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	Active
Core Processor	H8/300H
Core Size	16-Bit
Speed	25MHz
Connectivity	SCI, SmartCard
Peripherals	DMA, PWM, WDT
Number of I/O	70
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	100-BFQFP
Supplier Device Package	100-QFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df3029f25v

Table 2.8 Branching Instructions

Instruction	Size	Function		
Bcc	—	Branches to a specified address if address specified condition is met. The branching conditions are listed below.		
		Mnemonic	**Description**	**Condition**
		BRA (BT)	Always (true)	Always
		BRN (BF)	Never (false)	Never
		BHI	High	$C \vee Z = 0$
		BLS	Low or same	$C \vee Z = 1$
		Bcc (BHS)	Carry clear (high or same)	$C = 0$
		BCS (BLO)	Carry set (low)	$C = 1$
		BNE	Not equal	$Z = 0$
		BEQ	Equal	$Z = 1$
		BVC	Overflow clear	$V = 0$
		BVS	Overflow set	$V = 1$
		BPL	Plus	$N = 0$
		BMI	Minus	$N = 1$
		BGE	Greater or equal	$N \oplus V = 0$
		BLT	Less than	$N \oplus V = 1$
		BGT	Greater than	$Z \vee (N \oplus V) = 0$
		BLE	Less or equal	$Z \vee (N \oplus V) = 1$
JMP	—	Branches unconditionally to a specified address		
BSR	—	Branches to a subroutine at a specified address		
JSR	—	Branches to a subroutine at a specified address		
RTS	—	Returns from a subroutine		

Bit 3—Priority Level A3 (IPRA3): Selects the priority level of WDT, DRAM interface, and A/D converter interrupt requests.

Bit 3 IPRA3	Description
0	WDT, DRAM interface, and A/D converter interrupt requests have priority level 0 (low priority) (Initial value)
1	WDT, DRAM interface, and A/D converter interrupt requests have priority level 1 (high priority)

Bit 2—Priority Level A2 (IPRA2): Selects the priority level of 16-bit timer channel 0 interrupt requests.

Bit 2 IPRA2	Description
0	16-bit timer channel 0 interrupt requests have priority level 0 (low priority) (Initial value)
1	16-bit timer channel 0 interrupt requests have priority level 1 (high priority)

Bit 1—Priority Level A1 (IPRA1): Selects the priority level of 16-bit timer channel 1 interrupt requests.

Bit 1 IPRA1	Description
0	16-bit timer channel 1 interrupt requests have priority level 0 (low priority) (Initial value)
1	16-bit timer channel 1 interrupt requests have priority level 1 (high priority)

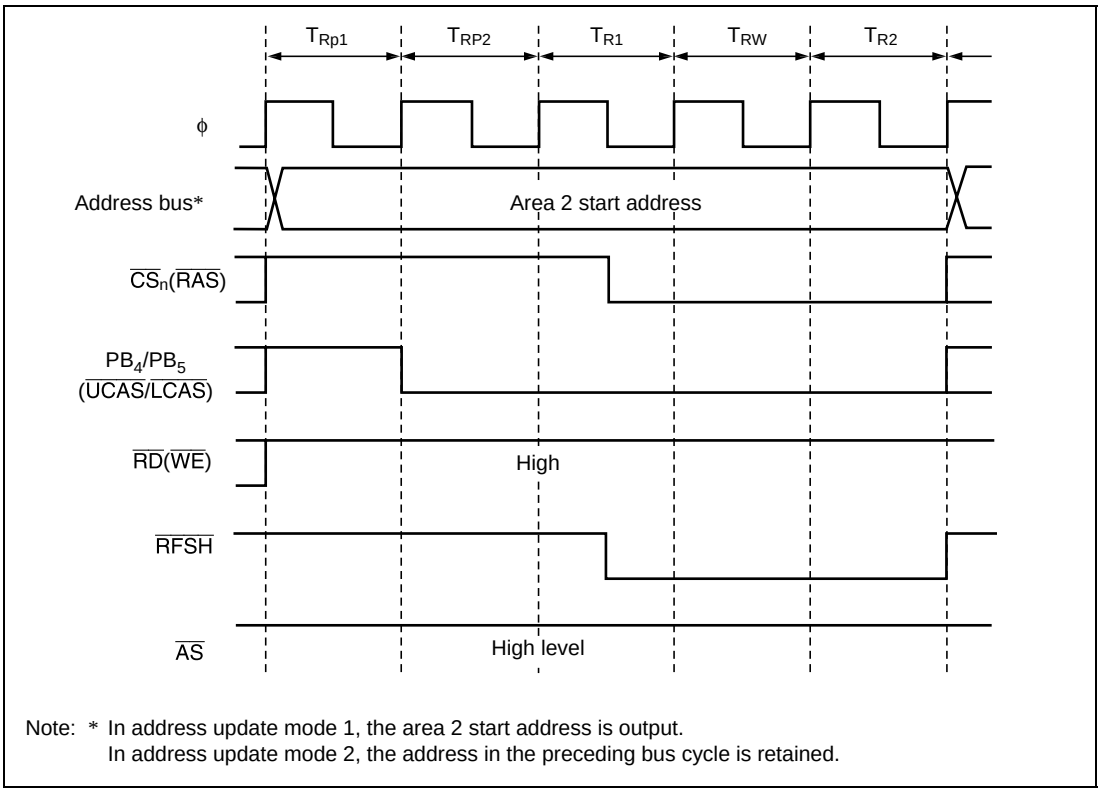
Bit 0—Priority Level A0 (IPRA0): Selects the priority level of 16-bit timer channel 2 interrupt requests.

Bit 0 IPRA0	Description
0	16-bit timer channel 2 interrupt requests have priority level 0 (low priority) (Initial value)
1	16-bit timer channel 2 interrupt requests have priority level 1 (high priority)

Use the RLW bit in DRCRB to adjust the $\overline{RAS}$ signal width. A single refresh wait state (T_{RW}) can be inserted between the T_{R1} state and T_{R2} state by setting the RLW bit to 1.

The RLW bit setting is valid only for CBR refresh cycles, and does not affect DRAM read/write cycles. The number of states in the CBR refresh cycle is not affected by the settings in ASTCR, WCRH, or WCRL, or by the state of the $\overline{WAIT}$ pin.

Figure 6.29 shows the timing when the TPC bit and RLW bit are both set to 1.



Note: * In address update mode 1, the area 2 start address is output.
In address update mode 2, the address in the preceding bus cycle is retained.

Figure 6.29 CBR Refresh Timing (CSEL = 0, TPC = 1, RLW = 1)

DRAM must be refreshed immediately after powering on in order to stabilize its internal state. When using the H8/3029 CAS-before-RAS refresh function, therefore, a DRAM stabilization period should be provided by means of interrupts by another timer module, or by counting the number of times bit 7 (CMF) of RTMCSR is set, for instance, immediately after bits DRAS2 to DRAS0 have been set in DRCRA.

Self-Refreshing: A self-refresh mode (battery backup mode) is provided for DRAM as a kind of standby mode. In this mode, refresh timing and refresh addresses are generated within the DRAM. The H8/3029 has a function that places the DRAM in self-refresh mode when the chip enters software standby mode.

Table 8.1 Port Functions

Port	Description	Pins	Expanded Modes					Single-Chip Modes
			Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 7
Port 1	• 8-bit I/O port Can drive LEDs	$P1_7$ to $P1_0$/ A_7 to A_0	Address output pins (A_7 to A_0)				Address output (A_7 to A_0) and generic input DDR = 0: generic input DDR = 1: address output	Generic input/output
Port 2	• 8-bit I/O port Built-in input pull-up transistors Can drive LEDs	$P2_7$ to $P2_0$/ A_{15} to A_8	Address output pins (A_{15} to A_8)				Address output (A_{15} to A_8) and generic input DDR = 0: generic input DDR = 1: address output	Generic input/output
Port 3	• 8-bit I/O port	$P3_7$ to $P3_0$/ D_{15} to D_8	Data input/output (D_{15} to D_8)					Generic input/output
Port 4	• 8-bit I/O port • Built-in input pull-up transistors	$P4_7$ to $P4_0$/ D_7 to D_0	Data input/output (D_7 to D_0) and 8-bit generic input/output 8-bit bus mode: generic input/output 16-bit bus mode: data input/output					Generic input/output
Port 5	• 4-bit I/O port • Built-in input pull-up transistors • Can drive LEDs	$P5_3$ to $P5_0$/ A_{19} to A_{16}	Address output (A_{19} to A_{16})				Address output (A_{19} to A_{16}) and 4-bit generic input DDR = 0: generic input DDR = 1: address output	Generic input/output
Port 6	• 7-bit I/O port and 1-bit input port	$P6_7$/ϕ	Clock output (ϕ) and generic input					
		$P6_6$/$\overline{LWR}$ $P6_5$/$\overline{HWR}$ $P6_4$/$\overline{RD}$ $P6_3$/$\overline{AS}$	Bus control signal output ($\overline{LWR}$, $\overline{HWR}$, $\overline{RD}$, $\overline{AS}$)					Generic input/output
		$P6_2$/$\overline{BACK}$ $P6_1$/$\overline{BREQ}$ $P6_0$/$\overline{WAIT}$	Bus control signal input/output ($\overline{BACK}$, $\overline{BREQ}$, $\overline{WAIT}$) and 3-bit generic input/output					
Port 7	• 8-bit input port	$P7_7$/AN_7/DA_1 $P7_6$/AN_6/DA_0	Analog input (AN_7, AN_6) to A/D converter, analog output (DA_1, DA_0) from D/A converter, and generic input					
		$P7_5$ to $P7_0$/ AN_5 to AN_0	Analog input (AN_5 to AN_0) to A/D converter, and generic input					
Port 8	• 5-bit I/O port • $P8_2$ to $P8_0$ have Schmitt inputs	$P8_4$/$\overline{CS_0}$	DDR = 0: generic input DDR = 1 (reset value): $\overline{CS_0}$ output				DDR = 0 (reset value): generic input DDR = 1: $\overline{CS_0}$ output	Generic input/output
		$P8_3$/$\overline{IRQ_3}$/ $\overline{CS_1}$/$\overline{ADTRG}$	$\overline{IRQ_3}$ input, $\overline{CS_1}$ output, external trigger input ($\overline{ADTRG}$) to A/D converter, and generic input DDR = 0 (after reset): generic input DDR = 1: $\overline{CS_1}$ output					$\overline{IRQ_3}$ input, external trigger input ($\overline{ADTRG}$) to A/D converter, and generic input/output
		$P8_2$/$\overline{IRQ_2}$/$\overline{CS_2}$ $P8_1$/$\overline{IRQ_1}$/$\overline{CS_3}$	$\overline{IRQ_2}$ and $\overline{IRQ_1}$ input, $\overline{CS_2}$ and $\overline{CS_3}$ output, and generic input* DDR = 0 (reset value): generic input DDR = 1: $\overline{CS_2}$ and $\overline{CS_3}$ output					$\overline{IRQ_2}$ and $\overline{IRQ_1}$ input and generic input/output
		$P8_0$/$\overline{IRQ_0}$ /$\overline{RFSH}$	$\overline{IRQ_0}$ input, $\overline{RFSH}$ output, and generic input/output					$\overline{IRQ_0}$ input and generic input/output

Note: * $P8_1$ can be used as an output port by making a setting in DRCRA.

8.11 Port A

8.11.1 Overview

Port A is an 8-bit input/output port that is also used for output (TP_7 to TP_0) from the programmable timing pattern controller (TPC), input and output, ($TIOCB_2$, $TIOCA_2$, $TIOCB_1$, $TIOCA_1$, $TIOCB_0$, $TIOCA_0$, TCLKD, TCLKC, TCLKB, TCLKA) by the 16-bit timer, input (TCLKD, TCLKC, TCLKB, TCLKA) to the 8-bit timer, output ($\overline{TEND}_1$, $\overline{TEND}_0$) from the DMA controller (DMAC), and address output (A_{23} to A_{20}). A reset or hardware standby transition leaves port A as an input port, except that in modes 3 and 4, one pin is always used for A_{20} output. See table 8.19 to 8.21 for the selection of pin functions.

Usage of pins for TPC, 16-bit timer, 8-bit timer, and DMAC input and output is described in the sections on those modules. For output of address bits A_{23} to A_{20} in modes 3, 4, and 5, see section 6.2.4, Bus Release Control Register (BRCR). Pins not assigned to any of these functions are available for generic input/output. Figure 8.10 shows the pin configuration of port A.

Pins in port A can drive one TTL load and a 30-pF capacitive load. They can also drive a darlington transistor pair. Port A has Schmitt-trigger inputs.

Table 8.21 Port A Pin Functions (Modes 1 to 5, 7)

Pin	Pin Functions and Selection Method
$PA_3/TP_3/TIOCB_0/TCLKD$	Bit PWM0 in TMDR, bits IOB2 to IOB0 in TIOR0, bits TPSC2 to TPSC0 in 16TCR2 to 16TCR0 of the 16-bit timer, bits CKS2 to CKS0 in 8TCR2 of the 8-bit timer, bit NDER3 in NDERA, and bit PA_3DDR select the pin function as follows.

16-bit timer channel 0 settings	(1) in table below	(2) in table below		
PA_3DDR	—	0	1	1
NDER3	—	—	0	1
Pin function	$TIOCB_0$ output	PA_3 input	PA_3 output	TP_3 output
		$TIOCB_0$ input*1		
	TCLKD input*2			

Notes: *1 $TIOCB_0$ input when IOB2 = 1 and PWM0 = 0.

*2 TCLKD input when TPSC2 = TPSC1 = TPSC0 = 1 in any of 16TCR2 to 16TCR0, or bits CKS2 to CKS0 in 8TCR2 are as shown in (3) in the table below.

16-bit timer channel 0 settings	(2)	(1)		(2)
IOB2	0			1
IOB1	0	0	1	—
IOB0	0	1	—	—

8-bit timer channel 2 settings	(4)	(3)		
CKS2	0	1		
CKS1	—	0		1
CKS0	—	0	1	—

Pin	Pin Functions and Selection Method

$PB_1/TP_9/TMIO_1/\overline{DREQ}_0/\overline{CS}_6$

Bits OIS3/2 and OS1/0 in 8TCSR1, bits CCLR1 and CCLR0 in TCR1, bit CS6E in CSCR, bit NDER9 in NDERB, and bit PB_1DDR select the pin function as follows.

OIS3/2 and OS1/0	All 0				Not all 0
CS6E	0			1	—
PB_1DDR	0	1	1	—	—
NDER9	—	0	1	—	—
Pin function	PB_1 input	PB_1 output	TP_9 output	$\overline{CS}_6$ output	$TMIO_1$ output
	$TMIO_1$ input*1				
	$\overline{DREQ}_0$ input*2				

Notes: *1 $TMIO_1$ input when CCLR1 = CCLR0 = 1.

*2 When an external request is specified as a DMAC activation source, $\overline{DREQ}_0$ input regardless of bits OIS3/2 and OS1/0, bits CCLR1/0, bit CS6E, bit NDER9, and bit PB_1DDR.

$PB_0/TP_8/TMO_0/\overline{CS}_7$

Bits OIS3/2 and OS1/0 in 8TCSR0, bit CS7E in CSCR, bit NDER8 in NDERB, and bit PB_0DDR select the pin function as follows.

OIS3/2 and OS1/0	All 0				Not all 0
CS7E	0			1	—
PB_0DDR	0	1	1	—	—
NDER8	—	0	1	—	—
Pin function	PB_0 input	PB_0 output	TP_8 output	$\overline{CS}_7$ output	TMO_0 output

9.1.4 Register Configuration

Table 9.3 summarizes the 16-bit timer registers.

Table 9.3 16-bit timer Registers

Channel	Address*[1]	Name	Abbre-viation	R/W	Initial Value
Common	H'FFF60	Timer start register	TSTR	R/W	H'F8
	H'FFF61	Timer synchro register	TSNC	R/W	H'F8
	H'FFF62	Timer mode register	TMDR	R/W	H'98
	H'FFF63	Timer output level setting register	TOLR	W	H'C0
	H'FFF64	Timer interrupt status register A	TISRA	R/(W)*[2]	H'88
	H'FFF65	Timer interrupt status register B	TISRB	R/(W)*[2]	H'88
	H'FFF66	Timer interrupt status register C	TISRC	R/(W)*[2]	H'88
0	H'FFF68	Timer control register 0	16TCR0	R/W	H'80
	H'FFF69	Timer I/O control register 0	TIOR0	R/W	H'88
	H'FFF6A	Timer counter 0H	16TCNT0H	R/W	H'00
	H'FFF6B	Timer counter 0L	16TCNT0L	R/W	H'00
	H'FFF6C	General register A0H	GRA0H	R/W	H'FF
	H'FFF6D	General register A0L	GRA0L	R/W	H'FF
	H'FFF6E	General register B0H	GRB0H	R/W	H'FF
	H'FFF6F	General register B0L	GRB0L	R/W	H'FF
1	H'FFF70	Timer control register 1	16TCR1	R/W	H'80
	H'FFF71	Timer I/O control register 1	TIOR1	R/W	H'88
	H'FFF72	Timer counter 1H	16TCNT1H	R/W	H'00
	H'FFF73	Timer counter 1L	16TCNT1L	R/W	H'00
	H'FFF74	General register A1H	GRA1H	R/W	H'FF
	H'FFF75	General register A1L	GRA1L	R/W	H'FF
	H'FFF76	General register B1H	GRB1H	R/W	H'FF
	H'FFF77	General register B1L	GRB1L	R/W	H'FF

10.7.7 Contention between 8TCNT Byte Write and Increment in 16-Bit Count Mode (Cascaded Connection)

If an increment pulse occurs in the T_3 state of an 8TCNT byte write cycle in 16-bit count mode, the counter write takes priority and the byte data for which the write was performed is not incremented. The byte data for which a write was not performed is incremented. Figure 10.24 shows the timing when an increment pulse occurs in the T_2 state of a byte write to 8TCNT (upper byte). If an increment pulse occurs in the T_2 state, on the other hand, the increment takes priority.

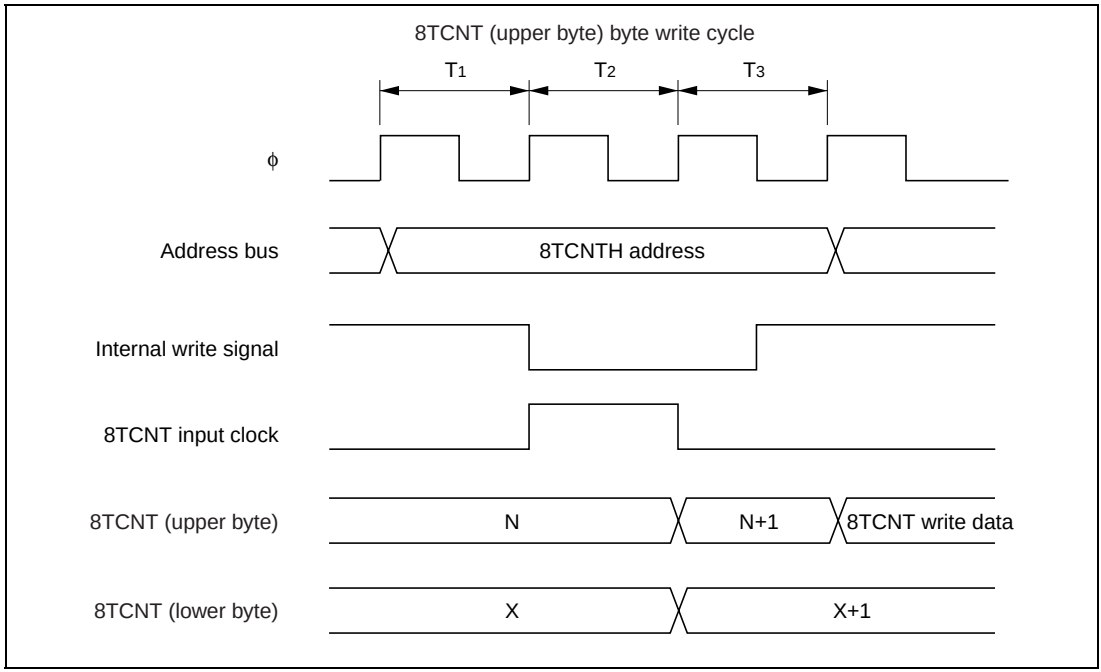
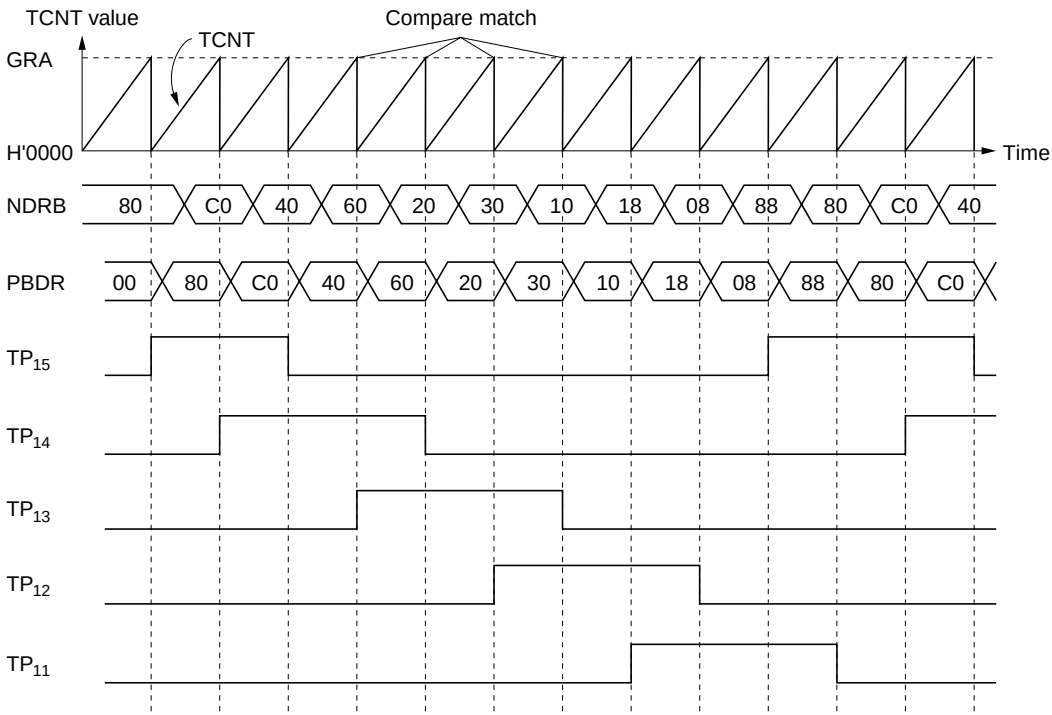


Figure 10.24 Contention between 8TCNT Byte Write and Increment in 16-Bit Count Mode

Example of Normal TPC Output (Example of Five-Phase Pulse Output): Figure 11.5 shows an example in which the TPC is used for cyclic five-phase pulse output.



- The 16-bit timer channel to be used as the output trigger channel is set up so that GRA is an output compare register and the counter will be cleared by compare match A. The trigger period is set in GRA. The IMIEA bit is set to 1 in TISRA to enable the compare match A interrupt.
- H'F8 is written in PBDDR and NDERB, and bits G3CMS1, G3CMS0, G2CMS1, and G2CMS0 are set in TPCR to select compare match in the 16-bit timer channel set up in step 1 as the output trigger. Output data H'80 is written in NDRB.
- The timer counter in this 16-bit timer channel is started. When compare match A occurs, the NDRB contents are transferred to PBDR and output. The compare match/input capture A (IMFA) interrupt service routine writes the next output data (H'C0) in NDRB.
- Five-phase overlapping pulse output (one or two phases active at a time) can be obtained by writing H'40, H'60, H'20, H'30, H'10, H'18, H'08, H'88… at successive IMFA interrupts. If the DMAC is set for activation by this interrupt, pulse output can be obtained without loading the CPU.

Figure 11.5 Normal TPC Output Example (Five-Phase Pulse Output)

Bit 7—Transmit Data Register Empty (TDRE): Indicates that the SCI has loaded transmit data from TDR into TSR and the next serial data can be written in TDR.

Bit 7 TDRE	Description
0	TDR contains valid transmit data [Clearing conditions] Read TDRE when TDRE = 1, then write 0 in TDRE The DMAC writes data in TDR
1	TDR does not contain valid transmit data (Initial value) [Setting conditions] The chip is reset or enters standby mode The TE bit in SCR is cleared to 0 TDR contents are loaded into TSR, so new data can be written in TDR

Bit 6—Receive Data Register Full (RDRF): Indicates that RDR contains new receive data.

Bit 6 RDRF	Description
0	RDR does not contain new receive data (Initial value) [Clearing conditions] The chip is reset or enters standby mode Read RDRF when RDRF = 1, then write 0 in RDRF The DMAC reads data from RDR
1	RDR contains new receive data [Setting condition] Serial data is received normally and transferred from RSR to RDR

Note: The RDR contents and the RDRF flag are not affected by detection of receive errors or by clearing of the RE bit to 0 in SCR. They retain their previous values. If the RDRF flag is still set to 1 when reception of the next data ends, an overrun error will occur and the receive data will be lost.

Clock: An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCI transmit/receive clock. The clock source is selected by the C/$\overline{A}$ bit in SMR and bits CKE1 and CKE0 in SCR. For details of SCI clock source selection, see table 13.9.

When an external clock is input at the SCK pin, it must have a frequency 16 times the desired bit rate.

When the SCI is operated on an internal clock, it can output a clock signal at the SCK pin. The frequency of this output clock is equal to the bit rate. The phase is aligned as shown in figure 13.3 so that the rising edge of the clock occurs at the center of each transmit data bit.

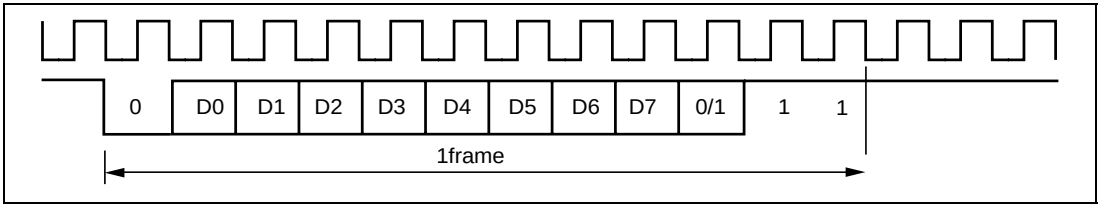


Figure 13.3 Phase Relationship between Output Clock and Serial Data (Asynchronous Mode)

Transmitting and Receiving Data:

- SCI Initialization (Asynchronous Mode): Before transmitting or receiving data, clear the TE and RE bits to 0 in SCR, then initialize the SCI as follows.

  When changing the communication mode or format, always clear the TE and RE bits to 0 before following the procedure given below. Clearing TE to 0 sets the TDRE flag to 1 and initializes TSR. Clearing RE to 0, however, does not initialize the RDRF, PER, FER, and ORER flags, or RDR, which retain their previous contents.

  When an external clock is used the clock should not be stopped during initialization or subsequent operation, since operation will be unreliable in this case.

14.1.2 Block Diagram

Figure 14.1 shows a block diagram of the smart card interface.

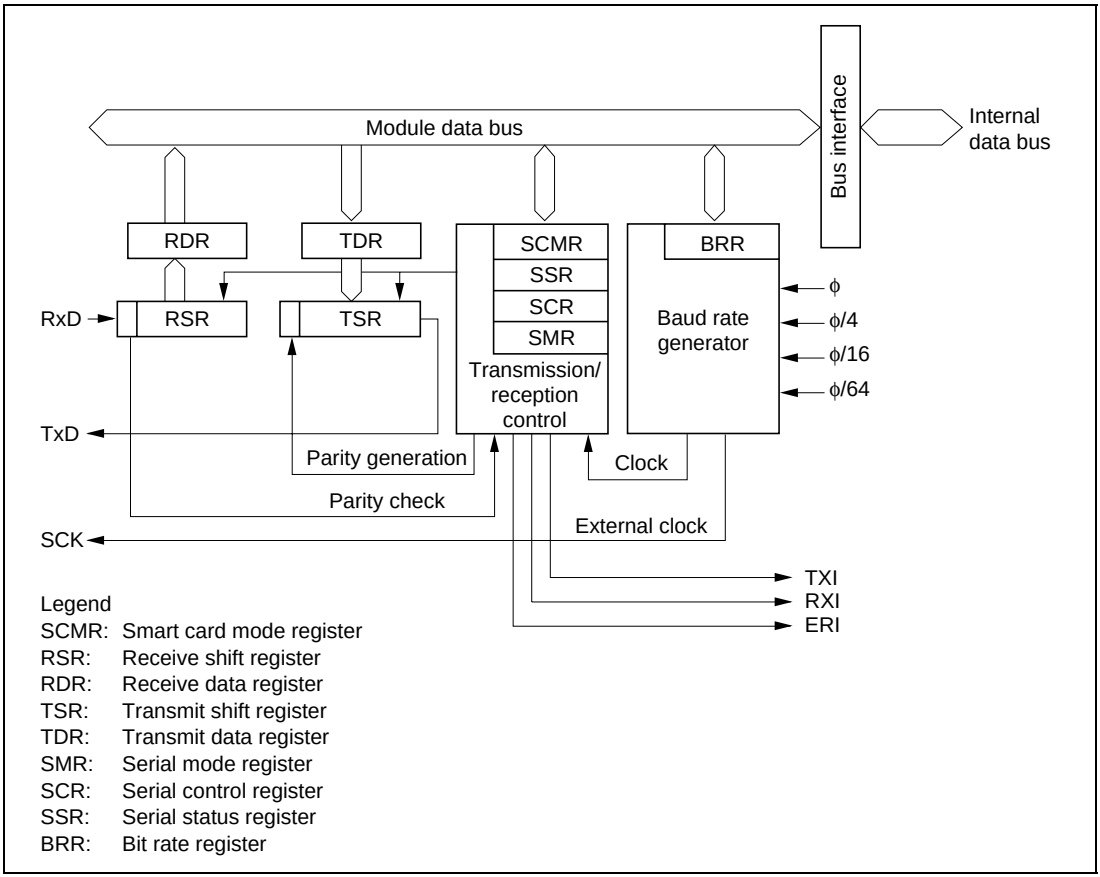


Figure 14.1 Block Diagram of Smart Card Interface

14.1.3 Pin Configuration

Table 14.1 shows the smart card interface pins.

Table 14.1 Smart Card Interface Pins

Pin Name	Abbreviation	I/O	Function
Serial clock pin	SCK	I/O	Clock input/output
Receive data pin	RxD	Input	Receive data input
Transmit data pin	TxD	Output	Transmit data output

(a) Flash programming/erasing frequency parameter (FPEFEQ: general register ER0 of CPU)

This parameter sets the operating frequency of the CPU.

For the range of the operating frequency of this LSI, see section 21.4.1, Clock Timing.

Bit :	31	30	29	28	27	26	25	24
	0	0	0	0	0	0	0	0
Bit :	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0
Bit :	15	14	13	12	11	10	9	8
	F15	F14	F13	F12	F11	F10	F9	F8
Bit :	7	6	5	4	3	2	1	0
	F7	F6	F5	F4	F3	F2	F1	F0

Bits 31 to 16—Unused: Only 0 may be written to these bits.

Bits 15 to 0—Frequency Set (F15 to F0): Set the operating frequency of the CPU. The setting value must be calculated as the following methods.

1. The operating frequency which is shown in MHz units must be rounded in a number to three decimal places and be shown in a number of two decimal places.
2. The centuplicated value is converted to the binary digit and is written to the FPEFEQ parameter (general register R0). For example, when the operating frequency of the CPU is 25.000 MHz, the value is as follows.
    - The number to three decimal places of 25.000 is rounded and the value is thus 25.00.
    - The formula that $25.00 \times 100 = 2500$ is converted to the binary digit and b'0000,1001,1100,0100 (H'09C4) is set to R0.

Table 18.25 AC Characteristics Status Read Mode

Condition : V_{CC} = 3.0 V to 3.6 V, V_{SS} = 0 V, T_a = 25°C ± 5°C

Code	Symbol	Min	Max	Unit	Note
Command write cycle	t_{nxtc}	20		µs	
$\overline{CE}$ hold time	t_{ceh}	0		ns	
$\overline{CE}$ setup time	t_{ces}	0		ns	
Data hold time	t_{dh}	50		ns	
Data setup time	t_{ds}	50		ns	
Programming pulse width	t_{wep}	70		ns	
$\overline{OE}$ output delay time	t_{oe}		150	ns	
Disable delay time	t_{df}		100	ns	
$\overline{CE}$ output delay time	t_{ce}		150	ns	
$\overline{WE}$ rise time	t_r		30	ns	
$\overline{WE}$ fall time	t_f		30	ns	

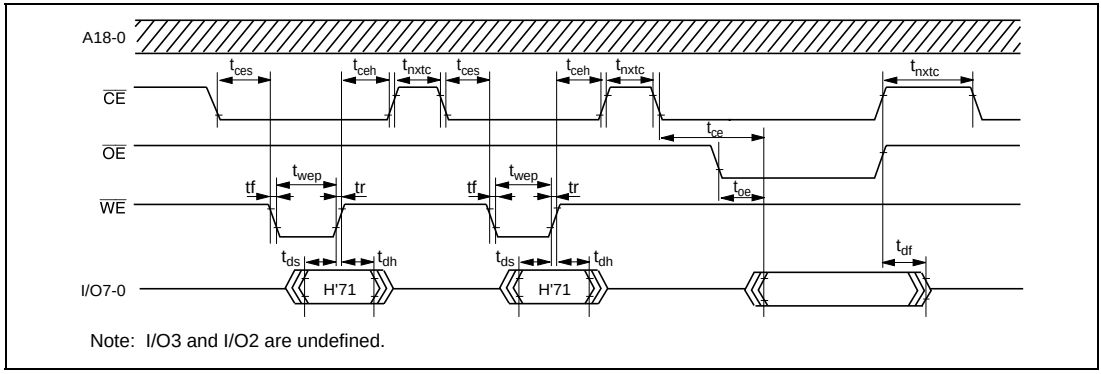


Figure 18.35 Timing in Status Read Mode

Table 18.26 Stipulated Transition Times to Command Wait State

Condition : V_{CC} = 3.0 V to 3.6 V, V_{SS} = 0 V, T_a = 25°C ± 5°C

Code	Symbol	Min	Max	Unit	Note
Standby release (oscillation settling time)	t_{osc1}	30		ms	
PROM mode setup time	t_{bmv}	10		ms	
V_{CC} hold time	t_{dwn}	0		ms	

20.4.4 Sample Application of Software Standby Mode

Figure 20.1 shows an example in which software standby mode is entered at the fall of NMI and exited at the rise of NMI.

With the NMI edge select bit (NMIEG) cleared to 0 in SYSCR (selecting the falling edge), an NMI interrupt occurs. Next the NMIEG bit is set to 1 (selecting the rising edge) and the SSBY bit is set to 1; then the SLEEP instruction is executed to enter software standby mode.

Software standby mode is exited at the next rising edge of the NMI signal.

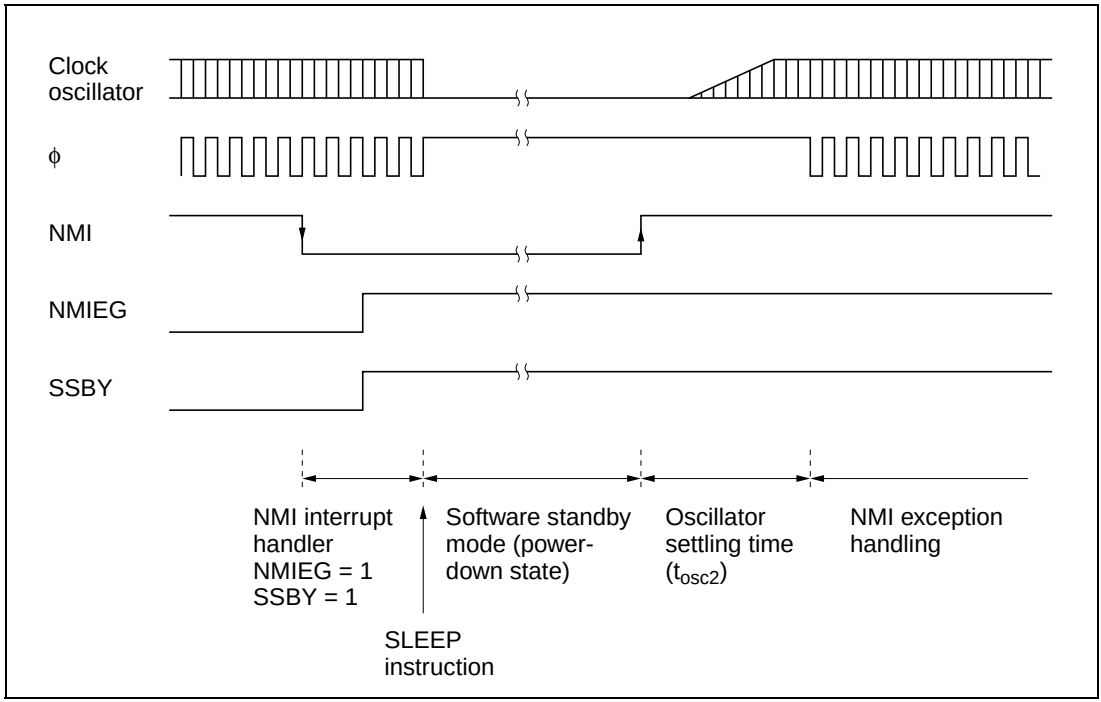


Figure 20.1 NMI Timing for Software Standby Mode (Example)

20.4.5 Note

The I/O ports retain their existing states in software standby mode. If a port is in the high output state, its output current is not reduced.

Address (Low)	Register Name	Data Bus Width	Bit Names								Module Name
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'EE080	—		—	—	—	—	—	—	—	—	
H'EE081	—		—	—	—	—	—	—	—	—	
H'EE082	—		—	—	—	—	—	—	—	—	
H'EE083	—		—	—	—	—	—	—	—	—	
H'EE084	—		—	—	—	—	—	—	—	—	
H'EE085	—		—	—	—	—	—	—	—	—	
H'EE086	—		—	—	—	—	—	—	—	—	
H'EE087	—		—	—	—	—	—	—	—	—	
H'EE088	—		—	—	—	—	—	—	—	—	
H'EE089	—		—	—	—	—	—	—	—	—	
H'EE08A	—		—	—	—	—	—	—	—	—	
H'EE08B	—		—	—	—	—	—	—	—	—	
H'EE08C	—		—	—	—	—	—	—	—	—	
H'EE08D	—		—	—	—	—	—	—	—	—	
H'EE08E	—		—	—	—	—	—	—	—	—	
H'EE08F	—		—	—	—	—	—	—	—	—	
H'EE090	—		—	—	—	—	—	—	—	—	
H'EE091	—		—	—	—	—	—	—	—	—	
H'EE092	—		—	—	—	—	—	—	—	—	
H'EE093	—		—	—	—	—	—	—	—	—	
H'EE094	—		—	—	—	—	—	—	—	—	
H'EE095	—		—	—	—	—	—	—	—	—	
H'EE096	—		—	—	—	—	—	—	—	—	
H'EE097	—		—	—	—	—	—	—	—	—	
H'EE098	—		—	—	—	—	—	—	—	—	
H'EE099	—		—	—	—	—	—	—	—	—	
H'EE09A	—		—	—	—	—	—	—	—	—	
H'EE09B	—		—	—	—	—	—	—	—	—	
H'EE09C	—		—	—	—	—	—	—	—	—	
H'EE09D	—		—	—	—	—	—	—	—	—	
H'EE09E	—		—	—	—	—	—	—	—	—	
H'EE09F	—		—	—	—	—	—	—	—	—	

B.2 Addresses (cont)

Address (Low)	Register Name	Data Bus Width	Bit Names								Module Name
			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'EE020	ABWCR	8	ABW7	ABW6	ABW5	ABW4	ABW3	ABW2	ABW1	ABW0	Bus controller
H'EE021	ASTCR	8	AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0	
H'EE022	WCRH	8	W71	W70	W61	W60	W51	W50	W41	W40	
H'EE023	WCRL	8	W31	W30	W21	W20	W11	W10	W01	W00	
H'EE024	BCR	8	ICIS1	ICIS0	BROME	BRSTS1	BRSTS0	—	RDEA	WAITE	
H'EE025	—		—	—	—	—	—	—	—	—	
H'EE026	DRCRA	8	DRAS2	DRAS1	DRAS0	—	BE	RDM	SRFMD	RFSHE	DRAM Interface
H'EE027	DRCRB	8	MXC1	MXC0	CSEL	RCYCE	—	TPC	RCW	RLW	
H'EE028	RTMCSR	8	CMF	CMIE	CKS2	CKS1	CKS0	—	—	—	
H'EE029	RTCNT	8									
H'EE02A	RTCOR	8									
H'EE02B	—		—	—	—	—	—	—	—	—	
H'EE02C	—		—	—	—	—	—	—	—	—	
H'EE02D	—		—	—	—	—	—	—	—	—	
H'EE02E	—		—	—	—	—	—	—	—	—	
H'EE02F	—		—	—	—	—	—	—	—	—	
H'EE030	—		—	—	—	—	—	—	—	—	
H'EE031	—		—	—	—	—	—	—	—	—	
H'EE032	—		—	—	—	—	—	—	—	—	
H'EE033	—		—	—							
H'EE034	—		—	—	—	—	—	—	—	—	
H'EE035	—		—	—	—	—	—	—	—	—	
H'EE036	—		—	—	—	—	—	—	—	—	
H'EE037	—		—	—	—	—	—	—	—	—	
H'EE038	Reserved area (access prohibited)										
H'EE039											
H'EE03A											
H'EE03B											
H'EE03C	P2PCR	8	$P2_7PCR$	$P2_6PCR$	$P2_5PCR$	$P2_4PCR$	$P2_3PCR$	$P2_2PCR$	$P2_1PCR$	$P2_0PCR$	Port 2
H'EE03D	—		—	—	—	—	—	—	—	—	
H'EE03E	P4PCR	8	$P4_7PCR$	$P4_6PCR$	$P4_5PCR$	$P4_4PCR$	$P4_3PCR$	$P4_2PCR$	$P4_1PCR$	$P4_0PCR$	Port 4
H'EE03F	P5PCR	8	—	—	—	—	$P5_3PCR$	$P5_2PCR$	$P5_1PCR$	$P5_0PCR$	Port 5

IER—IRQ Enable Register	H'EE015	Interrupt Controller

Bit	7	6	5	4	3	2	1	0
	———	———	IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IRQ_5 to IRQ_0 enable

0	IRQ_5 to IRQ_0 interrupts are disabled
1	IRQ_5 to IRQ_0 interrupts are enabled

ISR—IRQ Status Register	H'EE016	Interrupt Controller

Bit	7	6	5	4	3	2	1	0
	——	——	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
Initial value	0	0	0	0	0	0	0	0
Read/Write	——	——	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

IRQ5 to IRQ0 flags

Bits 5 to 0 IRQ5F to IRQ0F	Setting and Clearing Conditions
0	[Clearing conditions] • Read IRQnF when IRQnF = 1, then write 0 in IRQnF. • IRQnSC = 0, $\overline{\text{IRQn}}$ input is high, and interrupt exception handling is being carried out. • IRQnSC = 1 and IRQn interrupt exception handling is being carried out.
1	[Setting conditions] • IRQnSC = 0 and $\overline{\text{IRQn}}$ input is low. • IRQnSC = 1 and $\overline{\text{IRQn}}$ input changes from high to low.

(n = 5 to 0)

Note: * Only 0 can be written, to clear the flag.

RDR—Receive Data Register	H'FFFB5	SCI0