To our customers,

Old Company Name in Catalogs and Other Documents

   On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

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

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

RENESAS

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.

M16C/62P Group

Controlling Power Using Wait Mode

1.   Abstract

The following are steps for controlling power using wait mode. Figure 1 shows the operation timing, and Figures 2 to 3 show the set-up procedure. A reference program is an example when using the INT0,Timer B2 interrupt based on the setting procedure of Figure 2 to Figure 4.

2.   Introduction

This application note is applied to the M16C/62P group Microcomputers.

This program can be operated under the condition of M16C family products with the same SFR(Special Function Register) as M16C/62P Group products. Because some functions may be modified of the M16C family products, see the user's manual. When using the functions shown in this application note, evaluate them carefully for an operation

3. Specifications

Use the following peripheral functions:

● Timer mode of timer B

● Wait mode

A flag named "F_WIT" is used in the set-up procedure. The purpose of this flag is to decide whether or not to clear wait mode. If F_WIT="1"in the main program, the wait mode is entered; if F_WIT="0", the wait mode is cleared.

(1) Connect a 32.768-kHz oscillator to XCIN to serve as the timer count source. As interrupts occur every one second, which is a count the timer reaches, the controller returns from wait mode and count the clock a using a program. Also the RAM named "WACTH_CNT" is used in the reference program in order to count a clock.

(2) Clear wait mode if a $\overline{INT0}$ interrupt request occurs

4. Operation

(1) Switch the system clock from XIN to XCIN to get low-speed mode.

(2) Stop XIN and enter wait mode. in this instance, enable the timer B2 interrupt and the $\overline{INT0}$ interrupt.

(3) When a timer B2 interrupt request occurs (at 1-second intervals), start supplying the BCLK from XCIN. At this time, count the clock within the routine that handles the timer B2 interrupts and enter wait mode again.

(4) If a $\overline{INT0}$ interrupt occurs, start supplying the BCLK from XCIN. Start the XIN oscillation within the $\overline{INT0}$ interrupt, and switch the system clock to XIN.
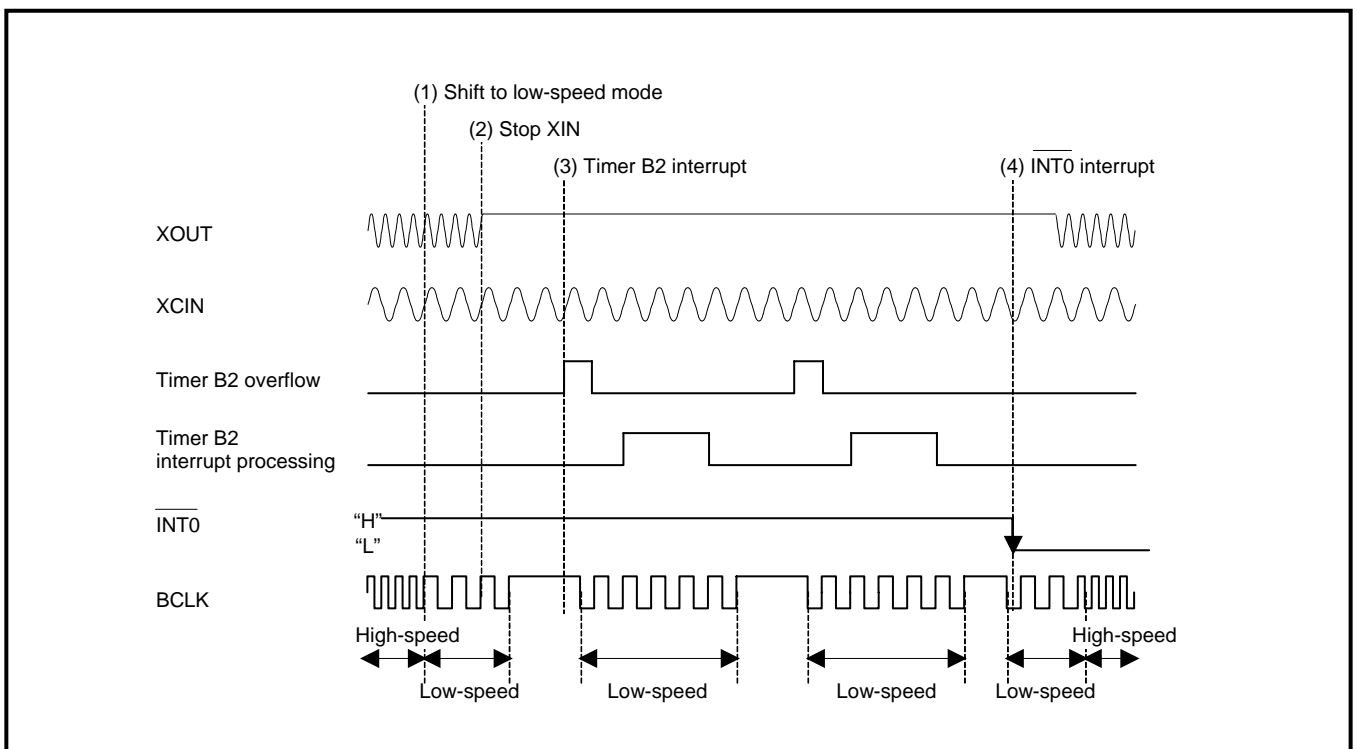

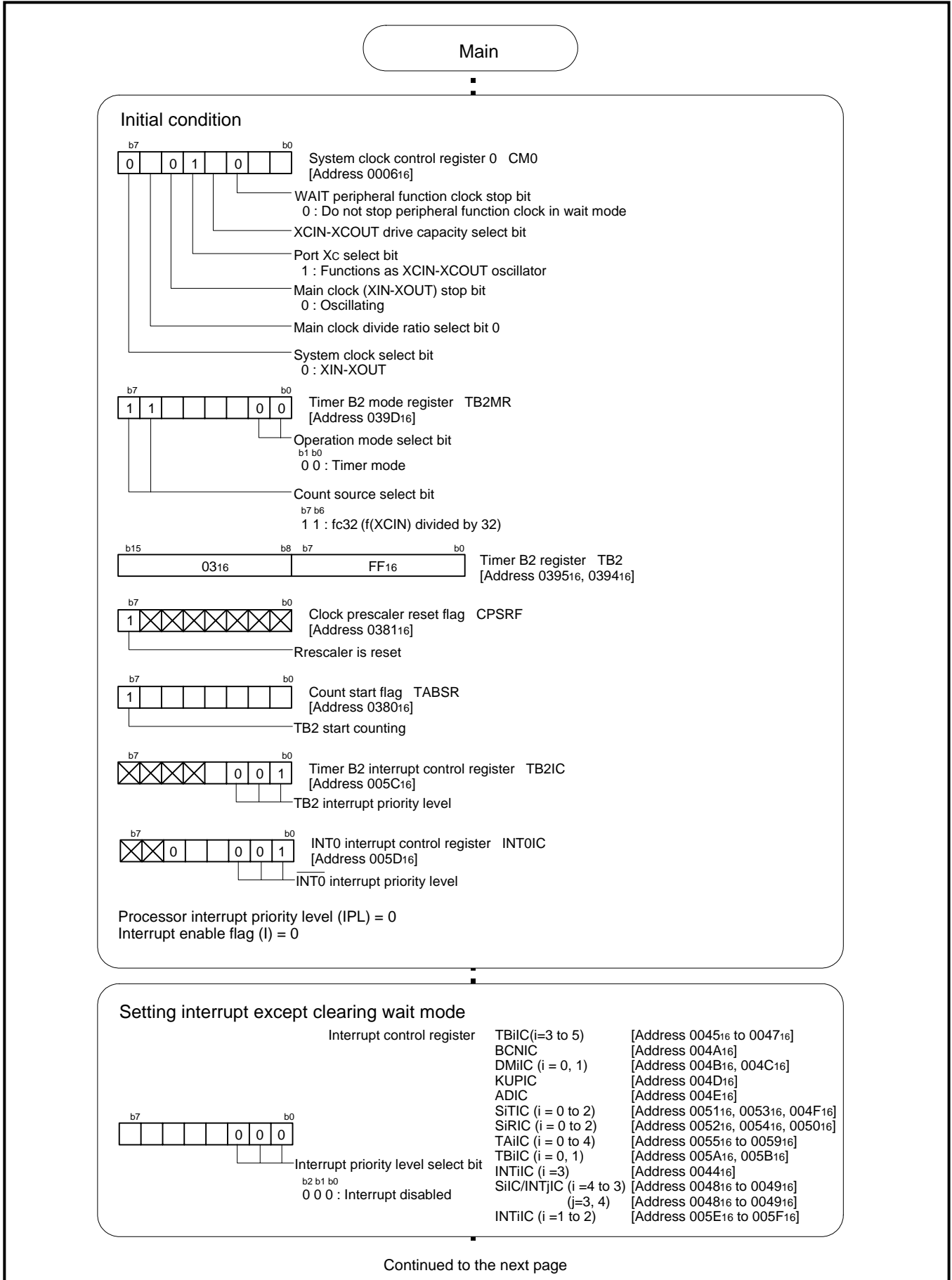**Figure 1. Operation timing of controlling power using wait mode**

Main

Initial condition

b7　　　　　　　b0
0 | | 0 | 1 | | 0 | |
System clock control register 0　CM0
[Address 0006_{16}]

WAIT peripheral function clock stop bit
　0 : Do not stop peripheral function clock in wait mode

XCIN-XCOUT drive capacity select bit

Port X_C select bit
　1 : Functions as XCIN-XCOUT oscillator

Main clock (XIN-XOUT) stop bit
　0 : Oscillating

Main clock divide ratio select bit 0

System clock select bit
　0 : XIN-XOUT

b7　　　　　　　b0
1 | 1 | | | | | 0 | 0
Timer B2 mode register　TB2MR
[Address $039D_{16}$]

Operation mode select bit
b1 b0
0 0 : Timer mode

Count source select bit
b7 b6
1 1 : fc32 (f(XCIN) divided by 32)

b15　　　　b8 b7　　　　b0
| 03_{16} | FF_{16} |
Timer B2 register　TB2
[Address 0395_{16}, 0394_{16}]

b7　　　　　　　b0
1 |XXXXXXXXX
Clock prescaler reset flag　CPSRF
[Address 0381_{16}]

Rrescaler is reset

b7　　　　　　　b0
1 | | | | | | |
Count start flag　TABSR
[Address 0380_{16}]

TB2 start counting

b7　　　　　　　b0
XXXXXX| | 0 | 0 | 1
Timer B2 interrupt control register　TB2IC
[Address $005C_{16}$]

TB2 interrupt priority level

b7　　　　　　　b0
XXX| 0 | | 0 | 0 | 1
INT0 interrupt control register　INT0IC
[Address $005D_{16}$]

INT0 interrupt priority level

Processor interrupt priority level (IPL) = 0
Interrupt enable flag (I) = 0

Setting interrupt except clearing wait mode

Interrupt control register

b7　　　　　　　b0
| | | | | 0 | 0 | 0

Interrupt priority level select bit
b2 b1 b0
0 0 0 : Interrupt disabled

TBiIC(i=3 to 5)	[Address 0045_{16} to 0047_{16}]
BCNIC	[Address $004A_{16}$]
DMiIC (i = 0, 1)	[Address $004B_{16}$, $004C_{16}$]
KUPIC	[Address $004D_{16}$]
ADIC	[Address $004E_{16}$]
SiTIC (i = 0 to 2)	[Address 0051_{16}, 0053_{16}, $004F_{16}$]
SiRIC (i = 0 to 2)	[Address 0052_{16}, 0054_{16}, 0050_{16}]
TAiIC (i = 0 to 4)	[Address 0055_{16} to 0059_{16}]
TBiIC (i = 0, 1)	[Address $005A_{16}$, $005B_{16}$]
INTiIC (i =3)	[Address 0044_{16}]
SiIC/INTjIC (i =4 to 3)	[Address 0048_{16} to 0049_{16}]
(j=3, 4)	[Address 0048_{16} to 0049_{16}]
INTiIC (i =1 to 2)	[Address $005E_{16}$ to $005F_{16}$]

Continued to the next page

Figure 2. Set-up procedure of controlling power using wait mode (1)

Continued from the previous page

Canceling protect

b7   b0

| ✕ | ✕ | ✕ | ✕ | ✕ | | | 1 |

Protect register   PRCR
[Address 000A₁₆]

Enables writing to system clock control registers 0 and 1 (address 0006₁₆ and 0007₁₆)
1 : write-enabled

Switching system clock

b7   b0

| 1 | | | | | | | |

System clock control register 0   CM0
[Address 0006₁₆]

System clock select bit
1 : XCIN-XCOUT

Stopping main clock

b7   b0

| | 1 | | | | | | |

System clock control register 0   CM0
[Address 0006₁₆]

Main clock (XIN-XOUT) stop bit
1 : Off

[F_WIT] = 1

Interrupt enable flag (I flag) ← "1"

WAIT instruction

INT0 interrupt request generated

TB2 interrupt request generated

=   [F_WIT] : 1   ≠

Starting main clock oscillator

b7   b0

| | 0 | | | | | | |

System clock control register 0   CM0
[Address 0006₁₆]

Main clock (XIN-XOUT) stop bit
0 : On

Wait until the main clock has stabilized

Switching system clock

b7   b0

| 0 | | | | | | | |

System clock control register 0   CM0
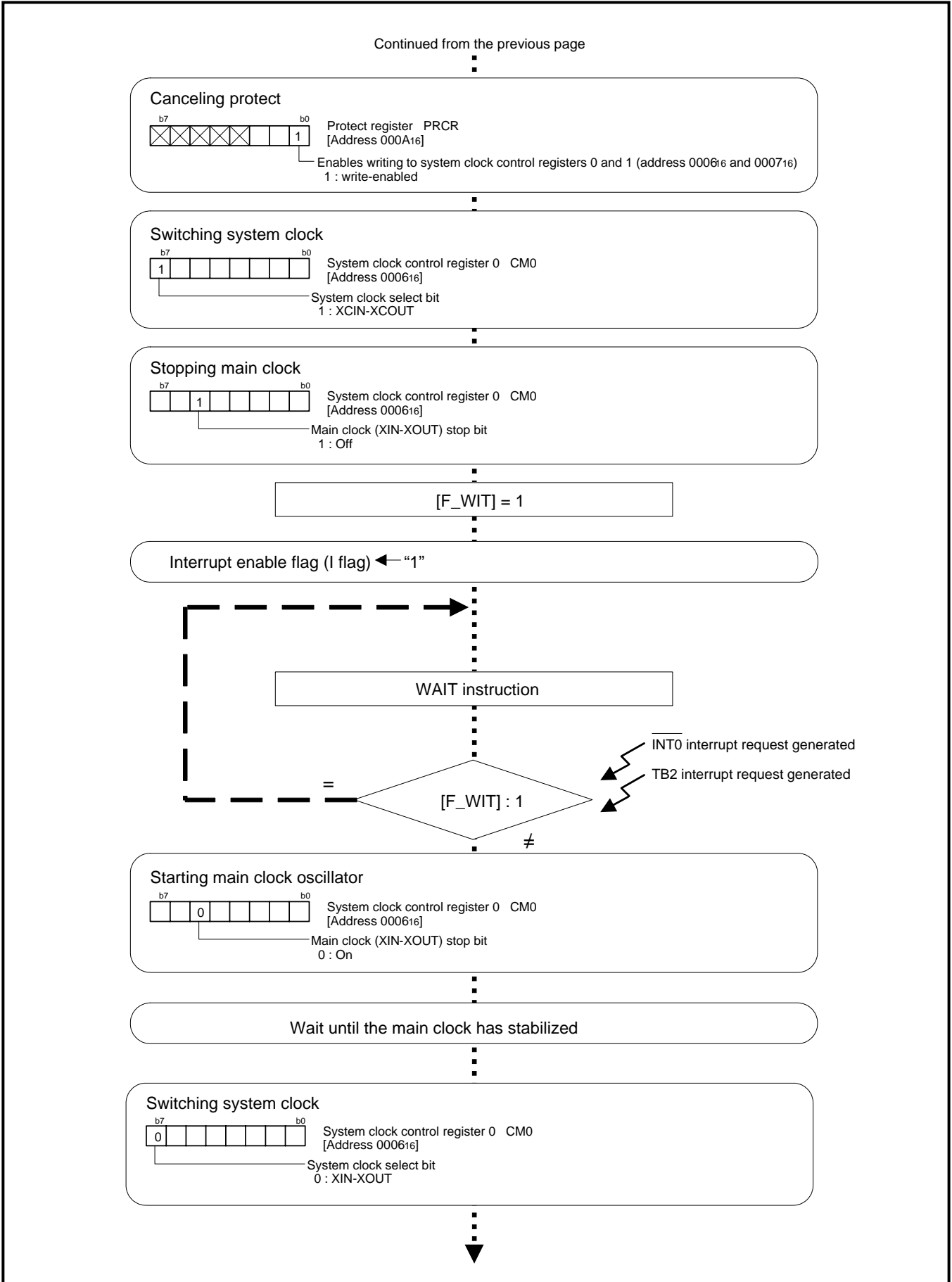[Address 0006₁₆]

System clock select bit
0 : XIN-XOUT

Figure 3. Set-up procedure of controlling power using wait mode(2)

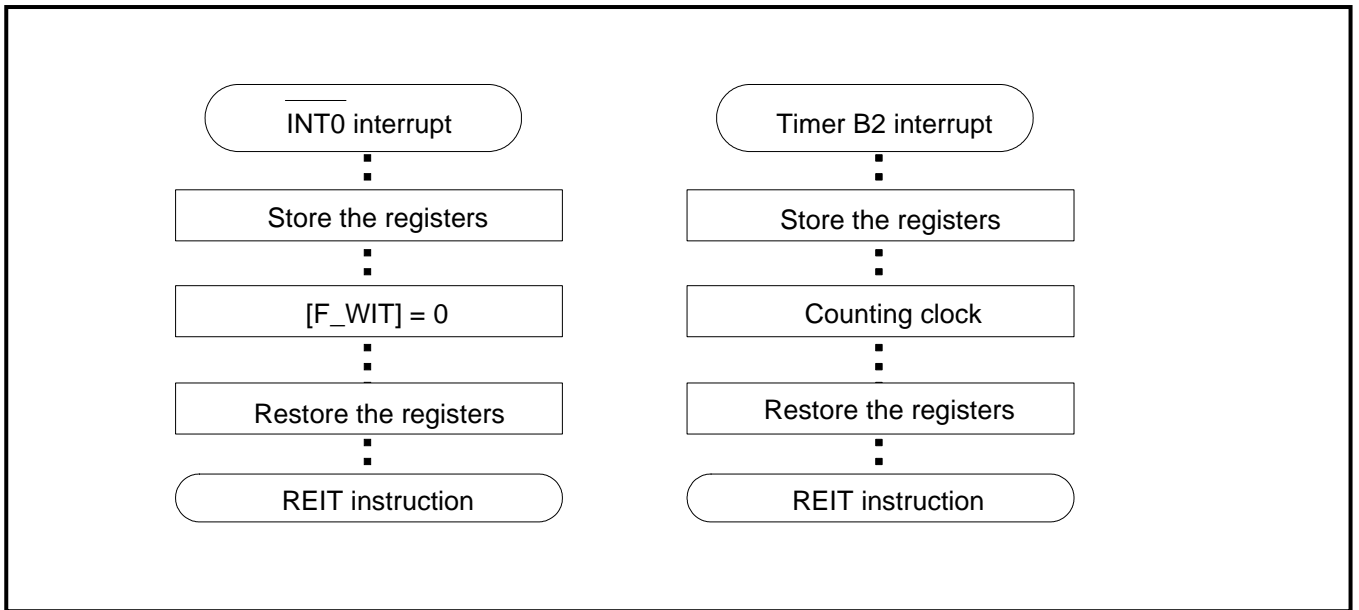**Figure 4. Set-up procedure of controlling power using wait mode (3)**

5. The example of reference program

```
;**************************************************************************************
;
;    M16C/62P Program Collection
;
;    FILE NAME : rjj05b0699_src.a30
;    CPU        : M16C/62P Group
;    FUNCTION  : Controlling Power Using Wait Mode
;    HISTORY   : 2004.12.24   Ver 1.00
;
;    Copyright(C)2004, Renesas Technology Corp.
;    Copyright(C)2004, Renesas Solutions Corp.
;    All rights reserved.
;
;**************************************************************************************
;
;**************************************************************************************
;
;         Include
;**************************************************************************************
;
;
         .LIST       off                ;Stops outputting lines to the assembler list file
         .INCLUDE   sfr62p.inc          ;Reads the file that defined SFR
         .LIST       on                 ;Starts outputting lines to the assembler list file
;
;**************************************************************************************
;
;         Symbol definition
;**************************************************************************************
;
;
RAM_TOP              .equ    00400h              ;Start address of RAM
RAM_END              .equ    013ffh              ;End address of RAM
ROM_TOP              .equ    0f4000h             ;Start address of ROM
VECT_TOP             .equ    0ffe00h             ;Start address of vect_top
FIXED_VECT_TOP       .equ    0fffdch             ;Start address of fixed_vect_top
SB_BASE              .equ    00380h              ;Base address of sb
;
;**************************************************************************************
;
;         WORKRAM area definition
;**************************************************************************************
;
;
         .section   work,data                     ;Declares section name and section type
         .org               RAM_TOP                ;Declares start address
;
workram_top:
;
FLAG:                .blkb              1          ;
WACTH_CNT:           .blkb              1          ;Count
;**************************************************************************************
;
```

```
;       Bit Symbol definition
;*******************************************************************************
;
F_WIT                .btequ          0,FLAG  ; Wait mode judgment flag
;
;*******************************************************************************
;
;       Program area
;*******************************************************************************
;
;==============================================================================
;       Start up
;==============================================================================
;
            .section  program,code          ;Declares section name and section type
            .org          ROM_TOP          ;Declares start address
            .sb           SB_BASE          ;
;
START:
            ldc           #RAM_END+1,isp ;Sets interrupt stack pointer
            ldc           #SB_BASE,sb       ;Sets sb register
;
            mov.b         #03h,prcr         ;Removes protect
                                            ;Set processor mode registers 0 and 1
            mov.w         #0800h,pm0        ;Single-chip mode
                                            ;No expansion, No wait
            mov.w         #2008h,cm0        ;Xcin-Xcout High
                                            ;Xin-Xout High, Main clock is No divison
            mov.b         #0,prcr           ;Protects all registers
;
            ldintb        #VECT_TOP         ;Sets interrupt table register
;
            mov.w         #0,r0             ;Clears WORKRAM area
            mov.w         #((RAM_END+1)-RAM_TOP)/2,r3
            mov.w         #RAM_TOP,a1
            sstr.w
;
;==============================================================================
;       Main program
;==============================================================================
MAIN:
            jsr           INIT
MAIN_A:
            btst          ir_ta0ic          ;Wait Oscillation stability(100ms)
            jnc           MAIN_A            ;
            bclr          ir_ta0ic          ;
            inc.b         r0l               ;
            cmp.b         #100,r0l          ;
            jnz           MAIN_A            ;
            mov.b         #00h,r0l          ;Clear
```

```
;
                mov.b           #00000001b,prcr   ;Removes protect
                mov.b           #10011000b,cm0    ;System clock control register 0
;                                 |    |+---------------;Xcin-Xcout(High)
;                                 |    +----------------;Xcin-Xcout generation function
;                                 +---------------------;Sub clock
;
                mov.b           #10111000b,cm0    ;System clock control register 0
;                                 |  ||+---------------;Xcin-Xcout(High)
;                                 |  |+----------------;Xcin-Xcout generation function
;                                 |  +-----------------;Main clock stop
;                                 +---------------------;Sub clock
                bset            F_WIT             ;Wait mode judgment flag
                mov.b           #10000000b,tabsr  ;Count start flag
                                                  ;Timer A0 stop
                mov.w           #500-1,ta0        ;Timer A0 register(16mhz:1ms)
                mov.b           #00000000b,ta0ic  ;Interrupt control register
;                                      +---------------;Interrupt request bit (0:interrupt not requested)
                fset            i                 ;Set interrupt enable flag
;
MAIN_B:
;
                jmp.b           MAIN_C            ;TN-16C-128A/JA(Japanese) countermeasure
;                                                 ;TN-16C-128A/EA(English) countermeasure
MAIN_C:

                wait                              ;Wait mode
;
                nop
                nop
                nop
                nop
;
                btst            F_WIT             ;Judge flag
                jc              MAIN_B
                mov.b           #10011000b,cm0    ;System clock control register 0
;                                 |  ||+---------------;Xcin-Xcout(High)
;                                 |  |+----------------;Xcin-Xcout generation function
;                                 |  +-----------------;Main clock start
;                                 +---------------------;Sub clock
;
                mov.b           #10000001b,tabsr  ;Count start flag
                                                  ;Timer A0 start
MAIN_D:

                btst            ir_ta0ic          ;Wait oscillation stability(1ms)
                jnc             MAIN_D            ;
```

```
                bclr                    ir_ta0ic            ;
                mov.b                   #00011000b,cm0    ;System clock control register 0
;                                         | | |+---------------;Xcin-Xcout(High)
;                                         | |+-----------------;Xcin-Xcout generation function
;                                         |   +------------------;Main clock start
;                                         +----------------------;Main clock
;
                mov.b                   #00000000b,prcr   ;Protects all registers
                jmp                     MAIN_A
;
;
;================================================================================
;         Initialize program
;================================================================================
INIT:
                mov.b                   #00000001b,prcr  ;Removes protect
                mov.b                   #00011000b,cm0   ;System clock control register 0
;                                          |+---------------;Xcin-Xcout(High)
;                                          +----------------;Xcin-Xcout generation function
                mov.b                   #00000000b,prcr  ;Protects all registers
                mov.b                   #11000000b,tb2mr;Timer B2 mode register
;                                         ++-------------------;Count source select bit(fc32)
                mov.w                   #03ffh,tb2          ;Timer B2 register
                mov.b                   #10000000b,cpsrf  ;Clock prescaler reset flag
                mov.b                   #00000001b,tb2ic  ;Interrupt control register
;                                            |+++----------;Interrupt priority level select bit
;                                            |             ;(001:Level 1, interrupt disabled)
;                                            +--------------;Interrupt request bit (0:interrupt not requested)
                mov.b                   #00000001b,int0ic ;Interrupt control register
;                                            |+++----------;Interrupt priority level select bit
;                                            |             ;(001:Level 1, interrupt disabled)
;                                            +--------------;Interrupt request bit (0:interrupt not requested)
;
                mov.b                   #10000000b,ta0mr;Timer A0 mode register
;                                         ++-------------------;Count source select bit(f32)
                mov.w                   #500-1,ta0          ;Timer A0 register(16mhz:1ms)
                mov.b                   #00000000b,ta0ic  ;Interrupt control register
;                                            |+++----------;Interrupt priority level select bit
;                                            |             ;(000:Level 0, interrupt disabled)
;                                            +--------------;Interrupt request bit (0:interrupt not requested)
                mov.b                   #00000001b,tabsr ;Count start flag
INIT_END:
                rts
;================================================================================
;         Interrupt program
;================================================================================
INT0_INT:
```

```
                pushm           r0,r1,r2,r3,a0,a1    ;Take shelter register
;
                bclr            F_WIT                ;Judgment flag clear
;
                popm            r0,r1,r2,r3,a0,a1    ;Comeback register
                reit
;
;=============================================================================
;        Interrupt program
;=============================================================================
TB2_INT:
                pushm           r0,r1,r2,r3,a0,a1    ;Take shelter register
;
                inc.b           WACTH_CNT           ;Count
;
                popm            r0,r1,r2,r3,a0,a1    ;Comeback register
                reit
;
;=============================================================================
;        Dummy interrupt processing program
;=============================================================================
DUMMY:
                reit
;
;*******************************************************************************
;        Setting of variable vector table
;*******************************************************************************
;
;
                .section   vect,romdata
                .org             VECT_TOP + (4 * 4)
;
                .lword          DUMMY                ;INT3 interrupt vector
                .lword          DUMMY                ;TB5 interrupt vector
                .lword          DUMMY                ;TB4 interrupt vector
                                                     ;UART1 bus collision detection interrupt vector
                .lword          DUMMY                ;TB3 interrupt vector
                                                     ;UART0 bus collision detection interrupt vector
                .lword          DUMMY                ;SI/04/INT5 interrupt vector
                .lword          DUMMY                ;SI/03/INT4 interrupt vector
                .lword          DUMMY                ;UART2 bus collision detection interrupt vector
                .lword          DUMMY                ;DMA0 interrupt vector
                .lword          DUMMY                ;DMA1 interrupt vector
                .lword          DUMMY                ;KEY interrupt vector
                .lword          DUMMY                ;A-D interrupt vector
                .lword          DUMMY                ;UART2 transmit/NACK interrupt vector
                .lword          DUMMY                ;UART2 receive/ACK interrupt vector
```

```
                .lword          DUMMY           ;UART0 transmit/NACK interrupt vector
                .lword          DUMMY           ;UART0 receive/ACK interrupt vector
                .lword          DUMMY           ;UART1 transmit/NACK interrupt vector
                .lword          DUMMY           ;UART1 receive/ACK interrupt vector
                .lword          DUMMY           ;TA0 interrupt vector
                .lword          DUMMY           ;TA1 interrupt vector
                .lword          DUMMY           ;TA2 interrupt vector
                .lword          DUMMY           ;TA3 interrupt vector
                .lword          DUMMY           ;TA4 interrupt vector
                .lword          DUMMY           ;TB0 interrupt vector
                .lword          DUMMY           ;TB1 interrupt vector
                .lword          TB2_INT         ;TB2 interrupt vector
                .lword          INT0_INT        ;INT0 interrupt vector
                .lword          DUMMY           ;INT1 interrupt vector
                .lword          DUMMY           ;INT2 interrupt vector
;
.********************************************************************************
;
;         Setting of fixed vector
.********************************************************************************
;
;
                .section   f_vect,romdata
                .org            FIXED_VECT_TOP
;
                .lword          DUMMY           ;Undefined instruction interrupt vector
                .lword          DUMMY           ;Overflow (INTO instruction) interrupt vector
                .lword          DUMMY           ;BRK instruction interrupt vector
                .lword          DUMMY           ;Address match interrupt vector
                .lword          DUMMY           ;Single-step interrupt vector
                .lword          DUMMY           ;Watchdog timer interrupt vector
                                                ;Oscillation stop and Re-oscillation detection interrupt
                                                ;vector
                                                ;Voltage down detection interrupt vector
                .lword          DUMMY           ;DBC interrupt vector
                .lword          DUMMY           ;NMI interrupt vector
                .lword          START           ;Sets start vector
;
                .end
```

6.   Referense

   Hardware manual
      M16C/62P group (M16C/62P,M16C/62PT) Hardware Manual Rev.2.30
      (Use the latest version on the web-site: http://www.renesas.com)

7.   Web-site and contact for support

   Renesas web-site
      http://www.renesas.com/

   Contact for Renesas technical support
      E-mail: support_apl@renesas.com

Revision

Rev.	Issue data	Revised	
		Page	Point
1.00	2004.12	-	First edition issued