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	H8S/2600
Core Size	16-Bit
Speed	34MHz
Connectivity	I²C, LPC, SCI
Peripherals	POR, PWM, WDT
Number of I/O	95
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	40K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	144-TQFP
Supplier Device Package	144-TQFP (16x16)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/r4f2164vte34v

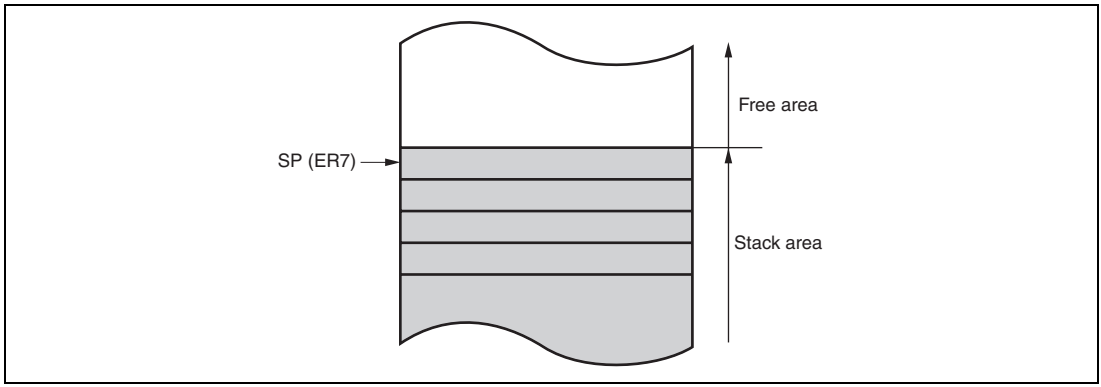

Figure 2.8 Stack

2.4.2 Program Counter (PC)

This 24-bit counter indicates the address of the next instruction the CPU will execute. The length of all CPU instructions is 2 bytes (one word), so the least significant PC bit is ignored. (When an instruction is fetched, the least significant PC bit is regarded as 0).

2.4.3 Extended Control Register (EXR)

EXR is an 8-bit register that manipulates the LDC, STC, ANDC, ORC, and XORC instructions. When these instructions, except for the STC instruction, are executed, all interrupts including NMI will be masked for three states after execution is completed.

Bit	Bit Name	Initial Value	R/W	Description
7	T	0	R/W	Trace Bit When this bit is set to 1, a trace exception is generated each time an instruction is executed. When this bit is cleared to 0, instructions are executed in sequence.
6 to 3	—	All 1	—	Reserved These bits are always read as 1.
2	I2	1	R/W	These bits designate the interrupt mask level (0 to 7). For details, refer to section 5, Interrupt Controller.
1	I1	1	R/W	
0	I0	1	R/W	

- ISCRH

Bit	Bit Name	Initial Value	R/W	Description
7	IRQ7SCB	0	R/W	IRQn Sense Control B
6	IRQ7SCA	0	R/W	IRQn Sense Control A
5	IRQ6SCB	0	R/W	00: Interrupt request generated at low level of $\overline{\text{IRQn}}$ or $\overline{\text{ExIRQn}}$ input
4	IRQ6SCA	0	R/W	01: Interrupt request generated at falling edge of $\overline{\text{IRQn}}$ or $\overline{\text{ExIRQn}}$ input
3	IRQ5SCB	0	R/W	10: Interrupt request generated at rising edge of $\overline{\text{IRQn}}$ or $\overline{\text{ExIRQn}}$ input
2	IRQ5SCA	0	R/W	11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQn}}$ or $\overline{\text{ExIRQn}}$ input
1	IRQ4SCB	0	R/W	(n = 7 to 4)
0	IRQ4SCA	0	R/W	

Note: * $\overline{\text{IRQn}}$ stands for $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ4}}$.

- ISCRL

Bit	Bit Name	Initial Value	R/W	Description
7	IRQ3SCB	0	R/W	IRQn Sense Control B
6	IRQ3SCA	0	R/W	IRQn Sense Control A
5	IRQ2SCB	0	R/W	00: Interrupt request generated at low level of $\overline{\text{IRQn}}$ or $\overline{\text{ExIRQn}}$ input
4	IRQ2SCA	0	R/W	01: Interrupt request generated at falling edge of $\overline{\text{IRQn}}$ or $\overline{\text{ExIRQn}}$ input
3	IRQ1SCB	0	R/W	10: Interrupt request generated at rising edge of $\overline{\text{IRQn}}$ or $\overline{\text{ExIRQn}}$ input
2	IRQ1SCA	0	R/W	11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQn}}$ or $\overline{\text{ExIRQn}}$ input
1	IRQ0SCB	0	R/W	(n = 3 to 0)
0	IRQ0SCA	0	R/W	

Note: * $\overline{\text{IRQn}}$ stands for $\overline{\text{IRQ3}}$ to $\overline{\text{IRQ0}}$.

Bit	Bit Name	Initial Value	R/W	Description
4	BRSTS1	1	R/W	Valid only in the normal extended mode. Burst Cycle Select 1 Selects the number of states in the burst cycle of the burst ROM interface. 0: 1 state 1: 2 states
3	BRSTS0	0	R/W	Valid only in the normal extended mode. Burst Cycle Select 0 Selects the number of words that can be accessed by burst access via the burst ROM interface. 0: Max, 4 words 1: Max, 8 words
2	—	0	R/W	Reserved The initial value should not be changed.
1	IOS1	1	R/W	IOS Select 1 and 0
0	IOS0	1	R/W	Select the address range where the $\overline{\text{IOS}}$ signal is output. See table 6.12.

8.10 Port A

Port A is an 8-bit I/O port. Port A pins can also function as the address output and event counter input pins. Port A has the following registers. PADDR and PAPIN are allocated to the same address.

- Port A data direction register (PADDR)
- Port A output data register (PAODR)
- Port A input data register (PAPIN)

8.10.1 Port A Data Direction Register (PADDR)

The individual bits of PADDR specify input or output for the port A pins.

Bit	Bit Name	Initial Value	R/W	Description
7	PA7DDR	0	W	When set to 1, the corresponding pins function as output port pins; when cleared to 0, function as input port pins. As the address of this register is the same as that of PAPIN, reading from this register indicates the state of port A.
6	PA6DDR	0	W	
5	PA5DDR	0	W	
4	PA4DDR	0	W	
3	PA3DDR	0	W	
2	PA2DDR	0	W	
1	PA1DDR	0	W	
0	PA0DDR	0	W	

8.14.4 Pin Functions

Port E pins can also function as LPC input/output pins. The pin function is switched according to whether the LPC module is enabled or disabled. The LPC is disabled when all of the bits LPC1E, LPC2E, and LPC3E in HICR0 and SCIFE in HICR5 are cleared to 0.

- PE7/SERIRQ

  The pin function is switched as shown below according to whether the LPC is enabled or disabled and the PE7DDR bit.

LPC	Disabled		Enabled
PE7DDR	0	1	X
Pin function	PE7 input pin	PE7 output pin	SERIRQ input/output pin

[Legend] X: Don't care.

- PE6/LCLK

  The pin function is switched as shown below according to whether the LPC is enabled or disabled and the PE6DDR bit.

LPC	Disabled		Enabled
PE6DDR	0	1	X
Pin function	PE6 input pin	PE6 output pin	LCLK input pin

[Legend] X: Don't care.

- PE5/$\overline{\text{LRESET}}$

  The pin function is switched as shown below according to whether the LPC is enabled or disabled and the PE5DDR bit.

LPC	Disabled		Enabled
PE5DDR	0	1	X
Pin function	PE5 input pin	PE5 output pin	$\overline{\text{LRESET}}$ input pin

[Legend] X: Don't care.

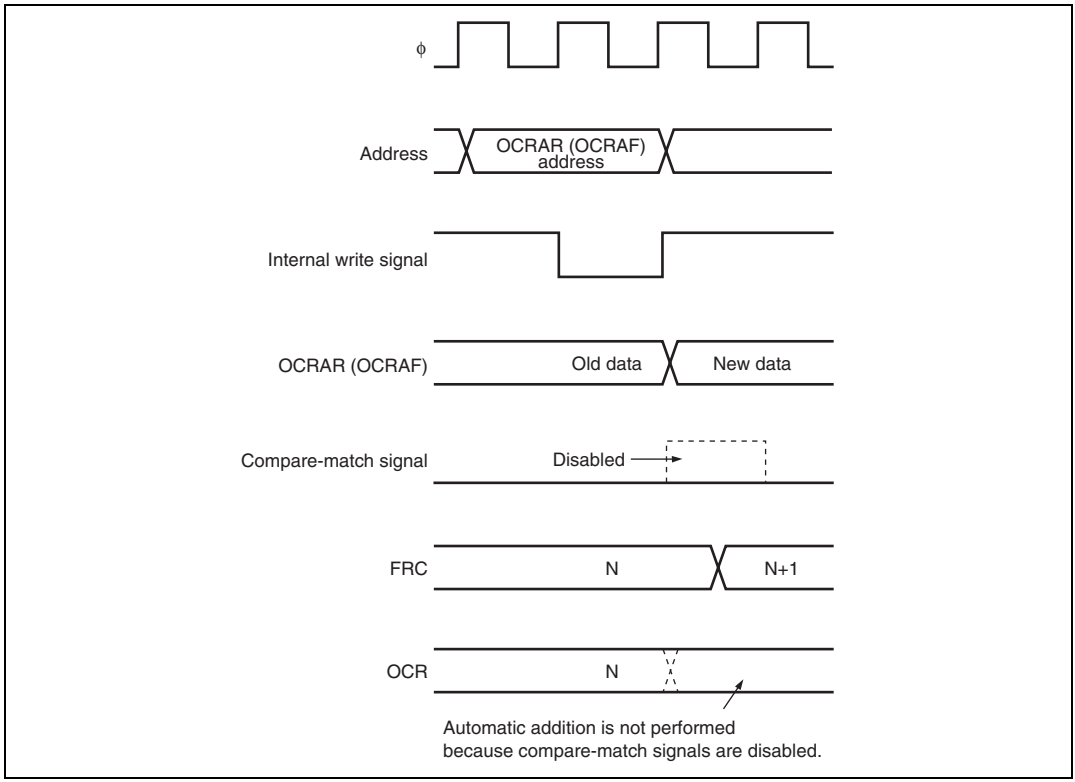

Figure 10.11 Conflict between OCR Write and Compare-Match (When Automatic Addition Function is Used)

10.5.4 Switching of Internal Clock and FRC Operation

When the internal clock is changed, the changeover may source FRC to increment. This depends on the time at which the clock is switched (bits CKS1 and CKS0 are rewritten), as shown in table 10.2.

When an internal clock is used, the FRC clock is generated on detection of the falling edge of the internal clock scaled from the system clock (ϕ). If the clock is changed when the old source is high and the new source is low, as in case no. 3 in table 10.2, the changeover is regarded as a falling edge that triggers the FRC clock, and FRC is incremented. Switching between an internal clock and external clock can also source FRC to increment.

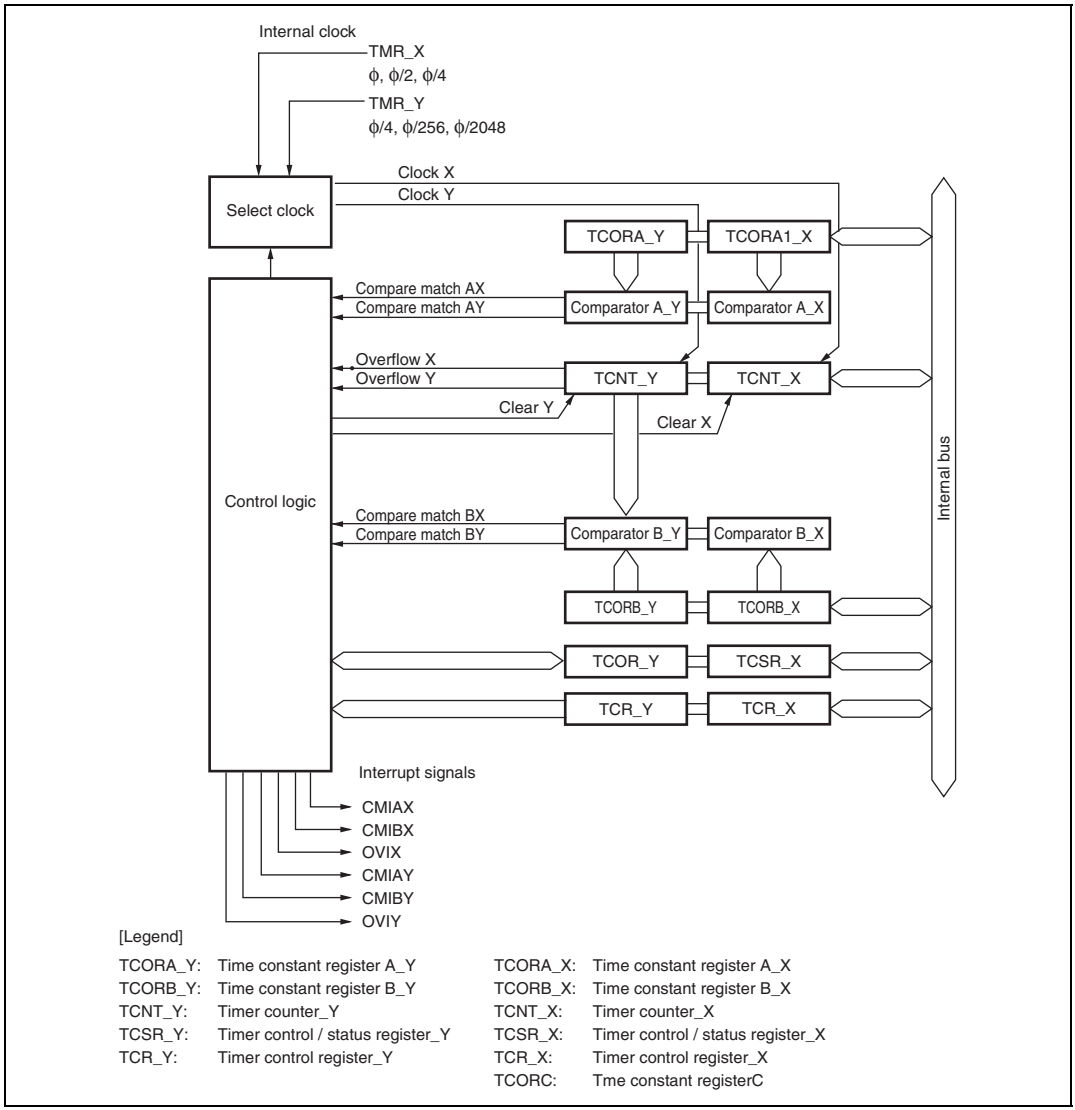

Figure 11.2 Block Diagram of 8-Bit Timer (TMR_Y and TMR_X)

14.2 Register Descriptions

The CRC operation circuit has the following registers.

- CRC control register (CRCCR)
- CRC data input register (CRCDIR)
- CRC data output register (CRCDOR)

14.2.1 CRC Control Register (CRCCR)

CRCCR initializes the CRC operation circuit, switches the operation mode, and selects the generating polynomial.

Bit	Bit Name	Initial Value	R/W	Description
7	DORCLR	0	W	CRCDOR Clear Setting this bit to 1 clears CRCDOR to H′0000.
6 to 3	—	All 0	R	Reserved The initial value should not be changed.
2	LMS	0	R/W	CRC Operation Switch Selects CRC code generation for LSB-first or MSB-first communication. 0: Performs CRC operation for LSB-first communication. The lower byte (bits 7 to 0) is first transmitted when CRCDOR contents (CRC code) are divided into two bytes to be transmitted in two parts. 1: Performs CRC operation for MSB-first communication. The upper byte (bits 15 to 8) is first transmitted when CRCDOR contents (CRC code) are divided into two bytes to be transmitted in two parts.
1	G1	0	R/W	CRC Generating Polynomial Select
0	G0	0	R/W	These bits select the polynomial. 00: Reserved 01: $X^8 + X^2 + X + 1$ 10: $X^{16} + X^{15} + X^2 + 1$ 11: $X^{16} + X^{12} + X^5 + 1$

(4) Suspension of Data Transmission

Figure 15.9 shows an example of the data transmission suspension flowchart.

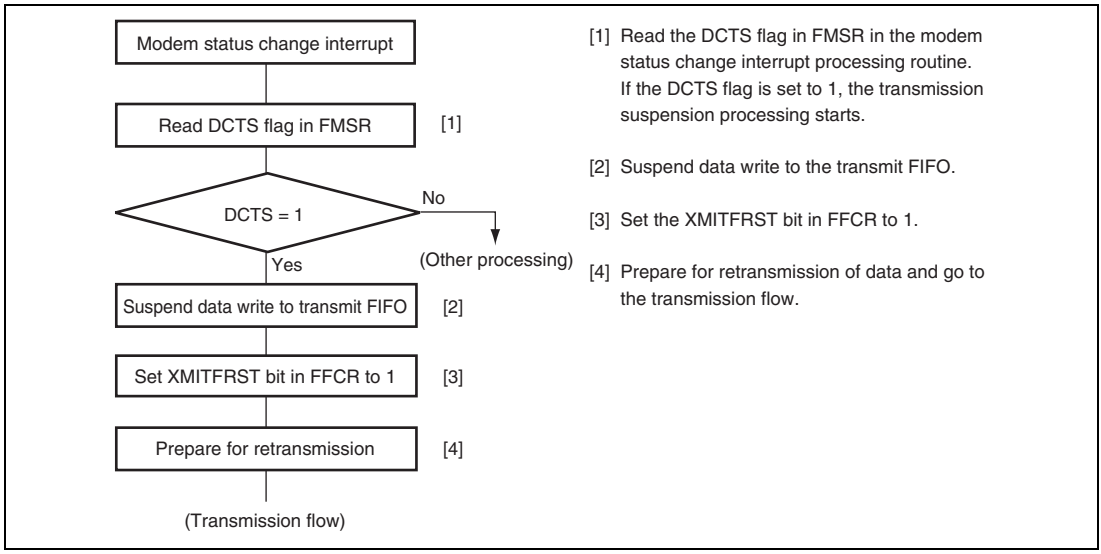

Figure 15.9 Example of Data Transmission Suspension Flowchart

Receive operations can be performed continuously by repeating steps 9 to 13.

14. Confirm that the ICDRF flag is set to 1, and read ICDR.
15. Clear the IRIC flag.

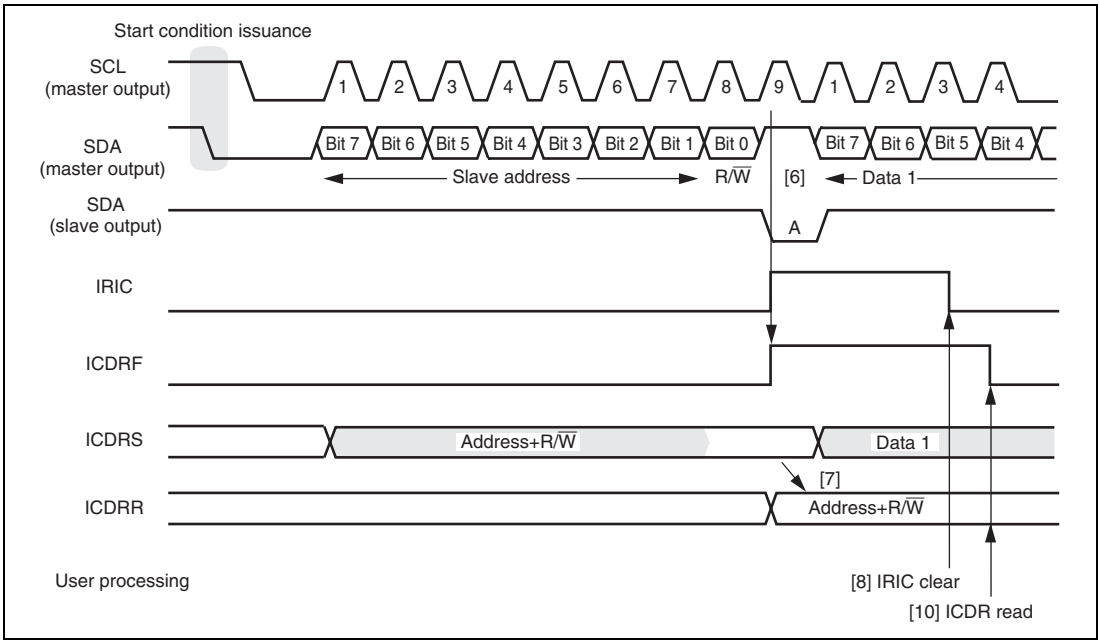

Figure 17.21 Slave Receive Mode Operation Timing Example (1)
(MLS = ACKB = 0, HNDS = 0)

- STR3 (TWRE = 0 and SELSTR3 = 1)

Bit	Bit Name	Initial Value	R/W Slave	R/W Host	Description
7	DBU37	0	R/W	R	Defined by User
6	DBU36	0	R/W	R	The user can use these bits as necessary.
5	DBU35	0	R/W	R	
4	DBU34	0	R/W	R	
3	C/$\overline{D}$3	0	R	R	Command/Data Flag When the host writes to IDR3, bit 2 of the I/O address is written into this bit to indicate whether IDR3 contains data or a command. 0: Content of input data register (IDR3) is a data 1: Content of input data register (IDR3) is a command
2	DBU32	0	R/W	R	Defined by User The user can use this bit as necessary.
1	IBF3A	0	R	R	Input Data Register Full Indicates whether or not there is receive data in IDR3. This bit is an internal interrupt source to the slave (this LSI). 0: There is not receive data in IDR3 [Clearing condition] When the slave reads IDR3 1: There is receive data in IDR3 [Setting condition] When the host writes to IDR3 in an I/O write cycle

The serialized interrupt transfer cycle frame configuration is as follows. Two of the states comprising each frame are the recover state in which the SERIRQ signal is returned to the 1-level at the end of the frame, and the turnaround state in which the SERIRQ signal is not driven. The recover state must be driven by the host or slave that was driving the preceding state.

Table 18.10 Serialized Interrupt Transfer Cycle Frame Configuration

	Serial Interrupt Transfer Cycle			
Frame Count	**Contents**	**Drive Source**	**Number of States**	**Notes**
0	Start	Slave Host	6	In quiet mode only, slave drive possible in the first state, then next 3 states 0-driven by host
1	IRQ0	Slave	3	Drive impossible
2	IRQ1	Slave	3	Drive possible in LPC channel 1 and SCIF
3	SMI	Slave	3	Drive possible in LPC channels 2, 3, and SCIF
4	IRQ3	Slave	3	Drive possible in SCIF or by IRQ3E
5	IRQ4	Slave	3	Drive possible in SCIF or by IRQ4E
6	IRQ5	Slave	3	Drive possible in SCIF or by IRQ5E
7	IRQ6	Slave	3	Drive possible in LPC channels 2, 3, and SCIF
8	IRQ7	Slave	3	Drive possible in SCIF or by IRQ7E
9	IRQ8	Slave	3	Drive possible in SCIF or by IRQ8E
10	IRQ9	Slave	3	Drive possible in LPC channels 2, 3, and SCIF
11	IRQ10	Slave	3	Drive possible in LPC channels 2, 3, and SCIF
12	IRQ11	Slave	3	Drive possible in LPC channels 2, 3, and SCIF
13	IRQ12	Slave	3	Drive possible in LPC channel 1 and SCIF
14	IRQ13	Slave	3	Drive possible in SCIF or by IRQ13E
15	IRQ14	Slave	3	Drive possible in SCIF or by IRQ14E
16	IRQ15	Slave	3	Drive possible in SCIF or by IRQ15E
17	IOCHCK	Slave	3	Drive impossible
18	Stop	Host	Undefined	First, 1 or more idle states, then 2 or 3 states 0-driven by host 2 states: Quiet mode next 3 states: Continuous mode next

Bit	Bit Name	Initial Value	R/W	Description
6	MD	—	R/W	Programming Mode Related Setting Error Detect Returns the check result that a high level signal is input to the FWE pin and the error protection state is not entered. When the low level signal is input to the FWE pin or the error protection state is entered, 1 is written to this bit. The state can be confirmed with the FWE and FLER bits in FCCS. For conditions to enter the error protection state, see section 21.5.3, Error Protection. 0: FWE and FLER settings are normal (FWE = 1, FLER = 0) 1: Programming cannot be performed (FWE = 0 or FLER = 1)
5	EE	—	R/W	Programming Execution Error Detect 1 is returned to this bit when the specified data could not be written because the user MAT was not erased. If this bit is set to 1, there is a high possibility that the user MAT is partially rewritten. In this case, after removing the error factor, erase the user MAT. If FMATS is set to H'AA and the user boot MAT is selected, an error occurs when programming is performed. In this case, both the user MAT and user boot MAT are not rewritten. Programming of the user boot MAT should be performed in boot mode or programmer mode. 0: Programming has ended normally 1: Programming has ended abnormally (programming result is not guaranteed)
4	FK	—	R/W	Flash Key Register Error Detect Returns the check result of the value of FKEY before the start of the programming processing. 0: FKEY setting is normal (FKEY = H'5A) 1: FKEY setting is error (FKEY = value other than H′5A)
3	—	—	—	Unused Returns 0.

21.4.4 Procedure Program and Storable Area for Programming Data

In the descriptions in the previous section, the programming/erasing procedure programs and storable areas for program data are assumed to be in the on-chip RAM. However, the program and the data can be stored in and executed from other areas, such as part of flash memory which is not to be programmed or erased, or somewhere in the external address space.

(1) Conditions that Apply to Programming/Erasing

1. The on-chip programming/erasing program is downloaded from the address in the on-chip RAM specified by FTDAR, therefore, this area is not available for use.
2. The on-chip programming/erasing program will use 128 bytes at the maximum as a stack. So, make sure that this area is secured.
3. Download by setting the SCO bit to 1 will lead to switching of the MAT. If, therefore, this operation is used, it should be executed from the on-chip RAM.
4. The flash memory is accessible until the start of programming or erasing, that is, until the result of downloading has been determined. When in a mode in which the external address space is not accessible, such as single-chip mode, the required procedure programs, NMI handling vector and NMI handler should be transferred to the on-chip RAM before programming/erasing of the flash memory starts.
5. The flash memory is not accessible during programming/erasing operations, therefore, the operation program is downloaded to the on-chip RAM to be executed. The NMI-handling vector and programs such as that which activate the operation program, and NMI handler should thus be stored in on-chip memory other than flash memory or the external address space.
6. After programming/erasing, the flash memory should be inhibited until FKEY is cleared.

   The reset state ($\overline{RES}$ = 0) must be in place for more than 100 μs when the LSI mode is changed to reset on completion of a programming/erasing operation.

   Transitions to the reset state, and hardware standby mode are inhibited during programming/erasing. When the reset signal is accidentally input to the chip, a longer period in the reset state than usual (100 μs) is needed before the reset signal is released.
7. Switching of the MATs by FMATS should be needed when programming/erasing of the user boot MAT is operated in user-boot mode. The program which switches the MATs should be executed from the on-chip RAM. See section 21.6, Switching between User MAT and User Boot MAT. Please make sure you know which MAT is selected when switching between them.
8. When the data storable area indicated by programming parameter FMPDR is within the flash memory area, an error will occur even when the data stored is normal. Therefore, the data

Table 22.3 Correspondence between Pins and Boundary Scan Register

Pin No.	Pin Name	Input/Output	Bit No.
	from ETDI		
1	VCC		—
			—
			—
2	P45	Input	336
		Enable	335
		Output	334
3	P46	Input	333
		Enable	332
		Output	331
4	P47	Input	330
		Enable	329
		Output	328
5	P56	Input	327
		Enable	326
		Output	325
6	P57	Input	324
		Enable	323
		Output	322
7	VSS	—	—
		—	—
		—	—
8	$\overline{\text{RES}}$	—	—
		—	—
		—	—
9	MD1	Input	321
		—	—
		—	—
10	MD0	Input	320
		—	—
		—	—
11	NMI	Input	319
		—	—
		—	—
12	$\overline{\text{STBY}}$	—	—
		—	—
		—	—
13	VCL	—	—
		—	—
		—	—
14	$\overline{\text{MD2}}$	Input	318
		—	—
		—	—
15	P51	Input	317
		Enable	316
		Output	315
16	P50	Input	314
		Enable	313
		Output	312
17	P97	Input	311
		Enable	310
		Output	309
18	P96	Input	308
		Enable	307
		Output	306
19	P95	Input	305
		Enable	304
		Output	303

- MSTPCRA

Bit	Bit Name	Initial Value	R/W	Corresponding Module
7 to 3	MSTPA7 to MSTPA3	All 0	R/W	Reserved The initial values should not be changed.
2	MSTPA2	0	R/W	14-bit PWM timer (PWMX_1)
1	MSTPA1	0	R/W	14-bit PWM timer (PWMX_0)
0	MSTPA0	0	R/W	Reserved The initial value should not be changed.

MSTPCR sets operation and stop by the combination of bits as follows:

MSTPCRH (bit 3) MSTP11	MSTPCRA (bit 2) MSTPA2	Function
0	0	14-bit PWM timer (PWMX_1) operates.
0	1	14-bit PWM timer (PWMX_1) stops.
1	x	Reserved

MSTPCRH (bit 3) MSTP11	MSTPCRA (bit 1) MSTPA1	Function
0	0	14-bit PWM timer (PWMX_0) operates.
0	1	14-bit PWM timer (PWMX_0) stops.
1	x	Reserved

Note: Bit 3 of MSTPCRH is the module stop bit for PWMX_0 and PWMX_1.

[Legend] X: Don't care

Register Abbreviation	Reset	WDT Reset	High-Speed/ Medium-Speed	Sleep	Module Stop	Software Standby	Hardware Standby	Module
ADDRH	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	ADC
ADCSR	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
ADCR	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
SMR0	Initialized	Initialized	—	—	—	—	Initialized	SMX
SMR1	Initialized	Initialized	—	—	—	—	Initialized	
P6PCR	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	PORT
PINFNCR	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
P4PCR	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
ICCR_3	Initialized	Initialized	—	—	—	—	Initialized	
ICSR_3	Initialized	Initialized	—	—	—	—	Initialized	
ICDR_3	—	—	—	—	—	—	—	IIC_3
SARX_3	Initialized	Initialized	—	—	—	—	Initialized	
ICMR_3	Initialized	Initialized	—	—	—	—	Initialized	
SAR_3	Initialized	Initialized	—	—	—	—	Initialized	
ICCR_2	Initialized	Initialized	—	—	—	—	Initialized	
ICSR_2	Initialized	Initialized	—	—	—	—	Initialized	
ICDR_2	—	—	—	—	—	—	—	IIC_2
SARX_2	Initialized	Initialized	—	—	—	—	Initialized	
ICMR_2	Initialized	Initialized	—	—	—	—	Initialized	
SAR_2	Initialized	Initialized	—	—	—	—	Initialized	
DADRA_1	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	PWMX_1
DACR_1	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
DADRB_1	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
DACNT_1	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
CRCCR	Initialized	Initialized	—	—	—	—	Initialized	CRC
CRCDIR	Initialized	Initialized	—	—	—	—	Initialized	
CRCDOR	Initialized	Initialized	—	—	—	—	Initialized	
ICXR_0	Initialized	Initialized	—	—	—	—	Initialized	IIC_0
ICXR_1	Initialized	Initialized	—	—	—	—	Initialized	IIC_1
ICSMBCR	Initialized	Initialized	—	—	—	—	Initialized	IIC
ICXR_2	Initialized	Initialized	—	—	—	—	Initialized	IIC_2

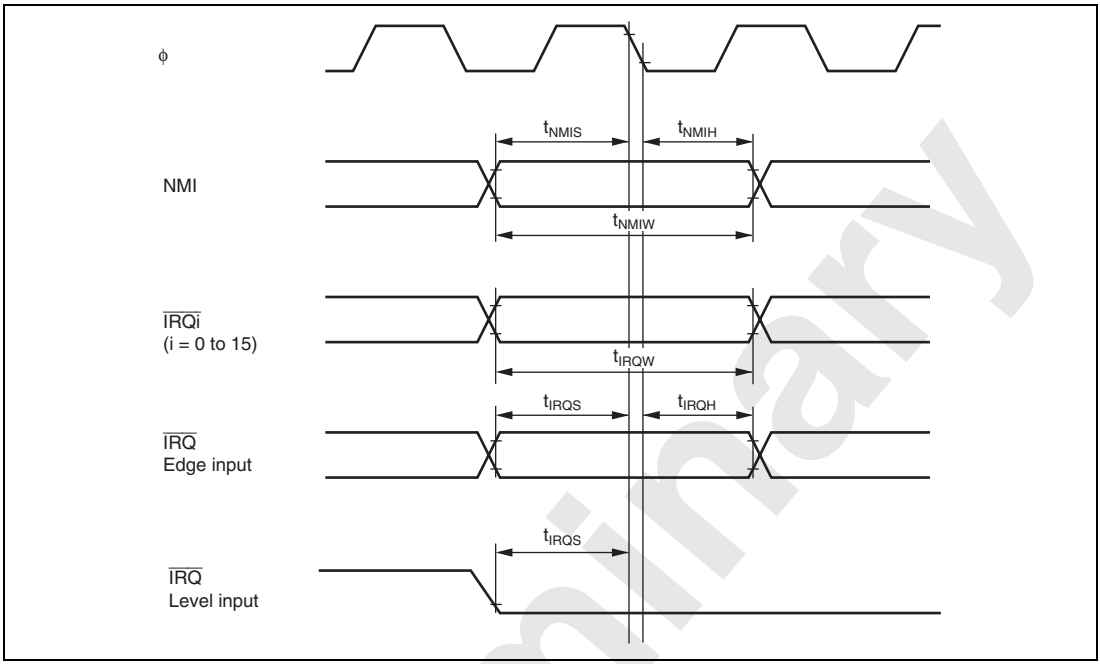

Figure 26.11 Interrupt Input Timing

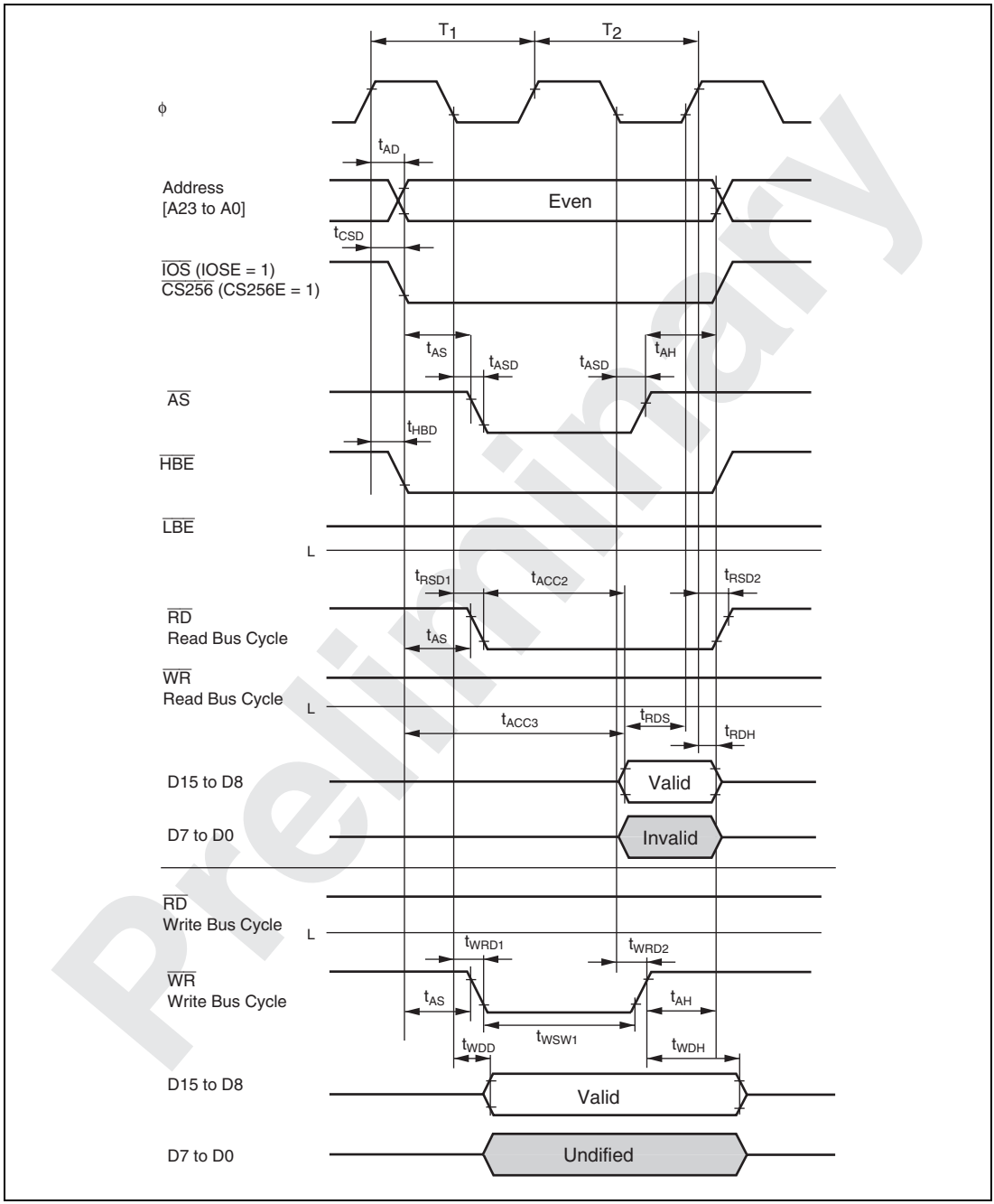

Figure 26.15 Even Byte Access (ADMXE = 0)