SH7136/SH7137 Group

SCI Clock Synchronous Simultaneous Transmit and Receive of Serial Data and DTC Data Transfer

Introduction

This application note describes an operation example of the clock synchronous serial transmit and receive functions of the serial communication interface (SCI), using the data transfer controller (DTC) of the SH7137.

Target Device

SH7137

Contents

1. Introduction

1.1 Specifications

This application note describes simultaneous clock synchronous transmit and receive of serial data by the serial communication interface (SCI), using the data transfer functions of the data transfer controller (DTC). Figure 1 shows the configuration.

- SCI channel 0 and the DTC are used.
- The SCI communication format is 8-bit fixed.
- The DTC is used to transfer the SCI transmit and receive data. The DTC uses two channels, one for SCI transmit and one for SCI receive.
- The DTC is activated for transmit operation by an SCI transmit data empty interrupt request. The DTC is activated for receive operation by an SCI receive data full interrupt request.
- The SCI transmit and receive data count is 32 bytes.

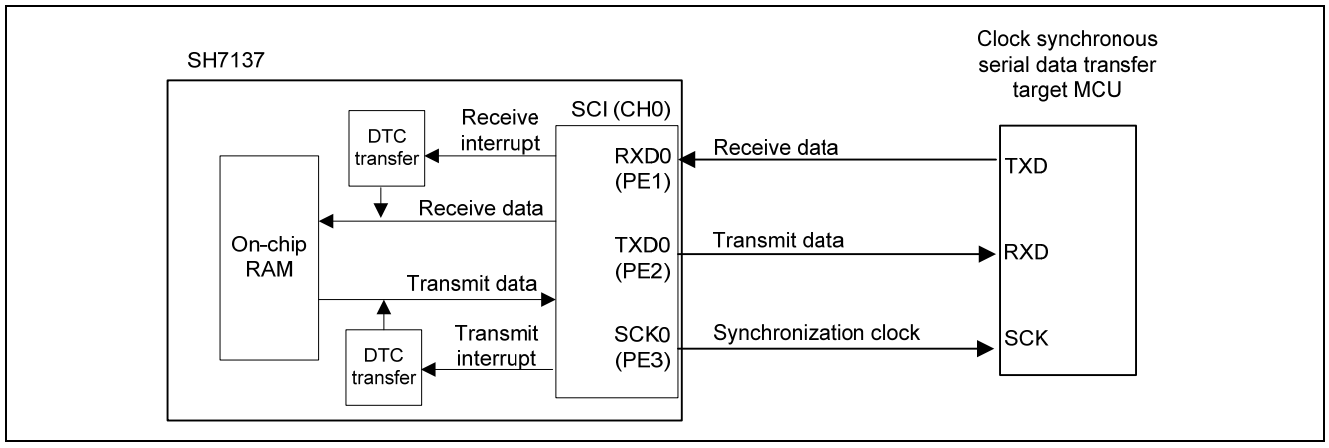


Figure 1 Example of Clock Synchronous Serial Data Transmit/Receive Using DTC

1.2 Module Used

- Data transfer controller (DTC)
- Serial communication interface (SCI) channel 0

1.3 Applicable Conditions

MCU:	SH7137
Operating frequencies:	Internal clock (Iϕ) = 80 MHz
	Bus clock (Bϕ) = 40 MHz
	Peripheral clock (Pϕ) = 40 MHz
	MTU2 clock (MPϕ) = 40 MHz
	MTU2S clock (MIϕ) = 80 MHz
C compiler:	Renesas Technology High-performance Embedded Workshop, Ver. 4.05.01.001, Renesas Technology SuperH RISC engine Family C/C++ Compiler Package, Ver. 9.03, Release 00
Compile options:	High-performance Embedded Workshop default settings

2. Description of the Sample Application

In this application example, the transmit data empty interrupt (TXI) and receive data full interrupt (RXI) of the serial communication interface (SCI) are used to activate the data transfer controller (DTC) and perform clock synchronous serial data simultaneous transmit and receive. The DTC is set to use two channels, one for SCI transmit and one for SCI receive. The DTC's normal transfer mode is used.

2.1 Operational Overview of Module Used

2.1.1 Serial Communication Interface (SCI)

The clock synchronous mode enables data transmit and receive operation in synchronization with a clock pulse, making it suitable for high-speed serial communication. Either an internal clock or an external clock input via the SCK pin may be selected as the clock source. When an internal clock is selected, the synchronization clock is output by the SCK pin. When an external clock is selected, the synchronization clock is input to the SCK pin.

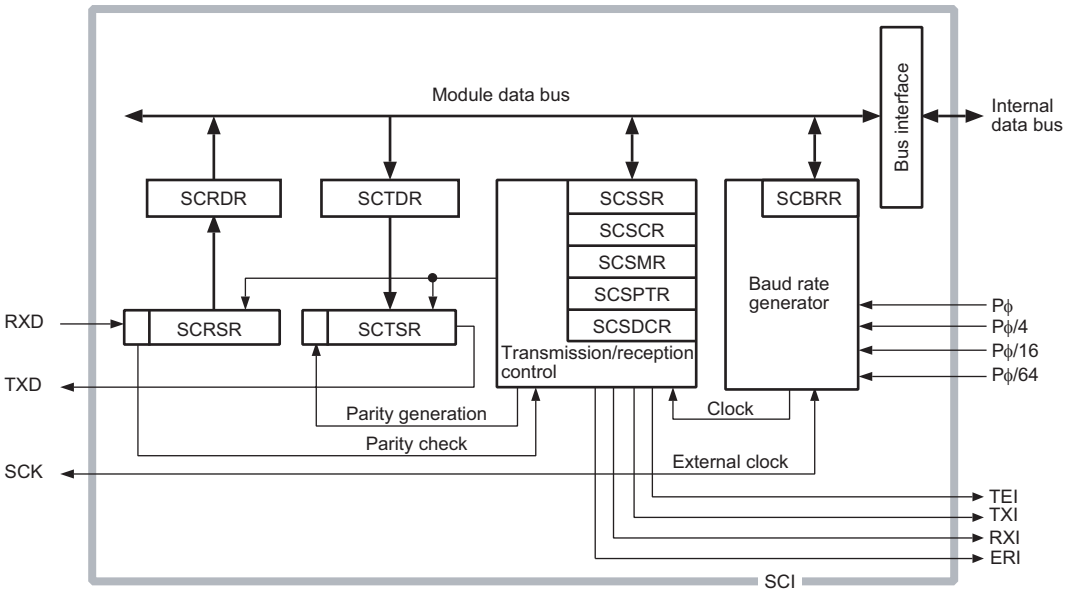
Internally, the SCI has independent transmit and receive blocks, and full-duplex communication is possible by using a common synchronization clock. The transmit and receive blocks each have a double-buffered configuration, so data can be read or written during transmission or reception, enabling continuous data transfer.

For details of the SCI, see the Serial Communication Interface (SCI) section in the SH7137 Group Hardware Manual (RJJ09B0392).

Table 1 shows an overview of clock synchronous communication. Figure 2 is a block diagram of the SCI.

Table 1 Overview of Clock Synchronous Serial Communication

Item	Description
Number of channels	3 channels (SCI_0, SCI_1, SCI_2)
Clock sources	• Internal clock: Pϕ, Pϕ/4, Pϕ/16, Pϕ/64 (Pϕ: peripheral clock) • External clock: Clock input on SCK pin
Data format	• Transfer data length: 8-bit data fixed • Transfer sequence: Selectable between LSB-first and MSB-first
Baud rate	• When internal clock selected: 250 bps to 5,000,000 bps (Pϕ = 40 MHz operation) • When external clock selected: Max. 6,666,666.7 bps (Pϕ = 40 MHz and external clock input = 6.6667 MHz operation)
Receive error detection	Overrun error
Interrupt requests	• Transmit data empty interrupt (TXI) • Transmit end interrupt (TEI) • Receive data full interrupt (RXI) • Receive error interrupt (ERI)
Clock source	• Selectable between internal clock and external clock • When internal clock selected: Clock of SCI's internal on-chip baud rate generator is used for operation. Synchronization clock is output on SCK pin. • When external clock selected: On-chip baud rate generator is not used. External synchronization clock input via SCK pin is used for operation.



[Legend]
SCRSR: Receive shift register
SCRDR: Receive data register
SCTSR: Transmit shift register
SCTDR: Transmit data register
SCSMR: Serial mode register
SCSCR: Serial control register
SCSSR: Serial status register
SCBRR: Bit rate register
SCSPTR: Serial port register
SCSDCR: Serial direction control register

Figure 2 SCI Block Diagram

2.1.2 Data Transfer Controller (DTC)

The data transfer controller (DTC) can be activated by interrupt requests from on-chip peripheral modules to perform data transfers.

The DTC has three transfer modes: normal transfer mode, repeat transfer mode, and block transfer mode. By storing transfer information in a data area, data transfer can be performed using a user-specified number of channels.

When the DTC is activated, the transfer information is read from the data area, data transfer starts, and then updated transfer information is written back to the data area after the end of the data transfer. The transfer information can be assigned to a data area in the on-chip RAM or in an external memory space.

For details of the DTC, see the Data Transfer Controller (DTC) section in the SH7137 Group Hardware Manual (RJJ09B0392).

Table 2 shows an overview of the DTC. Figure 3 is a block diagram of the DTC.

Table 2 Overview of DTC

Item	Description
Transfer modes	• Three transfer modes • Normal transfer mode • Repeat transfer mode • Block transfer mode
Transfer count	• Normal transfer mode: 1 to 65,536 • Repeat transfer mode: 1 to 256 • Block transfer mode: 1 to 65,536
Data size	The data size for data transfers may be set to byte, word, or longword.
CPU interrupt requests	• An interrupt request can be set to the CPU at the end of a single data transfer. • An interrupt request can be set to the CPU at the end of a specified number of data transfers.
Others	• Support for chain transfer (multiple data transfers triggered by a single activation source) • Support for transfer information read skip mode • Support for skipping write-back for fixed transfer source addresses and transfer destination addresses • Support for module stop mode • Support for short address mode • Selectable among five bus right release timings • Selectable between two priorities at DTC startup

Note: An on-chip peripheral module should be set as, at a minimum, either the transfer source or transfer destination. The DTC cannot be used for transfers among external memory, memory mapped external devices, and on-chip memory only.

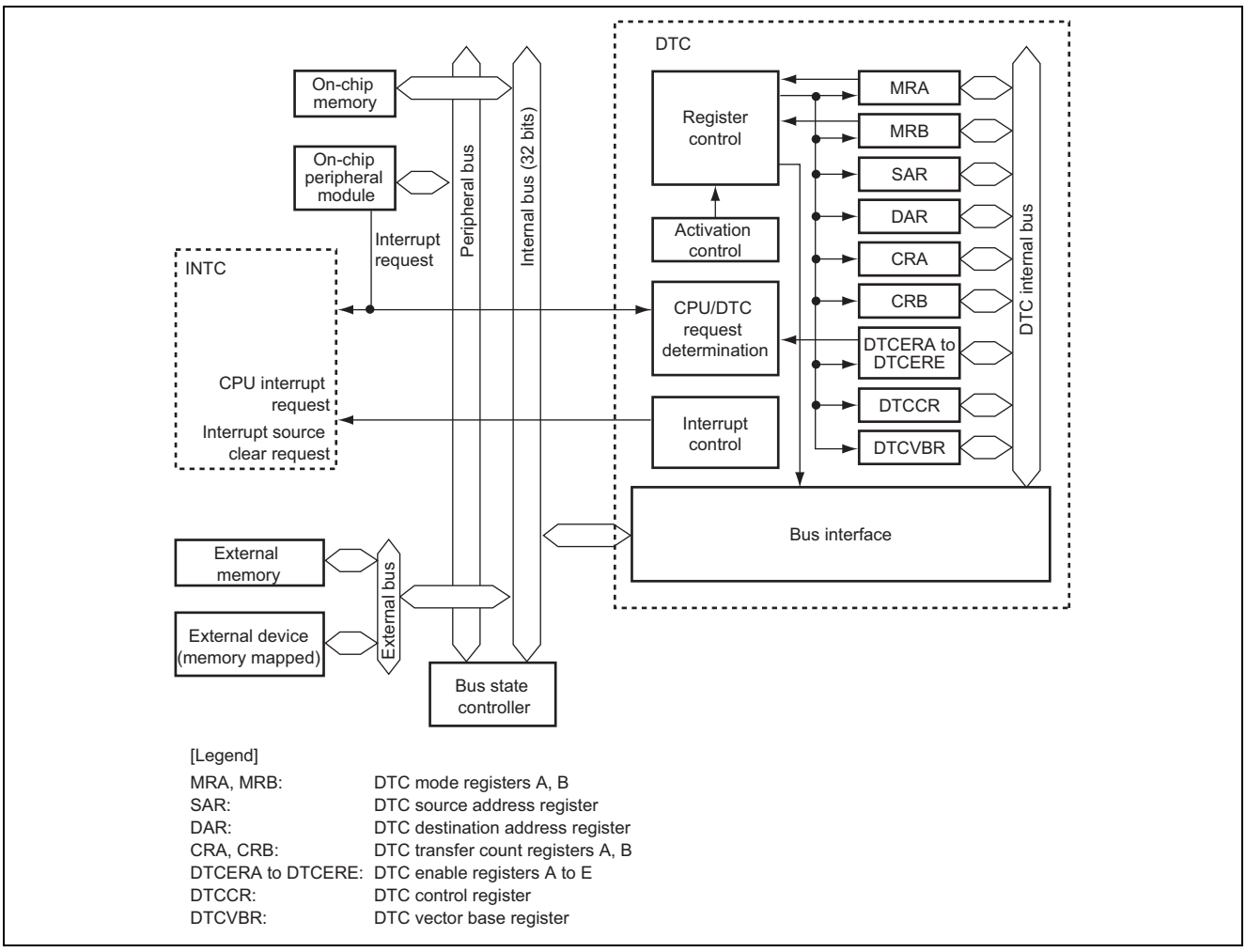


Figure 3 DTC Block Diagram

(1) Allocation of Transfer Information

Figure 4 shows the allocation of DTC transfer information in normal mode. The DTC transfer information is assigned to a data area in a location such as on-chip RAM. Use address 4n as the start address for transfer information. If an address other than 4n is specified, the lowest two bits are ignored when access is performed (lowest 2 bits = B'00).

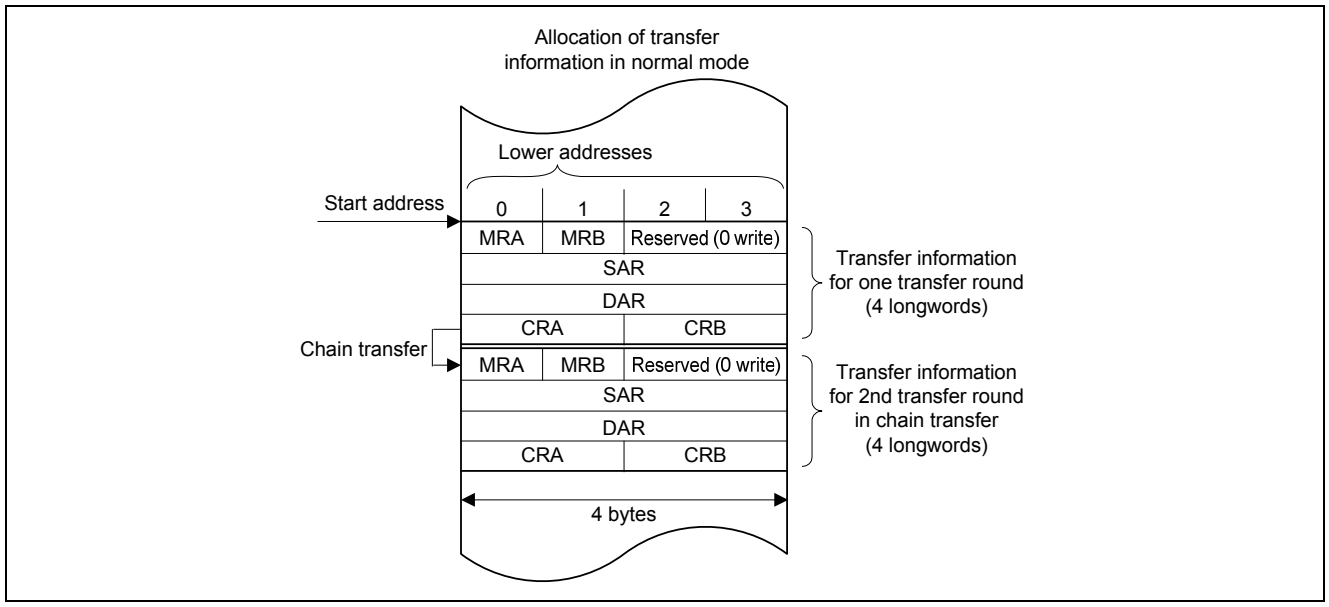


Figure 4 Allocation of Transfer Information in Data Area

(2) Setting the DTC Vector Address

For each activation source, the DTC reads the start address of the transfer information from a vector table, then reads the transfer information from this start address. Figure 5 shows the correspondence between the DTC vector table and the transfer information.

For information on the correspondence between DTC activation sources and vector addresses, see the Data Transfer Controller (DTC) section in the SH7137 Group Hardware Manual (RJJ09B0392).

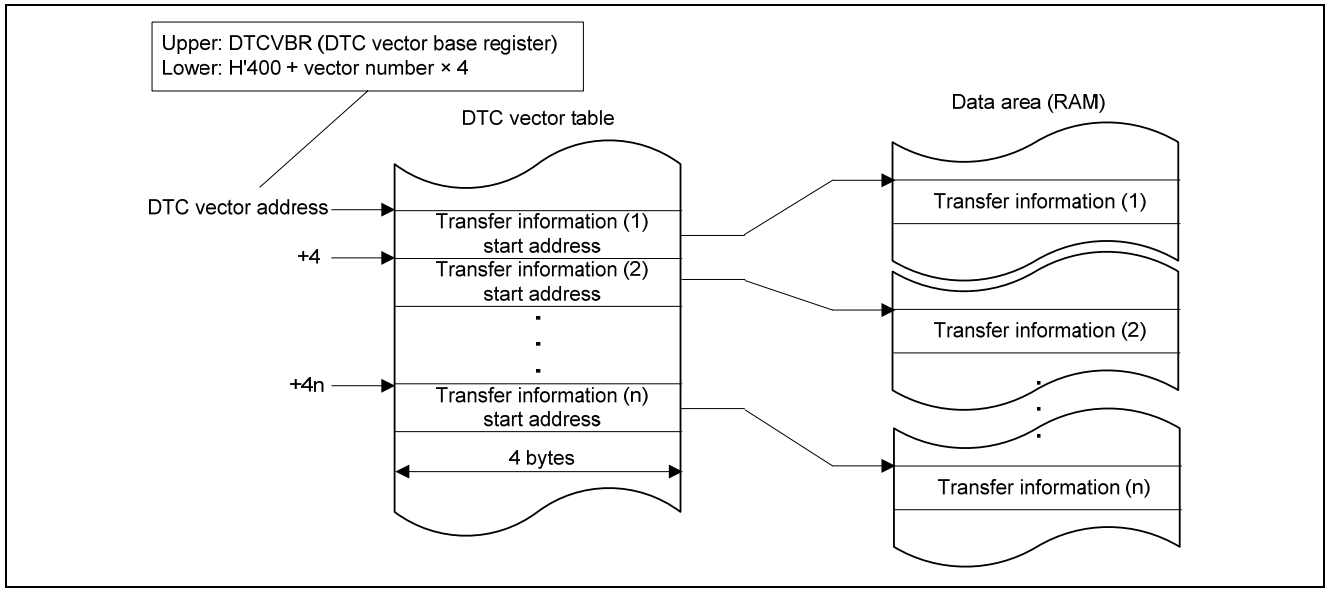


Figure 5 Correspondence between DTC Vector Table and Transfer Information

2.2 Operation of the Sample Program

2.2.1 The Sample Program Operation Specifications

Table 3 lists the SCI communication function settings used in this application note. The SCI performs simultaneous transmit and receive in clock synchronous mode. The DTC transfer function is used to transfer SCI transmit and receive data.

Table 3 SCI Communication Function Settings

Item	Description
Module	SCI channel 0
Communication mode	Clock synchronous mode
Interrupts	• Transmit data empty interrupt (TXI) • Receive data full interrupt (RXI) • Receive error interrupt (ERI)
Communication speed	100 Kbytes
Transmit/receive data count	32 bytes
Data length	8-bit data (fixed)
Bit sequence	LSB-first
Synchronization clock	Internal clock/synchronization clock output on SCK pin

Table 4 lists the DTC transfer conditions for this application note. The DTC is set to two channels, one for SCI transmit and one for SCI receive.

Table 4 DTC Transfer Conditions

	Description	
Item	SCI transmit side DTC transfer conditions (TXI_0)	SCI receive side DTC transfer conditions (RXI_0)
Transfer mode	Normal mode	Normal mode
Transfer count	32 times	32 times
Transfer size	Byte transfer	Byte transfer
Transfer source	On-chip RAM (SCI transmit data storage area)	SCI receive data register (SCRDR_0)
Transfer destination	SCI transmit data register (SCTDR_0)	On-chip RAM (SCI receive data storage area)
Transfer source address	Transfer source address is incremented following transfer.	Transfer source is fixed.
Transfer destination address	Transfer source is fixed.	Transfer destination address is incremented following transfer.
Activation source	DTC is activated at SCI channel 0 transmit data empty interrupt (TXI) request.	DTC is activated at SCI channel 0 receive data full interrupt (TXI) request.
Interrupt handling	Interrupt processing by the CPU (SCI TXI interrupt) is enabled following the completion of the specified data transfer count.	Interrupt processing by the CPU (SCI RXI interrupt) is enabled following the completion of the specified data transfer count.

2.2.2 Allocation of DTC Transfer Information

Figure 6 shows the allocation of DTC transfer information in memory.

In this application note, the address H'FFFF8000 is set as the DTC vector base register (DTCVBR) and the vector table is allocated to an area in the on-chip RAM.

The DTC transfer information is allocated to an area in the on-chip RAM. The SCI receive DTC transfer information is allocated to address H'FFF8800, and the SCI transmit DTC transfer information is allocated to address H'FFF8810.

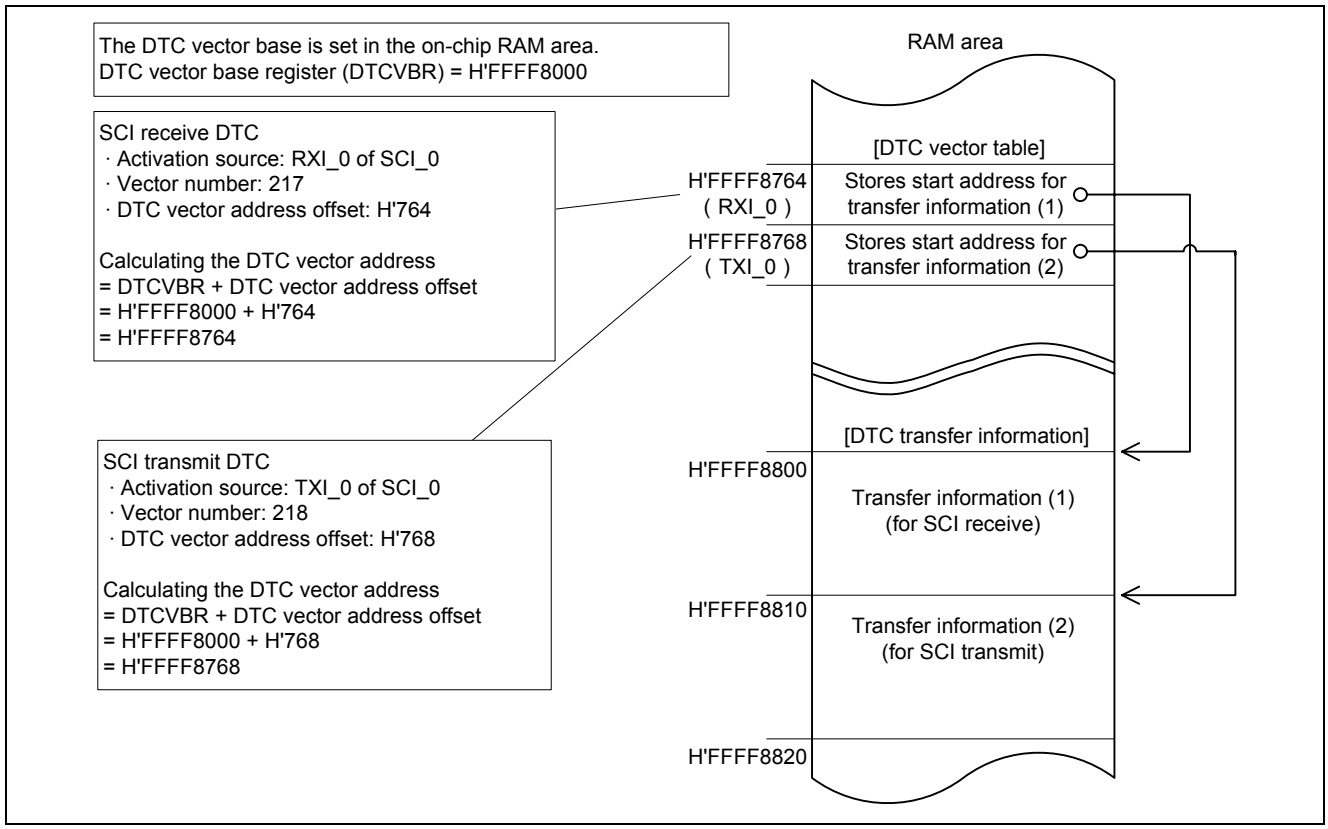


Figure 6 Allocation of DTC Transfer Information in Memory

2.2.3 Operation Description

Figure 7 provides an operation description. The transmit and receive operation setting bits (bits TE and RE) for SCI channel 0 are set to 1 simultaneously to start transmit and receive operation.

In transmit operation, when 1 byte of data is ready to be transmitted, the TDRE flag is set to 1, a TXI interrupt request is generated, and the DTC is activated. The DTC transfers the transmit data from the on-chip RAM to the SCI, and the TDRE flag is automatically cleared to 0. During this time the CPU processes no interrupts. After the specified transfer count of 32 DTC data transfers is completed, the TDRE flag remains set to 1 and a TXI interrupt is issued to the CPU. The TDRE flag is cleared to 0 by the interrupt handling routine.

In receive operation, when reception of 1 byte of data finishes, the RDRF flag is set to 1, an RXI interrupt request is generated, and the DTC is activated. The DTC transfers the receive data to the on-chip RAM, and the RDRF flag is automatically cleared to 0. During this time the CPU processes no interrupts. After the specified transfer count of 32 DTC data transfers is completed, the RDRF flag remains set to 1 and an RXI interrupt is issued to the CPU. The RDRF flag is cleared to 0 by the interrupt handling routine.

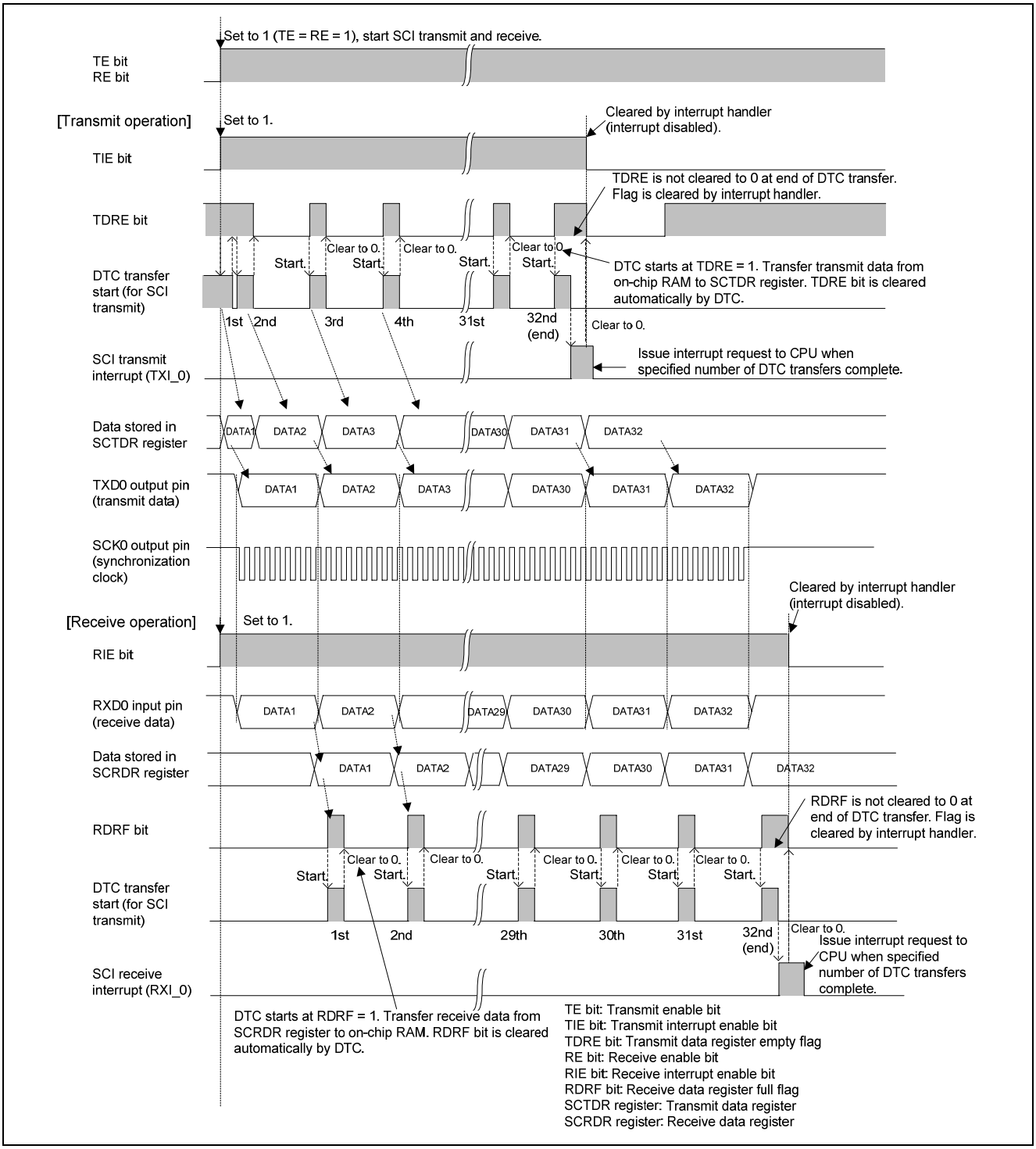


Figure 7 Operation Description

2.3 Configuration of the Sample Program

2.3.1 Description of Functions

Table 5 lists the modules used in the reference program.

Table 5 Functions Used

Function Name	Label	Description
Main	main ()	Makes initial settings for each module. Makes initial settings for the data transfer controller (DTC) and serial communication interface (SCI). Enables SCI simultaneous transmit and receive operation.
Module standby setting	stbcr_init ()	Cancels module standby settings (SCI ch0, DTC).
DTC initial setting	dtc_init()	Makes DTC initial settings for SCI (ch0) transmit and SCI (ch0) receive.
RFC initial setting	pfc_init ()	Makes pin function controller (PFC) initial settings. Sets SCI-related pins to function as serial pins.
RXI0 interrupt	int_sci_rxi()	SCI (ch0) receive data full (RDRF) interrupt (RXI). Generated when DTC data transfer ends.
TXI0 interrupt	int_sci_txi()	SCI (ch0) transmit data empty (TDRE) interrupt (TXI). Generated when DTC data transfer ends.
ERI0 interrupt	int_sci_eri()	Receive error (ORER) interrupt (ERI). Processing when an overrun error occurs.

2.3.2 Variables Usage

Table 6 lists the variables used in the reference program.

Table 6 Variables Usage

Label Name	Description	Name of Employing Module
Rxi_data[32]	Array for storing SCI receive data	main ()
Txi_data[32]	Array for storing SCI transmit data	dtc_init()
DTC_RXI0	Structure variable for storing DTC transfer information settings for SCI receive. Allocated in on-chip RAM.	dtc_init()
DTC_TXI0	Structure variable for storing DTC transfer information settings for SCI transmit. Allocated in on-chip RAM.	
*Dtc_Vect_rxi0	Pointer variable for storing the start address of the DTC transfer information (structure variable DTC_RXI0). Allocated at the DTC vector table address (on-chip RAM) corresponding to DTC activation source RXI_0.	dtc_init()
*Dtc_vect_txi0	Pointer variable for storing the start address of the DTC transfer information (structure variable DTC_TXI0). Allocated at the DTC vector table address (on-chip RAM) corresponding to DTC activation source TXI_0.	

2.4 Procedure for Setting Module Used

The setting procedure for SCI clock synchronous mode using the DTC is described below.

2.4.1 Main Function

Figure 8 shows the processing sequence of the main function.

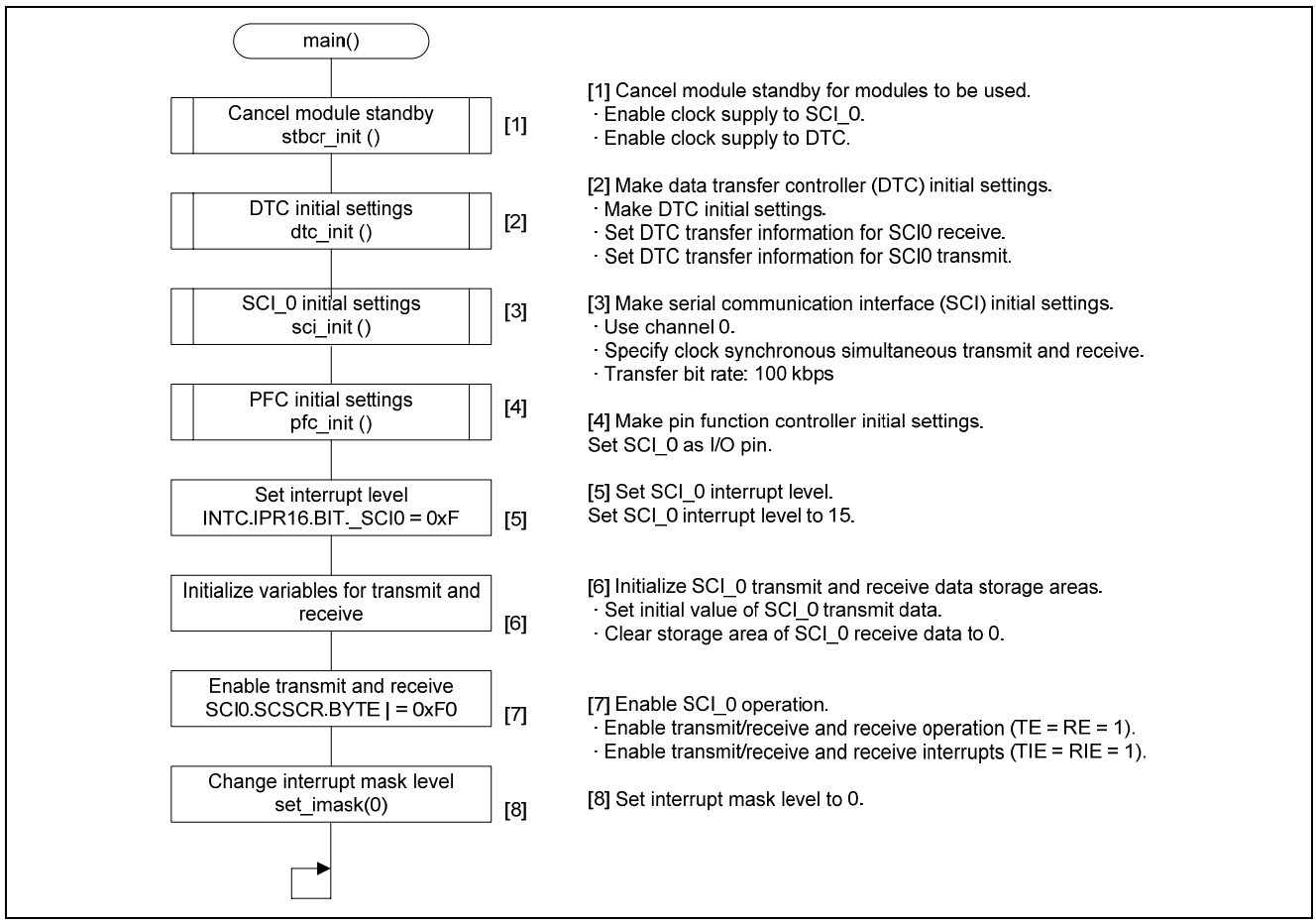


Figure 8 Processing Sequence of Main Function

2.4.2 Cancel Module Standby

Figure 9 shows the processing sequence of the function that cancels module standby.

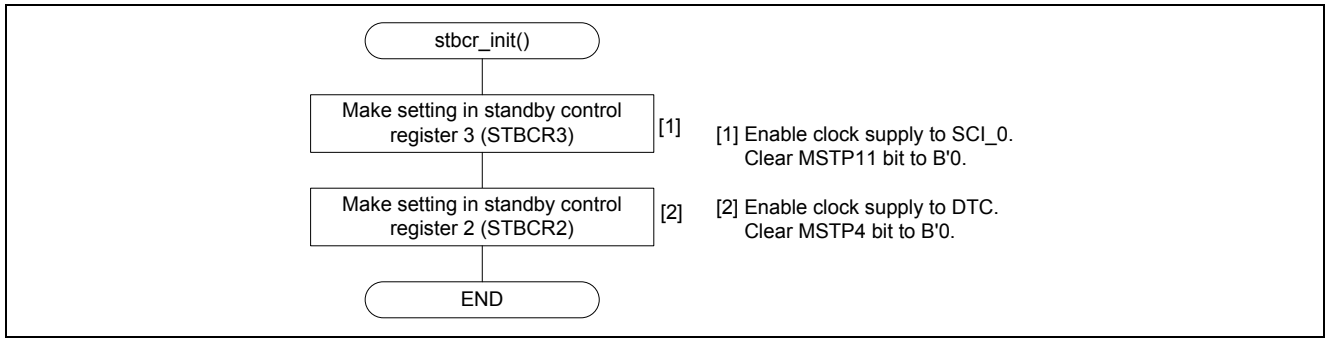


Figure 9 Setting sequence for canceling module standby.

2.4.3 Initialization of Data Transfer Controller (DTC)

Figure 10 shows the initial setting sequence for the data transfer controller (DTC).

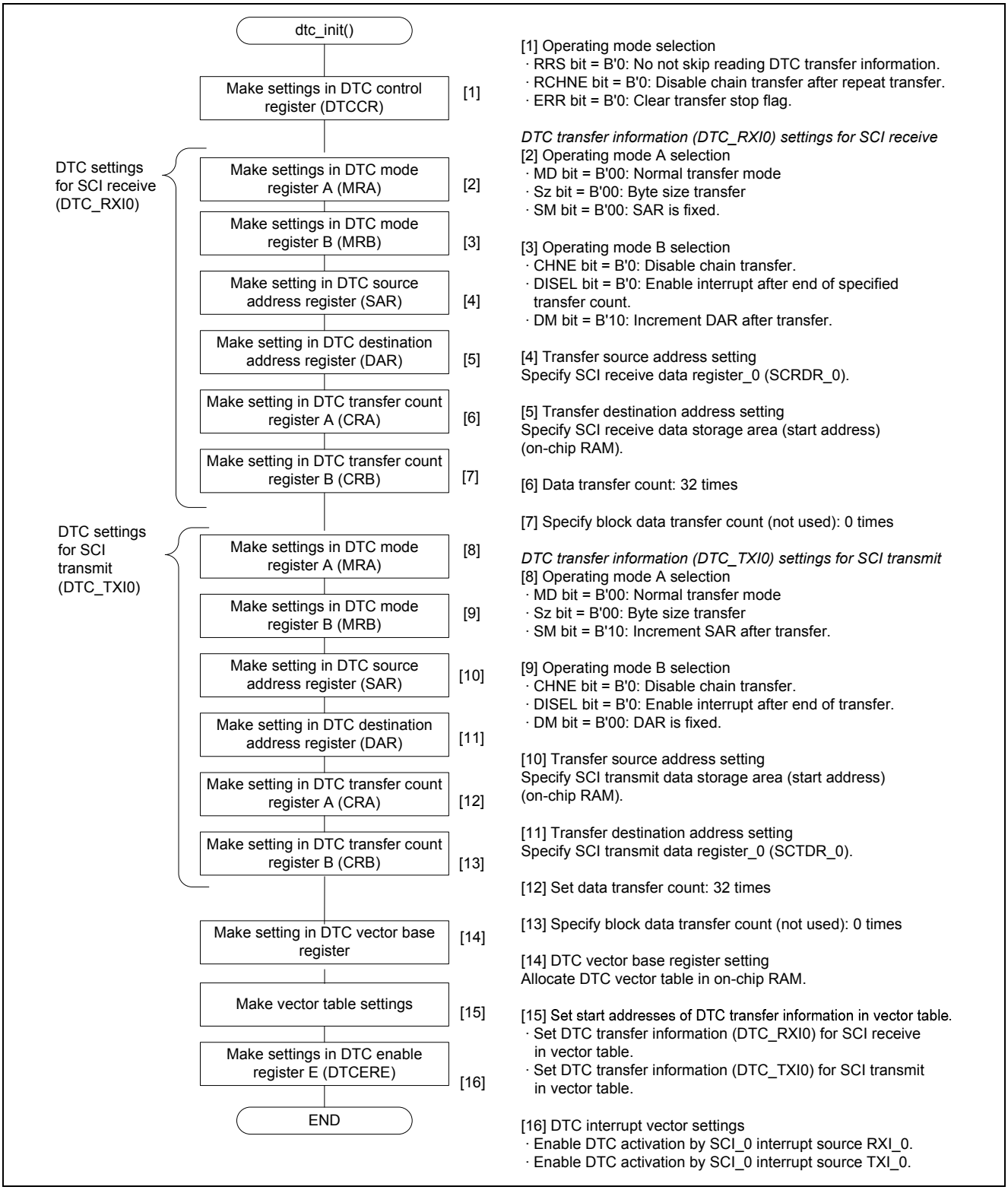


Figure 10 Initialization of Data Transfer Controller (DTC)

2.4.4 Initialization of Serial Communication Interface (SCI)

Figure 11 shows the initial setting sequence for the serial communication interface (SCI).

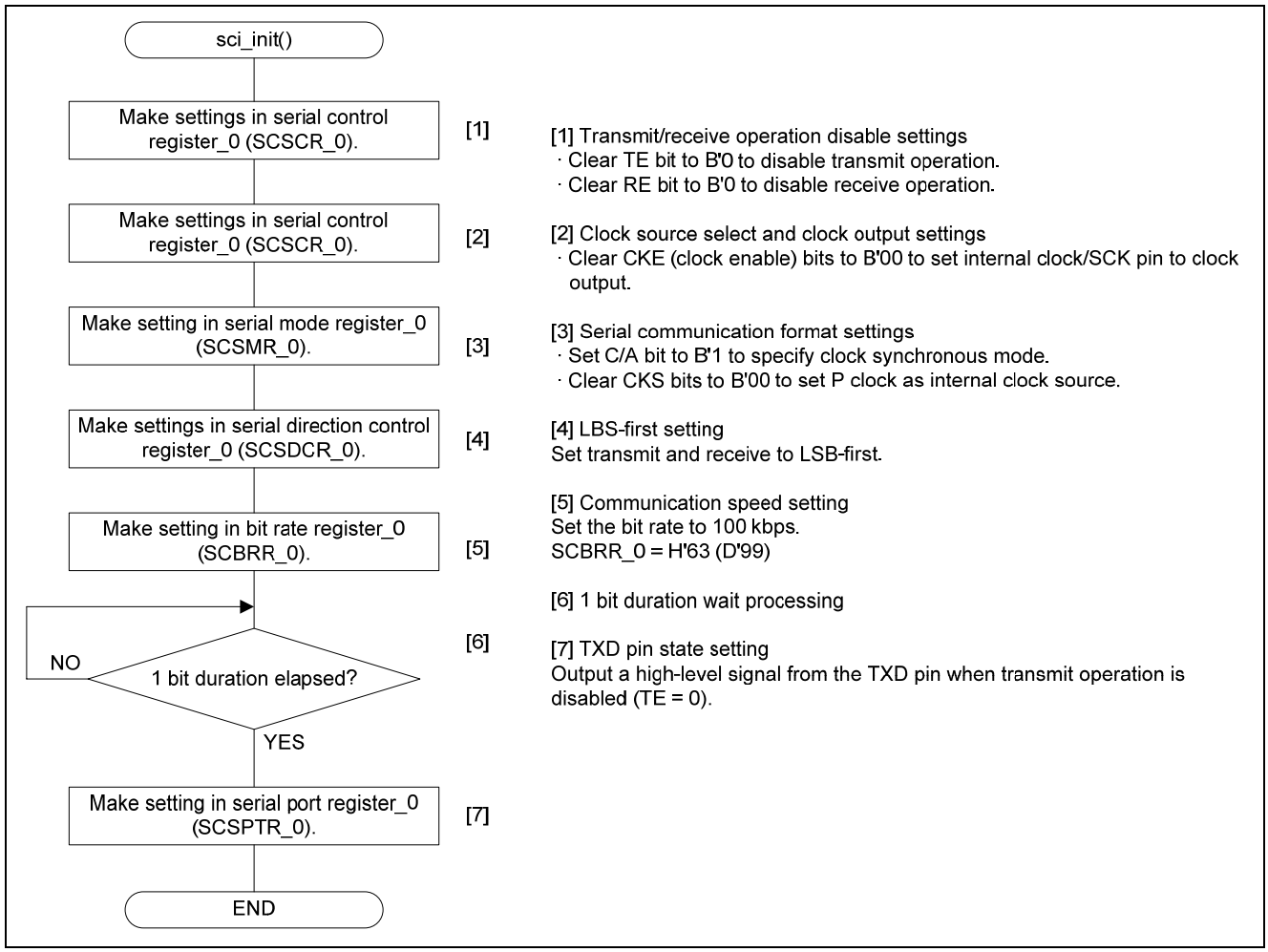


Figure 11 Initialization of Serial Communication Interface (SCI)

2.4.5 Initialization of Pin Function Controller (PFC)

Figure 12 shows the initial setting sequence for the pin function controller (PFC).

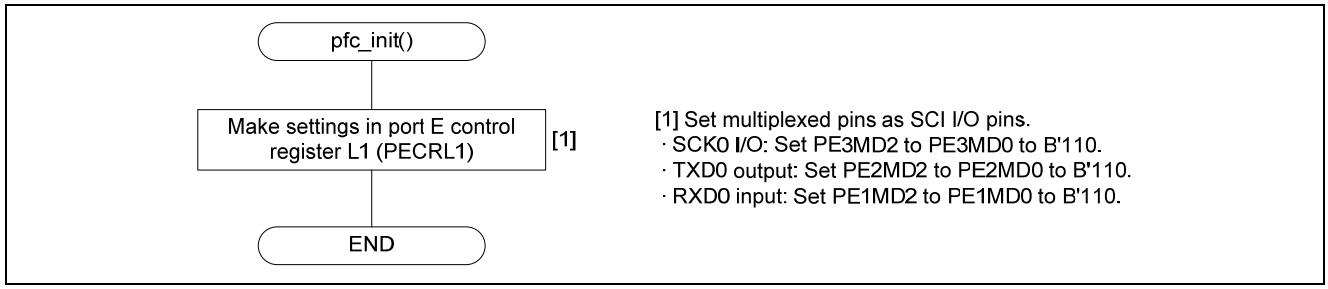


Figure 12 Initialization of Pin Function Controller (PFC)

2.4.6 Handling of SCI Receive Data Full Interrupt (RXI0)

Figure 13 shows the processing sequence of the SCI receive data full interrupt (RXI0) handler.

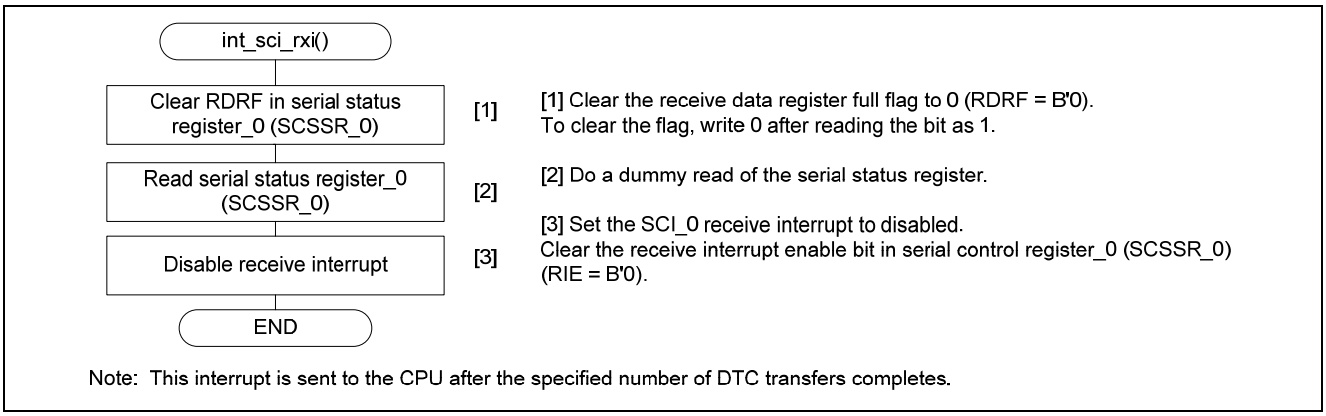


Figure 13 SCI Receive Data Full Interrupt (RXI0) Processing Sequence

2.4.7 Handling of SCI Transmit Data Empty Interrupt (TXI0)

Figure 14 shows the processing sequence of the SCI transmit data empty interrupt (TXI0) handler.

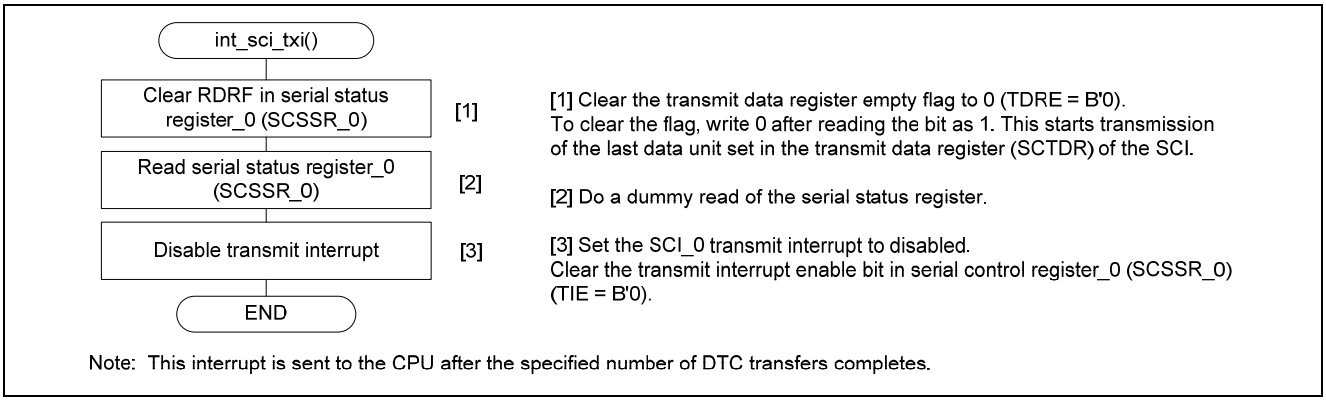


Figure 14 SCI Transmit Data Empty Interrupt (TXI0) Processing Sequence

2.4.8 Handling of SCI Receive Error Interrupt (ERI0)

Figure 15 shows the processing sequence of the SCI receive error interrupt (ERI0) handler.

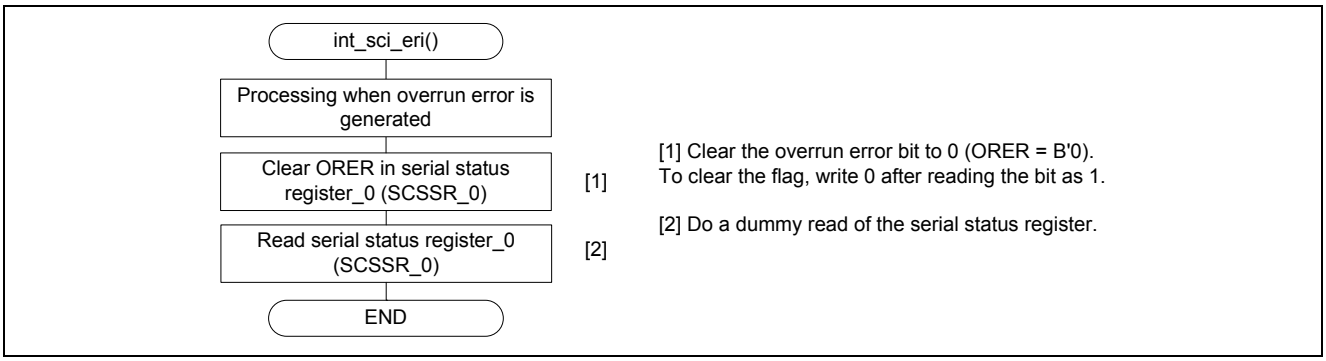


Figure 15 Receive Error Interrupt (ERI0) Processing Sequence

2.5 Settings of Registers in the Sample Program

The register setting values used in the reference program are listed below.

2.5.1 Clock Pulse Generator (CPG)

Table 7 shows the register settings for the clock pulse generator (CPG).

Table 7 Clock pulse generator (CPG)

Register Name	Address	Setting	Description
Frequency control register (FRQCR)	H'FFFFE800	H'0241	Specifies the operating frequency multiplication ratios. • IFC2 to IFC0 = B'000: Internal clock (Iϕ) × 1 • BFC2 to BFC0 = B'001: Bus clock (Bϕ) × 1/2 • PFC2 to PFC0 = B'001: Peripheral clock (Pϕ) × 1/2 • MIFC2 to MIFC0 = B'000: MTU2S clock (MIϕ) × 1 • MPFC2 to MPFC0 = B'001: MTU2 clock (MPϕ) × 1/2

2.5.2 Power-Down Mode

Table 8 shows the register settings for low-power mode.

Table 8 Power-Down Mode

Register Name	Address	Setting	Description
Standby control register_2 (STBCR_2)	H'FFFFE804	H'28	Specifies the operation settings for individual modules. • MSTP7 = B'0: Operate RAM. • MSTP6 = B'0: Operate ROM. • MSTP4 = B'0: Operate DTC.
Standby control register_3 (STBCR_3)	H'FFFFE806	H'F7	Specifies the operation settings for individual modules. • MSTP15 = B'1: Stop clock supply to I2C2. • MSTP13 = B'1: Stop clock supply to SCI_2. • MSTP12 = B'1: Stop clock supply to SCI_1. • MSTP11 = B'0: operate SCI_0. • MSTP10 = B'1: Stop clock supply to SSU. • MSTP8 = B'1: Stop clock supply to RCAN-ET_0.

2.5.3 Interrupt Controller (INTC)

Table 9 shows the register settings for the interrupt controller (INTC).

Table 9 Interrupt Controller (INTC)

Register Name	Address	Setting	Description
Interrupt priority register L (IPRL)	H'FFFFE992	H'F000	Sets interrupt priority levels (level 0 to 15). • Bits 15 to 12 = B'1111: SCI_0 interrupt level = 15 • Bits 11 to 8 = B'0000: SCI_1 interrupt level = 0 • Bits 7 to 4 = B'0000: SCI_2 interrupt level = 0 • Bits 3 to 0: Reserved The SCI_0 interrupt is used by the reference program.

Note: The SCI0 RXI and TXI interrupt priority is according to the offset address order of the interrupt vector addresses. For details on interrupt priority, see the Interrupt Exception Handling Vector Table item in the Interrupt Controller section of the SH7137 Group Hardware Manual.

2.5.4 Pin Function Controller (PFC)

Table 10 shows the register settings for the pin function controller (PFC).

Table 10 Pin Function Controller (PFC)

Register Name	Address	Setting	Description
Port E control register L1 (PECRL1)	H'FFFFD316	H'6660	Sets port E multiplexed pin functions. • PE3MD2 to PE3MD0 = B'110: PE3 functions as SCK0 I/O (SCI). • PE2MD2 to PE2MD0 = B'110: PE2 functions as TXD0 output (SCI). • PE1MD2 to PE1MD0 = B'110: PE1 functions as RXD0 input (SCI). • PE0MD1 and PE0MD0 = B'00: PE0 functions as PE0 I/O (port).

2.5.5 Data Transfer Controller (DTC)

Tables 11, 12, and 13 list the DTC settings for this application note.

Table 11 Data Transfer Controller (DTC) Common Settings

Register Name	Address	Setting	Description
DTC control register (DTCCR)	H'FFFFCC90	H'00	• RRS = B'0: No not skip reading transfer information. • RCHNE = B'0: Disable chain transfer. • ERR = B'0: No interrupt requests.
DTC vector base register (DTCVBR)	H'FFFFCC94	H'FFFF8000	Specify the on-chip RAM area as the base address used to calculate vector table addresses.
DTC enable register E(DTCERE)	H'FFFFCC88	H'C000	Select the interrupt source that activates the DTC. • DTCERE15 = B'1: Select RXI_0 as the activation source. • DTCERE14 = B'1: Select TXI_0 as the activation source.

Table 12 DTC Transfer Information (DTC_RXI0) for SCI Receive

Register Name	Address	Setting	Description
DTC mode register A(MRA)	H'FFFF8800 *1	H'00	• MD1 and MD0 = B'00: Normal transfer • Sz1 and Sz0 = B'00: Byte size transfer • SM1 and SM0 = B'00: SAR is fixed.
DTC mode register B(MRB)	H'FFFF8801 (MRA +1)	H'08	• CHNE = B'0: Disable chain transfer. • CHNS = B'0: Continuous chain transfer. • DISEL = B'0: Generate CPU interrupt request at end of specified data transfer count. • DTS = B'0: Set destination as repeat area or block area. • DM1 and DM0 = B'10: Increment DAR.
DTC source address register (SAR)	H'FFFF8804 (MRA +4)	SCRDR_0 Register	Specify transfer source address. Set SCI0 receive data register_0 (SCRDR_0).
DTC destination address register(DAR)	H'FFFF8808 (MRA +8)	On-chip RAM*2	Specify transfer destination address. Store start address of buffer array variable for receive (&rxi0_data[0]).
DTC transfer count register A(CRA)	H'FFFF880C (MRA+12)	H'0020	Specify DTC data transfer count. 32 times
DTC transfer count register B(CRB)	H'FFFF880E (MRA+14)	H'0000	Specify DTC block data transfer count for block transfer mode (not used).

Notes: 1. The transfer information is allocated to on-chip RAM as a structure variable with no initial value. The allocation of variables in memory is dependent on the section allocation settings of the optimizing linkage editor used to create the executable object code.
2. The array variables are allocated to on-chip RAM as variables with no initial value. The allocation of variables in memory depends on the results of the compile process used to generate the executable object code.

Table 13 DTC Transfer Information (DTC_TXI0) for SCI Transmit

Register Name	Address	Setting	Description
DTC mode register A(MRA)	H'FFFF8810 *1	H'08	• MD1 and MD0 = B'00: Normal transfer • Sz1 and Sz0 = B'00: Byte size transfer • SM1 and SM0 = B'10: Increment SAR.
DTC mode register B(MRB)	H'FFFF8811 (MRA +1)	H'00	• CHNE = B'0: Disable chain transfer. • CHNS = B'0: Continuous chain transfer. • DISEL = B'0: Generate CPU interrupt request at end of specified data transfer count. • DTS = B'0: Set destination as repeat area or block area. • DM1 and DM0 = B'00: DAR is fixed.
DTC source address register (SAR)	H'FFFF8814 (MRA +4)	On-chip RAM*2	Specify transfer source address. Store start address of buffer array variable for transmit (&txi0_data[0]).
DTC destination address register (DAR)	H'FFFF8818 (MRA +8)	SCTDR_0 Register	Specify transfer destination address. Set SCI0 transfer data register_0 (SCTDR_0).
DTC transfer count register A(CRA)	H'FFFF881C (MRA+12)	H'0020	Specify DTC data transfer count. 32 times
DTC transfer count register B(CRB)	H'FFFF881E (MRA+14)	H'0000	Specify DTC block data transfer count for block transfer mode (not used).

Notes: 1. The transfer information is allocated to on-chip RAM as a structure variable with no initial value. The allocation of variables in memory is dependent on the section allocation settings of the optimizing linkage editor used to create the executable object code.
2. The array variables are allocated to on-chip RAM as variables with no initial value. The allocation of variables in memory depends on the results of the compile process used to generate the executable object code.

2.5.6 Serial Communication Interface (SCI)

Table 14 shows the SCI register settings for this application note.

Table 14 SCI Register Settings

Register Name	Address	Setting	Description
Serial mode register_0 (SCSMR_0)	H'FFFEC000	H'80	• C/A = B'1: Clock synchronous mode • • CHR = B'0: 8 data bits • • CKS1 and CKS0 = B'00: Pϕ clock
Bit rate register_0 (SCBRR_0)	H'FFFEC002	H'63 (99)	Clock synchronous mode Bit rate: 100 Kbit/s*
Serial control register_0 (SCSCR_0)	H'FFFEC004	H'00	Initial settings • TIE = B'0: Disable transmit data empty interrupt (TXI) requests. • RIE = B'0: Disable receive data full interrupt (RXI) requests. • TE = B'0: Disable transmit operation. • RE = B'0: Disable receive operation. • MPIE = B'0: Disable multiprocessor mode. • TEIE = B'0: Disable transmit end interrupt (TEI) requests. • CKE1 and CKE0 = B'00: Internal clock/SCK pin set as synchronization clock output (clock synchronous mode)
		H'F0	When transmit and receive are enabled • TIE = B'1: Enable interrupt (TXI) requests. • RIE = B'1: Enable interrupt (RXI) requests. • TE = B'1: Enable transmit operation. • RE = B'1: Enable receive operation. • Set bits TE and RE to enable simultaneously.
Serial status register_0 (SCSSR_0)	H'FFFEC008	H'84 (initial value)	Status flags retain initial settings. • TDRE = B'1: Transmit data register empty flag • RDRF = B'0: Receive data register full flag • ORER = B'0: Overrun error flag • FER = B'0: Framing error flag • PER = B'0: Parity error flag • TEND = B'1: Transmit end flag
Serial direction control register_0 (SCSDCR_0)	H'FFFFC00C	H'F2	LSB-first/MSB-first selection • DIR = B'0: LSB-first for transmit and receive
Serial port register_0 (SCSPTR_0)	H'FFFFC00E	H'03	• EIO = B'0: When the RIE bit is set to 1, send RXI and ERI interrupts to the INTC. • SPB1IO = B'0: Do not output the value of the SPB1DT bit on the SCK pin. • PB0IO = PB0DT = B'1: Control the TXD pin according to the TE bit. TXD output is high-level when TE = 0.

Note: * For details on the bit rate setting, see the Bit Rate Register (SCBRR) item in the Serial Communication Interface section of the SH7137 Group Hardware Manual.

3. Documents for Reference

- Hardware Manual
  SH7211 Group Hardware Manual [RJJ09B0338]
  (The latest version can be downloaded from the Renesas Technology Web site.)
- Software Manual
  SH-2A/SH2A-FPU Software Manual [RJJ09B0086]
  (The latest version can be downloaded from the Renesas Technology Web site.)

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	Jul.16.09	—	First edition issued

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.