



*Instituto Politécnico Nacional
Escuela Superior de Cómputo*



Introducción a los microcontroladores

TIMER'S

Victor Hugo García Ortega



TIMER – Definición

Los TIMER's son contadores de 8, 16 o 32 bits que se incluyen como un periférico en los microcontroladores. Estos nos sirven para determinar intervalos de tiempo configurados por el usuario. Estos intervalos de tiempo se usan para poder realizar el control de ciertos procesos en el programa dentro del microcontrolador.

TIMER

Dependiendo del modelo que se utilice de microcontrolador se encontrarán una cantidad diferente de TIMER's. Estos TIMER's están designados como TIMER1, TIMER2, ..., etc.

En el DSPIC30F4013 se tienen 5 TIMER's de 16 bits: TIMER1, TIMER2, ... ,TIMER5.



TIMER – Registros

Los registros asociados a cada TIMER son los siguientes:

TMRx: Registro timer contador de 16 bits

PRx: Registro de periodo de 16 bits
asociado con el timer.

TxCON: Registro de control de 16 bits
asociado con el timer.

TIMER – Interrupciones

Cada TIMER tiene asociado bits de control para el manejo de interrupciones:

El bit de habilitación de interrupción (TxIE).

El bit de bandera de estado de la interrupción (TxIF).

Los bits de control de prioridad de la interrupción (TxIP<2:0>).

TIMER – Clasificación

Los TIMER's son clasificados en 3 tipos dependiendo de su funcionalidad:

TIMER tipo A

TIMER tipo B

TIMER tipo C



TIMER tipo A

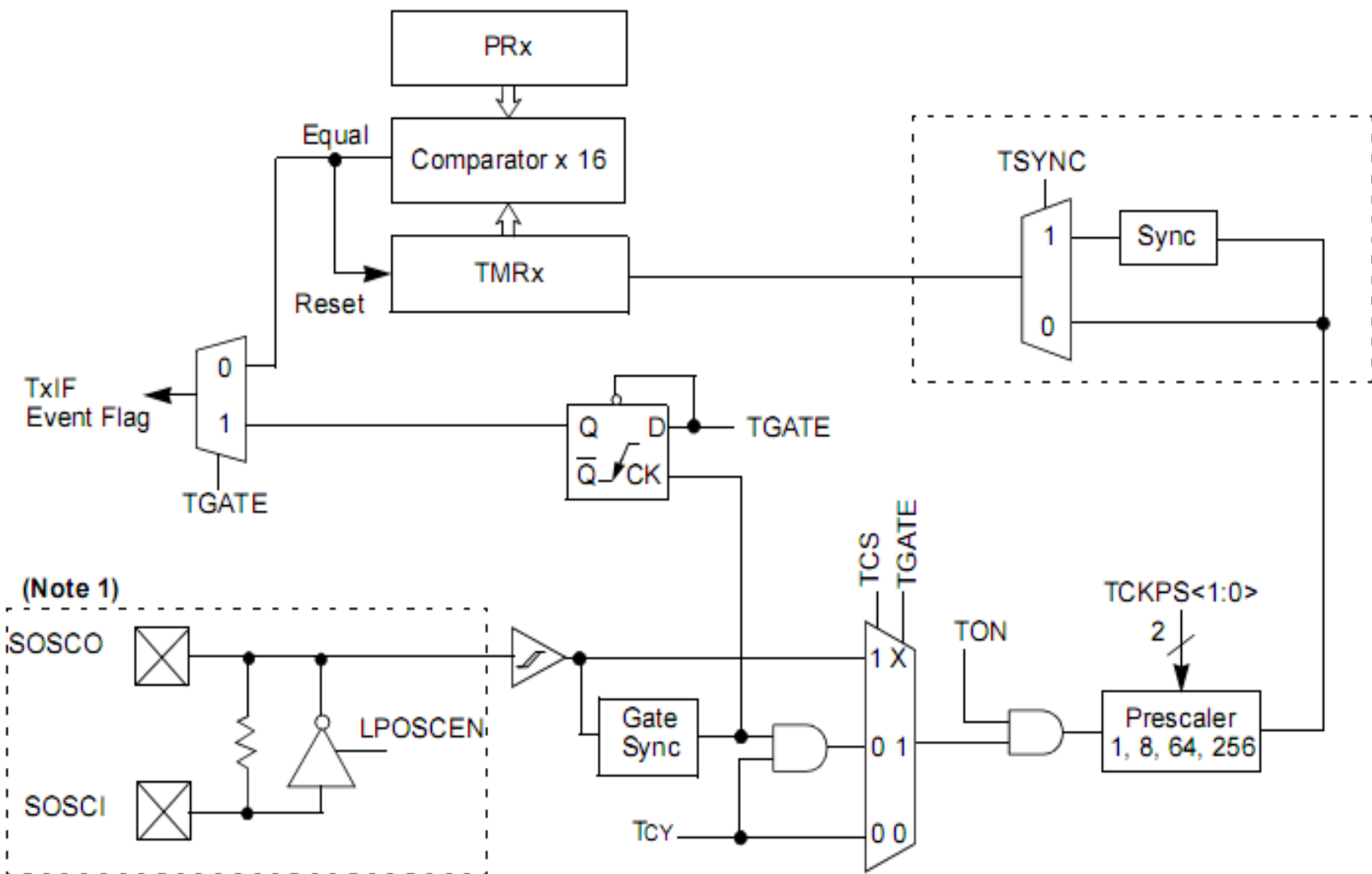
Los TIMER tipo A tienen las siguientes características:

Pueden trabajar con un oscilador externo de 32kHz. Esta característica permite usar al TIMER para configurar relojes de tiempo real (RTC).

Pueden operar en modo asíncrono usando una fuente de reloj externa.

En los DSPIC30F esta disponible al menos un TIMER tipo A y suele ser el TIMER 1.

Figure 12-1: Type A Timer Block Diagram



Note 1: Refer to **Section 7. "Oscillator"** for information on enabling the LP Oscillator.

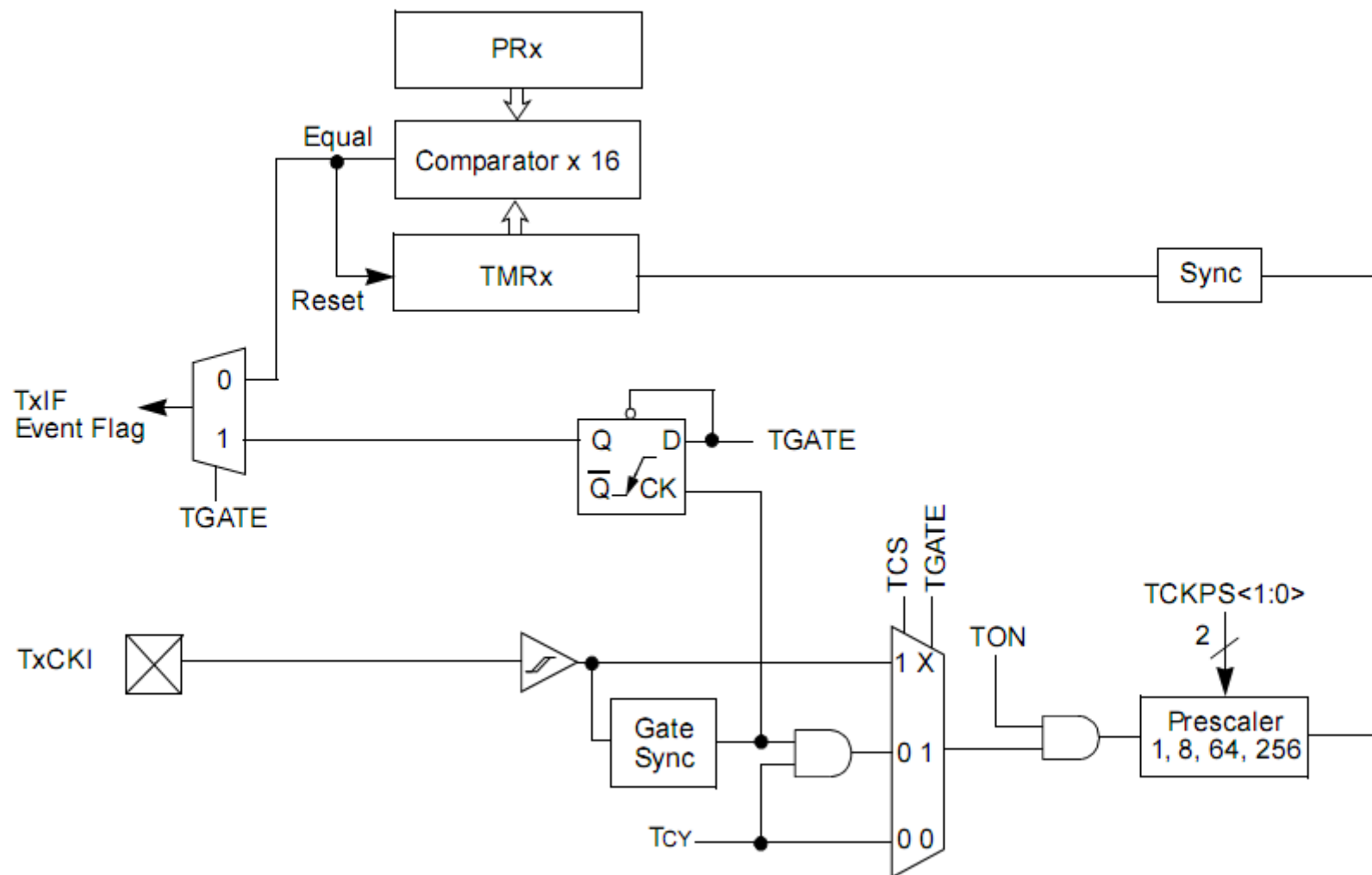
TIMER tipo B

Los TIMER tipo B tienen las siguientes características:

El TIMER tipo B puede ser concatenado con el TIMER tipo C para formar un TIMER de 32 bits. La sincronización del reloj se lleva a cabo después del preescalador.

En la mayoría de los microcontroladores DSPIC30F los TIMER's 2 y 4 (si existen) son tipo B.

Figure 12-2: Type B Timer Block Diagram



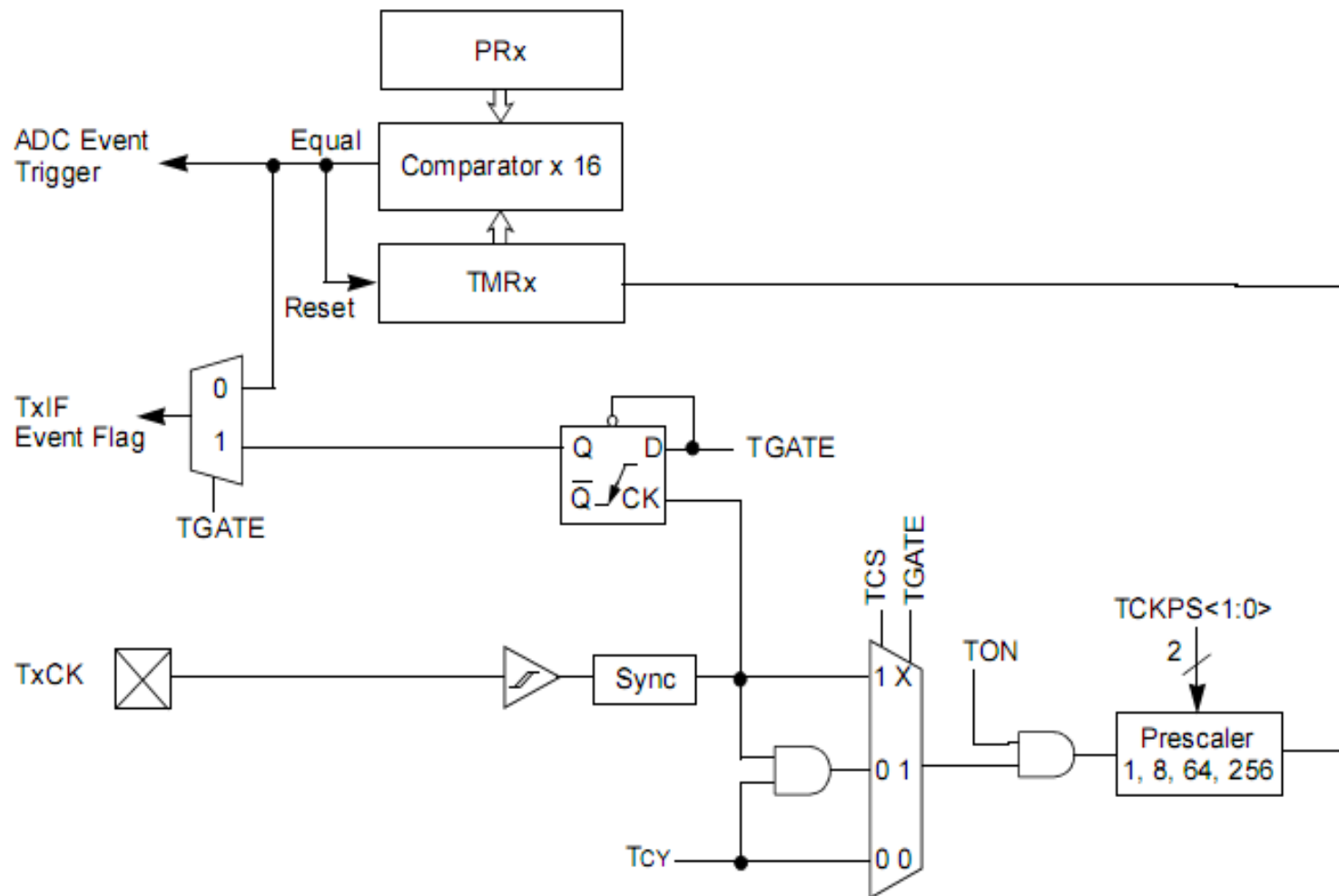
TIMER tipo C

Los TIMER tipo C tienen las siguientes características:

El TIMER tipo C puede ser concatenado con el TIMER tipo B para formar un TIMER de 32 bits. Este TIMER tiene la habilidad de lanzar el inicio de conversión del ADC.

En la mayoría de los microcontroladores DSPIC30F los TIMER's 3 y 5 (si existen) son tipo C.

Figure 12-3: Type C Timer Block Diagram



Note: In certain variants of the dsPIC30F family, the TxCK pin may not be available. Refer to the device data sheet for the I/O pin details. In such cases, the timer must use the system clock ($F_{OSC}/4$) as its input clock, unless it is configured for 32-bit operation.

TIMER – Modos de operación

Cada TIMER puede operar en uno de los siguientes modos:

- Timer síncrono o modo Timer
- Contador síncrono
- Timer acumulador
- Contador asíncrono (Solo en timer tipo A).

Estos modos son determinados por los bits:

TCS (TxCON<1>) Bit de selección de reloj

TSYNC (T1CON<1>) Bit de sincronía (Solo en timer tipo A).

TGATE (TxCON<6>) Bit de acumulación.

*Modo **TIMER***

Todos los tipos de **TIMER** pueden operar en este modo.

El reloj del **TIMER** es proporcionado por el reloj interno ($FCY = FOSC/4$) del **DSPIC**.

Este modo es seleccionado cuando el bit **TCS** = 0.

El bit **TSYNC** no tiene efecto.

Modo Contador Síncrono

El reloj del TIMER es proporcionado de forma externa y el TIMER es incrementado en cada flanco de subida del reloj colocado en el pin TxCK. En el TIMER tipo A se TIENE que habilitar la sincronización externa de reloj. Esto se hace colocando el bit $TSYNC = 1$.

Para los TIMERS tipo B y C la entrada de reloj externa esta siempre sincronizada al reloj interno del DSPIC.

Este modo es seleccionado cuando el bit $TCS = 1$.

*Modo **TIMER Acumulador***

Este modo permite el incremento del **TIMER** en base a la duración del tiempo en alto de un pulso aplicado en el pin **TxCK**. El incremento se realiza hasta que el **TIMER** coincide con el registro de periodo o hasta que el pulso pase a bajo.

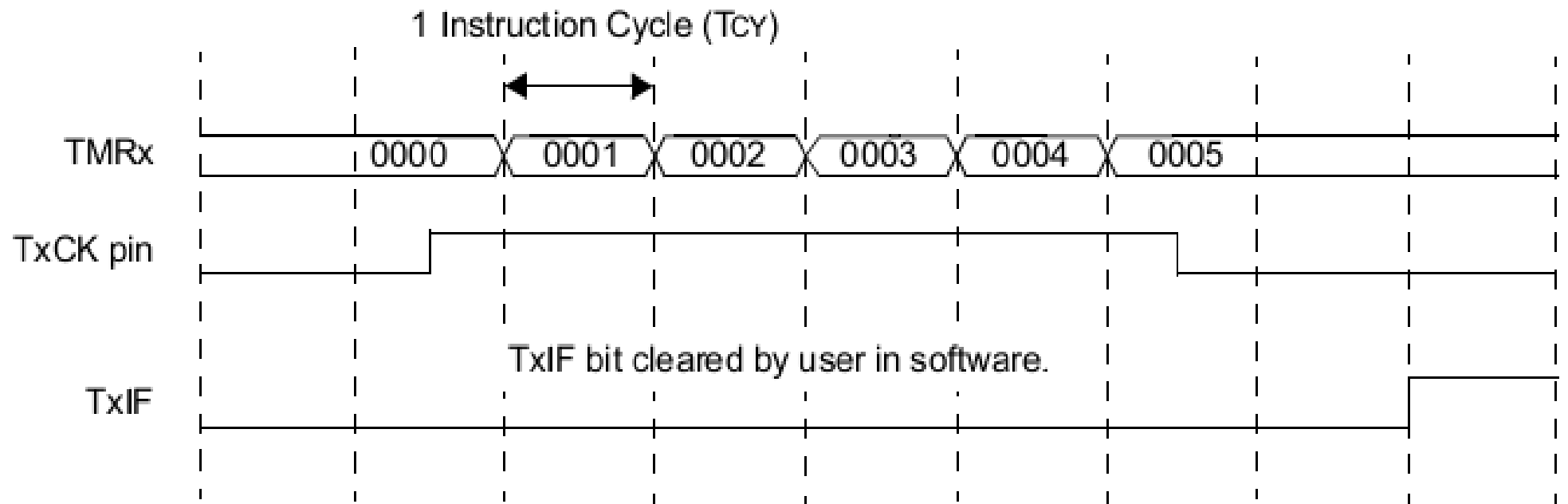
Cuando el pulso pasa a bajo se activa la bandera de interrupción **TxIF**.

El reloj del **TIMER** es proporcionado por el reloj interno ($FCY = FOSC/4$) del **DSPIC**.

En este modo el bit $TCS = 0$, $TGATE = 1$.

Modo *TIMER* Acumulador

El *TIMER* se tiene que limpiar por software.



Modo Contador asíncrono (Solo en TIMER tipo A).

El reloj del TIMER es proporcionado de forma externa y el TIMER es incrementado en cada flanco de subida del reloj colocado en el pin TxCK.

En este modo el bit TSYNC = 0, por lo que el reloj externo no esta sincronizado con el reloj interno del DSPIC.

Modo Contador asíncrono (Solo en **TIMER** tipo A).

Este modo es sirve para:

- Que el **TIMER** pueda operar en modo **SLEEP** y pueda generar interrupciones.
- El reloj pueda ser generado por un oscilador de 32KHz para generar un reloj de tiempo real (RTC).



Uso del Oscilador de 32KHz.

El oscilador es conectado en las terminales SOSCI y SOSCO.

El oscilador de 32KHz es habilitado cuando el bit LPOSCEN = 1 en el registro OSCCON.

Para activar el bit LPOSCEN se debe quitar la “protección contra escrituras accidentales” al registro OSCCON.

Uso del Oscilador de 32KHz.

Para escribir a la parte baja del registro OSCCON (OSCCONL) se tiene que ejecutar la siguiente secuencia de código sin ninguna otra instrucción entre ellas.

```
Byte Write 0x46 to OSCCONL  
Byte Write 0x57 to OSCCONL
```

Después de esta secuencia se permite la escritura de un byte a OSCCONL SOLO POR UN CICLO.

Uso del Oscilador de 32KHz.

Para escribir a la parte alta del registro OSCCON (OSCCONH) se tiene que ejecutar la siguiente secuencia de código sin ninguna otra instrucción entre ellas.

```
Byte Write 0x78 to OSCCONH
```

```
Byte Write 0x9A to OSCCONH
```

Después de esta secuencia se permite la escritura de un byte a OSCCONH SOLO POR UN CICLO.

Uso del Oscilador de 32KHz.

```
mov.b    #0x46,w1           ; follow write sequence ...
mov.b    #0x57,w2           ; for OSCCONL writes.
mov      #OSCCONL,w3
mov.b    w1,[w3]
mov.b    w2,[w3]
bset     OSCCONL,#LPOSCEN    ;enable 32Khz external xtal
```

Contacto

Gracias por su atención...

e-mail:

vgarciaortega@yahoo.com.mx

