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.

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.

H8/300L SLP Series

Using an External Interrupt to Start Counter Incrementation

Introduction

Incrementation of the 16-bit counter is started by an IRQ0 interrupt which is generated when the input signal from a switch on the $\overline{\text{IRQ0}}$ pin is turned on.

Target Device

H8/38024

Contents

1. Specifications 2
2. Description of Functions 2
3. Principle of Operation 4
4. Description of Software 5
5. Flowchart 6
6. Program Listing 7

1. Specifications

1. Incrementation of the 16-bit counter is started by an IRQ0 interrupt which is generated when the input signal from a switch on the $\overline{\text{IRQ0}}$ pin is turned on.
2. An IRQ0 interrupt is requested upon detection of a falling edge of the $\overline{\text{IRQ0}}$ pin input.
3. The LED is turned on and off each time the 16-bit counter set in counter_sub overflows.
4. The LED is connected to the P92 output pin of port 9.
5. The P92 pin is a large-current port.
6. Figure 1.1 shows an example of connecting a switch to the $\overline{\text{IRQ0}}$ input pin.

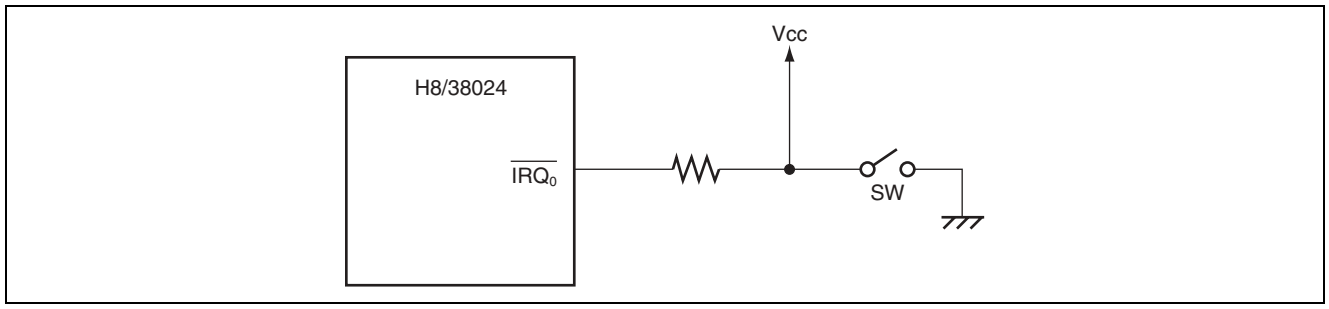


Figure 1.1 Example of Connecting a Switch to $\overline{\text{IRQ0}}$ Input Pin

2. Description of Functions

1. In this sample task, counting is started by an IRQ0 external interrupt.
   A. The external interrupts are described below.
   - There are 13 interrupt sources for external interrupts: WKP7 to WKP0 interrupts, and IRQ4, IRQ3, IRQAEC, IRQ1, and IRQ0 interrupts.
   - WKP7 to WKP0 interrupts are requested on detection of a rising or falling edge on the WKP7 to WKP0 pins depending on the edge sense setting. The edge sense is specified by the wakeup edge select register (WEGR).
   - Pins WKP7 to WKP0 are also used as I/O pins for port 5. To use these pins as WKP7 to WKP0 input pins, set the WKP7 to WKP0 bits in port mode register 5 (PMR5) to 1.
   - When pins are selected to function as WKP7 to WKP0 pin by PMR5 and a specified signal edge is input, the corresponding bit in the wakeup interrupt flag register (IWPR) is set to 1 and an interrupt request is issued.
   - Acceptance of interrupt requests can be disabled by clearing the IENWP bit in interrupt enable register 1 (IENR1) to 0.
   - IRQ4, IRQ3, IRQ1 and IRQ0 interrupts are requested by input signals on pins $\overline{\text{IRQ4}}$, $\overline{\text{IRQ3}}$, $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ0}}$. The edge sense, rising or falling edge, for these four interrupts can be set individually by the settings of the IEG4, IEG3, IEG1 and IEG0 bits in the interrupt edge select register (IEGR).
   - When pins are selected to function as $\overline{\text{IRQ4}}$, $\overline{\text{IRQ3}}$, $\overline{\text{IRQ1}}$ and $\overline{\text{IRQ0}}$ pins by the setting of port mode registers (PMRB, PMR2 and PMR1) and a specified signal edge is input, the corresponding bit in interrupt request register 1 (IRR1) is set to 1 and an interrupt request is issued.
   - Acceptance of interrupt requests can be disabled by clearing the IEN4, IEN3, IEN1 and IEN0 bits in interrupt enable register 1 (IENR1) to 0.
   - A IRQAEC interrupt is requested by the input signal on the IRQAEC pin or IECPWM.
   - When the IRQAEC pin is used for an external interrupt signal, the event counter PWM enable bit (ECPWME) in the input pin edge select register (AEGSR) should be set to 0.
   - The edge sense for the IRQAEC interrupt can be specified for rising, falling, or both edges by the setting of the input pin edge select register (AEGSR).

- When IRQAEC interrupt enable (IENEC2) in interrupt enable register 1(IENR1) is set to 1, and the specified signal edge is input, the IRQAEC interrupt request flag (IRREC2) in the interrupt request register 1 (IRR1) is set to 1 and an interrupt request is issued.
- All interrupts can be masked by setting the I bit of the condition code register (CCR) to 1.
- Interrupt operation is described below.
  (1) If an interrupt occurs with its corresponding bit in the interrupt enable register set to 1, an interrupt request signal is sent to the interrupt controller.
  (2) On receiving the interrupt request signal, the corresponding interrupt request flag is set.
  (3) If multiple interrupt requests are generated with their corresponding interrupt request flags set to 1, the interrupt with the highest priority at that time is requested. Other interrupt requests are held pending.
  (4) The CPU checks the I bit setting in CCR. If the I bit is cleared to 0, the interrupt request is accepted. If the I bit is set to 1, the interrupt request is held pending.
  (5) If the CPU accepts the interrupt, after processing of the current instruction is completed, both the program counter (PC) and CCR are saved to the stack. This PC value saved in the stack is the address of the first instruction to be executed upon return from interrupt handling.
  (6) The I bit in CCR is set to 1 to mask all other interrupts.
  (7) The CPU generates the vector address corresponding to the accepted interrupt, and the interrupt handling routine starts execution from the address indicated in that address.
- Disabling of interrupts by clearing bits in IENR1 and clearing of bits in IRR1 must be done while interrupts are masked (I bit is set to 1).
- If the above clearing operations are performed while the I bit is 0, a conflict may arise between the clearing instruction and an interrupt request. In such situation, exception handling for the interrupt will be executed after the execution of the clearing instruction has been completed.

2. Table 2.1 shows the function assignments in this sample task. The functions are assigned as shown in table 2.1 to start counting when an external interrupt occurs.

Table 2.1 Assignment of Functions

Pin/Register	Assigned Function
IRRI0	Indicates whether or not an IRQ0 interrupt has been requested.
IEN0	Enables $\overline{\text{IRQ0}}$ pin interrupt requests.
IEG0	Selects input edge of $\overline{\text{IRQ0}}$ pin.
$\overline{\text{IRQ0}}$	Input pin of the switch input
PDR9	Stores P92 output pin data.
P92	Output pin for LED output

3. Principle of Operation

1. Figure 3.1 illustrates the operation of this sample task. Counting is started by an external interrupt through hardware and software processing as shown in the figure.

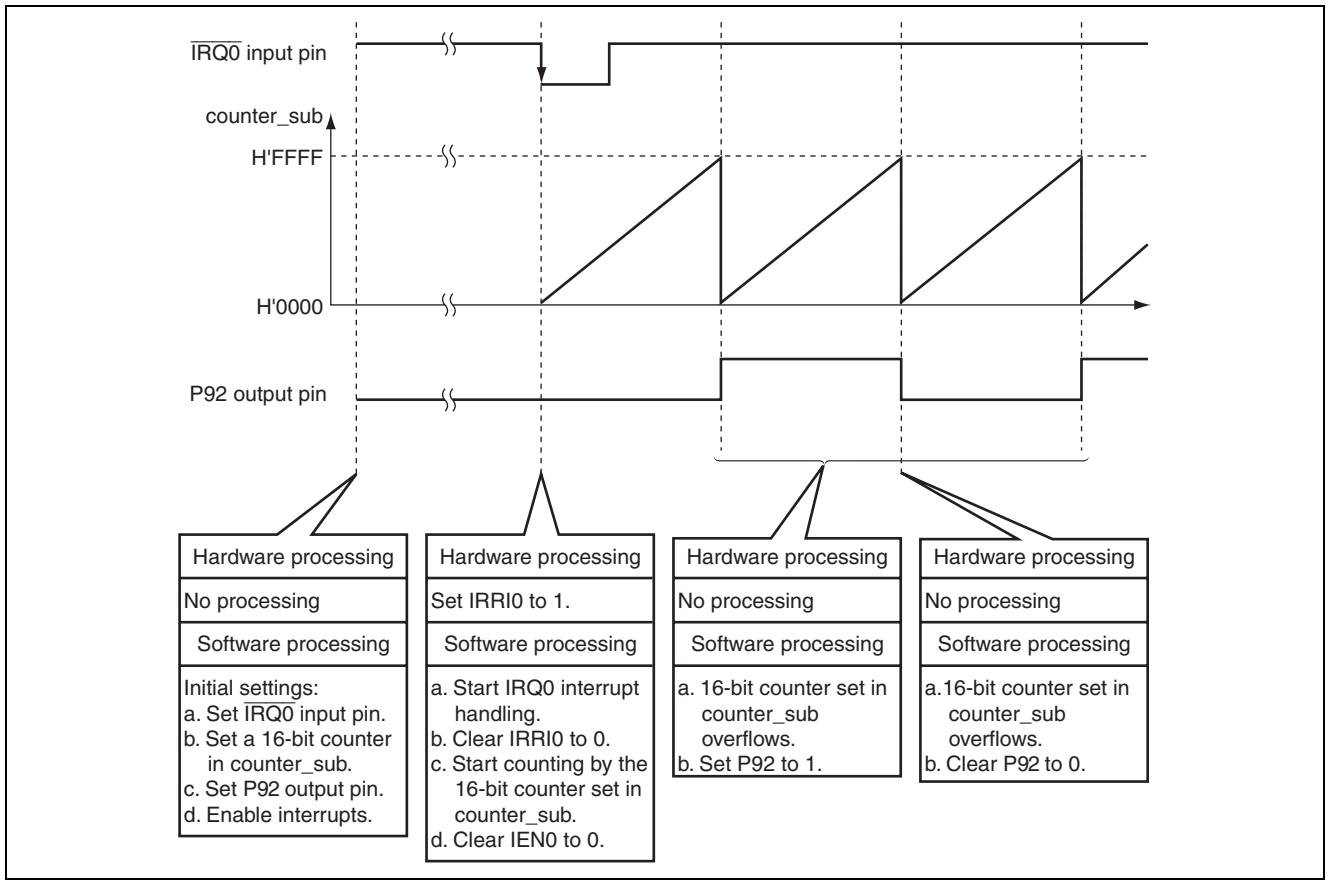


Figure 3.1 Operation Principle of Starting Counter Incrementation by External Interrupt

4. Description of Software

4.1 Modules

Table 4.1 describes the modules in this sample task.

Table 4.1 Description of Modules

Module	Label	Function
Main Routine	main	Sets IRQ0 interrupts, sets LED output pins, enables interrupts, increments the 16-bit counter and outputs to the LED.
Switch On	irq0int	An IRQ0 interrupt handling routine that sets SWONF to 1.

4.2 Arguments

Arguments are not used in this sample task.

4.3 Internal Registers

Table 4.2 describes the internal registers involved in this sample task.

Table 4.2 Description of Internal Registers

Register		Function	Address	Setting
IEGR	IEG0	IRQ Edge Select Register (IRQ0 Edge Select) If IEG0 = 0, falling edges of $\overline{IRQ0}$ pin input are detected. If IEG0 = 0, rising edges of $\overline{IRQ0}$ pin input are detected.	H'FFF2 Bit 0	0
IENR1	IEN0	Interrupt Enable Register 1 (IRQ0 Interrupt Enable) If IEN0 = 0, $\overline{IRQ0}$ pin interrupt request is disabled. If IEN0 = 1, $\overline{IRQ0}$ pin interrupt request is enabled.	H'FFF3 Bit 0	1
IRR1	IRRI0	Interrupt Request Register 1 (IRQ0 Interrupt Request Flag) If IRRI0 = 0, $\overline{IRQ0}$ pin interrupt is not requested. If IRRI0 = 1, $\overline{IRQ0}$ pin interrupt has been requested.	H'FFF6 Bit 0	0
PDR9	P92	Port Data Register 9 (Port Data Register 92) If P92 = 0, the output level of P92 pin is Low. If P92 = 1, the output level of P92 pin is High.	H'FFDC Bit 2	0
PMR2	IRQ0	Port Mode Register 2 (P43/IRQ0 pin function switch) If IRQ0 = 0, the P43/$\overline{IRQ0}$ pin functions as P43 output pin. If IRQ0 = 1, the P43/$\overline{IRQ0}$ pin functions as $\overline{IRQ0}$ input pin.	H'FFE0 Bit 0	1

4.4 Description of RAM

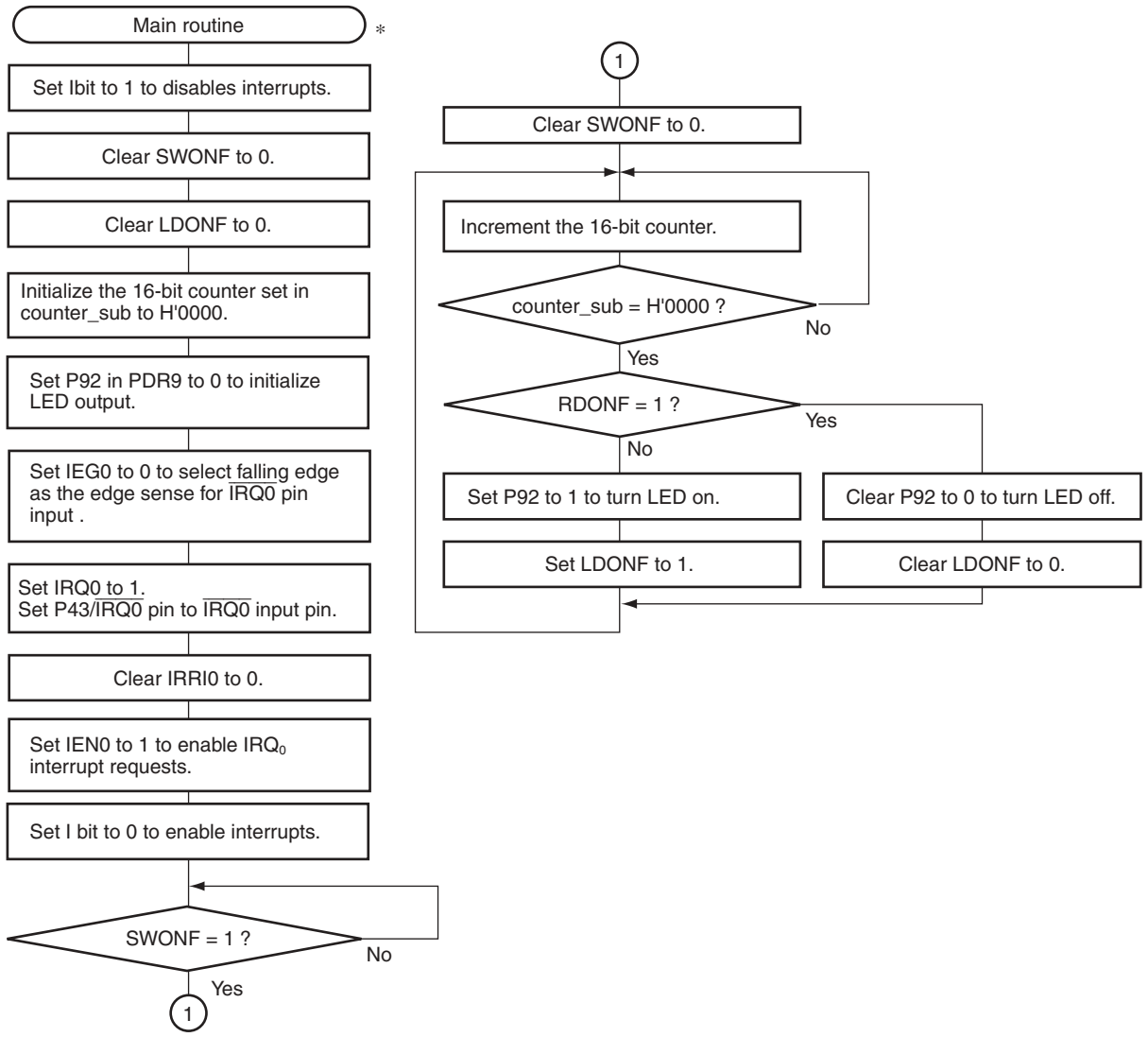
The RAM area used in this sample task are described in table 4.3.

Table 4.3 Description of RAM

Label		Function	Address	Used in
counter_sub		16-bit up counter	H'FB80	Main Routine
USRF	SWONF	Flag to judge ON/OFF of switch input	H'FB82 Bit 0	Main Routine Switch On
	LDONF	Flag to judge ON/OFF of LED	H'FB82 Bit 1	Main Routine

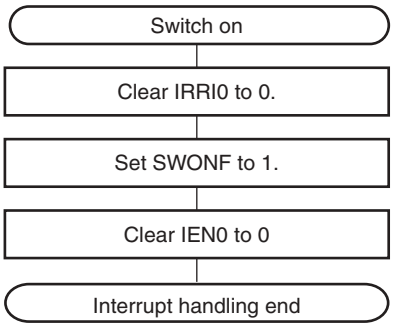
5. Flowchart

1. Main routine



Note: * In this sample task, the stack pointer is set in INIT.SRC (assembly language).

2. IRQ0 interrupt handling routine



6. Program Listing

INIT.SRC (Program listing)

```
    .EXPORT   _INIT
    .IMPORT   _main
;
    .SECTION  P,CODE
_INIT:
    MOV.W     #H'FF80,R7
    LDC.B     #B'10000000,CCR
    JMP       @_main
;
    .END
```

```
/************************************************************************************************************************/
/*                                                                                                                      */
/*  H8/300L Super Low Power Series                                                                                      */
/*      -H8/38024 Series-                                                                                               */
/*  Application Note                                                                                                    */
/*                                                                                                                      */
/*  'Count Start by External Interrupt'                                                                                 */
/*                                                                                                                      */
/*  Function                                                                                                            */
/*  : External Interrupt                                                                                                */
/*                                                                                                                      */
/*  External Clock : 10MHz                                                                                              */
/*  Internal Clock : 5MHz                                                                                               */
/*  Sub Clock      : 32.768kHz                                                                                          */
/*                                                                                                                      */
/************************************************************************************************************************/

#include    <machine.h>

/************************************************************************************************************************/
/*  Symbol Definition                                                                                                   */
/************************************************************************************************************************/
struct BIT {
    unsigned char   b7:1;       /* bit7 */
    unsigned char   b6:1;       /* bit6 */
    unsigned char   b5:1;       /* bit5 */
    unsigned char   b4:1;       /* bit4 */
    unsigned char   b3:1;       /* bit3 */
    unsigned char   b2:1;       /* bit2 */
    unsigned char   b1:1;       /* bit1 */
    unsigned char   b0:1;       /* bit0 */
};

#define     PDR9_BIT    (*(struct BIT *)0xFFDC)             /* Port Data Register 9                  */
#define     P92         PDR9_BIT.b2                         /* Port Data Register 92                 */
#define     IEGR1_BIT   (*(struct BIT *)0xFFF2)             /* Interrupt Edge Select Register 1      */
#define     IEG0        IEGR1_BIT.b0                        /* IEG0 Edge Select                      */
#define     IENR1_BIT   (*(struct BIT *)0xFFF3)             /* Interrupt Enable Register 1           */
#define     IEN0        IENR1_BIT.b0                        /* IEN0 Interrupt Enable                 */
#define     IRR1_BIT    (*(struct BIT *)0xFFF6)             /* Interrupt Request Register 1          */
#define     IRRI0       IRR1_BIT.b0                         /* IRRI0 Interrupt Request Register      */
#define     PMR2_BIT    (*(struct BIT *)0xFFC9)             /* Port Mode Register 2                  */
#define     WDCKS       PMR2_BIT.b2                         /* Watchdog Timer Source Clock           */
#define     IRQ0        PMR2_BIT.b0                         /* P43/IRQ0 Select                       */
```

```
#pragma interrupt   (irq0int)
/************************************************************************************************************************/
/*  Function define                                                                                                     */
/************************************************************************************************************************/
extern void INIT ( void );                                  /* SP Set                                                   */
void        main ( void );
void        irq0int ( void );

/************************************************************************************************************************/
/*  RAM define                                                                                                          */
/************************************************************************************************************************/
unsigned int    counter_sub;
unsigned char   USRF;                                       /* User Flag Area                                           */

#define     USRF_BIT    (*(struct BIT *)&USRF)
#define     SWONF       USRF_BIT.b0                         /* Switch On Flag                                           */
#define     LDONF       USRF_BIT.b1                         /* LED On Flag                                              */

/************************************************************************************************************************/
/*  Vector Address                                                                                                      */
/************************************************************************************************************************/
#pragma section     V1                                      /* Vector Section Set                                       */
void (*const VEC_TBL1[])(void) = {
    INIT                                                    /* 0x0000 Reset Vector                                      */
};
#pragma section     V2                                      /* Vector Section Set                                       */
void (*const VEC_TBL2[])(void) = {
    irq0int                                                 /* 0x0008 IRQ0 Interrupt Vector                             */
};

#pragma section                                             /* P                                                        */
/************************************************************************************************************************/
/*  Main Program                                                                                                        */
/************************************************************************************************************************/
void    main ( void )
{
    set_imask_ccr(1);                                       /* Interrupt Disable                                        */

    SWONF = 0;                                              /* Initialize SWONF                                         */
    LDONF = 0;                                              /* Initialize LDONF                                         */
    counter_sub = 0x0000;                                   /* Initialize 16bit Counter                                 */

    P92 = 0;                                                /* Initialize LED Output                                    */

    IEG0 = 1;                                               /* Initialize IRQ0 Terminal Input Edge                      */

    IRQ0 = 1;                                               /* Initialize Input TerminalIRQ0                            */

    IRRI0 = 0;                                              /* Initialize IRRI0                                         */

    IEN0 = 1;                                               /* IRQ0 Interrupt Request Enable                            */

    set_imask_ccr(0);                                       /* Interrupt Enable                                         */

    while(SWONF ! = 1){                                     /* SWONF = 0 ?                                              */
        ;
    }
```

```
    SWONF = 0;                                          /* Clear SWONF                  */

    while(1){
        do{
            counter_sub++;                              /* Increment 16bit Counter      */
        }while(counter_sub ! = 0x0000);                 /* 16bit Counter = H'0000 ?     */

        if(LDONF == 1){                                 /* LDONF = 1 ?                  */
            P92 = 0;                                    /* Turn Off LED                 */
            LDONF = 0;                                  /* Clear LDONF                  */
        }
        else{
            P92 = 1;                                    /* Turn On LED                  */
            LDONF = 1;                                  /* Set LDONF                    */
        }
    }
}

/************************************************************************************************************/
/*  IRQ0 Interrupt                                                                                          */
/************************************************************************************************************/
void    irq0int ( void )
{
    IRRI0 = 0;                                          /* Clear IRRI0                  */

    SWONF = 1;                                          /* Set SWONF                    */

    IEN0 = 0;                                           /* IRQ0 Interrupt Disable       */
}
```

Link address specifications

Section Name	Address
CV1	H'0000
CV2	H'001C
P	H'0100
B	H'FB80

Revision Record

		Description	
Rev.	Date	Page	Summary
1.00	Dec.19.03	—	First edition issued

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.