

致尊敬的顾客

关于产品目录等资料中的旧公司名称

NEC电子公司与株式会社瑞萨科技于2010年4月1日进行业务整合（合并），整合后的新公司暨“瑞萨电子公司”继承两家公司的所有业务。因此，本资料中虽还保留有旧公司名称等标识，但是并不妨碍本资料的有效性，敬请谅解。

瑞萨电子公司网址：<http://www.renesas.com>

2010年4月1日
瑞萨电子公司

【发行】瑞萨电子公司（<http://www.renesas.com>）

【业务咨询】<http://www.renesas.com/inquiry>

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/Tiny 系列

看门狗定时器的使用

1. 要点

本篇资料介绍如何使用 M16C/Tiny 系列单片机的看门狗定时器。

2. 说明

本篇资料，适用于 M16C/26A、M16C/28、M16C/29 群单片机。

本篇资料中的参考例程也适用于 M16C 族产品中与 M16C/26A、M16C/28、M16C/29 群具有相同 SFR（特殊功能寄存器）定义的产品。

由于 M16C 系列产品中有些功能会有所改进，请参看用户手册。如果在使用本篇资料中所列功能时，请仔细检查每个操作。

3. 规格

3.1 操作

- (1) 通过写看门狗定时器的启动寄存器，初始化看门狗定时器的值为“7FFFh”，并启动看门狗递减计数。
- (2) 看门狗定时器计数的过程中，如果重写看门狗定时器的启动寄存器，定时值会再次初始化为“7FFFh”，并重新开始递减计数。
- (3) 在执行等待命令或进入停止模式时，看门狗定时器将停止计数并保持当前计数值。解除等待或从停止模式返回时，看门狗定时器则从保持的计数值开始继续递减计数。
- (4) 如果看门狗定时器计数下溢，定时值会重新初始化为“7FFFh”并继续开始递减计数，同时产生看门狗定时器中断。

注意事项：

- 看门狗定时器和预分频器在复位后都是停止的，故需要通过写看门狗启动定时器从而启动看门狗定时器开始计数。写看门狗启动定时器的周期要保证比看门狗定时器的周期要短，以避免产生看门狗定时器中断。另外，在看门狗定时器的中断程序的开始也需要设置看门狗启动定时器。
- 如果处理器模式寄存器 1 (PM1) 中的看门狗定时器功能选择位置为“1”（看门狗定时器复位），当看门狗定时器中断产生时，引脚、CPU 和 SFR 都将被初始化，而程序也从复位向量地址处开始执行。

看门狗定时器的工作时序图如下所示：

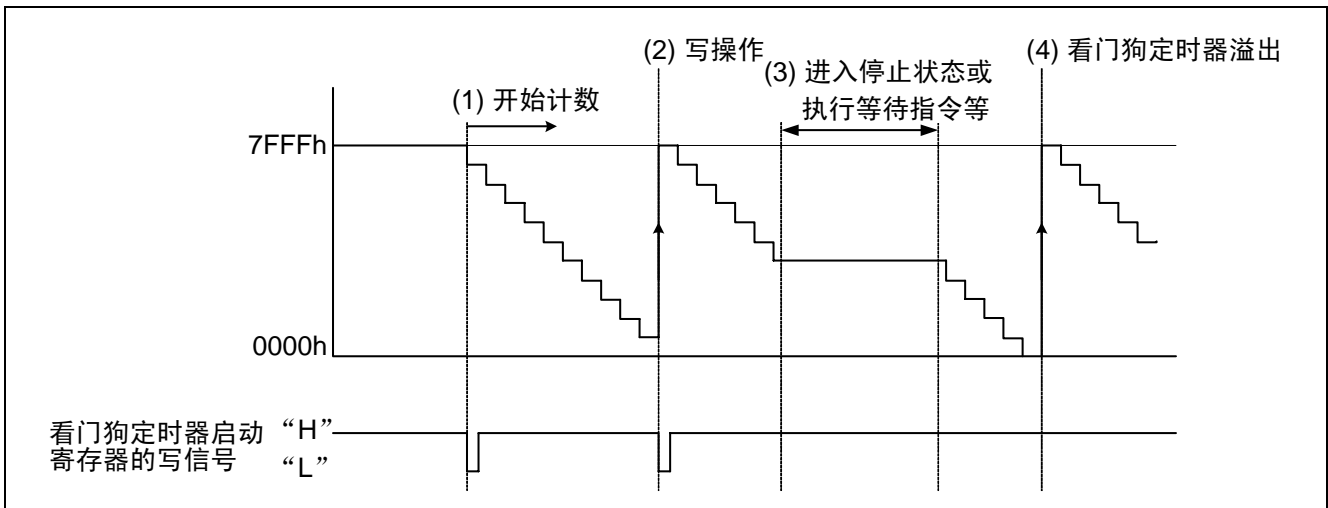
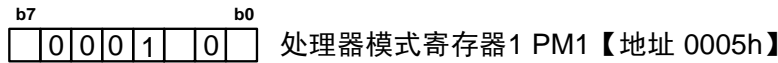


图 1. 看门狗定时器工作时序图

3.2 寄存器设置

本篇资料中给出的操作是基于 M16C/29 群单片机产品，其它 M16C/Tiny 系列产品的寄存器设置步骤请参阅相关硬件手册。

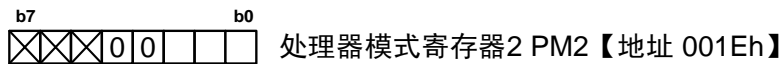
(1) 设置看门狗定时器功能选择位（注1）



<PM12> 看门狗定时器功能选择位
0: 看门狗定时器中断
1: 看门狗定时器复位(注2)

注1: 必须在将PRCR寄存器中的PRC1位置为“1”（允许写）后改写此该寄存器。
注2: 通过程序将PM12位置为“1”。（即使写“0”也不变化）

(2) 设置看门狗定时器计数源保护位



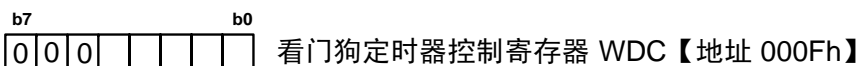
<PM22> 看门狗定时器计数源保护位（注1，注2）
0: 看门狗定时器的计数源为CPU时钟
1: 看门狗定时器的计数源为内部振荡器时钟

注1: 此位一旦被设置为“1”后，不能通过程序清“0”。

注2: 将PM22位设置为“1”，将产生以下结果：

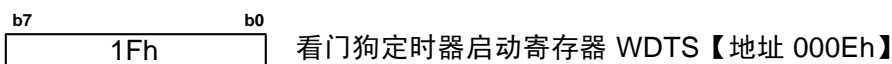
- 内部振荡器开始振荡，并且内部振荡器时钟为看门狗定时器的计数源。
- 禁止写CM1寄存器的CM10位。（即使写“1”也不变化，不能进入停止模式。）
- 在等待模式时，看门狗定时器不停止。

(3) 设置看门狗定时器控制寄存器



看门狗定时器的高位
保留位，必须置为“0”。
预分频器选择位
0: 16分频
1: 128分频

(4) 设置看门狗定时器启动寄存器



当对该寄存器进行写操作后，看门狗定时器被初始化并开始计数。
看门狗定时器不管写入何值，其计数值都初始化为“7FFFh”。

4. 参考例程

参考例程的硬件连接如图 2 所示。

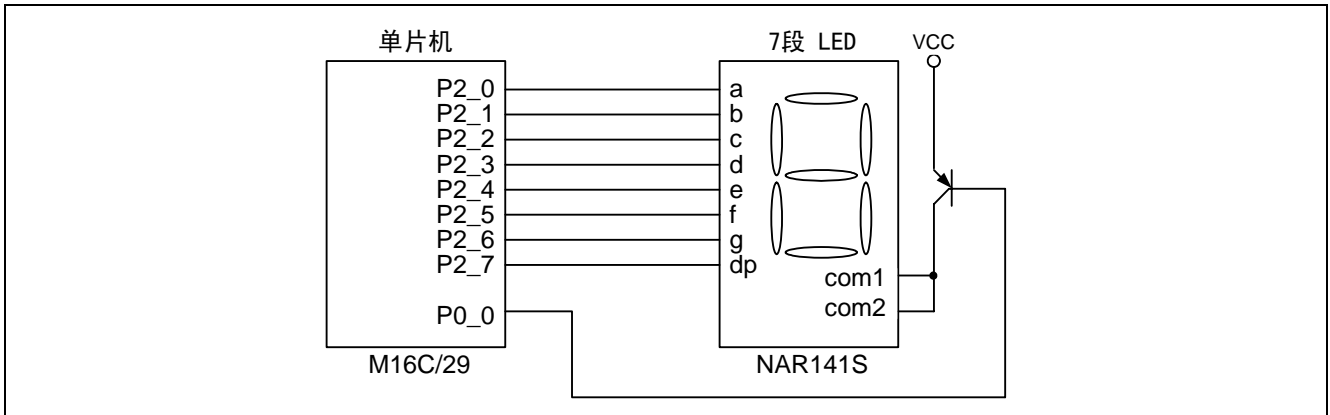


图 2 参考例程的硬件连接说明图

4.1 使用看门狗定时器中断程序

以下程序通过端口 P2 的递增来显示看门狗定时器的写入次数，当端口 P2 的值增加到“9”时，程序不再向看门狗定时器的启动定时器写入数据，并保持端口 P2 的当前显示值。

当看门狗定时器中断产生时，中断处理程序中通过端口 P2 的递减来显示看门狗定时器的写入次数，当端口 P2 的值递减到“0”时，保持端口 P2 的当前显示值。

```

/*****/
/* M16C/Tiny Series Program Collection */
/* */
/* FILE NAME : rec05b0006-0102_int.c */
/* CPU : M16C/29 Group */
/* FUNCTION : Operation of Watchdog Timer */
/* HISTORY : 2006.04.13 Ver 1.02 */
/* */
/* Copyright (C) 2005. Renesas Technology Corp. */
/* All right reserved. */
/*****/

/*****/
/* Include File */
/*****/
#include "sfr29.h" // Special function register header file

/*****/
/* Function Definition */
/*****/
void init_mcu(void); // SFR initialize
void wait_10ms(void); // Main clock oscillation stable wait routine
#pragma INTERRUPT wdt_int

/*****/
/* Define Label */
/*****/
#define PRODUCT_TYPE 0 // 28,29 group: 0 26A group: 1
#define PIN_TYPE 0 // 80 pin: 0 64 pin: 1 (28,29 group)
// 48 pin: 0 42 pin: 1 (26A group)

```

```

/*****
/*   Define Const                               */
/*****
/* port2_0 - port2_7 data : 0 1 2 3 4 5 6 7 8 9 */
unsigned char count_data[10] =
{0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xD8,0x80,0x90};

unsigned int i,j;

/*****
/*   Main Program                               */
/*****
void main(void) {
    init_mcu();                                // SFR initialize

    j = 0; i = 0;

    wdc = 0;                                   // Setting watchdog timer control register
                                           // Prescaler select bit is set to 0 (0: divided by 16)

    wdts = 1;                                  // Setting watchdog timer start register

    while (1) {
        while (ir_ta0ic == 0); // lms?

        ta0ic = 0x00;                          // Timer A0 interrupt level: 0
        i++;

        if ( i == 500 ) {
            i = 0;
            if ( j <= 9 ) {
                p2 = count_data[j];
            }
            j++;
        }

        if ( j <= 9 ) {
            wdts = 1;                            // Setting watchdog timer start register
        }
    }
}

/*****
/*   Watchdog Timer Interrupt Routine           */
/*****
void wdt_int() {
    wdts = 1;                                   // Set the WDTS register in the beginning of the
                                           // watchdog timer interrupt routine.

    while (1) {
        wdts = 1;                               // For long-time loop, set WDTS register again to
                                           // avoid the underflow of watchdog timer.

        while (ir_ta0ic == 0); // lms?

        ta0ic = 0x00;                          // Timer A0 interrupt level: 0
        i++;
    }
}

```

```

    if ( i == 500 ) {
        i = 0;
        if ( j != 0 ) {
            j--;
            if ( j <= 9 ) {
                p2 = count_data[j];
            }
        }
    }
}

/*****
/*    MCU Initialize    */
*****/
void init_mcu(void) {
    prcr = 0x03;        // Protect register off

                        // Set processor mode registers
    pm0 = 0x00;        // Single-chip mode
    pm1 = 0x08;        // No expansion, No wait

    wait_10ms();      // Waiting for main clock oscillation stable

                        // Set system clock control registers
    cm2 = 0x00;        // System clock select Main clock or PLL clock
    cm1 = 0x20;        // Xin-Xout High, Main clock is no division
    cm0 = 0x08;        // Xcin-Xcout drive capacity select bit (1: high)

    pclkr = 0x03;     // TimerA, B clock select bit (1: f1)

                        // WDT clock setting
    pm22 = 0;         // Set WDT count source protective bit
                        // <PM22> : select CPU clock
                        // 0: select CPU clock

                        // Set WDT function select bit
    pm12 = 0;         // <PM12> :0: WDT interrupt; 1: WDT reset
                        // (writing a "1" by program, writing a "0" has no effect)

    prcr = 0x00;     // Protect register on

    #if PRODUCT_TYPE // Product selection: 26A group
        ifsr2a = 1;  // Interrupt request cause select register2 IFSR2A
                        // <IFSR20> : Reserved bit (Must be set to "1")
        prcr = 0x04; // Protect register off
        #if PIN_TYPE // Port setting
            pacr = 0x01; // 42pin type
        #else
            pacr = 0x04; // 48pin type
        #endif
        prcr = 0x00; // Protect register on
    #else // Product selection: 28,29 group
        ifsr2a = 0;  // Interrupt request cause select register2 IFSR2A
                        // <IFSR20> : Reserved bit (Must be set to "0")
        prcr = 0x04; // Protect register off
    #endif
}

```



```

    #if PIN_TYPE          // Port setting
        pacr = 0x02;      // 64pin type
    #else
        pacr = 0x03;      // 80pin type
    #endif
    prcr = 0x00;          // Protect register on
#endif

p0 = 0x02;               // Select LED1
p2 = 0;                  // Port2 output
pd0 = 0xff;              // Port direction0: output mode
pd2 = 0xff;              // Port direction2: output mode

                          // Timer A0 setup
ta0mr = 0x40;           // Selection of timer mode
                          // Pulse output function select bit (0: pulse is not output)
                          // Gate function select bit (00: gate function not available)
                          // Count source (01: f8)
ta0 = 2500-1;           // Setting counter value (1msec @20MHz, f8)
ta0ic = 0x00;           // Setting interrupt priority levels in timer A0
ta0s = 1;               // Timer A0 count start
}

/*****
/*   Main Clock Oscillation Stable Wait 10ms Routine   */
*****/
void wait_10ms(void) {
    ta0mr = 0x00;        // Set Timer A0 mode register (Timer mode, count source: f1)

    ta0 = 20000-1;       // Setting counter value (10msec @4MHz/2, f1)

    ta0ic = 0x00;        // Clear interrupt request bit

    tabsr = 0x01;        // Timer A0 start counting

    while (ir_ta0ic == 0){    }

    ir_ta0ic = 0;        // Clear interrupt request bit

    tabsr = 0x00;        // Timer A0 stops counting
}

```

如下所示，为使程序正常运行，需定义看门狗定时器的中断向量地址，使之指向中断服务程序。必须在启动文件“sect30.inc”的中断向量表中，定义看门狗定时器的中断程序地址“_wdt_int”。

```

;/*****
; C Compiler for R8C/Tiny, M16C/60, 30, 20, 10
; Copyright(C) 1999(2000-2004). Renesas Technology Corp.
; and Renesas Solutions Corp., All rights reserved.
;
; Written by T.Aoyama
;
; sect30.inc      : section definition
; This program is applicable when using the basic I/O library
;
; $Id: sect30.inc,v 1.23.4.6 2004/10/29 14:06:39 simomura Exp $
;
;*****/

;=====
; fixed vector section
;-----
    .section fvector,ROMDATA
    .org 0ffffdch
    .glb _wdt_int
;UDI:
; .lword  dummy_int
;OVER_FLOW:
; .lword  dummy_int
;BRKI:
; .lword  dummy_int
;ADDRESS_MATCH:
; .lword  dummy_int
;SINGLE_STEP:
; .lword  dummy_int
    .org 0ffff0H
WDT:
    .lword  _wdt_int
;DBC:
; .lword  dummy_int
;NMI:
; .lword  dummy_int
    .org 0ffffcH
RESET:
    .lword  start
;*****/

```

4.2 使用看门狗定时器中断复位程序

以下程序通过端口 P2 的递增来显示看门狗定时器的写入次数，当端口 P2 的值增加到“9”时，程序不再向看门狗定时器的启动定时器写入数据，并保持端口 P2 的当前显示值。

当看门狗定时器计数值下溢后，程序复位并且重新开始执行。

```

/*****/
/*
/* M16C/Tiny Series Program Collection
/*
/* FILE NAME : rec05b0006-0102_rst.c
/* CPU : M16C/29 Group
/* FUNCTION : Operation of Watchdog Timer
/* HISTORY : 2006.04.13 Ver 1.02
/*
/* Copyright (C) 2005. Renesas Technology Corp.
/* All right reserved.
/*
/*****/

/*****/
/* Include File
/*****/
#include "sfr29.h" // Special function register header file

/*****/
/* Function Definition
/*****/
void init_mcu(void); // SFR initialize
void wait_10ms(void); // Main clock oscillation stable wait routine

/*****/
/* Define Label
/*****/
#define PRODUCT_TYPE 0 // 28,29 group: 0 26A group: 1
#define PIN_TYPE 0 // 80 pin: 0 64 pin: 1 (28,29 group)
// 48 pin: 0 42 pin: 1 (26A group)

/*****/
/* Define Const
/*****/
/* port2_0 - port2_7 data : 0 1 2 3 4 5 6 7 8 9 */
unsigned char count_data[10] =
{0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xD8,0x80,0x90};

unsigned int i,j;

/*****/
/* Main Program
/*****/
void main(void) {
    init_mcu(); // SFR initialize

    j = 0; i = 0;

    wdc = 0; // Setting watchdog timer control register
            // Prescaler select bit is set to 0 (0: divided by 16)

```

```

/* wdt reset */
wdts = 1;           // Setting watchdog timer start register

while (1) {
    while (ir_ta0ic == 0); // 1ms?

    ta0ic = 0x00;     // Timer A0 interrupt level: 0
    i++;

    if ( i == 500 ) {
        i = 0;
        if ( j <= 9 ) {
            p2 = count_data[j];
        }
        j++;
    }

    if ( j <= 9 ) {
        wdts = 1;     // Setting watchdog timer start register
    }
}

}

/*****
/*   MCU Initialize
*****/
void init_mcu(void) {
    prcr = 0x03;     // Protect register off

                    // Set processor mode registers
    pm0 = 0x00;     // Single-chip mode
    pm1 = 0x08;     // No expansion, No wait

    wait_10ms();   // Waiting for main clock oscillation stable

                    // Set system clock control registers
    cm2 = 0x00;     // System clock select Main clock or PLL clock
    cm1 = 0x20;     // Xin-Xout High, Main clock is no division
    cm0 = 0x08;     // Xcin-Xcout drive capacity select bit (1: high)

    pclkr = 0x03;  // TimerA, B clock select bit (1: f1)

                    // WDT clock setting
    pm22 = 0;      // Set WDT count source protective bit
                    // <PM22> : select CPU clock
                    // 0: select CPU clock

                    // Set WDT function select bit
    pm12 = 1;      // <PM12> :0: WDT interrupt; 1: WDT reset
                    // (writing a 1 by program, writing a 0 has no effect)

    prcr = 0x00;  // Protect register on

    #if PRODUCT_TYPE // Product selection: 26A group
        ifsr2a = 1;  // Interrupt request cause select register2 IFSR2A
                    // <IFSR20> : Reserved bit (Must be set to "1")
        prcr = 0x04; // Protect register off
    #endif
}

```

```

    #if PIN_TYPE          // Port setting
        pacr = 0x01;      // 42pin type
    #else
        pacr = 0x04;      // 48pin type
    #endif
    prcr = 0x00;          // Protect register on
#else
    // Product selection: 28,29 group
    ifsr2a = 0;           // Interrupt request cause select register2 IFSR2A
    // <IFSR20> : Reserved bit (Must be set to "0")

    prcr = 0x04;          // Protect register off
    #if PIN_TYPE          // Port setting
        pacr = 0x02;      // 64pin type
    #else
        pacr = 0x03;      // 80pin type
    #endif
    prcr = 0x00;          // Protect register on
#endif

p0 = 0x02;               // Select LED1
p2 = 0;                  // Port2 output
pd0 = 0xff;              // Port direction0: output mode
pd2 = 0xff;              // Port direction2: output mode

// Timer A0 setup
ta0mr = 0x40;            // Selection of timer mode
// Pulse output function select bit (0: pulse is not output)
// Gate function select bit (00: gate function not available)
// Count source (01: f8)
ta0 = 2500-1;           // Setting counter value (1msec @20MHz, f8)
ta0ic = 0x00;           // Setting interrupt priority levels in timer A0
ta0s = 1;                // Timer A0 count start
}

/*****
/*   Main Clock Oscillation Stable Wait 10ms Routine   */
*****/
void wait_10ms(void) {
    ta0mr = 0x00;        // Set Timer A0 mode register (Timer mode, count source: f1)

    ta0 = 20000-1;       // Setting counter value (10msec @4MHz/2, f1)

    ta0ic = 0x00;        // Clear interrupt request bit

    tabsr = 0x01;        // Timer A0 start counting

    while (ir_ta0ic == 0){ }

    ir_ta0ic = 0;        // Clear interrupt request bit

    tabsr = 0x00;        // Timer A0 stops counting
}

```

5. 参考文献

数据手册

M16C/26A 群 (M16C/26A、M16C/26T) 硬件手册 Rev.1.00

M16C/28 群硬件手册 Rev.1.00

M16C/29 群硬件手册 Rev.1.00

(最新版本请从瑞萨科技网页上取得)

技术信息/技术更新

(最新信息请从瑞萨科技网页上取得)

公司主页和咨询窗口

瑞萨科技公司主页

<http://www.cn.renesas.com>

咨询

<http://www.renesas.com/inquiry>

修订记录

Rev.	发行日	修订内容	
		页	要点
1.00	2006.04.14	—	初版发行

Keep safety first in your circuit designs!

1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (<http://www.renesas.com>).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.

注意

本文只是参考译文，前页所载英文具有正式效力。

请遵循安全第一进行电路设计

1. 虽然瑞萨科技尽力提高半导体产品的质量和可靠性，但是半导体产品也可能发生故障。半导体的故障可能导致人身伤害、火灾事故以及财产损害。在电路设计时，请充分考虑安全性，采用合适的如冗余设计、利用非易燃材料以及故障或者事故防止等的安全设计方法。

关于利用本资料时的注意事项

1. 本资料是为了让用户根据用途选择合适的瑞萨科技产品的参考资料，不转让属于瑞萨科技或者第三者所有的知识产权和其它权利的许可。
2. 对于因使用本资料所记载的产品数据、图、表、程序、算法以及其它应用电路的例子而引起的损害或者对第三者的权力的侵犯，瑞萨科技不承担责任。
3. 本资料所记载的产品数据、图、表、程序、算法以及其它所有信息均为本资料发行时的信息，由于改进产品或者其它原因，本资料记载的信息可能变动，恕不另行通知。在购买本资料所记载的产品时，请预先向瑞萨科技或者经授权的瑞萨科技产品经销商确认最新信息。
本资料所记载的信息可能存在技术不准确或者印刷错误。因这些错误而引起的损害、责任问题或者其它损失，瑞萨科技不承担责任。
同时也请通过各种方式注意瑞萨科技公布的信息，包括瑞萨科技半导体网站。
(<http://www.renesas.com>)
4. 在使用本资料所记载部分或者全部数据、图、表、程序以及算法等信息时，在最终做出有关信息和产品是否适用的判断前，务必对作为整个系统的所有信息进行评价。由于本资料所记载的信息而引起的损害、责任问题或者其它损失，瑞萨科技不承担责任。
5. 瑞萨科技的半导体产品不是为在可能和人命相关的环境下使用的设备或者系统而设计和制造的产品。在研讨将本资料所记载的产品用于运输、机动车辆、医疗、航空宇宙用、原子能控制、海底中继器的设备或者系统等特殊用途时，请与瑞萨科技或者经授权的瑞萨产品经销商联系。
6. 未经瑞萨科技的书面许可，不得翻印或者复制全部或者部分资料的内容。
7. 如果本资料所记载的某产品或者技术内容受日本出口管理限制，必须在得到日本政府的有关部门许可后才能出口，并且不准进口到批准目的地国家以外的国家。
禁止违反日本和（或者）目的地国家的出口管理法和法规的任何转卖、挪用或者再出口。
8. 如果需要了解本资料所记载的信息或者产品的详细，请与瑞萨科技联系。