Renesas Electronics America Inc - DF70844AN80FPV Datasheet

Welcome to E-XFL.COM

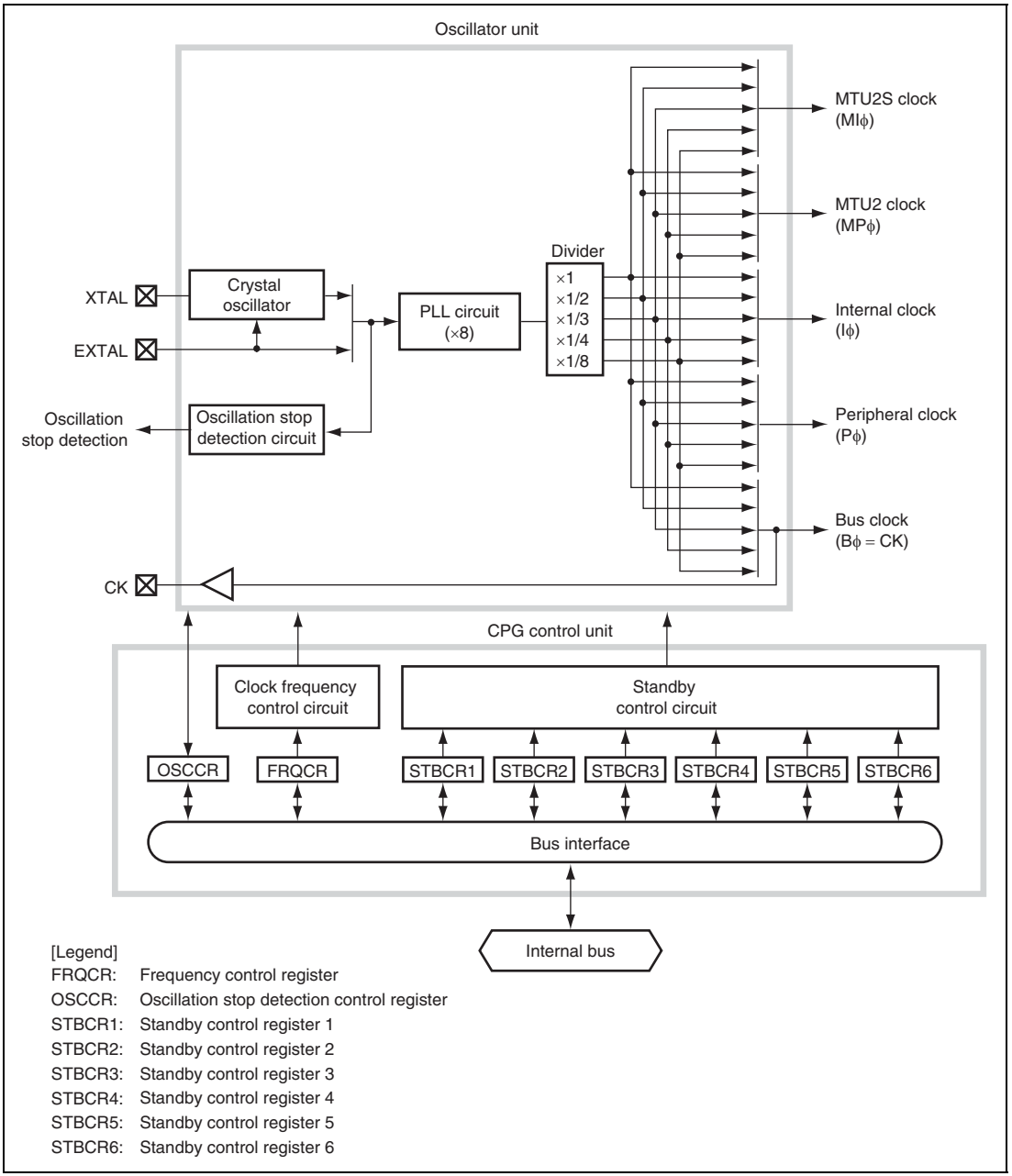
What is "Embedded - Microcontrollers"?

"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	Not For New Designs
Core Processor	SH-2
Core Size	32-Bit Single-Core
Speed	80MHz
Connectivity	EBI/EMI, FIFO, I²C, SCI, SSU
Peripherals	DMA, POR, PWM, WDT
Number of I/O	76
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LQFP
Supplier Device Package	112-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df70844an80fpv

Figure 4.1 shows a block diagram of the clock pulse generator.



[Legend]
FRQCR: Frequency control register
OSCCR: Oscillation stop detection control register
STBCR1: Standby control register 1
STBCR2: Standby control register 2
STBCR3: Standby control register 3
STBCR4: Standby control register 4
STBCR5: Standby control register 5
STBCR6: Standby control register 6

Figure 4.1 Block Diagram of Clock Pulse Generator

8.2.7 DTC Enable Registers A to E (DTCERA to DTCERE)

DTCER which is comprised of eight registers, DTCERA to DTCERE, is a register that specifies DTC activation interrupt sources. The correspondence between interrupt sources and DTCE bits is shown in table 8.2.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTCE15	DTCE14	DTCE13	DTCE12	DTCE11	DTCE10	DTCE9	DTCE8	DTCE7	DTCE6	DTCE5	DTCE4	DTCE3	DTCE2	DTCE1	DTCE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	DTCE15	0	R/W	DTC Activation Enable 15 to 0 If set to 1, the corresponding interrupt source is specified as a DTC activation source. [Clearing conditions] • Writing 0 to the bit after reading 1 from it • When the DISEL bit is 1 and the data transfer has ended • When the specified number of transfers have ended These bits are not cleared when the DISEL bit is 0 and the specified number of transfers have not ended [Setting condition] • Writing 1 to the bit after reading 0 from it
14	DTCE14	0	R/W	
13	DTCE13	0	R/W	
12	DTCE12	0	R/W	
11	DTCE11	0	R/W	
10	DTCE10	0	R/W	
9	DTCE9	0	R/W	
8	DTCE8	0	R/W	
7	DTCE7	0	R/W	
6	DTCE6	0	R/W	
5	DTCE5	0	R/W	
4	DTCE4	0	R/W	
3	DTCE3	0	R/W	
2	DTCE2	0	R/W	
1	DTCE1	0	R/W	
0	DTCE0	0	R/W	

Address	Area	Memory Type	Capacity	Bus Width
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*2
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*2
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8 or 16 bits*2
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection MPX-I/O	64 Mbytes	8 or 16 bits*2
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*2
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*2
H'20000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed.

1. The bus width is selected by the mode pins.
2. The bus width is selected by the register setting.

Bit	Bit Name	Initial Value	R/W	Description
0	HIZCNT	0	R/W	Hi-Z Control Specifies the state in software standby mode and when bus mastership is released for CKE, $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, and $\overline{\text{CASL}}$. 0: High impedance in software standby mode and when bus mastership is released for CKE, $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, and $\overline{\text{CASL}}$. 1: Driven in software standby mode and when bus mastership is released for CKE, $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, and $\overline{\text{CASL}}$.

9.4.2 CSn Space Bus Control Register (CSnBCR) (n = 0 to 8)

CSnBCR is a 32-bit readable/writable register that specifies the type of memory connected to the respective space, the data bus width of the space, and the number of wait cycles between access cycles.

Do not access external memory other than area 0 until the register initialization is complete.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	IWW[1:0]		-	IWRWD[1:0]		-	IWRWS[1:0]		-	IWRRD[1:0]		-	IWRRS[1:0]	
Initial value:	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1
R/W:	R	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TYPE[2:0]			-	BSZ[1:0]		-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1*	1*	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R	R

Note: * When the on-chip ROM is disabled, CS0BCR samples the value input through the MD0 and MD1 external pins that specify the bus width when a power-on reset is performed.

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Address Multiplexing: An address multiplexing is specified so that SDRAM can be connected without external address multiplexing circuitry according to the setting of bits BSZ[1:0] in CSnBCR, and AxROW[1:0] and AxCOL[1:0] in SDCR. Tables 9.20 to 9.25 show the relationship between the settings of bits BSZ[1:0], AxROW[1:0], and AxCOL[1:0] and the bits output at the address pins. Do not specify those bits in the manner other than this table, otherwise the operation of this LSI is not guaranteed. A25 to A18 are not multiplexed and the original values of address are always output at these pins.

When the data bus width is 16 bits (BSZ[1:0] = B'10), the A0 pin of SDRAM specifies a word address. Therefore, connect this A0 pin of SDRAM to the A1 pin of this LSI, then A1 pin to the A2 pin, and so on. When the data bus width is 32 bits (BSZ[1:0] = B'11), the A0 pin of SDRAM specifies a longword address. Therefore, connect this A0 pin of SDRAM to the A2 pin of this LSI, then A1 pin to the A3 pin, and so on.

Table 11.42 Setting of Bits BTE[1:0]

Bit 1	Bit 0	
BTE1	**BTE0**	**Description**
0	0	Enables transfer from the buffer registers to the temporary registers*[1] and does not link the transfer with interrupt skipping operation.
0	1	Disables transfer from the buffer registers to the temporary registers.*[1]
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping operation.*[1]*[2]
1	1	Setting prohibited

Notes: 1. Transfers from the temporary registers to the compare registers take place in accordance with the setting of the MD[3:0] bit field in TMDR, regardless of the setting of the BTE[1:0] bit field. For details, refer to section 11.4.8, Complementary PWM Mode.

2. When interrupt skipping is disabled (the T3AEN and T4VEN bits are cleared to 0 in the timer interrupt skipping set register (TITCR) or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0)), be sure to disable link of buffer transfer with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTER) to 0). If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

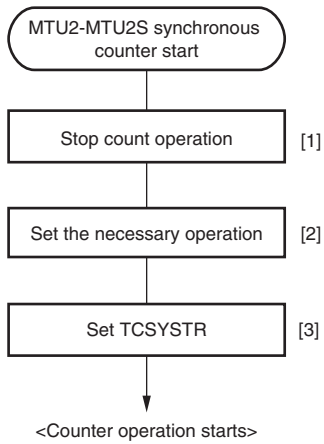
11.4.10 MTU2–MTU2S Synchronous Operation

(1) MTU2–MTU2S Synchronous Counter Start

The counters in the MTU2 and MTU2S which operate at different clock systems can be started synchronously by making the TCSYSTR settings in the MTU2.

(a) Example of MTU2–MTU2S Synchronous Counter Start Setting Procedure

Figure 11.83 shows an example of synchronous counter start setting procedure.



[1] Use TSTR registers in the MTU2 and MTU2S and halt the counters used for synchronous start operation.

[2] Specify necessary operation with appropriate registers such as TCR and TMDR.

[3] In TCSYSTR in the MTU2, set the bits corresponding to the counters to be started synchronously to 1. The TSTRs are automatically set appropriately and the counters start synchronously.

Notes: 1. Even if a bit in TCSYSTR corresponding to an operating counter is cleared to 0, the counter will not stop. To stop the counter, clear the corresponding bits in TSTR and TSTRS to 0 directly.

2. To start channels 3 and 4 in reset-synchronized PWM mode or complementary PWM mode, make appropriate settings in TCYSTR according to the TSTR setting for the respective mode. For details, refer to section 11.4.7, Reset-Synchronized PWM Mode, and section 11.4.8, Complementary PWM Mode.

Figure 11.83 Example of Synchronous Counter Start Setting Procedure

Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode: Figure 11.165 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

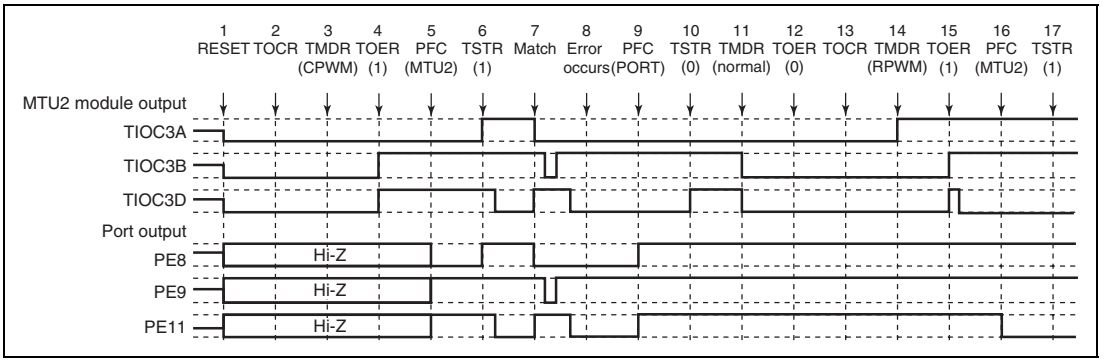


Figure 11.165 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode

1 to 10 are the same as in figure 11.161.

11. Set normal mode. (MTU2 output goes low.)
12. Disable channel 3 and 4 output with TOER.
13. Select the reset-synchronized PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set reset-synchronized PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set MTU2 output with the PFC.
17. Operation is restarted by TSTR.

Section 15 Serial Communication Interface (SCI)

This LSI has three independent serial communication interface (SCI) channels. The SCI can handle both asynchronous and clock synchronous serial communication. In asynchronous serial communication mode, serial data communication can be carried out with standard asynchronous communication chips such as a Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). A function is also provided for serial communication between processors (multiprocessor communication function).

15.1 Features

- Choice of asynchronous or clock synchronous serial communication mode
- Asynchronous mode:
  - Serial data communication is performed by start-stop in character units. The SCIF can communicate with a universal asynchronous receiver/transmitter (UART), an asynchronous communication interface adapter (ACIA), or any other communications chip that employs a standard asynchronous serial system. There are twelve selectable serial data communication formats.
  - Data length: 7 or 8 bits
  - Stop bit length: 1 or 2 bits
  - Parity: Even, odd, or none
  - Multiprocessor communications
  - Receive error detection: Parity, overrun, and framing errors
  - Break detection: Break is detected by reading the RXD pin level directly when a framing error occurs.
- Clock synchronous mode:
  - Serial data communication is synchronized with a clock signal. The SCIF can communicate with other chips having a clock synchronous communication function.
  - Data length: 8 bits
  - Receive error detection: Overrun errors
- Full duplex communication: The transmitting and receiving sections are independent, so the SCI can transmit and receive simultaneously. Both sections use double buffering, so high-speed continuous data transfer is possible in both the transmit and receive directions.
- On-chip baud rate generator with selectable bit rates
- Internal or external transmit/receive clock source: From either baud rate generator (internal clock) or SCK pin (external clock)

15.3.1 Receive Shift Register (SCRSR)

SCRSR receives serial data. The SCI converts serial data input to SCRSR via the RXD pin to parallel data. When one byte has been received, it is automatically transferred to SCRDR. The CPU cannot read or write to SCRSR directly.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

15.3.2 Receive Data Register (SCRDR)

SCRDR is a register that stores serial receive data. After receiving one byte of serial data, the SCI transfers the received data from the receive shift register (SCRSR) into SCRDR for storage and completes operation. After that, SCRSR is ready to receive data.

Since SCRSR and SCRDR work as a double buffer in this way, data can be received continuously.

SCRDR is a read-only register and SCRDR can be read but not written to by the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

15.3.3 Transmit Shift Register (SCTSR)

SCTSR transmits serial data. The SCI loads transmit data from the transmit data register (SCTDR) into SCTSR, then transmits the data serially from the TXD pin. After transmitting one data byte, the SCI automatically loads the next transmit data from SCTDR into SCTSR and starts transmitting again. If the TDRE flag in the serial status register (SCSSR) is set to 1, the SCI does not transfer data from SCTDR to SCTSR. The CPU cannot read or write to SCTSR directly.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	CKS[3:0]	0000	R/W	Transfer Clock Select 3 to 0 These bits should be set according to the necessary transfer rate (table 18.3) in master mode. In slave mode, these bits should be used to specify the data setup time in transmission mode. The setup time is set to 10 tpcyc when CKS3 = 0 or 20 tpcyc when CKS3 = 1 (tpcyc is one Pϕ cycle).

Table 18.3 Transfer Rate

Bit 3	Bit 2	Bit 1	Bit 0		Transfer Rate					
CKS3	CKS2	CKS1	CKS0	Clock	Pϕ=10 MHz	Pϕ=16 MHz	Pϕ=20 MHz	Pϕ=25 MHz	Pϕ=33 MHz	Pϕ=40 MHz
0	0	0	0	Pϕ/28	357 kHz	571 kHz	714 kHz	893 kHz	1.18 MHz	1.43 MHz
			1	Pϕ/40	250 kHz	400 kHz	500 kHz	625 kHz	825 kHz	1.00 MHz
		1	0	Pϕ/48	208 kHz	333 kHz	417 kHz	521 kHz	688 kHz	833 kHz
			1	Pϕ/64	156 kHz	250 kHz	313 kHz	391 kHz	516 kHz	625 kHz
	1	0	0	Pϕ/80	125 kHz	200 kHz	250 kHz	313 kHz	413 kHz	500 kHz
			1	Pϕ/100	100 kHz	160 kHz	200 kHz	250 kHz	330 kHz	400 kHz
		1	0	Pϕ/112	89.3 kHz	143 kHz	179 kHz	223 kHz	295 kHz	357 kHz
			1	Pϕ/128	78.1 kHz	125 kHz	156 kHz	195 kHz	258 kHz	313 kHz
1	0	0	0	Pϕ/112	89.3 kHz	143 kHz	179 kHz	223 kHz	295 kHz	357 kHz
			1	Pϕ/160	62.5 kHz	100 kHz	125 kHz	156 kHz	206 kHz	250 kHz
		1	0	Pϕ/192	52.1 kHz	83.3 kHz	104 kHz	130 kHz	172 kHz	208 kHz
			1	Pϕ/256	39.1 kHz	62.5 kHz	78.1 kHz	97.7 kHz	129 kHz	156 kHz
	1	0	0	Pϕ/320	31.3 kHz	50.0 kHz	62.5 kHz	78.1 kHz	103 kHz	125 kHz
			1	Pϕ/400	25.0 kHz	40.0 kHz	50.0 kHz	62.5 kHz	82.5 kHz	100 kHz
		1	0	Pϕ/448	22.3 kHz	35.7 kHz	44.6 kHz	55.8 kHz	73.7 kHz	89.3 kHz
			1	Pϕ/512	19.5 kHz	31.3 kHz	39.1 kHz	48.8 kHz	64.5 kHz	78.1 kHz

18.4.8 Using the IICRST Bit to Reset I²C Bus Interface 2

Some registers and the control part for I²C of the I²C bus interface 2 can be reset by writing 1 to the IICRST bit in ICCR2. Figure 18.19 shows an example of the sequence for resetting the I²C bus interface 2 by using the IICRST bit.

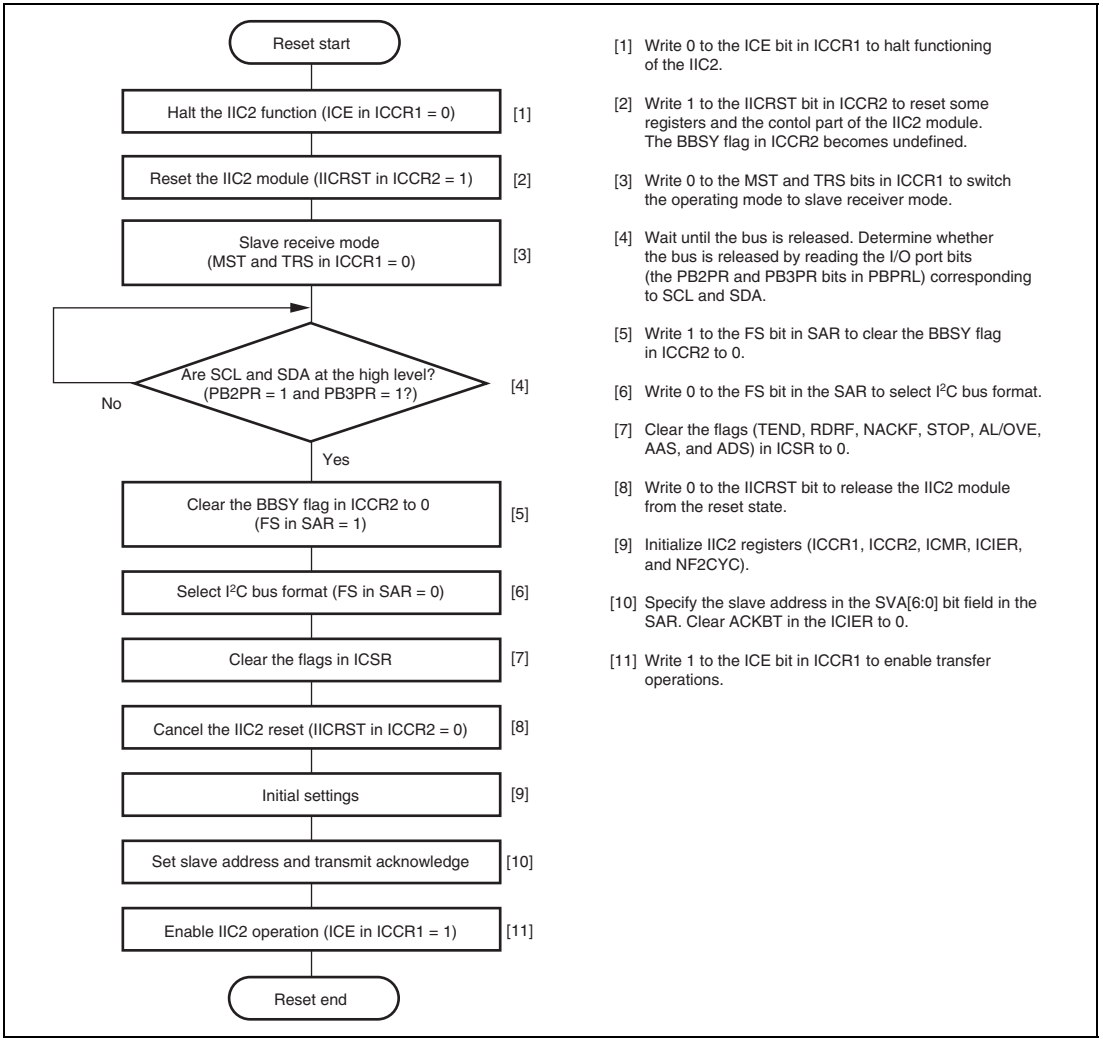


Figure 18.19 Sequence for Using the IICRST Bit to Reset I²C Bus Interface 2

18.5 Interrupt Sources and DTC

The IIC2 module has six interrupt sources; these are for the transmit data empty interrupt request (IITXI), transmit end interrupt request (IITEI), receive data full interrupt request (IIRXI), stop condition detection interrupt request (IISTPI), and NACK detection, arbitration lost or overrun error interrupt request (IINAKI).

The interrupt sources and their order of priority are listed in table 18.4. The TIE, RIE, TEIE, NAKIE, and STIE bits in the I²C bus interrupt enable register (ICIER) enable or disable the various interrupt sources. Furthermore, each of the corresponding interrupt requests is independently conveyed to the interrupt controller.

A IITXI interrupt request is generated when the TDRE flag in the I²C bus status register (ICSR) is set to 1. A IITXI interrupt request can activate the data transfer controller (DTC) to handle data transfer. When the DTC is activated to handle transfer, if the DISEL bit of the DTC is 0 and the value of the transfer counter is non-zero, the TDRE flag is automatically cleared to 0 once the DTC has written to the ICDRT, so a IITXI interrupt request is not sent to the CPU. If the DISEL bit of the DTC is 0 and the value of the transfer counter is 0, or if the DISEL bit is 1, writing to ICDRT does not lead to automatic clearing of the TDRE flag, so a IITXI interrupt request is subsequently generated for the CPU.

A IIRXI interrupt request is generated when the RDRF flag in ICSR is set to 1. A IIRXI interrupt request can activate the DTC to handle data transfer. When the DTC is activated to handle transfer, if the DISEL bit of the DTC is 0 and the value of the transfer counter is non-zero, the RDRF flag is automatically cleared to 0 once the DTC has read from ICDRR, so a IIRXI interrupt request is not sent to the CPU. If the DISEL bit of the DTC is 0 and the value of the transfer counter is 0, or if the DISEL bit is 1, Reading from ICDRR does not lead to automatic clearing of the RDRF flag, so a IIRXI interrupt request is subsequently generated for the CPU.

A IINAKI interrupt request is generated when the NACKF or AL/OVE flag in ICSR is set to 1. A IINAKI interrupt request is not capable of activating the DTC. Setting of the NACKF to 1 only leads to a IINAKI interrupt request when communications are in I²C format.

A IISTPI interrupt request is generated when the STOP flag in ICSR is set to 1. A IISTPI interrupt request is not capable of activating the DTC. Setting of the STOP flag to 1 only leads to a IISTPI interrupt request when communications are in I²C format.

A IITEI interrupt request is generated when the TEND flag in ICSR is set to 1. A IITEI interrupt request is not capable of activating the DTC.

Bit	Bit Name	Initial Value	R/W	Description
6	PE5MD2	0	R/W	PE5 Mode
5	PE5MD1	0	R/W	Select the function of the
4	PE5MD0	0	R/W	PE5/$\overline{CS6}$/TIOC1B/TXD3/$\overline{ASEBRKAK}$/$\overline{ASEBRK}$ pin. Fixed to $\overline{ASEBRKAK}$ output/$\overline{ASEBRK}$ input when using E10A (in $\overline{ASEMD0}$ = low). 000: PE5 I/O (port) 001: TIOC1B I/O (MTU2) 010: TXD3 output (SCIF) 101: $\overline{CS6}$ output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PE4MD2	0	R/W	PE4 Mode
1	PE4MD1	0	R/W	Select the function of the PE4/TIOC1A/RXD3/TCK
0	PE4MD0	0	R/W	pin. Fixed to TCK input when using E10A (in $\overline{ASEMD0}$ = low). 000: PE4 I/O (port) 001: TIOC1A I/O (MTU2) 010: RXD3 input (SCIF) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L1 (PECRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE3 MD2	PE3 MD1	PE3 MD0	-	PE2 MD2	PE2 MD1	PE2 MD0	-	PE1 MD2	PE1 MD1	PE1 MD0	-	-	PE0 MD1	PE0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PE5MD2	0	R/W	PE5 Mode
5	PE5MD1	0	R/W	Select the function of the
4	PE5MD0	0	R/W	PE5/$\overline{\text{CS6}}$/$\overline{\text{CE1B}}$/TIOC1B/TXD3/AUDATA1 pin. Fixed to AUDATA1 output when using the AUD function of the E10A. 000: PE5 I/O (port) 001: TIOC1B I/O (MTU2) 010: TXD3 output (SCIF) 101: $\overline{\text{CS6}}$/$\overline{\text{CE1B}}$ output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PE4MD2	0	R/W	PE4 Mode
1	PE4MD1	0	R/W	Select the function of the
0	PE4MD0	0	R/W	PE4/$\overline{\text{IOIS16}}$/TIOC1A/RXD3/AUDATA2 pin. Fixed to AUDATA2 output when using the AUD function of the E10A. 000: PE4 I/O (port) 001: TIOC1A I/O (MTU2) 010: RXD3 input (SCIF) 101: $\overline{\text{IOIS16}}$ input (BSC)* Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	UA31 to UA0	Undefined	R/W	User Branch Destination Address When the user branch is not required, address 0 (H'00000000) must be set. The user branch destination must be an area other than the flash memory, an area other than the RAM area in which on-chip program has been transferred, or the external bus space. Note that the CPU must not branch to an area without the execution code and get out of control. The on-chip program download area and stack area must not be overwritten. If CPU runaway occurs or the download area or stack area is overwritten, the value of flash memory cannot be guaranteed. The download of the on-chip program, initialization, initiation of the programming/erasing program must not be executed in the processing of the user branch destination. Programming or erasing cannot be guaranteed when returning from the user branch destination. The program data which has already been prepared must not be programmed. Store general registers R8 to R15. General registers R0 to R7 are available without storing them. Moreover, the programming/erasing interface registers must not be written to or RAM emulation mode must not be entered in the processing of the user branch destination. After the processing of the user branch has ended, the programming/erasing program must be returned to by using the RTS instruction. For the execution intervals of the user branch processing, see note 2 (User branch processing intervals) in section 23.8.3, Other Notes.

23.4.4 RAM Emulation Register (RAMER)

When the realtime programming of the user MAT is emulated, RAMER sets the area of the user MAT which is overlapped with a part of the on-chip RAM. The RAM emulation must be executed in user mode or in user program mode.

For the division method of the user-MAT area, see table 23.7. In order to operate the emulation function certainly, the target MAT of the RAM emulation must not be accessed immediately after RAMER is programmed. If it is accessed, the normal access is not guaranteed.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	RAMS	RAM[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	RAMS	0	R/W	RAM Select Sets whether the user MAT is emulated or not. When RAMS = 1, all blocks of the user MAT are in the programming/erasing protection state. 0: Emulation is not selected Programming/erasing protection of all user-MAT blocks is invalid 1: Emulation is selected Programming/erasing protection of all user-MAT blocks is valid
2 to 0	RAM[2:0]	000	R/W	User MAT Area Select These bits are used with bit 3 to select the user-MAT area to be overlapped with the on-chip RAM. (See table 23.7.)

The difference between the erasing procedures in user program mode and user boot mode depends on whether the MAT is switched or not as shown in figure 23.15.

MAT switching is enabled by writing a specific value to FMATS. However note that while the MATs are being switched, the LSI is in an unstable state, e.g. access to a MAT is not allowed until MAT switching is completed finished, and if an interrupt occurs, from which MAT the interrupt vector is read from is undetermined. Perform MAT switching in accordance with the description in section 23.8.1, Switching between User MAT and User Boot MAT.

Except for MAT switching, the erasing procedure is the same as that in user program mode.

The area that can be executed in the steps of the user procedure program (on-chip RAM, user MAT, and external space) is shown in section 23.9.2, Areas for Storage of the Procedural Program and Data for Programming.

The FLER bit is set to 1 in the following conditions:

- When the relevant bank area of flash memory is read during programming/erasing (including a vector read or an instruction fetch)
- When a SLEEP instruction (including software standby mode) is executed during programming/erasing

Error protection is cancelled (FLER bit is cleared) only by a power-on reset.

Note that the reset signal should only be released after providing a reset input over a period longer than the normal 100 µs. Since high voltages are applied during programming/erasing of the flash memory, some voltage may still remain even after the error protection state has been entered. For this reason, it is necessary to reduce the risk of damage to the flash memory by extending the reset period so that the charge is released.

The state-transition diagram in figure 23.16 shows transitions to and from the error protection state.

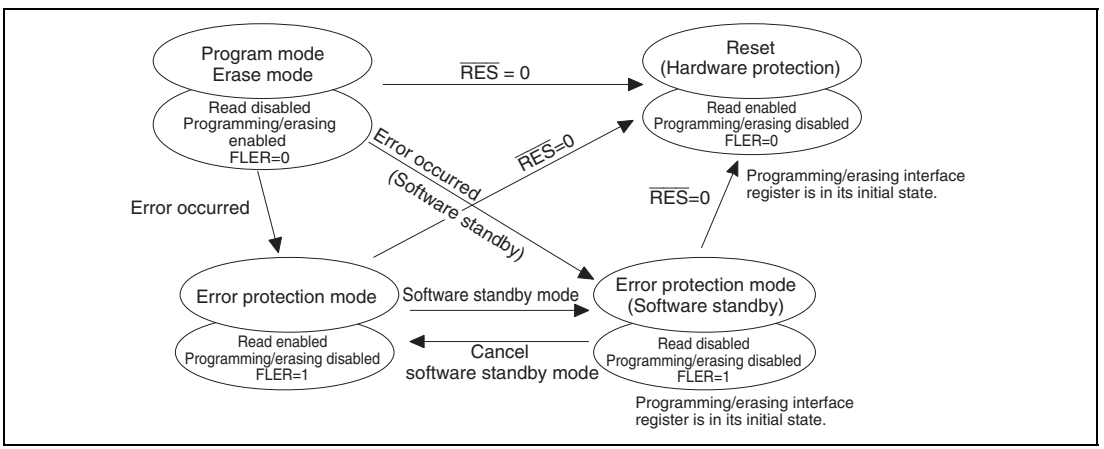


Figure 23.16 Transitions to and from Error Protection State

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
ADDR13	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D (Channel 2)
	AD1	AD0	—	—	—	—	—	—	
ADDR14	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDR15	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADCSR_2	ADF	ADIE	—	—	TRGE	—	CONADF	STC	
	CKSL[1:0]		ADM[1:0]		ADCS	CH[2:0]			
ADCR_2	—	—	ADST	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
FCCS	FWE	MAT	—	FLER	—	—	—	SCO	FLASH
FPCS	—	—	—	—	—	—	—	PPVS	
FECS	—	—	—	—	—	—	—	EPVB	
FKEY	K[7:0]								
FMATS	MS7	MS6	MS5	MS4	MS3	MS2	MS1	MS0	
FTDAR	TDER	TDA[6:0]							
DTCERA	DTCERA15	DTCERA14	DTCERA13	DTCERA12	DTCERA11	DTCERA10	DTCERA9	DTCERA8	DTC
	—	—	—	—	—	—	—	—	
DTCERB	DTCERB15	DTCERB14	DTCERB13	DTCERB12	DTCERB11	DTCERB10	DTCERB9	DTCERB8	
	DTCERB7	DTCERB6	DTCERB5	DTCERB4	DTCERB3	DTCERB2	DTCERB1	DTCERB0	
DTCERC	DTCERC15	DTCERC14	DTCERC13	DTCERC12	—	—	—	—	
	—	—	—	—	DTCERC3	DTCERC2	DTCERC1	DTCERC0	
DTCERD	DTCERD15	DTCERD14	DTCERD13	DTCERD12	DTCERD11	DTCERD10	DTCERD9	DTCERD8	
	DTCERD7	DTCERD6	DTCERD5	DTCERD4	DTCERD3	—	—	—	
DTCERE	DTCERE15	DTCERE14	DTCERE13	DTCERE12	DTCERE11	DTCERE10	DTCERE9	DTCERE8	
	DTCERE7	DTCERE6	DTCERE5	DTCERE4	—	—	—	—	
DTCCR	—	—	—	RRS	RCHNE	—	—	ERR	
DTCVBR									
					—	—	—	—	
	—	—	—	—	—	—	—	—	