FM3 Fujitsu Cortex M3

32-BIT MICROCONTROLLER

MB9A420L/120L/ MB9B120J Series FLASH PROGRAMMING MANUAL

For the information for microcontroller supports, see the following web site.

http://edevice.fujitsu.com/micom/en-support/

FUJITSU SEMICONDUCTOR LIMITED

ARM™

FUJITSU

Preface

■ Purpose of this manual and intended readers

This manual explains the functions, operations and serial programming of the flash memory of this series.
This manual is intended for engineers engaged in the actual development of products using this series.

■ Trademark

ARM and Cortex are the trademarks of ARM Limited in the EU and other countries.

The company names and brand names herein are the trademarks or registered trademarks of their respective owners.

■ Organization of this Manual

This manual consists of the following 3 chapters.

CHAPTER 1 Flash Memory

This chapter gives an overview of, and explains the structure, operation, and registers of the Flash memory.

CHAPTER 2 Flash Security

The flash security feature provides possibilities to protect the content of the flash memory.
This chapter section describes the overview and operations of the flash security.

CHAPTER 3 Serial Programming Connection

This chapter explains the basic configuration for serial programming to flash memory by using the Fujitsu Semiconductor Serial Programmer.

■ Sample programs and development environment

Fujitsu Semiconductor offers sample programs free of charge for operating the peripheral functions of the FM3 family. Fujitsu Semiconductor also makes available descriptions of the development environment required for this series. Feel free to use them to verify the operational specifications and usage of this Fujitsu Semiconductor microcontroller.

· Microcontroller support information:
   http://edevice.fujitsu.com/micom/en-support/

* : Note that the sample programs are subject to change without notice. Since they are offered as a way to demonstrate standard operations and usage, evaluate them sufficiently before running them on your system.
Fujitsu Semiconductor assumes no responsibility for any damage that may occur as a result of using a sample program.

- FUJITSU SEMICONDUCTOR LIMITED, its subsidiaries and affiliates (collectively, "FUJITSU SEMICONDUCTOR") reserves the right to make changes to the information contained in this document without notice. Please contact your FUJITSU SEMICONDUCTOR sales representatives before order of FUJITSU SEMICONDUCTOR device.

- Information contained in this document, such as descriptions of function and application circuit examples is presented solely for reference to examples of operations and uses of FUJITSU SEMICONDUCTOR device. FUJITSU SEMICONDUCTOR disclaims any and all warranties of any kind, whether express or implied, related to such information, including, without limitation, quality, accuracy, performance, proper operation of the device or non-infringement. If you develop equipment or product incorporating the FUJITSU SEMICONDUCTOR device based on such information, you must assume any responsibility or liability arising out of or in connection with such information or any use thereof. FUJITSU SEMICONDUCTOR assumes no responsibility or liability for any damages whatsoever arising out of or in connection with such information or any use thereof.

- Nothing contained in this document shall be construed as granting or conferring any right under any patents, copyrights, or any other intellectual property rights of FUJITSU SEMICONDUCTOR or any third party by license or otherwise, express or implied. FUJITSU SEMICONDUCTOR assumes no responsibility or liability for any infringement of any intellectual property rights or other rights of third parties resulting from or in connection with the information contained herein or use thereof.

- The products described in this document are designed, developed and manufactured as contemplated for general use including without limitation, ordinary industrial use, general office use, personal use, and household use, but are not designed, developed and manufactured as contemplated (1) for use accompanying fatal risks or dangers that, unless extremely high levels of safety is secured, could lead directly to death, personal injury, severe physical damage or other loss (including, without limitation, use in nuclear facility, aircraft flight control system, air traffic control system, mass transport control system, medical life support system and military application), or (2) for use requiring extremely high level of reliability (including, without limitation, submersible repeater and artificial satellite). FUJITSU SEMICONDUCTOR shall not be liable for you and/or any third party for any claims or damages arising out of or in connection with above-mentioned uses of the products.

- Any semiconductor devices fail or malfunction with some probability. You are responsible for providing adequate designs and safeguards against injury, damage or loss from such failures or malfunctions, by incorporating safety design measures into your facility, equipments and products such as redundancy, fire protection, and prevention of overcurrent levels and other abnormal operating conditions.

- The products and technical information described in this document are subject to the Foreign Exchange and Foreign Trade Control Law of Japan, and may be subject to export or import laws or regulations in U.S. or other countries. You are responsible for ensuring compliance with such laws and regulations relating to export or re-export of the products and technical information described herein.

- All company names, brand names and trademarks herein are property of their respective owners.

How to Use This Manual

■ Searching for a function

The following methods can be used to search for the explanation of a desired function in this manual:

· Search from the table of the contents
  The table of the contents lists the manual contents in the order of description.

· Search from the register
  The address where each register is located is not described in the text. To verify the address of a register, see "APPENDIXES Register Map" of "FM3 Family Peripheral Manual".

■ Terminology

This manual uses the following terminology.

Term	Explanation
Word	Indicates access in units of 32 bits.
Half word	Indicates access in units of 16 bits.
Byte	Indicates access in units of 8 bits.

■ Notations

· The notations in bit configuration of the register explanation of this manual are written as follows.
  · bit   : bit number
  · Field : bit field name
  · Attribute : Attributes for read and write of each bit
    · R   : Read only
    · W   : Write only
    · R/W : Readable/Writable
    · -   : Undefined
  · Initial value : Initial value of the register after reset
    · 0 : Initial value is "0"
    · 1 : Initial value is "1"
    · X : Initial value is undefined

· The multiple bits are written as follows in this manual.
  Example : bit7:0 indicates the bits from bit7 to bit0

· The values such as for addresses are written as follows in this manual.
  · Hexadecimal number : "0x" is attached in the beginning of a value as a prefix (example : 0xFFFF)
  · Binary number   : "0b" is attached in the beginning of a value as a prefix (example: 0b1111)
  · Decimal number   : Written using numbers only (example : 1000)

Table 1 Applicable Products

Type name*	Flash memory size
	64Kbyte
TYPE10	MB9BF121J
TYPE11	MB9AF421K MB9AF421L
	MB9AF121K MB9AF121L

* : These type names are used to group applicable products in FM3 family peripheral manual.

CONTENTS

FUJITSU SEMICONDUCTOR LIMITED

MAJOR CHANGES IN THIS EDITION

Page	Section	Change Results
-	-	First edition

CHAPTER : Flash Memory

This chapter explains the overview, configuration, operation, and registers of the Flash memory. This series is equipped with the Flash Memory of 64 Kbytes for which Flash erase (simultaneous data erase of all sectors), data erase of each sector, and data write with CPU can be implemented.

1. Overview
2. Configuration
3. Operation Explanation
4. Registers

CODE: 9B120JFLASH-E01.0

1. Overview

This series is equipped with 64Kbytes Flash Memory.

The built-in Flash Memory could erase data by-sector, or erase flash (erase data by-all-sector collectively), and write programmed data by byte(8 bits) or by half-word(16 bits) with the Cortex-MX3 CPU.

■ Flash Memory Features

- · Usable capacity: 64 Kbytes
- · Detection of write/erase completion with CPU interrupt
- · High-speed flash memory:
  Up to 40 MHz: 0Wait
  Up to 72 MHz: 0 Wait by enabling pre-fetch buffer (only for TYPE10 products)

- · Operating mode:
1. CPU mode
    This mode allows reading, writing, and erasing of flash memory from CPU (automatic algorithm*).
    The operation of writing data by byte (8 bits) or by half word (16 bits) is available.
    To rewrite data, execute a program on RAM.

2. ROM writer mode
    This mode allows reading, writing, and erasing of flash memory from a ROM writer (automatic algorithm*).

- · Built-in flash security function
  (Prevents reading of the content of flash memory by a third party)
  See "CHAPTER Flash Security" for details on the flash security function.

<Note>

This document explains the usage of flash memory in CPU mode.

For details on accessing the flash memory from a ROM writer, see the instruction manual of the ROM writer that is being used.

*: Automatic algorithm = Embedded Algorithm

2.   Configuration

This series consists of 64 Kbytes Flash Memory area, a s ecurity code area, and a C R trimming data area.

Table 2-1 shows the correspondence between Capacity of Flash Memory built into this series and Product TYPE. Figure 2-1 and Figure 2-2 show the address and sector structure of the Main Flash memory built into this series as well as the address of security/CR trimming data.

See "CHAPTER Flash Security" for details on the security.

See Section "4.6CRTRMM (CR Trimming Data Mirror Register)" and "CHAPTER High-Speed CR Trimming" in the "FM3 Family Peripheral Manual" for details on the High-Speed CR trimming data.

Table 2-1 Flash Memory Capacity by Product TYPE

Product TYPE	Flash Memory Capacity	
	64Kbytes	
TYPE10	MB9BF121J	
TYPE11	MB9AF421K MB9AF421L	
	MB9AF121K MB9AF121L	

Figure 2-1 Address and Sector Structure of 64 Kbytes Flash Memory
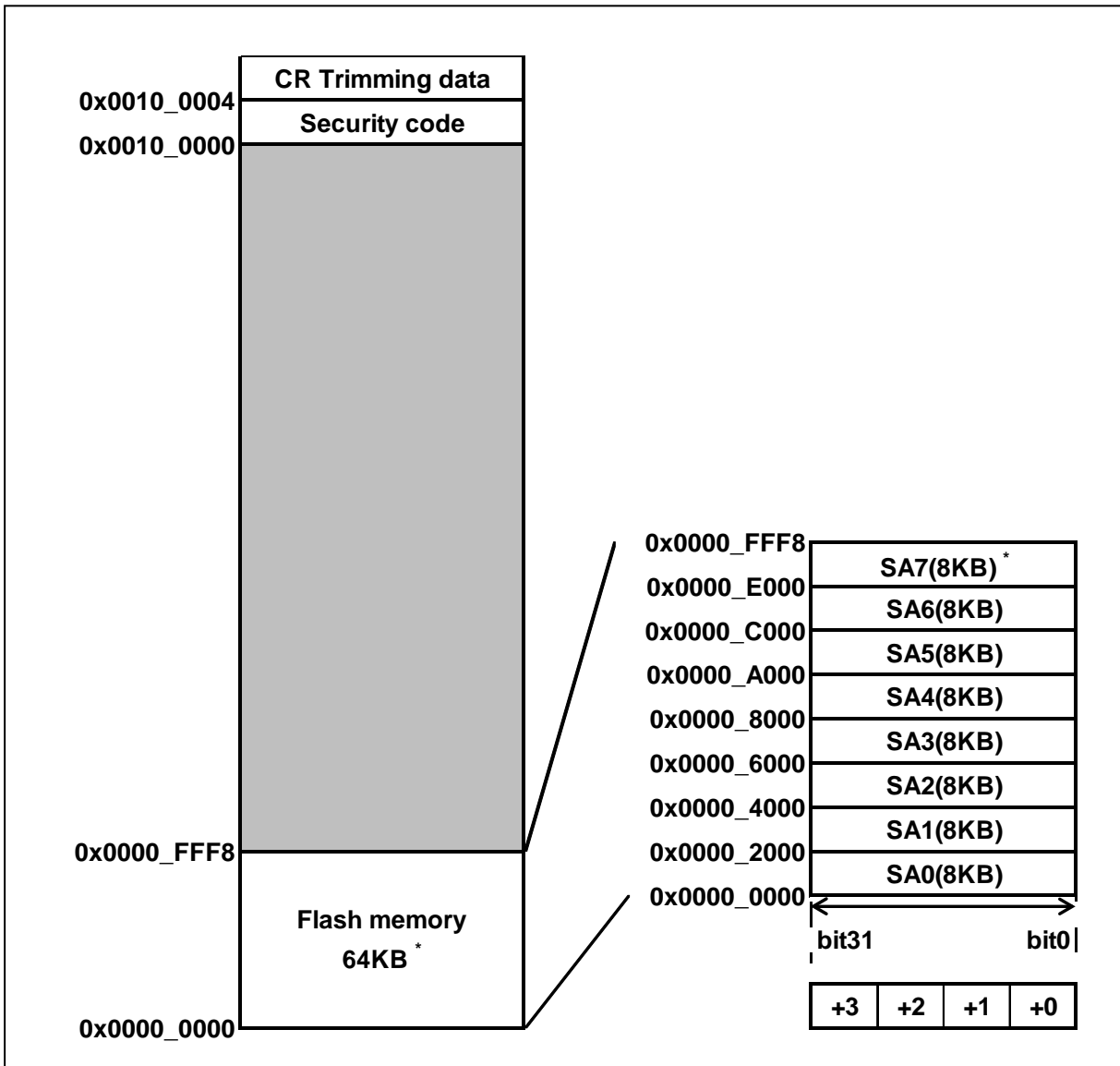
* : Eight bytes of SA7(8KB) are assigned to a Flash security code and a CR trimming data(0x0010_0000 to 0x0010_0007). Therefore, a Flash security code and CR trimming data are also erased at sector erase.

Figure 2-2 Address of Security/CR Trimming Data
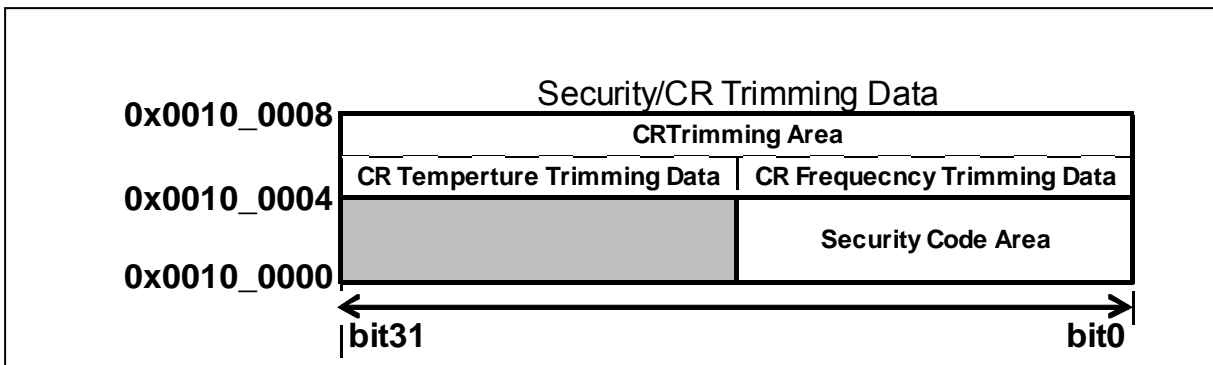
3.   Operation Explanation

This section explains the Flash memory operation.

3.1. Automatic Algorithm

Writing to and erasing Flash memory is performed by activating the automatic algorithm.
This section explains the automatic algorithm.

3.1.1   Command Sequences

3.1.2   Command Operations

3.1.3   Automatic Algorithm Run States

3.1.1. Command Sequences

The automatic algorithm is activated by writing data to the Flash memory in the specified method. This is called a command. Table 3-1 shows the command sequences.

Table 3-1 Command sequence chart

Command	Number of writes	1st write		2nd write		3rd write		4th write		5th write		6th write	
		Address	Data	Address	Data	Address	Data	Address	Data	Address	Data	Address	Data
Read/ Reset	1	0xXXX	0xF0	--	--	--	--	--	--	--	--	--	--
Write	4							0xA0	PA	PD	--	--	--
Flash erase	6											0x554	0x10
Sector erase (No Sector added)	6	0x554	0xAA	0xAA8	0x55	0x554	0x80	0x554	0xAA	0xAA8	0x55	SA	0x30
Sector erase (Sector added)	6 and on												0xE0*
Sector erase suspended	1	0xXXX	0xB0	--	--	--	--	--	--	--	--	--	--
Sector erase restarting	1	0xXXX	0x30	--	--	--	--	--	--	--	--	--	--

X : Any value

PA : Write address

SA : Sector address (Specify any address within the address range of the sector to be erased)

PD : Write data

*: To add sectors to be erased, repeat the 6th write operation for the required times. By writing 0x30 in the last sector address, the erase operation is started.

<Notes>

· The data notation in Table 3-1 only shows the lower 8 bits. The upper 8 bits can be set to any value.
· Write commands in a byte (8 bits) or half-words (16 bits).
· The address notation in Table 3-1 only shows the lower 12 bits. The upper 20 bits should be set to any address within the address range of the target flash memory.
  When the address outside the flash address range is specified, the command sequence would not be executed correctly since the flash memory cannot recognize the command.
· Specify 0x0010_0000 to the address to set a flash security code.
· To make the CR Trimming Data settings, assign 0x0010_0004 (CR Frequency Trimming Data) and 0x0010_0006 (CR Temperature Trimming Data) as addresses.
· When any of SA7 and 0x0010_0000 to 0x0010_0007 is assigned for "an arbitrary address within address range", SA7, Flash Security Code, and CR Trimming Data are erased thoroughly. Only any of them cannot be erased.

3.1.2. Command Operations

This section explains the command operations.
To execute the command operation program, store the program in RAM.

■ Read/Reset Command

The flash memory can be read and reset by sending the read/reset command.

When a read/reset command is issued, the flash memory maintains the read state until another command is issued.

When the execution of the automatic algorithm exceeds the time limit (HANG), the flash memory is returned to the read/reset state by issuing the read/reset command.

The read/reset command issued during each command operation is valid. In this case, commands previously issued are cleared. So, the commands issued should be issued again from the first command.

See Section "3.2.1 Read/Reset Operation" for details on the actual operation.

■ Program (Write) Command

The data is written in the address specified at the fourth time by issuing the write command to the target sector for four consecutive times. Data can be written in a byte (8 bits) or a half-word (16 bits) according to the data width specified at the fourth time. For the first to third commands, the data width is not judged.

Once the forth command issuance has finished, the automatic algorithm is activated and the automatic write to the flash memory starts. After executing the automatic write algorithm command sequence, there is no need to control the flash memory externally.

See Section "3.2.2 Write Operation" for details on the actual operation.

<Note>

Only a single byte or half-word of data can be written for each write command sequence.
To write multiple pieces of data, issue one write command sequence for each piece of data.

■ Flash Erase (All Sector Batch Erase) Command

All of the sectors in the flash memory can be batch-erased by sending the flash erase command in six consecutive writes. Once the sixth sequential write has finished, the automatic algorithm is activated and the flash erase operation starts.

<Note>

By flash erase command, the security/CR trimming data value can also be erased.

■ Sector Erase Command

By sending the sector erase command in six consecutive writes, the single sector specified by sixth write can be erased. When the sixth write data is "0x30", the automatic algorithm is activated and the sector erase operation begins.

To erase multiple sectors, issue the sector erase code (0xE0) which is the sixth write data. To erase more sectors, write the sector erase code, "0xE0" to sector address added in the seventh time and later. By writing the sector erase code, "0x30" in the last write data, the automatic algorithm is activated and the sector erase operation begins.

There is no restriction on number of sectors to be erased and all sectors can be erased collectively.

<Note>

· The sector erase is started only by writing 0x30 to the sector erase command. 0xE0 cannot start the erase operation.
· Since SA7, Flash Security Code, and CR Trimming Data are stored in the same sector, all of them would be erased if any one of them is selected to be erased. Only any one of them cannot be erased.

■ Sector Erase Suspended Command

By issuing the sector erase suspended command during sector erase, sector erase can be suspended. In the sector erase suspended state, the read and write operations of memory cells of the sector not to erase is possible.
See Section "3.2.5 Sector Erase Suspended Operation" for details on the actual operation.

<Notes>

· This command is only valid during sector erase. It is ignored even if it is issued during flash erase or during write.
· During the sector erase suspended state, the flash erase and the erase of sectors other than erase target sectors is not executed.

■ Sector Erase Restart Command

In order to restart the erase operation in the sector erase suspended state, issue the sector erase restart command. Issuing the sector erase restart command returns the flash memory to the sector erase state and restarts the erase operation.
See Section "3.2.6 Sector Erase Restart Operation" for details on the actual operation.

<Note>

This command is only valid during sector erase suspended. It is ignored even if it is issued during sector erase.

3.1.3.   Automatic Algorithm Run States

Writing and erasing of flash memory is performed by the automatic algorithm. Whether or not the automatic algorithm is currently executing can be checked by using the Flash Status Register (FSTR).

■ **Flash Status Register**

This indicates the status of the automatic algorithm. Figure 3-1 shows the bit structure of the Flash Status Register.

Figure 3-1 Bit structure of the Flash Status Register

bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	PGMS	SERS	ESPS	CERS	HNG	RDY

<Note>

Because the correct value might not be read out immediately after issuing a command, ignore the first value of the Flash Status Register that is read immediately after issuing a command.

● **Status of each bit and Flash memory**

For the correspondence between each bit of the Flash Status register and the status of the flash memory, see Table 3-2.

Table 3-2 List of each bit of Flash Status Flag Register

Parameter			PGMS	SERS	ESPS	CERS	HNG	RDY
Running	Program write operation		1	0	0	0	0	0
	Flash memory erase	Internal operation before erasing	0	0	0	0	0	0
		Erasing	0	0	0	1	0	0
	Sector erase		0	1	0	0	0	0
	Sector erase suspended	Program write operation (Sector not to erase)	1	1	1	0	0	0
		Other than above	0	1	1	0	0	1
Time limit exceeded	Program write operation		1	0	0	0	1	0
	Flash memory erase	Internal operation before erasing	0	0	0	0	1	0
		Erasing	0	0	0	1	1	0
	Sector erase		0	1	0	0	1	0
	Sector erase suspended	Program write operation (Sector not to erase)	1	1	1	0	1	0

(Note)　See "● Bit Descriptions" for the values that can be read.

● **Bit Descriptions**

[bit7:6] Reserved bits

[bit5] PGMS: Program Write Operation Status flag bit

    After issuing the Program Write Operation command, RDY bit becomes "0" and this bit is set to "1".
    This bit of "1" means the Program Write Operation status.
    After the Program Write Operation is finished, this bit is cleared to "0" and RDY bit becomes "1".

[bit4] SERS: Sector Erase Status flag bit

    After issuing the Sector Erase command, RDY bit becomes "0" and this bit is set to "1".
    This bit of "1" means the Sector Erase status.
    After the Sector Erase Operation is finished, this bit is cleared to "0" and RDY bit becomes "1".

    When the Sector Suspend command is issued during sector erasing, the erasing operation is suspended.
    During this suspended status, this bit continues to be "1".

[bit3] ESPS: Sector Erase Suspend Status flag bit

    When the Sector Erase Suspend command is issued during sector erasing, the erasing operation is suspended, RDY bit becomes "1", and this bit is set to "1". This bit of "1" means the Sector Erase Suspend Status.
    By issuing the Sector Erase Restart command, this bit is cleared to "0" and the Sector Erase Operation is restarted.
    In the Sector Erase Suspend Status, the Program Write Operation command can be issued to the sectors not to erase. By issuing the command, the RDY bit is set to "0" and PGMS bit is set to "1" to transfer to "Program Write Operation status".

[bit2] CERS: Flash Memory Erase Status flag bit

    When the Flash Erase command is issued, RDY bit becomes "0" and the Flash Memory is transferred to Pre-erasing Internal Operation Status. In this status, this bit is not set to "1" yet. After the Pre-erasing Internal Operation is completed, the Flash erasing is started and this bit is set to "1".
    This bit of "1" means Flash Erasing Status.
    After the Flash Erasing operation is completed, this bit is cleared to "0" and RDY bit becomes "1".

[bit1] HNG: HANG Status flag bit

    The Internal Timer is mounted on the Flash Memory to provide the execution limit time of the Automatic Algorithm. When the Automatic Algorithm is not completed within the executing limit time specified by the internal timer, the Flash Memory becomes HANG Status and this bit is set to "1".

    To clear this bit and to return to the normal status, it is required to generate the reset or to issue the reset command.

[bit0] RDY: RDY Status flag bit

    This bit indicates whether the Automatic Algorithm is being executed or not. When the Flash Memory is being written or erased, this bit is set to "0" and the execution command cannot be accepted in this status. When this bit is "1", the execution command can be accepted (except the Sector Suspend command during sector erasing.)

3.2. Explanation of Flash Memory Operation

The operation of the Flash memory is explained for each command.

3.2.1.   Read/Reset Operation

This section explains the read/reset operation.

To place the flash memory in the read/reset state, send read/reset commands to an arbitrary address within the address range of the flash memory.

Because the read/reset state is the default state of the flash memory, the flash memory always returns to this state when the power is turned on or when a command finishes successfully. When the power is turned on, there is no need to issue a data read command. Furthermore, because data can be read by normal read access and programs can be accessed by the CPU while in the read/reset state, there is no need to issue read/reset commands.

3.2.2. Write Operation

This section explains the write operation.

Writes are performed according to the following procedure.

1. The program (write) command is issued sequentially.
   The automatic algorithm activates and the data is written to the flash memory.
   After the write command is issued, there is no need to control the flash memory externally.

2. Confirm Flash Status Register (FSTR)
   After issuing the program write command, the RDY bit and PGMS bit of the Flash Status Register are set to "0" and "1" respectively. After the write operation is completed, the RDY bit and PGMS bit are set to "1" and "0" respectively.

See Figure 3-2 for an example of a write operation to the flash memory.

Figure 3-2 Example write operation
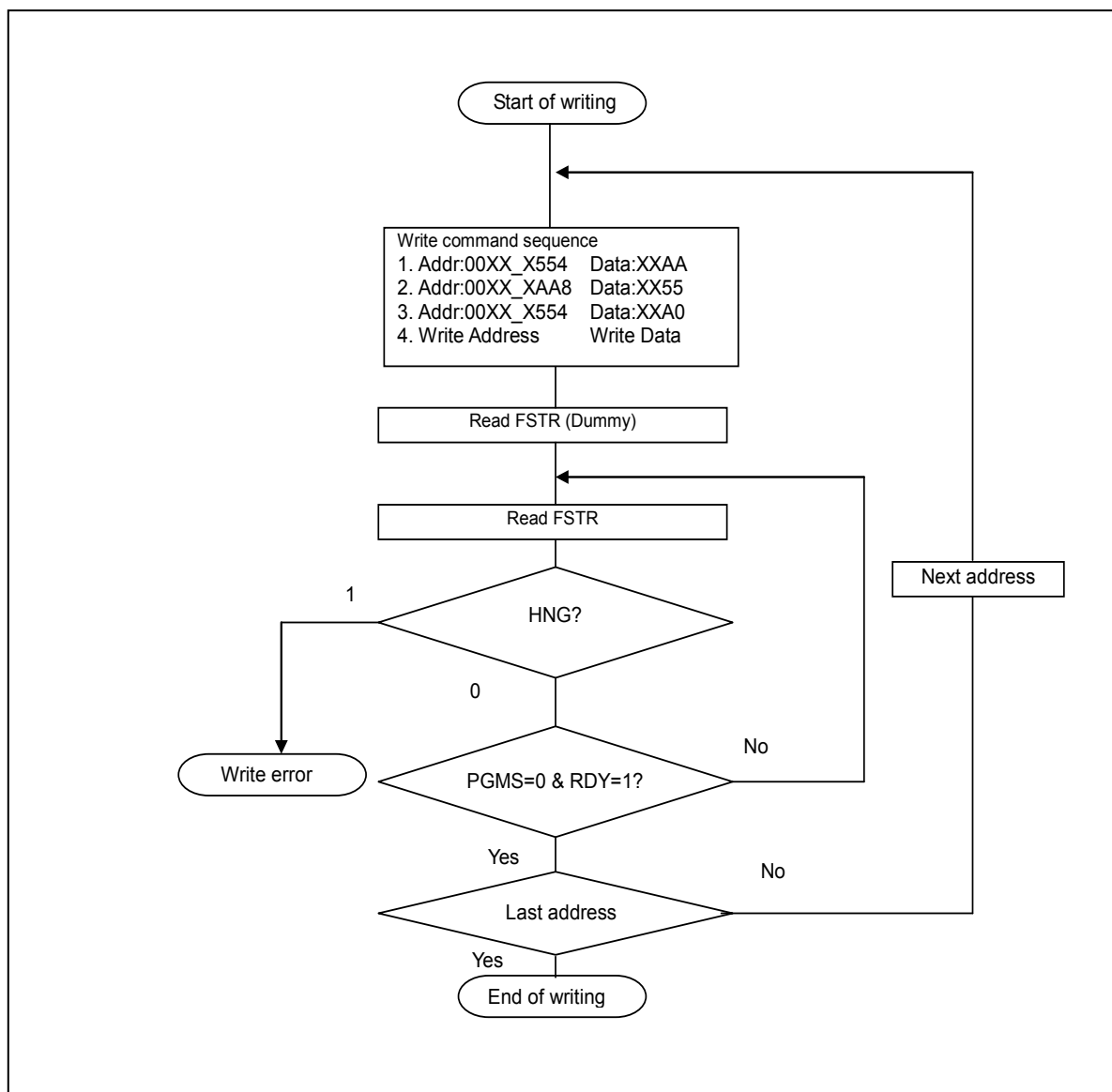
<Notes>

- See Section "3.1 Automatic Algorithm" for details on the write command.
- Because the value of the flash memory might not read correctly immediately after the command issued, ignore the first read value of Flash Status Register (FSTR) after the command issued.
- Although the flash memory can be written in any sequence of addresses regardless of crossing sector boundaries, only a single byte or a single half-word of data can be written with each write command sequence. To write multiple pieces of data, issue one write command sequence for each piece of data.
- All commands issued to the flash memory during the write operation are ignored.
- If the device is reset while the write is in progress, the data that is written is not guaranteed.

3.2.3.   Flash Erase Operation

This section explains the flash erase operation.

All sectors in flash memory can be erased in one batch. Erasing all of the sectors in one batch is called flash erase.

The automatic algorithm can be activated and all of the sectors can be erased in one batch by sending the flash erase command sequentially to the target sector.

See Section "3.1 Automatic Algorithm" for details on the flash erase command.

1. Issue the flash erase command sequentially
   The automatic algorithm is activated and the flash erase operation of the flash memory begins.

2. Confirm Flash Status Register (FSTR)
   After Flash Erase Command issued, the RDY bit of the Flash Status Register becomes "0". At this time, the flash memory, becomes pre-erasing internal operation status and CERS bit is held to be "0". After erasing operation is started, the CERS bit is set to "1".
   After the flash erase is completed, the RDY bit and CERS bit are cleared to "1" and "0" respectively. RDY bit and CERS bit are cleared to "1" and "0" respectively.

3.2.4. Sector Erase Operation

This section explains the sector erase operation.

Sectors in the flash memory can be selected and the data of only the selected sectors can be erased. Multiple sectors can be specified at the same time.

Sectors are erased according to the following sequence.

1. Issue the sector erase command sequentially to the target sector
   The automatic algorithm activates and the sector erase operation begins.

   To erase multiple sectors, write 0xE0 as the 6th write data (command data). By writing 0xE0 to the 7th sector address and later, sectors could be added to be erased. Write 0x30 in the last sector address, then the automatic algorithm is activated and the erase operation of multi sectors specified is started.
   There is no restriction on number of sectors added to be erased and all sectors can be erased collectively.

2. Confirm Flash Status Register (FSTR)
   After issuing Sector erase command, the RDY bit and SERS bit of the Flash Status Register are set to "0" and "1" respectively. After that, when the erase operation of all sectors specified are completed, the RDY bit and SERS bit are set to "1" and "0" respectively.

For an example of the sector erase procedure, see Figure 3-3.

Figure 3-3 Example of sector erase procedure
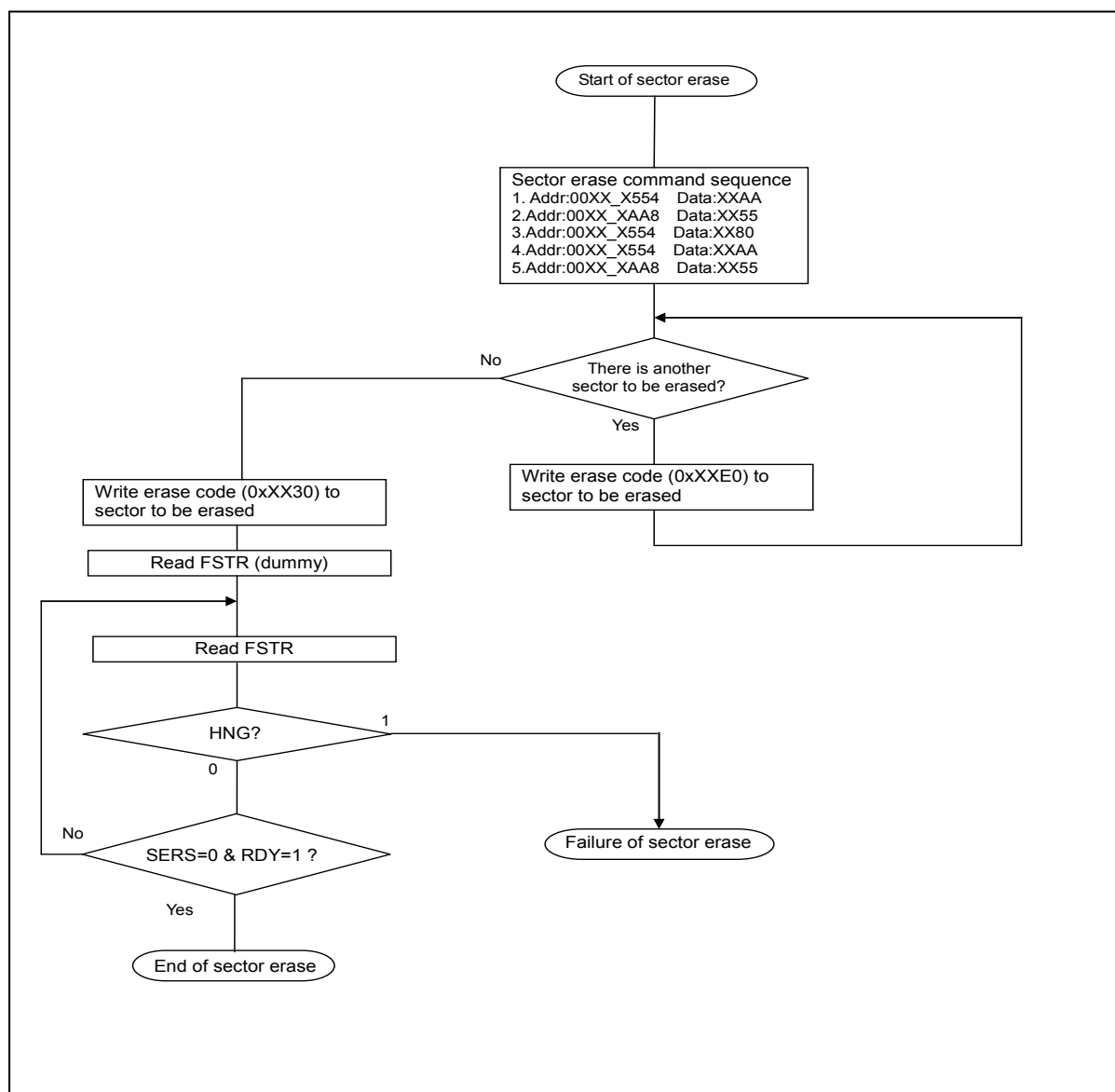
Once the sector erase operation has finished, the flash memory returns to read/reset mode.

<Notes>

· See Section "3.1 Automatic Algorithm" for details on the sector erase command.
· Because the value may not be read correctly immediately after issuing a command, ignore the first read value of the Flash Status Register (FSTR) after issuing the command.
· Since SA7, Flash Security Code, and CR Trimming Data are stored in the same sector, all of them would be erased if any one of them is selected to be erased. Only any one of them cannot be erased.

3.2.5.   Sector Erase Suspended Operation

This section explains the sector erase suspended operation.

When the sector erase suspended command is sent during sector erase, the flash memory makes a transition to the sector erase suspended state and temporarily suspends the erase operation.
By sending the erase restart command, the flash memory is returned to the sector erase state and can restart the suspended erase operation.

■ Sector Erase Suspended Operation

Sector erase is suspended in the following steps:

1. Write the sector erase suspended command to an arbitrary address within the address range of the flash memory during the erasing operation.
2. Confirm Flash Status Register (FSTR).
   Issuing the sector erase suspended command, the RDY bit and ESPS bit are set to "0" and "1" respectively. Under this condition, memory cells of a sector other than a sector to be erased are could be read and written.

<Notes>

· See Section "3.1 Automatic Algorithm" for details on the sector erase suspended command.
· When a sector to be erased is read after a sector erase suspend, the value of the sector is undefined.

3.2.6.   Sector Erase Restart Operation

This section explains the operation for restarting sector erase during sector erase suspended.

When the sector erase restart command is issued to an arbitrary address in the address range of the flash memory while sector erase is suspended, sector erase can be restarted.

When the sector erase restart command is issued, the sector erase operation during sector erase suspended is restarted.

See Section "3.1 Automatic Algorithm" for details on the sector erase restart command.

<Note>

The sector erase restart command is only valid during sector erase suspended. Even if the sector erase restart command is issued during sector erase, it is ignored.

3.3. Cautions When Using Flash Memory

This section explains the cautions when using the Flash memory.

- If this device is reset during the write, the data that is written cannot be guaranteed.
  Moreover, It is necessary to prevent an unexpected reset like Watchdog Timer from occurring during the writing and erasing.
- Immediately after issuing the automatic algorithm command to the flash memory, always perform a dummy read before reading the flash status register.
- If the device is forced to transfer to the low-power consumption mode, ensure the operations of the flash memory automatic algorithm is completed.
  See "CHAPTER Low Power Consumption Mode" of the "FM3 Family Peripheral Manual" for details on the low-power consumption mode.
- Since SA7, Flash Security Code, and CR Trimming Data are stored in the same sector, all of them would be erased if any one of them is selected to be erased. Only any one of them cannot be erased.

4. Registers

This section explains the registers.

■ List of Registers

Abbreviated Register Name	Register Name	Reference
FRWTR	Flash Read Wait Register	4.1
FSTR	Flash Status Register	4.2
FICR	Flash Interrupt Control Register	4.3
FISR	Flash Interrupt Status Register	4.4
FICLR	Flash Interrupt Clear Register	4.5
CRTRMM	CR Trimming Data Mirror Register	4.6

4.1. FRWTR (Flash Read Wait Register)

The FRWTR register specifies the wait cycle for the flash memory.

■ TYPE10 products

bit	7	6	5	4	3	2	1	0
Field	Reserved						RWT	
Attribute							R/W	R/W
Initial Value							1	0

[bit7:2] Reserved bits

The read values are undefined. Ignored on write.

[bit1:0] Read Wait Cycle

Specifies the read wait cycle for the Flash Memory.

Field	bit	Description

RWT	1:0	00: 0 cycle wait mode
		     This setting can be used when HCLK is 20MHz or less.
		01: 0 to 1 cycle wait mode
		     This setting can be used when HCLK is more than 20 MHz and 40MHz or less.
		10: Pre-fetch mode (Initial value)
		     This setting should be specified when HCLK is more than 40MHz.
		11: Setting prohibited

<Notes>

· When HCLK is more than 20MHz, the usage at the setting of RWT=00 is prohibited. At the setting of RWT=00, manage HCLK not to exceed 20MHz for even a moment.

· When HCLK is more than 40MHz, the usage at the setting RWT=00 or RWT=01 is prohibited. When the setting is RWT=00 or RWT=01, manage HCLK not to exceed 40MHz for even a moment.

· After changing this register, be sure to execute the dummy read for this register.

■ TYPE11 products

bit	7	6	5	4	3	2	1	0
Field	Reserved						RWT	
Attribute							R/W	R/W
Initial Value							0	1

[bit7:2] Reserved bits

The read values are undefined. Ignored on write.

[bit1:0] RWT: Read Wait Cycle

Specifies the read wait cycle for the Flash Memory.

Field	bit	Description

RWT	1:0	00: 0 cycle wait mode
		    This setting can be made when HCLK is 20MHz or less.
		01: 0 to 1cycle wait (Initial value)
		    This setting should be set when HCLK is more than 20MHz.
		10: Setting prohibited
		11: Setting prohibited

\<Notes\>

· When HCLK is more than 20MHz, the usage at the setting of RWT=00 is prohibited. At the setting of RWT=00, manage HCLK not to exceed 20MHz for even a moment.
· After changing this register, be sure to execute the dummy read for this register.

4.2. FSTR (Flash Status Register)

The FSTR register is a status register of the flash memory.

bit	7	6	5	4	3	2	1	0
Field	Reserved		PGMS	SERS	ESPS	CERS	HNG	RDY
Attribute			R	R	R	R	R	R
Initial Value			0	0	0	0	0	X

[bit7:6] Reserved bits

The read values are undefined. Ignored on write.

[bit5] PGM: Flash Program Status

Indicates the program (writing operation) status of the flash memory.

Field	bit	Description
PGMS	5	0: The program is not being written to the flash memory (Initial value) 1: The program is being written to the flash memory.

[bit4] SERS: Flash Sector Erase Status

Indicates the sector erase status of the flash memory.

Field	bit	Description
SERS	4	0: The sector is not being erased (Initial value) 1: The sector is being erased or the sector erase is being suspended.

[bit3] ESPS: Flash Erase Suspend Status

Indicates the sector erase suspend status of the flash memory.

Field	bit	Description
ESPS	3	0: The sector erase is not being suspended (Initial value) 1: The sector erase is being suspended.

[bit2] CERS: Flash Chip Erase Status

Indicates the all sector erase status of the flash memory.

Field	bit	Description
CERS	2	0: The flash memory is not being data erased (Initial value) 1: The flash memory is being data erased.

[bit1] HNG: Flash Hang Status

Indicates whether the flash memory is in the HANG state. The flash memory enters the HANG state if the timing is exceeded.

If this bit becomes "1", issue a reset command (See Section "3.1.1 Command Sequences").

Field	bit	Description
HNG	1	0: The flash memory HANG state has not been detected (Initial value) 1: The flash memory HANG state has been detected.

[bit0] RDY: Flash Ready Status

Indicates whether flash memory write or erase operation using the automatic algorithm is in progress or finished. While an operation is in progress, data cannot be written and the flash memory cannot be erased.

Field	bit	Description
RDY	0	0: Operation in progress (cannot accept write/erase command) 1: Operation finished (can accept write/erase command)

4.3. FICR (Flash Interrupt Control Register)

The FICR register specifies the interrupt enable setting of the FLASH memory.

bit	7	6	5	4	3	2	1	0
Field	Reserved						HANGIE	RDYIE
Attribute							R/W	R/W
Initial Value							0	0

[bit7:2] Reserved bits

The read values are undefined. Ignored on write.

[bit1] HANGIE: HANG Interrupt Enable

This bit enables the flash HANG status interrupt. When the HANGIF bit of the FISR register and this bit are both "1", an interrupt to CPU is generated.

Field	bit	Description
HANGIE	1	0: FLASH HANG interrupt is prohibited (Initial value). 1: FLASH HANG interrupt is permitted.

[bit0] RDYIE: RDY Interrupt Enable

This bit enables the flash RDY status interrupt. When the RDYIF bit of the FISR register and this bit are both "1", an interrupt to CPU is generated.

Field	bit	Description
RDYIE	0	0: FLASH RDY interrupt is prohibited (Initial value). 1: FLASH RDY interrupt is permitted.

<Note>

Clear the bit in question of the FISR register and set the bit of this register to "1" to enable the interrupt.

4.4. FISR (Flash Interrupt Status Register)

The FISR register indicates the interrupt status of the FLASH memory.

bit	7	6	5	4	3	2	1	0
Field	Reserved						HANGIF	RDYIF
Attribute							R/W	R/W
Initial Value							0	0

[bit7:2] Reserved bits

The read values are undefined. Ignored on write.

[bit1] HANGIF: HANG Interrupt Flag

This bit is set to "1" when the Flash HANG status is detected. This bit is set to "1" with the rising edge of HANG signal. This bit is cleared to be "0" by writing "1" to the HANGC bit of the FICLR register.

Field	bit	Description
HANGIF	0	0: FLASH HANG status is not detected (Initial value). 1: FLASH HANG status is detected.

[bit0] RDYIF: RDY Interrupt Flag

This bit is set to "1" when the Flash RDY status is detected. This bit is set to "1" with the rising edge of RDY signal. This bit is cleared to be "0" by writing "1" to the RDYC bit of the FICLR register.

Field	bit	Description
RDYIF	0	0: FLASH RDY status is not detected (Initial value). 1: FLASH RDY status is detected.

4.5. FICLR (Flash Interrupt Clear Register)

The FICLR register clears the interrupt flag of flash memory.

bit	7	6	5	4	3	2	1	0
Field	Reserved						HANGC	RDYC
Attribute							W	W
Initial Value							0	0

[bit7:2] Reserved bits

The read values are undefined. Ignored on write.

[bit1] HANGC: HANG Interrupt Clear

This bit is the clear bit of HANG interrupt flag. This bit clears the HANGIF bit of the FISR register to "0" by writing "1" to this bit.

Field	bit	Description
HANGC	1	At Write 0: FLASH HANG interrupt flag (HANGIF) is not changed (Initial value) 1: FLASH HANG interrupt flag(HANGIF) is cleared to be "0" At Read "0" is read.

[bit0] RDYC: RDY Interrupt Flag

This bit is the clear bit of RDY interrupt flag. This bit clears the RDYIF bit of the FISR register to "0" by writing "1" to this bit.

Field	bit	Description
RDYC	0	At Write 0: FLASH RDY interrupt flag (RDYIF) is not changed (Initial value) 1: FLASH RDY interrupt flag (RDYIF) is cleared to be "0". At Read "0" is read.

4.6. CRTRMM (CR Trimming Data Mirror Register)

The CRTRMM register is the mirror register of the CR trimming data.
A value of this register can be used in the user mode and the serial writer mode.

bit	31	21	20	16	15	10	9	0
Field	Reserved		TTRMM		Reserved		TRMM	
Attribute			R				R	
Initial Value			*				*	

[bit31:21] Reserved bits

The read values are undefined. Ignored on write.

[bit20:16] TTRMM : CR Temperature Trimming Data Mirror Bit

After a reset is released, bit[4:0](CR Temperature Trimming Data) of address "0x0010_1006" in Flash Memory area are stored in this bit.
For details of High-speed CR Trimming data, see Chapter "High-speed CR Trimming Data" in "FM3 Family Peripheral Manual".

Field	bit	Description
TTRMM	20:16	*: Reads bit[4:0] of Address "0x0010_1006".

[bit15:10] Reserved bits

The read values are undefined. Ignored on write.

[bit9:0] TRMM : CR Trimming Data Mirror Bit

After a reset is released, bit[9:0](CR Frequency Trimming Data) of Address "0x0010_1004" in Flash Memory area are stored in this bit.
For details of High-speed CR Trimming data, see Chapter "High-speed CR Trimming Data" in "FM3 Family Peripheral Manual" .

Field	bit	Description
TRMM	9:0	*: Reads bit[9:0] of Address "0x0010_1004".

\<Note\>

After erasing the flash memory data and issuing "Reset" signal inside the chip, this register is cleared. At this time, the stored CR trimming data is deleted. So, save the stored CR trimming data to RAM etc. before this register is cleared.

CHAPTER : Flash Security

The flash security function protects contents of the flash memory.
This chapter explains the overview and operations of the flash security.

1. Overview
2. Operation Explanation

1.　Overview

This section explains the overview of the flash security.

If the protection code of 0x0001 is written in the security code area of the flash memory, access to the flash memory is restricted. Once the flash memories are protected, performing the flash erase operation only can unlockThis function is suitable for applications requiring security of self-containing program and data stored in the flash memory.

Table 1-1 shows the address and the protection code of the security code.

Table 1-1 Address of security code and protection code

Address	Protection Code
0x0010_0000	0x0001

2. Operation Explanation

This section explains the operation of the flash security.

■ Setting Security

Write the protection code 0x0001 in address of the security code. The security is enabled and set after all the reset factors are generated or after turning on the power again.

■ Releasing Security

The security is released by all the reset factors or power-on after the execution of flash erase.

■ Operation with Security Enabled

The operations with security enabled vary depending on each mode.

Table 2-1 shows the security operations in each mode.

Table 2-1 Flash Operation with Security Enabled

Mode	Mode pins		Access to flash			Access from debugging pin
	MD1	MD0	Flash erase	Other commands	Read	
User mode	-	0	Enabled	Enabled	Valid data	Disabled
Serial writer mode	0	1	Enabled	Disabled	Invalid data	Disabled

\<Notes\>

- Writing the protection code is generally recommended to take place at the end of the flash programming. This is to avoid unnecessary protection during the programming.
- In user mode, there is no limit to the flash memory even during security is enabled. However, debugging pins are fixed not to access internally from these pins during security is enabled. To release security, perform the flash erase operation using a serial writer because the security cannot be released through debugging pins.
- When security enabled, the obstruction analysis of the flash memory cannot be performed.

CHAPTER : Serial Programming Connection

This series supports serial on-board programming (Fujitsu Semiconductor standard) to the flash memory. This chapter explains the basic configuration for serial programming to the flash memory by using the Fujitsu Semiconductor Serial Programmer.

1. Serial Programmer

CODE: 9B120JFSP-E01.0

1.   Serial Programmer

Fujitsu Semiconductor Serial Programmer (software) is an on-board programming tool for all microcontrollers with built-in flash memory.

This serial programmer supports RS-232C interface.

1.1. Basic Configuration

This section explains the basic configuration.

■ Basic Configuration of FUJITSU SEMICONDUCTOR MCU Programmer (Clock Asynchronous Serial Programming)

FUJITSU SEMICONDUCTOR MCU Programmer writes data, through clock asynchronous serial communication, to built-in flash memory of a microcontroller installed in the user system when the PC and the user system are connected through RS-232C cable.

In these series, serial programming (UART communication mode) is possible by any clock, crystal oscillator or external clock or built-in High-speed CR oscillator.

Figure 1-1 shows the basic configuration of FUJITSU SEMICONDUCTOR MCU Programmer, and Table 1-1 lists the system configuration.

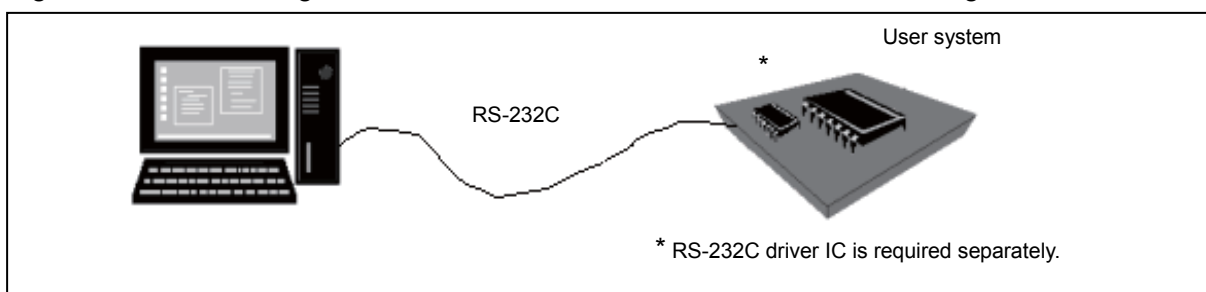Figure 1-1 Basic Configuration of FUJITSU SEMICONDUCTOR MCU Programmer

Table 1-1 System Configuration of FUJITSU SEMICONDUCTOR MCU Programmer

Name	Specifications
FUJITSU SEMICONDUCTOR MCU Programmer	Software (In case you request the data, contact to FUJITSU sales representatives.)
RS-232C cable	Sold on the market.

■ Connection Example of FUJITSU SEMICONDUCTOR MCU Programmer

The following shows a connection example of FUJITSU SEMICONDUCTOR MCU Programmer.

● When Crystal oscillator is used as the source oscillation clock

Figure 1-2 shows a connection example of FUJITSU SEMICONDUCTOR MCU Programmer when a crystal oscillator is used as a source oscillation clock and Table 1-2 available frequencies and communication baud rates.

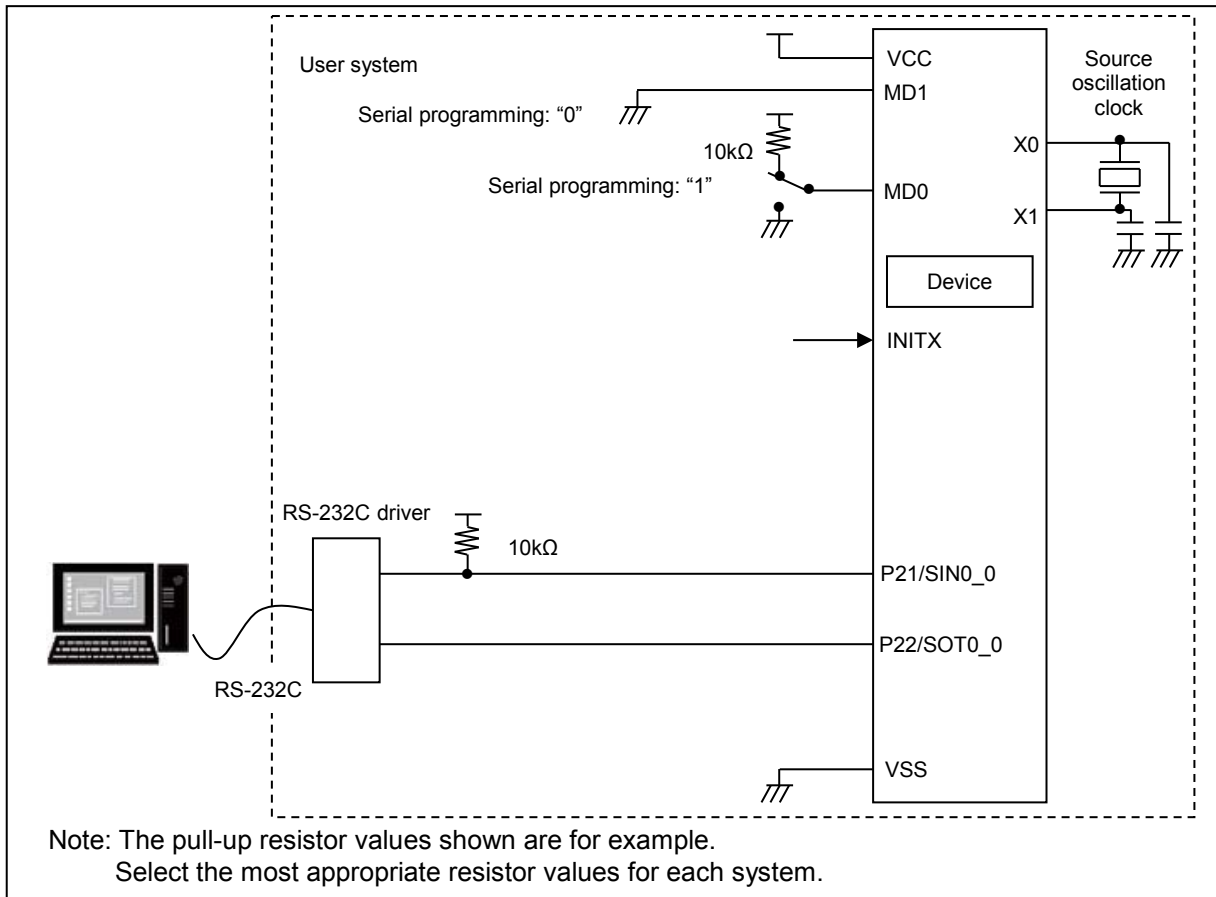Figure 1-2 Connection Example when Crystal Oscillator is Used

Note: The pull-up resistor values shown are for example.
Select the most appropriate resistor values for each system.

Table 1-2 Oscillating frequency and communication baud rate available for clock asynchronous serial communication

Source Oscillating Frequency	Communication Baud Rate
4MHz	9600bps
8MHz	19200bps
16MHz	38400bps
24MHz	57600bps
48MHz*	115200bps

*:Only for TYPE10 products

● **When built-in high-speed CR oscillator is used as a source oscillation clock**

Figure 1-3 shows a connection example of FUJITSU SEMICONDUCTOR MCU Programmer when a built-in high-speed CR oscillator is used as a source oscillation clock.
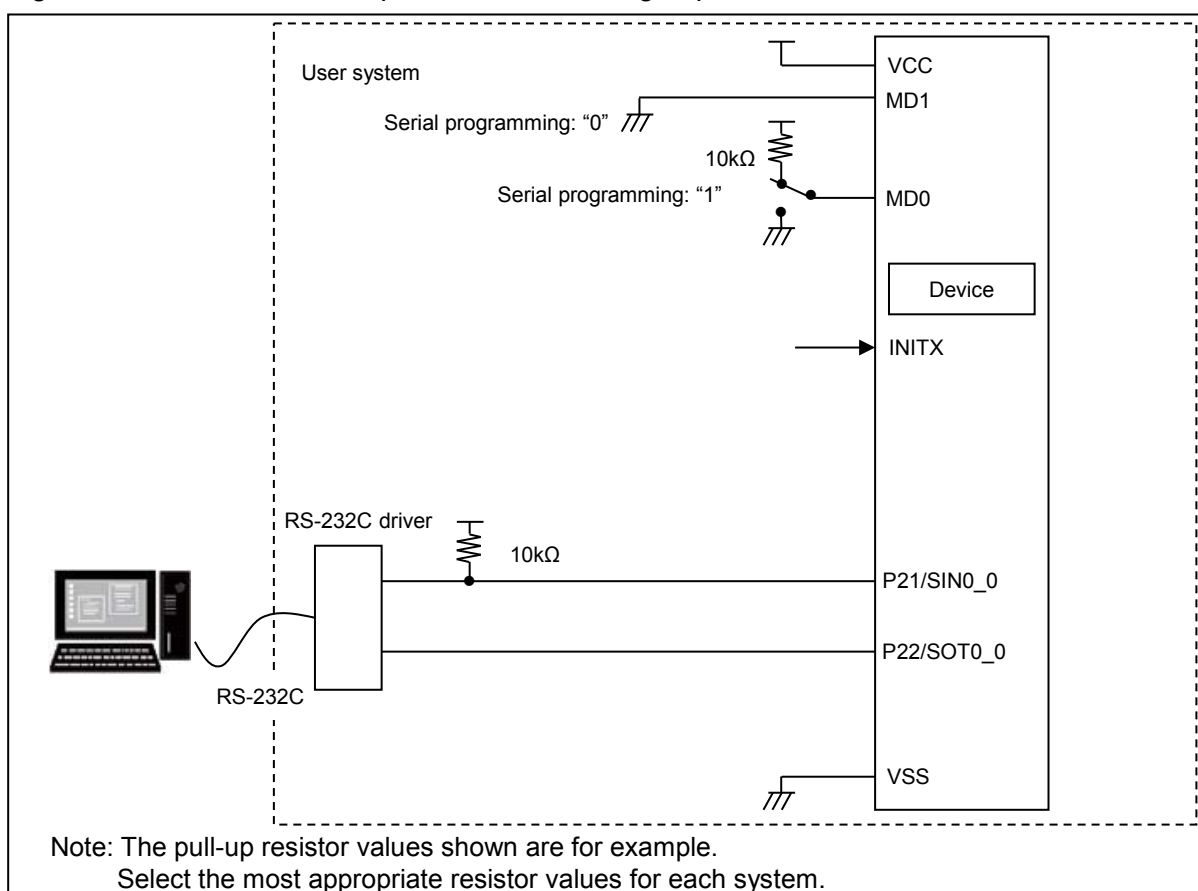
When neither crystal oscillator nor external clock is connected to X0/X1 pins, the built-in high-speed CR oscillator is connected for communication.

The communication baud rate is 9600[bps] when built-in high-speed CR oscillator is used

The following are the restrictions when built-in high-speed CR oscillator is used

- · Because the oscillation frequency of the built-in high-speed CR oscillator would fluctuate due to temperature and voltage change, the allowable baud rate error range might be exceeded.
- · For using the built-in high-speed CR oscillator, see "Built-in CR Oscillation Specifications" in "Data Sheet" of the product used.

Figure 1-3 Connection Example When Built-in High-speed CR Oscillator is used

Note: The pull-up resistor values shown are for example.
Select the most appropriate resistor values for each system.

1.2.   Pins Used

This section explains the used pins.

Table 1-3 Pins used for serial programming

Pins	Function	Supplement
MD0, MD1	Mode pin	Performing an external reset or turning on the power after setting MD0=H and MD1=L enters the serial programming mode. When attaching a pull-up or pull-down resistor, avoid long wiring.
X0, X1	Oscillation pin	See the "Data Sheet" of a product used for the source oscillation clock (main clock) frequencies that can be used in serial programming mode. (Restrictions apply to clock asynchronous communication. For details, see Table 1-2.) In UART communication mode, the write operation is available without main clock.
P22/SOT0_0	UART serial data output pin	When the communication mode is set to UART, this pin becomes a serial data output pin when communication begins after the serial programming mode is activated.
P21/SIN0_0	Clock synchronous/ asynchronous select pin/UART serial data input pin	Setting the input level of this pin to "H" until the start of communication enables the clock asynchronous communication mode, and setting it to "L" enables the clock synchronous communication mode. When the communication mode is set to UART, this pin can be used as a serial data input pin when communication begins after the serial programming mode is activated.
INITX	Reset pin	-
VCC	Power supply pin	For writing, supply power to the microcontroller from the user system.
VSS	GND pin	-

FUJITSU SEMICONDUCTOR • CONTROLLER MANUAL

32-BIT MICROCONTROLLER

FM3 Family

MB9A420L/120L/MB9B210J Series

FLASH PROGRAMMING MANUAL