EVM Application #6

Measuring the Period of an Input Square Wave Using the TMS320F240 EVM

APPLICATION REPORT: SPRA415

David Figoli

Digital Signal Processing Solutions
January 1999

TEXAS INSTRUMENTS

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE (“CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty, or endorsement thereof.

TRADEMARKS

TI is a trademark of Texas Instruments Incorporated.

Other brands and names are the property of their respective owners.

CONTACT INFORMATION

US TMS320 HOTLINE	(281) 274-2320
US TMS320 FAX	(281) 274-2324
US TMS320 BBS	(281) 274-2323
US TMS320 email	dsph@ti.com

Contents

EVM Application #6

Measuring the Period of an Input Square Wave Using the TMS320F240 EVM

Abstract

This EVM application measures the elapsed time between two rising edges of an input square wave using the capture input module in the Event Manager. The elapsed time (in microseconds) is output to a variable that can be viewed in the debugger environment. This application is written in C2xx Assembly code. The algorithm described in this application report was implemented using the Texas Instruments (TI™) TMS320F240 Evaluation Module (EVM).

The specific topics discussed include:

- PLL module
- Digital I/O ports
- Event Manager

Product Support

World Wide Web

Our World Wide Web site at www.ti.com contains the most up to date product information, revisions, and additions. Users registering with TI&ME can build custom information pages and receive new product updates automatically via email.

Email

For technical issues or clarification on switching products, please send a detailed email to dsph@ti.com. Questions receive prompt attention and are usually answered within one business day.

Overview

This application measures the elapsed time between two rising edges of an input square wave using the capture input module in the Event Manager. The elapsed time (in microseconds) is output to a variable that can be viewed in the debugger environment. This application is written in C2xx Assembly code. The algorithm described in this application report was implemented using the TI TMS320F240 EVM.

To view the results, the following commands need to be entered into the debugger environment when the program is loaded.

1) ba STOP

2) wa *USPERIOD,,u

The measured period will appear in a watch window for the variable USPERIOD.

Modules Used

- Event Manager
- General Purpose Timer 2
- Capture Input 1

Input

CAP1/QEP1

Output

None

Background and Methodology

The initial setup of the program is similar to previous applications. The PLL module, digital I/O ports, and the Event Manager all need to be configured before the program can be executed.

Because this program calculates the period of a signal, the PLL module needs to be set up so that the time base for the calculation of the period can be done correctly. The digital I/O port needs to be configured because the Event Manager Capture inputs share their pins with general I/O pins.

PLL Module

The PLL module is set up as in Application #1 (PWM0.ASM). The frequency that the CPUCLK is set to be is important in order to determine the period of the input square wave. The PLL divide by 2 is enabled and the multiplication ratio is set to 4; as a result, with a CLKIN (crystal oscillator) value of 10MHz, the CPUCLK is 20MHz. Therefore, the CPUCLK period is 0.05 μs.

Digital I/O Ports

Similar to the PWM applications where Output Control Register A (OCRA) had to be set so that the PWM signal could be output on the proper pins, Output Control Register B (OCRB) has to be set so the pins that the capture inputs share with the I/O pins are configured as capture inputs.

Event Manager

Once the digital I/O port and the PLL module have been set, then the Event Manager module registers can be configured to capture the rising edges of an input square wave. Since only one input capture port will be used, only one GP Timer in the Event Manager will be needed. In this application, the input will be placed into capture input 1. Since the capture units of the event manager can only use GP Timer 2 or 3, one of the timers needs to be selected. For this application, GP Timer 2 was selected as the timer that the capture input 1 will use.

Once the Capture and Timer units of the Event Manager have been set, the program can be executed. To obtain a reliable period measurement, 5 measurements are captured. The first value is discarded because of synchronization error that typically occurs. In the first capture, the timer is started and then the interrupts are enabled; as a result, the delay in enabling the interrupts after starting the timer could result in an invalid value because a rising edge could have occurred during the latency. The next four values are summed and averaged. The resulting average value is multiplied by the equivalent Q15 value for the period of the 20MHz CPUCLK. The final product is stored into the location USPERIOD and can be viewed in a watch window.

Since only one timer module will be used, the resolution that this application will be able to perform is limited by the size of the counter. In theory, the range of the frequencies that this application will be able to work with is bounded by the maximum and minimum value of the counter. Therefore, with a 16bit counter, the theoretical resolution of this application with a 20MHz CPUCLK running the counter, will be 20MHz (20Mhz/1) to 305Hz (20MHz/FFFFh).

However, because there is latency (28 cycles) associated with the interrupt service routine, the maximum frequency that can be measured is about 700kHz.

Event manager interrupt	1 cycle
Flush Pipeline, Check Vector Table	4 cycles
Branch to ISR	4 cycles

```
CAP_ISR  LDP   #232           2 cycles
         LACC  CAP1FIFO       1 cycle
         MAR   *,AR1          1 cycle
         SACL  *+,0,AR2       1 cycle
         LACC  *              1 cycle
         SUB   #1             1 cycle
         SACL  *,0,AR1        1 cycle
         BCND  LEAVE,EQ       2 cycles

         SPLK  #0000h,T2CNT   2 cycles
         LACC  EVIFRC         1 cycle
         SACL  EVIFRC         1 cycle

         CLRC  INTM           1 cycle

         RET                  4 cycle
```

Latency = 28 cycles

$$\text{Max Frequency} = \frac{1}{\text{cycles} \times \text{CPUCLK period}} = \frac{1}{28 \times 50\text{ns}} \approx 714{,}000\text{Hz}$$

Using this application to measure the period at the maximum and minimum frequencies becomes somewhat unreliable. At the maximum frequency, the interrupt service latency becomes a problem. At the minimum frequency, the rollover of the 16 bit counter becomes and issue. This application can measure with some accuracy periods equivalent to the frequencies in the range of 500Hz to 10kHz; however, as one measures higher frequencies, the accuracy decreases because of the ISR latency.

Once this program is executed, it will stop at the break point that was set at STOP. By restarting the program in debugger environment, another measurement can be taken. The results will show up in the watch window next to USPERIOD.

```
;*****************************************************************
; File Name:        cap0.asm
; Originator:       Digital control systems Apps group - Houston
; Target System: 'C24x Evaluation Board
;
; Description:   Capture Input of the Event Manager Module
;                is set up to measure the time elapsed between 2
;                rising edges of an input square wave
;
;                To view the results in the debugger environment the
;                following commands need to be entered before the
;                program is run
;                      ba STOP
;                      wa *USPERIOD,,u
;
;                The value in USPERIOD is the length of the period
;                in us. The value will be in base 10.
;
; Last Updated: 20 June 1997
;
;*****************************************************************
                .include f240regs.h
```

```
;-----------------------------------------------------------------------
; Vector address declarations
;-----------------------------------------------------------------------
                .sect ".vectors"

RSVECT          B    START    ; Reset Vector
INT1            B    PHANTOM  ; Interrupt Level 1
INT2            B    PHANTOM  ; Interrupt Level 2
INT3            B    PHANTOM  ; Interrupt Level 3
INT4            B    CAP_ISR  ; Interrupt Level 4
INT5            B    PHANTOM  ; Interrupt Level 5
INT6            B    PHANTOM  ; Interrupt Level 6
RESERVED        B    PHANTOM  ; Reserved
SW_INT8         B    PHANTOM  ; User S/W Interrupt
SW_INT9         B    PHANTOM  ; User S/W Interrupt
SW_INT10        B    PHANTOM  ; User S/W Interrupt
SW_INT11        B    PHANTOM  ; User S/W Interrupt
SW_INT12        B    PHANTOM  ; User S/W Interrupt
SW_INT13        B    PHANTOM  ; User S/W Interrupt
SW_INT14        B    PHANTOM  ; User S/W Interrupt
SW_INT15        B    PHANTOM  ; User S/W Interrupt
SW_INT16        B    PHANTOM  ; User S/W Interrupt
TRAP            B    PHANTOM  ; Trap vector
NMINT           B    PHANTOM  ; Non-maskable Interrupt
EMU_TRAP        B    PHANTOM  ; Emulator Trap
SW_INT20        B    PHANTOM  ; User S/W Interrupt
SW_INT21        B    PHANTOM  ; User S/W Interrupt
SW_INT22        B    PHANTOM  ; User S/W Interrupt
SW_INT23        B    PHANTOM  ; User S/W Interrupt
```

```
;=====================================================================
; M A I N   C O D E  - starts here
;=====================================================================
                .text
                NOP
START:          SETC  INTM          ;Disable interrupts
                SPLK  #0008h,IMR    ;Mask all core interrupts
                                       ;except INT4

                LACC  IFR           ;Read Interrupt flags
                SACL  IFR           ;Clear all interrupt flags

                CLRC  SXM           ;Clear Sign Extension Mode
                CLRC  OVM           ;Reset Overflow Mode
                CLRC  CNF           ;Config Block B0 to Data mem

;-----------------------------------
; Set up PLL Module
;-----------------------------------
                LDP   #00E0h

;The following line is necessary if a previous program set the PLL
;to a different setting than the settings which the application
;uses. By disabling the PLL, the CKCR1 register can be modified so
;that the PLL can run at the new settings when it is re-enabled.

                SPLK  #0000000001000001b,CKCR0
                                ;CLKMD=PLL Disable,SYSCLK=CPUCLK/2

;                      5432109876543210
                SPLK  #0000000010111011b,CKCR1
;                                  ;CLKIN(OSC)=10MHz,CPUCLK=20MHz

;CKCR1 - Clock Control Register 1
;Bits 7-4        (1011)CKINF(3)-CKINF(0) - Crystal or Clock-In
;                        Frequency
;                        Frequency = 10MHz
;Bit 3           (1)   PLLDIV(2) - PLL divide by 2 bit
;                        Divide PLL input by 2
;Bits 2-0        (011) PLLFB(2)-PLLFB(0) - PLL multiplication ratio
;                        PLL Multiplication Ration = 4

;                      5432109876543210
                SPLK  #0000000011000001b,CKCR0
                                ;CLKMD=PLL Enable,SYSCLK=CPUCLK/2

;CKCR0 - Clock Control Register 0
;Bits 7-6        (11)  CLKMD(1),CLKMD(0) - Operational mode of Clock
;                        Module PLL Enabled
;                        Run on CLKIN on exiting low power mode
;Bits 5-4        (00)  PLLOCK(1),PLLOCK(0) - PLL Status. READ ONLY
;Bits 3-2        (00)  PLLPM(1),PLLPM(0) - Low Power Mode
```

```
;                          LPM0
;Bit 1           (0)   ACLKENA - 1MHz ACLK Enable
;                          ACLK Disabled
;Bit 0           (1)   PLLPS - System Clock Prescale Value
;                          f(sysclk)=f(cpuclk)/2

;                         5432109876543210
                 SPLK  #0100000011000000b,SYSCR ;CLKOUT=CPUCLK

;SYSCR - System Control Register
;Bit 15-14       (01)    RESET1,RESET0 - Software Reset Bits
;                           No Action
;Bits 13-8       (000000) Reserved
;Bit 7-6         (11)    CLKSRC1,CLKSRC0 - CLKOUT-Pin Source Select
;                           CPUCLK: CPU clock output mode
;Bit 5-0         (000000) Reserved


                 SPLK  #006Fh, WDCR  ;Disable WD if VCCP=5V (JP5
                                        ;in pos. 2-3)
                 KICK_DOG            ;Reset Watchdog

;-----------------------------------
; Set up Digital I/O Port
;-----------------------------------
                 LDP   #225  ;DP=225, Data Page to Configure OCRA
;                         5432109876543210
                 SPLK  #0011100000000000b,OCRA

;OCRA  - Output Control Register A
;                 Bit 15   (0)      CRA.15 - IOPB7
;                 Bit 14   (0)      CRA.14 - IOPB6
;                 Bit 13   (1)      CRA.13 - T3PWM/T3CMP
;                 Bit 12   (1)      CRA.12 - T2PWM/T2CMP
;                 Bit 11   (1)      CRA.11 - T1PWM/T1CMP
;                 Bit 10   (0)      CRA.10 - IOPB2
;                 Bit 9    (0)      CRA.9 - IOPB1
;                 Bit 8    (0)      CRA.8 - IOPB0
;                 Bits 7-4 (0000)   Reserved
;                 Bit 3    (0)      CRA.3 - IOPA3
;                 Bit 2    (0)      CRA.2 - IOPA2
;                 Bit 1    (0)      CRA.1 - IOPA1
;                 Bit 0    (0)      CRA.0 - IOPA0

;                         76543210
                 SPLK  #11110000b,OCRB

;OCRB - Output Control Register B
;                 Bit 7 (1)   CRB.7 - CAP4
;                 Bit 6 (1)   CRB.6 - CAP3
;                 Bit 5 (1)   CRB.5 - CAP2/QEP2
;                 Bit 4 (1)   CRB.4 - CAP1/QEP1
```

```
;                 Bit 3 (0)   CRB.3 - BIO
;                 Bit 2 (0)   CRB.2 - XF
;                 Bit 1 (0)   CRB.1 - ?
;                 Bit 0 (0)   CRB.0 - IOPC0

;*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-**-*-*-*-*-*-*-*-*-*-*-*-*-
;- Event Manager Module Reset
;*
;-      This section resets all of the Event Manager Module
;*      Registers. This is necessary for silicon revision 1.1, but
;-      not for silicon revisions 2.0 and later.
;*
;-
;*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-**-*-*-*-*-*-*-*-*-*-*-

     LDP         #232        ;DP=232 Data Page for the Event Manager
     SPLK        #0000h,GPTCON ;Clear General Purpose Timer Control
     SPLK        #0000h,T1CON  ;Clear GP Timer 1 Control
     SPLK        #0000h,T2CON  ;Clear GP Timer 2 Control
     SPLK        #0000h,T3CON  ;Clear GP Timer 3 Control

     SPLK        #0000h,COMCON ;Clear Compare Control
     SPLK        #0000h,ACTR   ;Clear Full Compare Action Control
;                                    Register
     SPLK        #0000h,SACTR  ;Clear Simple Compare Action
;                                    Control Register
     SPLK        #0000h,DBTCON ;Clear Dead-Band Timer Control
;                                    Register

     SPLK        #0000h,CAPCON ;Clear Capture Control

     SPLK        #0FFFFh,EVIFRA;Clear Interrupt Flag Register A
     SPLK        #0FFFFh,EVIFRB;Clear Interrupt Flag Register B
     SPLK        #0FFFFh,EVIFRC;Clear Interrupt Flag Register C

     SPLK        #0000h,EVIMRA ;Clear Event Manager Mask Register A
     SPLK        #0000h,EVIMRB ;Clear Event Manager Mask Register B
     SPLK        #0000h,EVIMRC ;Clear Event Manager Mask Register C

;*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-**-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-
;-End of RESET section for silicon revision 1.1          *
;*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-**-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-

;-----------------------------------
; Set up Event Manager Module
;-----------------------------------

; Initialize the period register of Timer 2 to allow for
; the ability to count to the maximum value

T2COMPARE        .set  7FFFh ; T2Compare half of max value
T2PERIOD         .set  0FFFFh; T2Period Initialized to the max value
```

```
                .text
                LDP   #232              ;DP=232, Data Page for
                                        ;Event Manager Addresses

                SPLK  #T2COMPARE,T2CMPR   ;T2CMPR = T2PERIOD/2

;                      2109876543210
                SPLK  #0000001010101b,GPTCON

;GPTCON - GP Timer Control Register
;     Bit 15     (0)   T3STAT - GP Timer 3 Status.  READ ONLY
;     Bit 14     (0)   T2STAT - GP Timer 2 Status.  READ ONLY
;     Bit 13     (0)   T1STAT - GP Timer 1 Status.  READ ONLY
;     Bits 12-11 (00)  T3TOADC - ADC start by event of GP Timer 3
;                         No event starts ADC
;     Bits 10-9  (00)  T2TOADC - ADC start by event of GP Timer 2
;                         No event starts ADC
;     Bits 8-7   (00)  T1TOADC - ADC start by event of GP Timer 1
;                         No event starts ADC
;     Bit 6      (1)   TCOMPOE - Compare output enable
;                         Enable all three GP timer compare outputs
;     Bits 5-4   (01)  T3PIN - Polarity of GP Timer 3 compare output
;                         Active Low
;     Bits 3-2   (01)  T2PIN - Polarity of GP Timer 2 compare output
;                         Active Low
;     Bits 1-0   (01)  T1PIN - Polarity of GP Timer 1 compare output
;                            Active Low

                SPLK  #T2PERIOD,T2PR   ; T2PR = FFFFh
                SPLK  #0000h,T2CNT     ; Initialize Timer 2

;                      5432109876543210
                SPLK  #0000000000000100b,T1CON  ;Not Used

;T1CON - GP Timer 1 Control Register
;     Bits 15-14 (00)  FREE,SOFT - Emulation Control Bits
;                         Stop immediately on emulation suspend
;     Bits 13-11 (000) TMODE2-TMODE0 - Count Mode Selection
;                         Stop/Hold
;     Bits 10-8  (000) TPS2-TPS0 - Input Clock Prescaler
;                         Divide by 1
;     Bit 7      (0)   Reserved
;     Bit 6      (0)   TENABLE - Timer Enable
;                         Disable timer operations
;     Bits 5-4   (00)  TCLKS1,TCLKS0 - Clock Source Select
;                         Internal Clock Source
;     Bits 3-2   (00)  TCLD1,TCLD0 - Timer Compare Register
;                         Reload Condition
;                         When counter is 0
;     Bit 1      (0)   TECMPR - Timer compare enable
;                         Disable timer compare operation
```

```
;     Bit 0      (0)   Reserved

;                       5432109876543210
                SPLK  #0001000000000110b,T2CON

;T2CON - GP Timer 2 Control Register
;     Bits 15-14 (00)  FREE,SOFT - Emulation Control Bits
;                         Stop immediately on emulation suspend
;     Bits 13-11 (010) TMODE2-TMODE0 - Count Mode Selection
;                         Continuous Count-Up Mode
;     Bits 10-8  (000) TPS2-TPS0 - Input Clock Prescaler
;                         Divide by 1
;     Bit 7      (0)   TSWT1 - GP Timer 1 timer enable bit
;                         Use own TENABLE bit
;     Bit 6      (0)   TENABLE - Timer Enable
;                         Disable timer operations
;     Bits 5-4   (00)  TCLKS1,TCLKS0 - Clock Source Select
;                         Internal Clock Source
;     Bits 3-2   (01)  TCLD1,TCLD0 - Timer Compare Register
;                            Reload Condition
;                         When counter is 0 or equals period
;                            register value
;     Bit 1      (1)   TECMPR - Timer compare enable
;                         Enable timer compare operation
;     Bit 0      (0)   SELT1PR - Period Register select
;                         Use own period register

;                       5432109876543210
                SPLK  #0000000000000000b,T3CON  ;Not Used

;T3CON - GP Timer 3 Control Register
;     Bits 15-14 (00)  FREE,SOFT - Emulation Control Bits
;                         Stop immediately on emulation suspend
;     Bits 13-11 (000) TMODE2-TMODE0 - Count Mode Selection
;                         Stop/hold
;     Bits 10-8  (000) TPS2-TPS0 - Input Clock Prescaler
;                         Divide by 1
;     Bit 7      (0)   TSWT1 - GP Timer 1 timer enable bit
;                         Use own TENABLE bit
;     Bit 6      (0)   TENABLE - Timer Enable
;                         Disable timer operations
;     Bits 5-4   (00)  TCLKS1,TCLKS0 - Clock Source Select
;                         Internal Clock
;     Bits 3-2   (00)  TCLD1,TCLD0 - Timer Compare Register
;                            Reload Condition
;                         When counter is 0
;     Bit 1      (0)   TECMPR - Timer compare enable
;                         Disable timer compare operation
;     Bit 0      (0)   SELT1PR - Period Register select
;                         Use own period register
```

```
;                      5432109876543210
                SPLK  #1011110001010101b,CAPCON

;CAPCON - Capture Control Register
;     Bit 15     (0)   CAPRES - Capture Reset
;                         Clear all registers of capture units and
;                         QEP circuits to 0
;     Bits 14-13 (01)  CAPQEPN - Capture Units 1 & 2 and QEP Circuit
;                         Control
;                         Enable Capture Units 1 & 2. Disable QEP
;                         Circuit
;     Bit 12     (1)   CAP3EN - Capture Unit 3 Control
;                         Enable Capture Unit 3
;     Bit 11     (1)   CAP4EN - Capture Unit 4 Control
;                         Enable Capture Unit 4
;     Bit 10     (1)   CAP34TSEL - GP Timer Selection for Capture
;                         Units 3 & 4
;                         Select GP Timer 3
;     Bit 9      (0)   CAP12TSEL - GP Timer Selection for Capture
;                         Units 1 & 2
;                         Select GP Timer 2
;     Bit 8      (0)   CAP4TOADC - Capture Unit 4 starts ADC
;                         No Action
;     Bits 7-6   (01)  CAP1EDGE - Edge Detection for Capture Unit 1
;                         Detect Rising Edge
;     Bits 5-4   (01)  CAP2EDGE - Edge Detection for Capture Unit 2
;                         Detect Rising Edge
;     Bits 3-2   (01)  CAP3EDGE - Edge Detection for Capture Unit 3
;                         Detect Rising Edge
;     Bits 0-1   (01)  CAP4EDGE - Edge Detection for Capture Unit 4
;                         Detect Rising Edge

;                       876543210
                SPLK  #011111111b,CAPFIFO

;CAPFIFO - Capture FIFO Status Register
;     Bits 15-14       CAP4FIFO Status - READ ONLY
;     Bits 13-12       CAP3FIFO Status - READ ONLY
;     Bits 11-10       CAP2FIFO Status - READ ONLY
;     Bits 9-8         CAP1FIFO Status - READ ONLY
;     Bit 7      (1)   CAPFIFO15 - CAP4FIFO bit 15 Clear
;                         Clear Bit 15 of CAPFIFO
;     Bit 6      (1)   CAPFIFO14 - CAP4FIFO bit 14 Clear
;                         Clear Bit 14 of CAPFIFO
;     Bit 5      (1)   CAPFIFO13 - CAP3FIFO bit 13 Clear
;                         Clear Bit 13 of CAPFIFO
;     Bit 4      (1)   CAPFIFO12 - CAP3FIFO bit 12 Clear
;                         Clear Bit 12 of CAPFIFO
;     Bit 3      (1)   CAPFIFO11 - CAP2FIFO bit 11 Clear
;                         Clear Bit 11 of CAPFIFO
;     Bit 2      (1)   CAPFIFO10 - CAP2FIFO bit 10 Clear
;                         Clear Bit 10 of CAPFIFO
```

```
;     Bit 1      (1)   CAPFIFO9 - CAP1FIFO bit 9 Clear
;                        Clear Bit 9 of CAPFIFO
;     Bit 0      (1)   CAPFIFO8 - CAP1FIFO bit 8 Clear
;                        Clear Bit 8 of CAPFIFO


;                       3210
                SPLK  #0001b,EVIMRC

;EVIMRC - EV Interrupt Mask Register C
;     Bits 15-4          Reserved
;     Bit 3      (0)   CAP4INT Enable
;                        Disable
;     Bit 2      (0)   CAP3INT Enable
;                        Disable
;     Bit 1      (0)   CAP2INT Enable
;                        Disable
;     Bit 0      (1)   CAP1INT Enable
;                        Enable


;-----------------------------------------------------------------------
;               VARIABLES FOR CAP_ISR
;-----------------------------------------------------------------------

        .bss    VALUE1,1    ;1st timer value for 1st rising edge
        .bss    VALUE2,1    ;2nd timer value for 2nd rising edge
        .bss    VALUE3,1    ;3nd timer value for 3nd rising edge
        .bss    VALUE4,1    ;4nd timer value for 4nd rising edge
        .bss    VALUE5,1    ;5nd timer value for 5nd rising edge
        .bss    AVERAGE,1   ;Average of VALUE2 through VALUE5
        .bss    COUNTER,1   ;Counter to acquire 5 values
        .bss    USPERIOD,1  ;Period of the 2 rising edges
        .bss    CLKPERIOD,1;Q15 value for the period of the CPUCLK

        .text

        LAR     AR1,#VALUE1      ;AR1 = address of VALUE1
        LAR     AR2,#COUNTER     ;AR2 = address of COUNTER

        LDP     #0
        SPLK    #0000h,VALUE1    ;Initialize VALUE1
        SPLK    #0000h,VALUE2    ;Initialize VALUE2
        SPLK    #0000h,VALUE3    ;Initialize VALUE3
        SPLK    #0000h,VALUE4    ;Initialize VALUE4
        SPLK    #0000h,VALUE5    ;Initialize VALUE5
        SPLK    #0000h,AVERAGE   ;Initialize AVERAGE
        SPLK    #0005h,COUNTER   ;Counter set to acquire 5 values
        SPLK    #0000h,USPERIOD  ;Initialize USPERIOD
        SPLK    #0666h,CLKPERIOD
                         ;Q15 value of 0.05us = T => 20MHz CPUCLK
```

```
        LDP     #232
        LACC    EVIFRC          ;ACC = Interrupt Flags of EVIFRC
        SACL    EVIFRC          ;EVIFRC = ACC => clears all flags

        LDP     #0
        LACC    IFR             ;ACC = Interrupt Flags of IFR
        SACL    IFR             ;IFR = ACC => clears all flags

        LDP     #232
        SBIT1   T2CON,B6_MSK    ;Sets Bit 6 of T2CON

;TxCON - GP Timer x Control Register
;     Bit 6     (1)   TENABLE - Timer Enable
;                         Enable Timer Operations

        CLRC    INTM            ;Enable Interrupts

WAIT    B       WAIT            ;Wait for an interrupt


;-------------------------------------------------------------------
; CAPTURE INTERRUPT SERVICE ROUTINE
;-------------------------------------------------------------------

CAP_ISR LDP     #232        ;DP = 232, Data Page for Event
                               ;Manager Registers
        LACC    CAP1FIFO    ;ACC = value of GP Timer at capture
        MAR     *,AR1       ;ARP = AR1
        SACL    *+,0,AR2    ;VALUEx= value of GP Timer from
                               ;CAP1FIFO, ARP = AR2
        LACC    *           ;ACC = COUNTER
        SUB     #1          ;Decrement Counter
        SACL    *,0,AR1     ;Store new value of Counter, APR = AR1
        BCND    LEAVE,EQ    ;Exit if all values have been captured

        SPLK    #0000h,T2CNT  ;Else, Clear the counter
        LACC    EVIFRC        ;Clear the EVIFRC interrupt flags
        SACL    EVIFRC

        CLRC    INTM    ;Enable interrupts

        RET             ;Return to capture the next rising edge

LEAVE   LDP     #0      ;DP = 0, Data Page for the acquired values
        LACC    VALUE2  ;ACC = VALUE2
        ADD     VALUE3  ;ACC = VALUE2 + VALUE3
        ADD     VALUE4  ;ACC = VALUE2 + VALUE3 + VALUE4
        ADD     VALUE5  ;ACC = VALUE2 + VALUE3 + VALUE4 + VALUE5


        SFR             ;Shift ACC right = Divide by 2
        SFR                ;Shift ACC right = Divide by 2
```

```
        SACL    AVERAGE     ;Store value into AVERAGE

        LT      AVERAGE     ;TREG = AVERAGE
        MPYU    CLKPERIOD   ;PREG = AVERAGE(Q0)*CLKPERIOD(Q15)
        PAC                    ;ACC = PREG
        SACH    USPERIOD,1  ;USPERIOD = ACC(30:15),Shift left to
                               ; remove the extra sign bit

STOP    B       STOP        ;End the Program


;=====================================================================
; I S R  -  PHANTOM
;
; Description:   Dummy ISR, used to trap spurious interrupts.
;
; Modifies:      Nothing
;
; Last Update:   16 June 95
;=====================================================================
PHANTOM         KICK_DOG         ;Resets WD counter
                B  PHANTOM
```