Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	H8/300H
Core Size	16-Bit
Speed	16MHz
Connectivity	I²C, SCI, UART/USART
Peripherals	LVD, POR, PWM, WDT
Number of I/O	47
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	6K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b SAR; D/A 1x10b
Oscillator Type	External, Internal
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LFQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df36079lfzv

1.5 Pin Functions

Table 1.7 lists the Pin Functions.

Table 1.7 Pin Functions

Item	Pin Name	I/O	Description
Power supply input	VCC, VSS	—	Apply 1.8 V through 5.5 V to the VCC pin. Apply 0 V to the VSS pin.
Analog power supply input	AVCC, AVSS	—	Power supply input for the A/D converter. Connect a capacitor between pins AVCC and AVSS.
Reset input	$\overline{\text{RESET}}$	I	Applying a low level to this pin resets the MCU.
MODE	MODE	I	Connect this pin to the VCC pin via a resistor.
XIN clock input	XIN	I	I/O for the XIN clock generation circuit. Connect a ceramic resonator or a crystal oscillator between pins XIN and XOUT. (1) To use an external clock, input it to the XIN pin. P4_7 can be used as an I/O port at this time.
XIN clock output	XOUT	O	
$\overline{\text{INT}}$ interrupt input	$\overline{\text{INT0}}$ to $\overline{\text{INT3}}$	I	$\overline{\text{INT}}$ interrupt input.
Key input interrupt	$\overline{\text{KI0}}$ to $\overline{\text{KI3}}$	I	Key input interrupt input.
I/O ports	P1_0 to P1_7, P3_0 to P3_5, P3_7, P4_2, P4_5 to P4_7, PA_0	I/O	CMOS I/O ports. Each port has an I/O select direction register, enabling switching input and output for each port. For input ports other than PA_0, the presence or absence of a pull-up resistor can be selected by a program. P1_2 to P1_5, P3_3 to P3_5, and P3_7 can be used as LED drive ports.
Timer RJ2	TRJIO	I/O	Timer RJ2 I/O.
	TRJO	O	Timer RJ2 output.
Timer RB2	TRBO	O	Timer RB2 output.
Timer RC	TRCCLK	I	External clock input.
	TRCTRG	I	External trigger input.
	TRCIOA, TRCIOB, TRCIOC, TRCIOD	I/O	Timer RC I/O.
Serial interface	CLK0	I/O	Transfer clock I/O.
	RXD0	I	Serial data input.
	TXD0	O	Serial data output.
A/D converter	AN0 to AN4, AN7	I	Analog input for the A/D converter.
	$\overline{\text{ADTRG}}$	I	External trigger input for the A/D converter.
Comparator B	IVCMP1, IVCMP3	I	Analog voltage input for comparator B.
	IVREF1, IVREF3	I	Reference voltage input for comparator B.
	VCOUT1, VCOUT3	O	Comparison result output for comparator B.

Note:
1. Contact the oscillator manufacturer for oscillation characteristics.

7.2 Registers

Table 7.2 lists the Voltage Detection Circuit Register Configuration.

Table 7.2 Voltage Detection Circuit Register Configuration

Register Name	Symbol	After Reset	Address	Access Size
Voltage Monitor Circuit Edge Select Register	VCAC	00h	00058h	8
Voltage Detect Register 2	VCA2	(Note 1)	0005Ah	8
Voltage Detection 1 Level Select Register	VD1LS	00000111b	0005Bh	8
Voltage Monitor 0 Circuit Control Register	VW0C	(Note 1)	0005Ch	8
Voltage Monitor 1 Circuit Control Register	VW1C	10001010b	0005Dh	8

Note:
1. See the description of the individual registers.

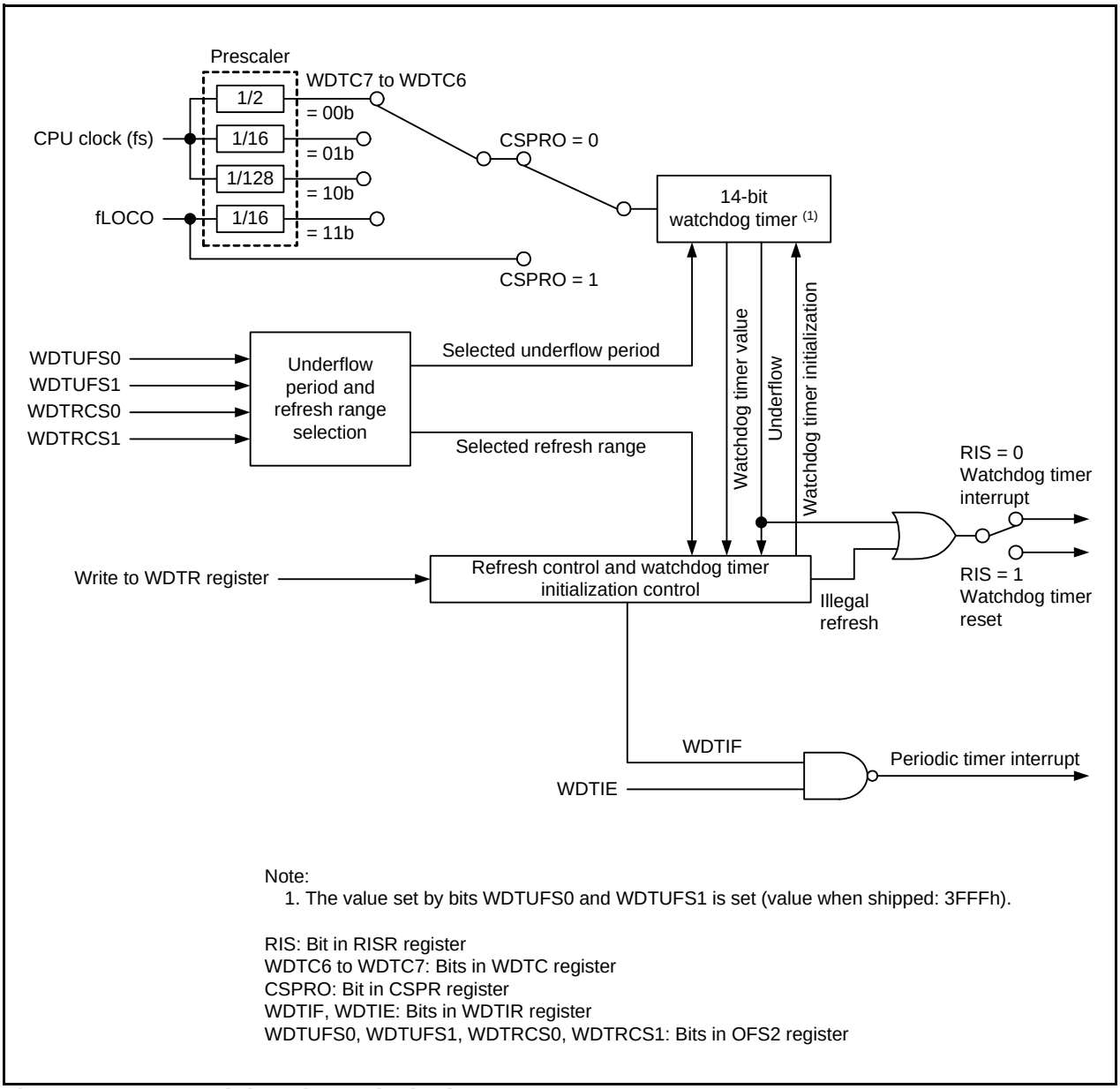
7.2.1 Voltage Monitor Circuit Edge Select Register (VCAC)

Address 00058h

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	—	—	—	—	—	—	VCAC1	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	—	Reserved	Set to 0.	R/W
b1	VCAC1	Voltage monitor 1 circuit edge select bit (1)	0: One-way edge 1: Two-way edge	R/W
b2	—	Reserved	Set to 0.	R/W
b3	—	Nothing is assigned. The write value must be 0. The read value is 0.		—
b4	—			
b5	—			
b6	—			
b7	—			

Note:
1. When the VCAC1 bit is 0 (one-way edge), the VW1C7 bit in the VW1C register can be used to select an interrupt generated when the voltage increases or decreases. Set the VCAC1 bit to 0 before setting the VW1C7 bit.



Note:
1. The value set by bits WDTUFS0 and WDTUFS1 is set (value when shipped: 3FFFh).

RIS: Bit in RISR register
WDTC6 to WDTC7: Bits in WDTC register
CSPRO: Bit in CSPR register
WDTIF, WDTIE: Bits in WDTIR register
WDTUFS0, WDTUFS1, WDTRCS0, WDTRCS1: Bits in OFS2 register

Figure 8.1 Watchdog Timer Block Diagram

9. Clock Generation Circuit

9.1 Overview

The following three circuits are included in the clock generation circuit:
• XIN clock oscillation circuit
• High-speed on-chip oscillator
• Low-speed on-chip oscillator
Table 9.1 lists the Clock Generation Circuit Specifications. Figure 9.1 shows the Clock Generation Circuit Block Diagram. Figure 9.2 shows the Supply of Peripheral Function Clocks. Table 9.2 lists the Clock Generation Circuit Pin Configuration.

Table 9.1 Clock Generation Circuit Specifications

Item	XIN Clock Oscillation Circuit	High-Speed On-Chip Oscillator	Low-Speed On-Chip Oscillator
Clock frequency	0 MHz to 20 MHz (2 MHz to 20 MHz when an oscillator is used)	Approx. 20 MHz	Approx. 125 kHz
Connectable oscillator	• Ceramic resonator • Crystal oscillator	—	—
Oscillator connect pins	XIN, XOUT (1)	—	—
Oscillation start and stop	Usable	Usable	Usable
State after reset	Stopped	Stopped	Oscillates
Others	• An externally generated clock can be input. • A feed-back resistor is included (connected or not connected can be selected).	The system clock can be output from P4_7.	The system clock can be output from P4_7.

Note:
1. When the on-chip oscillator clock instead of the XIN clock oscillation circuit is used as the CPU clock, these pins can be used as P4_6 and P4_7.

The system base clock when returning from wait mode by a peripheral interrupt is the clock set by the WAITRS bit in the CKRSCR register. At this time, bits PHISSEL0 to PHISSEL2 in the SCKCR register and the SCKSEL bit in the CKSTPR register are automatically changed according to bits PHISRS and WAITRS.
If the system base clock when returning is different from the clock used immediately before entering wait mode, a period until the clock supply (oscillation stabilization time) is generated automatically. If the system base clock when returning is the high-speed on-chip oscillator clock, oscillation is started when an interrupt request is generated. If the system base clock is the XIN clock, set pins P4_6 and P4_7 to XIN oscillation by a program to start oscillation before entering wait mode.
Depending on the clock to be used, set appropriate values for oscillation stabilization time using bits CKST0 to CKST3 in the CKRSCR register. It is unnecessary to generate a wait time by a program. When returning from wait mode using the same clock as the one used immediately before entering the mode, no oscillation stabilization time is generated.

FMR0 Register FMSTP Bit	FMR1 Register WTFMSTP Bit	VCA2 Register LPE Bit	Internal Power Stabilization Time (T0)	Time until Flash Memory Activation (T1)	Clock Stabilization Time (T2)	Time until CPU Clock Supply (T3)	Time for Interrupt Sequence (T4)	Remarks
0 (flash memory operates)	1 (flash memory is stopped in wait mode)	0 (low-power-consumption wait mode disabled)	0 µs	60 µs (max.)	Set by bits CKST0 to CKST3 in the CKRSCR register if the clocks are switched when returning from wait mode 0 if the clocks are not switched when returning from wait mode	CPU clock period × 2 cycles	CPU clock period × 20 cycles	The total on the left amounts to the time from wait mode until execution of an interrupt routine.
		1 (low-power-consumption wait mode enabled)	100 µs (max.)					
1 (flash memory is stopped)	—	0 (low-power-consumption wait mode disabled)	0 µs	— (Flash memory is not activated)				
		1 (low-power-consumption wait mode enabled)	100 µs (max.)					
0 (flash memory operates)	0 (flash memory operates in wait mode)	0 (low-power-consumption wait mode disabled)	0 µs					
		1 (low-power-consumption wait mode enabled)	—					Setting prohibited

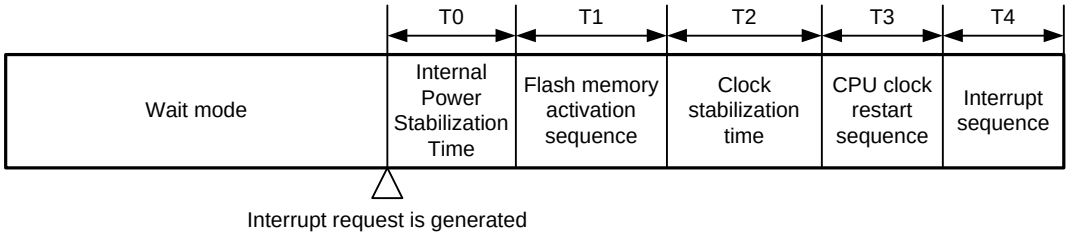


Figure 10.2 Sequence from Wait Mode to Interrupt Routine Execution after WAIT instruction is Executed

11.2.9 External Interrupt Flag Register (IRR3)

Address 00053h

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	—	—	IRKI	—	IRI3	IRI2	IRI1	IRI0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	IRI0	$\overline{\text{INT0}}$ interrupt request flag	0: No interrupt requested 1: Interrupt requested	R/W
b1	IRI1	$\overline{\text{INT1}}$ interrupt request flag		R/W
b2	IRI2	$\overline{\text{INT2}}$ interrupt request flag		R/W
b3	IRI3	$\overline{\text{INT3}}$ interrupt request flag		R/W
b4	—	Reserved	Set to 0.	R/W
b5	IRKI	Key input interrupt request flag	0: No interrupt requested 1: Interrupt requested	R/W
b6	—	Nothing is assigned. The write value must be 0. The read value is 0.		—
b7	—			

IRI0 Bit ($\overline{\text{INT0}}$ interrupt request flag)

Writing 0 after reading the value 1 sets the IRI0 bit to 0. This bit is also automatically set to 0 when the corresponding interrupt ($\overline{\text{INT0}}$) is acknowledged.

IRI1 Bit ($\overline{\text{INT1}}$ interrupt request flag)

Writing 0 after reading the value 1 sets the IRI1 bit to 0. This bit is also automatically set to 0 when the corresponding interrupt ($\overline{\text{INT1}}$) is acknowledged.

IRI2 Bit ($\overline{\text{INT2}}$ interrupt request flag)

Writing 0 after reading the value 1 sets the IRI2 bit to 0. This bit is also automatically set to 0 when the corresponding interrupt ($\overline{\text{INT2}}$) is acknowledged.

IRI3 Bit ($\overline{\text{INT3}}$ interrupt request flag)

Writing 0 after reading the value 1 sets the IRI3 bit to 0. This bit is also automatically set to 0 when the corresponding interrupt ($\overline{\text{INT3}}$) is acknowledged.

IRKI Bit (Key input interrupt request flag)

Writing 0 after reading the value 1 sets the IRKI bit to 0. This bit is also automatically set to 0 when the corresponding interrupt (key input) is acknowledged.

The interrupt priority level register must be rewritten only while no interrupt requests corresponding to that register are generated. See **11.9.7 Changing Interrupt Priority Levels and Flag Registers**.

12.3.8 Port 1 Function Mapping Expansion Register (PMH1E)

Address 000D1h

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	—	—	—	—	—	P15SEL2	—	P14SEL2
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	P14SEL2	The P1_4 pin function is selected in conjunction with bits P14SEL0 to P14SEL1 in the PMH1 register. For details, see **12.3.7 Port 1 Function Mapping Register 1 (PMH1)**.		R/W
b1	—	Nothing is assigned. The write value must be 0. The read value is 0.		—
b2	P15SEL2	The P1_5 pin function is selected in conjunction with bits P15SEL0 to P15SEL1 in the PMH1 register. For details, see **12.3.7 Port 1 Function Mapping Register 1 (PMH1)**.		R/W
b3	—	Nothing is assigned. The write value must be 0. The read value is 0.		—
b4	—			
b5	—			
b6	—			
b7	—			

The PMH1E register is used to select the port 1 function in conjunction with registers PML1 and PMH1.

12.4.7 Port 3 Function Mapping Register 1 (PMH3)

Address 000CDh

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	P37SEL1	P37SEL0	—	—	P35SEL1	P35SEL0	P34SEL1	P34SEL0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	P34SEL0	Port P3_4 function select bits	b1 b0 0 0: I/O port or IVREF3 input 0 1: TRCIOC 1 0: INT2 1 1: Do not set.	R/W
b1	P34SEL1			R/W
b2	P35SEL0	Port P3_5 function select bits	b3 b2 0 0: I/O port 0 1: TRCIOD 1 0: KI2 1 1: VCOUT3	R/W
b3	P35SEL1			R/W
b4	—	Nothing is assigned. The write value must be 0. The read value is 0.		—
b5	—			
b6	P37SEL0	Port P3_7 function select bits	b7 b6 0 0: I/O port 0 1: ADTRG 1 0: TRJO 1 1: TRCIOD	R/W
b7	P37SEL1			R/W

The PMH3 register is used to select the functions of pins P3_4, P3_5, and P3_7.

14.6 Interrupt Request

When the TRBIF bit in the TRBIR register is 1 (interrupt requested) and the TRBIE bit is 1 (interrupt enabled), an interrupt request is generated to the CPU. The conditions for setting the TRBIF bit to 1 differ depending on the mode. See the descriptions of the TRBIF bit and individual modes.

14.7 $\overline{\text{INT0}}$ Input Trigger Selection

In programmable one-shot and programmable wait one-shot generation modes, when 1 (one-shot count is started) is written to the TOSST bit in the TRBCR register or a trigger is input to the $\overline{\text{INT0}}$ pin with the TCSTF bit in the TRBCR register set to 1 (count is in progress), one-shot operation is started.

When using the trigger input from the $\overline{\text{INT0}}$ pin, make the following settings beforehand.

(1) Set the port mapping register to set port P1_4 or P4_5 as the $\overline{\text{INT0}}$ pin.
(2) Set bits INT0F0 to INT0F1 in the INTF0 register to select the digital filter sampling clock for the $\overline{\text{INT0}}$ pin.
(3) Set the INT0EN bit in the INTEN register to 1 (enabled) to enable an interrupt.
(4) Set the INOSEG bit in the TRBIOC register to select the falling or rising edge.
(5) Set the INOSTG bit in the TRBIOC register to 1 (one-shot trigger to $\overline{\text{INT0}}$ pin enabled).

When an interrupt request is generated by the trigger input from the $\overline{\text{INT0}}$ pin, note the following:

- Set bits INT0SA to INT0SB in the ISCR0 register to select the falling edge, rising edge, or two-way edge for the interrupt.

Even if a one-shot trigger is generated while the TOSSTF bit in the TRBOCR is 1 (one-shot is operating (including wait period)), timer RB2 operation is not influenced, but the IRI0 bit in the IRR3 register is changed.
For details on interrupts, see **11. Interrupts**.

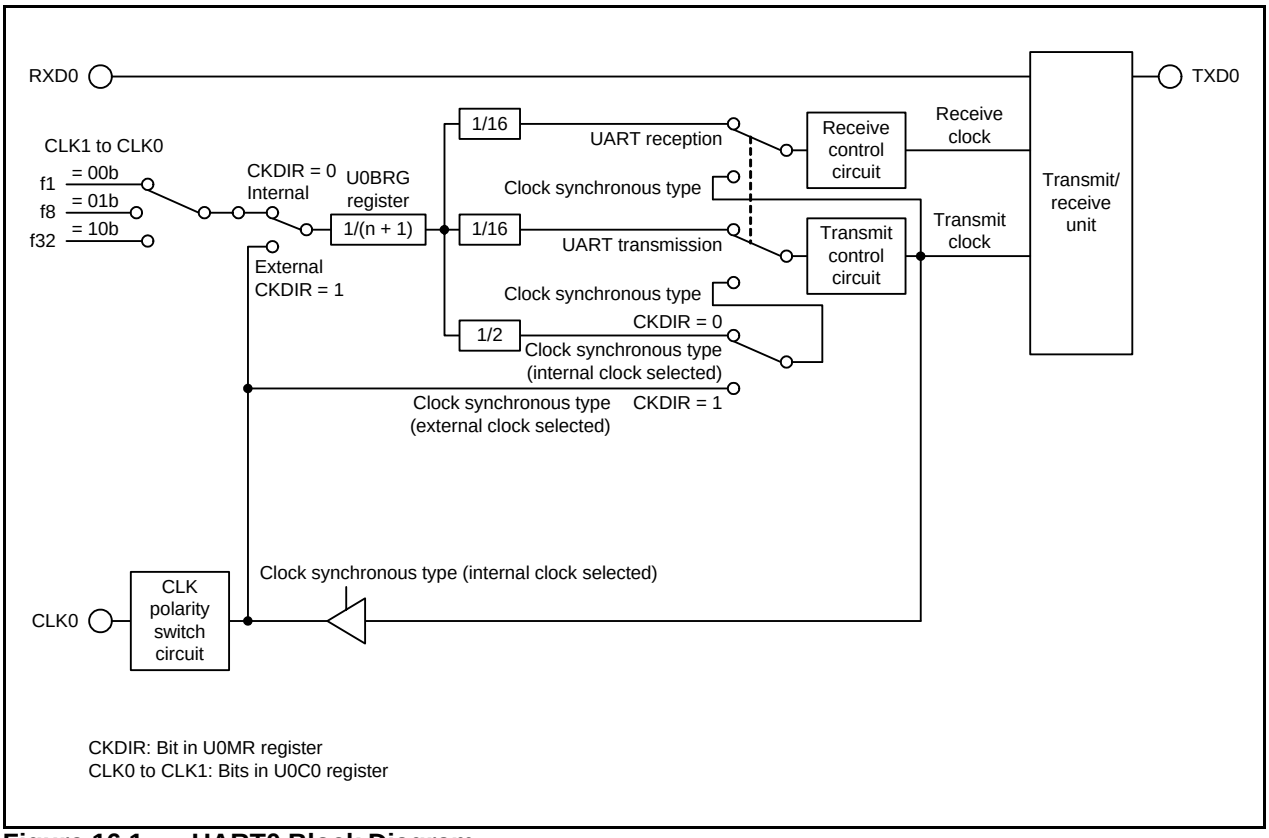


Figure 16.1 UART0 Block Diagram

18.3.2 Comparator Bi (i = 1 or 3) Setting Procedure and Operation Example

Comparator B1 and comparator B3 operate independently of each other.
Table 18.4 lists the Procedure for Setting Registers Associated with Comparator B.

Table 18.4 Procedure for Setting Registers Associated with Comparator B

Step	Register	Bit	Setting Value
1	Select the functions of pins IVCMPi and IVREFi. For the settings, see **12. I/O Ports**.		
2	WCBiINTR	WCBiF1 to WCBiF0	• Enable or disable the digital filter. • Select the sampling frequency.
3	WCMPR	WCB1M0	1 (operation enabled)
		WCB3M0	
4	Wait for the comparator stabilization time (100 μs max.).		
5	ILVL2	ILVL21 to ILVL20	When an interrupt is used: Select the interrupt priority level for comparator B1.
		ILVL25 to ILVL24	When an interrupt is used: Select the interrupt priority level for comparator B3.
6	WCBiINTR	WCBiS1 to WCBiS0	When an interrupt is used: Select the input polarity.
7	WCBiINTR	WCBiF	0 (no interrupt requested)
8	WCBiINTR	WCBiINTEN	When an interrupt is used: 1 (interrupt enabled)

i = 1 or 3

Figure 18.3 shows an Example of Comparator Bi (i = 1 or 3) Operation.
When the analog input voltage is higher than the reference input voltage, the WCBiOUT bit in the WCMPR register is set to 1. When the analog input voltage is lower than the reference input voltage, the WCBiOUT bit is set to 0.
When a comparator Bi interrupt (i = 1 or 3) is used, set the WCBiINTEN bit in the WCBiINTR register to 1 (interrupt enabled). If the comparison result changes at this time, a comparator Bi interrupt request is generated. For details on interrupts, see **11. Interrupts**.

19. Flash Memory

The flash memory supports two rewrite modes: CPU rewrite mode and standard serial I/O mode.

19.1 Overview

Table 19.1 lists the Flash Memory Specifications (see **Tables 1.3** and **1.4 Specifications** for items not listed in Table 19.1). Table 19.2 outlines Flash Memory Rewrite Mode.

Table 19.1 Flash Memory Specifications

Item		Specification
Flash memory operating modes		2 modes (CPU rewrite and standard serial I/O modes)
Erase block division		See **Figure 19.1 Flash Memory Block Diagram**.
Programming method		Byte units
Erase method		Block erase
Program/erase control method (1)		Program/erase control by software commands
Rewrite control method	Blocks 1 and 2 (program ROM) (2)	Rewrite protect control in block units by lock bits
	Blocks A and B (data flash)	Individual rewrite control on blocks A and B by bits FMR16 to FMR17 in the FMR1 register
Number of commands		6 commands
Program/erase endurance (3)	Blocks 1 and 2 (program ROM) (2)	10,000 times
	Blocks A and B (data flash)	
ID code check function (4)		Standard serial I/O mode supported

Notes:
1. When programming/erasing the program ROM and the data flash, use a VCC supply voltage in the range of 1.8 V to 5.5 V.
2. The number of blocks and their division differ depending on products. For details, see **Figure 19.1 Flash Memory Block Diagram**.
3. Definition of program/erase endurance
   The number of program/erase cycles is defined on a per-block basis.
   If the number of cycles is 10,000, each block can be erased 10,000 times.
   For example, if 1,024 cycles of 1-byte-write are performed to different addresses in 1 Kbyte of block A, and then the block is erased, the number of cycles is counted as one. When rewrites are performed 100 or more times, the actual erase count can be reduced by executing program operations in such a way that all blank areas are used up before performing an erase operation. Avoid rewriting only particular blocks and average out the number of programming/erasure of the blocks. It is also advisable to retain data on the number of erasure of each block and limit the number to a certain extent.
4. For details on the ID code check function, see **19.3 ID Code Check Function**.

Table 19.2 Flash Memory Rewrite Mode

Flash Memory Rewrite Mode	CPU Rewrite Mode	Standard Serial I/O Mode
Function	The user ROM area is rewritten by executing software commands from the CPU.	The user ROM area is rewritten using a dedicated serial programmer.
Rewritable area	User ROM	User ROM
Rewrite programs	User program	Standard boot program

19.5.5 Flash Memory Refresh Control Register (FREFR)

Address 001ADh

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	—	—	REF5	REF4	REF3	REF2	REF1	REF0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	REF0	Periodic refresh interval control bits	Value in the FREFR register = fs/10^3 (the result value is expressed as an integer) If the clock source for the CPU clock (fs) is the low-speed on-chip oscillator, it is taken to be the minimum fLOCO value (60 kHz). Ex: Value set in the FREFR register when the CPU clock (fs) is set to 12.5 kHz: (12.5 × 10^3/10^3) = 12 = 001100b	R/W
b1	REF1			R/W
b2	REF2			R/W
b3	REF3			R/W
b4	REF4			R/W
b5	REF5			R/W
b6	—	Nothing is assigned. The write value must be 0. The read value is 0.		—
b7	—			

The FREFR register is used to control the interval between refresh operations when the FMR27 bit in the FMR2 register is 1 (low-current-consumption read mode enabled). First set the FMR27 bit in the FMR2 register to 0 (low-current-consumption read mode disabled) to set the value in the register. After that, set the FMR27 bit to 1 (low-current-consumption read mode enabled).

19.6.6 Software Commands

The software commands are described below. Commands must be read or written and data in 8-bit units. Do not input any command other than those listed in the table below.

Table 19.8 Software Commands

Command	First Command			Second Command (1)		
	Mode	Address	Data	Mode	Address	Data
Read array	Write	x	FFh			
Clear status register	Write	x	50h			
Program	Write	WA	40h	Write	WA	WD
Block erase	Write	x	20h	Write	BA	D0h
Lock bit program	Write	BT	77h	Write	BT	D0h
Read lock bit status	Write	x	71h	Write	BT	D0h
Block blank check	Write	x	25h	Write	BA	D0h

WA: Write address
WD: Write data
BA: Any address in the block
BT: Start address in the block
Block 2 → 0E000h
Block 1 → 0F000h
x: Any address in the user ROM area
Note:
1. For block erase, lock bit program, read lock bit status, and block blank check commands, if FFh is written as the second command, the command code written as the first command becomes invalid. A command sequence error does not occur.

The data flash does not have a lock bit, so the lock bit program and the read lock bit status commands are handled as illegal.

19.6.6.1 Read Array

This command is used to read the flash memory.
When FFh is written as the first command, the MCU enters read array mode. When the read address is input in the following cycles, the content of the specified address can be read in 8-bit units.
Since read array mode is retained until another command is written, the contents of multiple addresses can be read continuously.
In addition, after a reset, the MCU enters read array mode after a program, block erase, block blank check, read lock bit status, or clear status register command, or after entering suspend.

19.6.6.2 Clear Status Register

The clear status register command is used to set bits FST4 and FST5 in the FST register to 0.
When 50h is written as the first command, bits FST4 and FST5 in the FST register are set to 0.

19.6.6.3 Program

This command writes data to the flash memory in 1-byte units.
When 40h is written as the first command and data is written to the write address with the second command, auto-programming (a data program and verify operation) starts. The address value for the first command must be the same address as the write address specified with the second command.
The FST7 bit in the FST register can be used to confirm whether auto-programming has completed. The FST7 bit is set to 0 during auto-programming and is set to 1 when auto-programming completes.
After auto-programming has completed, the auto-program result can be confirmed by the FST4 bit in the FST register (see **19.6.7 Full Status Check**).
Do not write additions to the already programmed addresses.
For each block in the program ROM, the program command can be disabled using the lock bit.
When the FMR16 bit in the FMR1 register is 1 (rewrite disabled), the program command for block A of data flash is not accepted. When the FMR17 bit is 1 (rewrite disabled), the program command for block B is not accepted.
Figure 19.8 shows the Program Flowchart (Flash Ready Status Interrupt Disabled and Suspend Disabled). Figure 19.9 shows the Program Flowchart in EW0 Mode (Flash Ready Status Interrupt Disabled and Suspend Enabled). Figure 19.10 shows the Program Flowchart in EW0 Mode (Flash Ready Status Interrupt Enabled and Suspend Enabled). Figure 19.11 shows the Program Flowchart in EW1 Mode (Flash Ready Status Interrupt Disabled and Suspend Enabled).

In EW1 mode, do not execute this command for any address where the rewrite control program is allocated.

When the RDYSTIE bit in the FMR0 register is 1 (flash ready status interrupt enabled), a flash ready status interrupt is generated when auto-programming is completed.
When the RDYSTIE bit in the FMR0 register is 1 (flash ready status interrupt enabled) and the FMR20 bit in the FMR2 register is 1 (suspend enabled), a flash ready status interrupt is generated when the FMR21 bit is set to 1 (suspend request) and auto-programming is suspended. The result can be confirmed by reading the FST register in the interrupt routine.

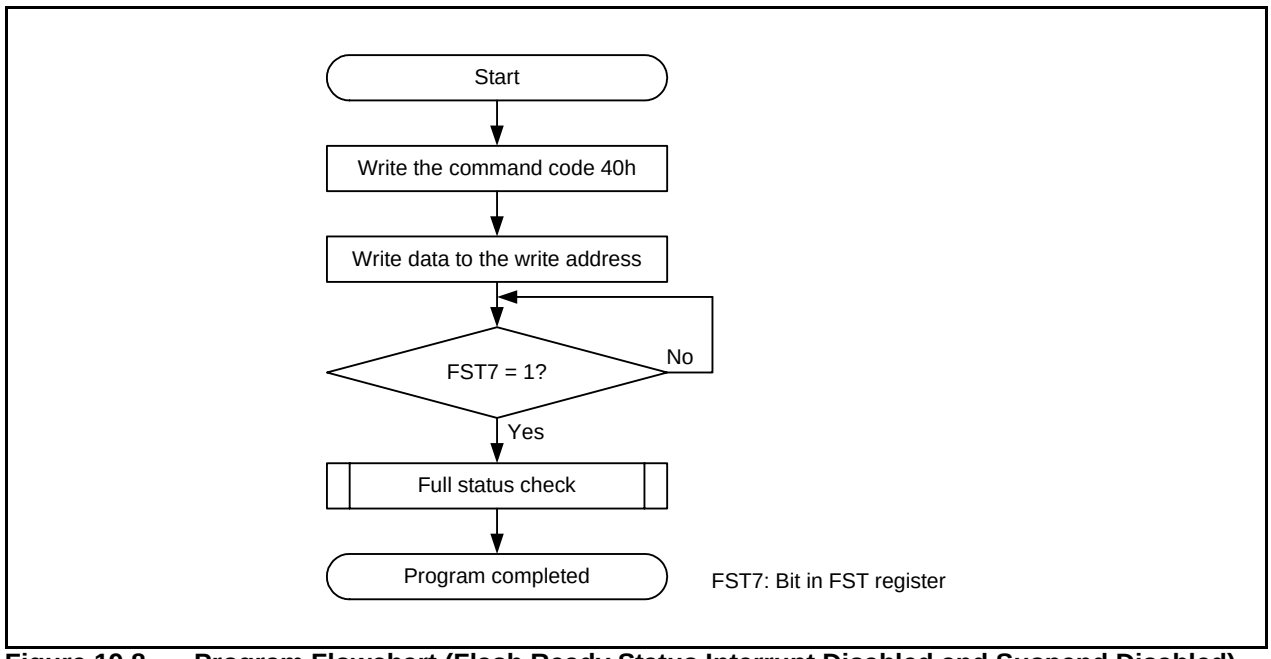


Figure 19.8 Program Flowchart (Flash Ready Status Interrupt Disabled and Suspend Disabled)

19.6.6.6 Read Lock Bit Status Command

This command is used to read the lock bit status for any block in the program ROM area.
When 71h written as the first command and D0h is written to the start address in the block with the second command, the lock bit status in the specified block is stored in the FST2 bit in the FST register. Read the FST2 bit after the FST7 bit in the FST register has changed to 1 (ready).
Figure 19.17 shows the Read Lock Bit Status Flowchart.

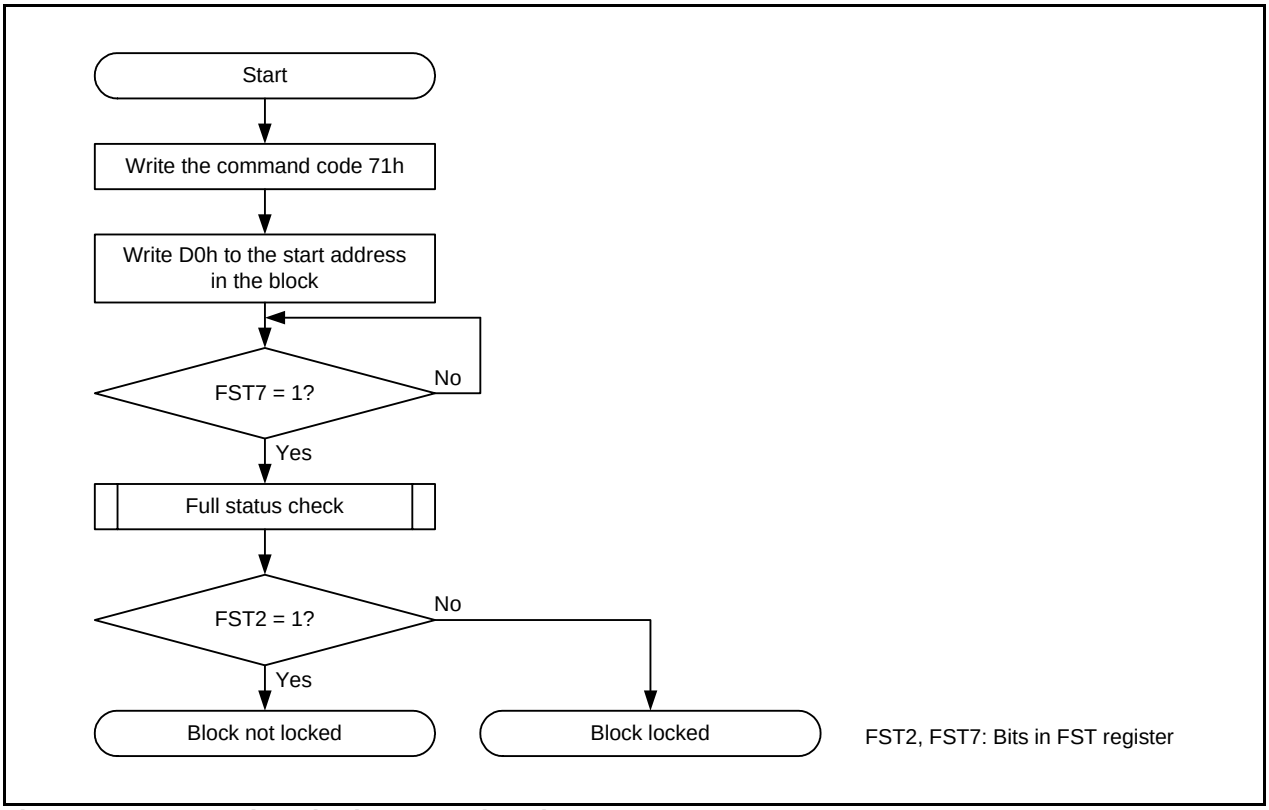


Figure 19.17 Read Lock Bit Status Flowchart

19.8.2 CPU Rewrite Mode

19.8.2.1 Prohibited Instructions

The following instructions cannot be used while the program ROM area is being rewritten in EW0 mode because they reference data in the flash memory:
UND, INTO, and BRK

19.8.2.2 Interrupts

Tables 19.12 and 19.13 list the Interrupt Handling during CPU Rewrite Operation.

Table 19.12 Interrupt Handling during CPU Rewrite Operation (EW0 Mode)

Interrupt Type	Data Flash/Program ROM	
	Suspend Enabled (FMR20 = 1)	Suspend Disabled (FMR20 = 0)
Maskable interrupt	When an interrupt request is acknowledged, interrupt handling is executed. (The interrupt vector is allocated in the RAM) The suspend state can be entered by either of the following: (1) When the FMR22 bit is 1 (suspend request enabled by interrupt request), the FMR21 bit is automatically set to 1 (suspend request). The flash memory suspends auto-erase or auto-programming after td(SR-SUS). (2) When the FMR22 bit is 0 (suspend request disabled by interrupt request) and suspend is required, set the FMR21 bit to 1 (suspend request) in the interrupt handling. The flash memory suspends auto-erase or auto-programming after td(SR-SUS). While auto-erase is suspended, auto-programming and reading can be executed for any block other than the blocks being auto-erased. While auto-programming is suspended, any block other than the blocks being auto-programmed can be read. Auto-erase can be restarted by setting the FMR21 bit to 0 (restart).	Interrupt handling is executed with auto-erase or auto-programming executed (The interrupt vector is allocated in the RAM)
Address match	Do not use during auto-erasing or auto-programming.	
UND, INTO, and BRK instructions		
Single-step		
Watchdog timer	When an interrupt request is acknowledged, auto-erase or auto-programming is forcibly stopped immediately and the flash memory is reset. After the specified period, the flash memory is restarted before interrupt handling is started. Since auto-erase or auto-programming is forcibly stopped, the correct values may not be read from the block being auto-erased or the address being auto-programmed. After the flash memory is restarted, execute auto-erase again and verify it complete normally. The watchdog timer does not stop while the command is executing, so interrupt requests may be generated. Initialize the watchdog timer periodically using the erase-suspend function. Since the flash memory control registers are initialized in this case, these registers must be set again. (1)	
Oscillation stop detection		
Voltage monitor 1		

FMR20, FMR21, FMR22: Bits in FMR2 register
Note:
1. Registers FMR0, FMR1, and FMR2 are initialized if a watchdog timer, oscillation stop detection, or voltage monitor 1 interrupt is generated while the flash memory is busy.
When the FMR01 bit in the FMR0 register is 1 (CPU rewrite mode enabled) and the FMSTP bit is 1 (flash memory is stopped), registers FMR0, FMR1, and FMR2 are initialized if a watchdog timer, oscillation stop detection, or voltage monitor 1 interrupt is generated.

19.8.3 Notes on Flash Memory Stop and Operation Transition

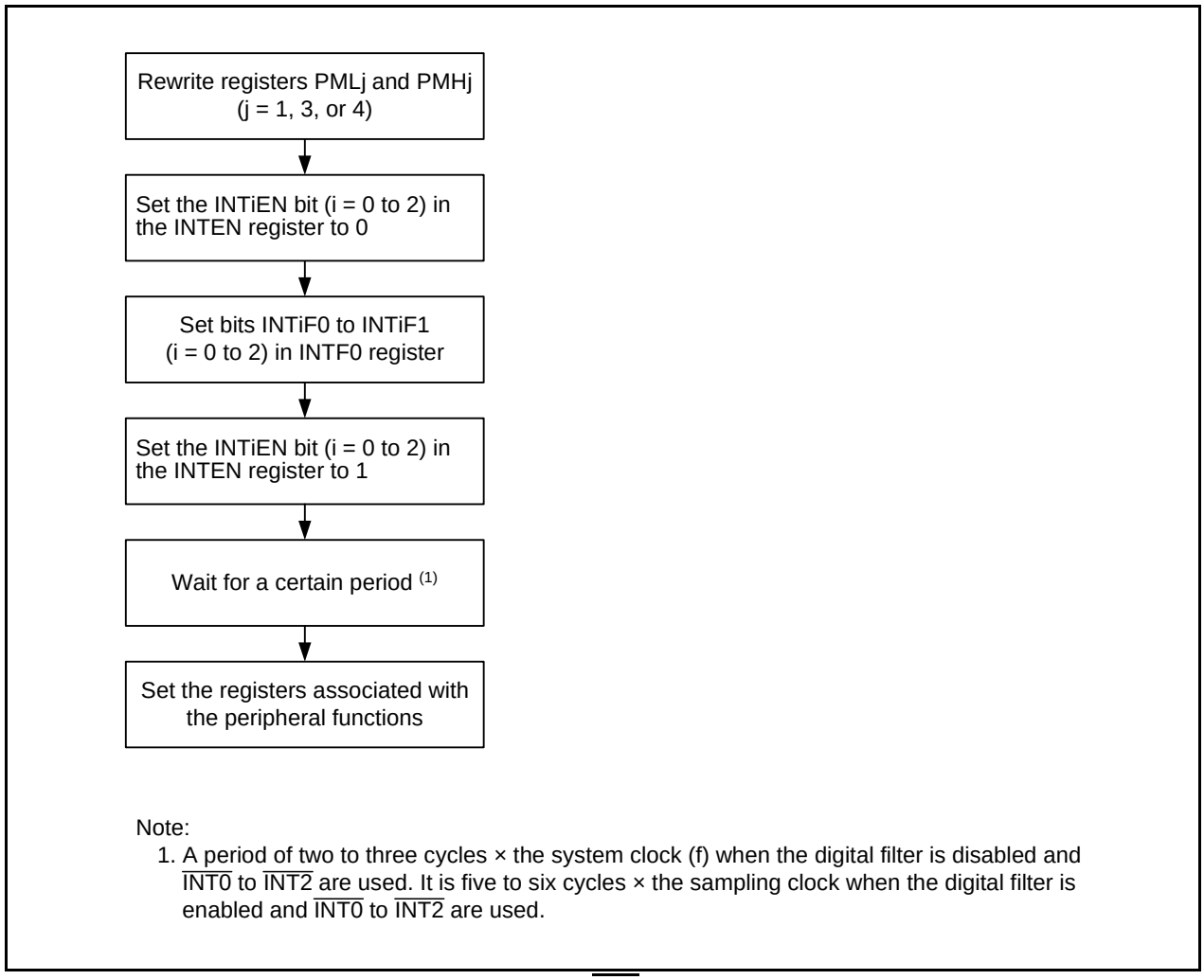
(1) Do not enter stop mode while the FMSTP bit is 1 (the flash memory is stopped).
(2) Do not enter wait mode while the FMSTP bit is 1 (the flash memory is stopped) and the WTFMSTP bit is 1 (the flash memory is stopped in wait mode).
(3) Do not enter flash memory stop state for 42 μs after entering from flash memory stop state to flash memory operation state. And do not rewrite the LOCODIS bit in the OCOCR register for 42 μs.

Conditions when entering flash memory operation state from flash memory stop state.
• Set the FMSTP bit to 0 (the flash memory operates).
• Return from wait mode while the WTFMSTP bit is 1 (the flash memory is stopped in wait mode).
• Return from stop mode.

Conditions when entering flash memory stop state from flash memory operation state.
• Set the FMSTP bit to 1 (the flash memory is stopped).
• Enter wait mode while the WTFMSTP bit is 1 (the flash memory is stopped in wait mode).
• Enter stop mode.

21.5.6 Setting Procedure When $\overline{\text{INTi}}$ Input Filter (i = 0 to 2) is Used for Peripheral Functions

Figure 21.3 shows the Register Setting Procedure When $\overline{\text{INTi}}$ Input Filter (i = 0 to 2) is Used for Peripheral Functions (Timer RJ2, Timer RB2, and Timer RC).



Note:
1. A period of two to three cycles × the system clock (f) when the digital filter is disabled and $\overline{\text{INT0}}$ to $\overline{\text{INT2}}$ are used. It is five to six cycles × the sampling clock when the digital filter is enabled and $\overline{\text{INT0}}$ to $\overline{\text{INT2}}$ are used.

Figure 21.3 Register Setting Procedure When $\overline{\text{INTi}}$ Input Filter (i = 0 to 2) is Used for Peripheral Functions (Timer RJ2, Timer RB2, and Timer RC)

21.12.3 Notes on Flash Memory Stop and Operation Transition

(1) Do not enter stop mode while the FMSTP bit is 1 (the flash memory is stopped).
(2) Do not enter wait mode while the FMSTP bit is 1 (the flash memory is stopped) and the WTFMSTP bit is 1 (the flash memory is stopped in wait mode).
(3) Do not enter flash memory stop state for 42 μs after entering from flash memory stop state to flash memory operation state. And do not rewrite the LOCODIS bit in the OCOCR register for 42 μs.

Conditions when entering flash memory operation state from flash memory stop state.
• Set the FMSTP bit to 0 (the flash memory operates).
• Return from wait mode while the WTFMSTP bit is 1 (the flash memory is stopped in wait mode).
• Return from stop mode.

Conditions when entering flash memory stop state from flash memory operation state.
• Set the FMSTP bit to 1 (the flash memory is stopped).
• Enter wait mode while the WTFMSTP bit is 1 (the flash memory is stopped in wait mode).
• Enter stop mode.