M16C/63, 64, 64A, 64C, 65, 65C, 6C, 5LD, 56D, 5L, 56, 5M, 57 Group

I2C-bus Interface Using UARTi Special Mode 1 (Master Transmit/Receive)

REJ05B1422-0101
Rev.1.01
Jan 31, 2011

# 1. Abstract

This document describes the master transmit/receive processes in I2C-bus interface single master communication using the M16C/63, 64, 64A, 64C, 65, 65C, 6C, 5LD, 56D, 5L, 56, 5M and 57 Groups serial interface (UART2) special mode 1 (I2C mode).
For details on UART2 special mode 1, refer to the **M16C Family I2C-bus Interface Using UARTi Special Mode 1** application note.
If this application note is applied for using channels except UART2, see the user's manual, and modify registers related to UARTi.

# 2. Introduction

The application example described in this document applies to the following microcomputer (MCU) and parameter:

- MCUs: M16C/65 Group and M16C/65C Group
- XIN Clock: 20 MHz

This application note can be used with other M16C Family MCUs which have the same special function registers (SFRs) as the above group. Check the user’s manual for any modifications to functions. Careful evaluation is recommended before using the program described in this application note.

# 3. Application Example

## 3.1 Program Outline

Transmission is performed in 3-byte data both in master transmission and reception. Master transmission and reception are repeated alternately. The transmission and reception procedures conform to the I²C-bus communication protocol when used under the following conditions:

- Slave address: 7 bits
- Transfer rate: Approximately 350 kbps (1)
- Transfer data length: 1 to 255 bytes (not including the slave address)
- Single master communication (multi-master is not supported)
- Restart condition generation is not supported

Note:

1. The setting value is 384 kbps.
   When the clock synchronous function is enabled, there is a sampling delay of the noise filter width plus 1 to 1.5 cycles of the U2BRG count source. As there is also a delay of the SCL clock when high is determined, the SCL clock high width is extended. Therefore, the actual SCL clock becomes slower than SCL clock transfer rate setting.
   In this application example, the actual transfer rate becomes approximately 350 kbps since the clock synchronous function is enabled (reference value: pull up voltage 5 V, pull up resistance 1 kΩ).
   Standard-mode and Fast-mode are supported.

Figure 3.1 shows the Communication Format, Figure 3.2 shows the Block Diagram, Figure 3.3 shows the Outline Flowchart, and Figure 3.4 to Figure 3.6 show Timing Diagrams.

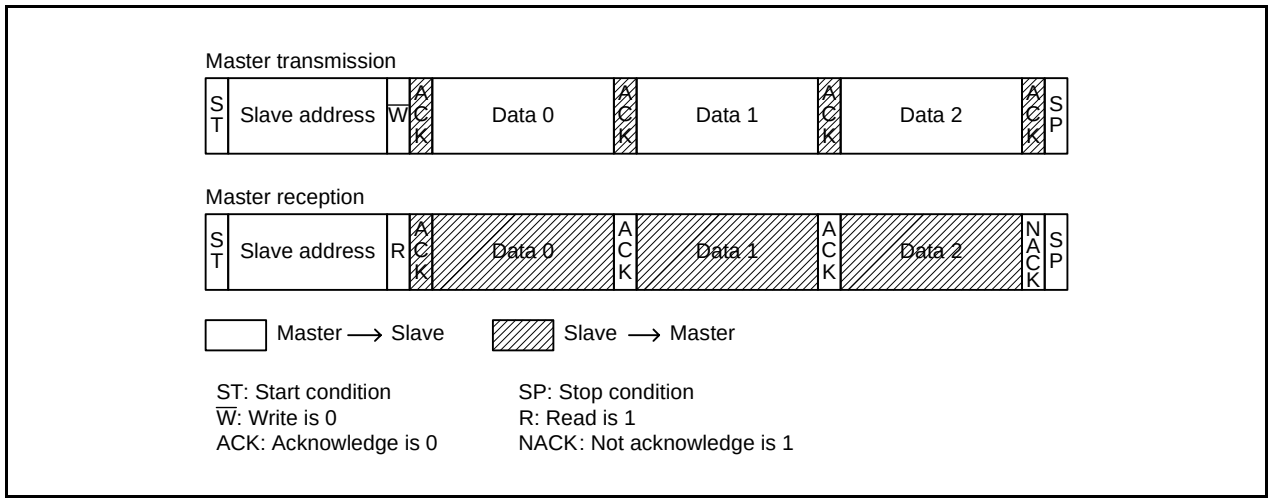



**Figure 3.1 Communication Format**

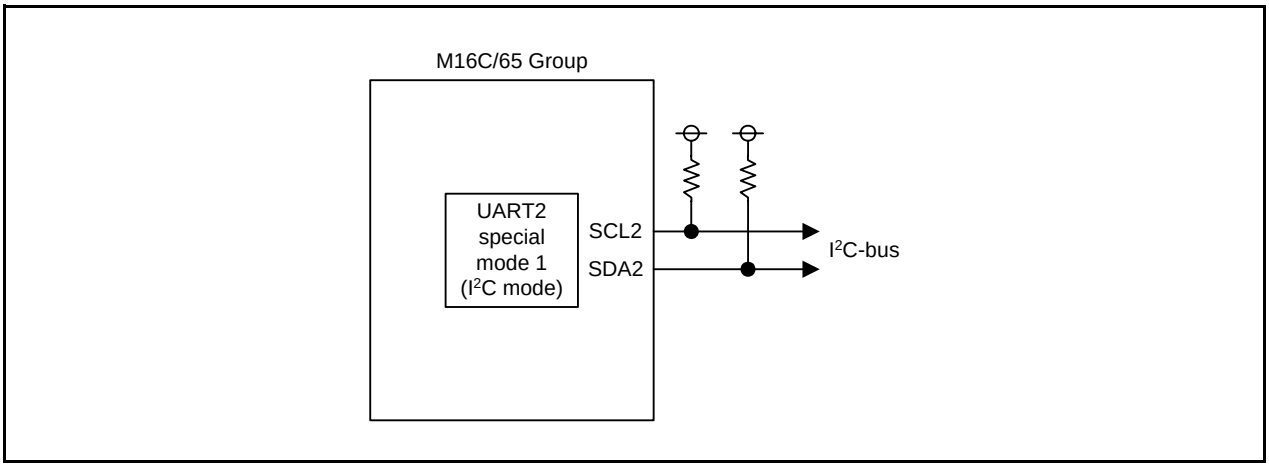



**Figure 3.2 Block Diagram**

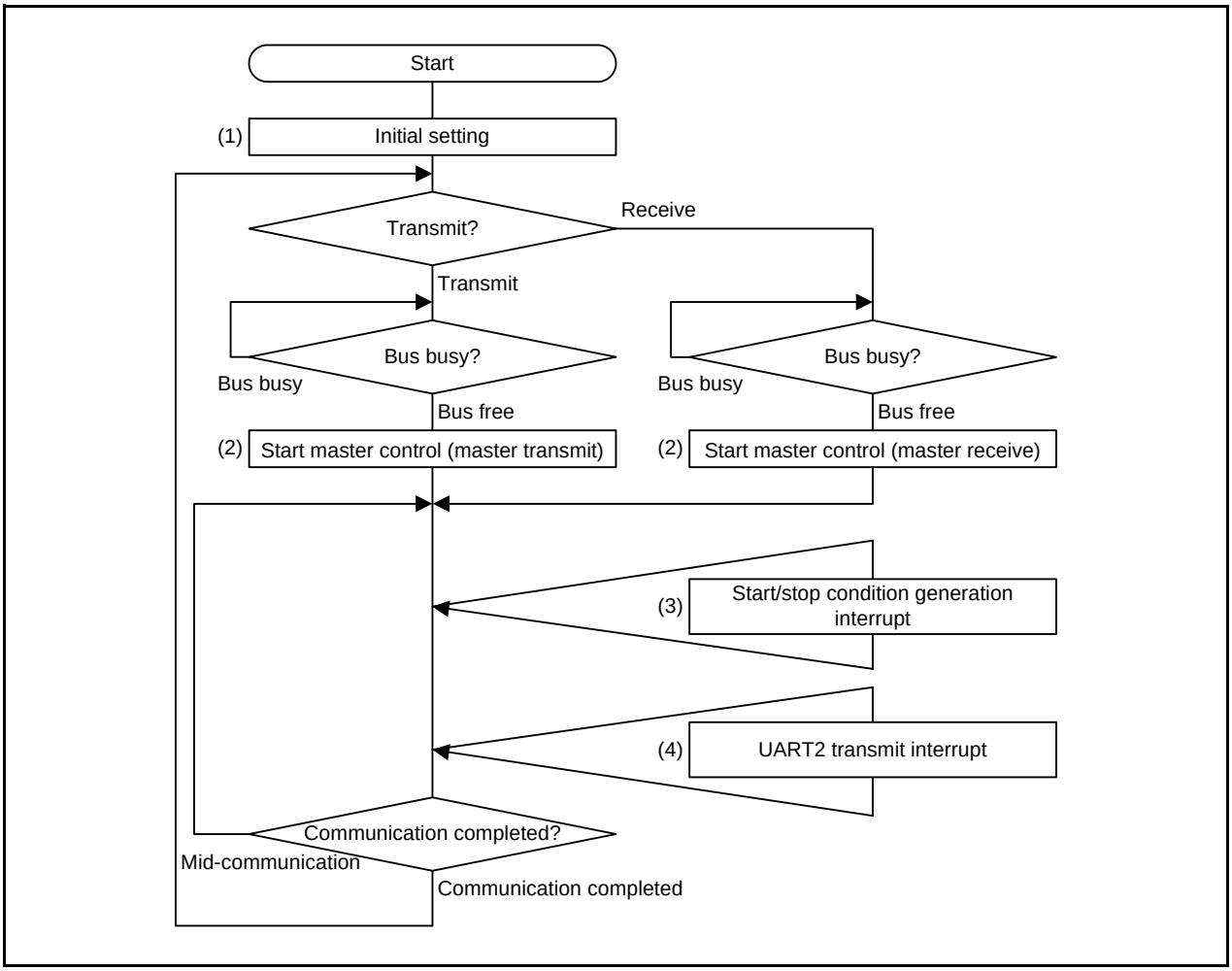



**Figure 3.3 Outline Flowchart**

The numbers in Figure 3.3 correspond to the numbers indicated in the program processing in the operating timing charts in Figure 3.4 to Figure 3.6.

(1) Initial setting
Initialize the system clock, UART2 associated SFRs, and variables used.

(2) Start master control
Enable the start/stop condition generation interrupt and generate a start condition.

(3) Start/stop condition generation interrupt
An interrupt request is generated when start condition generation is completed and a stop condition is detected. When start condition generation is completed, the UART2 transmit interrupt is enabled and the slave address is transmitted. When a stop condition is detected, SFR values which changed during communication are returned to their initial values.

(4) UART2 transmit interrupt
A UART2 transmit interrupt is generated at the falling edge of the ninth bit of the SCL clock. When transmitting, set the next byte transmit data. When receiving, set ACK/NACK for the next byte. When communication is completed, generate a stop condition.

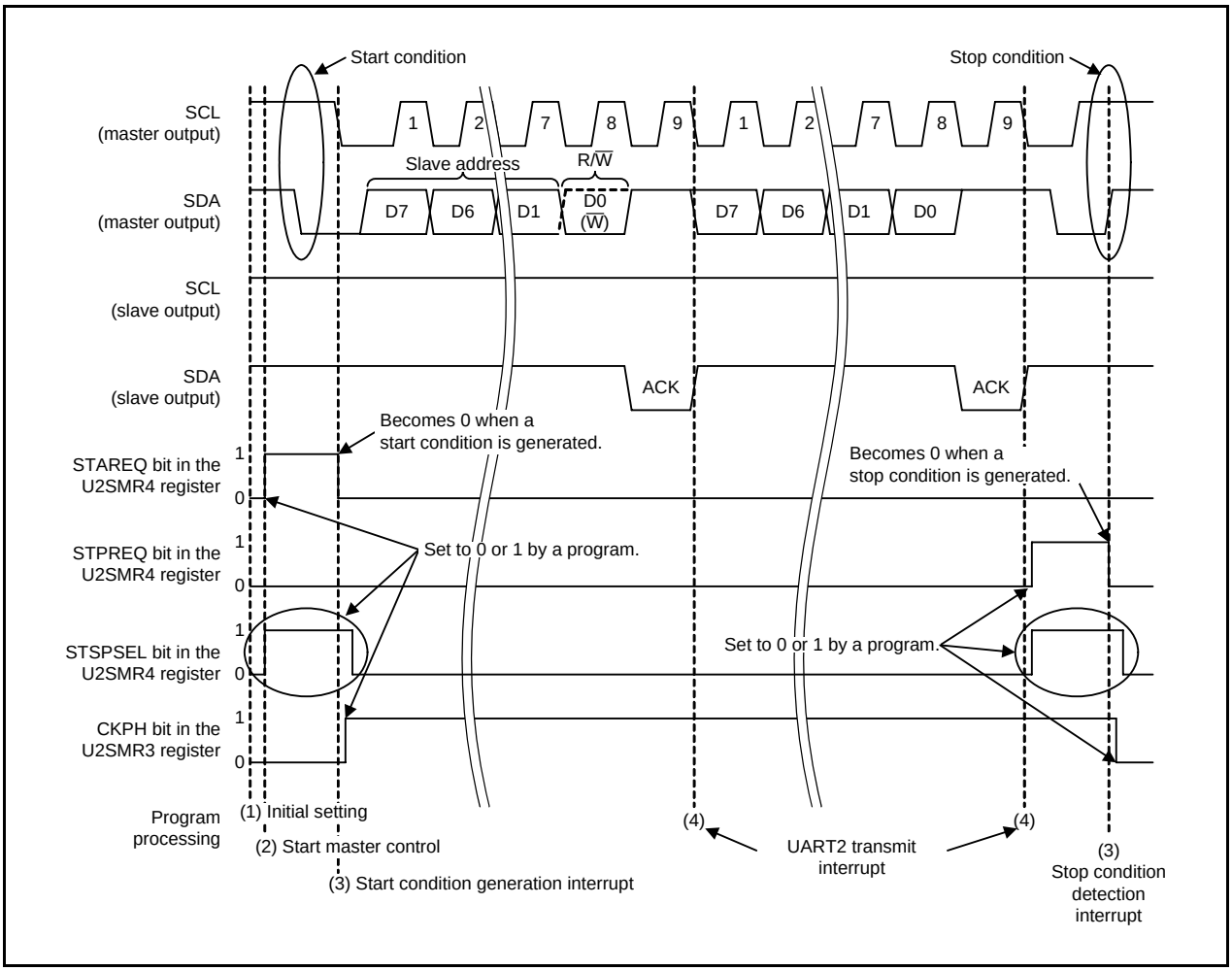



**Figure 3.4 Master Transmit Timing**

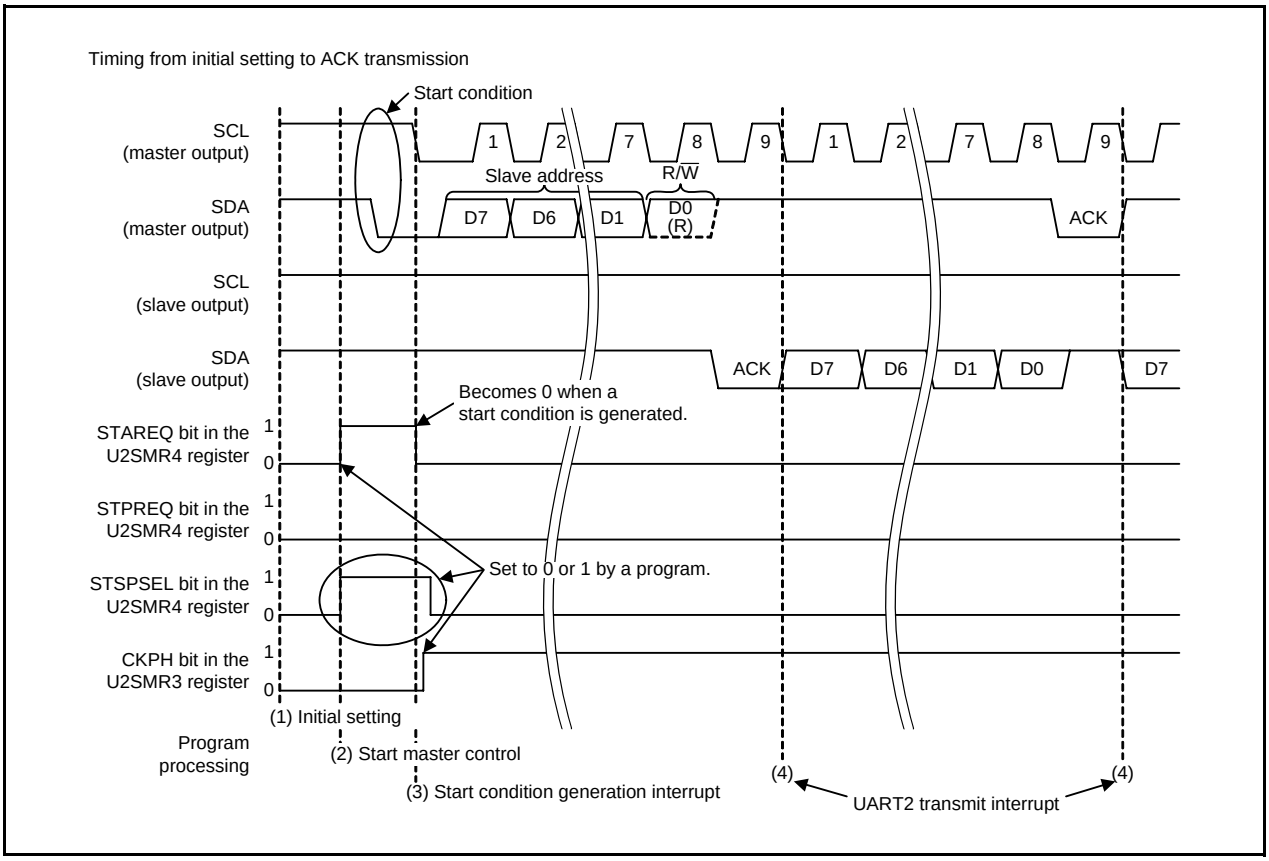



**Figure 3.5 Master Receive Timing (1)**

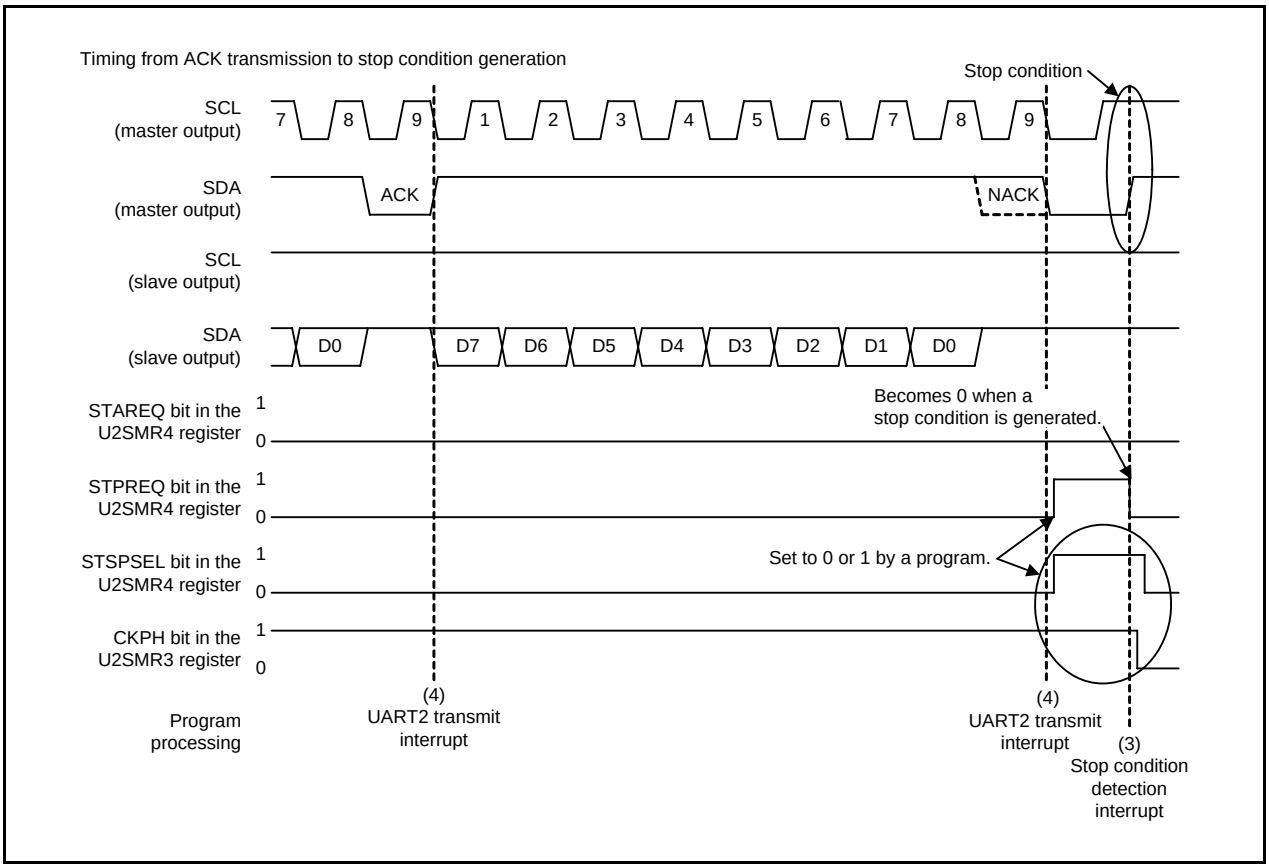



**Figure 3.6 Master Receive Timing (2)**

### 3.1.1 Peripheral Functions

Serial interface (UART2) special mode 1 ($I^2C$ mode) is used under the following setting conditions:

- $I^2C$ mode is used.
- Transfer clock is internal clock source.
- f1 is used as U2BRG count source.
- SDA2 and SCL2 pins are N-channel open drain.
- Transfer format uses MSB first.
- Transmission completed (TXEPT is 1) is selected as the UART2 transmit interrupt source.
- Clock delay is used.
- Seven to eight cycles of U2BRG count source is selected as SDA2 digital delay value.
- Clock synchronization is enabled.
- SCL2 wait function is disabled.
- SDA2 output disable function is not used.
- Start/stop condition generation interrupt is used.
- UART2 transmit interrupt is used.
- UART2 receive interrupt is not used.
- Transfer rate is approximately 384 kbps.

Calculating the transfer rate:
Transfer rate = U2BRG count source / (2 × (U2BRG register setting value + 1))
= 20 MHz (f1) / (2 × (25 + 1))
≈ 384.615 kbps

**Table 3.1 Pins Used and Their Function**

| Pin | I/O | Function |
|---|---|---|
| P7_1/SCL2 | I/O | $I^2C$ mode clock I/O pin |
| P7_0/SDA2 | I/O | $I^2C$ mode data I/O pin |

### 3.1.2 Notes on Using the Attached Sample Program

Note the following when using the program included with this application note:

- Do not use multiple interrupts.
- When setting the system clock to anything other than the XIN clock (20 MHz), change the setting value of the U2BRG count source and the U2BRG register according to the transfer rate calculation shown in **3.1.1 Peripheral Functions**.
- If pins SDAi and SCLi are not N-channel open drain, set the NCH bit in the UiC0 register to 1 (pins TXDi/SDAi and SCLi are N-channel open drain output).
- UART0 and UART1 transmit interrupt source select bits exist in the UCON register in M16C/60 Series. Set the U0IRS bit or U1IRS bit in the UCON register to 1 (transmission completed (TXEPT = 1)).

## 3.2 Memory

**Table 3.2 Memory**

| Memory | Size | Remarks |
|---|---|---|
| ROM | 572 bytes | In the iic.c module |
| RAM | 8 bytes | In the iic.c module |
| Maximum user stack | 19 bytes | |
| Maximum interrupt stack | 30 bytes | |

Usage memory size varies depending on C compiler version and compile options. The above applies under the following conditions:
C compiler: M16C/60, 30, 20, 10, Tiny, R8C/Tiny Series Compiler V.5.45 Release 01
Compile option: -c -finfo -dir “$(CONFIGDIR)”

# 4. Software

This chapter shows the program example to set the example described in chapter 3. Application Example. Refer to the latest hardware user's manual for details on individual registers.

## 4.1 Variables

| Variable Name | Size | Description |
|---|---|---|
| unsigned char iic_tx[BUFSIZE] | 255 bytes | Transmit buffer |
| unsigned char iic_rx[BUFSIZE] | 255 bytes | Receive buffer |
| unsigned char retry_counter | 1 byte | Count number of communication retries |

| Variable Name | | Size /Bit Number | Description |
|---|---|---|---|
| static byte_dt iic_str1 | | - | Structure to store slave address |
| Structure member | iic_slave_addr | 1 byte | Slave address |
| | iic_rw | b0 | R/$\overline{W}$ flag<br>0: Write ($\overline{W}$)<br>1: Read (R) |
| | - | b7 to b1 | 7-bit address |
| static byte_dt iic_str2 | | - | Structure to store status |
| Structure member | iic_status | 1 byte | All statuses |
| | iic_start | b0 | Mid-communication flag<br>0: Communication completed<br>1: Mid-communication |
| | iic_err_par | b1 | Parameter error flag<br>0: No error<br>1: Parameter error |
| | iic_err_nack | b2 | NACK detection error flag<br>0: No error<br>1: NACK detection error |
| | iic_err_addr | b3 | No address match error flag<br>0: No error<br>1: No address match error |
| | - | b7 to b4 | Not used (undefined) |
| unsigned char iic_length | | 1 byte | Transfer data length |
| unsigned char iic_index | | 1 byte | Number of transmit/receive bytes |
| unsigned char far *iic_pointer | | 4 bytes | Transmit/receive buffer pointer |

## 4.2 Function Tables

| | | | |
|---|---|---|---|
| Declaration | void main (void) | | |
| Outline | Main processing | | |
| Argument | Argument name | | Meaning |
| | None | | - |
| Variable (global) | Variable name | | Contents |
| | unsigned char iic_tx[BUFSIZE] | | Transmit buffer |
| | unsigned char iic_rx[BUFSIZE] | | Receive buffer |
| | unsigned char retry_counter | | Count number of communication retries |
| Returned value | Type | Value | Meaning |
| | None | - | - |
| Function | After initializing the system clock and UART2, master transmission and reception are repeated alternately. After calling the iic_master_start function to start master control, call the iic_master_end function and wait for master control to be completed.<br>When the iic_master_end function returns ADD_ERR (communication stop because of address not matched error), communication is retried. | | |

| | | | |
|---|---|---|---|
| Declaration | void mcu_init (void) | | |
| Outline | System clock setting | | |
| Argument | Argument name | | Meaning |
| | None | | - |
| Variable (global) | Variable name | | Contents |
| | None | | - |
| Returned value | Type | Value | Meaning |
| | None | - | - |
| Function | Call this function from the main processing. Set the system clock (XIN clock). | | |

| | | | |
|---|---|---|---|
| Declaration | void uart2_init (unsigned char ini) | | |
| Outline | UART2 initial setting | | |
| Argument | Argument name | Meaning | |
| | unsigned char ini | 0: I²C mode disabled<br>1: I²C mode enabled | |
| Variable (global) | Variable name | | Contents |
| | None | | - |
| Returned value | Type | Value | Meaning |
| | None | - | - |
| Function | Call this function from the main processing. Initialize the SFRs used for UART2 in special mode 1 (I²C mode). | | |

| | | | |
|---|---|---|---|
| Declaration | unsigned char iic_master_start (<br>unsigned char addr,<br>unsigned char rw,<br>unsigned char far *buf,<br>unsigned char len) | | |
| Outline | Master control start processing | | |
| Argument | Argument name | Meaning | |
| | unsigned char addr | 0x00 to 0x7F: Specify slave address | |
| | unsigned char rw | 0x00: Master transmit<br>0x01: Master receive | |
| | unsigned char far *buf | Transmit or receive buffer pointer | |
| | unsigned char len | 0x01 to 0xFF: Transfer data length | |
| Variable (global) | Variable name | Contents | |
| | (structure member) iic_status | All statuses | |
| | (structure member) iic_start | Mid-communication flag | |
| | (structure member) iic_err_par | Parameter error flag | |
| | (structure member) iic_slave_addr | Slave address | |
| | unsigned char iic_length | Transfer data length | |
| | unsigned char far *iic_pointer | Transmit/receive buffer pointer | |
| Returned value | Type | Value | Meaning |
| | unsigned char | BUSY | Bus busy |
| | | RDY | Bus free |
| | | PAR_ERR | Parameter error |
| Function | This function is called by the main function to perform master control start processing. Before executing this function, execute the uart2_init function to enable I²C mode. In the function header, all statuses are initialized and argument parameters are checked. If any parameter value is invalid, the parameter error flag is set to 1 and PAR_ERR is returned. Master control start processing is not performed when a parameter error is detected. Next, the bus status is checked.<br>• When the bus is busy, the returned value is BUSY and master control start processing is not performed.<br>• When the bus is free, the returned value is RDY and master control start processing is performed. Set the mid-communication flag to 1 and a start condition is generated. | | |

| | | | |
|---|---|---|---|
| Declaration | void _start_stop_condition_detection (void) | | |
| Outline | Start/stop condition generation interrupt handling | | |
| Argument | Argument name | | Meaning |
| | None | | - |
| Variable (global) | Variable name | | Contents |
| | None | | - |
| Returned value | Type | Value | Meaning |
| | None | - | - |
| Function | An interrupt is generated when the start condition generation is completed and a stop condition is detected.<br>The sta_int function is called when the start condition generation is completed.<br>The stp_int function is called when a stop condition is detected. | | |

| Declaration | static void sta_int (void) | | |
|---|---|---|---|
| Outline | Start condition detection processing | | |
| Argument | Argument name | | Meaning |
| | None | | - |
| Variable (global) | Variable name | | Contents |
| | (structure member) iic_slave_addr | | Slave address |
| | unsigned char iic_index | | Number of transmit/receive bytes |
| Returned value | Type | Value | Meaning |
| | None | - | - |
| Function | Called from the start/stop condition generation interrupt handling.<br>UART2 transmit/receive interrupt is enabled. Transmit the slave address. | | |

| Declaration | static void stp_int (void) | | |
|---|---|---|---|
| Outline | Stop condition detection processing | | |
| Argument | Argument name | | Meaning |
| | None | | - |
| Variable (global) | Variable name | | Contents |
| | unsigned char iic_index | | Number of transmit/receive bytes |
| | (structure member) iic_start | | Mid-communication flag |
| Returned value | Type | Value | Meaning |
| | None | - | - |
| Function | Called from the start/stop condition generation interrupt handling.<br>SFR associated with UART2 values which changed during communication are returned to their values, and the mid-communication flag is set to 0. | | |

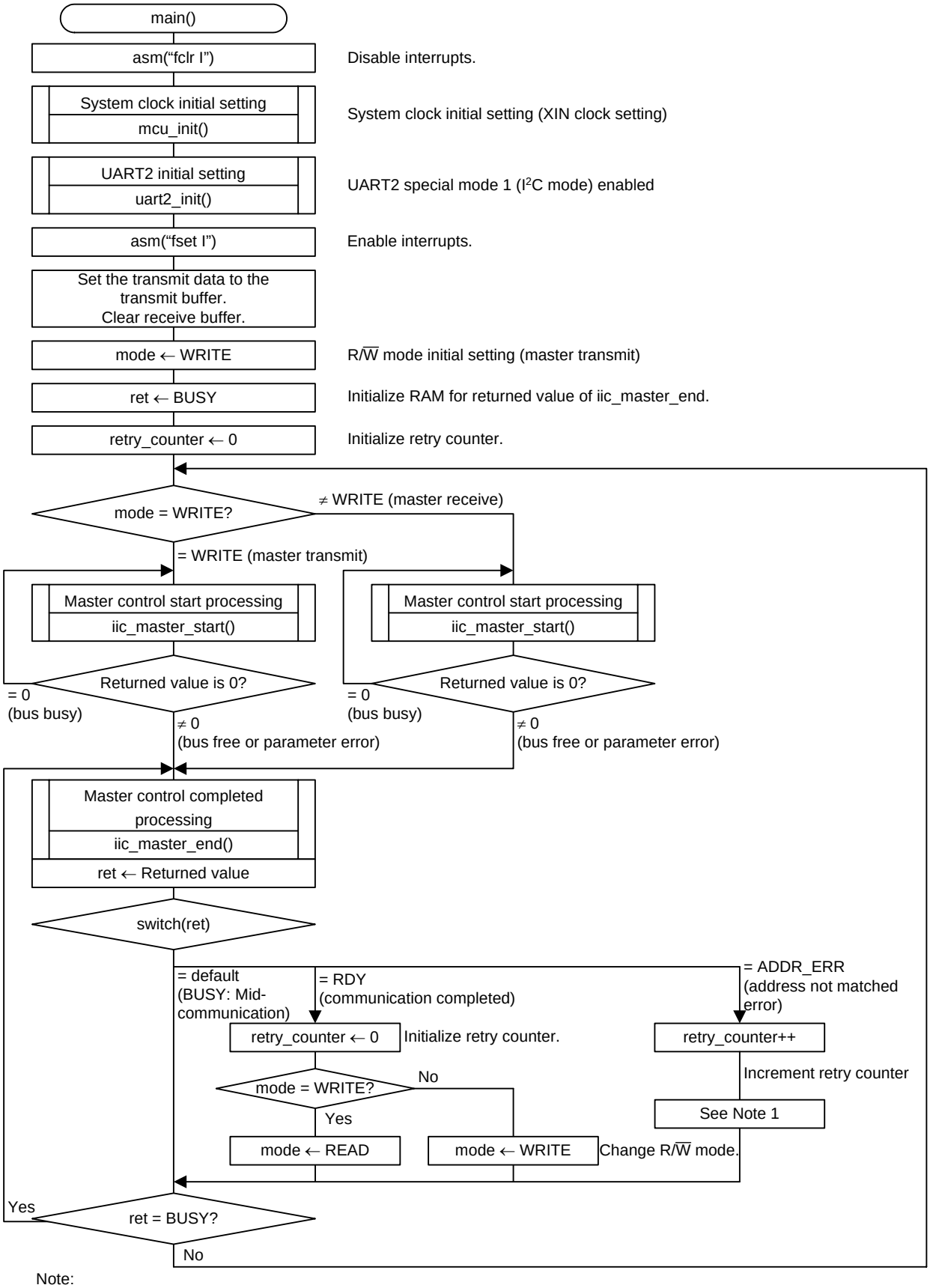
| Declaration | void _uart2_transmit (void) | | |
|---|---|---|---|
| Outline | UART2 transmit interrupt handling | | |
| Argument | Argument name | | Meaning |
| | None | | - |
| Variable (global) | Variable name | | Contents |
| | unsigned char iic_index | | Number of transmit/receive bytes |
| | (structure member) iic_err_addr | | No address match error flag |
| | (structure member) iic_rw | | R/$\overline{W}$ flag |
| Returned value | Type | Value | Meaning |
| | None | - | - |
| Function | An interrupt is generated at the falling edge of the ninth bit of the SCL clock.<br>Read the U2RB register in the function header.<br>When a NACK is detected during slave address transmission, set the no address match error flag to 1. At all other times, the master_trn_int function is called during master transmission and the master_rcv_int function is called during master reception.<br>When communication is completed, generate a stop condition. | | |

| | | | |
|---|---|---|---|
| Declaration | static unsigned char master_trn_int (unsigned short rb_data) | | |
| Outline | Master transmit processing | | |
| Argument | Argument name | | Meaning |
| | unsigned short rb_data | | Data read from the U2RB register |
| Variable (global) | Variable name | | Contents |
| | (structure member) iic_err_nack | | NACK detection error flag |
| | unsigned char iic_index | | Number of transmit/receive bytes |
| | unsigned char iic_length | | Transfer data length |
| | unsigned char far *iic_pointer | | Transmit/receive buffer pointer |
| Returned value | Type | Value | Meaning |
| | unsigned char | IIC_SP_ON | 0: Stop condition generated |
| | | IIC_SP_OFF | 1: Stop condition not generated |
| Function | Called from the UART2 transmit interrupt handling.<br>IIC_SP_OFF is returned in the following case:<br>• ACK is detected and not the last byte (starts the next transmission).<br>IIC_SP_ON is returned in the following cases:<br>• NACK is detected (NACK detect error flag is set to 1).<br>• When the last byte transmission is completed. | | |

| | | | |
|---|---|---|---|
| Declaration | static unsigned char master_rcv_int (unsigned short rb_data) | | |
| Outline | Master receive processing | | |
| Argument | Argument name | | Meaning |
| | unsigned short rb_data | | Data read from the U2RB register |
| Variable (global) | Variable name | | Contents |
| | unsigned char iic_index | | Number of transmit/receive bytes |
| | unsigned char iic_length | | Transfer data length |
| | unsigned char far *iic_pointer | | Transmit/receive buffer pointer |
| Returned value | Type | Value | Meaning |
| | unsigned char | IIC_SP_ON | 0: Stop condition generated |
| | | IIC_SP_OFF | 1: Stop condition not generated |
| Function | Called from the UART2 transmit interrupt handling.<br>The argument value is stored in the receive buffer (except for the slave address data).<br>NACK is set to the transmit register when the following data is the last byte. ACK is set to the transmit register when the following data is not the last byte. After setting ACK or NACK to the transmit register, the next receive operation starts.<br>IIC_SP_OFF is returned in the following case:<br>• The following data is not the last byte data.<br>IIC_SP_ON is returned in the following case:<br>• The last byte receive operation is completed. | | |

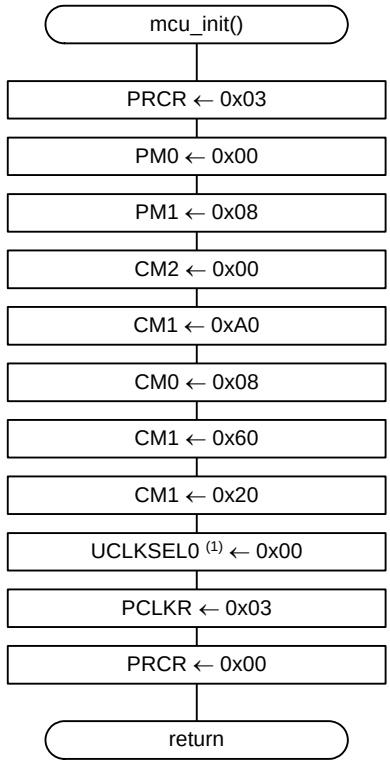
| | | | |
|---|---|---|---|
| Declaration | unsigned char iic_master_end (void) | | |
| Outline | Master control completed processing | | |
| Argument | Argument name | | Meaning |
| | None | | - |
| Variable (global) | Variable name | | Contents |
| | (structure member) iic_status | | All statuses |
| | (structure member) iic_start | | Mid-communication flag |
| | (structure member) iic_err_par | | Parameter error flag |
| | (structure member) iic_err_nack | | NACK detection error flag |
| | (structure member) iic_err_addr | | No address match error flag |
| Returned value | Type | Value | Meaning |
| | unsigned char | BUSY | Mid-communication |
| | | RDY | Communication completed |
| | | ADDR_ERR | Address not matched |
| Function | Called from the main function. Informs the user of the master control state completed. During communication, this function returns BUSY. When communication is completed, this function returns RDY. Additional processing after communication is completed can be added as needed. | | |

## 4.3 Main Processing




Note:
1. Additional processing can be added as needed.

## 4.4 System Clock Setting




Disable protection.

Single-chip mode

Internal area expansion: Entire area is usable

Start main clock oscillating.

Main clock division: divided-by-4

Enable bits CM16 and CM17.

Main clock division: divided-by-2

Main clock division: no division

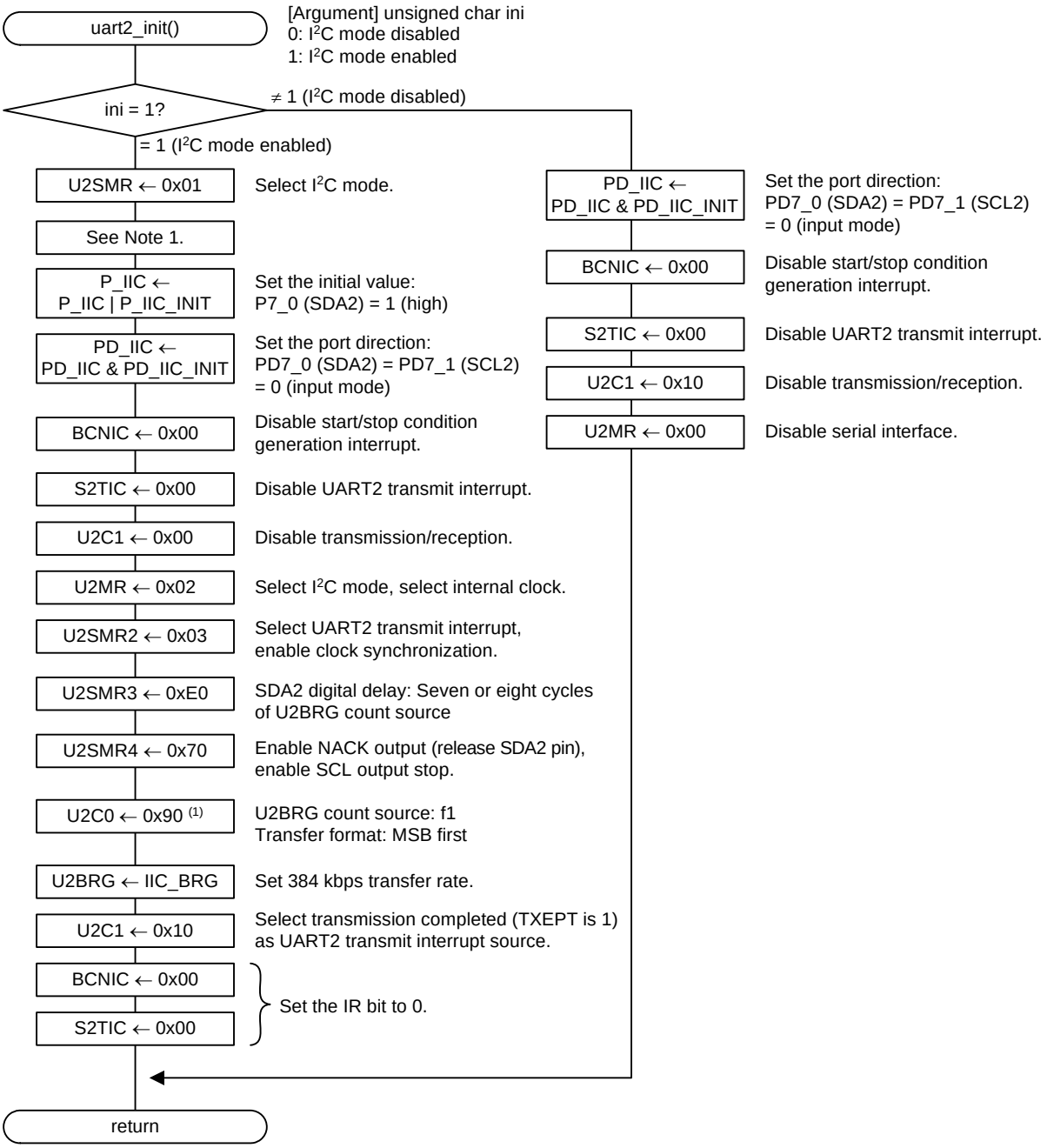
UART0 to UART2 clock prior to division: f1
UART5 to UART7 clock prior to division: f1

SI/O clock: f1SIO

Enable protection.

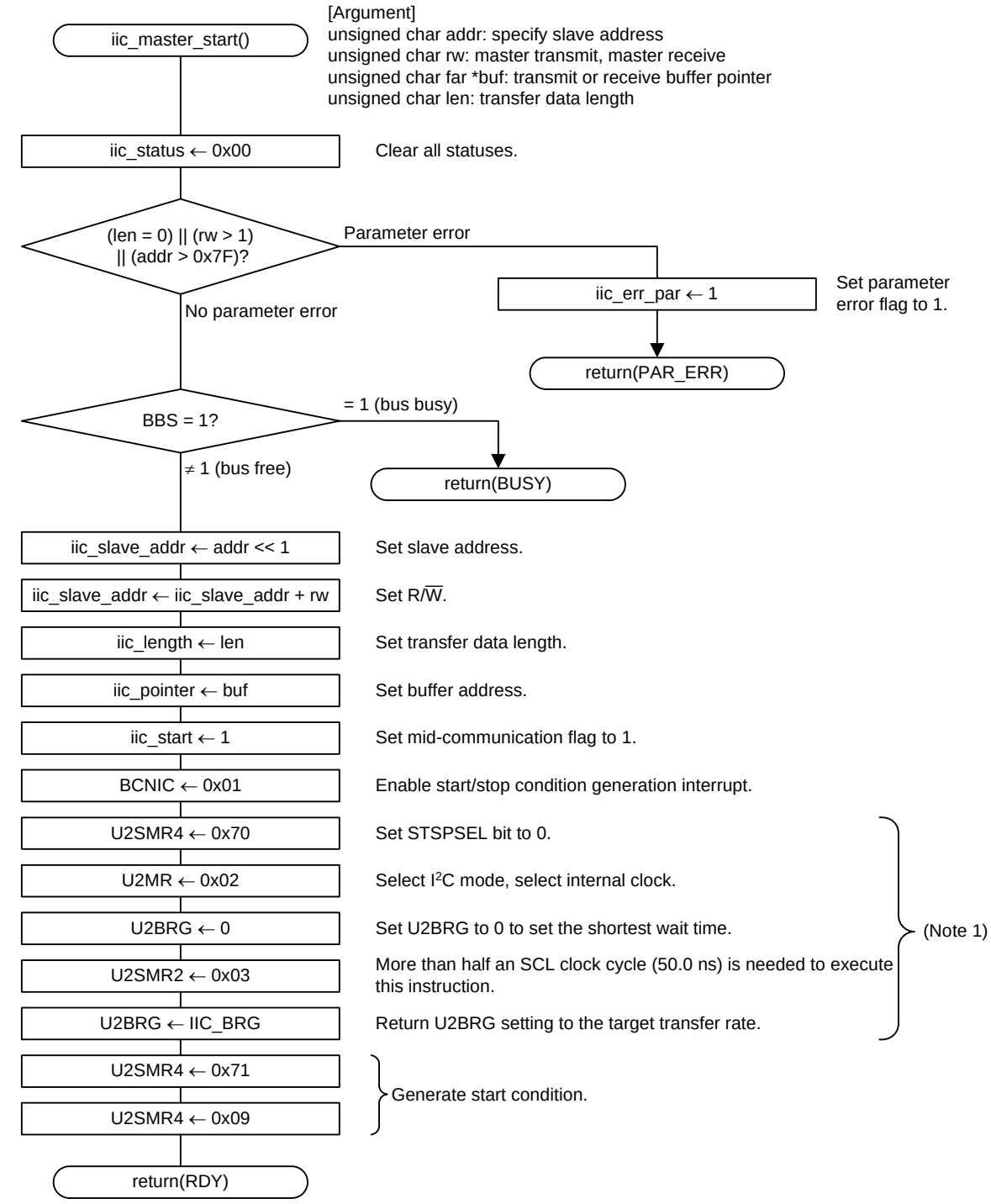
Note:
1.The UCLKSEL0 register may not be existed depending on MCU products.

## 4.5 UART2 Initial Setting



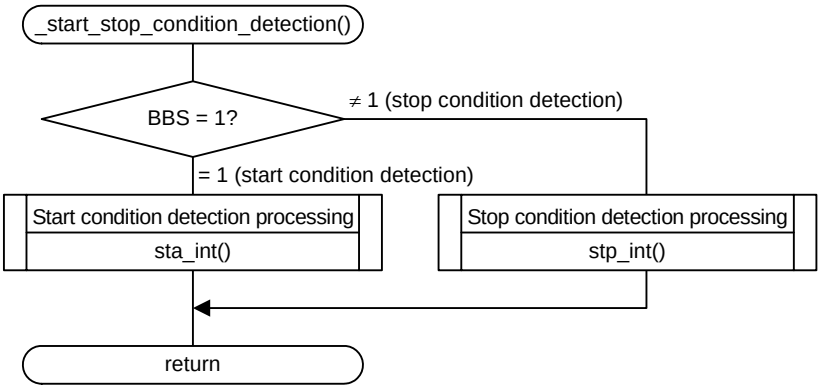

Note:
1. Ports P7_0 and P7_1 in the M16C/50 Series are not N-channel open drain outputs.
When using M16C/50 Series, set the NCH bit in the U2C0 register to 1 (pins TDXi/SDAi and SCLi are open drain output) before setting the initial value of SDAi output (P7_0).

## 4.6 Master Control Start Processing

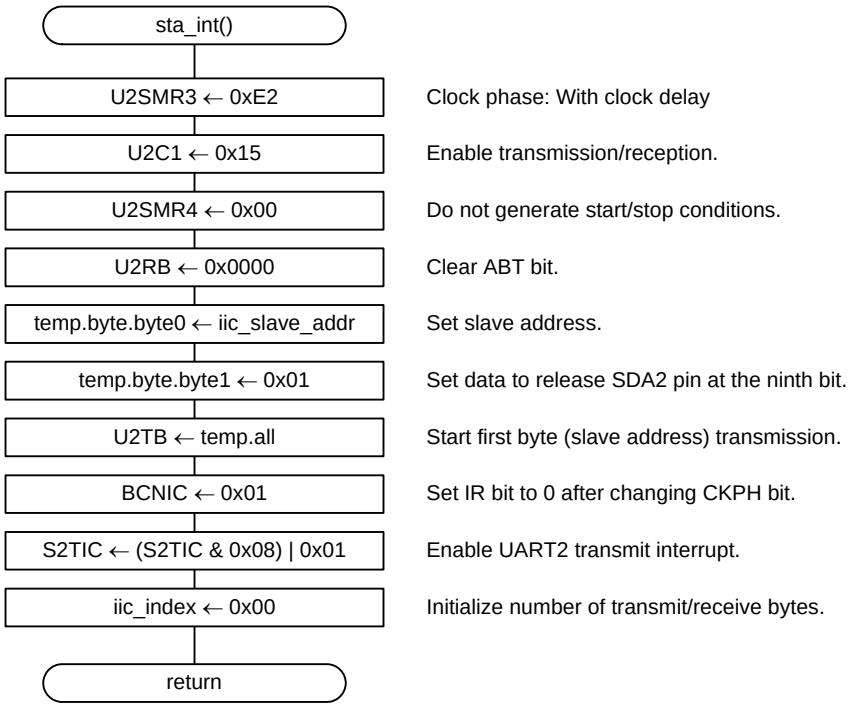



Note:
1. This process is considered in technical update (TN-16C-130A/EA).

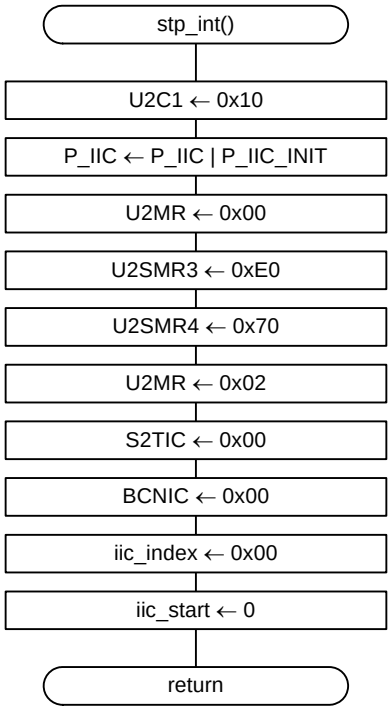
## 4.7 Start/Stop Condition Generation Interrupt Handling




## 4.8 Start Condition Detection Processing



## 4.9 Stop Condition Detection Processing




Disable transmission/reception.

Set the initial value: P7_0 (SDA2) = 1 (high)

Disable serial interface.

Set clock phase: No clock delay

Enable NACK output (release SDA2 pin), enable SCL output stop.

Select I²C mode, select internal clock.

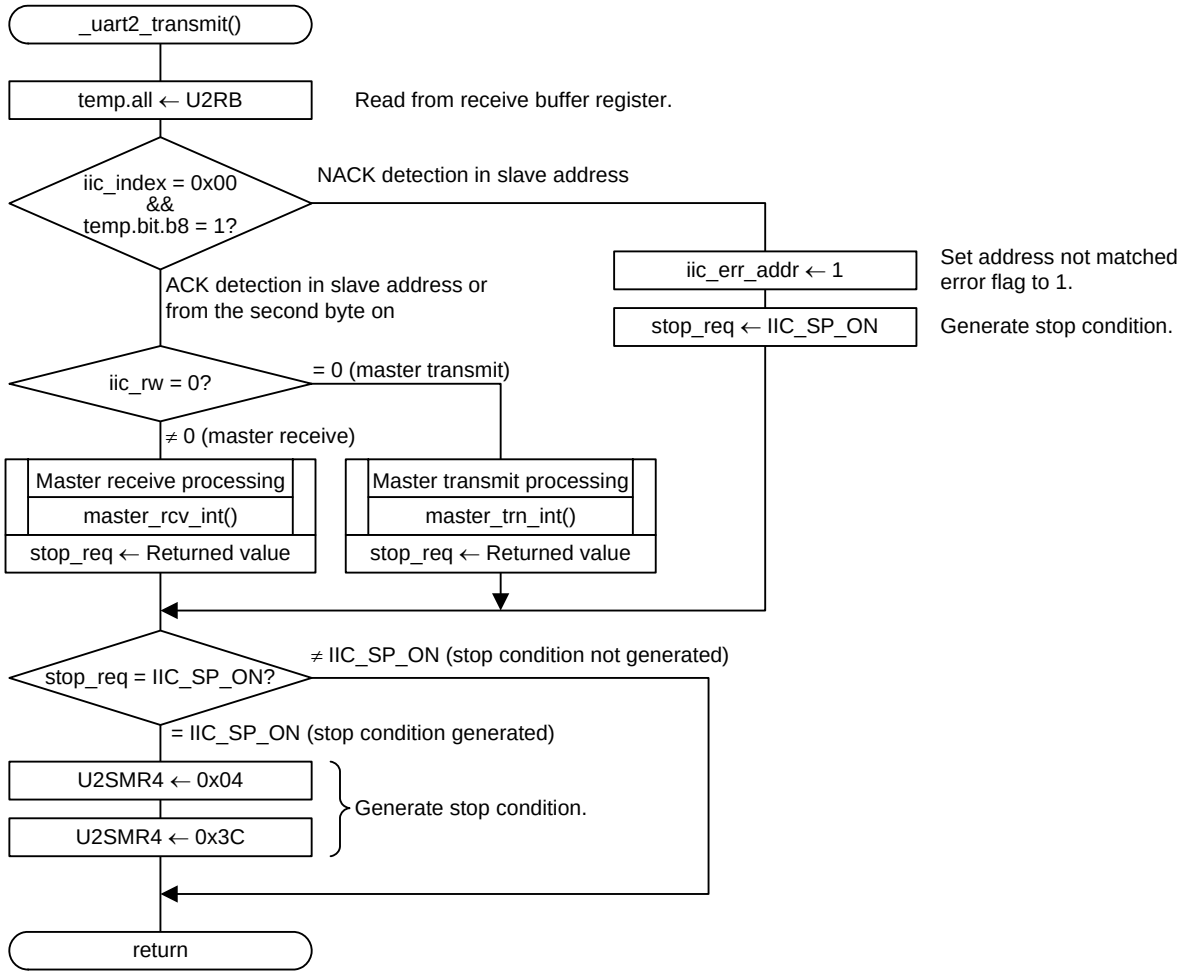
Disable UART2 transmit interrupt.

Disable start/stop condition generation interrupt.

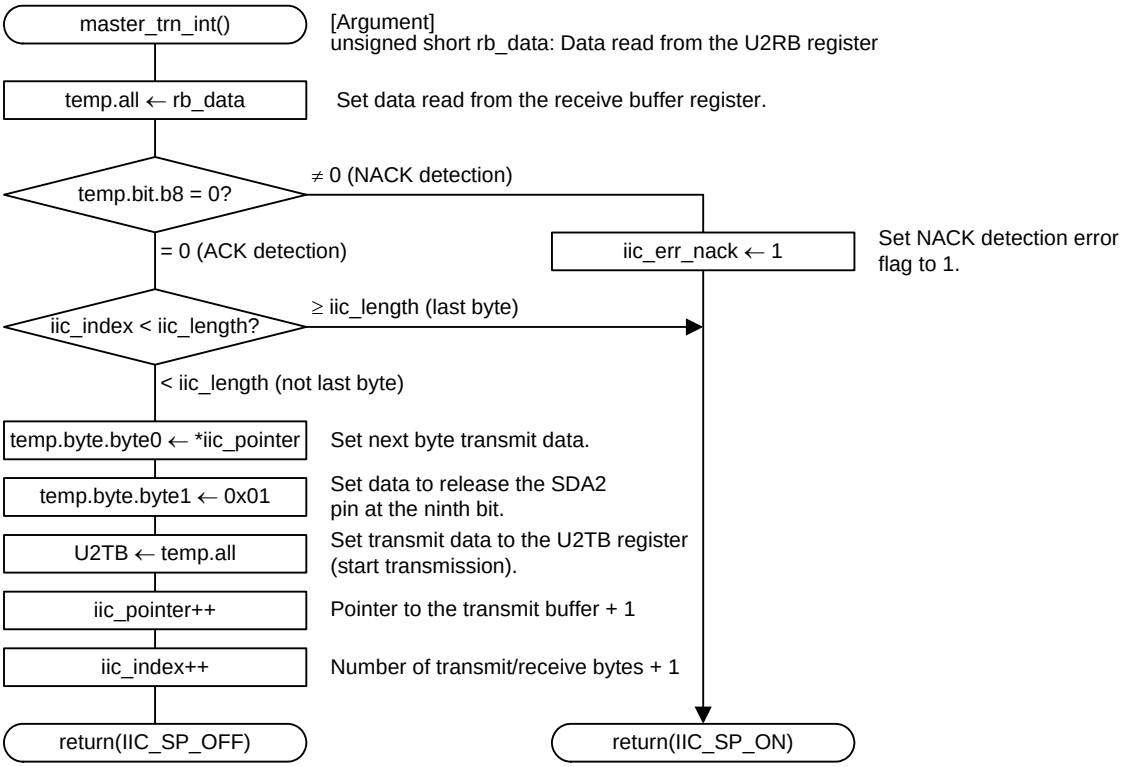
Initialize number of transmit/receive bytes.

Set mid-communication flag to 0.

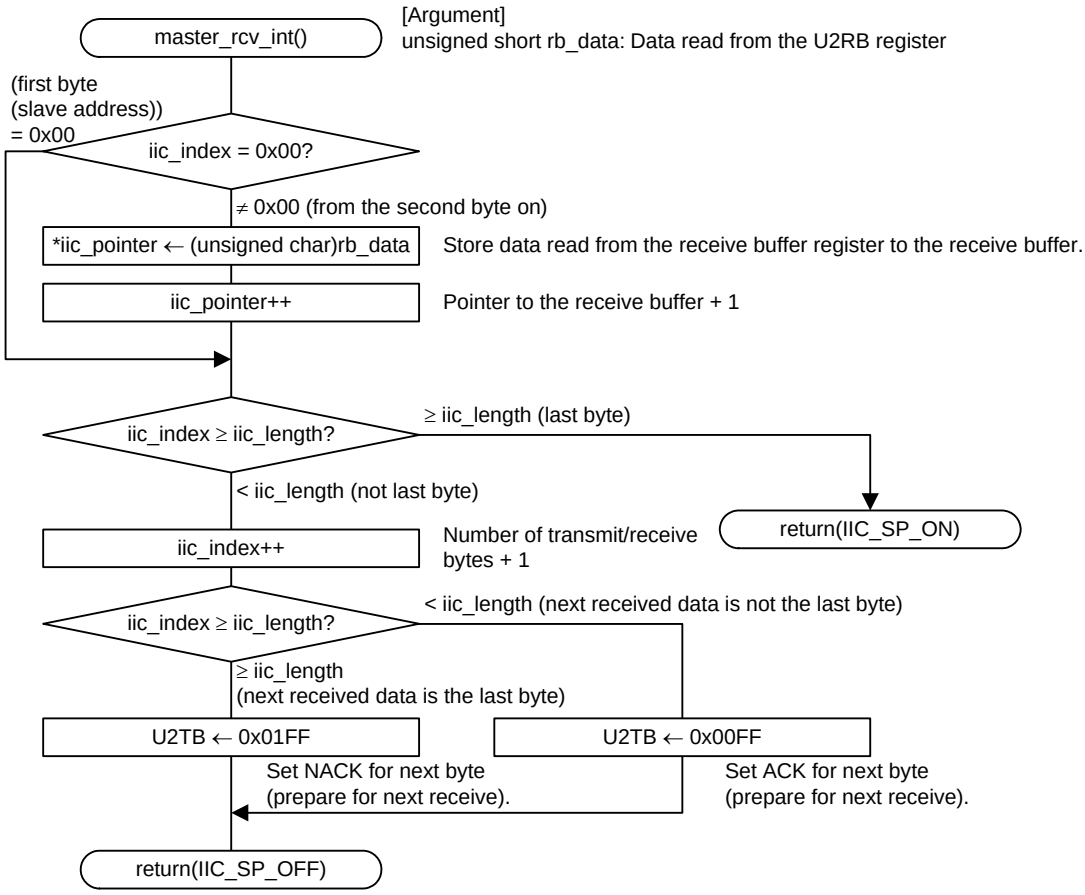
## 4.10 UART2 Transmit Interrupt Handling



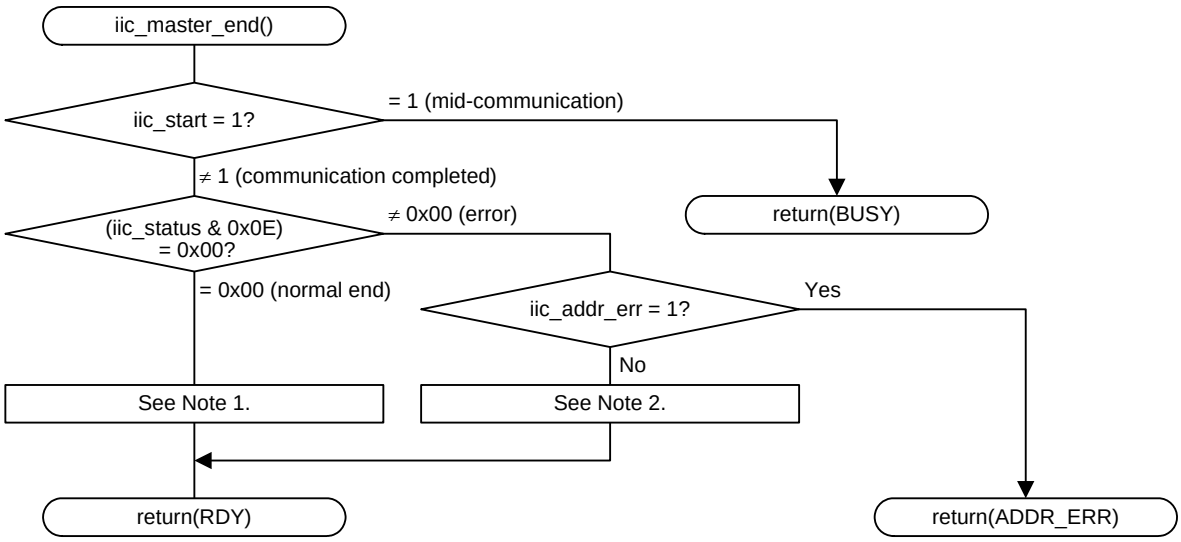
## 4.11 Master Transmit Processing



## 4.12 Master Receive Processing




## 4.13 Master Control Completed Processing




Notes:
1. Additional processing of communication completed normally can be added as needed.
2. Additional processing of communication completed with error can be added as needed.

# 5. Sample Program

A sample program can be downloaded from the Renesas Electronics website.
To download, click “Application Notes” in the left-hand side menu of the M16C Family page.

# 6. Reference Documents

Application Note
M16C Family I²C-Bus Interface Using UARTi Special Mode 1
(REJ05B1349-0100)
The latest version can be downloaded from the Renesas Electronics website.

M16C/63 Group User’s Manual: Hardware Rev.1.00
M16C/64 Group User’s Manual: Hardware Rev.1.05
M16C/64A Group User’s Manual: Hardware Rev.1.10
M16C/64C Group User’s Manual: Hardware Rev.0.10
M16C/65 Group User’s Manual: Hardware Rev.1.10
M16C/65C Group User’s Manual: Hardware Rev.0.10
M16C/6C Group User’s Manual: Hardware Rev.1.00
M16C/5LD Group, M16C/56DGroup User’s Manual: Hardware Rev.1.10
M16C/5L Group, M16C/56Group User’s Manual: Hardware Rev.1.00
M16C/5M Group, M16C/57Group User’s Manual: Hardware Rev.1.01
The latest version can be downloaded from the Renesas Electronics website.

Technical News/Technical Update
The latest information can be downloaded from the Renesas Electronics website.

C Compiler Manual
M16C Series, R8C Family C Compiler Package V.5.45
C Compiler User’s Manual Rev.2.00
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/inquiry

| Revision History | M16C/63, 64, 64A, 64C, 65, 65C, 6C, 5LD, 56D, 5L, 56, 5M, 57 Group<br>I²C-bus Interface Using UARTi Special Mode 1<br>(Master Transmit/Receive) |
|---|---|

| Rev. | Date | Description | |
|---|---|---|---|
| | | Page | Summary |
| 1.00 | Aug 31, 2010 | — | First edition issued |
| 1.01 | Jan 31, 2011 | — | Add M16C/64, M16C/64C,M16C/65C, M16C/56D, M16C/56, M16C/57 |
| | | | |

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

# 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 part number, confirm that the change will not lead to problems.

   — The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part numbers, implement a system-evaluation test for each of the products.

# 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.**
7F, 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 1.0