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 Super Low Power Series

Measuring Pulse Frequency by Event Counting

Introduction

Using the Timer C 8-bit event counter function, this function measures the pulse frequency input through Timer C Event Input Pin (TMIC).

Target Device

H8/38024

Contents

1. Specifications

1. Using the Timer C 8-bit event counter function, this function measures the pulse frequency input through Timer C Event Input Pin (TMIC).
2. Counting rising edge detection operations of a pulse input through TMIC input pin for 1 s, this function stores the pulse count for 1 s into the RAM.
3. The 1-s measurement time is measured using the Timer A clock time-base function.
4. The Timer Counter C (TCC) is set to the up-counter controlled by hardware with UD pin which is connected to GND.

2. Description of Functions Used

1. In this task sample, the Timer C 8-bit event counter function is used to measure the frequency of a pulse input at TMIC input pin.
   a. The block diagram of the Timer C 8-bit event counter function is shown in figure 1 and is described below .
      - Timer Mode Register C (TMC) is an 8-bit read/write register which selects the interval function, controls up/down for Timer Counter C (TCC), and selects the input clock. It can be selected whether TCC up/down control is performed by hardware using UD pin input, or whether TCC functions as an up-counter or a down-counter set by software control.
      - Timer Counter C (TCC) is an 8-bit read-only counter which is counted up/down by an internal clock/external event which is input. The input clock can be selected from a total of eight clocks, namely, clocks obtained by dividing the system clock by 8192, 2048, 512, 64, 16 and 4, and subclock/4, and an external clock. In this sample task, TCC is set to up-counter controlled by hardware, and edge detection of TMIC input pin is selected as the TCC input clock.
      - Timer C Interrupt Request Flag (IRRTC) is set to 1 when TCC overflows. A Timer C interrupt is accepted and Timer C interrupt handling is started when IRRTC is set to 1, Timer C interrupt enable (IENTC) in Interrupt Enable Register 2 (IENR2) is set to 1, and the I bit in Condition Code Register (CCR) is cleared to 0.
      - Timer C Event Input Pin (TMIC) functions as the input pin of a pulse whose frequency is measured.

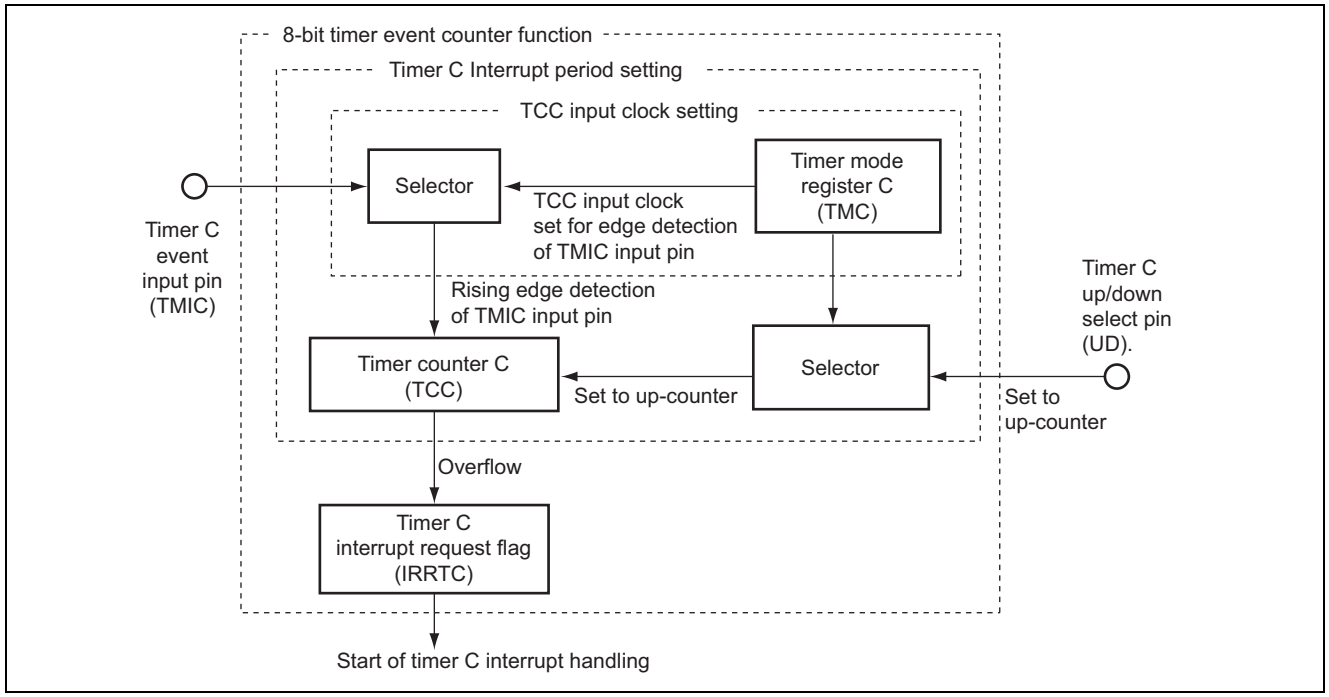


Figure 1 Block Diagram of Timer C Event Counter Function

b. The method to measure frequencies is described below.
- Inputting 256 rising edges to TMIC input pin, TCC overflows to generate a Timer C interrupt.
- The value of the 8-bit counter cnt1 is incremented during Timer C interrupt handling.
- When 1 s has elapsed, the TCC count value is stored in cnt2 and TCC counting up the pulse frequency at the TMIC input pin is completed.
- Frequencies of pulses input to TMIC pin is given by the equation below:

$$\text{Input pulse frequency (Hz)} = (\text{Timer C interrupts}) \times 256 + (\text{Counter value of TCC after 1 s})$$
$$= (\text{cnt1 value}) \times 256 + (\text{cnt2 value})$$

- The counter (cnt1) to count Timer C interrupts is an 8-bit counter, so frequencies of input pulses that can be measured should be 65.535 kHz maximum.
- When the 8-bit counter (cnt1) to count Timer C interrupts overflows, frequency measurement will be stopped immediately and operation ends by writing H'00 in cnt1 and the register (cnt2) which stores the counter value of TCC when 1 s has elapsed.

2. Table 1 shows function assignment in this sample task. The functions are assigned as shown in table 1 and frequencies are measured by the Timer C event counter function.

Table 1 Assignment of Functions

Function	Assignment
PSW	A 5-bit counter using a clock obtained by dividing 32.768 kHz by 4 as input
TMA	Selects PSW and sets the TCA overflow period.
TCA	An 8-bit counter using a clock obtained by dividing 32.768 kHz by 128 as input
TMC	Sets interval function, sets TCC to up-counter by hardware, and sets TCC input clock as edge detection of TMIC input pin
TCC	An 8-bit counter using edge detection of TMIC input pin as input
TLC	Sets reload value of TCC overflow to H”00
$\overline{\text{IRQ1}}$	Sets PB3/AN3/$\overline{\text{IRQ1}}$ pin to TMIC input pin
IEG1	Sets input sense of the TMIC pin to rising edge detection.
IENTA	Enables Timer A interrupt requests
IENTC	Enables Timer C interrupt requests
IRRTA	Indicates whether or not a Timer A interrupt is requested.
IRRTC	Indicates whether or not a Timer C interrupt is requested.
TMIC	Input pin of pulses whose frequency is measured.
UD pin	An input pin to set TCC to up-counter

3. Principle of Operation

1. Figure 2 illustrates the principle of operation of this sample task. As shown in figure 2, the pulse frequency is measured using Timer C 8-bit event counter function by hardware processing and software processing.

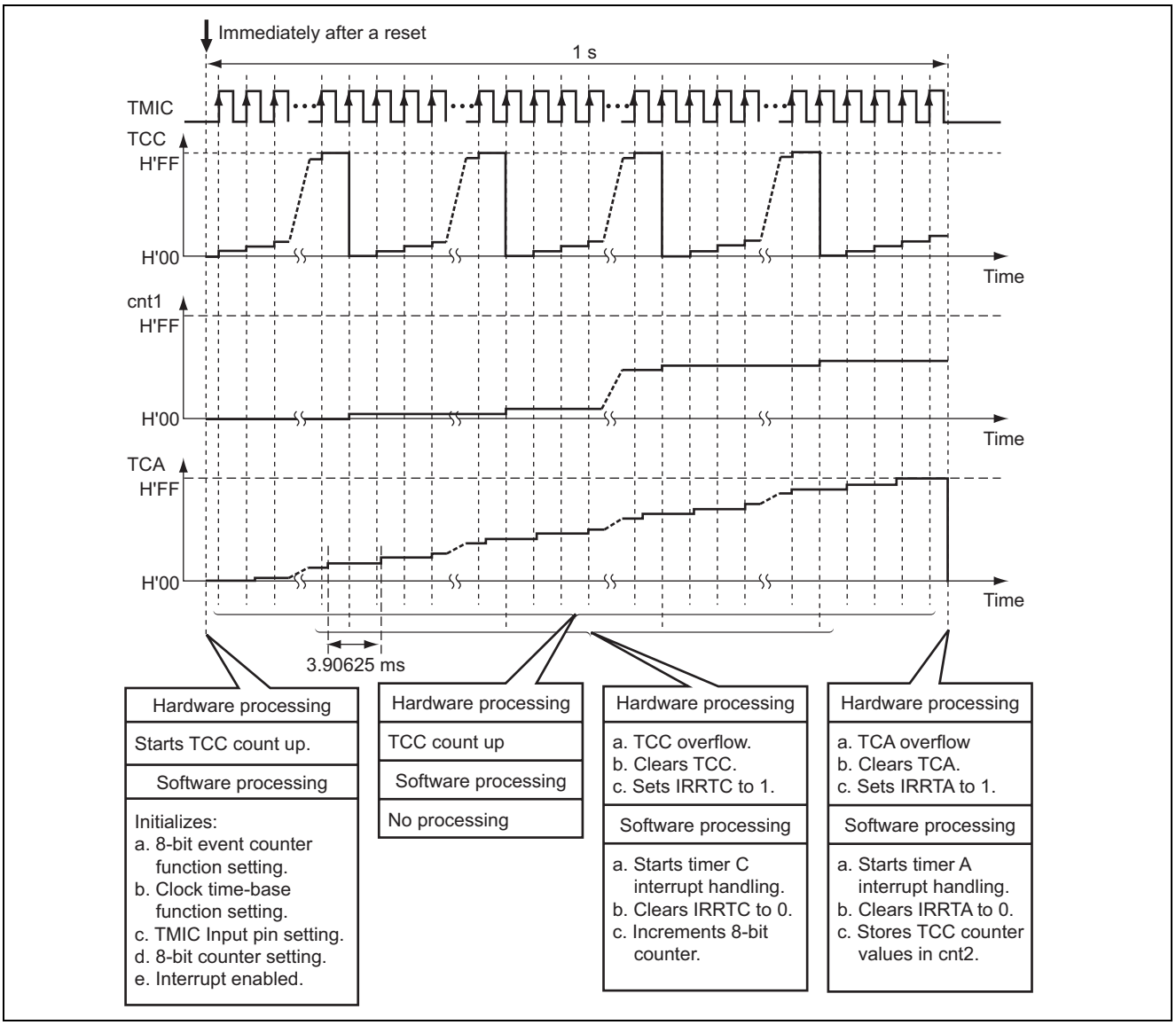


Figure 2 Operation Principle of Frequency Measurement by Timer C 8-bit Event Counter Function

4. Description of Software

4.1 Modules

Table 2 describes the modules in this sample task.

Table 2 Description of Modules

Module	Label	Function
Main Routine	main	Sets Timer C 8-bit event counter function, clock time-base function, the 8-bit counter, enables interrupts, and initializes Timer C after completing measurement.
After 1s	taint	Disables interrupts and stores TCC counter value in cnt2 during Timer A interrupt handling after 1s.
8-Bit Counter	tcint	Increments the 8-bit counter value and processes for cnt1 overflow during Timer C interrupt handling.

4.2 Arguments

Arguments used in this sample task are described in table 3.

Table 3 Description of Internal Registers

Argument	Function	Used in	Data Length	Input/ Output
cnt1	Stores counter value of 8-bit counter after 1 s.	8-bit counter	1 byte	Output
cnt2	Stores TCC counter value after 1 s.	After 1 s	1 byte	Output

4.3 Internal Registers

Table 4 describes the internal registers in this sample task.

Table 4 Description of Internal Registers

Register		Function	Address	Setting
TMA		Timer Mode Register A When TMA = H'18, the Timer A function is set to clock time-base function, TCA input clock source is set to PSW and TCA overflow period to 1s.	H'FFB0	H'18
TCA		Timer Counter A An 8-bit up-counter using a clock obtained by dividing 32.768 kHz by 128 as input	H'FFB1	H'00
TMC		Timer Mode Register C When TMC = H'7F, the Timer C function is set to the interval function, TCC is set to up-counter controlled by hardware, and the TCC input clock is set to input edge detection of TMIC pin.	H'FFB4	H'7F
TCC		Timer Counter C An 8-bit up-counter using input edge detection of TMIC pin as input	H'FFB5	H'00
TLC		Timer Load Register C When TLC = H'00, TCC starts counting up from H'00 and when TCC overflows, H'00 is loaded to TCC.	H'FFB5	H'00
PMR3	UD	Port Mode Register 3 (P30/UD pin switch) When UD = 0, P30/UD functions as P30 input/output pin. When UD = 1, P30/UD functions as UD input pin.	H'FFCA Bit 0	1
PMRB	IRQ1	Port Mode Register B (PB3/AN3/$\overline{\text{IRQ1}}$ pin switch) When $\overline{\text{IRQ1}}$ = 0, PB3/AN3/$\overline{\text{IRQ1}}$ functions as PB3/AN3 input pin. When $\overline{\text{IRQ1}}$ = 1, PB3/AN3/$\overline{\text{IRQ1}}$ functions as $\overline{\text{IRQ1}}$/TMIC input pin.	H'FFEE Bit 3	1
IEGR	IEG1	IRQ Edge Select Register (IRQ1 Edge Select) When IEG1 = 0, edge detection of TMIC input pin is set to falling edge detection. When IEG1 = 1, edge detection of TMIC input pin is set to rising edge detection.	H'FFF2 Bit 1	1
IENR1	IENTA	Interrupt Enable Register 1 (Timer A Interrupt Enable) When IENTA = 0, Timer A interrupt request is disabled. When IENTA = 1, Timer A interrupt request is enabled.	H'FFF3 Bit 7	1
IENR2	IENTC	Interrupt Enable Register 2 (Timer C Interrupt Enable) When IENTC = 0, Timer C interrupt request is disabled. When IENTC = 1, Timer C interrupt request is enabled.	H'FFF4 Bit 1	1
IRR1	IRRTA	Interrupt Request Register 1 (Timer A Interrupt Request Flag) When IRRTA = 0, Timer A interrupt is not requested. When IRRTA = 1, Timer A interrupt is requested.	H'FFF6 Bit 7	0
IRR2	IRRTC	Interrupt Request Register 2 (Timer C Interrupt Request Flag) When IRRTC = 0, Timer C interrupt is not requested. When IRRTC = 1, Timer C interrupt is requested.	H'FFF7 Bit 1	0

4.4 RAM

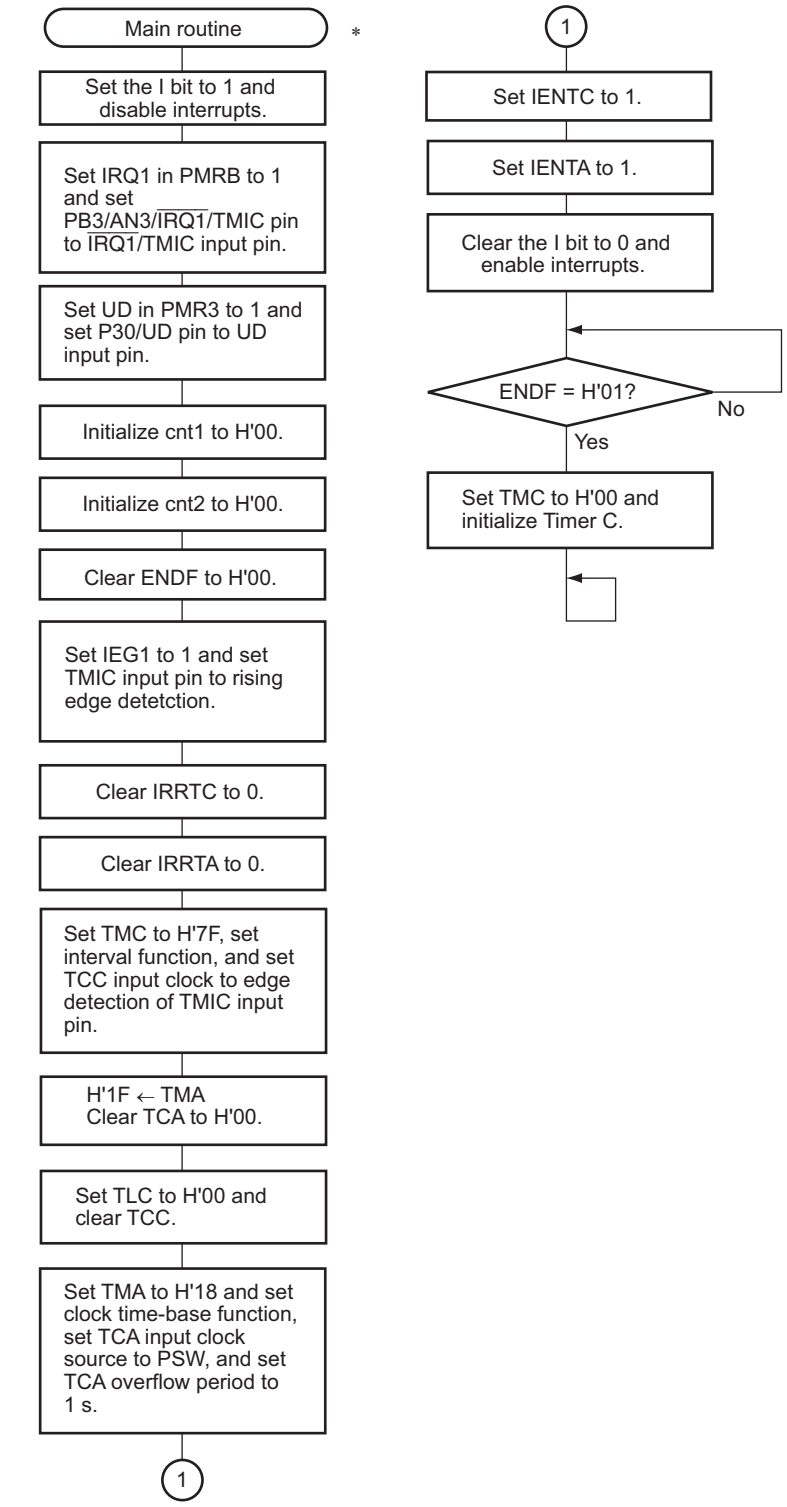
Table 5 describes the RAMs used in this sample task.

Table 5 Description of RAM Used

Label	Function	Address	Used in
cnt1	Stores the counter value of 8-bit counter after 1 s.	H'FB80	8-bit Counter
cnt2	Stores the counter value of TCC after 1 s.	H'FB81	After 1 s
ENDF	Data to indicate whether or not input pulse frequency measurement is completed. When measuring time is under 1 s and cnt1 < H'FF, ENDF = H'00 When 1 s has elapsed in measuring or cnt1 == H'FF, ENDF = H'01	H'FB82	Main Routine 8-bit Counter After 1 s

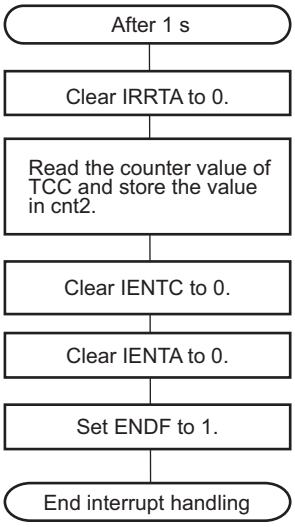
5. Flowchart

1. Main routine

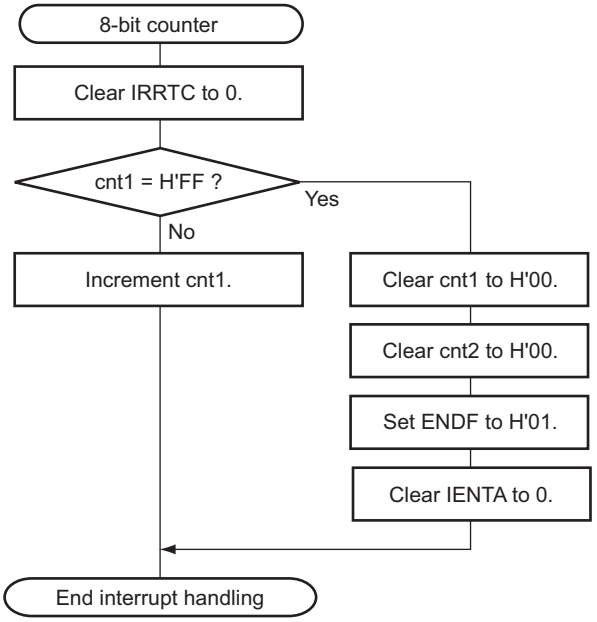


Note: * In this sample task, the stack pointer is set in INIT.SRC (assembly language).

2. Timer A interrupt handling routine



3. Timer C 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                                                                            */
/*                                                                                              */
/*  'Pulse Frequency Measurement by Event                                                       */
/*   Counter Function'                                                                          */
/*                                                                                              */
/*  Function                                                                                    */
/*  :Timer C 8bit Event Counter                                                                 */
/*                                                                                              */
/*  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      TMA         *(volatile unsigned char *)0xFFB0   /* Timer Mode Register A            */
#define      TCA         *(volatile unsigned char *)0xFFB1   /* Timer Counter A                  */
#define      TMC         *(volatile unsigned char *)0xFFB4   /* Timer Mode Register C            */
#define      TCC         *(volatile unsigned char *)0xFFB5   /* Timer Counter C                  */
#define      TLC         *(volatile unsigned char *)0xFFB5   /* Timer Load Register C            */
#define      PMR3_BIT    (*(struct BIT *)0xFFCA)             /* Port Mode Register 3             */
#define      UD          PMR3_BIT.b0                         /* Port Mode Register 3 bit0        */
#define      PMRB_BIT    (*(struct BIT *)0xFFEE)             /* Port Mode Register B             */
#define      IRQ1        PMRB_BIT.b3                         /* Port Mode Register B bit3        */
#define      IEGR_BIT    (*(struct BIT *)0xFFF2)             /* Interrupt Edge Select Register 1 */
#define      IEG1        IEGR_BIT.b1                         /* IRQ1 Edge Select                 */
#define      IENR1_BIT   (*(struct BIT *)0xFFF3)             /* Interrupt Enable Register 1      */
#define      IENTA       IENR1_BIT.b7                        /* Timer A Interrupt Enable         */
#define      IENR2_BIT   (*(struct BIT *)0xFFF4)             /* Interrupt Enable Register 2      */
#define      IENTC       IENR2_BIT.b1                        /* Timer C Interrupt Enable         */
#define      IRR1_BIT    (*(struct BIT *)0xFFF6)             /* Interrupt Request Register 1     */
#define      IRRTA       IRR1_BIT.b7                         /* Timer A Interrupt Request Flag   */
#define      IRR2_BIT    (*(struct BIT *)0xFFF7)             /* Interrupt Request Register 2     */
#define      IRRTC       IRR2_BIT.b1                         /* Timer C Interrupt Request Flag   */

#pragma interrupt        (taint)
#pragma interrupt        (tcint)
/************************************************************************************************/
/*  Function                                                                                    */
/************************************************************************************************/
extern void  INIT ( void );                                  /* SP Set                           */
void         main ( void );
void         taint ( void );
void         tcint ( void );

/************************************************************************************************/
/*  RAM define                                                                                  */
/************************************************************************************************/
unsigned char            cnt1;                               /* 8bit Counter                     */
unsigned char            cnt2;                               /* 8bit Counter                     */
unsigned char            ENDF;                               /* End Data                         */

/************************************************************************************************/
/*  Vector Address                                                                              */
/************************************************************************************************/
#pragma section          V1                                  /* Vector Section                   */
void (*const VEC_TBL1[])(void) = {                           /* 0x0000 - 0x000F                  */
    INIT                                                     /* 0x0000 Reset Vector              */
};
#pragma section          V2                                  /* Vector Section Set               */
void (*const VEC_TBL2[])(void) = {
    taint                                                    /* 0x0016 Timer A Interrupt Vector  */
};
#pragma section          V3                                  /* Vector Section Set               */
void (*const VEC_TBL3[])(void) = {
    tcint                                                    /* 0x001A Timer C Interrupt Vector  */
};

#pragma section                                              /* P                                */
```

```
/*****************************************************************************************/
/*  Main Program                                                                         */
/*****************************************************************************************/
void main ( void )
{
    set_imask_ccr(1);                           /* Interrupt Disable                    */

    IRQ1 = 1;                                   /* TMIC Input Select                    */
    UD   = 1;                                   /* UD Input Select                      */

    cnt1 = 0;                                   /* Initialize 8bit  Counter             */
    cnt2 = 0;                                   /* Initialize 8bit  Counter             */
    ENDF = 0;                                   /* Clear END Flag                       */

    IEG1 = 1;                                   /* TMIC Select                          */
    IRRTC = 0;                                  /* Clear IRRTC                          */
    IRRTA = 0;                                  /* Clear IRRTC                          */

    TMC = 0x7F;                                 /* Initialize 8bit Event Counter Function */
    TMA = 0x1F;                                 /* Initialize Timer Counter A           */
    TLC = 0x00;                                 /* Clear TCC                            */
    TMA = 0x18;                                 /* Initialize Clock Time Base Function  */

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

    set_imask_ccr(0);                           /* Interrupt Enable                     */

    while (ENDF =  = 0);

    TMC = 0;                                    /* Initialize Timer C Function          */

    while (1) {
        ;
    }
}

/*****************************************************************************************/
/*  Timer A Interrupt                                                                    */
/*****************************************************************************************/
void taint ( void )
{
    IRRTA = 0;                                  /* Clear IRRTA                          */

    cnt2  = TCC;                                /* Store TCC                            */

    IENTC = 0;                                  /* Timer C Interrupt Disable            */

    IENTA = 0;                                  /* Timer A Interrupt Disable            */

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

```
/*****************************************************************************************************/
/*  Timer C Interrupt                                                                                */
/*****************************************************************************************************/
void tcint ( void )
{
    IRRTC = 0;                                          /* Clear IRRTC                  */

    if ( cnt1 ==  0xff ) {                              /* 8bit Counter = 0xff?         */
        cnt1 = 0;                                       /* Clear cnt1                   */
        cnt2 = 0;                                       /* Clear cnt2                   */
        ENDF = 1;                                       /* ENDF                         */
        IENTA = 0;                                      /* Timer A Interrupt Disable    */
    }
    else {
        cnt1++;                                         /* cnt1 Increment               */
    }
}
```

Link address specifications

Section Name	Address
CV1	H'0000
CV2	H'0016
CV3	H'001A
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**
1.00	Dec.19.03	—	First edition issued
2.00	Nov.30.06	All pages	Content correction

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, 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 products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (http://www.renesas.com)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
   (1) artificial life support devices or systems
   (2) surgical implantations
   (3) healthcare intervention (e.g., excision, administration of medication, etc.)
   (4) any other purposes that pose a direct threat to human life

   Renesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas 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 applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.