User's Manual

32

RH850/E1x-FCC2
Flash Memory

User's Manual: Hardware Interface

RENESAS MCU

Renesas Electronics
www.renesas.com

Rev.1.00 Jun, 2016

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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of 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. 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 should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

   “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.

   “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.

   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may 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 should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

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

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

(2012.4)

Notes for CMOS devices

(1) Voltage application waveform at input pin:

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

(2) Handling of unused input pins:

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

(3) Precaution against ESD:

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

(4) Status before initialization:

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

(5) Power ON/OFF sequence:

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

(6) Input of signal during power off state:

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

Table of Contents

Section 1 Features

The features of the flash memory are described below. See *Section 34 Flash Memory* in the user's manual for information on the capacity, block configuration, and addresses of the flash memory in a given product.

Flash Memory Programming/Erasure

A dedicated sequencer for the flash memory (flash sequencer) executes programming and erasure via the peripheral-bus. The flash sequencer also supports the programming/processing suspension/ resumption and BGO (background operation)*1.

Note 1. This can be used during overwriting of the data flash memory and reading of the code flash memory.

Security Functions

The flash memory incorporates hardware functions to prevent illicit tampering.

Protection Functions

The flash memory incorporates hardware functions to prevent erroneous writing.

Interrupts

The flash memory supports an interrupt to indicate completion of processing by the flash sequencer and an error interrupt to indicate erroneous operations.

DMA

The data flash memory can be programmed using the DMA.

Section 2 Module Configuration

Modules related to the flash memory are configured as shown in **Figure 2.1**. The flash sequencer is configured of Flash Control Unit (FCU) and Flash Application Command Interface (FACI). The FCU executes basic control of overwriting of the flash memory. The FCURAM is RAM for the storage of firmware to control execution by the FCU. The FACI receives FACI commands via the peripheral-bus and controls FCU operations accordingly.

In the transfer operations in response to a reset, the FACI transfers the data from flash memory to the option byte storage registers in the ID control section (FACI reset transfer). The ID control section compares the ID transferred from the flash memory with the value in the SELFID0 to SELFID3 registers. Data set in the option bytes of the flash memory can be read out from the option byte storage registers via the peripheral-bus.

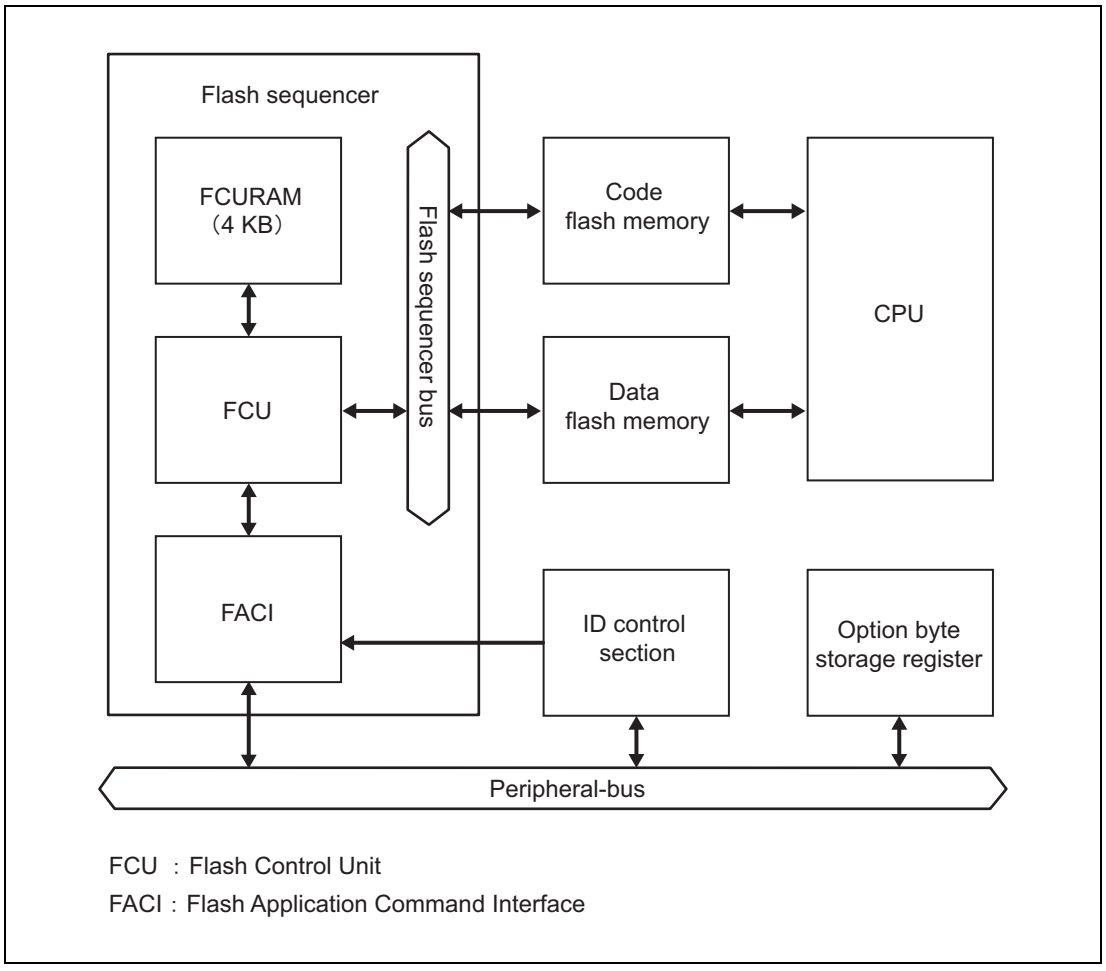


Figure 2.1 Configuration of Flash Memory Related Modules

Section 3 Address Map

Set the FCUSEL bit in the FCUFAREA register to 1 when reading areas for storage of the FCU firmware, that for the setting of configuration and that for the setting of OTP. **Table 3.1** gives information on all of these areas.

Table 3.1 Information on the Hardware Interface Area

Area	Address	Capacity	Peripheral Group
Area containing the various registers of the hardware	See **Section 4, Registers**	See **Section 4, Registers**	See **Appendix, List of Flash Memory Related Registers**
FACI command-issuing area	FFA2 0000_H	4 bytes	3
FCU firmware storage area	0001 7000_H to 0001 $7FFF_H$	4 Kbytes	—
FCU firmware storage area 2	0103 7000_H to 0103 $7FFF_H$	4 Kbytes	—
FCURAM area	FFA1 2000_H to FFA1 $2FFF_H$	4 Kbytes	3
Configuration setting area	FF30 0040_H to FF30 $008F_H$	80 bytes	0
OPT setting area	FF38 0040_H to FF38 $009F_H$	96 bytes	0

Refer to *Section 34 Flash Memory* in the user's manual for information on the addresses of the flash memory etc.

Section 4 Registers

This section gives information on the registers. For registers that are not specifically mentioned, only reset them to their initial states.

For the list of registers, see **Appendix, List of Flash Memory Related Registers**.

For information on the option bytes, see *Section 34 Flash Memory* in the User's Manual of each product.

4.1 FASTAT — Flash Access Status Register

FASTAT indicates access error status for code/data flash. If either of CFAE/CMDLK/DFAE bits in FASTAT is set to 1, flash sequencer enters the command lock state (see **8.2 Error Protection**). To cancel the command lock state, After setting the CFAE bit and DFAE bit in the FASTAT register to 0, and then issue a status clearing or forced stop command to FACI.

Access: This register can be read/written in 8-bit units.

Address: FFA1 0010_H

Initial value: 00_H

Bit	7	6	5	4	3	2	1	0
	CFAE	—	—	CMDLK	DFAE	—	—	ECRCT
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W*1	R	R	R	R/W*1	R	R	R

Note 1. Only 0 can be written to clear flag after 1 is read.

Table 4.1 FASTAT Register Contents (1/2)

Bit Position	Bit Name	Function
7	CFAE	Code Flash Access Error Indicates whether or not code flash access error has been generated. If this bit becomes 1, ILGLERR bit in FSTATR is set to 1 and flash sequencer enters the command lock state. 0: No code flash access error has occurred. 1: Code flash access error has occurred. [Setting Condition] FACI command with the following setting has been issued in code flash programming/erasure mode. (1) The settings for bits 23 to 0 in FSADDR are; 40_0000_H to FF_FFFF_H (Code Flash 4MB*1) 20_0000_H to FF_FFFF_H (Code Flash 2MB*1) (reversed are for the user area) (2) The settings for bits 23 to 0 in FSADDR are 00_8000_H to FF_FFFF_H (reserved area for the user boot area) [Clearing Condition] 0 is written after reading 1 from this bit.
6, 5	—	Reserved These bits are always read as 0. Write value should always be 0.
4	CMDLK	Command Lock Indicates whether flash sequencer is in the command lock state. 0: Flash sequencer is not in the command lock state. 1: Flash sequencer is in the command lock state. [Setting Condition] FACI detects error and enters the command lock state. [Clearing Condition] The flash sequencer starts the status clearing or forced stop command processing while the CFAE and DFAE bits in the FASTAT register are 0.

Table 4.1 FASTAT Register Contents (2/2)

Bit Position	Bit Name	Function
3	DFAE	Data Flash Access Error Indicates whether or not data flash access error has been generated. If this bit becomes 1, ILGLERR bit in FSTATR is set to 1 and flash sequencer enters the command lock state. 0: No data flash access error has occurred. 1: Data flash access error has occurred. [Setting Conditions] Commands have been issued in data flash programming/erasure mode under the following settings. • An FACI command has been issued when the setting of 18 to 0 in FSADDR register is; 1_0000_H to 7_FFFF_H (Data Flash 64KB*1) 0_8000_H to 7_FFFF_H (Date Flash 32KB*1) (reversed are for the data area) • The config programming command has been issued when the setting of bits 18 to 0 in the FSADDR register is 0_0000_H to 0_003F_H or 0_0100_H to 7_FFFF_H. • The OTP setting command has been issued when the settings for bits 18 to 0 in FSADDR are 0_0000_H to 0_003F_H or 0_00A0_H to 7_FFFF_H. [Clearing Condition] 0 is written after reading 1 from this bit.
2, 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	ECRCT	Error Correction Indicates that a 1-bit error has been corrected when the flash sequencer reads the flash memory (configuration setting, overwrite parameters, and OTP setting) or the FCURAM. 0: 1-bit error has not been corrected. 1: 1-bit error has been corrected. [Clearing conditions] • The flash sequencer starts the status clearing or forced stop command processing when bits CFGCRCT, TBLCRCT, and OTPCRCT in FSTATR are 1. • The flash sequencer starts forced stop command processing when the FRCRCT bit in FSTATR is 1.

Note 1. Capacities of the code flash memory and data flash memory vary from product to product. See the flash memory section in the user's manual of the applicable product.

4.2 FAEINT — Flash Access Error Interrupt Enable Register

FAEINT enables or disables output of flash access error (FLERR) interrupt.

In this product, a flash access error interrupt is treated as an error for the ECM.

Access: This register can be read/written in 8-bit units.

Address: FFA1 0014$_H$

Initial value: 99$_H$

Bit	7	6	5	4	3	2	1	0
	CFAEIE	—	—	CMDLKIE	DFAEIE	—	—	ECRCTIE
Value after reset	1	0	0	1	1	0	0	1
R/W	R/W	R	R	R/W	R/W	R	R	R/W

Table 4.2 FAEINT Register Contents

Bit Position	Bit Name	Function
7	CFAEIE	Code Flash Access Error Interrupt Enable Enables or disables the FLERR interrupt request when code flash access error occurs and CFAE bit in FASTAT becomes 1. 0: Does not generate the FLERR interrupt request when FASTAT.CFAE = 1. 1: Generates the FLERR interrupt request when FASTAT.CFAE = 1.
6, 5	—	Reserved These bits are always read as 0. Write value should always be 0.
4	CMDLKIE	Command Lock Interrupt Enable Enables or disables the FLERR interrupt request when flash sequencer enters the command lock state and CMDLK bit in FASTAT becomes 1. 0: Does not generate the FLERR interrupt request when FASTAT.CMDLK = 1. 1: Generates the FLERR interrupt request when FASTAT.CMDLK = 1.
3	DFAEIE	Data Flash Access Error Interrupt Enable Enables or disables the FLERR interrupt request when data flash access error occurs and DFAE bit in FASTAT becomes 1. 0: Does not generate the FLERR interrupt request when FASTAT.DFAE = 1. 1: Generates the FLERR interrupt request when FASTAT.DFAE = 1.
2, 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	ECRCTIE	Error Correction Interrupt Enable Enables or disables the FLERR interrupt request when a 1-bit error has been corrected and the ECRCT bit in FASTAT has been set to 1 on the flash memory read (configuration setting, overwrite parameters, and OTP setting) or the FCURAM read by the flash sequencer. 0: Does not generate the FLERR interrupt request when FASTAT.ECRCT = 1. 1: Generates the FLERR interrupt request when FASTAT.ECRCT = 1.

4.3 FSADDR — FACI Command Start Address Register

FSADDR specifies the start address of the target area for command processing when an FACI command (programming, DMA programming, block erasure, blank checking, config programming, lock bit programming, lock bit reading, or OTP setting) is issued.

FSADDR value is initialized when SUINIT bit in FSUINITR is set to 1. It is also initialized by a reset.

Access: This register can be read/written in 32-bit units.

Address: FFA1 0030_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FSADDR[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

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

Note 1. This bit can be written when the FRDY bit in the FSTATR register is 1. Writing to this bit while the FRDY bits is 0 is ignored.

Table 4.3 FSADDR Register Contents

Bit Position	Bit Name	Function
31 to 0	FSADDR[31:0]	Start Address of FACI Command Processing These bits specify the start address of the FACI command processing. Bits 31 to 24 are ignored in specifying the start address in the FACI command processing for the code flash memory. Bits 31 to 19 are ignored in the FACI command processing for the data flash memory. Lower address bits for smaller address than boundary below are also ignored. Bits 24 to 4 are used to generate the address parity.

Command	Address Boundary
Programming (code flash memory):	256 bytes
Programming (data flash memory):	
4-byte write:	4 bytes
DMA programming:	4 bytes
Block erasure (code flash memory):	8 Kbytes or 32 Kbytes
Block erasure (data flash memory):	64 bytes
Blank checking:	4 bytes
Config programming:	16 bytes
Lock bit programming:	8 Kbytes or 32 Kbytes
Lock bit reading:	8 Kbytes or 32 Kbytes
OTP setting:	16 bytes

4.4 FEADDR — FACI Command End Address Register

This register specifies the end address in the target area in blank checking command processing. When blank check addressing mode is set to incremental mode (i.e. FBCCNT.BCDIR = 0), address specified in FSADDR should be equal to or smaller than address in FEADDR. Conversely, address in FSADDR should be equal to or larger than address in FEADDR when blank check addressing mode is set to decremental mode (i.e. FBCCNT.BCDIR = 1). If setting of BCDIR, FSADDR, and FEADDR are inconsistent, FACI detects error and flash sequencer enters the command lock state. (See **Section 8.2, Error Protection**.)

FEADDR value is initialized when SUINIT bit in FSUINITR is set to 1. It is also initialized by a reset.

Access: This register can be read/written in 32-bit units.

Address: FFA1 0034_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FEADDR[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

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

Note 1. This bit can be written when the FRDY bit in the FSTATR register is 1. Writing to this bit while the FRDY bits is 0 is ignored.

Table 4.4 FEADDR Register Contents

Bit Position	Bit Name	Function
31 to 0	FEADDR[31:0]	End Address of FACI Command Target Area Specifies end address of target area in the blank checking command. Bits 31 to 19, 1 and 0 are ignored in the command processing.

4.5 FCURAME — FCURAM Enable Register

FCURAME enables or disables access to FCURAM area.

Access: This register can be read/written in 16-bit units.

Address: FFA1 0054_H

Initial value: 0000_H

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

Note 1. Written data is not stored in this bit. The value read is always 00_H.

Note 2. Writing to this bit is enabled only when $C4_H$ is written to the KEY bits.

Table 4.5 FCURAME Register Contents

Bit Position	Bit Name	Function
15 to 8	KEY	Key Code These bits enable or disable FRAMTRAN and FCRME bits modification.
7 to 2	—	Reserved These bits are always read as 0. Write value should always be 0.
1	FRAMTRAN	FCURAM Transfer Mode Specifies the FCURAM transfer mode. 0: Normal transfer mode Both read and write accesses to FCURAM are possible. 1: High-speed write mode High-speed writing to the FCURAM is possible. Though values can be written 2.5 times faster than when this bit is 0, reading is not possible
0	FCRME	FCURAM Enable Enables or disables access to the FCURAM. Before writing to the FCURAM, clear FENTRYR to 0000_H to stop the flash sequencer. 0: Disables access to FCURAM. 1: Enables access to FCURAM.

4.6 FSTATR — Flash Status Register

FSTATR indicates flash sequencer status.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: FFA1 0080_H

Initial value: $0000\ 8000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FRDY	ILGLERR	ERSERR	PRGERR	SUSRDY	DBFULL	ERSSPD	PRGSPD	—	FHVEERR	CFGDTCT	CFGCRCT	TBLDTCT	TBLCRCT	FRDTCT	FRCRCT
Value after reset	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 4.6 FSTATR Register Contents (1/5)

Bit Position	Bit Name	Function
31 to 19	—	Reserved These bits are always read as 0. Write value should always be 0.
18	EBFULL	FDMYECC Buffer Full Indicates the FDMYECC buffer status when issuing the programming command. The FACI incorporates a buffer for FDMYECC bit (ECC buffer). It is possible to use FDMYECC register as the ECC buffer by setting the ECCDISE bit in the FECCTMD register to 1. When FDMYECC is written to while EBFULL bit is 1, the FACI inserts a wait in the peripheral-bus. 0: The ECC buffer is empty. 1: The ECC buffer is full. [Setting condition] • The ECC buffer becomes full while issuing the programming command. [Clearing condition] • The ECC buffer becomes empty.
17	OTPDTCT	2-Bit Error Detection Monitor (OTP Setting) Indicates that a 2-bit error has been detected on reading the OTP value. The FACI reads the OTP value in the programming, block erasure, lock bit programming, lock bit reading, and OTP setting for the code flash memory. When this bit is 1, the flash sequencer is in the command lock state. 0: No 2-bit error has been detected. 1: A 2-bit error has been detected. [Clearing condition] • The flash sequencer starts status clearing or forced stop command processing.
16	OTPCRCT	1-Bit Error Correction Monitor (OTP Setting) Indicates that a 1-bit error has been corrected on reading the OTP value. The FACI reads the OTP value in the programming, block erasure, lock bit programming, lock bit reading, and OTP setting for the code flash memory. When this bit is 1, the flash sequencer continues the command processing and does not enter the command lock state. 0: 1-bit error has not been corrected. 1: 1-bit error has been corrected. [Clearing condition] • The flash sequencer starts status clearing or forced stop command processing.

Table 4.6 FSTATR Register Contents (2/5)

Bit Position	Bit Name	Function
15	FRDY	Flash Ready Indicates the processing state in flash sequencer. 0: Processing of the command (programming, DMA programming, block erasure, programming/erasure suspension, programming/erasure resumption, forced stop, blank checking, config programming, lock bit programming, lock bit reading, or OTP setting) is in progress. 1: None of the above is in progress. [Setting Conditions] • Flash sequencer completes processing. • Flash sequencer suspends processing by a programming/erasure suspensoin command. • Flash sequencer terminates processing by a forced stop command. [Clearing Conditions] • When the flash sequencer accepts the FACI command • For a programming, DMA programming, config programming, or OTP setting command, after the first write access to the FACI command-issuing area. • For other commands, after the last write access to the FACI command-issuing area.
14	ILGLERR	Illegal Command Error Indicates that flash sequencer has detected an illegal command or illegal flash memory access. When this bit is 1, flash sequencer is in the command lock state. 0: Flash sequencer has not detected any illegal command or illegal flash memory access. 1: Flash sequencer has detected an illegal command or illegal flash memory access [Setting conditions] (**Section 8.2, Error Protection**) • Flash sequencer has detected an illegal command. • Flash sequencer has detected an illegal flash memory access. • FENTRYR setting is illegal. [Clearing condition] Status clearing or forced stop command processing is started while the DFAE bit and CFAE bit in the FASTAT register are 0. When a status clearing or forced stop command is complete while the CFAE or DFAE bit in the FASTAT register is 1, the ILGLERR bit is set to 1. The ILGLERR bit is set to 0 temporarily during forced stop command processing. When the command is complete and the CFAE or DFAE is detected as 1, the ILGLERR bit is reset to 1.
13	ERSERR	Erasure Error Indicates result of code or data flash erasure by flash sequencer. When this bit is 1, flash sequencer is in the command lock state. 0: Erasure processing has been completed successfully 1: An error has occurred during erasure [Setting conditions] • An error has occurred during erasure. • An erasure command has been issued for the area protected by lock bit. [Clearing condition] Status clearing or forced stop command processing is started.

Table 4.6 FSTATR Register Contents (3/5)

Bit Position	Bit Name	Function
12	PRGERR	Programming Error Indicates the result of code or data flash programming by flash sequencer. When this bit is 1, flash sequencer is in the command lock state. 0: Programming has been completed successfully 1: An error has occurred during programming [Setting conditions] • An error has occurred during programming. • A programming or lock bit programming command has been issued for the area protected by lock bit. [Clearing condition] Status clearing or forced stop command processing is started.
11	SUSRDY	Suspend Ready Indicates whether flash sequencer is ready to accept the programming/erasure suspension command. 0: Flash sequencer cannot accept a programming/erasure suspension command. 1: Flash sequencer can accept a programming/erasure suspension command. [Setting condition] • After initiating programming or erasure, FACI entered a state where it is ready to accept a programming/erasure suspension command. [Clearing conditions] • FACI has accepted a programming/erasure suspension or forced stop command. (after the write access to the FACI command-issuing area is completed) • Flash sequencer has entered the command lock state during programming or erasure. • Programming/erasure processing is completed.
10	DBFULL	Data Buffer Full Indicates the data buffer status when issuing the programming command. The FACI incorporates a buffer for write data (data buffer). When issuing the flash memory write data to the FACI command-issuing area while the data buffer is full, the FACI inserts a wait in the peripheral-bus. 0: The data buffer is empty. 1: The data buffer is full. [Setting condition] • The data buffer becomes full while issuing the programming command. [Clearing condition] • The data buffer becomes empty.
9	ERSSPD	Erasure-Suspended Status Indicates that flash sequencer has entered the erasure command suspension process or erasure-suspended status. 0: Flash sequencer is in status other than the below mentioned. 1: Flash sequencer is in erasure suspension process or erasure-suspended status. [Setting condition] • Flash sequencer has initiated a programming/erase suspension command during erasure command processing. [Clearing conditions] • Flash sequencer has accepted a programming/erasure resumption command. (after the write access to the FACI command-issuing area is completed) • Forced stop command processing is started.

Table 4.6 FSTATR Register Contents (4/5)

Bit Position	Bit Name	Function
8	PRGSPD	Programming Suspension Status Indicates that flash sequencer has entered the programming command suspension process or programming suspension status. 0: Flash sequencer is in status other than the below mentioned. 1: Flash sequencer is in programming suspension process or programming suspension status. [Setting condition] • Flash sequencer has initiated a programming/erasure suspension command during programming command processing. [Clearing condition] • Flash sequencer has accepted a programming/erasure resumption command. (after the write access to the FACI command-issuing area is completed) • Forced stop command processing is started.
7	—	Reserved These bits are always read as 0. Write value should always be 0.
6	FHVEERR	Flash Write/Erase Protect Error Indicates the violation of the flash memory overwrite protection in the FHVE15 or FHVE3 register. When FHVEERR bit is 1, the flash sequencer is in the command lock state. 0: No error has occurred. 1: An error has occurred. [Clearing condition] • The flash sequencer starts forced stop command processing.
5	CFGDTCT	2-Bit Error Detection Monitor (Config Programming) Indicates that a 2-bit error has been detected on reading the config programming value. The FACI reads the config programming value in the config programming command. When this bit is 1, the flash sequencer is in the command lock state. 0: No 2-bit error has been detected. 1: A 2-bit error has been detected. [Clearing condition] • The flash sequencer starts status clearing or forced stop command processing.
4	CFGCRCT	1-Bit Error Correction Monitor (Config Programming) Indicates that a 1-bit error has been corrected on reading the config programming value. The FACI reads the config programming value in the config programming. When this bit is 1, the flash sequencer continues command processing and does not enter the command lock state. 0: 1-bit error has not been corrected. 1: 1-bit error has been corrected. [Clearing condition] • The flash sequencer starts status clearing or forced stop command processing.
3	TBLDTCT	2-Bit Error Detection Monitor (Overwrite Parameter Table) Indicates that a 2-bit error has been detected on reading the overwrite parameter table. The FACI reads the overwrite parameter table in programming, DMA programming, block erasure, blank checking, config programming, lock bit programming, and OTP setting for the flash memory. When this bit is 1, the flash sequencer is in the command lock state. 0: No 2-bit error has been detected. 1: A 2-bit error has been detected. [Clearing condition] • The flash sequencer starts status clearing or forced stop command processing.

Table 4.6 FSTATR Register Contents (5/5)

Bit Position	Bit Name	Function
2	TBLCRCT	1-Bit Error Correction Monitor (Overwrite Parameter Table) Indicates that a 1-bit error has been corrected on reading the overwrite parameter table. The FACI reads the overwrite parameter table in programming, DMA programming, block erasure, blank checking, config programming, lock bit programming, and OTP setting for the flash memory. When this bit is 1, the flash sequencer does not enter command lock state. 0: 1-bit error has not been corrected. 1: 1-bit error has been corrected. [Clearing condition] • The flash sequencer starts status clearing or forced stop command processing.
1	FRDTCT	2-Bit Error Detection Monitor (FCURAM) Indicates that 2-bit error has been detected in FCURAM read by the FCU. When the FRDTCT bit is 1, the flash sequencer enters the command lock state. 0: No 2-bit error has been detected. 1: A 2-bit error has been detected. [Clearing condition] • After the flash sequencer starts forced stop command processing.
0	FRCRCT	1-Bit Error Correction Monitor (FCURAM) Indicates that a 1-bit error has been corrected when the FCU reads the FCURAM. When this bit is 1, the flash sequencer does not enter the command lock state. 0: 1-bit error has not been corrected. 1: 1-bit error has been corrected. [Clearing condition] • After the flash sequencer starts forced stop command processing.

4.7 FENTRYR — Flash Programming/Erasure Mode Entry Register

FENTRYR specifies programming/erasure mode for code flash or data flash. To specify programming/erasure mode for code flash or data flash so that flash sequencer can accept FACI commands, set either of FENTRYD or FENTRYC bit to 1.

Note that if this register is set to a value other than 0000_H, 0001_H and 0080_H, ILGLERR bit in the FSTATR register will be set and flash sequencer will enter the command lock state.

FENTRY value is initialized when SUINIT bit in FSUINITR is set to 1. It is also initialized by a reset.

Access: This register can be read/written in 16-bit units.

Address: FFA1 0084_H

Initial value: 0000_H

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

Note 1. Written data is not stored in this bit. The value read is always 00_H.

Note 2. This bit can be written when the FRDY bit in the FSTATR register is 1. Writing to this bit while the FRDY bits is 0 is ignored.

Note 3. Writing to this bit is enabled only when AA_H is written to the KEY bits.

Note 4. In products incorporating G3MH, the value written to the FENTRYD bit may not be reflected if this bit is read right after it has been written. Dummy read this bit before reading it.

Table 4.7 FENTRYR Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 8	KEY	Key Code These bits enable or disable FENTRYD and FENTRYC bits modification.
7	FENTRYD	Data Flash Programming/Erasure Mode Entry This bit specifies programming/erasure mode for data flash. 0: Data flash is in read mode 1: Data flash is in programming/erasure mode [Setting condition] • 1 is written to FENTRYD while write enabling conditions are satisfied and FENTRYR is 0000_H. [Clearing conditions] • A value other than AA_H is written to KEY in FENTRYR while FRDY bit is 1. • 0 is written to FENTRYD while the write enabling conditions are satisfied. • FENTRYR is written to while FENTRYR is not 0000_H and the write enabling conditions are satisfied.
6 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.

Table 4.7 FENTRYR Register Contents (2/2)

Bit Position	Bit Name	Function
0	FENTRYC	Code Flash Programming/Erasure Mode Entry This bit specifies programming/erasure mode for data flash. 0: Code flash is in read mode 1: Code flash is in programming/erasure mode [Setting condition] • 1 is written to FENTRYC while write enabling conditions are satisfied and FENTRYR is 0000_H. [Clearing conditions] • A value other than AA_H is written to KEY in FENTRYR while FRDY bit is 1. • 0 is written to FENTRYC while the write enabling conditions are satisfied. • FENTRYR is written to while FENTRYR is not 0000_H and the write enabling conditions are satisfied.

4.8 FPROTR — Code Flash Protect Register

FPROTR enables or disables protection function through lock bits against programming and erasure.

FPROTR value is initialized when SUINIT bit in FSUINITR is set to 1. It is also initialized by a reset.

Access: This register can be read/written in 16-bit units.

Address: FFA1 0088_H

Initial value: 0000_H

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

Note 1. Written data is not stored in this bit. The value read is always 00_H.

Note 2. Writing to this bit is enabled only when 55_H is written to the KEY bits.

Table 4.8 FPROTR Register Contents

Bit Position	Bit Name	Function
15 to 8	KEY	Key Code These bits enable or disable FPROTCN bit modification.
7 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	FPROTCN	Lock Bit Protect Cancel Enables or disables protection through lock bits against programming and erasure. 0: Enables protection through lock bits 1: Disables protection through lock bits [Setting condition] • 1 is written to FPROTCN while write enabling conditions are satisfied and FENTRYR is not 0000_H. [Clearing conditions] • A value other than 55_H is written to KEY in FPROTR. • 0 is written to FPROTCN while the write enabling conditions are satisfied. • FENTRYR register value is 0000_H.

4.9 FSUINITR — Flash Sequencer Set-up Initialize Register

FSUINITR register is used for initialization of flash sequencer set-up.

Access: This register can be read/written in 16-bit units.

Address: FFA1 008C$_H$

Initial value: 0000$_H$

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

Note 1. Written data is not stored in this bit. The read value is always 00$_H$.

Note 2. This bit can be written when the FRDY bit in the FSTATR register is 1. Writing to this bit while the FRDY bits is 0 is ignored.

Note 3. Writing to this bit is enabled only when 2D$_H$ is written to the KEY bits.

Table 4.9 FSUINITR Register Contents

Bit Position	Bit Name	Function
15 to 8	KEY	Key Code These bits enable or disable SUINIT bit modification.
7 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	SUINIT	Set-up Initialization Initializes following flash sequencer set-up registers. • FEADDR • FSADDR • FPROTR • FENTRYR • FCPSR • FBCCNT 0: The above flash sequencer set-up registers keep its' value. 1: The above flash sequencer set-up registers are initialized.

4.10 FLKSTAT — Lock Bit Status Register

FLKSTAT indicates lock bit status which is read through the lock bit reading command execution.

Access: This register can be read in 8-bit units.

Address: FFA1 0090_H

Initial value: 00_H

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

Table 4.10 FLKSTAT Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	FLOCKST	Lock Bit Status Reflects the lock bit status read through the lock bit reading command execution. When FRDY bit becomes 1 after a lock bit reading command is issued, effective data for FLOCKST bit is stored. This bit value is retained until next lock bit reading command is completed. 0: Protected state 1: Non-protected state

4.11 FRFSTEADR — FCURAM First Error Address Register

FRFSTEADR indicates an address where the first ECC error has occurred on reading the FCURAM.

Access: This register can be read in 32-bit units.

Address: FFA1 0094_H

Initial value: $0000\ 0000_H$

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

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

Table 4.11 FRFSTEADR Register Contents

Bit Position	Bit Name	Function
31 to 12	—	Reserved These bits are always read as 0. Write value should always be 0.
11 to 0	FRFSTEADR	FCURAM First Error Address Indicate the first ECC error address on reading the FCURAM. An address offset from the top address of the FCURAM is stored.

4.12 FRTSTAT — FACI Reset Transfer Status Register

FRTSTAT indicates FACI reset transfer error status.

Access: This register can be read in 8-bit units.

Address: FFA1 0098_H

Initial value: $0X_H$

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

Table 4.12 FRTSTAT Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved These bits are always read as 0. Write value should always be 0.
1	RTEDTCT	FACI Reset Transfer Error Detect Indicates that 2-bit error has been detected in FACI reset transfer. Flash sequencer does not enter into the command lock state when this bit is 1. 0: No error has been detected. 1: An error has been detected. RTEDTCT bit is cleared when micro controller system is reset again and FACI reset transfer is finished without 2-bit error detection.
0	RTECRCT	FACI Reset Transfer Error Correct Indicates that 1-bit error has been corrected in FACI reset transfer. Flash sequencer does not enter into the command lock state when this bit is 1. 0: No error has been corrected. 1: An error has been corrected. RTECRCT bit is cleared when micro controller system is reset again and FACI reset transfer is finished without 1-bit error correction.

4.13 FRTEINT — FACI Reset Transfer Error Interrupt Enable Register

FRTEINT enables or disables output of FACI reset transfer error (FRTERR) interrupt.

In this product, an FACI reset transfer error interrupt is treated as an error for the ECM.

Access: This register can be read/written in 8-bit units.

Address: FFA1 009C$_H$

Initial value: 03$_H$

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

Table 4.13 FRTEINT Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved These bits are always read as 0. Write value should always be 0.
1	RTEDIE	FACI Reset Transfer Error Detection Interrupt Enable Enables or disables the FRTERR interrupt request when FACI reset transfer 2-bit error occurs and RTEDTCT bit in FRTSTAT register becomes 1. 0: Does not generate an FRTERR interrupt request when RTEDTCT = 1. 1: Generates a FRTERR interrupt request when RTEDTCT = 1.
0	RTECIE	FACI Reset Transfer Error Correction Interrupt Enable Enables or disables the FRTERR interrupt request when FACI reset transfer 1-bit error occurs and RTECRCT bit in FRTSTAT register becomes 1. 0: Does not generate an FRTERR interrupt request when RTECRCT = 1. 1: Generates an FRTERR interrupt request when RTEDTCT = 1.

4.14 FCMDR — FACI Command Register

FCMDR stores commands that FACI has accepted.

Access: This register can be read in 16-bit units.

Address: FFA1 00A0$_H$

Initial value: FFFF$_H$

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

Table 4.14 FCMDR Register Contents

Bit Position	Bit Name	Function
15 to 8	CMDR	Command These bits store the latest command accepted by FACI.
7 to 0	PCMDR	Previous Command These bits store previous command accepted by FACI.

Table 4.15 States of FCMDR after Acceptance of the Various Commands

Command	CMDR	PCMDR
Programming	E8$_H$	Previous command
DMA programming	EA$_H$	Previous command
Block erasure	D0$_H$	20$_H$
Programming/erasure suspension	B0$_H$	Previous command
Programming/erasure resumption	D0$_H$	Previous command
Status clearing	50$_H$	Previous command
Forced stop	B3$_H$	Previous command
Blank checking	D0$_H$	71$_H$
Config programming	40$_H$	Previous command
Lock bit Programming	D0$_H$	77$_H$
Lock bit reading	D0$_H$	71$_H$
OTP setting	45$_H$	Previous command

4.15 FPESTAT — Flash Programming/Erasure Status Register

This register is used to indicate the result of writing or erasure of the flash memory.

Access: This register can be read in 16-bit units.

Address: FFA1 00C0$_H$

Initial value: 0000$_H$

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

Table 4.16 FPESTAT Register Contents

Bit Position	Bit Name	Function
15 to 8	—	Reserved These bits are always read as 0. Write value should always be 0.
7 to 0	PEERRST	Programming/Erasure Error Status Indicates the source of error that occurs during programming/erasure for code flash or data flash. This bit value is only valid if ERSERR or PRGERR bit value in FSTATR register is 1, while FRDY bit in FSTATR register is 1. When ERSERR and PRGERR are 0, the PEERRST bit retains the value to indicate the source of error that previously occurred. 00$_H$: No error 01$_H$: A write error caused by a write attempt to an area protected by the lock bits 02$_H$: A write error caused by other source than the above 11$_H$: An erase error caused by an erase attempt to an area protected by the lock bits 12$_H$: An erase error caused by other source than the above Other than above: Reserved

4.16 FBCCNT — Data Flash Blank Check Control Register

FBCCNT specifies addressing mode in blank checking command processing. FBCCNT value is initialized when SUINIT bit in FSUINITR is set to 1. It is also initialized by a reset.

Access: This register can be read/written in 8-bit units.

Address: FFA1 00D0$_H$

Initial value: 00$_H$

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

Table 4.17 FBCCNT Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	BCDIR	Blank Check Direction Specifies addressing mode in blank checking operation. 0: Blank checking is executed from smaller address to larger address. (Incremental mode) 1: Blank checking is executed from larger address to smaller address. (Decremental mode)

4.17 FBCSTAT — Data Flash Blank Check Status Register

EEPBCSTAT stores check results by executing the blank checking command.

Access: This register can be read in 8-bit units.

Address: FFA1 00D4$_H$

Initial value: 00$_H$

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

Table 4.18 FBCSTAT Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	BCST	Blank Check Status Indicates the result of the blank checking command. 0: the target area is not written (no writing after erasing. blank) 1: The target area is filled with 0s and/or 1s.

4.18 FPSADDR — Programmed Area Start Address Register

FPSADDR indicates address of the first programmed data which is found in blank checking command execution.

Access: This register can be read in 32-bit units.

Address: FFA1 00D8$_H$

Initial value: 0000 0000$_H$

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

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

Table 4.19 FPSADDR Register Contents

Bit Position	Bit Name	Function
31 to 19	—	Reserved These bits are always read as 0. Write value should always be 0.
18 to 0	PSADR	Programmed Area Start Address Indicates address of the first programmed data which is found in blank checking command execution. These bits stores address offset from the top address in the data flash memory. This register value is only valid if BCST bit value in FBCSTAT register is 1, while FRDY bit in FSTATR register is 1. When BCST bit is 0, the PSDRA bit holds the address that previously checked.

4.19 FCPSR — Flash Sequencer Process Switch Register

FCPSR selects a function to make the FCU suspend erasure. FCPSR value is initialized when SUINIT bit in FSUINITR is set to 1. It is also initialized by a reset.

Access: This register can be read/written in 16-bit units.

Address: FFA1 $00E0_H$

Initial value: 0000_H

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

Table 4.20 FCPSR Register Contents

Bit Position	Bit Name	Function
15 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	ESUSPMD	Erasure-Suspended Mode Selects erasure-suspended mode to be entered when the programming/erasure suspension command is issued while flash sequencer is erasing flash memory. (see **6.3.12 Programming/Erasure Suspension Command**) ESUSPMD bit should be set before issuing the block erasure command. 0: Suspension-priority mode 1: Erasure-priority mode

4.20 FPCKAR — Flash Sequencer Processing Clock Notification Register

FPCKAR specifies the operating frequency of the flash sequencer while processing an FACI command. The highest operating frequency for this product is set as the initial value.

Access: This register can be read/written in 16-bit units.

Address: FFA1 00E4$_H$

Initial value: Maximum operating frequency of the FACI in the given product.

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

Note 1. Written data is not stored in this bit. The value read is always 00$_H$.

Note 2. This bit can be written when the FRDY bit in the FSTATR register is 1. Writing to this bit while the FRDY bits is 0 is ignored.

Note 3. Writing to this bit is enabled only when 1E$_H$ is written to the KEY bits.

Table 4.21 FPCKAR Register Contents

Bit Position	Bit Name	Function
15 to 8	KEY	Key Code These bits enable or disable the PCKA bit modification.
7 to 0	PCKA	Flash Sequencer Operating Clock Notify Specifies the operating frequency of the flash sequencer while processing an FACI command. Set the desired frequency in this bit before issuing an FACI command. Specifically, convert the frequency represented in MHz into a binary number and set it in this bit. Example: Frequency is 35.9 MHz (PCKA = 24$_H$) Round up the first decimal place of 35.9 MHz to a whole number (= 36) and convert it into a binary number. If the value set in this bit is smaller than the operating frequency of the flash sequencer, the flash memory overwrite characteristics cannot be guaranteed. If the value set in this bit is greater than the operating frequency of the flash sequencer, the flash memory overwrite characteristics can be guaranteed with the increased FACI command processing time such as overwrite time. (The minimum FACI command processing time is available when the operating frequency of the flash sequencer is the same as the PCKA value.) When SSCG is used, convert the center value of the operating frequency as described in the above example, and set the resulting value.

4.21 FECCEMON — Flash ECC Encoder Monitor Register

FECCEMON monitors the outputs from the address parity generator and ECC encoder.

Access: This register can be read in 16-bit units.

Address: FFA1 0100_H

Initial value: $FFFF_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FAPARM	FECCM8	FECCM7	FECCM6	FECCM5	FECCM4	FECCM3	FECCM2	FECCM1	FECCM0
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 4.22 FECCEMON Register Contents

Bit Position	Bit Name	Function
15 to 10	—	Reserved These bits are always read as 1.
9	FAPARM	Address Parity Monitor Indicates the output from the address parity generator. • In code flash programming/erasure mode This bit indicates the output from the address parity generator. • In data flash programming/erasure mode This bit is fixed to 1.
8 to 0	FECCM8 to FECCM0	ECC Monitor Indicates the ECC encoder output. • In code flash programming/erasure mode The FECCM8 to FECCM0 bits indicate the ECC encoder output for the code flash memory. • In data flash programming/erasure mode The FECCM8 and FECCM7 bits are fixed to 1. The FECCM6 to FECCM0 bits indicate the ECC encoder output for the data flash memory.

4.22 FECCTMD — Flash ECC Test Mode Register

FECCTMD sets the ECC test function for the flash memory.

Access: This register can be read/written in 16-bit units.

Address: FFA1 0104_H

Initial value: 0030_H

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

Note 1. Written data is not stored in this bit. The value read is always 00_H.

Note 2. Writing to this bit is enabled only when $A6_H$ is written to the KEY bits.

Table 4.23 FECCTMD Register Contents

Bit Position	Bit Name	Function
15 to 8	KEY	Key Code These bits enable or disable modification of the CECCVE, DECCVE, and ECCDISE bits.
7, 6	—	Reserved These bits are always read as 0. Write value should always be 0.
5	CECCVE	Code Flash Memory ECC Area Verify Enable Specifies the verify operation on overwriting the code flash memory. 0: Verifies the data area only. 1: Verifies the data area and the ECC area.
4	DECCVE	Data Flash Memory ECC Area Verify Enable Specifies the verify operation on overwriting the data flash memory. 0: Verifies the data area only. 1: Verifies the data area and the ECC area.
3 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	ECCDISE	ECC Encoder Disable Disables the address parity generator and the ECC encoder. If the address parity generator and the ECC encoder are disabled, the FDMYECC value is written to the flash memory. 0: The address parity generator and the ECC encoder are enabled. 1: The address parity generator and the ECC encoder are disabled.

4.23 FDMYECC — Flash Dummy ECC Register

FDMYECC specifies the address parity and ECC value to be written into the flash memory when the ECCDISE bit in the FECCTMD register is 1. The bit functions in code flash programming/erasure mode are different from those in data flash programming/erasure mode as shown below.

Access: This register can be read/written in 16-bit units.

Address: FFA1 0108$_H$

Initial value: FFFF$_H$

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

Table 4.24 FDMYECC Register Contents (in Code Flash Programming/Erasure Mode)

Bit Position	Bit Name	Function
15 to 10	—	Reserved These bits are always read as 1. Write value should always be 1.
9	DMYAPAR	Dummy Address Parity Specifies the address parity value when the ECCDISE bit is 1.
8 to 0	DMYECC[8:0]	Dummy ECC Specify the ECC value when the ECCDISE bit is 1.

Table 4.25 FDMYECC Register Contents (in Data Flash Programming/Erasure Mode)

Bit Position	Bit Name	Function
15 to 10	—	Reserved These bits are always read as 1. Write value should always be 1.
9	DMYAPAR	Reserved This bit is always read as 1. Write value should always be 1.
8, 7	DMYECC[8:7]	Reserved This bit is always read as 1. Write value should always be 1.
6 to 0	DMYECC[6:0]	Dummy ECC Specify the ECC value when the ECCDISE bit is 1.

4.24 FCUFAREA — FCU Firmware Area Select Register

FCUFAREA selects the FCU firmware storage area.

Access: This register can be read/written in 8-bit units.

Address: FFC5 9008_H

Initial value: *1

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

Note 1. This bit is set to 1 when booted in serial programming mode, and cleared to 0 when in user boot mode.

Table 4.26 FCUFAREA Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	FCUFSEL	Firmware Storage Area Select This bit switches the assigned area in 0001_7000_H to 0001_7FFF_H. And It is necessary to set this bit to 1 when the configuration area and OTP setting area are read. 0: The user area is assigned to 0001_7000_H to 0001_7FFF_H. 1: The firmware storage area is assigned to 0001_7000_H to 0001_7FFF_H. (The areas of Code Flash memory other than the described above are reserved ones.)

4.25 SELFID0 to SELFID3 — Self-Programming ID Input Registers

SELFID is for the input of an ID for use in authentication at the time of self-programming. The ID is authenticated by comparing the 128-bit ID that has been set in advance in a particular range of flash memory with the value in the SELFID0 to SELFID3 registers. The ID which is stored in a particular range of the flash memory can be set by the config programming command for the FACI.

Access: This register can be read/written in 32-bit units.

Address: FFA0 8000_H (SELFID0)
FFA0 8004_H (SELFID1)
FFA0 8008_H (SELFID2)
FFA0 $800C_H$ (SELFID3)

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SELFIDn[31:16]*1															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Note 1. n = 0 to 3

Table 4.27 SELFID0 to SELFID3 Register Contents

Bit Position	Bit Name	Function
31 to 0	SELFIDn[31:0]	ID for Use in Authentication of Self-Programming The ID for use in authentication at the time of self-programming is input to these bits. Authentication of the ID is executed by comparing the 128-bit ID that has been set in advance in a particular range of flash memory with the value in the SELFIDn[31:0] bits. The 128-bit ID is arranged in the respective sets of SELFIDn[31:0] bits in the way listed below. ID[31:0]: SELFID0[31:0] ID[63:32]: SELFID1[31:0] ID[95:64]: SELFID2[31:0] ID[127:96]: SELFID3[31:0]

4.26 SELFIDST — Self-Programming ID Authentication Status Register

SELFIDST indicates the result of authentication of an ID at the time of self-programming. That is, the SELFIDST register indicates the result of comparing the 128-bit ID that has been set in advance in a particular range of flash memory with the value in the SELFID0 to SELFID3 registers. The ID which is stored in a particular range of the flash memory can be set by the config programming command for the FACI.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: FFA0 8010_H

Initial value: 0000 $000X_H$

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

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

Table 4.28 SELFIDST Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. Write value should always be 0.
0	IDST	ID Authentication Status This bit indicates the result of comparing the 128-bit ID that has been set in advance in a particular range of flash memory with the value in the SELFID0 to SELFID3 registers. 0: The IDs match (protection unlocked). 1: The IDs do not match (protection locked).

4.27 Other Registers Related to Protecting the Flash Memory against Programming and Erasure

Other registers related to protecting the flash memory against programming and erasure are listed in **Table 4.29**.

Table 4.29 Other Registers Related to Protecting the Flash Memory against Programming and Erasure

Register Name	Symbol	R/W	Initial Value	Address	Access Size
FHVE15 control register	FHVE15	R/W	0000 0000_H	FFF8 $A430_H$	32
FHVE3 control register	FHVE3	R/W	0000 0000_H	FFF8 2410_H	32

4.27.1 FHVE15 — FHVE15 Control Register

FHVE15 is a readable and writable register for protecting the flash memory against programming and erasure.

To proceed with programming and erasure of the flash memory, set both the FHVE3 and FHVE15 registers to the value (0000 0001_H) that allows this.

If these registers are set to 0000 0000_H that does not allow programming and erasure of the flash memory, the following commands cannot be executed. Issuing any of the following commands leads to setting of the FHVEERR bit in the FSTATR register to 1.

- Programming
- Block erasure
- P/E resumption
- Configuration setting
- OTP setting
- DMA programming
- P/E suspension
- Blank checking
- Lock bit programming

Access: This register can be read/written in 32-bit units.

Address: FFF8 $A430_H$

Initial value: 0000 0000_H

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FHVE15CNT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 4.30 FHVE15 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	FHVE15CNT	0: Programming/erasure disabled 1: Programming/erasure enabled

4.27.2 FHVE3 — FHVE3 Control Register

FHVE3 is a readable and writable register for protecting the flash memory against programming and erasure.

To proceed with programming and erasure of the flash memory, set both the FHVE3 and FHVE15 registers to the value (0000 0001_H) that allows this.

If these registers are set to 0000 0000_H that does not allow programming and erasure of the flash memory, the following commands cannot be executed. Issuing any of the following commands leads to setting of the FHVEERR bit in the FSTATR register to 1.

- Programming
- Block erasure
- P/E resumption
- Configuration setting
- OTP setting
- DMA programming
- P/E suspension
- Blank checking
- Lock bit programming

Access: This register can be read/written in 32-bit units.

Address: FFF8 2410_H

Initial value: 0000 0000_H

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FHVE3 CNT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 4.31 FHVE3 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	FHVE3CNT	0: Programming/erasure disabled 1: Programming/erasure enabled

Section 5 Flash Sequencer Operating Modes

The flash sequencer has three operating modes as shown in **Figure 5.1**. The mode is shifted by the write to the FENTRYR register.

When the FENTRYR register is 0000_H, the flash sequencer is in read mode. In this mode, it does not accept the FACI command. The code flash memory and the data flash memory are both readable.

When the FENTRYR register is 0001_H, the flash sequencer is in code flash programming/erasure mode where the code flash memory can be programmed/erased by the FACI command. In this mode, the data flash memory is not readable. In addition, the code flash memory is also not readable. Under the condition where the BGO operation is enabled, the code flash memory is readable. As for the condition to enable the BGO operation, refer to the user's manual for this product.

When the FENTRYR register is 0080_H, the flash sequencer is in data flash programming/erasure mode where the data flash memory can be programmed/erased by the FACI command. In this mode, the data flash memory is not readable. However, the code flash memory is readable.

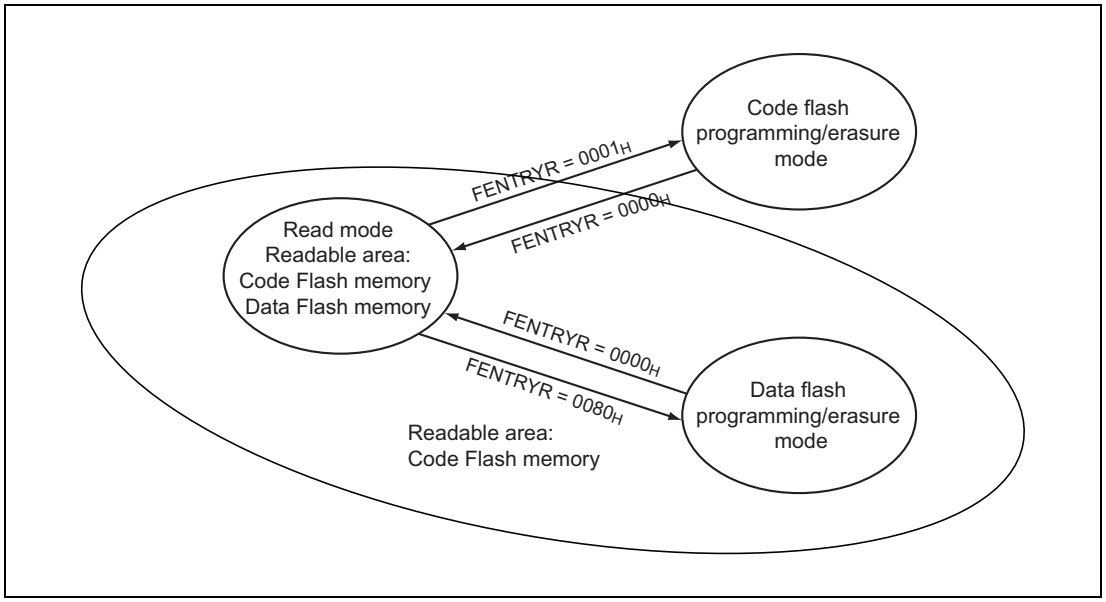


Figure 5.1 Flash Sequencer Modes

Section 6 FACI Command

6.1 List of FACI Commands

Table 6.1 List of FACI Commands

FACI Command	Function
Programming	User area and data area can be programmed. Programming unit is 256 bytes for user area and 4 bytes for data area.
DMA programming	Data area can be programmed using the DMA controller. Programming unit is 4 to 64 Kbytes (4 bytes step).
Block erasure	User area, Lock bit, and data area can be erased. Erasure unit is one block.
Programming/erasure suspension	Programming or erasure command operation can be suspended.
Programming/erasure resumption	Suspended programming or erasure command operation can be resumed.
Status clearing	The OTPDTCT, OTPCRCT, ILGLERR, ERSERR, PRGERR, CFGDTCT, CFGCRCT, TBLDTCT, TBLCRCT bits of the FSTATR register are initialized and flash sequencer is released from “Command Lock” state.
Forced stop	Any command operation can be stopped forcibly and the FSTATR register is initialized.
Blank checking	Data area can be checked. Blank checking unit is 4 to 64K bytes (4 bytes step).
Config programming	ID, security function, safety function, and option byte are set. Programming unit is 16 bytes.
Lock bit programming	Lock bit for one block of user area is programmed. Programming unit is 1 bit (lock bit for one block)
Lock bit reading	Lock bit for one block of user area is read out to FLKSTAT register. Reading unit is 1 bit (lock bit for one block)
OTP setting	OTPs are set for user area and user boot area. Setting unit is 16 bytes (OTP is set for 128 blocks)

The FACI commands are issued by the write access to the FACI command-issuing area (see **Table 3.1**). When the write access as shown in **Table 6.2** is issued in the specified state, the flash sequencer executes the processing corresponding to each command (see **Section 6.2, Relationship between Flash Sequencer Status and FACI Commands**).

Table 6.2 Flash Sequencer Command Format

FACI Command	Number of write access	Write Data to FACI command-issuing area			
		1st access	2nd access*1	3rd to (N+2)th access	(N+3)th access
Programming (user area) 256-byte programming N = 128	131	$E8_H$	80_H (=N)	WD_1 to WD_{128}	$D0_H$
Programming (data area) 4-byte programming	5	$E8_H$	02_H	WD_1 to WD_2	$D0_H$
DMA programming N = 2 to 32768 (even numbers only)	N+2	EA_H	N	WD_1 to WD_N	—
Block erasure	2	20_H	$D0_H$	—	—
Programming/erasure suspension	1	$B0_H$	—	—	—
Programming/erasure resumption	1	$D0_H$	—	—	—
Status clearing	1	50_H	—	—	—
Forced stop	1	$B3_H$	—	—	—
Blank checking	2	71_H	$D0_H$	—	—
Config programming N = 8	11	40_H	08_H (=N)	WD_1 to WD_8	$D0_H$
Lock bit programming	2	77_H	$D0_H$	—	—
Lock bit reading	2	71_H	$D0_H$	—	—
OTP setting N = 8	11	45_H	08_H (=N)	WD_1 to WD_8	$D0_H$

Note: WD_N (N=1,2,…): Nth 16-bit data to be programmed.

Note 1. 8-bit data is written by a command other than DMA programming. By the DMA programming command, 16-bit data is written.

Once the flash sequencer starts processing any command other than status clearing, it sets the FRDY bit in the FSTATR register to 0, and when processing of the command is complete, it sets the FRDY bit to 1 (see **Section 4.6, FSTATR — Flash Status Register**). When the value of the FRDY bit changes from 0 to 1, a flash ready (FRDY) interrupt is generated.

6.2 Relationship between Flash Sequencer Status and FACI Commands

The FACI commands are accepted according to the mode/state of the flash sequencer. The FACI command should be issued after the shift of the flash sequencer to the code flash programming/erasure mode or data flash programming/erasure mode and checking that the flash sequencer has shifted to the mode. To check the state of flash sequencer, use the FSTATR and FASTAT registers. In addition, error occurrence can be checked by the CMDLK bit in the FASTAT register. It is the logical OR of the OTPDTCT, ILGLERR, ERSERR, PRGERR, FHVEERR, CFGDTCT, TBLDTCT, and FRDTCT bits of the FSTATR register.

Table 6.3 summarizes available flash sequencer commands in each operating mode.

Table 6.3 Flash Sequencer Operation Mode and Available Commands

Operating Mode	FENTRYR	Available Command
Read Mode	0000_H	No command is available.
Code flash programming/erasure Mode	0001_H	Programming Block erasure Programming/erasure suspension Programming/erasure resumption Status clearing Forced stop Lock bit programming Lock bit reading
Data flash programming/erasure Mode	0080_H	Programming DMA programming Block erasure Programming/erasure suspension Programming/erasure resumption Status clearing Forced stop Blank checking Config programming OTP setting

Table 6.4 shows the flash sequencer state and the acceptable FACI commands. The table assumes appropriate flash sequencer operation mode is set before issuing the command.

Table 6.4 Flash Sequencer State and Acceptable FACI Commands

	Programming or Erasure Command Processing	Config Programming or OTP Setting Command Processing	Programming or Eerasure Command Suspension Processing	Blank Checking or Lock Bit Reading Command Processing	DMA Programming Command Processing	While Suspend Programming Command	While Suspend Erasure Command	While Suspend Erasure Command, and Programming Command Processing	Command Lock State (Frdy = 1)	Command Lock State (Frdy = 0)	Lock Bit Programming Command Processing	Forced Stop Command Processing	Other
FRDY bit	0	0	0	0	0	1	1	0	1	0	0	0	1
SUSRDY bit	1	0	0	0	0	0	0	0	0	0	0	0	0
ERSSPD bit	0	0	0/1	0/1	0	0	1	1	0/1	0/1	0	0	0
PRGSPD bit	0	0	0/1	0/1	0	1	0	0	0/1	0/1	0	0	0
CMDLK bit	0	0	0	0	0	0	0	0	1	1	0	0	0
Programming	X	X	X	X	X	X	O *3	X	X	X	X	X	O
DMA programming	X	X	X	X	X	X	O *1, *3	X	X	X	X	X	O *1
Block Erasure	X	X	X	X	X	X	X	X	X	X	X	X	O
Programming/erasure suspension	O	X	X	X	X	X	X	X	--	X	X	X	--
Programming/erasure resumption	X	X	X	X	X	O	O	X	X	X	X	X	X
Status clearing	X	X	X	X	X	O	O	X	O	X	X	X	O
Forced stop	O	O	O	O	O	O	O	O	O	O	O	O	O
Blank checking	X	X	X	X	X	O *1	O *1	X	X	X	X	X	O *1
Config programming	X	X	X	X	X	X	X	X	X	X	X	X	O *1
Lock bit programming	X	X	X	X	X	X	X	X	X	X	X	X	O *2
Lock bit reading	X	X	X	X	X	O *2	O *2, *4	X	X	X	X	X	O *2
OTP setting	X	X	X	X	X	X	X	X	X	X	X	X	O *1

O: Acceptable
X: Not acceptable (in Command Lock state)
--: Ignored

Note 1. Acceptable only in data flash programming/erasure mode.
Note 2. Acceptable only in code flash programming/erasure mode.

Note 3. Acceptable when programming area is other than erasure suspending sector.
Note 4. Undefined value is read out when a lock bit reading command is issued to erasure suspending sector.

6.3 Use FACI Command

This section describes the overview of FACI command usage.

6.3.1 Overview of the Command Usage in Code Flash Programming/Erasure Mode

The overview of the FACI command usage in code flash programming/erasure mode is shown below.

Table 6.3 lists the available commands in code flash programming/erasure mode. Note that security should be released by ID authentication before FACI commands are used for code flash memory.

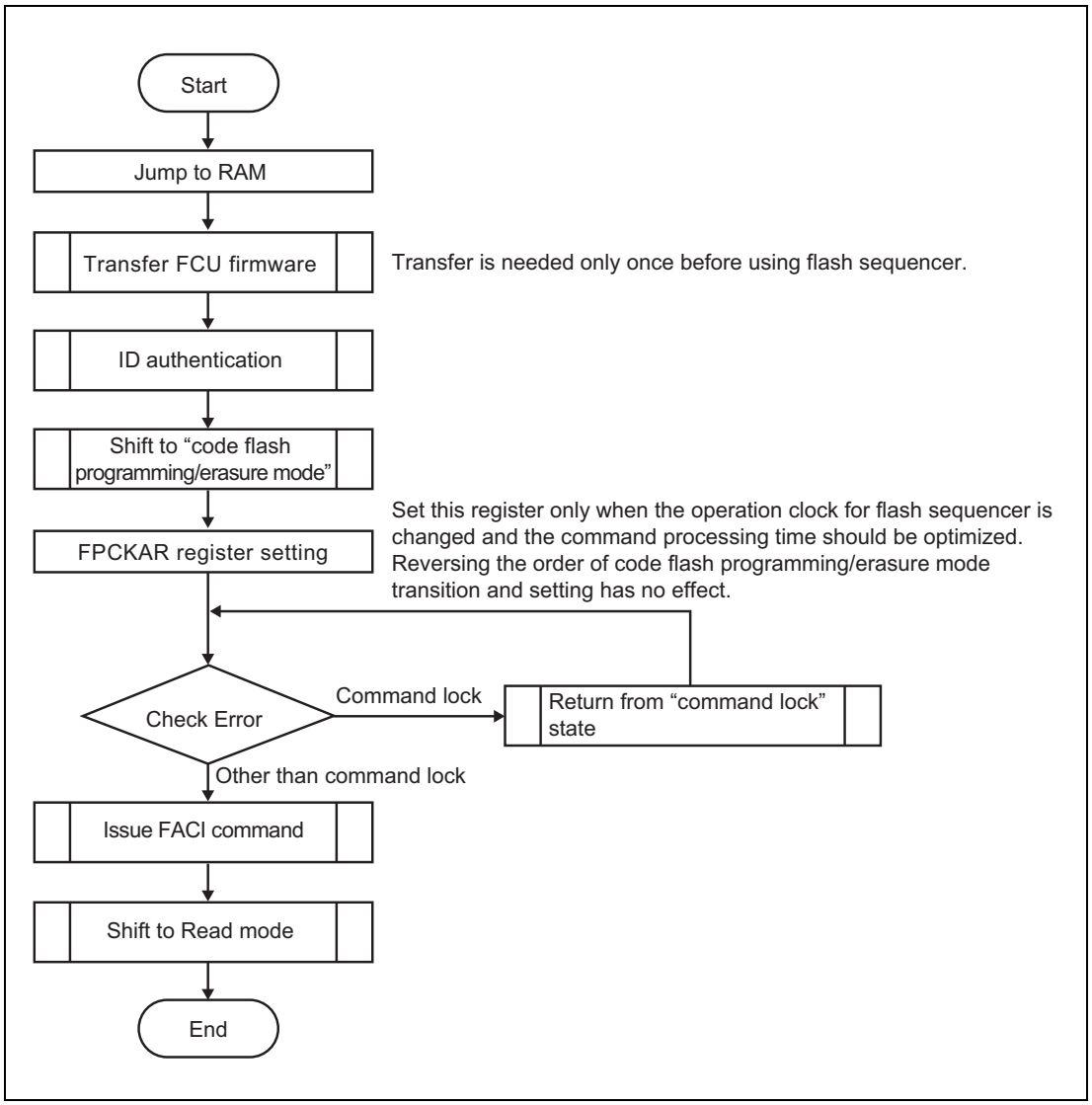


Figure 6.1 Overview of Command Usage in Code Flash Programming/Erasure Mode (1)

Processing by the E1x-FCC2 can jump to the on-chip RAM following transfer of the firmware to FCU by using FCU firmware transfer 2. This is shown in **Figure 6.2**.

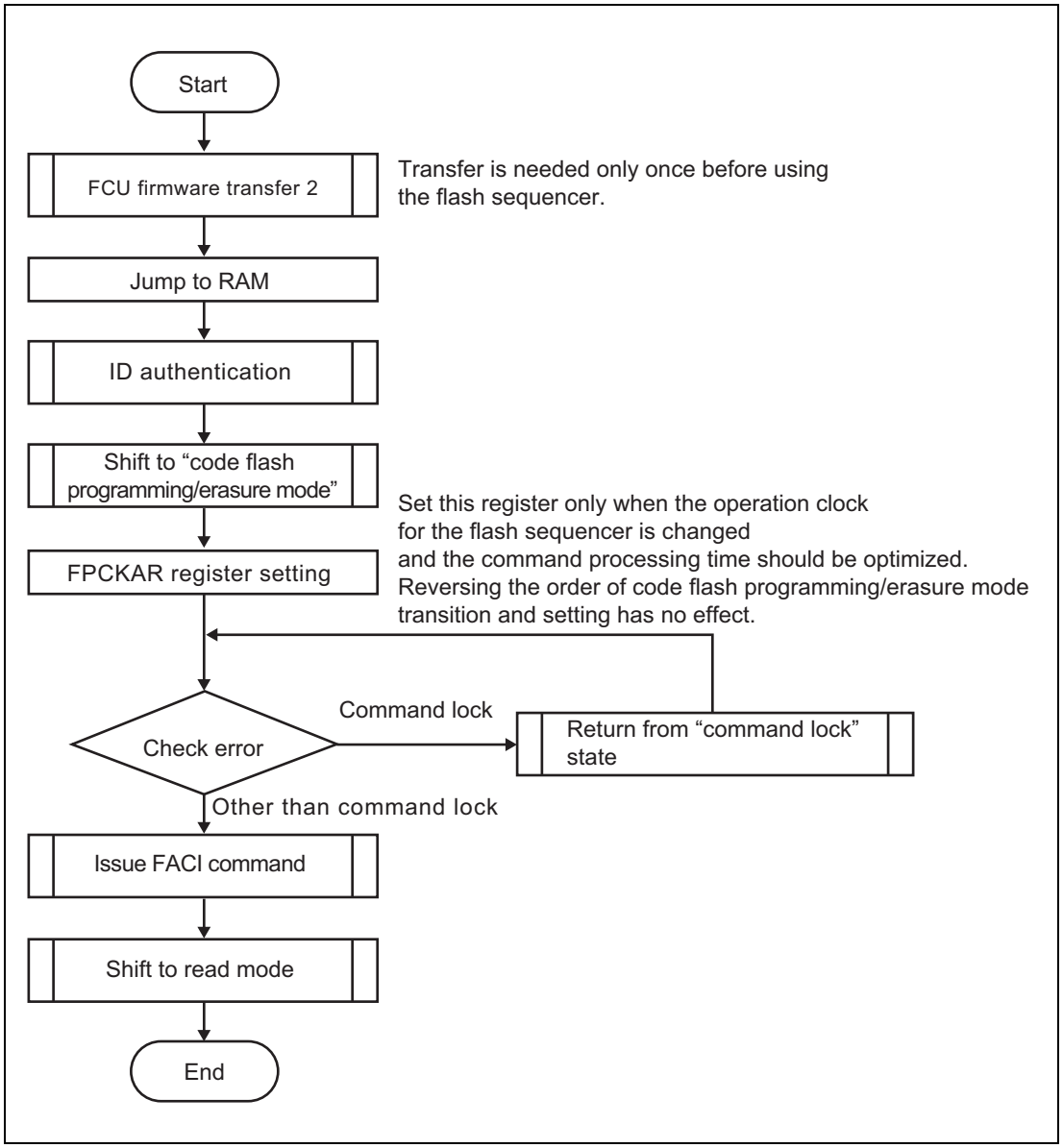


Figure 6.2 Overview of Command Usage in Code Flash Programming/Erasure Mode (2)

6.3.2 Overview of the Command Usage in Data Flash Programming/Erasure Mode

The overview of the FACI command usage in data flash programming/erasure mode is shown below.

As for the available commands in data flash programming/erasure mode, refer to **Table 6.3**.

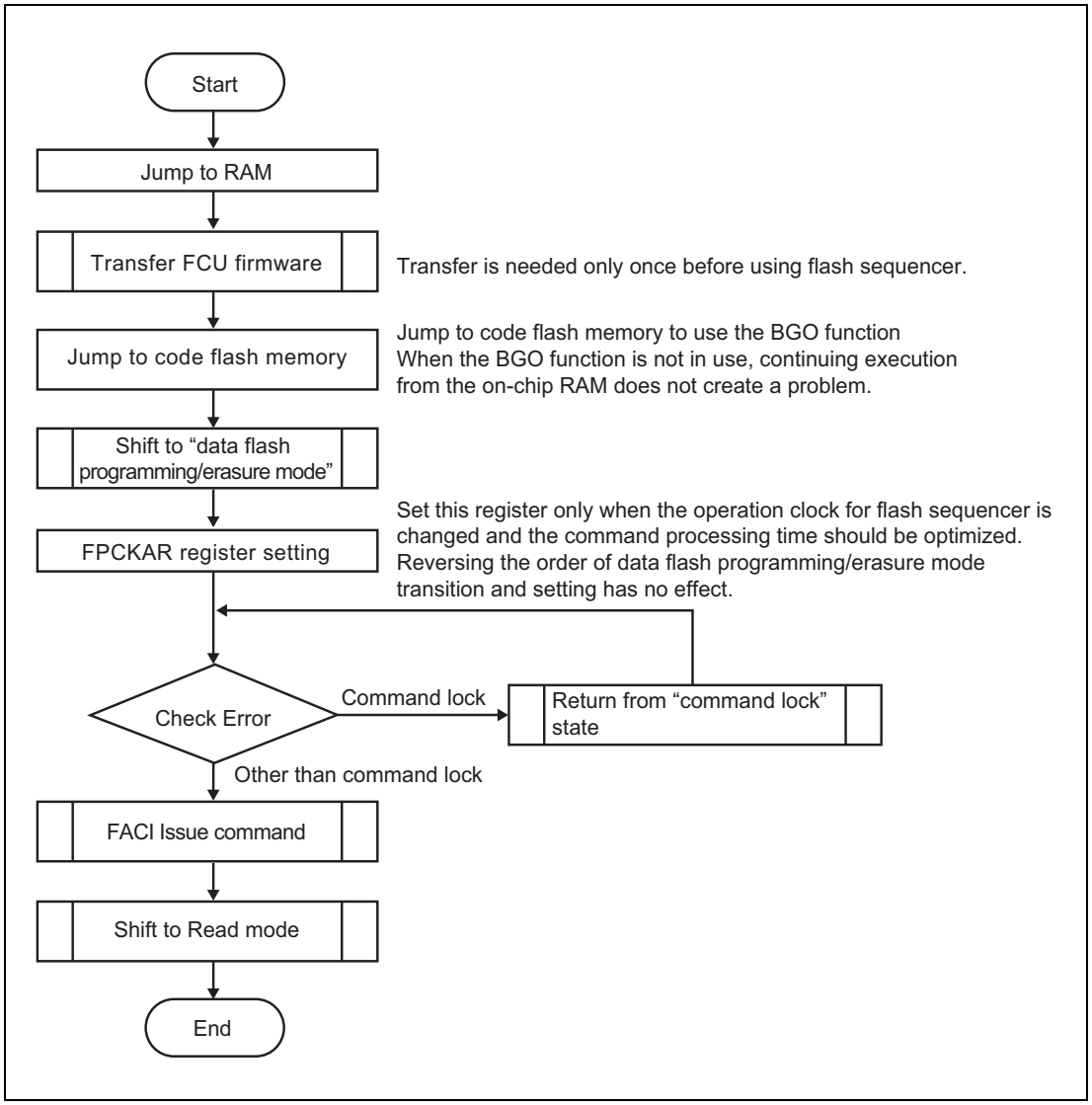


Figure 6.3 Overview of Command Usage in Data Flash Programming/Erasure Mode (1)

The jump of processing by the E1x-FCC2 to the on-chip RAM becomes unnecessary if FCU firmware transfer 2 is used in the way shown in **Figure 6.4**.

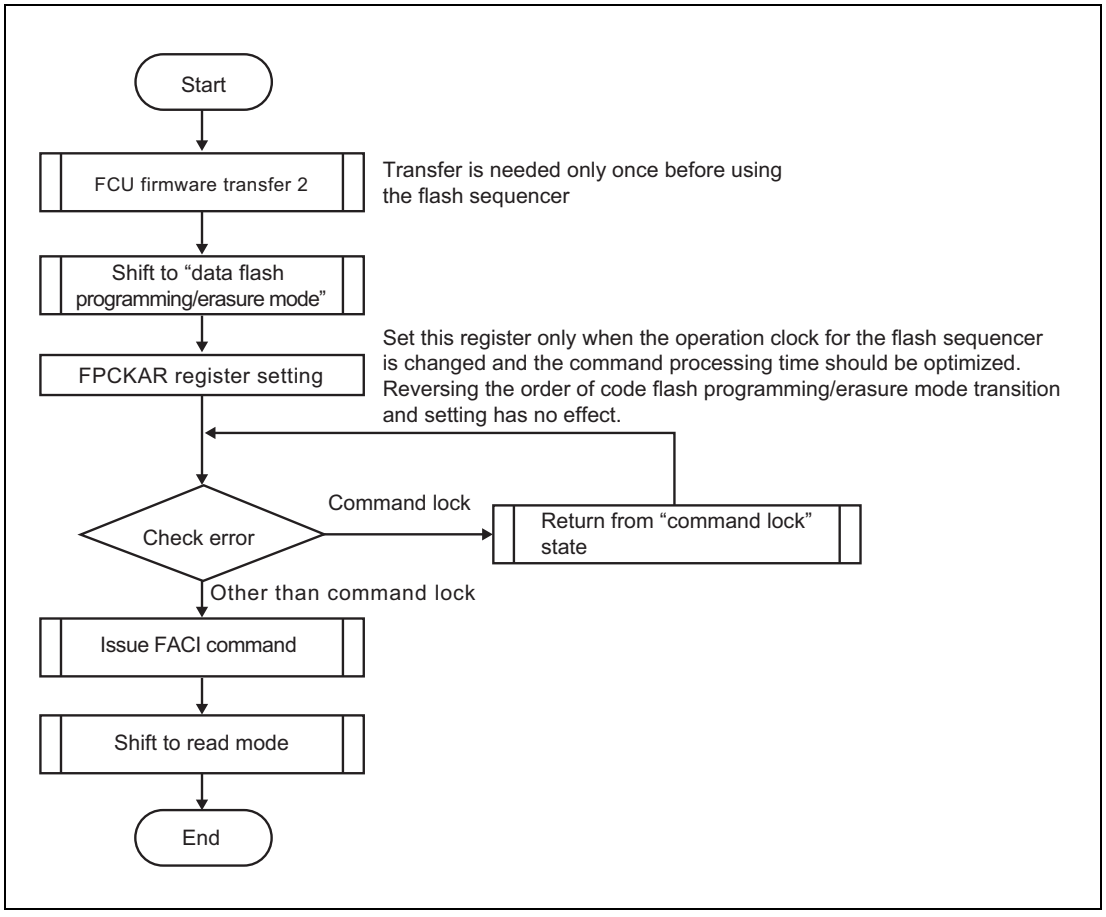


Figure 6.4 Overview of Command Usage in Data Flash Programming/Erasure Mode (2)

6.3.3 FCU Firmware Transfer

To use the flash sequencer, the FCU firmware should be stored in FCURAM. As the FCU firmware is not stored in FCURAM at power on, it is required to copy the FCU firmware from the FCU firmware storage area to FCURAM.

The flow of transfer differs in FCU firmware storage area 2, but both are the same FCU firmware. See **Figure 6.1** to **Figure 6.6**.

Executing the FACI command does not update the FCURAM. If FCU firmware is copied once before the flash sequencer is used, re-updating the FCURAM is not necessary.

As the FCURAM storage data is undefined at power on, an ECC error is generated by the write to FCURAM. After copying the FCU firmware, issue a forced stop command to initialize the FRCRCT and FRDTCT bits in the FSTATR register. In this case, reloading the FCU firmware after issuance of the forced end command is unnecessary.

Forced stop command processing is all implemented in the hardware. Thus, the forced stop command can be executed correctly even before FCU firmware is stored or even when copying of the FCU firmware has failed. An ECM “flash access error” source is generated by an ECC error in the FCURAM. Clear the error source by writing 1 to the ECMCLSSE106 bit in the ECMESSTC1 register. The ECMESSTC1 register is protected against unauthorized writing. Regarding writing to the register, see *Section 31, ECM*, in the user’s manual.

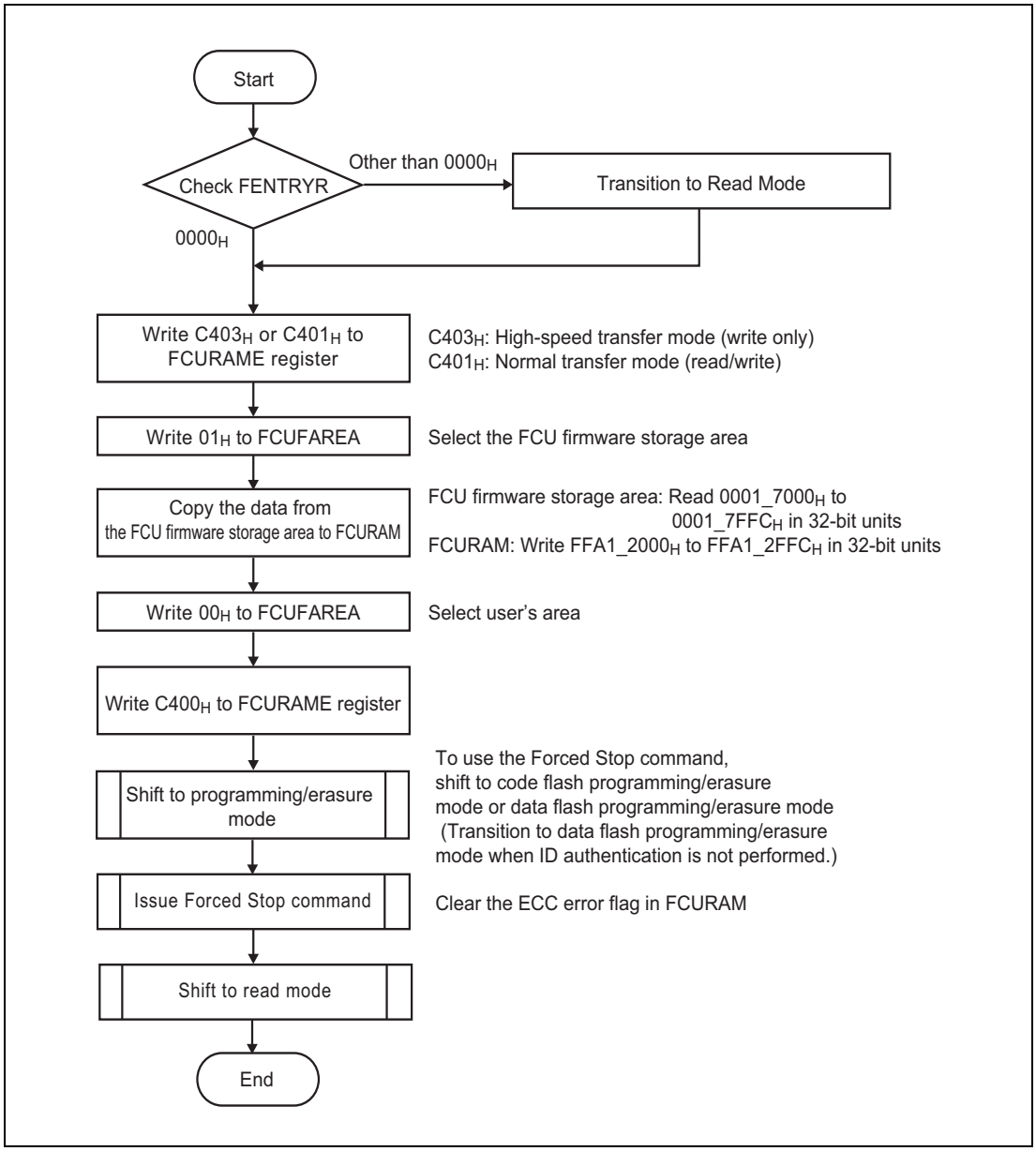


Figure 6.5 Transfer Flow of FCU Firmware (1)

The jump of processing by the E1x-FCC2 to the on-chip RAM becomes unnecessary if FCU firmware transfer 2 is used in the way shown in **Figure 6.6**. See also **Figure 6.2** and **Figure 6.4**.

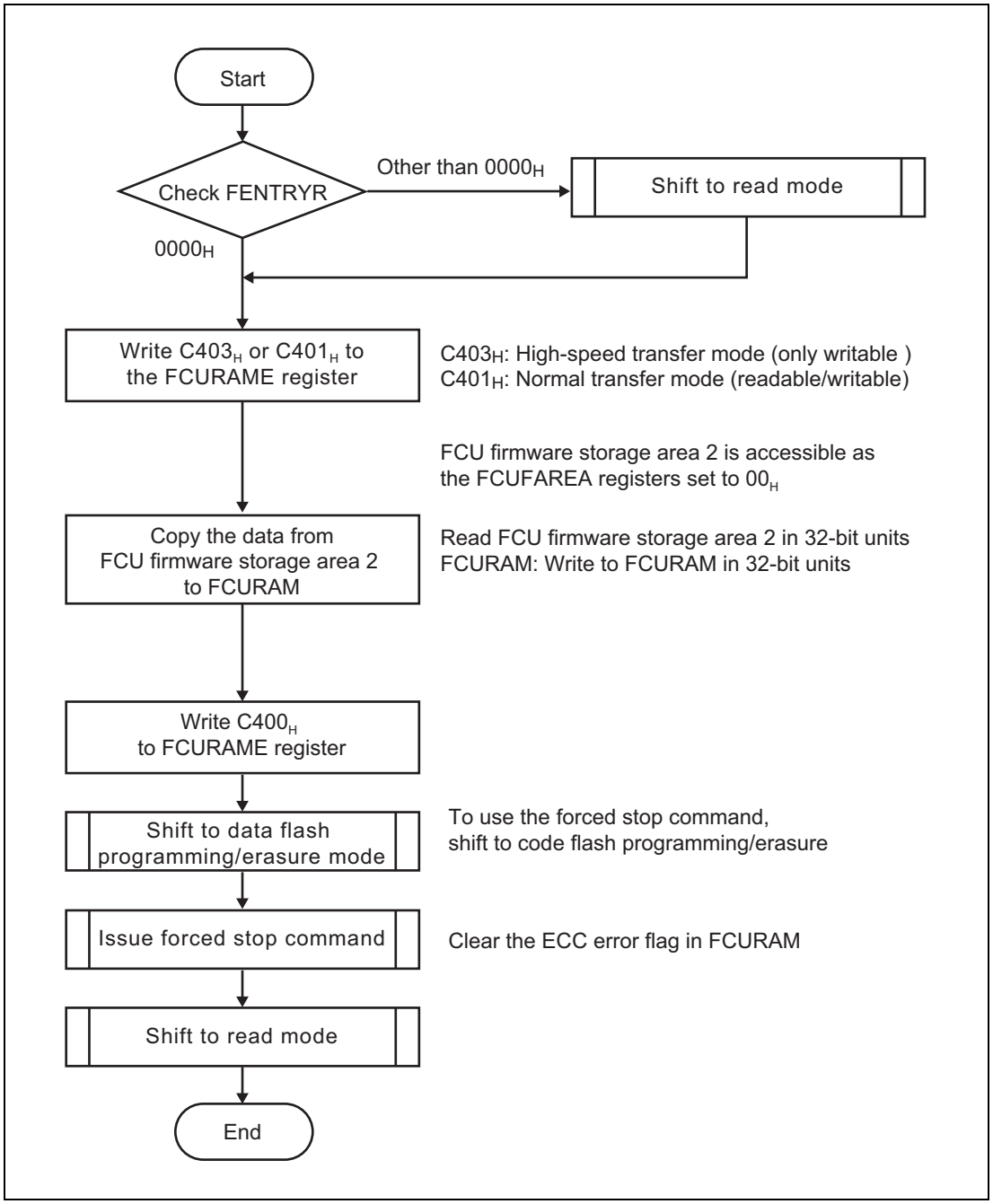


Figure 6.6 Transfer Flow of FCU Firmware (2)

Figure 6.7 shows the configuration of the FCU firmware. The area from BaseAddress to (BaseAddress + #CODE_END) holds the instruction codes to be executed by the FCU. The area from (BaseAddress + #CODE_END + 4) to (BaseAddress + 0001_7FF7_H) is reserved.

When data is copied from the FCU firmware storage area to FCURAM, data of 4 Kbytes including the reserved area needs to be copied.

Location (BaseAddress + $0FF8_H$) holds the number of bytes in the code area divided by 4.

Location (BaseAddress + $0FFC_H$) holds a checksum, which is the two-byte result of adding all values in the code area.

After the FCU firmware is transferred to the FCURAM from the FCU firmware storage area, check the contents of the FCURAM by calculating the checksum of the code area ($FFA1\_2000_H$ to #CODE_END_RAM) in the FCURAM and comparing it with the checksum stored at address (BaseAddress + $0FFC_H$) of the FCU firmware storage area.

FCU firmware storage area : BaseAddress = 0001_7000_H

FCU firmware storage area 2: BaseAddress = 0103_7000_H

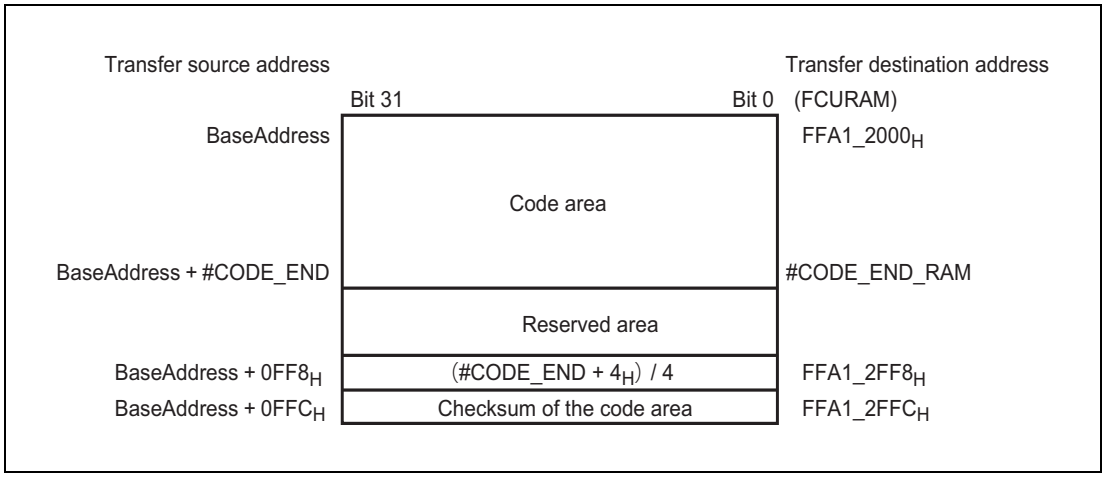


Figure 6.7 Configuration of FCU Firmware

6.3.4 Shift to Code Flash Programming/Erasure Mode

To use the FACI commands relating the code flash memory, operation should be shifted to the code flash programming/erasure mode. Set the FENTRYRC bit in the FENTRYR to 1 to shift to the code flash programming/erasure mode.

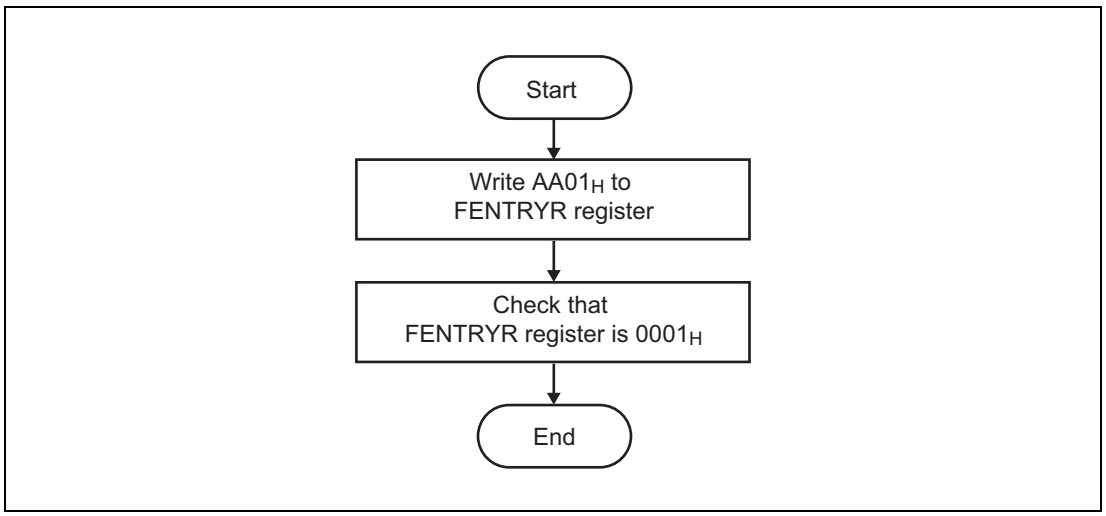


Figure 6.8 Flow of Shift to Code Flash Programming/Erasure Mode

6.3.5 Shift to Data Flash Programming/Erasure Mode

To use the FACI commands relating the data flash memory, operation should be shifted to the data flash programming/erasure mode. Set the FENTRYRD bit in the FENTRYR to 1 to shift to the data flash programming/erasure mode.

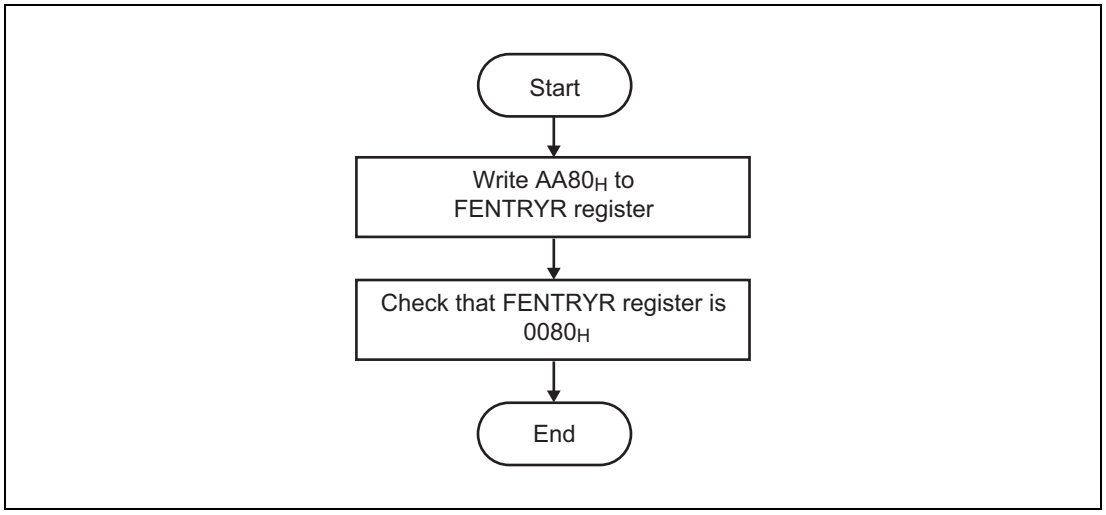
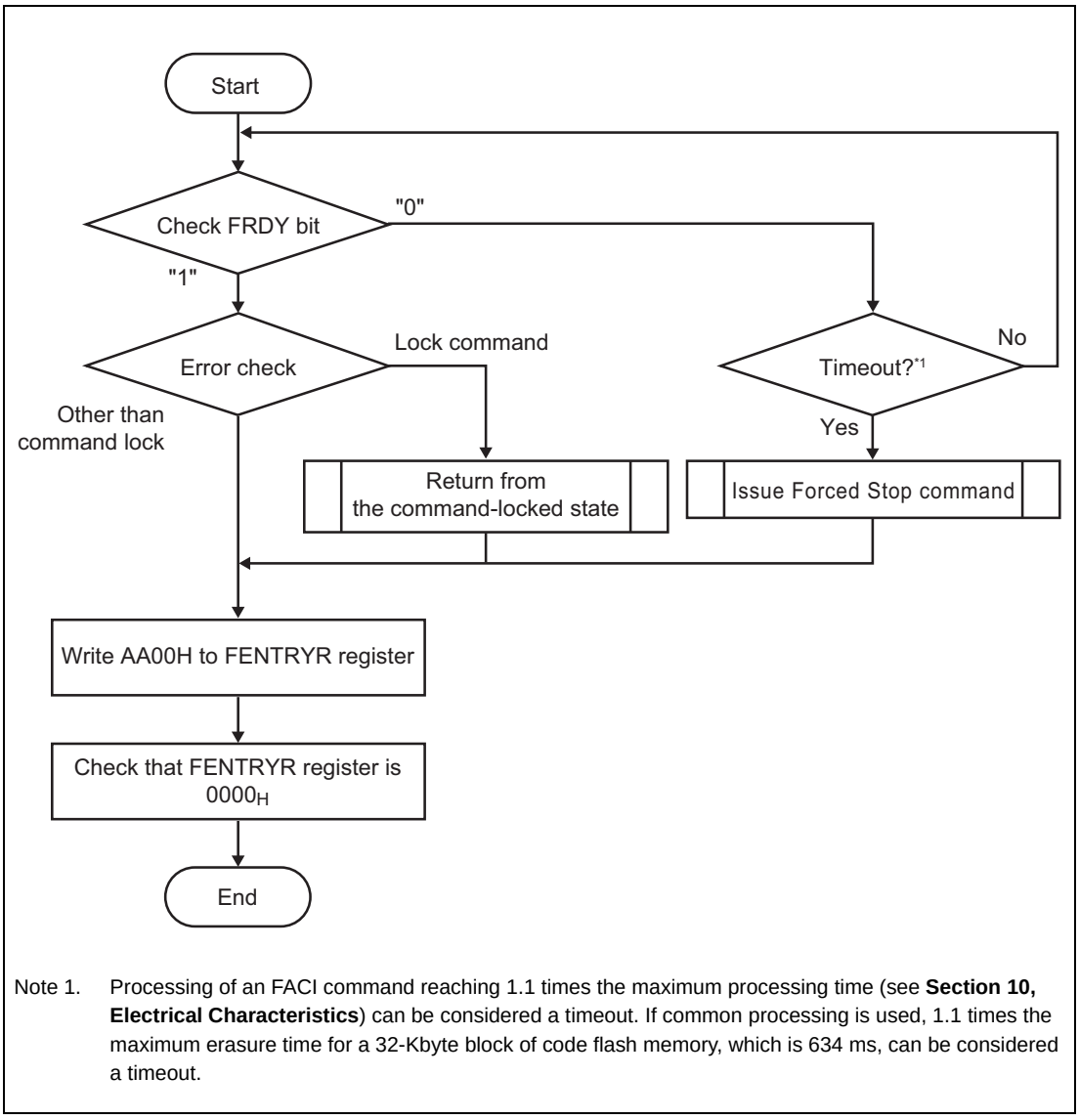


Figure 6.9 Flow of Shift to Data Flash Programming/Erasure Mode

6.3.6 Shift to Read Mode

To read the flash memory without using the BGO function, the operation should be shifted to the read mode. To shift to the read mode, set the FENTRYR register to 0000_H.

When entering the read mode, the flash sequencer processing should be completed and the operation is in other than command lock state.



Note 1. Processing of an FACI command reaching 1.1 times the maximum processing time (see **Section 10, Electrical Characteristics**) can be considered a timeout. If common processing is used, 1.1 times the maximum erasure time for a 32-Kbyte block of code flash memory, which is 634 ms, can be considered a timeout.

Figure 6.10 Flow of Shift to Read Mode

6.3.7 ID Authentication

To use the FACI command in code flash programming/erasure mode, release security by ID authentication and the IDST bit of the SELFIDST register should be 0.

When the IDST bit is 1, the FACI command is not accepted. **Figure 6.11** shows the ID compare method using SELFID0 to SELFID3, and how the compare result is checked by SELFIDST.

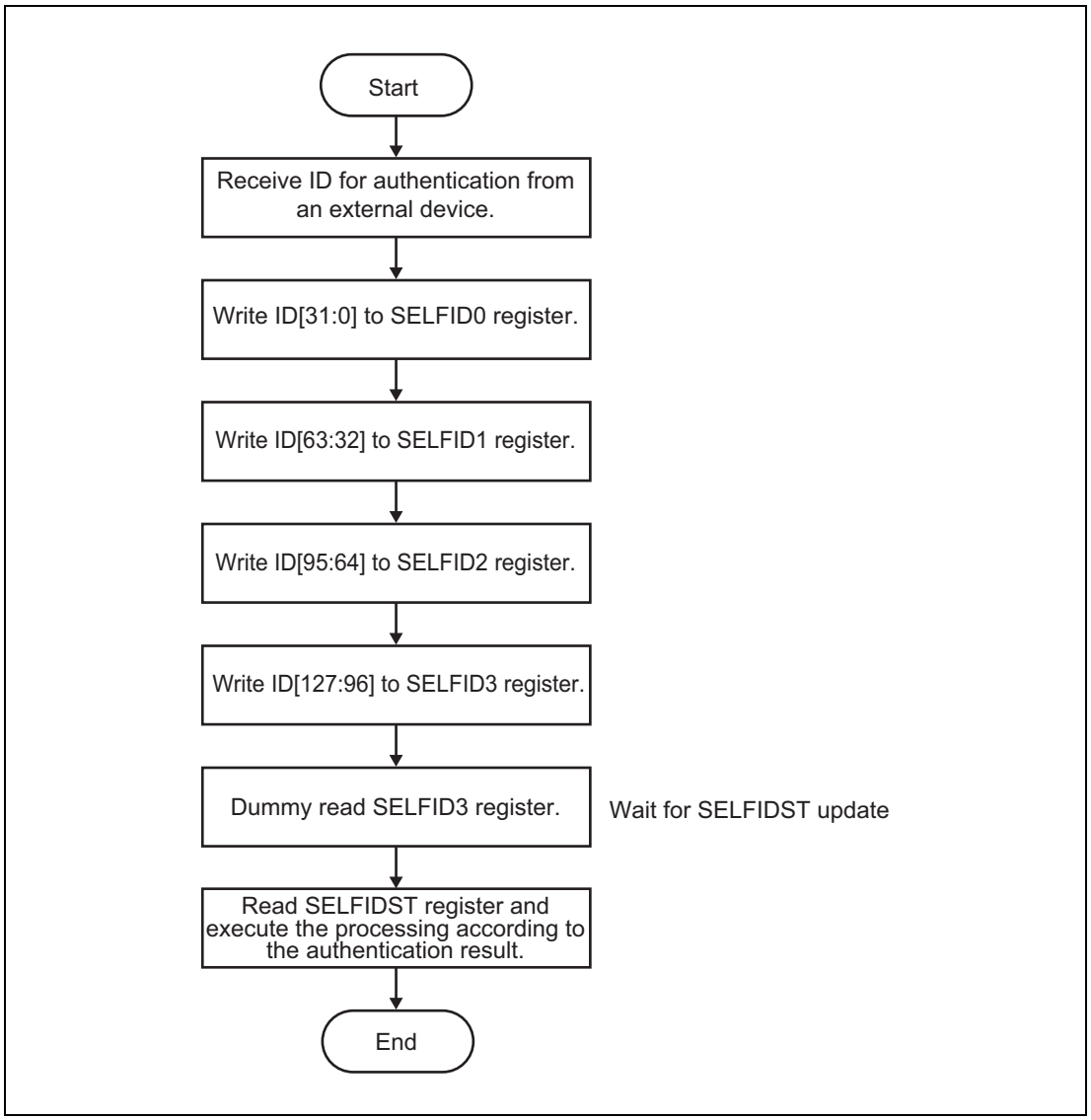


Figure 6.11 Flow of ID Compare

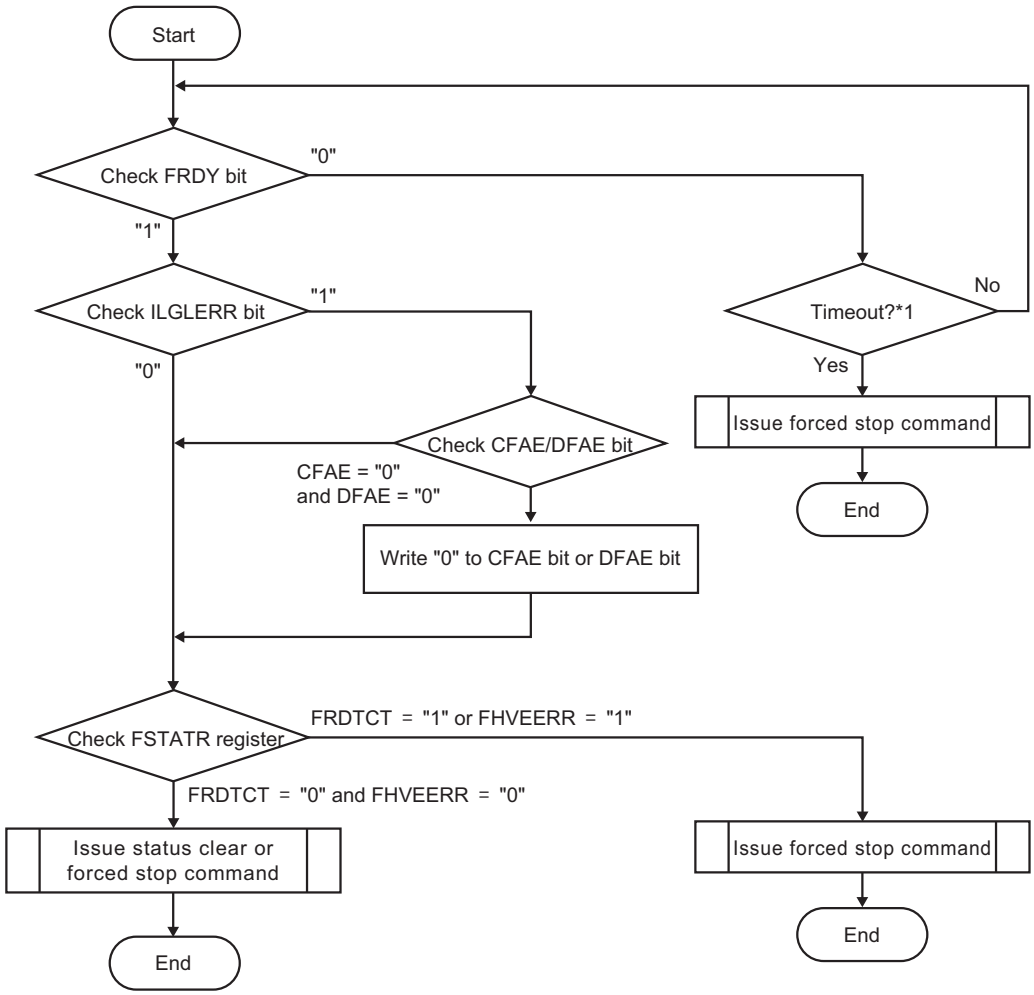
6.3.8 Return from Command Lock State

When the flash sequencer enters the command lock state, FACI commands cannot be accepted. To release the command lock state, use the status clearing command, forced stop command, or FASTAT register.

When the command lock state is detected by checking an error before issuing the programming/erasure suspension command, the FRDY bit in the FSTATR register may hold 0 without completing the command processing. If the processing is not completed within the maximum programming/erasure time specified by electrical characteristics, it is determined as timeout and the flash sequencer should be stopped by the forced stop command.

When the ILGLERR bit in the FSTATR register is 1, check the FASTAT value. If the CFAE bit or DFAE bit in the FASTAT register is 1, the command lock state cannot be released by the status clearing or forced stop command.

The FRDTCT, and FHVEERR bits in the FSTATR register are not changed from 1 to 0 by the status clearing command. When these bits are set to 1, use the forced stop command to release the command lock state. The other bits to be the command lock source can be changed from 1 to 0 by the status clearing or forced stop command.



Note 1. Processing of an FACI command reaching 1.1 times the maximum processing time (see **Section 10, Electrical Characteristics**) can be considered a timeout. If common processing is used, 1.1 times the maximum erasure time for a 32-Kbyte block of code flash memory, which is 634 ms, can be considered a timeout.

Figure 6.12 Return from Command Lock State

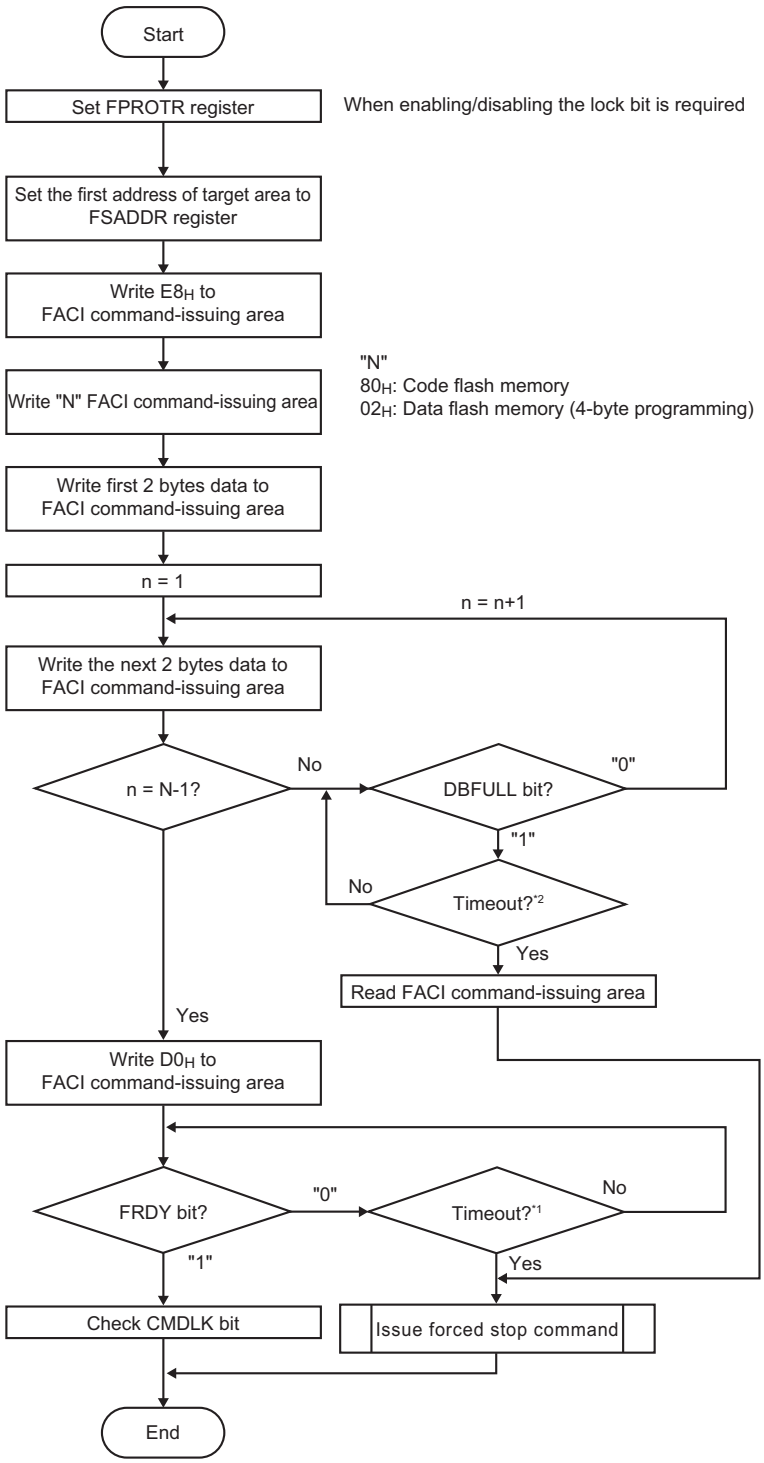
6.3.9 Issuing of Programming Command

The programming command is used to write to user area and data area.

Before issuing the programming command, set the first address of target area to the FSADDR register. Writing $D0_H$ to the FACI command-issuing area at the final access of the FACI command-issuing starts the programming command processing. If the target area of programming command processing contains the area not for writing, write $FFFF_H$ to the corresponding area.

Set the FPROTR register before issuing the programming command. To set the FPROTR register is required to switch enabling/disabling the lock bit.

If issuing the programming command is kept while the FACI internal data buffer is full, wait is generated in the peripheral-bus and it may affect the communication performance of other peripheral IPs. To avoid the wait generation, the DBFULL bit in FSTATR should be 0 when FACI commands are issued. In addition, writing to data area does not make the data buffer full.



Note 1. For the code flash memory, the time of 256-byte programming (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times is judged as a timeout. For the data flash memory, the time of 4-byte programming (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times is judged as a timeout. If common processing is used, the time of 256-byte programming of the code flash memory reaching 1.1 times is judged as a timeout.

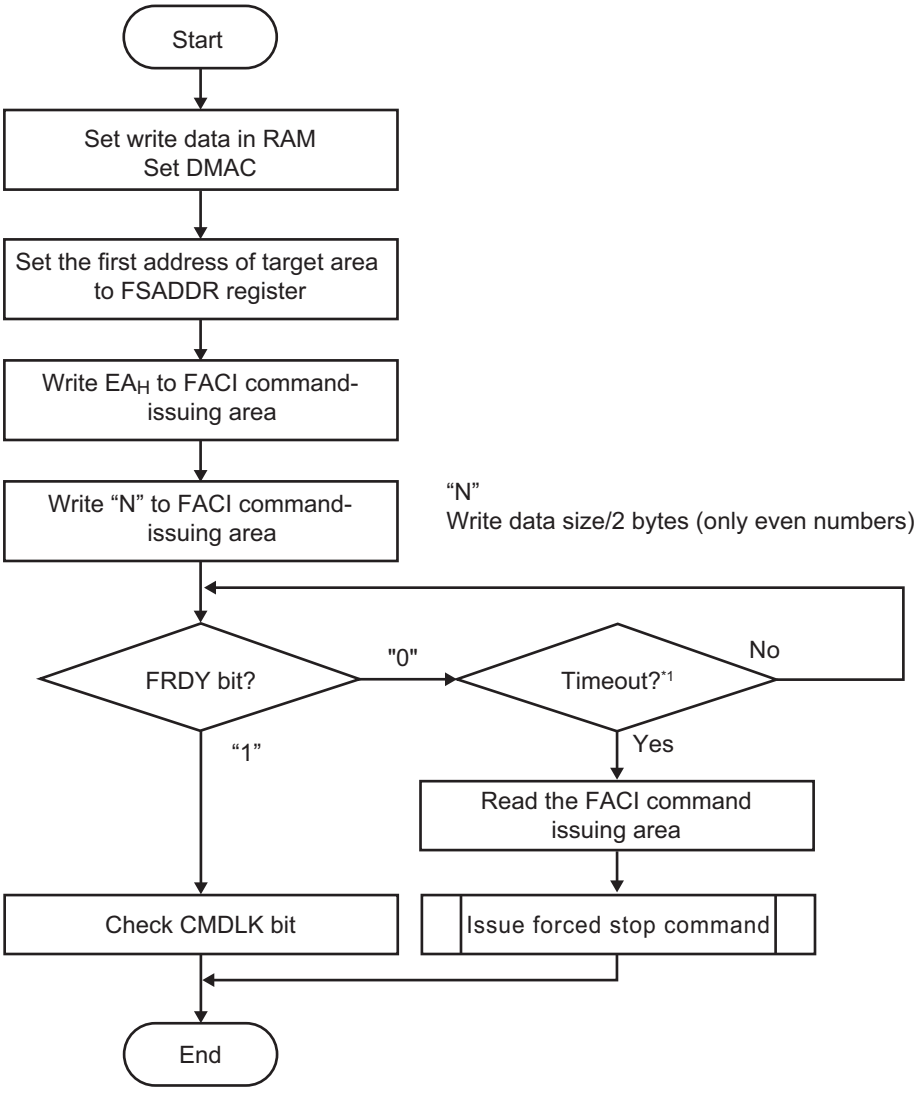
Note 2. The time reaching 5 us is judged as a timeout.

Figure 6.13 Programming Command Usage

6.3.10 DMA Programming Command

The DMA programming command is used to program multiple 4-byte data transferred from the DMAC to the data area. Thus, a large amount of data can be programmed continuously with reduced CPU load.

Before issuing a DMA programming command, set the first address of target area to the FSADDR register. Set the write data in the RAM and set the DMAC to perform DMA transfer from the pertinent area to the FACI command-issuing area. FACI issues a data transfer request to DMAC after reception of the DMA programming command and every time the writing of 4 bytes of data is completed. Set up the DMAC so that 2 bytes of data will be transferred twice for each data transfer request. For the usage of the DMAC, see *Section 7, DMA* in the user's manual.



Note 1. Judged based on the sum of the time of 4-byte programming of the data flash memory (the maximum time: see **Section 10, Electrical Characteristics**) × "N"/2 × 1.1 and the time required for DMA transfer of write data.

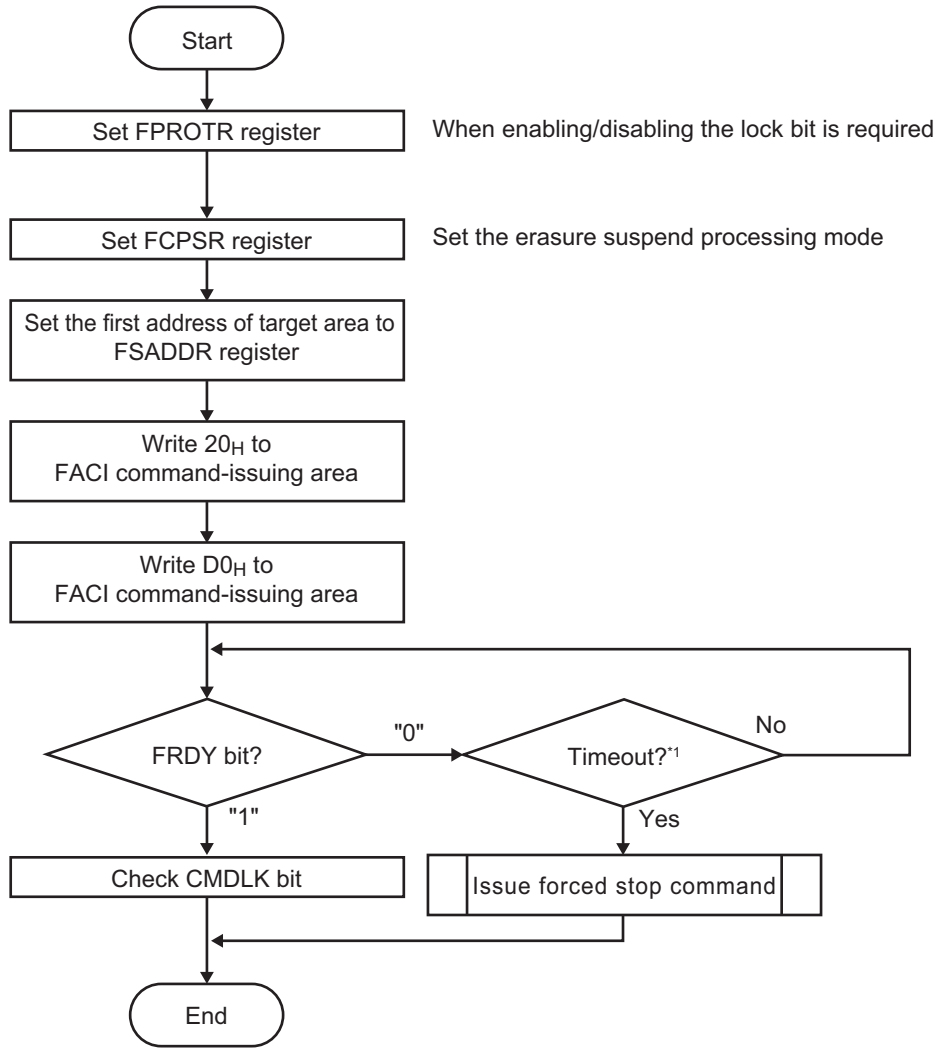
Figure 6.14 DMA Programming Command Usage

6.3.11 Block Erasure command

The block erasure command is used to erase the user area, lock bit, and data area.

Before issuing the block erasure command, set the first address of target area to the FSADDR register. Writing 20_H and $D0_H$ to the FACI command-issuing area starts the block erasure command processing.

Set the FPROTR and FCPSR registers before issuing the block erasure command. To set the FPROTR register is required to switch enabling/disabling the lock bit. To erase the lock bit, issue the block erasure command while the FPROTCN bit in the FPROTR register is 1. To set the FCPSR register is required to switch the suspending method by the programming/erasure suspension command (suspension-priority mode/erasure-priority mode).



Note 1. For the code flash memory, the time of erasure of a 32-Kbyte block (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times is judged as a timeout. For the data flash memory, the time of erasure of a 64-byte block (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times is judged as a timeout. If common processing is used, the time of erasure of a 32-Kbyte block of the code flash memory reaching 1.1 times is judged as a timeout.

Figure 6.15 Block Erasure Command Usage

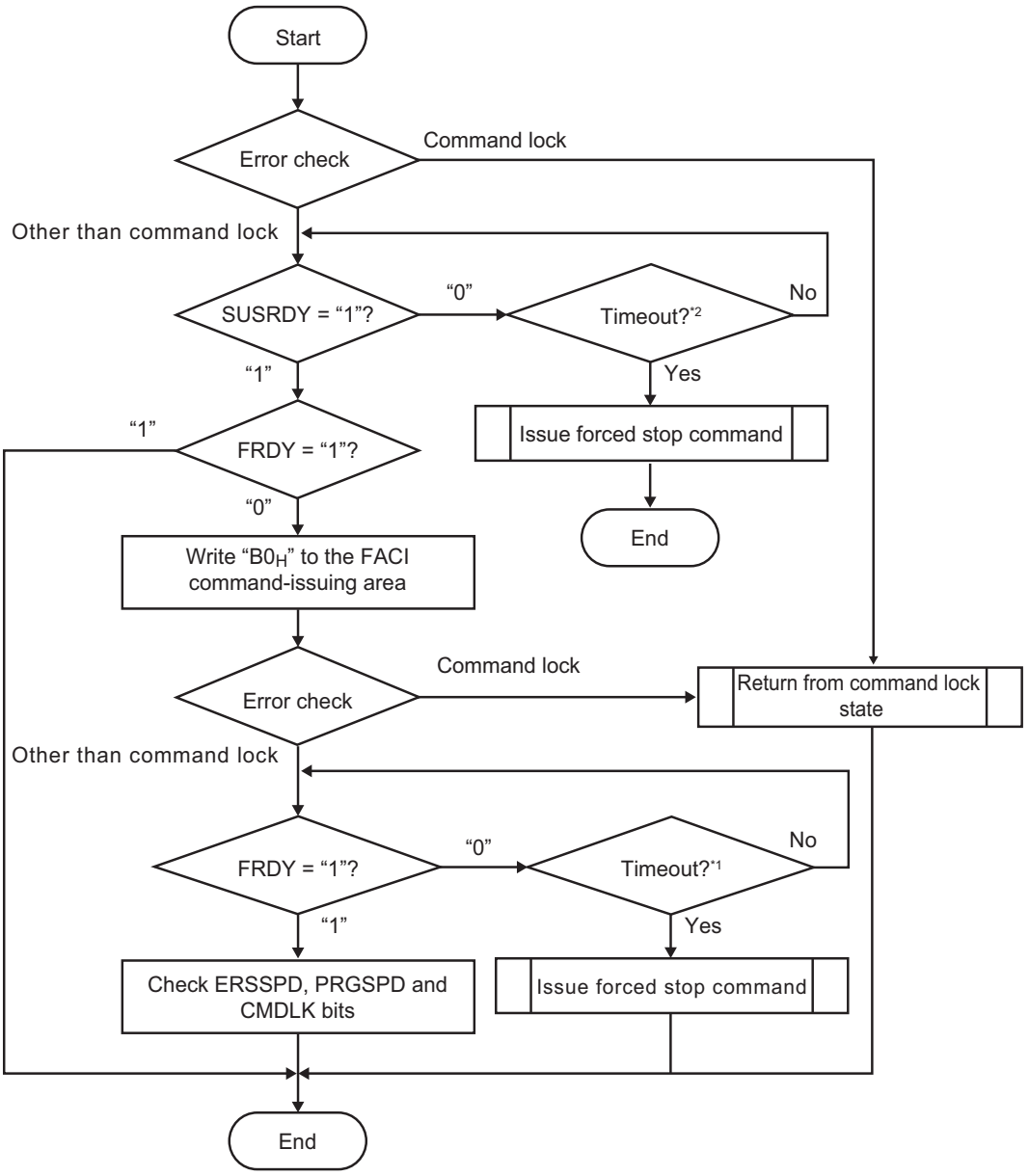
6.3.12 Programming/Erasure Suspension Command

The programming/erasure suspension command is used for suspending programming or erasure command processing.Before issuing the programming/erasure suspension command, check that CMDLK bit is 0 to ensure that programming or erasure command processing is being performed correctly. In addition, check that the SUSRDY bit is 1 to ensure that the programming/erasure suspension command is acceptable. After issuing the programming/erasure suspension command, check CMDLK bit to ensure no error has occurred.

If an error has occurred, the CMDLK bit is set to 1. If programming or erasure command processing is complete within the period from when the SUSRDY bit is ensured to be 1 until the programming/erasure suspension command is accepted, no error occurs, hence no transition to a suspended state (the FRDY bit is 1 and both ERSSPD and PRGSPD bits are 0).

Once the programming/erasure suspension command is accepted and programming or erasure command processing is normally suspended, flash sequencer enters a suspended state and that FRDY bit is 1 and ERSSPD or PRGSPD bit is 1. After issuing the programming/erasure suspension command and ensuring that flash sequencer has entered suspend state, determine which operation to perform in the succeeding process. If the programming/erasure resumption command is issued in the succeeding process while flash sequencer has not entered a suspended state, an illegal command error occurs and flash sequencer shifts to the command lock state. (see **8.2 Error Protection**)

When the operation shifts to the erasure suspend state, writing to the blocks other than those targeted for erasure is enabled. In addition, when the FENTRYR register is cleared, the operation shifts to the read mode.



Note 1. For the code flash memory, the suspend latency tSEED during erasure of the code flash memory (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times can be considered as a timeout.
For the data flash memory, the suspend latency tSEED during erasure of the data flash memory (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times can be considered as a timeout.
If common processing is used, 1.1 times tSEED (max.) of the code flash memory, which is 1.87 ms, can be considered as a timeout.

Note 2. The time reaching 13 µs is judged as a timeout.

Figure 6.16 Programming/Erasure Suspension Command Usage

(1) Suspend programming command

If a programming/erasure suspension command is issued while the flash memory is being programmed, the flash sequencer suspends programming. **Figure 6.17** show the suspending operation. Once sequencer enters a state where it is ready to accept the programming/erasure suspension command after the start of programming, SUSRDY bit is set to 1. If the programming/erasure suspension command is issued, the flash sequencer accepts the command and clears SUSRDY bit. If the flash sequencer accepts the command while applying a programming pulse, the flash sequencer continues applying the pulse. After a specified pulse application time has elapsed, the flash sequencer completes applying the pulse, resumes suspended programming, and sets the FSTATR.PRGSPD bit to 1.

Once the suspension process is complete, the flash sequencer sets the FRDY bit to 1 and enters programming suspended state. If the flash sequencer accepts the programming/erasure resumption command in this state, the flash sequencer clears the FRDY and PRGSPD bits to 0 and restarts programming.

Figure 6.17 gives an overview of operation for suspending programming command processing. Upon accepting programming command, sequencer clears FRDY bit to 0 and starts programming.

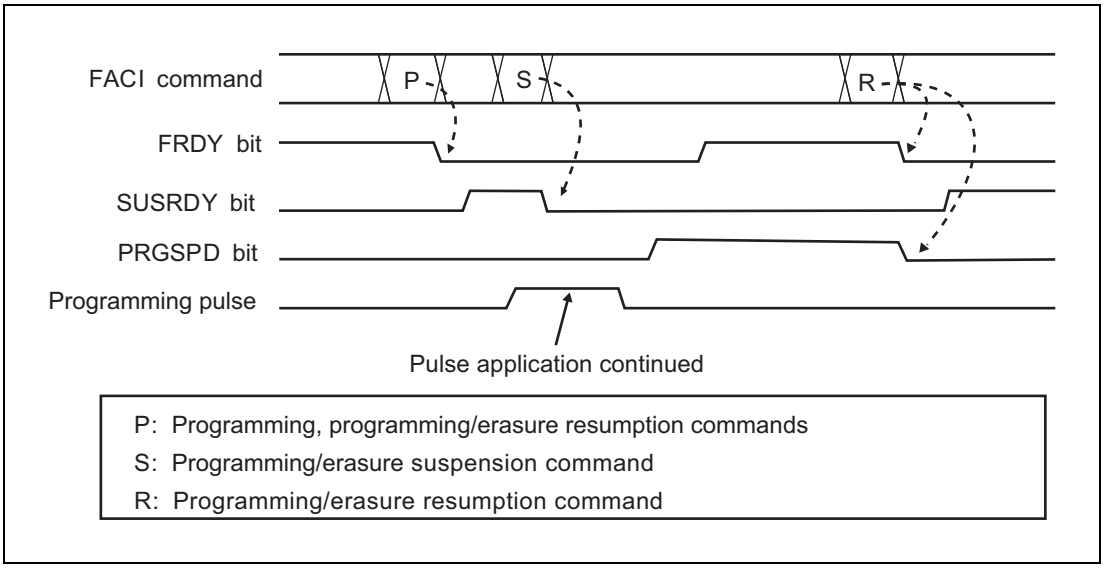


Figure 6.17 Suspend Programming Command

(2) Suspend erasure command in suspension-priority mode

The flash memory supports the suspension-priority mode as one of the methods for suspending an erasure command. **Figure 6.18** shows the operation for suspending an erasure command processing in suspension-priority mode (FCPSR.ESUSPMD = 0). Upon accepting an erasure command, sequencer clears the FRDY bit to 0 and starts erasing. Once sequencer enters a state where it is ready to accept the programming/erasure suspension command after the start of erasing, the SUSRDY bit is set to 1. If a programming/erasure suspension command is issued, sequencer accepts the command and clears the SUSRDY bit. If sequencer accepts interrupt request during its erasing operation, sequencer starts a suspending process even while applying a pulse and sets ERSSPD bit to 1. Once the suspending process completes, sequencer sets FRDY bit to 1 and enters erasing suspended state. If sequencer accepts programming/erasure resumption command in this state, sequencer clears FRDY and PRGSPD bits to 0 and restarts erasing. The operations of FRDY, SUSRDY, and ERSSPD bits are independent of the erasure-suspended mode.

The setting for erasure-suspended mode affects the control methods for an erasing pulse. In suspension-priority mode, if sequencer accepts interrupt request while applying an erasing pulse A, which has not been suspended previously, sequencer suspends the pulse application, and sequencer enters an erasure-suspended state. After sequencer resumes erasing by accepting a programming/erasure resumption command, sequencer accepts resume request while applying an erasing pulse A, sequencer continues applying the pulse. After a specified pulse application time has elapsed, sequencer completes applying the pulse, and sequencer enters an erasure-suspended state. Next, after sequencer accepts a programming/erasure resumption command, and sequencer starts applying a new pulse B, if sequencer accepts interrupt request, sequencer suspends the pulse application. In suspension-priority mode, delay due to suspension can be reduced because suspension process is given priority by suspending once every pulse application.

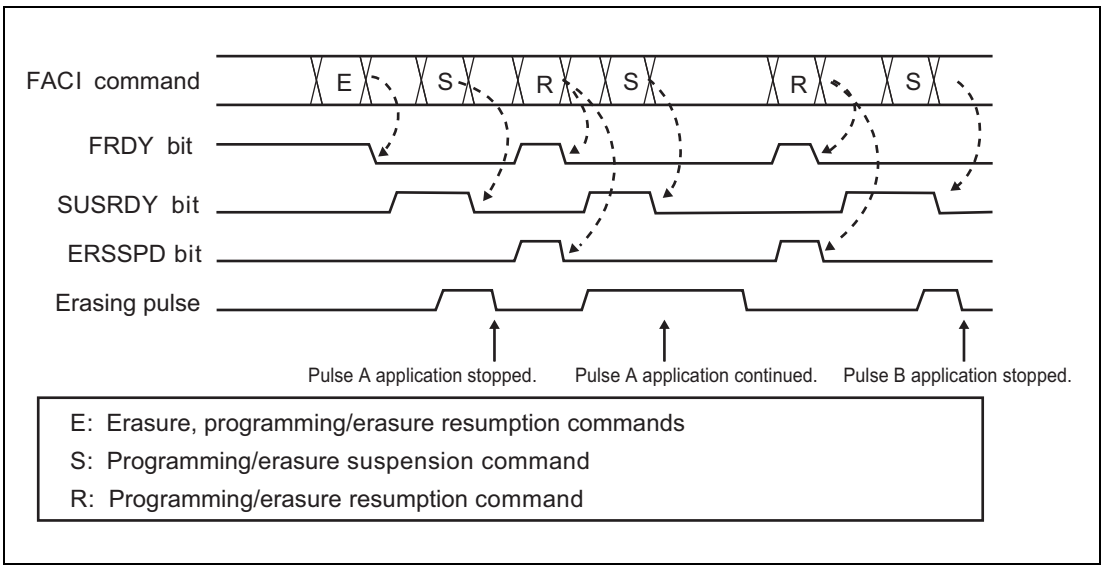


Figure 6.18 Suspend Erasure Command (Suspension-Priority Mode)

(3) Suspend erasure command in erasure-priority mode

The flash memory supports the erasure-priority mode as one of the methods for suspending an erasure command. **Figure 6.19** shows the operation for suspending an erasure command processing in erasure-priority mode (FCPSR.ESUSPMD = 1). The operation for suspending erasure command processing in erasure-priority mode is equivalent to that for suspending programming processing.

In erasure-priority mode, if sequencer accepts a programming/erasure suspension command while applying an erasing pulse, sequencer always continues applying the pulse. As processing to reapply an erasing pulse never takes place in this mode, the total time required for erasure command processing is shorter than in suspension-priority mode.

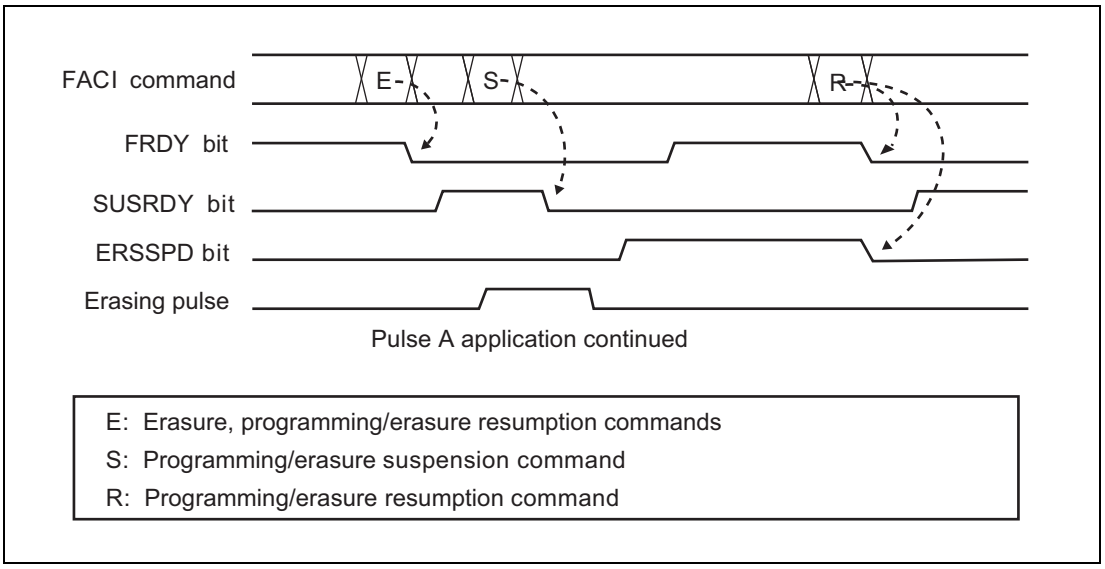
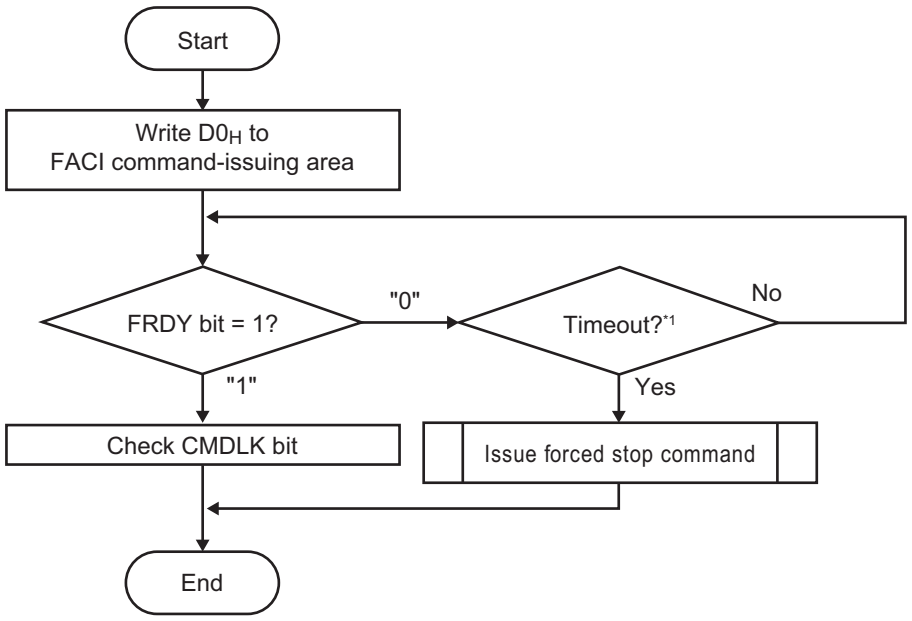


Figure 6.19 Suspend Erasure Command (Erasure-Priority Mode)

6.3.13 Programming/Erasure Resumption Command

The programming/erasure resumption command is used for resuming a programming or erasure command processing that has been suspended.

If the FENTRYR setting has been modified during suspension, issue a programming/erasure resumption command only after resetting FENTRYR to the previous value that was held before the programming/erasure suspension command was issued.



Note 1. Processing of an FACI command reaching 1.1 times the maximum processing time (see **Section 10, Electrical Characteristics**) can be considered a timeout. If common processing is used, 1.1 times the maximum erasure time for a 32-Kbyte block of code flash memory, which is 634 ms, can be considered a timeout.

Figure 6.20 Programming/Erasure Resumption Command Usage

6.3.14 Status Clearing Command

The status clearing command is used to clear the command lock state. (See **Section 6.3.8, Return from Command Lock State**.) To clear the OTPDTCT/ILGLERR/ERSERR/PRGERR/CFGDTCT/TBLDTCT bit in the FSTATR register in the command lock state, the status clearing command is available. In addition, to clear 1-bit correction flags (the OTPCRCT, CFGCRCT, and TBLCRCT bits), which do not cause transitions to the command lock state (except for FCURAM), the status clearing command is available. Status clearing command processing is all implemented in the hardware. Thus, the status clearing command can be executed correctly even when the FCU firmware is unauthorized.

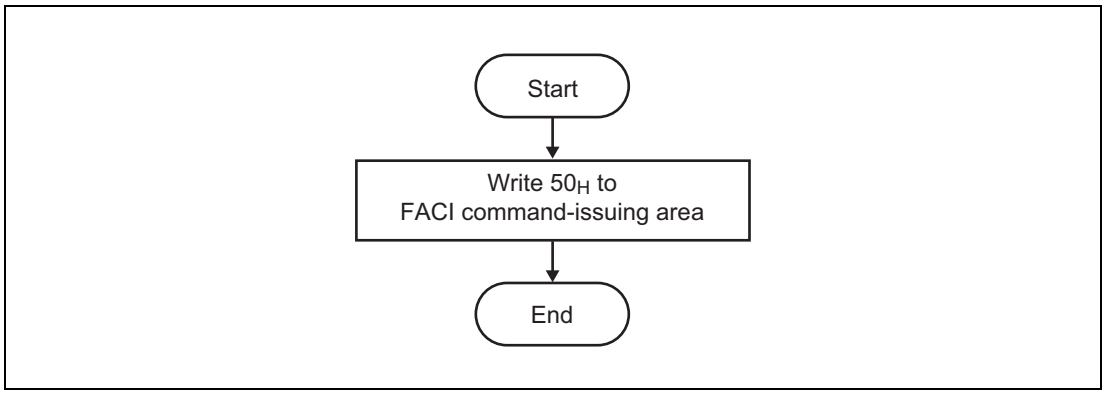


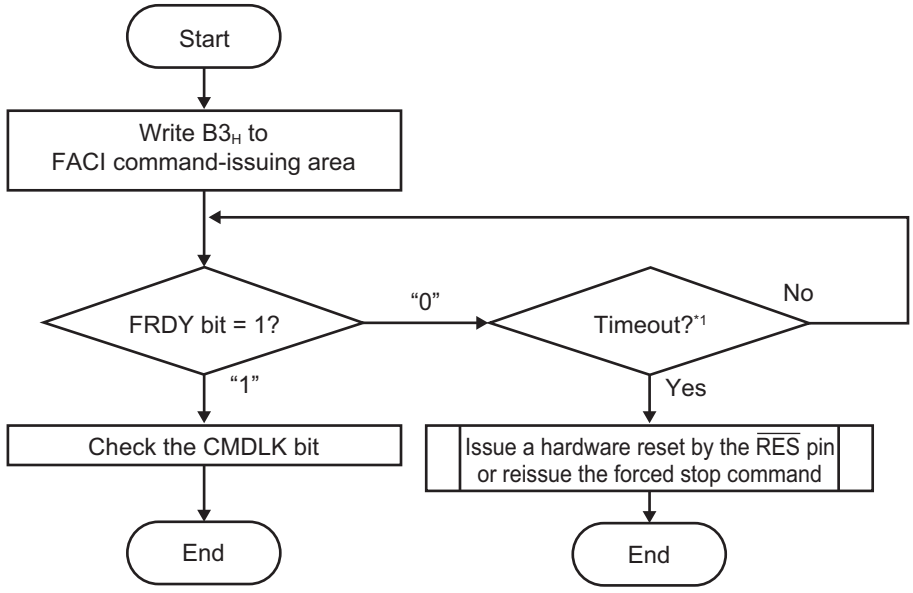
Figure 6.21 Status Clearing Command Usage

6.3.15 Forced Stop Command

The forced stop command is used to abort the command being processed by a flash sequencer. While the processing speed of this command is faster than that of programming/erasure suspension commands, it does not guarantee any result of the stopped command operation such as data in programmed or erased area. Furthermore, it is not possible to resume the suspended processing later. The suspended programming or erasure is counted as one from the perspective of programming endurance.

When a forced stop command is issued, the whole FCU and a part of FACI are initialized as well as the FSTATR register. This enables forced stop command in recovery from a command lock state or during handling of timeout in flash sequencer operation mode. (See **Section 6.3.8, Return from Command Lock State**.)

Forced stop command processing is all implemented in the hardware. Thus, the forced stop command can be executed correctly even when the FCU firmware is unauthorized.



Note 1. The forced stop command latency (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times is judged as a timeout.

Figure 6.22 Forced Stop Command Usage

- **Issuing the Forced Stop Command while Another Command is Being Issued**

  If the forced stop command is used to suspend processing when a timeout of the programming command occurs when checking the DBFULL bit, when a timeout for injection of ECC errors occurs when checking the DBFULL or EBFULL bit, or when a timeout of the DMA programming command occurs, writing to the FACI command issuing area may be handled as writing of data by the programming command. If this is the case, read the FACI command issuing area to intentionally lock commands and issue the forced stop command by following the procedure for returning from the command-locked state.

  Locking commands is possible in any case where the unit for reading the FACI command issuing area is 8, 16, or 32 bits.

6.3.16 Blank Checking Command

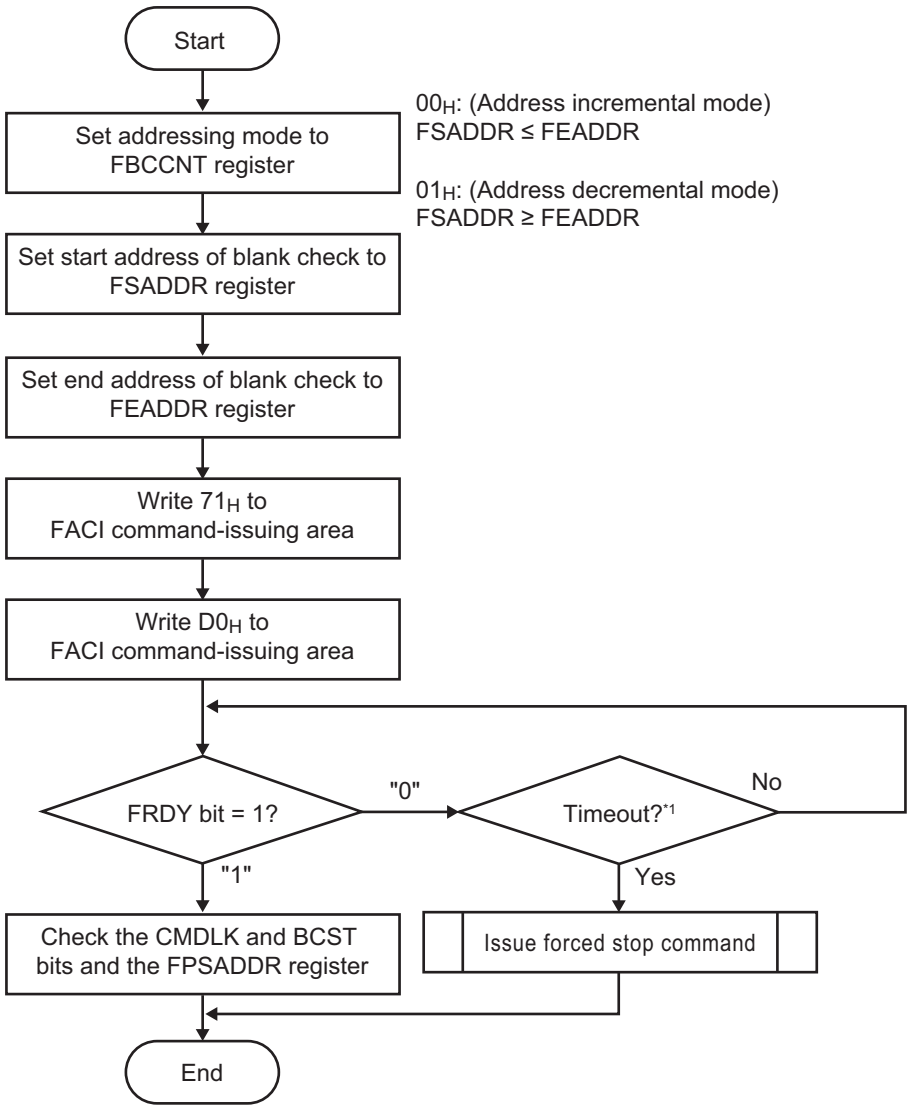
The values of the data flash memory in which no data are programmed after erasure (non-programmed state) are undefined. Thus, the blank checking command is required to confirm the non-programmed state. For how to confirm the non-programmed state of the code flash memory, see **Section 8.4, Blank Checking of Code Flash Memory**.

Before issuing the blank checking command, set addressing mode, start address, and end address to FBCCNT, FSADDR, and FEADDR register, respectively. When blank check addressing mode is set to incremental mode (i.e. FBCCNT.BCDIR = 0), address specified in FSADDR should be smaller than address in FEADDR. Conversely, address in FSADDR should be equal to or larger than address in FEADDR when blank check addressing mode is set to decremental mode (i.e. FBCCNT.BCDIR = 1). When the FBCCNT.BCDIR is 0 and addressing mode for blank check handling is in addition mode, the setting value of FSADDR needs to be set at the setting value of the FEADDR register or below.

If setting of BCDIR, FSADDR, and FEADDR are inconsistent, FACI detects error and flash sequencer enters command lock state. Blank checking unit can be set from 4 bytes to 64 K bytes.

Write 71_H and $D0_H$ to the FACI command-issuing area to start blank checking command processing. Completion of command processing can be confirmed by FRDY bit of FSTATR register. At the end of processing, the result of blank checking is stored in the BCST bit in the FBCSTAT register. If non-blank data exists within blank checked area, flash sequencer stops blank checking operation. In this case, address of non-blank data is indicated to FPSADDR register.

The erasure state can be checked by this command only for an area for which erasure processing has been correctly completed. If erasure is not correctly completed (for example, due to a reset input or power shutdown), the erasure state cannot be checked by this command.

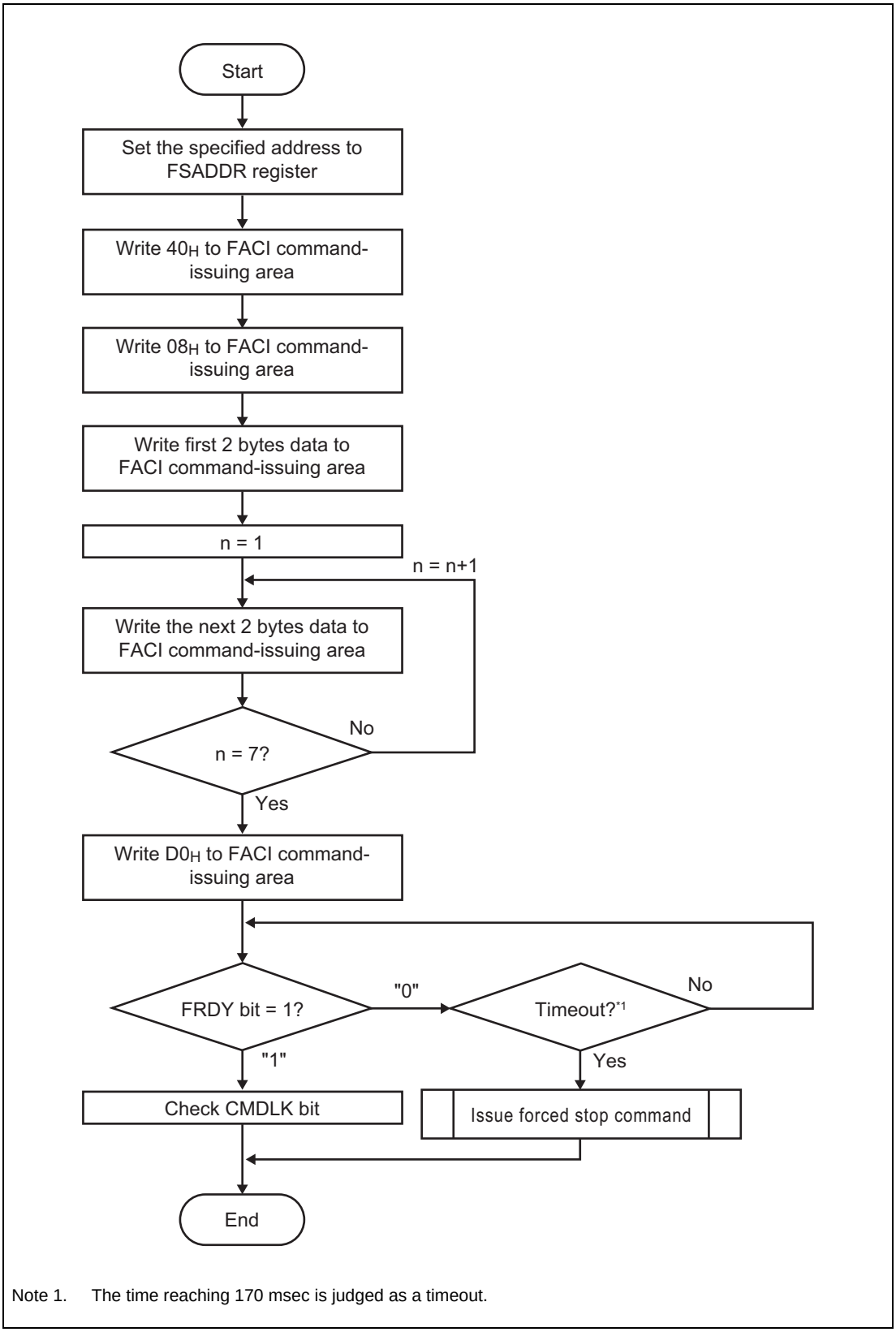


Note 1. The blank checking time (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times is judged as a timeout.

Figure 6.23 Blank Checking Command Usage

6.3.17 Configuration Programming Command

The configuration programming command is used to set the ID, security function, safety function, and option byte. Before issuing the configuration programming command, set the specified address (shown in **Table 6.5**) to the FSADDR register. Writing $D0_H$ to the FACI command-issuing area at the final access of the FACI command issue starts the configuration programming command processing.



Note 1. The time reaching 170 msec is judged as a timeout.

Figure 6.24 Config Programming Command Usage

As for the configuration programming settable data and the address value to be set in the FSADDR register, refer to **Table 6.5**. Once 0 is set as data in the security setting area, it cannot be changed to 1.

Data in other areas can be change to any value each time the configuration programming command is executed.

Table 6.5 Address Used by Configuration Programming Command

Address	Setting Data
FF30_0080$_H$	Option byte 32 to 17
FF30_0070$_H$	Option byte 16 to 1
FF30_0050$_H$	ID for authentication
FF30_0040$_H$	Security

Table 6.6 shows the addresses of option byte setting area. Initial settings for peripheral modules are stored in the option byte area. Bit positions of the option bytes, see *Section 34.10, Option Bytes* in the user's manual.

Table 6.6 Addresses of Option Byte Area

Address	Option byte area
FF30_008C$_H$	Option byte 32 to 29
FF30_0088$_H$	Option byte 28 to 25
FF30_0084$_H$	Option byte 24 to 21
FF30_0080$_H$	Option byte 20 to 17
FF30_007C$_H$	Option byte 16 to 13
FF30_0078$_H$	Option byte 12 to 9
FF30_0074$_H$	Option byte 8 to 5
FF30_0070$_H$	Option byte 4 to 1

Table 6.7 lists the security setting data when various security functions are enabled.

Table 6.7 List of Security Setting Data

Security Functions	Security Setting Data
ID authentication enabled in serial programming mode	FFFF FFFF FFFF FFFF FFFF FFFF 1EFF FFFF$_H$
Serial programmer connection disabled	FFFF FFFF FFFF FFFF FFFF FFFF F7FF FFFF$_H$
Block erasure command disabled	FFFF FFFF FFFF FFFF FFFF FFFF DFFF FFFF$_H$
Programming command disabled	FFFF FFFF FFFF FFFF FFFF FFFF BFFF FFFF$_H$
Read command disabled	FFFF FFFF FFFF FFFF FFFF FFFF 7FFF FFFF$_H$

6.3.18 Reading the Configuration Setting Area

When reading the configuration setting area for checking the write values by using configuration setting command, set the FCUFSEL bit in the FCUFAREA register to 1. When the FCUFSEL bit is set to 1, the user area cannot be read. Software which reads the configuration setting area needs to be executed from the on-chip RAM. For the address map in the configuration setting area, see **Table 6.5**, **Table 6.6** in **Section 6.3.17, Configuration Programming Command**. Configuration setting "ID for authentication" storage area can be read only after SELF ID authentication is complete.

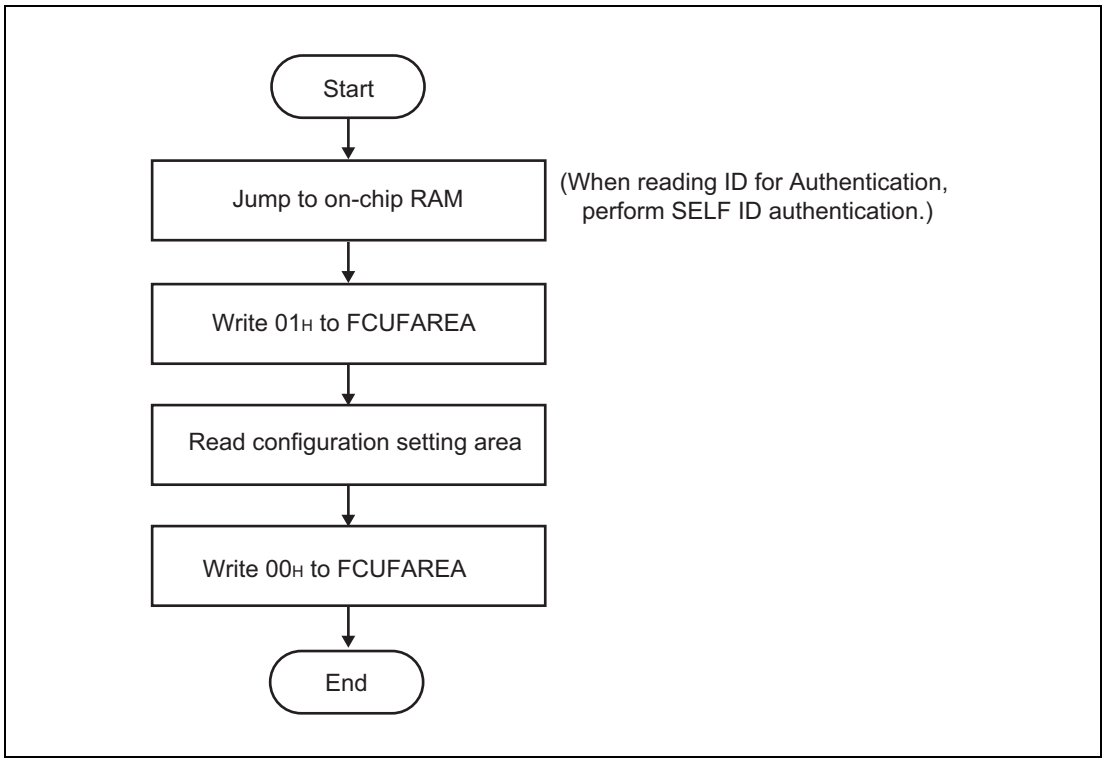


Figure 6.25 Flow of Reading Configuration Setting Area

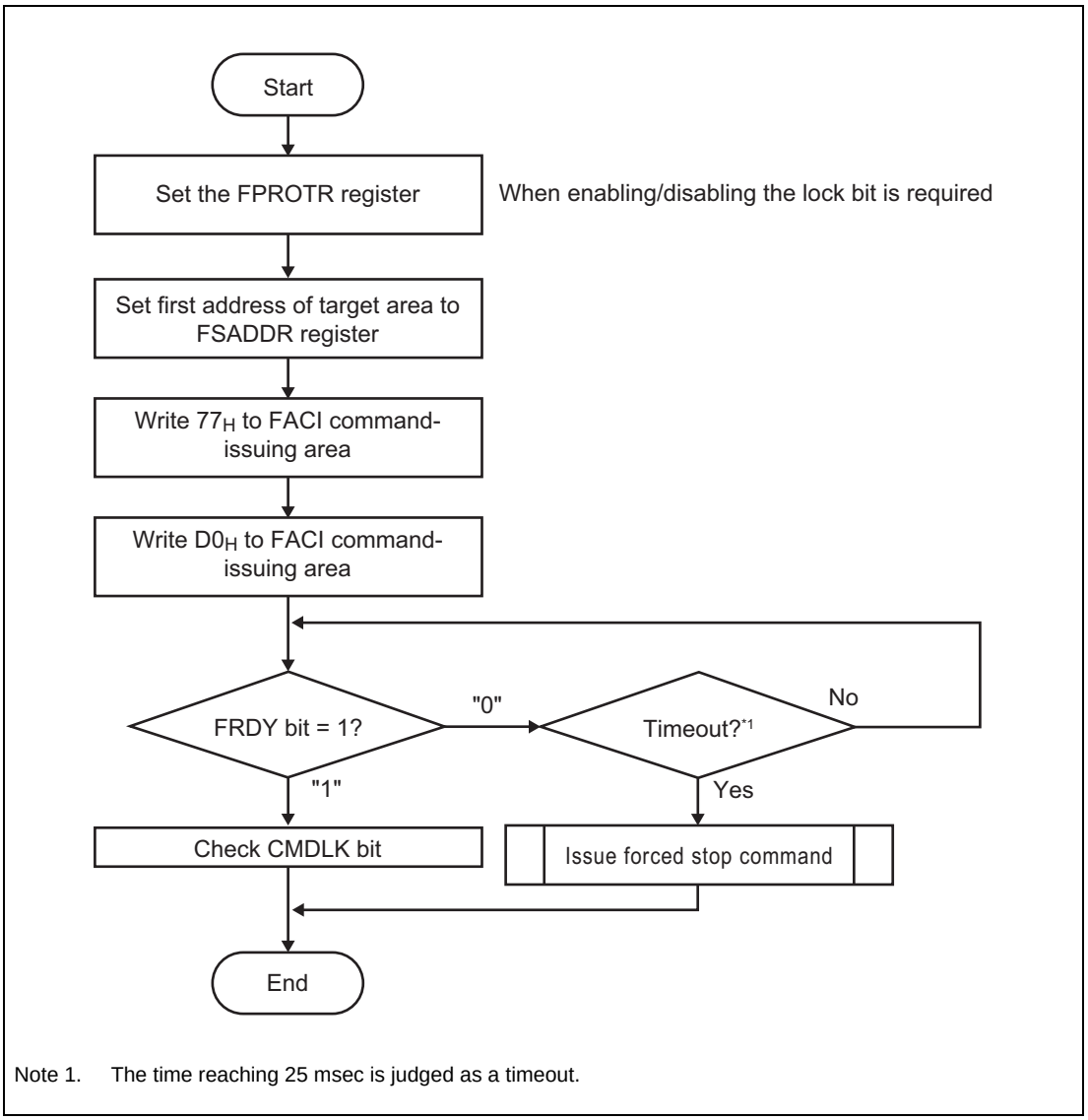
6.3.19 Lock Bit Programming Command

The lock bit programming command is used for programming lock bit. For the erasure of lock bits, use the block erasure command. (Refer to **Section 6.3.11, Block Erasure command**.)

Before issuing a lock bit programming command, set first address of target area to FSADRR register.

Writing 77_H and $D0_H$ to FACI command-issuing area starts the lock bit programming command processing.

Set the FPROTR register before issuing the lock bit programming command. To set the FPROTR register is required to switch enabling/disabling the lock bit.



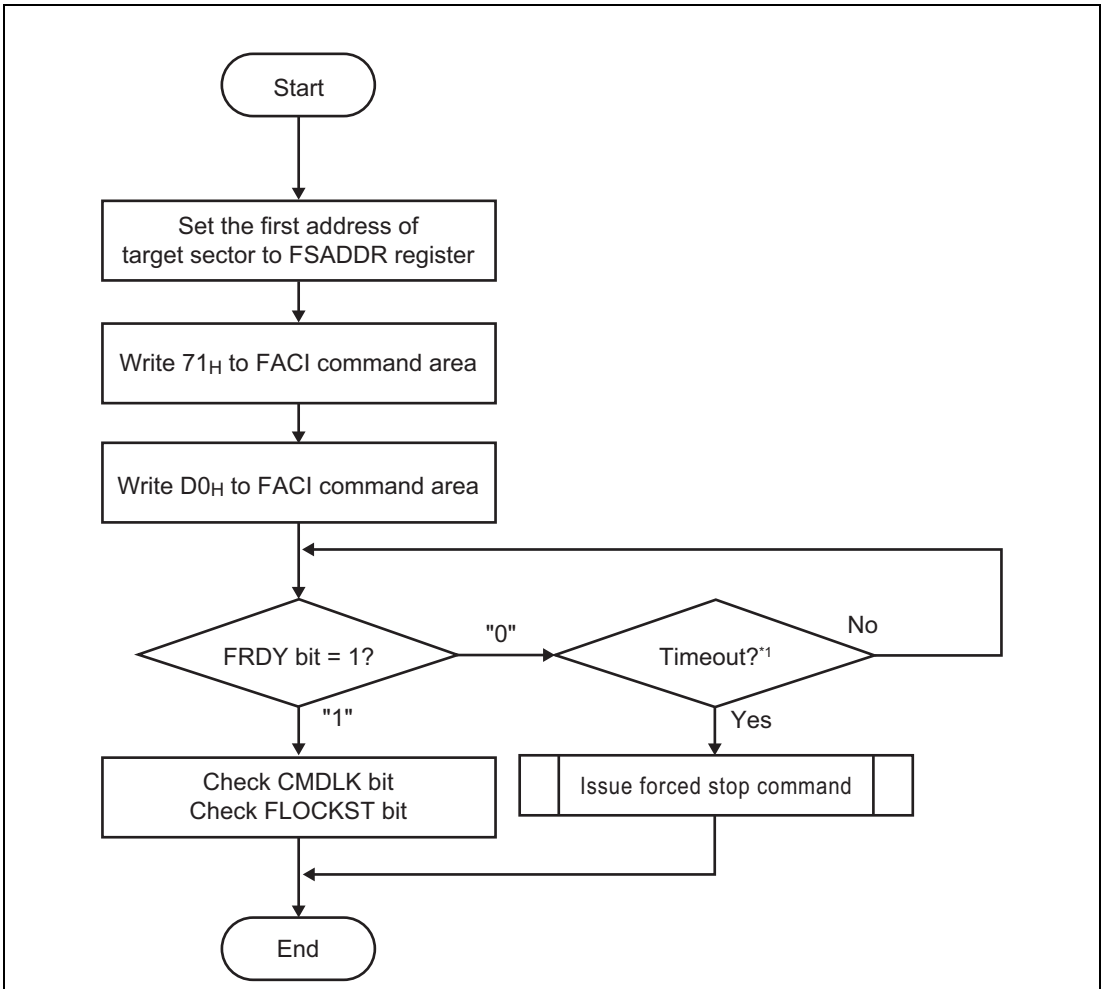
Note 1. The time reaching 25 msec is judged as a timeout.

Figure 6.26 Lock Bit Programming Command Usage

6.3.20 Lock Bit Reading Command

The lock bit reading command is used for read lock bit.

Before issuing a lock bit reading command, set first address of target area to FSADRR register. Writing 71_H and $D0_H$ to FACI command-issuing area starts lock bit reading command processing. Completion of command processing can be confirmed by FRDY bit of FSTATR register. After lock bit reading command processing is successfully completed, the result of lock bit reading is stored in FLOCKST register.

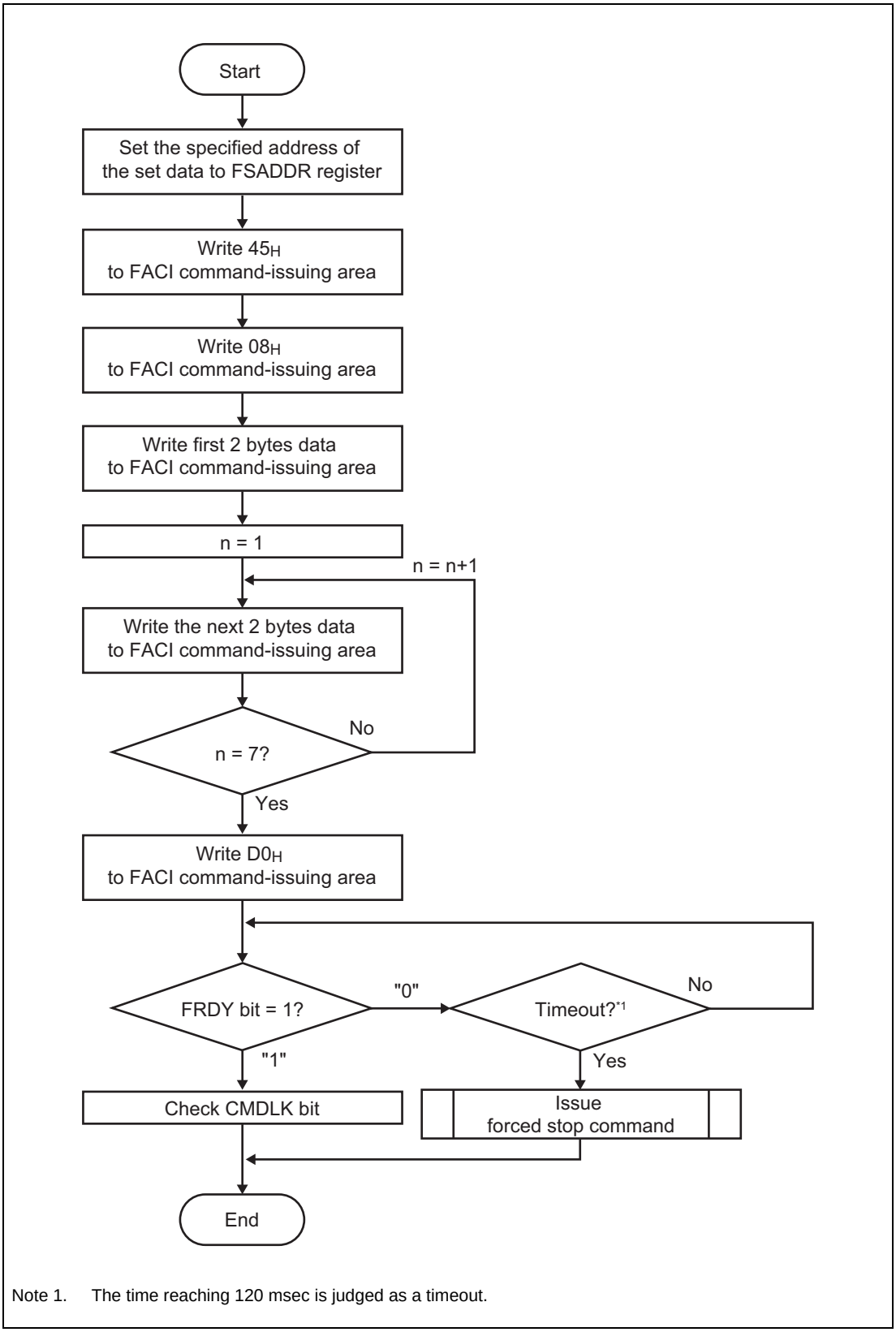


Note 1. The time reaching 10 µsec is judged as a timeout.

Figure 6.27 Lock Bit Reading Command Usage

6.3.21 OTP Setting Command

The OTP setting command is used to set OTP. Before issuing an OTP setting command, set the specified address of the set data (shown in **Table 6.8**) to FSADRR register. Writing $D0_H$ to the FACI command-issuing area at the final access of the FACI command issue starts the OTP setting command processing.



Note 1. The time reaching 120 msec is judged as a timeout.

Figure 6.28 OTP Setting Command Usage

Figure 6.29 shows the relationship between the user area blocks and OTP setting flags. OTP setting flags are allocated to each block of code flash memory. Products with 4 Mbytes of code flash memory have 8 Kbytes × 8 blocks + 32 Kbytes ×126 blocks and OTP flags OTPF0 to OTPF133 whereas the

products with 2 Mbytes of code flash memory have 8 Kbytes × 8 blocks + 32 Kbytes × 56 blocks and OTP flags OTPF0 to OTPF63.

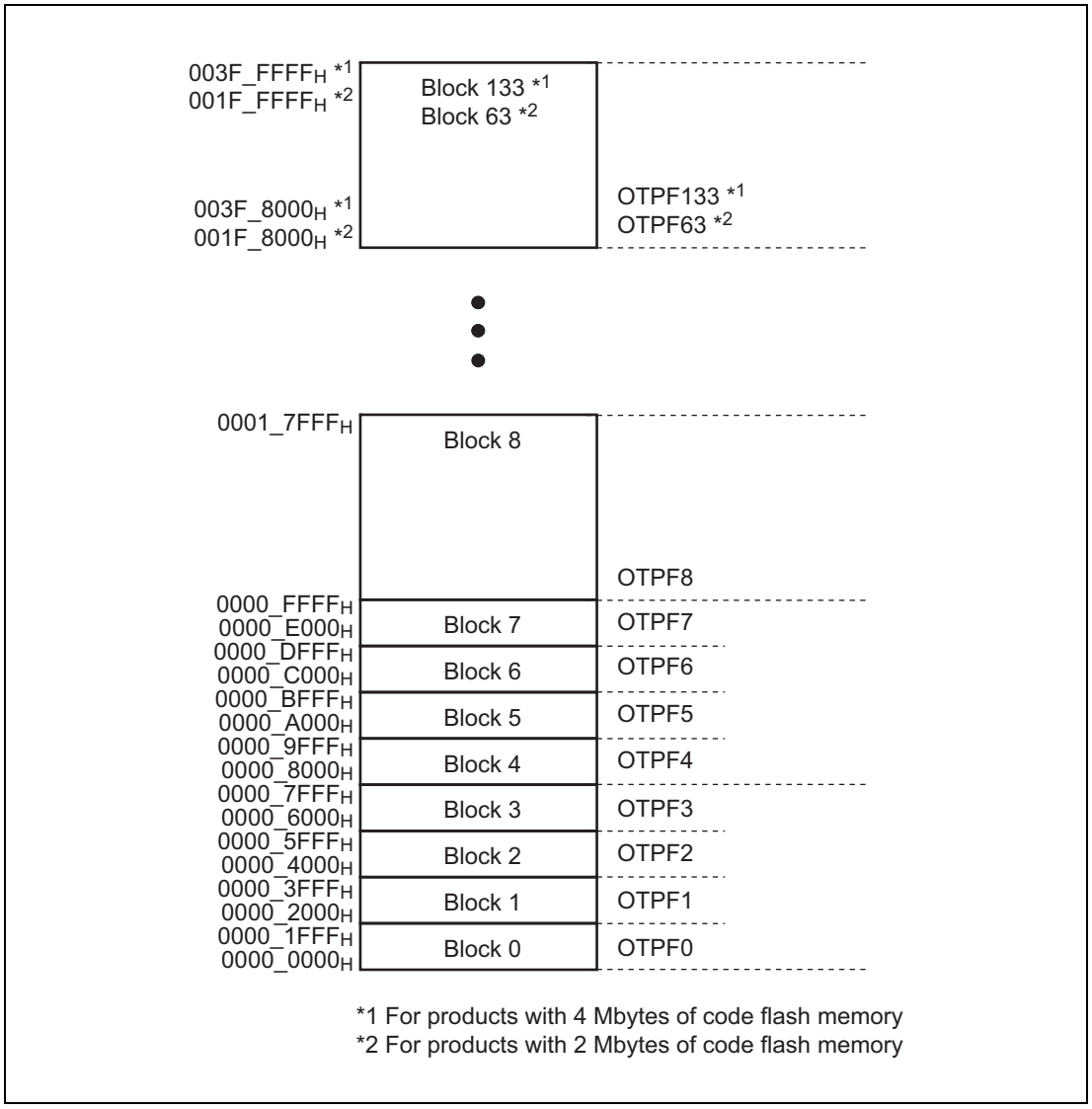


Figure 6.29 Relationship between User Area Blocks and OTP Setting Flags

Table 6.8 shows the addresses to be used by the OTP setting command. OTP is set for blocks corresponding to flags with the value 0. A flag with the setting 0 cannot be set to 1.

Table 6.8 Addresses to be Used in OTP Setting Command

Address	
FF38_0090$_H$	OTP flag for user boot area (bit0)
FF38_0080$_H$	Reserve*1
FF38_0070$_H$	Reserve*1
FF38_0060$_H$	Reserve*1
FF38_0050$_H$	For products with 4 Mbytes of code flash memory*2: Reserved *1 (bit127 to 6), OTPF133 (bit5) to OTPF128 (bit0) For products with 2 Mbytes of code flash memory *2: Reserved*1
FF38_0040$_H$	For products with 4 Mbytes of code flash memory*2: OTPF127 (bit127) to OTPF0 (bit0) For products with 2 Mbytes of code flash memory*2: Reserved*1(bit127 to 64), OTPF63 (bit63) to OTPF0 (bit0)

Note 1. Do not set 0 in the reserved area. The behavior and performance are not guaranteed when 0 is set.

Note 2. Capacities of the code flash memory and data flash memory vary from product to product. See the flash memory section in the user's manual of the applicable product.

6.3.22 Reading the OTP Setting Area

When reading the OTP setting area for checking the write value by using the OTP setting command, set the FCUFSEL bit in the FCUFAREA register to 1. When the FCUFSEL bit is set to 1, the user area cannot be read. Software which reads the OTP setting area needs to be executed from the on-chip RAM. For the address map in the OTP setting area, see **Table 6.8**, **Section 6.3.21, OTP Setting Command**.

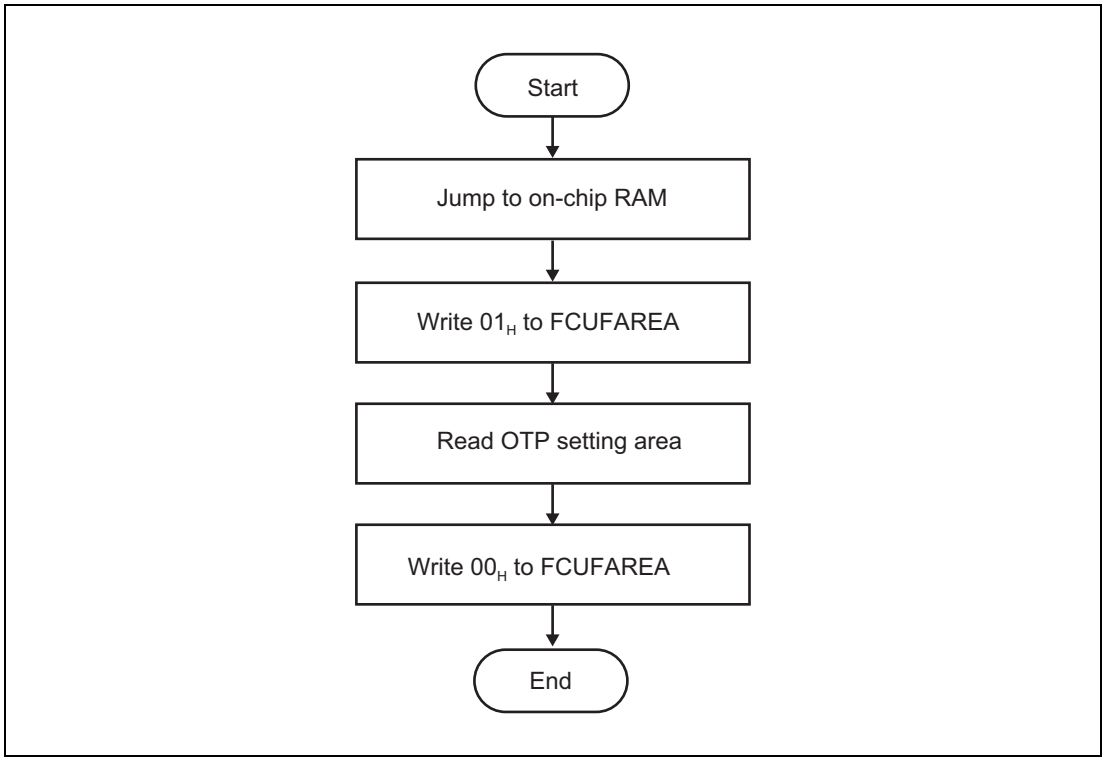


Figure 6.30 Flow of Reading OTP Setting Area

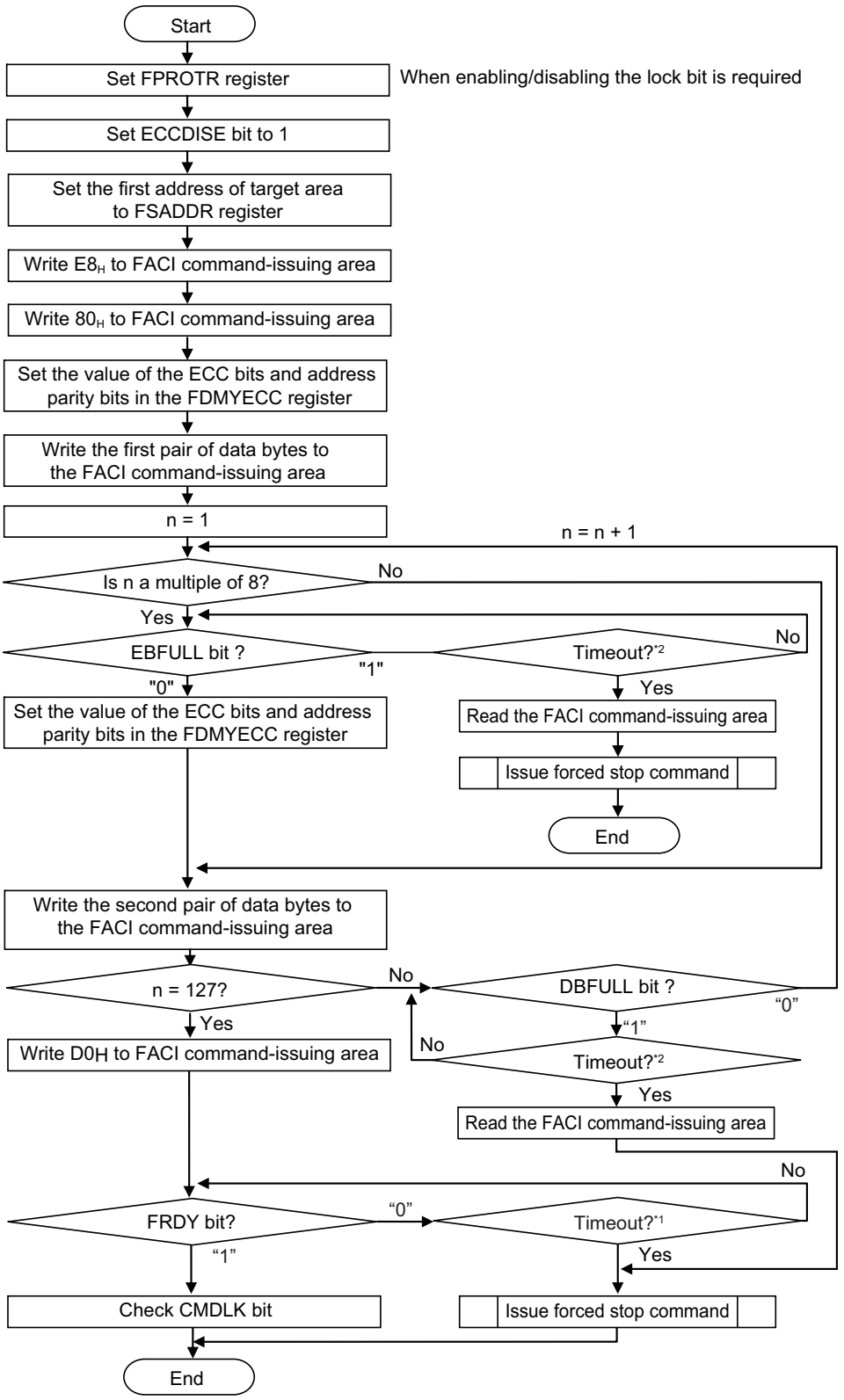
6.3.23 Injecting ECC Errors for the Flash Memory

Any value of the ECC bits and address parity bits in the FDMYECC register can be written to the flash memory by using a programming command. The function of injecting ECC error to data area can be used only by a programming command in 4 byte units.

Before writing the value set in the FDMYECC register to the flash memory, set the ECCDISE bit in the FECCTMD register to 1. In addition, set the values for the ECC bits and address parity bits in the

FDMYECC register before writing the data to the FACI command-issuing area. In the case of the code flash memory, the unit (256 bytes) for writing in response to the programming command differs from the unit (16 bytes) for which the ECC bits and address parity bits are to be added for the data. Therefore, every time 16 bytes of data are written to the FACI command-issuing area, change the setting in the FDMYECC register. In the case of the data flash memory, since the unit (4 bytes) for writing by the programming command is the same as that for the unit (4 bytes) of data for which the ECC bits are to be added, only change the setting in the FDMYECC register once before issuing the programming command.

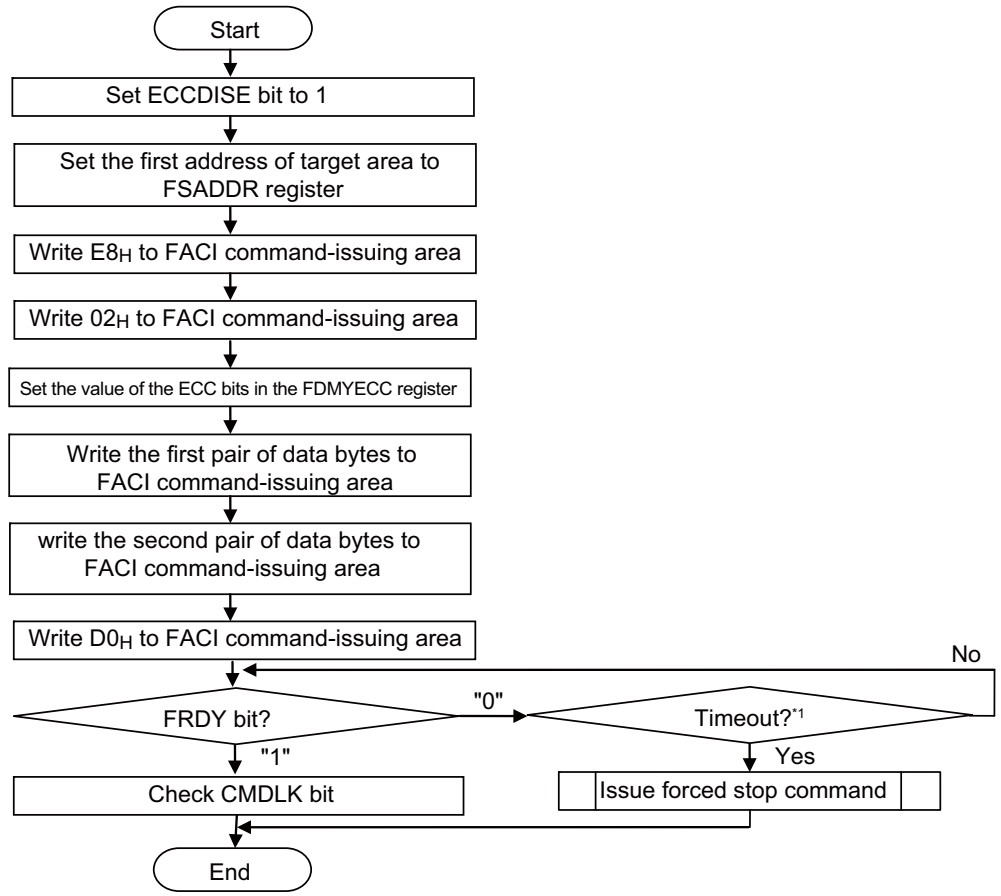
Issuing the command for writing to the FDMYECC register while the EBFULL bit in the FSTATR register is 1 may lead to a wait being generated on the peripheral bus, which will affect performance in communication with other peripheral IP modules. To avoid the generation of such a wait, write to the FDMYECC register while the EBFULL bit in the FSTATR register is 0.



Note 1. The time of 256-byte programming (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times is judged as a timeout.

Note 2. The time reaching 5 µs is judged as a timeout.

Figure 6.31 Injecting an ECC Error for the Code Flash Memory



Note 1. The time of 4-byte programming (the maximum time: see **Section 10, Electrical Characteristics**) reaching 1.1 times is judged as a timeout.

Figure 6.32 Injecting an ECC Error for the Data Flash Memory

Section 7 Security Function

7.1 FACI Command Protection by ID

In code flash programming/erasure mode, FACI commands can be used after security is released by the ID authentication. When the FACI command is issued while the IDST bit in the SELFIDST register is set to 1 (security lock state), the flash sequencer enters the command lock state.

As for the security releasing method by the ID authentication, see **6.3.7, ID Authentication**.

In data flash programming/erasure mode, FACI commands can be used regardless of the IDST bit setting.

7.2 OTP for Code Flash Memory

OTP can be set independently for each block in the code flash memory. Once an OTP is set, it cannot be canceled. If a programming, block erasure, or lock bit programming command is issued to an OTP setting block, the flash sequencer enters the command lock state.

Section 8 Protection Function

8.1 Software Protection

Software protection function disables flash sequencer command operation according to register settings or lock bit settings. If an attempt is made to issue flash sequencer command against software protection, flash sequencer enters command lock state.

8.1.1 Protection by FENTRYR

When FENTRYR register is set to 0000_H, flash sequencer is set to read mode. In read mode, FACI commands cannot be accepted. If an attempt is made to issue FACI command in read mode, flash sequencer enters command lock state.

8.1.2 Lock Bit

Each block in user area has lock bits. When the FPROTCN bit in the FPROTR register is 0, programming/erasing the block where the corresponding lock bit is 0 is disabled. To program/erase the block where the corresponding lock bit is 0 is disabled, set the FPROTCN bit to 1. If an attempt is made to issue a programming/block erasure/lock bit programming command against protection by lock bits, the flash sequencer enters the command lock state.

8.2 Error Protection

Error protection function detects an illegal FACI command issued, an illegal access, or a flash sequencer malfunction, and disables FACI command acceptance (command lock state). While flash sequencer is in the command lock state, flash memory cannot be programmed or erased. To cancel command lock state, issue status clearing or forced stop command while the CFAE bit and DFAE bit in the FASTAT register are 0. Status clearing command can be used only when FRDY bit is 1. Forced stop command can be used regardless of FRDY bit value. While the CMDLKIE bit in FAEINT register is 1, a flash access error (FLERR) interrupt is generated if flash sequencer enters command lock state (the CMDLK bit of the FASTAT register is 1).

If flash sequencer enters command lock state during programming or erasure processing by the command other than programming/erasure suspension, the flash sequencer continues programming or erasure processing, In this state, programming or erasure processing cannot be suspended by the programming/erasure suspension command. If a command is issued in command lock state, ILGLERR bit becomes 1 and the other bits retain the values set due to the previous error detection.

Table 8.1 shows error protection types and status bit values after error detection.

Table 8.1 Error Protection Type (1/2)

Error Type	Description	OTPDTCT	ILGLERR	ERSERR	PRGERR	FHVEERR	CFGDTCT	TBLDTCT	FRDTCT	CFAE	DFAE
FENTRYR setting error	The value set in FENTRYR is not 0000_H, 0001_H, or 0080_H.	0	1	0	0	0	0	0	0	0	0
	The FENTRYR setting for resuming operation does not match that for suspending operation.	0	1	0	0	0	0	0	0	0	0
Illegal command error	An undefined code has been written in the first access of FACI command.	0	1	0	0	0	0	0	0	0	0
	The value specified in the last access of the multiple-access FACI command is not $D0_H$ (except for DMA programming).	0	1	0	0	0	0	0	0	0	0
	The value (N) specified in the second write access of FACI command in the programming, DMA programming, config programming, or OTP setting command is wrong. (An odd value is not available in DMA programming.)	0	1	0	0	0	0	0	0	0	0
	Blank checking command has been issued with inconsistent BCDIR, FSADDR, and FEADDR settings. (See **Section 4.4, FEADDR — FACI Command End Address Register**.)	0	1	0	0	0	0	0	0	0	0
	FACI command has been issued against FACI command not acceptable mode. (See **Table 6.3**.)	0	1	0	0	0	0	0	0	0	0
	FACI command has been issued when command acceptance conditions are not satisfied. (See **Table 6.4**)	0/1	1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
Erasure error	An error has occurred during flash memory erasure.	0	0	1	0	0	0	0	0	0	0
	Block erasure command has been issued against lock bit protection.	0	0	1	0	0	0	0	0	0	0
Programming error	An error has occurred during flash memory program.	0	0	0	1	0	0	0	0	0	0
	Programming or lock bit programming command has been issued against lock bit protection.	0	0	0	1	0	0	0	0	0	0
FCURAM ECC error	A 2-bit error has been detected when FCURAM is read.	0	0	0	0	0	0	0	1	0	0
Code flash access error	FACI command has been issued to reserved user area in code flash programming/erasure mode. (See **Section 4.1, FASTAT — Flash Access Status Register**.)	0	1	0	0	0	0	0	0	1	0
Data flash access error	FACI command has been issued to reserved data area in data flash programming/erasure mode. (See **Section 4.1, FASTAT — Flash Access Status Register**.)	0	1	0	0	0	0	0	0	0	1
	Config programming command has been issued to reserved area. (See **Section 4.1, FASTAT — Flash Access Status Register**.)	0	1	0	0	0	0	0	0	0	1
	OTP setting command has been issued to reserved area. (See **Section 4.1, FASTAT — Flash Access Status Register**.)	0	1	0	0	0	0	0	0	0	1

Table 8.1 Error Protection Type (2/2)

Error Type	Description	OTPDTCT	ILGLERR	ERSERR	PRGERR	FHVEERR	CFGDTCT	TBLDTCT	FRDTCT	CFAE	DFAE
Security	Programming, block erasure, or lock bit programming command has been issued against OTP setting.	0	1	0	0	0	0	0	0	0	0
	FACI command has been issued in code flash programming/erasure mode against security not-released state by ID authentication.	0	1	0	0	0	0	0	0	0	0
Other	FACI command-issuing area has been accessed in read mode.	0	1	0	0	0	0	0	0	0	0
	FACI command-issuing area has been read in code flash programming/erasure mode or data flash programming/erasure mode.	0	1	0	0	0	0	0	0	0	0
OTP Setting ECC error	A 2-bit error has been detected when OTP setting is read.	1	0	0	0	0	0	0	0	0	0
FHVE setting error	FHVE15/FHVE3 of flash memory has changed to 0 while command processing is provided by the flash sequencer.	0	0	0/1	0/1	1	0	0	0	0	0
Config Programming ECC error	2-bit error has been detected when config programming value is read.	0	0	0	0	0	1	0	0	0	0
Overwrite parameter ECC error	2-bit error has been detected when overwrite parameter table is read.	0	0	0	0	0	0	1	0	0	0

8.3 Boot Program Protection

8.3.1 User Boot Protection

The user boot area can be overwritten by the serial programming. Since this area is usually write-protected for the self-programming, it can be used to store programs such as a boot program safely.

8.4 Blank Checking of Code Flash Memory

Note that an ECC error is detected and an exception is generated when code flash memory to which no data have been written after erasure (i.e. in the non-programmed state) is read. When an ECC error occurs, the data values cannot be guaranteed. To confirm the non-programmed state of the memory, confirm that code flash memory data, ECC bits, and address parity bits are all 1. For usage of the ECC function of the code flash memory, see *Section 30, Safety,* in the user's manual.

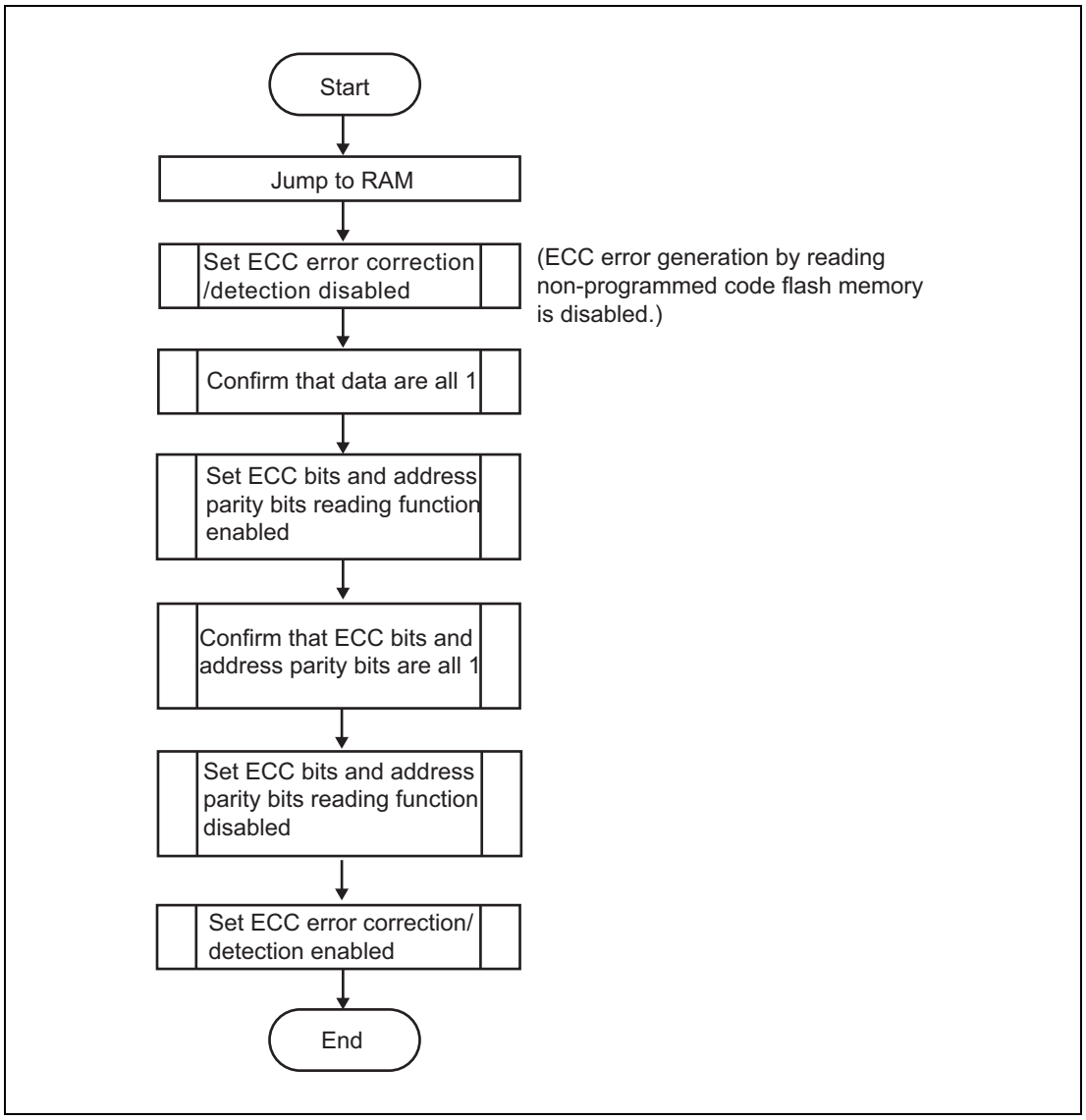


Figure 8.1 Blank Checking of Code Flash Memory

Section 9 Usage Notes

(1) Reading areas where programming or erasure was interrupted

When programming or erasure of an area of flash memory is interrupted, the data stored in the area become undefined. To avoid undefined data that are read out becoming the source of faulty operation, take care not to fetch instructions or read data from areas where programming or erasure was interrupted.

(2) Prohibition of additional writing

Writing to a given area twice is not possible. If you want to overwrite data in an area of flash memory after writing to the area has been completed, erase the area first.

(3) Resets during programming and erasure

In the case of an external reset during programming and erasure, wait for at least width of reset pulse more than the min value once the operating voltage is within the range stipulated in the electrical characteristics after assertion of the reset signal before releasing the device from the reset state.

(4) Allocation of vectors for interrupts and other exceptions during programming and erasure

Generation of an interrupt or other exception during programming or erasure may lead to fetching of the vector from the code flash memory. If this does not satisfy the conditions for using background operation, set the address for vector fetching to an address that is not in the code flash memory.

For how to change the address for vector fetching, see *Section 3 CPU System* and *Section 6 Interrupts* in the *User's Manual: Hardware*.

(5) Abnormal termination of programming and erasure

Even if programming/erasure ends abnormally due to the generation of an external reset or power shutoff, the programming/erasure state of the flash memory with undefined data cannot be verified or checked. For the area where programming/erasure ends abnormally, the blank check function cannot judge whether the area is erased successfully or not. Erase the area again to prove that the corresponding area is completely erased before using.

If programming and erasure of code flash memory are not completed normally, the lock bit for the target area may be enabled (locked). In such cases, erase the block to erase the lock bit while the lock bit is in the disabled state (the area is not locked).

(6) Items prohibited during programming and erasure

Do not perform the following operations while the flash memory is programmed or erased.

- Set the operating voltage from the power supply outside the allowed range.
- Update the FHVE15 and FHVE3 values.
- Change the operating frequency of the peripheral clock.

(7) Update of the FCUFAREA register

When the code flash area is switched by setting the FCUFAREA register, previous and subsequent processing and switching of the area must be synchronized.

The flow of synchronization when updating the FCUFAREA register is described in the following three cases.

(1) Processing for synchronization when updating the FCUFAREA register

After an update of the register, dummy-read the FCUFAREA register and execute the SYNCP instruction before a read instruction for the code flash memory or FCU firmware storage area (LD.W, etc.) to wait for the update of the FCUFAREA register.

(2) Processing for synchronization before switching to the code flash user area or user boot area

For switching to the code flash area following the completion of reading the FCU firmware storage area, execute the SYNCP instruction after the last read instruction (LD.W, etc.) for the FCU firmware storage area, and then follow step (1) above to change the value of the FCUFAREA register.

(3) Processing for synchronization after switching to the code flash user area or user boot area

To prevent execution of code flash codes that have been read before switching, dummy-read the FCUFAREA register and execute the SYNCP and SYNCI instructions after an update of the FCUFAREA register before executing the code flash instruction.
After the code flash area has been switched, clear the instruction cache and data buffer.

(8) Point for Caution on Selecting the FCU Firmware Storage Area

For transfer of the FCU firmware or for reading of the configuration setting area or OTP setting area, the FCU firmware storage area must be selected by setting the FCUFAREA register. When the FCU firmware storage area is selected, place the exception handler vector address for the CPU in the on-chip RAM to avoid any access to the FCU firmware storage area due to the generation of an interrupt. For how to change the vector address of the exception handler, see *Section 3 CPU System* and *Section 6 Interrupt* in the *User's Manual: Hardware*.

Note 1. This is not applicable when transferring the FCU firmware from FCU firmware area 2 (see **Figure 6.6, Transfer Flow of FCU Firmware (2)**) because the FCUFAREA register is not to be set.

(9) Securing Coherency after Rewriting Code Flash Memory

After rewriting a code flash area, if a code flash instruction is executed, secure coherency by clearing the instruction cache, and clearing the data buffer (see Usage Notes in *Section 3 CPU System* in the *User's Manual: Hardware*).

Section 10 Electrical Characteristics

This section explains the electrical characteristics when the hardware interface is used with the self-programming described in this manual.

Note that these electrical characteristics differ from those of when the serial programming is used.

10.1 Code Flash Characteristics

Table 10.1 Code Flash Basic Characteristics

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Programming endurance*1	CWRT	Retained for 20 years *2	1000	—	—	Times
Temperature range of programming	TPRG	Tj	-40	—	+150	°C
Temperature range of reading	TREAD	Tj	-40	—	+150	°C

Note 1. Programming endurance is defined as the number of times each block is erased. Where programming endurance is n times (n = 1000 in this case), each block is erasable n times. For example, given a memory device that has 32-Kbyte erasure blocks, programming in the address range of each 256-byte programming block (128 programming operations, one for each block) in an erasure block and then erasing the block counts as one time in terms of programming endurance. However, programming of a given address range more than once after erasure is not possible (overwriting after programming is prohibited).

Note 2. The retained period when the average Ta is 85°C. This retained period is from when the erasure of the code flash memory has been normally completed.

Table 10.2 Code Flash Programming Characteristics

Conditions: EVCC = 4.5V to 5.5V, TTLVCC = 4.5V to 5.5V or 3.0V to 3.6V, A0VCC,A1VCC = 4.5V to 5.5V,
A0VREFH = 4.5V to A0VCC, A1VREFH = 4.5V to A1VCC, ADSVCC = 4.5V to 5.5V,
ADSVREFH = 4.5V to ADSVCC, PLLVCC = 3.0V to 3.6V, SYSVCC = 3.0V to 3.6V,
VCC = 3.0V to 3.6V, LVDVCC = 3.0V to 3.6V, VDD = 1.15V to 1.35V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0V
Tj = -40°C to 150°C

Item	Condition	Block size	Min.	Typ.	Max.	Unit
Programming time	Programming endurance < 100 times	256 B	—	0.4	6	ms
		8 KB	—	13	80	ms
		32 KB	—	52	320	ms
		128 KB	—	208	1280	ms
	Programming endurance ≥ 100 times	256 B	—	0.5	7.2	ms
		8 KB	—	16	96	ms
		32 KB	—	64	384	ms
		128 KB	—	256	1536	ms
Erasing time	Programming endurance < 100 times	8 KB	—	39	120	ms
		32 KB	—	141	480	ms
		128 KB	—	564	1750	ms
	Programming endurance ≥ 100 times	8 KB	—	47	144	ms
		32 KB	—	169	576	ms
		128 KB	—	676	2100	ms

Table 10.3 Suspension/Resumption/Forced Stop

Conditions: EVCC = 4.5V to 5.5V, TTLVCC = 4.5V to 5.5V or 3.0V to 3.6V, A0VCC,A1VCC = 4.5V to 5.5V,
A0VREFH = 4.5V to A0VCC, A1VREFH = 4.5V to A1VCC, ADSVCC = 4.5V to 5.5V,
ADSVREFH = 4.5V to ADSVCC, PLLVCC = 3.0V to 3.6V, SYSVCC = 3.0V to 3.6V,
VCC = 3.0V to 3.6V, LVDVCC = 3.0V to 3.6V, VDD = 1.15V to 1.35V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0V
Tj = -40°C to 150°C

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Suspend latency during programming	t_{SPD}	—	—	—	120	µs
Programming resume time*1	t_{RPT}	—	—	—	50	µs
Suspend latency during erasing	t_{SESD1}	Priority on suspension The 1st suspend for the same pulse	—	—	120	µs
	t_{SESD2}	Priority on suspension The 2nd suspend for the same pulse	—	—	1.7	ms
	t_{SEED}	Priority on erasure	—	—	1.7	ms
Erasing resume time*1	t_{REST1}	Priority on suspension Resume after the 1st suspend for the same pulse	—	—	1.7	ms
	t_{REST2}	Priority on suspension Resume after the 2nd suspend for the same pulse	—	—	80	µs
	t_{REET}	Priority on erasure	—	—	80	µs
Forced stop command latency	t_{FD}	—	—	—	20	µs

Note 1. The time taken for resumption includes an overhead for the resumption of programming or erasure.
In suspension-priority mode, a time for reapplication of the erasing pulse that was cut off at the time of suspension is also required.
Resume time is defined as time added by programming or erasing due to those sources.

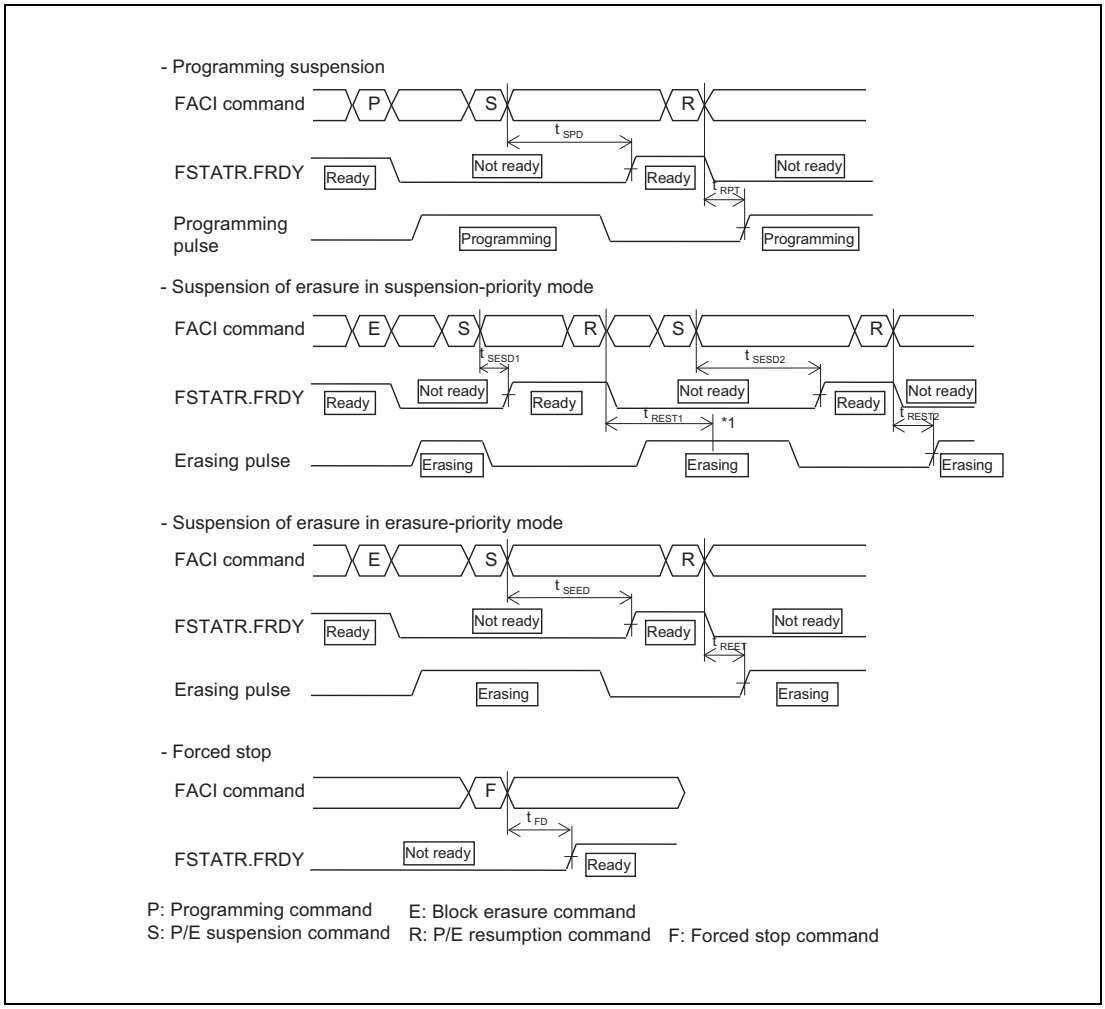


Figure 10.1 Timing of Suspension/Resumption/Forced Stop

Note 1. Time for resumption includes time for reapplying the erasing pulse (up to one full pulse) that was cut off at the time of suspension.

10.2 Data Flash Characteristics

Table 10.4 Data Flash Basic Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Programming endurance*1	CWRT	Retained for 20 years*2	125000	—	—	Times
		Retained for 3 years*2	250000	—	—	Times
Temperature range of programming	TPRG	Tj	-40	—	+150	°C
Temperature range of reading	TREAD	Tj	-40	—	+150	°C

Note 1. Programming endurance is defined as the number of times each block is erased. Where programming endurance is n times (n = 125000 in this case), each block is erasable n times. For example, given a memory device that has 64-byte erasure blocks, programming in the address range of each 4-byte programming block (16 programming operations, one for each block) in an erasure block and then erasing the block counts as one time in terms of programming endurance. However, programming of a given address range more than once after erasure is not possible (overwriting after programming is prohibited).

Note 2. The retained period when the average Ta is 85°C. This retained period is from when the erasure of the data flash memory has been normally completed.

Table 10.5 Data Flash Programming Characteristics

Conditions: EVCC = 4.5V to 5.5V, TTLVCC = 4.5V to 5.5V or 3.0V to 3.6V, A0VCC,A1VCC = 4.5V to 5.5V,
A0VREFH = 4.5V to A0VCC, A1VREFH = 4.5V to A1VCC, ADSVCC = 4.5V to 5.5V,
ADSVREFH = 4.5V to ADSVCC, PLLVCC = 3.0V to 3.6V, SYSVCC = 3.0V to 3.6V,
VCC = 3.0V to 3.6V, LVDVCC = 3.0V to 3.6V, VDD = 1.15V to 1.35V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0V
Tj = -40°C to 150°C

Item	Block size	Min.	Typ.	Max.	Unit
Programming time	4 B	—	0.16	1.7	ms
	64 B	—	2.6	13	ms
Erasing time	64 B	—	1.7	10	ms
Blank check time*1	4 B	—	—	30	µs
	64 B	—	—	100	µs
	2 KB	—	—	2.2	ms

Note 1. When the area size for blank checking is 2 Kbytes or larger, the time will be proportional to that of 2-Kbyte blank checking. When the area size for blank checking is 64 bytes or larger and less than 2 Kbytes, the time will be proportional to that of 64-byte blank checking.

Table 10.6 Suspension/Resumption/Forced Stop

Conditions: EVCC = 4.5V to 5.5V, TTLVCC = 4.5V to 5.5V or 3.0V to 3.6V, A0VCC,A1VCC = 4.5V to 5.5V,
A0VREFH = 4.5V to A0VCC, A1VREFH = 4.5V to A1VCC, ADSVCC = 4.5V to 5.5V,
ADSVREFH = 4.5V to ADSVCC, PLLVCC = 3.0V to 3.6V, SYSVCC = 3.0V to 3.6V,
VCC = 3.0V to 3.6V, LVDVCC = 3.0V to 3.6V, VDD = 1.15V to 1.35V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0V
Tj = -40°C to 150°C

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Suspension latency during programming	t_{SPD}	—	—	—	120	µs
Programming resume time*1	t_{RPT}	—	—	—	50	µs
Suspension latency during erasure	t_{SESD1}	Priority on suspension The 1st Suspension for the same pulse	—	—	120	µs
	t_{SESD2}	Priority on suspension The 2nd suspension for the same pulse	—	—	300	µs
	t_{SEED}	Priority on erase	—	—	300	µs
Erasing resume time*1	t_{REST1}	Priority on suspension Resumption after the 1st suspension for the same pulse	—	—	300	µs
	t_{REST2}	Priority on suspension Resumption after the 2nd suspension for the same pulse	—	—	70	µs
	t_{REET}	Priority on erasure	—	—	70	µs
Forced stop command latency	t_{FD}	—	—	—	20	µs

Note 1. The time taken for resumption includes an overhead for the resumption of programming or erasure.
In suspension-priority mode, a time for reapplication of the erasing pulse that was cut off at the time of suspension is also required.
Resume time is defined as time added by programming or erasing due to those sources.

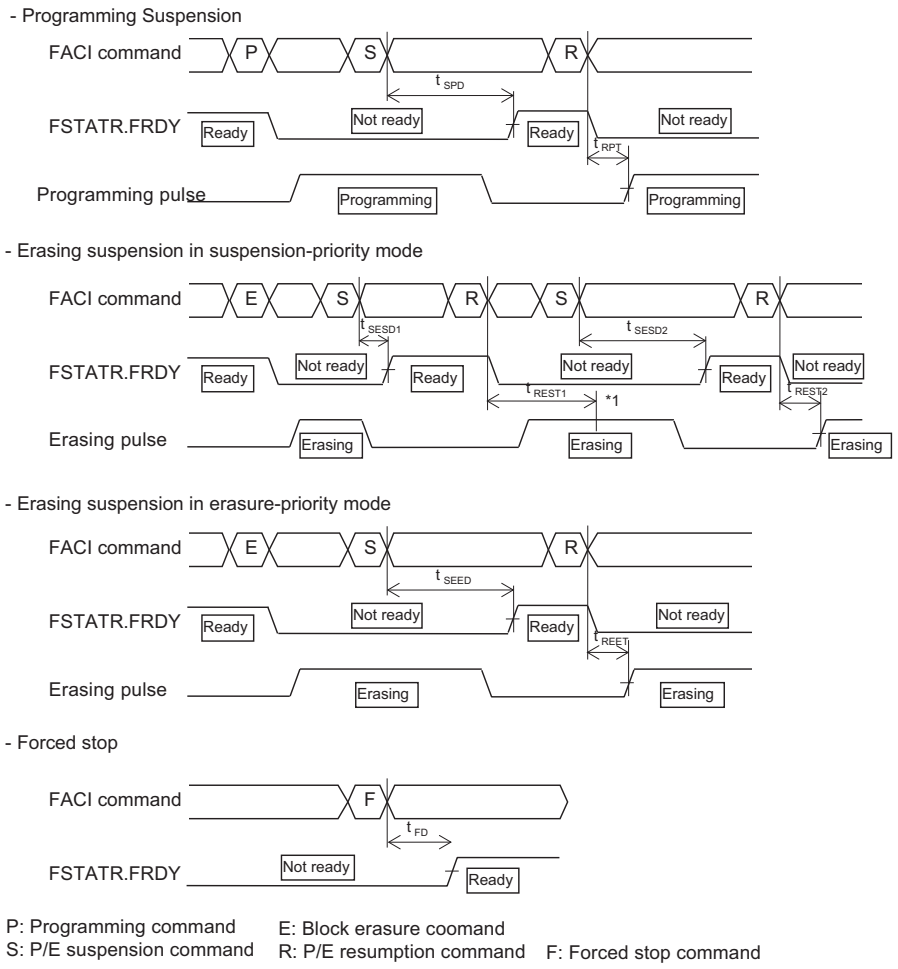


Figure 10.2 Timing of Suspension/Resumption/Forced Stop

Note 1. Time for resumption includes time for reapplying the erasing pulse (up to one full pulse) that was cut off at the time of suspension.

Appendix List of Flash Memory Related Registers

The following shows the registers related to the flash memory.

Table Appendix 1 List of Flash Memory Related Registers

Module Name	Register Name	Symbol	Initial Value	Address	Peripheral Group	Access Size
FACI	Flash access status register	FASTAT	00_H	FFA1 0010	3	8
FACI	Flash access error interrupt enable register	FAEINT	99_H	FFA1 0014	3	8
FACI	FACI command start address register	FSADDR	0000 0000_H	FFA1 0030	3	32
FACI	FACI command end address register	FEADDR	0000 0000_H	FFA1 0034	3	32
FACI	FCURAM enable register	FCURAME	0000_H	FFA1 0054	3	16
FACI	Flash status register	FSTATR	0000 8000_H	FFA1 0080	3	8, 16, 32
FACI	Flash programming/erasure mode entry register	FENTRYR	0000_H	FFA1 0084	3	16
FACI	Code flash protect register	FPROTR	0000_H	FFA1 0088	3	16
FACI	Flash sequencer set-up initialize register	FSUINITR	0000_H	FFA1 008C	3	16
FACI	Lock bit status register	FLKSTAT	00_H	FFA1 0090	3	8
FACI	FCURAM first error address register	FRFSTEADR	0000 0000_H	FFA1 0094	3	32
FACI	FACI reset transfer status register	FRTSTAT	$0X_H$	FFA1 0098	3	8
FACI	FACI reset transfer error interrupt enable register	FRTEINT	03_H	FFA1 009C	3	8
FACI	FACI command register	FCMDR	$FFFF_H$	FFA1 00A0	3	16
FACI	Flash programming/erasure status register	FPESTAT	0000_H	FFA1 00C0	3	16
FACI	Data flash blank check control register	FBCCNT	00_H	FFA1 00D0	3	8
FACI	Data flash blank check status register	FBCSTAT	00_H	FFA1 00D4	3	8
FACI	Data flash start address register	FPSADDR	0000 0000_H	FFA1 00D8	3	32
FACI	Flash sequencer process switch register	FCPSR	0000_H	FFA1 00E0	3	16
FACI	Flash sequencer process clock notification register	FPCKAR	0028_H	FFA1 00E4	3	16
FACI	Flash ECC encoder monitor register	FECCEMON	$FFFF_H$	FFA1 0100	3	16
FACI	Flash ECC test mode register	FECCTMD	0030_H	FFA1 0104	3	16
FACI	Flash dummy ECC register	FDMYECC	$FFFF_H$	FFA1 0108	3	16

Table Appendix 2 List of Product Information Related Registers

Module Name	Register Name	Symbol	Initial Value	Address	Peripheral Group	Access Size
FLASH	Product name storing register 1*1	PRDNAME1	XXXX $XXXX_H$	FFCD 00D0	3	32
FLASH	Product name storing register 2*1	PRDNAME2	XXXX $XXXX_H$	FFCD 00D4	3	32
FLASH	Product name storing register 3*1	PRDNAME3	XXXX $XXXX_H$	FFCD 00D8	3	32
FLASH	Product name storing register 4*1	PRDNAME4	XXXX $XXXX_H$	FFCD 00DC	3	32

Note 1. For details of the register, see the User's Manual: Hardware.

Table Appendix 3 List of Other Registers

Module Name	Register Name	Symbol	Initial Value	Address	Peripheral Group	Access Size
FLASH	Self-programming ID input register0	SELFID0	0000 0000_H	FFA0 8000	3	32
FLASH	Self-programming ID input register1	SELFID1	0000 0000_H	FFA0 8004	3	32
FLASH	Self-programming ID input register2	SELFID2	0000 0000_H	FFA0 8008	3	32
FLASH	Self-programming ID input register3	SELFID3	0000 0000_H	FFA0 800C	3	32
FLASH	Self-programming ID authentication status register	SELFIDST	*1	FFA0 8010	3	8, 16, 32
FACI	FCU Firmware Area Select Register	FCUFAREA	*3	FFC5 9008	0	8
FLASH	Data flash read cycle setting register*2	FRDCYCLD	$0F_H$	FFC5 9810	0	8
FLASH	FHVE15 control register	FHVE15	0000 0000_H	FFF8 A430	5	32
FLASH	FHVE3 control register	FHVE3	0000 0000_H	FFF8 2410	5	32

Note 1. The initial value of SELFIDST differs depending on ID code written to the special area in the flash memory. If all the ID code bits are 0, the initial value is 0000 0000_H. It is otherwise 0000 0001_H.

Note 2. For details of the register, see the User's Manual: Hardware.

Note 3. For start-up of user boot mode: 00_H
For start-up of serial programming mode: 01_H.

REVISION HISTORY	RH850/E1x-FCC2 User's Manual: Hardware Interface

Rev.	Date	Description	
		Page	Summary
0.10	Mar 16, 2015	—	First Edition issued
1.00	Jun 13, 2016	Section 2 Module Configuration	
		7	Description changed to "compares the ID transferred from the flash memory..."
		Section 3 Address Map	
		8	Table 3.1 Information on the Hardware Interface Area: "Peripheral Group" column added
		Section 4 Registers	
		All	"Value after reset:" on the hexadecimal notation side, changed to "Initial value:"
		10	Table 4.1 FASTAT Register Contents (1/2): Description of [Clearing Condition] changed in the Function column for Bit Name, CMDLK
		12	4.2 FAEINT — Flash Access Error Interrupt Enable Register: Description added
		14	Table 4.4 FEADDR Register Contents: "FACI Command End Address" deleted from the Function column
		15	4.5 FCURAME — FCURAM Enable Register: Note 1 corrected
		15	Table 4.5 FCURAME Register Contents: Description changed in the Function column for Bit Name, KEY
		17	Table 4.6 FSTATR Register Contents (2/5): Description changed in the Function column for Bit Name, FRDY
		17	Table 4.6 FSTATR Register Contents (2/5): Description changed in the Function column for Bit Name, ERSERR
		21	Section title changed to "4.7 FENTRYR — Flash Programming/Erasure Mode Entry Register"
		21	4.7 FENTRYR — Flash Programming/Erasure Mode Entry Register: Description corrected
		21	4.7 FENTRYR — Flash Programming/Erasure Mode Entry Register: Note 1 corrected
		21	4.7 FENTRYR — Flash Programming/Erasure Mode Entry Register: Note 4 added
		21	Table 4.7 FENTRYR Register Contents (1/2): Description changed in the Function column for Bit Name, FENTRYD
		22	Table 4.7 FENTRYR Register Contents (2/2): Description changed in the Function column for Bit Name, FENTRYC
		23	Section title changed to "4.8 FPROTR — Code Flash Protect Register"
		23	4.8 FPROTR — Code Flash Protect Register: Note 1 corrected
		23	Table 4.8 FPROTR Register Contents: Description changed in the Function column for Bit Name, FPROTCN
		24	4.9 FSUINITR — Flash Sequencer Set-up Initialize Register: Note 1 corrected
		25	Table 4.10 FLKSTAT Register Contents: Description changed in the Function column for Bit Name, FLOCKST
		28	4.13 FRTEINT — FACI Reset Transfer Error Interrupt Enable Register: Description added
		30	Section title changed to "4.15 FPESTAT — Flash Programming/Erasure Status Register"
		30	Table 4.16 FPESTAT Register Contents: Description changed in the Function column for Bit Name, PEERRST
		34	4.19 FCPSR — Flash Sequencer Process Switch Register: Description corrected
		35	Section title changed to "4.20 FPCKAR — Flash Sequencer Processing Clock Notification Register"
		35	4.20 FPCKAR — Flash Sequencer Processing Clock Notification Register: Note 1 corrected
		36	Table 4.22 FECCEMON Register Contents: Description of bits 15 to 10, changed
		37	4.22 FECCTMD — Flash ECC Test Mode Register: Note 1 corrected
		38	4.23 FDMYECC — Flash Dummy ECC Register: Description of "Access" corrected "read" → "read/written"
		40	Section title changed to "4.25 SELFID0 to SELFID3 — Self-Programming ID Input Registers"
		42	4.27.1 FHVE15 — FHVE15 Control Register: Description added
		43	4.27.2 FHVE3 — FHVE3 Control Register: Description added
		Section 6 FACI Command	
		45	Table 6.1 List of FACI Commands: Description changed in the Function column for FACI Command, "Status clearing" and "Forced stop"
		50	Figure 6.1 Overview of Command Usage in Code Flash Programming/Erasure Mode (1): Description changed
		51	Figure 6.2 Overview of Command Usage in Code Flash Programming/Erasure Mode (2): Description changed

REVISION HISTORY	RH850/E1x-FCC2 User's Manual: Hardware Interface

Rev.	Date	Description	
		Page	Summary
1.00	Jun 13, 2016	52	Figure 6.3 Overview of Command Usage in Data Flash Programming/Erasure Mode (1): Description changed
		53	Figure 6.4 Overview of Command Usage in Data Flash Programming/Erasure Mode (2): Description changed
		54	6.3.3 FCU Firmware Transfer: Description added
		56	Figure 6.6 Transfer Flow of FCU Firmware (2): Corrected
		57	6.3.3 FCU Firmware Transfer: Description added FCU firmware storage area: BaseAddress = 0001_7000_H FCU firmware storage area 2: BaseAddress = 0103_7000_H
		59	Figure 6.10 Flow of Shift to Read Mode: Note 1 added
		60	6.3.7 ID Authentication: Description changed
		62	Figure 6.12 Return from Command Lock State: Note 1 added
		63	6.3.9 Issuing of Programming Command: Description changed
		64	Figure 6.13 Programming Command Usage: Description changed
		64	Figure 6.13 Programming Command Usage: Note 1 and Note 2 added
		65	6.3.10 DMA Programming Command: Description changed
		65	Figure 6.14 DMA Programming Command Usage: Corrected (Description changed and processing added)
		65	Figure 6.14 DMA Programming Command Usage: Note 1 added
		66	6.3.11 Block Erasure command: Description changed
		66	Figure 6.15 Block Erasure Command Usage: Description changed
		66	Figure 6.15 Block Erasure Command Usage: Note 1 added
		68	Figure 6.16 Programming/Erasure Suspension Command Usage: (1) Figure changed (Description changed and processing added) (2) Note 1 and Note 2 added
		69	(1)Suspend programming command: Description of "Command" changed, and description added
		70	(2) Suspend erasure command in suspension-priority mode: Description changed
		70	Figure 6.18 Suspend Erasure Command (Suspension-Priority Mode): Description changed
		71	(3) Suspend erasure command in erasure-priority mode: Description changed
		71	Figure 6.19 Suspend Erasure Command (Erasure-Priority Mode): Description changed
		72	Figure 6.20 Programming/Erasure Resumption Command Usage: Note 1 added
		74	6.3.15 Forced Stop Command: Description changed
		74	Figure 6.22 Forced Stop Command Usage: (1) Figure changed (Description changed and processing added) (2) Note 1 added
		74	6.3.15 Forced Stop Command: Description added below Figure 6.22 • Issuing the Forced Stop Command while Another Command is Being Issued
		75	6.3.16 Blank Checking Command: Description changed
		76	Figure 6.23 Blank Checking Command Usage: Description changed
		76	Figure 6.23 Blank Checking Command Usage: Note 1 added
		78	Figure 6.24 Config Programming Command Usage
		78	Figure 6.24 Config Programming Command Usage: Note 1 added
		81	6.3.19 Lock Bit Programming Command: Description added
		81	Figure 6.26 Lock Bit Programming Command Usage: Figure description changed
		81	Figure 6.26 Lock Bit Programming Command Usage: Note 1 added
		82	6.3.20 Lock Bit Reading Command: Description changed
		82	Figure 6.27 Lock Bit Reading Command Usage: Note 1 added
		84	Figure 6.28 OTP Setting Command Usage: Description changed
		84	Figure 6.28 OTP Setting Command Usage: Note 1 added
		89	Figure 6.31 Injecting an ECC Error for the Code Flash Memory: (1) Figure changed (Description changed and processing added) (2) Note 1 and Note 2 added
		90	Figure 6.32 Injecting an ECC Error for the Data Flash Memory: Description changed
		90	Figure 6.32 Injecting an ECC Error for the Data Flash Memory: Note 1 added
		92	Section title changed to "8.1.2 Lock Bit"

REVISION HISTORY	RH850/E1x-FCC2 User's Manual: Hardware Interface

Rev.	Date	Description	
		Page	Summary
1.00	Jun 13, 2016	92	8.2 Error Protection: Description changed
		93	Table 8.1 Error Protection Type (1/2) (1) FENTRYR setting error: Description corrected (0001_H and 0080_H → 0000_H, 0001_H and 0080_H) (2) FCU error, deleted
		Section 9 Usage Notes	
		96	(1) Reading areas where programming or erasure was interrupted: Section title and description, changed
		96	(2) Prohibition of additional writing: Section title and description, changed
		96	(3) Resets during programming and erasure: Section title and description, changed
		96	(4) Allocation of vectors for interrupts and other exceptions during programming and erasure: Section title and description, changed
		96	(5) Abnormal termination of programming and erasure: Section title and description, changed
		96	(6) Items prohibited during programming and erasure: Section title and descriptionm, changed
		97	(7) Update of the FCUFAREA register: Section title and descriptionm, changed
		97	(8) Point for Caution on Selecting the FCU Firmware Storage Area: Note 1 added
		Section 10 Electrical Characteristics	
		98 to 103	Section added
		Appendix List of Flash Memory Related Registers	
		104	Table Appendix 1 List of Flash Memory Related Registers: (1) "Peripheral Group" column added (2) "*1" for the initial value of the FACI reset transfer status register, deleted (3) Note 1, deleted (4) Register names of symbols, FENTRYR and FPESTAT, changed
		104	Table Appendix 2 List of Product Information Related Registers: "Peripheral Group" column added
		105	Table Appendix 3 List of Another Registers: (1) "Peripheral Group" column added (2) Access size in the self-programming ID authentication status register, corrected (3) Module Name: "SYS" changed to "FLASH" (4) Note 1: Description corrected (OCDID0/OCDID1 → ID code)

RH850/E1x-FCC2
User's Manual: Hardware Interface

Publication Date: Rev.0.10 Mar 16, 2015
Rev.1.00 Jun 13, 2016

Published by:Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 4.0

RH850/E1x-FCC2
Flash Memory

Renesas Electronics Corporation

R01UH0552EJ0100