a single instruction.

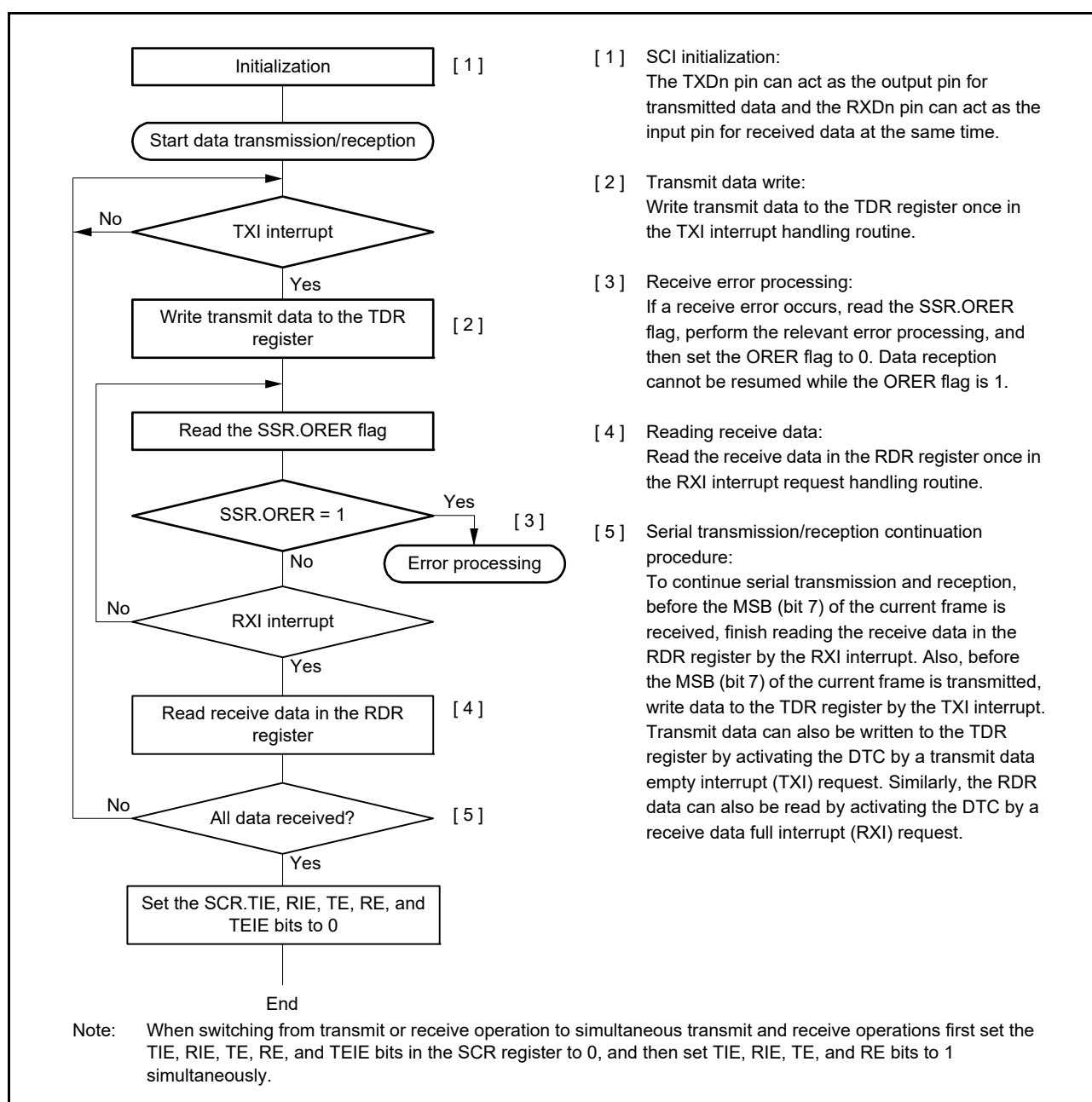


Figure 26.32 Example Flowchart of Simultaneous Serial Transmission and Reception in Clock Synchronous Mode

26.6 Operation in Smart Card Interface Mode

The SCI supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards), as an extended function of the SCI.

Smart card interface mode can be selected using the appropriate register.

26.6.1 Sample Connection

Figure 26.33 shows a sample connection between a smart card (IC card) and this MCU.

As in the figure, since this MCU communicates with an IC card using a single transmission line, interconnect the TXDn and RXDn pins and pull up the data transmission line to VCC using a resistor.

Setting the TE and RE bits in the SCR register to 1 with an IC card disconnected enables closed transmission/reception allowing self-diagnosis.

To supply an IC card with the clock pulses generated by the SCI, input the SCKn pin output to the CLK pin of an IC card. The output port of the this MCU can be used to output a reset signal.

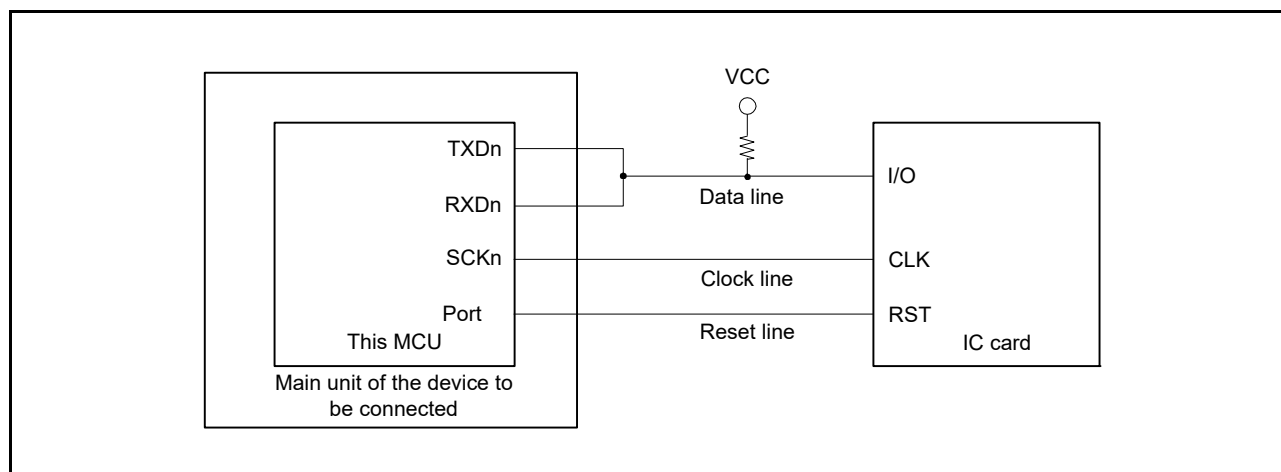


Figure 26.33 Sample Connection with a Smart Card (IC Card)

26.6.2 Data Format (Except in Block Transfer Mode)

Figure 26.34 shows the data transfer formats in smart card interface mode.

- One frame consists of 8-bit data and a parity bit in asynchronous mode.
- During transmission, at least 2 etu (elementary time unit: time required for transferring 1 bit) is secured as a guard time from the end of the parity bit until the start of the next frame.
- If a parity error is detected during reception, a low-level error signal is output for 1 etu after 10.5 etu has passed from the start bit.
- If an error signal is sampled during transmission, the same data is automatically retransmitted after at least 2 etu.

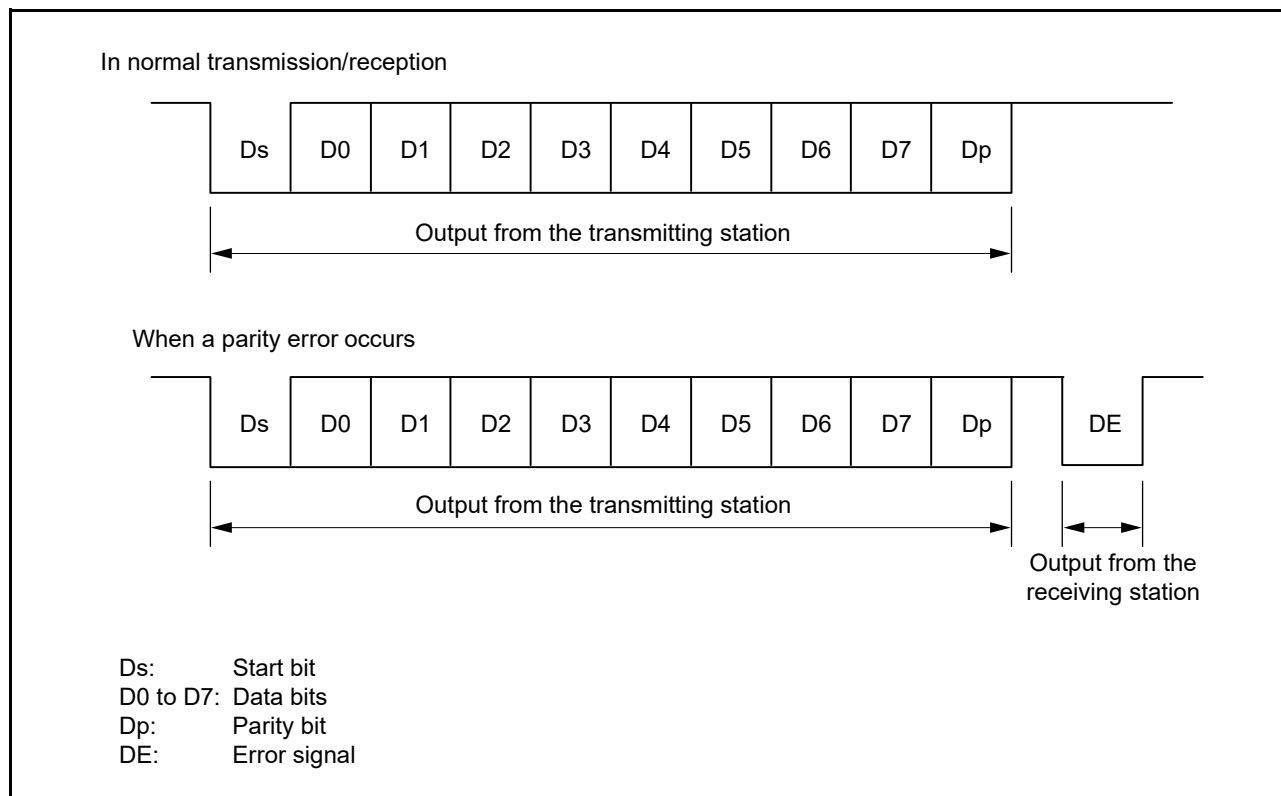


Figure 26.34 Data Formats in Smart Card Interface Mode

For communications with IC cards of the direct convention type and inverse convention type, follow the procedure below.

(1) Direct Convention Type

For the direct convention type, logic levels 1 and 0 correspond to states Z and A, respectively, and data is transferred with LSB first as the start character, as shown in Figure 26.35. Therefore, data in the start character in the figure is 3Bh. When using the direct convention type, write 0 to both the SDIR and SINV bits in the SCMR register. Write 0 to the SMR.PM bit in order to use even parity, which is prescribed by the smart card standard.

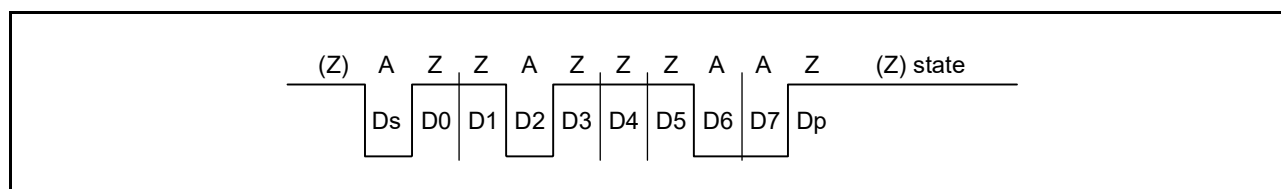


Figure 26.35 Direct Convention (SCMR.SDIR bit = 0, SCMR.SINV bit = 0, SMR.PM bit = 0)

(2) Inverse Convention Type

For the inverse convention type, logic levels 1 and 0 correspond to states A and Z, respectively and data is transferred with MSB first as the start character, as shown in Figure 26.36. Therefore, data in the start character in the figure is 3Fh. When using the inverse convention type, write 1 to both the SDIR and SINV bits in the SCMR register. The parity bit is logic level 0 to produce even parity, which is prescribed by the smart card standard, and corresponds to state Z. Since the SINV bit of this MCU only inverts data bits D7 to D0, write 1 to the SMR.PM bit to invert the parity bit for both transmission and reception.

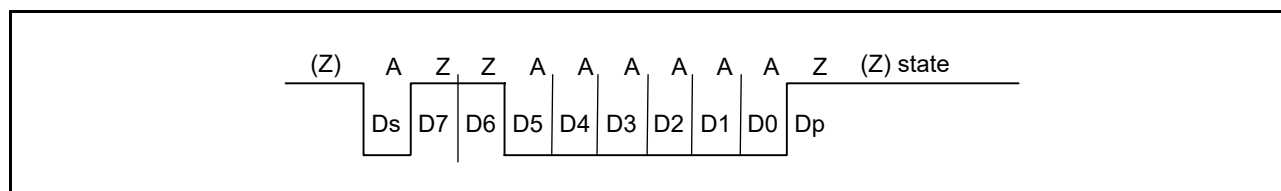


Figure 26.36 Inverse Convention (SCMR.SDIR bit = 1, SCMR.SINV bit = 1, SMR.PM bit = 1)

26.6.3 Block Transfer Mode

Block transfer mode is different from non-block transfer mode in the following respects.

- Even if a parity error is detected during reception, no error signal is output. Since the SSR.PER flag is set by error detection, clear the PER flag before receiving the parity bit of the next frame.
- During transmission, at least 1 etu is secured as a guard time from the end of the parity bit until the start of the next frame.
- Since the same data is not retransmitted during transmission, the SSR.TEND flag is set 11.5 etu after transmission start.
- In block transfer mode, the SSR.ERS flag indicates the error signal status as in non-block transfer mode, but the flag is read as 0 because no error signal is transferred.

26.6.4 Receive Data Sampling Timing and Reception Margin

Only the base clock generated by the on-chip baud rate generator can be used as a transmit/receive clock in smart card interface mode.

In this mode, the SCI can operate on a base clock with a frequency of 32, 64, 372, 256, 93, 128, 186, or 512 times the bit rate according to the settings of the SCMR.BCP2 bit and the SMR.BCP[1:0] bits.

For data reception, the falling edge of the start bit is sampled with the base clock to perform synchronization. Receive data is sampled on the 16th, 32nd, 186th, 128th, 46th, 64th, 93rd, and 256th rising edges of the base clock so that it can be latched at the middle of each bit as shown in Figure 26.37. The reception margin here is determined by the following formula.

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100 (\%)$$

M: Reception margin (%)

N: Ratio of bit rate to clock (N = 32, 64, 372, 256)

D: Duty cycle of clock (D = 0 to 1.0)

L: Frame length (L = 10)

F: Absolute value of clock frequency deviation

Assuming values of F = 0, D = 0.5, and N = 372 in the above formula, the reception margin is determined by the formula below.

$$M = \{0.5 - 1/(2 \times 372)\} \times 100 (\%) = 49.866 (\%)$$

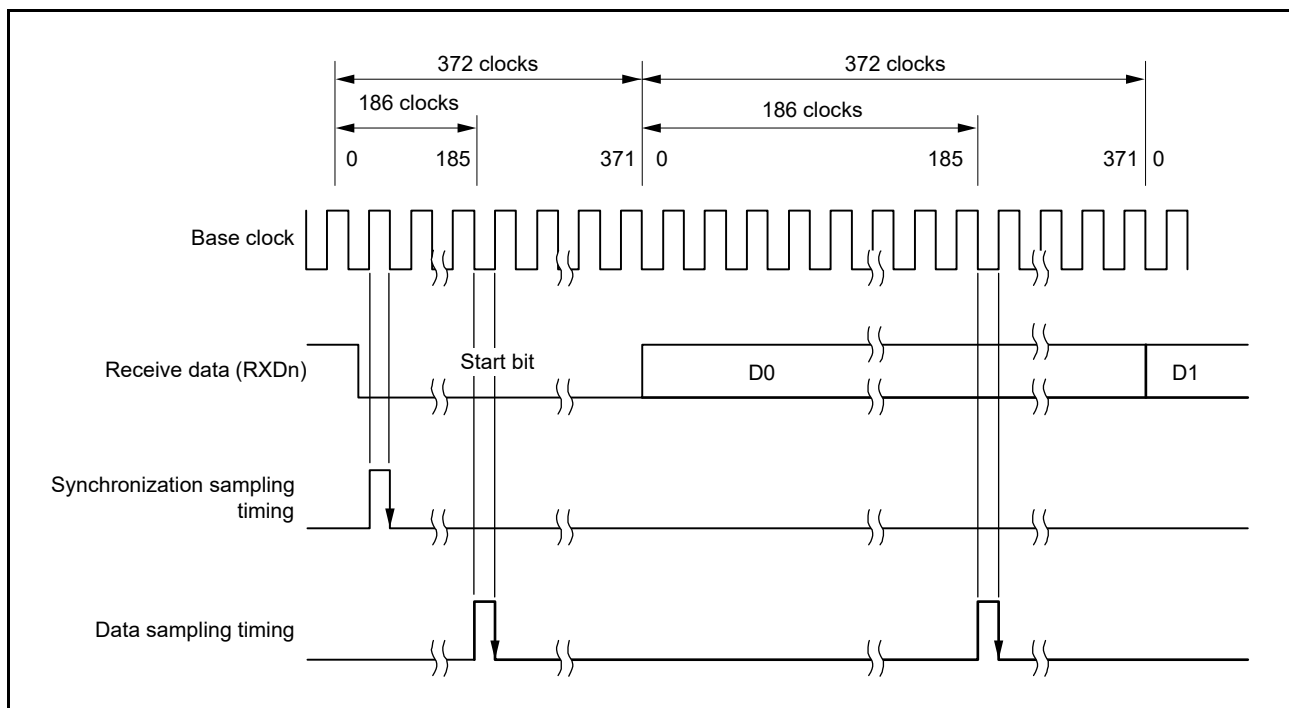


**Figure 26.37 Receive Data Sampling Timing in Smart Card Interface Mode
(When Clock Frequency is 372 Times the Bit Rate)**

26.6.5 SCI Initialization (Smart Card Interface Mode)

Initialize the SCI following the example of flowchart shown in Figure 26.38.

Initialize the SCR and SSR registers before switching from transmit mode to receive mode and vice versa. When not changing the bit rate, it is not necessary to set the CKE[1:0] bits to 00b. Even if the RE bit is set to 0, the RDR register is not initialized.

To change receive mode to transmit mode, first check that reception has completed, and then execute steps [1] and [3] in Figure 26.38. Set TE = 1 and RE = 0 at step [11]. Reception completion can be verified by reading the RXI request, SSR.ORER, or SSR.PER flag.

To change transmit mode to receive mode, first check that transmission has completed, and then execute steps [1] and [3] in Figure 26.38. Set TE = 0 and RE = 1 at step [11]. Transmission completion can be verified by reading the SSR.TEND flag.

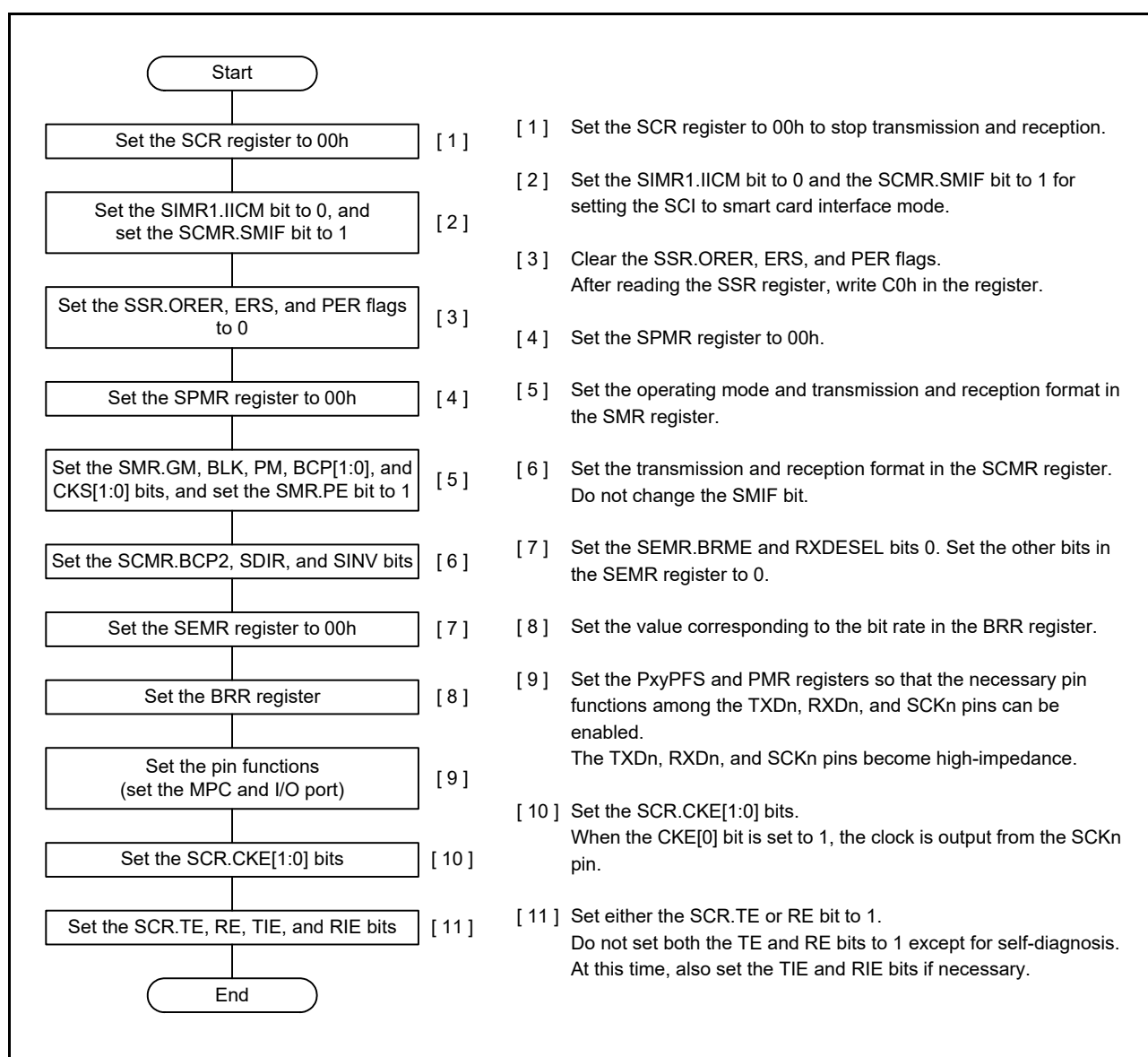


Figure 26.38 Example of SCI Initialization Flowchart (Smart Card Interface Mode)

Figure 26.39 shows an example of data transmission when the SCI is set to smart card interface mode according to the flow described in Figure 26.38 after a reset. When the pin functions are set to the SCK and TXD pins, they are still high-impedance because the SCR.CKE[0] and SCR.TE bits are 0. When the CKE[0] bit is set to 1, clock is output from the SCK pin. When the transmit data is written after setting the TE bit to 1, a data transmission starts. After the TE bit is set to 1, one frame of high-impedance is output from TXD pin (internal wait time) and then the data transmission starts. In smart card interface mode, the clock is continuously output while the CKE[0] bit is set to 1 (clock output) even if both the TE and RE bits are set to 0.

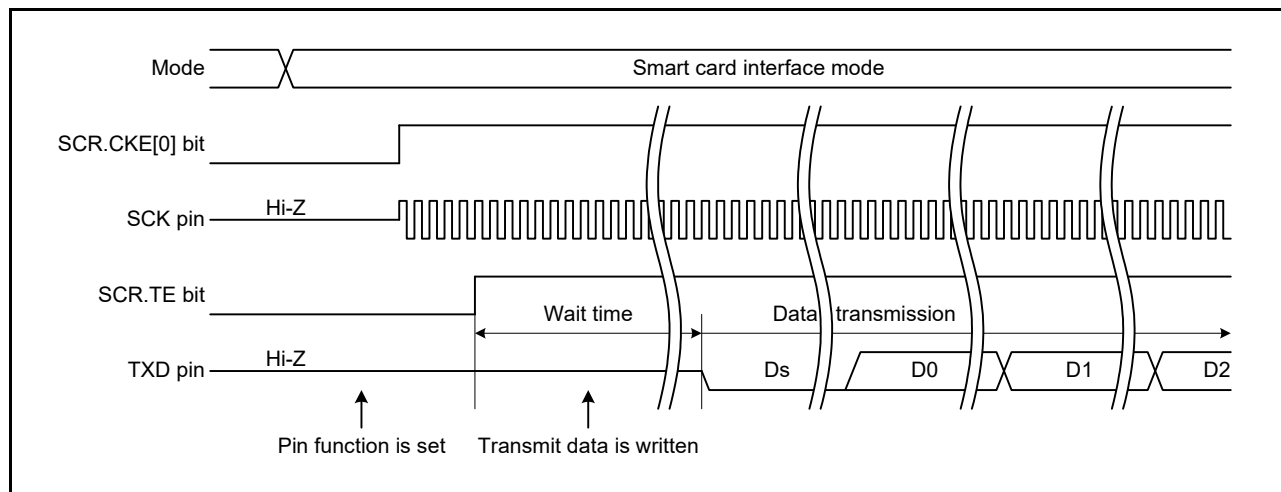


Figure 26.39 Example of Data Transmission Timing in Smart Card Interface Mode

26.6.6 Serial Data Transmission (Except in Block Transfer Mode)

Serial data transmission in smart card interface mode (except in block transfer mode), in that an error signal is sampled and data can be retransmitted, is different from that in non-smart card interface mode. Figure 26.40 shows the data retransmit operation during transmission.

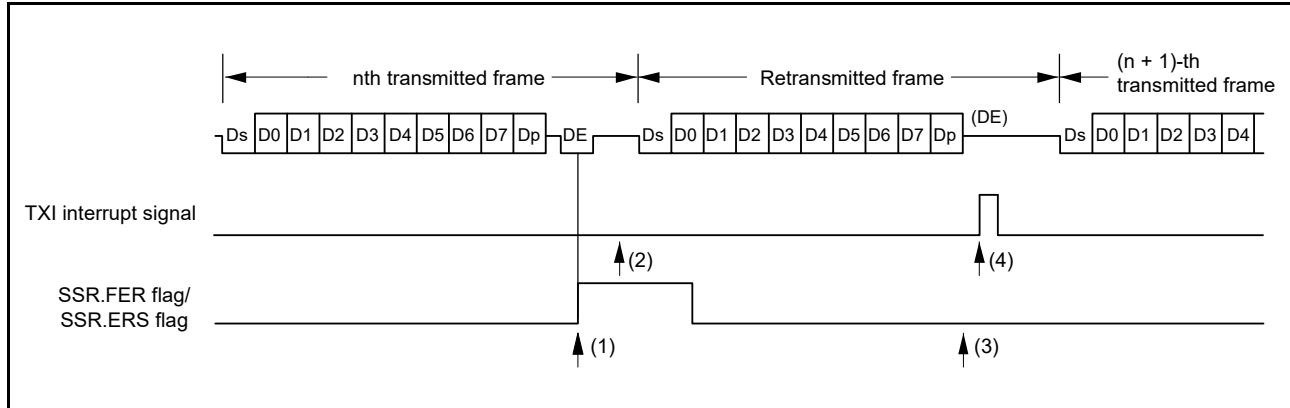


Figure 26.40 Data Retransmit Operation in SCI Transmit Mode

- (1) When an error signal from the receiver end is sampled after one-frame data has been transmitted, the SSR.ERS flag is set to 1. If the SCR.RIE bit is 1 at this time, an ERI interrupt request is generated. Clear the ERS flag before the next parity bit is sampled.
- (2) For a frame in which an error signal is received, the SSR.TEND flag is not set. Data is retransferred from the TDR register to the TSR register allowing automatic data retransmission.
- (3) If no error signal is returned from the receiver, the ERS flag is not set to 1.
- (4) In this case, the SCI judges that transmission of one-frame data (including retransmission) has been completed, and the TEND flag is set. If the SCR.TIE bit is 1 at this time, a TXI interrupt request is generated. Writing transmit data to the TDR register starts transmission of the next data.

Figure 26.41 shows a sample flowchart of serial transmission.

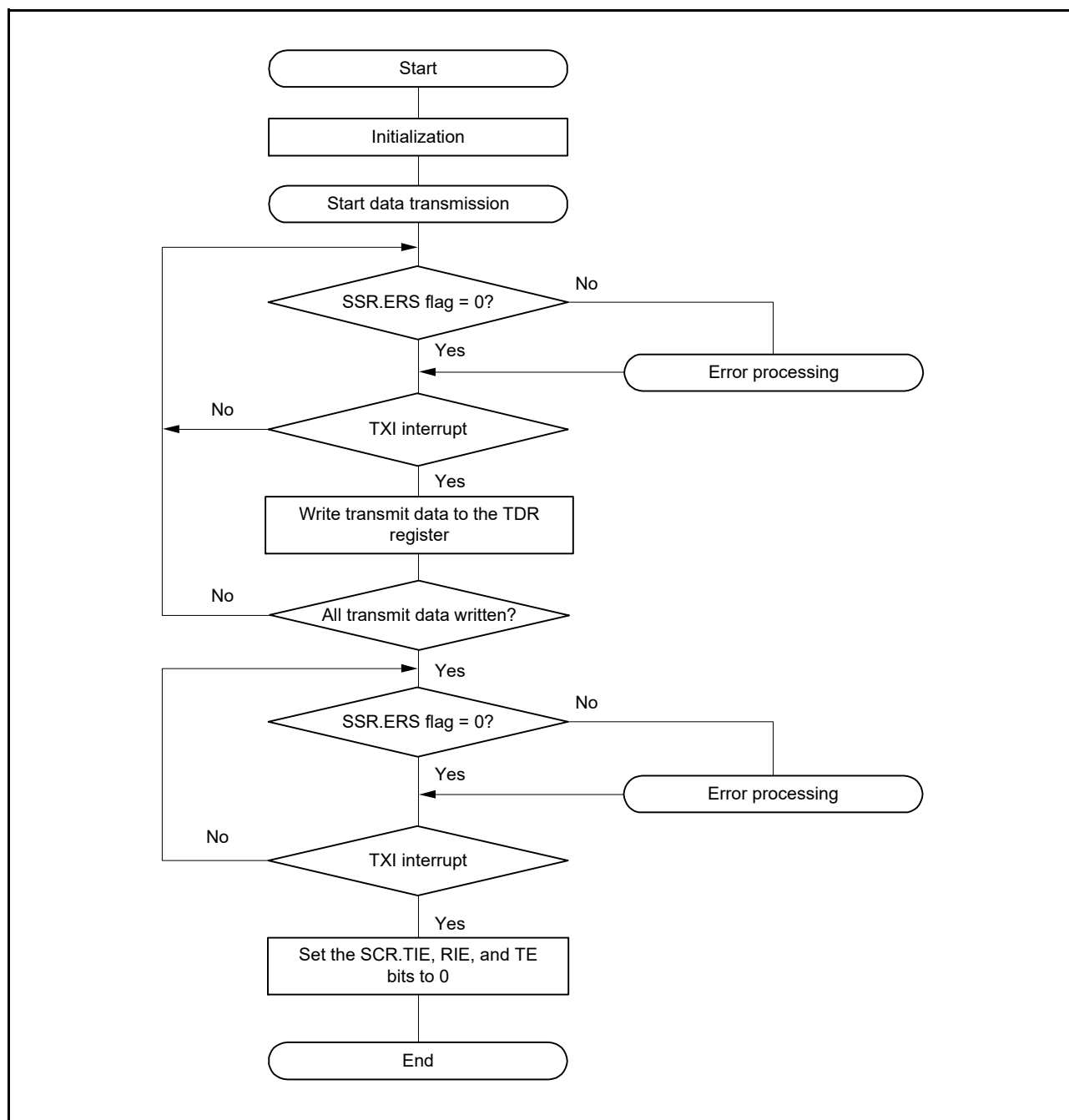


Figure 26.41 Sample Smart Card Interface Transmission Flowchart

All the processing steps are automatically performed using a TXI interrupt request to activate the DTC. When the SSR.TEND flag is set to 1 in transmission, if the SCR.TIE bit is 1, a TXI interrupt request is generated. The DTC is activated by a TXI interrupt request if the TXI interrupt request is specified as a source of DTC activation beforehand, allowing transfer of transmit data. The TEND flag is automatically set to 0 when the DTC transfers the data. If an error occurs, the SCI automatically retransmits the same data. During this retransmission, the TEND flag is kept to 0 and the DTC is not activated. Therefore, the SCI and DTC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, since the ERS flag is not automatically cleared, set the RIE bit to 1 beforehand to enable an ERI interrupt request to be generated at error occurrence, and clear the ERS flag.

When transmitting/receiving data using the DTC, be sure to make settings to enable the DTC before making SCI settings. For DTC settings, refer to **section 16, Data Transfer Controller (DTCb)**.

Note that the SSR.TEND flag is set in different timings depending on the SMR.GM bit setting. Figure 26.42 shows the TEND flag generation timing.

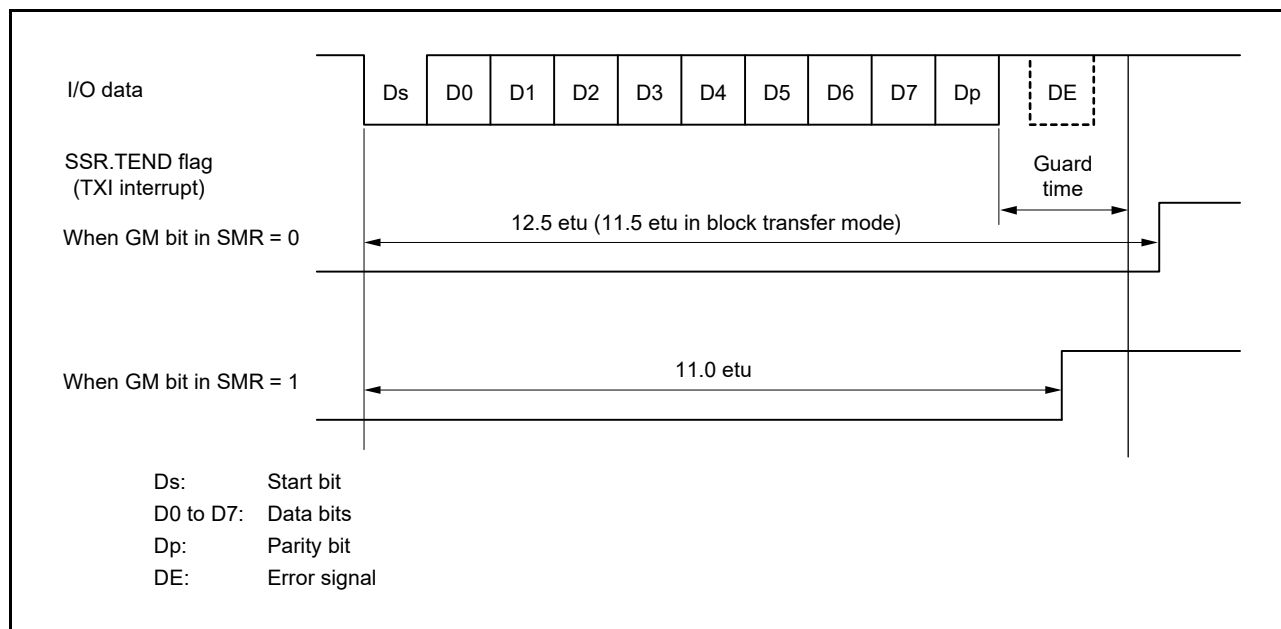


Figure 26.42 SSR.TEND Flag Generation Timing during Transmission

26.6.7 Serial Data Reception (Except in Block Transfer Mode)

Serial data reception in smart card interface mode is similar to that in non-smart card interface mode. Figure 26.43 shows the data retransmit operation in receive mode.

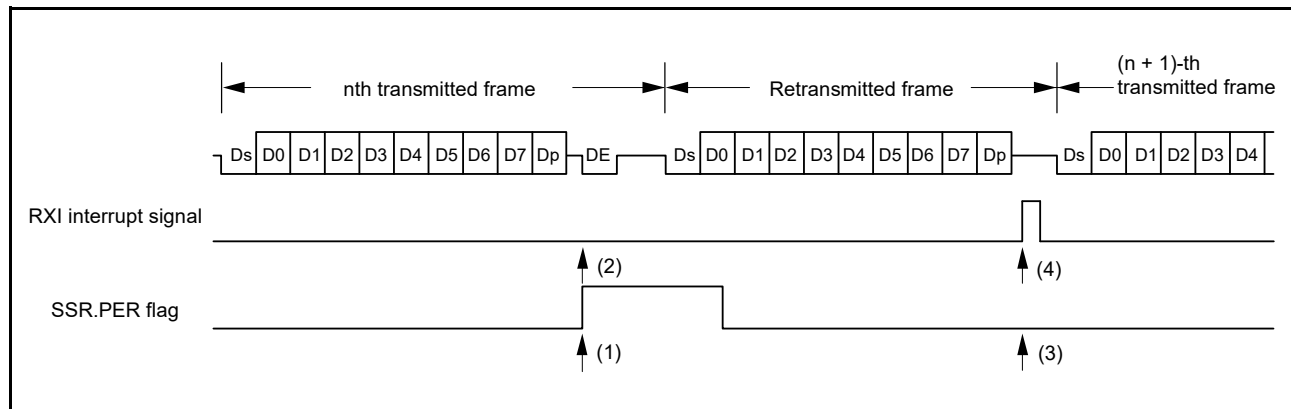


Figure 26.43 Data Retransmit Operation in SCI Receive Mode (Data Retransmit Operation during Reception)

- (1) If a parity error is detected in receive data, the SSR.PER flag is set to 1. When the SCR.RIE bit is 1 at this time, an ERI interrupt request is generated. Clear the PER flag before the next parity bit is sampled.
- (2) For a frame in which a parity error is detected, no RXI interrupt is generated.
- (3) When no parity error is detected, the SSR.PER flag is not set to 1.
- (4) In this case, data is determined to have been received successfully. When the SCR.RIE bit is 1, an RXI interrupt request is generated.

Figure 26.44 shows a sample flowchart for serial data reception.

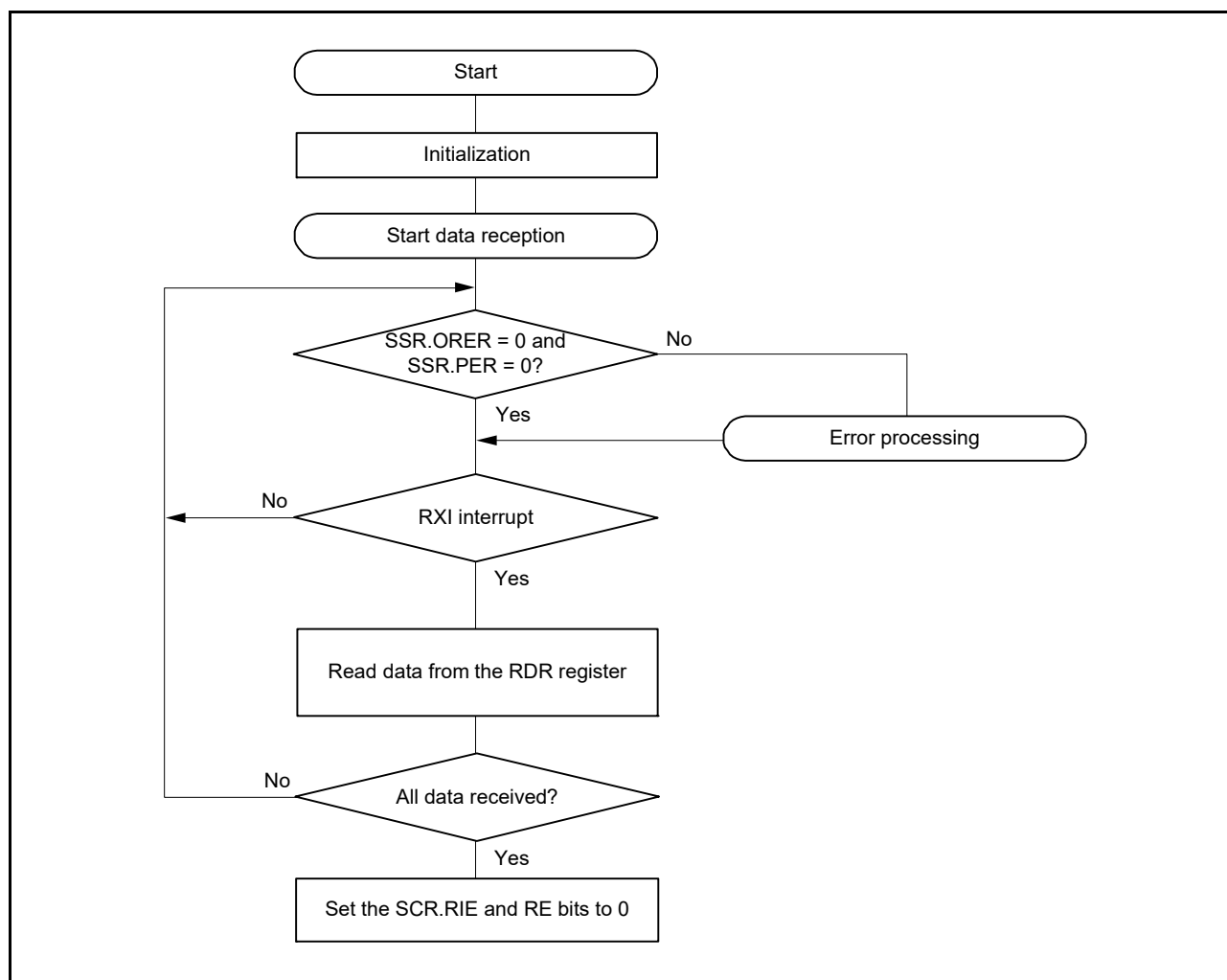


Figure 26.44 Sample Smart Card Interface Reception Flowchart

All the processing steps are automatically performed using an RXI interrupt request to activate the DTC.

In reception, setting the RIE bit to 1 allows an RXI interrupt request to be generated. The DTC is activated by an RXI interrupt request if the RXI interrupt request is specified as a source of DTC activation beforehand, allowing transfer of receive data.

If an error occurs during reception and either the ORER or PER flag in the SSR register is set to 1, a receive error interrupt (ERI) request is generated. Clear the error flag after the error occurrence. If an error occurs, the DTC is not activated and receive data is skipped. Therefore, the number of bytes of receive data specified in the DTC is transferred. Even if a parity error occurs and the PER flag is set to 1 during reception, receive data is transferred to the RDR register, thus allowing the data to be read.

When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read the RDR register because the received data which has not yet been read may be left in the RDR register.

Note 1. For operations in block transfer mode, refer to section 26.3, Operation in Asynchronous Mode.

26.6.8 Clock Output Control

Clock output can be fixed to high or low using the SCR.CKE[1:0] bits when the SMR.GM bit is 1. When the CKE[1:0] bits are set to 01b (clock output), the base clock is output from the SCK pin. For the settings of the base clock frequency (bit rate), refer to section 26.2.11, Bit Rate Register (BRR). When the CKE[1:0] bits are set to 00b (output fixed low) or 10b (output fixed to high), the SCK pin can be fixed to low or high.

Figure 26.45 shows a timing chart when the clock output is controlled.

If changing the CKE[1:0] bits while the SMR.GM bit is 0 (non-GSM mode), a pulse of unexpected width may output from SCK pin because the result is immediately reflected to the SCK pin.

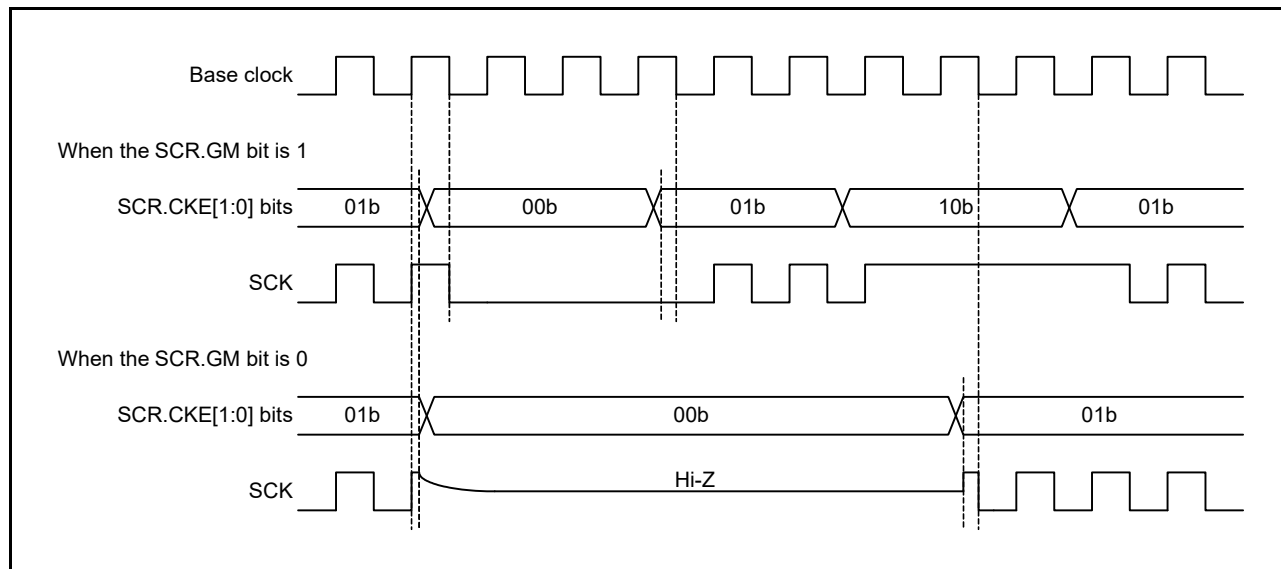


Figure 26.45 Clock Output Control

26.7 Operation in Simple I²C Mode

Simple I²C-bus format is composed of 8 data bits and an acknowledge bit. By continuing into a slave-address frame after a start condition or restart condition, a master device is able to specify a slave device as the partner for communications. The currently specified slave device remains valid until a new slave device is specified or a stop condition is satisfied. The 8 data bits in all frames are transmitted in order from the MSB.

The I²C-bus format and timing of the I²C-bus are shown in Figure 26.46 and Figure 26.47.

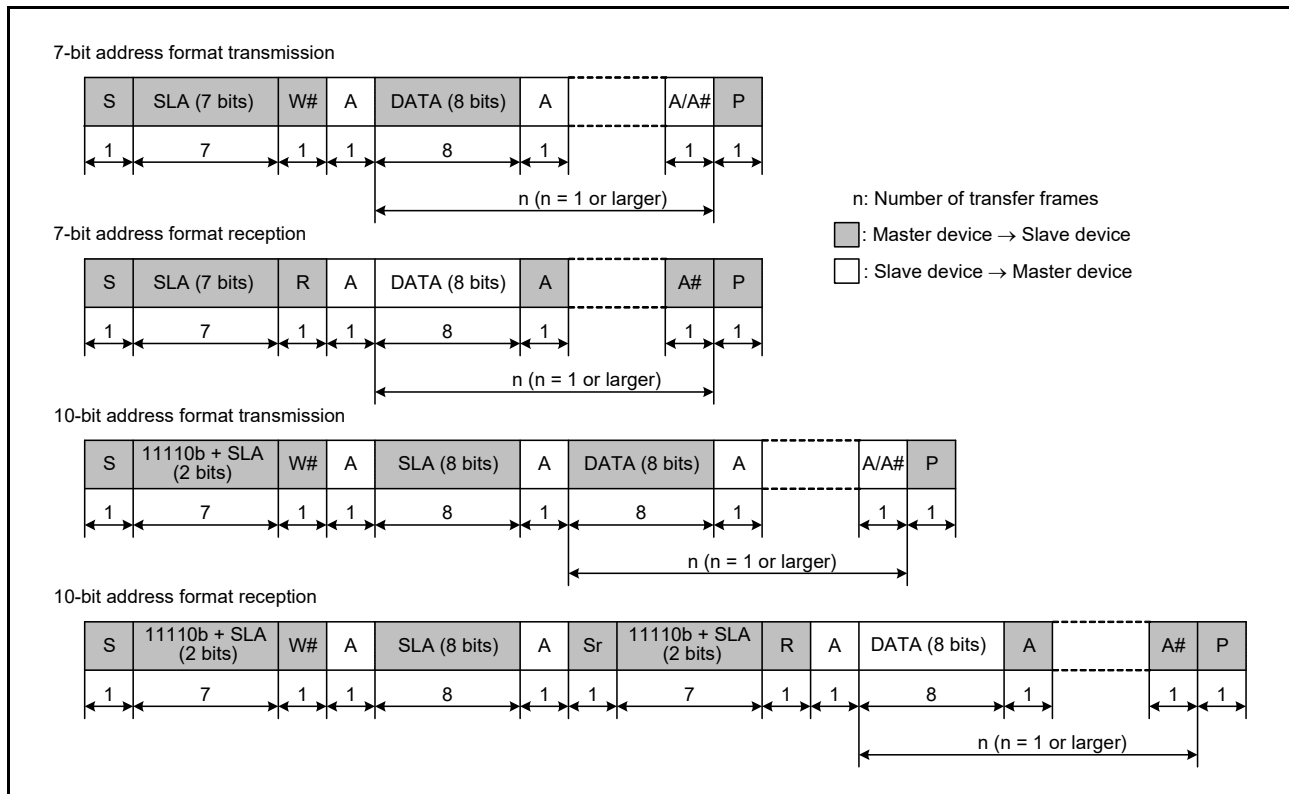


Figure 26.46 I²C-bus Format

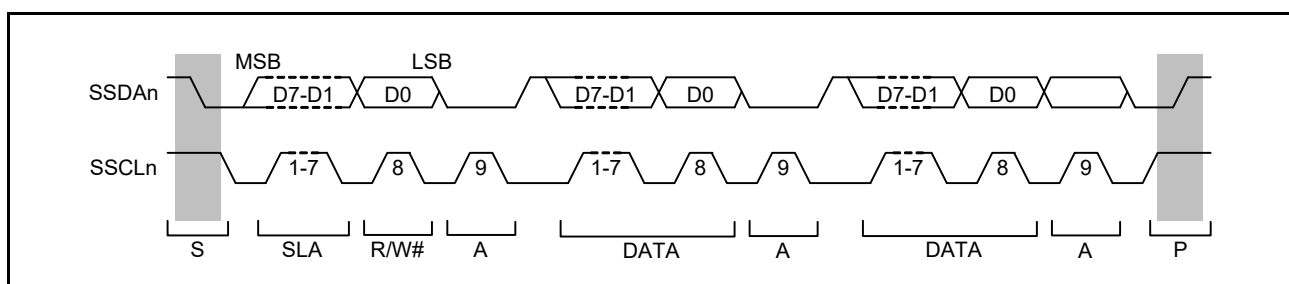


Figure 26.47 I²C-bus Timing (When SLA is 7 Bits)

- S: Indicates a start condition, i.e. the master device changing the level on the SSDAn line from the high to the low level while the SSCLn line is at the high level.
- SLA: Indicates a slave address, by which the master device selects a slave device.
- R/W#: Indicates the direction of transfer (reception or transmission). The value 1 corresponds to transfer from the slave device to the master device and 0 corresponds to transfer from the master device to the slave device.
- A/A#: Indicates an acknowledge bit. This is returned by the slave device for master transmission and by the master device for master reception. Return of the low level indicates ACK and return of the high level indicates NACK.
- Sr: Indicates a restart condition, i.e. the master device changing the level on the SSDAn line from the high to the low level while the SSCLn line is at the high level and after the setup time has elapsed.
- DATA: Indicates the data being received or transmitted.
- P: Indicates a stop condition, i.e. the master device changing the level on the SSDAn line from the low to the high level while the SSCLn line is at the high level.

26.7.1 Generation of Start, Restart, and Stop Conditions

Writing 1 to the SIMR3.IICSTAREQ bit causes the generation of a start condition. The generation of a start condition proceeds through the following operations.

- The level on the SSDAn line falls (from the high level to the low level) and the SSCLn line is kept in the released state.
- The hold time for the start condition is secured as half of a bit period at the bit rate determined by the setting of the BRR.
- The level on the SSCLn line falls (from the high level to the low level), the SIMR3.IICSTAREQ bit is set (to 0), and a start-condition generated interrupt is output.

Writing 1 to the SIMR3.IICRSTAREQ bit causes the generation of a start condition. The generation of a start condition proceeds through the following operations.

- The SSDAn line is released and the SSCLn line is kept at the low level.
- The period at low level for the SSCLn line is secured as half of a bit period at the bit rate determined by the setting of the BRR.
- The SSCLn line is released (transition from the low to the high level).
- Once the high level on the SSCLn line is detected, the setup time for the restart condition is secured as half of a bit period at the bit rate determined by the setting of the BRR.
- The level on the SSDAn line falls (from the high level to the low level).
- The hold time for the restart condition is secured as half of a bit period at the bit rate determined by the setting of the BRR.
- The level on the SSCLn line falls (from the high level to the low level), the SIMR3.IICRSTAREQ bit is set (to 0), and a restart-condition generated interrupt is output.

Writing 1 to the SIMR3.IICSTPREQ bit causes the generation of a stop condition. The generation of a stop condition proceeds through the following operations.

- The level on the SSDAn line falls (from the high level to the low level) and the SSCLn line is kept at the low level.
- The period at low level for the SSCLn line is secured as half of a bit period at the bit rate determined by the setting of the BRR.
- The SSCLn line is released (transition from the low to the high level).
- Once the high level on the SSCLn line is detected, the setup time for the stop condition is secured as half of a bit period at the bit rate determined by the setting of the BRR.
- The SSDAn is released (transition from the low to the high level), the SIMR3.IICSTPREQ bit is set (to 0), and a stop-condition generated interrupt is output.

Figure 26.48 shows the timing of operations in the generation of start, restart, and stop conditions.

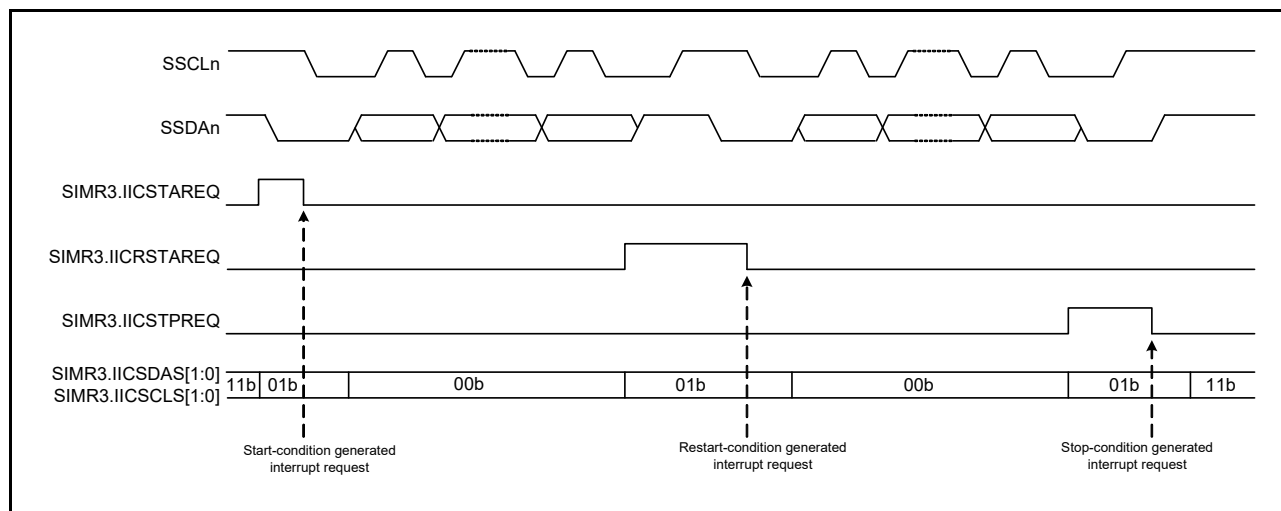


Figure 26.48 Timing of Operations in the Generation of Start, Restart, and Stop Conditions

26.7.2 Clock Synchronization

The SSCLn line may be placed at the low level in the case of a wait inserted by a slave device as the other side of transfer. Setting the SIMR2.IICCSC bit to 1 applies control to obtain synchronization when the levels of the internal SSCLn clock signal and the level being input on the SSCLn pin differ.

When the SIMR2.IICCSC bit is set to 1, the level of the internal SSCLn clock signal changes from low to high, counting to determine the period at high level is stopped while the low level is being input on the SSCLn pin, and counting to determine the period at high level starts after the transition of the input on the SSCLn pin to the high level. The interval from this time until counting to determine the period at high level starts on the transition of the SSCLn pin to the high level is the total of the delay of SSCLn output, delay for noise filtering of the input on the SSCLn pin (2 or 3 cycles of sampling clock for the noise filter), and delay for internal processing (1 or 2 cycles of PCLK). The period at high level of the internal SSCLn clock is extended even if other devices are not placing the low level on the SSCLn line. If the SIMR2.IICCSC bit is 1, synchronization is obtained for the transmission and reception of data by taking the logical AND of the input on the SSCLn pin and the internal SSCLn clock. If the SIMR2.IICCSC bit is 0, synchronization with the internal SSCLn clock is obtained for the transmission and reception of data.

If a slave device inserts a period of waiting into the interval until the transition of the internal SSCLn clock signal from the low to the high level after a request for the generation of a start, restart, or stop condition is issued, the time until generation is prolonged by that period.

If a slave device inserts a period of waiting after the transition of the internal SSCLn clock signal from the low to the high level, although the generation-completed interrupt is issued without stopping the period of waiting, generation of the condition itself is not guaranteed. Figure 26.49 shows an example of operations to synchronize the clocks.

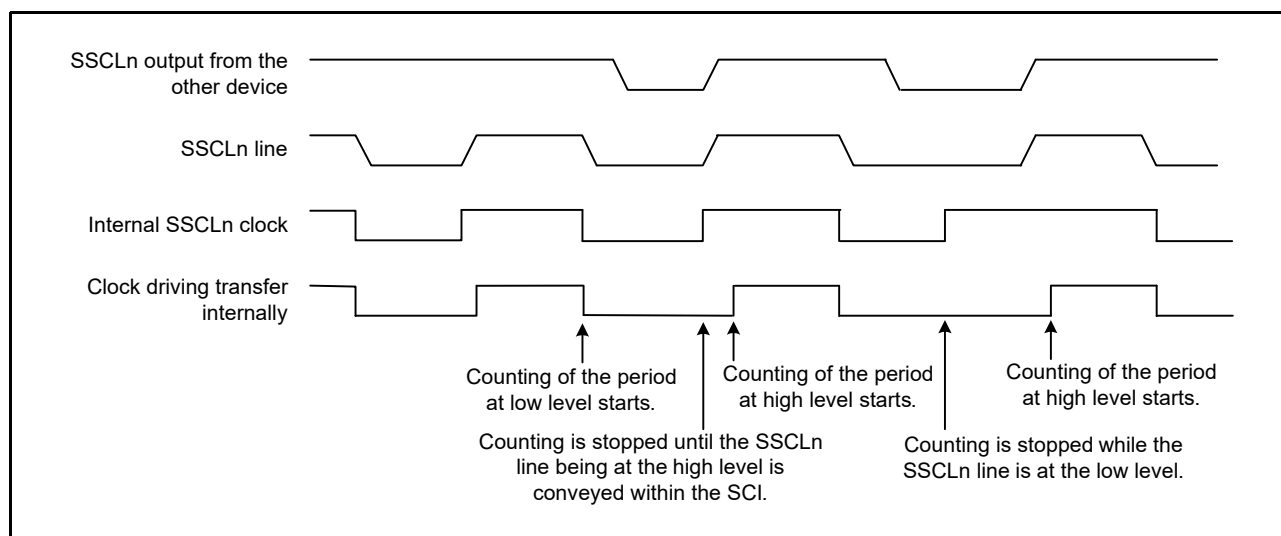


Figure 26.49 Example of Operations for Clock Synchronization

26.7.3 SSDA Output Delay

The SIMR1.IICDL[4:0] bits can be used to set a delay for output on the SSDAn pin relative to falling edges of output on the SSCLn pin. Delay-time settings from 0 to 31 are selectable, representing periods of the corresponding numbers of cycles of the clock signal from the on-chip baud rate generator (derived by frequency-dividing the base clock, PCLK, by the divisor selected by the SMR.CKS[1:0] bits). A delay for output on the SSDAn pin is for the start condition/restart condition/stop condition signal, 8-bit transmit data, and an acknowledge bit.

If the SSDA output delay is shorter than the time for the level on the SSCLn pin to fall, the change of the output on the SSDAn pin will start while the output level on the SSCLn pin is falling, creating a possibility of erroneous operation for slave devices. Ensure that settings for the delay of output on the SSDAn pin are for times greater than the time output on the SSCLn pin takes to fall (300 ns for I²C-bus in normal mode and fast mode).

Figure 26.50 shows the timing of delays in SSDA output.

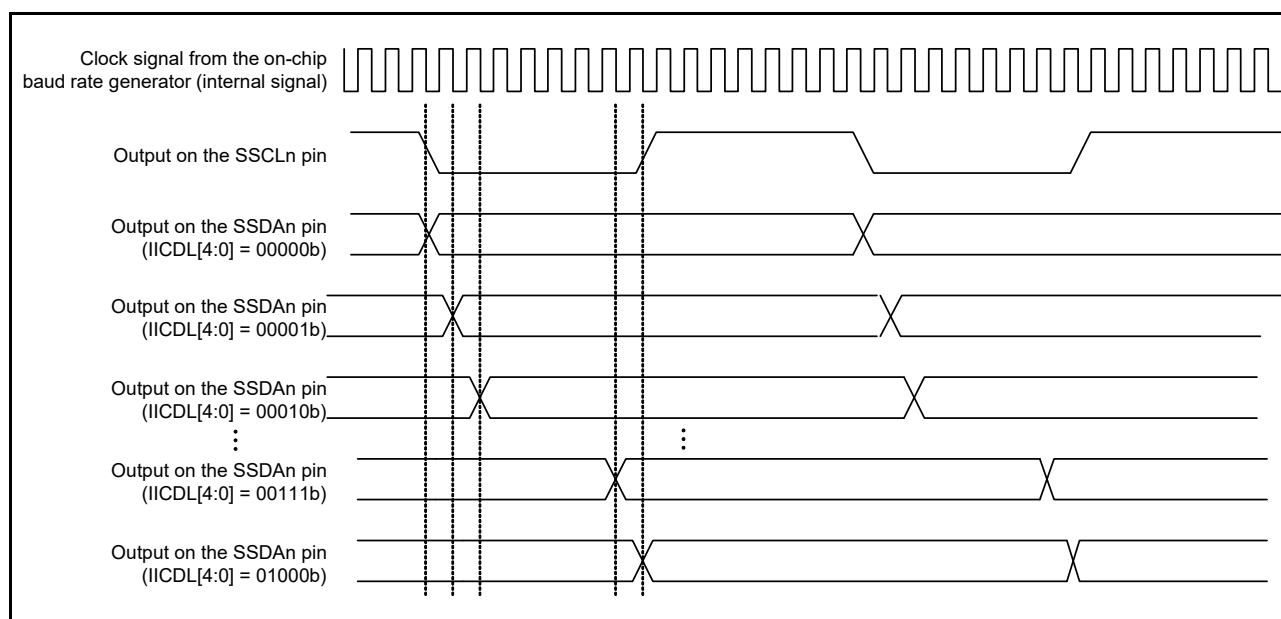


Figure 26.50 Timing of Delays in SSDA Output

26.7.4 SCI Initialization (Simple I²C Mode)

Before transferring data, write the initial value (00h) to the SCR register and initialize the interface following the example shown in Figure 26.51.

When changing the operating mode, transfer format, and so on, be sure to set the SCR register to its initial value before proceeding with the changes.

In simple I²C mode, the open-drain setting for the communication ports should be made on the port side.

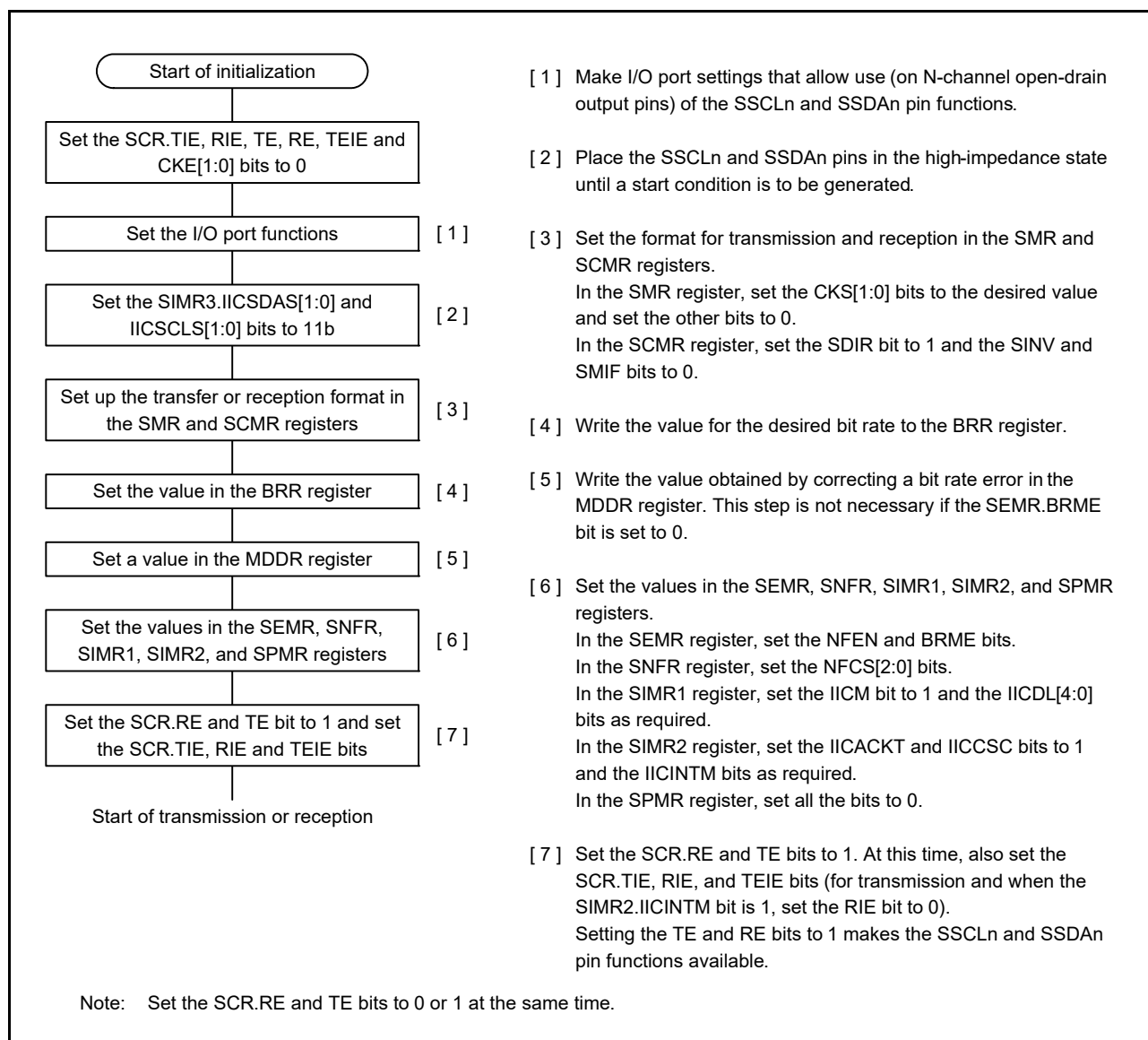


Figure 26.51 Example of the Flowchart of SCI Initialization (for Simple I²C Mode)

26.7.5 Operation in Master Transmission (Simple I²C Mode)

Figure 26.52 and Figure 26.53 show examples of operations in master transmission and Figure 26.54 is a flowchart showing the procedure for data transmission. Refer to Table 26.33 for more information on the STI interrupt.

When 10-bit slave addresses are in use, steps [3] and [4] in Figure 26.54 are repeated twice.

In simple I²C mode, the transmit data empty interrupt (TXI) is generated when communication of one frame is completed, unlike the TXI interrupt request generation timing during clock synchronous transmission.

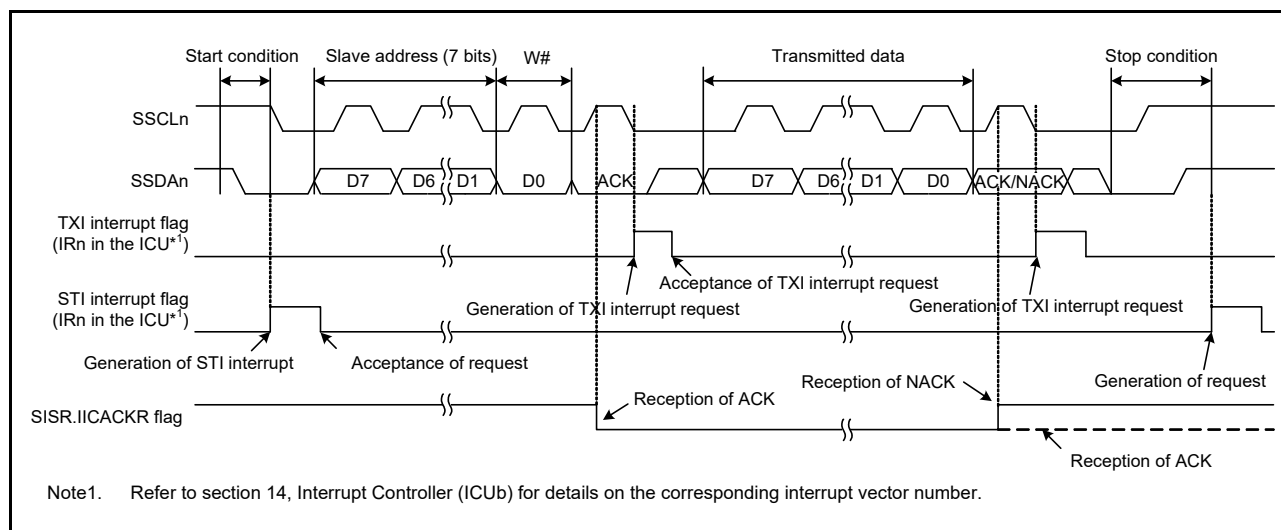


Figure 26.52 Example 1 of Operations for Master Transmission in Simple I²C-bus Mode (with 7-Bit Slave Addresses, Transmission Interrupts, and Reception Interrupts in Use)

When the SIMR2.IICINTM bit is set to 0 (use ACK/NACK interrupts) during master transmission, the DTC is activated by the ACK interrupt as the trigger and necessary number of data bytes are transmitted. When the NACK is received, error processing, such as transmission stop and retransmission, is performed by the NACK interrupt as the trigger.

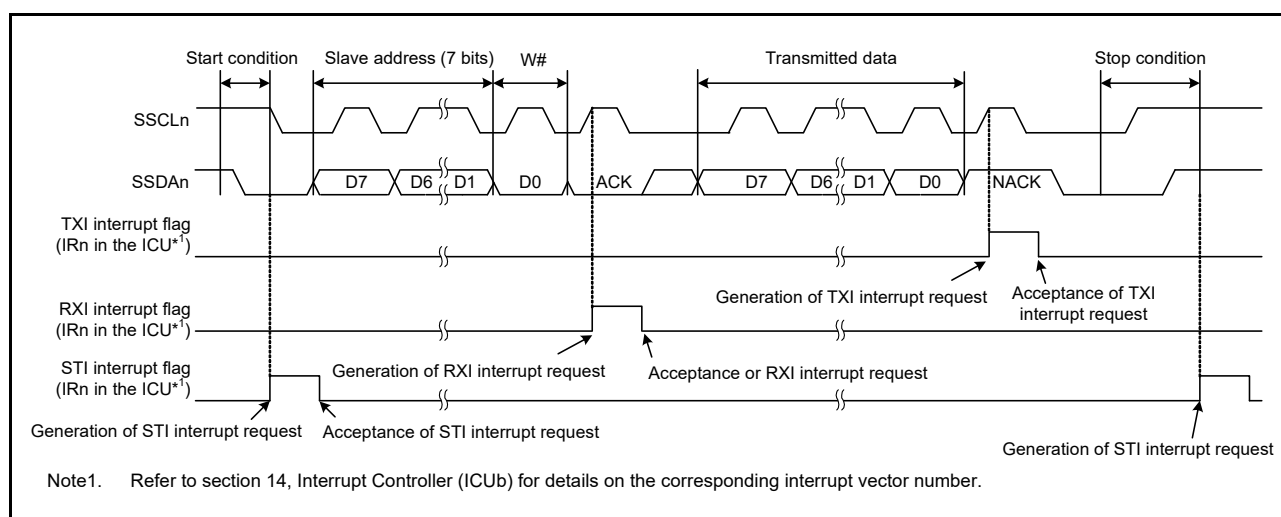


Figure 26.53 Example 2 of Operations for Master Transmission in Simple I²C-bus Mode (with 7-Bit Slave Addresses, ACK Interrupts, and NACK Interrupts in Use)

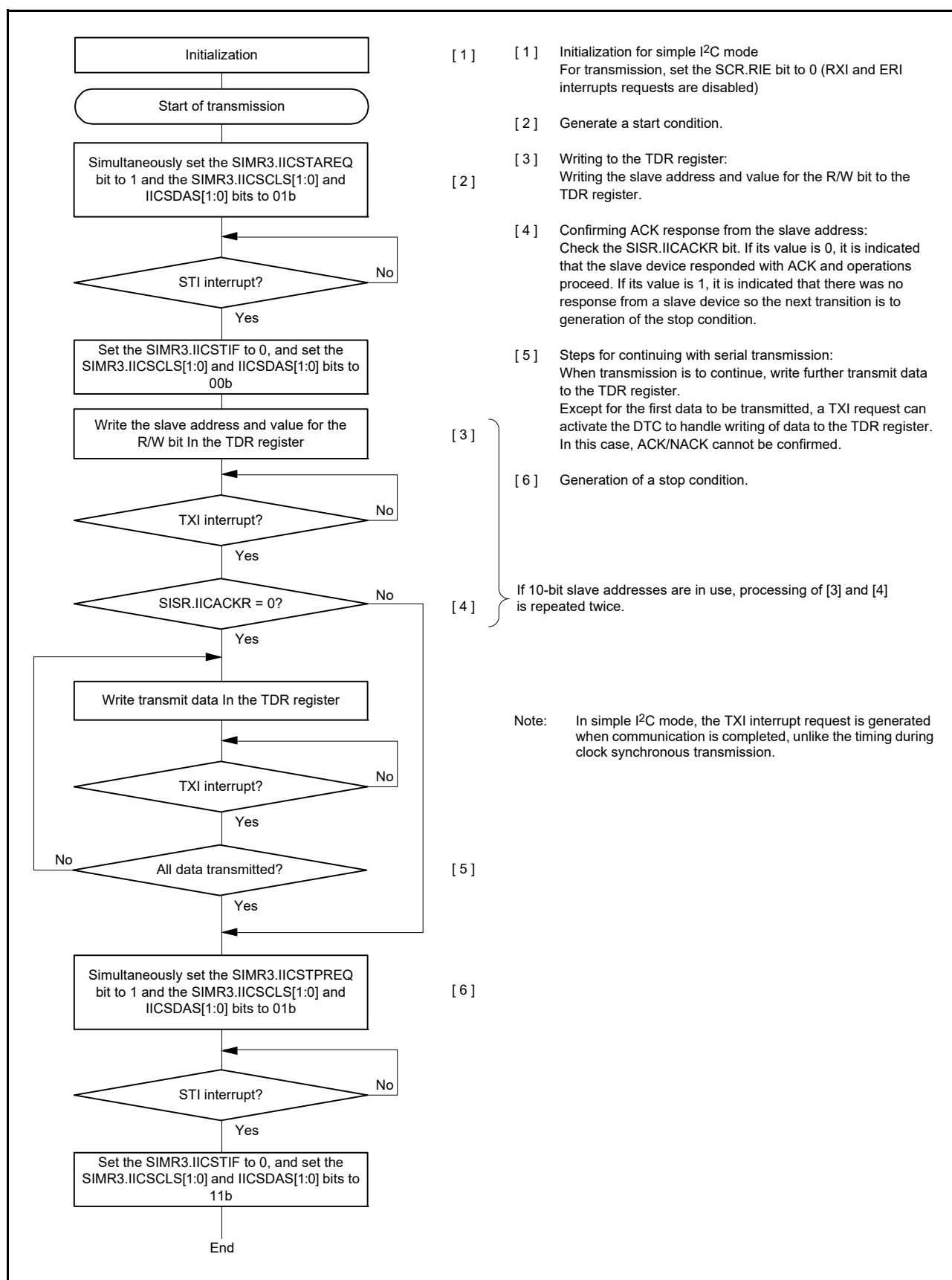


Figure 26.54 Example of the Procedure for Master Transmission Operations in Simple I²C Mode (with Transmission Interrupts and Reception Interrupts in Use)

26.7.6 Master Reception (Simple I²C Mode)

Figure 26.55 shows an example of operations in simple I²C mode master reception and Figure 26.56 is a flowchart showing the procedure for master reception.

The value of the SIMR2.IICINTM bit is assumed to be 1 (use reception and transmission interrupts).

In simple I²C mode, the transmit data empty interrupt (TXI) is generated when communication of one frame is completed, unlike the TXI interrupt request generation timing during clock synchronous transmission.

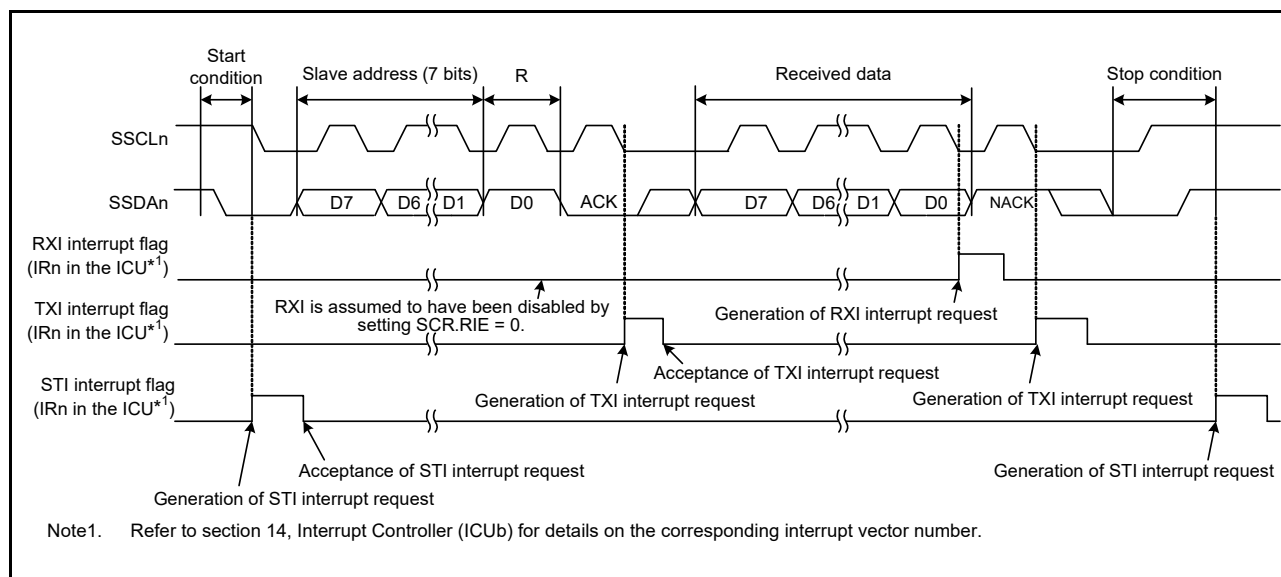


Figure 26.55 Example of Operations for Master Reception in Simple I²C-bus Mode (with 7-Bit Slave Addresses, Transmission Interrupts, and Reception Interrupts in Use)

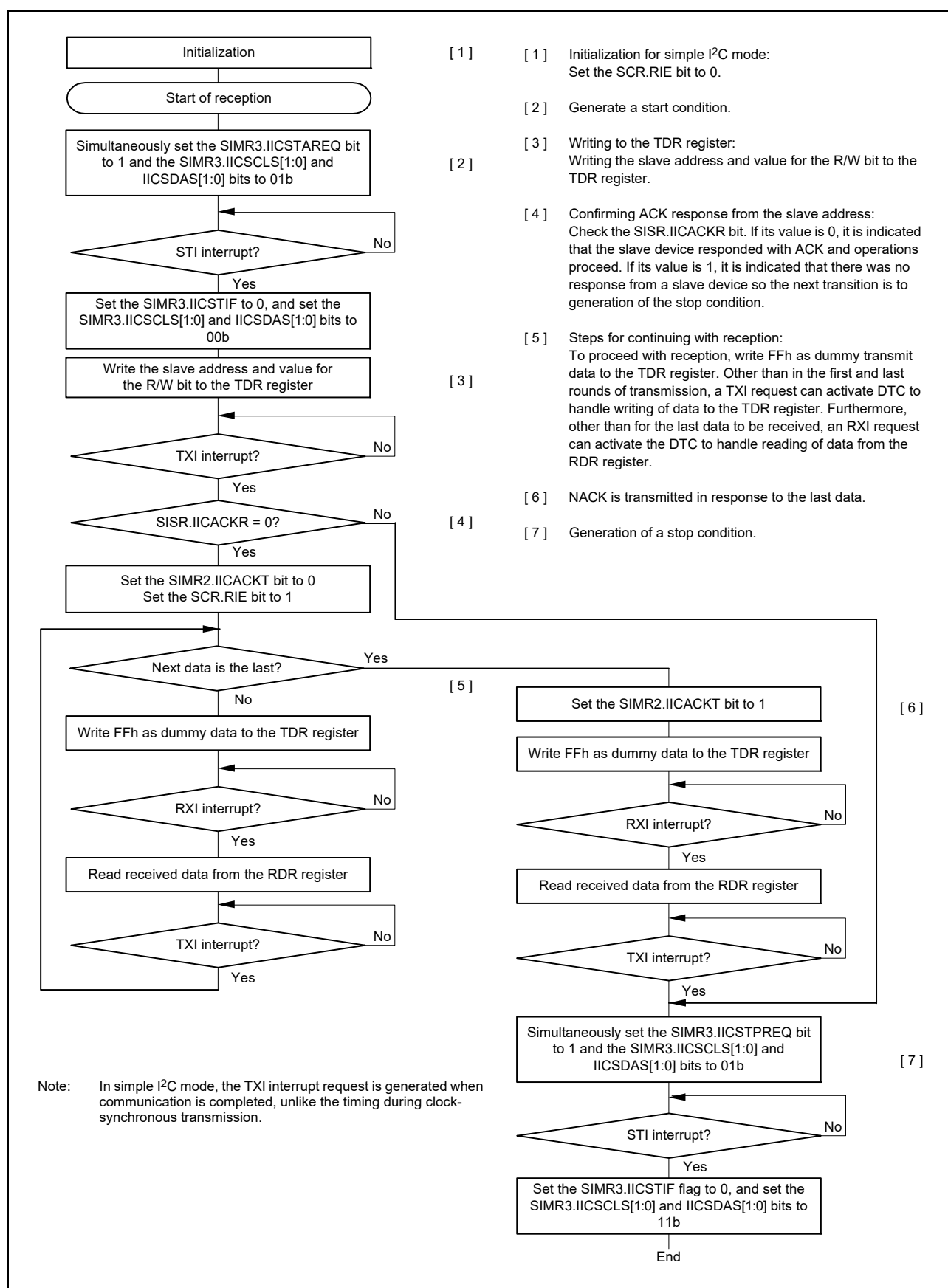


Figure 26.56 Example of the Procedure for Master Reception Operations in Simple I²C Mode (with Transmission Interrupts and Reception Interrupts in Use)

26.7.7 Recovery from Bus Hang-up

If the bus is stuck by an abnormal state in SCI because of the communication error, reset the SCI according to the following steps and release the bus.

- (1) Set the SCR.TE and RE bit to 0 at the same time to reset SCI.
- (2) Set the SIMR3 register to F0h to release the bus.
- (3) If the SSR.RDRF flag is 1, dummy-read the RDR register to clear the flag.
- (4) Set the SCR.TE and RE bit to 1 at the same time.

26.8 Operation in Simple SPI Mode

As an extended function, the SCI supports a simple SPI mode that handles transfer among one or multiple master devices and multiple slave devices.

Making the settings for clock synchronous mode (SCMR.SMIF = 0, SIMR1.IICM = 0, SMR.CM = 1) plus setting the SPMR.SSE bit to 1 places the SCI in simple SPI mode. However, the SS pin function on the master side is unnecessary for connection of the device used as the master in simple SPI mode when the configuration only has a single master, so set the SPMR.SSE bit to 0 in such cases.

Figure 26.57 shows an example of connections for simple SPI mode. Control a general port pin to produce the SS output signal from the master.

In simple SPI mode, data are transferred in synchronization with clock pulses in the same way as in clock synchronous mode. One character of data for transfer consists of 8 bits of data, and parity bits cannot be appended to this. The data can be inverted by setting the SCMR.SINV bit to 1.

Since the receiver and transmitter are independent of each other within the SCI module, full-duplex communications are possible, with a common clock signal. Furthermore, since both the transmitter and receiver have a double-buffered structure, writing of further transmit data while transmission is in progress and reading of previously received data while reception is in progress are both possible. Continuous transfer is thus possible.

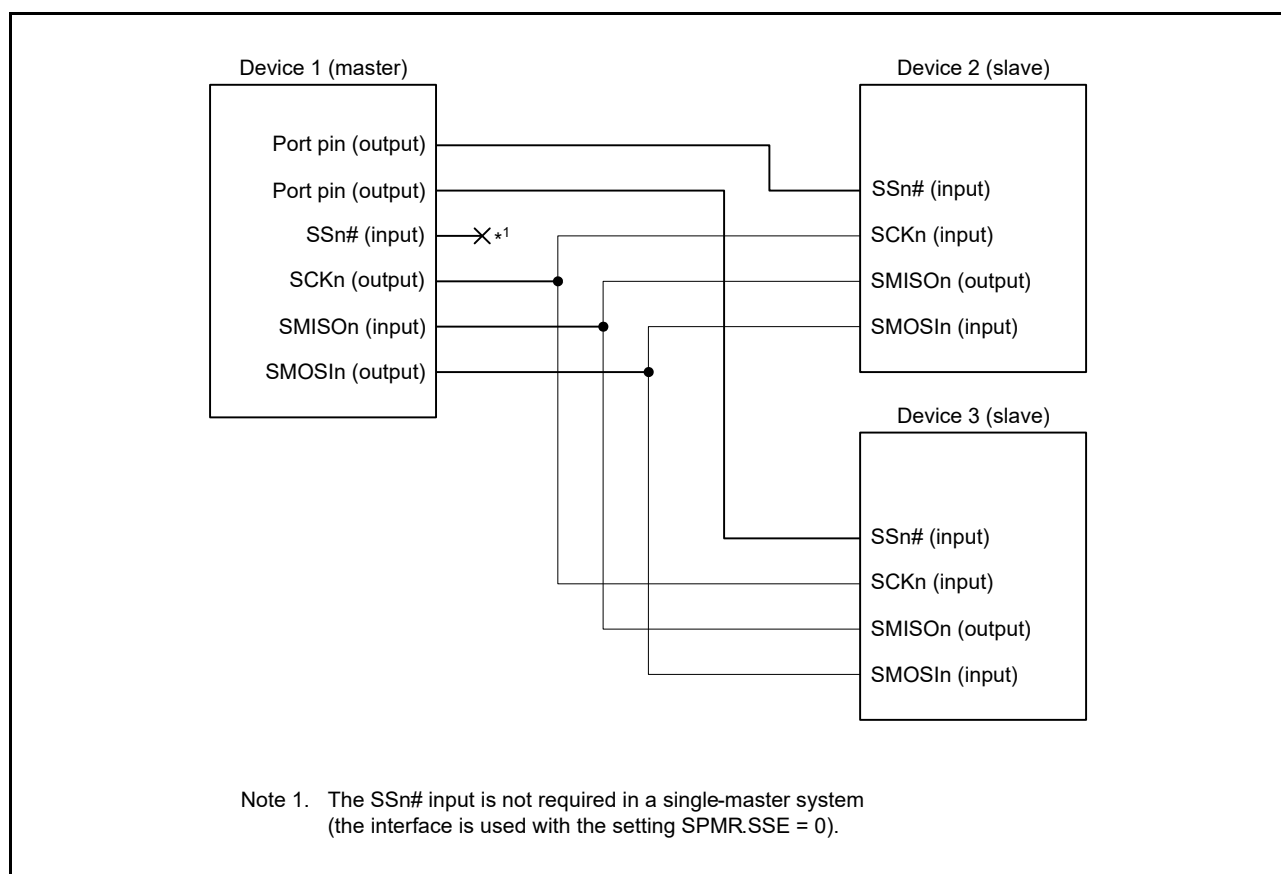


Figure 26.57 Example of Connections via a Simple SPI Mode (In Single Master Mode, SPMR.SSE Bit = 0)

26.8.1 States of Pins in Master and Slave Modes

The direction (input or output) of pins for the simple SPI mode interface differs according to whether the device is a master (SCR.CKE[1:0] = 00b or 01b and SPMR.MSS = 0) or slave (SCR.CKE[1:0] = 10b or 11b and SPMR.MSS = 1).

Table 26.29 lists the states of pins according to the mode and the level on the SSn# pin.

Table 26.29 States of Pins by Mode and Input Level on the SSn# Pin

Mode	Input on SSn# Pin	State of SMOSIn Pin	State of SMISOn Pin	State of SCKn Pin
Master mode*1	High level (transfer can proceed)	Output for data transmission*2	Input for received data	Clock output*3
	Low level (transfer cannot proceed)	High-impedance	Input for received data (but disabled)	High-impedance
Slave mode	High level (transfer cannot proceed)	Input for received data (but disabled)	High-impedance	Clock input (but disabled)
	Low level (transfer can proceed)	Input for received data	Output for data transmission	Clock input

Note 1. When there is only a single master (SPMR.SSE = 0), transfer is possible regardless of the input level on the SSn# pin (this is equivalent to input of a high level on the SSn# pin). Since the SSn# pin function is not required, the pin is available for other purposes.

Note 2. The SMOSIn pin output is in the high-impedance state when serial transmission is disabled (SCR.TE bit = 0).

Note 3. The SCKn pin output is in the high-impedance state when serial transmission is disabled (SCR.TE and RE bits = 00b) in a multi-master configuration (SPMR.SSE = 1).

26.8.2 SS Function in Master Mode

Setting the SCR.CKE[1:0] bits to 00b and the SPMR.MSS bit to 0 selects master operation. The SSn# pin is not used in single-master configurations (SPMR.SSE = 0), so transmission or reception can proceed regardless of the value of the SSn# pin.

When the level on the SSn# pin is high in a multi-master configuration (SPMR.SSE = 1), a master device outputs clock signals from the SCKn pin before starting transmission or reception to indicate that there are no other masters or another master is performing reception or transmission. When the level on the SSn# pin is low in a multi-master configuration (SPMR.SSE = 1), there are other masters, and this indicates that transmission or reception is in progress. At this time the SMOSIn output and SCKn pins will be placed in the high-impedance state and starting transmission or reception will not be possible. Furthermore, the value of the SPMR.MFF bit will be 1, indicating a mode fault error. In a multi-master configuration, start error processing by reading SPMR.MFF flag. Also, even if a mode fault error occurs while transmission or reception is in progress, transmission or reception will not be stopped, but the SMOSIn and SCKn pin output will be placed in the high-impedance state after the completion of the transfer.

Control a general port pin to produce the SS output signal from the master.

26.8.3 SS Function in Slave Mode

Setting the SCR.CKE[1:0] bits to 10b and the SPMR.MSS bit to 1 selects slave operation. When the level on the SSn# pin is high, the SMISOn output pin will be in the high-impedance state and clock input through the SCKn pin will be ignored. When the level on the SSn# pin is low, clock input through the SCKn pin will be effective and transmission or reception can proceed.

If the input on the SSn# pin changes from low to high level during transmission or reception, the SMISOn output pin will be placed in the high-impedance state. Meanwhile, the internal processing for transmission or reception will continue at the rate of the clock input through the SCKn pin until processing for the character currently being transmitted or received is completed, after which it stops. At that time, an interrupt (the appropriate one from among TXI, RXI, and TEI) will be generated.

26.8.4 Relationship between Clock and Transmit/Receive Data

The CKPOL and CKPH bits in the SPMR register can be used to set up the clock for use in transmission and reception in four different ways. The relation between the clock signal and the transmission and reception of data is shown in Figure 26.58. The relation is the same for both master and slave operation.

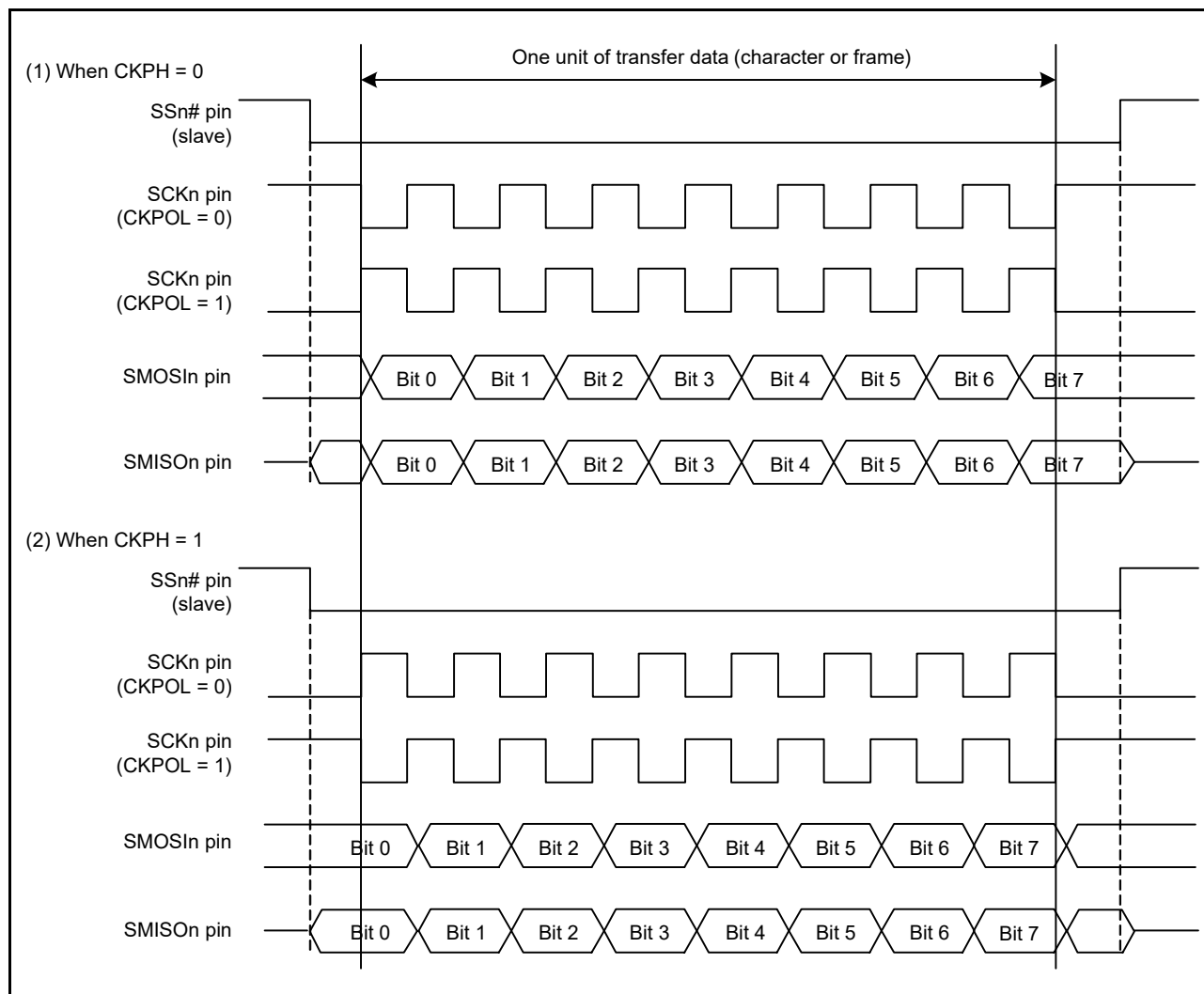


Figure 26.58 Relation between Clock Signal and Transmit/Receive Data in Simple SPI Mode

26.8.5 SCI Initialization (Simple SPI Mode)

The procedure is the same as for initialization in clock synchronous mode Figure 26.24, Sample SCI Initialization Flowchart. The CKPOL and CKPH bits in the SPMR register must be set to ensure that the kind of clock signal they select is suitable for both master and slave devices.

For initialization, changes to the operating mode, changes to the transfer format, and so on, initialize the SCR register before proceeding with changes.

As well as setting the RE bit to 0, note that the SSR. ORER, FER, and PER flags, as well as the RDR register, are not initialized.

26.8.6 Transmission and Reception of Serial Data (Simple SPI Mode)

In master operation, ensure that the SSn# pin of the slave device on the other side of the transfer is at the low level before starting the transfer and at the high level on completion of the transfer. Otherwise, the procedures are the same as in clock synchronous mode.

26.9 Bit Rate Modulation Function

The bit rate modulation function corrects the bit rate by thinning out the specified amount of clocks from those input to the baud rate generator.

When the SEMR.BRME bit is 1, the baud rate generator validates and counts the average interval of the number of clocks set in the MDDR register out of the total 256 clocks input.

Figure 26.59 assumes the SCI is in asynchronous mode, bits SMR.CKS[1:0] are 00b, the BRR register is 00h, and the MDDR register is 160. In this example, the cycle of the base clock is evenly corrected to 256/160, and the bit rate is corrected to 160/256. Note that there is an imbalance in thinning out the internal clock, and expansion and contraction occur in the pulse width of the base clock.

Note: Do not use this function in the highest speed settings (SMR.CKS[1:0] = 00b, SCR.CKE[1] = 0, and BRR = 0) in clock synchronous mode and simple SPI mode.

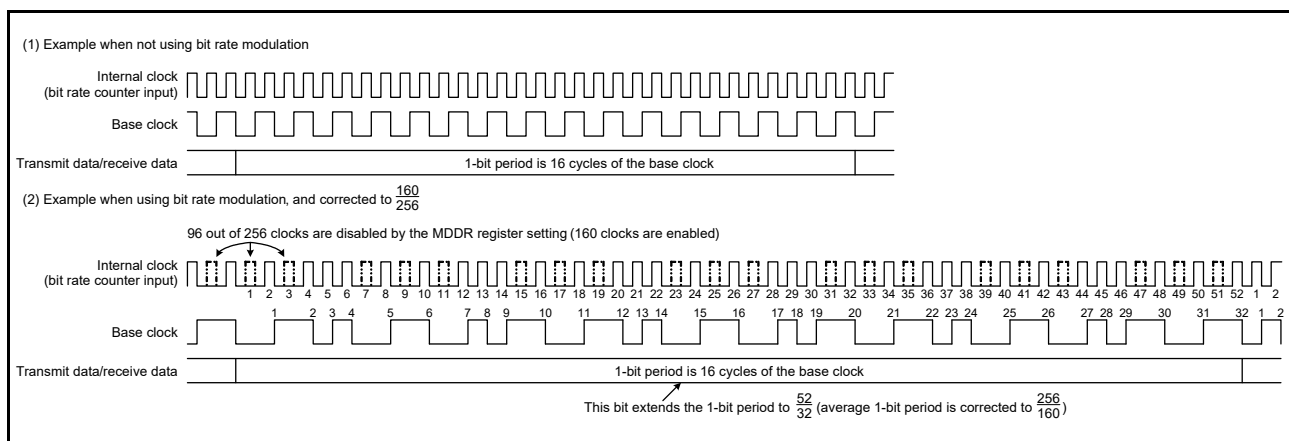


Figure 26.59 Example of the Base Clock When the Bit Rate Modulation Function is Used

The input of a clock signal with a shorter period to the baud rate generator reduces difference in the generated base clock period and, since the division ratio of the baud rate generator also becomes larger, reduces difference in the length of the 1-bit period.

26.10 Extended Serial Mode Control Section: Description of Operation

26.10.1 Serial Transfer Protocol

The extended serial mode control section of the SCI12 can realize the serial transfer protocol composed of Start Frames and Information Frames that is shown in Figure 26.60.

A Start Frame is composed of a Break Field, Control Field 0, and Control Field 1. An Information Frame is composed of a number of Data Fields, a CRC16 Upper Field, and a CRC16 Lower Field.

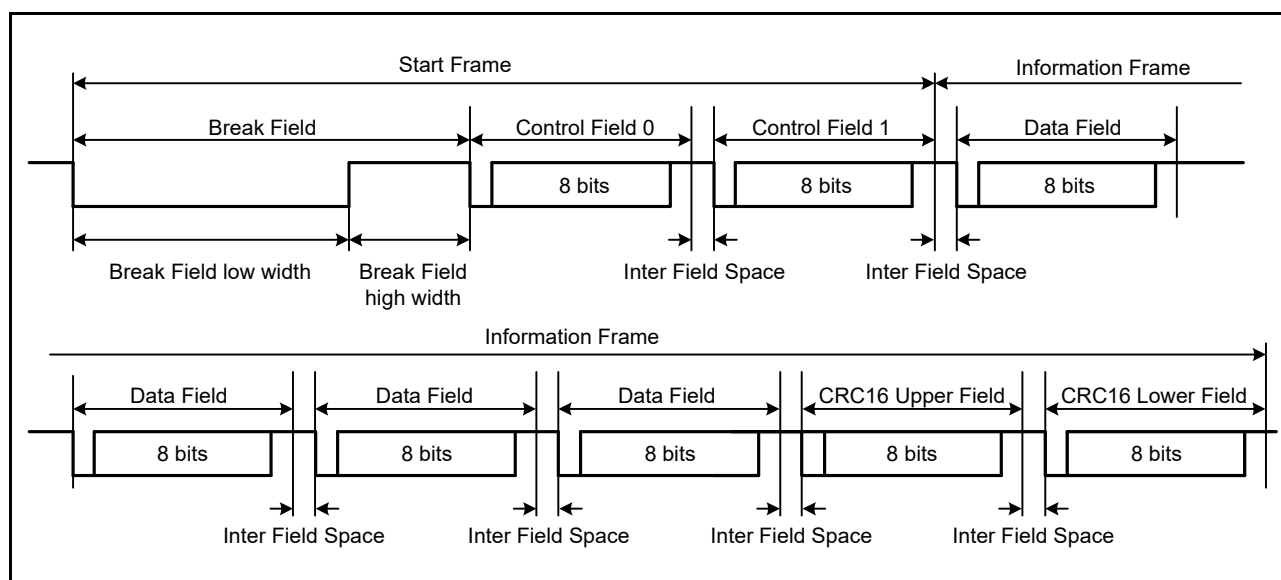


Figure 26.60 Protocol for Serial Transfer by the Extended Serial Mode Control Section

26.10.2 Transmitting a Start Frame

Figure 26.61 shows an example of operations to transmit a Start Frame, which is composed of the Break Field low width, Control Field 0, and Control Field 1. Figure 26.62 and Figure 26.63 are flowcharts for the transmission of a Start Frame.

Operations when the extended serial mode control section is to be used to transmit a Start Frame are as listed below. Be sure to use the SCI12 in asynchronous mode.

- (1) With Break Field low width output mode as the operating mode for the timer, writing 1 to the TCR.TCST bit starts counting by the timer, and the low level will be output from the TXDX12 pin over the period corresponding to registers TCNT and TPRE settings.
- (2) The output on the TXDX12 pin is inverted when the timer counter underflows, and the STR.BFDF flag is set to 1. An SCIX0 interrupt is also generated if the value of the ICR.BFDIE bit is 1.
- (3) Write 0 to the TCR.TCST bit to stop counting by the timer, and send the data for Control Field 0. After the Break Field low width output, stop counting before the next underflow occurs.
- (4) When the data for Control Field 0 have been transmitted, the data for Control Field 1 is transmitted.
- (5) When the data for Control Field 1 have been transmitted, an Information Frame is transmitted.

Omit the Break Field and Control Field 0 to suit the structure of the Start Frame.

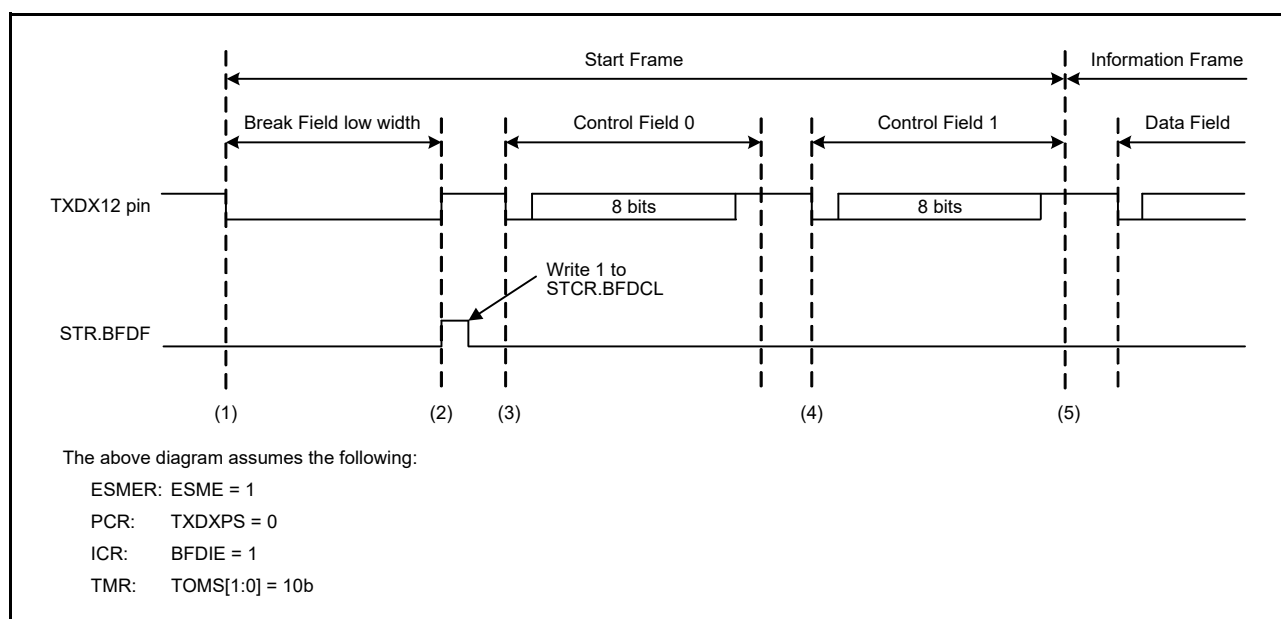


Figure 26.61 Example of Operations When Transmitting a Start Frame

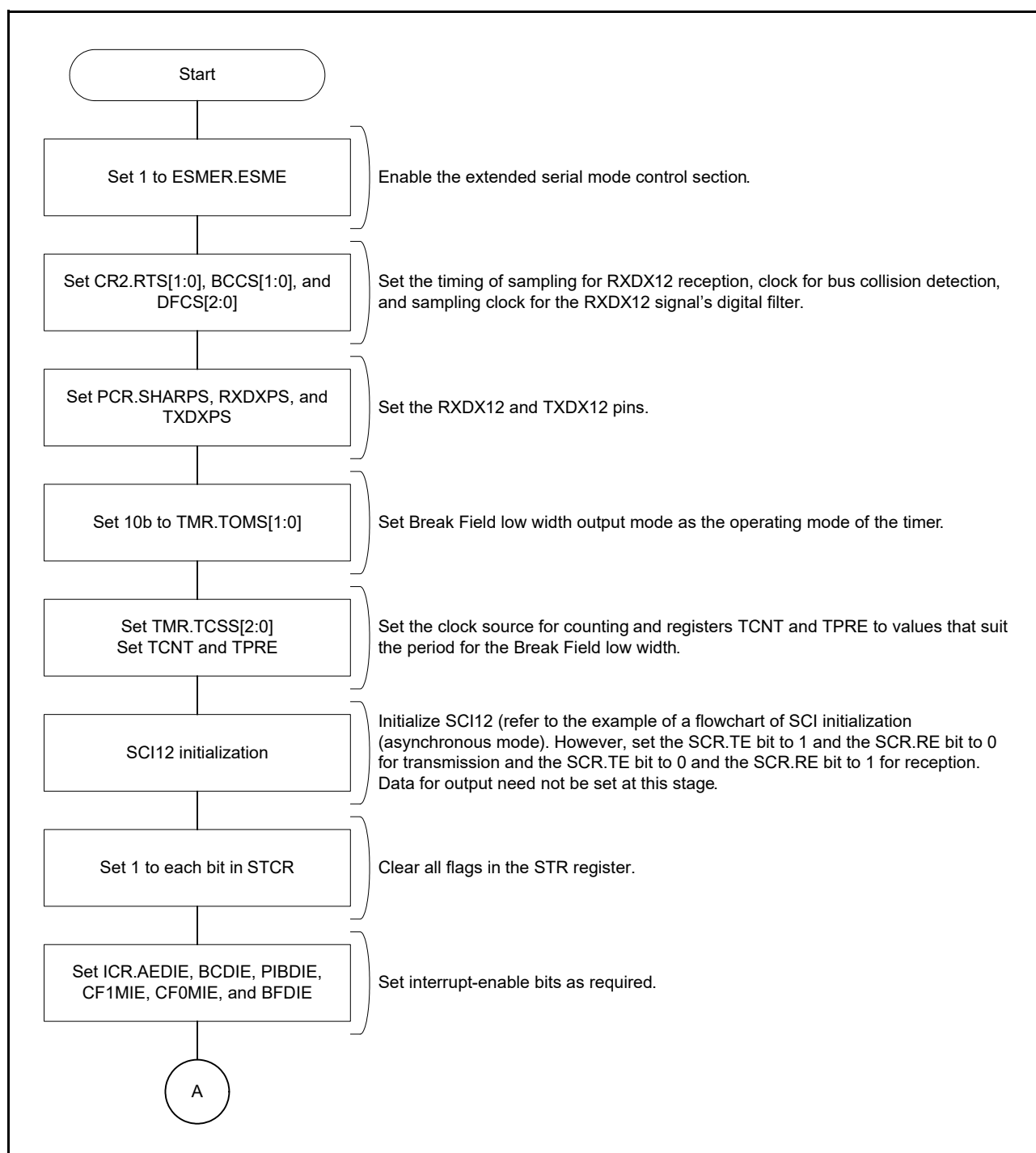


Figure 26.62 Example of Start Frame Transmission (1/2)

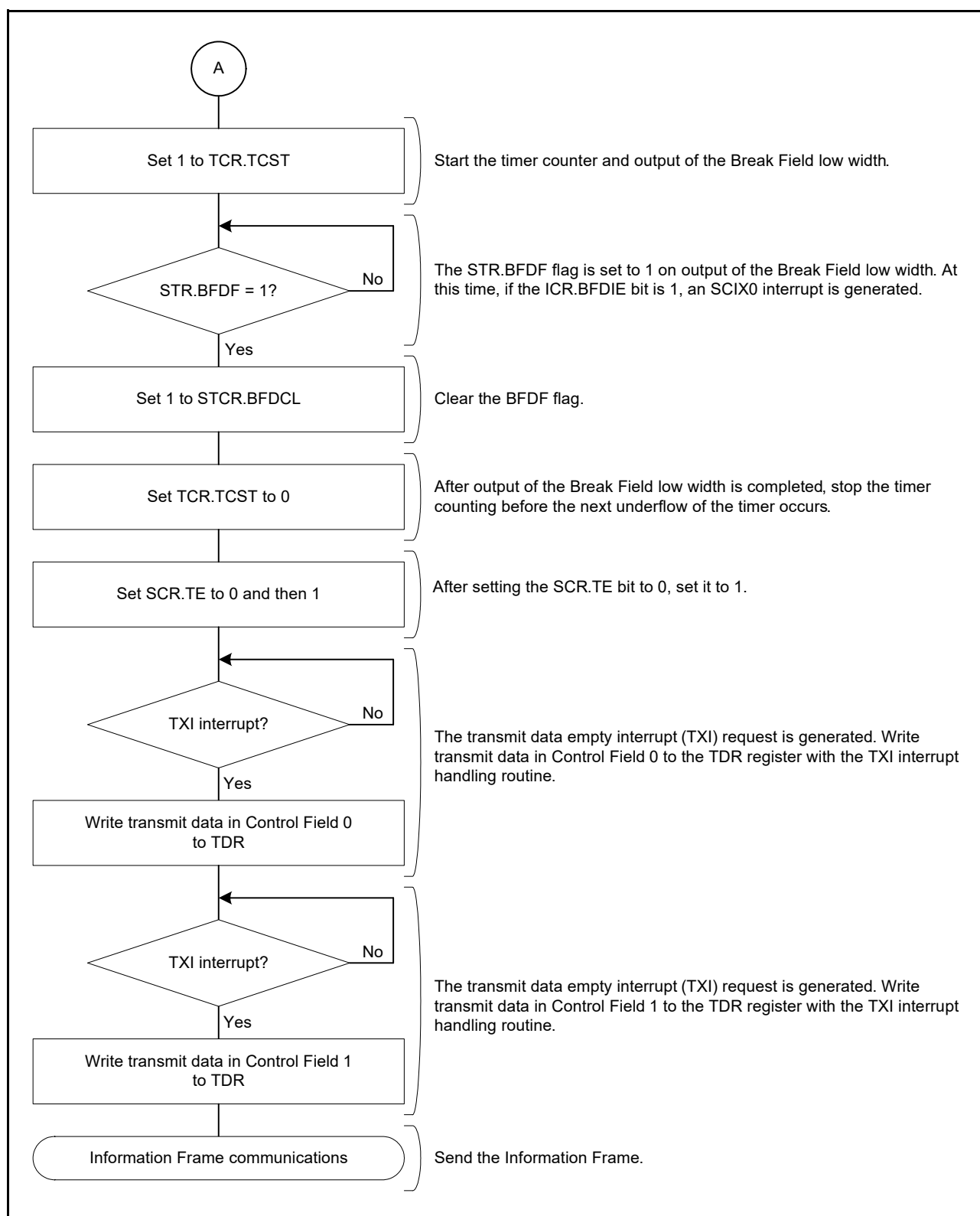


Figure 26.63 Example of Start Frame Transmission (2/2)

26.10.3 Receiving a Start Frame

The extended serial mode control section is capable of receiving Start Frames with the structures listed in Table 26.30.

Table 26.30 Structures of Start Frames

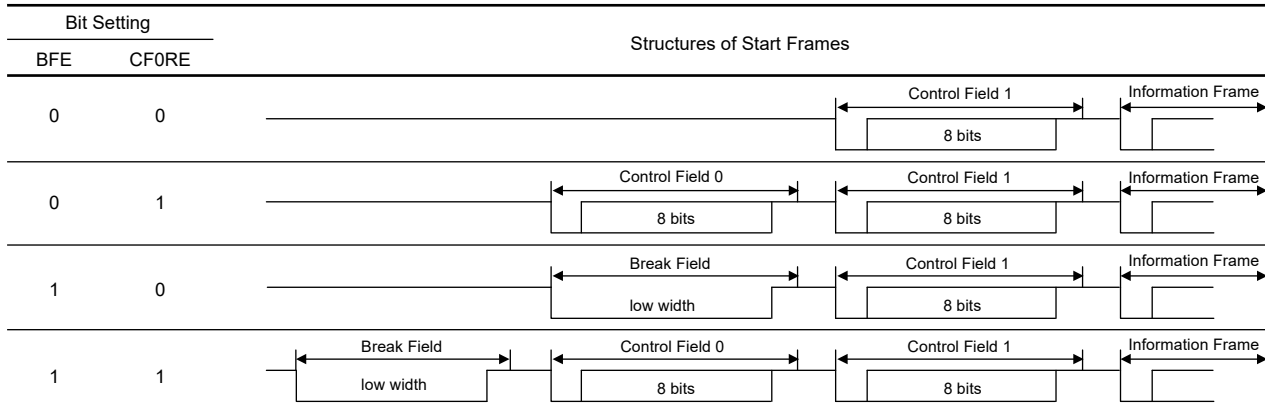


Figure 26.64 shows an example of operations to receive a Start Frame, which is composed of the Break Field low width, Control Field 0, and Control Field 1. Figure 26.65 and Figure 26.66 are flowcharts for the reception of a Start Frame, and Figure 26.67 is a state transition diagram when receiving a Start Frame.

Operations when the extended serial mode control section is to be used to receive a Start Frame are as listed below. Be sure to use the SCI12 in asynchronous mode.

- (1) With Break Field low width detection mode as the operating mode for the timer, writing 1 to the CR3.SDST bit enables detection of the Break Field low width.
- (2) Low-level input on the RXDX12 pin continuing over a period longer than that corresponding to the settings of registers TCNT and TPRE is detected as the Break Field low width. At this time, the STR.BFDF flag is set to 1. An SCIX0 interrupt is also generated if the value of the ICR.BFDIE bit is 1.
- (3) When the input from the RXDX12 pin goes high after the Break Field low width, the CR0.RXDSF flag becomes 0 and reception of Control Field 0 starts.
- (4) If the data received in Control Field 0 match the data set in the CF0DR register, the STR.CF0MF flag is set to 1. An SCIX1 interrupt is also generated if the value of the ICR.CF0MIE bit is 1. Reception of Control Field 1 starts after that. If the data received in Control Field 0 do not match the data set in the CF0DR register, a transition to the state prior to Break Field low width detection proceeds.
- (5) If the data received in Control Field 1 match the data set in registers PCF1DR and SCF1DR, the STR.CF1MF flag is set to 1. An SCIX1 interrupt is also generated if the value of the ICR.CF1MIE bit is 1. Transfer of the Information Frame starts after that. If the data received in Control Field 1 do not match the data set in either or both of registers PCF1DR and SCF1DR, a transition to the state prior to Break Field low width detection proceeds.

Omit the Break Field and Control Field 0 to suit the structure of the Start Frame.

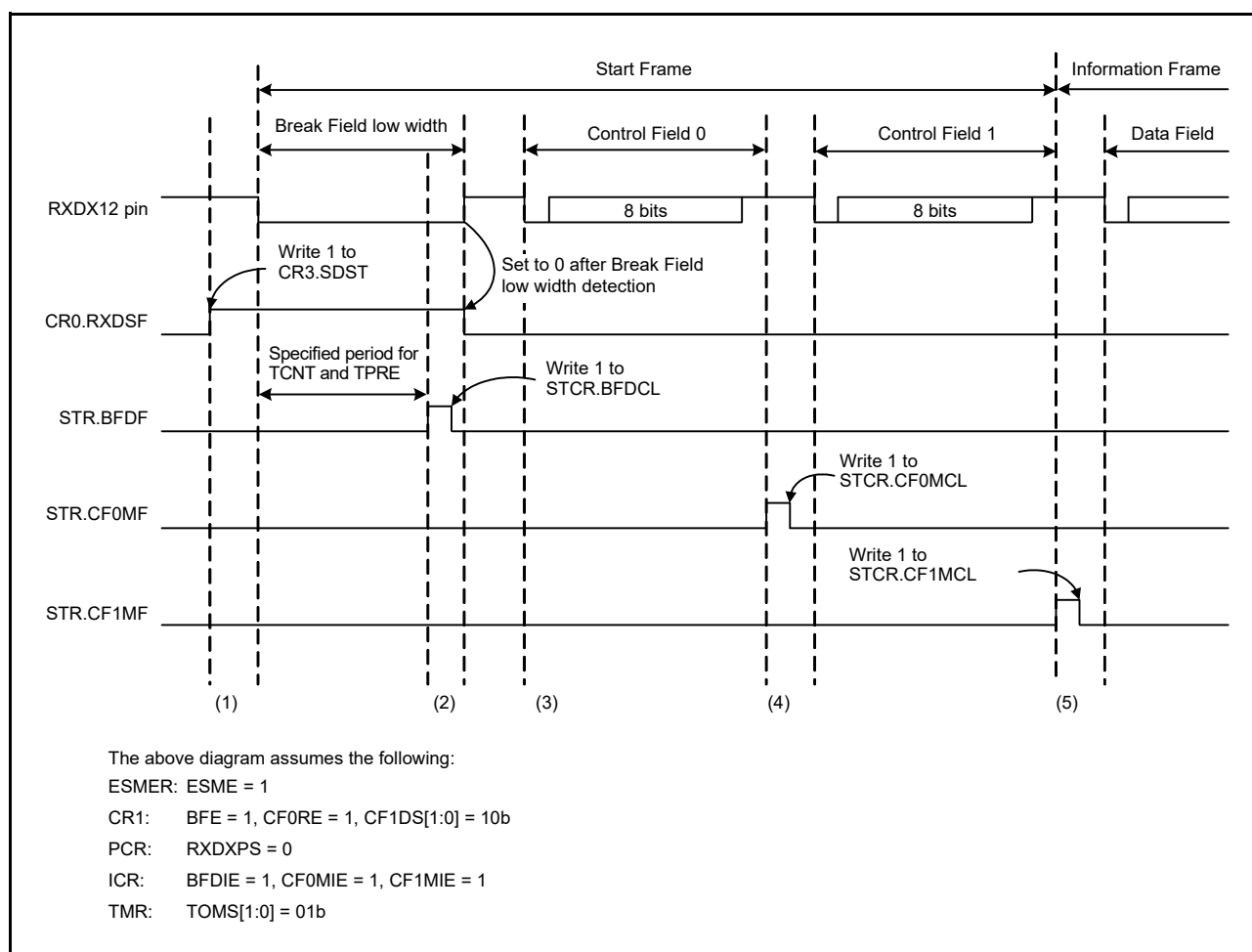


Figure 26.64 Example of Operations at the Time of Start Frame Reception

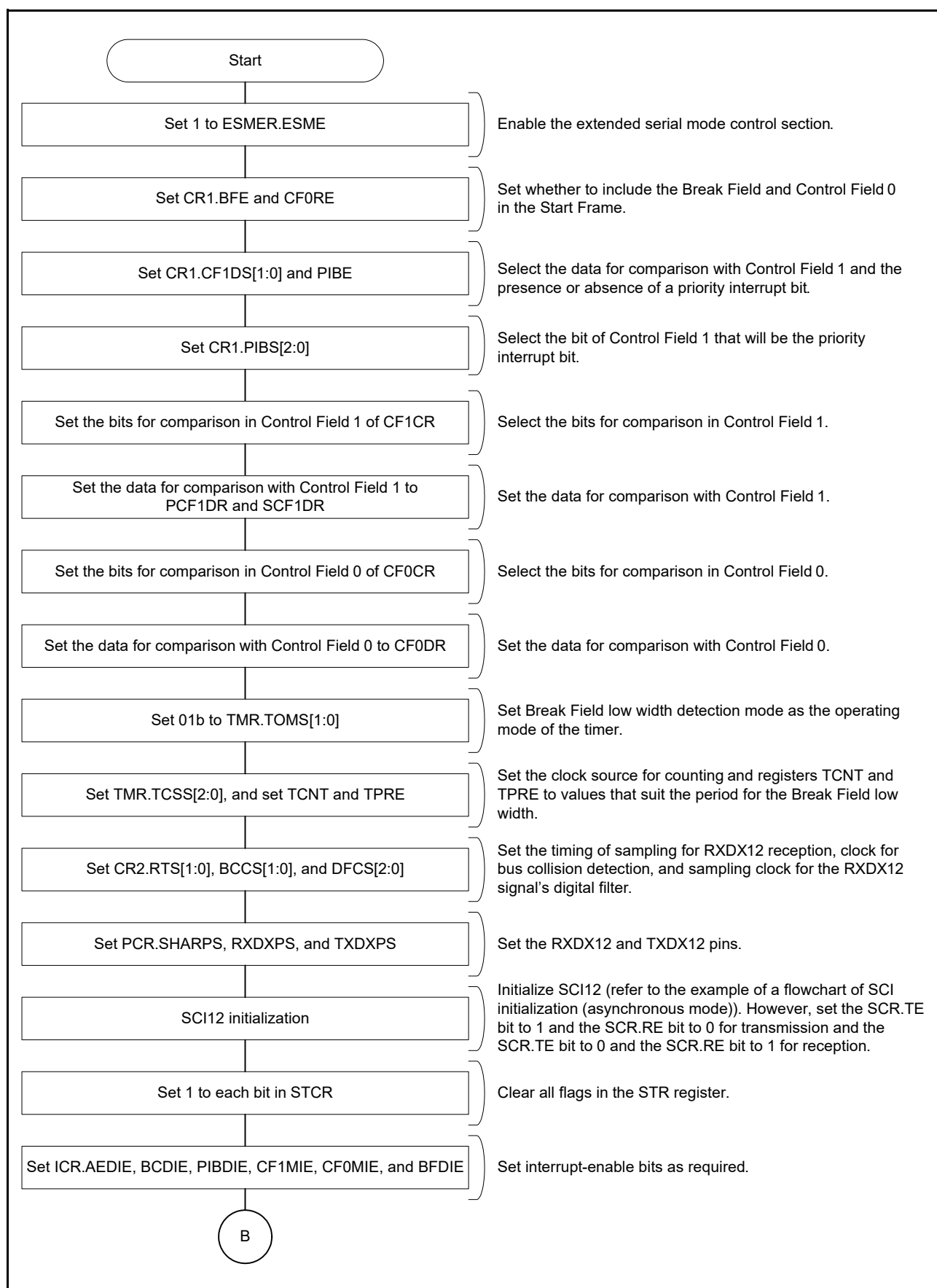


Figure 26.65 Sample Flowchart for Reception of a Start Frame (1)

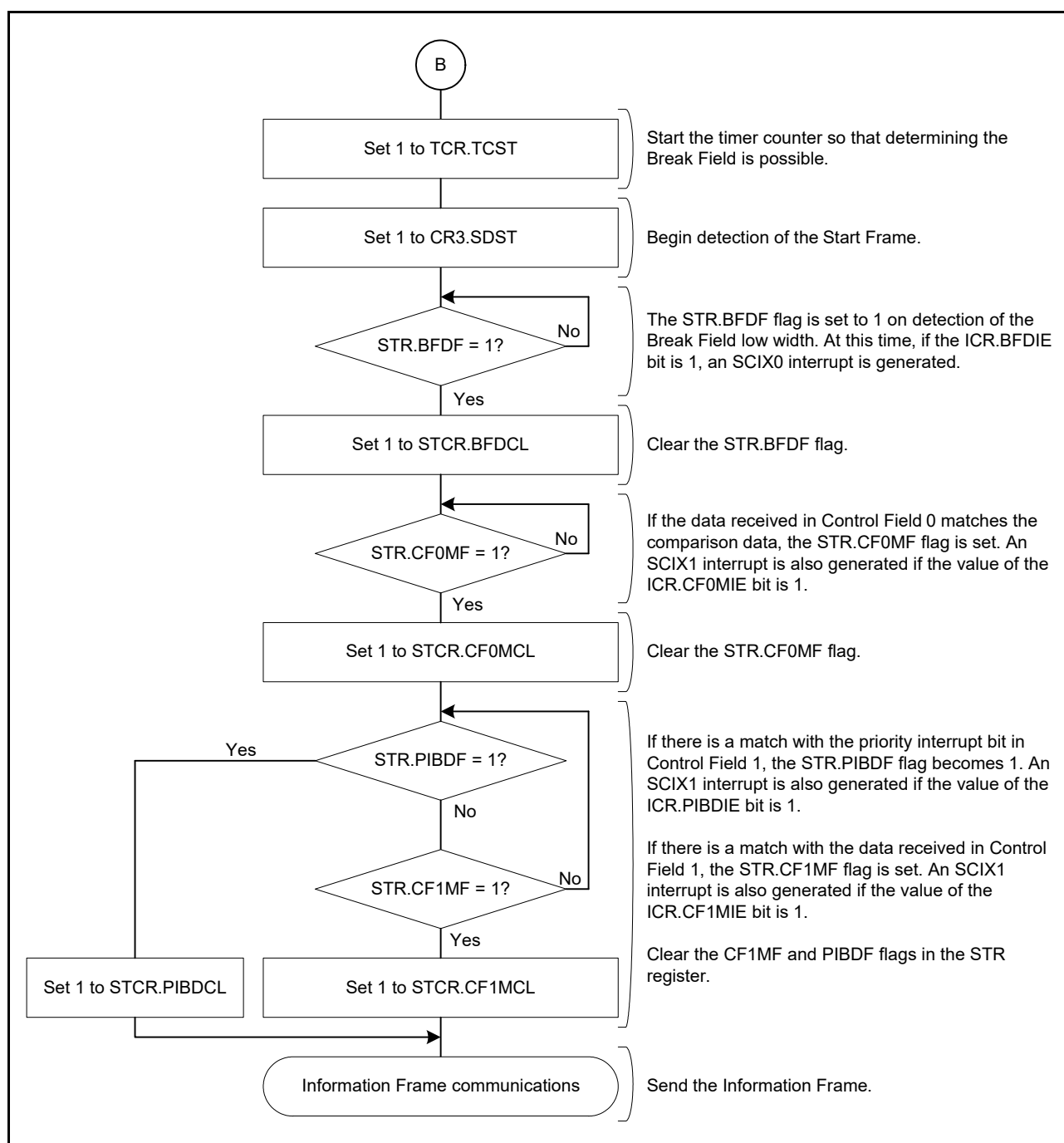
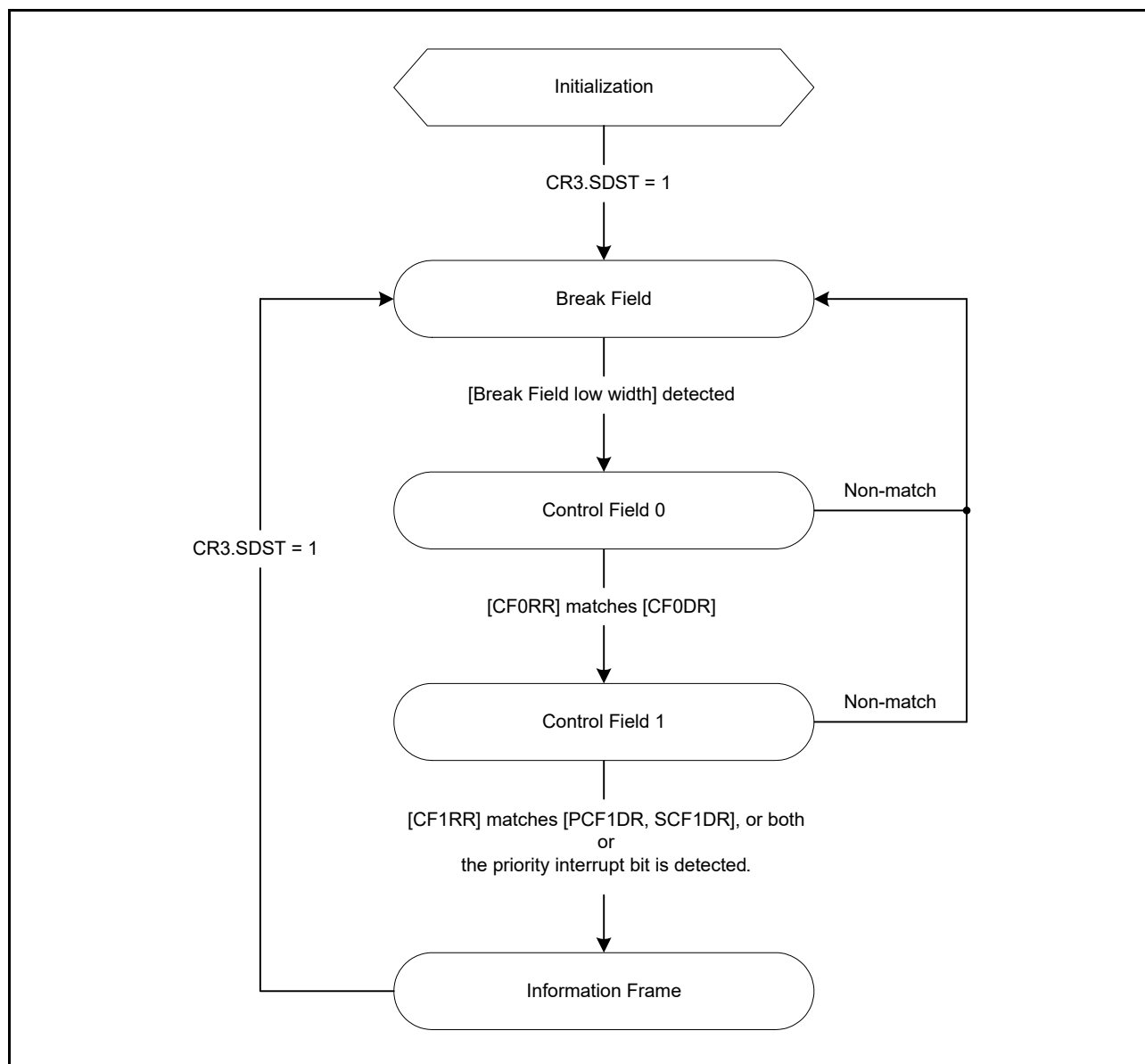


Figure 26.66 Sample Flowchart for Reception of a Start Frame (2)

**Figure 26.67** State Transitions When Receiving a Start Frame

26.10.3.1 Priority Interrupt Bit

Figure 26.68 shows an example of operation in Start Frame reception where a priority interrupt bit is in use. Setting the CR1.PIBE bit to 1 enables the use of a priority interrupt bit.

Operations of the extended serial mode control section in start Frame reception where a priority interrupt bit is in use are as described below.

Steps (1) to (4) are the same as in Figure 26.64, for Start Frame reception.

- (5) If the value of the bit selected by the CR1.PIBS[2:0] bits matches the corresponding bit in the PCF1DR register, the STR.PIBDF flag is set to 1. An SCIX1 interrupt is also generated if the value of the ICR.PIBDIE bit is 1. Transfer of the Information Frame starts after that. If the data received in Control Field 1 do not match the data set in either or both of registers PCF1DR and SCF1DR and the priority interrupt bit is not detected, a transition to the state prior to Break Field low width detection proceeds.

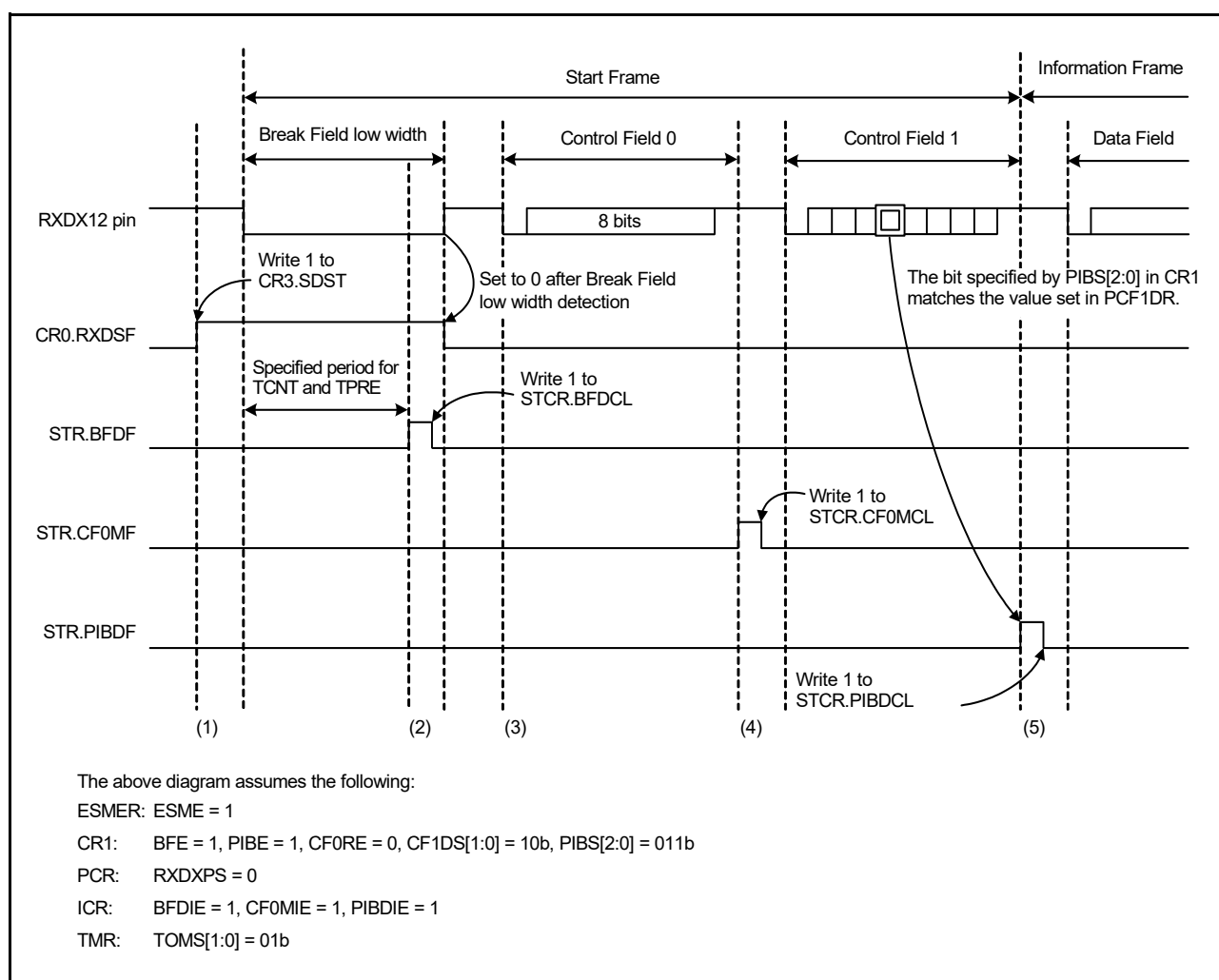


Figure 26.68 Example of Operations When Receiving a Start Frame While the CR1.PIBE Bit is 1

26.10.4 Detection of Bus Collisions

Detection of bus collisions operate for cases where output of the Break Field low width and transmission of data are in progress when the ESMER.ESME bit and the SCR.TE bit are set to 1.

Figure 26.69 shows an example of operations with bus collision detection. Signals output through TXDX12 and input through RXDX12 are sampled with the bus collision detection clock set with the CR2.BCCS[1:0] bits as the sampling clock, and the STR.BCDF flag is set to 1 if the signals fail to match three times in a row. An SCIX2 interrupt is also generated if the value of the ICR.BCDIE bit is 1.

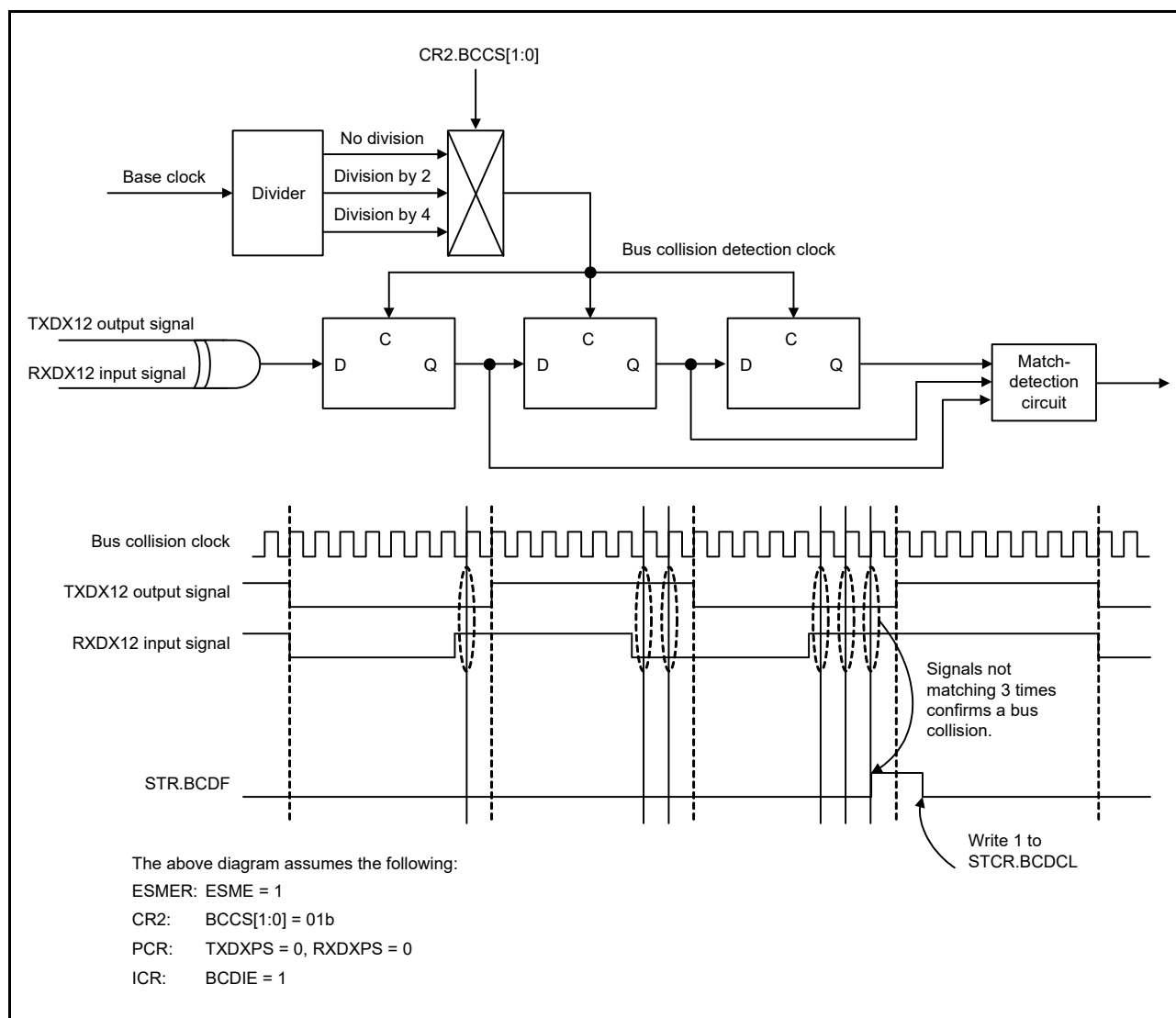


Figure 26.69 Example of Operations with Bus Collision Detection

26.10.5 Digital Filter for Input on the RXDX12 Pin

Signals input through the RXDX12 pin can be passed through a digital filter before they are conveyed to the internal circuits. The digital filter consists of three flip-flop circuit stages connected in series and a match-detecting circuit. The CR2.DFCS[2:0] bits select the sampling clock for the RXDX12 pin input signals. If the outputs of all three latches match, the given level is conveyed to subsequent circuits. If the levels do not match, the previous value is retained. In other words, levels are confirmed as being the signal if they are retained for at least three cycles of the sampling clock but judged to be noise rather than changes in the signal level if they change within three cycles of the sampling clock. Figure 26.70 shows an example of operations with the digital filter.

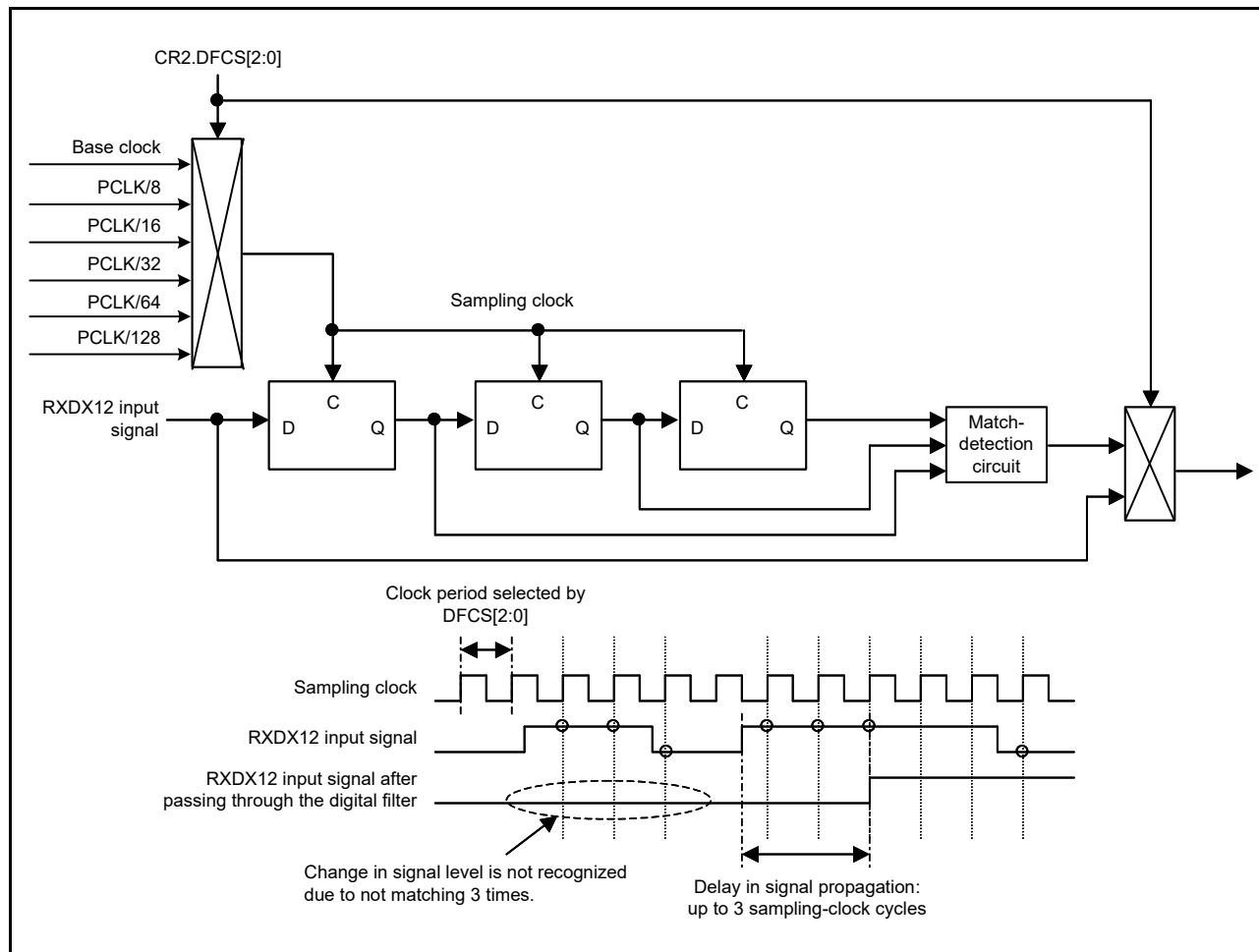


Figure 26.70 Example of Operations with the Digital Filter

26.10.6 Bit Rate Measurement

The bit rate measurement function measures the intervals between rising and falling edges and between falling and rising edges of the signal input from the RXDX12 pin. Figure 26.71 shows an example of operations for bit rate measurement.

- (1) Writing 1 to the CR0.BRME bit enables bit rate measurement. Only set the BRME bit to 1 when you wish to proceed with bit rate measurement. Furthermore, bit rate measurement will not proceed during a Break Field, even if the BRME bit is set to 1.
- (2) After detection of the Break Field low width, bit rate measurement starts when the level input on the RXDX12 pin becomes high.
- (3) Once bit rate measurement has started, counter values from the timer are retained in the read buffers on the input of valid edges from the RXDX12 pin (rising and falling edges) and the counter is reloaded. An SCIX3 interrupt is also generated if the value of the ICR.AEDIE bit is 1. Retention by registers TCNT and TPRES is released by reading these registers.
- (4) The bit rate as calculated from the values counted during intervals between valid edges can be used for adjusting the rate by changing the settings of the BRR register. To disable the bit rate measurement after a match with Control Field 1, write 0 to the CR0.BRME bit.

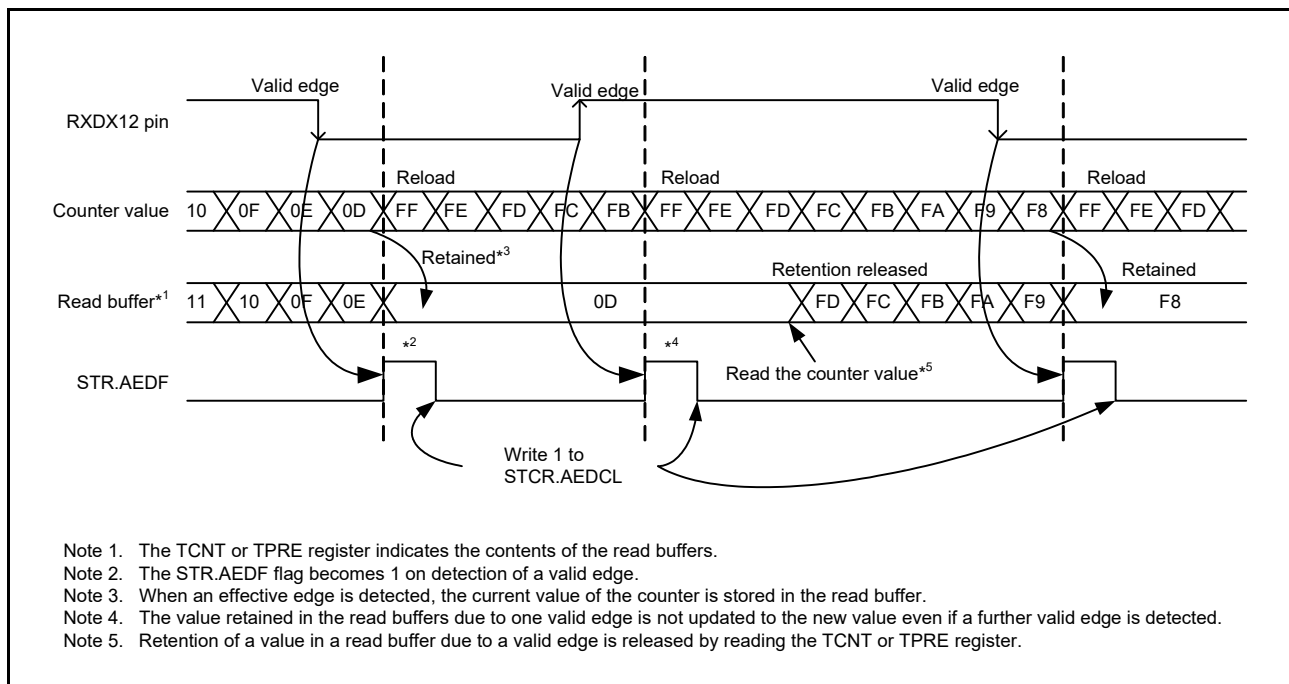


Figure 26.71 Example of Operations for Bit Rate Measurement

26.10.7 Selectable Timing for Sampling Data Received through RXDX12

The extended serial mode control section provides a way of adjusting the timing for the sampling of data received through the RXDX12 pin by setting the CR2.RTS[1:0] bits to select the rising edges of 8th, 10th, 12th, or 14th cycle of the base clock. If the value of the SEMR.ABCS bit is 1, the bits select the rising edges of 4th, 5th, 6th, or 7th cycle of the base clock. Figure 26.72 shows timing for the sampling of data received through RXDX12.

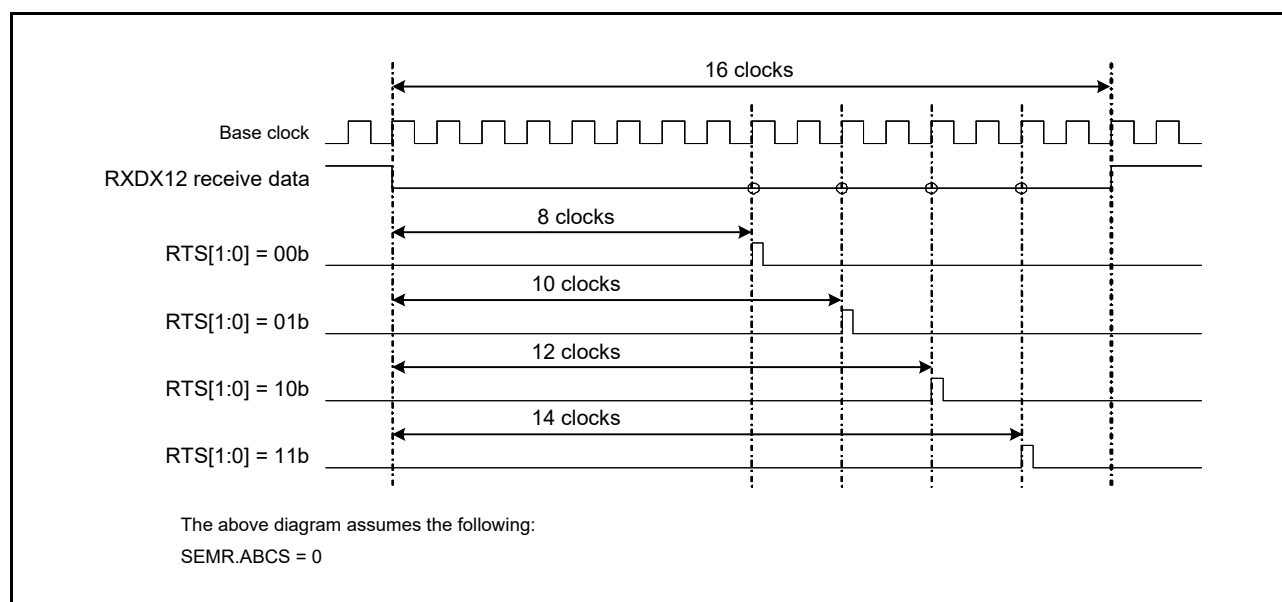


Figure 26.72 Timing for Sampling of Data Received through RXDX12

26.10.8 Timer

The timer has the following operating modes.

(1) Break Field Low Width Output Mode

This mode is for output through the TXDX12 pin of the low level over the Break Field low width at the transmission of a Start Frame. Setting the TMR.TOMS[1:0] bits to 10b switches operation to Break Field low width output mode. The TMR.TCSS[2:0] bits select the clock source for the counter. When the TCR.TCST bit is set to 1, the output on the TXDX12 pin goes to the low level and counting starts. When the timer underflows, the output on the TXDX12 pin goes to the high level and the STR.BFDF flag is set to 1. An SCIX0 interrupt is also generated if the value of the ICR.BFDIE bit is 1. When 0 is written to the TCR.TCST bit, counting stops after reloading of registers TPRES and TCNT. After output of the Break Field low width is completed, stop the timer before it underflows again. Figure 26.73 shows an example of operations in Break Field low width output mode.

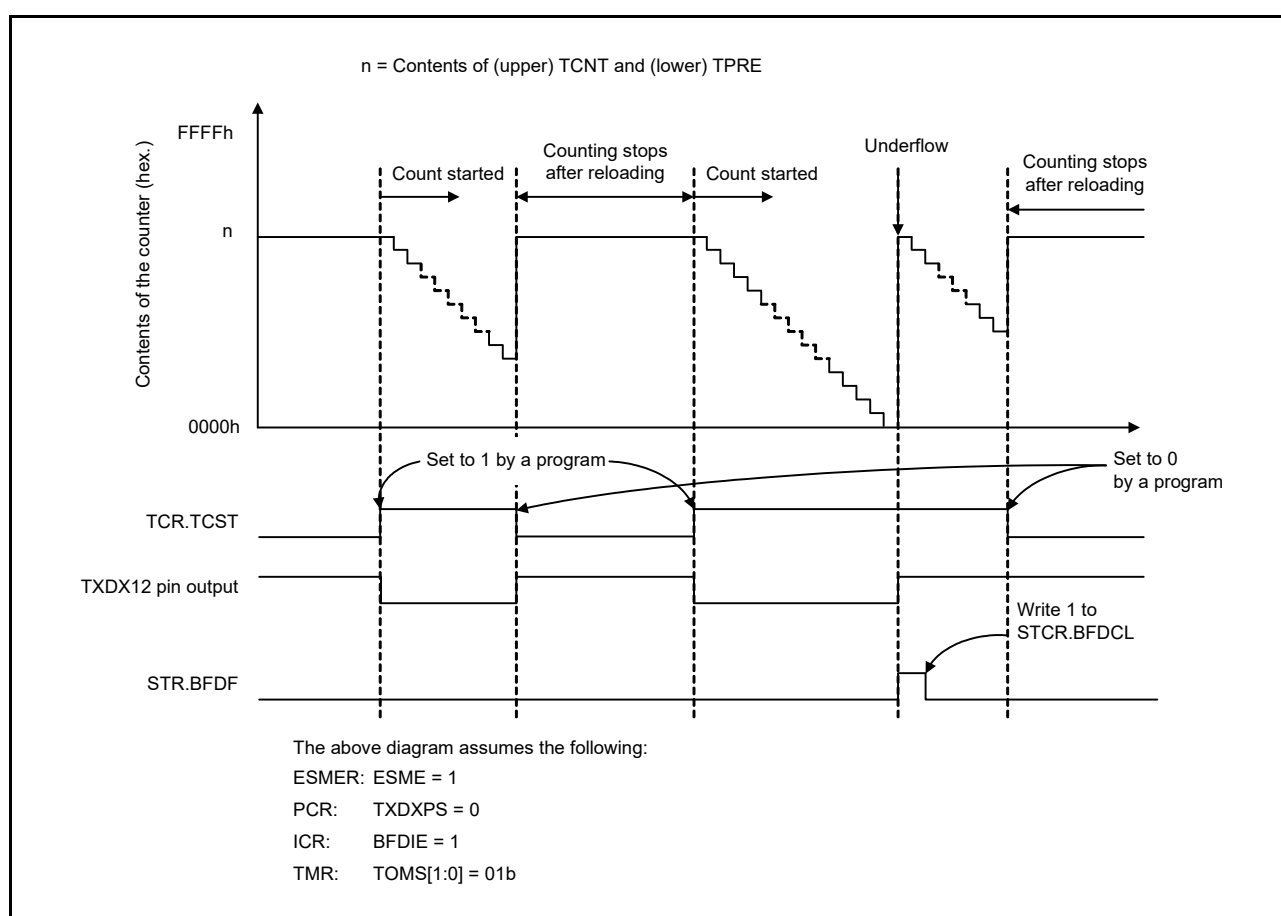


Figure 26.73 Example of Operations in Break Field Low Width Output Mode

(2) Break Field Low Width Determination Mode

This mode is for determining the Break Field low width in the input signal on the RXDX12 pin at the reception of a Start Frame. Setting the TMR.TOMS[1:0] bits to 01b switches operation to Break Field low width determination mode. The TMR.TCSS[2:0] bits select the clock source for the counter. When the TCR.TCST bit is set to 1, the interface enters the Break Field low width determinable state. Determination starts when a low level is input from the RXDX12 pin. When a high level is then input on the RXDX12 pin, registers TPRES and TCNT are reloaded and the interface enters the Break Field low width determinable state. When the timer underflows during Break Field low width determination, the STR.BFDF flag is set to 1. An SCIX0 interrupt is also generated if the value of the ICR.BFDIE bit is 1. If an underflow of the timer during data transfer cause a problem in the form of interrupt generation, stop the timer after Break Field low width determination. Figure 26.74 shows an example of operations in Break Field low width output mode.

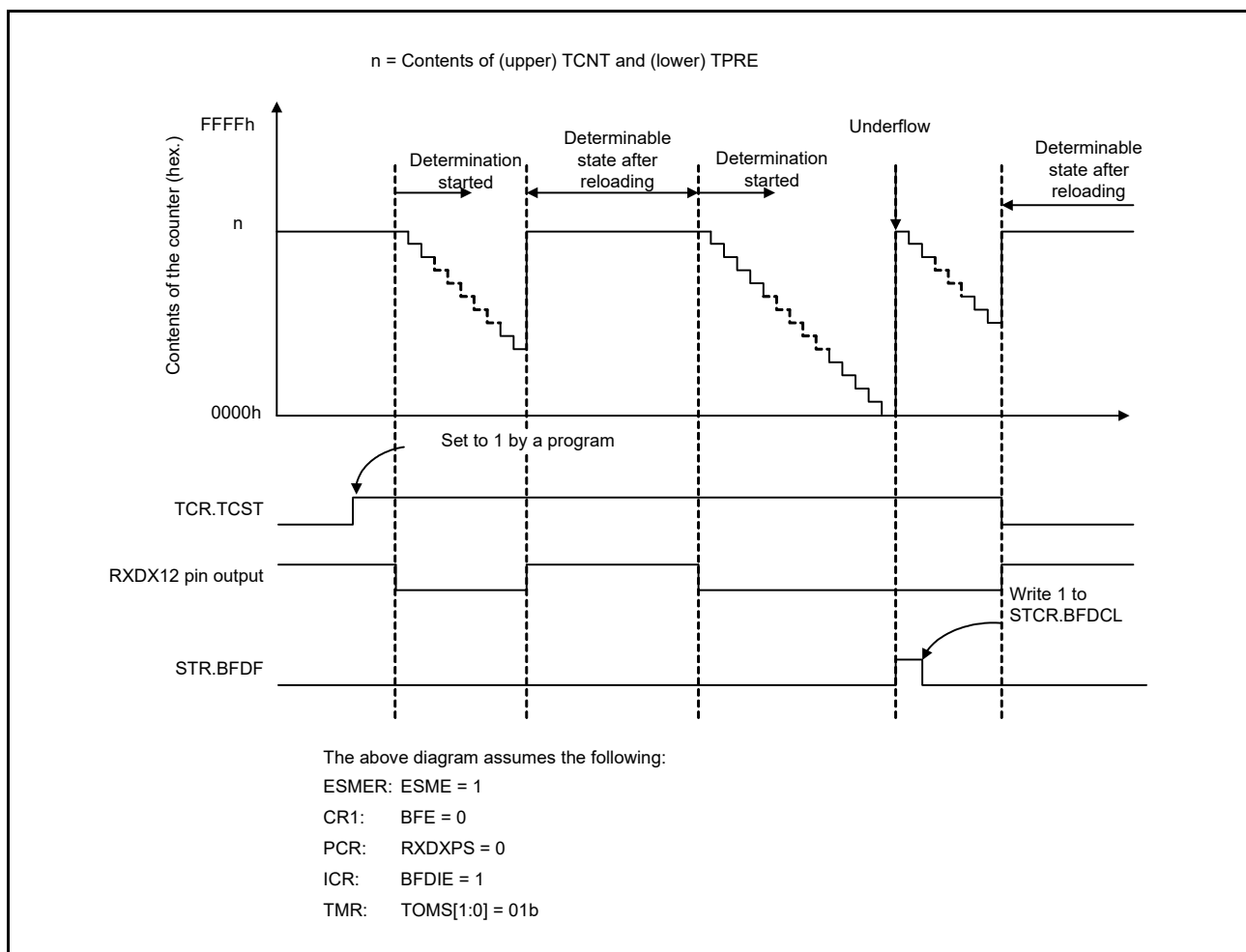


Figure 26.74 Example of Operations in Break Field Low Width Determination Mode

(3) Timer Mode

This mode is for counting cycles of the internal clock as the clock source. Setting the TMR.TOMS[1:0] bits to 00b switches operation to timer mode. The TMR.TCSS[2:0] bits select the clock source for the counter. Counting starts when 1 is written to the TCR.TCST bit and stops when 0 is written to the TCST bit. Registers TPRES and TCNT both count down. The TPRES register counts cycles of the clock source for counting, and underflows of the TPRES register provide the clock source for counting by the TCNT register. When the timer underflows, the STR.BFDF flag is set to 1. An SCIX0 interrupt is also generated if the value of the ICR.BFDIE bit is 1.

26.11 Noise Cancellation Function

Figure 26.75 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of two stages of flip-flop circuits and a match-detection circuit. When the level on the pin matches in three consecutive samples taken at the set sampling interval, the matching level continues to be conveyed internally until the level on the pin again matches in three consecutive samples.

In asynchronous mode, the noise cancellation function can be applied on the RXDn input signal. The period of the base clock (1/16th of a bit-period when SEMR.ABCS = 0 and 1/8th of a bit-period when SEMR.ABCS = 1) is the sampling interval.

In simple I²C mode, the noise cancellation function can be applied on the SSDAn and SSCLn input signals. The sampling clock is the clock signal produced by frequency-dividing the signal from the clock source for the internal baud-rate generator by one, two, four, or eight as selected by the setting of the SNFR.NFCS[2:0] bits.

If the base clock is stopped with the noise filter enabled and then the clock input is started again, the noise filter operation resumes from where the clock was stopped. If SCR.TE and SCR.RE are set to 0 during base clock input, all of the noise filter flip-flop values are initialized to 1. Accordingly, if the input data is 1 when reception operation resumes, it is determined that a level match is detected and is conveyed to the internal signal. When the level being input corresponds to 0, the initial output of the noise filter is retained until the level matches in three consecutive samples.

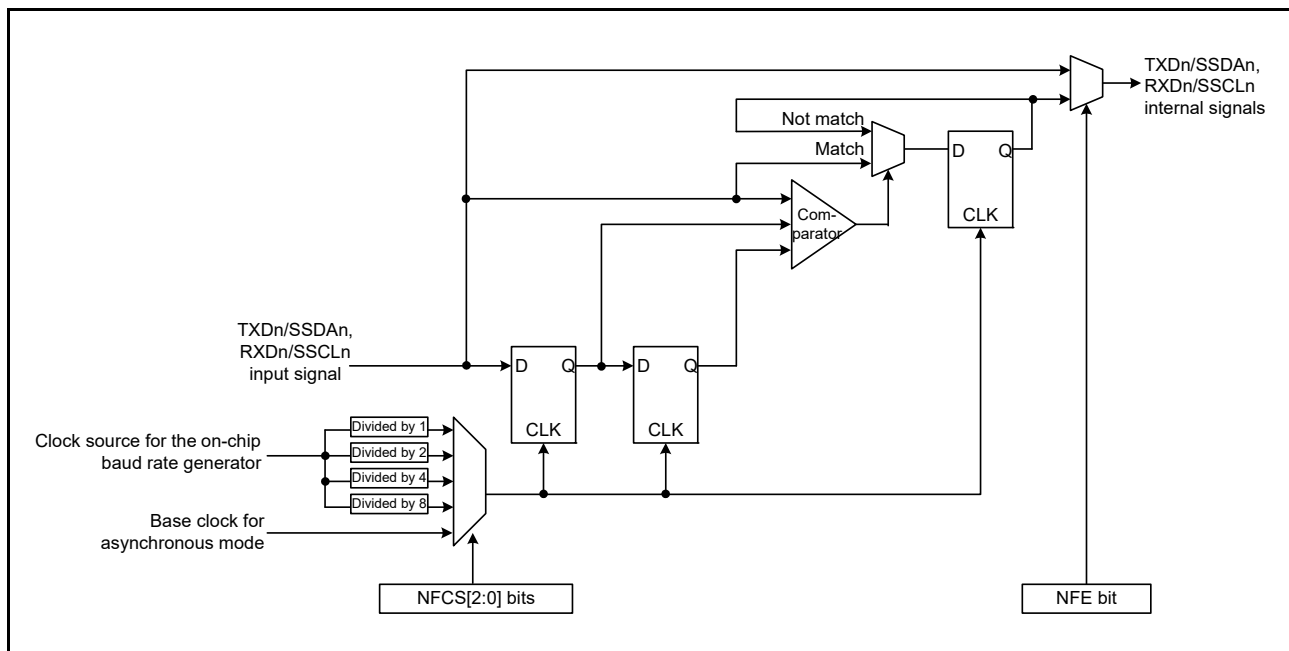


Figure 26.75 Block Diagram of Digital Noise Filter

26.12 Interrupt Sources

26.12.1 Buffer Operations for TXI and RXI Interrupts

If the conditions for a TXI and RXI interrupt are satisfied while the interrupt status flag in the interrupt controller is 1, the SCI does not output the interrupt request but retains it internally (with a capacity for retention of one request per source). When the value of the interrupt status flag in the interrupt controller becomes 0, the interrupt request retained within the SCI is output. The internally retained interrupt request is automatically discarded once the actual interrupt is output. Clearing of the corresponding interrupt enable bit (the TIE or RIE bit in the SCR register) can also be used to discard an internally retained interrupt request.

26.12.2 Interrupts in Asynchronous Mode, Clock Synchronous Mode, and Simple SPI Mode

Table 26.31 lists interrupt sources in asynchronous mode, clock synchronous mode, and simple SPI mode. Individual interrupt sources can be enabled or disabled with the enable bits in the SCR register.

If the SCR.TIE bit is 1, a TXI interrupt request is generated when transmit data is transferred from the TDR or TDRL register*¹ to the TSR register. A TXI interrupt request can also be generated by setting the SCR.TE bit to 1 after setting the SCR.TIE bit to 1 or by using a single instruction to set the SCR.TE and SCR.TIE bit to 1 at the same time. A TXI interrupt request can activate the DTC to handle data transfer.

A TXI interrupt request is not generated by setting the SCR.TE bit to 1 while the setting of the SCR.TIE bit is 0 or by setting the SCR.TIE bit to 1 while the setting of the SCR.TE bit is 1.*²

Note that setting the SCR.TE bit to 0 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt.

When new data is not written by the time of transmission of the last bit of the current transmit data and the setting of the SCR.TEIE bit is 1, the SSR.TEND flag becomes 1 and a TEI interrupt request is generated. Furthermore, when the setting of the SCR.TE bit is 1, the SSR.TEND flag retains the value 1 until further transmit data are written to the TDR or TDRL register*¹, and setting the SCR.TEIE bit to 1 leads to the generation of a TEI interrupt request.

Writing data to the TDR or TDRL register*¹ leads to clearing of the SSR.TEND flag and, after a certain time, discarding of the TEI interrupt request.

If the SCR.RIE bit is 1, an RXI interrupt request is generated when received data is stored in the RDR register. An RXI interrupt request can activate the DTC to handle data transfer.

Setting of any from among the ORER, FER, and PER flags in the SSR register to 1 while the SCR.RIE bit is 1 leads to the generation of an ERI interrupt request. An RXI interrupt request is not generated at this time. Clearing all three flags (ORER, FER, and PER) leads to discarding of the ERI interrupt request.

Note 1. In the case where asynchronous mode and 9-bit data length are selected

Note 2. To temporarily disable TXI interrupts at the time of transmission of the last of the data and so on when you wish a new round of transmission to start after handling of the transmission-completed interrupt, control disabling and enabling of the interrupt by using the interrupt request enable bit in the interrupt controller rather than using the SCR.TIE bit. This can prevent the suppression of TXI interrupt requests in the transfer of new data.

Table 26.31 Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DTC Activation
ERI	Receive error	ORER, FER, or PER	Not possible
RXI	Receive data full	RDRF	Possible
TXI	Transmit data empty	TDRE	Possible
TEI	Transmit end	TEND	Not possible

26.12.3 Interrupts in Smart Card Interface Mode

Table 26.32 lists interrupt sources in smart card interface mode. A transmit end interrupt (TEI) request cannot be used in this mode.

Table 26.32 SCI Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DTC Activation
ERI	Receive error or error signal detection	ORER, PER, or ERS	Not possible
RXI	Receive data full	—	Possible
TXI	Transmit data empty	TEND	Possible

Data transmission/reception using the DTC is also possible in smart card interface mode, similar to in the normal SCI mode. In transmission, when the SSR.TEND flag is set to 1, a TXI interrupt request is generated. This TXI interrupt request activates the DTC allowing transfer of transmit data if the TXI request is specified beforehand as a source of DTC activation. The TEND flag is automatically set to 0 when the DTC transfers the data.

If an error occurs, the SCI automatically retransmits the same data. During the retransmission, the TEND flag is kept to 0 and the DTC is not activated. Therefore, the SCI and DTC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, the SSR.ERS flag is not automatically cleared to 0 at error occurrence. Therefore, the ERS flag must be cleared by previously setting the SCR.RIE bit to 1 to enable an ERI interrupt request to be generated at error occurrence.

When transmitting/receiving data using the DTC, be sure to make settings to enable the DTC before making SCI settings. For DTC settings, refer to **section 16, Data Transfer Controller (DTCb)**.

In reception, an RXI interrupt request is generated when receive data is set to the RDR register. This RXI interrupt request activates the DTC allowing transfer of receive data if the RXI request is specified beforehand as a source of DTC activation. If an error occurs, the error flag is set. Therefore, the DTC is not activated and an ERI interrupt request is issued to the CPU instead; the error flag must be cleared.

26.12.4 Interrupts in Simple I²C Mode

The interrupt sources in simple I²C mode are listed in Table 26.33. The STI interrupt is allocated to the transmit end interrupt (TEI) request. The receive error interrupt (ERI) request cannot be used.

The DTC can also be used to handle transfer in simple I²C mode.

When the value of the SIMR2.IICINTM bit is 1, an RXI request will be generated on the falling edge of the SSCLn signal for the eighth bit. If the RXI has been set up as an activating request for the DTC beforehand, the RXI request will activate the DTC to handle transfer of the received data. Furthermore, a TXI request is generated on the falling edge of the SSCLn signal for the ninth bit (acknowledge bit). If the TXI has been set up as an activating request for the DTC beforehand, the TXI request will activate the DTC to handle transfer of the transmit data.

When the value of the SIMR2.IICINTM bit is 0, an RXI request (ACK detection) if the input on the SSDAn pin is at the low level or a TXI request (NACK detection) if the input on the SSDAn pin is at the high level will be generated on the rising edge of the SSCLn signal for the ninth bit (acknowledge bit). If the RXI has been set up as an activating request for the DTC beforehand, the RXI request will activate the DTC to handle transfer of the received data.

Also, if the DTC is used for data transfer in reception or transmission, be sure to set up and enable the DTC before setting up the SCI.

When the IICSTAREQ, IICRSTAREQ, and IICSTPREQ bits in the SIMR3 register are used to generate a start condition, restart condition, or stop condition, the STI request is issued when generation is complete.

Table 26.33 SCI Interrupt Sources

Name	Interrupt Source		Interrupt Flag	DTC Activation
	IICINTM bit = 0	IICINTM bit = 1		
RXI	ACK detection	Reception	—	Possible
TXI	NACK detection	Transmission	—	Possible*1
STI	Completion of generation of a start, restart, or stop condition		IICSTIF	Not possible

Note 1. Activation of the DTC is only possible when the SIMR2.IICINTM bit is 1 (use reception and transmission interrupts).

26.12.5 Interrupt Requests from the Extended Serial Mode Control Section

The extended serial mode control section has a total of six types of interrupt request for generating the SCIX0 interrupt (Break Field low width detected), SCIX1 interrupt (Control Field 0 match, Control Field 1 match, priority interrupt bit detected), SCIX2 interrupt (bus collision detected), and SCIX3 interrupt (valid edge detected). When any of the interrupt factors is generated, the corresponding status flag is set to 1. Details of all of the interrupt requests are listed in Table 26.34.

Table 26.34 Interrupt Sources of the Extended Serial Mode Control Section

Interrupt Request	Status Flag	Interrupt Factors
SCIX0 interrupt (Break Field low width detected)	BFDF	• Detection of a Break Field low width longer than the interval corresponding to the timer setting • Completion of the output of a Break Field low width over the interval corresponding to the timer setting • Underflow of the timer
SCIX1 interrupt (Control Field 0 match)	CF0MF	The data received in Control Field 0 matching the value set in the CF0DR register
SCIX1 interrupt (Control Field 1 match)	CF1MF	The data received in Control Field 1 matching the value set in the PCF1DR or SCF1DR register
SCIX1 interrupt (priority interrupt bit detected)	PIBDF	The value of the bit specified as the priority interrupt bit matching the value set in the PCF1DR register
SCIX2 interrupt (bus collision detected)	BCDF	The output level on the TXDX12 pin and the input level on the RXDX12 pin not matching on three consecutive cycles of the bus collision detection clock
SCIX3 interrupt (valid edge detected)	AEDF	Detection of a valid edge during bit rate measurement

26.13 Usage Notes

26.13.1 Setting the Module Stop Function

Module stop control register B (MSTPCRB) is used to stop and start SCI operations. With the value after a reset, SCI operations are stopped. Register access is enabled by releasing the module stop state. For details, refer to [section 11](#), Low Power Consumption.

26.13.2 Break Detection and Processing

When a framing error is detected, a break can be detected by reading the RXDn pin value directly. In a break, the input from the RXDn pin becomes all 0s, and so the SSR.FER flag is set to 1 (framing error has occurred), and the SSR.PER flag may also be set to 1 (parity error has occurred). When the SEMR.RXDESEL bit is 0, the SCI continues the receive operation even after a break is received. Therefore, note that even if the FER flag is set to 0 (no framing error occurred), it will be set to 1 again. When the SEMR.RXDESEL bit is 1, the SCI sets the SSR.FER flag to 1 and stops receiving operation until a start bit of the next data frame is detected. If the SSR.FER flag is set to 0 at this time, the SSR.FER flag retains 0 during the break. When the RXDn pin becomes high and the break ends, detecting the beginning of the start bit at the first falling edge of the RXDn pin allows the SCI to start the receiving operation.

26.13.3 Mark State and Sending Breaks

When the SCR.TE bit is 0 (serial transmission is disabled), the TXDn pin becomes high-impedance. To forcibly set the TXDn pin to mark or space state while the TE bit is 0, set the I/O port associated registers and switch the TXDn pin to general output port.

For holding the communication line in the mark (“1”) state until the TE bit is set to 1 (serial transmission is enabled), set the corresponding bit in the PODR register to 1 for high output from general output port. To start communications, set the TE bit to 1 and then the corresponding bit in the PMR register to 1.

To send a break (the space state for longer than a certain period of time) while data transmission, set the corresponding bit in the PODR register to 0 (low output), and set the corresponding bit in the PMR register to 0 (general I/O port). Then set the TE bit to 0 if necessary. When the TE bit is set to 0, the transmitter is initialized regardless of the current transmit status.

26.13.4 Receive Error Flags and Transmit Operations (Clock Synchronous Mode and Simple SPI Mode)

Transmission cannot be started when a receive error flag (SSR.ORER) is set to 1, even if data is written to the TDR register. Be sure to set the receive error flags to 0 before starting transmission. Note also that the receive error flags cannot be set to 0 even if the SCR.RE bit is set to 0 (serial reception is disabled).

26.13.5 Writing Data to the TDR Register

Data can be written to registers TDR, TDRH, and TDRL. However, if new data is written to registers TDR, TDRH, and TDRL when transmit data is remaining in registers TDR, TDRH, and TDRL, the previous data in registers TDR, TDRH, and TDRL is lost because it has not been transferred to the TSR register yet. Be sure to write transmit data to registers TDR, TDRH, and TDRL in the TXI interrupt request handling routine.

26.13.6 Restrictions on Clock Synchronous Transmission (Clock Synchronous Mode and Simple SPI Mode)

When the external clock source is used as a synchronization clock, the following restrictions apply.

(1) Start of transmission

Update the TDR register by the CPU or DTC and wait for at least five PCLK cycles before allowing the transmit clock to be input (refer to Figure 26.76).

(2) Continuous transmission

- (a) Write the next transmit data to the TDR or TDRL register before the falling edge of the transmit clock (bit 7) (refer to Figure 26.76).
- (b) When updating the TDR register after bit 7 has started to transmit, update the TDR register while the synchronization clock is in the low-level period, and set the high-level width of the transmit clock (bit 7) to four PCLK cycles or longer (refer to Figure 26.76).

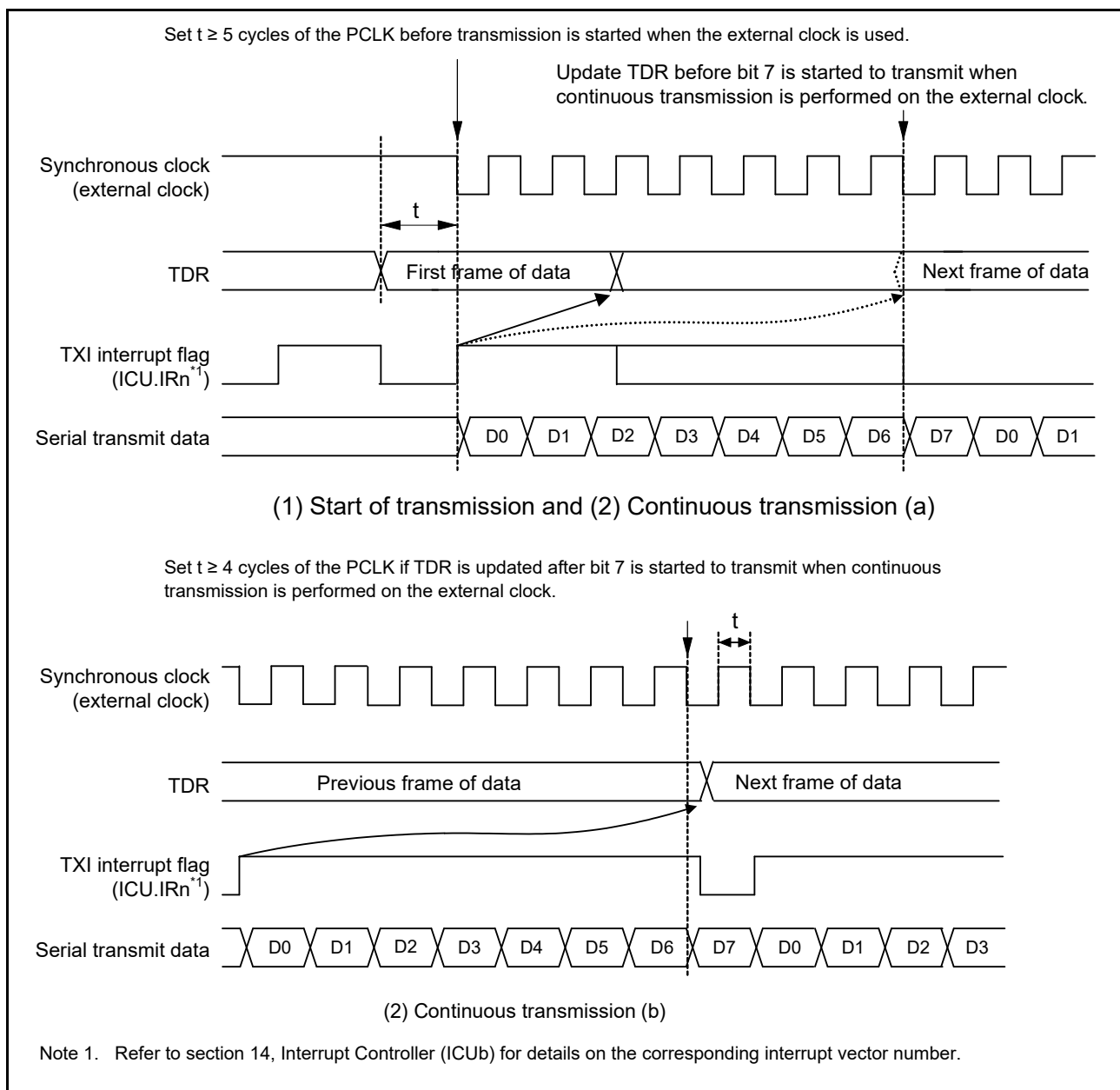


Figure 26.76 Restrictions on Use of External Clock in Clock Synchronous Transmission

26.13.7 Restrictions on Using DTC

When using the DTC to read the RDR, RDRH, and RDRL registers, be sure to set the receive data full interrupt (RXI) as the activation source of the relevant SCI.

26.13.8 Notes on Starting Transfer

At the point where transfer starts when the interrupt status flag (IRn.IR flag) in the interrupt controller is 1, follow the procedure below to clear interrupt requests before permitting operations (by setting the SCR.TE or SCR.RE bit to 1). For details on the interrupt status flag, refer to **section 14, Interrupt Controller (ICUb)**.

- Confirm that transfer has stopped (the setting of the SCR.TE or SCR.RE bits is 0).
- Set the corresponding interrupt enable bit (SCR.TIE or SCR.RIE) to 0.
- Read the corresponding interrupt enable bit (SCR.TIE or SCR.RIE bit) to check that it has become 0.
- Set the interrupt status flag (IRn.IR flag) in the interrupt controller to 0.

26.13.9 SCI Operations during Low Power Consumption State

(1) Transmission

When making settings for the module stopped state or in transitions to software standby, stop operations (by setting the TIE, TE, and TEIE bits in the SCR register to 0) after switching the TXDn pin to the general I/O port pin function. Setting the TE bit to 0 resets the TSR register and the SSR.TEND flag. Depending on the port settings, output pins may output the level before a transition to the low power consumption state is made after release from the module stopped state or software standby mode. When transitions to these states are made during transmission, the data being transmitted become indeterminate.

To transmit data in the same transmit mode after cancellation of the low power consumption state, set the TE bit to 1, read the SSR register, and write data to the TDR register sequentially to start data transmission. To transmit data with a different transmit mode, initialize the SCI first.

Figure 26.77 shows a sample flowchart for transition to software standby mode during transmission. Figure 26.78 and Figure 26.79 show the port pin states during transition to software standby mode.

Before specifying the module stop state or making a transition to software standby mode from the transmit mode using DTC transfer, stop the transmit operations (TE = 0). To start transmission after cancellation using the DTC, set the TE and TIE bits to 1. The TXI interrupt flag is set to 1 and transmission starts using the DTC.

(2) Reception

Before specifying the module stop state or making a transition to software standby mode, stop the receive operations (SCR.RE = 0). If transition is made during data reception, the data being received will be invalid.

To receive data in the same receive mode after cancellation of the low power consumption state, set the RE bit to 1, and then start reception. To receive data in a different receive mode, initialize the SCI first.

Figure 26.80 shows a sample flowchart for transition to software standby mode during reception.

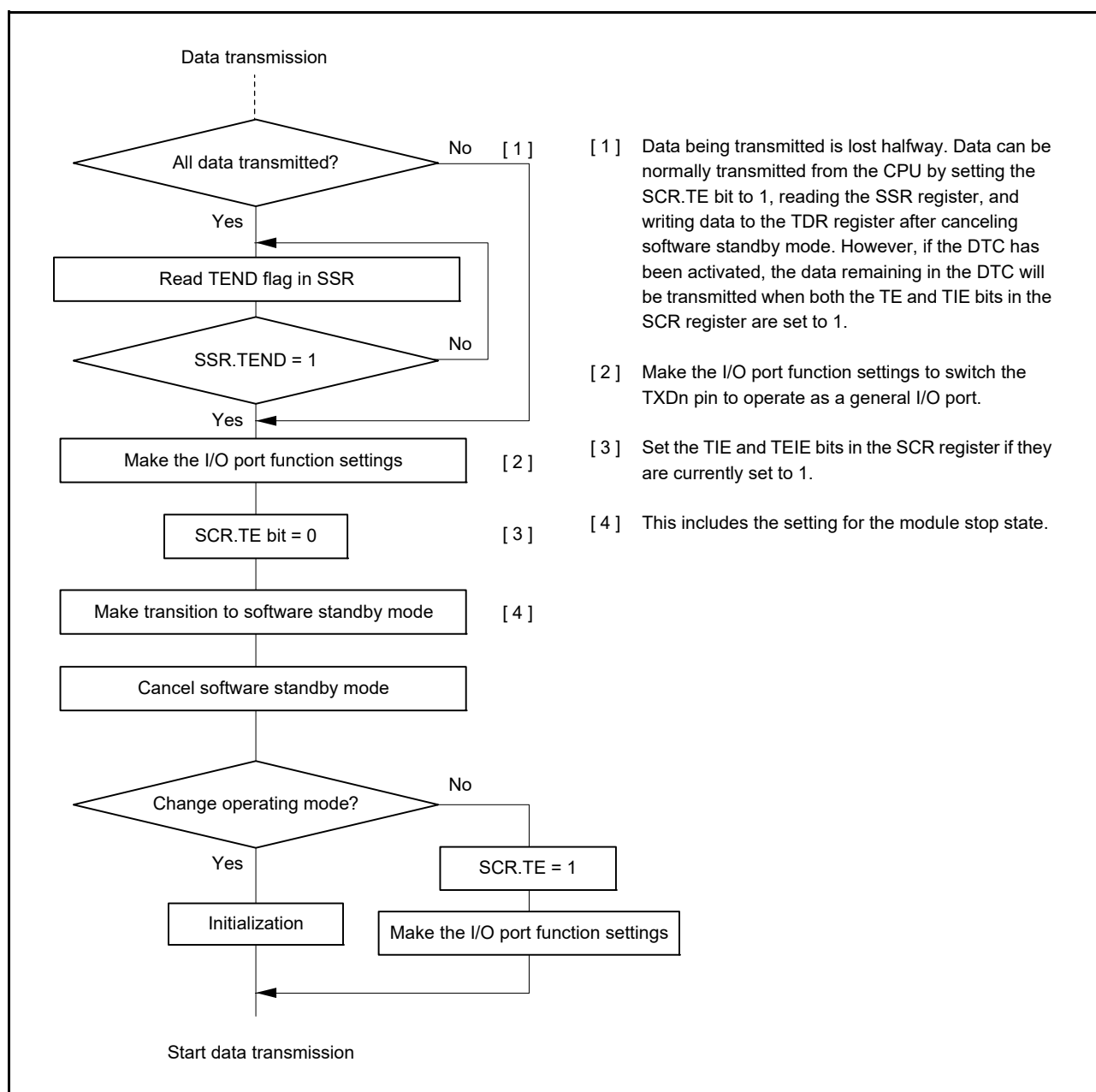


Figure 26.77 Example of Flowchart for Transition to Software Standby Mode during Transmission

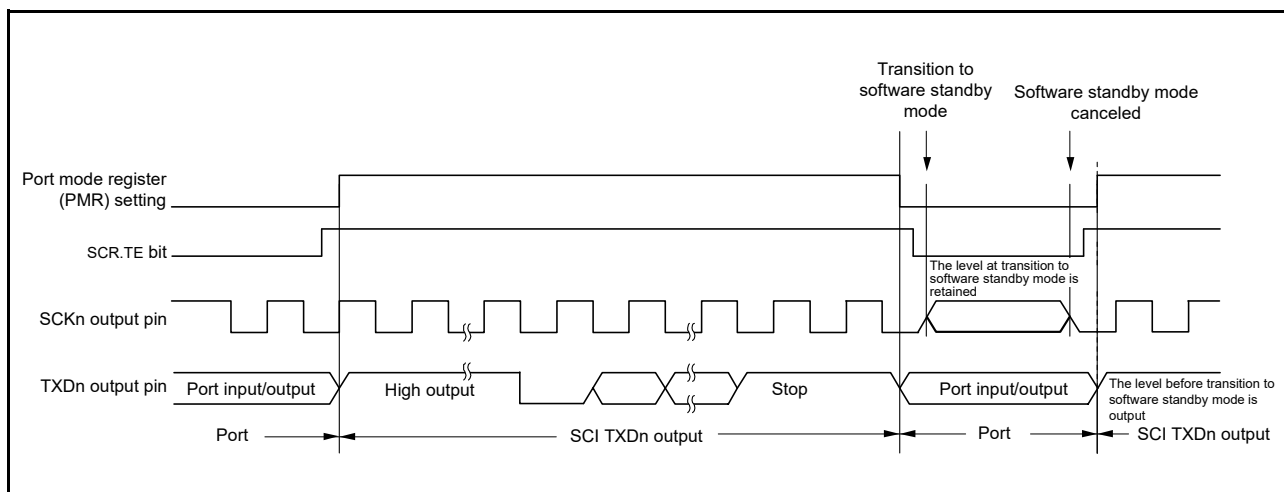


Figure 26.78 Port Pin States during Transition to Software Standby Mode (Internal Clock, Asynchronous Transmission)

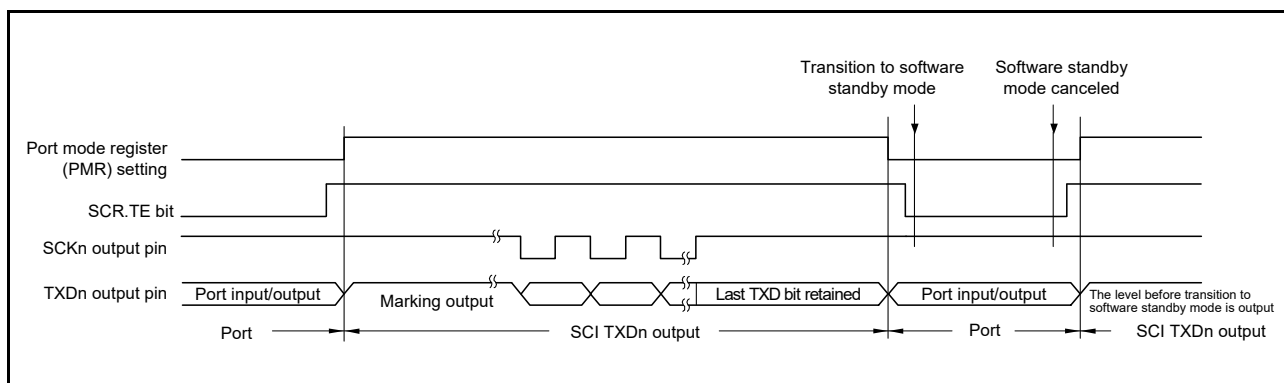


Figure 26.79 Port Pin States during Transition to Software Standby Mode (Internal Clock, Clock Synchronous Transmission)

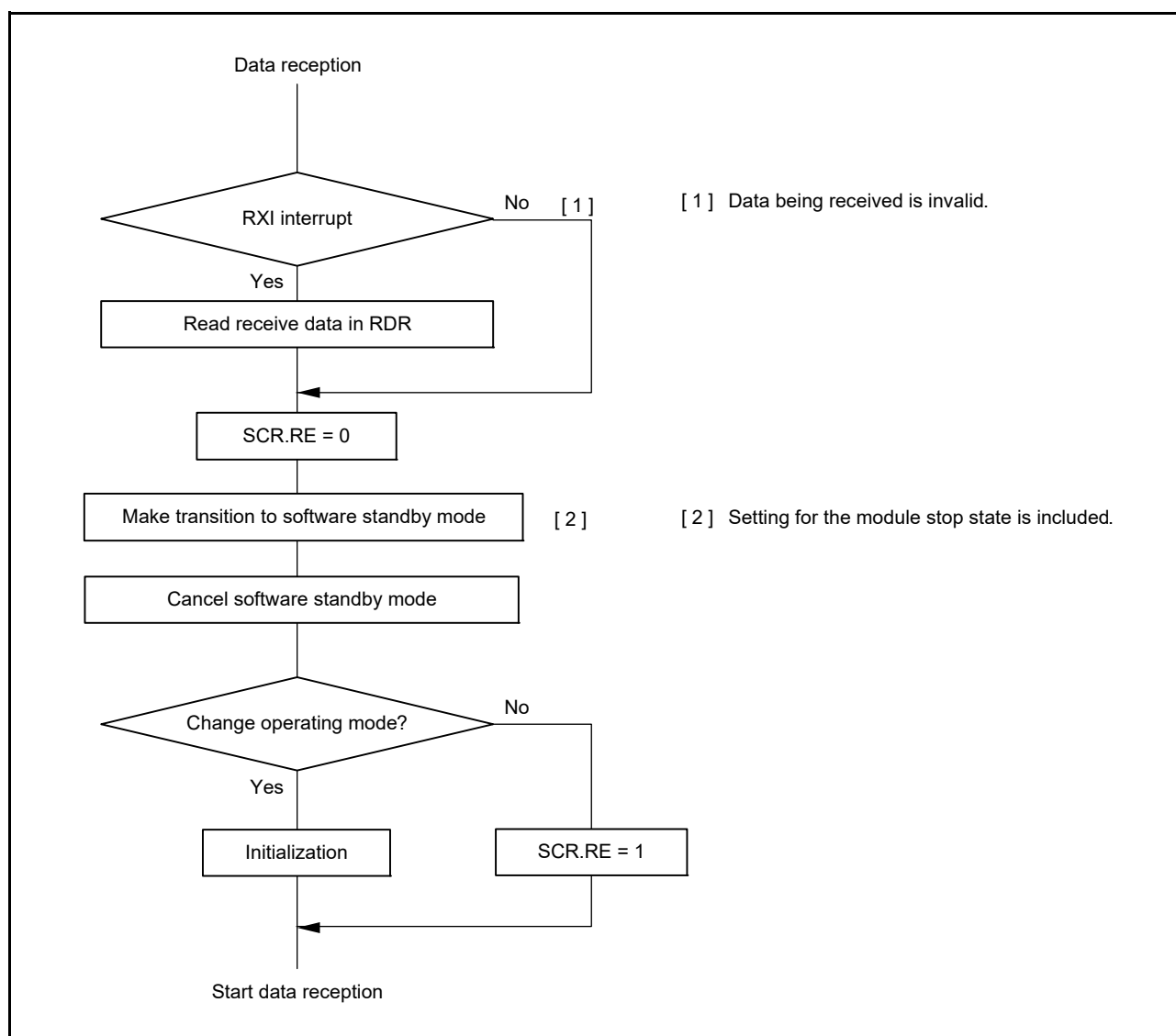


Figure 26.80 Example of Flowchart for Transition to Software Standby Mode during Reception

26.13.10 External Clock Input in Clock Synchronous Mode and Simple SPI Mode

In clock synchronous mode and simple SPI mode, the external clock SCKn must be input as follows:

High-pulse period, low-pulse period = 2 PCLK cycles or more, period = 6 PCLK cycles or more

26.13.11 Limitations on Simple SPI Mode

(1) Master Mode

- Use a resistor to pull up or pull down the clock line matching the initial settings for the transfer clock set by the SPMR.CKPH and CKPOL bits when the SPMR.SSE bit is 1.
This prevents the clock line from being placed in the high-impedance state when the SCR.TE bit is set to 0 or unexpected edges from being generated on the clock line when the SCR.TE bit is changed from 0 to 1. When the SPMR.SSE bit is 0 in single master mode, pulling up or pulling down the clock line is not necessary because the clock line is not placed in the high-impedance state even when the SCR.TE bit is set to 0.
- In the case of the setting for clock delay (SPMR.CKPH bit is 1), the receive data full interrupt (RXI) is generated before the final clock edge on the SCKn pin as indicated in Figure 26.81. If the TE and RE bits in the SCR register become 0 at this time before the final edge of the clock signal on the SCKn pin, the SCKn pin is placed in the high-impedance state, so the width of the last clock pulse of the transfer clock is shortened. Furthermore, an RXI interrupt may lead to the input signal on the SSn# pin of a connected slave going to the high level before the final edge of the clock signal on the SCKn pin, leading to incorrect operation of the slave.
- In a multi-master configuration, take care because the SCKn pin output becomes high-impedance while the input on the SSn# pin is at the low level if a mode fault error occurs as the current character is being transferred, stopping supply of the clock signal to the connected slave. Remake the settings for the connected slave to avoid misaligned bits when transfer is restarted.

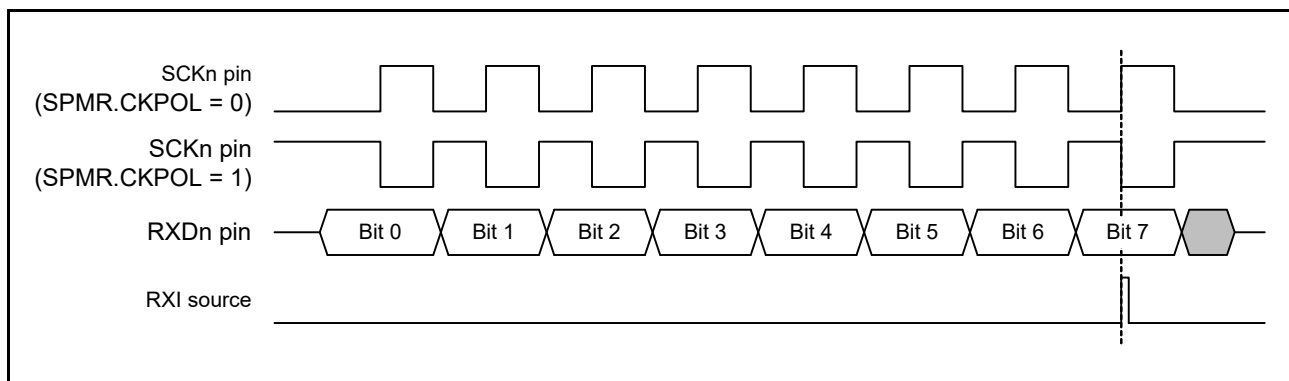


Figure 26.81 Timing of the RXI Interrupt in Simple SPI Mode (with Clock Delay)

(2) Slave Mode

- Secure at least five cycles of the PCLK from writing transmit data in the TDR register to start of the external clock input. Also secure at least five cycles of the PCLK from input of low level on the SSn# pin to start of the external clock input.
- Provide an external clock signal from the master the same as the transmit/receive data length.
- Control the input on the SSn# pin before the start and after the end of data transfer.
- When the level being input on the SSn# pin is to be changed from low to high while the current character is being transferred, set the TE and RE bits in the SCR register to 0 and, after remaking the settings, restart transfer of the first byte.

26.13.12 Limitation 1 on Usage of the Extended Serial Mode Control Section

When the PCR.SHARPS bit is set to 1, output on the TXDX12/RXDX12 pin is only possible when the following conditions apply.

- The timer is in Break Field low width output mode and the value of the TCR.TCST bit is 1 (when the TCST bit is set to 1, the high level continues to be output for up to one cycle of the clock source for counting by the timer counter before output of the low level)
- The value of the SCR.TE bit is 1.

26.13.13 Limitation 2 on Usage of the Extended Serial Mode Control Section

The TXI, RXI, ERI, and TEI interrupt requests are generated even if the extended serial mode is enabled. However, the RXI interrupt should not be enabled during reception of a Start Frame because the extended serial mode control section uses the receive data full signal.

To use the RXI interrupts during a reception of the Information Frame, use it in accordance with one of the following procedures. When a receive error is detected, clear the receive error flag and initialize the extended serial mode control section.

- (1) Set the SCR.RIE bit to 0 to disable the output of interrupt requests. Check the error flags in the SSR register on completion of the reception of a Start Frame, because an ERI interrupt is not generated if a receive error occurs. After reception of the Start Frame is completed, set the SCR.RIE bit to 1 by the time the first byte of the Information Frame is received.
- (2) Set the SCR.RIE bit to 1 to disable RXI interrupts and enable ERI interrupts for ICU. Clear the IRn.IR flag to enable the acceptance of RXI interrupts by ICU by the time the first byte of the Information Frame is received after the completion of Start Frame reception.

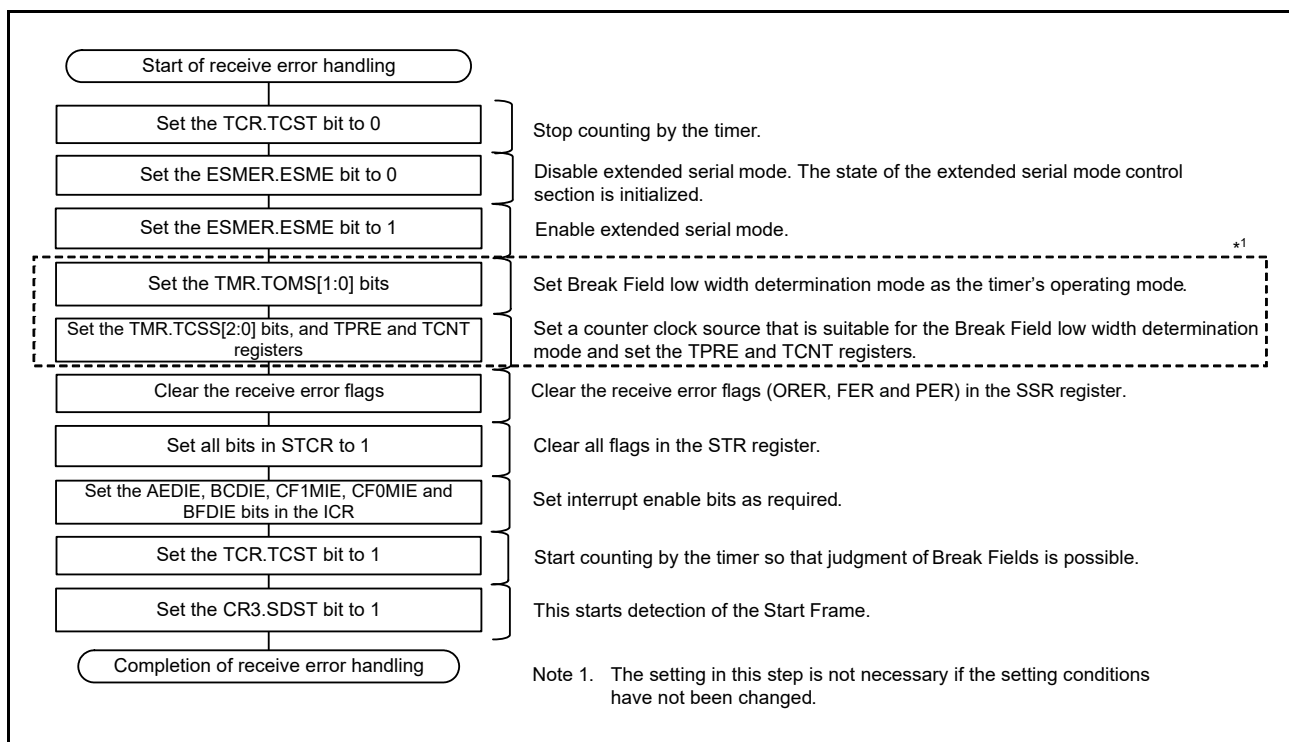


Figure 26.82 Example of Flowchart for Receive Error Handling (during Reception of the Start Frame)

26.13.14 Note on Transmit Enable Bit (TE Bit)

When setting the pin function to “TXDn” while the SCR.TE bit is 0 (serial transmission is disabled) or setting the TE bit to 0 while the pin function is “TXDn”, output of the TXDn pin becomes high-impedance.

Prevent the TXDn line from becoming high-impedance by any of the following ways:

- (1) Connect a pull-up resistor to the TXDn line.
- (2) Set the TE bit to 1*1 before changing the pin function to “TXDn”. Before setting the TE bit to 0, change the pin function to “general-purpose I/O port” and drive the pin high or low.

Note 1. An interrupt is generated when the TE bit is set to 1 while the TXI interrupt is enabled. If this creates a problem, change the pin function to “TXDn” first, and then set the corresponding ICU.IERm.IENj bit to 1.

26.13.15 Note on Stopping Reception When Using the RTS Function in Asynchronous Mode

One clock cycle of PCLK is required for the time from setting the SCR.RE bit to 0 to stopping the RTS signal generator in asynchronous mode.

When reading the RDR (or RDRL) register after setting the SCR.RE bit to 0, confirm that the RE bit has been set to 0 before reading the RDR (or RDRL) register to prevent these two processes from being performed consecutively.

27. I²C-bus Interface (RIICa)

This MCU has a single-channel I²C-bus interface (RIIC0).

The RIIC module conforms with the NXP I²C-bus (Inter-IC bus) interface and provides a subset of its functions.

In this section, “PCLK” is used to refer to PCLKB.

27.1 Overview

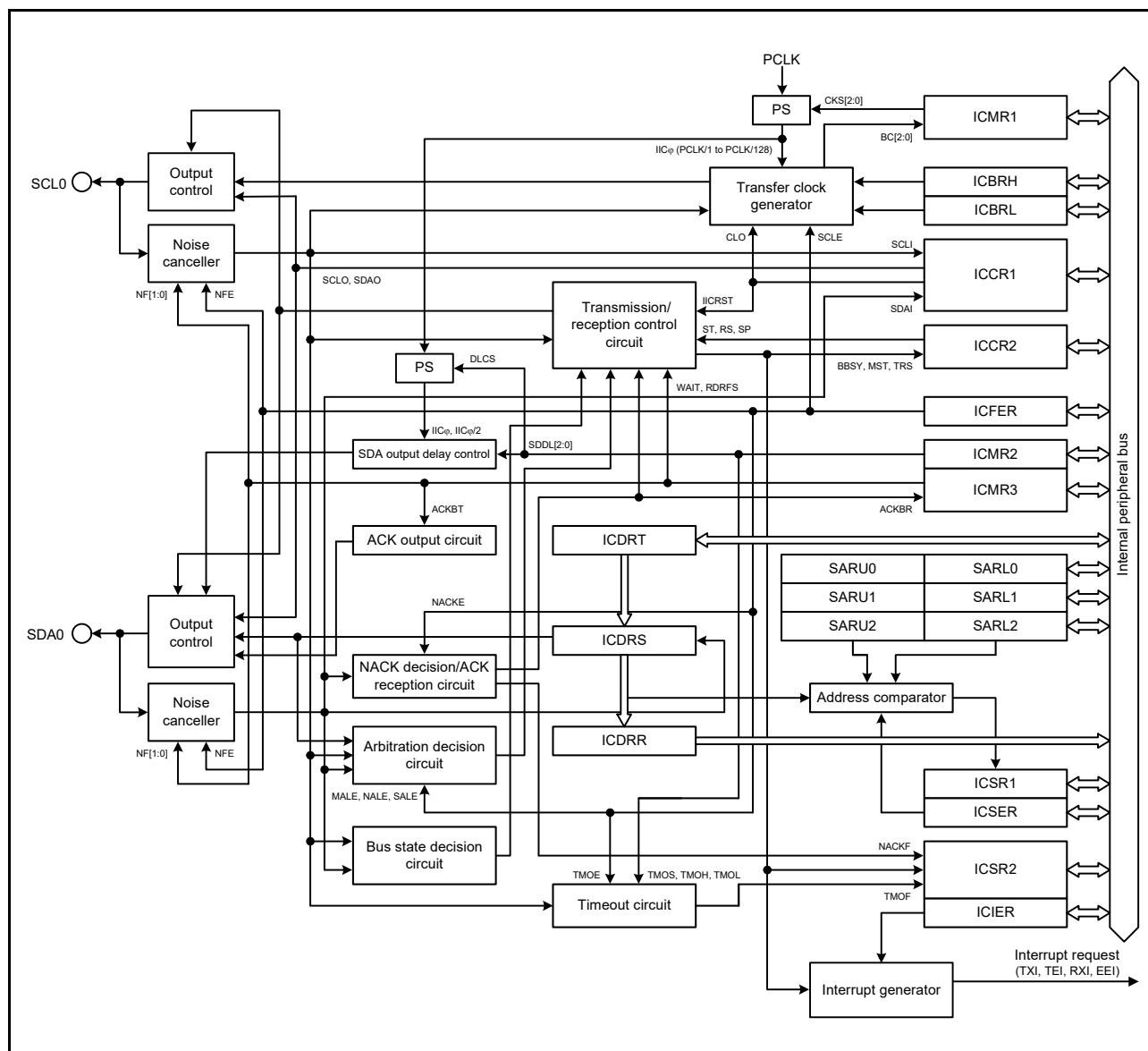
Table 27.1 lists the specifications of the RIIC, Figure 27.1 shows a block diagram of the RIIC. Table 27.2 lists the I/O pins of the RIIC.

Table 27.1 RIIC Specifications (1/2)

Item	Description
Communications format	- I²C-bus format or SMBus format - Master mode or slave mode selectable - Automatic securing of the various setup times, hold times, and bus-free times for the transfer rate
Transfer rate	Fast-mode is supported (up to 400 kbps)
Serial clock (SCL)	For master operation, the duty cycle of the SCL is selectable in the range from 4 to 96%.
Generating and detecting conditions	Start, restart, and stop conditions are automatically generated. Start conditions (including restart conditions) and stop conditions are detectable.
Slave address	- Up to three different slave addresses can be set. 7-bit and 10-bit address formats are supported (along with the use of both at once). - General call addresses, device ID addresses, and SMBus host addresses are detectable.
Acknowledgment	- For transmission, the acknowledgment bit is automatically loaded. Transfer of the next data for transmission can be automatically suspended on detection of a not-acknowledge signal. - For reception, the acknowledgment bit is automatically transmitted. If a wait between the eighth and ninth clock pulses has been selected, software control of the acknowledgment in response to the received data is possible.
Wait function	- During reception, cycles of waiting by holding the SCL line low can be inserted at the following two types of timing: - Waiting between the eighth and ninth clock pulses - Waiting between the ninth clock pulse and the first clock pulse of the next byte
SDA output delay function	Changes in the timing of the output of data bits for transmission, and of the acknowledgment bit, can be delayed relative to the falling edge of SCL.
Arbitration	- For multi-master operation - Clock synchronization for the SCL line in cases of conflict with the SCL signal from another master is possible. - When generating the start condition would create conflict on the bus, loss in arbitration is detected by testing for non-matching between the internal data level and the actual level on the SDA line. - During master operation, loss in arbitration is detected by testing for non-matching between the actual level on the SDA line and the internal data level. - Loss in arbitration due to detection of the start condition while the bus is busy is detectable (to prevent the generating of double start conditions). - Loss in arbitration in transfer of a not-acknowledge signal due to the internal signal (NACK) and the actual level on the SDA line not matching is detectable. - Loss in arbitration due to non-matching of internal data level and the actual level on the SDA line is detectable in slave transmission.
Timeout function	The internal timeout function is capable of detecting long-interval stop of the SCL.
Noise cancellation	The interface incorporates digital noise filters for both the SCL and SDA signals, and the width for noise cancellation by the filters is adjustable by software.
Interrupt sources	Four sources: - Communication error/communication event - Detection of arbitration-lost, NACK, timeout, a start condition including a restart condition, or a stop condition - Receive data full (including matching with a slave address) - Transmit data empty (including matching with a slave address) - Transmission end

Table 27.1 RIIC Specifications (2/2)

Item	Description
Low power consumption function	Module stop state can be set.
RIIC operating modes	Four Master transmit mode, master receive mode, slave transmit mode, and slave receive mode

**Figure 27.1 RIIC Block Diagram**

The logic levels of the input signals for RIIC are CMOS when the I²C-bus is selected (ICMR3.SMBS bit is 0), or TTL when the SMBus is selected (ICMR3.SMBS bit is 1).

Table 27.2 RIIC Pin Configuration

Channel	Pin Name	I/O	Function
RIIC0	SCL0	I/O	RIIC0 serial clock I/O pin
	SDA0	I/O	RIIC0 serial data I/O pin

27.2 Register Descriptions

27.2.1 I²C-bus Control Register 1 (ICCR1)

Address(es): RIIC0.ICCR1 0008 8300h

b7	b6	b5	b4	b3	b2	b1	b0
ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
Value after reset:	0	0	0	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	SDAI	SDA Line Monitor	0: SDA0 line is low. 1: SDA0 line is high.	R
b1	SCLI	SCL Line Monitor	0: SCL0 line is low. 1: SCL0 line is high.	R
b2	SDAO	SDA Output Control/Monitor	- Read: 0: The RIIC has driven the SDA0 pin low. 1: The RIIC has released the SDA0 pin. - Write: 0: The RIIC drives the SDA0 pin low. 1: The RIIC releases the SDA0 pin. (High level output is achieved through an external pull-up resistor.)	R/W
b3	SCLO	SCL Output Control/Monitor	- Read: 0: The RIIC has driven the SCL0 pin low. 1: The RIIC has released the SCL0 pin. - Write: 0: The RIIC drives the SCL0 pin low. 1: The RIIC releases the SCL0 pin. (High level output is achieved through an external pull-up resistor.)	R/W
b4	SOWP	SCLO/SDAO Write Protect	0: SCLO and SDA0 bits can be written. 1: SCLO and SDA0 bits are protected. (This bit is read as 1.)	R/W
b5	CLO	Additional SCL Output	0: Does not output an additional SCL (default). 1: Outputs an additional SCL. (The CLO bit is cleared automatically after one clock pulse is output.)	R/W
b6	IICRST	I ² C-bus Interface Internal Reset	0: Releases the RIIC reset or internal reset. 1: Initiates the RIIC reset or internal reset. (Clears the bit counter and the SCL0/SDA0 output latch)	R/W
b7	ICE	I ² C-bus Interface Enable	0: Disable (SCL0 and SDA0 pins in inactive state) 1: Enable (SCL0 and SDA0 pins in active state) (Combined with the IICRST bit to select either RIIC or internal reset.)	R/W

SDAO Bit (SDA Output Control/Monitor) and SCLO Bit (SCL Output Control/Monitor)

These bits are used to directly control the SDA0 and SCL0 signals output from the RIIC.

When writing to these bits, also write 0 to the SOWP bit.

The result of setting these bits is input to the RIIC via the input buffer. When slave mode is selected, a start condition may be detected and the bus may be released depending on the bit settings.

Do not rewrite these bits during a start condition, stop condition, restart condition, or during transmission or reception.

Operation after rewriting under the above conditions is not guaranteed.

When reading these bits, the state of signals output from the RIIC can be read.

CLO Bit (Additional SCL Output)

This bit is used to output an additional SCL for debugging or error processing.

Normally, set the bit to 0. Setting the bit to 1 in a normal communication state causes a communication error.

For details on this function, refer to section 27.11.2, Additional SCL Output Function.

IICRST Bit (I²C-bus Interface Internal Reset)

This bit is used to reset the internal states of the RIIC.

Setting this bit to 1 initiates an RIIC reset or internal reset.

Whether an RIIC reset or internal reset is initiated is determined according to the combination with the ICE bit. Table 27.3 lists the resets of the RIIC.

The RIIC reset initializes all registers and internal states of the RIIC, and the internal reset initializes the bit counter (ICMR1.BC[2:0] bits), the I²C-bus shift register (ICDRS), and the I²C-bus status registers (ICSR1 and ICSR2) as well as the internal states of the RIIC. For the reset conditions for each register, refer to section 27.14, Initialization of Registers and Functions When a Reset is Applied or a Condition is Detected.

An internal reset initiated with the IICRST bit set to 1 during operation (with the ICE bit set to 1) resets the internal states of the RIIC without initializing the port settings and the control and setting registers of the RIIC when the bus or RIIC hangs up due to a communication error.

If the RIIC hangs up in a low level output state, resetting the internal states cancels the low level output state and releases the bus with the SCL0 pin and SDA0 pin at a high impedance.

Note: If an internal reset is initiated using the IICRST bit for a bus hang-up occurred during communication with the master device in slave mode, the states may become different between the slave device and the master device (due to the difference in the bit counter information). For this reason, do not initiate an internal reset in slave mode, but initiate restoration processing from the master device. If it is necessary to perform an internal reset in slave mode, perform it during bus free state. If an internal reset is necessary because the RIIC has hung with the SCL0 line in a low level output state in slave mode, initiate an internal reset and then generate a restart condition from the master device or resume communications from the start condition after having generated a stop condition. If communication is restarted by initiating a reset solely in the slave device without generating a start condition or restart condition from the master device, synchronization will be lost because the master and slave devices operate asynchronously.

Table 27.3 RIIC Resets

IICRST	ICE	State	Specifications
1	0	RIIC reset	Resets all registers and internal states of the RIIC.
	1	Internal reset	Resets the ICMR1.BC[2:0] bits, registers ICSR1, ICSR2, and ICDRS, and the internal states of the RIIC.

ICE Bit (I²C-bus Interface Enable)

This bit selects the active or inactive state of the SCL0 and SDA0 pins. It can also be combined with the IICRST bit to initiate two types of resets. See Table 27.3, RIIC Resets, for the types of resets.

Set the ICE bit to 1 when using the RIIC. The SCL0 and SDA0 pins are placed in the active state when the ICE bit is set to 1.

Set the ICE bit to 0 when the RIIC is not to be used. The SCL0 and SDA0 pins are placed in the inactive state when the ICE bit is set to 0. Do not assign the SCL0 or SDA0 pin to the RIIC when setting up the multi-function pin controller (MPC). Note that the slave address comparison operation is carried out if the pins are assigned to the RIIC.

27.2.2 I²C-bus Control Register 2 (ICCR2)

Address(es): RIIC0.ICCR2 0008 8301h

b7	b6	b5	b4	b3	b2	b1	b0
BBSY	MST	TRS	—	SP	RS	ST	—
Value after reset: 0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	ST	Start Condition Generation Request	0: Does not request to generate a start condition. 1: Requests to generate a start condition.	R/W
b2	RS	Restart Condition Generation Request	0: Does not request to generate a restart condition. 1: Requests to generate a restart condition.	R/W
b3	SP	Stop Condition Generation Request	0: Does not request to generate a stop condition. 1: Requests to generate a stop condition.	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	TRS	Transmit/Receive Mode	0: Receive mode 1: Transmit mode	R/W*1
b6	MST	Master/Slave Mode	0: Slave mode 1: Master mode	R/W*1
b7	BBSY	Bus Busy Detection Flag	0: The I ² C-bus is released (bus free state). 1: The I ² C-bus is occupied (bus busy state).	R

Note 1. When the ICMR1.MTWP bit is set to 1, bits MST and TRS can be written to.

ST Bit (Start Condition Generation Request)

This bit is used to request transition to master mode and generation of a start condition.

When this bit is set to 1 to request to generate a start condition, a start condition is generated when the BBSY flag is set to 0 (bus free state).

For details on the start condition generation, refer to section 27.10, Start Condition/Restart Condition/Stop Condition Generating Function.

[Setting condition]

- When 1 is written to the ST bit

[Clearing conditions]

- When 0 is written to the ST bit
- When a start condition has been generated (a start condition is detected)
- When the ICSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

Note: Set the ST bit to 1 (requests to generate a start condition) when the BBSY flag is set to 0 (bus free state).
Note that arbitration may be lost due to a start condition generation error if the ST bit is set to 1 (requests to generate a start condition) when the BBSY flag is set to 1 (bus busy state).

RS Bit (Restart Condition Generation Request)

This bit is used to request that a restart condition be generated in master mode.

When this bit is set to 1 to request to generate a restart condition, a restart condition is generated when the BBSY flag is set to 1 (bus busy state) and the MST bit is set to 1 (master mode).

For details on the restart condition generation, refer to section 27.10, Start Condition/Restart Condition/Stop Condition Generating Function.

[Setting condition]

- When 1 is written to the RS bit with the ICCR2.BBSY flag set to 1

[Clearing conditions]

- When 0 is written to the RS bit
- When a restart condition has been generated (a start condition is detected)
- When the ICSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

Note: Do not set the RS bit to 1 while generating a stop condition.

Note: If 1 (requests to generate a restart condition) is written to the RS bit in slave mode, the restart condition is not generated but the RS bit remains set to 1. If the operating mode changes to master mode with the bit not being cleared, note that the restart condition may be generated.

SP Bit (Stop Condition Generation Request)

This bit is used to request that a stop condition be generated in master mode.

When this bit is set to 1 to request to generate a stop condition, a stop condition is generated when the BBSY flag is set to 1 (bus busy state) and the MST bit is set to 1 (master mode).

For details on the stop condition generation, refer to section 27.10, Start Condition/Restart Condition/Stop Condition Generating Function.

[Setting condition]

- When 1 is written to the SP bit with both the BBSY flag and the ICCR2.MST bit set to 1

[Clearing conditions]

- When 0 is written to the SP bit
- When a stop condition has been generated (a stop condition is detected)
- When the ICSR2.AL (arbitration-lost) flag is set to 1
- When a start condition and a restart condition are detected
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

Note: Writing to the SP bit is not possible while the setting of the BBSY flag is 0 (bus free state).

Note: Do not set the SP bit to 1 while a restart condition is being generated.

TRS Bit (Transmit/Receive Mode)

This bit indicates transmit or receive mode.

The RIIC is in receive mode when the TRS bit is set to 0 and is in transmit mode when the bit is set to 1. Combination of this bit and the MST bit indicates the operating mode of the RIIC.

The value of the TRS bit is automatically changed to 1 for transmission or 0 for reception in response to the generation or detection of a start condition and setting of the R/W# bit. Although writing to the TRS bit is possible when the ICMR1.MTWP bit is set to 1, writing to this bit is not necessary during normal usage.

[Setting conditions]

- When a start condition is generated normally according to the start condition generation request (when a start condition is detected with the ST bit set to 1)
- When a restart condition is generated normally according to the restart condition generation request (when a restart condition is detected with the RS bit set to 1)
- When the R/W# bit added to the slave address is set to 0 in master mode
- When the address received in slave mode matches the address enabled in the IC SER register, with the R/W# bit set to 1
- When 1 is written to the TRS bit with the ICMR1.MTWP bit set to 1

[Clearing conditions]

- When a stop condition is detected
- The ICSR2.AL (arbitration-lost) flag being set to 1
- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended
- In slave mode, a match between the received address and the address enabled in the IC SER register when the value of the received R/W# bit is 0 (including cases where the received address is the general call address)
- In slave mode, a restart condition is detected (that is, a start condition is detected while the ICCR2.BBSY flag is 1 and the ICCR2.MST bit is 0)
- When 0 is written to the TRS bit with the ICMR1.MTWP bit set to 1
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

MST Bit (Master/Slave Mode)

This bit indicates master or slave mode.

The RIIC is in slave mode when the MST bit is set to 0 and is in master mode when the bit is set to 1. Combination of this bit and the TRS bit indicates the operating mode of the RIIC.

The value of the MST bit is automatically changed to 1 for master mode or 0 for slave mode by generating of a start condition and generating or detection of a stop condition, etc. Although writing to the MST bit is possible when the ICMR1.MTWP bit is set to 1, writing to this bit is not necessary during normal usage.

[Setting conditions]

- When a start condition is generated normally according to the start condition generation request (when a start condition is detected with the ST bit set to 1)
- When 1 is written to the MST bit with the ICMR1.MTWP bit set to 1

[Clearing conditions]

- When a stop condition is detected
- When the ICSR2.AL (arbitration-lost) flag is set to 1
- When 0 is written to the MST bit with the ICMR1.MTWP bit set to 1
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

BBSY Flag (Bus Busy Detection Flag)

The BBSY flag indicates whether the I²C-bus is occupied (bus busy state) or released (bus free state).

This bit is set to 1 when the SDA0 line changes from high to low under the condition of SCL0 line = high, assuming that a start condition has been generated.

The RIIC recognizes the SDA0 line changing from low to high while the SCL0 line is high as generation of the stop condition. After that, this flag becomes 0 if the RIIC does not detect a start condition during the bus free time (the period set in the ICBRL register).

[Setting condition]

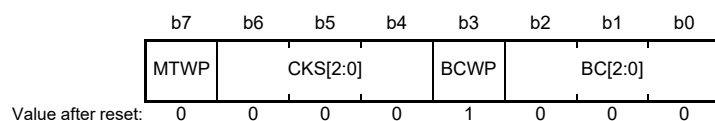
- When a start condition is detected

[Clearing conditions]

- When the bus free time (specified in the ICBRL register) start condition is not detected after detecting a stop condition
- When 1 is written to the ICCR1.IICRST bit with the ICCR1.ICE bit set to 0 (RIIC reset)

27.2.3 I²C-bus Mode Register 1 (ICMR1)

Address(es): RIIC0.ICMR1 0008 8302h



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	BC[2:0]	Bit Counter	b2 b0 0 0 0: 9 bits 0 0 1: 2 bits 0 1 0: 3 bits 0 1 1: 4 bits 1 0 0: 5 bits 1 0 1: 6 bits 1 1 0: 7 bits 1 1 1: 8 bits	R/W*1
b3	BCWP	BC Write Protect	0: Enables a value to be written in the BC[2:0] bits. (This bit is read as 1.)	R/W*1
b6 to b4	CKS[2:0]	Internal Reference Clock Select	Select the internal reference clock (IICφ) source for the RIIC. b6 b4 0 0 0: PCLK/1 0 0 1: PCLK/2 0 1 0: PCLK/4 0 1 1: PCLK/8 1 0 0: PCLK/16 1 0 1: PCLK/32 1 1 0: PCLK/64 1 1 1: PCLK/128	R/W
b7	MTWP	MST/TRS Write Protect	0: Disables writing to the ICCR2.MST and TRS bits. 1: Enables writing to the ICCR2.MST and TRS bits.	R/W

Note 1. Rewrite the BC[2:0] bits and set the BCWP bit to 0 at the same time.

BC[2:0] Bits (Bit Counter)

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a rising edge on the SCL0 line. Although these bits are writable and readable, it is not necessary to access these bits under normal conditions.

To write to these bits, specify the number of bits to be transferred plus one (data is transferred with an additional acknowledgment bit) between transferred bytes when the SCL0 line is low.

The values of the BC[2:0] bits return to 000b at the end of a data transfer including the acknowledgment bit or when a start condition including a restart condition is detected.

27.2.4 I²C-bus Mode Register 2 (ICMR2)

Address(es): RIIC0.ICMR2 0008 8303h

b7	b6	b5	b4	b3	b2	b1	b0
DLCS	SDDL[2:0]			—	TMOH	TMOL	TMOS
Value after reset:							
0	0	0	0	0	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOS	Timeout Detection Time Select	0: Long mode is selected. 1: Short mode is selected.	R/W
b1	TMOL	Timeout L Count Control	0: Count-up is disabled while the SCL0 line is low. 1: Count-up is enabled while the SCL0 line is low.	R/W
b2	TMOH	Timeout H Count Control	0: Count-up is disabled while the SCL0 line is high. 1: Count-up is enabled while the SCL0 line is high.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6 to b4	SDDL[2:0]	SDA Output Delay Counter	- When ICMR2.DLCS bit is 0 (IICφ) b6 b4 0 0 0: No output delay 0 0 1: 1 IICφ cycle 0 1 0: 2 IICφ cycles 0 1 1: 3 IICφ cycles 1 0 0: 4 IICφ cycles 1 0 1: 5 IICφ cycles 1 1 0: 6 IICφ cycles 1 1 1: 7 IICφ cycles - When ICMR2.DLCS bit is 1 (IICφ/2) b6 b4 0 0 0: No output delay 0 0 1: 1 or 2 IICφ cycles 0 1 0: 3 or 4 IICφ cycles 0 1 1: 5 or 6 IICφ cycles 1 0 0: 7 or 8 IICφ cycles 1 0 1: 9 or 10 IICφ cycles 1 1 0: 11 or 12 IICφ cycles 1 1 1: 13 or 14 IICφ cycles	R/W
b7	DLCS	SDA Output Delay Clock Source Select	0: The internal reference clock (IICφ) is selected as the clock source of the SDA output delay counter. 1: The internal reference clock divided by 2 (IICφ/2) is selected as the clock source of the SDA output delay counter.*1	R/W

Note 1. The DLCS bit setting of 1 (IICφ/2) only becomes valid when SCL pin is low. When SCL pin is high, the DLCS bit setting of 1 becomes invalid and the clock source becomes the internal reference clock (IICφ).

TMOS Bit (Timeout Detection Time Select)

This bit is used to select long mode or short mode for the timeout detection time when the timeout function is enabled (ICFER.TMOE bit is 1). When this bit is set to 0, long mode is selected. When this bit is set to 1, short mode is selected. In long mode, the timeout detection internal counter functions as a 16-bit counter. In short mode, the counter functions as a 14-bit counter. While the SCL0 line is in the state that enables this counter as specified by bits TMOH and TMOL, the counter counts up in synchronization with the internal reference clock (IICφ) as a count source.

For details on the timeout function, refer to section 27.11.1, Timeout Function.

TMOL Bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL0 line is held low when the timeout function is enabled (ICFER.TMOE bit is 1).

TMOH Bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL0 line is held high when the timeout function is enabled (ICFER.TMOE bit is 1).

SDDL[2:0] Bits (SDA Output Delay Counter)

The SDA output can be delayed by the SDDL[2:0] setting. This counter works with the clock source selected by the DLCS bit. The setting of this function can be used for all types of SDA output, including the transmission of the acknowledgment bit.

Set the SDA output delay time to meet the I²C-bus specification (within the data valid time/data valid acknowledge time*¹) or the SMBus specification (more than the data hold time (300 ns) and less than “clock low period – data setup time (250 ns)”). Note that, if a value outside the specification is set, communication with communication devices may malfunction or it may seemingly become a start condition or stop condition depending on the bus state.

For details on this function, refer to section 27.5, SDA Output Delay Function.

Note 1. Data valid time/data valid acknowledge time
3,450 ns (up to 100 kbps: Standard-mode (Sm))
900 ns (up to 400 kbps: Fast-mode (Fm))

27.2.5 I²C-bus Mode Register 3 (ICMR3)

Address(es): RIIC0.ICMR3 0008 8304h

b7	b6	b5	b4	b3	b2	b1	b0
SMBS	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1:0]	
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b1, b0	NF[1:0]	Noise Filter Stage Select	b1 b0 0 0: Noise of up to one IIC ϕ cycle is filtered out (single-stage filter). 0 1: Noise of up to two IIC ϕ cycles is filtered out (2-stage filter). 1 0: Noise of up to three IIC ϕ cycles is filtered out (3-stage filter). 1 1: Noise of up to four IIC ϕ cycles is filtered out (4-stage filter).	R/W
b2	ACKBR	Received Acknowledge	0: 0 is received as the acknowledgment bit (ACK reception). 1: 1 is received as the acknowledgment bit (NACK reception).	R
b3	ACKBT	Transmit Acknowledge	0: 0 is to be sent as the acknowledgment bit (ACK transmission). 1: 1 is to be sent as the acknowledgment bit (NACK transmission).	R/W*1
b4	ACKWP	ACKBT Write Protect	0: Modification of the ACKBT bit is disabled. 1: Modification of the ACKBT bit is enabled.	R/W*1
b5	RDRFS	RDRF Flag Set Timing Select	0: The RDRF flag is set at the rising edge of the ninth SCL. (The SCL0 line is not held low at the falling edge of the eighth clock pulse.) 1: The RDRF flag is set at the rising edge of the eighth SCL. (The SCL0 line is held low at the falling edge of the eighth clock pulse.) Low-hold is released by writing a value to the ACKBT bit.	R/W*2
b6	WAIT	WAIT	0: No WAIT (The period between ninth clock pulse and first clock pulse is not held low.) 1: WAIT (The period between ninth clock pulse and first clock pulse is held low.) Low-hold is released by reading the ICDRR register.	R/W*2
b7	SMBS	SMBus/I ² C-bus Select	0: The I ² C-bus is selected. 1: The SMBus is selected.	R/W

Note 1. Write to the ACKBT bit only while the ACKWP bit is already 1. If it is attempted to write 1 to both the ACKWP and ACKBT bits at the same time, the ACKBT bit will not be set to 1.

Note 2. The WAIT and RDRFS bits are valid only in receive mode (invalid in transmit mode).

NF[1:0] Bits (Noise Filter Stage Select)

These bits are used to select the number of stages in the digital noise filter.

For details on the digital noise filter function, refer to section 27.6, Digital Noise Filters.

Note: Set the noise range to be filtered out by the noise filter within a range less than the SCL0 line high period or low period. If the noise filter width is set to a value of [the shorter one of either SCL high width or SCL low width] – 1.5 × $t_{IIC\phi cyc}$ (cycle time of internal reference clock (IIC ϕ)) or a greater value, the serial clock is regarded as noise by the noise filter function of the RIIC, which may prevent the RIIC from operating normally.

ACKBR Bit (Received Acknowledge)

This bit is used to store the value of the acknowledgment bit received from the receiver in transmit mode.

[Setting condition]

- When 1 is received as the acknowledgment bit with the ICCR2.TRS bit set to 1

[Clearing conditions]

- When 0 is received as the acknowledgment bit with the ICCR2.TRS bit set to 1
- When 1 is written to the ICCR1.IICRST bit while the ICCR1.ICE bit is 0 (RIIC reset)

ACKBT Bit (Transmit Acknowledge)

This bit is used to set the bit to be sent at the acknowledgment timing in receive mode.

[Setting condition]

- When 1 is written to this bit with the ACKWP bit set to 1

[Clearing conditions]

- When 0 is written to this bit with the ACKWP bit set to 1
- When stop condition generation is detected (when a stop condition is detected with the ICCR2.SP bit set to 1)
- When 1 is written to the ICCR1.IICRST bit while the ICCR1.ICE bit is 0 (RIIC reset)

ACKWP Bit (ACKBT Write Protect)

This bit is used to control the modification of the ACKBT bit.

RDRFS Bit (RDRF Flag Set Timing Select)

This bit is used to select the RDRF flag set timing in receive mode and also to select whether to hold the SCL0 line low at the falling edge of the eighth SCL.

When the RDRFS bit is 0, the SCL0 line is not held low at the falling edge of the eighth SCL, and the RDRF flag is set to 1 at the rising edge of the ninth SCL.

When the RDRFS bit is 1, the RDRF flag is set to 1 at the rising edge of the eighth SCL and the SCL0 line is held low at the falling edge of the eighth SCL. The low-hold of the SCL0 line is released by writing a value to the ACKBT bit.

After data is received with this setting, the SCL0 line is automatically held low before the acknowledgment bit is sent.

This enables processing to send ACK (ACKBT bit is 0) or NACK (ACKBT bit is 1) according to receive data.

WAIT Bit (WAIT)

This bit is used to control whether to hold the period between the ninth SCL and the first SCL low until the I²C-bus receive data register (ICDRR) is completely read each time single-byte data is received in receive mode.

When the WAIT bit is 0, the receive operation is continued without holding the period between the ninth and the first SCL low. When both the RDRFS and WAIT bits are 0, continuous receive operation is enabled with the double buffer.

When the WAIT bit is 1, the SCL0 line is held low from the falling edge of the ninth SCL until the ICDRR register value is read each time single-byte data is received. This enables receive operation in byte units.

Note: When the value of the WAIT bit is to be read, be sure to read the ICDRR register beforehand.

SMBS Bit (SMBus/I²C-bus Select)

Setting this bit to 1 selects the SMBus and enables the IC SER.HOAE bit.

27.2.6 I²C-bus Function Enable Register (ICFER)

Address(es): RIIC0.ICFER 0008 8305h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
Value after reset:	0	1	1	1	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOE	Timeout Function Enable	0: The timeout function is disabled. 1: The timeout function is enabled.	R/W
b1	MALE	Master Arbitration-Lost Detection Enable	0: Master arbitration-lost detection is disabled. (Disables the arbitration-lost detection function and does not clear the ICCR2.MST and TRS bits automatically when arbitration is lost.) 1: Master arbitration-lost detection is enabled. (Enables the arbitration-lost detection function and clears the ICCR2.MST and TRS bits automatically when arbitration is lost.)	R/W
b2	NALE	NACK Transmission Arbitration-Lost Detection Enable	0: NACK transmission arbitration-lost detection is disabled. 1: NACK transmission arbitration-lost detection is enabled.	R/W
b3	SALE	Slave Arbitration-Lost Detection Enable	0: Slave arbitration-lost detection is disabled. 1: Slave arbitration-lost detection is enabled.	R/W
b4	NACKE	NACK Reception Transfer Suspension Enable	0: Transfer operation is not suspended during NACK reception (transfer suspension disabled). 1: Transfer operation is suspended during NACK reception (transfer suspension enabled).	R/W
b5	NFE	Digital Noise Filter Enable	0: Digital noise filters are not used. 1: Digital noise filters are used.	R/W
b6	SCLE	SCL Synchronization Enable	0: SCL synchronization is disabled. 1: SCL synchronization is enabled.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

TMOE Bit (Timeout Function Enable)

This bit is used to enable or disable the timeout function.

For details on the timeout function, refer to section 27.11.1, Timeout Function.

MALE Bit (Master Arbitration-Lost Detection Enable)

This bit is used to specify whether to use the arbitration-lost detection function in master mode. Normally, set this bit to 1.

NALE Bit (NACK Transmission Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in receive mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).

SALE Bit (Slave Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmit mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).

NACK Bit (NACK Reception Transfer Suspension Enable)

This bit is used to specify whether to continue or discontinue the data transfer when NACK is received in transmit mode. Normally, set this bit to 1.

When NACK is received with the NACK bit set to 1, the next transfer operation is suspended.

When the NACK bit is 0, the next transfer operation is continued regardless of the value of the received acknowledgment bit.

For details on the NACK reception transfer suspension function, refer to [section 27.8.2, NACK Reception Transfer Suspension Function](#).

SCLE Bit (SCL Synchronization Enable)

This bit is used to specify whether the SCL output is to be synchronized with the SCL input. Normally, set this bit to 1. When the SCLE bit is set to 0 (SCL synchronization is disabled), the RIIC does not synchronize the SCL output with the SCL input. In this setting, the RIIC outputs the clock with the transfer rate set in registers ICBRH and ICBRL regardless of the SCL0 line state. For this reason, if the load of the I²C-bus line is much larger than the specification value or if the SCL output overlaps in multiple masters, the short-cycle SCL that does not meet the specification may be output. When the SCL synchronization is not used, it also affects the generation of a start condition, restart condition, and stop condition, and the continuous output of additional SCL.

This bit must not be set to 0 except for checking the output of the set transfer rate.

27.2.7 I²C-bus Status Enable Register (ICSER)

Address(es): RIIC0.ICSER 0008 8306h

	b7	b6	b5	b4	b3	b2	b1	b0
	HOAE	—	DIDE	—	GCAE	SAR2E	SAR1E	SAR0E
Value after reset:	0	0	0	0	1	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	SAR0E	Slave Address Register 0 Enable	0: Slave address in registers SARL0 and SARU0 is disabled. 1: Slave address in registers SARL0 and SARU0 is enabled.	R/W
b1	SAR1E	Slave Address Register 1 Enable	0: Slave address in registers SARL1 and SARU1 is disabled. 1: Slave address in registers SARL1 and SARU1 is enabled.	R/W
b2	SAR2E	Slave Address Register 2 Enable	0: Slave address in registers SARL2 and SARU2 is disabled. 1: Slave address in registers SARL2 and SARU2 is enabled.	R/W
b3	GCAE	General Call Address Enable	0: General call address detection is disabled. 1: General call address detection is enabled.	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	DIDE	Device-ID Address Detection Enable	0: Device-ID address detection is disabled. 1: Device-ID address detection is enabled.	R/W
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	HOAE	Host Address Enable	0: Host address detection is disabled. 1: Host address detection is enabled.	R/W

SARyE Bit (Slave Address Register y Enable) (y = 0 to 2)

This bit is used to enable or disable the slave address set in registers SARLy and SARUy.

When this bit is set to 1, the slave address set in registers SARLy and SARUy is enabled and is compared with the received slave address.

When this bit is set to 0, the slave address set in registers SARLy and SARUy is disabled and is ignored even if it matches the received slave address.

GCAE Bit (General Call Address Enable)

This bit is used to specify whether to ignore the general call address (0000 000b + 0 (write): All 0) when it is received.

When this bit is set to 1, if the received slave address matches the general call address, the RIIC recognizes the received slave address as the general call address independently of the slave addresses set in registers SARLy and SARUy (y = 0 to 2) and performs data receive operation.

When this bit is set to 0, the received slave address is ignored even if it matches the general call address.

DIDE Bit (Device-ID Address Detection Enable)

This bit is used to specify whether to recognize and execute the device-ID address when a device ID (1111 100b) is received in the first byte after a start condition or restart condition is detected.

When this bit is set to 1, if the received first byte matches the device ID, the RIIC recognizes that the device-ID address has been received. When the following R/W# bit is 0 (write), the RIIC recognizes the second and the following bytes as slave addresses and continues the receive operation.

When this bit is set to 0, the RIIC ignores the received first byte even if it matches the device ID address and recognizes the first byte as a normal slave address.

For details on the device-ID address detection, refer to section 27.7.3, Device-ID Address Detection.

HOAE Bit (Host Address Enable)

This bit is used to specify whether to ignore received host address (0001 000b) when the ICMR3.SMBS bit is 1.

When this bit is set to 1 while the ICMR3.SMBS bit is 1, if the received slave address matches the host address, the RIIC recognizes the received slave address as the host address independently of the slave addresses set in registers SARLy and SARUy (y = 0 to 2) and performs the receive operation.

When the ICMR3.SMBS bit or the HOAE bit is set to 0, the received slave address is ignored even if it matches the host address.

27.2.8 I²C-bus Interrupt Enable Register (ICIER)

Address(es): RIIC0.ICIER 0008 8307h

b7	b6	b5	b4	b3	b2	b1	b0
TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	TMOIE	Timeout Interrupt Request Enable	0: Timeout interrupt (TMOI) request is disabled. 1: Timeout interrupt (TMOI) request is enabled.	R/W
b1	ALIE	Arbitration-Lost Interrupt Request Enable	0: Arbitration-lost interrupt (ALI) request is disabled. 1: Arbitration-lost interrupt (ALI) request is enabled.	R/W
b2	STIE	Start Condition Detection Interrupt Request Enable	0: Start condition detection interrupt (STI) request is disabled. 1: Start condition detection interrupt (STI) request is enabled.	R/W
b3	SPIE	Stop Condition Detection Interrupt Request Enable	0: Stop condition detection interrupt (SPI) request is disabled. 1: Stop condition detection interrupt (SPI) request is enabled.	R/W
b4	NAKIE	NACK Reception Interrupt Request Enable	0: NACK reception interrupt (NAKI) request is disabled. 1: NACK reception interrupt (NAKI) request is enabled.	R/W
b5	RIE	Receive Data Full Interrupt Request Enable	0: Receive data full interrupt (RXI) request is disabled. 1: Receive data full interrupt (RXI) request is enabled.	R/W
b6	TEIE	Transmission End Interrupt Request Enable	0: Transmission end interrupt (TEI) request is disabled. 1: Transmission end interrupt (TEI) request is enabled.	R/W
b7	TIE	Transmit Data Empty Interrupt Request Enable	0: Transmit data empty interrupt (TXI) request is disabled. 1: Transmit data empty interrupt (TXI) request is enabled.	R/W

TMOIE Bit (Timeout Interrupt Request Enable)

This bit is used to enable or disable timeout interrupt (TMOI) requests when the ICSR2.TMOF flag is set to 1. A TMOI interrupt request is canceled by setting the TMOF flag or the TMOIE bit to 0.

ALIE Bit (Arbitration-Lost Interrupt Request Enable)

This bit is used to enable or disable arbitration-lost interrupt (ALI) requests when the ICSR2.AL flag is set to 1. An ALI interrupt request is canceled by setting the AL flag or the ALIE bit to 0.

STIE Bit (Start Condition Detection Interrupt Request Enable)

This bit is used to enable or disable start condition detection interrupt (STI) requests when the ICSR2.START flag is set to 1. An STI interrupt request is canceled by setting the START flag or the STIE bit to 0.

SPIE Bit (Stop Condition Detection Interrupt Request Enable)

This bit is used to enable or disable stop condition detection interrupt (SPI) requests when the ICSR2.STOP flag is set to 1. An SPI interrupt request is canceled by setting the STOP flag or the SPIE bit to 0.

NAKIE Bit (NACK Reception Interrupt Request Enable)

This bit is used to enable or disable NACK reception interrupt (NAKI) requests when the ICSR2.NACKF flag is set to 1. An NAKI interrupt request is canceled by setting the NACKF flag or the NAKIE bit to 0.

RIE Bit (Receive Data Full Interrupt Request Enable)

This bit is used to enable or disable receive data full interrupt (RXI) requests when the ICSR2.RDRF flag is set to 1.

TEIE Bit (Transmission End Interrupt Request Enable)

This bit is used to enable or disable transmission end interrupt (TEI) requests when the ICSR2.TEND flag is set to 1. An TEI interrupt request is canceled by setting the TEND flag or the TEIE bit to 0.

TIE Bit (Transmit Data Empty Interrupt Request Enable)

This bit is used to enable or disable transmit data empty interrupt (TXI) requests when the ICSR2.TDRE flag is set to 1.

27.2.9 I²C-bus Status Register 1 (ICSR1)

Address(es): RIIC0.ICSR1 0008 8308h

b7	b6	b5	b4	b3	b2	b1	b0
HOA	—	DID	—	GCA	AAS2	AAS1	AAS0
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	AAS0	Slave Address 0 Detection Flag	0: Slave address 0 is not detected. 1: Slave address 0 is detected.	R/(W) *1
b1	AAS1	Slave Address 1 Detection Flag	0: Slave address 1 is not detected. 1: Slave address 1 is detected.	R/(W) *1
b2	AAS2	Slave Address 2 Detection Flag	0: Slave address 2 is not detected. 1: Slave address 2 is detected.	R/(W) *1
b3	GCA	General Call Address Detection Flag	0: General call address is not detected. 1: General call address is detected.	R/(W) *1
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	DID	Device-ID Address Detection Flag	0: Device-ID command is not detected. 1: Device-ID command is detected. • This bit is set to 1 when the first byte received immediately after a start condition is detected matches a value of (device ID (1111 100b) + 0 (write)).	R/(W) *1
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	HOA	Host Address Detection Flag	0: Host address is not detected. 1: Host address is detected. • This bit is set to 1 when the received slave address matches the host address (0001 000b).	R/(W) *1

Note 1. Only 0 can be written to clear the flag.

AASy Flag (Slave Address y Detection Flag) (y = 0 to 2)

[Setting conditions]

For 7-bit address format: SARUy.FS bit = 0

- When the received slave address matches the SARLy.SVA[6:0] bits value with the ICSEr.SARyE bit set to 1 (slave address y detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL in the first byte.

For 10-bit address format: SARUy.FS bit = 1

- When the received slave address matches a value of (11110b + SARUy.SVA[1:0] bits) and the following address matches the SARLy value with the ICSEr.SARyE bit set to 1 (slave address y detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL in the second byte.

[Clearing conditions]

- When 0 is written to the AASy flag after reading the AASy flag to be 1
- When a stop condition is detected
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

For 7-bit address format: SARUy.FS bit = 0

- When the received slave address does not match the SARLy.SVA[6:0] bits value with the ICSEr.SARyE bit set to 1 (slave address y detection enabled)

This flag is set to 0 at the rising edge of the ninth SCL in the first byte.

For 10-bit address format: SARUy.FS bit = 1

- When the received slave address does not match a value of (11110b + SARUy.SVA[1:0] bits) with the ICSER.SARyE bit set to 1 (slave address y detection enabled)
This flag is set to 0 at the rising edge of the ninth SCL in the first byte.
- When the received slave address matches a value of (11110b + SARUy.SVA[1:0] bits) and the following address does not match the SARLy value with the ICSER.SARyE bit set to 1 (slave address y detection enabled)
This flag is set to 0 at the rising edge of the ninth SCL in the second byte.

GCA Flag (General Call Address Detection Flag)

[Setting condition]

- When the received slave address matches the general call address (0000 000b + 0 (write)) with the ICSER.GCAE bit set to 1 (general call address detection is enabled)
This flag is set to 1 at the rising edge of the ninth SCL in the first byte.

[Clearing conditions]

- When 0 is written to the GCA flag after reading GCA flag to be 1
- When a stop condition is detected
- When the received slave address does not match the general call address (0000 000b + 0 (write)) with the ICSER.GCAE bit set to 1 (general call address detection is enabled)
This flag is set to 0 at the rising edge of the ninth SCL in the first byte.
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

DID Flag (Device-ID Address Detection Flag)

[Setting condition]

- When the first byte received immediately after a start condition or restart condition is detected matches a value of (device ID (1111 100b) + 0 (write)) with the ICSER.DIDE bit set to 1 (device-ID address detection is enabled)
This flag is set to 1 at the rising edge of the ninth SCL in the first byte.

[Clearing conditions]

- When 0 is written to the DID flag after reading DID flag to be 1
- When a stop condition is detected
- When the first byte received immediately after a start condition or restart condition is detected does not match a value of (device ID (1111 100b)) with the ICSER.DIDE bit set to 1 (device-ID address detection is enabled)
This flag is set to 0 at the rising edge of the ninth SCL in the first byte.
- When the first byte received immediately after a start condition or restart condition is detected matches a value of (device ID (1111 100b) + 0 (write)) and the second byte does not match any of slave addresses 0 to 2 with the ICSER.DIDE bit set to 1 (device-ID address detection is enabled)
This flag is set to 0 at the rising edge of the ninth SCL in the second byte.
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

HOA Flag (Host Address Detection Flag)

[Setting condition]

- When the received slave address matches the host address (0001 000b) with the ICSER.HOAE bit set to 1 (host address detection is enabled)
This flag is set to 1 at the rising edge of the ninth SCL in the first byte.

[Clearing conditions]

- When 0 is written to the HOA flag after reading HOA flag to be 1
- When a stop condition is detected
- When the received slave address does not match the host address (0001 000b) with the ICSER.HOAE bit set to 1

(host address detection is enabled)

This flag is set to 0 at the rising edge of the ninth SCL in the first byte.

- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

27.2.10 I²C-bus Status Register 2 (ICSR2)

Address(es): RIIC0.ICSR2 0008 8309h

	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOF	Timeout Detection Flag	0: Timeout is not detected. 1: Timeout is detected.	R/(W) *1
b1	AL	Arbitration-Lost Flag	0: Arbitration is not lost. 1: Arbitration is lost.	R/(W) *1
b2	START	Start Condition Detection Flag	0: Start condition is not detected. 1: Start condition is detected.	R/(W) *1
b3	STOP	Stop Condition Detection Flag	0: Stop condition is not detected. 1: Stop condition is detected.	R/(W) *1
b4	NACKF	NACK Detection Flag	0: NACK is not detected. 1: NACK is detected.	R/(W) *1
b5	RDRF	Receive Data Full Flag	0: The ICDRR register contains no receive data. 1: The ICDRR register contains receive data.	R/(W) *1
b6	TEND	Transmission End Flag	0: Data is being transmitted. 1: Data has been transmitted.	R/(W) *1
b7	TDRE	Transmit Data Empty Flag	0: The ICDRT register contains transmit data. 1: The ICDRT register contains no transmit data.	R

Note 1. Only 0 can be written to clear the flag.

TMOF Flag (Timeout Detection Flag)

This flag is set to 1 when the RIIC recognizes timeout after the SCL0 line state remains unchanged for a certain period.

[Setting condition]

- When the SCL0 line state remains unchanged for the period specified by bits ICMR2.TMOH, TMOL, and TMOS while the ICFER.TMOE bit is 1 (the timeout function is enabled) in master mode or in slave mode and the received slave address matches.

[Clearing conditions]

- When 0 is written to the TMOF bit after reading TMOF = 1
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

AL Flag (Arbitration-Lost Flag)

This flag shows that bus mastership has been lost (loss in arbitration) due to a bus conflict or some other reason when a start condition is generated or an address and data are transmitted. The RIIC monitors the level on the SDA0 line during transmission and, if the level on the line does not match the value of the bit being output, sets the value of the AL flag to 1 to indicate that the bus is occupied by another device.

The RIIC can also detect loss of arbitration during NACK transmission in receive mode or during data transmission in slave mode.

[Setting conditions]

When master arbitration-lost detection is enabled: ICFER.MALE = 1

- When the internal SDA output state does not match the SDA0 line level at the rising edge of SCL except for the ACK period during data (including slave address) transmission in master transmit mode (when the SDA0 line is driven low while the internal SDA output is high (the SDA0 pin is in the high-impedance state))
- When a start condition is detected while the ICCR2.ST bit is 1 (requests to generate a start condition) or the internal SDA output state does not match the SDA0 line level
- When the ICCR2.ST bit is set to 1 (requests to generate a start condition) with the ICCR2.BBSY flag set to 1.

When NACK arbitration-lost detection is enabled: ICFER.NALE = 1

- When the internal SDA output state does not match the SDA0 line level at the rising edge of SCL in the ACK period during NACK transmission in receive mode

When slave arbitration-lost detection is enabled: ICFER.SALE = 1

- When the internal SDA output state does not match the SDA0 line level at the rising edge of SCL except for the ACK period during data transmission in slave transmit mode

[Clearing conditions]

- When 0 is written to the AL flag after reading AL = 1
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

Table 27.4 Relationship between Arbitration-Lost Generation Sources and Arbitration-Lost Enable Functions

ICFER			ICSR2	Error	Arbitration-Lost Generation Source
MALE	NALE	SALE	AL		
1	x	x	1	Start condition generation error	When internal SDA output state does not match SDA0 line level when a start condition is detected while the ICCR2.ST bit is 1
			1	Transmit data mismatch	When ICCR2.ST bit is set to 1 with ICCR2.BBSY flag set to 1
x	1	x	1	NACK transmission mismatch	When ACK is detected during transmission of NACK in master receive mode or slave receive mode
x	x	1	1	Transmit data mismatch	When transmit data does not match the bus state in slave transmit mode

x: Don't care

START Flag (Start Condition Detection Flag)

[Setting condition]

- When a start condition (or a restart condition) is detected

[Clearing conditions]

- When 0 is written to the START bit after reading START = 1
- When a stop condition is detected
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

STOP Flag (Stop Condition Detection Flag)

[Setting condition]

- When a stop condition is detected

[Clearing conditions]

- When 0 is written to the STOP bit after reading STOP = 1
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

NACKF Flag (NACK Detection Flag)

[Setting condition]

- When ACK is not received (NACK is received) from the receiver in transmit mode with the ICFER.NACKE bit set to 1 (transfer suspension enabled)

[Clearing conditions]

- When 0 is written to the NACKF bit after reading NACKF = 1
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

Note: When the NACKF flag is set to 1, the RIIC suspends data transmission/reception. Writing to the ICDRT register in transmit mode or reading from the ICDRR register in receive mode with the NACKF flag set to 1 does not enable data transmit/receive operation. To restart data transmission/reception, set the NACKF flag to 0.

RDRF Flag (Receive Data Full Flag)

[Setting conditions]

- When receive data has been transferred from the ICDRS register to the ICDRR register
This flag is set to 1 at the rising edge of the eighth or ninth SCL (selected by the ICMR3.RDRFS bit)
- When the received slave address matches after a start condition (or a restart condition) is detected with the ICCR2.TRS bit set to 0

[Clearing conditions]

- When 0 is written to the RDRF bit after reading RDRF = 1
- When data is read from the ICDRR register
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

TEND Flag (Transmission End Flag)

[Setting condition]

- At the rising edge of the ninth SCL while the TDRE flag is 1

[Clearing conditions]

- When 0 is written to the TEND bit after reading TEND = 1
- When data is written to the ICDRT register
- When a stop condition is detected
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

TDRE Flag (Transmit Data Empty Flag)

[Setting conditions]

- When data has been transferred from the ICDRT register to the ICDRS register and the ICDRT register becomes empty
- When the ICCR2.TRS bit is set to 1
- When the received slave address matches while the TRS bit is 1

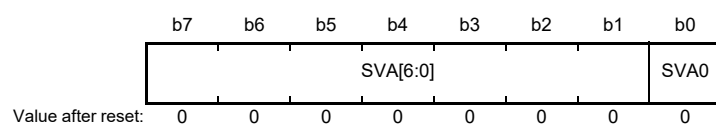
[Clearing conditions]

- When data is written to the ICDRT register
- When the ICCR2.TRS bit is set to 0
- When 1 is written to the ICCR1.IICRST bit to apply an RIIC reset or an internal reset

Note: The NACKF flag becoming 1 while the ICFER.NACKE bit is 1 suspends data transmission and reception by the RIIC. Even if the next data for transmission has already been written to the ICDRT register (the TDRE flag is 0), the data in the ICDRT register is retained but not transferred to the ICDRS register. At this point, the TDRE flag does not become 1.

27.2.11 Slave Address Register Ly (SARLy) (y = 0 to 2)

Address(es): RIIC0.SARL0 0008 830Ah, RIIC0.SARL1 0008 830Ch, RIIC0.SARL2 0008 830Eh



Bit	Symbol	Bit Name	Description	R/W
b0	SVA0	10-Bit Address LSB	Set a slave address	R/W
b7 to b1	SVA[6:0]	7-Bit Address/10-Bit Address Lower Bits	Set a slave address	R/W

SVA0 Bit (10-Bit Address LSB)

When the 10-bit address format is selected (SARUy.FS bit is 1), this bit functions as the LSB of a 10-bit address and forms the lower 8 bits of a 10-bit address in combination with the SVA[6:0] bits.

When the ICSEr.SARyE bit is set to 1 (SARLy and SARUy enabled) and the SARUy.FS bit is 1, this bit is valid. While the SARUy.FS bit or SARyE bit is 0, the setting of this bit is ignored.

SVA[6:0] Bits (7-Bit Address/10-Bit Address Lower Bits)

When the 7-bit address format is selected (SARUy.FS bit is 0), these bits function as a 7-bit address. When the 10-bit address format is selected (SARUy.FS bit is 1), these bits function as the lower 8 bits of a 10-bit address in combination with the SVA0 bit.

While the ICSEr.SARyE bit is 0, the setting of these bits is ignored.

27.2.12 Slave Address Register Uy (SARUy) (y = 0 to 2)

Address(es): RIIC0.SARU0 0008 830Bh, RIIC0.SARU1 0008 830Dh, RIIC0.SARU2 0008 830Fh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	SVA[1:0]	FS	
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	FS	7-Bit/10-Bit Address Format Select	0: The 7-bit address format is selected. 1: The 10-bit address format is selected.	R/W
b2, b1	SVA[1:0]	10-Bit Address Upper Bits	Set a slave address	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

FS Bit (7-Bit/10-Bit Address Format Select)

This bit is used to select 7-bit address or 10-bit address for slave address y (in registers SARLy and SARUy).

When the ICSEr.SARyE bit is set to 1 (registers SARLy and SARUy enabled) and the SARUy.FS bit is 0, the 7-bit address format is selected for slave address y, the SARLy.SVA[6:0] bits setting is valid, and the settings of the SVA[1:0] bits and the SARLy.SVA0 bit are ignored.

When the ICSEr.SARyE bit is set to 1 (registers SARLy and SARUy enabled) and the SARUy.FS bit is 1, the 10-bit address format is selected for slave address y and the settings of the SVA[1:0] bits and SARLy are valid.

While the ICSEr.SARyE bit is 0 (registers SARLy and SARUy disabled), the setting of the SARUy.FS bit is invalid.

SVA[1:0] Bits (10-Bit Address Upper Bits)

When the 10-bit address format is selected (FS = 1), these bits function as the upper 2 bits of a 10-bit address.

When the ICSEr.SARyE bit is set to 1 (SARLy and SARUy enabled) and the SARUy.FS bit is 1, these bits are valid.

While the SARUy.FS bit or SARyE bit is 0, the setting of these bits is ignored.

27.2.13 I²C-bus Bit Rate Low-Level Register (ICBRL)

Address(es): RIIC0.ICBRL 0008 8310h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	BRL[4:0]	Bit Rate Low Period	Low period of SCL	R/W
b7 to b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

ICBRL is a 5-bit register to set the low period of SCL.

It also works to generate the data setup time for automatic SCL low-hold operation (refer to [section 27.8, Automatic Low-Hold Function for SCL](#)); when the RIIC is used only in slave mode, this register needs to be set to a value longer than the data setup time*1.

ICBRL counts the low period with the internal reference clock (IIC ϕ) specified by the ICMR1.CKS[2:0] bits.

If the digital noise filter is enabled (the ICFER.NFE bit is 1), set the ICBRL register to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the ICMR3.NF[1:0] bits.

Note 1. Data setup time (t_{SU;DAT})

250 ns (up to 100 kbps: Standard-mode (Sm))

100 ns (up to 400 kbps: Fast-mode (Fm))

27.2.14 I²C-bus Bit Rate High-Level Register (ICBRH)

Address(es): RIIC0.ICBRH 0008 8311h



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	BRH[4:0]	Bit Rate High Period	High period of SCL	R/W
b7 to b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

ICBRH is a 5-bit register to set the high period of SCL. ICBRH is valid in master mode. If the RIIC is used only in slave mode, this register need not to set the high period.

ICBRH counts the high period with the internal reference clock (IIC ϕ) specified by the ICMR1.CKS[2:0] bits.

If the digital noise filter is enabled (the ICFER.NFE bit is 1), set the ICBRH register to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the ICMR3.NF[1:0] bits.

The I²C transfer rate and the SCL duty are calculated using the following expression.

Transfer rate = $1 / \{[(ICBRH + 1) + (ICBRL + 1)] / IIC\phi + SCL0 \text{ line rise time } [tr] + SCL0 \text{ line fall time } [tf]\}$

Duty cycle = $\{SCL0 \text{ line rise time } [tr]^2 + (ICBRH + 1) / IIC\phi\} / \{SCL0 \text{ line fall time } [tf]^2 + (ICBRL + 1) / IIC\phi\}$

Note 1. $IIC\phi = PCLK \times \text{Division ratio}$

Note 2. The SCL0 line rise time [tr] and SCL0 line fall time [tf] depend on the total bus line capacitance [Cb] and the pull-up resistor [Rp]. For details, refer to the I²C-bus specification from NXP Semiconductors.

Table 27.5 lists examples of ICBRH/ICBRL settings.

Table 27.5 Examples of ICBRH/ICBRL Settings for Transfer Rate

Transfer Rate (kbps)	Operating Frequency PCLK (MHz)								
	8			10			12.5		
	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL
10	100b	22 (F6h)	25 (F9h)	101b	13 (EDh)	15 (EFh)	101b	16 (F0h)	20 (F4h)
50	010b	16 (F0h)	19 (F3h)	010b	21 (F5h)	24 (F8h)	011b	12 (ECh)	15 (EFh)
100	001b	15 (EFh)	18 (F2h)	001b	19 (F3h)	23 (F7h)	001b	24 (F8h)	29 (FDh)
400	000b	4 (E4h)	10 (EAh)	000b	5 (E5h)	12 (ECh)	000b	7 (E7h)	16 (F0h)

Transfer Rate (kbps)	Operating Frequency PCLK (MHz)								
	16			20			25		
	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL
10	101b	22 (F6h)	25 (F9h)	110b	13 (EDh)	15 (EFh)	110b	16 (F0h)	20 (F4h)
50	011b	16 (F0h)	19 (F3h)	011b	21 (F5h)	24 (F8h)	100b	12 (ECh)	15 (EFh)
100	010b	15 (EFh)	18 (F2h)	010b	19 (F3h)	23 (F7h)	010b	24 (F8h)	29 (FDh)
400	000b	9 (E9h)	20 (F4h)	000b	11 (EBh)	25 (F9h)	001b	7 (E7h)	16 (F0h)

Transfer Rate (kbps)	Operating Frequency PCLK (MHz)								
	30			32			33		
	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL
10	110b	20 (F4h)	24 (F8h)	110b	22 (F6h)	25 (F9h)	110b	22 (F6h)	26 (FAh)
50	100b	15 (EFh)	18 (F2h)	100b	16 (F0h)	19 (F3h)	100b	17 (F1h)	20 (F4h)
100	011b	14 (EEh)	17 (F1h)	011b	15 (EFh)	18 (F2h)	011b	16 (F0h)	19 (F3h)
400	001b	8 (E8h)	19 (F3h)	001b	9 (E9h)	20 (F4h)	001b	9 (E9h)	21 (F5h)

Transfer Rate (kbps)	Operating Frequency PCLK (MHz)		
	40		
	CKS[2:0]	ICBRH	ICBRL
10	111b	13 (EDh)	15 (EFh)
50	100b	21 (F5h)	24 (F8h)
100	011b	19 (F3h)	23 (F7h)
400	001b	11 (EBh)	25 (F9h)

Note: ICBRH/ICBRL settings in these tables are calculated using the following values:
 SCL0 line rise time (tr): 100 kbps or less (Sm): 1000 ns, 400 kbps or less (Fm): 300 ns
 SCL0 line fall time (tf): 400 kbps or less (Sm/Fm): 300 ns
 For the specified values of rise time (tr) and fall time (tf) of the SCL0 signal, refer to the I²C-bus specification from NXP Semiconductors.

27.2.15 I²C-bus Transmit Data Register (ICDRT)

Address(es): RIIC0.ICDRT 0008 8312h



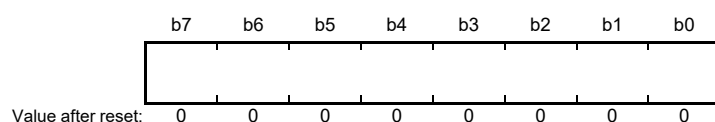
When the ICDRT register detects a space in the I²C-bus shift register (ICDRS), it transfers the transmit data that has been written to the ICDRT register to the ICDRS register and starts transmitting data in transmit mode.

The double-buffer structure of the ICDRT register and the ICDRS register allows continuous transmit operation if the next transmit data has been written to the ICDRT register while the ICDRS register data is being transmitted.

The ICDRT register can always be read and written. Write transmit data to the ICDRT register once when a transmit data empty interrupt (TXI) request is generated.

27.2.16 I²C-bus Receive Data Register (ICDRR)

Address(es): RIIC0.ICDRR 0008 8313h



When 1 byte of data has been received, the received data is transferred from the I²C-bus shift register (ICDRS) to the ICDRR register to enable the next data to be received.

The double-buffer structure of the ICDRS register and the ICDRR register allows continuous receive operation if the received data has been read from the ICDRR register while the ICDRS register is receiving data.

The ICDRR register cannot be written. Read data from the ICDRR register once when a receive data full interrupt (RXI) request is generated.

If the ICDRR register receives the next receive data before the current data is read from the ICDRR register (while the ICSR2.RDRF flag is 1), the RIIC automatically holds the SCL line low one cycle before the RDRF flag is set to 1 next.

27.2.17 I²C-bus Shift Register (ICDRS)

The ICDRS register is an 8-bit shift register to transmit and receive data.

During transmission, transmit data is transferred from the ICDRT register to the ICDRS register and is sent from the SDA0 pin. During reception, data is transferred from the ICDRS register to the ICDRR register after 1 byte of data has been received.

The ICDRS register cannot be accessed directly.

27.3 Operation

27.3.1 Communication Data Format

The I²C-bus format consists of 8-bit data and 1-bit acknowledgment. The first byte following a start condition or restart condition is an address byte used to specify a slave device with which the master device communicates. The specified slave is valid until a new slave is specified or a stop condition is generated.

Figure 27.2 shows the I²C-bus format, and Figure 27.3 shows the I²C-bus timing.

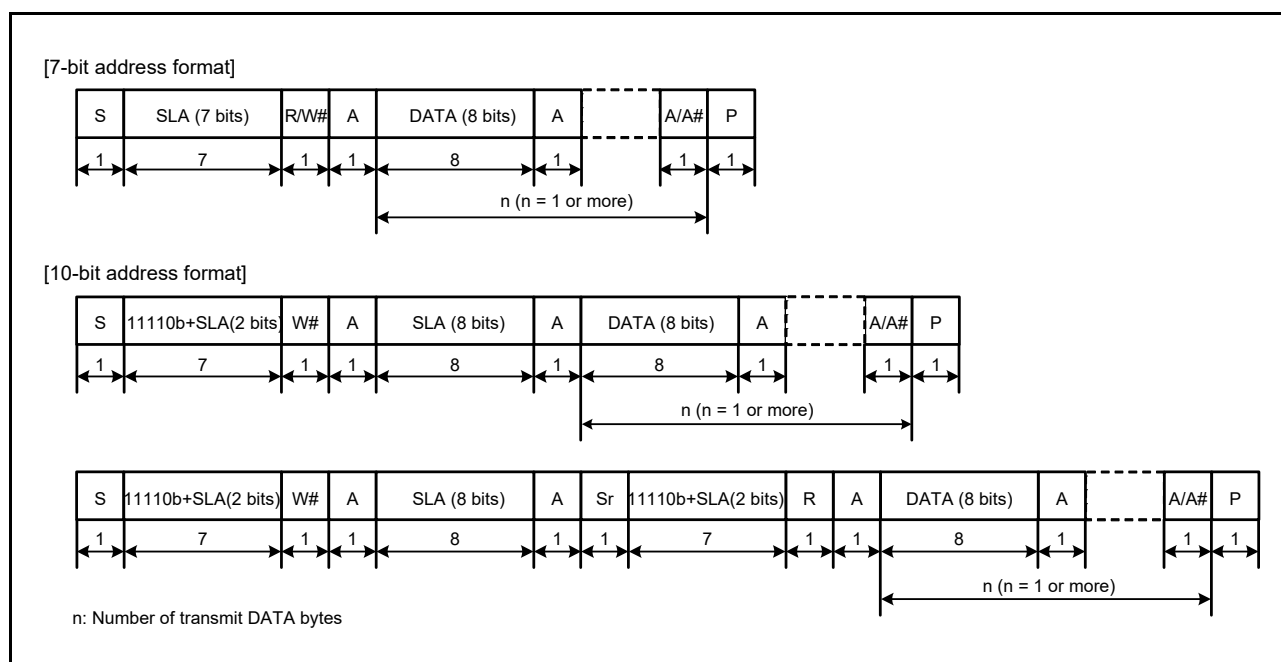


Figure 27.2 I²C-bus Format

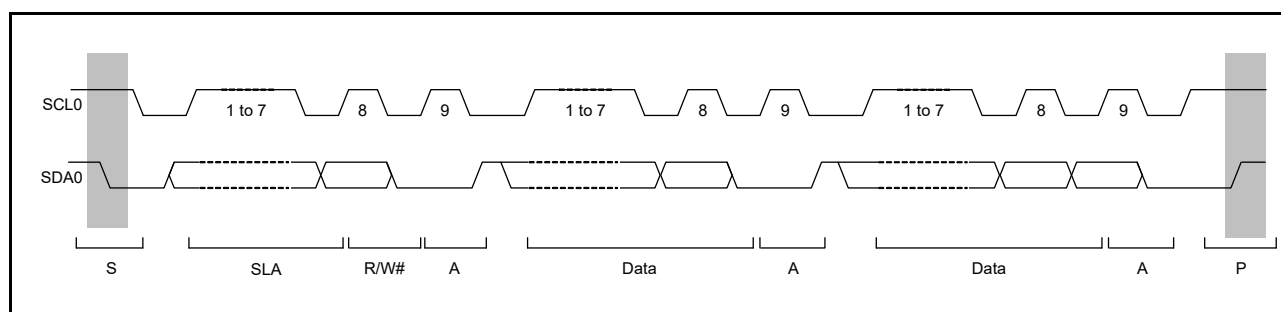


Figure 27.3 I²C-bus Timing (SLA = 7 Bits)

- S: Start condition. The master device drives the SDA0 line low from high while the SCL0 line is high.
- SLA: Slave address, by which the master device selects a slave device.
- R/W#: Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0.
- A: Acknowledge. The receiver drives the SDA0 line low. (In master transmit mode, the slave device returns acknowledge. In master receive mode, the master device returns acknowledge.)
- A#: Not Acknowledge. The receiver drives the SDA0 line high.
- Sr: Restart condition. The master device drives the SDA0 line low from high after the setup time has elapsed with the SCL0 line high.
- DATA: Transmitted or received data
- P: Stop condition. The master device drives the SDA0 line high from low while the SCL0 line is high.

27.3.2 Initial Settings

Before starting data transmission and reception, initialize the RIIC according to the procedure in Figure 27.4.

Set the ICCR1.ICE bit to 1 (internal reset) after setting the ICCR1.IICRST bit to 1 (RIIC reset) with the ICCR1.ICE bit set to 0 (SCL0 and SDA0 pins in inactive state). This initializes the various flags and internal state of the ICSR1 register. After that, set registers SARLy, SARUy, ICSEr, ICMR1, ICBRH, and ICBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see Figure 27.4). When the necessary register settings have been completed, set the ICCR1.IICRST bit to 0 (releases the RIIC reset). This step is not necessary if initialization of the RIIC has already been completed.

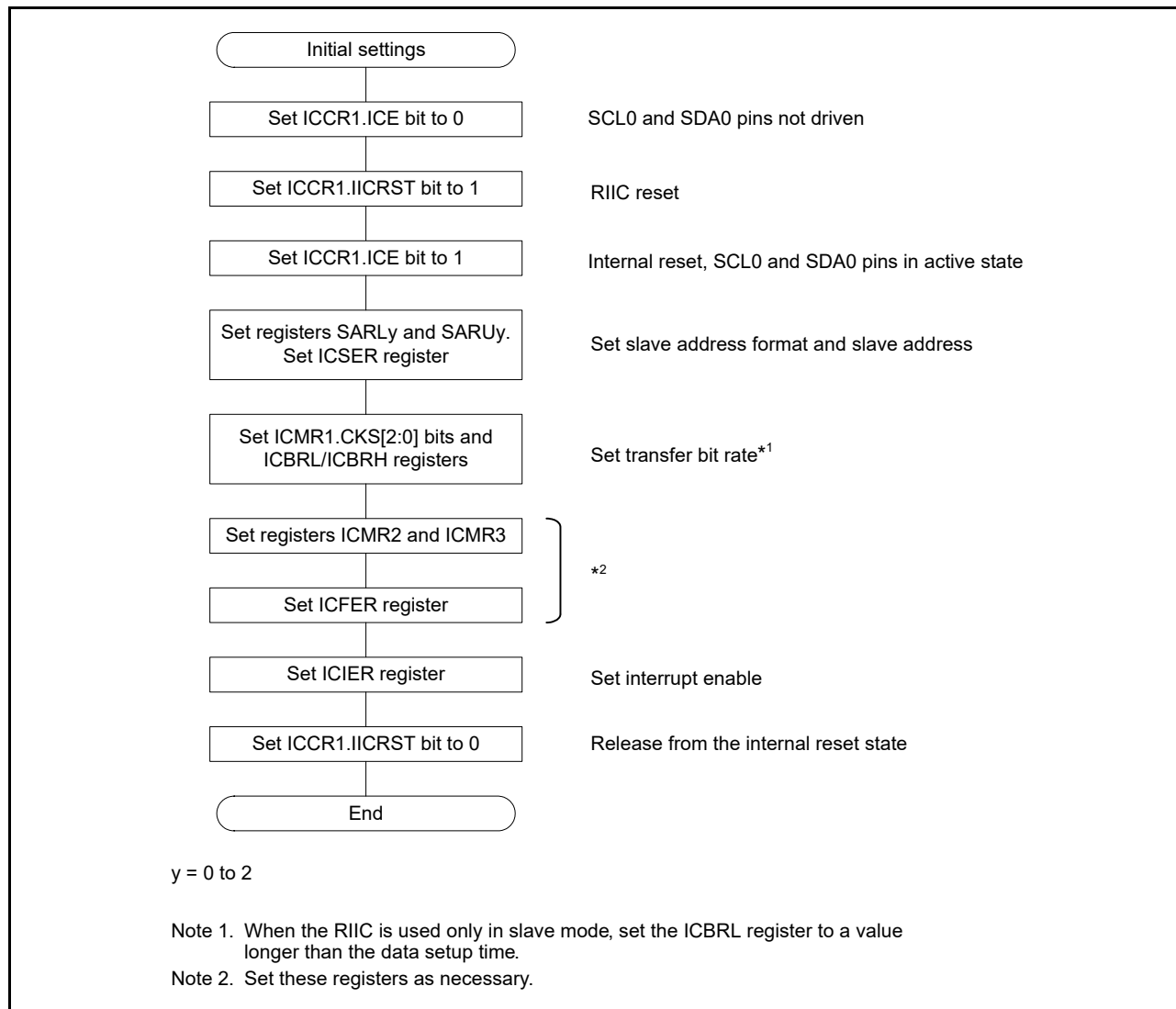


Figure 27.4 Example of RIIC Initialization Flowchart

27.3.3 Master Transmit Operation

In master transmit operation, the RIIC generates clock signals and sends data as the master device, and the slave device returns acknowledgments. Figure 27.5 shows an example of usage of master transmission and Figure 27.6 to Figure 27.8 show the timing of operations in master transmission.

The following describes the procedure and operations for master transmission.

- (1) Initial settings. For details, refer to **section 27.3.2, Initial Settings**.
- (2) Read the ICCR2.BBSY flag to check that the bus is open, and then set the ICCR2.ST bit to 1 (requests to generate a start condition). Upon receiving the request, the RIIC generates a start condition. At the same time, the BBSY flag and the ICSR2.START flag are automatically set to 1 and the ST bit is automatically set to 0. At this time, if the start condition is detected and the internal levels for the SDA output state and the levels on the SDA0 line have matched while the ST bit is 1, the RIIC recognizes that generating of the start condition as requested by the ST bit has been successfully completed, and bits MST and TRS in the ICCR2 register are automatically set to 1, placing the RIIC in master transmit mode. The ICSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.
- (3) Check that the ICSR2.TDRE flag is 1, and then write the value for transmission (the slave address and the R/W# bit) to the ICDRT register. Once the data for transmission are written to the ICDRT register, the TDRE flag is automatically set to 0, the data are transferred from the ICDRT register to the ICDRS register, and the TDRE flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRS bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, the RIIC continues in master transmit mode. Because the ICSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the ICCR2.SP bit to generate a stop condition. For data transmission with an address in the 10-bit format, start by writing 1111 0b, the 2 higher-order bits of the slave address, and W to the ICDRT register as the first address transmission. Then, as the second address transmission, write the 8 lower-order bits of the slave address to the ICDRT register.
- (4) After confirming that the ICSR2.TDRE flag is 1, write the data for transmission to the ICDRT register. The RIIC automatically holds the SCL0 line low until the data for transmission are ready or a stop condition is generated.
- (5) After all bytes of data for transmission have been written to the ICDRT register, wait until the ICSR2.NACKF or ICSR2.TEND flag becomes 1, and then set the ICCR2.SP bit to 1 (requests to generate a stop condition). Upon receiving a stop condition generation request, the RIIC generates the stop condition.
- (6) Upon detecting the stop condition, the RIIC automatically sets bits MST and TRS in the ICCR2 register to 00b and enters slave receive mode. Furthermore, it automatically sets the TDRE and TEND flags to 0, and sets the ICSR2.STOP flag to 1.
- (7) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

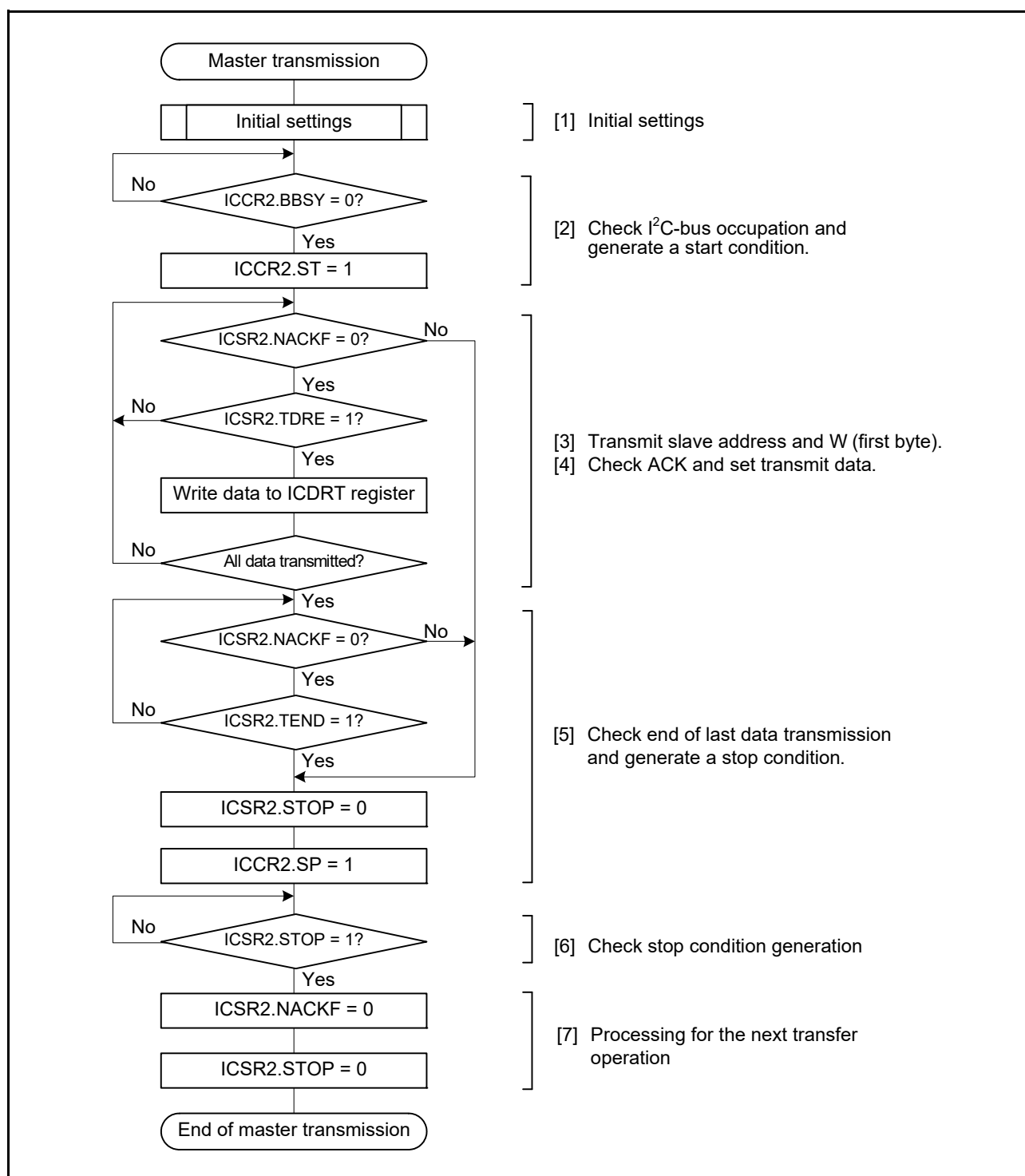


Figure 27.5 Example of Master Transmission Flowchart

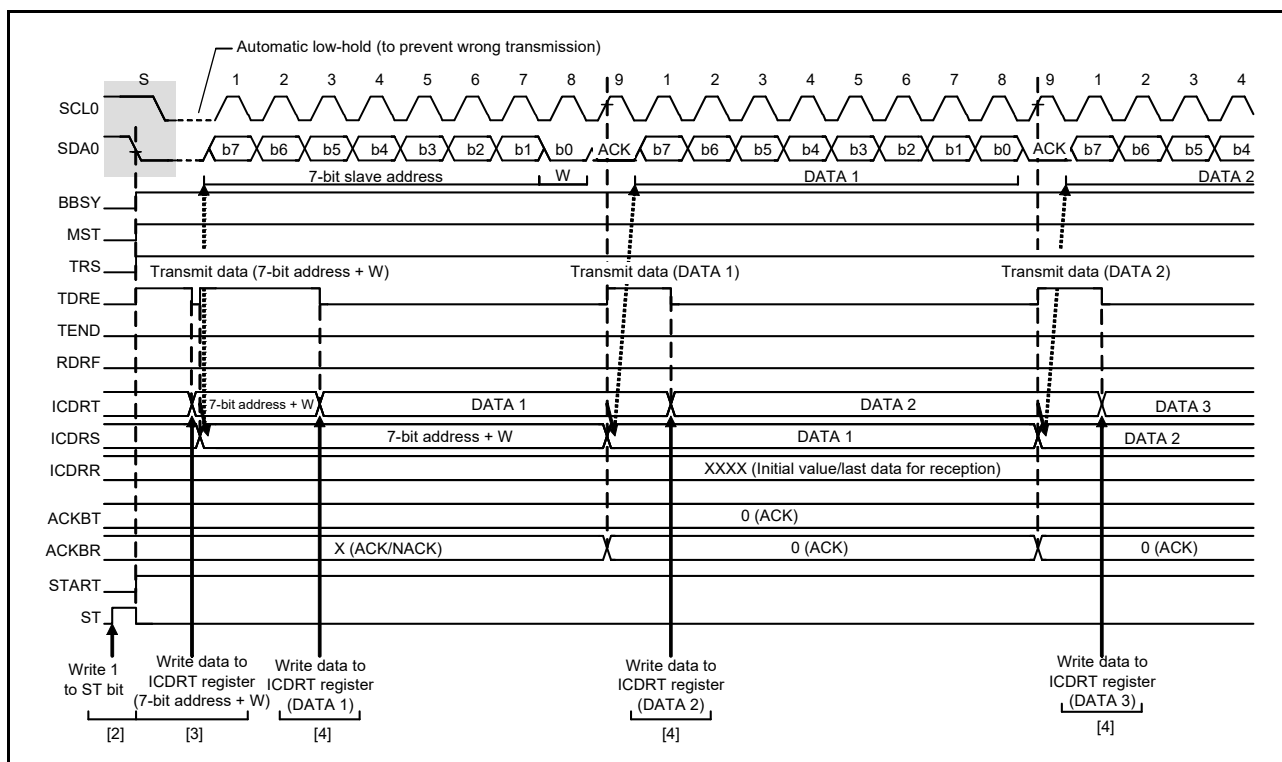


Figure 27.6 Master Transmit Operation Timing (1) (7-Bit Address Format)

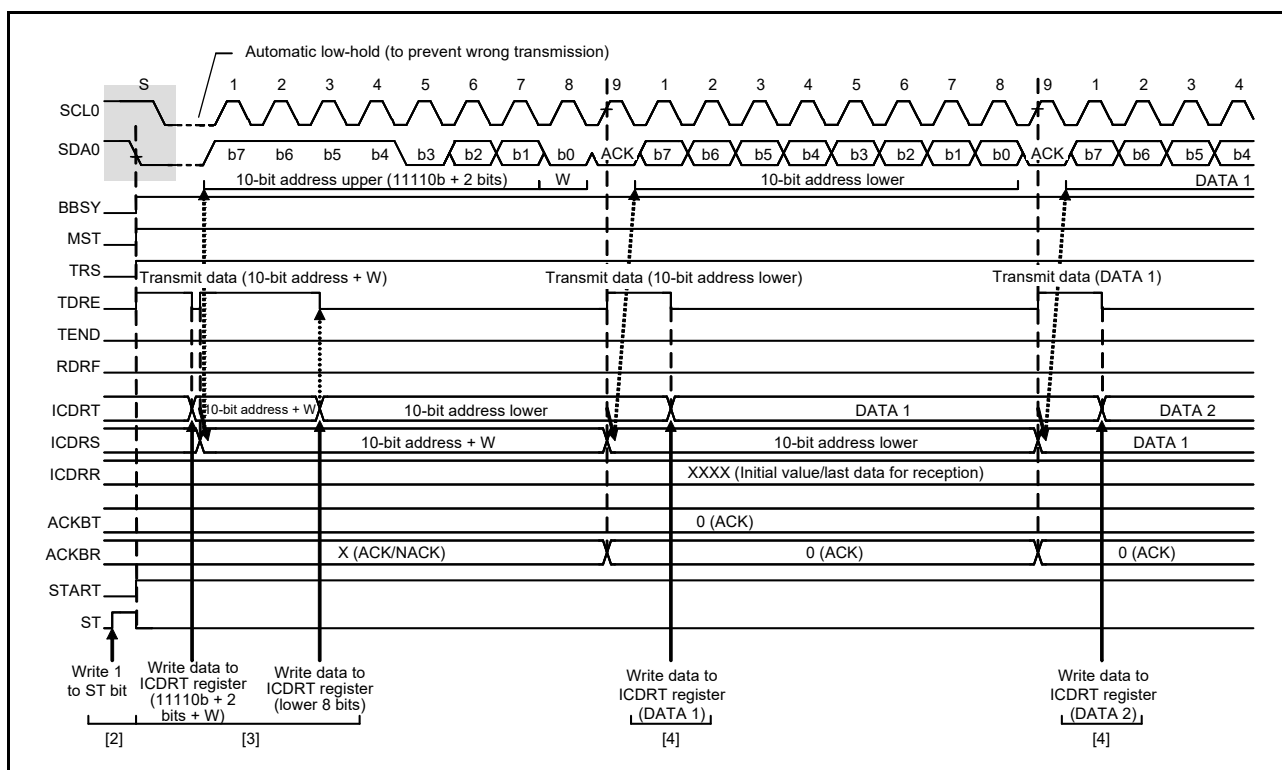


Figure 27.7 Master Transmit Operation Timing (2) (10-Bit Address Format)

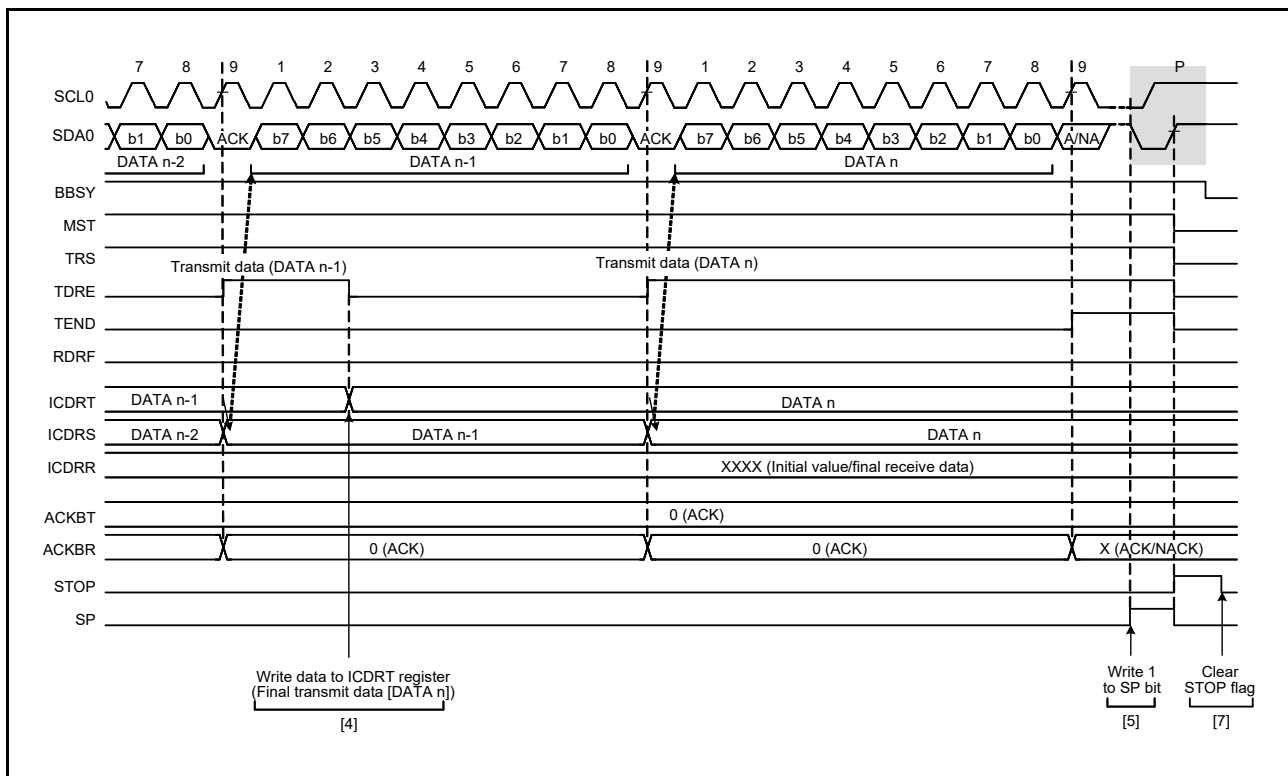


Figure 27.8 Master Transmit Operation Timing (3)

27.3.4 Master Receive Operation

In master receive operation, the RIIC as a master device generates clock signals, receives data from the slave device, and returns acknowledgments. Because the RIIC must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmit mode, but the subsequent steps are in master receive mode.

Figure 27.9 and Figure 27.10 show examples of usage of master reception (7-bit address format) and Figure 27.11 to Figure 27.13 show the timing of operations in master reception.

The following describes the procedure and operations for master reception.

- (1) Initial settings. For details, refer to section 27.3.2, Initial Settings.
- (2) Read the ICCR2.BBSY flag to check that the bus is open, and then set the ICCR2.ST bit to 1 (requests to generate a start condition). Upon receiving the request, the RIIC generates a start condition. When the RIIC detects the start condition, the BBSY flag and the ICSR2.START flag are automatically set to 1 and the ST bit is automatically set to 0. At this time, if the start condition is detected and the levels for the SDA output and the levels on the SDA0 line have matched while the ST bit is 1, the RIIC recognizes that generating of the start condition as requested by the ST bit has been successfully completed, and bits MST and TRS in the ICCR2 register are automatically set to 1, placing the RIIC in master transmit mode. The ICSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.
- (3) Check that the ICSR2.TDRE flag is 1, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to the ICDRT register. Once the data for transmission are written to the ICDRT register, the TDRE flag is automatically set to 0, the data are transferred from the ICDRT register to the ICDRS register, and the TDRE flag is again set to 1. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the ICCR2.TRS bit is automatically updated to select transmit or receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the TRS bit is set to 0 on the rising edge of the ninth SCL, placing the RIIC in master receive mode. At this time, the TDRE flag is set to 0 and the ICSR2.RDRF flag is automatically set to 1.

Because the ICSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the ICCR2.SP bit to generate a stop condition.

For master reception from a device with a 10-bit address, start by using master transmission to send the 10-bit address, and then generate a restart condition. After that, transmitting 1111 0b, the two higher-order bits of the slave address, and the R bit places the RIIC in master receive mode.

- (4) Dummy read the ICDRR register after confirming that the ICSR2.RDRF flag is 1; this makes the RIIC start output of the SCL and start data reception.
- (5) After 1 byte of data has been received, the ICSR2.RDRF flag is set to 1 on the rising edge of the eighth or ninth SCL (the clock signal) as selected by the ICMR3.RDRFS bit. Reading the ICDRR register at this time will produce the received data, and the RDRF flag is automatically set to 0 at the same time. Furthermore, the value of the acknowledgment bit received during the ninth SCL is returned as the value set in the ICMR3.ACKBT bit. Furthermore, if the next byte to be received is the next to last byte, set the ICMR3.WAIT bit to 1 (for wait insertion) before reading the ICDRR register (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the ICMR3.ACKBT bit to 1 (NACK) in step (6), due to other interrupts, etc., this fixes the SCL0 line to low on the falling edge of the ninth clock pulse in reception of the last byte, so the state is such that generating a stop condition is possible.
- (6) When the ICMR3.RDRFS bit is 0 and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the ICMR3.ACKBT bit to 1 (NACK).
- (7) After reading the byte before last from the ICDRR register, if the value of the ICSR2.RDRF flag is confirmed to be 1, write 1 to the ICCR2.SP bit (requests to generate a stop condition) and then read the last byte from the ICDRR register. When the ICDRR register is read, the RIIC is released from the wait state and generates the stop condition after low-level output in the ninth clock pulse is completed or the SCL0 line is released from the low-hold state.
- (8) Upon detecting the stop condition, the RIIC automatically sets bits MST and TRS in the ICCR2 register to 00b and enters slave receive mode. Furthermore, detection of the stop condition leads to setting of the ICSR2.STOP flag to 1.
- (9) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

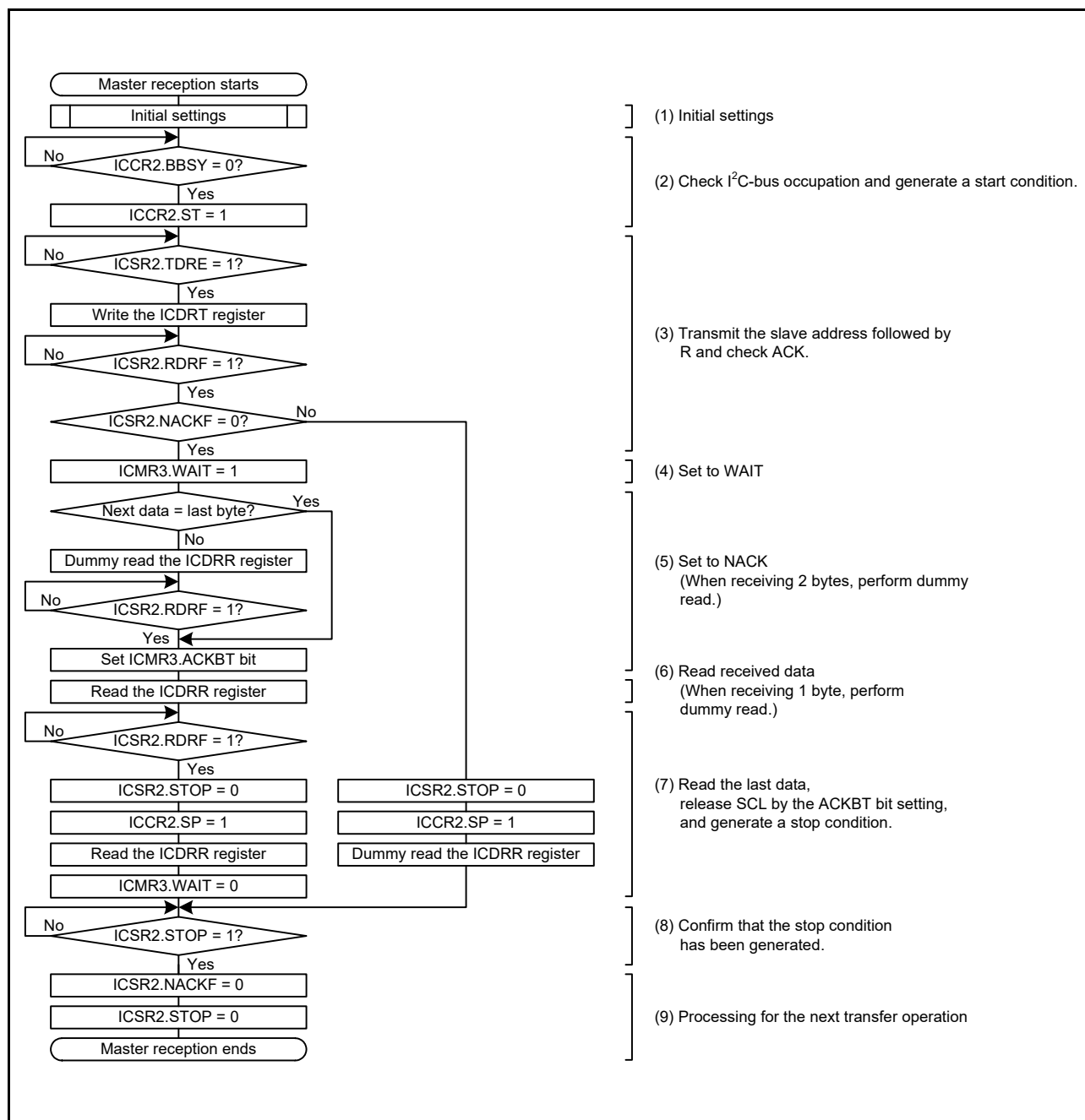


Figure 27.9 Example of Master Reception (7-Bit Address Format, 1 or 2 bytes)

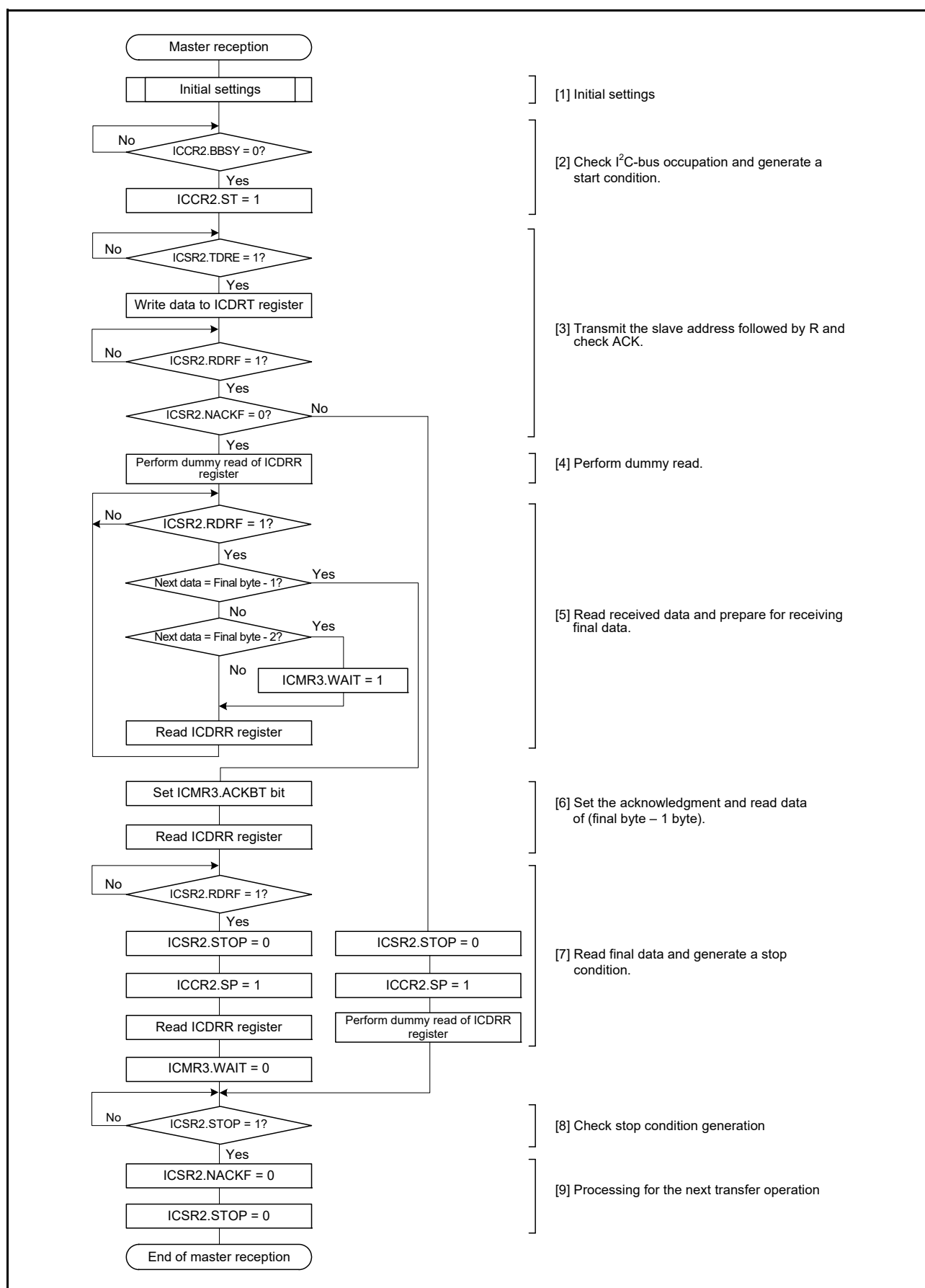


Figure 27.10 Example of Master Reception (7-Bit Address Format, 3 Bytes or More)

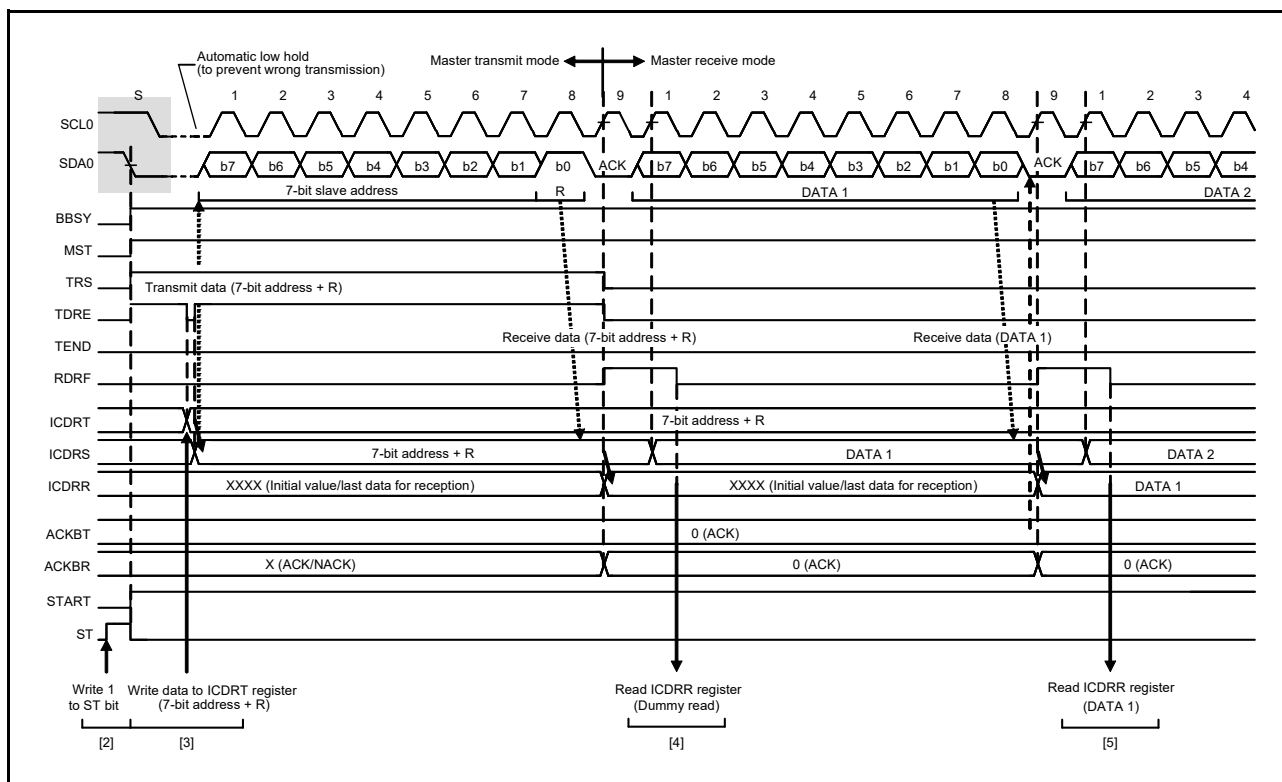


Figure 27.11 Master Receive Operation Timing (1) (7-Bit Address Format, When RDRFS bit is 0)

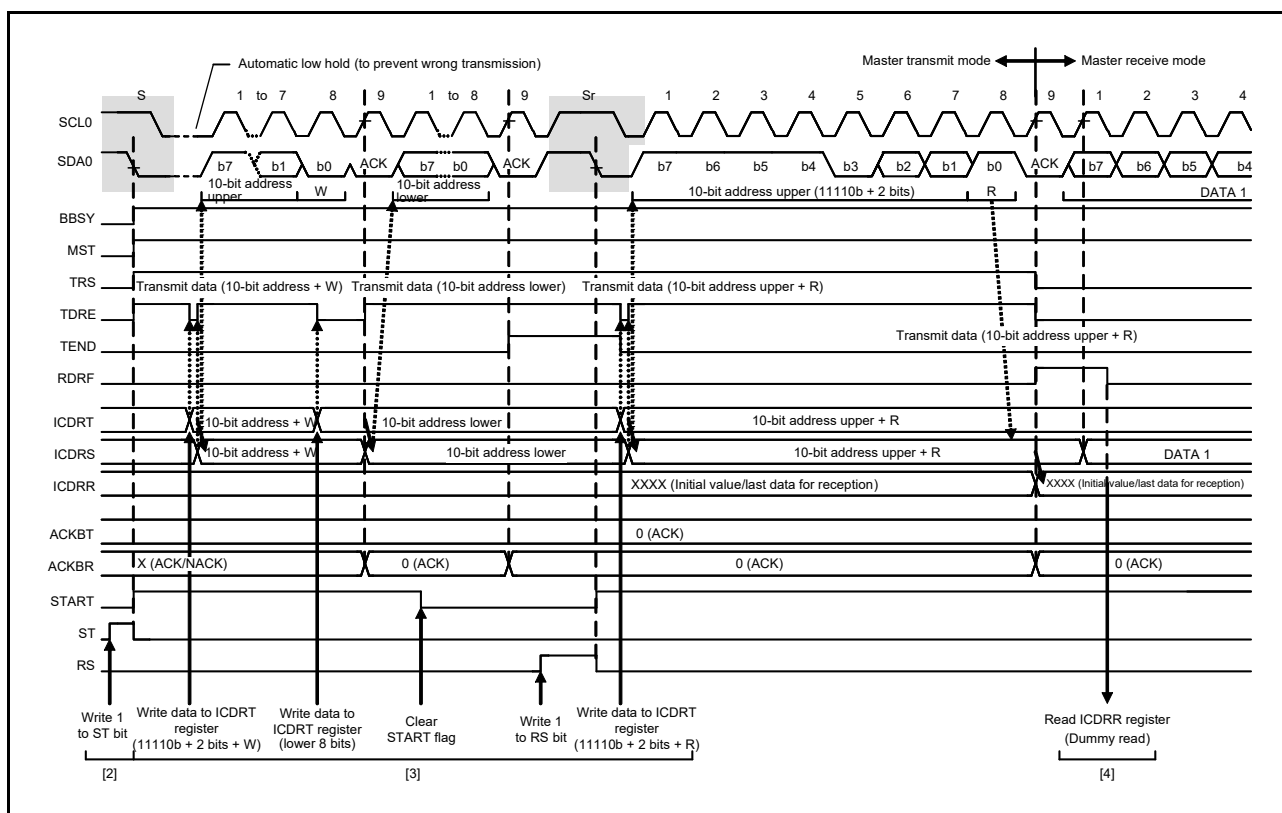


Figure 27.12 Master Receive Operation Timing (2) (10-Bit Address Format, When RDRFS bit is 0)

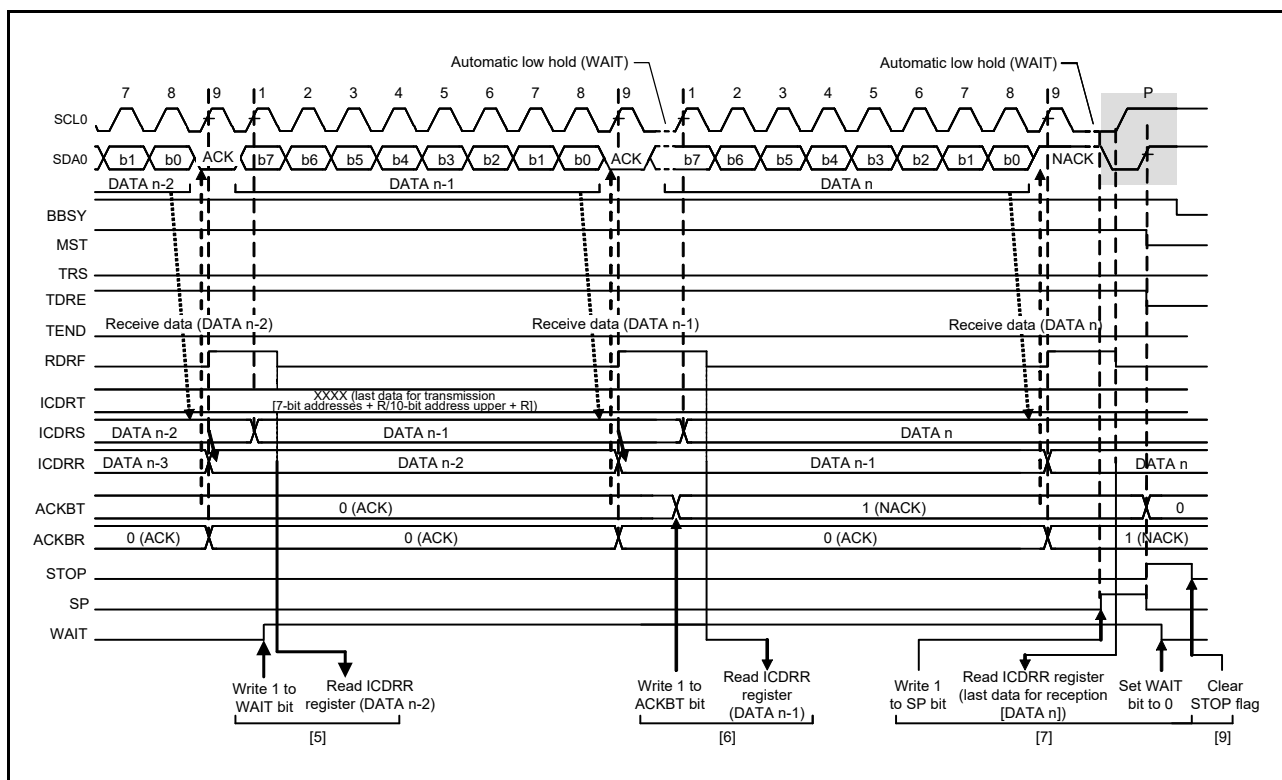


Figure 27.13 Master Receive Operation Timing (3) (When RDRFS bit is 0)

27.3.5 Slave Transmit Operation

In slave transmit operation, the master device outputs the SCL, the RIIC transmits data as a slave device, and the master device returns acknowledgments.

Figure 27.14 shows an example of usage of slave transmission and Figure 27.15 and Figure 27.16 show the timing of operations in slave transmission.

The following describes the procedure and operations for slave transmission.

- (1) Initial settings. For details, refer to **section 27.3.2, Initial Settings**.
After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.
- (2) After receiving a matching slave address, the RIIC sets one of the corresponding bits ICSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth SCL (the clock signal) and outputs the value set in the ICMR3.ACKBT bit to the acknowledgment bit on the ninth SCL. If the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmit mode by setting both the ICCR2.TRS bit and the ICSR2.TDRE flag to 1.
- (3) After the ICSR2.TDRE flag is confirmed to be 1, write the data for transmission to the ICDRT register. At this time, if the RIIC does not receive ACK from the master device (receives an NACK signal) while the ICFER.NACKF bit is 1, the RIIC suspends transfer of the next data.
- (4) Wait until the ICSR2.TEND flag is set to 1 while the ICSR2.TDRE flag is 1, after the ICSR2.NACKF flag is set to 1 or the last byte for transmission is written to the ICDRT register. When the ICSR2.NACKF flag or the TEND flag is 1, the RIIC drives the SCL0 line low on the falling edge of ninth SCL.
- (5) When the ICSR2.NACKF flag or the ICSR2.TEND flag is 1, dummy read the ICDRR register to complete the processing. This releases the SCL0 line.
- (6) Upon detecting the stop condition, the RIIC automatically sets bits ICSR1.HOA, GCA, and AASy (y = 0 to 2), flags ICSR2.TDRE and TEND, and the ICCR2.TRS bit to 0, and enters slave receive mode.
- (7) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

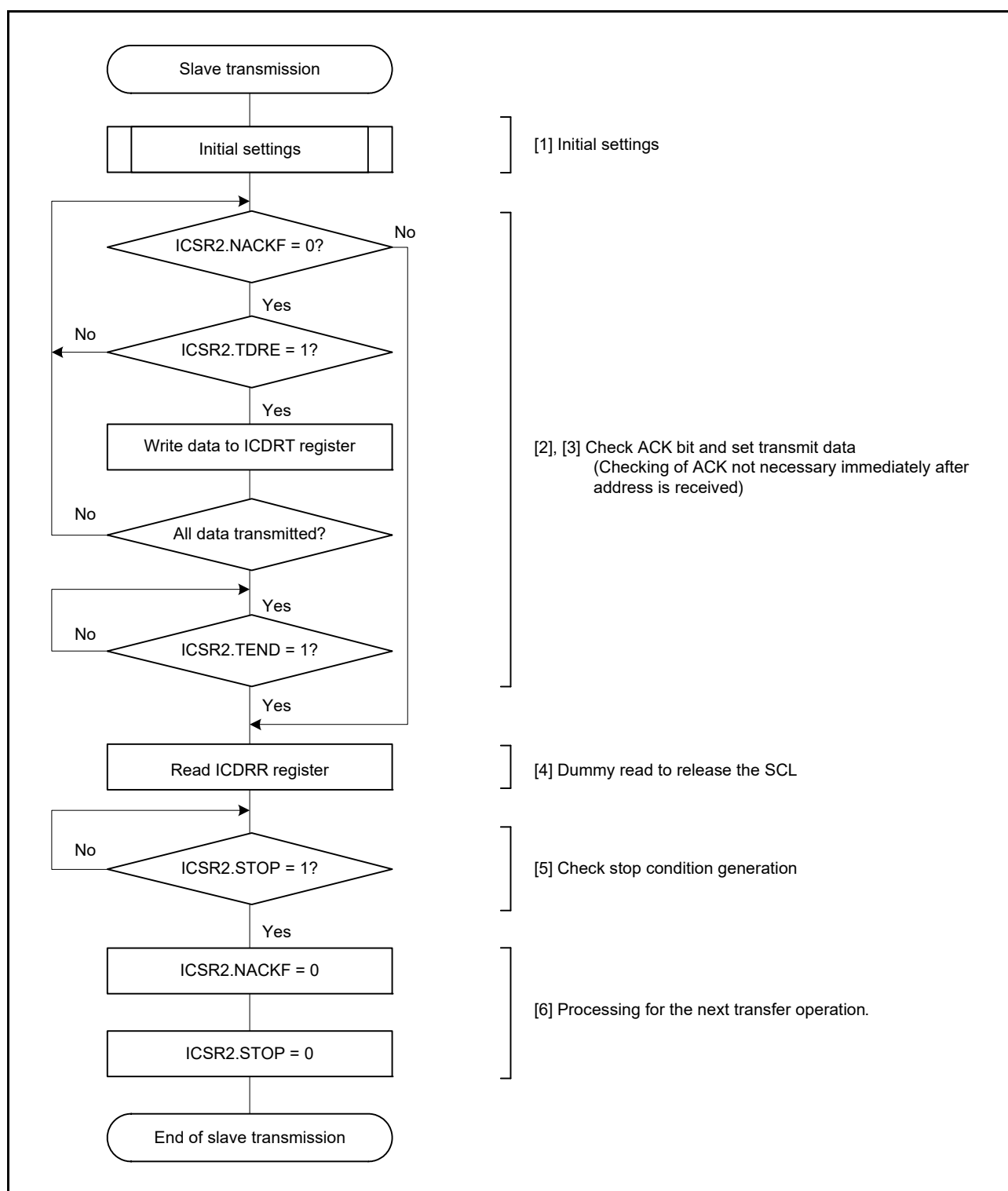


Figure 27.14 Example of Slave Transmission Flowchart

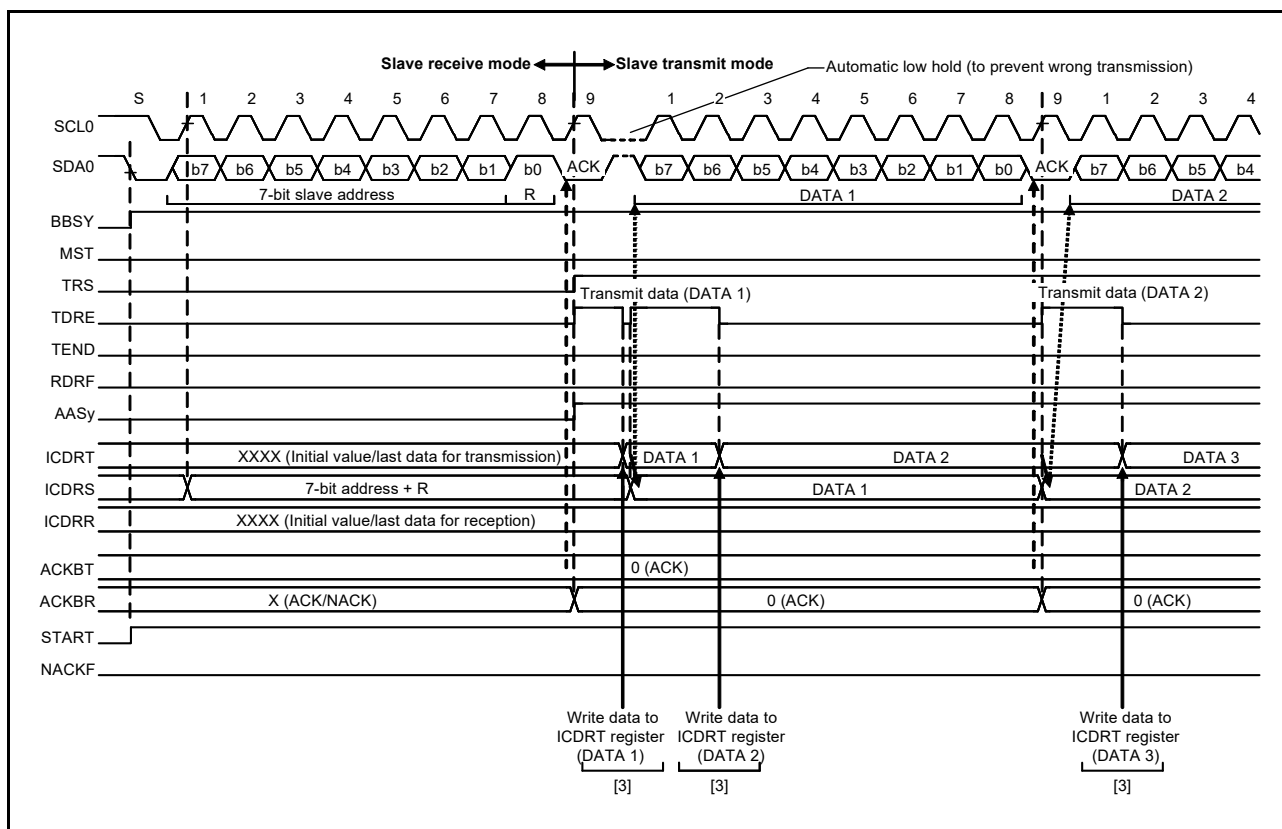


Figure 27.15 Slave Transmit Operation Timing (1) (7-Bit Address Format)

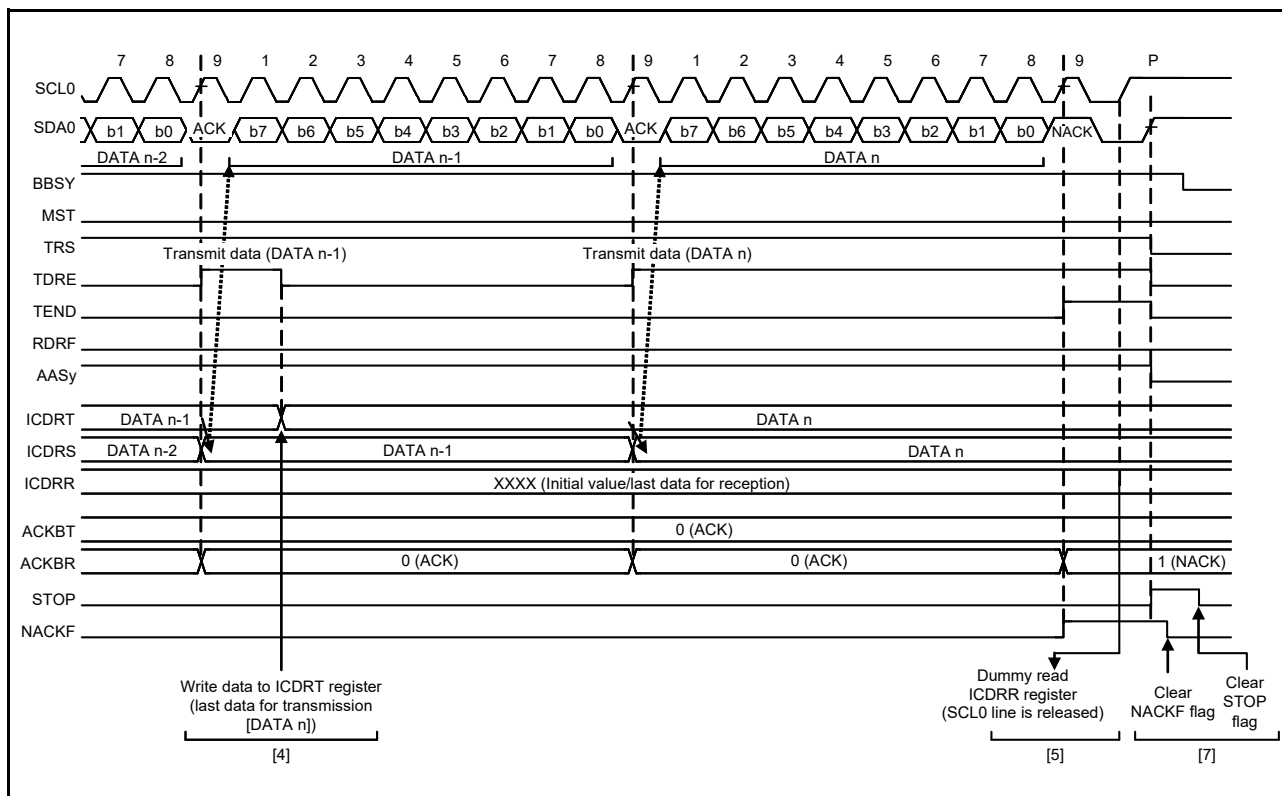


Figure 27.16 Slave Transmit Operation Timing (2)

27.3.6 Slave Receive Operation

In slave receive operation, the master device outputs the SCL and transmit data, and the RIIC returns acknowledgments as a slave device.

Figure 27.17 shows an example of usage of slave reception and Figure 27.18 and Figure 27.19 show the timing of operations in slave reception.

The following describes the procedure and operations for slave reception.

- (1) Initial settings. For details, refer to section 27.3.2, Initial Settings.
After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.
- (2) After receiving a matching slave address, the RIIC sets one of the corresponding bits ICSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth SCL (the clock signal) and outputs the value set in the ICMR3.ACKBT bit to the acknowledgment bit on the ninth SCL. If the value of the R/W# bit that was also received at this time is 0, the RIIC continues to place itself in slave receive mode and sets the ICSR2.RDRF flag to 1.
- (3) After the ICSR2.STOP flag is confirmed to be 0 and the ICSR2.RDRF flag to be 1, dummy read the ICDRR register (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower 8 bits when the 10-bit address format is selected).
- (4) When the ICDRR register is read, the RIIC automatically sets the ICSR2.RDRF flag to 0. If reading of the ICDRR register is delayed and a next byte is received while the RDRF flag is still set to 1, the RIIC holds the SCL0 line low from one SCL cycle before the timing with which RDRF should be set. In this case, reading the ICDRR register releases the SCL0 line from being held low.
When the ICSR2.STOP flag is 1 and the ICSR2.RDRF flag is also 1 or when all the data is completely received, read the ICDRR register.
- (5) Upon detecting the stop condition, the RIIC automatically clears bits ICSR1.HOA, GCA, and AASy (y = 0 to 2) to 0.
- (6) After checking that the ICSR2.STOP flag is 1, set the ICSR2.STOP flag to 0 for the next transfer operation.

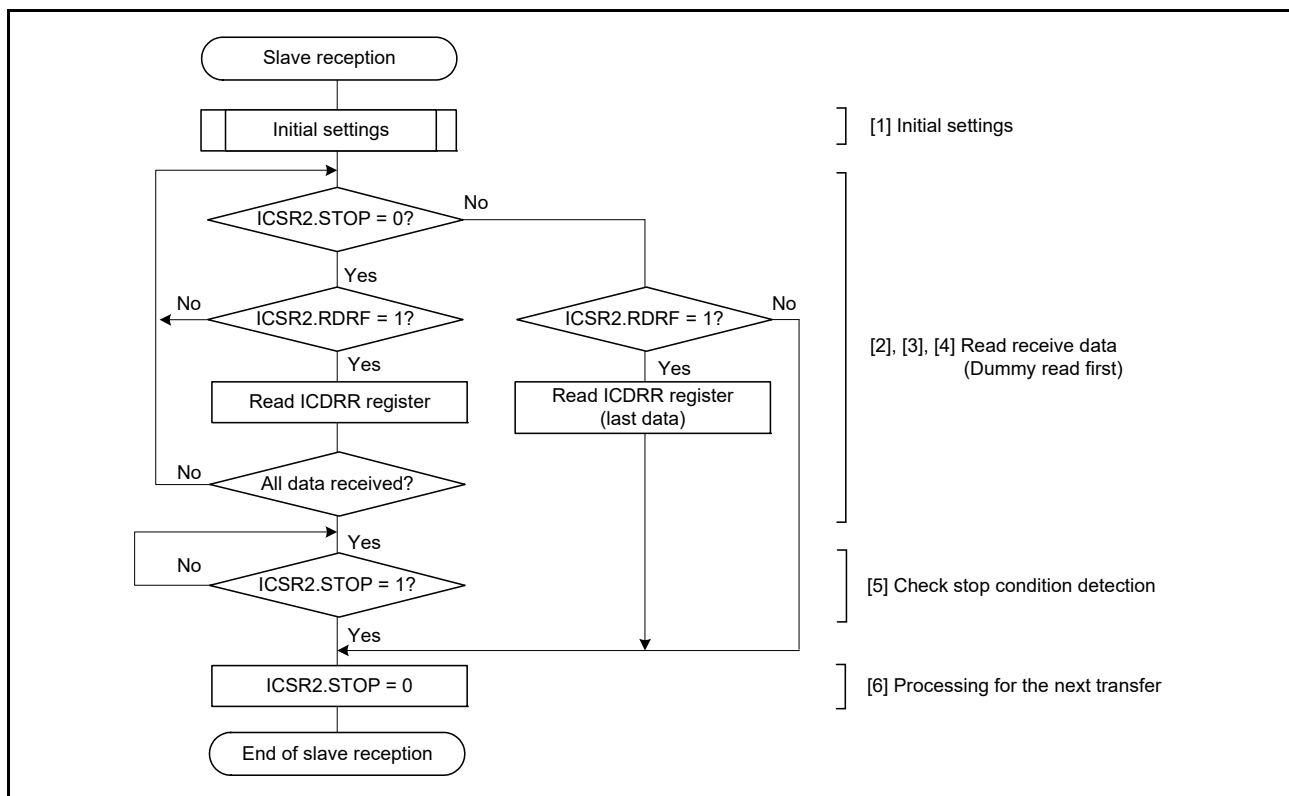


Figure 27.17 Example of Slave Reception Flowchart

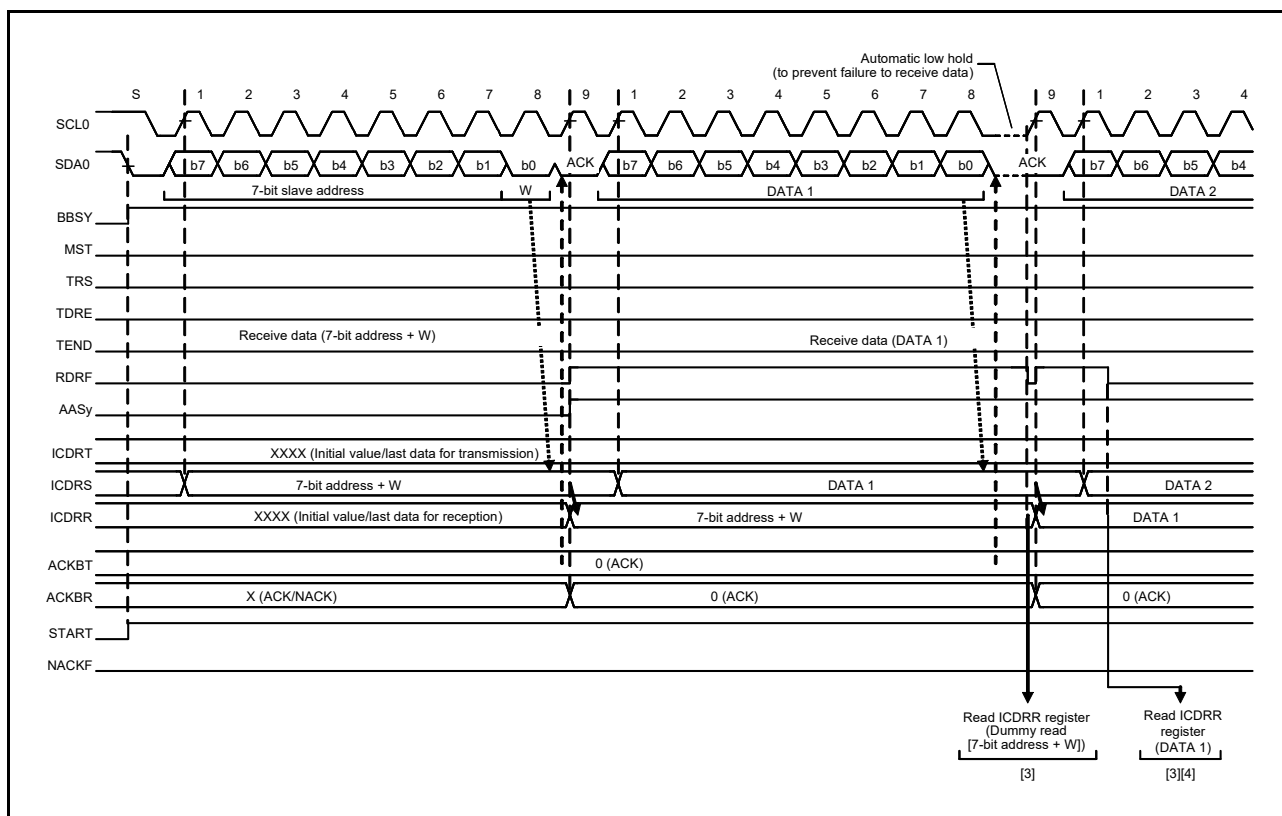


Figure 27.18 Slave Receive Operation Timing (1) (7-Bit Address Format, when RDRFS bit is 0)

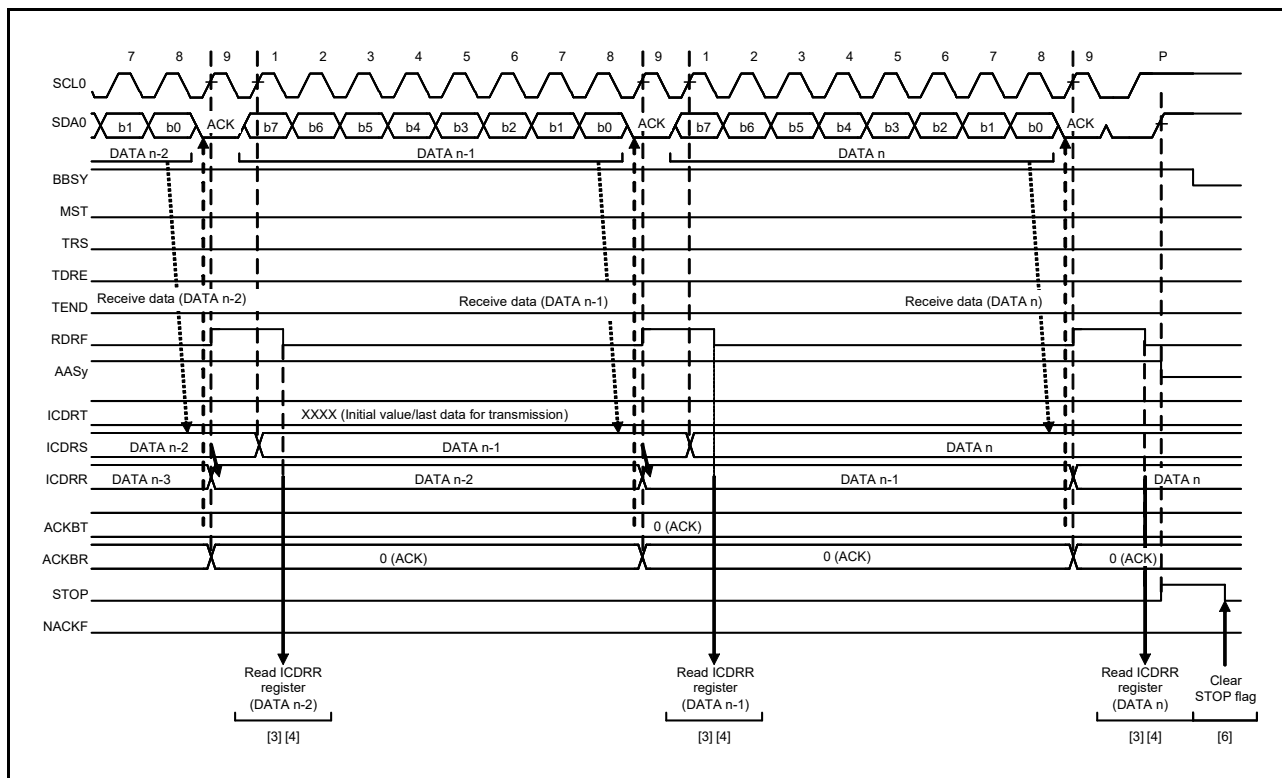


Figure 27.19 Slave Receive Operation Timing (2) (when RDRFS bit is 0)

27.4 SCL Synchronization Circuit

In generation of the SCL, the RIIC starts counting out the value for width at high level specified in the ICBRH register when it detects a rising edge on the SCL0 line and drives the SCL0 line low once counting of the width at high level is complete. When the RIIC detects the falling edge of the SCL0 line, it starts counting out the width at low period specified in the ICBRL register, and then stops driving the SCL0 line (releases the line) once counting of the width at low level is complete. The SCL is thus generated.

If multiple master devices are connected to the I²C-bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since synchronization of the SCL signals must be handled bit by bit, the RIIC is equipped with a facility (the SCL synchronization circuit) for obtaining bit-by-bit synchronization of the SCL signals by monitoring the SCL0 line while in master mode.

When the RIIC has detected a rising edge on the SCL0 line and thus started counting out the width at high level specified in the ICBRH register, and the level on the SCL0 line falls because an SCL signal is being generated by another master device, the RIIC stops counting when it detects the falling edge, drives the level on the SCL0 line low, and starts counting out the width at low level specified in the ICBRL register. When the RIIC finishes counting out the width at low level, it stops driving the SCL0 line low (i.e. releases the line). At this time, if the width at low level of the SCL signal from the other master device is longer than the width at low level set in the RIIC, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCL0 line has been released. When the RIIC finishes outputting the low period of the SCL, the SCL0 line is released and the SCL rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the ICFER.SCLE bit is set to 1.

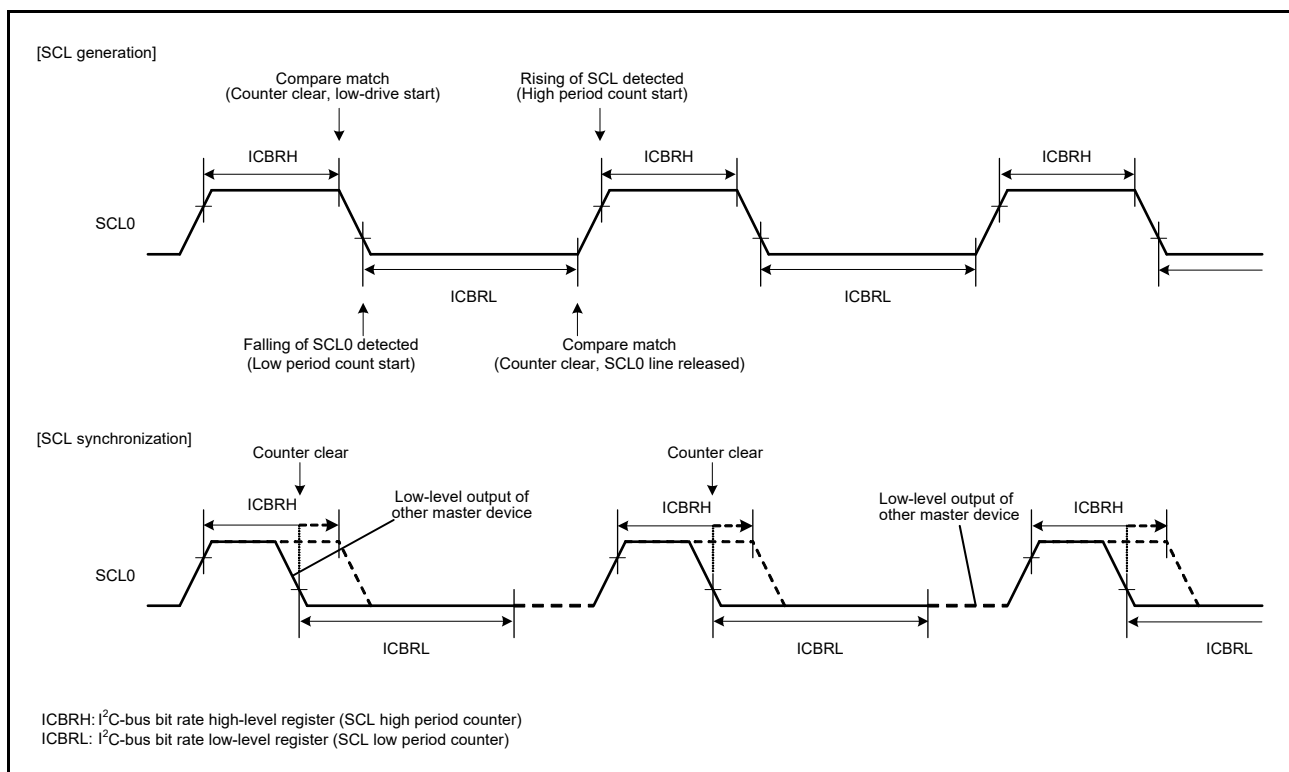


Figure 27.20 Generation and Synchronization of the SCL Signal from the RIIC

27.5 SDA Output Delay Function

The RIIC module incorporates a function for delaying output on the SDA line. The delay can be applied to all output (generation of the start, restart, and stop conditions, data, and the ACK and NACK signals) on the SDA line.

The SDA output delay function is used to delay the SDA output timing relative to falling edges of SCL to ensure that the SDA signal changes while the SCL is low and can be used to prevent erroneous operation of communications devices.

This function is also used to satisfy the 300 ns (min.) data hold time prescribed by the SMBus specification.

The output delay function is enabled by setting the ICMR2.SDDL[2:0] bits to any value other than 000b, and disabled by setting the same bits to 000b.

When the SDA output delay function is enabled (the SDDL[2:0] bits are not “000b”), the SDA output delay counter counts the number of cycles set in the SDDL[2:0] bits of the count source selected by the ICMR2.DLCS bit (the internal reference clock (IIC ϕ) or internal reference clock divided by 2 (IIC ϕ /2)). On completion of counting of cycles of delay, the RIIC changes the bit being output as the SDA signal (generation of the start, restart, or stop condition, a new bit, or an ACK or NACK signal).

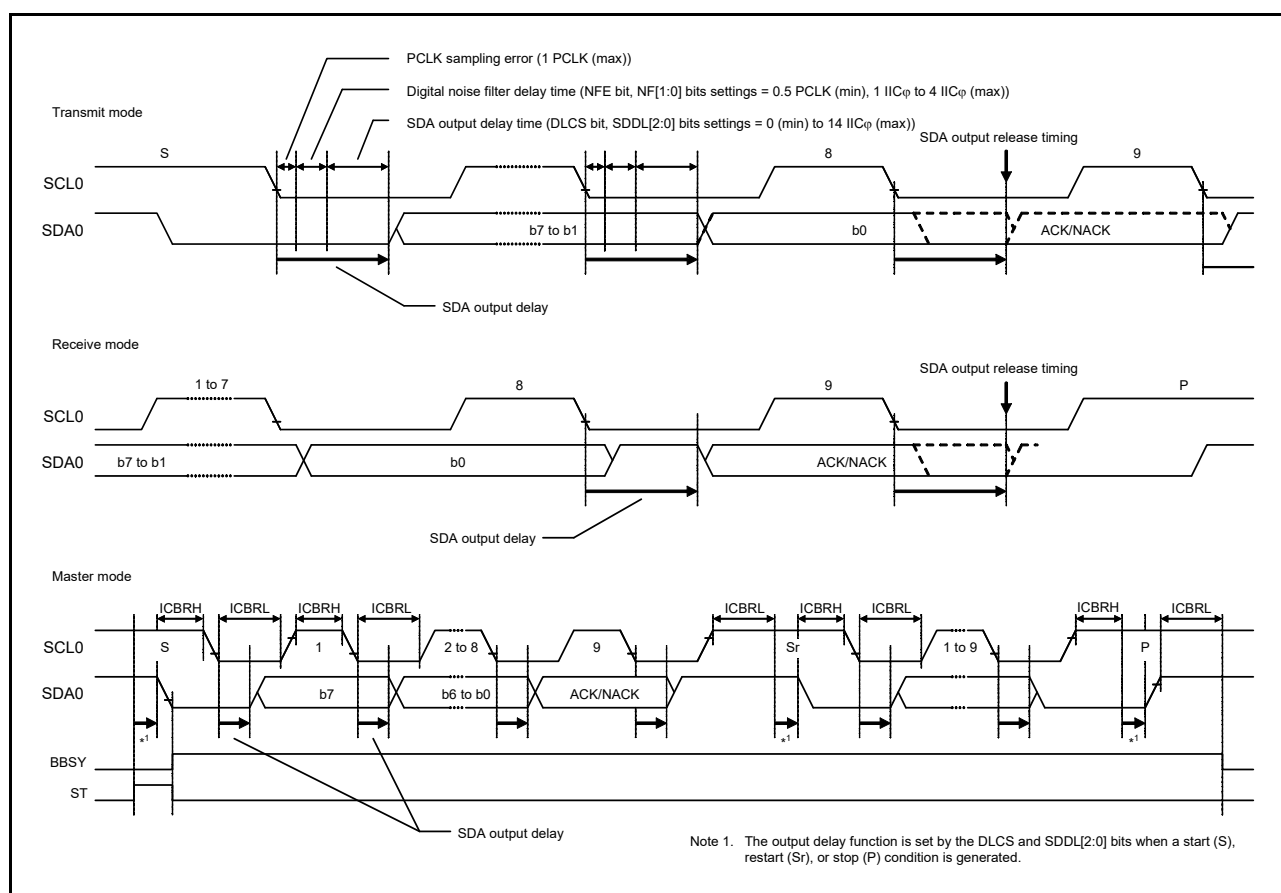


Figure 27.21 SDA Output Delay Function

27.6 Digital Noise Filters

The states of the SCL0 and SDA0 pins are conveyed to the internal circuitry through digital noise filters. Figure 27.22 is a block diagram of the digital noise filter.

The on-chip digital noise filter of each RIIC consists of four flip-flop circuit stages connected in series and a match detection circuit.

The number of effective stages in the digital noise filter is selected by the ICMR3.NF[1:0] bits. The selected number of effective stages determines the noise-filtering capability as a period from one to four cycles of IIC ϕ .

The input signal to the SCL0 pin (or SDA0 pin) is sampled on falling edges of the IIC ϕ signal. When the input signal level matches the output level for the number of effective flip-flop circuit stages selected by the ICMR3.NF[1:0] bits, the signal level is conveyed to the subsequent stages. If the signal levels do not match, the previous value is retained.

If the ratio between the frequency of the internal operating clock (PCLK) and the transfer rate is relatively small (e.g. data transfer at 400 kbps with PCLK = 4 MHz), the characteristics of the digital noise filter may lead to the elimination of needed signals as noise.

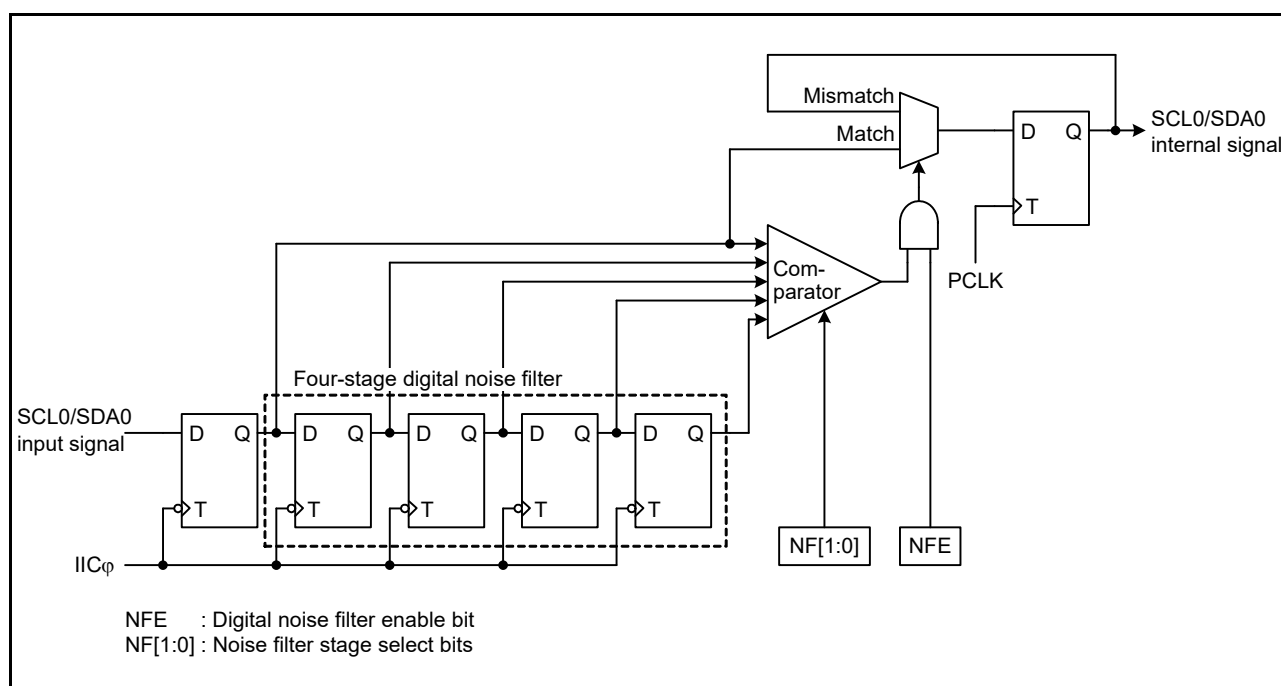


Figure 27.22 Block Diagram of the Digital Noise Filter

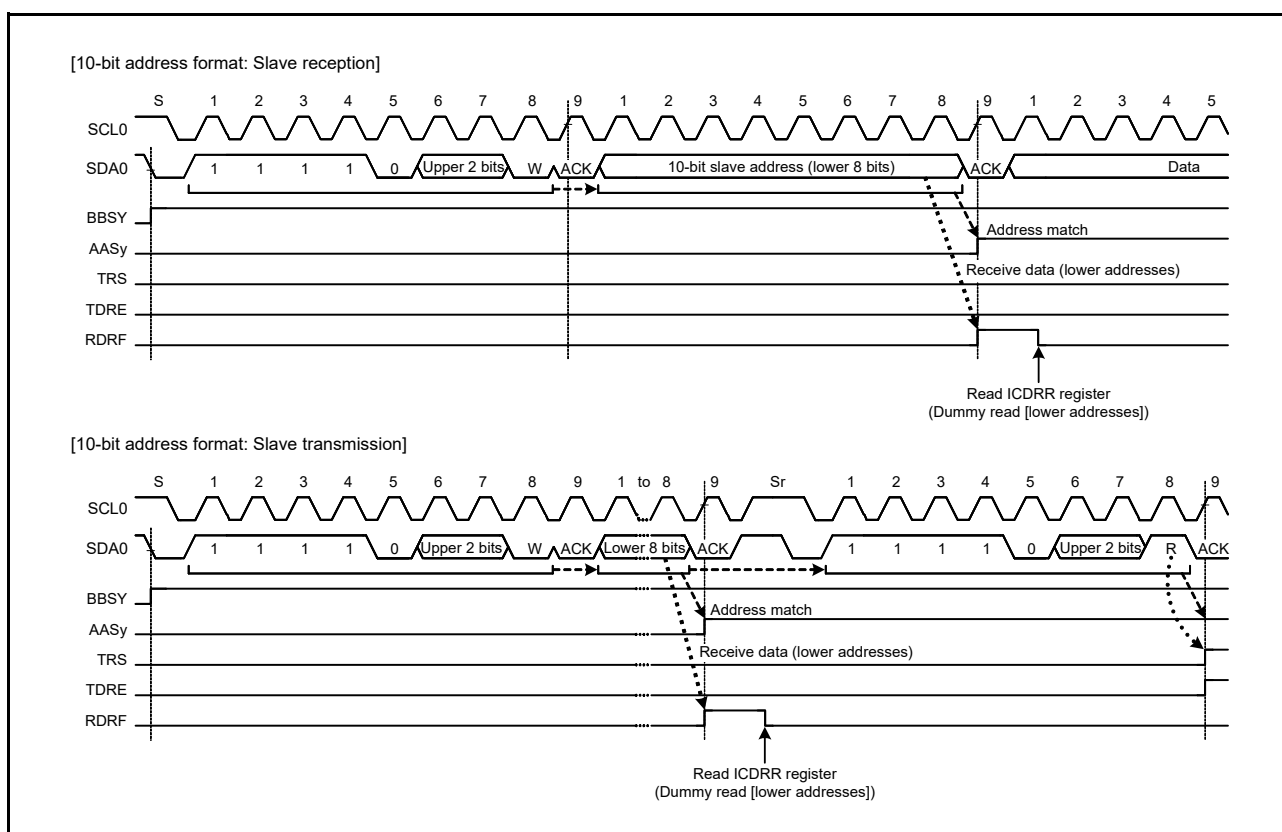


Figure 27.24 AASy Flag Set Timing with 10-Bit Address Format Selected

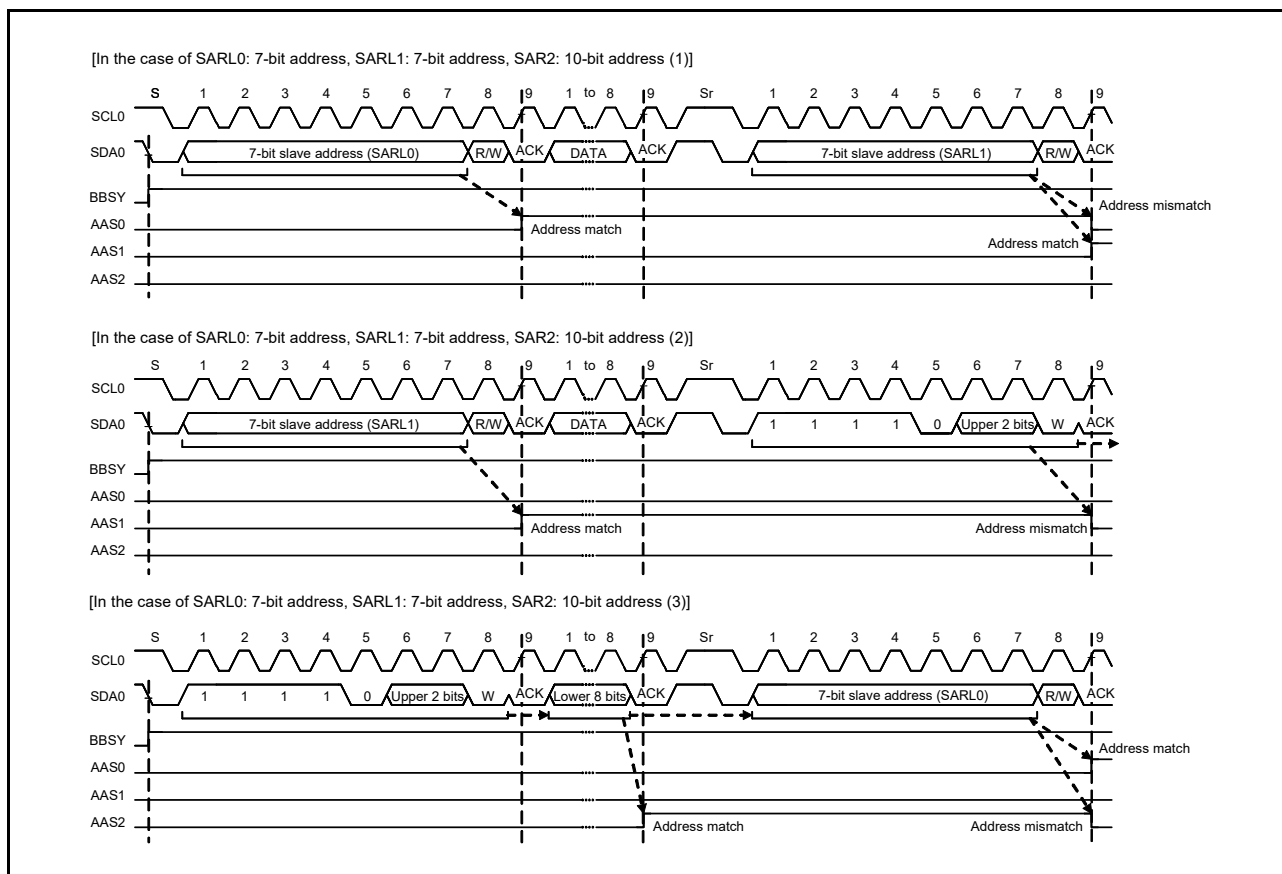


Figure 27.25 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed

27.7.2 Detection of the General Call Address

The RIIC also has a facility for detecting the general call address (0000 000b + 0 (write)). This is enabled by setting the ICSER.GCAE bit to 1.

If the address following a start or restart condition is 0000 000b + 1 (read) (start byte), the RIIC recognizes this as the address of a slave device with an “all-zero” address but not as the general call address.

When the RIIC detects the general call address, both the ICSR1.GCA flag and the ICSR2.RDRF flag are set to 1 on the rising edge of the ninth SCL. This leads to the generation of a receive data full interrupt (RXI). The value of the GCA flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

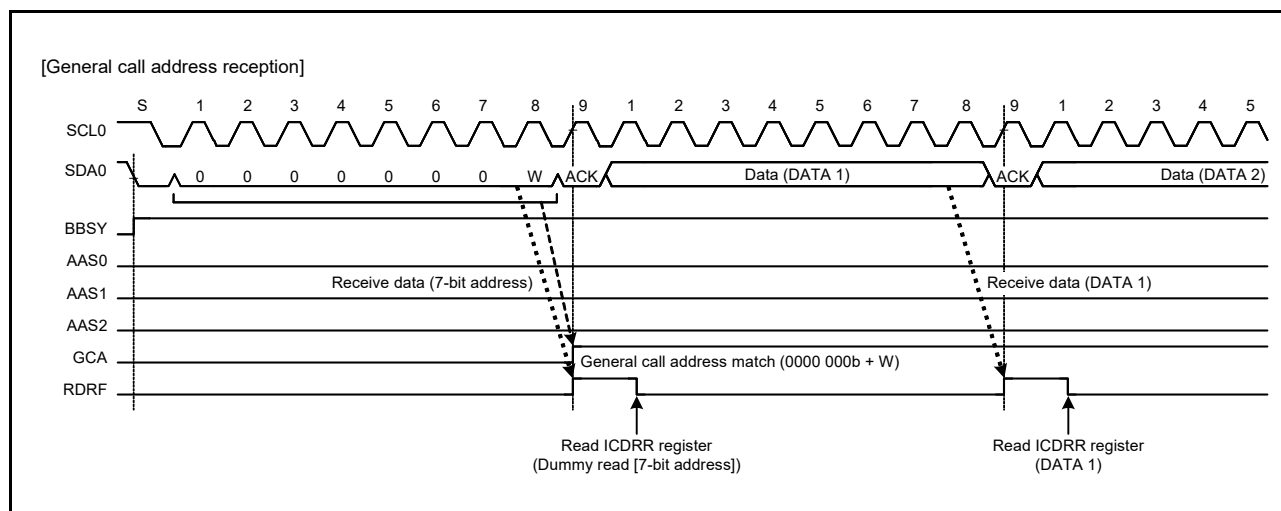


Figure 27.26 Timing of GCA Flag Setting during Reception of General Call Address

27.7.3 Device-ID Address Detection

The RIIC module has a function to detect device-ID addresses complying with the I²C-bus specification. When the RIIC receives 1111 100b as the first seven bits of the first byte following a start condition or a restart condition while the ICSER.DIDE bit is 1, the RIIC recognizes the address as a device-ID address, sets the ICSR1.DID flag to 1 on the rising edge of the ninth SCL when the following R/W# bit is 0, and then compares the second and following bytes with its own slave address. If the received address matches the value in the slave address register, the RIIC sets the corresponding ICSR1.AASy flag (y = 0 to 2) to 1.

After that, when the first byte received after a start or restart condition is generated matches the device ID address (1111 100b) again and the following R/W# bit is 1, the RIIC does not compare the second and subsequent bytes and sets the ICSR2.TDRE flag to 1.

In the device-ID address detection function, the RIIC sets the DID flag to 0 if a match with the RIIC's own slave address is not obtained or a match with the device ID address is not obtained after a match with the RIIC's own slave address and the detection of a restart condition. If the first byte after detection of a start or restart condition matches the device ID address (1111 100b) and the R/W# bit is 0, the RIIC sets the DID flag to 1 and compares the second and subsequent bytes with the RIIC's slave address. If the R/W# bit is 1, the DID flag holds the previous value and the RIIC does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DID flag after confirming that TDRE flag is 1.

Furthermore, prepare the device-ID fields (3 bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details of the information that must be included in device-ID fields, contact NXP Semiconductors.

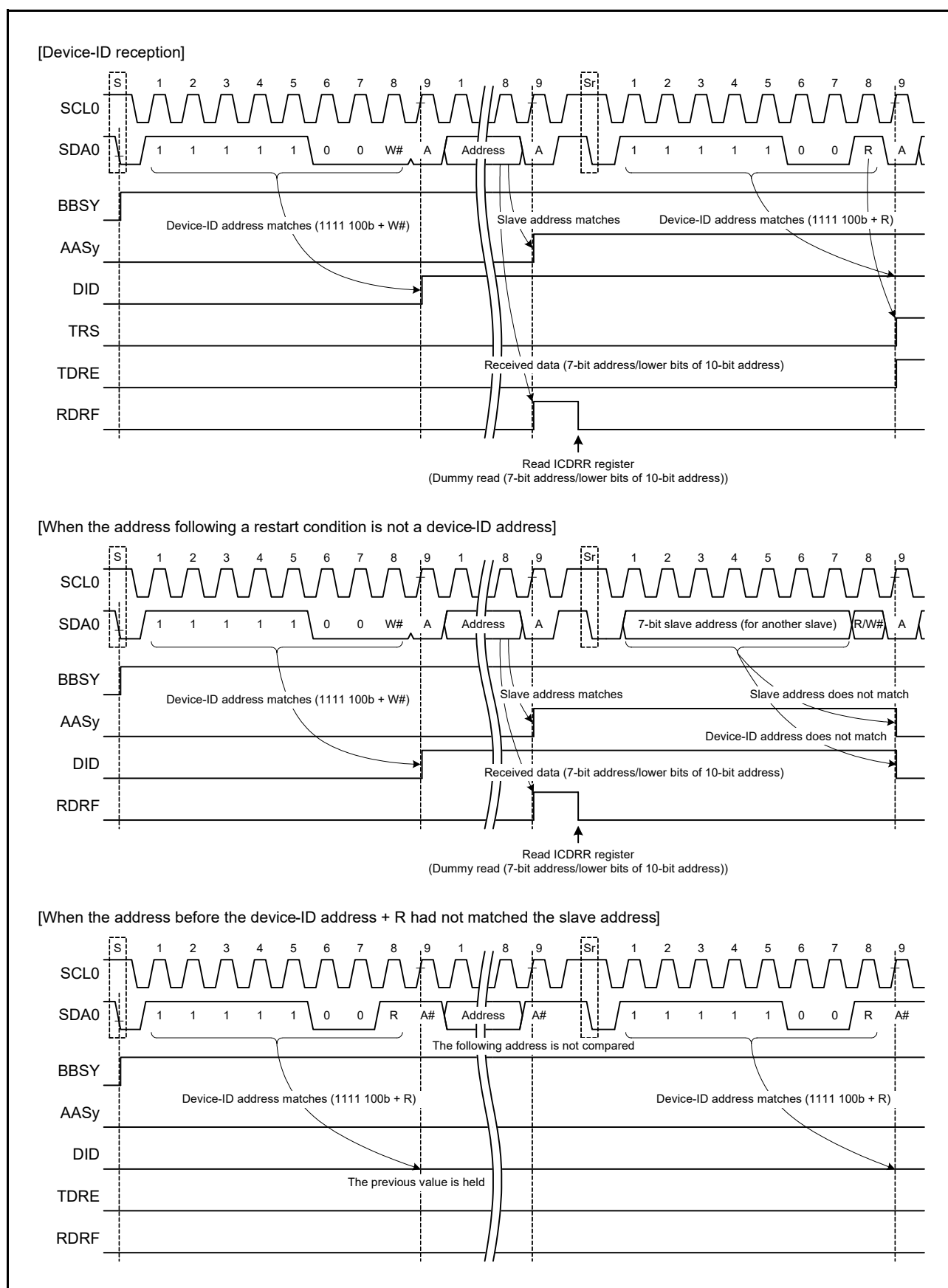


Figure 27.27 Set/Clear Timing of the AASy and DID Flags during Reception of Device-ID Address

27.7.4 Host Address Detection

The RIIC has a function to detect the host address while the SMBus is operating. When the ICSER.HOAE bit is set to 1 while the ICMR3.SMBS bit is 1, the RIIC can detect the host address (0001 000b) in slave receive mode (bits MST and TRS in the ICCR2 register are 00b).

When the RIIC detects the host address, the ICSR1.HOA flag is set to 1 at the rising edge of the ninth SCL, and at the same time, the ICSR2.RDRF flag is set to 1 when the R/W# bit is 0 (Wr bit). This causes a receive data full interrupt (RXI) to be generated. The HOA flag is used to recognize that the host address was sent from the smart battery or other devices.

If the bit following the host address (0001 000b) is an Rd bit (R/W# bit is 1), the RIIC can also detect the host address. After the host address is detected, the RIIC operates in the same manner as normal slave operation.

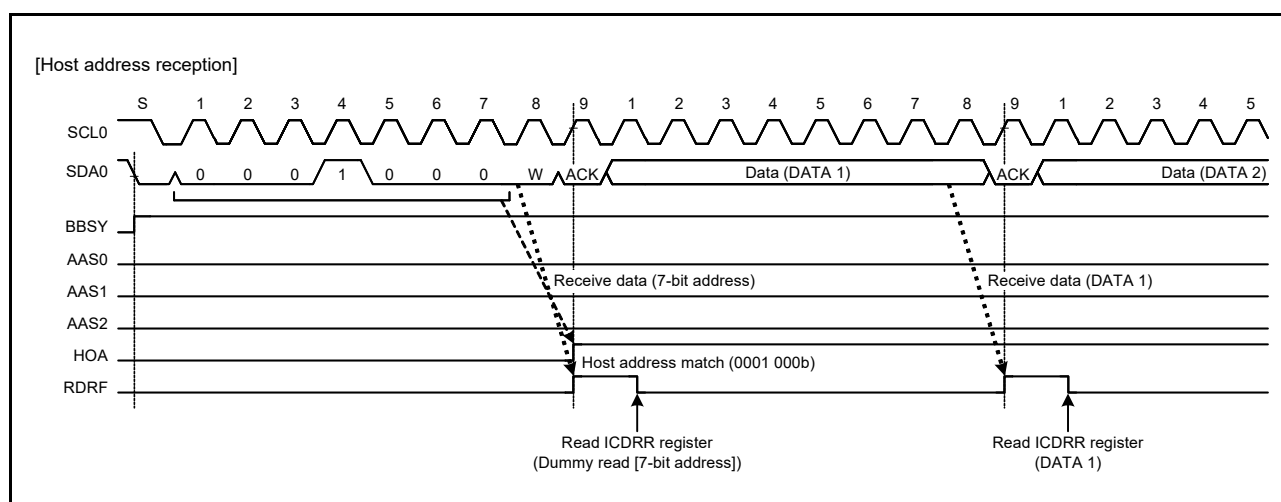


Figure 27.28 HOA Flag Set Timing during Reception of Host Address

27.8 Automatic Low-Hold Function for SCL

27.8.1 Function to Prevent Wrong Transmission of Transmit Data

If the shift register (ICDRS) is empty when data have not been written to the I²C-bus transmit data register (ICDRT) with the RIIC in transmission mode (ICCR2.TRS bit is 1), the SCL0 line is automatically held low over the intervals shown below. This low period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

Master transmit mode

- Low period after a start condition or restart condition is generated
- Low period between the ninth clock pulse of one transfer and the first clock pulse of the next

Slave transmit mode

- Low period between the ninth clock pulse of one transfer and the first clock pulse of the next

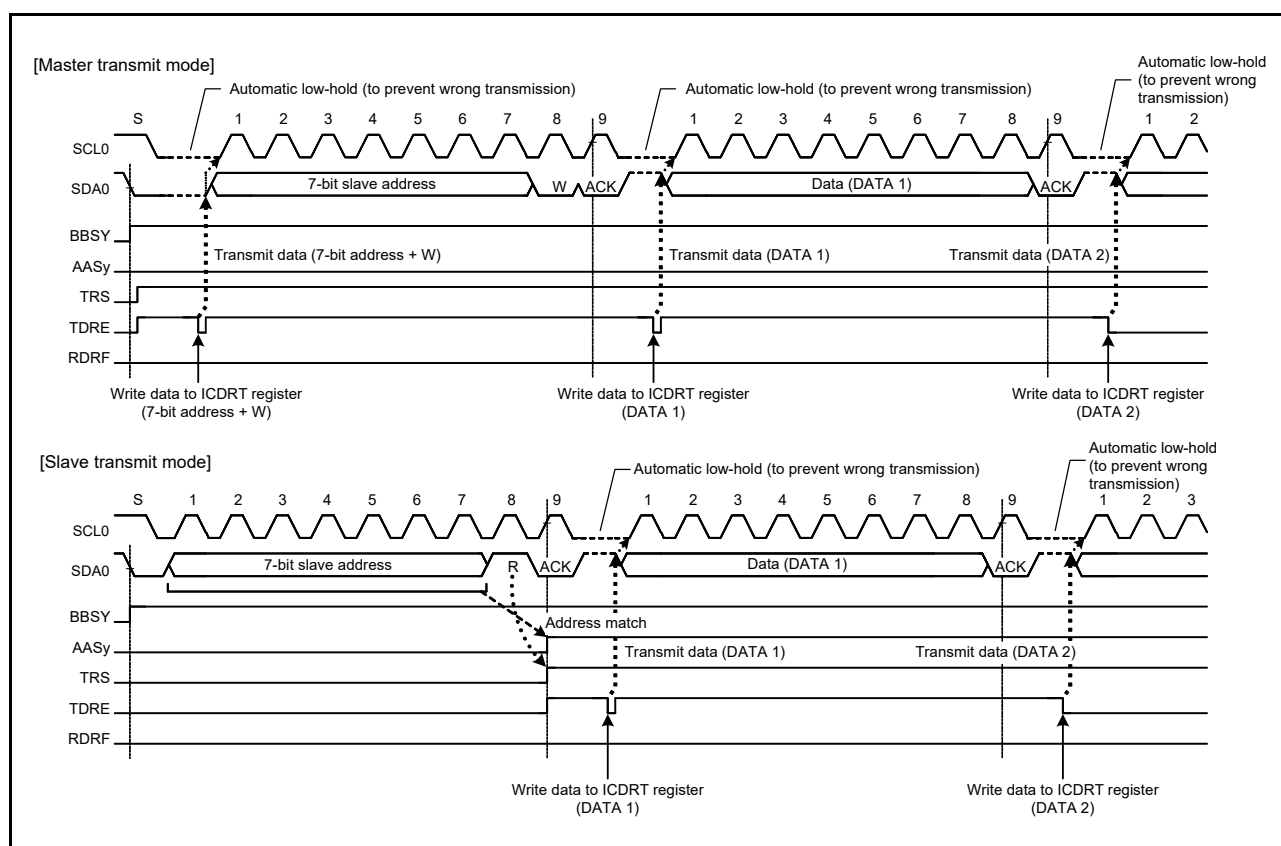


Figure 27.29 Automatic Low-Hold Operation in Transmit Mode

27.8.2 NACK Reception Transfer Suspension Function

The RIIC has a function to suspend transfer operation when NACK is received in transmit mode (ICCR2.TRS bit is 1). This function is enabled when the ICFER.NACKE bit is set to 1 (transfer suspension enabled). If the next transmit data has already been written (ICSR2.TDRE flag is 0) when NACK is received, next data transmission at the falling edge of the ninth SCL is automatically suspended. This prevents the SDA0 line output level from being held low when the MSB of the next transmit data is 0.

If the data transmission is suspended (ICSR2.NACKF flag is 1) by this function, the following data transmission and data reception are not started. To resume data transfer, set the NACKF flag to 0. In master transmit mode, restart data transfer by setting the NACKF flag to 0 after generating a restart condition, or restart data transfer from a start condition after generating a stop condition then setting the NACKF flag to 0.

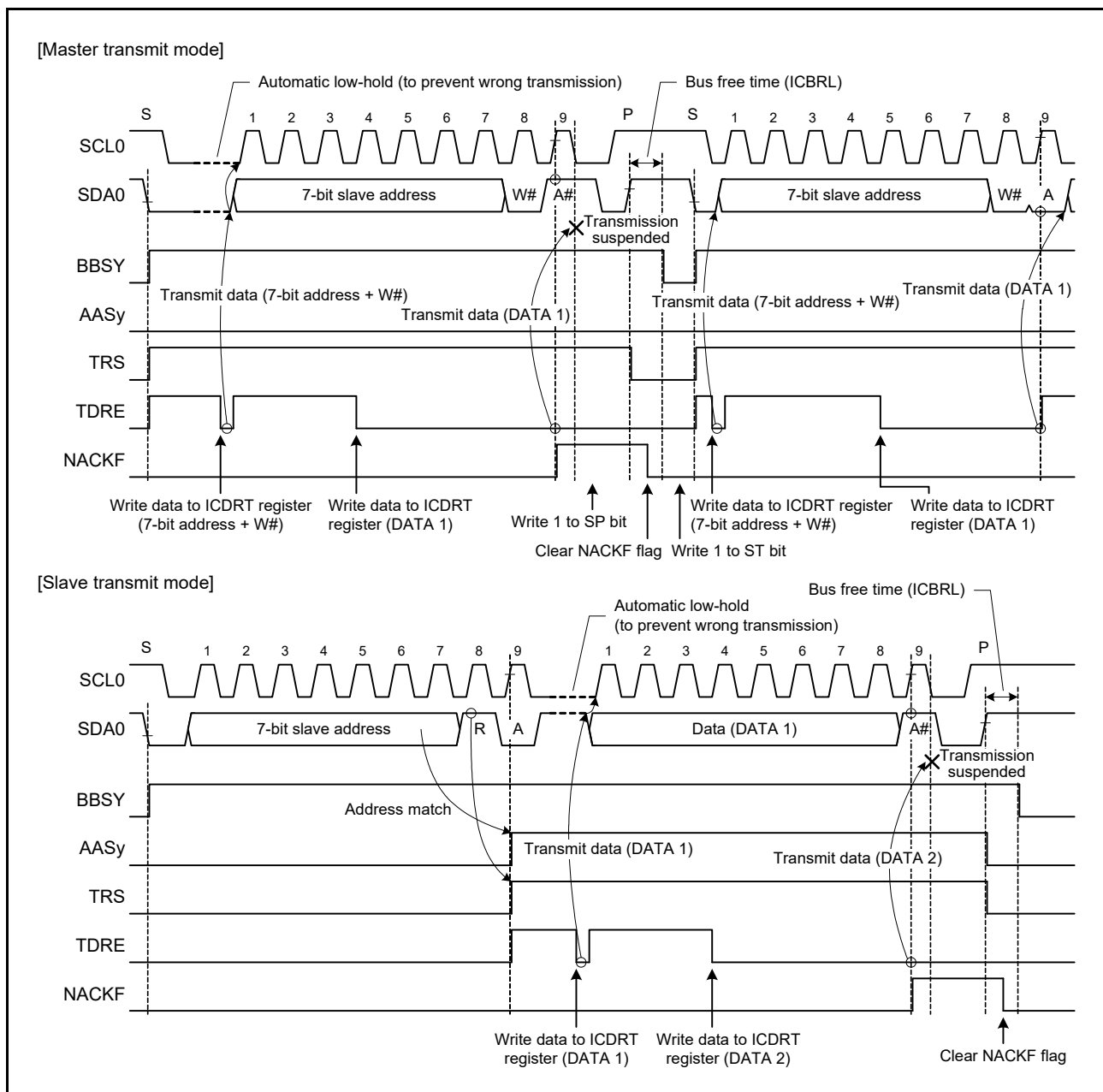


Figure 27.30 Suspension of Data Transmission When NACK is Received (NACKE = 1)

27.8.3 Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (ICDRR) read is delayed for a period of one transfer byte or more with receive data full (ICSR2.RDRF flag is 1) in receive mode (ICCR2.TRS bit is 0), the RIIC holds the SCL0 line low automatically immediately before the next data is received to prevent failure to receive data.

This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, the RIIC's own slave address is designated after a stop condition is generated. This function does not disturb other communication because the RIIC does not hold the SCL0 line low when a mismatch with its own slave address occurs after a stop condition is generated.

Sections in which the SCL0 line is held low can be selected with a combination of the WAIT and RDRFS bits in the ICMR3 register.

(1) 1-Byte Receive Operation and Automatic Low-Hold Function Using the WAIT Bit

When the ICMR3.WAIT bit is set to 1, the RIIC performs 1-byte receive operation using the WAIT bit function.

Furthermore, when the ICMR3.RDRFS bit is 0, the RIIC automatically sends the ICMR3.ACKBT bit value for the acknowledgment bit in the period from the falling edge of the eighth SCL to the falling edge of the ninth SCL, and automatically holds the SCL0 line low at the falling edge of the ninth SCL using the WAIT bit function. This low-hold is released by reading data from the ICDRR register, which enables byte-wise receive operation.

The WAIT bit function is enabled for receive bytes after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

(2) 1-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the RDRFS Bit

When the ICMR3.RDRFS bit is set to 1, the RIIC performs 1-byte receive operation using the RDRFS bit function.

When the RDRFS bit is set to 1, the ICSR2.RDRF flag (receive data full) is set to 1 at the rising edge of the eighth SCL, and the SCL0 line is automatically held low at the falling edge of the eighth SCL. This low-hold is released by writing a value to the ICMR3.ACKBT bit, but cannot be released by reading data from the ICDRR register, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.

The RDRFS bit function is enabled for receive bytes after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

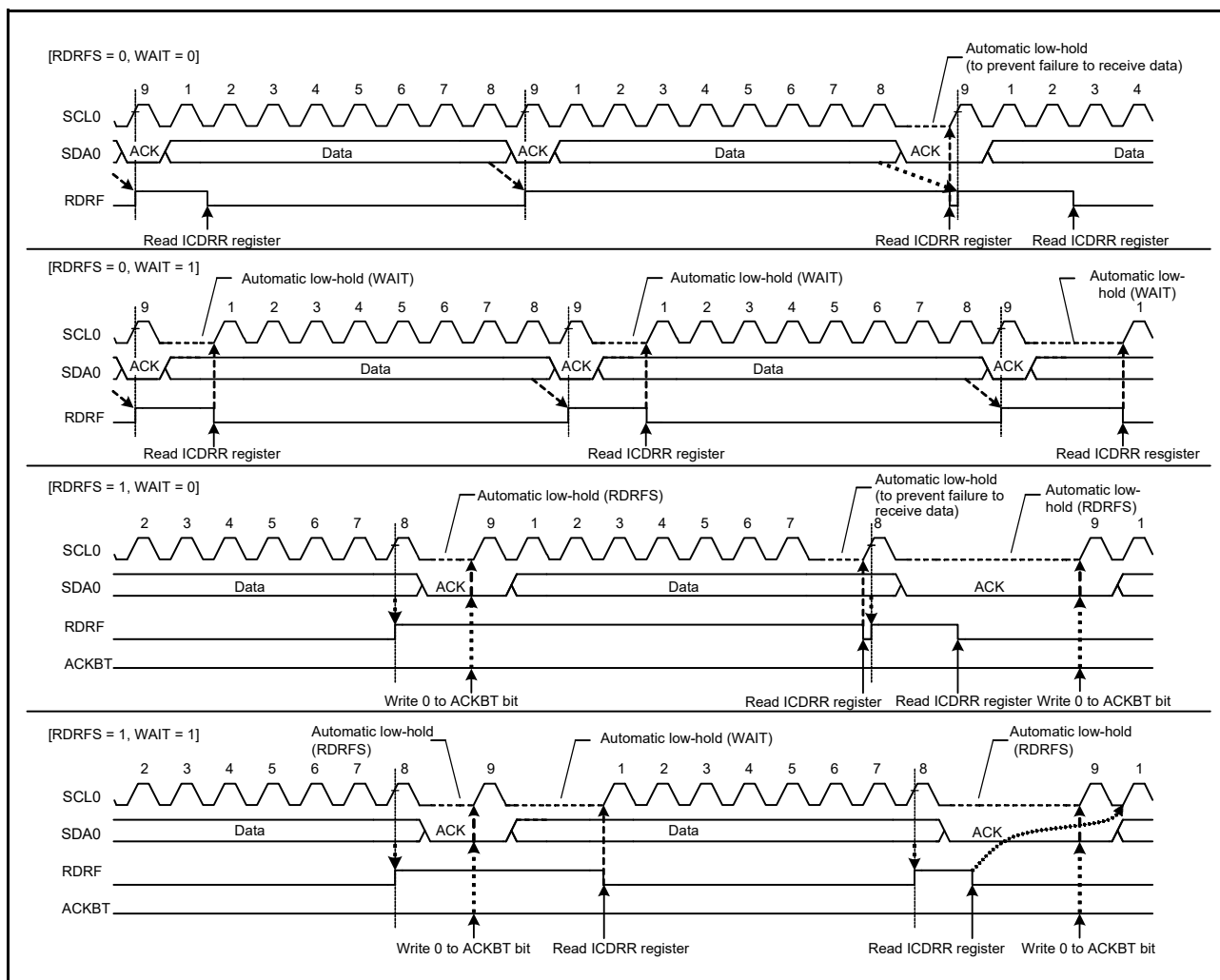


Figure 27.31 Automatic Low-Hold Operation in Receive Mode (Using RDRFS and WAIT Bits)

27.9 Arbitration-Lost Detection Functions

In addition to the normal arbitration-lost detection function defined by the I²C-bus specification, the RIIC has functions to prevent double-generation of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration-lost in slave transmit mode.

27.9.1 Master Arbitration-Lost Detection (MALE Bit)

The RIIC drives the SDA0 line low to generate a start condition. However, if the SDA0 line has already been driven low by another master device generating a start condition, the RIIC causes arbitration to be lost, so priority is given to transfer by the other master device. Similarly, if the ICCR2.ST bit is set to 1 while the ICCR2.BBSY flag is 1 (bus busy state), arbitration is lost, so priority is given to transfer by the other master device. No start condition is generated in this case. When a start condition is generated successfully, if the data for transmission including the address bits (i.e. the internal SDA output level) and the level on the SDA0 line do not match (the high output as the internal SDA output; i.e. the SDA0 pin is in the high-impedance state) and the low is detected on the SDA0 line, the RIIC loses the arbitration. After a master arbitration-lost is generated, the RIIC immediately enters slave receive mode. If a slave address (including the general call address) matches its own address at this time, the RIIC continues in slave operation.

A master arbitration-lost is detected when the following conditions are met while the ICFER.MALE bit is 1 (master arbitration-lost detection enabled).

Conditions for detecting master arbitration-lost

- Non-matching of the internal level for output on SDA and the level on the SDA0 line after a start condition was generated by setting the ICCR2.ST bit to 1 while the ICCR2.BBSY flag was set to 0 (erroneous generation of a start condition)
- Setting of the ICCR2.ST bit to 1 while the BBSY flag is set to 1 (start condition double-generation error)
- When the transmit data excluding acknowledgment bit (internal SDA output level) does not match the level on the SDA0 line in master transmit mode (bits MST and TRS in the ICCR2 register = 11b)

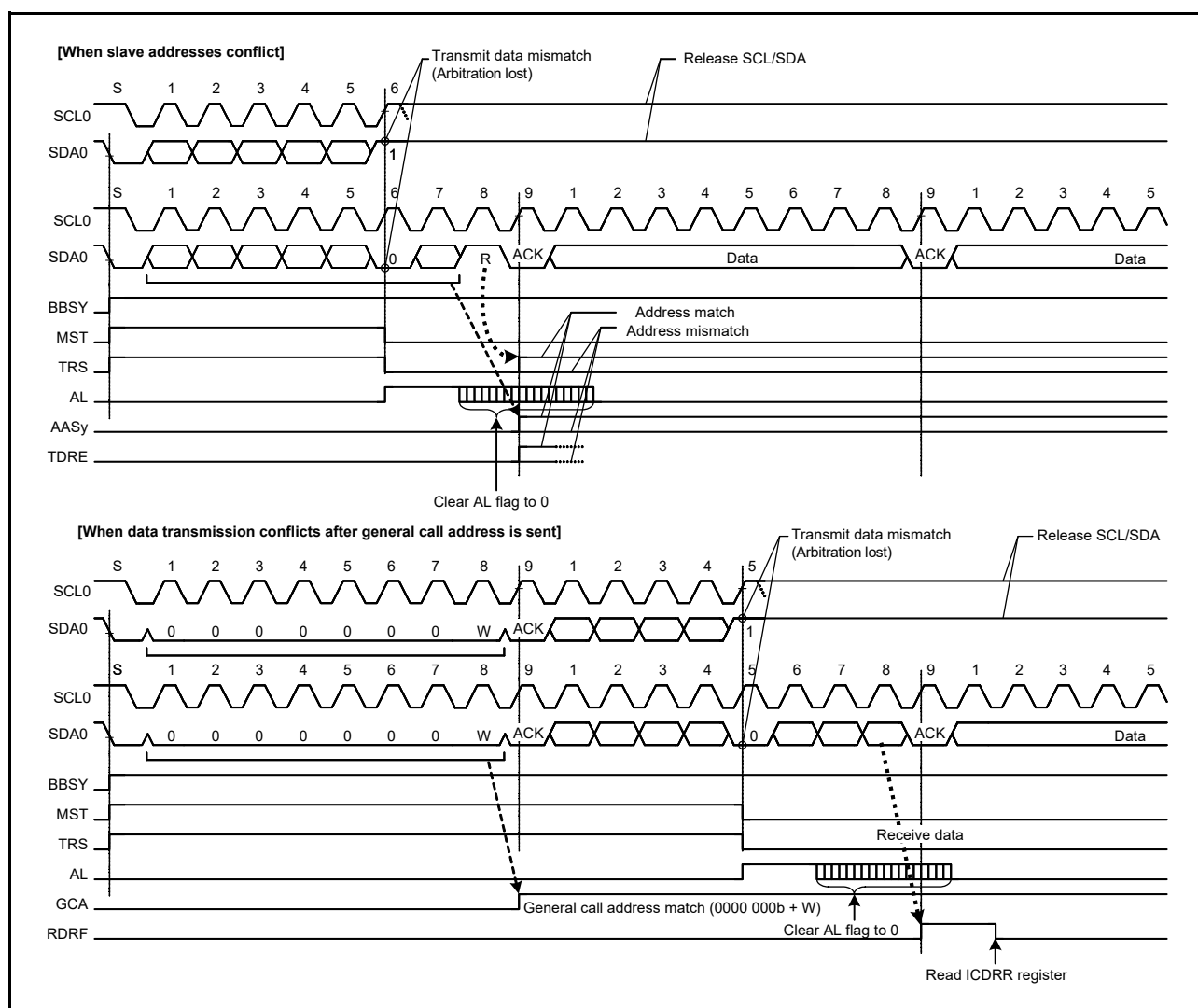


Figure 27.32 Examples of Master Arbitration-Lost Detection (MALE = 1)

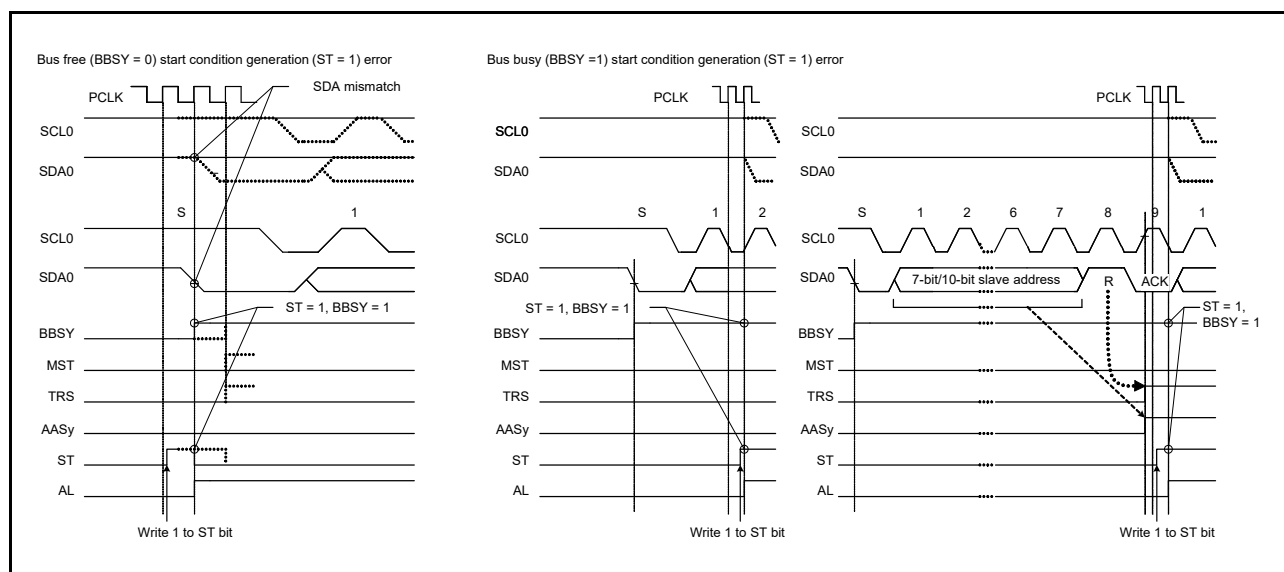


Figure 27.33 Arbitration-Lost When a Start Condition is Generated (MALE = 1)

27.9.2 NACK Transmission Arbitration-Lost Detection Function (NALE Bit)

The RIIC has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDA0 line (the high output as the internal SDA output; i.e. the SDA0 pin is in the high-impedance state) and the low is detected on the SDA0 line during transmission of NACK in receive mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device. Figure 27.34 shows an example of NACK transmission arbitration-lost detection.

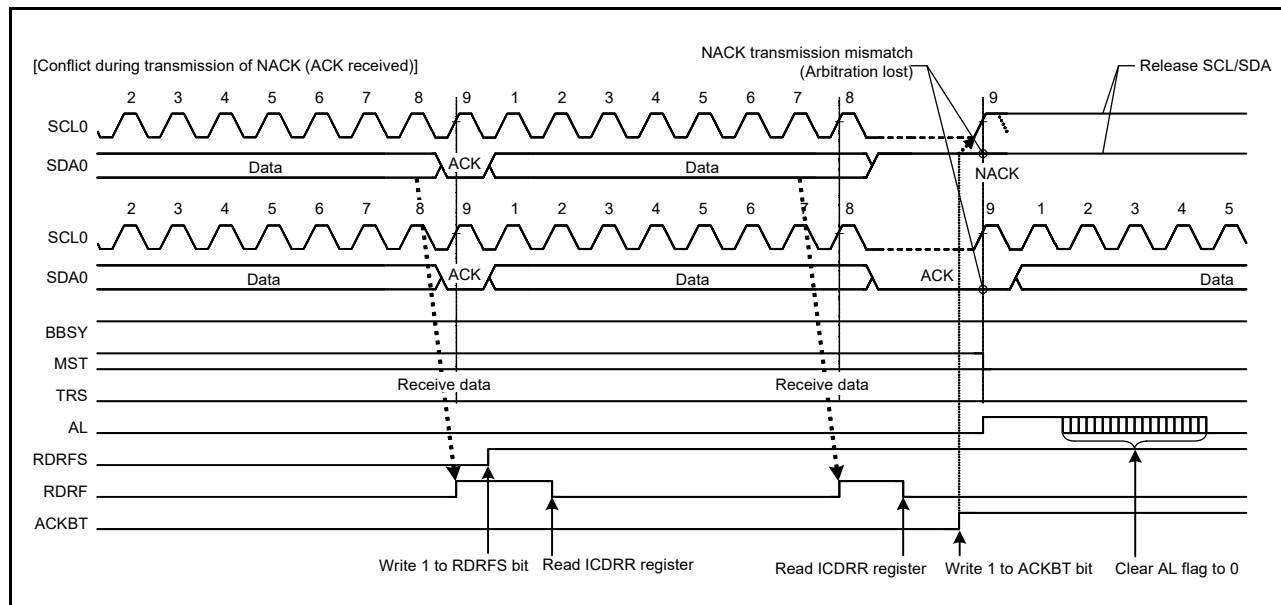


Figure 27.34 Example of NACK transmission arbitration-lost Detection (NALE = 1)

The following explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives two bytes of data from the slave device, and master B receives four bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. Here, master A sends NACK when it has received second byte of data from the slave device. Meanwhile, master B sends ACK because it has not received necessary four bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and generates a stop condition. Therefore, the generation of the stop condition conflicts with the SCL output of master B, which disturbs communication.

When the RIIC receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If a NACK transmission arbitration-lost occurs, the RIIC immediately cancels the slave address matched state and enters slave receive mode. This prevents a stop condition from being generated, preventing a communication failure on the bus. Also, in the ARP command processing of SMBus, it is possible to omit surplus processing (FFh transmission processing) after NACK transmission when the UDID (Unique Device Identifier) of “Assign Address” does not match and after the NACK transmission of the “Get UDID (General)” after the “Assign Address” is confirmed.

The RIIC detects NACK transmission arbitration-lost when the following condition is met with the ICFER.NALE bit set to 1 (NACK transmission arbitration-lost is enabled).

Condition for detecting NACK transmission arbitration-lost

- When the internal SDA output level does not match the SDA0 line (ACK is received) during transmission of NACK (ICMR3.ACKBT bit = 1)

27.9.3 Slave Arbitration-Lost Detection (SALE Bit)

The RIIC has a function to cause arbitration to be lost if the data for transmission (i.e. the internal SDA output level) and the level on the SDA0 line do not match (the high output as the internal SDA output; i.e. the SDA0 pin is in the high-impedance state and the low is detected on the SDA0 line in slave transmit mode). The slave arbitration-lost detection function is mainly used when transmitting a UDID (Unique Device Identifier) of SMBus.

When the slave arbitration-lost occurs, the RIIC is immediately released from the slave address matched state and enters slave receive mode. This function can detect a data conflict during UDID transmission of SMBus and omit surplus processing (FFh transmission processing) after the data conflict.

The RIIC detects slave arbitration-lost when the following condition is met with the ICFER.SALE bit set to 1 (slave arbitration-lost detection enabled).

Condition for detecting slave arbitration-lost

- When transmit data excluding acknowledgment bit (internal SDA output level) does not match the SDA0 line in slave transmit mode (bits MST and TRS in the ICCR2 register are 01b)

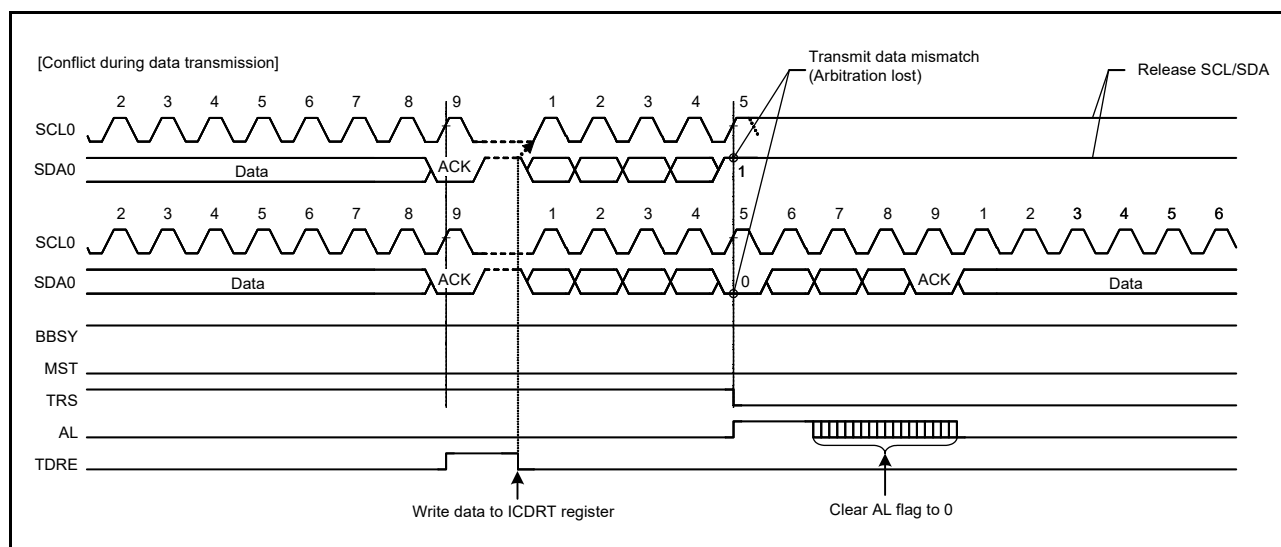


Figure 27.35 Example of Slave Arbitration-Lost Detection (SALE = 1)

27.10 Start Condition/Restart Condition/Stop Condition Generating Function

27.10.1 Generating a Start Condition

The RIIC generates a start condition when the ICCR2.ST bit is set to 1.

When the ST bit is set to 1, a start condition generation request is made and the RIIC generates a start condition when the ICCR2.BBSY flag is 0 (bus free state). When a start condition is generated normally, the RIIC automatically shifts to the master transmit mode.

A start condition is generated in the following sequence.

Start condition generation

- (1) Drive the SDA0 line low (high to low).
- (2) Secure the time set in the ICBRH register and the start condition hold time.
- (3) Drive the SCL0 line low (high to low).
- (4) Detect the low level on the SCL0 line and secure the low period of the signal on the SCL0 line set in the ICBRL register.

27.10.2 Generating a Restart Condition

The RIIC generates a restart condition when the ICCR2.RS bit is set to 1.

When the RS bit is set to 1, a restart condition generation request is made and the RIIC generates a restart condition when the ICCR2.BBSY flag is 1 (bus busy state) and the ICCR2.MST bit is 1 (master mode).

A restart condition is generated in the following sequence.

Restart condition generation

- (1) Release the SDA0 line.
- (2) Secure the low period of the signal on the SCL0 line set in the ICBRL register.
- (3) Release the SCL0 line (low to high).
- (4) Detect the high level on the SCL0 line and secure the time set in the ICBRL register and the restart condition setup time.
- (5) Drive the SDA0 line low (high to low).
- (6) Secure the time set in the ICBRH register and the restart condition hold time.
- (7) Drive the SCL0 line low (high to low).
- (8) Detect the low level on the SCL0 line and secure the low period of the signal on the SCL0 line set in the ICBRL register.

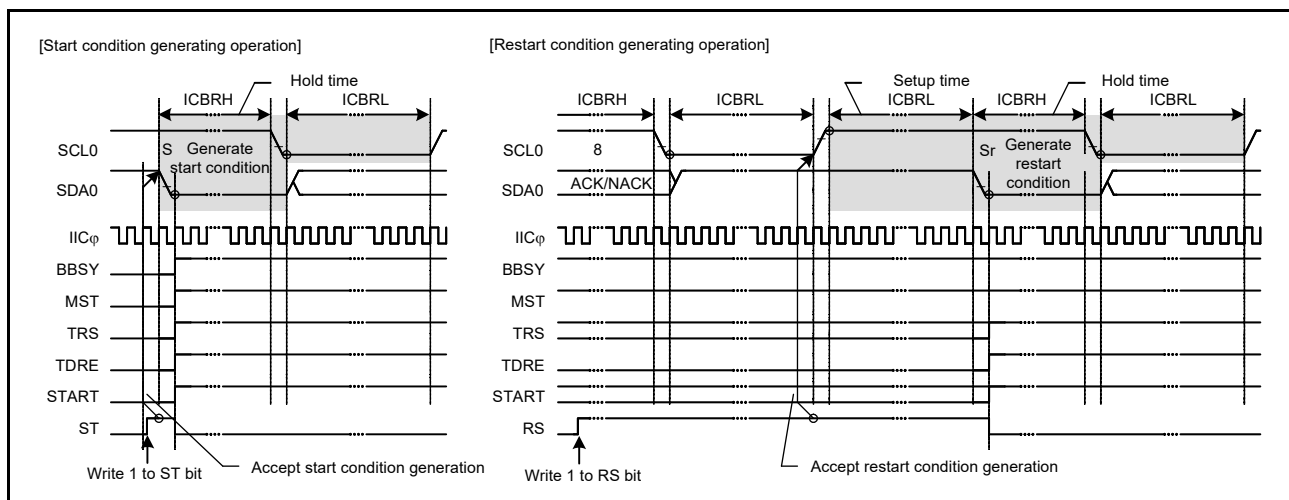


Figure 27.36 Start Condition/Restart Condition Generation Timing (ST and RS Bits)

27.10.3 Generating a Stop Condition

The RIIC generates a stop condition when the ICCR2.SP bit is set to 1.

When the SP bit is set to 1, a stop condition generation request is made and the RIIC generates a stop condition when the ICCR2.BBSY flag is 1 (bus busy state) and the ICCR2.MST bit is 1 (master mode).

A stop condition is generated in the following sequence.

Stop condition generation

- (1) Drive the SDA0 line low (high to low).
- (2) Secure the low period of the signal on the SCL0 line set in the ICBRL register.
- (3) Release the SCL0 line (low to high).
- (4) Detect the high level on the SCL0 line and secure the time set in the ICBRH register and the stop condition setup time.
- (5) Release the SDA0 line (low to high).
- (6) Secure the time set in the ICBRL register and the bus free time.
- (7) Set the BBSY flag to 0 (to release the bus mastership).

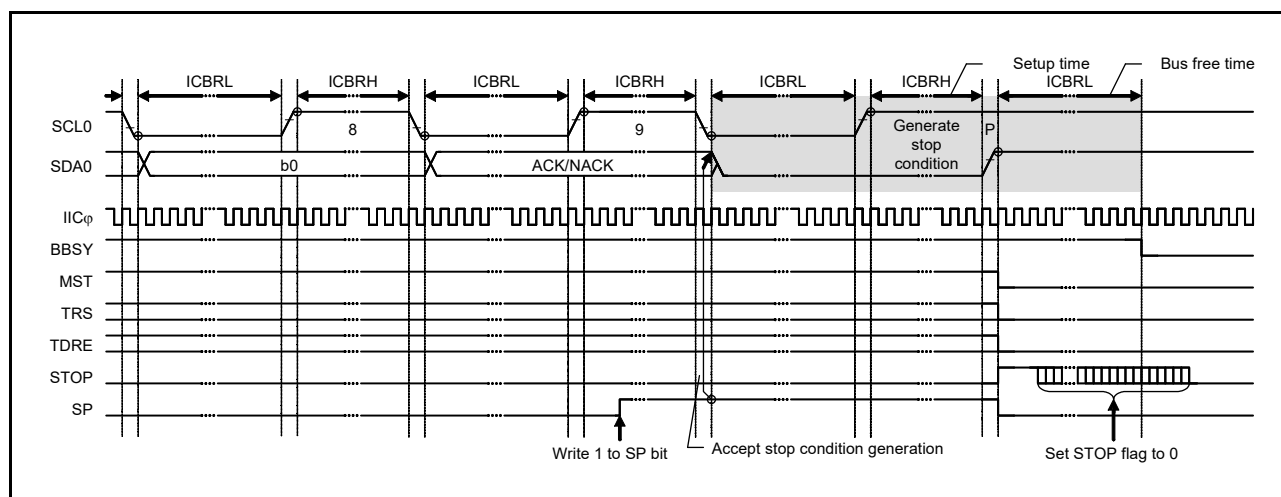


Figure 27.37 Stop Condition Generation Timing (SP Bit)

27.11 Bus Hanging

If the clock signals from the master and slave devices go out of synchronization due to noise or other factors, the I²C-bus might hang with a fixed level on the SCL0 line and/or SDA0 line.

As measures against the bus hanging, the RIIC has a timeout function to detect hanging by monitoring the SCL0 line, a function for the output of an additional SCL to release the bus from a hung state due to clock signals being out of synchronization, the RIIC reset function, and internal reset function.

By checking bits SCLO, SDAO, SCLI, and SDAI in the ICCR1 register, it is possible to see whether the RIIC or its partner in communications is placing the low level on the SCL0 or SDA0 lines.

27.11.1 Timeout Function

The RIIC includes a timeout function for detecting when the SCL0 line has been stuck longer than the predetermined time. The RIIC can detect an abnormal bus state by monitoring that the SCL0 line is stuck low or high for a predetermined time.

The timeout function monitors the SCL0 line state and counts the low period or high period using the internal counter. The timeout function resets the internal counter each time the SCL0 line changes (rising or falling), but continues to count unless the SCL0 line changes. If the internal counter overflows due to no SCL0 line change, the RIIC can detect the timeout and report the bus hung state.

This timeout function is enabled when the ICFER.TMOE bit is 1. It detects a hung state that the SCL0 line is stuck low or high during the following conditions:

- The bus is busy (ICCR2.BBSY flag is 1) in master mode (ICCR2.MST bit is 1).
- The RIIC's own slave address is detected (ICSR1 register is not 00h) and the bus is busy (ICCR2.BBSY flag is 1) in slave mode (ICCR2.MST bit is 0).
- The bus is free (ICCR2.BBSY flag is 0) while generation of a start condition is requested (ICCR2.ST bit is 1).

The internal counter of the timeout function works using the internal reference clock (IIC ϕ) set by the ICMR1.CKS[2:0] bits as a count source. It functions as a 16-bit counter when long mode is selected (ICMR2.TMOS bit is 0) or a 14-bit counter when short mode is selected (TMOS bit is 1).

The SCL0 line level (low/high or both levels) during which this counter is activated can be selected by the setting of bits TMOH and TMOL in the ICMR2 register. If both bits TMOL and TMOH are set to 0, the internal counter does not work.

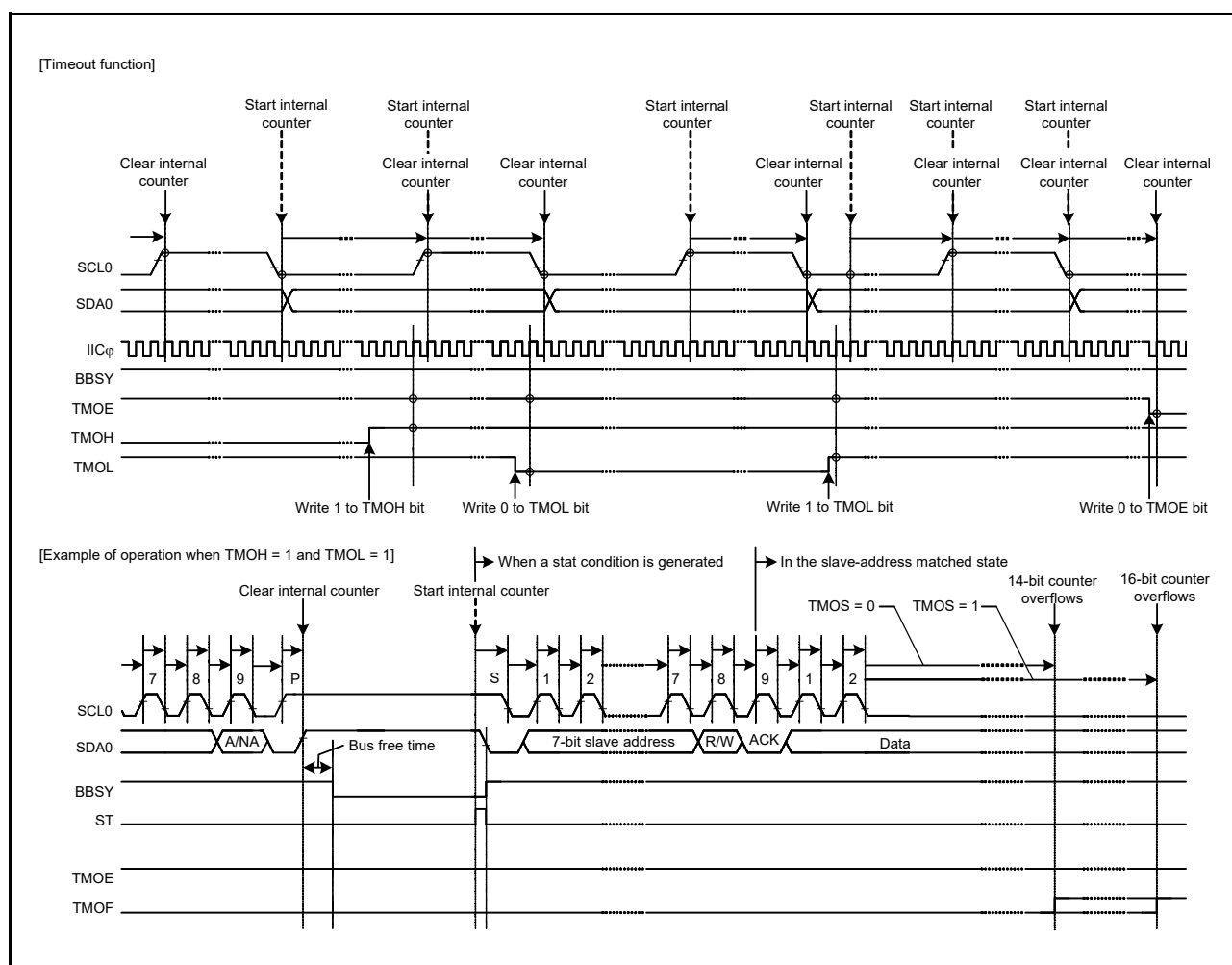


Figure 27.38 Timeout Function

27.11.2 Additional SCL Output Function

In master mode, the RIIC module has a facility for the output of additional SCL to release the SDA0 line from being held low by the slave device due to the master being out of synchronization with the slave device.

This function is mainly used in master mode to release the SDA0 line from the state of being stuck low by including additional SCL output from the RIIC with single cycles of the SCL as the unit if the RIIC cannot generate a stop condition because the slave device is holding the SDA0 line low. Do not use this function in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.

When the ICCR1.CLO bit is set to 1, an additional clock pulse at the frequency set by the ICMR1.CKS[2:0] bits and the ICBRH and ICBRL registers is output from the SCL0 pin. After output of this clock pulse, the CLO bit automatically becomes 0. The SCL0 pin is held low when the ICCR2.BBSY flag is 1 and held high when the BBSY flag is 0.

Consecutive additional clock pulses can be output by writing 1 to the CLO bit after confirming the CLO bit to be 0.

When the RIIC module is in master mode and the slave device is holding the SDA0 line low because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a stop condition is not possible. The additional SCL output function can be used to output additional clock pulses one by one to make the slave device release the SDA0 line from being held low, thus recovering the bus from an unusable state. Release of the SDA0 line by the slave device can be monitored by reading the ICCR1.SDAI bit. After confirming release of the SDA0 line by the slave device, complete communications by regenerating a stop condition.

Use this function with the ICFER.MALE bit set to 0 (master arbitration-lost detection disabled).

Conditions for using the ICCR1.CLO bit

- When the bus is free (ICCR2.BBSY flag is 0) or in master mode (ICCR2.MST bit is 1 and ICCR2.BBSY flag is 1)
- When the communication device does not hold the SCL0 line low

Figure 27.39 shows the operation timing of the additional SCL output function (CLO bit).

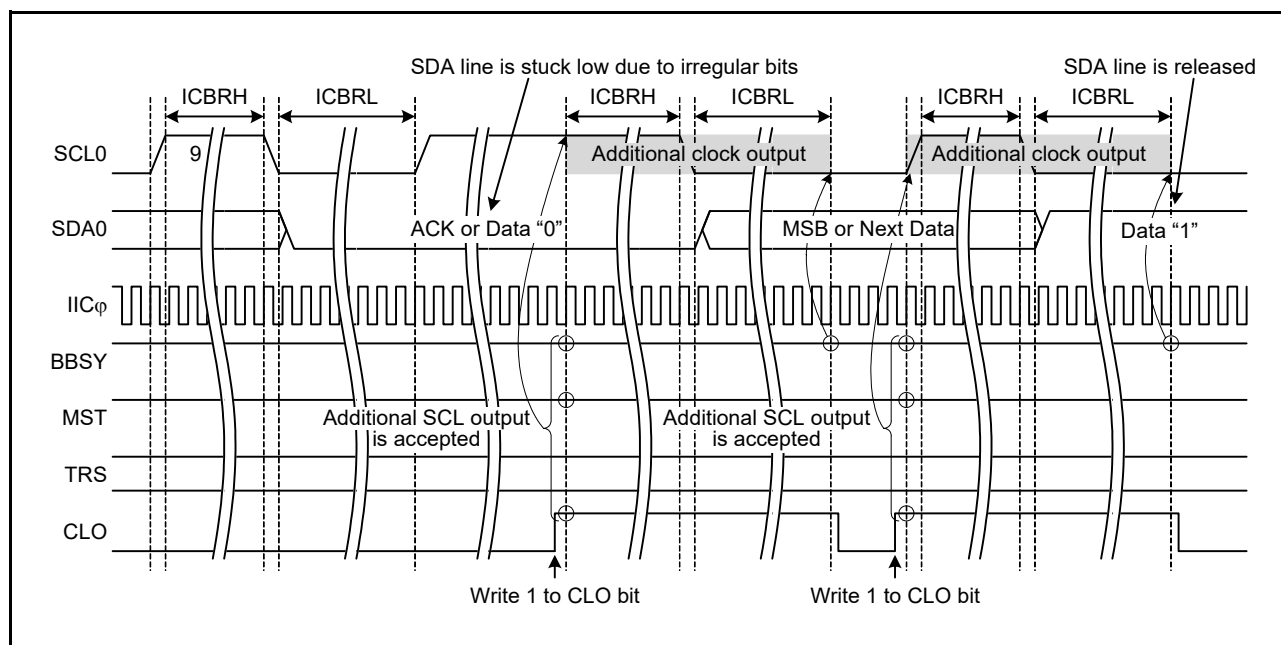


Figure 27.39 Additional SCL Output Function (CLO Bit)

27.11.3 RIIC Reset and Internal Reset

The RIIC module incorporates a function for resetting itself. There are two types of reset. One is referred to as an RIIC reset; this initializes all registers including the ICCR2.BBSY flag. The other is referred to as an internal reset; this releases the RIIC from the slave-address matched state and initializes the internal counter while retaining other settings. After applying a reset, be sure to set the ICCR1.IICRST bit to 0.

Both types of reset are effective for release from bus-hung states because both restore the output state of the SCL0 and SDA0 pins to the high-impedance state.

Applying a reset during slave operation may lead to a loss of synchronization between the master device clock and the slave device clock, so avoided this where possible. Note that monitoring of the bus state, such as for the presence of a start condition, is not possible during an RIIC reset (bits ICE and IICRST in the ICCR1 register are 01b).

For a detailed description of the RIIC and internal resets, refer to section 27.14, Initialization of Registers and Functions When a Reset is Applied or a Condition is Detected.

27.12 SMBus Operation

The RIIC is available for data communication conforming to the SMBus (Version 2.0). To perform SMBus communication, set the ICMR3.SMBS bit to 1. To use the transfer rate within a range of 10 kbps to 100 kbps of the SMBus specification, set the ICMR1.CKS[2:0] bits, the ICBRH register, and the ICBRL register. In addition, determine the values of the ICMR2.DLCS bit and the ICMR2.SDDL[2:0] bits to meet the data hold time (300 ns (min.)). If the RIIC is used only as a slave device, the transfer rate setting is not necessary, whereas the ICBRL register needs to be set to a value longer than the data setup time (250 ns).

For the SMBus device default address (1100 001b), use one of the slave address registers L0 to L2 (registers SARL0, SARL1, and SARL2), and set the corresponding SARUy.FS bit (7-bit/10-bit address format select) (y = 0 to 2) to 0 (7-bit address format).

When transmitting the UDID (Unique Device Identifier), set the ICFER.SALE bit to 1 to enable the slave arbitration-lost detection function.

27.12.1 SMBus Timeout Measurement

(1) Measuring timeout of slave device

The following period (timeout interval: $t_{\text{LOW:SEXT}}$) must be measured for slave devices in SMBus communication.

- From start condition to stop condition

To measure timeout for slave devices, measure the period from start condition detection to stop condition detection with the MTU or TMR timer using a start condition detection interrupt (STI) and stop condition detection interrupt (SPI) of the RIIC. The measured timeout period must be within the cumulative clock low extend time (slave device) ($t_{\text{LOW:SEXT}}$: 25 ms (max.)) of the SMBus specification.

If the time measured with the MTU or TMR exceeds the detect clock low timeout (t_{TIMEOUT} : 25 ms (min.)) of the SMBus specification, the slave device must release the bus by writing 1 to the ICCR1.IICRST bit to apply an internal reset of the RIIC. When an internal reset is applied to the RIIC, it stops driving the SCL0 and SDA0 pins of the bus and makes the SCL0/SDA0 pin outputs high-impedance, thus releasing the bus.

(2) Measuring timeout of master device

The following periods (timeout interval: $t_{\text{LOW:MEXT}}$) must be measured for master devices in SMBus communication.

- From start condition to acknowledgment bit
- Between acknowledgment bits
- From acknowledgment bit to stop condition

To measure timeout for master devices, measure these periods with the MTU or TMR timer using a start condition detection interrupt (STI), stop condition detection interrupt (SPI), and transmission end interrupt (TEI) or receive data full interrupt (RXI) of the RIIC. The measured timeout period must be within the cumulative clock low extend time (master device) ($t_{\text{LOW:MEXT}}$: 10 ms (max.)) of the SMBus specification, and the total of all $t_{\text{LOW:MEXT}}$ from start condition to stop condition must be within $t_{\text{LOW:SEXT}}$: 25 ms (max.).

For the ACK receive timing (rising edge of the ninth SCL), monitor the ICSR2.TEND flag in master transmit mode (master transmitter) and the ICSR2.RDRF flag in master receive mode (master receiver). For this reason, perform bitwise transmit operation in master transmit mode, and hold the ICMR3.RDRFS bit 0 until the byte just before reception of the final byte in master receive mode. While the RDRFS bit is 0, the RDRF flag is set to 1 at the rising edge of the ninth SCL.

If the period measured with the MTU or TMR exceeds the cumulative clock low extend time (master device) ($t_{\text{LOW:MEXT}}$: 10 ms (max.)) of the SMBus specification or the total of measured periods exceeds the detect clock low timeout (t_{TIMEOUT} : 25 ms (min.)) of the SMBus specification, the master device must stop the transaction by generating a stop condition. In master transmit mode, immediately stop the transmit operation (writing data to the ICDRT register).

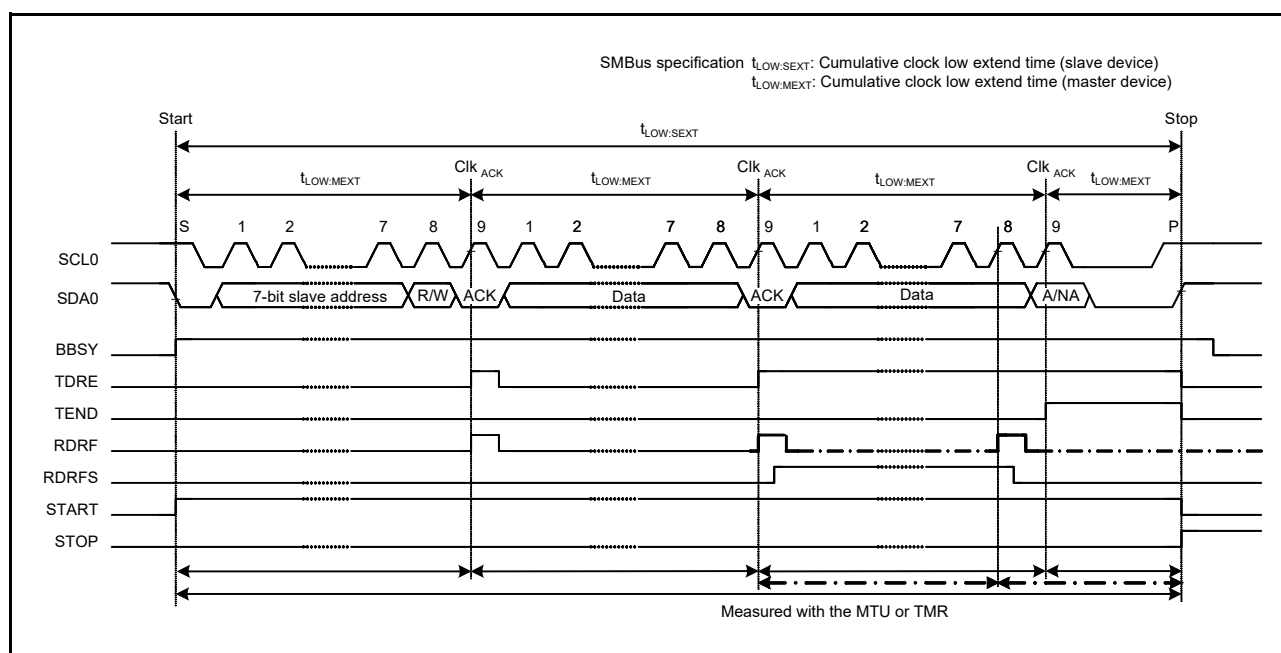


Figure 27.40 SMBus Timeout Measurement

27.12.2 Packet Error Code (PEC)

This MCU incorporates a CRC calculator. The CRC calculator enables transmission of a packet error code (PEC) or checking the received data of the SMBus in data communication of the RIIC. For the CRC generating polynomials of the CRC calculator, refer to [section 28, CRC Calculator \(CRCA\)](#).

The PEC data in master transmit mode can be generated by writing all transmit data to the CRC data input register (CRCDIR) in the CRC calculator.

The PEC data in master receive mode can be checked by writing all receive data to CRCDIR in the CRC calculator and comparing the obtained value in the CRC data output register (CRCDOR) with the received PEC data.

To send ACK or NACK according to the match or mismatch result when the final byte is received as a result of the PEC code check, set the ICMR3.RDRFS bit to 1 before the rising edge of the eighth SCL during reception of the final byte, and hold the SCL0 line low at the falling edge of the eighth clock pulse.

27.12.3 SMBus Host Notification Protocol (Notify ARP Master Command)

In communications over an SMBus, a slave device can temporarily act as a master device to notify the SMBus host (or ARP master) of its own slave address or to request its own slave address from the SMBus host.

For a product of this MCU to operate as an SMBus host (or ARP master), the host address (0001 000b) sent from the slave device must be detected as a slave address, so the RIIC has a function for detecting the host address. To detect the host address as a slave address, set the ICMR3.SMBS bit and the ICSER.HOAE bit to 1. Operation after the host address has been detected is the same as normal slave operation.

27.13 Interrupt Sources

The RIIC generates four types of interrupt requests: communication error or communication event (arbitration-lost detection, NACK detection, timeout detection, start condition detection, and stop condition detection), receive data full, transmit data empty, and transmission end.

Table 27.6 lists details of the several interrupt requests. The receive data full and transmit data empty interrupt requests allow the DTC to start data transfer.

Table 27.6 Interrupt Sources

Symbol	Interrupt Source	Interrupt Flag	Start DTC Transfer	Interrupt Generation Condition
EEI	Communication error/ communication event	AL	Not possible	AL = 1 and ALIE = 1
		NACKF		NACKF = 1 and NAKIE = 1
		TMOF		TMOF = 1 and TMOIE = 1
		START		START = 1 and STIE = 1
		STOP		STOP = 1 and SPIE = 1
RXI*2	Receive data full	RDRF	Possible	RDRF = 1 and RIE = 1
TXI*1	Transmit data empty	TDRE	Possible	TDRE = 1 and TIE = 1
TEI*3	Transmission end	TEND	Not possible	TEND = 1 and TEIE = 1

Note: There is a delay time between the execution of a write instruction for a peripheral module by the CPU and actual writing to the module. Thus, when an interrupt flag has been cleared or an interrupt request has been masked, read the relevant flag again to check whether clearing or masking has been completed, and then return from interrupt handling. Returning from interrupt handling without checking that writing to the module has been completed creates a possibility of repeated processing of the same interrupt.

Note 1. Because TXI is an edge-detected interrupt, it does not require clearing. Furthermore, the ICSR2.TDRE flag (a condition for TXI) is automatically set to 0 when data for transmission are written to the ICDRT register or a stop condition is detected (ICSR2.STOP flag is 1).

Note 2. Because RXI is an edge-detected interrupt, it does not require clearing. Furthermore, the ICSR2.RDRF flag (a condition for RXI) is automatically set to 0 when data are read from the ICDRR register.

Note 3. When using the TEI interrupt, clear the ICSR2.TEND flag in the TEI interrupt handling.
Note that the ICSR2.TEND flag is automatically set to 0 when data for transmission are written to the ICDRT register or a stop condition is detected (ICSR2.STOP flag is 1).

Clear the each flag or mask the interrupt request during interrupt handling.

27.13.1 Buffer Operation for TXI and RXI Interrupts

If the conditions for generating a TXI and RXI interrupt are satisfied while the corresponding ICU.IRn.IR flag is 1, the interrupt request is not output for the ICU but retained internally (the capacity for internal retention is one request per source).

An interrupt request that was being retained internally is output to the ICU when the value of the IR flag becomes 0.

Internally retained interrupt requests are automatically cleared under normal conditions of usage.

Internally retained interrupt requests can also be cleared by writing 0 to the corresponding interrupt enable bit in the ICIER register.

27.14 Initialization of Registers and Functions When a Reset is Applied or a Condition is Detected

The RIIC can be reset by MCU reset, RIIC reset, and internal reset functions. Table 27.7 lists the reset states of registers and functions when a reset is applied or a condition is detected.

Table 27.7 Reset States of Registers and Functions When a Reset is Applied or a Condition is Detected

		MCU Reset	RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/ Restart Condition Detection	Stop Condition Detection
ICCR1	SDAO, SCLO	To be reset	To be reset	To be reset	Retained	Retained
	IICRST, ICE		Retained	Retained		
	Others		To be reset			
ICCR2	ST, RS	To be reset	To be reset	To be reset	To be reset	Retained
	SP					To be reset
	TRS				See note 1	
	MST				See note 1	
	BBSY			Retained	Becomes 1	
ICMR1	BC[2:0]	To be reset	To be reset	To be reset	To be reset	Retained
	Others			Retained	Retained	
ICMR2		To be reset	To be reset	Retained	Retained	Retained
ICMR3	ACKBT	To be reset	To be reset	Retained	Retained	To be reset
	Others					Retained
ICFER		To be reset	To be reset	Retained	Retained	Retained
ICSER		To be reset	To be reset	Retained	Retained	Retained
ICIER		To be reset	To be reset	Retained	Retained	Retained
ICSR1		To be reset	To be reset	To be reset	Retained	To be reset
ICSR2	START	To be reset	To be reset	To be reset	Becomes 1	To be reset
	STOP				Retained	Becomes 1
	TEND					To be reset
	TDRE				See note 1	
	Others				Retained	Retained
SARL0, SARL1, SARL2, SARU0, SARU1, SARU2		To be reset	To be reset	Retained	Retained	Retained
ICBRH, ICBRL		To be reset	To be reset	Retained	Retained	Retained
ICDRT		To be reset	To be reset	Retained	Retained	Retained
ICDRR		To be reset	To be reset	Retained	Retained	Retained
ICDRS		To be reset	To be reset	To be reset	Retained	Retained
Timeout function		To be reset	To be reset	To be reset	Operation	Operation
Bus free time measurement		To be reset	To be reset	Operation	Operation	Operation

Note 1. This bit is not reset. This bit becomes 0 or 1 in accordance with the conditions.

27.15 Usage Notes

27.15.1 Setting Module Stop Function

Module stop state can be entered or released using module stop control register B (MSTPCRB). The initial setting is for operation of the RIIC to be stopped. RIIC register access is enabled by releasing the module stop state.

For details on module stop control register B, refer to **section 11, Low Power Consumption**.

27.15.2 Notes on Starting Transfer

If the IR flag corresponding to the RIIC interrupt is 1 when transfer is started (ICCR1.ICE bit is 1), follow the procedure below to clear interrupts before enabling operations. Starting transfer with the IR flag set to 1 while the ICCR1.ICE bit is 1 leads to an interrupt request being internally retained after transfer starts, and this can lead to unanticipated behavior of the IR flag.

1. Confirm that the ICCR1.ICE bit is 0.
2. Set the relevant interrupt enable bits (ICIER.TIE, etc.) on the peripheral function side to 0.
3. Read the relevant interrupt enable bits (ICIER.TIE, etc.) on the peripheral function side and confirm that its value is 0.
4. Set the IR flag to 0.

28. CRC Calculator (CRCA)

The CRC (Cyclic Redundancy Check) calculator generates CRC codes.

28.1 Overview

Table 28.1 lists the specifications of the CRC calculator, and Figure 28.1 shows a block diagram of the CRC calculator.

Table 28.1 CRC Specifications

Item	Description
Data size	8 bits 32 bits
Data for CRC calculation*1	CRC codes are generated for any desired data in 8n-bit units (where n is a whole number) CRC codes are generated for any desired data in 32n-bit units (where n is a whole number)
CRC processor unit	8-bit parallel processing 32-bit parallel processing
CRC generating polynomial	One of three generating polynomials is selectable • 8-bit CRC $X^8 + X^2 + X + 1$ • 16-bit CRC $X^{16} + X^{15} + X^2 + 1$ $X^{16} + X^{12} + X^5 + 1$ One of two generating polynomials is selectable • 32-bit CRC $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8$ $+ X^7 + X^5 + X^4 + X^2 + X + 1$ $X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19}$ $+ X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^8 + X^6 + 1$
CRC calculation switching	The order of the bits produced by CRC calculation can be switched for LSB first or MSB first communication
Low power consumption	Module stop state can be set.

Note 1. The circuit does not have a function to divide data for calculation into CRC calculation units. Write data in 8-bit or 32-bit units.

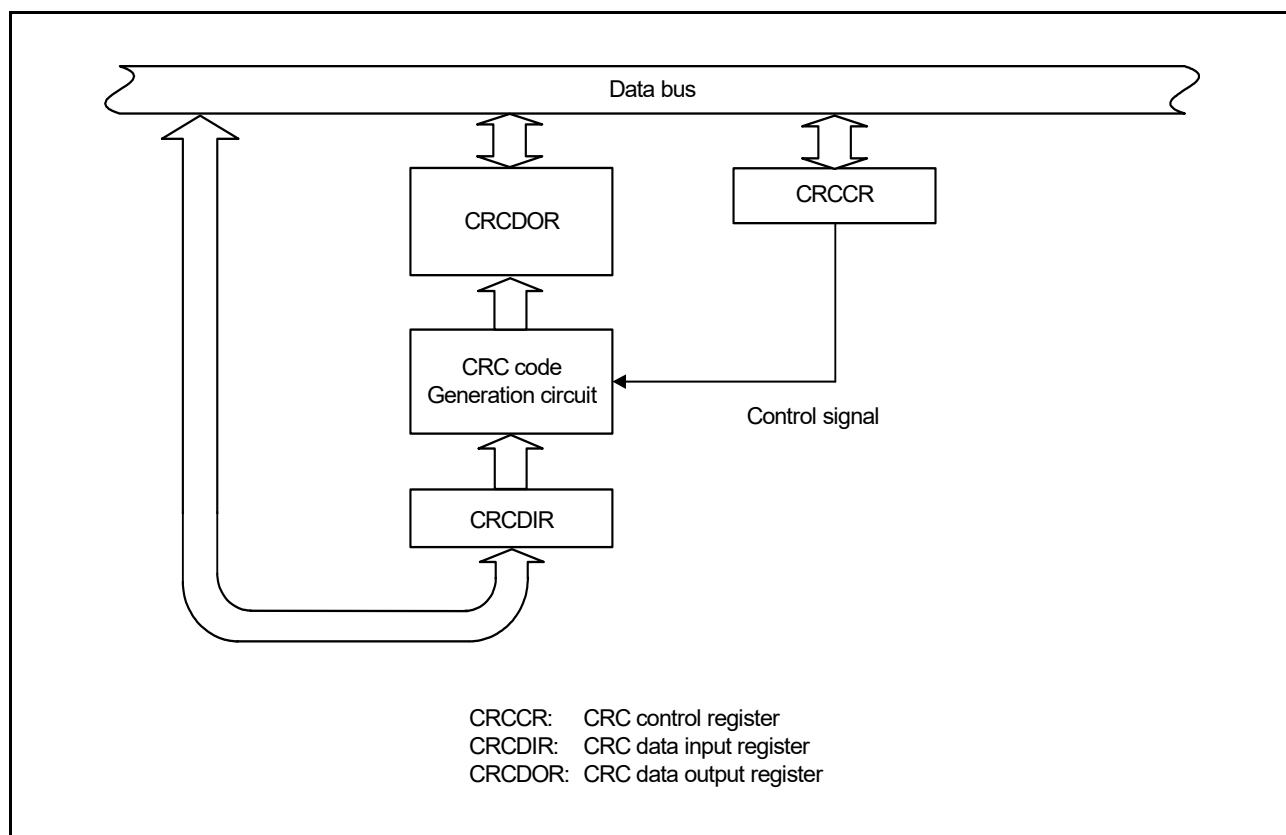


Figure 28.1 CRC Block Diagram

28.2 Register Descriptions

28.2.1 CRC Control Register (CRCCR)

Address(es): CRC.CRCCR 0008 8280h

b7	b6	b5	b4	b3	b2	b1	b0
DORCLR	LMS	—	—	—	GPS[2:0]		
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	GPS[2:0]	CRC Generating Polynomial Switching	b2 b0 0 0 0: No calculation is executed. 0 0 1: 8-bit CRC ($X^8 + X^2 + X + 1$) 0 1 0: 16-bit CRC ($X^{16} + X^{15} + X^2 + 1$) 0 1 1: 16-bit CRC ($X^{16} + X^{12} + X^5 + 1$) 1 0 0: 32-bit CRC ($X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$) 1 0 1: 32-bit CRC ($X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^8 + X^6 + 1$) 1 1 0: No calculation is executed. 1 1 1: No calculation is executed.	R/W
b5 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	LMS	CRC Calculation Switching	0: Generates CRC for LSB first communication. 1: Generates CRC for MSB first communication.	R/W
b7	DORCLR	CRCDOR Register Clear	1: Clears the CRCDOR register. This bit is read as 0.	W*1

Note 1. Only 1 can be written.

LMS Bit (CRC Calculation Switching)

The setting this bit selects the order of the bits of generated CRC codes. The bit selects transmission of the lower-order byte of the CRC code first for LSB first communication, or the higher-order byte first for MSB first communication. For details on the transmission and reception of CRC codes, refer to **section 28.3, Operation**.

DORCLR Bit (CRCDOR Register Clear)

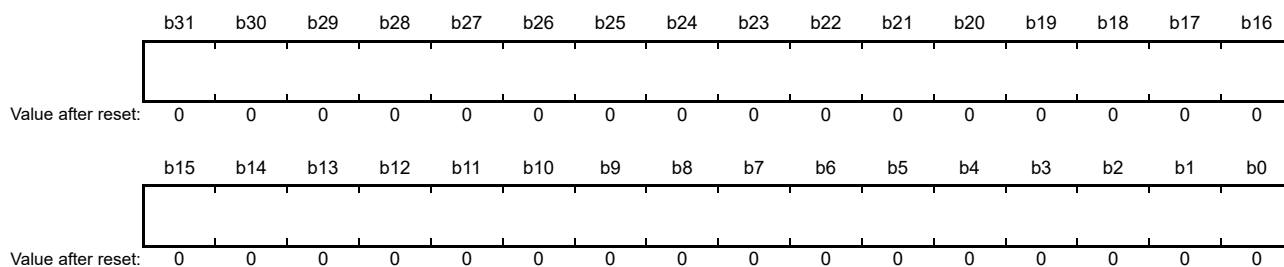
Write 1 to this bit so that the CRCDOR register is set to 0000 0000h.

This bit is read as 0. Writing 0 to this bit has no effect.

28.2.2 CRC Data Input Register (CRCDIR)

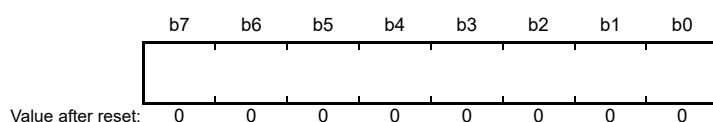
- When 32-bit CRC is selected

Address(es): CRC.CRCDIR 0008 8284h



- When 16-bit or 8-bit CRC is selected

Address(es): CRC.CRCDIR 0008 8284h



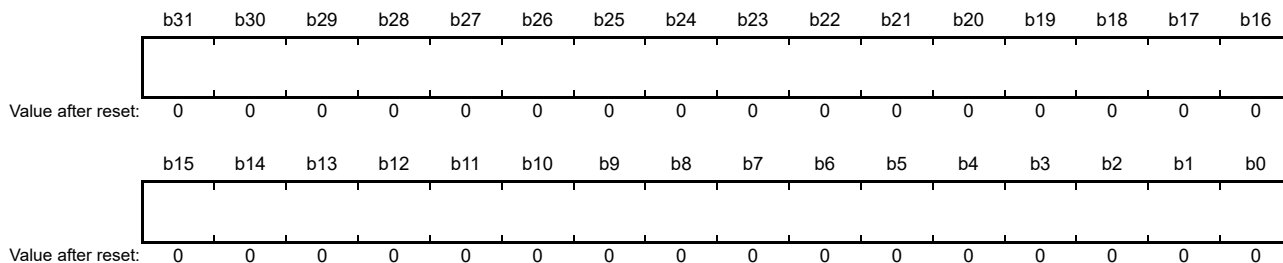
The CRCDIR register is a readable and writable register. Write data for CRC calculation to this register.

When generating a 32-bit CRC, the CRCDIR register should be accessed in longword units. When generating a 8-bit or 16-bit CRC, the CRCDIR register should be accessed in byte units.

28.2.3 CRC Data Output Register (CRCDOR)

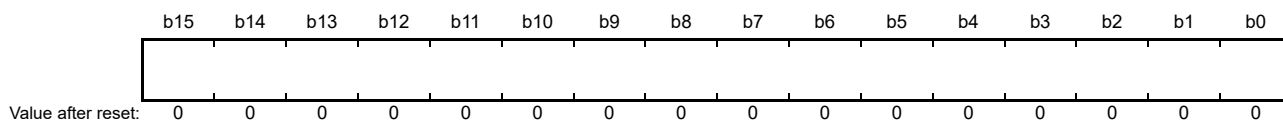
- When 32-bit CRC is selected

Address(es): CRC.CRCDOR 0008 8288h



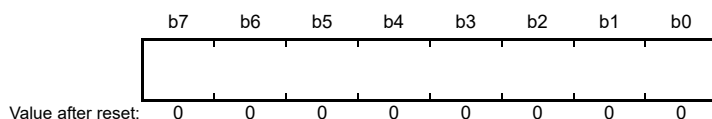
- When 16-bit CRC is selected

Address(es): CRC.CRCDOR 0008 8288h



- When 8-bit CRC is selected

Address(es): CRC.CRCDOR 0008 8288h



The CRCDOR register is a readable and writable register.

The value after a reset is 0000 0000h. When calculating with the initial value set to a value other than 0000 0000h, rewrite the CRCDOR register.

Writing data to the CRCDIR register stores result of calculation in the CRCDOR register. In addition, following communication data, when a CRC code is written to the CRCDIR register and if the calculation result is 0000 0000h, there is no error in the communication data.

When a 32-bit CRC is selected, the CRCDOR register should be accessed in longword units. When a 16-bit CRC is selected, the CRCDOR register should be accessed in word units. When an 8-bit CRC is selected, the CRCDOR register should be accessed in byte units.

28.3 Operation

The CRC calculator generates CRC codes for use in LSB first or MSB first communication.

The following shows examples of generating the CRC code for input data (F0h) using the 16-bit CRC generating polynomial ($X^{16} + X^{12} + X^5 + 1$). In these examples, the value of the CRC data output register (CRCDOR) is cleared by setting the DORCLR bit to 1 to set the initial value for CRC calculation to 0000 0000h.

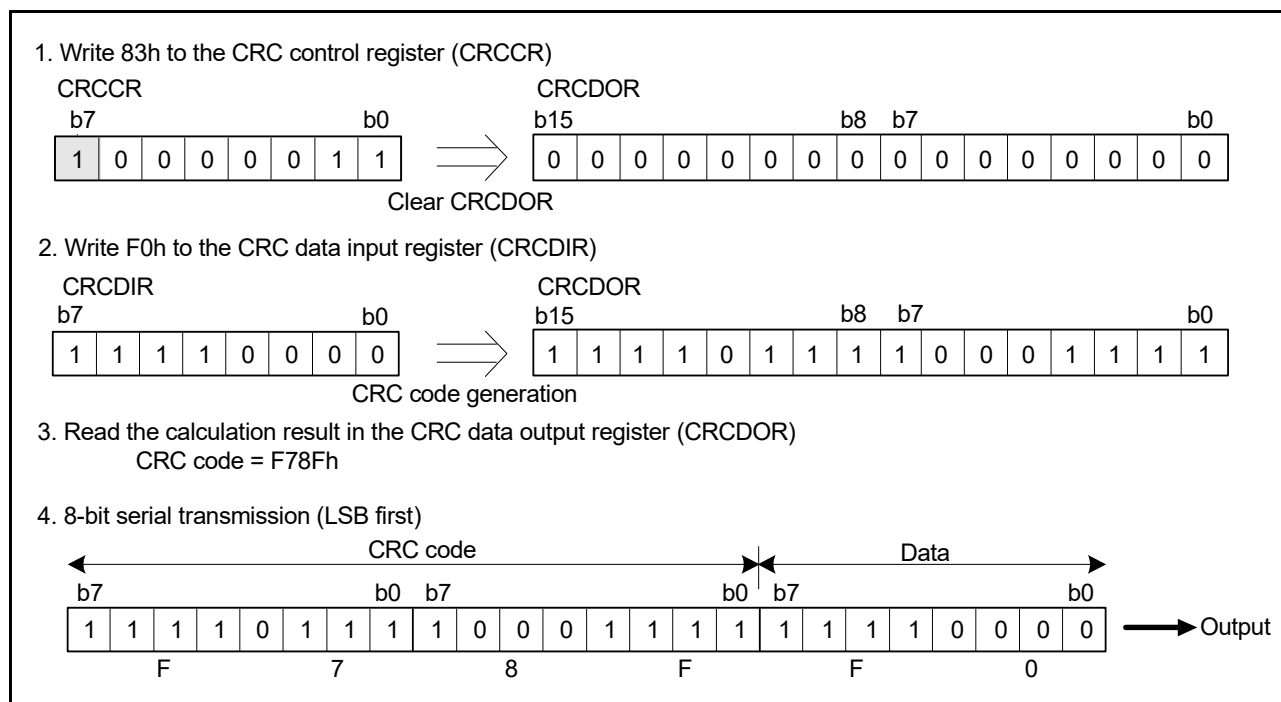


Figure 28.2 LSB First Data Transmission

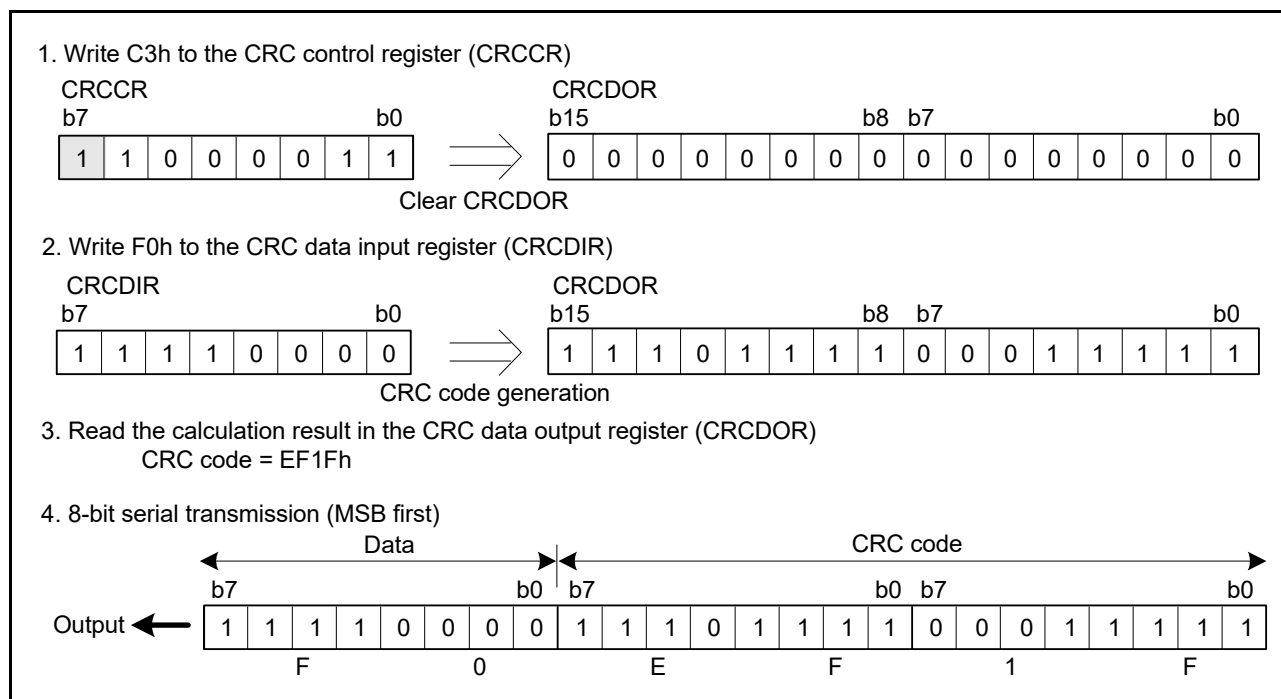


Figure 28.3 MSB First Data Transmission

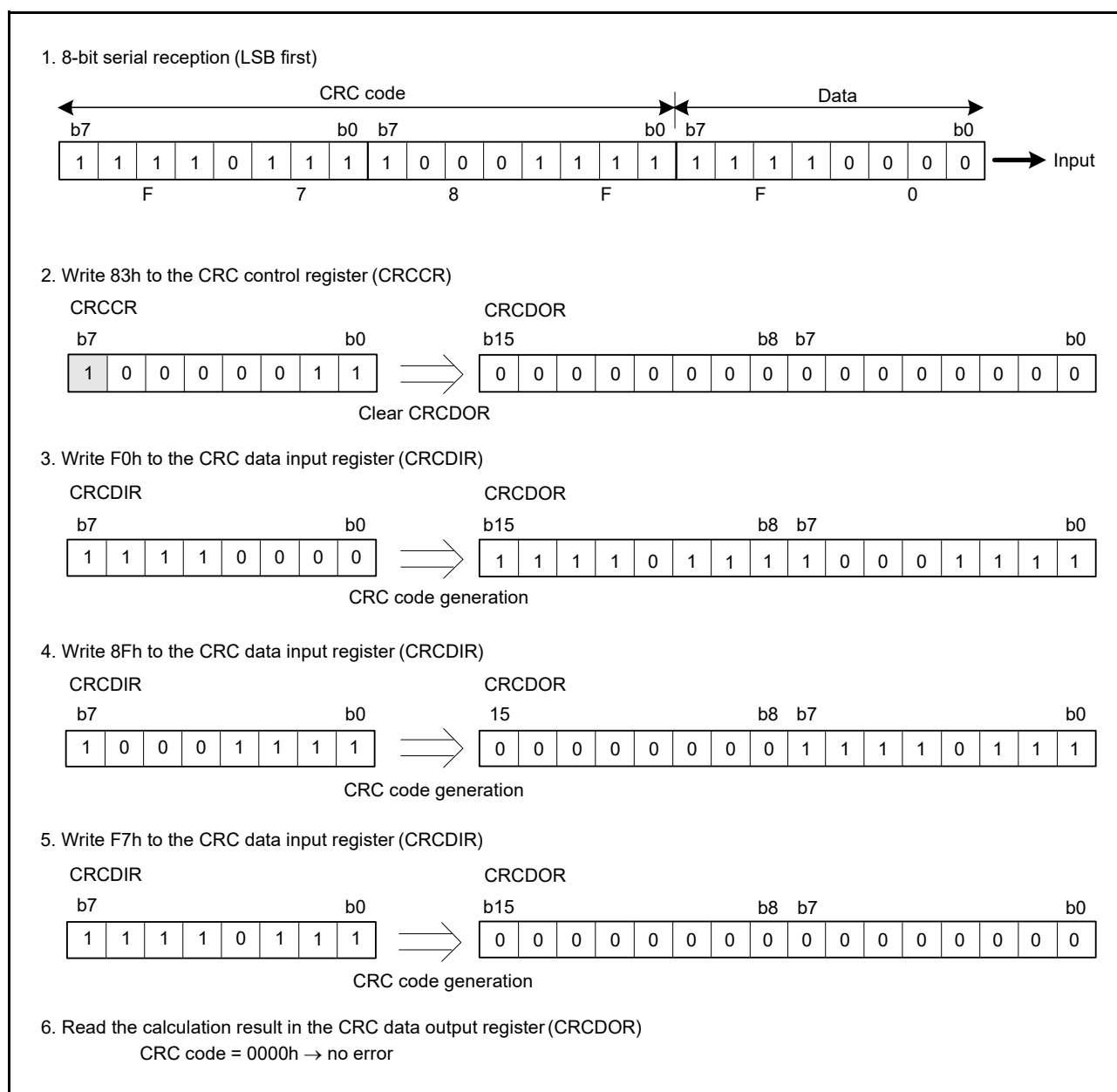


Figure 28.4 LSB First Data Reception

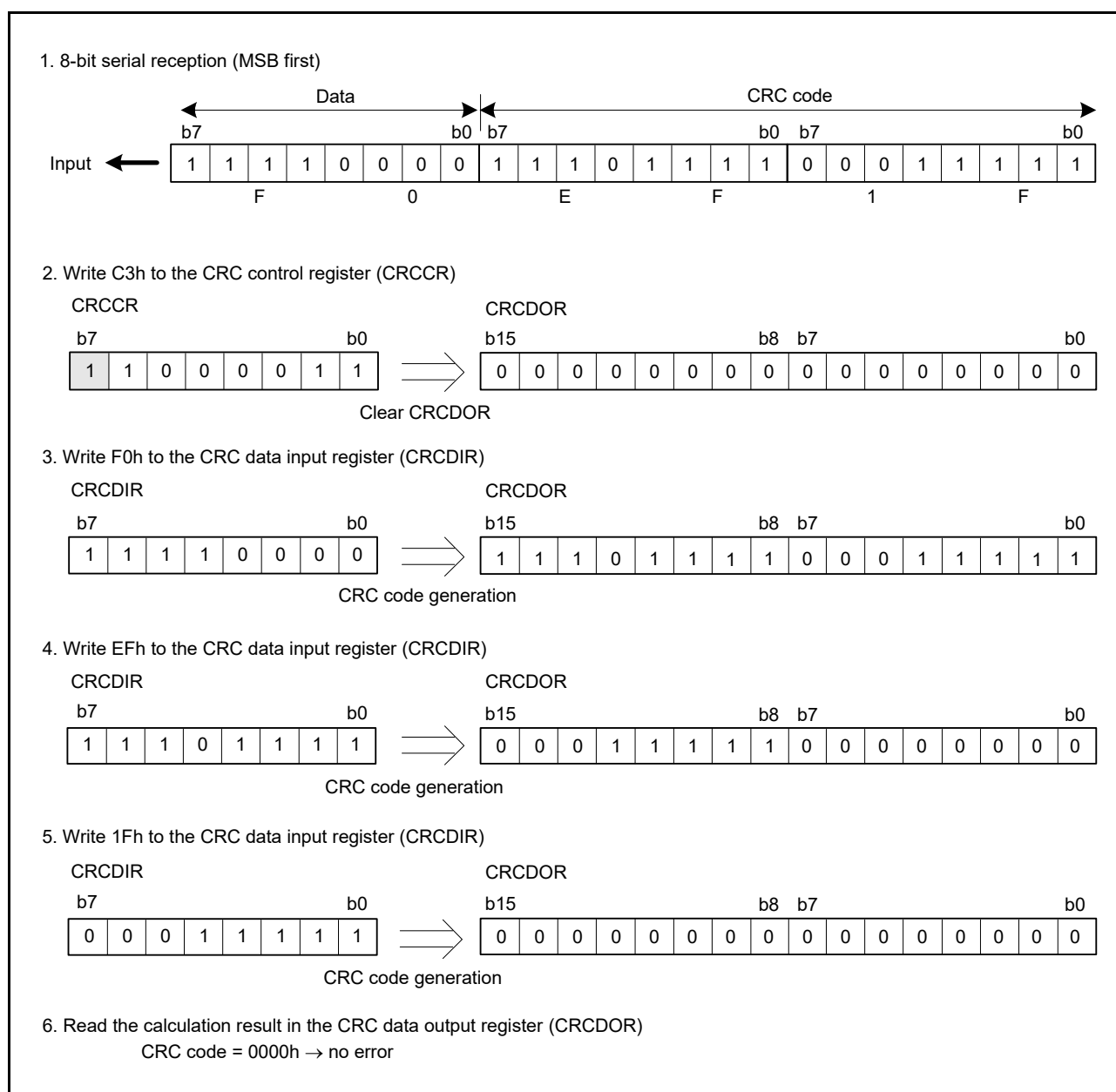


Figure 28.5 MSB First Data Reception

28.4 Usage Notes

28.4.1 Module Stop Function Setting

Operation of the CRC calculator can be disabled or enabled using the module stop control register B (MSTPCRB). After a reset, the CRC is in the module stop state. Register access is enabled by releasing the module stop state.

For details, refer to section 11, Low Power Consumption.

28.4.2 Note on Transmission

Note that the sequence of transmission for the CRC code differs according to whether transmission is LSB first or MSB first.

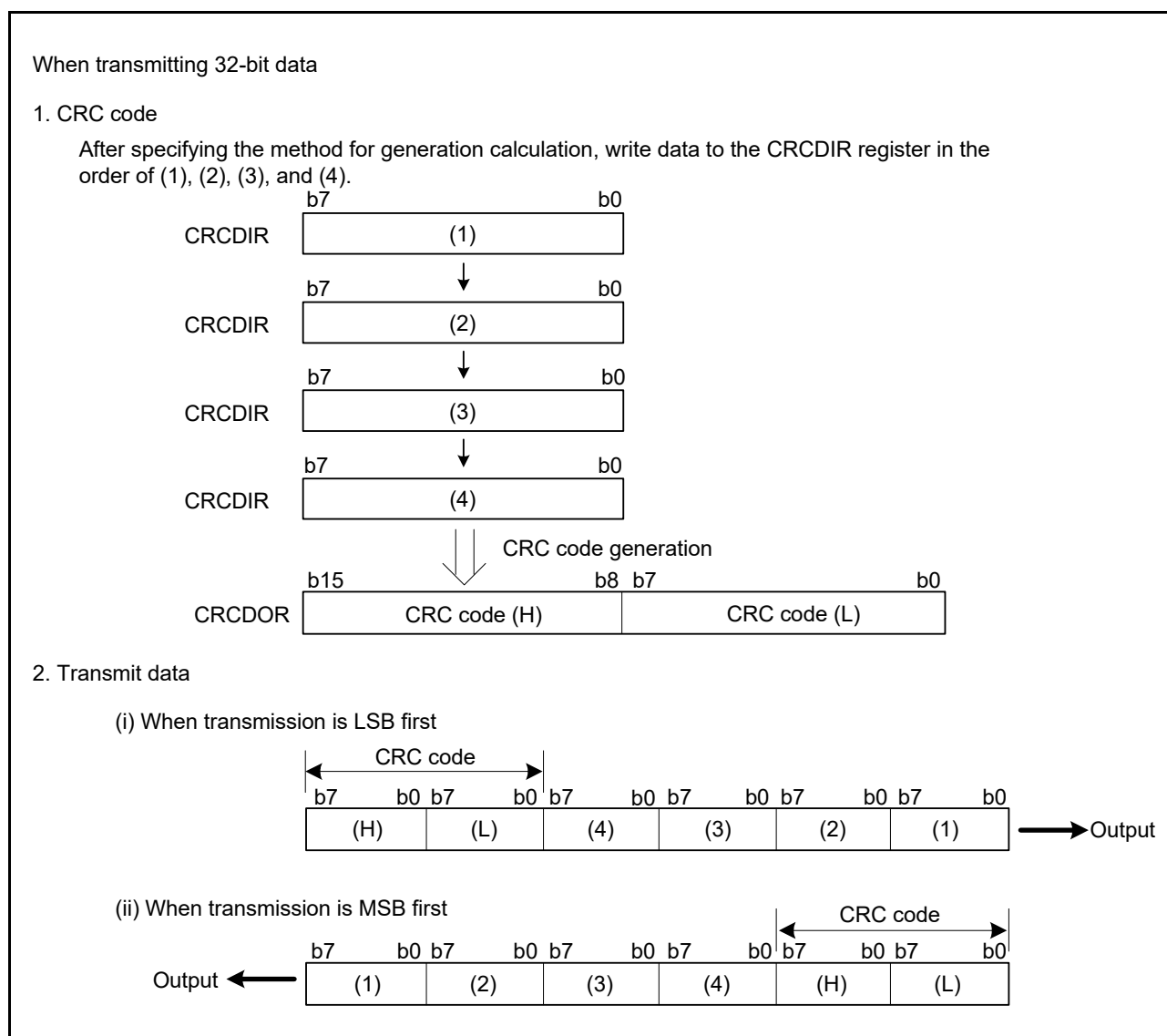


Figure 28.6 LSB First and MSB First Data Transmission

29. Arithmetic Unit for Trigonometric Functions (TFUv1)

This MCU has an arithmetic unit for trigonometric functions (TFU).

The TFU handles the high-speed calculation of $\sin$, $\cos$, $\tan$, and hypot functions.

An option must be specified to make the C or C++ compiler produce instructions that use the TFU. For details, refer to the manual for the compiler you are using.

The ICLK is used as the operating clock for the TFU.

29.1 Overview

Table 29.1 lists the specifications of the TFU.

Table 29.1 TFU specifications

Item	Description				
Arithmetic Processing	Calculation of sine, cosine, arctangent, and hypotenuse - A sine and cosine can be simultaneously calculated. - An arctangent and hypotenuse can be simultaneously calculated.				
Range and Unit of Values	Arithmetic Processing	I/O		Range	Unit
	Calculating sine	Input	Angle θ	$-\text{float\_max} \leq \theta \leq \text{float\_max}^{*1}$	radians
		Output	$\sin\theta$	$-1.0 \leq \sin\theta \leq 1.0$	—
	Calculating cosine	Input	Angle θ	$-\text{float\_max} \leq \theta \leq \text{float\_max}^{*1}$	radians
		Output	$\cos\theta$	$-1.0 \leq \cos\theta \leq 1.0$	—
	Calculating arctangent	Input	x and y coordinates	$-\text{float\_max} \leq x \leq \text{float\_max}^{*1}$ $-\text{float\_max} \leq y \leq \text{float\_max}^{*1}$	—
		Output	$\text{atan}(y/x)$	$-\pi \leq \text{atan}(y/x) \leq \pi$	radians
	Calculating hypotenuse	Input	x and y coordinates	$-\text{float\_max} \leq x \leq \text{float\_max}^{*1}$ $-\text{float\_max} \leq y \leq \text{float\_max}^{*1}$	—
		Output	$\sqrt{x^2 + y^2}$	$0 \leq \sqrt{x^2 + y^2} \leq \infty$	—
Data Type for Processing	Single-precision floating-point				
Number of cycles for calculation	Sine: 14 Cosine: 14 Arctangent: 14 Hypotenuse: 14 + 2 for software processing				

Note 1. float_max is the maximum value that can be expressed as single-precision floating-point: $(2 - 2^{-23}) \times 2^{127}$.

30. 12-Bit A/D Converter (S12ADF)

In this section, “PCLK” is used to refer to PCLKB, “ADCLK” is used to refer to PCLKD.

30.1 Overview

This MCU incorporates two units of 12-bit successive approximation A/D converters. Each unit can select eight channels of analog input. However, since AN001/PGAIN01 and PGAIN02 are connected to both units, the maximum number of analog inputs that can be used simultaneously is 14. Additionally, the temperature sensor output and internal reference voltage can only be selected on Unit 1.

The 12-bit A/D converter converts a maximum of 14 selected channels of analog inputs and temperature sensor output, or internal reference voltage (for unit 1 only) into a 12-bit digital value through successive approximation.

The A/D converter has three operating modes: single scan mode in which the analog inputs of eight arbitrarily selected channels are converted in ascending channel order; continuous scan mode in which the analog inputs of eight arbitrarily selected channels are continuously converted in ascending channel order; and group scan mode in which eight arbitrarily selected channels are arbitrarily divided into two groups (groups A and B) or three groups (groups A, B, and C) and converted in ascending channel order in each group.

In group scan mode, either two groups (groups A and B) or three groups (groups A, B, and C) is selected.

The conditions for scanning start of each group (A and B or A, B, and C) (synchronous trigger) can be independently selected, thus allowing A/D conversion of each group to be started independently.

During group priority operation, in addition to the above-mentioned operation, a trigger to start scanning for the priority group is accepted during scan for the low-priority group, and scan for the priority group is started after scan for the low-priority group was discontinued. The priority order is group A > group B > group C. Accordingly, as priority operation, when a trigger to start scanning for group B is accepted during scan for group C, group C scan is discontinued, and scan for group B is started. Likewise, when a trigger to start scanning for group A is accepted during scan for group C, group C scan is discontinued and scan for group A is started. In the same way, when a trigger to start scanning for group A is accepted during scan for group B, group B scan is discontinued and scan for group A is started. Note, however, that only the unit 1 can be selected from the temperature sensor output or internal reference voltage.

The discontinued scan operation can be restarted after the scanning of the priority group is completed.

In double trigger mode, one analog input channel arbitrarily selected is converted in single scan mode or group scan mode (group A), and the resulting data of A/D conversion started by the first and second synchronous triggers are stored into different registers (duplication of A/D conversion data).

The temperature sensor output or internal reference voltage cannot be selected in continuous scan mode.

Self-diagnosis is executed once at the beginning of each scan, and one of the three voltages internally generated in the 12-bit A/D converter is converted.

Table 30.1 lists the specifications of the 12-bit A/D converter and Table 30.2 lists the functions of the 12-bit A/D converter. Figure 30.1 show block diagrams of the 12-bit A/D converter.

For the correspondence between each unit of the A/D converter, analog input, and the storage destination of the A/D conversion results, please refer to section 30.3, Operation.

Table 30.1 Specifications of 12-Bit A/D Converter (1/2)

Item	Description
Number of units	Two units (S12AD, S12AD1)
Input channels	Eight channels for S12AD, eight channels for S12AD1 (AN001/PGAIN01 and PGAIN02 are common)
Extended analog function	Temperature sensor output, internal reference voltage (S12AD1 only)
A/D conversion method	Successive approximation method
Resolution	12 bits
Conversion time	0.5 μ s per channel (when A/D conversion clock ADCLK = 64 MHz)
A/D conversion clock	Peripheral module clock PCLK*1 and A/D conversion clock ADCLK*1 can be set so that the frequency ratio should be one of the following. PCLK to ADCLK frequency ratio = 1:1, 1:2, 2:1, 4:1, 8:1 ADCLK is set using the clock generation circuit.
Data registers	16 registers for analog input (Eight for S12AD, eight for S12AD1), 1 for A/D-converted data duplication in double trigger mode, and 2 for A/D-converted data duplication during extended operation in double trigger mode unit. - One register for temperature sensor (S12AD1) - One register for internal reference (S12AD1) - One register for self-diagnosis per unit - The results of A/D conversion are stored in 12-bit A/D data registers. 12-bit accuracy output for the results of A/D conversion - The value obtained by adding up A/D-converted results is stored as a value in the number of bit for conversion accuracy + 2 bits/4 bits*2 in the A/D data registers in A/D-converted value addition mode. - Double trigger mode (selectable in single scan and group scan modes): The first piece of A/D-converted analog-input data on one selected channel is stored in the data register for the channel, and the second piece is stored in the duplication register. - Extended operation in double trigger mode (available for specific triggers): A/D-converted analog-input data on one selected channel is stored in the duplication register that is prepared for each type of trigger.
Operating modes	Operating modes can be set independently for two units. - Single scan mode: A/D conversion is performed only once on the analog inputs arbitrarily selected. A/D conversion is performed only once on the temperature sensor output (S12AD1). A/D conversion is performed only once on the internal reference voltage (S12AD1). - Continuous scan mode: A/D conversion is performed repeatedly on the analog inputs arbitrarily selected. - Group scan mode: Two (groups A and B) or three (groups A, B, and C) can be selected as the number of the groups to be used. Only the combination of groups A and B can be selected when the number of the groups is two. Analog inputs arbitrarily selected are divided into two groups (group A and B) or three groups (group A, B, and C), and A/D conversion of the analog input selected on a group basis is performed only once. The conditions for scanning start of groups A, B, and C (synchronous trigger) can be independently selected, thus allowing A/D conversion of each group to be started independently. - Group scan mode (when group priority control selected): If a priority-group trigger is input during scanning of the low-priority group, scan of the low-priority group is stopped and scan of the priority group is started. The priority order is group A (highest) > group B > group C (lowest). Whether or not to restart scanning of the low-priority group after processing for the high-priority group completes, is selectable. Rescan can also be set to start either from the beginning of the selected channel or the channel on which A/D conversion is not completed.
Conditions for A/D conversion start	- Software trigger - Synchronous trigger Trigger by the multi-function timer pulse unit (MTU), general PWM timer (GPTW), or 8-bit timer (TMR). - Asynchronous trigger A/D conversion can be triggered by the external trigger ADTRG0# (S12AD), ADTRG1# (S12AD1) pin (independently for two units).

Table 30.1 Specifications of 12-Bit A/D Converter (2/2)

Item	Description
Functions	- Input signal amplification function of the programmable gain amplifier (3 channels for S12AD, 2 channels for S12AD1 (AN001/PGAIN01 and PGAIN02 are common)) - Variable sampling time (settable for each channel) - Self-diagnosis of 12-bit A/D converter - Selectable A/D-converted value addition mode or average mode - Analog input disconnection detection assist function (discharge function/precharge function) - Double trigger mode (duplication of A/D conversion data) - Automatic clear function of A/D data registers
Interrupt sources	- In the modes except double trigger mode and group scan mode, a scan end interrupt request (S12ADI, S12ADI1) can be generated on completion of single scan (independently for two units). - In double trigger mode, a scan end interrupt request (S12ADI, S12ADI1) can be generated on completion of double scan (independently for two units). - In group scan mode, a scan end interrupt request (S12ADI, S12ADI1) can be generated on completion of group A scan, whereas a scan end interrupt request (GBADI, GBADI1) for group B can be generated on completion of group B scan, and a group C scan end interrupt request (GCADI, GCADI1) can be generated on completion of group C scan. - When double trigger mode is selected in group scan mode, an A/D scan end interrupt request (S12ADI, S12ADI1) can be generated on completion of double scan of group A, and the corresponding scan end interrupt request (GBADI/GCADI, GBADI1/GCADI1) can be generated on completion of group B and group C scan. - The S12ADI/S12ADI1, GBADI/GBADI1, and GCADI/GCADI1 interrupts can activate the data transfer controller (DTC).
Low power consumption function	Module stop state can be set. *3, *4

Note 1. The frequency of PCLK, the peripheral module clock, becomes that set in the SCKCR.PCKB[3:0] bits, and that of ADCLK, the A/D conversion clock, becomes the frequency set in the SCKCR.PCKD[3:0] bits.

Note 2. The number of extended bits during addition differs depending on the addition count.
 2-bit extension: 1-time to 4-time conversion (add zero to three times) 4-bit extension: 16-time conversion (add 15 times)

Note 3. Refer to section 11, Low Power Consumption for details.

Note 4. Wait for 1 μs or longer to start A/D conversion after release from the module stop state.

Table 30.2 Functions of 12-Bit A/D Converter (1/2)

Item			Pin Name, Abbreviation	
			Unit 0 (S12AD)	Unit 1 (S12AD1)
Analog input channels			AN000/PGAIN00 to AN002/PGAIN02, AN003 to AN007	AN001/PGAIN01, PGAIN02, AN102 to AN107
Conditions for A/D conversion start	Software	Software trigger	Enabled	
	Asynchronous trigger	Trigger input pin	ADTRG0#	ADTRG1#
	Synchronous trigger	Compare match/input capture from MTU0.TGRA	TRGA0N	
		Compare match/input capture from MTU1.TGRA	TRGA1N	
		Compare match/input capture from MTU2.TGRA	TRGA2N	
		Compare match/input capture from MTU3.TGRA	TRGA3N	
		Compare match/input capture from MTU4.TGRA or underflow (trough) of MTU4.TCNT in complementary PWM mode	TRGA4N	
		Compare match from MTU0.TGRE	TRG0N	
		Compare match between MTU4.TADCORA and MTU4.TCNT	TRG4AN	
		Compare match between MTU4.TADCORB and MTU4.TCNT	TRG4BN	
		Compare match between MTU4.TADCORA and MTU4.TCNT, or compare match between MTU4.TADCORB and MTU4.TCNT	TRG4AN or TRG4BN	
		Compare match between MTU4.TADCORA and MTU4.TCNT, and compare match between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is used)	TRG4ABN	
		Compare match in GPTW0.GTADTRA	GTADTRA0N	
		Compare match in GPTW0.GTADTRB	GTADTRB0N	
		Compare match in GPTW1.GTADTRA	GTADTRA1N	
		Compare match in GPTW1.GTADTRB	GTADTRB1N	
		Compare match in GPTW2.GTADTRA	GTADTRA2N	
		Compare match in GPTW2.GTADTRB	GTADTRB2N	
		Compare match in GPTW0.GTADTRA, or compare match in GPTW0.GTADTRB	GTADTRA0N or GTADTRB0N	
		Compare match in GPTW1.GTADTRA, or compare match in GPTW1.GTADTRB	GTADTRA1N or GTADTRB1N	
		Compare match in GPTW2.GTADTRA, or compare match in GPTW2.GTADTRB	GTADTRA2N or GTADTRB2N	
		Compare match/input capture from MTU0.TGRA, or compare match in MTU0.TGRE	TRGA0N or TRG0N	
		Compare match in TMR0.TCORA, and compare match in TMR0.TCNT	TMTRG0AN_0	
		Compare match in TMR2.TCORA, and compare match in TMR2.TCNT	TMTRG0AN_1	
		Compare match/input capture from MTU0.TGRA, and compare match in MTU0.TGRE	TRG0AEN	
		Compare match in GPTW0.GTADTRA, or compare match in GPTW0.GTADTRB, or compare match in GPTW1.GTADTRA, or compare match in GPTW1.GTADTRB, or compare match in GPTW2.GTADTRA, or compare match in GPTW2.GTADTRB	GTADTRA0N or GTADTRB0N or GTADTRA1N or GTADTRB1N or GTADTRA2N or GTADTRB2N	

Table 30.2 Functions of 12-Bit A/D Converter (2/2)

Item			Pin Name, Abbreviation	
			Unit 0 (S12AD)	Unit 1 (S12AD1)
Conditions for A/D conversion start	Synchronous trigger	Compare match in GPTW0.GTADTRA, or compare match in GPTW0.GTADTRB, or compare match in GPTW1.GTADTRA, or compare match in GPTW1.GTADTRB	GTADTRA0N or GTADTRB0N or GTADTRA1N or GTADTRB1N	
PGA		Target channels	AN000 to AN002	AN001 to AN002
Interrupts			S12ADI, GBADI, GCADI interrupt	S12ADI1, GBADI1, GCADI1 interrupt
Setting of module stop function*1, *2			MSTPCRA.MSTPA17 bit	MSTPCRA.MSTPA16 bit

Note: When setting an A/D conversion start trigger to ADTRG0#, ADTRG1#, set the pin mode control bit in the port mode register for the corresponding pin to 1 (peripheral functions), and set the pin function select bit in the pin function control register to ADTRG0#, ADTRG1#. Refer to section 17, I/O Ports for details.

Note 1. Refer to section 11, Low Power Consumption for details.

Note 2. Wait for 1 μ s or longer to start A/D conversion after release from the module stop state.

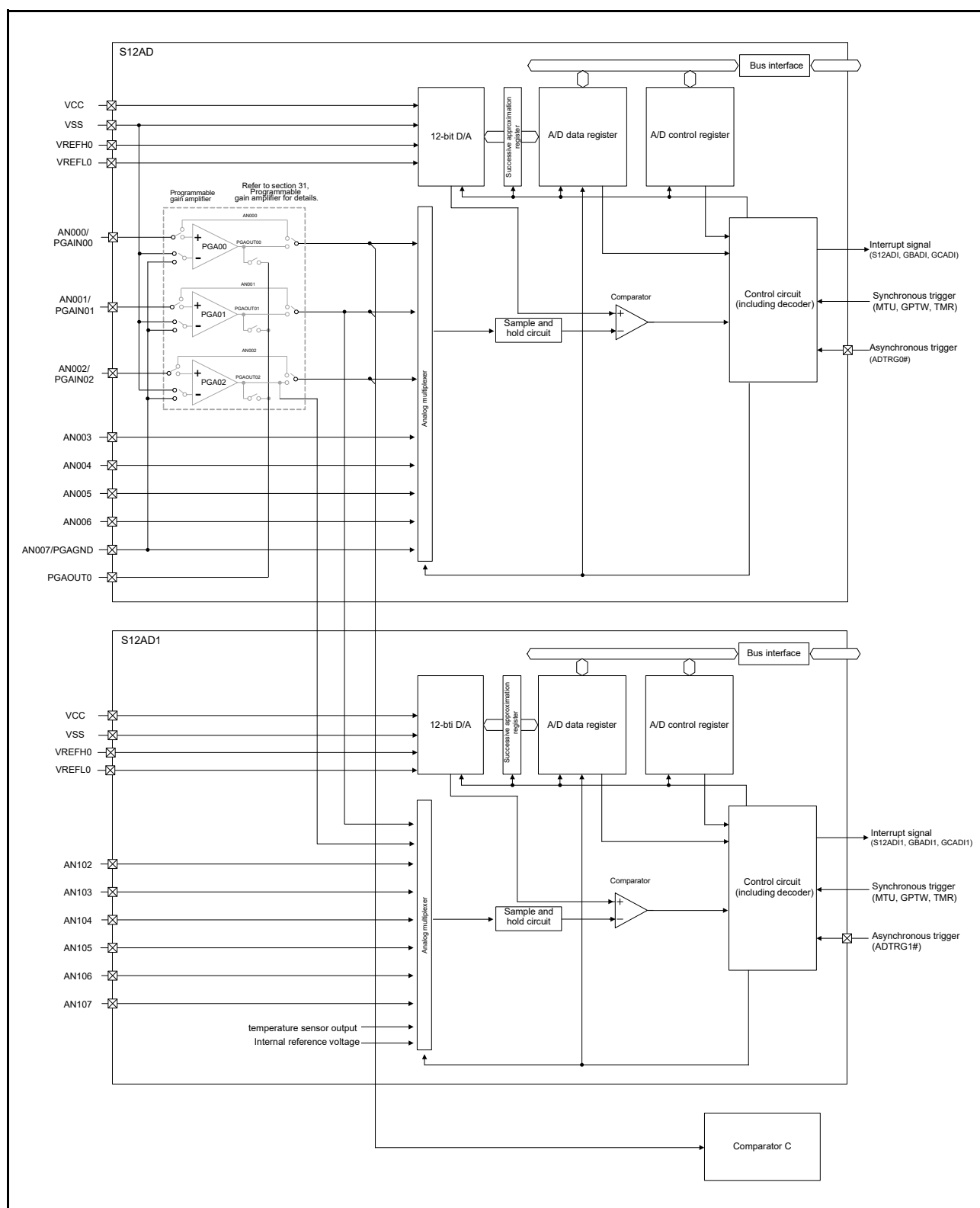


Figure 30.1 Block Diagram of 12-Bit A/D Converter

Table 30.3 lists the input/output pins of the 12-bit A/D converter.

The 12-bit A/D converter consists of two units, unit 0 (S12AD), unit 1 (S12AD1). These units can be operated independently. The input channels of S12AD, S12AD1 can be divided into three groups for operation. A programmable gain amplifier (PGA) is included in AN000 to AN002.

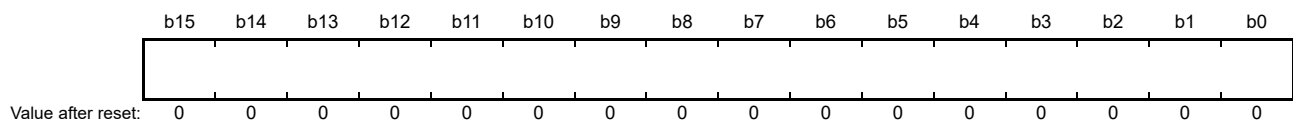
Table 30.3 Input/Output Pins of 12-Bit A/D Converter

Unit	Pin Name	I/O	Function
Unit 0 (S12AD)	AN000	Input	Analog input pin
	AN001	Input	Analog input pin
	AN002	Input	Analog input pin
	AN003	Input	Analog input pin
	AN004	Input	Analog input pin
	AN005	Input	Analog input pin
	AN006	Input	Analog input pin
	AN007	Input	Analog input pin
	ADTRG0#	Input	External trigger input pin for starting A/D conversion
	ADST0	Output	ADST bit state output pin
	ADSS0	Output	Sampling state output pin
Unit 1 (S12AD1)	AN001	Input	Analog input pin
	AN102	Input	Analog input pin
	AN103	Input	Analog input pin
	AN104	Input	Analog input pin
	AN105	Input	Analog input pin
	AN106	Input	Analog input pin
	AN107	Input	Analog input pin
	ADTRG1#	Input	External trigger input pin for starting A/D conversion
	ADST1	Output	ADST bit state output pin
	ADSS1	Output	Sampling state output pin
Common to unit 0 and unit1	VCC	—	Power supply pin
	VSS	—	Ground pin
	VREFH0	—	Reference power supply pin
	VREFL0	—	Reference power supply ground pin

30.2 Register Descriptions

30.2.1 A/D Data Registers y (ADDRy) (y = 0 to 7), A/D Data Duplication Register (ADDBLDR), A/D Data Duplication Register A (ADDBLDRA), A/D Data Duplication Register B (ADDBLDRB), A/D Temperature Sensor Data Register (ADTSDR), A/D Internal Reference Voltage Data Register (ADOCDR)

Address(es): S12AD.ADDR0 0008 9020h, S12AD.ADDR1 0008 9022h, S12AD.ADDR2 0008 9024h,
S12AD.ADDR3 0008 9026h, S12AD.ADDR4 0008 9028h, S12AD.ADDR5 0008 902Ah,
S12AD.ADDR6 0008 902Ch, S12AD.ADDR7 0008 902Eh, S12AD.ADDBLDR 0008 9018h,
S12AD.ADDBLDRA 0008 9084h, S12AD.ADDBLDRB 0008 9086h, S12AD1.ADDR0 0008 9220h,
S12AD1.ADDR1 0008 9222h, S12AD1.ADDR2 0008 9224h, S12AD1.ADDR3 0008 9226h,
S12AD1.ADDR4 0008 9228h, S12AD1.ADDR5 0008 922Ah, S12AD1.ADDR6 0008 922Ch,
S12AD1.ADDR7 0008 922Eh, S12AD1.ADDBLDR 0008 9218h, S12AD1.ADDBLDRA 0008 9284h,
S12AD1.ADDBLDRB 0008 9286h, S12AD1.ADTSDR 0008 921Ah, S12AD1.ADOCDR 0008 921Ch



The ADDRy registers (y = 0 to 7) are 16-bit read-only registers which store the A/D conversion results.

The ADDBLDR register is a 16-bit read-only register used in double trigger mode. The ADDBLDR register stores the results of A/D conversion when the conversion is started by the second trigger.

The ADDBLDRA and ADDBLDRB registers are 16-bit read-only registers that store the A/D conversion results in response to the respective triggers during extended operation in double trigger mode.

The ADTSDR register is a 16-bit read-only register that stores the A/D-converted value of the temperature sensor output.

The ADOCDR register is a 16-bit read-only register that stores the A/D conversion results of the internal reference voltage.

The format of each register differs depending on the conditions below.

- Settings of the A/D data register format select bit (ADCER.ADRFMT) (flush-right or flush-left)
- Settings of the addition count select bits (ADADC.ADC[2:0]) (2-, 3-, 4-, or 16-time conversion)
- Settings of the average mode enable bit (ADADC.AVEE) (add or average)

The data formats for each given condition are shown below.

(1) When A/D-Converted Value Addition/Average Mode is Not Selected

- Flush-right format
The A/D-converted value is stored in bits 11 to 0. Bits 15 to 12 are read as 0.
- Flush-left format
The A/D-converted value is stored in bits 15 to 4. Bits 3 to 0 are read as 0.

(2) When A/D-Converted Average Mode is Selected

- Flush-right format
The mean value of the A/D-converted results of the same channel is stored in bits 11 to 0. Bits 15 to 12 are read as 0.
- Flush-left format
The mean value of the A/D-converted results of the same channel is stored in bits 15 to 4. Bits 3 to 0 are read as 0.

A/D-converted value average mode can be set only when 2- or 4-time conversion is selected in A/D-converted value addition mode.

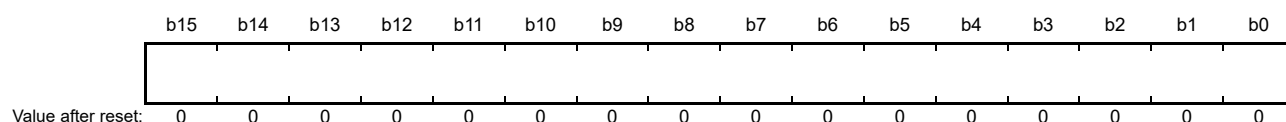
(3) When A/D-Converted Value Addition Mode is Selected

- Flush-right format (in A/D-converted value addition mode and when number of conversions is selected for 1 to 4 times)
The value added by the A/D-converted value of the same channel is stored in bits 13 to 0.
Bits 15 and 14 are read as 0.
- Flush-right format (in A/D-converted value addition mode and when number of conversions is selected for 16 times)
The value added by the A/D-converted value of the same channel is stored in bits 15 to 0.
- Flush-left format (in A/D-converted value addition mode and when number of conversions is selected for 1 to 4 times)
The value added by the A/D-converted value of the same channel is stored in bits 15 to 2.
Bits 1 and 0 are read as 0.
- Flush-left format (in A/D-converted value addition mode and when number of conversions is selected for 16 times)
The value added by the A/D-converted value of the same channel is stored in bits 15 to 0.

When A/D-converted addition mode is selected, the value added by the A/D-converted value of the same channel is indicated. The number of A/D conversions can be set to 1, 2, 3, 4, or 16 times. If A/D-converted addition mode is selected, when the conversion count is set to 1 to 4 times, the value added by the A/D conversion result is retained in the A/D data register as 2-bit extended data of the conversion accuracy bits; when the conversion count is set to 16 times, the value added by the A/D conversion result is retained in the A/D data register as 4-bit extended data of the conversion accuracy bits. Even if A/D-converted value addition mode is selected, the value is stored in the A/D data register according to the settings of the A/D data register format select bits.

30.2.2 A/D Self-Diagnosis Data Register (ADRD)

Address(es): S12AD.ADRD 0008 901Eh, S12AD1.ADRD 0008 921Eh



ADRD is a 16-bit read-only register that stores the A/D conversion results based on the 12-bit A/D converter's self-diagnosis. In addition to the A/D-converted value, the self-diagnosis status is included in. In the ADRD register, the different formats are used depending on the conditions below.

- Settings of the A/D data register format select bit (ADCER.ADRFMT) (flush-right or flush-left)

The A/D-converted value addition mode and A/D-converted value average mode cannot be applied to the A/D self-diagnosis function. For details of self-diagnosis, refer to section 30.2.9, A/D Control Extended Register (ADCER).

The data formats for each given condition are shown below.

- Flush-right format
The A/D-converted value is stored in bits 11 to 0. The self-diagnosis status is stored in bits 15 and 14.
Bits 13 and 12 are read as 0.
- Flush-left format
The A/D-converted value is stored in bits 15 to 4. The self-diagnosis status is stored in bits 1 and 0.
Bits 3 and 2 are read as 0.

Table 30.4 Self-Diagnosis Status Description

Bits 15 and 14 for flush-right format setting Bits 1 and 0 for flush-left format setting	Self-diagnosis status
00b	Self-diagnosis has never been executed since power-on.
01b	Self-diagnosis using the voltage of 0 V has been executed.
10b	Self-diagnosis using the voltage of reference power supply $\times 1/2$ has been executed.
11b	Self-diagnosis using the voltage of reference power supply has been executed.

Note: For details of self-diagnosis, refer to section 30.2.9, A/D Control Extended Register (ADCER).

30.2.3 A/D Control Register (ADCSR)

Address(es): S12AD.ADCSR 0008 9000h, S12AD1.ADCSR 0008 9200h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
ADST	ADCS[1:0]	ADIE	—	—	TRGE	EXTRG	DBLE	GBADIE	—	DBLANS[4:0]					
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	DBLANS[4:0]	Double Trigger Channel Select	These bits select one analog input channel for double triggered operation. The setting is only effective while double trigger mode is selected.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	GBADIE	Group B Scan End Interrupt Enable	0: Disables interrupt generation upon group B scan completion. 1: Enables interrupt generation upon group B scan completion.	R/W
b7	DBLE	Double Trigger Mode Select	0: Deselects double trigger mode. 1: Selects double trigger mode.	R/W
b8	EXTRG	Trigger Select*1	0: A/D conversion is started by synchronous trigger. 1: A/D conversion is started by asynchronous trigger.	R/W
b9	TRGE	Trigger Start Enable	0: Disables A/D conversion to be started by synchronous or asynchronous trigger. 1: Enables A/D conversion to be started by synchronous or asynchronous trigger.	R/W
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	ADIE	Scan End Interrupt Enable	0: Disables interrupt generation upon scan completion. 1: Enables interrupt generation upon scan completion.	R/W
b14, b13	ADCS[1:0]	Scan Mode Select	b14 b13 0 0: Single scan mode 0 1: Group scan mode 1 0: Continuous scan mode 1 1: Setting prohibited	R/W
b15	ADST	A/D Conversion Start	0: Stops A/D conversion process. 1: Starts A/D conversion process.	R/W

Note 1. Starting A/D conversion using an external pin (asynchronous trigger)

Set the ADCSR.TRGE and EXTRG bits to 1 while a high-level signal is input to the external pin (ADTRG0#, ADTRG1#). Then, if the ADTRG0#, ADTRG1# signal is changed to low, the falling edge is detected and the scan process is started. In this case, the pulse width of the low-level input must be at least 1.5 clock cycles of PCLK.

The relationship between each unit and the external pin (asynchronous trigger) is shown below.

Unit	External pin (asynchronous trigger)
S12AD	ADTRG0#
S12AD1	ADTRG1#

ADCSR sets double trigger mode, A/D conversion start trigger; enables/disables scan end interrupt; selects the scan mode; and starts or stops A/D conversion.

DBLANS[4:0] Bits (Double Trigger Channel Select)

The DBLANS[4:0] bits select one of the channels for A/D conversion data duplication in double trigger mode. The A/D conversion results of the analog input of the channel selected by the DBLANS[4:0] bits are stored into the A/D data register y when conversion is started by the first and every subsequent odd-numbered time trigger, and into the A/D data duplication register when started by the second and every subsequent even-numbered time trigger. Table 30.5 shows selection of the channel for double triggered operation.

When double trigger mode is selected, the channels selected by the ADANSA0 register are invalid, and the channel selected by the DBLANS[4:0] bits is subjected to A/D conversion instead.

The DBLANS[4:0] bits should be set while the ADST bit is 0. They should not be set simultaneously when 1 is written to the ADST bit.

Table 30.5 Relationship between DBLANS[4:0] Bits Settings and Double Trigger Enabled Channels

S12AD (Unit 0)		S12AD1 (Unit 1)	
DBLANS[4:0]	Duplication Channel	DBLANS[4:0]	Duplication Channel
00000b	AN000/PGAIN00	00000b	AN001/PGAIN01
00001b	AN001/PGAIN01	00001b	PGAIN02
00010b	AN002/PGAIN02	00010b	AN102
00011b	AN003	00011b	AN103
00100b	AN004	00100b	AN104
00101b	AN005	00101b	AN105
00110b	AN006	00110b	AN106
00111b	AN007	00111b	AN107

Note: Duplication cannot be selected for the A/D conversion data of self-diagnosis, temperature sensor output, and internal reference voltage.

GBADIE Bit (Group B Scan End Interrupt Enable)

The GBADIE bit enables or disables group B scan end interrupt in group scan mode.

A scan end interrupt for group B is provided individually for each unit. Table 30.6 shows the relationship between each unit and the scan end interrupt for group B.

Table 30.6 Relationship between Each Unit and Group B Scan End Interrupt

Unit	Group B Scan End Interrupt
S12AD	GBADI
S12AD1	GBADI1

DBLE Bit (Double Trigger Mode Select)

Double trigger mode has a function to store the resulting data of A/D conversion started by the first and every subsequent odd-numbered time, and second and every subsequent even-numbered time synchronous triggers into separate registers. When double trigger mode is selected, the channels specified in the ADANSA0 register are invalid and the channel selected by the DBLANS[4:0] bits is effective instead. Double trigger mode can be only operated by the synchronous trigger selected by the ADSTRGR.TRSA[5:0] bits. Do not generate an asynchronous or software trigger. The A/D conversion results started by the first and every subsequent odd-numbered time trigger are stored into the A/D data register y and those started by the second and every subsequent even-numbered time trigger are stored into the A/D data duplication register. In this case, if the ADIE bit is set to 1, the interrupt is generated not upon completion of the first and every subsequent odd-numbered time conversion but upon completion of the second and every subsequent even-numbered time conversion.

In continuous scan mode, double trigger mode should not be selected. In addition, double trigger mode should not be used for self-diagnosis, conversion of the temperature sensor output, or conversion of the internal reference voltage. The DBLE bit should be set after the ADST bit has been set to 0.

EXTRG Bit (Trigger Select)

The EXTRG bit selects the synchronous trigger or the asynchronous trigger as the trigger for starting A/D conversion.

In group scan mode, the setting of this bit is valid for the selected trigger of group A.

For groups B and C, A/D conversion is started by the selected synchronous trigger regardless of this bit setting.

TRGE Bit (Trigger Start Enable)

The TRGE bit enables or disables A/D conversion by the synchronous trigger and the asynchronous trigger. This bit should be set to 1 in group scan mode.

ADIE Bit (Scan End Interrupt Enable)

The ADIE bit enables or disables the A/D scan end interrupt (S12ADI) in scans except for groups B and C scan in group scan mode.

With double trigger mode deselected, the A/D scan conversion end is generated after the first scan is completed if the ADIE bit is set to 1.

With double trigger mode selected, the A/D scan conversion end is generated after the second and every subsequent even-numbered time scan is completed if the ADIE bit is set to 1 as long as the scan is started by the synchronous trigger selected by the ADSTRGR.TRSA[5:0] bits. When scan is started by a software trigger, even with double trigger mode selected, the A/D scan conversion end interrupt is generated if the ADIE bit is set to 1 when the scan is completed. The A/D scan conversion end interrupt is provided individually for each unit. Table 30.7 shows the relationship between each unit and the A/D scan conversion end interrupt.

Table 30.7 Relationship between Each Unit and A/D scan conversion end Interrupt

Unit	A/D scan conversion end interrupt
S12AD	S12ADI
S12AD1	S12ADI1

ADCS[1:0] Bits (Scan Mode Select)

The ADCS[1:0] bits select the scan mode.

In single scan mode, A/D conversion is performed for the analog inputs selected with the ADANSA0 register in the ascending order of the channel number, and when one cycle of A/D conversion is completed for all the selected channels, the scan conversion is stopped.

In continuous scan mode, while the ADCSR.ADST bit is 1, A/D conversion is performed for the analog inputs selected with the ADANSA0 register in the ascending order of the channel number, and when one cycle of A/D conversion is completed for all the selected channels, A/D conversion is repeated from the first channel. If the ADCSR.ADST bit is set to 0 during continuous scan, A/D conversion is stopped even if scanning is in progress.

In group scan mode, A/D conversion is performed for the analog inputs (group A) selected with the ADANSA0 register in the ascending order of the channel number after scanning is started by the synchronous trigger selected by the ADSTRGR.TRSA[5:0] bits, and when one cycle of A/D conversion is completed for all the selected channels, A/D conversion is stopped. A/D conversion is also performed for the analog inputs (group B or C) selected with the ADANSB0 register and the ADANSC0 register in the ascending order of the channel number after A/D conversion is started by the synchronous trigger selected by the ADSTRGR.TRSA[5:0] bits and ADGCTRGR.TRSC[5:0] bits, and when A/D conversion is completed for all the selected channels, A/D conversion is stopped.

When selecting group scan mode, different channels and triggers should be selected for groups A, B, and C.

When using two groups while group scan mode is set, use groups A and B (ADGCTRGR.GRCE bit = 0). When using three groups, use groups A, B, and C (ADGCTRGR.GRCE bit = 1).

When selecting the temperature sensor output or internal reference voltage for S12AD1, select single scan mode, and deselect all the channels selected with the ADANSA0 register before performing A/D conversion. When A/D conversion of the selected temperature sensor output or internal reference voltage is completed, A/D conversion is stopped.

The ADCS[1:0] bits should be set while the ADST bit is 0. They should not be set simultaneously when 1 is written to the ADST bit.

ADST Bit (A/D Conversion Start)

The ADST bit starts or stops A/D conversion process.

Before the ADST bit is set to 1, set the A/D conversion clock, the conversion mode, and conversion target analog input.

[Setting conditions]

- 1 is written by software.
- The synchronous trigger selected by the ADSTRGR.TRSA[5:0] bits is detected with ADCSR.EXTRG and ADCSR.TRGE bits being set to 0 and 1, respectively.
- The synchronous trigger selected by the ADSTRGR.TRSB[5:0] bits is detected with the ADCSR.TRGE bit being set to 1 in group scan mode.
- The asynchronous trigger is detected with the ADCSR.TRGE and ADCSR.EXTRG bits being set to 1 and the ADSTRGR.TRSA[5:0] bits being set to 000000b.
- With group priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), a group B or C trigger is detected and A/D conversion of group B or C is started.
- With group priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), the ADGSPCR.GBRSCN bit is set to 1 and A/D conversion of group B or C is restarted.
- With group priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), the ADGSPCR.GBRP bit is set to 1 and A/D conversion of the lowest-priority group is started.

[Clearing conditions]

- 0 is written by software.
- The A/D conversion of all the selected channels or the temperature sensor output or internal reference voltage (for S12AD1 only) is completed in single scan mode.
- Group A scan is completed in group scan mode.
- Group B scan is completed in group scan mode.
- Group C scan is completed in group scan mode.
- With group priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), the ADGSPCR.GBRSCN bit is set to 1 and the scanning of the low-priority group started by a trigger is stopped.

Note: When group priority control operation mode has been enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), do not set the ADST bit to 1.

Note: When group priority control operation mode has been enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1) and ADGSPCR.GBRP = 1, do not set the ADST bit to 0. When forcibly terminating A/D conversion, follow the procedure for clearing the ADST bit.

Note: When the single-scan continuous function is used (ADGSPCR.GBRP bit = 1) with group priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), the ADST bit remains as 1.

30.2.4 A/D Channel Select Register A0 (ADANSA0)

(1) S12AD.ADANSA0

Address(es): 0008 9004h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	—	ANSA0 07	ANSA0 06	ANSA0 05	ANSA0 04	ANSA0 03	ANSA0 02	ANSA0 01	ANSA0 00
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ANSA000	A/D Conversion Channel Select	0: AN000/PGAIN00 is not subjected to conversion. 1: AN000/PGAIN00 is subjected to conversion.	R/W
b1	ANSA001		0: AN001/PGAIN01 is not subjected to conversion. 1: AN001/PGAIN01 is subjected to conversion.	R/W
b2	ANSA002		0: AN002/PGAIN02 is not subjected to conversion. 1: AN002/PGAIN02 is subjected to conversion.	R/W
b3	ANSA003		0: AN003 is not subjected to conversion. 1: AN003 is subjected to conversion.	R/W
b4	ANSA004		0: AN004 is not subjected to conversion. 1: AN004 is subjected to conversion.	R/W
b5	ANSA005		0: AN005 is not subjected to conversion. 1: AN005 is subjected to conversion.	R/W
b6	ANSA006		0: AN006 is not subjected to conversion. 1: AN006 is subjected to conversion.	R/W
b7	ANSA007		0: AN007 is not subjected to conversion. 1: AN007 is subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

S12AD.ADANSA0 selects analog input channels for A/D conversion from among AN000/PGAIN00 to AN002/PGAIN02, AN003 to AN007. In group scan mode, this register selects group A channels.

ANSA0n Bit (n = 00 to 07) (A/D Conversion Channel Select)

The ANSA0n bit select analog input channels for A/D conversion from among AN000/PGAIN00 to AN002/PGAIN02, AN003 to AN007. The channels to be selected and the number of channels can be arbitrarily set.

When double trigger mode is selected, the channel selected by S12AD.ADCSR.DBLANS[4:0] bits is selected in group A, and the ANSA0n bit setting is invalid.

The ANSA0n bit should be set while the S12AD.ADCSR.ADST bit is 0.

(2) S12AD1.ADANSA0

Address(es): 0008 9204h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSA007	ANSA006	ANSA005	ANSA004	ANSA003	ANSA002	ANSA001	ANSA000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ANSA000	A/D Conversion Channel Select	0: AN001/PGAIN01 is not subjected to conversion. 1: AN001/PGAIN01 is subjected to conversion.	R/W
b1	ANSA001		0: PGAIN02 is not subjected to conversion. 1: PGAIN02 is subjected to conversion.	R/W
b2	ANSA002		0: AN102 is not subjected to conversion. 1: AN102 is subjected to conversion.	R/W
b3	ANSA003		0: AN103 is not subjected to conversion. 1: AN103 is subjected to conversion.	R/W
b4	ANSA004		0: AN104 is not subjected to conversion. 1: AN104 is subjected to conversion.	R/W
b5	ANSA005		0: AN105 is not subjected to conversion. 1: AN105 is subjected to conversion.	R/W
b6	ANSA006		0: AN106 is not subjected to conversion. 1: AN106 is subjected to conversion.	R/W
b7	ANSA007		0: AN107 is not subjected to conversion. 1: AN107 is subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

S12AD1.ADANSA0 selects analog input channels for A/D conversion from among AN001/PGAIN01, PGAIN02, AN102 to AN107. In group scan mode, this register selects group A channels.

ANSA0n Bit (n = 00 to 07) (A/D Conversion Channel Select)

The ANSA0n bit select analog input channels for A/D conversion from among AN001/PGAIN01, PGAIN02, AN102 to AN107. The channels to be selected and the number of channels can be arbitrarily set.

When performing A/D conversion of the temperature sensor output or internal reference voltage, do not select analog input channels. The setting value of this register should be 0000h.

When double trigger mode is selected, the channel selected by the S12AD1.ADCSR.DBLANS[4:0] bits is selected in group A, and the ANSA0n bit setting is invalid.

The ANSA0n bit should be set while the S12AD1.ADCSR.ADST bit is 0.

30.2.5 A/D Channel Select Register B0 (ADANSB0)

(1) S12AD.ADANSB0

Address(es): 0008 9014h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSB0 07	ANSB0 06	ANSB0 05	ANSB0 04	ANSB0 03	ANSB0 02	ANSB0 01	ANSB0 00
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ANSB000	A/D Conversion Channel Select	0: AN000/PGAIN00 is not subjected to conversion. 1: AN000/PGAIN00 is subjected to conversion.	R/W
b1	ANSB001		0: AN001/PGAIN01 is not subjected to conversion. 1: AN001/PGAIN01 is subjected to conversion.	R/W
b2	ANSB002		0: AN002/PGAIN02 is not subjected to conversion. 1: AN002/PGAIN02 is subjected to conversion.	R/W
b3	ANSB003		0: AN003 is not subjected to conversion. 1: AN003 is subjected to conversion.	R/W
b4	ANSB004		0: AN004 is not subjected to conversion. 1: AN004 is subjected to conversion.	R/W
b5	ANSB005		0: AN005 is not subjected to conversion. 1: AN005 is subjected to conversion.	R/W
b6	ANSB006		0: AN006 is not subjected to conversion. 1: AN006 is subjected to conversion.	R/W
b7	ANSB007		0: AN007 is not subjected to conversion. 1: AN007 is subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

S12AD.ADANSB0 selects analog input channels for A/D conversion from among AN000/PGAIN00 to AN002/PGAIN02, AN003 to AN007 in group B when group scan mode is selected. The S12AD.ADANSB0 register is not used in any scan mode other than group scan mode.

ANSB0n Bit (n = 00 to 07) (A/D Conversion Channel Select)

The ANSB0n bit select analog input channels for A/D conversion from among AN000/PGAIN00 to AN002/PGAIN02, AN003 to AN007 in group B when group scan mode is selected. The S12AD.ADANSB0 register is used for group scan mode only; not used for any other modes. The channels specified in group A (the channels corresponding to group A, selected with the S12AD.ADANSA0 register and the S12AD.ADCSR.DBLANS[4:0] bits in double trigger mode) should be excluded as the channels to be selected and the number of channels to be set.

The ANSB0n bit should be set while the S12AD.ADCSR.ADST bit is 0.

(2) S12AD1.ADANSB0

Address(es): 0008 9214h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSB007	ANSB006	ANSB005	ANSB004	ANSB003	ANSB002	ANSB001	ANSB000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ANSB000	A/D Conversion Channel Select	0: AN001/PGAIN01 is not subjected to conversion. 1: AN001/PGAIN01 is subjected to conversion.	R/W
b1	ANSB001		0: PGAIN02 is not subjected to conversion. 1: PGAIN02 is subjected to conversion.	R/W
b2	ANSB002		0: AN102 is not subjected to conversion. 1: AN102 is subjected to conversion.	R/W
b3	ANSB003		0: AN103 is not subjected to conversion. 1: AN103 is subjected to conversion.	R/W
b4	ANSB004		0: AN104 is not subjected to conversion. 1: AN104 is subjected to conversion.	R/W
b5	ANSB005		0: AN105 is not subjected to conversion. 1: AN105 is subjected to conversion.	R/W
b6	ANSB006		0: AN106 is not subjected to conversion. 1: AN106 is subjected to conversion.	R/W
b7	ANSB007		0: AN107 is not subjected to conversion. 1: AN107 is subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

S12AD1.ADANSB0 selects analog input channels for A/D conversion from among AN001/PGAIN01, PGAIN02, AN102 to AN107 in group B when group scan mode is selected. The S12AD1.ADANSB0 register is not used in any scan mode other than group scan mode.

ANSB0n Bit (n = 00 to 07) (A/D Conversion Channel Select)

The ANSB0n bit select analog input channels for A/D conversion from among AN001/PGAIN01, PGAIN02, AN102 to AN107 in group B when group scan mode is selected. The S12AD1.ADANSB0 register is used for group scan mode only; not used for any other modes. The channels specified in group A (the channels corresponding to group A, selected with the S12AD1.ADANSA0 register and the ADCSR.DBLANS[4:0] bits in double trigger mode) should be excluded as the channels to be selected and the number of channels to be set.

The ANSB0n bit should be set while the S12AD1.ADCSR.ADST bit is 0.

30.2.6 A/D Channel Select Register C0 (ADANSC0)

(1) S12AD.ADANSC0

Address(es): 0008 90D4h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	—	ANSC0 07	ANSC0 06	ANSC0 05	ANSC0 04	ANSC0 03	ANSC0 02	ANSC0 01	ANSC0 00
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ANSC000	A/D Conversion Channel Select	0: AN000/PGAIN00 is not subjected to conversion. 1: AN000/PGAIN00 is subjected to conversion.	R/W
b1	ANSC001		0: AN001/PGAIN01 is not subjected to conversion. 1: AN001/PGAIN01 is subjected to conversion.	R/W
b2	ANSC002		0: AN002/PGAIN02 is not subjected to conversion. 1: AN002/PGAIN02 is subjected to conversion.	R/W
b3	ANSC003		0: AN003 is not subjected to conversion. 1: AN003 is subjected to conversion.	R/W
b4	ANSC004		0: AN004 is not subjected to conversion. 1: AN004 is subjected to conversion.	R/W
b5	ANSC005		0: AN005 is not subjected to conversion. 1: AN005 is subjected to conversion.	R/W
b6	ANSC006		0: AN006 is not subjected to conversion. 1: AN006 is subjected to conversion.	R/W
b7	ANSC007		0: AN007 is not subjected to conversion. 1: AN007 is subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

S12AD.ADANSC0 selects analog input channels for A/D conversion from among AN000/PGAIN00 to AN002/PGAIN02, AN003 to AN007 in group C when group scan mode is selected. The S12AD.ADANSC0 register is not used in any scan mode other than group scan mode.

ANSC0n Bit (n = 00 to 07) (A/D Conversion Channel Select)

The ANSC0n bit select analog input channels for A/D conversion from among AN000/PGAIN00 to AN002/PGAIN02, AN003 to AN007 in group C when group scan mode is selected. The S12AD.ADANSC0 register is used for group scan mode only; not used for any other modes.

The ANSC0n bit should be set while the S12AD.ADCSR.ADST bit is 0.

(2) S12AD1.ADANSC0

Address(es): 0008 92D4h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSC0 07	ANSC0 06	ANSC0 05	ANSC0 04	ANSC0 03	ANSC0 02	ANSC0 01	ANSC0 00
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ANSC000	A/D Conversion Channel Select	0: AN001/PGAIN01 is not subjected to conversion. 1: AN001/PGAIN01 is subjected to conversion.	R/W
b1	ANSC001		0: PGAIN02 is not subjected to conversion. 1: PGAIN02 is subjected to conversion.	R/W
b2	ANSC002		0: AN102 is not subjected to conversion. 1: AN102 is subjected to conversion.	R/W
b3	ANSC003		0: AN103 is not subjected to conversion. 1: AN103 is subjected to conversion.	R/W
b4	ANSC004		0: AN104 is not subjected to conversion. 1: AN104 is subjected to conversion.	R/W
b5	ANSC005		0: AN105 is not subjected to conversion. 1: AN105 is subjected to conversion.	R/W
b6	ANSC006		0: AN106 is not subjected to conversion. 1: AN106 is subjected to conversion.	R/W
b7	ANSC007		0: AN107 is not subjected to conversion. 1: AN107 is subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

S12AD1.ADANSC0 selects analog input channels for A/D conversion from among AN001/PGAIN01, PGAIN02, AN102 to AN107 in group C when group scan mode is selected. The S12AD1.ADANSC0 register is not used in any scan mode other than group scan mode.

ANSC0n Bit (n = 00 to 07) (A/D Conversion Channel Select)

The ANSC0n bit select analog input channels for A/D conversion from among AN001/PGAIN01, PGAIN02, AN102 to AN107 in group C when group scan mode is selected. The S12AD1.ADANSC0 register is used for group scan mode only; not used for any other modes.

The ANSC0n bit should be set while the S12AD1.ADCSR.ADST bit is 0.

30.2.7 A/D-Converted Value Addition/Average Function Channel Select Register 0 (ADADS0)

(1) S12AD.ADADS0

Address(es): 0008 9008h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ADS007	ADS006	ADS005	ADS004	ADS003	ADS002	ADS001	ADS000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ADS000	A/D-Converted Value Addition/ Average Channel Select	0: A/D-converted value addition/average mode for AN000/ PGAIN00 is not selected. 1: A/D-converted value addition/average mode for AN000/ PGAIN00 is selected.	R/W
b1	ADS001		0: A/D-converted value addition/average mode for AN001/ PGAIN01 is not selected. 1: A/D-converted value addition/average mode for AN001/ PGAIN01 is selected.	R/W
b2	ADS002		0: A/D-converted value addition/average mode for AN002/ PGAIN02 is not selected. 1: A/D-converted value addition/average mode for AN002/ PGAIN02 is selected.	R/W
b3	ADS003		0: A/D-converted value addition/average mode for AN003 is not selected. 1: A/D-converted value addition/average mode for AN003 is selected.	R/W
b4	ADS004		0: A/D-converted value addition/average mode for AN004 is not selected. 1: A/D-converted value addition/average mode for AN004 is selected.	R/W
b5	ADS005		0: A/D-converted value addition/average mode for AN005 is not selected. 1: A/D-converted value addition/average mode for AN005 is selected.	R/W
b6	ADS006		0: A/D-converted value addition/average mode for AN006 is not selected. 1: A/D-converted value addition/average mode for AN006 is selected.	R/W
b7	ADS007		0: A/D-converted value addition/average mode for AN007 is not selected. 1: A/D-converted value addition/average mode for AN007 is selected.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

S12AD.ADADS0 selects channels AN000/PGAIN00 to AN002/PGAIN02, AN003 to AN007 on which A/D conversion is performed successively 2, 3, 4, or 16 times and then converted values are added (integrated) or averaged.

ADS0n Bit (n = 00 to 07) (A/D-Converted Value Addition/Average Channel Select)

When the ADS0n bit of the number that is the same as that of A/D-converted channel selected by the S12AD.ADANSA0.ANSA0n bit (n = 00 to 07) or S12AD.ADCSR.DBANS[4:0] bits and S12AD.ADANSB0.ANSB0n bit is set to 1, A/D conversion of analog input of the selected channels is performed successively 2, 3, 4, or 16 times that is set with the S12AD.ADADC.ADC[2:0] bits.

When the S12AD.ADADC.AVEE bit is 0, the value obtained by addition (integration) is stored in the A/D data register.

When the S12AD.ADADC.AVEE bit is 1, the mean value of the results obtained by addition (integration) is stored in the A/D data register.

As for the channel on which the A/D conversion is performed and addition/average mode is not selected, a normal one-time conversion is executed and the conversion result is stored to the A/D data register.

The ADS0n bit should be set while the S12AD.ADCSR.ADST bit is 0.

(2) S12AD1.ADADS0

Address(es): 0008 9208h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ADS007	ADS006	ADS005	ADS004	ADS003	ADS002	ADS001	ADS000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ADS000	A/D-Converted Value Addition/ Average Channel Select	0: A/D-converted value addition/average mode for AN001/ PGAIN01 is not selected. 1: A/D-converted value addition/average mode for AN001/ PGAIN01 is selected.	R/W
b1	ADS001		0: A/D-converted value addition/average mode for PGAIN02 is not selected. 1: A/D-converted value addition/average mode for PGAIN02 is selected.	R/W
b2	ADS002		0: A/D-converted value addition/average mode for AN102 is not selected. 1: A/D-converted value addition/average mode for AN102 is selected.	R/W
b3	ADS003		0: A/D-converted value addition/average mode for AN103 is not selected. 1: A/D-converted value addition/average mode for AN103 is selected.	R/W
b4	ADS004		0: A/D-converted value addition/average mode for AN104 is not selected. 1: A/D-converted value addition/average mode for AN104 is selected.	R/W
b5	ADS005		0: A/D-converted value addition/average mode for AN105 is not selected. 1: A/D-converted value addition/average mode for AN105 is selected.	R/W
b6	ADS006		0: A/D-converted value addition/average mode for AN106 is not selected. 1: A/D-converted value addition/average mode for AN106 is selected.	R/W
b7	ADS007		0: A/D-converted value addition/average mode for AN107 is not selected. 1: A/D-converted value addition/average mode for AN107 is selected.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

S12AD1.ADADS0 selects channels AN001/PGAIN01, PGAIN02, AN102 to AN107 on which A/D conversion is performed successively 2, 3, 4, or 16 times and then converted values are added (integrated) or averaged.

ADS0n Bit (n = 00 to 07) (A/D-Converted Value Addition/Average Channel Select)

When the ADS0n bit of the number that is the same as that of A/D-converted channel selected by the S12AD1.ADANSA0.ANSA0n bit (n = 00 to 07) or S12AD1.ADCSR.DBLANS[4:0] bits and S12AD1.ADANSB0.ANSB0n bit and S12AD1.ADANSC0.ANSC0n bit is set to 1, A/D conversion of analog input of the selected channels is performed successively 2, 3, 4, or 16 times that is set with the S12AD1.ADADC.ADC[2:0] bits. When the S12AD1.ADADC.AVEE bit is 0, the value obtained by addition (integration) is stored in the A/D data register. When the S12AD1.ADADC.AVEE bit is 1, the mean value of the results obtained by addition (integration) is stored in the A/D data register.

As for the channel on which the A/D conversion is performed and addition/average mode is not selected, a normal one-

time conversion is executed and the conversion result is stored to the A/D data register.
The ADS0n bit should be set while the S12AD1.ADCSR.ADST bit is 0.

Figure 30.2 shows a scanning operation sequence in which both the S12AD.ADADS0.ADS002 and S12AD.ADADS0.ADS006 bits are set to 1.

It is assumed that addition mode is selected (S12AD.ADADC.AVEE = 0) in continuous scan mode (S12AD.ADCSR.ADCS[1:0] = 10b), the addition count is set to three times (S12AD.ADADC.ADC[2:0] = 011b), and channels AN000 to AN007 are selected (S12AD.ADANSA0.ANSA0n = FFh). The conversion process begins with AN000. The AN002 conversion is performed successively four times (add three times), and the added (integrated) value is stored in A/D data register 2. After that the AN003 conversion is started. The AN006 conversion is performed successively four times and the added (integrated) value is stored in A/D data register 6. After conversion of AN007, the conversion operation is once again performed in the same sequence from AN000.

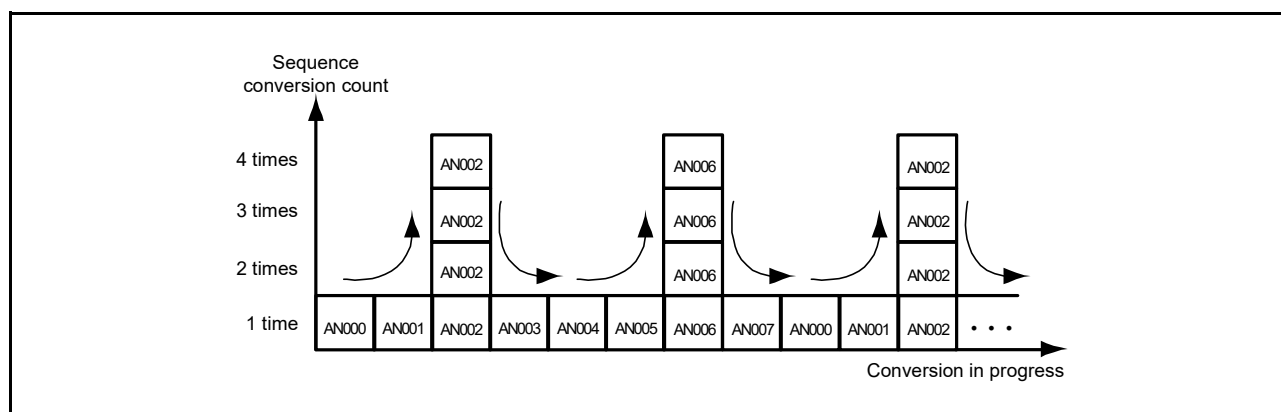


Figure 30.2 Scan Conversion Sequence with S12AD.ADADC.ADC[2:0] = 011b, S12AD.ADADC.AVEE = 0, S12AD.ADADS0.ADS002 = 1, and S12AD.ADADS0.ADS006 = 1

30.2.8 A/D-Converted Value Addition/Average Count Select Register (ADADC)

Address(es): S12AD.ADADC 0008 900Ch, S12AD1.ADADC 0008 920Ch

b7	b6	b5	b4	b3	b2	b1	b0
AVEE	—	—	—	—	ADC[2:0]		
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	ADC[2:0]	Addition Count Select	b2 b0 0 0 0: 1-time conversion (no addition; same as normal conversion) 0 0 1: 2-time conversion (addition once) 0 1 0: 3-time conversion (addition twice)*1 0 1 1: 4-time conversion (addition three times) 1 0 1: 16-time conversion (addition 15 times)*1 Settings other than above are prohibited.	R/W
b6 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	AVEE	Average Mode Enable	0: Addition mode is selected. 1: Average mode is selected.	R/W

Note 1. The AVEE bit is enabled only when 2-time or 4-time conversion is selected. When average mode is selected (ADADC.AVEE bit = 1), do not set 3-time conversion (ADADC.ADC[2:0] = 010b) nor 16-time conversion (ADADC.ADC[2:0] = 101b).

ADADC sets the addition count for A/D conversion of the channel, temperature sensor output, or internal reference voltage (S12AD1 only) for which A/D-converted value addition/average mode is selected, and selects either addition or average mode.

ADC[2:0] Bits (Addition Count Select)

The ADC[2:0] bits set the addition count common to the channels for which A/D conversion and A/D-converted value addition/average mode is selected, including the channels selected in double trigger mode (by ADCSR.DBLANS[4:0] bits), and to A/D conversion of temperature sensor output, or internal reference voltage.

When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the addition count to one time (ADADC.ADC[2:0] = 000b), three times (ADADC.ADC[2:0] = 010b), or 16 times (ADADC.ADC[2:0] = 101b).

The ADC[2:0] bits should be set while the ADCSR.ADST bit is 0.

AVEE Bit (Average Mode Enable)

The AVEE bit selects addition or average mode for A/D conversion of the channel for which A/D conversion and A/D-converted value addition/average mode is selected, including the channels selected in double trigger mode (by ADCSR.DBLANS[4:0] bits) and temperature sensor output, or internal reference voltage.

When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the addition count to one time (ADADC.ADC[2:0] = 000b), three times (ADADC.ADC[2:0] = 010b), or 16 times (ADADC.ADC[2:0] = 101b). The mean value of 1-time, 3-time, and 16-time conversion cannot be obtained.

The AVEE bit should be set while the ADCSR.ADST bit is 0.

30.2.9 A/D Control Extended Register (ADCER)

Address(es): S12AD.ADCER 0008 900Eh, S12AD1.ADCER 0008 920Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADRFMT	—	—	—	DIAGM	DIAGLD	DIAGVAL[1:0]	—	—	ACE	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b5	ACE	A/D Data Register Automatic Clearing Enable	0: Disables automatic clearing. 1: Enables automatic clearing.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b9, b8	DIAGVAL[1:0]	Self-Diagnosis Conversion Voltage Select	b9 b8 0 0: Setting prohibited in self-diagnosis voltage fixed mode 0 1: Uses the voltage of 0 V for self-diagnosis. 1 0: Uses the voltage of reference power supply × 1/2 for self-diagnosis.*1 1 1: Uses the voltage of reference power supply for self-diagnosis.*1	R/W
b10	DIAGLD	Self-Diagnosis Mode Select	0: Rotation mode for self-diagnosis voltage 1: Fixed mode for self-diagnosis voltage	R/W
b11	DIAGM	Self-Diagnosis Enable*2	0: Disables self-diagnosis of 12-bit A/D converter. 1: Enables self-diagnosis of 12-bit A/D converter.	R/W
b14 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	ADRFMT	A/D Data Register Format Select	0: Flush-right is selected for the A/D data register format. 1: Flush-left is selected for the A/D data register format.	R/W

Note 1. The reference voltage refers to the voltage on the pin selected in the ADHVREFCNT register.

Note 2. When the setting of this bit is 1, set the ADDISCR register to 00h (disabling the disconnection detection assist function).

ADCER sets self-diagnosis mode, format of A/D data registers y (ADDRy), and automatic clearing of A/D data registers.

ACE Bit (A/D Data Register Automatic Clearing Enable)

The ACE bit enables or disables automatic clearing (all 0) of ADDRy, ADDR, ADDBLDR, ADDBLDRA, ADDBLDRB, or ADOCDR after any of these registers have been read by the CPU and DTC. Automatic clearing of the A/D data register is enabled to detect a failure which has not been updated in the A/D data register.

DIAGVAL[1:0] Bits (Self-Diagnosis Conversion Voltage Select)

The DIAGVAL[1:0] bits select the voltage value used in self-diagnosis voltage fixed mode. For details, refer to the descriptions of the ADCER.DIAGLD bit.

Self-diagnosis should not be executed by setting the ADCER.DIAGLD bit to 1 when the ADCER.DIAGVAL[1:0] bits are set to 00b.

DIAGLD Bit (Self-Diagnosis Mode Select)

The DIAGLD bit selects whether the three voltage values are rotated or the fixed voltage is used in self-diagnosis.

Setting this bit (ADCER.DIAGLD) to 0 allows conversion of the voltages in rotation mode where 0, the reference power supply × 1/2, and the reference power supply are converted in this order. When self-diagnosis rotation mode is selected after a reset, self-diagnosis is performed from 0 V. When self-diagnosis voltage fixed mode is selected, the fixed voltage specified by the ADCER.DIAGVAL[1:0] bits is converted. In self-diagnosis voltage rotation mode, the self-diagnosis voltage value does not return to 0 when scan conversion is completed. When scan conversion is restarted, therefore,

rotation starts at the voltage value following the previous value. If fixed mode is switched to rotation mode, rotation starts at the fixed voltage value.

The DIAGLD bit should be set while the ADCSR.ADST bit is 0.

DIAGM Bit (Self-Diagnosis Enable)

The DIAGM bit enables or disables self-diagnosis.

Self-diagnosis is used to detect a failure of the 12-bit A/D converter. Specifically, one of the internally generated voltage values 0, the reference power supply $\times 1/2$, and the reference power supply is converted. When conversion is completed, information on the converted voltage and the conversion result is stored into the self-diagnosis data register (ADRD).

ADRD can then be read out by software to determine whether the conversion result falls within the normal range (normal) or not (abnormal). Self-diagnosis is executed once at the beginning of each scan, and one of the three voltages is converted. When self-diagnosis is selected in group scan mode, self-diagnosis is separately executed in groups A, B, and C. The DIAGM bit should be set while the ADCSR.ADST bit is 0.

ADRFMT Bit (A/D Data Register Format Select)

The ADRFMT bit specifies flush-right or flush-left for the data to be stored in ADDR_y, ADDBLDR, ADDBLDRA, ADDBLDRB, ADTSDR, ADOCDR, or ADRD.

The ADRFMT bit should be set while the ADCSR.ADST bit is 0.

For details on the format of each data register, refer to section 30.2.1, A/D Data Registers y (ADDR_y) (y = 0 to 7), A/D Data Duplication Register (ADDBLDR), A/D Data Duplication Register A (ADDBLDRA), A/D Data Duplication Register B (ADDBLDRB), A/D Temperature Sensor Data Register (ADTSDR), A/D Internal Reference Voltage Data Register (ADOCDR), and section 30.2.2, A/D Self-Diagnosis Data Register (ADRD).

30.2.10 A/D Conversion Start Trigger Select Register (ADSTRGR)

Address(es): S12AD.ADSTRGR 0008 9010h, S12AD1.ADSTRGR 0008 9210h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	TRSA[5:0]						—	—	TRSB[5:0]					
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	TRSB[5:0]	A/D Conversion Start Trigger Select for Group B	Select the A/D conversion start trigger for group B in group scan mode.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13 to b8	TRSA[5:0]	A/D Conversion Start Trigger Select	Select the A/D conversion start trigger in single scan mode and continuous scan mode. In group scan mode, the A/D conversion start trigger for group A is selected.	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

ADSTRGR selects the A/D conversion start trigger.

TRSB[5:0] Bits (A/D Conversion Start Trigger Select for Group B)

The TRSB[5:0] bits select the trigger to start scanning of the analog input selected in group B. The TRSB[5:0] bits require to be set only in group scan mode and are not used in any other scan mode. For the scan conversion start trigger for group B, setting a software trigger or an asynchronous trigger is prohibited. Therefore, the TRSB[5:0] bits should be set to the value other than 000000b and the ADCSR.TRGE bit should be set to 1 in group scan mode.

When two groups are selected (ADGCTRGR.GRCE = 0) during group priority operation in group scan mode, setting the ADGSPCR.GBRP bit to 1 allows group B to continuously operate in single scan mode. When setting the ADGSPCR.GBRP bit to 1, set the TRSB[5:0] bits to 3Fh. Note that the issuance period of trigger for A/D conversion must be more than or equal to the actual scan conversion time (t_{SCAN}). If the issuance period is less than t_{SCAN} , A/D conversion by the trigger may have no effect.

When the trigger from the module (MTU, GPTW, TMR) operated in PCLKB is selected as an A/D conversion start trigger, a delay of the period for synchronization processing occurs. Refer to section 30.3.8, Analog Input Sampling Time and Scan Conversion Time for details.

Table 30.8 lists the selection of A/D activation sources by the TRSA[5:0], TRSB[5:0] and TRSC[5:0] bits.

TRSA[5:0] Bits (A/D Conversion Start Trigger Select)

The TRSA[5:0] bits select the trigger to start A/D conversion in single scan mode and continuous scan mode. In group scan mode, the trigger to start scanning of the analog input selected in group A is selected. When scanning is executed in group scan mode or double trigger mode, set the ADCSR.TRGE bit to 1.

- When using the A/D conversion startup source of a synchronous trigger, set the ADCSR.TRGE bit to 1 and set the ADCSR.EXTRG bit to 0.
- When using the asynchronous trigger, set the ADCSR.TRGE bit to 1 and set the ADCSR.EXTRG bit to 1.
- Software trigger (ADCSR.ADST) is enabled regardless of the settings of the ADCSR.TRGE bit, the ADCSR.EXTRG bit, and the TRSA[5:0] bits.

Note that the issuance period of trigger for A/D conversion must be more than or equal to the actual scan conversion time (t_{SCAN}). If the issuance period is less than t_{SCAN} , A/D conversion by a trigger may have no effect. When the trigger from the module (MTU, GPTW, TMR) operated in PCLKB is selected as an A/D conversion start trigger, a delay of the period for synchronization processing occurs. Refer to section 30.3.8, Analog Input Sampling Time and Scan Conversion

Time for details.

Table 30.8 lists the selection of A/D activation sources by the TRSA[5:0], TRSB[5:0] and TRSC[5:0] bits.

30.2.11 A/D Conversion Extended Input Control Register (ADEXICR)

Address(es): S12AD1.ADEXICR 0008 9212h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	OCSA	TSSA	—	—	—	—	—	—	OCSAD	TSSAD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TSSAD	Temperature Sensor Output A/D-Converted Value Addition/Average Mode Select	0: Temperature sensor output A/D-converted value addition/average mode is not selected. 1: Temperature sensor output A/D-converted value addition/average mode is selected.	R/W
b1	OCSAD	Internal Reference Voltage A/D-Converted Value Addition/Average Mode Select	0: Internal reference voltage A/D-converted value addition/average mode is not selected. 1: Internal reference voltage A/D-converted value addition/average mode is selected.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	TSSA	Temperature Sensor Output A/D Conversion Select*1	0: A/D conversion of temperature sensor output is not performed. 1: A/D conversion of temperature sensor output is performed.	R/W
b9	OCSA	Internal Reference Voltage A/D Conversion Select*1	0: A/D conversion of internal reference voltage is not performed. 1: A/D conversion of internal reference voltage is performed.	R/W
b15 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. When the setting of this bit is 1, set the ADCER.DIAGM bit to 0 (disabling the self-diagnosis).

ADEXICR specifies the settings of A/D conversion of the temperature sensor output and internal reference voltage.

TSSAD Bit (Temperature Sensor Output A/D-Converted Value Addition/Average Mode Select)

When the TSSAD bit is set to 1, A/D conversion of the temperature sensor output is selected and performed successively 2, 3, 4, or 16 times that is set with the ADADC.ADC[2:0] bits. When the ADADC.AVEE bit is 0, the value obtained by addition (integration) is stored in the A/D temperature sensor data register (ADTSDR). When the ADADC.AVEE bit is 1, the mean value is stored in the A/D temperature sensor data register (ADTSDR).

The TSSAD bit should be set while the ADCSR.ADST bit is 0.

OCSAD Bit (Internal Reference Voltage A/D-Converted Value Addition/Average Mode Select)

When the OCSAD bit is set to 1, A/D conversion of the internal reference voltage is selected and performed successively 2, 3, 4, or 16 times that is set with the ADADC.ADC[2:0] bits. When the ADADC.AVEE bit is 0, the value obtained by addition (integration) is stored in the A/D internal reference voltage data register (ADOCDR). When the ADADC.AVEE bit is 1, the mean value is stored in ADOCDR.

The OCSAD bit should be set while the ADCSR.ADST bit is 0.

TSSA Bit (Temperature Sensor Output A/D Conversion Select)

This bit selects A/D conversion of the temperature sensor output in single scan mode. When A/D conversion of the temperature sensor output is to be performed, all the bits in the ADANSA0, ADANSB0 registers and the ADCSR.DBLE and OCSA bits should all be set to 0 in single scan mode.

The TSSA bit should be set while the ADCSR.ADST bit is 0. For A/D conversion of the temperature sensor output, the ADDISCR.ADNDIS[4:0] bits should be automatically set to 0Fh to discharge the A/D converter before sampling.

Setting the TSSA bit to 1 automatically activates the temperature sensor. Set the TSSA bit to 1, wait for temperature sensor start time, and then start A/D conversion. For details of start time, see Table 38.56 in section 38, Electrical Characteristics.

Sampling starts after discharging is completed during A/D conversion of the temperature sensor output, an auto discharging period of 15 ADCLK cycles is inserted before sampling.

OCSA Bit (Internal Reference Voltage A/D Conversion Select)

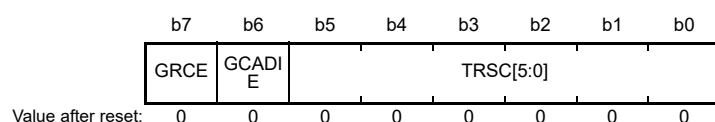
This bit selects A/D conversion of the internal reference voltage in single scan mode. When A/D conversion of the internal reference voltage is to be performed, set all the bits in the ADANSA0, ADANSB0, register and the ADCSR.DBLE bit should be set to all 0 in single scan mode.

The OCSA bit should be set while the ADCSR.ADST bit is 0. For A/D conversion of the internal reference voltage, the ADDISCR.ADNDIS[4:0] bits should be automatically set to 0Fh to discharge the A/D converter before sampling. Set the OCSA bit to 1, wait until the internal reference voltage output is stabilized, and then start A/D conversion. For details of the time until the internal reference voltage output is stabilized, see Table 38.51 in section 38, Electrical Characteristics.

Sampling starts after discharging is completed during A/D conversion of the internal reference voltage, so an auto discharging period of 15 ADCLK cycles is inserted before sampling.

30.2.12 A/D Group C Trigger Select Register (ADGCTRGR)

Address(es): S12AD.ADGCTRGR 0008 90D9h, S12AD1.ADGCTRGR 0008 92D9h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	TRSC[5:0]	Group C A/D Conversion Start Trigger Select	Select the A/D conversion start trigger for group C in group scan mode.	R/W
b6	GCADIE	Group C Scan End Interrupt Enable	0: Disables interrupt generation after completion of group C scan 1: Enables interrupt generation after completion of group C scan	R/W
b7	GRCE	Group C A/D Conversion Operation Enable	Enables A/D conversion operation for group C. 0: Group C is not used 1: Group C is used	R/W

ADGCTRGR enables operation for group C and selects the A/D conversion start trigger. For details on group priority operation, refer to Table 30.14 and Table 30.15.

TRSC[5:0] Bits (Group C A/D Conversion Start Trigger Select)

These bits select the trigger to start scanning of the analog input selected in group C. These bits are used for group scan mode only; not used for any other modes. Software trigger or asynchronous trigger cannot be set as the scan conversion trigger for group C. When using group C in group scan mode, set the TRSC[5:0] bits to a value other than 000000b, set the ADCSR.TRGE bit to 1, and set the GRCE bit to 1.

When group C is used during group priority control in group scan mode and the ADGSPCR.GBRP bit is set to 1, group C can be continuously operated in single scan mode. When continuously operating group C in single scan mode, set the TRSC[5:0] bits to 3Fh and disable trigger selection.

Note that the issuance period of trigger for A/D conversion must be more than or equal to the actual scan conversion time (t_{SCAN}). If the issuance period is less than t_{SCAN} , A/D conversion by a trigger may have no effect.

When the trigger from the module (MTU, GPTW, TMR) operated in PCLKB is selected as an A/D conversion start trigger, a delay of the period for synchronization processing occurs. Refer to section 30.3.8, Analog Input Sampling Time and Scan Conversion Time for details.

Table 30.8 lists the selection of A/D activation sources by the TRSA[5:0], TRSB[5:0] and TRSC[5:0] bits.

Table 30.8 Selection of A/D Activation Sources by the TRSA[5:0], TRSB[5:0] and TRSC[5:0] Bits

			Extended Double Trigger Mode	TRSc[5:0] (x = A, B, C)						
				Binary						Hexadecimal
Module	Source	Remarks	b5	b4	b3	b2	b1	b0		
Trigger source deselection state			—	1	1	1	1	1	1	3F
MTU3	TRGA0N	Compare match/input capture from MTU0.TGRA	—	0	0	0	0	0	1	1
	TRG0N	Compare match from MTU0.TGRE	—	0	0	1	0	0	0	8
	TRG0AEN	Compare match/input capture from MTU0.TGRA, and compare match in MTU0.TGRE	—	1	0	0	0	1	0	22
	TRGA1N	Compare match/input capture from MTU1.TGRA	—	0	0	0	0	1	0	2
	TRGA2N	Compare match/input capture from MTU2.TGRA	—	0	0	0	0	1	1	3
	TRGA3N	Compare match/input capture from MTU3.TGRA	—	0	0	0	1	0	0	4
	TRGA4N	Compare match/input capture from MTU4.TGRA or underflow (trough) of MTU4.TCNT in complementary PWM mode	—	0	0	0	1	0	1	5
	TRG4AN	Compare match between MTU4.TADCORA and MTU4.TCNT	—	0	0	1	0	0	1	9
	TRG4BN	Compare match between MTU4.TADCORB and MTU4.TCNT	—	0	0	1	0	1	0	A
	TRG4ABN	Compare match between MTU4.TADCORA and MTU4.TCNT, and compare match between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is used)	—	0	0	1	1	0	0	C
	TRGA0N or TRG0N	Compare match/input capture from MTU0.TGRA, or compare match in MTU0.TGRE	✓	0	1	1	1	0	0	1C
	TRG4AN or TRG4BN	Compare match between MTU4.TADCORA and MTU4.TCNT, or compare match between MTU4.TADCORB and MTU4.TCNT	✓	0	0	1	0	1	1	B
GPTW	GTADTRA0N	Compare match in GPTW0.GTADTRA	—	0	1	0	0	0	1	11
	GTADTRB0N	Compare match in GPTW0.GTADTRB	—	0	1	0	0	1	0	12
	GTADTRA1N	Compare match in GPTW1.GTADTRA	—	0	1	0	0	1	1	13
	GTADTRB1N	Compare match in GPTW1.GTADTRB	—	0	1	0	1	0	0	14
	GTADTRA2N	Compare match in GPTW2.GTADTRA	—	0	1	0	1	0	1	15
	GTADTRB2N	Compare match in GPTW2.GTADTRB	—	0	1	0	1	1	0	16
	GTADTRA0N or GTADTRB0N	Compare match in GPTW0.GTADTRA, or compare match in GPTW0.GTADTRB	✓	0	1	1	0	0	1	19
	GTADTRA1N or GTADTRB1N	Compare match in GPTW1.GTADTRA, or compare match in GPTW1.GTADTRB	✓	0	1	1	0	1	0	1A
	GTADTRA2N or GTADTRB2N	Compare match in GPTW2.GTADTRA, or compare match in GPTW2.GTADTRB	✓	0	1	1	0	1	1	1B
	GTADTRA0N or GTADTRB0N or GTADTRA1N or GTADTRB1N	Compare match in GPTW0.GTADTRA, or compare match in GPTW0.GTADTRB, or compare match in GPTW1.GTADTRA, or compare match in GPTW1.GTADTRB	✓	1	0	1	0	1	0	2A
	GTADTRA0N or GTADTRB0N or GTADTRA1N or GTADTRB1N or GTADTRA2N or GTADTRB2N	Compare match in GPTW0.GTADTRA, or compare match in GPTW0.GTADTRB, or compare match in GPTW1.GTADTRA, or compare match in GPTW1.GTADTRB, or compare match in GPTW2.GTADTRA, or compare match in GPTW2.GTADTRB	✓	1	0	1	0	0	1	29
	TMR	TMTRG0AN_0	Compare match in TMR0.TCORA, and compare match in TMR0.TCNT	—	0	1	1	1	0	1
TMTRG0AN_1		Compare match in TMR2.TCORA, and compare match in TMR2.TCNT	—	0	1	1	1	1	0	1E

GCADIE Bit (Group C Scan End Interrupt Enable)

This bit enables or disables scan end interrupt generation for group C. A scan end interrupt for group C is provided individually for each unit. Table 30.9 shows the relationship between each unit and the scan end interrupt for group C.

Table 30.9 Relationship between Each Unit and Group C Scan End Interrupt

Unit	Group C Scan End Interrupt
S12AD	GCADI
S12AD1	GCADI1

GRCE Bit (Group C A/D Conversion Operation Enable)

When using group C in group scan mode, set the GRCE bit to 1.

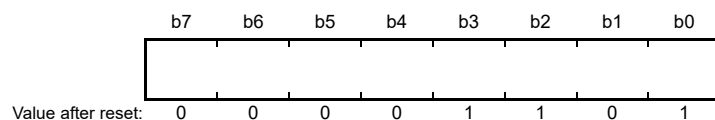
When the GRCE bit is 0, trigger input for group C is disabled.

During group priority operation (the ADGSPCR.PGS bit is 1) with group C used, when the ADGSPCR.GBRP bit is set to 1, single scan for group C is continuously operated. When the GRCE bit is set to 1, single scan for group B is not continuously operated.

The GRCE bit should be set while the ADCSR.ADST bit is 0.

30.2.13 A/D Sampling State Register n (ADSSTRn) (n = 0 to 7, T, O)

Address(es): S12AD.ADSSTR0 0008 90E0h, S12AD.ADSSTR1 0008 90E1h, S12AD.ADSSTR2 0008 90E2h, S12AD.ADSSTR3 0008 90E3h, S12AD.ADSSTR4 0008 90E4h, S12AD.ADSSTR5 0008 90E5h, S12AD.ADSSTR6 0008 90E6h, S12AD.ADSSTR7 0008 90E7h, S12AD1.ADSSTR0 0008 92DEh, S12AD1.ADSSTR1 0008 92DFh, S12AD1.ADSSTR2 0008 92E0h, S12AD1.ADSSTR3 0008 92E1h, S12AD1.ADSSTR4 0008 92E2h, S12AD1.ADSSTR5 0008 92E3h, S12AD1.ADSSTR6 0008 92E4h, S12AD1.ADSSTR7 0008 92E5h, S12AD1.ADSSTR8 0008 92E6h, S12AD1.ADSSTR9 0008 92E7h



The ADSSTRn register sets the sampling time for analog input.

Set the sampling time based on the number of clock cycles for ADCLK (A/D conversion clock). The sampling time (t_{SPL}) can be determined by the following formula:

$$t_{SPL} = (ADSSTR + 0.5) \times ADCLK$$

The initial value of ADSSTR is 13, resulting in a sampling time of 13.5 clock cycles. If the impedance of the analog input signal source is high and the sampling time is insufficient, or if ADCLK is slow, the sampling time can be adjusted. The ADSSTRn register should be set when the ADCSR.ADST bit is “0”. The setting value of this register should be 10 or more.

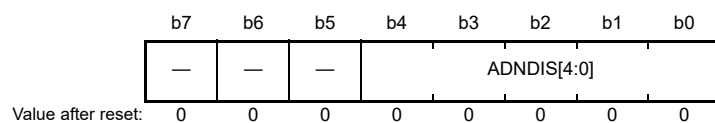
Table 30.10 shows the relationship between the A/D sampling state register and the relevant channels. For details, refer to section 30.3.8, Analog Input Sampling Time and Scan Conversion Time.

Table 30.10 Relationship between A/D Sampling State Register and Relevant Channels

Unit	Register Name	Relevant Channels
S12AD	ADSSTR0 register	AN000/PGAIN00, Self-Diagnosis
	ADSSTR1 register	AN001/PGAIN01
	ADSSTR2 register	AN002/PGAIN02
	ADSSTR3 register	AN003
	ADSSTR4 register	AN004
	ADSSTR5 register	AN005
	ADSSTR6 register	AN006
	ADSSTR7 register	AN007
S12AD1	ADSSTR0 register	AN001/PGAIN01, Self-Diagnosis
	ADSSTR1 register	PGAIN02
	ADSSTR2 register	AN102
	ADSSTR3 register	AN103
	ADSSTR4 register	AN104
	ADSSTR5 register	AN105
	ADSSTR6 register	AN106
	ADSSTR7 register	AN107
	ADSSTR7 register	Temperature sensor output
	ADSSTRO register	Internal reference voltage

30.2.14 A/D Disconnection Detection Control Register (ADDISCR)

Address(es): S12AD.ADDISCR 0008 907Ah, S12AD1.ADDISCR 0008 927Ah



Bit	Symbol	Bit Name	Description	R/W
b4 to b0	ADNDIS[4:0]	A/D Disconnection Detection Assist Setting	b4 ADNDIS[4]: Discharge/precharge selected 0: Discharge 1: Precharge b3 to b0 ADNDIS[3:0]: Discharge/precharge period	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

ADDISCR sets the disconnection detection assist function.

ADNDIS[4:0] Bits (A/D Disconnection Detection Assist Setting)

These bits select either precharge or discharge and the period of precharge/discharge for the A/D disconnection detection assist function. Setting the ADNDIS[4] bit = 1 allows to select precharge and setting the ADNDIS[4] bit = 0 allows to select discharge. The period of precharge/discharge can be set with the ADNDIS[3:0] bits. When the ADNDIS[3:0] bits = 0000b, the disconnection detection assist function is not effective. Setting of the ADNDIS[3:0] bits to 0001b is prohibited. Except for the case of ADNDIS[3:0] = 0000b or 0001b, the specified value indicates the number of clock cycles for the period of precharge/discharge. The ADNDIS[4:0] bits should be set when the ADCSR.ADST bit is 0. When the ADEXICR.OCOA or TSSA bit is set to 1 to perform A/D conversion of the temperature sensor output or internal reference voltage, ADNDIS[4:0] are automatically fixed to 0Fh, and discharging is executed prior to A/D conversion (auto-discharging). An auto-discharge period of 15 ADCLK cycles is inserted before sampling each time the temperature sensor output or internal reference voltage is A/D-converted.

30.2.15 A/D Group Scan Priority Control Register (ADGSPCR)

Address(es): S12AD.ADGSPCR 0008 9080h, S12AD1.ADGSPCR 0008 9280h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	GBRP	LGRRS	—	—	—	—	—	—	—	—	—	—	—	—	GBRSCN	PGS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PGS	Group Priority Control Setting* ¹	0: Operation is without group priority control 1: Operation is with group priority control	R/W
b1	GBRSCN	Low-Priority Group Restart Setting* ²	(Enabled only when PGS = 1. Reserved when PGS = 0.) 0: Scanning for the group is not restarted after having been discontinued due to group priority control. 1: Scanning for the group is restarted after having been discontinued due to group priority control.	R/W
b13 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b14	LGRRS	Restart Channel Select	(Enabled when PGS = 1 and GBRSCN = 1. Reserved when PGS = 0 or GBRSCN = 0.) 0: Scanning is restarted from the scan start channel. 1: Scanning is restarted from the channel on which A/D conversion is not completed.	R/W
b15	GBRP	Single Scan Continuous Start* ³	(Enabled only when PGS = 1. Reserved when PGS = 0.) 0: Single scan is not continuously activated. 1: Single scan for the lowest-priority group is continuously activated.	R/W

Note 1. When the PGS bit is to be set to 1, the ADCSR.ADCS[1:0] bits must be set to 01b (group scan mode). If the bits are set to any other values, proper operation is not guaranteed.

Note 2. When the GBRSCN bit is to be set to 1, the frequency ratio of peripheral module clock PCLK to A/D conversion clock ADCLK should be set to 1:1.

Note 3. When the GBRP bit has been set to 1, single scan is performed continuously for the lowest-priority group regardless of the setting of the GBRSCN bit.

ADGSPCR is used to discontinue scanning of the low-priority group and make settings for priority control of scanning for the priority group in group scan mode.

For the settings on group priority operation, refer to Table 30.14 and Table 30.15.

PGS Bit (Group Priority Control Setting)

This bit sets the priority of operation in group scan mode. Set this bit to 1 when giving priority to operation on the group. When the PGS bit is to be set to 1, the ADCSR.ADCS[1:0] bits must be set to 01b (group scan mode).

During group priority operation, a trigger to start scanning for the priority group is accepted during scan for the low-priority group, and scan for the priority group is started after scan for the low-priority group was discontinued. The priority order is group A > group B > group C.

When a trigger to start scanning for group B is accepted during scan for group C, group C scan is discontinued, and scan for group B is started. When a trigger to start scanning for group A is accepted during scan for group C, group C scan is discontinued, and scan for group A is started.

Likewise, when a trigger to start scanning for group A is accepted during scan for group B, group B scan is discontinued and scan for group A is started.

When setting the PGS bit to 0, clearing should be performed by software according to section 30.7.2, Notes on Stopping A/D Conversion. When setting the PGS bit to 1, follow the procedure described in section 30.3.6.3, Operation under Group Priority Control.

GBRSCN Bit (Low-Priority Group Restart Setting)

This bit controls the restarting of scan operation during group priority control.

If a scan operation on the low-priority group has been stopped by a priority group trigger input with the GBRSCN bit set to 1, the scan operation is restarted after the scanning of the priority group is completed. Also, if a low-priority trigger is input during scan for the priority group, the scan operation on the low-priority group is restarted after the scan for the priority group is completed.

If the GBRSCN bit has been set to 0, triggers that are input during A/D conversion are ignored. Also, the ADCSR.ADST bit must be 0 when the GBRSCN bit is to be set.

The setting of the GBRSCN bit is enabled when the PGS bit is set to 1.

LGRRS Bit (Restart Channel Select)

This bit sets the channel on which scan is restarted during group priority control. The setting of the LGRRS bit is enabled when the PGS and GBRSCN bits are set to 1.

If a scan operation on the low-priority group has been stopped due to group priority operation with the LGRRS bit set to 0, the scan operation is restarted from the start channel after the scan for the priority group is completed.

If a scan operation on the low-priority group has been stopped due to group priority operation with the LGRRS bit set to 1, the scan operation is restarted*1 on the channel on which A/D conversion is not completed after the scan for the priority group is completed.

The LGRRS bit should be set while the ADCSR.ADST bit is 0.

Note 1. If A/D conversion on the addition set channel is not completed for the set number of times when scanning is stopped, A/D conversion on the channel is restarted for the set number of times when scanning is restarted.

GBRP Bit (Single Scan Continuous Start)

This bit is set when the lowest-priority group is continuously operated in single scan mode while group priority operation is set. The lowest-priority group is group C when groups A, B, and C are used; group B when groups A and B are used. Setting the GBRP bit to 1 starts a single scan on the lowest-priority group. On completion of the scan, another single scan on the lowest-priority group is automatically started.

If scanning has been stopped due to group priority operation, single scan on the lowest-priority group is automatically restarted on completion of the A/D conversion on the priority group.

Disable the trigger input for the lowest-priority group before setting the GBRP bit to 1. When the GBRP bit is set to 1, only the lowest-priority group is scanned again even if the GBRSCN bit is 0.

The ADCSR.ADST bit must be 0 when the GBRP bit is to be set.

The setting of the GBRP bit is enabled when the PGS bit is 1.

30.2.16 A/D High-Potential/Low-Potential Reference Voltage Control Register (ADHVREFCNT)

Address(es): S12AD.ADHVREFCNT 0008 908Ah, S12AD1.ADHVREFCNT 0008 928Ah

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	LVSEL	—	—	HVSEL[1:0]	
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	HVSEL[1:0]	High-Potential Reference Voltage Select	b1 b0 0 0: VCC is selected as the high-potential reference voltage. 0 1: VREFH0 is selected as the high-potential reference voltage.*1 Settings other than above are prohibited.	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	LVSEL	Low-Potential Reference Voltage Select	0: VSS is selected as the low-potential reference voltage. 1: VREFL0 is selected as the low-potential reference voltage.*1	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Set the P13 and P14 pins to analog inputs/outputs (PORT1.PMR.B3 = 0, PORT1.PMR.B4 = 0, PORT1.PDR.B3 = 0, PORT1.PDR.B4 = 0, P13PFS.ASEL = 1, and P14PFS.ASEL = 1).

The ADHVREFCNT register specifies the high-potential and low-potential reference voltages. Set this register before performing A/D conversion.

HVSEL[1:0] Bits (High-Potential Reference Voltage Select)

These bits are used to set the high-potential reference voltage. VCC or VREFH0 is selectable as the high-potential reference voltage.

LVSEL Bit (Low-Potential Reference Voltage Select)

This bit is used to set the low-potential reference voltage. VSS or VREFL0 is selectable as the low-potential reference voltage.

30.3 Operation

30.3.1 A/D Converter Initialization Flowchart

Initialize the A/D converter following the example of flowchart shown in Figure 30.3.

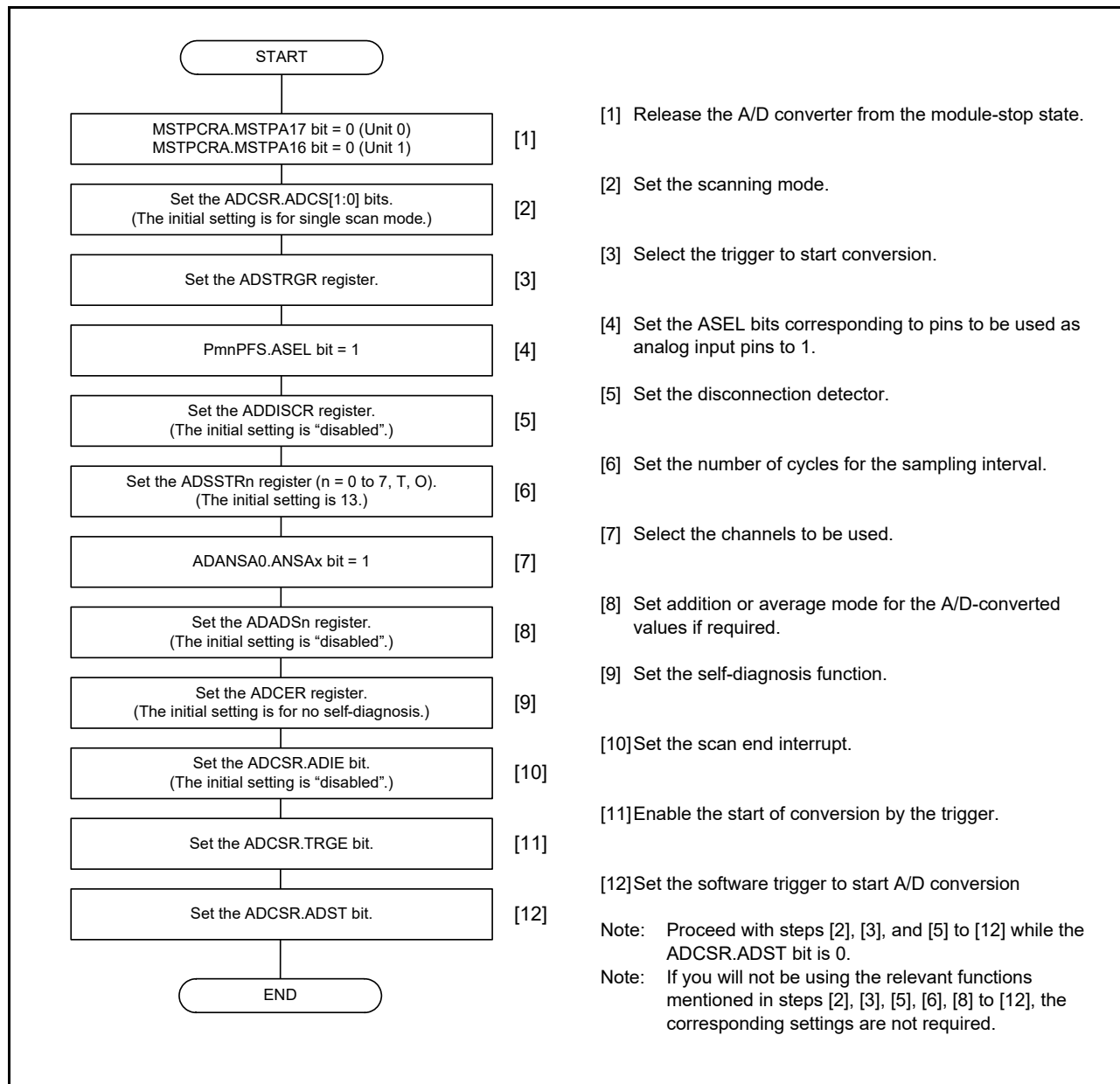


Figure 30.3 A/D Converter Initialization Flowchart

30.3.2 A/D Conversion Units and Analog Inputs

The correspondence between the analog input of the A/D converter and the storage destination register of the A/D conversion results is as shown in Table 30.11.

Table 30.11 Analog Input Settings and A/D Conversion Result Storage Destination

Analog Input Settings				
unit	Input channel	A/D channel select register (n = A, B, C)	PGA setting	A/D conversion result storage destination
Unit 0	AN000	S12AD.ADANSn0.ANSn000 = 1	PGA0.PGA0EN = 0	S12AD.ADDR0
	PGAIN00		PGA0.PGA0EN = 1	
	AN001	S12AD.ADANSn0.ANSn001 = 1	PGA0.PGA1EN = 0	S12AD.ADDR1
	PGAIN01		PGA0.PGA1EN = 1	
	AN002	S12AD.ADANSn0.ANSn002 = 1	PGA0.PGA2EN = 0	S12AD.ADDR2
	PGAIN02		PGA0.PGA2EN = 1	
	AN003	S12AD.ADANSn0.ANSn003 = 1	—	S12AD.ADDR3
	AN004	S12AD.ADANSn0.ANSn004 = 1	—	S12AD.ADDR4
	AN005	S12AD.ADANSn0.ANSn005 = 1	—	S12AD.ADDR5
	AN006	S12AD.ADANSn0.ANSn006 = 1	—	S12AD.ADDR6
	AN007	S12AD.ADANSn0.ANSn007 = 1	—	S12AD.ADDR7
Unit 1	AN001	S12AD1.ADANSn0.ANSn000 = 1	PGA0.PGA1EN = 0	S12AD1.ADDR0
	PGAIN01		PGA0.PGA1EN = 1	
	—	S12AD1.ADANSn0.ANSn001 = 1	PGA0.PGA2EN = 0	—
	PGAIN02		PGA0.PGA2EN = 1	S12AD1.ADDR1
	AN102	S12AD1.ADANSn0.ANSn002 = 1	—	S12AD1.ADDR2
	AN103	S12AD1.ADANSn0.ANSn003 = 1	—	S12AD1.ADDR3
	AN104	S12AD1.ADANSn0.ANSn004 = 1	—	S12AD1.ADDR4
	AN105	S12AD1.ADANSn0.ANSn005 = 1	—	S12AD1.ADDR5
	AN106	S12AD1.ADANSn0.ANSn006 = 1	—	S12AD1.ADDR6
	AN107	S12AD1.ADANSn0.ANSn007 = 1	—	S12AD1.ADDR7

30.3.3 Scanning Operation

In scanning, A/D conversion is performed sequentially on the analog inputs of the specified channels.

A scan conversion is performed in three operating modes: single scan mode, continuous scan mode, and group scan mode. In single scan mode, one or more specified channels are scanned once. In continuous scan mode, one or more specified channels are scanned repeatedly until the ADCSR.ADST bit is cleared to 0 from 1 by software. In group scan mode, the selected channels of groups A, B, and C are scanned once after starting to be scanned according to the respective synchronous trigger.

In single scan mode and continuous scan mode, A/D conversion is performed for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n. In group scan mode, A/D conversion is performed for ANn channels of groups A, B, and C selected by the ADANSA0 register, ADANSB0 register, and ADANSC0 register, respectively, starting from the channel with the smallest number n.

When self-diagnosis is selected, it is executed once at the beginning of each scan and one of the three voltages internally generated in the 12-bit A/D converter is converted.

When the temperature sensor output or internal reference voltage is to be A/D converted, selecting both as the targets for A/D conversion at the same time is prohibited.

Double trigger mode is to be used with single scan mode or group scan mode. With double trigger mode being enabled, A/D conversion data of a channel selected by the ADCSR.DBLANS[4:0] bits is duplicated only if the conversion is

started by the synchronous trigger selected by the ADSTRGR.TRSA[5:0] bits. In group scan mode, the double trigger function can be used only for group A.

Extended double trigger mode indicates a state when the synchronous trigger corresponding to the extended double trigger mode shown in Table 30.8 in section 30.2.12 (two or more synchronous trigger sources enabled) is selected by the TRSA[5:0] bits in the A/D conversion start trigger select register (ADSTRGR) in double trigger mode.

In extended double trigger mode, in addition to normal operations in double trigger mode, A/D conversion data is stored in A/D data duplication register A (ADDBLDRA) or A/D data duplication register B (ADDBLDRB) depending on the trigger type. If two or more types of triggers have occurred simultaneously in this mode, A/D conversion data is not sorted by the trigger sources and is stored in data duplication register B (ADDBLDRB).

Note that if a new trigger is input during A/D conversion caused by another trigger, the new trigger is ignored.

Table 30.12 shows the selection of scan mode, double trigger mode, and targets for A/D conversion.

Table 30.12 Selection of Scan Mode, Double Trigger Mode, and Targets for A/D Conversion

Scan Mode Setup	Double Trigger Mode Setting	Targets for A/D Conversion				
		Self-Diagnosis	Analog Input (Including Group A)	Analog Input (Group B and Group C)	Temperature Sensor Output	Internal Reference Voltage
Single scan	DBLE = 0	✓*1	✓*2	x	✓*3	✓*3
	DBLE = 1	x	✓ (1 channel only)	x	x	x
Continuous scan	DBLE = 0*4	✓	✓	x	x	x
Group scan	DBLE = 0	✓	✓	✓	x	x
	DBLE = 1	x	✓ (1 channel only)	✓*5	x	x

✓: Selectable, x: Not selectable

Note 1. When the temperature sensor output or internal reference voltage is to be A/D converted, selecting self-diagnosis is prohibited.

Note 2. When the temperature sensor output or internal reference voltage is to be A/D converted, selecting analog input is prohibited.

Note 3. Selecting the temperature sensor output and internal reference voltage at the same time is prohibited.

Note 4. Setting the double-trigger mode (DBLE = 1) in the continuous scan mode is prohibited.

Note 5. When the double trigger mode in the group scan mode is selected, operations of groups B and C are the same as that of the single scan mode started by a synchronous trigger.

The ADST0 output is used to output the S12AD.ADCSR.ADST bit state, the ADST1 output is used to output the S12AD1.ADCSR.ADST bit state.

The ADSS0 output is used to output the sampling state of S12AD, the ADSS1 output is used to sampling state of S12AD1.

30.3.4 Single Scan Mode

30.3.4.1 Basic Operation

In basic operation of single scan mode, A/D conversion is performed once on the analog input of the specified channels as below.

- (1) When the ADCSR.ADST bit is set to 1 (A/D conversion start) by software, or synchronous or asynchronous trigger input, A/D conversion is performed for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n. During the sampling period, the ADSS0 pin becomes High.
- (2) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
- (3) When A/D conversion of all the selected channels is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (4) The ADCSR.ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the 12-bit A/D converter enters a wait state.

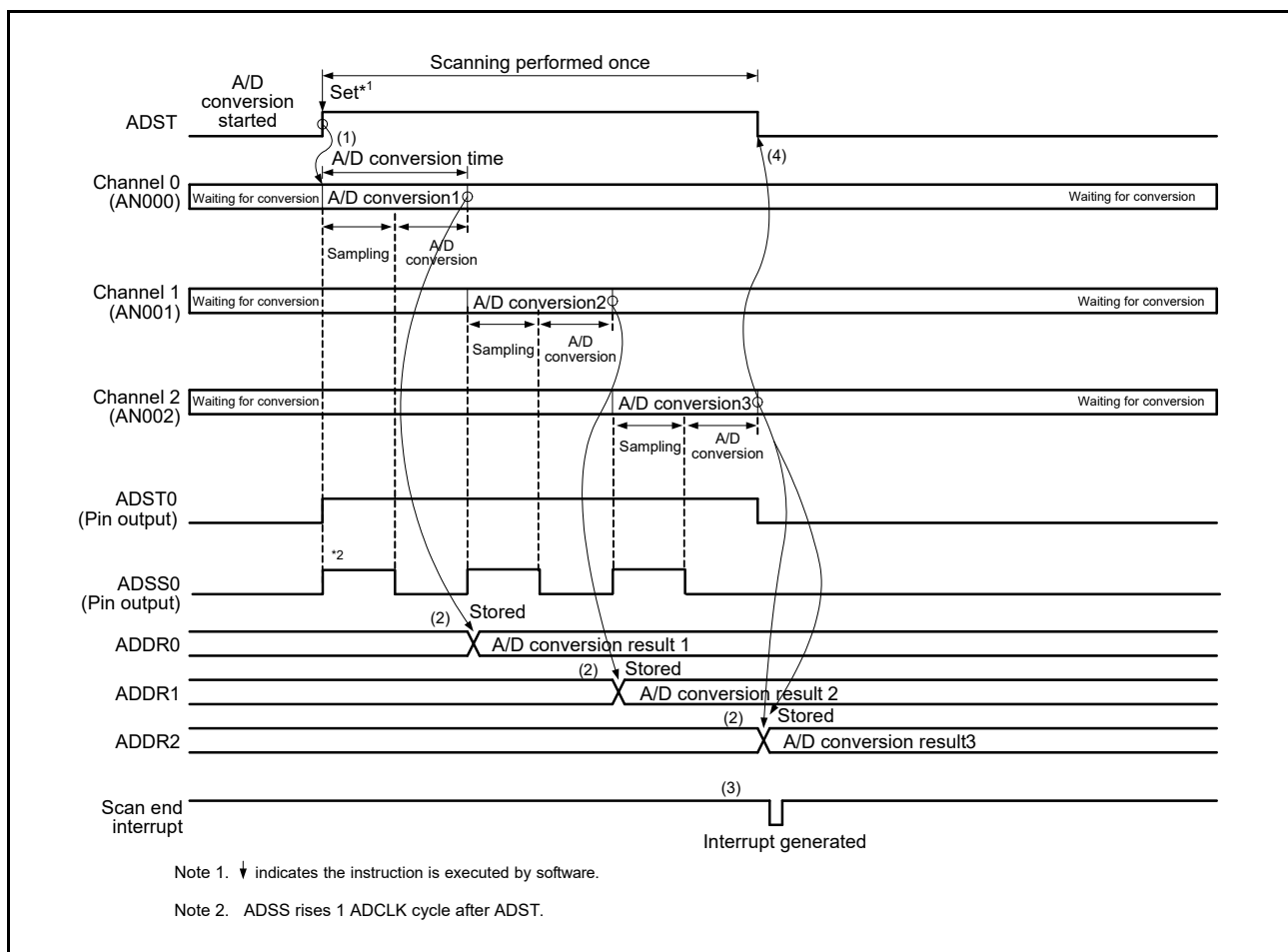


Figure 30.4 Example of Operation in Single Scan Mode (Basic Operation: AN000, AN001, AN002 Selected)

30.3.4.2 Channel Selection and Self-Diagnosis

When channels and self-diagnosis are selected, A/D conversion is performed once for the reference voltage supplied to the 12-bit A/D converter as below. After that, A/D conversion is performed only once on the analog input of the selected channels.

- (1) A/D conversion for self-diagnosis is started when the ADCSR.ADST bit is set to 1 (A/D conversion start) by software, or synchronous or asynchronous trigger input. During the sampling period, the ADSS0 pin becomes High.
- (2) When A/D conversion for self-diagnosis is completed, A/D conversion result is stored into the A/D self-diagnosis data register (ADRD), and A/D conversion is performed for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n. During the sampling period, the ADSS0 pin becomes High.
- (3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
- (4) When A/D conversion of all the selected channels is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (5) The ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the 12-bit A/D converter enters a wait state.

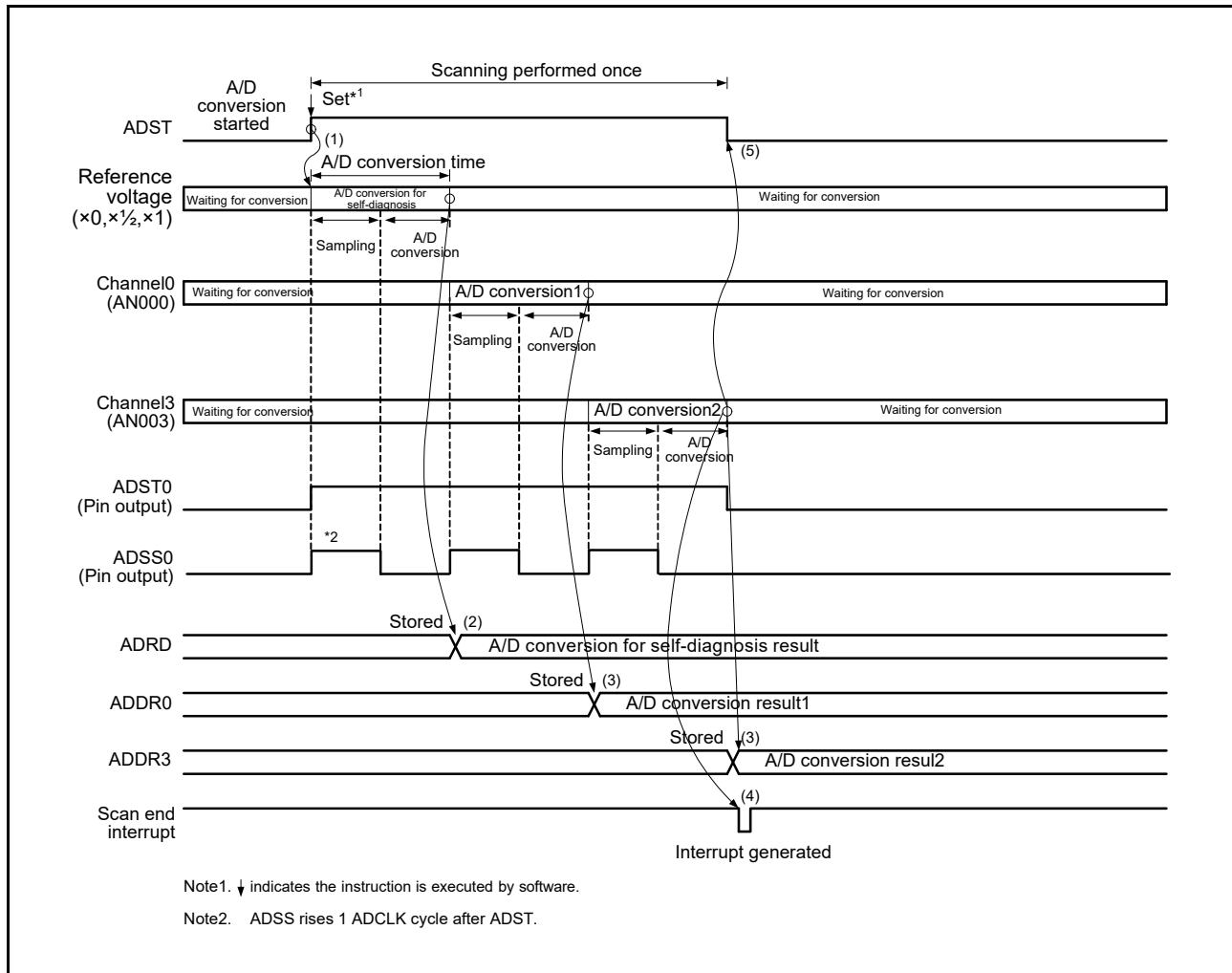


Figure 30.5 Example of Operation in Single Scan Mode
(Basic Operation: AN000, AN003 Selected + Self-Diagnosis)

30.3.4.3 A/D Conversion of Temperature Sensor Output/Internal Reference Voltage

A/D conversion of the temperature sensor output or internal reference voltage can be selected for S12AD1 only. A/D conversion of the temperature sensor output or internal reference voltage is performed in single scan mode as below. All channels should be deselected (by setting the ADANSA0 register bits to all 0 and the ADCSR.DBLE bit to 0). When A/D conversion of the temperature sensor output is to be selected, set the ADEXICR.OCSA bit to 0 so that A/D conversion of the internal reference voltage does not proceed. When A/D conversion of the internal reference voltage is to be selected, set the ADEXICR.TSSA bit to 0 so that A/D conversion of the temperature sensor output does not proceed.

- (1) Set the sampling time. For the necessary sampling time, refer to section 38, Electrical Characteristics.
- (2) After switching to A/D conversion of the internal reference voltage or the temperature sensor output, start A/D conversion by setting the ADST bit to 1.
- (3) When A/D conversion is completed, the conversion result is stored into the corresponding A/D temperature sensor data register (ADTSDR) or A/D internal reference voltage data register (ADOCDR). If the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled), a scan end interrupt request is generated.
- (4) The ADST bit remains 1 during A/D conversion, and is automatically cleared to 0 upon completion of A/D conversion. Then the 12-bit A/D converter enters a wait state.

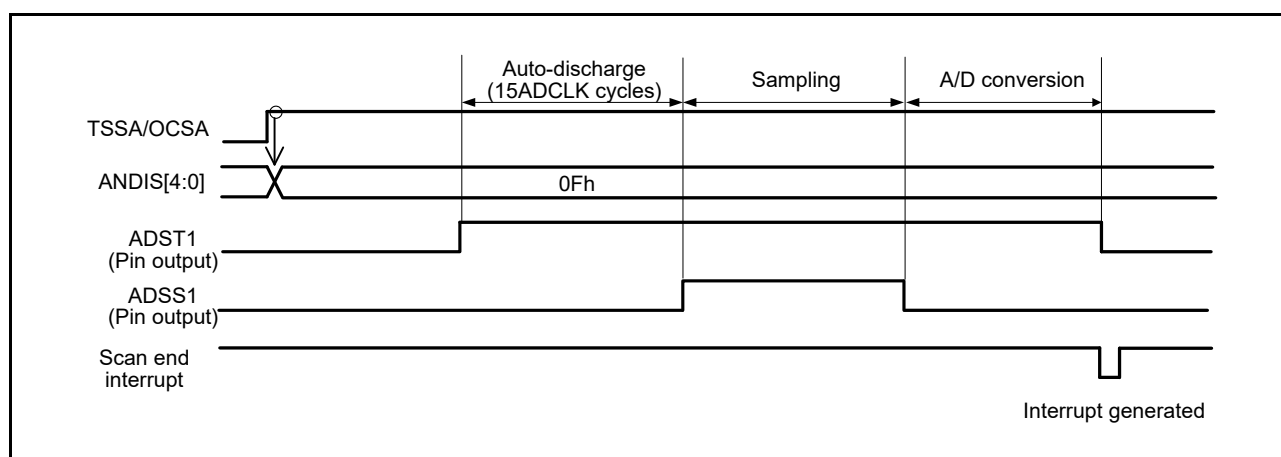


Figure 30.6 Example of Operation in Single Scan Mode (Temperature Sensor output and Internal Reference Voltage Selected: S12AD1 only)

30.3.4.4 A/D Conversion in Double Trigger Mode

In single scan mode with double trigger mode, single scan operation started by synchronous trigger is performed twice as below.

Self-diagnosis should be deselected, and the temperature sensor A/D conversion select bit (S12AD1.ADEXICR.TSSA) and internal reference voltage A/D conversion select bit (S12AD1.ADEXICR.OCSA) should be set to 0.

Duplication of A/D conversion data is enabled by setting the channel numbers to be duplicated to the ADCSR.DBLANS[4:0] bits and setting the ADCSR.DBLE bit to 1. When the ADCSR.DBLE bit is set to 1, channel selection using the ADANSA0 register is invalid. In double trigger mode, synchronous triggers should be selected using the ADSTRGR.TRSA[5:0] bits, the ADCSR.EXTRG bit should be set to 0, and the ADCSR.TRGE bit should be set to 1. Software trigger should not be used.

- (1) When the ADCSR.ADST bit is set to 1 (A/D conversion start) by synchronous trigger input, A/D conversion is started on the single channel selected by the ADCSR.DBLANS[4:0] bits.
- (2) When A/D conversion is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
- (3) The ADCSR.ADST bit is automatically cleared to 0 and the 12-bit A/D converter enters a wait state. Here, a scan end interrupt request is not generated irrespective of the ADCSR.ADIE bit setting (interrupt generation upon scanning completion enabled).
- (4) When the ADCSR.ADST bit is set to 1 (A/D conversion start) by the second trigger input, A/D conversion is started on the single channel selected by the ADCSR.DBLANS[4:0] bits.
- (5) When A/D conversion is completed, the A/D conversion result is stored into the A/D data duplication register (ADDBLDR), which is exclusively used in double trigger mode.
- (6) If the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled), a scan end interrupt request is generated.
- (7) The ADCSR.ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion is completed. Then the 12-bit A/D converter enters a wait state.

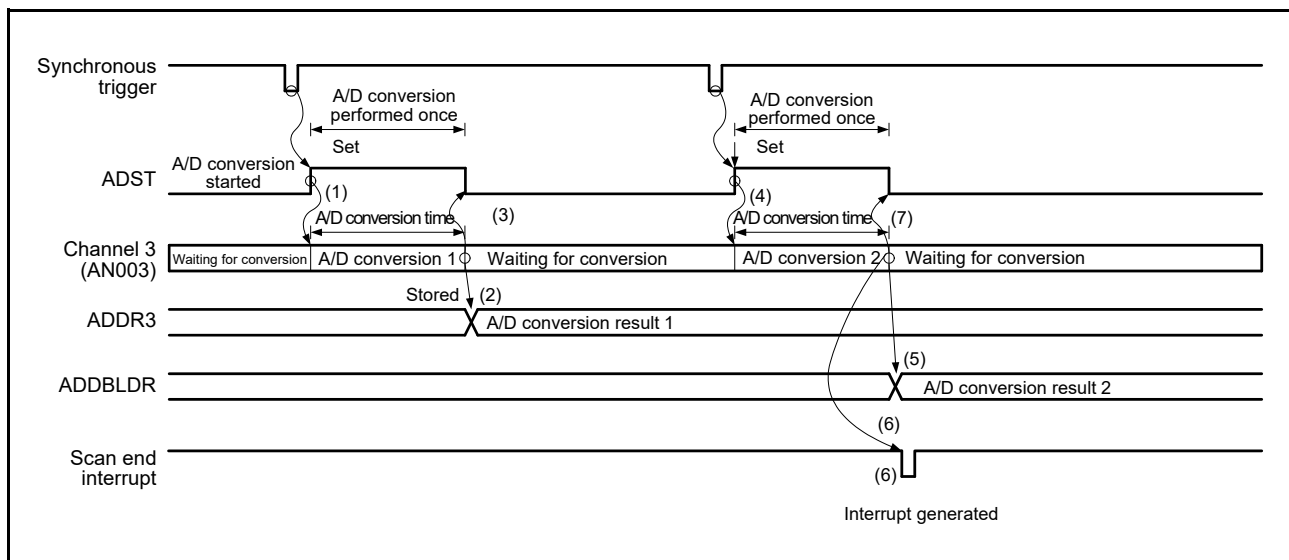


Figure 30.7 Example of Operation in Single Scan Mode (Double Trigger Mode Selected; AN003 Duplicated)

30.3.4.5 A/D Conversion in Extended Double Trigger Mode

When the double-trigger mode is selected in single-scanning mode, with the synchronous trigger corresponding to the extended double trigger mode shown in Table 30.8 in section 30.2.12 selected in the TRSA[5:0] bits of the A/D conversion start trigger select register (ADSTRGR), proceed with single scanning multiple times as follows.

Self-diagnosis should be deselected, and the temperature sensor A/D conversion select bit (ADEXICR.TSSA) and internal reference voltage A/D conversion select bit (S12AD1.ADEXICR.OCSA) should be set to 0.

Duplication of A/D conversion data is enabled by setting the channel numbers to be duplicated to the ADCSR.DBLANS[4:0] bits and setting the ADCSR.DBLE bit to 1. When the ADCSR.DBLE bit is set to 1, channel selection using the ADANSA0 register is invalid. In extended double trigger mode, the ADCSR.EXTRG bit should be set to 0, and the ADCSR.TRGE bit should be set to 1. Software trigger should not be used.

In extended double trigger mode, if two types of triggers have occurred simultaneously with TRGnAN or TRGnBN (n = 0, 4) or GTADTRAmN or GTADTRBmN (m = 0 to 2) selected, results are not sorted by the trigger sources and are stored in A/D data duplication register B (ADDBLDRB). Note that if a new trigger source is input during A/D conversion caused by another trigger source, the former new trigger source is ignored and sorting is performed by the latter trigger source.

Figure 30.8 shows an example of the basic operation in extended double trigger mode.

- (1) When the ADCSR.ADST bit is set to 1 (A/D conversion start) by TRG4AN input, A/D conversion is started on the single channel selected by the ADCSR.DBLANS[4:0] bits.
- (2) When A/D conversion is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy) and A/D data-duplication register A (ADDBLDRA).
- (3) The ADCSR.ADST bit is automatically cleared to 0 and the 12-bit A/D converter enters a wait state. Here, a scan end interrupt request is not generated irrespective of the ADCSR.ADIE bit setting (interrupt generation upon scanning completion enabled).
- (4) When the ADCSR.ADST bit is set to 1 (A/D conversion start) by TRG4BN input, A/D conversion is started on the single channel selected by the ADCSR.DBLANS[4:0] bits.
- (5) When A/D conversion is completed, the A/D conversion result is stored into A/D data duplication register (ADDBLDR) and A/D data duplication register B (ADDBLDRB).
- (6) If the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled), a scan end interrupt request is generated.
- (7) The ADCSR.ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion is completed. Then the 12-bit A/D converter enters a wait state.

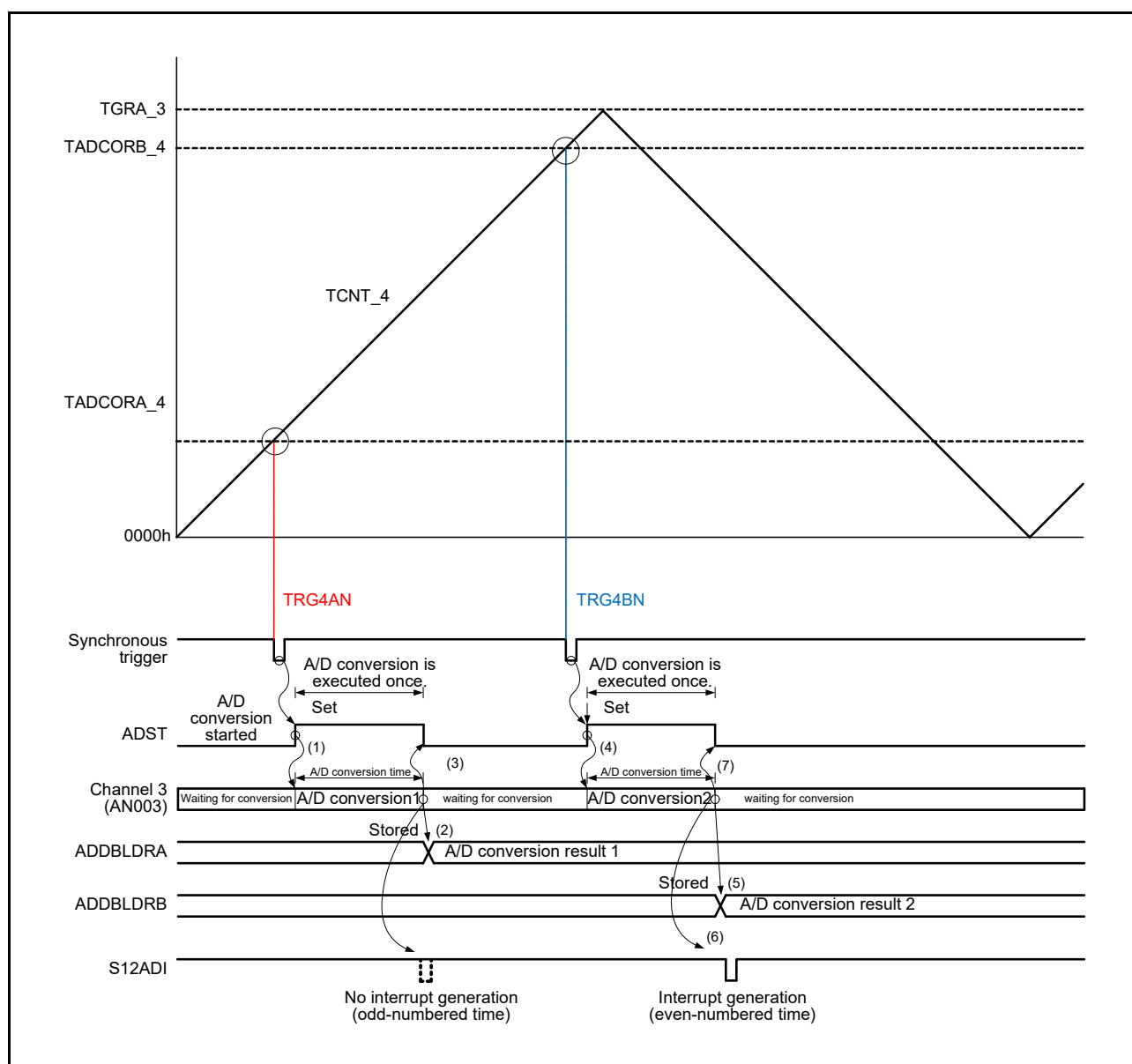


Figure 30.8 Example of Extended Operation in Double Trigger Mode (1)
(Duplication Selected for AN003, TRG4AN or TRG4BN Selected, First Trigger is TRG4AN)

Figure 30.9 shows an example of the operation in extended double trigger mode when there is another trigger input during A/D conversion.

Additionally, Figure 30.10 shows an example of the operation when the trigger GTADTRA0N, GTADTRB0N, GTADTRA1N, or GTADTRB1N are selected.

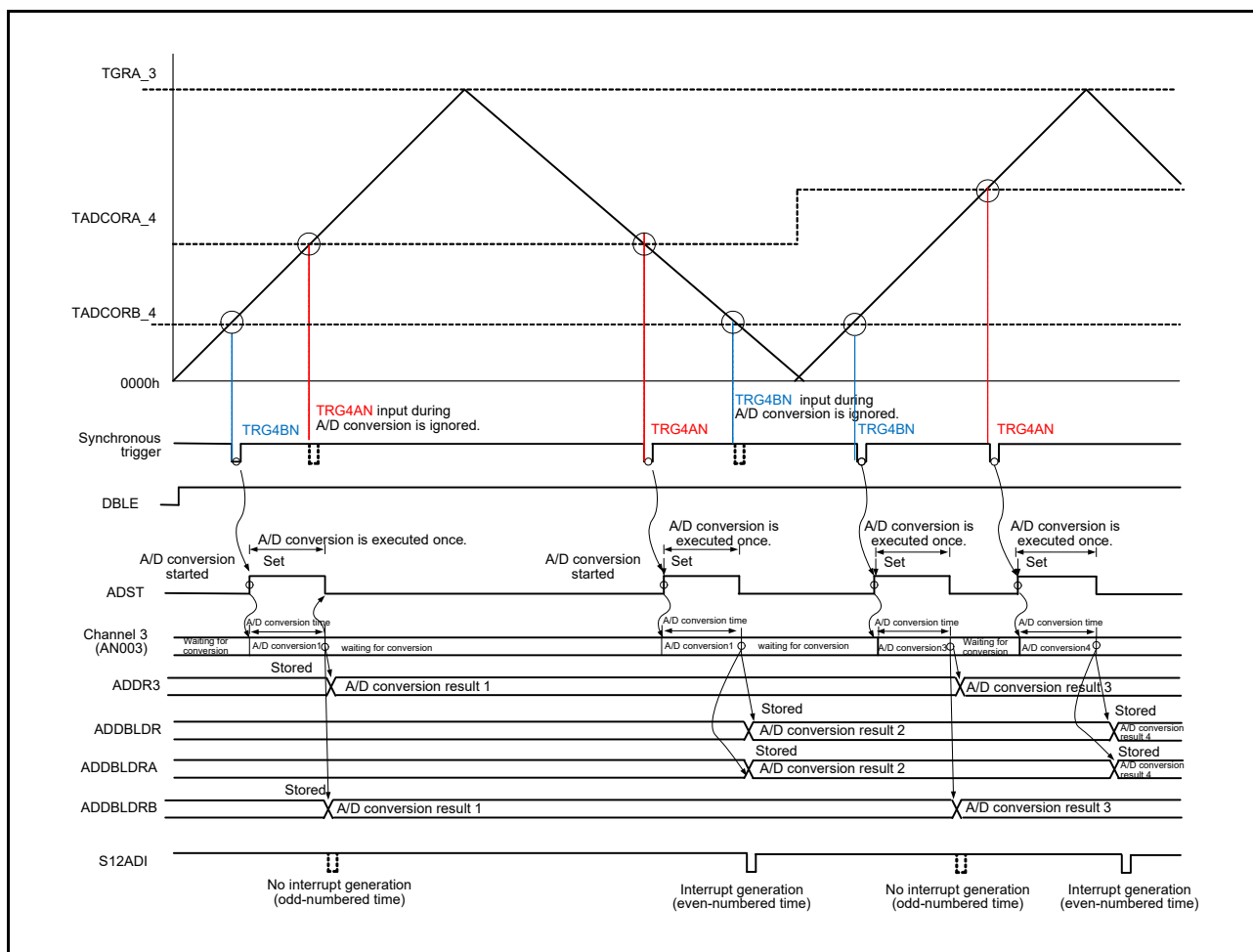


Figure 30.9 Example of Extended Operation in Double Trigger Mode (2)
 (Duplication Selected for AN003, TRG4AN or TRG4BN Selected, there is another trigger input during A/D conversion)

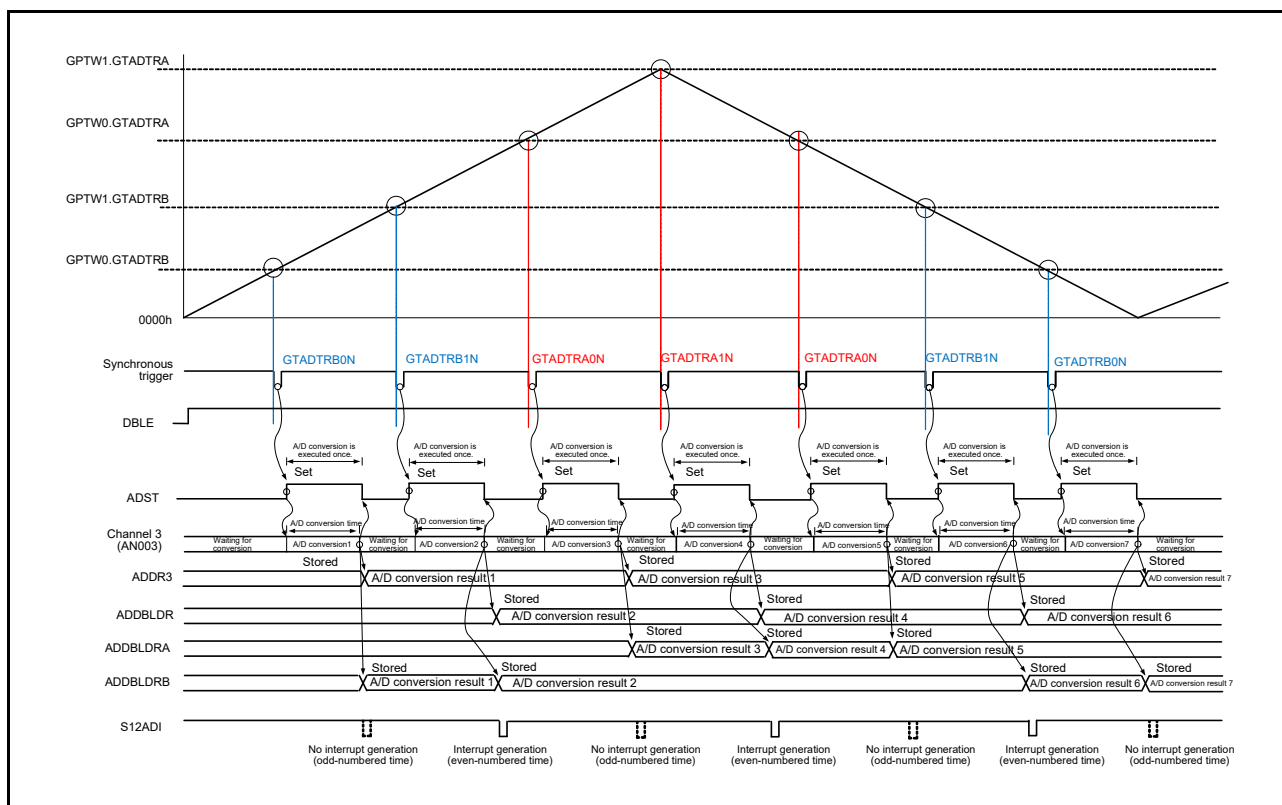


Figure 30.10 Example of Extended Operation in Double Trigger Mode (3)
(Duplication Selected for AN003, GTADTRA0N, GTADTRB0N, GTADTRA1N, or GTADTRB1N are Selected)

30.3.5 Continuous Scan Mode

30.3.5.1 Basic Operation

In basic operation of continuous scan mode, A/D conversion is performed repeatedly on the analog input of the specified channels as below.

In continuous scan mode, the temperature sensor output A/D conversion select bit (S12AD1.ADEXICR.TSSA) and the internal reference voltage A/D conversion select bit (S12AD1.ADEXICR.OCSA) should be set to 0 (deselected).

- (1) When the ADCSR.ADST bit is set to 1 (A/D conversion start) by software, or synchronous or asynchronous trigger input, A/D conversion is performed for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n.
- (2) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
- (3) When A/D conversion of all the selected channels is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
The 12-bit A/D converter sequentially starts A/D conversion for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n.
- (4) The ADCSR.ADST bit is not automatically cleared to 0 and steps 2 and 3 are repeated as long as the bit remains 1 (A/D conversion start). When the ADCSR.ADST bit is set to 0 (A/D conversion stop), A/D conversion stops and the 12-bit A/D converter enters a wait state.
- (5) When the ADST bit is later set to 1 (A/D conversion start), A/D conversion is started again for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n.

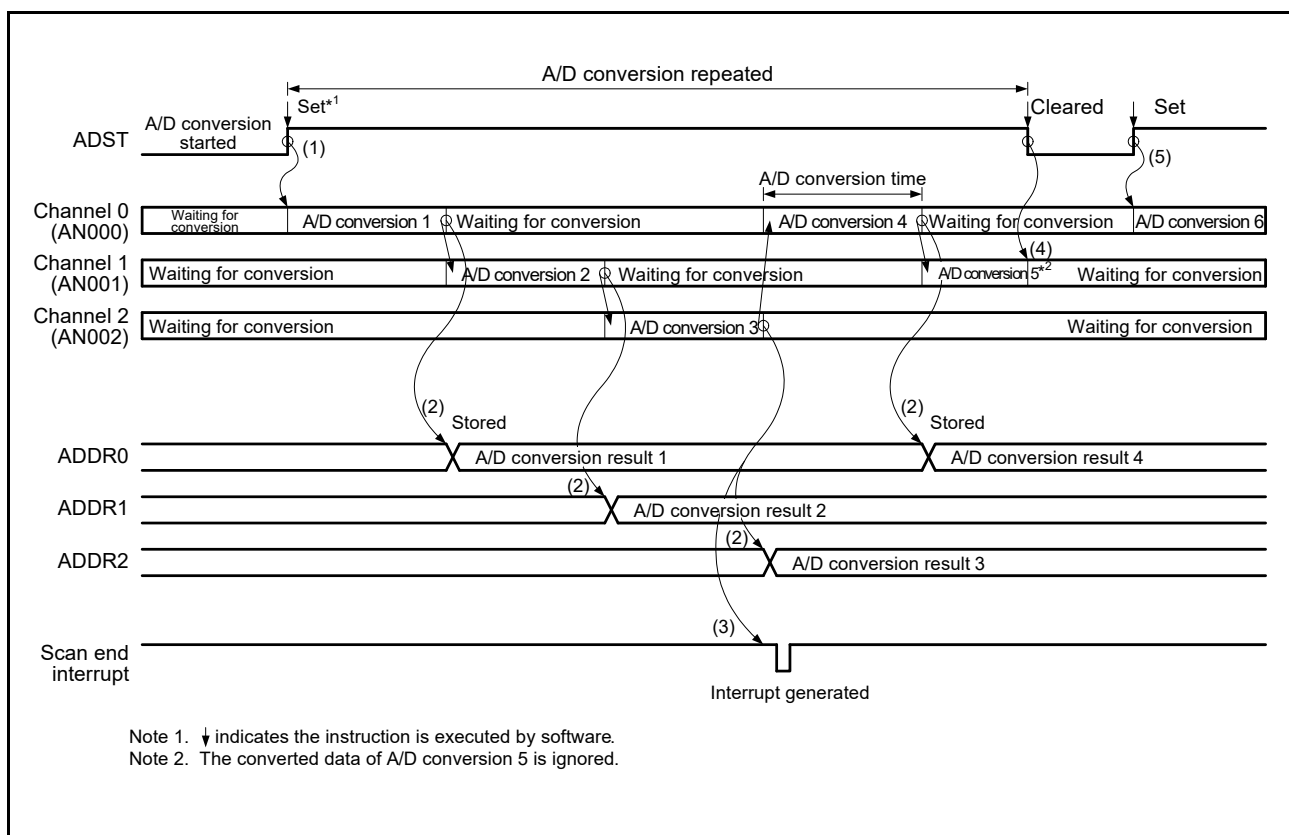


Figure 30.11 Example of Operation in Continuous Scan Mode (Basic Operation: AN000 to AN002 Selected)

30.3.5.2 Channel Selection and Self-Diagnosis

When channels and self-diagnosis are selected at the same time, A/D conversion is first performed for the reference voltage supplied to the 12-bit A/D converter, and then A/D conversion is performed on the analog input of the selected channels, which sequence is repeated as below. In continuous scan mode, the temperature sensor output A/D conversion select bit (S12AD1.ADEXICR.TSSA) and internal reference voltage A/D conversion select bit (S12AD1.ADEXICR.OCSA) should be set to 0 (deselected).

- (1) When the ADCSR.ADST bit is set to 1 (A/D conversion start) by software, or synchronous or asynchronous trigger input, A/D conversion for self-diagnosis is started first.
- (2) When A/D conversion for self-diagnosis is completed, the A/D conversion result is stored into the A/D self-diagnosis data register (ADRD). A/D conversion is then performed for ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n.
- (3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
- (4) When A/D conversion of all the selected channels is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled). At the same time, the 12-bit A/D converter starts A/D conversion for self-diagnosis and then starts A/D conversion on ANn channels selected by the ADANSA0 register, starting from the channel with the smallest number n.
- (5) The ADCSR.ADST bit is not automatically cleared and steps 2 to 4 are repeated as long as the bit remains 1. When the ADST bit is set to 0 (A/D conversion stop), A/D conversion stops and the 12-bit A/D converter enters a wait state.
- (6) When the ADCSR.ADST bit is later set to 1 (A/D conversion start), the A/D conversion for self-diagnosis is started again.

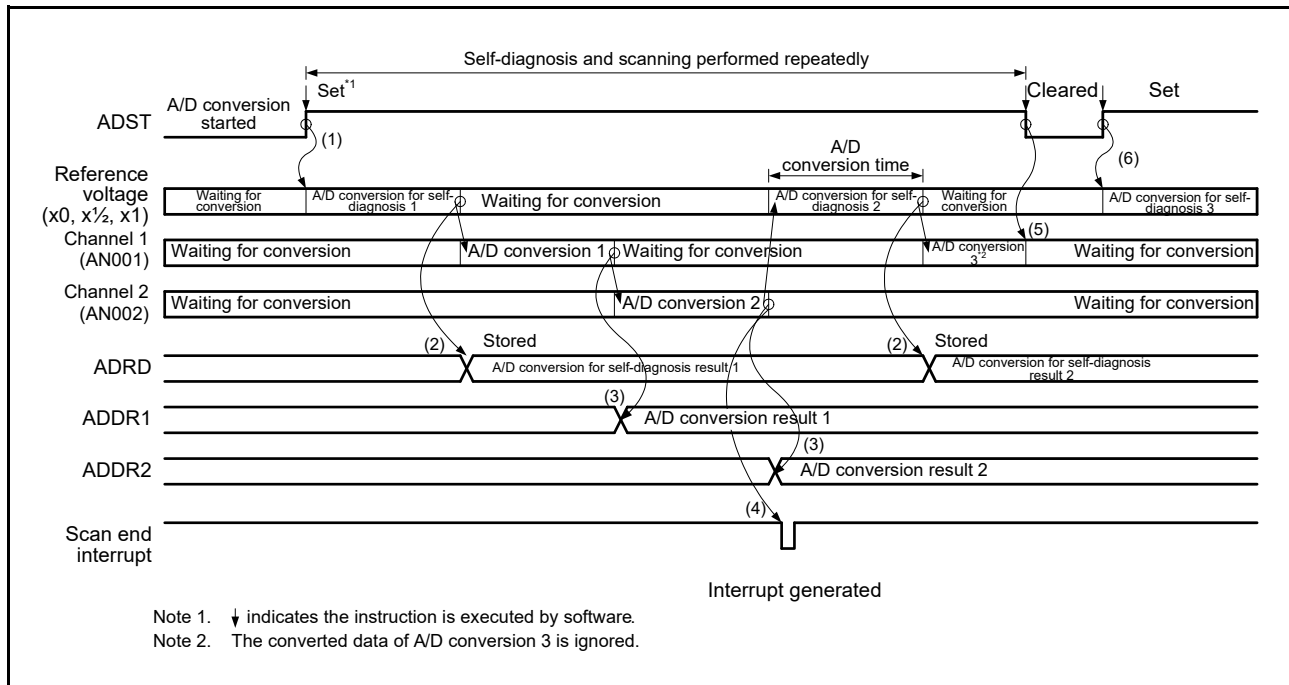


Figure 30.12 Example of Operation in Continuous Scan Mode
(Basic Operation; AN001 and AN002 Selected + Self-Diagnosis)

30.3.6 Group Scan Mode

30.3.6.1 Basic Operation

Either two (groups A and B) or three (groups A, B, and C) can be selected as the number of the groups to be used in group scan mode.

In basic operation of group scan mode, A/D conversion is performed once on the analog inputs of all the specified channels in groups A and B, or groups A, B, and C after scan is started by a synchronous trigger as below. Scan operation of each group is similar to the scan operation in single scan mode.

The synchronous triggers of groups A and B, or groups A, B, and C can be selected using the TRSA[5:0], TRSB[5:0], and TRSC[5:0] bits in ADSTRGR, respectively. Different triggers should be used for each group A, B, and C so that scanning of groups A, B, and C does not occur simultaneously. Software trigger should not be used.

The channels to be scanned are selected using register ADANSA0 for group A, register ADANSB0 for group B, and register ADANSC0 for group C.

In group scan mode, the temperature sensor output A/D conversion select bit (S12AD1.ADEXICR.TSSA) and internal reference voltage A/D conversion select bit (S12AD1.ADEXICR.OCSA) should be set to 0 (deselected).

When self-diagnosis is selected in group scan mode, self-diagnosis is separately executed for groups A and B, or groups A, B, and C.

The following describes operation in group scan mode using a trigger from the MTU. The TRG4AN, TRG4BN, and TRG4ABN triggers from the MTU are assumed to be used to start conversion of groups A, B and C, respectively.

- (1) Scanning of group A is started by the TRG4AN trigger from the MTU.
- (2) When group A scanning is completed, a scan end interrupt is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (3) Scanning of group B is started by the TRG4BN trigger from the MTU.
- (4) When group B scanning is completed, a group B scan end interrupt is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan completion enabled).
- (5) Scanning of group C is started by the TRG4ABN trigger from the MTU.
- (6) When group C scanning is completed, a group C scan end interrupt is generated if the ADGCTRGR.GCADIE bit is 1 (interrupt generation upon group C scan completion enabled).

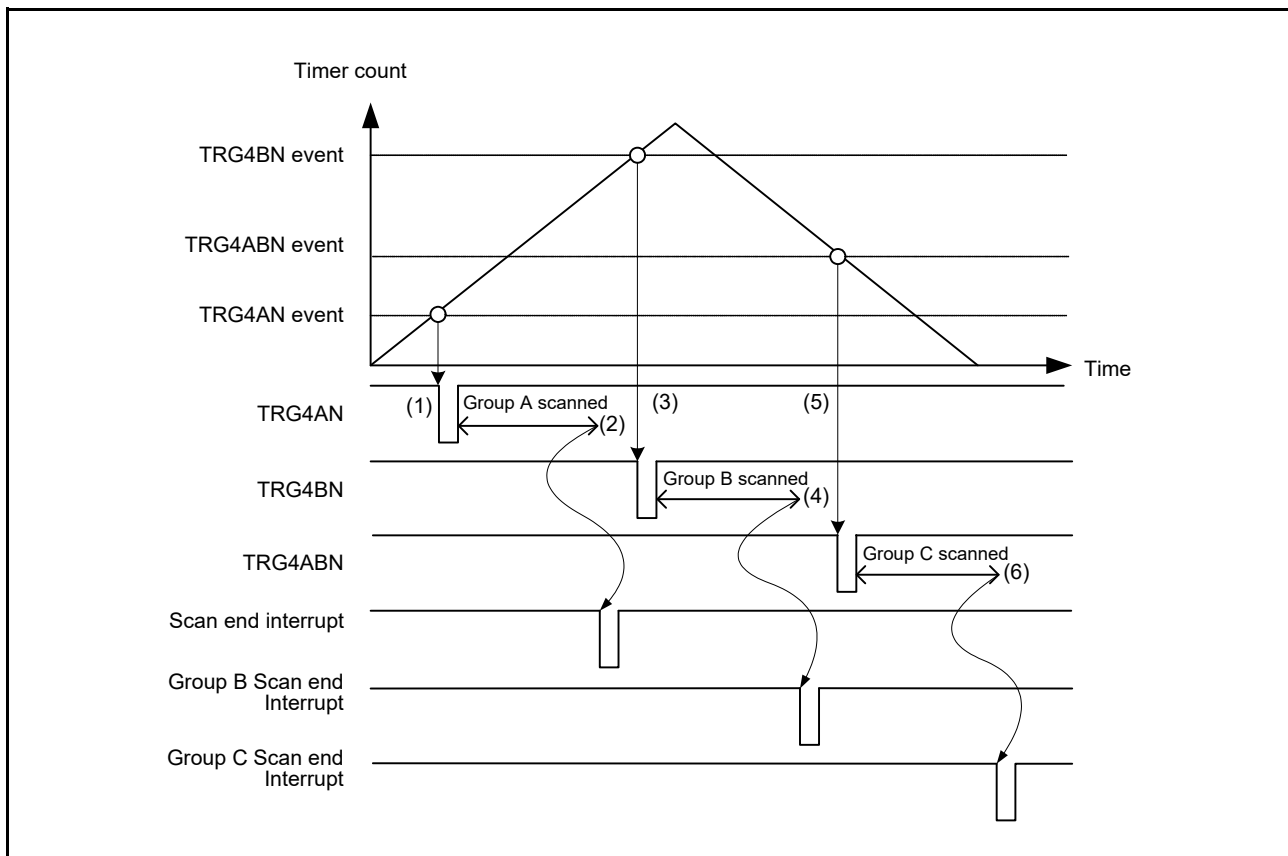


Figure 30.13 Example of Operation in Group Scan Mode
(Basic Operation: Synchronous Triggers from MTU Used)

30.3.6.2 A/D Conversion in Double Trigger Mode

When double trigger mode is selected in group scan mode, two rounds of single scan operation started by a synchronous trigger are performed as a sequence for group A. For groups B and C, single scan operation started by a synchronous trigger is performed once.

In group scan mode, the synchronous triggers of groups A and B, or groups A, B, and C can be selected using the TRSA[5:0], TRSB[5:0], and TRSC[5:0] bits in ADSTRGR, respectively. Different triggers should be used for each group A, B, and C so that scanning of groups A, B, and C does not occur simultaneously. Software trigger and asynchronous trigger should not be used. When the synchronous trigger corresponding to the extended double trigger mode shown in Table 30.8 in section 30.2.12 is selected in the ADSTRGR.TRSA[5:0] bits as the synchronous trigger of the group A, operation is in the extended double trigger mode. The channels to be scanned are selected using bits ADCSR.DBLANS[4:0] for group A, register ADANSB0 for group B, and register ADANSC0 for group C.

In group scan mode, the temperature sensor output A/D conversion select bit (S12AD1.ADEXICR.TSSA) and internal reference voltage A/D conversion select bit (S12AD1.ADEXICR.OCSA) should be set to 0 (deselected).

When double trigger mode is selected in group scan mode, self-diagnosis cannot be selected.

Duplication of A/D conversion data is enabled by setting the channel numbers to be duplicated to the ADCSR.DBLANS[4:0] bits and setting the ADCSR.DBLE bit to 1.

The following describes operation in group scan mode with double trigger mode using a synchronous trigger from the MTU. The TRG4ABN, TRGA0N, and TRGA1N triggers from the MTU are assumed to be used to start conversion of groups A, B, and C, respectively.

- (1) Scanning of group C is started by the TRGA1N trigger from the MTU.
- (2) When group C scanning is completed, a group C scan interrupt is generated if the ADGCTRGR.GCADIE bit is 1 (interrupt generation upon group C scan completion enabled).
- (3) Scanning of group B is started by the TRGA0N trigger from the MTU.
- (4) When group B scanning is completed, a group B scan end interrupt is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan enabled).
- (5) The first scanning of group A is started by the first TRG4ABN trigger from the MTU.
- (6) When the first scanning of group A is completed, the conversion result is stored into the corresponding A/D data register (ADDRy); a scan end interrupt request is not generated irrespective of the ADIE bit setting in ADCSR.
- (7) The second scanning of group A is started by the second TRG4ABN trigger from the MTU.
- (8) When the second scanning of group A is completed, the conversion result is stored into ADDBLDR. a scan end interrupt is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (9) The second scanning of group B is started by the second TRGA0N trigger from the MTU.
- (10) When the second scanning of group B is completed, a group B scan interrupt is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan completion enabled).
- (11) The second scanning of group C is started by the second TRGA1N trigger from the MTU.
- (12) When the second scanning of group C is completed, a group C scan interrupt is generated if the ADGCTRGR.GCADIE bit is 1 (interrupt generation upon group C scan completion enabled).

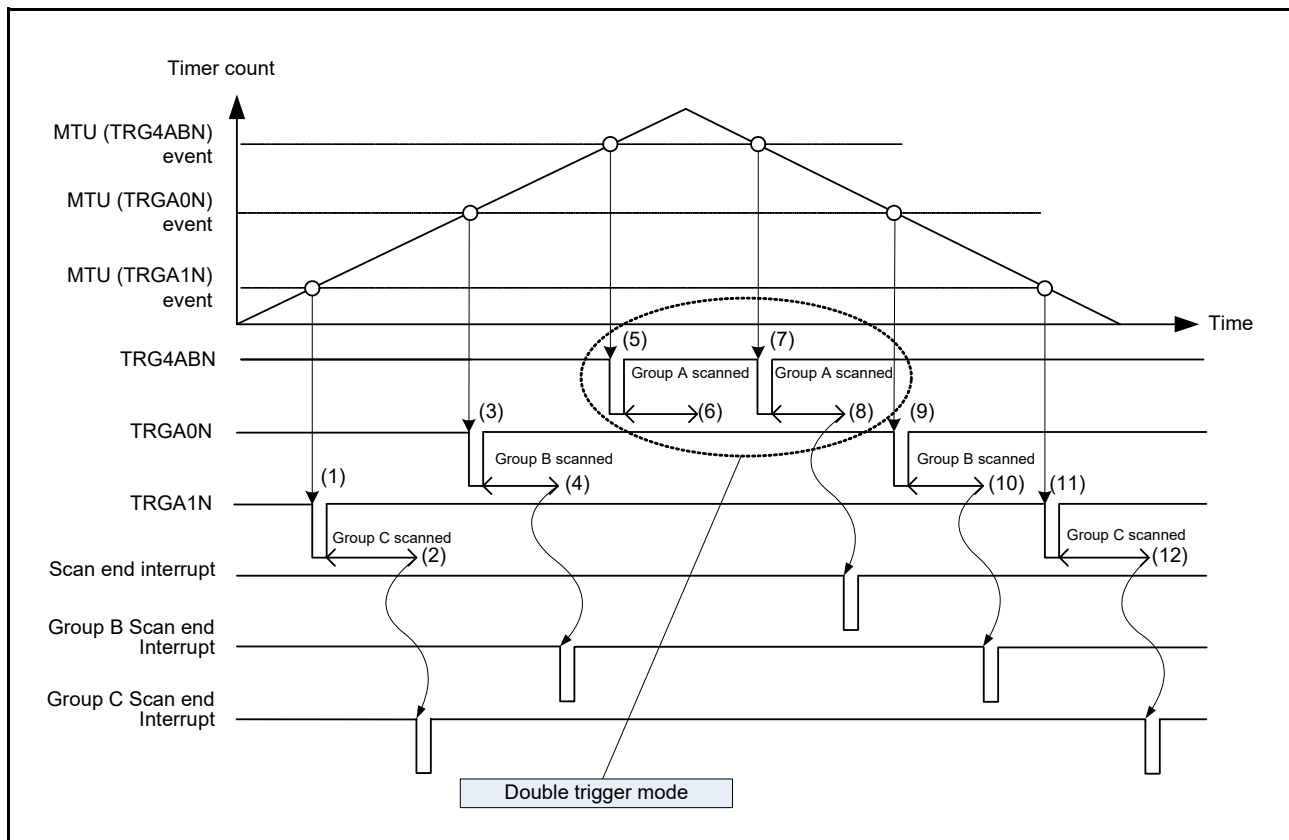


Figure 30.14 Example of Operation in Group Scan Mode with Double Trigger Mode
(Basic Operation: Synchronous Triggers from MTU Used)

30.3.6.3 Operation under Group Priority Control

Setting the ADGSPCR.PGS bit to 1 in group scan mode makes operation proceed under group priority control. The group priority order is group A > group B > group C. Either two (groups A and B) or three (groups A, B, and C) can be selected as the number of the groups to be used in group scan mode. When setting the ADGSPCR.PGS bit to 1, follow the procedure described in Figure 30.15. If the procedure is not followed, proper scanning operation and data storage are not guaranteed.

In basic operation of group scan mode, if group A, B, or C is scanning, all other trigger inputs are ignored. Under group priority control, if a priority group trigger is input during A/D conversion for the low-priority group, A/D conversion for the low-priority group is discontinued and A/D conversion for the priority group proceeds.

If the ADGSPCR.GBRSCN bit is 0, the converter enters a wait state for the low-priority group on completion of the A/D conversion for the priority group.

The trigger input for the low-priority group during scanning is ignored.

If the ADGSPCR.GBRSCN bit is 1, the converter automatically restarts scanning for the low-priority group from the head of the group after A/D conversion for the priority group.

Also, the trigger input for the low-priority group generated during scanning for the priority group is enabled, and the converter automatically restarts scanning for the low-priority group after scanning for the priority group.

When the ADGSPCR.LGRRS bit is 0 while the ADGSPCR.GBRSCN bit is 1, the converter restarts scanning for the low-priority group from the head of the group. When the ADGSPCR.LGRRS bit is 1, the converter restarts scanning for the low-priority group from the channel on which scanning is discontinued.

However, when self-diagnosis is used, the converter restarts scanning from the channel on which scanning is discontinued after self-diagnosis is completed.

Table 30.13 summarizes operations in response to the input of a trigger during scanning with the settings of the ADGSPCR.GBRSCN bit.

When the ADGSPCR.GBRP bit is set to 1, scan operations in the lowest-priority group are continuously performed.

For the trigger settings in group scan mode, select a synchronous trigger for group A using the ADSTRGR.TRSA[5:0] bits, select a synchronous trigger different from that of group A for group B using the ADSTRGR.TRSB[5:0] bits, and select a synchronous trigger different from those of groups A and B for group C using the ADGCTRGR.TRSC[5:0] bits. Set the ADSTRGR.TRSB[5:0] bits to 3Fh when setting group scan mode for two groups (the ADGCTRGR.GRCE bit to 0) and setting the ADGSPCR.GBRP bit to 1.

Set the ADGCTRGR.TRSC[5:0] bits to 3Fh when setting group scan mode for three groups (the ADGCTRGR.GRCE bit to 1) and setting the ADGSPCR.GBRP bit to 1.

Furthermore, as targets for scanning, select channels for group A using the ADANSA0 register, select channels for group B using the ADANSB0 register, and select channels for group C using the ADANSC0 register.

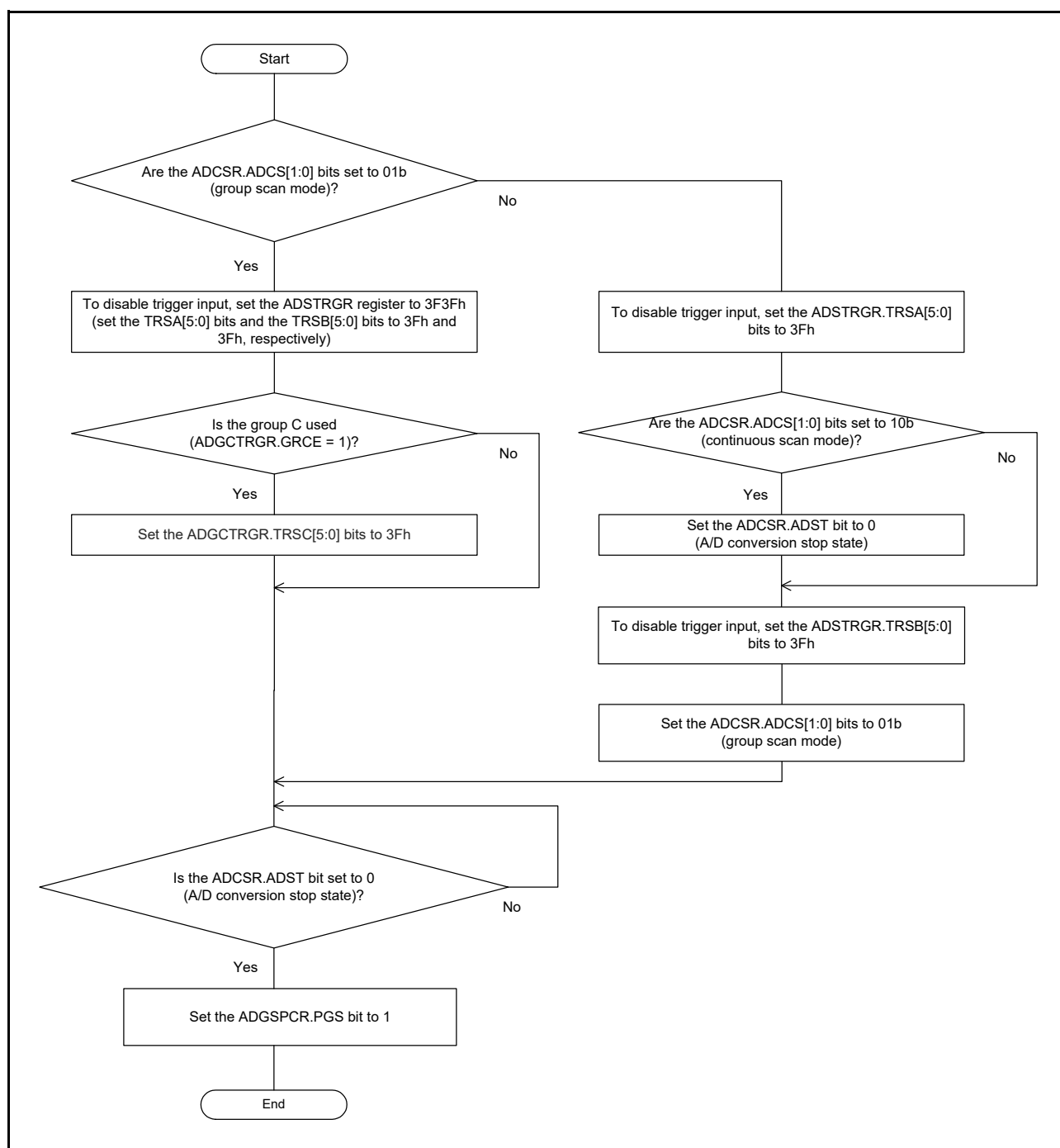


Figure 30.15 Flow of Setting the ADGSPCR.PGS Bit

Table 30.13 Control of Scanning Operations According to the Settings of the ADGSPCR.GBRSCN Bit

Scanning Operation	Trigger Input	ADGSPCR.GBRSCN = 0	ADGSPCR.GBRSCN = 1
When A/D conversion for group A is in progress	Input of trigger for group A	Trigger input is ineffective.	Trigger input is ineffective.
	Input of trigger for group B	Trigger input is ineffective.	A/D conversion is performed on group B after A/D conversion on group A is completed.
	Input of trigger for group C	Trigger input is ineffective.	A/D conversion is performed on group C after A/D conversion on group A is completed.
When A/D conversion for group B is in progress	Input of trigger for group A	A/D conversion in progress for group B discontinued and conversion for group A starts.	- A/D conversion in progress for group B is discontinued and conversion for group A starts. - A/D conversion for group B starts after conversion for group A is completed.
	Input of trigger for group B	Trigger input is ineffective.	Trigger input is ineffective.
	Input of trigger for group C	Trigger input is ineffective.	A/D conversion is performed on group C after A/D conversion on group B is completed.
When A/D conversion for group C is in progress	Input of trigger for group A	A/D conversion in progress for group C discontinued and conversion for group A starts.	- A/D conversion in progress for group C is discontinued and conversion for group A starts. - A/D conversion for group C starts after conversion for group A is completed.
	Input of trigger for group B	A/D conversion in progress for group C is discontinued and conversion for group B starts.	- A/D conversion in progress for group C is discontinued and conversion for group B starts. - A/D conversion for group C starts after conversion for group B is completed.
	Input of trigger for group C	Trigger input is ineffective.	Trigger input is ineffective.

When using group priority operation mode, refer to the following tables to select the desirable operating mode and set the registers.

Table 30.14 Group Priority Operation Setting and Operating Mode for Two Groups (ADGSPCR.PGS = 1, ADGCTRGR.GRCE = 0)

ADGSPCR			Operation
GBRSCN	LGRRS	GBRP	
0	x	0	Group priority operation for two groups (groups A and B) When a group A trigger is input, group B scan is completed (not restarted)
1	0	0	Group priority operation for two groups (groups A and B) After group B scan is discontinued, group B scan is restarted from the head of the channel selected with the ADANSB0 register after group A scan is completed.
1	1	0	Group priority operation for two groups (groups A and B) After group B scan is discontinued, group B scan is restarted from the channel on which scan was discontinued*1, among the channels selected with the ADANSB0 register after group A scan is completed.
x	0	1	Group priority operation for two groups (groups A and B) Single scan for group B is started continuously without start trigger input. After group B scan is discontinued, single scan is restarted from the head of the channel selected with the ADANSB0 register after scan for group A is completed.
1	1	1	Group priority operation for two groups (groups A and B) Single scan for group B is started continuously without start trigger input. After group B scan was discontinued, single scan is restarted from the channel on which scan was discontinued*1, among the channels selected with the ADANSB0 register after group A scan is completed.

x = Don't care

Note 1. When self-diagnosis is used (ADCER.DIAGM = 1), A/D conversion of the discontinued channel is started after self-diagnosis.

Table 30.15 Group Priority Operation Setting and Operating Mode for Three Groups
(ADGSPCR.PGS = 1, ADGCTRGR.GRCE = 1)

ADGSPCR			Operation
GBRSCN	LGRRS	GBRP	
0	x	0	Group priority operation for three groups (groups A, B, and C) - When a group A trigger is input, group B scan is completed (not restarted) - When a trigger for group A or B is input, group C scan is not completed (not restarted).
0	x	1	Group priority operation for three groups (groups A, B, and C) - When a group A trigger is input, group B scan is completed (not restarted) - Single scan for group C is started continuously without start trigger input. After group C scan is discontinued, scan is restarted from the head of the channel selected with the ADANSC0 register after scan for groups A and B is completed.
1	0	0	Group priority operation for three groups (groups A, B, and C) - After group B scan is discontinued, scan is restarted from the head of the channel selected with the ADANSB0 register after group A scan is completed. - After group C scan is discontinued, scan is restarted from the head of the channel selected with the ADANSC0 register after scan for groups A and B is completed.
1	1	0	Group priority operation for three groups (groups A, B, and C) - After group B scan is discontinued, scan is restarted from the head of the channel selected with the ADANSB0 register after group A scan is completed. - After group C scan is discontinued, scan is restarted from the channel on which scan was discontinued*1, among the channels selected with the ADANSC0 register after scan for groups A and B is completed.
1	0	1	Group priority operation for three groups (groups A, B, and C) - After group B scan is discontinued, scan is restarted from the head of the channel selected with the ADANSB0 register after group A scan is completed. - Single scan for group C is started continuously without start trigger input. After group C scan is discontinued, single scan is restarted from the head of the channel selected with the ADANSC0 register after scan for groups A and B is completed.
1	1	1	Group priority operation for three groups (groups A, B, and C) - After scanning for group B is discontinued, once scanning for group A completes, scanning is resumed starting from the channel on which scanning was discontinued*1, among those channels selected with the ADANSB0 register. - Single scan for group C is started continuously without start trigger input. After group C scan is discontinued, single scan is restarted from the channel on which scan was discontinued*1, among the channels selected with the ADANSC0 register after scan for groups A and B is completed.

x = Don't care

Note 1. When self-diagnosis is used (ADCER.DIAGM = 1), A/D conversion of the discontinued channel is started after self-diagnosis.

(1) Group Priority Operation for Two Groups (ADGSPCR.PGS = 1, ADGCTRGR.GRCE = 0)

The following examples 1 to 5 show group priority operation in group scan mode (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0) when channel 0 is selected for group A and channels 1 to 3 are selected for group B.

Operation example 1: Group A trigger input during group B scan, with rescan setting

- (1) When a group B trigger input sets the ADCSR.ADST bit to 1 (A/D conversion start), scan for the ANn channels selected in the ADANSB0 register, starts from the channel with the smallest number n.
- (2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (3) If a group A trigger is input during scan for group B, group B scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group A selected in the ADANSA0 register, starts from the channel with the smallest number n.
If A/D conversion is not completed when scan was discontinued, the result is not stored in the corresponding A/D data register (ADDRy).
- (4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (5) A scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (6) If the ADGSPCR.GBRSCN bit is 1 (scan for the group is restarted after having been discontinued due to group priority operation), scan for the ANn channels of group B selected in the ADANSB0 register, restarts from the channel with the smallest number n while the ADCSR.ADST bit remains 1.
- (7) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (8) A group B scan end interrupt request is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan completion enabled).
- (9) The ADCSR.ADST bit is automatically cleared to 0 when scan of all selected channels is completed, and the 12-bit A/D converter enters a wait state.

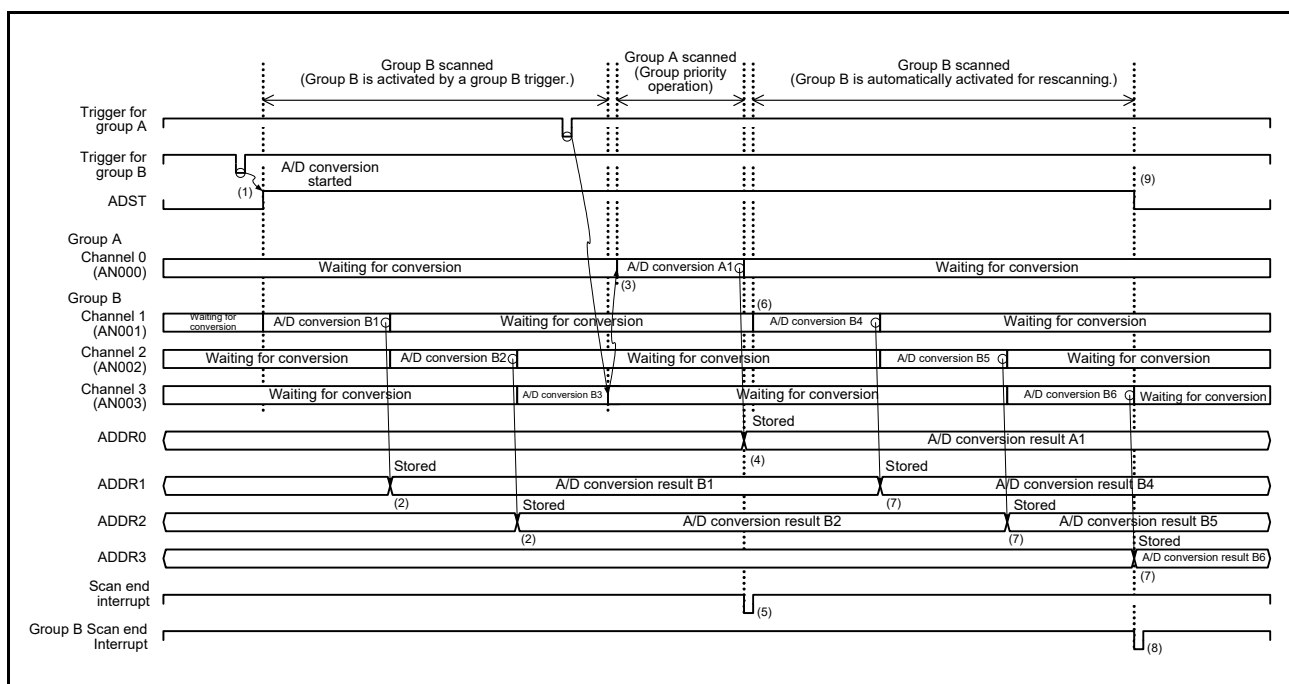


Figure 30.16 Example 1 of Group Priority Operation: Group A Trigger Input during Group B Scan, With Rescan Setting (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0)

Operation example 2: Group A trigger input during group B rescan, with rescan setting

Figure 30.17 shows an example when a group A trigger is input during rescan operation on group B.

If a group A trigger is input, scan for group A starts even while rescan operation is in progress. Scan for group B starts after scan for group A is completed.

Operations of the ADCSR.ADST bit, storing to the A/D conversion result in the A/D data register (ADDRy), and interrupt requests are the same as example 1.

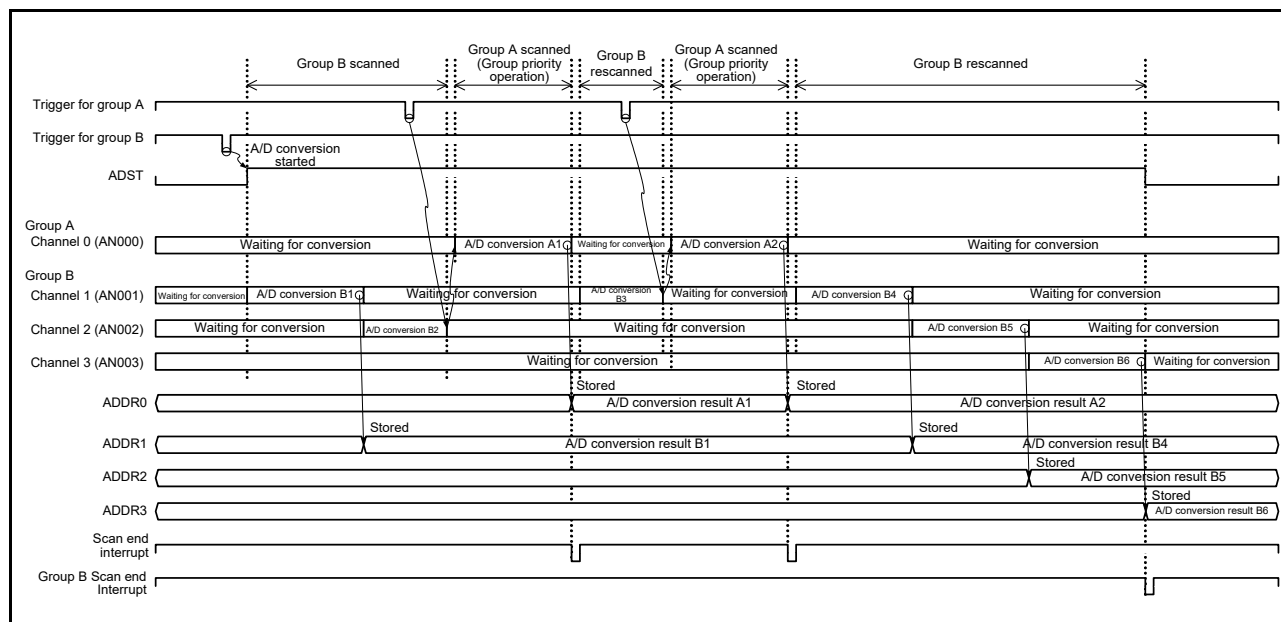


Figure 30.17 Example 2 of Group Priority Operation: Group A Trigger Input during Group B Rescan, With Rescan Setting ($ADGSPCR.GBRSCN = 1$, $ADGSPCR.GBRP = 0$, $ADGSPCR.LGRRS = 0$)

Operation example 3: Group B trigger input during group A scan, with rescan setting

The following describes an example when a group B trigger is input during scan operation on group A when the ADGSPCR.GBRSCN bit is 1 (scan for the group is restarted after having been discontinued due to group priority operation).

If the ADGSPCR.GBRSCN bit is 0, all group B triggers that are input during scan operation on group A are disabled.

- (1) When a group A trigger input sets the ADCSR.ADST bit to 1 (A/D conversion start), scan for the ANn channels of group A selected in the ADANSA0 register, starts from the channel with the smallest number n.
- (2) If a group B trigger is input during scan for group A, scan for group B can be started.
- (3) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (4) After scan for group A is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (5) After scan for group A is completed, scan for the ANn channels of group B selected in the ADANSB0 register, starts from the channel with the smallest number n while the ADCSR.ADST bit remains 1. When a group A trigger is input during scan for group B, group A scan starts as in example 1, and group B scan starts after group A scan is completed.
- (6) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (7) After scan for group B is completed, a group B scan end interrupt request is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan completion enabled).
- (8) The ADCSR.ADST bit is automatically cleared to 0 when scan of all selected channels is completed, and the 12-bit A/D converter enters a wait state.

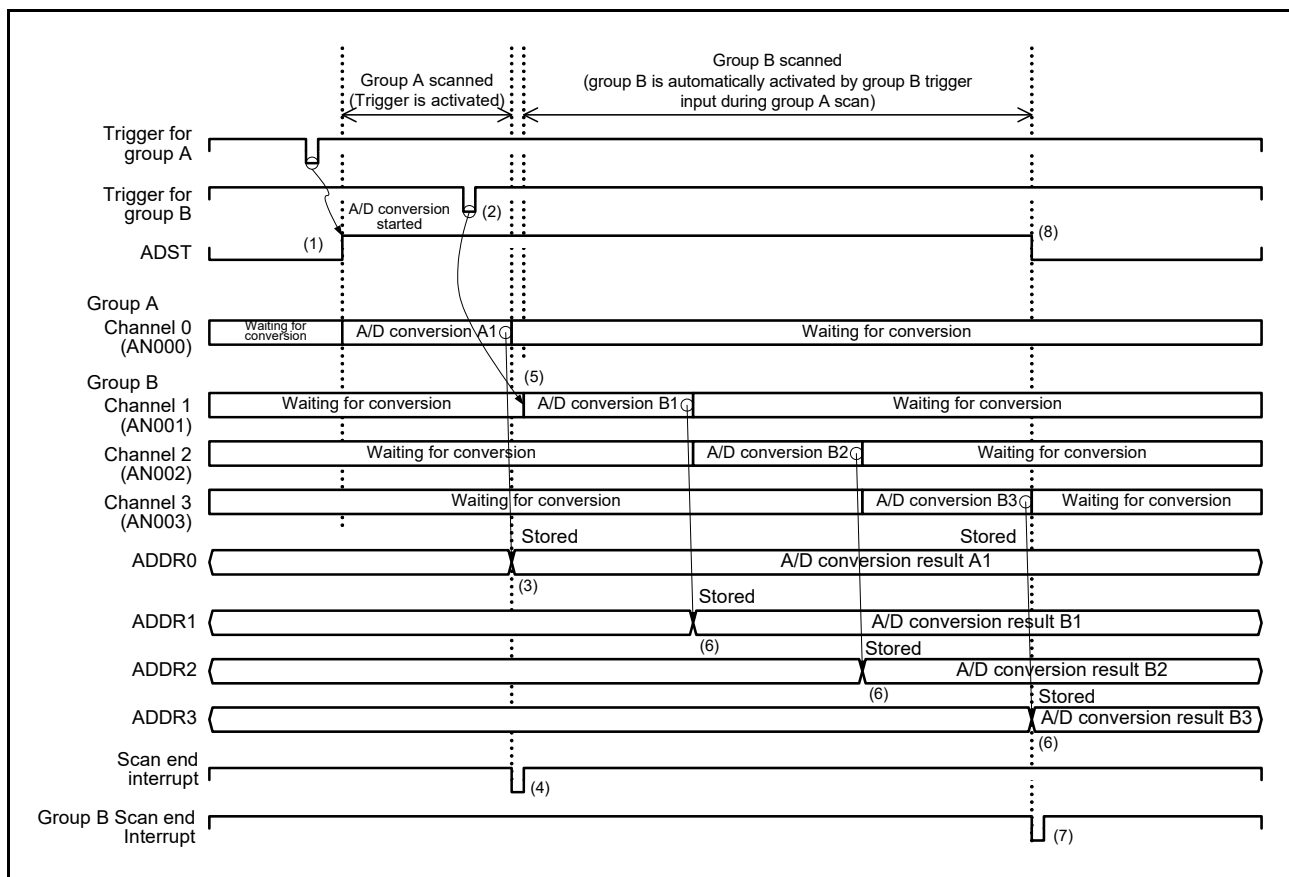


Figure 30.18 Example 3 of Group Priority Operation: Group B Trigger Input during Group A Scan, With Rescan Setting (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0)

Operation example 4 shows group priority operation in group scan mode (ADGSPCR.GBRSCN = 0, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0) when channel 0 is selected for group A and channels 1 to 3 are selected for group B.

Operation example 4: Group A trigger input during group B scan, without rescan setting

- (1) When a group B trigger input sets the ADCSR.ADST bit to 1 (A/D conversion start), scan for the ANn channels of group B selected in the ADANSB0 register, starts from the channel with the smallest number n.
- (2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (3) If a group A trigger is input during scan for group B, group B scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group A selected in the ADANSA0 register, starts from the channel with the smallest number n.
If A/D conversion is not completed when scan was discontinued, the result is not stored in the corresponding A/D data register (ADDRy).
- (4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (5) After scan for group A is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (6) The ADST bit is automatically cleared to 0 when scan for group A is completed, and the 12-bit A/D converter enters a wait state. Scan for group B is not started until the next group B trigger is input.

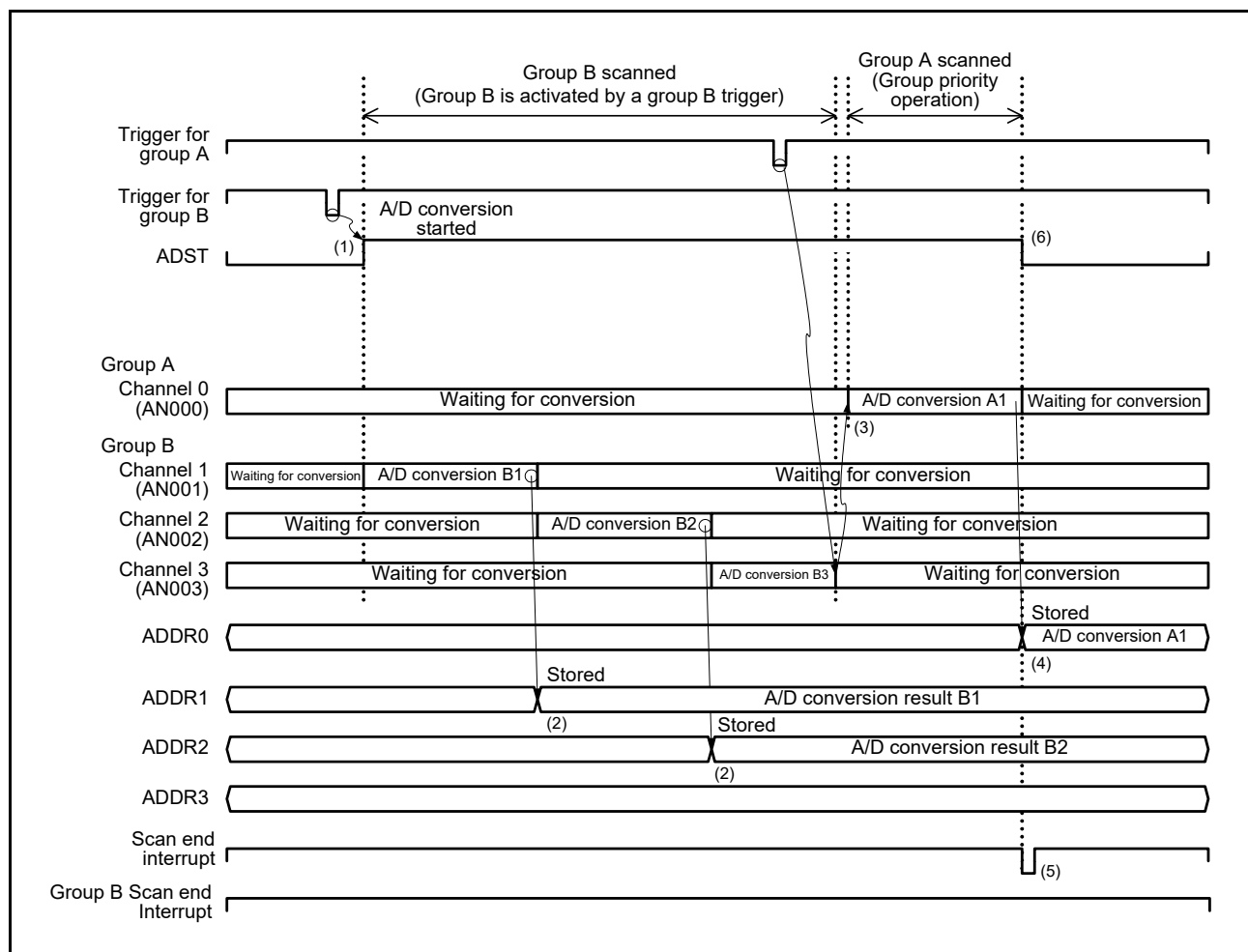


Figure 30.19 Example 4 of Group Priority Operation: Group A Trigger Input during Group B Scan, Without Rescan Setting (ADGSPCR.GBRSCN = 0, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0)

Operation example 5 shows group priority operation in group scan mode (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 1, ADGSPCR.LGRRS = 0) when channel 0 is selected for group A and channels 1 and 2 are selected for group B. When the ADGCTRGR.GRCE bit is set to 1, single scan mode is continuously operated on group C and scan for group B is started by trigger input.

Operation example 5: Continuous single scan operation on group B

- (1) When setting ADGSPCR.GBRP to 1 sets the ADCSR.ADST bit to 1 (A/D conversion start), scan for the ANn channels selected in the ADANSB0 register, starts from the channel with the smallest number n.
- (2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (3) If a group A trigger is input during scan for group B, group B scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group A selected in the ADANSA0 register, starts from the channel with the smallest number n. If A/D conversion is not completed when scan was discontinued, the result is not stored in the corresponding A/D data register (ADDRy).
- (4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (5) After scan for group A is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (6) If the ADGSPCR.GBRP bit is set to 1 (single scan is continuously operated), scan for the ANn channels of group B selected in the ADANSB0 register, restarts from the channel with the smallest number n while the ADCSR.ADST bit remains 1.
- (7) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (8) A group B scan end interrupt request is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan completion enabled).
- (9) If the ADGSPCR.GBRP bit is set to 1 (single scan is continuously operated), scan for the ANn channels of group B selected in the ADANSB0 register, restarts from the channel with the smallest number n while the ADCSR.ADST bit remains 1.

To continuously operate single scan for group B, disable trigger input for group B.

Steps 6 to 9 are repeated as long as the ADGSPCR.GBRP bit remains 1. Clearing of the ADCSR.ADST bit to 0 is prohibited while the ADGSPCR.GBRP bit is set to 1. To forcibly stop scanning when ADGSPCR.GBRP = 1, follow the procedures for clear operation by software through the ADCSR.ADST bit shown in section 30.7.2, Notes on Stopping A/D Conversion.

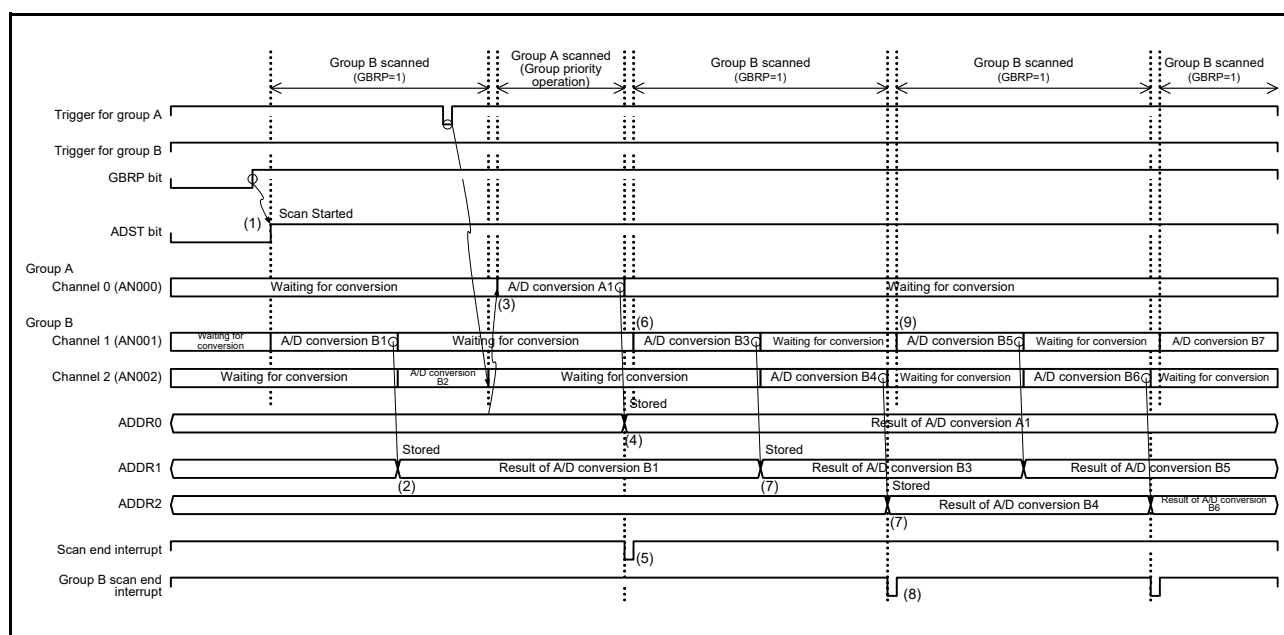


Figure 30.20 Example 5 for Group Priority Operation: Continuous Single Scan Operation on Group B
(ADGSPCR.GBRP = 1, ADGSPCR.LGRRS = 1, ADGCTRGR.GRCE = 0)

(2) Group Priority Operation for Three Groups (ADGSPCR.PGS = 1, ADGCTRGR.GRCE = 1)

The following examples 1 to 5 show group priority operation in group scan mode (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 1) when channel 0 is selected for group A, channels 1 and 2 are selected for group B, and channels 3 and 4 are selected for group C.

The priority groups mean groups A and B for group C and group A for group B.

Operation example 1: Priority group trigger input during low-priority group scan, with rescan setting

- (1) When a group C trigger input sets the ADCSR.ADST bit to 1 (A/D conversion start), scan for the ANn channels selected in the ADANSC0 register, starts from the channel with the smallest number n.
- (2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (3) If a group B trigger is input during scan for group C, group C scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group B selected in the ADANSB0 register, starts from the channel with the smallest number n. On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (5) If a group A trigger is input during scan for group B, group B scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group A selected in the ADANSA0 register, starts from the channel with the smallest number n. If A/D conversion is not completed when scan was discontinued, the result is not stored in the corresponding A/D data register (ADDRy).
- (6) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (7) A scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (8) If the ADGSPCR.GBRSCN bit is 1 (scan for the group is restarted after having been discontinued due to group priority operation), scan for the ANn channels of group B selected in the ADANSB0 register, restarts from the channel with the smallest number n while the ADCSR.ADST bit remains 1. If the ADGSPCR.LGRRS bit is set to 1 at this time, scan for group B starts from the channel on which A/D conversion was discontinued.
- (9) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (10) A group B scan end interrupt request is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan completion enabled).
- (11) If the ADGSPCR.GBRSCN bit is 1, scan for the ANn channels of group C selected in the ADANSC0 register, restarts from the channel with the smallest number n while the ADCSR.ADST bit remains 1. If the ADGSPCR.LGRRS bit is set to 1 at this time, scan for group C starts from the channel on which A/D conversion was discontinued.
- (12) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (13) A group C scan end interrupt is generated if the ADGCTRGR.GCADIE bit is 1 (interrupt generation upon group C scan completion enabled).
- (14) The ADST bit is automatically cleared to 0 when scan for group A is completed, and the 12-bit A/D converter enters a wait state.

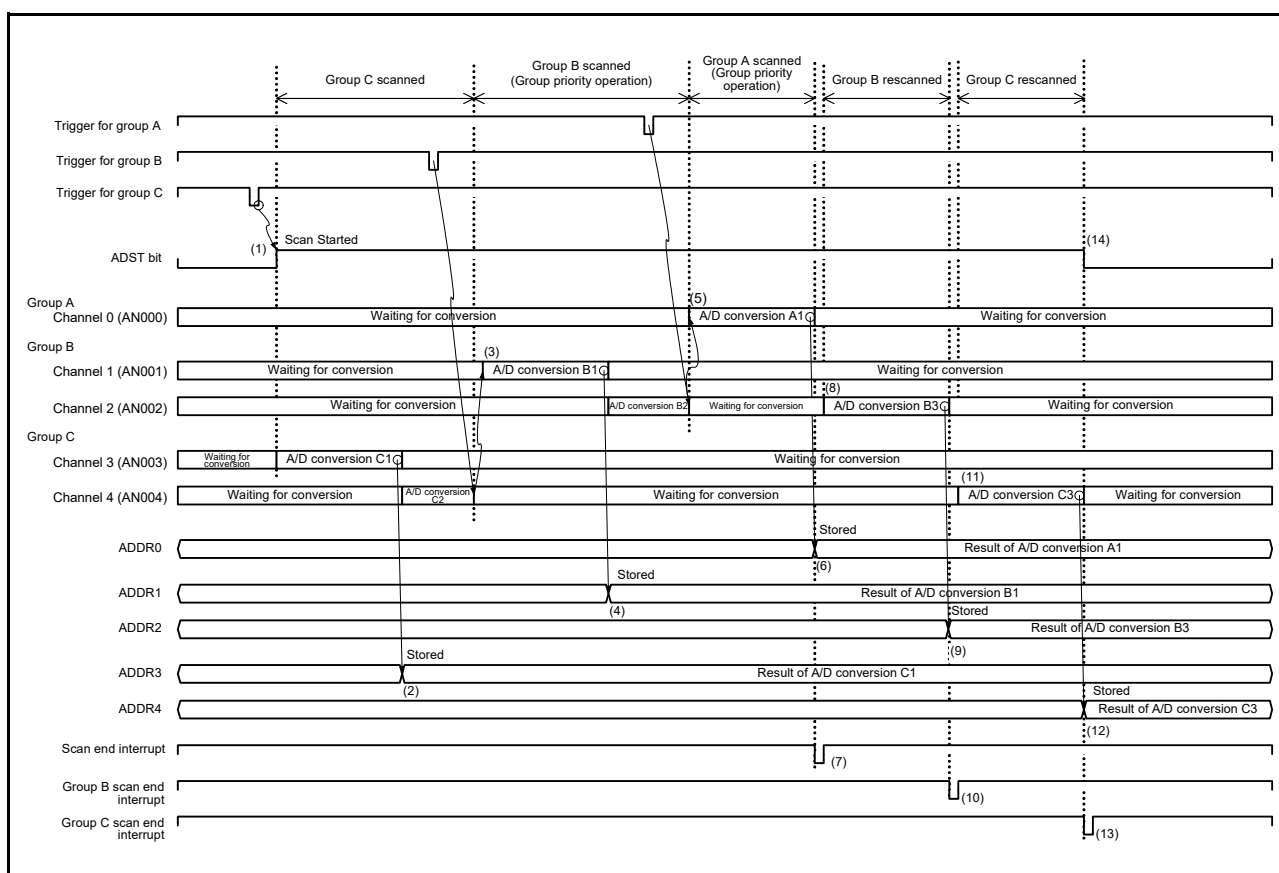


Figure 30.21 Example 1 of Group Priority Operation: Priority Group Trigger Input during Low-Priority Group Scan, With Rescan Setting (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 1)

Operation example 2: Priority group trigger input during low-priority group rescan, with rescan setting

Figure 30.22 shows an example when a group A trigger is input during rescan operation on group B.

If a trigger for the priority groups (groups A and B for group C and group A for group B) is input, scan for the priority group starts even while rescan operation on the low-priority group is in progress. After scan for the priority group is completed, scan for the low-priority group is restarted after having been discontinued.

Operations of the ADCSR.ADST bit, storing to the A/D conversion result in the A/D data register (ADDRy), and interrupt requests are the same as example 1.

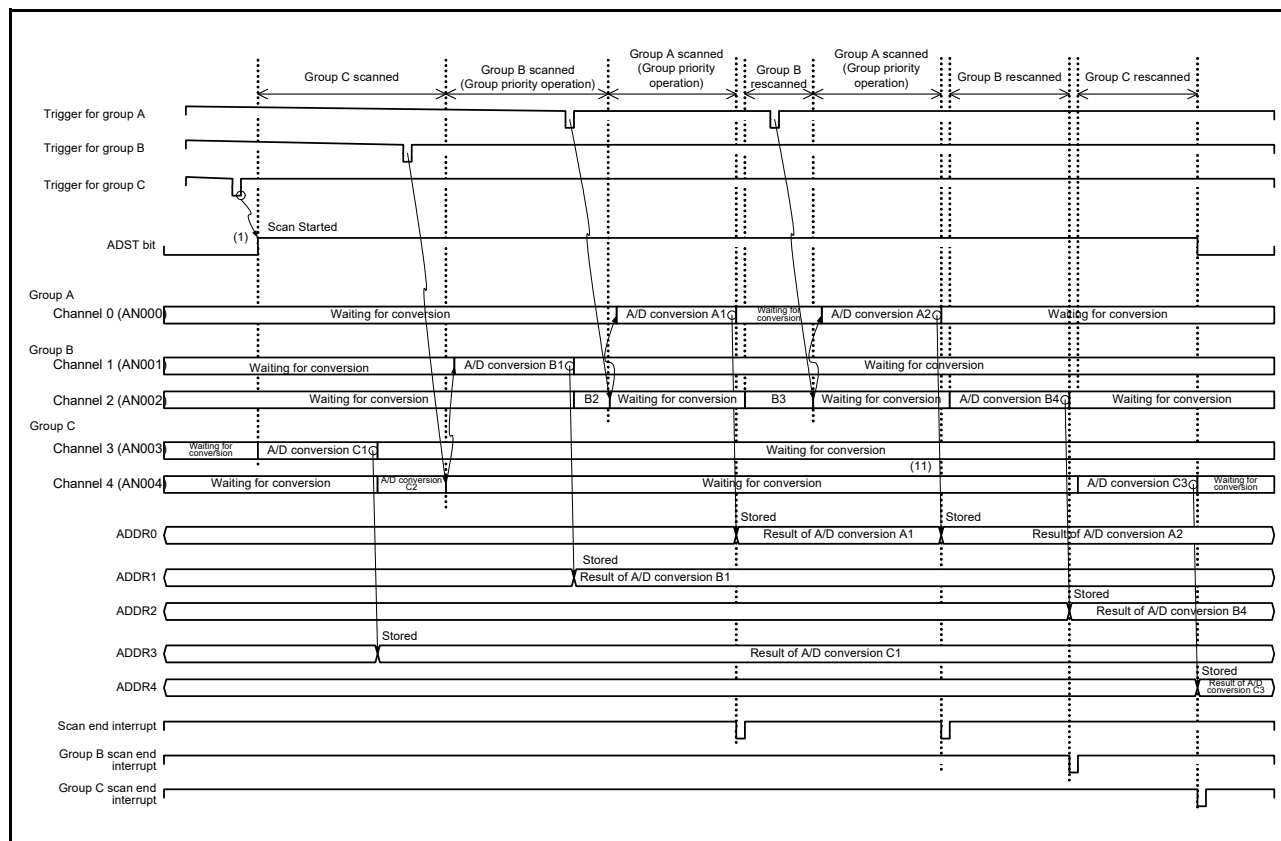


Figure 30.22 Example 2 of Group Priority Operation: Priority Group Trigger Input during Low-Priority Group Rescan, With Rescan Setting
($ADGSPCR.GBRSCN = 1$, $ADGSPCR.GBRP = 0$, $ADGSPCR.LGRRS = 1$)

Operation example 3: Low-priority group trigger input during priority group scan, with rescan setting

The following describes an example when a trigger for the low-priority group is input during scan operation on the priority group when the ADGSPCR.GBRSCN bit is 1 (scan for the group is restarted after having been discontinued due to group priority operation). If the ADGSPCR.GBRSCN bit is 0, all triggers for the low-priority group that are input during scan operation on the priority group are disabled.

- (1) When a group A trigger input sets the ADCSR.ADST bit to 1 (A/D conversion start), scan for the ANn channels of group A selected in the ADANSA0 register, starts from the channel with the smallest number n.
- (2) If a group B trigger is input during scan for group A, scan for group B can be started.
- (3) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (4) After scan for group A is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (5) After scan for group A is completed, scan for the ANn channels of group B selected in the ADANSB0 register, starts from the channel with the smallest number n while the ADCSR.ADST bit remains 1. If the ADGSPCR.LGRRS bit is set to 1 at this time, scan for group B starts from the channel on which A/D conversion was discontinued.
When a group A trigger is input during scan for group B, group A scan starts as in example 1, and group B scan starts after group A scan is completed.
- (6) If a group C trigger is input during scan for group B, scan for group C can be started.
- (7) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (8) After scan for group B is completed, a group B scan end interrupt request is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan completion enabled).
- (9) After scan for group B is completed, scan for the ANn channels of group C selected in the ADANSC0 register, starts from the channel with the smallest number n while the ADCSR.ADST bit remains 1. If the ADGSPCR.LGRRS bit is set to 1 at this time, scan for group C starts from the channel on which A/D conversion was discontinued.
When a group A or B trigger is input during scan for group C, group A or B scan starts as in example 1, and group C scan starts after group A or B scan is completed.
- (10) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy)
- (11) After scan for group C is completed, a group C scan end interrupt request is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group C scan completion enabled).
- (12) The ADCSR.ADST bit is automatically cleared to 0 when scan of all selected channels is completed, and the 12-bit A/D converter enters a wait state.

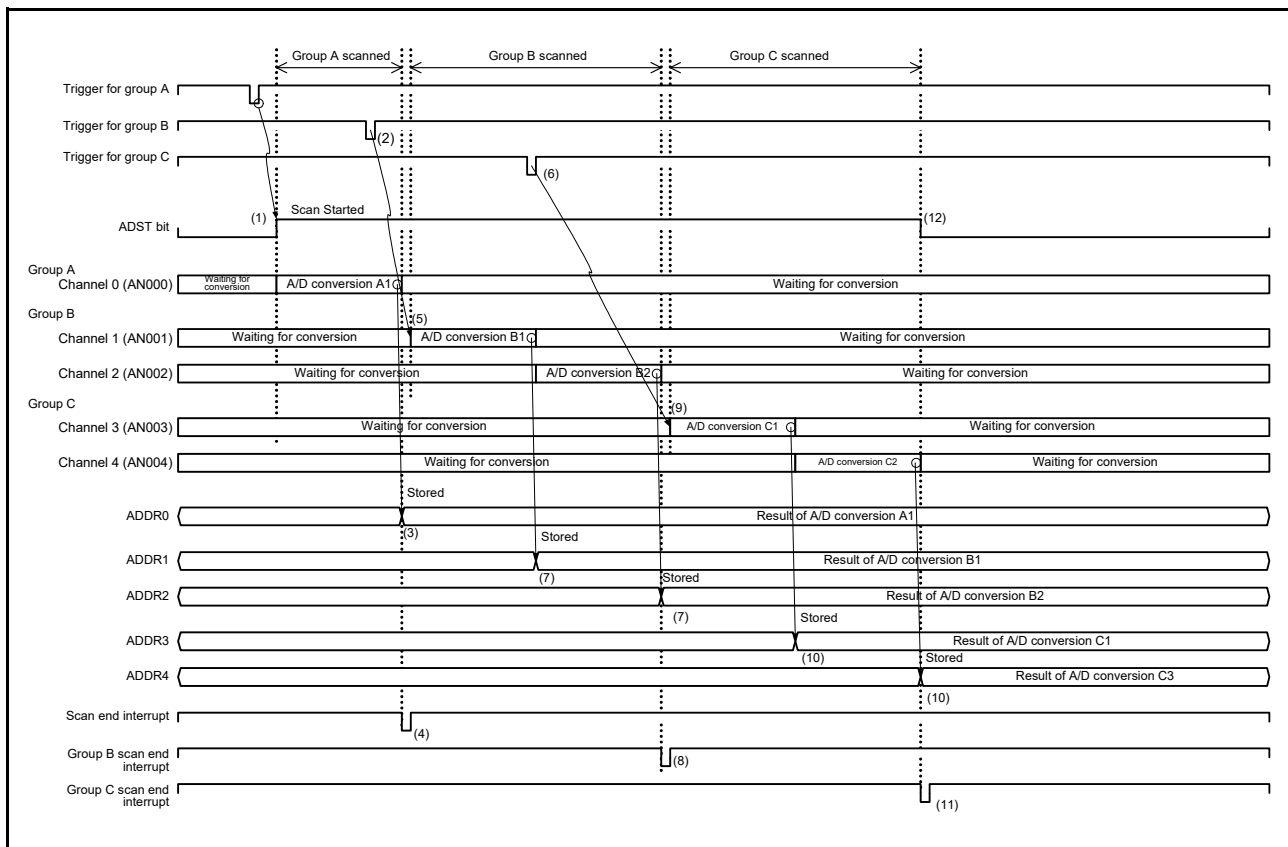


Figure 30.23 Example 3 of Group Priority Operation: Low-Priority Group Trigger Input during Priority Group Scan, With Rescan Setting
(ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 1, ADGCTRGR.GRCE = 1)

Operation example 4 shows group priority operation in group scan mode (ADGSPCR.GBRSCN = 0, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 0) when channel 0 is selected for group A, channels 1 and 2 are selected for group B, and channels 3 and 4 are selected for group C.

Operation example 4: Priority group trigger input during low-priority group scan, without rescan setting

- (1) When a group C trigger input sets the ADCSR.ADST bit to 1 (A/D conversion start), scan for the ANn channels selected in the ADANSC0 register, starts from the channel with the smallest number n.
- (2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (3) If a group B trigger is input during scan for group C, group C scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group B selected in the ADANSB0 register, starts from the channel with the smallest number n. If A/D conversion is not completed when scan was discontinued, the result is not stored in the corresponding A/D data register (ADDRy).
- (4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (5) If a group A trigger is input during scan for group B, group B scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group A selected in the ADANSB0 register, starts from the channel with the smallest number n. If A/D conversion is not completed when scan was discontinued, the result is not stored in the corresponding A/D data register (ADDRy).
- (6) After scan for group A is completed, a scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).

- (7) The ADST bit is automatically cleared to 0 when scan for group A is completed, and the 12-bit A/D converter enters a wait state. Scan for groups C and B is not started until the next trigger corresponding to the group is input.

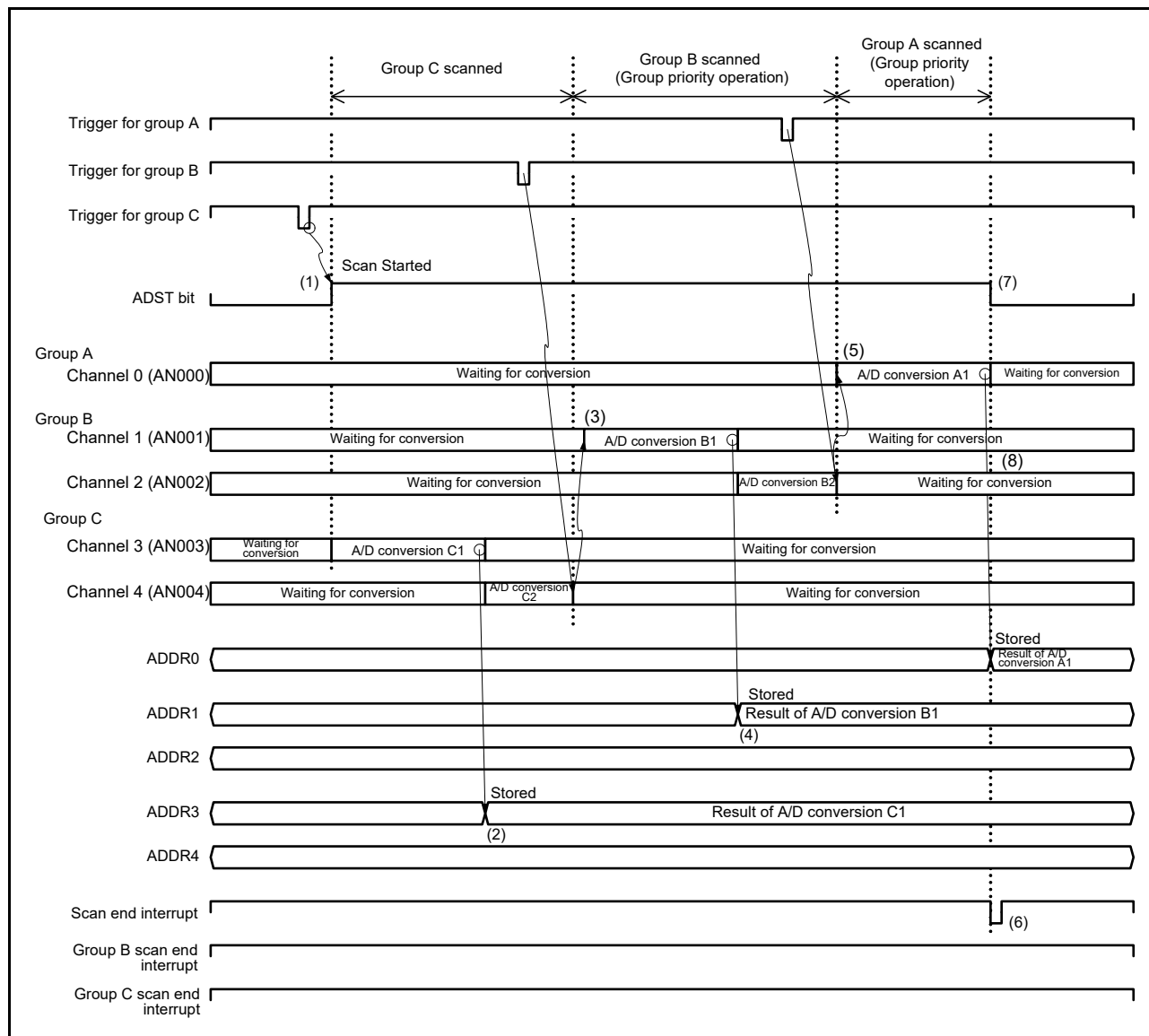


Figure 30.24 Example 4 of Group Priority Operation: Priority Group Trigger Input during Low-Priority Group Scan, Without Rescan Setting
(ADGSPCR.GBRSCN = 0, ADGSPCR.GBRP = 0, ADGSPCR.LGRRS = 1)

Operation example 5 shows group priority operation in group scan mode (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 1, ADGSPCR.LGRRS = 1) when channel 0 is selected for group A, channel 1 is selected for group B, and channels 2 and 3 are selected for group C.

When the ADGCTRGR.GRCE bit is set to 1, single scan mode is continuously operated on group B and trigger input for group C is disabled.

Operation example 5: Continuous single scan operation on group C

- (1) When setting ADGSPCR.GBRP to 1 sets the ADCSR.ADST bit to 1 (A/D conversion start), scan for the ANn channels selected in the ADANSC0 register, starts from the channel with the smallest number n.
- (2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (3) If a group B trigger is input during scan for group C, group C scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group B selected in the ADANSB0 register, starts from the channel with the smallest number n. If A/D conversion is not completed when scan was discontinued, the result is not stored in the corresponding A/D data register (ADDRy).
- (4) If a group A trigger is input during scan for group B, group B scan is discontinued while the ADCSR.ADST bit remains 1, and scan for the ANn channels of group A selected in the ADANSA0 register, starts from the channel with the smallest number n. If A/D conversion is not completed when scan was discontinued, the result is not stored in the corresponding A/D data register (ADDRy).
- (5) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (6) A scan end interrupt request is generated if the ADCSR.ADIE bit is 1 (interrupt generation upon scanning completion enabled).
- (7) If the ADGSPCR.GBRSCN bit is 1 (scan for the group is restarted after having been discontinued due to group priority operation), scan for the ANn channels of group B selected in the ADANSB0 register, restarts from the channel with the smallest number n while the ADCSR.ADST bit remains 1. If the ADGSPCR.LGRRS bit is set to 1 at this time, scan for group B starts from the channel on which A/D conversion was discontinued.
- (8) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (9) A group B scan end interrupt request is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group B scan completion enabled).
- (10) If the ADGSPCR.GBRSCN bit is 1 (scan for the group is restarted after having been discontinued due to group priority operation), scan for the ANn channels of group C selected in the ADANSC0 register, restarts from the channel with the smallest number n while the ADCSR.ADST bit remains 1. If the ADGSPCR.LGRRS bit is set to 1 at this time, scan for group C starts from the channel on which A/D conversion was discontinued.
- (11) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
- (12) A group C scan end interrupt request is generated if the ADCSR.GBADIE bit is 1 (interrupt generation upon group C scan completion enabled).
- (13) If the ADGSPCR.GBRP bit is set to 1 (single scan is continuously operated), scan for the ANn channels of group C selected in the ADANSC0 register, restarts from the channel with the smallest number n while the ADCSR.ADST bit remains 1.

To continuously operate single scan for group C, disable trigger input for group C.

Steps 13, 11, 12, and then 13 are repeated as long as the ADGSPCR.GBRP bit remains 1.

Clearing of the ADCSR.ADST bit to 0 is prohibited while the ADGSPCR.GBRP bit is set to 1.

To forcibly stop scanning when ADGSPCR.GBRP = 1, follow the procedures for clear operation by software through the ADCSR.ADST bit shown in section 30.7.2, Notes on Stopping A/D Conversion.

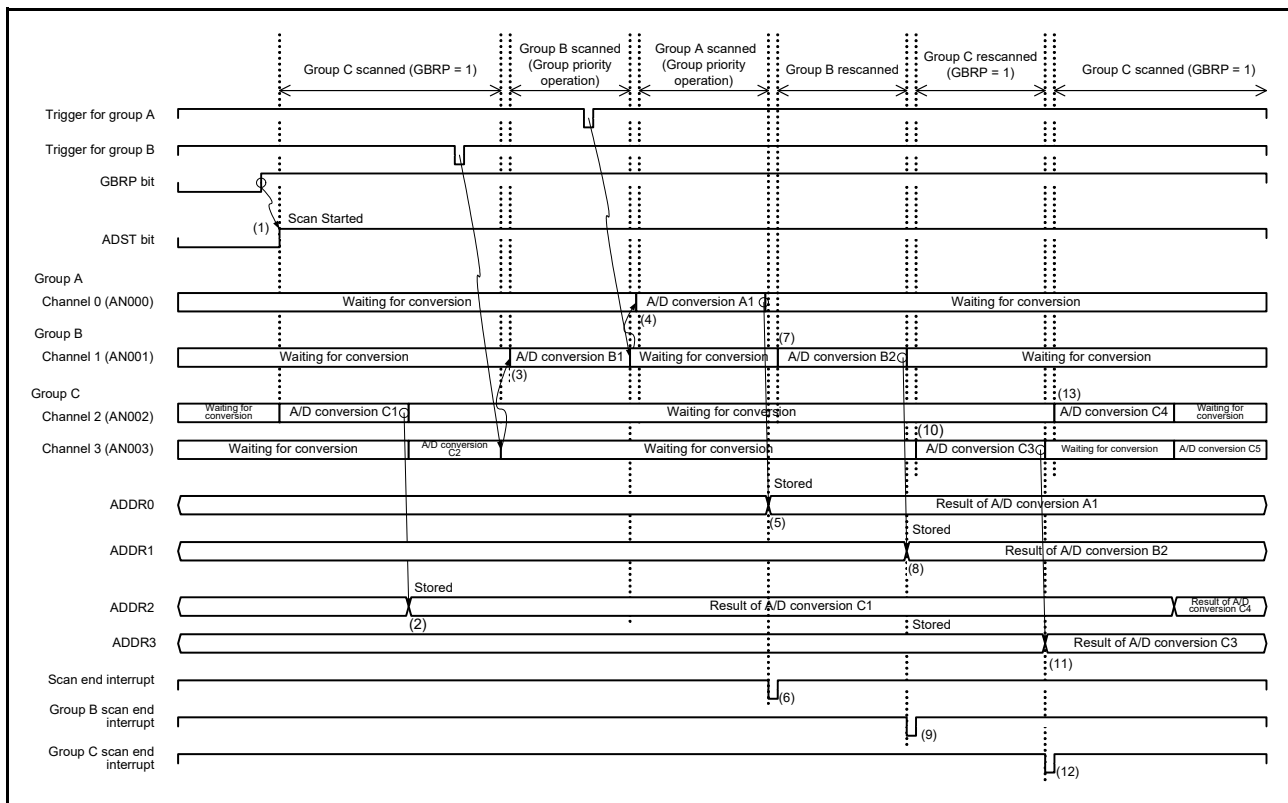


Figure 30.25 Example 5 for Group Priority Operation: Continuous Single Scan Operation on Group C (ADGSPCR.GBRP = 1, ADGSPCR.LGRRS = 1)

30.3.7 Time Interleaved Operation

Time interleaved operation is that two A/D converter units convert same one pin alternately in order to obtain the data faster. This function can be used with the channel AN001/PGAIN01 and PGAIN02 can be used in single scan mode and continuous scan mode. Regarding trigger, it is recommended to use synchronous trigger in order to ensure A/D conversion timing.

30.3.7.1 Trigger Input Timing

Trigger input timing for each A/D converter unit should be controlled not to overlap each sampling period as shown in Figure 30.26. The sampling period can be monitored at the ADSS0 and ADSS1 pins. ADSS0 and ADSS1 pins output high during $t_{SPL} + t_{DSAM}$ period. If the sampling periods overlap, A/D conversion characteristics and reliability are not guaranteed. For timing parameter (t_{TRD} , t_{SPL} , and t_{DSAM}), see Table 30.16.

Table 30.16 Setting for Trigger Input Timing in Time Interleaved Operation

Item	Symbol	Min (t_{TRDmin})
Trigger input interval between units	t_{TRD}	$t_{SPL} + t_{DSAM} + 1 \times ADCLK^{*1}$

Note 1. The parameter t_{SPL} is equal to the setting value of ADSSSTRn ($n = 1, 2$) $\times ADCLK + 0.5 \times ADCLK$.
The parameter t_{DSAM} is equal to $0.5 \times ADCLK$.

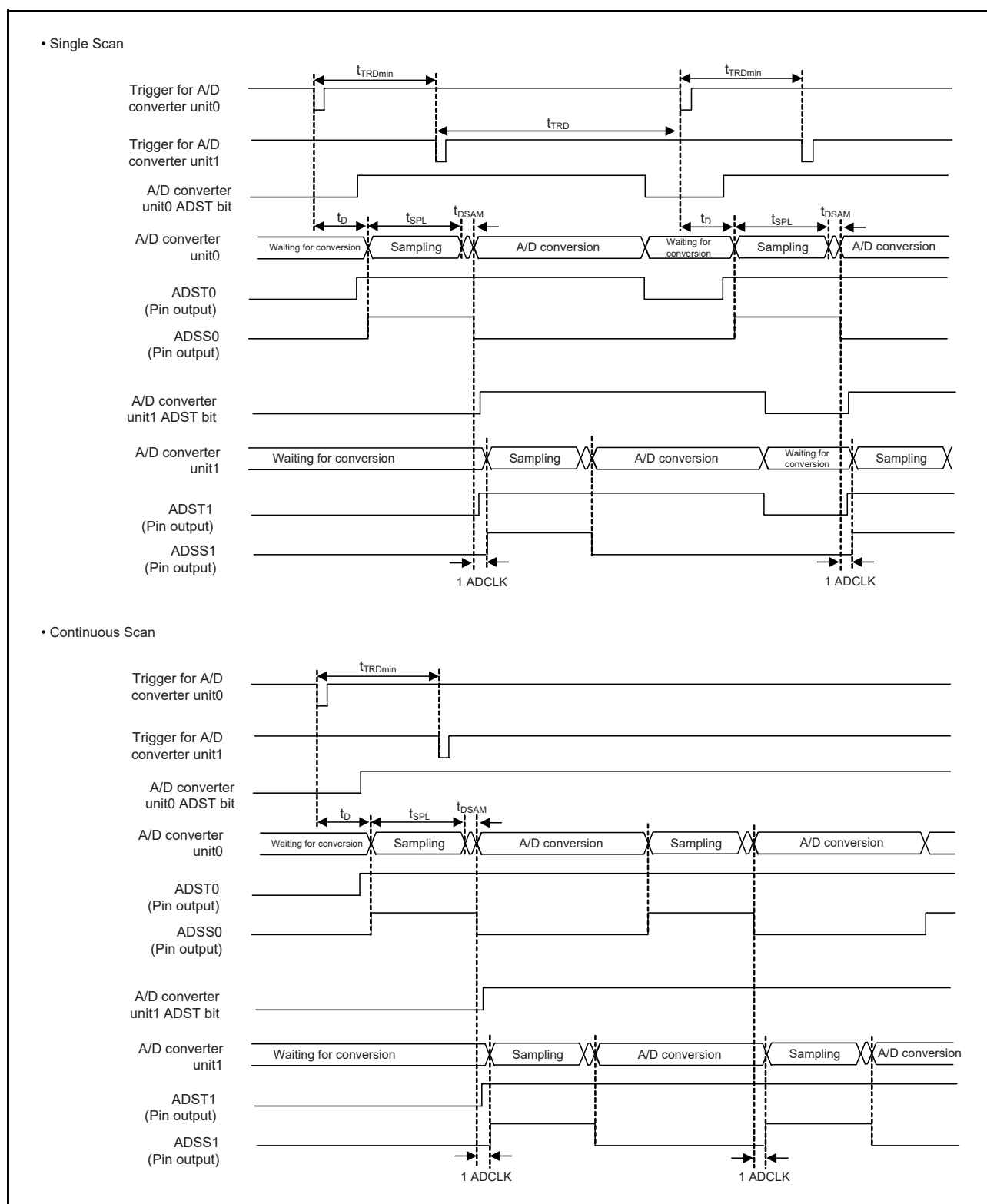


Figure 30.26 Trigger Input Timing in Time Interleaved Operation

30.3.7.2 Single Scan Mode

In this operation, Set each A/D converter to the same sampling period and conversion period.

The operation is as follows:

1. When the ADCSR.ADST bit of A/D converter unit 0 becomes 1 by the synchronous or asynchronous trigger input, A/D conversion of A/D converter unit 0 is performed for the selected channel.
2. After a certain period of time has elapsed since the start of conversion of A/D converter unit 0 (refer to section 30.3.7.1, Trigger Input Timing), when the ADCSR.ADST bit of A/D converter unit 1 becomes 1 by the synchronous or asynchronous trigger input for the channel of the same analog input pin as A/D converter unit 0, A/D conversion of A/D converter unit 1 is performed for the selected channel.
3. Before the A/D conversion of A/D converter unit 1 is completed, the A/D conversion of A/D converter unit 0 is completed, and a S12ADI interrupt request is generated.
4. The A/D conversion of A/D converter unit 0 is started by the trigger input of A/D converter unit 0.
5. When the A/D conversion of A/D converter unit 1 is completed, a S12ADI1 interrupt request is generated.
6. The above steps 1 to 5 are repeated.

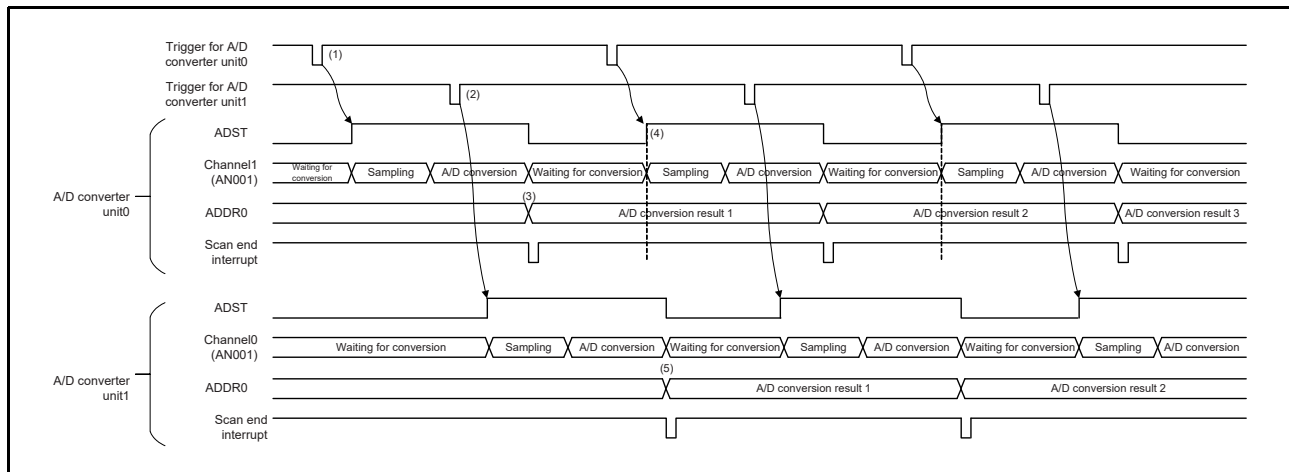


Figure 30.27 Example Operation in Single Scan Mode in Time Interleaved Operation

30.3.7.3 Single Scan Mode with Extended Double Trigger Mode

In this operation, set each A/D converter to the same sampling period and conversion period. Details of operation when extended double trigger mode is selected, refer to section 30.3.4.5, A/D Conversion in Extended Double Trigger Mode. In this operation, conversion data can be read from consecutive addresses of the S12AD.ADDBLDRA and S12AD.ADDBLDRB registers, and the S12AD1.ADDBLDRA and S12AD1.ADDBLDRB registers.

Below is an example of single scanning operation (extended double trigger mode selected, time interleaved operation) by synchronous trigger from GPTW. A/D converter unit 0 is set to start conversion with GTADTRA0N/GTADTRB0N trigger, and A/D converter unit 1 is set to start conversion with GTADTRA1N/GTADTRB1N trigger.

1. When the ADCSR.ADST bit of A/D converter unit 0 becomes 1 by the GTADTRA0N trigger input, A/D conversion starts on the single channel selected in the ADCSR.DBLANS[4:0] bits.
2. After a certain period of time has elapsed since the start of conversion of A/D converter unit 0 (refer to section 30.3.7.1, Trigger Input Timing), when the ADCSR.ADST bit of A/D converter unit 1 becomes 1 by the GTADTRA1N trigger input, A/D conversion of A/D converter unit 1 is started for the selected channel.
3. Before the A/D conversion of A/D converter unit 1 is completed, the A/D conversion of A/D converter unit 0 is completed. The A/D conversion result is stored in the relevant A/D data register y (ADDRy) and A/D data duplication register A (ADDBLDRA) of A/D converter unit 0. A S12ADI interrupt request is not generated.
4. A/D conversion of A/D converter unit 0 is started by the GTADTRB0N trigger input.
5. The A/D conversion of A/D converter unit 1 is completed. The A/D conversion result is stored in the relevant A/D data register y (ADDRy) and A/D data duplication register A (ADDBLDRA) of A/D converter unit 1. A S12ADI1 interrupt request is not generated.
6. A/D conversion of A/D converter unit 1 is started by the GTADTRB1N trigger input.
7. The A/D conversion of A/D converter unit 0 is completed. The A/D conversion result is stored in the A/D data duplication register (ADDBLDR) and A/D data duplication register B (ADDBLDRB) of A/D converter unit 0. A S12ADI interrupt request is generated.
8. A/D conversion of A/D converter unit 0 is started by the GTADTRA0N trigger input.
9. The A/D conversion of A/D converter unit 1 is completed. The A/D conversion result is stored in the A/D data duplication register (ADDBLDR) and A/D data duplication register B (ADDBLDRB) of A/D converter unit 1. A S12ADI1 interrupt request is generated.
10. The above steps 1 to 9 are repeated.

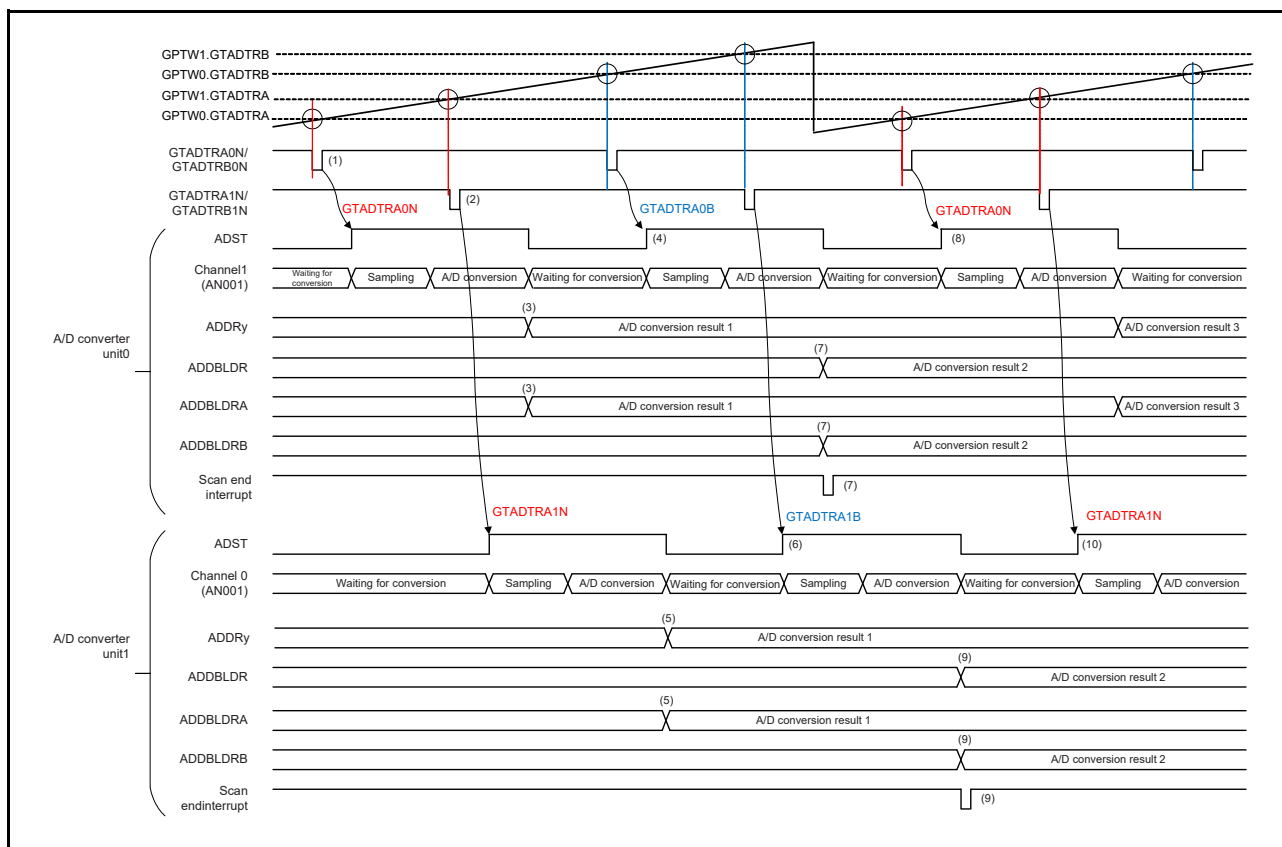


Figure 30.28 Example of Single Scan Mode Operation (Extended Double Trigger Mode Selected, Time Interleaved Operation by Synchronous Trigger from GPTW)

30.3.8 Analog Input Sampling Time and Scan Conversion Time

Figure 30.29 shows the scan conversion timing in single scan mode, in which scan conversion is activated by software or a synchronous trigger. Figure 30.30 shows the scan conversion timing in single scan mode, in which scan conversion is activated by an asynchronous trigger. The scan conversion time (t_{SCAN}) includes the start-of-scanning-delay time (t_D), disconnection detection assistance processing time (t_{DIS})*¹, self-diagnosis A/D conversion processing time (t_{DIAG})*², A/D conversion processing time (t_{CONV}), and end-of-scanning-delay time (t_{ED}).

The A/D conversion processing time (t_{CONV}) consists of sampling time (t_{SPL}) and time for conversion by successive approximation (t_{SAM}). The sampling time (t_{SPL}) is used to charge sample-and-hold circuits in the A/D converter. If there is not sufficient sampling time due to the high impedance of an analog input signal source, or if the A/D conversion clock (ADCLK) is slow, sampling time can be adjusted using the ADSSTRn register.

The time for conversion by successive approximation (t_{SAM}) is at 21 ADCLK clock cycles. Table 30.17 shows the scan conversion time.

The scan conversion time (t_{SCAN}) in single scan mode for which the number of selected channels is n can be determined as follows:

$$t_{SCAN} = t_D + (t_{DIS} \times n) + t_{DIAG} + (t_{CONV} \times n)^{*3} + t_{ED}$$

The scan conversion time for the first cycle in continuous scan mode is t_{SCAN} for single scan minus t_{ED} .

The scan conversion time for the second and subsequent cycles in continuous scan mode is fixed to $(t_{DIS} \times n) + t_{DIAG} + t_{DSD} + (t_{CONV} \times n)^{*3}$.

Note 1. When disconnection detection assistance is not selected, $t_{DIS} = 0$. The auto-discharge period of 15 ADCLK clock cycles is inserted only when the temperature sensor output or internal reference voltage is A/D-converted.

Note 2. When the self-diagnosis function is not used, $t_{DIAG} = 0$, $t_{DSD} = 0$.

Note 3. $t_{CONV} \times n$ when the sampling time (t_{SPL}) of selected channels is the same, but it is the total of the sampling time of each channel and time for conversion by successive approximation (t_{SAM}).

Table 30.17 Times for Conversion during Scanning (in Numbers of Cycles of ADCLK and PCLK)

Item			Symbol	Type/Conditions			Unit
				Synchronous Trigger (MTU, GPTW, TMR)	Asynchronous Trigger	Software Trigger	
Scan start processing time*1, *2	A/D conversion on group under group priority control.	The low-priority group is to be stopped. (The priority group is activated after low-priority group is stopped due to an A/D conversion source of the priority group.)	t _D	3 PCLK + 6 ADCLK (6 PCLK+3 ADCLK*4)	—	—	Cycle
		The low-priority group is not to be stopped. (Activation by an A/D conversion source of the priority group.)		3 PCLK + 4 ADCLK	—	—	
	A/D conversion when self-diagnosis is enabled	A/D conversion for self-diagnosis is to be started.		3 PCLK + 6 ADCLK	5 PCLK + 6 ADCLK	6 ADCLK	
	Other than above	3 PCLK + 4 ADCLK		5 PCLK + 4 ADCLK	4 ADCLK		
	Disconnection detection assistance processing time			t _{DIS}	The setting of ADDISCR.ADNDIS[3:0] × ADCLK*3		
Self-diagnosis conversion processing time*1	Sampling time	t _{DIAG}	t _{SPL}	The setting of ADSSTR0 × ADCLK + 0.5 ADCLK*5			
	Successive approximation start delay time		t _{DSAM}	0.5 ADCLK			
	Time for conversion by successive approximation		t _{SAM}	21 ADCLK			
	Wait time between self-diagnosis conversion end and analog channel sampling start.		t _{DED}	2 ADCLK			
	Wait time between last channel conversion end and self-diagnosis sampling start in continuous scan mode.	t _{DSD}	2 ADCLK				
A/D conversion processing time*1	Sampling time	t _{CONV}	t _{SPL}	The setting of ADSSTRn (n = 0 to 7, T, O) × ADCLK + 0.5 ADCLK*5			
	Successive approximation start delay time		t _{DSAM}	0.5 ADCLK			
	Time for conversion by successive approximation		t _{SAM}	21 ADCLK			
Scan end processing time*1			t _{ED}	1 PCLK + 3 ADCLK (2 PCLK + 2 ADCLK*4)			

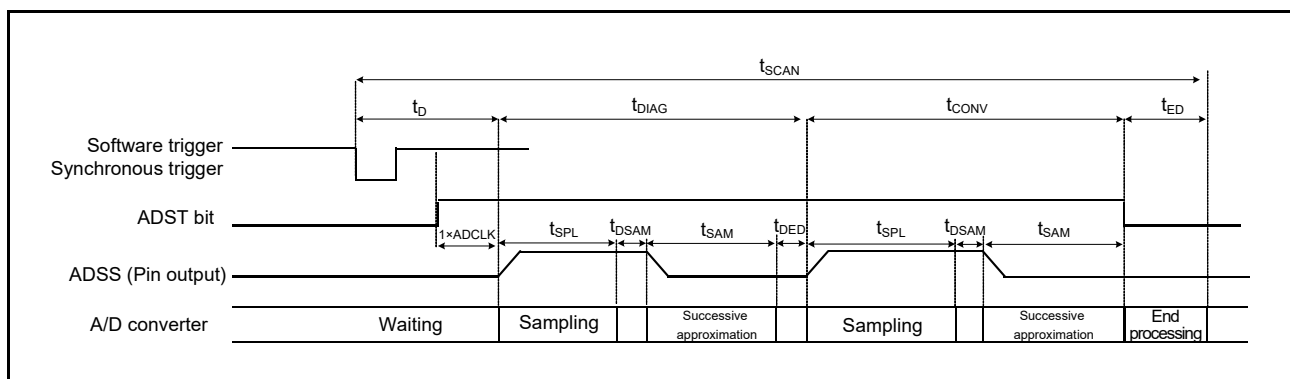
Note 1. For t_D , t_{DIAG} , t_{CONV} , and t_{ED} , refer to Figure 30.29 and Figure 30.30.

Note 2. This is the maximum time required from software writing or trigger input to A/D conversion start.

Note 3. The value is fixed to 0Fh (15 ADCLK) when the temperature sensor output and internal reference voltage is A/D-converted.

Note 4. When ADCLK is faster than PCLK (PCLK to ADCLK frequency ratio = 1:2).

Note 5. The required sampling time (ns) is specified according to the voltage conditions. See section 38.6, A/D Conversion Characteristics.

**Figure 30.29 Scan Conversion Timing (Activated by Software or Synchronous Trigger)**

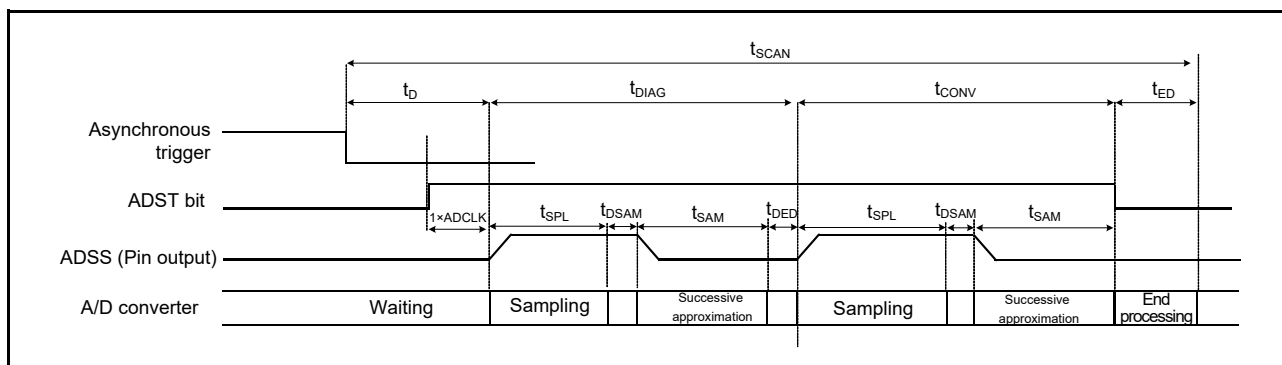


Figure 30.30 Scan Conversion Timing (Activated by Asynchronous Trigger)

30.3.8.1 Timing of Suspension and Starting of Scanning in Operation under Group Priority Control

The timings for suspension and starting of scanning in operation under group priority control that must be considered are listed below.

1. The timing for suspending a scan of a group with a lower-priority and the timing for starting a scan of a group with a higher-priority.
2. The time at which scanning by the group with a lower-priority is resumed on completion of scanning by the higher-priority group when the trigger for scanning by the lower-priority group is accepted during scanning by the higher-priority group.
3. The timing for performing sequential single scans by a lower-priority group.

Figure 30.31 shows the timing diagram of each of the above cases.

The times at which scanning by group A is completed and scanning by group C resumes or scanning by group B is completed and scanning by group C resumes are the same as those for group A and group B in Figure 30.31.

The timing for consecutive single scans is the same for group B and group C.

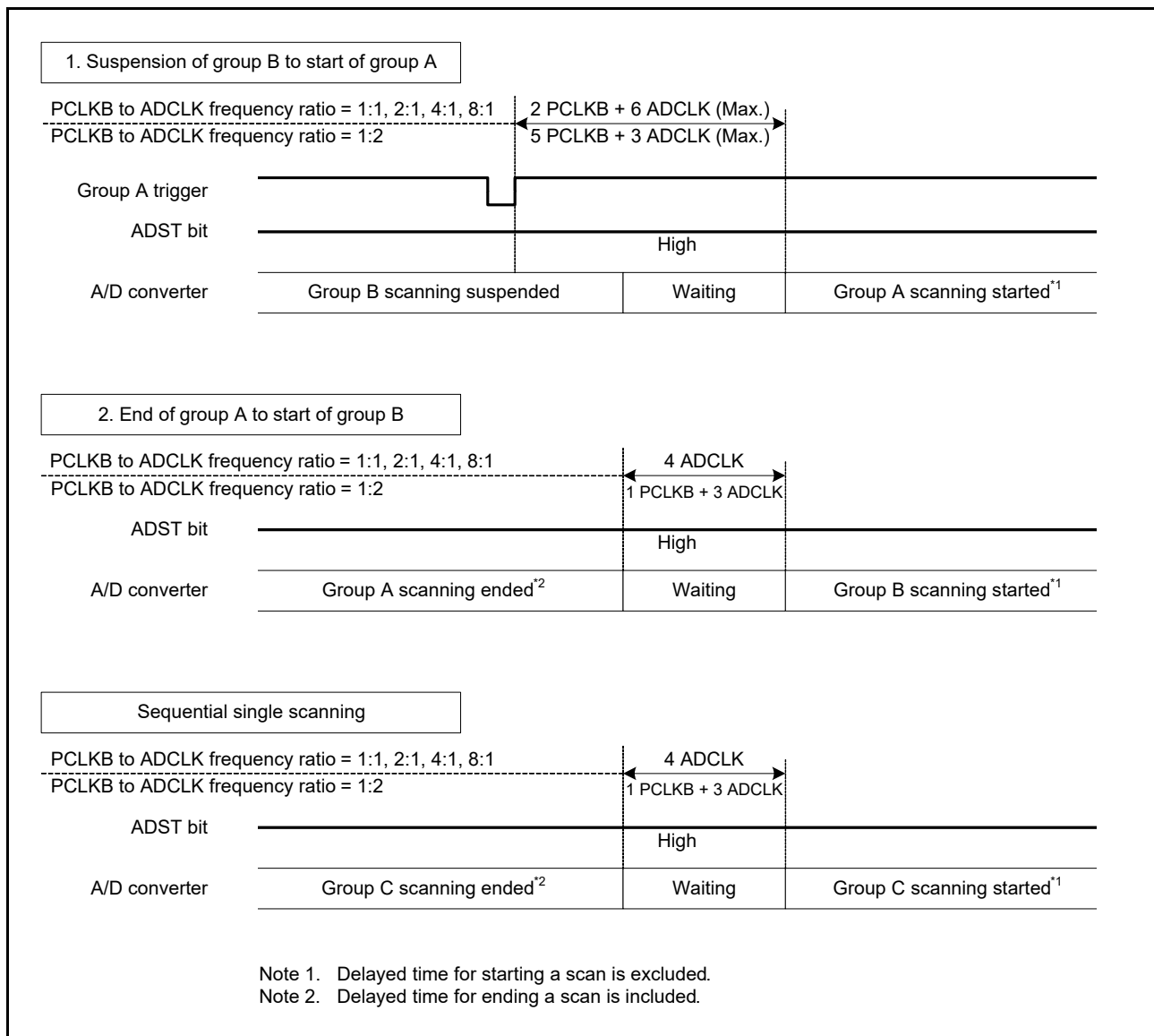


Figure 30.31 Timing of Stop/Start of Scanning in Group Priority Mode

30.3.9 Usage Example of A/D Data Register Automatic Clearing Function

Setting the ADCER.ACE bit to 1 automatically clears the A/D data registers (ADDRy, ADRD, ADTSDR, ADOCDR, ADDBLDR, ADDBLDRA, ADDBLDRB) to 0000h when the A/D data registers (ADDRy, ADRD, ADTSDR, ADOCDR, ADDBLDR, ADDBLDRA, ADDBLDRB) are read by the CPU or DTC.

This function enables detection of update failures of the A/D data registers (ADDRy, ADRD, ADTSDR, ADOCDR, ADDBLDR, ADDBLDRA, ADDBLDRB). The following describes the examples in which the function to automatically clear the ADDRy register is enabled and disabled.

In a case where the ADCER.ACE bit is 0 (automatic clearing disabled), if the A/D conversion result (0222h) is not written to the ADDRy register for some reason, the old data (0111h) will be the ADDRy value. Furthermore, if this ADDRy value is read into a general register using an A/D scan end interrupt, the old data (0111h) can be saved in the general register. When checking whether there is an update failure, it is necessary to frequently save the old data in the RAM or a general register.

In a case where the ADCER.ACE bit is 1 (automatic clearing enabled), when ADDRy = 0111h is read by the CPU or DTC, ADDRy is automatically cleared to 0000h. After that, if the A/D conversion result 0222h cannot be transferred to ADDRy for some reason, the cleared data (0000h) remains as the ADDRy value. If this ADDRy value is read into a general register using an A/D scan end interrupt at this point, 0000h will be saved in the general register.

Occurrence of an ADDRy update failure can be determined by simply checking that the read data value is 0000h.

30.3.10 A/D-Converted Value Addition/Average Mode

In A/D-converted value addition mode, the same channel is A/D-converted 2, 3, 4, or 16 consecutive times and the sum of the converted values is stored in the data register. In A/D-converted value average mode, the same channel is A/D-converted two or four consecutive times and the mean of the converted values is stored in the data register. The use of the average of these results can improve the accuracy of A/D conversion, depending on the types of noise components that are present. This function, however, cannot always guarantee an improvement in A/D conversion accuracy.

The A/D-converted value addition/average mode can be used when A/D conversion of the channel select analog input, temperature sensor output or internal reference voltage (for S12AD1 only) is selected.

30.3.11 Disconnection Detection Assist Function

This converter incorporates the function to fix the charge for sampling capacitance to the specified state (reference voltage selected by the A/D high-potential/low-potential reference voltage control register) before start of A/D conversion. This function enables disconnection detection in wiring of analog inputs.

Figure 30.32 illustrates the A/D conversion operation when the disconnection detection assist function is used. Figure 30.33 shows an example of disconnection detection when precharge is selected. Figure 30.34 shows an example of disconnection detection when discharge is selected.

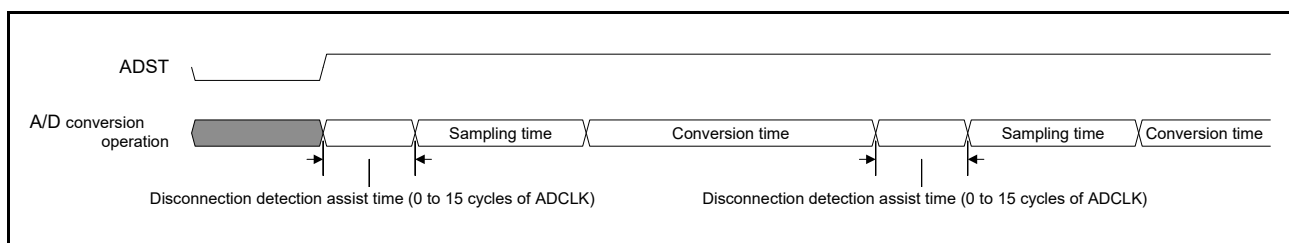


Figure 30.32 Operation of A/D Conversion When the Disconnection Detection Assist Function is Used

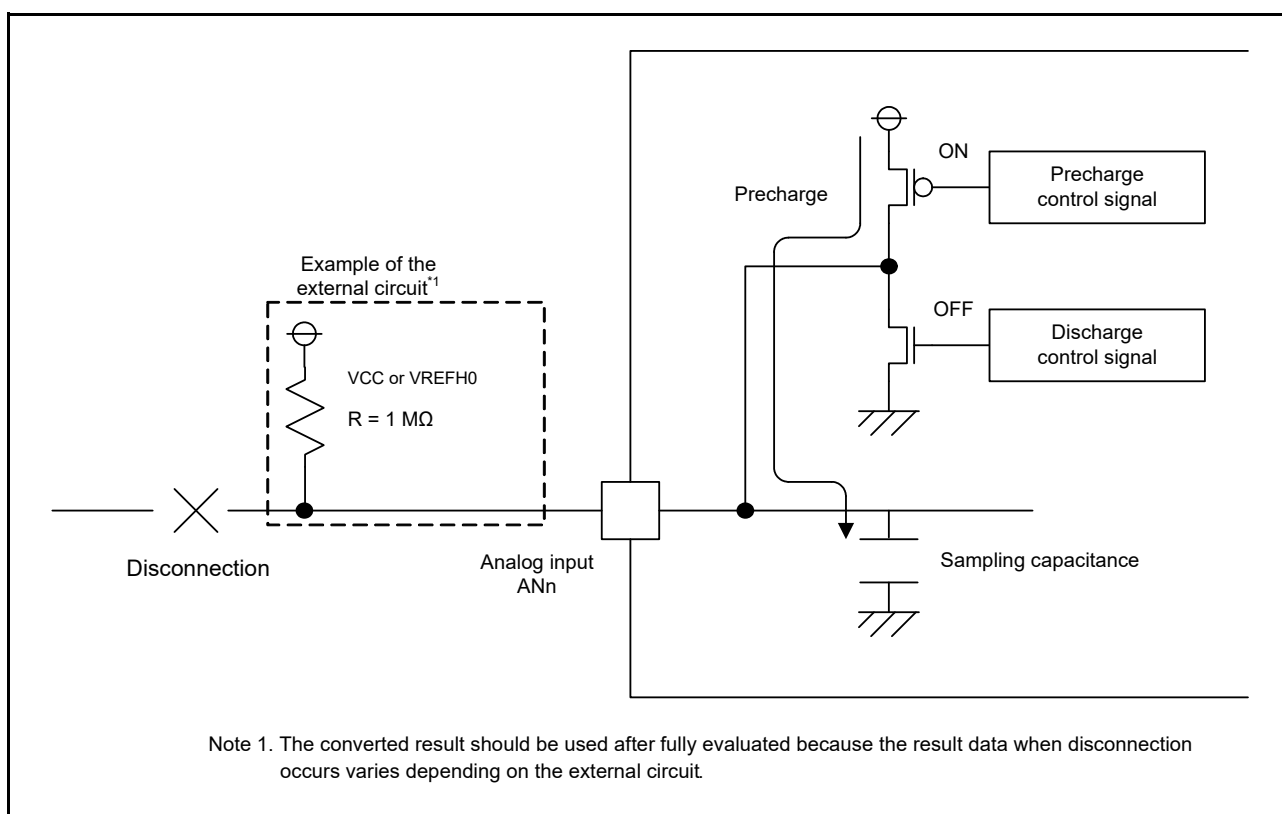


Figure 30.33 Example of Disconnection Detection When Precharge is Selected

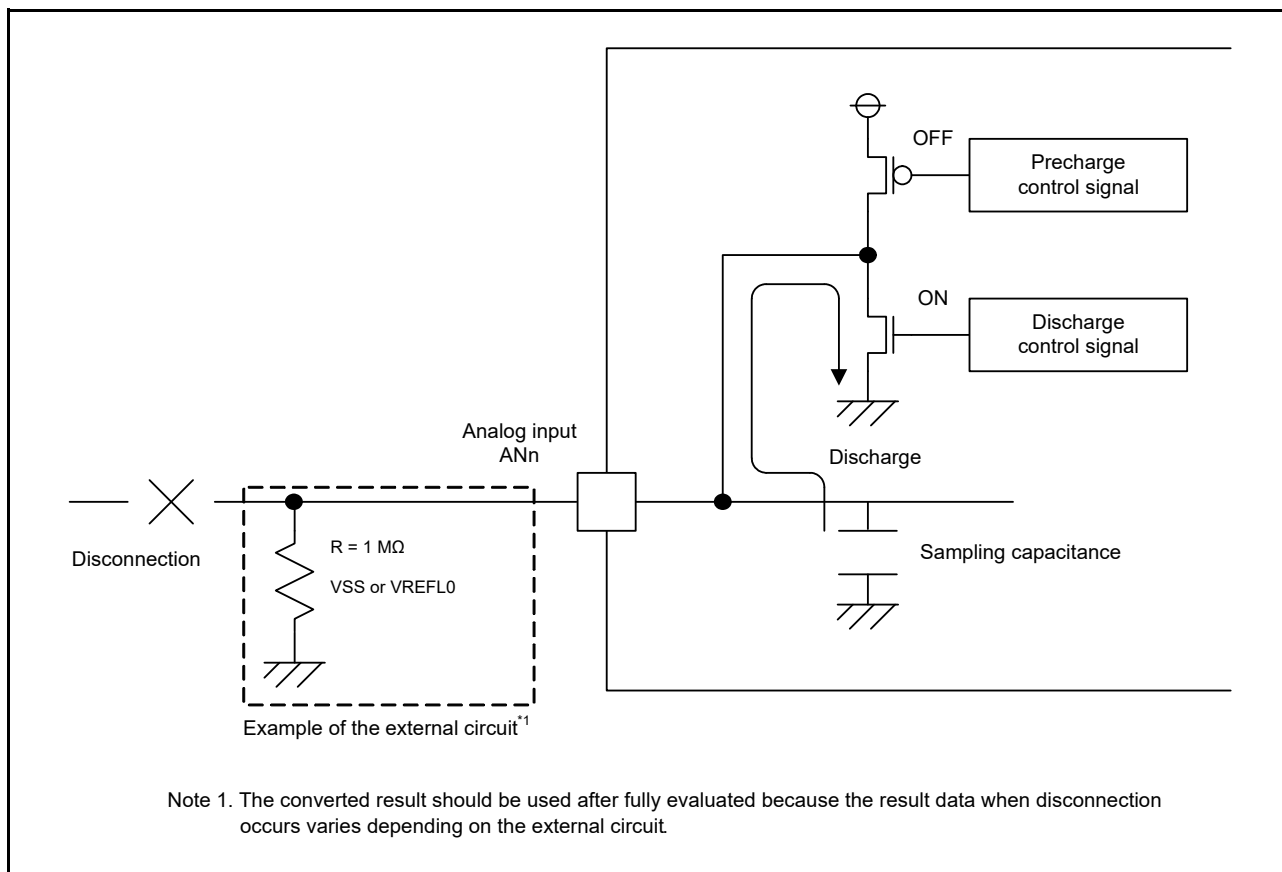


Figure 30.34 Example of Disconnection Detection When Discharge is Selected

30.3.12 Starting A/D Conversion with Asynchronous Trigger

The A/D conversion can be started by the input of an asynchronous trigger. To start up the A/D converter by an asynchronous trigger, the A/D conversion start trigger select bits (ADSTRGR.TRSA[5:0]) should be set to 000000b, and a high-level signal should be input to the asynchronous trigger (ADTRG0# pin). Then, the ADCSR.TRGE and ADCSR.EXTRG bits should be set to 1. Figure 30.35 shows a timing of the asynchronous trigger input.

For the time from when the ADST bit is set to 1 until conversion starts, refer to section 30.7.3, A/D Conversion Restarting Timing and Termination Timing. An asynchronous trigger cannot be selected in group scan mode.

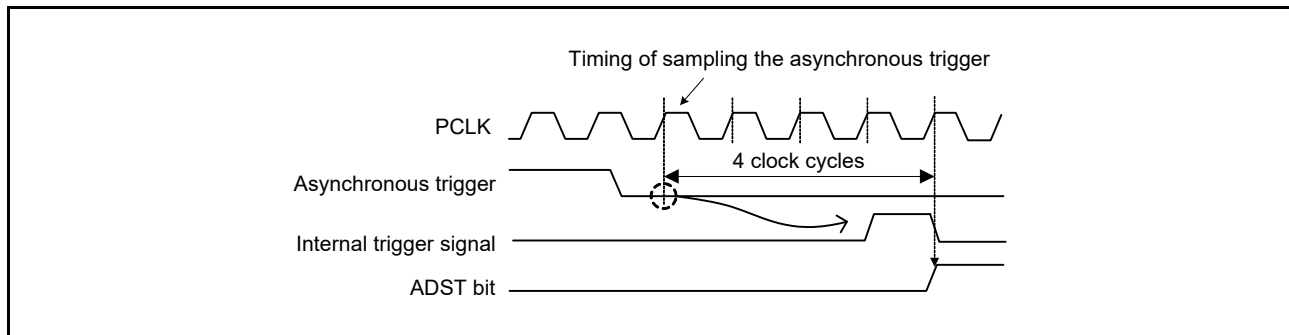


Figure 30.35 Timing of Sampling Asynchronous Trigger

30.3.13 Starting A/D Conversion with Synchronous Trigger from Peripheral Module

The A/D conversion can be started by a synchronous trigger. To start the A/D conversion by a synchronous trigger, the ADCSR.TRGE bit should be set to 1, the ADCSR.EXTRG bit should be cleared to 0, and the relevant sources should be selected by the ADSTRGR.TRSA[5:0], ADSTRGR.TRSB[5:0], and ADSTRGR.TRSC[5:0] bits.

30.3.14 Programmable Gain Amplifier

A programmable gain amplifier is included in AN000 to AN002. For details, refer to section 31, Programmable Gain Amplifier (PGA).

30.4 Interrupt Sources and DTC Transfer Requests

30.4.1 Interrupt Requests

The 12-bit A/D converter can send scan end interrupt requests S12ADI/S12ADI1, GBADI/GBADI1, and GCADI/GCADI1 to the CPU.

Setting the ADCSR.ADIE bit to 1 and 0 enables and disables an S12ADI/S12ADI1 interrupt, respectively; similarly, setting the ADCSR.GBADIE bit to 1 and 0 enables and disables a GBADI/GBADI1 interrupt, respectively; similarly, setting the ADGCTRGR.GCADIE bit to 1 and 0 enables and disables a GCADI/GCADI1 interrupt, respectively.

In addition, the DTC can be activated when an S12ADI/S12ADI1, a GBADI/GBADI1, or GCADI/GCADI1 interrupt is generated. Using an S12ADI/S12ADI1, a GBADI/GBADI1, or GCADI/GCADI1 interrupt to allow the DTC to read the converted data enables sequence conversion without burden on software.

For details on DTC settings, refer to section 16, Data Transfer Controller (DTCb).

Interrupts are output as shown in Table 30.18, depending on the settings of scan mode and double trigger mode.

Table 30.18 Relationship between Mode Setting and ADI Interrupt Output

Scan mode	Double trigger mode (ADCSR.DBLE)	Trigger	S12ADI interrupt (ADCSR.ADIE = 1)	GBADI interrupt (ADCSR.GBADIE = 1)	GCADI interrupt (ADGCTRGR.GCADIE = 1)
Single scan	DBLE = 0	Software trigger	Output on completion of each scan		
		Synchronous trigger			
		Asynchronous trigger			
	DBLE = 1	Software trigger	Setting prohibited		
		Synchronous trigger	Output on completion of each evenorder scan		
		Asynchronous trigger	Setting prohibited		
Continuous scan	Setting prohibited	Software trigger	Output on completion of each scan		
		Synchronous trigger			
		Asynchronous trigger			
Group scan	DBLE = 0	Software trigger	Setting prohibited	Setting prohibited	Setting prohibited
		Synchronous trigger	Output on completion of each scan (performed for group A only)	Output on completion of each scan (performed for group B only)	Output on completion of each scan (performed for group C only)
		Asynchronous trigger	Setting prohibited	Setting prohibited	Setting prohibited
	DBLE = 1	Software trigger	Setting prohibited	Setting prohibited	Setting prohibited
		Synchronous trigger	Output on completion of each evenorder scan for group A	Output on completion of each scan (performed for group B only)	Output on completion of each scan (performed for group C only)
		Asynchronous trigger	Setting prohibited	Setting prohibited	Setting prohibited

30.5 Selecting Reference Voltage

For the A/D converter, the high-potential reference voltage can be selected from VREFH0 and VCC, and the low potential reference voltage can be selected from VREFL0 and VSS, respectively. Set these before starting A/D conversion. For details of this setting, refer to section 30.2.16, A/D High-Potential/Low-Potential Reference Voltage Control Register (ADHVREFCNT).

30.6 Allowable Impedance of Signal Source

Figure 30.36 shows an equivalent circuit of an analog input pin and an external sensor.

To perform A/D conversion accurately, the internal capacitor (C_s) must be fully charged within the sampling time. If the impedance (R_0) of the signal source is high and it takes time to charge C_s , extend the sampling time with the ADSSTRn register. Conversely, if R_0 is small, the sampling time can be shortened. Refer to the electrical characteristics for the permissible signal source impedance under various operating conditions.

When converting only a single pin input in single scan mode, the influence of R_0 can be ignored because the input load becomes practically only the internal input resistor (R_s) by connecting an external high-capacity capacitor (C). However, because a low-pass filter is formed by R_0 and C , it may not be possible to follow the analog signal that changes at high speed. Insert a low-impedance buffer when converting high speed analog signals or when converting multiple pins in scan mode.

For allowable impedance of the signal source and sampling time, refer to section 38, Electrical Characteristics.

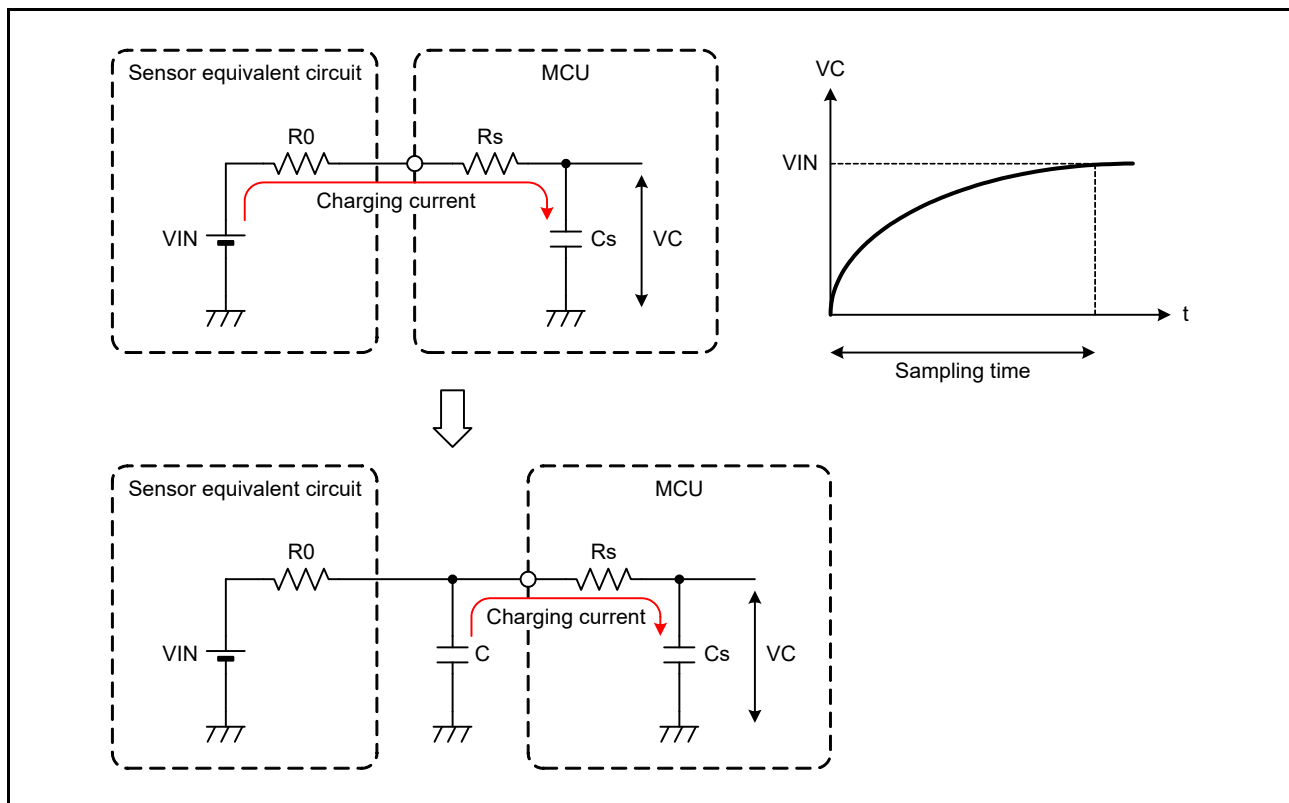


Figure 30.36 Equivalent Circuit of Analog Input Pin and External Sensor

30.7 Usage Notes

30.7.1 Notes on Reading Data Registers

The A/D data registers, A/D data duplication registers, A/D data duplication register A, A/D data duplication register B, A/D temperature sensor data register (for S12AD1 only), A/D internal reference voltage data register (for S12AD1 only), and A/D self-diagnosis data register should be read in 16-bit units. If a register is read twice in 8-bit units, that is, the higher-order byte and lower-order byte are separately read, the A/D-converted value having been read first may disagree with the A/D-converted value having been read for the second time. To prevent this, the data registers should never be read in 8-bit units.

30.7.2 Notes on Stopping A/D Conversion

30.7.2.1 Procedure of Stopping A/D Conversion

To stop A/D conversion when an asynchronous trigger or a synchronous trigger has been selected as the condition for starting A/D conversion, follow the procedure in Figure 30.37.

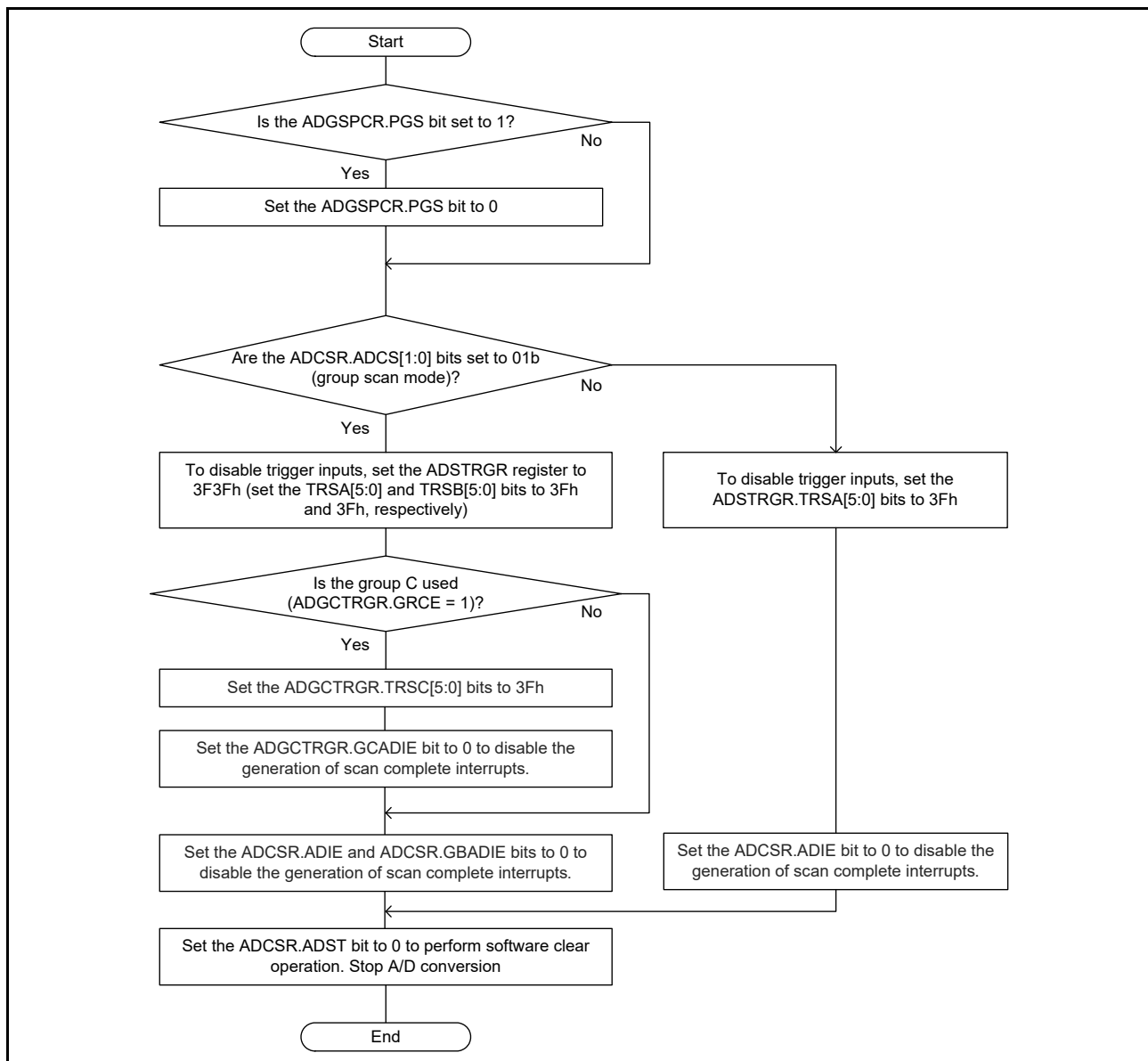


Figure 30.37 Procedure for Clear Operation by Software through the ADCSR.ADST Bit

30.7.2.2 Notes on Modes and Status Flags

The following values should be initialized or reset as described below: the voltage status used in self-diagnosis, the odd-even determination for the double trigger mode.

- Re-set the voltage for use in self-diagnosis by setting the ADCER.DIAGLD bit to 1 and setting the ADCER.DIAGVAL[1:0] bits.
- Make double trigger mode effective starting from the first scan by changing the ADCSR.DBLE bit from 0 to 1.

30.7.3 A/D Conversion Restarting Timing and Termination Timing

It takes a maximum of six ADCLK cycles for the idle analog unit of the 12-bit A/D converter to be restarted by setting the ADCSR.ADST bit to 1. It takes a maximum of three ADCLK cycles (1 PCLK + 2 ADCLK when ADCLK is faster than PCLK (PCLK to ADCLK frequency ratio = 1:2)) for the operating analog unit of the 12-bit A/D converter to be terminated by setting the ADCSR.ADST bit to 0.

30.7.4 Notes on Scan End Interrupt Handling

When scanning the same analog input twice using any trigger, the first A/D-converted data is overwritten with the second A/D-converted data in the case that the CPU does not complete reading the A/D-converted data by the time the A/D conversion of the first analog input for the second scan ends after the first scan end interrupt is generated.

30.7.5 Module Stop Function Setting

Operation of the 12-bit A/D converter can be disabled or enabled by setting module stop control register. The initial setting is for operation of the 12-bit A/D converter to be halted. Register access is enabled by releasing the module stop state.

After the module stop state is released, wait for 1 μ s to start A/D conversion. For details, refer to **section 11, Low Power Consumption**.

30.7.6 Notes on Entering Low Power Consumption States

Before entering the module stop state or software standby mode, make sure to stop A/D conversion. Here, set the ADCSR.ADST bit to 0, and secure certain period of time until the analog unit of the 12-bit A/D converter is stopped. Follow the procedure given below to secure this time.

Set the ADCSR.ADST bit to 0 by following the procedure in **Figure 30.37 Procedure for Clear Operation by Software through the ADCSR.ADST Bit**. Then wait for three clock cycles of ADCLK before entering the module stop mode or software standby mode.

30.7.7 Notes on Canceling Software Standby Mode

After software standby mode is canceled, wait until the crystal oscillation stabilization time or the PLL circuit stabilization time elapses, and then wait for 1 μ s before starting A/D conversion. For details, refer to **section 11, Low Power Consumption**.

30.7.8 Pin Setting when the 12-bit A/D Converter is Used

If the pins near the analog input pins change outputs during A/D conversion, it may affect A/D conversion accuracy.

30.7.9 Error in Absolute Accuracy When Disconnection Detection Assistance is in Use

Using disconnection detection assistance leads to an error in absolute accuracy of the A/D converter. This is because an error voltage is input to the analog input pins due to the resistive voltage division between the pull-up or pull-down resistor (R_p) and the resistance of the signal source (R_s). This error in absolute accuracy is calculated from the following formula. Only use disconnection detection assistance after thorough evaluation.

Maximum error in absolute accuracy (LSB) = $4095 \times R_s / R_p$

30.7.10 Voltage Range of Analog Power Supply Pins

If this MCU is used with the voltages outside the following ranges, the reliability of the MCU may be affected.

- Analog input voltage range

Voltage applied to analog input pins AN000 to AN007, AN102 to AN107: $V_{SS} \leq V_{ANn} \leq V_{CC}$

Conversion will not succeed if the voltage applied to analog input pins ANn is greater than V_{REFH0} (see Figure 30.38).

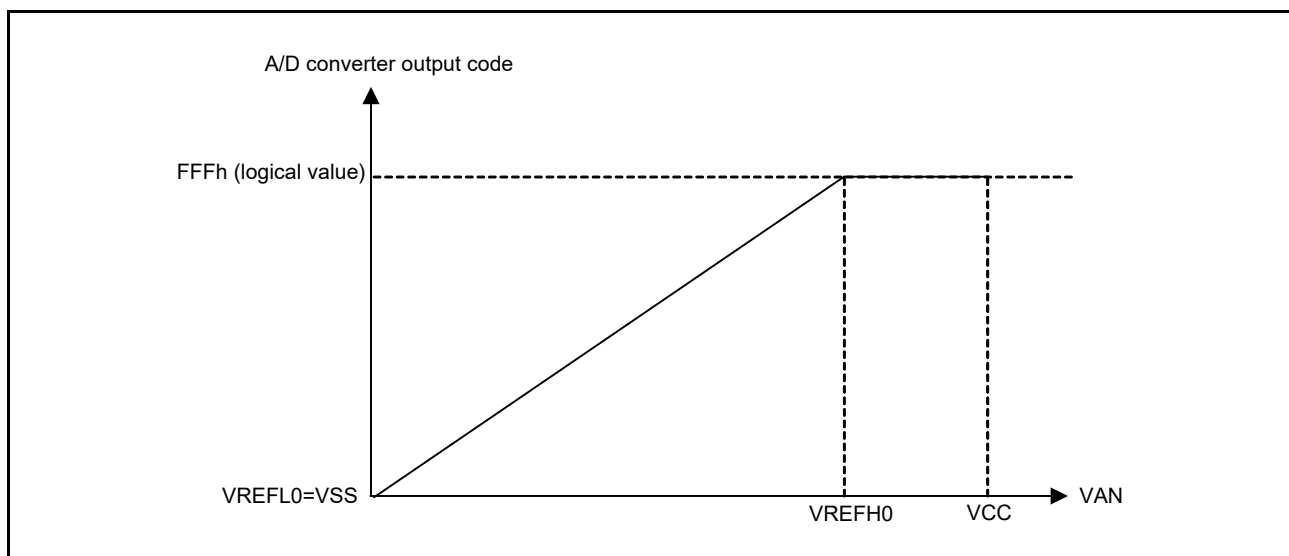


Figure 30.38 Relationship Between Voltage Applied to Analog Input Pins and Output Code

- Relationship between power supply pin pairs (VREFH0 – VREFL0, VCC – VSS)

A 0.1- μ F capacitor should be connected between each pair of power supply pins to create a closed loop with the shortest route possible as shown in Figure 30.39, and connection should be made so that the following condition is satisfied at the supply side: VREFL0 = VSS.

Do not connect the capacitors if VCC and VSS are to be used as the reference voltage for the A/D converter, and if VREFH0 and VREFL0 are to be used for other purposes.

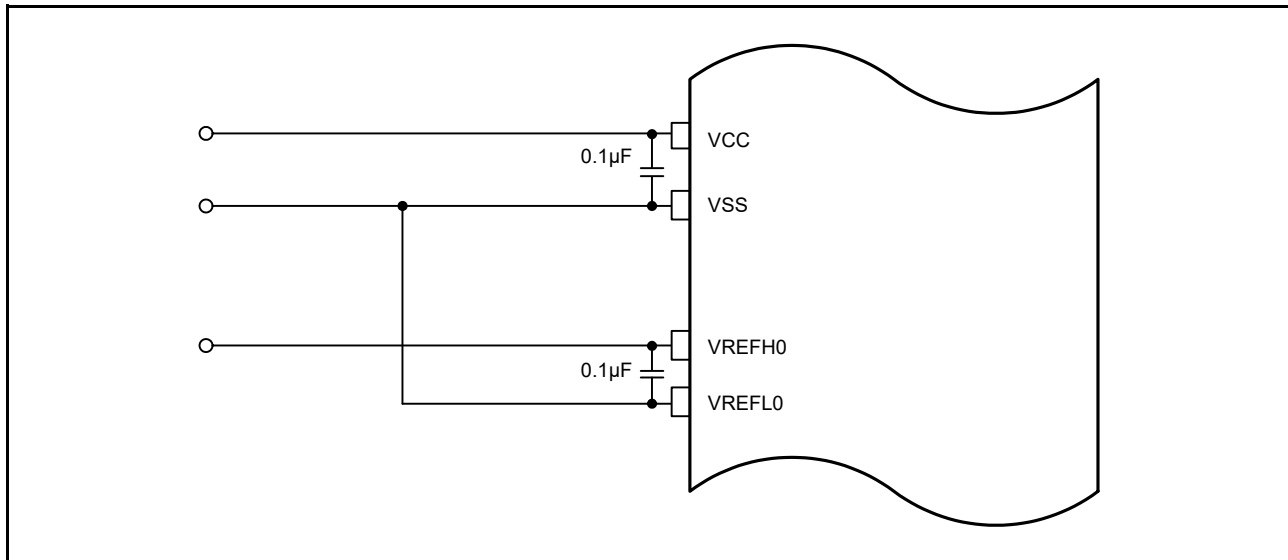


Figure 30.39 Power Supply Pin Connection Example

30.7.11 Notes on Board Design

The board should be designed so that digital circuits and analog circuits are separated from each other as far as possible. In addition, digital circuit signal lines and analog circuit signal lines should not intersect or placed near each other. If these rules are not followed, noise will be produced on analog signals and A/D conversion accuracy will be affected. The analog input pins (AN000 to AN007, AN102 to AN107), analog reference power supply (VREFH0) and analog reference ground (VREFL0) should be separated from digital circuits.

30.7.12 Notes on Noise Prevention

To prevent the analog input pins (AN000 to AN007, AN102 to AN107) from being destroyed by abnormal voltage such as excessive surge, a capacitor should be inserted between VCC and VSS, VREFH0 and VREFL0, and a protection circuit should be connected to protect the above analog input pins as shown Figure 30.40.

Do not connect the capacitors if VCC and VSS are to be used as the reference voltage for the A/D converter, and if VREFH0 and VREFL0 are to be used for other purposes.

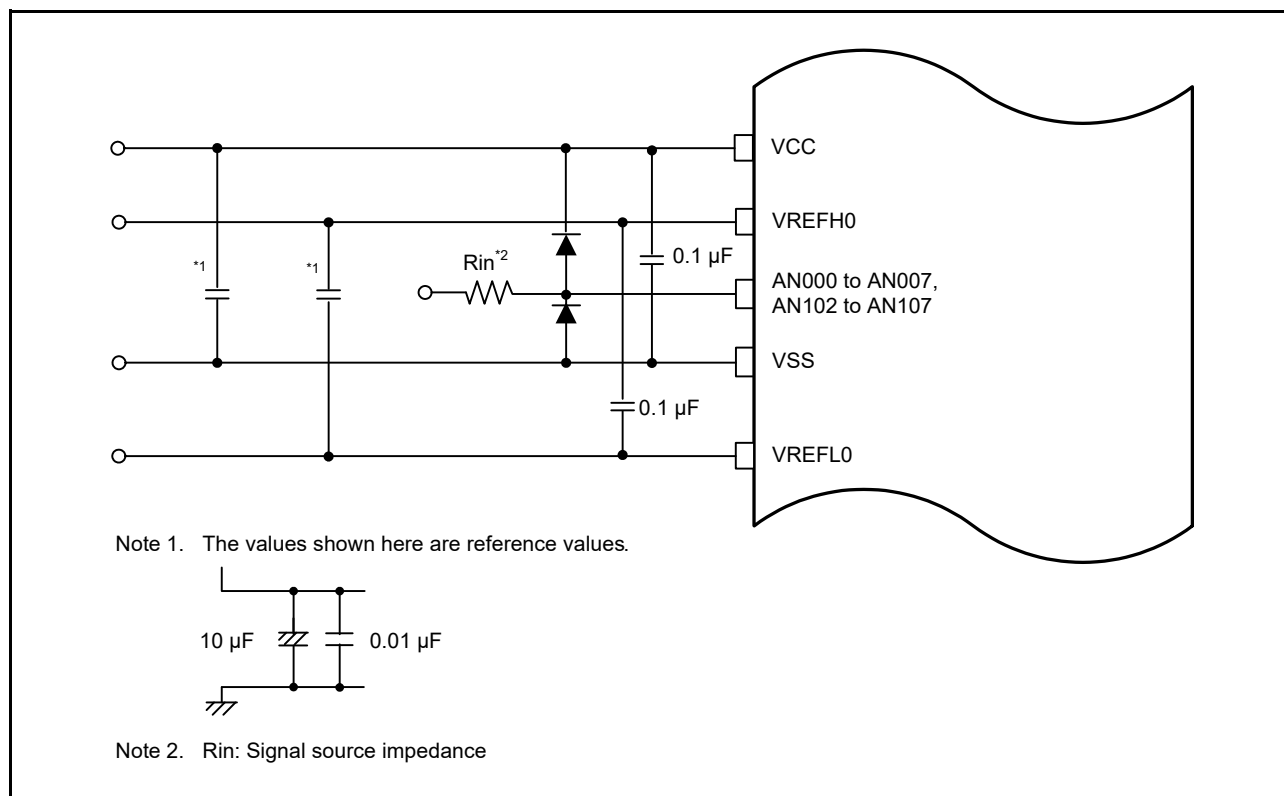


Figure 30.40 Sample Protection Circuit for Analog Inputs

31. Programmable Gain Amplifier (PGA)

31.1 Overview

This MCU incorporates programmable gain amplifiers (PGA00 to PGA02), which can amplify the input signals from the analog input pins PGAIN00 to PGAIN02. The amplified signals can be used as analog inputs for the A/D converter and the comparator C.

Table 31.1 lists the specifications of the programmable gain amplifiers, Table 31.2 lists the input/output pins of the programmable gain amplifiers and Table 31.3 lists the output signals (internal signals) of the programmable gain amplifiers. Figure 31.1 shows a block diagram of the programmable gain amplifiers.

Table 31.1 Specifications of Programmable Gain Amplifier

Item	Description
Number of units	Three units (PGA00, PGA01, and PGA02)
Gain	Selectable from 4, 8, 16, 32 for each unit
Ground of the feedback resistor for the programmable gain amplifier	Selectable from VSS or PGAGND for each unit
Programmable gain amplifier output monitor	Any of output from PGA00 to PGA02 can be output from the PGAOUT0 pin.

Table 31.2 Input/Output Pins of Programmable Gain Amplifier

Pin Name	I/O	Function
AN000/PGAIN00	Input	Analog input pin
AN001/PGAIN01	Input	Analog input pin
AN002/PGAIN02	Input	Analog input pin
VCC	Input	Power supply pin
VSS	Input	Ground pin
PGAGND	Input	Dedicated ground pin for programmable gain amplifier
PGAOUT0	Output	Programmable gain amplifier output monitor pin

Table 31.3 Output signal of Programmable Gain Amplifier (Internal Signal)

Signal Name	Description
PGAOUT00	Output of PGA00
PGAOUT01	Output of PGA01
PGAOUT02	Output of PGA02

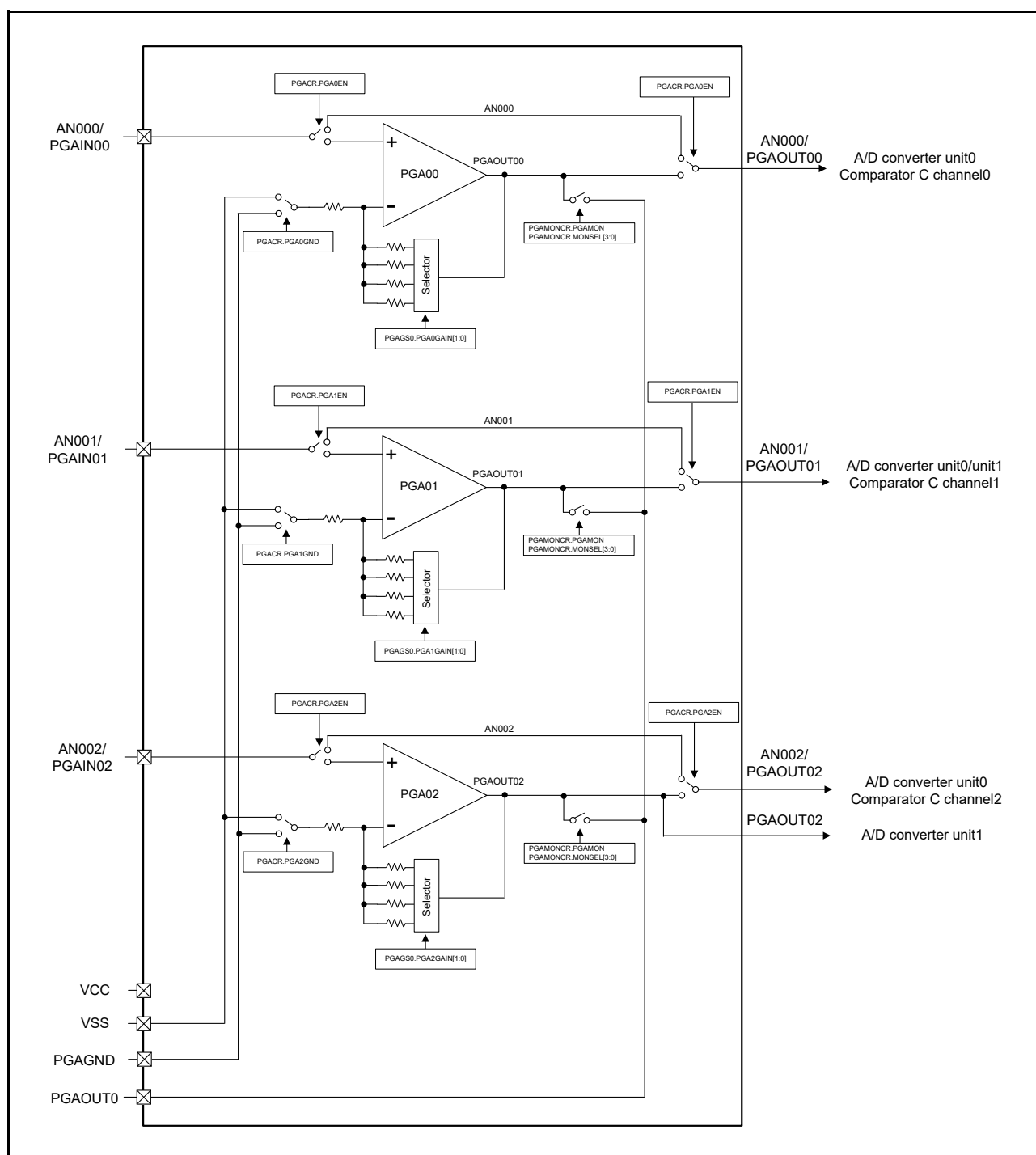


Figure 31.1 Block Diagram of Programmable Gain Amplifier

31.2 Register Descriptions

31.2.1 Programmable Gain Amplifier Control Register (PGACR)

Address(es): PGA0.PGACR 0008 91A0h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	PGA2GND	PGA1GND	PGA0GND	—	PGA2EN	—	—	—	PGA1EN	—	—	—	PGA0EN	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b2	PGA0EN	PGA0 Enable	0: PGA00 is disabled. 1: PGA00 is enabled.	R/W
b5 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	PGA1EN	PGA1 Enable	0: PGA01 is disabled. 1: PGA01 is enabled.	R/W
b9 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b10	PGA2EN	PGA2 Enable	0: PGA02 is disabled. 1: PGA02 is enabled.	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b12	PGA0GND	Ground Pin for PGA0 Select	0: VSS is selected.*1 1: PGAGND is selected.	R/W
b13	PGA1GND	Ground Pin for PGA1 Select	0: VSS is selected.*1 1: PGAGND is selected.	R/W
b14	PGA2GND	Ground Pin for PGA2 Select	0: VSS is selected.*1 1: PGAGND is selected.	R/W
b15	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note 1. Setting this bit to 0 causes the characteristics of the programmable gain amplifier to deteriorate as the current flowing into VSS increases.

The PGA0.PGACR register is used to enable or disable the programmable gain amplifiers PGA00 to PGA02.

PGA0EN Bit (PGA0 Enable)

This bit is used to enable or disable the programmable gain amplifier PGA00.

PGA1EN Bit (PGA1 Enable)

This bit is used to enable or disable the programmable gain amplifier PGA01.

PGA2EN Bit (PGA2 Enable)

This bit is used to enable or disable the programmable gain amplifier PGA02.

PGA0GND Bit (Ground Pin for PGA0 Select)

This bit is used to select the ground of the feedback resistor for the programmable gain amplifier PGA00.

PGA1GND Bit (Ground Pin for PGA1 Select)

This bit is used to select the ground of the feedback resistor for the programmable gain amplifier PGA01.

PGA2GND Bit (Ground Pin for PGA2 Select)

This bit is used to select the ground of the feedback resistor for the programmable gain amplifier PGA02.

31.2.2 Programmable Gain Amplifier Gain Setting Register 0 (PGAGS0)

Address(es): PGA0.PGAGS0 0008 91A2h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PGA2GAIN[1:0]	—	—	—	PGA1GAIN[1:0]	—	—	—	PGA0GAIN[1:0]	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	PGA0GAIN[1:0]	PGA0 Gain Setting	The relationship between each setting and the gain is as follows: b1 b0 0 0: × 4 0 1: × 8 1 0: × 16 1 1: × 32	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b5, b4	PGA1GAIN[1:0]	PGA1 Gain Setting	The relationship between each setting and the gain is as follows: b5 b4 0 0: × 4 0 1: × 8 1 0: × 16 1 1: × 32	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b9, b8	PGA2GAIN[1:0]	PGA2 Gain Setting	The relationship between each setting and the gain is as follows: b9 b8 0 0: × 4 0 1: × 8 1 0: × 16 1 1: × 32	R/W
b15 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The PGA0.PGAGS0 register is used to set the gain of programmable gain amplifiers PGA00 to PGA02.

PGA0GAIN[1:0] Bits (PGA0 Gain Setting)

These bits are used to set the gain of programmable gain amplifier PGA00.

PGA1GAIN[1:0] Bits (PGA1 Gain Setting)

These bits are used to set the gain of programmable gain amplifier PGA01.

PGA2GAIN[1:0] Bits (PGA2 Gain Setting)

These bits are used to set the gain of programmable gain amplifier PGA02.

31.2.3 Programmable Gain Amplifier Monitor Output Control Register (PGAMONCR)

Address(es): PGA0.PGAMONCR 0008 91E0h

b7	b6	b5	b4	b3	b2	b1	b0
MONSEL[3:0]				—	—	—	PGAMON
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	PGAMON	PGAOUT0 Enable	0: Programmable gain amplifier output monitor is disabled. 1: Programmable gain amplifier output monitor is enabled.	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7 to b4	MONSEL[3:0]	PGAOUT0 Select	b7 b4 0 0 0 0: Not select monitor signal 1 0 0 0: PGA00 is selected. 1 0 0 1: PGA01 is selected. 1 0 1 0: PGA02 is selected. Settings other than above are prohibited.	R/W

The PGA0.PGAMONCR register is used to control the programmable gain amplifier output monitor function.

PGAMON Bit (PGAOUT0 Enable)

This bit is used to select whether or not to output the programmable gain amplifier signal from the PGAOUT0 pin.

MONSEL[3:0] Bits (PGAOUT0 Select)

These bits are used to select which programmable gain amplifier to monitor.

31.3 Operation

The programmable gain amplifier amplifies the input signals from the analog input pins PGAIN00 to PGAIN02 within the MCU. The amplified signals can be used as analog inputs for the A/D converter and the comparator C.

For the input pins of the programmable gain amplifier and the units and storage locations of the A/D converter, please refer to Table 30.11 Analog Input Settings and A/D Conversion Result Storage Destination in section 30, 12-Bit A/D Converter (S12ADF).

For the corresponding input pins of the comparator C, please refer to Table 34.3 Analog Input Pin Connections for Comparator C in section 34, Comparator C (CMPCa).

The procedure for starting the operation of the programmable gain amplifier is shown in Figure 31.2, and the procedure for stopping the operation is shown in Figure 31.3.

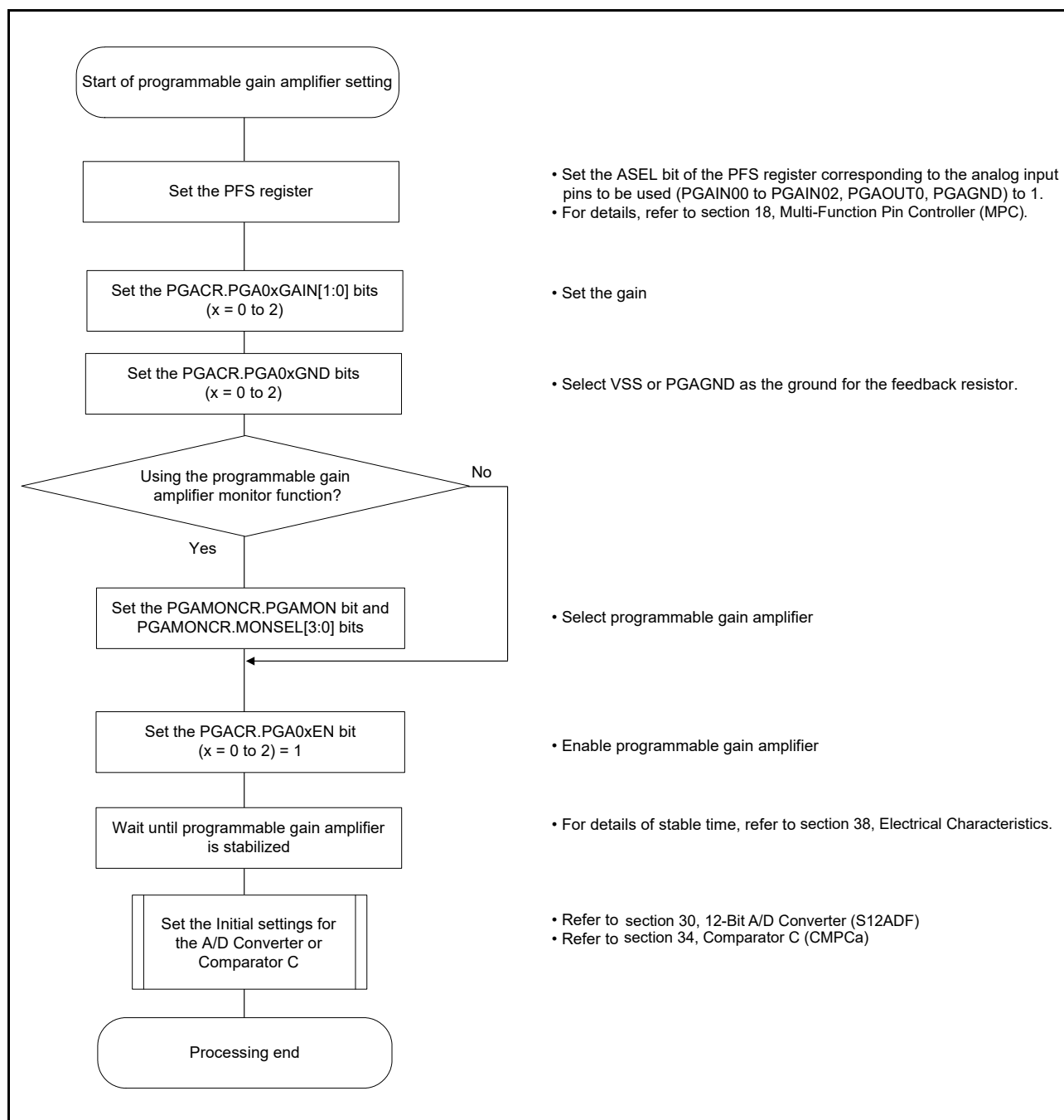


Figure 31.2 Operation Setting Flow of Programmable Gain Amplifier

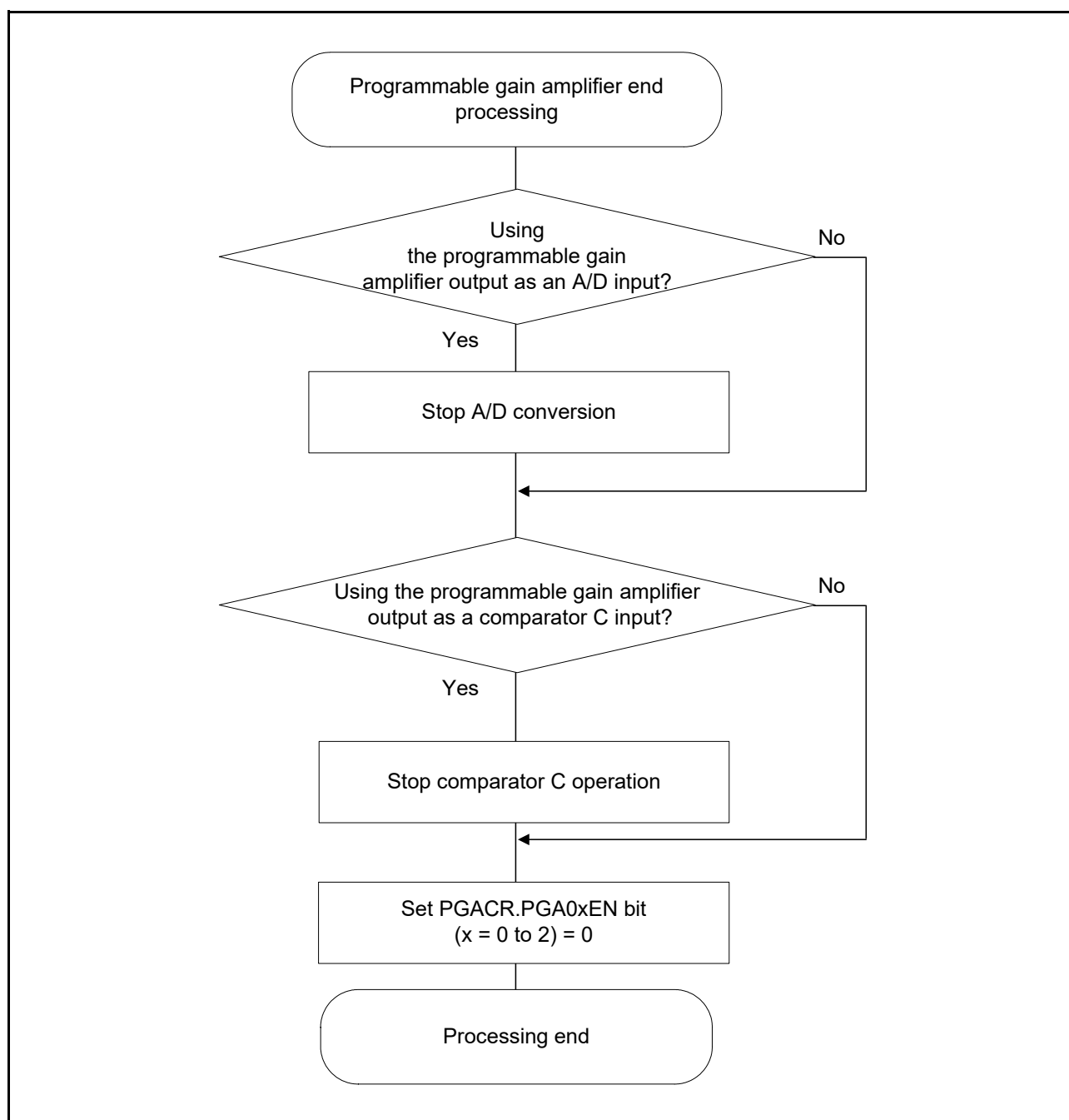


Figure 31.3 Operation Stopping Flow of Programmable Gain Amplifier

31.4 Usage Notes

31.4.1 Restrictions on Programmable Gain Amplifier Monitor Output Function

The programmable gain amplifier monitor output function is an auxiliary feature intended for MCU program development and debugging. Do not connect anything other than an oscilloscope to the PGAOUT0 pin. Additionally, external gain adjustment is not possible.

31.4.2 Notes on Using the PGAGND Pin

When $PGACR.PGA0xGND = 1$ ($x = 0$ to 2), apply the same voltage to PGAGND as that on VSS.

31.4.3 Module Stop Function Setting

The programmable gain amplifier is stopped after released from the reset state. Register access is enabled by releasing the module stop state. For details, refer to section 11, Low Power Consumption.

31.4.4 Programmable Gain Amplifier Operation in Module Stop State

When the module stop state is entered while programmable gain amplifier is operating, analog circuits in the programmable gain amplifier is not stopped and the analog power supply current is the same as that when programmable gain amplifier is being used. If the analog power supply current needs to be reduced in the module stop state, set the $PGACR.PGAxEN$ bit ($x = 0$ to 2) to 0 to stop programmable gain amplifier.

31.4.5 Programmable Gain Amplifier Operation in Software Standby Mode

When software standby mode is entered while programmable gain amplifier is operating, analog circuits in the programmable gain amplifier is not stopped and the analog power supply current is the same as that when programmable gain amplifier is being used. If the analog power supply current needs to be reduced in software standby mode, set the $PGACR.PGAxEN$ bit ($x = 0$ to 2) to 0 to stop programmable gain amplifier.

32. D/A Converter (DAa)

32.1 Overview

This MCU includes two channels of 8-bit D/A converter.

Channel 1 is used for generating comparator C reference input voltage. There is no analog output pin. Channel 0 is used for generating both comparator C reference input voltage and analog output voltage.

Table 32.1 lists the specifications of the 8-bit D/A converter and Figure 32.1 shows a block diagram of the 8-bit D/A converter.

Table 32.1 Specifications of 8-Bit D/A Converter

Item	Specifications
Resolution	8 bits
Output channels	Two channels
Measure against mutual interference between analog modules	Measure against interference between D/A and A/D conversion D/A converted data update timing is controlled by the 12-bit A/D converter synchronous D/A conversion enable signal from the 12-bit A/D converter (unit 1). Therefore, the degradation of A/D conversion accuracy due to interference is reduced by controlling the timing in which the 8-bit D/A converter inrush current occurs, with the enable signal.
Low power consumption function	Module stop state can be set.

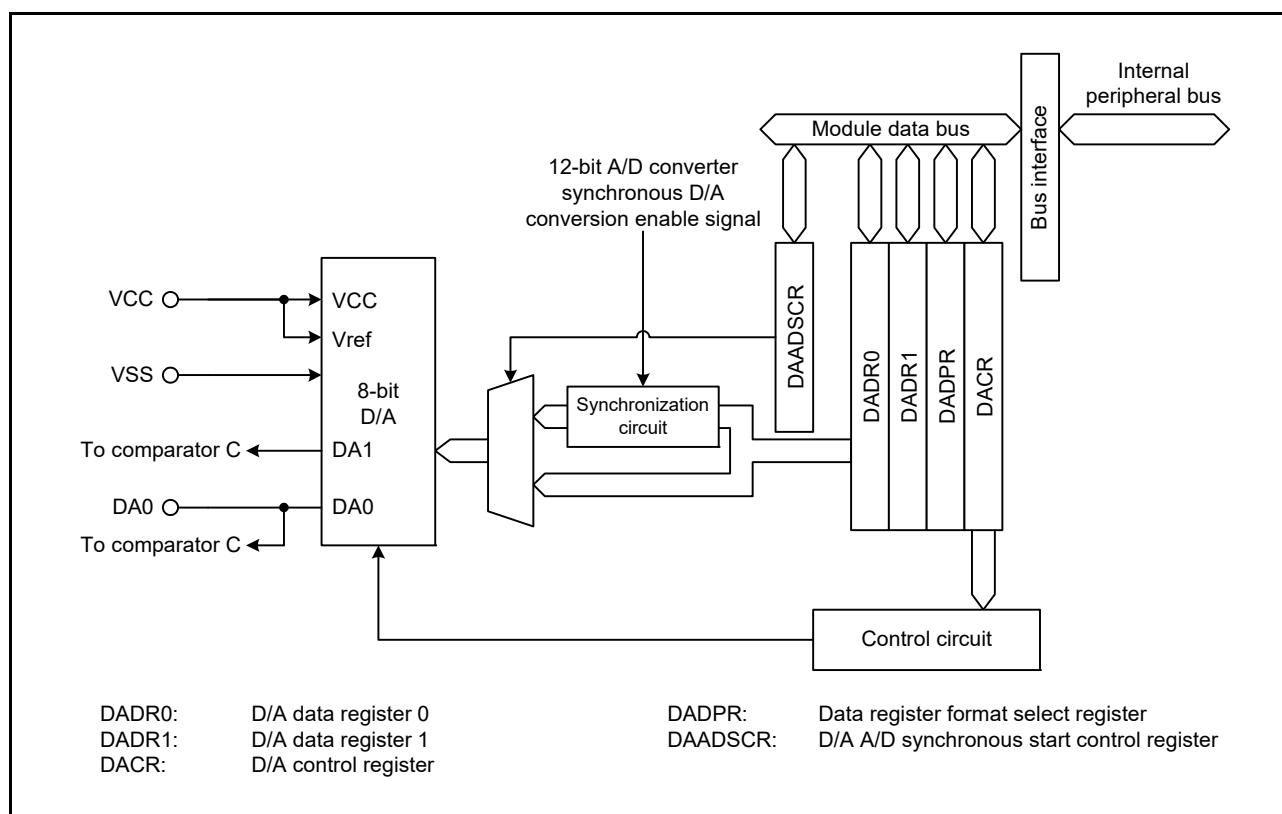


Figure 32.1 Block Diagram of 8-Bit D/A Converter

Table 32.2 lists the pin configuration of the 8-bit D/A converter.

Table 32.2 Pin Configuration of 8-Bit D/A Converter

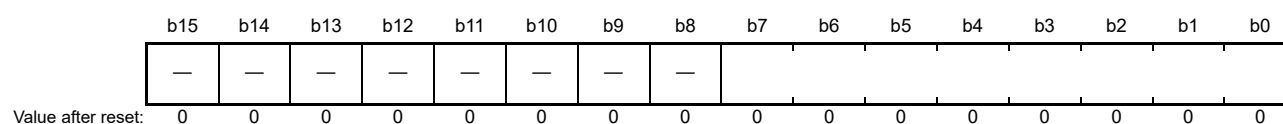
Pin Name	I/O	Function
VCC	Input	Power supply pin
VSS	Input	Ground pin
DA0	Output	Channel 0 analog output pin

32.2 Register Descriptions

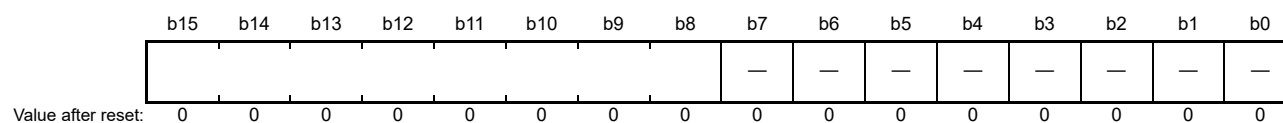
32.2.1 D/A Data Register m (DADRM) (m = 0, 1)

Address(es): DA.DADR0 0008 80C0h, DA.DADR1 0008 80C2h

- DADPR.DPSEL bit = 0 (data is right-justified)



- DADPR.DPSEL bit = 1 (data is left-justified)



The DADRM register is a 16-bit readable/writable register, which stores data to which D/A conversion is to be performed. Whenever an analog output is enabled, the values in DADRM are converted and output from the D/A converter.

8-bit data can be relocated by setting the DADPR.DPSEL bit.

Bits “—” are read as 0. The write value should be 0.

32.2.2 D/A Control Register (DACR)

Address(es): DA.DACR 0008 80C4h

b7	b6	b5	b4	b3	b2	b1	b0
DAOE1	DAOE0	—	—	—	—	—	—
0	0	0	1	1	1	1	1

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R
b6	DAOE0	D/A Output Enable 0	0: Analog output of channel 0 (DA0) is disabled. 1: D/A conversion of channel 0 is enabled. Analog output of channel 0 (DA0) is enabled.	R/W
b7	DAOE1	D/A Output Enable 1	0: D/A conversion of channel 1 (DA1) is disabled. 1: D/A conversion of channel 1 is enabled.	R/W

This register should be set when the DAADSCR.DAADST bit is 1 (measure against interference between D/A and A/D conversion is enabled) while the 12-bit A/D converter (unit 1) is halted (the ADCSR.ADST bit is 0). At that time, the software trigger should be selected for the 12-bit A/D converter (unit 1) trigger to securely stop the 12-bit A/D converter (unit 1).

DAOE0 Bit (D/A Output Enable 0)

The DAOE0 bit controls the D/A conversion and analog output.

DAOE1 Bit (D/A Output Enable 1)

The DAOE1 bit controls the D/A conversion.

32.2.3 Data Register Format Select Register (DADPR)

Address(es): DA.DADPR 0008 80C5h

b7	b6	b5	b4	b3	b2	b1	b0
DPSEL	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	DPSEL	Format Select	0: Data is right-justified. 1: Data is left-justified.	R/W

32.2.4 D/A A/D Synchronous Start Control Register (DAADSCR)

Address(es): DA.DAADSCR 0008 80C6h

	b7	b6	b5	b4	b3	b2	b1	b0
	DAADST	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	DAADST	D/A A/D Synchronous Conversion	0: 8-bit D/A converter operation does not synchronize with 12-bit A/D converter (unit 1) operation. (measure against interference between D/A and A/D conversion is disabled) 1: 8-bit D/A converter operation synchronizes with 12-bit A/D converter (unit 1) operation. (measure against interference between D/A and A/D conversion is enabled)	R/W

As a measure against interference between D/A and A/D conversion, the DAADSCR register selects whether or not the timing for starting 8-bit D/A conversion is synchronized with the 12-bit A/D converter synchronous D/A conversion enable signal from the 12-bit A/D converter (unit 1).

This register should be set while the 12-bit A/D converter (unit 1) is halted (while the ADCSR.ADST bit is 0 after selecting software trigger as the 12-bit A/D converter (unit 1) trigger).

DAADST Bit (D/A A/D Synchronous Conversion)

Setting the DAADST bit to 0 allows the DADDR_m register value ($m = 0, 1$) to be converted into analog data at any time. Setting the DAADST bit to 1 allows synchronous D/A conversion with the 12-bit A/D converter synchronous D/A conversion enable signal from the 12-bit A/D converter (unit 1). Therefore, even if the DADDR_m register value is modified, D/A conversion does not start until the 12-bit A/D converter (unit 1) completes A/D conversion.

Set this bit while the ADCSR.ADST bit is set to 0. At this time, the software trigger should be selected for the 12-bit A/D converter trigger to securely stop the 12-bit A/D converter (unit 1).

32.3 Operation

The 8-bit D/A converter includes D/A conversion circuits for two channels, each of which can operate independently. When the DACR.DA0Em bit (m = 0, 1) is set to 1, D/A converter is enabled and the conversion result is output. An operation example of D/A conversion on channel 0 is shown below. Figure 32.2 shows the timing of this operation.

- (1) Set the data for D/A conversion in the DADPR.DPSEL bit and the DADR0 register.
- (2) Set the DACR.DA0E0 bit to 1 to start D/A conversion. The DA0 output settles to the voltage corresponding to the setting value after the conversion time t_{DCONV} has elapsed. The DA0 output voltage is held at this level until the DADR0 register is updated or the DA0E0 bit is set to 0. The output voltage (reference) is expressed by the following formula:

$$\frac{\text{Value of DADRm register}}{256} \times V_{\text{CC}}$$

- (3) When the DADR0 register is updated, the conversion starts. The DA0 output settles at the new output voltage after the conversion time t_{DCONV} has elapsed.
When the DAADSCR.DAADST bit is 1 (measure against interference between D/A and A/D conversion is enabled), it takes a maximum of one A/D conversion time for D/A conversion to start. When ADCLK is faster than the peripheral module clock, it may take longer than one A/D conversion time.
- (4) When the DA0E0 bit is set to 0, analog output is disabled.

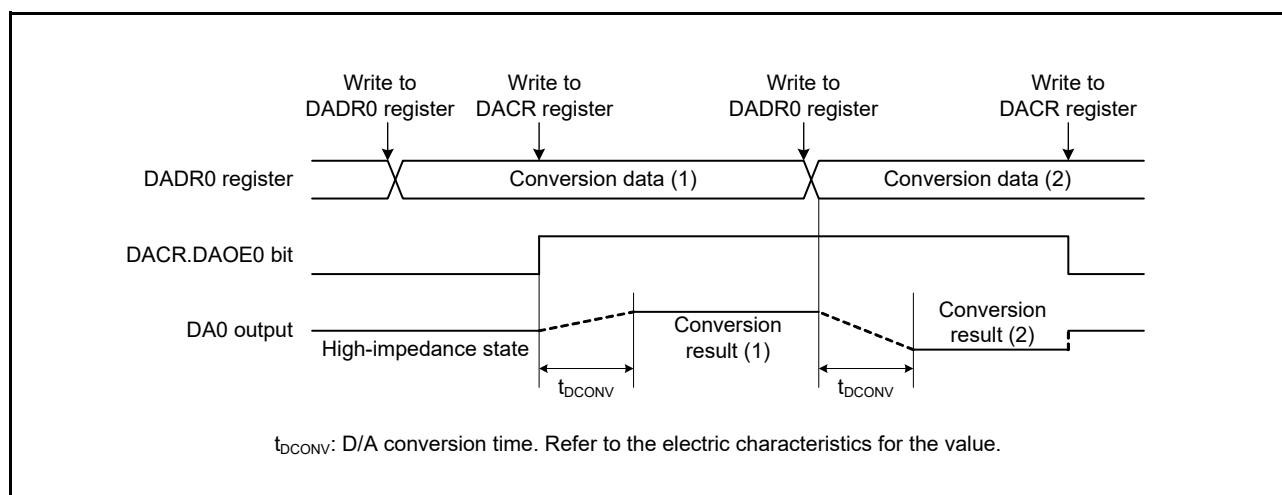


Figure 32.2 Example of 8-Bit D/A Converter Operation

32.3.1 Measure against Interference between D/A and A/D Conversion

When D/A conversion starts, an inrush current occurs to the 8-bit D/A converter. Since the same analog power supply is shared by the 8-bit D/A converter and 12-bit A/D converter (unit 1), the inrush current may interfere with the proper operation of the 12-bit A/D converter (unit 1).

With the DAADSCR.DAADST bit being 1, even if the DADRm register data ($m = 0, 1$) is modified during 12-bit A/D converter (unit 1) operation, D/A conversion does not start immediately but starts synchronously with A/D conversion completion. It takes a maximum of one A/D conversion time for the DADRm register data update to be reflected as the D/A conversion circuit input. Before reflection, the DADRm register value does not correspond to the analog output value.

When this function is enabled, it is impossible to check by any software means whether the DADRm register value has been D/A converted or not.

Even with the DAADSCR.DAADST bit being 1, when the DADRm register data is modified while the 12-bit A/D converter (unit 1) is halted, D/A conversion starts in one PCLKB cycle.

Figure 32.3 shows an example of channel 0 D/A conversion, in which the 8-bit D/A converter operates synchronously with the 12-bit A/D converter (unit 1).

- (1) Confirm that the 12-bit A/D converter (unit 1) is halted. Set the DAADSCR.DAADST bit to 1.
- (2) Confirm that the 12-bit A/D converter (unit 1) is halted. Set the DACR.DAOE0 bit to 1.
- (3) Set the DADR0 register. When ADCLK is faster than the peripheral module clock, it may take longer than one A/D conversion time.
 - If the 12-bit A/D conversion is halted (ADCSR.ADST bit = 0) when the DADR0 register is modified, D/A conversion starts in one PCLKB cycle.
 - If the 12-bit A/D conversion is in progress (ADCSR.ADST bit = 1) when the DADR0 register is modified, D/A conversion starts upon A/D conversion completion. If the DADR0 register is modified twice during A/D conversion, the first update may not be converted.

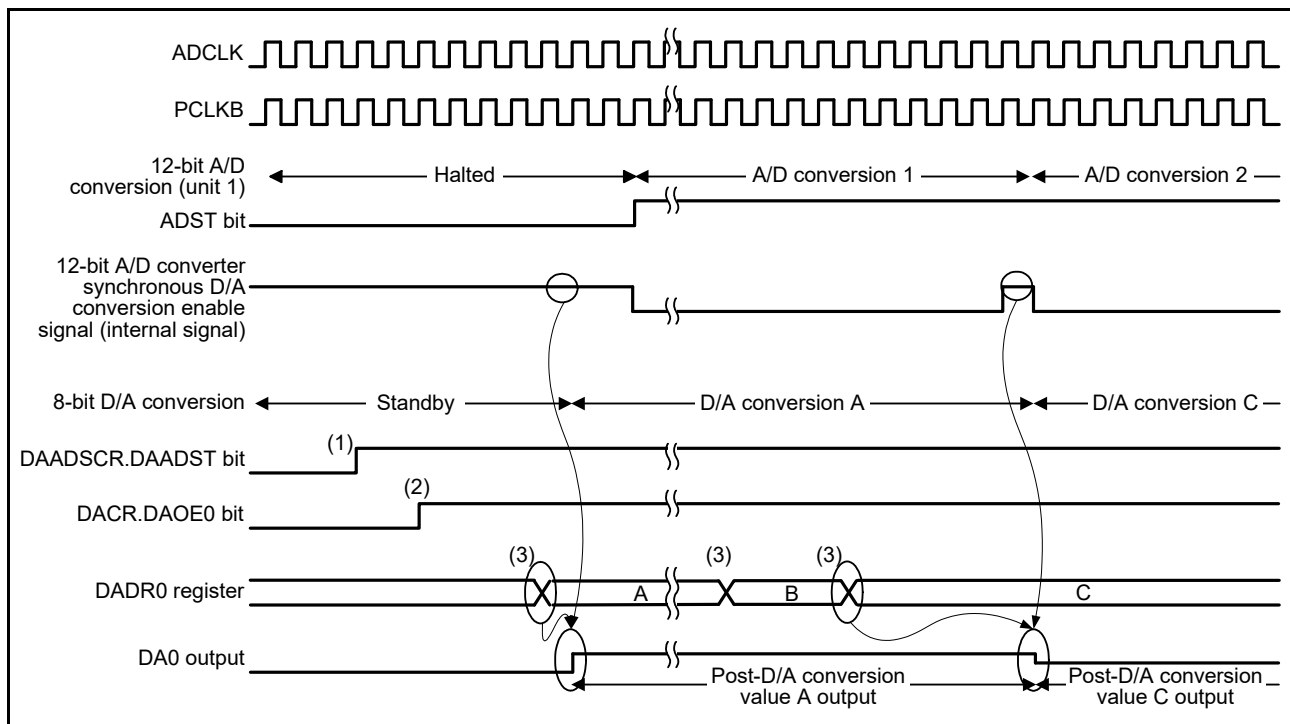


Figure 32.3 Example of Conversion When the 8-Bit D/A Converter is Synchronized with the 12-Bit A/D Converter (Unit 1)

When ADCLK is faster than PCLKB, the 8-bit D/A converter may not be able to capture a 12-bit A/D converter synchronous D/A conversion enable signal for one ADCLK cycle which is output between A/D conversion 1 and A/D conversion 2. Figure 32.4 shows example when the 8-bit D/A converter cannot capture the 12-bit A/D converter synchronous D/A conversion enable signal. In this case, the DA0 output is held at the level of the post-D/A conversion value A.

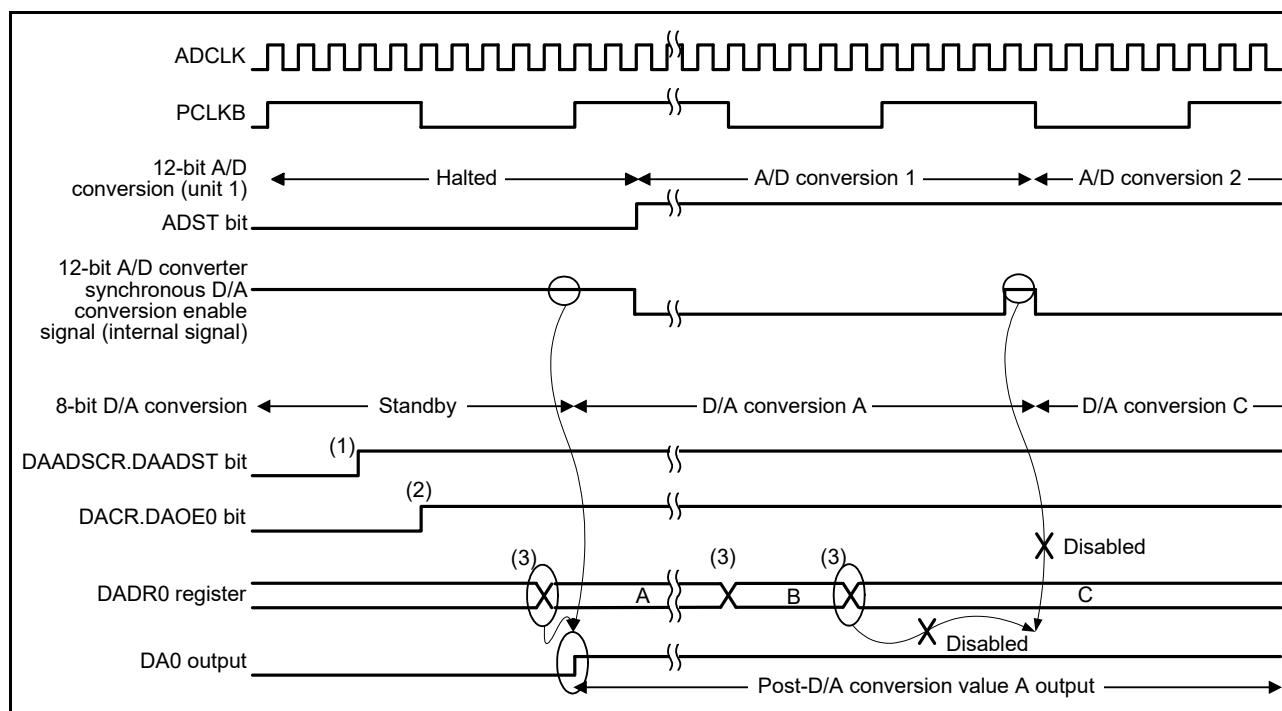


Figure 32.4 Example When the 8-Bit D/A Converter Cannot Capture the 12-Bit A/D Converter Synchronous D/A Conversion Enable Signal

32.4 Usage Notes

32.4.1 Module Stop Function Setting

Operation of the 8-bit D/A converter can be disabled or enabled using the module stop control register. The initial setting is for operation of the 8-bit D/A converter to be stopped. Register access is enabled by releasing the module stop state. For details, refer to section 11, Low Power Consumption.

32.4.2 Operation of the D/A Converter in Module Stop State

When the MCU enters the module stop state with D/A conversion enabled, the D/A converter outputs are retained, and the analog power supply current is the same as during D/A conversion. If the analog power supply current has to be reduced in the module stop state, disable D/A conversion by setting the DACR.DAOE1, and DAOE0 bits to 0.

32.4.3 Operation of the D/A Converter in Software Standby Mode

When the MCU enters software standby mode with D/A conversion enabled, the D/A converter outputs are retained, and the analog power supply current is the same as during D/A conversion. If the analog power supply current has to be reduced in software standby mode, disable D/A conversion by setting the DACR.DAOE1, and DAOE0 bits to 0.

32.4.4 Setting the D/A Converter

When using the D/A converter output voltage as a reference input voltage of the comparator C, set the D/A converter and wait for the D/A converter output settling time ($t_{D\text{CONV}}$) before enabling the comparator. Similarly, before making any changes to the settings of the D/A converter stop the comparator temporarily, and after the changes are made, wait for the D/A converter output settling time before enabling the comparator.

32.4.5 Note on Usage When Measure against Interference between D/A and A/D Conversion is Enabled

When the DAADSCR.DAADST bit is 1 (measure against interference between D/A and A/D conversion is enabled), do not place the 12-bit A/D converter (unit 1) in the module stop state. It may halt D/A conversion in addition to A/D conversion.

32.4.6 Notes on Simultaneous Use of DA0 Pin Output and Comparator C Reference Voltage

When the output voltage of the D/A converter is used simultaneously as the DA0 pin output and the reference input voltage for the comparator C, the D/A conversion characteristics specified in the electrical characteristics may not be satisfied. A full evaluation is recommended before use when using them simultaneously.

33. Temperature Sensor (TEMPSA)

33.1 Overview

This MCU has a built-in temperature sensor. The temperature sensor outputs a voltage proportional to temperature. By converting the output voltage of the temperature sensor into a digital value with a 12-bit A/D converter unit 1 and converting it into temperature, the internal temperature of the MCU can be obtained.

Table 33.1 lists the specifications of the temperature sensor. Figure 33.1 shows a overall block diagram of the temperature sensor system.

Table 33.1 Temperature Sensor Specifications

Item	Description
Temperature sensor voltage output	The temperature sensor voltage is output to the 12-bit A/D converter.

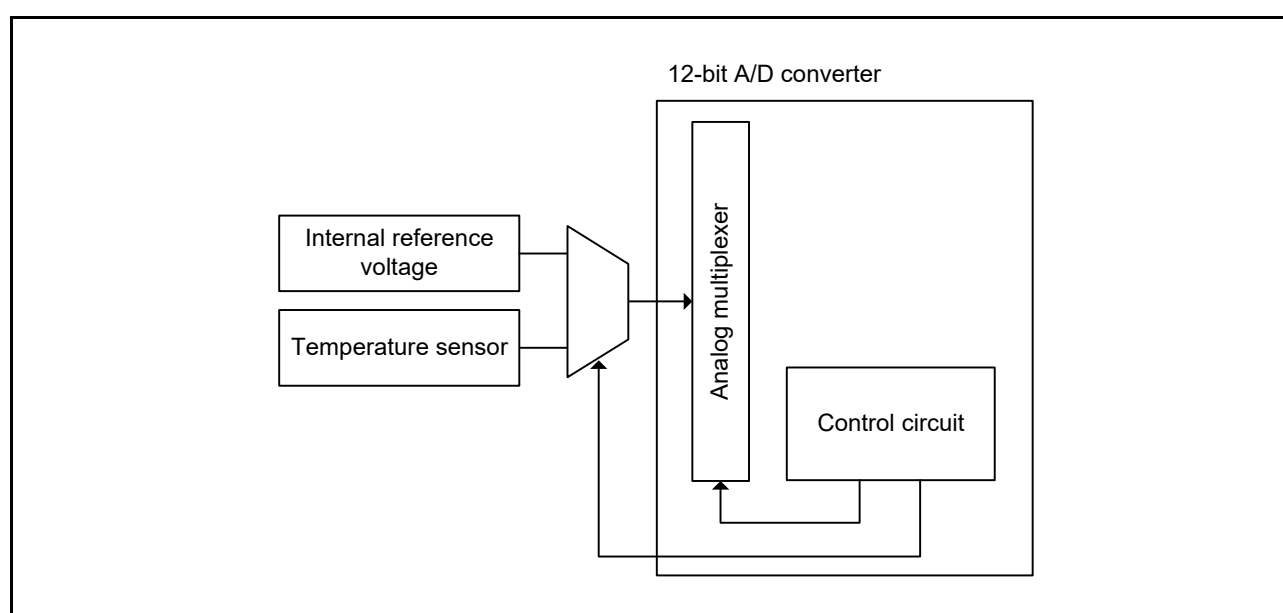
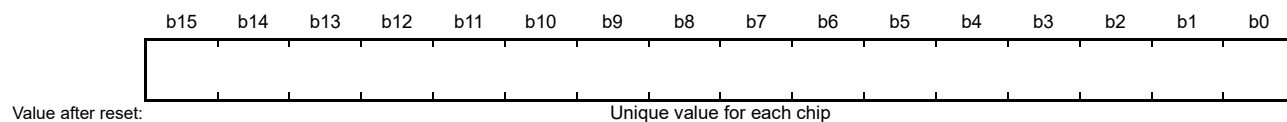


Figure 33.1 Block Diagram of Temperature Sensor System

33.2 Register Descriptions

33.2.1 Temperature Sensor Calibration Data Register (TSCDR)

Address(es): TEMPS.TSCDR 007F C228h



The TSCDR register stores temperature sensor calibration data measured for each chip at factory shipment.

Temperature sensor calibration data is the value (CAL_{140}) obtained by digitally converting the temperature sensor output voltage at $T_j = 140^{\circ}\text{C}^{*1}$ using the 12-bit A/D converter under the condition of $VCC = VREFH0 = 5\text{ V}$.

The output voltage $V1$ of the temperature sensor at $T_j = 140^{\circ}\text{C}$ can be calculated from the converted value CAL_{140} according to the formula below.

$$V1 = 5 \times CAL_{140} / 4096 \quad (\text{V})$$

Note 1. The temperature sensor output voltage $V1$ is independent of both the VCC and $VREFH0$ voltages.

33.3 Using the Temperature Sensor

The temperature sensor outputs a voltage proportional to temperature. By converting this voltage into a digital value with a 12-bit A/D converter and converting it into temperature, the internal temperature of the MCU can be obtained.

33.3.1 Preparation for Using the Temperature Sensor

Perform a calibration of the temperature sensor as shown below. The voltage output by the temperature sensor is proportional to temperature, which can be calculated according to the following formula.

Formula for the temperature characteristic:

$$T = (V_s - V_1)/\text{Slope} + T_1$$

T: Measured temperature (°C)

V_s: Voltage output by the temperature sensor when the temperature is measured (V)

T₁: Sample temperature measurement at first point (°C)

V₁: Voltage output by the temperature sensor when T₁ is measured (V)

T₂: Sample temperature measurement at second point (°C)

V₂: Voltage output by the temperature sensor when T₂ is measured (V)

Slope: Temperature slope of the temperature sensor (V/°C); Slope = (V₂ - V₁)/(T₂ - T₁)

Since there are individual differences in temperature sensors, it is recommended to prepare a temperature slope by performing trial measurements at two different temperatures as shown below.

Use the 12-bit A/D converter to measure the voltage V₁ output by the temperature sensor at temperature T₁.

Again, using the 12-bit A/D converter, measure the voltage V₂ output by the temperature sensor at a different temperature T₂. Obtain the temperature slope (Slope = (V₂ - V₁)/(T₂ - T₁)) from these results.

Subsequently, obtain temperatures by substituting the slope into the formula for the temperature characteristic (T = (V_s - V₁)/Slope + T₁).

If you are using the temperature slope given in **section 38, Electrical Characteristics**, use the A/D converter to measure the voltage V₁ output by the temperature sensor at temperature T₁, and then calculate the temperature characteristic by using the formula below.

However, this method produces less accurate temperatures than measurement at two points.

$$T = (V_s - V_1)/\text{Slope} + T_1$$

In this MCU, the TSCDR register stores the temperature value (CAL₁₄₀) of the temperature sensor measured under the condition of T_j = 140°C and VCC = VREFH0 = 5 V. By using this value as the sample measurement result at the first point, preparation before using the temperature sensor can be omitted.

If V₁ is calculated from CAL₁₄₀,

$$V_1 = 5 \times \text{CAL}_{140}/4096 \quad (\text{V})$$

Using this, the measured temperature can be calculated according to the formula below.

$$T = (V_s - V_1)/\text{Slope} + 140 \quad (^\circ\text{C})$$

T: Measured temperature (internal temperature of the MCU) ($^{\circ}\text{C}$)

Vs: Voltage output by the temperature sensor when the temperature is measured (V)

V1: Voltage output by the temperature sensor when $T_j = 140^{\circ}\text{C}$ and $V_{CC} = V_{REFH0} = 5\text{ V}$ (V)

Slope: Temperature slope listed in Table 38.56 $\div 1000$ ($\text{V}/^{\circ}\text{C}$)

Error in the measured temperature (the range of variation is 3σ) is shown in Figure 33.2.

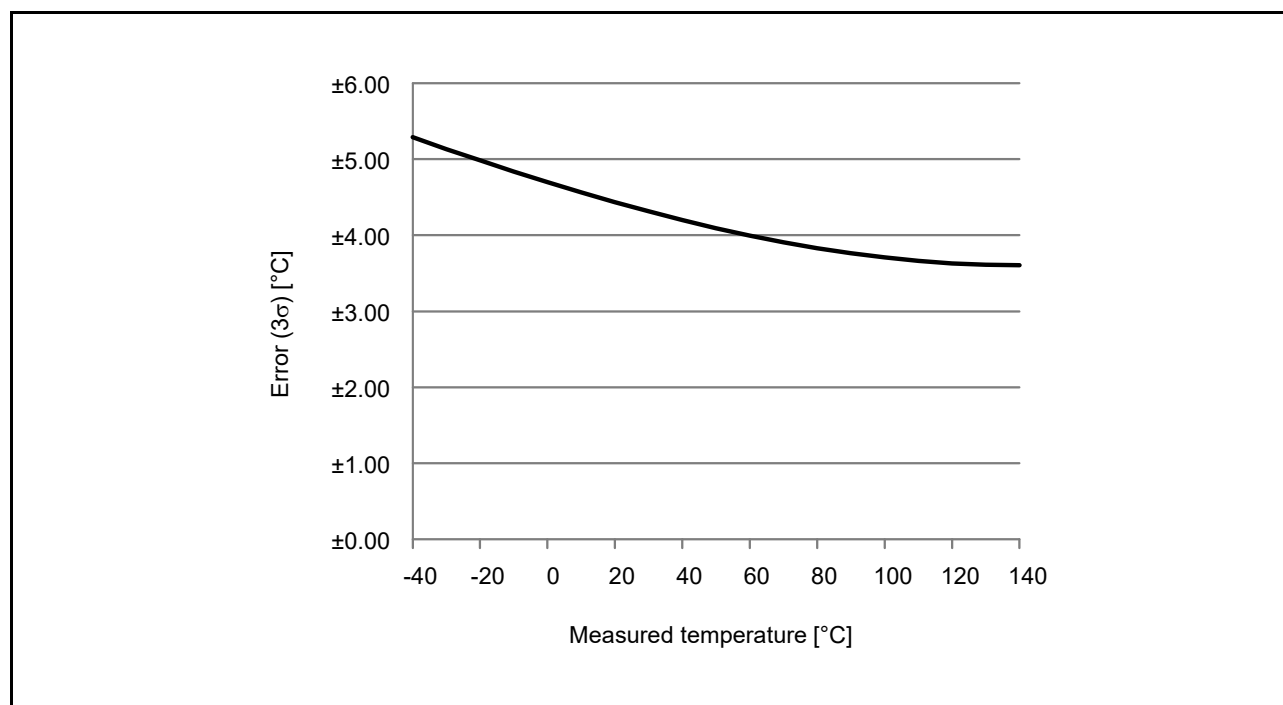


Figure 33.2 Error in the Measured Temperature (Designed Value)

33.3.2 Setting the 12-Bit A/D Converter

For details, refer to section 30, 12-Bit A/D Converter (S12ADF).

34. Comparator C (CMPCa)

34.1 Overview

Comparator C compares a reference input voltage to an analog input voltage.

The comparison result can be read by software and output externally, and an interrupt request can be generated upon any changes to the comparison result.

The reference input voltage of comparator C can be selected from the input from the CVREFC0 pin, or the output from the on-chip D/A converter 0 or the on-chip D/A converter 1.

There are four analog inputs, one of which is to be selected.

Table 34.1 lists the specification of comparator C, Figure 34.1 shows a block diagram of comparator C, Table 34.2 shows a comparator C pin configuration, and Table 34.3 shows the analog input pin connections for comparator C.

In this section, “PCLK” is used to refer to PCLKB.

Table 34.1 Comparator C Specifications

Item	Specification
Number of channels	Three (comparator C0 to comparator C2)
Analog input voltages	Input voltage from the CMPCnm pin (n = channel number; m = 0 to 3)
Reference input voltage	Input voltage from the CVREFC0 pin, or output voltage from on-chip D/A converter 0 or on-chip D/A converter 1
Comparison result	The comparison result can be output externally.
Digital noise filter function	- The noise filter function can be enabled or disabled. - One of eight sampling periods can be selected. - A noise-filtered signal can be used to generate the interrupt request output and POE source output*1, and the signal can be used to read the comparison result via registers.
Interrupt request	- An interrupt request is generated upon detecting a valid edge of the comparison result. - A valid edge can be selected from a rising or a falling edge or both edges.
Low power consumption function	Module stop state can be set.

Note 1. The POE only uses the level detection signal, and the POEG uses the level detection and edge detection signals.

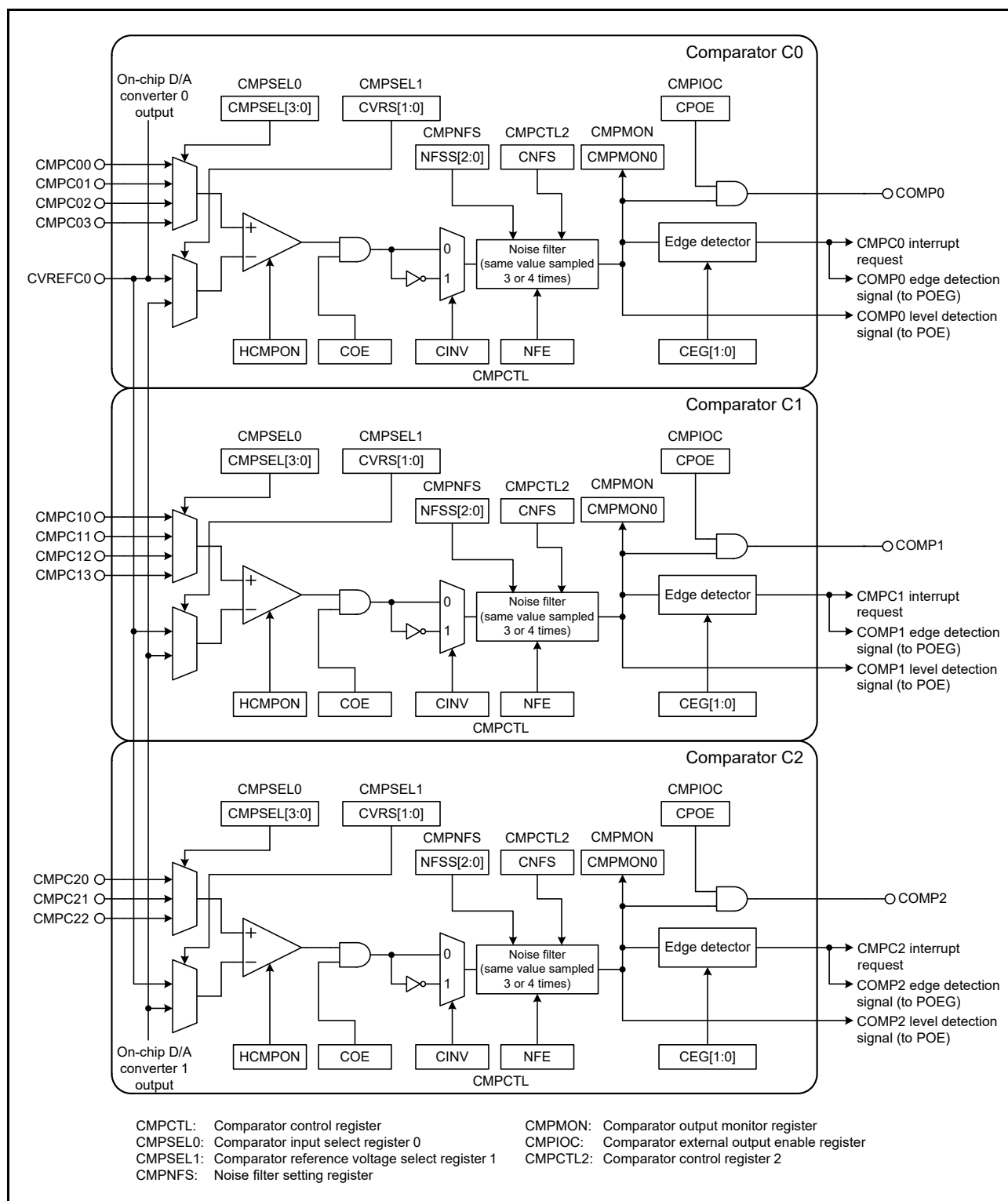


Figure 34.1 Block Diagram of Comparator C

Table 34.2 Comparator C Pin Configuration

Pin Name	I/O	Function
CMPC00, CMPC01, CMPC02, CMPC03	Input	Comparator C0 analog input pins
CMPC10, CMPC11, CMPC12, CMPC13	Input	Comparator C1 analog input pins
CMPC20, CMPC21, CMPC22	Input	Comparator C2 analog input pins
CVREFC0	Input	Reference input voltage pin 0
COMP0	Output	Comparator C0 comparison result output pin
COMP1	Output	Comparator C1 comparison result output pin
COMP2	Output	Comparator C2 comparison result output pin

Table 34.3 Analog Input Pin Connections for Comparator C

Analog Input Pin	Connection
CMPC00	AN000 pin
CMPC01	Programmable gain amplifier output for AN000 pin
CMPC02	AN102 pin
CMPC03	AN004 pin
CMPC10	AN001 pin
CMPC11	Programmable gain amplifier output for AN001 pin
CMPC12	AN105 pin
CMPC13	AN006 pin
CMPC20	AN002 pin
CMPC21	Programmable gain amplifier output for AN002 pin
CMPC22	AN106 pin

34.2 Register Descriptions

34.2.1 Comparator Control Register (CMPCTL)

Address(es): CMPC0.CMPCTL 000A 0C80h, CMPC1.CMPCTL 000A 0CA0h, CMPC2.CMPCTL 000A 0CC0h

b7	b6	b5	b4	b3	b2	b1	b0
HCMP ON	—	NFE	CEG[1:0]	—	COE	CINV	
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CINV	Comparator Output Polarity Select*1, *4	0: Comparator output not inverted 1: Comparator output inverted	R/W
b1	COE	Comparator Output Enable	0: Comparator output disabled (the output signal is fixed to 0) 1: Comparator output enabled	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4, b3	CEG[1:0]	Comparator Edge Interrupt Detection Select	b4 b3 0 0: Interrupt request is not generated. 0 1: Rising edge 1 0: Falling edge 1 1: Both edges	R/W
b5	NFE	Noise Filter Enable	0: Noise filter is disabled. 1: Noise filter is enabled.	R/W
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	HCMPON	Comparator Operation Enable*3	0: Operation stopped (the output signal is fixed to 0) 1: Operation enabled (input to the comparator pins is enabled)	R/W

Note: Set this register before setting registers in the POE when using comparator output as a POE source.

Note 1. Rewrite the NFE and CINV bits only after disabling the comparator output (COE bit = 0).

Note 2. If the NFE bit is changed from 0 (noise filter is disabled) to 1 (noise filter is enabled), allow four sampling times to elapse until the filter output is updated, and then use the CMPCn interrupt request output and a POE source output.

Note 3. The operation stabilization wait time is required after enabling comparator operation (HCMPON bit = 1). As for the value, refer to section 38, Electrical Characteristics.

Note 4. Rewriting the CINV bit or NFE bit may generate a CMPCn interrupt request and POE source. Before changing these bits, set the registers in the POE/POEG so that comparator output is not used for output disabling control. After changing these bits, also set the corresponding interrupt status flag (IR) in the interrupt request register, the POE comparator channel n output detection flag (n = 0 to 2), and the GPTW or CMPC output stop request detection flag in the POEG to 0.

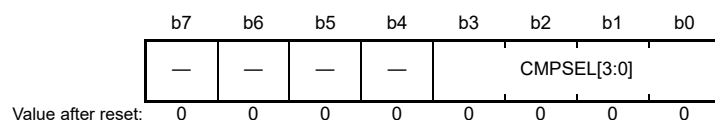
CEG[1:0] Bits (Comparator Edge Interrupt Detection Select)

These bits select which edge of comparator output signal is used to generate an interrupt request.

The valid edge is set for the signal after the comparator polarity is selected by the CINV bit and the filter is selected by NFE bit.

34.2.2 Comparator Input Select Register (CMPSEL0)

Address(es): CMPC0.CMPSEL0 000A 0C84h, CMPC1.CMPSEL0 000A 0CA4h, CMPC2.CMPSEL0 000A 0CC4h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	CMPSEL[3:0]	Comparator Input Select*1, *2	• Comparator C0 b3 b0 0 0 0 0: No input 0 0 0 1: CMPC00 selected 0 0 1 0: CMPC01 selected 0 1 0 0: CMPC02 selected 1 0 0 0: CMPC03 selected Settings other than above are prohibited. • Comparator C1 b3 b0 0 0 0 0: No input 0 0 0 1: CMPC10 selected 0 0 1 0: CMPC11 selected 0 1 0 0: CMPC12 selected 1 0 0 0: CMPC13 selected Settings other than above are prohibited. • Comparator C2 b3 b0 0 0 0 0: No input 0 0 0 1: CMPC20 selected 0 0 1 0: CMPC21 selected 0 1 0 0: CMPC22 selected Settings other than above are prohibited.	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

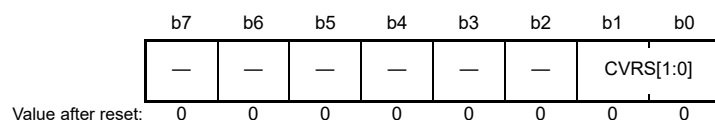
Note 1. Rewrite the CMPSEL[3:0] bits in the following procedure. Writing a value other than 0000b while the value of these bits is not 0000b is invalid. Writing 1 to two or more bits is also invalid. In both cases, the previous value is retained.

- (1) Set the CMPCTL.COE bit to 0.
- (2) Set the CMPSEL[3:0] bits to 0000b.
- (3) Set a new value to the CMPSEL[3:0] bits (with 1 set in only one of the bits).
- (4) Wait for the stabilization time for input selection. As for the value, refer to section 38, Electrical Characteristics.
- (5) Set the CMPCTL.COE bit to 1.
- (6) Set the corresponding interrupt status flag (IR) in the interrupt request register to 0.
- (7) Set the GPTW or CMPC output stop request detection flag (POEGGA.IOCF) in the corresponding POEG to 0.

Note 2. If COMPn level detection signal is used as a POE source, note that write access to these bits after the setting of any register in the POE may generate a POE source.

34.2.3 Comparator Reference Voltage Select Register (CMPSEL1)

Address(es): CMPC0.CMPSEL1 000A 0C88h, CMPC1.CMPSEL1 000A 0CA8h, CMPC2.CMPSEL1 000A 0CC8h



Bit	Symbol	Bit Name	Description	R/W
b1, b0	CVRS[1:0]	Reference Input Voltage Select *1, *2, *3	b1 b0 0 0: No input 0 1: On-chip D/A converter 1 output voltage selected as reference input voltage 1 0: On-chip D/A converter 0 output voltage or input voltage to the CVREFC0 pin selected as reference input voltage Settings other than above are prohibited.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. When the on-chip D/A converter output voltage is used, set the D/A converter for generating comparator C reference voltage before enabling comparator operation (CMPCTL.HCMPON bit = 1). For details on setting the D/A converter, refer to section 32, D/A Converter (DAa).

Note 2. Rewrite the CVRS[1:0] bits in the following procedure. Be sure to set the CVRS[1:0] bits to 00b before changing the set value. Direct rewriting from 01b to 10b or 10b to 01b, for example, will be ignored even if it is tried.

- (1) Set the CMPCTL.COE bit to 0.
- (2) Set the CVRS[1:0] bits to 00b.
- (3) Set a new value to the CVRS[1:0] bits (with 1 set in only one of the bits).
- (4) Wait for the stabilization time for input selection. As for the value, refer to section 38, Electrical Characteristics.
- (5) Set the CMPCTL.COE bit to 1.
- (6) Set the corresponding interrupt status flag (IR) in the interrupt request register to 0.
- (7) Set the GPTW or CMPC output stop request detection flag (POEGGA.IOCF) in the corresponding POEG to 0.

Note 3. If COMPn level detection signal is used as a POE source, note that write access to these bits after the setting of any register in the POE may generate a POE source.

34.2.4 Comparator Output Monitor Register (CMPMON)

Address(es): CMPC0.CMPMON 000A 0C8Ch, CMPC1.CMPMON 000A 0CACH, CMPC2.CMPMON 000A 0CCCh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	CMPMON0
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	CMPMON0	Comparator Output Monitor Flag *1	0: Comparator output is 0. 1: Comparator output is 1.	R
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. When comparator operation is enabled (CMPCTL.HCMPON and COE bits are 1) while the noise filter is disabled (CMPCTL.NFE bit is 0), read the CMPMON0 flag twice and use the value only when the results match.

34.2.5 Comparator External Output Enable Register (CMPIOC)

Address(es): CMPC0.CMPIOC 000A 0C90h, CMPC1.CMPIOC 000A 0CB0h, CMPC2.CMPIOC 000A 0CD0h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	CPOE
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	CPOE	External Pin Output Enable	Comparison result by the comparator is output to an external pin. 0: Output to the COMPn pin (n = 0 to 2) is disabled (the output signal is fixed to low) 1: Output to the COMPn pin is enabled	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

34.2.6 Comparator Control Register 2 (CMPCTL2)

Address(es): CMPC0.CMPCTL2 000A 0C98h, CMPC1.CMPCTL2 000A 0CB8h, CMPC2.CMPCTL2 000A 0CD8h

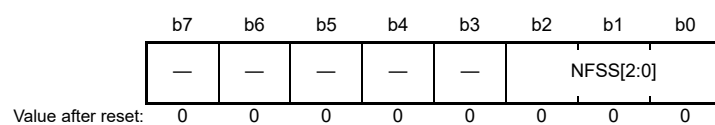
b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	CNFS
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	CNFS	Noise Filter Judgment Standard Setting	0: Output changes when the same value is sampled 3 times. 1: Output changes when the same value is sampled 4 times.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

34.2.7 Noise Filter Setting Register (CMPNFS)

Address(es): CMPC0.CMPNFS 000A 0C9Ch, CMPC1.CMPNFS 000A 0CBCh, CMPC2.CMPNFS 000A 0CDCh



Bit	Symbol	Bit Name	Description	R/W
b2 to b0	NFSS[2:0]	Noise Filter Sampling Select* ¹	b2 b0 0 0 0: Sampling frequency is PCLK/2. 0 0 1: Sampling frequency is PCLK/4. 0 1 0: Sampling frequency is PCLK/8. 0 1 1: Sampling frequency is PCLK/16. 1 0 0: Sampling frequency is PCLK/32. 1 0 1: Sampling frequency is PCLK/64. 1 1 0: Sampling frequency is PCLK/128. 1 1 1: Sampling frequency is PCLK/256.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Rewrite the NFSS[2:0] bits only after disabling the comparator output (CMPCTL.COE bit = 0).

34.3 Operation

34.3.1 Comparator Operation Example

Figure 34.2 shows an operation example of the comparator. The COMPn level detection signal (n = 0 to 2) becomes high when the analog input voltage is higher than the reference input voltage, and the COMPn level detection signal becomes low when the analog input voltage is lower than the reference input voltage (when the CMPCTL.CINV bit is 0). When the CPOE bit in the corresponding CMPIOC register is 1, the COMPn level detection signal is output from the COMPn pin. Interrupt request and COMPn edge detection signal are output in response to changes in the comparator output.

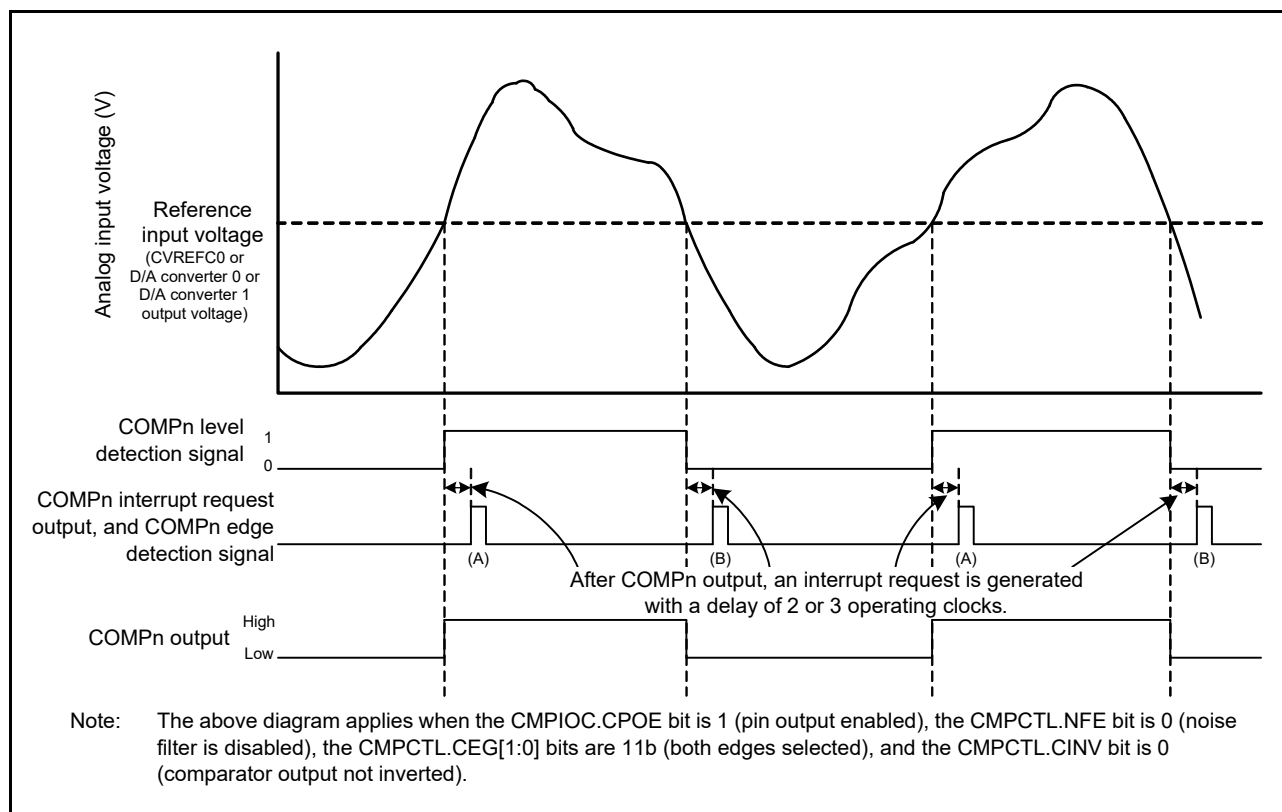


Figure 34.2 Comparator Operation Example (n = 0 to 2)

34.3.2 Noise Filter

Comparator C contains a noise filter. The sampling clock can be selected by the CMPNFS.NFSS[2:0] bits and the number of matches can be selected by the CMPCTL2.CNFS bit.

The comparator output signal is sampled every sampling clock, and if the same value is sampled three or four times, that value is determined as the noise filter output at the next sampling clock.

Figure 34.3 shows the configuration of the noise filter and edge detector and Figure 34.4 shows an example of the comparator noise filter and interrupt operation.

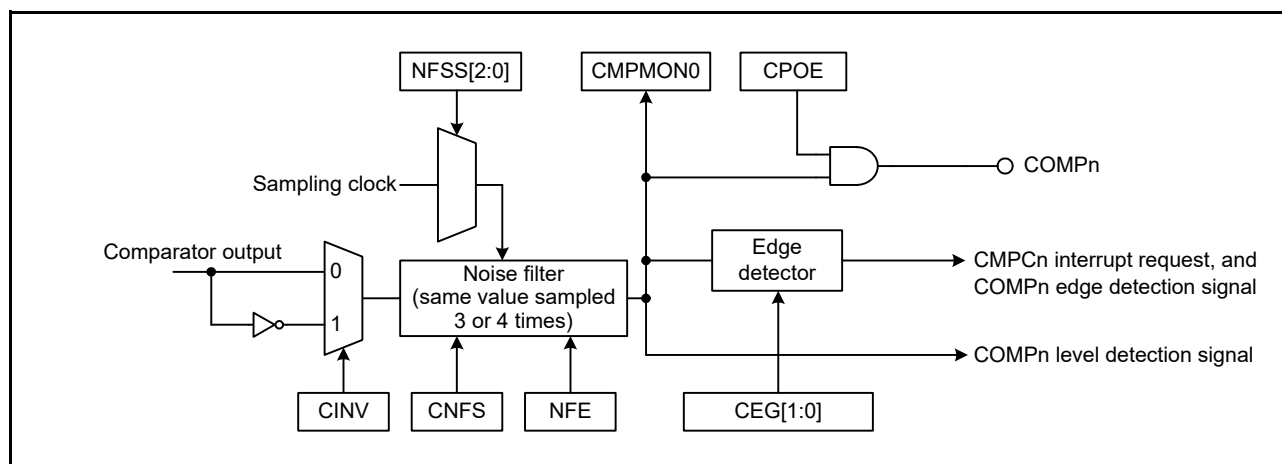


Figure 34.3 Noise Filter and Edge Detector Configuration (n = 0 to 2)

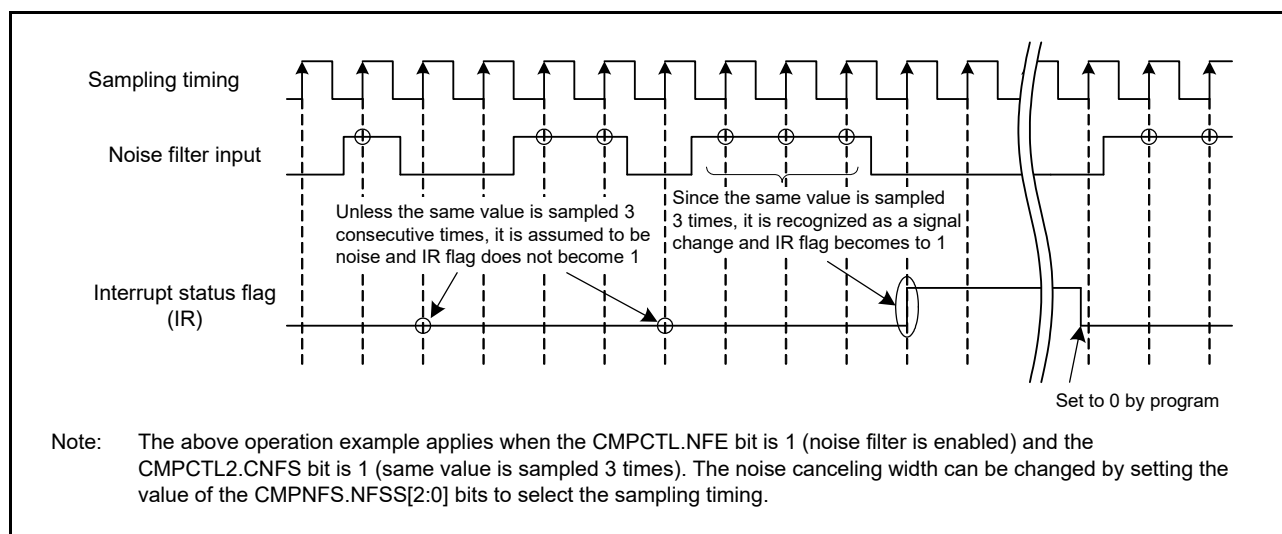


Figure 34.4 Noise Filter and Interrupt Operation Example

34.3.3 Interrupts

Comparator C generates an interrupt request upon detecting any changes in the comparison result.

When using the CMPCn interrupt, set at least one of bits CMPCTL.CEG[1:0] to 1 (to a value other than 00b (interrupt request is not generated)).

To use the CMPCn interrupt, use the following setting procedure. Note that steps (1), (2), and (3) can be set in any order.

- (1) When using the on-chip D/A converter output voltage as the reference input voltage, set the D/A converter for generating comparator C reference voltage and enable operation.
- (2) Set the CMPSEL0 or CMPSEL1 register to set the input of the comparator.
- (3) Set the CMPCTL.CINV bit, the CMPNFS.NFSS[2:0] bits, and the CMPCTL2.CNFS bit to select inversion or non-inversion processing and the sampling timing and number of matches of the noise filter.
- (4) Set the CMPCTL.NFE bit to 1.
- (5) Enable the edge for interrupt detection (set the CMPCTL.CEG[1:0] bits to a value other than 00b).
- (6) Enable input of the comparator (set the CMPCTL.HCMPON bit to 1) and wait for the time until the comparator operation is stabilized. As for the value, refer to **section 38, Electrical Characteristics**.
- (7) Enable output of the comparator (set the CMPCTL.COE bit to 1).

34.3.4 Comparator Pin Output

The comparison results can be output to the COMPn pins (n = 0 to 2). The CMPCTL.CINV bit can be used to set the output polarity (non-inverted output or inverted output), and the CMPIOC.CPOE bit can be used to enable or disable the output.

To output the comparison result to the external pin COMPn, use the following setting procedure. Note that the ports are set to input after reset.

- (1) Execute steps (1) to (3) and steps (5) and (6) shown in **section 34.3.3, Interrupts**.
- (2) Output the comparison result by the comparator to an external pin (set the CMPIOC.CPOE bit to 1).
- (3) Set the port mode register and the pin function control register corresponding to each comparator output pin.

34.3.5 Comparator Setting Flowchart

Figure 34.5 shows the flowchart for setting the comparator-related registers.

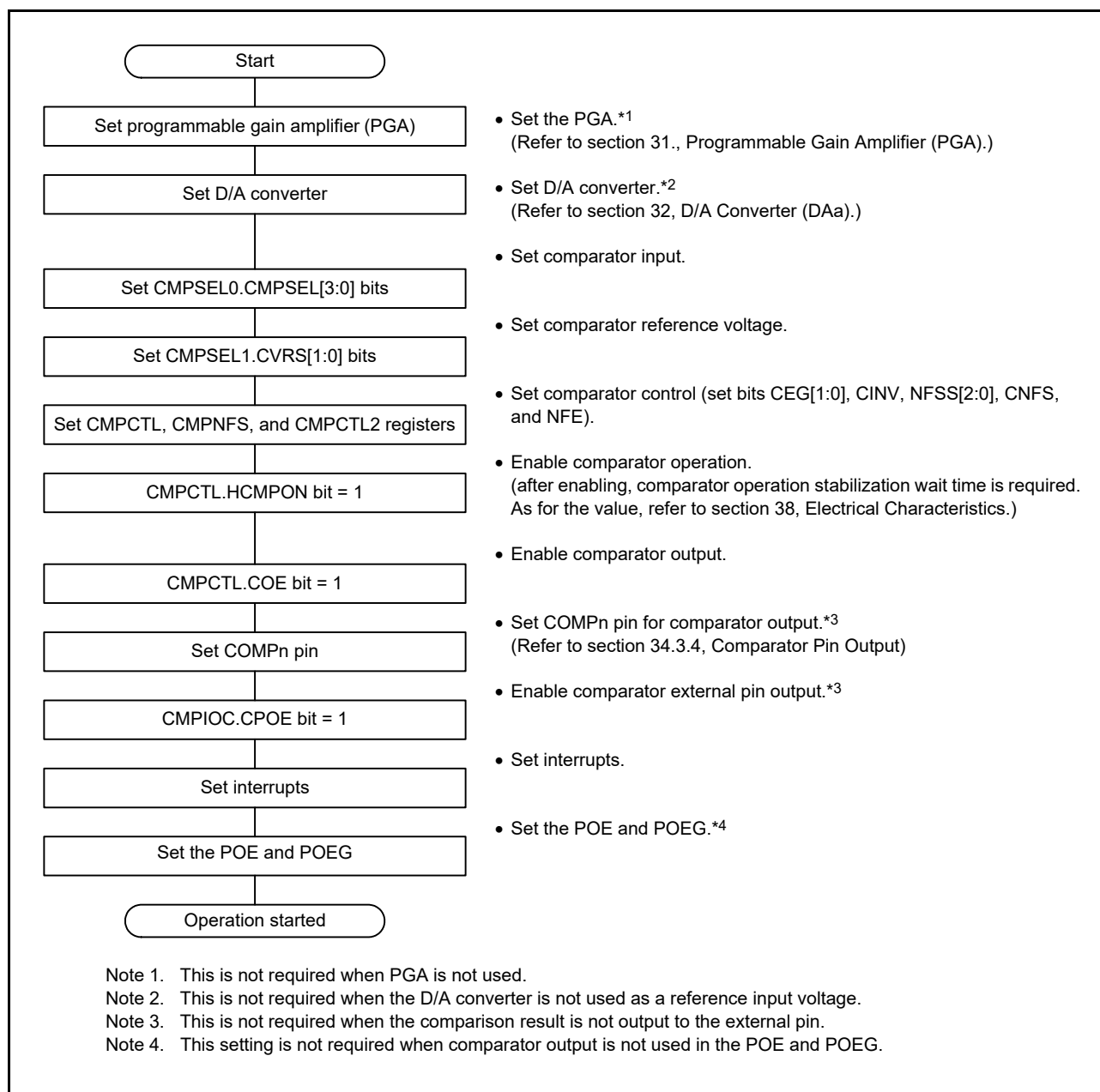


Figure 34.5 Comparator Operation Setting Flowchart (n = 0 to 2)

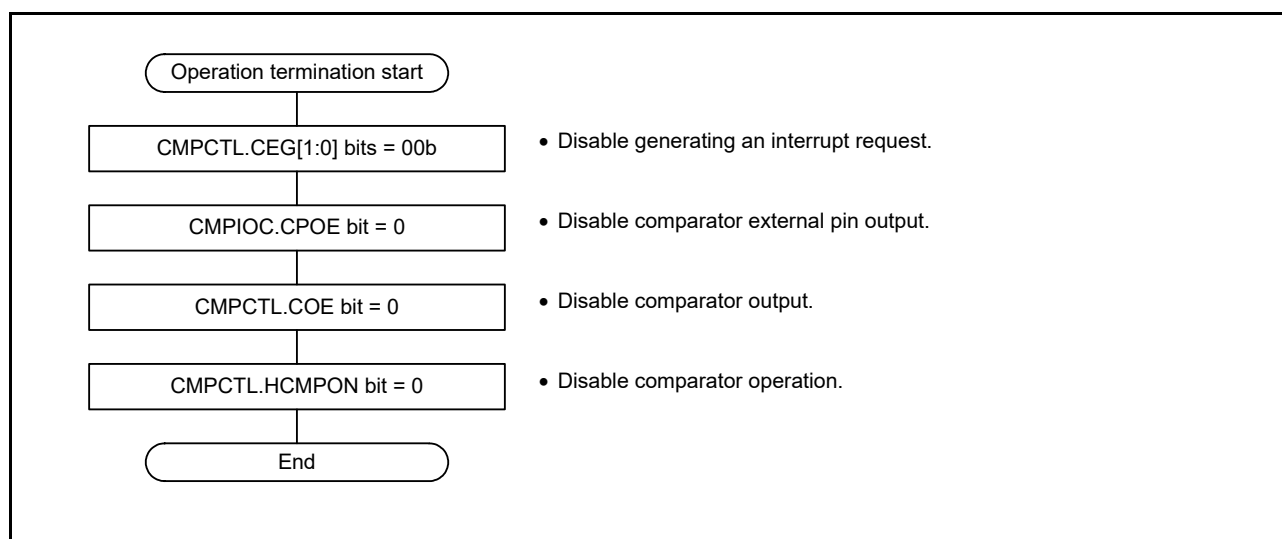


Figure 34.6 Comparator Operation Termination Flowchart

34.4 Usage Notes

34.4.1 Module Stop Function Setting

Operation of comparator C can be disabled or enabled using module stop control register B (MSTPCRB). After the reset, comparator C is halted. Register access is enabled by releasing the module stop state. For details, refer to section 11, Low Power Consumption.

34.4.2 Comparator C Operation in Module Stop State

When the module stop state is entered while comparator C is operating, analog circuits in the comparator C is not stopped and the analog power supply current is the same as that when comparator C is being used. If the analog power supply current needs to be reduced in the module stop state, set the CMPCTL.HCMPON bit to 0 to stop comparator C.

34.4.3 Comparator C Operation in Software Standby Mode

When software standby mode is entered while comparator C is operating, analog circuits in the comparator C is not stopped and the analog power supply current is the same as that when comparator C is being used. If the analog power supply current needs to be reduced in software standby mode, set the CMPCTL.HCMPON bit to 0 to stop comparator C.

34.4.4 Comparator Operation while the 12-Bit A/D Converter is in the Module-Stop State

The same module stop signal controls the programmable gain amplifiers (PGAs) and the 12-bit A/D converter. The comparison of PGA output for the following pins is not possible while the 12-bit A/D converter is in the module stop state.

- PGA output for AN000 pin
- PGA output for AN001 pin
- PGA output for AN002 pin

The comparison for the following analog input pins is possible since they are directly connected to the comparator, even if the 12-bit A/D converter is in the module-stop state.

- AN000 pin
- AN001 pin
- AN002 pin
- AN102 pin
- AN105 pin
- AN106 pin
- AN004 pin
- AN006 pin

34.4.5 Setting the D/A Converter for Generating Reference Voltage

Set the D/A converter for generating reference voltage and wait for the D/A converter output settling time before enabling the comparator. Similarly, before making any changes to the settings of the D/A converter stop the comparator temporarily, and after the changes are made, wait for the D/A converter output settling time before enabling the comparator.

34.4.6 Appearance of a Pulse in the Result of Comparison

If the analog input voltage or reference input voltage of the comparator changes rapidly with a slope exceeding $\pm 3.0 \text{ V}/\mu\text{s}$, a pulse with a width of up to 150 ns may appear in the result of comparison.

If this effect causes a problem, apply one of the following measures.

- Set the CMPCTL.NFE bit to 1 to enable the noise filter.
- Insert a noise filter for the output from the COMPn pin (n = 0 to 2).
- In polling of the CMPMON.CMPMON0 flag, ensure that short pulses can be eliminated by setting the polling interval to at least 150 ns, or by increasing the number of required matches.

35. Data Operation Circuit (DOC)

35.1 Overview

The data operation circuit (DOC) is used to compare, add, or subtract 16-bit values.

Table 35.1 lists the specifications of the DOC and Figure 35.1 is a block diagram of the DOC.

An interrupt can be generated if the result of 16-bit comparison meets one of the set interrupt conditions.

Table 35.1 DOC Specifications

Item	Description
Data operation function	16-bit data comparison, addition, and subtraction
Lower power consumption function	The DOC can be placed in a module-stop state.
Interrupts	- The result of data comparison meets the detection condition. - The result of data addition is greater than FFFFh, which is an overflow. - The result of data subtraction is less than 0000h, which is an underflow.

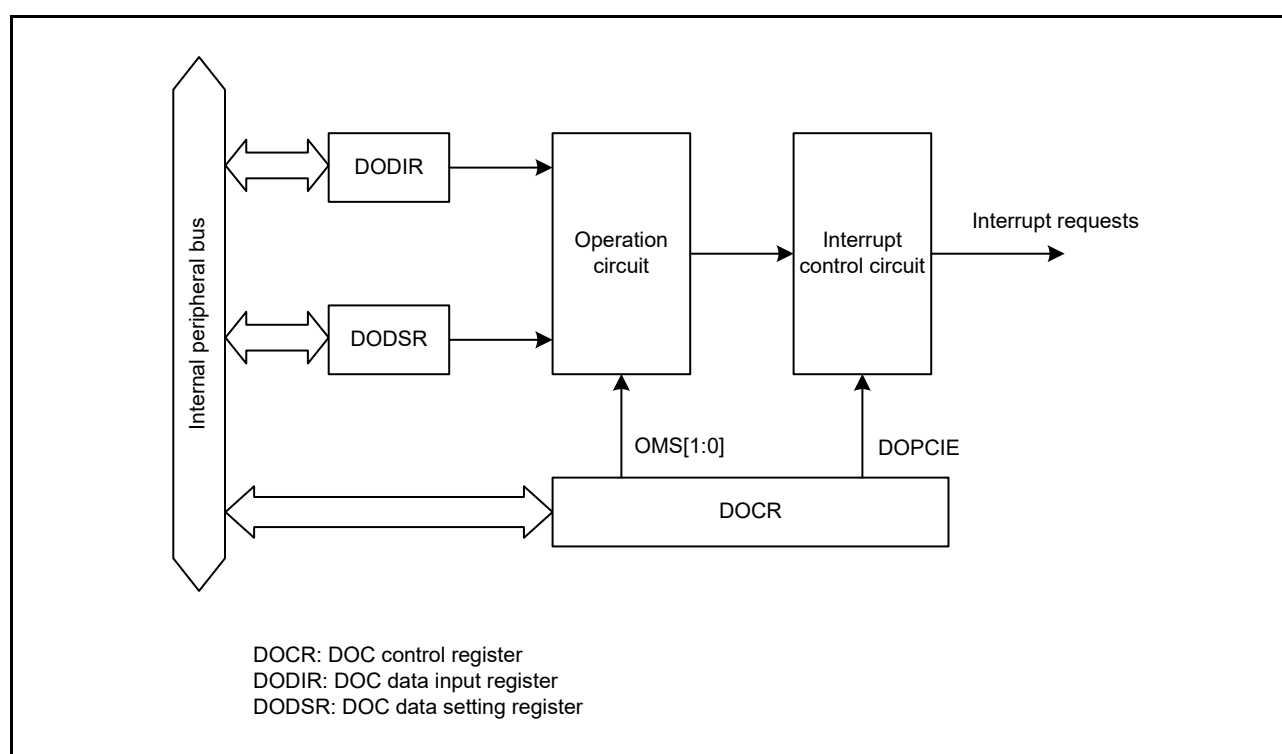


Figure 35.1 DOC Block Diagram

35.2 Register Descriptions

35.2.1 DOC Control Register (DOCR)

Address(es): DOC.DOCR 0008 B080h

b7	b6	b5	b4	b3	b2	b1	b0
—	DOPCF CL	DOPCF	DOPCI E	—	DCSEL	OMS[1:0]	
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	OMS[1:0]	Operating Mode Select	b1 b0 0 0: Data comparison mode 0 1: Data addition mode 1 0: Data subtraction mode 1 1: Setting prohibited	R/W
b2	DCSEL	Detection Condition Select*1	0: 'Not equal to' is to be detected. 1: 'Equal to' is to be detected.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	DOPCIE	Data Operation Circuit Interrupt Enable	0: Interrupt disabled 1: Interrupt enabled	R/W
b5	DOPCF	Data Operation Result Flag	Indicates the result of an operation.	R
b6	DOPCFCL	Data Operation Result Clear	0: Retain the value of the DOPCF flag. 1: Clears the DOPCF flag.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note 1. Valid only when data comparison mode is selected.

The DOCR register specifies the operation of DOC, or enabling or disabling of the interrupt.

OMS[1:0] Bits (Operating Mode Select)

These bits select the operating mode of the DOC.

DCSEL Bit (Detection Condition Select)

This bit is valid only when data comparison mode is selected.

This bit selects the condition for detection in data comparison mode.

DOPCIE Bit (Data Operation Circuit Interrupt Enable)

Setting this bit to 1 enables interrupts from the DOC.

DOPCF Flag (Data Operation Result Flag)

[Setting conditions]

- The condition selected by the DCSEL bit is met
- A result of data addition is greater than FFFFh
- A result of data subtraction is less than 0000h

[Clearing condition]

- Writing 1 to the DOPCFCL bit

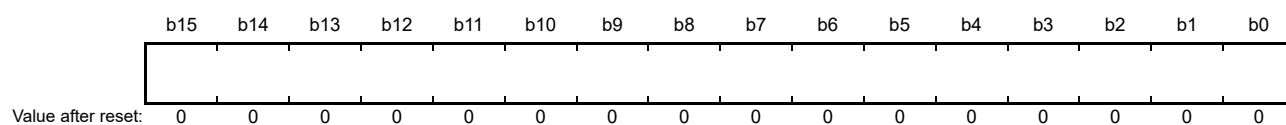
DOPCFCL Bit (Data Operation Result Clear)

Writing 1 to this bit clears the DOPCF flag.

This bit is read as 0.

35.2.2 DOC Data Input Register (DODIR)

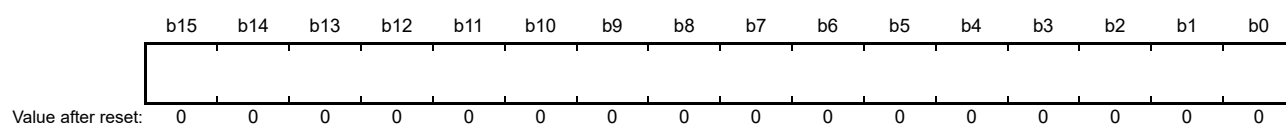
Address(es): DOC.DODIR 0008 B082h



The DODIR register is a readable and writable register that holds values for use in operations.

35.2.3 DOC Data Setting Register (DODSR)

Address(es): DOC.DODSR 0008 B084h



The DODSR register is a readable and writable register that holds values for use in comparison or the results of other operations.

In data comparison mode, store the standard value for use in comparison in this register.

In data addition or data subtraction mode, this register holds the results of operations.

35.3 Operation

35.3.1 Data Comparison Mode

Figure 35.2 shows an example of the steps involved in data comparison mode operation by the DOC.

An example of operation when DCSEL is set to 0 ('not equal to' is to be detected as the result of data comparison) is shown below.

- (1) Writing 00b to the DOCR.OMS[1:0] bits places the DOC in the data comparison mode.
- (2) Specify the standard value for comparison in the DODSR register.
- (3) Write the value for comparison in the DODIR register.
- (4) Write all values for use in comparison to the DODIR register.
- (5) If the value written to the DODIR register is not equal to the value set in the DODSR register, the DOCR.DOPCF flag becomes 1. If the DOCR.DOPCIE bit is 1, a data operation circuit interrupt is also issued.

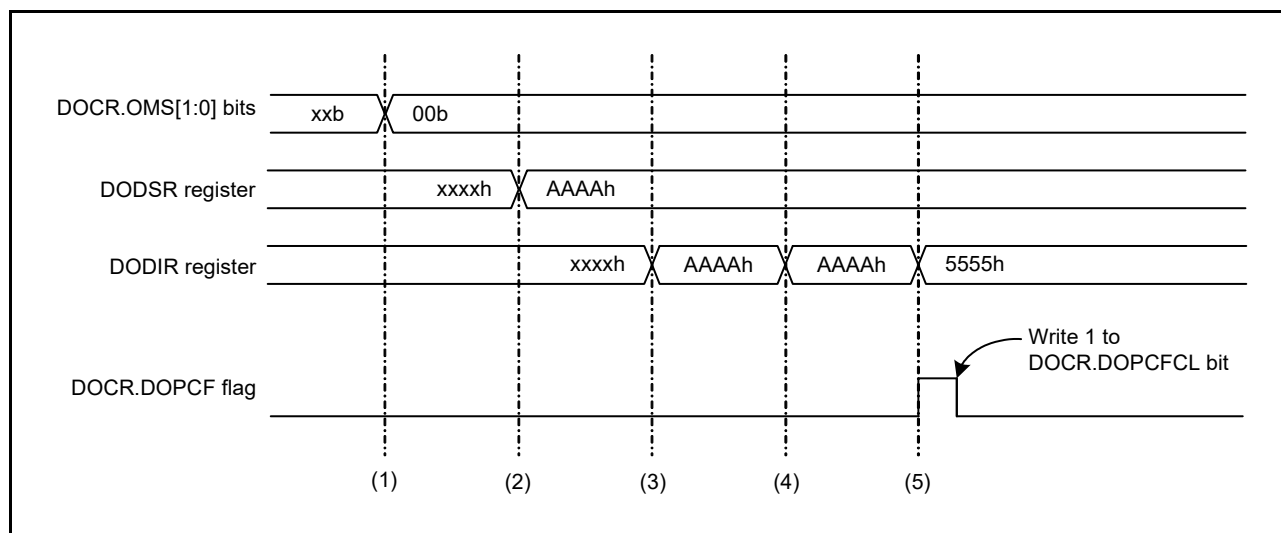


Figure 35.2 Example of Operation in Data Comparison Mode

35.3.2 Data Addition Mode

Figure 35.3 shows an example of the steps involved in data addition mode operation by the DOC.

- (1) Writing 01b to the DOCR.OMS[1:0] bits selects data addition mode.
- (2) Set the initial value in the DODSR register.
- (3) Write the value for addition in the DODIR register. The result of the operation is stored in DODSR.
- (4) Write all values for use in addition to the DODIR register.
- (5) If the result of the operation is greater than FFFFh, the DOCR.DOPCF flag becomes 1. If the DOCR.DOPCIE bit is 1, a data operation circuit interrupt is also issued.

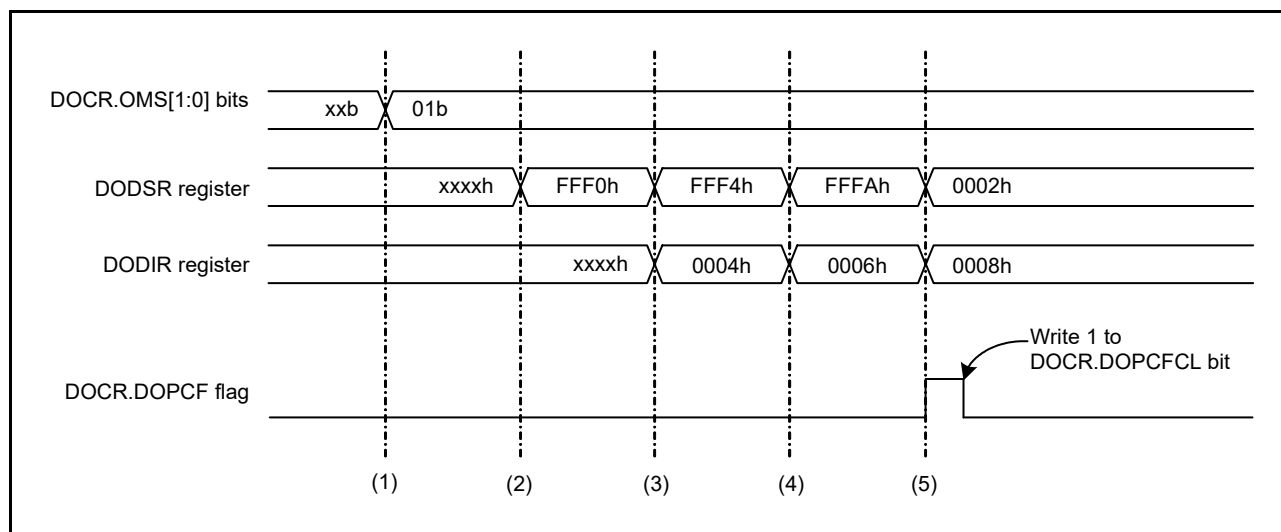


Figure 35.3 Example of Operation in Data Addition Mode

35.3.3 Data Subtraction Mode

Figure 35.4 shows an example of the steps involved in data subtraction mode operation by the DOC.

- (1) Writing 10b to the DOCR.OMS[1:0] bits selects data subtraction mode.
- (2) Set the initial value in the DODSR register.
- (3) Write the value for subtraction in the DODIR register. The result of the operation is stored in DODSR.
- (4) Write all values for use in subtraction to the DODIR register.
- (5) If the result of the operation is less than 0000h, the DOCR.DOPCF flag becomes 1. If the DOCR.DOPCIE bit is 1, a data operation circuit interrupt is also issued.

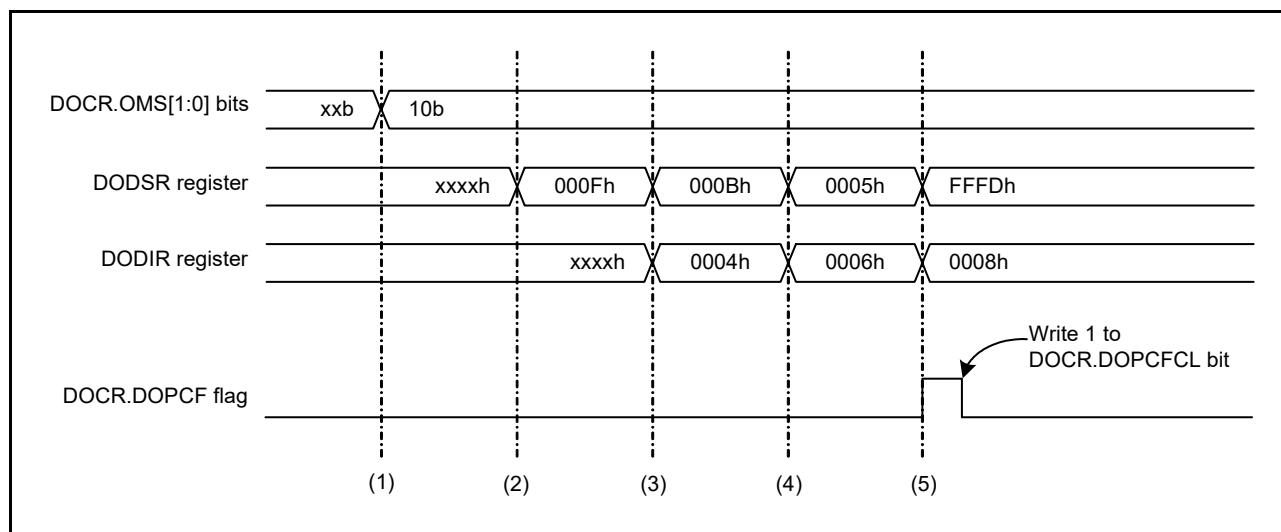


Figure 35.4 Example of Operation in Data Subtraction Mode

35.4 Interrupt Requests

The data operation circuit interrupt (DOPCI) is the interrupt request generated by the DOC. The DOCR.DOPCF flag becomes 1 when the interrupt source condition is satisfied.

Table 35.2 lists the details of the interrupt request.

Table 35.2 Interrupt Request from DOC

Interrupt Request	Data Operation Result Flag	Interrupt Generation Timing
Data operation circuit interrupt (DOPCI)	DOPCF	- The result of data comparison meets the detection condition. - The result of data addition is greater than FFFFh. - The result of data subtraction is less than 0000h.

35.5 Usage Note

35.5.1 Module Stop Function Setting

Operation of the DOC can be enabled or disabled by setting the MSTPB6 bit in module stop control register B (MSTPCRB). The DOC is initially disabled after a reset. Register access is enabled by releasing the module stop state. For details, refer to section 11, Low Power Consumption.

36. RAM

This MCU has a 12-Kbyte high-speed static RAM (RAM), which runs in the no-wait state.

36.1 Overview

Table 36.1 lists the specifications of the RAM.

Table 36.1 Specifications of RAM

Item	RAM
Capacity	12 Kbytes
Address	0000 0000h to 0000 2FFFh
Memory bus	Memory bus 1
Access	• Single-cycle access is possible for both reading and writing.*1 • Enabling or disabling of the RAM is selectable.*2
Low power consumption function	Transitions to the module stop state are possible.

Note 1. When accessing across the 8-byte boundary, the number of cycles is doubled.

Note 2. Selectable by the RAME bit in SYSCR1. For details on SYSCR1, see section 3.2.2, System Control Register 1 (SYSCR1).

36.2 Usage Notes

36.2.1 Low Power Consumption Function

Power consumption can be reduced by setting module stop control register C (MSTPCRC) to stop supply of the clock signal to the RAM.

Setting the MSTPCRC.MSTPC0 bit to 1 stops supply of the clock signal to the RAM.

Stopping supply of the clock signal places the RAM in the module stop state.

The RAM operates after a reset.

The RAM is not accessible in the module stop state.

Do not allow transitions to the module stop state while accessing to the RAM.

Access to the RAM in the module stop state is prohibited. If access is attempted, correct operation is not guaranteed.

For details on the MSTPCRC register, refer to section 11, Low Power Consumption.

36.2.2 Notes on Self-Diagnosis of the RAM

A write buffer is mounted for the RAM. When the same address is read after a write operation, data in the write buffer, rather than in the memory cell of the RAM may be read. When the RAM is self-diagnosed, confirm that the data have been written by following the procedure below so that data will not be read from the write buffer.

- (1) Write data to the address targeted for diagnosis.
- (2) Write data to an address which is at least 4 addresses away from the that in (1).
- (3) Read the data from the address in (1).

37. Flash Memory (FLASH)

This MCU has packages with 128 Kbyte flash memory (ROM) for storing code and 4-Kbyte flash memory (E2 DataFlash) for storing data.

In this section, “PCLK” is used to refer to PCLKB.

37.1 Overview

Table 37.1 lists the flash memory specifications.

Table 37.8 lists the I/O pins used in boot mode.

Table 37.1 Flash Memory Specifications

Item	Description
Memory space	- User area: Up to 128 Kbytes - Data area: 4 Kbytes - Extra area: Stores the start-up area information, access window information, and unique ID
Operating clock	- FCLK: 1 to 48 MHz (for ROM P/E mode and E2 DataFlash P/E mode) up to 48 MHz (for E2 DataFlash read mode) - HOCO clock: 24 MHz, 32 MHz, or 48 MHz (for ROM P/E mode and E2 DataFlash P/E mode)
Software commands	- The following commands are implemented: Program, blank check, block erase, and all-block erase - The following commands are implemented for programming the extra area: Start-up area information program, access window protection, and access window information program
Value after erasure	- ROM: FFh - E2 DataFlash: FFh
Interrupt	An interrupt (FRDYI) is generated upon completion of software command processing or forced stop processing.
On-board programming	Boot mode (SCI Interface)*1 - Channel 1 of the serial communications interface (SCI1) is used for asynchronous serial communication. - The user area and data area are rewritable. Boot mode (FINE interface)*1 - The FINE is used. - The user area and data area are rewritable. Self-programming in single-chip mode - The user area and data area are rewritable using the flash rewrite routine in the user program.
Off-board programming	The user area and data area are rewritable using a flash programmer compatible with this MCU.
ID code protection	- Connection with the serial programmer can be enabled or disabled using ID codes in boot mode. - Connection with the on-chip debugging emulator can be enabled or disabled using ID codes.
Start-up program protection	This function is used to safely rewrite block 0 to block 7.
Area protection	This function enables rewriting only the selected blocks in the user area and disables the other blocks during self-programming.
Background Operation (BGO)	Programs on the ROM can be executed while rewriting the E2 DataFlash.

Note 1. Refer to the manual of each serial programmer and “Renesas Flash Programmer Flash memory programming software User’s Manual” for more details.

37.2 ROM Area and Block Configuration

The ROM size of this MCU is 128 Kbytes. The ROM area is divided into blocks. A block is 2-Kbyte area. When executing the block erase command, the memory is erased by the block. Figure 37.1 shows the ROM area and block configuration.

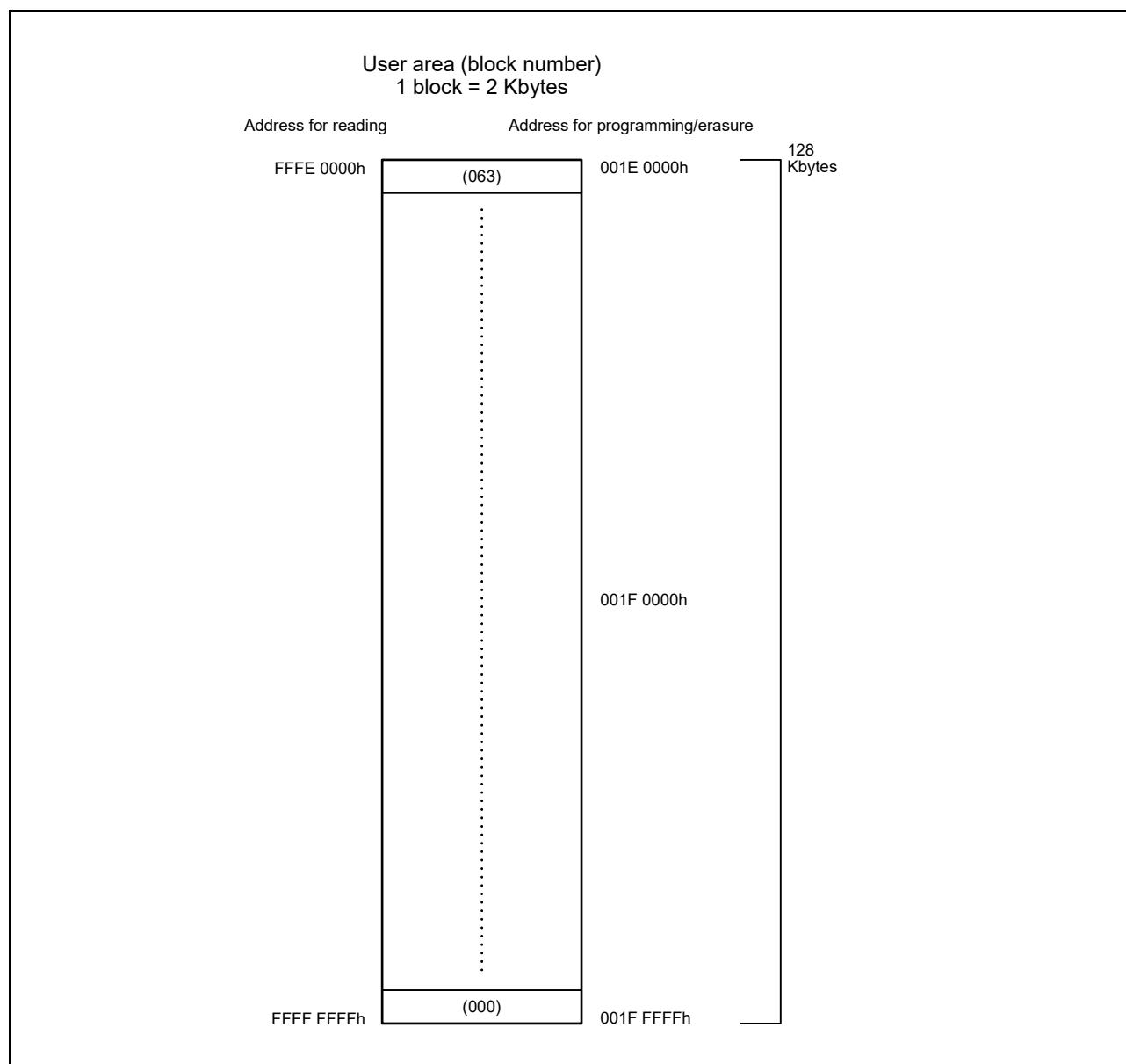


Figure 37.1 ROM Area and Block Configuration

Table 37.2 Correspondence Between ROM Capacity and Addresses for Reading

ROM Capacity	Addresses for Reading
128 Kbytes	FFFE 0000h to FFFF FFFFh

37.3 E2 DataFlash Area and Block Configuration

The E2 DataFlash is 4 Kbytes in the MCU. The E2 DataFlash is divided into blocks and erased in block units. Figure 37.2 shows the E2 DataFlash area and the block configuration.

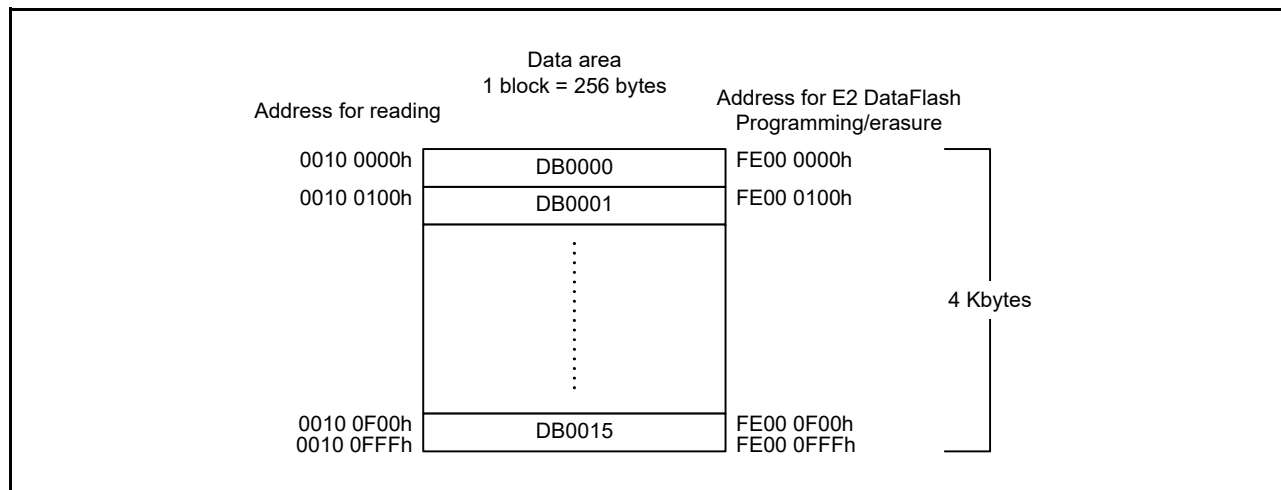


Figure 37.2 E2 DataFlash Area and Block Configuration

37.4 Register Descriptions

37.4.1 E2 DataFlash Control Register (DFLCTL)

Address(es): FLASH.DFLCTL 007F C090h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	DFLEN
0	0	0	0	0	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	DFLEN	E2 DataFlash Access Enable	0: Access to E2 DataFlash and access to the extra area in P/E mode*1 disabled 1: Access to E2 DataFlash and access to the extra area in P/E mode*1 enabled	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Start-up area information program, access window protection, and access window information program

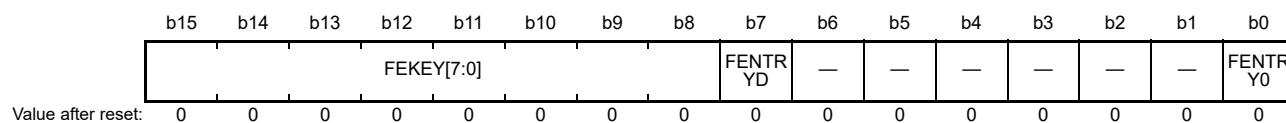
The DFLCTL register is used to enable or disable access (read, program, and erase) to the E2 DataFlash and access (start-up area information program, access window protection, and access window information program) to the extra area in P/E mode.

When reading, programming, and erasing the E2 DataFlash, set the DFLCTL.DFLEN bit to 1 and wait for the E2 DataFlash STOP recovery time (tDSTOP) to elapse before reading the E2 DataFlash and entering E2 DataFlash P/E mode. Do not read the E2 DataFlash or enter E2 DataFlash P/E mode until tDSTOP has elapsed.

Refer to section 37.7.1, Sequencer Modes for details on E2 DataFlash P/E mode. Refer to section 38, Electrical Characteristics for E2 DataFlash STOP recovery time (tDSTOP).

37.4.2 Flash P/E Mode Entry Register (FENTRYR)

Address(es): FLASH.FENTRYR 007F FFB0h



Bit	Symbol	Bit Name	Description	R/W
b0	FENTRY0	ROM P/E Mode Entry 0	0: ROM is in read mode. 1: ROM can be placed in P/E mode.	R/W
b6 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	FENTRYD	E2 DataFlash P/E Mode Entry	0: E2 DataFlash is in read mode. 1: E2 DataFlash can be placed in P/E mode.	R/W
b15 to b8	FEKEY[7:0]	Key Code	The FEKEY[7:0] bits are used to control rewriting of the FENTRYR register. When rewriting the value of the lower 8 bits, set the FEKEY[7:0] bits to AAh at the same time (write this register in 16 bits). The FEKEY[7:0] bits are read as 00h.	R/W

To rewrite the ROM or E2 DataFlash, the FENTRYD or FENTRY0 bit must be set to 1 after oscillating HOCO to place the ROM or E2 DataFlash in P/E mode.

When returning to read mode, set the FENTRYR register and confirm that its value has been rewritten before reading the ROM or E2 DataFlash.

Refer to section 37.7.1, Sequencer Modes for details on P/E mode and read mode.

FENTRY0 Bit (ROM P/E Mode Entry 0)

This bit is used to place the ROM in P/E mode.

[Setting condition]

- AA01h is written to the FENTRYR register when the FENTRYR register is 0000h.

Note: When entering ROM P/E mode, the instruction fetch address must be transferred to an area other than the ROM so that instruction fetching is not executed to the ROM. Copy necessary instruction code to the internal RAM and jump to the RAM. Note that E2 DataFlash can be rewritten by a program in the ROM.

[Clearing condition]

- AA00h is written to the FENTRYR register.

FENTRYD Bit (E2 DataFlash P/E Mode Entry)

This bit is used to place the E2 DataFlash in P/E mode.

[Setting condition]

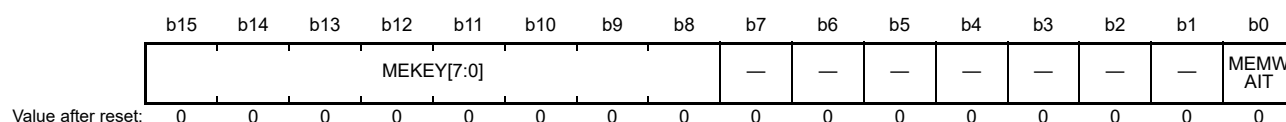
- AA80h is written to the FENTRYR register when the FENTRYR register is 0000h.

[Clearing condition]

- AA00h is written to the FENTRYR register.

37.4.3 Memory Wait Cycle Setting Register (MEMWAITR)

Address(es): FLASH.MEMWAITR 007F FFC0h



Bit	Symbol	Bit Name	Description	R/W
b0	MEMWAIT	Memory Wait Cycle Setting	0: No wait states 1: Wait states	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	MEKEY[7:0]	Key Code	These bits are used to control rewriting of the MEMWAITR register. When rewriting the value of the lower 8 bits, set these bits to AAh at the same time (write this register in 16-bit units). These bits are read as 00h.	R/W

The MEMWAITR register is used to control the wait cycle of the ROM.

MEMWAIT Bit (Memory Wait Cycle Setting)

This bit is used to specify the wait cycle of the ROM.

When setting the system clock (ICLK) frequency higher than 32 MHz, set the MEMWAIT bit to 1. Set the MEMWAIT bit to 0 in middle-speed operating mode.

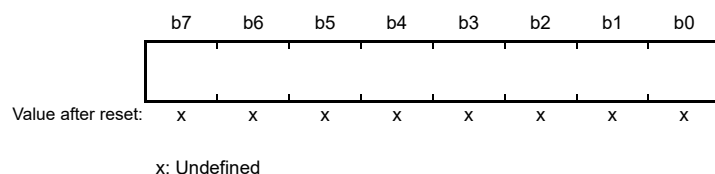
Change the value of the MEMWAIT bit when the MCU is in high-speed operating mode and the ICLK frequency is 32 MHz or less.

Table 37.3 Restrictions on Setting of the MEMWAIT bit

MEMWAIT bit	High-Speed Operating Mode		Middle-Speed Operating Mode
	ICLK > 32 MHz	ICLK ≤ 32 MHz	
0	Setting prohibited	Setting possible	Setting possible
1	Setting possible	Setting possible	Setting prohibited

37.4.4 Protection Unlock Register (FPR)

Address(es): FLASH.FPR 007F C180h



This write-only register is used to protect the FPMCR register from being rewritten inadvertently when the CPU runs out of control. Writing to the FPMCR register is enabled only when the following procedure is used to access the register.

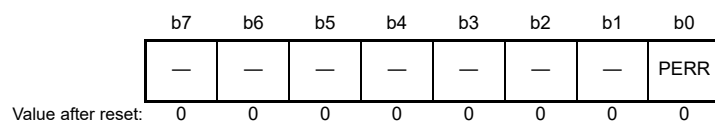
Procedure to unlock protection

- (1) Write A5h to the FPR register.
- (2) Write a set value to the FPMCR register.
- (3) Write the inverted set value to the FPMCR register.
- (4) Write a set value to the FPMCR register again.

When a procedure other than the above is used to write data, the FPSR.PERR flag is set to 1.

37.4.5 Protection Unlock Status Register (FPSR)

Address(es): FLASH.FPSR 007F C184h



Bit	Symbol	Bit Name	Description	R/W
b0	PERR	Protect Error Flag	0: No error 1: An error occurs.	R
b7 to b1	—	Reserved	These bits are read as 0.	R

PERR Flag (Protect Error Flag)

When the FPMCR register is not accessed as described in the procedure to unlock protection, data is not written to the register and this flag is set to 1.

[Setting condition]

- The FPMCR register is not accessed as described in the procedure to unlock protection.

[Clearing condition]

- The FPMCR register is accessed according to the procedure to unlock protection described in section 37.4.4, Protection Unlock Register (FPR).

37.4.6 Flash P/E Mode Control Register (FPMCR)

Address(es): FLASH.FPMCR 007F C100h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	FMS1	RPDIS	—	FMS0	—
0	0	0	0	1	0	0	0

Value after reset:

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	FMS0	Flash Operating Mode Select 0	FMS1 FMS0 0 0: ROM/E2 DataFlash read mode 0 1: ROM P/E mode 1 0: E2 DataFlash P/E mode 1 1: Setting prohibited.	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	RPDIS	ROM P/E Disable	0: ROM programming/erasure enabled 1: ROM programming/erasure disabled	R/W
b4	FMS1	Flash Operating Mode Select 1	See the FMS0 bit.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The FPMCR register is used to set the operating mode of the flash memory.

This register is protected. Set its value using the procedure to unlock protection. For details, refer to [section 37.4.4, Protection Unlock Register \(FPR\)](#).

When entering ROM P/E mode or during ROM P/E mode, an instruction must be executed on the RAM.

FMS0 and FMS1 Bits (Flash Operating Mode Select 0 and Flash Operating Mode Select 1)

These bits are used to set the operating mode of the flash memory.

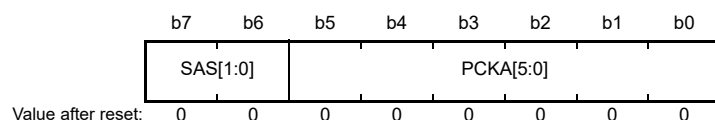
- Transition from read mode to ROM P/E mode
Set the FMS1 bit = 0, the FMS0 bit = 1, and the RPDIS bit = 0.
- Transition from ROM P/E mode to read mode
Set the FMS1 bit = 0, the FMS0 bit = 0, and the RPDIS bit = 1.
Wait for ROM mode transition wait time (tMS, refer to [section 38, Electrical Characteristics](#)).
- Transition from read mode to E2 DataFlash P/E mode
Set the FMS1 bit = 1, the FMS0 bit = 0, and the RPDIS bit = 0.
- Transition from E2 DataFlash P/E mode to read mode
Set the FMS1 bit = 0, the FMS0 bit = 0, and the RPDIS bit = 1.
Wait for ROM mode transition wait time (tMS, refer to [section 38, Electrical Characteristics](#)).

RPDIS Bit (ROM P/E Disable)

This bit is used to disable the execution of ROM programming/erasure with software.

37.4.7 Flash Initial Setting Register (FISR)

Address(es): FLASH.FISR 007F C1D8h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PCKA[5:0]	Peripheral Clock Notification	These bits are used to set the frequency of the FlashIF clock (FCLK).	R/W
b7, b6	SAS[1:0]	Start-Up Area Select	b7 b6 0 x: The start-up area is selected according to the start-up area settings of the extra area. 1 0: The start-up area is switched to the default area temporarily. 1 1: The start-up area is switched to the alternate area temporarily.	R/W

x: Don't care

Data can be written to the FISR register in ROM P/E mode or E2 DataFlash P/E mode.

PCKA[5:0] Bits (Peripheral Clock Notification)

These bits are used to set the frequency of the FlashIF clock (FCLK) when programming/erasing the ROM/E2 DataFlash.

Set the FCLK frequency in the PCKA[5:0] bits before programming/erasure. Do not change the frequency during programming/erasure of the ROM/E2 DataFlash.

- When FCLK is higher than 4 MHz
Set a rounded-up value for a non-integer frequency.
For example, set 32 MHz (PCKA[5:0] bits = 011111b) when the frequency is 31.5 MHz.
- When FCLK is 4 MHz or lower
Do not use a non-integer frequency.
Use the FCLK at a frequency of 1, 2, 3, or 4 MHz.

Note: When the PCKA[5:0] bits are set to a frequency different from the FCLK, the data in the ROM/E2 DataFlash may be damaged.

Table 37.4 Example of FlashIF Clock Frequency Settings

FlashIF Clock Frequency (MHz)	PCKA[5:0] Bit Setting	FlashIF Clock Frequency (MHz)	PCKA[5:0] Bit Setting	FlashIF Clock Frequency (MHz)	PCKA[5:0] Bit Setting
48	100111b	46	100110b	44	100101b
42	100100b	40	100011b	38	100010b
36	100001b	34	100000b	32	011111b
31	011110b	30	011101b	29	011100b
28	011011b	27	011010b	26	011001b
25	011000b	24	010111b	23	010110b
22	010101b	21	010100b	20	010011b
19	010010b	18	010001b	17	010000b
16	001111b	15	001110b	14	001101b
13	001100b	12	001011b	11	001010b
10	001001b	9	001000b	8	000111b
7	000110b	6	000101b	5	000100b
4	000011b	3	000010b	2	000001b
1	000000b	—	—	—	—

SAS[1:0] Bits (Start-Up Area Select)

These bits are used to select the start-up area. To change the start-up area, the following three methods can be used.

- When selecting the start-up area according to the start-up area settings of the extra area
With the SAS[1:0] bits set to 00b or 01b, the start-up area is selected according to the start-up area settings of the extra area. The settings are enabled after a reset is released.
- When switching the start-up area to the default area temporarily
When 10b is written to the SAS[1:0] bits, the start-up area is switched to the default area immediately after data is written to the register, regardless of the start-up area settings of the extra area.
When a reset is generated after this, the area is selected according to the start-up area settings of the extra area.
- When switching the start-up area to the alternative area temporarily
When 11b is written to the SAS[1:0] bits, the start-up area is switched to the alternative area, regardless of the start-up area settings of the extra area.
When a reset is generated after this, the area is selected according to the start-up area settings of the extra area.

37.4.8 Flash Reset Register (FRESETR)

Address(es): FLASH.FRESETR 007F C124h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	FRESE T
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FRESET	Flash Reset	0: Flash control circuit reset is released. 1: Flash control circuit is reset.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

FRESET Bit (Flash Reset)

When this bit is set to 1, registers FASR, FSARH, FSARL, FEARH, FEARL, FWB0, FWB1, FWB2, FWB3, FCR, and FEXCR are reset. Also, the values of registers FEAMH and FEAML are undefined. Do not access these registers during a reset. To release the reset, set this bit to 0.

Do not write to this register while executing a software command or rewriting the extra area.

37.4.9 Flash Area Select Register (FASR)

Address(es): FLASH.FASR 007F C104h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	EXS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	EXS	Extra Area Select	0: User area or data area 1: Extra area	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Data can be written to the FASR register in ROM P/E mode or E2 DataFlash P/E mode.

This register is initialized by a reset or setting the FRESETR.FRESET bit to 1.

Data cannot be written to this register while the FRESETR.FRESET bit is 1.

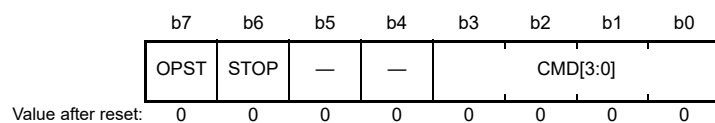
EXS Bit (Extra Area Select)

Set this bit to 1 before issuing a software command (start-up area information program, access window protection, or access window information program) for the extra area. Set this bit to 0 before issuing a software command (program, blank check, block erase, or all-block erase) for the user area.

After issuing a software command, do not change the value until changing it for issuing the next software command.

37.4.10 Flash Control Register (FCR)

Address(es): FLASH.FCR 007F C114h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	CMD[3:0]	Software Command Setting*1	b3 b0 0 0 0 1: Program 0 0 1 1: Blank check 0 1 0 0: Block erase 0 1 1 0: All-block erase Settings other than above are prohibited.*2	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	STOP	Forced Processing Stop	When this bit is set to 1, the processing being executed can be forcibly stopped.	R/W
b7	OPST	Processing Start	0: Processing ends. 1: Processing starts.	R/W

Note 1. These bits are rewritable only when both the OPST bit and the FSTATR1.FRDY flag are set to 0.

Note 2. This does not include writing 00h to the FCR register to set the OPST bit to 0.

Data can be written to the FCR register when in ROM P/E mode and the ROM can be programmed/erased or in E2 DataFlash P/E mode.

This register is initialized by a reset or setting the FRESETR.FRESET bit to 1. Data cannot be written to this register while the FRESETR.FRESET bit is 1.

Note that this register cannot be initialized by the FRESETR.FRESET bit while a software command is being executed.

CMD[3:0] Bits (Software Command Setting)

These bits are used to set a software command (program, blank check, block erase, or all-block erase).

The function of each command is described below.

- **Program**
Write the value set in registers FWB0, FWB1, FWB2, and FWB3 to the address set in registers FSARH and FSARL.
- **Blank check**
Check whether there is data in the area from the address set in registers FSARH and FSARL to the address set in registers FEARH and FEARL. Confirm that data is not programmed in the area. This command does not guarantee whether the area remains erased.
- **Block erase**
Erase consecutive areas specified in the flash memory by the blocks. Set the beginning address of the block in registers FSARH and FSARL and the end address in registers FEARH and FEARL.
- **All-block erase**
Erase all blocks in the ROM or E2 DataFlash.
All-block erase requires less time to erase the memory compared to block erase. When erasing the whole of the ROM area, set the beginning address of the ROM area in registers FSARH and FSARL, and the end address in registers FEARH and FEARL. Table 37.5 lists the setting address for all-block erase.

Table 37.5 Setting Address for All-Block Erase

Target	Memory Size	FSARH/FSARL	FEARH/FEARL
ROM	128 Kbytes	001E 0000h	001F FFF8h
E2 DataFlash	4 Kbytes	FE00 0000h	FE00 0FFFh

STOP Bit (Forced Processing Stop)

This bit is used to forcibly stop the processing (blank check, block erase, or all-block erase) being executed.

After setting this bit to 1, wait until the FSTATR1.FRDY flag is 1 (processing completed) before setting the OPST bit to 0.

OPST Bit (Processing Start)

This bit is used to execute the command set in the CMD[3:0] bits.

This bit is not set to 0 again even when the processing is completed. Confirm that the FSTATR1.FRDY flag is 1 (processing completed) before setting the OPST bit to 0 again. After that, confirm that the FRDY flag is 0 before executing the next processing.

37.4.11 Flash Extra Area Control Register (FEXCR)

Address(es): FLASH.FEXCR 007F C1DCh

b7	b6	b5	b4	b3	b2	b1	b0
OPST	—	—	—	—	CMD[2:0]		
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	CMD[2:0]	Software Command Setting*1	b2 b0 0 0 1: Start-up area information program/access window protection*2 0 1 0: Access window information program Settings other than above are prohibited.*3	R/W
b6 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	OPST	Processing Start	0: Processing ends. 1: Processing starts.	R/W

Note 1. These bits are rewritable only when both the OPST bit and the FSTATR1.EXRDY flag are set to 0.

Note 2. Once protected, the protection cannot be removed.

Note 3. This does not include writing 00h to the FEXCR register to set the OPST bit to 0.

Data can be written to the FEXCR register when in ROM P/E mode and the ROM can be programmed/erased.

This register is initialized by a reset or setting the FRESETR.FRESET bit to 1. Data cannot be written to this register while the FRESETR.FRESET bit is 1.

Note that this register cannot be initialized by the FRESETR.FRESET bit while a software command is being executed.

CMD[2:0] Bits (Software Command Setting)

These bits are used to set a software command (start-up area information program, access window protection, or access window information program).

The details of each command are described below.

- Start-up area information program

This command is used to switch the start-up area used for start-up program protection.

When setting the start-up area to the default area, set the FWB1 and FWB0 registers to FFFFh, and execute this command.

When setting the start-up area to the alternative area, set the FWB1 register to FFFFh, set the FWB0 register to FEFFh, and execute this command.

When the FWB1 and FWB0 registers are set to values other than the above, do not execute the start-up area information program.

- Access window protection

This command is used to protect the settings for the access window.

When the access window protection command is executed after setting the FWB1 register to FFFFh, bit 14 of the FWB0 register to 0, bit 8 to the same value as the FSCMR.SASMF flag, and the other bits to 1, the settings for the access window are protected. Once protected, the protection cannot be removed.

- Access window information program

This command is used to set the access window used for area protection.

Set the access window in block units.

Specify the access window start address, which is the beginning address of the access window in the FWB0 register, specify the access window end address, which is the next address of the last address of the access window in the FWB1 register, and issue this command. Set bit 21 to bit 11 of the address for programming/erasure in each register. If the same value is set as the start address and end address, all areas can be accessed. Do not set the start address to

a value larger than the value of the end address.

OPST Bit (Processing Start)

This bit is used to execute the command set in the CMD[2:0] bits.

This bit is not set to 0 again even when the processing is completed. Confirm that the FSTATR1.EXRDY flag is 1 (processing completed) before setting the OPST bit to 0 again. After that, confirm that the FSTATR1.EXRDY flag is 0 before executing the next processing.

Writing to the extra area is started by writing 1 to the OPST bit. Do not write to the CMD[2:0] bits while a software command is being executed.

37.4.12 Flash Processing Start Address Register H (FSARH)

Address(es): FLASH.FSARH 007F C110h



The FSARH register is used to set the target processing address or the start address of the target processing range in the flash memory when a software command is executed.

Set bit 31 to bit 25 and bit 20 to bit 16 of the flash memory address for programming/erasure in this register.

Data can be written to this register in ROM P/E mode or E2 DataFlash P/E mode.

This register is initialized by a reset or setting the FRESETR.FRESET bit to 1. Data cannot be written to this register while the FRESETR.FRESET bit is 1.

If this register is read while executing a software command set by the FEXCR register, an undefined value is read.

Refer to Figure 37.1 and Figure 37.2 for details on the addresses of the flash memory.

37.4.13 Flash Processing Start Address Register L (FSARL)

Address(es): FLASH.FSARL 007F C108h



The FSARL register is used to set the target processing address or the start address of the target processing range in the flash memory when a software command is executed.

Set bit 15 to bit 0 of the flash memory address for programming/erasure in this register.

When the target is the ROM, set bit 2 to bit 0 to 000b.

Data can be written to this register in ROM P/E mode or E2 DataFlash P/E mode.

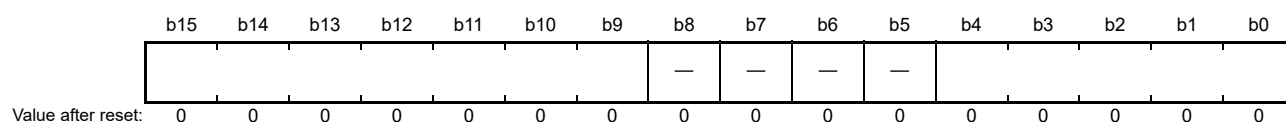
This register is initialized by a reset or setting the FRESETR.FRESET bit to 1. Data cannot be written to this register while the FRESETR.FRESET bit is 1.

This register is incremented by 8h if the ROM is specified and 1h if the E2 DataFlash is specified after a program command is executed. Therefore, it is not necessary to set the target address to be written to this register when executing a program command sequentially.

If this register is read while executing a software command set by the FEXCR register, an undefined value is read.
Refer to Figure 37.1 and Figure 37.2 for details on the addresses of the flash memory.

37.4.14 Flash Processing End Address Register H (FEARH)

Address(es): FLASH.FEARH 007F C120h



The FEARH register is used to set the end address of the target processing range in the flash memory when a software command is executed.

Set bit 31 to bit 25 and bit 20 to bit 16 of the flash memory address for programming/erasure in this register.

Data can be written to this register in ROM P/E mode or E2 DataFlash P/E mode.

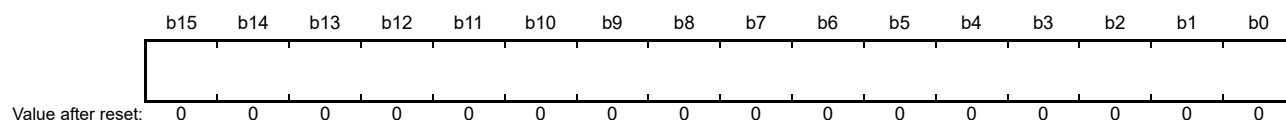
This register is initialized by a reset or setting the FRESETR.FRESET bit to 1. Data cannot be written to this register while the FRESETR.FRESET bit is 1.

If this register is read while executing a software command set by the FEXCR register, an undefined value is read.

Refer to Figure 37.1 and Figure 37.2 for details on the addresses of the flash memory.

37.4.15 Flash Processing End Address Register L (FEARL)

Address(es): FLASH.FEARL 007F C118h



The FEARH register is used to set the end address of the target range for processing when a software command is executed.

Set bit 15 to bit 0 of the flash memory address for programming/erasure in this register.

When the target is the ROM, set bit 2 to bit 0 to 000b.

Data can be written to this register in ROM P/E mode or E2 DataFlash P/E mode.

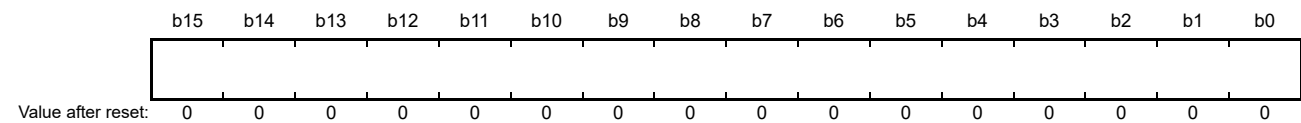
This register is initialized by a reset or setting the FRESETR.FRESET bit to 1. Data cannot be written to this register while the FRESETR.FRESET bit is 1.

If this register is read while executing a software command set by the FEXCR register, an undefined value is read.

Refer to Figure 37.1 and Figure 37.2 for details on the addresses of the flash memory.

37.4.16 Flash Write Buffer Register n (FWBn) (n = 0 to 3)

Address(es): FLASH.FWB0 007F C130h, FLASH.FWB1 007F C138h, FLASH.FWB2 007F C140h, FLASH.FWB3 007F C144h



This register is used to set the data for programming the ROM, E2 DataFlash, or extra area. The data can be written in ROM P/E mode or E2 DataFlash P/E mode.

This register is initialized by a reset or setting the FRESETR.FRESET bit to 1. Data cannot be written to this register while the FRESETR.FRESET bit is 1.

The read value of this register is undefined while executing a software command set by the FCR register or the FEXCR register.

When programming the extra area, set the 4-byte data for programming in registers FWB0 and FWB1.

When programming the E2 DataFlash, set the data for programming in the lower 8 bits in the FWB0 register.

When programming the ROM, set the 8-byte data for programming in registers FWB0 to FWB3. Figure 37.3 shows the relationship between the addresses indicated by registers FSARH and FSARL and the data set in the FWBn register.

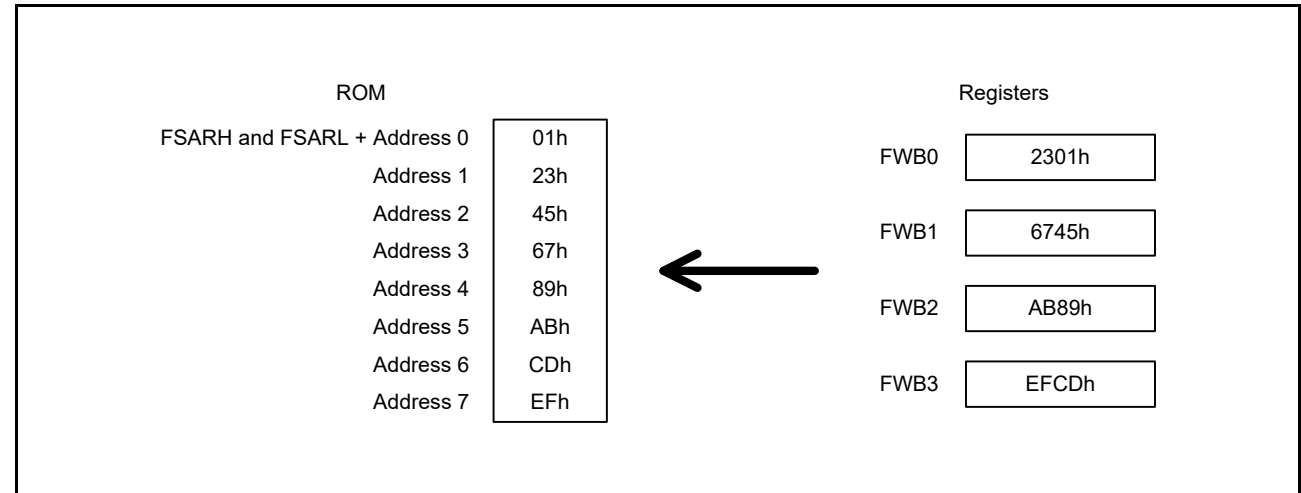


Figure 37.3 FWBn Register Setting Values and Data Allocation in the ROM

37.4.17 Flash Status Register 0 (FSTATR0)

Address(es): FLASH.FSTATR0 007F C128h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	EILGLE RR	ILGLER R	BCERR	—	PRGER R	ERERR
Value after reset:	x	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ERERR	Erase Error Flag	0: Erasure terminates normally. 1: An error occurs during erasure.	R
b1	PRGERR	Program Error Flag	0: Programming terminates normally. 1: An error occurs during programming.	R
b2	—	Reserved	The read value is undefined.	R
b3	BCERR	Blank Check Error Flag	0: Blank checking terminates normally. 1: An error occurs during blank checking.	R
b4	ILGLERR	Illegal Command Error Flag	0: No illegal software command or illegal access is detected. 1: An illegal command or illegal access is detected.	R
b5	EILGLERR	Extra Area Illegal Command Error Flag	0: No illegal command or illegal access to the extra area is detected. 1: An illegal command or illegal access to the extra area is detected.	R
b7, b6	—	Reserved	The read value is undefined.	R

This register is a status register used to confirm the result of executing a software command. Each error flag is set to 0 when the next software command is executed.

ERERR Flag (Erase Error Flag)

This flag indicates the result of the erase processing for the ROM/E2 DataFlash.

[Setting condition]

- An error occurs during erasure.

[Clearing condition]

- The next software command is executed.

The value read from this flag is undefined when the FCR.STOP bit is set to 1 (processing is forcibly stopped) during erasure.

PRGERR Flag (Program Error Flag)

This flag indicates the result of the program processing for the ROM/E2 DataFlash.

[Setting condition]

- An error occurs during programming.

[Clearing condition]

- The next software command is executed.

BCERR Flag (Blank Check Error Flag)

This flag indicates the result of the blank check processing for the ROM/E2 DataFlash.

[Setting condition]

- An error occurs during blank checking.

[Clearing condition]

- The next software command is executed.

The value read from this flag is undefined when the FCR.STOP bit is set to 1 (processing is forcibly stopped) during blank checking.

ILGLERR Flag (Illegal Command Error Flag)

This flag indicates the result of executing a software command.

[Setting conditions]

- Programming/erasure is executed to an area other than the access window range.
- A blank check or block erase command is executed when the set value of registers FSARH and FSARL is larger than the set value of registers FEARH and FEARL.
- Program and block erase commands are executed when the FASR.EXS bit is 1.
- An all-block erase command is executed while the access window is set.
- An all-block erase command is executed without setting registers FSARH and FSARL and registers FEARH and FEARL properly.
- The E2 DataFlash address is set in registers FSARH and FSARL and a software command is executed when the ROM is in P/E mode.
- The ROM address is set in registers FSARH and FSARL and a software command is executed when the E2 DataFlash is in P/E mode.
- The ROM and E2 DataFlash are set to P/E mode and a software command is executed.

[Clearing condition]

- The next software command is executed.

EILGLERR Flag (Extra Area Illegal Command Error Flag)

This flag indicates the result of executing a software command for the extra area.

[Setting condition]

- A software command for the extra area is executed when the FASR.EXS bit is 0.

[Clearing condition]

- The next software command is executed.

37.4.18 Flash Status Register 1 (FSTATR1)

Address(es): FLASH.FSTATR1 007F C12Ch

b7	b6	b5	b4	b3	b2	b1	b0
EXRDY	FRDY	—	—	—	—	—	—
Value after reset: 0	0	0	0	0	1	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0.	R
b2	—	Reserved	This bit is read as 1.	R
b5 to b3	—	Reserved	These bits are read as 0.	R
b6	FRDY	Flash Ready Flag	0: Other than below 1: 00h can be written to the FCR register (processing to complete the software command).	R
b7	EXRDY	Extra Area Ready Flag	0: Other than below 1: 00h can be written to the FEXCR register (processing to complete the software command).	R

This register is a status register used to confirm the result of executing a software command. Each flag is set to 0 when the next software command is executed.

FRDY Flag (Flash Ready Flag)

This flag is used to confirm whether a software command is executed.

This flag becomes 1 when processing of the executed software command or the forced stop processing is completed, and this flag becomes 0 when setting the FCR.OPST bit to 0.

Also, an interrupt (FRDYI) is generated when this flag becomes 1.

EXRDY Flag (Extra Area Ready Flag)

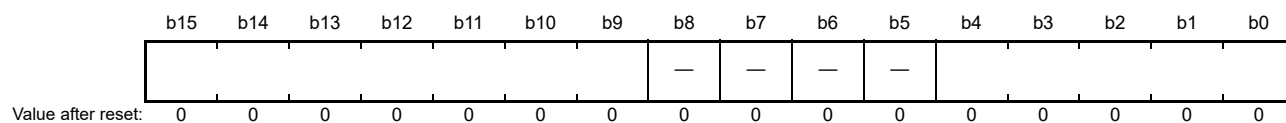
This flag is used to confirm whether a software command for the extra area is executed.

This flag is set to 1 when processing of the executed software command is completed, and 0 when the FEXCR.OPST bit is set to 0.

Also, an interrupt (FRDYI) is generated when this flag becomes 1.

37.4.19 Flash Error Address Monitor Register H (FEAMH)

Address(es): FLASH.FEAMH 007F C1E8h



This register is used to check the address where the error has occurred if an error occurs during processing of a software command. This register stores bit 31 to bit 25 and bit 20 to bit 16 of the address where the error has occurred for the program command or blank check command, or it stores bit 31 to bit 25 and bit 20 to bit 16 of the beginning address of the area where the error has occurred for the block erase command or all-block erase command.

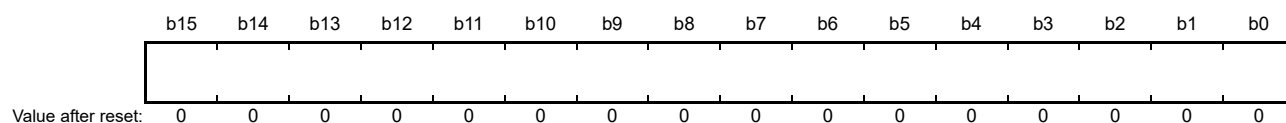
Since this register value becomes undefined if setting the FRESETR.FRESET bit to 1, read the value before error processing.

If the software command terminates normally, this register stores bit 31 to bit 25 and bit 20 to bit 16 of the end address at execution of the command.

Refer to Figure 37.1 and Figure 37.2 for details on the addresses of the flash memory.

37.4.20 Flash Error Address Monitor Register L (FEAML)

Address(es): FLASH.FEAML 007F C1E0h



This register is used to check the address where the error has occurred if an error occurs during processing of a software command. This register stores bit 15 to bit 0 of the address where the error has occurred for the program command or blank check command, or it stores bit 15 to bit 0 of the beginning address of the area where the error has occurred for the block erase command or all-block erase command.

Since this register value becomes undefined if setting the FRESETR.FRESET bit to 1, read the value before error processing.

When the software command is normally completed, this register stores bit 15 to bit 0 of the last address at execution of the command.

When executing a software command for the ROM, lower 3 bits become 000b.

Refer to Figure 37.1 and Figure 37.2 for details on the addresses of the flash memory.

37.4.21 Flash Start-Up Setting Monitor Register (FSCMR)

Address(es): FLASH.FSCMR 007F C1C0h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	AWPR	—	—	—	—	—	SASMF	—	—	—	—	—	—	—	—
Value after reset:	0	Value set by user*1	1	1	0	1	1	Value set by user*2	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0.	R
b8	SASMF	Start-Up Area Setting Monitor Flag	0: Setting to start up using the alternative area 1: Setting to start up using the default area	R
b10, b9	—	Reserved	These bits are read as 1. Writing to these bits has no effect.	R
b11	—	Reserved	This bit is read as 0. Writing to this bit has no effect.	R
b13, b12	—	Reserved	These bits are read as 1. Writing to these bits has no effect.	R
b14	AWPR	Access Window Protection Flag	0: The settings for the access window cannot be rewritten. 1: The settings for the access window can be rewritten.	R
b15	—	Reserved	This bit is read as 0. Writing to this bit has no effect.	R

Note 1. The value of the blank product is 1. It is set to the same value set in bit 14 in the FWB0 register after the access window protection command is executed.

Note 2. The value of the blank product is 1. It is set to the same value set in bit 8 in the FWB0 register after the start-up area information program command is executed.

SASMF Flag (Start-Up Area Setting Monitor Flag)

This flag is used to confirm the settings of the start-up area.

When this flag is 0, the user program is set to start up using the alternative area.

When this flag is 1, the user program is set to start up using the default area.

AWPR Flag (Access Window Protection Flag)

This flag is used to confirm whether the settings of the access window are protected or not.

When this flag is 0, the access window start and end addresses cannot be rewritten.

When this flag is 1, the access window start and end addresses can be rewritten.

37.4.22 Flash Access Window Start Address Monitor Register (FAWSMR)

Address(es): FLASH.FAWSMR 007F C1C8h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—											
Value after reset:	0	0	0	0	0											

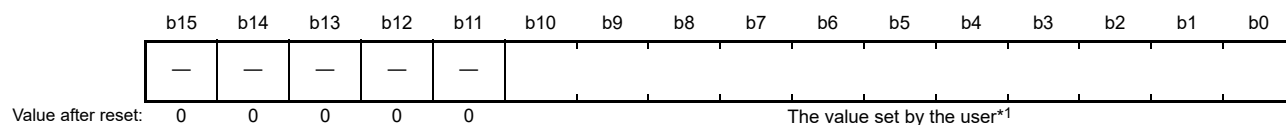
The value set by the user*1

Note 1. The value of the blank product is 1. It is set to the same value set in bit 10 to bit 0 the FWB0 register after the access window information program command is executed.

This register is used to confirm the set value of the access window start address used for area protection.

37.4.23 Flash Access Window End Address Monitor Register (FAWEMR)

Address(es): FLASH.FAWEMR 007F C1D0h

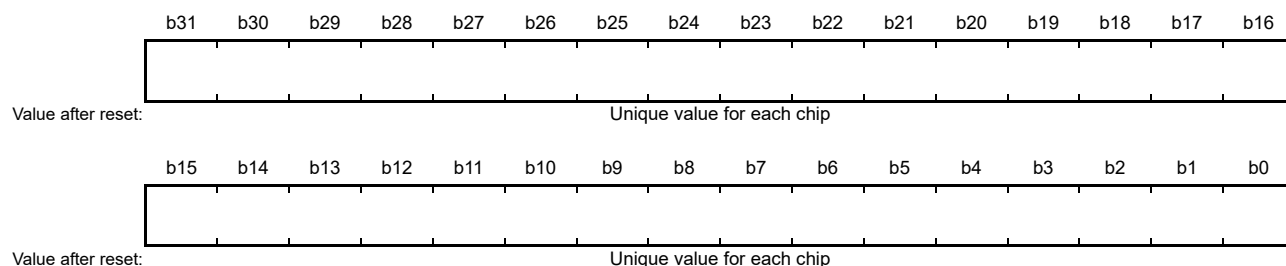


Note 1. The value of the blank product is 1. It is set to the same value set in bit 10 to bit 0 in the FWB1 register after the access window information program command is executed.

This register is used to confirm the set value of the access window end address used for area protection.

37.4.24 Unique ID Register n (UIDRn) (n = 0 to 3)

Address(es): FLASH.UIDR0 007F C350h, FLASH.UIDR1 007F C354h, FLASH.UIDR2 007F C358h, FLASH.UIDR3 007F C35Ch



The UIDRn register stores a 16-byte ID code (unique ID) for identifying the individual MCU.

The unique ID is stored in the extra area of the flash memory and cannot be rewritten by the user.

37.5 Start-Up Program Protection

When rewriting the start-up program*¹ by self-programming, if the rewrite operation is interrupted due to temporary blackout, the start-up program may not be successfully programmed and the user program may not start properly.

This problem can be avoided by rewriting the start-up program without erasing the existing start-up program using the start-up program protection. This function is available in products with a 32-Kbyte or larger ROM.

Figure 37.4 shows the overview of the start-up program protection. In this figure, the default area indicates block 0 to block 7, and the alternate area indicates block 8 to block 15.

Note 1. Program to perform operation to start the user program. It includes the fixed vector table.

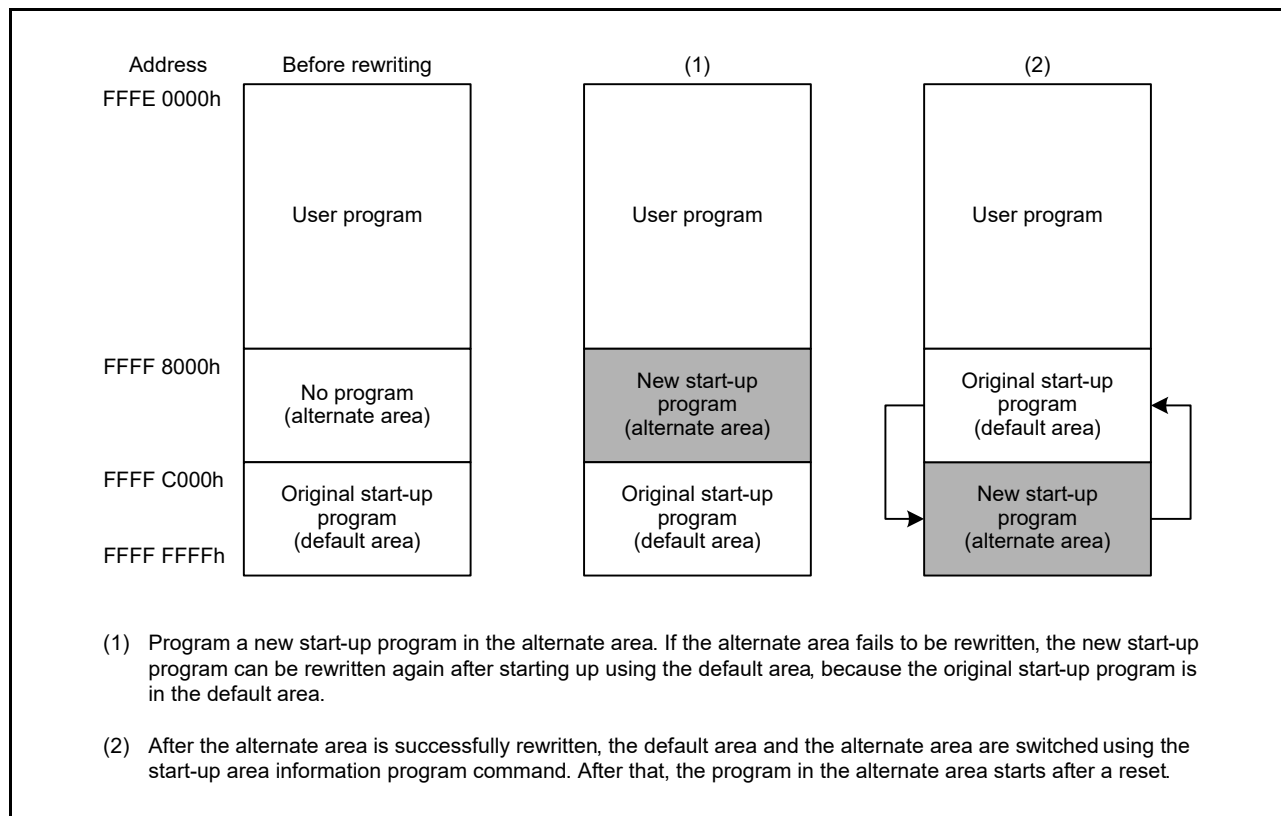


Figure 37.4 Overview of the Start-Up Program Protection

37.6 Area Protection

Area protection enables rewriting only the selected blocks (access window) in the user area and disables rewriting the other blocks during self-programming. The access window cannot be set in the data area.

Specify the start address and end address to set the access window. While the access window can be set in boot mode or by self-programming, area protection is enabled only during self-programming in single-chip mode.

Figure 37.5 shows the overview of the area protection.

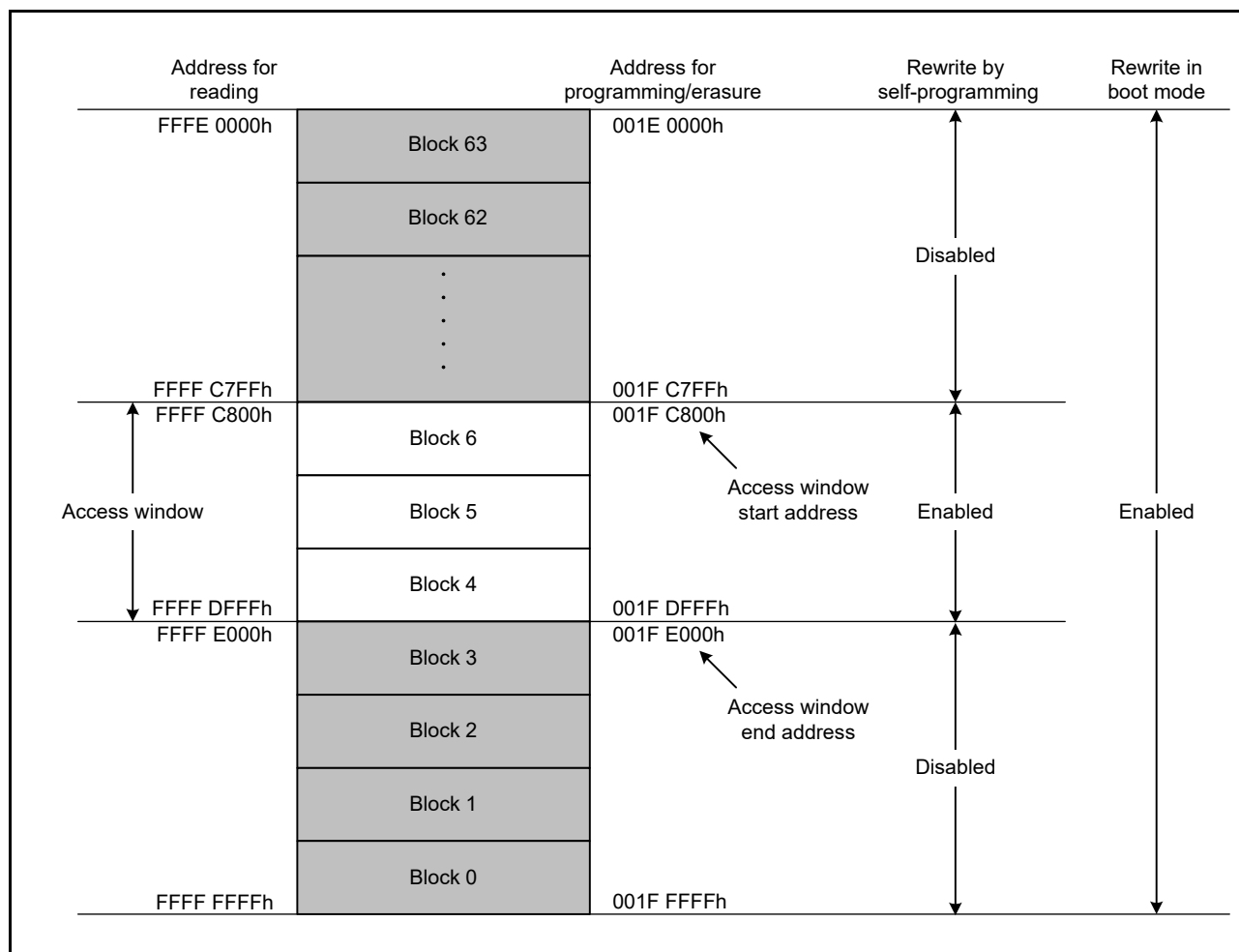


Figure 37.5 Area Protection Overview (When Blocks 4 to 6 are Set as the Access Window in Products with 128-Kbyte ROM)

37.7 Programming and Erasure

The ROM and E2 DataFlash can be programmed and erased by changing the mode of the dedicated sequencer for programming and erasure, and by issuing commands for programming and erasure.

The mode transitions and commands required to program or erase the ROM and E2 DataFlash are described below. The descriptions apply in common to boot mode and single-chip mode.

37.7.1 Sequencer Modes

The sequencer has four modes. Transitions between modes are caused by writing to the DFLCTL and FENTRYR registers and setting the FPMCR register. Figure 37.6 is a diagram of mode transitions of the flash memory.

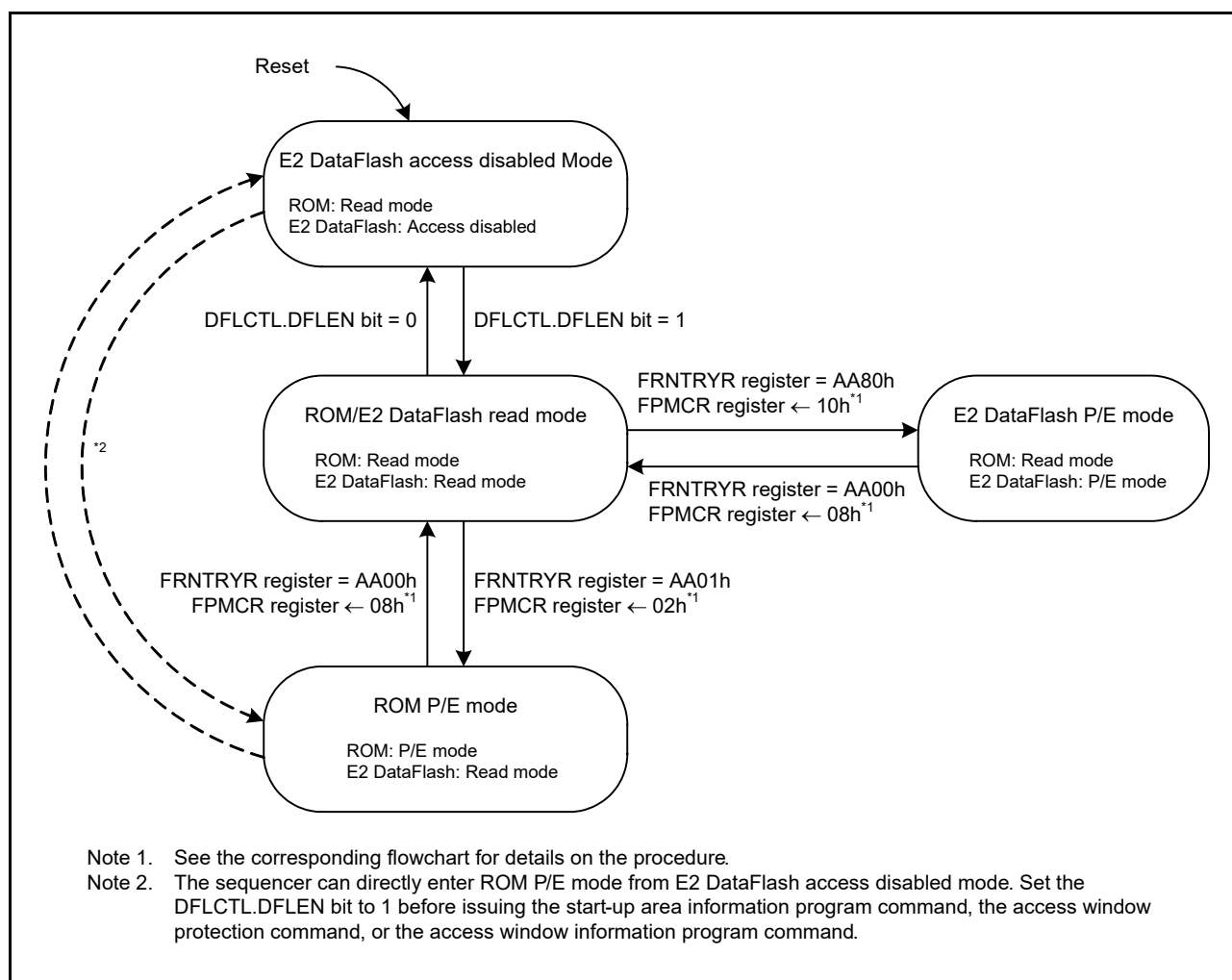


Figure 37.6 Mode Transitions of the Flash Memory

37.7.1.1 E2 DataFlash Access Disabled Mode

In E2 DataFlash access disabled mode, access to the E2 DataFlash is disabled. After a reset, the sequencer enters this mode.

When setting the DFLCTL.DFLEN bit to 1, the E2 DataFlash is placed in read mode.

37.7.1.2 Read Mode

Read mode is for high-speed reading of the ROM/E2 DataFlash. Reading from a ROM address for reading can be accomplished in one ICLK clock.

(1) ROM/E2 DataFlash Read Mode

In this mode, both the ROM and E2 DataFlash are in read mode. The sequencer enters this mode from P/E mode when setting the FPMCR register to 08h, setting the FENTRYR.FENTRYD bit to 0, and setting the FENTRYR.FENTRY0 bit to 0.

37.7.1.3 P/E Modes

The P/E mode is for programming and erasure of the ROM/E2 DataFlash.

(1) ROM P/E Mode

In this mode, the ROM is in P/E mode, and the E2 DataFlash is in read mode. The sequencer enters this mode when setting the FENTRYR.FENTRYD to 0, setting the FENTRYR.FENTRY0 bit to 1, and setting the FPMCR register 02h.

(2) E2 DataFlash P/E Mode

In this mode, the ROM is in read mode, and the E2 DataFlash is in P/E mode. The sequencer enters this mode when the setting the FENTRYR.FENTRYD to 1, setting the FENTRYR.FENTRY0 bit to 0, and setting the FPMCR register 10h.

37.7.2 Mode Transitions

37.7.2.1 Transition from E2 DataFlash Access Disable Mode to Read Mode

Reading of the E2 DataFlash requires switching from E2 DataFlash access disabled mode to ROM/E2 DataFlash read mode.

Set the DFLCTL.DFLEN bit to 1 to switch to ROM/E2 DataFlash read mode.

Figure 37.7 shows the procedure for transition from E2 DataFlash access disabled mode to ROM/E2 DataFlash read mode.

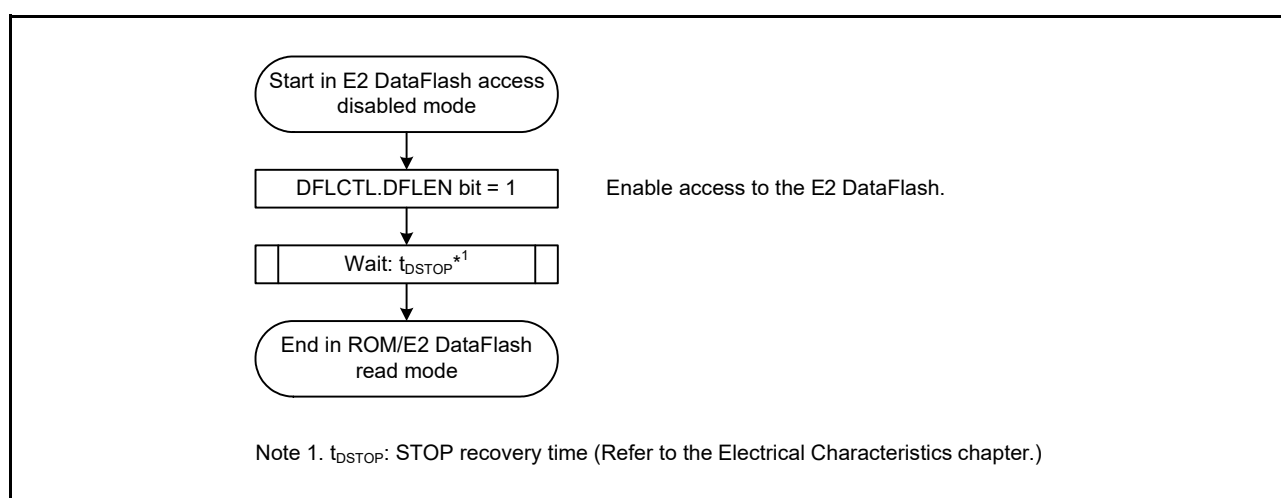


Figure 37.7 Procedure for Transition from E2 DataFlash Access Disabled Mode to ROM/E2 DataFlash Read Mode

37.7.2.2 Transition from Read Mode to P/E Mode

Switching to ROM P/E mode is required before executing a software command for the ROM.

Figure 37.8 shows the procedure for transition from ROM/E2 DataFlash read mode to ROM P/E mode. Figure 37.9 shows the procedure for transition from ROM/E2 DataFlash read mode to E2 DataFlash P/E mode.

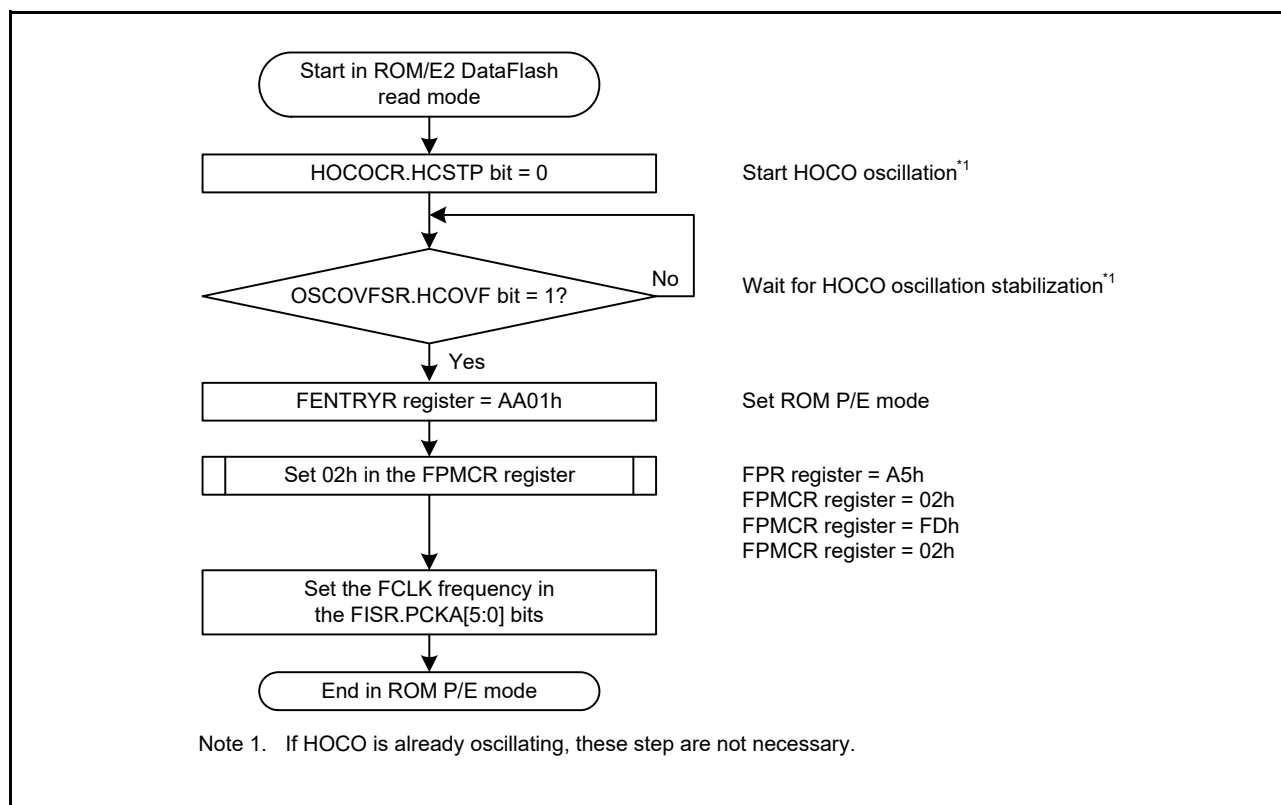


Figure 37.8 Procedure for Transition from ROM/E2 DataFlash Read Mode to ROM P/E Mode

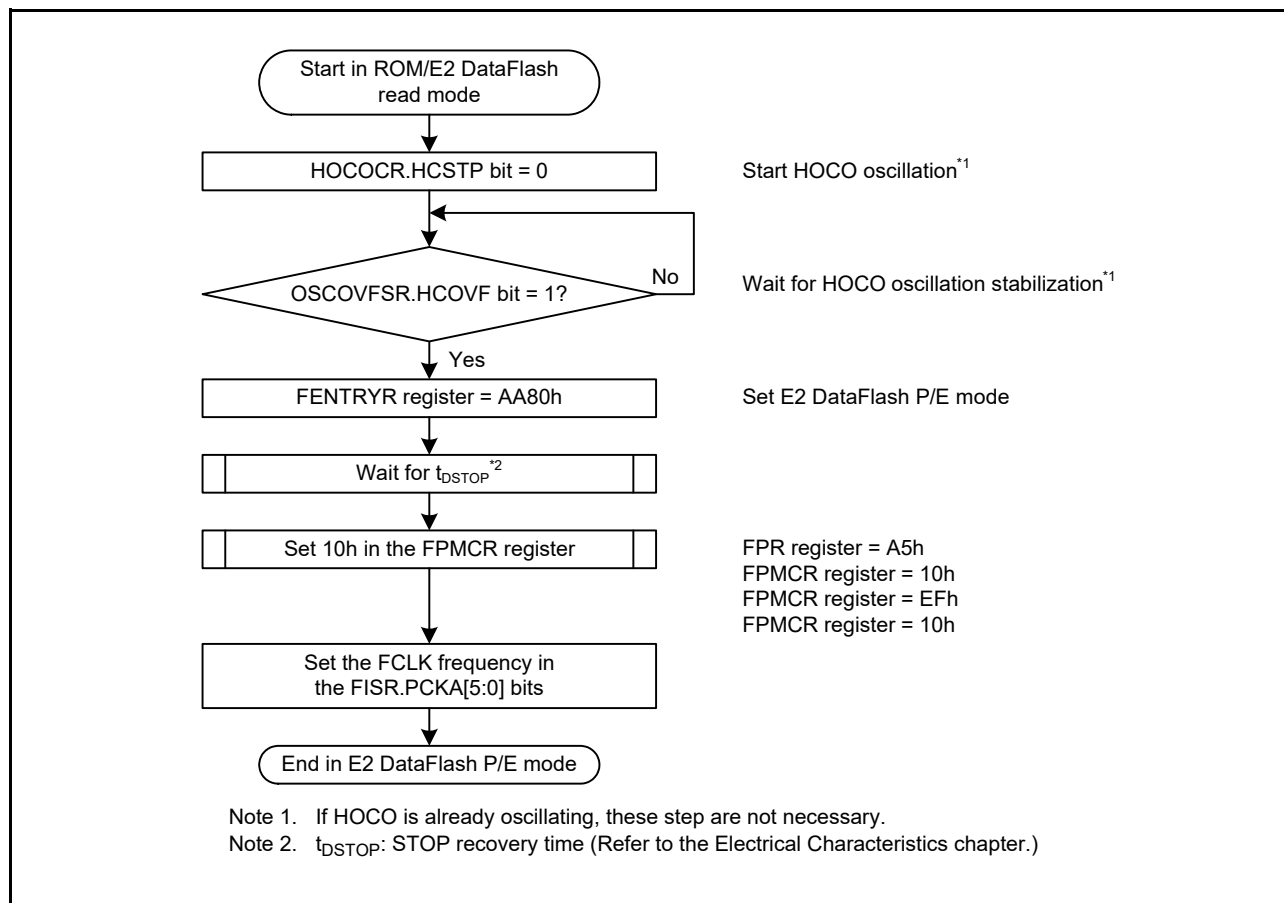


Figure 37.9 Procedure for Transition from ROM/E2 DataFlash Read Mode to E2 DataFlash P/E Mode

37.7.2.3 Transition from P/E Mode to Read Mode

High-speed reading of the ROM requires switching to ROM/E2 DataFlash read mode.

Figure 37.10 shows the procedure for transition from ROM P/E mode to ROM/E2 DataFlash read mode. Figure 37.11 shows the procedure for transition from E2 DataFlash P/E mode to ROM/E2 DataFlash read mode.

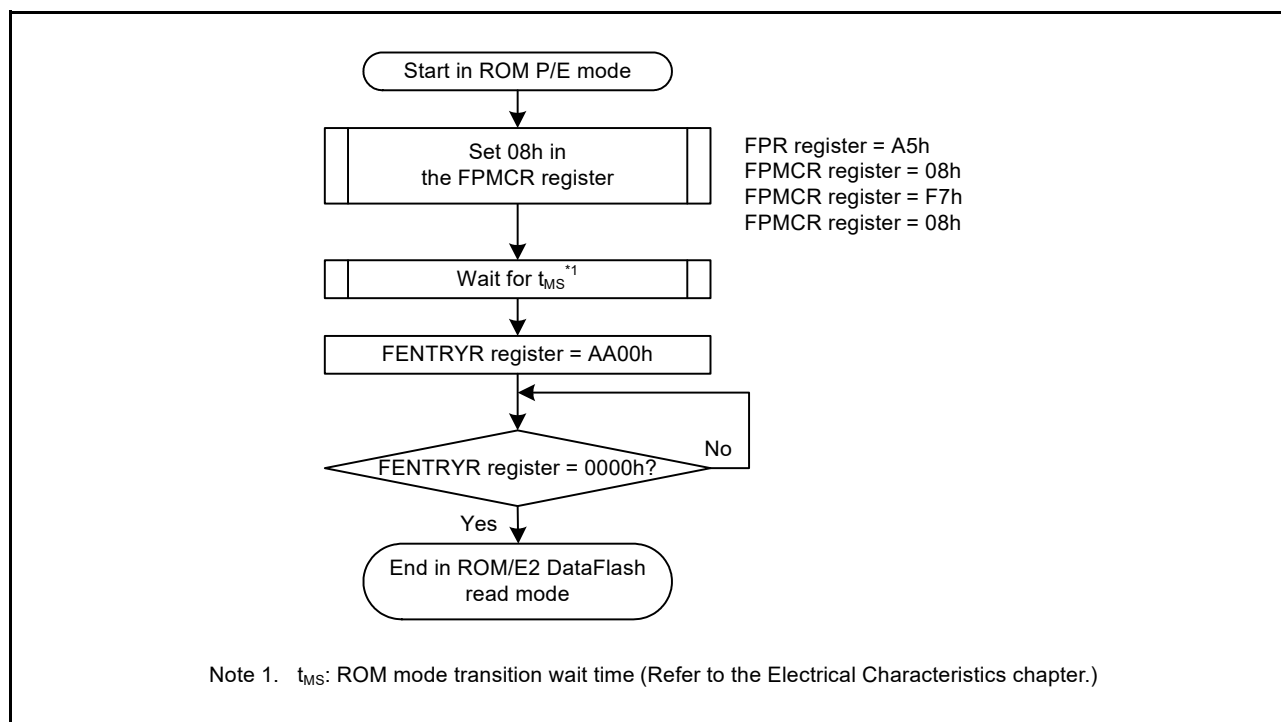


Figure 37.10 Procedure for Transition from ROM P/E Mode to ROM/E2 DataFlash Read Mode

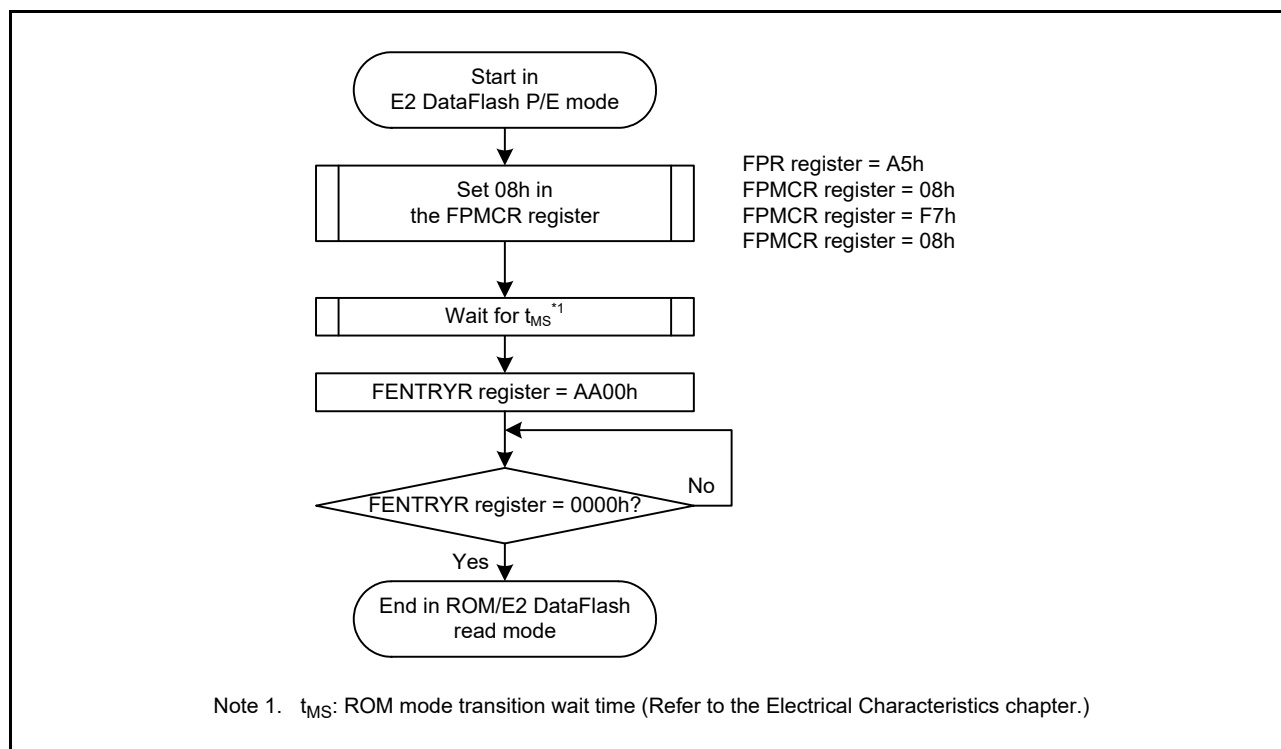


Figure 37.11 Procedure for Transition from E2 DataFlash P/E Mode to ROM/E2 DataFlash Read Mode

37.7.3 Software Commands

Software commands consist of commands for programming and erasure and commands for programming start-up program area information and access window information. Table 37.6 lists the software commands for use with the flash memory.

Table 37.6 Software Commands

Command	Function
Program	• ROM programming (8 bytes) • E2 DataFlash programming (1 byte)
Block erase	ROM/E2 DataFlash erasure
All-block erase	Erase of all blocks in the ROM/E2 DataFlash
Blank check	Check whether the specified area is blank. Confirm that data is not programmed in the area. This command does not guarantee whether the area remains erased.
Start-up area information program	Rewrite the start-up area switching information used for start-up program protection.
Access window protection	Protect the settings for the access window used for area protection from being rewritten.
Access window information program	Set the access window used for area protection.

37.7.4 Software Command Usage

This section describes how to use each software command, using flowcharts.

37.7.4.1 Program

Figure 37.12 and Figure 37.13 show the procedure to issue the program command.

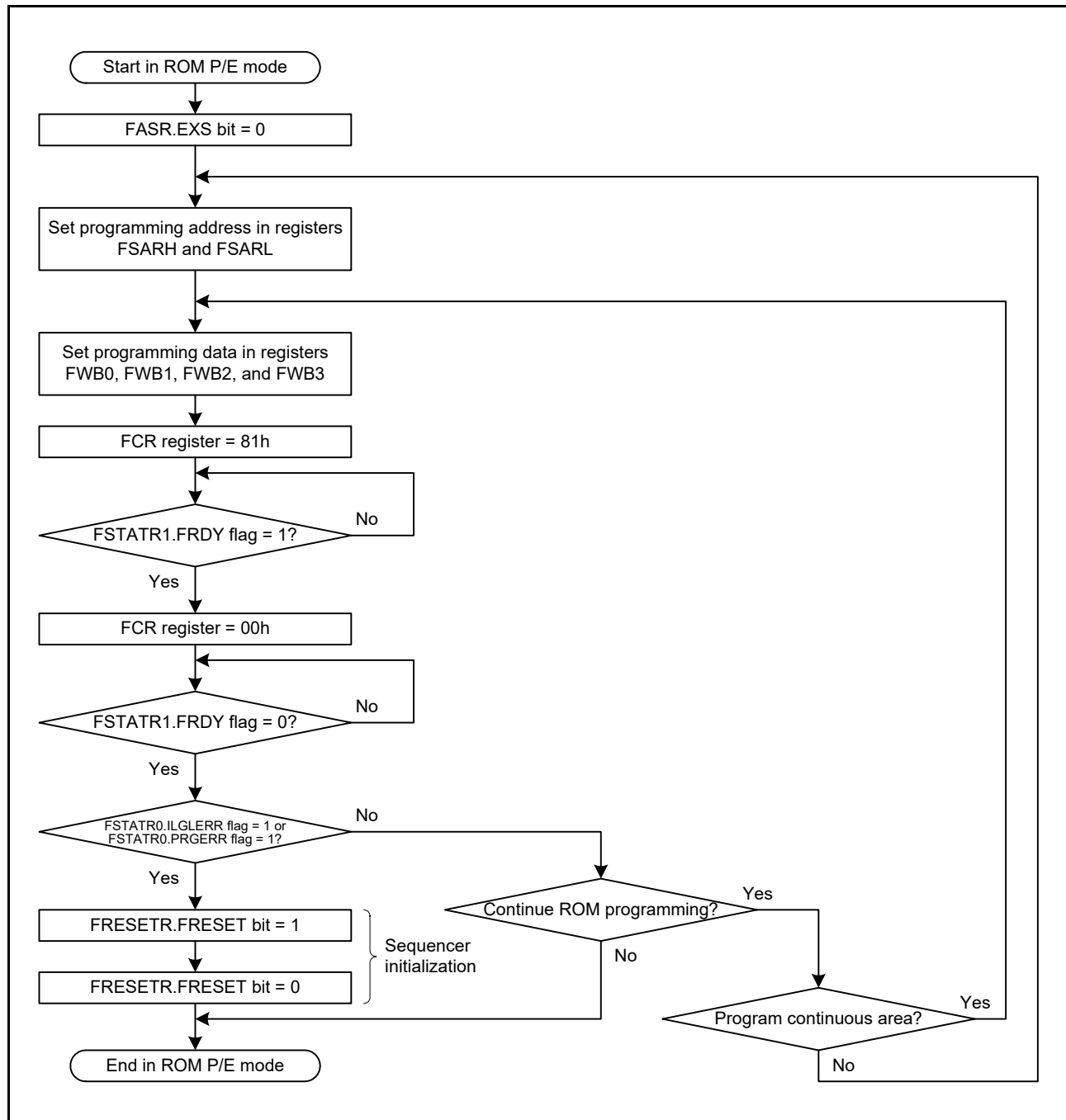


Figure 37.12 Procedure to Issue the Program Command for the ROM

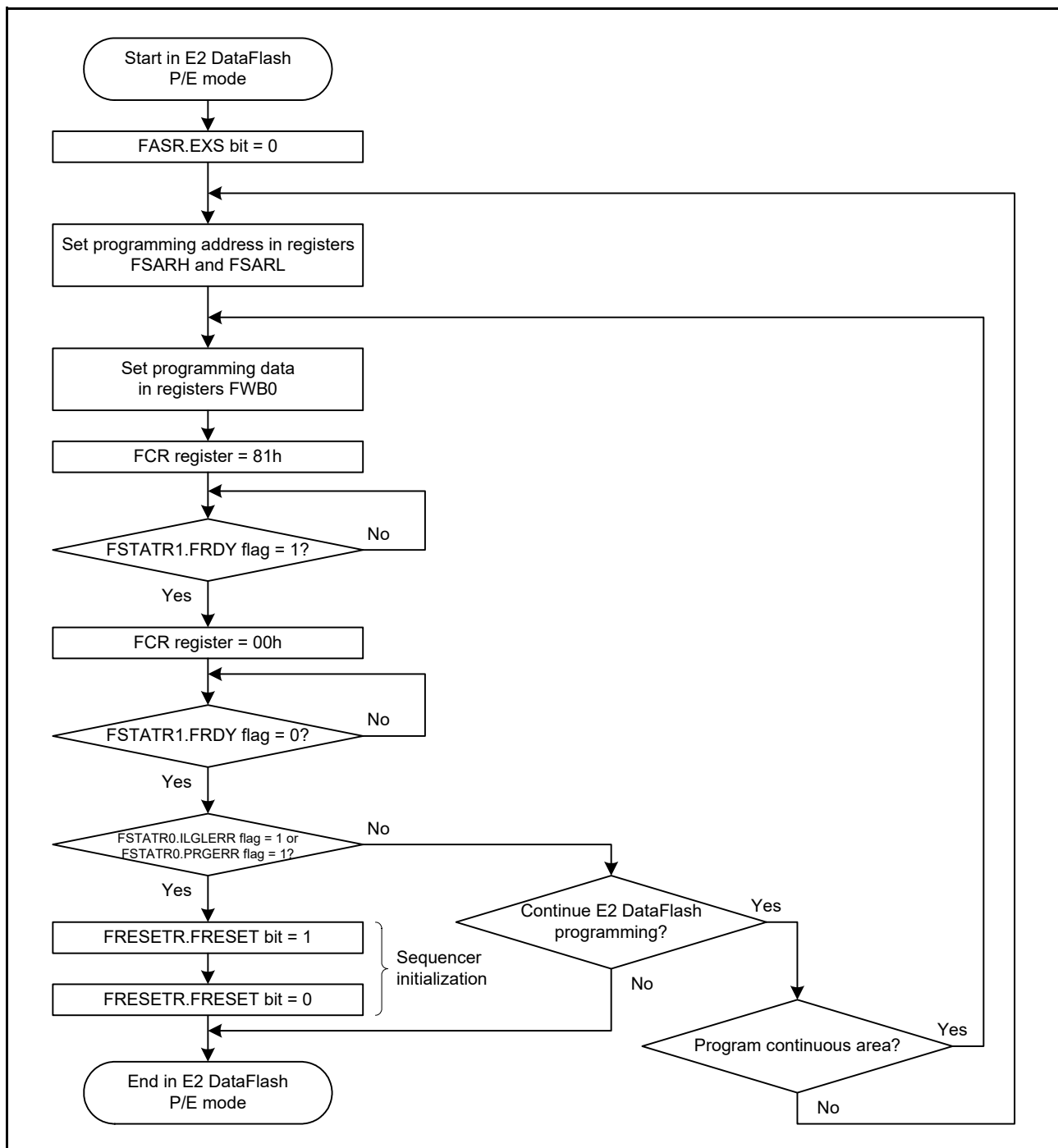


Figure 37.13 Procedure to Issue the Program Command for the E2 DataFlash

37.7.4.2 Block Erase

Figure 37.14 and Figure 37.15 show the procedure to issue the block erase command.

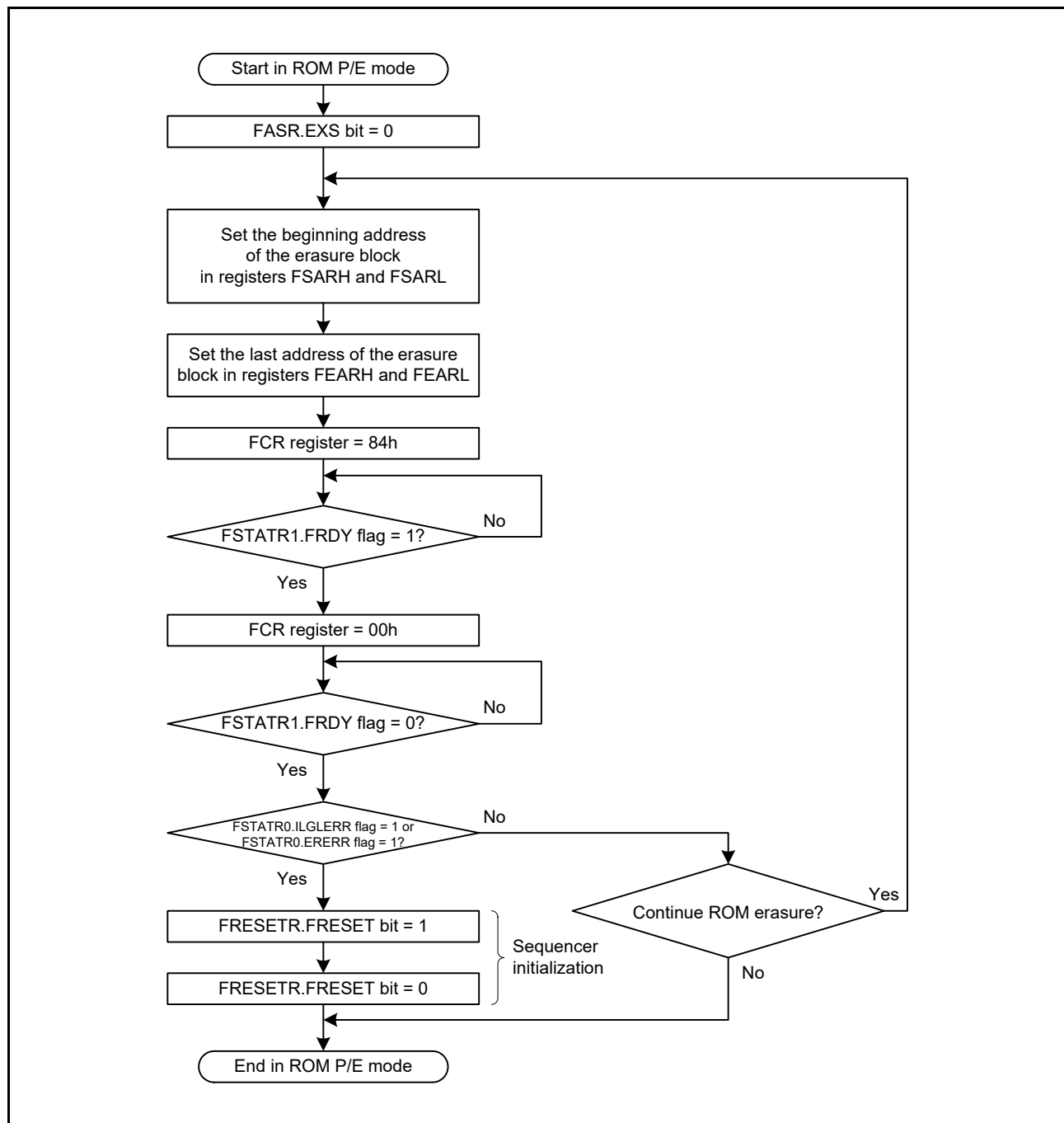


Figure 37.14 Procedure to Issue the Block Erase Command for the ROM

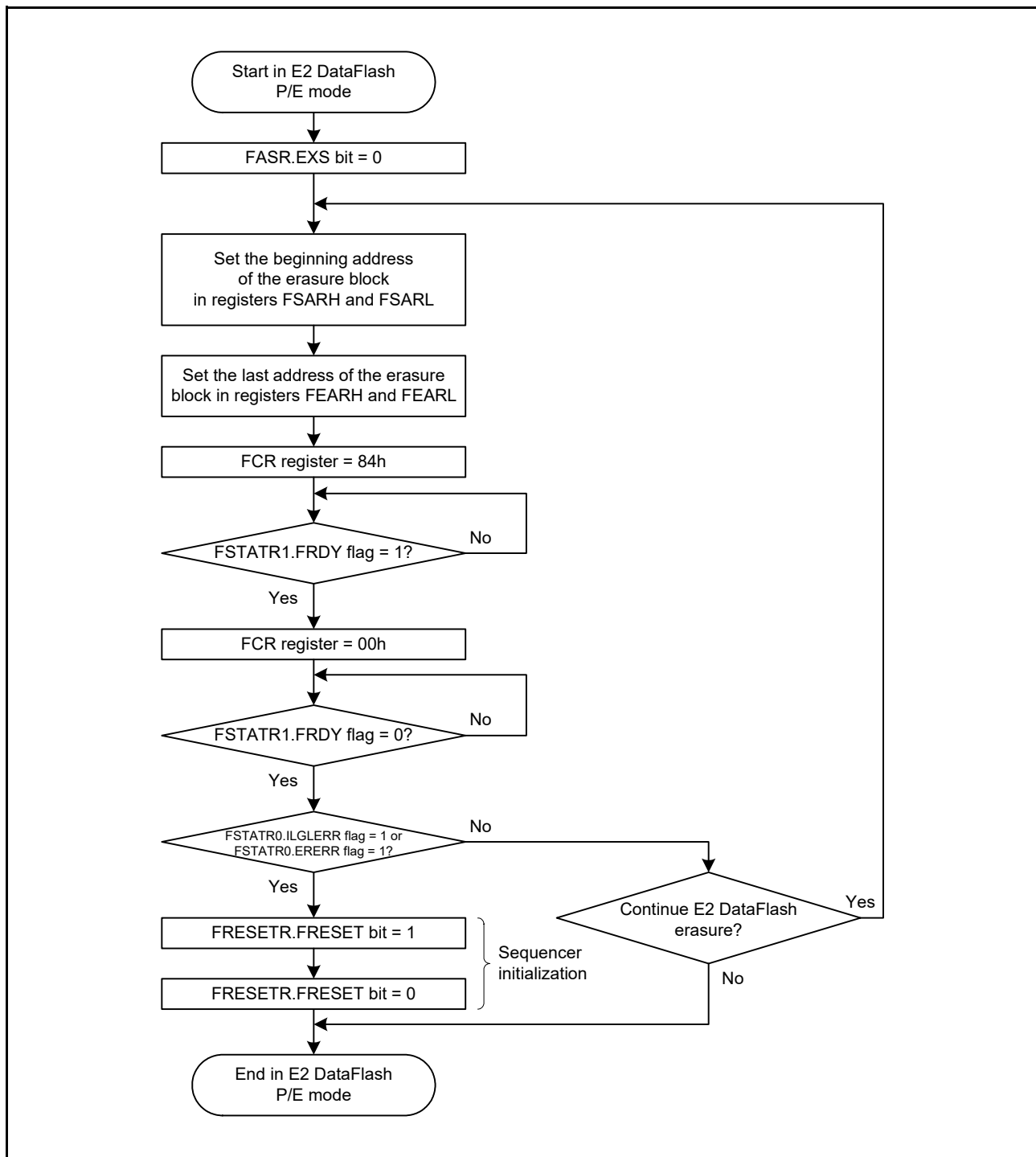


Figure 37.15 Procedure to Issue the Block Erase Command for the E2 DataFlash

37.7.4.3 All-Block Erase

Figure 37.16 and Figure 37.17 show the procedure to issue the all-block erase command.

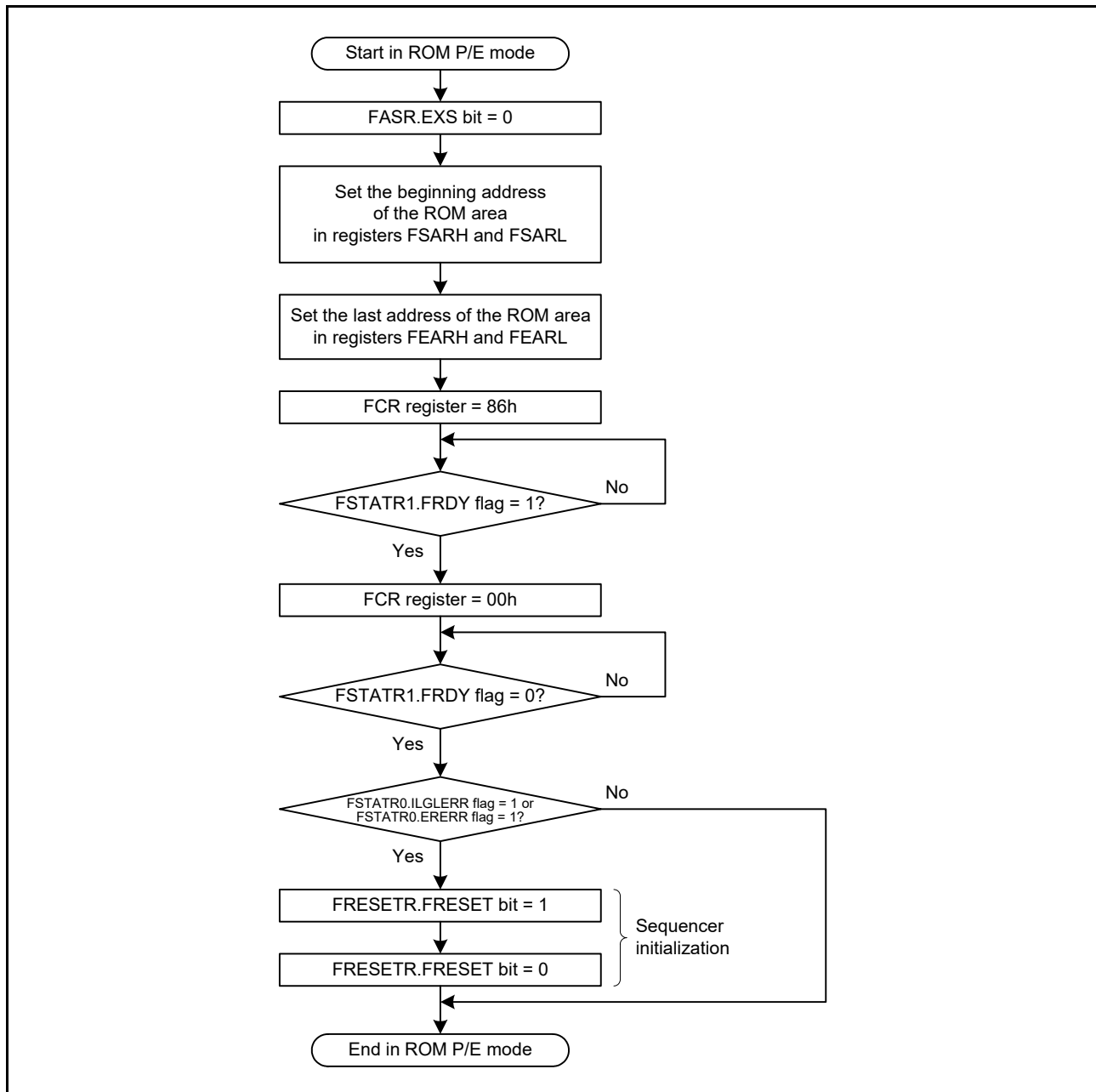


Figure 37.16 Procedure to Issue the All-Block Erase Command for the ROM

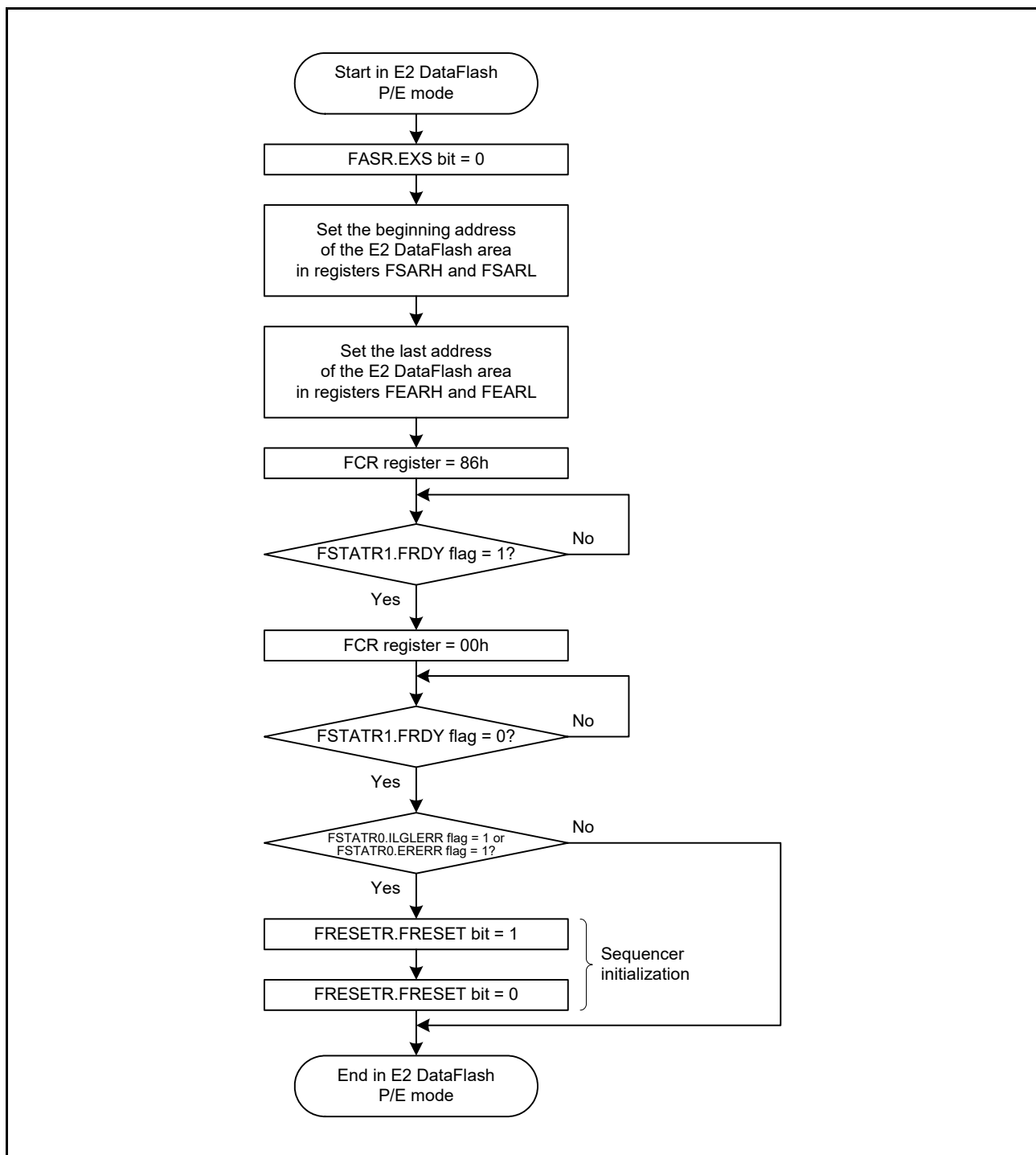


Figure 37.17 Procedure to Issue the All-Block Erase Command for the E2 DataFlash

37.7.4.4 Blank Check

Figure 37.18 and Figure 37.19 show the procedure to issue the blank check command.

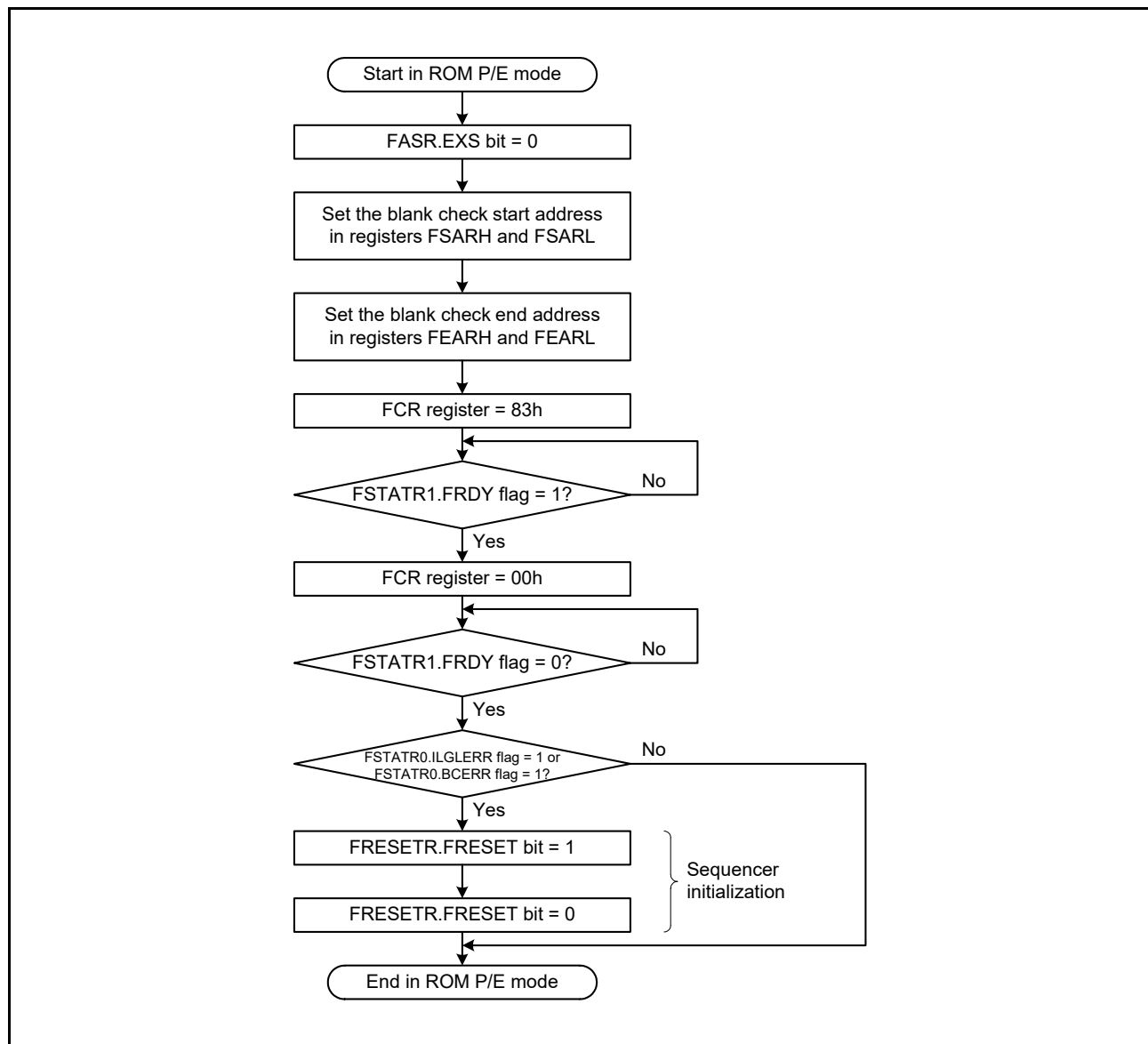


Figure 37.18 Procedure to Issue the Blank Check Command for the ROM

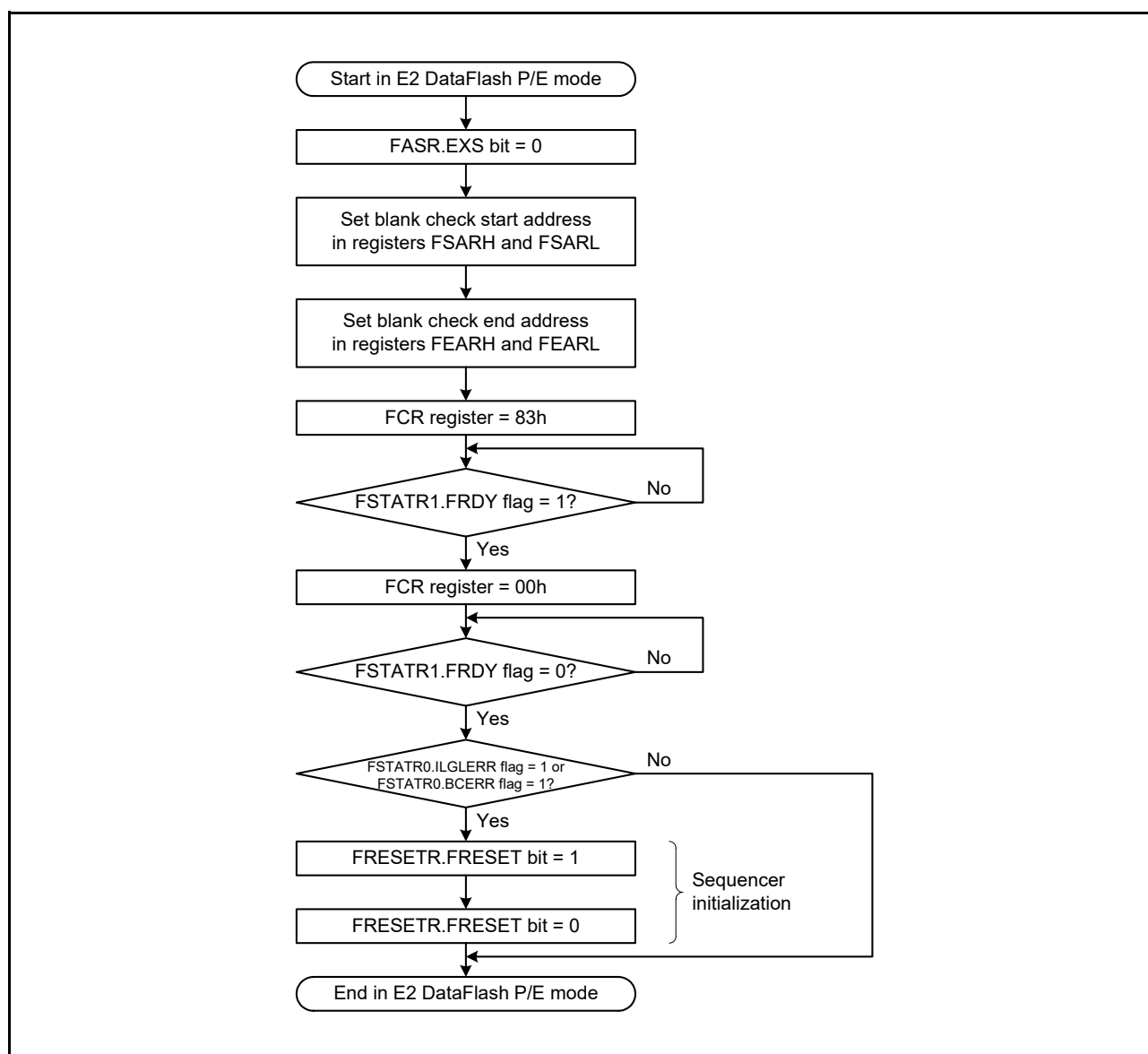


Figure 37.19 Procedure to Issue the Blank Check Command for the E2 DataFlash

37.7.4.5 Start-Up Area Information Program/Access Window Protection/Access Window Information Program

Figure 37.20 shows the procedure to issue the start-up area information program command, access window protection command, and access window information program command.

When the sequencer has directly entered ROM/PE mode from E2 DataFlash access disabled mode, set the DFLCTL.DFLEN bit to 1 at the beginning of the procedure.

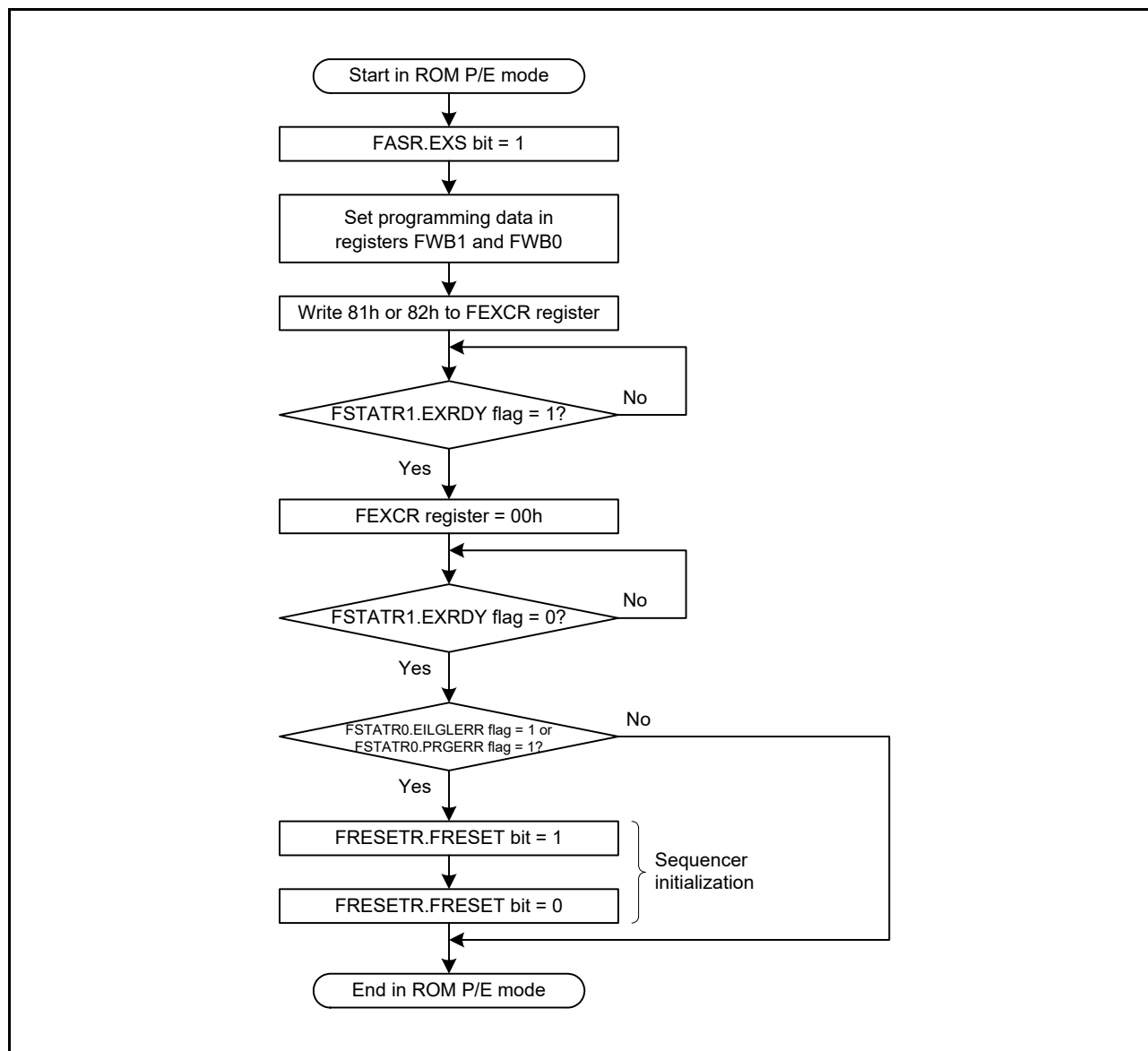


Figure 37.20 Procedure to Issue the Start-Up Area Information Program Command/Access Window Protection Command/Access Window Information Program Command

37.7.4.6 Forced Stop of Software Commands

Perform the procedure shown in Figure 37.21 to forcibly stop the blank check command or block erase command. When the command processing is forcibly stopped, registers FEAMH and FEAML store the address at the time of the forced stop. For blank check, the stopped processing can be continued by copying the FEAMH and FEAML register values to registers FSARH and FSARL.

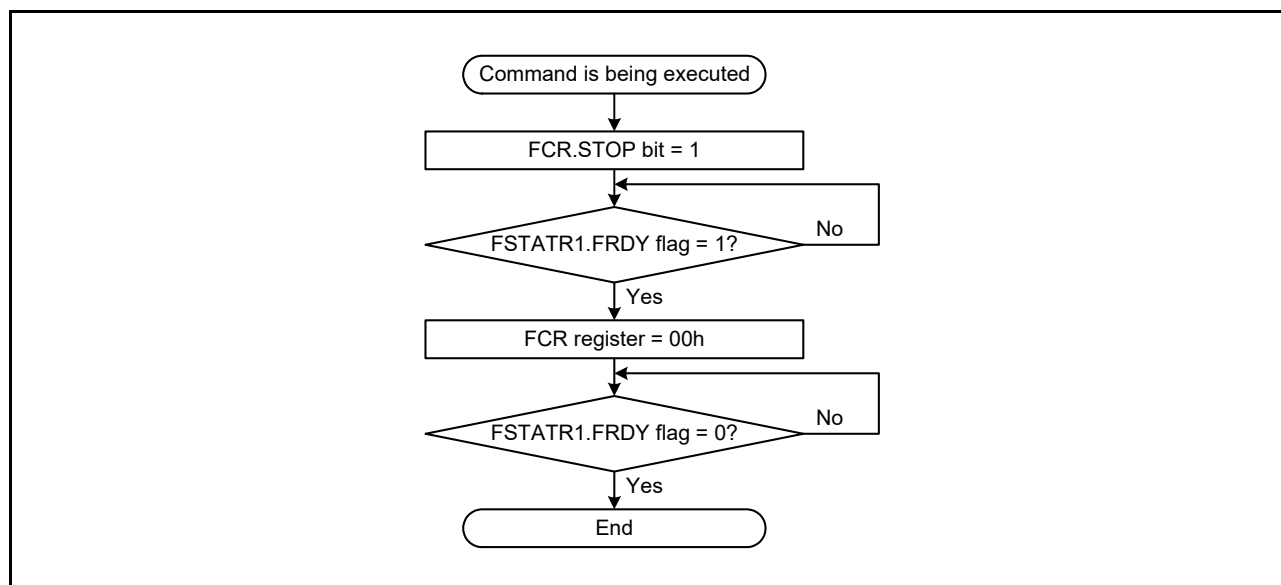


Figure 37.21 Procedure for Forced Stop of Software Commands

37.7.5 Interrupt

When software command processing or forced stop processing is completed, an interrupt (FRDYI) is generated.

When the FSTATR1.FRDY flag becomes 0 by setting the FCR.OPST bit to 0 and the FSTATR1.EXRDY flag becomes 0 by setting the FEXCR.OPST bit to 0, the next interrupt (FRDYI) can be accepted.

Clear the IRn.IR flag before setting the IERm.IENj bit of the ICU corresponding to this interrupt.

37.8 Boot Mode

The SCI or FINE interface is used in boot mode.

Table 37.7 lists the programmable and erasable areas and peripheral modules used in boot mode. Table 37.8 lists the I/O pins used in boot mode.

Table 37.7 Programmable and Erasable Areas and Peripheral Modules Used in Boot Mode

Item	Boot Mode	
	SCI Interface	FINE Interface
Programmable and erasable areas	User area	User area
	Data area	Data area
Peripheral module	SCI1 (asynchronous serial communication)	FINE

Table 37.8 I/O Pins Used in Boot Mode

Pin Name	I/O	Mode	Description
MD	Input	Boot mode	Select operating mode (refer to section 3, Operating Modes).
MD/FINED	I/O	Boot mode (FINE interface)	Select operating mode/FINE data I/O
PB6/RXD1	Input	Boot mode (SCI interface)	Receive data*1
PB5/TXD1	Output		Transmit data*1

Note 1. When using the SCI, connect (pull up) this pin to VCC via a resistor.

37.8.1 Boot Mode (SCI Interface)

The flash memory can be programmed and erased using asynchronous serial communication in boot mode (SCI interface). The user area and data area can be rewritten.

When a reset is released while the MD pin is low, the MCU starts in boot mode (SCI interface).

Contact the manufacturer for details on the serial programmer.

37.8.1.1 Operating Conditions in Boot Mode (SCI Interface)

SCI1 is used to communicate with the serial programmer in boot mode (SCI interface).

Figure 37.22 shows an example of pin connections in boot mode (SCI interface). Table 37.9 lists pin handling in boot mode (SCI interface).

An example of pin connections shown in Figure 37.22 is a simplified circuit. Operations are not guaranteed in all systems.

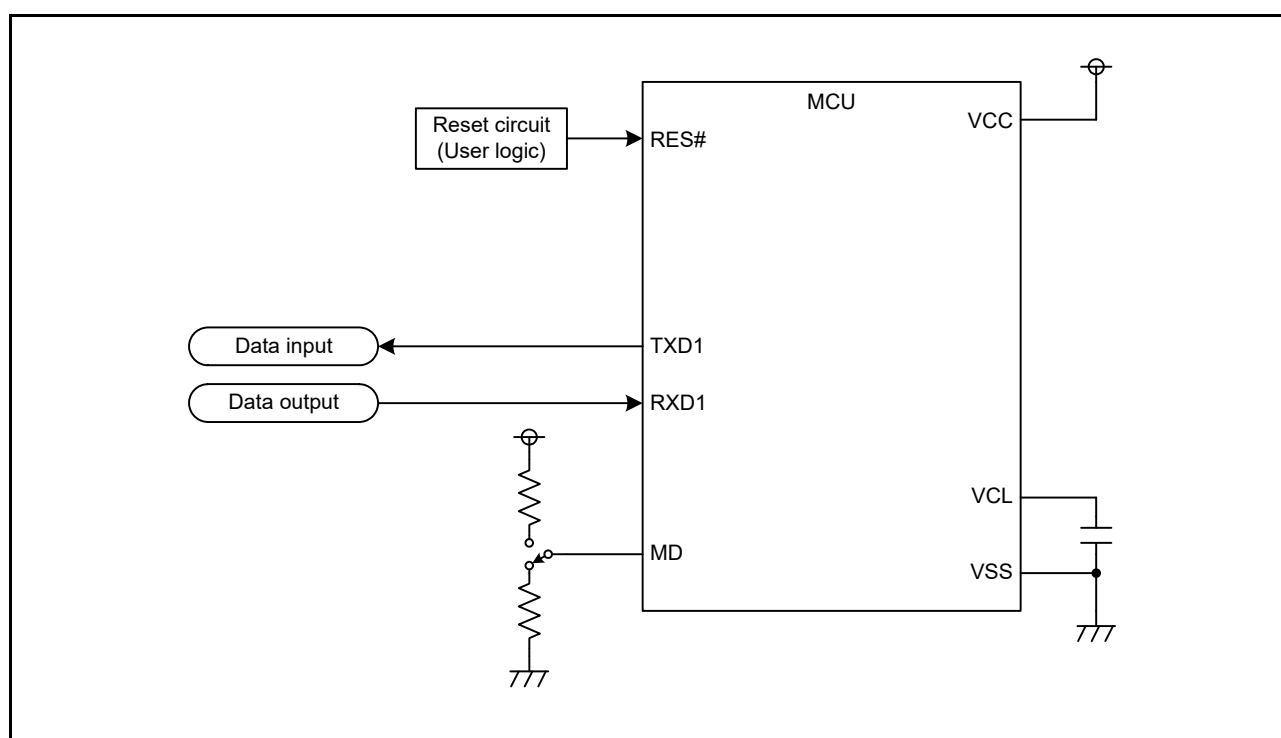


Figure 37.22 Example of Pin Connections in Boot Mode (SCI Interface)

Table 37.9 Pin Handling in Boot Mode (SCI Interface)

Pin Name	Name	I/O	Function
VCC, VSS	Power supply	—	Input 2.7 V or higher to the VCC pin. Input 0 V to the VSS pin.
VCL	Decoupling capacitor connect pin	—	Connect to the VSS pin via a decoupling capacitor for stabilizing the internal voltage.
MD	Operating mode control	Input	Input low.
RES#	Reset input	Input	Reset pin. Connect to the reset circuit.
PB6/RXD1	Data input RXD	Input	Input pin for serial data
PB5/TXD1	Data output TXD	Output	Output pin for serial data

As shown in Figure 37.23, set the format to 8-bit data, 1 stop bit, no parity, and LSB first to communicate with the serial programmer.

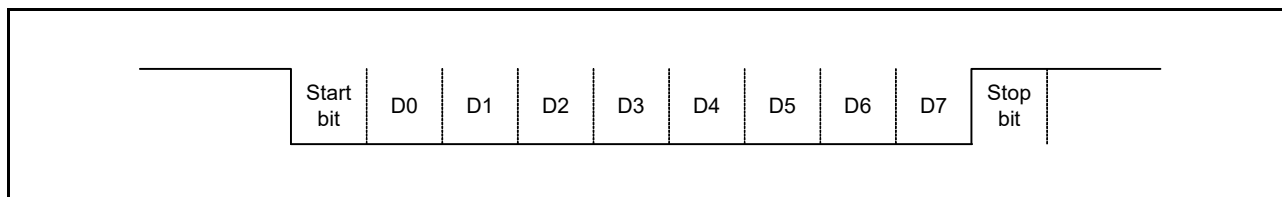


Figure 37.23 Communication Format

Initial communication with the programmer is performed at 9,600 or 19,200 bps. The communication bit rate can be changed after the MCU is connected with the programmer.

The maximum communication bit rate for communication in boot mode (SCI interface) is 2 Mbps.

37.8.1.2 Starting Up in Boot Mode (SCI Interface)

To start the MCU in boot mode (SCI interface), a reset must be released by changing the RES# pin from low to high while the MD pin is low. After starting up in boot mode (SCI interface), wait at least 400 ms until communication with the MCU is enabled in boot mode (SCI interface).

As shown in Figure 37.24, keep the signal of each pin unchanged for 400 ms after the reset is released. Use resets according to the range described in section 38.5.2, Reset Timing.

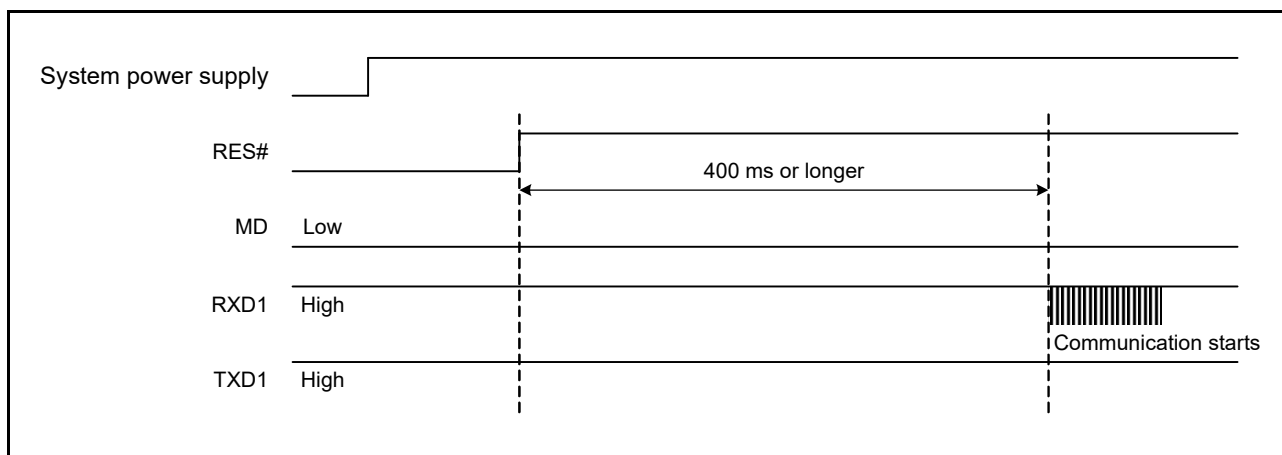


Figure 37.24 Wait Time until Communication Becomes Possible in Boot Mode (SCI Interface)

37.8.2 Boot Mode (FINE Interface)

The flash memory can be programmed and erased using the FINE in boot mode (FINE interface). The user area and data area can be rewritten.

Contact the manufacturer for details on the serial programmer.

37.8.2.1 Operating Conditions in Boot Mode (FINE Interface)

FINE is used to communicate with the serial programmer in boot mode (FINE interface).

Figure 37.25 shows an example of pin connections in boot mode (FINE interface). Table 37.10 lists pin handling in boot mode (FINE interface).

An example of pin connections shown in Figure 37.25 is a simplified circuit. Operations are not guaranteed in all systems.

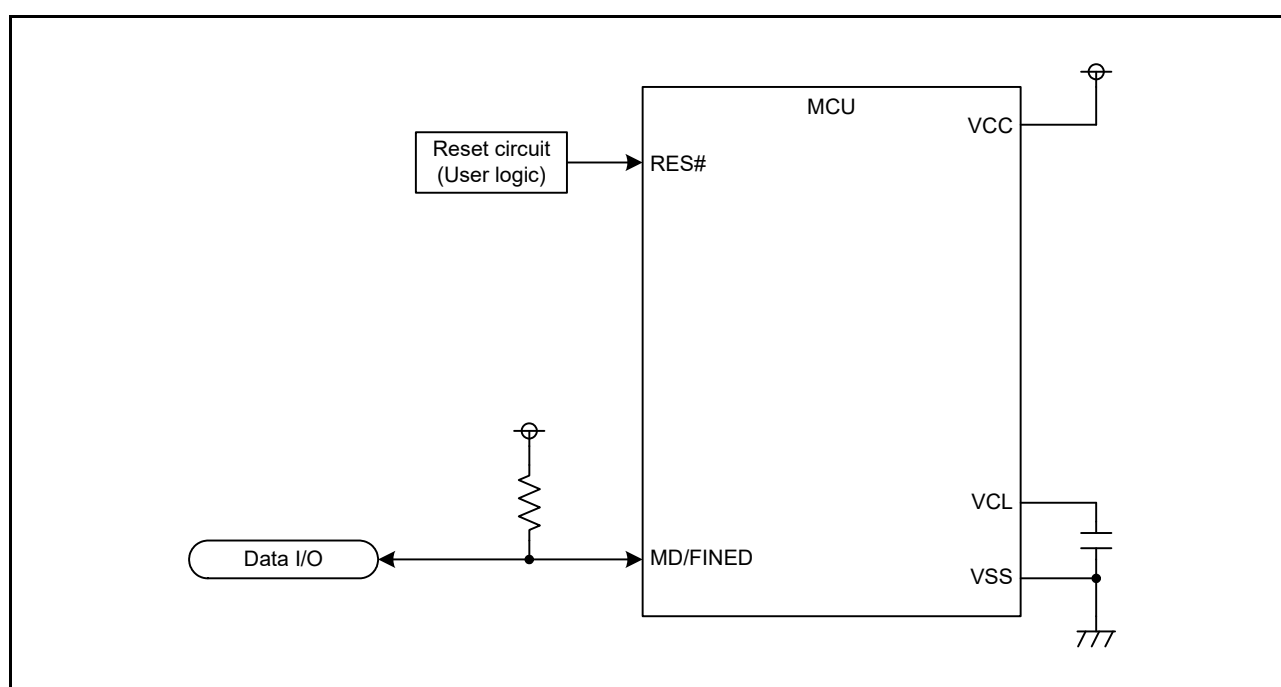


Figure 37.25 Example of Pin Connections in Boot Mode (FINE Interface)

Table 37.10 Pin Handling in Boot Mode (FINE Interface)

Pin Name	Name	I/O	Function
VCC, VSS	Power supply	—	Input 2.7 V or higher to the VCC pin. Input 0 V to the VSS pin.
VCL	Decoupling capacitor connect pin	—	Connect to the VSS pin via a decoupling capacitor for stabilizing the internal voltage.
MD/FINED	Operating mode control/data I/O	I/O	Connect to the VCC pin via a resistor (pull up).
RES#	Reset input	Input	Reset pin. Connect to the reset circuit.

37.9 Flash Memory Protection

Flash memory protection prevents the flash memory from being read or rewritten by the third party.

The boot mode ID code protection is for connecting the serial programmer, and the on-chip debugging emulator ID code protection is for connecting the on-chip debugging emulator.

37.9.1 ID Code Protection

There are two types of ID code protection: Boot mode ID code protection for connecting the serial programmer and on-chip debugging emulator ID code protection is for connecting the on-chip debugging emulator. The same ID codes are used for both functions, but operations differ.

ID codes consist of the control code and ID code 1 to ID code 15. Set ID codes to four 32-bit data in 32-bit units. Figure 37.26 shows the ID code configuration.

	31	24 23	16 15	8 7	0
FFFF FFA0h	Control code	ID code 1	ID code 2	ID code 3	
FFFF FFA4h	ID code 4	ID code 5	ID code 6	ID code 7	
FFFF FFA8h	ID code 8	ID code 9	ID code 10	ID code 11	
FFFF FFACH	ID code 12	ID code 13	ID code 14	ID code 15	

Figure 37.26 ID Code Configuration

The following shows a program example for setting ID codes.

This is an example when setting the control code to 45h and setting ID codes to 01h, 02h, 03h, 04h, 05h, 06h, 07h, 08h, 09h, 0Ah, 0Bh, 0Ch, 0Dh, 0Eh, and 0Fh (from the ID code 1 field to the ID code 15 field).

C language:

```
#pragma address ID_CODE = 0xFFFFF0A0
const unsigned long ID_CODE [4] = {0x45010203, 0x04050607, 0x08090A0B, 0x0C0D0E0F};
```

Assembly language:

```
.SECTION ID_CODE, CODE
.ORG 0xFFFFF0A0h
.LWORD 45010203h
.LWORD 04050607h
.LWORD 08090A0Bh
.LWORD 0C0D0E0Fh
```

37.9.1.1 Boot Mode ID Code Protection

Boot mode ID code protection disables reading and programming of the user area and data area when the serial programmer is connected by the third party.

When the control code indicates 45h or 52h (boot mode ID code protection is enabled), the MCU compares 16-byte ID code sent from the serial programmer with the ID code in the user area. According to the comparison result, reading and programming the user area and data area are enabled.

When the control code indicates a value other than 45h and 52h (boot mode ID code protection is disabled), all blocks in the user area and data area are erased, and reading and programming the user area and data area are enabled.

The control code is used to enable or disable protection. Table 37.11 lists the specifications of boot mode ID code protection, and Figure 37.27 shows the authentication flow of boot mode ID code protection.

ID code 1 to ID code 15 can be set to any desired value.

However, only when disabling connection with the serial programmer, the ID codes must be set to 50h, 72h, 6Fh, 74h, 65h, 63h, 74h, FFh, FFh, FFh, FFh, FFh, FFh, FFh, and FFh (from the ID code 1 field to the ID code 15 field).

Table 37.11 Boot Mode ID Code Protection Specifications

ID Code		Protection	ID Code Matching Result	Operation
Control Code	ID Code 1 to ID Code 15			
45h	Any desired value	Enabled	Matched	Exit the boot mode ID code authentication state and enter the program/erase host command wait state.
			Not matched	Continue the boot mode ID code authentication state.
			Not matched three times consecutively	Erase all blocks in the user area and data area, and continue boot mode ID code authentication state.
52h	50h, 72h, 6Fh, 74h, 65h, 63h, 74h, FFh, ..., and FFh (8 bytes are all FFh)	Enabled	N/A	Disable reading or rewriting of the flash memory, regardless of the codes sent from the serial programmer.
	Other than above		Matched	Exit the boot mode ID code authentication state and enter the program/erase state.
			Not matched	Continue the boot mode ID code authentication state.
Other than above	Any desired value	Disabled	N/A	Erase all blocks in the user area and data area.

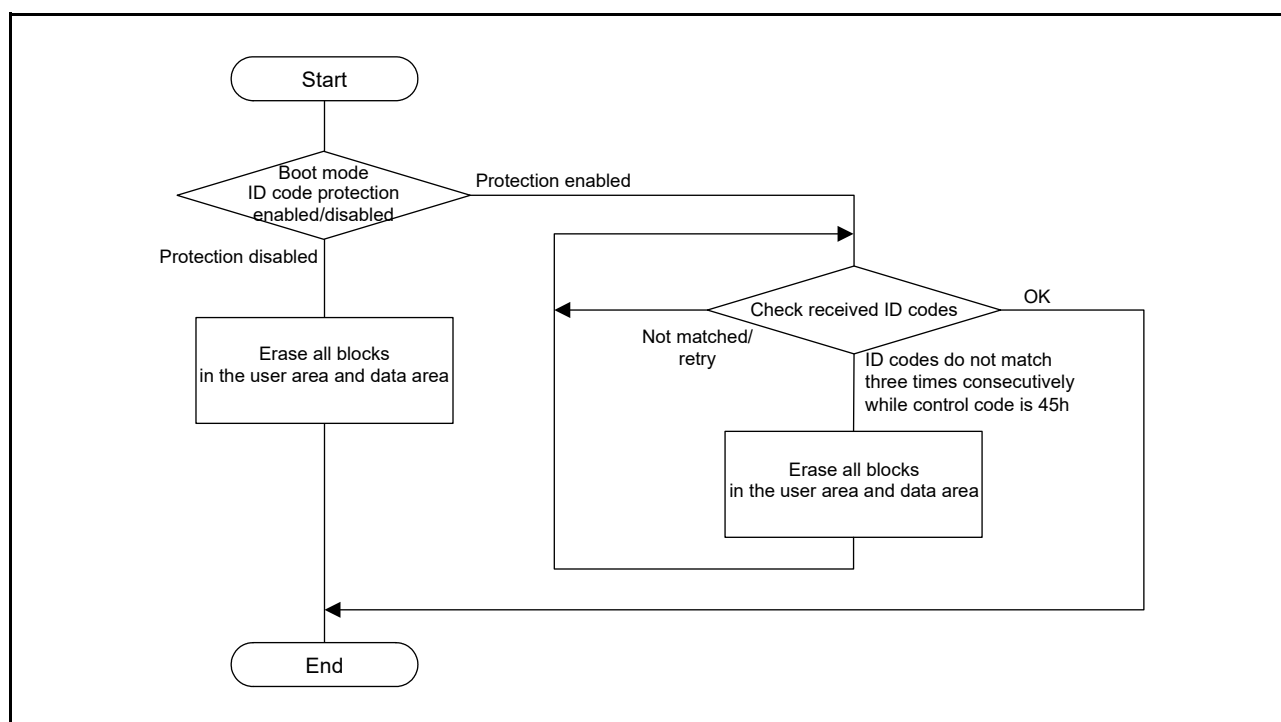


Figure 37.27 Authentication for Boot Mode ID Code Protection

37.9.1.2 On-Chip Debugging Emulator ID Code Protection

On-chip debugging emulator ID code protection enables or disables connection with the on-chip debugging emulator. When the on-chip debugging emulator ID code protection is disabled, connection with the on-chip debugging emulator is enabled. When 16-byte ID codes sent from the on-chip debugging emulator and ID codes in the user area match while on-chip debugging emulator ID code protection is enabled, connection with the on-chip debugging emulator is also enabled.

Table 37.12 lists the specifications of on-chip debugging emulator ID code protection.

Table 37.12 On-Chip Debugging Emulator ID Code Protection Specifications

ID Code		Protection	ID Code Matching Result	Operation
Control Code	ID Code 1 to ID Code 15			
FFh	FFh,..., and FFh (15 bytes are all FFh)	Disabled	N/A	Enable connection with the on-chip debugging emulator.
52h	50h, 72h, 6Fh, 74h, 65h, 63h, and 74h + any 8 bytes	Enabled	N/A	Disable connection with the on-chip debugging emulator, regardless of the codes sent from the on-chip debugging emulator.
Other than above	Other than above	Enabled	Matched	Enable connection with the on-chip debugging emulator.
			Not matched	Continue the ID code wait state.

37.10 Communication Protocol

This section describes the protocol used in boot mode. When developing a serial programmer, control with this communication protocol.

37.10.1 State Transition in Boot Mode (SCI Interface)

Figure 37.28 shows the state transition in boot mode (SCI interface).

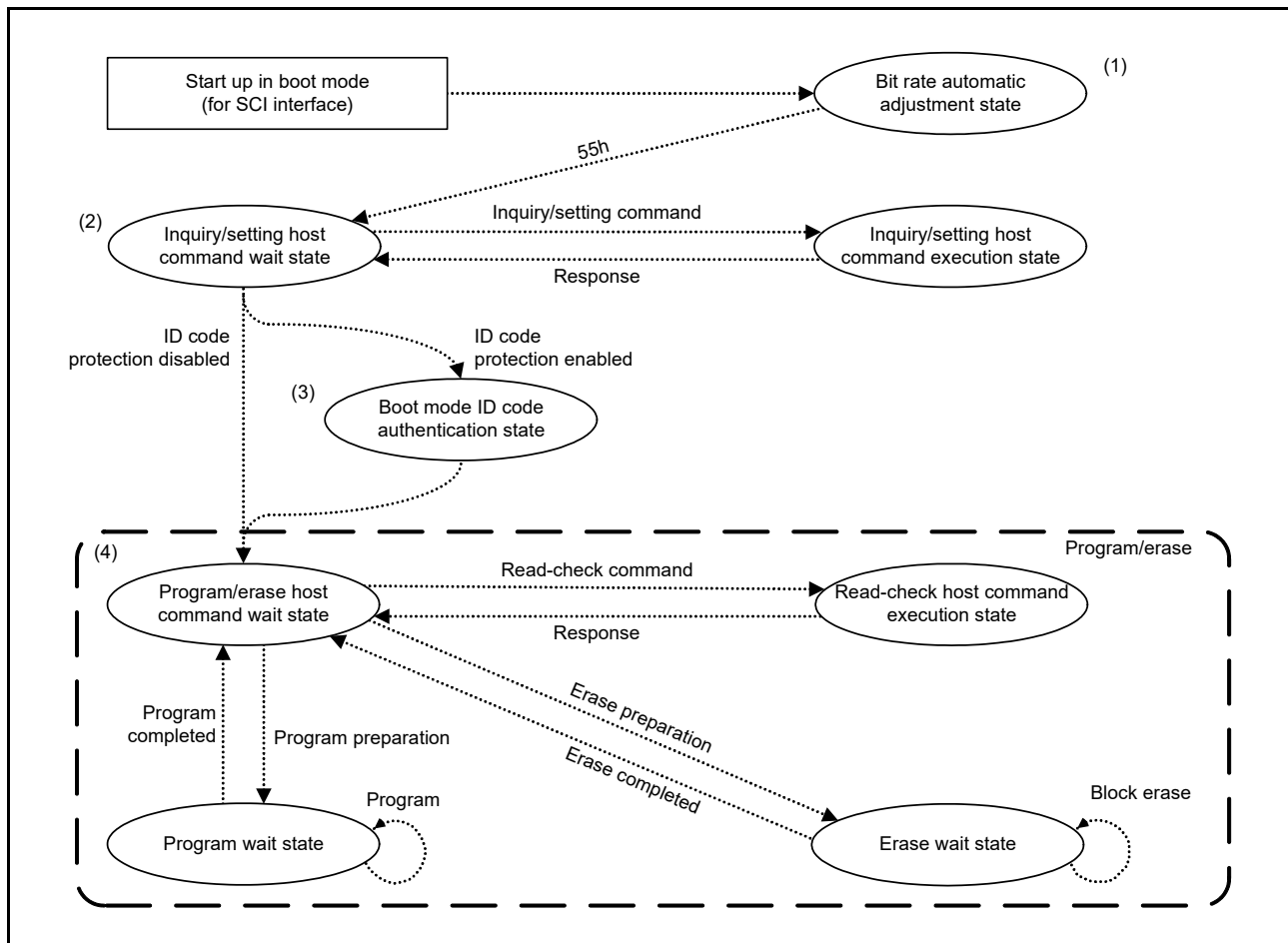


Figure 37.28 Boot Mode (SCI Interface) State Transition

(1) Bit rate automatic adjustment state

In this state, the bit rate is automatically adjusted to 9,600 or 19,200 bps for communication with the host.

When the bit rate adjustment is completed, the MCU sends 00h to the host. After that, when the MCU receives 55h sent from the host, the MCU sends E6h to the host, and enters the inquiry/setting host command wait state. The host must not send data until 400 ms elapse after a reset of the MCU is released.

(2) Inquiry/setting host command wait state

In this state, the host can make inquiries for the MCU information including block configuration, size, and addresses where the user area and data area are allocated, and select the endian of data and a bit rate.

When the MCU receives the program/erase host state transition command from the host, it determines whether boot mode ID code protection is enabled or disabled. If boot mode ID code protection is disabled, the MCU enters the inquiry/setting host command wait state. If boot mode ID code protection is enabled, the MCU enters the boot mode ID code authentication state.

Refer to section 37.10.5, Inquiry Commands and section 37.10.6, Setting Commands for details on inquiry/setting commands.

(3) Boot mode ID code authentication state

In this state, the MCU accepts the ID code authentication command.

When boot mode ID codes do not match, the MCU remains in the boot mode ID code authentication state.

Refer to section 37.9.1.1, Boot Mode ID Code Protection for details on boot mode ID code protection. Refer to section 37.10.7, ID Code Authentication Command for details on the ID code authentication command.

(4) Program/erase state

In this state, the MCU executes program/erase or read-check commands according to commands sent from the host.

Refer to section 37.10.8, Program/Erase Commands for details on program/erase commands. Refer to section 37.10.9, Read-Check Commands for details on read-check commands.

37.10.2 Command and Response Configuration

The communication protocol is composed of a “Command” sent from the host to the MCU and a “Response” sent from the MCU to the host. Commands include 1-byte commands and multiple-byte commands. Responses include 1-byte responses, multiple-byte responses, and error responses.

A multiple-byte command and multiple-byte response have “Size” for informing the number of transmit/receive data bytes and “SUM” for detecting communication errors.

“Size” indicates the number of transmit/receive data bytes excluding Command code (the first byte), Size, and SUM.

“SUM” indicates byte data that is calculated so the total bytes of Command or Response becomes 00h.

The flash memory addresses for reading are used as the following addresses: the program address specified in the program command, the block start address specified in the block erase command, the AW start and end addresses specified in the access window information program command, and the AW start and end addresses received in the access window read command.

37.10.3 Response to Undefined Commands

When the MCU receives an undefined command, it sends a command error as a response. The contents of the response are shown below. “Command code” in the error response stores the first byte of the command sent from the MCU.

Error response	80h	Command code

37.10.4 Boot Mode Status Inquiry

This command is used to check the current state and which type of an error occurred immediately after a command issued in the boot program.

Table 37.13 and Table 37.14 list a state or error that the MCU responds to.

The boot mode status inquiry command can be used in the inquiry/setting host command wait state and program/erase host command wait state.

Command	4Fh				
Response	5Fh	Size	State	Error	SUM

Size (1 byte): Total bytes of "State" and "Error" (the value is always 02h)

State (1 byte): MCU's current state (see Table 37.13)

Error (1 byte): Information about the error occurred in response to a command issued immediately before (see Table 37.14)

SUM (1 byte): Value that is calculated so the sum of response data is 00h

Table 37.13 Information Regarding the States

Code	State*1	Description
11h	Inquiry/setting host command wait state	Device selection wait state
12h/13h		Operating frequency selection wait state
1Fh		Program/erase host command wait state transition command wait state
31h	Boot mode ID code authentication state	The user area and data area are being erased
3Fh	Program/erase host command wait state	Program/erase command wait state
4Fh		Program data reception wait state
5Fh		Block erase specification wait state

Note 1. Refer to Figure 37.28 for details on the states.

Table 37.14 Error Information

Code	Description
00h	No error
11h	SUM error
21h	Device code error
24h	Bit rate selection error
29h	Block start address error
2Ah	Address error
2Bh	Data length error
51h	Erase error
52h	Not blank (blank check error)
53h	Program error
61h	ID code do not match
63h	ID code do not match and erase error
80h	Command error
FFh	Bit rate automatic adjustment error

37.10.5 Inquiry Commands

Inquiry commands are used to obtain necessary information for sending setting commands, program/erase commands, and read-check commands. Table 37.15 lists the inquiry commands. These commands can only be used in the inquiry/setting host command wait state.

Table 37.15 Inquiry Commands

Command	Description
Supported device inquiry	Inquiry for the device code and series name
Data area availability inquiry	Inquiry for the availability of the data area
User area information inquiry	Inquiry for the number of user areas, and the start and end addresses of the user area
Data area information inquiry	Inquiry for the number of data areas, and the start and end addresses of the data area
Block information inquiry	Inquiry for the start address, the block size, and the number of blocks of each of the user and data areas

37.10.5.1 Supported Device Inquiry

This command is used to obtain the device information for identifying the endian of developed software.

When the MCU receives this command, it sends the device information when developed software uses little endian data and the device information when developed software uses big endian data in this order.

Command	20h		
Response	30h	Size	Number of devices
	Number of characters	Device code for little endian	
	Number of characters	Device code for big endian	
	SUM	Series name for little endian	

Size (1 byte): Total bytes of Number of Devices, Characters, Device code, and Series name

Number of devices (1 byte): Number of endian types that the MCU supports (the value is always 02h)

Number of characters (1 byte): Number of characters for the device code and device name

Device code (4 bytes): Identification code indicating the endian of developed software

Series name (n bytes): The series name of the MCU (ASCII code) and the classification of little endian/big endian

SUM (1 byte): Value that is calculated so the sum of response data is 00h

37.10.5.2 Data Area Availability Inquiry

When the MCU receives this command, it sends the result indicating that the data area is available, area protection can be used, the data area program command is available, and the access window protection is available.

Command	2Ah			
Response	3Ah	Size	Availability	SUM

Size (1 byte): Number of characters of Availability (the value is always 01h)

Availability (1 byte): Availability of the data area (the value is always 5Dh)

5Dh represents the data area is available, area protection can be used, data area program command is available, and the access window protection is available.

SUM (1 byte): Value that is calculated so the sum of response data is 00h (the value is always 68h)

37.10.5.3 User Area Information Inquiry

When the MCU receives this command, it sends the number of user areas and addresses.

Command	25h			
Response	35h	Size	Number of areas	
	Area start address			
	Area end address			
	SUM			

Size (1 byte): Total bytes of Number of areas, Area start address, and Area end address (the value is always 09h)

Number of areas (1 byte): Number of user areas (the value is always 01h)

Area start address (4 bytes): Start address of the user area

Area end address (4 bytes): End address of the user area

SUM (1 byte): Value that is calculated so the sum of the response data is 00h

37.10.5.4 Data Area Information Inquiry

When the MCU receives this command, it sends the number of data areas and addresses.

Command	2Bh		
Response	3Bh	Size	Number of areas
	Area start address		
	Area end address		
	SUM		

Size (1 byte): Total bytes of data of Number of areas, Area start address, and Area end address (the value is always 09h)

Number of areas (1 byte): Number of areas in the data area (the value is always 01h)

Area start address (4 bytes): Start address of the data area (the value is always 0010 0000h)

Area end address (4 bytes): End address of the data area (the value is always 0010 0FFFh)

SUM (1 byte): Value that is calculated so the sum of the response data is 00h (the value is always 8Dh)

37.10.5.5 Block Information Inquiry

When the MCU receives this command, it sends the start address, the size of one block, and the number of blocks in the user area and data area.

Command	26h		
Response	36h	Size	DDh
	Start address of the user area		
	Block size of one block for the user area		
	Number of blocks of the user area		
	Start address of the data area		
	Block size of one block for the data area		
	Number of blocks of the data area		
	SUM		

Size (2 bytes): Total bytes of data from DDh to Number of blocks of the data area (the value is always 00 19h)

Start address of the user area (4 bytes): Start address of the user area

Block size of one block for the user area (4 bytes): Memory size of one block (the value is always 00 00 08 00h)

Number of blocks of the user area (4 bytes): Number of blocks in the user area

Start address of the data area (4 bytes): Start address of the data area (the value is always 00 10 00 00h)

Block size of one block for the data area (4 bytes): Memory size of one block (the value is always 00 00 01 00h)

Number of blocks of the data area (4 bytes): Number of blocks in the data area (the value is always 00 00 00 10h)

SUM (1 byte): Value that is calculated so the sum of response data is 00h

37.10.6 Setting Commands

Setting commands are used to configure the settings necessary to execute program/erase commands in the MCU.

Table 37.16 lists setting commands. These commands can be used only in the inquiry/setting host command wait state.

Table 37.16 Setting Commands

Command	Function
Device select	Select a device code.
Operating frequency select	Change the bit rate for communication.
Program/erase host command wait state transition	Enter the program/erase host command wait state or boot mode ID code authentication state.

37.10.6.1 Device Select

This command is used to specify the endian of developed software. Select a device code from among the device codes obtained in the response to the support device inquiry command.

If the received device code matches the supported device, the MCU sends a response (46h).

If the device is not supported or the SUM of the received command does not match, the MCU sends an error response.

Command	10h	Size	Device code	SUM

Size (1 byte): Number of characters of the device code (the value is always 04h)

Device code (4 bytes): Identification code to identify an endian of the developed software
(code in the response to the support device inquiry command)

SUM (1 byte): Value that is calculated so the sum of command data is 00h

Response	46h

Error response	90h	Error

Error (1 byte): Error code

11h: SUM error

21h: Device code error

37.10.6.2 Operating Frequency Select

This command is used to specify the operating frequency of the MCU and a bit rate for communication with the flash memory programmer. The bit rate selected in this command should be set to a value with error of less than 4% compared to the bit rate obtained by dividing 32 MHz.

If the specified settings can be supported, the MCU sends a response (06h). If the bit rate error is 4% or more or the SUM of the received command does not match, the MCU sends an error response.

After the host receives a response, wait for at least a 1-bit period at the old bit rate, and send communication confirmation data at the new bit rate.

If the MCU successfully receives communication confirmation data, the MCU sends a response (06h). If the MCU fails to receive the communication confirmation data, the MCU sends an error response.

Command	3Fh	Size	Bit rate		Dummy data
	Number of clocks	Multiplier 1	Multiplier 2		
	SUM				

Size (1 byte): Total bytes of data of Bit rate, Dummy data, Number of clocks, and Multiplier (the value is always 07h)

Bit rate (2 bytes): New bit rate

The value is calculated by dividing the bit rate by 100 (Example: Set 00C0h for 19200 bps)

Dummy data (2 bytes): The value should always be set to 0000h

Number of clocks (1 byte): Types of clocks for multiplier setting (the value is always 02h)

Multiplier 1 (1 byte): Multiplier of the system clock (ICLK) (the value is always 01h)

Multiplier 2 (1 byte): Multiplier of the peripheral module clock (PCLK) (the value is always 01h)

SUM (1 byte): Value that is calculated so the sum of command data including dummy data is 00h

Response	06h

Error response	BFh	Error

Error (1 byte): Error code

11h: SUM error

24h: Bit rate selection error

Communication confirmation	06h

Response	06h

Error response	FFh

- Bit rate selection error

A bit rate selection error occurs when the bit rate specified with the operating frequency select command cannot be set to a value with error of less than 4%. When the new bit rate specified with the operating frequency select command is B, and 32 (MHz) is Pφ, the bit rate error is calculated by the following formula:

$$\text{Error}(\%) = \left(\frac{P\phi \times 10^6}{B \times 16 \times N} - 1 \right) \times 100$$

$$N = \text{INT} \left(\frac{P\phi \times 10^6}{B \times 16} \right)$$

Pφ: 32 (MHz)

B: New bit rate (bps)

N: Ratio between Pφ and the new bit rate multiplied by 16 (however, $1 \leq N \leq 256$)

37.10.6.3 Program/Erase Host Command Wait State Transition

This command is used for the transition from the inquiry/setting host command wait state to the program/erase host command wait state.

When the MCU receives this command, it determines whether boot mode ID code protection is enabled or disabled.

When boot mode ID code protection is disabled, all blocks in the user area and data area are erased.

When all blocks are successfully erased, the MCU sends a response (06h) and enters the program/erase host command wait state. If not all blocks are successfully erased, the MCU sends an error response.

When boot mode ID code protection is enabled, the MCU sends a response (16h) and enters boot mode ID code authentication state.

Command

40h

Response

ACK

ACK (1 byte): ACK code

06h: ID code protection is disabled.

16h: ID code protection is enabled.

Error response

C0h	Error

Error (1 byte): Error code

51h: Erase error

37.10.7 ID Code Authentication Command

This command is used for ID code authentication when boot mode ID code protection is enabled.

Table 37.17 lists ID code authentication command. This command can be used only in the boot mode ID code authentication state.

Table 37.17 ID Code Authentication Command

Command	Function
ID code check	Compare the 16-byte code sent from the host and ID code.

37.10.7.1 ID Code Check

This command is used to unlock boot mode ID code protection.

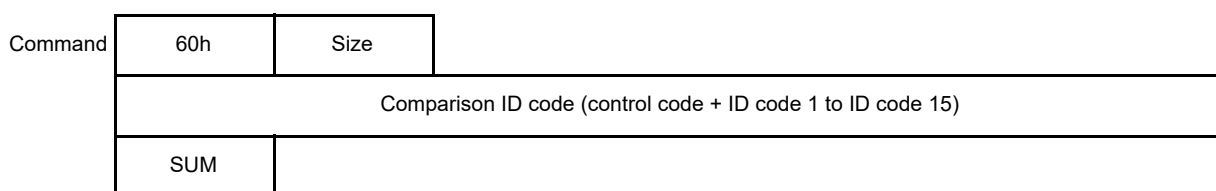
The comparison ID code specified with the command should be set to the same value as the control code and ID code 1 to ID code 15.

If the comparison ID code sent from the host matches the ID code programmed in the user area, the MCU sends a response (06h) and enters program/erase host command wait state.

If the codes do not match or the SUM of the received command does not match, the MCU sends an error response.

When the ID codes do not match three times consecutively while the control code is 45h, all blocks in the user area and data area are erased. If an error occurs during erasure, the MCU sends an error response.

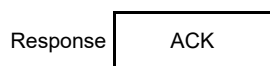
Also, even if all blocks are successfully erased, the MCU sends an error response and continues the boot mode ID code state. Reset the MCU to enter the program/erase host command wait state.



Size (1 byte): Number of bytes of ID codes (the value is always 10h)

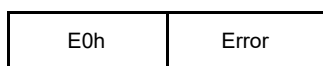
ID code (16 bytes): Control code (1 byte) + ID code 1 to ID code 15 (15 bytes)

SUM (1 byte): Value that is calculated so the sum of the command data is 00h



ACK (1 byte): ACK code

06h: The MCU enters the program/erase host command wait state.



Error (1 byte): Error code

11h: SUM error

61h: ID codes do not match

63h: ID codes do not match and erase error

37.10.8 Program/Erase Commands

Program/erase commands are used to program or erase the user area or data area based on the response to inquiry commands. Table 37.18 lists commands used in each of the program/erase host command wait state, program wait state, and erase wait state. Table 37.19 lists commands that can be accepted in each state.

When a command that is not listed in Table 37.19 is received in each state, the MCU sends a command error response.

Table 37.18 Program/Erase Commands

Command	Function
User/data area program preparation	Select the user area or data area to program, and enter the program wait state.
Program	Program the specified data to the selected area in the user area or data area. Or enter the program/erase host command wait state (end of program).
Data area program	Program the specified-size data to the selected area in the data area. Or enter the program/erase host command wait state (end of program).
Erase preparation	Enter the erase wait state.
Block erase	Erase the selected block, or enter the program/erase host command wait state (end of erase).

Table 37.19 Acceptable Commands for Each State

State	Acceptable Command
Program/erase host command wait state	User/data area program preparation command, and erase preparation command
Program wait state	Program command and data area program command
Erase wait state	Block erase command

37.10.8.1 User/Data Area Program Preparation

This command is used to prepare for accepting the program command and the data area program command.

When the MCU receives this command, it recognizes that an instruction to prepare for the program command is issued from the host. Then, the MCU enters the program wait state, where only the program command and the data area program command can be accepted, and sends a response (06h).

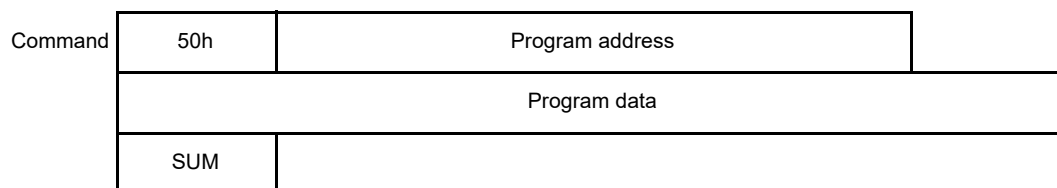
Command	43h
Response	06h

37.10.8.2 Program

This command is used to program the specified data to the user area or data area. Set the lower 8 bits to 0 for the program address selected in this command. When the data length is shorter than 256 bytes, the data cannot be programmed. Fill the gaps with FFh.

When the program from the selected address is successfully completed, the MCU sends a response (06h). If the SUM of the received command does not match or an error occurs during a program operation, the MCU sends an error response.

To enter the program/erase host command wait state after the program operation ends, send 50h FFh FFh FFh FFh B4h from the host. The MCU sends a response (06h), and enters the program/erase host command wait state.



Program address (4 bytes): Address for program destination

Set the lower 8 bits to 0

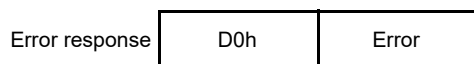
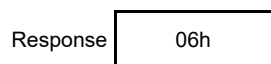
Set FFFF FFFFh for end of program

Program data (n bytes): Program data (n = 256, 0 for end of program)

When the program data is less than n bytes, set FFh for the missing data.

No program data for the end of program

SUM (1 byte): Value that is calculated so the sum of command data is 00h



Error (1 byte): Error code

11h: SUM error

2Ah: Address error (the address is not in the selected area.)

53h: Program error (the data cannot be programmed.)

37.10.8.3 Data Area Program

This command is used to program the specified data to the data area. Set the lower 2 bits to 0 for the program address selected in this command. When the data length is shorter than 4 bytes, the data cannot be programmed. Fill the gaps with FFh.

When the program from the selected address is successfully completed, the MCU sends a response (06h). If the SUM of the received command does not match or an error occurs during a program operation, the MCU sends an error response.

To enter the program/erase host command wait state after the program operation ends, send 51h FFh FFh FFh 00h B3h from the host. The MCU sends a response (06h), and enters the program/erase host command wait state.

Command	51h	Program address	Program data length
	Program data		
	SUM		

Program address (4 bytes): Address for program destination

Set the lower 2 bits of the selected address to 0

Set FFFF FFFFh for end of data area program

Program data length (1 byte): Size of program data

Set 4-byte data

Set 00h for end of data area program

Program data (n bytes): Program data for the data area (n = program data length, 0 for end of program)

Set data of the program data length

When the program is less than n bytes, set FFh for the missing data.

No program data for the end of data area program

SUM (1 byte): Value that is calculated so the sum of command data is 00h

Response	06h

Error response	D1h	Error

Error (1 byte): Error code

11h: SUM error

2Ah: Address error (the address is not in the selected area.)

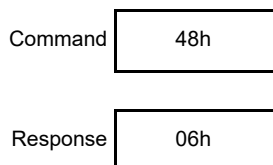
2Bh: Program data length error

53h: Program error (the data or program data cannot be programmed.)

37.10.8.4 Erase Preparation

This command is used to prepare for accepting the block erase command.

When the MCU receives this command, it recognizes that an instruction to prepare for the erase command is issued from the host. Then, the MCU enters the erase wait state, where only the block erase command can be accepted, and sends a response (06h).



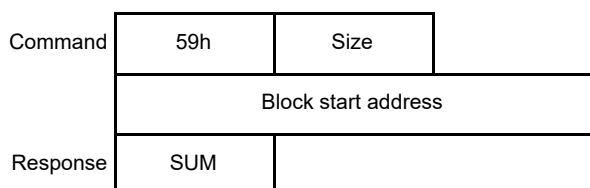
37.10.8.5 Block Erase

This command is used to erase the selected block in the user area or data area.

Specify the block start address selected in the command by calculating the address based on the response to the block information inquiry command.

When the block selected in the block start address is successfully erased, the MCU sends a response (06h). If the SUM of the received command does not match or an error occurs during an erase operation, the MCU sends an error response.

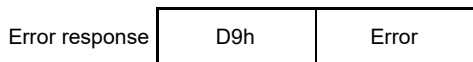
To enter the program/erase host command wait state after the erase operation ends, send 59h 04h FFh FFh FFh FFh A7h from the host. The MCU enters the program/erase host command wait state and sends a response (06h).



Size (1 byte): Total bytes of Block start address (the value is always 04h)

Block start address (4 bytes): Start address of the block that is erased
Set FFFF FFFFh for end of erase

SUM (1 byte): Value that is calculated so the sum of response data is 00h



Error (1 byte): Error code

11h: SUM error

29h: Block start address error

51h: Erase error (the selected block cannot be erased)

37.10.9 Read-Check Commands

Read-check commands are used to read data or check whether data is programmed in the user area or data area in the MCU based on the response to inquiry commands.

Table 37.20 lists read-check commands used in the program/erase host command wait state.

Table 37.20 Read-Check Commands

Command	Function
Memory read	Read data from the user area or data area.
User area checksum	Obtain the checksum of the entire user area.
Data area checksum	Obtain the checksum of the entire data area.
User area blank check	Check whether data is programmed in the user area.
Data area blank check	Check whether data is programmed in the data area.
Access window information program	Set the access window.
Access window read	Read the settings of the access window.

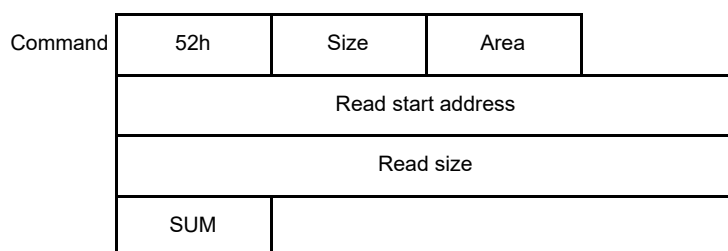
37.10.9.1 Memory Read

This command is used to read data programmed in the user area or data area.

For a read start address selected in the command, set a value within the range from the area start address to the area end address received in the response to the user area information inquiry command or the data area information inquiry command.

For a read size selected in the command, set a value so the sum of the read start address and the read size is within the range from the area start address to the area end address received in the response to the user area information inquiry command or the data area information inquiry command.

When the MCU performs a read successfully, it sends data of the specified range. If the SUM of the received command does not match or the MCU fails to perform a read successfully, it sends an error response.



Size (1 byte): Total bytes for Read start address and Read size

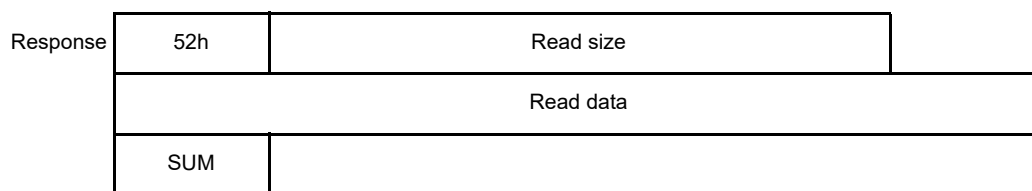
Area (1 byte): Area that is read

01h: User area or data area

Read start address (4 bytes): Start address of the area that is read

Read size (4 bytes): Size of data that is read (in bytes)

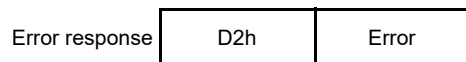
SUM (1 byte): Value that is calculated so the sum of response data is 00h



Read size (4 bytes): Size of Data that is read (in bytes)

Read data (n bytes): Data read from the specified range (n = read size)

SUM (1 byte): Value that is calculated so the sum of response data is 00h



Error (1 byte): Error code

11h: SUM error

2Ah: Address error

- A value other than 01h is set for the "Area" field.
- The read start address is not in the selected area.

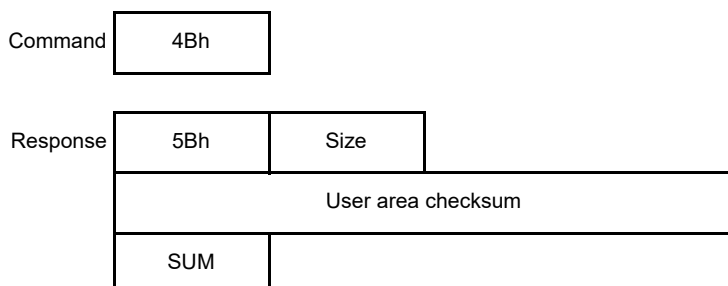
2Bh: Size error

- The read size is set to 0000 0000h.
- The read size exceeds the area size.
- The address calculated from the read start address and read size is not in the selected area.

37.10.9.2 User Area Checksum

This command used to obtain the checksum of the entire user area.

When the MCU receives this command, it adds data from the start address to the end address in bytes in the user area, and sends the calculated result (checksum) as a response.



Size (1 byte): Number of bytes for checksum of the user area (the value is always 04h)

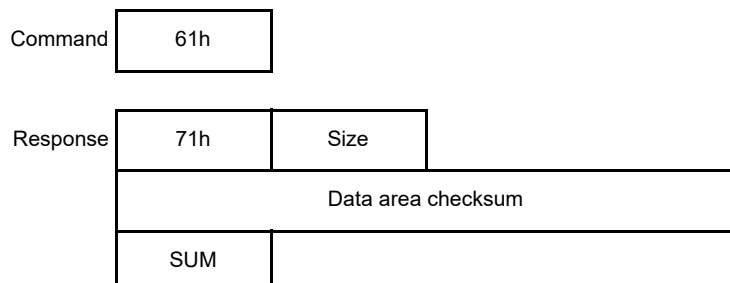
User area checksum (4 bytes): Calculated result of the data in the user area in bytes

SUM (1 byte): Value that is calculated so the sum of response data is 00h

37.10.9.3 Data Area Checksum

This command used to obtain the checksum of the entire data area.

When the MCU receives this command, it adds data from the start address to the end address in bytes in the data area, and sends the calculated result (checksum) as a response.



Size (1 byte): Number of bytes for checksum of the data area (the value is always 04h)

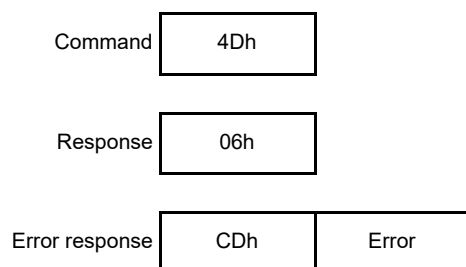
Data area checksum (4 bytes): Calculated result of the data in the data area in bytes

SUM (1 byte): Value that is calculated so the sum of response data is 00h

37.10.9.4 User Area Blank Check

This command is used to check whether data is programmed in the user area.

When the MCU receives this command, it checks whether there is data in the entire user area. If there is no programmed data, the MCU sends a response (06h). If there is at least 1 byte of programmed data, the MCU sends an error response.



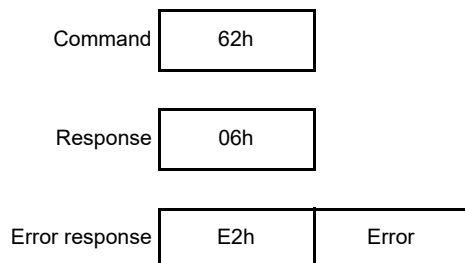
Error (1 byte): Error code

52h: Not blank

37.10.9.5 Data Area Blank Check

This command is used to check whether data is programmed in the user area.

When the MCU receives this command, it checks whether there is programmed data in the entire user area. If there is no programmed data, the MCU sends a response (06h). If there is at least 1 byte of programmed data, the MCU sends an error response.



Error (1 byte): Error code
52h: Not blank

37.10.9.6 Access Window Information Program

This command is used to set the access window used for area protection.

For the access window start address selected in the command, set the start address of the start block. For the access window end address, set the end address of the end block.

When the specified access window settings are successfully completed, the MCU sends a response (06h). If the SUM of the received command does not match or an error occurs during the access window settings, the MCU sends an error response.

For details on the access window, see [section 37.6, Area Protection](#).

Command	74h	05h	Access window	
	Access window start address LH	Access window start address HL	Access window end address LH	Access window end address HL
	SUM			

Access window (1 byte): Set the access window or clear the access window settings
Set 00h to set the access window
Set FFh to clear the access window settings

Access window start address LH (1 byte): Start address of the access window (A15 to A8)
Set A15 to A8 of the start address of the start block.
Set FFh to clear the access window settings

Access window start address HL (1 byte): Start address of the access window (A23 to A16)
Set A23 to A16 of the start address of the start block.
Set FFh to clear the access window settings

Access window end address LH (1 byte): End address of the access window (A15 to A8)
Set A15 to A8 of the end address of the end block.
Set FFh to clear the access window settings

Access window end address HL (1 byte): End address of the access window (A23 to A16)
Set A23 to A16 of the end address of the end block.
Set FFh to clear the access window settings

SUM (1 byte): Value that is calculated so the sum of response data is 00h

Response	06h

Error response	F4h	Error

Error (1 byte): Error code

11h: SUM error

2Ah: Address error (specified address is not in the area)

53h: Program error (access window cannot be set)

37.10.9.7 Access Window Read

This command is used to check the set range of the access window.

When the MCU successfully obtains the access window range, the MCU sends the access window start address and end address that it read. If the SUM of the received command does not match, the MCU sends an error response.

Command	73h	01h	FFh	8Dh
Response	73h	05h		
	Access window start address LH	Access window start address HL	Access window end address LH	Access window end address HL
	FFh			
	SUM			

Access window start address LH (1 byte): Start address of the access window range (A15 to A8)

Access window start address HL (1 byte): Start address of the access window range (A23 to A16)

Access window end address LH (1 byte): End address of the access window range (A15 to A8)

Access window end address HL (1 byte): End address of the access window range (A23 to A16)

SUM (1 byte): Value that is calculated so the sum of response data is 00h

Error response	F3h	Error

Error (1 byte): Error code

11h: SUM error

37.10.9.8 Access Window Protection

This command is used to protect the settings for the access window used for area protection.

When the protection is successfully set, the MCU sends a response (06h). If the SUM of the received command does not match or an error occurs during the protect settings, the MCU sends an error response.

Command	7Ch	01h	FFh	84h

Response	06h

Error response	FCh	Error

Error (1 byte): Error code

11h: SUM error

53h: Program error (protection cannot be set)

37.10.9.9 Access Window Protection Flag Read

This command is used to check whether the settings for the access window are protected or not.

When the protection status is successfully obtained, the MCU sends the status. If the SUM of the received command does not match, the MCU sends an error response.

Command	7Bh	01h	FFh	85h

Response	7Bh	01h	Status	SUM

Status (1 byte): Protection status

01h: Protection is enabled

00h: Protection is disabled

SUM (1 byte): Value that is calculated so the sum of response data is 00h

Error response	FBh	Error

Error (1 byte): Error code

11h: SUM error

37.11 Serial Programmer Operation in Boot Mode (SCI Interface)

The following describes the procedure for the serial programmer to program/erase the user area and data area in boot mode (SCI Interface).

1. Automatically adjust the bit rate
2. Receive the MCU information*¹
3. Select the device and change the bit rate
4. Enter the program/erase host command wait state
5. Unlock boot mode ID code protection
6. Erase the user area and data area*², *³
7. Program the user area and data area*², *³
8. Check data in the user area*²
9. Check data in the data area*²
10. Set the access window in the user area*²
11. Protect the access window settings*²
12. Reset the MCU

Note 1. If the necessary information has been already received, step 2 can be skipped.

Note 2. Processing steps from 6 to 11 can be proceeded as necessary, and their order other than step 12 can be changed.

Note 3. When a timeout occurs or invalid response data is received, stop the operation and perform step 12 (reset the MCU).

Refer to section 37.10.5, Inquiry Commands, section 37.10.6, Setting Commands, section 37.10.7, ID Code Authentication Command, section 37.10.8, Program/Erase Commands, and section 37.10.9, Read-Check Commands for details on the commands used in the above steps 2 to 11.

37.11.1 Bit Rate Automatic Adjustment

The MCU measures the low width of data 00h that is sent from the serial programmer at 9,600 or 19,200 bps to automatically adjust the bit rate.

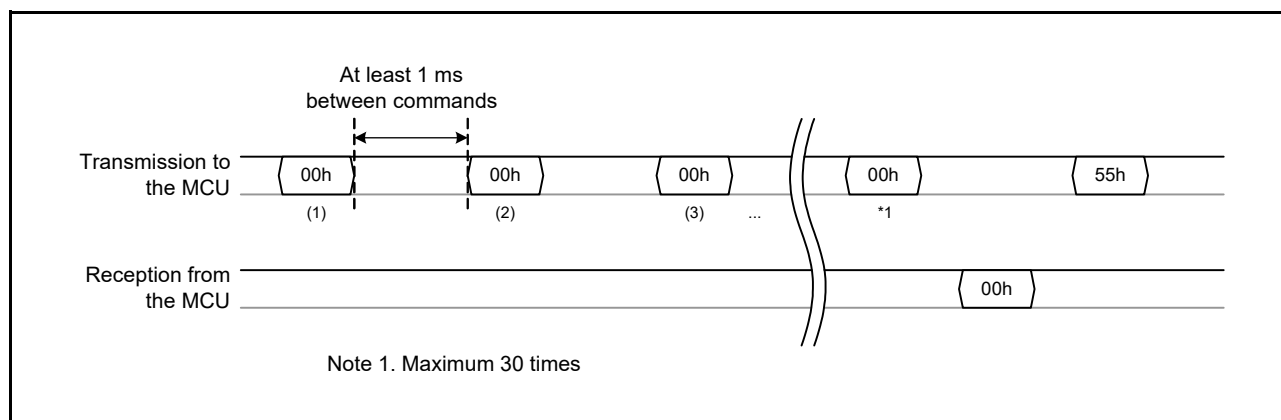


Figure 37.29 Transmit/Receive Data for Bit Rate Automatic Adjustment

After starting up in boot mode, wait for at least 400 ms and then send 00h to the MCU from the serial programmer. When the bit rate adjustment is completed, the MCU sends 00h to the programmer. When the programmer receives 00h, send 55h to the MCU from the programmer. When the programmer can not receive 00h, wait for at least 1 ms and send 00h to the MCU again. When the programmer fails to receive 00h even if it send 00h 30 times, restart the MCU in boot mode and perform the automatic adjustment for the bit rate again.

When the MCU receives 55h, the MCU sends E6h and enters the inquiry/setting command wait state. If the MCU fails to receive 55h, the MCU sends FFh. When the programmer receives FFh, restart the MCU in boot mode, and perform the automatic adjustment for the bit rate again.

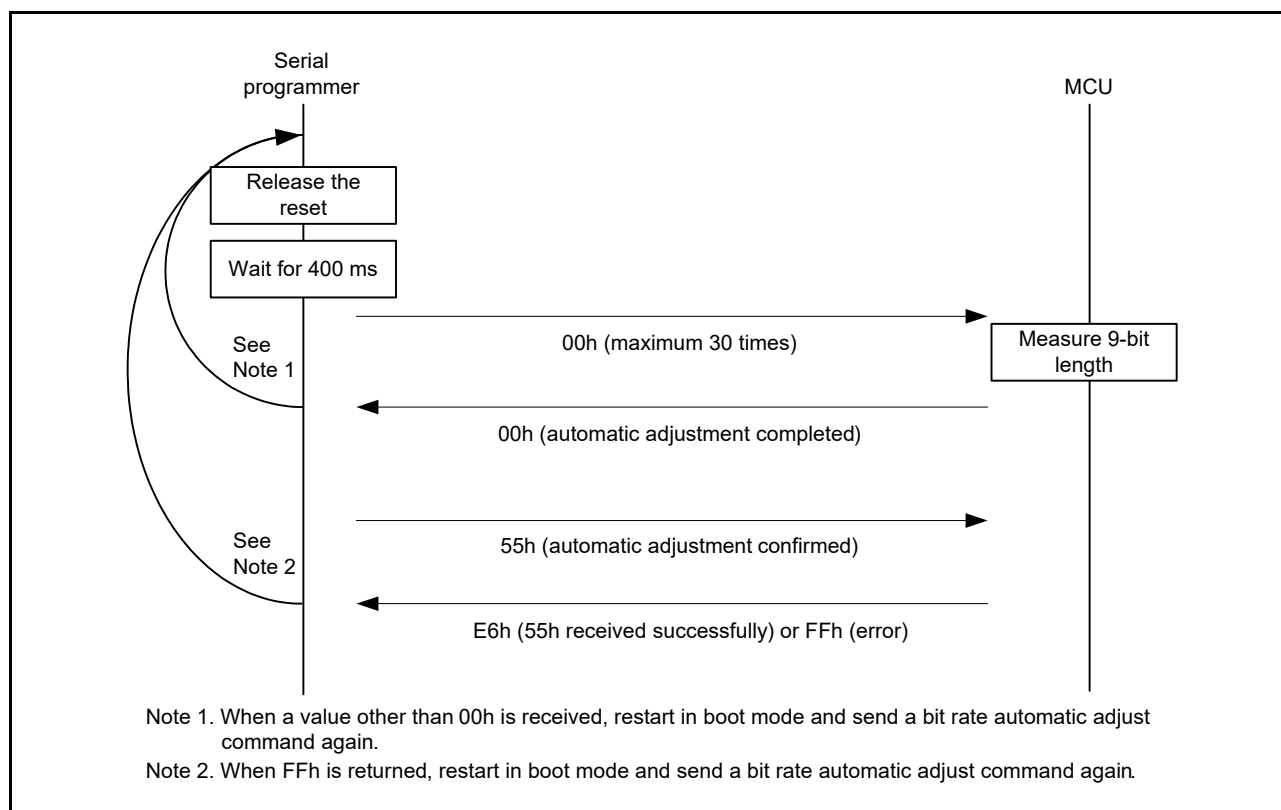


Figure 37.30 Bit Rate Automatic Adjustment Procedure

37.11.2 Receive the MCU Information

Procedure to send inquiry commands, and receive the information necessary to send setting commands, program/erase commands, and read-check commands is as follows.

- (1) Send a support device inquiry command (20h) to check what type of endianness the MCU supports. The MCU returns all device codes and series names that it supports.
- (2) Send a user area information inquiry command (25h) to check the start and end addresses of the user area. The MCU returns the start and end addresses of the user area.
- (3) Send a block information inquiry command (26h) to check the block configuration. The MCU returns the start address, the size of one block, and the number of blocks for the user area and data area.
- (4) Send a data area information inquiry command (2Bh) to check the start and end addresses of the data area. The MCU returns the start and end addresses of the data area.

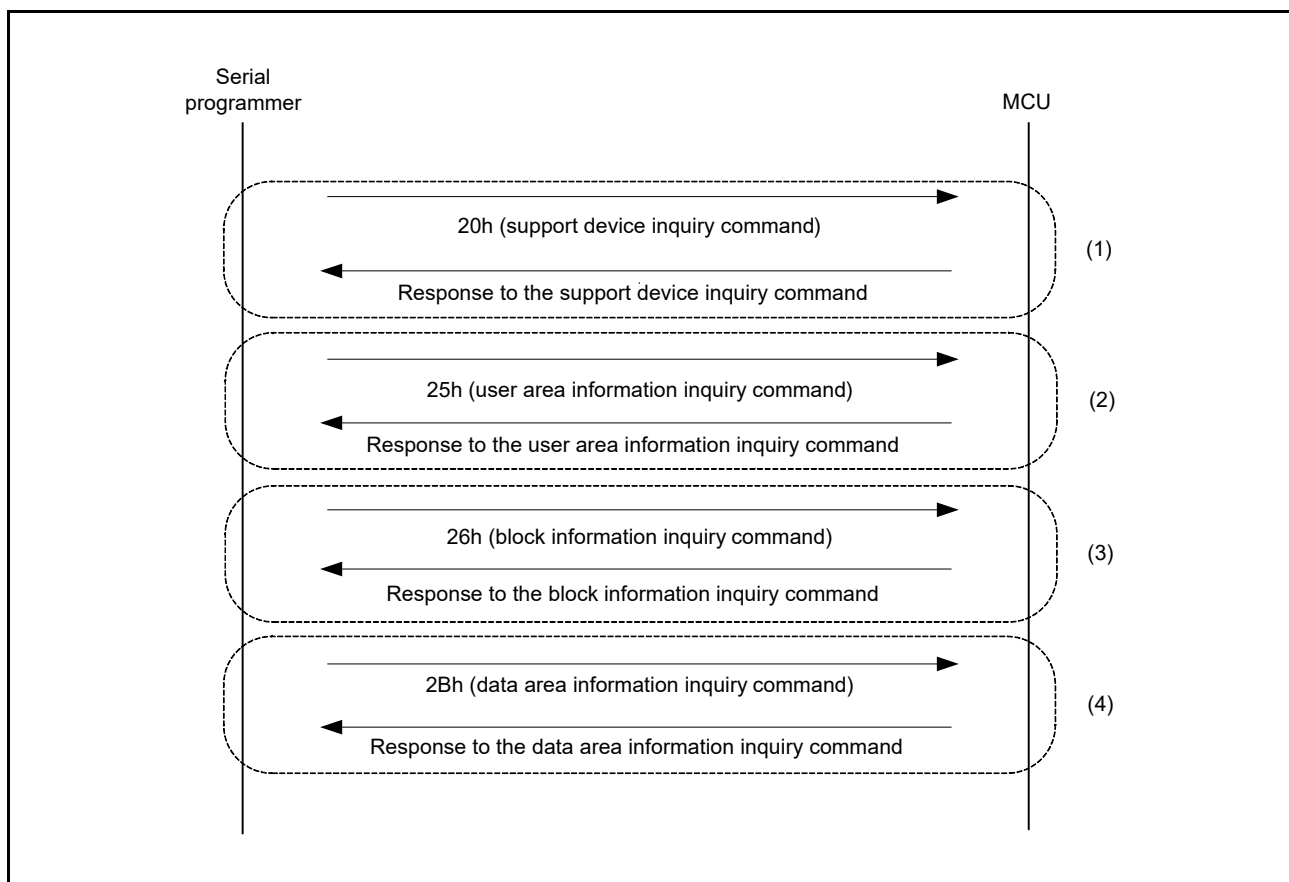


Figure 37.31 Procedure to Receive the MCU Information

37.11.3 Select the Device and Change the Bit Rate

Procedure to select the device to connect with the serial programmer and to change the bit rate for communication is as follows.

- (1) Send the device select command (10h). Select the device code according to the endian of developed software.
- (2) Send the operating frequency select command (3Fh) to change the communication bit rate from 9,600 or 19,200 bps.

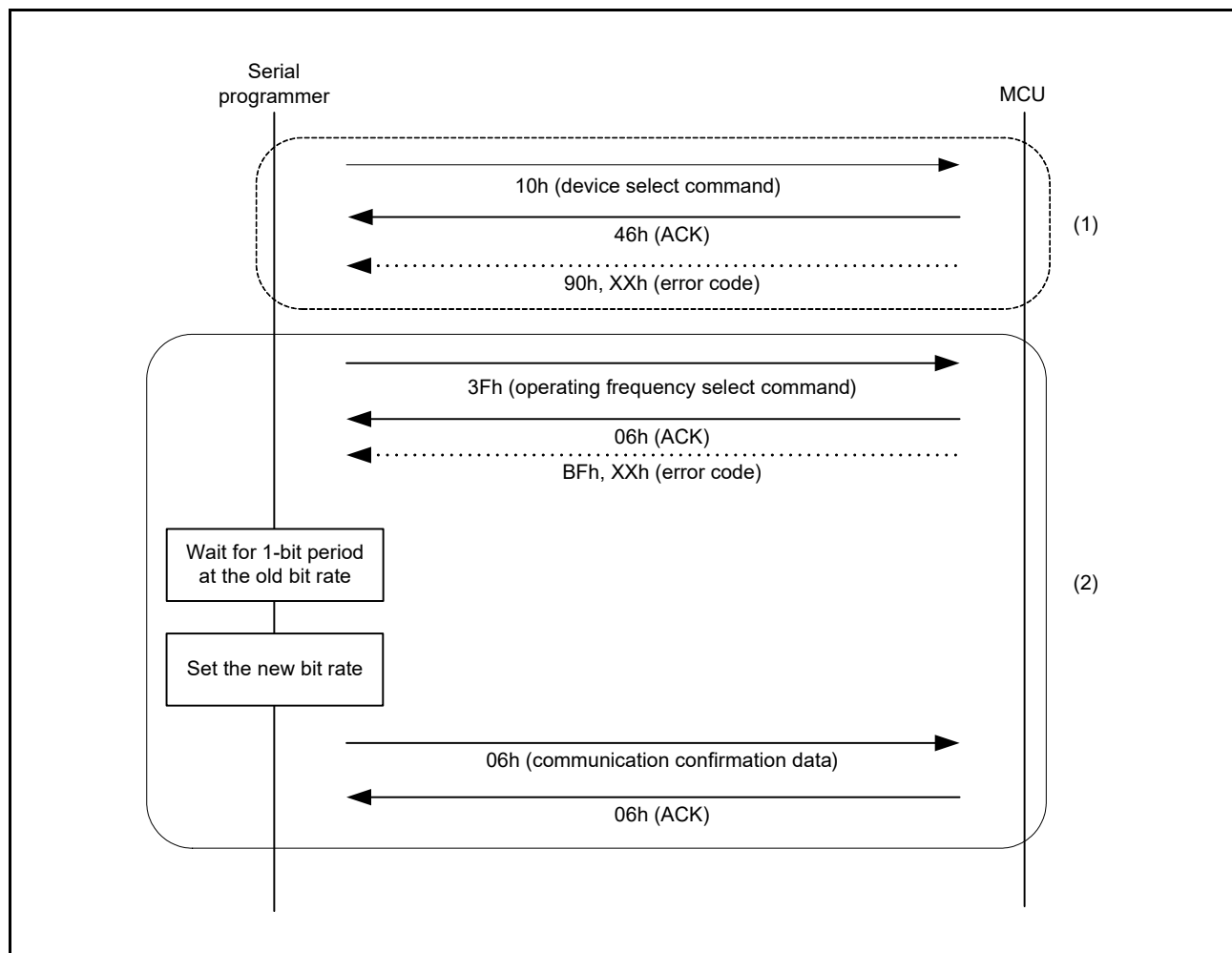


Figure 37.32 Procedure to Select the Device and Change the Bit Rate

37.11.4 Enter the Program/Erase Host Command Wait State

Send the program/erase host command wait state transition command to perform program/erase operations. The MCU sends a response according to whether boot mode ID code protection is enabled or disabled.

- (1) When boot mode ID code protection is disabled, the MCU sends a response (06h), and enters the program/erase host command wait state. Use the serial programmer to start from the operation described in section 37.11.6, Erase the User Area and Data Area.
- (2) When the boot mode ID code protection is enabled, the MCU sends a response (16h), and enters the ID code authentication wait state. Use the serial programmer to start from the operation described in section 37.11.5, Unlock Boot Mode ID Code Protection.

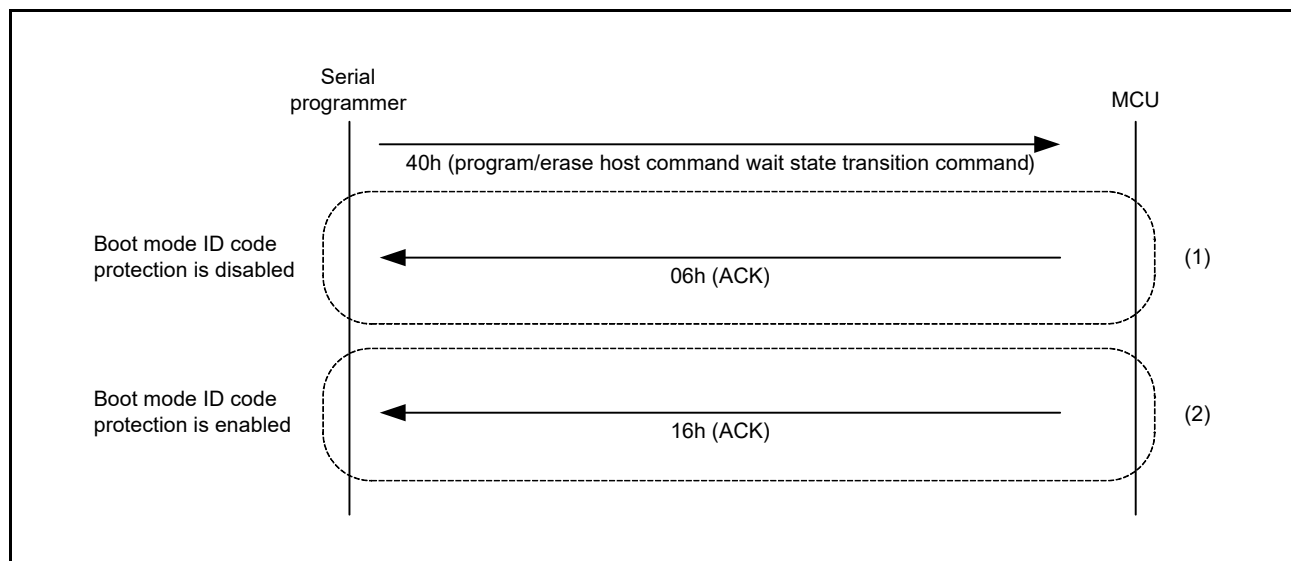


Figure 37.33 Procedure to Transition to the Program/Erase Host Command Wait State

37.11.5 Unlock Boot Mode ID Code Protection

Send the ID code check command to unlock boot mode ID code protection.

- (1) When ID codes match, the MCU enters the program/erase host command wait state. Data in the user area and data area are not erased. Use the serial programmer to start from the operation described in [section 37.11.6, Erase the User Area and Data Area](#).
- (2) If ID codes do not match consecutively, the MCU remains in the boot mode ID code authentication state. Reset the MCU, and then use the serial programmer to start again from [section 37.11.1, Bit Rate Automatic Adjustment](#).

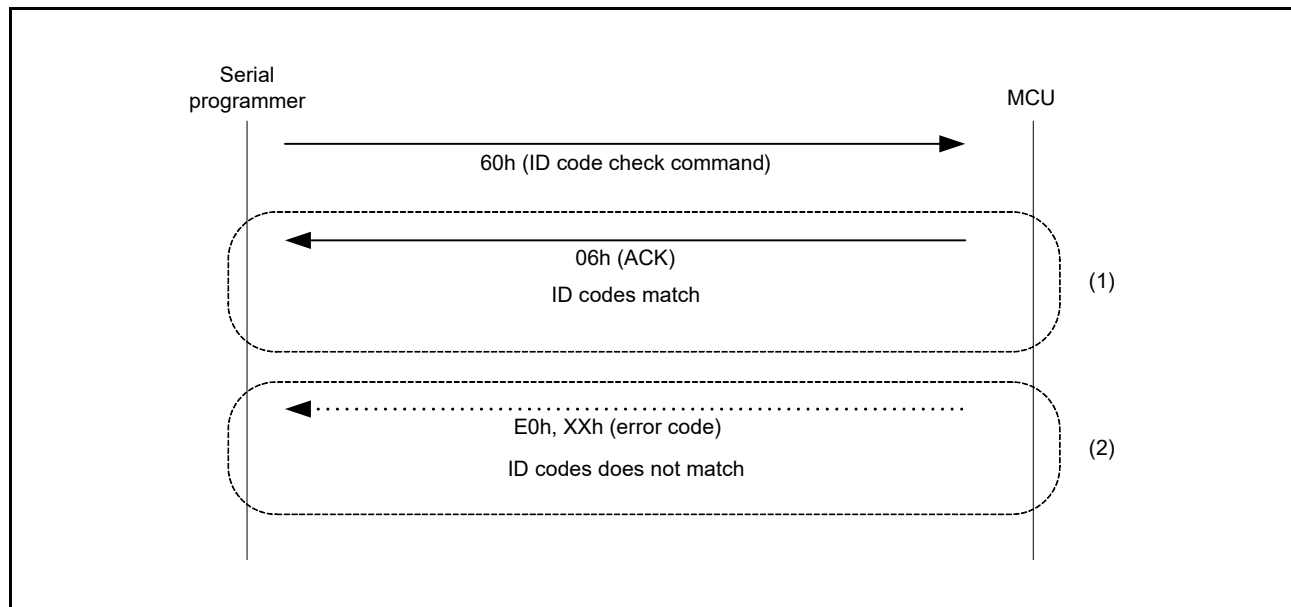


Figure 37.34 Procedure to Unlock ID Code Protection

37.11.6 Erase the User Area and Data Area

Procedure to erase blocks that are programmed in the user area and data area to program a user program and data is as follows.

- (1) Send an erase preparation command (48h).
- (2) Send a block erase command (59h).
- (3) To place the MCU in the program/erase host command wait state, send a block erase command for ending the erasure (59h 04h FFh FFh FFh FFh A7h).

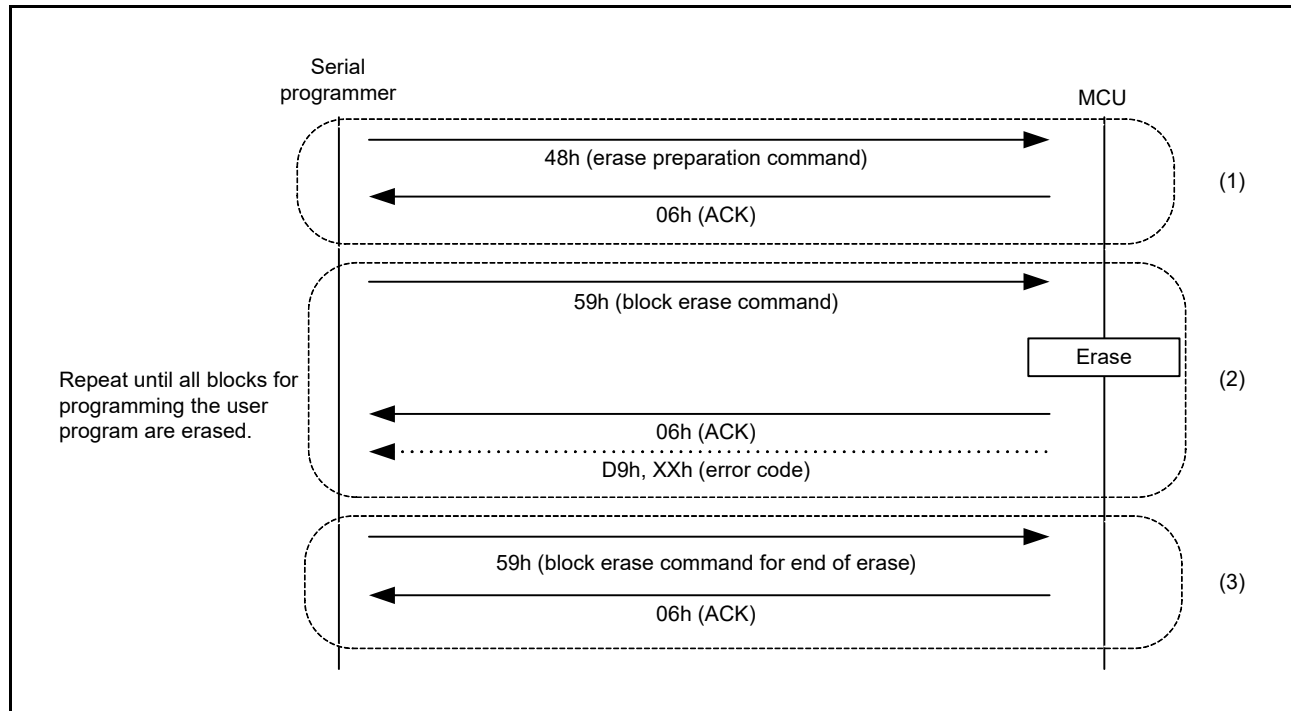


Figure 37.35 Procedure to Erase the User Area and Data Area

37.11.7 Program the User Area and Data Area

Procedure to program a user program and data in the user area and data area is as follows.

- (1) Send the user/data area program preparation command (43h).
- (2) Send the program command (50h) or the data area program command (51h).
- (3) To place the MCU in the program/erase host command wait state, send the program command (50h FFh FFh FFh FFh B4h) or the data area program command (51h FFh FFh FFh FFh 00h B3h) for ending the programming.

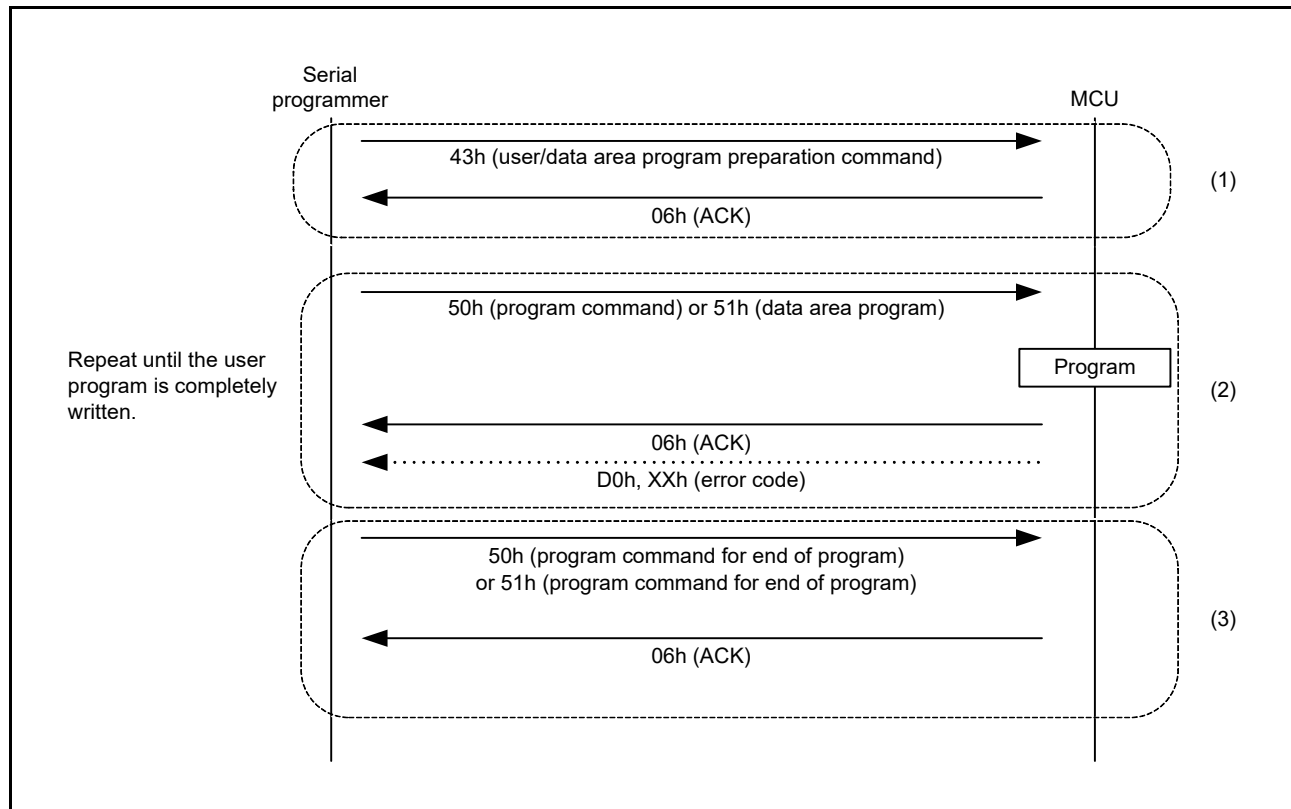


Figure 37.36 Procedure to Program the User Area and Data Area

37.11.8 Check Data in the User Area

Procedure to read and check, checksum, and blank check the user area to check the programmed data in the user area is as follows.

- (1) The read and check operation is used to read data in the user area and compare the read data with the programmed data to check if the program operation is performed successfully. Send a memory read command (52h) to read data in the user area.
- (2) Send the user area checksum command (4Bh) to check program data using the checksum of user area.
- (3) Send a user area blank check command (4Dh) to check if the user area has data.

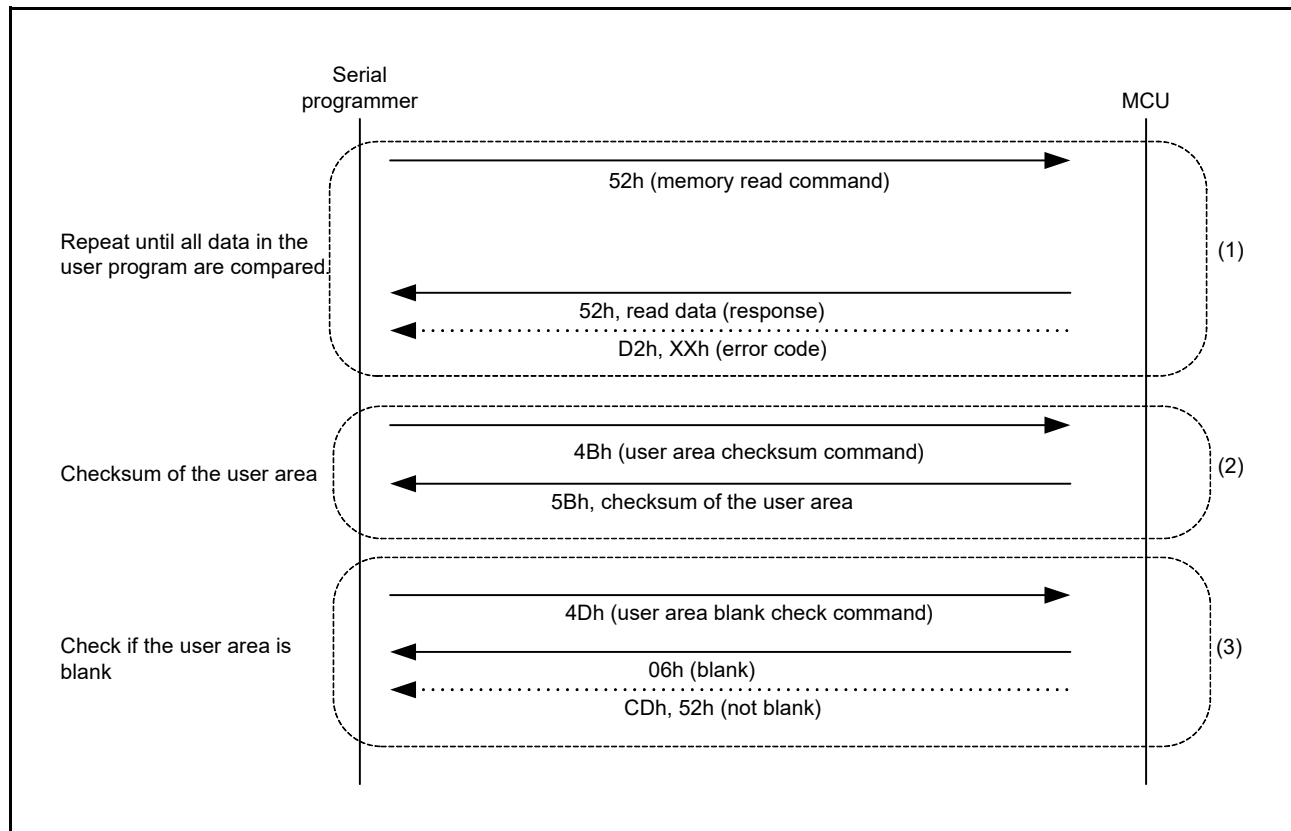


Figure 37.37 Procedure to Check Data in the User Area

37.11.9 Check Data in the Data Area

Procedure to read and check, checksum, and blank check the data area to check the programmed data in the data area is as follows.

- (1) The read and check operation is used to read data in the data area and compare the read data with the programmed data to check if the program operation is performed successfully. Send a memory read command (52h) to read data in the data area.
- (2) Send the data area checksum command (61h) to check program data using the checksum of data area.
- (3) Send the data area blank check command (62h) to check if the data area has data.

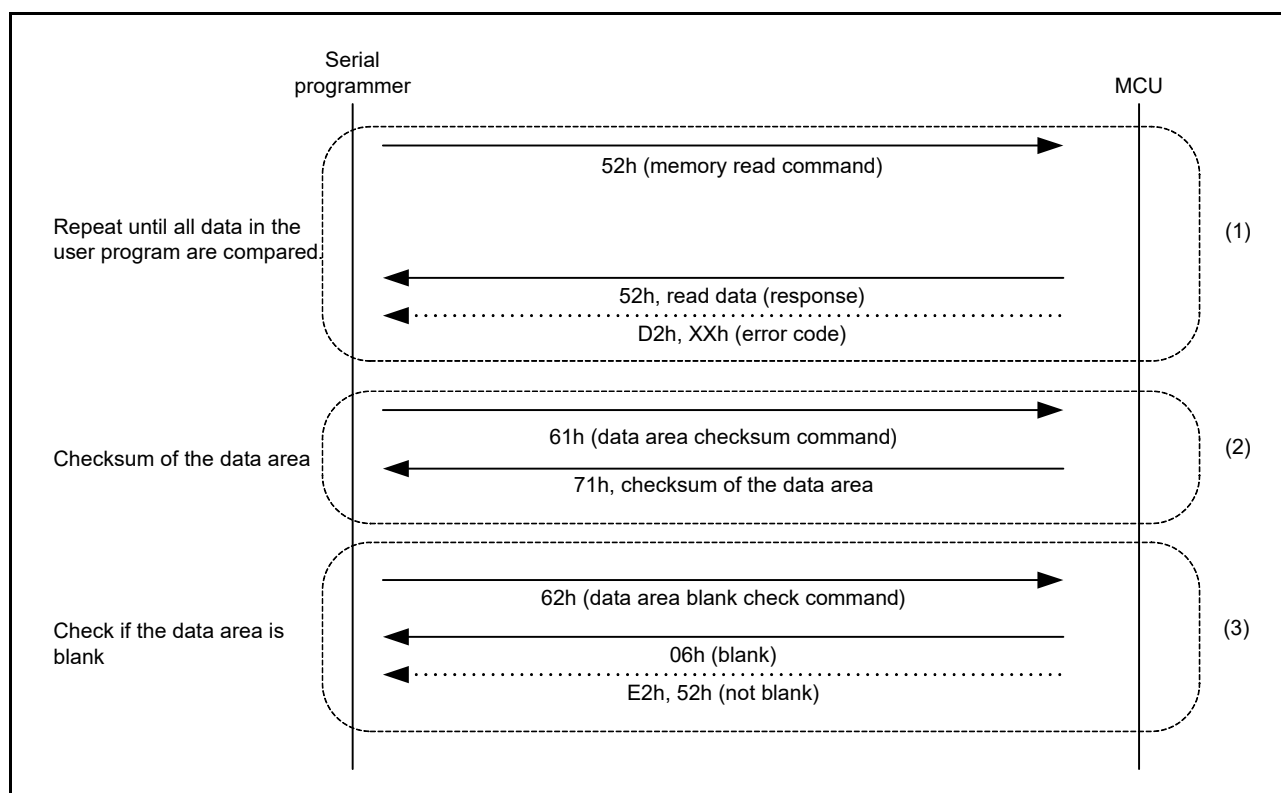


Figure 37.38 Procedure to Check Data in the Data Area

37.11.10 Set the Access Window in the User Area

Procedure to set the access window to avoid unintentionally rewriting the user area during the self-programming is as follows.

- (1) Send the access window program command (74h) to set the access window settings.
- (2) Send the access window read command (73h) to confirm the access window settings.

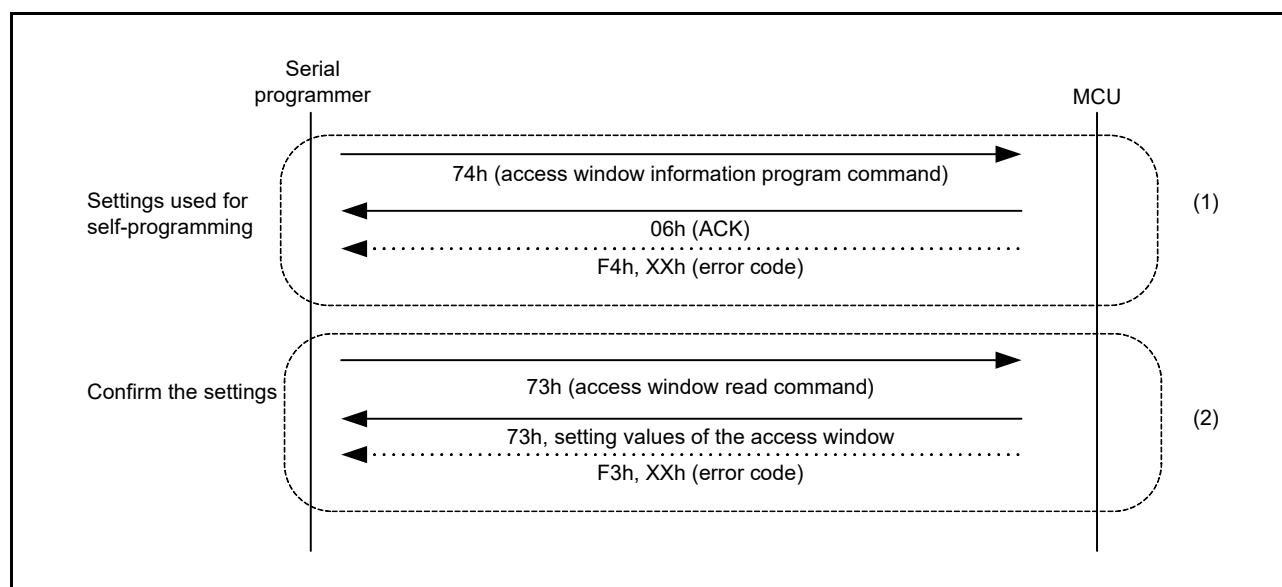


Figure 37.39 Procedure to Set the Access Window in the User Area

37.11.11 Protect the Access Window Settings

Procedure to protect the access window settings to avoid unintentionally rewriting the addresses of the access window during the self-programming is as follows.

- (1) Send the access window protection command “7Ch” to protect the access window settings.
- (2) Send the access window protection flag read command “7Bh” to confirm the setting of the protection.

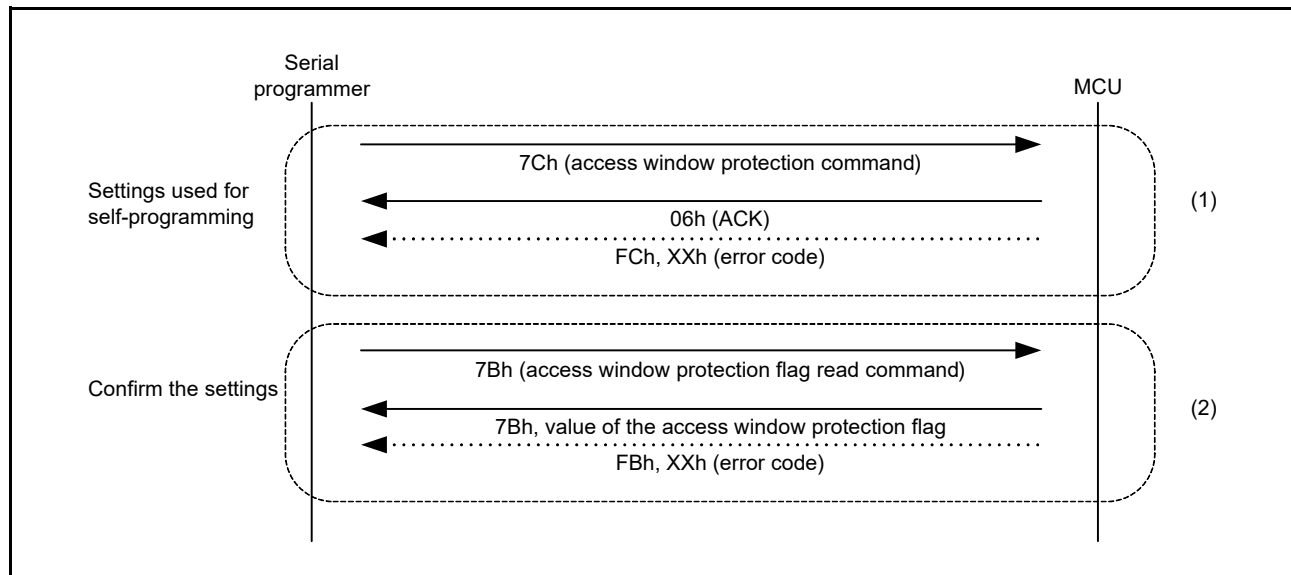


Figure 37.40 Procedure to Protect the Access Window

37.12 Rewriting by Self-Programming

37.12.1 Overview

The MCU supports rewriting of the flash memory by the user program. The ROM and E2 DataFlash can be rewritten by preparing a routine to rewrite the flash memory (flash rewrite routine) in the user program.

When rewriting the E2 DataFlash, the BGO can be used to execute the flash rewrite routine on the ROM. The E2 DataFlash can also be rewritten by executing the flash rewrite routine that is transferred on the RAM in advance.

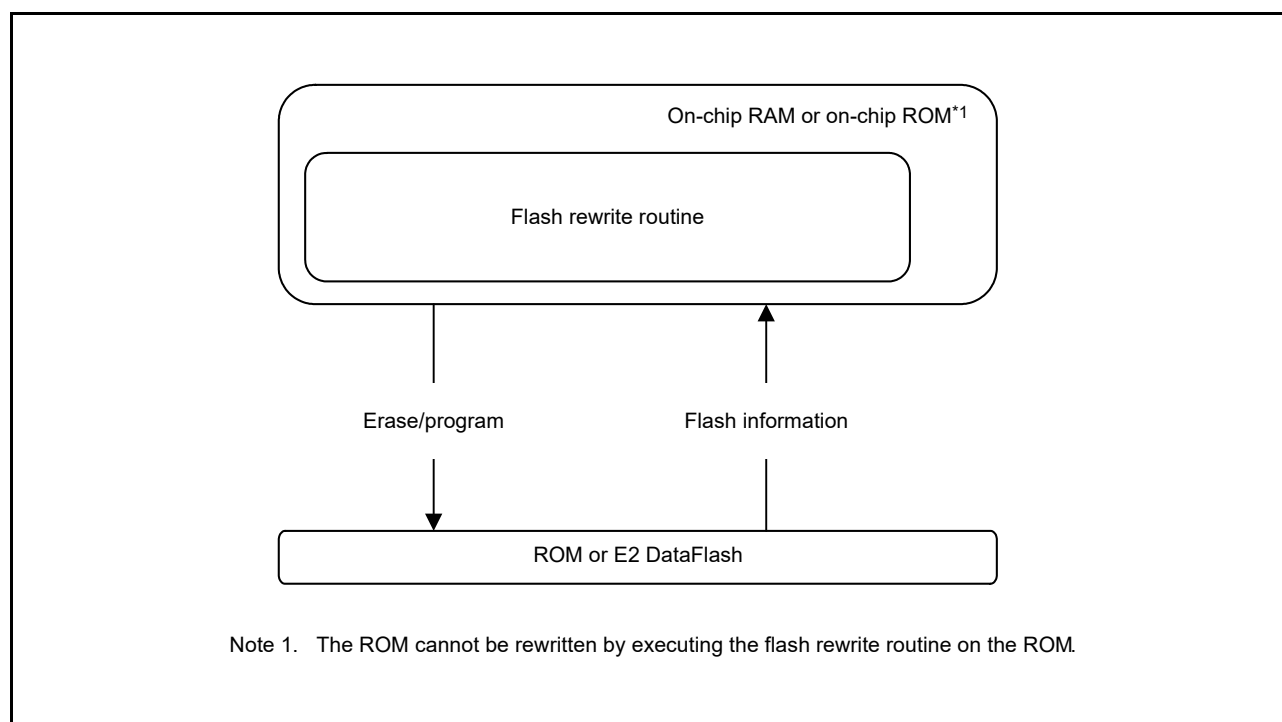


Figure 37.41 Self-Programming Overview

37.13 Usage Notes

(1) Access the Block Where Erase Operation is Forcibly Stopped

When forcibly stopping an erase operation, data in the block where the erase operation is aborted is undefined. To avoid malfunctions caused by reading undefined data, do not execute instructions or read data in the block where an erase operation is forcibly stopped.

(2) Processing After Forced Stop of Erase Operation

When an erase operation is forcibly stopped, issue a block erase command again to the same block.

(3) Additional Programming Disabled

The same address cannot be programmed more than once. When programming an area that has been already programmed, erase the area first.

(4) Reset during Program/Erase

If inputting a reset from the RES# pin, release the reset after reset input time of at least tRESW (refer to section 38, Electrical Characteristics) within the range of the operating voltage defined in the electrical characteristics. The IWDt reset and software reset can be used regardless of tRESW.

(5) Location of Interrupt Vectors and Exception Vectors during Program/Erase Operation

When an interrupt or an exception occurs during a program/erase operation, the vector may be fetched from the ROM. To avoid fetching the vector from the ROM, allocate the interrupt vector table and exception vector table to the area other than the ROM with the INTB and EXTB registers in the CPU.

(6) Abnormal Termination during Program/Erase

When the voltage exceeds the range of the operating voltage during a program/erase operation or when a program/erase operation is not completed successfully due to a reset or prohibited actions described in (7), erase the area again.

(7) Actions Prohibited during Program/Erase

To prevent the damage to the flash memory, comply with the following instructions.

- Do not use the MCU power supply that is outside the operating voltage range.
- Do not update the value of the OPCCR.OPCM[2:0] bits.
- Do not change the clock source select bit in the SCKCR3 register.
- Do not change the division ratio of the flash interface clock (FCLK).
- Do not place the MCU in deep sleep mode or software standby mode.
- Do not access the E2 DataFlash during a program/erase operation to the ROM.
- Do not change the DFLCTL.DFLEN bit value during a program/erase operation to the E2 DataFlash.

(8) FCLK during Program/Erase

For programming/erasure by self-programming, set the frequency of the FlashIF clock (FCLK), and specify an integer FCLK frequency (MHz) in FISR.PCKA[5:0] bits. Note that when the FCLK is 4 to 32 MHz, a rounded-up value should be set for a non-integer frequency such as 12.5 MHz (i.e. 12.5 MHz should be set rounded up to 13 MHz). If the FCLK is equal to or less than 4 MHz, only 1, 2, 3, or 4 MHz can be used.

37.14 Usage Notes in Boot Mode

(1) Notes on Communication Errors in Boot Mode

When communication with the MCU cannot be performed properly, reset and start up in boot mode again.

(2) Notes on Option-Setting Memory in Boot Mode

The settings of option function select register 0 (OFS0), option function select register 1 (OFS1), and endian select register (MDE) are disabled in boot mode.

(3) Notes on Switching the Start-Up Area

Switch the start-up area by self-programming.

38. Electrical Characteristics

38.1 Absolute Maximum Ratings

Table 38.1 Absolute Maximum Ratings

Item		Symbol	Value	Unit
Power supply voltage		VCC	−0.3 to +6.5	V
Input voltage	5-V tolerant pins: P97, PB0	V_{in}	−0.3 to +6.5	V
	Pins other than above		−0.3 to VCC + 0.3	V
Reference power supply voltage		VREFH0	−0.3 to VCC + 0.3	V
Output sink current	Per pin	I_{IO}	−16	mA
	Total of all pins ($T_j = -40$ to 125°C)	ΣI_{IO}	−164	mA
	Total of all pins ($T_j = -40$ to 140°C)		−97	mA
Output source current	Per pin	I_{IO}	16	mA
	Total of all pins ($T_j = -40$ to 125°C)	ΣI_{IO}	76	mA
	Total of all pins ($T_j = -40$ to 140°C)		45	mA
Junction temperature		T_j	−40 to +140	$^{\circ}\text{C}$
Storage temperature		T_{stg}	−65 to +150	$^{\circ}\text{C}$

Caution: Permanent damage to the MCU may result if absolute maximum ratings are exceeded.

To preclude any malfunctions due to noise interference, insert capacitors of high frequency characteristics between the VCC and VSS pins and between the VREFH0 and VREFL0 pins. Place capacitors of about 0.1 μF as close as possible to every power supply pin and use the shortest and heaviest possible traces.

Connect the VCL pin to a VSS pin via a 4.7 μF capacitor. The capacitor must be placed close to the pin, refer to section 38.15.1, Connecting VCL Capacitor and Bypass Capacitors.

Do not input signals or an I/O pull-up power supply to pins other than 5-V tolerant pins while the device is not powered.

The current injection that results from input of such a signal or I/O pull-up may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements.

Even if −0.3 to +6.5 V is input to 5-V tolerant pins, it will not cause problems such as damage to the MCU.

38.2 Recommended Operating Conditions

Table 38.2 Recommended Operating Conditions (1)

Item		Symbol	Min.	Typ.	Max.	Unit
Power supply voltages		VCC	2.7	—	5.5	V
		VSS	—	0	—	
Reference power supply voltages		VREFH0	2.7	—	VCC	V
		VREFL0	—	0	—	
Input voltage	Ports for 5 V tolerant: P97, PB0	V_{in}	−0.3	—	5.8	V
	PGAGND		—	0	—	
	Ports other than above		−0.3	—	VCC + 0.3	
Operating temperature	G version	T_{opr}	−40	—	105	°C
	M version		−40	—	125	
Junction temperature	G version	T_j	−40	—	125	°C
	M version		−40	—	140	

Table 38.3 Recommended Operating Conditions (2)

Item	Symbol	Value
Decoupling capacitance for stabilizing the internal voltage	C_{VCL}	4.7μF ± 30%*1

Note 1. Use a multilayer ceramic capacitor with a nominal capacitance of 4.7 μF, for which the sum of the capacitance tolerance and change in the capacitance under the usage conditions will be no greater than ±30%.

38.3 Test Conditions

Unless otherwise specified, minimum and maximum values are guaranteed by either design simulation, characterization results or test in production.

Supported peripheral functions and pins differ from one product name to another.

Unless otherwise specified, the electrical characteristics of the MCU are defined under the following conditions:

- $V_{CC} = 2.7$ to 5.5 V
- $V_{REFH0} = 2.7$ V to V_{CC}
- $V_{SS} = V_{REFL0} = 0$ V
- $T_a = T_{opr} = -40$ to 125°C

Unless otherwise specified, typical values are those measured at room temperature of 25°C , and when $V_{CC} = V_{REFH0} = 5.0$ V.

Figure 38.1 shows the timing conditions.

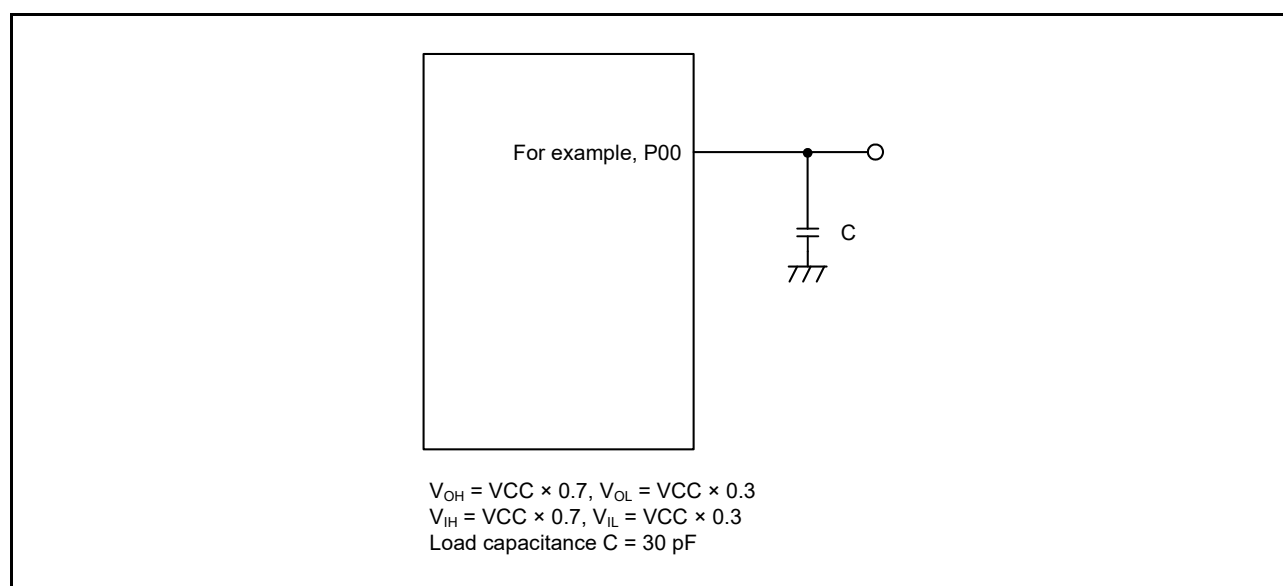


Figure 38.1 Input or Output Timing Measurement Conditions

38.4 DC Characteristics

38.4.1 Characteristics of the Input and Output Pins

Table 38.4 I/O Input Voltage Characteristics

	Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Schmitt trigger input voltage	RIIC input pins (P97, PB0) (except for SMBus)	V_{IH}	$0.7 \times VCC$	—	—	V	
		V_{IL}	—	—	$0.3 \times VCC$		
		ΔV_T	$0.05 \times VCC$	—	—		
	RIIC input pins (P26, P31) (except for SMBus)	V_{IH}	$0.8 \times VCC$	—	—		
		V_{IL}	—	—	$0.2 \times VCC$		
		ΔV_T	$0.1 \times VCC$	—	—		
	IRQ input pin, MTU input pin, GPTW input pin, POE3 input pin, POEG input pin, TMR input pin, SCI input pin, CAC input pin, ADTRG# input pin, RES#, NMI, MD	V_{IH}	$0.8 \times VCC$	—	—		
		V_{IL}	—	—	$0.2 \times VCC$		
		ΔV_T	$0.1 \times VCC$	—	—		
Input level voltage (except for schmitt trigger input pins)	EXTAL (external clock input)	V_{IH}	$0.8 \times VCC$	—	—	V	
		V_{IL}	—	—	$0.2 \times VCC$		
	RIIC input pins (P97, PB0, P26, P31) (SMBus)	V_{IH}	2.2	—	—		VCC = 3.6 to 5.5 V
			2.0	—	—		VCC = 2.7 to 3.6 V
		V_{IL}	—	—	0.8		VCC = 3.6 to 5.5 V
			—	—	0.5		VCC = 2.7 to 3.6 V
	General input pins other than those above	V_{IH}	$0.8 \times VCC$	—	—		
		V_{IL}	—	—	$0.2 \times VCC$		

Table 38.5 I/O Input Leakage Current

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input leakage current	RES#, PE2	$ I_{in} $	—	—	1.0	μA	$V_{in} = 0\text{ V}$, VCC

Table 38.6 I/O Three-state Leakage Current

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Three-state leakage current (off-state)	P97, PB0	$ I_{TSI} $	—	—	1.0	μA	$V_{in} = 0\text{ V}$, 5.8 V
	P13, P14		—	—	1.0		$V_{in} = 0\text{ V}$, VCC
	Other than above		—	—	0.2		$V_{in} = 0\text{ V}$, VCC

Table 38.7 I/O Input Capacitance

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input capacitance	PE2	C_{in}	—	—	30	pF	$V_{in} = 0\text{ V}$, $f = 1\text{ MHz}$, $T_a = 25^\circ\text{C}$
	Other than above		—	—	15		

Table 38.8 I/O Input Pull-up Resistor

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input pull-up resistor	All ports (except for PE2)	R_U	10	20	50	$\text{k}\Omega$	$V_{in} = 0\text{ V}$

Table 38.9 I/O Output Voltage Characteristics

Item			Symbol	Min.	Max.	Unit	Test Conditions
Low-level output voltage	RIIC pins	P97, PB0	V _{OL}	—	0.6	V	I _{OL} = 6.0 mA
		P26, P31		—	0.5		I _{OL} = 2.0 mA
	Ports other than above			—	0.5		I _{OL} = 2.0 mA
High-level output voltage	All output ports		V _{OH}	VCC – 0.5	—	V	I _{OH} = –2.0 mA

Table 38.10 Normal I/O Pin VOH Voltage Characteristics (Reference Values)Conditions: $V_{CC} = 3.3\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = 25^\circ\text{C}$

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Output high level voltage	All output pins	V_{OH}	—	$V_{CC} - 0.02$	—	V	$I_{OH} = -0.5\text{ mA}$
			—	$V_{CC} - 0.05$	—		$I_{OH} = -1.0\text{ mA}$
			—	$V_{CC} - 0.10$	—		$I_{OH} = -2.0\text{ mA}$
			—	$V_{CC} - 0.22$	—		$I_{OH} = -4.0\text{ mA}$

Table 38.11 Normal I/O Pin VOH Voltage Characteristics (Reference Values)Conditions: $V_{CC} = 5.0\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = 25^\circ\text{C}$

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Output high level voltage	All output pins	V_{OH}	—	$V_{CC} - 0.02$	—	V	$I_{OH} = -0.5\text{ mA}$
			—	$V_{CC} - 0.04$	—		$I_{OH} = -1.0\text{ mA}$
			—	$V_{CC} - 0.08$	—		$I_{OH} = -2.0\text{ mA}$
			—	$V_{CC} - 0.15$	—		$I_{OH} = -4.0\text{ mA}$

Table 38.12 Normal I/O Pin VOH Voltage Characteristics (Reference Values)Conditions: $V_{CC} = 3.3\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = 25^\circ\text{C}$

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Output low voltage	All output pins	V_{OL}	—	0.01	—	V	$I_{OL} = 0.5\text{ mA}$
			—	0.02	—		$I_{OL} = 1.0\text{ mA}$
			—	0.04	—		$I_{OL} = 2.0\text{ mA}$
			—	0.08	—		$I_{OL} = 4.0\text{ mA}$
			—	0.17	—		$I_{OL} = 8.0\text{ mA}$

Table 38.13 Normal I/O Pin VOH Voltage Characteristics (Reference Values)Conditions: $V_{CC} = 5.0\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = 25^\circ\text{C}$

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Output low voltage	All output pins	V_{OL}	—	0.01	—	V	$I_{OL} = 0.5\text{ mA}$
			—	0.01	—		$I_{OL} = 1.0\text{ mA}$
			—	0.03	—		$I_{OL} = 2.0\text{ mA}$
			—	0.06	—		$I_{OL} = 4.0\text{ mA}$
			—	0.12	—		$I_{OL} = 8.0\text{ mA}$

38.4.2 Operating and Standby Currents

Table 38.14 Operating Current in the High-Speed Operating Mode*1 (1/3)

Item			Symbol	Typ.	Max.	Unit	Test Conditions
Normal operating mode	Supply of the clock signal to the peripheral modules is stopped.	$T_a = 25^{\circ}\text{C}$	I_{CC}	2.4	—	mA	ICLK = 48 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^{\circ}\text{C}$			—		
		$T_a = 105^{\circ}\text{C}$			—		
		$T_a = 125^{\circ}\text{C}$			—		
		$T_a = 25^{\circ}\text{C}$		1.8	—		ICLK = 32 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^{\circ}\text{C}$			—		
		$T_a = 105^{\circ}\text{C}$			—		
		$T_a = 125^{\circ}\text{C}$			—		
		$T_a = 25^{\circ}\text{C}$		1.5	—		ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^{\circ}\text{C}$			—		
		$T_a = 105^{\circ}\text{C}$			—		
		$T_a = 125^{\circ}\text{C}$			—		
		$T_a = 25^{\circ}\text{C}$		1.0	—		ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^{\circ}\text{C}$			—		
		$T_a = 105^{\circ}\text{C}$			—		
		$T_a = 125^{\circ}\text{C}$			—		
	The clock signal is being supplied to the peripheral modules.	$T_a = 25^{\circ}\text{C}$		12.8	25.3		ICLK = 48 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 48 MHz
		$T_a = 85^{\circ}\text{C}$			25.0		
		$T_a = 105^{\circ}\text{C}$			24.9		
		$T_a = 125^{\circ}\text{C}$			24.6		
		$T_a = 25^{\circ}\text{C}$		9.9	19.8		ICLK = 32 MHz Clock Source = HOCO FCLK, PCLKB = 32 MHz PCLKD = 64 MHz
		$T_a = 85^{\circ}\text{C}$			19.6		
		$T_a = 105^{\circ}\text{C}$			19.5		
		$T_a = 125^{\circ}\text{C}$			19.2		
		$T_a = 25^{\circ}\text{C}$		7.3	15.1		ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 24 MHz
		$T_a = 85^{\circ}\text{C}$			14.9		
		$T_a = 105^{\circ}\text{C}$			14.8		
		$T_a = 125^{\circ}\text{C}$			14.6		
		$T_a = 25^{\circ}\text{C}$		3.7	8.6		ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 8 MHz
		$T_a = 85^{\circ}\text{C}$			8.3		
		$T_a = 105^{\circ}\text{C}$			8.3		
		$T_a = 125^{\circ}\text{C}$			8.2		
Sleep mode	Peripheral module clocks are stopped	$T_a = 25^{\circ}\text{C}$		1.3	—		ICLK = 48 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^{\circ}\text{C}$			—		
		$T_a = 105^{\circ}\text{C}$			—		
		$T_a = 125^{\circ}\text{C}$			—		
		$T_a = 25^{\circ}\text{C}$		1.0	—		ICLK = 32 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^{\circ}\text{C}$			—		
		$T_a = 105^{\circ}\text{C}$			—		
		$T_a = 125^{\circ}\text{C}$			—		

Table 38.14 Operating Current in the High-Speed Operating Mode*1 (2/3)

Item			Symbol	Typ.	Max.	Unit	Test Conditions
Sleep mode	Peripheral module clocks are stopped	$T_a = 25^\circ\text{C}$	I_{CC}	0.8	—	mA	ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
		$T_a = 25^\circ\text{C}$		0.7	—		ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
	Peripheral module clocks are supplied	$T_a = 25^\circ\text{C}$	7.4	—			ICLK = 48 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 48 MHz
		$T_a = 85^\circ\text{C}$		—			
		$T_a = 105^\circ\text{C}$		—			
		$T_a = 125^\circ\text{C}$		—			
		$T_a = 25^\circ\text{C}$	6.2	—			ICLK = 32 MHz Clock Source = HOCO FCLK, PCLKB = 32 MHz PCLKD = 64 MHz
		$T_a = 85^\circ\text{C}$		—			
		$T_a = 105^\circ\text{C}$		—			
		$T_a = 125^\circ\text{C}$		—			
		$T_a = 25^\circ\text{C}$	4.4	—			ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 24 MHz
		$T_a = 85^\circ\text{C}$		—			
		$T_a = 105^\circ\text{C}$		—			
		$T_a = 125^\circ\text{C}$		—			
		$T_a = 25^\circ\text{C}$	2.6	—			ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 8 MHz
		$T_a = 85^\circ\text{C}$		—			
		$T_a = 105^\circ\text{C}$		—			
		$T_a = 125^\circ\text{C}$		—			
Deep sleep mode	Peripheral module clocks are stopped	$T_a = 25^\circ\text{C}$	0.9	—			ICLK = 48 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^\circ\text{C}$		—			
		$T_a = 105^\circ\text{C}$		—			
		$T_a = 125^\circ\text{C}$		—			
		$T_a = 25^\circ\text{C}$	0.7	—			ICLK = 32 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^\circ\text{C}$		—			
		$T_a = 105^\circ\text{C}$		—			
		$T_a = 125^\circ\text{C}$		—			
		$T_a = 25^\circ\text{C}$	0.6	—			ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^\circ\text{C}$		—			
		$T_a = 105^\circ\text{C}$		—			
		$T_a = 125^\circ\text{C}$		—			
		$T_a = 25^\circ\text{C}$	0.6	—			ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times$ 1/64
		$T_a = 85^\circ\text{C}$		—			
		$T_a = 105^\circ\text{C}$		—			
		$T_a = 125^\circ\text{C}$		—			

Table 38.14 Operating Current in the High-Speed Operating Mode*1 (3/3)

Item			Symbol	Typ.	Max.	Unit	Test Conditions
Deep sleep mode	Peripheral module clocks are supplied	T _a = 25°C	I _{CC}	6.6	—	mA	ICLK = 48 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 48 MHz
		T _a = 85°C			—		
		T _a = 105°C			—		
		T _a = 125°C			—		
		T _a = 25°C		5.7	—		ICLK = 32 MHz Clock Source = HOCO FCLK, PCLKB = 32 MHz PCLKD = 64 MHz
		T _a = 85°C			—		
		T _a = 105°C			—		
		T _a = 125°C			—		
		T _a = 25°C		4.0	—		ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 24 MHz
		T _a = 85°C			—		
		T _a = 105°C			—		
		T _a = 125°C			—		
		T _a = 25°C		2.4	—		ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 8 MHz
		T _a = 85°C			—		
		T _a = 105°C			—		
		T _a = 125°C			—		
Increase during background operation*2				2.1	—		

Note 1. Operating current values do not include output charge/discharge current from all pins. The values apply when internal pull-up resistors are in the off state.

Note 2. This is the increase for programming or erasure of the ROM or E2 DataFlash during program execution. The operating current during background operation is not included.

Table 38.15 Operating Current in Middle-Speed Operating Mode*1 (1/2)

Item			Symbol	Typ.	Max.	Unit	Test Conditions
Normal operating mode	Peripheral module clocks are stopped	$T_a = 25^\circ\text{C}$	I_{CC}	1.4	—	mA	ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times 1/64$
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
		$T_a = 25^\circ\text{C}$		0.9	—		ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times 1/64$
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
		$T_a = 25^\circ\text{C}$		0.2	—		ICLK = 1 MHz Clock Source = LOCO FCLK, PCLKB, PCLKD = Clock Source $\times 1/64$
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
	Peripheral module clocks are supplied	$T_a = 25^\circ\text{C}$		7.1	14.7		ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 24 MHz
		$T_a = 85^\circ\text{C}$			14.5		
		$T_a = 105^\circ\text{C}$			14.4		
		$T_a = 125^\circ\text{C}$			14.2		
		$T_a = 25^\circ\text{C}$		3.5	8.1		ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 8 MHz
		$T_a = 85^\circ\text{C}$			7.9		
		$T_a = 105^\circ\text{C}$			7.9		
		$T_a = 125^\circ\text{C}$			7.8		
		$T_a = 25^\circ\text{C}$		1.7	4.9		ICLK = 1 MHz Clock Source = LOCO FCLK, PCLKB, PCLKD = 1 MHz
		$T_a = 85^\circ\text{C}$			4.7		
		$T_a = 105^\circ\text{C}$			4.6		
		$T_a = 125^\circ\text{C}$			4.6		
Sleep mode	Peripheral module clocks are stopped	$T_a = 25^\circ\text{C}$		0.8	—		ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times 1/64$
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
		$T_a = 25^\circ\text{C}$		0.7	—		ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source $\times 1/64$
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
		$T_a = 25^\circ\text{C}$		0.2	—		ICLK = 1 MHz Clock Source = LOCO FCLK, PCLKB, PCLKD = Clock Source $\times 1/64$
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
	Peripheral module clocks are supplied	$T_a = 25^\circ\text{C}$		4.4	—		ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 24 MHz
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		
		$T_a = 25^\circ\text{C}$		2.6	—		ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 8 MHz
		$T_a = 85^\circ\text{C}$			—		
		$T_a = 105^\circ\text{C}$			—		
		$T_a = 125^\circ\text{C}$			—		

Table 38.15 Operating Current in Middle-Speed Operating Mode*1 (2/2)

Item			Symbol	Typ.	Max.	Unit	Test Conditions
Sleep mode	Peripheral module clocks are supplied	T _a = 25°C	I _{CC}	1.4	—	mA	ICLK = 1 MHz Clock Source = LOCO FCLK, PCLKB, PCLKD = 1 MHz
		T _a = 85°C			—		
		T _a = 105°C			—		
		T _a = 125°C			—		
Deep sleep mode	Peripheral module clocks are stopped	T _a = 25°C	0.6	—		ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source × 1/64	
		T _a = 85°C		—			
		T _a = 105°C		—			
		T _a = 125°C		—			
		T _a = 25°C	0.6	—	ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = Clock Source × 1/64		
		T _a = 85°C		—			
		T _a = 105°C		—			
		T _a = 125°C		—			
		T _a = 25°C	0.1	—	ICLK = 1 MHz Clock Source = LOCO FCLK, PCLKB, PCLKD = Clock Source × 1/64		
		T _a = 85°C		—			
		T _a = 105°C		—			
		T _a = 125°C		—			
		Peripheral module clocks are supplied	T _a = 25°C	4.0	—	ICLK = 24 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 24 MHz	
			T _a = 85°C		—		
			T _a = 105°C		—		
			T _a = 125°C		—		
	T _a = 25°C		2.4	—	ICLK = 8 MHz Clock Source = HOCO FCLK, PCLKB, PCLKD = 8 MHz		
	T _a = 85°C			—			
	T _a = 105°C			—			
	T _a = 125°C			—			
	T _a = 25°C	1.3	—	ICLK = 1 MHz Clock Source = LOCO FCLK, PCLKB, PCLKD = 1 MHz			
	T _a = 85°C		—				
	T _a = 105°C		—				
	T _a = 125°C		—				
Increase during background operation*2				2.1	—		

Note 1. Operating current values do not include output charge/discharge current from all pins. The values apply when internal pull-up resistors are in the off state.

Note 2. This is the increase for programming or erasure of the ROM or E2 DataFlash during program execution. The operating current during background operation is not included.

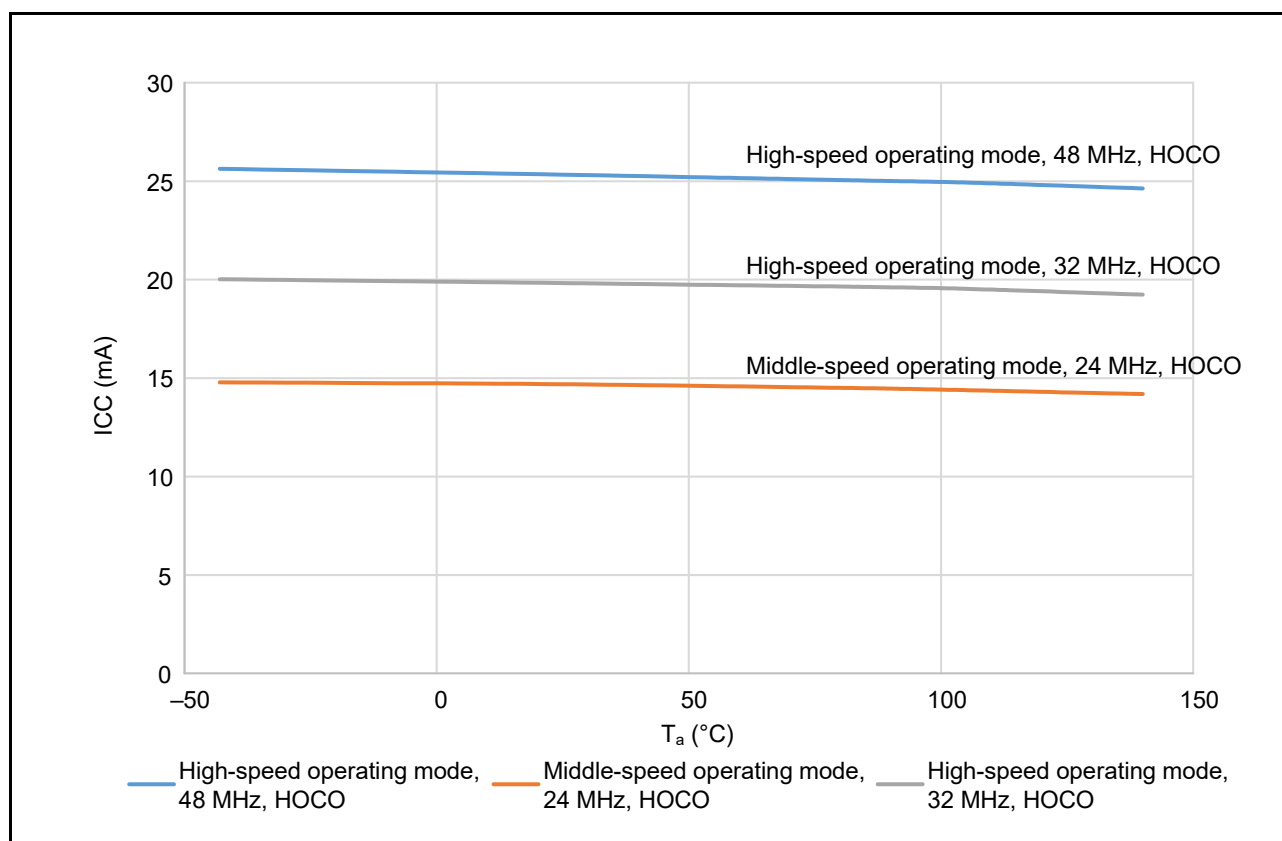


Figure 38.2 Temperature Dependence in the Normal Operating Mode

Table 38.16 Standby Current in Software Standby Mode*1, *2

Item		Symbol	Typ.	Max.	Unit	Test Conditions	
Standby current (3.3 V)	T _a = 25°C	I _{STBY}	0.24	0.54	μA		
	T _a = 55°C		0.41	2.02			
	T _a = 85°C		1.10	8.51			
	T _a = 105°C		2.58	20.98			
	T _a = 125°C		6.24	48.36			
Standby current (5.0 V)	T _a = 25°C		0.27	0.57			
	T _a = 55°C		0.44	2.15			
	T _a = 85°C		1.15	9.02			
	T _a = 105°C		2.68	22.21			
	T _a = 125°C		6.49	51.27			

Note 1. Standby current values are with all output pins unloaded and all input pull-up resistors in the off state.

Note 2. The IWD, LVD, and CMPC are stopped.

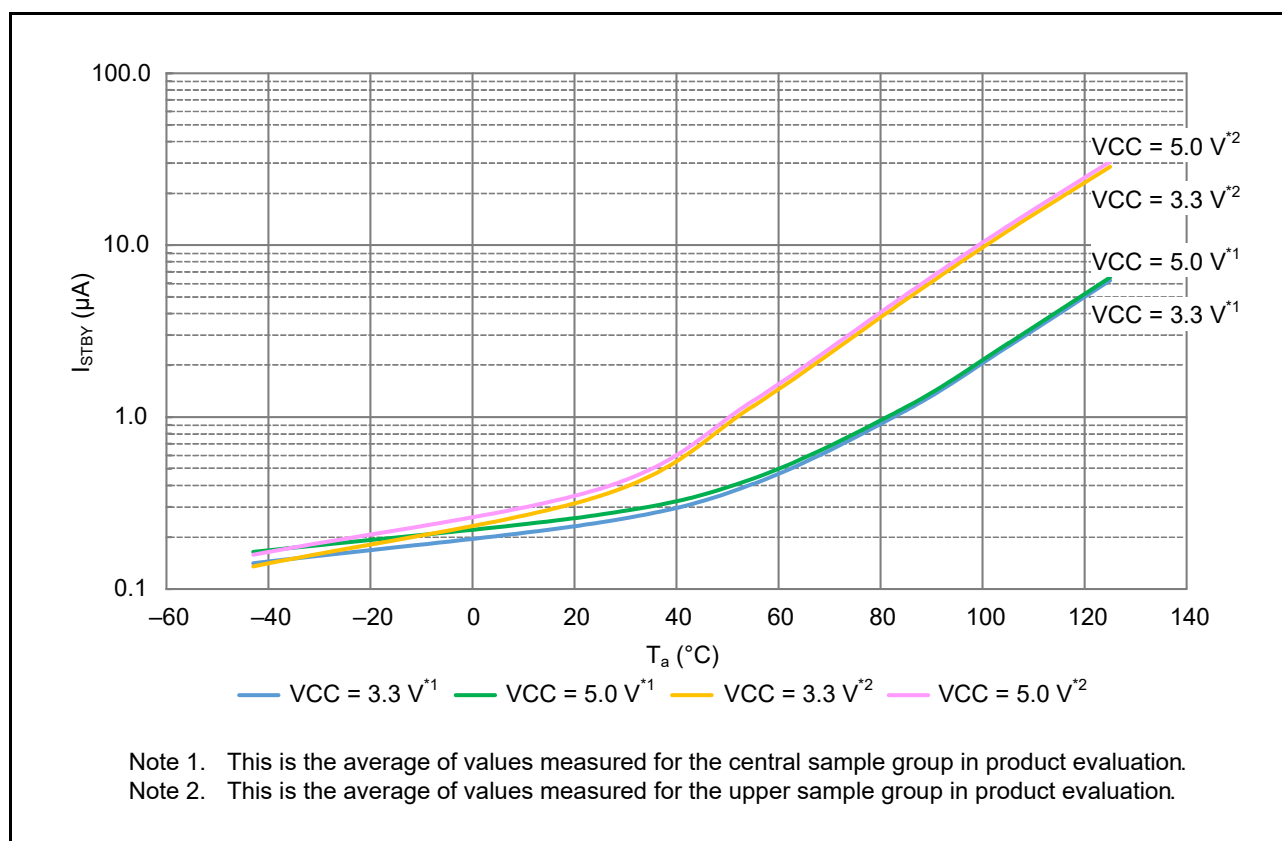
**Figure 38.3 Temperature Dependency in Software Standby Mode**

Table 38.17 Operating Current by Functions

Conditions: ICLK = PCLKB = PCLKD = 48 MHz unless otherwise specified.

Item	Min.	Typ.	Max.	Unit
Voltage detection circuit (LVD0)	—	0.04	—	μA
Voltage detection circuit (LVD1, LVD2) (per channel)	—	0.12	—	
Independent watchdog timer	—	0.32	—	
Clock frequency accuracy measurement circuit	—	0.11	—	mA
Data transfer controller	—	2.50	—	
Multi-function timer pulse unit 3	—	1.32	—	
Port output enable 3	—	0.14	—	
General PWM timer	—	0.69	—	
GPTW port output enable	—	0.05	—	
8-bit timer (per unit)	—	0.06	—	
Compare match timer	—	0.05	—	
Serial communications interface (per unit)	—	0.41	—	
I ² C-bus interface	—	0.20	—	
CRC calculator	—	0.08	—	
Arithmetic unit for trigonometric functions	—	0.33	—	
12-bit A/D converter (per unit)*1	—	1.39	1.73	
12-bit A/D converter reference (per unit)*1	—	0.11	0.13	
Programmable gain amplifier	—	0.42	0.80	
D/A converter (per unit)	—	0.35	0.53	
Temperature sensor	—	0.12	—	
Comparator C (per unit)	—	0.09	0.17	
Data operation circuit	—	0.09	—	

Note 1. This applies when PCLKD = 64 MHz.

38.4.3 VCC Rising Gradient and Ripple Characteristics

Table 38.18 VCC Rising Gradient

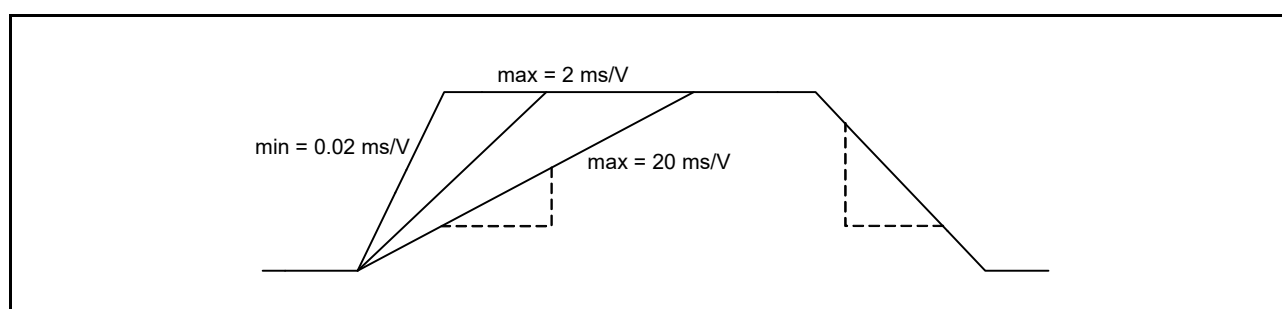
Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Power-on VCC rising gradient	At normal startup*1	SrVCC	0.02	—	20	ms/V	
	During fast startup time*2		0.02	—	2		
	Voltage monitoring 0 reset enabled at startup*3, *4		0.02	—	—		

Note 1. When OFS1.(FASTSTUP, LVDAS) = 11b.

Note 2. When OFS1.(FASTSTUP, LVDAS) = 01b.

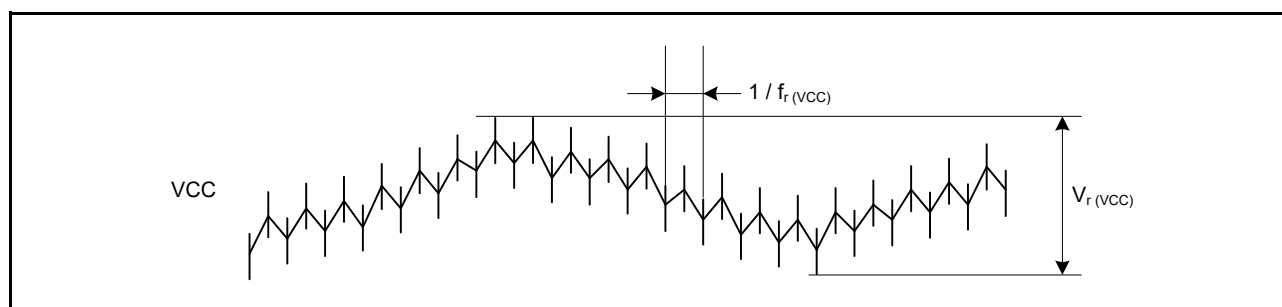
Note 3. When OFS1.LVDAS = 0.

Note 4. Turn on the power supply voltage according to the normal startup rising gradient because the register settings set by OFS1 are not read in boot mode.


Figure 38.4 VCC Ramp Rate at Power-On
Table 38.19 Ripple Characteristics

The ripple voltage must meet the allowable ripple frequency $f_r(VCC)$ within the range between the VCC upper limit and lower limit. When VCC change exceeds $VCC \pm 10\%$, the allowable voltage change rising/falling gradient $dt/dVCC$ must be met.

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Allowable ripple frequency	$f_r(VCC)$	—	—	10	kHz	Figure 38.5 $V_r(VCC) \leq 0.2 \times VCC$
		—	—	1	MHz	Figure 38.5 $V_r(VCC) \leq 0.08 \times VCC$
		—	—	10	MHz	Figure 38.5 $V_r(VCC) \leq 0.06 \times VCC$
Allowable voltage change rising/falling gradient	$dt/dVCC$	1.0	—	—	ms/V	When VCC change exceeds $VCC \pm 10\%$


Figure 38.5 Ripple Waveform

38.4.4 Voltage of the VCL Pin

Table 38.20 Voltage of the VCL Pin

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Output voltage of the VCL pin	V _{CL}	—	1.5	—	V	

38.4.5 RAM Data Standby Characteristics

Table 38.21 RAM Data Standby Characteristics

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
RAM standby voltage	V _{RAM}	2.7	—	—	V	

38.4.6 Thermal Resistance Value (Reference Values)

Table 38.22 Thermal Resistance Value (Reference Values)

Item	Package	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Thermal resistance	64-pin LFQFP (PLQP0064KB-C)	θ_{ja}	—	—	53.7	°C/W	JESD51-2 and JESD51-7 compliant
	64-pin LQFP (PLQP0064GA-A)		—	—	53.3		
	52-pin LQFP (PLQP0052JA-A)		—	—	45.7		
	48-pin LFQFP (PLQP0048KB-B)		—	—	64		
	44-pin LQFP (PLQP0044GF-A)		—	—	53		
	32-pin LQFP (PLQP0032GB-A)		—	—	63.2		
	48-pin HWQFN (PWQN0048KC-A)		—	—	28.7* ¹		
	32-pin HWQFN (PWQN0032KE-A)		—	—	35.8* ¹		
	24-pin HWQFN (PWQN0024KG-A)		—	—	43.6* ¹		
	64-pin LFQFP (PLQP0064KB-C)	Ψ_{jt}	—	—	2.02		
	64-pin LQFP (PLQP0064GA-A)		—	—	2.02		
	52-pin LQFP (PLQP0052JA-A)		—	—	1.89		
	48-pin LFQFP (PLQP0048KB-B)		—	—	5.01		
	44-pin LQFP (PLQP0044GF-A)		—	—	2.02		
	32-pin LQFP (PLQP0032GB-A)		—	—	5.13		
	48-pin HWQFN (PWQN0048KC-A)		—	—	0.21* ¹		
	32-pin HWQFN (PWQN0032KE-A)		—	—	0.3* ¹		
	24-pin HWQFN (PWQN0024KG-A)		—	—	0.43* ¹		

Note: The values are reference values when the 4-layer board is used. Thermal resistance depends on the number of layers or size of the board. For details, refer to the JEDEC standards.

Note 1. This value applies when the exposed die pad for this purpose is connected to VSS.

38.5 AC Characteristics

38.5.1 Clock Timing

Table 38.23 Operating Frequency Value (High-Speed Operating Mode)

Item		Symbol	Min.	Typ.	Max.	Unit
Maximum operating frequency	System clock (ICLK)	f	—	—	48	MHz
	FlashIF clock (FCLK)*1, *2		—	—	48	
	Peripheral module clock (PCLKB)		—	—	48	
	Peripheral module clock (PCLKD)		1	—	64	

Note 1. The lower-limit frequency of FCLK is 1 MHz during programming or erasing of the flash memory. When using FCLK at below 4 MHz, the frequency can be set to 1 MHz, 2 MHz, or 3 MHz. A non-integer frequency such as 1.5 MHz cannot be set.

Note 2. The frequency accuracy of FCLK should be $\pm 3.5\%$.

Table 38.24 Operating Frequency Value (Middle-Speed Operating Mode)

Item		Symbol	Min.	Typ.	Max.	Unit
Maximum operating frequency	System clock (ICLK)	f	—	—	24	MHz
	FlashIF clock (FCLK)*1, *2		—	—	24	
	Peripheral module clock (PCLKB)		—	—	24	
	Peripheral module clock (PCLKD)		1	—	24	

Note 1. The lower-limit frequency of FCLK is 1 MHz during programming or erasing of the flash memory. When using FCLK at below 4 MHz, the frequency can be set to 1 MHz, 2 MHz, or 3 MHz. A non-integer frequency such as 1.5 MHz cannot be set.

Note 2. The frequency accuracy of FCLK should be $\pm 3.5\%$.

Table 38.25 EXTAL Clock Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
EXTAL external clock input cycle time	t_{Xcyc}	50	—	—	ns	Figure 38.6
EXTAL external clock input frequency	f_{XMAIN}	—	—	20	MHz	
EXTAL external clock input high pulse width	t_{XH}	20	—	—	ns	
EXTAL external clock input low pulse width	t_{XL}	20	—	—	ns	
EXTAL external clock rise time	t_{Xr}	—	—	5	ns	
EXTAL external clock fall time	t_{Xf}	—	—	5	ns	
EXTAL external clock input wait time*1	t_{XWT}	0.5	—	—	μ s	

Note 1. Time until the clock can be used after the main clock oscillator stop bit (MOSCCR.MOSTP) is set to 0 (operating).

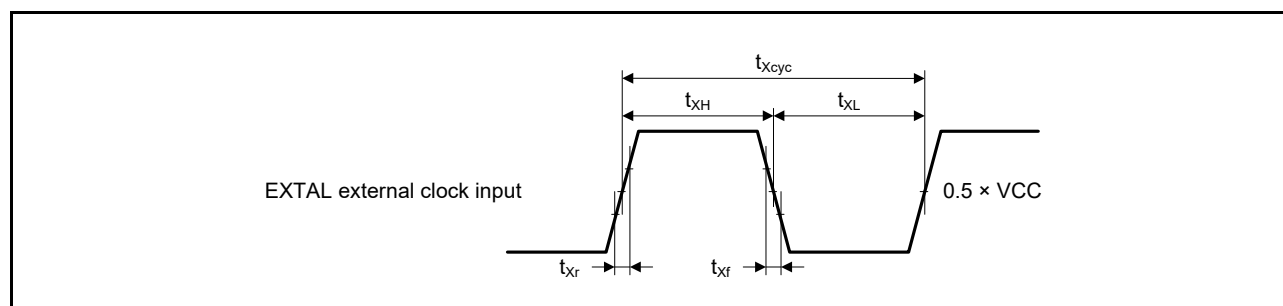


Figure 38.6 EXTAL External Clock Input Timing

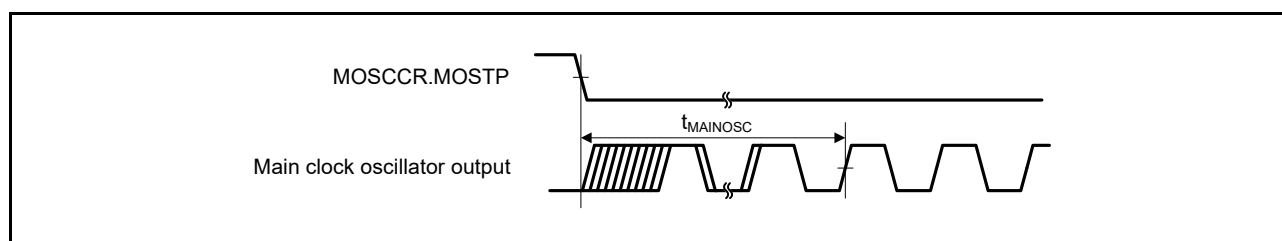
Table 38.26 Main Clock Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Main clock oscillator oscillation frequency	f_{MAIN}	1	—	20	MHz	
Main clock oscillation stabilization time (crystal)*1	t_{MAINOSC}	—	3	—	ms	Figure 38.7
Main clock oscillation stabilization time (ceramic resonator)*1	t_{MAINOSC}	—	50	—	μs	

Note 1. Reference values when an 8-MHz resonator is used.

When specifying the main clock oscillator stabilization time, set the MOSCWTCR register with a stabilization time value that is equal to or greater than the resonator-manufacturer-recommended value.

After changing the setting of the MOSCCR.MOSTP bit so that the main clock oscillator operates, read the OSCOVFSR.MOOVF flag to confirm that it has become 1, and then start using the main clock.

**Figure 38.7 Main Clock Oscillation Start Timing****Table 38.27 LOCO and IWDT-Dedicated Low-Speed Clock Timing**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
LOCO clock oscillation frequency	f_{LOCO}	3.44 (-14%)	4.0	4.56 (+14%)	MHz	
LOCO clock oscillation frequency error	Δf_{LOCO}	—	—	± 14	%	
LOCO clock oscillation stabilization time	t_{LOCO}	—	—	0.5	μs	Figure 38.8
IWDT-dedicated clock oscillation frequency	f_{ILOCO}	12.75 (-15%)	15	17.25 (+15%)	kHz	
IWDT-dedicated clock oscillation frequency error	Δf_{ILOCO}	—	—	± 15	%	
IWDT-dedicated clock oscillation stabilization time	t_{ILOCO}	—	—	80	μs	Figure 38.9

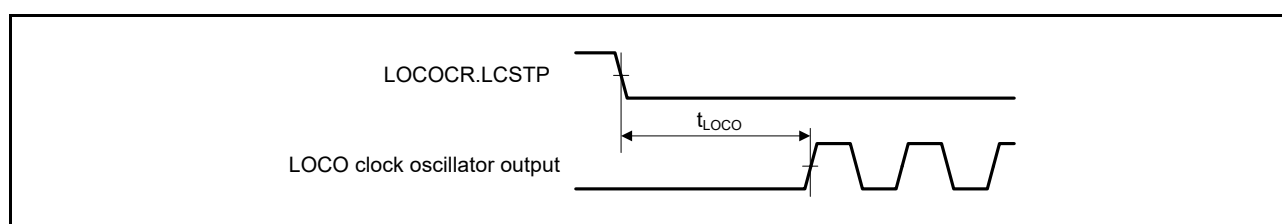
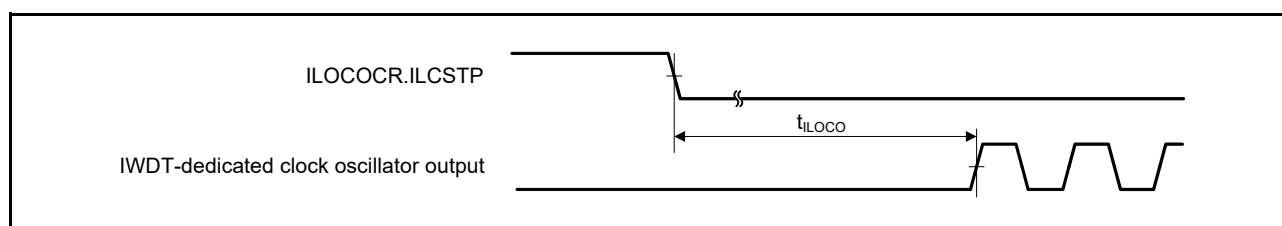
**Figure 38.8 LOCO Clock Oscillation Start Timing****Figure 38.9 IWDT-Dedicated Clock Oscillation Start Timing**

Table 38.28 HOCO Clock Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
HOCO clock oscillation frequency	f_{HOCO}	23.76 (−1.0%)	24	24.24 (+1.0%)	MHz	$T_a = -40 \text{ to } +125^\circ\text{C}$
		31.68 (−1.0%)	32	32.32 (+1.0%)		
		47.52 (−1.0%)	48	48.48 (+1.0%)		
		63.36 (−1.0%)	64	64.64 (+1.0%)		
HOCO oscillation frequency error	Δf_{HOCO}	—	—	± 1.0	%	$T_a = -40 \text{ to } +125^\circ\text{C}$
HOCO clock oscillation stabilization time	t_{HOCO}	—	—	4.95	μs	Figure 38.11

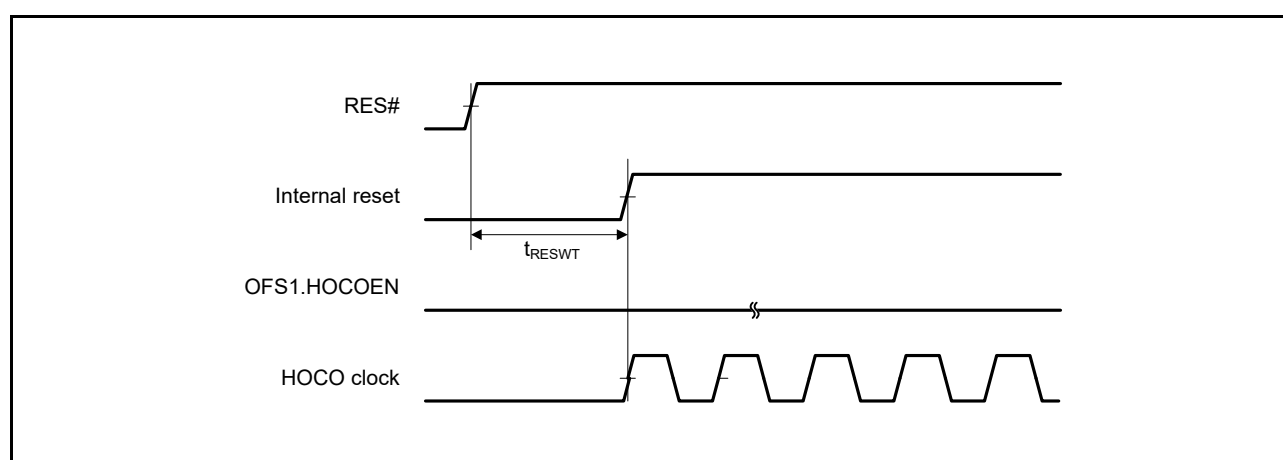
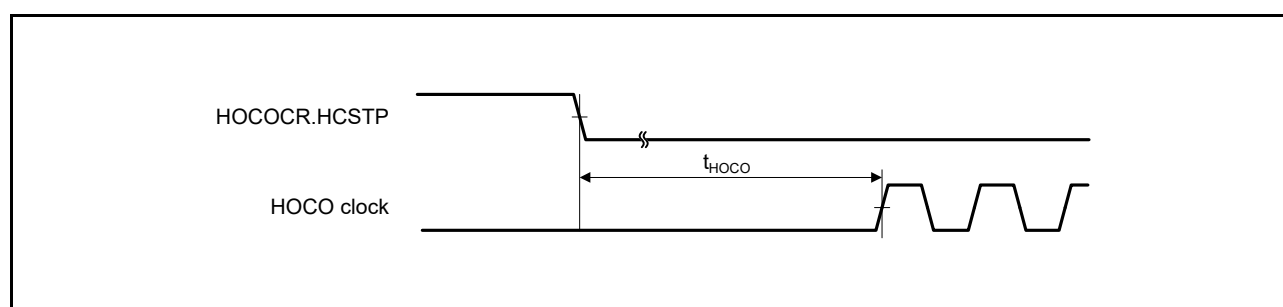
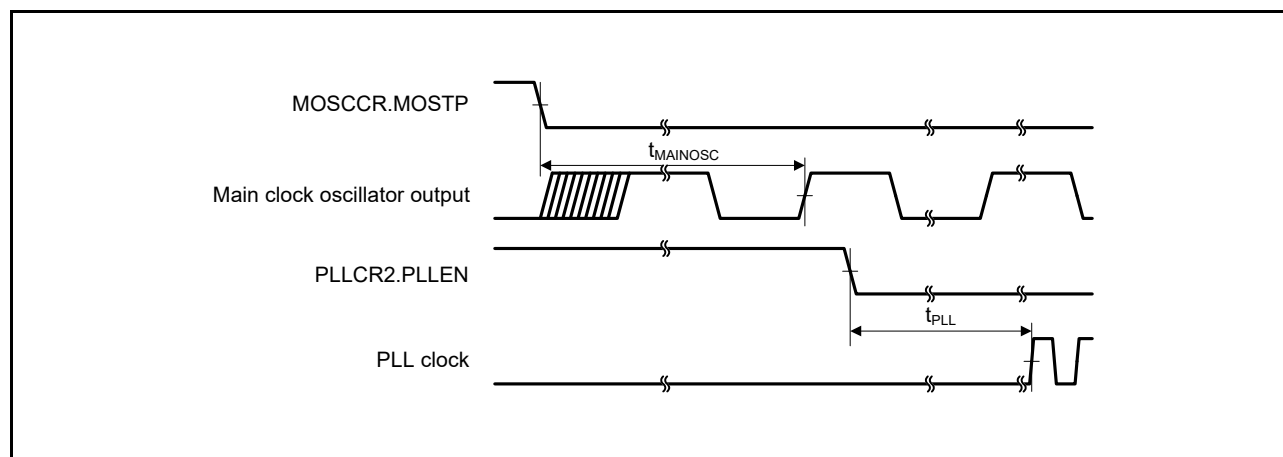
**Figure 38.10 HOCO Clock Oscillation Start Timing (After Reset is Canceled by Setting OFS1.HOCOEN Bit to 0)****Figure 38.11 HOCO Clock Oscillation Start Timing (Oscillation is Started by Setting HOCOCR.HCSTP Bit)**

Table 38.29 PLL Clock Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
PLL input frequency	f_{PLLIN}	4	—	12.5	MHz	Figure 38.12
PLL circuit oscillation frequency	f_{PLL}	24	—	64	MHz	
PLL clock oscillation stabilization time	t_{PLL}	—	—	81.4	μs	
PLL free-running oscillation frequency	f_{PLLFR}	—	9	—	MHz	

**Figure 38.12 PLL Clock Oscillation Start Timing (PLL is Operated after Main Clock Oscillation Has Settled)**

38.5.2 Reset Timing

Table 38.30 Reset Timing

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
RES# pulse width	At power-on	t_{RESWP}	10.5	—	—	ms	Figure 38.13
	Other than above	t_{RESW}	30	—	—	μ s	Figure 38.14
Wait time after RES# cancellation (cold start)	At normal startup*1	t_{RESWT}	—	27.5	—	ms	Figure 38.13
	During fast startup time*2	t_{RESWT}	—	850	—	μ s	
	Voltage monitoring 0 reset enabled at startup*4, *5	t_{RESWT}	—	850	—	μ s	
Wait time after RES# cancellation (warm start)	LVD0 disabled*3	t_{RESWT}	—	—	—	—	Figure 38.14
	LVD0 enabled*4		—	850	—	μ s	
Internal reset time (independent watchdog timer reset, software reset)	LVD0 disabled*3	t_{RESWT2}	—	190	—	μ s	
	LVD0 enabled*4		—	910	—	μ s	

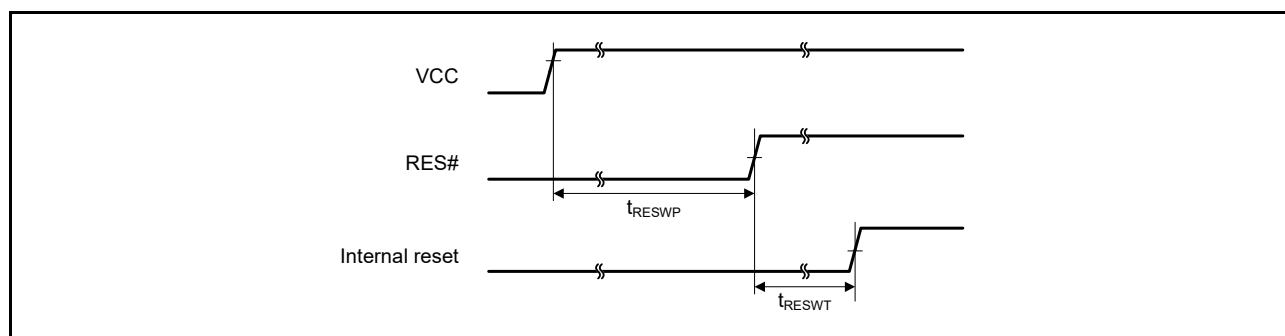
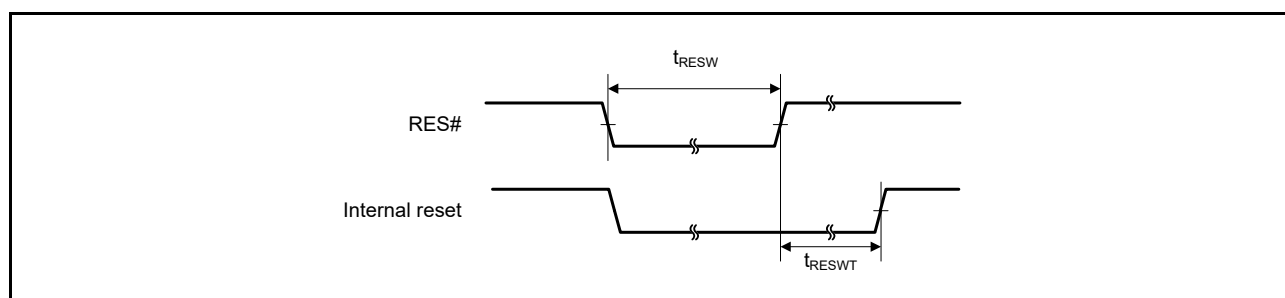
Note 1. When OFS1.(FASTSTUP, LVDAS) = 11b.

Note 2. When OFS1.(FASTSTUP, LVDAS) = 01b.

Note 3. When OFS1.LVDAS = 1.

Note 4. When OFS1.LVDAS = 0.

Note 5. Turn on the power supply voltage according to the normal startup rising gradient because the register settings set by OFS1 are not read in boot mode.

**Figure 38.13 Reset Input Timing at Power-On****Figure 38.14 Reset Input Timing**

38.5.3 Timing of Recovery from Low Power Consumption Modes

Table 38.31 Timing of Recovery from Low Power Consumption Modes (1)

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Oscillation stabilization wait time*1	High-speed operating mode/ Middle-speed operating mode	Main clock oscillator	$t_{\text{SBYOSCWTMC}}$	—	—	$t_{\text{LOCO}} + (16 + \text{Number of cycles specified in MOSCWTCR}) / f_{\text{LOCO}} + 3 / f_{\text{MOSC}} + 1 / f_{\text{ICLK}}$	μs
		PLL circuit (Main clock oscillator)	$t_{\text{SBYOSCWTTPC}}$	—	—	$t_{\text{LOCO}} + (280 + \text{Number of cycles specified in MOSCWTCR}) / f_{\text{LOCO}} + 3 / f_{\text{PLL}} + 1 / f_{\text{ICLK}}$	
		HOCO	$t_{\text{SBYOSCWTTHO}}$	—	—	$t_{\text{LOCO}} + 16 / f_{\text{LOCO}} + 3 / f_{\text{HOCO}} + 1 / f_{\text{ICLK}}$	
		PLL circuit (HOCO)	$t_{\text{SBYOSCWTTPH}}$	—	—	$t_{\text{LOCO}} + 296 / f_{\text{LOCO}} + 3 / f_{\text{PLL}} + 1 / f_{\text{ICLK}}$	
		LOCO	$t_{\text{SBYOSCWTLO}}$	—	—	$t_{\text{LOCO}} + 1 / f_{\text{ICLK}}$	
	Time required for operations by the sequencer before release from software standby mode*2		t_{SBYSEQ}	—	—	$4 / f_{\text{LOCO}} + 11 / f_{\text{ICLK}} + 3 / f_{\text{PCLKB}} + 3n / f_{\text{source clock}}$	
Recovery time from software standby mode*3	High-speed operating mode/ Middle-speed operating mode	Main clock oscillator	t_{SBYMC}	—	—	$t_{\text{SBYOSCWTMC}} + t_{\text{SBYSEQ}}$	Figure 38.15
		PLL circuit (Main clock oscillator)	t_{SBYPC}	—	—	$t_{\text{SBYOSCWTTPC}} + t_{\text{SBYSEQ}}$	
		HOCO	t_{SBYHO}	—	—	$t_{\text{SBYOSCWTTHO}} + t_{\text{SBYSEQ}}$	
		PLL circuit (HOCO)	t_{SBYPH}	—	—	$t_{\text{SBYOSCWTTPH}} + t_{\text{SBYSEQ}}$	
		LOCO	t_{SBYLO}	—	—	$t_{\text{SBYOSCWTLO}} + t_{\text{SBYSEQ}}$	

Note 1. When multiple oscillators are operating before entry to software standby mode, the oscillation stabilization wait time will be selected as the largest value among those for the operating oscillators.

Note 2. For n, the largest value is selected from among the internal clock division settings. The source clock is specified by the SCKCR3.CKSEL bits.

Note 3. The time for recovery from software standby mode is determined by the value obtained by adding the oscillation stabilization wait time and the time required for operations by the sequencer before release from software standby mode.

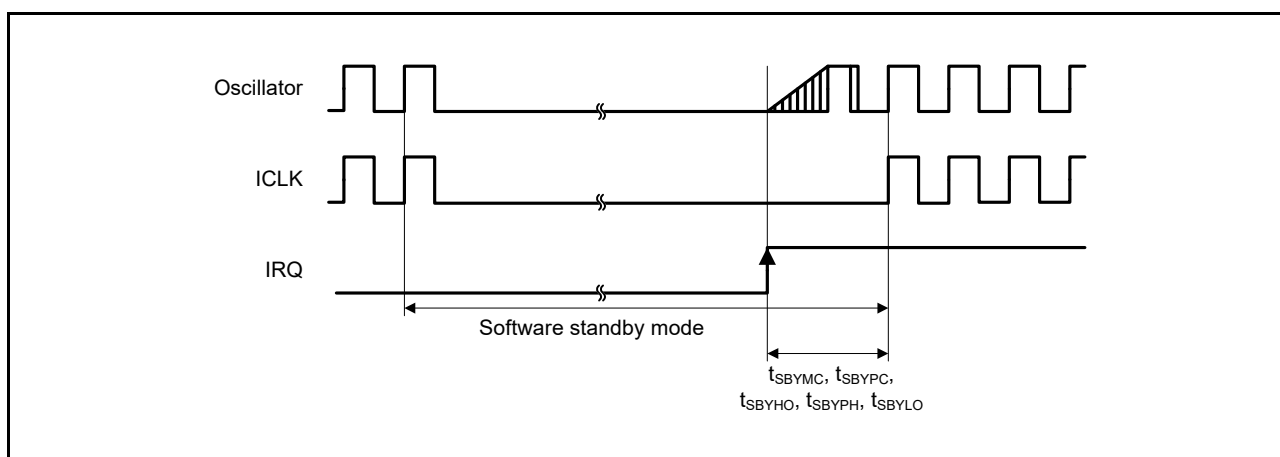


Figure 38.15 Software Standby Mode Recovery Timing

Table 38.32 Timing of Recovery from Low Power Consumption Modes (5)

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Recovery time from deep sleep mode*1, *2	High-speed operating mode	t_{DSLP}	—	—	$4 / f_{\text{LOCO}} + 8 / f_{\text{ICLK}} + 2 / f_{\text{PCLKB}} + 3n / f_{\text{source clock}}$	μs	Figure 38.16
	Middle-speed operating mode						

Note 1. Oscillators continue oscillating in deep sleep mode.

Note 2. For n, the largest value is selected from among the internal clock division settings. The source clock is specified by the SCKCR3.CKSEL bits.

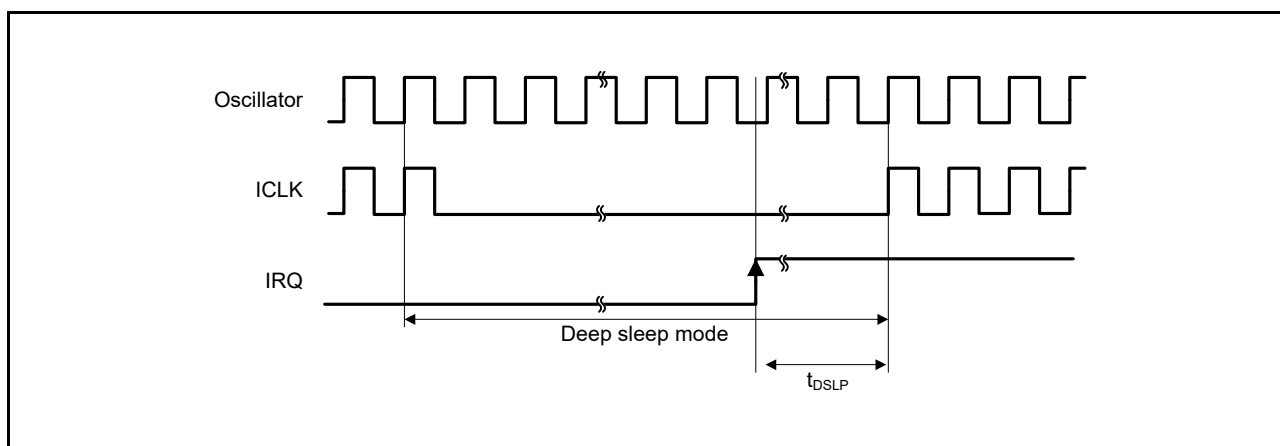


Figure 38.16 Deep Sleep Mode Recovery Timing

Table 38.33 Operating Mode Transition Time

Mode before Transition	Mode after Transition	ICLK Frequency	Transition Time			Unit
			Min.	Typ.	Max.	
High-speed operating mode	Middle-speed operating mode	24 MHz	—	$5 / f_{\text{ICLK}} + 3 / f_{\text{FCLK}}$	—	μs
Middle-speed operating mode	High-speed operating mode	24 MHz	—	$5 / f_{\text{ICLK}} + 3 / f_{\text{FCLK}}$	—	

38.5.4 Control Signal Timing

Table 38.34 Control Signal Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions	
NMI pulse width	t_{NMIW}	200	—	—	ns	NMI digital filter disabled (NMIFLTE.NFLTEN = 0)	$t_{\text{PBcyc}} \times 2 \leq 200 \text{ ns}$
		$t_{\text{PBcyc}} \times 2^{*1}$	—	—			$t_{\text{PBcyc}} \times 2 > 200 \text{ ns}$
		200	—	—		NMI digital filter enabled (NMIFLTE.NFLTEN = 1)	$t_{\text{NMICK}} \times 3 \leq 200 \text{ ns}$
		$t_{\text{NMICK}} \times 3.5^{*2}$	—	—			$t_{\text{NMICK}} \times 3 > 200 \text{ ns}$
IRQ pulse width	t_{IRQW}	200	—	—	ns	IRQ digital filter disabled (IRQFLTE0.FLTENi = 0)	$t_{\text{PBcyc}} \times 2 \leq 200 \text{ ns}$
		$t_{\text{PBcyc}} \times 2^{*1}$	—	—			$t_{\text{PBcyc}} \times 2 > 200 \text{ ns}$
		200	—	—		IRQ digital filter enabled (IRQFLTE0.FLTENi = 1)	$t_{\text{IRQCK}} \times 3 \leq 200 \text{ ns}$
		$t_{\text{IRQCK}} \times 3.5^{*3}$	—	—			$t_{\text{IRQCK}} \times 3 > 200 \text{ ns}$

Note: 200 ns minimum in software standby mode.

Note 1. t_{PBcyc} indicates the cycle of PCLKB.

Note 2. t_{NMICK} indicates the cycle of the NMI digital filter sampling clock.

Note 3. t_{IRQCK} indicates the cycle of the IRQi digital filter sampling clock (i = 0 to 7).

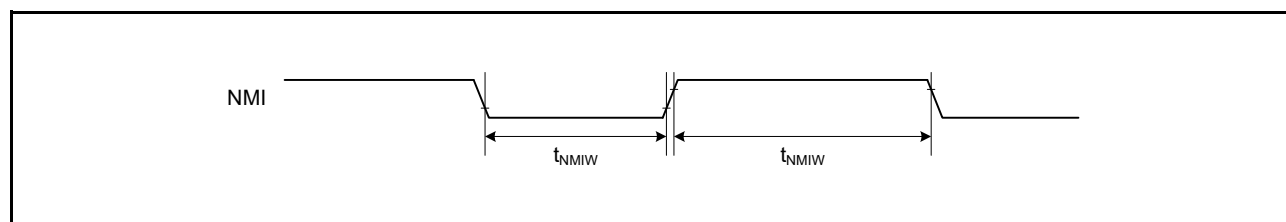


Figure 38.17 NMI Interrupt Input Timing

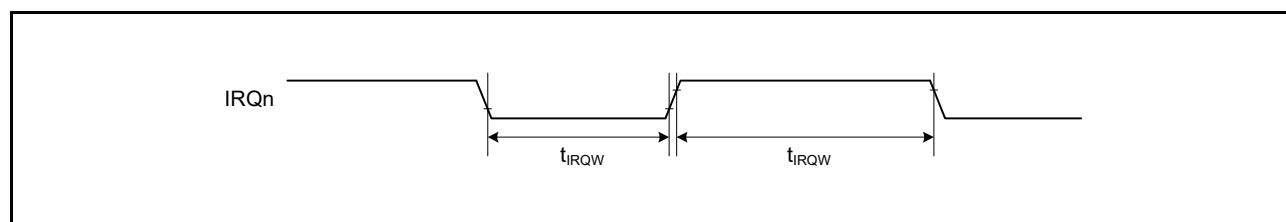


Figure 38.18 IRQ Interrupt Input Timing

38.5.5 Timing of On-Chip Peripheral Modules

38.5.5.1 I/O Port Input Timing

Table 38.35 I/O Port Input Timing

Item		Symbol	Min.	Max.	Unit *1	Test Conditions
I/O ports	Input data pulse width	t_{PRW}	1.5	—	t_{PBcyc}	Figure 38.19

Note 1. t_{PBcyc} : PCLKB cycle

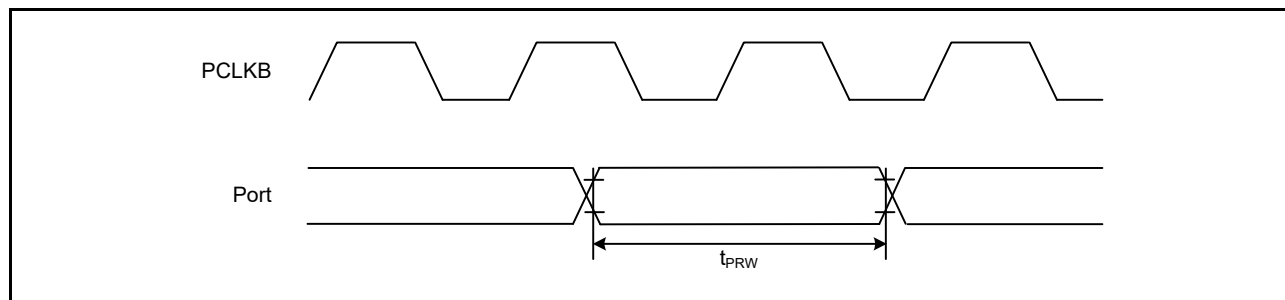


Figure 38.19 I/O Port Input Timing

38.5.5.2 MTU

Table 38.36 MTU Timing

Item			Symbol	Min.	Max.	Unit *1	Test Conditions
MTU	Input capture input pulse width	Single-edge setting	t_{MTICW}	1.5	—	t_{PBcyc}	Figure 38.20
		Both-edge setting		2.5	—		
	Input capture input rise/fall time		t_{TICr} , t_{TICf}	—	0.1	$\mu\text{s/V}$	Figure 38.21
	Timer clock pulse width	Single-edge setting	t_{MTCKWH} , t_{MTCKWL}	1.5	—	t_{PBcyc}	
		Both-edge setting		2.5	—		
		Phase counting mode		2.5	—		
	Timer clock rise/fall time		t_{TCKr} , t_{TCKf}	—	0.1	$\mu\text{s/V}$	

Note 1. t_{PBcyc} : PCLKB cycle

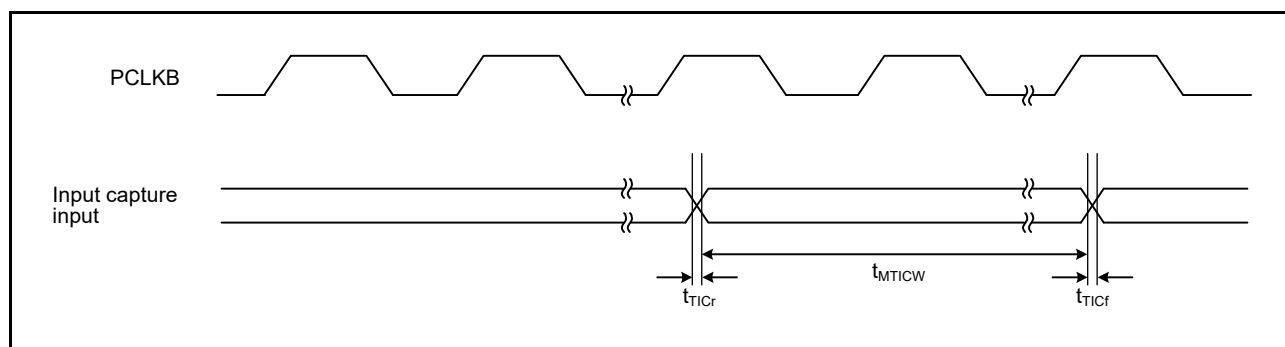
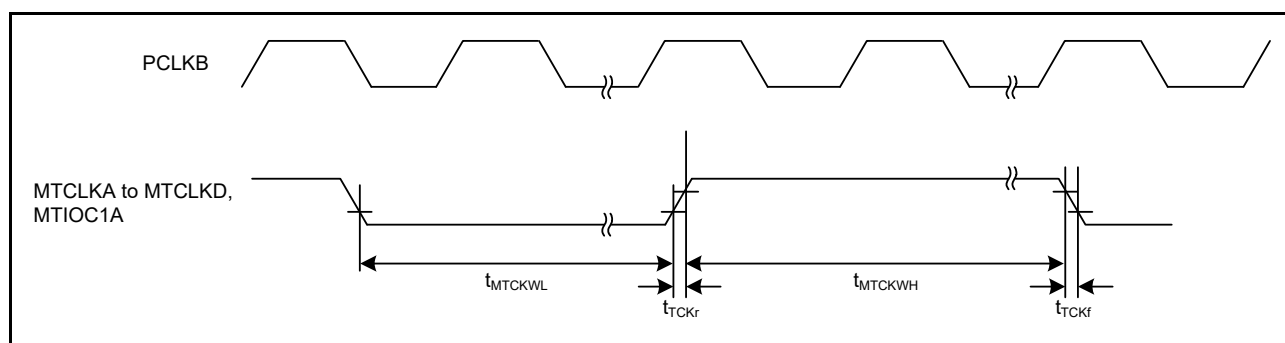


Figure 38.20 MTU Input/Output Timing

**Figure 38.21 MTU Clock Input Timing**

38.5.5.3 POE

Table 38.37 POE Timing

Item		Symbol	Min.	Max.	Unit *1	Test Conditions
POE	POE# input pulse width	t_{POEW}	1.5	—	t_{PBcyc}	Figure 38.22
	POE# input rise/fall time	t_{POEr} , t_{POEf}	—	0.1	$\mu s/V$	
	Output disable time	Transition of the POEn# signal level	t_{POEDI}	—	$5 \times t_{PBcyc} + 0.24$	Figure 38.23 When detecting falling edges (ICSRm.POEnM[3:0] = 0000 (m = 1, 3 to 5, 7, n = 0, 8, 10 to 12))
		Simultaneous conduction of output pins	t_{POEDO}	—	$3 \times t_{PBcyc} + 0.2$	Figure 38.24
		Detection of comparator outputs	t_{POEDC}	—	$5 \times t_{PBcyc} + 0.2$	Figure 38.25 The time is that when the noise filter for comparator C is not in use (CMPCTL.NFE = 0) and excludes the time for detection by comparator C.
		Register setting	t_{POEDS}	—	$t_{PBcyc} + 0.2$	Figure 38.26 Time for access to the register is not included.
		Oscillation stop detection	t_{POEDOS}	—	8	Figure 38.27

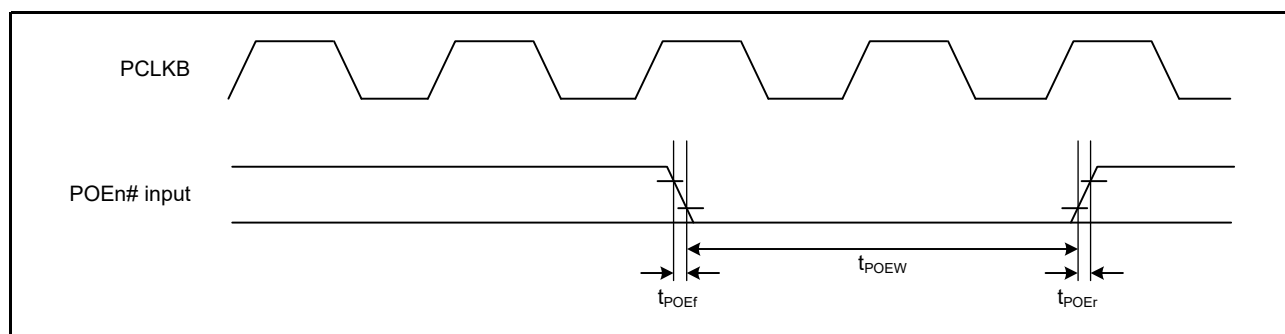
Note 1. t_{PBcyc} : PCLKB cycle

Figure 38.22 POE# Input Timing (n = 0, 8, 10 to 12)

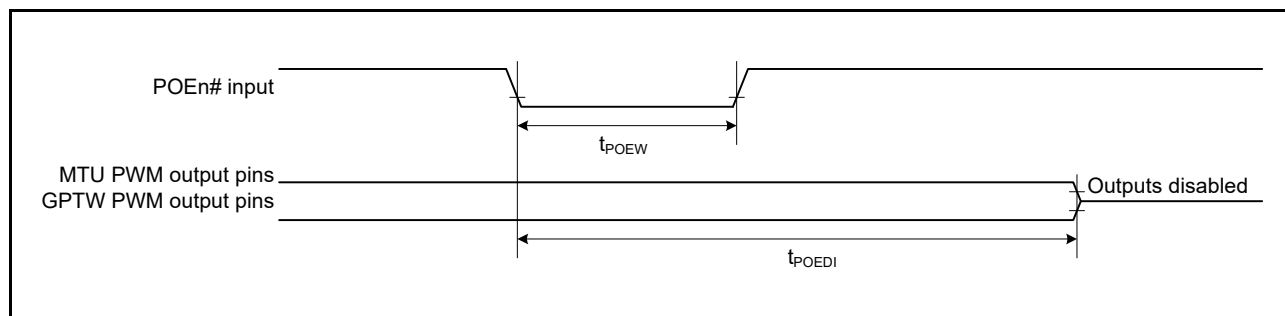


Figure 38.23 Output Disable Time for POE in Response to Transition of the POEn# Signal Level (n = 0, 8, 10 to 12)

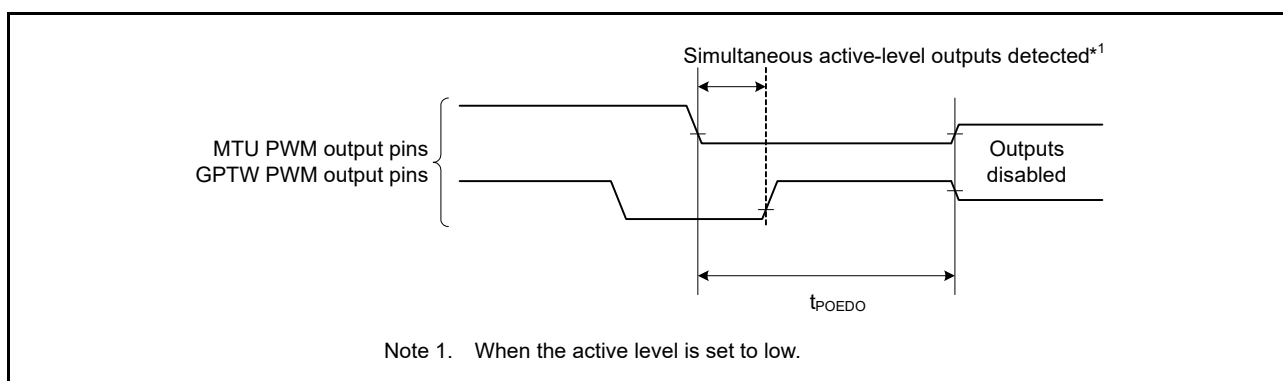


Figure 38.24 Output Disable Time for POE in Response to the Simultaneous Conduction of Output Pins

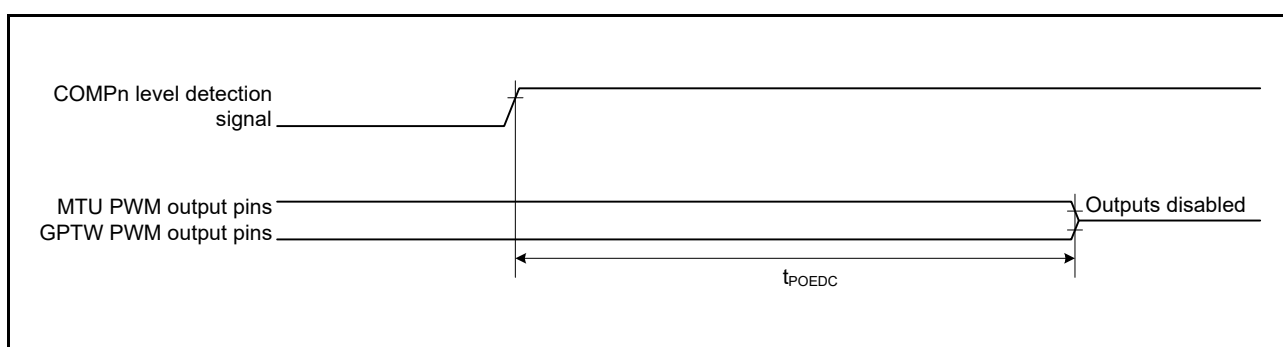


Figure 38.25 Output Disable Time for POE in Response to Detection of the Comparator Outputs (n = 0 to 2)

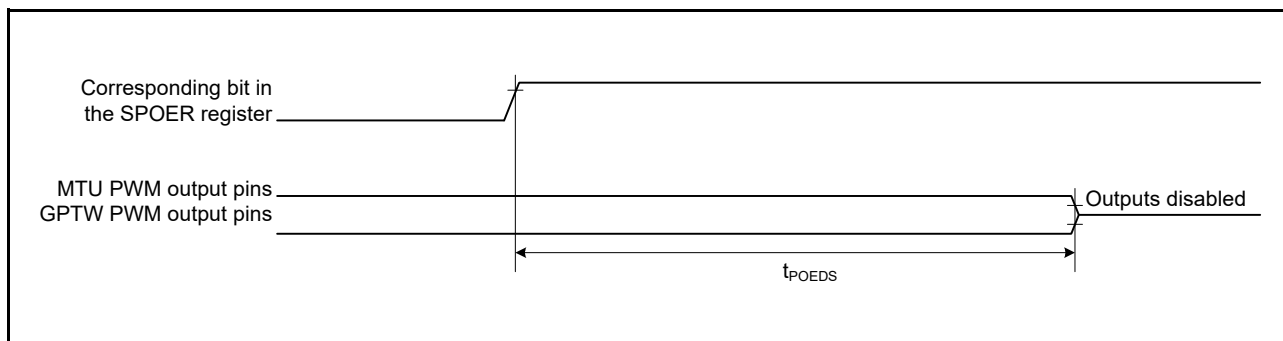


Figure 38.26 Output Disable Time for POE in Response to the Register Setting

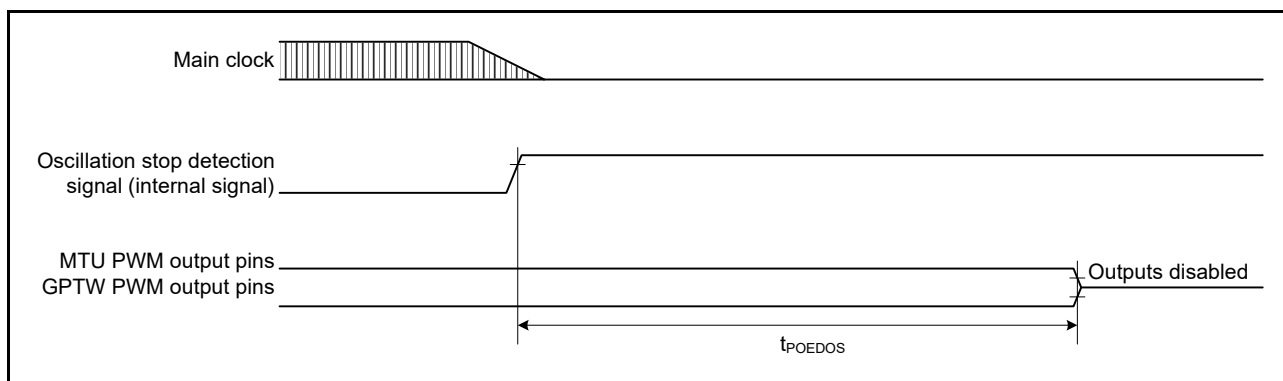
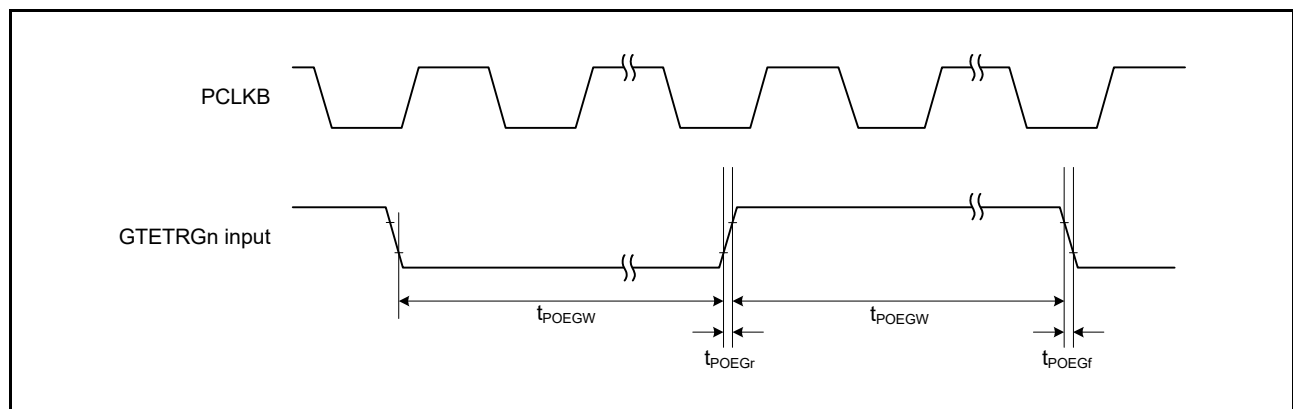


Figure 38.27 Output Disable Time for POE in Response to the Oscillation Stop Detection

38.5.5.4 POEG

Table 38.38 POEG Timing

Item		Symbol	Min.	Max.	Unit*1	Test Conditions
POEG	GTETRn input pulse width ($n = A$)	t_{POEGW}	1.5	—	t_{PBcyc}	Figure 38.28
	GTETRGA input rise/fall time	t_{POEGr} t_{POEGf}	—	0.1	μs	
	Output disable time	Input level detection of the GTETRn pin (via flag)	t_{POEGDI}	—	$3 \times t_{PBcyc} + 0.34$	μs Figure 38.29 When the digital noise filter is not in use (POEGn.NFEN = 0 ($n = A$))
		Detection of the output stopping signal from GPTW (deadtime error, simultaneous high output, or simultaneous low output)	t_{POEGDE}	—	0.5	μs Figure 38.30
		Edge detection signal from a comparator	t_{POEGDC}	—	$4 \times t_{PBcyc} + 0.5$	μs Figure 38.31 The time is that when the noise filter for comparator C is not in use (CMPCTL.NFE = 0) and excludes the time for detection by comparator C.
		Register setting	t_{POEGDS}	—	$t_{PBcyc} + 0.3$	μs Figure 38.32 Time for access to the register is not included.
		Oscillation stop detection	$t_{POEGDOS}$	—	8	μs Figure 38.33
		Input level detection of the GTETRn pin (direct path)	$t_{POEGDDI}$	—	$3 \times t_{PBcyc} + 0.34$	μs Figure 38.34
		Level detection signal from a comparator	$t_{POEGDDC}$	—	$t_{PBcyc} + 0.3$	μs Figure 38.35 The time is that when the noise filter for comparator C is not in use (CMPCTL.NFE = 0) and excludes the time for detection by comparator C.

Note 1. t_{PBcyc} : PCLKB cycleFigure 38.28 POEG Input Timing ($n = A$)

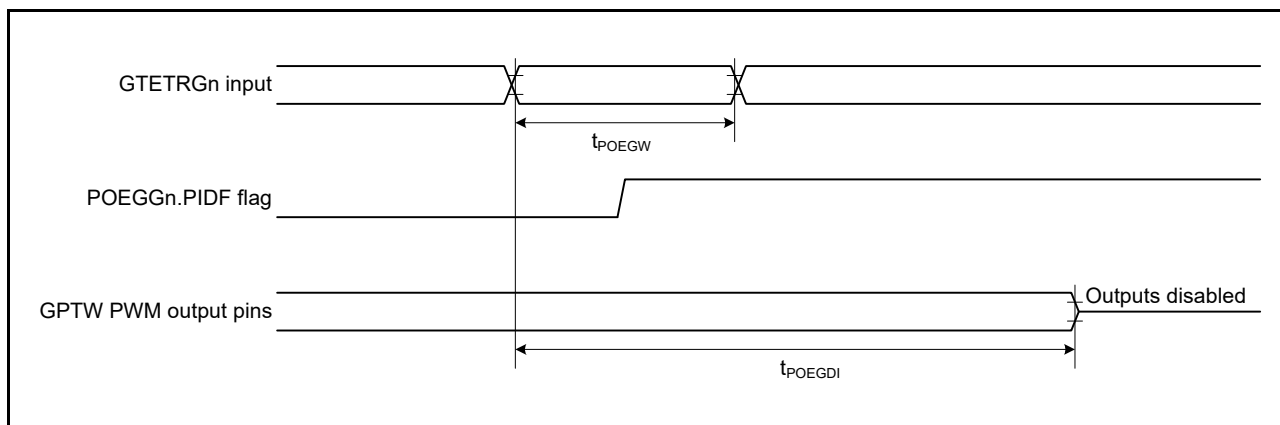


Figure 38.29 Output Disable Time for POEG via Detection Flag in Response to the Input Level Detection of the GTETRn pin (n = A)

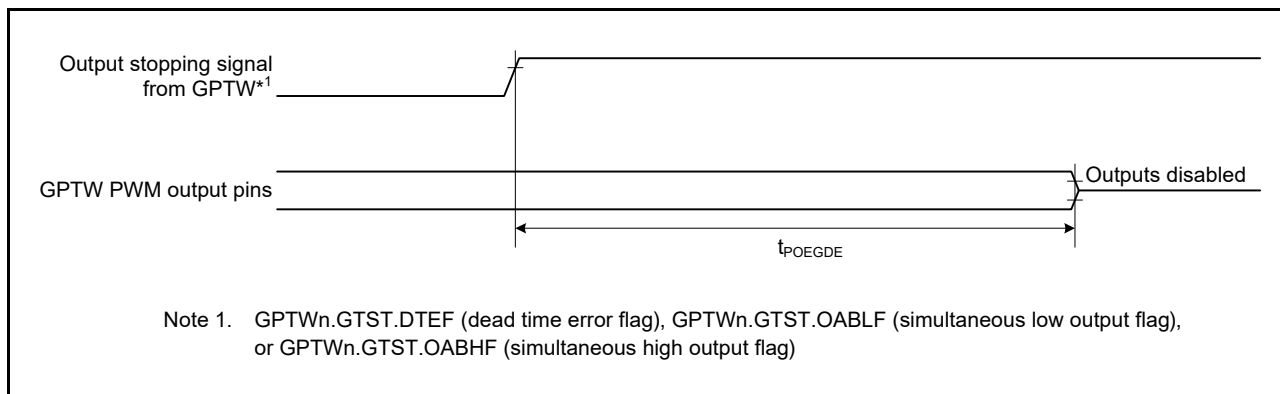


Figure 38.30 Output Disable Time for POEG in Response to Detection of the Output Stopping Signal from GPTW (n = 0 to 2)

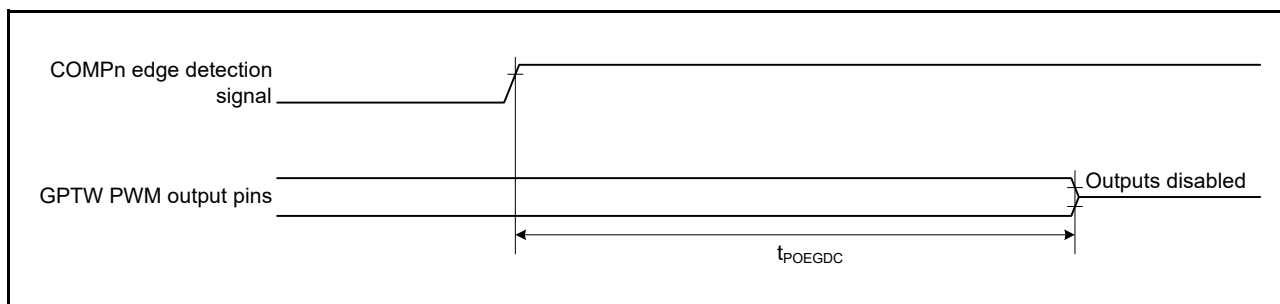


Figure 38.31 Output Disable Time for POEG in Response to Edge Detection Signal from a Comparator (n = 0 to 2)

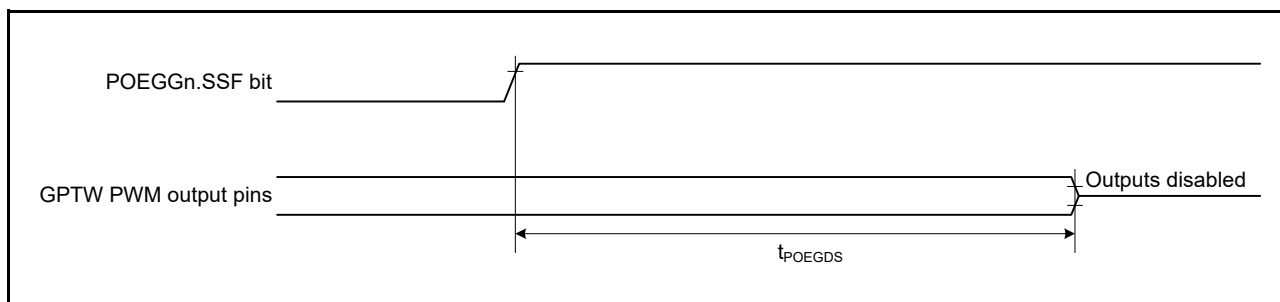


Figure 38.32 Output Disable Time for POEG in Response to the Register Setting (n = A)

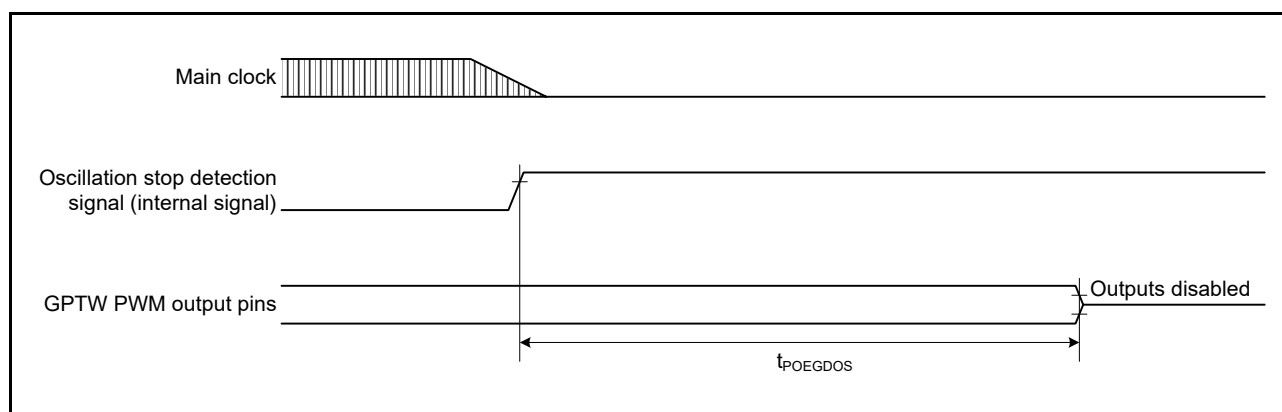


Figure 38.33 Output Disable Time of POEG in Response to the Oscillation Stop Detection

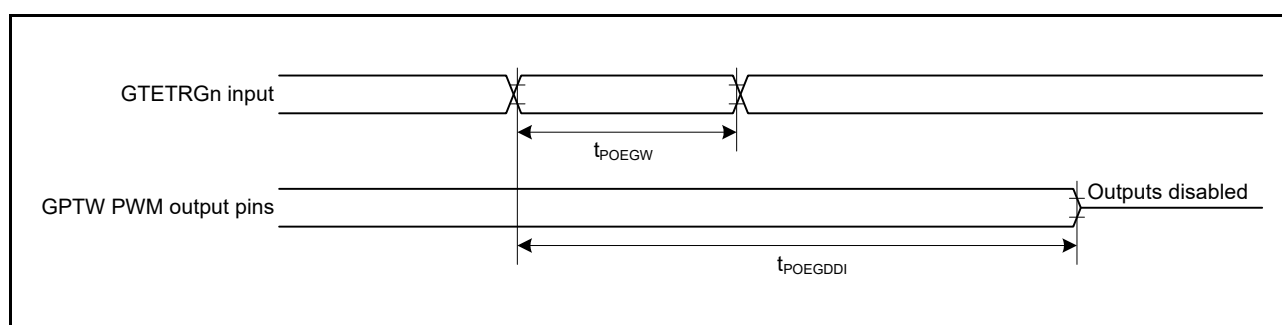


Figure 38.34 Output Disable Time for POEG in Direct Response to the Input Level Detection of the GTETRn pin (n = A)

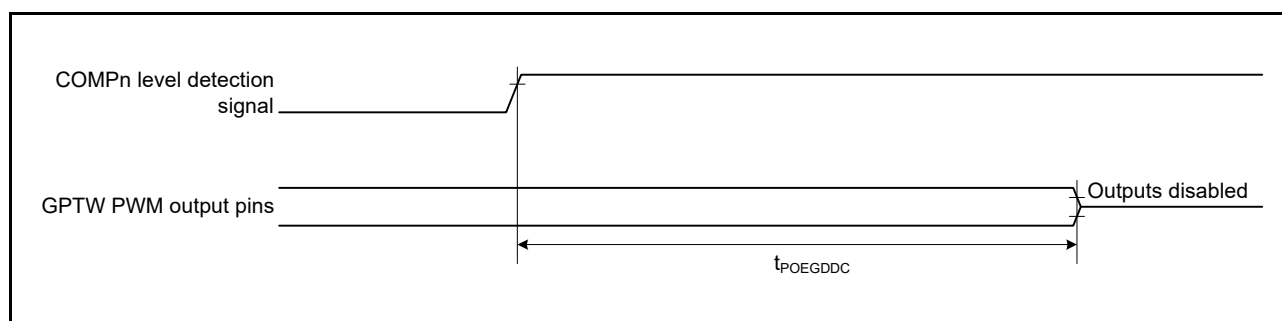


Figure 38.35 Output Disable Time for POEG in Response to Level Detection Signal from a Comparator (n = 0 to 2)

38.5.5.5 GPTW

Table 38.39 GPTW Timing

Item			Symbol	Min.	Max.	Unit*1	Test Conditions
GPTW	Input capture input pulse width	Single-edge setting	t_{GTICW}	1.5	—	t_{PBcyc}	Figure 38.36
		Both-edge setting		2.5	—		
	Input capture rise/fall time		t_{GTICr}/t_{GTICf}	—	0.1	$\mu s/V$	Figure 38.36
	External trigger input pulse width	Single-edge setting	t_{GTEW}	1.5	—	t_{PBcyc}	Figure 38.37
		Both-edge setting		2.5	—		
	Timer clock pulse width		t_{GTCKWH}/t_{GTCKWL}	1.5	—	t_{PBcyc}	Figure 38.38
	Timer clock rise/fall time		t_{GTCKr}/t_{GTCKf}	—	0.1	$\mu s/V$	Figure 38.38

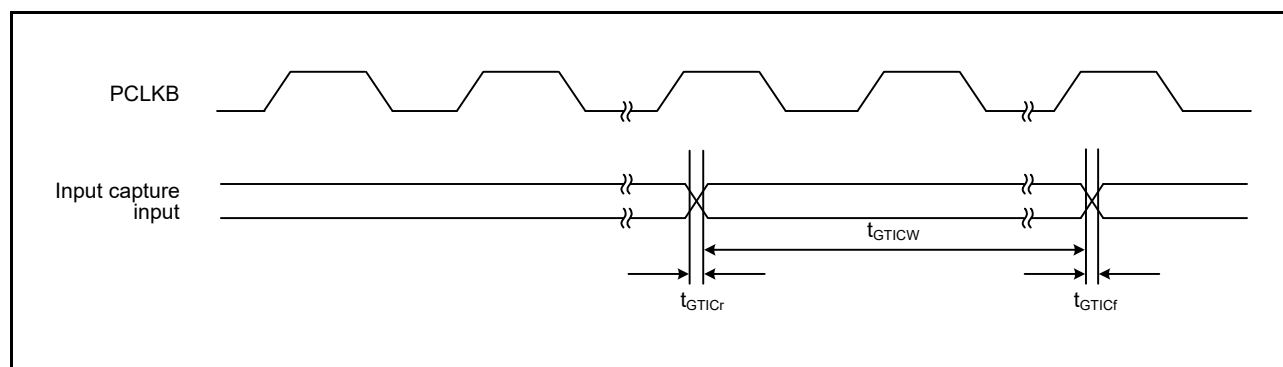
Note 1. t_{PBcyc} : PCLKB cycle

Figure 38.36 GPTW Input Capture Input Timing

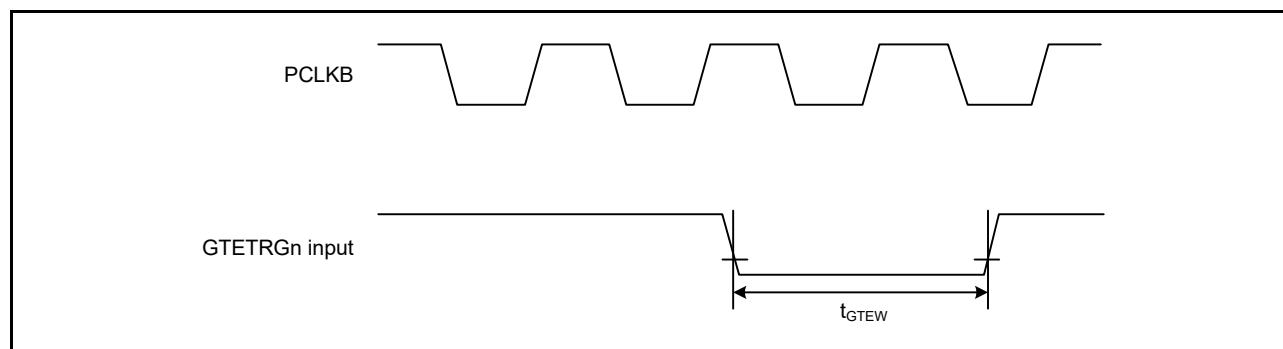


Figure 38.37 GPTW External Trigger Input Timing (n = A)

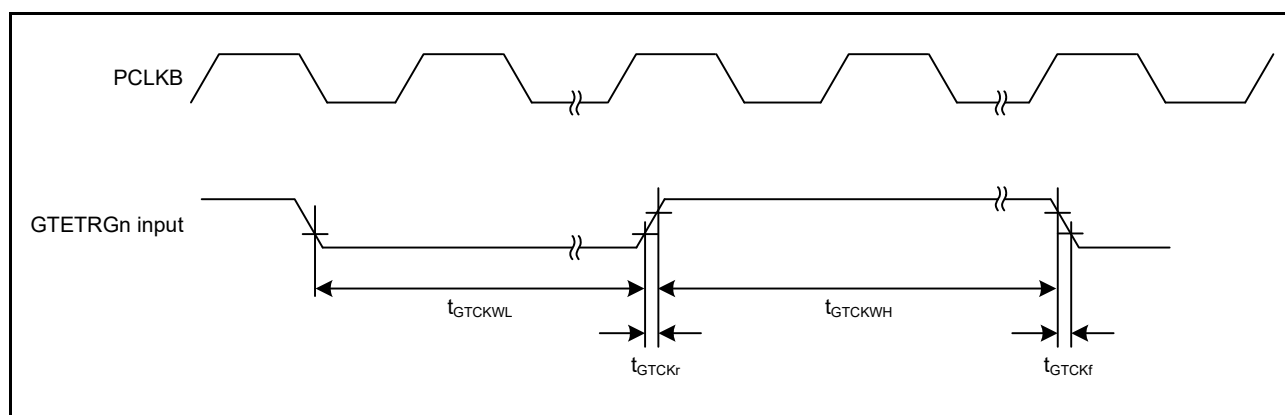


Figure 38.38 GPTW Clock Input Timing (n = A)

38.5.5.6 TMR

Table 38.40 TMR Timing

Item			Symbol	Min.	Max.	Unit *1	Test Conditions
TMR	Timer clock pulse width	Single-edge setting	t_{TMCWH}	1.5	—	t_{PBcyc}	Figure 38.39
		Both-edge setting	t_{TMCWL}	2.5	—		
	Timer clock rise/fall time		t_{TMCr} , t_{TMcf}	—	0.1	$\mu s/V$	

Note 1. t_{PBcyc} : PCLKB cycle

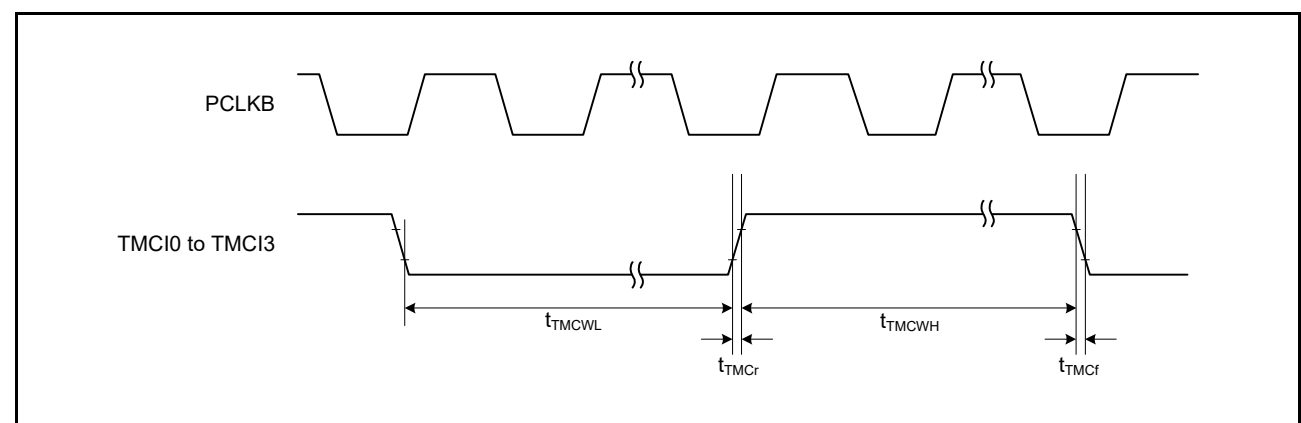
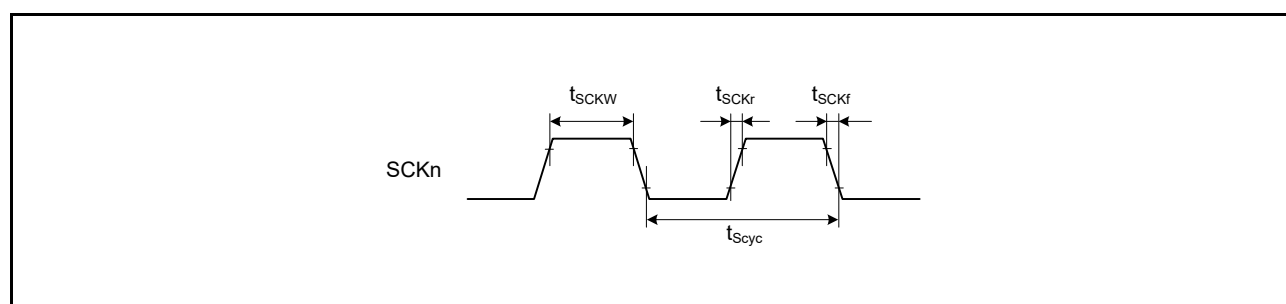


Figure 38.39 TMR Clock Input Timing

38.5.5.7 SCI

Table 38.41 SCI Timing

Item				Symbol	Min.	Max.	Unit *1	Test Conditions
SCI (channel 1, 5, 6, 12)	Input clock cycle time		Asynchronous	t_{Scyc}	4	—	t_{PBcyc}	Figure 38.40
			Clock synchronous		6	—		
	Input clock pulse width			t_{SCKW}	0.4	0.6	t_{Scyc}	
	Input clock rise time			t_{SCKr}	—	20	ns	
	Input clock fall time			t_{SCKf}	—	20	ns	
	Output clock cycle time	Asynchronous		t_{Scyc}	8	—	t_{PBcyc}	Figure 38.41
		Clock synchronous			4	—		
	Output clock pulse width			t_{SCKW}	0.4	0.6	t_{Scyc}	
	Output clock rise time			t_{SCKr}	—	20	ns	
	Output clock fall time			t_{SCKf}	—	20	ns	
Transmit data delay time (master)		Clock synchronous		t_{TXD}	—	40	ns	
Transmit data delay time (slave)	Clock synchronous	$4.0\text{ V} \leq \text{VCC}$	—		40	ns		
		$2.7\text{ V} \leq \text{VCC}$	—		65	ns		
Receive data setup time (master)	Clock synchronous	$4.0\text{ V} \leq \text{VCC}$	t_{RXS}	40	—	ns		
		$2.7\text{ V} \leq \text{VCC}$		65	—	ns		
Receive data setup time (slave)	Clock synchronous			40	—	ns		
Receive data hold time		Clock synchronous		t_{RXH}	40	—	ns	

Note 1. t_{PBcyc} : PCLKB cycle**Figure 38.40 SCK Clock Input Timing (n = 1, 5, 6, 12)**

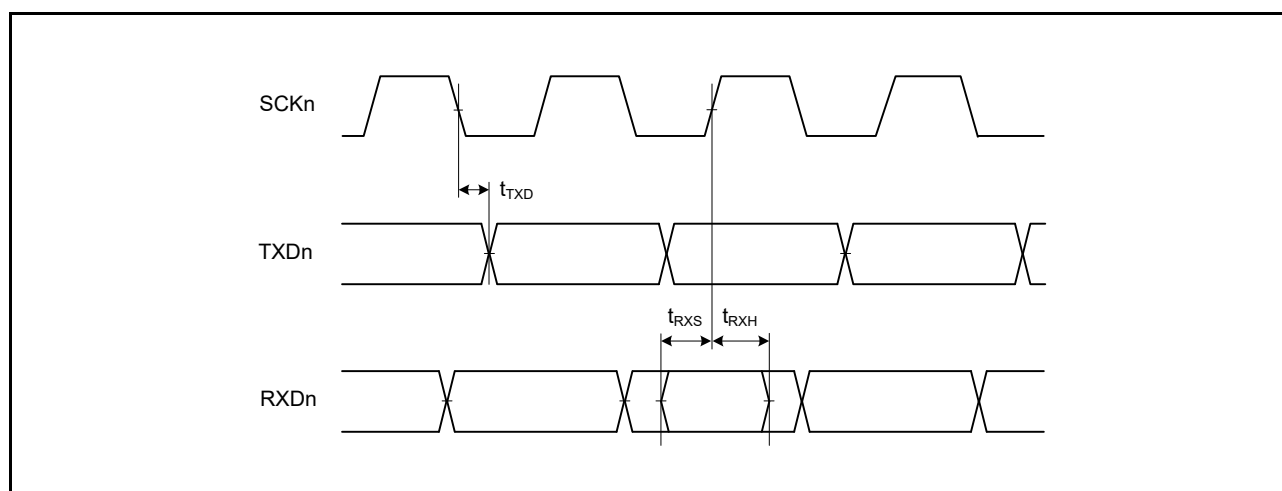


Figure 38.41 SCI Input/Output Timing: Clock Synchronous Mode (n = 1, 5, 6, 12)

Table 38.42 Simple I²C Timing

Item		Symbol	Min.	Max.	Unit	Test Conditions
Simple I ² C (standard mode)	SDA rise time	t_{Sr}	—	1000	ns	Figure 38.42
	SDA fall time	t_{Sf}	—	300	ns	
	SDA spike pulse removal time	t_{SP}	0	$4 \times t_{PBcyc}$	ns	
	Data setup time	t_{SDAS}	250	—	ns	
	Data hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load	C_b^{*1}	—	400	pF	
Simple I ² C (fast mode)	SDA rise time	t_{Sr}	—	300	ns	Figure 38.42
	SDA fall time	t_{Sf}	—	300	ns	
	SDA spike pulse removal time	t_{SP}	0	$4 \times t_{PBcyc}$	ns	
	Data setup time	t_{SDAS}	100	—	ns	
	Data hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load	C_b^{*1}	—	400	pF	

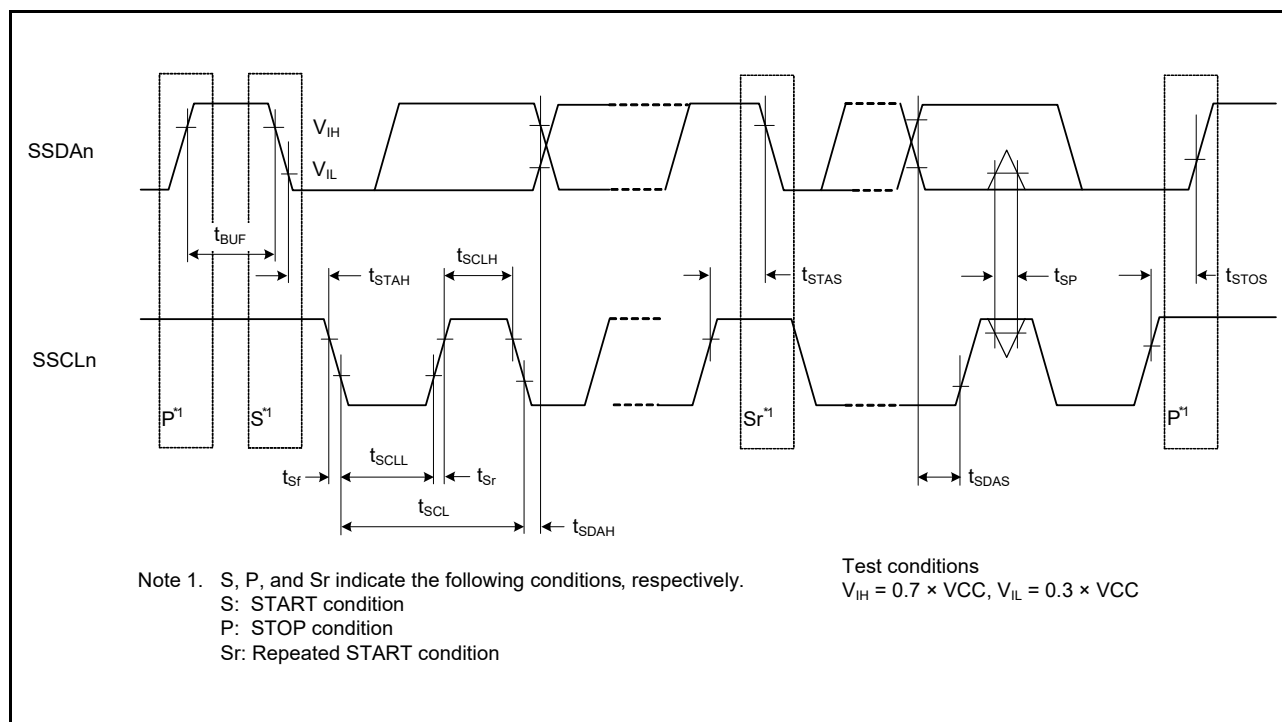
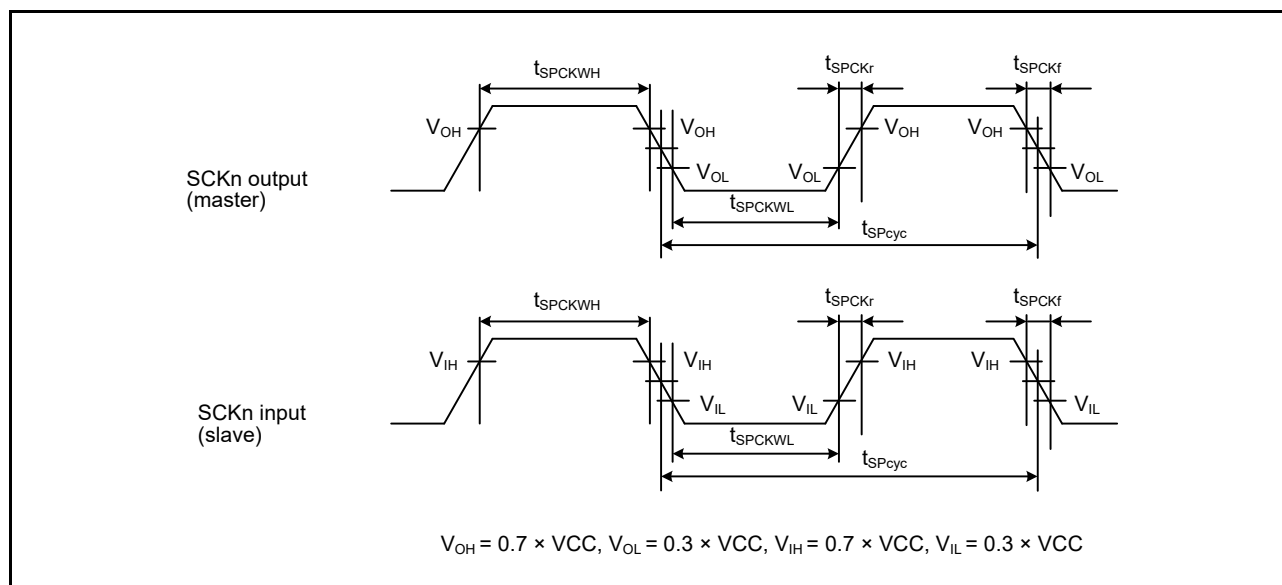
Note: t_{PBcyc} : PCLKB cycleNote 1. C_b is the total capacitance of the bus lines.**Figure 38.42 Output Timing and Simple I²C Bus Interface Input/Output Timing (n = 1, 5, 6, 12)**

Table 38.43 Simple SPI Timing

Item			Symbol	Min.	Max.	Unit*1	Test Conditions
Simple SPI	SCK clock cycle output (master)		t_{SPCyc}	4	—	t_{PBcyc}	Figure 38.43
	SCK clock cycle input (slave)			6	—	t_{PBcyc}	
	SCK clock high pulse width		t_{SPCKWH}	0.4	0.6	t_{SPCyc}	
	SCK clock low pulse width		t_{SPCKWL}	0.4	0.6	t_{SPCyc}	
	SCK clock rise/fall time		t_{SPCKr} , t_{SPCKf}	—	20	ns	
	Data input setup time (master)	4.0 V ≤ VCC	t_{SU}	40	—	ns	Figure 38.44, Figure 38.45
		2.7 V ≤ VCC		65	—		
	Data input setup time (slave)			40	—		
	Data input hold time		t_H	40	—	ns	
	SS input setup time		t_{LEAD}	1	—	t_{SPCyc}	
	SS input hold time		t_{LAG}	1	—	t_{SPCyc}	
	Data output delay time (master)		t_{OD}	—	40	ns	
	Data output delay time (slave)	4.0 V ≤ VCC		—	40		
		2.7 V ≤ VCC		—	65		
	Data output hold time (master)		t_{OH}	−10	—	ns	
	Data output hold time (slave)			−10	—		
	Data rise/fall time		t_{Dr} , t_{Df}	—	20	ns	
	SSL input rise/fall time		t_{SSLr} , t_{SSLf}	—	20	ns	
	Slave access time		t_{SA}	—	6	t_{PBcyc}	Figure 38.46, Figure 38.47
	Slave output release time		t_{REL}	—	6	t_{PBcyc}	

Note 1. t_{PBcyc} : PCLKB cycle**Figure 38.43 Simple SPI Clock Timing (n = 1, 5, 6, 12)**

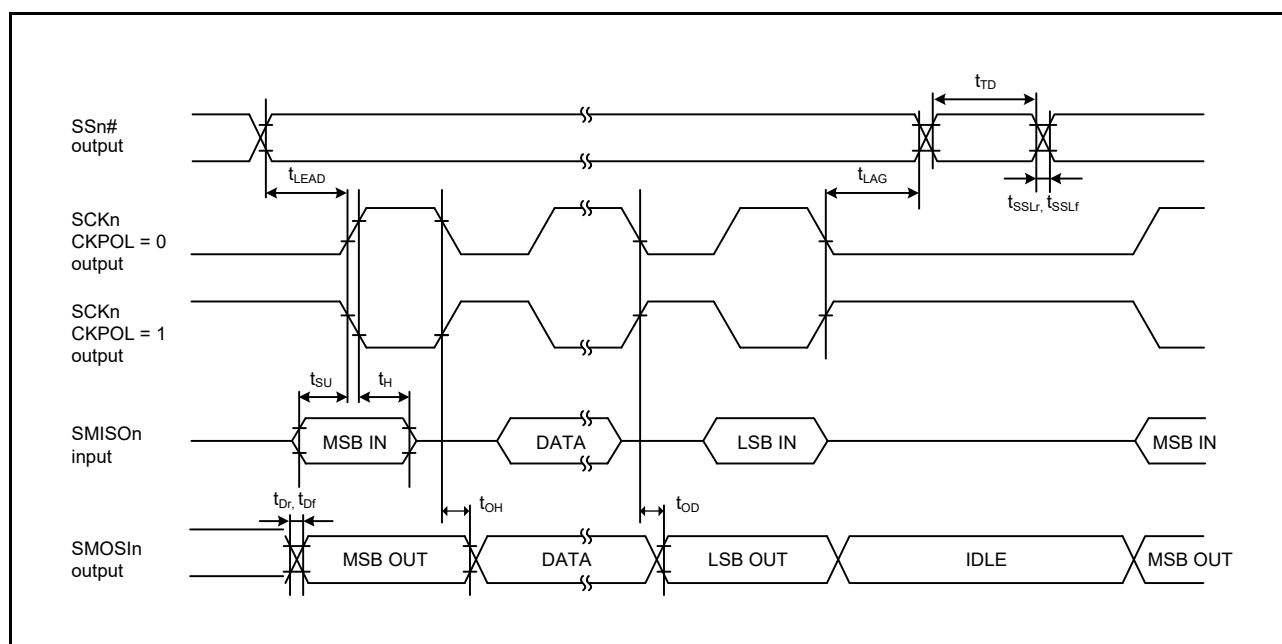


Figure 38.44 Simple SPI Timing (Master, CKPH = 1) (n = 1, 5, 6, 12)

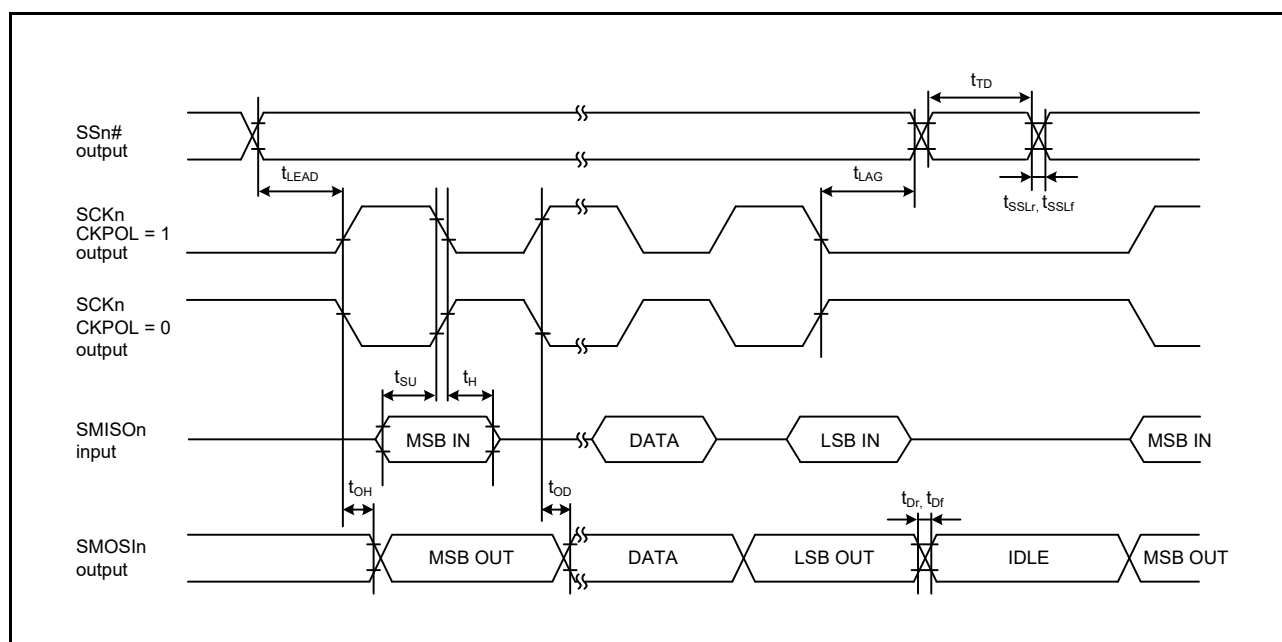


Figure 38.45 Simple SPI Timing (Master, CKPH = 0) (n = 1, 5, 6, 12)

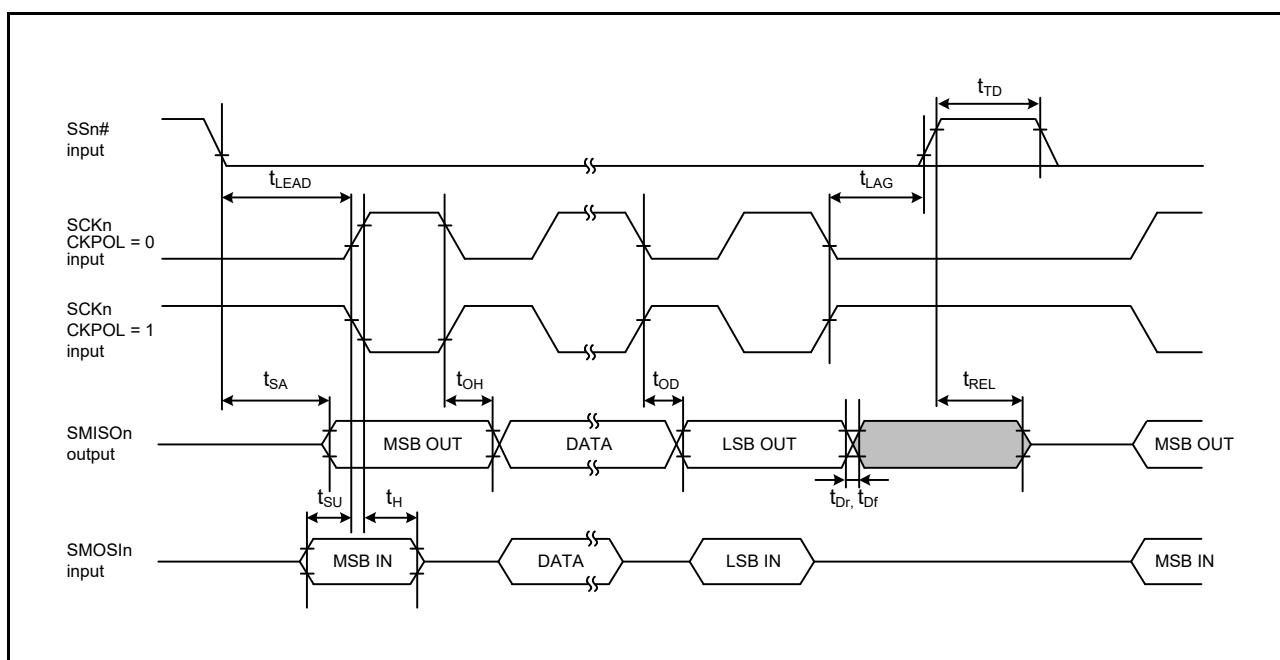


Figure 38.46 Simple SPI Clock Timing (Slave, CKPH = 1) (n = 1, 5, 6, 12)

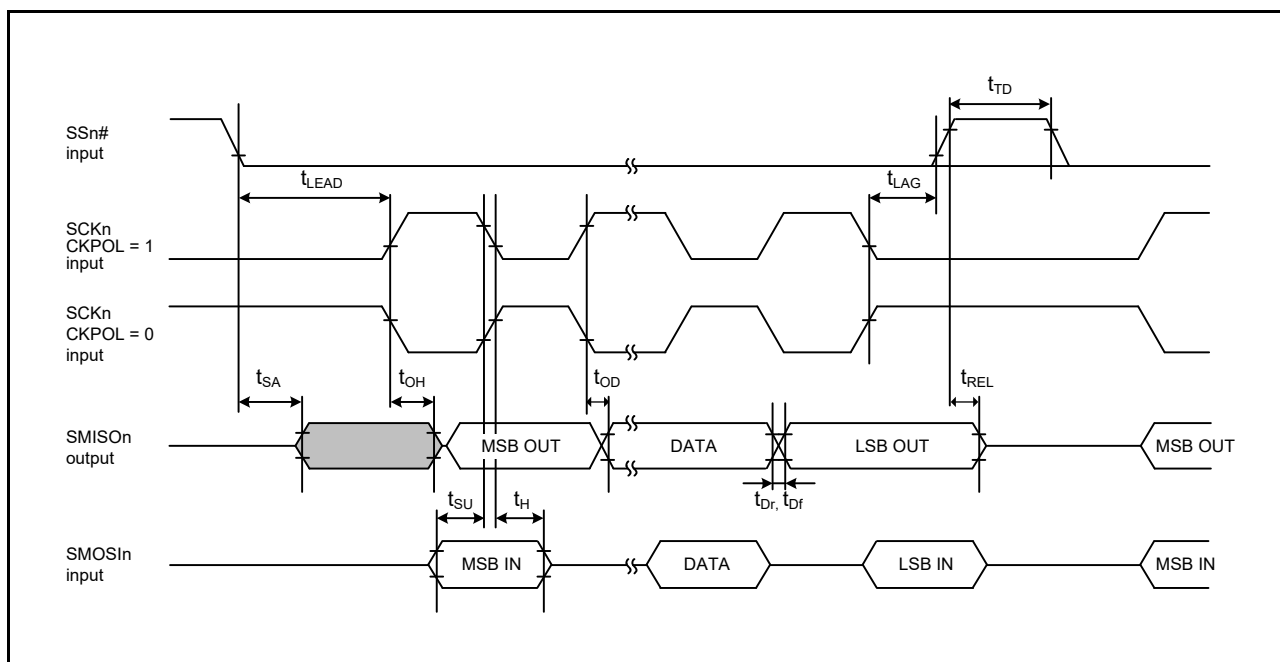


Figure 38.47 Simple SPI Timing (Slave, CKPH = 0) (n = 1, 5, 6, 12)

38.5.5.8 RIIC

Table 38.44 RIIC Timing

Item		Symbol	Min. *1	Max.	Unit	Test Conditions
RIIC (standard mode, SMBus)	SCL cycle time	t_{SCL}	$6 (12) \times t_{IICcyc} + 1300$	—	ns	Figure 38.48
	SCL high pulse width	t_{SCLH}	$3 (6) \times t_{IICcyc} + 300$	—	ns	
	SCL low pulse width	t_{SCLL}	$3 (6) \times t_{IICcyc} + 300$	—	ns	
	SCL, SDA rise time	t_{Sr}	—	1000	ns	
	SCL, SDA fall time	t_{Sf}	—	300	ns	
	SCL, SDA spike pulse removal time	t_{SP}	0	$1 (4) \times t_{IICcyc}$	ns	
	SDA bus free time	t_{BUF}	$3 (6) \times t_{IICcyc} + 300$	—	ns	
	START condition input hold time	t_{STAH}	$t_{IICcyc} + 300$	—	ns	
	Repeated START condition input setup time	t_{STAS}	1000	—	ns	
	STOP condition input setup time	t_{STOS}	1000	—	ns	
	Data setup time	t_{SDAS}	$t_{IICcyc} + 50$	—	ns	
	Data hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load	C_b^{*2}	—	400	pF	
RIIC (fast mode)	SCL cycle time	t_{SCL}	$6 (12) \times t_{IICcyc} + 600$	—	ns	Figure 38.48
	SCL high pulse width	t_{SCLH}	$3 (6) \times t_{IICcyc} + 300$	—	ns	
	SCL low pulse width	t_{SCLL}	$3 (6) \times t_{IICcyc} + 300$	—	ns	
	SCL, SDA rise time	t_{Sr}	$20 \times (\text{External pull-up voltage}/5.5 \text{ V})$	300	ns	
	SCL, SDA fall time	t_{Sf}	$20 \times (\text{External pull-up voltage}/5.5 \text{ V})$	300	ns	
	SCL, SDA spike pulse removal time	t_{SP}	0	$1 (4) \times t_{IICcyc}$	ns	
	SDA bus free time	t_{BUF}	$3 (6) \times t_{IICcyc} + 300$	—	ns	
	START condition input hold time	t_{STAH}	$t_{IICcyc} + 300$	—	ns	
	Repeated START condition input setup time	t_{STAS}	300	—	ns	
	STOP condition input setup time	t_{STOS}	300	—	ns	
	Data setup time	t_{SDAS}	$t_{IICcyc} + 50$	—	ns	
	Data hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load	C_b^{*2}	—	400	pF	

Note: t_{IICcyc} : RIIC internal reference count clock (IIC ϕ) cycle

Note 1. The value in parentheses is used when the ICMR3.NF[1:0] bits are set to 11b while a digital filter is enabled with the ICFER.NFE bit = 1.

Note 2. C_b is the total capacitance of the bus lines.

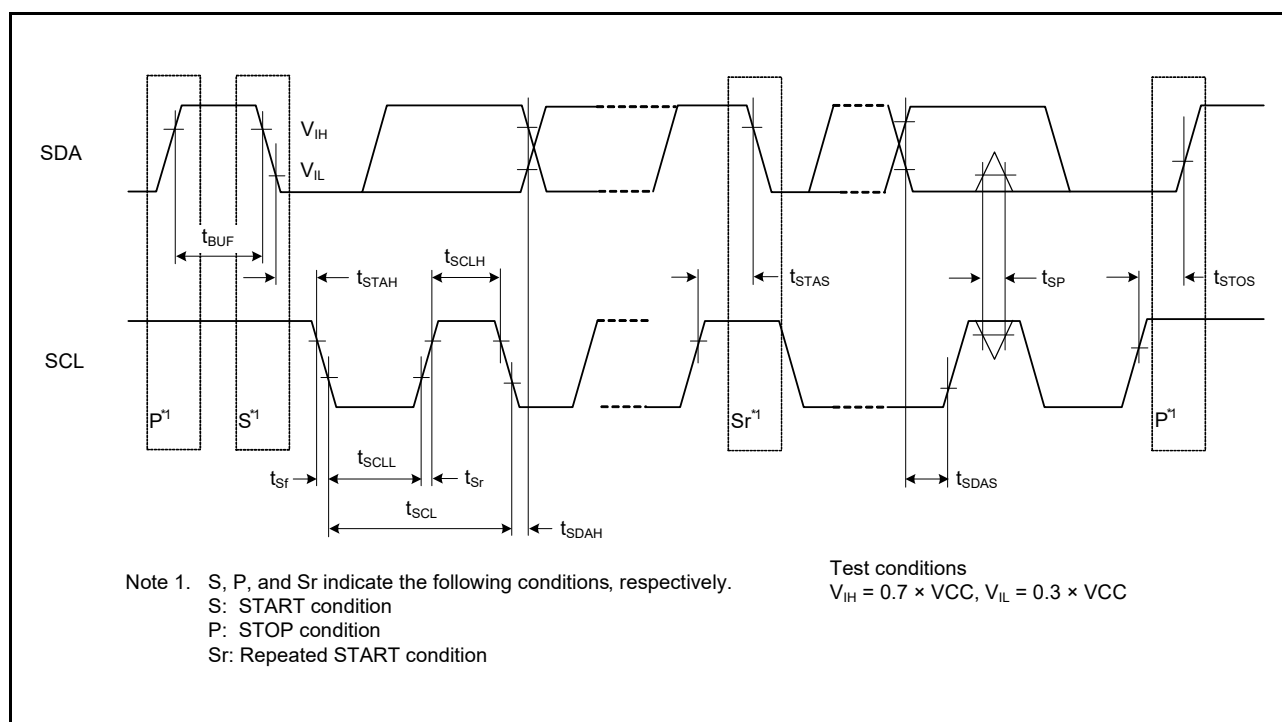


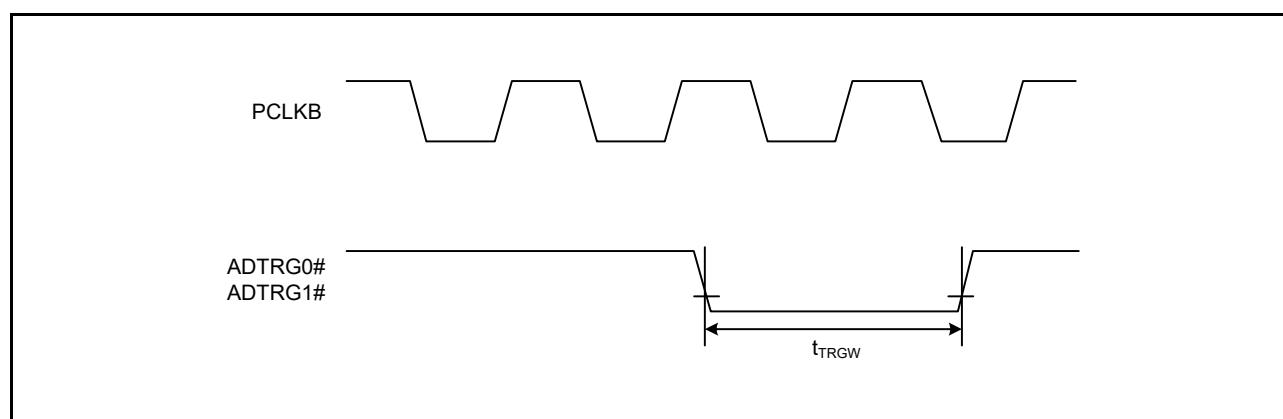
Figure 38.48 IIC Bus Interface Input/Output Timing

38.5.5.9 A/D Converter Trigger

Table 38.45 A/D Converter Trigger Timing

Item		Symbol	Min.	Max.	Unit *1	Test Conditions
A/D converter	Trigger input pulse width	t_{TRGW}	1.5	—	t_{PBcyc}	Figure 38.49

Note 1. t_{PBcyc} : PCLKB cycle

**Figure 38.49 A/D Converter External Trigger Input Timing**

38.5.5.10 CAC

Table 38.46 CAC Timing

Item			Symbol	Min.	Max.	Unit *1	Test Conditions
CAC	CACREF input pulse width	$t_{PBcyc} \leq t_{cac}^{*2}$	t_{CACREF}	$4.5 \times t_{cac} + 3 \times t_{PBcyc}$	—	ns	
		$t_{PBcyc} > t_{cac}^{*2}$		$5 \times t_{cac} + 6.5 \times t_{PBcyc}$			
	CACREF input rise/fall time		$t_{CACREFr}$, $t_{CACREFf}$	—	0.1	$\mu s/V$	

Note 1. t_{PBcyc} : PCLKB cycle

Note 2. t_{cac} : CAC count clock source cycle

38.5.5.11 CLKOUT

Table 38.47 CLKOUT Timing

	Item	Symbol	Min.	Max.	Unit	Test Conditions
CLKOUT	CLKOUT pin output cycle*2	t_{Cyc}	62.5	—	ns	Figure 38.50
	CLKOUT pin high pulse width*1	t_{CH}	15	—	ns	
	CLKOUT pin low pulse width*1	t_{CL}	15	—	ns	
	CLKOUT pin output rise time	t_{Cr}	—	12	ns	
	CLKOUT pin output fall time	t_{Cf}	—	12	ns	

Note 1. When the LOCO is selected as the clock output source (CKOCR.CKOSEL[3:0] bits = 0000b), set the clock output division ratio selection to divided by 2 (CKOCR.CKODIV[2:0] bits = 001b).

Note 2. When the XTAL external clock input or an oscillator is used with divided by 1 (CKOCR.CKOSEL[3:0] bits = 010b and CKOCR.CKODIV[2:0] bits = 000b) to output from CLKOUT, the above should be satisfied with an input duty cycle of 45 to 55%.

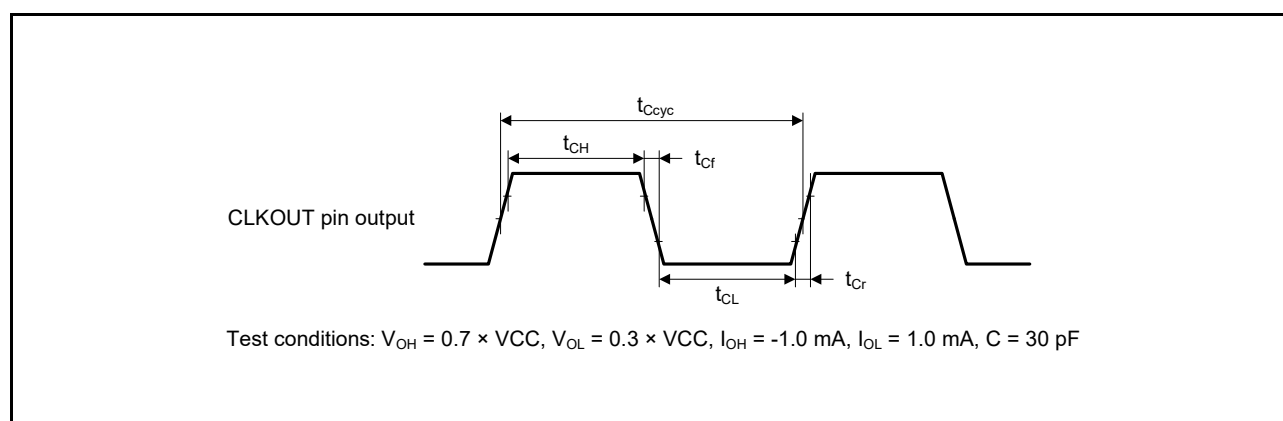


Figure 38.50 CLKOUT Output Timing

38.6 A/D Conversion Characteristics

Table 38.48 A/D Conversion Characteristics (1)

Conditions: 1 to 64 MHz, $4.5\text{ V} \leq V_{CC} \leq 5.5\text{ V}$, $4.5\text{ V} \leq V_{REFH0} \leq V_{CC}$, $V_{SS} = V_{REFL0} = 0\text{ V}$, $T_a = -40$ to 125°C , signal source impedance = $0.5\text{ k}\Omega$

Item		Min.	Typ.	Max.	Unit	Test Conditions
Frequency		1	—	64	MHz	
Resolution		—	—	12	Bit	
Conversion time*1		0.50 (0.164)	—	—	μs	ADSSTRn = 0Ah
Analog input capacitance	Cs	—	—	9*2	pF	
Analog input resistance	Rs	—	—	1.3*2	$\text{k}\Omega$	
Analog input effective range		0	—	VREFH0	V	
Offset error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 4.5	LSB	$V_{CC} = V_{REFH0}$
		—	± 1.0	± 5.0	LSB	$V_{CC} > V_{REFH0}$
	When the reference voltage is VCC or VSS	—	± 1.0	± 7.0	LSB	
Full-scale error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 4.5	LSB	$V_{CC} = V_{REFH0}$
		—	± 1.0	± 5.0	LSB	$V_{CC} > V_{REFH0}$
	When the reference voltage is VCC or VSS	—	± 1.0	± 7.0	LSB	
Quantization error		—	± 0.5	—	LSB	
Absolute accuracy	When the reference voltage is VREFH0 or VREFL0	—	± 2.5	± 5.0	LSB	$V_{CC} = V_{REFH0}$
		—	± 2.5	± 5.5	LSB	$V_{CC} > V_{REFH0}$
	When the reference voltage is VCC or VSS	—	± 2.5	± 7.5	LSB	
DNL differential nonlinearity error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 1.5	LSB	$V_{CC} = V_{REFH0}$
		—	± 1.0	± 2.0	LSB	$V_{CC} > V_{REFH0}$
	When the reference voltage is VCC or VSS	—	± 1.0	± 1.5	LSB	
INL integral nonlinearity error	When the reference voltage is VREFH0 or VREFL0	—	± 1.5	± 3.0	LSB	$V_{CC} = V_{REFH0}$
		—	± 1.5	± 3.5	LSB	$V_{CC} > V_{REFH0}$
	When the reference voltage is VCC or VSS	—	± 1.5	± 3.0	LSB	

Note: The listed values apply when all pins other than one analog input pin that is a target of conversion are at fixed levels, and all peripheral modules (except for the unit handling the A/D conversion) and the CPU are stopped. Other conditions may lead to accuracy and errors falling beyond the ranges in the above specifications.

Note: Absolute accuracy includes quantization errors. Offset error, full-scale error, DNL differential nonlinearity error, and INL integral nonlinearity error do not include quantization errors.

Note 1. The conversion time is the sum of the sampling time and the comparison time. The values in () show the sampling times.

Note 2. The values are reference values.

Table 38.49 A/D Conversion Characteristics (2)

Conditions: 1 to 48 MHz, $4.5\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$, $4.5\text{ V} \leq \text{VREFH0} \leq \text{VCC}$, $\text{VSS} = \text{VREFL0} = 0\text{ V}$, $T_a = -40$ to 125°C , signal source impedance = $0.5\text{ k}\Omega$

Item		Min.	Typ.	Max.	Unit	Test Conditions
Frequency		1	—	48	MHz	
Resolution		—	—	12	Bit	
Conversion time*1		0.67 (0.219)	—	—	μs	$\text{ADSSTRn} = 0\text{Ah}$
Analog input capacitance	Cs	—	—	9*2	pF	
Analog input resistance	Rs	—	—	1.3*2	$\text{k}\Omega$	
Analog input effective range		0	—	VREFH0	V	
Offset error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 4.5	LSB	$\text{VCC} = \text{VREFH0}$
		—	± 1.0	± 5.0	LSB	$\text{VCC} > \text{VREFH0}$
	When the reference voltage is VCC or VSS	—	± 1.0	± 7.0	LSB	
Full-scale error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 4.5	LSB	$\text{VCC} = \text{VREFH0}$
		—	± 1.0	± 5.0	LSB	$\text{VCC} > \text{VREFH0}$
	When the reference voltage is VCC or VSS	—	± 1.0	± 7.0	LSB	
Quantization error		—	± 0.5	—	LSB	
Absolute accuracy	When the reference voltage is VREFH0 or VREFL0	—	± 2.5	± 5.0	LSB	$\text{VCC} = \text{VREFH0}$
		—	± 2.5	± 5.5	LSB	$\text{VCC} > \text{VREFH0}$
	When the reference voltage is VCC or VSS	—	± 2.5	± 7.5	LSB	
DNL differential nonlinearity error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 1.5	LSB	$\text{VCC} = \text{VREFH0}$
		—	± 1.0	± 2.0	LSB	$\text{VCC} > \text{VREFH0}$
	When the reference voltage is VCC or VSS	—	± 1.0	± 1.5	LSB	
INL integral nonlinearity error	When the reference voltage is VREFH0 or VREFL0	—	± 1.5	± 3.0	LSB	$\text{VCC} = \text{VREFH0}$
		—	± 1.5	± 3.5	LSB	$\text{VCC} > \text{VREFH0}$
	When the reference voltage is VCC or VSS	—	± 1.5	± 3.0	LSB	

Note: The listed values apply when all pins other than one analog input pin that is a target of conversion are at fixed levels, and all peripheral modules (except for the unit handling the A/D conversion) and the CPU are stopped. Other conditions may lead to accuracy and errors falling beyond the ranges in the above specifications.

Note: Absolute accuracy includes quantization errors. Offset error, full-scale error, DNL differential nonlinearity error, and INL integral nonlinearity error do not include quantization errors.

Note 1. The conversion time is the sum of the sampling time and the comparison time. The values in () show the sampling times.

Note 2. The values are reference values.

Table 38.50 A/D Conversion Characteristics (3)

Conditions: 1 to 48 MHz, $2.7\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$, $2.7\text{ V} \leq \text{VREFH0} \leq \text{VCC}$, $\text{VSS} = \text{VREFL0} = 0\text{ V}$, $T_a = -40$ to 125°C , signal source impedance = $0.5\text{ k}\Omega$

Item		Min.	Typ.	Max.	Unit	Test Conditions
Frequency		1	—	48	MHz	
Resolution		—	—	12	Bit	
Conversion time*1		0.79 (0.344)	—	—	μs	ADSSTRn = 10h
Analog input capacitance	Cs	—	—	9*2	pF	
Analog input resistance	Rs	—	—	1.9*2	$\text{k}\Omega$	
Analog input effective range		0	—	VREFH0	V	
Offset error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 4.5	LSB	VCC = VREFH0
		—	± 1.0	± 6.0	LSB	VCC > VREFH0
	When the reference voltage is VCC or VSS	—	± 1.0	± 8.5	LSB	
Full-scale error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 4.5	LSB	VCC = VREFH0
		—	± 1.0	± 6.0	LSB	VCC > VREFH0
	When the reference voltage is VCC or VSS	—	± 1.0	± 8.5	LSB	
Quantization error		—	± 0.5	—	LSB	
Absolute accuracy	When the reference voltage is VREFH0 or VREFL0	—	± 2.5	± 5.5	LSB	VCC = VREFH0
		—	± 2.5	± 7.0	LSB	VCC > VREFH0
	When the reference voltage is VCC or VSS	—	± 2.5	± 9.5	LSB	
DNL differential nonlinearity error	When the reference voltage is VREFH0 or VREFL0	—	± 1.0	± 1.5	LSB	VCC = VREFH0
		—	± 1.0	± 2.0	LSB	VCC > VREFH0
	When the reference voltage is VCC or VSS	—	± 1.0	± 1.5	LSB	
INL integral nonlinearity error	When the reference voltage is VREFH0 or VREFL0	—	± 1.5	± 3.0	LSB	VCC = VREFH0
		—	± 1.5	± 4.0	LSB	VCC > VREFH0
	When the reference voltage is VCC or VSS	—	± 1.5	± 3.0	LSB	

Note: The listed values apply when all pins other than one analog input pin that is a target of conversion are at fixed levels, and all peripheral modules (except for the unit handling the A/D conversion) and the CPU are stopped. Other conditions may lead to accuracy and errors falling beyond the ranges in the above specifications.

Note: Absolute accuracy includes quantization errors. Offset error, full-scale error, DNL differential nonlinearity error, and INL integral nonlinearity error do not include quantization errors.

Note 1. The conversion time is the sum of the sampling time and the comparison time. The values in () show the sampling times.

Note 2. The values are reference values.

Table 38.51 Signal Source Impedance and Required Minimum Sampling Time

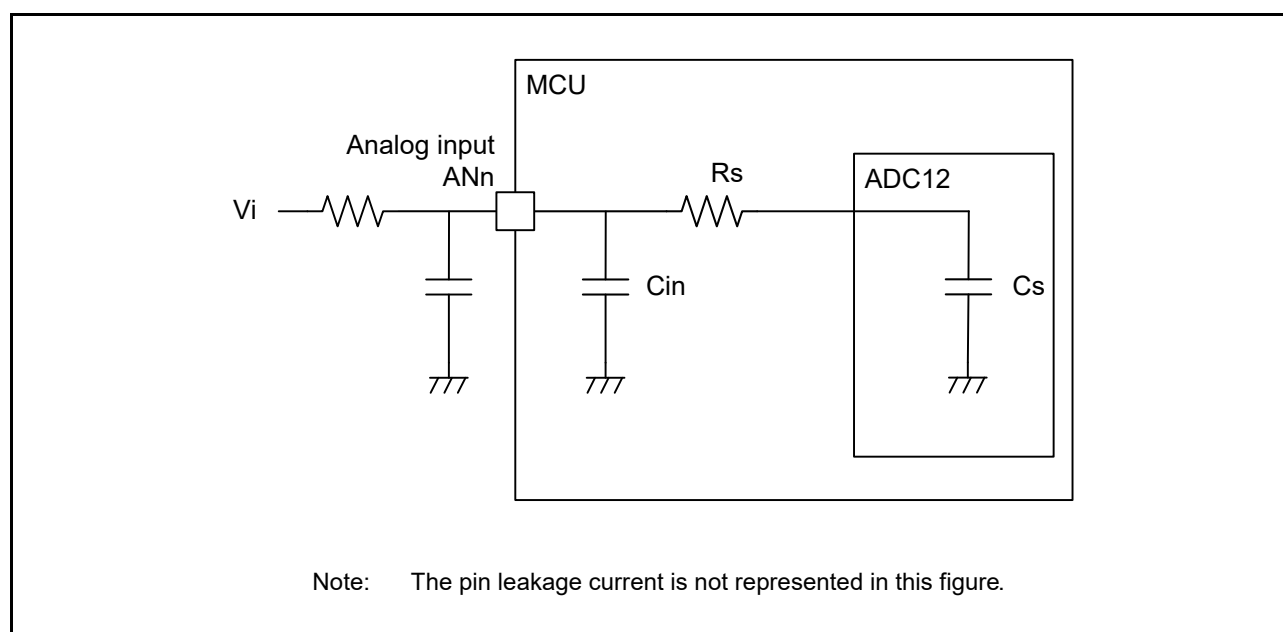
Signal source impedance [kΩ]	Required Sampling Time [ADCLK]		
	4.5 V ≤ VCC ≤ 5.5 V 4.5 V ≤ VREFH0 ≤ VCC		2.7 V ≤ VCC ≤ 5.5 V 2.7 V ≤ VREFH0 ≤ VCC
	ADCLK = 64 MHz	ADCLK = 48 MHz	ADCLK = 48 MHz
0.5	10	10	16
2.2	25	24	24
4.7	52	43	43
10	101	80	80
22	210	160	160

Table 38.52 A/D Internal Reference Voltage Characteristics

Item	Min.	Typ.	Max.	Unit	Test Conditions
Internal reference voltage input channel*1	1.42	1.48	1.54	V	
Time until the internal reference voltage output becomes stable	—	—	5	μs	
Sampling time*2	3.9	—	—	μs	

Note 1. The A/D internal reference voltage indicates the voltage when the internal reference voltage is input to the A/D converter.

Note 2. Set the S12AD1.ADSSTRO register such that the sampling time of the 12-bit A/D converter satisfies this specification.

**Figure 38.51 Equivalent Circuit of an Analog Input**

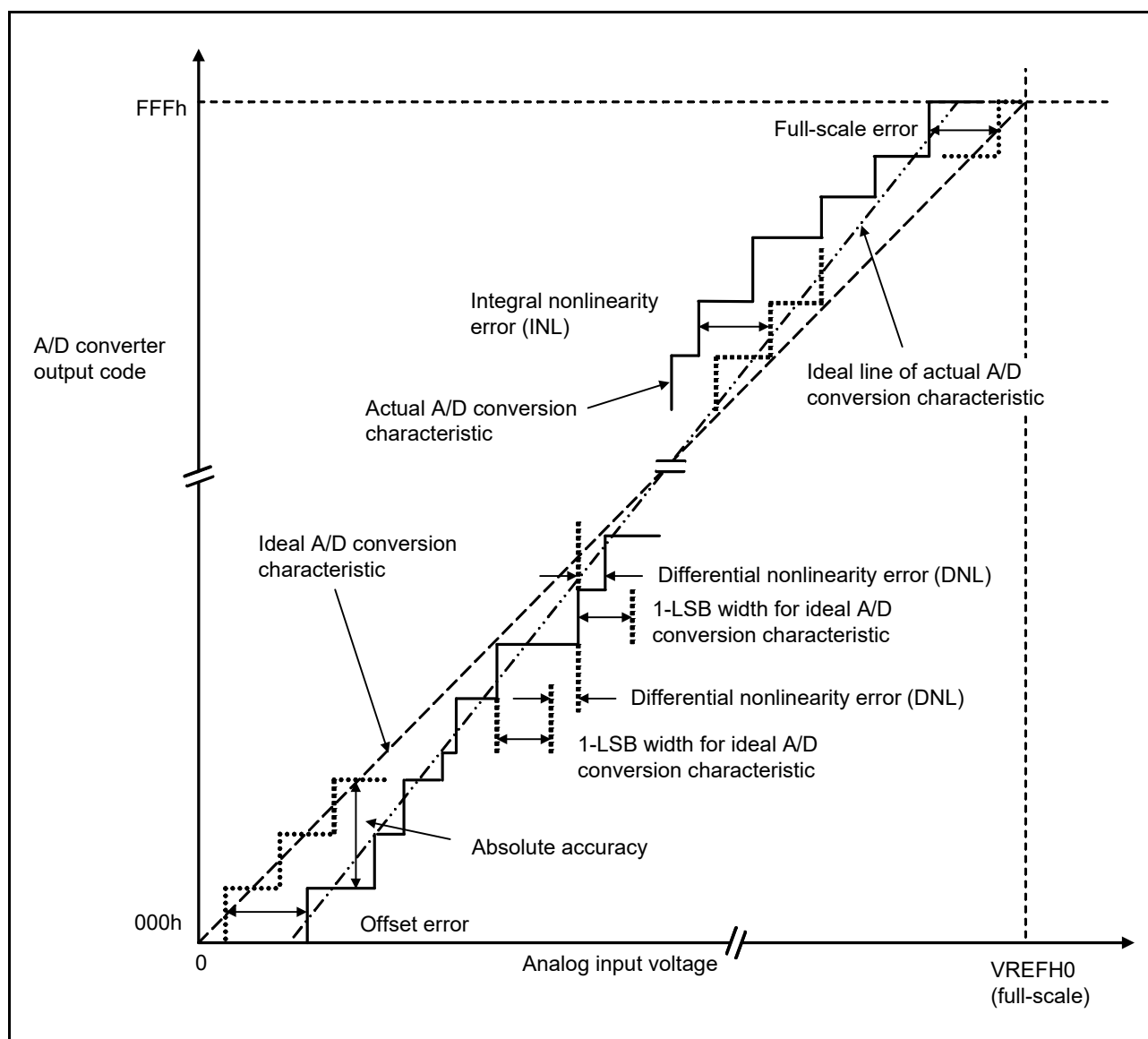


Figure 38.52 Illustration of A/D Converter Characteristic Terms

Absolute accuracy

Absolute accuracy is the difference between output code based on the theoretical A/D conversion characteristics, and the actual A/D conversion result. When measuring absolute accuracy, the voltage at the midpoint of the width of analog input voltage (1-LSB width), that can meet the expectation of outputting an equal code based on the theoretical A/D conversion characteristics, is used as an analog input voltage. For example, if 12-bit resolution is used and if reference voltage ($V_{REFH0} = 3.072\text{ V}$), then 1-LSB width becomes 0.75 mV, and 0 mV, 0.75 mV, 1.5 mV, ... are used as analog input voltages.

If analog input voltage is 6 mV, absolute accuracy = ± 5 LSB means that the actual A/D conversion result is in the range of 003h to 00Dh though an output code, 008h, can be expected from the theoretical A/D conversion characteristics.

Integral nonlinearity error (INL)

Integral nonlinearity error is the maximum deviation between the ideal line when the measured offset and full-scale errors are zeroed, and the actual output code.

Differential nonlinearity error (DNL)

Differential nonlinearity error is the difference between 1-LSB width based on the ideal A/D conversion characteristics and the width of the actual output code.

Offset error

Offset error is the difference between a transition point of the ideal first output code and the actual first output code.

Full-scale error

Full-scale error is the difference between a transition point of the ideal last output code and the actual last output code.

38.7 Programmable Gain Amplifier Characteristics

Table 38.53 Programmable Gain Amplifier Characteristics

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input offset voltage*1	V_{IO}	—	—	± 5.7	mV	
Input voltage range	V_{ISR}	0	—	VCC	V	
Output voltage range	V_{OR}	$0.07 \times VCC$	—	$0.93 \times VCC$	V	
Gain	G	4, 8, 16, 32				
Gain error*1	E_G	—	± 0.04	± 0.47	%	G = 4, 8
		—	± 0.02	± 0.62	%	G = 16
		—	± 0.03	± 1.23	%	G = 32
Slew rate	SR	6.1	—	—	V/ μ s	4.0 V $\leq$ VCC $\leq$ 5.5 V, G = 4, 8, 16
		3.4	—	—	V/ μ s	4.0 V $\leq$ VCC $\leq$ 5.5 V, G = 32
		2	—	—	V/ μ s	2.7 V $\leq$ VCC $\leq$ 4.0 V
Operation stabilization time	t_{start}	—	—	2.66	μ s	

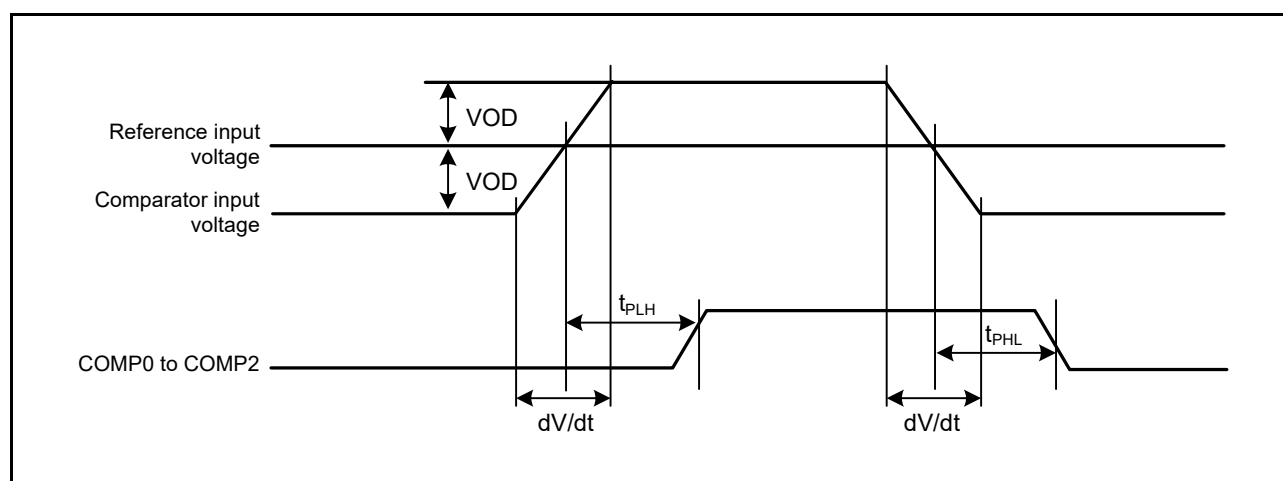
Note 1. The listed values apply when PGAGND is selected as the ground for the feedback resistor. If VSS is selected, the values may fall beyond the listed ranges.

38.8 Comparator Characteristics

Table 38.54 Comparator Characteristics

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Offset voltage	V_{IO}	—	± 5	± 10.3	mV	
Reference input voltage range	V_{ref}	0	—	VCC	V	
Input voltage slope*1	dV/dt	—	—	± 3.0	V/ μs	
Response time	t_{PLH}	—	50	76	ns	VOD = 100 mV CMPCTL.NFE = 0
	t_{PHL}	—	50	73	ns	
Stabilization wait time for input selection	t_{cwait}	74	—	—	ns	
Operation stabilization time	t_{cmp}	—	—	0.51	μs	

Note 1. If this characteristic is not satisfied, take measures described in section 34.4.6, Appearance of a Pulse in the Result of Comparison.

**Figure 38.53 Comparator Response Time**

38.9 D/A Conversion Characteristics

Table 38.55 D/A Conversion Characteristics

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Resolution	—	—	—	8	Bit	
Conversion time	t_{DCONV}	—	—	3.0	μs	35-pF capacitive load
Absolute accuracy	—	—	± 1.0	± 3.0	LSB	2-M Ω resistive load
	—	—	± 1.0	± 2.0	LSB	4-M Ω resistive load
Differential nonlinearity error (DNL)	—	—	± 1.0	± 2.0	LSB	
Output load resistance	—	4	—	—	M Ω	
Output load capacity	—	—	—	35	pF	
Output resistance	—	—	9.0	—	k Ω	

38.10 Temperature Sensor Characteristics

Table 38.56 Temperature Sensor Characteristics

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Relative accuracy	—	—	±1.5	—	°C	
Temperature slope	—	—	−3.3	—	mV/°C	
Output voltage (25°C)	—	—	1.05	—	V	
Temperature sensor start time	t _{START}	—	—	5	μs	
Sampling time*1	—	3.9	—	—	μs	

Note 1. Set the S12AD1.ADSSTRT register so that the sampling time of the 12-bit A/D converter meets this specification.

38.11 Power-On Reset Circuit and Voltage Detection Circuit Characteristics

Table 38.57 Power-On Reset Circuit and Voltage Detection Circuit Characteristics (1)

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Voltage detection level	Power-on reset (POR)*1	V _{POR}	1.35	1.50	1.65	V	Figure 38.54, Figure 38.55
	Voltage detection circuit (LVD0)*2	V _{det0_0}	3.67	3.85	3.97	V	Figure 38.56 At falling edge VCC
		V _{det0_1}	2.70	2.85	3.00		
		V _{det0_2}	2.37	2.53	2.67		
	Voltage detection circuit (LVD1)*3	V _{det1_0}	4.12	4.29	4.42	V	Figure 38.57 At falling edge VCC
		V _{det1_1}	3.98	4.16	4.28		
		V _{det1_2}	3.86	4.03	4.16		
		V _{det1_3}	3.68	3.86	3.98		
		V _{det1_4}	2.99	3.10	3.29		
		V _{det1_5}	2.89	3.00	3.19		
		V _{det1_6}	2.79	2.90	3.09		
		V _{det1_7}	2.68	2.80	2.98		
		V _{det1_8}	2.57	2.68	2.87		
		V _{det1_9}	2.47	2.59	2.67		
Voltage detection level	Voltage detection circuit (LVD2)*4	V _{det2_0}	4.08	4.32	4.48	V	Figure 38.58 At falling edge VCC
		V _{det2_1}	3.95	4.17	4.35		
		V _{det2_2}	3.82	4.03	4.22		
		V _{det2_3}	3.62	3.84	4.02		

Note: These characteristics apply when noise is not superimposed on the power supply. When a setting is made so that the voltage detection level overlaps with that of the voltage detection circuit (LVD2), it cannot be specified which of LVD1 and LVD2 is used for voltage detection.

Note 1. The MCU starts operation after the release from a power-on reset. The electrical characteristics of the voltage detection circuit are guaranteed when the operating voltage is at least 2.7 V.

Note 2. n in the symbol V_{det0_n} denotes the value of the LDSEL1[1:0] bits.

Note 3. n in the symbol V_{det1_n} denotes the value of the LVDLVLRLVD1LVL[3:0] bits.

Note 4. n in the symbol V_{det2_n} denotes the value of the LVDLVLRLVD2LVL[1:0] bits.

Table 38.58 Power-On Reset Circuit and Voltage Detection Circuit Characteristics (2)

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Wait time after power-on reset cancellation	At normal startup*1	t _{POR}	—	31.5	—	ms	Figure 38.55
	During fast startup time*2	t _{POR}	—	5.0	—		
Wait time after voltage monitoring 0 reset cancellation		t _{LVD0}	—	860	—	μs	Figure 38.56
Wait time after voltage monitoring 1 reset cancellation	LVD0 disabled*4	t _{LVD1}	—	160	—	μs	Figure 38.57
	LVD0 enabled*5		—	860	—	μs	
Wait time after voltage monitoring 2 reset cancellation	LVD0 disabled*4	t _{LVD2}	—	160	—	μs	Figure 38.58
	LVD0 enabled*5		—	860	—	μs	
PDR response delay time		t _{det}	—	—	500	μs	Figure 38.54
LVD0 response delay time			—	—	500	μs	Figure 38.54
LVD1 response delay time			—	—	360	μs	Figure 38.54
LVD2 response delay time			—	—	600	μs	Figure 38.54
POR/LVD0 minimum VCC down time*3		t _{VOFF}	500	—	—	μs	Figure 38.54, VCC = 1.0 V or above
LVD1 minimum VCC down time*3			300	—	—	μs	Figure 38.54, VCC = 1.0 V or above
LVD2 minimum VCC down time*3			600	—	—	μs	Figure 38.54, VCC = 1.0 V or above
Power-on reset enable time		t _{W(POR)}	1	—	—	ms	Figure 38.55, VCC = below 1.0 V
LVD1 operation stabilization time (after LVD is enabled)		t _{d(E-A)}	—	—	300	μs	Figure 38.57
LVD2 operation stabilization time (after LVD is enabled)		t _{d(E-A)}	—	—	1200	μs	Figure 38.58
Hysteresis width (power-on rest (POR))		V _{PORH}	—	110	—	mV	
Hysteresis width (LVD0, LVD1, and LVD2)		V _{LVH}	—	60	—	mV	LVD0 selected
			—	110	—		Vdet1_0 to Vdet1_2 selected
			—	70	—		Vdet1_3 to 9 selected
			—	90	—		LVD2 selected

Note: These characteristics apply when noise is not superimposed on the power supply. When a setting is made so that the voltage detection level overlaps with that of the voltage detection circuit (LVD1), it cannot be specified which of LVD1 and LVD2 is used for voltage detection.

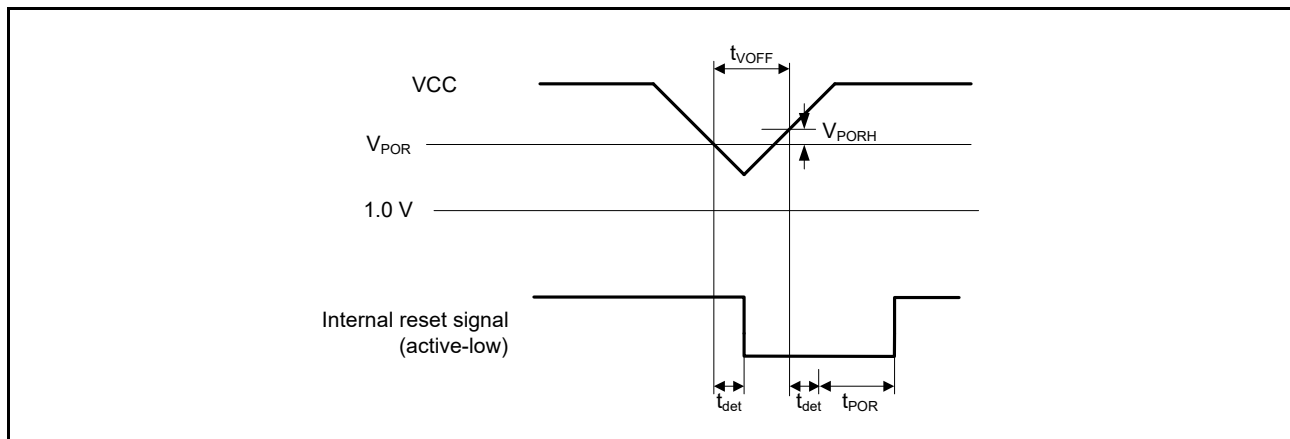
Note 1. When OFS1.(LVDAS, FASTSTUP) = 11b.

Note 2. When OFS1.(LVDAS, FASTSTUP) $\neq$ 11b.

Note 3. The minimum VCC down time indicates the time when VCC is below the minimum value of voltage detection levels V_{POR} , V_{det0} , V_{det1} , and V_{det2} for the POR/LVD.

Note 4. When OFS1.LVDAS = 1b.

Note 5. When OFS1.LVDAS = 0b.

**Figure 38.54 Voltage Detection Reset Timing**

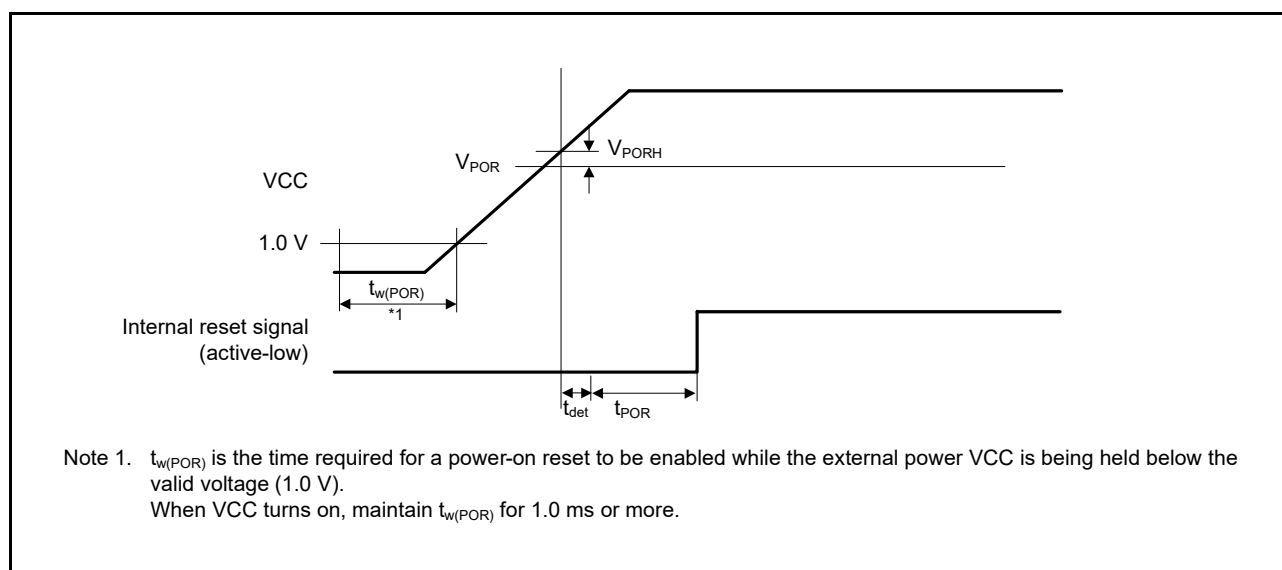
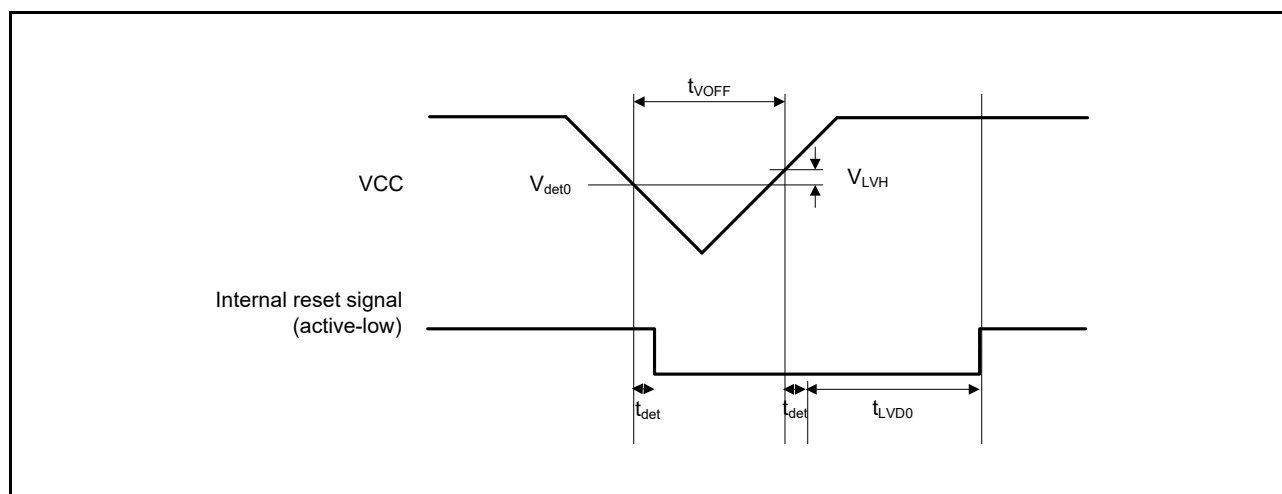
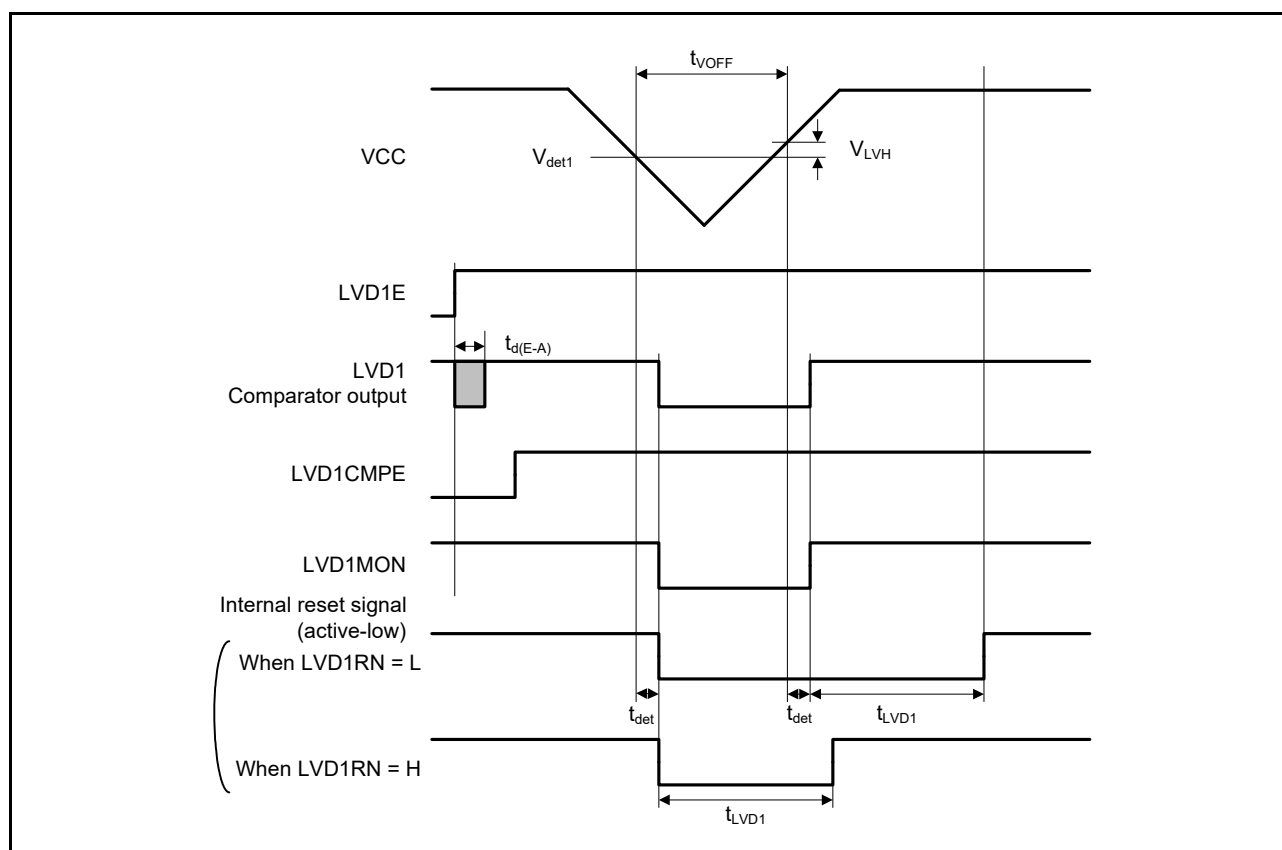
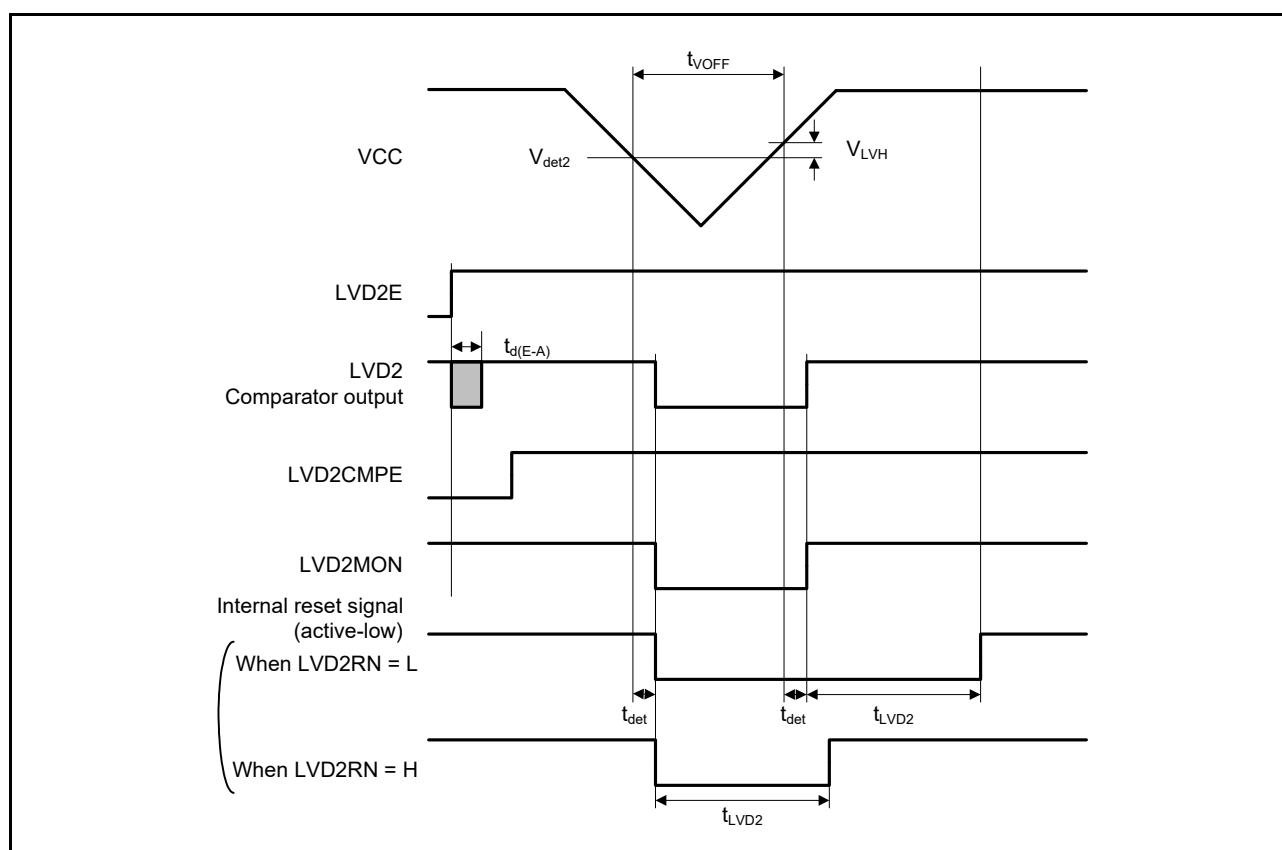


Figure 38.55 Power-On Reset Timing

Figure 38.56 Voltage Detection Circuit Timing (V_{det0})

Figure 38.57 Voltage Detection Circuit Timing (V_{det1})Figure 38.58 Voltage Detection Circuit Timing (V_{det2})

38.12 Oscillation Stop Detection Timing

Table 38.59 Oscillation Stop Detection Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Detection time	t_{dr}	—	—	1	ms	Figure 38.59

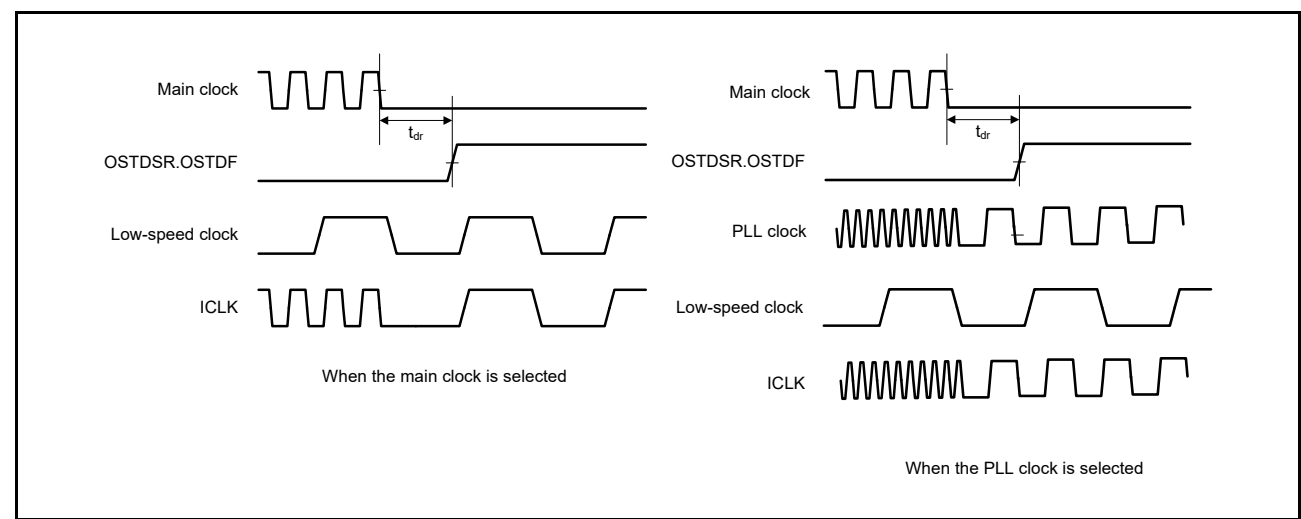


Figure 38.59 Oscillation Stop Detection Timing

38.13 ROM (Flash Memory for Code Storage) Characteristics

Table 38.60 ROM (Flash Memory for Code Storage) Characteristics (1)

Item	Symbol	Min.	Typ.	Max.	Unit	Conditions
Program/erase cycles*1	N_{PEC}	10K	—	—	Times	
Data retention*2, *3	After 10K times of N_{PEC}	t_{DRP}	20	—	Year	$T_a \leq +105^{\circ}\text{C}$
			10	—		$T_a \leq +125^{\circ}\text{C}$

Note 1. Definition of program/erase cycle: The program/erase cycle is the number of erasing for each block.

For instance, when 8-byte program is performed 256 times for different addresses in a 2-Kbyte block and then the block is erased, the program/erase cycle is counted as one. However, the same address cannot be programmed more than once before the next erase cycle (overwriting is prohibited).

Note 2. Characteristic when using the flash memory programmer and the self-programming library provided from Renesas Electronics.

Note 3. This result is obtained from reliability testing.

Table 38.61 ROM (Flash Memory for Code Storage) Characteristics (2) High-Speed Operating Mode

Item		Symbol	FCLK = 1 MHz			FCLK = 32 MHz			FCLK = 48 MHz			Unit
			Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	
Programming time	8-byte	t_{P8}	—	94.0	843.5	—	45.4	448.7	—	45.1	446.0	μs
Erasure time	2-Kbyte	t_{E2K}	—	8.3	282.0	—	5.4	220.4	—	5.4	220.1	ms
	128-Kbyte	t_{E128K}	—	204	4447	—	20.3	535.3	—	19.6	520.7	ms
Blank check time	8-byte	t_{BC8}	—	—	45.0	—	—	8.9	—	—	8.7	μs
	2-Kbyte	t_{BC2K}	—	—	1573	—	—	120	—	—	115	μs
Erase operation forcible stop time		t_{SED}	—	—	22.8	—	—	11.1	—	—	11.0	μs
Start-up area switching setting time		t_{SAS}	—	8.2	503.3	—	5.6	438.0	—	5.6	437.7	ms
Access window setting time		t_{AWS}	—	8.2	503.3	—	5.6	438.0	—	5.6	437.7	ms
ROM mode transition wait time		t_{MS}	15	—	—	15	—	—	15	—	—	μs

Note: Does not include the time until each operation of the flash memory is started after instructions are executed by software.

Note: The lower-limit frequency of FCLK is 1 MHz during programming or erasing of the flash memory. When using FCLK at below 4 MHz, the frequency can be set to 1 MHz, 2 MHz, or 3 MHz. A non-integer frequency such as 1.5 MHz cannot be set.

Note: The frequency accuracy of FCLK should be $\pm 3.5\%$.

Table 38.62 ROM (Flash Memory for Code Storage) Characteristics (3) Middle-Speed Operating Mode

Item		Symbol	FCLK = 1 MHz			FCLK = 24 MHz			Unit
			Min.	Typ.	Max.	Min.	Typ.	Max.	
Programming time	8-byte	t_{P8}	—	94.0	843.5	—	45.7	450.7	μ s
Erasure time	2-Kbyte	t_{E2K}	—	8.3	282.0	—	5.4	220.2	ms
	128-Kbyte	t_{E128K}	—	204	4447	—	19.6	521	ms
Blank check time	8-byte	t_{BC8}	—	—	45	—	—	9	μ s
	2-Kbyte	t_{BC2K}	—	—	1573	—	—	115	μ s
Erase operation forcible stop time		t_{SED}	—	—	22.8	—	—	11.2	μ s
Start-up area switching setting time		t_{SAS}	—	8.2	503.3	—	5.6	437.7	ms
Access window setting time		t_{AWS}	—	8.2	503.3	—	5.6	437.7	ms
ROM mode transition wait time		t_{MS}	15	—	—	15	—	—	μ s

Note: Does not include the time until each operation of the flash memory is started after instructions are executed by software.

Note: The lower-limit frequency of FCLK is 1 MHz during programming or erasing of the flash memory. When using FCLK at below 4 MHz, the frequency can be set to 1 MHz, 2 MHz, or 3 MHz. A non-integer frequency such as 1.5 MHz cannot be set.

Note: The frequency accuracy of FCLK should be $\pm 3.5\%$.

38.14 E2 DataFlash Characteristics (Flash Memory for Data Storage)

Table 38.63 E2 DataFlash Characteristics (1)

Item		Symbol	Min.	Typ.	Max.	Unit	Conditions
Program/erase cycles*1		N _{DPEC}	100K	1000K	—	Times	
Data retention	After 10K times of N _{DPEC}	t _{DDRP}	20*2, *3	—	—	Year	T _a ≤ +105°C
			10*2, *3	—	—	Year	T _a ≤ +125°C
	After 100K times of N _{DPEC}		5*2, *3	—	—	Year	T _a ≤ +125°C
	After 1000K times of N _{DPEC}		—	1*2, *3	—	Year	T _a = +25°C

Note 1. Definition of program/erase cycle: The program/erase cycle is the number of erasing for each block.

For instance, when 1-byte program is performed 256 times for different addresses in a 256-byte block and then the block is erased, the program/erase cycle is counted as one. However, the same address cannot be programmed more than once before the next erase cycle (overwriting is prohibited).

Note 2. Characteristic when using the flash memory programmer and the self-programming library provided from Renesas Electronics.

Note 3. These results are obtained from reliability testing.

Table 38.64 E2 DataFlash Characteristics (2) High-speed operating mode

Item		Symbol	FCLK = 1 MHz			FCLK = 32 MHz			FCLK = 48 MHz			Unit
			Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	
Programming time	1-byte	t _{DP1}	—	83.0	729.5	—	35.1	341.2	—	34.8	338.8	μs
Erasure time	256-byte	t _{DE256}	—	8.3	282.0	—	5.4	220.4	—	5.4	220.1	ms
	4-Kbyte	t _{DE4K}	—	55.0	1273.7	—	9.0	295.4	—	8.8	291.7	ms
Blank check time	1-byte	t _{DBC1}	—	—	44.6	—	—	8.9	—	—	8.2	μs
	256-byte	t _{DBC256}	—	—	1573	—	—	120	—	—	115	μs
Erase operation forcible stop time		t _{DSER}	—	—	22.8	—	—	11.1	—	—	11.0	μs
DataFlash STOP recovery time		t _{DSTOP}	250	—	—	250	—	—	250	—	—	ns

Note: Does not include the time until each operation of the flash memory is started after instructions are executed by software.

Note: The lower-limit frequency of FCLK is 1 MHz during programming or erasing of the flash memory. When using FCLK at below 4 MHz, the frequency can be set to 1 MHz, 2 MHz, or 3 MHz. A non-integer frequency such as 1.5 MHz cannot be set.

Note: The frequency accuracy of FCLK should be ±3.5%.

Table 38.65 E2 DataFlash Characteristics (3) Middle-speed operating mode

Item		Symbol	FCLK = 1 MHz			FCLK = 8 MHz			Unit
			Min.	Typ.	Max.	Min.	Typ.	Max.	
Programming time	1-byte	t _{DP1}	—	83.0	729.5	—	35.3	343.2	μs
Erasure time	256-byte	t _{DE256}	—	8.3	282.0	—	5.4	220.2	ms
	4-Kbyte	t _{DE4K}	—	55.0	1273.7	—	8.8	291.8	ms
Blank check time	1-byte	t _{DBC1}	—	—	44.6	—	—	9.0	μs
	256-byte	t _{DBC256}	—	—	1573	—	—	115	ms
Erase operation forcible stop time		t _{DSER}	—	—	22.8	—	—	11.2	μs
DataFlash STOP recovery time		t _{DSTOP}	250	—	—	250	—	—	ns

Note: Does not include the time until each operation of the flash memory is started after instructions are executed by software.

Note: The lower-limit frequency of FCLK is 1 MHz during programming or erasing of the flash memory. When using FCLK at below 4 MHz, the frequency can be set to 1 MHz, 2 MHz, or 3 MHz. A non-integer frequency such as 1.5 MHz cannot be set.

Note: The frequency accuracy of FCLK should be ±3.5%.

38.15 Usage Notes

38.15.1 Connecting VCL Capacitor and Bypass Capacitors

This MCU integrates an internal voltage-down circuit, which is used for lowering the power supply voltage in the internal MCU to adjust automatically to the optimum level. A 4.7- μ F capacitor needs to be connected between this internal voltage-down power supply (VCL pin) and VSS pin. Place an external capacitor close to the pins. Do not apply the power supply voltage to the VCL pin.

Insert a multilayer ceramic capacitor as a bypass capacitor between each pair of the power supply pins. Implement a bypass capacitor to the MCU power supply pins as close as possible. Use a recommended value of 0.1 μ F as the capacitance of the capacitors. For the capacitors related to crystal oscillation, see [section 9, Clock Generation Circuit](#). For the capacitors related to analog modules, also see [section 30, 12-Bit A/D Converter \(S12ADF\)](#).

For notes on designing the printed circuit board, see the descriptions of the application note “Hardware Design Guide” (R01AN1411EJ). The latest version can be downloaded from Renesas Electronics Website.

Appendix 1. Port States in Each Processing Mode

Table 1.1 Port States in Each Processing Mode

Port Name (Pin Name)	Reset	Software Standby Mode	
P00 (IRQ4)	Hi-Z	Keep-O*1	
P01 (IRQ2)	Hi-Z	Keep-O*1	
P02 (IRQ5)	Hi-Z	Keep-O*1	
P03, P04	Hi-Z	Keep-O	
P10 (IRQ0)	Hi-Z	Keep-O*1	
P11 (IRQ1)	Hi-Z	Keep-O*1	
P12 (IRQ7)	Hi-Z	Keep-O*1	
P13 (IRQ4)	Hi-Z	Keep-O*1	
P14 (IRQ6)	Hi-Z	Keep-O*1	
P15	Hi-Z	Keep-O	
P22 (IRQ2)	Hi-Z	Keep-O*1	
P23 (IRQ4)	Hi-Z	Keep-O*1	
P24 (IRQ3)	Hi-Z	Keep-O*1	
P25	Hi-Z	Keep-O	
P26 (IRQ1)	Hi-Z	Keep-O*1	
P30 to P33	Hi-Z	Keep-O	
P36 (IRQ3)	Hi-Z	Keep-O*1	
P37 (IRQ0)	Hi-Z	Keep-O*1	
P40 to P43, P45 to P47	Hi-Z	Keep-O	
P44 (DA0)	Hi-Z	DA0 output (DAOE0 = 1)	DA output retained
		Other than the above (DAOE0 = 0)	Keep-O
P70 (IRQ5)	Hi-Z	Keep-O*1	
P71 (IRQ6)	Hi-Z	Keep-O*1	
P72 (IRQ7)	Hi-Z	Keep-O*1	
P73 to P76	Hi-Z	Keep-O	
P90 (IRQ6)	Hi-Z	Keep-O*1	
P91, P95, P96	Hi-Z	Keep-O	
P92 (IRQ0)	Hi-Z	Keep-O*1	
P93 (IRQ1)	Hi-Z	Keep-O*1	
P94 (IRQ4)	Hi-Z	Keep-O*1	
P97 (IRQ2)	Hi-Z	Keep-O*1	
PB0, PB2, PB4, PB7	Hi-Z	Keep-O	
PB1 (IRQ2)	Hi-Z	Keep-O*1	
PB3 (IRQ3)	Hi-Z	Keep-O*1	
PB5 (IRQ7)	Hi-Z	Keep-O*1	
PB6 (IRQ5)	Hi-Z	Keep-O*1	
PD3, PD7	Hi-Z	Keep-O	
PD4 (IRQ2)	Hi-Z	Keep-O*1	
PD5 (IRQ3)	Hi-Z	Keep-O*1	
PD6 (IRQ5)	Hi-Z	Keep-O*1	
PE2 (NMI/IRQ0)	Hi-Z	Keep*1	
PG7/MD	Pullup*2	Keep-O	

Keep-O: Output pins retain their previous values, and input pins become high-impedance.

Keep: Retained the pin state in software standby mode (Pull-up and open-drain settings are also retained).

Hi-Z: High-impedance

Note 1. Input is possible, but only when the pin is specified as the software standby mode canceling source while it is used as an external interrupt pin.

Note 2. This pin functions as MD after release from the reset state, and the pull-up resistor connected to the MD pin is enabled.

Appendix 2. Package Dimensions

Information on the latest version of the package dimensions or mountings has been displayed in “Packages” on Renesas Electronics Corporation website.

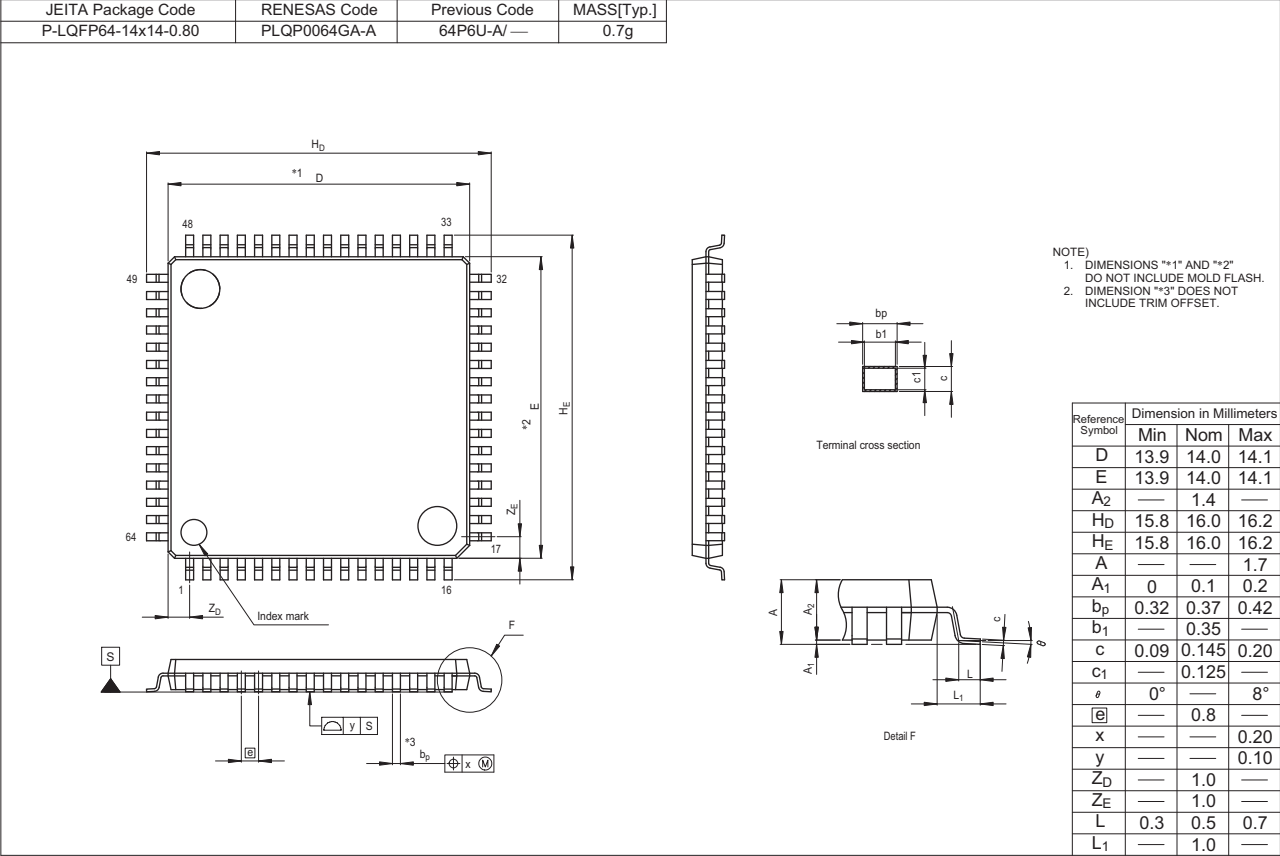


Figure A 64-Pin LQFP (PLQP0064GA-A)

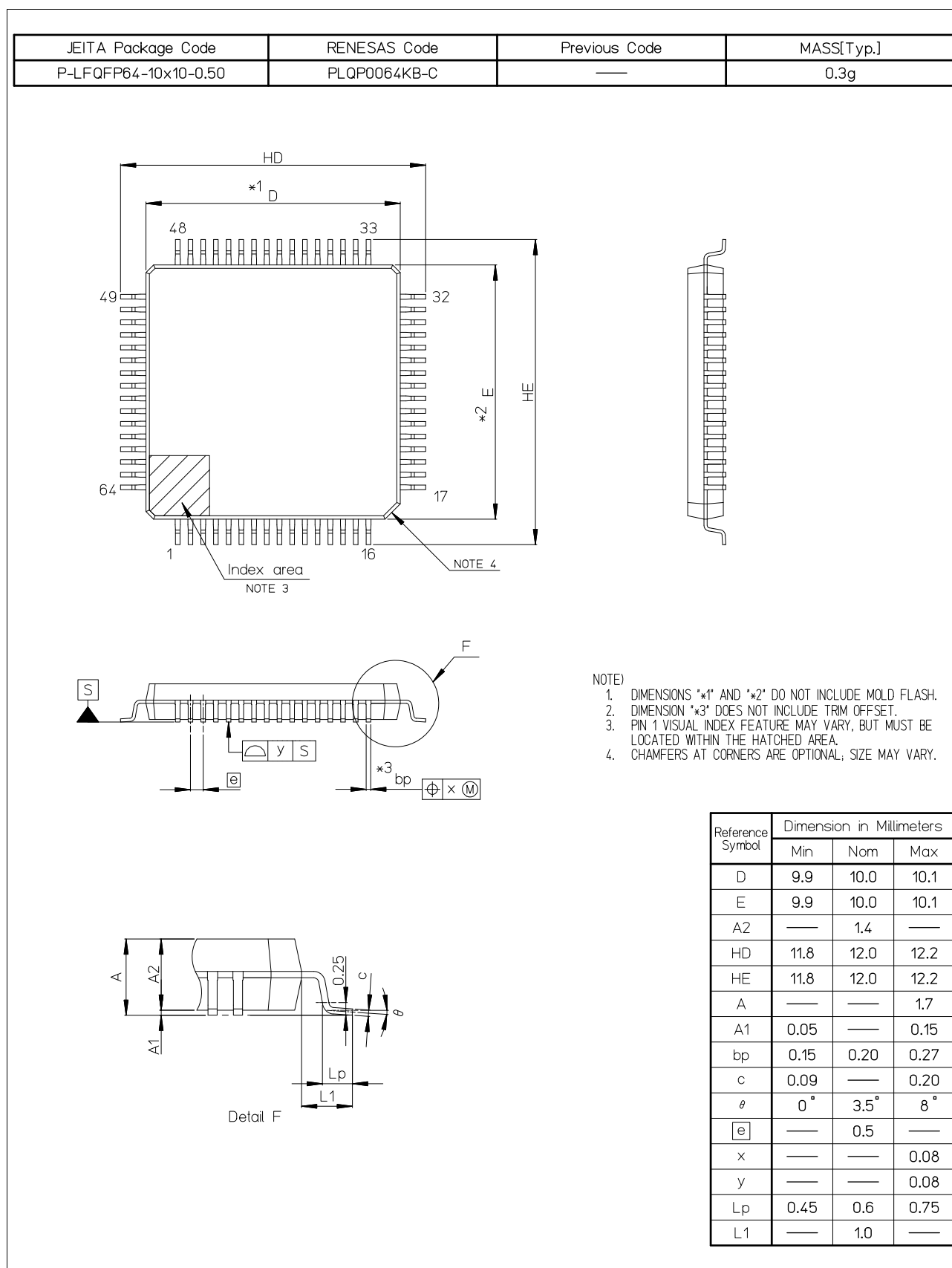


Figure B 64-Pin LQFP (PLQP0064KB-C)

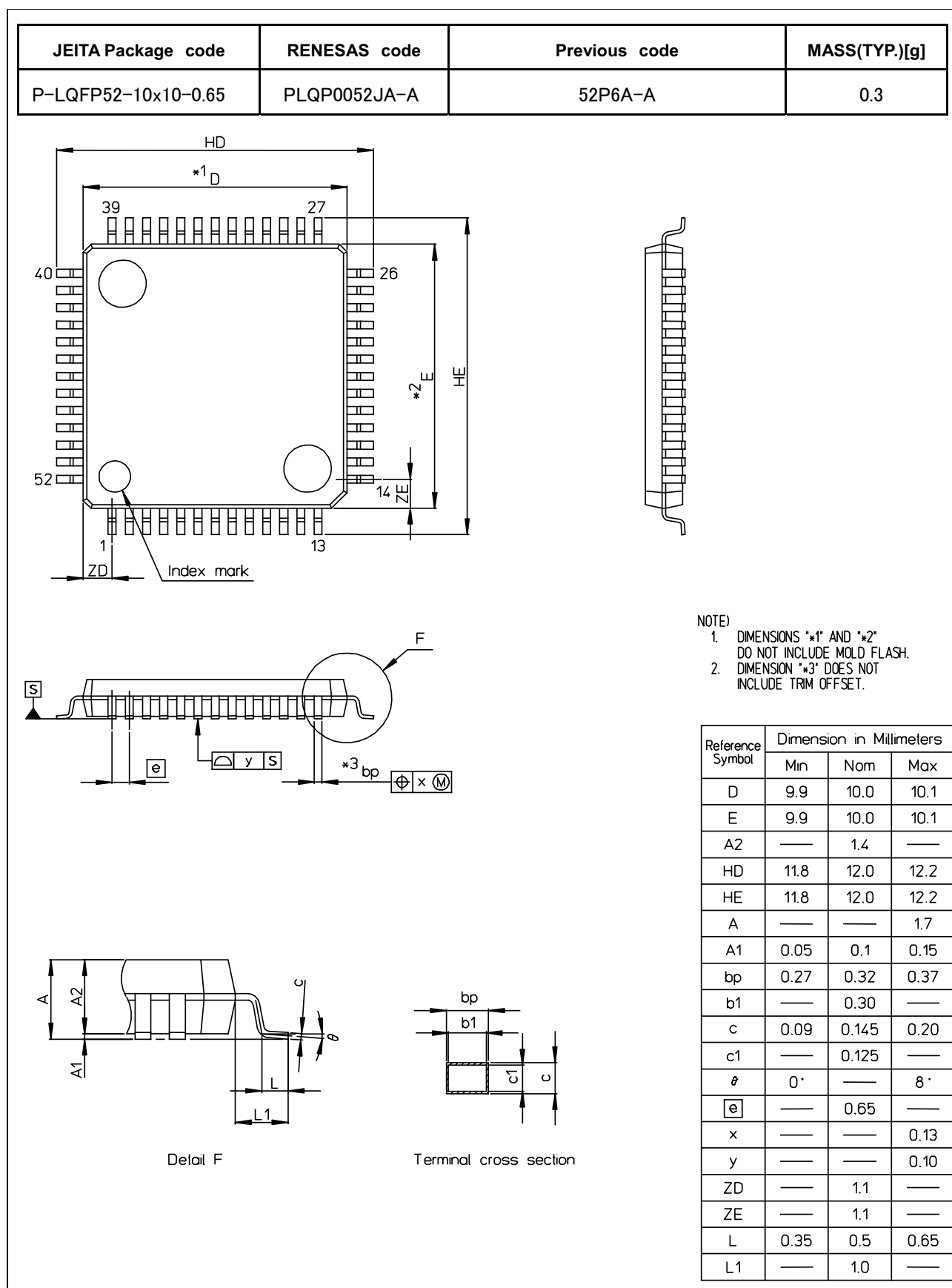


Figure C 52-Pin LQFP (PLQP0052JA-A)

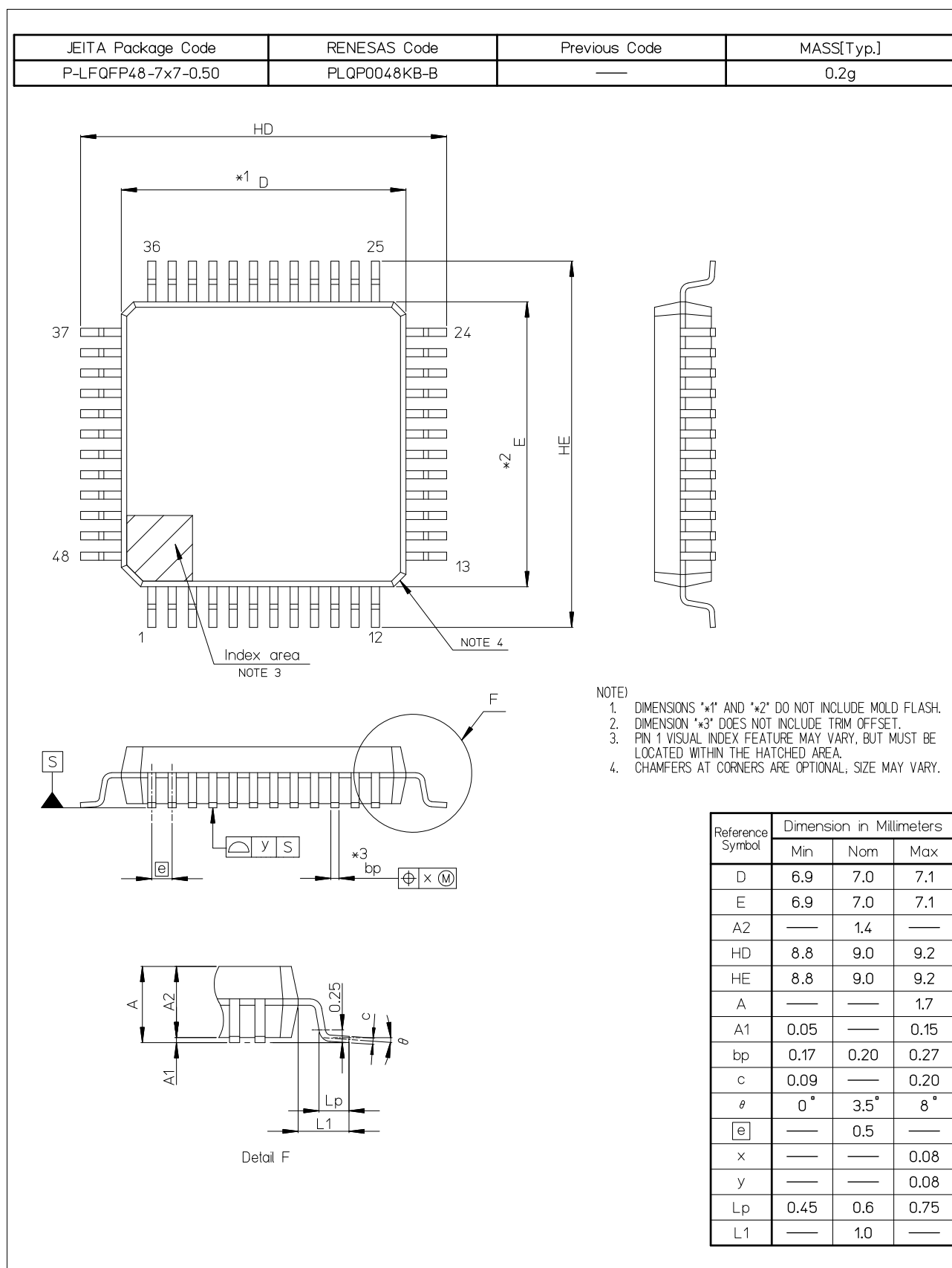


Figure D 48-Pin LFQFP (PLQP0048KB-B)

Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
A	—	—	0.80
A ₁	0.00	0.02	0.05
A ₃	0.203 REF.		
b	0.20	0.25	0.30
D	7.00 BSC		
E	7.00 BSC		
e	0.50 BSC		
L	0.30	0.40	0.50
K	0.20	—	—
D ₂	5.25	5.30	5.35
E ₂	5.25	5.30	5.35
aaa	0.15		
bbb	0.10		
ccc	0.10		
ddd	0.05		
eee	0.08		
fff	0.10		

Page 1421 of 1428

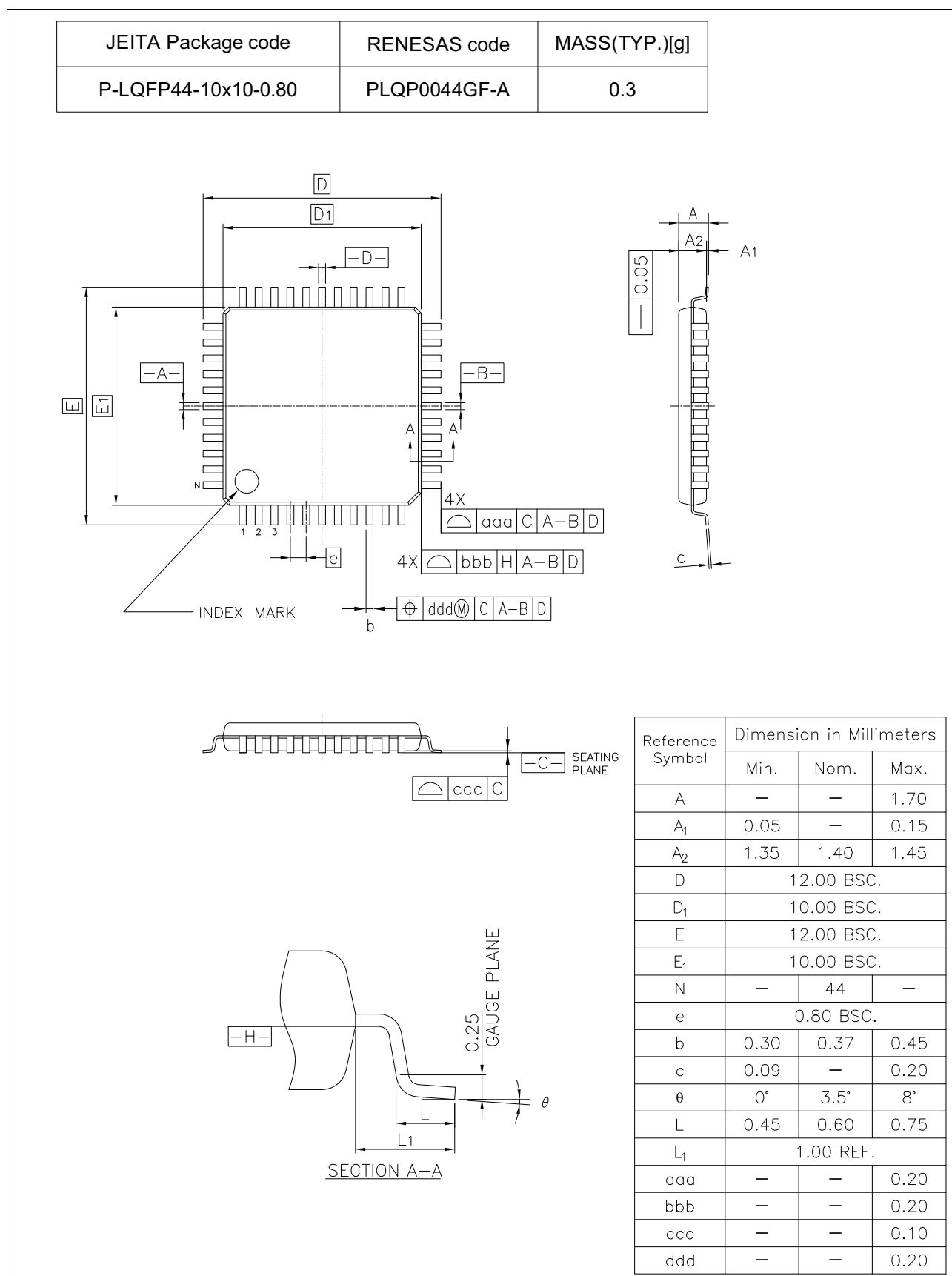


Figure F 44-Pin LQFP (PLQP0044GF-A)

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LQFP32-7x7-0.80	PLQP0032GB-A	P32GA-80-GBT-1	0.2

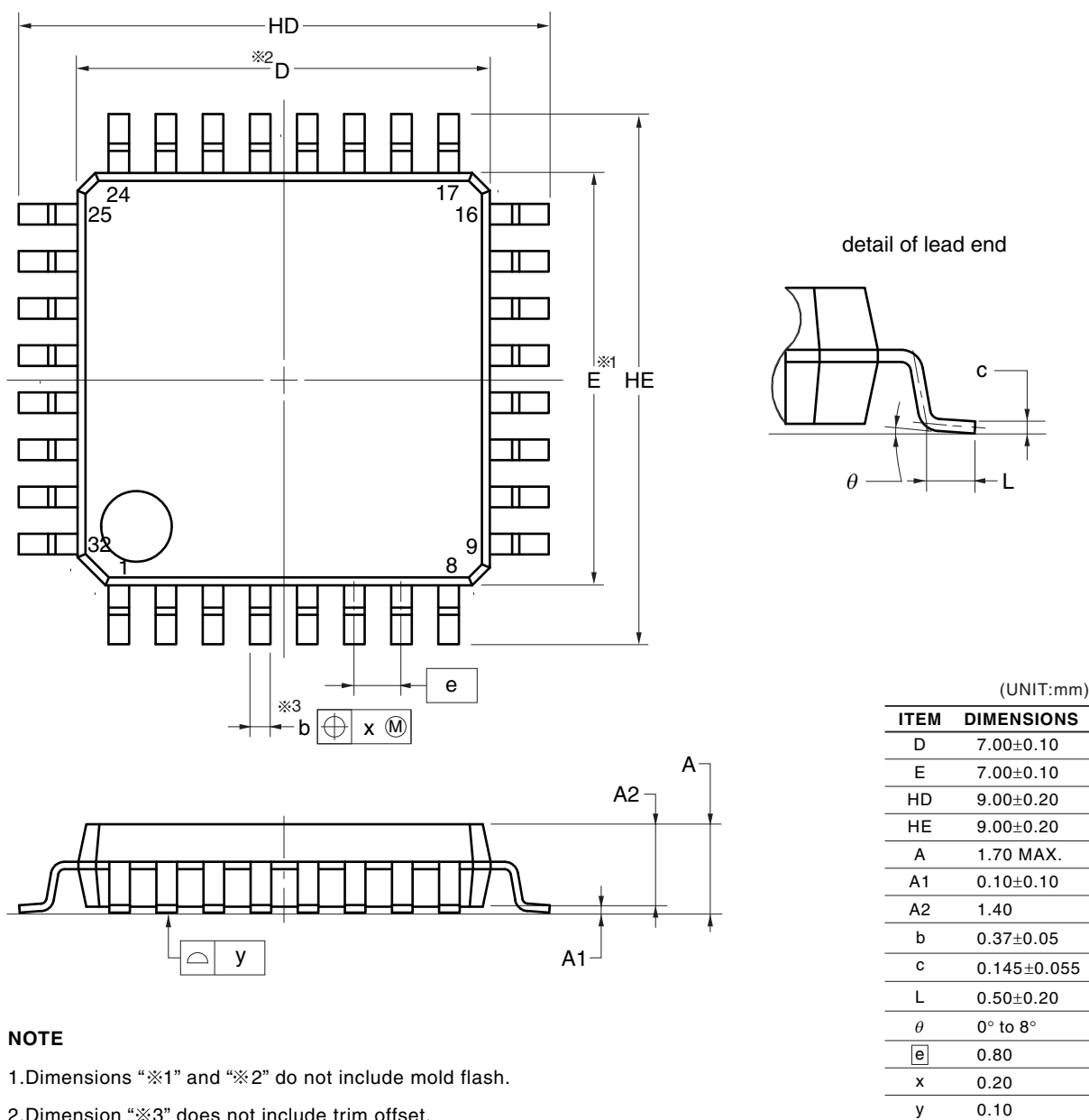
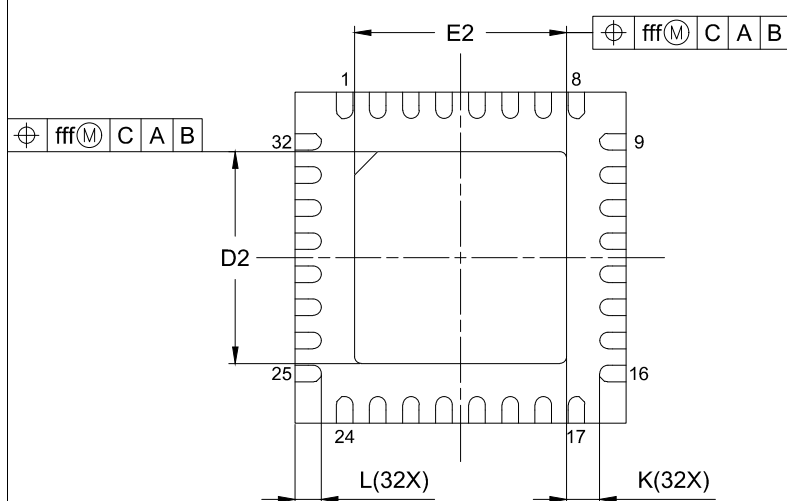
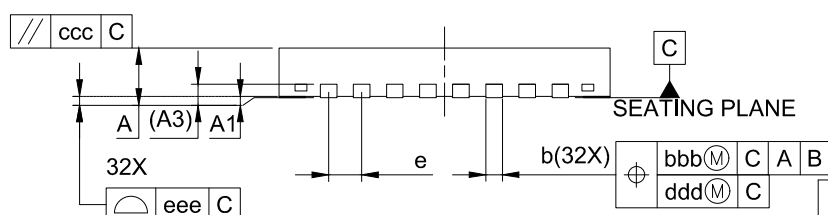
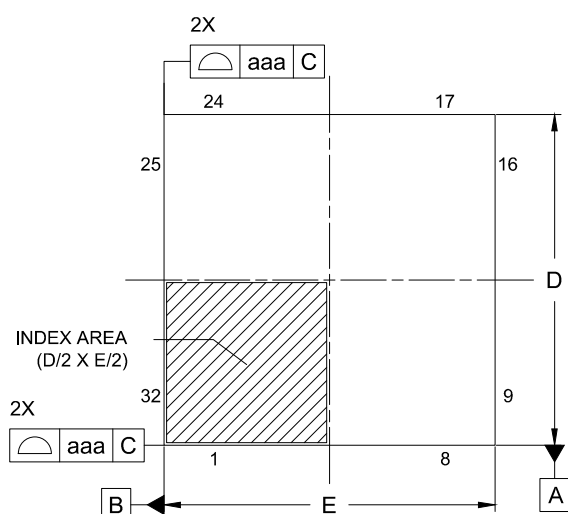


Figure G 32-Pin LQFP (PLQP0032GB-A)

JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN032-5x5-0.50	PWQN0032KE-A	0.06



Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
A	—	—	0.80
A ₁	0.00	0.02	0.05
A ₃	0.203 REF.		
b	0.18	0.25	0.30
D	5.00 BSC		
E	5.00 BSC		
e	0.50 BSC		
L	0.35	0.40	0.45
K	0.20	—	—
D ₂	3.15	3.20	3.25
E ₂	3.15	3.20	3.25
aaa	0.15		
bbb	0.10		
ccc	0.10		
ddd	0.05		
eee	0.08		
fff	0.10		

Figure H 32-Pin HWQFN (PWQN0032KE-A)

JEITA Package Code	RENESAS Code	MASS (Typ.) [g]
P-HWQFN24-4 × 4-0.50	PWQN0024KG-A	0.04

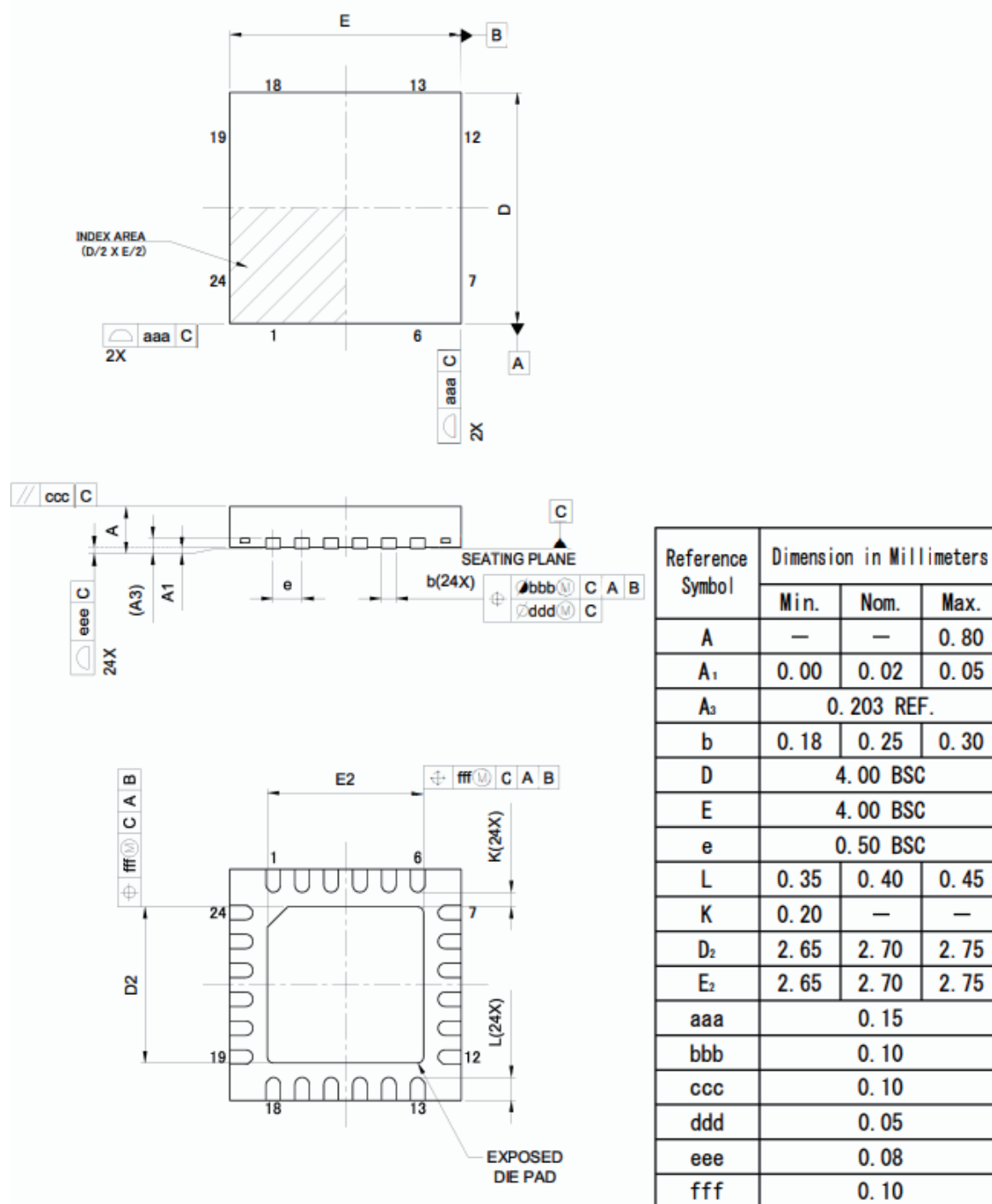


Figure I 24-Pin HWQFN (PWQN0024KG-A)

REVISION HISTORY	RX14T Group User's Manual: Hardware

Classifications

- Items with Technical Update document number: Changes according to the corresponding issued Technical Update
- Items without Technical Update document number: Minor changes that do not require Technical Update to be issued

Rev.	Date	Description		Classification
		Page	Summary	
1.00	Sep 30, 2025	—	First edition, issued	
1.01	Oct 22, 2025	—	52-pin package code, changed	
1.02	Dec 12, 2025	11. Low Power Consumption		
		208	11.4 Module Stop Function, changed	
		19. Multi-Function Timer Pulse Unit 3 (MTU3c)		
		532	19.6.7 Contention between Buffer Register Write Operation and Compare Match, changed	
		533	19.6.8 Contention between Buffer Register Write and TCNT Clear Operations, changed	
		20. Port Output Enable 3 (POE3E)		
		650	20.3.2 Input-Level Detection Operation (1) Edge Detection, changed	
		656	Figure 20.13 Example of Procedure for Setting the POE (Input-Level Detection Operation), changed	
		38. Electrical Characteristics		
		1397	Table 38.48 A/D Conversion Characteristics (1) Note 1, changed, Note 2, deleted	
		1398	Table 38.49 A/D Conversion Characteristics (2) Note 1, changed, Note 2, deleted	
		1399	Table 38.50 A/D Conversion Characteristics (3) Note 1, changed, Note 2, deleted	
		1403	Table 38.53 Programmable Gain Amplifier Characteristics, changed	

RX14T Group User's Manual: Hardware

Publication Date: Rev.1.00 Sep 30, 2025
 Rev.1.02 Dec 12, 2025

Published by: Renesas Electronics Corporation

RX14T Group