Order this document
by EB282/D

Motorola Semiconductor Engineering Bulletin

EB282

Using the Output Compare Function on the Time Processor Unit and an Example that Includes PPWA

By Sharon Darley
Austin, Texas

Introduction

The output compare function (OC) on the time processor unit (TPU) has three modes of operation:

- Read TCR1/TCR2
- Host-initiated pulse
- Continuous pulse

In the read TCR1/TCR2 mode of operation, the TPU reads the most recent values of TCR1 and TCR2 and returns them in memory locations $FFFFEC and $FFFFEE. In the host-initiated pulse mode, the TPU performs the same function as in the read TCR1/TCR2 mode of operation, except that it also immediately forces the pin high or low as specified in the PSC field. It then forces the pin again at a designated time in the future. In the continuous pulse mode, the TPU outputs a continuous train of 50% duty cycle pulses. The OC function can receive links (a link is a service request from another channel) only in the continuous mode.

For a detailed discussion of the OC function and all of its parameters, see *Output Compare TPU Function (OC)*, Motorola document order number TPUPN12/D. This document can be ordered from Literature Distribution Center or it can be accessed on the Worldwide Web at

MOTOROLA

http://www.mcu.motsps.com/lit/manuals/tpulitpak/pdf.html. The application note *Timing Performance of TPU I/O Hardware*, Motorola document order number AN1236/D, also has a detailed example of how to initialize the OC function.

Using OC Function

This section gives several examples of how to use the OC function in the host-initiated pulse mode and also in the continuous pulse mode. These examples were assembled on the IASM32 assembler, available from P&E Microsystems. They are written for CPU32-based microcontrollers such as the MC68332 and MC68333. Translating the examples to CPU16 code mainly involves changing the MOVE instructions to LDD and STD instructions. These examples use these equates:

```
TPUMCR          EQU             $FFFE00
TICR            EQU             $FFFE08
CIER            EQU             $FFFE0A
CISR            EQU             $FFFE20
CFSR3           EQU             $FFFE12
HSQR1           EQU             $FFFE16
CPR1            EQU             $FFFE1E
HSRR1           EQU             $FFFE1A
PRAM0_0         EQU             $FFFF00
PRAM0_1         EQU             $FFFF02
PRAM0_2         EQU             $FFFF04
PRAM0_3         EQU             $FFFF06
PRAM0_4         EQU             $FFFF08
PRAM1_0         EQU             $FFFF10
PRAM1_1         EQU             $FFFF12
PRAM1_2         EQU             $FFFF14
PRAM3_0         EQU             $FFFF30
PRAM3_2         EQU             $FFFF34
PRAM3_3         EQU             $FFFF36
```

Host-Initiated Pulse Mode

Example 1

This example initializes channel 0 in the host-initiated pulse mode. The pin will be forced high immediately upon initialization and will then be forced low $2000 TCR1 counts after initialization.

```
               ORG $400                     ;begin the program at $400
               MOVE.W #$0003,(CPR1).L       ;give channel high priority (CPR1)
               MOVE.W #$000E,(CFSR3).L      ;select OC function in CFSR3
               MOVE.W #$0000,(HSQR1).L      ;matches and pulses scheduled (HSQR1)
               MOVE.W #$0089,(PRAM0_0).L    ;PSC = high, PAC = low
               MOVE.W #$2000,(PRAM0_1).L    ;Offset = $2000
               MOVE.W #$00EC,(PRAM0_2).L    ;REF_ADDR1 = TCR1
               MOVE.W #$0001,(HSRR1).L      ;host service request for host-init mode
LOOP           BRA  LOOP
```

Example 2

This example initializes channel 0 in the host-initiated pulse mode. The output of the pin will be a continuous pulse train. The continuous train is accomplished by re-initializing the channel in the interrupt routine.

NOTE: *Make sure that the vector base register (VBR) is set to $400 before running this program*

```
               ORG $400
               MOVE.W #$00E0,(CFSR3).L      ;channel 0 output compare
               MOVE.W #$0000,(HSQR1).L      ;matches and pulses scheduled
               MOVE.W #$000C,(CPR1).L       ;high priority
               ANDI.W #$0000,(CISR).L       ;read and clear interrupt status register
               MOVE.L #INT,($604).L         ;starting address of interrupt routine
                                            ;(assuming VBR = $400)
               ORI.W #$0005,(TPUMCR).L      ;IARB field = 5
               MOVE.W #$0680,(TICR).L       ;interrupt level 6, vector $80
               ANDI.W #$F5FF,SR             ;mask out interrupts at level 5 and below
                                            ;assuming reset state of SR
               MOVE.W #$008E,(PRAM1_0).L    ;psc=0,pac=toggle, capture/match tcr1
               MOVE.W #$0200,(PRAM1_1).L    ;tcr1 offset from REF_ADDR1 for channel 1
               MOVE.W #$00EC,(PRAM1_2).L    ;REF_ADDR1 = tcr1 for channel 1
               MOVE.W #$0004,(HSRR1).L      ;host-init pulse request, ch 1 will go
                                            ;high
               MOVE.W #$0002,(CIER).L       ;enable interrupt for channel 1
MNLOOP         BRA MNLOOP

INT
               ANDI.W #$FFFD,(CISR).L       ;read and clear interrupt
               MOVE.W (PRAM1_0).L,D0
               CMPI.B #$8E,D0               ;see whether ch1 should go low or high
               BNE NXT
               MOVE.W #$0089,(PRAM1_0).L  ;capture/match tcr1,psc=high,pac=low
               MOVE.W #$001A,(PRAM1_2).L    ;REF_ADDR1=ACTUAL_MAT_TIME for channel 1
               MOVE.W #$0004,(HSRR1).L      ;channel 1 will go low
               BRA EN
NXT            MOVE.W #$008E,(PRAM1_0).L    ;psc=low,pac=toggle, capture/match tcr1
               MOVE.W #$0004,(HSRR1).L      ;host-init pulse request, ch 1 will go
                                            ;high
               EN        RTE
```

Continuous Pulse Mode

This example uses OC with the PPWA function to divide the input frequency to the PPWA channel by four. The scaling factor is located in the RATIO field of the OC function. A RATIO of $FF corresponds to a scaling factor of 1. RATIO scales the number at the address pointed to by REF_ADDR2 to determine the OC pulse HIGHTIME (not the period). In this example, RATIO scales the period accumulation, indicated by PPWA_LW.

The PPWA function repeatedly accumulates 16 input periods and then generates a link to the OC function. The OC function scales the accumulated period and then generates the scaled output waveform. The PPWA function is on channel 0, and the OC function is on channel 3.

A function generator (or other type of signal generator) must be physically connected to channel 0. The frequency of the input signal must be smaller than $FFFF (65,535) TCR1 counts, since PPWA is initialized in the 16-bit mode. The output pulse train can be observed on channel 3. No other physical connections are necessary.

```
               ORG    $400
               MOVE.W #$E00F,(CFSR3).L      ;ppwa 0, oc 3
               MOVE.W #$0001,(HSQR1).L      ;16 bit mode, links, pulse accum.
               MOVE.W #$00FF,(CPR1).L       ;high priority
               MOVE.W #$00C0,(TPUMCR).L     ;tcr1 as fast as possible

*** PPWA initialization

               MOVE.W #$310B,(PRAM0_0).L    ;pulse accumulate, link to ch 3
               MOVE.W #$1000,(PRAM0_1).L    ;max count = 16 ($10)
               MOVE.W #$FF00,(PRAM0_4).L    ;service rate = slowest

***OC initialization

               MOVE.W #$008A,(PRAM3_0).L    ;low on match, low at init.
               MOVE.W #$2004,(PRAM3_2).L    ;ref addr1 = lastaccum, ratio=1/4.
                                            ;Here, since PPWA is a period accum. and,
                                            ;not a pulse hightime accum., PPWA_LW
                                            ;must be divided by two.  Thus,
                                            ;RATIO = (OC period/PPWA_LW) x (255/2)
                                            ;Since the desired ratio of the output
                                            ;period to the input period is 4/1, and
                                            ;PPWA_LW represents 16 periods,
                                            ;(OC pd./PPWA_LW) = (4/1) x (1/16) =1/4
                                            ;Thus, RATIO = (1/4) x (255/2) = $20.
               MOVE.W #$0A04,(PRAM3_3).L    ;ref addr2=ppwalw, refaddr3=lastaccum
               MOVE.W #$00C2,(HSRR1).L      ;host service requests
DONE           BRA DONE
```


How to reach us:
USA/EUROPE/Locations Not Listed: Motorola Literature Distribution, P.O. Box 5405, Denver, Colorado 80217. 1-800-441-2447 or 1-303-675-2140. Customer Focus Center, 1-800-521-6274
JAPAN: Motorola Japan Ltd.: SPD, Strategic Planning Office, 141, 4-32-1 Nishi-Gotanda, Shinagawa-Ku, Tokyo, Japan, 03-5487-8488
ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd., Silicon Harbour Centre, 2 Dai King Street, Tai Po Industrial Estate, Tai Po, New Territories, Hong Kong, 852-26629298
Mfax™, Motorola Fax Back System: RMFAX0@email.sps.mot.com; http://sps.motorola.com/mfax/; TOUCHTONE, 1-602-244-6609; US and Canada ONLY, 1-800-774-1848
HOME PAGE: http://motorola.com/sps/

MOTOROLA