

Nom i Cognoms: \_\_\_\_\_

1) Configurem la unitat CCP1 en mode Compare amb els valors (**Fosc=8MHz**): CCP1CON=0x02, CCPTMRS0=0x02, CCPR1=0x0336, T5CON=0x73, TMR5GE=0 i CCP1TRIS=0.

Quína freqüència del senyal esperem tenir a la sortida del pin CCP1? (1,5punts)

CCP1CON=2 vol dir que estem en mode Toggle. CCPTMRS0=0x02 vol dir que triem el Timer 5 per la unitat de CCP, T5CON=0x73 vol dir que hem triat Fosc amb un PREscaler de 8, per tant arribarà un tick de rellotge cada 1us. El TMR5GE implica que no utilitzem el circuit de Gate i el TRIS està ben configurat com a sortida.

Si CCPR1=0x0336, que en base 10 és 822, vol dir que cada 822us farem un Toggle de la sortida (suposem que es resetejarà el TIMER) i per tant el senyal generat estarà 822us a 0 i 822us a 1, donant un període de 1644us i una freqüència (invers) de 608,27 Hz.

2) Volem tenir una interrupció d'alta prioritat del Timer0 cada 25ms i comptar les interrupcions que van arribant. Configura tot el necessari si tenim un **Fosc=20MHz**. (2,5punts)

*volatile int* Count\_INT;

```
void main(void)
{
    Count_INT=0;

    TMR0IF=0; TMR0IP=1; TMR0IE=1;

    T0CON=0x80; //PRE=2

    TMR0=65536-62500;

    IPEN=1; GIEH=1; //activem

    ...
}
```

```
void interrupt RSI_High(void)
{
    if (TMR0IF&&TMR0IE)
    {
        TMR0=65536-62500;

        TMR0IF=0;

        Count_INT++;

    }
}
```

Al timer 0 arriba Fosc/4 això són 50ns\*4=200ns. Per comptar 25ms cal comptar 25ms/200ns=125000 ticks de rellotge. Com que això no cap en 16 bits posem el PREscaler a 2 i comptem 62500 ticks.

3) Si s'activessin en el mateix moment 4 flags d'interrupció (per ex. INT1IF, TMR1IF, ADIF, CCP3IF), i estan les 4 habilitades i configurades com a baixa prioritat, què determinarà en quin ordre les atendrem? (1punt)

Donat que quan es comprovin els interrupt flags, estaran 4 activats, i es fa la OR dels 4 per determinar que cal llançar la rutina d'interrupció de l'adreça 0x18, el que determina l'ordre d'atenció serà el codi fet per l'usuari a la rutina. Si comencem amb un:

If (INT1IF && INT1IE) la INT1... serà la primera a atendre's.

REGISTER 14-3: CCPTRM0: PWM TIMER SELECTION CONTROL REGISTER 0				
RW-0	RW-0	U-0	RW-0	RW-0
bit 7	bit 6	bit 5	bit 4	bit 3
bit 2	bit 1	bit 0	bit 0	bit 0

REGISTER 14-1: CCPXCON: STANDARD CCPX CONTROL REGISTER				
U-0	U-0	RW-0	RW-0	RW-0
bit 7	bit 6	bit 5	bit 4	bit 3
bit 2	bit 1	bit 0	bit 0	bit 0

bit 1-0  
**CTISEL<1:0>**: CCP1 Timer Selection bits  
 00 = CCP1 – Capture/Compare modes use Timer1, PWM modes use Timer2  
 01 = CCP1 – Capture/Compare modes use Timer3, PWM modes use Timer4  
 10 = CCP1 – Capture/Compare modes use Timer5, PWM modes use Timer6  
 11 = Reserved

bit 7-6  
**DC8<1:0>**: PWM Duty Cycle least significant bits  
 Capture mode:  
 Unused  
 Compare mode:  
 Unused  
 PWM mode:  
 These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPXCL.  
**CCPXM<3:0>**: CCPX Mode Selected bits  
 0000 = Capture/Compare/PWM off (resets the module)  
 0001 = Reserved  
 0010 = Compare mode: toggle output on match  
 0011 = Reserved  
 0100 = Capture mode: every falling edge  
 0101 = Capture mode: every rising edge  
 0110 = Capture mode: every 4th rising edge  
 0111 = Capture mode: every 16th rising edge  
 1000 = Compare mode: set output on compare match (CCPx pin is set, CCPXIF is set)  
 1001 = Compare mode: clear output on compare match (CCPx pin is cleared, CCPXIF is set)  
 1010 = Compare mode: generate software interrupt on compare match (CCPx pin is unaffected, CCPXIF is set)  
 1011 = Compare mode: Special Event Trigger (CCPx pin is unaffected, CCPXIF is set)  
 11xx = PWM mode  
 ADON is set, starting A/D conversion if A/D module is enabled<sup>(1)</sup>

7	6	5	4	3	2	1	0
TMROON	TO8BIT	TOCS	TOSE	PSA	TOPS2	TOPS1	TOPSO
1	1	1	1	1	1	1	1

TMROON: *Timer0 on/off control bit*  
 0 = stops Timer0  
 1 = Enables Timer0

TO8BIT: *Timer0 8-bit/16-bit control bit*  
 0 = Timer0 is configured as a 16-bit timer  
 1 = Timer0 is configured as an 8-bit timer

TOCS: *Timer0 clock source select*  
 0 = Instruction cycle clock  
 1 = Transition on TOCKI pin

TOSE: *Timer0 source edge select bit*  
 0 = Increment on falling edge transition on TOCKI pin  
 1 = Increment on rising edge transition on TOCKI pin

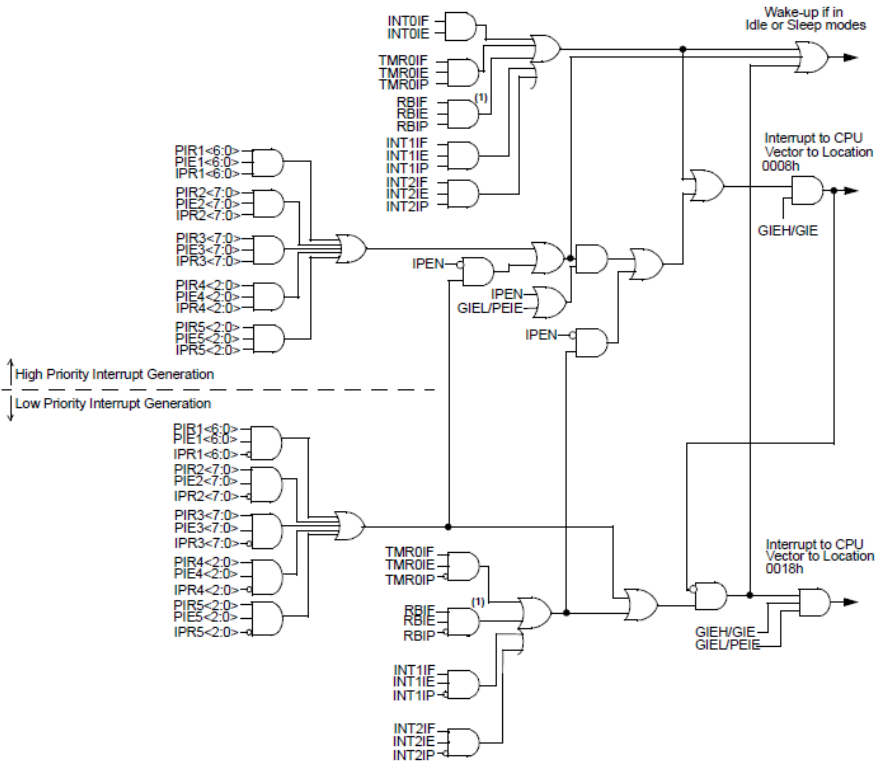
PSA: *Timer0 prescaler assignment bit*  
 0 = Timer0 prescaler is assigned. Timer0 clock input comes from prescaler  
 1 = Timer0 prescaler is not assigned. Timer0 clock input bypasses prescaler

TOPS2:TOPSO: *Timer0 prescaler select bits*  
 000 = 1:2 prescaler value  
 001 = 1:4 prescaler value  
 010 = 1:8 prescaler value  
 011 = 1:16 prescaler value  
 100 = 1:32 prescaler value  
 101 = 1:64 prescaler value  
 110 = 1:128 prescaler value  
 111 = 1:256 prescaler value

Figure 8.2 TOCON register (reprint with permission of Microchip)

TABLE 9-1: REGISTERS ASSOCIATED WITH INTERRUPTS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
INTCON2	RBPU	INTEDG0	INTEDG1	INTEDG2	—	TMR0IP	—	RBIP
INTCON3	INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	—	—	—	—
IPR1	—	ADIP	RC1IP	TX1IP	SSP1IP	CCP1IP	TMR2IP	TMR1IP
IPR2	OSCFIP	C1IP	C2IP	EEIP	BCL1IP	HLVDIP	TMR3IP	CCP2IP
IPR3	SSP2IP	BCL2IP	RC2IP	TX2IP	CTMUIP	TMR5GIP	TMR3GIP	TMR1GIP
IPR4	—	—	—	—	—	CCP5IP	CCP4IP	CCP3IP
IPR5	—	—	—	—	—	TMR6IP	TMR5IP	TMR4IP
PIE1	—	ADIE	RC1IE	TX1IE	SSP1IE	CCP1IE	TMR2IE	TMR1IE
PIE2	OSCFIE	C1IE	C2IE	EEIE	BCL1IE	HLVDIE	TMR3IE	CCP2IE
PIE3	SSP2IE	BCL2IE	RC2IE	TX2IE	CTMUIE	TMR5GIE	TMR3GIE	TMR1GIE
PIE4	—	—	—	—	—	CCP5IE	CCP4IE	CCP3IE
PIE5	—	—	—	—	—	TMR6IE	TMR5IE	TMR4IE
PIR1	—	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF
PIR2	OSCFIF	C1IF	C2IF	EEIF	BCL1IF	HLVDIF	TMR3IF	CCP2IF
PIR3	SSP2IF	BCL2IF	RC2IF	TX2IF	CTMUIF	TMR5GIF	TMR3GIF	TMR1GIF
PIR4	—	—	—	—	—	CCP5IF	CCP4IF	CCP3IF
PIR5	—	—	—	—	—	TMR6IF	TMR5IF	TMR4IF
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
RCON	IPEN	SBOREN	—	RI	TO	PD	POR	BOR



Nom i Cognoms: \_\_\_\_\_

4) Volem utilitzar la unitat de Capture de dos CCPs per saber l'amplada d'un pols. La CCP1 estarà configurada per trobar el flanc de pujada i la CCP2 per trobar el flanc de baixada. Fosc=8MHz, el timer associat utilitza Fosc com a entrada i PRE=1. Després de l'arribada dels dos flancs tenim que CCPR1=17 i CCPR2=17, què podem saber respecte l'amplada del pols? (2 punts)

Amb Fosc=8MHz, Prescaler=1 i l'entrada del Timer triada a Fosc, vol dir que el nostre Timer s'incrementa cada 125ns (1/Fosc). Si quan han arribat els dos flancs del senyal (ascendent i descendent) els CCPR1 i CCPR2 valen el mateix, en aquest cas 17, tenim dues opcions:

- Els dos flancs han arribat en menys de 125ns, i els dos CCP han captat el mateix valor del timer.
- Hi ha hagut overflows i els dos flancs han arribat en  $125 * N * 65536$  ns, on N pot valdre 0 (cas anterior), 1, 2, 3 ...

5) Se'ns demana, a partir d'un Fosc=12MHz, generar un PWM de freqüència 12,5 KHz. Quines configuracions ens permetran tenir com a mínim 6 bits de resolució? (2 punts)

Si apliquem la fórmula del període del PWM tenim que:

$$1/12,5\text{KHz} = (PR+1) * 4 * 1/12\text{MHz} * PRE \quad \text{D'aquí treiem que:} \quad (PR+1)*PRE = 240$$

Provem amb els PREscalers possibles (1, 4, 16).

Si PRE=1 llavors PR+1=240 i els Duty Cicles possibles són 960, amb una resolució de 9,9 bits.

Si PRE=4 llavors PR+1=60 i els Duty Cicles possibles són 240, amb una resolució de 7,9 bits.

Si PRE=16 llavors PR+1=15 i els Duty Cicles possibles són 60, amb una resolució de 5,9 bits.

Aquesta última configuració **no servirà**.

6) Volem utilitzar el timer 4 com a base de temps, tot generant una interrupció periòdica basada en TMR4IF. Si Fosc=8MHz, quina serà la freqüència mínima amb que ens poden arribar les interrupcions? (1 punt)

Per tenir la freqüència mínima hem de tenir els PREscalers, POSTscalers i comparador al màxim, per consumir el màxim de temps (període màxim).

Tindrem Fosc/4 a l'entrada, amb un PRE=16, PRx=255 (màxim en 8 bits) i POST=16.

La freqüència que resulta és de 30,5 Hz que equival a 0,032768 segons de període.

REGISTER 12-1: TXCON: TIMER1/3/5 CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/0	R/W-0/u
TMRxCS<1:0>	TxCKPS<1:0>	TxSOSCEN	TxSYNC	TxRD16	TMRxON		
bit 7							bit 0

- bit 7-6 **TMRxCS<1:0>**: Timer1/3/5 Clock Source Select bits  
 11 = Reserved. Do not use.  
 10 = Timer1/3/5 clock source is pin or oscillator:  
   If TxSOSCEN = 0:  
     External clock from TxCKI pin (on the rising edge)  
   If TxSOSCEN = 1:  
     Crystal oscillator on SOSC/SOSCO pins  
 01 = Timer1/3/5 clock source is system clock (Fosc)  
 00 = Timer1/3/5 clock source is instruction clock (Fosc/4)
- bit 5-4 **TxCKPS<1:0>**: Timer1/3/5 Input Clock Prescale Select bits  
 11 = 1:8 Prescale value  
 10 = 1:4 Prescale value  
 01 = 1:2 Prescale value  
 00 = 1:1 Prescale value
- bit 3 **TxSOSCEN**: Secondary Oscillator Enable Control bit  
 1 = Dedicated Secondary oscillator circuit enabled  
 0 = Dedicated Secondary oscillator circuit disabled
- bit 2 **TxSYNC**: Timer1/3/5 External Clock Input Synchronization Control bit  
 TMRxCS<1:0> = 1X  
 1 = Do not synchronize external clock input  
 0 = Synchronize external clock input with system clock (Fosc)
- TMRxCS<1:0> = 0X  
 This bit is ignored. Timer1/3/5 uses the internal clock when TMRxCS<1:0> = 1X.
- bit 1 **TxRD16**: 16-Bit Read/Write Mode Enable bit  
 1 = Enables register read/write of Timer1/3/5 in one 16-bit operation  
 0 = Enables register read/write of Timer1/3/5 in two 8-bit operation
- bit 0 **TMRxON**: Timer1/3/5 On bit  
 1 = Enables Timer1/3/5  
 0 = Stops Timer1/3/5  
 Clears Timer1/3/5 Gate flip-flop

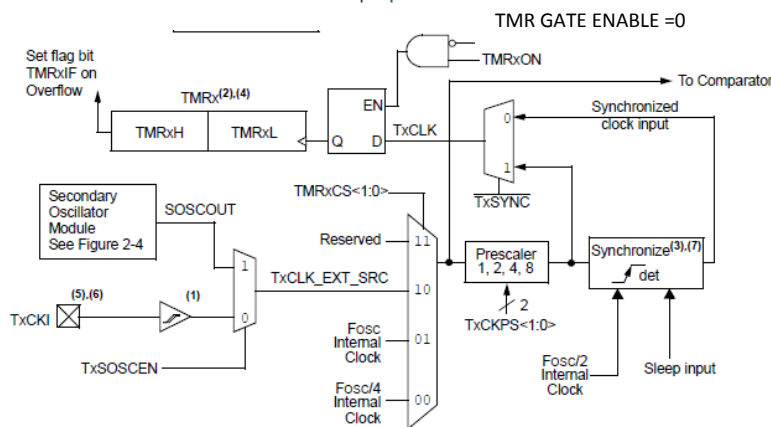
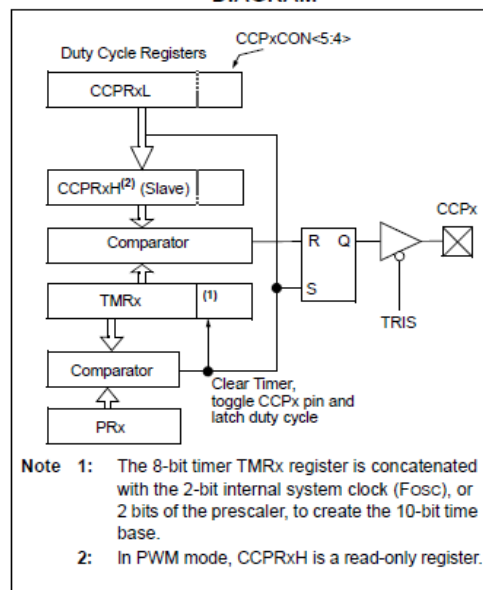


FIGURE 14-4: SIMPLIFIED PWM BLOCK DIAGRAM



$$PWM\ Period = [(PRx) + 1] \cdot 4 \cdot T_{osc} \cdot (TMRx\ Prescale\ Value)$$

**Note 1:**  $T_{osc} = 1/F_{osc}$

$$Pulse\ Width = (CCPRxL:CCPxCON<5:4>) \cdot T_{osc} \cdot (TMRx\ Prescale\ Value)$$

$$Resolution = \frac{\log[4(PRx + 1)]}{\log(2)}\ bits$$

FIGURE 13-1: TIMER2/4/6 BLOCK DIAGRAM

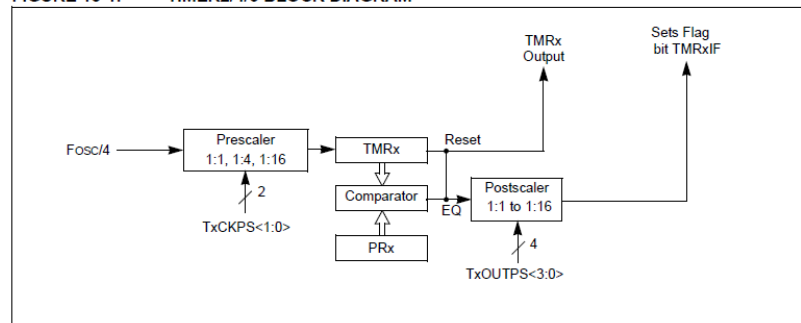


FIGURE 14-1: CAPTURE MODE OPERATION BLOCK DIAGRAM

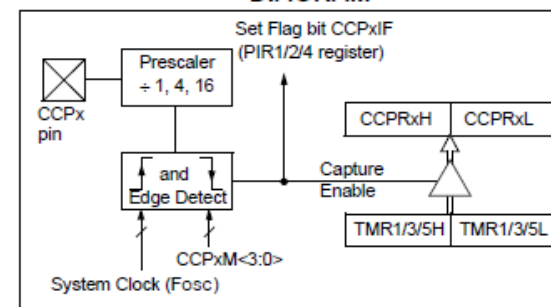


FIGURE 14-2: COMPARE MODE OPERATION BLOCK DIAGRAM

