SH7216 Group

Using IIC3 to Reprogram Flash Memory in User Program Mode

R01AN0291EJ0220
Rev.2.20
May 18, 2011

Introduction

In the sample application, the user program mode of the SH7216 is used to reprogram the on-chip flash memory. The data used to program the on-chip flash is stored in an external device connected to the SH7216, and the I^2C bus interface 3 (IIC3) is used for data transfer.

The program used to reprogram the on-chip flash in the sample application is located in the user MAT of the SH7216. The simple flash API (standard API) for the SH-2 and SH-2A, supplied by Renesas Electronics, is used to reprogram the on-chip flash.

Target Device

SH7216

Contents

1. Introduction.. 2

2. Overview and Functions 4

3. Operation of Sample Program 9

1. Introduction

In the sample application, the SH7216 receives data from a connected external device and reprograms the on-chip flash in user program mode.

1.1 Specifications

Figure 1 shows a system overview of the sample application.

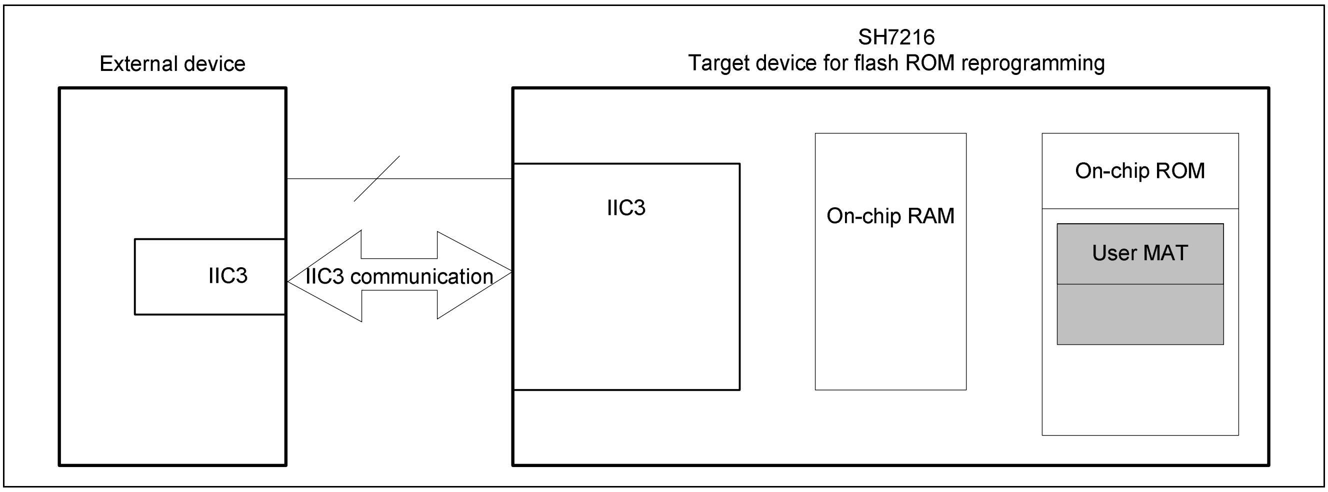


Figure 1.1 System Configuration

- The SH7216 operating mode is user program mode.
- The data to be written to the SH7216 is stored in the external device.
- The SH7216 and external device are connected by the IIC3, via which the write data is transferred.
- The SH7216 executes a program in on-chip RAM during reprogramming of the on-chip flash.
- The SH7216 has two data buffer areas (256 bytes each), which are used in parallel for writing to the on-chip flash and for downloading data.
- The standard API is used to program and erase the on-chip flash.

Table 1 SH7216 Mode Pin Settings

	Pin Settings		
Mode	**FWE**	**MD1**	**MD0**
User program mode	1	1	0

1.2 Functions Used

- I^{2}C bus interface 3 (IIC3)
- Dedicated sequencer for on-chip flash (FCU)

1.3 Applicable Conditions

MCU	SH7216
Operating frequency	Internal clock: 200 MHz Bus clock: 50 MHz Peripheral clock: 50 MHz MTU2S clock: 100 MHz AD clock: 50 MHz
Integrated development environment	Renesas Electronics High-performance Embedded Workshop, Ver. 4.07.00
C compiler	Renesas Electronics SuperH RISC engine Family C/C++ Compiler Package, Ver. 9.03.00, Release 02
Compile options	High-performance Embedded Workshop default settings (-cpu=sh2afpu -pic=1 –object=”$(CONFIGDIR)\$(FILELEAF).obg” -debug –gbr=auto –chgincpath –errorpath –global_volatile=0 -opt_range=all –infinite_loop=0 –del_vacant_loop=0 -struct_alloc=1 –nologo)

1.4 Related Application Notes

The following application note is related to this application note. Refer to it as necessary in conjunction with this application note.

SH Family: Simple Flash API for SH-2 and SH-2A

2. Overview and Functions

In the sample application, the SH7216 and external device are connected by the IIC3. In addition, a dedicated sequencer (FCU) is used on the SH7216 to program and erase the on-chip flash.

2.1 Description of Functions Used

2.1.1 I²C Bus Interface 3 (IIC3) Functions

The IIC3 provides a subset of the functions of the Inter IC Bus (I²C Bus) interface standard developed by Philips. In the sample application, it is used the transfer between the SH7216 and the external device the data to be written to the on-chip flash.

Figure 2 is a block diagram of the IIC3.

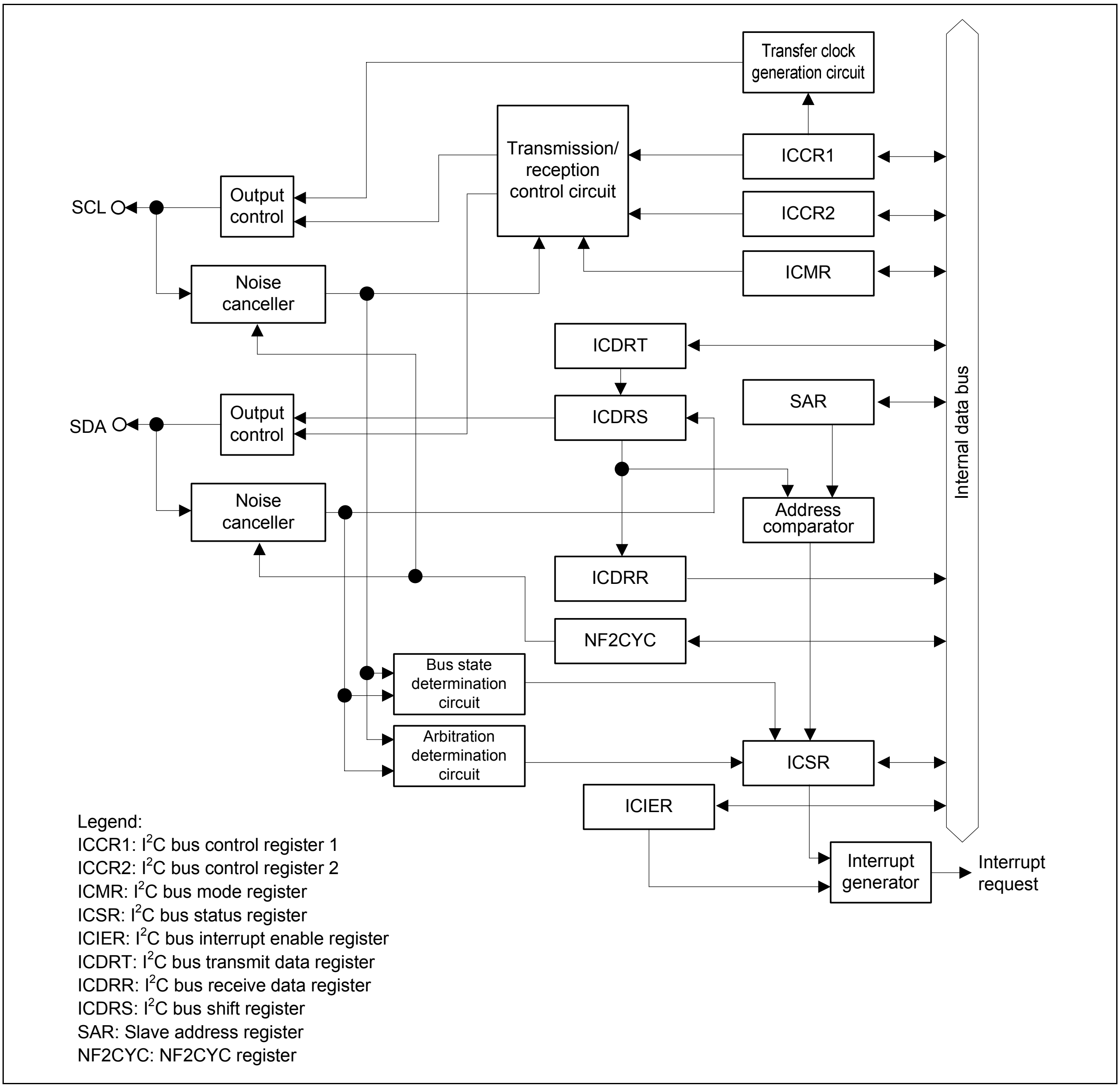
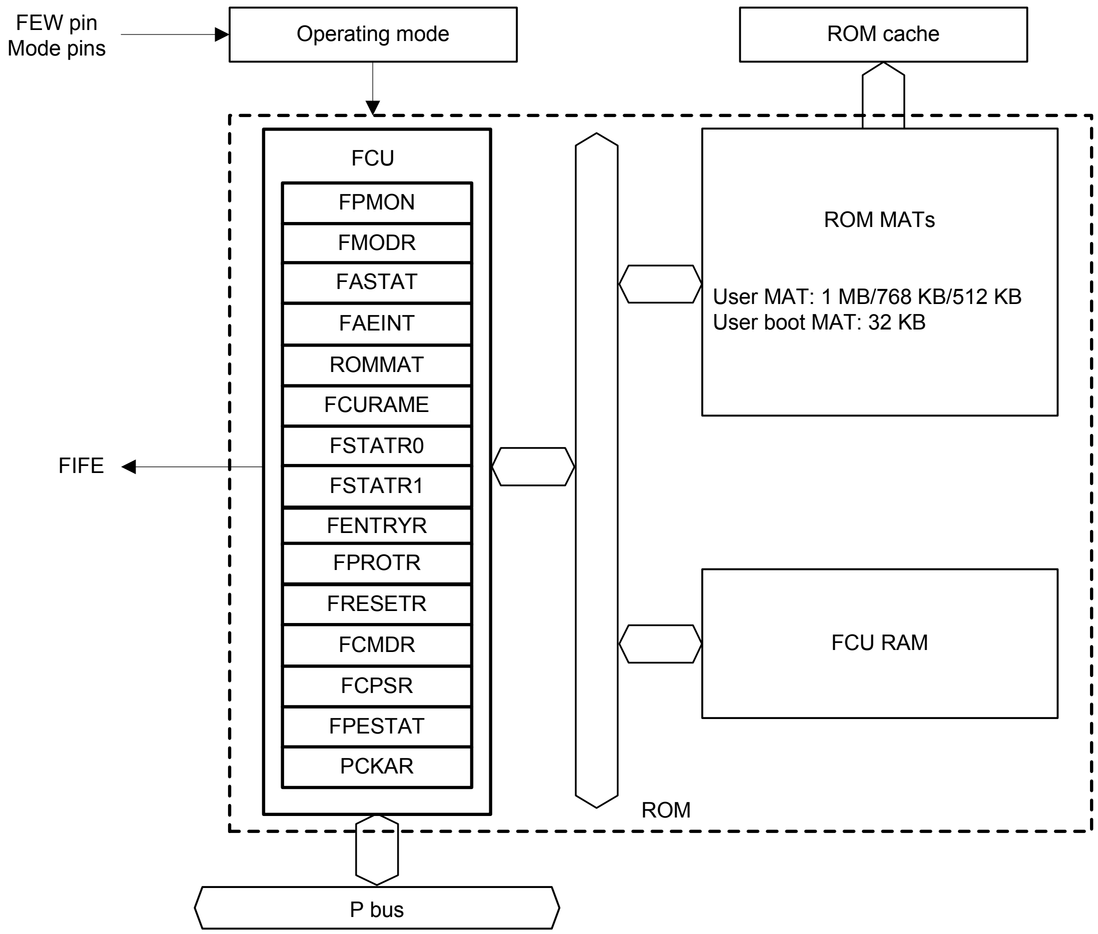


Figure 2 Block Diagram of IIC3

2.1.2 On-Chip Flash Dedicated Sequencer (FCU) Functions

The SH7216 uses the FCU to reprogram the on-chip flash.

Figure 3 is a block diagram of the on-chip flash.



Legend:
FPMON: Flash pin monitor register
FMODR: Flash mode register
FASTAT: Flash access status register
FAEINT: Flash access error interrupt enable register
ROMMAT: ROM MAT select register
FCURAME: FCU RAM enable register
FSTATR0, FSTATR1: Flash status registers 0 and 1
FENTRYR: Flash P/E mode entry register
FPROTR: Flash protect register
FRESETR: Flash reset register
FCMDR: FCU command register
FCPSR: FCU processing switch register
FPESTAT: Flash P/E status register
PCKAR: Peripheral clock notification register
FIFE: Flash interface error interrupt

Figure 3 Block Diagram of On-Chip Flash

2.2 On-Chip Flash Programming/Erasing Operation

The SH7216 uses the FCU to program and erase the on-chip flash. In the sample application, a standard API is used to program and erase the on-chip flash. The operations described below are carried out within the standard API. For details of the standard API, see the related application note.

2.2.1 Preparation for On-Chip Flash Programming/Erasing

In order to use the FCU, the FCU firmware must be stored in the FCU RAM. After the FCU firmware has been transferred, the FCU can be used to program or erase the on-chip flash by issuing FCU commands to it.

The FCU firmware is stored in the FCU firmware area of the device, and it must be transferred to the FCU RAM at startup. In addition, FCU RAM access is disabled at device startup, so access must be enabled by making the appropriate register setting.

2.2.2 On-Chip Flash Erasing

On the SH7216, the on-chip flash is divided into multiple blocks, and erasing is performed in block units. After the FCU firmware has been transferred, the FCU performs a block erase when an erase command*[1] and an execute command are written to the address of the erasure target block.

Figure 4 shows the division of the SH7216 erasure blocks, and table 2 lists the addresses of the individual blocks.

Note: 1. The erase command may be written to any valid program/erase address in the on-chip flash.

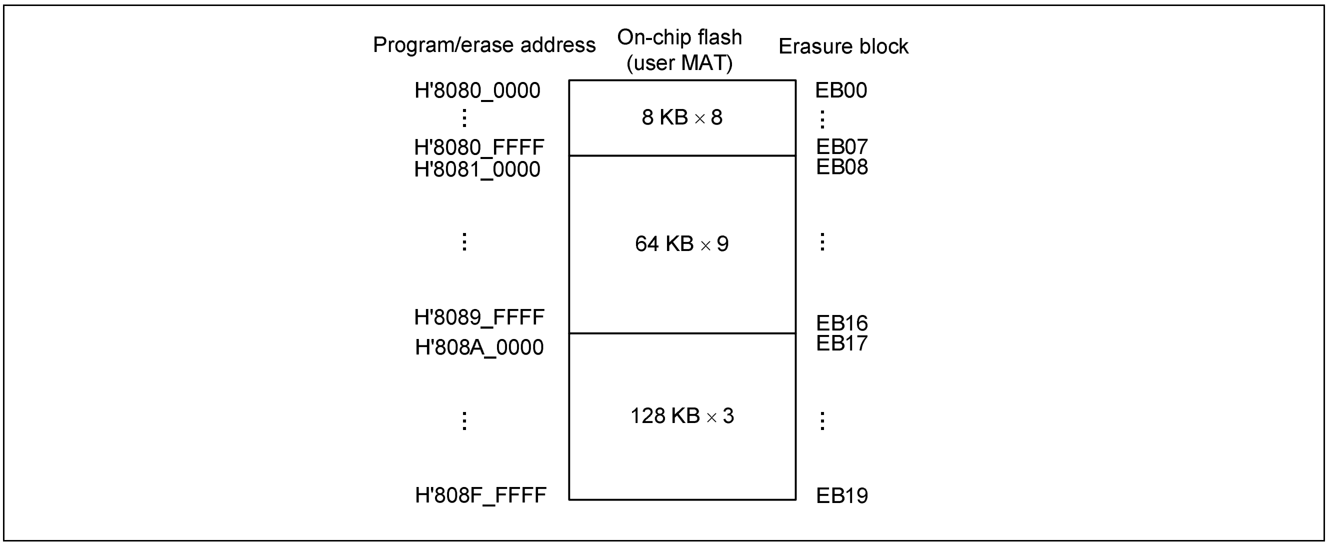


Figure 4 Division of On-Chip Flash Erasure Blocks

Table 2 Erasure Blocks and Addresses

Erasure block	Actual address	Program/erase address	Unit capacity
EB00	H'0000_0000 to H'0000_1FFF	H'8080_0000 to H'8080_1FFF	8 KB
EB01	H'0000_2000 to H'0000_3FFF	H'8080_2000 to H'8080_3FFF	
EB02	H'0000_4000 to H'0000_5FFF	H'8080_4000 to H'8080_5FFF	
EB03	H'0000_6000 to H'0000_7FFF	H'8080_6000 to H'8080_7FFF	
EB04	H'0000_8000 to H'0000_9FFF	H'8080_8000 to H'8080_9FFF	
EB05	H'0000_A000 to H'0000_BFFF	H'8080_A000 to H'8080_BFFF	
EB06	H'0000_C000 to H'0000_DFFF	H'8080_C000 to H'8080_DFFF	
EB07	H'0000_E000 to H'0000_FFFF	H'8080_E000 to H'8080_FFFF	
EB08	H'0001_0000 to H'0001_FFFF	H'8081_0000 to H'8081_FFFF	64 KB
EB09	H'0002_0000 to H'0002_FFFF	H'8082_0000 to H'8082_FFFF	
EB10	H'0003_0000 to H'0003_FFFF	H'8083_0000 to H'8083_FFFF	
EB11	H'0004_0000 to H'0004_FFFF	H'8084_0000 to H'8084_FFFF	
EB12	H'0005_0000 to H'0005_FFFF	H'8085_0000 to H'8085_FFFF	
EB13	H'0006_0000 to H'0006_FFFF	H'8086_0000 to H'8086_FFFF	
EB14	H'0007_0000 to H'0007_FFFF	H'8087_0000 to H'8087_FFFF	
EB15	H'0008_0000 to H'0008_FFFF	H'8088_0000 to H'8088_FFFF	
EB16	H'0009_0000 to H'0009_FFFF	H'8089_0000 to H'8089_FFFF	
EB17	H'000A_0000 to H'000B_FFFF	H'808A_0000 to H'808B_FFFF	128 KB
EB18	H'000C_0000 to H'000D_FFFF	H'808C_0000 to H'808D_FFFF	
EB19	H'000E_0000 to H'000F_FFFF	H'808E_0000 to H'808F_FFFF	

2.2.3 On-Chip Flash Programming

Programming of the on-chip flash can only take place when the target area is in the erased state. A single write to the user MAT comprises a 256-byte unit of data. As with erasing, the programming procedure consists of issuing a command to the FCU, after which the FCU performs the operation. A write command and the write size*[1] are issued to the program/erase address, followed by writing*[2] the write data (256 bytes) to the write destination address.*[3]

Notes: 1. The write size is fixed at 256 bytes when writing to the user MAT and user boot MAT (issue H'80 as the size).
2. The write data is written to the program/erase address in word size.
3. This address (the program/erase address) is the write address plus H'8080_0000.

2.3 Data Buffer for Reprogramming On-Chip Flash

In the sample application, a buffer area in the SH7216 is used to save the write data to be programmed to the on-chip RAM. The capacity of the buffer area is 256 bytes, which corresponds to the size of one on-chip flash write operation. A double-buffer configuration is used to enable data transfers from the external device and writing to the on-chip flash to take place in parallel.

The operation of the buffers is determined by using the buff0_full and buff1_full flags. When the value of buff0_full is BUF_ON, the data in buffer 0 (Buff0) is written to the on-chip flash while simultaneously the next unit of write data is downloaded to buffer 1 (Buff1). When the value of buff1_full is BUF_ON, the operations are reversed.

Figure 5 shows an outline of buffer operation.

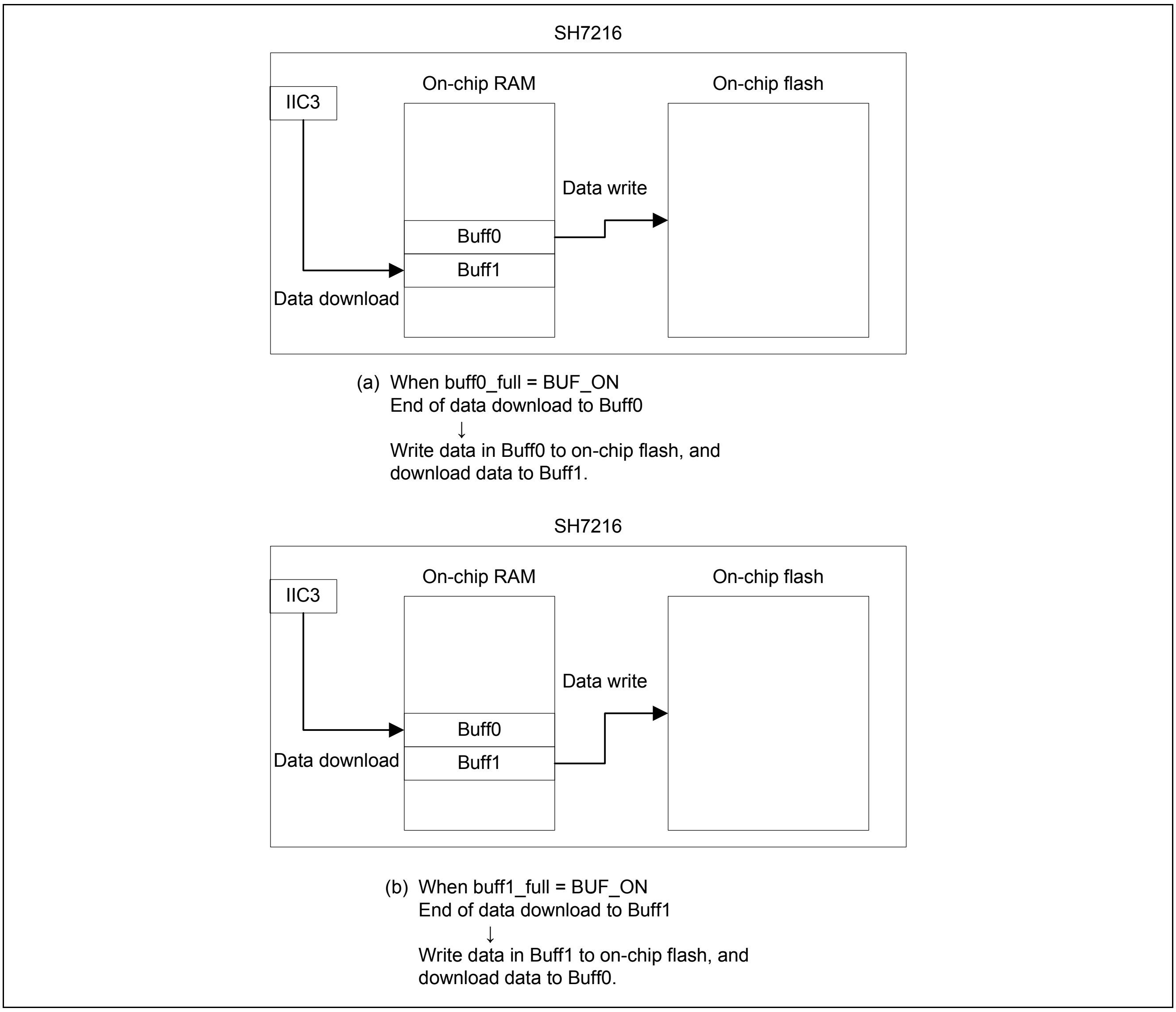


Figure 5 Outline of Buffer Operation

3. Operation of Sample Program

3.1 Outline of Operation Overall

The sample application erases and programs the user MAT area except for block EB00 (EB01 to EB19).

Figure 6 shows the overall operation sequence.

The above-mentioned area in the on-chip flash programming target device (SH7216) is erased, and a start condition is issued. After the start condition is issued, the SH7216 receives data from the external device until a write end command is received.

After all the data has been written, the SH7216 issues the end command and processing ends.

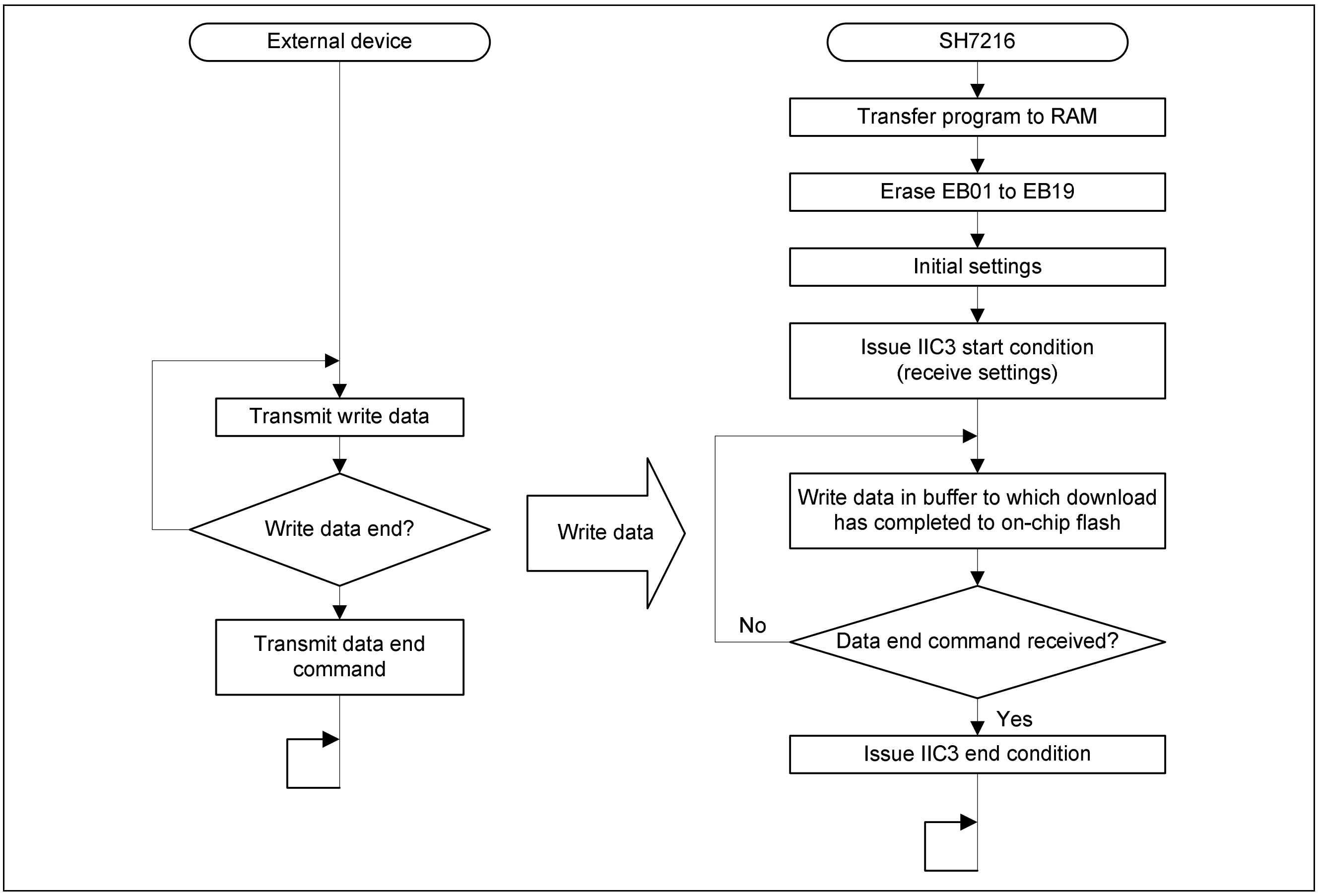


Figure 6 Overall Operation Sequence

3.2 Basic Specifications of Sample Program

3.2.1 Functions Used by Sample Program

Table 3 lists the control flags used in the sample application.

Table 3 Control Flags

Variable	Function	Remarks
buff_no	Indicates buffer number (0/1) used for download.	
write_area	Write start address in on-chip flash	
on_write0	Indicates the on-chip flash write state of buffer 0. 0 (BUFF_OFF): No data write in progress 1 (BUFF_ON): Data write in progress	
on_write1	Indicates the on-chip flash write state of buffer 1. 0 (BUFF_OFF): No data write in progress 1 (BUFF_ON): Data write in progress	
buff0_full	Indicates the download state of buffer 0. 0 (BUFF_OFF): Buffer empty 1 (BUFF_ON): Buffer data full	
buff1_full	Indicates the download state of buffer 1. 0 (BUFF_OFF): Buffer empty 1 (BUFF_ON): Buffer data full	
dl_number	Data download count	

3.2.2 Control Functions of Sample Program

Table 4 lists the functions used in the sample application.

Table 4 List of Functions Used

Function Name	Function	Remarks
main	Erases on-chip flash and makes initial settings.	Figure 7
ROM_WE_MAIN	Processes writing to on-chip flash (writing of buffer data using control flags).	Figure 8
INTR_IIC3	IIC3 receive interrupt (data reception from external device)	Figure 12
i2c_init	Initializes IIC3.	Figure 9
i2c_recv	Issues I^2C bus start condition.	Figure 10
i2c_end	Issues I^2C bus end condition.	Figure 11
R_FlashErase	Erases specified block.	Standard API
R_FlashWrite	Writes data to specified address.	

3.2.3 Section Settings of Sample Program

Table 5 lists the section settings used in the sample application.

Table 5 Sample Program Section Settings

Section Name	Address	Description	Remarks
DVECTTBL	H'0000_0000	Vector table	Allocated in on-chip flash.
DINTTBL			
PResetPRG		Reset handler	
PIntPRG		Exception handler	
P	H'0000_1000	Program area	
PFRAM		Standard API	
C$BSEC		Constant area	
C$DSEC			
D		Initialization data area	
RDVECTTBL	H'FFF8_0000	Vector table (allocated in RAM)	Allocated in on-chip RAM.
RDINTTBL			
RPResetPRG		Reset handler (allocated in RAM)	
RPIntPRG		Exception handler (allocated in RAM)	
RP	H'FFF8_1000	Program area (allocated in RAM)	
RPFRAM		Standard API (allocated in RAM)	
RC$BSEC		Constant area (allocated in RAM)	
RC$DSEC			
B	H'FFF8_4000	Uninitialized data area	
R		Initialization data area (allocated in RAM)	
S	H'FFF9_FC00	Stack area	

3.2.4 Register Settings of Sample Program

Table 6 lists the register settings used in the sample application.

Table 6 Register Settings

Module	Register Name	Address	Setting Value	Description
Pin function controller (PFC)	Port B control register L (PBCRL4)	H'FFFE3890	H'0006	PB12MD[2:0] = "B'110": SCL I/O PB13MD[2:0] = "B'110": SDA I/O
	Port B pull-up MOS control register L (PBCRL2)	H'FFFE38AA	H'3000	PB12PCR = "B'1" PB13PCR = "B'1" Input pull-up MOS on
I^2C bus interface 3 (IIC3)	ICCR1 bus control register 1 (ICCR1)	H'FFFEE000	H'00 H'01 H'00 H'02 H'00 H'01 H'10 H'11	ICE = B'0: Function halted ICE = B'1: Enabled for transfer operations RCVD = B'0: Next receive operation enabled CKS[4:0] = B'10: Pϕ/84 (595 kHz) MST = Master/slave select TRS = Transmit/receive select 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode
	ICCR2 bus control register 2 (ICCR2)	H'FFFEE001	H'01 H'00	BBSY = B'1: Bus busy SCP = Start/stop issue condition disable
	ICMR bus mode register (ICMR)	H'FFFEE002	H'30	MLS = B'0: MSB first BC[2:0] = B'000: 9 bits
	ICIER bus interrupt enable register (ICIER)	H'FFFEE003	H'02 H'00	ACKBR = B'1: Receive acknowledge ACKBT = B'1: 0 sent at acknowledge timing
	ICSR bus status register (ICSR)	H'FFFEE004	H'00	TDRE = B'0: Transmit data register empty TEND = B'0: Transmit end STOP = B'0: Stop condition detection flag
	ICDRT bus transmit data register (ICDRT)	H'FFFEE006	—	8-bit readable/writable register that stores transmit data
	ICDRR bus receive data register (ICDRR)	H'FFFEE007	—	8-bit register that stores receive data

3.3 Flowcharts

Figures 7 to 12 show the operation sequences of the functions used in the sample application.

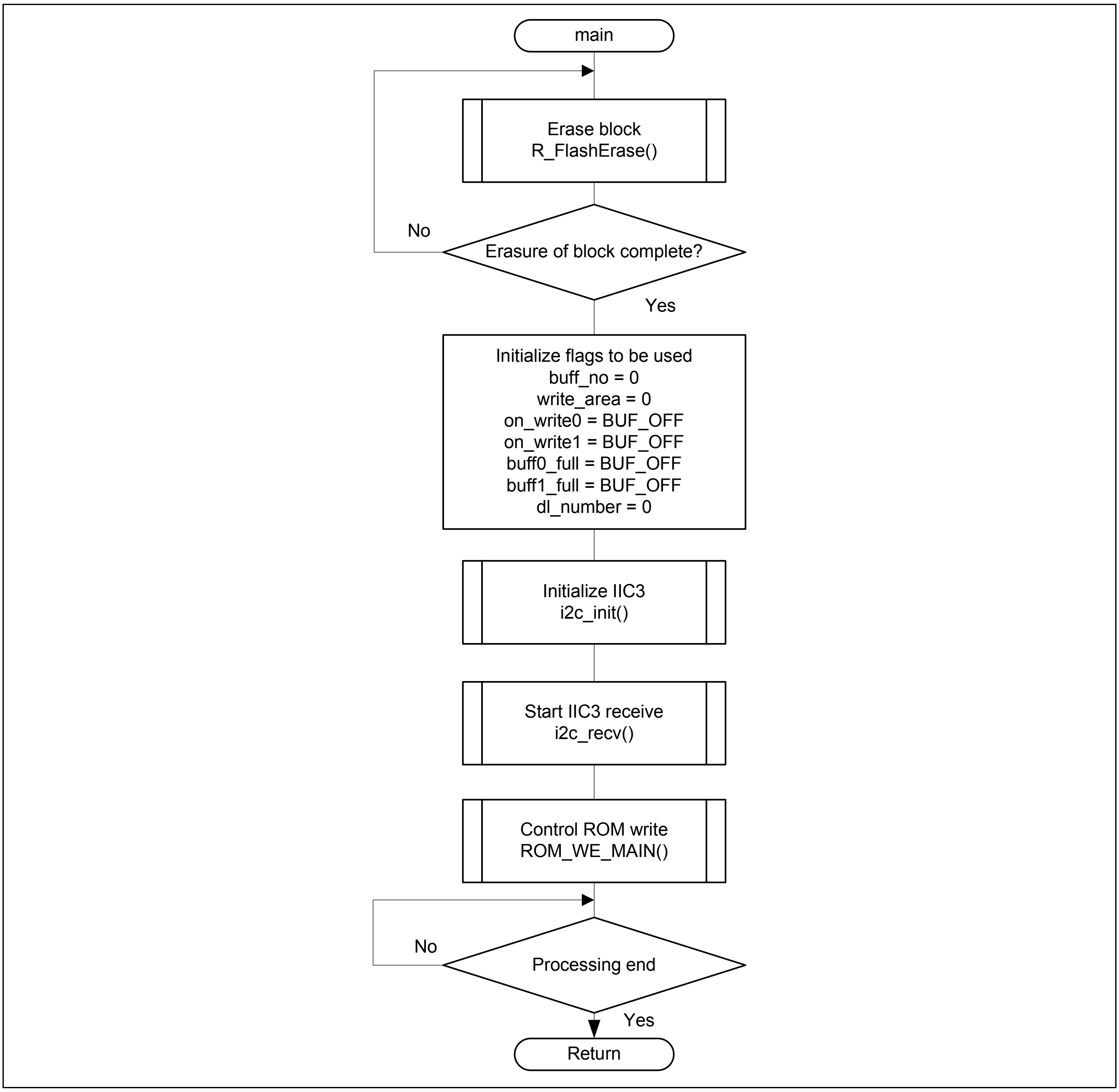


Figure 7 Main Process Sequence

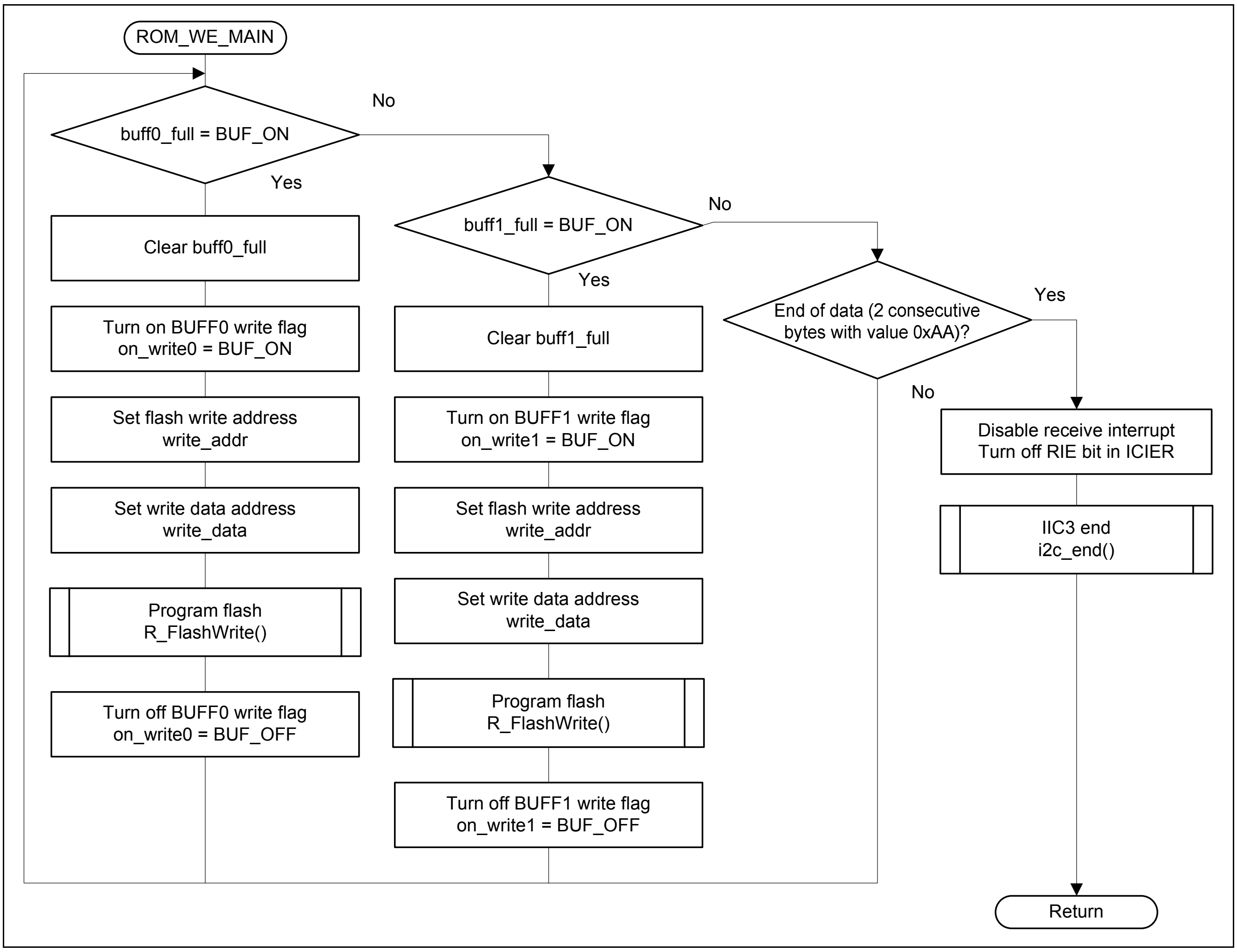


Figure 8 On-Chip Flash Programming Sequence

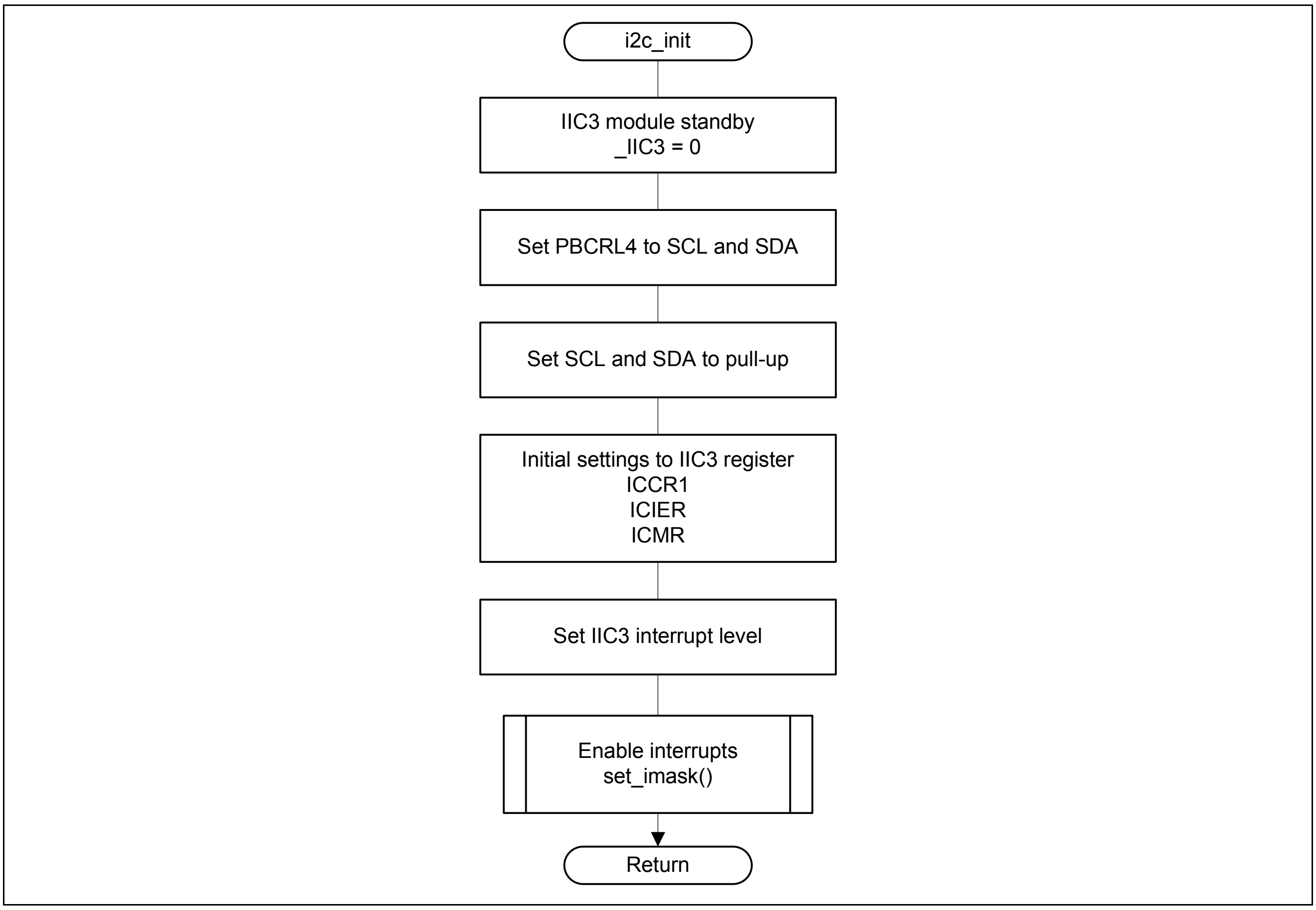


Figure 9 IIC3 Initialization Sequence

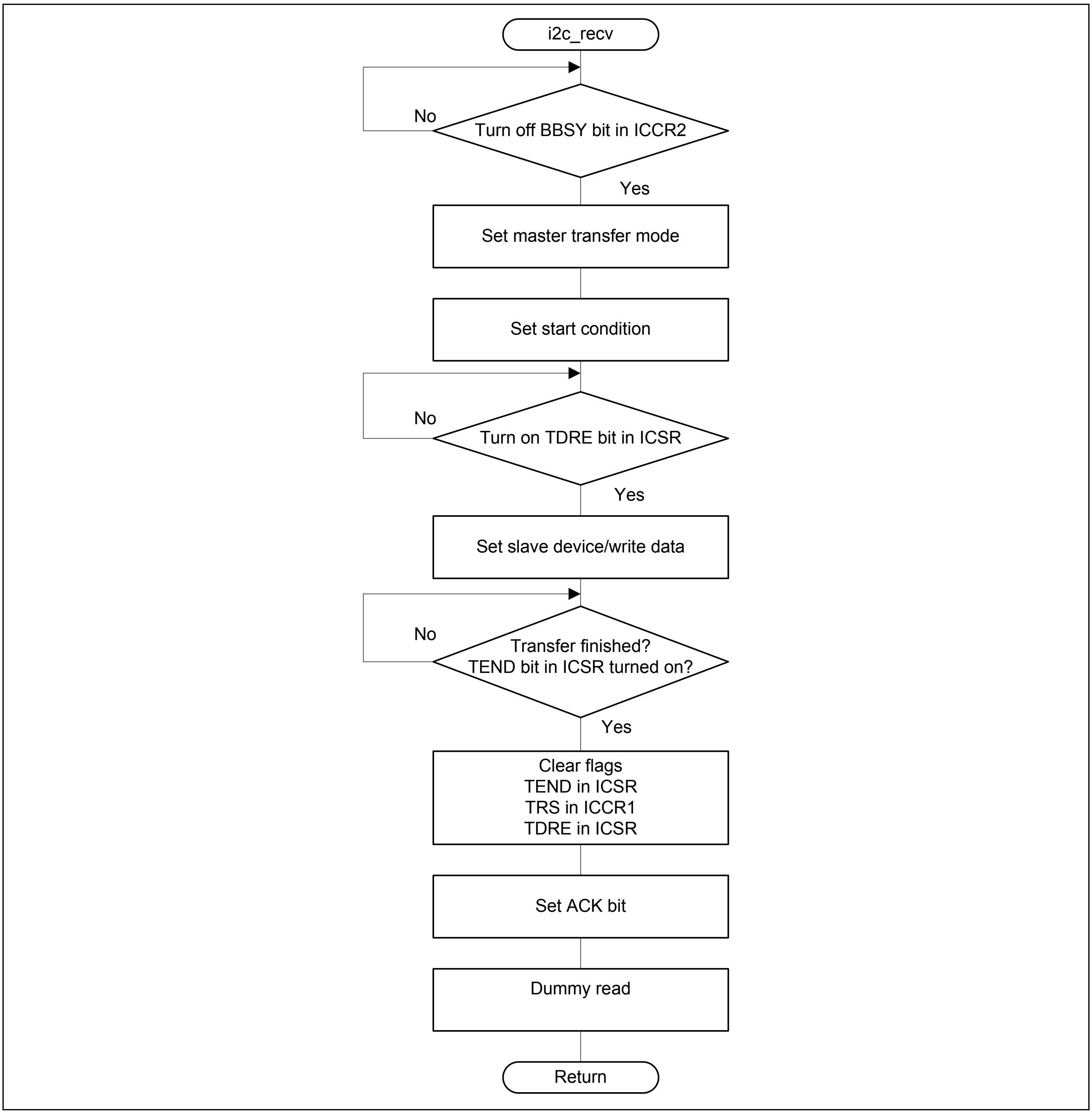


Figure 10 Start Condition Issue Sequence (Receive)

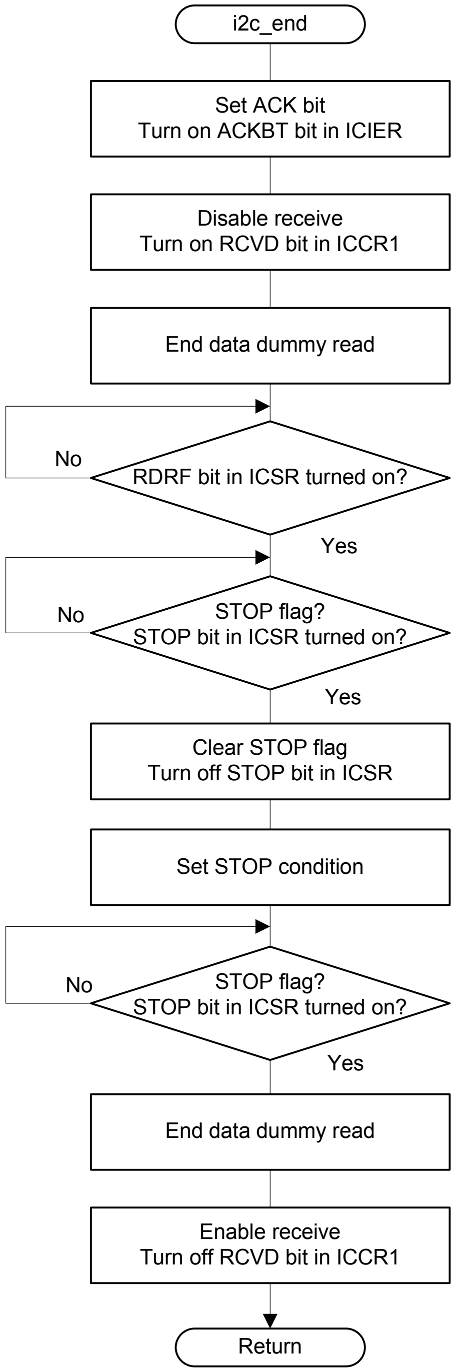


Figure 11 End Condition Issue Sequence

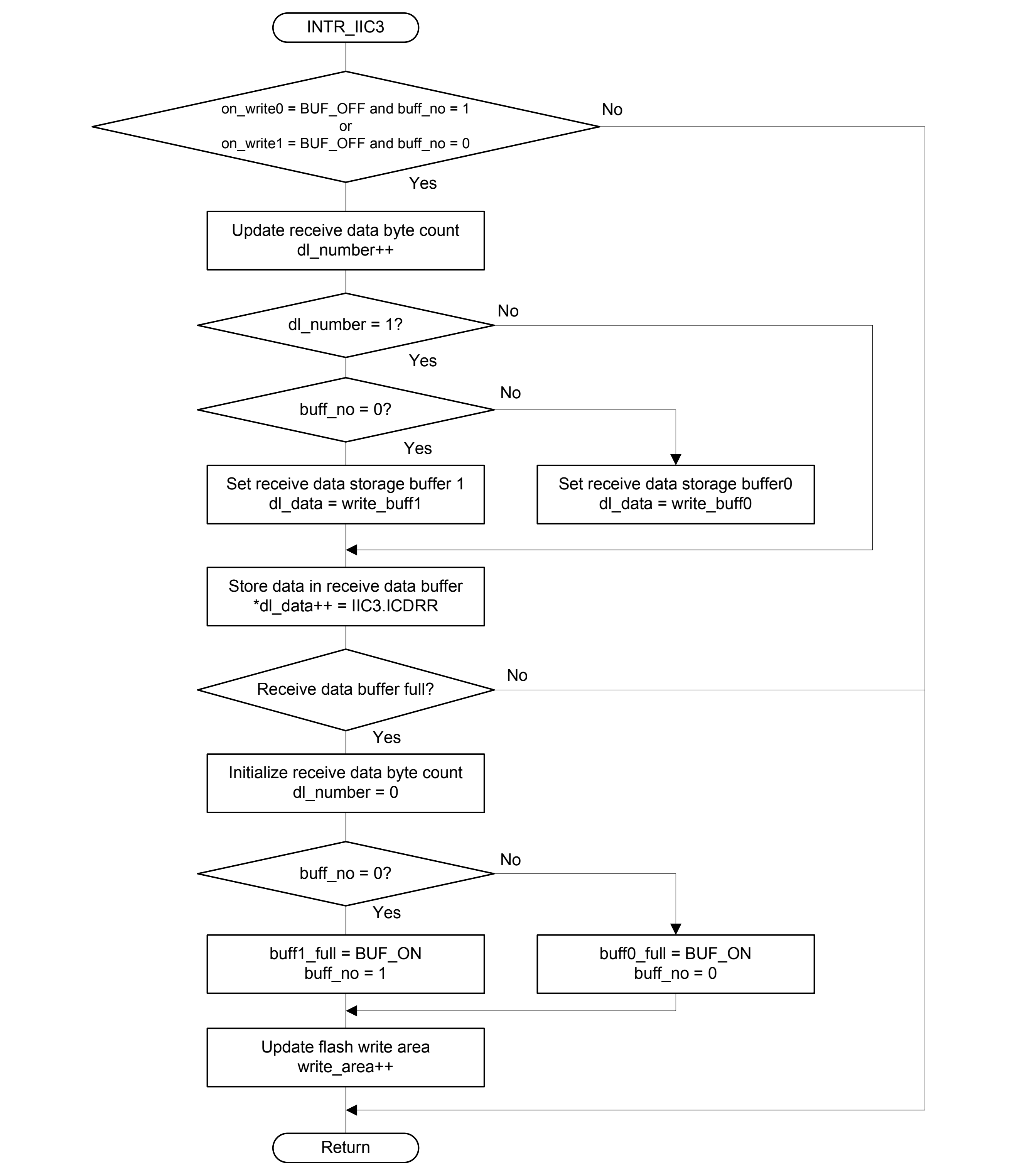


Figure 12 Receive Interrupt Sequence

Website and Support

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

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

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Jun.20.10	—	First edition issued
2.00	Dec.14.10	—	Amended to use simple flash API for SH-2 and SH-2A Revised overall
2.10	Feb.28.11	—	Added read after FRQCR settings
2.20	May.18.11	—	Source project revised

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 manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

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 one with a different type number, confirm that the change will not lead to problems.
   — The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.

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