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.

32176 Group

Application of Timer TIO (Measurement Clear Input Mode)

1. Overview

The following article shows sample program of 32176 group using timer TIO.

2. Introduction

The sample task described here uses the following microcomputer, under the respective conditions.

- Microcomputer: 32176 Group (M32176FnVFP, M32176FnTFP)
- Operating frequency: 20 to 40MHz (The sample program is compiled assuming a frequency of 40 MHz.)
- Operating Board: Starter kit for 32176 Group

3. Explanation of an applied technology

3.1 Outline of Multijunction Timers

The multijunction timers (abbreviated MJT) have input event buses and output event buses. Therefore, in addition to be used as a single unit, the timers can be internally connected to each other. This capability allows for highly flexible timer configuration, making it possible to meet various applications needs. It is because the timers are connected to internal event bus at multiple points that they are called the “multijunction” timers.

MJT is detailed in 32176 Group User's Manual.

4. Measurement Clear Input Mode Sample Program

4.1 Outline of the sample program

In this sample program, TIO4 is used and the clock bus 0 is counted as count source.
The external capture signal inputted from TIN0 is inputted into TIO4 though input event bus 2 course, and the time of rising edge is measured. A measurement result is read out within the interrupt processing started by the rising edge of TIN0. The read value is outputted from a port 10 and a port 11 as 16-bit data.

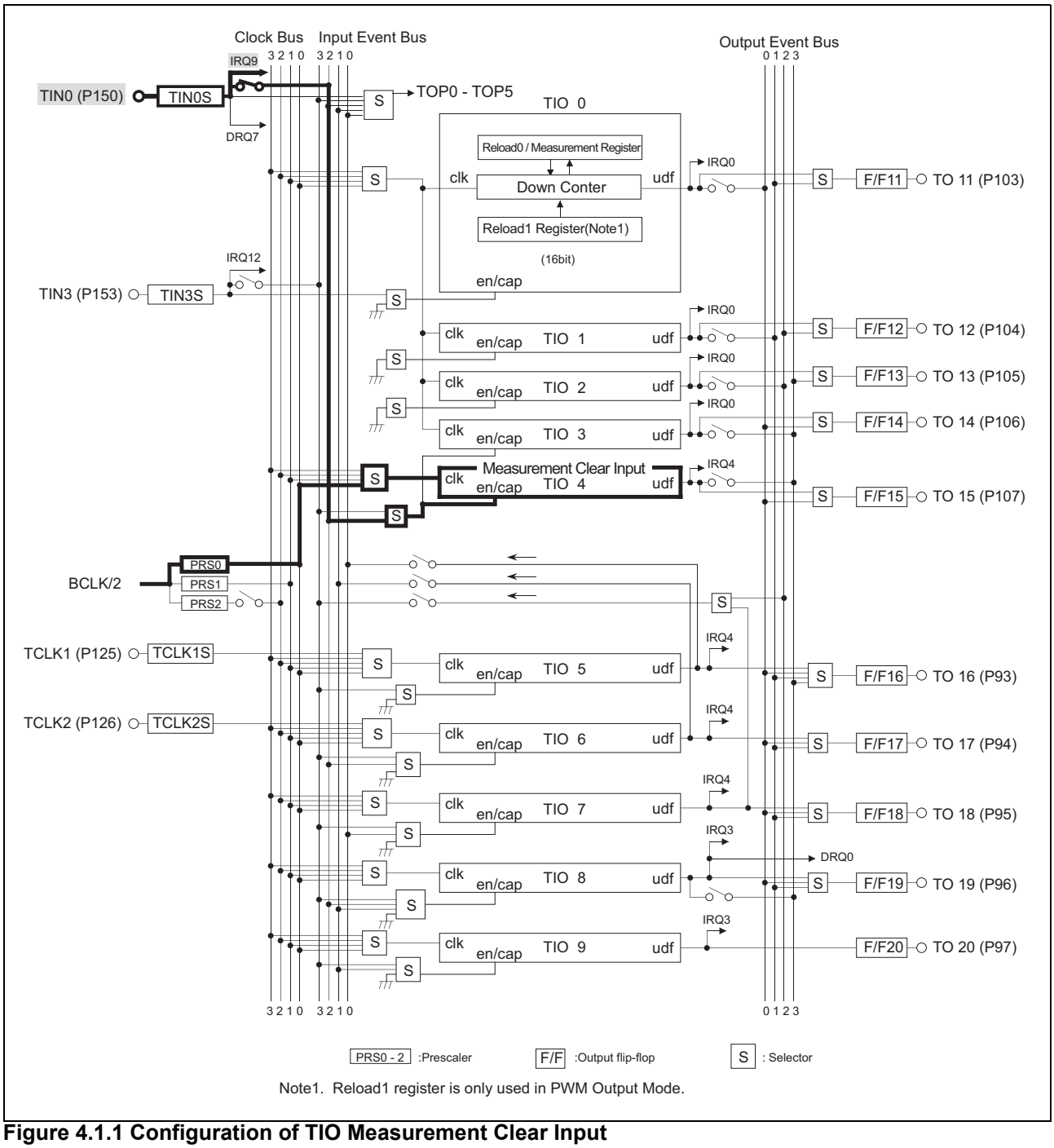

Figure 4.1.1 Configuration of TIO Measurement Clear Input

4.2 Processing procedure

The basic processing flow of a timer setup is shown in Figure 4.2.1.

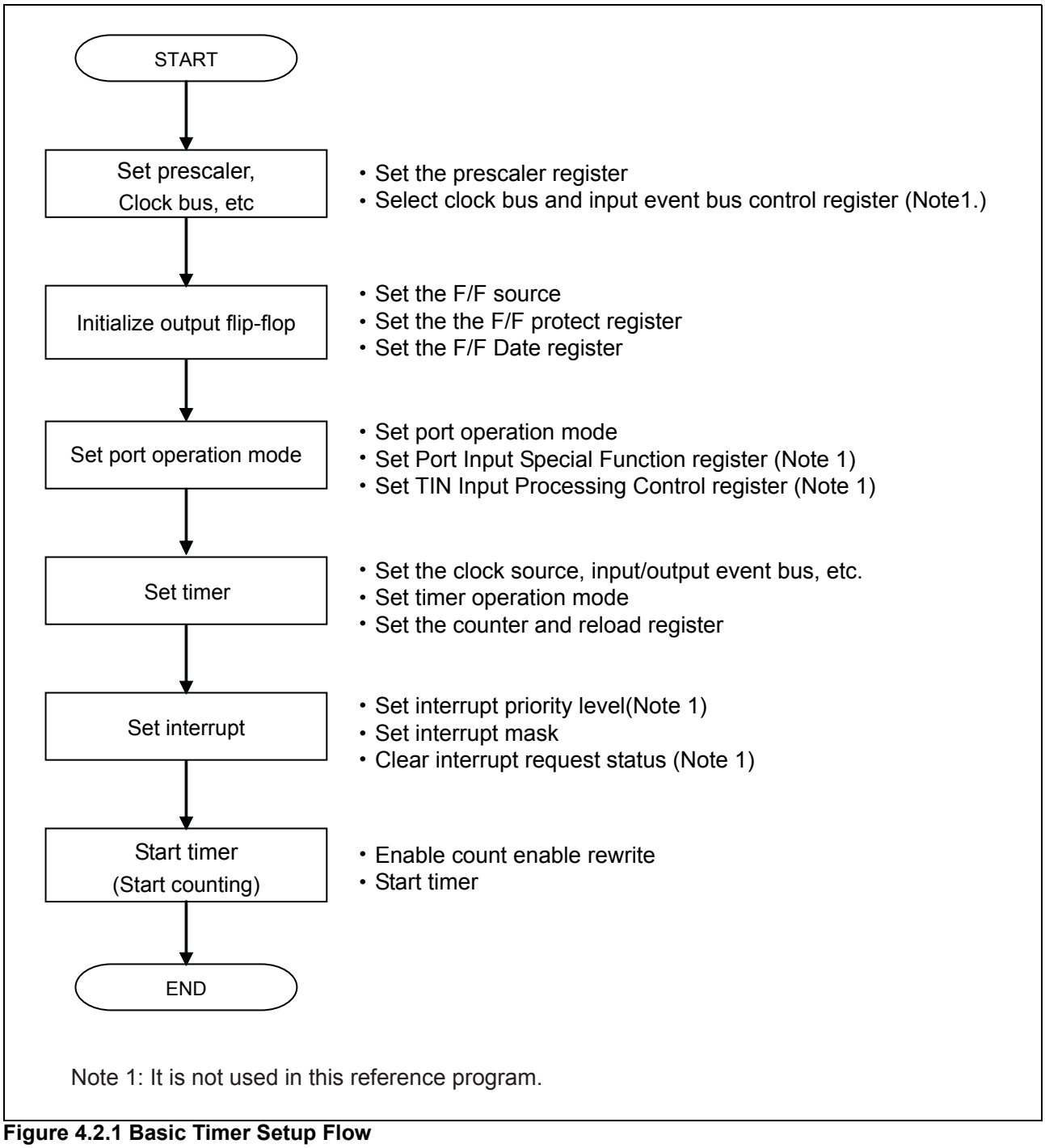

Figure 4.2.1 Basic Timer Setup Flow

4.3 Description of a reference program

Note. The registers used are indicated as (register name: bit name)

4.3.1 Interrupt setting function (int_init())

(1) Set the Interrupt priority level of TIO4 output

- Set the priority level of MJT Input Interrupt Control Register 1 (TIN0 Interrupt) as 0 (top priority). (IMJTICR1: ILEVEL)

4.3.2 Timer initialization function (timer_init())

(1) Set the prescaler's divide-by value

- Set prescaler register 0 as “the prescaler's divide-by value-1”. (PRS0)

4.3.3 Port initialization function (port_init())

(1) Initial setting of output port

- Set Port input enable bit of Port Input Special Function Control Register as permit inputting. (PICNT: PIEN0)
- Initialize P10 Data Register. (P10DATA)
- Set P10 Direction Register as output mode. (P10DIR)
- Set P10 Operation Mode Register as input/output port. (P10MOD)
- Initialize P11 Data Register. (P11DATA)
- Set P11 Direction Register as output mode. (P11DIR)
- Set P11 Operation Mode Register as input/output port. (P11MOD)

Note. If a Direction Register is set as output before setting up a Data Register, an unfixed value is outputted until writing will be performed to a Data Register.

4.3.4 Various initialization function (init_func())

(1) Call the port initialization function
(2) Call the timer initialization function
(3) Call the Interrupt Setting function

4.3.5 Main function (main())

(1) Call the interrupt disable function
(2) Call the various initialization function
(3) Call the TIO4 Measure clear input mode initial setting function
(4) Call the interrupt enable function
(5) Infinite loop waiting for Interrupt

4.3.6 TIN0 to TIN2 Input interrupt processing function (TIN0_2_Int())

(1) TIN0 Interrupt request status check

- Call the interrupt disable function.
- Clear TIN0 Interrupt Request Status.
- Call the interrupt enable function.
- Call TIN0 interrupt processing function.

4.3.7 TIO4 Measure clear input mode initial setting function (TIO4_MesClr_init())

(1) Set Input Event Bus
- Select TIN0 when inputs in Input Event Bus 2. (CKIEBCR: IEB2S)

(2) Initial setting TIO4
- Set Measurement Clear Input Mode. (TIO4CR: TIO4M)
- Set Clock Bus 0 to Clock sources. (TIO4CR: TIO4CKS)
- Set External terminal as permission. (TIO4CR: TIO4EEN)
- Set Input Event Bus 2 to Measurement Input Sources. (TIO4CR: TIO34ENS)

(3) Initial setting Input terminal
- Set TIN0 as availableness of rising edge. (TINCR0: TIN0S)
- Set P150 Operation Mode bit of P15 Operation Mode Register as TIN0. (P15MOD: P150MOD)

(4) Set Interrupt
- Set TIO4 Interrupt Request as disable. (TIOIR1: TIOIM4)
- Set TIN0 Interrupt Request as enable. (TINIR0: TINIM0)

(5) Initialize TIO4 Counter
- Initialize Counter with H'FFFF.

(6) Start counting TIO4
- Set Enable protect bit as enable for rewriting. (TIOPRO: TIO4PRO)
- Start counting. (TIOCEN: TIO4CEN)

4.3.8 TIN0 (TIO4 Measurement Input) Interrupt Processing Function (TIN0_Int())

(1) Read out the result of TIO4 measurement
- Call the interrupt disable function.
- Read out Measurement register.

(2) Judgment of TIO4 Underflow
- It is confirmed whether Underflow has been done in TIO4 Interrupt Request status.
- When Underflow is being done, TIO4 Interrupt Request status is cleared, and H'10000 is added to a measurement result.

(3) Call the interrupt enable function

4.3.9 Startup routine (startup.ms)

(1) Set interrupt
- Set Interrupt sources of ICU Vector Table, which is MJT input interrupt 1 (H'0000 00A0), as the head address of interrupt processing function (TIN0_2_Int())

4.4 Sample Programming Code

The sample program of TIO4 Measurement Clear Input Mode is shown below.
In this sample program, TIO4 is used and measures time until rising edge of the signal inputted into TIN0. And also it is measurable till twice as many times (17 bits) as counter width by checking an Underflow interrupt request. The result of measurement is outputted from port 10 and 11 as 16 bit data.

Note that the sample program below requires the SFR definition file. The latest SFR definition file can be downloaded from Renesas Technology website. When using the SFR definitions file, adjust the path setting to match the operating computer environment.

4.4.1 TIO4_mes_main.c

```
/*""FILE COMMENT""*****************************************************
 *      M32R C Programming              Rev. 1.01
 *               < Sample Program for 32176 >
 *               < TIO4 measure clear input (main routine) >
 *
 *      Copyright (c) 2004 Renesas Technology Corporation
 *                         All Rights Reserved
 ***********************************************************************/

/************************************************************************/
/*              Include file                                            */
/************************************************************************/

#include                "..\inc\sfr32176_pragma.h"

/************************************************************************/
/*              Function prototype declaration                          */
/************************************************************************/

        void            main(void);                             /* Main function */
        void            init_func(void);                        /* Initial setup function */
        void            port_init(void);                        /* Initialize port */
        void            timer_init(void);                       /* Timer initialization */
        void            int_init(void);                         /* Initialize ICU */
        void            TIN0_2_Int(void);                       /* Process TIN0-2 input interrupt */

/************************************************************************/
/*              Definition of external reference                        */
/************************************************************************/

extern  void            DisInt( void );                         /* Interrupt disable function */
extern  void            EnInt( void );                          /* Interrupt enable function */

extern  void            TIO4_MesClr_init( void );              /* Initialize TIO4 measurement clear input
mode */
extern  void            TIN0_Int(void);                         /* Process TIN0 (TIO4 measurement input)
interrupt */

/************************************************************************/
/*              Global variable                                         */
/************************************************************************/

        volatile ULONG  MesTime;                                /* Measurement count value */

/*""FUNC COMMENT""*****************************************************
 * Function name: int_init()
 *----------------------------------------------------------------------
 * Description  : Initialize ICU
 *----------------------------------------------------------------------
 * Argument     : -
 *----------------------------------------------------------------------
 * Returns      : -
 *----------------------------------------------------------------------
 * Notes        :
 *""FUNC COMMENT END""************************************************/
void int_init(void)
{
        IMJTICR1 = 0x00;                                        /* TINO-2 ILEVEL = 0 */
}

/*""FUNC COMMENT""*****************************************************
 * Function name: timer_init()
 *----------------------------------------------------------------------
 * Description  : Timer initialization
 *----------------------------------------------------------------------
 * Argument     : -
 *----------------------------------------------------------------------
 * Returns      : -
 *----------------------------------------------------------------------
```

```
 * Notes        :
 *""FUNC COMMENT END""***************************************************/
void timer_init(void)
{
        PRS0 = ( 100 - 1);                                      /* Set prescaler(10us@10MHz) */
}

/*""FUNC COMMENT""*********************************************************
 * Function name: port_init()
 *-------------------------------------------------------------------------
 * Description  : Initialize port
 *-------------------------------------------------------------------------
 * Argument     : -
 *-------------------------------------------------------------------------
 * Returns      : -
 *-------------------------------------------------------------------------
 * Notes        : -
 *""FUNC COMMENT END""***************************************************/
void port_init(void)
{
        PICNT = PIEN0;                                          /* Enable port input */

        P10DATA = 0x00;                                        /* Output data (must be set prior to mode) */
        P10DIR = 0xff;                                          /* P100-P107 : Output mode */
        P10MOD = 0x00;                                          /* P100-P107 : Input/output port */

        P11DATA = 0x00;                                        /* Output data (must be set prior to mode) */
        P11DIR = 0xff;                                          /* P110-P117 : Output mode */
        P11MOD = 0x00;                                          /* P110-P117 : Input/output port */
}

/*""FUNC COMMENT""*********************************************************
 * Function name: init_func()
 *-------------------------------------------------------------------------
 * Description  : Call various initialization functions
 *-------------------------------------------------------------------------
 * Argument     : -
 *-------------------------------------------------------------------------
 * Returns      : -
 *-------------------------------------------------------------------------
 * Notes        : -
 *""FUNC COMMENT END""***************************************************/
void init_func(void)
{
        port_init();                                            /* Initialize those related to port */
        timer_init();                                           /* Initialize those related to timer */
        int_init();                                             /* Initialize those related to int */
}

/*""FUNC COMMENT""*********************************************************
 * Function name: main()
 *-------------------------------------------------------------------------
 * Description  : Initialize those related to intWhile using TIO4 in measurement clear input mode,
 *              : this function measures the duration of a rising edge on the input signal entered
 *              : from the TIN0 pin. The 16 low-order bits of the measurement result are output from ports
10 and 11.
 *              : The measurement resolution is 10 us (when the source clock frequency = 10 MHz).
 *              :
 *-------------------------------------------------------------------------
 * Argument     : -
 *-------------------------------------------------------------------------
 * Returns      : -
 *-------------------------------------------------------------------------
 * Notes        :
 *""FUNC COMMENT END""***************************************************/
void main(void)
{
        volatile USHORT  *P10P11DATA;

        P10P11DATA = (USHORT *)&P10DATA;
        MesTime = 0ul;

        DisInt();                                               /* Disable interrupt */

        init_func();

        TIO4_MesClr_init();                                     /* T104 count start */

        EnInt();                                                /* Enable interrupt */

        while(1) {
                *P10P11DATA = MesTime;
        }
}

/*""FUNC COMMENT""*********************************************************
 * Function name: TIN0_2_Int()
```

```
154   *-----------------------------------------------------------------------
155   * Description  : Enable interruptProcess TIN0-2 input interrupt
156   *              : If a TIN0 interrupt, clear request status and process the TIN0 interrupt.
157   *-----------------------------------------------------------------------
158   * Argument     : -
159   *-----------------------------------------------------------------------
160   * Returns      : -
161   *-----------------------------------------------------------------------
162   * Notes        : -
163   *""FUNC COMMENT END""***************************************************/
164  void TIN0_2_Int(void)
165  {
166  /*** Interrupt judgment (TIN0) ***/
167
168          if(( TINIR0 & TINIS0) != 0u) {
169                  DisInt();                                       /* Disable interrupt */
170                  TINIR0 &= ~TINIS0;                              /* Clear TIN0 interrupt request */
171                  EnInt();                                        /* Enable interrupt */
172
173                  TIN0_Int();                                     /* Process TIN0 interrupt */
174          }
175  }
```

4.4.2 TIO4_mes.c

```
/*""FILE COMMENT""*****************************************************
 *      M32R C Programming              Rev. 1.01
 *              < Sample Program for 32176 >
 *              < TIO4 measure clear input mode >
 *
 *      Copyright (c) 2004 Renesas Technology Corporation
 *                      All Rights Reserved
 ***********************************************************************/

/************************************************************************/
/*              Include file                                            */
/************************************************************************/

#include                "..\inc\sfr32176_pragma.h"

/************************************************************************/
/*              Externally referenced variable                          */
/************************************************************************/

extern  volatile ULONG  MesTime;                                /* Measurement count value */

/************************************************************************/
/*              Definition of external reference                        */
/************************************************************************/

extern void             DisInt( void );                         /* Interrupt disable function */
extern void             EnInt( void );                          /* Interrupt enable function */

/************************************************************************/
/*              Function prototype declaration                          */
/************************************************************************/

        void            TIO4_MesClr_init( void );               /* Set TIO4 measurement clear input mode */
        void            TIN0_Int( void );                       /* Process TIN0 (TIO4 measurement input)
interrupt */

/************************************************************************/
/*              Define macro                                            */
/************************************************************************/

/*** Measurement clear input (TIO4) ***/

                                                /* 0123 4567
*/
#define IEB2_MASK             0x30u               /* 0011 0000B
*/
#define IEB2_Tin0             0x00u               /* 0000 0000B
*/
                                               /*   ++-------- Input event bus 2 : TIN0
*/

                                                /* 0123 4567
*/
#define TIO4_MASK             0xffu               /* 1111 1111B
*/
#define TIO4_MesClr           0x34u               /* 0011 0100B
*/
                                              /* |||| |+++--- Set TIO4 measurement clear input mode
*/
                                               /* |||+-+------ Measurement input source :
*/
                                                /* |||                input event bus 2 selected
*/
                                               /* ||+--------- Enable external input
*/
                                                /* ++---------- Select clock bus 0
*/

                                                /* 0123 4567 89AB CDEF
*/
#define TIN0_MASK             0x0003u             /* 0000 0000 0000 0011B
*/
#define TIN0_REdge            0x0001u             /* 0000 0000 0000 0001B
*/
                                                      /*                 ++- Set rise on TIN0 to be the
active edge  */

/*""FUNC COMMENT""*****************************************************
 * Function name: TIO4_MesClr_init()
 *-----------------------------------------------------------------------
 * Description  : Initial settings for TIO4 measurement clear input mode
 *              : - While using TIO4 in measurement clear input mode,
 *              :   this function measures the period of a rising edge on TIN0 which is entered via input
event bus 2
```

```
 *              : - The count source used for this operation is clock bus 0
 *-----------------------------------------------------------------------
 * Argument     : -
 *-----------------------------------------------------------------------
 * Returns      : -
 *-----------------------------------------------------------------------
 * Notes        : The prescaler, clock bus, etc. are set separately from the above
 *              : ICU's interrupt control related registers are set separately from the above
 *              : Port input function must be enabled
 *              : This function must be executed while interrupts are disabled
 *""FUNC COMMENT END""**************************************************/
void    TIO4_MesClr_init( void )
{
        UCHAR   temp;
        USHORT  temp16;

/*** Setting input event bus 2 ***/

        temp = CKIEBCR;
        CKIEBCR = ( temp & ~IEB2_MASK) | IEB2_Tin0;             /* Select input event bus 2 for input */

/*** Setting measurement clear input (TIO4) ***/

        temp = TIO4CR;
        TIO4CR = ( temp & ~TIO4_MASK) | TIO4_MesClr;            /* Set TIO4 measurement clear input mode */

/*** Setting P150 (TIN0) rising edge active ***/

        temp16 = TINCR0;
        TINCR0 = ( temp16 & ~TIN0_MASK) | TIN0_REdge;           /* Setting TIN0 rising edge active */
        P15MOD |= 0x80u;                                        /* Select P150 for TIN0 */

/*** Setting interrupt (TIO4) ***/

        temp = TIOIR1;
        temp |= ( TIOIS7 | TIOIS6 | TIOIS5) | TIOIM4;           /* Disable TIO4 interrup */
        temp &= ~TIOIS4;                                        /* Clear TIO4 interrupt request status */
        TIOIR1 = temp;

/*** Setting interrupt (TINO) ***/

        TINIR0 &= ~( TINIS0 | TINIM0);                          /* Enable TIN0 interrupt */

/*** Initializing counter ***/

        TIO4CT = 0xffff;                                        /* Set initial value in TIO4 counter */

/*** Starting count ***/

        TIOPRO = (~TIO4PRO) & 0xFFFFu;                          /* Enable TIO4 enable protect rewrite */
        TIOCEN = 0xffff;                                        /* Starting count TIO4 */
}

/*""FUNC COMMENT""*******************************************************
 * Function name: TIN0_Int()
 *-----------------------------------------------------------------------
 * Description  : Read out captured value (TIO4 measured time) in TIN0 interrupt
 *-----------------------------------------------------------------------
 * Argument     : -
 *-----------------------------------------------------------------------
 * Returns      : -
 *-----------------------------------------------------------------------
 * Notes        : TIO4 interrupt processing is needed for judgment of twice or more overflow occurrences
 *""FUNC COMMENT END""**************************************************/
void    TIN0_Int( void )
{
        UCHAR   temp;

        DisInt();                                               /* Interrupt disable */

        MesTime = ( 0xffff - TIO4RL0);                          /* Read out measured time */
        if(( TIOIR1 & TIOIS4) != 0u) {                          /* Determine overflow */
                temp = TIOIR1;
                temp |= ( TIOIS5 | TIOIS6 | TIOIS7);
                temp &= ~TIOIS4;
                TIOIR1 = temp;                                  /* Clear interrupt request status */
                MesTime += 0x10000ul;                           /* Add to compensate for overflow */
        }

        EnInt();                                                /* Interrupt enable */
}
```

4.4.3 startup.ms (A part is extracted.)

(abbreviation)

```
;****************************************
; ICU Vector Table
;****************************************
;
        .SECTION        ICUVECT, DATA, ALIGN=4
;
    .IMPORT             $TIN0_2_Int
;
vectbl:
        .DATA.W         EIT_reset               ; H'0000 0094   MJT Input Interrupt 4:TIN3-TIN6
        .DATA.W         EIT_reset               ; H'0000 0098   MJT Input Interrupt 3:TIN20-TIN23
        .DATA.W         EIT_reset               ; H'0000 009C   MJT Input Interrupt 2:TIN12-TIN19
        .DATA.W         $TIN0_2_Int             ; H'0000 00A0   MJT Input Interrupt 1:TIN0-TIN2
        .DATA.W         EIT_reset               ; H'0000 00A4   MJT Input Interrupt 0:TIN7-TIN11
        .DATA.W         EIT_reset               ; H'0000 00A8   MJT Output Interrupt 7:TMS0,TMS1
        .DATA.W         EIT_reset               ; H'0000 00AC   MJT Output Interrupt 6:TOP8,TOP9
        .DATA.W         EIT_reset               ; H'0000 00B0   MJT Output Interrupt 5:TOP10
        .DATA.W         EIT_reset               ; H'0000 00B4   MJT Output Interrupt 4:TIO4-TIO7
        .DATA.W         EIT_reset               ; H'0000 00B8   MJT Output Interrupt 3:TIO8,TIO9
        .DATA.W         EIT_reset               ; H'0000 00BC   MJT Output Interrupt 2:TOP0-TOP5
        .DATA.W         EIT_reset               ; H'0000 00C0   MJT Output Interrupt 1:TOP6,TOP7
        .DATA.W         EIT_reset               ; H'0000 00C4   MJT Output Interrupt 0:TIO0-TIO3
        .DATA.W         EIT_reset               ; H'0000 00C8   DMAC0-4 Interrupt:DMA0-DMA4
        .DATA.W         EIT_reset               ; H'0000 00CC   SIO1 Receive Interrupt
        .DATA.W         EIT_reset               ; H'0000 00D0   SIO1 Transmit Interrupt
        .DATA.W         EIT_reset               ; H'0000 00D4   SIO0 Receive Interrupt
        .DATA.W         EIT_reset               ; H'0000 00D8   SIO0 Transmit Interrupt
        .DATA.W         EIT_reset               ; H'0000 00DC   A-D0 Conversion Interrupt
        .DATA.W         EIT_reset               ; H'0000 00E0   TID0 Output Interrupt
        .DATA.W         EIT_reset               ; H'0000 00E4   TOD0 Output Interrupt
        .DATA.W         EIT_reset               ; H'0000 00E8   DMAC5-9 Interrupt:DMA5-DMA9
        .DATA.W         EIT_reset               ; H'0000 00EC   SIO2,3 Transmit/Receive Interrupt
        .DATA.W         EIT_reset               ; H'0000 00F0   RTD Interrupt
        .DATA.W         EIT_reset               ; H'0000 00F4   TID1 Output Interrupt
        .DATA.W         EIT_reset               ; H'0000 00F8   TOD1,TOM0 Output Interrupt
        .DATA.W         EIT_reset               ; H'0000 00FC   SIO4,5 Transmit/Receive Interrupt
        .DATA.W         EIT_reset               ; H'0000 0100   A-D1 Conversion Interrupt
        .DATA.W         EIT_reset               ; H'0000 0104   TID2 Output Interrupt
        .DATA.W         EIT_reset               ; H'0000 0108   TML1 Input Interrupt
        .DATA.W         EIT_reset               ; H'0000 010C   CAN0 Transmit/Receive & Error Interrupt
        .DATA.W         EIT_reset               ; H'0000 0110   CAN1 Transmit/Receive & Error Interrupt
;
```

(abbreviation)

4.5 Timing of operation

Timing of operation in this reference program is shown below.

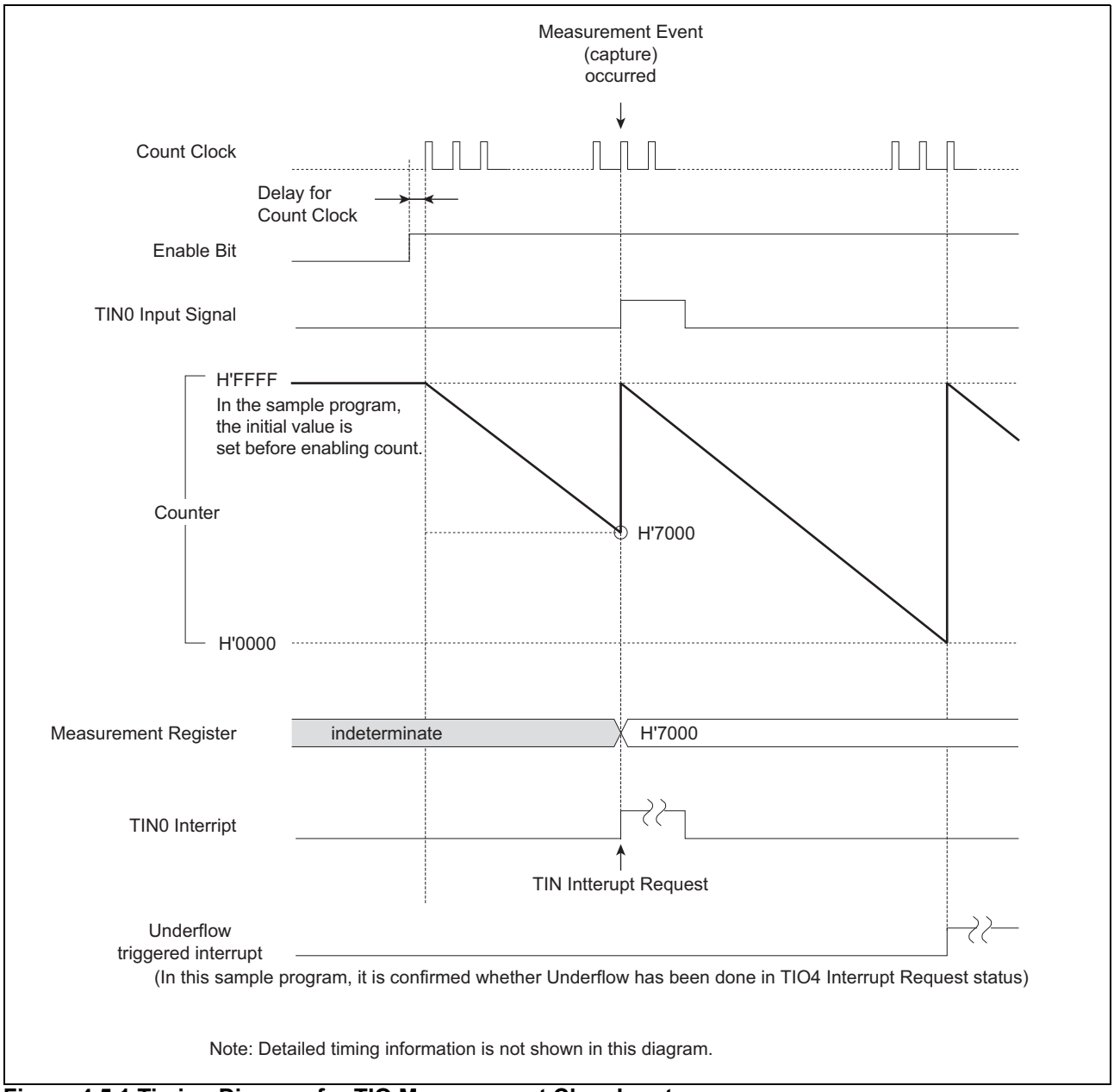

Figure 4.5.1 Timing Diagram for TIO Measurement Clear Input

5. Reference of Document

- 32176 Group User's Manual Rev.1.01
- M32R Family Software Manual Rev.1.20
- M3T-CC32R V.4.30 User's Manual (Compiler)
- M3T-AS32R V.4.30 User's Manual (Assembler)

(Please get the latest one from Renesas Technology Corp. website.)

6. Website and Support Center

- Renesas Technology Corp. website
  http://www.renesas.com/

- Customer Support Center for all Products and Technical Support Center for M32R Family
  Customer Support Center: csc@renesas.com

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Jan 13, 2006	–	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 non-flammable 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.