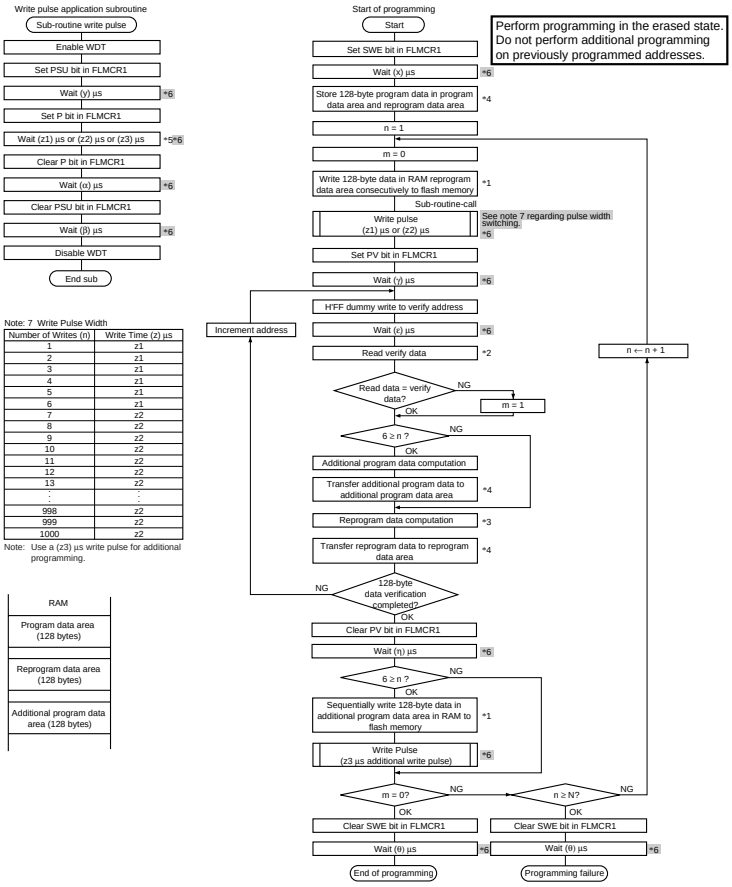
Welcome to E-XFL.COM

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	H8S/2000
Core Size	32-Bit
Speed	20MHz
Connectivity	SCI, SmartCard, UART/USART
Peripherals	DMA, POR, PWM, WDT
Number of I/O	87
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x10b SAR; D/A 2x8b
Oscillator Type	External, Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	128-BFQFP
Supplier Device Package	128-QFP (14x20)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/d12390f20iv

Item	Page	Revision (See Manual for Details)
19.18.2 Program-Verify Mode Figure 19-48 Program/Program-Verify Flowchart	639	Figure 19-48 amended, note *6 added 
22.3.6 Flash Memory Characteristics Table 22-21 Flash Memory Characteristics (HD64F2398F20, HD64F2398TE20)	724	Table 22-21 title amended
Table 22-22 Flash Memory Characteristics (HD64F2398F20T, HD64F2398TE20T)	726	Table 22-22 added

Pin No.		Pin Name					
TFP-120	FP-128B	Mode 4*[1]	Mode 5*[1]	Mode 6	Mode 7	PROM Mode	Flash Memory Programmer Mode
97	107	$P4_2$/AN2	$P4_2$/AN2	$P4_2$/AN2	$P4_2$/AN2	NC	NC
98	108	$P4_3$/AN3	$P4_3$/AN3	$P4_3$/AN3	$P4_3$/AN3	NC	NC
99	109	$P4_4$/AN4	$P4_4$/AN4	$P4_4$/AN4	$P4_4$/AN4	NC	NC
100	110	$P4_5$/AN5	$P4_5$/AN5	$P4_5$/AN5	$P4_5$/AN5	NC	NC
101	111	$P4_6$/AN6/ DA0	$P4_6$/AN6/ DA0	$P4_6$/AN6/ DA0	$P4_6$/AN6/ DA0	NC	NC
102	112	$P4_7$/AN7/ DA1	$P4_7$/AN7/ DA1	$P4_7$/AN7/ DA1	$P4_7$/AN7/ DA1	NC	NC
103	113	AV_{SS}	AV_{SS}	AV_{SS}	AV_{SS}	V_{SS}	V_{SS}
104	114	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}
105	115	$P1_7$/PO15/ TIOCB2/ TCLKD	$P1_7$/PO15/ TIOCB2/ TCLKD	$P1_7$/PO15/ TIOCB2/ TCLKD	$P1_7$/PO15/ TIOCB2/ TCLKD	NC	NC
106	116	$P1_6$/PO14/ TIOCA2	$P1_6$/PO14/ TIOCA2	$P1_6$/PO14/ TIOCA2	$P1_6$/PO14/ TIOCA2	NC	NC
107	117	$P1_5$/PO13/ TIOCB1/ TCLKC	$P1_5$/PO13/ TIOCB1/ TCLKC	$P1_5$/PO13/ TIOCB1/ TCLKC	$P1_5$/PO13/ TIOCB1/ TCLKC	NC	NC
108	118	$P1_4$/PO12/ TIOCA1	$P1_4$/PO12/ TIOCA1	$P1_4$/PO12/ TIOCA1	$P1_4$/PO12/ TIOCA1	NC	NC
109	119	$P1_3$/PO11/ TIOCD0/ TCLKB	$P1_3$/PO11/ TIOCD0/ TCLKB	$P1_3$/PO11/ TIOCD0/ TCLKB	$P1_3$/PO11/ TIOCD0/ TCLKB	NC	NC
110	120	$P1_2$/PO10/ TIOCC0/ TCLKA	$P1_2$/PO10/ TIOCC0/ TCLKA	$P1_2$/PO10/ TIOCC0/ TCLKA	$P1_2$/PO10/ TIOCC0/ TCLKA	NC	NC
111	121	$P1_1$/PO9/ TIOCB0/ $\overline{\text{DACK1}}$	$P1_1$/PO9/ TIOCB0/ $\overline{\text{DACK1}}$	$P1_1$/PO9/ TIOCB0/ $\overline{\text{DACK1}}$	$P1_1$/PO9/ TIOCB0/ $\overline{\text{DACK1}}$	NC	NC
112	122	$P1_0$/PO8/ TIOCA0/ $\overline{\text{DACK0}}$	$P1_0$/PO8/ TIOCA0/ $\overline{\text{DACK0}}$	$P1_0$/PO8/ TIOCA0/ $\overline{\text{DACK0}}$	$P1_0$/PO8/ TIOCA0/ $\overline{\text{DACK0}}$	NC	NC
113	123	MD_0	MD_0	MD_0	MD_0	V_{SS}	V_{SS}
114	124	MD_1	MD_1	MD_1	MD_1	V_{SS}	V_{SS}
115	125	MD_2	MD_2	MD_2	MD_2	V_{SS}	V_{SS}
116	126	PG_0/$\overline{\text{CAS}}$	PG_0/$\overline{\text{CAS}}$	PG_0/$\overline{\text{CAS}}$	PG_0	NC	NC
117	127	PG_1/$\overline{\text{CS3}}$	PG_1/$\overline{\text{CS3}}$	PG_1/$\overline{\text{CS3}}$	PG_1	NC	NC
118	128	PG_2/$\overline{\text{CS2}}$	PG_2/$\overline{\text{CS2}}$	PG_2/$\overline{\text{CS2}}$	PG_2	NC	NC
119	1	PG_3/$\overline{\text{CS1}}$	PG_3/$\overline{\text{CS1}}$	PG_3/$\overline{\text{CS1}}$	PG_3	NC	NC
120	2	PG_4/$\overline{\text{CS0}}$	PG_4/$\overline{\text{CS0}}$	PG_4/$\overline{\text{CS0}}$	PG_4	NC	NC
—	3	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}
—	4	NC	NC	NC	NC	NC	NC

Notes: NC pins should be connected to V_{SS} or left open.

1. In ROMless version, only modes 4 and 5 are available.
2. This pin functions as the $\overline{\text{WDTOVF}}$ pin function in ZTAT, and masked ROM products, and in the H8S/2352. In the H8S/2357F-ZTAT, the $\overline{\text{WDTOVF}}$ pin function is not available, because this pin is used as the FWE pin. In the H8S/2398, H8S/2394, H8S/2392, and H8S/2390, the $\overline{\text{WDTOVF}}$ pin function is not available, because this pin is used as the V_{CL} pin.
3. The pin names in parentheses are available other than the H8S/2357 F-ZTAT.

7.2.3 Execute Transfer Count Register (ETCR)

ETCR is a 16-bit readable/writable register that specifies the number of transfers. The setting of this register is different for sequential mode and idle mode on the one hand, and for repeat mode on the other.

(1) Sequential Mode and Idle Mode

Transfer Counter

Bit	:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETCR	:																
Initial value	:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
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

*: Undefined

In sequential mode and idle mode, ETCR functions as a 16-bit transfer counter (with a count range of 1 to 65,536). ETCR is decremented by 1 each time a transfer is performed, and when the count reaches H'0000, the DTE bit in DMABCR is cleared, and transfer ends.

(2) Repeat Mode

Transfer Number Storage

Bit	:	15	14	13	12	11	10	9	8
ETCRH	:								
Initial value	:	*	*	*	*	*	*	*	*
R/W	:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Transfer Counter

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

*: Undefined

In repeat mode, ETCR functions as transfer counter ETCRL (with a count range of 1 to 256) and transfer number storage register ETCRH. ETCRL is decremented by 1 each time a transfer is performed, and when the count reaches H'00, ETCRL is loaded with the value in ETCRH. At this point, MAR is automatically restored to the value it had when the count was started. The DTE bit in DMABCR is not cleared, and so transfers can be performed repeatedly until the DTE bit is cleared by the user.

ETCR is not initialized by a reset or in standby mode.

7.2.5 DMA Band Control Register (DMABCR)

Bit	:	15	14	13	12	11	10	9	8
DMABCRH	:	FAE1	FAE0	SAE1	SAE0	DTA1B	DTA1A	DTA0B	DTA0A
Initial value	:	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

Bit	:	7	6	5	4	3	2	1	0
DMABCRL	:	DTE1B	DTE1A	DTE0B	DTE0A	DTIE1B	DTIE1A	DTIE0B	DTIE0A
Initial value	:	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

DMABCR is a 16-bit readable/writable register that controls the operation of each DMAC channel.

DMABCR is initialized to H'0000 by a reset, and in hardware standby mode.

Bit 15—Full Address Enable 1 (FAE1): Specifies whether channel 1 is to be used in short address mode or full address mode.

Bit 15 FAE1	Description	
0	Short address mode	(Initial value)
1	Full address mode	

In short address mode, channels 1A and 1B are used as independent channels.

Bit 14—Full Address Enable 0 (FAE0): Specifies whether channel 0 is to be used in short address mode or full address mode.

Bit 14 FAE0	Description	
0	Short address mode	(Initial value)
1	Full address mode	

In short address mode, channels 0A and 0B are used as independent channels.

Bit 13—Single Address Enable 1 (SAE1): Specifies whether channel 1B is to be used for transfer in dual address mode or single address mode.

Bit 13 SAE1	Description	
0	Transfer in dual address mode	(Initial value)
1	Transfer in single address mode	

This bit is invalid in full address mode.

Bit 8—Data Transfer Acknowledge 0A (DTA0A): Enables or disables clearing, when DMA transfer is performed, of the internal interrupt source selected by the channel 0A data transfer factor setting.

Bit 8 DTA0A	Description	
0	Clearing of selected internal interrupt source at time of DMA transfer is disabled	(Initial value)
1	Clearing of selected internal interrupt source at time of DMA transfer is enabled	

Bits 7 to 4—Data Transfer Enable (DTE): When DTE = 0, data transfer is disabled and the activation source selected by the data transfer factor setting is ignored. If the activation source is an internal interrupt, an interrupt request is issued to the CPU or DTC. If the DTIE bit is set to 1 when DTE = 0, the DMAC regards this as indicating the end of a transfer, and issues a transfer end interrupt request to the CPU or DTC.

The conditions for the DTE bit being cleared to 0 are as follows:

- When initialization is performed
- When the specified number of transfers have been completed in a transfer mode other than repeat mode
- When 0 is written to the DTE bit to forcibly abort the transfer, or for a similar reason

When DTE = 1, data transfer is enabled and the DMAC waits for a request by the activation source selected by the data transfer factor setting. When a request is issued by the activation source, DMA transfer is executed.

The condition for the DTE bit being set to 1 is as follows:

- When 1 is written to the DTE bit after the DTE bit is read as 0

Bit 7—Data Transfer Enable 1B (DTE1B): Enables or disables data transfer on channel 1B.

Bit 7 DTE1B	Description	
0	Data transfer disabled	(Initial value)
1	Data transfer enabled	

Bit 6—Data Transfer Enable 1A (DTE1A): Enables or disables data transfer on channel 1A.

Bit 6 DTE1A	Description	
0	Data transfer disabled	(Initial value)
1	Data transfer enabled	

Bit 5—Data Transfer Enable 0B (DTE0B): Enables or disables data transfer on channel 0B.

Bit 5 DTE0B	Description	
0	Data transfer disabled	(Initial value)
1	Data transfer enabled	

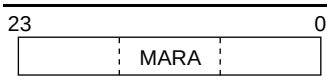
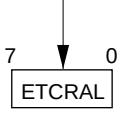
7.5.7 Block Transfer Mode

In block transfer mode, transfer is performed with channels A and B used in combination. Block transfer mode can be specified by setting the FAE bit in DMABCR and the BLKE bit in DMACRA to 1.

In block transfer mode, a transfer of the specified block size is carried out in response to a single transfer request, and this is executed the specified number of times. The transfer source is specified by MARA, and the transfer destination by MARB. Either the transfer source or the transfer destination can be selected as a block area (an area composed of a number of bytes or words).

Table 7-11 summarizes register functions in block transfer mode.

Table 7-11 Register Functions in Block Transfer Mode

Register	Function	Initial Setting	Operation
	Source address register	Start address of transfer source	Incremented/decremented every transfer, or fixed
	Destination address register	Start address of transfer destination	Incremented/decremented every transfer, or fixed
	Holds block size	Block size	Fixed
	Block size counter	Block size	Decremented every transfer; ETCRH value copied when count reaches H'00
	Block transfer counter	Number of block transfers	Decremented every block transfer; transfer ends when count reaches H'0000

Legend:
MARA: Memory address register A
MARB: Memory address register B
ETCRA: Transfer count register A
ETCRB: Transfer count register B

MARA and MARB specify the start addresses of the transfer source and transfer destination, respectively, as 24 bits. MAR can be incremented or decremented by 1 or 2 each time a byte or word is transferred, or can be fixed.

Incrementing, decrementing, or holding a fixed value can be set separately for MARA and MARB.

Whether a block is to be designated for MARA or for MARB is specified by the BLKDIR bit in DMACRA.

To specify the number of transfers, if M is the size of one block (where M = 1 to 256) and N transfers are to be performed (where N = 1 to 65,536), M is set in both ETCRAH and ETCRAL, and N in ETCRB.

Figure 7-13 illustrates operation in block transfer mode when MARB is designated as a block area.

7.5.11 DMAC Bus Cycles (Single Address Mode)

Single Address Mode (Read): Figure 7-27 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and byte-size single address mode transfer (read) is performed from external 8-bit, 2-state access space to an external device.

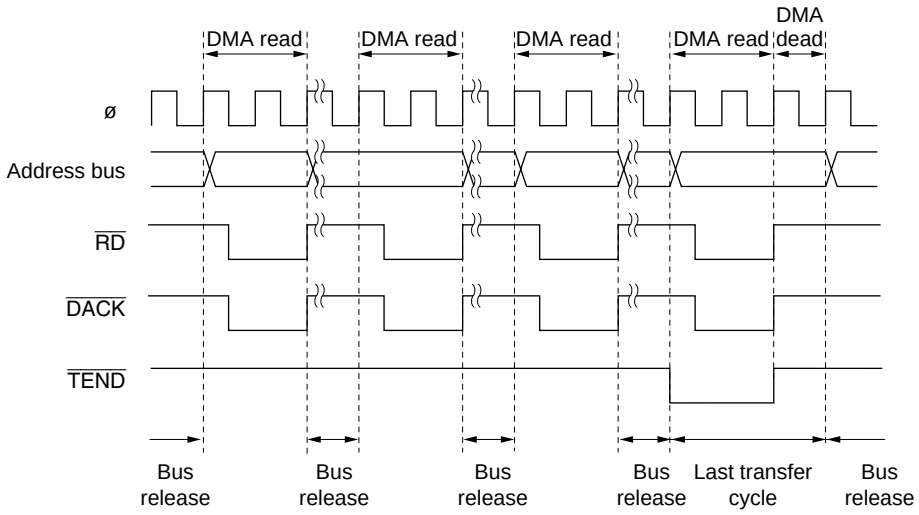


Figure 7-27 Example of Single Address Mode (Byte Read) Transfer

Figure 7-28 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and word-size single address mode transfer (read) is performed from external 8-bit, 2-state access space to an external device.

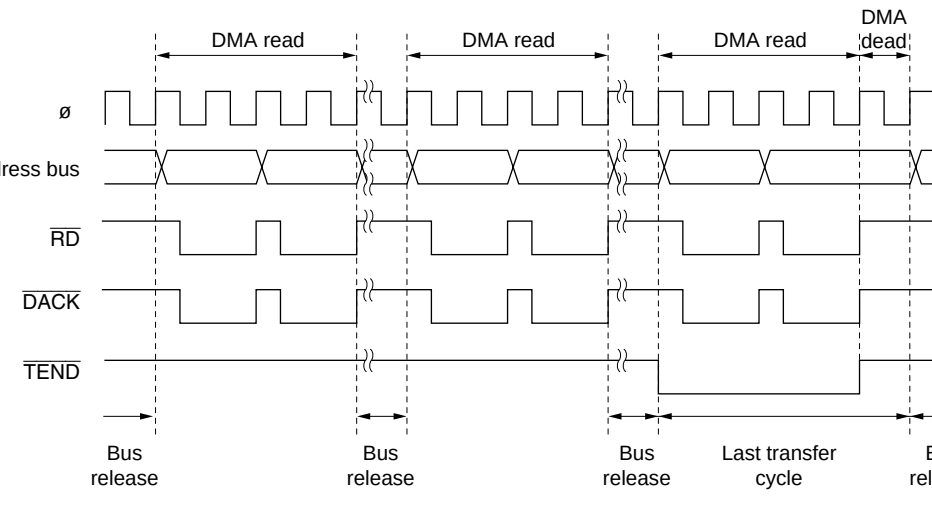


Figure 7-28 Example of Single Address Mode (Word Read) Transfer

A one-byte or one-word transfer is performed for one transfer request, and after the transfer the bus is released. While the bus is released, one or more bus cycles are inserted by the CPU or DTC.

In the transfer end cycle (the cycle in which the transfer counter reaches 0), a one-state DMA dead cycle is inserted after the DMA write cycle.

[4] Set the SCI to the appropriate receive mode. Set the RIE bit in SCR to 1 to enable the reception data full (RXI) interrupt. Since the generation of a receive error during the SCI reception operation will disable subsequent reception, the CPU should be enabled to accept receive error interrupts.

[5] Each time reception of one byte of data ends on the SCI, the RDRF flag in SSR is set to 1, an RXI interrupt is generated, and the DTC is activated. The receive data is transferred from RDR to RAM by the DTC. DAR is incremented and CRA is decremented. The RDRF flag is automatically cleared to 0.

[6] When CRA becomes 0 after the 128 data transfers have ended, the RDRF flag is held at 1, the DTCE bit is cleared to 0, and an RXI interrupt request is sent to the CPU. The interrupt handling routine should perform wrap-up processing.

(2) Chain Transfer

An example of DTC chain transfer is shown in which pulse output is performed using the PPG. Chain transfer can be used to perform pulse output data transfer and PPG output trigger cycle updating. Repeat mode transfer to the PPG's NDR is performed in the first half of the chain transfer, and normal mode transfer to the TPU's TGR in the second half. This is because clearing of the activation source and interrupt generation at the end of the specified number of transfers are restricted to the second half of the chain transfer (transfer when CHNE = 0).

[1] Perform settings for transfer to the PPG's NDR. Set MRA to source address incrementing (SM1 = 1, SM0 = 0), fixed destination address (DM1 = DM0 = 0), repeat mode (MD1 = 0, MD0 = 1), and word size (Sz = 1). Set the source side as a repeat area (DTS = 1). Set MRB to chain mode (CHNE = 1, DISEL = 0). Set the data table start address in SAR, the NDRH address in DAR, and the data table size in CRAH and CRAL. CRB can be set to any value.

[2] Perform settings for transfer to the TPU's TGR. Set MRA to source address incrementing (SM1 = 1, SM0 = 0), fixed destination address (DM1 = DM0 = 0), normal mode (MD1 = MD0 = 0), and word size (Sz = 1). Set the data table start address in SAR, the TGRA address in DAR, and the data table size in CRA. CRB can be set to any value.

[3] Locate the TPU transfer register information consecutively after the NDR transfer register information.

[4] Set the start address of the NDR transfer register information to the DTC vector address.

[5] Set the bit corresponding to TGIA in DTCER to 1.

[6] Set TGRA as an output compare register (output disabled) with TIOR, and enable the TGIA interrupt with TIER.

[7] Set the initial output value in PODR, and the next output value in NDR. Set bits in DDR and NDER for which output is to be performed to 1. Using PCR, select the TPU compare match to be used as the output trigger.

[8] Set the CST bit in TSTR to 1, and start the TCNT count operation.

[9] Each time a TGRA compare match occurs, the next output value is transferred to NDR and the set value of the next output trigger period is transferred to TGRA. The activation source TGFA flag is cleared.

[10] When the specified number of transfers are completed (the TPU transfer CRA value is 0), the TGFA flag is held at 1, the DTCE bit is cleared to 0, and a TGIA interrupt request is sent to the CPU. Termination processing should be performed in the interrupt handling routine.

(3) Software Activation

An example is shown in which the DTC is used to transfer a block of 128 bytes of data by means of software activation. The transfer source address is H'1000 and the destination address is H'2000. The vector number is H'60, so the vector address is H'04C0.

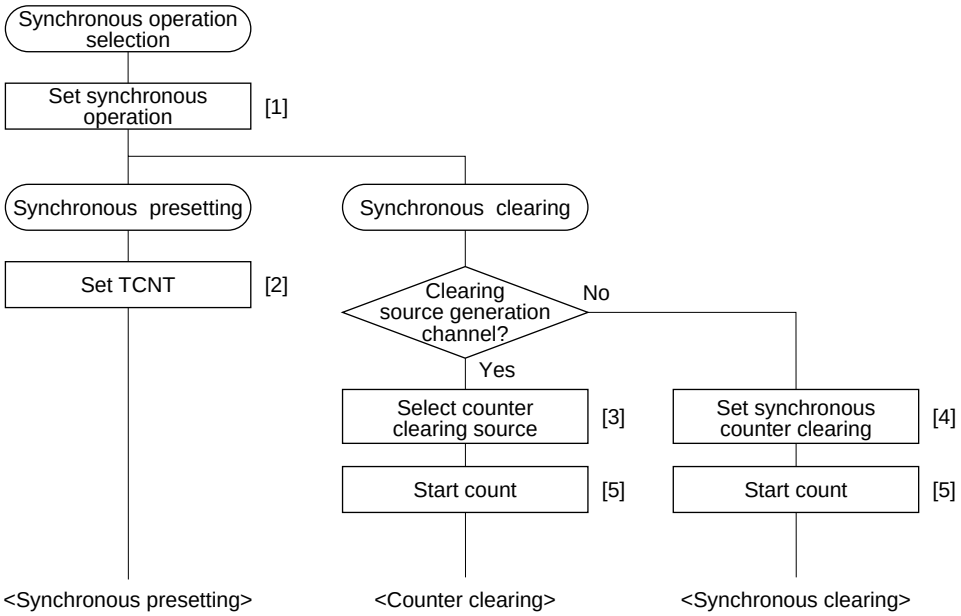
10.4.3 Synchronous Operation

In synchronous operation, the values in a number of TCNT counters can be rewritten simultaneously (synchronous presetting). Also, a number of TCNT counters can be cleared simultaneously by making the appropriate setting in TCR (synchronous clearing).

Synchronous operation enables TGR to be incremented with respect to a single time base.

Channels 0 to 5 can all be designated for synchronous operation.

Example of Synchronous Operation Setting Procedure: Figure 10-14 shows an example of the synchronous operation setting procedure.



[1] Set to 1 the SYNC bits in TSYR corresponding to the channels to be designated for synchronous operation.

[2] When the TCNT counter of any of the channels designated for synchronous operation is written to, the same value is simultaneously written to the other TCNT counters.

[3] Use bits CCLR2 to CCLR0 in TCR to specify TCNT clearing by input capture/output compare, etc.

[4] Use bits CCLR2 to CCLR0 in TCR to designate synchronous clearing for the counter clearing source.

[5] Set to 1 the CST bits in TSTR for the relevant channels, to start the count operation.

Figure 10-14 Example of Synchronous Operation Setting Procedure

Example of Synchronous Operation: Figure 10-15 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for channels 0 to 2, TGR0B compare match has been set as the channel 0 counter clearing source, and synchronous clearing has been set for the channel 1 and 2 counter clearing sources.

Three-phase PWM waveforms are output from pins TIOC0A, TIOC1A, and TIOC2A. At this time, synchronous presetting, and synchronous clearing by TGR0B compare match, is performed for channel 0 to 2 TCNT counters, and the data set in TGR0B is used as the PWM cycle.

Differences between Boot Mode and User Program Mode

Table 19-9 Differences between Boot Mode and User Program Mode

	Boot Mode	User Program Mode
Entire memory erase	Yes	Yes
Block erase	No	Yes
Programming control program*	Program/program-verify	Program/program-verify Erase/erase-verify

Note: * To be provided by the user, in accordance with the recommended algorithm.

Block Configuration: The flash memory is divided into two 32-kbyte blocks, two 8-kbyte blocks, one 16-kbyte block, one 28-kbyte block, and four 1-kbyte blocks.

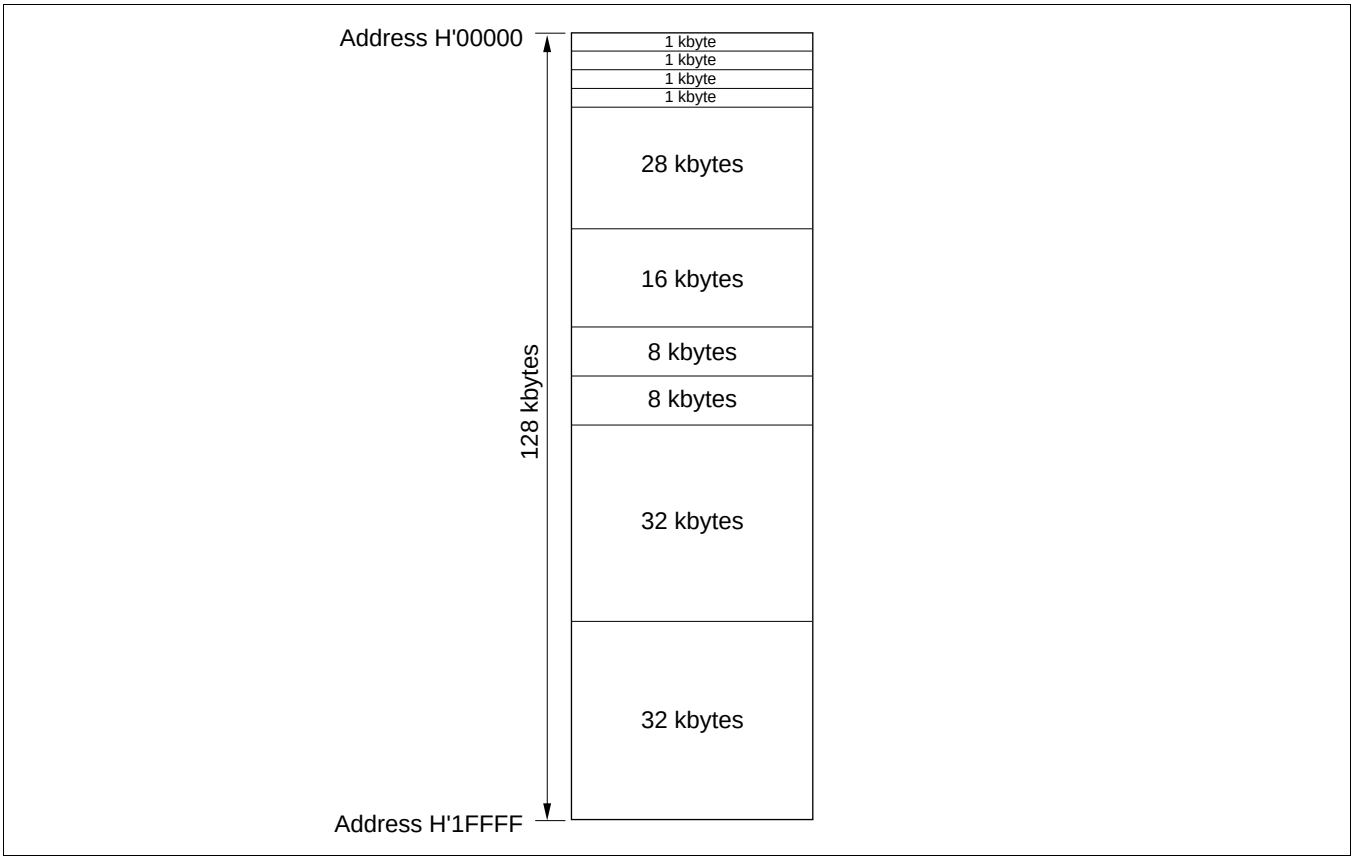


Figure 19-13 Flash Memory Block Configuration

one programming operation on a 128-byte programming unit block. Programming should be carried out with the entire programming unit block erased.

Before programming, check that the chip is correctly mounted in the PROM programmer: Overcurrent damage to the device can result if the index marks on the PROM programmer socket, socket adapter, and chip are not correctly aligned.

Do not touch the socket adapter or chip during programming: Touching either of these can cause contact faults and write errors.

Item	Symbol	Condition			Test Conditions
		Min	Max	Unit	
$\overline{WR}$ hold time	t_{WCH}	0.5 × t_{cyc} – 10	—	ns	Figure 22-8 to Figure 22-15
$\overline{CAS}$ setup time	t_{CSR}	0.5 × t_{cyc} – 10	—	ns	Figure 22-12
$\overline{WAIT}$ setup time	t_{WTS}	30	—	ns	Figure 22-10
$\overline{WAIT}$ hold time	t_{WTH}	5	—	ns	
$\overline{BREQ}$ setup time	t_{BRQS}	30	—	ns	Figure 22-16
$\overline{BACK}$ delay time	t_{BACD}	—	15	ns	
Bus-floating time	t_{BZD}	—	50	ns	
$\overline{BREQO}$ delay time	t_{BRQOD}	—	30	ns	Figure 22-17

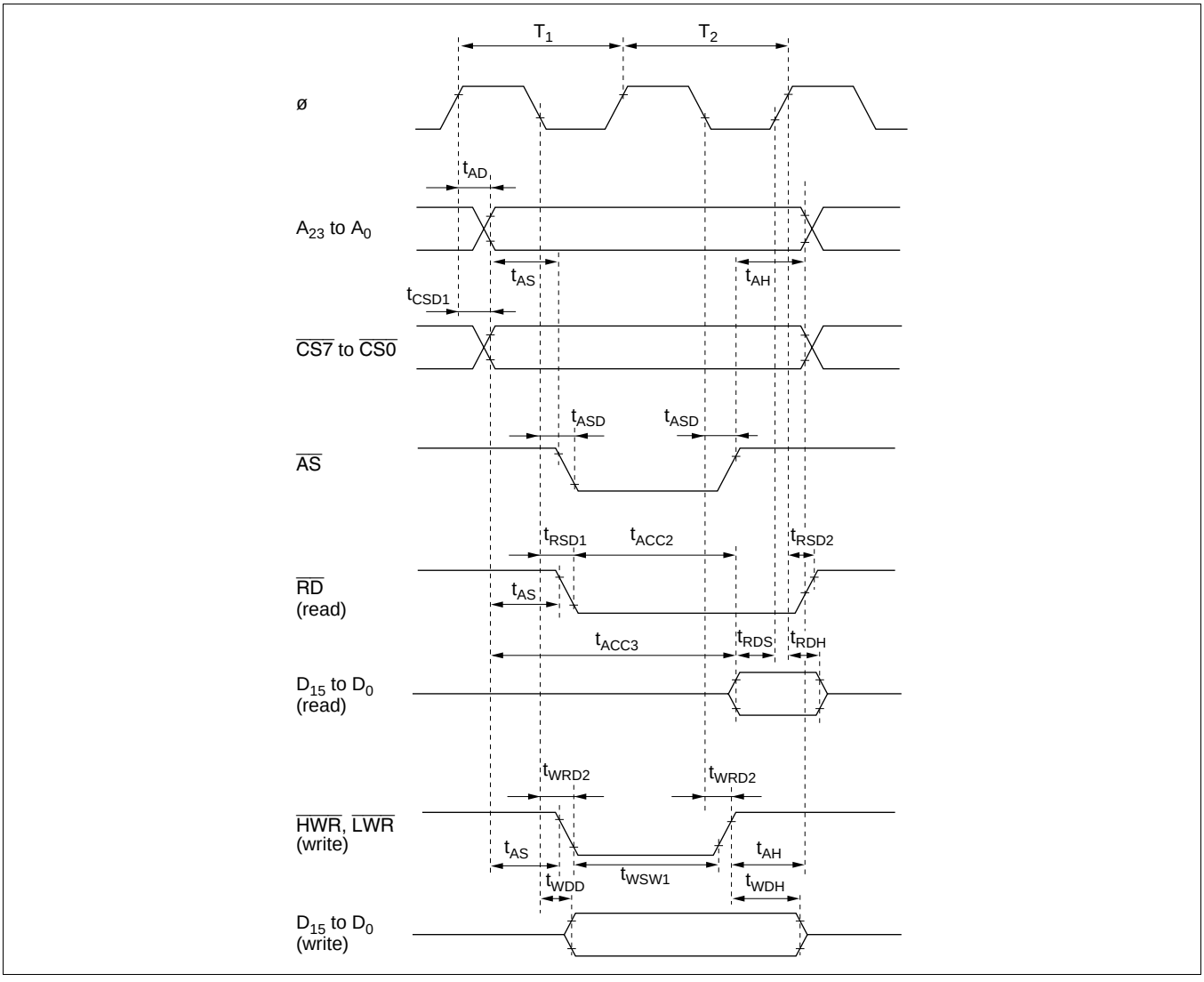


Figure 22-8 Basic Bus Timing (Two-State Access)

(3) Bus Timing

Table 22-28 lists the bus timing.

Table 22-28 Bus Timing

Condition A: V_{CC} = 2.7 to 5.5 V, AV_{CC} = 2.7 to 5.5 V, V_{ref} = 2.7 V to AV_{CC},
V_{SS} = AV_{SS} = 0 V, ø = 2 to 10 MHz, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 5.0 V ± 10%, AV_{CC} = 5.0 V ± 10%, V_{ref} = 4.5 V to AV_{CC},
V_{SS} = AV_{SS} = 0 V, ø= 2 to 20 MHz, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 to 5.5 V, AV_{CC} = 3.0 to 5.5 V, V_{ref} = 3.0 V to AV_{CC},
V_{SS} = AV_{SS} = 0 V, ø = 2 to 13 MHz, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

		Condition A		Condition B		Condition C			Test
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**	**Conditions**
Address delay time	t_{AD}	—	40	—	20	—	40	ns	Figure 22-72 to Figure 22-79
Address setup time	t_{AS}	0.5 × t_{cyc} – 30	—	0.5 × t_{cyc} – 15	—	0.5 × t_{cyc} – 30	—	ns	
Address hold time	t_{AH}	0.5 × t_{cyc} – 20	—	0.5 × t_{cyc} – 10	—	0.5 × t_{cyc} – 20	—	ns	
Precharge time	t_{PCH}	1.5 × t_{cyc} – 40	—	1.5 × t_{cyc} – 20	—	1.5 × t_{cyc} – 40	—	ns	
$\overline{CS}$ delay time 1	t_{CSD1}	—	40	—	20	—	40	ns	
$\overline{CS}$ delay time 2	t_{CSD2}	—	40	—	20	—	40	ns	
$\overline{CS}$ delay time 3	t_{CSD3}	—	40	—	25	—	40	ns	
$\overline{AS}$ delay time	t_{ASD}	—	40	—	20	—	40	ns	
$\overline{RD}$ delay time 1	t_{RSD1}	—	40	—	20	—	40	ns	
$\overline{RD}$ delay time 2	t_{RSD2}	—	40	—	20	—	40	ns	
$\overline{CAS}$ delay time	t_{CASD}	—	40	—	20	—	40	ns	
Read data setup time	t_{RDS}	30	—	15	—	30	—	ns	
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns	
Read data access time 1	t_{ACC1}	—	1.0 × t_{cyc} – 50	—	1.0 × t_{cyc} – 25	—	1.0 × t_{cyc} – 50	ns	
Read data access time 2	t_{ACC2}	—	1.5 × t_{cyc} – 50	—	1.5 × t_{cyc} – 25	—	1.5 × t_{cyc} – 50	ns	
Read data access time 3	t_{ACC3}	—	2.0 × t_{cyc} – 50	—	2.0 × t_{cyc} – 25	—	2.0 × t_{cyc} – 50	ns	
Read data access time 4	t_{ACC4}	—	2.5 × t_{cyc} – 50	—	2.5 × t_{cyc} – 25	—	2.5 × t_{cyc} – 50	ns	
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} – 50	—	3.0 × t_{cyc} – 25	—	3.0 × t_{cyc} – 50	ns	
$\overline{WR}$ delay time 1	t_{WRD1}	—	40	—	20	—	40	ns	
$\overline{WR}$ delay time 2	t_{WRD2}	—	40	—	20	—	40	ns	
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} – 40	—	1.0 × t_{cyc} – 20	—	1.0 × t_{cyc} – 40	—	ns	

TMDR4—Timer Mode Register 4 **H'FE91** **TPU4**

Bit	:	7	6	5	4	3	2	1	0
		—	—	—	—	MD3	MD2	MD1	MD0
Initial value	:	1	1	0	0	0	0	0	0
Read/Write	:	—	—	—	—	R/W	R/W	R/W	R/W

Mode

0	0	0	0	Normal operation
			1	Reserved
		1	0	PWM mode 1
			1	PWM mode 2
	1	0	0	Phase counting mode 1
			1	Phase counting mode 2
		1	0	Phase counting mode 3
			1	Phase counting mode 4
1	×	×	×	—

×: Don't care

Note: MD3 is a reserved bit. In a write, it should always be written with 0.

TMDR5—Timer Mode Register 5 **H'FEA1** **TPU5**

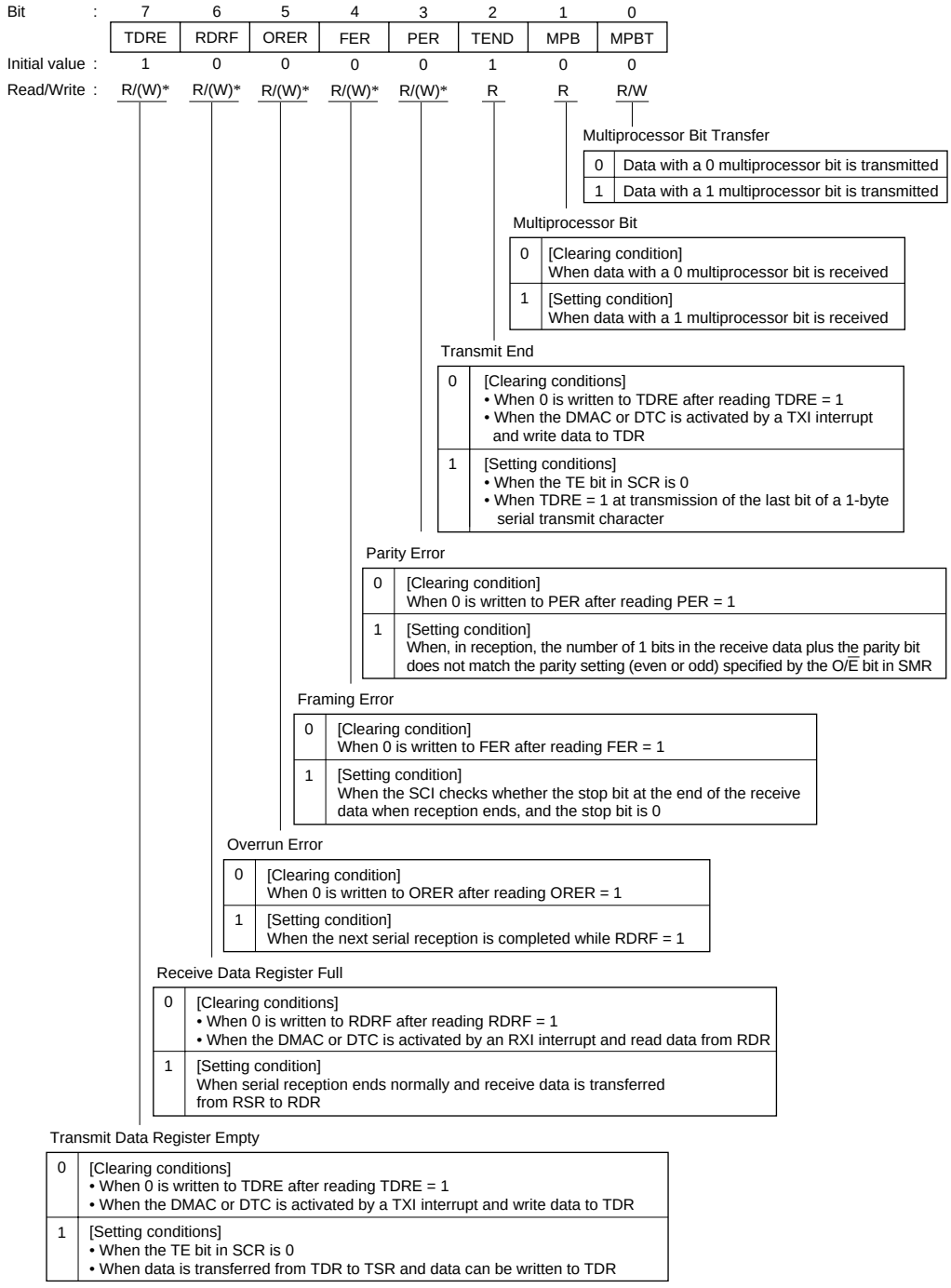
Bit	:	7	6	5	4	3	2	1	0
		—	—	—	—	MD3	MD2	MD1	MD0
Initial value	:	1	1	0	0	0	0	0	0
Read/Write	:	—	—	—	—	R/W	R/W	R/W	R/W

Mode

0	0	0	0	Normal operation
			1	Reserved
		1	0	PWM mode 1
			1	PWM mode 2
	1	0	0	Phase counting mode 1
			1	Phase counting mode 2
		1	0	Phase counting mode 3
			1	Phase counting mode 4
1	×	×	×	—

×: Don't care

Note: MD3 is a reserved bit. In a write, it should always be written with 0.



Note: * Can only be written with 0 for flag clearing.

Bit	7	6	5	4	3	2	1	0
	TDRE	RDRF	ORER	ERS	PER	TEND	MPB	MPBT
Initial value	1	0	0	0	0	1	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/W

Multiprocessor Bit Transfer

0	Data with a 0 multiprocessor bit is transmitted
1	Data with a 1 multiprocessor bit is transmitted

Multiprocessor Bit

0	[Clearing condition] When data with a 0 multiprocessor bit is received
1	[Setting condition] When data with a 1 multiprocessor bit is received

Transmit End

0	[Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When the DMAC or DTC is activated by a TXI interrupt and write data to TDR
1	[Setting conditions] • On reset, or in standby mode or module stop mode • When the TE bit in SCR is 0 and the ERS bit is 0 • When TDRE = 1 and ERS = 0 (normal transmission) 2.5 etu after a 1-byte serial character is sent when GM = 0 • When TDRE = 1 and ERS = 0 (normal transmission) 1.0 etu after a 1-byte serial character is sent when GM = 1

Note: etu: Elementary Time Unit (time for transger of 1 bit)

Parity Error

0	[Clearing condition] When 0 is written to PER after reading PER = 1
1	[Setting condition] When, in reception, the number of 1 bits in the receive data plus the parity bit does not match the parity setting (even or odd) specified by the O/$\overline{E}$ bit in SMR

Error Signal Status

0	[Clearing conditions] • On reset, or in standby mode or module stop mode • When 0 is written to ERS after reading ERS =1
1	[Setting condition] When the error signal is sampled at the low level

Note: Clearing the TE bit in SCR to 0 does not affect the ERS flag, which retains its prior state.

Overrun Error

0	[Clearing condition] When 0 is written to ORER after reading ORER = 1
1	[Setting condition] When the next serial reception is completed while RDRF = 1

Receive Data Register Full

0	[Clearing conditions] • When 0 is written to RDRF after reading RDRF = 1 • When the DMAC or DTC is activated by an RXI interrupt and read data from RDR
1	[Setting condition] When serial reception ends normally and receive data is transferred from RSR to RDR

Transmit Data Register Empty

0	[Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When the DMAC or DTC is activated by a TXI interrupt and write data to TDR
1	[Setting conditions] • When the TE bit in SCR is 0 • When data is transferred from TDR to TSR and data can be written to TDR

Note: * Can only be written with 0 for flag clearing.

Appendix G Product Code Lineup

Table G.1 H8S/2357, H8S/2352 Group Product Code Lineup

Product Type		Product Code	Mark Code	Package (Hitachi Package Code)
H8S/2357	Masked ROM	HD6432357	HD6432357TE	120-pin TQFP (TFP-120)
			HD6432357F	128-pin QFP (FP-128B)
	ZTAT	HD6472357	HD6472357TE	120-pin TQFP (TFP-120)
			HD6472357F	128-pin QFP (FP-128B)
	F-ZTAT	HD64F2357	HD64F2357TE	120-pin TQFP (TFP-120)
			HD64F2357F	128-pin QFP (FP-128B)
H8S/2352	ROMless	HD6412352	HD6412352TE	120-pin TQFP (TFP-120)
			HD6412352F	128-pin QFP (FP-128B)

Table G.2 H8S/2398, H8S/2394, H8S/2392, H8S/2390 Group Product Code Lineup

Product Type		Product Code	Mark Code	Package (Hitachi Package Code)
H8S/2398	Masked ROM	HD6432398	HD6432398TE*[1]	120-pin TQFP (TFP-120)
			HD6432398F*[1]	128-pin QFP (FP-128B)
	F-ZTAT	HD64F2398	HD64F2398TE*[1]	120-pin TQFP (TFP-120)
			HD64F2398F*[1]	128-pin QFP (FP-128B)
			HD64F2398TET	120-pin TQFP (TFP-120)
			HD64F2398FT	128-pin QFP (FP-128B)
H8S/2394	ROMless	HD6412394	HD6412394TE*[1]	120-pin TQFP (TFP-120)
			HD6412394F*[1]	128-pin QFP (FP-128B)
H8S/2392	ROMless	HD6412392	HD6412392TE	120-pin TQFP (TFP-120)
			HD6412392F	128-pin QFP (FP-128B)
H8S/2390	ROMless	HD6412390	HD6412390TE	120-pin TQFP (TFP-120)
			HD6412390F	128-pin QFP (FP-128B)