Renesas Electronics America Inc - DF71251AN50FPV Datasheet

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	SH-2
Core Size	32-Bit Single-Core
Speed	50MHz
Connectivity	SCI
Peripherals	POR, PWM, WDT
Number of I/O	37
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-20°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LFQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df71251an50fpv

SH7125 Group and SH7124 Group manuals:

Document Title	Document No.
SH7125 Group, SH7124 Group Hardware Manual	This manual
SH-1/SH-2/SH-DSP Software Manual	REJ09B0171

User's manuals for development tools:

Document Title	Document No.
SuperH™ RISC engine C/C++ Compiler, Assembler, Optimizing Linkage Editor Compiler Package V.9.00 User's Manual	REJ10B0152
SuperH™ RISC engine High-performance Embedded Workshop 3 User's Manual	REJ10B0025
SuperH RISC engine High-Performance Embedded Workshop 3 Tutorial	REJ10B0023

Application note:

Document Title	Document No.
SuperH RISC engine C/C++ Compiler Package Application Note	REJ05B0463

1.3 Pin Assignments

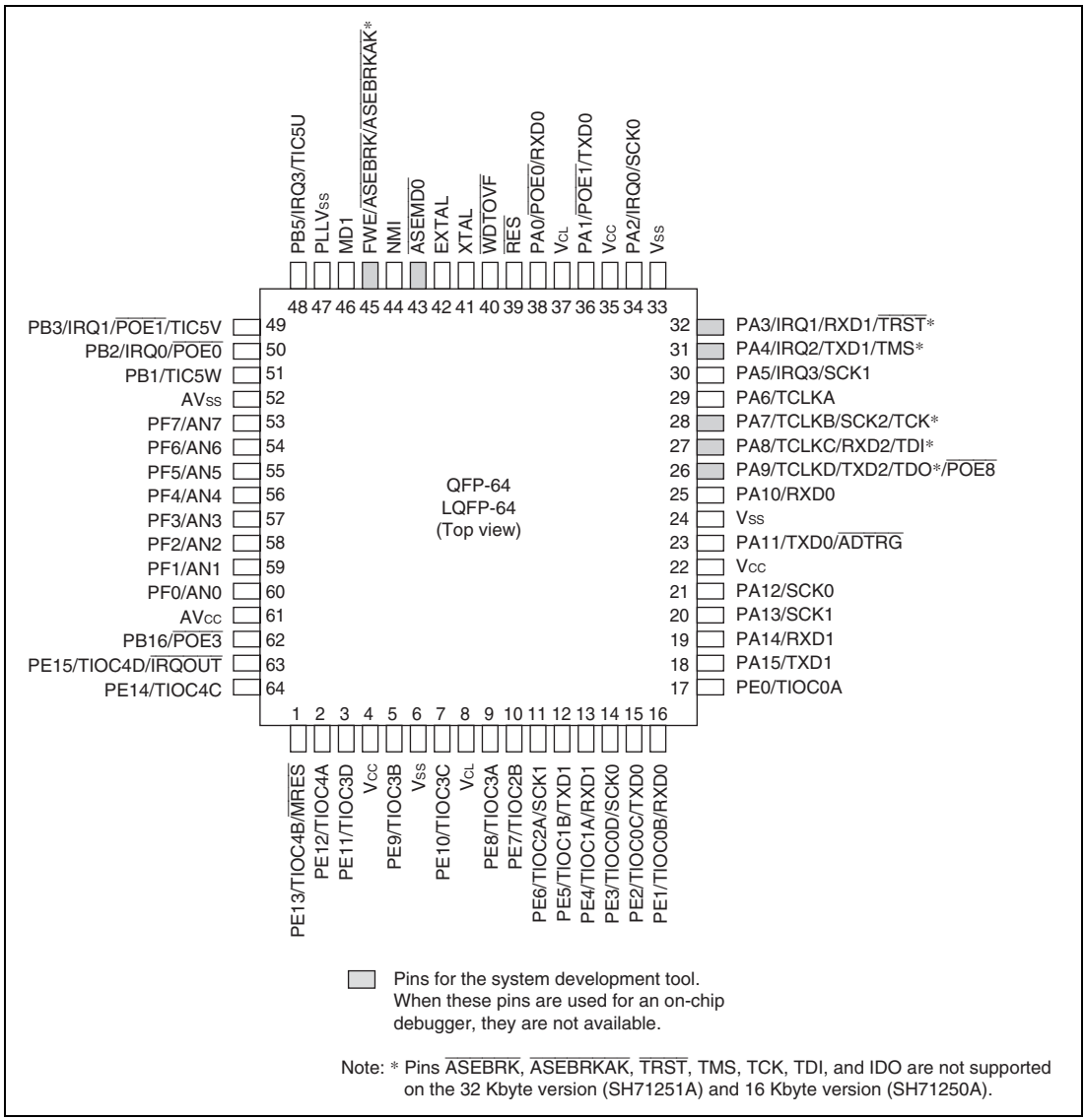


Figure 1.2 (1) Pin Assignments of SH7125

5.5.3 Illegal Slot Instructions

An instruction placed immediately after a delayed branch instruction is called "instruction placed in a delay slot". When the instruction placed in the delay slot is an undefined code, illegal slot exception handling starts after the undefined code is decoded. Illegal slot exception handling also starts when an instruction that changes the program counter (PC) value is placed in a delay slot and the instruction is decoded. The CPU handles an illegal slot instruction as follows:

1. The status register (SR) is saved to the stack.
2. The program counter (PC) is saved to the stack. The PC value saved is the target address of the delayed branch instruction immediately before the undefined code or the instruction that rewrites the PC.
3. The start address of the exception handling routine is fetched from the exception handling vector table that corresponds to the exception that occurred. Program execution branches to that address and the program starts. This branch is not a delayed branch.

5.5.4 General Illegal Instructions

When an undefined code placed anywhere other than immediately after a delayed branch instruction (i.e., in a delay slot) is decoded, general illegal instruction exception handling starts. The CPU handles the general illegal instructions in the same procedures as in the illegal slot instructions. Unlike processing of illegal slot instructions, however, the program counter value that is stacked is the start address of the undefined code.

Section 7 User Break Controller (UBC)

The user break controller (UBC) provides functions that simplify program debugging. These functions make it easy to design an effective self-monitoring debugger, enabling the chip to debug programs without using an in-circuit emulator. Break conditions that can be set in the UBC are instruction fetch or data read/write access, data size, data contents, address value, and stop timing in the case of instruction fetch.

7.1 Features

The UBC has the following features:

1. The following break comparison conditions can be set.
   Number of break channels: two channels (channels A and B)
   User break can be requested as either the independent or sequential condition on channels A and B (sequential break setting: channel A and then channel B match with break conditions, but not in the same bus cycle).

- Address
  Comparison bits are maskable in 1-bit units.
  One of the two address buses (L-bus address (LAB) and I-bus address (IAB)) can be selected.
- Data
  32-bit maskable.
  One of the two data buses (L-bus data (LDB) and I-bus data (IDB)) can be selected.
- Bus cycle
  Instruction fetch or data access
- Read/write
- Operand size
  Byte, word, and longword

2. A user-designed user-break condition interrupt exception processing routine can be run.
3. In an instruction fetch cycle, it can be selected that a user-break is set before or after an instruction is executed.
4. Maximum repeat times for the break condition (only for channel B): $2^{12} - 1$ times.
5. Four pairs of branch source/destination buffers.

Note: * The user break controller is not supported on the 32 Kbyte (SH71251A and SH71241A) and 16 Kbyte (SH71250A and SH71240A) versions.

Bit	Bit Name	Initial Value	R/W	Description
12	SCMFDB	0	R/W	I Bus Cycle Condition Match Flag B When the I bus cycle condition in the break conditions set for channel B is satisfied, this flag is set to 1. In order to clear this flag, write 0 into this bit. 0: The I bus cycle condition for channel B does not match 1: The I bus cycle condition for channel B matches
11	PCTE	0	R/W	PC Trace Enable 0: Disables PC trace 1: Enables PC trace
10	PCBA	0	R/W	PC Break Select A Selects the break timing of the instruction fetch cycle for channel A as before or after instruction execution. 0: PC break of channel A is set before instruction execution 1: PC break of channel A is set after instruction execution
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	DBEA	0	R/W	Data Break Enable A Selects whether or not the data bus condition is included in the break condition of channel A. 0: No data bus condition is included in the condition of channel A 1: The data bus condition is included in the condition of channel A
6	PCBB	0	R/W	PC Break Select B Selects the break timing of the instruction fetch cycle for channel B as before or after instruction execution. 0: PC break of channel B is set before instruction execution 1: PC break of channel B is set after instruction execution

Table 9.11 Setting of Operation Mode by Bits MD0 to MD3

Bit 3 MD3	Bit 2 MD2	Bit 1 MD1	Bit 0 MD0	Description
0	0	0	0	Normal operation
			1	Setting prohibited
		1	0	PWM mode 1
			1	PWM mode 2*[1]
	1	0	0	Phase counting mode 1*[2]
			1	Phase counting mode 2*[2]
		1	0	Phase counting mode 3*[2]
			1	Phase counting mode 4*[2]
1	0	0	0	Reset synchronous PWM mode*[3]
			1	Setting prohibited
		1	x	Setting prohibited
	1	0	0	Setting prohibited
			1	Complementary PWM mode 1 (transmit at crest)*[3]
		1	0	Complementary PWM mode 2 (transmit at trough)*[3]
			1	Complementary PWM mode 2 (transmit at crest and trough)*[3]

[Legend]

x: Don't care

Notes: 1. PWM mode 2 cannot be set for channels 3 and 4.

2. Phase counting mode cannot be set for channels 0, 3, and 4.

3. Reset synchronous PWM mode and complementary PWM mode can only be set for channel 3. When channel 3 is set to reset synchronous PWM mode or complementary PWM mode, the channel 4 settings become ineffective and automatically conform to the channel 3 settings. However, do not set channel 4 to reset synchronous PWM mode or complementary PWM mode. Reset synchronous PWM mode and complementary PWM mode cannot be set for channels 0, 1, and 2.

Table 9.15 TIOR_2 (Channel 2)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_2 Function**	**TIOC2B Pin Function*2**
0	0	0	0	Output compare register	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. The TIOC2B pin input/output function is supported only by the SH7125.

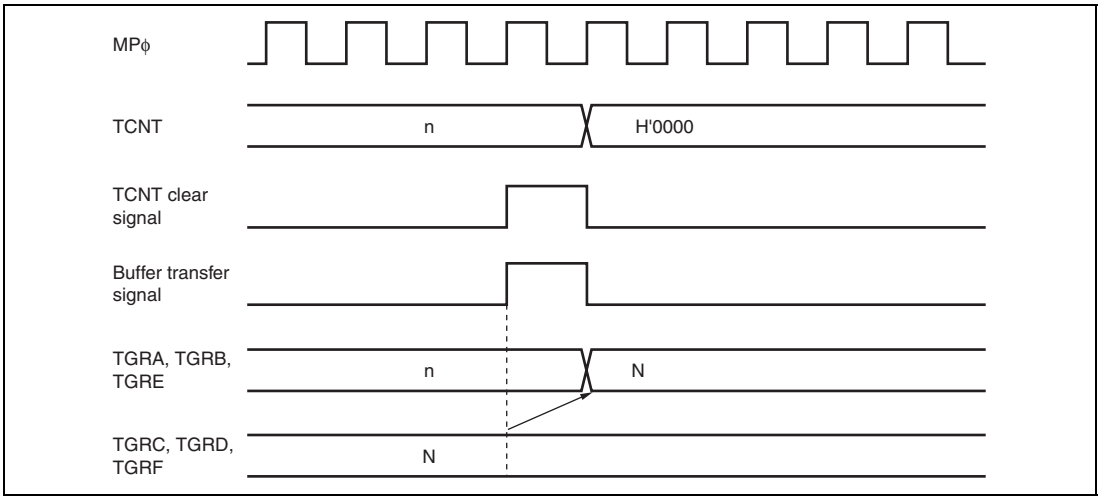


Figure 9.95 Buffer Transfer Timing (when TCNT Cleared)

Buffer Transfer Timing (Complementary PWM Mode): Figures 9.96 to 9.98 show the buffer transfer timing in complementary PWM mode.

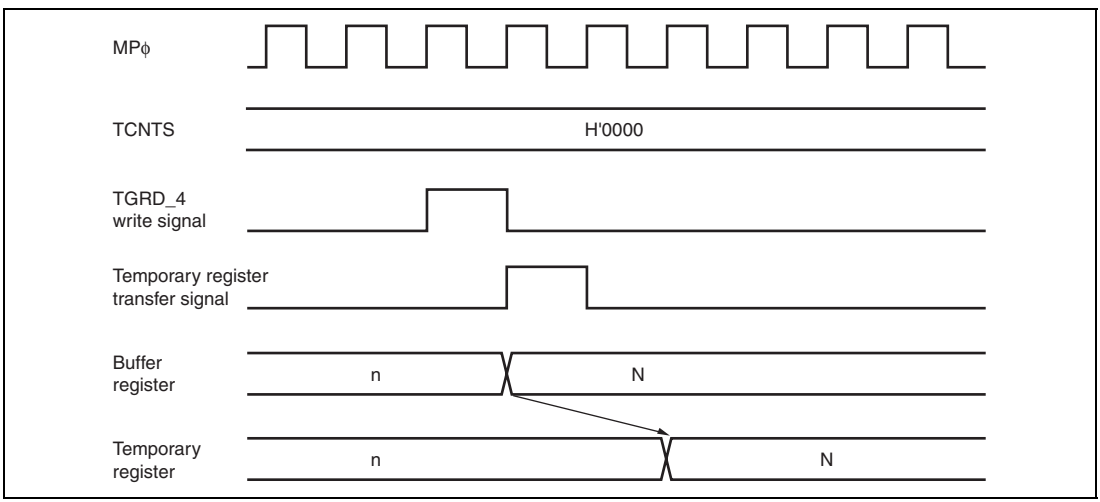


Figure 9.96 Transfer Timing from Buffer Register to Temporary Register (TCNTS Stop)

Section 11 Watchdog Timer (WDT)

This LSI includes the watchdog timer (WDT).

This LSI can be reset by the overflow of the counter when the value of the counter has not been updated because of a system runaway.

The watchdog timer (WDT) is a single-channel timer that uses a peripheral clock as an input and counts the clock settling time when clearing software standby mode. It can also be used as an interval timer.

11.1 Features

- Can be used to ensure the clock settling time: Use the WDT to revoke software standby mode.
- Can switch between watchdog timer mode and interval timer mode.
- Generates internal resets in watchdog timer mode: Internal resets occur after counter overflow.
- An interrupt is generated in interval timer mode
  An interval timer interrupt is generated when the counter overflows.
- Choice of eight counter input clocks
  Eight clocks (×1 to ×1/4096) that are obtained by dividing the peripheral clock can be chosen.
- Choice of two resets
  Power-on reset and manual reset are available.

(2) 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 the serial mode register (SCSMR) and bits CKE1 and CKE0 in the serial control register (SCSCR) (table 12.14).

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

When the SCI operates on an internal clock, it can output a clock signal at the SCK pin. The frequency of this output clock is equal to 16 times the desired bit rate.

(3) Transmitting and Receiving Data

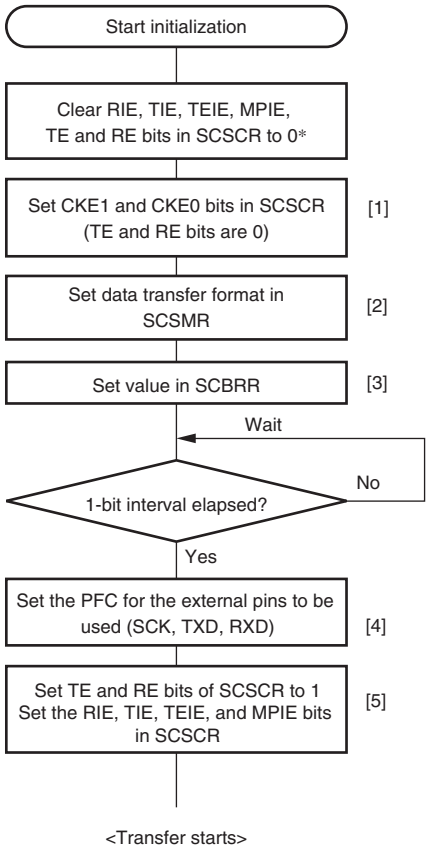
SCI Initialization (Asynchronous Mode):

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCSCR), then initialize the SCI as follows.

When changing the operation mode or the communication format, always clear the TE and RE bits to 0 before following the procedure given below. Clearing the TE bit to 0 sets the TDRE flag to 1 and initializes the transmit shift register (SCTSR). Clearing the RE bit to 0, however, does not initialize the RDRF, PER, FER, and ORER flags or receives data register (SCRDR), which retains their previous contents.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCI operation becomes unreliable if the clock is stopped.

Figure 12.9 shows a sample flowchart for initializing the SCI.



[1] Set the clock selection in SCSCR.

[2] Set the data transfer format in SCSMR.

[3] Write a value corresponding to the bit rate to SCBRR. Not necessary if an external clock is used.

[4] Set PFC of the external pin used. Set RXD input during receiving and TXD output during transmitting. Set SCK input/output according to contents set by CKE1 and CKE0.

[5] Set the TE bit or RE bit in SCR to 1.* Also make settings of the RIE, TIE, TEIE, and MPIE bits. At this time, the TXD, RXD, and SCK pins are ready to be used. The TXD pin is in a mark state during transmitting. When synchronous clock output (clock master) is set during receiving in clock synchronous mode, outputting clocks from the SCK pin starts.

Note: * In simultaneous transmit and receive operations, the TE and RE bits should both be cleared to 0 or set to 1 simultaneously.

Figure 12.9 Sample Flowchart for SCI Initialization

Receiving Serial Data (Clock Synchronous Mode): Figure 12.12 shows a sample flowchart for receiving serial data. Use the following procedure for serial data reception after enabling the SCIF for reception.

When switching from asynchronous mode to clock synchronous mode, make sure that the ORER, PER, and FER flags are all cleared to 0. If the FER or PER flag is set to 1, the RDRF flag will not be set and data reception cannot be started.

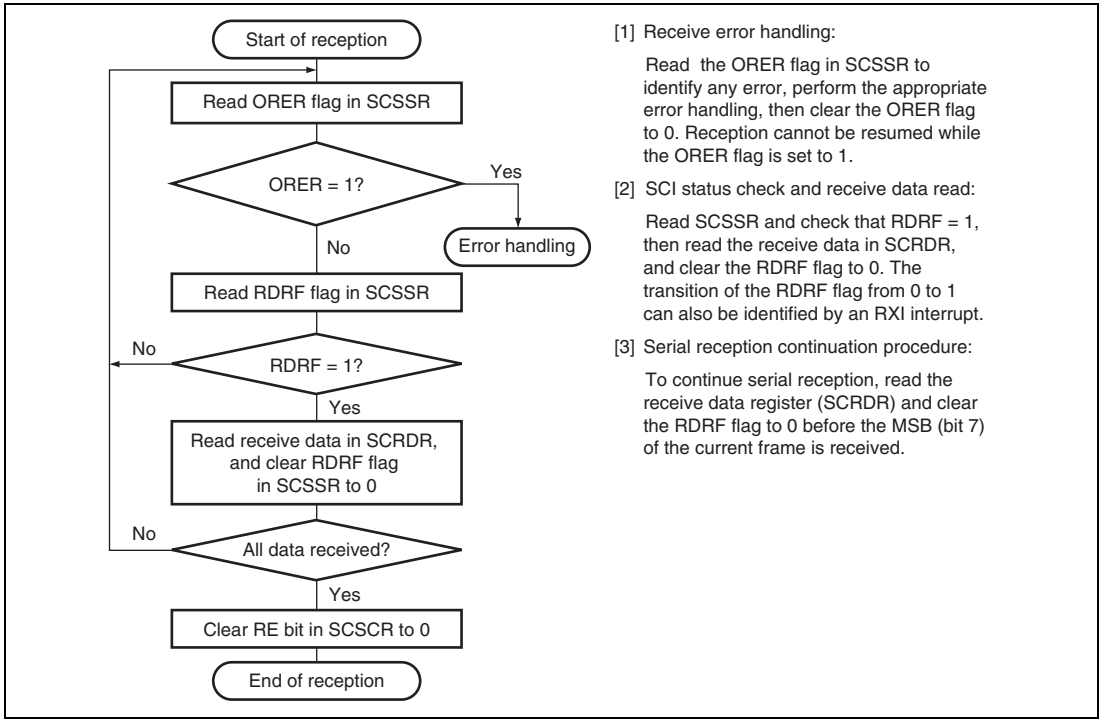


Figure 12.12 Sample Flowchart for Receiving Serial Data (1)

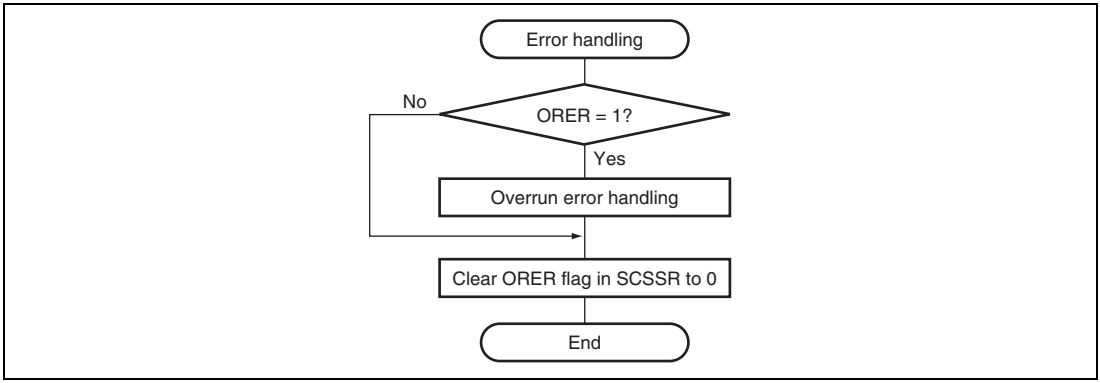


Figure 12.12 Sample Flowchart for Receiving Serial Data (2)

In receiving, the SCI operates as follows:

1. The SCI synchronizes with serial clock input or output and initializes internally.
2. Receive data is shifted into SCRSR in order from the LSB to the MSB. After receiving the data, the SCI checks whether the RDRF flag is 0 and the receive data can be transferred from SCRSR to SCRDR. If this check is passed, the SCI sets the RDRF flag to 1 and stores the received data in SCRDR. If a receive error is detected, the SCI operates as shown in table 12.16. In this state, subsequent reception cannot be continued. In addition, the RDRF flag will not be set to 1 after reception; be sure to clear the RDRF flag to 0.
3. After setting RDRF to 1, if the receive-data-full interrupt enable bit (RIE) is set to 1 in SCSCR, the SCI requests a receive-data-full interrupt (RXI). If the ORER bit is set to 1 and the RIE bit in SCSCR is also set to 1, the SCI requests a receive error interrupt (ERI).

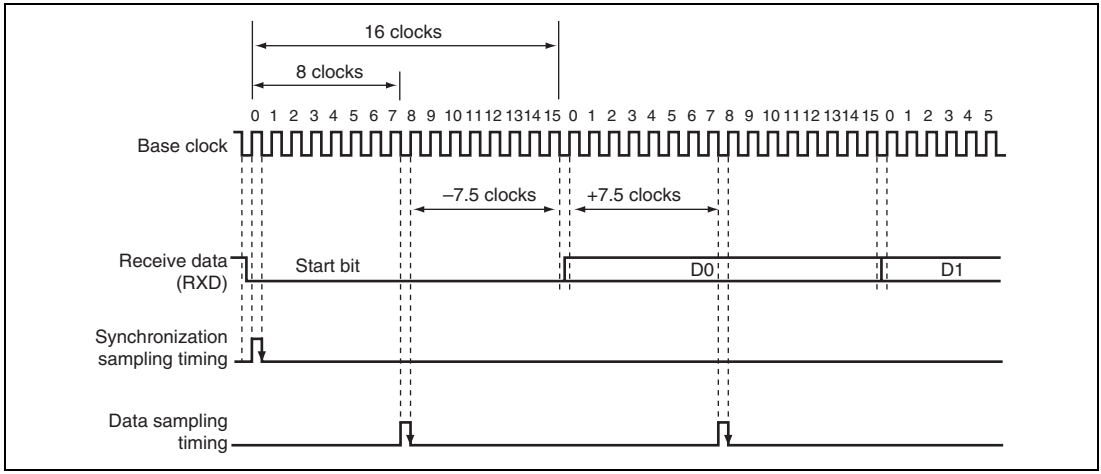


Figure 12.21 Receive Data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

Equation 1:

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)\,F - \frac{|D - 0.5|}{N}(1+F) \right| \times 100\ \%$$

Where: M: Receive margin (%)
N: Ratio of bit rate to clock (N = 16)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

Equation 2:

When D = 0.5 and F = 0:

$$M = (0.5 - 1/(2 \times 16)) \times 100\%$$
$$= 46.875\%$$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

Table 15.2 SH7124 Multiplexed Pins

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
A	PA0 I/O (port)	$\overline{\text{POE0}}$ input (POE)	RXD0 input (SCI)	—	—
	PA1 I/O (port)	$\overline{\text{POE1}}$ input (POE)	TXD0 output (SCI)	—	—
	PA3 I/O (port)	IRQ1 input (INTC)	RXD1 input (SCI)	$\overline{\text{TRST}}$ input (H-UDI)*2	—
	PA4 I/O (port)	IRQ2 input (INTC)	TXD1 output (SCI)	TMS input (H-UDI)*2	—
	PA6 I/O (port)	TCLKA input (MTU2)	—	—	—
	PA7 I/O (port)	TCLKB input (MTU2)	SCK2 I/O (SCI)	TCK input (H-UDI)*2	—
	PA8 I/O (port)	TCLKC input (MTU2)	RXD2 input (SCI)	TDI input (H-UDI)*2	—
	PA9 I/O (port)	TCLKD input (MTU2)	TXD2 output (SCI)	$\overline{\text{POE8}}$ input (POE)	TDO output (H-UDI)*2
B	PB1 I/O (port)	TIC5W input (MTU2)	—	—	—
	PB3 I/O (port)	IRQ1 input (INTC)	$\overline{\text{POE1}}$ input (POE)	TIC5V input (MTU2)	—
	PB5 I/O (port)	IRQ3 input (INTC)	TIC5U input (MTU2)	—	—
E	PE0 I/O (port)	TIOC0A I/O (MTU2)	—	—	—
	PE1 I/O (port)	TIOC0B I/O (MTU2)	RXD0 input (SCI)	—	—
	PE2 I/O (port)	TIOC0C I/O (MTU2)	TXD0 output (SCI)	—	—
	PE3 I/O (port)	TIOC0D I/O (MTU2)	SCK0 I/O (SCI)	—	—
	PE8 I/O (port)	TIOC3A I/O (MTU2)	—	—	—
	PE9 I/O (port)	TIOC3B I/O (MTU2)	—	—	—
	PE10 I/O (port)	TIOC3C I/O (MTU2)	—	—	—
	PE11 I/O (port)	TIOC3D I/O (MTU2)	—	—	—
	PE12 I/O (port)	TIOC4A I/O (MTU2)	—	—	—
	PE13 I/O (port)	TIOC4B I/O (MTU2)	$\overline{\text{MRES}}$ input (INTC)	—	—
	PE14 I/O (port)	TIOC4C I/O (MTU2)	—	—	—
	PE15 I/O (port)	TIOC4D I/O (MTU2)	$\overline{\text{IRQOUT}}$ output (INTC)	—	—

- Port E Control Register L1 (PECRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE3 MD2	PE3 MD1	PE3 MD0	-	PE2 MD2	PE2 MD1	PE2 MD0	-	PE1 MD2	PE1 MD1	PE1 MD0	-	-	PE0 MD1	PE0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PE3MD2	0	R/W	PE3 Mode
13	PE3MD1	0	R/W	Select the function of the PE3/TIOC0D/SCK0 pin.
12	PE3MD0	0	R/W	000: PE3 I/O (port) 001: TIOC0D I/O (MTU2) 110: SCK0 I/O (SCI) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE2MD2	0	R/W	PE2 Mode
9	PE2MD1	0	R/W	Select the function of the PE2/TIOC0C/TXD0 pin.
8	PE2MD0	0	R/W	000: PE2 I/O (port) 001: TIOC0C I/O (MTU2) 110: TXD0 output (SCI) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PE1MD2	0	R/W	PE1 Mode
5	PE1MD1	0	R/W	Select the function of the PE1/TIOC0B/RXD0 pin.
4	PE1MD0	0	R/W	000: PE1 I/O (port) 001: TIOC0B I/O (MTU2) 110: RXD0 input (SCI) Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MOA31 to MOA0	Undefined	R/W	MOA31 to MOA0 Store the start address of the programming destination on the user MAT. The consecutive 128-byte programming is executed starting from the specified start address of the user MAT. The MOA6 to MOA0 bits are always 0 because the start address of the programming destination is at the 128-byte boundary.

(3.2) Flash multipurpose data destination area parameter (FMPDR: general register R4 of CPU)

This parameter indicates the start address in the area which stores the data to be programmed in the user MAT. When the storage destination of the program data is in flash memory, an error occurs. The error occurrence is indicated by the WD bit (bit 2) in FPFR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MOD31	MOD30	MOD29	MOD28	MOD27	MOD26	MOD25	MOD24	MOD23	MOD22	MOD21	MOD20	MOD19	MOD18	MOD17	MOD16
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
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:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MOD15	MOD14	MOD13	MOD12	MOD11	MOD10	MOD9	MOD8	MOD7	MOD6	MOD5	MOD4	MOD3	MOD2	MOD1	MOD0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
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 0	MOD31 to MOD0	Undefined	R/W	MOD31 to MOD0 Store the start address of the area which stores the program data for the user MAT. The consecutive 128-byte data is programmed to the user MAT starting from the specified start address.

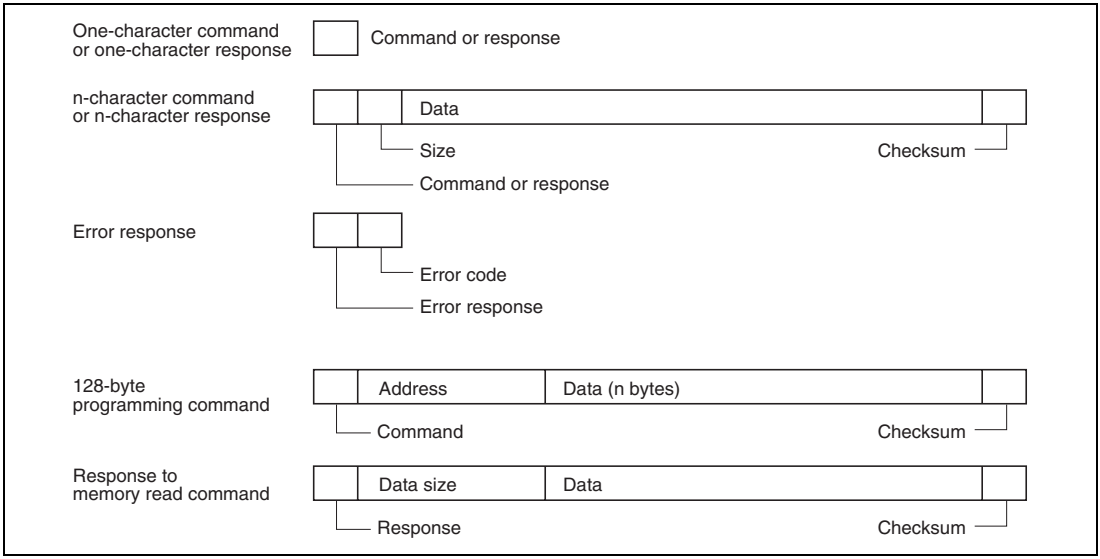
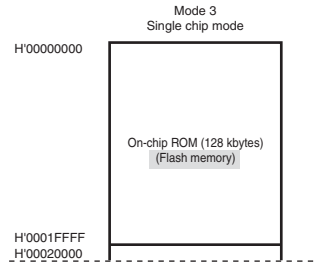
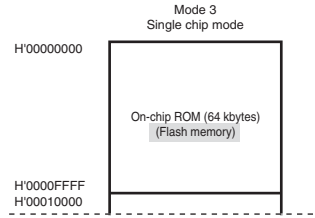
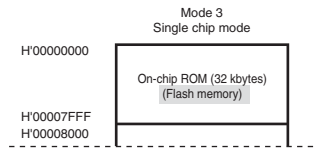


Figure 17.17 Formats in the Communications Protocol

— Command (1 byte): Inquiry, selection, programming, erasure, checking, etc.
— Response (1 byte): Response to an inquiry
— Size (one or two bytes): The length of data for transfer, excluding the command/response code, size, and checksum.
— Data (n bytes): Particular data for the command or response
— Checksum (1 byte): Set so that the total sum of byte values from the command code to the checksum and change lower one byte to H'00.
— Error response (1 byte): Error response to a command
— Error code (1 byte): Indicates the type of error.
— Address (4 bytes): Address for programming
— Data (n bytes): Data to be programmed. "n" is known from the response to the command used to inquire about the programming size.
— Data size (4 bytes): Four-byte field included in the response to a memory reading command.

Register Name	Abbreviation	Number of Bits	Address	Module	Access Size	Number of Access States
Timer control register V_5	TCRV_5	8	H'FFFFC494	MTU2	8	MPϕ reference
Timer I/O control register V_5	TIORV_5	8	H'FFFFC496		8	B: 2, W: 2, L: 4
Timer counter W_5	TCNTW_5	16	H'FFFFC4A0		16, 32	
Timer general register W_5	TGRW_5	16	H'FFFFC4A2		16	
Timer control register W_5	TCRW_5	8	H'FFFFC4A4		8	
Timer I/O control register W_5	TIORW_5	8	H'FFFFC4A6		8	
Timer status register_5	TSR_5	8	H'FFFFC4B0		8	
Timer interrupt enable register_5	TIER_5	8	H'FFFFC4B2		8	
Timer start register_5	TSTR_5	8	H'FFFFC4B4		8	
Timer compare match clear register	TCNTCMPCLR	8	H'FFFFC4B6		8	
A/D data register 0	ADDR0	16	H'FFFFC900	A/D	16	Pϕ reference
A/D data register 1	ADDR1	16	H'FFFFC902	(Channel 0)	16	B: 2, W: 2
A/D data register 2	ADDR2	16	H'FFFFC904		16	
A/D data register 3	ADDR3	16	H'FFFFC906		16	
A/D control/status register_0	ADCSR_0	16	H'FFFFC910		16	
A/D control register_0	ADCR_0	16	H'FFFFC912		16	
A/D data register 4	ADDR4	16	H'FFFFC980	A/D	16	Pϕ reference
A/D data register 5	ADDR5	16	H'FFFFC982	(Channel 1)	16	B: 2, W: 2
A/D data register 6	ADDR6	16	H'FFFFC984		16	
A/D data register 7	ADDR7	16	H'FFFFC986		16	
A/D control/status register_1	ADCSR_1	16	H'FFFFC990		16	
A/D control register_1	ADCR_1	16	H'FFFFC992		16	
Flash code control/status register	FCCS	8	H'FFFFCC00	FLASH	8	Pϕ reference
Flash program code select register	FPCS	8	H'FFFFCC01		8	B: 5
Flash erase code select register	FECS	8	H'FFFFCC02		8	
Flash key code register	FKEY	8	H'FFFFCC04		8	
Flash transfer destination address register	FTDAR	8	H'FFFFCC06		8	

Item	Page	Revision (See Manual for Details)
3.1 Selection of Operating Modes Table 3.1 Selection of Operating Modes	49	Table amended Pin Setting / On-Chip ROM Mode No. / FWE / MD1 / Mode Name / (Flash memory) Note added Note: 2. Prohibited in SH71251A, SH71250A, SH71241A and SH71240A.
3.4 Address Map Figure 3.1 Address Map in SH7125, SH7124 (128 Kbytes Flash Memory Version)	51	Figure amended 
Figure 3.2 Address Map in SH7125, SH7124 (64 Kbytes Flash Memory Version)	52	Figure amended 
Figure 3.3 Address Map in SH71251A and SH71241A (32 Kbytes Flash Memory Version)	53	Figure amended 
Figure 3.4 Address Map in SH71250A and SH71240A (16 Kbytes Flash Memory Version)	54	Figure added