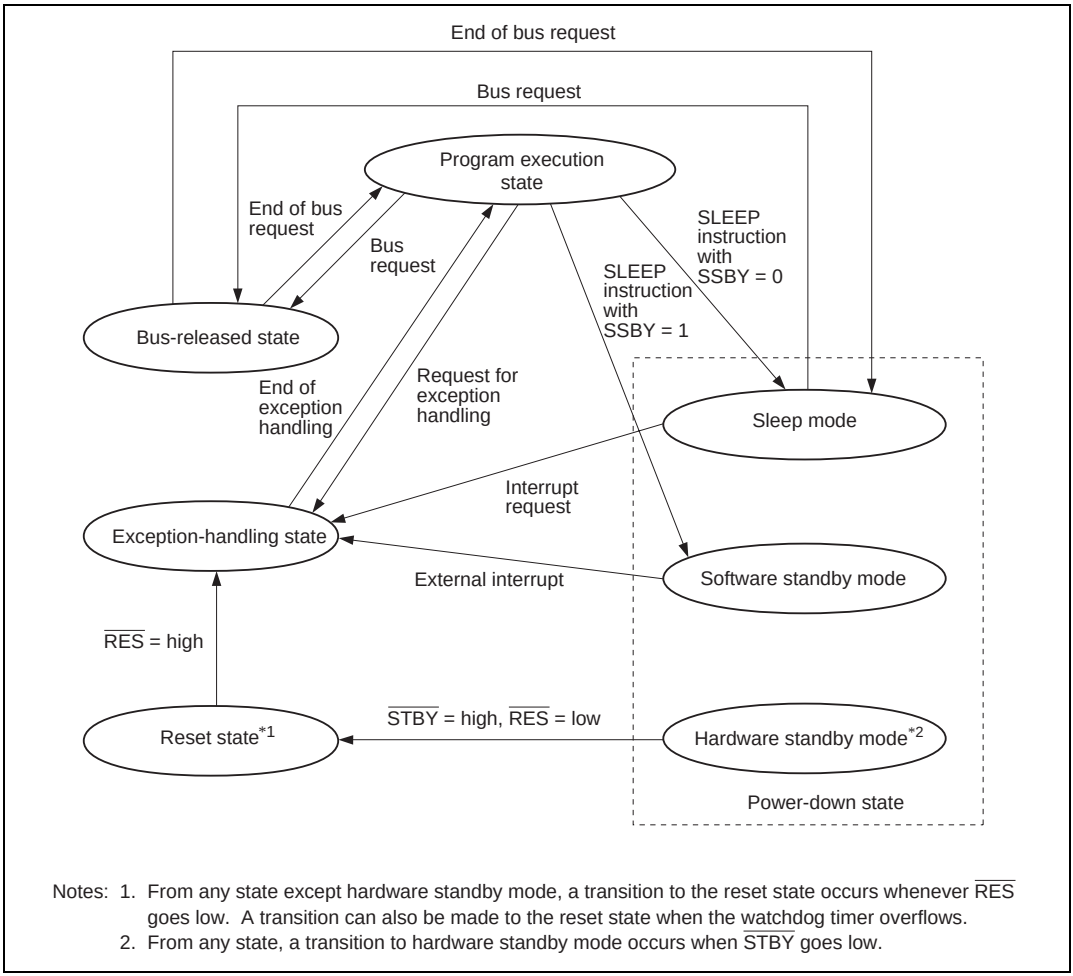
Welcome to E-XFL.COM

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	16-Bit
Speed	25MHz
Connectivity	SCI, SmartCard
Peripherals	DMA, POR, PWM, WDT
Number of I/O	86
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 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	120-TQFP
Supplier Device Package	120-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df2328bvte25v



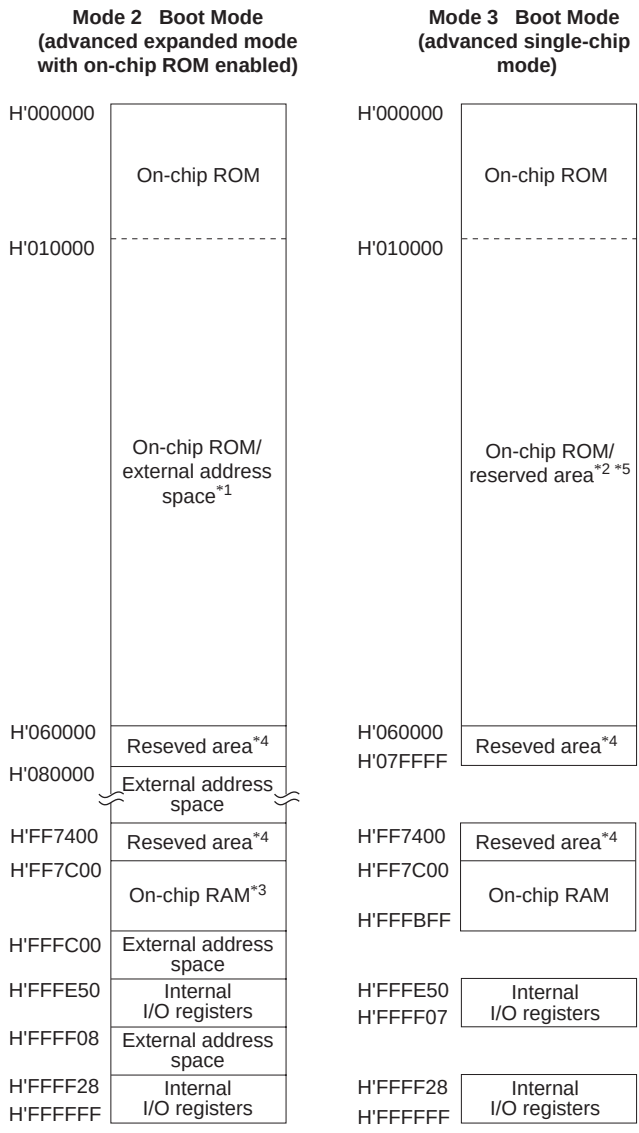
Notes: 1. From any state except hardware standby mode, a transition to the reset state occurs whenever $\overline{RES}$ goes low. A transition can also be made to the reset state when the watchdog timer overflows.
2. From any state, a transition to hardware standby mode occurs when $\overline{STBY}$ goes low.

Figure 2.12 State Transitions

2.8.2 Reset State

When the $\overline{RES}$ input goes low all current processing stops and the CPU enters the reset state. All interrupts are masked in the reset state. Reset exception handling starts when the $\overline{RES}$ signal changes from low to high.

The reset state can also be entered by a watchdog timer overflow. For details, refer to section 13, Watchdog Timer.



Notes: 1. External addresses when EAE = 1 in BCRL; on-chip ROM when EAE = 0.
2. Reserved area when EAE = 1 in BCRL; on-chip ROM when EAE = 0.
3. On-chip RAM is used for flash memory programming. Do not clear the RAME bit in SYSCR to 0.
4. Access to the reserved areas H'060000 to H'07FFFF and H'FF7400 to H'FF7BFF is prohibited.
5. Do not access a reserved area.

Figure 3.1 (a) H8S/2329B Memory Map in Each Operating Mode

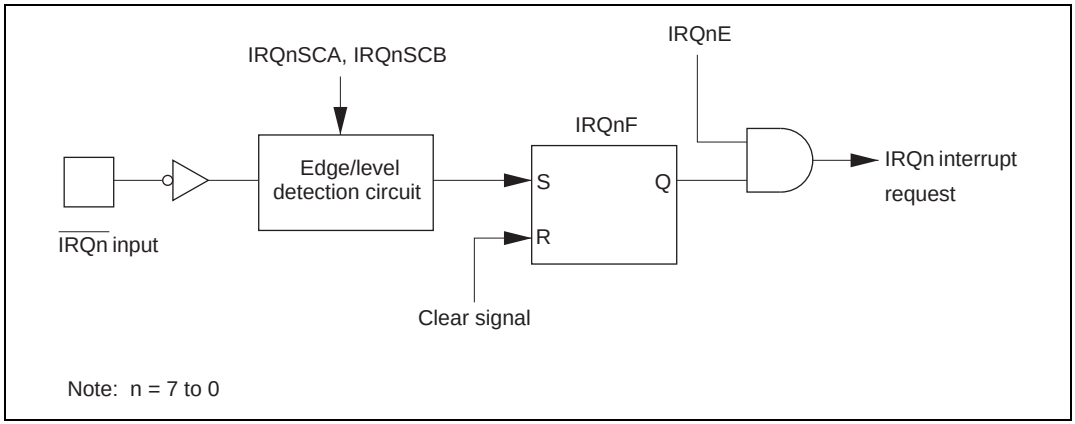


Figure 5.2 Block Diagram of Interrupts IRQ7 to IRQ0

Figure 5.3 shows the timing of setting IRQnF.

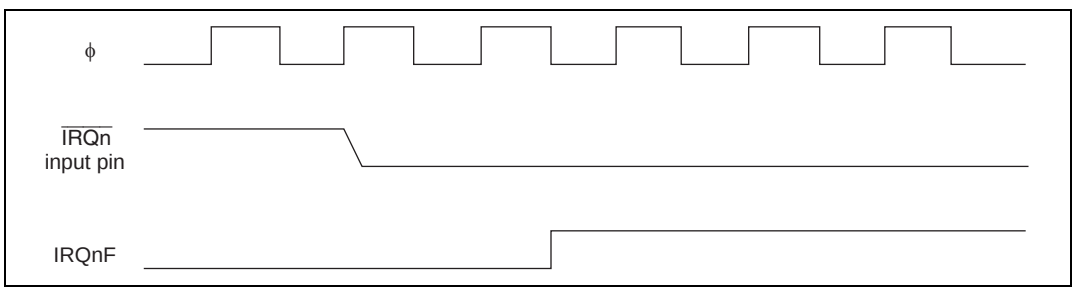


Figure 5.3 Timing of Setting IRQnF

The vector numbers for IRQ7 to IRQ0 interrupt exception handling are 23 to 16.

Detection of IRQ7 to IRQ0 interrupts does not depend on whether the relevant pin has been set for input or output. Therefore, when a pin is used as an external interrupt input pin, do not clear the corresponding DDR bit to 0 and use the pin as an I/O pin for another function. The pins that can be used for IRQ4 to IRQ7 interrupt input can be switched by means of the IRQPAS bit in SYSCR.

- Idle cycle insertion
  - An idle cycle can be inserted in case of an external read cycle between different areas
  - An idle cycle can be inserted when an external read cycle is immediately followed by an external write cycle
- Write buffer functions
  - External write cycle and internal access can be executed in parallel
  - DMAC* single address mode and internal access can be executed in parallel
- Bus arbitration function
  - Includes a bus arbiter that arbitrates bus mastership among the CPU, DMAC, and DTC
- Other features
  - Refresh counter (refresh timer)* can be used as an interval timer
  - External bus release function

Note: * The DRAM interface, DMAC, and refresh counter are not supported in the H8S/2321.

6.5.6 Basic Timing

Figure 6.15 shows the basic access timing for DRAM space. The basic DRAM access timing is 4 states. Unlike the basic bus interface, the corresponding bits in ASTCR control only enabling or disabling of wait insertion, and do not affect the number of access states. When the corresponding bit in ASTCR is cleared to 0, wait states cannot be inserted in the DRAM access cycle.

The 4 states of the basic timing consist of one T_p (precharge cycle) state, one T_r (row address output cycle), and two T_c (column address output cycle) states, T_{c1} and T_{c2}.

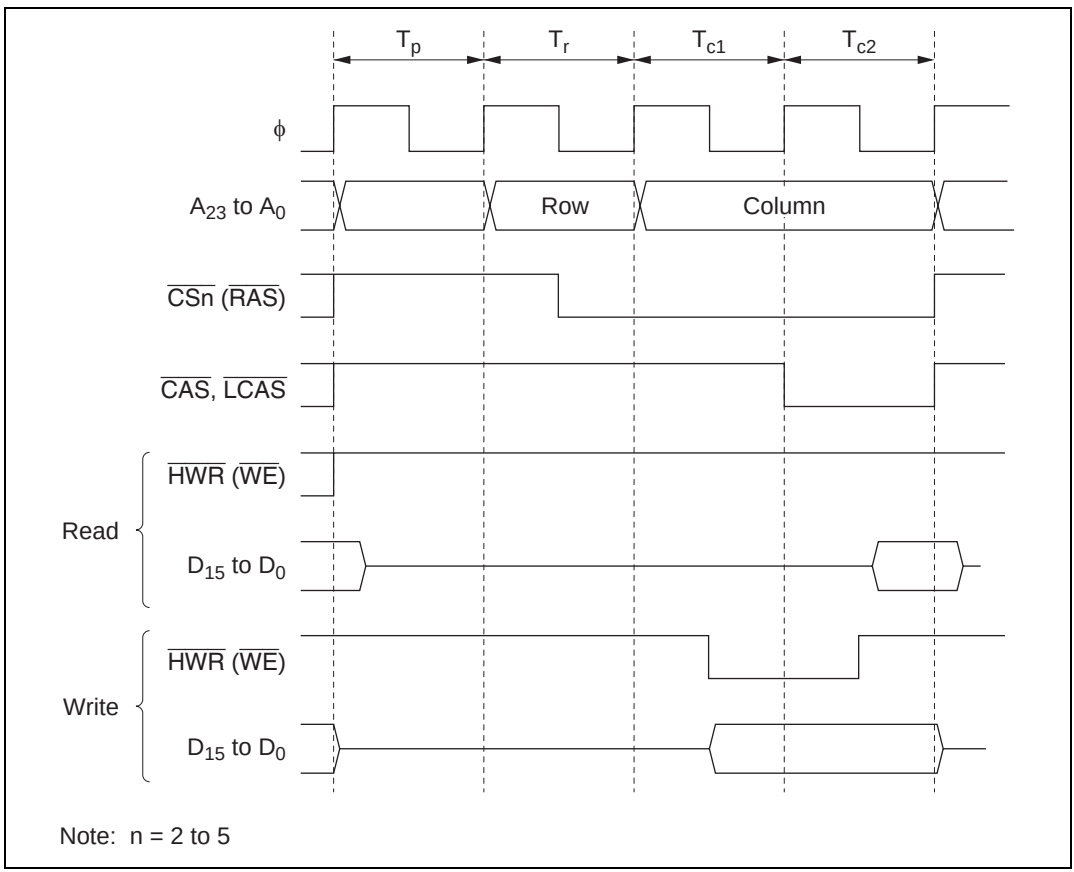


Figure 6.15 Basic Access Timing

6.5.11 Refresh Control

The chip is provided with a DRAM refresh control function. Either of two refreshing methods can be selected: CAS-before-RAS (CBR) refreshing, or self-refreshing.

CAS-before-RAS (CBR) Refreshing: To select CBR refreshing, set the RFSHE bit in DRAMCR to 1, and clear the RMODE bit to 0.

With CBR refreshing, RTCNT counts up using the input clock selected by bits CKS2 to CKS0 in DRAMCR, and when the count matches the value set in RTCOR (compare match), refresh control is performed. At the same time, RTCNT is reset and starts counting again from H'00. Refreshing is thus repeated at fixed intervals determined by RTCOR and bits CKS2 to CKS0. Set a value in RTCOR and bits CKS2 to CKS0 that will meet the refreshing interval specification for the DRAM used.

When bits CKS2 to CKS0 are set, RTCNT starts counting up. RTCNT and RTCOR settings should therefore be completed before setting bits CKS2 to CKS0.

Do not clear the CMF flag when refresh control is being performed (RFSHE = 1).

RTCNT operation is shown in figure 6.23, compare match timing in figure 6.24, and CBR refresh timing in figure 6.25.

Access to other normal space can be performed during the CBR refresh interval.

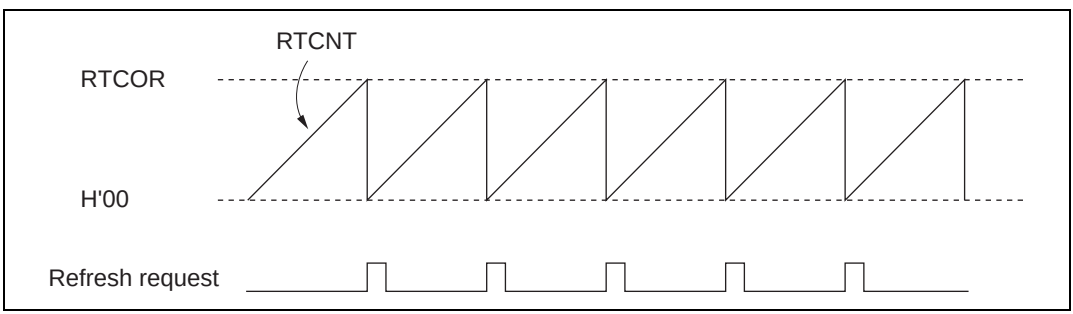
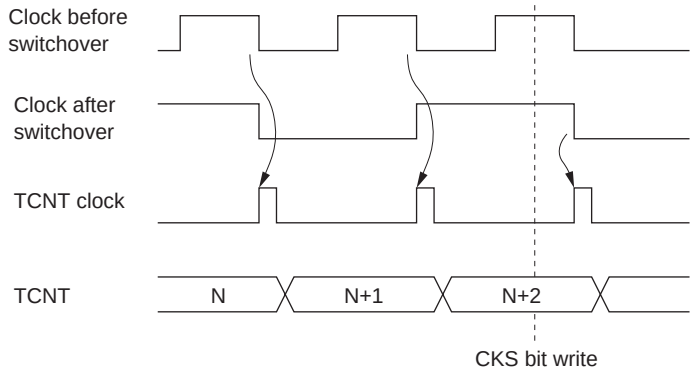


Figure 6.23 RTCNT Operation

No.	Timing of Switchover by Means of CKS1 and CKS0 Bits	TCNT Clock Operation
4	Switching from high to high	

Notes: 1. Includes switching from low to stop, and from stop to low.
2. Includes switching from stop to high.
3. Includes switching from high to stop.
4. Generated on the assumption that the switchover is a falling edge; TCNT is incremented.

12.6.6 Interrupts and Module Stop Mode

If module stop mode is entered when an interrupt has been requested, it will not be possible to clear the CPU interrupt source or the DMAC* or DTC activation source. Interrupts should therefore be disabled before entering module stop mode.

Note: * The DMAC is not supported in the H8S/2321.

Section 13 Watchdog Timer

13.1 Overview

The chip has a single-channel on-chip watchdog timer (WDT) for monitoring system operation. The WDT outputs an overflow signal ($\overline{\text{WDTOVF}}$)* if a system crash prevents the CPU from writing to the timer counter, allowing it to overflow. At the same time, the WDT can also generate an internal reset signal for the chip.

When this watchdog function is not needed, the WDT can be used as an interval timer. In interval timer operation, an interval timer interrupt is generated each time the counter overflows.

13.1.1 Features

WDT features are listed below.

- Switchable between watchdog timer mode and interval timer mode
- $\overline{\text{WDTOVF}}$ output when in watchdog timer mode*
  If the counter overflows, the WDT outputs $\overline{\text{WDTOVF}}$*. It is possible to select whether or not the entire chip is reset at the same time
- Interrupt generation when in interval timer mode
  If the counter overflows, the WDT generates an interval timer interrupt
- Choice of eight counter clock sources

Note: * The $\overline{\text{WDTOVF}}$ pin function cannot be used in the F-ZTAT versions.

14.3.3 Multiprocessor Communication Function

The multiprocessor communication function performs serial communication using the multiprocessor format, in which a multiprocessor bit is added to the transfer data, in asynchronous mode. Use of this function enables data transfer to be performed among a number of processors sharing a single serial communication line.

When multiprocessor communication is carried out, each receiving station is addressed by a unique ID code.

The serial communication cycle consists of two component cycles: an ID transmission cycle which specifies the receiving station, and a data transmission cycle. The multiprocessor bit is used to differentiate between the ID transmission cycle and the data transmission cycle.

The transmitting station first sends the ID of the receiving station with which it wants to perform serial communication as data with a 1 multiprocessor bit added. It then sends transmit data as data with a 0 multiprocessor bit added.

The receiving station skips the data until data with a 1 multiprocessor bit is sent.

When data with a 1 multiprocessor bit is received, the receiving station compares that data with its own ID. The station whose ID matches then receives the data sent next. Stations whose ID does not match continue to skip the data until data with a 1 multiprocessor bit is again received. In this way, data communication is carried out among a number of processors.

Figure 14.9 shows an example of inter-processor communication using the multiprocessor format.

Data Transfer Formats

There are four data transfer formats.

When the multiprocessor format is specified, the parity bit specification is invalid.

For details, see table 14.10.

Retransfer Operations (Except Block Transfer Mode): Retransfer operations are performed by the SCI in receive mode and transmit mode as described below.

- Retransfer operation when SCI is in receive mode
  Figure 15.11 illustrates the retransfer operation when the SCI is in receive mode.

[1] If an error is found when the received parity bit is checked, the PER bit in SSR is automatically set to 1. If the RIE bit in SCR is enabled at this time, an ERI interrupt request is generated. The PER bit in SSR should be kept cleared to 0 until the next parity bit is sampled.

[2] The RDRF bit in SSR is not set for a frame in which an error has occurred.

[3] If no error is found when the received parity bit is checked, the PER bit in SSR is not set.

[4] If no error is found when the received parity bit is checked, the receive operation is judged to have been completed normally, and the RDRF flag in SSR is automatically set to 1. If the RIE bit in SCR is enabled at this time, an RXI interrupt request is generated.
  If DMAC* or DTC data transfer by an RXI source is enabled, the contents of RDR can be read automatically. When the RDR data is read by the DMAC* or DTC, the RDRF flag is automatically cleared to 0.

[5] When a normal frame is received, the pin retains the high-impedance state at the timing for error signal transmission.

Note: * The DMAC is not supported in the H8S/2321.

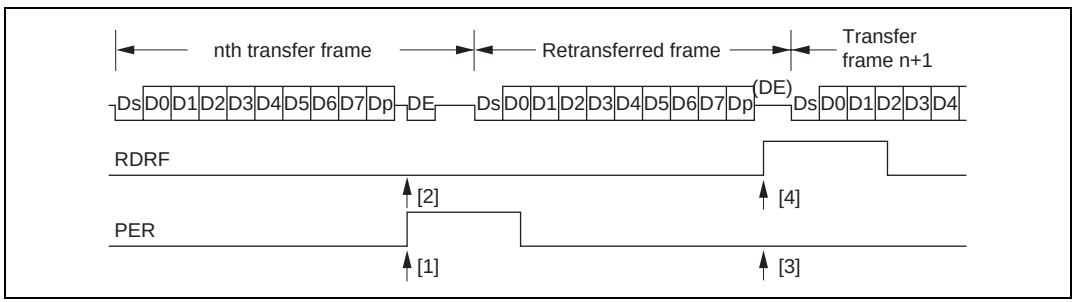


Figure 15.11 Retransfer Operation in SCI Receive Mode

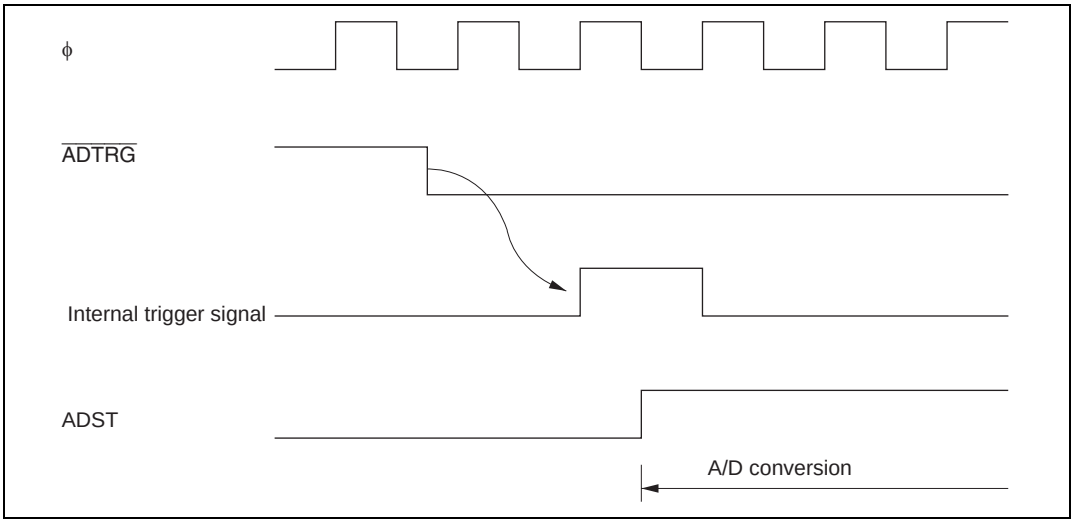


Figure 16.6 External Trigger Input Timing

16.5 Interrupts

The A/D converter generates an A/D conversion end interrupt (ADI) at the end of A/D conversion. ADI interrupt requests can be enabled or disabled by means of the ADIE bit in ADCSR.

The DTC or DMAC* can be activated by an ADI interrupt. Having the converted data read by the DTC or DMAC* in response to an ADI interrupt enables continuous conversion to be achieved without imposing a load on software.

The A/D converter interrupt source is shown in table 16.6.

Note: * The DMAC is not supported in the H8S/2321.

Table 16.6 A/D Converter Interrupt Source

Interrupt Source	Description	DTC Activation	DMAC Activation*
ADI	Interrupt due to end of conversion	Possible	Possible

Note: * The DMAC is not supported in the H8S/2321.

19.11.5 Auto-Program Mode

- In auto-program mode, 128 bytes are programmed simultaneously. For this purpose, 128 consecutive byte data transfers should be performed.
- A 128-byte data transfer must be performed even if writing fewer than 128 bytes; in this case, H'FF data must be written to the extra addresses.
- The lower 7 bits of the transfer address must be held low. If an invalid address is input, memory programming will be started but a programming error will occur.
- Memory address transfer is executed in the second cycle (figure 19.25). Do not perform transfer later than the second cycle.
- Do not perform a command write during a programming operation.
- Perform one auto-programming operation for a 128-byte block for each address. One or more additional programming operations cannot be carried out on address blocks that have already been programmed.
- Confirm normal end of auto-programming by checking I/O_6. Alternatively, status read mode can also be used for this purpose (the I/O_7 status polling pin is used to identify the end of an auto-program operation).
- Status polling I/O_6 and I/O_7 information is retained until the next command write. As long as the next command write has not been performed, reading is possible by enabling $\overline{CE}$ and $\overline{OE}$.

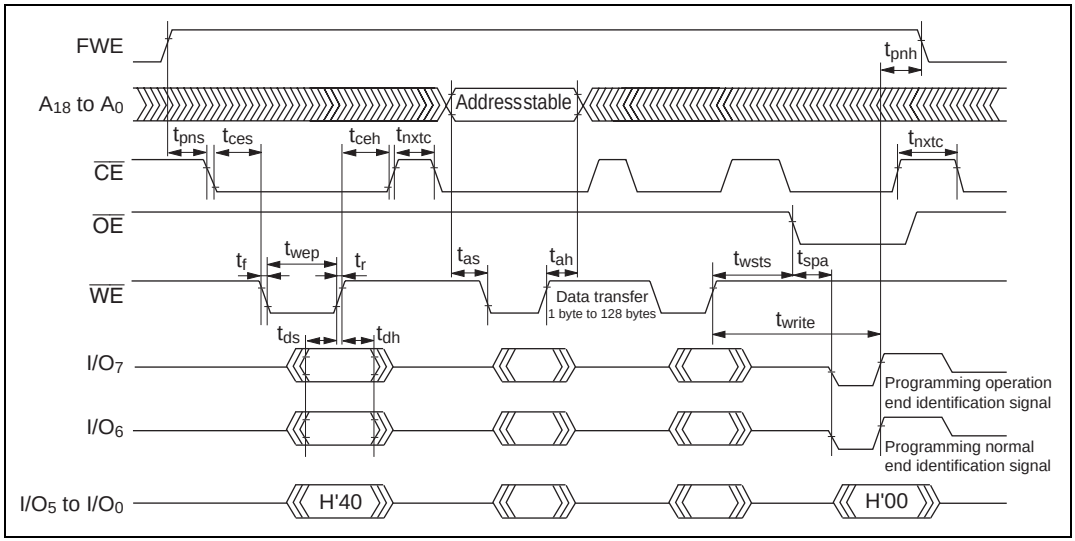


Figure 19.82 Auto-Program Mode Timing Waveforms

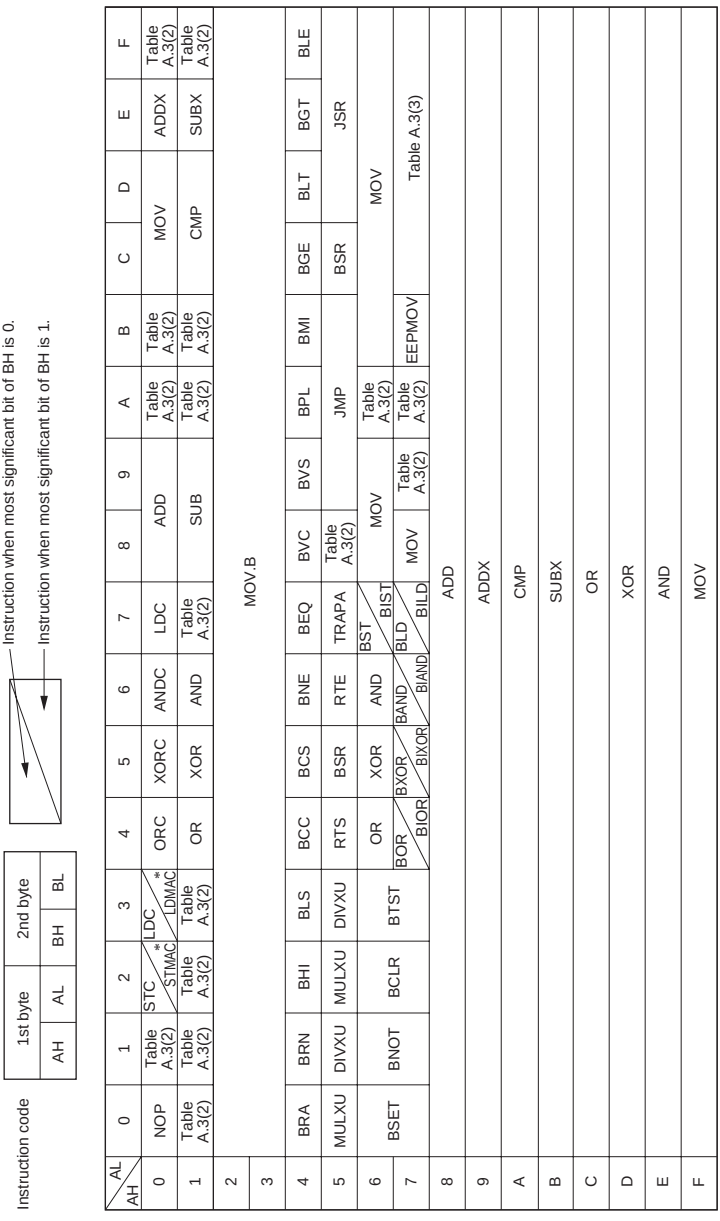
19.29.6 Auto-Erase Mode

- Auto-erase mode supports only total memory erasing.
- Do not perform a command write during auto-erasing.
- Confirm normal end of auto-erasing by checking I/O_6. Alternatively, status read mode can also be used for this purpose (the I/O_7 status polling pin is used to identify the end of an auto-erase operation).
- Status polling I/O_6 and I/O_7 pin information is retained until the next command write. As long as the next command write has not been performed, reading is possible by enabling $\overline{CE}$ and $\overline{OE}$.

A.3 Operation Code Map

Table A.3 shows the operation code map.

Table A.3 Operation Code Map (1)



Note: * Cannot be used in the chip.

Module	Register	Abbreviation	R/W	Initial Value	Address[*1]
PPG	PPG output control register	PCR	R/W	H'FF	H'FF46
	PPG output mode register	PMR	R/W	H'F0	H'FF47
	Next data enable register H	NDERH	R/W	H'00	H'FF48
	Next data enable register L	NDERL	R/W	H'00	H'FF49
	Output data register H	PODRH	R/(W)[*8]	H'00	H'FF4A
	Output data register L	PODRL	R/(W)[*8]	H'00	H'FF4B
	Next data register H	NDRH	R/W	H'00	H'FF4C[*9] H'FF4E
	Next data register L	NDRL	R/W	H'00	H'FF4D[*9] H'FF4F
	Port 1 data direction register	P1DDR	W	H'00	H'FEB0
	Port 2 data direction register	P2DDR	W	H'00	H'FEB1
	Module stop control register	MSTPCR	R/W	H'3FFF	H'FF3C
8-bit timer 0	Timer control register 0	TCR0	R/W	H'00	H'FFB0
	Timer control/status register 0	TCSR0	R/(W)[*10]	H'00	H'FFB2
	Timer constant register A0	TCORA0	R/W	H'FF	H'FFB4
	Timer constant register B0	TCORB0	R/W	H'FF	H'FFB6
	Timer counter 0	TCNT0	R/W	H'00	H'FFB8
8-bit timer 1	Timer control register 1	TCR1	R/W	H'00	H'FFB1
	Timer control/status register 1	TCSR1	R/(W)[*10]	H'10	H'FFB3
	Timer constant register A1	TCORA1	R/W	H'FF	H'FFB5
	Timer constant register B1	TCORB1	R/W	H'FF	H'FFB7
	Timer counter 1	TCNT1	R/W	H'00	H'FFB9
Both 8-bit timer channels	Module stop control register	MSTPCR	R/W	H'3FFF	H'FF3C

TIOR5—Timer I/O Control Register 5 H'FEA2 TPU5

Bit	:	7	6	5	4	3	2	1	0
		IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0
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

TGR5A I/O Control

0	0	0	0	TGR5A is output compare register	Output disabled	
			1		Initial output is 0 output	0 output at compare match
		1	0			1 output at compare match
			1			Toggle output at compare match
	1	0	0		Output disabled	
			1		Initial output is 1 output	0 output at compare match
		1	0			1 output at compare match
			1			Toggle output at compare match
1	*	0	0	TGR5A is input capture register	Capture input source is $TIOCA_5$ pin	Input capture at rising edge
			1			Input capture at falling edge
		1	*			Input capture at both edges

* : Don't care

TGR5B I/O Control

0	0	0	0	TGR5B is output compare register	Output disabled	
			1		Initial output is 0 output	0 output at compare match
		1	0			1 output at compare match
			1			Toggle output at compare match
	1	0	0		Output disabled	
			1		Initial output is 1 output	0 output at compare match
		1	0			1 output at compare match
			1			Toggle output at compare match
1	*	0	0	TGR5B is input capture register	Capture input source is $TIOCB_5$ pin	Input capture at rising edge
			1			Input capture at falling edge
		1	*			Input capture at both edges

* : Don't care

PORTG—Port G Register **H'FF5F** **Port G**

Bit	7	6	5	4	3	2	1	0
	—	—	—	PG4	PG3	PG2	PG1	PG0
Initial value	Undefined	Undefined	Undefined	—*	—*	—*	—*	—*
Read/Write	—	—	—	R	R	R	R	R

State of port G pins

Note: * Determined by the state of pins PG_4 to PG_0.

P1DR—Port 1 Data Register **H'FF60** **Port 1**

Bit	7	6	5	4	3	2	1	0
	P17DR	P16DR	P15DR	P14DR	P13DR	P12DR	P11DR	P10DR
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

Stores output data for port 1 pins ($P1_7$ to $P1_0$)

P2DR—Port 2 Data Register **H'FF61** **Port 2**

Bit	7	6	5	4	3	2	1	0
	P27DR	P26DR	P25DR	P24DR	P23DR	P22DR	P21DR	P20DR
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

Stores output data for port 2 pins ($P2_7$ to $P2_0$)

PDDR—Port D Data Register **H'FF6C** **Port D**

Bit	7	6	5	4	3	2	1	0
	PD7DR	PD6DR	PD5DR	PD4DR	PD3DR	PD2DR	PD1DR	PD0DR
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

Stores output data for port D pins (PD_7 to PD_0)

PEDR—Port E Data Register **H'FF6D** **Port E**

Bit	7	6	5	4	3	2	1	0
	PE7DR	PE6DR	PE5DR	PE4DR	PE3DR	PE2DR	PE1DR	PE0DR
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

Stores output data for port E pins (PE_7 to PE_0)

PFDR—Port F Data Register **H'FF6E** **Port F**

Bit	7	6	5	4	3	2	1	0
	PF7DR	PF6DR	PF5DR	PF4DR	PF3DR	PF2DR	PF1DR	PF0DR
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

Stores output data for port F pins (PF_7 to PF_0)

SSR2—Serial Status Register 2 H'FF8C Smart Card Interface 2

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	Transmission in progress [Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When the DMAC*1 or DTC is activated by a TXI interrupt and writes data to TDR
1	Transmission has ended [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 transmission of a 1-byte serial character when GM = 0 and BLK = 0 • When TDRE = 1 and ERS = 0 (normal transmission) 1.5 etu after transmission of a 1-byte serial character when GM = 0 and BLK = 1 • When TDRE = 1 and ERS = 0 (normal transmission) 1.0 etu after transmission of a 1-byte serial character when GM = 1 and BLK = 0 • When TDRE = 1 and ERS = 0 (normal transmission) 1.0 etu after transmission of a 1-byte serial character when GM = 1 and BLK = 1

Notes: etu: Elementary time unit (time for transfer of 1 bit)
1. The DMAC is not supported in the H8S/2321.

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	Data has been received normally, and there is no error signal [Clearing conditions] • On reset, or in standby mode or module stop mode • When 0 is written to ERS after reading ERS =1
1	Error signal indicating detection of parity error has been sent by receiving device [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*1 or DTC is activated by an RXI interrupt and reads data from RDR
1	[Setting condition] When serial reception ends normally and receive data is transferred from RSR to RDR

Note: 1. The DMAC is not supported in the H8S/2321.

Transmit Data Register Empty

0	[Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When the DMAC*1 or DTC is activated by a TXI interrupt and writes 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: 1. The DMAC is not supported in the H8S/2321.

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

C.2 Port 2

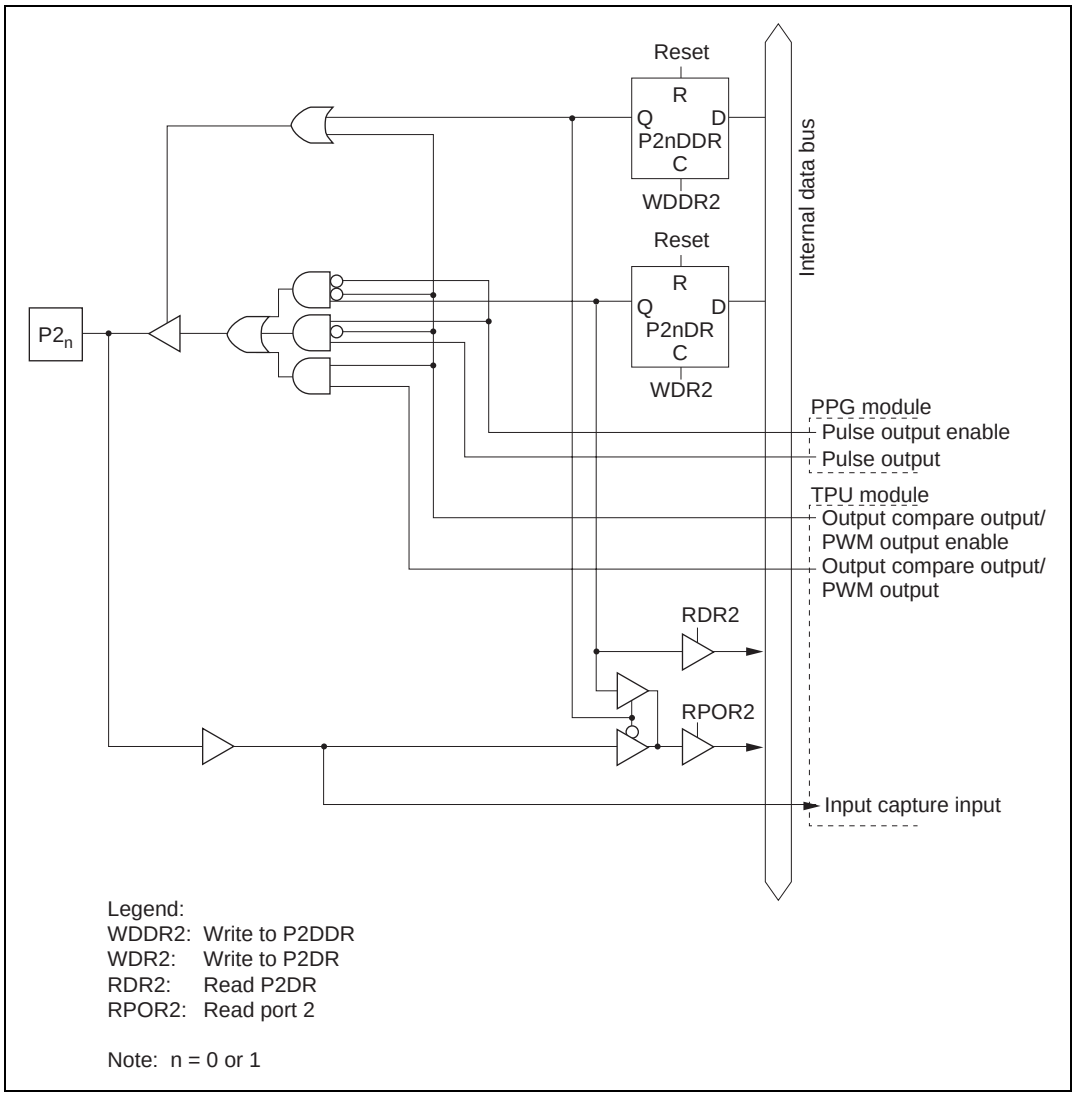


Figure C.2 (a) Port 2 Block Diagram (Pins $P2_0$ and $P2_1$)