UNIVERSIDAD NACIONAL

AUTÓNOMA DE MÉXICO



FACULTAD DE INGENIERIA

Trabajo:

EQUIPO:

SANTIAGO CRUZ CARLOS

ASIGNATURA:

GRUPO:

viernes, 26 de mayo de 2006, Ciudad Universitaria, México, DF

**SALIDA PARA COMPARACIÓN**

Programa un evento que debe ocurrir a un tiempo especifico (cuando el contador libre alcanza un valor especificado) Cada salida de comparación tiene un registro de 16 bits de referencia para comparación y un comparador de 16 bits dedicado. El valor programado en el registro de referencia se compara con el valor del contador libre por cada ciclo de reloj. Cuando el registro de 16 bits de referencia para comparación se iguala con el valor del contador libre, una bandera de salida del estado de comparación se pone en ‘1’ esta bandera se puede usar para iniciar funciones automáticas producto de tales valores de comparación.

**0CI**

**OCF**

I

**OR**

**16 BITS**

**TMSK1**

**TFLG1**

**Del contador libre (TCNT)**

**IRQ**

**TOC-H TOC-L**

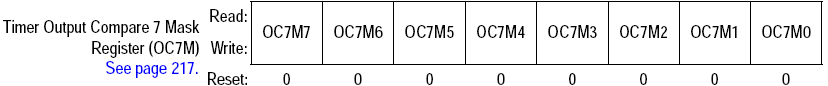
**(CXF)**

**FOC**

**CFORC**

**OC**

**(ptTn)**

****

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
| CFORC | FOC7 | FOC6 | FOC5 | FOC4 | FOC3 | FOC2 | FOC1 | FOC0 | $0081 |
| TIMER  COMPARE  FORCE  REGISTER |  |  |  |  |  |  |  |  |  |
| Al reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
| OC7M | OC7M7 | OC7M6 | OC7M5 | OC7M4 | OC7M3 | OC7M2 | OC7M1 | OC7M0 | $0082 |
| OUTPUT  COMPARE  7 MASK |  |  |  |  |  |  |  |  |  |
| Al reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |

|  |  |  |
| --- | --- | --- |
|  | 0 | portT actúa como un pin controlado por el pin DDR |
| 0C7Mn |  |  |
|  | 1 | Especifica cuales salidas del PORT, serán usadas por una comparación exitosa en OC7 |

Los bits of OC7M corresponden bit a bit con los bits del Puerto T, poniendo un ‘1’ en algún OC7Mn a pesar del estado del bit DDRTn cuando el correspondiente bit TIOSn es ‘1’ para ser un output compare. Esto no cambia el estado de los bits del DDRT.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
| OC7D | OC7D7 | OC7D6 | OC7D5 | OC7D4 | OC7D3 | OC7D2 | OC7D1 | OC7D0 |  |
| OUTPUT  COMPARE 7  DATA  REGISTER |  |  |  |  |  |  |  |  |  |
| Al reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |

|  |  |  |
| --- | --- | --- |
|  | 0 | De acuerdo al resultado de la comparación exitosa en 0C7 |
| 0C7Dn |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| TCTL1 | OM7 | OL7 | OM6 | OL6 | OM5 | OL5 | OM4 | OL4 | $0088 |
|  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| TCTL2 | OM3 | OL3 | OM2 | OL2 | OM1 | OL1 | OM0 | OL0 | $0089 |
|  |  |  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **OMn** | **OLn** | **Resultado después de una comparación exitosa** |
| 0 | 0 | Ninguna (temporizador desconectado) |
| 0 | 1 | Cambio de estado en la línea OCn (toggle) |
| 1 | 0 | Línea OCn=0 |
| 1 | 1 | Línea OCn=1 |

**FUNCIÓN DE SALIDA PARA COMPARACIÓN DEL HC12[[1]](#footnote-1)**

* EL hc12 le permite forzar un evento para que ocurra en alguna de las 8 terminales del puerto T.
* Un evento externo es un flanco de subida, flanco de bajada, o en modo toggle.
* Para usar la función salida para comparación:
  1. Habilite el subsistema del timer (ponga en **‘1’** el bit **TEN** del registro **TSCR1**)
  2. Diga al HC12 que usted quiere usar el Bit X del puerto T para salida para comparación.
  3. Diga al HC12 lo que usted quiere hacer sobre el Bit X del puerto T (si es que quiere generar un flanco de bajada, o un flanco de subida o modo toggle[[2]](#footnote-2).)
  4. Diga al HC12 que tiempo que quiere que ocurra el evento.
  5. Diga al HC12 si usted quiere que se genere una interrupción cuando el evento sea forzado a ocurrir.

Hay algunas características mas complicadas para el subsistema de salida para comparación las cuales son activadas usando los registros **CFORC, OC7M, OC7D, y TTOV.**

* Escribiendo un ‘1’ en el correspondiente bit del registro CFORC forza a ocurrir un evento de salida de comparación, lo mismo como si una comparación satisfactoria sucediera…

**USANDO LA SALIDA PARA COMPARACIÓN DEL HC12**

**En el programa principal:**

1. Encienda el subsistema timer (registro TSCR1)
2. Ponga el preescalador (registro TSCR2)
3. Configure PTx como OC (output compare) (registro TIOS)
4. Ponga la acción (registros TCTL 1-2, bits OMx OLx)
5. Borrar la bandera (registro TFLG1)
6. Habilite interrupción (registro TIE)

**Rutina de interrupción**

* Configure el tiempo de la siguiente acción a ocurrir (escriba en el registro TCx)
* Para eventos periódicos sume el tiempo al registro TCx
* Borre la bandera (registro TFLG1)

**PROGRAMA**

Desarrollar un programa que genere una frecuencia de 1 [kHz] mediante salida para comparación por el pin 2 del puerto T con un ciclo de trabajo del 50%.

**C2I**

**OC2F**

**OR**

**16 BITS**

**TMSK1**

**TFLG1**

**Del contador libre (TCNT)**

**IRQ**

**TC2-H Tc2-L**

**(CXF)**

**FOC2**

**CFORC**

**OC2**

**(ptTn)**

Especificaciones: f=1[KHz] ciclo de trabajo 50%

f=1 [KHz] 🡪 T= 1[ms]; T/2=0.5 [ms]

ECLK=8[MHz], si elegimos preescalar : 4 ECLK:4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **pr2** | **PR1** | **PR0** | F | Cuenta | TOF | PREESCALAR |
| 0 | 1 | 0 | 2 [MHz] | 0.500 [µs] | 37.77 [ms] | 4 |

Por lo tanto: ECLK=2[MHz] 🡪 Una cuenta cada 0.5 [µs]

Por lo tanto el número de cuentas: 1000 🡪 3E8 ($400)

INICIO

Habilitar timer

Deshabilitar interrupciones

Programa preescalar

Programar puerto 2 como salida para comparación OC2

Programar OC2 en modo toggle

Iniciar con salida alta

Cargar el valor actual del TCNT

Suma el ancho de pulso

Almacena resultado en TC2H

Limpia bandera salida para comparación OC2F

OCF2=1?

NO

SI

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| PORTT |  | EQU |  | $00AE |  |  |
| TSCR |  | EQU |  | $0086 |  |  |
| TCTL4 |  | EQU |  | $008B |  |  |
| TMSK1 |  | EQU |  | $008C |  | ;Timer Interrupt Mask 1 Register (TMSK1) |
| TMSK2 |  | EQU |  | $008D |  |  |
| TIOS |  | EQU |  | $0080 |  | ; Timer Input Capture/Output Compare  Select Register (TIOS) |
| TCTL2 |  | EQU |  | $0089 |  |  |
| TCNT |  | EQU |  | $0084 |  |  |
| TC2H |  | EQU |  | $0094 |  |  |
| TFLG1 |  | EQU |  | $008E |  |  |
| ANCHOP |  | EQU |  | $0400 |  |  |
|  |  |  |  |  |  |  |
|  |  | LDX |  | #$0000 |  | ; |
|  |  | MOVB |  | #$80,TSCR |  | ; habilita timer, pone en ‘1’, bit7 registro TSCR, |
|  |  | CLR |  | TMSK1 |  | ; poner en ceros TMSK1 |
|  |  | MOVB |  | #$02,TMSK2 |  | ; mover byte 02, a registro TMSK2, actúa como sal p/comp. |
|  |  | MOVB |  | #$04,TIOS |  | ; mover byte 04, a registro TIOS |
|  |  | MOVB |  | #$10,TCTL |  | ; |
|  |  | BSET |  | PORTT,X,$04 |  | ; |
| **BUCLE** |  | LDD |  | TCNT,X |  |  |
|  |  | ADDD |  | #ANCHOP |  |  |
|  |  | STD |  | TC2H,X |  |  |
|  |  | BSET |  | TFLG1,X,$04 |  |  |
| **WTFLG** |  | BRCLR |  | TFLG1,X,$04,**WTFLG** |  |  |
|  |  | BRA |  | **BUCLE** |  |  |

**INTERRUPCIÓN POR SOBREFLUJO (por tiempo programable)**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
| TMSK2 | TOI | - | - | - | - | PR2 | PR1 | PR0 | $008D |
|  |  |  |  |  |  |  |  |  |  |
| Al reset |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **pr2** | **PR1** | **PR0** | F | Cuenta | TOF | PREESCALAR |
| 1 | 0 | 1 | 0.25 [MHz] | 4 [µs] | 262.14 [ms] | 32 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
| TFLG2 | TOF |  |  |  |  |  |  |  | $008F |
|  |  |  |  |  |  |  |  |  |  |
| Al reset |  |  |  |  |  |  |  |  |  |

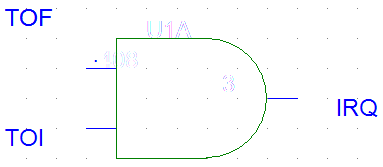
**(TOF & TOI)** 🡪 **interrupción IRQ**

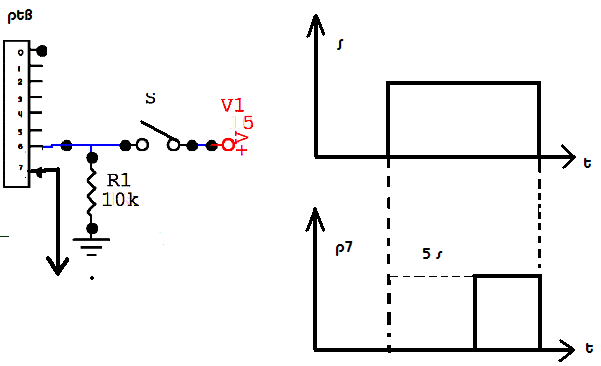
**0.26414[s]\*X=5[s]**

**X=5/0.26414**

**X=18.92910≈1910**

**Por lo tanto X=1910=1316**

****

****

Puerto B pin PB6 (resistencia de pulldown, interruptor de contacto sostenido)

Salida en pin PB7

INICIO

Programar puerto B

Programar contador para TOF’s

Programar tiempo para TOF’s

Leer estado de interruptor ‘s’

S=1?

NO

TOF=1?

SI

Incrementar contador TOF’s

Reset de TOF, TOF=1

NO

SI

S=1?

Contador de TOF=12?

SI

NO

Habilita interrupción TOI

SI

TOF=1?

Reset de TOF, TOF=1

Restablece contador de TOF’s

no

si

NO

Deshabilita interrupción TOI

pinPB7=1

Interruptor s=1?

SI

pinPB7=0

No

RTI

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| PORTB |  | EQU |  | $0001 |  |  |
| DDRB |  | EQU |  | $0003 |  |  |
| TMSK2 |  | EQU |  | $008D |  |  |
| TFLG2 |  | EQU |  | $008F |  |  |
|  |  |  |  |  |  |  |
|  |  | LDX |  | #$0000 |  |  |
|  |  | MOVB |  | #$80,DDRB |  | ;programar puerto B, ‘1’ salida, ‘0’ entrada |
|  |  | CLRB |  |  |  | ;programa contador de TOF, se usa el acum. B |
|  |  | MOVB |  | #$05,TMSK2 |  | ;programar tiempos TOF cada 262.14 [ms] (÷32) |
|  |  |  |  |  |  |  |
| **LEER** |  | BRCLR |  | PORTB,X,$40,**LEER** |  | ;Lee estado del interruptor ‘s’, bit 0**1**00 0000 |
| **ESPERA** |  | BRCLR |  | TFLG2,X,$80,**ESPERA** |  | ;pregunta si TOF=1 |
|  |  | INCB |  |  |  | ;incrementa contador de TOF’s, usa acum.. B |
|  |  | BSET |  | TFLG2,X,$80 |  | ;reset de TOF, (TOF=1) |
|  |  | BRCLR |  | PORTB,X,$40,**RESTABLECER** |  | ;verifica si ‘s’ esta cerrado. |
|  |  | CMPB |  | #$12 |  |  |
|  |  | BNE |  | **ESPERA** |  | ; |
|  |  | BSET |  | TMSK2,X,$80 |  | ;habilita interrupción (TOI=1, TMSK2) |
| **ESP** |  | BRCLR |  | TFLG2,X,$80,**ESP** |  | ;pregunta si TOF=1 |
| **RESTABLECER** |  | BSET |  | TFLG2,X,$80 |  | ;reset de TOF, pone ‘1’ bit TOF |
|  |  | CLRB |  |  |  | ;restablece contador de TOF’s |
|  |  | BRA |  | LEER |  | ; leer estado del interruptor ‘s’ |
|  |  |  |  |  |  |  |
| **INTERRUPCION** |  | BCLR |  | TMSK2,X,80 |  | ;pone ‘0’ en bit TOI |
|  |  | BSET |  | PORTB,X,$80 |  | ;pone ‘1’ en bit ptB7 |
| **SALTA** |  | BRSET |  | PORTB,X,$40,**SALTA** |  | ;lee ‘s’ |
|  |  | BCLR |  | PORTB,X,$80 |  | ;pone ‘0’ en bit ptB7 |
|  |  | RTI |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  | **ORG** |  | $F7DE |  | ;RACOM |
|  |  | DW |  | **INTERRUPCION** |  | ;RACOM |
|  |  |  |  |  |  |  |
|  |  | **ORG** |  | $0B1E |  | ;amigo12 |
|  |  | DW |  | **INTERRUPCION** |  | ;amigo12 |

1. Tomado de apuntes de internet [↑](#footnote-ref-1)
2. Hallar traducción. [↑](#footnote-ref-2)