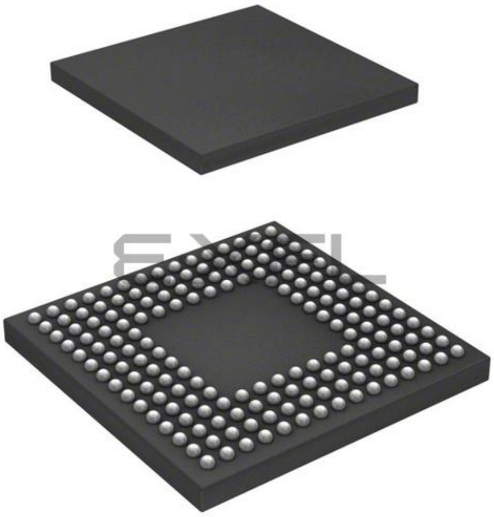
Renesas Electronics America Inc - DF2505BR26DV Datasheet



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	H8S/2000
Core Size	16-Bit
Speed	26MHz
Connectivity	I²C, SCI
Peripherals	POR, PWM, WDT
Number of I/O	104
Program Memory Size	384KB (384K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 16x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	176-LFBGA
Supplier Device Package	176-LFBGA (13x13)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df2505br26dv

		Pin No.			
Type	Symbol	FP-144J, FP-144JV	BP-176V*2	I/O	Function
Bus control	$\overline{\text{WAIT}}$	79	L12	I	Requests insertion of wait cycles in a bus cycle when the access is made to the external address space.
16-bit timer-pulse unit (TPU)	TCLKD	109	A15	I	These pins input an external clock.
	TCLKC	111	B13		
	TCLKB	113	B12		
	TCLKA	114	D11		
	TIOCA0	116	B11	I/O	Pins for the TGRA_0 to TGRD_0 input capture input, output compare output, or PWM output.
	TIOCB0	115	C11		
	TIOCC0	114	D11		
	TIOCD0	113	B12		
	TIOCA1	112	C12	I/O	Pins for the TGRA_1 and TGRB_1 input capture input, output compare output, or PWM output.
	TIOCB1	111	B13		
	TIOCA2	110	C13	I/O	Pins for the TGRA_2 and TGRB_2 input capture input, output compare output, or PWM output.
	TIOCB2	109	A15		
	TIOCA3	101	D15	I/O	Pins for the TGRA_3 and TGRD_3 input capture input, output compare output, or PWM output.
	TIOCB3	102	D14		
	TIOCC3	103	D13		
	TIOCD3	104	C15		
	TIOCA4	105	D12	I/O	Pins for the TGRA_4 and TGRB_4 input capture input, output compare output, or PWM output.
	TIOCB4	106	C14		
	TIOCA5	107	B15	I/O	Pins for the TGRA_5 and TGRB_5 input capture input, output compare output, or PWM output.
	TIOCB5	108	B14		
8-bit timer	TMO3	129	D6, C6, A6, B6	O	Compare-match output pins
	TMO2	130			
	TMO1	131			
	TMO0	132			
	TMCI23	133	C5	I	Pins for external clock input to the counter
	TMCI01	134	A5		
	TMRI23	133	C5	I	Counter reset input pins.
	TMRI01	134	A5		

Symbol	Description
→	Move
~	NOT (logical complement)
:8/:16/:24/:32	8-, 16-, 24-, or 32-bit length

Note: * General registers include 8-bit registers (R0H to R7H, R0L to R7L), 16-bit registers (R0 to R7, E0 to E7), and 32-bit registers (ER0 to ER7).

Table 2.3 Data Transfer Instructions

Instruction	Size*[1]	Function
MOV	B/W/L	(EAs) → Rd, Rs → (EAd) Moves data between two general registers or between a general register and memory, or moves immediate data to a general register.
MOVFPE	B	Cannot be used in this LSI.
MOVTPE	B	Cannot be used in this LSI.
POP	W/L	@SP+ → Rn Pops a general register from the stack. POP.W Rn is identical to MOV.W @SP+, Rn. POP.L ERn is identical to MOV.L @SP+, ERn.
PUSH	W/L	Rn → @–SP Pushes a general register onto the stack. PUSH.W Rn is identical to MOV.W Rn, @–SP. PUSH.L ERn is identical to MOV.L ERn, @–SP.
LDM*[2]	L	@SP+ → Rn (register list) Pops two or more general registers from the stack.
STM*[2]	L	Rn (register list) → @–SP Pushes two or more general registers onto the stack.

Notes: 1. Refers to the operand size.

B: Byte

W: Word

L: Longword

2. Only register ER0 to ER6 should be used when using the STM/LDM instruction.

Bit	Bit Name	Initial Value	R/W	Description
0	Sz	Undefined	—	DTC Data Transfer Size Specifies the size of data to be transferred. 0: Byte-size transfer 1: Word-size transfer

Legend:

X: Don't care

8.2.2 DTC Mode Register B (MRB)

MRB is an 8-bit register that selects the DTC operating mode.

Bit	Bit Name	Initial Value	R/W	Description
7	CHNE	Undefined	—	DTC Chain Transfer Enable This bit specifies a chain transfer. For details, see section 8.5.4, Chain Transfer. In data transfer with CHNE set to 1, determination of the end of the specified number of transfers, clearing of the interrupt source flag, and clearing of DTCER, are not performed. 0: DTC data transfer completed (waiting for start) 1: DTC data transfer (reads new register information and transfers data)
6	DISEL	Undefined	—	DTC Interrupt Select This bit specifies whether CPU interrupt is disabled or enabled after a data transfer. 0: Interrupt request is issued to the CPU when the specified data transfer is completed. (The DTC clears the interrupt request flag that causes the activation.) 1: The DTC issues interrupt request to the CPU in every data transfer. (The DTC does not clear the interrupt request flag that causes the activation.)
5 to 0	—	Undefined	—	Reserved These bits have no effect on the DTC operation. The write value should always be 0.

9.10 Port C

Port C is an 8-bit I/O port and has the following registers.

- Port C data direction register (PCDDR)
- Port C data register (PCDR)
- Port C register (PORTC)
- Port C pull-up MOS control register (PCPCR)

9.10.1 Port C Data Direction Register (PCDDR)

PCDDR specifies input or output the port C pins using the individual bits. PCDDR cannot be read; if it is, the read value is undefined. Since this register is a write-only register, bit-manipulation instructions should not be used when writing. See section 2.9.4, Access Method for Registers with Write-only Bits.

Bit	Bit Name	Initial Value	R/W	Description
7	PC7DDR	0	W	When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port C pin an output pin. Clearing this bit to 0 makes the pin an input pin.
6	PC6DDR	0	W	
5	PC5DDR	0	W	
4	PC4DDR	0	W	
3	PC3DDR	0	W	
2	PC2DDR	0	W	
1	PC1DDR	0	W	
0	PC0DDR	0	W	

9.11 Port D

Port D is an 8-bit I/O port and has the following registers.

- Port D data direction register (PDDDR)
- Port D data register (PDDR)
- Port D register (PORTD)
- Port D pull-up MOS control register (PDPCR)

9.11.1 Port D Data Direction Register (PDDDR)

PDDDR specifies input or output the port D pins using the individual bits. PDDDR cannot be read; if it is, the read value is undefined. Since this register is a write-only register, bit-manipulation instructions should not be used when writing. See section 2.9.4, Access Method for Registers with Write-only Bits.

Bit	Bit Name	Initial Value	R/W	Description
7	PD7DDR	0	W	When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port D pin an output pin. Clearing this bit to 0 makes the pin an input pin.
6	PD6DDR	0	W	
5	PD5DDR	0	W	
4	PD4DDR	0	W	
3	PD3DDR	0	W	
2	PD2DDR	0	W	
1	PD1DDR	0	W	
0	PD0DDR	0	W	

9.12.2 Port E Data Register (PEDR)

PEDR stores output data for port E pins.

Bit	Bit Name	Initial Value	R/W	Description
7	PE7DR	0	R/W	Output data for a pin is stored when the pin is specified as a general purpose output port.
6	PE6DR	0	R/W	
5	PE5DR	0	R/W	
4	PE4DR	0	R/W	
3	PE3DR	0	R/W	
2	PE2DR	0	R/W	
1	PE1DR	0	R/W	
0	PE0DR	0	R/W	

9.12.3 Port E Register (PORTE)

PORTE shows port E pin states. This register cannot be modified.

Bit	Bit Name	Initial Value	R/W	Description
7	PE7	—*	R	If a port E read is performed while PEDDR bits are set to 1, the PEDR values are read. If a port E read is performed while PEDDR bits are cleared to 0, the pin states are read.
6	PE6	—*	R	
5	PE5	—*	R	
4	PE4	—*	R	
3	PE3	—*	R	
2	PE2	—*	R	
1	PE1	—*	R	
0	PE0	—*	R	

Note: * Determined by the states of pins PE7 to PE0.

9.15.1 Port H Data Direction Register (PHDDR)

PHDDR specifies input or output the port H pins using the individual bits. PHDDR cannot be read; if it is, the read value is undefined. Since this register is a write-only register, bit-manipulation instructions should not be used when writing. See section 2.9.4, Access Method for Registers with Write-only Bits.

Bit	Bit Name	Initial Value	R/W	Description
7	PH7DDR	0	W	When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port H pin an output pin. Clearing this bit to 0 makes the pin an input pin.
6	PH6DDR	0	W	
5	PH5DDR	0	W	
4	PH4DDR	0	W	
3	PH3DDR	0	W	
2	PH2DDR	0	W	
1	PH1DDR	0	W	
0	PH0DDR	0	W	

9.15.2 Port H Data Register (PHDR)

PHDR stores output data for port H pins.

Bit	Bit Name	Initial Value	R/W	Description
7	PH7DR	0	R/W	Output data for a pin is stored when the pin is specified as a general purpose output port.
6	PH6DR	0	R/W	
5	PH5DR	0	R/W	
4	PH4DR	0	R/W	
3	PH3DR	0	R/W	
2	PH2DR	0	R/W	
1	PH1DR	0	R/W	
0	PH0DR	0	R/W	

11.4 Operation

11.4.1 Pulse Output

Figure 11.2 shows an example of arbitrary duty pulse output.

1. Set the CCR1 bit in TCR to 0 and the CCLR0 bit to 1 to clear TCNT by a TCORA compare-match.
2. Set the OS3 to OS0 bits in TCSR to B'0110 to output 1 by a TCORA compare-match and 0 by a TCORB compare-match.

By the above settings, waveforms with the cycle of TCORA and the pulse width of TCORB can be output without software intervention.

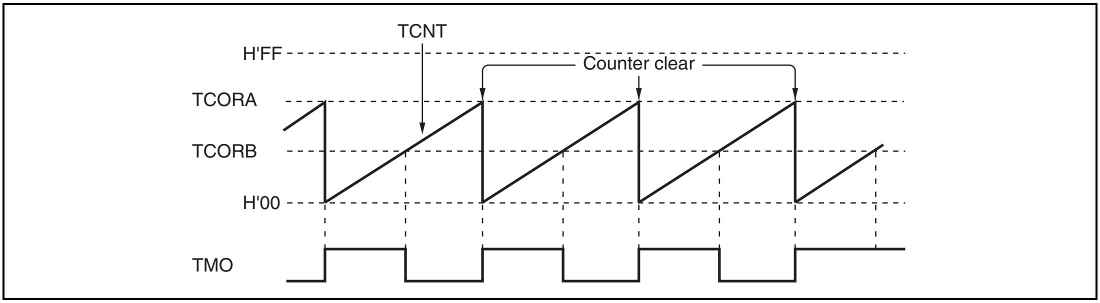


Figure 11.2 Example of Pulse Output

Table 13.12 Interrupt Sources of Serial Communication Interface Mode

Channel	Name	Interrupt Source	Interrupt Flag	DTC Activation	Priority*
0	ERI0	Receive Error	ORER, FER, PER	Not possible	High
	RXI0	Receive Data Full	RDRF	Possible	
	TXI0	Transmit Data Empty	TDRE	Possible	
	TEI0	Transmission End	TEND	Not possible	
1	ERI1	Receive Error	ORER, FER, PER	Not possible	
	RXI1	Receive Data Full	RDRF	Possible	
	TXI1	Transmit Data Empty	TDRE	Possible	
	TEI1	Transmission End	TEND	Not possible	
2	ERI2	Receive Error	ORER, FER, PER	Not possible	
	RXI2	Receive Data Full	RDRF	Possible	
	TXI2	Transmit Data Empty	TDRE	Possible	
	TEI2	Transmission End	TEND	Not possible	
3	ERI3	Receive Error	ORER, FER, PER	Not possible	
	RXI3	Receive Data Full	RDRF	Possible	
	TXI3	Transmit Data Empty	TDRE	Possible	
	TEI3	Transmission End	TEND	Not possible	
4	ERI4	Receive Error	ORER, FER, PER	Not possible	
	RXI4	Receive Data Full	RDRF	Possible	
	TXI4	Transmit Data Empty	TDRE	Possible	
	TEI4	Transmission End	TEND	Not possible	Low

Note: * Indicates the initial state immediately after a reset. Priorities in channels can be changed by the interrupt controller.

Bit	Bit Name	Initial Value	R/W	Description
3	UE	0	R/W	Underrun Error Indicates that an underrun error has occurred during data transmission. The IEB detects an underrun error occurrence when the IEB fetches data from IETBR while the TxRDY flag is set to 1, and the IEB sets the TxE flag and enters the wait state. Accordingly, when the TxRDY flag is not cleared even if data is written to IETBR, an underrun error occurs and data transmission is terminated. Note that the TxRDY flag must be cleared in data transmission by the CPU. [Setting condition] • When the IEB loads data from IETBR to the transmit shift register while the TxRDY flag is set to 1 [Clearing condition] • When writing 0 after reading UE = 1
2	TTME	0	R/W	Timing Error Set to 1 if data is not transmitted at the timing specified by the IEBus protocol during data transmission. The IEB sets the TxE flag and enters the wait state. [Setting condition] • When a timing error occurs during data transmission [Clearing condition] • When writing 0 after reading TTME = 1

18.3.6 Transmit Wait Cancel Register (TXCR)

TXCR controls the cancellation of transmit wait messages in mailboxes (buffers).

Bit	Bit Name	Initial Value	R/W	Description
15	TXCR7	0	R/W	These bits cancel the transmit wait message in the corresponding mailboxes 1 to 15. When TXCRn (n = 1 to 15) is set to 1, the transmit wait message in mailbox n is canceled. [Clearing condition] • Completion of TXPR clearing when transmit message is canceled normally Bit 8 is reserved. This bit is always read as 0 and the write value should always be 0.
14	TXCR6	0	R/W	
13	TXCR5	0	R/W	
12	TXCR4	0	R/W	
11	TXCR3	0	R/W	
10	TXCR2	0	R/W	
9	TXCR1	0	R/W	
8	—	0	R	
7	TXCR15	0	R/W	
6	TXCR14	0	R/W	
5	TXCR13	0	R/W	
4	TXCR12	0	R/W	
3	TXCR11	0	R/W	
2	TXCR10	0	R/W	
1	TXCR9	0	R/W	
0	TXCR8	0	R/W	

18.3.14 Receive Error Counter (REC)

REC is an 8-bit read-only register that functions as a counter indicating the number of receive message errors on the CAN bus. The count value is stipulated in the CAN protocol.

18.3.15 Transmit Error Counter (TEC)

TEC is an 8-bit read-only register that functions as a counter indicating the number of transmit message errors on the CAN bus. The count value is stipulated in the CAN protocol.

18.3.16 Unread Message Status Register (UMSR)

UMSR is a status register that indicates, for individual mailboxes, that a received message has been overwritten by a new receive message before being read. When overwritten by a new message, data in the unread receive message is lost.

Bit	Bit Name	Initial Value	R/W	Description
15	UMSR7	0	R/(W)*	Indicates that a received massage has been overwritten by a new message before being read. [Setting condition] When a new message is received before RXPR is cleared [Clearing condition] Writing 1
14	UMSR6	0	R/(W)*	
13	UMSR5	0	R/(W)*	
12	UMSR4	0	R/(W)*	
11	UMSR3	0	R/(W)*	
10	UMSR2	0	R/(W)*	
9	UMSR1	0	R/(W)*	
8	UMSR0	0	R/(W)*	
7	UMSR15	0	R/(W)*	
6	UMSR14	0	R/(W)*	
5	UMSR13	0	R/(W)*	
4	UMSR12	0	R/(W)*	
3	UMSR11	0	R/(W)*	
2	UMSR10	0	R/(W)*	
1	UMSR9	0	R/(W)*	
0	UMSR8	0	R/(W)*	

Note: * Only 1 can be written to this bit for clearing the flag.

CPU Interrupt Source Settings: CPU interrupt source settings are made in the interrupt mask register (IMR) and mailbox interrupt register (MBIMR). The message to be received is also specified. Data frame and remote frame receive wait interrupt requests can be generated for individual mailboxes in the MBIMR. Interrupt sources of the interrupt register (IRR) are enabled by IMR.

Arbitration Field Setting: To receive a message, the message identifier must be set in advance in the message control registers (MCx[1] to MCx[8]) for the receiving mailbox. When a message is received, all the bits in the receive message identifier are compared with those in each message control register identifier, and if a 100% match is found, the message is stored in the matching mailbox. Mailbox 0 has a local acceptance filter mask (LAFM) that allows Don't Care settings to be made. The LAFM setting can be made only for mailbox 0. By making the Don't Care setting for all the bits in the receive message identifier, messages of multiple identifiers can be received.

Examples:

- When the identifier of mailbox 1 is 010_1010_1010 (standard format), only one kind of message identifier can be received by mailbox 1:

  Identifier 1: 010_1010_1010
- When the identifier of mailbox 0 is 010_1010_1010 (standard format) and the LAFM setting is 000_0000_0011 (0: Care, 1: Don't Care), a total of four kinds of message identifiers can be received by mailbox 0:

  Identifier 1: 010_1010_1000

  Identifier 2: 010_1010_1001

  Identifier 3: 010_1010_1010

  Identifier 4: 010_1010_1011

Message Reception: When a message is received, a CRC check is performed automatically. If the result of the CRC check is normal, ACK is transmitted in the ACK field irrespective of whether the message can be received or not.

- Data frame reception

  If the received message is confirmed to be error-free by the CRC check, the identifier in the mailbox (and also LAFM in the case of mailbox 0 only) and the identifier of the receive message, are compared. If a complete match is found, the message is stored in the mailbox. The message identifier comparison is carried out on each mailbox in turn, starting with mailbox 0 and ending with mailbox 15. If a complete match is found, the comparison ends at that point, the message is stored in the matching mailbox, and the corresponding receive complete bit (RXPR0 to RXPR15) is set in the receive complete register (RXPR). However, when a mailbox 0 LAFM comparison is carried out, even if the identifier matches, the mailbox

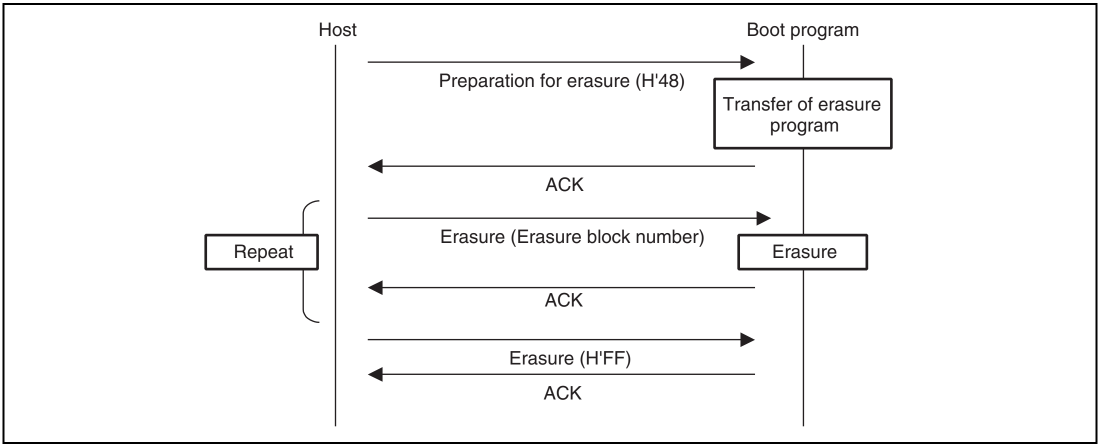

Host
Boot program
Preparation for erasure (H'48)
Transfer of erasure program
ACK
Repeat
Erasure (Erasure block number)
Erasure
ACK
Erasure (H'FF)
ACK

Figure 20.28 Erasure Sequence

(a) Erasure Selection

The boot program will transfer the erasure program. User MAT data is erased by the transferred erasure program.

Command	H'48

- Command, H'48, (one byte): Erasure selection

Response	H'06

- Response, H'06, (one byte): Response for erasure selection
  After the erasure program has been transferred, the boot program will return ACK.

Error Response	H'C8	ERROR

- Error Response, H'C8, (one byte): Error response to erasure selection
- ERROR: (one byte): Error code

  H'54: Selection processing error (transfer error occurs and processing is not completed)

(b) Block Erasure

The boot program will erase the contents of the specified block.

Command	H'58	Size	Block number	SUM

- Command, H'58, (one byte): Erasure
- Size (one byte): The number of bytes that represents the erasure block number
  This is fixed to 1.
- Block number (one byte): Number of the block to be erased

Register Abbreviation	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module
TCSR_0	OVF	WT / $\overline{\text{IT}}$	TME	—	—	CKS2	CKS1	CKS0	WDT_0
TCNT_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
RSTCSR	WOVF	RSTE	RSTS	—	—	—	—	—	
SMR_0*[1]	C/$\overline{\text{A}}$ (GM)	CHR (BLK)	PE (PE)	O/$\overline{\text{E}}$ (O/$\overline{\text{E}}$)	STOP (BCP1)	MP (BCP0)	CKS1 (CKS1)	CKS0 (CKS0)	SCI_0
BRR_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCR_0	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SSR_0*[1]	TDRE (TDRE)	RDRF (RDRF)	ORER (ORER)	FER (ERS)	PER (PER)	TEND (TEND)	MPB (MPB)	MPBT (MPBT)	
RDR_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCMR_0	—	—	—	—	SDIR	SINV	—	SMIF	
SMR_1*[1]	C/$\overline{\text{A}}$ (GM)	CHR (BLK)	PE (PE)	O/$\overline{\text{E}}$ (O/$\overline{\text{E}}$)	STOP (BCP1)	MP (BCP0)	CKS1 (CKS1)	CKS0 (CKS0)	SCI_1
BRR_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCR_1	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SSR_1*[1]	TDRE (TDRE)	RDRF (RDRF)	ORER (ORER)	FER (ERS)	PER (PER)	TEND (TEND)	MPB (MPB)	MPBT (MPBT)	
RDR_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCMR_1	—	—	—	—	SDIR	SINV	—	SMIF	
SMR_2*[1]	C/$\overline{\text{A}}$ (GM)	CHR (BLK)	PE (PE)	O/$\overline{\text{E}}$ (O/$\overline{\text{E}}$)	STOP (BCP1)	MP (BCP0)	CKS1 (CKS1)	CKS0 (CKS0)	SCI_2
BRR_2	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCR_2	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR_2	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SSR_2*[1]	TDRE (TDRE)	RDRF (RDRF)	ORER (ORER)	FER (ERS)	PER (PER)	TEND (TEND)	MPB (MPB)	MPBT (MPBT)	
RDR_2	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCMR_2	—	—	—	—	SDIR	SINV	—	SMIF	

23.3 Register States in Each Operating Mode

Register Abbreviation	Reset	High-Speed	Medium-Speed	Sleep	Module Stop	Watch	Software Standby	Hardware Standby	Module
MRA	Initialized	—	—	—	—	—	—	Initialized	DTC
SAR	Initialized	—	—	—	—	—	—	Initialized	
MRB	Initialized	—	—	—	—	—	—	Initialized	
DAR	Initialized	—	—	—	—	—	—	Initialized	
CRA	Initialized	—	—	—	—	—	—	Initialized	
CRB	Initialized	—	—	—	—	—	—	Initialized	
IECTR	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	IEB
IECMR	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEMCR	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEAR1	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEAR2	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IESA1	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IESA2	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IETBFL	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IETBR	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEMA1	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEMA2	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IERCTL	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IERBFL	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IERBR	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IELA1	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IELA2	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEFLG	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IETSR	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEIET	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IETEF	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IERSR	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEIER	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	
IEREF	Initialized	—	—	—	Initialized	Initialized	Initialized	Initialized	

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Reference current	During A/D conversion, D/A conversion	AI_{CC}	—	2.0	4.0	mA	
	Waiting for A/D conversion, D/A conversion		—	2.0	6.0	μA	
RAM standby voltage		V_{RAM}	3.0	—	—	V	

Notes: 1. If the A/D and D/A converters are not used, do not leave the AV_{CC}, V_{ref}, and AV_{SS} pins open. Apply a voltage 3.0 V to 5.5 V to the AV_{CC} and V_{ref} pins by connecting them to V_{CC}, for instance. Set $V_{ref} \leq AV_{CC}$.

2. Current consumption values are for $P1V_{CC} = P2V_{CC} = AV_{CC} = V_{CC}$, V_{IH} min. = V_{CC} – 0.2 V, V_{IL} max. = 0.2 V with all output pins unloaded and the on-chip pull-up MOS in the off state.

3. I_{CC} depends on V_{CC} and f as follows:

I_{CC} max. = –9.49 (mA) + 5.31 (mA/V) × Vcc + 1.00 (mA/MHz) × f + 0.03 (mA/(MHz • V)) × V_{CC} × f (normal operation)

I_{CC} max. = –8.58 (mA) + 5.04 (mA/V) × Vcc + 0.60 (mA/MHz) × f + 0.026 (mA/(MHz • V)) × V_{CC} × f (sleep mode)

Item	Symbol	Min.	Max.	Unit	Test Conditions
NMI setup time	t_{NMIS}	250	—	ns	Figure 24.8
NMI hold time	t_{NMIH}	10	—	ns	
NMI pulse width (exiting software standby mode)	t_{NMIW}	200	—	ns	
$\overline{IRQ}$ setup time	t_{IRQS}	250	—	ns	
$\overline{IRQ}$ hold time	t_{IRQH}	10	—	ns	
$\overline{IRQ}$ pulse width (exiting software standby mode)	t_{IRQW}	200	—	ns	

Note: * The regular specifications are supported in the H8S/2506 Group only.

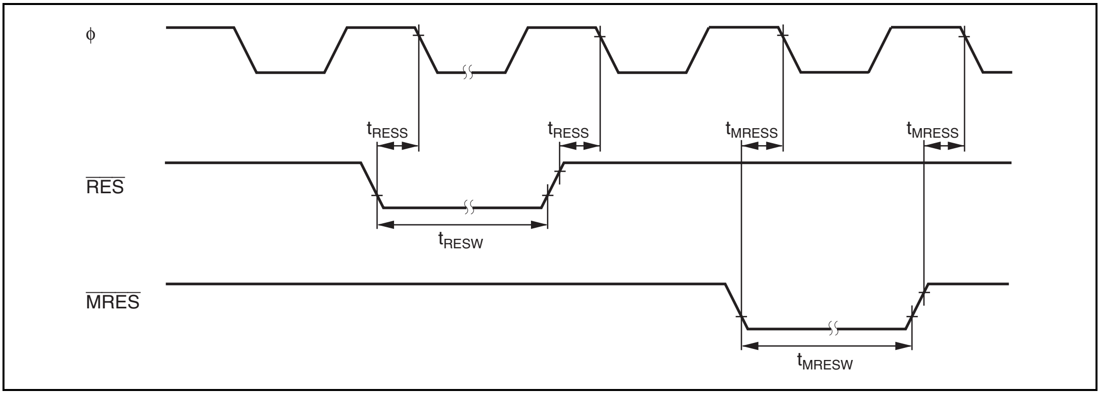


Figure 24.7 Reset Input Timing