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

Demonstrating the Watchdog Timer

Introduction

Watchdog operation is performed using the watchdog timer function. The overflow period of Timer Counter W is 393.216 ms. By turning on the switch connected to the IRQ0 input pin, Timer Counter W overflows, and a watchdog operation is performed.

Target Device

H8/38024

Contents

1. Specifications

1. Watchdog operation is performed using the watchdog timer function.
2. The overflow period of Timer Counter W is set to 393.216 ms and an internal reset signal is generated unless Timer Counter W is initialized within 393.216 ms.
3. In normal operation, settings are made so that the LED repeats turning on and off at a certain time interval and that Timer Counter W is initialized before it overflows.
4. By turning on the switch connected to the $\overline{\text{IRQ0}}$ input pin, Timer Counter W is not initialized but overflows, and an internal reset signal is generated.
5. The LED is connected to the P92 output pin of Port 9.
6. The P92 pin is a high-current port.
7. Figure 1.1 shows an example of connecting a switch to the $\overline{\text{IRQ0}}$ input pin.

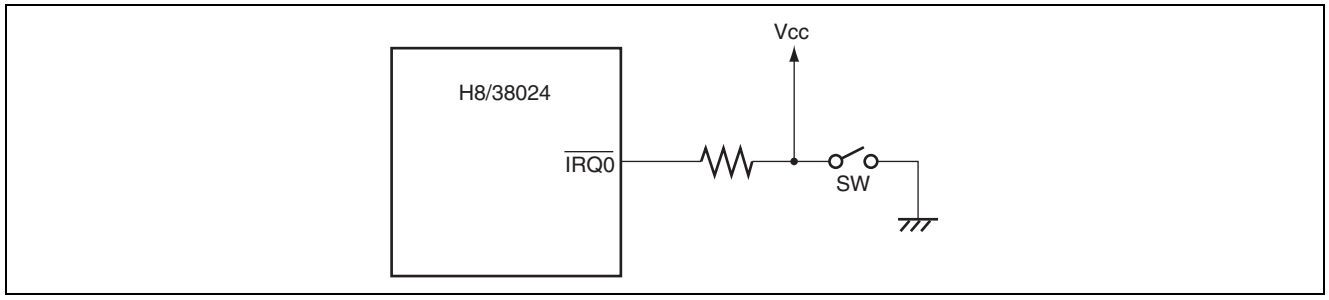


Figure 1.1 Example of Connecting Switch to $\overline{\text{IRQ0}}$ Input Pin

2. Description of Functions

1. In this task example, watchdog operation is performed using the watchdog timer function.
   A. Figure 2.1 shows the block diagram of the watchdog timer function which is described below.
      - The system clock (ϕ) is a 5 MHz clock and a reference clock to operate the CPU and its peripheral functions.
      - Prescaler S (PSS) is a 13-bit counter using ϕ as its input clock and is counted up every cycle.
      - The Timer Counter W (TCW) is an 8-bit read/write up counter and is counted up by an internal clock which is input. The clock input is ϕ/8192.
      - The Timer Control/Status Register W (TCSRW) is an 8-bit read/write register and controls TCSRW and TCW writing, controls watchdog timer operation, and indicates operation status.
      - The calculation method of the TCW overflow period in this task example is shown below.

$$\text{TCW overflow period} = \frac{1}{\text{System clock} / 8192} \times (256 - (\text{TCW reload value}))$$
$$= 1.638\ \text{ms} \times (256 - 16)$$
$$= 393.216\ \text{ms}$$

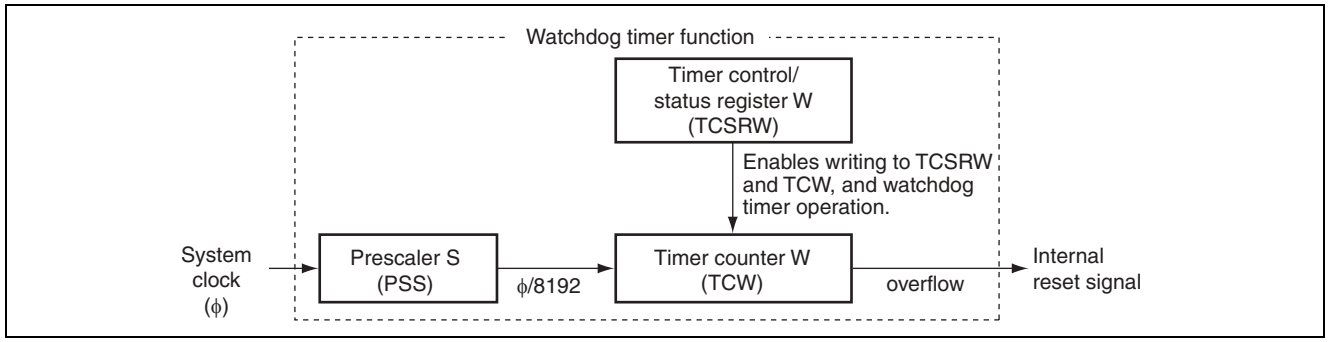


Figure 2.1 Block Diagram of Watchdog Timer Functions

2. Table 2.1 shows the assignment of functions in this task example. The functions are assigned as shown in table 2.1 to perform watchdog operation by the watchdog timer function.

Table 2.1 Assignment of Functions

Function	Assignment
PSS	A 13-bit counter which uses the system clock as input
TCSRW	Controls TCSRW and TCW writing, controls watchdog timer operation, and indicates operation status.
TCW	An 8-bit counter which uses a clock obtained by dividing the system clock by 8192 as input
WDCKS	Selects the system clock/8192 for the clock source of the watchdog timer.
IEG0	Selects the interrupt edges of $\overline{\text{IRQ0}}$ pin.
IEN0	Enables IRQ0 interrupts of SW ($\overline{\text{IRQ0}}$ pin).
IRRI0	Judges ON/OFF of SW ($\overline{\text{IRQ0}}$ pin).
IRQ0	Input pin of SW input
P92	LED output

3. Principle of Operation

1. Figure 3.1 illustrates the principle of operation of this sample task. As shown in figure 3.1, watchdog operation is performed by the watchdog timer function by means of hardware processing and software processing.

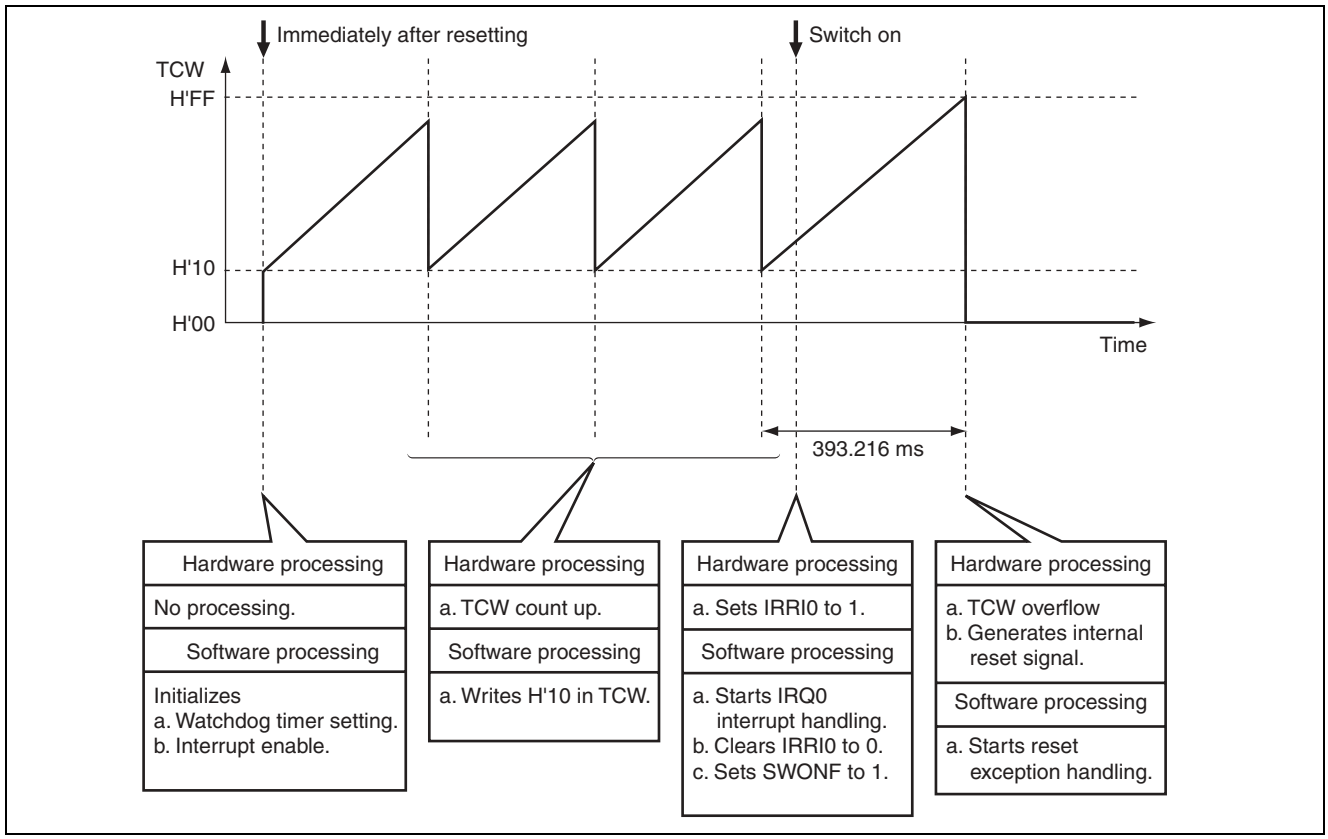


Figure 3.1 Operation Principle of Watchdog Operation by Watchdog Timer Function

4. Description of Software

4.1 Modules

Table 4.1 describes the modules in this task example.

Table 4.1 Description of Modules

Module	Label	Function
Main Routine	main	Sets the watchdog timer function, enables interrupts, controls the LED, and judges whether or not the switch connected to $\overline{\text{IRQ0}}$ is turned on.
Switch On	irq0int	An $\overline{\text{IRQ0}}$ interrupt handling routine. Sets SWONF to 1.

4.2 Arguments

Arguments are not used in this task example.

4.3 Internal registers

Table 4.2 describes the internal registers in this task example.

Table 4.2 Description of Internal Registers

Register		Function	Address	Setting
TCSRW	B6WI	Timer Control/Status Register W (Bit 6 Write Disable) If B6WI = 0, writing to bit 6 in TCSRW is enabled. If B6WI = 1, writing to bit 6 in TCSRW is disabled.	H'FFC0 Bit 7	1
	TCWE	Timer Control/Status Register W (Timer Counter W Write Enable) If TCWE = 1, writing of 8-bit data to TCW is enabled.	H'FFC0 Bit 6	1
	B4WI	Timer Control/Status Register W (Bit 4 Write Disable) If B4WI = 0, writing to bit 4 in TCSRW is enabled. If B4WI = 1, writing to bit 4 in TCSRW is disabled.	H'FFC0 Bit 5	1
	TCSRWE	Timer Control/Status Register W (Timer Control/Status Register W Write Enable) If TCSRWE = 1, writing to bits 2 and 0 in TCSRW is enabled.	H'FFC0 Bit 4	1
	B2WI	Timer Control/Status Register W (Bit 2 Write Disable) If B2WI = 0, writing to bit 2 in TCSRW is enabled. If B2WI = 1, writing to bit 2 in TCSRW is disabled.	H'FFC0 Bit 3	1
	WDON	Timer Control/Status Register W (Watchdog Timer On) If WDON = 1, watchdog timer operation is enabled.	H'FFC0 Bit 2	1
	B0WI	Timer Control/Status Register W (Bit 0 Write Disable) If B0WI = 0, writing to bit 0 in TCSRW is enabled. If B0WI = 1, writing to bit 0 in TCSRW is disabled.	H'FFC0 Bit 1	1
	WRST	Timer Control/Status Register W (Watchdog Timer Reset) If WRST = 0, TCW has not overflowed and no internal reset signal is generated. If WRST = 1, TCW has overflowed and an internal reset signal is generated.	H'FFC0 Bit 0	0

Register		Function	Address	Setting
TCW		Timer Counter W An 8-bit counter to input a clock obtained by dividing the system clock by 8192.	H'FFC1	H'10
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	WDCKS	Port Mode Register 2 (WatchDog Timer Source Clock) If WDCKS = 0, source clock of watchdog timer is ϕ (system clock)/8192. If WDCKS = 1, source clock of watchdog timer is ϕw (subclock) /32.	H'FFE0 Bit 2	0
	IRQ0	Port Mode Register 2 (P43/$\overline{\text{IRQ0}}$ Pin Function Switch) If $\overline{\text{IRQ0}}$ = 0, P43/$\overline{\text{IRQ0}}$ pin functions as P43 input/output pin. If $\overline{\text{IRQ0}}$ = 1, P43/$\overline{\text{IRQ0}}$ pin functions as $\overline{\text{IRQ0}}$ input pin.	H'FFE0 Bit 0	1
IEGR	IEG0	IRQ Edge Select Register (IRQ Edge Select If IEG0 = 0, the falling edge is selected as $\overline{\text{IRQ0}}$ pin input detection edge. If IEG0 = 1, the rising edge is selected as $\overline{\text{IRQ0}}$ pin input detection edge.	H'FFF2 Bit 0	0
IENR1	IEN0	Interrupt Enable Register 1 ($\overline{\text{IRQ0}}$ Interrupt Request Enable) If IEN0 = 0, interrupt requests from the $\overline{\text{IRQ0}}$ pin are disabled. If IEN0 = 1, interrupt requests from the $\overline{\text{IRQ0}}$ pin are enabled.	H'FFF3 Bit 0	1
IRR1	IRRI0	Interrupt Request Register 1 (IRQ0 Interrupt Request Flag) If IRRI0 = 0, $\overline{\text{IRQ0}}$ pin interrupt is not requested. If IRRI0 = 1, $\overline{\text{IRQ0}}$ pin interrupt is requested.	H'FFF6 Bit 0	0

4.4 Description of RAM

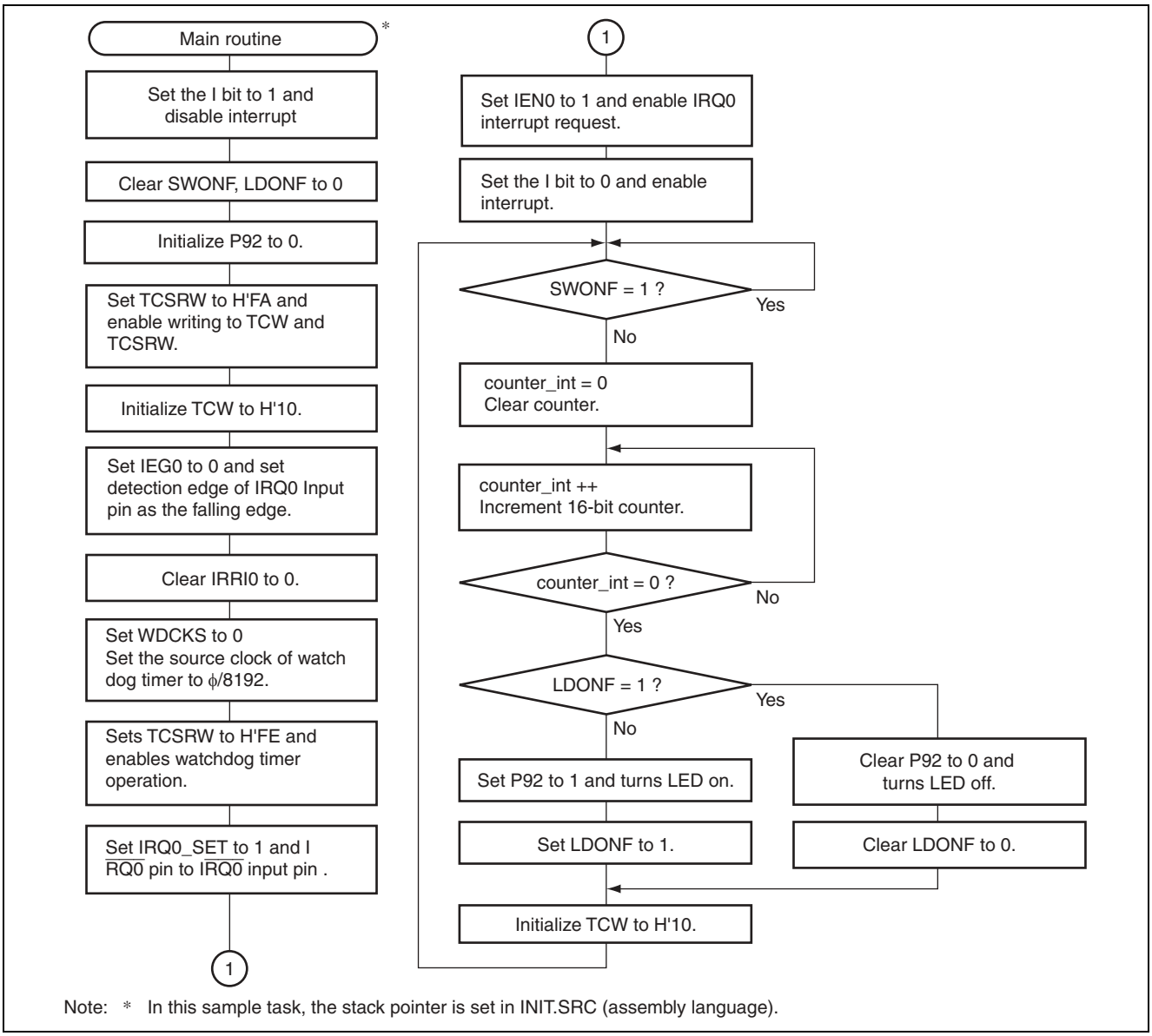
The RAMs used in this task example are described in table 4.3.

Table 4.3 Description of RAM

Label		Function	Address	Used in
counter_int		Up-counter that turns on or off the LED	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 the LED	H'FB82 Bit 1	Main Routine

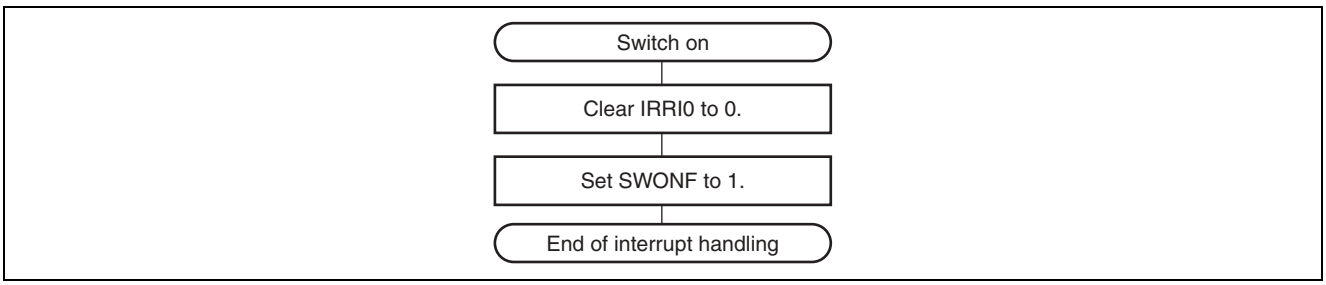
5. Flowchart

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                                                                                                    */
/*                                                                                                                      */
/*  'Watchdog Timer'                                                                                                    */
/*                                                                                                                      */
/*  Function                                                                                                            */
/*  : Watchdog Timer                                                                                                    */
/*                                                                                                                      */
/*  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     TCSRW       *(volatile unsigned char *)0xFFB2       /* Timer Control/Status Register W          */
#define     TCSRW_BIT   (*(struct BIT *)0xFFB2)                 /* Timer Control/Status Register W          */
#define     B6WI        TCSRW_BIT.b7                            /* Bit-6 Write Disable                      */
#define     TCWE        TCSRW_BIT.b6                            /* Timer Counter W Write Enable             */
#define     B4WI        TCSRW_BIT.b5                            /* Bit-4 Write Disable                      */
#define     TCSRWE      TCSRW_BIT.b4                            /* Timer Control/Status Register W          */
                                                                /*                              Write Enable */
#define     B2WI        TCSRW_BIT.b3                            /* Bit-2 Write Disable                      */
#define     WDON        TCSRW_BIT.b2                            /* Watchdog Timer ON                        */
#define     B0WI        TCSRW_BIT.b1                            /* Bit-0 Write Disable                      */
#define     WRST        TCSRW_BIT.b0                            /* Watchdog Timer Reset                     */
```

```
#define     TCW         *(volatile unsigned char *)0xFFB3       /* Timer Counter W                          */
#define     PDR9_BIT    (*(struct BIT *)0xFFDC)                 /* Port Data Register 9                     */
#define     P92         PDR9_BIT.b2                             /* Port Data Register 9 bit2                */
#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_int;
unsigned char   USRF;                                           /* User Flag Area                           */

#define     USRF_BIT    (*(struct BIT *)&USRF)
#define     SWONF       USRF_BIT.b0
#define     LDONF       USRF_BIT.b1

/************************************************************************************************************/
/*  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                         */

    P92 = 0;                                                    /* Initialize P92 Terminal Output Level     */

    TCSRW = 0x5A;                                               /* TCW And TCSRW Write Enable               */

    TCW = 0x10;                                                 /* Initialize TCW                           */
```

```
    IEG0 = 0;                                   /* Initialize IRQ0  Terminal Input Edge     */
    IRRI0 = 0;                                  /* Initialize IRQ0 Interrupt Request Flag   */
    WDCKS = 0;                                  /* Initialize IRQ0  Input Terminal          */

    TCSRW = 0xFE;                               /* Watchdog Timer On                        */

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

    set_imask_ccr(0);                           /* Interrupt Enable                         */

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

        counter_int = 0;                        /* counter Clear                            */
        do{
            counter_int ++;                     /* counter Countup                          */
        }while(counter_int ! = 0);              /* counter_int = FFFF ?                     */

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

        TCW = 0x10;                             /* Initialize TCW                           */
    }
}
/*******************************************************************************************************************/
/*  IRQ0 Interrupt                                                                                                 */
/*******************************************************************************************************************/
void    irq0int ( void )
{
    IRRI0 = 0;                                  /* Clear IRRI0                              */

    SWONF = 1;                                  /* Set SWONF                                */
}
```

Link address specifications

Section Name	Address
CV1	H'0000
CV2	H'0008
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.