M16C/64C and M16C/65C Groups

Power Control
Low Current Consumption Setting

R01AN0644EJ0100
Rev. 1.00
Dec. 28, 2011

1. Abstract

This application note describes the setting method for low current consumption in the M16C/64C and M16C/65C Groups.

2. Introduction

The application example described in this document applies to the following microcomputers (MCUs):

- MCU: M16C/64C and M16C/65C Groups

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

3. Outline

The amount of current consumption correlates with the number of operating clocks and frequency. If there are fewer operating clocks and a lower frequency, current consumption will be low.
This section introduces a method to reduce current consumption in the M16C/65C Group. Because peripheral functions and electrical characteristics differ when other MCUs are used, refer to the corresponding user's manual.

3.1 Clock Operating Modes

3.1.1 Normal Operating Mode

In normal operating mode, the CPU clock and the peripheral function clocks are both supplied, thus the CPU and the peripheral functions are operating. Power control is exercised by controlling the CPU clock frequency. The higher the CPU clock frequency, the higher the processing capability. The lower the CPU clock frequency, the lower the power consumption in the chip. If unnecessary oscillators are stopped, current consumption is further reduced.

3.1.2 Wait Mode

In wait mode, the CPU clock stops, then the CPU, watchdog timer, and $\overline{\text{NMI}}/\overline{\text{SD}}$ digital filter also stop as they are operated by the CPU clock. However, if the CSPRO bit in the CSPR register is 1 (count source protection enabled), the watchdog timer remains active. Since the clock oscillator does not stop, peripheral functions which are provided peripheral function clocks keep operating.

Peripheral Function Clock Stop Function:
When the CM02 bit in the CM0 register is 1 (peripheral function clock f1 stops in wait mode), the f1 clock stops while in wait mode, then current consumption is reduced. With the exception of f1, peripheral clocks (i.e. fOCO40M, fOCO-F, fOCO-S, fC, and fC32) are not stopped by the CM02 bit.

3.1.3 Stop Mode

In stop mode, all oscillators stop. Therefore the CPU clock and peripheral function clocks stop, then the CPU and peripheral functions using these clocks stop operating. The least amount of current is consumed in this mode. If the voltage applied to pins VCC1 and VCC2 is VRAM or greater, the contents of the internal RAM are retained. When applying 2.7 V or less to pins VCC1 and VCC2, make sure VCC1 ≥ VCC2 ≥ VRAM.
The peripheral functions activated by external signals keep operating.

3.2 Peripheral Clock Stop Register 1 (PCLKSTP1)

The peripheral clock stop register 1 (PCLKSTP1) in the M16C/64C and M16C/65C Groups can be used to reduce current consumption by disabling f1 provision to peripheral functions which do not use f1.

3.2.1 PCKSTP1A (Peripheral Clock Stop Bit)

When not using f1 as the clock source for the following peripherals, set the PCKSTP1A bit to 1 (f1 provide disabled) to reduce current consumption:

Real-time clock, pulse width modulator, remote control signal receiver,
serial interface UART0 to UART2, UART5 to UART7, SI/O3, SI/O4
multi-master I^2C-bus interface, A/D converter

3.2.2 PCKSTP11 (Timer Peripheral Clock Stop Bit)

When not using f1 as the clock source for the count source of timer A and timer B, set the PCKSTP11 bit to 1 (f1 provide disabled) to reduce current consumption.

3.2.3 PCKSTP17 (Timer Clock Source Select Bit)

Change the PCKSTP17 bit when both of the following conditions are met:

- f1 and the main clock are both stably provided.
- All timers in timer A and timer B are stopped.

The function of the PCKSTP17 bit in the PCLKSTP1 register provides the main clock to timer A and timer B.
When in PLL operating mode, high-speed mode, medium-speed mode, or wait mode, the main clock can be used as the count source of timer A and timer B.
Do not use the main clock as the count source of timer A and timer B in other normal operating modes.

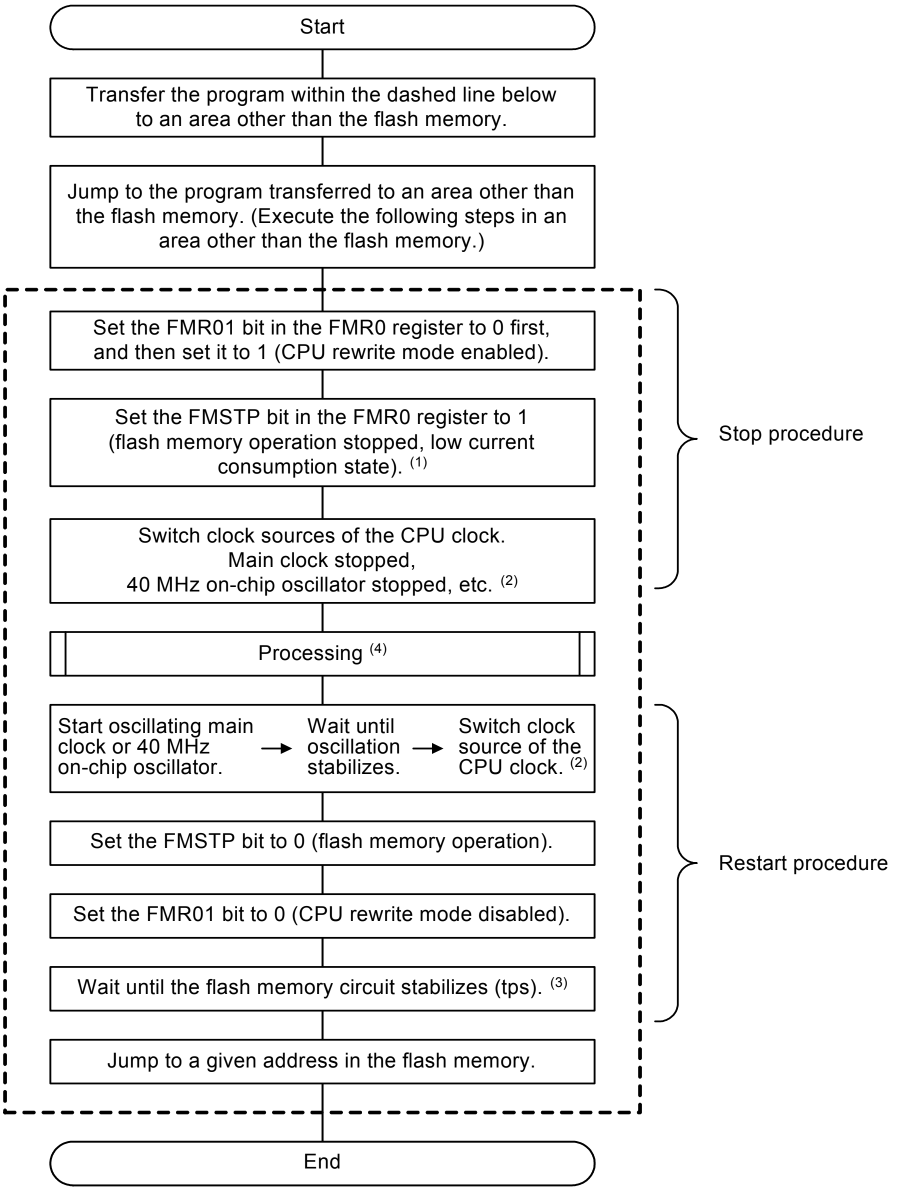
When using the main clock as the clock source of timer A and timer B count sources (PCKSTP17 bit in the PCLKSTP1 register is 1), follow the notes below.

- Set the PCKSTP11 bit in the PCLKSTP1 register to 0 (f1 provide enabled).
- The main clock can be used as the count source for timer A and timer B when in PLL operating mode, high-speed mode, medium-speed mode, or wait mode. When in normal operating modes other than aforementioned modes, the main clock cannot be used as the count source for timer A and timer B.
- When the PCKSTP17 bit is set to 1, and timer A and timer B are still operating during wait mode, set the CM02 bit to 0 (peripheral function clock f1 does not stop in wait mode).

3.3 Power Control in Flash Memory

3.3.1 Stopping Flash Memory

When the flash memory is stopped, current consumption is reduced. While the flash memory is stopped, execute a program in any area other than the flash memory. Figure 3.1 shows the Stopping and Restarting the Flash Memory.



Notes:
1. Set the FMSTP bit to 1 after the FMR01 bit is set to 1.
2. Wait until the target clock stabilizes before switching the clock source of the CPU clock.
3. Add tps wait time by a program. Do not access the flash memory during this wait time.

Figure 3.1 Stopping and Restarting the Flash Memory

3.3.2 Reading the Flash Memory

Current consumption while reading the flash memory can be reduced by setting bits FMR22 and FMR23 in the FMR2 register.

3.3.2.1 Slow Read Mode

Slow read mode can be used when f(BCLK) is slower than or equal to f(SLOW_R) and the PM17 bit in the PM1 register is 1 (1 wait). When the sub clock or 125 kHz on-chip oscillator clock is used as the clock source of the CPU clock, a wait is not necessary.
(Technical update number: TN-16C-A179A/E).
Figure 3.2 shows Setting and Canceling Slow Read Mode.

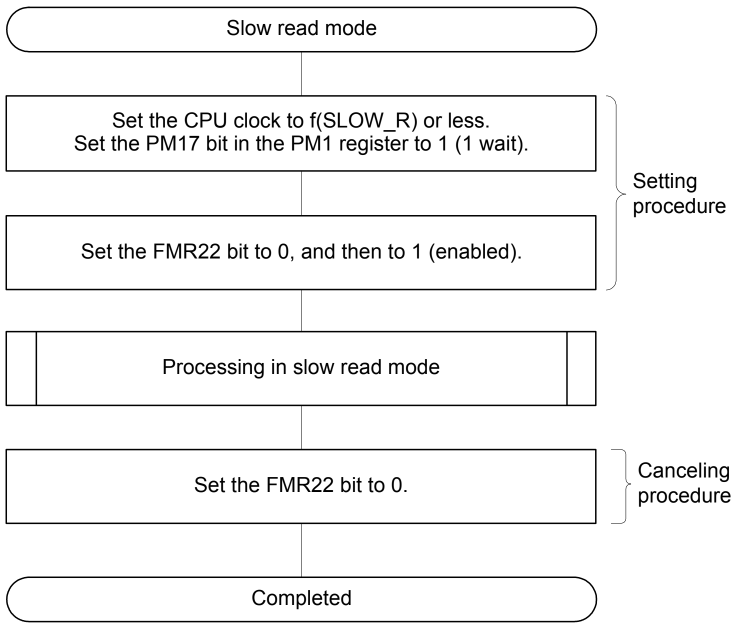


Figure 3.2 Setting and Canceling Slow Read Mode

Do not set the FMR22 bit in the FMR2 register to 1 (slow read mode enabled) when the FMR01 bit in the FMR0 register is 1 (CPU rewrite mode enabled).

3.3.2.2 Low Current Consumption Read Mode

Low current consumption read mode can be used when the CM07 bit in the CM0 register is 1 (sub clock used as CPU clock). Figure 3.3 shows Setting and Canceling Low Current Consumption Read Mode.

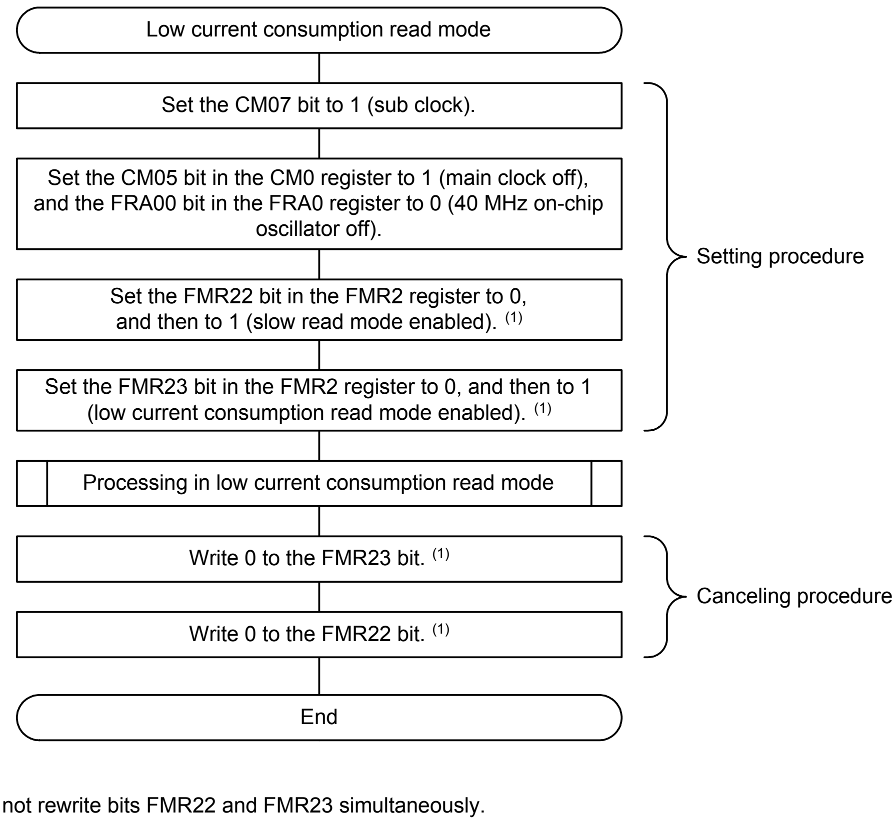


Figure 3.3 Setting and Canceling Low Current Consumption Read Mode

Do not enter wait mode from low current consumption read mode. To enter wait mode from this mode, set the FMR23 bit in the FMR2 register to 0 (low current consumption read mode disabled).
Do not enter wait mode from CPU rewrite mode. To enter wait mode from this mode, set the FMR01 bit in the FMR0 register to 0 (CPU rewrite mode disabled), and disable DMA transfer.

Do not enter stop mode from low current consumption read mode. To enter stop mode from this mode, set the FMR23 bit in the FMR2 register to 0 (low current consumption read mode disabled).
Do not enter stop mode from CPU rewrite mode. To enter stop mode from this mode, set the FMR01 bit in the FMR0 register to 0 (CPU rewrite mode disabled), and disable DMA transfer.

Enter low current consumption read mode through slow read mode.
When the FMR23 bit in the FMR2 register is 1 (low current consumption read mode enabled), do not set the FMSTP bit to 1 (flash memory stopped). Also, when the FMSTP bit is 1, do not set the FMR23 bit to 1.
When the FMR01 bit in the FMR0 register to 1 (CPU rewrite mode enabled), do not set the FMR23 bit in the FMR2 register to 1 (low current consumption read mode enabled).

3.4 Reducing Current Consumption

To reduce current consumption, refer to the descriptions below when designing a system or writing a program.

3.4.1 Ports

The MCU retains the state of each I/O port even when it enters wait mode or stop mode. A current flows in the active output ports. A shoot-through current flows to the input ports in the high-impedance state. When entering wait mode or stop mode, first set unused ports to input and stabilize the potential.

3.4.2 A/D Converter

When not performing A/D conversion, set the ADSTBY bit in the ADCON1 register to 0 (A/D operation stop). Also, before entering wait mode or stop mode, fix analog pins to the stabilized potential.

3.4.3 D/A Converter

When not performing D/A conversion, set the DAiE bit in the DACON register to 0 (output disabled) and the DAi register to 00h (i = 0, 1).

3.4.4 Stopping Peripheral Functions

Use the PCLKSTP1 register to disable providing f1 to the peripheral functions not using f1.
Use the CM02 bit in the CM0 register to stop unnecessary peripheral functions while in wait mode.

3.4.5 Switching the Oscillation-Driving Capacity

Set the driving capacity to low when oscillation is stable.

4. Application Example

4.1 Settings in the Sample Code

The sample code configures the settings below from the settings for low current consumption described in 3. Outline. The M16C/65C Group is used in the sample code.

Table 4.1 lists the Settings of Sample Code.

Table 4.1 Settings of Sample Code

Item		When in Low Current Consumption Mode	When in Normal Operating Mode
Program location		RAM	ROM
Clocks	CPU clock	Sub clock (low drive capacity)	Main clock (no division) (low drive capacity)
	Main clock	Stopped	Oscillating
	125 kHz on-chip oscillator		Stopped (oscillating only after reset)
	40 MHz on-chip oscillator		Stopped
	Sub clock	Oscillating	Oscillating
Peripheral clock (except timers A and B)		f1 provision disabled	
Timer peripheral clock (timers A and B)		f1 provision disabled	
Flash memory operation		Stopped	Operating

Settings for the other peripheral functions are as follows:

- A/D converter: A/D operation stopped
- D/A converter: D/A0 output disabled, D/A1 output disabled
- Unused ports: Input mode (pull-down [(1)])

Note:

1. Pull-down is performed externally in the sample code.

4.2 Sample Code Operation

The following shows the sample code operation.

(1) Initialize the CPU and peripheral functions.
(2) Wait until the $\overline{\text{INT0}}$ interrupt occurs.
(3) Transfer the program used in the RAM.
(4) Set the start address of the relocatable vector table for the RAM in the INTB register.
(5) Jump to the RAM, and execute the program.
 (5)-1 Set the sub clock as the CPU clock.
 (5)-2 Stop the main clock and on-chip oscillator.
 (5)-3 Enable CPU rewrite mode.
 (5)-4 Stop the flash memory, and set to low current consumption status.
 (5)-5 Wait until the $\overline{\text{INT1}}$ interrupt occurs.
 (5)-6 Enable flash memory operation. [1]
 (5)-7 Disable CPU rewrite mode.
 (5)-8 Wait until the flash memory circuit becomes stable.
 (5)-9 Set the main clock (no division) as the CPU clock after the main clock oscillates.
(6) Set the start address of relocatable vector table for the ROM in the INTB register, and return to (2).

Note:
1. When enabling flash memory operation (FMSTP bit is set to 0), wait for tps or longer after the flash memory stops (FMSTP bit is set to 1).
 tps: Wait time until the flash memory circuit stabilizes.

Figure 4.1 shows the Operation Outline.

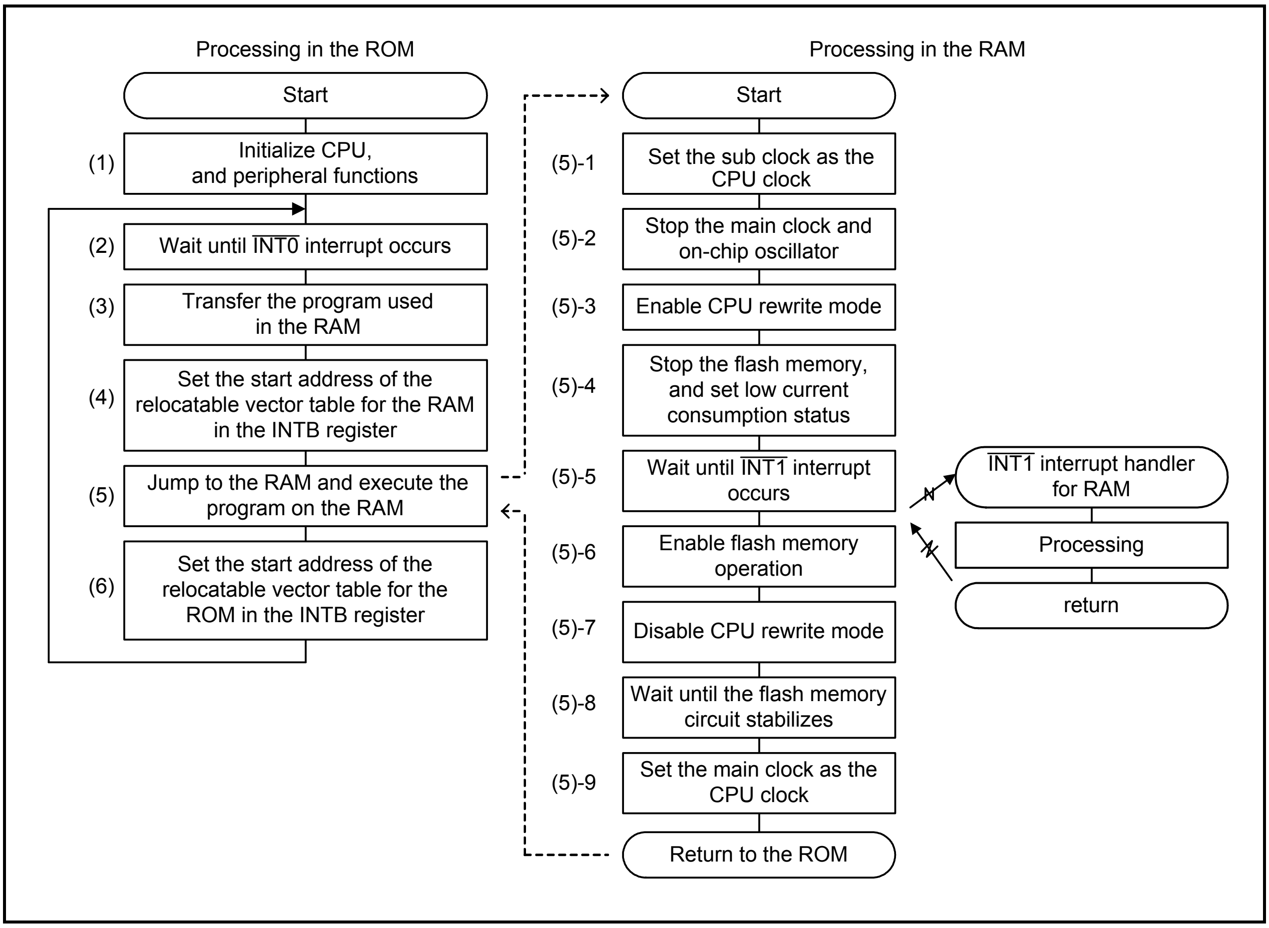


Figure 4.1 Operation Outline

4.3 Function Tables

Declaration	void main(void)
Outline	Main function
Argument	None
Variable (global)	None
Returned value	None
Function	When the $\overline{\text{INT0}}$ interrupt occurs, the program written in the internal flash memory is transferred to the RAM. Then the transferred program on the RAM is executed.

Declaration	void mcu_init(void)
Outline	CPU initialization
Argument	None
Variable (global)	None
Returned value	None
Function	Set the main clock (no division) as the CPU clock.

Declaration	void peripheral_init(void)
Outline	Peripheral low current consumption setting
Argument	None
Variable (global)	None
Returned value	None
Function	Disable f1 provision to the peripheral functions, disable f1 provision to the timer peripheral functions, set XIN-XOUT drive capacity to low, and set ports, the A/D converter, and the D/A converter.

<table>
<tr><td>Declaration</td><td colspan="3">void stopping_flash_memory(void)</td></tr>
<tr><td>Outline</td><td colspan="3">Flash memory stop/operate function</td></tr>
<tr><td>Argument</td><td colspan="3">None</td></tr>
<tr><td rowspan="2">Variable (global)</td><td>Type</td><td>Variable name</td><td>Contents</td></tr>
<tr><td>unsigned char</td><td>flg_wait_int1_int</td><td>For checking an $\overline{\text{INT1}}$ interrupt occurrence</td></tr>
<tr><td>Returned value</td><td colspan="3">None</td></tr>
<tr><td>Function</td><td colspan="3">Enable CPU rewrite mode, disable the flash memory operation, and set the mode to low current consumption mode. Enable the flash memory operation after the $\overline{\text{INT1}}$ interrupt occurs.</td></tr>
</table>

Declaration	void cpu_slow(void)
Outline	System clock slow function
Argument	None
Variable (global)	None
Returned value	None
Function	Set the sub clock as the CPU clock. Stop the main clock and on-chip oscillator.

Declaration	void cpu_fast(void)
Outline	System clock fast function
Argument	None
Variable (global)	None
Returned value	None
Function	Oscillate the main clock by setting the CM05 bit in the CM0 register. After the main clock oscillation has been confirmed by the CM23 bit in the CM2 register 10 times continuously, set the main clock (no division) as the CPU clock.

Declaration	void send_to_ram(void)
Outline	Send to RAM function
Argument	None
Variable (global)	None
Returned value	None
Function	Transfer the flash memory stop/operate function, system clock slow function, and system clock fast function to the RAM.

Declaration	void send_to_ram_vector(void)
Outline	Send to RAM vector function
Argument	None
Variable (global)	None
Returned value	None
Function	Transfer an interrupt handler used in the RAM.

Declaration	void renewal_of_ram_vector_t(void)
Outline	Renewal of RAM vector table function
Argument	None
Variable (global)	None
Returned value	None
Function	Create the relocatable vector table used in the RAM.

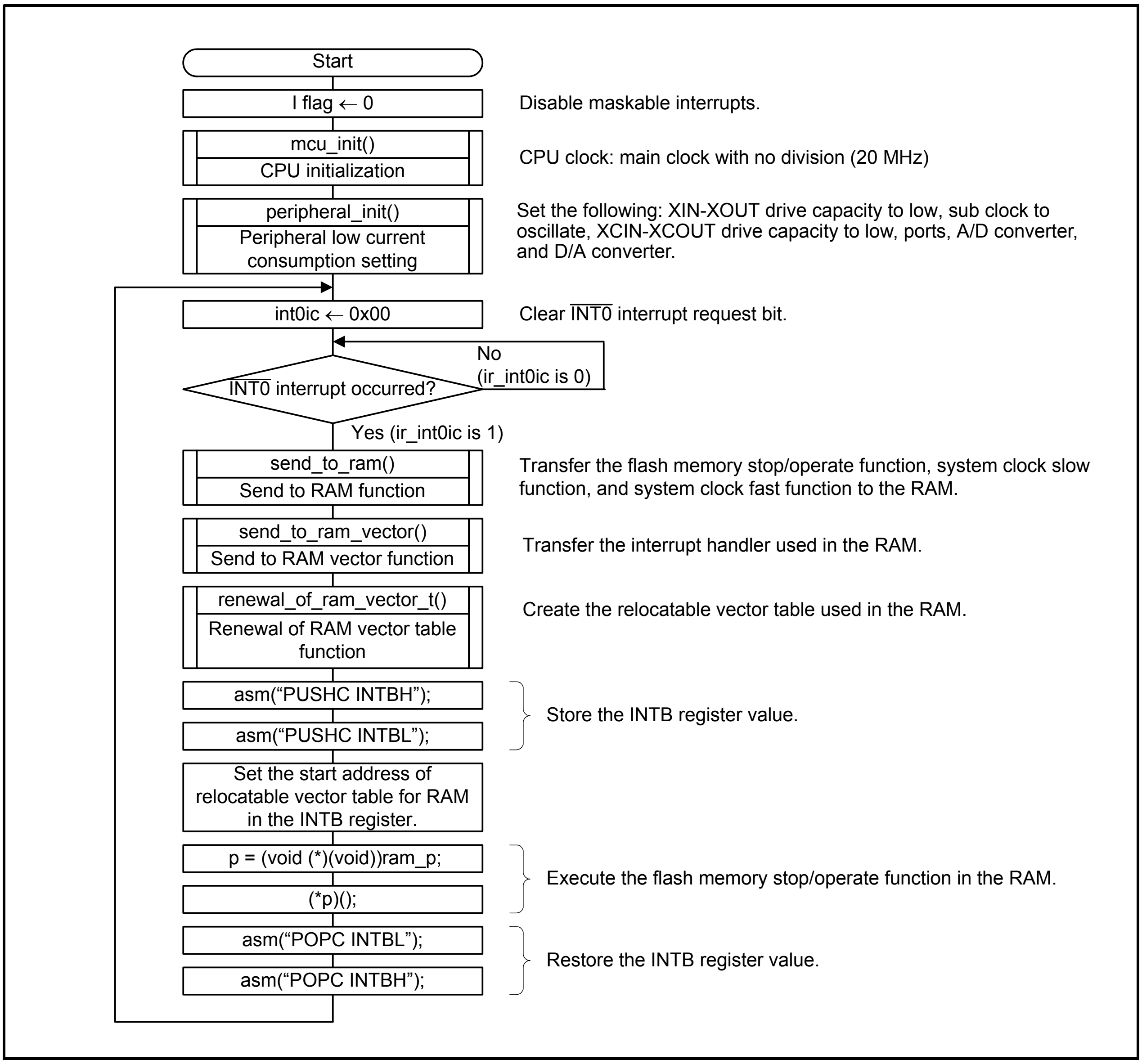
Declaration	void asm_smovf(void_far *_source, void _near *_dest, unsigned int _size)	
Outline	RAM transfer function	
Argument	Argument name	Meaning
	void _far *_source	Transferred source address (program)
	void _near *_dest	Transferred destination address (RAM area)
	unsigned int _size	Transfer size
Variable (global)	None	
Returned value	None	
Function	Transfer the specified area to the RAM area.	

Declaration	void ram_int_dummy(void)
Outline	RAM interrupt dummy function
Argument	None
Variable (global)	None
Returned value	None
Function	Dummy function for the RAM. Add a processing if needed.

Declaration	void ram_int_int1(void)		
Outline	RAM $\overline{\text{INT1}}$ interrupt function		
Argument	None		
Variable (global)	Type	Variable name	Contents
	unsigned char	flg_wait_int1_int	For checking an $\overline{\text{INT1}}$ interrupt occurrence
Returned value	None		
Function	Invert port P0_0.		

4.4 Flowcharts

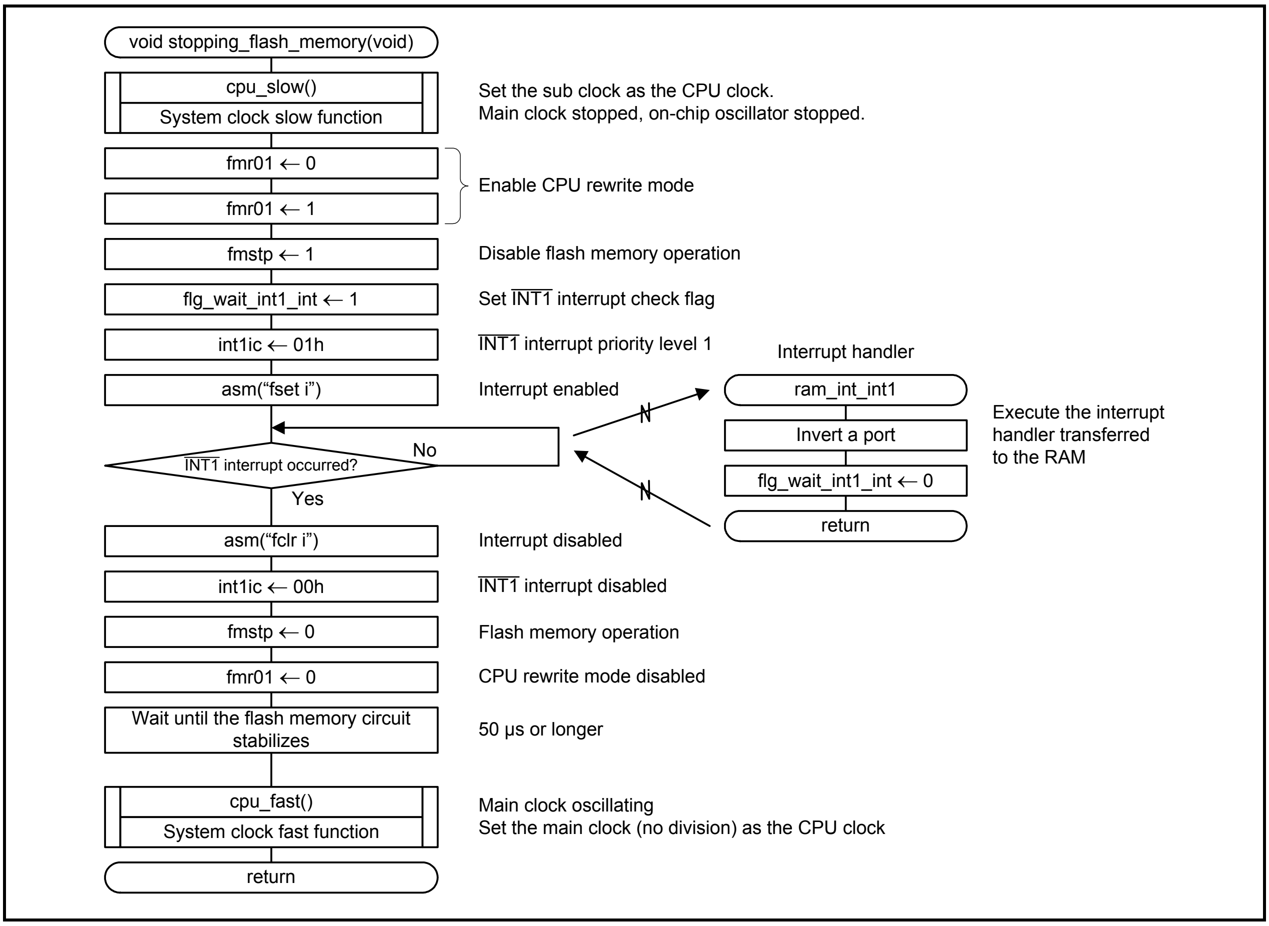
4.4.1 Main Processing



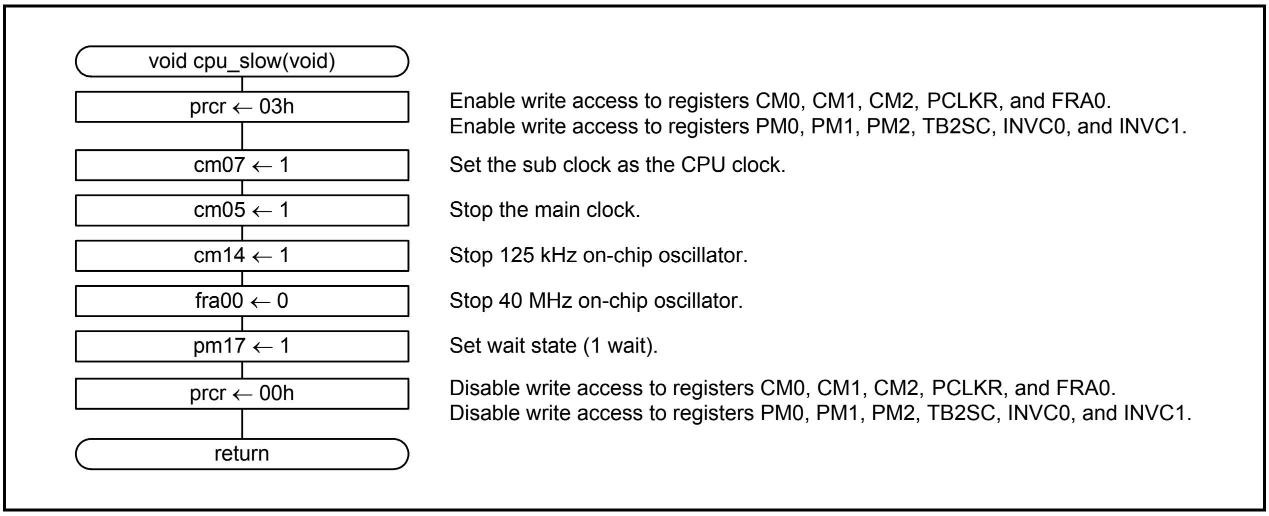
4.4.2 Peripheral Low Current Consumption Setting

void peripheral_init(void)	
prcr ← 01h	Enable write access to registers CM0, CM1, CM2, PCLKR, and FRA0.
pclkstp1 ← (pclkstp1 & 0x80) | 0x03	Disable f1 provision to the peripheral functions (other than timer A and timer B). Disable f1 provision to the timer peripheral functions (timer A and timer B).
cm15 ← 0	Set the XIN-XOUT drive capacity to low.
cm04 ← 1	Set the XCIN-XCOUT oscillation function.
Wait until sub clock oscillation stabilizes	Wait for approximately 1 second.
cm03 ← 0	Set the XCIN-XCOUT drive capacity to low.
prcr ← 00h	Disable write access to registers CM0, CM1, CM2, PCLKR, and FRA0.
Port initialization	Initialize unused ports.
adstby ← 0	Stop A/D operation.
da0e ← 0	Disable D/A0 output.
da1e ← 0	Disable D/A1 output.
da0 ← 00h	Set D/A0 output value to 0.
da1 ← 00h	Set D/A1 output value to 0.
return	

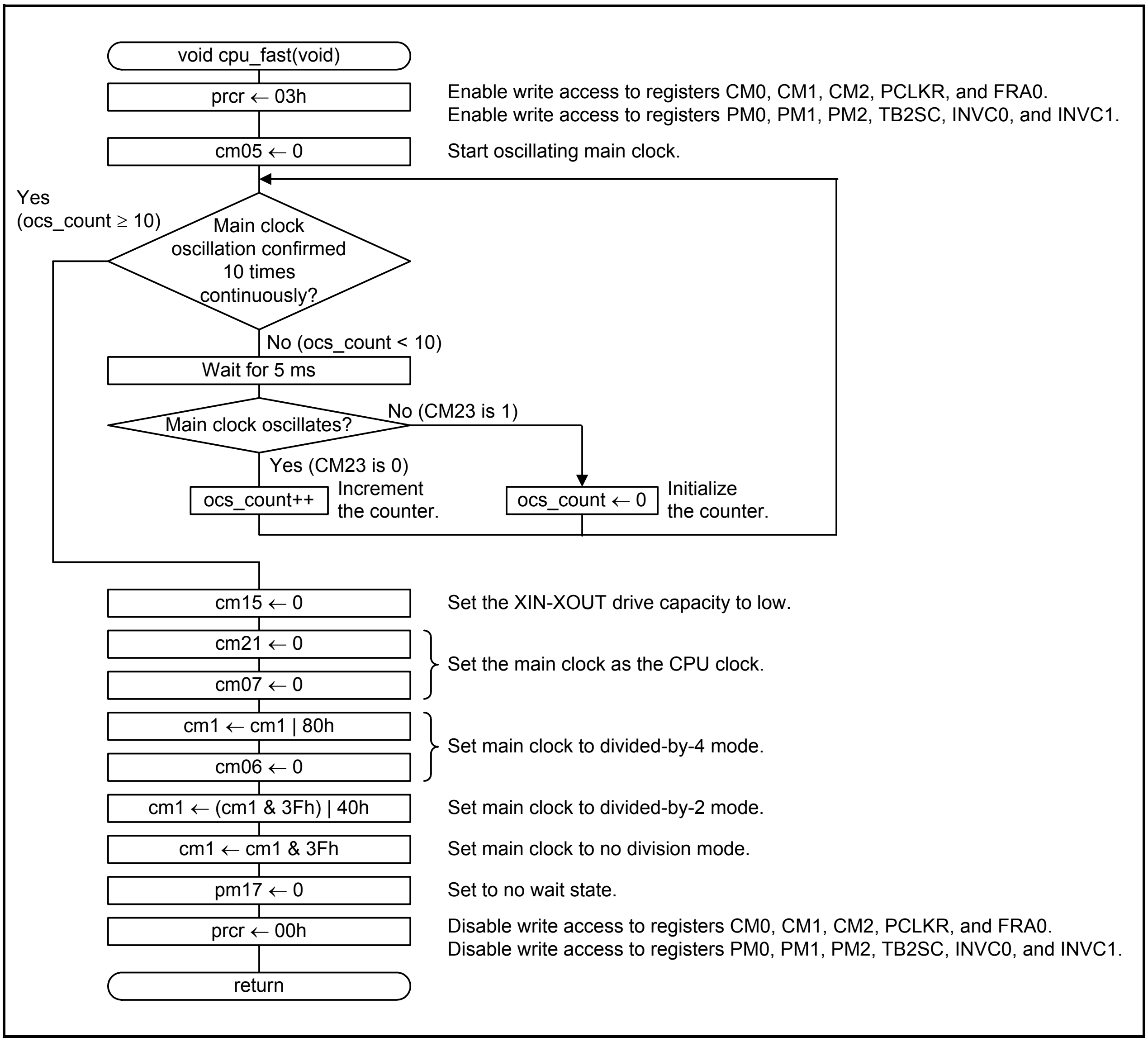
4.4.3 Flash Memory Stop/Operate Function



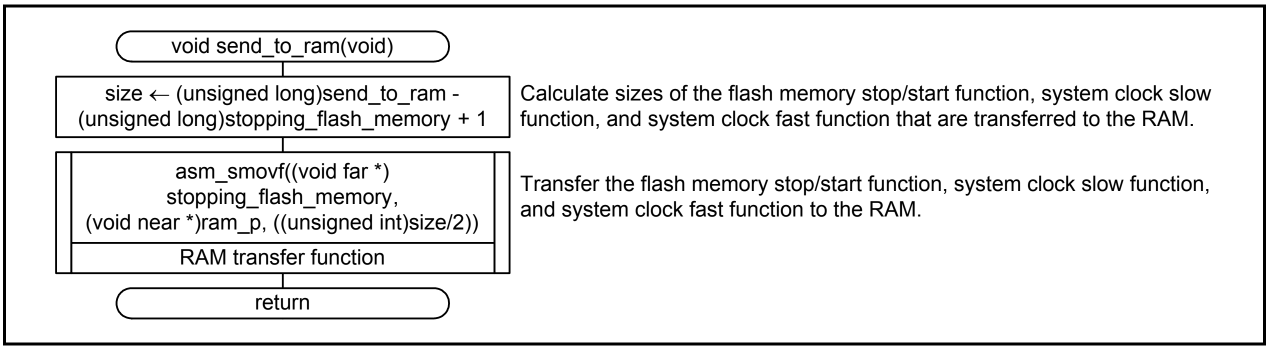
4.4.4 System Clock Slow Function



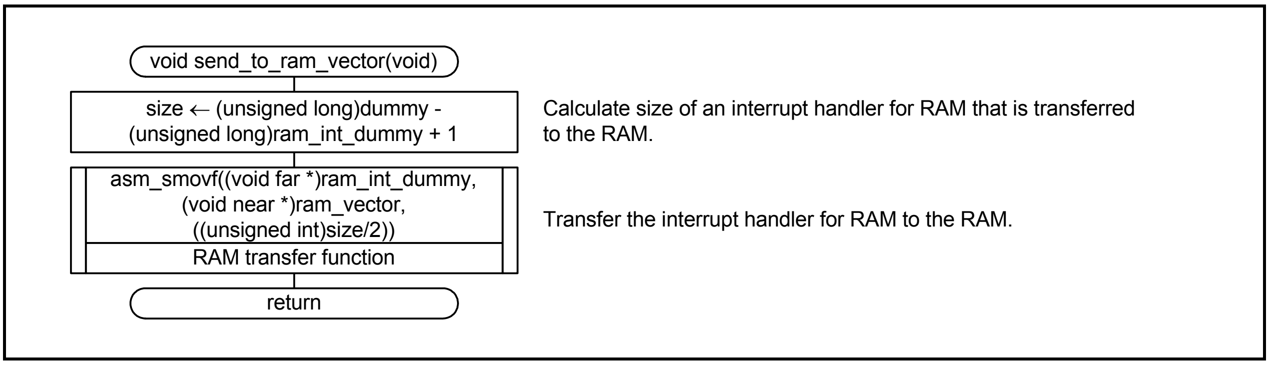
4.4.5 System Clock Fast Function



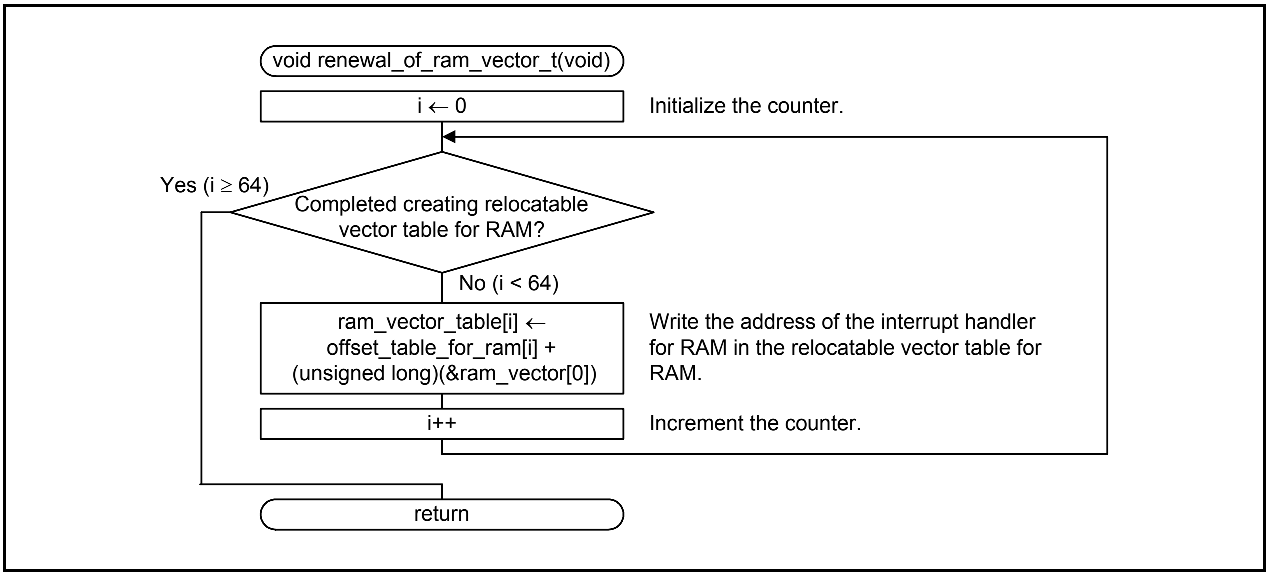
4.4.6 Send to RAM Function



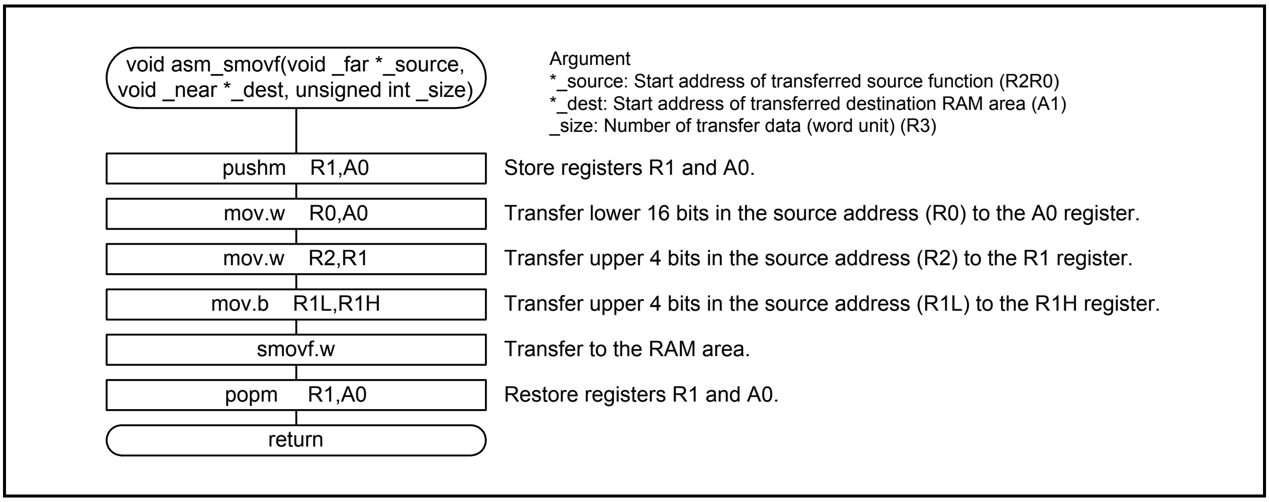
4.4.7 Send to RAM Vector Function



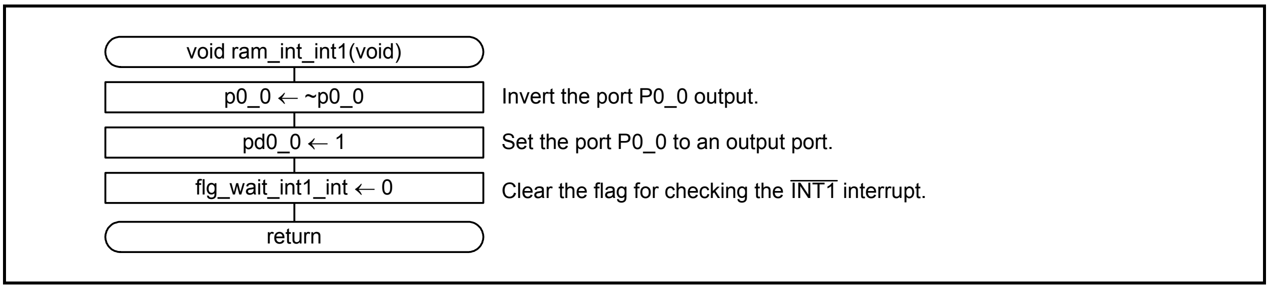
4.4.8 Renewal of RAM Vector Table Function



4.4.9 RAM Transfer Function



4.4.10 RAM $\overline{\text{INT1}}$ Interrupt Function



5. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

6. Reference Documents

M16C/64C Group User's Manual: Hardware Rev. 1.00
M16C/65C Group User's Manual: Hardware Rev. 1.00
The latest versions can be downloaded from the Renesas Electronics website.

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

C Compiler Manual
M16C Series/R8C Series 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/64C and M16C/65C Groups Power Control: Low Current Consumption Setting

Rev.	Date	Description	
		Page	Summary
1.00	Dec. 28, 2011	—	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 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.

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

Colophon 1.1