

To our customers,

---

## Old Company Name in Catalogs and Other Documents

---

On April 1<sup>st</sup>, 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 1<sup>st</sup>, 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.

## M16C/62

### Using the M16C/62 Timer in Pulse Period/Width Measurement Mode

#### 1.0 Abstract

Measuring the frequency (1/period) or the pulse width of an input signal is useful in applications such as tachometers, DC motor control, power usage calculations, and so on. The following article describes how to use timer B to measure the period and pulse width of an input waveform, referred to as “Pulse Period/Pulse Width Measurement Mode”.

#### 2.0 Introduction

The M16C/62 is a 16-bit MCU, based on the M16C CPU core, with an impressive list of features including 10-bit A/D, D/A, UARTS, timers, DMA, etc., and up to 256k bytes of user flash. The MCU has 6 B timers and all 6 timers can operate in Pulse Period/Pulse Width Measurement Mode.

Timer B has the following additional modes of operation:

- Timer Mode
- Event Counter Mode

Figure 1 illustrates the operation of timer B. The remainder of this article focuses on setting up timer B0 to measure pulse width, and timer B1 to measure pulse period.

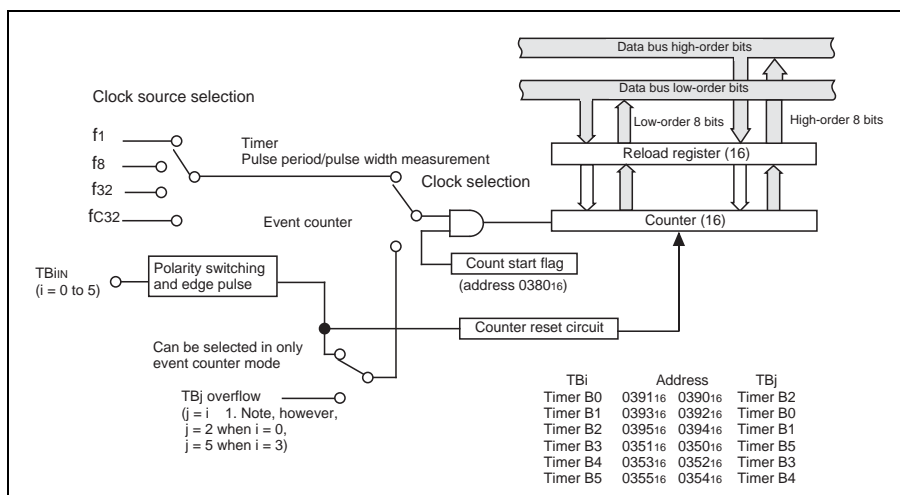


Figure 1 Block Diagram of Timer B

### 3.0 Pulse Period Measurement

In Period Measurement Mode (e.g. falling edge to falling edge), after the 'start count flag' is set, the counter counts up using the selected clock source and every time a falling edge is detected on the TBiIN pin, the value in the counter is transferred to the reload register, the counter is reset to zero, and then continues counting. At the same time, the timer interrupt request bit is set and an interrupt is generated if the timer interrupt priority level is set above the current CPU priority level (if the I flag in the CPU flag register is cleared, the interrupt will not be serviced until the flag is set). If the timer's counter overflows within a period, it will also generate the interrupt and the MR3 bit in the TBIMR is set to distinguish between the interrupt causes. Note that the measurement is free running and the reload register contains the most recent measurement. The user has the option of polling the TBi register or reading it in an interrupt service routine. Also note that the value of the counter immediately after the 'start count flag' is set is indeterminate and an overflow could occur before the first falling edge. Figure 2 illustrates this.

### 3.1 Pulse Width Measurement

Pulse Width Measurement Mode operates in much the same way except the count register is transferred to the reload register for every edge detected on the TBiIN pin, and the counter resets and resumes counting, as shown in Figure 3. Again, note that the value of the counter immediately after the 'start count flag' is set is indeterminate and an overflow could occur before the first falling edge. This measurement is also free running but now the user must determine by software whether the measurement is for the high or low width.

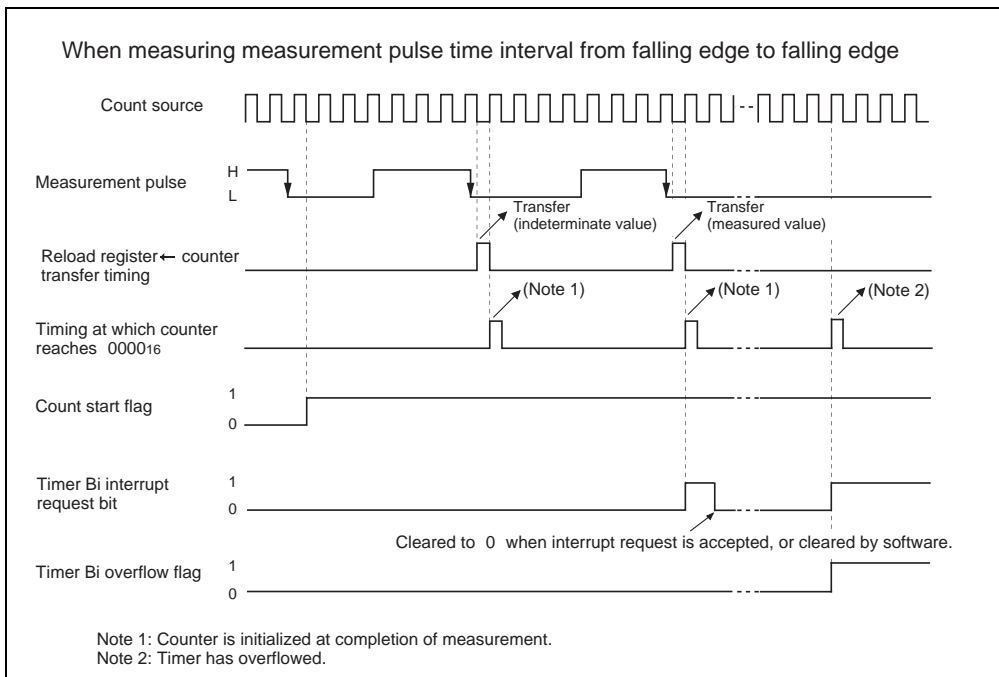
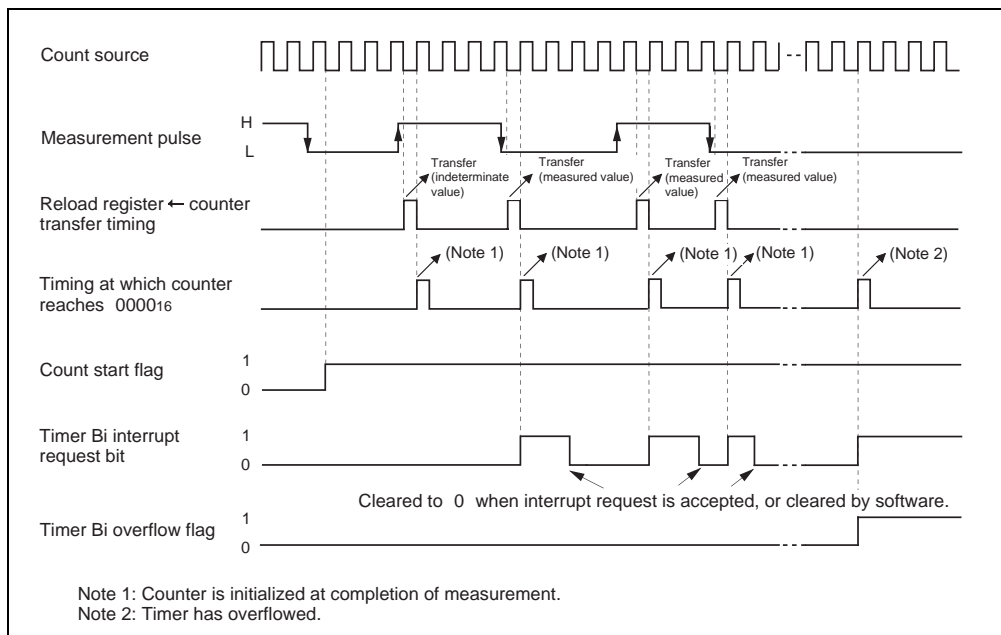


Figure 2 Operation Timing When Measuring a Pulse Period

### 3.2 Pulse Width Measurement

Pulse Width Measurement Mode operates in much the same way except the count register is transferred to the reload register for every edge detected on the TBIIN pin, and the counter resets and resumes counting, as shown in 0. Again, note that the value of the counter immediately after the 'start count flag' is set is indeterminate and an overflow could occur before the first falling edge. This measurement is also free running but now the user must determine by software whether the measurement is for the high or low width.



**Figure 3 Operation Timing When Measuring a Pulse Width**

### 4.0 Configuring Pulse Period/Pulse Width Measurement Mode

To configure a timer for Pulse Period/Pulse Width Measurement Mode:

1. Load the timer mode register, TBI<sub>MR</sub>.
  - Select Measurement Mode: bits T<sub>MOD0</sub> = 0, T<sub>MOD1</sub> = 1.
  - Set the MR<sub>0</sub> and MR<sub>1</sub> bits for period or width measurement.
  - Clear the MR<sub>2</sub> bit for period or width measurement.
  - MR<sub>3</sub> is the timer Bi overflow flag (can be cleared but not set).
  - Select the clock source (f<sub>1</sub>, f/8, f/32, or f<sub>c</sub>/32): bits T<sub>CK0</sub>, T<sub>CK1</sub> register.
2. Set the timer 'interrupt priority level', TBI<sub>IC</sub>, to at least 1 if required.
3. Enable interrupts (CPU I flag set).
4. Set the 'start count flag' bit, TBI<sub>S</sub>, in the 'count start flag' register, TAB<sub>SR</sub> or TBS<sub>R</sub>.

It is not necessary to perform these steps in the order listed, but the mode register should be loaded before the 'start count' flag is set. Also, the priority level should not be modified when there is a possibility of an interrupt occurring.

The required registers are shown in Figure 4 through Figure 8.

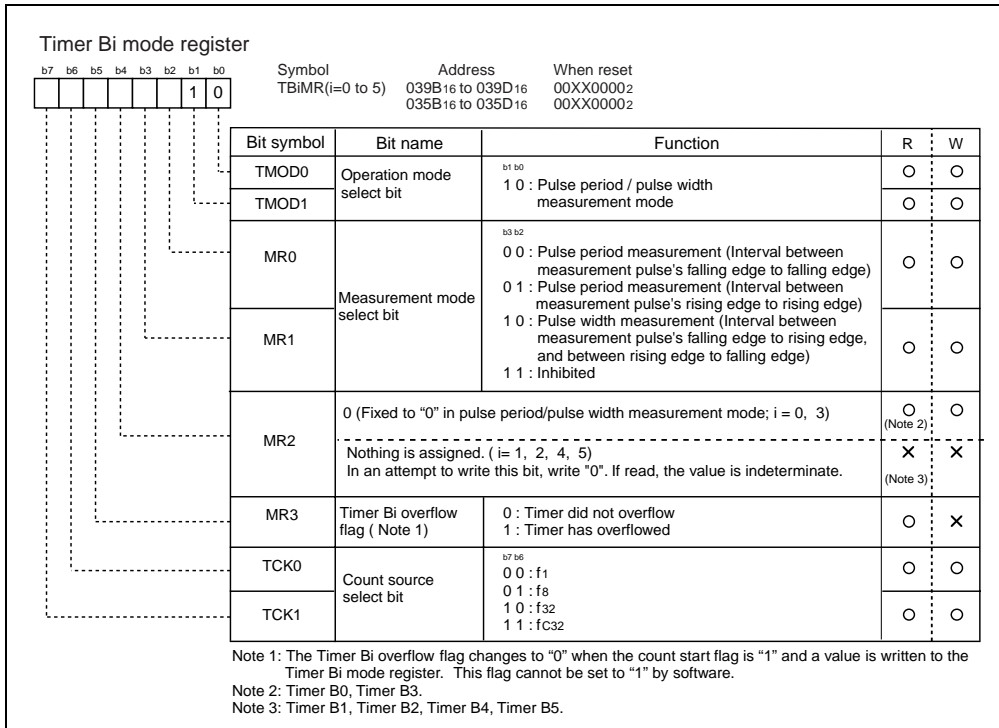


Figure 4 Timer Bi Mode Register in Pulse Period/Pulse Width Measurement Mode

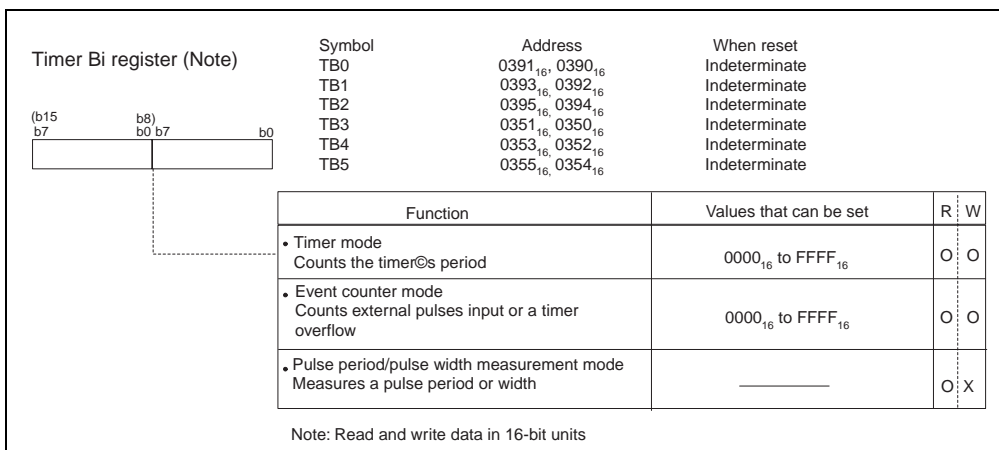


Figure 5 Timer Bi Register

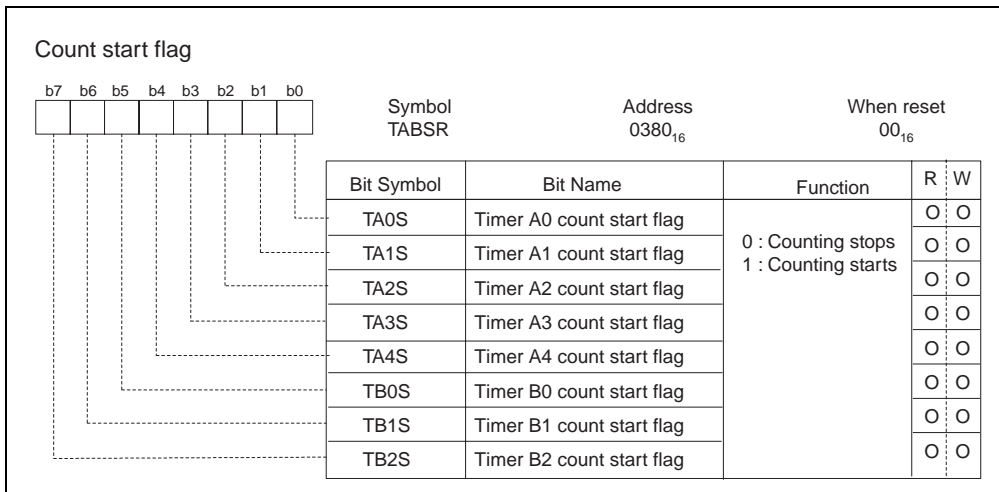


Figure 6 Count Start Flag Register

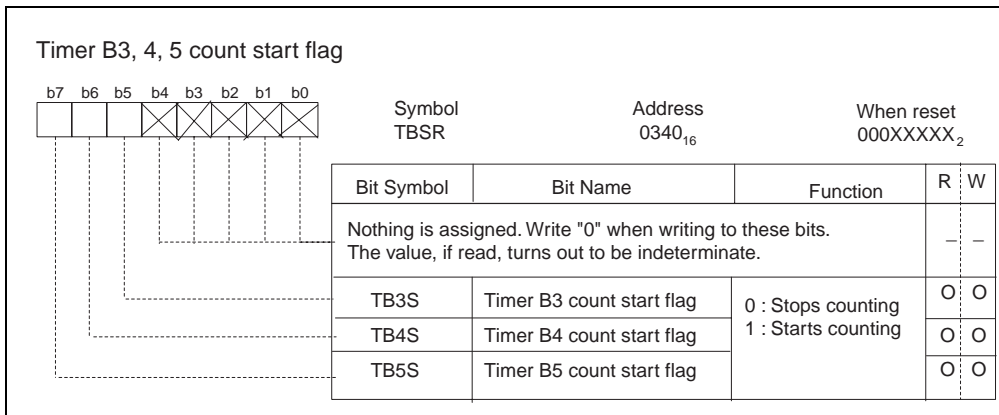


Figure 7 Timer B3, 4, 5 Count Start Flag Register

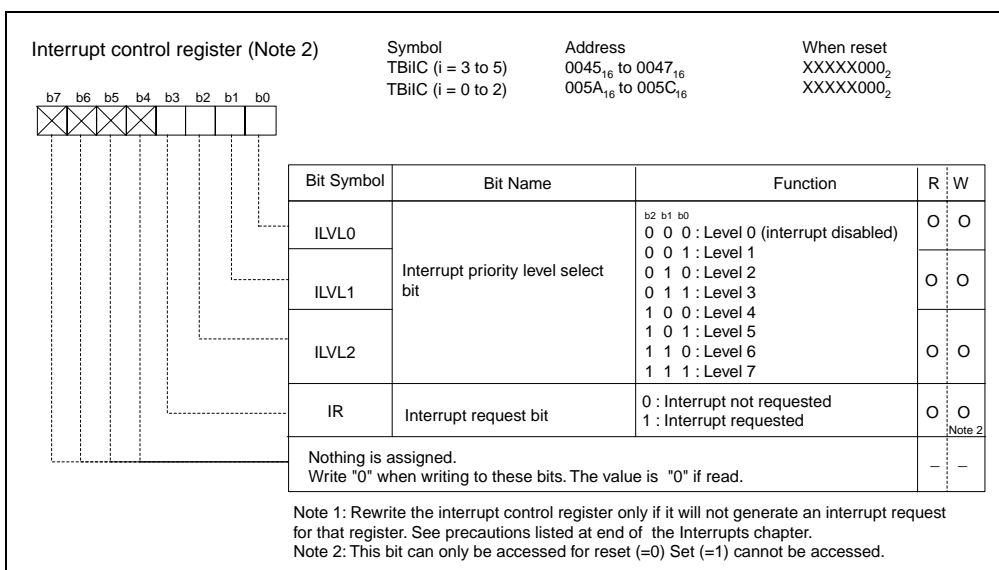


Figure 8 Interrupt Control Register

## 5.0 Reference

### Renesas Technology Corporation Semiconductor Home Page

<http://www.renesas.com>

### E-mail Support

[support\\_apl@renesas.com](mailto:support_apl@renesas.com)

### Data Sheet

- M16C/62 datasheets, 62aeds.pdf

### User's Manual

- NC30 Ver. 4.0 User's Manual, NC30UE.pdf
- M16C/60 and M16C/20 C Language Programming Manual, 6020EC.pdf
- M16C/62 User's Manual, 62eum.pdf
- Application Note: Writing Interrupt Handlers in C for the M16C

## 6.0 Software Code

Below is a program written for Renesas' NC30 compiler to illustrate how to configure Pulse Period/Pulse Width Measurement Mode. The program can measure up to about a 16msec period and runs on the MSV1632 Starter Kit Board. Using the KD30 debugger, the program can be 'stopped' and the global variables 'widthlow', 'width\_hi', and 'period' viewed from the global watch window.

To become familiar with this mode, try changing the clock source or even switch to a different timer (e.g. TB2, TB3, etc.).

```
/*  
*  
* File Name: period_width.c  
*  
* Content: Example program using Timer B in 'pulse width / period measurement  
* mode'. This program is written for the "Pulse Period/Width  
* Measurement" application note. Timer B0 is configured to measure  
* pulse width(TB0in pin),and timer B1,measures the period(TB1in pin).  
* Tested using a 250Hz square wave with a 0.5msec pulse width(high).  
* This program works with the MSV1632 starter kit board.  
*/
```



```

*
*   Compiled with NC30 ver. 3.20.00.
*
*   All timing based on 16 Mhz Xtal
*
*   Copyright, 2003 Renesas Technology Corporation, Inc.
*=====
*   $Log:$
*===== */
#include "sfr62.h"

#define B1TIME_CONFIG 0x42 /* 01000010 value to load into timer B1 mode register
    |||||_  TMOD0,TMOD1: PULSE MEASUREMENT MODE
    |||||_  MR0,MR1:    PULSE PERIOD MODE
    |||_    MR2:       = 0 FOR PULSE MEASUREMENT
    ||_    MR3:       OVERFLOW FLAG
    ||_    TCK0,TCK1:  F DIVIDED BY 8 SELECTED */

#define B0TIME_CONFIG 0x4a /* 01001010 value to load into timer B0 mode register
    |||||_  TMOD0,TMOD1: PULSE MEASUREMENT MODE
    |||||_  MR0,MR1:    PULSE WIDTH MODE
    |||_    MR2:       = 0 FOR PULSE MEASUREMENT
    ||_    MR3:       OVERFLOW FLAG
    ||_    TCK0,TCK1:  F DIVIDED BY 8 SELECTED */

#define CNTR_IPL 0x03 // TB0 priority interrupt level

int period,widthlow,width_hi;

//prototypes
void init(void);

#pragma INTERRUPT /B TimerB0Int
void TimerB0Int(void);

```

/\*\*\*\*\*\*

Name: TimerB0Int()

Parameters: none

Returns: nothing

Description: Timer B0 Interrupt Service Routine. The overflow flag is checked to determine if the TB0 register contains valid data. If so, the input is tested to determine if the value in the TB0 register is the high pulse width or low width and stored in the appropriate variable.

\*\*\*\*\* \*/

```
void TimerB0Int(void)
```

```
{
    if (mr3_tb0mr ==1)          // check for timer overflow
    { tb0mr = B0TIME_CONFIG;    // if so clear flag and
      return;                  // data invalid, so leave
    }
    if (p9_0== 1)
        widthlow = tb0 ;      // if input now hi, just measured a low width
    else
        width_hi = tb0;

}
```

/\*\*\*\*\*\*

Name: main()

Parameters: none

Returns: nothing

Description: initializes variables. Then the variable 'period' is constantly updated with the period count in timer TB1. This is to illustrate that the period measurement is free running. Note that the first few times TB1 is read, the data may not be valid.

\*\*\*\*\*/

```

void main (void)
{

    init();
    while (1)
    {
        period = tb1 ;           // period measured in polled mode
    }
}

/*****
Name:  initial()
Parameters:  none
Returns:  nothing
Description:  Timer TB0 setup for pulse width interrupts and TB1 configured for
              pulse period measurement (no interrupts).
*****/
void init()
{

/* the following procedure for writing an Interrupt Priority Level follows that as
described in the M16C
data sheets under 'Interrupts' */

    _asm ("  fclr i" ) ;           // turn off interrupts before modifying IPL
    tb0ic |= CNTR_IPL;           // use read-modify-write instruction to write IPL
    tb0mr = B0TIME_CONFIG;
    _asm ("  fset i" );

    tb0s = 1; //start counting

    tb1mr = B1TIME_CONFIG;
    tb1s = 1; //start counting

}

```

In order for this program to run properly, timer B0's interrupt vector needs to point to the function. The interrupt vector table is near the end of the startup file "sect30.inc". Insert the function label "TimerB0Int" into the interrupt vector table at vector 26 as shown below.

```

;*****
;
;   C Compiler for M16C/62
;
;   Copyright,2003 Renesas Technology Corporation, Inc.
;   All Rights Reserved.
;
;
;   Modified for use on MSV1632 Starter Kit.
;   sect30.inc      : section definition
;   This program is applicable when using KD30 and the ROM Monitor.
;
;   $Id:
;
;*****

:
:
:
:
:
:

.lword      dummy_int      ; timer A0(for user)(vector 21)
.lword      dummy_int      ; timer A1(for user)(vector 22)
.lword      dummy_int      ; timer A2(for user)(vector 23)
.lword      dummy_int      ; timer A3(for user)(vector 24)
.lword      dummy_int      ; timer A4(for user)(vector 25)
.glb
.lword      _TimerB0Int    ; timer B0(for user)(vector 26)
.lword      dummy_int      ; timer B1(for user)(vector 27)
.lword      dummy_int      ; timer B2(for user)(vector 28)
.lword      dummy_int      ; int0 (for user)(vector 29)
.lword      dummy_int      ; int1 (for user)(vector 30)
.lword      dummy_int      ; int2 (for user)(vector 31)

.lword      dummy_int      ; vector 32 (for user or MR30)
.lword      dummy_int      ; vector 33 (for user or MR30)
.lword      dummy_int      ; vector 34 (for user or MR30)
.lword      dummy_int      ; vector 35 (for user or MR30)
.lword      dummy_int      ; vector 36 (for user or MR30)
.lword      dummy_int      ; vector 37 (for user or MR30)
:
:
:

```

## Keep safety first in your circuit designs!

- Renesas Technology Corporation 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

- These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation 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 Corporation or a third party.
- Renesas Technology Corporation 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.
- 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 Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation 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 Corporation assumes no responsibility for any damage, liability, or other loss arising from these inaccuracies or errors.  
Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (<http://www.renesas.com>).
- 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 Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
- Renesas Technology Corporation 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 Corporation or an authorized Renesas Technology Corporation 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.
- The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
- 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.
- Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.