

Description

The 16-bit timer/event counter (TM0) in the 7805x/78005x subseries can be used for an interval timer, external event counter, PWM output, square-wave output, one-shot pulse output or for pulse width measurement.

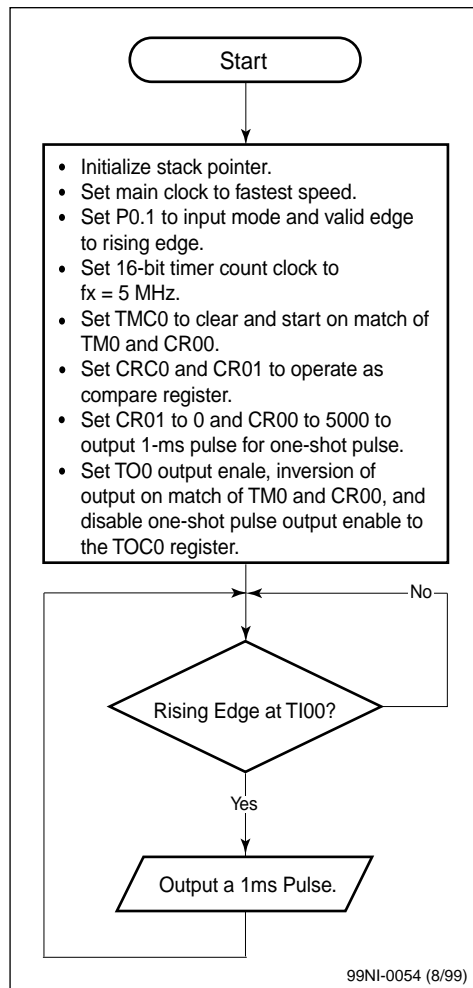
This program demonstrates how the 16-bit timer/event counter operates in the one-shot pulse mode with external trigger. Each time a rising edge occurs at timer input 0 (TI00/INTP0/P00), the timer 0 (TM0) outputs a 1 ms pulse at TO0/P03.

The pulse length is determined by the value set in capture/compare register 0 (CR00). Also by using capture/compare register 1 (CR01), the user can delay the start of the pulse.

Program Specifications

- Trigger: one-shot pulse through rising edge at TI00/INTP0
- Count clock: $f_x = 5 \text{ MHz}$
- One-shot length: 1 ms
- Pulse output delay: 0 μs
- Pins used in program
 - TI00/P00/INTP0: triggers a one-shot pulse
 - TO0/P30: timer output of the pulse

Flowchart



Assembly Language Program

```

;*****
; Date:          07/27/1999
;
; Parameters:   - fastest CPU clock,
;                (fx = 5 MHz; 1 CPU clock cycle = 200 ns
;                - Count clock:                fx = 5 MHz
;                - One-shot trigger source:     Rising edge at TI00
;                - One-shot pulse length:      1 ms
;                - Delay time before one-shot output: 0 µs
;                - T00/P30:                    Output the pulse
;
;*****
;=====
;=      Specify Interrupt Vectors      =
;=====
Res_Vec  CSEG AT 0000h          ; Set main program start vector.
        DW Start

;=====
;=      Main Program                    =
;=====
MAIN     CSEG
Start:   DI                    ; Disable interrupts
        MOVW    AX, #0FE20h    ; Load SP address
        MOVW    SP, AX        ; Set Stack Pointer
        MOV     OSMS, #01h     ; Don't use scaler
        MOV     PCC, #00h     ; Main system clock at fastest setting
        MOV     P0, #00h     ; Set port 0 latch to all 0
        MOV     PM0, #01h    ; Set port 0.0 as input
        MOV     P3, #00h     ; set port 3 latch to 0
        MOV     PM3, #00h    ; Set port 3 to output mode
        MOV     INTM0, #04h   ; Set TI00/INTP0 active with rising edge
        MOV     TCL0, #20h    ; Select counter clock to fx = 5 MHz
        MOVW    CR01, #0      ; Start output one-shot pulse immediately
                                ; after one-shot trigger is issued.
        MOVW    CR00, #5000   ; Set Compare register to 5000
                                ; 1 millisecond interval
        MOV     TOC0, #33h    ; Set T00 output enable, inversion of
                                ; output on match of TM0 and CR00, TM0 and CR01,
                                ; one-shot output mode enable

        MOV     TMC0, #08h;   ; Set to clear & start on match of TM0 and CR00
        CLR1    CRC0.0        ; Set CR00 operation to compare register
        CLR1    CRC0.2        ; set CR01 operation to compare register
Loop:    NOP                  ; Needed NOP for interrupt acknowledge
        BF     TMIF01, $Loop  ; Wait for triggering the pulse start
        CLR1    TMIF01        ; Clear pulse start interrupt flag
Loop2:   NOP                  ; Needed NOP for interrupt acknowledge
        BF     TMIF00, $Loop2 ; Wait for pulse end
        CLR1    TMIF00        ; Clear pulse end interrupt flag
        BR     $Loop          ; Branch back
END

```

C Language Program

```

/*****
; Date:          08/03/1999
;
; Parameters:   - fastest CPU clock,
;               (fx = 5 MHz; 1 CPU clock cycle = 200 ns
;               - Count clock:                fx = 5 MHz
;               - One-shot trigger source:     Rising edge at TI00
;               - One-shot pulse length:      1 ms
;               - Delay time before one-shot output: 0 µs
;               - T00/P30:                    Output the pulse
;
; *****/
/* extension functions in K0/K0S compiler */
#pragma sfr /* key word to allow SFR names in C code */
#pragma asm /* key word to allow ASM statements in C code */

/*=====
; Constants/Variables =
;=====*/
#define TRUE      1
#define FALSE     0

/*=====
; Main Program =
;=====*/
void main(void)
{
    OSMS = 0x01; /* Don't use scaler */
    PCC = 0x00; /* Main system clock at fastest setting */
    P0 = 0; /* Latch port 0 to low
    PM0 = 0x01; /* Set port 0.0 as input to use one-shot repeat */
    P3 = 0; /* Latch port 3 low */
    PM3 = 0x00; /* Set port 3 as output */
    INTM0 = 0x04; /* Set TI00/INTP0 active with rising edge */
    TCL0 = 0x20; /* Select counter clock to fx = 5 MHz */
    CR01 = 00; /* Start output one-shot pulse immediately after
                one-shot trigger is issued. */
    CR00 = 5000; /* Set Compare register to 5000
                 1 millisecond interval */
    TOC0 = 0x33; /* Set T00 output enable, inversion of output
                 on match of TM0 and CR00, TM0 and CR01 and
                 one-shot output mode enable */
    TMC0 = 0x08; /* Set to clear & start on match TM0 and CR00 */
    CRC0.0 = 0; /* Set CR00 operation to compare register */
    CRC0.2 = 0; /* Set CR01 operation to compare register */
    while(TRUE) /* loop here */
    {
        while( !TMIF01); /* Wait for triggering the pulse start */
        TMIF01 = 0; /* Clear pulse start interrupt flag */
        while( !TMIF00); /* Wait for pulse end */
        TMIF00 = 0; /* Clear pulse end interrupt flag
    }
    /* End of while(TRUE) */
}
/* End of function main() */

```



For literature, call **1-800-366-9782** 7 a.m. to 6 p.m. Pacific time
or FAX your request to **1-800-729-9288**
or visit our web site at **www.necel.com**

In North America: No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics Inc. (NECEL). The information in this document is subject to change without notice. All devices sold by NECEL are covered by the provisions appearing in NECEL Terms and Conditions of Sales only. Including the limitation of liability, warranty, and patent provisions. NECEL makes no warranty, express, statutory, implied or by description, regarding information set forth herein or regarding the freedom of the described devices from patent infringement. NECEL assumes no responsibility for any errors that may appear in this document. NECEL makes no commitments to update or to keep current information contained in this document. The devices listed in this document are not suitable for use in applications such as, but not limited to, aircraft control systems, aerospace equipment, submarine cables, nuclear reactor control systems, and life support systems. "Standard" quality grade devices are recommended for computers, office equipment, communication equipment, test and measurement equipment, machine tools, industrial robots, audio and visual equipment, and other consumer products. For automotive and transportation equipment, traffic control systems, anti-disaster and anti-crime systems, it is recommended that the customer contact the responsible NECEL salesperson to determine the reliability requirements for any such application and any cost adder. NECEL does not recommend or approve use of any of its products in life support devices or systems or in any application where failure could result in injury or death. If customers wish to use NECEL devices in applications not intended by NECEL, customer must contact the responsible NECEL salespeople to determine NECEL's willingness to support a given application.