To all our customers

Regarding the change of names mentioned in the document, such as Mitsubishi Electric and Mitsubishi XX, to Renesas Technology Corp.

The semiconductor operations of Hitachi and Mitsubishi Electric were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Mitsubishi Electric, Mitsubishi Electric Corporation, Mitsubishi Semiconductors, and other Mitsubishi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Note : Mitsubishi Electric will continue the business operations of high frequency & optical devices and power devices.

Renesas Technology Corp.
Customer Support Dept.
April 1, 2003

M16C/62

Event Counter Mode on the M16C/62

1. Abstract

Event counters are useful in automated packaging lines, tachometers, and mechanical equipment monitoring. Also, the event counters on the M16C/62 can be configured to interrupt on a single event, adding to the interrupt input pins. The following article describes how to configure the M16C/62 timers as event counters, referred to as “Event Counter Mode”.

2. Introduction

The Mitsubishi 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 5 ‘A’ timers and 6 ‘B’ timers. All 11 timers can operate in “Event Counter Mode”.

Timer A has the following additional modes of operation:

- Timer Mode
- PWM Mode
- One-Shot Mode

Timer B has the following additional modes of operation:

- Timer Mode
- Pulse Period/Pulse Width Measurement Mode

Figure 1 illustrates the operation of timer A, and Figure 2, timer B. Note that there are some differences between the two timers but both operate similarly in Event Counter Mode. The remainder of this article focuses on setting up timer A0 in Event Counter Mode.

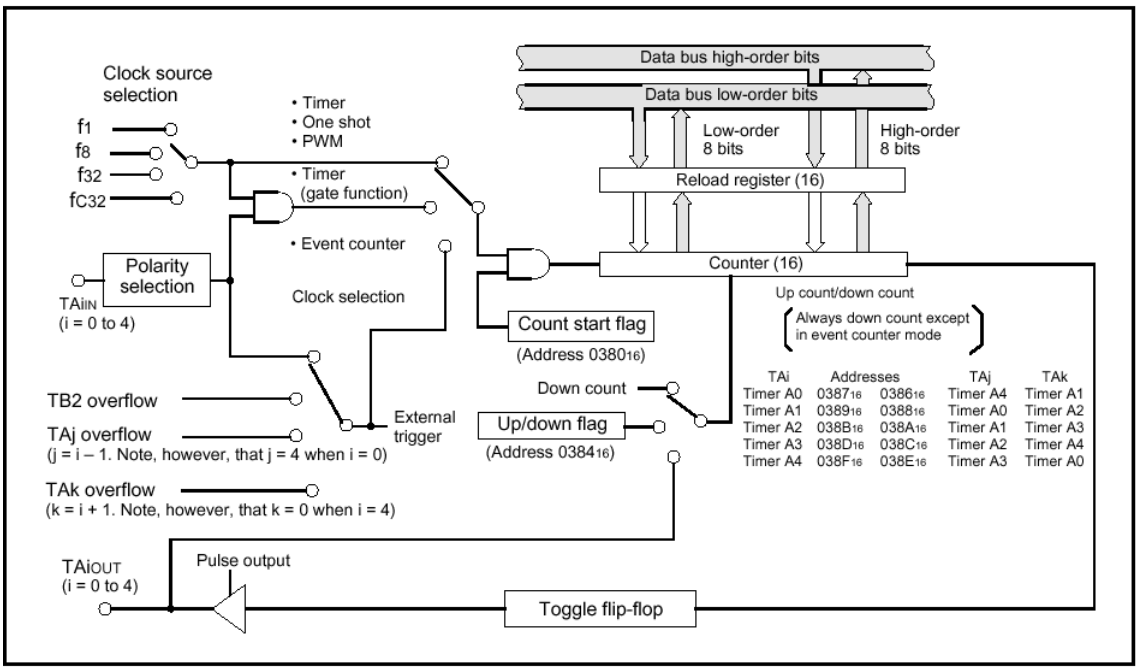


Figure 1. Block Diagram of Timer A

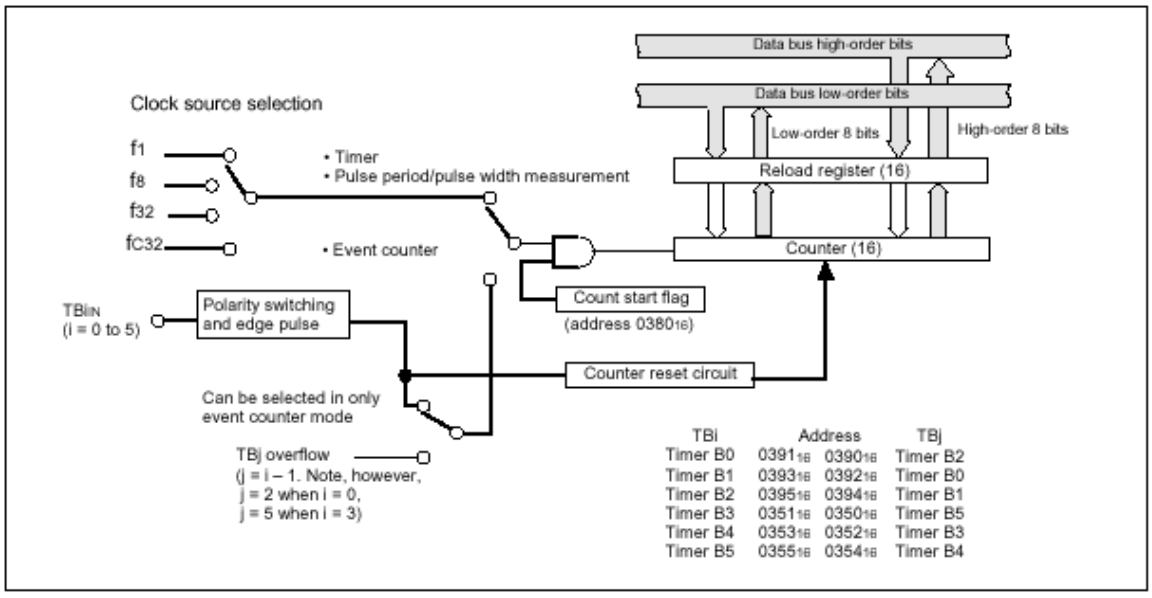


Figure 2. Block Diagram of Timer B

3. Event Counter Mode Description

In general, the Timer TAi or TBi register counts an input signal and, at any time, the count value can be read. When the timer overflows (for up-count) or underflows (down-count) 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 registers is cleared, the interrupt will not be serviced until the flag is set). If at any time during counting the count start flag is cleared, counting is suspended until set. This is illustrated in Figure 3.

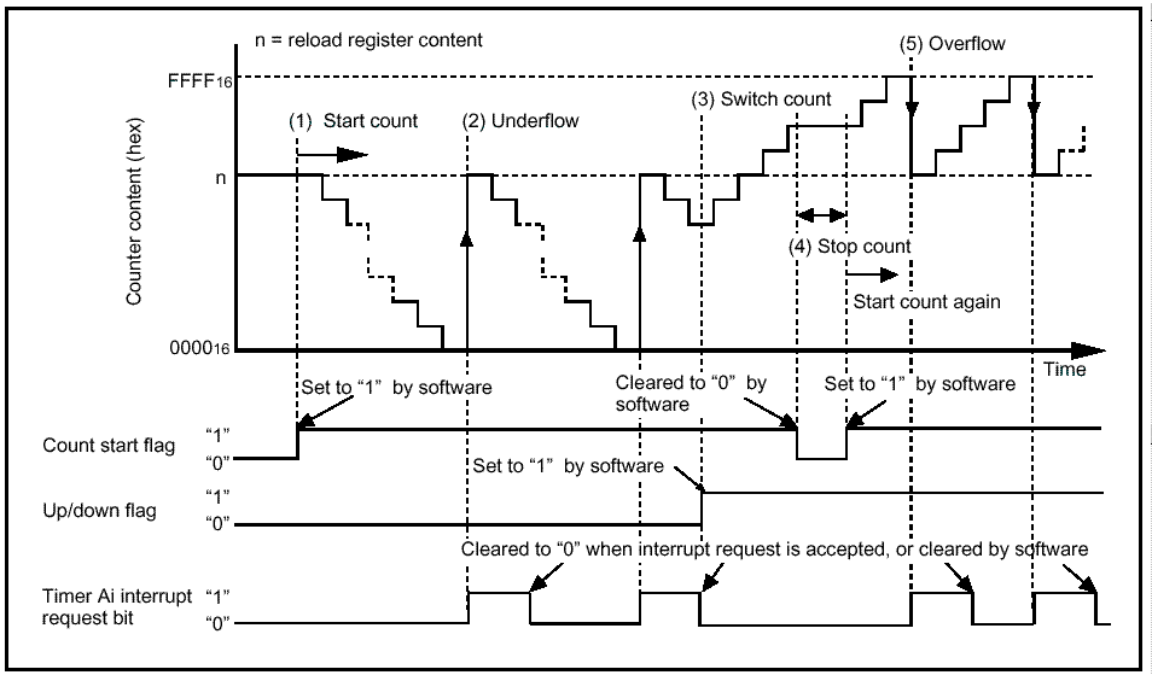


Figure 3. Operation Timing of Event Counter Mode, Reload Type Selected

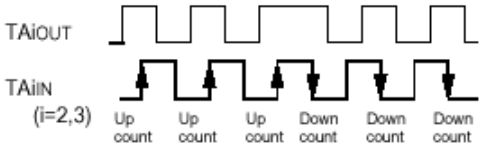
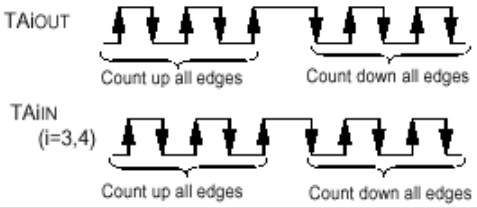
Besides having the option of counting up or down, Event Counter Mode has many other options such as count source (TAiIN or TBiIN input pin or another timer), reload or free running type, etc. and these options vary depending on which timer is used. The options and the timers they are associated with are summarized in Table 1, Table 2, and Table 3.

Table 1. Timer A Specifications in Event Counter Mode (Single Phase Mode Only)

Item	Specification
Count source	• External signals input to TAiIN pin (effective edge can be selected by software) • TB2 overflow, TAj overflow
Count operation	• Up count or down count can be selected by external signal or software • When the timer overflows or underflows, it reloads the reload register con tents before continuing counting (Note)
Divide ratio	1/ (FFFF16 - n + 1) for up count 1/ (n + 1) for down count n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	The timer overflows or underflows
TAiIN pin function	Programmable I/O port or count source input
TAiOUT pin function	Programmable I/O port, pulse output, or up/down count select input
Read from timer	Count value can be read out by reading timer Ai register
Write to timer	• When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter • When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)
Select function	• Free-run count function Even when the timer overflows or underflows, the reload register content is not reloaded to it • Pulse output function Each time the timer overflows or underflows, the TAiOUT pin's polarity is reversed

Note: This does not apply when the free-run function is selected.

Table 2. Timer Specifications in Event Counter Mode (when processing two-phase pulse signal with timers A2, A3, and A4)

Item	Specification
Count source	• Two-phase pulse signals input to TAiIN or TAiOUT pin
Count operation	• Up count or down count can be selected by two-phase pulse signal • When the timer overflows or underflows, the reload register content is reloaded and the timer starts over again (Note)
Divide ratio	1/ (FFFF16 - n + 1) for up count 1/ (n + 1) for down count n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	Timer overflows or underflows
TAiIN pin function	Two-phase pulse input
TAiOUT pin function	Two-phase pulse input
Read from timer	Count value can be read out by reading timer A2, A3, or A4 register
Write to timer	• When counting stopped When a value is written to timer A2, A3, or A4 register, it is written to both reload register and counter • When counting in progress When a value is written to timer A2, A3, or A4 register, it is written to only reload register. (Transferred to counter at next reload time.)
Select function	• Normal processing operation The timer counts up rising edges or counts down falling edges on the TAiIN pin when input signal on the TAiOUT pin is "H"  • Multiply-by-4 processing operation If the phase relationship is such that the TAiIN pin goes "H" when the input signal on the TAiOUT pin is "H", the timer counts up rising and falling edges on the TAiOUT and TAiIN pins. If the phase relationship is such that the TAiIN pin goes "L" when the input signal on the TAiOUT pin is "H", the timer counts down rising and falling edges on the TAiOUT and TAiIN pins. 

Note: This does not apply when the free-run function is selected.

Table 3. Timer B Specifications in Event Counter Mode

Item	Specification
Count source	• External signals input to TBiIN pin • Effective edge of count source can be a rising edge, a falling edge, or falling and rising edges as selected by software
Count operation	• Counts down • When the timer underflows, it reloads the reload register contents before continuing counting
Divide ratio	1/(n+1) n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	The timer underflows
TBiIN pin function	Count source input
Read from timer	Count value can be read out by reading timer Bi register
Write to timer	• When counting stopped When a value is written to timer Bi register, it is written to both reload register and counter • When counting in progress When a value is written to timer Bi register, it is written to only reload register (Transferred to counter at next reload time)

4. Configuring Event Counter Mode

To configure a timer for Event Counter Mode:

1. Load the timer mode register, TAiMR.

   Select Event Counter Mode: bits TMOD0 = 1, TMOD1 = 0.

   Set the remaining bits (MR0, MR1, MR2, TCK0, TCK1) depending on required functions (see mode register diagrams below).

2. Load the TAi or TBi register with the count source.
3. Select the trigger via the TRGSR or ONSF register (N/A for Timer B).
4. Select up or down count via the UDF register (N/A for Timer B, Timer B counts down only).
5. Set the timer 'interrupt priority level', TAiIC or TBiIC, to at least 1 if required.
6. Enable interrupts (CPU I flag set).
7. Set the 'start count' flag bit, TAiS or TBiS, in the 'count start flag' register, TABSR or TBSR.

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 to Figure 7, Figure 9, Figure 10 and Figure 12.

Timer Ai mode register

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	1

Symbol: TAiMR(i = 0, 1) — Address: 039616, 039716 — When reset: 0016

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode (Note 1)	○	○
TMOD1			○	○
MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin is a normal port pin) 1 : Pulse is output (Note 2) (TAiOUT pin is a pulse output pin)	○	○
MR1	Count polarity select bit (Note 3)	0 : Counts external signal's falling edge 1 : Counts external signal's rising edge	○	○
MR2	Up/down switching cause select bit	0 : Up/down flag's content 1 : TAiOUT pin's input signal (Note 4)	○	○
MR3	0 (Must always be fixed to "0" in event counter mode)		○	○
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	○	○
TCK1	Invalid in event counter mode Can be "0" or "1"		○	○

Note 1: In event counter mode, the count source is selected by the event / trigger select bit (addresses 038216 and 038316).
Note 2: The settings of the corresponding port register and port direction register are invalid.
Note 3: Valid only when counting an external signal.
Note 4: When an "L" signal is input to the TAiOUT pin, the downcount is activated. When "H", the upcount is activated. Set the corresponding port direction register to "0".

Figure 4. Timer Ai Mode Register in Event Counter Mode

Timer Ai mode register
(When not using two-phase pulse signal processing)

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	1

Symbol: TAiMR(i = 2 to 4) — Address: 039816 to 039A16 — When reset: 0016

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	○	○
TMOD1			○	○
MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin is a normal port pin) 1 : Pulse is output (Note 1) (TAiOUT pin is a pulse output pin)	○	○
MR1	Count polarity select bit (Note 2)	0 : Counts external signal's falling edges 1 : Counts external signal's rising edges	○	○
MR2	Up/down switching cause select bit	0 : Up/down flag's content 1 : TAiOUT pin's input signal (Note 3)	○	○
MR3	0 : (Must always be "0" in event counter mode)		○	○
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	○	○
TCK1	Two-phase pulse signal processing operation select bit (Note 4)(Note 5)	0 : Normal processing operation 1 : Multiply-by-4 processing operation	○	○

Note 1: The settings of the corresponding port register and port direction register are invalid.
Note 2: This bit is valid when only counting an external signal.
Note 3: Set the corresponding port direction register to "0".
Note 4: This bit is valid for the timer A3 mode register.
For timer A2 and A4 mode registers, this bit can be "0 "or "1".
Note 5: When performing two-phase pulse signal processing, make sure the two-phase pulse signal processing operation select bit (address 038416) is set to "1". Also, always be sure to set the event/trigger select bit (addresses 038216 and 038316) to "00".

Figure 5. Timer Ai Mode Register (When Not Using Two-Phase Pulse Signal Processing)

Timer Ai mode register
(When using two-phase pulse signal processing)

b7	b6	b5	b4	b3	b2	b1	b0
		0	1	0	0	0	1

Symbol: TAiMR(i = 2 to 4) — Address: 039816 to 039A16 — When reset: 0016

Bit symbol	Bit name	Function	R W
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	○ ○
TMOD1			○ ○
MR0	0 (Must always be "0" when using two-phase pulse signal processing)		○ ○
MR1	0 (Must always be "0" when using two-phase pulse signal processing)		○ ○
MR2	1 (Must always be "1" when using two-phase pulse signal processing)		○ ○
MR3	0 (Must always be "0" when using two-phase pulse signal processing)		○ ○
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	○ ○
TCK1	Two-phase pulse processing operation select bit (Note 1)(Note 2)	0 : Normal processing operation 1 : Multiply-by-4 processing operation	○ ○

Note 1: This bit is valid for timer A3 mode register.
For timer A2 and A4 mode registers, this bit can be "0" or "1".
Note 2: When performing two-phase pulse signal processing, make sure the two-phase pulse signal processing operation select bit (address 038416) is set to "1". Also, always be sure to set the event/trigger select bit (addresses 038216 and 038316) to "00".

Figure 6. Timer Ai Mode Register (When Using Two-Phase Pulse Signal Processing)

Timer Bi register (Note)

(b15) b7 — (b8) b0 b7 — b0

Symbol	Address	When reset
TB0	039116, 039016	Indeterminate
TB1	039316, 039216	Indeterminate
TB2	039516, 039416	Indeterminate
TB3	035116, 035016	Indeterminate
TB4	035316, 035216	Indeterminate
TB5	035516, 035416	Indeterminate

Function	Values that can be set	R W
• Timer mode Counts the timer's period	000016 to FFFF16	○ ○
• Event counter mode Counts external pulses input or a timer overflow	000016 to FFFF16	○ ○
• Pulse period / pulse width measurement mode Measures a pulse period or width	——	○ ✕

Note: Read and write data in 16-bit units.

Figure 7. Timer Bi Register

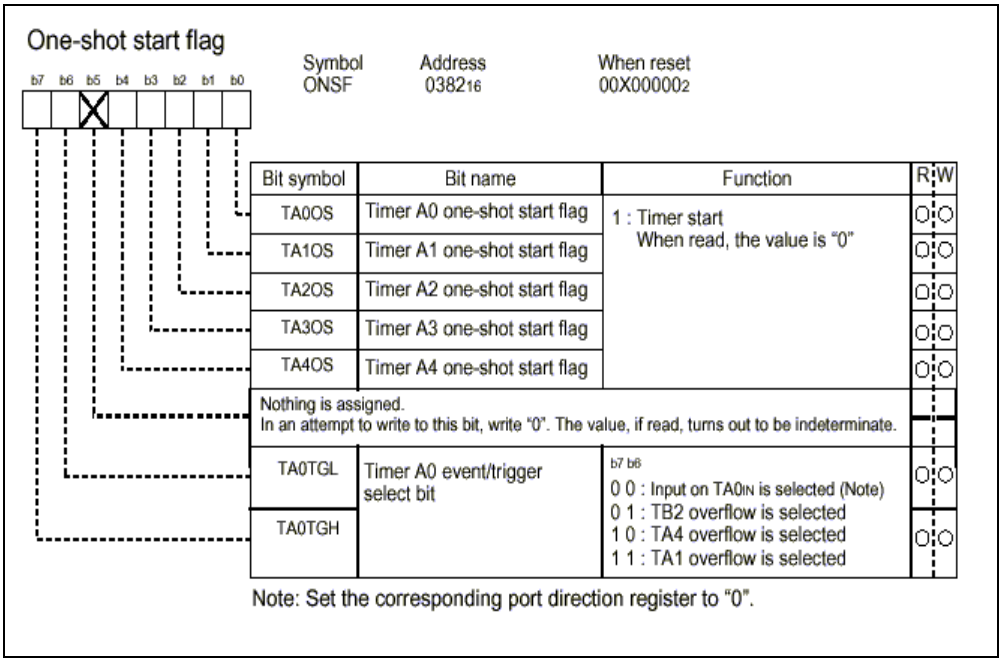


Figure 8. One-Shot Start Flag

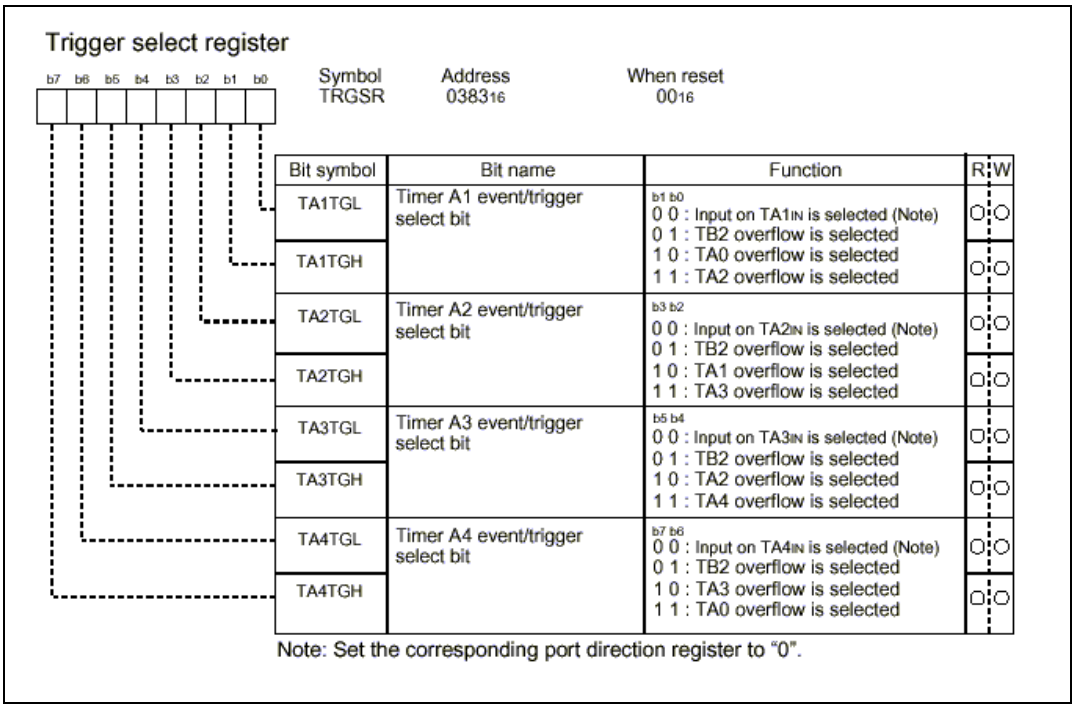
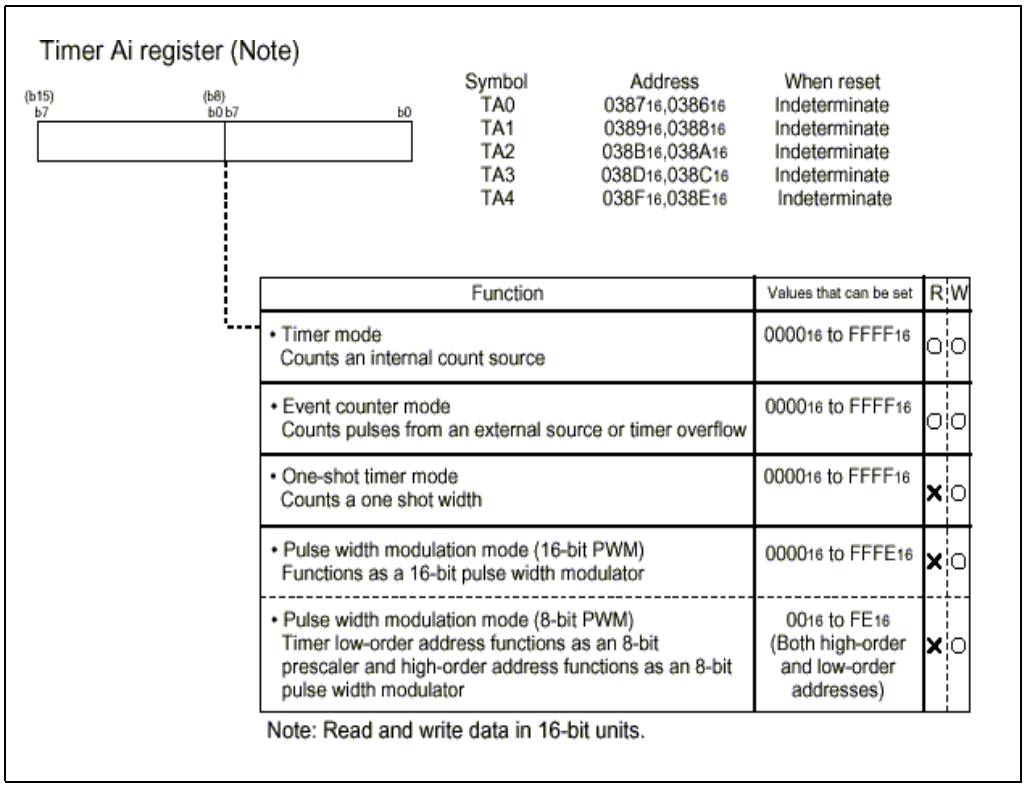


Figure 9. Trigger Select Register



Timer Ai register (Note)

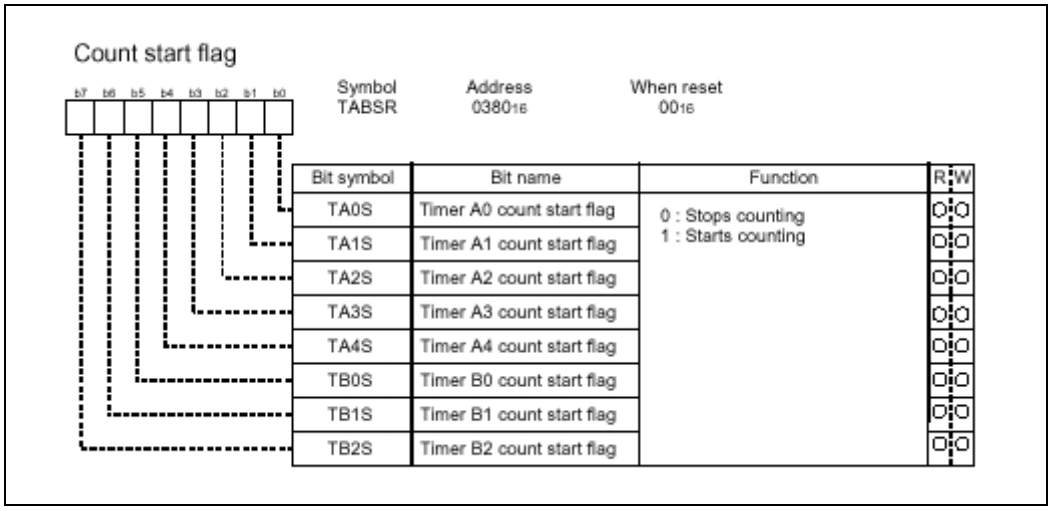
(b15) b7 (b8) b0 b7 b0

Symbol	Address	When reset
TA0	038716,038616	Indeterminate
TA1	038916,038816	Indeterminate
TA2	038B16,038A16	Indeterminate
TA3	038D16,038C16	Indeterminate
TA4	038F16,038E16	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts an internal count source	000016 to FFFF16	○	○
• Event counter mode Counts pulses from an external source or timer overflow	000016 to FFFF16	○	○
• One-shot timer mode Counts a one shot width	000016 to FFFF16	×	○
• Pulse width modulation mode (16-bit PWM) Functions as a 16-bit pulse width modulator	000016 to FFFE16	×	○
• Pulse width modulation mode (8-bit PWM) Timer low-order address functions as an 8-bit prescaler and high-order address functions as an 8-bit pulse width modulator	0016 to FE16 (Both high-order and low-order addresses)	×	○

Note: Read and write data in 16-bit units.

Figure 10. Timer Ai Register



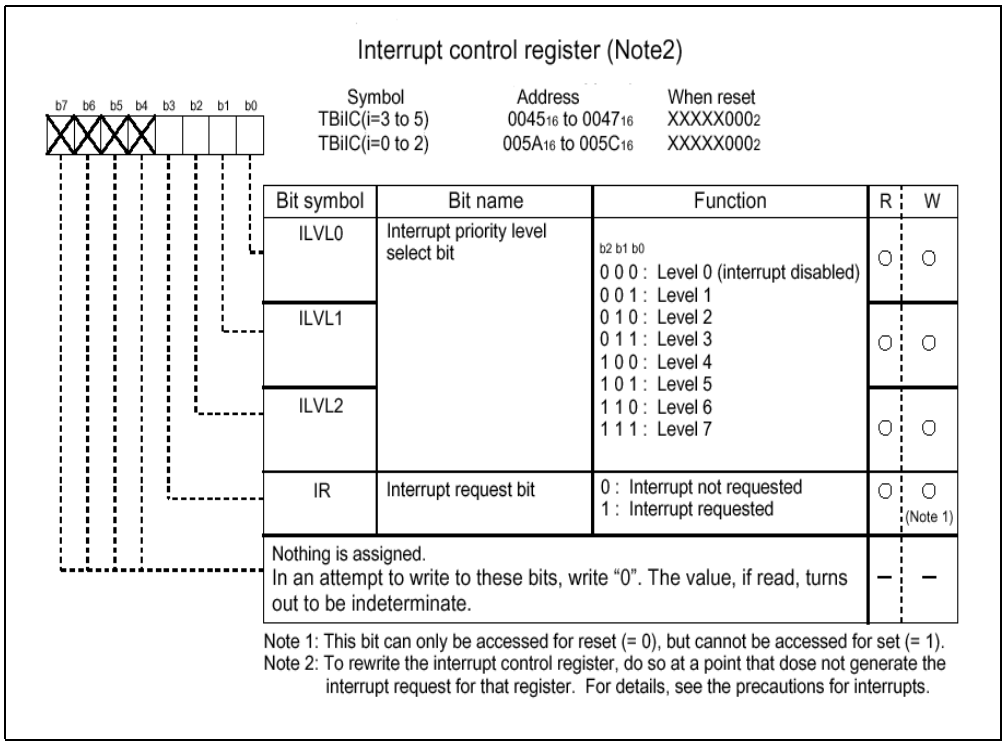
Count start flag

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	When reset
TABSR	038016	0016

Bit symbol	Bit name	Function	R	W
TA0S	Timer A0 count start flag	0 : Stops counting 1 : Starts counting	○	○
TA1S	Timer A1 count start flag		○	○
TA2S	Timer A2 count start flag		○	○
TA3S	Timer A3 count start flag		○	○
TA4S	Timer A4 count start flag		○	○
TB0S	Timer B0 count start flag		○	○
TB1S	Timer B1 count start flag		○	○
TB2S	Timer B2 count start flag		○	○

Figure 11. Count Start Flag



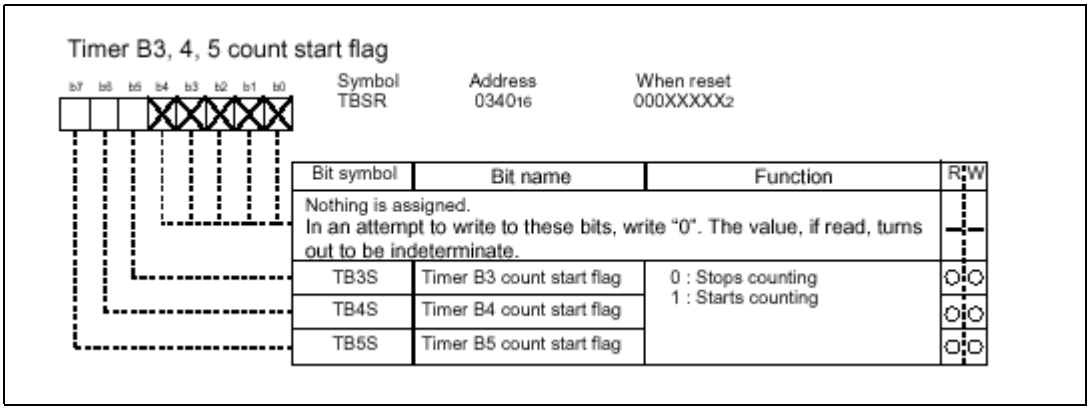
Interrupt control register (Note2)

Symbol	Address	When reset
TBiIC(i=3 to 5)	0045_{16} to 0047_{16}	XXXXX000$_2$
TBiIC(i=0 to 2)	$005A_{16}$ to $005C_{16}$	XXXXX000$_2$

Bit symbol	Bit name	Function	R	W
ILVL0	Interrupt priority level select bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	○	○
ILVL1			○	○
ILVL2			○	○
IR	Interrupt request bit	0 : Interrupt not requested 1 : Interrupt requested	○	○ (Note 1)
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate.			–	–

Note 1: This bit can only be accessed for reset (= 0), but cannot be accessed for set (= 1).
Note 2: To rewrite the interrupt control register, do so at a point that dose not generate the interrupt request for that register. For details, see the precautions for interrupts.

Figure 12. Interrupt Control Register



Timer B3, 4, 5 count start flag

Symbol	Address	When reset
TBSR	0340_{16}	000XXXXX$_2$

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate.			–	–
TB3S	Timer B3 count start flag	0 : Stops counting 1 : Starts counting	○	○
TB4S	Timer B4 count start flag		○	○
TB5S	Timer B5 count start flag		○	○

Figure 13. Timer B3, 4, 5 Count Start Flag

5. References

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

6. Software Code

The software supplied by Mitsubishi Electric & Electronics USA, Inc. is intended and supplied for use on Mitsubishi Electric products. This software is owned by Mitsubishi Electric & Electronics USA, Inc. or Mitsubishi Electric Corporation and is protected under applicable copyright laws. All rights are reserved.

THIS SOFTWARE IS PROVIDED "AS IS". NO WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE APPLY TO THIS SOFTWARE. MITSUBISHI ELECTRIC & ELECTRONICS USA, INC. AND MITSUBISHI ELECTRIC CORPORATION RESERVE THE RIGHT, WITHOUT NOTICE, TO MAKE CHANGES TO THIS SOFTWARE. NEITHER MITSUBISHI ELECTRIC & ELECTRONICS USA, INC. NOR MITSUBISHI ELECTRIC CORPORATION SHALL, IN ANY CIRCUMSTANCES, BE LIABLE FOR SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES FOR ANY REASON WHATSOEVER ARISING OUT OF THE USE OR APPLICATION OF THIS SOFTWARE.

Below is a program written for Mitsubishi's NC30 compiler to illustrate how to configuring Event Counter Mode. The program counts 100 falling edges on the P7.1 (TA0IN) pin then flashes D6 on the MDECE0620 Starter Kit Board.

To get familiar with this mode, try changing to up-count, the count value or even switch to a different timer (e.g. TA1, TB0, etc).

```
/***********************************************************************
*
*     File Name: event_mode.c
*
*     Content: Example program using Timer A in "Event Counter Mode". This program
*           is written for the Event Counter Mode application note. This program
*           works with the MDECE0620 starter kit board.
*      Compiled with NC30 ver. 3.20.00.
*      All timing based on 16 Mhz Xtal
*
*     Copyright,2001 MITSUBISHI ELECTRIC CORPORATION
*     AND MITSUBISHI SEMICONDUCTOR SYSTEM CORPORATION
*     and Mitsubishi Electric and Electronics USA
*=======================================================================
*      $Log:$
*=======================================================================*/
#include "sfr62.h"
#define  TIME_CONFIG 0x01  /*  00000001 value to load into timer mode register
                               ||||||||_  TMOD0,TMOD1: EVENT COUNTER MODE
                               ||||||____ MR0:         NO PULSE OUTPUT
                               |||||_____ MR1:         COUNT FALLING EDGES
                               ||||_______MR2:         USE UP/DOWN FLAG
                               |||________ MR3:         = 0 IN EVENT COUNTER MODE
                               ||_________ TCK0:        RELOAD TYPE
                               |__________TCK1:        BIT NOT USED
*/
```

```
#define  CNTR_IPL 0x03                // TA0  priority interrupt level
#define  LED  p7_7                    //p6_0  LED port on MSV1632 board
#define  LED_PORT_DIRECTION  pd7_7    //pd6_0 LED port direction on MSV1632 board
#define  OUTPUT 1



//prototypes
 void init(void);

#pragma INTERRUPT /B TimerA0Int
 void TimerA0Int(void);

/***********************************************************************
Name:     TimerA0Int()
Parameters: none
Returns:  nothing
Description:Timer A0 Interrupt Service Routine. Interrupts  every 100 falling
            edges on the TA0in pin. Flashes the LED and increments 'count'.
************************************************************************ */

void TimerA0Int(void)
{
  int delaycntr;
  delaycntr = 0;
  count++;               // e.g for an automated packaging line, counts # of cases
  LED = 1;
  while( delaycntr <0xffff)  //software delay for flashing LED
       delaycntr++;
  LED = 0;


}
/***********************************************************************
Name:    main()
Parameters: none
Returns:  nothing
Description: initializes variables and LED port. Then does nothing but
             wait for TA0 interrupts.
************************************************************************ */
void main (void)
{ int temp;
  count = 0;
  LED_PORT_DIRECTION = OUTPUT;
  init();
  while (1);

}
```

```
/************************************************************************
Name:  initial()
Parameters: none
Returns:  nothing
Description: Timer TA0 setup for 5msec interrupts.
************************************************************************ */
 void init()
  {
   ta0 = 100;    //e.g for an automated packaging line, 100 items per cases

/* 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
   ta0ic |= CNTR_IPL;          // use read-modify-write instruction to write IPL
   ta0mr = TIME_CONFIG;
   _asm ("   fset i");

   ta0s = 1;  //start counting
  }
```

In order for this program to run properly, timer A0'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 "_TimerA0Int" into the interrupt vector table at vector 21 as shown below.

```
;*************************************************************************
;
;      C Compiler for M16C/62
;
;   Copyright,2000 MITSUBISHI ELECTRIC CORPORATION
;      AND MITSUBISHI SEMICONDUCTOR SYSTEM CORPORATION
;       and Mitsubishi Electric and Electronics USA
;       All Rights Reserved.
;
;      Written by T.Aoyama
;      Modified  for use on MSV1632 Starter Kit.
;       sect30.inc     : section definition
;       This program is applicable when using KD30 and the ROM Monitor.
;*************************************************************************
;-----------------------------------------------------------------
                      :
                      :
                      :
                      :
                      :
```

```
.lword      dummy_int          ; A-D(for user)(vector 14)
.lword      dummy_int          ; uart2 transmit(for user)(vector 15)
.lword      dummy_int          ; uart2 receive(for user)(vector 16)
.lword      dummy_int          ; uart0 transmit(for user)(vector 17)
.lword      dummy_int          ; uart0 receive(for user)(vector 18)
.lword      0fcb6bh            ; uart1 transmit(for user)(vector 19)
.lword      0fcb6bh            ; uart1 receive(for user)(vector 20)
.glb      _ TimerA0Int
.lword      _TimerA0Int        ; 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)
.lword      dummy_int          ; 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)
              :
              :
              :
              :
              :
```

RENESAS
Renesas Technology Corp.

Keep safety first in your circuit designs!

- Mitsubishi Electric Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always a possibility of trouble occurring. 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 backup, auxiliary circuits, (ii) use of non-flammable material, or (iii) prevention against any malfunction or mishap.

Notes regarding this information

- This information is intended as a reference to assist our consumers in the selection of the Mitsubishi semiconductor product best suited to the customer's application; it does not convey any license under any intellectual property rights, or any other rights, belonging to Mitsubishi Electric Corporation or a third party.
- Mitsubishi Electric 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 is subject to change by Mitsubishi Electric Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for the latest product information before purchasing a product listed herein.
- The information described herein may contain technical inaccuracies or typographical errors. Mitsubishi Electric 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 Mitsubishi Electric Corporation by various means, including the Mitsubishi Semiconductor home page (http://www.mitsubishichips.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. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability, or other loss resulting from the information contained herein.
- Mitsubishi Electric 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 Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor 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 Mitsubishi Electric Corporation is necessary to reprint or reproduce these materials in whole or in part.
- If these products or technologies are subject to 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 Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for further details on these materials or the products contained therein.

RENESAS
Renesas Technology Corp.

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

USA Headquarters

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

1050 East Arques Avenue
Sunnyvale, CA 94085-4601
Phone: 408-730-5900
FAX: 408-732-9382

Direct Sales Offices – USA

Northwest

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

1050 East Arques Avenue
Sunnyvale, CA 94085-4601
Phone: 408-730-5900
FAX: 408-737-1129

Southwest

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

20 Fairbanks, Suite 181
Irvine, CA 92618
Phone: 949-859-9453
FAX: 949-859-9450

Rocky Mountain

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

PO Box 273377
Fort Collins, CO 80527
Phone: 970-669-4068
FAX: 970-669-4129

North Central

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

3300 Edinborough Way, Suite 511
Edina, MN 55435
Phone: 952-837-9053
FAX: 952-837-9059

Mitsubishi Electric & Electronics USA, Inc.
(Automotive Market Only)

50 West Big Beaver Rd., Suite 136
Troy, MI 48084
Phone: 248-526-9580
FAX: 248-526-9583

South Central

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

8310 Capital of Texas Hwy. N.,
Suite 260
Austin, TX 78731
Phone: 512-346-4200
FAX: 512-346-4434

Northeast

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

92 Montvale Avenue, Suite 2500
Stoneham, MA 02180
Phone: 781-245-6956
FAX: 781-245-4233

Mid-Atlantic

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

92 Montvale Avenue, Suite 2500
Stoneham, MA 02180
Phone: 781-245-6956
FAX: 781-245-4233

South Atlantic

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

2625 Meridian Parkway, Suite 100
Durham, NC 27713
Phone: 919-767-7900
FAX: 919-767-7902

Southeast

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

2810 Premiere Parkway, Suite 400
Duluth, GA 30097
Phone: 678-258-4518
FAX: 678-258-4519

Mexico Inquiries

Western Mexico

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

20 Fairbanks, Suite 181
Irvine, CA 92618
Phone: 949-859-9453
FAX: 949-859-9450

Eastern Mexico

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

8310 Capital of Texas Hwy. N.,
Suite 260
Austin, TX 78731
Phone: 512-346-4200
FAX: 512-346-4434

Puerto Rico Inquiries

Puerto Rico

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

2810 Premiere Parkway, Suite 400
Duluth, GA 30097
Phone: 678-258-4518
FAX: 678-258-4519

Mitsubishi Electric Sales Canada, Inc.

Direct Sales Offices – Canada

Central & Western Canada

Mitsubishi Electric Sales Canada, Inc.

4299 14th Avenue
Markham, Ontario,
Canada L3R 0J2
Phone: 905-475-7728
FAX: 905-475-1918

Eastern Canada

Mitsubishi Electric Sales Canada, Inc.

340 March Road, Suite 300
Kanata, Ontario,
Canada K2K 2E4
Phone: 613-591-3348
FAX: 613-591-3948

JQA-QS0161

Mitsubishi Electric MCU products sold into the Automotive marketplace are QS-9000 certified.

© 2002 Mitsubishi Electric & Electronics USA, Inc., Electronic Device Group.

Mitsubishi Electric & Electronics USA, Inc. is a wholly owned subsidiary of Mitsubishi Electric Corporation. All trademarks are the property of their respective owners. The information supplied by Mitsubishi Electric & Electronics USA, Inc. is believed to be accurate and reliable but in no event shall Mitsubishi Electric & Electronics USA, Inc. be liable for any damages whatsoever arising out of the use or inability to use the information or any errors that may appear in this publication.

THE INFORMATION IS PROVIDED AS IS WITHOUT ANY WARRANTIES OF ANY KIND, EITHER EXPRESS OR IMPLIED. MITSUBISHI ELECTRIC & ELECTRONICS USA, INC. RESERVES THE RIGHT, WITHOUT NOTICE, TO MAKE CHANGES TO THE INFORMATION OR TO THE DESIGN AND SPECIFICATIONS OF ITS HARDWARE AND/OR SOFTWARE PRODUCTS. PRODUCTS SUBJECT TO AVAILABILITY.

Printed in U.S.A.

www.mitsubishichips.com