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.

H8/300H Tiny Series

Entering Subsleep Mode

Introduction

Subsleep mode is entered.

Target Device

H8/3664

Contents

1. Specifications

- Subsleep mode is entered.
- This LSI goes directly from active mode to subactive mode when a SLEEP instruction is executed while the SSBY bit is set to X (either 1 or 0) in SYSCR1, the LSON and DTON bits are set to 1 and the SMSEL bit is set to X (either 1 or 0) in SYSCR2.
- This LSI goes from subactive mode to subsleep mode when a SLEEP instruction is executed while the SSBY bit is cleared to 0 in SYSCR1, the DTON bit is cleared to 0, and the SMSEL and LSON bits are set to 1 in SYSCR2.
- Subsleep mode is canceled and this LSI returns to subactive mode on receiving a timer A interrupt.
- Timer A interrupt handling controls the LED and counts the number of times a timer A interrupt has been requested. A timer A interrupt occurs every 0.5 s. After a timer A interrupt has been requested for the 120th time, timer A interrupt requests are disabled, and execution stops. The LED is turned on and off every 0.5 s.
- After transiting to subactive mode because a timer A interrupt has occurred, the number of times a timer A interrupt has been requested is counted, and the LSI then reenters subsleep mode. This processing is repeated until a timer A interrupt has occurred 120 times.
- The LED is connected to the P74 output pin of port 7.

2. Description of Functions Used

In this sample task, this LSI enters subsleep mode, a power-down mode. Figure 1 shows a diagram of transition to subsleep mode. The subsleep mode functions are described below.

- This LSI transits directly from active mode to subactive mode when a SLEEP instruction is executed while the SSBY bit is set to X (either 1 or 0) in SYSCR1, the LSON and DTON bits are set to 1 and the SMSEL bit is set to X (either 1 or 0) in SYSCR2. Then when a SLEEP instruction is executed while the SSBY bit is cleared to 0 in SYSCR1, the DTON bit is cleared to 0, and the SMSEL and LSON bits are set to 1 in SYSCR2, this LSI goes from subactive mode to subsleep mode.
- In subsleep mode, operation of the on-chip peripheral modules other than timer A, timer V, the watchdog timer, and the I^2C bus interface is halted.
- As long as the rated voltage is supplied, the contents of the CPU registers, on-chip RAM, and some on-chip peripheral module registers are retained. The I/O ports keep the same states as before the transition.
- Subsleep mode is cleared by a timer A, IRQ3 to IRQ0, or WKP5 to WKP0 interrupt, or by input at the $\overline{\text{RES}}$ pin.
- In the case of clearing subsleep mode with an interrupt, when an interrupt is requested, subsleep mode is cleared and interrupt handling starts.
- Subsleep mode is not cleared if the I bit in the condition code register (CCR) is set to 1 or the requested interrupt is disabled in the corresponding interrupt enable register.
- In using the $\overline{\text{RES}}$ pin to initiate the transition from subsleep mode, the IC enters the reset state and cancels subsleep mode when a low level is placed on the $\overline{\text{RES}}$ pin. Once the pulse generator output has become stable, the $\overline{\text{RES}}$ pin is driven high, after which the CPU starts reset exception handling. Since system clock signals are supplied to the entire LSI as soon as the system clock pulse generator starts functioning, the $\overline{\text{RES}}$ pin must be kept low until the pulse generator output is stable.
- In this sample task, subsleep mode is cleared by a timer A interrupt. After exit from subsleep mode, a transition is made to subactive mode.
- If a SLEEP instruction is executed while the SSBY bit is set to X (either 1 or 0) in SYSCR1, and the SMSEL bit is set to X (either 1 or 0), the LSON bit is cleared to 0, and the DTON bit is set to 1 in SYSCR2 in subactive mode, a direct transition is made to active mode after the waiting time set in the STS2 to STS0 bits in SYSCR1 has elapsed.
- The oscillation stabilization waiting time after exit from subactive mode is set by the STS2 to STS0 bits in SYSCR1.
- In this sample task, the operating frequency is 16 MHz, and the waiting time is 131,072 states (oscillation stabilization waiting time: 8.2 ms).

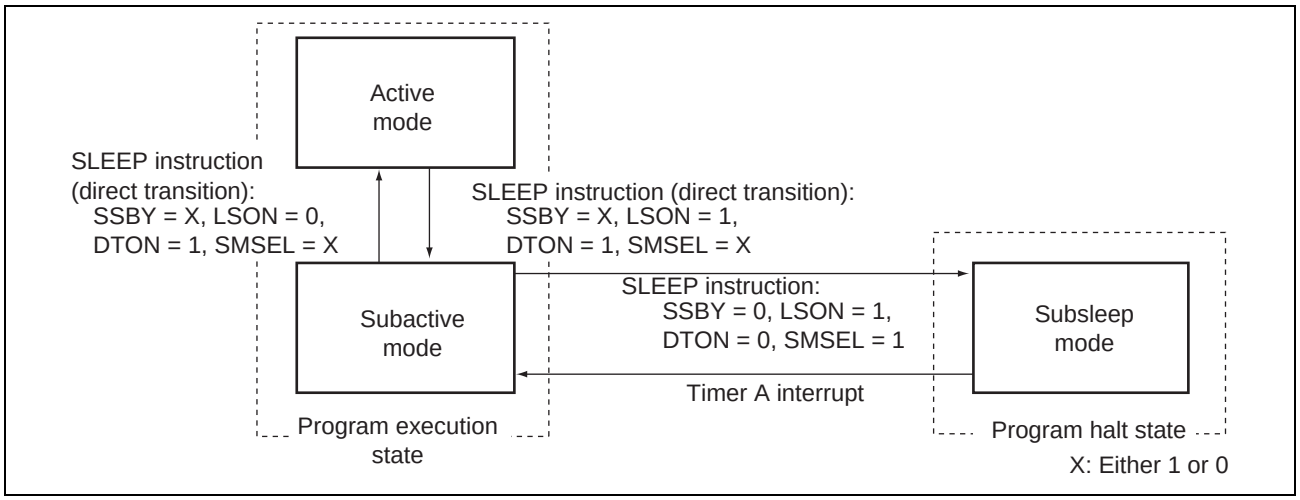


Figure 1 Transitions to and from Subsleep Mode

Table 1 lists the function allocation for this sample task. The functions listed in table 1 are allocated for a transition to subsleep mode.

Table 1 Function Allocation

Function	Description
SYSCR1	Controls power-down mode
SYSCR2	Controls power-down mode
PCR7	Sets P74 output pin function
PDR7	Stores P74 output pin data
P74	LED output pin
TMA	Selects the clock time-base function for timer A and sets the TCA overflow cycle
TCA	8-bit up-counter that overflows every 0.5 s by the clock time-base function
IRRTA	Indicates whether or not a timer A interrupt request is issued
IENTA	Enables timer A interrupt requests

3. Description of Operations

Figure 2 shows this sample task's principle of operation. The hardware and software processing shown in figure 2 performs a transition to subsleep mode.

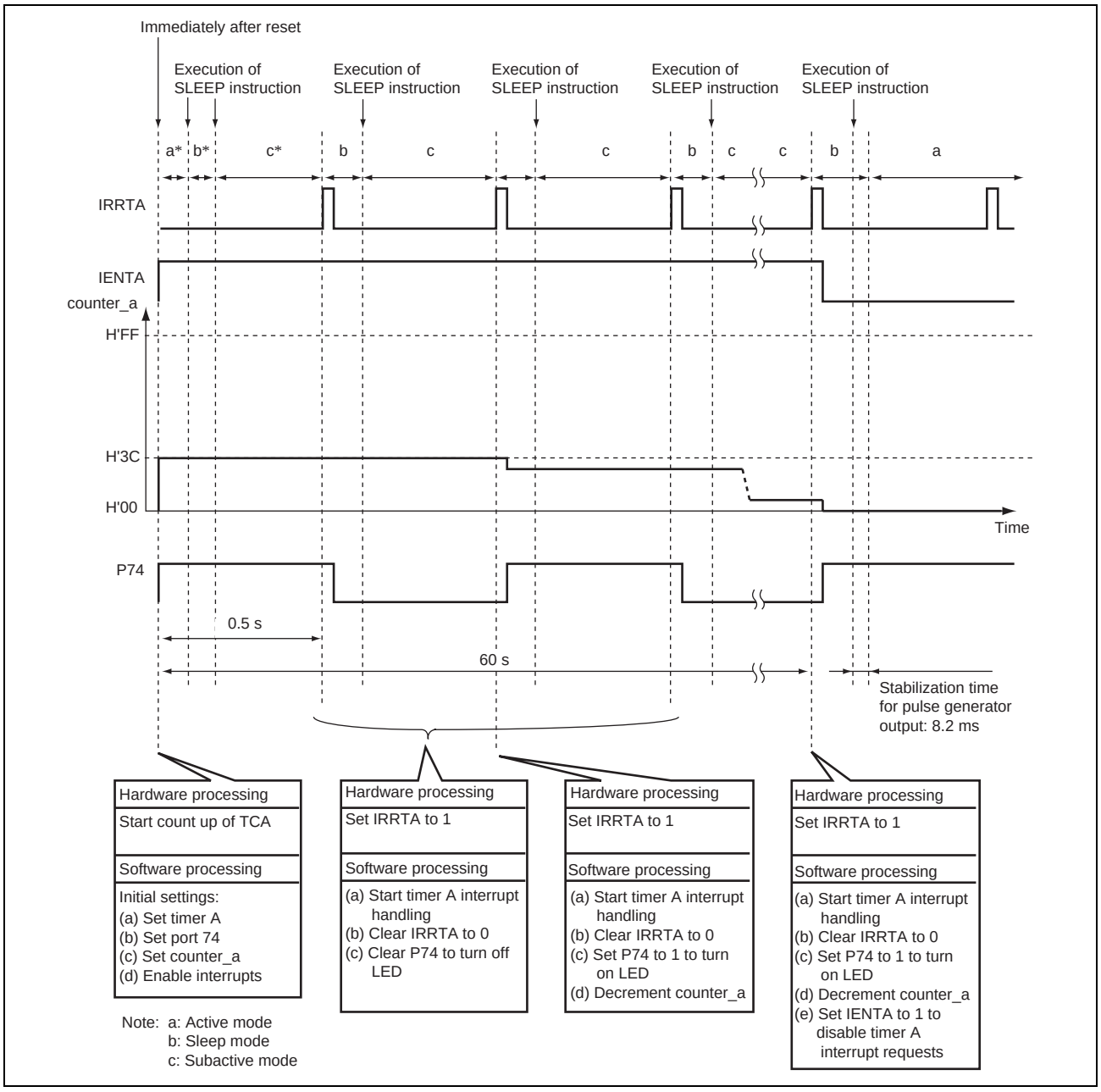


Figure 2 Operation Principle: Transition to Subsleep Mode

4. Description of Software

4.1 Description of Modules

Table 2 describes the software used in this sample task.

Table 2 Description of Modules

Module Name	Label Name	Function
Main routine	main	Sets timer A interrupts, port 7, and counter_a, enables interrupts, transits to subactive mode, subsleep mode, and active mode.
LED control	taint	During the timer A interrupt handling routine, controls the LED, decrements the 8-bit counter that counts timer A interrupts, and disables timer A interrupt requests after 60 s.
Direct transition	dtint	During the direct transition interrupt handling routine, clears the direct transition interrupt request flag.

4.2 Description of Arguments

No arguments are used in this sample task.

4.3 Description of Internal Registers

Table 3 describes the internal registers used in this sample task.

Table 3 Description of Internal Registers

Register Name		Function	Address	Setting
TMA		Timer mode register A: When TMA is set to H'19, timer A is set to the clock time-base function, and the TCA overflow cycle is set to 0.5 s.	H'FFA6	H'19
TCA		Timer counter A: 8-bit counter that overflows every 0.5 s by clock time-base and has clock input of PSW output clock	H'FFA7	H'00
PDR7	P74	Port data register 7 (port data register 74): When P74 is cleared to 0, the P74 pin output level is low. When P74 is set to 1, the P74 pin output level is high.	H'FFDA Bit 4	 0
PCR7	PCR74	Port control register 7 (port control register 74): When PCR74 is set to 1, the P74 pin functions as an output pin.	H'FFEA Bit 4	 1

Table 4 Description of Internal Registers (cont)

Register	Name	Function	Address	Setting
SYSCR1	SSBY	System control register 1 (software standby): When SSBY is set to X (either 1 or 0), after execution of a SLEEP instruction, a direct transition is made to subactive mode or active mode.	H'FFF0 Bit 7	1 (in this sample task)
		System control register 1 (standby timer select 2 to 0):	H'FFF0	
	STS2	When STS2 is set to 1 and STS1 and STS0 are both cleared to 0, the wait time is set to 131.072 states.	Bit 6	STS2 = 1
	STS1		Bit 5	STS1 = 0
	STS0		Bit 4	STS0 = 0
SYSCR2	SMSEL	System control register 2 (sleep mode selection): When SMSEL is cleared to 0, after execution of a SLEEP instruction, a transition is made to sleep mode.	H'FFF1 Bit 7	0
	LSON	System control register 2 (low speed on flag): When LSON is set to 1, sleep mode, subsleep mode, or subactive mode (direct transition) is selected as the mode to transit to after execution of a SLEEP instruction.	H'FFF1 Bit 6	1
	DTON	System control register 2 (direct transfer on flag): When DTON is set to 1, active mode or subactive mode is selected as the mode to transit to after execution of a SLEEP instruction.	H'FFF1 Bit 5	1
		System control register 2 (active mode clock select 2 to 0:)	H'FFF1	
	MA2		Bit 4	MA2 = 1
	MA1	When MA2, MA1, and MA0 are all set to 1, ϕOSC/64 is selected as the clock in active mode.	Bit 3	MA1 = 1
	MA0		Bit 2	MA0 = 1
		System control register 2 (subactive mode clock select 1 and 0:)	H'FFF1	
	SA1		Bit 1	SA1 = 0
	SA0	When SA1 and SA0 are both cleared to 0, ϕw/8 is selected as the CPU operating clock in subactive mode.	Bit 0	SA0 = 0
IENR1	IENDT	Interrupt enable register 1 (direct transition interrupt enable): When IENDT is cleared to 0, direct transition interrupt requests are disabled. When IENDT is set to 1, direct transition interrupt requests are enabled.	H'FFF4 Bit 7	1
	IENTA	Interrupt enable register 1 (timer A interrupt enable): When IENTA is cleared to 0, timer A interrupt requests are disabled. When IENTA is set to 1, timer A interrupt requests are enabled.	H'FFF4 Bit 6	1

Table 4 Description of Internal Registers (cont)

Register	Name	Function	Address	Setting
IRR1	IRRDT	Interrupt request register 1 (direct transition interrupt request flag): When IRRDT is cleared to 0, no direct transition interrupt is requested. When IRRDT is set to 1, a direct transition interrupt is requested.	H'FFF6 Bit 7	0
	IRRTA	Interrupt request register 1 (timer A interrupt request flag): When IRRTA is cleared to 0, no timer A interrupt is requested. When IRRTA is set to 1, a timer A interrupt is requested.	H'FFF6 Bit 6	0

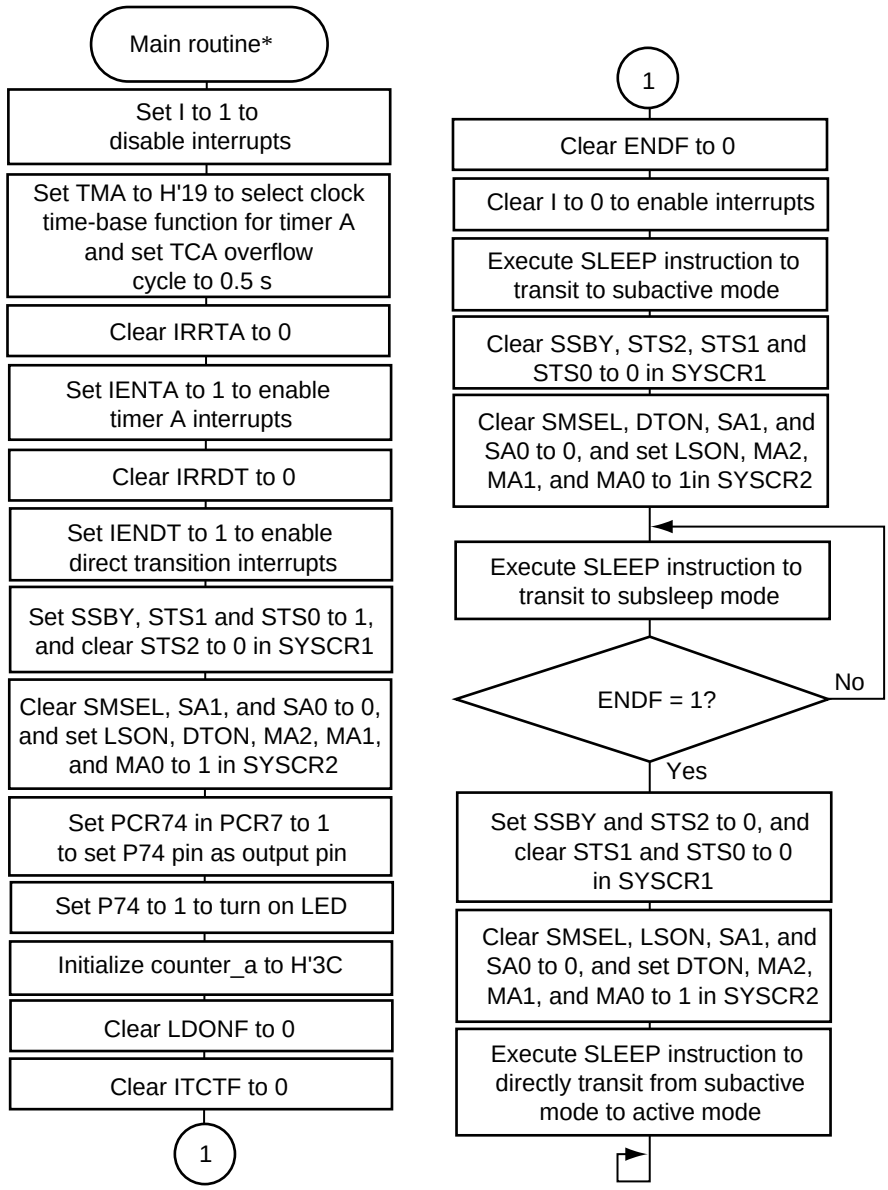
4.4 Description of RAM

Table 5 describes the RAM used in this sample task.

Table 5 Description of RAM

Label	Name	Function	Address	Used in
counter_a		Down-counter for counting the number of timer A interrupts	H'FB80	Main routine
USRF	ENDF	Flag for judging whether or not 60 s has elapsed	H'FB81 Bit 2	Main routine
	ITCNF	Flag for judging whether or not the timer A interrupt count is even or odd	H'FB81 Bit 1	LED control
	LDONF	Flag for judging on/off of the LED	H'FB81 Bit 0	LED control

5. Flowcharts



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

Figure 3 Flowchart for Main Routine

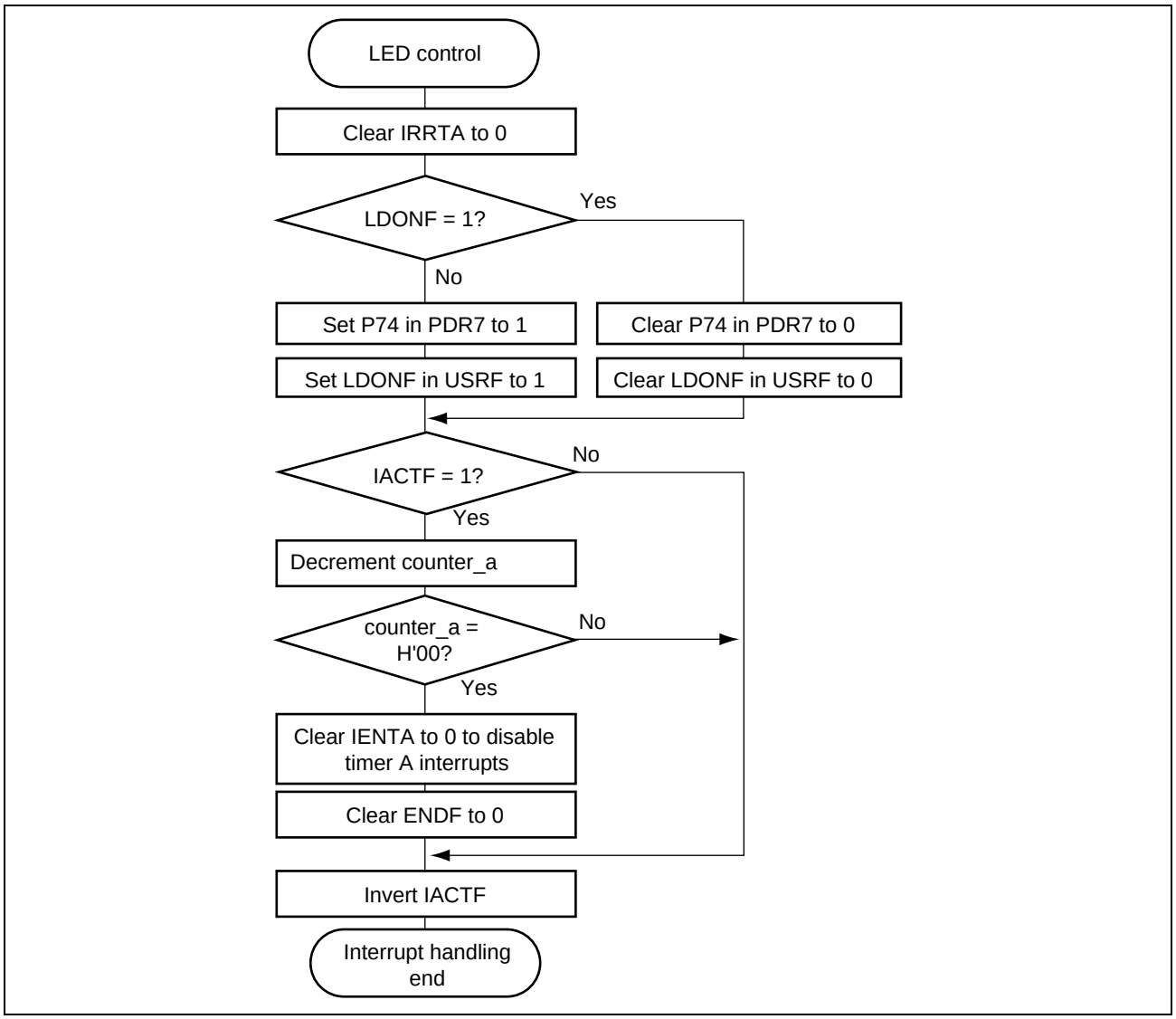


Figure 4 Flowchart for Timer A Interrupt Handling Routine

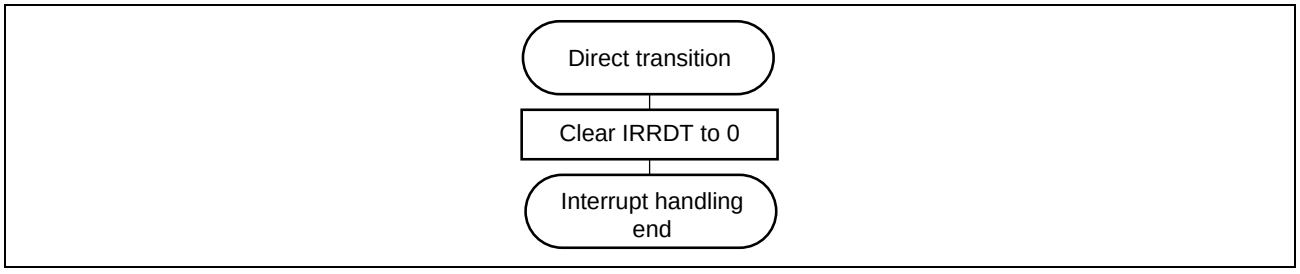


Figure 5 Flowchart for Direct Transition Interrupt Handling Routine

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
```

```
/*******************************************************/
/*                                                     */
/*    H8/300H Tiny Series -H8/3664-                    */
/*    Application Note                                 */
/*                                                     */
/*    'Transition to Sucsleep Mode'                    */
/*                                                     */
/*    Function                                         */
/*    : Power-Down Mode                                */
/*       Subsleep Mode                                 */
/*                                                     */
/*    External Clock : 16MHz                           */
/*    Internal Clock : 16MHz                           */
/*    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  TMA        *(volatile unsigned char *)0xFFA6 /* Timer Mode Register A                        */
#define  TCA        *(volatile unsigned char *)0xFFA7 /* Timer Counter A                              */
#define  PDR7_BIT   (*(struct BIT *)0xFFDA)           /* Port Data Register 7                         */
#define  P74        PDR7_BIT.b4                       /* Port Data Register 7 bit4                    */
#define  PCR7_BIT   (*(struct BIT *)0xFFEA)           /* Port Control Register 7                      */
#define  PCR74      PCR7_BIT.b4                       /* Port Control Register 7 bit4                 */
#define  SYSCR1     *(volatile unsigned char *)0xFFF0 /* System Control Register 1                    */
#define  SYSCR1_BIT (*(struct BIT *)0xFFF0)           /* System Control Register 1                    */
#define  SSBY       SYSCR1_BIT.b7                     /* Software Standby                             */
#define  STS2       SYSCR1_BIT.b6                     /* Standby Timer Select 2                       */
#define  STS1       SYSCR1_BIT.b5                     /* Standby Timer Select 1                       */
#define  STS0       SYSCR1_BIT.b4                     /* Standby Timer Select 0                       */
#define  NESEL      SYSCR1_BIT.b3                     /* Noise Elimination Sampling Frequency Select */
#define  SYSCR2     *(volatile unsigned char *)0xFFF1 /* System Control Register 2                    */
#define  SYSCR2_BIT (*(struct BIT *)0xFFF1)           /* System Control Register 2                    */
#define  LSON       SYSCR2_BIT.b6                     /* Low Speed On Flag                            */
#define  DTON       SYSCR2_BIT.b5                     /* Direct Transfer On Flag                      */
#define  MA1        SYSCR2_BIT.b3                     /* Active Mode Clock Select 1                   */
#define  MA0        SYSCR2_BIT.b2                     /* Active Mode Clock Select 0                   */
#define  SA1        SYSCR2_BIT.b1                     /* Subactive Mode Clock Select 1                */
#define  SA0        SYSCR2_BIT.b0                     /* Subactive Mode Clock Select 0                */
#define  IENR1_BIT  (*(struct BIT *)0xFFF4)           /* Interrupt Enable Register 1                  */
#define  IENDT      IENR1_BIT.b7                      /* Direct Transfer Interrupt Enable             */
#define  IENTA      IENR1_BIT.b6                      /* Timer A Interrupt Enable                     */
#define  IRR1_BIT   (*(struct BIT *)0xFFF6)           /* Interrupt Request Register 1                 */
#define  IRRDT      IRR1_BIT.b7                       /* Direct Transfer Interrupt Request Flag       */

#define  IRRTA      IRR1_BIT.b6                       /* Timer A Interrupt Request Flag               */

#pragma        interrupt    (dtint)
#pragma        interrupt    (taint)
```

```
/*******************************************************/
/*     Function Definition                             */
/*******************************************************/
extern    void    INIT ( void );                    /* SP Set                          */
void      main     ( void );
void      dtint    ( void );
void      taint    ( void );
void      sleep    ( void );

/*******************************************************/
/*    RAM define                                       */
/*******************************************************/
    unsigned char    counter_a;
    unsigned char    USRF;                            /* User Flag Erea                */

#define        USRF_BIT    (*(struct BIT *)&USRF)
#define        ENDF        USRF_BIT.b2                 /* End Flag                      */
#define        IACTF        USRF_BIT.b1                /* Timer A Interrupt Counter Flag */
#define        LDONF        USRF_BIT.b0                /* LED On Flag                   */

/*******************************************************/
/*    Vector Address                                   */
/*******************************************************/
#pragma    section        V1                           /* VECTOR SECTOIN SET            */
void (*const VEC_TBL1[])(void) = {
    INIT                                               /* 00 Reset                      */
};
#pragma    section        V2                           /* VECTOR SECTOIN SET            */
void (*const VEC_TBL2[])(void) = {
    dtint                                              /* Direct Transfer  Interrupt    */
};
#pragma    section        V3                           /* VECTOR SECTOIN SET            */
void (*const VEC_TBL3[])(void) = {
    taint                                              /* timer A  Interrupt            */
};
#pragma    section                                     /* P                             */
```

```
/*******************************************************/
/*    Main Program                                     */
/*******************************************************/
void    main ( void )
{

    set_imask_ccr(1);                           /* Interrupt Disable                        */

    TMA = 0x19;                                 /* Initialize Timer A Function              */

    IRRTA = 0;                                  /* Clear IRRTA                              */

    IENTA = 1;                                  /* Timer A Interrupt Enable                 */

    IRRDT = 0;                                  /* Clear IRRDT                              */

    IENDT = 1;                                  /* Direct Transfer Interrupt Enable         */

    SYSCR1 = 0xB0;                              /* Initialize Function of Subactive Mode 1  */
    SYSCR2 = 0x7C;                              /* Initialize Function of Subactive Mode 2  */

    P74 = 0;                                    /* Initialize P74                           */
    PCR74 = 1;                                  /* Initialize P74 Output Port               */

    counter_a = 0x3C;                           /* Initialize 8bit Timer A Interrupt Counter */

    LDONF = 0;                                  /* Initialize LDONF                         */

    IACTF = 0;                                  /* Initialize IACTF                         */

    ENDF = 0;                                   /* Initialize ENDF                          */

   set_imask_ccr(0);                            /* Interrupt Enable                         */

    sleep();                                    /* Transition to Subactive Mode             */

    SYSCR1 = 0x00;                              /* Initialize Function of Subsleep Mode 1   */
    SYSCR2 = 0x5C;                              /* Initialize Function of Subsleep Mode 2   */
```

```
    do{
        sleep();                                    /* Transion to Subsleep Mode                */
    }while(ENDF != 1);                              /* ENDF = "1"?                             */

    SYSCR1 = 0xC0;                                  /* Initialize Function of Active Mode 1     */
    SYSCR2 = 0x3C;                                  /* Initialize Function of Active Mode 2     */

    sleep();                                        /* Transition to Active Mode                */

    while(1){
       ;
    }
}

/*******************************************************/
/*     Timer A Interrupt                               */
/*******************************************************/
void taint ( void )
{

    IRRTA = 0;                                      /* Clear IRRTA                              */

    if(LDONF == 1){                                 /* LDONF = "1" ?                            */
        P74 = 0;                                    /* Turn Off LED                             */
        LDONF = 0;                                  /* Clear LDONF                              */
    }
        else{
            P74 = 1;                                /* Turn On LED                              */
            LDONF = 1;                              /* Set LDONF                                */
        }

    if(IACTF == 1){                                 /* IACTF = "1" ?                            */
        counter_a--;                                /* Decrement 8bit Timer A Interrupt Counter */
        if(counter_a == 0x00){                      /* 8bit Timer A Interrupt Counter = H'00 ?  */
            IENTA = 0;                              /* Timer A Interrupt Disable                */
            ENDF = 1;                               /* Set ENDF                                 */
        }
    }

    IACTF = ~IACTF;                                 /* Invert IACTF                             */

}
```

```
/******************************************************/
/*    Direct Transfer Interrupt                       */
/******************************************************/
void dtint ( void )
{

    IRRDT = 0;                                  /* Clear IRRDT                          */

}
```

Link Address Setting:

Section Name	Address
CV1	H'0000
CV2	H'001A
CV3	H'0026
P	H'0100
B	H'FB80

Website and Support

Renesas Technology Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/inquiry
csc@renesas.com

Revision Record

		Description	
Rev.	**Date**	**Page**	**Summary**
2.00	Sep.01.06	All pages	Format has been changed from Hitachi version to Renesas version.

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.