RX63N Group, RX631 Group

Asynchronous SCIc Transmission/Reception Using DMACA

R01AN1712EJ0100
Rev. 1.00
Apr. 1, 2014

Abstract

This application note describes how to perform asynchronous transmission/reception using the serial communications interface (SCI) with the DMA controller (DMAC) in the RX63N Group, RX631 Group.

Products

RX63N Group, 176-Pin and 177-Pin Packages, ROM Capacities: 768 Kbytes to 2 Mbytes

RX63N Group, 144-Pin and 145-Pin Packages, ROM Capacities: 768 Kbytes to 2 Mbytes

RX63N Group, 100-Pin Package, ROM Capacities: 768 Kbytes to 2 Mbytes

RX631 Group, 176-Pin and 177-Pin Packages, ROM Capacities: 256 Kbytes to 2 Mbytes

RX631 Group, 144-Pin and 145-Pin Packages, ROM Capacities: 256 Kbytes to 2 Mbytes

RX631 Group, 100-Pin Package, ROM Capacities: 256 Kbytes to 2 Mbytes

When using this application note with other Renesas MCUs, careful evaluation is recommended after making modifications to comply with the alternate MCU.

Contents

1. Specifications

This document describes performing asynchronous serial communication using the SCI.

Transmit data is prestored in the transmit data storage area in the RAM and transmitted using the DMAC. Receive data is stored in the RAM's receive data storage area using the DMAC.

Serial transmission/reception starts when a falling edge is detected on the IRQ15 interrupt request pin.

- Bit rate: 38,400 bps
- Communication format: 8-bit length, LSB first
- Stop bit: 1 bit
- Parity: None
- Hardware flow control: None

Table 1.1 lists the Peripheral Functions and Their Applications, and Figure 1.1 shows the Block Diagram.

Table 1.1 Peripheral Functions and Their Applications

Peripheral Function	Application
SCIc channel 9 (SCI9)	Asynchronous serial transmission/reception
DMACA channel 0 (DMAC0)	Transfer data received by SCI9 to the RAM
DMACA channel 1 (DMAC1)	Transfer transmit data in the RAM to SCI9
IRQ15	Start trigger for serial transmission/reception

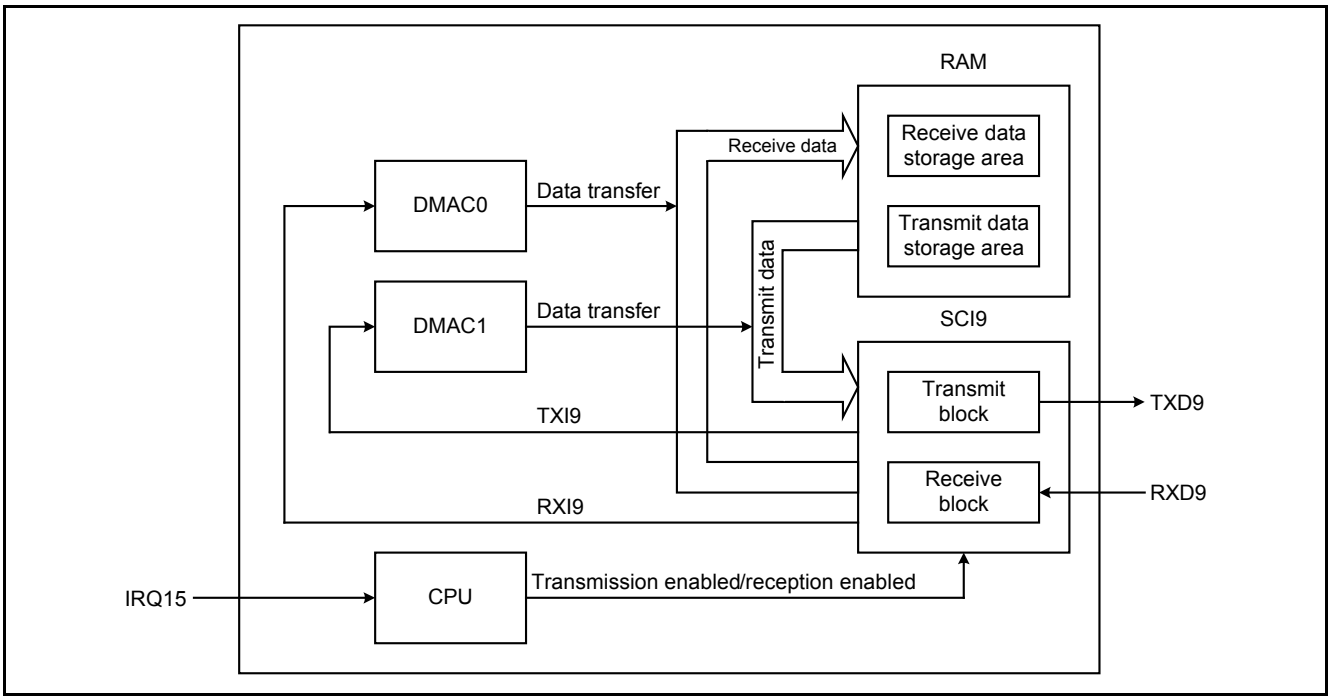

Figure 1.1 Block Diagram

2. Operation Confirmation Conditions

The sample code accompanying this application note has been run and confirmed under the conditions below.

Table 2.1 Operation Confirmation Conditions

Item	Contents
MCU used	R5F563NBDDFC (RX63N Group)
Operating frequencies	• Main clock: 12 MHz • PLL clock: 192 MHz (main clock divided by 1 and multiplied by 16) • System clock (ICLK): 96 MHz (PLL divided by 2) • Peripheral module clock B (PCLKB): 48 MHz (PLL divided by 4)
Operating voltage	3.3 V
Integrated development environment	Renesas Electronics Corporation High-performance Embedded Workshop Version 4.09.01
C compiler	Renesas Electronics Corporation C/C++ Compiler Package for RX Family V.1.02 Release 01
	Compile options -cpu=rx600 -output=obj="$(CONFIGDIR)\$(FILELEAF).obj" -debug -nologo The integrated development environment default settings are used.
iodefine.h version	Version 1.6A
Endian	Little endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample code version	Version 1.00
Board used	Renesas Starter Kit+ for RX63N (product part number: R0K50563NC000BE)

3. Reference Application Note

For additional information associated with this document, refer to the following application note.

- RX63N Group, RX631 Group Initial Setting Rev. 1.10 (R01AN1245EJ0110)

The initial setting functions in the reference application note are used in the sample code in this application note. The revision number of the reference application note is the one when this application note was made. However, the latest version is always recommended. Visit the Renesas Electronics Corporation website to check and download the latest version.

4. Hardware

4.1 Hardware Configuration

Figure 4.1 shows a Connection Example.

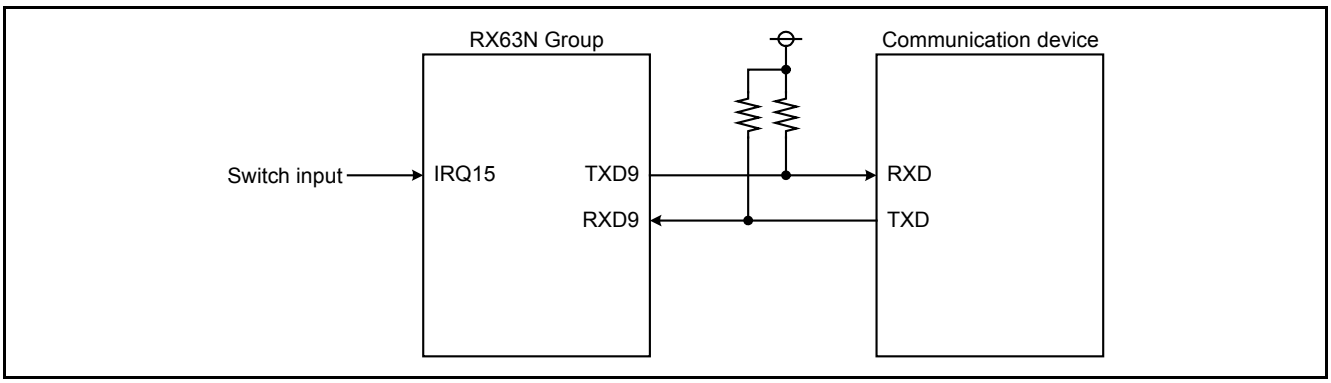

Figure 4.1 Connection Example

4.2 Pins Used

Table 4.1 lists the Pins Used and Their Functions.

This table assumes the 176-pin package is used. When using packages with less than 176 pins, select the pins appropriate to the package used.

Table 4.1 Pins Used and Their Functions

Pin Name	I/O	Function
P07/IRQ15	Input	Switch input to start transmission and reception
PB6/RXD9	Input	Receive data input to SCI9
PB7/TXD9	Output	Transmit data output from SCI9

5. Software

In the sample code, SCI9 transmission and reception are processed automatically using the DMAC. SCI9 data transmission and reception are started by pressing a switch.

When data transmission is enabled, the TXI9 interrupt request is generated which becomes the DMAC1 transfer request. Using DMAC1, data in the transmit data storage area is transferred to the TDR register and then transmitted.

When reception is complete, the RXI9 interrupt request is generated which becomes the DMAC0 transfer request. Using DMAC0, receive data is transferred to the receive data storage area.

After the transmit data is transferred 256 times, the DMAC1 interrupt is generated. At this point, the TXI9 interrupt is disabled and the TEI9 interrupt is enabled.

After the receive data is transferred 256 times, the DMAC0 interrupt is generated. At this point, SCI9 reception is disabled, the RXI9 interrupt is disabled, and the receive end flag becomes 1.

After 256 bytes of data are transmitted, the TEI9 interrupt is generated. At this point, SCI9 transmission and TEI9 interrupt are disabled, and the transmit end flag becomes 1.

Settings for the peripheral functions are listed below.

SCI9

- Communication mode: Asynchronous mode
- Clock source: PCLKB/4
- Communication speed: 38,400 bps (BRR register setting value = (PCLKB ÷ (64 × 2 × 38,400 bps)) – 1)
- Data length: 8 bits
- Stop bits: 1
- Parity: None
- Data transfer direction: LSB first
- Interrupts: Transmit end interrupt (TEI9), transmit data empty interrupt (TXI9), receive data full interrupt (RXI9), and receive error interrupt (ERI9) are used

DMAC0

- Activation source: RXI9 interrupt request. The IR flag for the RXI9 interrupt is cleared to 0 when transfer starts.
- Transfer source address: SCI9.RDR register
- Transfer source address update mode: Address fixed
- Transfer destination address: RAM (start address in the receive data storage area)
- Transfer destination address update mode: Increment
- Transfer mode: Normal transfer
- Data transfer size: 8 bits
- Number of transfers: 256
- Interrupts: Transfer end interrupt (DMAC0I) is used

DMAC1

- Activation source: TXI9 interrupt request. The IR flag for the TXI9 interrupt is cleared to 0 when transfer starts.
- Transfer source address: RAM (start address in the transmit data storage area)
- Transfer source address update mode: Increment
- Transfer destination address: SCI9.TDR register
- Transfer destination address update mode: Address fixed
- Transfer mode: Normal transfer
- Data transfer size: 8 bits
- Number of transfers: 256
- Interrupts: Transfer end interrupt (DMAC1I) is used

IRQ15 input pin

- Detection method: Falling edge
- Digital filter: Enabled (sampling clock: PCLKB/8)
- Interrupts: Not used

5.1 Operation Overview

5.1.1 Transmitting

(1) Initial setting
After the initial setting, the program waits for the switch input to start transmission and reception.

(2) Detecting transmit/receive start switch input
When the switch input to start transmission and reception is detected, the IR flag for the IRQ15 interrupt is set to 0. Determine the value of the transmit end flag and receive end flag to confirm that transmission and reception are complete. If completion is confirmed, the transmit end flag is set to 0 (transmitting). Set the DMAC1 transfer source address, set the number of transfers, and enable DMA transfer. Set bits SCI9.SCR.TEIE, TIE, RIE, TE, and RE to 1 simultaneously to enable transmission and reception. By setting bits SCI9.SCR.TIE and TE to 1 simultaneously, the IR flag for the TXI9 interrupt becomes 1.

(3) Start data transfer
When the TXI9 interrupt is enabled, DMAC1 is activated, and the IR flag for the TXI9 interrupt becomes 0. The first byte of transmit data is transferred from the transmit data storage area in the RAM to the SCI9.TDR register.

(4) Start data transmission
Data is transferred from the SCI9.TDR register to the SCI9.TSR register, the IR flag for the TXI9 interrupt becomes 1, and the first byte of transmit data is output from the TXD9 pin. The TXI9 interrupt request triggers DMAC1 activation, and the second byte of transmit data is transferred.

(5) DMAC1I interrupt
After the 256th byte of data is transferred, the DMAC1I interrupt request is generated. The TXI9 interrupt is disabled and the TEI9 interrupt is enabled in the DMAC1I interrupt handling.

(6) TEI9 interrupt
When the last bit in the 256th byte of data is transmitted, the SCI9.TDR register is not updated, so the TEI9 interrupt request is generated. Transmission is disabled and the TEI9 interrupt is disabled in the TEI9 interrupt handling. Then the transmit end flag is set to 1 (transmission ended).

This procedure is repeated starting from step (2) above.

Figure 5.1 shows the Timing Diagram When Transmitting Data.

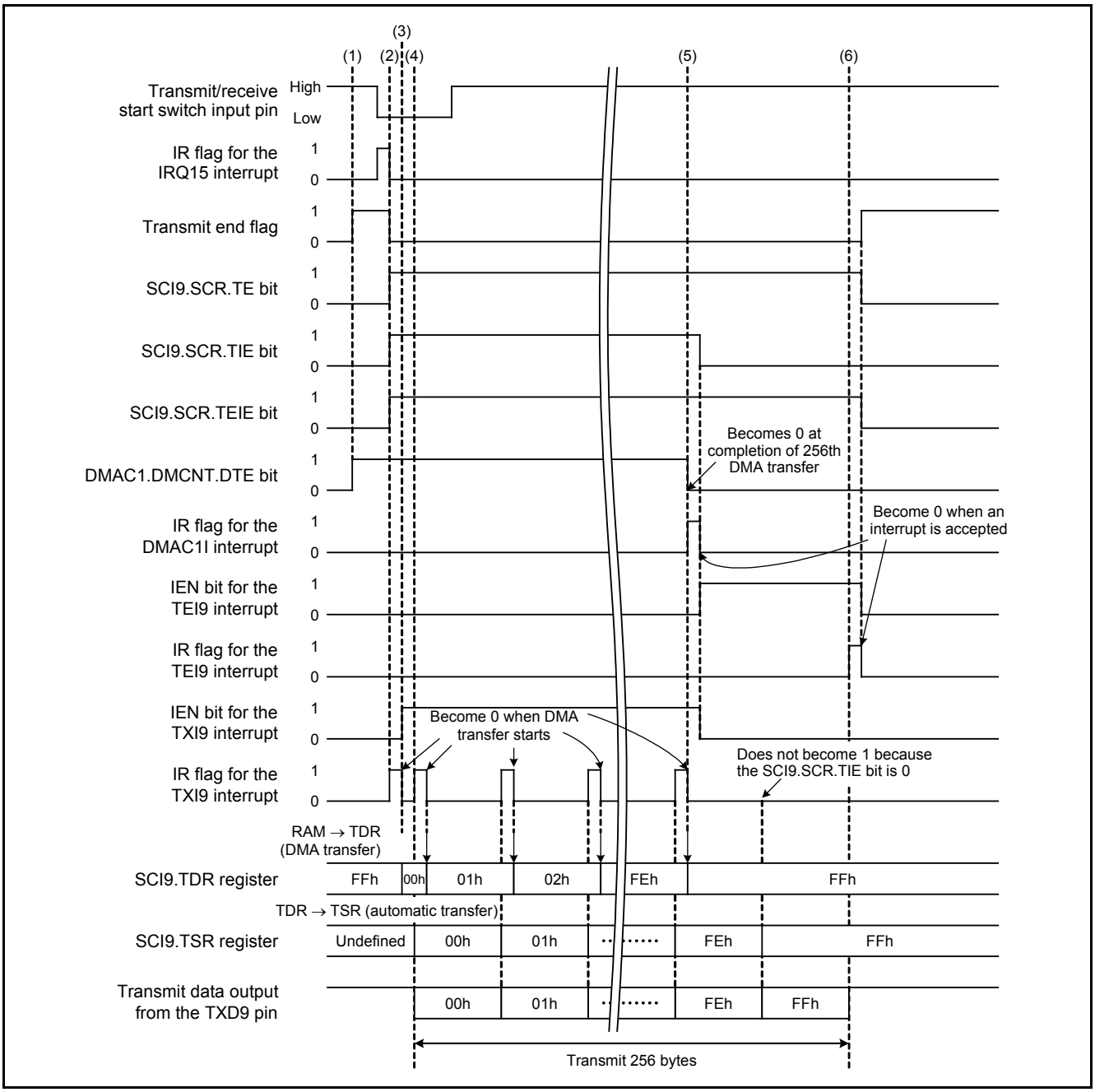

Figure 5.1 Timing Diagram When Transmitting Data

5.1.2 Receiving

(1) Initial setting
After the initial setting, the program waits for the switch input to start transmission and reception.

(2) Detecting transmit/receive start switch input
When the switch input to start transmission and reception is detected, the IR flag for the IRQ15 interrupt becomes 0. Determine the value of the transmit end flag and receive end flag to confirm that transmission and reception are complete. If completion is confirmed, the receive end flag is set to 0 (receiving). Set the DMAC0 transfer destination address, set the number of transfers, and enable DMA transfer. Set bits SCI9.SCR.TEIE, TIE, RIE, TE, and RE to 1 simultaneously to enable transmission and reception, and enable the RXI9 interrupt.

(3) Complete data reception
After the first byte of data is received, data is transferred from the SCI9.RSR register to the SCI9.RDR register, and the IR flag for the RXI9 interrupt becomes 1.

(4) Start data transfer
The RXI9 interrupt request activates DMAC0, and the IR flag for the RXI9 interrupt becomes 0. The first byte of receive data is transferred from the SCI9.RDR register to the receive data storage area in the RAM.

(5) DMAC0I interrupt
After the 256th byte of data is transferred, the DMAC0I interrupt request is generated. Reception is disabled and the RXI9 interrupt is disabled in the DMAC0I interrupt handling. The receive end flag is set to 1 (reception ended).

This procedure is repeated starting from step (2) above.

Figure 5.2 shows the Timing Diagram When Receiving Data.

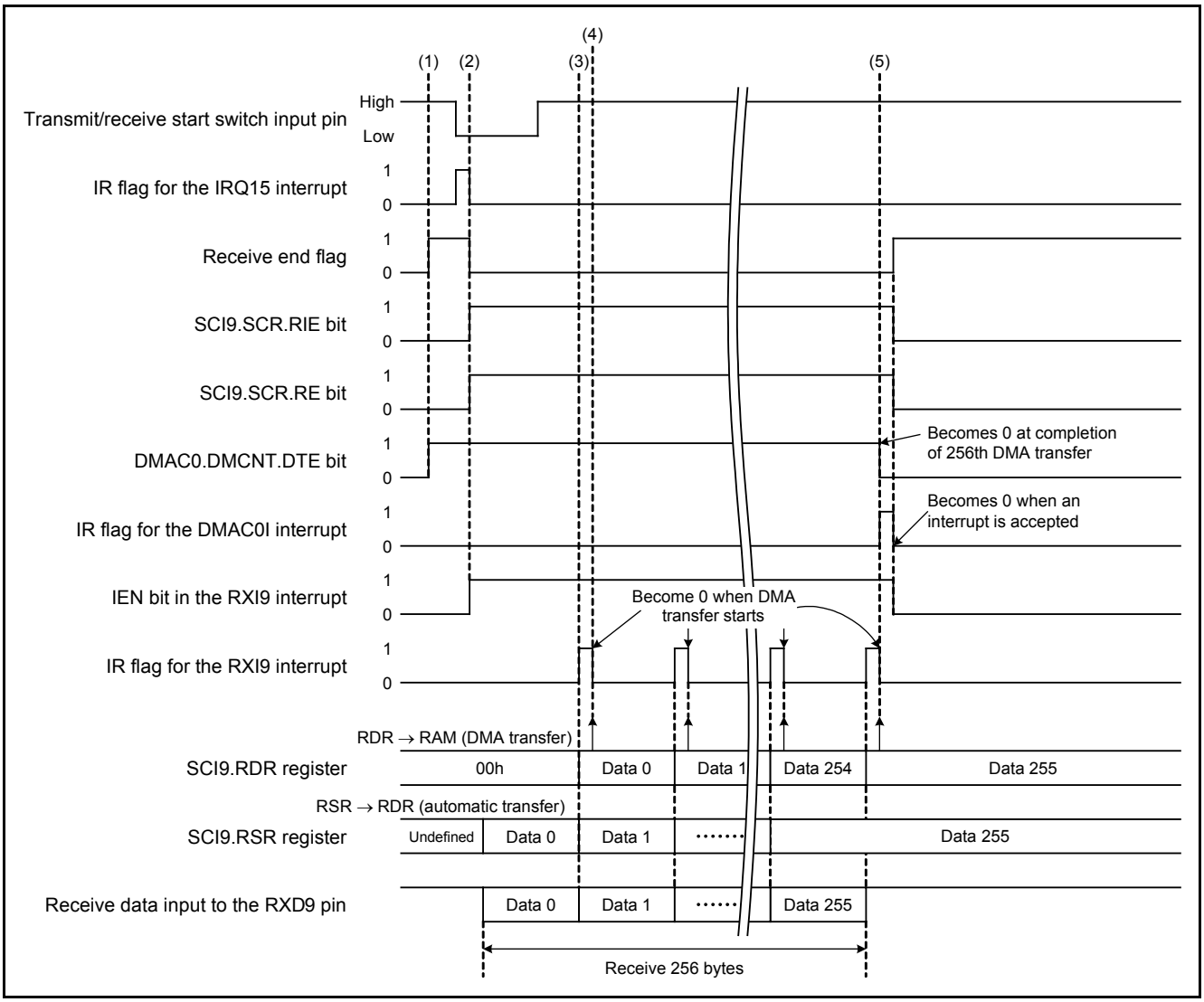

Figure 5.2 Timing Diagram When Receiving Data

Notes when incorporating the sample code into the user system

When incorporating the sample codes accompanying this application note into the user system, note the following:

When an interrupt used in this application note is delayed for a prolonged time due to other interrupt handlers, the sample code may not be executed properly.

5.2 File Composition

Table 5.1 lists the Files Used in the Sample Code. Files generated by the integrated development environment are not included in this table.

Table 5.1 Files Used in the Sample Code

File Name	Outline	Remarks
main.c	Main processing	
r_init_stop_module.c	Stop processing for active peripheral functions after a reset	
r_init_stop_module.h	Header file for r_init_stop_module.c	
r_init_non_existent_port.c	Nonexistent port initialization	
r_init_non_existent_port.h	Header file for r_init_non_existent_port.c	
r_init_clock.c	Clock initialization	
r_init_clock.h	Header file for r_init_clock.c	

5.3 Option-Setting Memory

Table 5.2 lists the Option-Setting Memory Configured in the Sample Code. When necessary, set a value suited to the user system.

Table 5.2 Option-Setting Memory Configured in the Sample Code

Symbol	Addresses	Setting Value	Contents
OFS0	FFFF FF8Fh to FFFF FF8Ch	FFFF FFFFh	The IWDT is stopped after a reset. The WDT is stopped after a reset.
OFS1	FFFF FF8Bh to FFFF FF88h	FFFF FFFFh	The voltage monitor 0 reset is disabled after a reset. HOCO oscillation is disabled after a reset.
MDES	FFFF FF83h to FFFF FF80h	FFFF FFFFh	Little endian

5.4 Constants

Table 5.3 lists the Constants Used in the Sample Code.

Table 5.3 Constants Used in the Sample Code

Constant Name	Setting Value	Contents
BUF_SIZE	256	Transmit or receive data storage area size
DMAC_CNT	BUF_SIZE	Number of DMAC transfers
SW_ON	1	Switch input on
SW_OFF	0	Switch input off

5.5 Variables

Table 5.4 lists the Global Variables.

Table 5.4 Global Variables

Type	Variable Name	Contents	Function Used
unsigned char	rcv_end_flag	Receive end flag 0: Receiving 1: Reception ended	main Excep_DMAC_DMAC0I Excep_SCI9_TEI9
unsigned char	trn_end_flag	Transmit end flag 0: Transmitting 1: Transmission ended	main Excep_DMAC_DMAC0I Excep_SCI9_TEI9
unsigned char	rcvbuf[BUF_SIZE]	Receive data storage area	dmac0_init sci9_start
unsigned char	trnbuf[BUF_SIZE]	Transmit data storage area	main dmac1_init sci9_start

5.6 Functions

Table 5.5 lists the Functions.

Table 5.5 Functions

Function Name	Outline
main	Main processing
port_init	Port initialization
R_INIT_StopModule	Stop processing for active peripheral functions after a reset
R_INIT_NonExistentPort	Nonexistent port initialization
R_INIT_Clock	Clock initialization
peripheral_init	Peripheral function initialization
sci9_init	SCI9 initialization
dmac0_init	DMAC0 initialization
dmac1_init	DMAC1 initialization
sci9_start	SCI9 transmission/reception start
Excep_DMAC_DMAC0I	DMAC0 transfer end interrupt handling
Excep_DMAC_DMAC1I	DMAC1 tranfer end interrupt handling
Excep_SCI9_TEI9	SCI9 transmit end interrupt handling
Excep_ICU_GROUP12	Group 12 interrupt handling (SCI9 receive error interrupt)

5.7 Function Specifications

The following tables list the specifications for the functions in the sample code.

main	
Outline	Main processing
Header	None
Declaration	void main(void)
Description	After the initial settings, when the switch input to start transmission and reception is detected, SCI9 transmission and reception are started.
Arguments	None
Return Value	None

port_init	
Outline	Port initialization
Header	None
Declaration	static void port_init(void)
Description	This function initializes the ports.
Arguments	None
Return Value	None

R_INIT_StopModule	
Outline	Stop processing for active peripheral functions after a reset
Header	r_init_stop_module.h
Declaration	void R_INIT_StopModule(void)
Description	This function configures settings to enter module-stop state.
Arguments	None
Return Value	None
Remark	Transition to the module-stop state is not performed in the sample code. For more information on this function, refer to the RX63N Group, RX631 Group Initial Setting Rev. 1.10 application note.

R_INIT_NonExistentPort	
Outline	Nonexistent port initialization
Header	r_init_non_existent_port.h
Declaration	void R_INIT_NonExistentPort(void)
Description	This function initializes port direction registers for ports that do not exist in products with less than 176 pins.
Arguments	None
Return Value	None
Remarks	The number of pins in the sample code is set for the 176-pin package (PIN_SIZE=176). After this function is called, when writing in byte units to the PDR and PODR registers which have nonexistent ports, set the corresponding bits for nonexistent ports as follows: set the I/O select bits in the PDR registers to 1 and set the output data store bits in the PODR registers to 0. For more information on this function, refer to the RX63N Group, RX631 Group Initial Setting Rev. 1.10 application note.

R_INIT_Clock	
Outline	Clock initialization
Header	r_init_clock.h
Declaration	void R_INIT_Clock(void)
Description	This function initializes the clocks.
Arguments	None
Return Value	None
Remark	In the sample code, the PLL clock is selected as the system clock, and the sub-clock is not used. For more information on this function, refer to the RX63N Group, RX631 Group Initial Setting Rev. 1.10 application note.

peripheral_init	
Outline	Peripheral function initialization
Header	None
Declaration	static void peripheral_init(void)
Description	This function initializes the peripheral functions being used.
Arguments	None
Return Value	None

sci9_init	
Outline	SCI9 initialization
Header	None
Declaration	void sci9_init(void)
Description	This function initializes channel SCI9.
Arguments	None
Return Value	None

dmac0_init	
Outline	DMAC0 initialization
Header	None
Declaration	void dmac0_init(void)
Description	This function initializes DMAC0.
Arguments	None
Return Value	None

dmac1_init	
Outline	DMAC1 initialization
Header	None
Declaration	void dmac1_init(void)
Description	This function initializes DMAC1.
Arguments	None
Return Value	None

sci9_start

Outline	SCI9 transmission/reception start
Header	None
Declaration	void sci9_start(void)
Description	This function starts transmission and reception on channel SCI9.
Arguments	None
Return Value	None

Excep_DMAC_DMAC0I

Outline	DMAC0 transfer end interrupt handling
Header	None
Declaration	static void Excep_DMAC_DMAC0I(void)
Description	This function disables reception, disables the RXI9 interrupt, and sets the receive end flag.
Arguments	None
Return Value	None

Excep_DMAC_DMAC1I

Outline	DMAC1 transfer end interrupt handling
Header	None
Declaration	static void Excep_DMAC_DMAC1I(void)
Description	This function disables the TXI9 interrupt and enables the TEI9 interrupt.
Arguments	None
Return Value	None

Excep_SCI9_TEI9

Outline	SCI9 transmit end interrupt handling
Header	None
Declaration	static void Excep_SCI9_TEI9(void)
Description	This function disables transmission, disables the TEI9 interrupt, and sets the transmit end flag.
Arguments	None
Return Value	None

Excep_ICU_GROUP12

Outline	Group 12 interrupt handling (SCI9 receive error interrupt)
Header	None
Declaration	static void Excep_ICU_GROUP12(void)
Description	This function performs group 12 interrupt handling (SCI9 receive error processing).
Arguments	None
Return Value	None
Remarks	SCI9 receive error processing is not performed in the sample code (infinite loop). Add a program as necessary.

5.8 Flowcharts

5.8.1 Main Processing

Figure 5.3 shows the Main Processing.

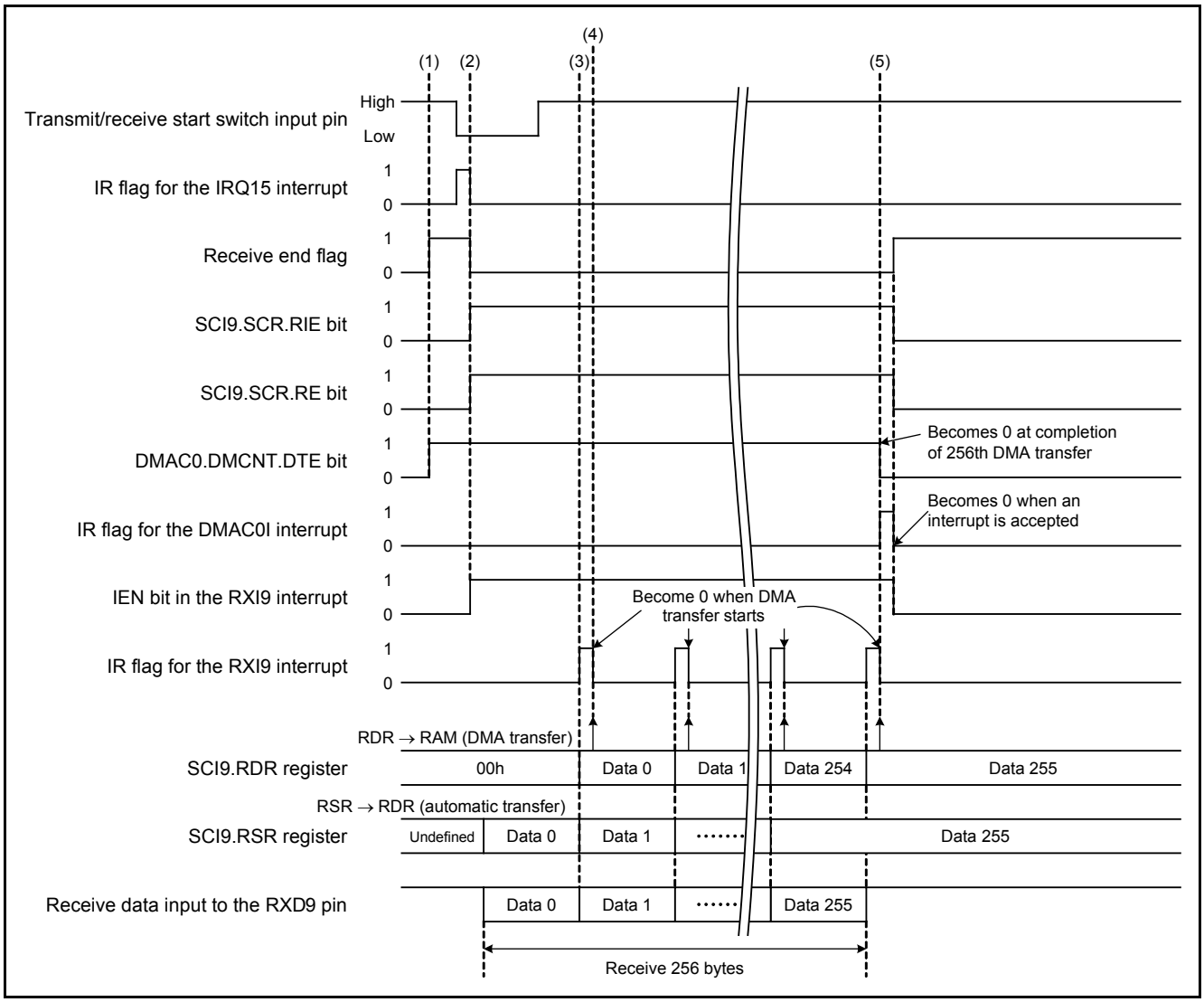

Figure 5.3 Main Processing

5.8.2 Port Initialization

Figure 5.4 shows the Port Initialization.

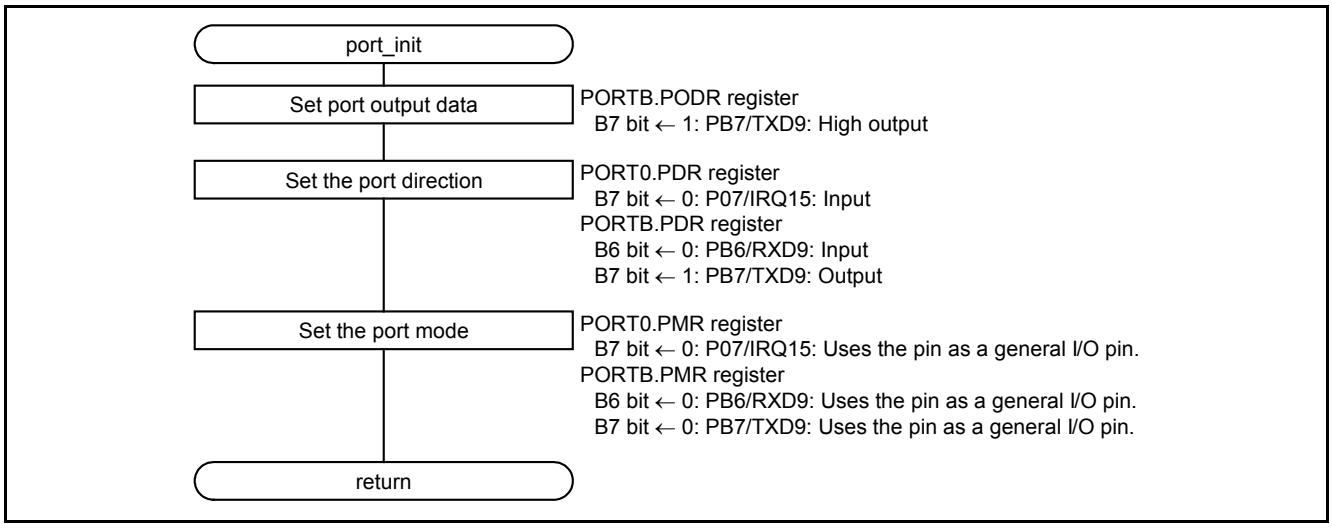

Figure 5.4 Port Initialization

5.8.3 Peripheral Function Initialization

Figure 5.5 shows the Peripheral Function Initialization.

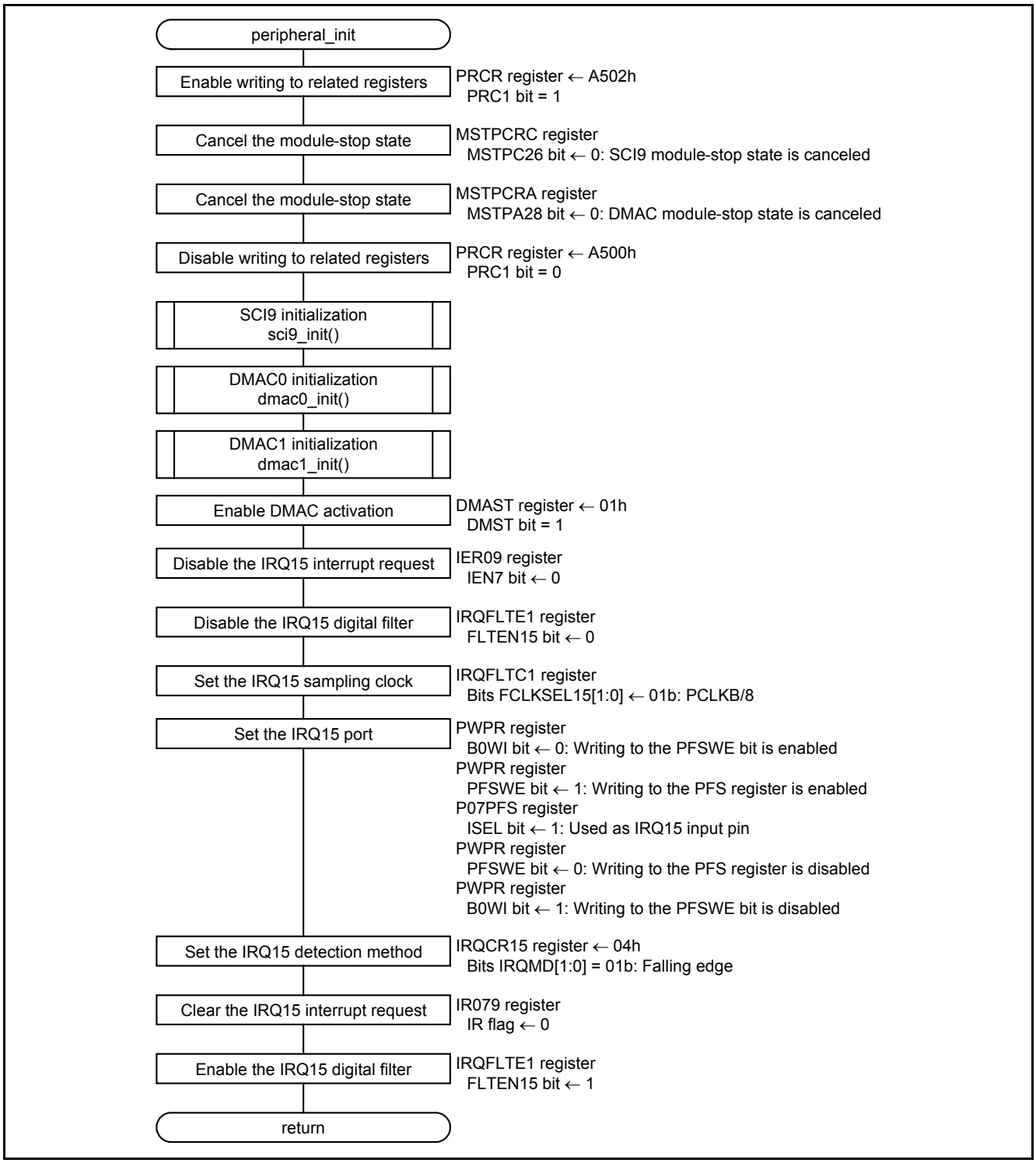

Figure 5.5 Peripheral Function Initialization

5.8.4 SCI9 Initialization

Figure 5.6 shows the SCI9 Initialization.

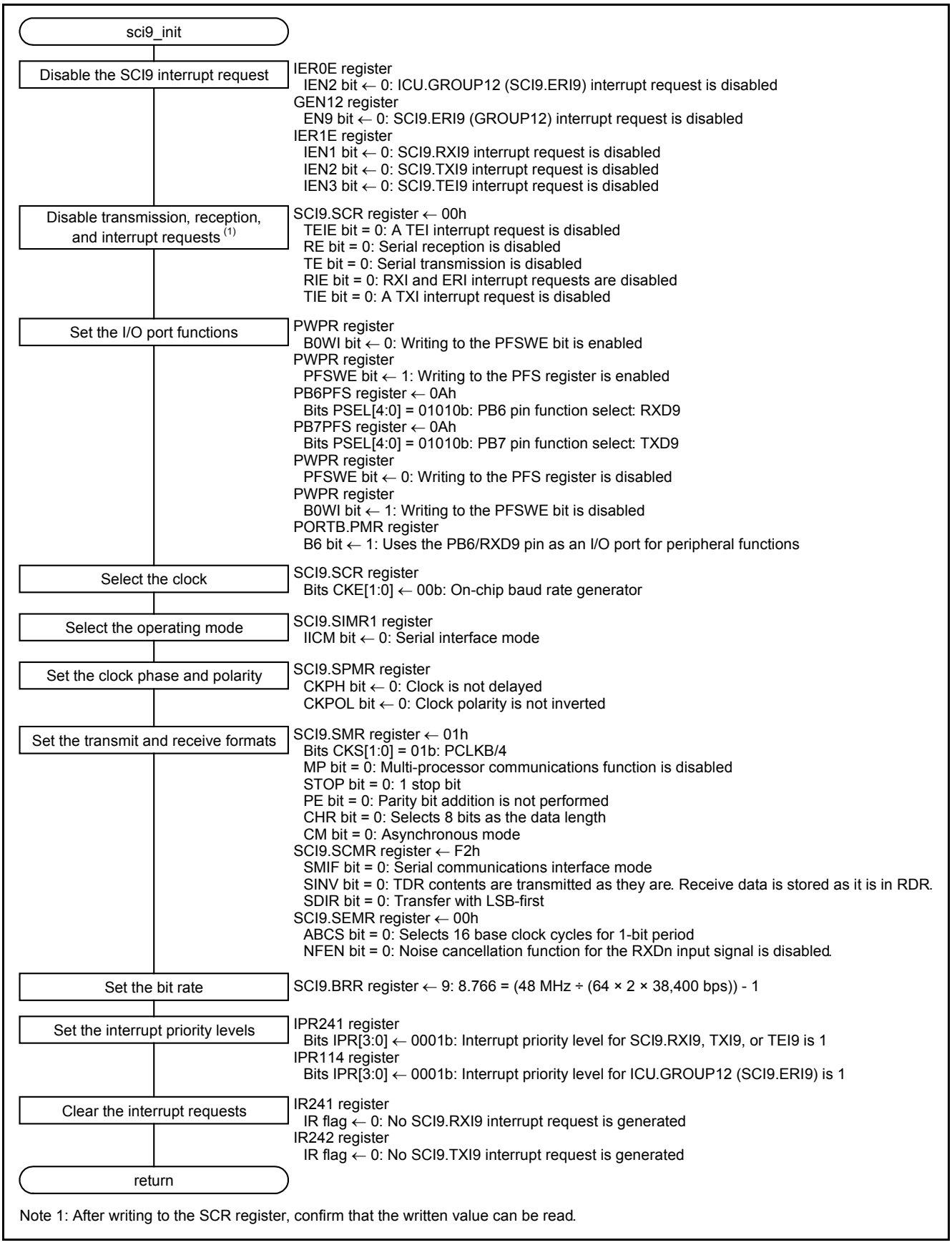

Figure 5.6 SCI9 Initialization

5.8.5 DMAC0 Initialization

Figure 5.7 shows the DMAC0 Initialization.

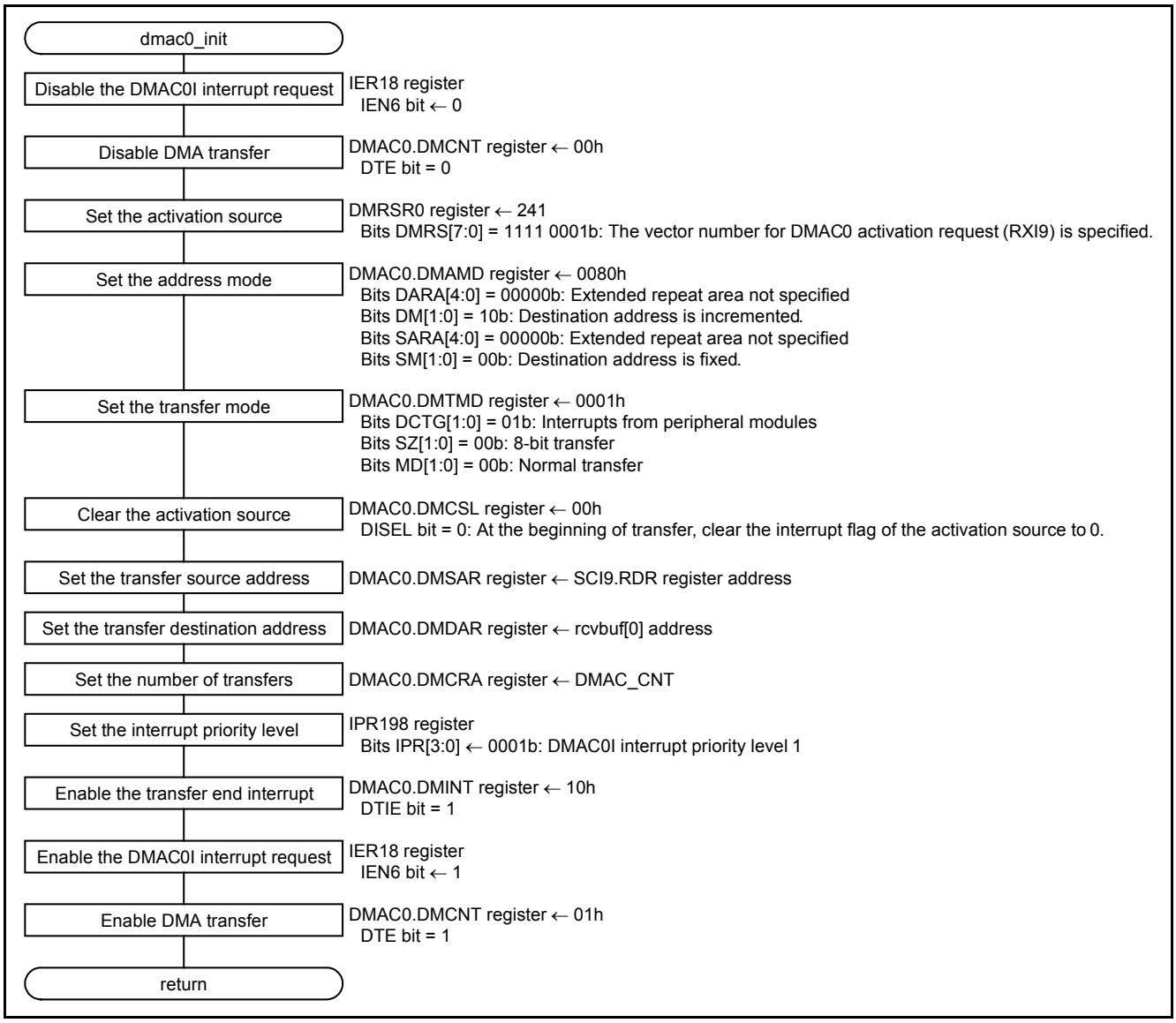

Figure 5.7 DMAC0 Initialization

5.8.6 DMAC1 Initialization

Figure 5.8 shows the DMAC1 Initialization.

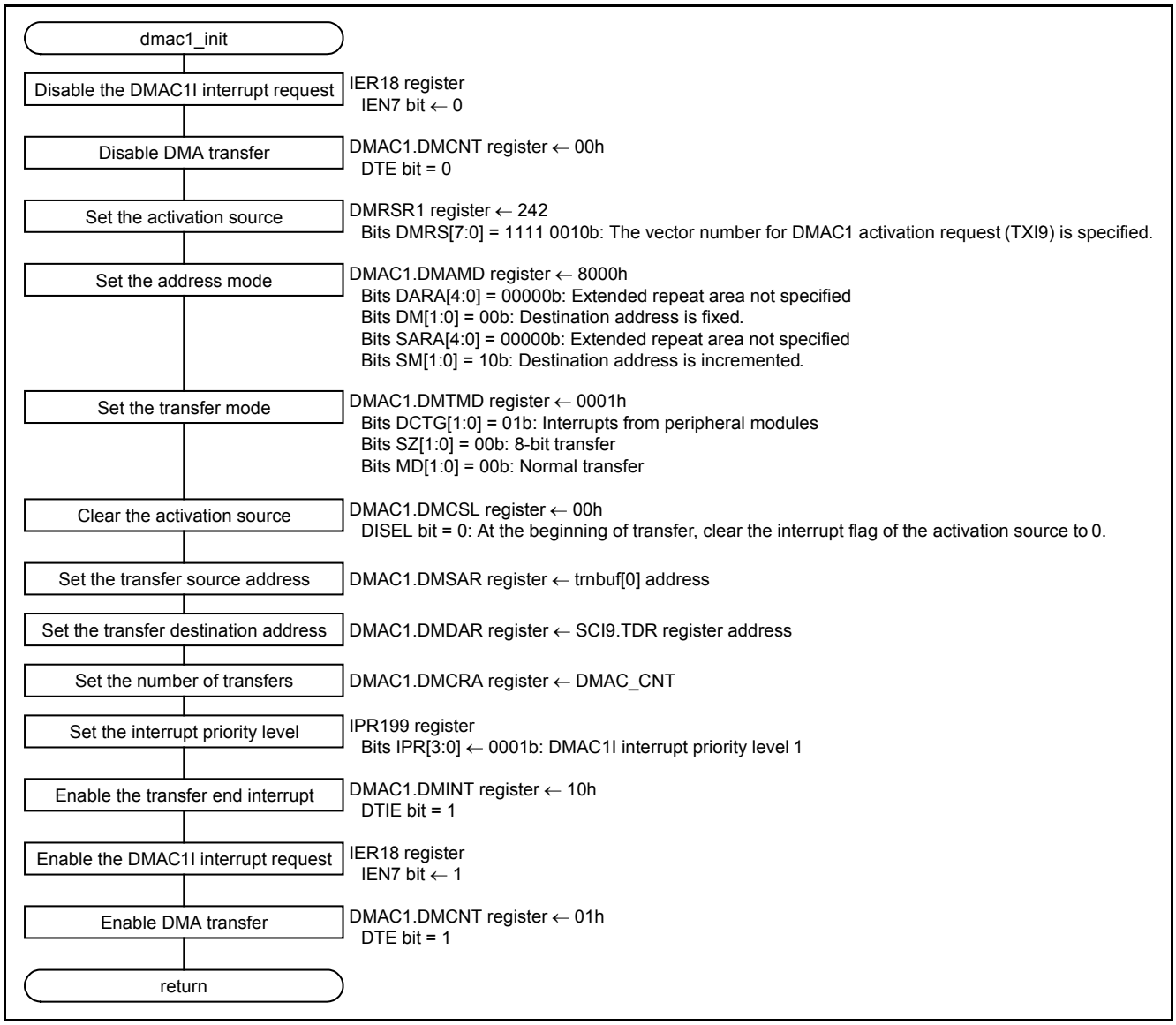

Figure 5.8 DMAC1 Initialization

5.8.7 SCI9 Transmission/Reception Start

Figure 5.9 shows SCI9 Transmission/Reception Start.

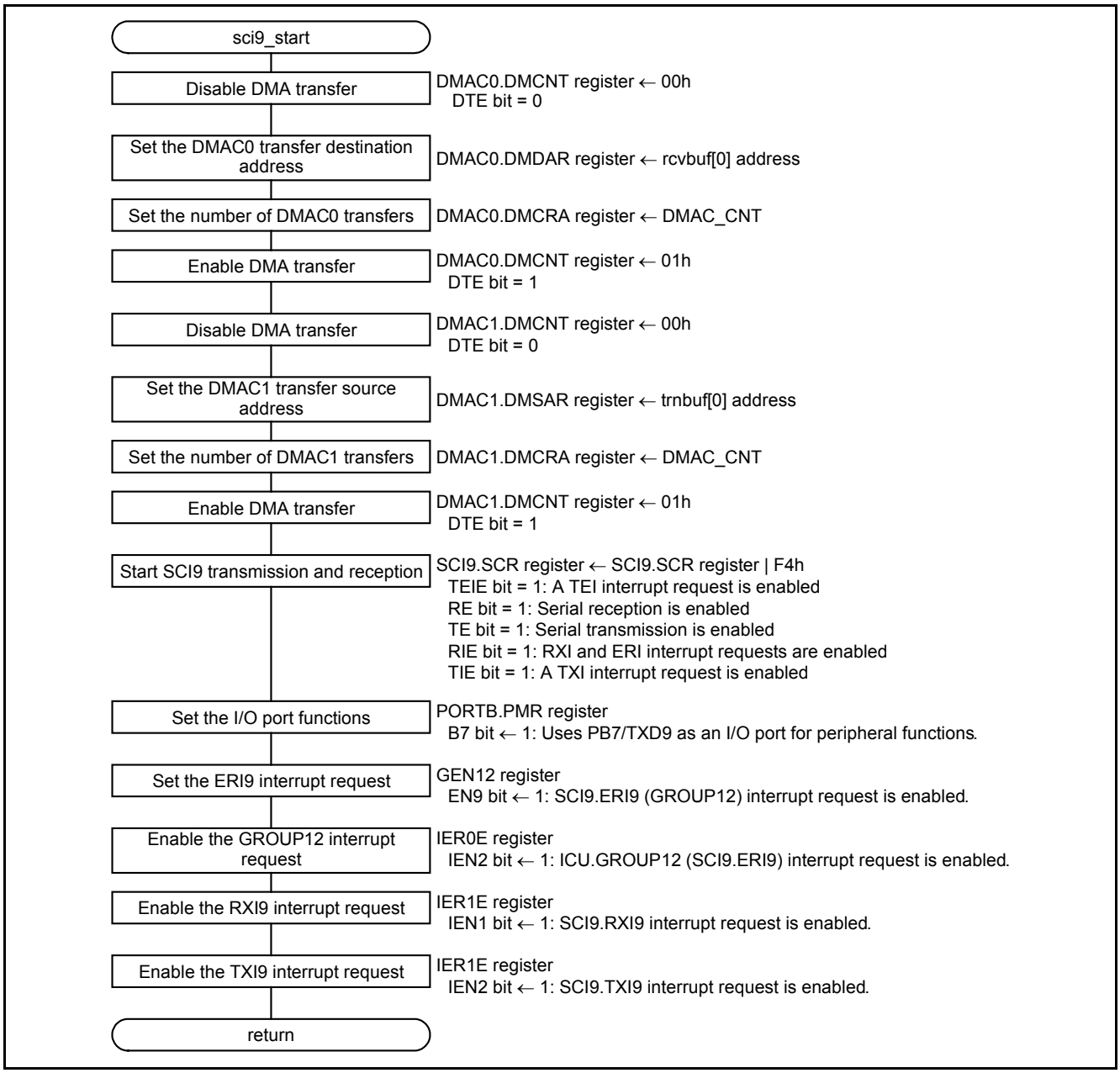

Figure 5.9 SCI9 Transmission/Reception Start

5.8.8 DMAC0 Transfer End Interrupt Handling

Figure 5.10 shows DMAC0 Transfer End Interrupt Handling.

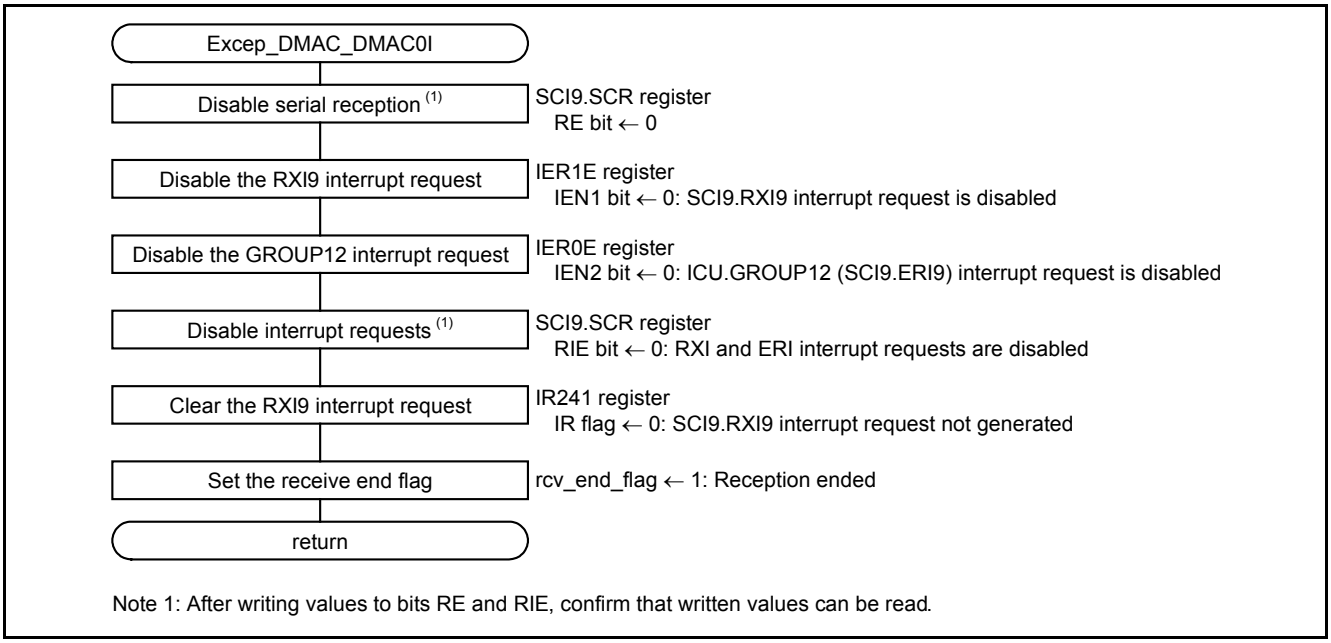

Figure 5.10 DMAC0 Transfer End Interrupt Handling

5.8.9 DMAC1 Transfer End Interrupt Handling

Figure 5.11 shows DMAC1 Transfer End Interrupt Handling.

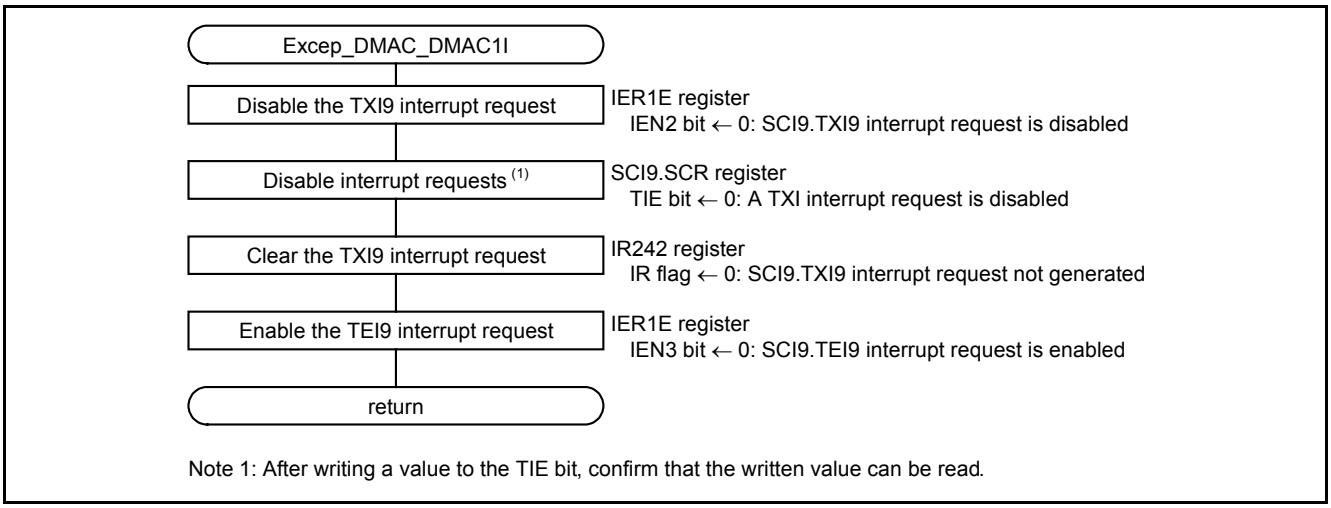

Figure 5.11 DMAC1 Transfer End Interrupt Handling

5.8.10 SCI9 Transmit End Interrupt Handling

Figure 5.12 shows SCI9 Transmit End Interrupt Handling.

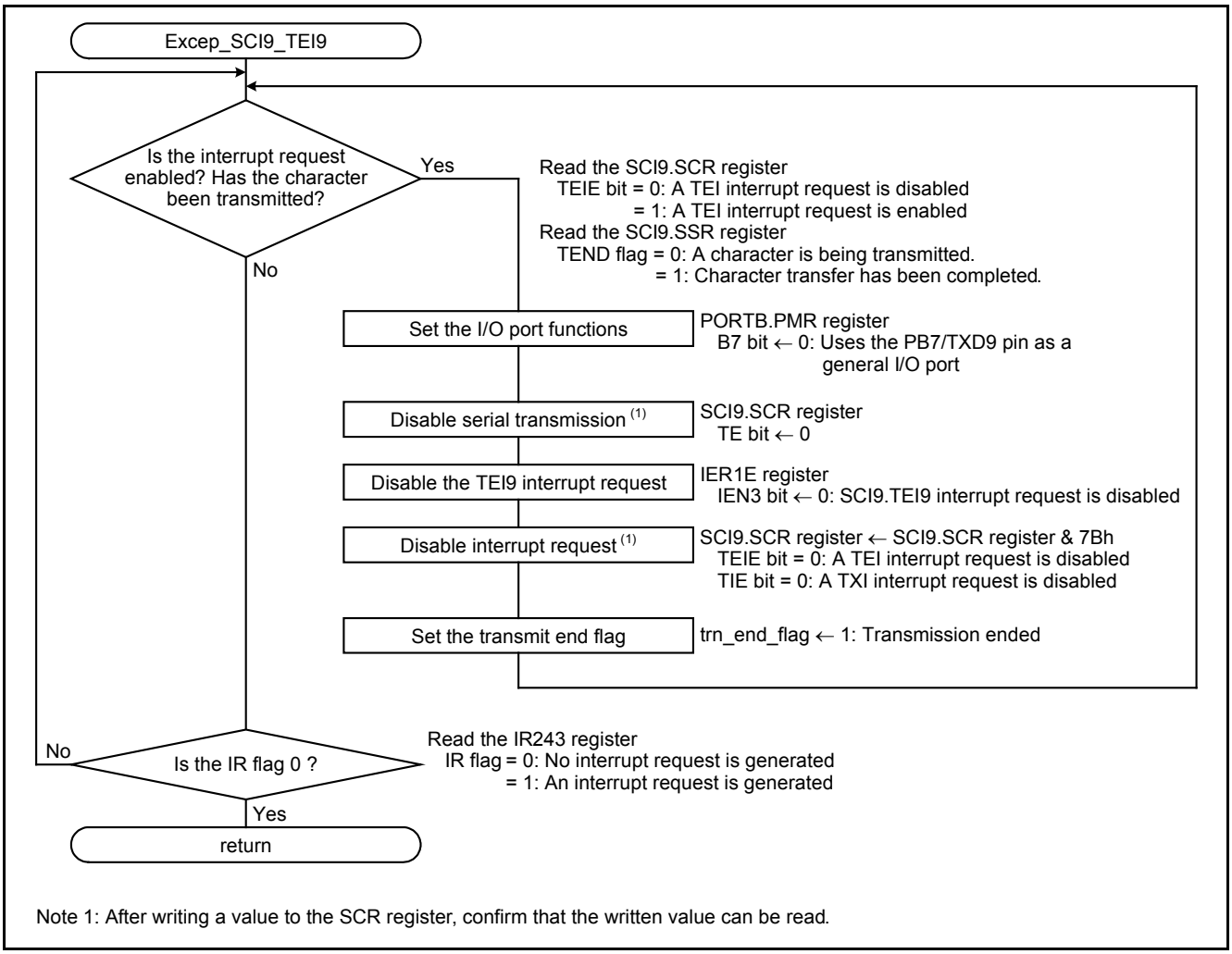

Figure 5.12 SCI9 Transmit End Interrupt Handling

5.8.11 Group 12 Interrupt Handling (SCI9 Receive Error Interrupt)

Figure 5.13 shows Group 12 Interrupt Handling (SCI9 Receive Error Interrupt).

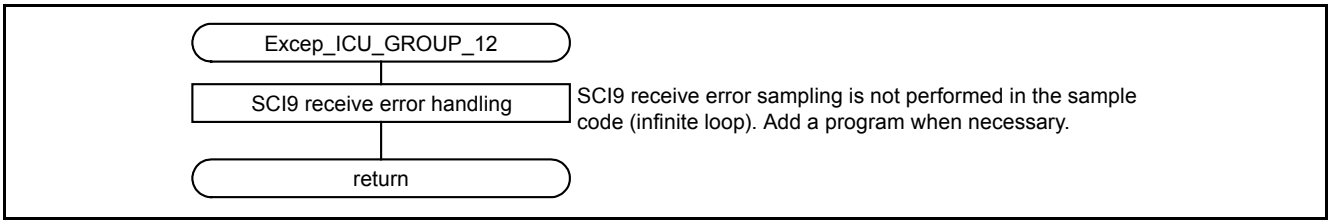

Figure 5.13 Group 12 Interrupt Handling (SCI9 Receive Error Interrupt)

6. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

7. Reference Documents

User's Manual: Hardware

RX63N Group, RX631 Group User's Manual: Hardware Rev.1.70 (R01UH0041EJ)
The latest version can be downloaded from the Renesas Electronics website.

Technical Update/Technical News

The latest information can be downloaded from the Renesas Electronics website.

User's Manual: Development Tools

RX Family C/C++ Compiler Package V.1.01 User's Manual Rev.1.00 (R20UT0570EJ)
The latest version can be downloaded from the Renesas Electronics website.

Website and Support

Renesas Electronics website

http://www.renesas.com

Inquiries

http://www.renesas.com/contact/

REVISION HISTORY	RX63N Group, RX631 Group Application Note Asynchronous SCIc Transmission/Reception Using DMACA

Rev.	Date	Description	
		Page	Summary
1.00	Apr. 1, 2014	—	First edition issued

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins

   Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment when power is supplied.

   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

   — The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. 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.
2. 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.
3. 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.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "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 etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). 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 for which it is not intended. 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 which the product is not intended by Renesas Electronics.
6. 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.
7. 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 systems manufactured by you.
8. 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.
9. 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. You should not use Renesas Electronics products or 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. When exporting the Renesas Electronics 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.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
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.

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141