Renesas Electronics America Inc - UPD70F3806GB(R)-GAH-AX Datasheet

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	V850ES
Core Size	32-Bit Single-Core
Speed	20MHz
Connectivity	CSI, EBI/EMI, I²C, UART/USART
Peripherals	DMA, LVD, PWM, WDT
Number of I/O	50
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.2V ~ 3.6V
Data Converters	A/D 10x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd70f3806gb-r-gah-ax

Pin functions

ADTRG:	A/D trigger input	$\overline{\text{RESET}}$:	Reset
ANI0 to ANI9:	Analog input	RTC1HZ,	Real-time Counter Clock Output
ANO0:	Analog output	RTCCL, RTCDIV	
ASCKA0:	Asynchronous serial clock	RTP00 to RTP05:	Real-time output port
AV_{REF0}, AV_{REF1}:	Analog reference voltage	RXDA0 to RXDA2:	Receive data
AV_{SS}:	Analog V_{SS}	$\overline{\text{SCKB0}}$ to $\overline{\text{SCKB4}}$:	Serial clock
DCK:	Debug clock	SCL00 to SCL02:	Serial clock
DDI:	Debug data input	SDA00 to SDA02:	Serial data
DDO:	Debug data output	SIB0 to SIB4:	Serial input
DMS:	Debug mode select	SOB0 to SOB4:	Serial output
$\overline{\text{DRST}}$:	Debug reset	TIP00, TIP01,	Timer input
EV_{DD}:	Power supply for external pin	TIP10, TIP11,	
EV_{SS}:	Ground for external pin	TIP20, TIP21,	
FLMD0, FLMD1:	Flash programming mode	TIP31,	
INTP0 to INTP7:	External interrupt input	TIP40, TIP41,	
KR0 to KR7:	Key return	TIP50, TIP51,	
NMI:	Non-maskable interrupt request	TIQ00 to TIQ03:	
P02 to P06:	Port 0	TOP00, TOP01,	Timer output
P10:	Port 1	TOP10, TOP11,	
P30 to P35,		TOP20, TOP21,	
P38, P39:	Port 3	TOP31,	
P40 to P42:	Port 4	TOP40, TOP41,	
P50 to P55:	Port 5	TOP50, TOP51,	
P70 to P79:	Port 7	TOQ00 to TOQ03:	Transmit data
P90 to P94,		TXDA0 to TXDA2:	
P96 to P915:	Port 9	V_{DD}:	Power supply
PCM0:	Port CM	V_{SS}:	Ground
PDL15:	Port DL	X1, X2:	Crystal for main clock
REGC:	Regulator control	XT1, XT2:	Crystal for subclock

(1) Interrupt status saving registers (EIPC and EIPSW)

EIPC and EIPSW are used to save the status when an interrupt occurs.

If a software exception or a maskable interrupt occurs, the contents of the program counter (PC) are saved to EIPC, and the contents of the program status word (PSW) are saved to EIPSW (these contents are saved to the NMI status saving registers (FEPC and FEPSW) if a non-maskable interrupt occurs).

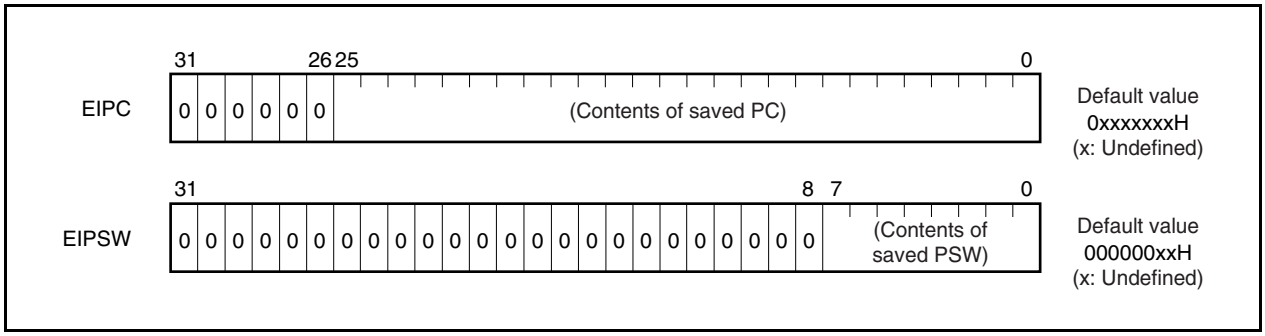
The address of the instruction next to the instruction under execution, except some instructions (see **19.8 Periods in Which Interrupts Are Not Acknowledged by CPU**), is saved to EIPC when a software exception or a maskable interrupt occurs.

The current contents of the PSW are saved to EIPSW.

Because only one set of interrupt status saving registers is available, the contents of these registers must be saved by program when multiple interrupts are enabled.

Bits 31 to 26 of EIPC and bits 31 to 8 of EIPSW are reserved for future function expansion (these bits are always fixed to 0).

The value of EIPC is restored to the PC and the value of EIPSW to the PSW by the RETI instruction.



PFCE03	PFC03	Specification of P03 pin alternate function
0	0	INTP0 input
0	1	ADTRG input
1	0	Setting prohibited
1	1	RTC1HZ output

(7) Port 0 function register (PF0)

(a) V850ES/JC3-L (40-pin), V850ES/JC3-L (48-pin)

After reset: 00H R/W Address: FFFFFC60H

	7	6	5	4	3	2	1	0
PF0	0	0	PF05	0	PF03	PF02	0	0

PF0n	Specification of normal output (CMOS output) or N-ch open-drain output (n = 2, 3, 5)
0	Normal output (CMOS output)
1	N-ch open-drain output

(b) V850ES/JE3-L

After reset: 00H R/W Address: FFFFFC60H

	7	6	5	4	3	2	1	0
PF0	0	PF06	PF05	PF04	PF03	PF02	0	0

PF0n	Specification of normal output (CMOS output) or N-ch open-drain output (n = 2 to 6)
0	Normal output (CMOS output)
1	N-ch open-drain output

Caution When an output pin is pulled up to EV_{DD} or higher, be sure to set the PF0n bit to 1.

(3) Operation of interval timer based on input of external event count

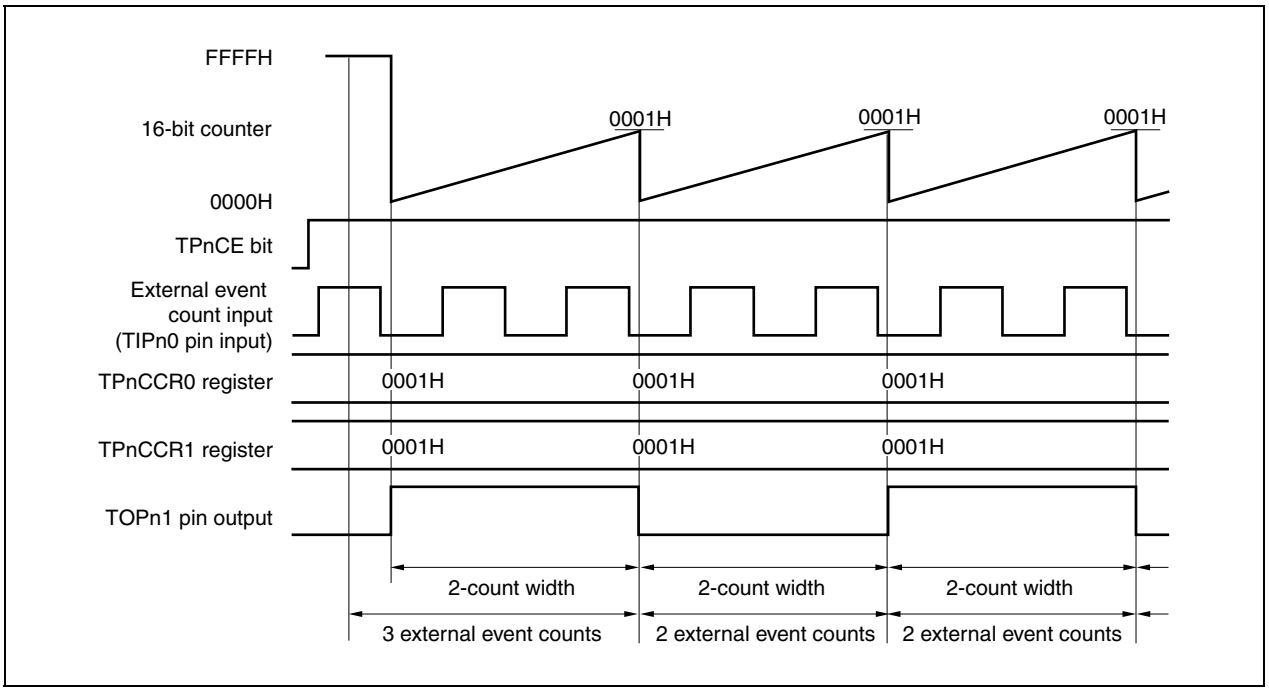
(a) Operation

When the 16-bit counter is incrementing based on the valid edge of the external count input (TIPn0 pin) in the interval timer mode, one external event count valid edge must be input immediately after the TPnCE bit changes from 0 to 1 to start the counter incrementing after the 16-bit counter is cleared from FFFFH to 0000H.

Once the TPnCCR0 and TPnCCR1 registers are set to 0001H (that is, the same value as was previously set), the TOPn1 pin output is inverted every two counts of the 16-bit counter.

Note that the TPnCTL1.TPnEEE bit can only be set to 1 when timer output (TOPn1) is used based on the input of an external event count.

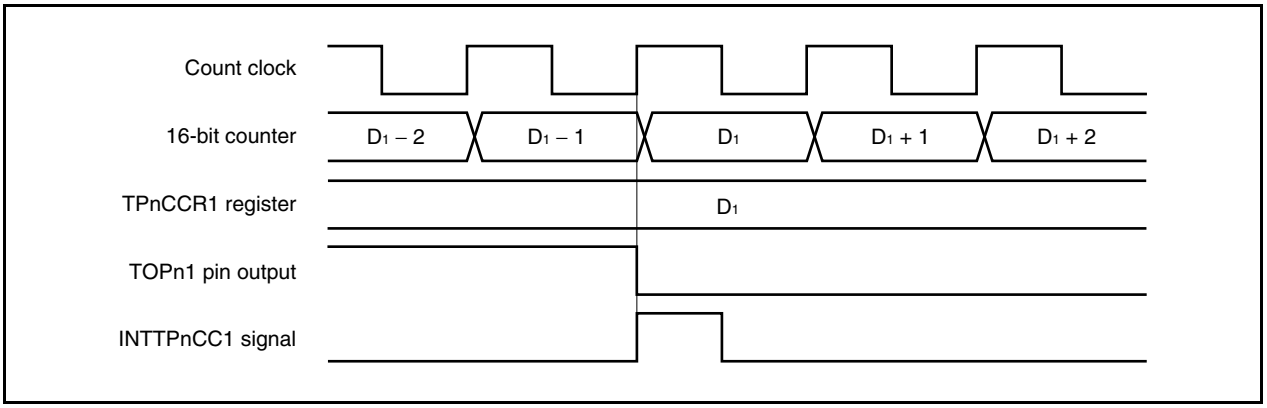
Figure 6-16. Operation of Interval Timer Based on Input of External Event Count (TIPn0)



(e) Timing of generating the compare match interrupt request signal (INTTPnCC1)

In the external trigger pulse output mode, the INTTPnCC1 signal is generated when the value of the 16-bit counter matches the value of the TPnCCR1 register.

Figure 6-37. Timing of Generating Compare Match Interrupt Signal (INTTPnCC1)



(1) 16-bit counter

This is a 16-bit counter that counts internal clocks and external events.

This counter can be read by using the TQ0CNT register.

When the TQ0CTL0.TQ0CE bit is 0 and the counter is stopped, the counter value is FFFFH. If the TQ0CNT register is read at this time, 0000H is read.

Reset sets the TQ0CE bit to 0, stopping the counter, and setting its value to FFFFH.

(2) CCR0 buffer register

This is a 16-bit compare register that compares the value of the 16-bit counter.

When the TQ0CCR0 register is used as a compare register, the value written to the TQ0CCR0 register is transferred to the CCR0 buffer register. If the value of the 16-bit counter matches the value of the CCR0 buffer register, a compare match interrupt request signal (INTTQ0CC0) is generated.

The CCR0 buffer register cannot be read or written directly.

The CCR0 buffer register is cleared to 0000H after reset because the TQ0CCR0 register is cleared to 0000H.

(3) CCR1 buffer register

This is a 16-bit compare register that compares the value of the 16-bit counter.

When the TQ0CCR1 register is used as a compare register, the value written to the TQ0CCR1 register is transferred to the CCR1 buffer register. If the count value of the 16-bit counter matches the value of the CCR1 buffer register, a compare match interrupt request signal (INTTQ0CC1) is generated.

The CCR1 buffer register cannot be read or written directly.

The CCR1 buffer register is cleared to 0000H after reset because the TQ0CCR1 register is cleared to 0000H.

(4) CCR2 buffer register

This is a 16-bit compare register that compares the value of the 16-bit counter.

When the TQ0CCR2 register is used as a compare register, the value written to the TQ0CCR2 register is transferred to the CCR2 buffer register. If the count value of the 16-bit counter matches the value of the CCR2 buffer register, a compare match interrupt request signal (INTTQ0CC2) is generated.

The CCR2 buffer register cannot be read or written directly.

The CCR2 buffer register is cleared to 0000H after reset because the TQ0CCR2 register is cleared to 0000H.

(5) CCR3 buffer register

This is a 16-bit compare register that compares the value of the 16-bit counter.

When the TQ0CCR3 register is used as a compare register, the value written to the TQ0CCR3 register is transferred to the CCR3 buffer register. If the count value of the 16-bit counter matches the value of the CCR3 buffer register, a compare match interrupt request signal (INTTQ0CC3) is generated.

The CCR3 buffer register cannot be read or written directly.

The CCR3 buffer register is cleared to 0000H after reset because the TQ0CCR3 register is cleared to 0000H.

(6) Edge detector

This circuit detects the valid edges input to the TIQ00 to TIQ03 pins. No edge, rising edge, falling edge, or both the rising and falling edges can be selected as the valid edge by using the TQ0IOC1 and TQ0IOC2 registers.

(5) Sub-count register (RC1SUBC)

The RC1SUBC register is a 16-bit register that counts the reference time of 1 second of the real-time counter.

It takes a value of 0000H to 7FFFH and counts one second with a clock of 32.768 kHz.

This register is read-only, in 16-bit units.

Reset sets this register to 0000H.

Cautions 1. When a correction is made by using the RC1SUBU register, the value may become 8000H or more.

2. This register is also cleared by writing to the second count register.

3. The value read from this register is not guaranteed if it is read during operation, because a changing value is read.

After reset: 0000H R Address: FFFFFAD0H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RC1SUBC																

(6) Second count register (RC1SEC)

The RC1SEC register is an 8-bit register that takes a value of 0 to 59 (decimal) and indicates the count value of seconds.

It counts up when the sub-counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (2 × 32.768 kHz) later. Set a decimal value of 00 to 59 to this register in BCD code. If a value outside this range is set, the register value returns to the normal value after one period.

This register can be read or written in 8-bit units.

Reset sets this register to 00H.

Caution Setting the RC1SEC register to values other than 00 to 59 is prohibited.

Remark See **10.4.1 Initial settings**, **10.4.2 Rewriting each counter during real-time counter operation**, and **10.4.3 Reading each counter during real-time counter operation** when reading or writing the RC1SEC register.

After reset: 00H R/W Address: FFFFFAD2H

	7	6	5	4	3	2	1	0
RC1SEC	0							

14.4 Operation

14.4.1 Operation in normal mode

D/A conversion is performed using a write operation to the DA0CS0 register as the trigger.
The setting method is described below.

<1> Set the PM10 bit to 1 (input mode).
<2> Clear the DA0M.DA0MD0 bit to 0 (normal mode).
<3> Set the analog voltage value to be output to the ANO0 pin to the DA0CS0 register.
Steps <2> and <3> above constitute the initial settings.
<4> Set the DA0M.DA0CE0 bit to 1 (D/A conversion enable).
The D/A converted analog voltage value is output from the ANO0 pin when this setting is performed.
<5> To change the analog voltage value, write to the DA0CS0 register.
The analog voltage value set immediately before is held until the next write operation is performed.

Remark For the alternate-function pin settings, refer to **Table 4-17 Settings When Pins Are Used for Alternate Functions**.

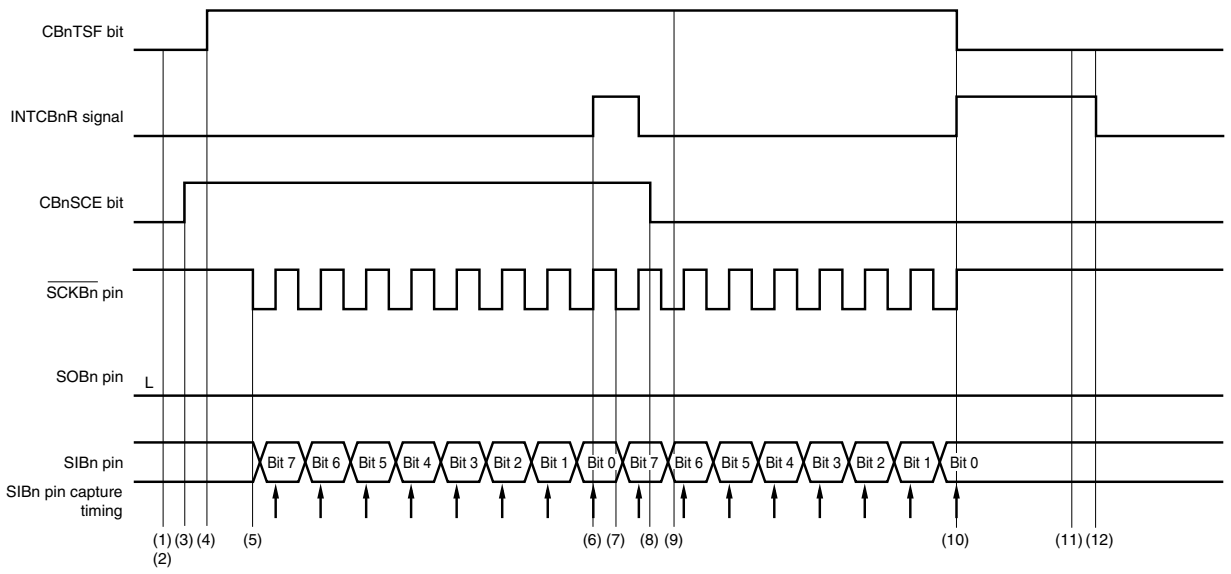
14.4.2 Operation in real-time output mode

D/A conversion is performed using the interrupt request signal of TMP2 (INTTP2CC0) as the trigger.
The setting method is described below.

<1> Set the PM10 bit to 1 (input mode).
<2> Set the DA0M.DA0MD0 bit to 1 (real-time output mode).
<3> Set the analog voltage value to be output to the ANO0 pin to the DA0CS0 register.
<4> Set the DA0M.DA0CE0 bit to 1 (D/A conversion enable).
Steps <2> to <4> above constitute the initial settings.
<5> Operate TMP2.
<6> The D/A converted analog voltage value is output from the ANOn pin when the INTTP2CC0 signal is generated.
Set the analog voltage value to be output to the DA0CS0 register next, before the next INTTP2CC0 signal is generated.
<7> After that, the value set to the DA0CS0 register is output from the ANO0 pin every time the INTTP2CC0 signal is generated.

Remarks 1. The output value of the ANO0 pin up to <6> above is undefined.
2. For the output value of the ANO0 pin in the IDLE1, IDLE2, HALT, and STOP modes, refer to **CHAPTER 21 STANDBY FUNCTION**.

Figure 16-20. Continuous Transfer Mode Operation Timing (Master Mode, Reception Mode)



(1) Write 00H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = f_{XX}/2, and master mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write A3H to the CBnCTL0 register, and select the reception mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by performing a dummy read of the CBnRX register, and reception is started.

(5) When reception is started, output the serial clock to the $\overline{\text{SCKBn}}$ pin, and capture the receive data of the SIBn pin in synchronization with the serial clock.

(6) When reception is completed, the reception complete interrupt request signal (INTCBnR) is generated, and reading receive data from the CBnRX register is enabled.

(7) Because the CBnCTL0.CBnSCE bit was 1 when communication ended, the next communication is started immediately.

(8) To end continuous reception with the current reception, clear the CBnSCE bit to 0.

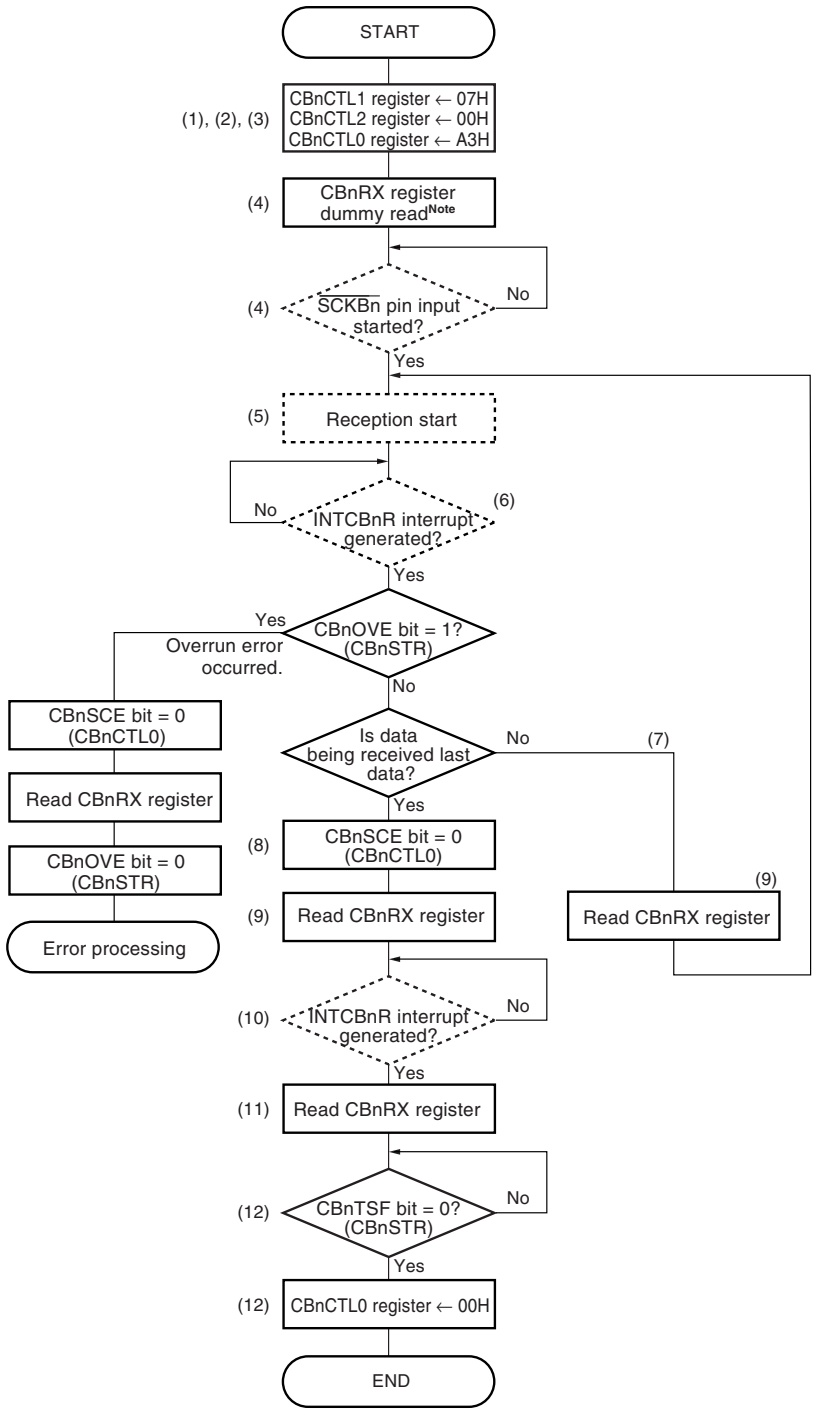
(9) Read the CBnRX register.

(10) When reception is completed, the INTCBnR signal is generated and reading receive data from the CBnRX register is enabled. If the CBnSCE bit is set to 0 before communication is complete, stop the serial clock output to the $\overline{\text{SCKBn}}$ pin and clear the CBnTSF bit to 0 to end the receive operation.

(11) Read the CBnRX register.

(12) To disable reception, clear the CBnCTL0.CBnPWR and CBnCTL0.CBnRXE bits to 0 after confirming that the CBnTSF bit is 0.

Figure 16-25. Continuous Transfer Mode Operation (Slave Mode, Reception Mode)

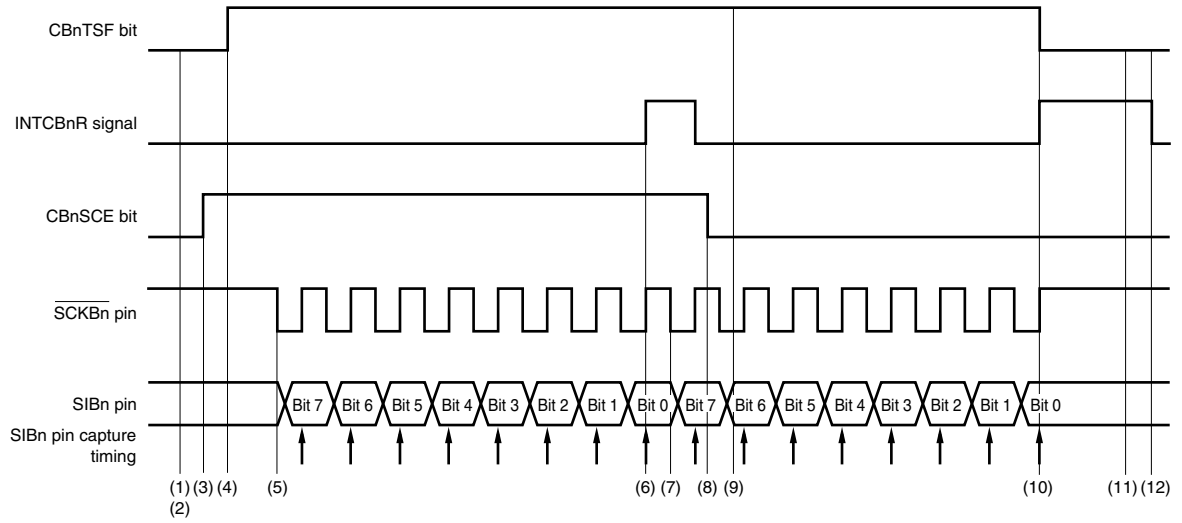


Note If the serial clock is input via the $\overline{\text{SCKBn}}$ pin of the master before a dummy-read of the CBnRX register is executed, data cannot be received normally. In this case, initialize both the master and the slave.

Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in **Figure 16-26**.

Figure 16-26. Continuous Transfer Mode Operation Timing (Slave Mode, Reception Mode)



(1) Write 07H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$), and slave mode.
(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.
(3) Write A3H to the CBnCTL0 register, and select the reception mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).
(4) The CBnSTR.CBnTSF bit is set to 1 by performing a dummy read of the CBnRX register, and the device waits for serial clock input.
(5) When the serial clock is input, capture the receive data of the SIBn pin in synchronization with the serial clock.
(6) When reception is completed, the reception complete interrupt request signal (INTCBnR) is generated, and reading receive data from the CBnRX register is enabled.
(7) When the serial clock is input with the CBnCTL0.CBnSCE bit set to 1, continuous reception is started.
(8) To end continuous reception with the current reception, clear the CBnSCE bit to 0.
(9) Read the CBnRX register.
(10) When reception is completed, the INTCBnR signal is generated, and reading receive data from the CBnRX register is enabled. If the CBnSCE bit is set to 0 before communication is complete, clear the CBnTSF bit to 0 to end the receive operation.
(11) Read the CBnRX register.
(12) To disable reception, clear the CBnCTL0.CBnPWR and CBnCTL0.CBnRXE bits to 0 after confirming that the CBnTSF bit is 0.

17.14 Communication Reservation

17.14.1 When communication reservation function is enabled (IICFn.IICRSVn bit = 0)

To start master device communications when the V850ES/JC3-L and V850ES/JE3-L are not currently using the bus, a communication reservation can be made to enable transmission of a start condition when the bus is released. There are two modes in which the bus is not used by the V850ES/JC3-L and V850ES/JE3-L.

- When arbitration results in the V850ES/JC3-L and V850ES/JE3-L being neither the master nor a slave
- When an extension code is received and slave operation is disabled ($\overline{ACK}$ is not returned and the bus was released when the IICCn.LRELn bit was set to 1).

If the IICCn.STTn bit is set to 1 while the bus is not used by the V850ES/JC3-L and V850ES/JE3-L, a start condition is automatically generated and a wait status is set after the bus is released (after a stop condition is detected).

When the bus release is detected (when a stop condition is detected), writing to the IICn register causes master address transfer to start. At this point, the IICCn.SPIEn bit should be set to 1.

When STTn has been set to 1, the operation mode (as start condition or as communication reservation) is determined according to the bus status.

If the bus has been released .....A start condition is generated
If the bus has not been released (standby mode).....Communication reservation

To detect which operation mode has been determined for the STTn bit, set the STTn bit to 1, wait for the wait period, then check the IICSn.MSTSn bit.

The wait periods, which should be set via software, are listed in Table 17-6. These wait periods can be set by using the SMCn, CLn1, and CLn0 bits of the IICCLn register and the IICXn.CLXn bit.

17.16 Communication Operations

Next the following three operations are shown using flowcharts.

(1) Master operation in single master system

The flowchart when using the V850ES/JC3-L and V850ES/JE3-L as the master in a single master system is shown below.

This flowchart is broadly divided into the initial settings and communication processing. Execute the initial settings at startup. If communication with the slave is required, prepare the communication and then execute communication processing.

(2) Master operation in multimaster system

In the I²C0n bus multimaster system, whether the bus is released or used cannot be judged by the I²C bus specifications when the bus takes part in a communication. Here, when data and clock are at a high level for a certain period (1 frame), the V850ES/JC3-L and V850ES/JE3-L take part in a communication with bus released state.

This flowchart is broadly divided into the initial settings, communication waiting, and communication processing. The processing when the V850ES/JC3-L and V850ES/JE3-L lose in arbitration and are specified as the slave is omitted here, and only the processing as the master is shown. Execute the initial settings at startup to take part in a communication. Then, wait for the communication request as the master or wait for the specification as the slave. The actual communication is performed in the communication processing, and it supports the transmission/reception with the slave and the arbitration with other masters.

(3) Slave operation

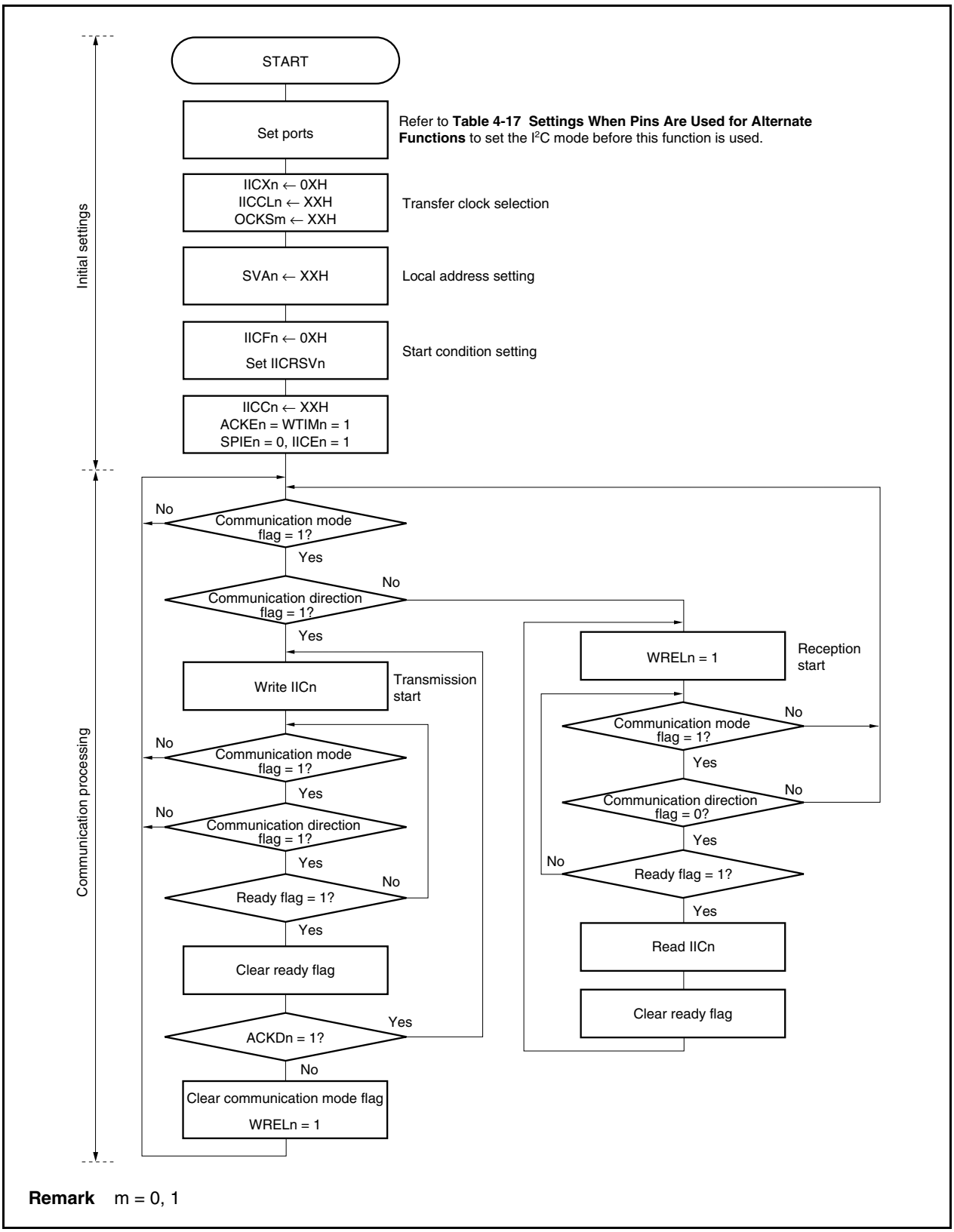
An example of when the V850ES/JC3-L and V850ES/JE3-L are used as the slave of the I²C0n bus is shown below.

When used as the slave, operation is started by an interrupt. Execute the initial settings at startup, then wait for the INTIICn interrupt occurrence (communication waiting). When the INTIICn interrupt occurs, the communication status is judged and its result is passed as a flag over to the main processing.

By checking the flags, necessary communication processing is performed.

For reception, the required number of data items are received and $\overline{ACK}$ is not returned for the next data immediately after transfer is complete. After that, the master device generates the stop condition or restart condition. This causes exit from communications.

Figure 17-21. Slave Operation Flowchart (1)



Remark m = 0, 1

Table 19-1. Interrupt Source List (1/3)

Type	Default Priority	Name	Trigger	Generating Unit	Exception Code	Handler Address	Interrupt Control Register
Reset	–	RESET	$\overline{\text{RESET}}$ pin input Reset input by internal source	RESET	0000H	00000000H	–
Non-maskable	–	NMI	NMI pin valid edge input	Pin	0010H	00000010H	–
	–	INTWDT2	WDT2 overflow	WDT2	0020H	00000020H	–
Software exception	–	TRAP0n[Note 2]	TRAP instruction	–	004nH[Note 2]	00000040H	–
	–	TRAP1n[Note 2]	TRAP instruction	–	005nH[Note 2]	00000050H	–
Exception trap	–	ILGOP/ DBG0	Illegal opcode/ DBTRAP instruction	–	0060H	00000060H	–
Maskable	0	INTLVI	Low voltage detection	POCLVI	0080H	00000080H	LVIIC
	1	INTP0	External interrupt pin input edge detection (INTP0)	Pin	0090H	00000090H	PIC0
	2	INTP1[Note 3]	External interrupt pin input edge detection (INTP1)	Pin	00A0H	000000A0H	PIC1
	3	INTP2	External interrupt pin input edge detection (INTP2)	Pin	00B0H	000000B0H	PIC2
	4	INTP3[Note 3]	External interrupt pin input edge detection (INTP3)	Pin	00C0H	000000C0H	PIC3
	5	INTP4[Note 3]	External interrupt pin input edge detection (INTP4)	Pin	00D0H	000000D0H	PIC4
	6	INTP5	External interrupt pin input edge detection (INTP5)	Pin	00E0H	000000E0H	PIC5
	7	INTP6	External interrupt pin input edge detection (INTP6)	Pin	00F0H	000000F0H	PIC6
	8	INTP7	External interrupt pin input edge detection (INTP7)	Pin	0100H	00000100H	PIC7
	9	INTTQ0OV	TMQ0 overflow	TMQ0	0110H	00000110H	TQ0OVIC
	10	INTTQ0CC0	TMQ0 capture 0/compare 0 match	TMQ0	0120H	00000120H	TQ0CCIC0
	11	INTTQ0CC1	TMQ0 capture 1/compare 1 match	TMQ0	0130H	00000130H	TQ0CCIC1
	12	INTTQ0CC2	TMQ0 capture 2/compare 2 match	TMQ0	0140H	00000140H	TQ0CCIC2
	13	INTTQ0CC3	TMQ0 capture 3/compare 3 match	TMQ0	0150H	00000150H	TQ0CCIC3
	14	INTTP0OV	TMP0 overflow	TMP0	0160H	00000160H	TP0OVIC
	15	INTTP0CC0	TMP0 capture 0/compare 0 match	TMP0	0170H	00000170H	TP0CCIC0
	16	INTTP0CC1	TMP0 capture 1/compare 1 match	TMP0	0180H	00000180H	TP0CCIC1
	17	INTTP1OV	TMP1 overflow	TMP1	0190H	00000190H	TP1OVIC
	18	INTTP1CC0	TMP1 capture 0/compare 0 match	TMP1	01A0H	000001A0H	TP1CCIC0
	19	INTTP1CC1	TMP1 capture 1/compare 1 match	TMP1	01B0H	000001B0H	TP1CCIC1
	20	INTTP2OV	TMP2 overflow	TMP2	01C0H	000001C0H	TP2OVIC
	21	INTTP2CC0	TMP2 capture 0/compare 0 match	TMP2	01D0H	000001D0H	TP2CCIC0
	22	INTTP2CC1	TMP2 capture 1/compare 1 match	TMP2	01E0H	000001E0H	TP2CCIC1

Notes 1. The software that generated the exception event can be checked using the exception code set to the EICC bit of the ECR register.

2. n = 0 to FH

3. V850ES/JE3-L only

21.4 IDLE1 Mode

21.4.1 Setting and operation status

The IDLE1 mode is set by clearing the PSMR.PSM1 and PSMR.PSM0 bits to 00 and setting the PSC.STP bit to 1 in the normal operation mode.

In the IDLE1 mode, the clock oscillator, PLL, and flash memory continue operating but clock supply to the CPU and other on-chip peripheral functions stops.

As a result, program execution stops and the contents of the internal RAM before the IDLE1 mode was set are retained. The CPU and other on-chip peripheral functions stop operating. However, the on-chip peripheral functions that can operate with the subclock or an external clock continue operating.

Table 21-5 shows the operating status in the IDLE1 mode.

The IDLE1 mode can reduce the power consumption more than the HALT mode because it stops the operation of the on-chip peripheral functions. The main clock oscillator does not stop, so the normal operation mode can be restored without waiting for the oscillation stabilization time after the IDLE1 mode has been released, in the same manner as when the HALT mode is released.

Cautions 1. Insert five or more NOP instructions after the instruction that stores data in the PSC register to set the IDLE1 mode.

2. If the IDLE1 mode is set while an unmasked interrupt request signal is being held pending, the CPU does not shift to the IDLE1 mode but executes the next instruction.

CHAPTER 23 CLOCK MONITOR

23.1 Functions

The clock monitor monitors the main clock by using the internal oscillator clock and generates a reset request signal when oscillation of the main clock is stopped.

Once the operation of the clock monitor has been enabled by an operation enable flag, it cannot be cleared to 0 by any means other than a reset.

When a reset by the clock monitor occurs, the RESF.CLMRF bit is set. For details on the RESF register, see **22.3 Registers to Check Reset Source**.

The clock monitor automatically stops under the following conditions.

- During oscillation stabilization time after STOP mode is released
- When the main clock is stopped (from when the PCC.MCK bit = 1 during subclock operation, until the PCC.CLS bit = 0 during main clock operation)
- When the monitoring clock (internal oscillator clock) is stopped
- When the CPU operates with the internal oscillator clock

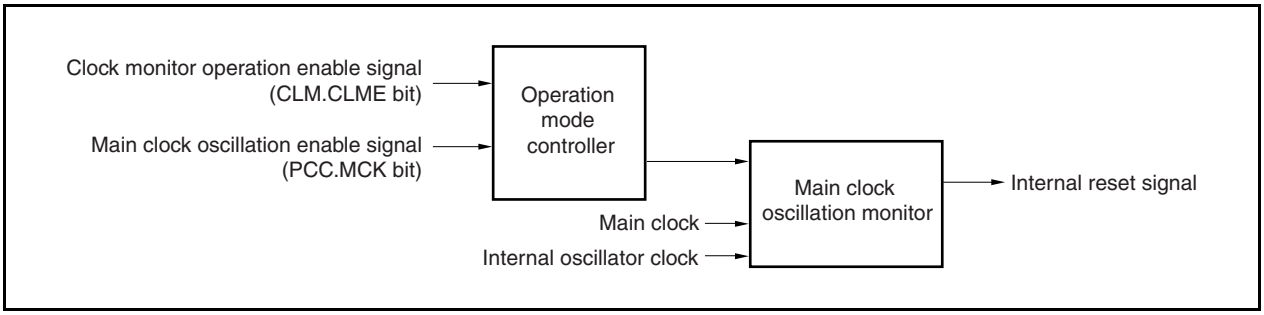
23.2 Configuration

The clock monitor includes the following hardware.

Table 23-1. Configuration of Clock Monitor

Item	Configuration
Control register	Clock monitor mode register (CLM)

Figure 23-1. Block Diagram of Clock Monitor



29.2.3 Allocation of user resources

The user must prepare the following resources to perform communication between MINICUBE2 and the target device and implement each debug function. These items need to be set in the user program or using the compiler options.

(1) Allocation of memory space

The shaded portions in Figure 29-5 are the areas reserved for placing the debug monitor program, so user programs and data cannot be allocated to these spaces. These spaces must be secured so as not to be used by the user program.

(2) Security ID setting

The ID code must be embedded in the area between 0000070H and 0000079H in Figure 29-5, to prevent the memory from being read by an unauthorized person. For details, see **29.3 ROM Security Function**.

32.9 Flash Memory Programming Characteristics

(1) Basic characteristics

(T_A = –40 to +85°C, V_{DD} = EV_{DD} = AV_{REF0} = AV_{REF1} = 2.7 to 3.6 V, V_{SS} = EV_{SS} = AV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Operating frequency	f_{CPU}			2.5		20	MHz
Supply voltage	V_{DD}	2.5 MHz ≤ f_{XX} ≤ 20 MHz		2.7		3.6	V
Number of rewrites	C_{WRT}	Used for updating programs When using flash memory programmer and Renesas Electronics self programming library	Retained for 15 years	1,000			times
		Used for updating data When using Renesas Electronics EEPROM emulation library (usable ROM size: 12 KB of 6 consecutive blocks, or 6 KB of 3 consecutive blocks)	Retained for 5 years	10,000			times
Programming temperature	t_{PRG}			–40		+85	°C

(2) Serial write operation characteristics

(T_A = –40 to +85°C, V_{DD} = EV_{DD} = AV_{REF0} = AV_{REF1} = 2.7 to 3.6 V, V_{SS} = EV_{SS} = AV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
FLMD0, FLMD1 setup time	t_{MDSET}		2		3000	ms
FLMD0 count start time from $\overline{\text{RESET}}$↑	t_{RFCF}	f_X = 2.5 to 10 MHz	800			μs
FLMD0 counter high-level width/ low-level width	t_{CH}/t_{CL}		10		100	μs
FLMD0 counter rise time/fall time	t_R/t_F				1	μs

Flash write mode setup timing