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	SH2A-FPU
Core Size	32-Bit Single-Core
Speed	100MHz
Connectivity	CANbus, SCI, SPI, UART/USART
Peripherals	DMA, POR, PWM, WDT
Number of I/O	85
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	120-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/r5f72394bdfp-v0

Section 1 Overview

1.1 Features

SH7239 Group and SH7237 Group are single-chip RISC (Reduced Instruction Set Computer) microprocessors that integrate a Renesas original RISC CPU core with peripheral functions required for system configuration.

The CPU in the SH7239 and SH7237 Groups has a RISC-type instruction set and uses a superscalar architecture and a Harvard architecture, which greatly improves instruction execution speed. In addition, the 32-bit internal-bus architecture enhances data processing power. With this CPU, it has become possible to assemble low-cost, high-performance, and high-functioning systems, even for applications that were previously impossible with microprocessors, such as realtime control, which demands high speeds. The SH7239 Group also includes the floating-point unit (FPU).

In addition, the SH7239 and SH7237 Groups include on-chip peripheral functions necessary for system configuration, such as a large-capacity ROM, a ROM cache, a RAM, a direct memory access controller (DMAC), a data transfer controller (DTC), multi-function timer pulse units 2 (MTU2 and MTU2S), a serial communication interface with FIFO (SCIF), a serial communication interface (SCI), a Renesas serial peripheral interface (RSPI), an A/D converter, an interrupt controller (INTC), I/O ports, a controller area network (RCAN-ET), and data flash (FLD).

The SH7239 and SH7237 Groups also provide an external memory access support function to enable direct connection to various memory devices or peripheral LSIs.

These on-chip functions significantly reduce costs of designing and manufacturing application systems.

The features of this LSI are listed in table 1.1.

4.6 Oscillator

The source of click supply can be selected from a connected crystal resonator or an external clock input through a pin.

4.6.1 Connecting Crystal Resonator

A crystal resonator can be connected as shown in figure 4.2. Use the damping resistance (R_d) shown in table 4.6. Use a crystal resonator that has a resonance frequency of 10 to 12.5 MHz.

It is recommended to consult the crystal resonator manufacturer concerning the compatibility of the crystal resonator and the LSI.

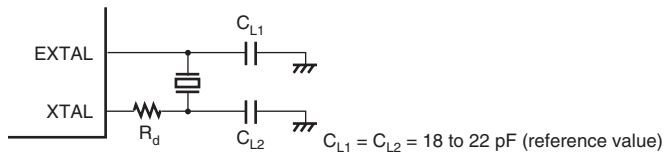


Figure 4.2 Example of Crystal Resonator Connection

Table 4.6 Damping Resistance Values (Reference Values)

Frequency (MHz)	10	12.5
R_d (Ω) (reference value)	0	0

Figure 4.3 shows an equivalent circuit of the crystal resonator. Use a crystal resonator with the characteristics shown in table 4.7.

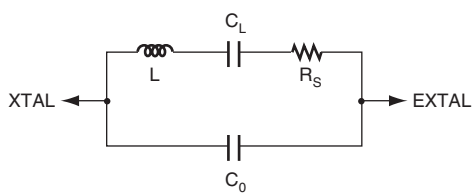


Figure 4.3 Crystal Resonator Equivalent Circuit

5.4 Register Bank Errors

5.4.1 Register Bank Error Sources

(1) Bank Overflow

In the state where saving has already been performed to all register bank areas, bank overflow occurs when acceptance of register bank overflow exception has been set by the interrupt controller (the BOVE bit in IBNR of the INTC is set to 1) and an interrupt that uses a register bank has occurred and been accepted by the CPU.

(2) Bank Underflow

Bank underflow occurs when an attempt is made to execute a RESBANK instruction while saving has not been performed to register banks.

5.4.2 Register Bank Error Exception Handling

When a register bank error occurs, register bank error exception handling starts. The CPU operates as follows:

1. The exception service routine start address which corresponds to the register bank error that occurred is fetched from the exception handling vector table.
2. The status register (SR) is saved to the stack.
3. The program counter (PC) is saved to the stack. The PC value saved is the start address of the instruction to be executed after the last executed instruction for a bank overflow, and the start address of the executed RESBANK instruction for a bank underflow.

   To prevent multiple interrupts from occurring at a bank overflow, the interrupt priority level that caused the bank overflow is written to the interrupt mask level bits (I3 to I0) of the status register (SR).
4. After jumping to the address fetched from the exception handling vector table, program execution starts. The jump that occurs is not a delayed branch.

Table 11.24 TIORH_3 (Channel 3)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_3 Function**	**TIOC3A Pin Function**
0	0	0	0	Output compare register	Output retained*
			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

Note: * After power-on reset, 0 is output until TIOR is set.

11.3.15 Timer Start Register (TSTR)

TSTR is an 8-bit readable/writable register that selects operation/stoppage of TCNT for channels 0 to 4.

TSTR_5 is an 8-bit readable/writable register that selects operation/stoppage of TCNTU_5, TCNTV_5, and TCNTW_5 for channel 5.

When setting the operating mode in TMDR or setting the count clock in TCR, first stop the TCNT counter.

- TSTR

Bit:	7	6	5	4	3	2	1	0
	CST4	CST3	-	-	-	CST2	CST1	CST0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CST4	0	R/W	Counter Start 4 and 3
6	CST3	0	R/W	These bits select operation or stoppage for TCNT. If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: TCNT_4 and TCNT_3 count operation is stopped 1: TCNT_4 and TCNT_3 performs count operation
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Figure 11.3 shows an example of the PWM output level setting procedure in buffer operation.

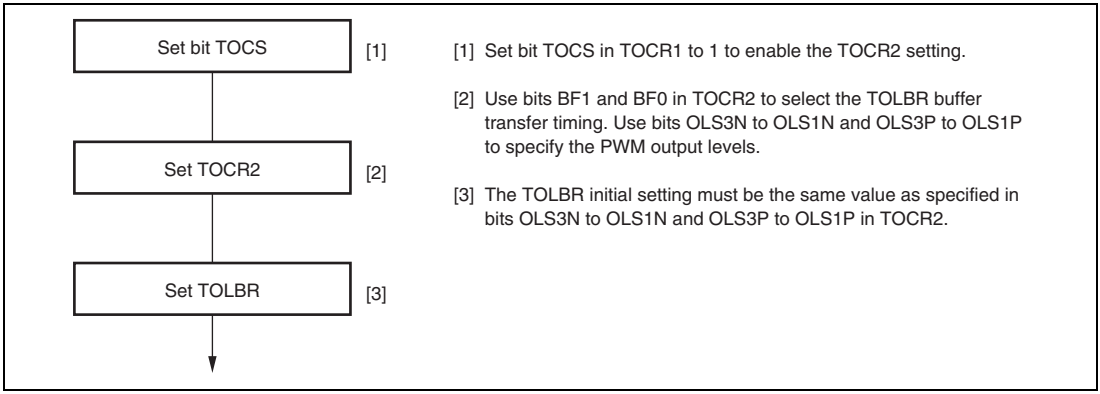


Figure 11.3 PWM Output Level Setting Procedure in Buffer Operation

11.3.23 Timer Gate Control Register (TGCR)

TGCR is an 8-bit readable/writable register that controls the waveform output necessary for brushless DC motor control in reset-synchronized PWM mode/complementary PWM mode. These register settings are ineffective for anything other than complementary PWM mode/reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	-	BDC	N	P	FB	WF	VF	UF
Initial value:	1	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6	BDC	0	R/W	Brushless DC Motor This bit selects whether to make the functions of this register (TGCR) effective or ineffective. 0: Ordinary output 1: Functions of this register are made effective

11.4.7 Reset-Synchronized PWM Mode

In reset-synchronized PWM mode, three-phase output of positive and negative PWM waveforms that share a common wave transition point can be obtained by combining channels 3 and 4.

When set for reset-synchronized PWM mode, the TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, and TIOC4D pins function as PWM output pins and TCNT3 functions as an upcounter.

Table 11.52 shows the PWM output pins used. Table 11.53 shows the settings of the registers.

Table 11.52 Output Pins for Reset-Synchronized PWM Mode

Channel	Output Pin	Description
3	TIOC3B	PWM output pin 1
	TIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
4	TIOC4A	PWM output pin 2
	TIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	TIOC4B	PWM output pin 3
	TIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)

Table 11.53 Register Settings for Reset-Synchronized PWM Mode

Register	Description of Setting
TCNT_3	Initial setting of H'0000
TCNT_4	Initial setting of H'0000
TGRA_3	Set count cycle for TCNT_3
TGRB_3	Sets the turning point for PWM waveform output by the TIOC3B and TIOC3D pins
TGRA_4	Sets the turning point for PWM waveform output by the TIOC4A and TIOC4C pins
TGRB_4	Sets the turning point for PWM waveform output by the TIOC4B and TIOC4D pins

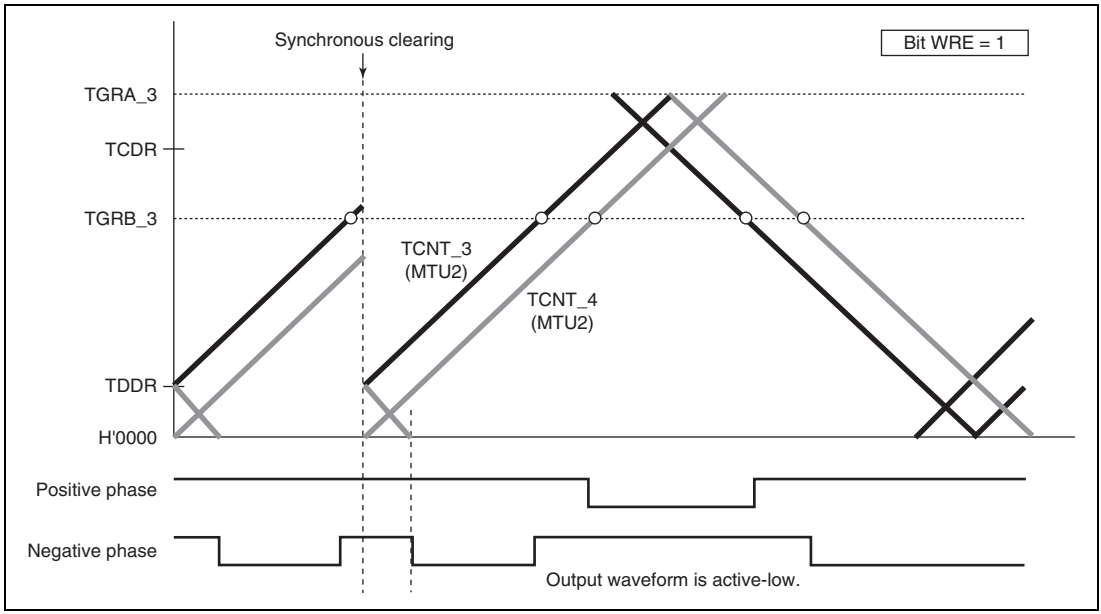


Figure 11.58 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 11.56; Bit WRE of TWCR in MTU2 is 1)

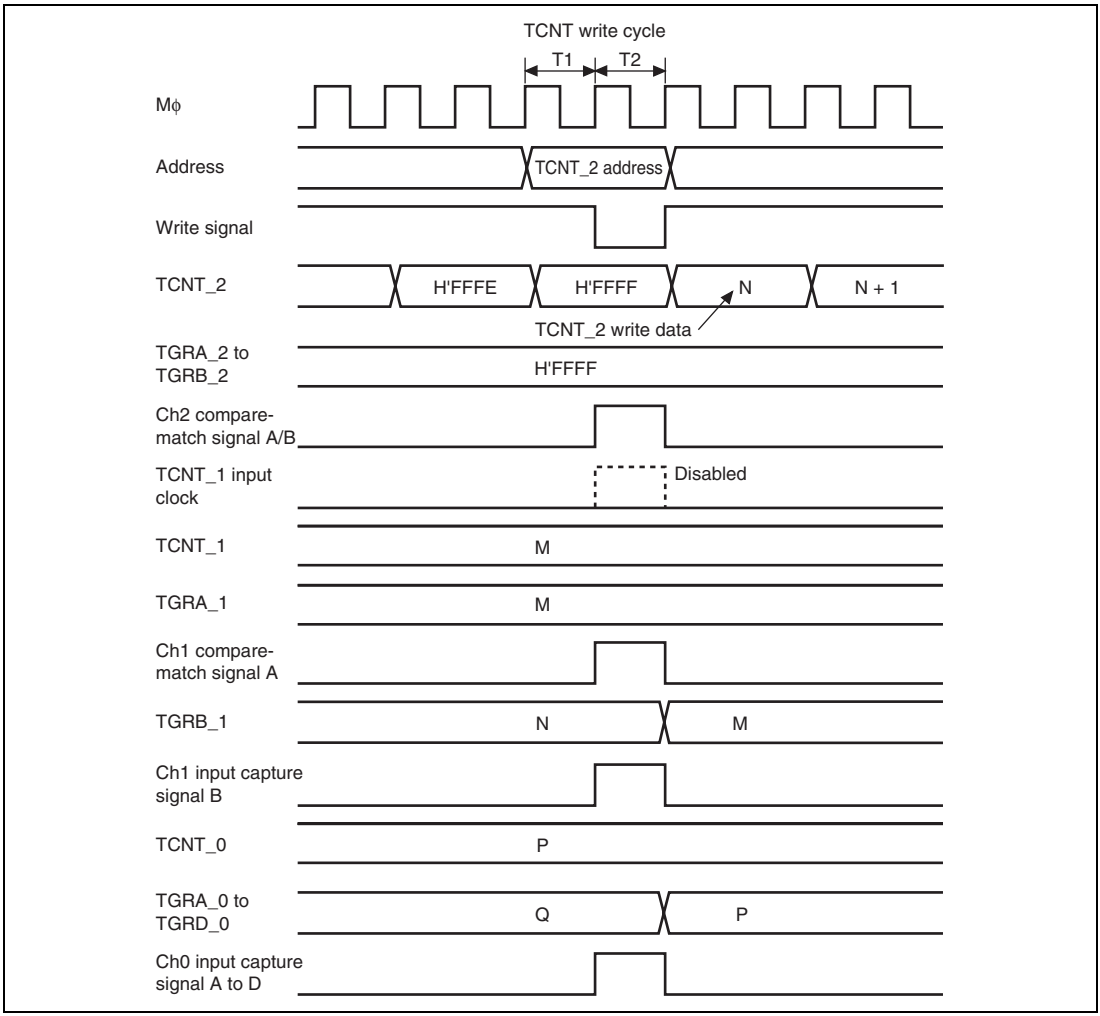


Figure 11.131 TCNT_2 Write and Overflow/Underflow Contention with Cascade Connection

Register Name	Abbreviation	R/W	Initial value	Address	Access Size
Timer counter U_5S	TCNTU_5S	R/W	H'0000	H'FFFE4880	16, 32
Timer general register U_5S	TGRU_5S	R/W	H'FFFF	H'FFFE4882	16
Timer control register U_5S	TCRU_5S	R/W	H'00	H'FFFE4884	8
Timer I/O control register U_5S	TIORU_5S	R/W	H'00	H'FFFE4886	8
Timer counter V_5S	TCNTV_5S	R/W	H'0000	H'FFFE4890	16, 32
Timer general register V_5S	TGRV_5S	R/W	H'FFFF	H'FFFE4892	16
Timer control register V_5S	TCRV_5S	R/W	H'00	H'FFFE4894	8
Timer I/O control register V_5S	TIORV_5S	R/W	H'00	H'FFFE4896	8
Timer counter W_5S	TCNTW_5S	R/W	H'0000	H'FFFE48A0	16, 32
Timer general register W_5S	TGRW_5S	R/W	H'FFFF	H'FFFE48A2	16
Timer control register W_5S	TCRW_5S	R/W	H'00	H'FFFE48A4	8
Timer I/O control register W_5S	TIORW_5S	R/W	H'00	H'FFFE48A6	8
Timer status register_5S	TSR_5S	R/W	H'00	H'FFFE48B0	8
Timer interrupt enable register_5S	TIER_5S	R/W	H'00	H'FFFE48B2	8
Timer start register_5S	TSTR_5S	R/W	H'00	H'FFFE48B4	8
Timer compare match clear register S	TCNTCMPCLRS	R/W	H'00	H'FFFE48B6	8

Note: * For details on the above registers, see section 11.3.9, Timer Synchronous Clear Register S (TSYCRS) and figure 11.85, Example of Procedure for Specifying MTU2S Counter Clearing by MTU2 Flag Setting Source in section 11, Multi-Function Timer Pulse Unit 2 (MTU2).

Figure 13.1 shows a block diagram of the POE2.

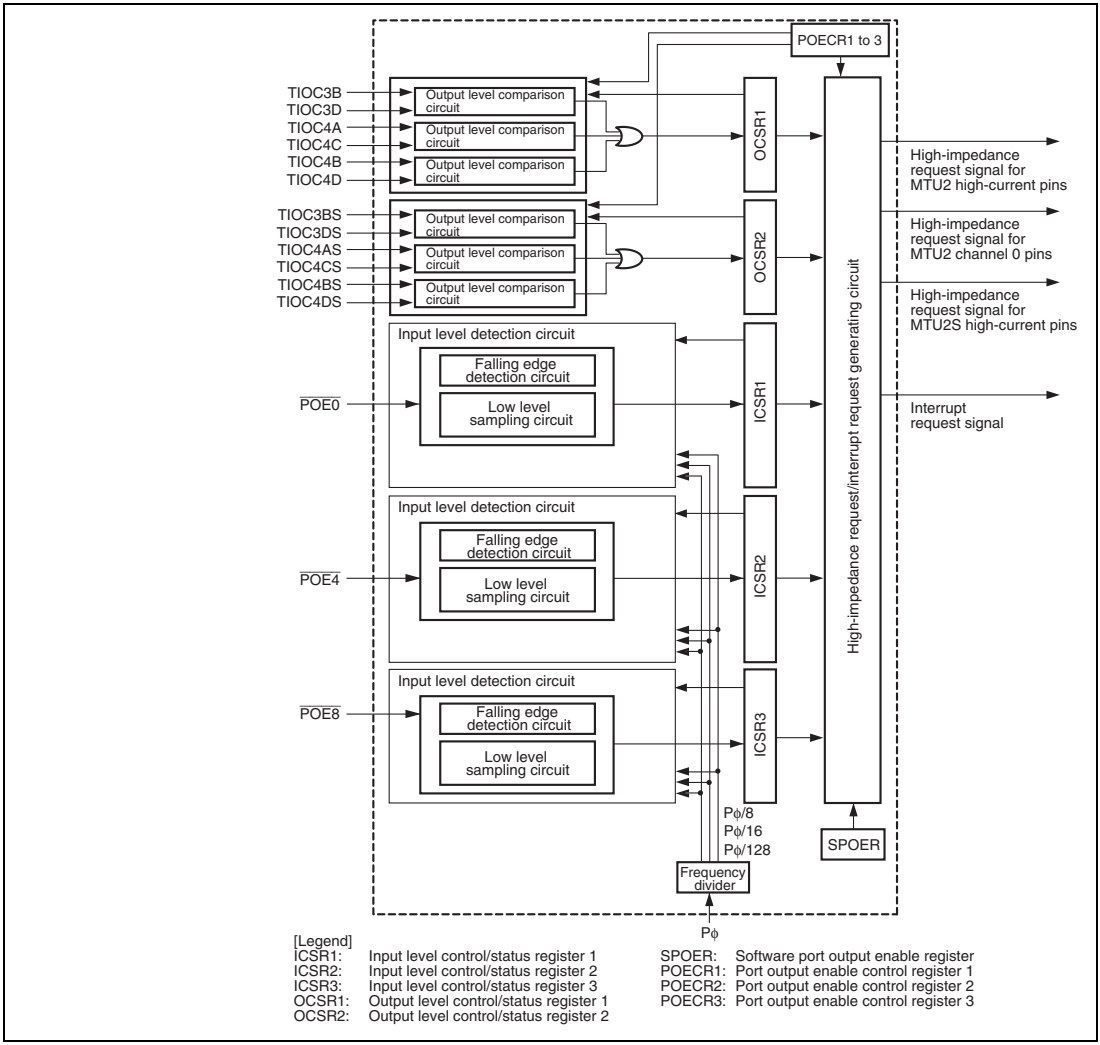


Figure 13.1 Block Diagram of POE2

17.4.3 Operation in Clocked Synchronous Mode

In clocked synchronous mode, the SCIF transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

The SCIF transmitter and receiver are independent, so full-duplex communication is possible while sharing the same clock. The transmitter and receiver are also 16-byte FIFO buffered, so continuous transmitting or receiving is possible by reading or writing data while transmitting or receiving is in progress.

Figure 17.9 shows the general format in clocked synchronous serial communication.

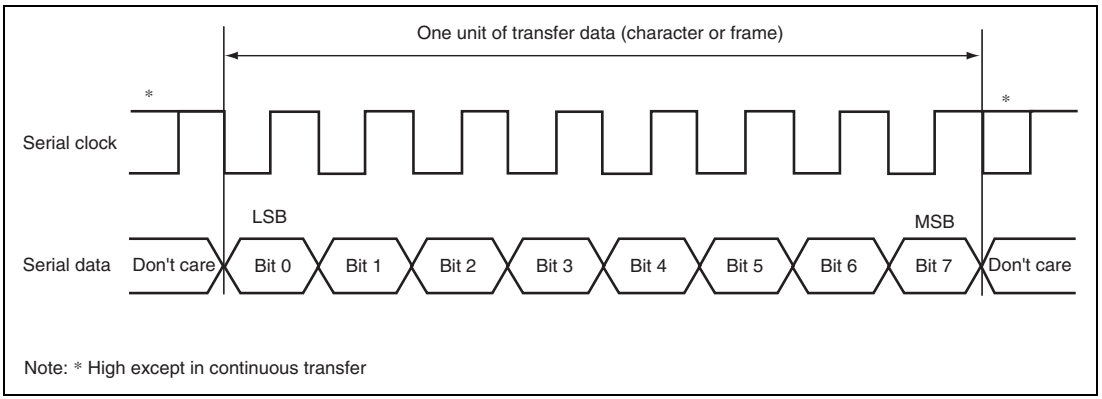


Figure 17.9 Data Format in Clocked Synchronous Communication

In clocked synchronous serial communication, each data bit is output on the communication line from one falling edge of the serial clock to the next. Data is guaranteed valid at the rising edge of the serial clock.

In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB.

In clocked synchronous mode, the SCIF receives data by synchronizing with the rising edge of the serial clock.

18.3.3 RSPI Pin Control Register (SPPCR)

SPPCR sets the modes of the RSPI pins. SPPCR can be read from or written to by the CPU. If the contents of this register are changed by the CPU while the RSPI function is enabled by setting the SPE bit in the RSPI control register (SPCR) to 1, operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	–	–	MOIFE	MOIFV	–	SPOM	–	SPLP
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5	MOIFE	0	R/W	MOSI Idle Value Fixing Enable Fixes the MOSI output value when the RSPI in master mode is in an SSL negation period (including the SSL retention period during a burst transfer). When MOIFE is 0, the RSPI outputs the last data from the previous serial transfer during the SSL negation period. When MOIFE is 1, the RSPI outputs the fixed value set in the MOIFV bit to the MOSI bit. 0: MOSI output value equals final data from previous transfer 1: MOSI output value equals the value set in the MOIFV bit
4	MOIFV	0	R/W	MOSI Idle Fixed Value If the MOIFE bit is 1 in master mode, the RSPI, according to MOIFV bit settings, determines the MOSI signal value during the SSL negation period (including the SSL retention period during a burst transfer). 0: MOSI Idle fixed value equals 0 1: MOSI Idle fixed value equals 1
3	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

18.4.3 RSPI System Configuration Example

(1) Single Master/Single Slave (with This LSI Acting as Master)

Figure 18.2 shows a single-master/single-slave RSPI system configuration example when this LSI is used as a master. In the single-master/single-slave configuration, the SSL0 to SSL3 outputs of this LSI (master) are not used. The SSL input of the RSPI slave is fixed to 0, and the RSPI slave is always maintained in a select state. In the transfer format corresponding to the case where the CPHA bit in the RSPI control register (SPCR) is 0, there are slave devices for which the SSL signal cannot be fixed to the active level. In situations where the SSL signal cannot be fixed, the SSL output of this LSI should be connected to the SSL input of the slave device.

This LSI (master) always drives the RSPCK and MOSI signals. The RSPI slave always drives the MISO signal.

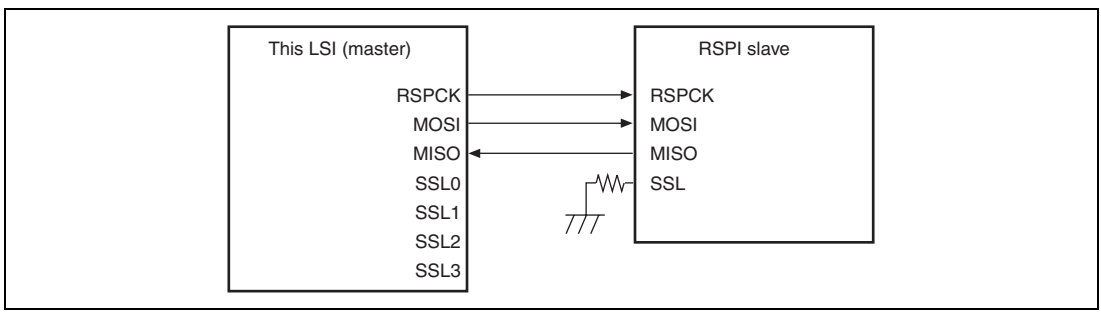


Figure 18.2 Single-Master/Single-Slave Configuration Example (This LSI = Master)

(2-4) Burst Transfer

If the SSLKP bit in the RSPI command register (SPCMD) that the RSPI references during the current serial transfer is 1, the RSPI keeps the SSL signal level during the serial transfer until the beginning of the SSL signal assertion for the next serial transfer. If the SSL signal level for the next serial transfer is the same as the SSL signal level for the current serial transfer, the RSPI can execute continuous serial transfers while keeping the SSL signal assertion status (burst transfer).

Figure 18.22 shows an example of an SSL signal operation for the case where a burst transfer is implemented using SPCMD0 and SPCMD1 settings. The text below explains the RSPI operations (1) to (7) as depicted in figure 18.22. It should be noted that the polarity of the SSL output signal depends on the settings in the RSPI slave select polarity register (SSLP).

1. Based on SPCMD0, the RSPI asserts the SSL signal and inserts RSPCK delays.
2. The RSPI executes serial transfers according to SPCMD0.
3. The RSPI inserts SSL negation delays.
4. Because the SSLKP bit in SPCMD0 is 1, the RSPI keeps the SSL signal value on SPCMD0. This period is sustained for next-access delay of SPCMD0 + 2 Pϕ at a minimum. If the shift register is empty after the passage of a minimum period, this period is sustained until such time as the transmit data is stored in the shift register for another transfer.
5. Based on SPCMD1, the RSPI asserts the SSL signal and inserts RSPCK delays.
6. The RSPI executes serial transfers according to SPCMD1.
7. Because the SSLKP bit in SPCMD1 is 0, the RSPI negates the SSL signal. In addition, a next-access delay is inserted according to SPCMD1.

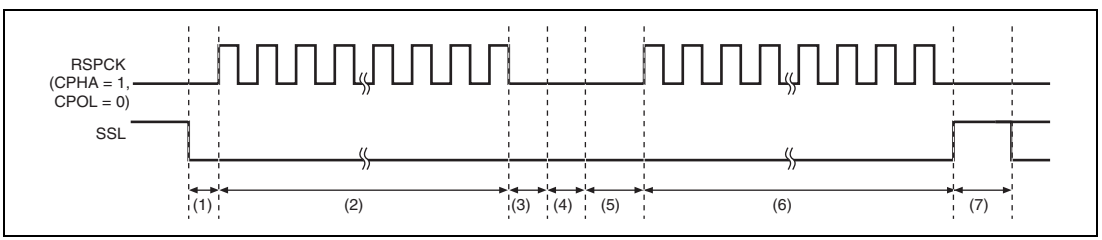


Figure 18.22 Example of Burst Transfer Operation using SSLKP Bit

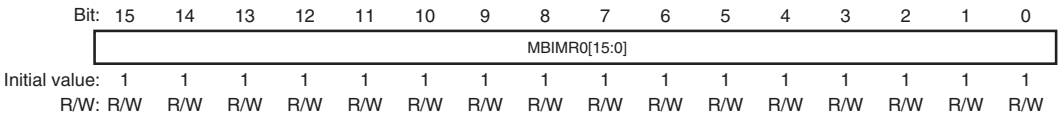
If the SSL signal settings in the SPCMD in which 1 is assigned to the SSLKP bit are different from the SSL signal output settings in the SPCMD to be used in the next transfer, the RSPI switches the SSL signal status to SSL signal assertion ((5) in figure 18.22) corresponding to the command for the next transfer. Notice that if such an SSL signal switching occurs, the slaves that drive the MISO signal compete, and the possibility arises of the collision of signal levels.

(7) Mailbox Interrupt Mask Register (MBIMR)

The MBIMR1 and MBIMR0 are 16-bit read / write registers. The MBIMR only prevents the setting of IRR related to the Mailbox activities, that are IRR[1] – Data Frame Received Interrupt, IRR[2] – Remote Frame Request Interrupt, IRR[8] – Mailbox Empty Interrupt, and IRR[9] – Message OverRun/OverWrite Interrupt. If a mailbox is configured as receive, a mask at the corresponding bit position prevents the generation of a receive interrupt (IRR[1] and IRR[2] and IRR[9]) but does not prevent the setting of the corresponding bit in the RXPR or RFPR or UMSR. Similarly when a mailbox has been configured for transmission, a mask prevents the generation of an Interrupt signal and setting of an Mailbox Empty Interrupt due to successful transmission or abortion of transmission (IRR[8]), however, it does not prevent the RCAN-ET from clearing the corresponding TXPR/TXCR bit + setting the TXACK bit for successful transmission, and it does not prevent the RCAN-ET from clearing the corresponding TXPR/TXCR bit + setting the ABACK bit for abortion of the transmission.

A mask is set by writing a '1' to the corresponding bit position for the mailbox activity to be masked. At reset all mailbox interrupts are masked.

- MBIMR0



Bit 15 to 0 — Enable or disable interrupt requests from individual Mailbox-15 to Mailbox-0 respectively.

Bit[15:0]: MBIMR0	Description
0	Interrupt Request from IRR1/IRR2/IRR8/IRR9 enabled
1	Interrupt Request from IRR1/IRR2/IRR8/IRR9 disabled (initial value)

21.1.11 Port D Control Registers L1 to L4 (PDCRL1 to PDCRL4)

PDCRL1 to PDCRL4 are 16-bit readable/writable registers that are used to select the functions of the multiplexed pins on port D.

(1) Port D Control Register L4 (PDCRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD15MD[2:0]			-	PD14MD[2:0]			-	PD13MD[2:0]			-	PD12MD[2:0]		
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/W	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 to 12	PD15MD[2:0]	000	R/W	PD15 Mode 000: PD15 I/O (port) 001: D15 I/O (BSC)* 010: Setting prohibited 011: Setting prohibited 100: Setting prohibited 101: TIOC4DS I/O (MTU2S) 110: Setting prohibited 111: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

22.2.3 Port B Port Registers H and L (PBPRH and PBPRL)

PBPRH and PBPRL are 16-bit read-only registers, which return the states of the pins. However, note that the pin state cannot be read when PB3 is set to the SCIF function and the TE bit in SCSCR and SPB2IO bit in SCSPTR are 0.

(1) Port B Port Register H (PBPRH)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PB21 PR	PB20 PR	PB19 PR	PB18 PR	PB17 PR	PB16 PR
Initial value:	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	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PB21PR	Pin state	R	The pin state is returned. These bits cannot be modified.
4	PB20PR	Pin state	R	
3	PB19PR	Pin state	R	
2	PB18PR	Pin state	R	
1	PB17PR	Pin state	R	
0	PB16PR	Pin state	R	

Module Name	Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
ROM/FLD	FPMON	FWE	—	—	—	—	—	—	—
	FMODR	—	—	—	FRDMD	—	—	—	—
	FASTAT	ROMAE	—	—	CMDLK	EEPAE	EEPIFE	EEPRPE	EEPWPE
	FAEINT	ROMAEIE	—	—	CMDLKIE	EEPAEIE	EEPIFEIE	EEPRPEIE	EEPWPEIE
	ROMMAT	KEY							
		—	—	—	—	—	—	—	ROMSEL
	FCURAME	KEY							
		—	—	—	—	—	—	—	FCRME
	FSTATR0	FRDY	ILGLERR	ERSERR	PRGERR	SUSRDY	—	ERSSPD	PRGSPD
	FSTATR1	FCUERR	—	—	FLOCKST	—	—	—	—
	FENTRYR	FKEY							
		FENTRYD	—	—	—	—	—	—	FENTRY0
	FPROTR	FPKEY							
		—	—	—	—	—	—	—	FPROTCN
	FRESETR	FPKEY							
		—	—	—	—	—	—	—	FRESET
	FCMDR	CMDR							
		PCMDR							
	FCPSR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ESUSPMD
	EEPBCCNT	—	—	—	BCADR				
		BCADR					—	—	BCSIZE
	FPESTAT	—	—	—	—	—	—	—	—
		PEERRST							
	EEPBCSTAT	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	BCST
	PCKAR	—	—	—	—	—	—	—	—
		PCKA							
	EEPRE0	KEY							
		DBRE07	DBRE06	DBRE05	DBRE04	DBRE03	DBRE02	DBRE01	DBRE00
	EEPRE1	KEY							
		DBRE15	DBRE14	DBRE13	DBRE12	DBRE11	DBRE10	DBRE09	DBRE08
	EEPWE0	KEY							
		DBWE07	DBWE06	DBWE05	DBWE04	DBWE03	DBWE02	DBWE01	DBWE00
	EEPWE1	KEY							
		DBWE15	DBWE14	DBWE13	DBWE12	DBWE03	DBWE02	DBWE01	DBWE00

29.3.14 I/O Port Timing

Table 29.23 I/O Port Timing

Item	Symbol	Min.	Max.	Unit	Figure
Output data delay time	t_{PORTD}	—	50	ns	Figure 29.33
Input data setup time	t_{PORTS}	20	—		
Input data hold time	t_{PORTH}	20	—		

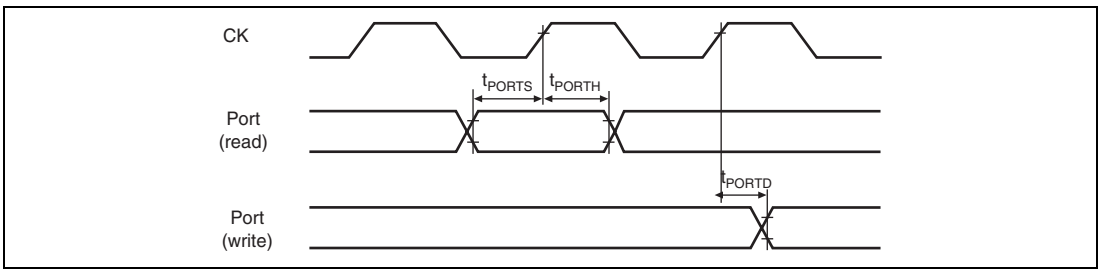


Figure 29.33 I/O Port Timing