

# Microcontroladores

## Semana 6

Semestre 2025-2

Por Kalun José Lau Gan

### Preguntas previas

- ¿Cómo funcionan los sensores ultrasónicos?
  - Funcionando con señales ultrasónicas, se envia un pulso, éste rebota y regresa y se mide el tiempo recorrido por dicha señal
- ~~Para los del EL51, el siguiente miércoles cae 1 de mayo. ¿Cómo serán las clases?~~
  - ~~Ese mismo día serán las clases: 9:00AM - 11:00AM la teoría y laboratorio único 1PM - 5PM.~~
- ~~¿La semana 7 es LB2?~~
  - ~~Si, en el horario publicado en el punto anterior.~~
- No pude rendir la PC1. ¿Algo se podrá hacer?
  - Tienes posibilidad de rendir el examen de recuperación en la semana 17, para ello debes de hacer el trámite correspondiente, más información con tu profesor de campus

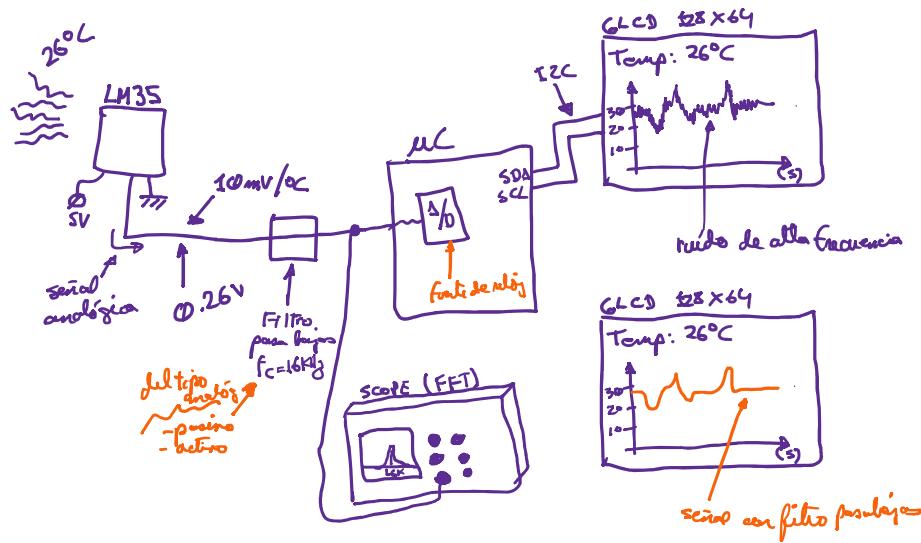
## Agenda:

- Los periféricos temporizadores en el PIC18F57Q43
- El módulo Timer0
  - Modo 8bit
  - Modo 16bit
  - Registros de configuración y operación
- Cálculo de temporización
- Generación de ondas cuadradas periódicas

## Los temporizadores en el PIC18F57Q43

- Según datasheet este microcontrolador tiene siete temporizadores: tres de 8 bits y cuatro de 16 bits.
- Pero en el índice del datasheet solo aparecen tres: Timer0, Timer1 y Timer2
- Timer3 y Timer5 tienen la misma funcionalidad que el Timer1
- Timer4 y Timer6 tienen la misma funcionalidad que el Timer2
- El Timer0 lo utilizamos para aplicaciones genéricas de temporización (su precisión con respecto al tiempo real no es alta)
- El Timer1 se emplea para aplicaciones de reloj en tiempo real, sigue trabajando cuando el CPU entra en modo SLEEP.
- El Timer2 está enfocado a la generación de señal PWM trabajando en conjunto con el módulo CCP.

## Importancia de la temporización en un sistema de adquisición de señal analógica

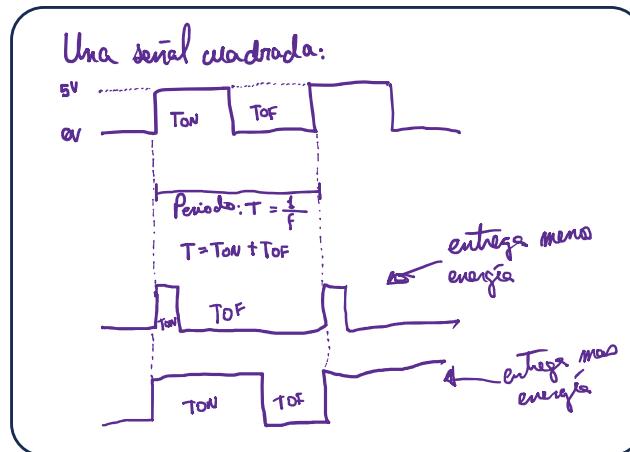


¿Para qué utilizamos temporizadores si con `delay()` se puede hacer también igual?

- La función `delay()`, en nuestro caso el conjunto de instrucciones “nop”, ocasiona que el CPU se encuentre atendiendo dicha rutina de retardo, no dejando hacer otra cosa.
- Lo peor que puedes utilizar en un programa son retardos, éstos hacen que el desempeño global de tu aplicación se vea afectada.
- Una rutina de retardo en software es perder el tiempo en el CPU.

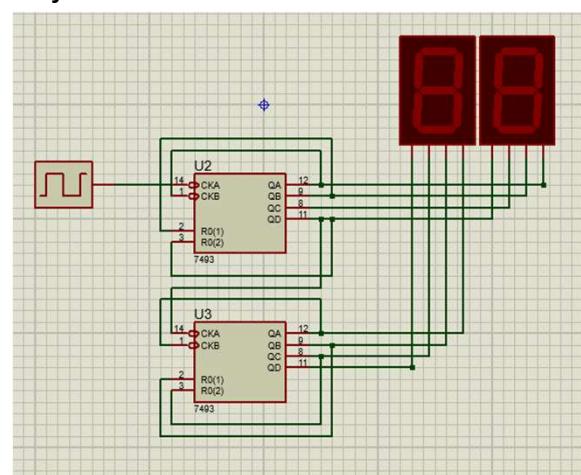
## Señal PWM

- Pulse Width Modulation: Modulación de ancho de pulso
- Se utilizar para obtener una señal de determinada cantidad de energía partiendo de una señal cuadrada.



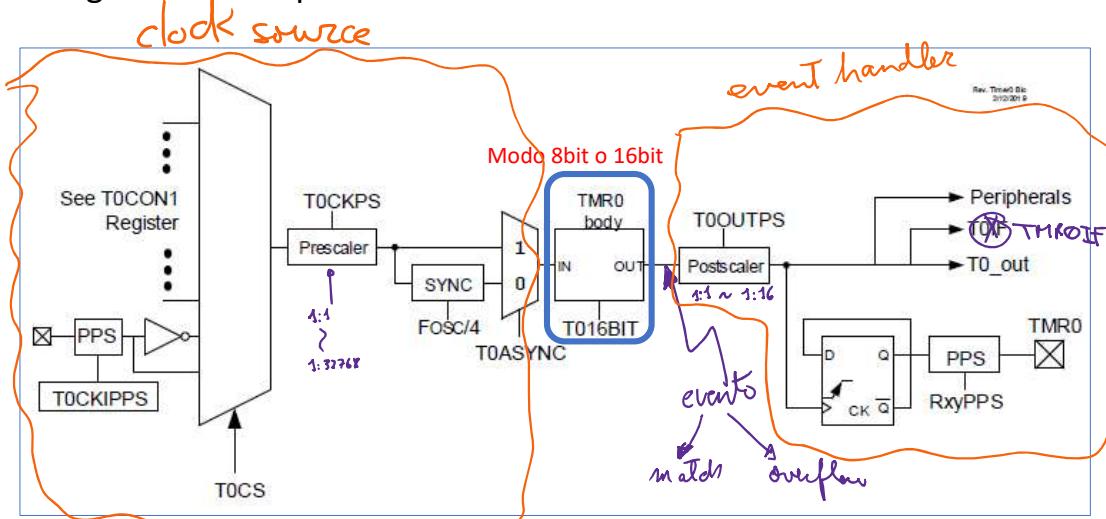
## La previa: contadores digitales

- Sistemas secuenciales, poseen reloj, un registro de cuenta, dependiendo de su función, puede incrementar o puede decrementar la cuenta según la entrada de reloj.



## El módulo Timer 0:

- Diagrama de bloques



## El módulo Timer 0

- (Ref. Item 24 de la hoja técnica del microcontrolador PIC18F57Q43)
- Temporizador de cuenta ascendente
- Modo de trabajo: 8 bits (0-254) ó 16 bits (0-65535)
- Las cuentas del Timer0 se alojan en los siguientes registros:
  - TMROH:TMROL (16 bits)
  - TMROL (8 bits)
- Diversas fuentes de reloj (revisar T0CON1 y TOCS)
- Divisor de frecuencia al reloj de entrada PRESCALER (1:1 – 1:32768)
- POSTSCALER de 1:1 a 1:16 (incrementos de uno en uno)
- Si se trabaja en modo 16 bits y con carga de cuenta inicial, el POSTSCALER debe de ser solo de 1:1.**
- Al activarse TMROIF=1 ó T0IF=1 se debe de bajar manualmente la bandera para que se pueda detectar un nuevo desborde ó evento de match (simplemente haciendo "bcf PIR3, 7"; siendo el bit 7 el TMROIF).
- En un evento de match o de desborde puede emitir interrupción al CPU si TMROIE=1, revisar capítulo 11 de la hoja técnica.

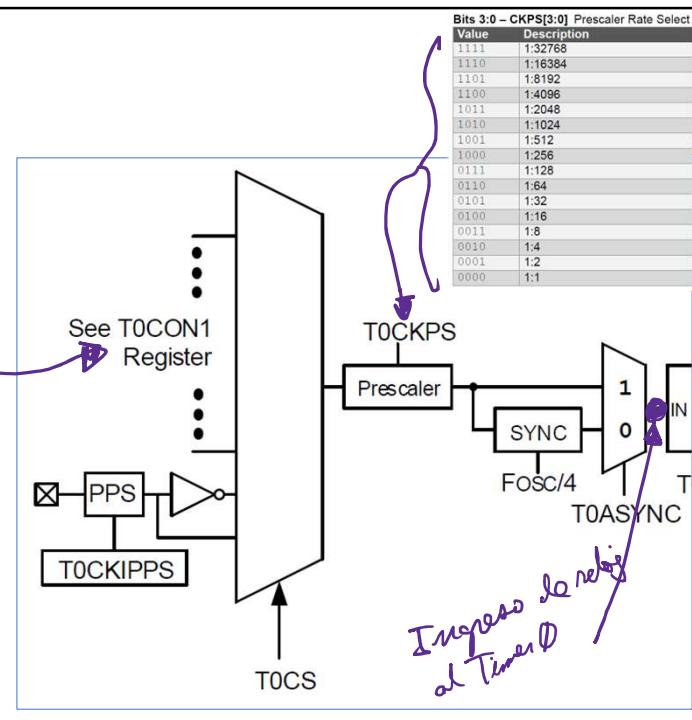
# El módulo Timer0

- Fuente de reloj
- Hay que revisar los bits T0CS del registro T0CON1

Bits 7:5 – CS[2:0] Timer0 Clock Source Select

Value	Description
111	CLC1_OUT
110	SOSC
101	MFINTOSC (500 kHz)
100	LFINTOSC
011	HFINTOSC
010	Fosc/4
001	Pin selected by TOCKIPPS (Inverted)
000	Pin selected by TOCKIPPS (Non-inverted)

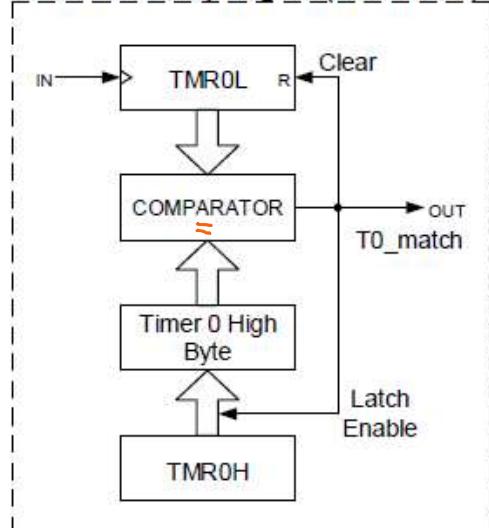
- El PRESCALER es el divisor de frecuencia, va de 1:1 hasta 1:32768
- Revisar cap 21 (PPS) de la hoja técnica para el TOCKIPPS



## El módulo Timer 0:

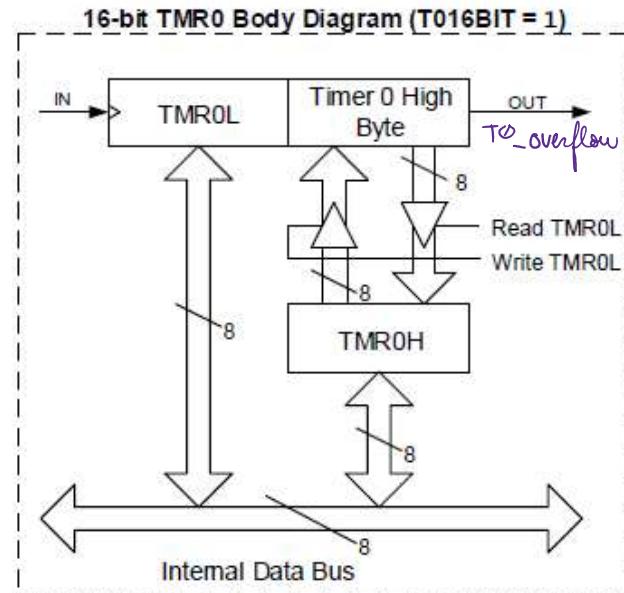
- Modo de trabajo en 8 bits:
  - Mejorado con respecto al Timer0 presente en el PIC18F4550 ó PIC18F45K50
  - El TMR0L se usa como el registro de cuenta
  - El TMR0H se utiliza como valor de comparación
  - Cuenta máxima es cuando TMR0H es 255, haciendo que se mande a cero la cuenta actual, quiere decir que tendrás un rango de 0-254
  - Cuando TMR0L es igual a TMR0H se produce un evento de "match" (T0\_match) el cual limpia la cuenta y actualiza el valor de comparación.
  - Nunca se desborda

8-bit TMR0 Body Diagram (T016BIT = 0)



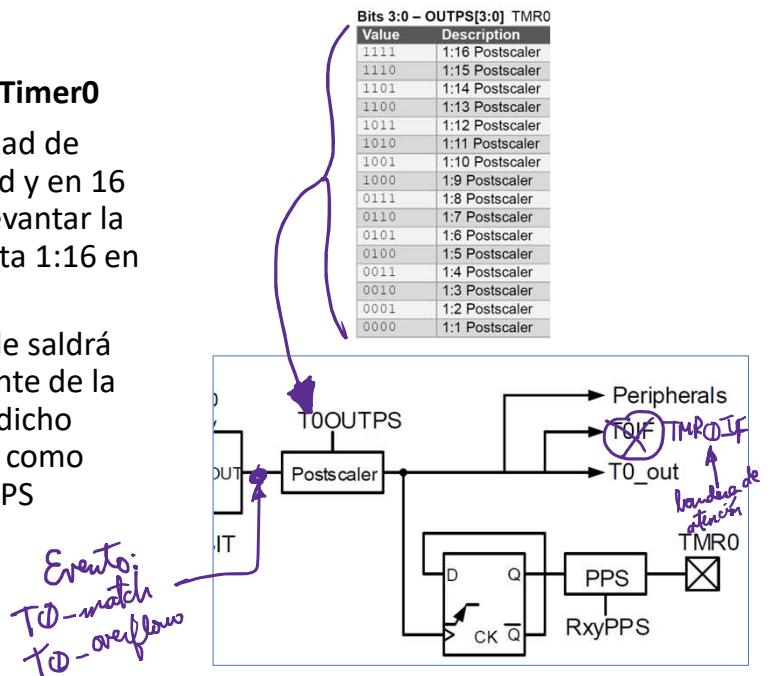
## El módulo Timer 0:

- Modo de trabajo en 16 bits:
  - Tener en consideración el procedimiento estricto sobre el proceso de carga de un valor en la cuenta en modo 16 bits: Primero cargar en TMROH y luego en TMROL.
  - El evento de desborde ( $T0\_overflow$ ) se produce cuando la cuenta esta en el valor mas alto (65535) y se recibe un pulso de reloj, ocasionando que la cuenta pase a 0 y levantándose la bandera de desborde ( $TMR0IF=1$  ó  $T0IF=1$ ) siempre y cuando POSTSCALER 1:1



## El Timer0

- **El gestionador de eventos del Timer0**
- El postscaler va a contar cantidad de eventos (en 8 bits es la igualdad y en 16 bits es el desborde) antes de levantar la bandera  $TMR0IF$ . Va de 1:1 hasta 1:16 en incrementos de uno en uno.
- Se tiene una salida  $TMR0$  donde saldrá una señal basculante proveniente de la salida del postscaler, para ello dicho pueto debe de de configurarse como salida digital y también el RxyPPS



## El Timer0

- Salida de señal del Timer0 por un pin de E/S
- Se tiene que revisar RxyPPS para saber por cuál pin podemos sacar la señal (capítulo 21 de la hoja técnica).
- Solo se puede sacar la señal por RCx o por RFx.

Table 21-2. PPS Output Selection Table

RxyPPS	Output Source	Available Output Ports												
		28-Pin Devices		40-Pin Devices		48-Pin Devices								
0x45	ADGRDB	A	—	C	A	—	C	—	—	A	—	—	—	F
0x44	ADGRDA	A	—	C	A	—	C	—	—	A	—	—	—	F
0x43	DSM1	A	—	C	A	—	D	—	—	A	—	—	D	—
0x42	CLKR	—	B	C	—	B	C	—	—	B	—	—	E	—
0x41	NCO3	—	B	C	—	B	—	—	E	—	B	—	E	—
0x40	NCO2	—	B	C	—	B	—	D	—	—	B	—	D	—
0x3F	NCO1	A	—	C	A	—	D	—	A	—	D	—	—	—
0x3E - 0x3A	Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—
0x39	TMR0	—	B	C	—	B	C	—	—	C	—	—	F	—
0x38	I2C1 SDA <sup>(1)</sup>	—	B	C	—	B	C	—	—	B	C	—	—	—

## El módulo Timer 0:

- Temporización máxima si HFINTOSC = 4MHz
  - Empleando Opción FOSC/4 = 1MHz (1μs) como fuente de reloj al TMR0
  - Prescaler = 1:32768 (32768 μs por cuenta)
  - Modo = 16bits (65536 cuentas)
  - En el evento de desborde: 2147483648 μs (35 minutos)
  - Postscaler = 1:16
  - Nos sale al final 34,359,738,368 μs de temporización máxima! (aprox 9.5 horas)

## El módulo Timer 0:

- Temporización máxima si LFINTOSC = 31KHz
  - Empleando Opción LFINTOSC = 31KHz (32.258ms) como fuente de reloj al TMRO
  - Prescaler = 1:32768 (1.057 s por cuenta)
  - Modo = 16bits (65536 cuentas)
  - En el evento de desborde: 69273.666s (~1154 minutos ó aprox 19 horas)
  - Postscaler = 1:16
  - Nos sale al final 1108378.656 s de temporización máxima! (aprox 307 horas!)

## El módulo Timer 0

- Registros implicados en la operación del Timer0:

- Registros de cuenta
  - TMROH:TMROL (16bits)
  - TMROH es valor de comparación y TMROL es el registro de cuenta (8bits)
- Registros de configuración **T0CON0** y **T0CON1**
- Registros PIE3 (habilitadores), PIR3 (banderas) y IPR3 (prioridades) ubicando en la VIC (vectored interrupt controller cap 11)

} bank 3

Address	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x0318	TMROL	7:0	TMROL[7:0]							
0x0319	TMROH	7:0	TMROH[7:0]							
0x031A	T0CON0	7:0	EN		OUT	MD16		OUTPS[3:0]		
0x031B	T0CON1	7:0		CS[2:0]		ASYNC		CKPS[3:0]		
0x04A1	PIE3	7:0	TMROIE	CCP1IE	TMR1GIE	TMR1IE	TMR2IE	SPI1IE	SPI1TXIE	SPI1RXIE
0x04B1	PIR3	7:0	TMROIF	CCP1IF	TMR1GIF	TMR1IF	TMR2IF	SPI1IF	SPI1TXIF	SPI1RXIF
0x0365	IPR3	7:0	TMROIP	CCP1IP	TMR1GIP	TMR1IP	TMR2IP	SPI1IP	SPI1TXIP	SPI1RXIP

## El módulo Timer 0

- Registro T0CON0:
  - Habilitador del módulo
  - Señal OUT
  - Modo (8 ó 16 bits)
  - Postscaler

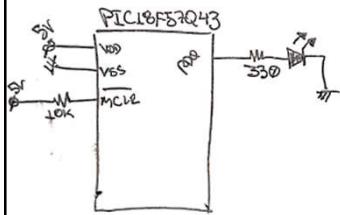
Timer0 Control Register 0								
Bit	7	6	5	4	3	2	1	0
Access	R/W		R	R/W	R/W	R/W	R/W	R/W
Reset	0		0	0	0	0	0	0
<b>Bit 7 – EN</b> TMR0 Enable								
<b>Value</b>	<b>Description</b>							
1	The module is enabled and operating							
0	The module is disabled							
<b>Bit 5 – OUT</b> TMR0 Output								
<b>Bit 4 – MD16</b> 16-Bit Timer Operation Select								
<b>Value</b>	<b>Description</b>							
1	TMR0 is a 16-bit timer							
0	TMR0 is an 8-bit timer							
<b>Bits 3:0 – OUTPS[3:0]</b> TMR0 Output Postscaler (Divider) Select								
<b>Value</b>	<b>Description</b>							
1111	1:16 Postscaler							
1110	1:15 Postscaler							
1101	1:14 Postscaler							
1100	1:13 Postscaler							
1011	1:12 Postscaler							
1010	1:11 Postscaler							
1001	1:10 Postscaler							
1000	1:9 Postscaler							
0111	1:8 Postscaler							
0110	1:7 Postscaler							
0101	1:6 Postscaler							
0100	1:5 Postscaler							
0011	1:4 Postscaler							
0010	1:3 Postscaler							
0001	1:2 Postscaler							
0000	1:1 Postscaler							

Timer0 Control Register 1								
Bit	7	6	5	4	3	2	1	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0
<b>Bits 7:5 – CS[2:0]</b> Timer0 Clock Source Select								
<b>Value</b>	<b>Description</b>							
111	CLC1_OUT							
110	SOSC							
101	MFINTOSC (500 kHz)							
100	LFINTOSC							
011	HFINTOSC							
010	Fosc/4							
001	Pin selected by TUCKIPPS (Inverted)							
000	Pin selected by TOCKIPPS (Non-inverted)							
<b>Bit 4 – ASYNC</b> TMR0 Input Asynchronization Enable								
<b>Value</b>	<b>Description</b>							
1	The input to the TMR0 counter is not synchronized to system clocks							
0	The input to the TMR0 counter is synchronized to Fosc/4							
<b>Bits 3:0 – CKPS[3:0]</b> Prescaler Rate Select								
<b>Value</b>	<b>Description</b>							
1111	1:32768							
1110	1:16384							
1101	1:8192							
1100	1:4096							
1011	1:2048							
1010	1:1024							
1001	1:512							
1000	1:256							
0111	1:128							
0110	1:64							
0101	1:32							
0100	1:16							
0011	1:8							
0010	1:4							
0001	1:2							
0000	1:1							

## Ejercicios:

- Prender y apagar un LED en RD0 con un periodo de 1000ms aproximadamente (TON/TOF = 500ms) empleando el Timer0 como fuente de temporización.

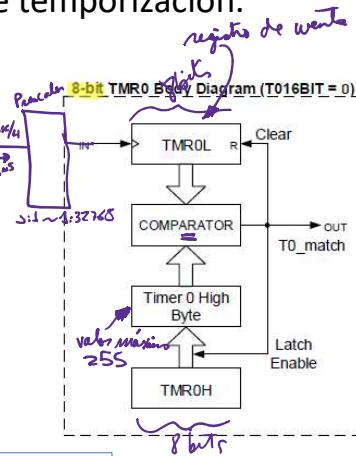
Circuito de la aplicación:



$$T = \frac{1}{f}$$

Si uso modo 8 bits y fuente de reloj para el Timer0 FOSC/4 sabiendo que FOSC=4MHz

- Si PSC es 1:1 cada cuenta es 1μs, voy a poder temporizar solo 255μs (evento match)
- Si PSC es 1:32768 entonces cada cuenta es 32768μs, la temporización máxima será 8.3 segundos aprox!
- Si PSC es 1:8192, cada cuenta es ahora 8192μs, la temporización máxima será 2 segundos aprox.
- Si PSC es 1:2048, quiere decir que cada cuenta es de 2048μs, la temporización máxima será de aprox 500 ms (522.24 ms)!
- Si valor de comparación lo reduzco a 244, obtendré 499.71 ms, mucho mas cercano al valor solicitado)

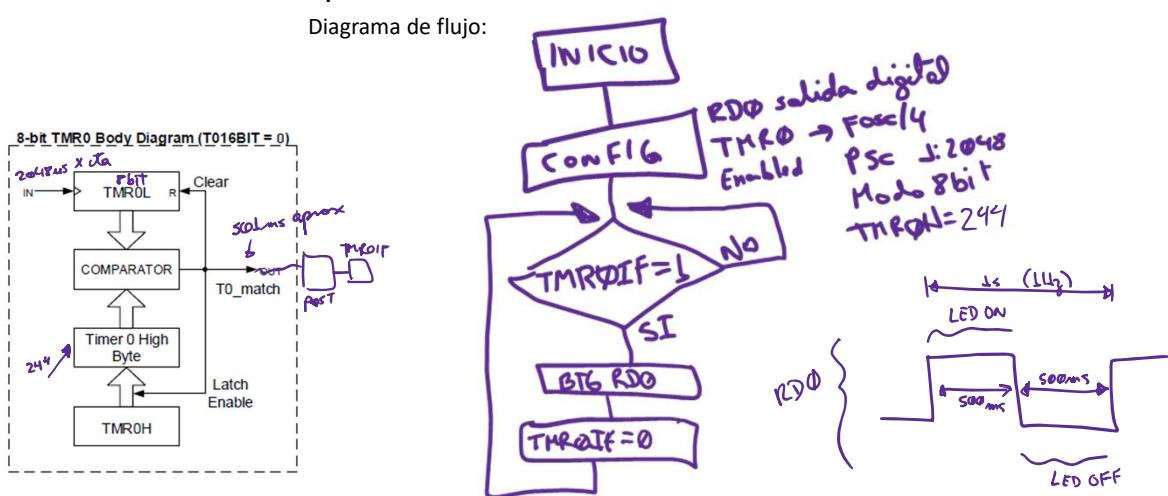


Lo que tenemos en configurar en el Timer0:  
TOCON0: 80H (Timer0 enabled, POSTSCALER 1:1)  
TOCON1: 4BH (FOSC/4, PRESCALER 1:2048)  
TMROH: 244

## Ejercicios:

- Prender y apagar un LED en RD0 con un periodo de 1000ms aproximadamente (TON/TOF = 500ms) empleando el Timer0 como fuente de temporización.

Diagrama de flujo:



## Ejercicios:

- Prender y apagar un LED en RD0 con un periodo de 500ms aproximadamente empleando el Timer0 como fuente de temporización.

Código en XC8 PIC Assembler:

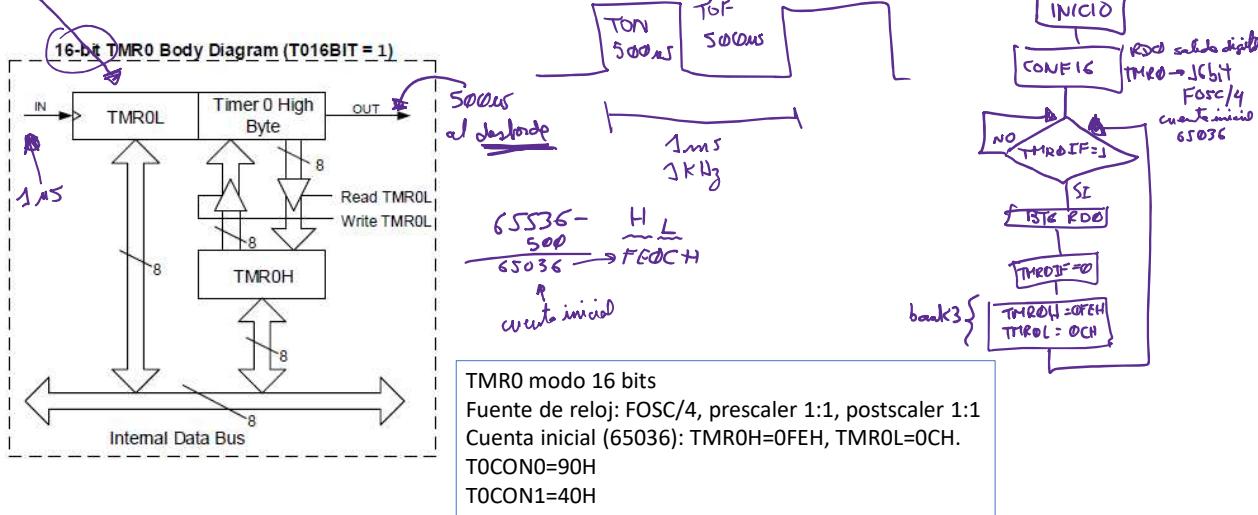
```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSECT upcino, class=CODE, reloc=2, abs
5  upcino:
6      ORG 000000H
7      bra configuro
8
9      ORG 000100H
10 configuro:
11     movlw 0H
12     movlw 60H
13     movwf OSCCON1, 1
14     movlw 02H
15     movwf OSCFRQ, 1
16     movlw 40H
17     movwf OSCEN, 1
18     movlb 3H
19     movlw 80H
20     movwf TOCON0, 1 ;TMR0 enabled, 8bit mode, postscaler 1:1
21     movlw 4BH
22     movwf TOCON1, 1 ;FOSC/4, async, prescaler 1:2048
23     movlw 245
24     movwf TMR0H ;Valor de comparación 245
25     movlb 4H
26     bcf TRISD, 0, 1 ;RD0 como salida
27     bcf ANSEL0, 0, 1 ;RD0 como digital
28
29 inicio:
30     btfss PIR3, 7, 1 ;Pregunto si hubo match en TMR0, TMROIF=1
31     bra inicio
32     btg LATD, 0, 1 ;Complementamos RD0
33     bcf PIR3, 7, 1 ;Bajamos la bandera TMRO IF
34     bra inicio
35
36 end upcino

```

## Ejercicios:

- Justo arriba*
- Como generamos una señal cuadrada de 1KHz 50%DC ahora usando el modo 16 bits



## Ejercicios:

- Cómo generamos una señal cuadrada de 1KHz 50%DC ahora usando el modo 16 bits

Código en XC8 PIC Assembler:

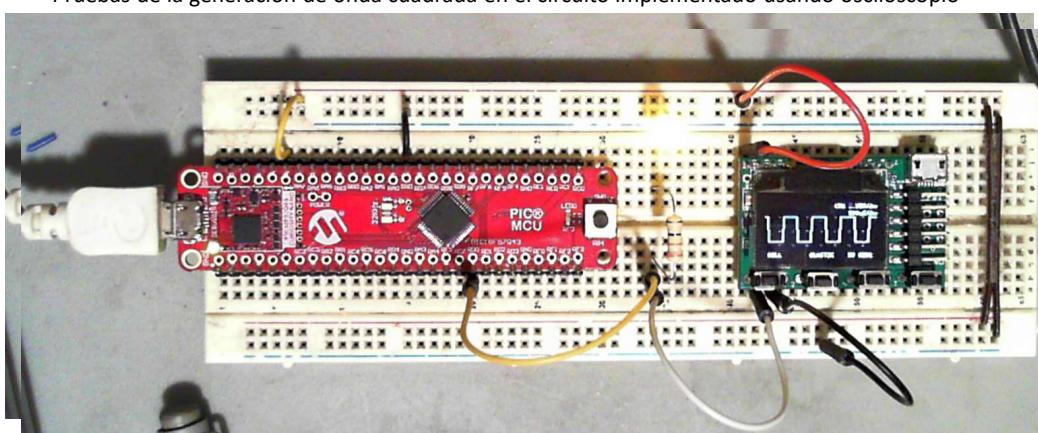
```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSECT upcino, class=CODE, reloc=2, abs
5  upcino:
6      ORG 000000H
7      bra configuro
8
9      ORG 000100H
10     configuro:
11         movlb 0H
12         movlw 60H
13         movwf OSCCON1, 1
14         movlw 02H
15         movwf OSCFRC, 1
16         movlw 40H
17         movwf OSCEN, 1
18         movlb 3H
19         movlw 90H
20         movwf T0CON0, 1      ;TMRO enabled, 16bit, postscaler 1:1
21         movlw 40H
22         movwf T0CON1, 1      ;FOSC/4, prescaler 1:1, async
23         movlb 4H
24         bcf TRISD, 0, 1    ;RD0 como salida
25         bcf ANSELD, 0, 1    ;RD0 como digital
26
27         inicio:
28             movlb 4H
29             btfs PIR3, 7, 1    ;Pregunto si TMROIF=1
30             bra inicio          ;Aun no se levanta TMROIF
31             btg LATD, 0, 1      ;Complemento a RD0
32             bcf PIR3, 7, 1      ;Bajamos bandera TMROIF
33             movlb 3H
34             movlw 0FEH
35             movwf TMROH, 1
36             movlw 0CH
37             movwf TMROL, 1      ;Precarga de cuenta 65036 a TMRO
38             bra inicio
39
40         end upcino

```

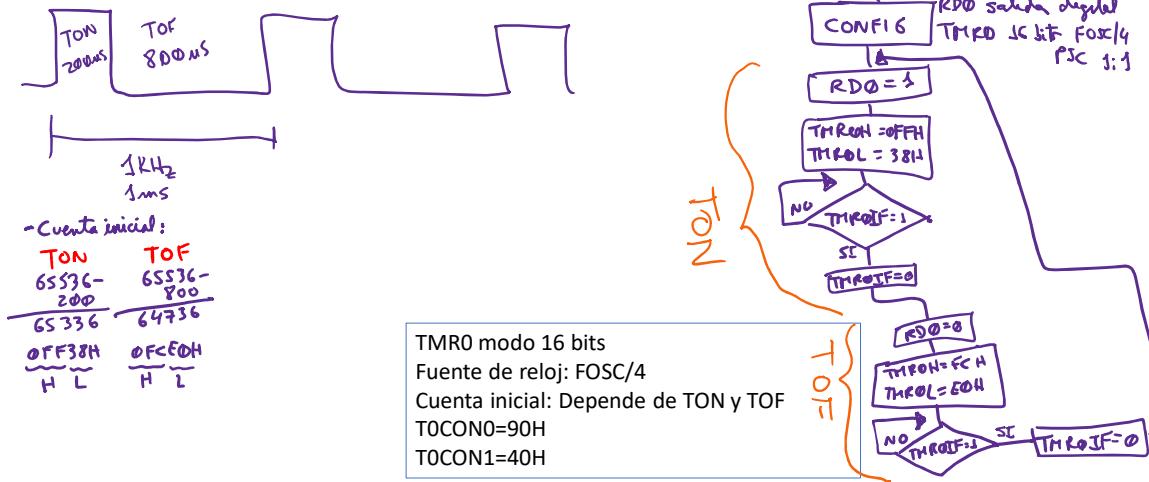
## Ejercicios:

- Cómo generamos una señal cuadrada de 1KHz 50%DC ahora usando el modo 16 bits
- Pruebas de la generación de onda cuadrada en el circuito implementado usando osciloscopio



## Ejercicios:

- Cómo generamos una señal cuadrada de 1KHz 20%DC ahora usando el modo 16 bits



## Ejercicios:

- Cómo generamos una señal cuadrada de 1KHz 20%DC ahora usando el modo 16 bits

```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSECT upcino, class=CODE, reloc=2, abs
5  upcino:
6  ORG 000000H
7  bra configuro
8
9  ORG 000100H
10 configuro:
11  movlb 0H
12  movlw 60H
13  movwf OSCCON1, 1
14  movlw 02H
15  movwf OSCFRQ, 1
16  movlw 40H
17  movwf OSCEN, 1
18  movlb 3H
19  movlw 90H
20  movwf T0CON0, 1      ;TMRO enabled, 16bit, postscaler 1:1
21  movlw 40H
22  movwf T0CON1, 1      ;FOSC/4, prescaler 1:1, async
23  movlb 4H
24  bcf TRISD, 0, 1     ;RD0 como salida
25  bcf ANSEL0, 0, 1     ;RD0 como digital

```

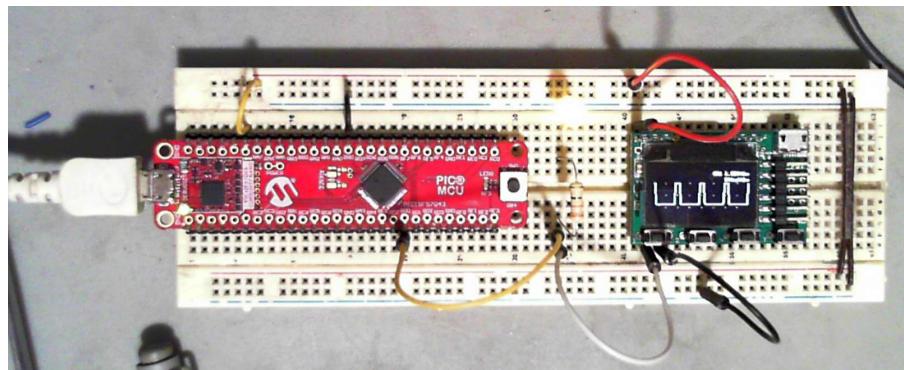
```

27  inicio:
28  movlb 4H
29  bsf LATD, 0, 1      ;RD0 a uno
30  movlb 3H
31  movlw 0FFH
32  movwf TMROH, 1
33  movlw 38H
34  movwf TMROL, 1      ;Precarga de cuenta 65036 a TMRO
35  movlb 4H
36  otro:
37  btfss PIR3, 7, 1    ;Pregunto si TMROIF=1
38  bra otro            ;Aun no se levanta TMROIF
39  bcf PIR3, 7, 1      ;Bajamos bandera TMROIF
40  bcf LATD, 0, 1      ;RD0 a cero
41  movlb 3H
42  movlw 0FCH
43  movwf TMROH, 1
44  movlw 0E0H
45  movwf TMROL, 1      ;Precarga de cuenta 64736 a TMRO
46  movlb 4H
47  otro2:
48  btfss PIR3, 7, 1    ;Pregunto si TMROIF=1
49  bra otro2           ;Aun no se levanta TMROIF
50  bcf PIR3, 7, 1      ;Bajamos bandera TMROIF
51  bra inicio
52
53  end upcino

```

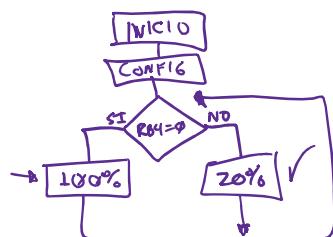
## Ejercicios:

- Cómo generamos una señal cuadrada de 1KHz 20%DC ahora usando el modo 16 bits



¿Cómo intercambio el duty cycle de 20% a 100% mediante el pulsador en RB4?

- Preguntando al inicio por dicho botón



# ¿Cómo intercambio el duty cycle de 20% a 100% mediante el pulsador en RB4?

<pre> 1  PROCESSOR 18F57Q43 2  #include "cabecera.inc" 3 4  PSECT upcino, class=CODE, reloc=2, abs 5  upcino: 6    ORG 000000H 7    bra configuro 8 9    ORG 000080H 10 configuro: 11   movlb 0H 12   movlw 60H 13   movwf OSCCON1, 1 14   movlw 02H 15   movwf OSCFRQ, 1 16   movlw 40H 17   movwf OSCEN, 1 18   movlb 3H 19   movlw 90H 20   movwf T0CON0, 1      ;TMR0 enabled, postsc 1:1 modo 16bit 21   movlw 40H 22   movwf T0CON1, 1      ;FOSC/4 Presc 1:1 23   movlw 0FEH 24   movwf TMROH, 1 25   movlw 0CH 26   movwf TMROL, 1      ;cuenta inicial de 65036 al arranque 27   movlb 4H 28   bcf TRISD, 0, 1     ;RD0 como salida 29   bcf ANSEL0, 0, 1     ;RD0 como digital 30   bsf TRISB, 4, 1     ;RB4 como entrada 31   bcf ANSELB, 4, 1     ;RB4 como digital 32   bsf WFUEB, 4, 1     ;RB4 con pullup activado </pre>	<pre> 34  inicio: 35    btfs PORTB, 4, 1  ;preguntamos si presionaste el boton en RB4 36    bra nopresione 37    bsf LATD, 0, 1    ;RD0 en uno 38    bra inicio 39 40  nopresione: 41    bsf LATD, 0, 1    ;RD0 en uno 42    movlb 3H 43    movlw 0FFH 44    movwf TMROH, 1 45    movlw 38H 46    movwf TMROL, 1    ;cuenta inicial de 65036 al arranque 47    movlb 4H 48    btfs PIR3, 7, 1   ;pregunto si TMROIF=1 49    bra \$-2 50    bcf PIR3, 7, 1   ;falso vuelvo a preguntar 51    bcf LATD, 0, 1    ;RD0 en cero 52    movlb 3H 53    movlw 0FCH 54    movwf TMROH, 1 55    movlw 0E0H 56    movwf TMROL, 1    ;cuenta inicial de 65036 al arranque 57    movlb 4H 58    btfs PIR3, 7, 1   ;pregunto si TMROIF=1 59    bra \$-2 60    bcf PIR3, 7, 1   ;falso vuelvo a preguntar 61    bra inicio         ;retorno a inicio 62 63  end upcino </pre>
--	--

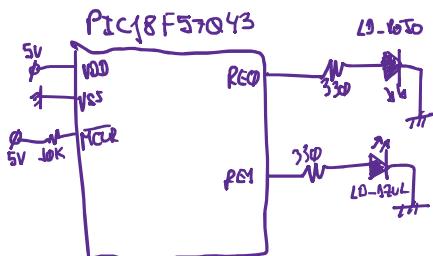
## Observaciones

- No sale exacto en las pruebas, esto debido a que no se ha contemplado el tiempo en que se demora en ejecutar las instrucciones
- Se tiene que hacer una compensación haciendo que el TMRO cuente menos cuentas (periodo menor de temporizado) y se ajusta empleando nops
- Haciendo este tipo de compensaciones nos acercaremos a la frecuencia solicitada pero no será exacto.
- Esto nos hace pensar que no se va a poder hacer sistemas en tiempo real.

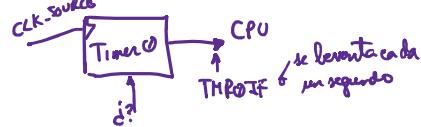
## Ejercicio 2024-1

- Hacer una señal de cruce de tren empleando RE0 y RE1 como salidas hacia dos LEDs y el modulo Timer0 haciendo la base de tiempo de cambio de un segundo

a) Hardware

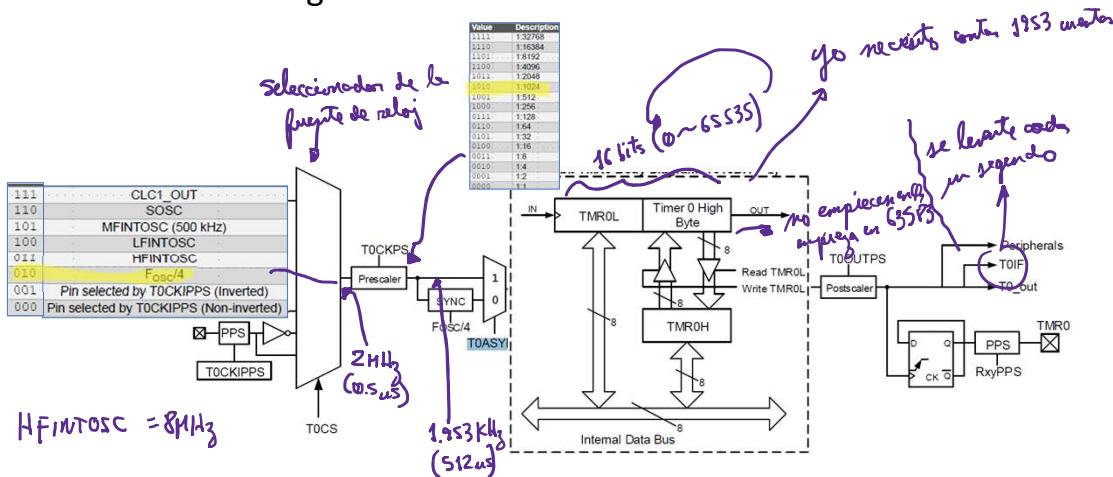


b) Análisis del Timer0 → Me avisa cada 1 segundo



## Ejercicio 2024-1

- Hacer una señal de cruce de tren empleando RE0 y RE1 como salidas hacia dos LEDs y el modulo Timer0 haciendo la base de tiempo de cambio de un segundo



## Ejercicio 2024-1

- Hacer una señal de cruce de tren empleando RE0 y RE1 como salidas hacia dos LEDs y el modulo Timer0 haciendo la base de tiempo de cambio de un segundo

*Detalles de la configuración del Timer0:*

- Fuente de reloj: LFINTOSC/4
- Modo: 16 bit ✓
- Prescaler: 1:1024 ✓
- Postscaler: 1:1 ✓
- Cuenta inicial: 63583

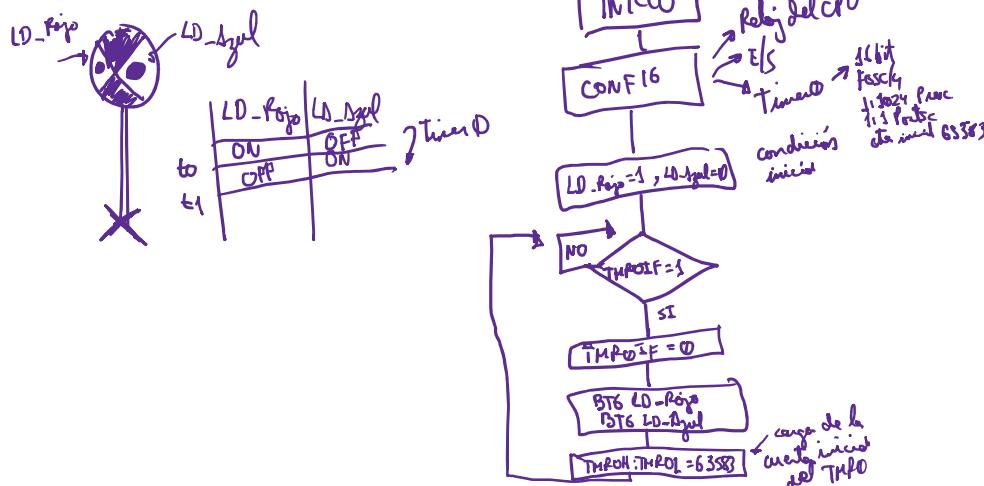
Name: T0CON0		Address: 0x31A	
Timer0 Control Register 0			
Bit	7	6	5
Access	0	X	0
Reset	0	0	0
Value	0	0	0
Description	TMRO is enabled and operating	The module is disabled	
Bit 5 – OUT	TMRO Output		
Bit 4 – M0M16	16-Bit Timer Operation Select		
Value	TMRO is a 16-bit timer	TMRO is an 8-bit timer	
Bits 3:0 – OUTPS[3:0]	TMRO Output Postscaler (Divider) Select		
Value	1:16 Postscaler	1:15 Postscaler	
Description	1:16 Postscaler	1:15 Postscaler	
1111	1:16 Postscaler	1:15 Postscaler	
1110	1:15 Postscaler	1:14 Postscaler	
1100	1:14 Postscaler	1:13 Postscaler	
1011	1:12 Postscaler	1:11 Postscaler	
1010	1:10 Postscaler	1:9 Postscaler	
1000	1:8 Postscaler	1:7 Postscaler	
0111	1:9 Postscaler	1:8 Postscaler	
0110	1:7 Postscaler	1:6 Postscaler	
0100	1:6 Postscaler	1:5 Postscaler	
0011	1:4 Postscaler	1:3 Postscaler	
0110	1:3 Postscaler	1:2 Postscaler	
0001	1:2 Postscaler	1:1 Postscaler	
0000	1:1 Postscaler		

Name: T0CON1		Address: 0x31B	
Timer0 Control Register 1			
Bit	7	6	5
Access	0	CEN	0
Reset	0	0	0
Value	0	0	0
Description	TMRO is enabled and operating	TMRO is disabled	
Bit 5 – CS[2:0] Timer0 Clock Source Select			
Value	000	001	100
Description	Pin selected by T0CKIPPS (Inverted)	Pin selected by T0CKPPS (Not Inverted)	
Bit 4 – ASYNC_TMR0 Input Asynchronization Enable			
Value	0	1	
Description	The input to the TMRO counter is not synchronized to system clocks	The input to the TMRO counter is synchronized to Fosc/4	
Bits 3:0 – CKPS[3:0] Prescaler Rate Selected			
Value	1:32768	1:16384	
Description	1:32768	1:16384	
1111	1:32768	1:16384	
1110	1:16384		
1101	1:8192		
1100	1:4096		
1011	1:2048		
1010	1:1024		
1001	1:512		
0000	1:256		
0111	1:128		
0110	1:64		
0101	1:32		
0100	1:16		
0011	1:8		
0010	1:4		
0001	1:2		
0000	1:1		

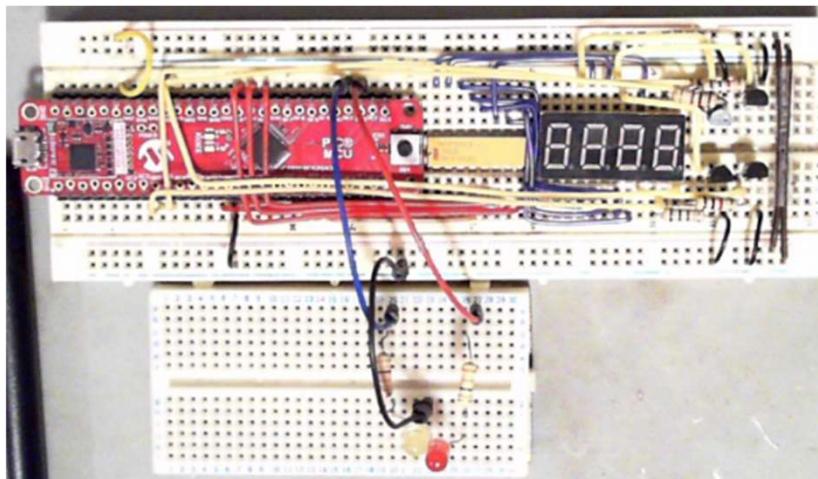
## Ejercicio 2024-1

- Hacer una señal de cruce de tren empleando RE0 y RE1 como salidas hacia dos LEDs y el modulo Timer0 haciendo la base de tiempo de cambio de un segundo



## Ejercicio 2024-1

- Hacer una señal de cruce de tren empleando RE0 y RE1 como salidas hacia dos LEDs y el modulo Timer0 haciendo la base de tiempo de cambio de un segundo



## Ejercicio 2024-1

```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSECT upcino, class=CODE, reloc=2, abs
5  upcino:
6    ORG 000000H
7    bra configuro
8
9    ORG 000100H
10 configuro:
11   movlb 0H
12   movlw 60H
13   movwf OSCCON1, 1
14   movlw 03H
15   movwf OSCFRC, 1
16   movlw 40H
17   movwf OSCEN, 1      ;HFINTOSC a 8MHz
18   movlb 3H
19   movlw 90H
20   movwf T0CON0, 1
21   movlw 4AH           ;TMRO 16bit,psc 1:1024, post 1:1
22   movwf T0CON1, 1      ;TMRO HFINTOSC/4
23   movlw 0F8H
24   movwf TMROH, 1
25   movlw 5FH
26   movwf TMROL, 1      ;cuenta inicial de 63583
27   movlb 4H
28   bcf TRISE, 0, 1     ;RE0 como salida
29   bcf TRISE, 1, 1     ;RE1 como salida
30   bcf ANSEL0, 0, 1     ;RE0 como digital
31   bcf ANSEL0, 1, 1     ;RE1 como digital
32   bsf LATB, 0, 1       ;LD_Rojo encendido
33   bsf LATB, 1, 1       ;LD_Azul apagado

```

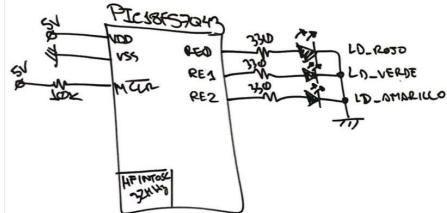
- Hacer una señal de cruce de tren empleando RE0 y RE1 como salidas hacia dos LEDs y el modulo Timer0 haciendo la base de tiempo de cambio de un segundo

35	inicio:	
36	btfs PIR3, 7, 1	;Pregunto si se levanto bandera de TMRO
37	bra \$-2	;Falso, vuelve a preguntar
38	baf PIR3, 7, 1	;Verdad, bajo la bandera TMRO
39	btg LATB, 0, 1	;Bascula RE0
40	btg LATB, 1, 1	;Bascula RE1
41	movlb 3H	
42	movlw 0F8H	
43	movwf TMROH, 1	
44	movlw 5FH	
45	movwf TMROL, 1	;cuenta inicial de 63583
46	movlb 4H	
47	bra inicio	
48		
49	end upcino	

Pizarra EL54-2 25/04/2024

¿Temporizar?

$$T = \frac{\lambda}{f}$$

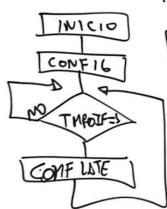


Jueves 9 de Mayo 2PM ~ 6PM  
BYOB

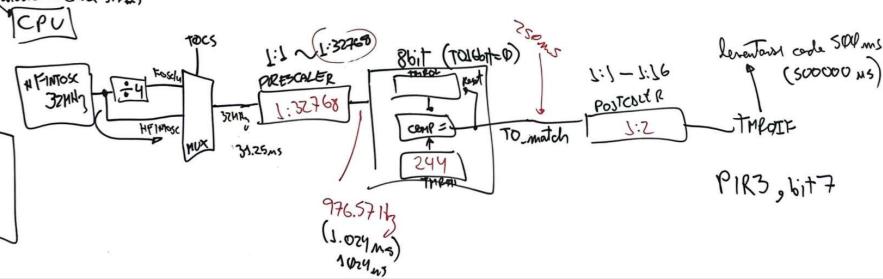
342

REL REL	REC	
OFF	OFF	ON
OFF	ON	OFF
ON	OFF	OFF
OFF	ON	OFF

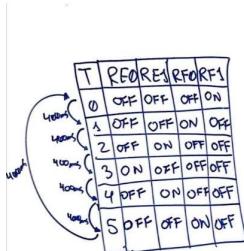
El TinerO de los Temposijas 500ms  
Atención cada 500ms



$$\text{TOCON} \varphi = 81\text{H} \rightarrow \text{SON}$$



Pizarra EL56-1 26/04/2024



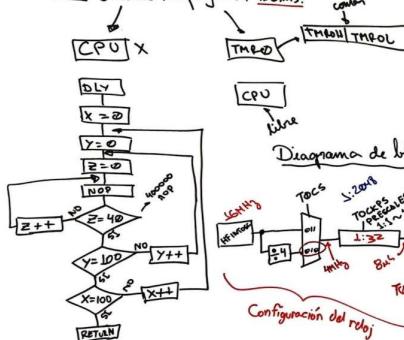
Hora de presentación:  
12:30 PM

## El Timer Ⓛ

$$T = \frac{1}{f}$$

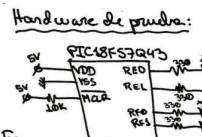
- Sirve para temporizar
  - Revisar cap. 24 del datasheet

Caso: Generar temporización

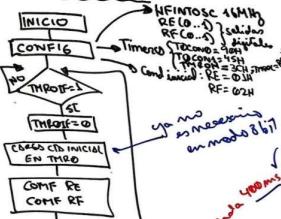


$$T_{OCNO} = 90H$$

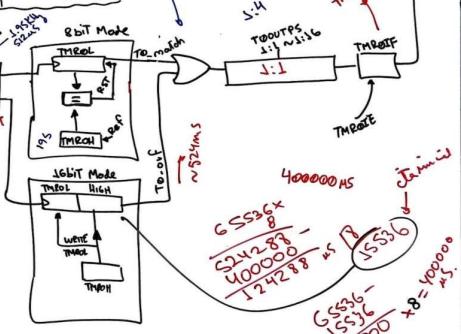
$$T_{OCNI} = 45H$$



## Funcionamiento:

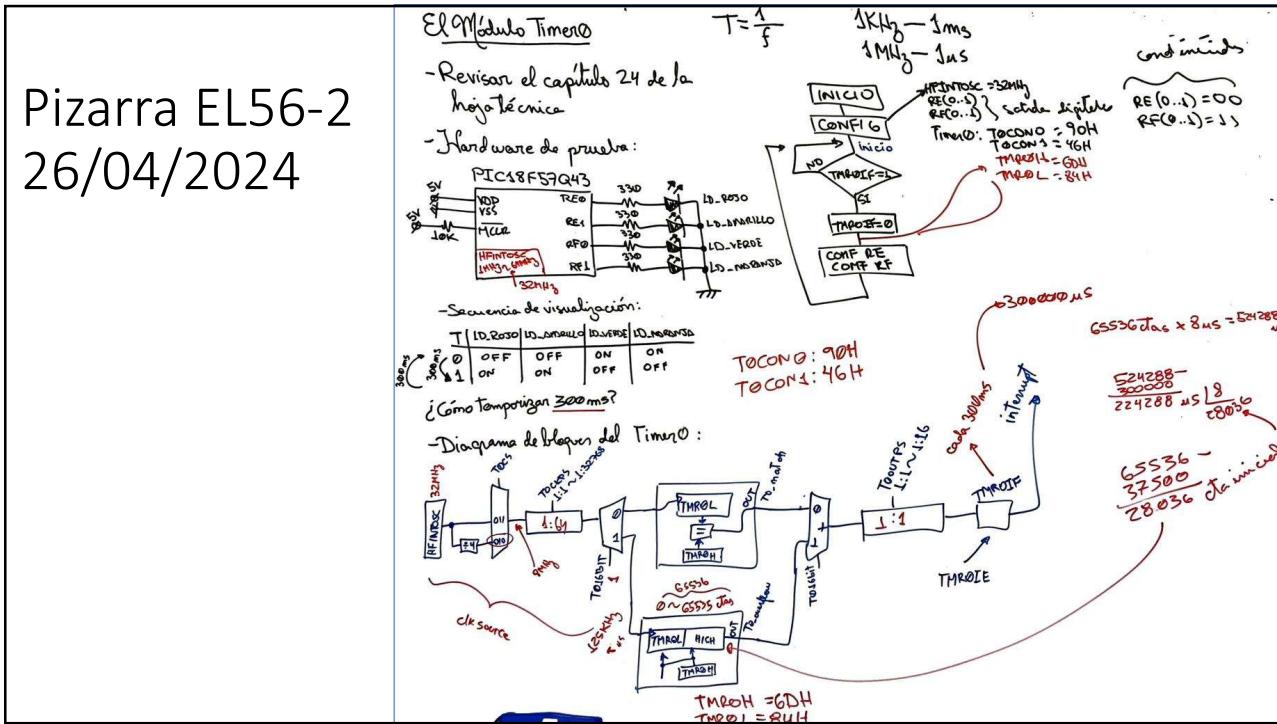


## Diagrama de bloques del Timer 0



# Pizarra EL56-2

26/04/2024



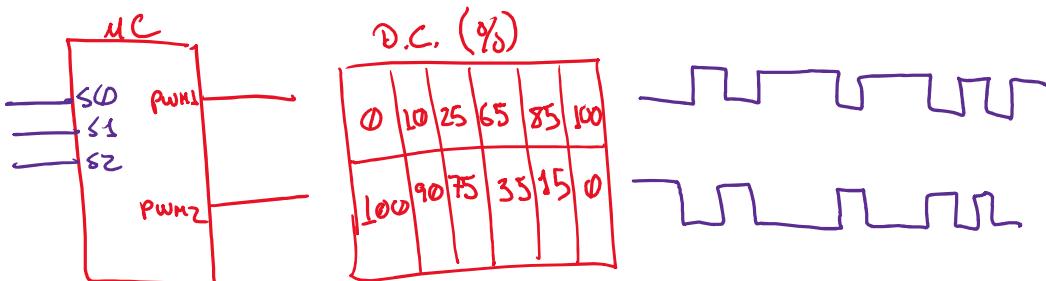
```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSECT upcino, class=CODE, reloc=2, abs
5
6  TAQUIS EQU 500H
7
8 upcino:
9   ORG 000000H
10  bra configuro
11
12  ORG 000700H
13 tabla1: DB 01H, 02H, 00H, 00H, 00H, 02H, 01H, 02H, 01H, 02H, 00H, 03H, 00H, 03H, 00H
14
15  ORG 000800H
16 tabla2: DB 00H, 00H, 01H, 02H, 01H, 00H, 02H, 01H, 02H, 01H, 00H, 03H, 00H, 03H, 00H
17
18  ORG 000100H
19 configuro:
20  movlb 0H
21  movlw 60H
22  movwf OSCCON1, 1
23  movlw 05H
24  movwf OSCFRQ, 1
25  movlw 40H
26  movwf OSCEN, 1      ;HFINTOSC a 32MHz
27  movlb 3H
28  movlw 90H
29  movwf TOCON0, 1
30  movlw 44H
31  movwf TOCON1, 1      ;TMRO 16bit, FOSC/4, PRESC 1:64, POSTS 1:1
32  movlw 6DH
33  movwf TMR0H, 1
34  movlw 84H
35  movwf TMROL, 1      ;cuenta inicial 28036 (6D84H)
36  movlb 5H
37  clrf TAQUIS, 1
38  movlb 4H
39  movlw 0FCH
40  movwf TRISE, 1
41  movwf ANSEL1, 1      ;RE(0..1) salidas digitales
42
43  movwf TRISE, 1
44  movwf ANSELF, 1      ;RF(0..1) salidas digitales
45  movlw 01H
46  movwf LATE, 1
47  movlw 02H
48  movwf LATF, 1      ;condiciones iniciales
49  clrf TBLPTRU, 1
50  movlw 07H
51  movwf TBLPTRH, 1
52  clrf TBLPTRL, 1
53 inicio:
54  btfs PIR3, 7, 1
55  bra $-2
56  bcf PIR3, 7, 1
57  movlw 6DH
58  movwf TMR0H, 1
59  movlw 84H
60  movwf TMROL, 1      ;cuenta inicial 28036 (6D84H)
61  movlw 07H
62  movwf TBLPTRH, 1
63  movlw 3H
64  movlw 14
65  cpfseq TAQUIS, 1
66  bra aunno
67  clrf TAQUIS, 1
68  bra siguiente
69 aunno:
70  incf TAQUIS, 1, 1
71 siguiente:
72  movf TAQUIS, 0, 1
73  movlb 4H
74  movwf TBLPTRL, 1
75  TBLRD+
76  movff TABLAT, LATE
77  movlw 08H
78  movwf TBLPTRH, 1
79  TBLRD+
80  movff TABLAT, LATF
81  bra inicio
82
83 end upcino

```

## Ejercicios propuestos

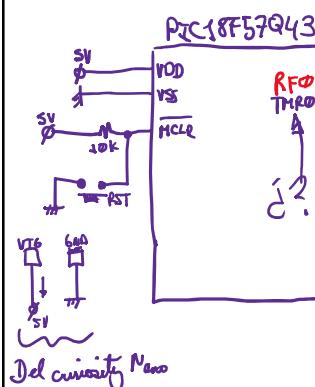
- En un PoR. ¿En qué estado se encuentra el Timer0, encendido o apagado?
- Si Fosc = 24MHz. ¿Cuál es la temporización máxima del Timer0 en modo 16 bits?
- Desarrollar un generador de PWM 2KHz con dos salidas complementarias y con opciones de dutycycle siguientes: 0%, 10%, 25%, 65%, 85% y 100%



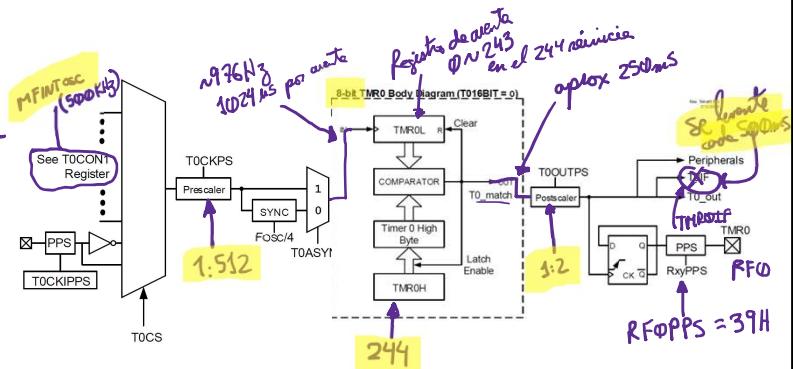
## Ejercicio 2024-2

- Realizar el parpadeo de un LED con TON y TOF de 500ms (el periodo total es de 1 segundo) aproximadamente y empleando el Timer0 en modo 8 bits, el LED estará conectado en el puerto RF0 y éste será la salida del TMRO del PIC18F57Q43.

Hardware:



Analisis del temporizado de Timer0:



# Ejercicio 2024-2

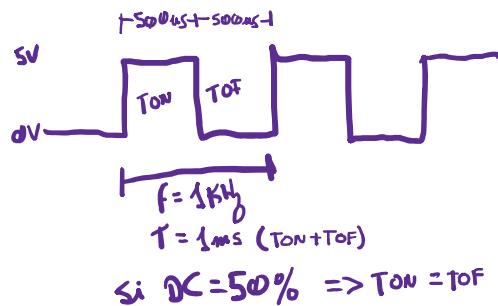
<p><b>Name:</b> T0CON0      <b>Address:</b> 0x31A      <b>84H</b></p> <p>Timer0 Control Register 0</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10px;">Bit</td> <td style="width: 10px;">7</td> <td style="width: 10px;">0</td> <td style="width: 10px;">1</td> <td style="width: 10px;">0</td> <td style="width: 10px;">1</td> <td style="width: 10px;">0</td> <td style="width: 10px;">0</td> <td style="width: 10px;">0</td> <td style="width: 10px;">0</td> <td style="width: 10px;">1</td> <td style="width: 10px;">R/W</td> </tr> <tr> <td>Access</td> <td>R/W</td> <td>0</td> <td>R/W</td> <td>0</td> <td>R/W</td> <td>0</td> <td>R/W</td> <td>0</td> <td>R/W</td> <td>0</td> <td>R/W</td> </tr> </table> <p><b>Bit 7 – EN</b> TMRO Enable</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10px;"><b>Value</b></td> <td style="width: 10px;"><b>Description</b></td> </tr> <tr> <td>1</td> <td>The module is enabled and operating ✓</td> </tr> <tr> <td>0</td> <td>The module is disabled</td> </tr> </table> <p><b>Bit 5 – OUT</b> TMRO Output</p> <p><b>Bit 4 – MD16</b> 16-Bit Timer Operation Select</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10px;"><b>Value</b></td> <td style="width: 10px;"><b>Description</b></td> </tr> <tr> <td>1</td> <td>TMRO is a 16-bit timer ✓</td> </tr> <tr> <td>0</td> <td>TMRO is an 8-bit timer</td> </tr> </table> <p><b>Bits 3:0 – OUTPS[3:0]</b> TMRO Output Postscaler (Divider) Select</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10px;"><b>Value</b></td> <td style="width: 10px;"><b>Description</b></td> </tr> <tr> <td>1111</td> <td>1:16 Postscaler</td> </tr> <tr> <td>1110</td> <td>1:15 Postscaler</td> </tr> <tr> <td>1101</td> <td>1:14 Postscaler</td> </tr> <tr> <td>1100</td> <td>1:13 Postscaler</td> </tr> <tr> <td>1011</td> <td>1:12 Postscaler</td> </tr> <tr> <td>1010</td> <td>1:11 Postscaler</td> </tr> <tr> <td>1001</td> <td>1:10 Postscaler</td> </tr> <tr> <td>1000</td> <td>1:9 Postscaler</td> </tr> <tr> <td>0111</td> <td>1:8 Postscaler</td> </tr> <tr> <td>0110</td> <td>1:7 Postscaler</td> </tr> <tr> <td>0101</td> <td>1:6 Postscaler</td> </tr> <tr> <td>0100</td> <td>1:5 Postscaler</td> </tr> <tr> <td>0011</td> <td>1:4 Postscaler</td> </tr> <tr> <td>0010</td> <td>1:3 Postscaler</td> </tr> <tr> <td>0001</td> <td>1:2 Postscaler ✓</td> </tr> <tr> <td>0000</td> <td>1:1 Postscaler</td> </tr> </table>	Bit	7	0	1	0	1	0	0	0	0	1	R/W	Access	R/W	0	R/W	<b>Value</b>	<b>Description</b>	1	The module is enabled and operating ✓	0	The module is disabled	<b>Value</b>	<b>Description</b>	1	TMRO is a 16-bit timer ✓	0	TMRO is an 8-bit timer	<b>Value</b>	<b>Description</b>	1111	1:16 Postscaler	1110	1:15 Postscaler	1101	1:14 Postscaler	1100	1:13 Postscaler	1011	1:12 Postscaler	1010	1:11 Postscaler	1001	1:10 Postscaler	1000	1:9 Postscaler	0111	1:8 Postscaler	0110	1:7 Postscaler	0101	1:6 Postscaler	0100	1:5 Postscaler	0011	1:4 Postscaler	0010	1:3 Postscaler	0001	1:2 Postscaler ✓	0000	1:1 Postscaler	<p><b>Name:</b> T0CON1      <b>Address:</b> 0x31B      <b>A9H</b></p> <p>Timer0 Control Register 1</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10px;">Bit</td> <td style="width: 10px;">1</td> <td style="width: 10px;">0</td> <td style="width: 10px;">1</td> <td style="width: 10px;">0</td> <td style="width: 10px;">1</td> <td style="width: 10px;">0</td> <td style="width: 10px;">0</td> <td style="width: 10px;">1</td> <td style="width: 10px;">R/W</td> </tr> <tr> <td>Access</td> <td>R/W</td> <td>0</td> <td>R/W</td> <td>0</td> <td>R/W</td> <td>0</td> <td>R/W</td> <td>0</td> <td>R/W</td> </tr> </table> <p><b>Bits 7:5 – CS[2:0]</b> Timer0 Clock Source Select</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10px;"><b>Value</b></td> <td style="width: 10px;"><b>Description</b></td> </tr> <tr> <td>111</td> <td>CLC1_OUT</td> </tr> <tr> <td>110</td> <td>OSCINTOSC</td> </tr> <tr> <td>101</td> <td>MFINTOSC (600 kHz)</td> </tr> <tr> <td>100</td> <td>LFIINTOSC</td> </tr> <tr> <td>011</td> <td>HFINTOSC</td> </tr> <tr> <td>010</td> <td>Fosc/4</td> </tr> <tr> <td>001</td> <td>Pin selected by TOCKIPPS (Inverted)</td> </tr> <tr> <td>000</td> <td>Pin selected by TOCKIPPS (Non-inverted)</td> </tr> </table> <p><b>Bit 4 – ASYNC</b> TMRO Input Asynchronization Enable</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10px;"><b>Value</b></td> <td style="width: 10px;"><b>Description</b></td> </tr> <tr> <td>1</td> <td>The input to the TMRO counter is not synchronized to system clocks ✓</td> </tr> <tr> <td>0</td> <td>The input to the TMRO counter is synchronized to Fosc/4</td> </tr> </table> <p><b>Bits 3:0 – CKPS[3:0]</b> Prescaler Rate Select</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10px;"><b>Value</b></td> <td style="width: 10px;"><b>Description</b></td> </tr> <tr> <td>1111</td> <td>1:32768</td> </tr> <tr> <td>1110</td> <td>1:16384</td> </tr> <tr> <td>1101</td> <td>1:8192</td> </tr> <tr> <td>1100</td> <td>1:4096</td> </tr> <tr> <td>1011</td> <td>1:2048</td> </tr> <tr> <td>1010</td> <td>1:1024 ✓</td> </tr> <tr> <td>1001</td> <td>1:512</td> </tr> <tr> <td>1000</td> <td>1:256</td> </tr> <tr> <td>0111</td> <td>1:128</td> </tr> <tr> <td>0110</td> <td>1:64</td> </tr> <tr> <td>0101</td> <td>1:32</td> </tr> <tr> <td>0100</td> <td>1:16</td> </tr> <tr> <td>0011</td> <td>1:8</td> </tr> <tr> <td>0010</td> <td>1:4</td> </tr> <tr> <td>0001</td> <td>1:2</td> </tr> <tr> <td>0000</td> <td>1:1</td> </tr> </table>	Bit	1	0	1	0	1	0	0	1	R/W	Access	R/W	0	R/W	0	R/W	0	R/W	0	R/W	<b>Value</b>	<b>Description</b>	111	CLC1_OUT	110	OSCINTOSC	101	MFINTOSC (600 kHz)	100	LFIINTOSC	011	HFINTOSC	010	Fosc/4	001	Pin selected by TOCKIPPS (Inverted)	000	Pin selected by TOCKIPPS (Non-inverted)	<b>Value</b>	<b>Description</b>	1	The input to the TMRO counter is not synchronized to system clocks ✓	0	The input to the TMRO counter is synchronized to Fosc/4	<b>Value</b>	<b>Description</b>	1111	1:32768	1110	1:16384	1101	1:8192	1100	1:4096	1011	1:2048	1010	1:1024 ✓	1001	1:512	1000	1:256	0111	1:128	0110	1:64	0101	1:32	0100	1:16	0011	1:8	0010	1:4	0001	1:2	0000	1:1								
Bit	7	0	1	0	1	0	0	0	0	1	R/W																																																																																																																																										
Access	R/W	0	R/W	0	R/W	0	R/W	0	R/W	0	R/W																																																																																																																																										
<b>Value</b>	<b>Description</b>																																																																																																																																																				
1	The module is enabled and operating ✓																																																																																																																																																				
0	The module is disabled																																																																																																																																																				
<b>Value</b>	<b>Description</b>																																																																																																																																																				
1	TMRO is a 16-bit timer ✓																																																																																																																																																				
0	TMRO is an 8-bit timer																																																																																																																																																				
<b>Value</b>	<b>Description</b>																																																																																																																																																				
1111	1:16 Postscaler																																																																																																																																																				
1110	1:15 Postscaler																																																																																																																																																				
1101	1:14 Postscaler																																																																																																																																																				
1100	1:13 Postscaler																																																																																																																																																				
1011	1:12 Postscaler																																																																																																																																																				
1010	1:11 Postscaler																																																																																																																																																				
1001	1:10 Postscaler																																																																																																																																																				
1000	1:9 Postscaler																																																																																																																																																				
0111	1:8 Postscaler																																																																																																																																																				
0110	1:7 Postscaler																																																																																																																																																				
0101	1:6 Postscaler																																																																																																																																																				
0100	1:5 Postscaler																																																																																																																																																				
0011	1:4 Postscaler																																																																																																																																																				
0010	1:3 Postscaler																																																																																																																																																				
0001	1:2 Postscaler ✓																																																																																																																																																				
0000	1:1 Postscaler																																																																																																																																																				
Bit	1	0	1	0	1	0	0	1	R/W																																																																																																																																												
Access	R/W	0	R/W	0	R/W	0	R/W	0	R/W																																																																																																																																												
<b>Value</b>	<b>Description</b>																																																																																																																																																				
111	CLC1_OUT																																																																																																																																																				
110	OSCINTOSC																																																																																																																																																				
101	MFINTOSC (600 kHz)																																																																																																																																																				
100	LFIINTOSC																																																																																																																																																				
011	HFINTOSC																																																																																																																																																				
010	Fosc/4																																																																																																																																																				
001	Pin selected by TOCKIPPS (Inverted)																																																																																																																																																				
000	Pin selected by TOCKIPPS (Non-inverted)																																																																																																																																																				
<b>Value</b>	<b>Description</b>																																																																																																																																																				
1	The input to the TMRO counter is not synchronized to system clocks ✓																																																																																																																																																				
0	The input to the TMRO counter is synchronized to Fosc/4																																																																																																																																																				
<b>Value</b>	<b>Description</b>																																																																																																																																																				
1111	1:32768																																																																																																																																																				
1110	1:16384																																																																																																																																																				
1101	1:8192																																																																																																																																																				
1100	1:4096																																																																																																																																																				
1011	1:2048																																																																																																																																																				
1010	1:1024 ✓																																																																																																																																																				
1001	1:512																																																																																																																																																				
1000	1:256																																																																																																																																																				
0111	1:128																																																																																																																																																				
0110	1:64																																																																																																																																																				
0101	1:32																																																																																																																																																				
0100	1:16																																																																																																																																																				
0011	1:8																																																																																																																																																				
0010	1:4																																																																																																																																																				
0001	1:2																																																																																																																																																				
0000	1:1																																																																																																																																																				

# Ejercicio 2024-2

<pre> 1      PROCESSOR 18F57Q43 2      #include "cabecera.inc" 3 4      PSECT upcino, class=CODE, reloc=2, abs 5      upcino: 6          ORG 000000H 7          bra configuro 8 9          ORG 000100H 10         configuro: 11             ;configurar la fuente de reloj 12             movlb 0H 13             movlw 60H 14             movwf OSCCON1, 1 15             movlw 02H 16             movwf OSCFRQ, 1 17             movlw 40H 18             movwf OSCEN, 1 19             ;configuro el PPS 20             movlb 2H 21             movlw 39H 22             movwf RF0PPS, 1    ;el RF0 conectado a TMRO 23             ;configuro el Timer0 24             movlb 3H 25             movlw 244 26             movwf TMROH, 1      ;valor de ref de comparador 244 27             movlw 81H 28             movwf T0CON0, 1      ;TMRO enabled, 8bit, POSTS 1:2 29             movlw 0A9H 30             movwf T0CON1, 1      ;MFINTOSC, SYNC, PRESC 1:512 </pre>	<pre> 31      ;configurar las E/S 32      movlb 4H 33      bcf TRISF, 0, 1      ;RF0 es salida 34      bcf ANSELF, 0, 1      ;RF0 es digital 35 36      inicio: 37      bra inicio 38      ;el CPU no hace nada despues de la configuración! 39      end upcino </pre>
---	--

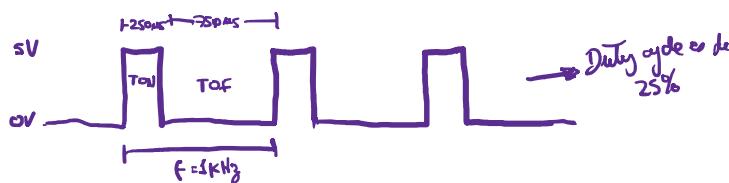
Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Análisis



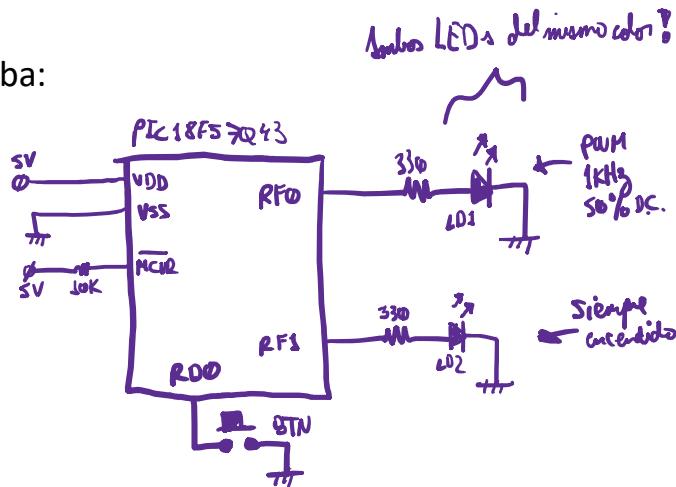
Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- ¿Si necesito bajar el duty cycle a 25%?



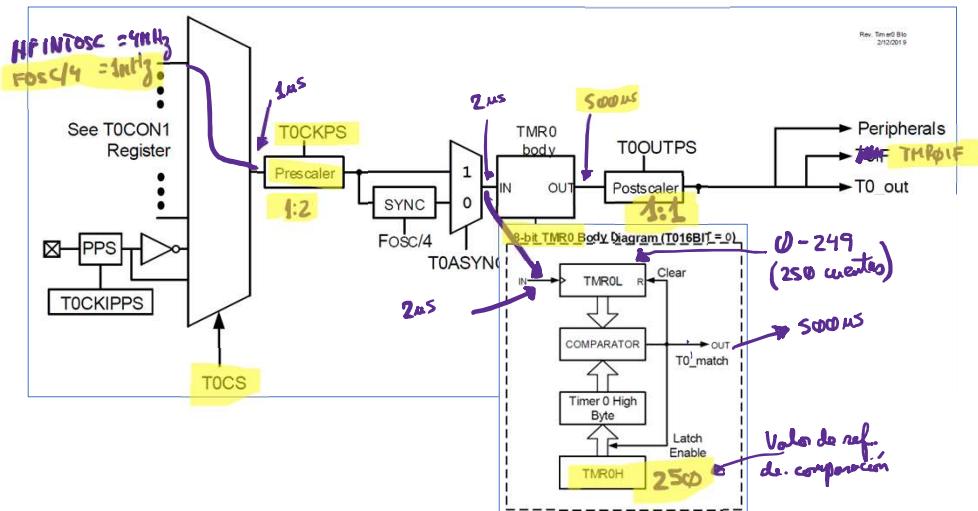
## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Circuito de prueba:



## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Configuración del Timer0 (necesito 500us)



## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Configuracion del Timer0

0000 - 0  
 0001 - 1  
 0010 - 2  
 0011 - 3  
 0100 - 4  
 0101 - 5  
 0110 - 6  
 0111 - 7  
 1000 - 8  
 1001 - 9  
 :  
 :

Name: T0CON0		→ 804	
Address: 0x31A			
Timer0 Control Register 0			
Bit 7 ~ EN TMRO Enable			
Access	Bit	Value	Description
Reset	7	1	The module is enabled and operating
	6	0	The module is disabled
Bit 5 ~ OUT TMRO Output			
Bit 4 ~ MD16 16-bit Timer Operation Select			
Access	Bit	Value	Description
Reset	4	1	TMRO is a 16-bit timer
	0	0	TMRO is an 8-bit timer
Bits 3:0 ~ OUTPS[3:0] TMRO Output Postscaler (Divider) Select			
Access	Bit	Value	Description
Reset	3	1111	1:16 Postscaler
	2	1110	1:15 Postscaler
	1	1101	1:14 Postscaler
	0	1100	1:13 Postscaler
	3	1011	1:12 Postscaler
	2	1010	1:11 Postscaler
	1	1001	1:10 Postscaler
	0	1000	1:9 Postscaler
	3	0111	1:8 Postscaler
	2	0110	1:7 Postscaler
	1	0101	1:6 Postscaler
	0	0100	1:5 Postscaler
	3	0011	1:4 Postscaler
	2	0010	1:3 Postscaler
	1	0001	1:2 Postscaler
	0	0000	1:1 Postscaler ✓

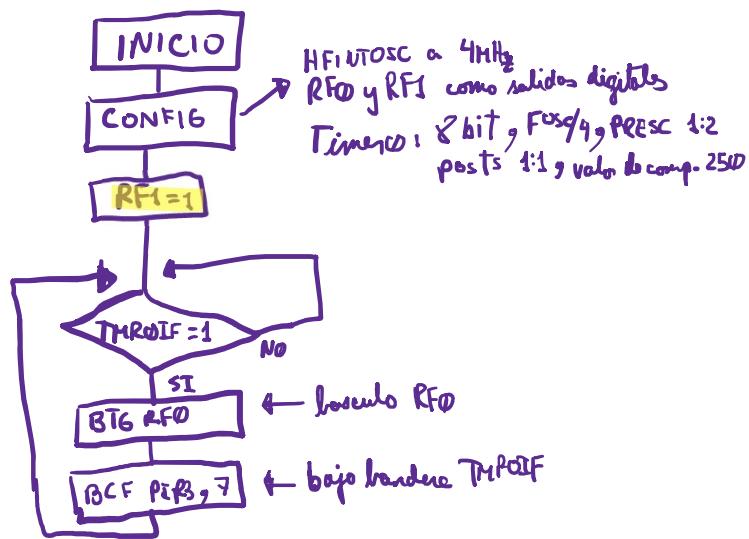
## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Configuracion del Timer0

Name: T0CON1		→ 414	
Address: 0x31B			
Timer0 Control Register 1			
Bit 7:5 ~ CS[2:0] Timer0 Clock Source Select			
Access	Bit	Value	Description
Reset	7	0	COSC
	6	1	SOSC
	5	101	MFINTOSC (500 kHz)
	4	100	LFINTOSC
	3	011	HFINTOSC
	2	010	Fosc/4
	1	001	Pin selected by TOCKI/PSS (Inverted)
	0	000	Pin selected by TOCKI/PSS (Non-inverted)
Bit 4 ~ ASYNC TMRO Input Asynchronization Enable			
Access	Bit	Value	Description
Reset	4	1	The input to the TMRO counter is not synchronized to system clocks
	0	0	The input to the TMRO counter is synchronized to Fosc/4 ✓
Bits 3:0 ~ CKPS[3:0] Prescaler Rate Select			
Access	Bit	Value	Description
Reset	3	1111	1:1
	2	1110	1:16384
	1	1101	1:192
	0	1100	1:4096
	3	1011	1:2048
	2	1010	1:1024
	1	1001	1:512
	0	1000	1:256
	3	0111	1:128
	2	0110	1:64
	1	0101	1:32
	0	0100	1:16
	3	0011	1:8
	2	0010	1:4 ✓
	1	0001	1:2
	0	0000	1:1

## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Diagrama de flujo:



## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- No encuentro el TMROIF. ¿En dónde está?
  - En el bit 7 del registro PIR3 perteneciente al módulo de interrupciones vectorizadas

0x04B0	PIR2	7:0	DMA1AIF	DMA1ORIF	DMA1DCNTIF	DMA1SCNTIF				ADTIF
0x04B1	PIR3	7:0	TMROIF	CCP1IF	TMR1GIF	TMR1IF	TMR2IF	SPI1IF	SPI1TXIF	SPI1RXIF
0x04B2	PIR4	7:0	PWM1IF	PWM1PIF			UI1IF	UI1EIF	UI1TXIF	UI1RXIF

## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Código en Assembler:

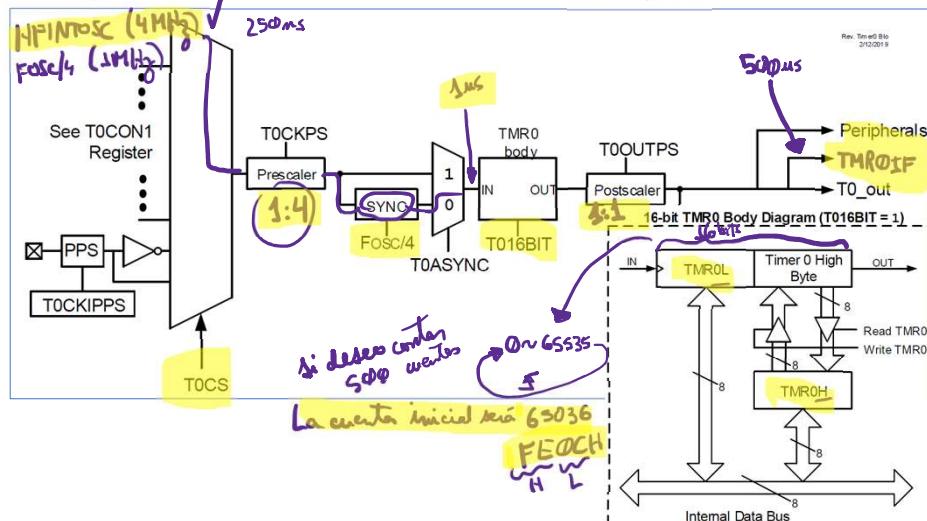
```

1 ;Programa para generar senial de 1KHZ con 50%DC por
2      PROCESSOR 18F57Q43
3      #include "cabecera.inc"
4
5      PSECT upcino, class=CODE, reloc=2, abs
6      upcino:
7          ORG 000000H
8          bra configuro
9
10     ORG 000100H
11 configuro:
12     ;configuracion del modulo de oscilador
13     movlb 0H
14     movlw 60H
15     movwf OSCCON1, 1
16     movlw 02H
17     movwf OSCFRQ, 1
18     movlw 40H
19     movwf OSCEN, 1
20
21     ;configuracion del Timer0
22     movlb 3H
23     movlw 80H
24     movwf T0CON0, 1      ;TMRO enabled, 8bit mode, posts 1:1
25     movlw 41H
26     movwf T0CON1, 1      ;FOSC/4, sync, presc 1:2
27     movlw 250
28     movwf TMROH, 1        ;valor de referencia de comparacion
29     ;configuracion de las E/S
30     movlb 4H
31     movlw 0FCH
32     movwf TRISF, 1 ;RF1 y RF0 como salidas
33     movwf ANSEL, 1 ;RF1 y RF0 como digitales
34     ;condiciones iniciales de operacion
35     bcf LATF, 1, 1 ;RF1 en uno permanentemente
36
37 inicio:
38     btfss PIR3, 7, 1    ;Pregunto si TMROIF se levanto
39     bra inicio          ;No se levanto el TMROIF
40     btg LATF, 0, 1      ;Se levanto el TMROIF
41     bcf PIR3, 7, 1      ;Bajamos el TMROIF
42     bra inicio
43
44 end upcino

```

## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Configuración del Timer0 (necesito 500us)



## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Configuracion del Timer0

Name: T0CON0	
Address: 0x31A	
Timer0 Control Register 0	
Bit	
7	EN
6	CS[0]
5	MD16
4	M16
3	OUTPS[3:0]
2	OUTPS[3:0]
1	OUTPS[3:0]
0	OUTPS[3:0]
Access	R/W
Reset	0
Bit 7 ~ EN TMRO Enable	
Value	Description
1	The module is enabled and operating
0	The module is disabled
Bit 5 ~ OUT TMRO Output	
Value	Description
1	TMRO is a 16-bit timer
0	TMRO is an 8-bit timer
Bits 3:0 ~ OUTPS[3:0] TMRO Output Postscaler (Divider) Select	
Value	Description
1111	1:16 Postscaler
1110	1:15 Postscaler
1101	1:14 Postscaler
1100	1:13 Postscaler
1011	1:12 Postscaler
1010	1:11 Postscaler
1001	1:10 Postscaler
1000	1:9 Postscaler
0111	1:8 Postscaler
0110	1:7 Postscaler
0101	1:6 Postscaler
0100	1:5 Postscaler
0011	1:4 Postscaler
0010	1:3 Postscaler
0001	1:2 Postscaler
0000	1:1 Postscaler

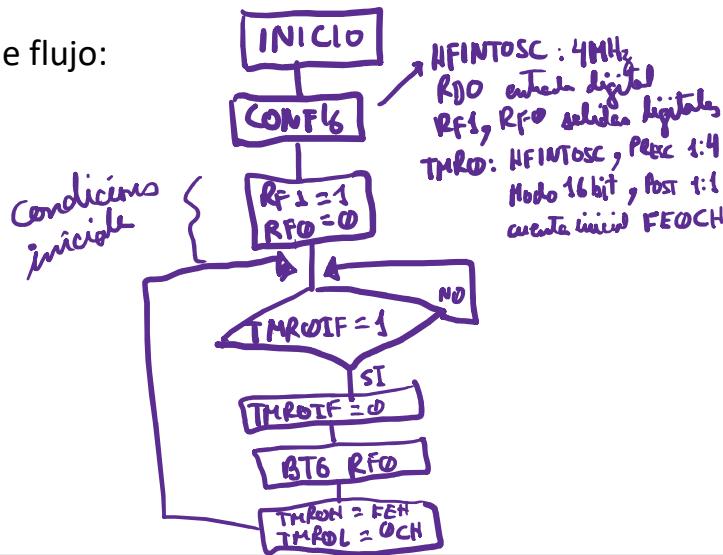
## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Configuracion del Timer0

Name: T0CON1	
Address: 0x31B	
Timer0 Control Register 1	
Bit	
7	CS[0]
6	CKPS[0]
5	RC
4	CKPS[3:0]
3	CKPS[3:0]
2	CKPS[3:0]
1	CKPS[3:0]
0	CKPS[3:0]
Access	R/W
Reset	0
Bits 7:5 ~ CS[2:0] Timer0 Clock Source Select	
Value	Description
111	OSCOUT
110	SOSC
101	MFINTOSC (500 kHz)
100	LFINTOSC
011	HFINTOSC
010	Fosc/4
001	Pin selected by TOCKIPPS (Inverted)
000	Pin selected by TOCKIPPS (Non-inverted)
Bit 4 ~ ASYNC TMRO Input Asynchronization Enable	
Value	Description
1	The input to the TMRO counter is not synchronized to system clocks
0	The input to the TMRO counter is synchronized to Fosc/4
Bits 3:0 ~ CKPS[3:0] Prescaler Rate Select	
Value	Description
1111	1:3
1110	1:16384
1101	1:8192
1100	1:4096
1011	1:2048
1010	1:1024
1001	1:512
1000	1:256
0111	1:128
0110	1:64
0101	1:32
0100	1:16
0011	1:8
0010	1:4
0001	1:2
0000	1:1

## Ejemplo 2025-1: generar una señal cuadrada de 1KHz y 50% de ciclo de trabajo (duty cycle)

- Diagrama de flujo:

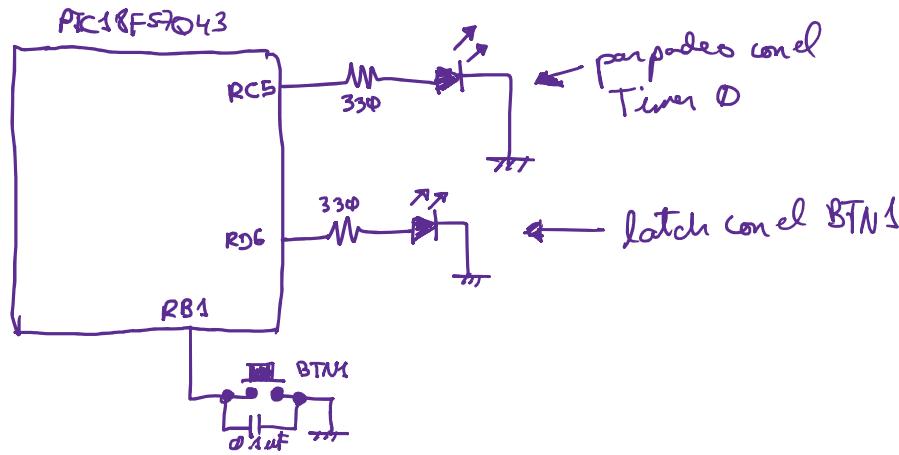


## Ejercicio 2025-1:

- Modificar el código anterior para que el LED en RF0 se encienda y apague de manera gradual.

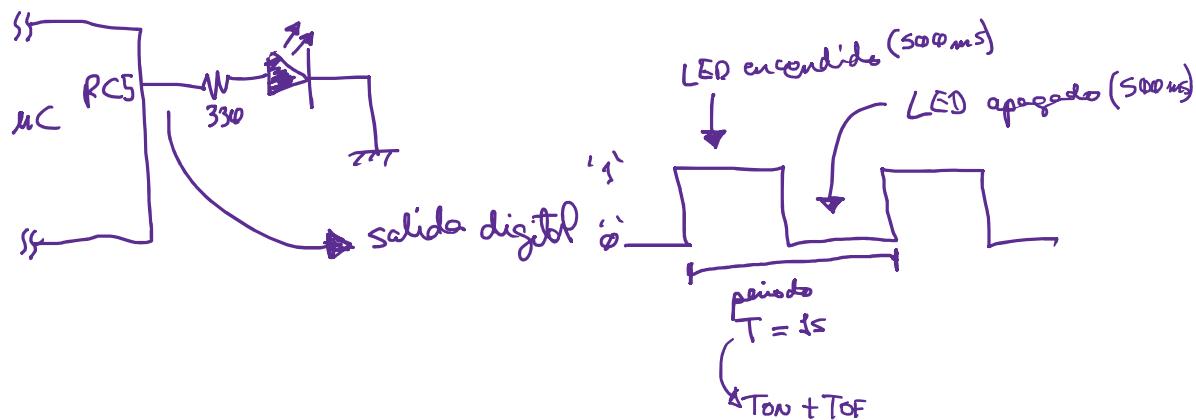
## Ejercicio 2025-2

- Titilar un LED con un periodo de 1 segundo empleando el Timer0 mientras que otro LED funcionará en modo LATCH con un pulsador activo en bajo.



## Ejercicio 2025-2

- Titilar un LED con un periodo de 1 segundo empleando el Timer0 mientras que otro LED funcionará en modo LATCH con un pulsador activo en bajo.

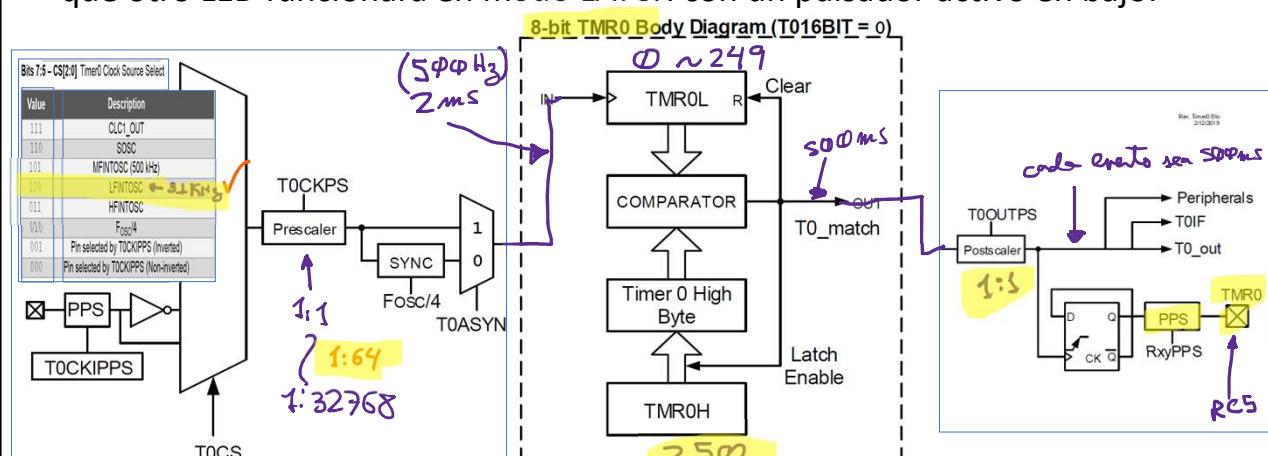


## Ejercicio 2025-2

- Al analizar lo anterior, el Timer0 debe temporizar 500ms
- Cada evento del Timer0 basculará la salida RC5 para que se aprecie el efecto de titileo en el LED.
- El efecto de basculación ya está implementado en el gestor de eventos del Timer0 a través de un flipflop tipo D

## Ejercicio 2025-2

- Titilar un LED con un periodo de 1 segundo empleando el Timer0 mientras que otro LED funcionará en modo LATCH con un pulsador activo en bajo.

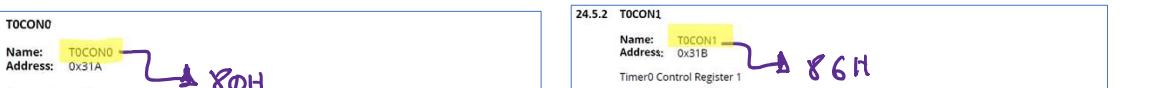


## Ejercicio 2025-2

- Según la configuración anterior, no se puede tener una frecuencia exacta de 500Hz a la entrada del TMRO, se logró llegar a 484Hz(2.066ms) empleando el LFINTOSC y con el divisor de frecuencia a 64.
- Al colocar el valor 250 como referencia del comparador en modo 8 bit, el TMRO estará temporizando 516ms aproximadamente.
- Si quieres que se aproxime al valor de 500ms, debes de restar 8 al valor de referencia (242), resultando en 499.97ms la temporización

## Ejercicio 2025-2

- Configuración de registros del Timer0: TOCON0 y TOCON1

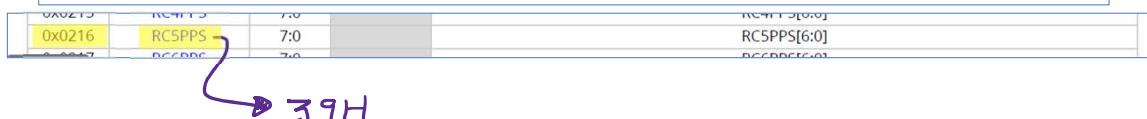
24.5.1 TOCON0		24.5.2 TOCON1																																																																					
Name: TOCON0 Address: 0x31A		Name: TOCON1 Address: 0x31B																																																																					
Timer0 Control Register 0		Timer0 Control Register 1																																																																					
																																																																							
<table border="1"> <thead> <tr> <th>Bit</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th> </tr> </thead> <tbody> <tr> <td>Access</td><td>R/W</td><td>1</td><td>0</td><td>0</td><td>MD16</td><td>0</td><td>OUT</td><td>0</td></tr> <tr> <td>Reset</td><td>0</td><td>0</td><td>0</td><td>0</td><td>R/W</td><td>0</td><td>R/W</td><td>0</td></tr> </tbody> </table>		Bit	7	6	5	4	3	2	1	0	Access	R/W	1	0	0	MD16	0	OUT	0	Reset	0	0	0	0	R/W	0	R/W	0	<table border="1"> <thead> <tr> <th>Bit</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th> </tr> </thead> <tbody> <tr> <td>Access</td><td>R/W</td><td>1</td><td>0</td><td>CSE[2:0]</td><td>0</td><td>0</td><td>CKPS[3:0]</td><td>0</td></tr> <tr> <td>Reset</td><td>0</td><td>0</td><td>0</td><td>R/W</td><td>0</td><td>0</td><td>R/W</td><td>0</td></tr> </tbody> </table>		Bit	7	6	5	4	3	2	1	0	Access	R/W	1	0	CSE[2:0]	0	0	CKPS[3:0]	0	Reset	0	0	0	R/W	0	0	R/W	0														
Bit	7	6	5	4	3	2	1	0																																																															
Access	R/W	1	0	0	MD16	0	OUT	0																																																															
Reset	0	0	0	0	R/W	0	R/W	0																																																															
Bit	7	6	5	4	3	2	1	0																																																															
Access	R/W	1	0	CSE[2:0]	0	0	CKPS[3:0]	0																																																															
Reset	0	0	0	R/W	0	0	R/W	0																																																															
<p><b>Bit 7 - EN</b> TMRO Enable</p> <table border="1"> <thead> <tr> <th>Value</th><th>Description</th> </tr> </thead> <tbody> <tr> <td>1</td><td>The module is enabled and operating ✓</td> </tr> <tr> <td>0</td><td>The module is disabled</td> </tr> </tbody> </table>		Value	Description	1	The module is enabled and operating ✓	0	The module is disabled	<p><b>Bit 7:5 - CSE[2:0]</b> Timer0 Clock Source Select</p> <table border="1"> <thead> <tr> <th>Value</th><th>Description</th> </tr> </thead> <tbody> <tr> <td>111</td><td>CLC1_OUT</td> </tr> <tr> <td>110</td><td>SOC</td> </tr> <tr> <td>101</td><td>MFINTOSC (500 kHz)</td> </tr> <tr> <td>100</td><td>LFINTOSC</td> </tr> <tr> <td>011</td><td>HFINTOSC</td> </tr> <tr> <td>010</td><td>Fosc/4</td> </tr> <tr> <td>001</td><td>Pin selected by TOCKIPPS (Inverted)</td> </tr> <tr> <td>000</td><td>Pin selected by TOCKIPPS (Non-inverted) ✓</td> </tr> </tbody> </table>		Value	Description	111	CLC1_OUT	110	SOC	101	MFINTOSC (500 kHz)	100	LFINTOSC	011	HFINTOSC	010	Fosc/4	001	Pin selected by TOCKIPPS (Inverted)	000	Pin selected by TOCKIPPS (Non-inverted) ✓																																												
Value	Description																																																																						
1	The module is enabled and operating ✓																																																																						
0	The module is disabled																																																																						
Value	Description																																																																						
111	CLC1_OUT																																																																						
110	SOC																																																																						
101	MFINTOSC (500 kHz)																																																																						
100	LFINTOSC																																																																						
011	HFINTOSC																																																																						
010	Fosc/4																																																																						
001	Pin selected by TOCKIPPS (Inverted)																																																																						
000	Pin selected by TOCKIPPS (Non-inverted) ✓																																																																						
<p><b>Bit 5 - OUT</b> TMRO Output</p>		<p><b>Bit 4 - MD16</b> 16-Bit Timer Operation Select</p> <table border="1"> <thead> <tr> <th>Value</th><th>Description</th> </tr> </thead> <tbody> <tr> <td>1</td><td>TMRO is a 16-bit timer</td> </tr> <tr> <td>0</td><td>TMRO is an 8-bit timer ✓</td> </tr> </tbody> </table>		Value	Description	1	TMRO is a 16-bit timer	0	TMRO is an 8-bit timer ✓																																																														
Value	Description																																																																						
1	TMRO is a 16-bit timer																																																																						
0	TMRO is an 8-bit timer ✓																																																																						
<p><b>Bits 3:0 - OUTPS[3:0]</b> TMRO Output Postscaler (Divider) Select</p> <table border="1"> <thead> <tr> <th>Value</th><th>Description</th> </tr> </thead> <tbody> <tr> <td>1111</td><td>1:16 Postscaler</td> </tr> <tr> <td>1110</td><td>1:15 Postscaler</td> </tr> <tr> <td>1101</td><td>1:14 Postscaler</td> </tr> <tr> <td>1100</td><td>1:13 Postscaler</td> </tr> <tr> <td>1011</td><td>1:12 Postscaler</td> </tr> <tr> <td>1010</td><td>1:11 Postscaler</td> </tr> <tr> <td>1001</td><td>1:10 Postscaler</td> </tr> <tr> <td>1000</td><td>1:9 Postscaler</td> </tr> <tr> <td>0111</td><td>1:8 Postscaler</td> </tr> <tr> <td>0110</td><td>1:7 Postscaler</td> </tr> <tr> <td>0101</td><td>1:6 Postscaler</td> </tr> <tr> <td>0100</td><td>1:5 Postscaler</td> </tr> <tr> <td>0011</td><td>1:4 Postscaler</td> </tr> <tr> <td>0010</td><td>1:3 Postscaler</td> </tr> <tr> <td>0001</td><td>1:2 Postscaler</td> </tr> <tr> <td>0000</td><td>1:1 Postscaler ✓</td> </tr> </tbody> </table>		Value	Description	1111	1:16 Postscaler	1110	1:15 Postscaler	1101	1:14 Postscaler	1100	1:13 Postscaler	1011	1:12 Postscaler	1010	1:11 Postscaler	1001	1:10 Postscaler	1000	1:9 Postscaler	0111	1:8 Postscaler	0110	1:7 Postscaler	0101	1:6 Postscaler	0100	1:5 Postscaler	0011	1:4 Postscaler	0010	1:3 Postscaler	0001	1:2 Postscaler	0000	1:1 Postscaler ✓	<p><b>Bit 4 - ASYNC</b> TMRO Input Asynchronization Enable</p> <table border="1"> <thead> <tr> <th>Value</th><th>Description</th> </tr> </thead> <tbody> <tr> <td>1</td><td>The input to the TMRO counter is not synchronized to system clocks</td> </tr> <tr> <td>0</td><td>The input to the TMRO counter is synchronized to Fosc/4 ✓</td> </tr> </tbody> </table>		Value	Description	1	The input to the TMRO counter is not synchronized to system clocks	0	The input to the TMRO counter is synchronized to Fosc/4 ✓																												
Value	Description																																																																						
1111	1:16 Postscaler																																																																						
1110	1:15 Postscaler																																																																						
1101	1:14 Postscaler																																																																						
1100	1:13 Postscaler																																																																						
1011	1:12 Postscaler																																																																						
1010	1:11 Postscaler																																																																						
1001	1:10 Postscaler																																																																						
1000	1:9 Postscaler																																																																						
0111	1:8 Postscaler																																																																						
0110	1:7 Postscaler																																																																						
0101	1:6 Postscaler																																																																						
0100	1:5 Postscaler																																																																						
0011	1:4 Postscaler																																																																						
0010	1:3 Postscaler																																																																						
0001	1:2 Postscaler																																																																						
0000	1:1 Postscaler ✓																																																																						
Value	Description																																																																						
1	The input to the TMRO counter is not synchronized to system clocks																																																																						
0	The input to the TMRO counter is synchronized to Fosc/4 ✓																																																																						
<p><b>Bits 3:0 - CKPS[3:0]</b> Prescaler Rate Select</p> <table border="1"> <thead> <tr> <th>Value</th><th>Description</th> </tr> </thead> <tbody> <tr> <td>1111</td><td>1:1</td> </tr> <tr> <td>1110</td><td>1:16384</td> </tr> <tr> <td>1101</td><td>1:8192</td> </tr> <tr> <td>1100</td><td>1:4096</td> </tr> <tr> <td>1011</td><td>1:2048</td> </tr> <tr> <td>1010</td><td>1:1024</td> </tr> <tr> <td>1001</td><td>1:512</td> </tr> <tr> <td>1000</td><td>1:256</td> </tr> <tr> <td>0111</td><td>1:128</td> </tr> <tr> <td>0110</td><td>1:64</td> </tr> <tr> <td>0101</td><td>1:32</td> </tr> <tr> <td>0100</td><td>1:16</td> </tr> <tr> <td>0011</td><td>1:8</td> </tr> <tr> <td>0010</td><td>1:4</td> </tr> <tr> <td>0001</td><td>1:2</td> </tr> <tr> <td>0000</td><td>1:1</td> </tr> </tbody> </table>		Value	Description	1111	1:1	1110	1:16384	1101	1:8192	1100	1:4096	1011	1:2048	1010	1:1024	1001	1:512	1000	1:256	0111	1:128	0110	1:64	0101	1:32	0100	1:16	0011	1:8	0010	1:4	0001	1:2	0000	1:1	<p><b>Bit 3 - CKPS[3:0]</b> Prescaler Rate Select</p> <table border="1"> <thead> <tr> <th>Value</th><th>Description</th> </tr> </thead> <tbody> <tr> <td>1111</td><td>1:1</td> </tr> <tr> <td>1110</td><td>1:16384</td> </tr> <tr> <td>1101</td><td>1:8192</td> </tr> <tr> <td>1100</td><td>1:4096</td> </tr> <tr> <td>1011</td><td>1:2048</td> </tr> <tr> <td>1010</td><td>1:1024</td> </tr> <tr> <td>1001</td><td>1:512</td> </tr> <tr> <td>1000</td><td>1:256</td> </tr> <tr> <td>0111</td><td>1:128</td> </tr> <tr> <td>0110</td><td>1:64</td> </tr> <tr> <td>0101</td><td>1:32</td> </tr> <tr> <td>0100</td><td>1:16 ✓</td> </tr> <tr> <td>0011</td><td>1:8</td> </tr> <tr> <td>0010</td><td>1:4</td> </tr> <tr> <td>0001</td><td>1:2</td> </tr> <tr> <td>0000</td><td>1:1</td> </tr> </tbody> </table>		Value	Description	1111	1:1	1110	1:16384	1101	1:8192	1100	1:4096	1011	1:2048	1010	1:1024	1001	1:512	1000	1:256	0111	1:128	0110	1:64	0101	1:32	0100	1:16 ✓	0011	1:8	0010	1:4	0001	1:2	0000	1:1
Value	Description																																																																						
1111	1:1																																																																						
1110	1:16384																																																																						
1101	1:8192																																																																						
1100	1:4096																																																																						
1011	1:2048																																																																						
1010	1:1024																																																																						
1001	1:512																																																																						
1000	1:256																																																																						
0111	1:128																																																																						
0110	1:64																																																																						
0101	1:32																																																																						
0100	1:16																																																																						
0011	1:8																																																																						
0010	1:4																																																																						
0001	1:2																																																																						
0000	1:1																																																																						
Value	Description																																																																						
1111	1:1																																																																						
1110	1:16384																																																																						
1101	1:8192																																																																						
1100	1:4096																																																																						
1011	1:2048																																																																						
1010	1:1024																																																																						
1001	1:512																																																																						
1000	1:256																																																																						
0111	1:128																																																																						
0110	1:64																																																																						
0101	1:32																																																																						
0100	1:16 ✓																																																																						
0011	1:8																																																																						
0010	1:4																																																																						
0001	1:2																																																																						
0000	1:1																																																																						

## Ejercicio 2025-2

- Configuración de registros del Timer0: PPS

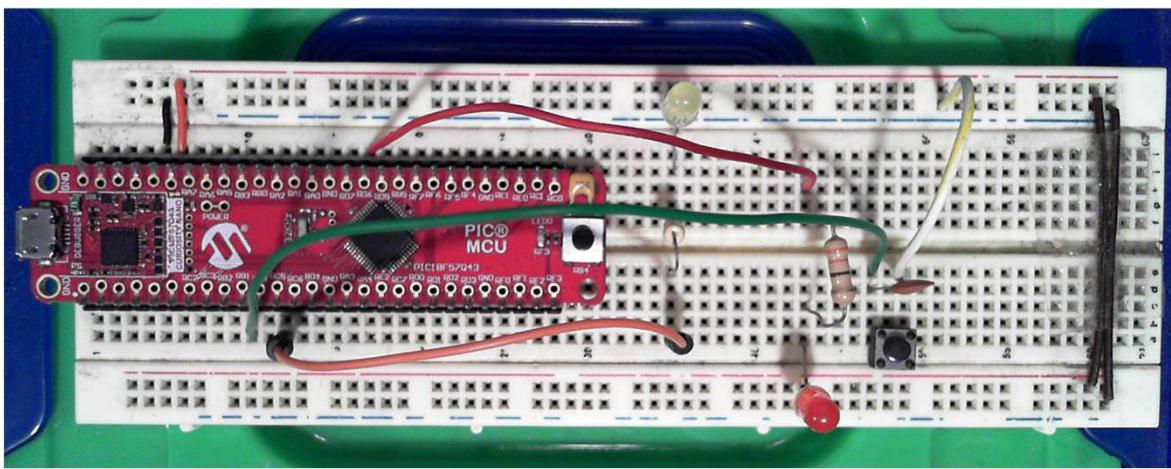
Table 21-2. PPS Output Selection Table

RxyPPS	Output Source	Available Output Ports					
		28-Pin Devices		40-Pin Devices		48-Pin Devices	
0x45	ADGRDB	A	—	C	A	—	F
0x44	ADGRDA	A	—	C	A	—	F
0x43	DSM1	A	—	C	A	—	D
0x42	CLKR	—	B	C	—	B	E
0x41	NCO3	—	B	C	—	E	B
0x40	NCO2	—	B	C	—	D	B
0x3F	NCO1	A	—	C	A	—	D
0x3E - 0x3A	Reserved	—	—	—	—	—	—
0x39	TMRO	—	B	C	—	B	F
0x38	I2C1 SDA <sup>(1)</sup>	—	B	C	—	B	C
0x37	I2C1 SCL <sup>(1)</sup>	—	B	C	—	B	C



## Ejercicio 2025-2

- Circuito implementado:



```

1      PROCESSOR 18F57Q43
2      #include "cabecera.inc"
3
4      PSECT upecenos, class=CODE, reloc=2, abs
5      upecenos:
6          ORG 000000H
7          bra configuro
8
9          ORG 000200H
10         configuro:
11             ;conf modulo de oscilador
12             movlw 0H
13             movlw 60H
14             movwf OSCCON1, 1
15             movlw 02H
16             movwf OSCFRQ, 1
17             movlw 50H
18             movwf OSCEN, 1      ;HFINTOSC & LFINTOSC enabled
19             ;conf del PPS
20             movlb 2H
21             movlw 39H
22             movwf RC5PPS, 1    ;RC5 conectado a TMRO
23             ;conf del TMRO
24             movlb 3H
25             movlw 80H
26             movwf T0CON0, 1     ;TMRO enabled, 8bit mode, posts 1:1
27             movlw 86H
28             movwf T0CON1, 1     ;clksrc = LFINTOSC, sync, presc 1:64
29             movlw 250
30             movwf TMR0H, 1       ;valor de ref de comp 250
31             ;conf E/S
32             movlb 4H
33             bcf TRISC, 5, 1    ;RC5 como salida
34             bcf ANSELC, 5, 1   ;RC5 como digital
35             bcf ANSELB, 1, 1   ;RB1 como digital
36             bsf WPUB, 1, 1     ;RB1 con pullup activado
37             bcf TRISD, 6, 1    ;RD6 como salida
38             bcf ANSELD, 6, 1   ;RD6 como digital

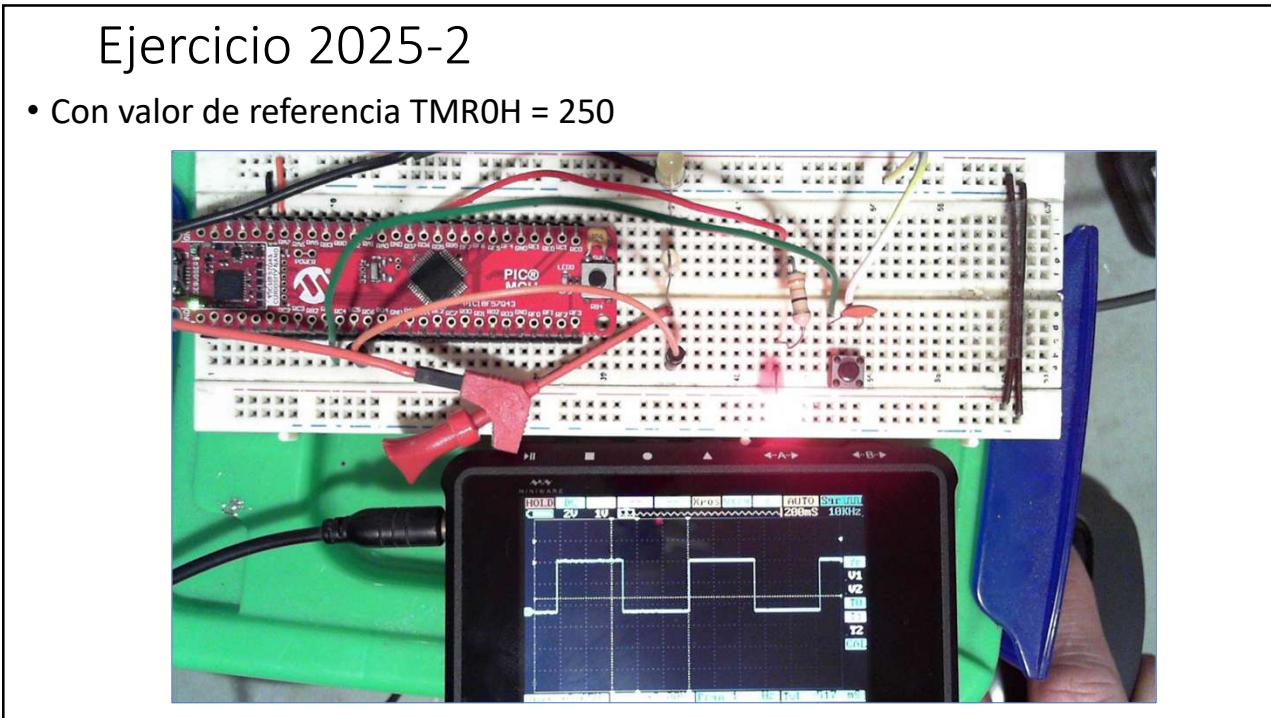
```

## Ejercicio 2025-2

```

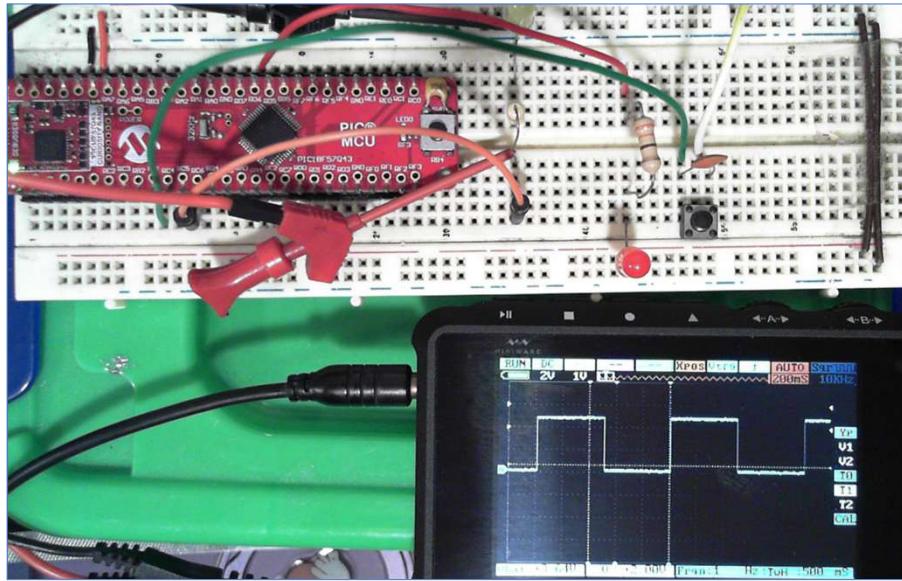
40         inicio:           ;zona de prog de usuario
41             btfsc PORTB, 1, 1
42             bra $-2
43             btg LATD, 6, 1
44             btfs PORTB, 1, 1
45             bra $-2
46             bra inicio
47
48         end upecenos

```



## Ejercicio 2025-2

- Con valor de referencia TMR0H = 242



## Ejercicio 2025-2

- Adicionar un pulsador en RE1 para cambiar el periodo de parpadeo del LED entre 1 segundo y medio segundo.

```

1  PROCESSOR 18F57043
2  #include "cabecera.inc"
3
4  PSECT upecenos, class=CODE, reloc=2, abs
5  upecenos:
6  ORG 000000H
7  bra configuro
8
9  ORG 000200H
10 configuro:
11 ;conf modulo de oscilador
12 movlb 0H
13 movlw 60H
14 movwf OSCCON1, 1
15 movlw 02H
16 movwf OSCFRC, 1
17 movlw 50H
18 movwf OSCEN, 1 ;HFINTOSC & LFINTOSC enabled
19 ;conf del PPS
20 movlb 2H
21 movlw 39H
22 movwf RC5PPS, 1 ;RC5 conectado a TMRO
23 ;conf del TMRO
24 movlb 3H
25 movlw 80H
26 movwf T0CON0, 1 ;TMRO enabled, 8bit mode, posts 1:1
27 movlw 86H
28 movwf T0CON1, 1 ;clksrc = LFINTOSC, sync, presc 1:64
29 movlw 250
30 movwf TMROH, 1 ;valor de ref de comp 250

```

```

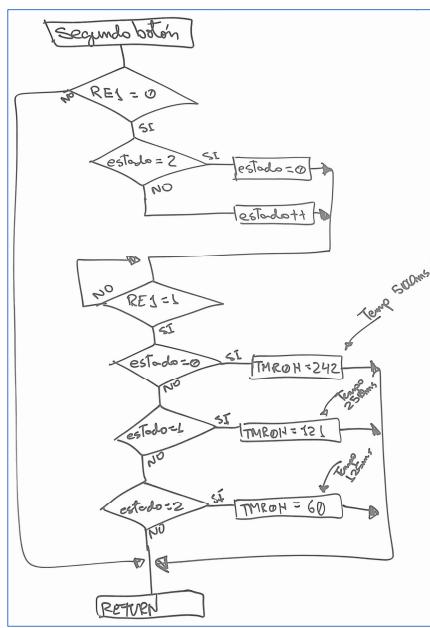
31 ;conf E/S
32 movlb 4H
33 bcf TRISC, 5, 1 ;RC5 como salida
34 bcf ANSELc, 5, 1 ;RC5 como digital
35 bcf ANSELb, 1, 1 ;RB1 como digital
36 bcf WPUB, 1, 1 ;RB1 con pullup activado
37 bcf TRISD, 6, 1 ;RD6 como salida
38 bcf ANSELd, 6, 1 ;RD6 como digital
39 bcf ANSELe, 1, 1 ;RE1 como digital
40 bcf WPUE, 1, 1 ;RE1 con pullup activado
41
42 inicio:           ;zona de prog de usuario
43     btfsc PORTB, 1, 1
44     bra alRE1
45     btg LATD, 6, 1
46     btfss PORTB, 1, 1
47     bra $-2
48 alRE1:
49     btfsc PORTE, 1, 1
50     bra inicio
51     movlb 3H
52     movlw 250
53     cpfseq TMROH, 1
54     bra noes250
55     movlw 62
56     movwf TMROH, 1
57     bra final
58 noes250:
59     movlw 250
60     movwf TMROH, 1
61 final:
62     movlb 4H
63     btfss PORTE, 1, 1
64     bra $-2
65     bra inicio
66
67 end upecenos

```

## Ejercicio 2025-2

- Adicionar un pulsador en RE1 para cambiar el periodo de parpadeo del LED entre 1 segundo, medio segundo y un cuarto de segundo.
- Tener en consideración hacer primero el diagrama de flujo.

## Ejercicio 2025-2

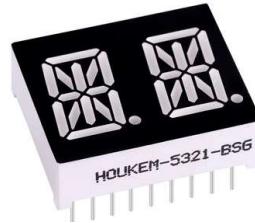


```

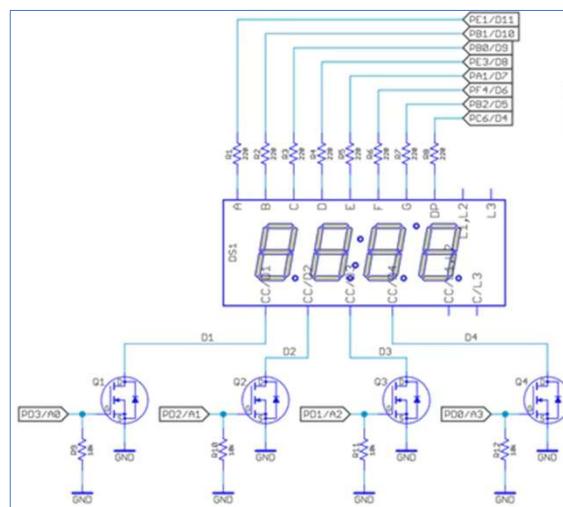
estado EQU 500H
segundo_boton:
    btfsc PORTE, 1, 1
    bra final
    movlb 5H
    movlw 2
    cpfseq estado, 1
    bra suma
    clrf estado, 1
    bra $+4
suma:
    incf estado, 1, 1
    btfss PORTE, 1, 1
    bra $-2
    movlw 0
    cpfseq estado, 1
    bra siguiente1
    movlb 3H
    movlw 242
    movwf TMR0H, 1
    bra final
siguiente1:
    movlw 1
    cpfseq estado, 1
    bra siguiente2
    movlb 3H
    movlw 121
    movwf TMR0H, 1
siguiente2:
    movlw 2
    cpfseq estado, 1
    bra final
    movlb 3H
    movlw 60
    movwf TMR0H, 1
final:
    return
  
```

## Ejercicios propuestos

- Conectar el microcontrolador PIC18F57Q43 con el siguiente dispositivo (display de dos dígitos de 14 segmentos multiplexados de ánodo común):



## Usando MOSFETs



Fin de la sesión