RX630 Group

Communication with EEPROM Using the Serial Communications Interface (Simple I²C)

R01AN0596EJ0100
Rev.1.00
Dec 13, 2011

Introduction

This application note describes communication with a connected EEPROM using the simple I²C function of the RX630 Group SCI (serial communications interface) module.

Target Device

RX630 Group

This application note can also be used with other RX family microcontrollers that have the same I/O registers (peripheral unit control registers) as the RX630 Group products. Note, however, that since there are changes between devices, such as additional functionality in certain functions, these operations must be verified with the manual for the device actually used. When using the methods described in this application note, full testing in the actual user system is required.

Contents

1. Specifications

This sample program uses the SCI (serial communications interface) simple I^2C function to communicate with an EEPROM and write 8 bytes of data to that EEPROM. It then reads back that 8 bytes of data.

Table 1.1 lists the peripheral functions used and their uses.

Table 1.1 Peripheral Functions and their Uses

Peripheral Function	Use
SCI (serial communications interface)	• Simple I^2C function • Transfer rate: 100 kHz • Master/slave mode: single master mode

2. Conditions of Checking the Operation of the Software

The sample code described in this application note has been confirmed to run normally under the operating conditions given below.

Table 2.1 Operating Conditions

Item	Description
MCU	RX630 (R5F5630EDDFP)
Memory used for evaluation	Xin clock: 12 MHz Subclock: 32.768 kHz
Operating voltage	3.3 V
Integrated development environment	Version 4.09.00.007
C compiler	RX Standard Toolchain (V.1.0.1.0) C/C++ compiler package for RX family V.1.01.00
	Compiler options: The default settings for the integrated development environment are used.
Operating mode	Single chip mode
Version of the sample code	Version 1.40
Board used	R0K505630C001BR

3. Software Description

3.1 Operation Overview

The simple I²C bus format consists of 8 bits of data plus one acknowledge bit. The frame, which follows a start condition or restart condition, is a slave address frame and is used to specify the salve device which is the communication destination for the master device. The specified slave device is valid until either a new slave device is specified or a stop condition is issued. Each 8-bit data item in the frame is transmitted in order starting with the MSB.

Although both the SCI c and SCI d SCI modules support simple I²C mode, this application note uses channel 0 in SCI c.

3.1.1 Specifications

This application note communicates with an EEPROM and writes 8 bytes of data to that EEPROM. It then reads back that 8 bytes of data.

Figure 3.1 shows the connections used for this application note.

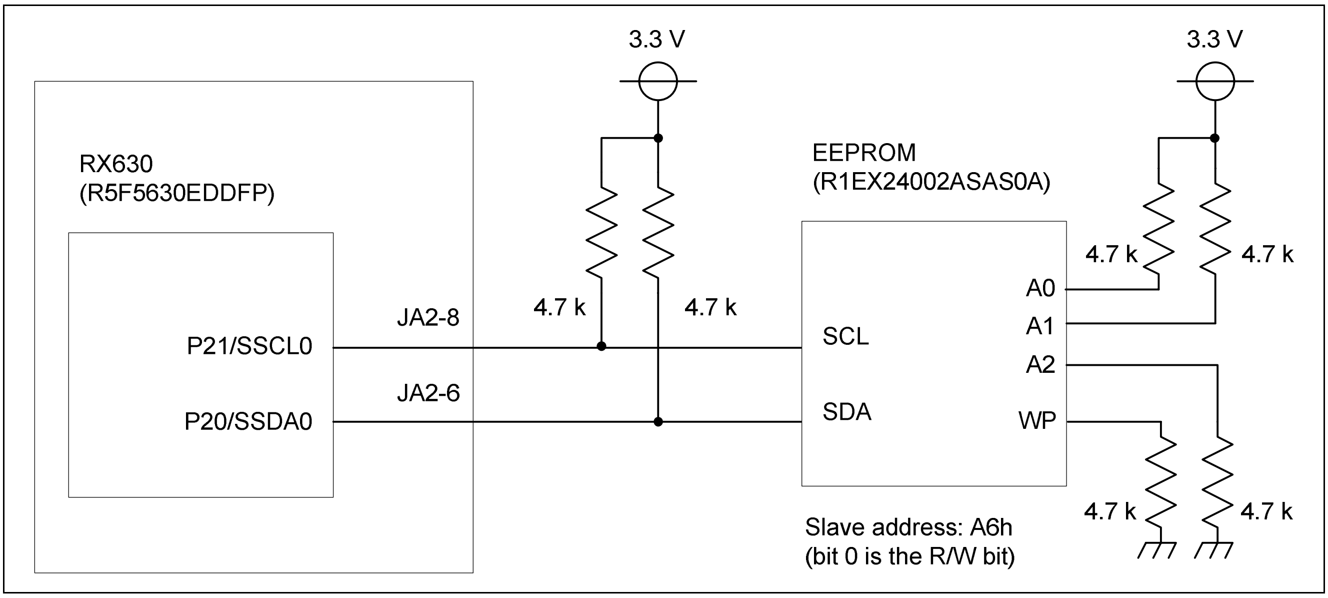

Figure 3.1 Connection Diagram

Table 3.1 lists the SCI communication function settings and conditions used in this application note.

Table 3.1 SCI Settings and Conditions

Channels used	SCI0
Communication mode	Simple I²C mode
Interrupts	Unused
Transfer rate	100 KHz
Master/slave mode	Single master mode
Address format	7-bit address format

3.1.2 Writing to EEPROM

This sample program uses master mode transmission to write to an EEPROM. The RIIC generates a start condition (S) and then transmits the EEPROM slave address. Since the eighth bit here is the R/W bit, a 0 is transmitted for writing (master transmission). It then sends sequentially the memory address, which consists of two 8-bit units, and then the data to be written to the EEPROM. The two 8-bit unit memory address transmitted here indicates the address in the EEPROM to be written. After all the data has been written the sample program issues a stop condition (P) to release the bus. Note that the memory address written by this program is 0000h.

Figure 3.2 shows examples of the signals used when writing to EEPROM.

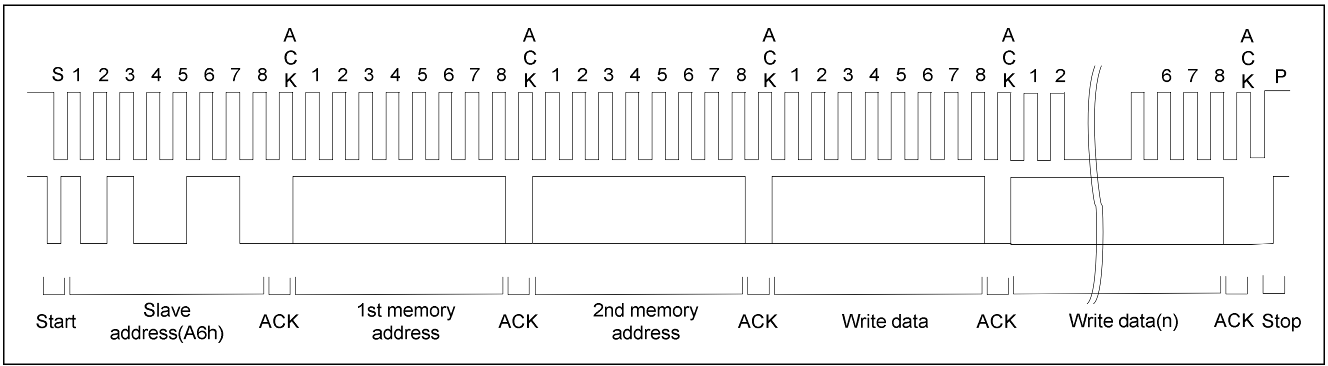

Figure 3.2 Signals when Writing to EEPROM

3.1.3 Reading from EEPROM

This sample program uses master transmission and master reception (compound format) to read data from the EEPROM. First, the RIIC generates a start condition (S) and then transmits the EEPROM slave address as two 8-bit units. In the EEPROM slave address transmission here, a 0 is transmitted for the R/W bit (master transmission). Then, it generates a restart condition (Sr) and transmits the EEPROM slave address again. In the EEPROM slave address transmission here, a 1 is transmitted for the R/W bit (master reception). After transmitting the EEPROM slave address, the RIIC reads out the EEPROM data by continuing to generate the clock. During this read operation, the RIIC issues an ACK after receiving each byte. However, it returns a NACK after the last byte, and only after the last byte. It then generates a stop condition (P). Note that the memory address read by this program is 0000h.

Figure 3.3 shows examples of the signals used when reading from EEPROM.

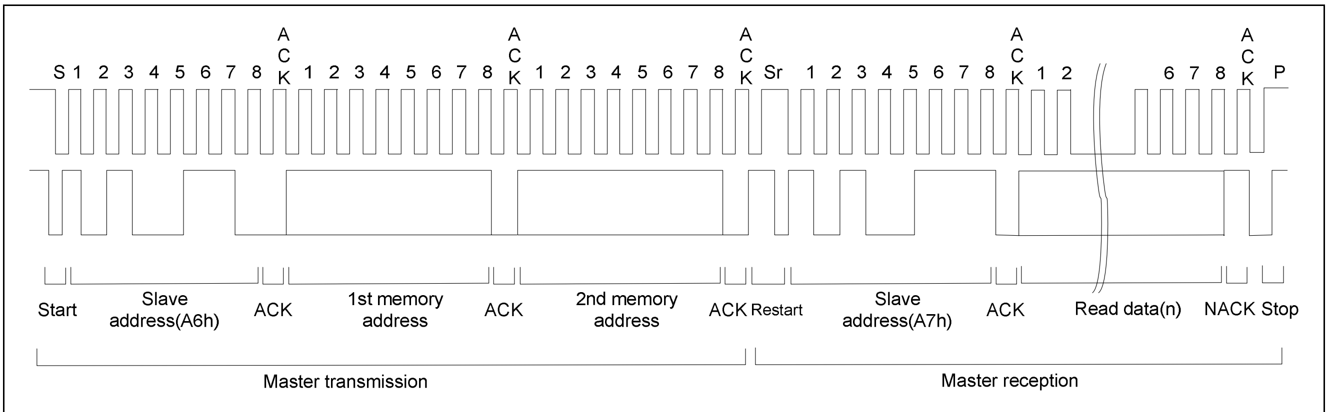

Figure 3.3 Signals when Reading from EEPROM

3.2 File Structure

Table 3.2 lists the files used in the sample code.

Table 3.2 File Structure

File Name	Function	Notes
main_i2c.c	• The main processing • Interrupt grouping control related processing • SCI interrupt processing	
dbsct.c	B and R section settings	File automatically generated by the IDE
intprg.c	Interrupt handling (The SCI interrupt handler, which is used by this program, has been removed from this file.)	File automatically generated by the IDE
resetprg.c	Reset handling	File automatically generated by the IDE
sbrk.c	sbrk() function	File automatically generated by the IDE
vecttbl.c	Vector table related processing	File automatically generated by the IDE to which option and memory settings have been added
iodefine.h	I/O register related header file	
lowsrc.h	I/O streams related header file	File automatically generated by the IDE
sbrk.h	sbrk() function header file	File automatically generated by the IDE
stacksct.h	Stack area header file	File automatically generated by the IDE
typedefine.h	Integer type definitions header file	File automatically generated by the IDE
vect.h	Vector table related header file	File automatically generated by the IDE

3.3 Constants

Table 3.3 lists the constants used by the sample code.

Table 3.3 Constants Used in the Sample Code

Constant	Set Value	Description
TARGET_SLAVE_ADDRESS	0xA6	EEPROM slave address
EEPROM_ADDRESS_LENGTH	2	EEPROM slave address length

3.4 Variables

Table 3.4 lists the global variables.

Table 3.4 Global Variables

Type	Name	Description	Functions Where Used
unsigned char	trm_eeprom_adr[2]	EEPROM slave address storage buffer (during write)	main, IIC_EepWrite
unsigned char	rcv_eeprom_adr[2]	EEPROM slave address storage buffer (during read)	main, IIC_EepRead
unsigned char	trm_buff[256]	Transmit data buffer	main, IIC_EepWrite
unsigned char	rcv_buff[256]	Receive data buffer	main, IIC_EepRead

3.5 Structures

Figure 3.4 shows the structures used in the sample code.

```
struct str_IIC_API_T
{
    unsigned char    SlvAdr;         /* Slave address */
                                     /* Since the low-order bit is the R/W bit, it should always be set to 0. */
    unsigned short   PreCnt;         /* Memory address counter */
    unsigned char    *pPreData;      /* Pointer to memory address storage buffer */
                                     /* On write: Address in EEPROM to which data is written (write destination) */
                                     /* On read: Address in EEPROM from which data is read (read source) */
    unsigned long    RWCnt;          /* Data counter */
                                     /* On write: Number of data items to be written to EEPROM */
                                     /* On read: Number of data items to be read from EEPROM */
    unsigned char    *pRWData;       /* Pointer to data storage buffer */
                                     /* On write: Data storage source for data to be written to EEPROM */
                                     /* On read: Data storage destination for data to be read from EEPROM */
};

typedef struct str_IIC_API_T IIC_API_T;
```

Figure 3.4 Structures Used in Sample Code

3.6 Functions

Table 3.5 lists the functions defined in the sample code.

Table 3.5 Functions

Function Name	Overview
mcu_init	CPU initialization
clock_setting	CPU clock settings
peripheral_init	Peripheral function initialization
SCI_init	Serial communications interface (SCI) related initialization
IIC_EepWrite	Writes to the EEPRIM
IIC_EepRead	Reads from the EEPROM
Create_StartCondition	Start condition generation
Create_RestartCondition	Restart condition generation
Create_StopCondition	Stop condition generation

3.7 Function Specifications

This section lists the specifications of the functions in the sample code.

Name	mcu_init
Overview	CPU initialization
Header	Iodefine.h
Declaration	void mcu_init(void)
Description	Sets the CPU clock.
Arguments	None
Return values	None
Notes	

Name	clock_setting
Overview	Sets the CPU clock.
Header	Iodefine.h
Declaration	void clock_setting(void)
Description	• Stops the subclock oscillator. • Stops the high-speed clock oscillator. • Sets the oscillator stabilization time for the main clock oscillator to 131,072 cycles. • Sets the PLL oscillator stabilization time to 4,194,304 cycles. • Sets the PLL frequency multiplier to 16x. • Sets the main clock oscillator to the operating state. • Sets the PLL circuit to the operating state. • Sets the system clock to divided by 2, the FlashIF clock to divided by 4, the external bus clock to divided by 4, and the peripheral module clock to divided by 4. • Sets the clock source to be the PLL circuit.
Arguments	None
Return values	None
Notes	

Name	peripheral_init
Overview	Peripheral function initialization
Header	Iodefine.h
Declaration	void peripheral_init(void)
Description	Initializes SCI related items.
Arguments	None
Return values	None
Notes	

Name	SCI_init
Overview	Serial communications interface (SCI) related initialization
Header	Iodefine.h
Declaration	void SCI_init(void)
Description	• Clears SCI0 module stop function. • Sets pins P21 (SSCL0) and P20 (SSDA0) to be used as SCI0 pins. • Disables SCI0 serial reception operation (RE) and serial transmission operation (TE). • Sets the SCI0 SSCL0 and SSDA0 pins to the high-impedance state. • Sets the SCI0 clock to be PCLK/4 clock. • Sets SCI0 to serial interface mode, MSB first. • Sets the SCI0 baud rate to 100 kHz. • Sets the noise filter to use a divided-by-1 clock. • Sets SCI0 to simple I2C mode. • Sets the SCI0 smart card mode selection to serial interface mode. • Enables the SCI0 SS#0 pin function. • Sets the SCI0 interrupt level to level 1. • Clears the SCI0 interrupt request flag. • Sets the SCI0 interrupt request enable to interrupts disable.
Arguments	None
Return values	None
Notes	

Name	IIC_EepWrite
Overview	Writes to the EEPRIM
Header	Iodefine.h
Declaration	void IIC_EepWrite(IIC_API_T iic_buf)
Description	Writes to the EEPROM using simple I2C master transmission operation.
Arguments	IIC_API_T iic_buf: Structure used as an argument
Return values	None
Notes	

Name	IIC_EepRead
Overview	Reads from the EEPROM
Header	Iodefine.h
Declaration	void IIC_EepRead(IIC_API_T iic_buf)
Description	Reads from the EEPROM using simple I2C master transmission operation and master reception operation.
Arguments	IIC_API_T iic_buf: Structure used as an argument
Return values	None
Notes	

Name	Create_StartCondition
Overview	Start condition generation
Header	Iodefine.h
Declaration	void Create_StartCondition (void)
Description	Generates a start condition.
Arguments	None
Return values	None
Notes	

Name	Create_RestartCondition
Overview	Restart condition generation
Header	Iodefine.h
Declaration	void Create_RestartCondition (void)
Description	Generates a restart condition.
Arguments	None
Return values	None
Notes	

Name	Create_StopCondition
Overview	Stop condition generation
Header	Iodefine.h
Declaration	void Create_StopCondition (void)
Description	Generates a stop condition.
Arguments	None
Return values	None
Notes	

3.8 Flowcharts

3.8.1 Main Processing

Figure 3.5 shows the main processing flowchart.

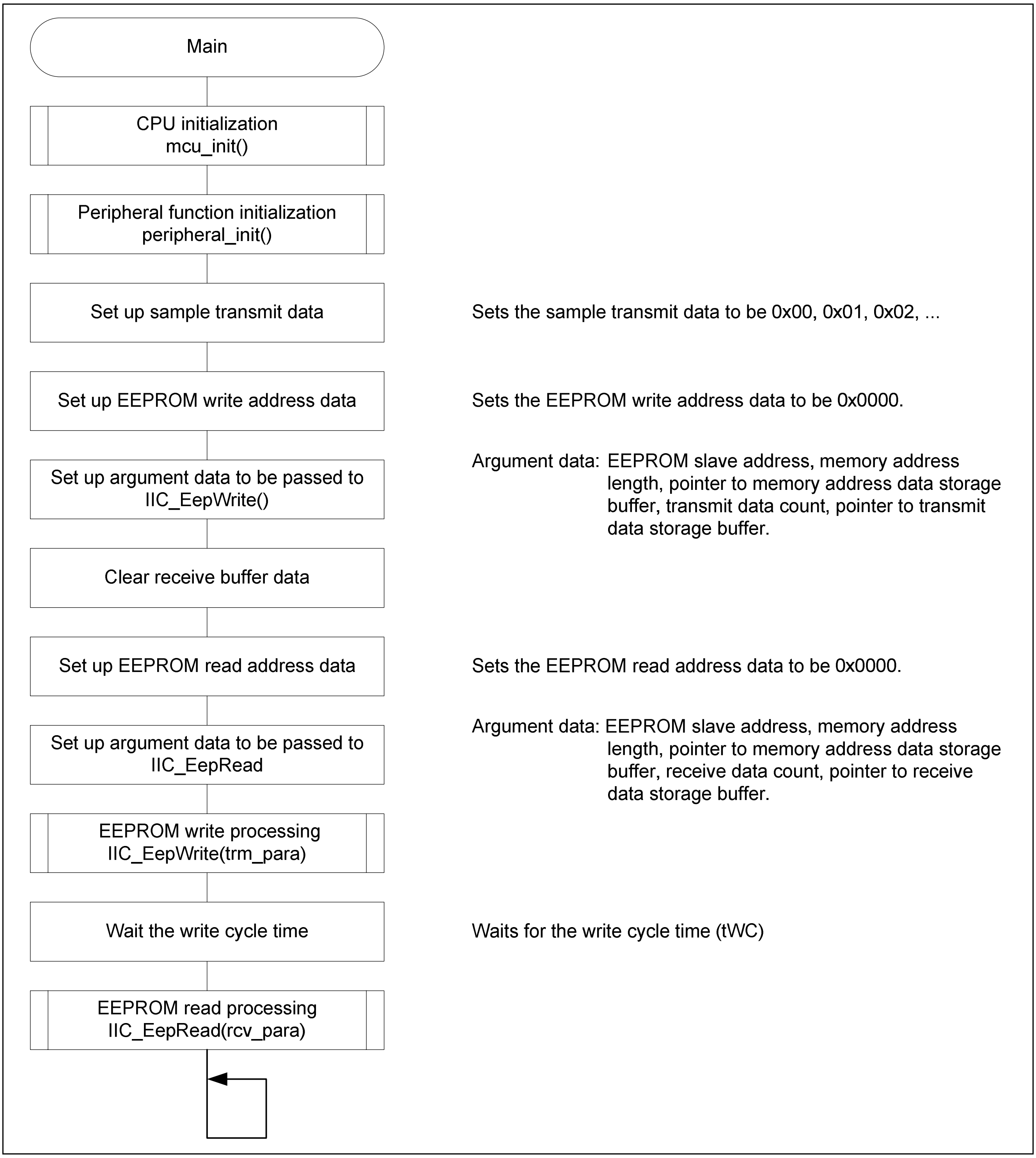

Figure 3.5 Main Processing

3.8.2 CPU Initialization

Figure 3.6 shows the flowchart for CPU initialization.

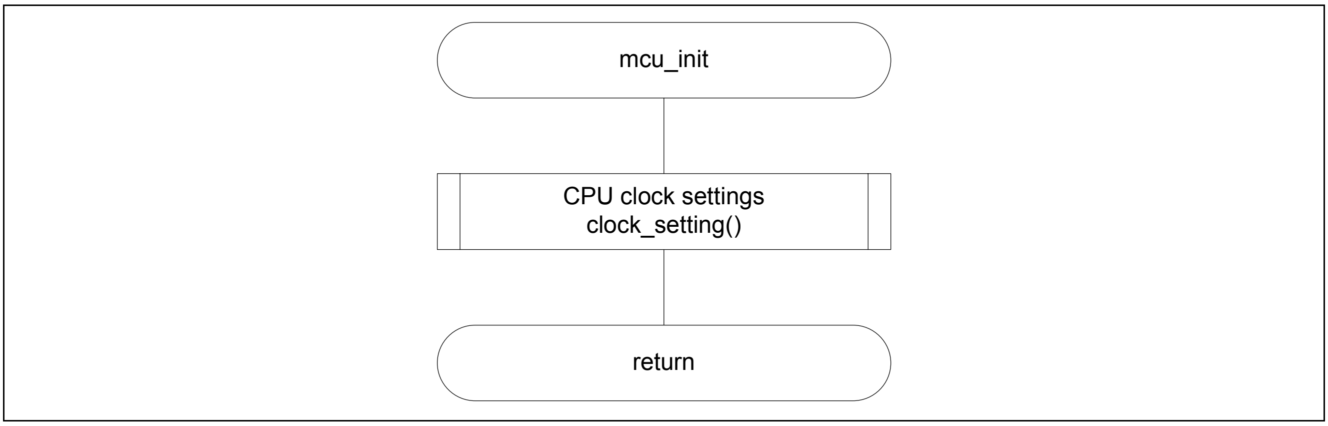

Figure 3.6 CPU Initialization

3.8.3 CPU Clock Settings

Figure 3.7 shows the flowchart for the CPU clock settings.

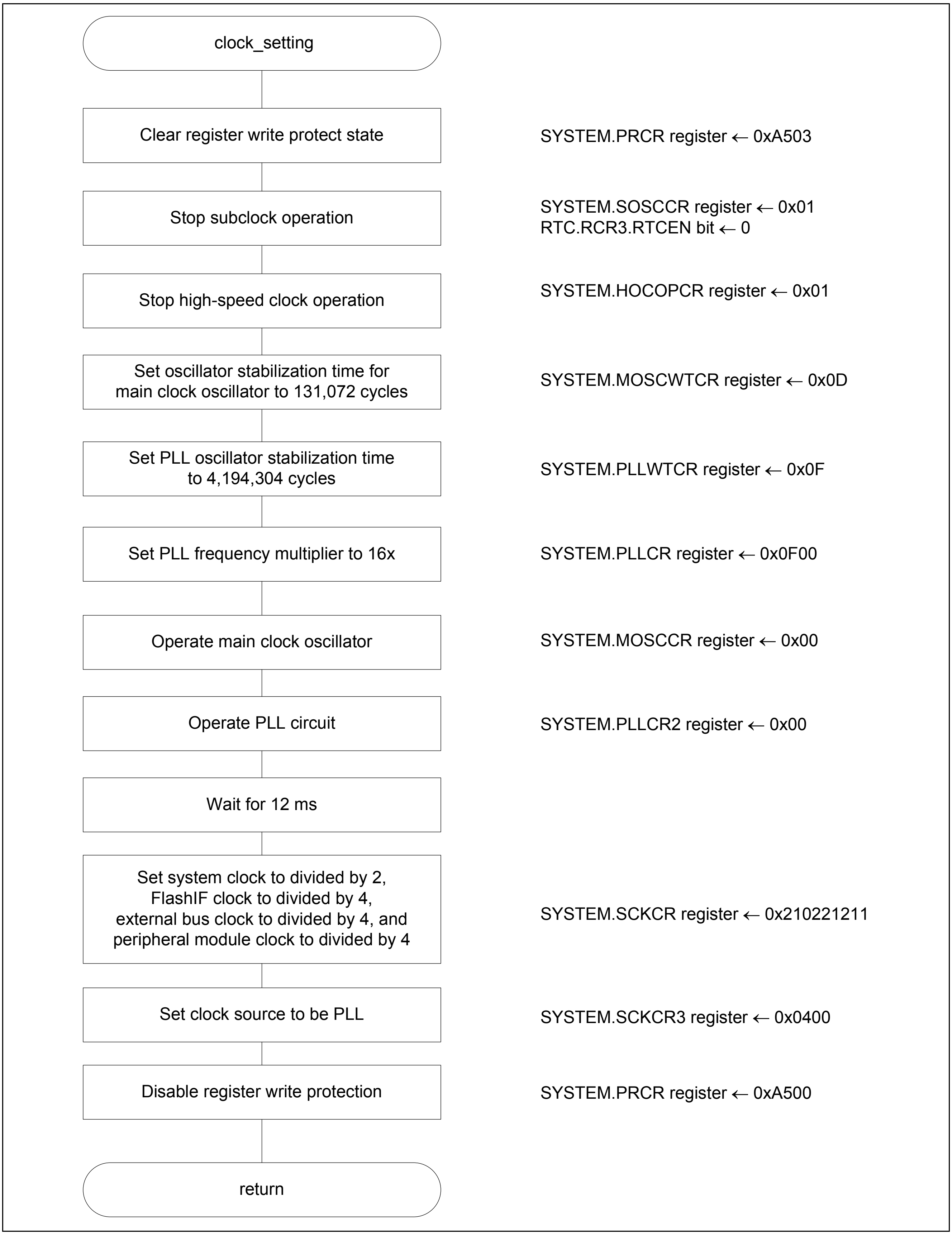

Figure 3.7 CPU Clock Settings

3.8.4 Peripheral Function Initialization

Figure 3.8 shows the flowchart for peripheral function initialization.

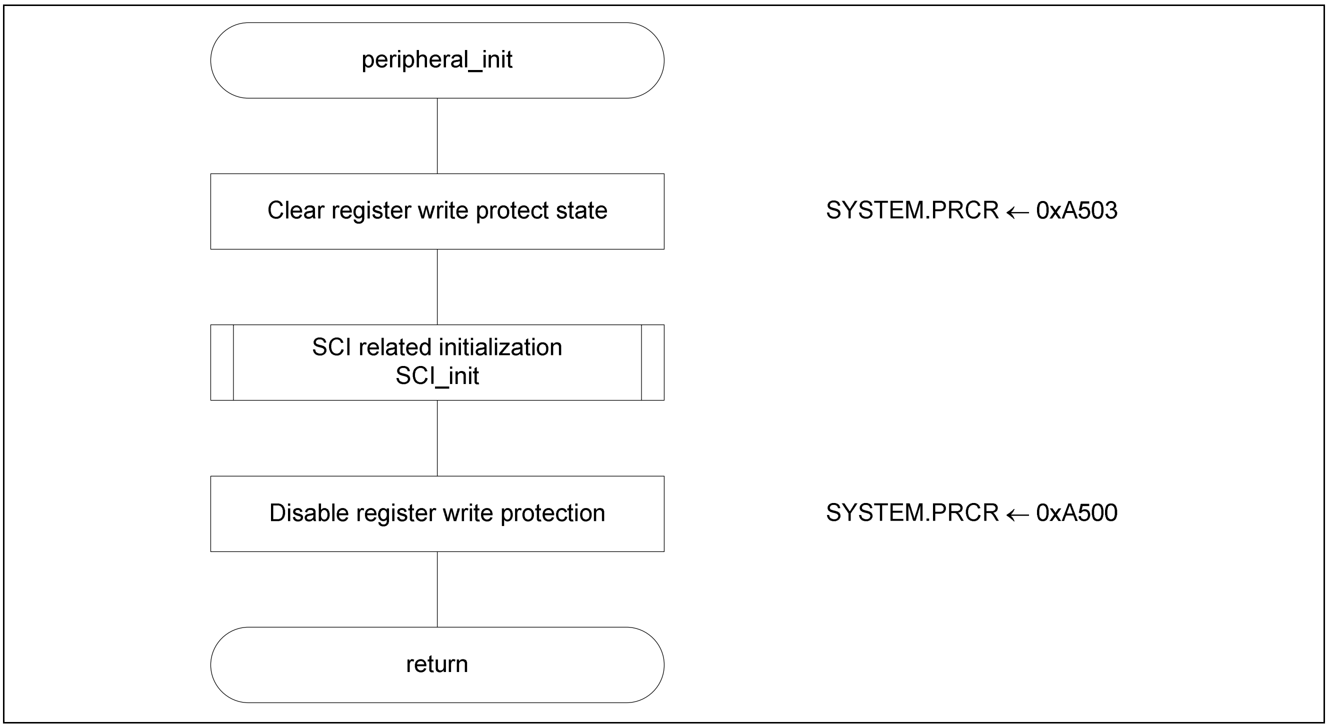

Figure 3.8 Peripheral Function Initialization

3.8.5 Serial Communications Interface (SCI) Related Initialization

Figures 3.9 and 3.10 show the flowcharts for the serial communications interface (SCI) related initialization.

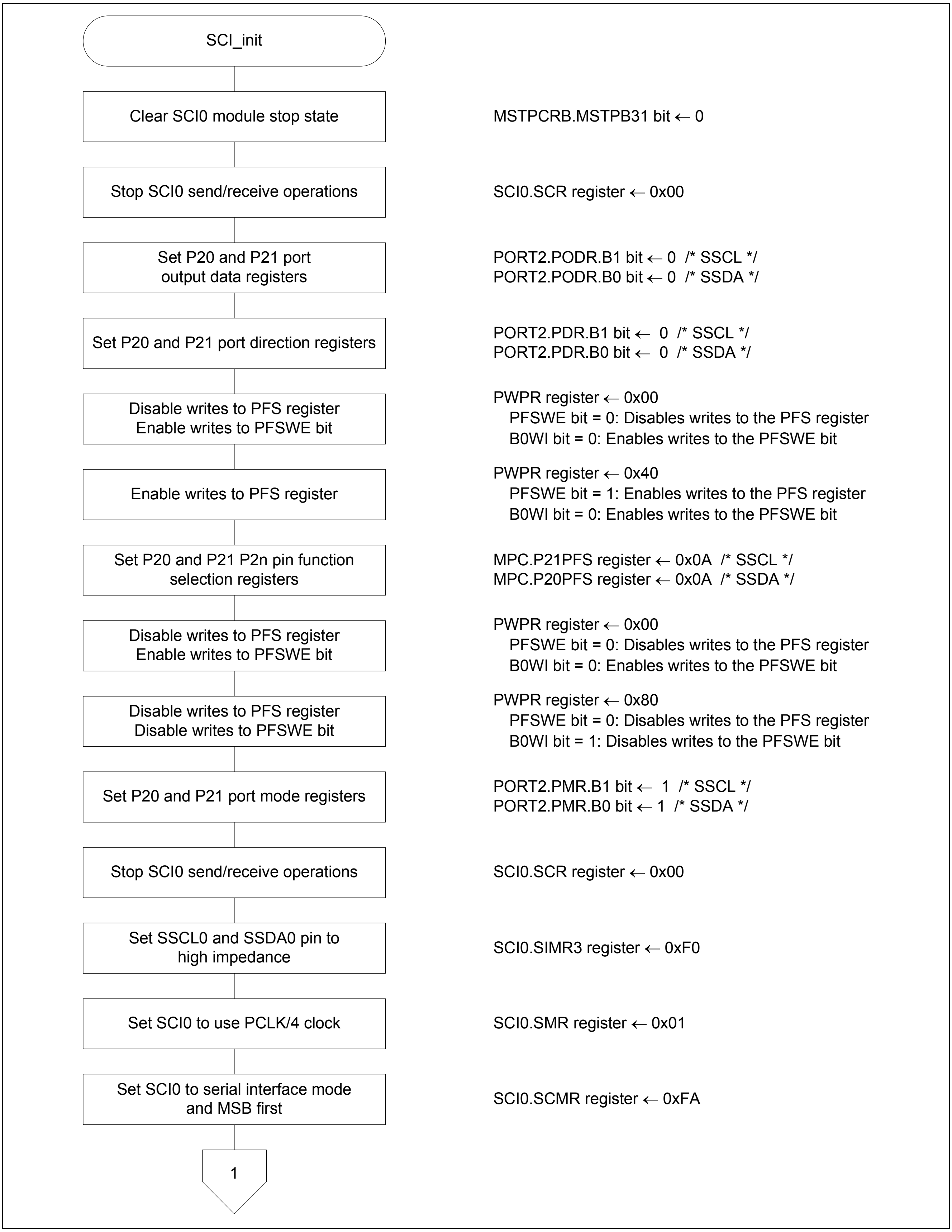

Figure 3.9 Serial Communications Interface (SCI) Related Initialization (1)

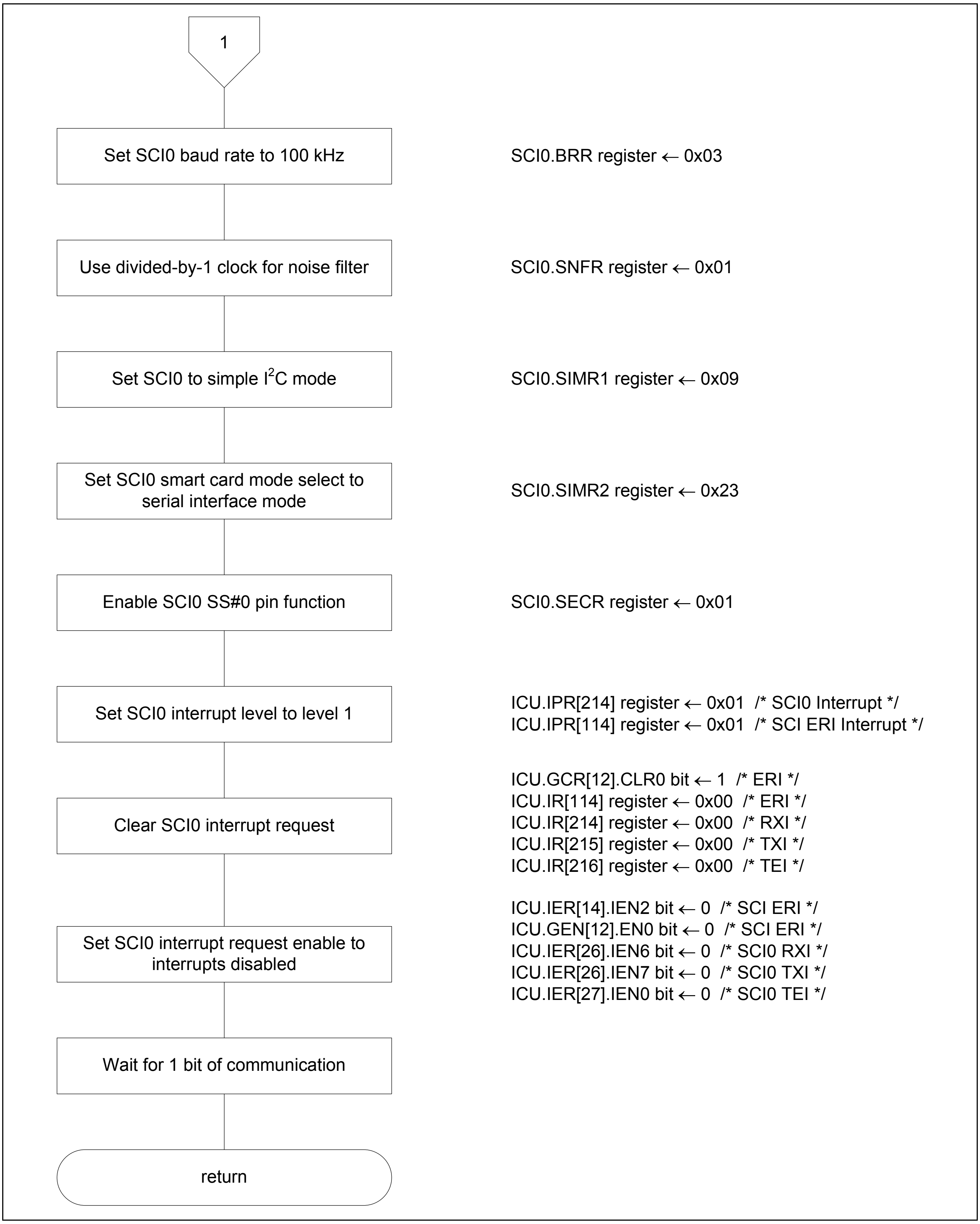

Figure 3.10 Serial Communications Interface (SCI) Related Initialization (2)

3.8.6 EEPROM Write Processing

Figure 3.11 shows the flowchart for EEPROM write processing.

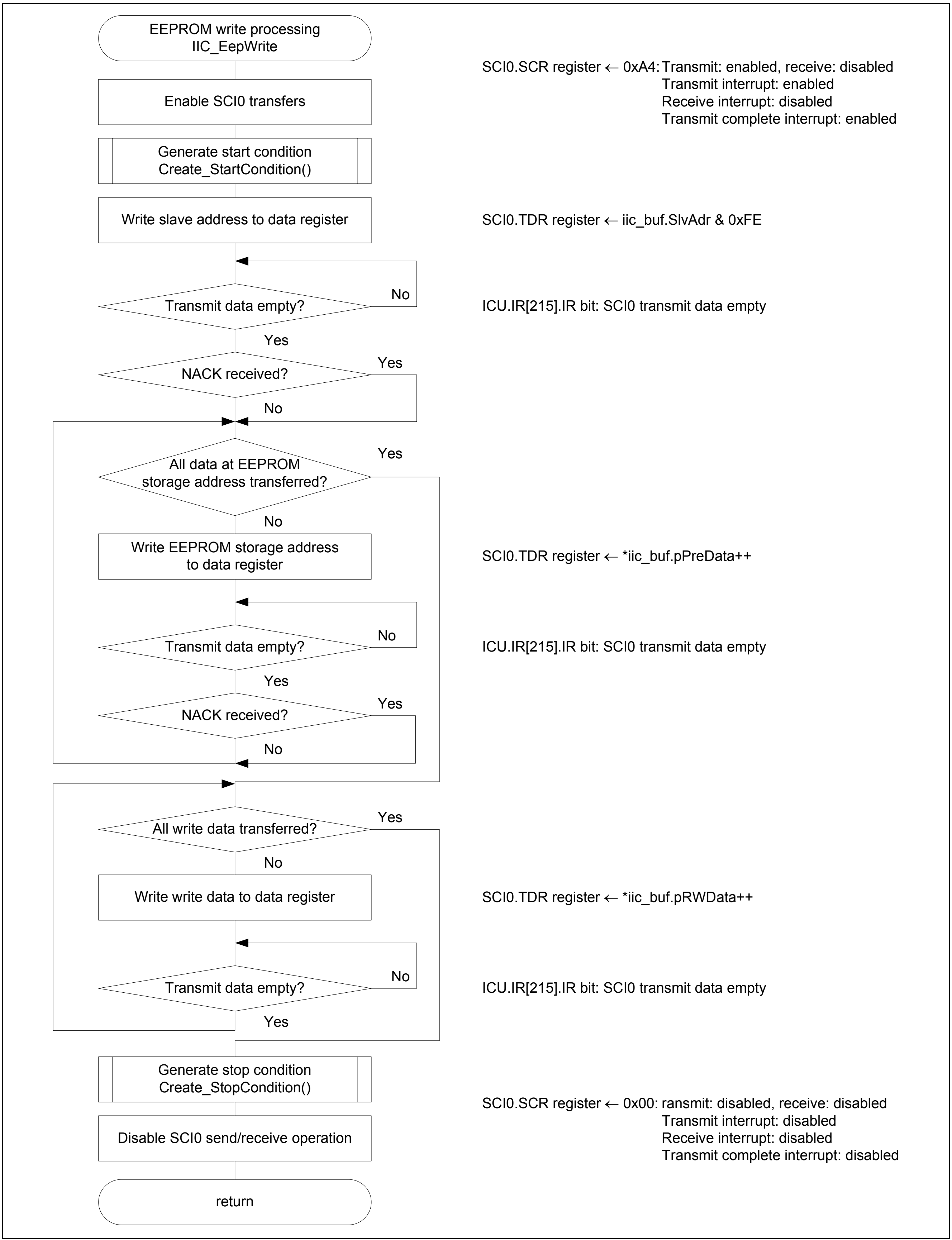

Figure 3.11 EEPROM Write Processing

3.8.7 EEPROM Read Processing

Figures 3.12 and 3.13 show the flowcharts for EEPROM read processing.

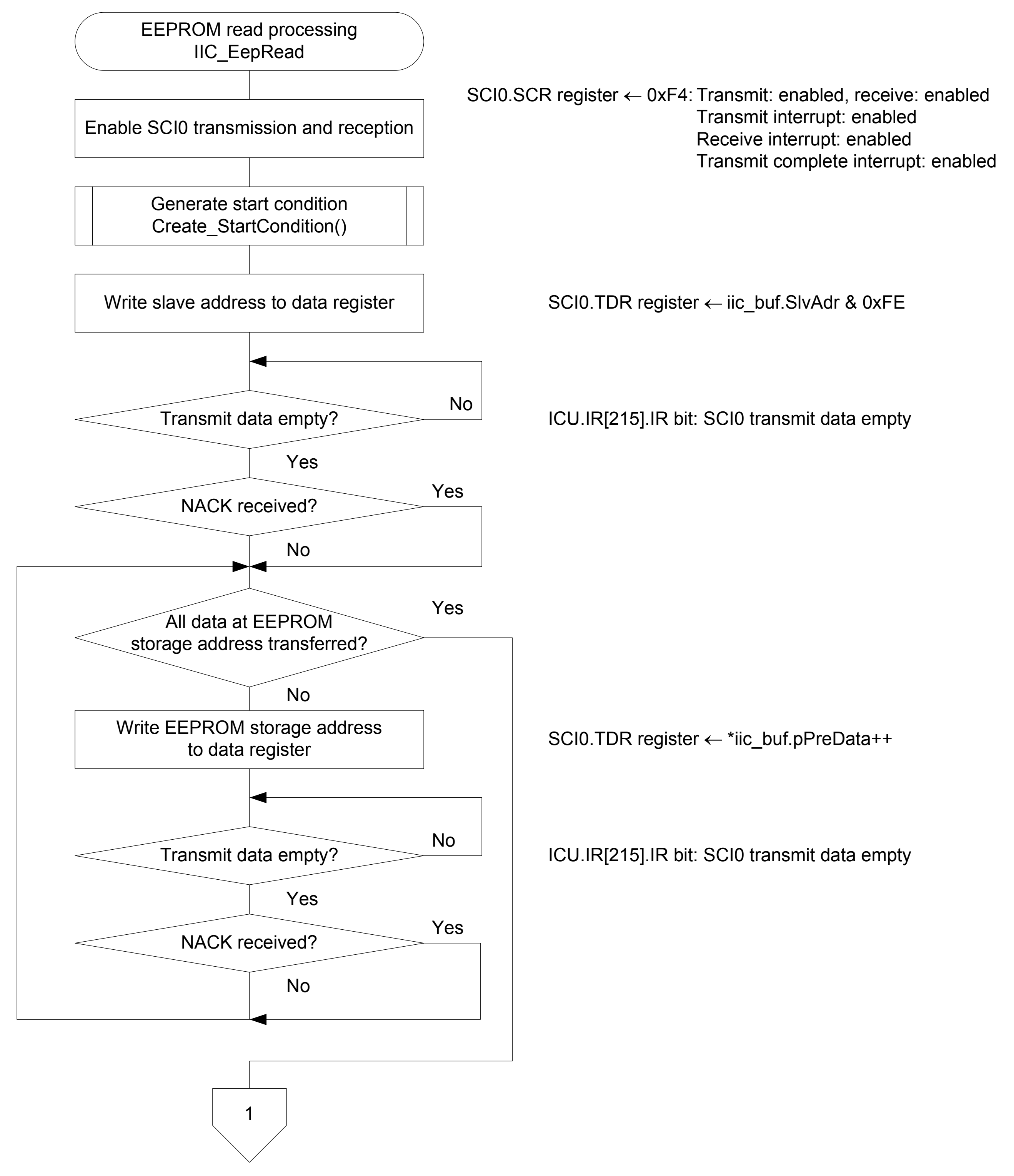

Figure 3.12 EEPROM Read Processing (1)

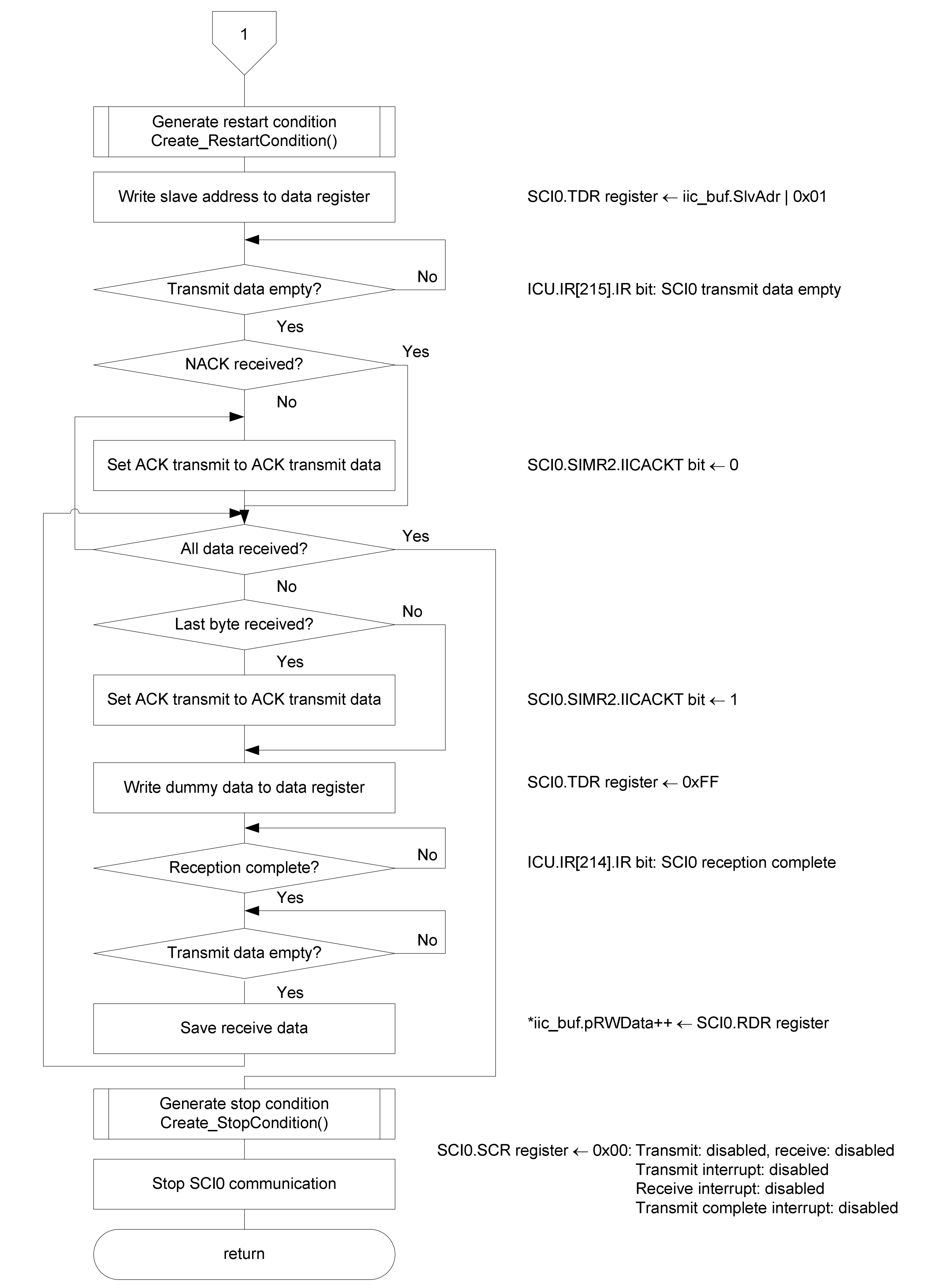

Figure 3.13 EEPROM Read Processing (2)

3.8.8 Start Condition Generation

Figure 3.14 shows the flowchart for start condition generation.

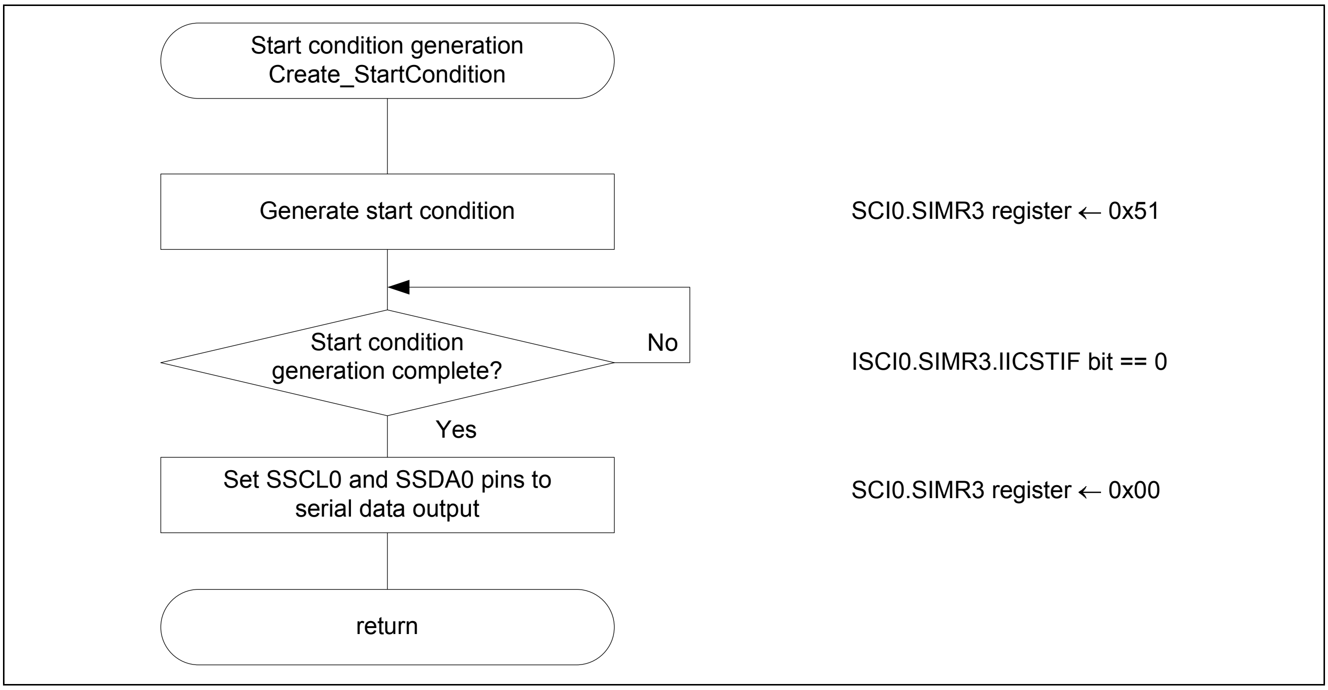

Figure 3.14 Start Condition Generation

3.8.9 Restart Condition Generation

Figure 3.15 shows the flowchart for restart condition generation.

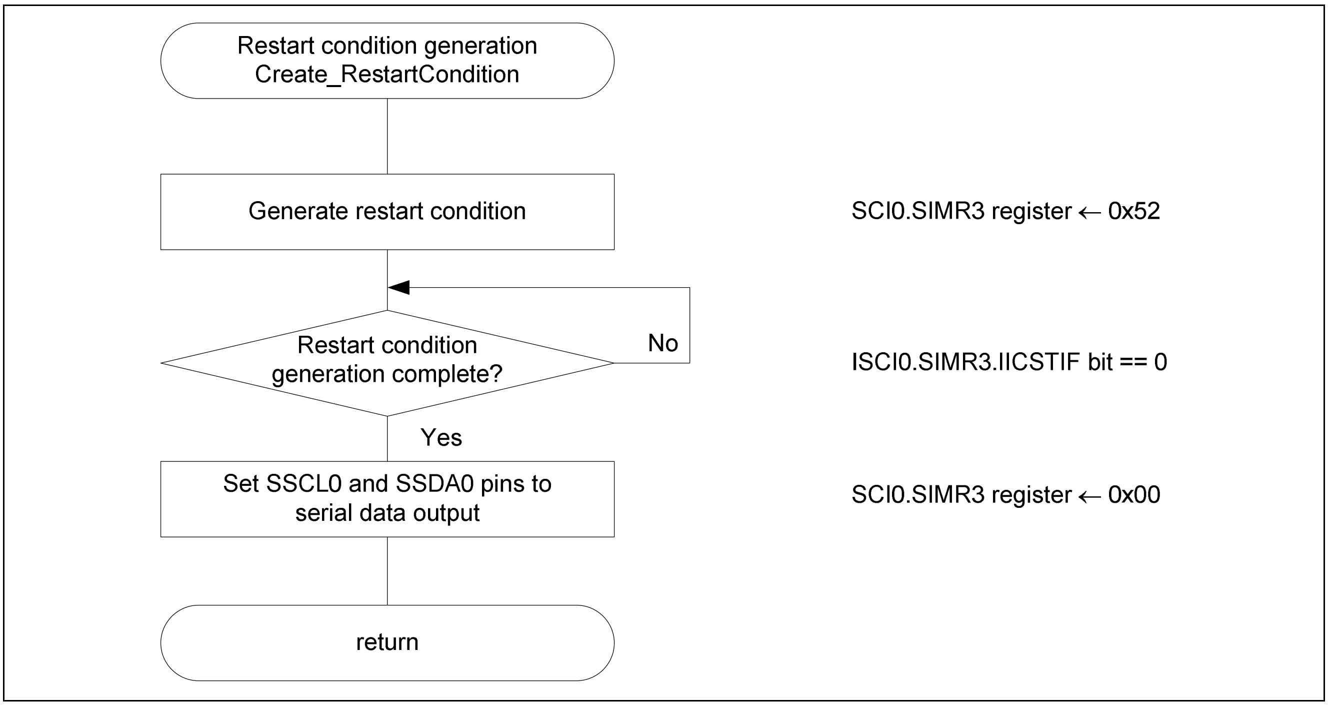

Figure 3.15 Restart Condition Generation

3.8.10 Stop Condition Generation

Figure 3.16 shows the flowchart for stop condition generation.

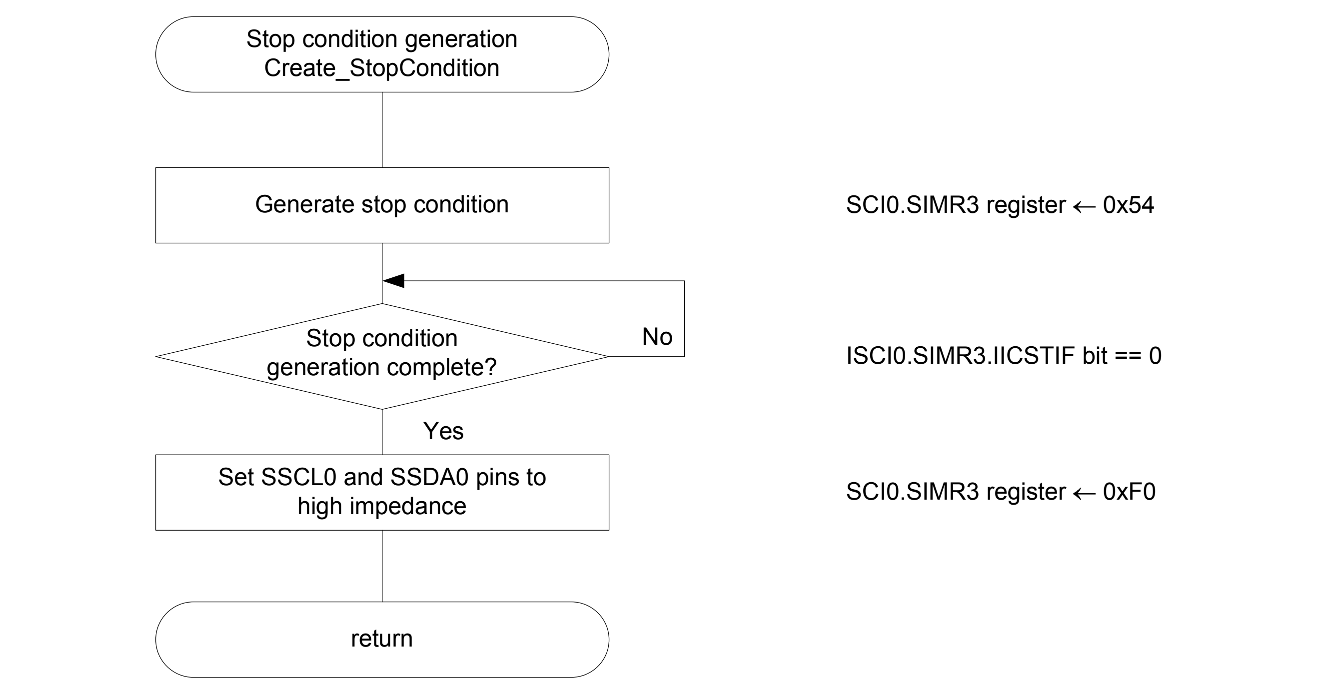

Figure 3.16 Stop Condition Generation

4. Sample Programs

The sample program can be downloaded from the Renesas Electronics Web site.

5. Reference Documents

- RX630 Group User's Manual: Hardware, Rev.1.00
  (The latest version can be downloaded from the Renesas Electronics Web site.)
- Technical Updates/Technical News
  (The latest information can be downloaded from the Renesas Electronics Web site.)
- C Compiler Manual
  RX Family C/C++ Compiler Package User's Manual V.1.0.1.0
  (The latest version can be downloaded from the Renesas Electronics Web site.)

Website and Support

Renesas Electronics Website

http://www.renesas.com/

Inquiries

http://www.renesas.com/inquiry

All trademarks and registered trademarks are the property of their respective owners.

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Dec.13.11	—	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 accord 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. 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.

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.
2880 Scott Boulevard Santa Clara, CA 95050-2554, 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: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

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: +852-2886-9318, Fax: +852 2886-9022/9044

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

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

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.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

© 2011 Renesas Electronics Corporation. All rights reserved.
Colophon 1.1