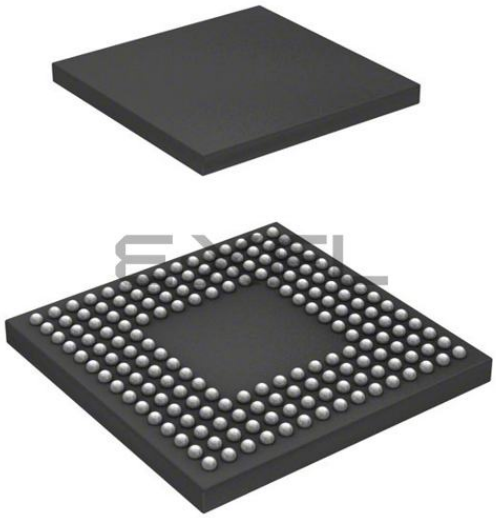


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

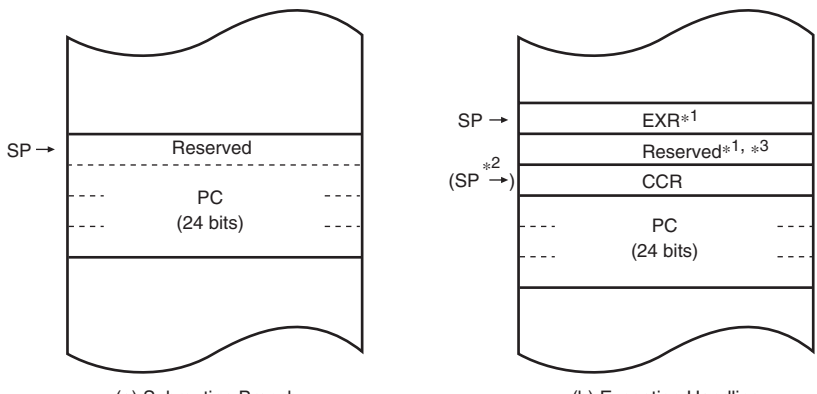
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Not For New Designs
Core Processor	H8S/2600
Core Size	16-Bit
Speed	34MHz
Connectivity	Ethernet, I²C, LPC, PECI, SCI, SSU, USB
Peripherals	DMA, POR, PWM, WDT
Number of I/O	110
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	40K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-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/r4f2472vbr34dv

The memory indirect addressing mode (@@aa:8) employed in the JMP and JSR instructions uses an 8-bit absolute address included in the instruction code to specify a memory operand that contains a branch address. In advanced mode the operand is a 32-bit longword operand, providing a 32-bit branch address. The upper 8 bits of these 32 bits is a reserved area that is regarded as H'00. Branch addresses can be stored in the area from H'00000000 to H'000000FF. Note that the first part of this range is also used for the exception vector table.

- Stack Structure

  In advanced mode, when the program counter (PC) is pushed onto the stack in a subroutine call, and the PC, condition-code register (CCR), and extended control register (EXR) are pushed onto the stack in exception handling, they are stored as shown in figure 2.4. When EXR is not pushed onto the stack in interrupt control mode 0. For details, see section 4, Exception Handling.



Notes: 1. When EXR is not used it is not stored on the stack.
2. SP when EXR is not used.
3. Ignored when returning.

Figure 2.4 Stack Structure in Advanced Mode

8.1.9 Port 9

Port 9 is an 8-bit I/O port. Port 9 pins can function as the bus control input/output pins. The pin functions change according to the operating mode. Port 9 has the following registers.

- Port 9 data direction register (P9DDR)
- Port 9 data register (P9DR)

(1) Port 9 Data Direction Register (P9DDR)

The individual bits of P9DDR specify input or output for the port 9 pins.

Bit	Bit Name	Initial Value	R/W	Description
7	P97DDR	0	W	If port 9 pins are specified for use as the general I/O port, the corresponding pins function as output port when the P9DDR bits are set to 1, and as input port when cleared to 0.
6	P96DDR	0	W	
5	P95DDR	0	W	
4	P94DDR	0	W	
3	P93DDR	0	W	
2	P92DDR	0	W	
1	P91DDR	0	W	
0	P90DDR	0	W	

- PA5/$\overline{\text{ExIRQ5}}$/EVENT5/WOL

  The pin function is switched as shown below according to the setting of the and the PA5DDR bit.

  Setting the ISS5 bit in ISSR makes the pin to function as the $\overline{\text{ExIRQ5}}$ input pin.

  When using this pin as the $\overline{\text{ExIRQ5}}$ input, or EVENT5 input pin, clear the PA5DDR bit to 0. Though the settings for the EVENT input pin have been made, set the PA5DDR bit to 1 to use the pin as the PA5 output pin.

  When the module stop mode is cleared in both the EtherC and E-DMAC, this pin functions as the WOL output pin.

PA5DDR	0	1
Pin function	PA5 input pin	PA5 output pin
	$\overline{\text{ExIRQ5}}$ input pin/EVENT5 input pin	

- PA4/$\overline{\text{ExIRQ4}}$/EVENT4, PA3/$\overline{\text{ExIRQ3}}$/EVENT3, PA2/$\overline{\text{ExIRQ2}}$/EVENT2, PA1/$\overline{\text{ExIRQ1}}$/EVENT1, PA0/$\overline{\text{ExIRQ0}}$/EVENT0

  The pin function is switched as shown below according to the PAnDDR bit.

  Setting the ISSn bit in ISSR makes the pin to function as the $\overline{\text{ExIRQn}}$ input pin.

  When using this pin as the $\overline{\text{ExIRQn}}$ input or EVENTn input pin, clear the PAnDDR bit to 0. Though the settings for the EVENT input pin have been made, set the PAnDDR bit to 1 to use the pin as the PAn output pin.

PAnDDR	0	1
Pin function	PAn input pin	PAn output pin
	$\overline{\text{ExIRQn}}$ input pin/EVENTn input pin	

[Legend] n = 4 to 0

(5) Input Pull-Up MOS

Port A has built-in input pull-up MOSs that can be controlled by software. This input pull-up MOS can be used in any operating mode, and can be specified as on or off on a bit-by-bit basis.

PAnDDR	0		1
PAnODR	1	0	X
PAn pull-up MOS	ON	OFF	OFF

[Legend] n = 7 to 0, X: Don't care.

9.2 Input/Output Pins

Table 9.1 lists the PWMX (D/A) module input and output pins.

Table 9.1 Pin Configuration

Name	Abbreviation	I/O	Function
PWMX output pin 0	PWX0	Output	PWM timer pulse output of PWMX_0 channel A
PWMX output pin 1	PWX1	Output	PWM timer pulse output of PWMX_0 channel B
PWMX output pin 2	PWX2	Output	PWM timer pulse output of PWMX_1 channel A
PWMX output pin 3	PWX3	Output	PWM timer pulse output of PWMX_1 channel B

9.3 Register Descriptions

The PWMX (D/A) module has the following registers. For details on the module stop control register, see section 28.1.3, Module Stop Control Registers H, L, and A (MSTPCRH, MSTPCRL, MSTPCRA).

- PWMX (D/A) counter (DACNT)
- PWMX (D/A) data register A (DADRA)
- PWMX (D/A) data register B (DADRB)
- PWMX (D/A) control register (DACR)
- Peripheral clock select register (PCSR)

Note: The same addresses are shared by DADRA and DACR, and by DADRB and DACNT. Switching is performed by the REGS bit in DACNT or DADRB.

11.5 Interrupt Sources

TMR_0, TMR_1, TMR_Y and TMR_X can generate three types of interrupts: CMIA, CMIB, and OVI.

Table 11.3 shows the interrupt sources and priorities. Each interrupt source can be enabled or disabled independently by interrupt enable bits in TCR or TCSR. Independent signals are sent to the interrupt controller for each interrupt.

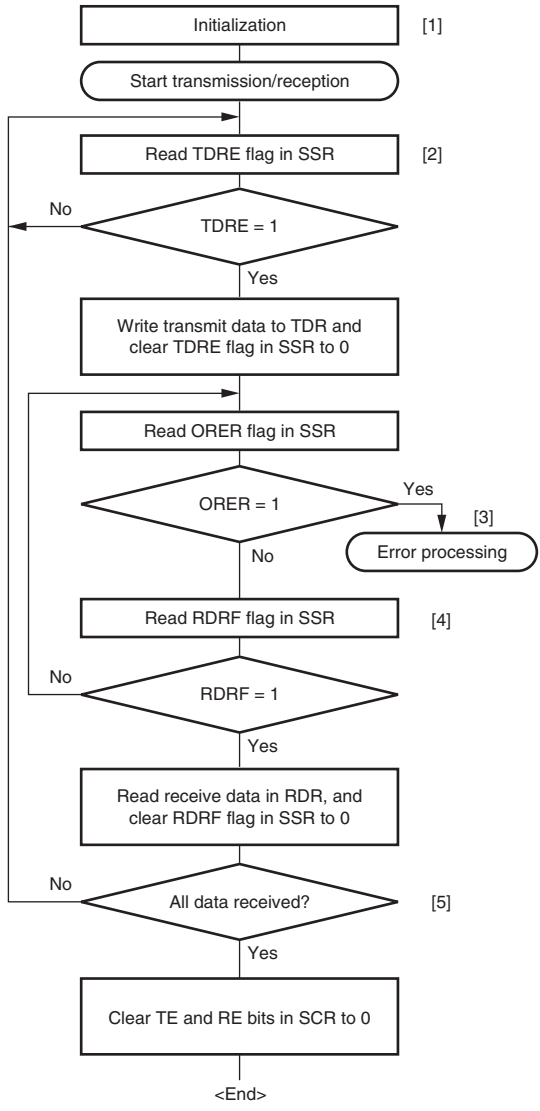
The CMIA and CMIB interrupts can be used as on-chip DTC activation interrupt sources.

Table 11.3 Interrupt Sources of 8-Bit Timers TMR_0, TMR_1, TMR_Y, and TMR_X

Channel	Name	Interrupt Source	Interrupt Flag	DTC Activation	Interrupt Priority
TMR_X	CMIAX	TCORA_X compare-match	CMFA	Possible	High
	CMIBX	TCORB_X compare-match	CMFB	Possible	
	OVIX	TCNT_X overflow	OVF	Not possible	
TMR_0	CMIA0	TCORA_0 compare-match	CMFA	Possible	
	CMIB0	TCORB_0 compare-match	CMFB	Possible	
	OVI0	TCNT_0 overflow	OVF	Not possible	
TMR_1	CMIA1	TCORA_1 compare-match	CMFA	Possible	
	CMIB1	TCORB_1 compare-match	CMFB	Possible	
	OVI1	TCNT_1 overflow	OVF	Not possible	
TMR_Y	CMIAY	TCORA_Y compare-match	CMFA	Possible	
	CMIBY	TCORB_Y compare-match	CMFB	Possible	
	OVIY	TCNT_Y overflow	OVF	Not possible	Low

Bit	Bit Name	Initial Value	R/W	Description
5	ORER	0	R/(W)*	Overrun Error [Setting condition] When the next serial reception is completed while RDRF = 1 [Clearing condition] When 0 is written to ORER after reading ORER = 1
4	FER	0	R/(W)*	Framing Error [Setting condition] When the stop bit is 0 [Clearing condition] When 0 is written to FER after reading FER = 1 In 2-stop-bit mode, only the first stop bit is checked.
3	PER	0	R/(W)*	Parity Error [Setting condition] When a parity error is detected during reception [Clearing condition] When 0 is written to PER after reading PER = 1
2	TEND	1	R	Transmit End [Setting conditions] • When the TE bit in SCR is 0 • When TDRE = 1 at transmission of the last bit of a 1-byte serial transmit character [Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When a TXI interrupt request is issued allowing DTC to write data to TDR
1	MPB	0	R	Multiprocessor Bit MPB stores the multiprocessor bit in the receive frame. When the RE bit in SCR is cleared to 0, its previous state is retained.
0	MPBT	0	R/W	Multiprocessor Bit Transfer MPBT stores the multiprocessor bit to be added to the transmit frame.

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



[1] SCI initialization:
The TxD pin is designated as the transmit data output pin, and the RxD pin is designated as the receive data input pin, enabling simultaneous transmit and receive operations.

[2] SCI status check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0.
Transition of the TDRE flag from 0 to 1 can also be identified by a TXI interrupt.

[3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, and after performing the appropriate error processing, clear the ORER flag to 0. Transmission/reception cannot be resumed if the ORER flag is set to 1.

[4] SCI status check and receive data read:
Read SSR and check that the RDRF flag is set to 1, then read the receive data in RDR and clear the RDRF flag to 0. Transition of the RDRF flag from 0 to 1 can also be identified by an RXI interrupt.

[5] Serial transmission/reception continuation procedure:
To continue serial transmission/reception, before the MSB (bit 7) of the current frame is received, finish reading the RDRF flag, reading RDR, and clearing the RDRF flag to 0. Also, before the MSB (bit 7) of the current frame is transmitted, read 1 from the TDRE flag to confirm that writing is possible. Then write data to TDR and clear the TDRE flag to 0.
However, the TDRE flag is checked and cleared automatically when the DTC is initiated by a transmit data empty interrupt (TXI) request and writes data to TDR. Similarly, the RDRF flag is cleared automatically when the DTC is initiated by a receive data full interrupt (RXI) and reads data from RDR.

Note: When switching from transmit or receive operation to simultaneous transmit and receive operations, first clear the TE bit and RE bit to 0, then set both these bits to 1 simultaneously.

Figure 13.20 Sample Flowchart of Simultaneous Serial Transmission and Reception

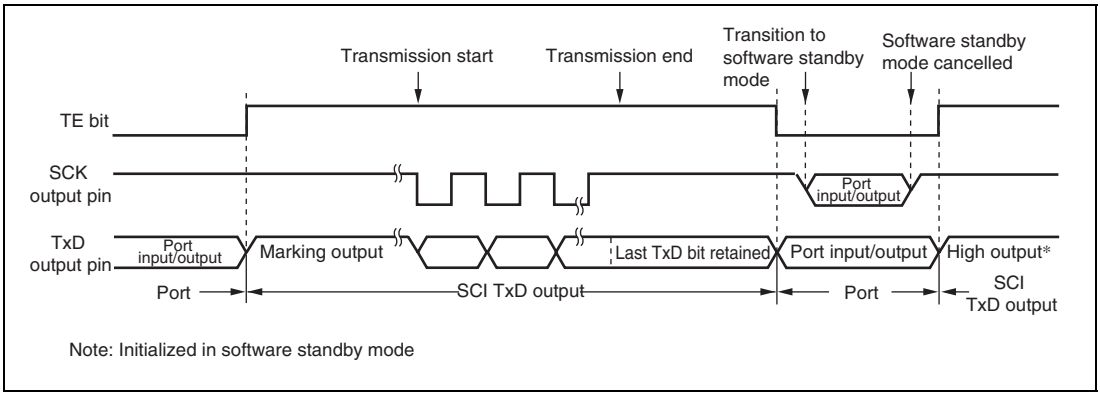


Figure 13.36 Pin States during Transmission in Clock Synchronous Mode (Internal Clock)

Reception: Before making the transition to module stop or software standby mode, stop reception (RE = 0). RSR, RDR, and SSR are reset. If transition is made during data reception, the data being received will be invalid.

To receive data in the same reception mode after mode cancellation, set RE to 1, and then start reception. To receive data in a different reception mode, initialize the SCI first.

Figure 13.37 shows a sample flowchart for mode transition during reception.

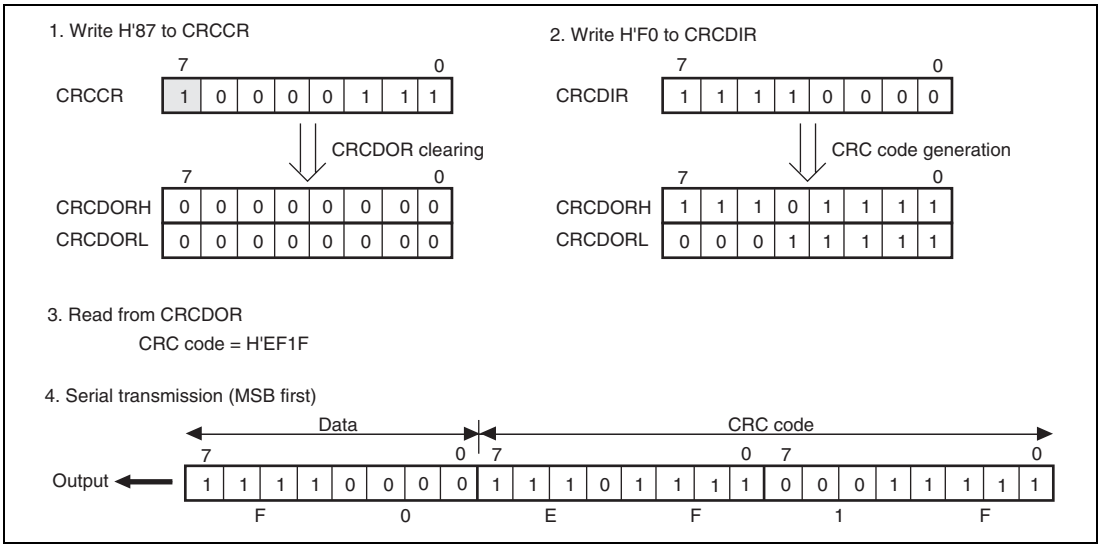


Figure 14.3 MSB-First Data Transmission

The following registers are necessary for SMIC mode

- SMIC flag register (SMICFLG)
- SMIC control/status register (SMICCSR)
- SMIC data register (SMICDTR)
- SMIC interrupt register 0 (SMICIR0)
- SMIC interrupt register 1 (SMICIR1)

The following registers are necessary for BT mode

- BT status register 0 (BTSR0)
- BT status register 1 (BTSR1)
- BT control/status register 0 (BTCSR0)
- BT control/status register 1 (BTCSR1)
- BT control register (BTCR)
- BT data buffer (BTDTR)
- BT interrupt mask register (BTIMSR)
- FIFO valid size register 0 (BTFVSR0)
- FIFO valid size register 1 (BTFVSR1)

Bit	Bit Name	Initial Value	R/W Slave	R/W Host	Description
0	OBF3A	0	R/(W)*	R	Output Data Register Full Indicates whether or not there is transmit data in ODR3. 0: There is not receive data in ODR3 [Clearing conditions] • When the host reads ODR3 in an I/O read cycle • When the slave writes 0 to bit OBF3A 1: There is receive data in ODR3 [Setting condition] • When the slave writes to ODR3

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

19.3.19 SCIF Address Register (SCIFADRH, SCIFADRL)

SCIFADR sets the host address for the SCIF. Do not change the contents of SCIFADR while the SCIF is operating (i.e. while SCIFE is set to 1).

- SCIFADRH

Bit	Bit Name	Initial Value	R/W		Description
			Slave	Host	
7	—	0	R/W	—	SCIF Address 15 to 8
6	—	0	R/W	—	These bits set the host address for the SCIF.
5	—	0	R/W	—	
4	—	0	R/W	—	
3	—	0	R/W	—	
2	—	0	R/W	—	
1	—	1	R/W	—	
0	—	1	R/W	—	

- SCIFADRL

Bit	Bit Name	Initial Value	R/W		Description
			Slave	Host	
7	—	1	R/W	—	SCIF Address 7 to 0
6	—	1	R/W	—	These bits set the host address for the SCIF.
5	—	1	R/W	—	
4	—	1	R/W	—	
3	—	1	R/W	—	
2	—	0	R/W	—	
1	—	0	R/W	—	
0	—	0	R/W	—	

Note: When the SCIF is in use, SCIFADR must be set to an address that is different from those for LPC channels 1, 2, and 3.

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

Table 19.10 Serialized Interrupt Transfer Cycle Frame Configuration

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

Table 19.13 HIRQ Setting and Clearing Conditions when SCIF Channels are Used

Host Interrupt	Setting Condition	Clearing Condition
SMI HIRQi (i = 1, 3 to 15)	The SCIF interrupt corresponding to the host interrupt request selected by SIRQCR3 occurs.	Relevant SCIF interrupt is cleared

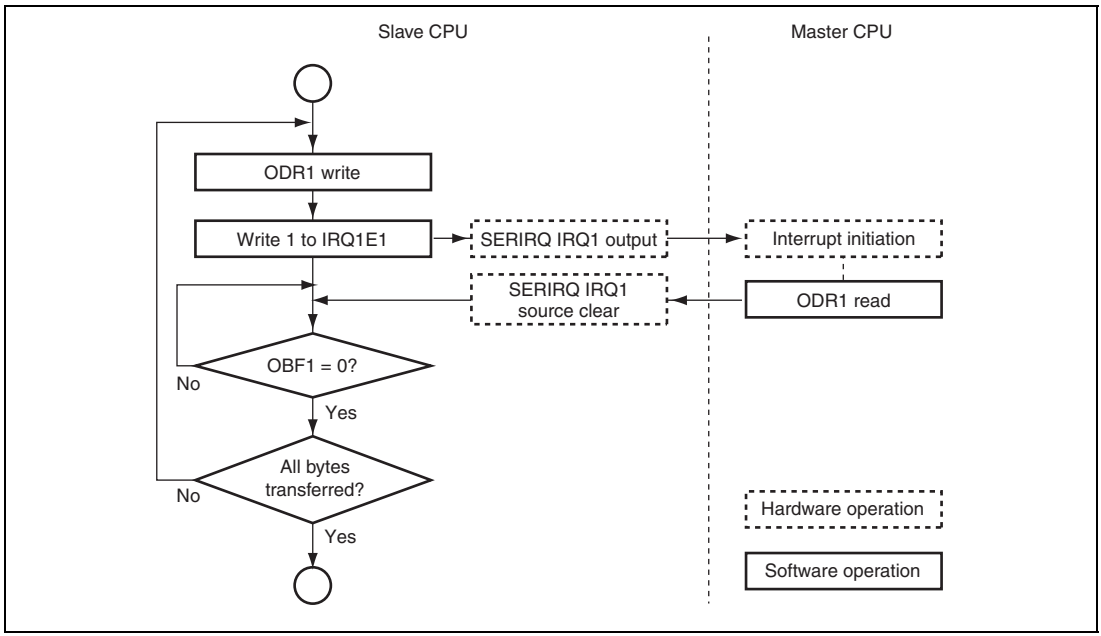


Figure 19.12 HIRQ Flowchart (Example of Channel 1)

Bit	Bit Name	Initial value	R/W	Description
15 to 3	—	All 0	—	Reserved These bits are always read as 0. The initial value should not be changed.
2	RFD2	1	R	Receive Byte Flow Control Threshold
1	RFD1	1	R	000: When (256 – 32) bytes of data is stored in the receive FIFO
0	RFD0	1	R	001: When (512 – 32) bytes of data is stored in the receive FIFO : : 110: When (1792 – 32) bytes of data is stored in the receive FIFO 001: When (2048 – 64) bytes of data is stored in the receive FIFO

21.2.18 Bit Rate Setting Register (ECBRR)

ECBRR sets the bit rate for retransmission and reception.

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RTM	0	R/W	Transmit/Receive Rate 0: 10 Mbps 1: 100 Mbps

21.2.19 Transmit Interrupt Register (TRIMD)

TRIMD is a 32-bit readable/writable register that specifies whether or not to notify write-back completion for each frame using the TWB bit in EESR and an interrupt on transmit operations.

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TIS	0	R/W	Transmit Interrupt Setting 0: Write-back completion for each frame is not notified 1: Write-back completion for each frame using the TWB bit in EESR is notified

(2) Bit-Rate-Adjustment State

The bit rate is calculated by measuring the period of transfer of a low-level byte (H'00) from the host. The bit rate can be changed by the command for a new bit rate selection. After the bit rate has been adjusted, the boot program enters the inquiry and selection state. The bit-rate-adjustment sequence is shown in figure 25.21.

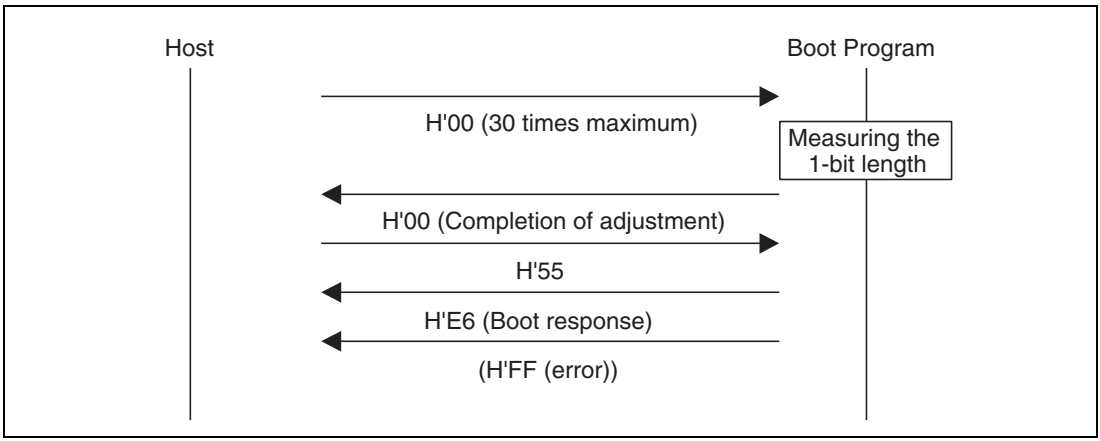


Figure 25.21 Bit-Rate-Adjustment Sequence

(3) Communications Protocol

After adjustment of the bit rate, the protocol for communications between the host and the boot program is as shown below.

1. 1-byte commands and 1-byte responses

   These commands and responses are comprised of a single byte. These are consists of the inquiries and the ACK for successful completion.
2. n-byte commands or n-byte responses

   These commands and responses are comprised of n bytes of data. These are selections and responses to inquiries.

   The amount of programming data is not included under this heading because it is determined in another command.
3. Error response

   The error response is a response to inquiries. It consists of an error response and an error code and comes two bytes.

Register Abbreviation	Reset	WDT Reset	High-Speed/ Medium-Speed	Sleep	Module Stop	Software Standby	Hardware Standby	Module
SMR_3	Initialized	Initialized	—	—	—	—	Initialized	SCI_3
BRR_3	Initialized	Initialized	—	—	—	—	Initialized	
SCR_3	Initialized	Initialized	—	—	—	—	Initialized	
TDR_3	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
SSR_3	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
RDR_3	Initialized	Initialized	—	—	Initialized	Initialized	Initialized	
SCMR_3	Initialized	Initialized	—	—	—	—	Initialized	
TCSR_1	Initialized	Initialized	—	—	—	—	Initialized	WDT_1
TCNT_1	Initialized	Initialized	—	—	—	—	Initialized	
TCR_X	Initialized	Initialized	—	—	—	—	Initialized	TMR_X
TCSR_X	Initialized	Initialized	—	—	—	—	Initialized	TMR_Y
TCNT_X	Initialized	Initialized	—	—	—	—	Initialized	
TCORA_X	Initialized	Initialized	—	—	—	—	Initialized	
TCORB_X	Initialized	Initialized	—	—	—	—	Initialized	
TCR_Y	Initialized	Initialized	—	—	—	—	Initialized	
TCSR_Y	Initialized	Initialized	—	—	—	—	Initialized	
TCORA_Y	Initialized	Initialized	—	—	—	—	Initialized	
TCORB_Y	Initialized	Initialized	—	—	—	—	Initialized	
TCNT_Y	Initialized	Initialized	—	—	—	—	Initialized	
TCONRS	Initialized	Initialized	—	—	—	—	Initialized	

Notes: 1. The registers related to USB are supported only by the H8S/2472 Group.
2. The registers related to PECI are supported only by the H8S/2472 Group and the H8S/2462 Group.