To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

H8SX Family

Using a Timer-Generated Clock Signal to Drive SCI Transmission and Reception: DMAC Volume

Introduction

Compare-match output from 8-bit TMR unit 2 is selectable as the clock source for asynchronous mode transfer on serial communications interfaces 5 and 6 (SCI_5, 6) of the H8SX/1653. In this sample task, data are transmitted and received at 375 kbps when a peripheral clock signal (Pϕ) running at 16 MHz is input to the timer clock.

Target Device

H8SX/1638, H8SX/1648, H8SX/1653, H8SX/1658R, H8SX/1663, H8SX/1668R Groups

Preface

Although the writing of this application note is in accord with the hardware manual for the H8SX/1653 Group, the program covered in this application note can be run on the target devices indicated above. However, since some functional modules may be changed for the addition of functionality etc., be sure to perform a thorough evaluation by confirming the details with the hardware manual for the target device.

Contents

1. Specifications 2
2. Applicable Conditions 3
3. Description of Modules Used 4
4. Description of Operation 12
5. Description of Software 17
6. Note on Usage 42

1. Specifications

Compare-match output of 8-bit TMR unit 2 is selectable as the clock source for asynchronous mode transfer over SCI_5 and SCI_6 of the H8SX/1653. In this sample task, the timer compare-match output is selected as the base clock for SCI_5, and data are transmitted and received at 375 kbps with Pϕ running at 16 MHz.

- Connect the H8SX/1653 as shown in figure 1.
- Table 1 shows the communications format.
- After a power-on reset of the master side, pin P13 on the same side outputs a low-level trigger, and the master side starts operations for the simultaneous reception and transmission of 128-byte blocks of data.
- When the low-level trigger is input to the $\overline{\text{IRQ3}}$ pin on the slave side, the slave side starts operations for the simultaneous transmission and reception of 128-byte blocks of data.
- In this sample task, interrupt-driven DMAC asynchronously handles transmission and reception of the 128-byte blocks.

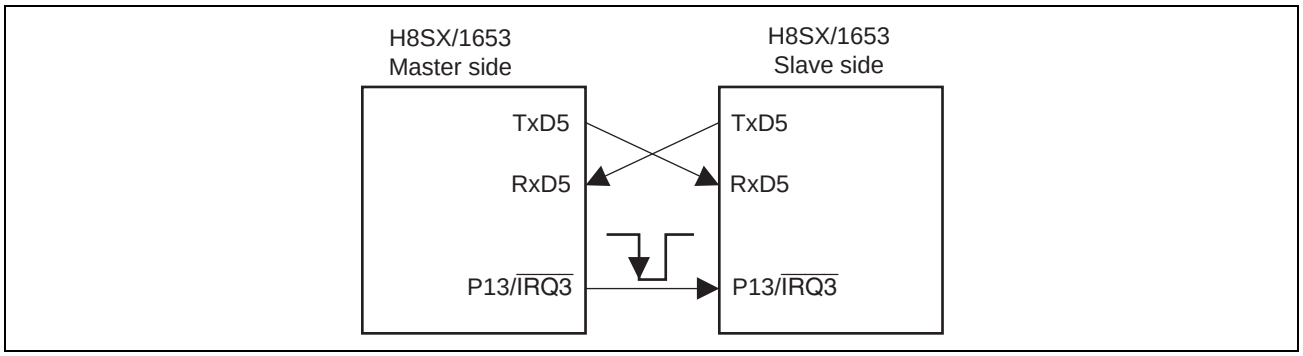


Figure 1 Setup for Asynchronous Communications with Timing from the Timer Clock Input

Table 1 Asynchronous Serial Transmission and Reception Format

Format	Setting
Pϕ	16 MHz
Serial communications mode	Asynchronous
Clock source	Timer compare-match output
Transfer rate	375 kbps
Data length	8 bits
Parity bit	None
Stop bit	1 bit
Serial/parallel conversion format	LSB first

2. Applicable Conditions

Table 2 Applicable Conditions

Item	Setting	
Operating frequency	Input clock:	16 MHz
	System clock:	16 MHz
	Peripheral module clock (Pϕ):	16 MHz
	External bus clock (Bϕ):	16 MHz
Operating mode	Mode 6 (MD2 = 1, MD1 = 1, MD0 = 0) MD_CLK = 0	
Development tool	High-performance Embedded Workshop Ver. 4.00.02	
C/C++ compiler	From Renesas Technology Corp. H8S, H8/300 Series C/C++ Compiler Ver. 6.01.00	
Compiler options	-cpu = h8sxa:24:md, -code = machinecode, -optimize = 1, -regparam = 3 -speed = (register, shift, struct, expression)	

Table 3 Section Settings

Address	Section Name	Description
H'001000	P	Program area
	C	Data table storage
H'FF2000	B	Non-initialized data area (RAM area)

3. Description of Modules Used

3.1 Description in Outline

Peripheral modules of the H8SX/1653 which are used in this sample task are shown in Figure 2. The following description concerns the blocks shown in Figure 2.

1. SCI_5
   Transmits and receives data with timing provided by the input from the timer clock.
   a. During SCI transmission
      - When TSR_5 is not full, data for transmission are written to TDR_5, transferred to TSR_5, and then output on the TxD5 pin.
      - When the data are transferred from TDR_5 to TSR_5, a TXI_5 interrupt is generated.
   b. During SCI reception
      - After one frame of data has been received via the RxD5 pin, the received data are transferred from RSR_5 to RDR_5.
      - Once the data have been successfully received and then transferred from RSR_5 to RDR_5, a received data full (RXI_5) interrupt is generated.
2. TMR unit 2
   TMR_4 and TMR_5 of the TMR unit 2 are set up to generate SCK5, the internal base clock for SCI_5.
3. DMAC channels 0 and 1
   a. During SCI transmission
      - Channel 0 is activated by the transmission data empty interrupt (TXI_5) from SCI_5 and transfers data from the area where data for transmission are stored to the TDR_5 register.
   b. During SCI reception
      - Channel 1 is activated by the received data full interrupt (RXI_5) of SCI_5 and transfers data from RDR_5 to the area where received data is to be stored.

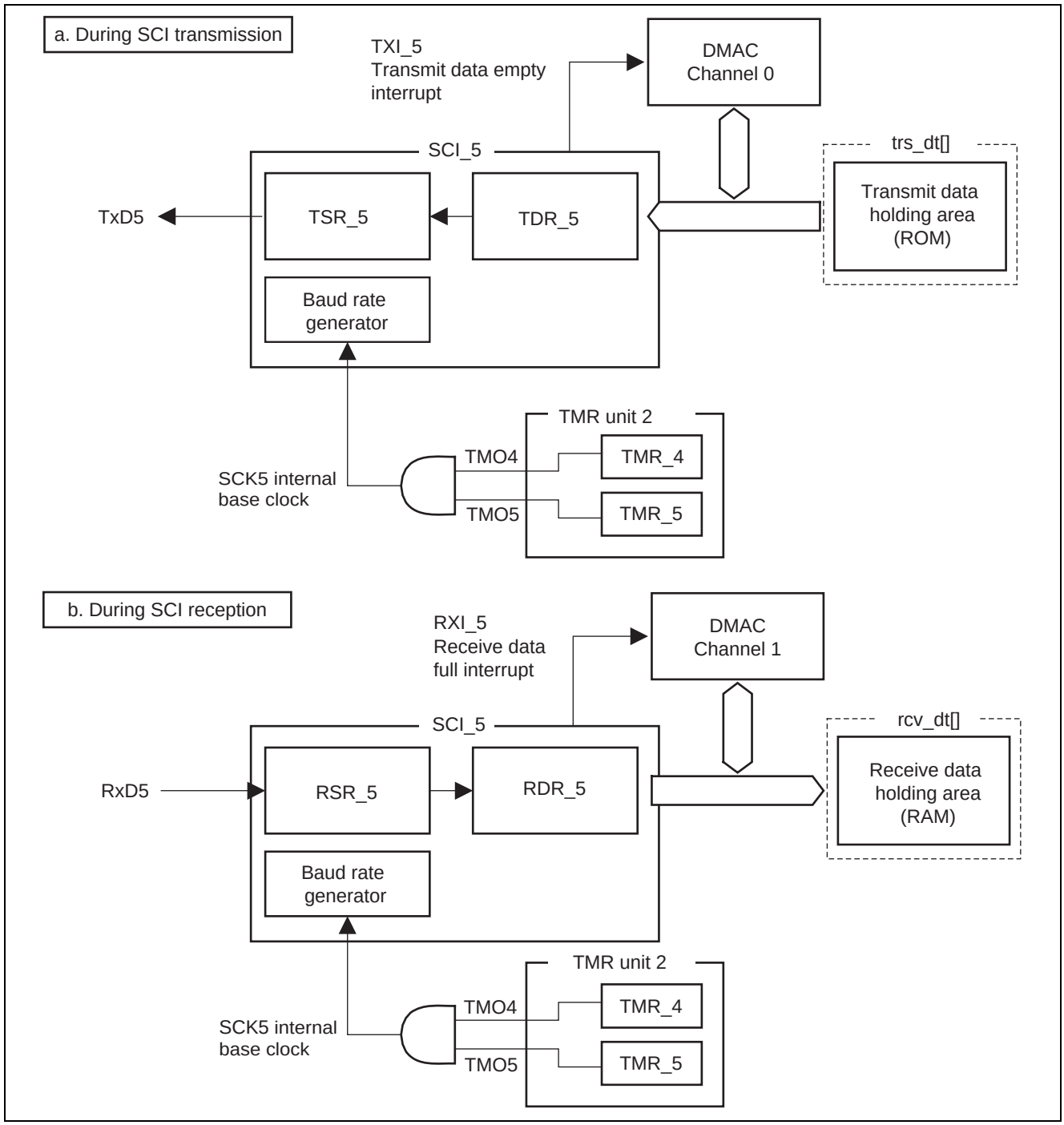


Figure 2 Functions of the H8SX/1653

3.2 SCI_5

In this sample task, SCI_5 is used for asynchronous serial data transmission. Figure 3 is a block diagram of SCI_5, and the following is a description of the functions in the diagram.

- On-Chip Peripheral Clock Pϕ
  This is the base clock for the operation of on-chip peripheral functions and is generated by a clock oscillator.
- Receive Shift Register_5 (RSR_5)
  This register is used to receive serial data. Serial data on RSR_5 are input via the RxD5 pin. When one frame of data has been received, the data bits are automatically transferred to the Receive Data Register (RDR_5). RSR_5 is not accessible by the CPU.
- Receive Data Register_5 (RDR_5)
  Received data are stored in this 8-bit register. After RSR_5 has received one frame, the data bits are automatically transferred from RSR_5 to RDR_5. Since RSR_5 and RDR_5 function as a double buffer, continuous reception is possible. RDR_5 is for reception only, and so is seen as a read-only register by the CPU.
- Transmit Shift Register_5 (TSR_5)
  This register is used to transmit serial data. In transmission, data are transferred from the Transmit Data Register (TDR_5) to TSR_5, and then output on the TxD5 pin. TSR_5 is not directly accessible from the CPU.
- Transmit Data Register_5 (TDR_5)
  Data for transmission are stored in this 8-bit register. When SCI_5 detects that TDR_5 is empty, data that have been written to TDR_5 are automatically transferred to TSR_5. Since TDR_5 and TSR_5 function as a double buffer, if the next byte is written to TDR_5 before transmission of the frame including the byte currently in TSR_5 is complete, the byte can be transferred to TSR_5 immediately on completion of the transmission. This allows continual transmission. Although TDR can be read from or written to by the CPU at all times, only write data for transmission data after having confirmed setting of the TDRE bit in the Serial Status Register (SSR_5) to 1.
- Serial Mode Register_5 (SMR_5)
  This 8-bit register is used to select the format of serial data communications and the clock source for the on-chip baud-rate generator.
- Serial Control Register_5 (SCR_5)
  This register is used to control transmission, reception, and interrupts, and to select the clock source for transmission and reception.
- Serial Status Register_5 (SSR_5)
  This register consists of status flags for SCI_5 and multiprocessor bits for transmission and reception. TDRE, RDRF, ORER, PER, and FER can only be cleared.
- Smart Card Mode Register_5 (SCMR_5)
  This register is used to select the smart-card or normal interface mode for SCMR_5, and to set up the format for the smart-card mode. For this task, the setting in SCMR_5 selects the normal asynchronous or clock synchronous mode.
- Serial Extended Mode Register_5 (SEMR_5)
  SEMR_5 and SEMR_6 are used to select the clock source for SCI_5 and SCI_6 in the asynchronous mode. The base (peripheral) clock is automatically specified when average transfer rate operation is selected. TMO output from TMR units 2 and 3 can also be set as the base clock for serial transfer.

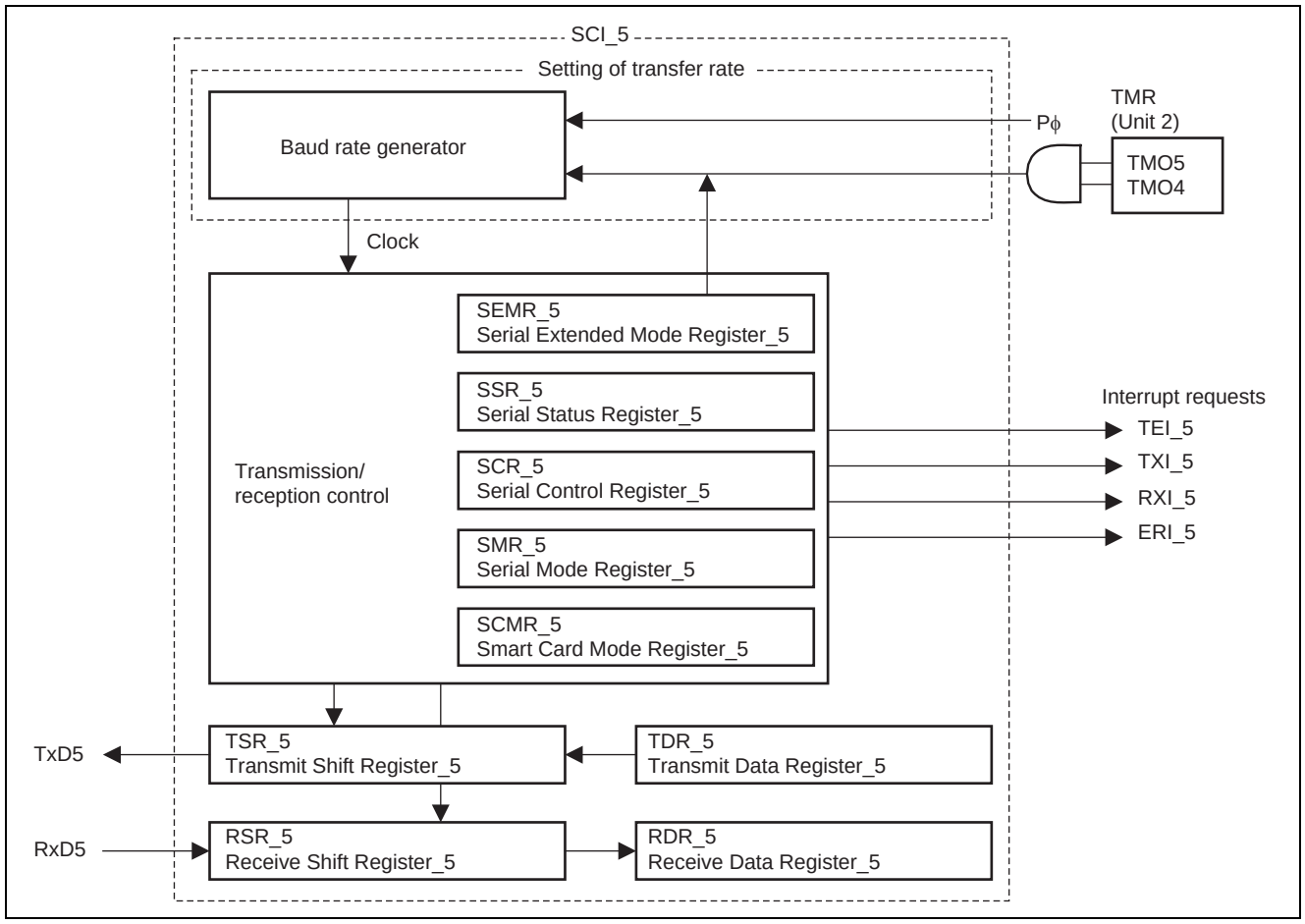


Figure 3 Block Diagram of SCI_5

3.3 TMR Unit 2

In this sample task, the clock source for SCI_5 is generated from logic values output by TMR unit 2 (TMR_4 and TMR_5). Figure 4 is a block diagram of TMR unit 2 and the usage of TMR unit 2 functions is described below.

- Internal peripheral clock Pϕ
  This is the standard operating clock for the internal peripheral functions and is generated by using the clock oscillator.

- Timer counter_4 (TCNT_4)
- Timer counter_5 (TCNT_5)
  Each TCNT is an 8-bit readable/writable register. Bits CKS2 to CKS0 in TCR and bits ICKS1 and ICKS0 in TCCR are used to select the clock signal to drive counting. Clearing of a TCNT register by an external reset input signal, compare match A signal, or compare match B signal is selectable by bits CCLR1 and CCLR0 in the corresponding TCR. The initial value of these registers is H'00.

- Time constant register A_4 (TCORA_4)
- Time constant register A_5 (TCORA_5)
  Each TCORA is an 8-bit readable/writable register. The value in TCORA is continually compared with the value in the corresponding TCNT. When a match is detected, the corresponding CMFA flag in the corresponding TCSR is set to 1. The settings of bits OS1 and OS0 in TCSR select whether and what kind of timer output is produced on the TMO terminal by this compare-match signal (compare match A). TCORA is initialized to H'FF.

- Time constant register B_4 (TCORB_4)
- Time constant register B_5 (TCORB_5)
  Each TCORB is an 8-bit readable/writable register. TCORB is continually compared with the value in the corresponding TCNT. When a match is detected, the CMFB flag in the corresponding TCSR is set to 1. The settings of bits OS3 and OS2 in TCSR select whether and what kind of timer output is produced on the TMO terminal by this compare-match signal (compare match A). The initial value of these registers is H'FF.

- Timer control register_4 (TCR_4)
- Timer control register_5 (TCR_5)
  Each TCR selects the TCNT clock source and the condition for clearing the corresponding TCNT, and enables/disables interrupt requests.

- Timer counter control register_4 (TCCR_4)
- Timer counter control register_5 (TCCR_5)
  Each TCCR selects the TCNT internal clock source and controls sensing of external resets.

- Timer control/status register_4 (TCSR_4)
- Timer control/status register_5 (TCSR_5)
  Each TCSR contains status flags, and controls compare match output.

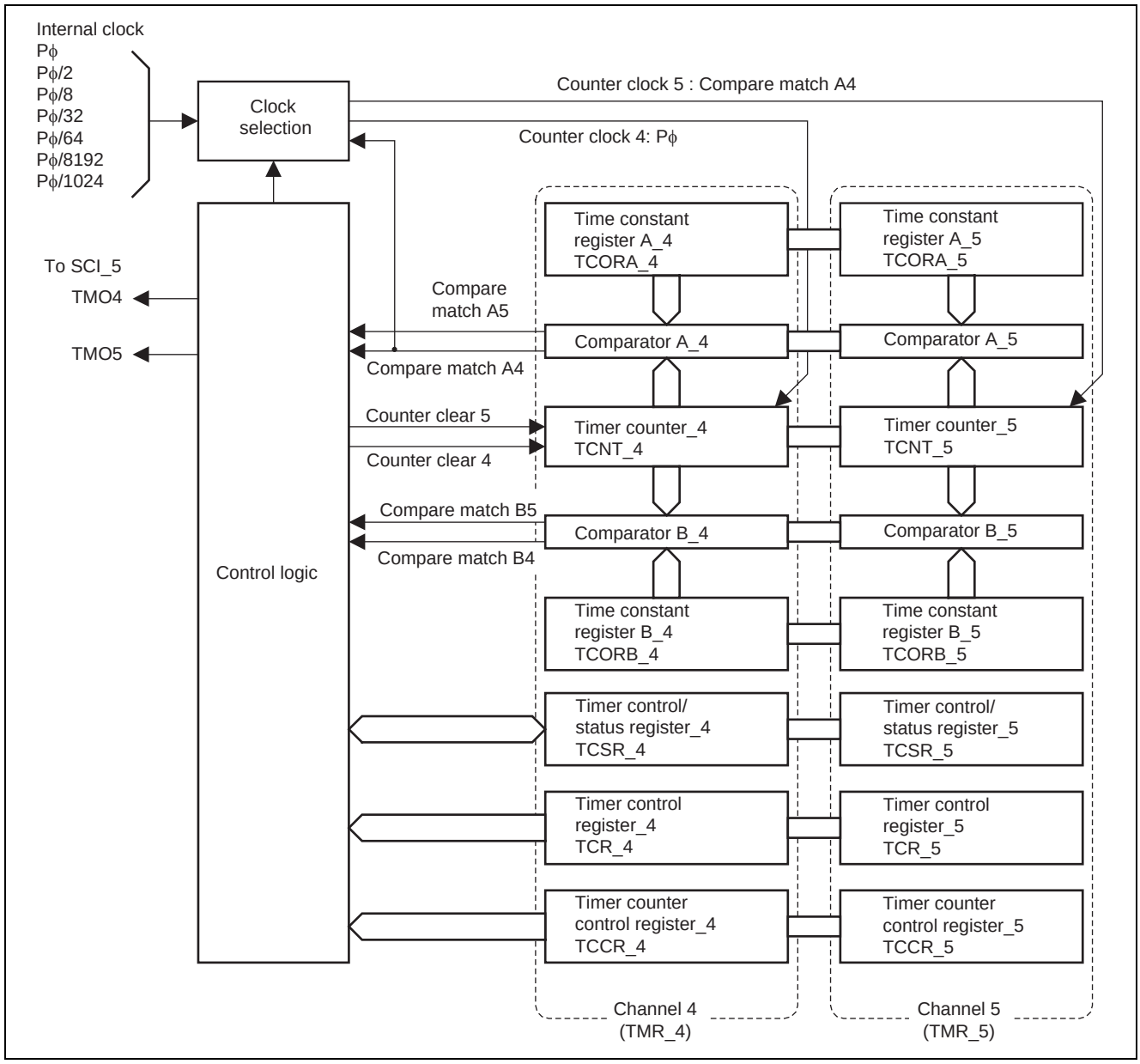


Figure 4 Block Diagram of TMR Unit 2

3.4 Channels 0 and 1 of the DMAC

In this sample task, DMAC channel 0 is activated by the TXI_5 interrupt of SCI_5 and DMAC channel 1 is activated by the RXI_5 interrupt of SCI_5. A block diagram of the DMAC is given as figure 5. The following description is with reference to figure 5.

- DMA source address register _0 (DSAR_0)
- DMA source address register _1 (DSAR_1)
  These registers are 32-bit readable/writable registers and specify the source address for the transfer. Each register is equipped with an address-updating function, so the source address is updated to that for the next transfer each time a transfer operation takes place.

- DMA destination address register _0 (DDAR_0)
- DMA destination address register _1 (DDAR_1)
  These registers are 32-bit readable/writable registers and specify the destination address for the transfer. Each register is equipped with an address-updating function, so the destination address is updated to that for the next transfer each time a transfer operation takes place.

- DMA transfer count register _0 (DTCR_0)
- DMA transfer count register _1 (DTCR_1)
  These registers are 32-bit readable/writable registers and specify the amount of data to be transferred (total size for transfer). After each data transfer operation, the value is reduced by the amount that corresponds to the transferred amount of data. In this sample task, both are set for 1536 bytes (H'00000600) of data, and the byte is selected as the unit of data access. Four is subtracted from the value on each DMAC operation, to indicate the amount still to be transferred.

- DMA mode control register _0 (DMDR_0)
- DMA mode control register _1 (DMDR_1)
  These registers control DMAC operation.

- DMA address control register_0 (DACR_0)
- DMA address control register_1 (DACR_1)
  These registers set the operating mode and transfer method.

- DMA module request select register_0 (DMRSR_0)
- DMA module request select register_1 (DMRSR_1)
  These registers set the activation source.

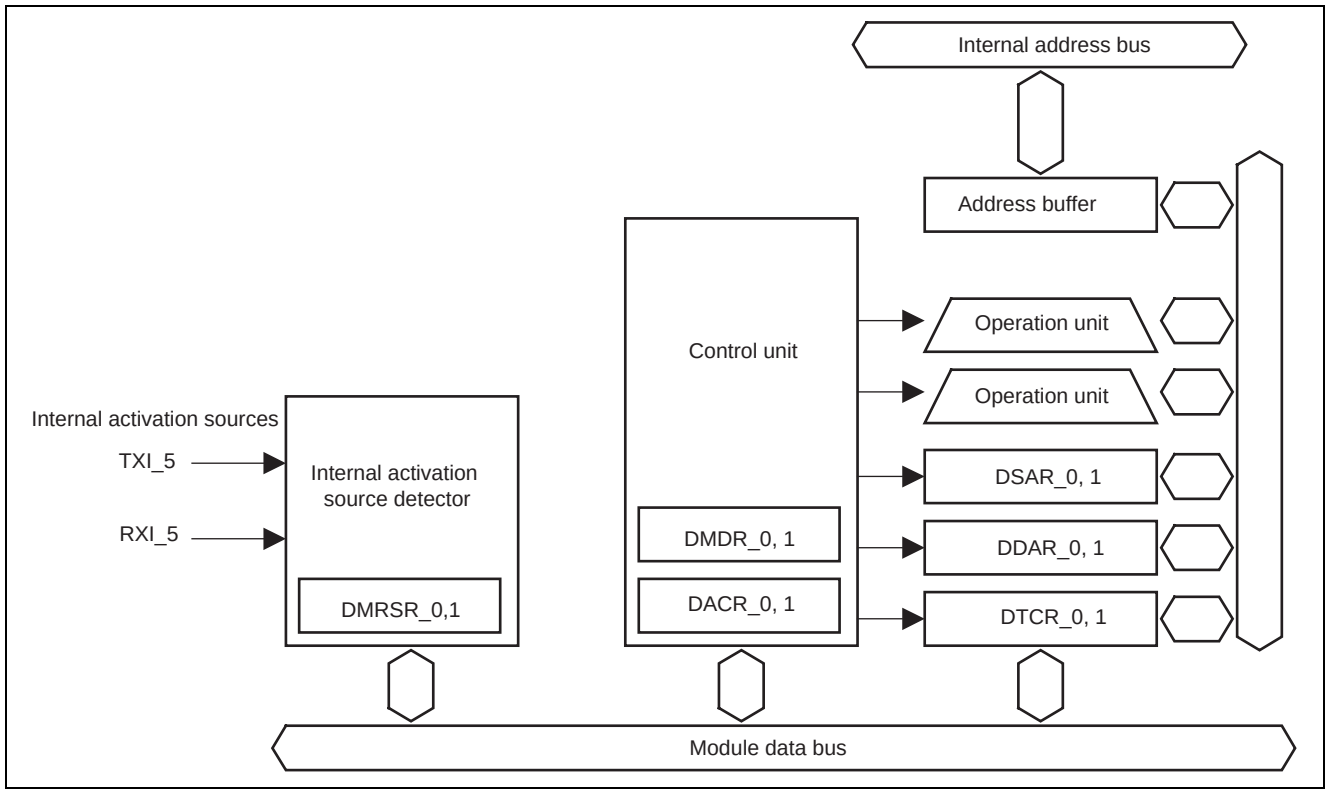


Figure 5 Block Diagram of the DMAC

4. Description of Operation

4.1 Outline

An outline of operation for this sample task is given in figure 6. 128-byte blocks of data are simultaneously transferred in both directions between the master and slave sides.

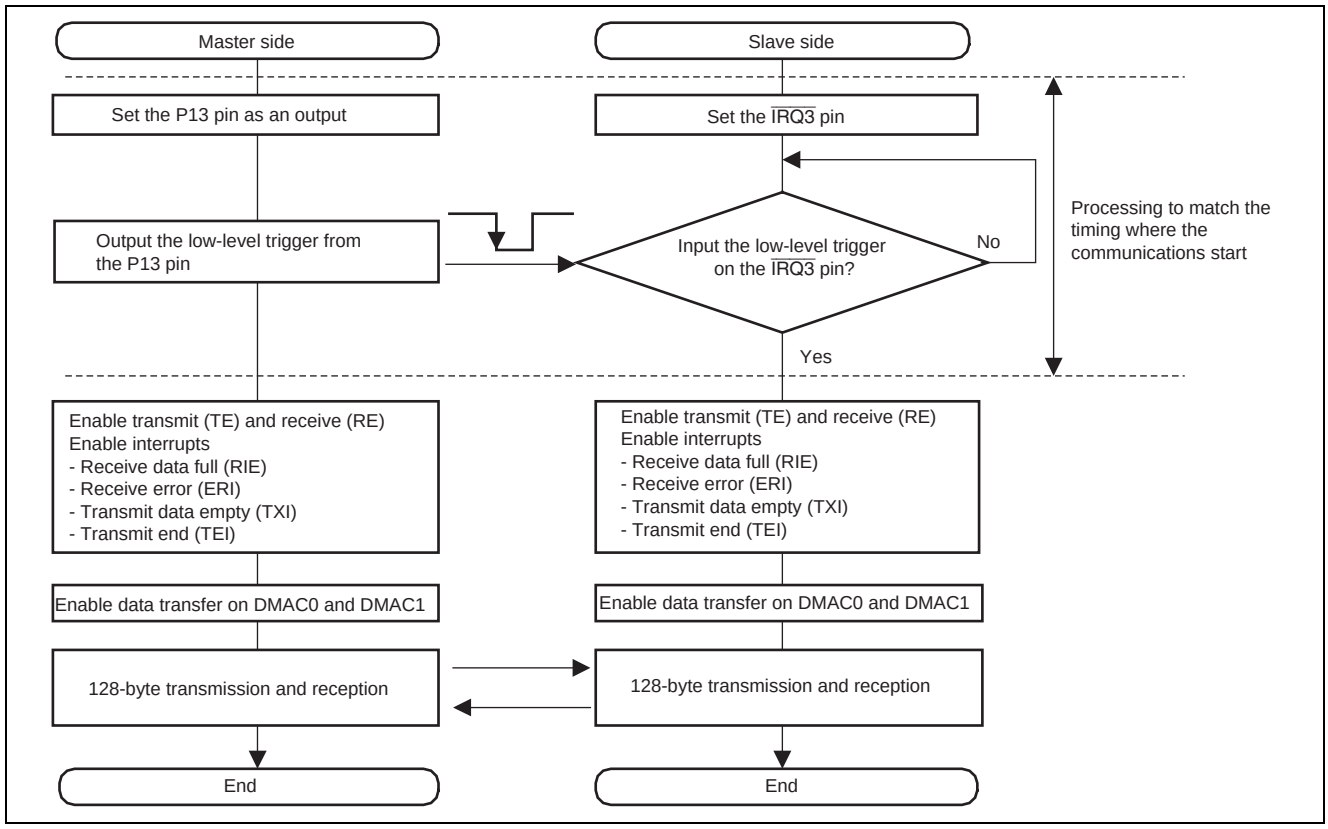


Figure 6 Outline of Operation

4.2 Transmission

The timing of transmission operations is illustrated in figure 7. Table 4 is a list of the hardware and software processing at the numbered points in figure 7.

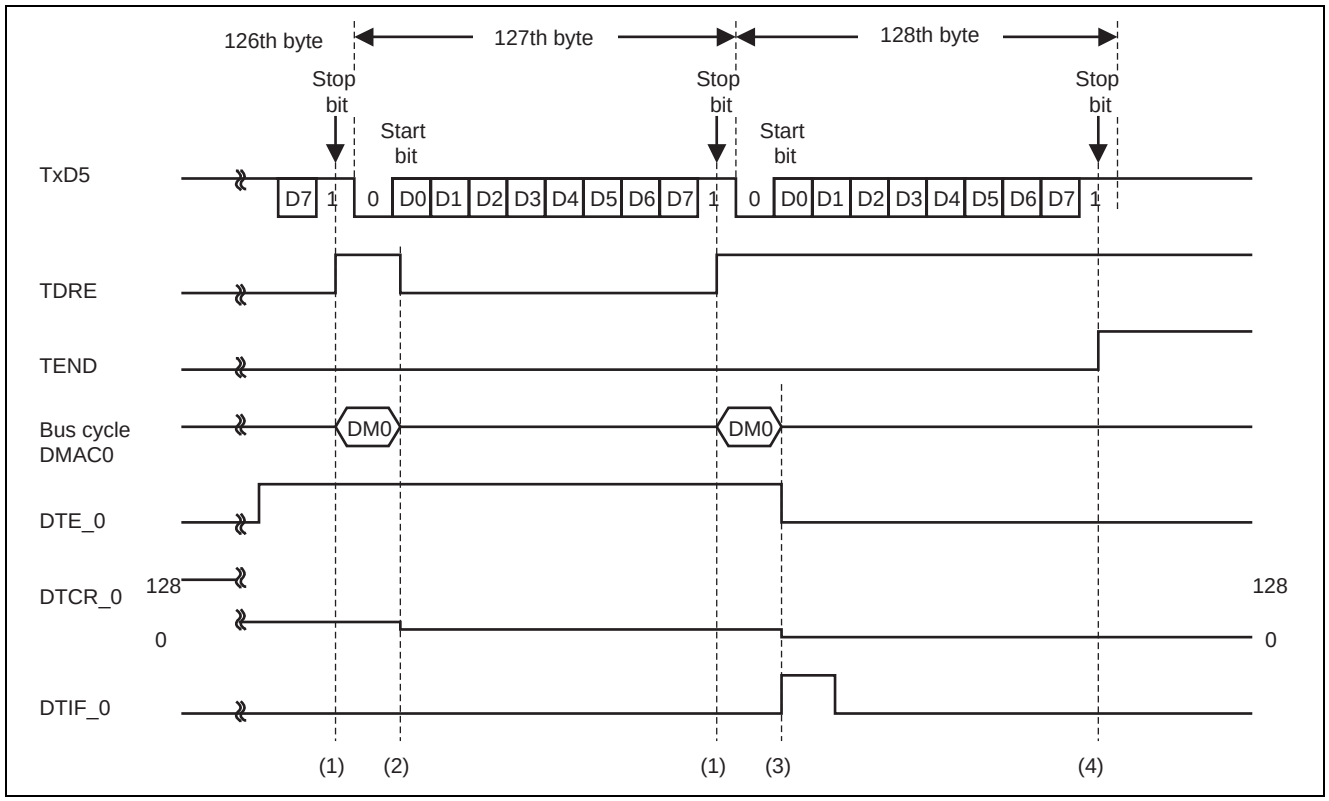


Figure 7 Timing of Transmission

Table 4 Processing

	Hardware Processing	Software Processing
(1)	a. Set TDRE to 1. b. Activate DMAC_0, and transfer data for transmission from RAM to TDR_5. c. Clear TDRE to 0.	None
(2)	a. Decrement DTCR_0. b. Transfer the contents of TDR_5 to TSR_5. c. Output the contents of data of TSR_5 on pin TxD5.	None
(3)	a. Decrement DTCR_0 (Producing DTCR_0 = 0) b. Transfer the contents of TDR_5 to TSR_5. c. Output the contents of TSR_5 from pin TxD5.	DMA transfer end interrupt processing a. Disable transmission and transmission end interrupts.
(4)	a. Set TEND to 1.	TEI interrupt processing a. Clear TE (to 0). b. Disable TEI interrupts.

4.3 Reception

The timing of reception operations is illustrated in figure 8. Table 5 is a list of the hardware and software processing at the numbered points in figure 8.

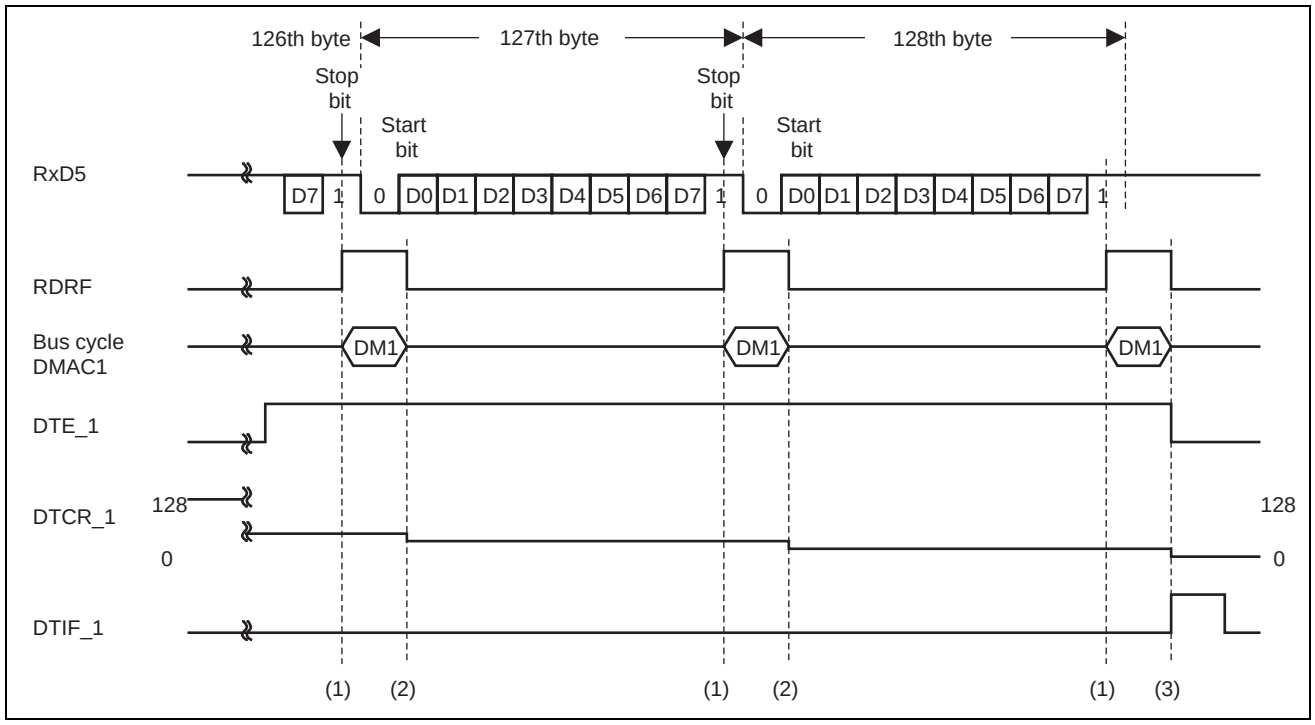


Figure 8 Timing of Reception

Table 5 Processing

	Hardware Processing	Software Processing
(1)	a. Set RDRF to 1. b. Each time a byte is successfully received in RSR_5, transfer it to RDR_5. c. Activate DMAC_1, and transfer received data from RDR_5 to RAM. d. Clear RDRF5 to 0.	None
(2)	a. Decrement DTCR_1.	None
(3)	a. Decrement DTCR_0 (DTCR_1 = 0).	Transfer end interrupt processing a. Disable reception and reception interrupts.

4.4 Internal Base Clock Settings for SCK5

An internal 6-MHz base clock is derived from the 16-MHz peripheral clock signal (Pϕ) and then used to produce an average transfer rate of 375 kbps. The procedure is described below.

1. TMR4 Settings
   Output of an 8-MHz signal as a base clock on TMO4 (refer to figure 9)
   a. To select incrementation of TCNT_4 on rising edges of Pϕ, set CKS2 to CKS0 in TCR_4 to B'011 and ICKS1 and ICKS0 in TCCR_4 to B'10.
   b. Select clearing of TCNT_4 on matches with TCORA_4 by setting CCLR1 in TCR_4 to 0 = B'01.
   c. To select the output of a 0 on matches with TCORA_4 and of a 1 on matches with TCORB_4, set bits OS3 to OS0 in TCSR_4 to B'1001.
   d. Set TCORA_4 = 1 and TCORB_4 = 0 to obtain an 8-MHz base clock signal on TMO4.

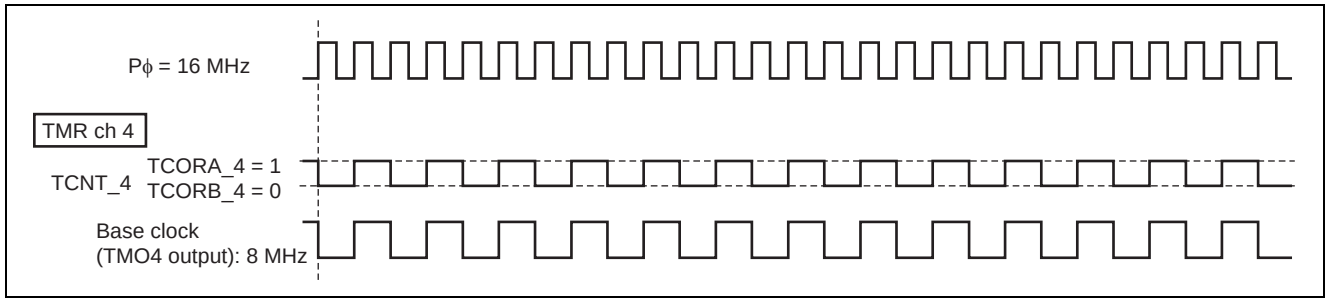


Figure 9 Deriving the Output Waveform on TMO4

2. TMR5 Settings
   Output of a clock-enable signal with a duty cycle of 3/4 on TMO5 (refer to figure 10).
   a. To select incrementation of TCNT_5 on matches of TCNT4 with TCORA_4 (compare match A), set CKS2 to CKS0 in TCR_5 to B'100.
   b. To select clearing of TCNT_5 on matches with TCORA_5, set bit field CCLR1, 0 in TCR_5 to B'01.
   c. To select the output of a 0 on matches with TCORA_5 and of a 1 on matches with TCORB_5, set bits OS3 to OS0 in TCSR_5 to B'1001.
   d. When TCORA_5 = 3 and TCORB_5 = 0, a clock-enable signal with a duty cycle of 3/4 is output on TMO5.

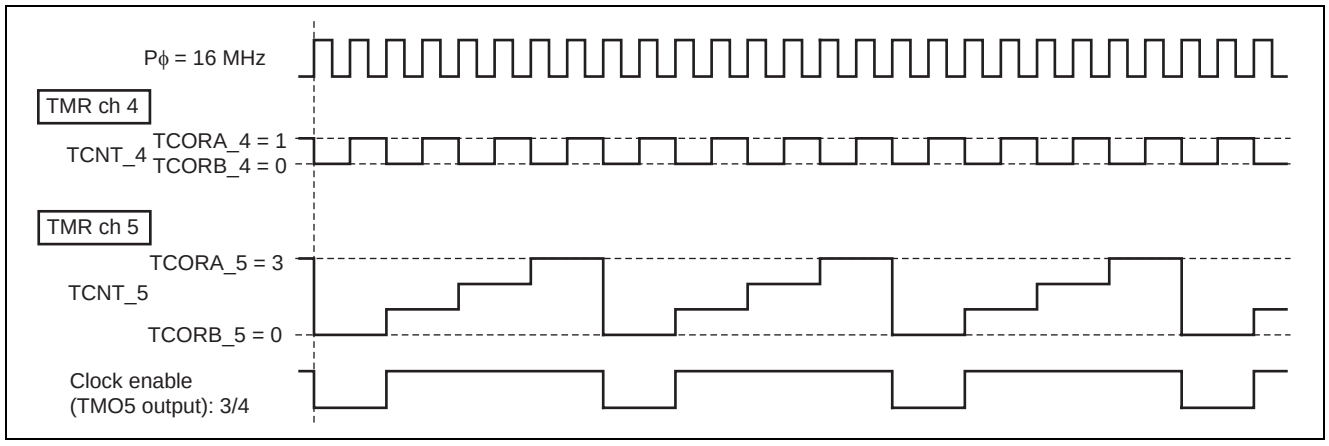


Figure 10 Deriving the Output Waveform on TMO5

3. Internal base clock of SCK5 and average transfer rate

   The logical AND of outputs TMO4 and TMO5 provides the internal base-clock (6-MHz) signal for SCK5. Derivation of this waveform is illustrated in figure 11.

   When the ABCS bit in SEMR_5 is 0, one bit of data is transferred every 16 cycles of the internal base clock. This leads to an average transfer rate of 375 kbps, as given by the formula below.

$$\text{Average transfer rate} = \frac{\text{Internal base clock for SCK5}}{\text{16 clocks}} = \frac{\text{6 MHz}}{16} = 375 \text{ kbps}$$

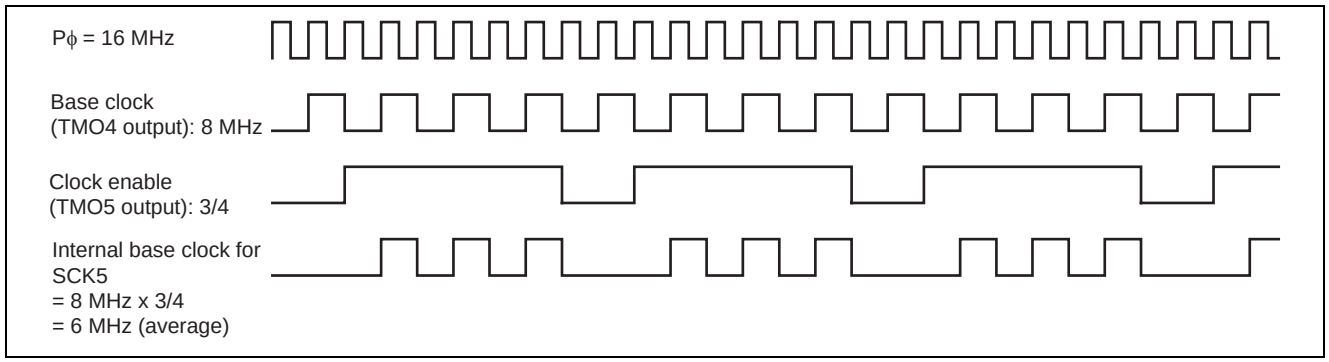


Figure 11 Deriving the Internal Base-Clock Waveform for SCK5

4.5 One-bit Period for Communications Data

The one-bit period in this sample task will vary according to the number of base-clock cycles omitted from the bit period. This will be either five or six, so the one-bit period is given by the corresponding formula below.

$$\text{Higher period for one bit: } \frac{1}{\text{Base clock frequency}} \times (\text{16 clocks + No. of cycles omitted: 6}) = \frac{1}{\text{8 MHz}} \times (16 + 6) = 2.750\ \mu\text{s}$$

$$\text{Lower period for one bit: } \frac{1}{\text{Base clock frequency}} \times (\text{16 clocks + No. of cycles omitted: 5}) = \frac{1}{\text{8 MHz}} \times (16 + 5) = 2.625\ \mu\text{s}$$

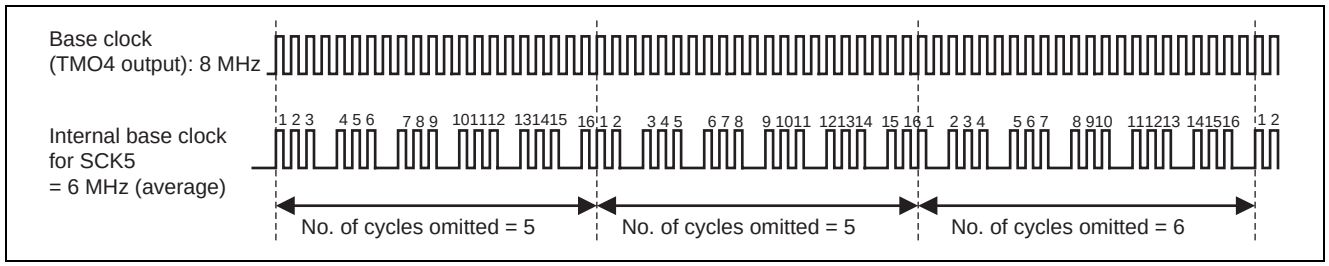


Figure 12 One-Bit Periods for Data Transfer

5. Description of Software

5.1 List of Functions

Table 6 lists the functions used in this sample task. Figure 13 shows the structure of hierarchy.

Table 6 List of Functions

Function Name	Functions
init	Initialization routine: Takes the chip out of module stop mode, performs clock settings, and calls the main function.
main	Main routine: Makes initial SCI settings for communications at the transfer rate of 375 kbps when operating at Pϕ = 16 MHz.
DMAC0_trs_init	DMAC_0 initialization: Processing for transfer on TXI from ROM to TDR_5
DMAC1_rcv_init	DMAC_1 initialization: Processing for transfer on RXI from RDR_5 to RAM
dmtend0_int	DMAC_0 transfer end interrupt handler: Disables SCI transmission and SCI transmission interrupts.
dmtend1_int	DMAC_1 transfer end interrupt handler: Disables SCI reception and SCI reception interrupts.
eri5_int	Receive error interrupt handler: In cases of error in reception, writes the contents of SSR_5 to RAM and then initializes SSR_5.
tei5_int	Transmit end interrupt handler: Disables TEI interrupt requests. Sets endflg to 1.

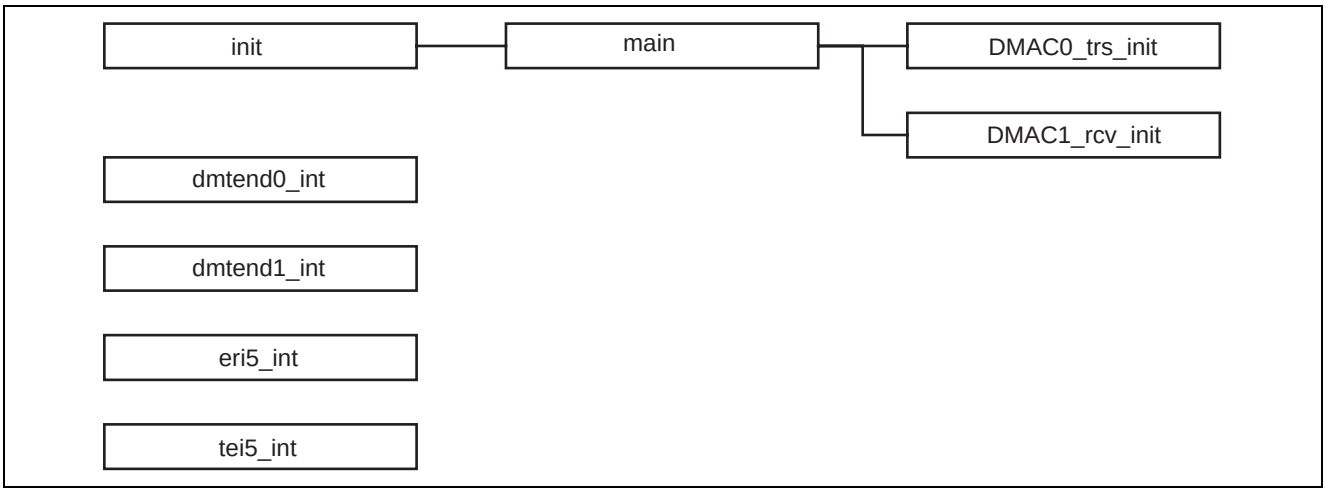


Figure 13 Structure of Hierarchy

5.2 Vector Table

Table 7 Exception Handling Vector Table

Exception Source	Vector Number	Address in Vector Table	Target Handling Routine
Reset	0	H'000000	init
DMAC_0 DMTEND0	128	H'000200	dmtend0_int
DMAC_1 DMTEND1	129	H'000204	dmtend1_int
SCI_5 ERI5	222	H'000378	eri5_int
SCI_5 TEI5	223	H'00037C	tei5_int

5.3 RAM Usage

Table 8 RAM Usage

Type	Variable Name	Description	Used in
unsigned char	endflg	Transmission end flag 0: Transmission in progress 1: Transmission ended	main, tei5_int
unsigned char	errbuf	Reception error buffer The contents of SSR_5 are stored when an overrun, framing, or parity error occurs.	main, eri5_int
unsigned char	tcnt	Transmission counter	main, txi5_int
unsigned char	rcnt	Reception counter	main, rxi5_int
unsigned char	rcv_dt[128]	RAM area for storing received data	main, rxi5_int

5.4 Data Table

Table 9 Data Table

Type	Array Name	Description	Used in
unsigned char	trs_dt[128]	ROM area where data for transmission are stored 128 bytes of data H'00, H'01, …, H'7F	main, txi5_int

5.5 Macro Definitions

Table 10 Macro Definitions

Identifier	Description	Used in
MASTER	If this is defined, compilation generates the master-side program.	main
SLAVE	If this is defined, compilation generates the slave-side program.	main

5.6 Description of Functions

5.6.1 init Function

1. Overview
   Initialization routine: Takes the module out of module stop mode, sets the clock, and calls the main function.
2. Arguments
   None
3. Return value
   None
4. Description of internal register usage
   Usage of internal registers in this task is described below. The given settings are those used in the task and differ from the initial settings.

- **System clock control register (SCKCR) Address: H'FFFDC4**

Bit	Bit Name	Setting	R/W	Description
10	ICK2	0	R/W	System Clock (Iϕ) Select
9	ICK1	1	R/W	Selects the frequency of the CPU, DMAC, and DTC module and
8	ICK0	0	R/W	system clock. 010: Input clock x 1
6	PCK2	0	R/W	Peripheral Module Clock (Pϕ) Select
5	PCK1	1	R/W	Selects the frequency of the peripheral module clock.
4	PCK0	0	R/W	010: Input clock x 1
2	BCK2	0	R/W	External Bus Clock (Bϕ) Select
1	BCK1	1	R/W	Selects the frequency of the external bus clock.
0	BCK0	0	R/W	010: Input clock x 1

- Registers MSTPCRA, MSTPCRB, and STPCRC control the module stop mode. Setting a bit to 1 makes the corresponding module enter the module stop mode, while clearing the same bit to 0 takes the module out of stop mode.

- **Module stop control register A (MSTPCRA) Address: H'FFFDC8**

Bit	Bit Name	Setting	R/W	Description
15	ACSE	0	R/W	All-Module-Clock-Stop Mode Enable Enables/disables all-module-clock-stop mode for reducing current drawn by stopping the bus controller and I/O port operations when the CPU executes the SLEEP instruction after the module stop mode has been set for all of the on-chip peripheral modules controlled by MSTPCR. 0: All-module-clock-stop mode disabled 1: All-module-clock-stop mode enabled
13	MSTPA13	0	R/W	DMA controller (DMAC)
12	MSTPA12	1	R/W	Data transfer controller (DTC)
9	MSTPA9	1	R/W	8-bit timers (TMR_3, TMR_2)
8	MSTPA8	1	R/W	8-bit timers (TMR_1, TMR_0)
5	MSTPA5	1	R/W	D/A converter (channel 1, channel 0)
3	MSTPA3	1	R/W	A/D converter (unit 0)
0	MSTPA0	1	R/W	16-bit timer pulse unit (TPU channels 5 to 0)

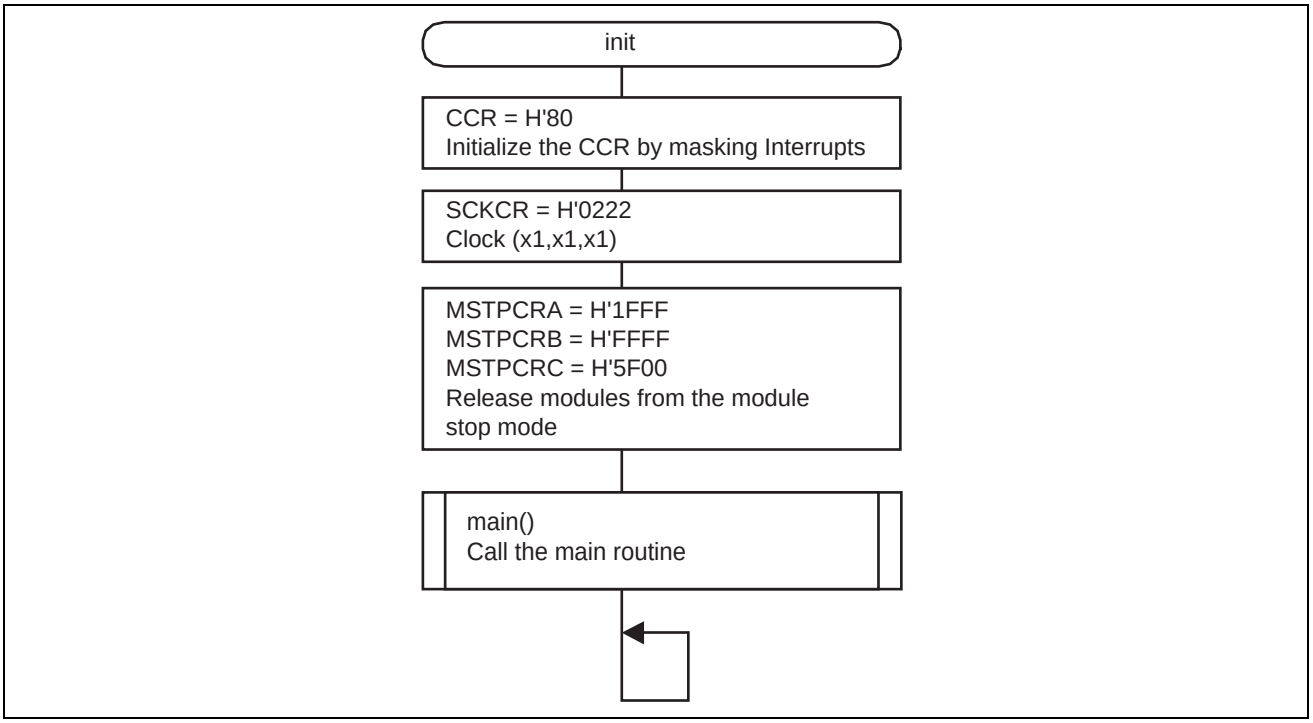
- **Module stop control register B (MSTPCRB) Address: H'FFFDCA**

Bit	Bit Name	Setting	R/W	Description
15	MSTPB15	1	R/W	Programmable pulse generator (PPG)
12	MSTPB12	1	R/W	Serial communications interface_4 (SCI_4)
10	MSTPB10	1	R/W	Serial communications interface_2 (SCI_2)
9	MSTPB9	1	R/W	Serial communications interface_1 (SCI_1)
8	MSTPB8	1	R/W	Serial communications interface_0 (SCI_0)
7	MSTPB7	1	R/W	I^2C bus interface_1 (IIC_1)
6	MSTPB6	1	R/W	I^2C bus interface_0 (IIC_0)

- **Module stop control register C (MSTPCRC) Address: H'FFFDCC**

Bit	Bit Name	Setting	R/W	Description
15	MSTPC15	0	R/W	Serial communications interface _5 (SCI_5), (IrDA)
14	MSTPC14	1	R/W	Serial communications interface _6 (SCI_6)
13	MSTPC13	0	R/W	8-bit timers (TMR_4, TMR_5)
12	MSTPC12	1	R/W	8-bit timers (TMR_6, TMR_7)
11	MSTPC11	1	R/W	Universal serial bus interface (USB)
10	MSTPC10	1	R/W	Cyclic redundancy checker
4	MSTPC4	0	R/W	On-chip RAM_4 (H'FF2000 to H'FF3FFF)
3	MSTPC3	0	R/W	On-chip RAM _3 (H'FF4000 toH'FF5FFF)
2	MSTPC2	0	R/W	On-chip RAM _2 (H'FF6000 to H'FF7FFF)
1	MSTPC1	0	R/W	On-chip RAM _1 (H'FF8000 to H'FF9FFF)
0	MSTPC0	0	R/W	On-chip RAM _0 (H'FFA000 to H'FFBFFF)

5. Flowchart



5.6.2 main Function

1. Overview
   Main routine: Sets the timer clock input and SCI, calls functions DMAC0_trs_init and DMAC1_rcv_init, and transmits and receives a total of 256 bytes of data.
2. Arguments
   None
3. Return value
   None
4. Description of internal register usage
   Usage of internal registers in this task is described below. The given settings are those used in the task and differ from the initial settings.

- **Port 1 data direction register (P1DDR) Address: H'FFFB80**

Bit	Bit Name	Setting	R/W	Description
3	P13DDR	1	W	0: Pin P13 is an input. 1: Pin P13 is an output.

- **Port 1 input buffer control register (P1ICR) Address: H'FFFB90**

Bit	Bit Name	Setting	R/W	Description
5	P15ICR	1	R/W	0: P15 pin input buffer is disabled. Input signal is fixed to the high level. 1: P15 pin input buffer is valid. The pin state reflects the peripheral modules.
3	P13ICR	1	R/W	0: P13 pin input buffer is disabled. Input signal is fixed to the high level. 1: P15 pin input buffer is valid. The pin state reflects the peripheral modules.

- **Port function control register C (PFCRC) Address: H'FFFBCC**

Bit	Bit Name	Setting	R/W	Description
3	ITS3	1	R/W	$\overline{\text{IRQ3}}$ Pin Selection 0: Selects $\overline{\text{IRQ3}}$-A input on pin P13 1: Selects $\overline{\text{IRQ3}}$-B input on pin P53

- **IRQ sense control register L (ISCRL) Address: H'FFFD6A**

Bit	Bit Name	Setting	R/W	Description
7	IRQ3SR	0	R/W	IRQ3 Sense Control Rise
6	IRQ3SF	1	R/W	IRQ3 Sense Control Fall 01: Interrupt requests are sensed on falling edges of $\overline{\text{IRQ3}}$ input

- **Timer control register_4 (TCR_4) Address: H'FFEA40**

Bit	Bit Name	Setting	R/W	Description
4	CCLR1	0	R/W	Counter Clear 1, 0
3	CCLR0	1	R/W	01: TCNT_4 is cleared on matches with TCORA_4.
2	CKS2	0	R/W	Clock Select 2 to 0
1	CKS1	1	R/W	Refer to table 11.
0	CKS0	1	R/W	011, ICKS 1, 0 = B'10: Counting on rising edges of Pϕ.

- **Timer control register_5 (TCR_5) Address: H'FFEA41**

Bit	Bit Name	Setting	R/W	Description
4	CCLR1	0	R/W	Counter Clear 1, 0
3	CCLR0	1	R/W	01: TCNT_5 is cleared on matches with TCORA_5.
2	CKS2	1	R/W	Clock Select 2 to 0
1	CKS1	0	R/W	Refer to table 11.
0	CKS0	0	R/W	100: Counting is driven by matches with TCORA_4 (compare match A).

- **Timer control/status register_4 (TCSR_4) Address: H'FFEA42**

Bit	Bit Name	Setting	R/W	Description
3	OS3	1	R/W	Output Select 3, 2
2	OS2	0	R/W	Selects how a match between TCORB_4 and TCNT_4 affects the output on the TMO_4 terminal. 10: 1 output
1	OS1	0	R/W	Output Select 1, 0
0	OS0	1	R/W	Selects how a match between TCORA_4 and TCNT_4 affects the output on the TMO_4 terminal. 01: 0 output

- **Timer control/status register_5 (TCSR_5) Address: H'FFEA43**

Bit	Bit Name	Setting	R/W	Description
3	OS3	1	R/W	Output Select 3, 2
2	OS2	0	R/W	Selects how a match between TCORB_5 and TCNT_5 affects the output on the TMO_5 terminal. 10: 1 output
1	OS1	0	R/W	Output Select 1, 0
0	OS0	1	R/W	Selects how a match between TCORA_5 and TCNT_5 (compare match A) affects the output on the TMO_5 terminal. 01: 0 output

- **Timer constant register A_4 (TCORA_4) Address: H'FFEA44**

  Function: This is an 8-bit readable/writable register. Its value is continually compared with that in TCNT_4. When a match is detected, the CMFA flag in TCSR_4 is set to 1.

  Setting: H'01

- **Timer constant register A_5 (TCORA_5) Address: H'FFEA45**
  Function: This is an 8-bit readable/writable register. Its value is continually compared with that in TCNT_5. When a match is detected, the CMFA flag in TCSR_5 is set to 1.
  Setting: H'03

- **Timer constant register B_4 (TCORB_4) Address: H'FFEA46**
  Function: This is an 8-bit readable/writable register. Its value is continually compared with that in TCNT_4. When a match is detected, the CMFB flag in TCSR_4 is set to 1.
  Setting: H'00

- **Timer constant register B_5 (TCORB_5) Address: H'FFEA47**
  Function: This is an 8-bit readable/writable register. Its value is continually compared with that in TCNT_5. When a match is detected, the CMFB flag in TCSR_5 is set to 1.
  Setting: H'00

- **Timer counter__4 (TCNT_4) Address: H'FFEA48**
  Function: This is an 8-bit readable/writable register. In this sample task, this register is cleared by the compare match A signal from TMR_4.
  Setting: H'00

- **Timer counter__5 (TCNT_5) Address: H'FFEA49**
  Function: This is an 8-bit readable/writable register. In this sample task, this register is cleared by the compare match A signal from TMR_5.
  Setting: H'00

- **Timer counter control register_4 (TCCR_4) Address: H'FFEA4A**

Bit	Bit Name	Setting	R/W	Description
1	ICKS1	1	R/W	Internal Clock Select 1, 0
0	ICKS0	0	R/W	Refer to table 11.

- **Timer counter control register_5 (TCCR_5) Address: H'FFEA4B**

Bit	Bit Name	Setting	R/W	Description
1	ICKS1	1	R/W	Internal Clock Select 1, 0
0	ICKS0	0	R/W	Refer to table 11.

Table 11 Clock Input for TCNT and Condition for Counting (Units 2 and 3)

	TCR			TCCR		
	Bit 2	Bit 1	Bit 0	Bit 1	Bit 0	
Channel	CKS2	CKS1	CKS0	ICKS1	ICKS0	Description
TMR_4	0	0	0	—	—	Clock input prohibited
	0	0	1	0	0	The internal peripheral clock is used; counting is on rising edges of Pϕ/8.
				0	1	The internal peripheral clock is used; counting is on rising edges of Pϕ/2.
				1	0	The internal peripheral clock is used; counting is on falling edges of Pϕ/8.
				1	1	The internal peripheral clock is used; counting is on falling edges of Pϕ/2.
	0	1	0	0	0	The internal peripheral clock is used; counting is on rising edges of Pϕ/64.
				0	1	The internal peripheral clock is used; counting is on rising edges of Pϕ/32.
				1	0	The internal peripheral clock is used; counting is on falling edges of Pϕ/64.
				1	1	The internal peripheral clock is used; counting is on falling edges of Pϕ/32.
	0	1	1	0	0	The internal peripheral clock is used; counting is on rising edges of Pϕ/8192.
				0	1	The internal peripheral clock is used; counting is on rising edges of Pϕ/1024.
				1	0	The internal peripheral clock is used; counting is on rising edges of Pϕ/8192.
				1	1	The internal peripheral clock is used; counting is on falling edges of Pϕ/1024.
	1	0	0	—	—	Counting is on TCNT_5 overflow signals

	TCR			TCCR		
	Bit 2	Bit 1	Bit 0	Bit 1	Bit 0	
Channel	CKS2	CKS1	CKS0	ICKS1	ICKS0	Description
TMR_5	0	0	0	—	—	Clock input prohibited
	0	0	1	0	0	The internal peripheral clock is used; counting is on rising edges of Pϕ/8.
				0	1	The internal peripheral clock is used; counting is on rising edges of Pϕ/2.
				1	0	The internal peripheral clock is used; counting is on falling edges of Pϕ/8.
				1	1	The internal peripheral clock is used; counting is on falling edges of Pϕ/2.
	0	1	0	0	0	The internal peripheral clock is used; counting is on rising edges of Pϕ/64.
				0	1	The internal peripheral clock is used; counting is on rising edges of Pϕ/32.
				1	0	The internal peripheral clock is used; counting is on falling edges of Pϕ/64.
				1	1	The internal peripheral clock is used; counting is on falling edges of Pϕ/32.
	0	1	1	0	0	The internal peripheral clock is used; counting is on rising edges of Pϕ/8192.
				0	1	The internal peripheral clock is used; counting is on rising edges of Pϕ/1024.
				1	0	The internal peripheral clock is used; counting is on rising edges of Pϕ/8192.
				1	1	The internal peripheral clock is used; counting is on falling edges of Pϕ/1024.
	1	0	0	—	—	Counting is of TCNT_4 compare match A events

- **Serial mode register_5 (SMR_5) Address: H'FFF600**

Bit	Bit Name	Setting	R/W	Description
7	C/A	0	R/W	Communications Mode 0: Asynchronous 1: Clock synchronous
6	CHR	0	R/W	Character Length 0: Selects 8 bits as the data length 1: Selects 7 bits as the data length
5	PE	0	R/W	Parity Enable 0: No parity bit 1: Parity bit included
3	STOP	0	R/W	Stop Bit Length Selects the length of the stop-bit field in transmission. 0: 1 stop bit 1: 2 stop bits In reception, only the first of the stop bits is checked, and when the second stop bit is 0, it is treated as the start bit of the next frame to be transmitted.

- **Serial control register_5 (SCR_5) Address: H'FFF602**

Bit	Bit Name	Setting	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable 0: Disables TXI interrupts 1: Enables TXI interrupts
6	RIE	0	R/W	Receive Interrupt Enable 0: Disables RXI, ERI interrupts 1: Enables RXI, ERI interrupts
5	TE	0	R/W	Transmit Enable 0: Disables transmission 1: Enables transmission
4	RE	0	R/W	Receive Enable 0: Disables reception 1: Enables reception
2	TEIE	0	R/W	Transmit End Interrupt Enable 0: Disables TEI interrupts 1: Enables TEI interrupts
1	CKE1	1	R/W	Clock Enable 1, 0
0	CKE0	X	R/W	Select the clock source 00: Internal baud rate generator 1X: Timer clock input or average transfer rate generator

Legend
X: Don't care.

- **Serial status register_5 (SSR_5) Address: H'FFF604**

Bit	Bit Name	Setting	R/W	Description
7	TDRE	Undefined	R/(W)*	Transmit Data Register Empty Indicates whether TDR contains data for transmission [Setting conditions] • Clearing of the TE bit in SCR (to 0) • Transfer of data from TDR to TSR [Clearing conditions] • Writing 0 to TDRE after having read TDRE = 1 • Generation of a TXI interrupt request allowing DMAC to write transmit data to TDR
6	RDRF	0	R/(W)*	Receive Data Register Full Indicates whether RDR holds received data [Setting condition] • The normal end of serial reception and the transfer of received data from RSR to RDR [Clearing conditions] • Writing of 0 to RDRF after having read RDRF = 1 • Generation of an RXI interrupt request allowing DMAC or DTC to read data from RDR. The RDRF flag is not affected and retains its previous value even though the RE bit in SCR is cleared to 0. Note that when the next reception is completed while the RDRF flag is being set to 1, an overrun error occurs and the received data are lost.
5	ORER	0	R/(W)*	Overrun Error [Setting condition] • Occurrence of an overrun error during reception [Clearing condition] • Writing of 0 to ORER after having read ORER = 1
4	FER	0	R/(W)*	Framing Error [Setting condition] • Occurrence of a framing error during reception [Clearing condition] • Writing of 0 to FER after having read FER = 1
3	PER	0	R/(W)*	Parity Error [Setting condition] • Occurrence of a parity error during reception [Clearing condition] • Writing of 0 to PER after having read PER = 1
2	TEND	Undefined	R	Transmit End [Setting condition] • Clearing of the TE bit in SCR (to 0) • TDRE = 1 on transmission of the last bit of a character [Clearing conditions] • Writing of 0 to PER after having read PER = 1 • Generation of a TXI interrupt request allowing its value DMAC to write data to TDR

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

- **Smart card mode register_5 (SCMR_5) Address: H'FFF606**

Bit	Bit Name	Setting	R/W	Description
0	SMIF	0	R/W	Smart Card Interface Mode Select 0: Operation is in the normal asynchronous or clock synchronous mode 1: Operation is in smart card interface mode

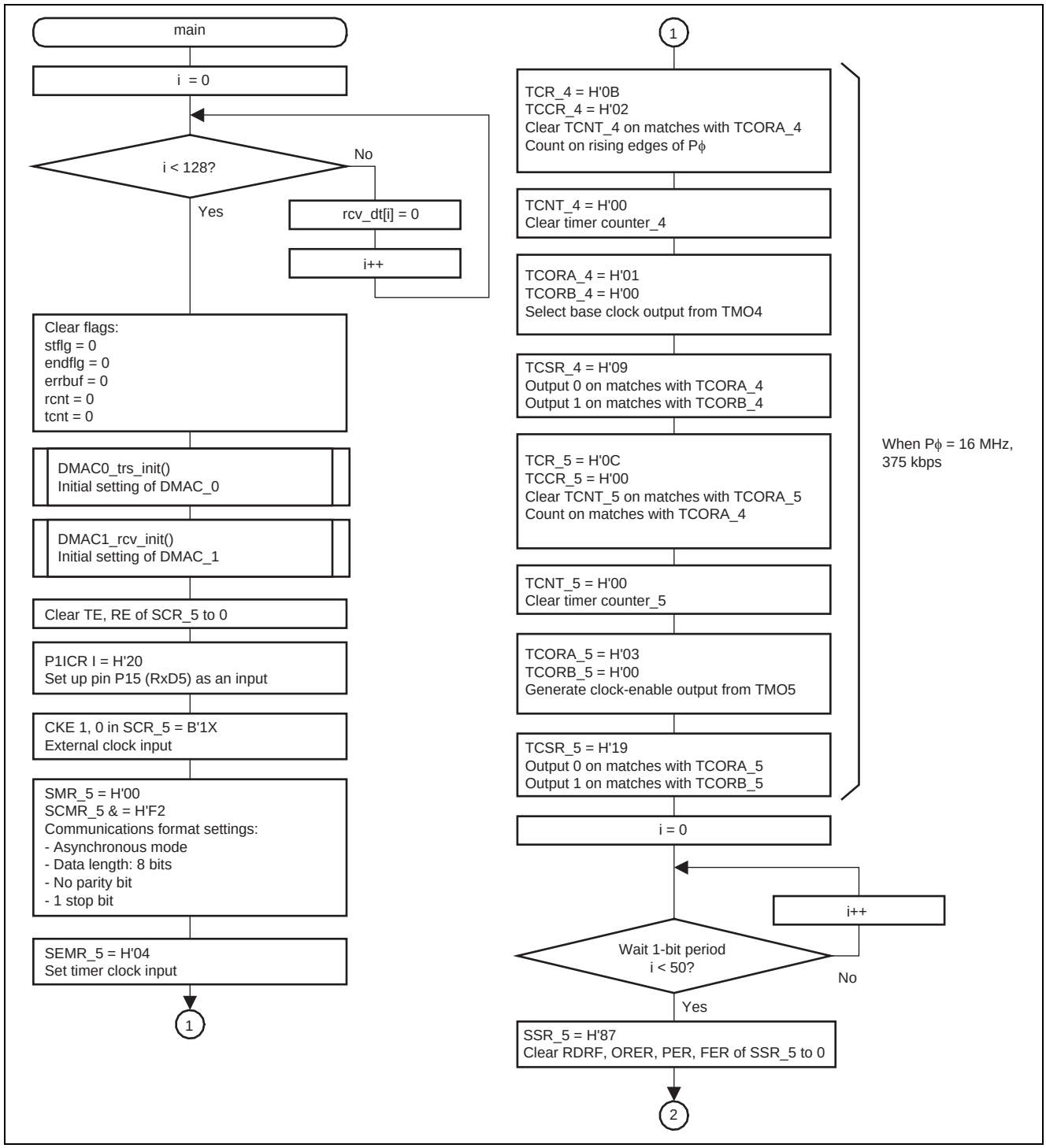
- **Serial extended mode register_5 (SEMR_5) Address: H'FFF608**

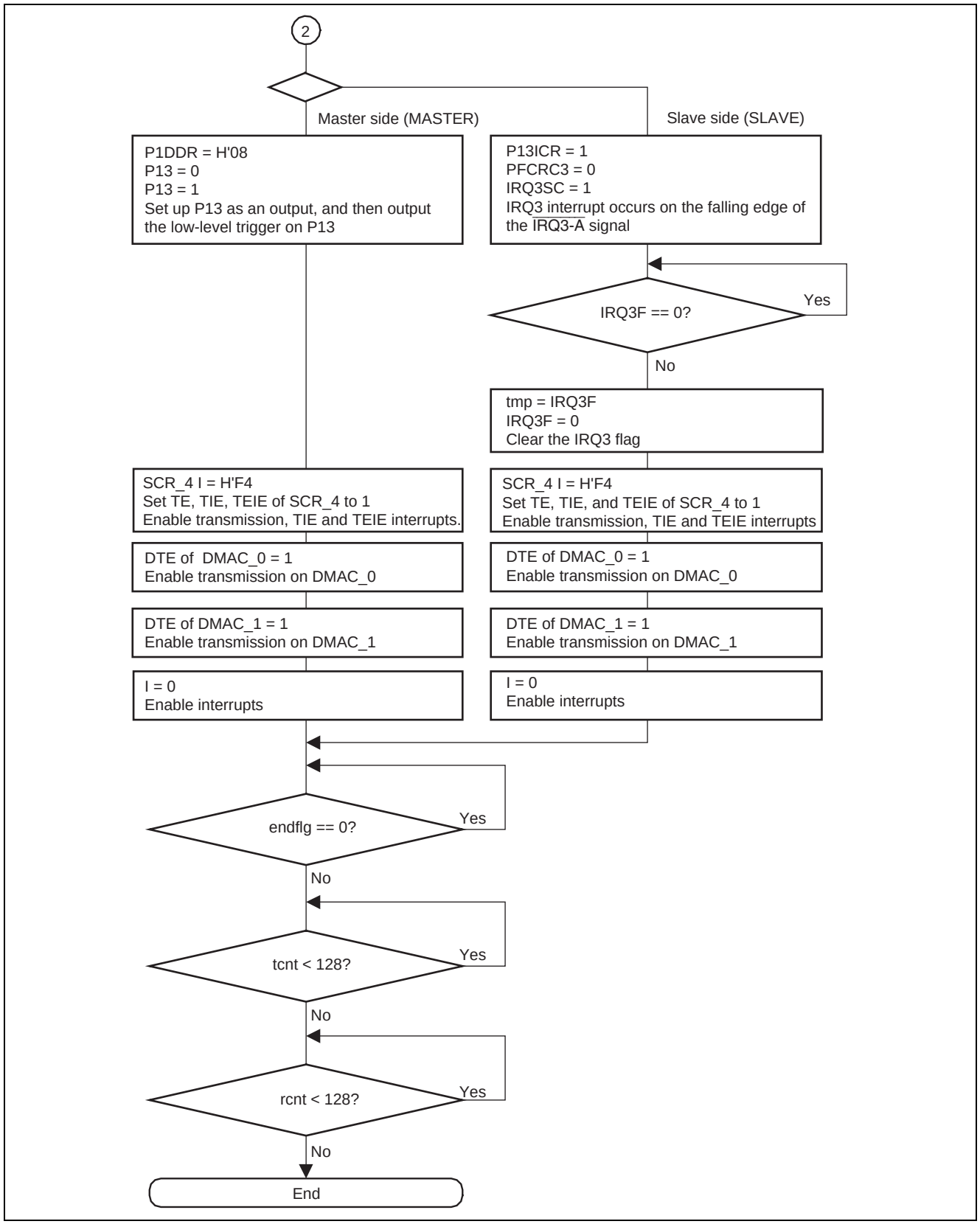
Bit	Bit Name	Setting	R/W	Description
4	ABCS	0	R/W	Asynchronous Mode Base Clock Selection (only valid in the asynchronous mode) Selects the base clock for a 1-bit period 0: The base clock has a frequency 16 times the transfer rate. 1: The base clock has a frequency 8 times the transfer rate.
3	ACS3	0	R/W	Asynchronous Clock Source Selection
2	ACS2	1	R/W	Selects the clock source in the asynchronous mode: See table 12
1	ACS1	0	R/W	0011: Selects the average transfer rate of 921.569 kbps specifically for Pϕ = 16 MHz
0	ACS0	0	R/W	Note 1: When the average transfer rate is selected, the base clock is automatically set regardless of the ACS bit in the SEMR_5 register (asynchronous base clock selection). Note 2: The setting only has the desired effect when bits ACS3 to ACS0 are in the asynchronous mode (C/$\overline{A}$ bit of SMR register is 0), and the external clock input is selected (CKE1 bit of SCR register is 1).

Table 12 List of Settings for Asynchronous Clock Source Select

ACS3 to 0	Transfer Rate	Pϕ (MHz)	Functions
0000	(Set by the ABCS bit)	—	The average transfer rate generator is not used.
0001	1/16th of the base clock frequency for average-rate transfer	10.667	Average transfer rate 115.152 kbps
0010	1/8th of the base clock frequency for average-rate transfer	10.667	Average transfer rate 460.606 kbps
0011	1/8th of the base clock frequency for average-rate transfer	16	Average transfer rate 921.569 kbps
		8	Average transfer rate 460.784 kbps
0100	(Set by the ABCS bits)	—	Selects TMR-clock input: compare-match output of TMR provides the base clock for transfer
0101	1/16th of the base clock frequency for average-rate transfer	16	Average transfer rate 115.196 kbps
0110	1/16th of the base clock frequency for average-rate transfer	16	Average transfer rate 460.784 kbps
0111	1/8th of the base clock frequency for average-rate transfer	24	Average transfer rate 720 kbps
1000	1/16th of the base clock frequency for average-rate transfer	24	Average transfer rate 115.132 kbps
1001	1/16th of the base clock frequency for average-rate transfer	24	Average transfer rate 460.526 kbps
		12	Average transfer rate 230.263 kbps
1010	1/8th of the base clock frequency for average-rate transfer	24	Average transfer rate 720 kbps
1011	1/8th of the base clock frequency for average-rate transfer	24	Average transfer rate 921.053 kbps
		12	Average transfer rate 460.526 kbps
1100	1/16th of the base clock frequency for average-rate transfer	32	Average transfer rate 720 kbps

5. Flowchart


main
i = 0
i < 128?
No
Yes
rcv_dt[i] = 0
i++
Clear flags:
stflg = 0
endflg = 0
errbuf = 0
rcnt = 0
tcnt = 0
DMAC0_trs_init()
Initial setting of DMAC_0
DMAC1_rcv_init()
Initial setting of DMAC_1
Clear TE, RE of SCR_5 to 0
P1ICR I = H'20
Set up pin P15 (RxD5) as an input
CKE 1, 0 in SCR_5 = B'1X
External clock input
SMR_5 = H'00
SCMR_5 & = H'F2
Communications format settings:
- Asynchronous mode
- Data length: 8 bits
- No parity bit
- 1 stop bit
SEMR_5 = H'04
Set timer clock input
1
1
TCR_4 = H'0B
TCCR_4 = H'02
Clear TCNT_4 on matches with TCORA_4
Count on rising edges of Pϕ
TCNT_4 = H'00
Clear timer counter_4
TCORA_4 = H'01
TCORB_4 = H'00
Select base clock output from TMO4
TCSR_4 = H'09
Output 0 on matches with TCORA_4
Output 1 on matches with TCORB_4
TCR_5 = H'0C
TCCR_5 = H'00
Clear TCNT_5 on matches with TCORA_5
Count on matches with TCORA_4
TCNT_5 = H'00
Clear timer counter_5
TCORA_5 = H'03
TCORB_5 = H'00
Generate clock-enable output from TMO5
TCSR_5 = H'19
Output 0 on matches with TCORA_5
Output 1 on matches with TCORB_5
When Pϕ = 16 MHz,
375 kbps
i = 0
i++
Wait 1-bit period
i < 50?
No
Yes
SSR_5 = H'87
Clear RDRF, ORER, PER, FER of SSR_5 to 0
2


2
Master side (MASTER)
Slave side (SLAVE)
P1DDR = H'08
P13 = 0
P13 = 1
Set up P13 as an output, and then output the low-level trigger on P13
P13ICR = 1
PFCRC3 = 0
IRQ3SC = 1
IRQ3 interrupt occurs on the falling edge of the IRQ3-A signal
IRQ3F == 0?
Yes
No
tmp = IRQ3F
IRQ3F = 0
Clear the IRQ3 flag
SCR_4 I = H'F4
Set TE, TIE, TEIE of SCR_4 to 1
Enable transmission, TIE and TEIE interrupts.
SCR_4 I = H'F4
Set TE, TIE, and TEIE of SCR_4 to 1
Enable transmission, TIE and TEIE interrupts
DTE of DMAC_0 = 1
Enable transmission on DMAC_0
DTE of DMAC_0 = 1
Enable transmission on DMAC_0
DTE of DMAC_1 = 1
Enable transmission on DMAC_1
DTE of DMAC_1 = 1
Enable transmission on DMAC_1
I = 0
Enable interrupts
I = 0
Enable interrupts
endflg == 0?
Yes
No
tcnt < 128?
Yes
No
rcnt < 128?
Yes
No
End

5.6.3 DMAC0_trs_init Function

1. Overview
   DMAC_0 initial settings.
2. Arguments
   None
3. Return value
   None
4. Description of internal register usage
   Usage of internal registers in this task is described below. The given settings are those used in the task and differ from the initial settings.

- **DMA source address register_0 (DSAR_0) Address: H'FFFC00**
  Function: Specifies the source address for the transfer
  Setting: &trs_dt
- **DMA destination address register_0 (DDAR_0) Address: H'FFFC04**
  Function: Specifies the destination address for the transfer
  Setting: &TDR_5
- **DMA transfer count register_0 (DTCR_0) Address: H'FFFC0C**
  Function: Selects the amount of data to be transferred as 128 bytes
  Setting: 128

- **DMA mode control register_0 (DMDR_0) Address: H'FFFC14**

Bit	Bit Name	Setting	R/W	Description
31	DTE	0	R/W	Data Transfer Enable 0: Disables data transfer 1: Enables data transfer
26	NRD	0	R/W	Next Request Delay 0: Starts accepting the next transfer request after completion of the current transfer 1: Starts accepting the next transfer request one cycle after completion of the current round of transfer
17	ESIF	0	R/(W)*	Transfer Escape Interrupt Flag 0: A transfer escape end interrupt request has not been issued. 1: A transfer escape end interrupt request has been issued.
16	DTIF	0	R/(W)*	Data Transfer Interrupt Flag 0: A transfer end interrupt request has not been issued by the transfer counter. 1: A transfer end interrupt request has been issued by the transfer counter.
15	DTSZ1	0	R/W	Data Access Size 1 and 0
14	DTSZ0	0	R/W	00: Data access size for transfer is in bytes (8 bits).
13	MDS1	0	R/W	Transfer Mode Select 1 and 0
12	MDS0	0	R/W	00: Sets the normal transfer mode
9	ESIE	0	R/W	Transfer Escape interrupt Enable 0: Disables transfer escape interrupt 1: Enables transfer escape interrupt
8	DTIE	0	R/W	Data Transfer End Interrupt Enable 0: Disables transfer end interrupt 1: Enables transfer end
7	DTF1	1	R/W	Data Transfer Factor 1 and 0
6	DTF0	0	R/W	10: DMAC activation source is an on-chip module interrupt.
5	DTA	1	R/W	Data Transfer Acknowledge When DTF 1, 0 = H'10, the DTA bit should be set to 1.

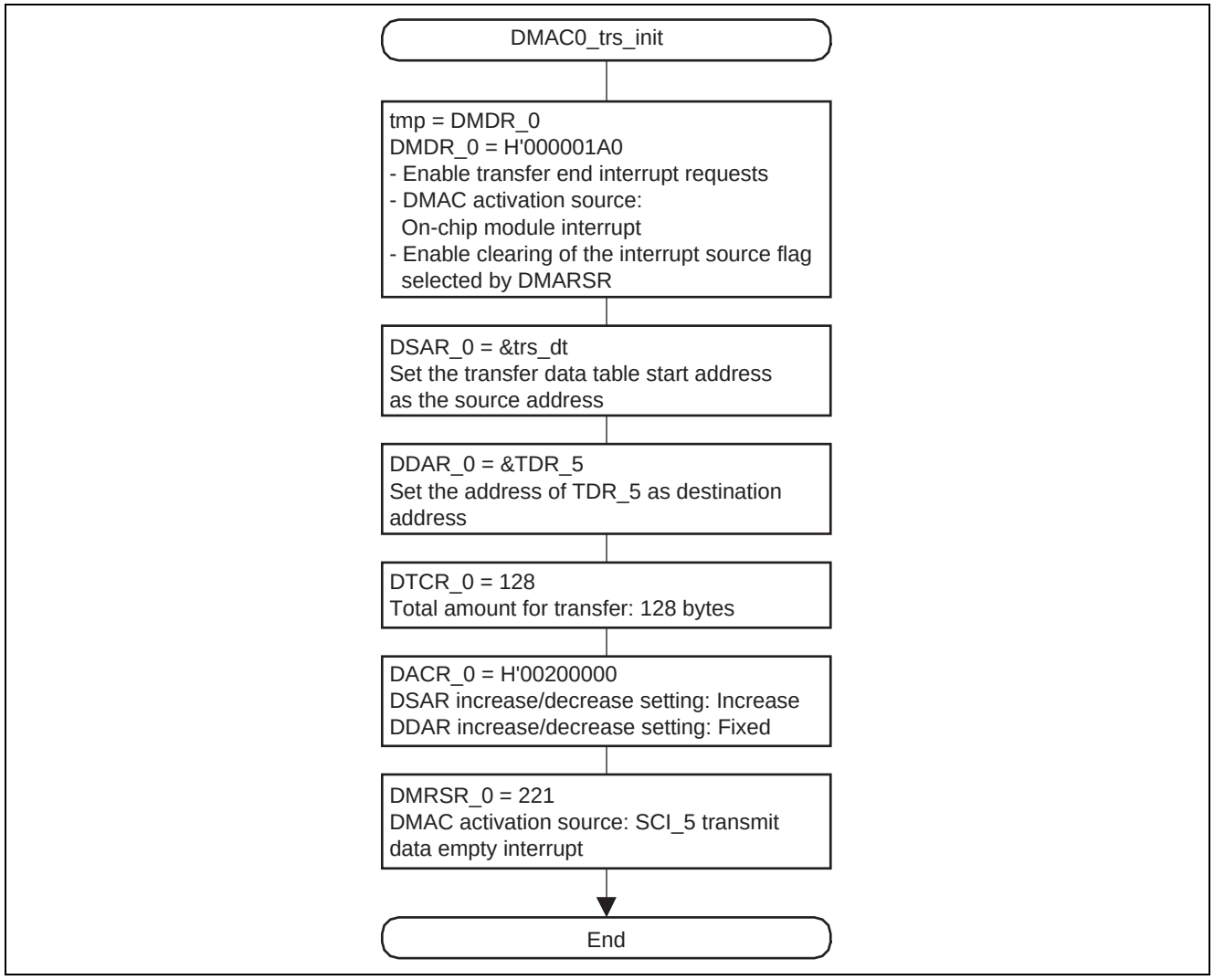
Note: * Only 0 can be written here after having been read as 1, to clear the flag.

- **DMA address control register_0 (DACR_0) Address: H'FFFC18**

Bit	Bit Name	Setting	R/W	Description
31	AMS	0	R/W	Address Mode Select 0: Dual address mode 1: Single address mode
21	SAT1	1	R/W	Source Address Update Mode 1 and 0
20	SAT0	0	R/W	10: Increment the Source address
17	DAT1	0	R/W	Destination Address Update Mode 1 and 0
16	DAT0	0	R/W	00: Destination address is fixed.

- **DMA module request select register_0 (DMRSR_0) Address: H'FFFD20**
  Function: Specifies the source of on-chip module interrupts. The setting 221 corresponds to DMAC activation by SCI_5 transmission data empty interrupts.
  Setting: 221

5. Flowchart



5.6.4 DMAC1_rcv_init Function

1. Overview
   DMAC_1 initialization. Sets up the registers of DMAC channel for the transfer of received data from SCI_5.
2. Arguments
   None
3. Return value
   None
4. Description of internal register usage
   Usage of internal registers in this task is described below. The given settings are those used in the task and differ from the initial settings.

- **DMA source address register_1 (DSAR_1) Address: H'FFFC20**
  Function: Specifies the source address for the transfer
  Setting: &RDR_5
- **DMA destination address register_1 (DDAR_1) Address: H'FFFC24**
  Function: Specifies the destination address for the transfer
  Setting: &rcv_dt
- **DMA transfer count register_1 (DTCR_1) Address: H'FFFC2C**
  Function: Selects the amount of data to be transferred as 128 bytes
  Setting: 128

- **DMA mode control register_1 (DMDR_1) Address: H'FFFC34**

Bit	Bit Name	Setting	R/W	Description
31	DTE	0	R/W	Data Transfer Enable 0: Disables data transfer 1: Enables data transfer
26	NRD	0	R/W	Next Request Delay 0: Starts accepting the next transfer request after completion of the current transfer 1: Starts accepting the next transfer request one cycle after completion of the current transfer
17	ESIF	0	R/(W)*	Transfer Escape Interrupt Flag 0: A transfer escape end interrupt has not been requested. 1: A transfer escape end interrupt has been requested.
16	DTIF	0	R/(W)*	Data Transfer Interrupt Flag 0: A transfer end interrupt by the transfer counter has not been requested. 1: A transfer end interrupt by the transfer counter has been requested.
15	DTSZ1	0	R/W	Data Access Size 1, 0
14	DTSZ0	0	R/W	00: Data access size for transfer is bytes (8 bits).
13	MDS1	0	R/W	Transfer Mode Select 1, 0
12	MDS0	0	R/W	00: Normal transfer mode setting
9	ESIE	0	R/W	Transfer Escape interrupt Enable 0: Disables transfer escape end interrupts 1: Enables Transfer escape interrupt requests
8	DTIE	0	R/W	Data Transfer End Interrupt Enable 0: Disables transfer end interrupts 1: Enables transfer end interrupts
7	DTF1	1	R/W	Data Transfer Factor 1, 0
6	DTF0	0	R/W	10: DMAC activation source is an on-chip module interrupt.
5	DTA	1	R/W	Data Transfer Acknowledge When DTF 1, 0 = H'10, the DTA bit is set to 1.

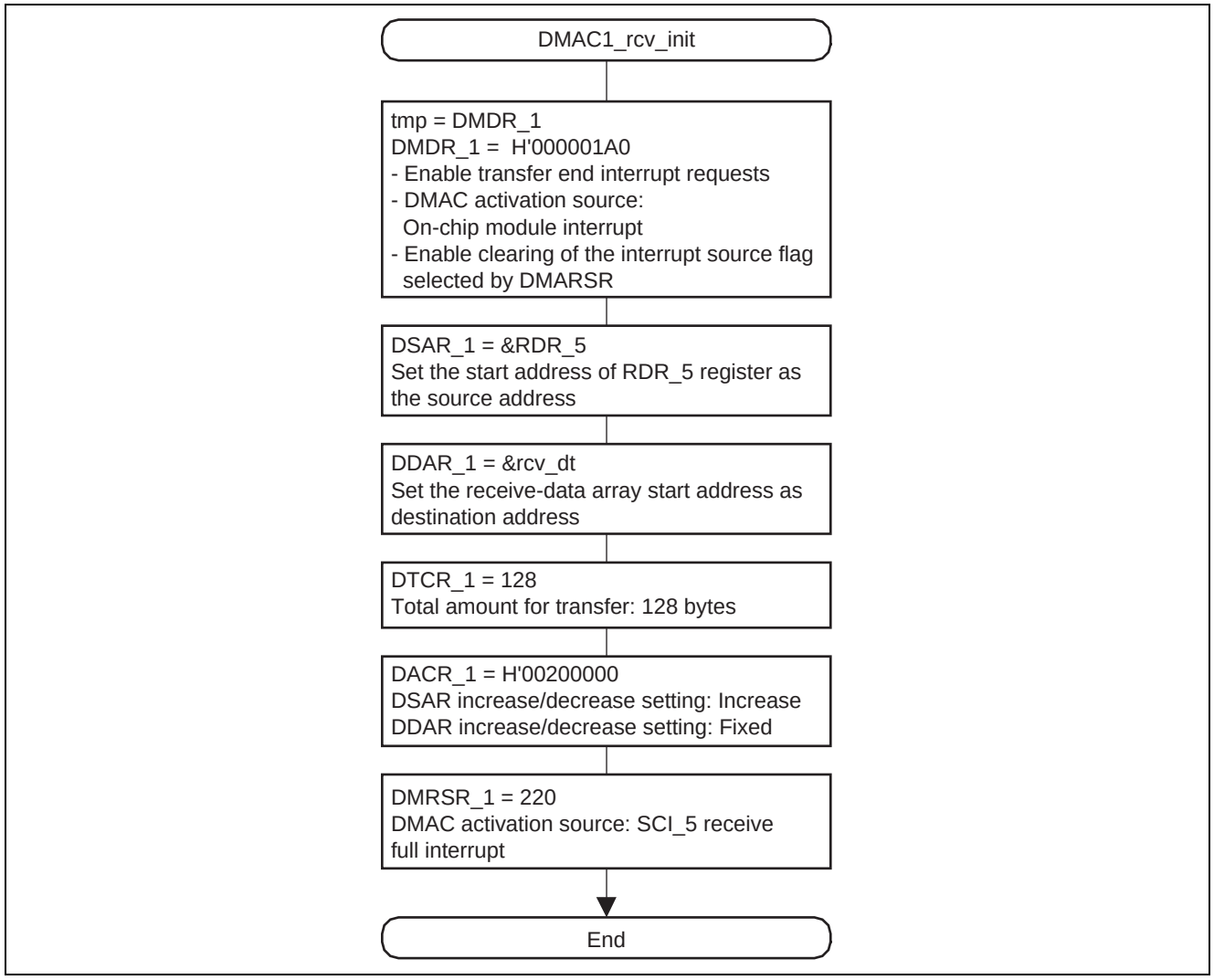
Note: * Only 0 can be written here after having been read as 1, to clear the flag.

- **DMA address control register_1 (DACR_1) Address: H'FFFC38**

Bit	Bit Name	Setting	R/W	Description
31	AMS	0	R/W	Address Mode Select 0: Dual address mode 1: Single address mode
21	SAT1	1	R/W	Source Address Update Mode 1, 0
20	SAT0	0	R/W	00: Source address is fixed.
17	DAT1	0	R/W	Destination Address Update Mode 1, 0
16	DAT0	0	R/W	10: Destination address is updated by with an offset.

- **DMA module request select register_1 (DMRSR_1) Address: H'FFFD21**
  Function: Specifies the source of on-chip module interrupts. The setting 220 corresponds to DMAC activation by the SCI_5 received data full interrupts.
  Setting: 220

5. Flowchart



5.6.5 dmtend0_int Function

1. Overview
   Handler for the DMAC_0 transfer end interrupt. Stops the SCI transmission processing.
2. Arguments
   None
3. Return value
   None
4. Description of internal register usage
   Usage of internal registers in this task is described below. The given settings are those used in the task and differ from the initial settings.

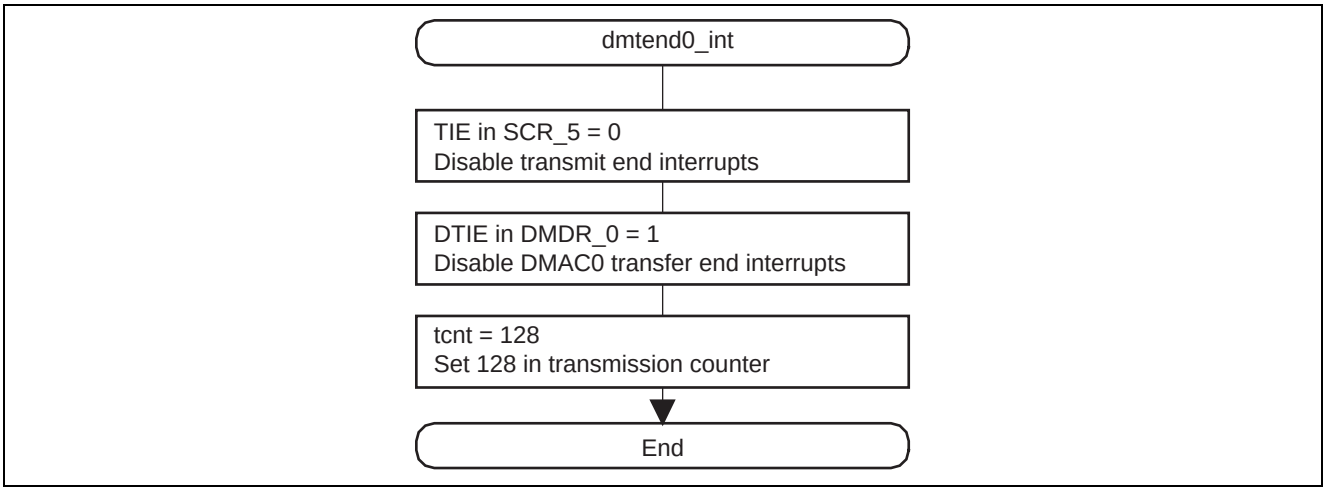
- **Serial control register_5 (SCR_5) Address: H'FFF602**

Bit	Bit Name	Setting	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable 0: Disables TXI interrupt requests 1: Enables TXI interrupt requests

- **DMA mode control register_0 (DMDR_0) Address: H'FFFC14**

Bit	Bit Name	Setting	R/W	Description
8	DTIE	0	R/W	Data Transfer Interrupt Enable 0: Disables transfer end interrupt requests 1: Enables transfer end interrupt requests

5. Flowchart



5.6.6 dmtend1_int Function

1. Overview
   Handler for the DMAC_1 transfer end interrupt. Stops the SCI reception processing.
2. Arguments
   None
3. Return value
   None
4. Description of internal register usage
   Usage of internal registers in this task is described below. The given settings are those used in the task and differ from the initial settings.

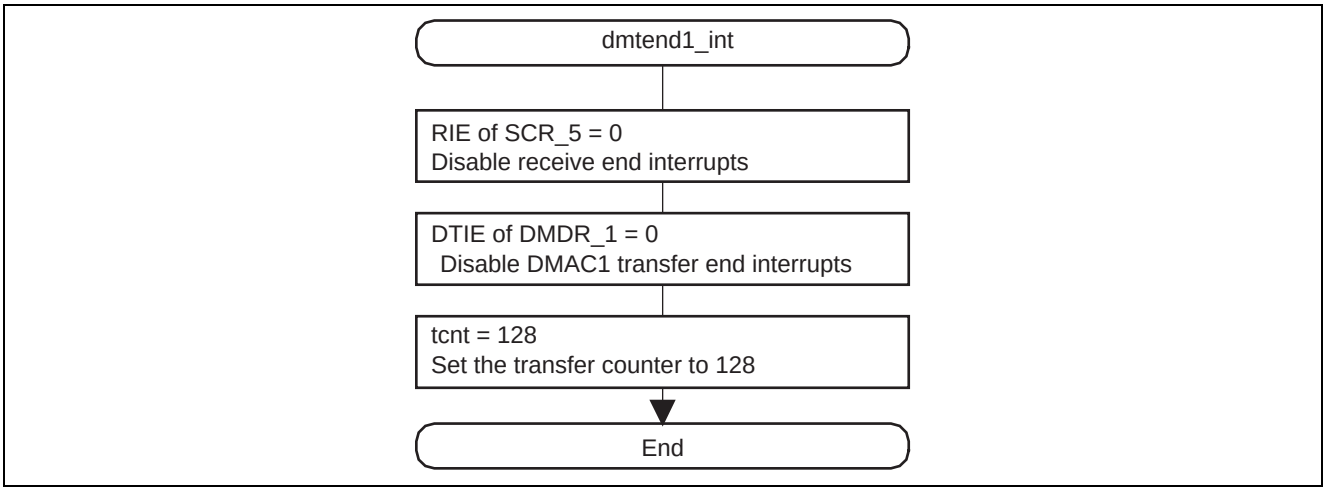
- **Serial control register_5 (SCR_5) Address: H'FFF602**

Bit	Bit Name	Setting	R/W	Description
6	RIE	0	R/W	Receive Interrupt Enable 0: Disables RXI and ERI interrupt requests 1: Enables RXI and ERI interrupt requests

- **DMA mode control register_1 (DMDR_1) Address: H'FFFC34**

Bit	Bit Name	Setting	R/W	Description
8	DTIE	0	R/W	Data Transfer Interrupt Enable 0: Disables transfer end interrupt requests 1: Enables transfer end interrupt requests

5. Flowchart



5.6.7 eri5_int Function

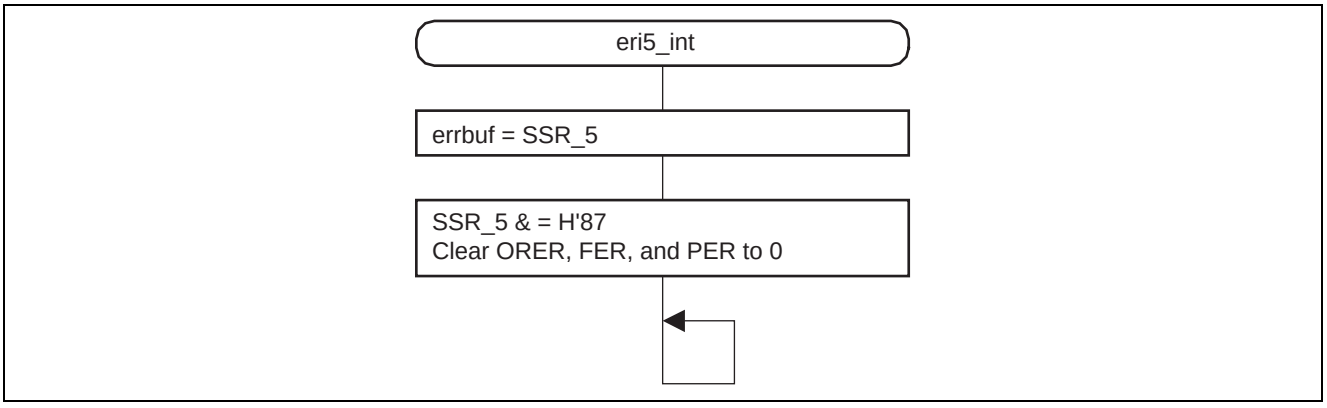
1. Overview
   Handler for reception error interrupts. Transfers one byte.
2. Arguments
   None
3. Return value
   None
4. Description of internal register usage
   Usage of internal registers in this task is described below. The given settings are those used in the task and differ from the initial settings.

- **Serial status register_5 (SSR_5) Address: H'FFF604**

Bit	Bit Name	Setting	R/W	Description
5	ORER	0	R/(W)*	Overrun Error [Setting condition] • Occurrence of an overrun error during reception [Clearing condition] • Writing of 0 to ORER after having read ORER = 1
4	FER	0	R/(W)*	Framing Error [Setting condition] • Occurrence of a framing error during reception [Clearing condition] • Writing of 0 to FER after having read FER = 1
3	PER	0	R/(W)*	Parity Error [Setting condition] • Occurrence of a parity error during reception [Clearing condition] • Writing of 0 to PER after having read PER = 1

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

5. Flowchart



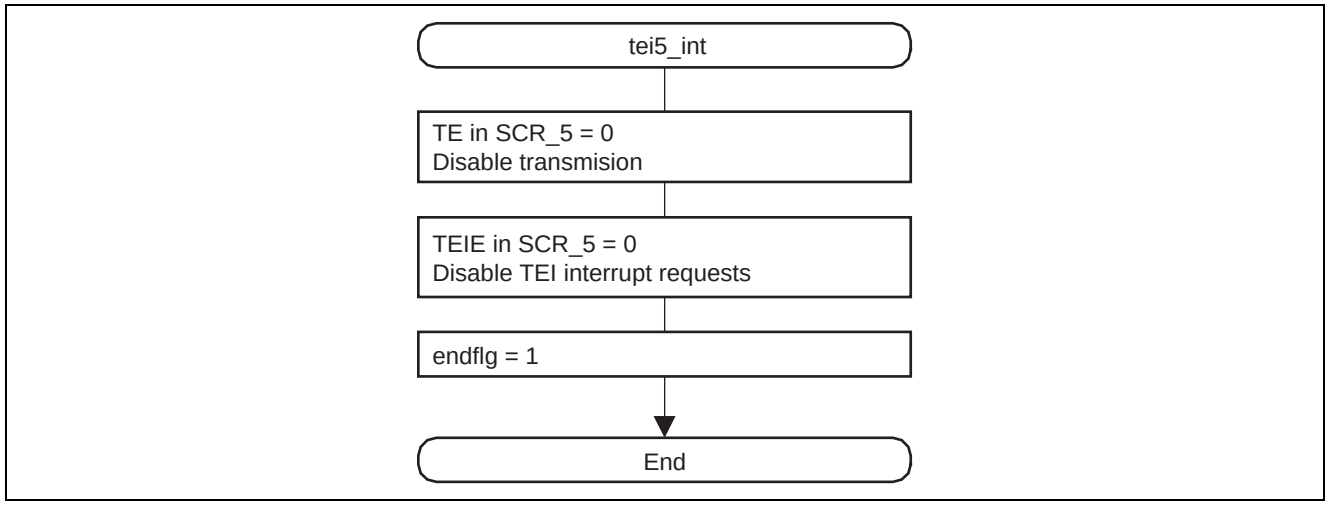
5.6.8 tei5_int Function

1. Overview
   Handler for the transmission end interrupt function. Transfers one byte.
2. Arguments
   None
3. Return value
   None
4. Description of internal register usage
   Usage of internal registers in this task is described below. The given settings are those used in the task and differ from the initial settings.

- **Serial control register_5 (SCR_5) Address: H'FFF602**

Bit	Bit Name	Setting	R/W	Description
5	TE	0	R/W	Transmit Enable 0: Disables transmission 1: Enables transmission
2	TEIE	0	R/W	Transmit End Interrupt Enable 0: Disables TEI interrupt requests 1: Enables TEI interrupt requests

5. Flowchart



6. Note on Usage

When the pin of the device functions as an input for the peripheral modules, the corresponding bits of the input buffer control register (PnICR) should be set to 1. For details, see the hardware manual.

Website and Support

Renesas Technology Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/inquiry
csc@renesas.com

Revision Record

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Mar.10.06	—	First edition issued
2.00	Mar.07.08	1, 42	Page 1: Target devices added Page 42: Note on usage added

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (http://www.renesas.com)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
   (1) artificial life support devices or systems
   (2) surgical implantations
   (3) healthcare intervention (e.g., excision, administration of medication, etc.)
   (4) any other purposes that pose a direct threat to human life

   Renesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.