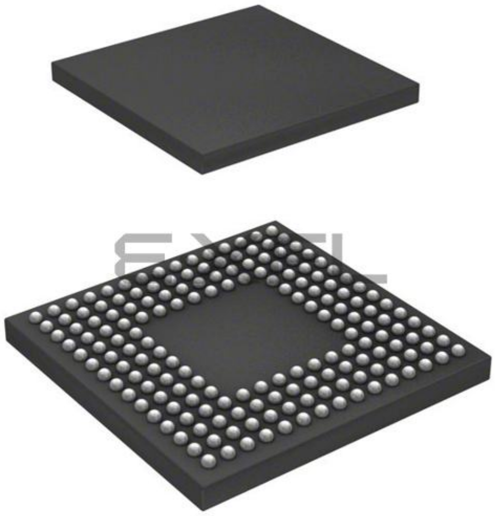


What is "Embedded - Microcontrollers"?

"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	SH2A-FPU
Core Size	32-Bit Single-Core
Speed	200MHz
Connectivity	CANbus, Ethernet, I²C, SCI, SPI, USB
Peripherals	DMA, PWM, WDT
Number of I/O	112
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x12b
Oscillator Type	External
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/r5f72167bdbg-u1

Origin of Activation Source	Activation Source	Vector Number	DTC Vector Address Offset	DTCE*[1]	Transfer Source	Transfer Destination	Priority
RSPI	SPRI	234	H'000007A8	DTCERD5	SPDR	Any location*[2]	High
	SPTI	235	H'000007AC	DTCERD4	Any location*[2]	SPDR	↑
SCI4	RXI4	237	H'000007B4	DTCERD3	SCRDR_4	Any location*[2]	
	TXI4	238	H'000007B8	DTCERD2	Any location*[2]	SCTDR_4	
SCI0	RXI0	241	H'000007C4	DTCERE15	SCRDR_0	Any location*[2]	
	TXI0	242	H'000007C8	DTCERE14	Any location*[2]	SCTDR_0	
SCI1	RXI1	245	H'000007D4	DTCERE13	SCRDR_1	Any location*[2]	
	TXI1	246	H'000007D8	DTCERE12	Any location*[2]	SCTDR_1	
SCI2	RXI2	249	H'000007E4	DTCERE11	SCRDR_2	Any location*[2]	
	TXI2	250	H'000007E8	DTCERE10	Any location*[2]	SCTDR_2	
SCIF3	RXI3	254	H'000007F8	DTCERE9	SCFRDR_3	Any location*[2]	↓
	TXI3	255	H'000007FC	DTCERE8	Any location*[2]	SCFTDR_3	Low

Notes: 1. The DTCE bits with no corresponding interrupt are reserved, and the write value should always be 0.

2. An external memory, a memory-mapped external device, an on-chip memory, or an on-chip peripheral module (except for DTC, BSC, UBC, AUD, FLASH, and DMAC) can be selected as the source or destination. Note that at least either the source or destination must be an on-chip peripheral module; transfer cannot be done among an external memory, a memory-mapped external device, and an on-chip memory.

3. Read to a message control field in mailbox 0 by using a block transfer mode or etc.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{\text{CS4}}$ Assertion to $\overline{\text{RD}}$, $\overline{\text{WRxx}}$ Assertion Specify the number of delay cycles from address and $\overline{\text{CS4}}$ assertion to $\overline{\text{RD}}$ and $\overline{\text{WRxx}}$ assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles
10 to 7	WR[3:0]	1010	R/W	Number of Read Access Wait Cycles Specify the number of cycles that are necessary for read access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)

(9) Relationship between Refresh Requests and Bus Cycles

If a refresh request occurs during bus cycle execution, the refresh cycle must wait for the bus cycle to be completed. If a refresh request occurs while the bus is released by the bus arbitration function, the refresh will not be executed until the bus mastership is acquired. This LSI has the $\overline{\text{REFOUT}}$ pin to request the bus while waiting for refresh execution. For $\overline{\text{REFOUT}}$ pin function selection, see section 22, Pin Function Controller (PFC). This LSI continues to assert $\overline{\text{REFOUT}}$ (low level) until the bus is acquired.

On receiving the asserted $\overline{\text{REFOUT}}$ signal, the external device must negate the $\overline{\text{BREQ}}$ signal and return the bus. If the external bus does not return the bus for a period longer than the specified refresh interval, refresh cannot be executed and the SDRAM contents may be lost.

If a new refresh request occurs while waiting for the previous refresh request, the previous refresh request is deleted. To refresh correctly, a bus cycle longer than the refresh interval or the bus mastership occupation must be prevented from occurring.

If a bus mastership is requested during self-refresh, the bus will not be released until the refresh is completed.

10.3.3 DMA Transfer Count Registers (DMATCR)

The DMA transfer count registers (DMATCR) are 32-bit readable/writable registers that specify the number of DMA transfers. The transfer count is 1 when the setting is H'00000001, 16,777,215 when H'00FFFFFF is set, and 16,777,216 (the maximum) when H'00000000 is set. During a DMA transfer, these registers indicate the remaining transfer count.

The upper eight bits of DMATCR are always read as 0, and the write value should always be 0. To transfer data in 16 bytes, one 16-byte transfer (128 bits) counts one.

DMATCR is initialized to H'00000000 by a reset and retains the value in software standby mode and module standby mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Item	**Channel 0**	**Channel 1**	**Channel 2**	**Channel 3**	**Channel 4**	**Channel 5**
Phase counting mode	—	√	√	—	—	—
Buffer operation	√	—	—	√	√	—
Dead time compensation counter function	—	—	—	—	—	√
DMAC activation	TGRA_0 compare match or input capture	TGRA_1 compare match or input capture	TGRA_2 compare match or input capture	TGRA_3 compare match or input capture	TGRA_4 compare match or input capture and TCNT overflow or underflow	—
DTC activation	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture or TCNT overflow or underflow	TGR compare match or input capture
A/D converter start trigger	TGRA_0 compare match or input capture TGRE_0 compare match	TGRA_1 compare match or input capture	TGRA_2 compare match or input capture	TGRA_3 compare match or input capture	TGRA_4 compare match or input capture TCNT_4 underflow (trough) in complementary PWM mode	—

(16) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Phase Counting Mode

Figure 11.154 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in phase counting mode after re-setting.

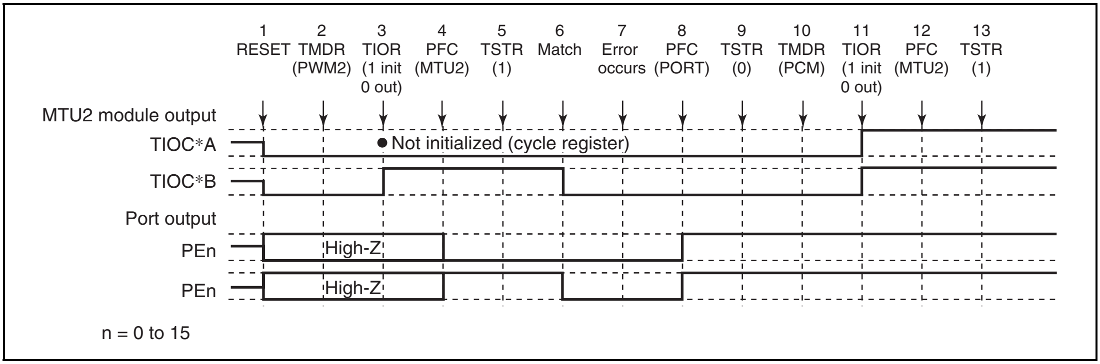


Figure 11.154 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

1 to 9 are the same as in figure 11.151.

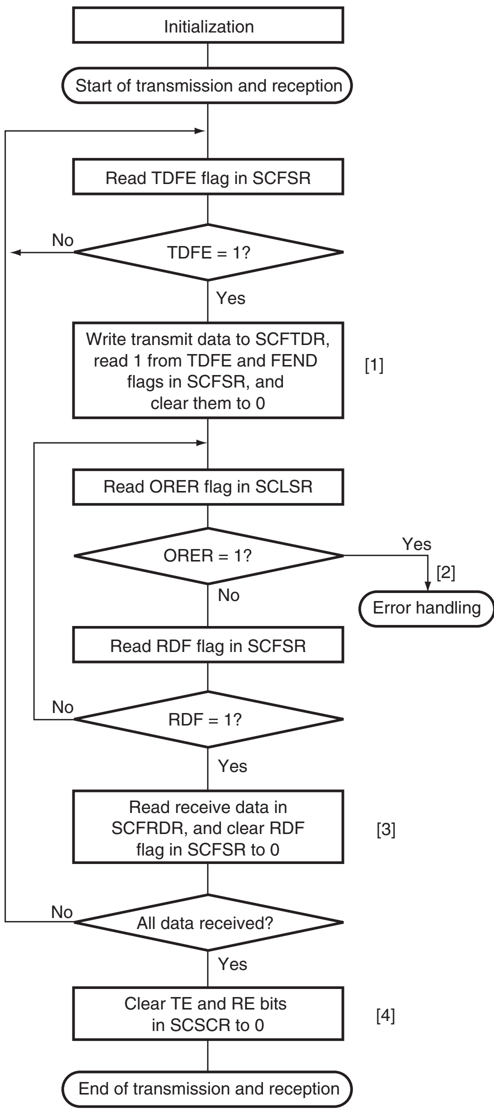
10. Set phase counting mode.
11. Initialize the pins with TIOR.
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Bit	Bit Name	Initial value	R/W	Description
2	SPB1DT	Undefined	W	Clock Port Data in Serial Port Specifies the data output through the SCK pin in the serial port. Output should be enabled by the SPB1IO bit (for details, refer to the SPB1IO bit description). When output is enabled, the SPB1DT bit value is output through the SCK pin. 0: Low level is output 1: High level is output
1	—	0	—	Reserved This bit is always read as 0. The write value should always be 0.
0	SPB0DT	1	W	Serial Port Break Data Controls the TXD pin by the TE bit in SCSCR. However, TXD pin function should be selected by the pin function controller (PFC). This is a read-only bit. The read value is undefined. **TE bit setting in SCSCR** / **SPB0DT bit setting** / **TXD pin state** 0 / 0 / Low output 0 / 1 / High output (initial state) 1 / * / Transmit data output in accord with serial core logic. Note: * Don't care

Bit	Bit Name	Initial Value	R/W	Description
6	RIE	0	R/W	Receive Interrupt Enable Enables or disables the receive FIFO data full (RXI) interrupts requested when the RDF flag or DR flag in serial status register (SCFSR) is set to 1, receive-error (ERI) interrupts requested when the ER flag in SCFSR is set to 1, and break (BRI) interrupts requested when the BRK flag in SCFSR or the ORER flag in line status register (SCLSR) is set to 1. 0: Receive FIFO data full interrupt (RXI), receive-error interrupt (ERI), and break interrupt (BRI) requests are disabled 1: Receive FIFO data full interrupt (RXI), receive-error interrupt (ERI), and break interrupt (BRI) requests are enabled* Note: * RXI interrupt requests can be cleared by reading the DR or RDF flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE to 0. ERI or BRI interrupt requests can be cleared by reading the ER, BR or ORER flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE and REIE to 0.
5	TE	0	R/W	Transmit Enable Enables or disables the serial transmitter. 0: Transmitter disabled 1: Transmitter enabled* Note: * Serial transmission starts after writing of transmit data into SCFTDR. Select the transmit format in SCSMR and SCFCR and reset the transmit FIFO before setting TE to 1.

- Transmitting and Receiving Serial Data Simultaneously (Clocked Synchronous Mode)

Figure 17.16 shows a sample flowchart for transmitting and receiving serial data simultaneously. Use the following procedure for the simultaneous transmission/reception of serial data, after enabling the SCIF for transmission/reception.



[1] SCIF status check and transmit data write:

Read SCFSR and check that the TDFE and TEND flags are set to 1, then write transmit data to SCFTDR. Read 1 from the TDFE and TEND flags, then clear these flags to 0. The transition of the TDFE flag from 0 to 1 can also be identified by a TXI interrupt.

[2] Receive error handling:

Read the ORER flag in SCLSR to identify any error, perform the appropriate error handling, then clear the ORER flag to 0. Reception cannot be resumed while the ORER flag is set to 1.

[3] SCIF status check and receive data read:

Read SCFSR and check that RDF flag = 1, then read the receive data in SCFRDR, and clear the RDF flag to 0. The transition of the RDF flag from 0 to 1 can also be identified by an RXI interrupt.

[4] Serial transmission and reception continuation procedure:

To continue serial transmission and reception, read 1 from the RDF flag and the receive data in SCFRDR, and clear the RDF flag to 0 before receiving the MSB in the current frame. Similarly, read 1 from the TDFE flag to confirm that writing is possible before transmitting the MSB in the current frame. Then write data to SCFTDR and clear the TDFE flag to 0.

Note: When switching from a transmit operation or receive operation to simultaneous transmission and reception operations, clear the TE and RE bits to 0, and then set them simultaneously to 1.

Figure 17.16 Sample Flowchart for Transmitting/Receiving Serial Data

18.4.2 Controlling RSPI Pins

According to the MSTR, MODFEN and SPMS bits in the RSPI control register (SPCR) and the SPOM bit in the RSPI pin control register (SPPCR), the RSPI can automatically switch pin directions and output modes. Table 18.6 shows the relationship between pin states and bit settings.

Table 18.6 Relationship between Pin States and Bit Settings

		Pin State*1	
Mode	**Pin**	**SPOM = 0**	**SPOM = 1**
Single-master mode (SPI) (MSTR = 1, MODFEN = 0, SPMS = 0)	RSPCK	CMOS output	Open-drain output
	SSL0 to SSL3	CMOS output	Open-drain output
	MOSI	CMOS output	Open-drain output
	MISO	Input	Input
Multi-master mode (SPI) (MSTR = 1, MODFEN = 1, SPMS = 0)	RSPCK*2	CMOS output/Hi-Z	Open-drain output/Hi-Z
	SSL0	Input	Input
	SSL1 to SSL3*2	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MOSI*2	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MISO	Input	Input
Slave mode (SPI) (MSTR = 0, SPMS = 0)	RSPCK	Input	Input
	SSL0	Input	Input
	SSL1 to SSL3	Hi-Z	Hi-Z
	MOSI	Input	Input
	MISO*3	CMOS output/Hi-Z	Open-drain output/Hi-Z
Master (clock synchronous) (MSTR = 1, MODFEN = 0, SPMS = 1)	RSPCK	CMOS output	Open-drain output
	SSL0 to SSL3*4	Hi-Z	Hi-Z
	MOSI	CMOS output	Open-drain output
	MISO	Input	Input

21.4 Application Note

21.4.1 Test Mode Settings

The RCAN-ET has various test modes. The register TST[2:0] (MCR[10:8]) is used to select the RCAN-ET test mode. The default (initialised) settings allow RCAN-ET to operate in Normal mode. The following table is examples for test modes.

Test Mode can be selected only while in configuration mode. The user must then exit the configuration mode (ensuring BCR0/BCR1 is set) in order to run the selected test mode.

Bit10: TST2	Bit9: TST1	Bit8: TST0	Description
0	0	0	Normal Mode (initial value)
0	0	1	Listen-Only Mode (Receive-Only Mode)
0	1	0	Self Test Mode 1 (External)
0	1	1	Self Test Mode 2 (Internal)
1	0	0	Write Error Counter
1	0	1	Error Passive Mode
1	1	0	Setting prohibited
1	1	1	Setting prohibited

Normal Mode: RCAN-ET operates in the normal mode.

Listen-Only Mode: ISO-11898 requires this mode for baud rate detection. The Error Counters are cleared and disabled so that the TEC/REC does not increase the values, and the Tx Output is disabled so that RCAN-ET does not generate error frames or acknowledgment bits. IRR13 is set when a message error occurs.

Self Test Mode 1: RCAN-ET generates its own Acknowledge bit, and can store its own messages into a reception mailbox (if required). The Rx/Tx pins must be connected to the CAN bus.

Self Test Mode 2: RCAN-ET generates its own Acknowledge bit, and can store its own messages into a reception mailbox (if required). The Rx/Tx pins do not need to be connected to the CAN bus or any external devices, as the internal Tx is looped back to the internal Rx. Tx pin outputs only recessive bits and Rx pin is disabled.

Table 22.3 Multiplexed Pins (Port C)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)
C	PC15 I/O (Port)	A15 output (BSC)	—	IRQ2 input (INTC)	TCLKD input (MTU2)	—	—	—
	PC14 I/O (Port)	A14 output (BSC)	—	IRQ1 input (INTC)	TCLKC input (MTU2)	—	—	—
	PC13 I/O (Port)	A13 output (BSC)	—	IRQ0 input (INTC)	TCLKB input (MTU2)	—	—	—
	PC12 I/O (Port)	A12 output (BSC)	—	—	TCLKA input (MTU2)	—	—	—
	PC11 I/O (Port)	A11 output (BSC)	—	—	TIOC1B I/O (MTU2)	CTx0 output (RCAN-ET)	TXD0 output (SCI)	—
	PC10 I/O (Port)	A10 output (BSC)	—	—	TIOC1A I/O (MTU2)	CRx0 input (RCAN-ET)	RXD0 input (SCI)	—
	PC9 I/O (Port)	A9 output (BSC)	—	—	—	CTx0 output (RCAN-ET)	TXD0 output (SCI)	—
	PC8 I/O (Port)	A8 output (BSC)	—	—	—	CRx0 input (RCAN-ET)	RXD0 input (SCI)	—
	PC7 I/O (Port)	A7 output (BSC)	—	—	—	—	—	—
	PC6 I/O (Port)	A6 output (BSC)	—	—	—	—	—	—
	PC5 I/O (Port)	A5 output (BSC)	—	—	—	—	—	—
	PC4 I/O (Port)	A4 output (BSC)	—	—	—	—	—	—
	PC3 I/O (Port)	A3 output (BSC)	—	—	—	—	—	—
	PC2 I/O (Port)	A2 output (BSC)	—	—	—	—	—	—
	PC1 I/O (Port)	A1 output (BSC)	—	—	—	—	—	—
	PC0 I/O (Port)	A0 output (BSC)	—	IRQ4 input (INTC)	—	$\overline{\text{POE0}}$ input (POE2)	—	—

Register	Bit	Transfer Mode	Interrupt Source	Description	Interrupt Request Signal	DMAC/DTC Activation
USBIFR2	0	Bulk_out transfer (EP1)	EP1FULL	EP1FIFO full	USI0 or USI1	USBRXI0
	1	Bulk_in transfer (EP2)	EP2ALLEMP	EP2FIFO all empty	USI0 or USI1	×
	2		EP2EMPTY	EP2FIFO empty	USI0 or USI1	USBTXI0
	3		EP2TR	EP2 transfer request	USI0 or USI1	×
	4	Interrupt_in transfer (EP3)	EP3TS	EP3 transmit complete	USI0 or USI1	×
	5		EP3TR	EP3 transfer request	USI0 or USI1	×
	6	—	Reserved	—	—	—
	7	—	Reserved	—	—	—
USBIFR3	0	Bulk_out transfer (EP4)	EP4FULL	EP4FIFO full	USI0 or USI1	USBRXI1
	1	Bulk_in transfer (EP5)	EP5ALLEMP	EP5FIFO all empty	USI0 or USI1	×
	2		EP5EMPTY	EP5FIFO empty	USI0 or USI1	USBTXI1
	3		EP5TR	EP5 transfer request	USI0 or USI1	×
	4	Interrupt_in transfer (EP6)	EP6TS	EP6 transmit complete	USI0 or USI1	×
	5		EP6TR	EP6 transfer request	USI0 or USI1	×
	6	—	Reserved	—	—	—
	7	—	Reserved	—	—	—
USBIFR4	0	Bulk_out transfer (EP7)	EP7FULL	EP7FIFO full	USI0 or USI1	×
	1	—	Reserved	—	—	—
	2	Bulk_in transfer (EP8)	EP8EMPTY	EP8FIFO empty	USI0 or USI1	×
	3		EP8TR	EP8 transfer request	USI0 or USI1	×
	4	Interrupt_in transfer (EP9)	EP9TS	EP9 transmit complete	USI0 or USI1	×
	5		EP9TR	EP9 transfer request	USI0 or USI1	×
	6	—	Reserved	—	—	—
	7	—	Reserved	—	—	—

Note: * EP0-related interrupt sources must be assigned to the same interrupt request signal.

25.3.6 MAC Address Low Register (MALR)

MALR is a 32-bit readable/writable register that specifies the lower 16 bits of 48-bit MAC address. This register is normally set in the initialization process after a reset. The MAC address setting must not be changed while the transmitting and receiving functions are enabled. Reset the EtherC and E-DMAC with the SWR bit in EDMR of the E-DMAC, and then set the MAC address again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MA[15:0]	All 0	R/W	MAC Address Bits 15 to 0 These bits are used to set the lower 16 bits of the MAC address. If the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'89AB in this register.

25.3.20 Automatic PAUSE Frame Register (APR)

APR is used to set the TIME parameter value of an automatic PAUSE frame. When an automatic PAUSE frame is transmitted, the value set in this register is used as the TIME parameter of the PAUSE frame.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	AP[15:0]	All 0	R/W	Automatic PAUSE These bits set the TIME parameter value of an automatic PAUSE frame. One bit is equivalent to 512-bit time.

Bit	Bit Name	Initial Value	R/W	Description
7	USBSEL*[1]	1	R/W	USB Clock Select Selects the on-chip CPG or the USB oscillator as the source of the USB clock. 0: On-chip CPG 1: USB oscillator
6	MSTP66*[2]	1	R/W	Module Stop 66 When the MSTP66 bit is set to 1, the supply of the clock to the USB is halted. 0: USB runs. 1: Clock supply to USB halted.
5	USBCLK	0	R/W	USB Oscillator Stop When the USBCLK bit is set to 1, the oscillator dedicated for the USB stops. 0: USB oscillator operates. 1: USB oscillator stops.
4	MSTP64	1	R/W	Module Stop 64 When the MSTP64 bit is set to 1, the supply of the clock to the RCAN-ET is halted. 0: RCAN-ET runs. 1: Clock supply to RCAN-ET halted.
3 to 0	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.

Notes: When using the USB, Follow the notes shown below. Otherwise the clock will not be generated correctly so that USB can be operated improperly.

1. When selecting the on-chip CPG, set the frequency of the input clock to 12MHz.
2. When using the USB, set the frequency of the peripheral clock (Pϕ) to 13 MHz or more.

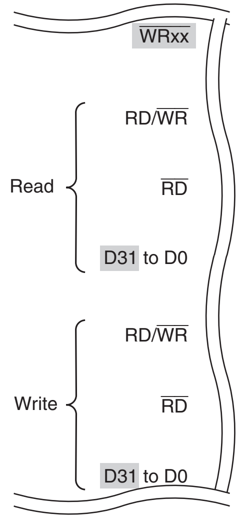
33.3.12 RSPI Timing

Table 33.16 SPI Timing

Conditions: $V_{CC}Q = PLLV_{CC} = DrV_{CC} = 3.0$ to 3.6 V, $AV_{CC} = AVREF = 4.5$ to 5.5V,
$V_{SS} = PLLV_{SS} = DrV_{SS} = AVREFV_{SS} = AV_{SS} = 0$ V,
Ta = –40°C to +85°C (Industrial specifications)

Item		Symbol	Min.	Typ.	Max.	Unit	Figure
RSPCK clock cycle*1	Master	t_{SPcyc}	2	—	4096	t_{Pcyc}	Figure 33.49
	Slave		8	—	4096		
RSPCK clock cycle high pulse width	Master	t_{SPCKWH}	$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF})/2 - 3$	—	—	ns	
	Slave		$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF})/2$	—	—		
RSPCK clock cycle low pulse width	Master	t_{SPCKWL}	$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF})/2 - 3$	—	—	ns	
	Slave		$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF})/2$	—	—		
RSPCK clock rise/fall time*2	Master	t_{SPCKR}, t_{SPCKF}	—	—	5	ns	
	Slave		—	—	1	t_{Pcyc}	
Data input setup time	Master	t_{SU}	25	—	—	ns	Figures 33.50 to 33.53
	Slave		$20 - 2 \times t_{Pcyc}$	—	—		
Data input hold time	Master	t_H	0	—	—	ns	
	Slave		$20 + 2 \times t_{Pcyc}$	—	—		
SSL setup time	Master	t_{LEAD}	1	—	8	t_{SPcyc}	
	Slave		4	—	—	t_{Pcyc}	
SSL hold time	Master	t_{LAG}	1	—	8	t_{SPcyc}	
	Slave		4	—	—	t_{Pcyc}	
Data output delay time	Master	t_{OD}	—	—	10	ns	
	Slave		—	—	$3 \times t_{Pcyc} + 15$		
Data output hold time	Master	t_{OH}	0	—	—	ns	
	Slave		0	—	—		
Continuous transmission delay time	Master	t_{TD}	$t_{SPcyc} + 2 \times t_{Pcyc}$	—	$8 \times t_{SPcyc} + 2 \times t_{Pcyc}$	ns	
	Slave		$4 \times t_{Pcyc}$	—	—		

Pin Function		Pin State									
		Reset State					Power-Down State				
		Power-On									
		Expansion without ROM									
Type	Pin Name	16 Bits	32 Bits	Expansion with ROM	Single Chip	Manual	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
MTU2	TIOC1B (PE5), TIOC2A (PE6)	Z				I/O	Z (MZIZEL in HCPCR = 0)	I/O	I/O	I/O*8	I/O
							K*1 (MZIZEL in HCPCR = 1)				
	TIOC1B (PC11), TIOC2A (PB0)	Z				I/O	K*1	I/O	I/O	I/O	I/O
	TIOC2B	Z				I/O	K*1	I/O	I/O	I/O	I/O
	TIOC3A, TIOC3C	Z				I/O	K*1	I/O	I/O	I/O	I/O
	TIOC3B, TIOC3D	Z				I/O	Z (MZIZEH in HCPCR = 0)	I/O	I/O	I/O*7	Z
							K*1 (MZIZEH in HCPCR = 1)				
	TIOC4A, TIOC4B, TIOC4C, TIOC4D	Z				I/O	Z (MZIZEH in HCPCR = 0)	I/O	I/O	I/O*7	Z
							K*1 (MZIZEH in HCPCR = 1)				
	TIC5U, TIC5V, TIC5W	Z				I	Z	I	I	I	I
MTU2S	TIOC3AS, TIOC3CS	Z				I/O	K*1	I/O	I/O	I/O	I/O
	TIOC3BS (PD10), TIOC3DS (PD11), TIOC4AS (PD12), TIOC4BS (PD13), TIOC4CS (PD14), TIOC4DS (PD15)	Z				I/O	Z (MZIZDL in HCPCR = 0)	I/O	I/O	I/O*6	Z
							K*1 (MZIZDL in HCPCR = 1)				

<table>
<tr><th>Item</th><th>Page</th><th>Revision (See Manual for Details)</th></tr>
<tr><td>Figure 9.37 Basic Access Timing for SRAM with Byte Selection (BAS = 0)
to
Figure 9.39 Wait Timing for SRAM with Byte Selection (BAS = 1) (SW[1:0] = 01, WR[3:0] = 0001, HW[1:0] = 01)</td><td>372
to
374</td><td>Amended

</td></tr>
<tr><td>Figure 9.40 Example of Connection with 16-Bit Data-Width SRAM with Byte Selection</td><td>375</td><td>Figure replaced</td></tr>
<tr><td>Figure 9.41 Example of Connection with 16-Bit Data-Width SRAM with Byte Selection</td><td>376</td><td>Figure added</td></tr>
<tr><td>Table 9.21 Conditions for Determining Number of Idle Cycles</td><td>380</td><td>Amended and added
<table>
<tr><th>No.</th><th>Condition</th><th>Description</th></tr>
<tr><td>(5)</td><td>Read data transfer cycle</td><td>One idle cycle is inserted after a read access is completed. This idle cycle is not generated for the first or middle cycles in divided access cycles. This is neither generated when the HW[1:0] bits in CSnWCR are not B'00.</td></tr>
</table>
Note: * This is the case for consecutive read operations when the data read are stored in separate registers.</td></tr>
</table>