

# Microcontroladores de 32 bits

Dr. Casimiro Gómez González  
Facultad de Electrónica, UPAEP  
correo: [casimiro.gomez@upaep.mx](mailto:casimiro.gomez@upaep.mx)  
Tel: 222 229 9428

Primavera 2013



# Prólogo

El presente material ha sido desarrollado como parte del *Diplomado en Sistemas embebidos para Automatización y Robótica* y es una recopilación de distintos materiales que se han ido desarrollando en el Laboratorio de Sistemas Embebidos (LSE) de la Universidad Popular Autónoma del Estado de Puebla (UPAEP) en México.

El autor  
Casimiro Gómez González  
Doctor en Ingeniería Mecatrónica



# Índice general

<b>Prólogo</b>	<b>III</b>
<b>1. Introducción a Microprocesadores</b>	<b>1</b>
1.1. Descripción general . . . . .	1
1.2. Conexiones Mínimas . . . . .	3
1.3. Capacitores de desacoplo . . . . .	3
1.3.1. Capacitores de Ruido de Fuente . . . . .	5
1.3.2. Capacitor del Regulador Interno de Voltaje ( $V_{CAP}$ ) . . . . .	5
1.3.3. Pin <i>Master Clear</i> $\overline{MCLR}$ . . . . .	6
1.3.4. Pins sin utilizar . . . . .	6
1.4. El oscilador . . . . .	6
1.4.1. Generación de el reloj de sistema (SYSCLK) . . . . .	7



# Índice de cuadros





# Índice de figuras

1.1. Familias Microchip . . . . .	1
1.2. Pines del Microcontrolador 32MX220F032B . . . . .	2
1.3. Conexión del PicKit3 para programar el Microcontrolador . . . . .	2
1.4. Conexión Interna del PicKit3 para programar el Microcontrolador . . . . .	2
1.5. Circuito para quemar PICs con el PicKit3 . . . . .	3
1.6. Conexiones Mínimas del 32MX220F032B . . . . .	4
1.7. Capacitores de desacoplo . . . . .	4
1.8. Algoritmo para configurar Puertos . . . . .	5
1.9. Conexión del reset al microcontrolador . . . . .	6
1.10. Opciones del oscilador . . . . .	8
1.11. Opciones del oscilador Primario . . . . .	8
1.12. Conexiones del cristal . . . . .	8



# Capítulo 1

## Introducción a Microprocesadores

En este capítulo se describen las principales características de los microcontroladores de 32 bits.

### 1.1. Descripción general

La Familia de microcontroladores de 32 bits de Microchip (PIC32) cuenta con un funcionamiento más amplio y con una memoria de mayor capacidad, lo cual nos permite realizar diseños embebidos complejos. Para emplear dichos micros, Microchip ofrece una plataforma de software gratuito conocido como MPLABX IDE, el cual usamos a lo largo del curso.

Para programar el PIC32MX220F032B utilizamos el PICKit 3, para ello armamos un circuito para que el micro pueda ser reconocido por el PICKit3 y posteriormente grabarlo. El PICKit viene con un manual que muestra un diagrama de la forma de conectar el microprocesador. El pin uno de los PICKit está situado en el triángulo blanco.

El PICKit tiene seis orificios en su extremo, cinco de ellos se encuentran conectados a cinco pines correspondientes del microprocesador. Sin embargo nos dimos cuenta que al conectar el PICKit 3 no identificaba el microprocesador, pues aparecía un error en MPLABX.IDE a la hora de quemar el micro. Por ello decidimos adicionarle dos capacitores de  $0.1\mu\text{f}$  entre VDD y VSS para ayudar a estabilizar al sistema colocándolos muy cerca del PIC, y así purificar la señal.

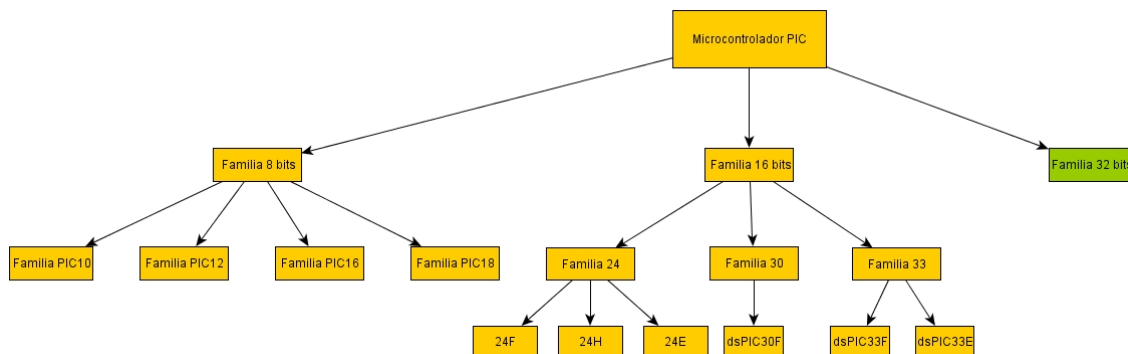


Figura 1.1: Familias Microchip

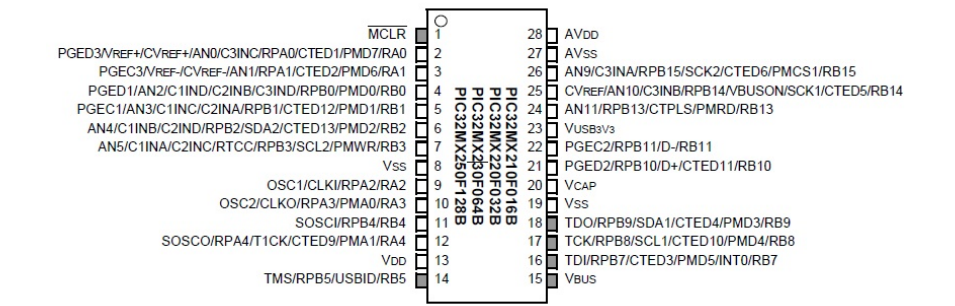


Figura 1.2: Pines del Microcontrolador 32MX220F032B

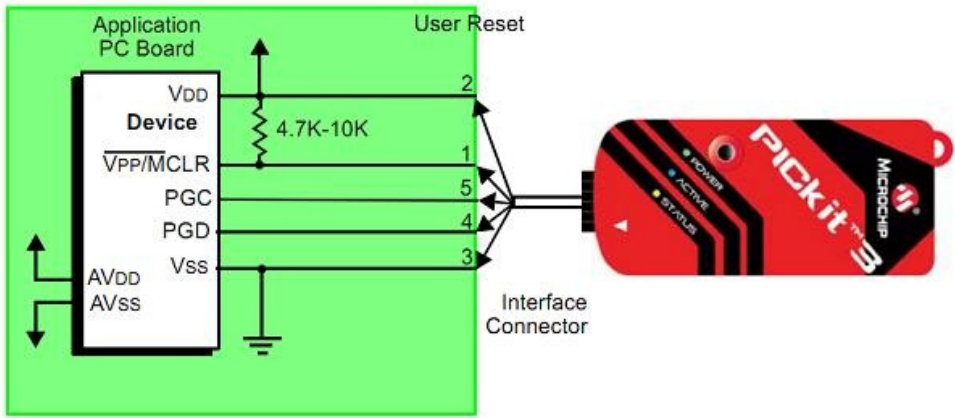


Figura 1.3: Conexión del PicKit3 para programar el Microcontrolador

Target Connector Pinout

Pin	Signal
1	MCLR/Vpp
2	Vpp Target
3	Vss Ground
4	PGD (ICSPDAT)
5	PGC (ICSPCLK)
6	Do not connect*

\*Reserved for future use.

PICKIT 3 Connector Pinout

Pin	Signal
1	MCLR/Vpp
2	Vpp Target
3	Vss Ground
4	PGD (ICSPDAT)
5	PGC (ICSPCLK)
6	Do not connect*

\*Reserved for future use.

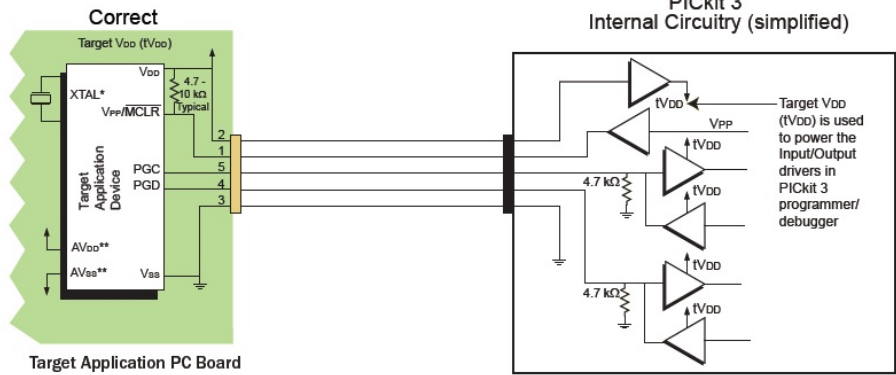


Figura 1.4: Conexión Interna del PicKit3 para programar el Microcontrolador

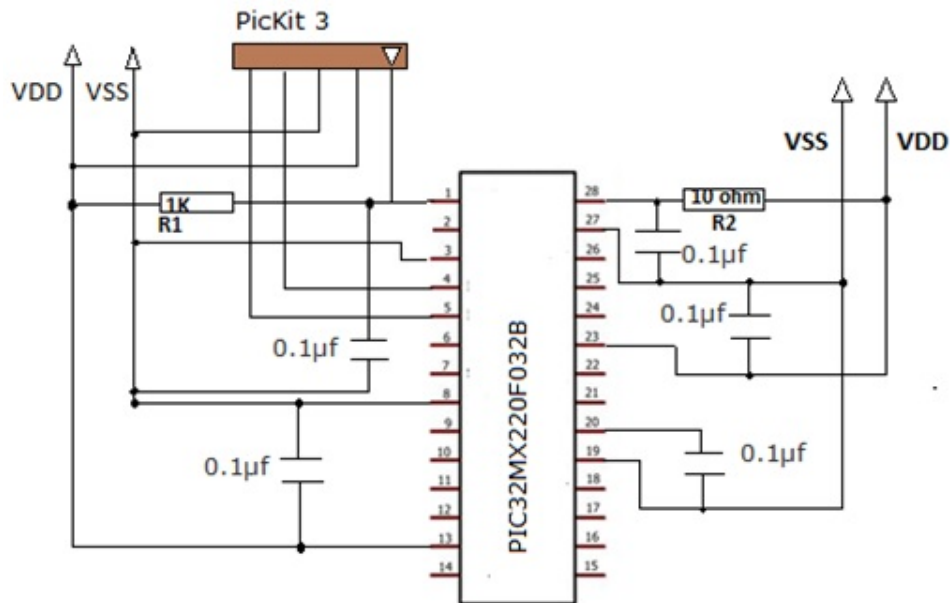


Figura 1.5: Circuito para quemar PICs con el PicKit3

## 1.2. Conexiones Mínimas

Para utilizar el microcontrolador PIC32MX220F032B se requiere tener en cuenta las conexiones mínimas. A continuación se describe una lista de PINES, los cuales SIEMPRE deben estar conectados:

- Todos los pines  $V_{dd}$  y  $V_{ss}$
- Todos los pines  $AV_{ss}$  y  $AV_{dd}$ , aún y cuando el módulo del Convertidor Analógico Digital (CAD) no sea utilizado
- EL pin  $V_{CAP}$
- EL pin  $\overline{MCLR}$
- Los pines OSC1 y OSC2 cuando se utiliza un oscilador externo.

## 1.3. Capacitores de desacoplo

Para la correcta operación del Microcontrolador es necesario conectar capacitores de desacoplo en los pines de potencia, tales como,  $V_{DD}$ ,  $V_{SS}$ ,  $AV_{DD}$  y  $AV_{SS}$  como se muestra

Cuando se utilizan capacitores de desacoplo deben considerarse los siguientes criterios:

- Valor y tipo de capacitor: Un valor de  $0,1\mu F$ , 10 – 20V es recomendado. Se recomienda utilizar capacitores cerámicos

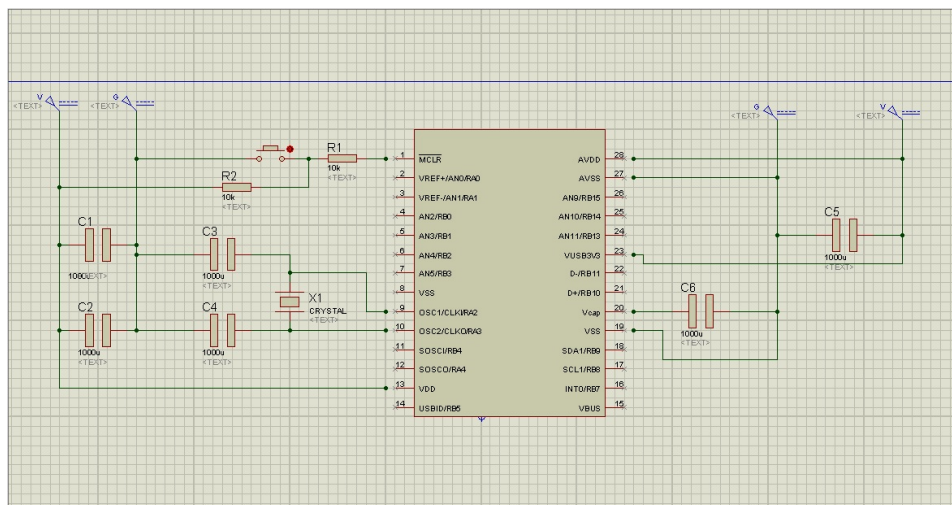


Figura 1.6: Conexiones Mínimas del 32MX220F032B

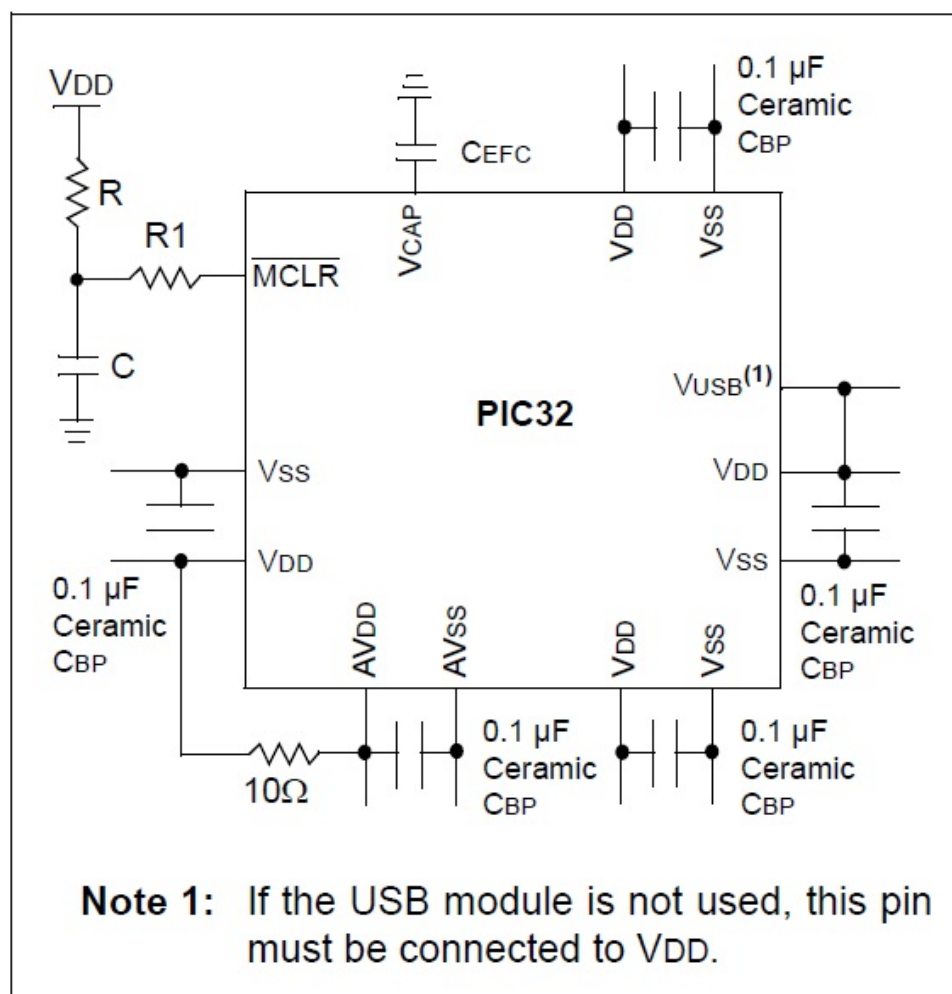


Figura 1.7: Capacitores de desacoplo

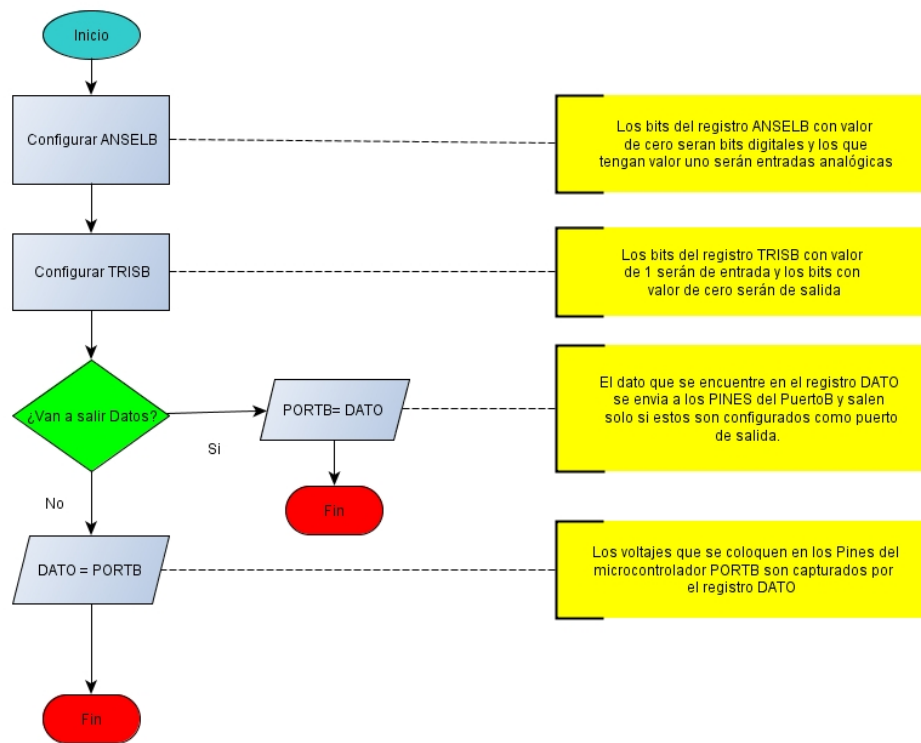


Figura 1.8: Algoritmo para configurar Puertos

- Colocación en el PCB: Los capacitores de desacoplo deben colocarse tan cerca de los PINES del microcontrolador como sea posible. Se recomienda colocar los capacitores del mismo lado de la tarjeta de la que se encuentra el Microcontrolador. Si el espacio es limitado, el capacitor puede colocarse en otra capa del PCB usando una via, sin embargo, debemos asegurarnos que la distancia del pin al capacitor este en los  $6mm$  de largo.
- Manejando ruidos de Alta Frecuencia: Si la tarjeta experimenta ruido de alta frecuencia, arriba de  $10MHz$ , se tiene que adicionar un segundo capacitor ceramico en paralelo a los descritos en los capacitores de desacoplo. El valor de el segundo capacitor puede estar en el rango de  $0,01\mu F$  a  $0,001\mu F$ . Se tiene que colar este segundo capacitor cerca del capacitor primario de desacoplo. Para el diseño de circuitos de alta frecuencia, se debe considerar instalar un par de capacitores tan cerca de los pines de potencia con diferencia de decadas, por ejemplo  $0,1\mu F$  en paralelo con  $0,001\mu F$ .

### 1.3.1. Capacitores de Ruido de Fuente

El uso de capacitores en la fuente de alimentación es recomendable para mejorar la estabilidad de la alimentación. Valores típicos en el rango de  $4,7\mu F$  a  $47\mu F$  son recomendables. Este capacitor debe estar localizado lo mas cerca posible del dispositivo.

### 1.3.2. Capacitor del Regulador Interno de Voltaje ( $V_{CAP}$ )

Un capacitor de bajo-ESR es necesario en el pin  $V_{CAP}$ , el cual es utilizado para estabilizar la salida el regulador interno. El pin  $V_{CAP}$  no debe ser conectado a  $V_{DD}$ , y debe tener una capacitor

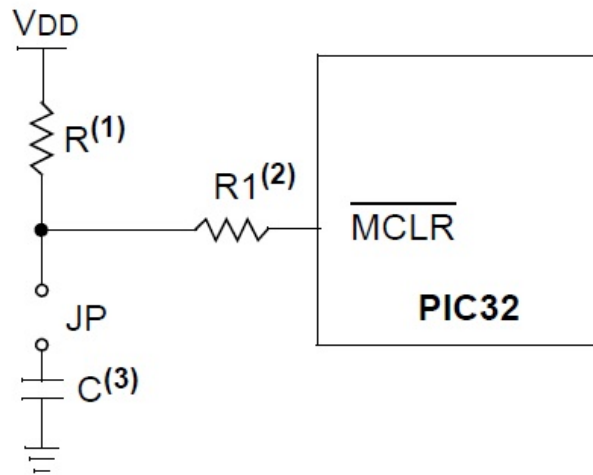


Figura 1.9: Conexión del reset al microcontrolador

$C_{EFC}$ , con al menos 6V de voltaje, conectado a tierra. El tipo de capacitor es de cerámica o tantalio, con un valor en el rango de  $8\mu F$  a  $10\mu F$

### 1.3.3. Pin Master Clear $\overline{MCLR}$

El Pin  $\overline{MCLR}$  tiene dos funciones en el microcontrolador:

- Reset del Microcontrolador
- Programación y depuración del microcontrolador

Conectando el Pin  $\overline{MCLR}$  a tierra resetea el Microcontrolador.

En la Figura 1.9, la resistencia  $R1 \leq 470\Omega$  limitará cualquier corriente fluyendo a través del Pin  $\overline{MCLR}$  desde el capacitor externo debido a descargas electrostáticas (ESD) o sobre estres eléctrico (EOS), asegurando que el pin  $\overline{MCLR}$  alcanza las especificaciones de  $V_{IH}$  y  $V_{IL}$ . La resistencia  $R \leq 10k\Omega$  es recomendada. Se recomiendan valores iniciales de  $10k\Omega$  para asegurar que el pin  $\overline{MCLR}$  alcanza los valores  $V_{IH}$  y  $V_{IL}$  son alcanzados. El capacitor debe ser de tal tamaño que prevenga Resets accidentalmente debido a glitches o extender el reset del microcontrolador durante el periodo de  $POR$ .

### 1.3.4. Pins sin utilizar

Pins de I/O que esten sin se utilizados no deben de dejarse flotantes. Estos pueden ser configurados como salida y manejados a un valor lógico bajo.

Alternativamente, las entradas pueden reservarse conectando el pin a  $V_{ss}$  a través de una resistencia de  $1k$  a  $10k$  y configurando el pin como entrada.

## 1.4. El oscilador

El sistema de oscilación del PIC32 tiene los siguientes módulos y características:



- Cuatro osciladores externos e internos como fuentes de reloj
- Un PLL con un divisor y multiplicador, lo cual incrementa la frecuencia de operación ya sea de la fuente interna o externa
- Un divisor postescalador integrado que selecciona la fuente de la cual proviene
- Un interruptor seleccionable por software que elige entre varias fuentes.
- Un monitor de operación segura del reloj (FSCM) que detecta las fallas del reloj y permite a las aplicaciones recuperarse o apagarse.

La familia de dispositivos PIC32 tiene múltiples relojes internos que se derivan de las fuentes de reloj internas o externas. Algunas de estas fuentes de reloj tienen lazos de PLL (PLL), un divisor de salida programable, un divisor de entrada, para escalar de entrada de frecuencia y adaptarla a la aplicación. La fuente de reloj puede ser cambiada en ejecución por software. El registro de control del oscilador está cerrado por hardware, y debe ser abierto por una serie de escrituras antes de que el reloj realice un cambio.

Hay tres relojes principales en los dispositivos PIC32:

- Reloj del Sistema (SYSCLK), usado en el CPU y algunos periféricos
- Reloj del bus de periféricos (PBCLK) usado por la mayoría de periféricos
- Reloj USB (USBCLK), usado por los periféricos USB

El reloj del PIC32 surge de alguna de las siguientes fuentes:

- Oscilador Primario ( $P_{OSC}$ ) en los pines OSC1 y OSC2
- El oscilador secundario ( $S_{OSC}$ ) en los pines SOSC1 y SOSC2
- El oscilador rápido interno (FRC)
- El oscilador de baja potencia (LPRC)

Cada una de las fuentes de reloj tiene opciones de configuración únicas, tales como el PLL, el divisor de entrada y/o divisor de salida.

#### 1.4.1. Generación de el reloj de sistema (SYSCLK)

La señal SYSCLK es usado principalmente por el CPU y algunos periféricos tales como el DMA, el controlador de interrupciones y el cache de preFetch. La señal SYSCLK proviene de alguna de las cuatro fuentes de reloj:

- $P_{OSC}$
- $S_{OSC}$
- El oscilador interno rápido FRC
- El oscilador LPRC

Oscillator Mode	Oscillator Source	POSCMOD<1:0>	FNOSC<2:0>
FRC Oscillator with Postscaler (FRCDIV)	Internal	xx	111
FRC Oscillator divided by 16 (FRCDIV16)	Internal	xx	110
LPRC Oscillator	Internal	xx	101
Sosc (Timer1/RTCC)	Secondary	xx	100
Posc in HS mode with PLL Module (HSPLL)	Primary	10	011
Posc in XT mode with PLL Module (XTPLL)	Primary	01	011
Posc in EC mode with PLL Module (ECPLL)	Primary	00	011
Posc in HS mode	Primary	10	010
Posc in XT mode	Primary	01	010
Posc in EC mode	Primary	00	010
Internal FRC Oscillator with PLL Module (FRCPLL)	Internal	10	001
Internal FRC Oscillator	Internal	xx	000

Figura 1.10: Opciones del oscilador

Oscillator Mode	Description
HS	High-speed crystal
XT	Resonator, crystal or resonator
EC	External clock input
HSPLL	Crystal, PLL enabled
XTPLL	Crystal resonator, PLL enabled
ECPLL	External clock input, PLL enabled

Figura 1.11: Opciones del oscilador Primario

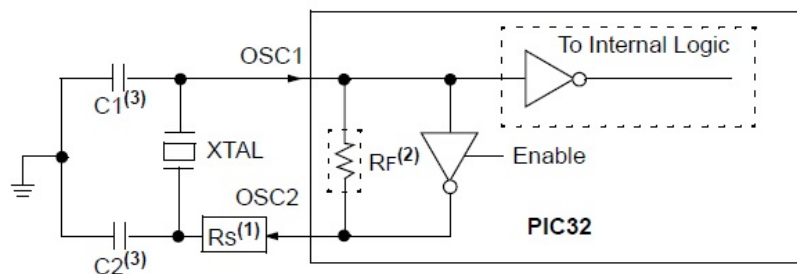


Figura 1.12: Conexiones del cristal

Algunas de las fuentes de reloj tienen opciones específicas de multiplicador y/o divisor. La fuente SYSCCLK es seleccionada por la configuración del microcontrolador y puede cambiarse durante la operación por software. La habilidad para cambiar las fuentes de reloj durante la operación permite a la aplicación reducir su consumo de potencia y reducir la velocidad de reloj.

