

CAPITULO 1

Microcontroladores PIC[®] de 8 Pines

Tabla de Contenidos

HARDWARE

TIP #1:	Dual Speed RC Oscillator	1-2
TIP #2:	Input/Output Multiplexing.....	1-2
TIP #3:	Read Three States From One Pin....	1-3
TIP #4:	Reading DIP Switches.....	1-3
TIP #5:	Scanning Many Keys With One Input.....	1-4
TIP #6:	Scanning Many Keys and Wake-up From Sleep.....	1-4
TIP #7:	8x8 Keyboard with 1 Input.....	1-5
TIP #8:	One Pin Power/Data.....	1-5
TIP #9:	Decode Keys and ID Settings	1-6
TIP #10:	Generating High Voltages	1-6
TIP #11:	V _{DD} Self Starting Circuit.....	1-7
TIP #12:	Using PIC [®] MCU A/D For Smart Current Limiter.....	1-7
TIP #13:	Reading A Sensor With Higher Accuracy.....	1-8
TIP #13.1:	Reading A Sensor With Higher Accuracy – RC Timing Method.....	1-8
TIP #13.2:	Reading A Sensor With Higher Accuracy – Charge Balancing Method	1-10
TIP #13.3:	Reading A Sensor With Higher Accuracy – A/D Method.....	1-11
TIP #14:	Delta Sigma Converter	1-11

SOFTWARE

TIP #15:	Delay Techniques	1-12
TIP #16:	Optimizing Destinations.....	1-13
TIP #17:	Conditional Bit Set/Clear	1-13
TIP #18:	Swap File Register with W	1-14
TIP #19:	Bit Shifting Using Carry Bit.....	1-14

INTRODUCCIÓN

Microchip continúa presentando sus productos innovadores que son más pequeños, rápidos, fáciles de usar y confiables. Los "8-pin Flash PIC[®] microcontrollers (MCU)" son usados en un amplio rango de productos cotidianos, desde cepillos de dientes y secadores de pelo, hasta productos industriales y de medicina.

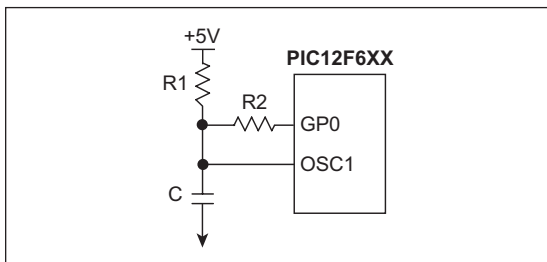
El PIC12F629/675 MCU reúne todas las ventajas de la arquitectura del PIC[®] MCU y la flexibilidad de la memoria Flash dentro de un integrado de 8 pins. Provee las características y la inteligencia que antes no estaban disponibles por costos y limitaciones de espacio. Las características incluyen un set de instrucciones de 14-bit, encapsulado pequeño, amplio rango de operación desde 2.0 hasta 5.5 volts, oscilador interno programable de 4Mhz, memoria EEPROM on-board, referencias de voltaje on-chip y hasta 4 canales de 10 bits A/D. La flexibilidad de la memoria Flash y las excelentes herramientas de desarrollo, que incluyen "low-cost In-Circuit Debugger", "In-Circuit Serial Programming™" y "MPLAB[®] ICE 2000 emulation", hacen que estos dispositivos sean ideales para cualquier aplicación de control embebida.

HARDWARE

La siguiente serie de de "Tips 'n Tricks" puede ser aplicada a una variedad de aplicaciones que ayudan a construir las aplicaciones más dinámicas.

TIP #1 Dual Speed RC Oscillator

Figura 1-1



1. Luego de un reset, el pin I/O está en lta (Z)
 2. Output '1' en el pin I/O
 3. R1, R2 y C determinan la frecuencia de OSC
 4. También funciona con capacitores adicionales
- La frecuencia del PIC® MCU en "RC oscillator mode" depende de la resistencia y capacitancia en el pin OSC1. La resistencia cambia con el voltaje de salida en GP0. GP0 en '1' pone R2 en paralelo con R1 reduciendo la resistencia en OSC 1 y aumentando la frecuencia. GP0 como entrada aumenta la resistencia en OSC1 minimizando la corriente en R2, disminuyendo la frecuencia y el consumo de potencia.

Resumen:

GP0 = Input: Baja velocidad -> corrientes pequeñas
 GP0 = Output : Alta velocidad -> procesamiento rápido

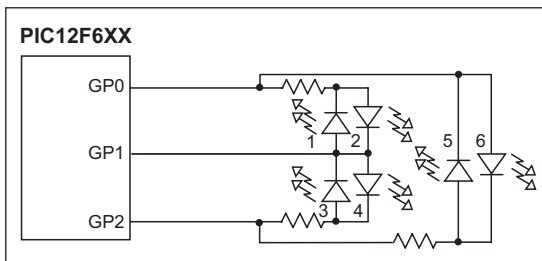
TIP #2 Input/Output Multiplexing

Un diodo o una combinación de diodos puede ser habilitada alternando los I/Os entre high y low, o estableciendolos como "inputs" (Z). El número de diodos que se pueden controlar depende del numero de I/Os (GP) utilizados. Se cumple que: $D = GP \times (GP - 1)$.

Ejemplo 2-1: 6 LEDs en 3 Pines I/O

GPx	LEDs
0 1 2	1 2 3 4 5 6
0 0 0	0 0 0 0 0 0
0 1 Z	1 0 0 0 0 0
1 0 Z	0 1 0 0 0 0
Z 0 1	0 0 1 0 0 0
Z 1 0	0 0 0 1 0 0
0 Z 1	0 0 0 0 1 0
1 Z 0	0 0 0 0 0 1
0 0 1	0 0 1 0 1 0
0 1 0	1 0 0 1 0 0
0 1 1	1 0 0 0 1 0
1 0 0	0 1 0 0 0 1
1 0 1	0 1 1 0 0 0
1 1 0	0 0 0 1 0 1
1 1 1	0 0 0 0 0 0

Figura 2-1



TIP #3 Read Three States From One Pin

Para chequear estado Z:

- Lleve el pin "Output" a "high"
- Setee como "Input"
- Leer 1
- Lleve el pin "Output" a "low"
- Setee como "Input"
- Leer 0

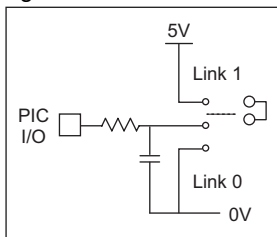
Para chequear estado 0:

- Leer 0 en el pin

Para chequear estado 1:

- Leer 1 en el pin

Figure 3-1



State	Link 0	Link 1
0	closed	open
1	open	closed
NC	open	open

El jumper tiene 3 posibles estados:

no conectado, Link 1 y Link 0. El capacitor se carga y descarga dependiendo del voltaje de I/O permitiendo el estado "no conectado".

El software debe chequear el estado NC primero llevando el I/O a H, leyendo el 1 y llevando el I/O a L y leyendo el 0. Los estados "Link 1" y "Link 0" son leídos directamente.

TIP #4 Reading DIP Switches

Ejemplo 4-1

El input del timer

puede ser usado para testear que switch(s) está cerrado. El input del Timer1 contiene una resistencia pull-up. Secuencialmente, cada switch I/O es seteado como I y se chequea si hubo un incremento en el Timer1 que indica que el switch está cerrado.

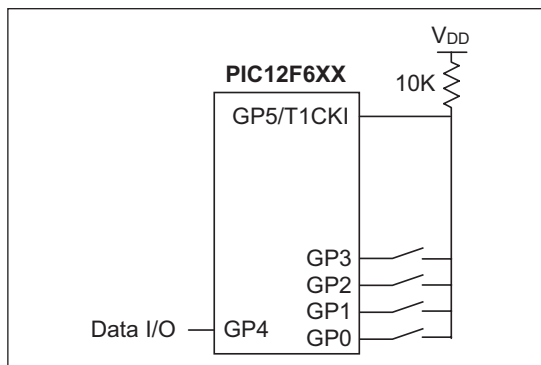
```

movlw    b'11111111'
movwf    TRISIO
DIP
movlw    b'00000111'
movwf    T1CON
movlw    b'11111110'
movwf    Mask
clrf     GPIO

LOOP
clrf     TMR1L
movf     Mask,W
movwf    TRISIO
btfsc   TMR1L,0
andwf   DIP,F
bsf     STATUS,C
rlf     Mask,F
btfsc   Mask,4
goto    Loop
retlw   0
    
```

Cada bit del registro del DP representa la posición del switch correspondiente. Estableciendo el Timer1 en FFFFh y habilitando su interrupción, un incremento causará un "rollover" y generará una interrupción. Esto simplifica el software ya que se elimina el testeo del bit en el registro TMR1L. Secuencialmente establezca cada GPIO en "Input" y verifique si hubo un incremento en el TMR1 (o 0 si se usa el pin I/O estándar)

Figura 4-1



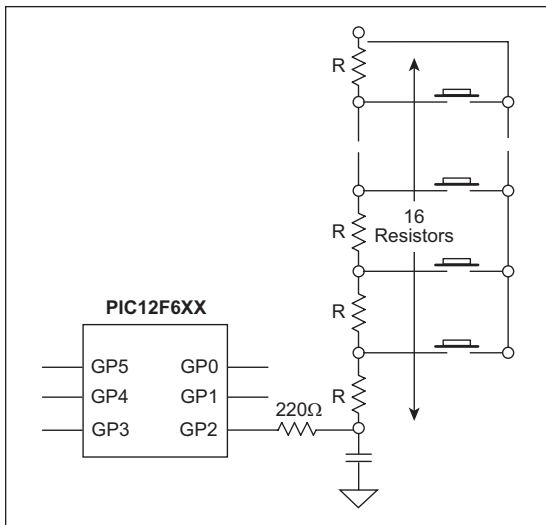
TIP #5 Scanning Many Keys With One Input

El tiempo requerido para cargar un capacitor depende de la resistencia entre Vdd y el capacitor. Al apretar un pulsador, Vdd se conecta a un punto diferente del conjunto de resistencias haciendo que la resistencia entre y el capacitor disminuya, lo cual reduce el tiempo de carga del capacitor. Un timer es usado con un comparador o con una entrada digital para medir el tiempo de carga del capacitor. Este tiempo es utilizado para determinar qué pulsador fue apretado.

Secuencia del software:

1. Configurar GP2 que entregue una baja tensión para descargar el capacitor mediante la R de I/O
2. Configurar GP2 como un comparador de entrada y CVREF.
3. Utilizar un timer para medir el tiempo. Si este tiempo medido es mayor que el máximo permitido entonces se debe repetir; sino se puede determinar cual de los pulsadores fue presionado. Cuando se aprieta un pulsador, el divisor resistivo provoca un cambio en el valor de RC.

Figura 5-1



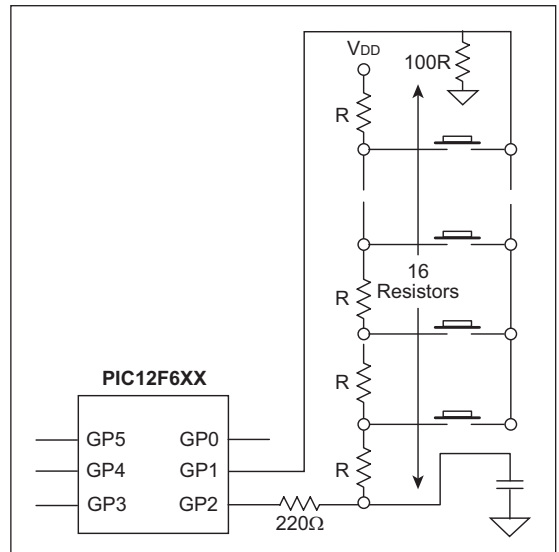
Ver AN512, "Implementing Ohmmeter/ Temperature Sensor".

TIP #6 Scanning Many Keys and Wake-up From Sleep

Se puede agregar un I/O adicional para despertar el sistema cuando se presiona un pulsador. Antes de entrar en modo "sleep", configurar GP1 como entrada habilitando el "interrupt on change" y GP2 como salida "high". La resistencia de pull down mantiene GP1 "low" hasta que se aprieta un pulsador. Así, GP1 es llevado a high via GP2 y Vdd generando la interrupción. Luego, GP2 se configura como salida "low" para descargar el capacitor por medio de la R=220Ω. GP1 se configura como salida "high" y GP2 como entrada para medir el tiempo de carga del capacitor.

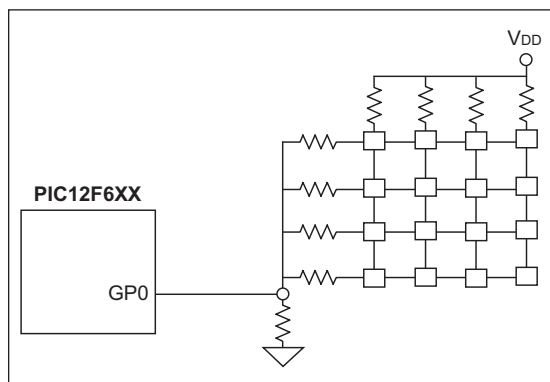
- GP1 pin conectado al común de los pulsadores.
- Habilitar el "wake-up on port change"
- Establecer GP1-> entrada y GP2 "high" antes del "Sleep"
- Al presionar un pulsador el PIC® MCU despierta, GP2 debe llevarse a low para descargar el capacitor.
- Establecer GP1 high con el "wake-up" para detectar pulsador

Figura 6-1

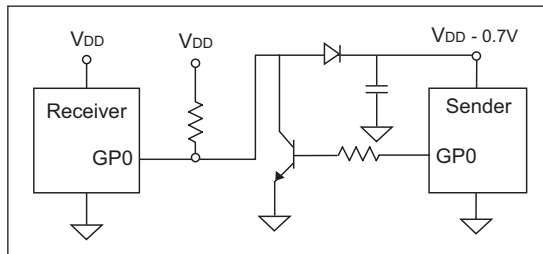


TIP #7 4x4 Keyboard with 1 Input

Seleccionando cuidadosamente el valor de las resistencias, cada pulsador genera un único voltaje. Este voltaje puede ser medido con el A/D para determinar qué pulsador fue. Resistencias de mayor precisión deben ser utilizadas para maximizar la unidad. El A/D mide alrededor de 0 cuando no se presiona ningún pulsador.

Figura 7-1**TIP #8 One Pin Power/Data**

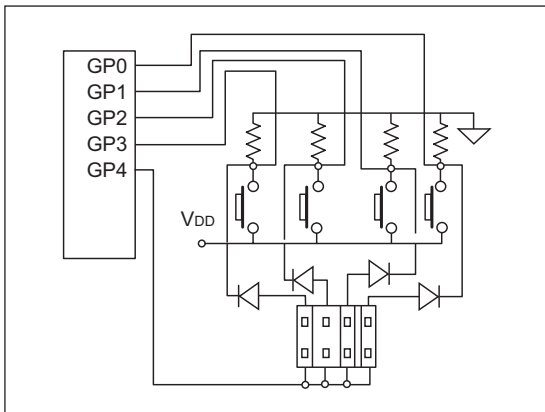
Un único I/O puede ser usado para datos unidireccionales y como una fuente para otro microcontrolador. El I/O se mantiene en "high" por medio de la R de "pull-up" conectada a Vdd. El emisor utiliza un transistor "pull-down" para llevar la línea de datos a "low" o inhabilita al transistor que permita que el "pull-up" pase a "high" para enviar datos al receptor. El Vdd se suministra al emisor por medio de la línea de datos. El capacitor estabiliza el Vdd del emisor y el diodo previene la descarga del capacitor mediante la línea de I/O cuando esta en low. El Vdd del emisor es una caída de diodo menor que el del receptor.

Figura 8-1

TIP #9 Decode Keys and ID Settings

Pulsadores y jumpers pueden compartir I/Os usando otro I/O para seleccionar cual va a ser leído. Ambos comparten una resistencia de "pull-down". Por esta razón, se leerá un '0' hasta se apriete un pulsador o un jumper sea conectado. Cada input (GP3/2/1/0) comparte un jumper y un pulsador. Para leer la configuración del jumper hay que establecer GP4 como "high" y cada jumper conectado sera leído como un '1' en su correspondiente I/O o '0' si no está conectado. Con GP4 en L, un pulsador apretado será leído como '1', o se leerá un '0' de lo contrario.

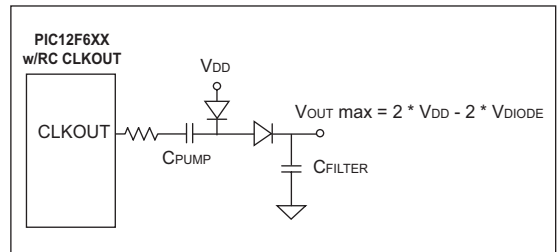
Figura 9-1



- Con GP4 = 1 y ningún pulsador apretado, lee configuración ID.
- Con GP4 = 0, lee pulsadores.

TIP #10 Generating High Voltages

Figura 10-1



Voltajes mayores que Vdd pueden ser generadas alternando I/O. El pin CLKOUT/OSC2 del PIC® MCU alterna a un cuarto de la frecuencia de OSC1 cuando se utiliza el modo "external RC oscillator". Cuando OSC2 esta en L, el diodo de Vdd está en directa y conduce corriente cargando Cpump. Después que OSC2 esté en "high", el otro diodo comienza a conducir y carga el Cfilter. Esto resulta en una carga igual al doble de Vdd menos dos caídas de diodo. Esto puede ser usado con PWM, alternando I/O o alternando otro pin.

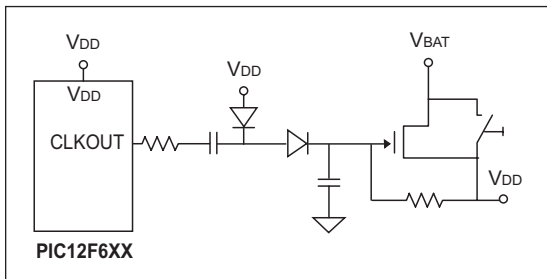
TIP #11 V_{DD} Self Starting Circuit

Teniendo en cuenta el tip anterior, la misma "bomba de carga" puede ser usada por el MCU para suministrarse su propio V_{DD}. Antes que el "switch" sea presionado, V_{BAT} tiene tensión y los puntos V_{DD} no tienen tensión. Cuando se presiona el pulsador, V_{DD} tiene tensión y el MCUs CLKOUT (en modo "external RC oscillator") empieza a alternar. El voltaje generado por la "bomba" enciende el FET permitiendo a V_{DD} permanecer con tensión. Para apagar el MCU, hay que ejecutar la instrucción "Sleep" permitiendo al MCU desconectarse de su fuente via software.

Ventajas:

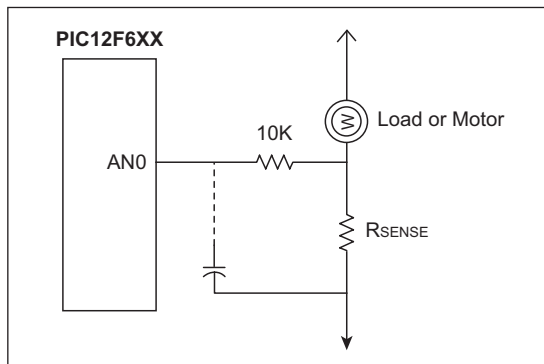
- PIC® MCU "leakage current" aproximadamente 0
- Bajo costo (usa n-channel FET)
- Confiable
- No requiere pins I/O adicionales

Figura 11-1



TIP #12 Using PIC® MCU A/D For Smart Current Limiter

Figura 12-1



- Detectar la corriente que pasa por R_{sense}
 - Capacitor "peak filter" opcional
 - La respuesta a niveles variados de sobrecorriente puede realizarse en el software
- Agregando una resistencia (R_{SENSE}) en serie con un motor, el A/D puede ser usado para medir la "in rush current", proveer la corriente limitante, recuperación ante sobrecorriente o funcionar como un fusible inteligente. La resistencia de 10K limita la corriente en el canal analógico y no excede el limite de la impedancia de la fuente del A/D.

TIP #13 Reading a Sensor With Higher Accuracy

Los sensores puede ser leídos directamente con el A/D pero algunas aplicaciones, factores como la temperatura, la precisión externa del componente, la no linealidad del sensor y/o caídas de tensión en las baterías deben ser consideradas. En otras aplicaciones se necesitan más de 10 bits de precisión y se aceptan sensados más lentos. El siguiente tip tiene en cuenta estos factores y muestra como optimizar al máximo el PIC® MCU.

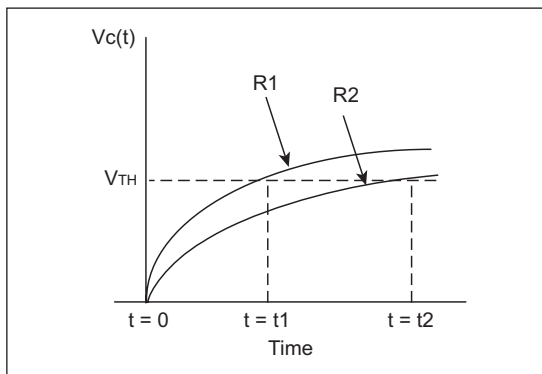
- 13.1. Método “RC Timing” (con R de referencia)
- 13.2. Método “Charge Balancing”
- 13.3. Método “A/D”

Tip #13.1 Reading a Sensor With Higher Accuracy – RC Timing Method

Método “RC Timing”:

Simple RC step response
 $V_C(t) = V_{DD} * (1 - e^{-t/(RC)})$
 $t = -RC \ln(1 - V_{TH}/V_{DD})$
 V_{TH}/V_{DD} es cte
 $R2 = (t2/t1) * R1$

Figura 13-1



Una R de referencia puede ser usada para mejorar la precisión de la lectura del sensor analógico. En el diagrama, el tiempo de carga de la combinación RC se mide usando un timer y un puerto como entrada o un comparador que varía entre '0' y '1'. La curva R1 usa la R de referencia y la curva R2 usa el sensor. El tiempo de carga de la curva R1 es sabida y puede ser usada para calibrar la lectura desconocida del sensor, R2. Esto reduce el efecto de la temperatura, tolerancias de los componentes y el ruido.

Notas de aplicación:

AN512, "Implementing Ohmmeter/Temperature Sensor"

AN611, "Resistance and Capacitance Meter Using a PIC16C622"

Se presenta el esquemático y el flujo del software usando una R de referencia para mejorar la precisión de una lectura de un sensor analógico. La RREF y la RSEN son asignados a un I/O y tienen un capacitor en común. GP0 es usado para descargar el capacitor y representa la tensión del capacitor.

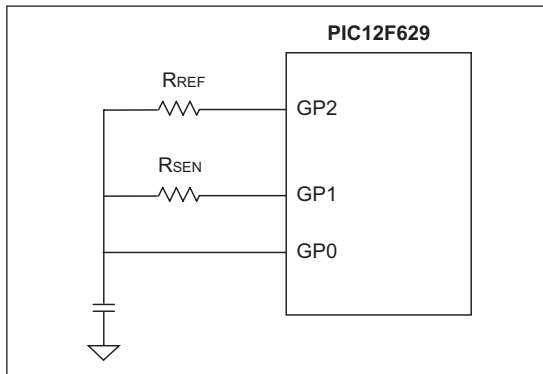
Mediante el software, se usa un timer para medir cuando GP0 cambia de '0' a '1' para las mediciones del sensor y las de referencia.

Cualquier diferencia presente entre la medida de referencia y la calibrada es utilizada para ajustar el sensor logrando así una medición más precisa aún.

El comparador y la referencia del comparador en el PIC12F629/675 pueden ser usados en vez de un pin de un puerto para una medición más precisa. Capacitores de polypropyleno son muy estables y apropiados para estos casos.

1. Establecer GP1 y GP2 como entrada, y GP0 como "low output" para descargar C.
2. Establecer GP0 -> entrada y GP1-> salida "high"
3. Medir tRSEN (GP0 cambia a 1)
4. Repetir paso 1
5. Establecer GP0 -> entrada y GP2 -> salida "high"
6. Medir tRREF (GP0 cambia a 1)
7. Usar capacitores de polypropyleno
8. $R_{TH} = x R_{REF} t_{RSEN}/t_{RREF}$

Figura 13-2



Otra alternativa: comparador de voltaje en el PIC12F6XX para medir la tensión del capacitor en GP0.

Tip #13.2 Reading a Sensor With High Accuracy – Charge Balancing

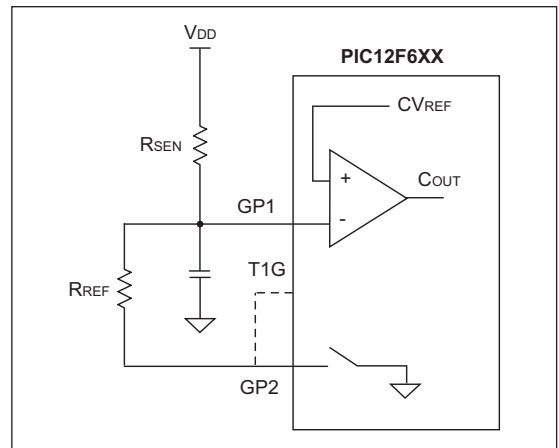
Method

1. El sensor carga un capacitor
 2. La R de referencia descarga el capacitor
 3. Modular la R de referencia para mantener constante el promedio de carga del capacitor
 4. Usar comparador para determinar la modulación
- Para mejorar la precisión más allá de 10 o 12 bits se puede usar la técnica de “Charge Balancing”. Se basa en que el MCU mantenga un voltaje constante en un capacitor ya sea dejándolo cargar por un sensor o dejándolo descargar por una R de referencia. Se usa un timer para muestrear el voltaje del capacitor durante intervalos regulares hasta una determinada cantidad de muestras. Contando el número de veces que el voltaje del capacitor está por encima de un umbral arbitrario se determina el voltaje del sensor.

El comparador y el voltaje de referencia del comparador (CV REF) en el PIC12F629/675 son ideales para esta aplicación.

1. GP1 voltaje promedio = CVREF
2. Base de tiempo como velocidad de muestreo
3. Al finalizar cada período de la base de tiempo:
 - Si GP1 > CVREF, entonces GP2 “Output Low”
 - Si GP1 < CVREF, entonces GP2 “Input mode”
4. Acumular la cantidad “GP2 lows” en varias muestras
5. El número de muestras determina la precisión
6. La cantidad de “GP2 lows” determina el “duty cycle” de RREF

Figura 13-3



Tip #13.3 Reading a Sensor With Higher Accuracy – A/D Method

Los NTC (Negative Temperature Coefficient) son sensores con respuesta no lineal a cambios de temperatura. A medida que la temperatura disminuye, la variación de la resistencia disminuye. Estos sensores tienen un rango de uso limitado ya que la precisión se hace menor que la del A/D a medida que cae la temperatura. Cambiando el divisor de tensión del Rsen, el rango de temperaturas puede ser ampliado.

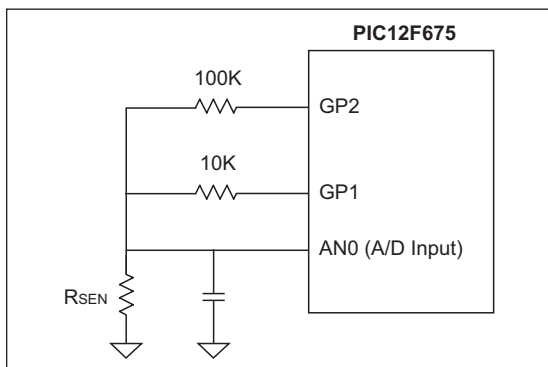
Para seleccionar la temperatura máxima, GP1 entrega un '1' y GP2 configurado como entrada. Para el mínimo, GP2 entrega un '1' y GP1 es configurado como entrada. El mínimo del rango va a incrementar el cambio del sensor de voltaje a medida que la temperatura caiga, permitiendo así usar un rango más amplio.

Resumen:

Rango mayor: GP1 output '1' y GP2 input
 Rango menor: GP1 input y GP2 output '1'

1. Resistencias de 10K y 100K son usadas para determinar el rango
2. VREF para A/D = VDD
3. El cálculo de Rth es independiente de VDD
4. $Count = RSEN / (RSEN + RREF) \times 255$
5. No olvidarse de permitir el "acquisition time" para el A/D

Figura 13-4

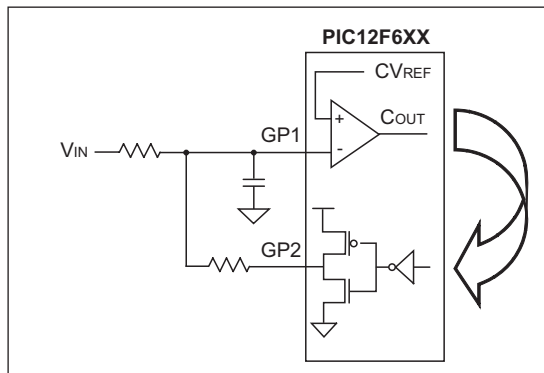


TIP #14 Delta-Sigma Converter

La carga del capacitor en GP1 se mantiene casi igual al CVREF por medio del monitoreo del COUT por parte del MCU y alternando GP2 entre "input mode" y "output low" adecuadamente. Se usa un timer para muestrear el bit COUT periódicamente. Cada vez que GP2 pasa a "low", se incrementa un contador. El valor del contador corresponde al voltaje de entrada.

Para minimizar los efectos de las tolerancias, temperaturas, etc., se puede calibrar el circuito. Aplicar un voltaje conocido a la entrada y permitir al microcontrolador contar muestras hasta llegar al resultado esperado. Tomando la misma cantidad de muestras para mediciones posteriores hace que sean mediciones calibradas.

Figura 14-1



1. GP1 voltaje promedio = CVREF
2. Base de tiempo como velocidad de muestreo
3. Al finalizar cada periodo de la base de tiempo:
 - Si GP1 > CVREF, entonces GP2 "Output Low"
 - Si GP1 < CVREF, entonces GP2 "Output High"
4. Acumular la cantidad "GP2 lows" en varias muestras
5. El número de muestras determina la precisión

TIPS ‘N TRICKS CON SOFTWARE

Para reducir costos, los diseñadores deben aprovechar al máximo la memoria disponible en el MCU. La memoria programable es una de las causas más importantes en el costo del MCU. La optimización del código le evita tener que comprar más memoria que la necesaria. Aquí se presentan ideas para reducir el código.

TIP #15 Delay Techniques

- Use GOTO “next instruction” en vez de dos NOPs.
- Use CALL Rtrn como quad, 1 instrucción NOP (donde “Rtrn” es la etiqueta de salida de la subrutina existente).

Ejemplo 15-1

NOP NOP	;2 instructions, 2 cycles
GOTO \$+1	;1 instruction, 2 cycles
CALL Rtrn . . Rtrn RETURN	;1 instruction, 4 cycles

Los MCU son usados como interfaz con el mundo externo por medio de bus de datos, LEDs, pulsadores, etc. Como el MCU corre a una frecuencia fija, se van a necesitar rutinas de retrasos para cumplir con los tiempos de setup de otros dispositivos, pausas para un “handshake” o bien reducir la tasa de datos de un bus compartido.

Instrucciones como DECFSZ e INCFSZ son adecuadas para retrasos prolongados, en donde una variable es incrementada o decrementada hasta llegar a cero, en donde se ejecuta un condicional. Para retrasos menores de unos pocos ciclos se presentan las siguientes ideas. Para un delay de dos ciclos, es común usar dos instrucciones NOP que usan dos posiciones memoria. El mismo resultado se puede obtener usando “goto \$+1”. El “\$” representa el valor actual del contador en MPASM™ Assembler. Cuando se encuentra esta instrucción, el MCU salta a la próxima posición de memoria. Esto hubiera hecho si se hubieran usado dos NOPs pero como la instrucción GOTO utiliza dos ciclos para ejecutarse, un retraso de dos ciclos fue creado. Este retraso de dos ciclos solamente ocupa una posición de memoria del programa. Para crear un retraso de 4 ciclos, agregue una etiqueta a una instrucción RETURN existente. En este ejemplo, la etiqueta “Rtrn” fue agregada al RETURN de la subrutina existente dentro del código. Cuando se ejecuta “CALL Rtrn”, el MCU tarda dos ciclos de instrucciones para ejecutar el CALL y dos más para ejecutar el RETURN. En vez de utilizar cuatro instrucciones NOP para crear el retardo de cuatro ciclos, el mismo resultado se obtiene agregando una sola instrucción CALL.

TIP #16 Optimizing Destinations

- Observe el movimiento de los datos y reestructura

Ejemplo 16-1

Example: $A + B \rightarrow A$			
MOVWF	A, W	MOVWF	B, W
ADDWF	B, W	ADDWF	A, F
MOVWF	A		
3 instructions		2 instructions	

Un uso cuidadoso de los bits de destino en instrucciones pueden ahorrar memoria. Aquí, el registro A se suma al B y el resultado es guardado en A. Una opción de destino está disponible para operaciones lógicas y aritméticas. En el primer ejemplo, el resultado de la instrucción ADDWF se guarda en el "working register". El MOVWF se usa para mover el resultado desde el registro de trabajo hacia el registro A. En el segundo ejemplo, la instrucción ADDWF usa el bit de destino para guardar el resultado en el registro A ahorrando así una instrucción.

TIP #17 Conditional Bit Set/Clear

- Para mover un solo bit de dato desde REGA hacia REGB
- Condición previa REGB bit
- Testear REGA bit y arreglar REGB si es necesario

Ejemplo 17-1

BTFSS	REGA, 2	BCF	REGB, 5
BCF	REGB, 5	BTFSC	REGA, 2
BTFSC	REGA, 2	BSF	REGB, 5
BSF	REGB, 5		
4 instructions		3 instructions	

Una técnica para mover un bit desde el registro REGA hacia REGB es haciendo "bit tests". En el primer ejemplo, el bit en REGA se prueba usando la instrucción BTFSS. Si el bit está en "clear" se ejecuta la instrucción BCF y limpia el bit REGB, y si el bit está en "set", se saltea la instrucción. El segundo testeo de bit determina si el bit está en "set", y de ser así, va a ejecutar el BSF y setear el bit REGB, de lo contrario se saltea la instrucción. Esta secuencia requiere de cuatro instrucciones. Una técnica más eficiente es suponer como "clear" el bit en REGA, y limpiar el bit REGB, y testear si el bit de REGA está en "clear". De ser así, la suposición fue correcta y se saltea la instrucción BSF, de lo contrario el bit REGB es seteado. La secuencia en el segundo ejemplo usa tres instrucciones porque un testeo de bit no fue necesario.

Un punto importante, es que el segundo ejemplo va a crear un "glitch" si el REGB es un puerto en "high". Esto es causado por las instrucciones BCF y BTFSC que van a ser ejecutadas sin importar el valor del bit en REGA.

TIP #18 Swap File Register with W

Ejemplo 18-1

```

SWAPWF    MACRO    REG
           XORWF   REG, F
           XORWF   REG, W
           XORWF   REG, F
           ENDM
    
```

El siguiente macro intercambia los contenidos de W y REG sin usar un segundo registro.

Necesita:

0 TEMP registers

3 Instructions

3 TCY

Una manera eficiente de intercambiar los contenidos de un registro con el “working register” es usando tres instrucciones XORWF. No requiere de registros temporarios. Aquí se presenta un ejemplo:

W	REG	Instruction
10101100	01011100	XORWF REG,F
10101100	11110000	XORWF REG,W
01011100	11110000	XORWF REG,F
01011100	10101100	Result

TIP #19 Bit Shifting Using Carry Bit

Rotar un byte por medio del “carry” sin usar RAM RAM variable para el loop:

- Adaptado fácilmente para rutinas de transmisiones de interface en serie.
- El “carry” bit se limpia (excepto en el ultimo ciclo) y ciclo se repite hasta el “zero bit” esté seteado indicando el final.

Ejemplo 19-1

```

LIST P=PIC12f629
INCLUDE P12f629.INC
buffer    equ    0x20

bsf       STATUS,C      ;Set 'end of loop' flag
rlf       buffer,f      ;Place first bit into C
bcf       GPIO,Dout     ;precondition output
btfsc    STATUS,C       ;Check data 0 or 1 ?
bsf       GPIO,Dout
bcf       STATUS,C      ;Clear data in C
rlf       buffer,f      ;Place next bit into C
movf     buffer,f       ;Force Z bit
btfss    STATUS,Z       ;Exit?
goto     Send_Loop
    
```

CAPITULO 2

Control de potencia con MCU PIC®

Soluciones

Tabla de contenidos

HARDWARE

TIP #1	Switching Off External Circuits/ Duty Cycle	2-2
TIP #2	Power Budgeting	2-3
TIP #3	WDT Alternative Wake-ups	2-4
TIP #4	Stretched Dog	2-4
TIP #5	Low Power Timer1 Oscillator.....	2-4
TIP #6	Ultra Low-Power Wake-Up	2-4
TIP #7	Low Energy Power Supplies	2-5
TIP #8	Low Power Timer1.....	2-5

SOFTWARE

TIP #9	Configuring Port Pins	2-6
TIP #10	I/O Initialization.....	2-7
TIP #11	Two-Speed Start-Up.....	2-7
TIP #12	How to Use a Comparator Reference as a D/A.....	2-8
TIP #13	How to Detect a Loss of Crystal/Resonator Oscillator.....	2-8
TIP #14	Enabling Idle Modes.....	2-8
TIP #15	How to Eliminate an External Crystal, Resonator or RC Timing Network	2-8

HARDWARE y SOFTWARE

TIP #16	Clock Switching PIC16F Dual Clock ..	2-9
TIP #17	Calibration	2-9

INTRODUCTION

Microchip continúa presentando sus productos innovadores que son más pequeños, rápidos, fáciles de usar y confiables. Los "8-pin Flash microcontrollers (MCU)" son usados en un amplio rango de productos cotidianos, desde cepillos de dientes y secadores de pelo, hasta productos industriales y de medicina. La familia de "PIC16F/18F Power Managed featuring nanoWatt Technology" reúne todas las ventajas de la arquitectura del PIC® MCU y la flexibilidad de la memoria Flash con una serie de características nuevas en cuanto a la alimentación. Estos dispositivos se convierten en una solución para sistemas inteligentes o complejos que requieren una extensa vida útil de la batería y su uso eficiente.

La flexibilidad de la memoria Flashy las excelentes herramientas de desarrollo, que incluyen "low-cost In-Circuit Debugger", "In-Circuit SerialProgramming™" y "MPLAB® ICE 2000 emulation", hacen que estos dispositivos sean ideales para cualquier aplicación de control. La siguiente serie de Tips 'n Tricks pueden ser aplicados a una variedad de aplicaciones que ayudan a obtener lo máximo del "PIC16F/18F Power Managed family featuring nanoWatt Technology".

HARDWARE

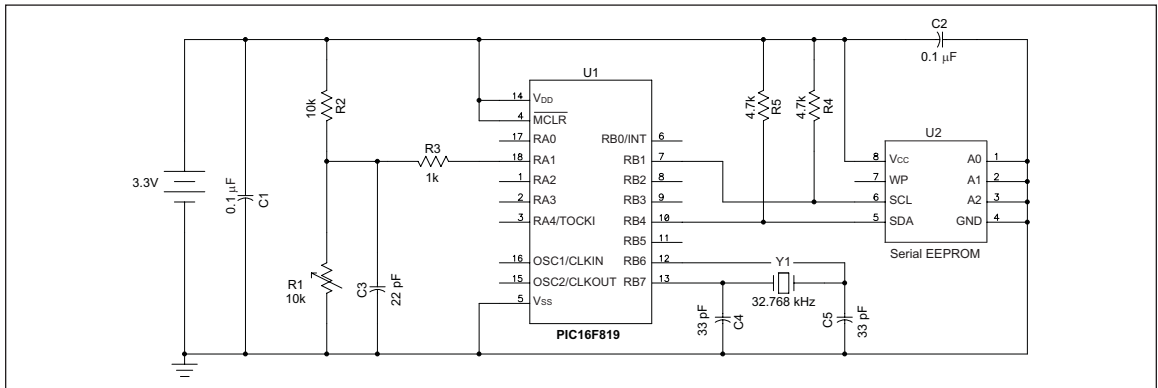
suministrado evita el uso de componentes externo reduciendo el costo total. Aquí hay algunos tips que ayudan a explotar al máximo la familia de nanoWatt.

TIP #1 Switching Off External Circuits/Duty Cycle

Todos los “low power modes” en el mundo no ayudarían a su aplicación si no es capaz de controlar la energía usada en los circuitos externos al microprocesador. Encender un LED es equivalente a correr la mayoría de los PIC® MCUs a 5V-20 MHz. Cuando planea su circuito, decida qué modos físicos o estados son requeridos y particione la electrónica para poder apagar los circuitos innecesarios.

Ejemplo:
La aplicación es un “data recorder” de larga duración. Tiene un sensor, una EEPROM, una batería y un microprocesador. Cada dos segundos, debe llevar a cabo una lectura del sensor, adaptar el dato, guardarlo en la EEPROM y esperar a la siguiente lectura.

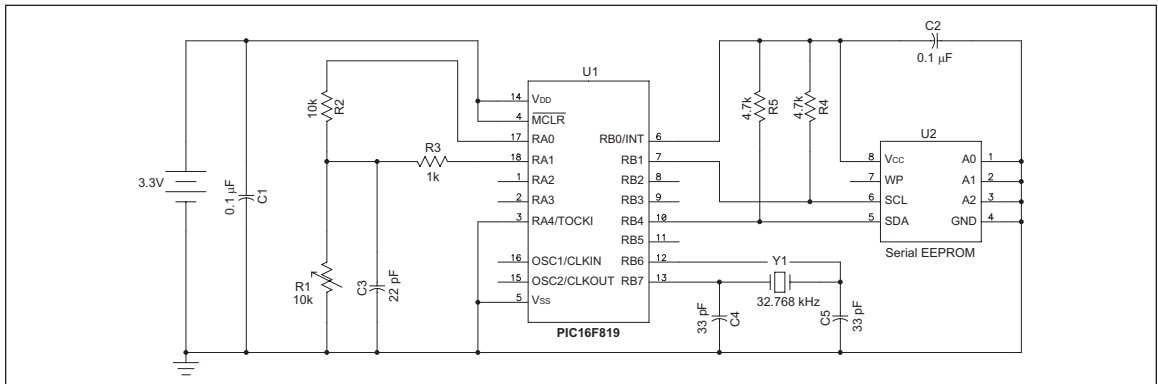
Figura: 1-1



El sistema de arriba es bastante simple y tiene todas sus partes indentificadas. Desafortunadamente tiene unos problemas ya que la EEPROM, el sensor y el circuito, están alimentados todo el tiempo. Para obtener la mínima corriente para este esquema sería ventajoso cerrar los distintos circuitos cuando no sean necesarios.

En la Figura 1-2, los pins I/O pins son usados para alimentar la EEPROM y el sensor. Como los pines I/O pueden entregar 20 mA, no es necesario utilizar componentes adicionales.

Figura: 1-2



TIP #2 Power Budgeting

“Power budgeting” es una técnica que es crítica para predecir el consumo de corriente y batería.

Mode	Time in Mode (mS)	Current (mA)		Charge Current * Time (Amp * Sec)
		By Part	Total	
Sleeping CPU Sensor EEPROM	1989		0.001	1.989 e-6
Sensor Warm-up CPU Sensor EEPROM	1	0.001 0.000 0.000	0.166	0.166 e-6
Sensing CPU Sensor EEPROM	1	0.048 0.165 0.000	0.213	0.213 e-6
Scaling CPU Sensor EEPROM	1	0.048 0.000 0.000	0.048	0.048 e-6
Storing CPU Sensor EEPROM	8	0.048 0.000 2.000	2.048	16.384 e-6
Total Time (mS)	2000	Total Charge (Amp*Sec)		18.800 e-6

$$\begin{aligned} \text{Average Current (mA)} &= \frac{\text{Total Charge}}{\text{Total Time}} \\ &= \frac{18.8 \text{ e-6}}{2000 \text{ e-3}} \frac{\text{Amp*Sec}}{\text{Sec}} \\ &= 0.009 \text{ mA} \\ \text{Peak Current} &= 2.048 \text{ mA} \end{aligned}$$

El siguiente ejemplo muestra el “power budget” para la Figura 2 en el Tip #1.

Calculando el “Battery Life”

Suponiendo una corriente promedio de .009 mA (Basado en el anterior “Power Budget”)

Battery	Capacity (mAHr)	Life			
		Hours	Days	Months	Years
CR1212	18	2000	83	2.8	0.23
CR1620	75	8333	347	11.6	0.96
CR2032	220	24444	1019	34.0	2.83

Después de completar un “power budget” es sencillo determinar la batería necesaria requerida por la aplicación. Si se consume mucha potencia, es muy simple determinar dónde se requiere un esfuerzo adicional para reducir el consumo de potencia.

TIP #3 WDT Alternative Wake-ups

La mayoría de las aplicaciones controlan la potencia del microprocesador usando el Sleep periódicamente. Hay tres formas de levantar a un PIC® MCU del "Sleep"

1. Recibir una interrupcion
2. Esperar al "Watchdog Timer"
3. Usar un periférico Ultra Low-Power Wake-Up (ULPWU)

Los nuevos nanoWatt PIC16F/18F tienen un Watchdog Timer (WDT) de baja corriente que consume 2-3 μA . Además, los propios dispositivos pueden apagar o encender el WDT dinámicamente para un mayor ahorro de corriente.

TIP #4 Stretched Dog

El Watchdog Timer (WDT) es comúnmente utilizado para despertar el PIC® MCU del "sleep". Cuanto más tiempo el PIC® MCU permanezca en Sleep menor es el consumo de las aplicaciones. Entonces es apropiado tener una duración límite del watchdog que sea lo suficientemente largo para su aplicación. Si la aplicación requiere de un muestreo con una tasa de una muestra por minuto, entonces el WDT debe despertar el PIC® MCU una vez por minuto. Los últimos PIC como el PIC18F1320 y el PIC16F684, tienen un WDT extendido que permite que el período del WDT sea ampliado hasta dos minutos.

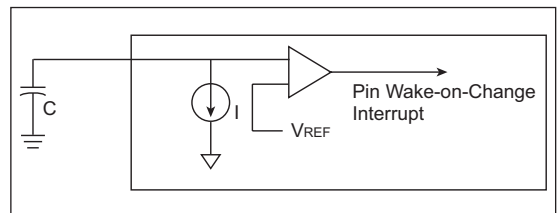
TIP #5 Low Power Timer1 Oscillator

Los dispositivos nanoWatt también ofrecen un oscilador (Timer1) robusto y de baja potencia que consume 2-3 μA . El Timer1 puede ser utilizado para generar interrupciones para levantar periódicamente del "Sleep" o de otros modos, y puede ser usado como base de un reloj de tiempo real. El overflow normal de dos seg del Timer1 (usando cristal de 32.786 kHz) puede extenderse a 16 segundos utilizando el "prescaler" de 1:8. Algunos dispositivos nanoWatt pueden usar al Timer1 como fuente de clock del sistema en vez del oscilador principal en los pines OSC1/OSC2. Reduciendo la velocidad de ejecución se reduce el total de corriente consumida.

TIP #6 Ultra Low-Power Wake-Up

Los dispositivos más nuevos tienen una modificación en el PORTA que crea un "Ultra Low-Power Wake-Up" (ULPWU). Un sumidero de corriente y un comparador fueron agregados que permiten usar un capacitor externo como un "wake-up timer".

Figura 6-1: Ultra Low-Power Wake-Up Peripheral



Si la precisión del Watchdog Timer no es necesaria, este periférico puede ahorrar una corriente considerada.

TIP #7 Low Energy Power Supplies

Para diseñar una fuente para un dispositivo pequeño puede ser engañoso. Hay varios factores que deben ser considerados:

1. Requerimientos Voltaje/Corriente
2. Aspectos químicos de la batería
3. Performance de la batería
4. Capacidad de la batería
5. Tamaño/Peso de la batería
6. Costo de la batería

Las baterías vienen de todo tipo de tamaño y tipo. Baterías de alta capacidad típicamente tienen una resistencia interna mayor, por lo que no son muy útiles para aplicaciones con alta demanda de corriente. Las más adecuadas para altas demandas de corriente tienen una menor capacidad o un peso mayor que una de alta resistencia de dimensiones similares. Baterías primarias también tienen mayor capacidad que las secundarias (recargables). Si V_{dd} debe ser mantenido constante, una batería con descarga plana de voltaje puede ser usada – dos ejemplos LiMg (primary) y NiMH (secondary). Si se necesita un mejor control del suministro de tensión se puede usar un regulador de tensión.

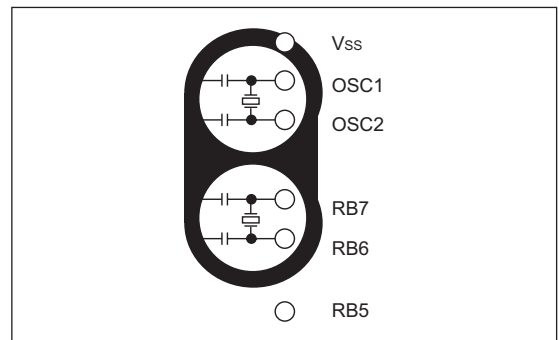
TIP #8 Low Power Timer1

Las aplicaciones que requieran que el Timer1 tenga un cristal conectado a los pines T1OSO y T1OSI deben tener en cuenta el layout del PCB.

El nuevo Timer1 de bajo consumo utiliza muy poca corriente, y esto provoca a veces que el circuito oscilador sea sensible a circuitos vecinos. El circuito oscilador, cristal y capacitores, deben estar situados lo más próximo al microcontrolador posible.

No deberían pasar circuitos por las proximidades del circuito oscilador. Si es inevitable tener un circuito de alta velocidad cerca del circuito oscilador, debe utilizarse un anillo de protección alrededor del circuito oscilador y los pines del microcontrolador como muestra la figura de abajo. Utilizar un plano de tierra debajo de los componentes del oscilador también ayuda a prevenir la interacción con circuitos de alta velocidad.

Figura 8-1: Guard Ring Around Oscillator Circuit and MCU Pins



TIPS 'N TRICKS CON SOFTWARE

Para reducir costos, los diseñadores deben aprovechar al máximo la memoria disponible en el MCU. La memoria programable es una de las causas más importantes en el costo del MCU. La optimización del código le evita tener que comprar más memoria que la necesaria. Aquí se presentan ideas para reducir el código.

TIP #9 Configuring Port Pins

Todos los PIC® MCUs tienen pines I/O bidireccionales. Algunos de estos pines pueden manejar entradas analógicas. Es muy importante prestar atención al tipo de señales aplicadas a estos pines para que consuman la menor potencia posible.

Pines de puertos no utilizados

Si un pin de un puerto no es utilizado, se puede dejar desconectado pero configurado como pin de salida tanto "high" como "low", o bien puede ser configurado como un entrada con una R externa (de 10 k Ω) llevándolo a Vdd o Vss. Si se configura como entrada, sólo va a fluir la corriente de "leakage" por el pin (la misma corriente fluiría si el pin se conectara directamente a Vdd or Vss). Las dos opciones permiten al pin ser usado luego para una entrada o salida sin cambios significativos en el hardware.

Entradas analógicas

Un pin de entrada digital consume la menor cantidad de potencia cuando la entrada esta cerca de Vdd o Vss. Si la tensión de entrada está en el medio de Vdd y Vss, los transistores dentro de la entrada digital están polarizados cerca de la region lineal y van a consumir una cantidad de corriente significante. Si el pin puede ser configurado como entrada analógica, el "buffer" digital es apagado reduciendo tanto la corriente del pin como la corriente total del microcontrolador.

Las entradas analógicas tienen una muy alta impedancia de entrada por lo que consumen poca corriente. Van a consumir menos corriente que una entrada digital si la tensión aplicada va a estar normalmente entre VDD y VSS. A veces es apropiado y posible configurar entradas digitales como entradas analógicas cuando la entrada digital debe ir a un estado de poco consumo.

Salidas Digitales

Una salida digital no consume corriente adicional otra que la corriente que fluye por el pin para alimentar al circuito externo. Preste atención a los circuitos externos para reducir al máximo su consumo de corriente.

TIP #10 I/O Initialization

A pesar que la siguiente práctica puede ser rutinaria, la inicialización PORT I/O es comunmente pasada por alto. En un POR (Power-on Reset), los registros PORT (por ejemplo) tienen un valor desconocido. Si los registros TRISB se configuran antes que los registros PORTB sean modificados, los pines de salida pueden generar pulsos no deseados durante la inicialización. La secuencia de instrucciones debajo son un ejemplo de como debe ser manejada la inicialización.

Ejemplo:

Clear PORTB and configure all PORTB I/O as outputs:

```
BANKSEL    PORTB    ;bank 0
CLRF      PORTB    ;clear PORTB
BANKSEL    TRISB    ;bank 1
CLRF      TRISB    ;configure for outputs
```

TIP #11 Two-Speed Start-Up

Esta característica es nueva para la familia de PICs y está disponible en algunos de los dispositivos nanoWatt. Usando este oscilador interno, el usuario puede ejecutar código mientras espera el Oscillator Start-up (OST) timer que expire (LP, XT or HS modes). Esta característica (llamada "Two-Speed Start-up") se habilita usando la configuración de bits IESO. El "Two-Speed Start-up" va a ser de clock desde INTRC(32 kHz) hasta que OST haya expirado.

Cambiar a una frecuencia interna de oscilación más rápida durante el "start-up" es posible usando el registro OSCCON. El siguiente ejemplo muestra las distintas etapas de como conseguir esto. La cantidad de cambio de frecuencias depende de la discreción del usuario. Se asume un cristal de 20 MHz (HS Mode) en un PIC16F para el siguiente ejemplo.

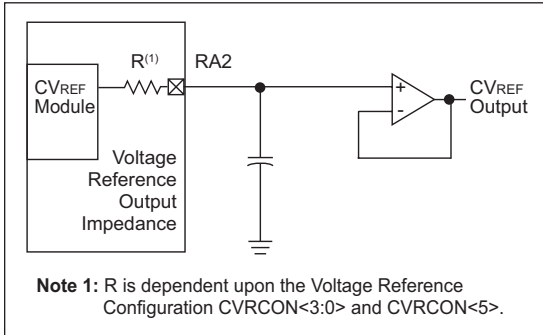
Ejemplo:

<u>T_{cy}</u> (Instruction Time)	<u>Instruction</u>	
	ORG 0x05	;Reset vector
125 μs @ 32 kHz	BSF STATUS,RP0	;bank1
125 μs @ 32 kHz	BSF OSCCON,IRCF2	;switch to 1 MHz
4 μs @ 1 MHz	BSF OSCCON,IRCF1	;switch to 4 MHz
1 μs @ 4 MHz	BSF OSCCON,IRCF0	;switch to 8 MHz
500 ns	application code	
500 ns	application code	
...	...	
..	...	
(eventually OST expires, 20 MHz crystal clocks the device)		
200 ns	application code	
...	...	
..	...	

TIP #12 How To Use a Comparator Reference as a D/A

El módulo del voltaje de referencia usado normalmente como referencia para los comparadores pueden ser usados como salidas simple del D/A con capacidades limitadas de conducción en RA2. Setear el bit CVROE (CVRCON<6>), y configure el pin como entrada analógica. Debido a la limitación, se debe usar un buffer externo en la salida de la referencia de tensión para conexiones externas a Vref. Ver Figura 12-1.

Figura 12-1: External Buffer for External Connections to VREF



TIP #13 How To Detect a Loss of Crystal/Resonator Oscillator

El "Fail-Safe Clock Monitor" puede ser usado para detectar la pérdida del oscilador de cristal/resonador u otra fuente de "clock" externa. Cuando se detecta la pérdida, un "clock" interno va a proveer el "clock", permitiendo un apagado elegante o entrar en un modo "limp-along" si el apagado no fuera necesario. Simplemente establezca el bit FCMEN bit en H en el "Configuration Word"(CONFIG1H<6>). Una velocidad mayor de "limp-along" puede elegirse en los bits de IRCF(OSCCON<6:4>) antes o después que ocurra la pérdida.

TIP #14 Enabling Idle Modes

La familia de PIC18F nanoWatt presentan múltiples modos Idle que pueden ser usados para reducir la potencia consumida. Seteando el bit de Idle (OSCCON<7>) y ejecutando un "Sleep" puede apagar el CPU y permitir a los periféricos continuar corriendo. En estos estados el consumo de potencia puede reducirse hasta un 96%.

TIP #15 How To Eliminate an External Crystal, Resonator or RC Timing Network

Si no se necesita una frecuencia de "clock" muy precisa use el clock interno. Tiene mejor estabilidad de frecuencia que un oscilador RC externo. El clock interno puede generar también una de las varias frecuencias para usar en el controlador, permitiendo que para reducir la demanda de corriente se reduzca la frecuencia del sistema. Cuando se requiere una velocidad mayor, puede seleccionar como se desea por medio del control del programa.

TIPS 'N TRICKS PARA HARDWARE/ SOFTWARE COMBINADOS

Esta seccion combina tips de hardware y software que ayudan a reducir la cantidad de componentes externos y reducir codigo.

TIP #16 Clock Switching PIC16F Dual Clock

La familia de PIC16F62X está equipada con un segundo oscilador interno de baja velocidad. Este oscilador está disponible cuando la fuente de clock que se configura es la RC interna (INTRC), Externa RC* (EXTRC) o External Resistor** (ER). El oscilador interno puede ser usado para operar el microcontrolador a velocidades bajas para reducir la potencia consumida. La velocidad actual del oscilador no está calibrada, por lo que se espera un 20%-40% de tolerancia. Para cambia los osciladores, simplemente cambie el bit 3 (OSCF) en el registro PCON. Cuando el OSCF esta clear, el oscilador de baja velocidad es usado. Si el OSCF está set, se utiliza el oscilador seleccionado en los bits de CONFIG.

* El modo EXTRC está disponible en "A parts".

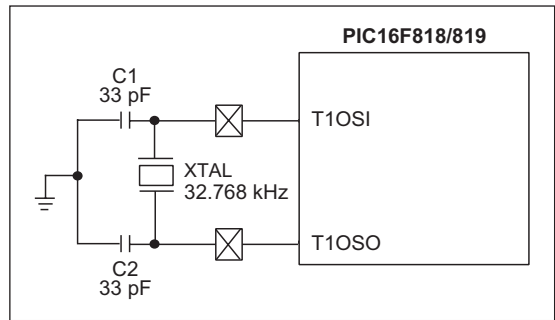
** El modo ER está disponible en "non-A parts".

Los dispositivos mas nuevos tiene un clock interno multi-speedclock. Pueden variar desde 8 8 MHZ hasta 31 kHz en 8 pasos. Esta velocidad es seleccionada usando el registro OSCCON.

TIP #17 Calibration

Un oscilador RC interno calibrado de fabrica puede requerir una nueva calibración debido al cambio de temperatura y de Vdd. El Timer1 puede ser usado para calibrar el oscilador interno conectando un cristal a 32.768. Recurra a AN244, "Internal RC Oscillator Calibration" para detalles completos de aplicación.

Figure 17-1: Timer1 Usado para Calibrar el Oscilador Interno



El calibrado se basa en una frecuencia medida del oscilador interno RC. Por ejemplo, la frecuencia elegida es 4 MHz, sabemos que el tiempo de una instruccion es de 1 μ s ($F_{osc}/4$) y el Timer1 tiene un período de 30.5 μ s ($1/32.768$ kHz). Esto significa que en un período del Timer1 el core puede ejecutar 30.5 instrucciones. Si los registros del Timer1 TMR1H:TMR1L son cargados con un valor conocido, se puede calcular la cantidad de intruccionnes que van a ser ejecutadas en un Timer1 "overflow". El número calculado se compara luego con el número de instrucciones ejecutadas por el core. Con el resultado, podemos deternimar si se necesita una re-calibración, y si la frecuencia debe ser aumentada o disminuida. Este ajuste usa el registro OSCTUNE, que tiene un rango de $\pm 12\%$ en 0.8% pasos.

Visite el centro de diseño de baja potencia en www.microchip.com para información adicional.

NOTES:

CAPITULO 3

Microcontroladores PIC® CCP y ECCP

Soluciones

Tabla de contenidos

CAPTURA

TIP #1	Measuring the Period of a Square Wave.....	3-3
TIP #2	Measuring the Period of a Square Wave with Averaging	3-3
TIP #3	Measuring Pulse Width	3-4
TIP #4	Measuring Duty Cycle	3-4
TIP #5	Measuring RPM Using an Encoder	3-5
TIP #6	Measuring the Period of an Analog Signal	3-6

COMPARA

TIP #7	Periodic Interrupts	3-8
TIP #8	Modulation Formats.....	3-9
TIP #9	Generating the Time Tick for a RTOS	3-10
TIP #10	16-Bit Resolution PWM	3-10
TIP #11	Sequential ADC Reader	3-11
TIP #12	Repetitive Phase Shifted Sampling	3-12

PWM

TIP #13	Deciding on PWM Frequency.....	3-14
TIP #14	Unidirectional Brushed DC Motor Control Using CCP	3-14
TIP #15	Bidirectional Brushed DC Motor Control Using ECCP.....	3-15
TIP #16	Generating an Analog Output.....	3-16
TIP #17	Boost Power Supply	3-17
TIP #18	Varying LED Intensity	3-18
TIP #19	Generating X-10 Carrier Frequency ...	3-18

COMBINACION DE CAPTURA Y COMPARA

TIP #20	RS-232 Auto-baud.....	3-19
TIP #21	Dual-Slope Analog-to-Digital Converter.....	3-21

INTRODUCCION

Microchip continúa presentando sus productos innovadores que son más pequeños, rápidos, fáciles de usar y confiables. Los "8-pin Flash microcontrollers (MCU)" son usados en un amplio rango de productos cotidianos, desde cepillos de dientes y secadores de pelo, hasta productos industriales y de medicina.

Los módulos de "Capture", "Compare" y PWM(CCP) que son encontrados en muchos de los microcontroladores de Microchip son usados principalmente para medir y controlar señales de pulsos basados en tiempo. El "Enhanced CPP"(ECCP), disponible en algunos dispositivos de Microchip, se diferencia del CPP regular porque provee una funcionalidad mejorada del PWM como soporte "full-bridge" y "half-bridge", "dead-band delay" programable y PWM "auto-shutdown" mejorado. Los módulos ECCP y CCP son capaces de hacer una amplia variedad de tareas. Este documento describe algunas de las guías básicas de cómo llevar adelante estos módulos en cada modo y también dar sugerencias para aplicaciones prácticas.

ECCP/CCP Register Listing

	Capture Mode	Compare Mode	PWM Mode
CCPxCON	Select mode	Select mode	Select mode, LSB of duty cycle
CCPRxL	Timer1 capture (LSB)	Timer1 compare (LSB)	MSB of duty cycle
CCPRxH	Timer1 capture (MSB)	Timer1 compare (MSB)	N/A
TRISx	Set CCPx pin to input	Set CCPx pin to output	Set CCPx pin(s) to output(s)
T1CON	Timer1 on, prescaler	Timer1 on, prescaler	N/A
T2CON	N/A	N/A	Timer2 on, prescaler
PR2	N/A	N/A	Timer2 period
PIE1	Timer1 interrupt enable	Timer1 interrupt enable	Timer2 interrupt enable
PIR1	Timer1 interrupt flag	Timer1 interrupt flag	Timer2 interrupt flag
INTCON	Global/peripheral interrupt enable	Global/peripheral interrupt enable	Global/peripheral interrupt enable
PWM1CON ⁽¹⁾	N/A	N/A	Set dead band, auto-restart control
ECCPAS ⁽¹⁾	N/A	N/A	Auto-shutdown control

Note 1: Only on ECCP module.

TIPS ‘N TRICKS Modo CAPTURE

En el modo "Capture", el valor de 16-bits del Timer 1 se captura en CCPRxH:CCPRxL cuando ocurre un evento en el pin CCPx. Un evento es definido como uno de los siguientes y es configurado con el CCPxCON<3:0>:

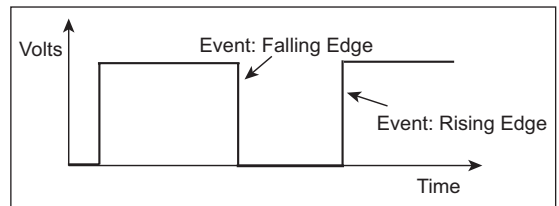
- Cada "falling edge"
- Cada "rising edge"
- Cada "4th rising edge"
- Cada "16th rising edge"

“Cuando debo usar el "Capture Mode"?”

El modo de captura es usado para medir el tiempo que transcurre entre dos eventos. Un evento, por lo general, es tanto un flanco ascendente o descendente. (Ver Figura 1)

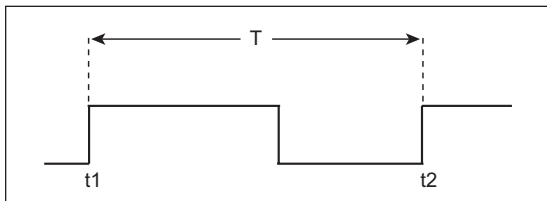
Un ejemplo de una aplicación donde el modo "Capture" es útil es leyendo un acelerómetro. Éstos varían el "duty cycle" de una onda cuadrada proporcionalmente a la aceleración que actúa en el sistema. Configurando el módulo CPP en modo "Capture", el microcontrolador PIC puede medir el "duty cycle" del acelerómetro con poca intervención del firmware del microcontrolador. El tip #4 entra en detalle de como medir el "duty cycle" configurando el módulo CPP en modo "Capture".

Figura 1: Defining Events



TIP #1 Measuring the Period of a Square Wave

Figura 1-1: Period



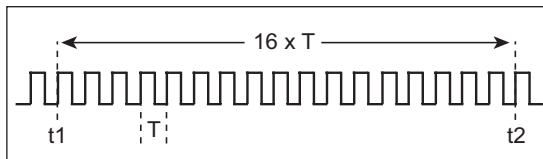
1. Configure los bits de control CCPxM3:CCPxM0 (CCPxCON<3:0>) para capturar cada flanco ascendente de la señal.
2. Configure el preescaler del Timer1 preescaler para que corra TMAX sin "overflow".
3. Habilite la interrupción del CCP (CCPxIE bit).
4. Cuando el CCP interrumpe:
 - a) Reste el tiempo capturado (t_2) menos el tiempo capturado (t_1) y guárdelo. (use el flag de la interrupcion del timer1 como indicador de "overflow").
 - b) Guarde el tiempo capturado(t_2).
 - c) Limpie el flag del Timer1 si es necesario.

El resultado obtenido en el paso 4 es el período (T).

Nota 1: T_{MAX} es el máximo período del pulso que puede ocurrir.

TIP #2 Measuring the Period of a Square Wave with Averaging

Figura 2-1: Medición del Período



1. Configure los bits de control CCPxM3:CCPxM0 (CCPxCON<3:0>) para capturar cada "16th rising edge" de la señal.
2. Configure el preescaler del Timer1 para que corra 16 TMAX sin "overflow".
3. Habilite la interrupción del CCP (CCPxIE bit).
4. Cuando el CPP interrumpe:
 - a) Reste el tiempo capturado (t_2) menos el tiempo capturado (t_1) y guardelo (use el flag de la interrupcion del timer1 como indicador de "overflow").
 - b) Guarde el tiempo capturado (t_2).
 - c) Limpie el flag del Timer1 si es necesario.
 - d) Desplace a la derecha 4 veces el valor obtenido en 4a para dividirlo por 16. Este resultado es el período (T).

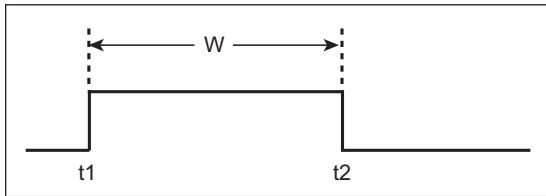
Nota 1: T_{MAX} es el máximo período del pulso que puede ocurrir.

Las siguientes son ventajas de este método comparado con medir los períodos manualmente:

- Menos interrupciones del CCP facilitan el flujo del programa.
- El uso de promedios brindan una excelente inmunidad hacia el ruido.

TIP #3 Measuring Pulse Width

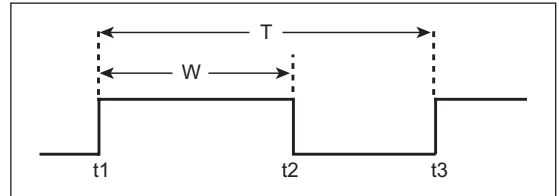
Figura 3-1: Pulse Width



1. Configure los bits de control CCPxM3:CCPxM0 (CCPxCON<3:0>) para capturar cada flanco ascendente de la señal.
2. Configure el prescaler del Timer1 para que corra WMAX sin "overflow".
3. Habilite la interrupción del CCP (CCPxIE bit).
4. Cuando el CCP interrumpa, guarde el valor capturado del timer (t1) y reconfigure los bits de control para capturar los flancos descendentes.
5. Cuando interrumpa el CCP nuevamente, reste el valor capturado (t2) menos el valor guardado (t1) – el resultado es el ancho del pulso (W).
6. Reconfigure los bits de control para capturar el siguiente flanco ascendente y empiece el proceso de nuevo. (repetiendo los pasos de 3 a 6).

TIP #4 Measuring Duty Cycle

Figura 4-1: Duty Cycle



El "duty cycle" de una señal es la relación entre el ancho de pulso (W) y el período (T). Los acelerómetros, por ejemplo, varían el "duty cycle" de su salida basados en la aceleración actuando sobre el sistema. El módulo CPP, configurado en modo "Capture", puede ser usado para medir el "duty cycle" de este tipo de sensores. Aquí se muestra como:

1. Configure los bits de control CCPxM3:CCPxM0 (CCPxCON<3:0>) para capturar cada flanco ascendente de la señal.
2. Configure el prescaler del Timer1 para que corra TMAX sin "overflow".
3. Habilite la interrupción del CCP (CCPxIE bit).
4. Cuando el CCP interrumpa, guarde el valor capturado del timer (t1) y reconfigure los bits de control para capturar los flancos descendentes.

Nota 1: T_{MAX} es el máximo período del pulso que puede ocurrir.

5. Cuando el CCP interrumpa nuevamente, reste el valor capturado (t2) menos el valor salvado previamente. Este resultado es el ancho del pulso (W).
6. Reconfigure los bits de control para detectar el próximo flanco ascendente.
7. Cuando ocurra la próxima interrupción, reste el valor capturado (t3) menos el valor guardado (t1). Este es el período (T) de la señal.
8. Divida T por W – este resultado es el "Duty Cycle".
9. Repita los pasos desde el 4 al 8.

TIP #5 Measuring RPM Using an Encoder

"Revolutions Per Minute"(RPM), o cuan rápido gira algo, puede ser sensado de varias maneras. Dos de los sensores más comunes usados para determinar RPM son los encoders ópticos y los sensores "Hall effect". Los encoders ópticos detectan la presencia de luz a través de una rueda ranurada montada sobre un eje giratorio (ver Figura 5-1). A medida que gira el eje, las ranuras de la rueda pasan por el ojo del encoder óptico. Comunemente, una fuente infrarroja situada del otro lado de la rueda emite luz que es recibida por el encoder óptico a través de las ranuras. Los sensores de "Hall effect" trabajan sensando la posición de los imanes de un motor eléctrico, o sensando un imán permanente montado en un objeto giratorio. (Ver figura 5-2). Estos sensores entregan uno o más pulsos por revolución (dependiendo del sensor).

Figura 5-1: Optical Encoder

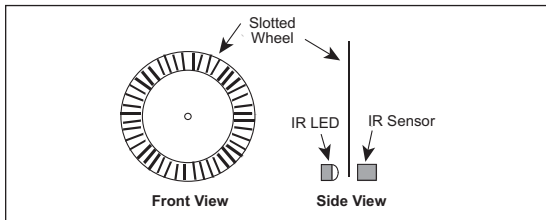
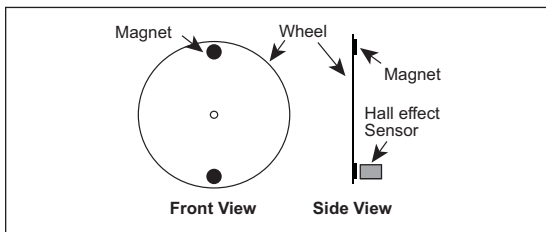


Figura 5-2: Hall Effect Sensor



En la Figura 5-3 y 5-4, la señal está en alto cuando la luz está pasando por medio de la ranura de la rueda del encoder y llegando al sensor óptico. En el caso del sensor "Hall effect", el estado alto corresponde al tiempo que el imán esté frente al sensor. Las figuras muestran la diferencia de las formas de onda para distintos RPMs. Note que a medida que el RPM incrementa, el período (T) y el ancho de pulso (W) disminuyen. Ambos son proporcionales al RPM. Sin embargo, como el período es el mayor de los intervalos, es una buena práctica medir el período de manera que el RMP leído por el sensor tenga la mejor precisión. Vea el Tip #1 para medir un período. La técnica para medir períodos utilizando promedios explicada en el Tip #2 es útil para la medición de altos RPMs.

Figura 5-3: Low RPM

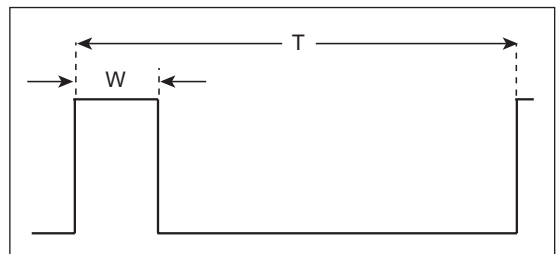
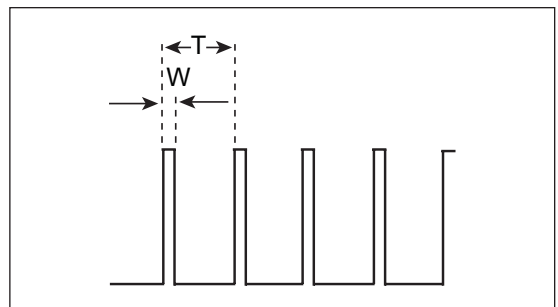


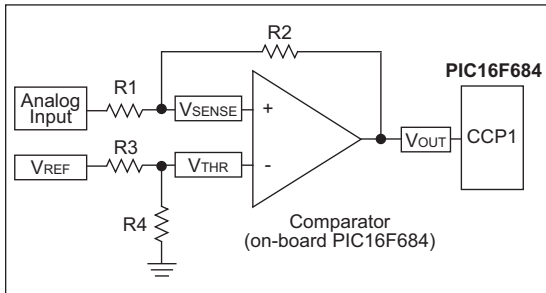
Figura 5-4: High RPM



TIP #6 Measuring the Period of an Analog Signal

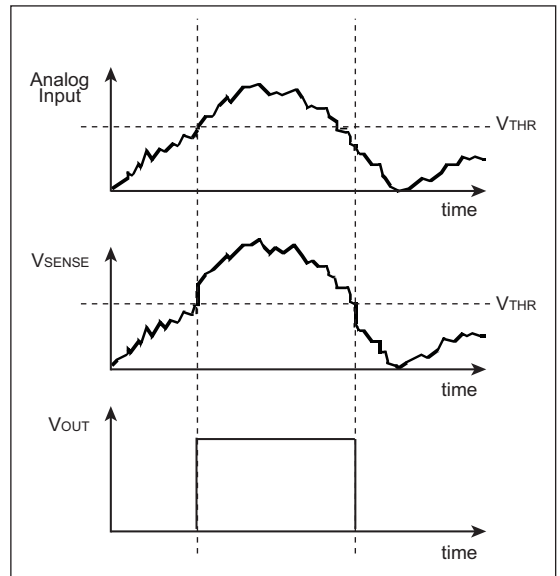
Microcontroladores con módulos "Analog Comparator" on-board, sumado al módulo CCP (o ECCP), pueden ser configurados facilmente para medir el período de una señal analógica. La Figura 6-1 muestra un ejemplo usando los periféricos del PIC16F684.

Figure 6-1: Circuit



R3 y R4 fijan el umbral para el comparador. Cuando la entrada analógica alcanza esa tensión umbral, Vout va a cambiar de LOW a HIGH. R1 y R2 proveen histéresis para asegurarse que cambios leves en la entrada no provoquen jitter en el circuito. La Figura 6-2 muestra el efecto de histéresis en la entrada. Mire específicamente a qué valor de Vsense la entrada analógica alcanza la tensión de umbral.

Figure 6-2: Signal Comparison



El módulo CPP, configurado como modo "Capture", puede medir el tiempo entre los flancos ascendentes de la salida del comparador (Vout). Este es el período de la entrada analógica, siempre y cuando la señal analógica llegue al Vthr cada período.

COMPARE TIPS 'N TRICKS

En el modo "Compare", el valor del registro de 16-bit es comparado constantemente con el del TMR1. Cuando coinciden el pin CCPx:

- Pasa a alto
 - Pasa a bajo
 - Permanece sin cambios, o
 - Alterna basado en la configuración del módulo. La acción del pin se determina mediante los bits de control bits CCPxM3:CCPxM0 (CCPxCON<3:0>).
- Una interrupción CCP se genera cuando ocurre una coincidencia.

Special Event Trigger

Normalmente, el Timer1 no es limpiado durante una interrupción CCP cuando el módulo CPP no es configurado en el modo "Compare". La única excepción es cuando el módulo CPP es configurado en el modo "Special Event Trigger". En este modo, cuando el Timer1 y el CCPRX son iguales, se genera la interrupción CCPx, el Timer1 es limpiado, y comienza la conversión A/D (si el A/D está habilitado).

“Por qué usar el modo "Compare"?”

El modo "Compare" trabaja como si fuese un cronómetro o cuenta regresiva. En este caso, se carga un valor de tiempo predeterminado y comienza la cuenta regresiva desde ese momento hasta que se alcance el cero.

El modo "Compare" trabaja de la misma manera excepto en un aspecto - cuenta desde cero hasta el valor predeterminado. Este módulo es útil para generar acciones específicas en tiempos precisos. Un timer puede ser usado para llevar a cabo la misma función. Sin embargo, implica estar cargando el valor reiteradamente. El modo "Compare" también tiene el beneficio de alterar automáticamente el estado del pin CCPx dependiendo del tipo de configuración optada.

TIP #7 Periodic Interrupts

Generar interrupciones periódicamente es una técnica muy útil implementada en varias aplicaciones. Esta técnica le permite al "loop" principal correr continuamente, y después, en intervalos periódicos, saltar a la rutina de interrupción para ejecutar tareas específicas (ej. leer el A/D). Normalmente, una interrupción mediante el "overflow" del timer es adecuado para generar interrupciones periódicas. Sin embargo, algunas veces es necesario interrumpir en intervalos que no pueden ser conseguidos por el "overflow" de un timer. El CCP configurado en modo "Compare" hace que esto sea posible.

Problema de ejemplo:

Un PIC16F684 corriendo con su oscilador interno de 8 MHz necesita ser configurado de manera que refresca un LCD cinco veces por segundo.

Paso #1: Determina un prescales del Timer1 que permitan un "overflow" mayor que 0.2 segundos.

- a) El "overflow" del Timer1 es: $Tosc * 4 * 65536 * prescaler$
- b) Para un prescaler de 1:1, el "overflow" del Timer1 es de 32.8 ms.
- c) Un prescale de 8 va a causar un "overflow" a un tiempo mayor que 0.2 segundos.
 $8 \times 32.8 \text{ ms} = 0.25\text{s}$

Paso #2: Calcular CCPR1 (CCPR1L y CCPR1H) para acortar el "time-out" a exactamente 0.2 segundos

- a) $CCPR1 = Interval \text{ Time} / (TOSC * 4 * prescaler) = 0.2 / (125 \text{ ns} * 4 * 8) = 5000 = 0xC350$
- b) Entonces, $CCPR1L = 0x50$, y $CCPR1H = 0xC3$

Paso #3: Configurando CCP1CON

El módulo CCP debe ser configurado en modo "Trigger Special Event". Este modo genera una interrupción cuando el Timer1 iguala al valor especificado en CCPR1L. Para este modo, es limpiado automáticamente
 $CCP1CON = 'b00001011'$.

Note 1: El modo "Trigger Special Event" también comienza la conversión A/D si el A/D está habilitado. Si no se desea esta función, el módulo CCP debe ser configurado en el modo "generate software interrupt-on-match only" (i.e., $CCP1CON = b'00001010'$). El Timer 1 se debe limpiar manualmente durante la interrupción del CCP.

TIP #8 Modulation Formats

El módulo CCP, configurado en modo "Compare" puede ser usado para generar una variedad de formatos de modulación. Las siguientes figuras muestran 4 formatos de modulaciones comunes:

Figure 8-1: Pulse-width Modulation

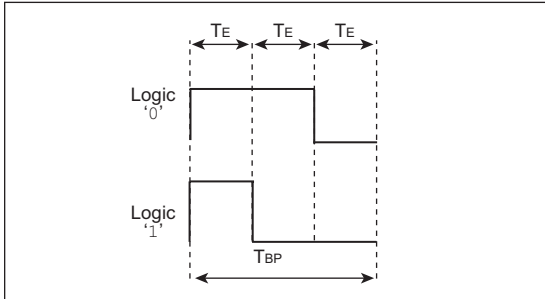


Figure 8-2: Manchester

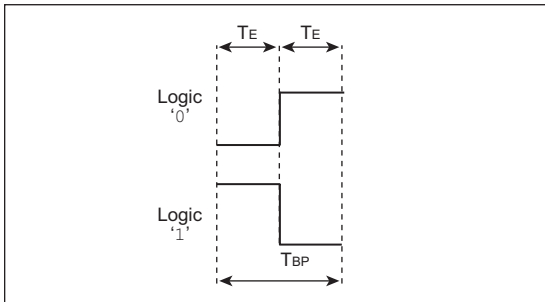


Figure 8-3: Pulse Position Modulation

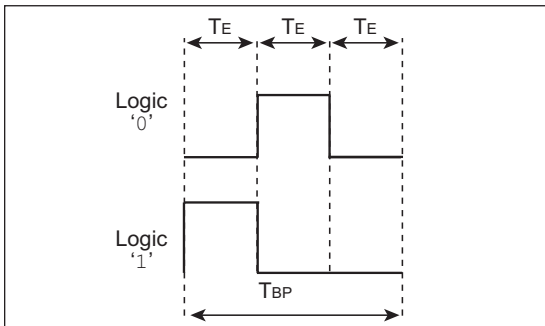
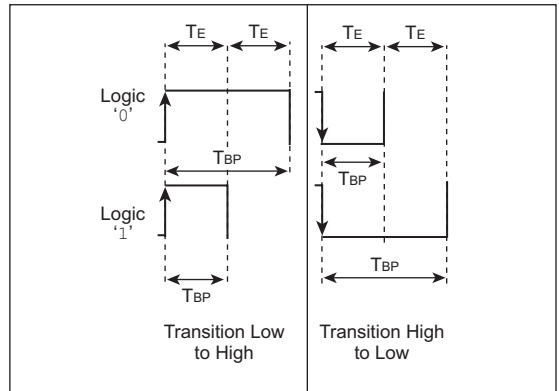


Figure 8-4: Variable Pulse-width Modulation



Las figuras muestran como son un '0' lógico o un '1' lógico para cada modulación. Una transmisión comunmente se asemeja a una transmisión serie que consiste en un bit de Start, seguido de 8 bits de datos y un bit de Stop.

TE es el elemento básico de tiempo de cada modulación y va a cambiar basado en la tasa de baudios deseada.

El modo "Trigger Special Event" puede ser usado para generar TE, (elemento de tiempo básico). Cuando se genera la interrupción CCPx, el código en el ISR va a implementar el formato de la modulación deseada (pueden usarse otros formatos adicionales)

TIP #9 Generating the Time Tick for a RTOS

Los "Real Time Operating Systems" (RTOS) requieren una interrupción periódica para operar. Esta interrupción, o "tick rate", es la base del sistema empleado por los RTOS. Por ejemplo, si se usa un tick de 2ms, el RTOS va a ejecutar tareas en los múltiplos de 2ms. Un RTOS también asigna prioridades a cada tarea, asegurándose que las tareas más críticas sean ejecutadas primero. La tabla 9-1 muestran un ejemplo de una lista de tareas, la prioridad de cada una y el intervalo de tiempo en el cual deben ser ejecutadas.

Tabla 9-1: Tasks

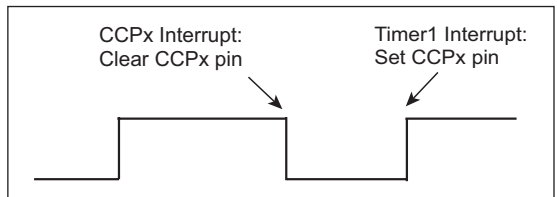
Task	Interval	Priority
Read ADC Input 1	20 ms	2
Read ADC Input 2	60 ms	1
Update LCD	24 ms	2
Update LED Array	36 ms	3
Read Switch	10 ms	1
Dump Data to Serial Port	240 ms	1

Las técnicas descritas en el Tip #7 pueden ser usadas para generar una interrupción periódica de 2ms usando el módulo CPP en modo "Compare".

Nota: Para más información de RTOSs y su uso, vea la "Application Note AN777" "Multitasking on the PIC16F877 with the Salvo™ RTOS".

TIP #10 16-Bit Resolution PWM

Figura 10-1: 16-Bit Resolution PWM



1. Configure el CCPx para limpiar la salida (CCPx pin) cuando haya coincidencia en el modo "Compare" (CCPxCON <CCPSM3:CCPxM0>).
2. Habilite la interrupción del Timer1.
3. Establezca el período de la forma de onda por medio del prescaler del Timer1 (T1CON <5:4>).
4. Establezca el "duty cycle" de la forma de onda usando CCPRxL and CCPRxH.
5. Setear el pin CCPx cuando se atienda la interrupción de "overflow" del Timer1

Nota 1: No se puede conseguir el 100% del "duty cycle" con esta implementación debido a la latencia de atender al Timer1. El período no es afectado porque la latencia de la interrupción va a ser la misma en todos los periodos siempre y cuando la interrupción del Timer1 sea atendida primero.

El Timer1 tiene 4 prescalers configurables. Estos son 1:1, 1:2, 1:4 y 1:8. Las posibilidades de frecuencia del PWM descritas arriba son determinadas por la ecuación 10-1.

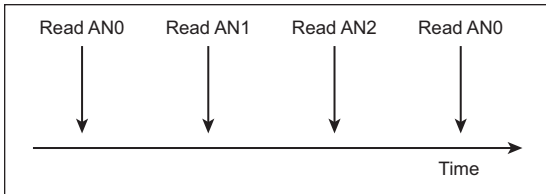
Ecuación 10-1

$$F_{PWM} = F_{osc} / (65536 * 4 * prescaler)$$

Para un microcontrolador corriendo con un oscilador de 20MHz (Fosc) equivalen a frecuencias de 76.3 Hz, 38.1 Hz, 19.1 Hz y 9.5 Hz para valores crecientes del prescaler.

TIP #11 Sequential ADC Reader

Figure 11-1: Timeline



El modo "Trigger Special Event" (un sub-modo del modo "Compare") genera un interrupción periódica y además da comienzo automáticamente al conversor A/D cuando el Timer1 coincide con CCPRxL y CCPRxH. El siguiente ejemplo demuestra esencialmente como leer los canales A/D periódicamente.

Ejemplo

Dado un PIC16F684 corriendo con su oscilador interno a 8 MHz, configúrelo para leer secuencialmente los pines analógicos AN0, AN1 y AN2 cada 30ms.

Paso #1: Determinar Timer1 Prescaler

- El "overflow" del Timer1 es: $Tosc \cdot 4 \cdot 65536 \cdot \text{prescaler}$.
- Para un prescaler de 1:1, el "overflow" del Timer1 ocurre en 32.8 ms.
- Esto es mayor que 30 ms, entonces un prescaler de 1 es adecuado.

Paso #2: Calcular CCPR1 (CCPR1L y CCPR1H)

- $CCPR1 = \text{Interval Time} / (Tosc \cdot 4 \cdot \text{prescaler}) = 0.030 / (125 \text{ ns} \cdot 4 \cdot 1) = 6000 = 0xEA60$
- Entonces, CCPR1L = 0x60, y CCPR1H = 0xEA

Paso #3: Configurar CCP1CON

El módulo ECCP debe ser configurado en el modo "Trigger Special Event". Este modo genera una interrupción cuando el Timer1 es igual al valor especificado en CCPR1. Timer1 se limpia automáticamente y el bit GO del ADCON0 se setea automáticamente. Para este modo, CCP1CON = 'b00001011'.

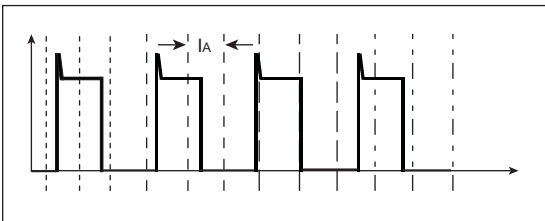
Paso #4: Agregar "Interrupt Service Routine Logic"

Cuando se genera la interrupción del ECCP elija el siguiente pin del A/D para leer alterando el registro ADCON0.

TIP #12 Repetitive Phase Shifted Sampling

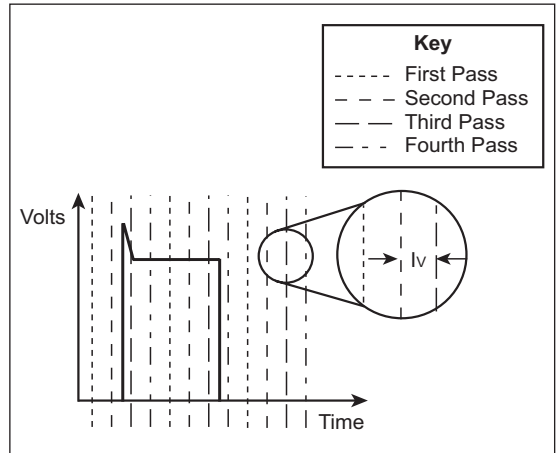
El muestreo "Repetitive Phase Shifted" es una técnica utilizada para incrementar artificialmente la tasa de muestreo del conversor A/D cuando se muestrean señales que son periódicas y constantes de período en período. Esta técnica funciona capturando regularmente muestras espaciadas de la forma de onda desde el principio hasta el final de su período. El muestreo de la próxima forma de onda se lleva a cabo de la misma manera sólo que la secuencia de muestro se retrasa un porcentaje del período de muestreo. Las formas de onda subsecuentes también son muestreadas, con cada secuencia retrasada levemente una de otra, hasta que el retraso sea igual a un período. Intercalando las distintas muestras producen un set de muestras a una tasa mayor. La Figura 12-1 muestra un ejemplo de una forma de onda con alta frecuencia.

Figure 12-1: High Frequency Periodic Waveform



Como se indica, la línea punteada muestra cuando las lecturas del A/D son llevadas a cabo durante el primer período de la señal. La línea con puntos mas extensos muestra las lecturas del A/D durante el segundo período, etc. La Figura 12-2 muestra estas mediciones pasadas a un sólo período.

Figure 12-2: Transposed Waveform



El módulo CCP se configura en el modo "Compare Special Event Trigger" para cumplir esta tarea. El cambio de fase se implemente eligiendo valores de CCPRxL y CCPRxH que no son sincrónicos con el período de muestro. Por ejemplo, si el período de una señal es 100 µs entonces el muestreo de una vez cada 22 µs nos da el siguiente set de muestreos a lo largo de 11 períodos (todos los valores en µs)

1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th
0	10	20	8	18	6	16	4	14	2	12
22	32	42	30	40	28	38	26	36	24	34
44	54	64	52	62	50	60	48	58	46	56
66	76	86	74	84	72	82	70	80	68	78
88	98		96		94		92		90	

Cuando estos valores se ordenan secuencialmente, revelan un intervalo de muestreo virtual (Iv) de 2µs desde 0 µs hasta 100µs, a pesar que el muestreo actual sea de 22 µs.

PWM TIPS 'N TRICKS

Los módulos ECCP y CCP producen un Pulse-Width Modulated (PWM) con una precisión de 10 bits en el pin CCPx. El módulo ECCP es capaz de transmitir una señal PWM en uno de los cuatro pines, designados P1A hasta P1D. Los modos de PWM disponibles en el módulo ECCP son:

- Single output (P1A only)
- Half-bridge output (P1A and P1B only)
- Full-bridge output forward
- Full-bridge output reverse

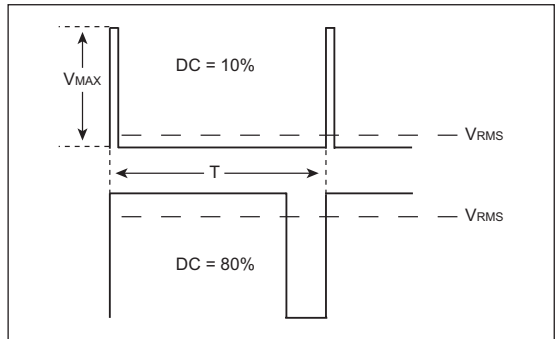
Una de las siguientes configuraciones debe ser elegidas cuando se use el módulo ECCP con el modo "Full Bridge" del PWM:

- P1A, P1C active-high; P1B, P1D active-high
- P1A, P1C active-high; P1B, P1D active-low
- P1A, P1C active-low; P1B, P1D active-high
- P1A, P1C active-low; P1B, P1D active-low

“Por qué usar el modo PWM?”

Los siguientes Tips 'n Tricks demuestran como el Pulse-Width Modulation (PWM) puede ser usado para cumplir una variedad de tareas como dimmers para LEDs hasta controlar la velocidad de un motor eléctrico DC. Todas estas aplicaciones están basadas en un principio básico de señales PWM - a medida que el "duty cycle" del PWM incrementa, la potencia entregada por el PWM incrementa. No sólo incrementa con el "duty cycle" sino que también incrementa linealmente. Las siguientes figuras ilustran este punto. Note que el RMS y la máxima tensión son funciones del "duty cycle" (DC) en la siguiente figura 12-3.

Figura 12-3: Duty Cycle Relation to VRMS



La Ecuación 12-1 muestra la relación entre VRMS y VMAX.

Ecuación 12-1: Relación entre VRMS y VMAX

$$V_{RMS} = DC \times V_{MAX}$$

TIP #13 Decidiendo la frecuencia PWM

En general, la frecuencia de PWM depende de la aplicación. Sin embargo, hay dos reglas de oro que con respecto a la frecuencia:

1. A medida que la frecuencia aumenta, también lo hace la demanda de corriente debido a pérdidas de alternado.
2. La capacitancia e inductancia de la carga tiende a limitar la respuesta en frecuencia del circuito.

En condiciones de bajo consumo, es una buena idea usar el mínimo de frecuencia posible para cumplir con una tarea con el fin de reducir pérdidas. En los circuitos la capacitancia y/o inductancia son factores los cuales deben ser considerados para la elección de la frecuencia del PWM analizando el circuito.

Control de un motor

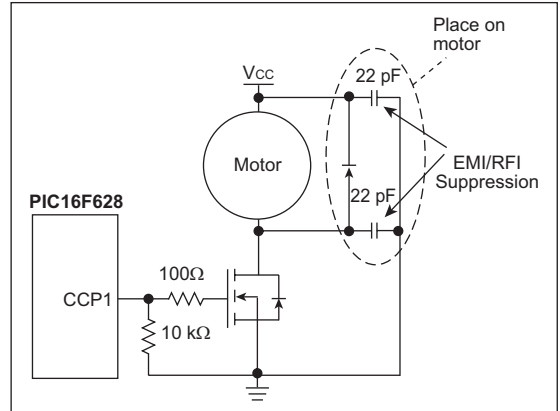
PWM es usado frecuentemente en el control de un motor debido a la eficiencia de los sistemas conmutados en contraste con aquellos lineales. Un aspecto importante a la hora de elegir la frecuencia de PWM para el control de un motor es la respuesta del motor ante cambios en el "duty cycle" del PWM. Un motor va a tener una respuesta más rápida a cambio de "duty cycle" a frecuencias mayores. Otra consideración es el ruido provocado por el motor. Motores DC del tipo "brushed" hacen un ruido intolerable en el rango audible cuando es llevado a altas frecuencias. Para eliminar esto, hay que llevar a este tipo de motores a frecuencias mayores que los 4kHz. (Los humanos pueden escuchar frecuencias hasta los 20kHz, sin embargo, la mecánica de los motores va a atenuar el ruido)

LEDs y Bombillas

PWM también se usa como reguladores de intensidad para LEDs y bombillas. Los parpadeos pueden ser percibidos con frecuencias menores a los 50Hz. Por lo tanto, es una buena regla usar el PWM a frecuencias de 100Hz o mayores.

TIP #14 Unidirectional Brushed DC Motor Control Using CCP

Figura 14-1: Brushed DC (BDC) Motor Control Circuit



La Figura 14-1 muestra un controlador unidireccional de velocidad para un "brushed DC motor". La velocidad del motor es proporcional al "duty cycle" de la salida del PWM en el pin CCP1. Los siguientes pasos muestran como configurar el PIC16F628 para generar un PWM a 20kHz y un 50% de "duty cycle". El microcontrolador corre con un cristal de 20 MHz.

Paso #1: Elegir el Prescaler del Timer2

- a) $F_{PWM} = F_{osc} / ((PR2 + 1) * 4 * \text{prescaler}) = 19531 \text{ Hz}$ para $PR2 = 255$ y prescaler de 1
- b) Esta frecuencia es menor a 20 kHz, por consiguiente el prescaler de 1 es adecuado.

Paso #2: Calcular PR2

$$PR2 = F_{osc} / (F_{PWM} * 4 * \text{prescaler}) - 1 = 249$$

Paso #3: Determinar CCPR1L y CCP1CON<5:4>

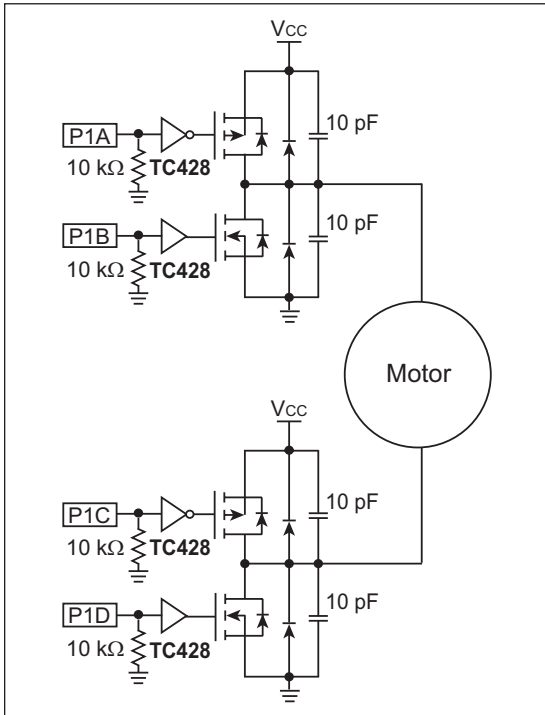
- a) $CCPR1L:CCP1CON<5:4> = \text{DutyCycle} * 0x3FF = 0x1FF$
- b) $CCPR1L = 0x1FF \gg 2 = 0x7F$,
 $CCP1CON<5:4> = 3$

Paso #4: Configurar CCP1CON

El módulo CCP se configura en modo PWM con los bits menos significativos del set de "duty cycle", entonces, $CCP1CON = 'b001111000'$.

TIP #15 Bidirectional Brushed DC Motor Control Using ECCP

Figura 15-1: Full-Bridge BDC Drive Circuit



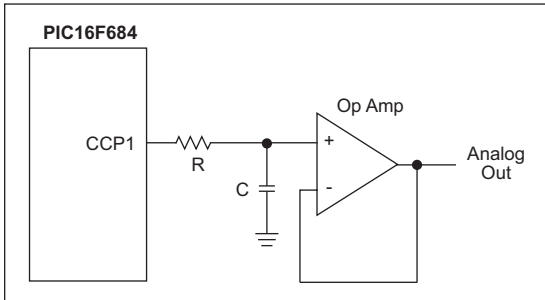
El módulo ECCP tiene incorporado opciones para controlar un "brushed DC motor". La Figura 15-1 muestra un "full-bridge drive circuit" conectado a un motor BDC. Las conexiones P1A, P1B, P1C y P1D son las salidas del ECCP cuando está configurado en los modos "Full-bridge Output Forward" o en "Full-bridge Output Reverse" (CCP1CON<7:6>). Para el circuito de la Figura 15-1, el módulo ECCP debe ser configurado en modo PWM: P1A, P1C active high; P1B, P1D active high (CCP1CON<3:1>). La razón de esto es que los drivers MOSFET (TC428) son configurados de forma que una entrada alta va a encender el MOSFET respectivo. La siguiente tabla muestra la relación entre los estados de operación, los estados de los pines del ECCP y el registro de configuración del ECCP.

State	P1A	P1B	P1C	P1D	CCP1CON
Forward	1	tri-state	tri-state	mod	'b01xx1100'
Reverse	tri-state	mod	1	tri-state	'b11xx1100'
Coast	tri-state	tri-state	tri-state	tri-state	N/A
Brake	tri-state	1	1	tri-state	N/A

Legend: '1' = high, '0' = low, mod = modulated, tri-state = pin configured as input

TIP #16 Generating an Analog Output

Figura 16-1: Low-Pass Filter



Las señales moduladas por ancho de pulso pueden ser usadas para crear un convertor "Digital-to-Analog" (D/A) con sólo unos pocos componentes externos. La conversión de las formas de onda del PWM a señales analógicas involucra el uso de un filtro pasa bajos analógico. Para eliminar las armónicas no deseadas causadas por el PWM al máximo posible, la frecuencia de la señal del PWM (FPWM) debe ser significativamente mayor que el ancho de banda (FBW) de la señal analógica deseada. La ecuación Ecuación 16-1 muestra esta relación.

Ecuación 16-1

$$F_{PWM} = K * F_{BW}$$

Donde las armónicas decrecen a medida que K crece

R y C son elegidos basado en la siguiente ecuación:

Ecuación 16-2

$$RC = 1/(2\pi F_{BW})$$

Elija un valor arbitrario de C y después calcule R. La atenuación de la frecuencia del PWM para un filtro RC dado es:

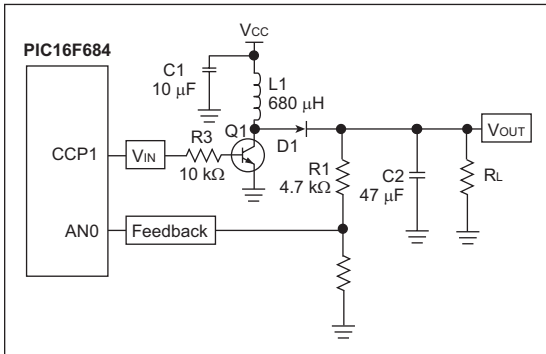
Ecuación 16-3

$$Att(dB) = - 10 * \log[1 + (2\pi F_{PWM} RC)^2]$$

Si la atenuación calculada en la Ecuación 16-3 no es suficiente, entonces debe incrementar el K de la Ecuación 16-1. Vea la "Application Note" AN538 "Using PWM to Generate Analog Output in PIC17C42" para más detalles de como usar un PWM para generar una salida analógica.

TIP #17 Boost Power Supply

Figura 17-1: Boost Power Supply Circuit



Hardware

PWM juega un rol clave en el diseño de "boost power supply". La Figura 17-1 muestra un circuito típico. El circuito funciona con Q1 haciendo de tierra al inductor (L1) durante la fase "high" de la señal PWM generada por CCP!. Esto causa un incremento en la corriente que fluye por L1 mientras se aplica Vcc. Durante la fase "low" de la señal PWM, la energía almacenada en L1 fluye por D1 hacia el capacitor de almacenamiento (C2) y la carga. Vout esta realcionado con Vin por medio de la Ecuación 17-1.

Nota: Technical Brief TB053 "Generating High Voltage Using the PIC16C781/782" provee detalles sobre el diseño de "boost power supply"

El primer parámetro que hay que determinar es el "duty cycle" basado en las tensiones de entrada y salida. Vea la Ecuación 17-1.

Ecuación 17-1

$$\frac{V_{OUT}}{V_{IN}} = \frac{1}{1 - D}$$

Luego, se debe elegir el valor del inductor basado en la corriente máxima requerido por la carga, la frecuencia de "switching" y el "duty cycle". La Ecuación 17-2, presenta la inductancia en términos de la corriente de carga, donde T es el período del PWM, D es el "duty cycle", and Iout es la máxima corriente de carga.

Ecuación 17-2

$$L = \frac{V_{IN} (1 - D) DT}{2 I_{OUT}}$$

El valor de L es elegido arbitrariamente para satisfacer la ecuación dada Iout, un "duty cycle" máximo de 75% y una frecuencia PWM en el rango de 10kHz hasta 100kHz.

Usando el valor de L elegido, la corriente de "ripple" es calculada usando la Ecuación 17-3.

Ecuación 17-3

$$I_{RIPPLE} = \frac{V_{IN} DT}{L}$$

La IRIPPLE no puede exceder la corriente de saturación del inductor. Si el valor de L produce una corriente de "ripple" mayor que ISAT, se necesita un inductor mayor.

Nota: Todas las ecuaciones dadas asumen un modo de corriente discontinua.

Firmware

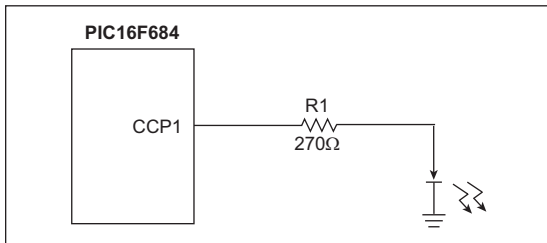
El "duty cycle" del PWM se varía con el microcontrolador de manera de mantener la tensión de salida en condiciones de carga fluctuantes. Un "firmware implemented PID control loop" es usado para regular el "duty cycle". El feedback del circuito "boost power supply" proveen un entrada para el control del PID.

Nota: La Application Note AN258 "Low Cost USB Microcontroller Programmer" provee detalles en el control de "firmware-based PID"

TIP #18 Varying LED Intensity

La intensidad de un LED puede ser variada con un PWM modulando la tensión sobre el LED. Un microcontrolador típicamente se conecta a un LED con el circuito mostrado en la Figura 18-1. El objetivo de R1 es limitar la corriente que pasa por el LED para que funcione con su corriente específica. Modulando el pin del LED en el microcontrolador va a variar la corriente promedio que atraviesa el LED y su intensidad. Como se mencionó en el TIP #13, los LEDs y otro tipo de fuentes luminosas deben ser moduladas a no menos de 100Hz para prevenir parpadeos.

Figure 18-1: LED Drive

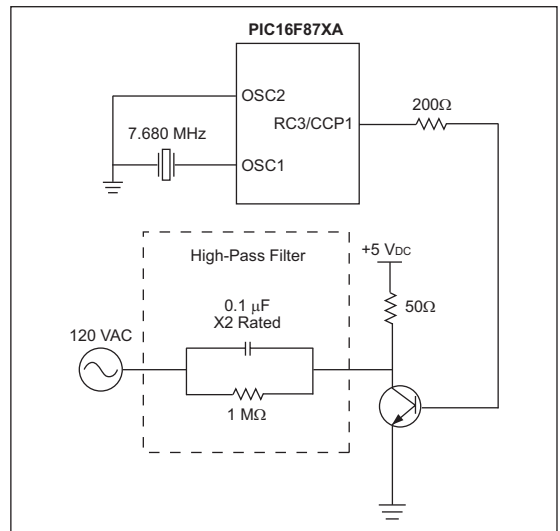


El módulo CCP, configurado en el modo PWM, es ideal para variar la intensidad de un LED. Los ajustes para la intensidad del LED son hechos simplemente variando el "duty cycle" de la señal PWM que alimenta al LED. Esto es llevado a cabo variando el registro CCPRxL entre 0 y 0xFF.

TIP #19 Generating X-10® Carrier Frequency

X-10 usa una onda cuadrada "piggybacked" de 120kHz (a un 50% de "duty cycle") para transmitir información sobre líneas de tensión de 60 Hz. El módulo CCP, corriendo en el modo PWM, puede crear una onda cuadrada precisa de 120 kHz como frecuencia de "carrier". La figura 19-1 muestra como la frecuencia de portadora de 120kHz es "piggybacked" sobre la sinusoidal de 60Hz.

Figura 19-1: Carrier Frequency Con Sinusoidal Waveform



X-10 especifica una frecuencia portadora de 120kHz (\pm 2kHz). El oscilador del sistema en la Figura 18-1 se opta por ser de 7.680 MHz, para que el módulo CCP pueda generar precisamente 120 kHz. X-10 requiere que la frecuencia portadora sea encendida y apagada en diferentes puntos en señal de los 60 Hz. Esto se consigue configurando el registro TRIS para el pin CCP1 tanto como entrada (frecuencia de carrier apagada) como salida (frecuencia de carrier encendida). Refiérase a la "Application Note" AN236 "X-10 Home Atomation Using the PIC16F877A" para más detalles de X-10 y seteo del módulo CPP adecuadamente.

COMBINATION CAPTURE AND COMPARE TIPS

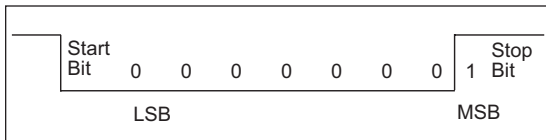
Los módulos CCP y ECCP pueden ser configurados "on the fly". Por lo tanto, estos módulos pueden llevar a cabo diferentes funciones en la misma aplicación siempre y cuando estas funciones se hagan una a la vez. Esta sección muestra ejemplos del módulo CCP trabajando en diferentes modos en la misma aplicación.

TIP #20 RS-232 Auto-baud

Una comunicación serie RS-232 tiene una variedad de "baud rates" para elegir. Múltiples tasas requieren un software que detecte la tasa de transmisión y ajuste las rutinas de envío y recepción adecuadas. El auto-baud es usado en aplicaciones donde pueden ocurrir múltiples tasas. El módulo CCP puede ser configurado en modo "Capture" para detectar el "baud rate" y luego ser configurado en modo "Compare" para generar o recibir transmisiones RS-232.

Para que el auto-baud funcione, se debe transmitir una calibración de un dispositivo a otro. Una posible calibración se muestra en la Figura 20-1. Midiendo el tiempo de esta calibración provee al dispositivo el "baud rate" a utilizar para las siguientes comunicaciones.

Figura 20-1: RS-232 Calibration Character



Implementación de la rutina de Auto-baud

1. Configure el módulo CCP para capturar un flanco descendente (comiendo del bit de Start).
2. Cuando se detecta el flanco descendente, guarde el valor de CCPR1.
3. Configure el módulo de CCP para capturar un flanco ascendente.
4. Cuando se detecta el flanco ascendente, guarde el valor de CCPR1.
5. Reste el valor en el paso 4 menos el valor en el paso 2. Este es el tiempo para los 8 bits.
6. Desplace el valor calculado en el paso 5 tres veces a la derecha para dividir por 8. Este resultado es el período del bit (TB).
7. Desplace el valor obtenido en el paso 6 a la derecha una vez. Este resultado es la mitad de un período de un bit.

El siguiente segmento de código muestra el proceso para transmitir y recibir datos. Esta misma función puede ser llevada a cabo por el módulo CCP configurándolo en modo "Compare" y generando una interrupción del CCP cada período de bit. Cuando se use este método, un bit se recibe o se envía cuando ocurre la interrupción.

Nota: Refierase al la "Application Note" AN712 "RS-232 Auto-baud for the PIC16C5X Devices" para más detalles de auto-baud.

Ejemplo 20-1: Rutina de Transmisión

```
TxRoutine
  MOVLW 8          ;preload bit counter
                  ;with 8
  MOVWF counter
  BCF TxLine      ;line initially high,
                  ;toggle low for START
                  ;bit
TxLoop
  CALL DelayTb    ;wait Tb (bit period)
  RRF RxByte,f    ;rotate LSB first into
                  ;the Carry flag
  BTFSS STATUS,C ;Tx line state equals
                  ;state of Carry flag
  BCF TxLine
  BTFSC STATUS,C
  BSF TxLine
  DECFSZ Counter,f ;Repeat 8 times
  GOTO TxLoop
  CALL Delay Tb   ;Delay Tb before
                  ;sending STOP bit
  BSF TxLine      ;send STOP bit
```

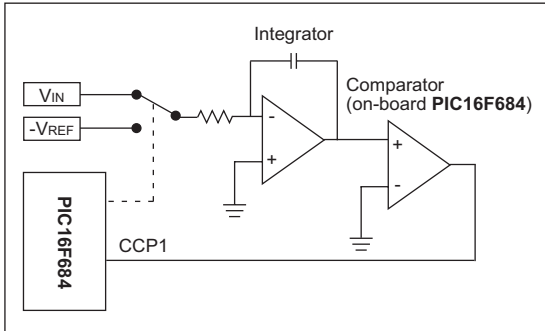
Ejemplo 20-2: Rutina de Recepción

```
RxRoutine
  BTFSC RxLine    ;wait for receive
                  ;line to go low
  GOTO RxRoutine
  MOVLW 8          ;initialize bit
                  ;counter to 8
  MOVWF Counter
  CALL Delay1HalfTb ;delay 1/2 Tb here
                  ;plus Tb in RxLoop
                  ;in order to sample
                  ;at the right time
RxLoop
  CALL DelayTb    ;wait Tb (bit
                  ;period)
  BTFSS RxLine    ;Carry flag state
                  ;equals Rx line
                  ;state
  BCF STATUS,C
  BTFSC RxLine
  BSF STATUS,C
  BTFSC RxLine
  BSF STATUS,C
  RRF RxByte,f    ;Rotate LSB first
                  ;into receive type
  DECFSZ Counter,f ;Repeat 8 times
  GOTO RxLoop
```

TIP #21 Dual-Slope Analog-to-Digital Converter

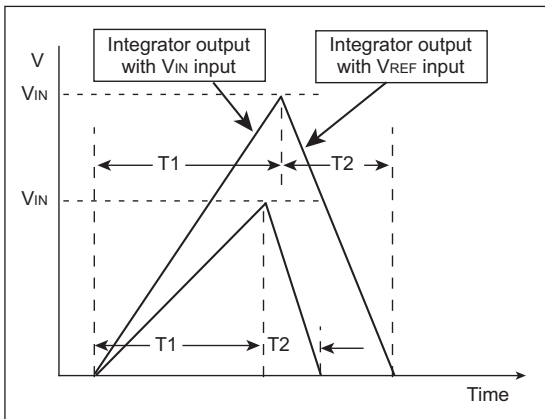
Un circuito para llevar a cabo una conversión A/D "dual-slope" utilizando el módulo CCP se muestra en la Figura 21-1.

Figura 21-1: Dual-Slope Analog-to-Digital Converter



Conversiones A/D "Dual-slope" funciona integrando la señal de entrada (V_{IN}) durante un tiempo fijo (T_1). La entrada después se cambia a una referencia negativa ($-V_{REF}$) y se integra hasta que la salida del integrador sea cero (T_2). V_{IN} es una función de V_{REF} y de la relación entre T_2 y T_1 .

Figura 21-2: V vs. Tiempo



Los componentes de este tipo de conversión son el tiempo fijo y el "timing" del flanco descendente. El módulo CCP puede llevar a cabo ambos componentes por medio del modo "Compare" y "Capture" respectivamente:

1. Configure el módulo CCP en modo "Compare mode, Special Event Trigger".
2. Cambie la entrada analógica al integrador de V_{REF} a V_{IN} .
3. Use el módulo CCP para esperar T_1 (T_1 elegido basándose en el valor de C).
4. Cuando ocurre la interrupción del CCP, cambie la entrada analógica al regulador de V_{IN} a V_{REF} y reconfigure el módulo en modo "Capture" esperando el flanco descendente.
5. Cuando ocurra la próxima interrupción del CCP, el tiempo capturado por el módulo es T_2 .
6. Calcule V_{IN} usando la Ecuación 21-1.

Ecuación 21-1

$$V_{IN} = V_{REF} \frac{T_2}{T_1}$$

NOTES:

CAPITULO 4

Microcontroladores PIC® Comparador Soluciones

Tabla de contenidos

INTRODUCCION

TIP #1:	Low Battery Detection	4-2
TIP #2:	Faster Code for Detecting Change.....	4-3
TIP #3:	Hysteresis.....	4-4
TIP #4:	Pulse Width Measurement	4-5
TIP #5:	Window Comparison	4-6
TIP #6:	Data Slicer.....	4-7
TIP #7:	One-Shot.....	4-8
TIP #8:	Multi-Vibrator (Square Wave Output) .	4-9
TIP #9:	Multi-Vibrator (Ramp Wave Output) ...	4-10
TIP #10:	Capacitive Voltage Doubler	4-11
TIP #11:	PWM Generator	4-12
TIP #12:	Making an Op Amp Out of a Comparator	4-13
TIP #13:	PWM High-Current Driver	4-14
TIP #14:	Delta-Sigma ADC	4-15
TIP #15:	Level Shifter	4-16
TIP #16:	Logic: Inverter.....	4-16
TIP #17:	Logic: AND/NAND Gate	4-17
TIP #18:	Logic: OR/NOR Gate.....	4-18
TIP #19:	Logic: XOR/XNOR Gate.....	4-19
TIP #20:	Logic: Set/Reset Flip Flop	4-20

INTRODUCCION

Microchip continúa presentando sus productos innovadores que son más pequeños, rápidos, fáciles de usar y confiables. Los "8-pin Flash microcontrollers (MCU)" son usados en un amplio rango de productos cotidianos, desde cepillos de dientes y secadores de pelo, hasta productos industriales y de medicina.

La familia de PIC12F/16F con comparadores de tensión "on-chip" mezcla todas las ventajas de la arquitectura del PIC® MCU y la flexibilidad de la memoria Flash programable con la naturaleza de una señal de un comparador de voltaje. Juntos forman un híbrido digital/analógico con la potencia y flexibilidad para trabajar en un mundo analógico.

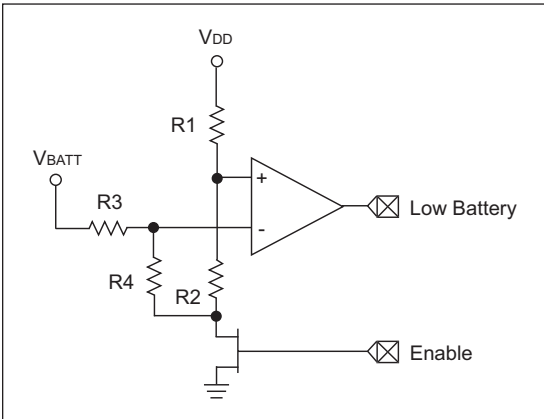
La flexibilidad de la memoria Flash y las excelentes herramientas de desarrollo, que incluyen "low-cost In-Circuit Debugger", "In-Circuit Serial Programming™" y "MPLAB® ICE 2000 emulation", hacen que estos dispositivos sean ideales para cualquier aplicación de control.

La siguiente serie de Tips 'n Tricks puede ser aplicada a una variedad de aplicaciones para ayudar a crear los más discretos comparadores de tensión o microcontroladores con comparadores de tensión "on-chip".

TIP #1 Low Battery Detection

Cuando se opera con una batería como suministro de energía, es importante en un circuito ser capaz de determinar cuando la carga de la batería es insuficiente para el funcionamiento normal. Comúnmente, esto lo hace un circuito comparador similar al periférico "Programmable Low Voltage Detect"(PVLD). Si el PVLD no está disponible en el microcontrolador, se puede construir un circuito similar usando un comparador y algunos componentes externos.(Ver Figura 1-1 y 1-2). El circuito en la Figura 1-1 asume que el microcontrolador está operando desde una fuente de tensión regulada. El circuito en la Figura 1-2 asume que la fuente del microcontrolador no está regulada.

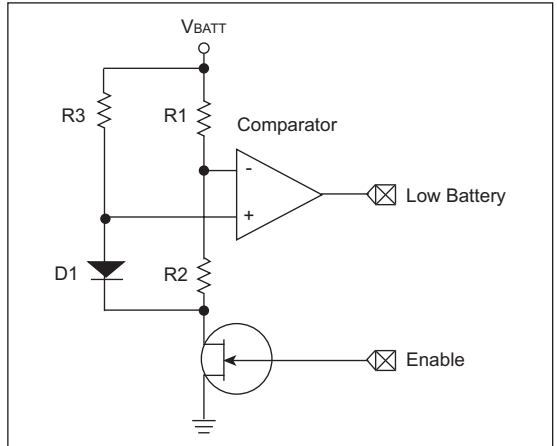
Figura 1-1: Regulated Supply



El comparador va a disparar cuando el voltaje, VBATT = 5.7V: R1 = 33k, R2 = 10k, R3 = 39k, R4 = 10k, VDD = 5V.

En la Figura 1-1, las resistencias R1 y R2 son elegidos para poner la tensión en la entrada no inversora a aproximadamente 25% de Vdd. R3 y R4 son elegidos para setear la tensión de la entrada inversora igual a la no inversora cuando la tensión de la batería es igual al mínimo de la tensión de operación del sistema.

Figure 1-2: Unregulated Supply



El comparador va a disparar cuando VBATT = 3V: R1 = 33k, R2 = 10k and R3 = 470Ω.

En la Figura 1-2, la resistencia R3 se elige para polarizar el diodo D1 arriba de su tensión de directa cuando Vbatt es igual al mínimo voltaje tolerado por el sistema. Las resistencias R1 y R2 se eligen para fijar la tensión de la entrada inversora igual a la tensión de directa de D1.

TIP #2 Faster Code for Detecting Change

Cuando se usa un comparador para monitorear un sensor, es importante saber cuando ocurre un cambio. Para detectar este cambio en la salida de un comparador, el método tradicional, fue siempre guardar una copia de la salida y comparar periódicamente el valor guardado con el valor actual en la salida para determinar el cambio. Un ejemplo de este tipo de rutinas se muestra a continuación.

Ejemplo 2-1

```
Test
  MOVF  hold,w      ;get old Cout
  XORWF CMCON,w    ;compare to new Cout
  ANDLW COUTMASK
  BTFSC STATUS,Z
  RETLW 0           ;if = return "no change"
  MOVF  CMCON,w    ;if not =, get new Cout
  ANDLW COUTMASK  ;remove all other bits
  MOVWF hold       ;store in holding var.
  IORLW CHNGBIT   ;add change flag
  RETURN
```

Esta rutina requiere de 5 instrucciones para cada testeo, 9 instrucciones si se produce un cambio, y 1 posición de RAM para guardar la salida anterior. Un método más rápido para microcontroladores con un comparador es usar un flag de interrupción del comparador para determinar si se produjo un cambio.

Ejemplo 2-2

```
Test
  BTFSS PIR1,CMIF ;test comparator flag
  RETLW 0         ;if clear, return a 0
  BTFSS CMCON,COUT ;test Cout
  RETLW CHNGBIT  ;if clear return
                  ;CHNGFLAG
  RETLW COUTMASK + CHNGBIT;if set,
                  ;return both
```

Esta rutina requiere de 2 instrucciones para cada testeo, 3 instrucciones si se produce algún cambio, y no requiere memoria RAM. Si el "flag" de interrupción no puede ser usado, o si dos comparadores comparten un "flag" de interrupción, un método alternativo que usa la polaridad de salida del comparador puede ser usado.

Ejemplo 2-3

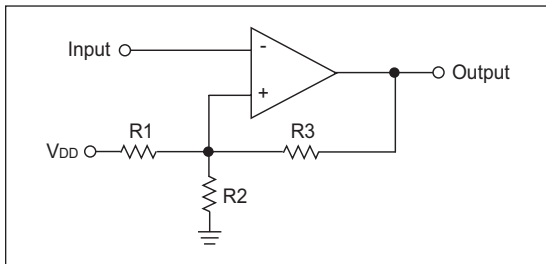
```
Test
  BTFSS CMCON,COUT ;test Cout
  RETLW 0         ;if clear, return 0
  MOVLW CINVBIT  ;if set, invert Cout
  XORWF CMCON,f  ;forces Cout to 0
  BTFSS CMCON,CINV ;test Cout polarity
  RETLW CHNGFLAG ;if clear, return
                  ;CHNGFLAG
  RETLW COUTMASK + CHNGFLAG;if set,
                  ;return both
```

Esta rutina requiere de 2 instrucciones para cada testeo, 5 instrucciones si ocurre algún cambio, y no requiere almacenamiento.

TIP #3 Hysteresis

Cuando las tensiones de entrada de un comparador son casi iguales, ruido externo o ruido de "switching" desde adentro del microcontrolador pueden causar que la salida del comparador oscile. Para prevenir esto, la salida del comparador es realimentada para formar histéresis (ver Figura 3-1). Histéresis sube el umbral del comparador cuando la entrada esta por debajo del umbral, y lo baja cuando la entrada esta por encima del umbral. El resultado es que la entrada debe exceder el umbral para causar un cambio en la salida. Si el exceso es mayor que el ruido presente en la entrada, la salida del comparador no va a oscilar.

Figura 3-1: Comparador con Hysteresis



Para calcular los valores de resistencia requeridos, primero determine los valores de umbral superior e inferior para prevenir oscilaciones (VTH and VTL). Usando VTH y VTL, el umbral promedio se puede calcular usando la siguiente ecuación.

Equation 3-1

$$V_{AVG} = \frac{V_{DD} * V_{TL}}{V_{DD} - V_{TH} + V_{TL}}$$

Luego, elija los valores de resistencia que cumplan con la Ecuación 3-2 y calcule la resistencia usando la Ecuación 3-3.

Nota:
Una corriente continua va a fluir por R1 y R2. Para limitar la disipación de potencia en R1 y R2 la resistencia de R1 y R2 deben ser por lo menos de 1k. La resistencia total de R1 y R2 debe ser menor de 10K para mantener a R3 pequeña. Grandes valores de R3, 100k-1MΩ, puede producir tensiones de offset en la entrada no inversora debido a la corriente de polarización del comparador.

Ecuación 3-2

$$V_{AVG} = \frac{V_{DD} * R_2}{R_1 + R_2}$$

Ecuación 3-3

$$R_{EQ} = \frac{R_1 * R_2}{R_1 + R_2}$$

Luego determine el divisor del realimentador, usando la Ecuación 3-4.

Ecuación 3-4

$$D_R = \frac{(V_{TH} - V_{TL})}{V_{DD}}$$

Finalmente, calcule la resistencia de "feedback" R3 usando la Ecuación 3-5.

Ecuación 3-5

$$R_3 = R_{EQ} [(\frac{1}{D_R}) - 1]$$

Ejemplo:

- A VDD = 5.0V, VH = 3.0V and VL = 2.5V
- VAVG = 2.77V
- R = 8.2k and R2 = 10k, gives a VAVG = 2.75V
- REQ = 4.5k
- DR = .1
- R3 = 39k (40.5 calculated)
- VHACT = 2.98V
- VLACT = 2.46V

TIP #4 Pulse Width Measurement

Para medir el ancho de pulso alto o bajo de una señal analógica entrante, el comparador puede ser combinado con el Timer1 y la opción "Timer1 Gate input". (Vea Figura 4-1). El "Timer1 Gate" actúa como un contador para el Timer1. Si el input es low, el Timer1 va a contar. Si T1G input es high, el Timer1 no va a contar. Usando T1G con el comparador permite al diseñador medir el tiempo entre un cambio a la salida entre un cambio "high-to-low" y un "low-to-high". Para hacer una medición entre un cambio "low-to-high" y un "high-to-low", el único cambio requerido es setear el bit CINV en el registro CMCON del comparador que invierte la salida. Como la salida del comparador puede cambiar asincrónicamente con el clock del Timer1, solamente los comparadores con habilidad para sincronizar sus salidas con el clock del Timer1 deben ser usados y sus bit C2SYNC deben estar seteados.

Si el comparador "on-chip" no tiene la habilidad de sincronizar su salida con el clock del Timer1, la salida puede ser sincronizada externamente con un flip-flop D discreto (Ver Figura 4-2).

Nota: El flip-flop debe ser activado con flanco flanco descendente para evitar carreras

Figura 4-2: Comparador Sincronizado Externamente

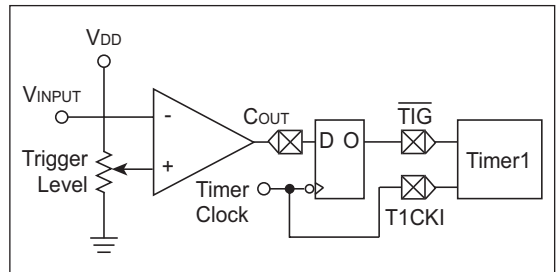
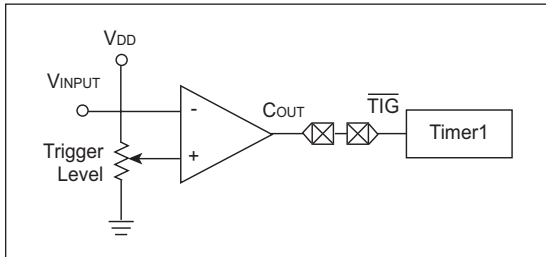


Figura 4-1: Comparador con Timer1 y T1G



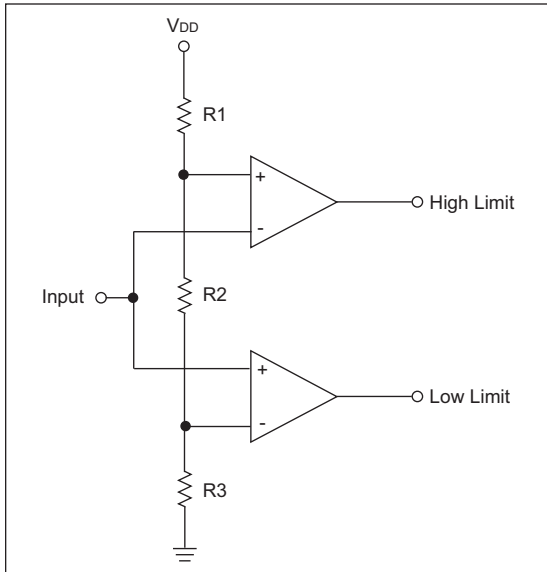
TIP #5 Window Comparison

Cuando se monitorea un sensor externo, es conveniente poder saber cuando la señal se ha extendido del rango seguro pre establecido o ventana de operación. Esta ventana provee al circuito una alarma cuando la señal excede los límites seguros, ignorando fluctuaciones menores dentro del rango seguro de operación. Para implementar una ventana de comparación se necesitan dos comparadores de tensión y 3 resistencias (ver Figura 5-1).

Para calcular los valores de R1, R2 y R3, busque valores que satisfagan la Ecuación 5-1 y 5-2

Nota: Una corriente continua va a fluir por R1, R2 y R3. Para limitar la disipación de potencia en las resistencias, la resistencia total de R1, R2 y R3 debe ser al menos 1k. El total de la resistencia de R1, R2 y R3 debe ser menor que 1MΩ para prevenir voltajes de offset debido a las corrientes de polarización del comparador.

Figure 5-1: Window Comparator



Las resistencias R1, R2 y R3 forman un divisor de tensión que generan el umbral superior e inferior. Las salidas "HIGH LIMIT" y "LOW LIMIT" son ambas "active high", generando un uno lógico en la salida "HIGH LIMIT" cuando la tensión de entrada sube por encima del umbral superior, y un uno lógico en la salida "LOW LIMIT" cuando la tensión de entrada está por debajo del umbral inferior.

Ecuación 5-1

$$V_{TH-HI} = \frac{V_{DD} * (R3 + R2)}{R1 + R2 + R3}$$

Ecuación 5-2

$$V_{TH-LO} = \frac{V_{DD} * R3}{R1 + R2 + R3}$$

Ejemplo:

- V_{DD} = 5.0V, V_{TH} = 2.5V, V_{TL} = 2.0V
- R1 = 12k, R2 = 2.7k, R3 = 10k
- V_{TH} (actual) = 2.57V, V_{TL} (actual) = 2.02V

Agregando histéresis:

Para agregar histéresis al "HIGH LIMIT" del comparador, siga el procedimiento indicado en el Tip #3. Use combinaciones de R2 y R3 como la resistencia R2 en el Tip #3.

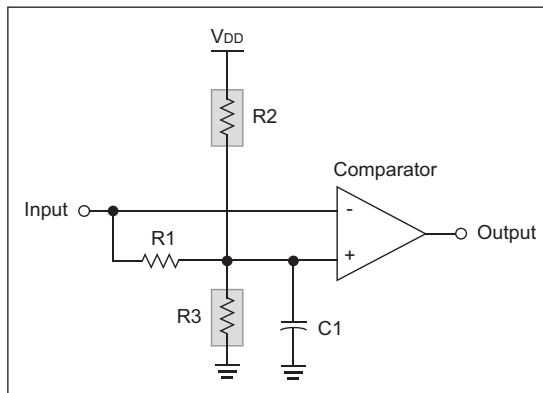
Para agregar histéresis al "LOW LIMIT" del comparador, elija un valor adecuado de Req, 1k hasta 10kΩ, y ubíquela entre la entrada del circuito y la entrada no inversora del "LOW LIMIT" del comparador. Luego calcule la resistencia de realimentación usando la Ecuación 3-4 y 3-5.

TIP #6 Data Slicer

En transmisiones de datos guiados o no guiados, la señal de datos puede estar sujeta a cambios de offset de continua debido a cambios de temperaturas y otros factores en el sistema. Cuando esto pasa, usando un simple nivel de comparación para recuperar los datos no es posible porque el offset de continua puede exceder la amplitud pico a pico de la señal. El circuito típico usado para recuperar la señal en esta situación es un "data slicer".

El "data slicer" que se muestra en la Figura 6-1 opera comparando la señal de entrada con una referencia "deslizante" que proviene del promedio del valor de DC de la señal de entrada. El promedio de DC se encuentra usando un simple filtro pasa-bajo RC (R1 y C1). La frecuencia del filtro RC debe ser lo suficientemente alta para ignorar el nivel de DC y lo suficientemente baja para dejar pasar los datos transferidos. Las resistencias R2 y R3 son opcionales. Proveen una leve polarización a la referencia, ya sea "high" o "low", para darle una referencia al estado de salida cuando no se recibe ningún dato. R2 va a polarizar la salida "low" y R3 va a polarizar la salida "high". Sólo un resistor debe ser usado a la vez, y su valor debe ser por lo menos de 50 a 100 veces mayor que R1.

Ejemplo 6-1: Data Slicer



Ejemplo:

Tasa de datos 10 kbits/segundo. Un filtro pasa-bajos de frecuencia de 500 Hz:
 R1 = 10k, C1 = 33 μ F, R2
 o R3 debe ser de 500k a 1 MB.

TIP #7 One-Shot

Quando se utilizan señales de corta duración o "glitches", es conveniente extender el evento usando un mono-estable, "multi-vibrator" o "one-shot". Cuando se reciben pulsos en la entrada, el "one-shot" dispara manteniendo su salida por un período de tiempo preseteado. Esto extiende la entrada "short trigger" en una larga salida que el microcontrolador pueda capturar.

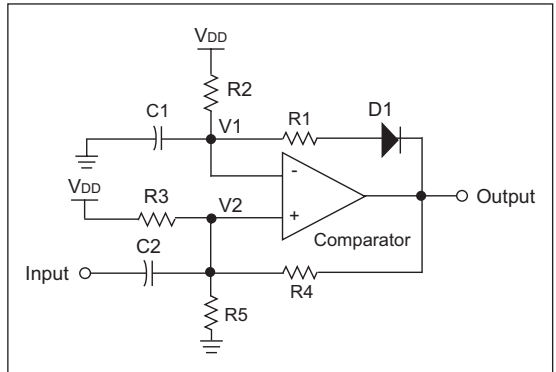
El circuito está diseñado con dos realimentadores alrededor de un comparador. El primero es un realimentador positivo de histéresis que fija dos valores de umbrales, V_{hi} y V_{lo} , basado en el estado de la salida del comparador. El segundo realimentador es un circuito RC.

El circuito "one-shot" presentado en la Figura 7-1 se dispara con una transición "low-high" en su entrada y genera un pulso de salida "high". Usando los valores de componentes del ejemplo, la operación del circuito es la siguiente. Antes de disparar, el C1 se va a cargar a una tensión levemente mayor a 0.7V debido a la resistencia R2 y D1 ($R1 \ll R2$). La salida del comparador va a ser "low", manteniendo la entrada no inversora levemente menor que 0.7V debido a la realimentación con histéresis R3, R4 y R5 (el límite inferior de la histéresis está diseñada para ser menor que 0.7V). Con la entrada no inversora mantenida en "low", el C2 se va a cargar hasta la diferencia entre la entrada del circuito y la tensión presente de la entrada no inversora.

Quando la entrada del circuito es un pulso "high", la tensión presente en la entrada no inversora es llevada por sobre 0.7V debido a la carga en C2. Esto causa que la salida del comparador pase a "high", y la tensión de histéresis en la entrada no inversora pasa al umbral superior, y el C1 comienza a cargarse por medio de R2.

Quando la tensión en C1 excede el umbral superior, la salida del comparador pasa a "low", C1 se descarga hasta una tensión levemente superior a 0.7V, la entrada no inversora es llevada por debajo de los 0.7V, y el circuito se resetea para el pulso siguiente, esperando el siguiente disparo.

Figure 7-1: One-Shot Circuit



Para diseñar un "one-shot", primero se debe crear un realimentador con histéresis usando técnicas del Tip #3. Recuerde de setear el umbral inferior debajo de los 0.7V. Luego, elija valores de R2 y C1 usando la Ecuación 7-1.

Ecuación 7-1

$$T_{PULSE} = \frac{R2 * C1 * \ln(V_{TH}/V_{TL})}{4}$$

D1 puede ser cualquier diodo "switching" de bajo voltaje. R1 debe ser 1% o 2% de R2 y C2 debe estar entre 100 y 220 pF.

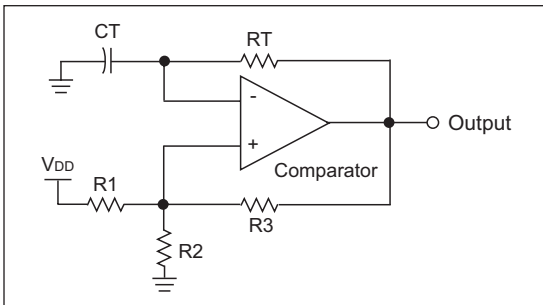
Ejemplo:

- $V_{DD} = 5V$, $V_{TH} = 3.0V$, $V_{TL} = 2.5V$
- Del Tip #3, $R4 = 1k$, $R5 = 1.5k$ y $R3 = 12k$
- $T_{PULSE} = 1ms$, $C1 = .1 \mu F$ and $R2 = 15k$
- D1 es un 1N4148, $R1 = 220\Omega$ y $C2 = 150 pF$

TIP #8 Multi-Vibrator (Square Wave Output)

Un "multi-vibrator" es un oscilador diseñado alrededor de un comparador de tensión o amplificador operacional (vea la Figura 8-1). Las resistencias R1 hasta R3 forman un realimentador con histéresis desde la salida hasta la entrada no inversora. La resistencia RT y el capacitor CT forman un "delay network" entre la salida y la entrada inversora. Al comienzo del ciclo, CT se descarga manteniendo la entrada no inversora a tierra, forzando la salida a "high". Una salida "high" fuerza a la entrada no inversora hacia el umbral superior (ver Tip #3) y carga CT por medio de RT. Cuando el voltaje en CT alcanza el umbral superior, la salida es forzada a "low". Una salida "low" empuja a la entrada no inversora hacia el umbral inferior y CT se descarga por medio de RT. Cuando la tensión en CT alcanza el umbral inferior, la salida es forzada a "high" y comienza el ciclo nuevamente.

Figura 8-1: Circuito Multi-Vibrator



Para diseñar un "multi-vibrator", primero diseñe el realimentador con histéresis usando el procedimiento en el Tip #3. Tenga cuidado al elegir los umbrales (V_{th} y V_{tl}) que son separados igualmente dentro de un rango de modo común del comparador y centrado en $V_{dd}/2$. Luego use V_{dd} y V_{tl} para calcular los valores de R_T y C_T que van a resultar en la frecuencia de oscilación F_{osc} deseada. La Ecuación 8-1 define la relación entre R_T , C_T , V_{th} , V_{tl} y F_{osc} .

Ecuación 8-1

$$F_{OSC} = \frac{1}{2 * R_T * C_T * \ln(V_{TH}/V_{TL})}$$

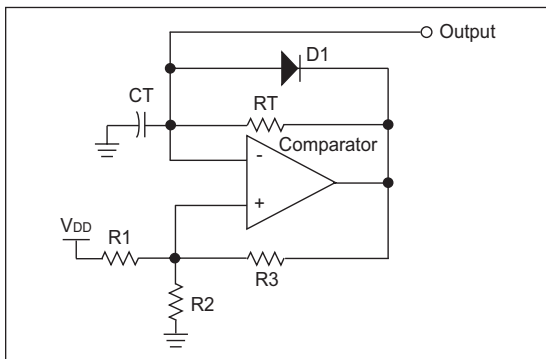
Ejemplo:

- $V_{DD} = 5V$, $V_{TH} = 3.333$, $V_{TL} = 1.666V$
- R_1 , to R_2 , to $R_3 = 10k$
- $R_T = 15\text{ kHz}$, $C_T = .1\ \mu F$ para $F_{osc} = 480\text{ Hz}$

TIP #9 Multi-Vibrator (Ramp Wave Output)

Un "multi-vibrator (ramp wave output) es un oscilador diseñado alrededor de un comparador de tensión o un amplificador operacional que produce una salida asimétrica (Vea la Figura 9-1). Las resistencias R1 hasta R3 forman un realimentador con histéresis desde la salida hasta la entrada no inversora. La resistencia RT, el diodo D1 y el capacitor CT forman un "delay network" entre la salida y la entrada inversora. Al principio del ciclo, CT se descarga manteniendo la entrada no inversora a tierra, forzando la salida a "high". Una salida en "high" fuerza la entrada no inversora hacia el umbral superior (ver Tip #3) y carga CT por medio de RT. Cuando la tensión CT alcanza el umbral superior, la salida se fuerza a "low". Una salida en "low" lleva a la entrada no inversora hacia el umbral inferior y descarga el CT por medio de D1. Debido a que la resistencia dinámica del diodo es significativamente menor que RT, la descarga de CT es pequeña comparada con el tiempo de carga, y el resultado es una pseudo rampa con una fase de carga y una breve fase de descarga.

Figura 9-1: Ramp Waveform Multi-Vibrator



Para diseñar un "multi-vibrator" primero diseñe el realimentador con histéresis usando el procedimiento del Tip #3. Recuerde que la amplitud pico a pico de la rampa va a estar determinada por los límites de la histéresis. También, tenga cuidado al elegir los umbrales de tensión (Vth y Vtl) que son separados igualmente dentro del rango de modo común del comparador.

Luego, utilice Vth y Vtl para calcular los valores de RT y CT que van a resultar en la frecuencia de oscilación Fosc deseada. La Ecuación 9-1 define la relación entre RT, CT, Vth, Vtl y Fosc.

Ecuación 9-1

$$F_{osc} = \frac{1}{RT * CT * \ln(V_{TH}/V_{TL})}$$

Esto asume que la resistencia dinámica de D1 es mucho menor que RT.

Ejemplo:

- VDD = 5V, VTH = 1.666V y VTL = 3.333V
- R1, R2 and R3 = 10k
- RT = 15k, CT = .1 µF para una Fosc = 906 Hz

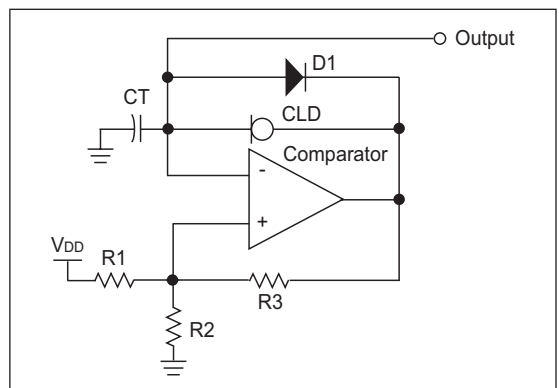
Nota:

Reemplazando Rt con un diodo limitante de corriente va a mejorar significativamente la linealidad de la rampa. Usando el ejemplo anterior, un CCL1000 (1 mA Central Semiconductor CLD), va a producir una salida bastante lineal de 6kHz (ver Ecuación 9-2).

Ecuación 9-2

$$F_{osc} = \frac{I_{CLD}}{C (V_{TH} - V_{TL})}$$

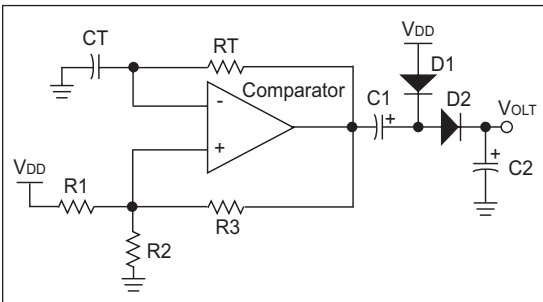
Figura 9-2: Alternate Ramp Waveform Multi-Vibrator Usando Un CLD



TIP #10 Capacitive Voltage Doubler

Este tip toma el "multi-vibrator" descrito en el Tip #8 y construye un duplicador capacitivo de tensión alrededor (ver Figura 10-1). El circuito funciona alternando la carga del capacitor C1 por medio del diodo D1, y después balanceando la carga en C1 y C2 por medio de D2. Al principio del ciclo, la alida del "multi-vibrator" esta en "low" y las corrientes de carga fluyen desde Vdd por medio de D1 hasta C1. Cuando la salida del "multi-vibrator" pasa a "high", D1 está polarizado en inversa, y la corriente de carga de detiene. La tensión en C1 es sumada a la tensión de salida del "multi-vibrator" creando una tensión en el terminal positivo de C1 que es 2 veces Vdd. Esta tensión pone a D2 en directa y la carga en C1 se comparte con C2. Cuando la salida del "multi-vibrator" vuelve a "low" vuelve a comenzar el ciclo

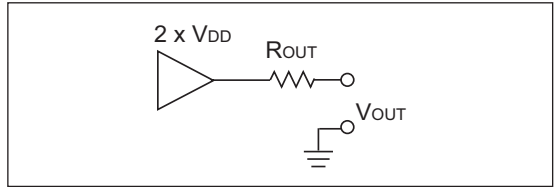
Figure 10-1: Capacitive Voltage Doubler



La tensión de salida de un duplicador capacitivo no es regulado y va a bajar con el incremento de la corriente de carga. Comunemente, la salida es modelada como una fuente de tensión con una serie de resistencias. (Ver Figura 10-2).

Nota

Figure 10-2: Equivalent Output Model



Para diseñar un duplicador de tensión, primero determine la máxima resistencia de salida tolerable basada en la corriente de salida requerida y la tensión de salida mínima tolerable. Recuerde que la corriente de salida va a estar limitada a la mida de la capacidad del comparador. Luego elija una capacitancia de transferencia y frecuencia de "switching" usando la Ecuación 10-1.

Ecuación 10-1

$$R_{OUT} = \frac{1}{F_{SWITCH} * C1}$$

Nota:

Rout va a ser levemente mayor debido a la resistencia dinámica de los diodos. La resistencia equivalente o ESR, de los capacitores y la resistencia de salida del comparador. Vea TC7660 datasheet para más información.

Una vez que la frecuencia de "switching" está determinada, diseñe un "multi-vibrator" de onda cuadrada como en el Tip #8. Finalmente, elija los diodos D1 y D2 para su contribución de corriente y fije C2 igual a C1.

Ejemplo:

Del Tip #8, los valores son modificados a FOSC de 4.8 kHz.

- C1 y C2 = 10 µF
- ROUT = 21Ω

TIP #11 PWM Generator

Este tip muestra como el "multi-vibrador"(ramp wave) puede ser usado para generar una señal PWM controlada por voltaje. El "ramp wave multi-vibrador" opera como se mencionó en el Tip #9, generando una "ramp wave" en sentido positivo. Un segundo comparador compara la tensión instantánea de la "ramp wave" con la tensión de entrada para generar la salida PWM (ver Figura 11-2).

Cuando comienza la rampa, está por debajo de la tensión de entrada, y la salida del segundo comparador es llevado a "high" comenzando así el pulso PWM. La salida se mantiene en "high" hasta que la tensión "ramp wave" exceda la entrada, luego la salida del segundo comparador pasa a "low" terminando así el pulso PWM. La salida del segundo comparador permanece en "low" por el resto de la rampa. Cuando ésta vuelve a cero al comienzo del próximo ciclo, la salida del segundo comparador vuelve a "high" y comienza el ciclo de nuevo.

Figura 11-1: PWM Wave Forms

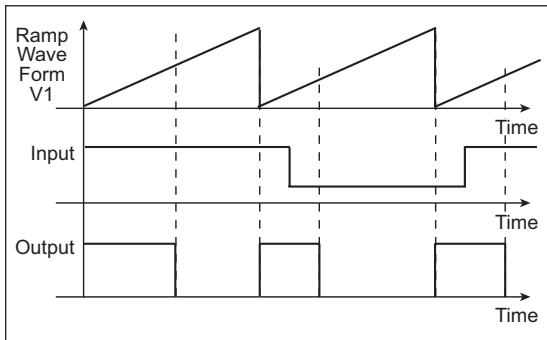
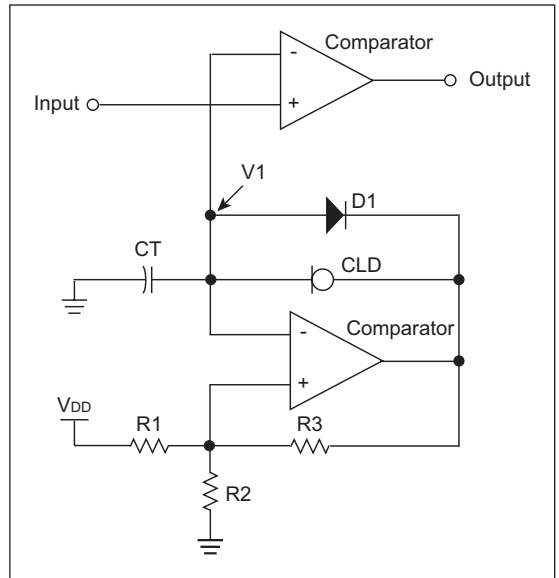


Figura 11-2: Circuito PWM



Para diseñar un generador de PWM, comience con el diseño de la "ramp wave multi-vibrador" usando el procedimiento del Tip#9. Elija umbrales (superior e inferior) para la histéresis del "multi-vibrators" que sean levemente mayor y menor que las tensiones de control de PWM deseadas.

Nota:

La tensión de control del PWM va a producir un 0% de "duty cycle" para entradas menores al umbral inferior del "multi-vibrador". La tensión de control del PWM va a producir un 100 de "duty cycle" para entradas mayores que el umbral superior del "multi-vibrador".

Usando los valores del ejemplo del Tip #9 va a resultar un mínimo ancho de pulso a una tensión de entrada 1.7V y un máximo a una entrada de 3.2V.

TIP #12 Making an Op Amp Out of a Comparator

Cuando se hace una interfaz a un sensor, es común necesitar una cierta ganancia para adaptar el rango completo del sensor al rango del ADC. Generalmente esto se hace usando un amplificador operacional. Sin embargo, en aplicaciones sensibles al costo, incorporar un componente activo puede llegar a exceder el presupuesto. Este tip muestra cómo usar un comparador "on-chip" como un amplificador operacional. Ambas topologías Inversor y No Inversor se muestran en las Figuras 12-1 y 12-2.

Figura 12-1: Amplificador No Inversor

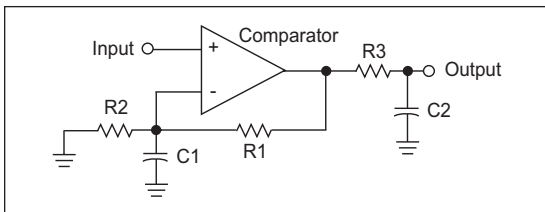
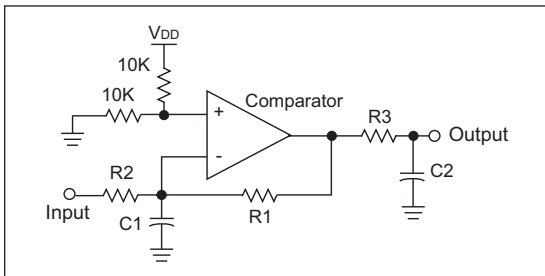


Figura 12-2: Amplificador Inversor



Para diseñar un amplificador no inversor, elija las resistencias R1 y R2 usando la fórmula de ganancia de un AO (ver Ecuación 12-1).

Ecuación 12-1

$$\text{Gain} = \frac{R1 + R2}{R2}$$

Una vez que la ganancia haya sido determinada, se pueden determinar los valores de R3 y C2. Ambos forman un filtro pasa-bajos en la salida del amplificador. La frecuencia tope del filtro debe ser de 2 a 3 veces la frecuencia máxima de la señal para prevenir atenuaciones, y R3 debe ser pequeña para reducir la impedancia de salida del amplificador. La Ecuación 12-2 muestra la relación entre R3, C2 y la frecuencia tope del filtro pasa-bajos.

Ecuación 12-2

$$F_{\text{CORNER}} = \frac{1}{2 * \pi * R3 * C2}$$

Un valor de C1 puede ser determinado usando la Ecuación 12-3. La frecuencia tope debe ser la misma que en la Ecuación 12-3.

Ecuación 12-3

$$F_{\text{CORNER}} = \frac{1}{2 * \pi * (R1 \parallel R2) * C2}$$

Para diseñar un amplificador inversor, elija las resistencias R1 y R2 usando la fórmula de Ganancia de un AO (ver Ecuación 12-4).

Ecuación 12-4

$$\text{Gain} = \frac{R1}{R2}$$

Luego elija valores del divisor resistivo formado por R4 y R5. Finalmente elija C1 y C2 como se muestra en el diseño de un amplificador no inversor.

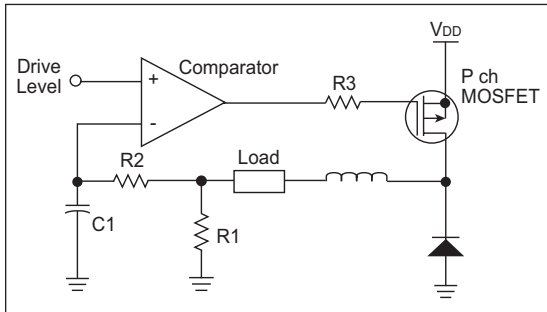
Ejemplo:

- Para C2 va a fijar la frecuencia tope F
- Ganancia = 6.156, R1 = R3 = 19.8k
- R2 = 3.84k, C1 = .047 µF, F_{CORNER} = 171 Hz
- C2 = .22 µF

TIP #13 PWM High-Current Driver

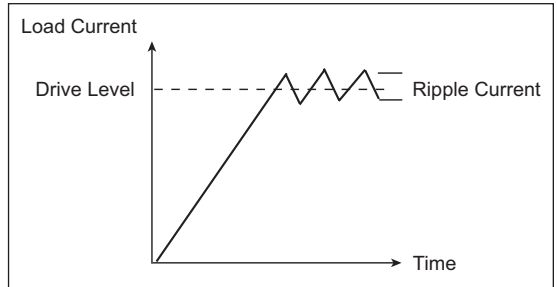
Este tip combina un comparador con un transistor MOSFET y un inductor para crear un circuito "switch mode high-current driver" (Ver Figura 13-1) La operación del circuito comienza con el MOSFET apagado y sin corriente en el inductor y la carga. Con la tensión de sensado en R1 en cero y la tensión de DC presente en la entrada, la salida del coparador pasa a "low". La salida en "low" enciende el MOSFET y comienza a circular una corriente por el transistor, inductor, carga y R1.

Figura 13-1: High Current Driver



Cuando la corriente es lo suficientemente grande como para generar una tensión en R1 igual al nivel del "drive", la salida del comparador pasa a "high" apagando el MOSFET. La voltaje de juntura del MOSFET y del inductor cae hasta que D1 se polariza en directa. La corriente continua decayendo desde su pico hasta hacerse cero. Cuando la tensión en la resistencia R1 cae por debajo del nivel del "drive", la salida del comparador pasa a "low", el MOSFET se enciende, y el ciclo vuelve a comenzar. R2 y C1 forman una "time delay network" que limita la velocidad de "switching" del driver y causa que varíe levemente el nivel del "drive" cuando opera. Este límite es necesario para mantener la velocidad de "switching" baja, de manera que el MOSFET alterne eficientemente. Si R2 y C1 no estuvieran presentes, el sistema va a correr a una velocidad fijada por el retardo del comparador y a la velocidad de "switching" del MOSFET. En esas condiciones, la el tiempo de "switching" del MOSFET sería una parte significativa del tiempo de "switching" total y la eficiencia sería muy baja.

Figura 13-1: Current Through the Load



Para diseñar un driver de alta corriente PWM, primero determine la velocidad de "switching" (Fswx) que sea apropiada para el sistema. Luego, elija un MOSFTE y un D1 capaces de manejar los requerimientos de la corriente de carga. Luego elija los valores R2 y C1 usando la Ecuación 13-1.

Ecuación 13-1

$$F_{SWX} = \frac{2}{R2 * C1}$$

Luego determine el "ripple" máximo que la carga puede tolerar, y calcule la inductancia requerida L1 usando la Ecuación 13-2.

Ecuación 13-2

$$L = \frac{V_{DD} - V_{LOAD}}{I_{RIPPLE} * F_{SWX} * 2}$$

Finalmente, elija el valor de Ri que produzca una tensión de ripple de realimentación de 100mV para la máxima corriente de "ripple" (Iripple).

Ejemplo:

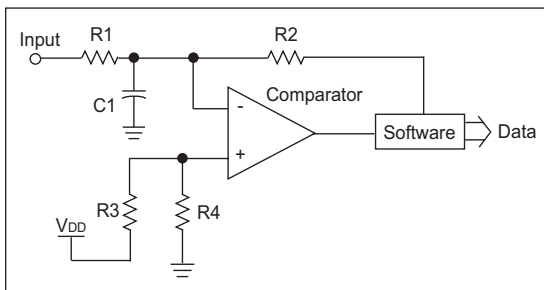
- F_{SWX} = 10 kHz, R2 = 22k, C1 = .01 µF
- I_{RIPPLE} = 100 mA, V_{DD} = 12V, V_L = 3.5V
- L = 4.25 mH

TIP #14 Delta-Sigma ADC

Este tip describe como crear un ADC "Delta-Sigma" basado en hardware/software. Un ADC "Delta-Sigma" está basado en un modulador "Delta-Sigma" compuesto por un integrador, un comparador, un clock de muestreo y una salida de un bit del DAC. En este ejemplo, el integrador esta formado por R1 y C1. EL comparador es un comparador de tensión "on-chip". El clock de muestreo esta implementado por software y la salida de 1 bit del DAC es un simple pin I/O. La salida del DAC es realimentada al integrador por medio de R2.

Las resistencias R3 y R4 forman una referencia de $V_{DD}/2$ para el circuito (ver Figura 14-1).

Figura 14-1: Delta-Sigma Modulator



En operación, la salida del realimentador del software es una copia muestreada de la salida del comparador. En operación normal, la salida del modulador generan una señal PWM que es inversamente proporcional a la tensión de entrada. A medida que ésta aumenta, la señal PWM va a reducir el "duty cycle" para compensar. A medida que la entrada decrece, el "duty cycle" aumenta.

Para hacer una conversión A->D, el "duty cycle" debe ser integrado en el tiempo, digitalmente, para integrarlo a un valor binario. El software comienza dos timers. El primero cuenta el numero total de muestras en la conversión y el segundo cuenta el numero de muestras que estuvieron "low". La razón de los dos contadores es igual a la razón de la tensión de entrada con V_{DD} .

Nota:

Se asume que R1 y R2 son iguales y R3 es igual a R4. Si R1 y R2 no son iguales, la tensión de entrada es escalada por la razón de R2 con R1 y R3 debe seguir siendo igual a R4.

Para una descripción más completa de la operación de un ADC "Delta-Sigma" vea la nota de aplicación AN700 "Make A Delta-Sigma Converter Using a Microcontroller's Analog Comparator Module".

Ejemplo:

- R3 = R4 = 10 kΩ
- R1 = R2 = 5.1k
- C1 = 1000 pF

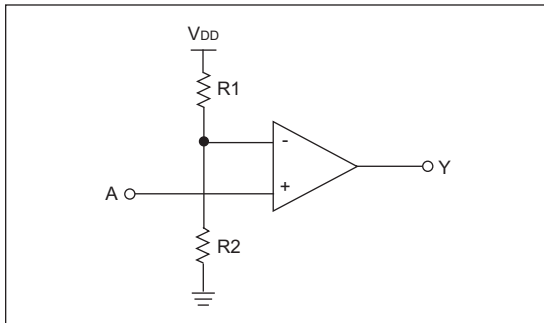
TIP #15 Level Shifter

Este tip muestra como usar un comparador como un "digital logic level shifter". La entrada inversora es polarizada en el centro del rango de la tensión de entrada ($V_{in}/2$). La entrada no inversora es usada como entrada del circuito. Cuando la entrada es menor que el umbral de $V_{in}/2$, la salida es "low". Cuando la entrada está por encima de $V_{in}/2$, la salida es "high". Los valores de $R1$ y $R2$ no son críticos, aunque la razón de ambos debe resultar en el umbral de tensión $V_{in}/2$ en el punto medio del rango de la señal de entrada. Algunos microcontroladores tienen la opción de conectar la entrada inversora a una referencia interna. Para usar la referencia interna, simplemente seleccione la referencia interna y configúrela a un medio del rango de la tensión de entrada.

Nota:

El típico retardo de propagación del circuito es de 250-350 ns usando el comparador on-chip típico de un microcontrolador.

Figura 15-1: Level Shifter



Ejemplo:

- $V_{IN} = 0 - 2V$, $V_{IN}/2 = 1V$, $V_{DD} = 5V$
- $R2 = 10k$, $R3 = 3.9k$

TIP #16 Logic: Inverter

Cuando diseñe aplicaciones de control, es común necesitar de una compuerta externa. Usando el comparador, varias compuertas simples pueden ser implementadas. Este tip muestra como usar el comparador como un inverter.

La entrada no inversora es polarizada en el centro del rango de la tensión de entrada, típicamente $V_{dd}/2$. La entrada inversora es usada como entrada al circuito. Cuando la entrada está por debajo de $V_{dd}/2$, la salida está en "high". Cuando la entrada está por encima de $V_{dd}/2$, luego la salida es "low".

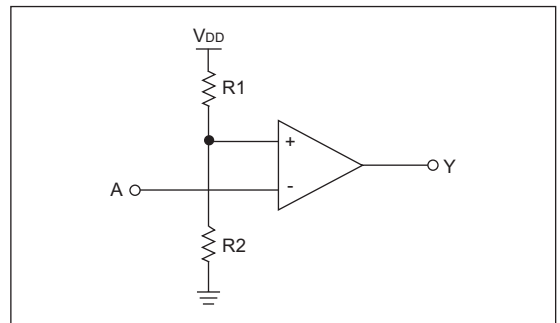
Los valores de $R1$ y $R2$ no son críticos, aunque deben ser iguales para para fijar el umbral en $V_{dd}/2$.

Algunos microcontroladores tienen la opción de conectar la entrada inversora a una referencia interna. Para usar esta referencia interna, mueva la entrada hacia la entrada no inversora y setee el bit de polaridad de salida en el registro del control de comparador para invertir la salida del comparador.

Nota:

El típico retardo de propagación del circuito es de 250-350 ns usando el comparador on-chip típico de un microcontrolador.

Figura 16-1: Inversor



TIP #17 Logic: AND/NAND Gate

Este tipo muestra como usar un comparador para implementar una compuerta AND is su complemento NAND (ver Figura 17-2). Las resistencias R1 y R2 llevan a la entrada no inversora a 2/3 de la tensión. Las resistencias R3 y R4 promedian la tensión de entrada A y B en la entrada inversora. Si A o B están en "low", el promedio va a ser la mitad de V_{DD} y la salida del comparador va a permanecer "low". La salida va pasar a "high" solo si las dos entradas A y B están en "high", lo cual lleva la entrada inversora a 2/3 de de V_{DD}.

La operación de la compuerta NAND es idéntica a la compuerta AND, excepto que la salida es invertida debido al intercambio de las entradas inversora y no inversora.

Nota:

El típico retardo de propagación del circuito es de 250-350 ns usando el comparador on-chip típico de un microcontrolador. Los tiempos de retardo fueron medidos con una resistecia de 10k.

A pesar que el circuito es bastante simple, hay ciertos requerimientos que se deben tener en cuenta para un funcionamiento correcto:

1. Las entradas A y B deben pasar de "ground" a a V_{DD} para que el circuito funcione bien.
2. La combinación de R1 y R2 van a inducir una corriente constante, por lo tanto deben ser grandes para minimizar la corriente.
3. Todas las resistencias de la entrada inversora actúan con la capacitancia de entrada del comparador. Entonces, la velocidad de la compuerta va a ser afectada por la resistencia de la fuente A y B y por el valor de R3 y R4.
4. La resistencia R2 debe ser 2 x R1.
5. La resistencia R3 debe ser igual a R4.

Figura 17-1: Compuerta AND

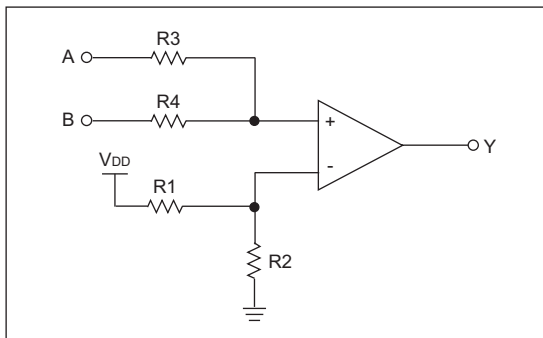
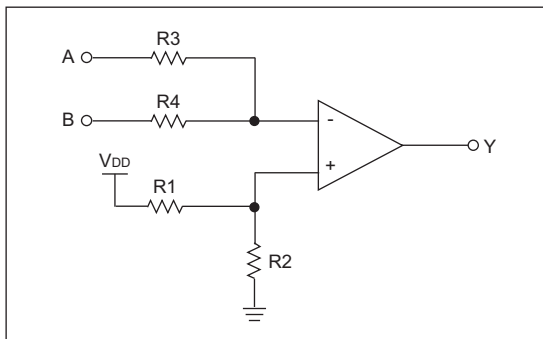


Figura 17-2: Compuerta NAND



Ejemplo:

- V_{DD} = 5V, R3 = R4 = 10k
- R1 = 5.1k, R2 = 10k

TIP #18 Logic: OR/NOR Gate

Este tip muestra como utilizar un comparador para implementar una compuerta OR, y su complemento, NOR.

Las resistencias R1 y R2 llevan a la entrada no inversora del comparador a 1/3 de Vdd. Las resistencias R3 y R4 promedian las tensiones de las entradas A y B en la entrada inversora. Si A o B están en "high", el promedio es 1/2 Vdd y la salida del comparador es "high". Sólo si A y B están en "low" el promedio en la entrada no inversora cae por debajo de 1/3 de la tensión, causando que la salida del comparador pase a "low". La operación de la compuerta NOR es idéntica a la compuerta OR, excepto que la salida es invertida debido al intercambio de la entradas inversora y no inversora.

Nota:

El típico retardo de propagación del circuito es de 250-350 ns usando el comparador on-chip típico de un microcontrolador. Los tiempos de retardo fueron medidos con una resistecia de 10k.

A pesar que el circuito es bastante simple, hay ciertos requerimientos que se deben tener en cuenta para un funcionamiento correcto:

1. Las entradas A y B deben pasar de "ground" a a VDD para que el circuito funcione bien.
2. La combinación de R1 y R2 van a inducir una corriente constante, por lo tanto deben ser grandes para minimizar la corriente.
3. Todas las resistencias de la entrada inversora actúan con la capacitancia de entrada del comparador. Entonces, la velocidad de la compuerta va a ser afectada por la resistencia de la fuente A y B y por el valor de R3 y R4.
4. La resistencia R2 debe ser 2 x R1.
5. La resistencia R3 debe ser igual a R4.

Figura 18-1: Compuerta OR

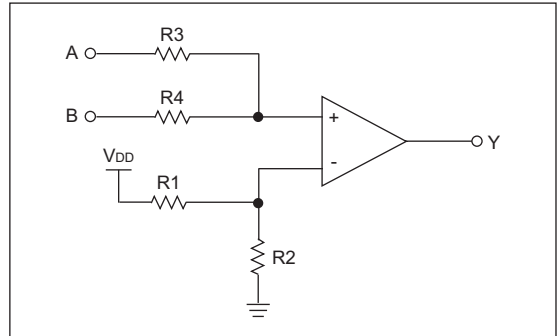
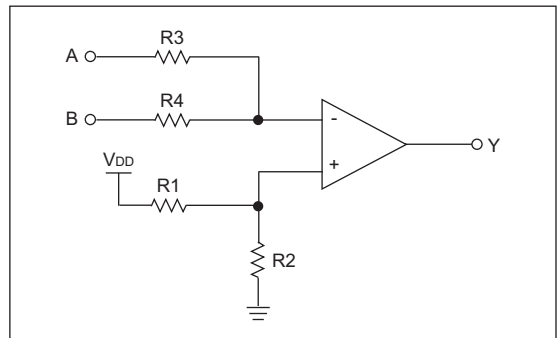


Figura 18-2: Compuerta NOR



Ejemplo:

- VDD = 5V, R3 = R4 = 10k
- R1 = 10k, R2 = 5.1k

TIP #19 Logic: XOR/XNOR Gate

Este tipo muestra como usar un comparador para implementar una compuerta XOR y su complemento XNOR.

La operación se describe mejor en tres pasos:

1. Las dos entradas A y B están en "low". Si los dos están en "low", la entrada inversora se mantiene en $.7V$ y la no inversora se mantiene en tierra. Esta combinación resulta en una salida "low".
2. Las dos entradas A y B están en "high". Si los dos están en "high", la entrada inversora es llevada a VDD y la entrada no inversora es igual a $2/3 VDD$ (el promedio de VDD y GND). Esta combinación resulta en una salida "low".
3. La entrada A o B está en "high". Con una de las entradas en "high" y la otra en "low", la entrada inversora se mantiene en $.7V$ y la entrada no inversora es igual a $1/3VDD$ (el promedio de VDD y GND). Esta combinación resulta una salida "high".

Nota:

El típico retardo de propagación del circuito es de 250-350 ns usando el comparador on-chip típico de un microcontrolador. Los tiempos de retardo fueron medidos con una resistencia de 10k.

A pesar de que el circuito es bastante simple, se deben tener en cuenta ciertos requerimientos para su funcionamiento correcto:

1. Las entradas A y B deben ir de "ground" a VDD para que el circuito funcione bien.
2. Todas las resistencias en ambas entradas actúan con la capacitancia de entrada del comparador, por consiguiente, la velocidad de la compuerta va a estar afectada por la resistencia de la fuente de A y B, y por los valores de R1, R2, R3 y R4.
3. La resistencia R1, R2 y R3 deben ser iguales.
4. La resistencia R4 debe ser lo suficientemente pequeña para producir una caída de 1.0V o menor en D1 y D2.

Figura 19-1: Compuerta XOR

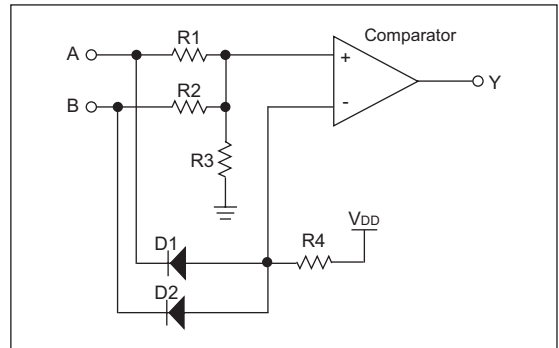
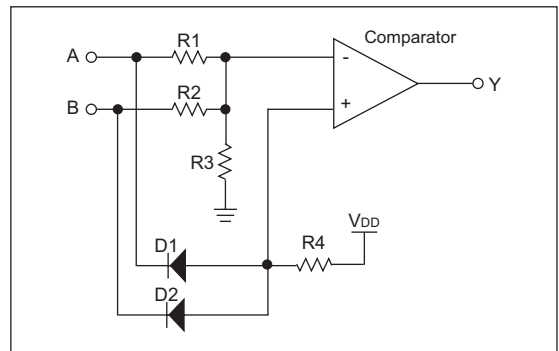


Figura 19-2: Compuerta XNOR



Ejemplo:

- D1 = D2, = 1N4148
- R4 = 10k, R1 = R2 = R3 = 5.1k

TIP #20 Logic: Set/Reset Flip Flop

Este tip muestra como usar un comparador para implementar un "Set/Reset Flip Flop". La entrada inversora y la no inversora son polarizadas en $V_{DD}/2$ por las resistencias R1 hasta R4. La entrada no inversora tambien reciben realimentación positiva de la salida por medio de R5. Las tensiones comunes de polarización y la realimentación positiva transforman al comparador en un latch biestable. Si la salida Q está en "high", la entrada no inversora también es llevada a "high", lo cual refuerza la salida "high". Si Q está en "low", la entrada no inversora tambien es lleva a "low", lo cual refuerza la salida "low". Para cambiar el estado, la entrada apropiada debe ser llevada a "low" para vencer la realimentación positiva. Los diodos previenen que un estado positivo en cualquiera de las entradas lleve la polarización por encima de $V_{DD}/2$.

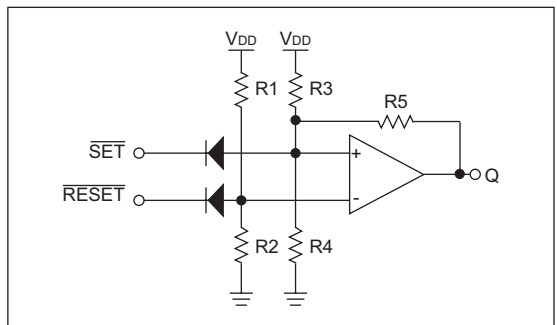
Nota:

El típico retardo de propagación del circuito es de 250-350 ns usando el comparador on-chip típico de un microcontrolador. Los tiempos de ratardo fueron medidos con una resistecia de 10k.

A pesar que el circuito es bastante simple, hay una serie de requerimientos que deben ser tenidos en cuenta para un funcionamiento correcto:

1. Las entradas Set y Reste deben ser llevados cerca de tierra para que el circuito funcione correctamente.
2. La combinación de R1/R2 y R3/R4 van a inducir una corriente constante, por lo que deben ser de valores pequeños para mantener la corriente baja.
3. R1 hasta R4 deben ser iguales para un "trip level" de $V_{DD}/2$.
4. R5 debe ser mayor o igual que R3.
5. R1 hasta R4 van a actuar con la capacitancia de entrada del comparador, por lo que mayores valores van a limitar el mínimo ancho de pulso de entrada.

Figura 20-1: Set/Reset Flip Flop



Ejemplo:

- Diodes = 1N4148
- R1 = R2 = R3 = R4 = 10k
- R5 = 10k

CAPITULO 5

Control de Motores en DC

Soluciones

Tabla de Contenidos

INTRODUCCION

TIP #1:	Brushed DC Motor Drive Circuits	5-2
TIP #2:	Brushless DC Motor Drive Circuits	5-3
TIP #3:	Stepper Motor Drive Circuits	5-4
TIP #4:	Drive Software	5-6
TIP #5:	Writing a PWM Value to the CCP Registers with a Mid-Range PIC® MCU	5-7
TIP #6:	Current Sensing	5-8
TIP #7:	Position/Speed Sensing	5-9
Application Note References		5-11
Motor Control Development Tools		5-11

INTRODUCCIÓN

Cada circuito para controlar un motor puede ser dividido en la parte electrónica y el software controlador. Estas dos partes pueden ser muy simples o muy complicadas dependiendo del tipo de motor, los requerimientos del sistema y la complejidad del hardware/software.

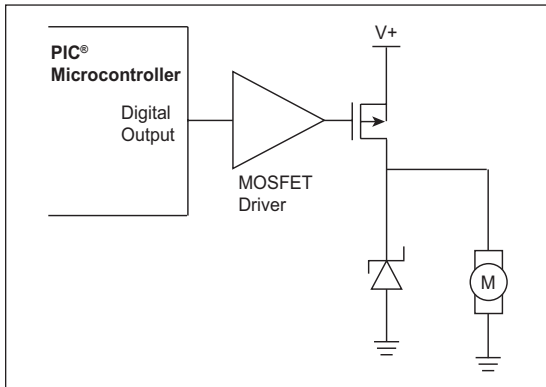
Generalmente, sistemas de mejor performance requieren de hardware más complicado. Este folleto describe varios circuitos básicos y bloques de software usados comunmente para controlar motores. También se provee referencias a notas de aplicación de Microchip que describen los conceptos con mayor detalle. Las notas de aplicación se pueden encontrar en el sitio web de Microchip: www.microchip.com.

Información adicional del diseño de un controlador de motor puede encontrarse en: www.microchip.com/motor.

TIP #1 Brushed DC Motor Drive Circuits

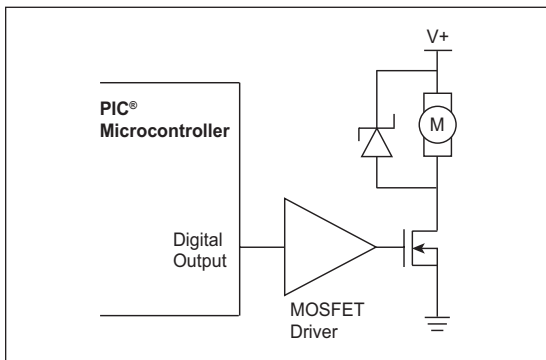
Todos los motores requieren un "drive" para controlar la corriente que fluye por los bobinados del motor. Esto incluye la dirección y la magnitud de la corriente. El motor más simple es el "Brushed DC". Los circuitos se muestran debajo.

Figura 1-1: High Side Drive



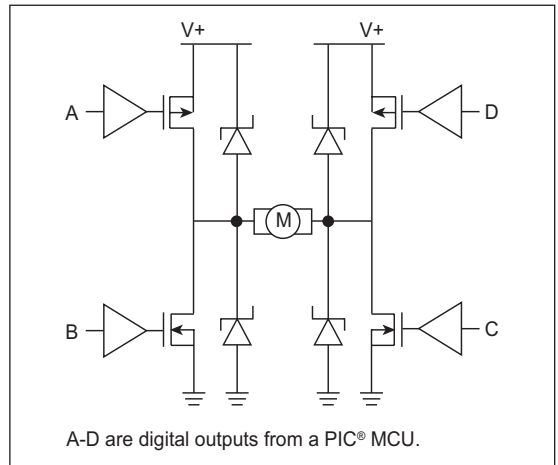
Este "drive" puede controlar un motor "Brushed DC" en una dirección. Es usado frecuentemente en aplicaciones críticas y seguras ya que un corto en los terminales del motor no pueden encender el motor.

Figura 1-2: Low Side Drive



Este es el "drive" de menor costo ya que el MOSFET lo hace simple. La mayoría de las aplicaciones simplemente usar un pin de salida del microcontrolador para encender el MOSFET.

Figura 1-3: H-Bridge Drive



El "H-Bridge" recibe ese nombre por la manera común de dibujar el circuito. Esta es la única manera de mantener en estado sólido de operar un motor en ambas direcciones. Las notas de aplicación de motores "Brushed DC" se nombran debajo y pueden ser encontradas en la web de Microchip: www.microchip.com

- AN847, "RC Model Aircraft Motor Control" (DS00847)
- AN893, "Low-cost Bidirectional Brushed DC Motor Control Using the PIC16F684" (DS00893)
- AN905, "Brushed DC Motor Fundamentals" (DS00905)

Un motor "Brushless DC" es un buen ejemplo de hardware simplificado incrementando la complejidad de control. El motor no puede

conmutar los embobinados, por lo que el circuito de control y el software deben controlar el flujo de corriente de manera pareja. El circuito es un simple "half-bridge" en cada uno de los tres embobinados del motor.

Hay dos métodos básicos de conmutación para motores "Brushless DC"; con sensores o sin sensores. Como es crítico saber la posición del motor para energizar la bobina correcta, algún método de detección de la posición del motor es requerido. Un motor con sensores va a reportar directamente su posición al controlador. Usar un motor con sensores requiere de una tabla. La posición actual se correlaciona directamente con un patrón de conmutación del circuito.

Cuando no se dispone de sensores se debe usar otra propiedad del motor para sensar su posición. Un método popular es medir la fuerza electromotriz que es generada naturalmente por los imanes y bobinas del motor. El voltaje inducido en la bobina no utilizada puede ser sensada y usada para determinar la velocidad actual del motor. Luego, el siguiente patrón de conmutación puede ser determinado con un retaso del patrón anterior.

Los motores sin sensor obviamente tienen un menor costo, pero son más complicados de manejar. Este tipo de motores funcionan bien en aplicaciones en donde no se requiere que el motor frene y arranque. Un motor con sensores es más adecuado para aplicaciones en donde se requiere que el motor frene periódicamente.

Figura 2-1: 3 Phase Brushless DC Motor Control

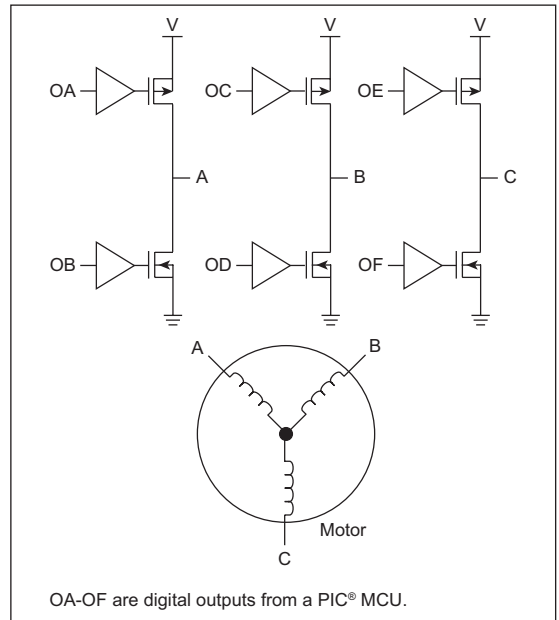


Figura 2-2: Back EMF Sensing (Sensorless Motor)

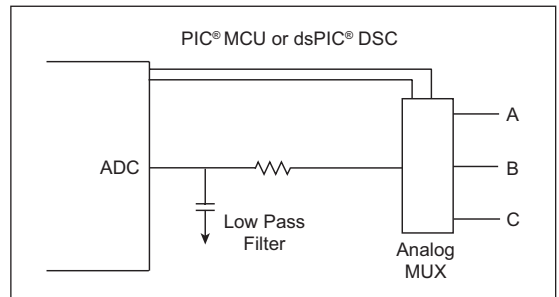
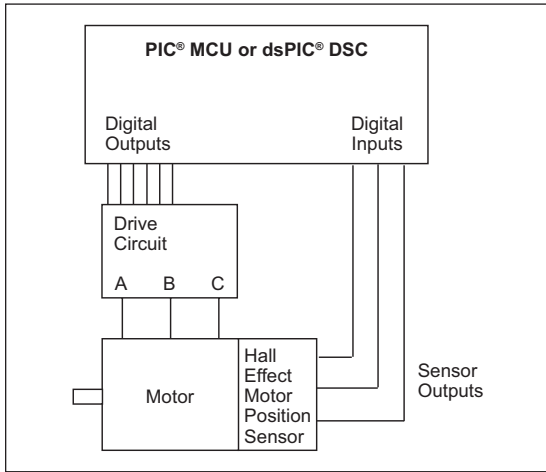


Figura 2-3: Quadrature Decoder (Sensor Motor) TIP #3 Stepper Motor Drive Circuits



A continuación se mencionan las notas de aplicación que describen el control de motores "Brushless DC" y se pueden encontrar en la web de Microchip: www.microchip.com

- AN857, "Brushless DC Motor Control Made Easy" (DS00857)
- AN885, "Brushless DC Motor Fundamentals" (DS00885)
- AN899, "Brushless DC Motor Control Using PIC18FXX31" (DS00899)
- AN901, "Using the dsPIC30F for Sensorless BLDC Control" (DS00901)
- AN957, "Sensored BLDC Motor Control Using dsPIC30F2010" (DS00957)
- AN992, "Sensorless BLDC Motor Control Using dsPIC30F2010" (DS00992)
- AN1017, "Sinusoidal Control of PMSM with dsPIC30F DSC" (DS01017)
- GS005, "Using the dsPIC30F Sensorless Motor Tuning Interface" (DS93005)

Los motores paso a paso son similares a los "Brushless DC" en que el sistema de control debe conmutar el motor por medio de una rotación de ciclo completa. A diferencia del motor "brushless", la posición y la velocidad de un motor paso a paso es predecible y no requiere del uso de sensores.

Hay dos tipos básicos de motores paso a paso, a pesar de que algunos motores son contruidos para funcionar en cualquiera de los modos. El motor paso a paso más simple es el de motor unipolar. Este motor tiene cuatro conexiones y uno o dos cables que son atados a tierra o a V, dependiendo de la implementación. Otro tipo de motores son los bipolares o varias combinaciones de unipolares con bipolares como se muestra en las Figuras 3-1 y 3-2. Cuando cada una de las conexiones se energiza, se impulsa una bobina y el motor rota un paso. El proceso se repite hasta que todas las bobinas hayan sido energizadas. Para incrementar la tasa de pasos, frecuentemente se incrementa la tensión. Si se incrementa la tensión, se requiere de algún método para prevenir una sobrecarga de corriente.

Hay muchas formas de controlar la corriente en las bobinas, pero la más popular es un sistema "chopper" que corta la corriente cuando se llega a un límite superior y permite a la corriente circular más tarde. Sistemas de sensores de corriente se discuten en el Tip #6. Algunos sistemas son contruidos con un "chopper" de corriente, pero no detectan la corriente, sino que el sistema es diseñado para empezar un período fijo de "chopping" luego de que el motor haya pasado al paso siguiente. Estos son sistemas fáciles de construir ya que sólo requieren de un cambio en el software.

Figura 3-1: 4 and 5 Wire Stepper Motors

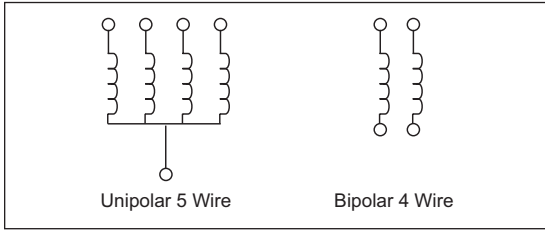


Figura 3-2: 6 and 8 Wire Stepper Motors

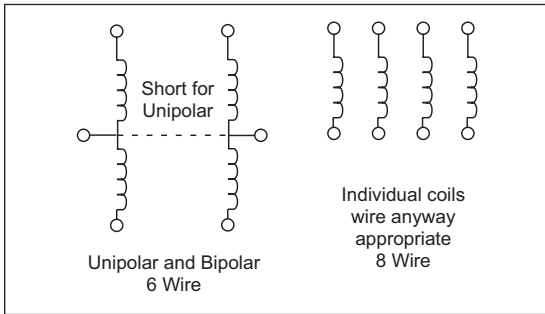


Figura 3-3: Unipolar Motor (4 Low Side Switches)

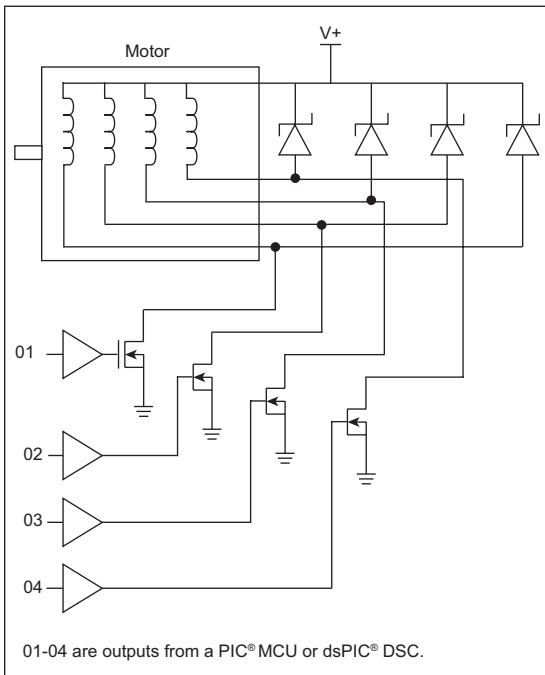
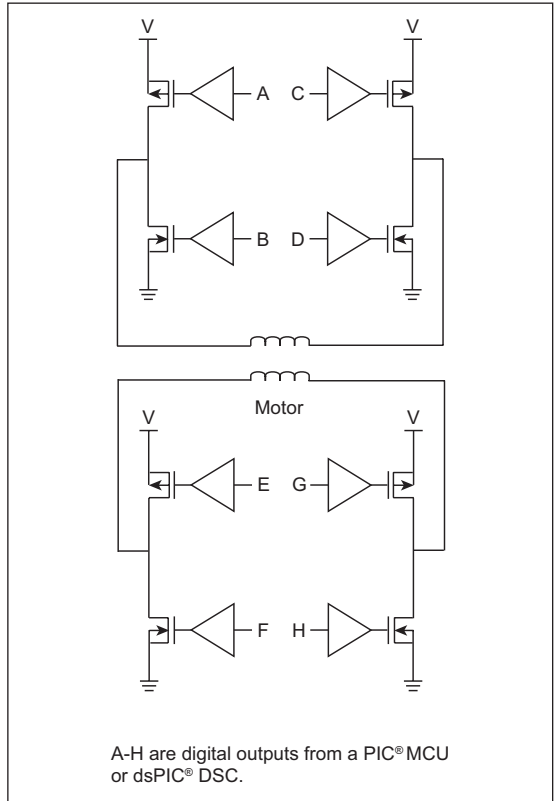


Figura 3-4: Motor Bipolar (4 Half-Bridges)



TIP #4 Drive Software

Algoritmos Pulse-Width Modulation (PWM)

La modulación PWM es crítica para el control de motores digitales modernos. Ajustando el ancho del pulso, la velocidad del motor puede ser controlada eficientemente sin grandes etapas de potencia lineal. Algunos dispositivos PIC y todos los dsPIC DSCs tienen los módulos PWM. Estos módulos son contruidos en el periférico "Capture/Compare PWM"(CCP). Estos periféricos están hechos para una salida PWM, mientras que los "Enhanced CCP" (ECCP) están diseñados para producir una salida "H-Bridge" para el control direccional de motores "Brushed DC". Si el costo es crítico en el diseño, un PIC con módulo CCP puede no estar disponible, entonces el PWM generado por software aparece como una buena alternativa.

Los siguientes algoritmos son diseñados para producir eficientemente una salida de 8 bits PWM en la familia "Mid-Range" de los microcontroladores PIC. Estos algoritmos son implementados como macros. Si quiere que estos macros sean una subrutina de su programa, simplemente remueva las declaraciones de macro y cámbielos por una etiqueta y un "return".

Ejemplo 4-1: 1 Salida 8-Bit PWM

```
pwm_counter equ xxx ;variable
pwm          equ xxx ;variable

set_pwm macro A          ;sets the pwm
                  ;setpoint to the
                  ;value A

    MOVLW A
    MOVWF pwm
endm

update_PWM macro        ;performs one update
                  ;of the PWM signal
                  ;place the PWM output
                  ;pin at bit 0 or 7 of
                  ;the port

    MOVF pwm_counter,w
    SUBWF pwm, w        ;if the output
                  ;is on bit 0
    RLF          PORTC,f ;replace PORTC with
                  ;the correct port if
                  ;the output is on bit
                  ;7 of the port
                  ;replace the rlf with
                  ;rrf incf
                  ;pwm_counter,f
```

Ejemplo 4-2: 8 Salidas 8-Bit PWM

```
pwm_counter equ xxx      ;variable
pwm0        equ xxx      ;
pwm1        equ pwm0+1
pwm2        equ pwm1+1
pwm3        equ pwm2+1
pwm4        equ pwm3+1
pwm5        equ pwm4+1
pwm6        equ pwm5+1
pwm7        equ pwm6+1
output      equ pwm7+1

set_pwm macro A,b        ;sets pwm b with
                  ;the value A

    MOVLW pwm0
    ADDLW b
    MOVWF fsr
    MOVLW a
    MOVWF indf
endm

update_PWM macro        ;performs one
                  ;update of all 8
                  ;PWM signals
                  ;all PWM signals
                  ;must be on the
                  ;same port

    MOVF          pwm_counter,w
    SUBWF         pwm0,w
    RLF          output,f
    MOVF          pwm_counter,w
    SUBWF         pwm1,w
    RLF          output,f
    MOVF          pwm_counter,w
    SUBWF         pwm2,w
    RLF          output,f
    MOVF          pwm_counter,w
    SUBWF         pwm3,w
    RLF          output,f
    MOVF          pwm_counter,w
    SUBWF         pwm4,w
    RLF          output,f
    MOVF          pwm_counter,w
    SUBWF         pwm5,w
    RLF          output,f
    MOVF          pwm_counter,w
    SUBWF         pwm6,w
    RLF          output,f
    MOVF          pwm_counter,w
    SUBWF         pwm7,w
    RLF          output,w
    MOVWF        PORTC
    INCF          pwm_counter,f
endm
```


TIP #5 Writing a PWM Value to the the CCP Registers With a Mid-Range PIC® Microcontroller

Los dos "PWM LSb" se ubican en el registro CCPCON del CCP. Esto puede hacer que el cambio de período del PWM sea frustrante para un diseñador. El Ejemplo 5-1 hasta el Ejemplo 5-3 muestran 3 macros escritos para una familia de "mid-range" que pueden ser usados para setear el período del PWM. El primer macro lleva un valor de 16-bits y usa los 10 MSb's para setear el período de PWM. El segundo macro lleva un valor de 16 bits y usa los 10 LSb's para setear el período de PWM. El último macro lleva un valor de 8 bits y setea el período de PWM. Esto asume que el CCP este configurado para no más de 8 bits.

Ejemplo 5-1: Left Justified 16-Bit Macro

```
pwm_tmp    equ xxx    ;this variable must be
                ;allocated someplace
setPeriod  macro a    ;a is 2 SFR's in
                ;Low:High arrangement
                ;the 10 MSb's are the
                ;desired PWM value
    RRF      a,w      ;This macro will
                ;change w
    MOVWF   pwm_tmp
    RRF     pwm_tmp,w
    ANDLW  0x30
    IORLW  0x0F
    MOVWF  CCP1CON
    MOVF   a+1,w
    MOVWF  CCP1L
```

Ejemplo 5-2: Right Justified 16-Bit Macro

```
pwm_tmp    equ xxx    ;this variable must be
                ;allocated someplace
setPeriod  macro a    ;a is 2 bytes in
                ;Low:High arrangement
                ;the 10 LSb's are the
                ;desired PWM value
    SWAPF  a,w      ;This macro will
                ;change w
    ANDLW  0x30
    IORLW  0x0F
    MOVWF  CCP1CON
    RLF    a,w
    IORLW  0x0F
    MOVWF  pwm_tmp
    RRF    pwm_tmp,f
    RRF    pwm_tmp,w
    MOVWF  CCP1L
```

Ejemplo 5-3: 8-Bit Macro

```
pwm_tmp    equ xxx    ;this variable must be
                ;allocated someplace
setPeriod  macro a    ;a is 1 SFR
    SWAPF  a,w      ;This macro will
                ;change w
    ANDLW  0x30
    IORLW  0x0F
    MOVWF  CCP1CON
    RRF    a,w
    MOVWF  pwm_tmp
    RRF    pwm_tmp,w
    MOVWF  CCP1L
```

TIP #6 Current Sensing

El torque de un motor eléctrico puede ser monitoreado y controlado siguiendo de cerca la corriente que fluye por el motor. El torque es directamente proporcional a la corriente. La corriente puede ser sensada por medio de la medición de la caída de tensión en una resistencia conocida o midiendo el campo magnético de un inductor conocido. La corriente es sensada generalmente en uno de los siguientes dos lugares, el lado de suministro del circuito "drive" ("high side current sense") o el lado sumidero del circuito "drive" ("low side current sense"). El "Low side sense" es mucho más simple pero el motor no es vuelto a poner en tierra, causando una cuestión de seguridad en algunas aplicaciones. El "High side sense" requiere un amplificador diferencial con un rango de modo común entre el voltaje de suministro.

Figura 6-1: Resistive High Side Current Sensing

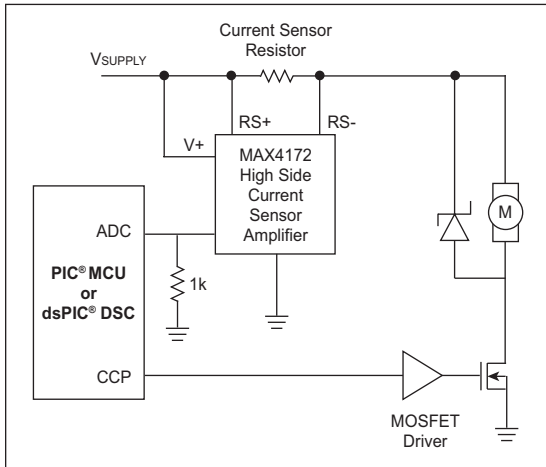
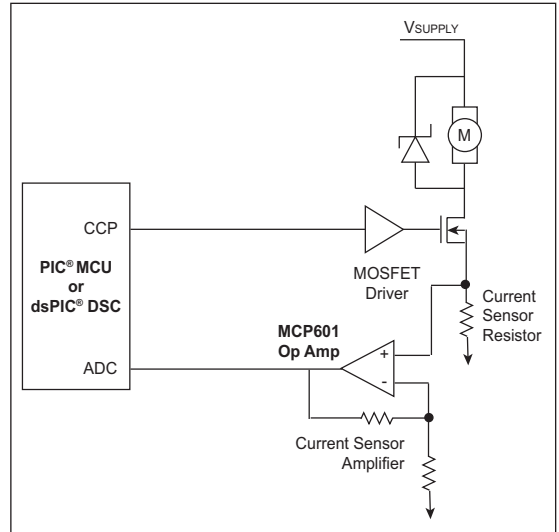
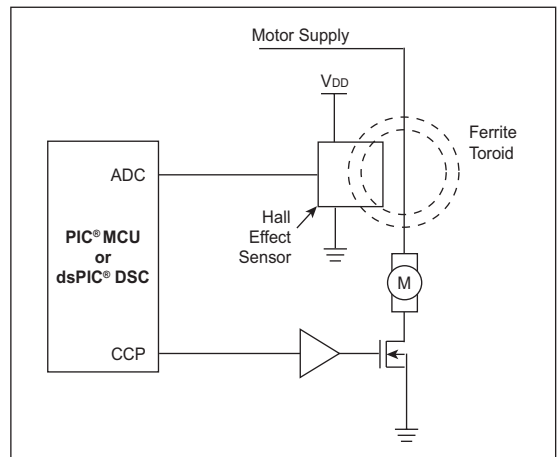


Figura 6-2: Resistive Low Side Current Sensing



Mediciones de corriente pueden ser llevadas a cabo usando sensores de efecto Hall para medir el campo magnético que rodea un cable. Naturalmente, este sensor puede ser ubicado en el "high side" o en el "low side" de la carga. La ubicación actual del sensor no importa porque no se confía en la tensión del cable. Este es un método no intrusivo que puede ser usado para medir la corriente del motor.

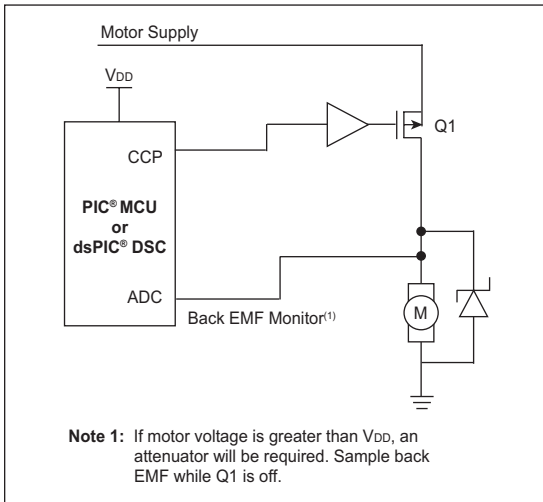
Figura 6-3: Magnetic Current Sensing



TIP #7 Position/Speed Sensing

El RPM de un motor puede ser medido entendiéndolo que el motor es un generador. Siempre que el motor esté girando, va a producir una tensión que es proporcional al RPM del motor. Esto es llamado EMF. Si la señal PWM del motor es apagada y se mide la tensión en las bobinas, la EMF puede ser sensada y de ahí se puede conocer el RPM.

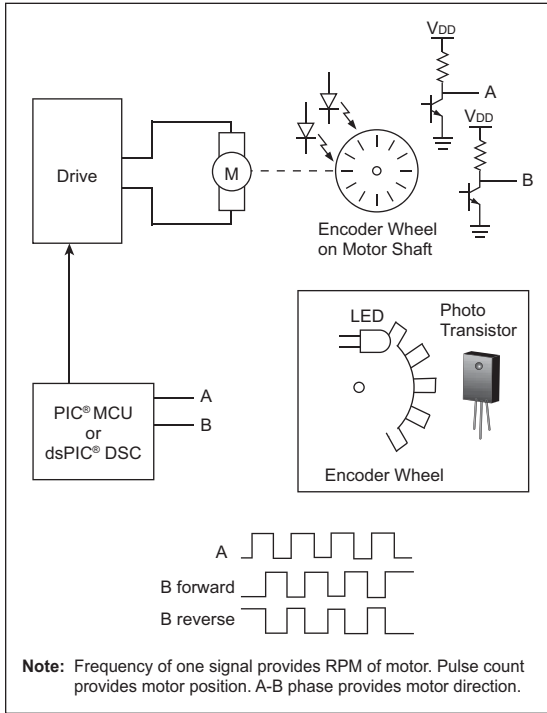
Figura 7-1: Back EMF Motor Speed Sensing



Rotary Encoder Sensing

Los "Rotary Encoders" son usados típicamente para proveer un feedback físico de la posición del motor y/o velocidad. Un "rotary encoder" consiste en un elemento giratorio ligado al motor que tiene una característica física, medida por un componente estacionario. Estas mediciones pueden entregar la velocidad del motor y a veces la posición del motor. "Rotary encoders" son construídos usando diferentes tecnologías. El más común es el óptico que es usado en los "mouse" de computadora que tienen una bolita. Está hecho con una disco encoder que está ligado al motor. El disco encoder tiene varias ranuras separadas en intervalos específicos. Un LED y un fotodetector son usados para contar las ranuras que pasan. Midiendo el tiempo y la cantidad de ranuras se determina la velocidad. Sensar la posición del motor requiere de un segundo LED y fotodetector. Este par se monta de manera que los pulsos de salida sean 90° defasados del primer par. Las dos salidas representan el movimiento del disco encoder como un tren de pulsos modulados en cuadratura. Agregando una tercera señal, se puede saber la posición exacta del rotor. Un encoder con salidas en cuadratura puede ser usado para saber la posición relativa desde un punto de referencia. Otro tipo de encoders usan un disco codificado binariamente de manera tal que la posición del rotor es conocida en todo momento. Este tipo de encoders es conocido como "absolute encoder".

Figura 7-2: Optical Speed/Direction/Position Sensing



El sensado en cuadratura puede ser llevado a cabo fácilmente por software, pero generalmente hay un límite superior para el RPM. Usando un par de compuertas, el sensado puede ser hecho parcialmente por hardware y parcialmente por software. El nuevo PIC18FXX31 y la familia de los dsPIC 16-bit Digital Signal Controller incluyen una interface encoder que permite un límite muchísimo mayor para RPM y que pueden ser medidos con una excelente precisión.

Métodos Antiguos de Sensado de Motor

Los tacómetros analógicos y los "resolvers" son dos de los métodos más antiguos para medir la posición/velocidad del motor. El tacómetro es simplemente un generador eléctrico con una salida lineal sobre un rango específico de RPMs. Sabiendo las características de la salida, el RPM puede saberse midiendo la tensión en los terminales del tacómetro.

El "resolver " es un par de bobinas que son excitadas por una señal AC externa. Las dos bobinas están a 90° una de otra de manera que levantan la señal de AC a diferentes fuerzas, dependiendo de la orientación. El resultado es una salida seno o coseno relacionada con el ángulo del "resolver" en referencia a la señal AC. La inversa del seno/coseno produce un ángulo del sensor. Este tipo de sensores pueden ser muy precisos y siguen siendo usados donde se debe conocer la posición absoluta.

Referencias de notas de aplicación

- AN532, “*Servo Control of a DC Brush Motor*” (DS00532)
- AN696, “*PIC18CXXX/PIC16CXXX DC Servomotor*” (DS00696)
- AN718, “*Brush-DC Servomotor Implementation using PIC17C756A*” (DS00718)
- AN822, “*Stepper Motor Microstepping with the PIC18C452*” (DS00822)
- AN843, “*Speed Control of 3-Phase Induction Motor Using PIC18 Microcontrollers*” (DS00843)
- AN847, “*RC Model Aircraft Motor Control*” (DS00847)
- AN857, “*Brushless DC Motor Control Made Easy*” (DS00857)
- AN885, “*Brushless DC (BLDC) Motor Fundamentals*” (DS00885)
- AN899, “*Brushless DC Motor Control Using the PIC18FXX31*” (DS00899)
- AN893, “*Low-cost Bidirectional Brushed DC Motor Control Using the PIC16F684*” (DS00893)
- AN894, “*Motor Control Sensor Feedback Circuits*” (DS00894)
- AN898, “*Determining MOSFET Driver Needs for Motor Drive Applications*” (DS00898)
- AN901, “*Using the dsPIC30F for Sensorless BLDC Control*” (DS00901)
- AN905, “*Brushed DC Motor Fundamentals*” (DS00905)
- AN906, “*Stepper Motor Control Using the PIC16F684*” (DS00906)
- AN907, “*Stepper Motor Fundamentals*” (DS00907)
- AN1017, “*Sinusoidal Control of PMSM Motors with dsPIC30F DSC*” (DS01017)
- GS001, “*Getting Started with BLDC Motors and dsPIC30F Devices*” (DS93001)

Las notas de aplicación pueden encontrarse en la página web de Microchip: www.microchip.com.

Motor Control Development Tools

- PICDEM™ MC Development Board (DM183011)
Used to evaluate the PIC18FXX31 8-bit microcontroller family.
- PICDEM™ MCLV Development Board (DM183021)
- dsPIC30F Motor Control Development System (DM300020)
Used to evaluate the dsPIC30F 16-bit Digital Signal Controller family.
- Motor Control (MC) Graphical User Interface (GUI)

El MC-GUI permite al usuario configurar el motor y una amplia gama de parámetros del sistema para un tipo de motor seleccionado

El MC-GUI es **gratís** y puede ser bajado en: www.microchip.com

Visite el "Motor Control Design Center" en: www.microchip.com/motor para información adicional.

NOTAS:

CAPITULO 6

Microcontroladores PIC® LCD

Soluciones

Tabla de contenidos

INTRODUCCION

TIP #1: Typical Ordering Considerations and Procedures for Custom Liquid Displays	6-2
TIP #2: LCD PIC® MCU Segment/Pixel Table	6-2
TIP #3: Resistor Ladder for Low Current	6-3
TIP #4: Contrast Control with a Buck Regulator.....	6-5
TIP #5: Contrast Control Using a Boost Regulator.....	6-5
TIP #6: Software Controlled Contrast with PWM for LCD Contrast Control	6-6
TIP #7: Driving Common Backlights	6-7
TIP #8: In-Circuit Debug (ICD).....	6-8
TIP #9: LCD in Sleep Mode	6-8
TIP #10: How to Update LCD Data Through Firmware	6-9
TIP #11: Blinking LCD.....	6-9
TIP #12: 4 x 4 Keypad Interface that Conserves Pins for LCD Segment Drivers	6-10
Application Note References	6-11

INTRODUCCIÓN

Usar un "LCD PIC® MCU" para cualquier aplicación puede brindar los beneficios de un sistema de control y una interface con los humanos por medio de un LCD. Diseñar prácticas para aplicaciones con LCD pueden ser mejoradas por medio de la implementación de los siguientes "Tips 'n Tricks" sugeridos.

Este folleto describe varios circuitos básicos y bloques de software comunes usados para manejar displays LCD. También brinda referencias a notas de aplicación de Microchip que describen varios conceptos sobre LCD con mayor detalle.

TIP #1 Typical Ordering Considerations and Procedures for Custom Liquid Displays

1. Considere qué información útil necesita ser mostrada en el display LCD y qué combinación alfanumérica e íconos especiales van a ser necesarios.
2. Analice en qué entorno va a operar el LCD. La tensión de operación y la temperatura influyen en el contraste del LCD y pueden limitar potencialmente el tipo de LCD que puede ser usado.
3. Determine el número de segmentos necesarios para lograr el display deseado en el LCD y refierace al "PIC Microcontroller LCD Matrix" para el microcontrolador LCD PIC apropiado.
4. Haga un esquema de la impresión y una descripción del LCD y analice los pines de salida del LCD.
5. Envíe el esquema del LCD propuesto y una descripción por lo menos a 3 proveedores para determinar aspectos relacionados con precios, cronogramas y calidad.
 - a) Tenga en cuenta el costo NRE, precio por unidad, y costos de configuración.
 - b) Permita un mínimo de dos semanas para hacer dibujos formales de la mecánica y asignación de pines.

6. Pida un mínimo prototipo inicial del LCD para asegurarse que el LCD fue desarrollado correctamente y asegurarse el funcionamiento correcto en la aplicación.

- a) Permita un lapso de 4-6 semanas para la entrega del prototipo inicial del LCD.

7. Cuando reciba el prototipo, confirme que funciona correctamente antes de dar una aprobación definitiva y comenzar con la producción del LCD.

Nota:

Asegúrese de tener anotado todo el material que haya intercambiado con el proveedor como esquemas iniciales, dibujos, asignaciones de pines, etc.

TIP #2 LCD PIC® MCU Segment/Pixel Table

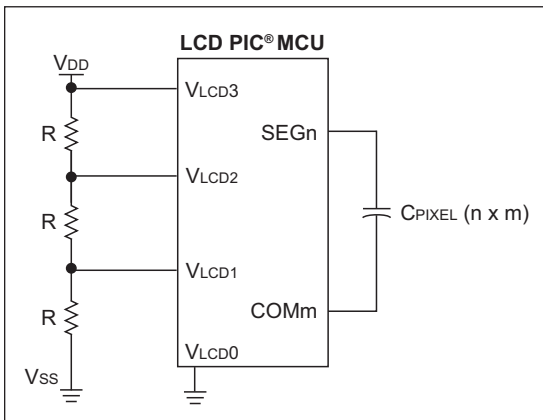
Tabla 2-1: Segment Matrix Table

Multiplex Commons	Maximum Number of Segments/Pixels					Bias
	PIC16F913/ 916	PIC16F914/ 917	PIC16F946	PIC18F6X90 (PIC18F6XJ90)	PIC18F8X90 (PIC18F8XJ90)	
Static (COM0)	15	24	42	32/ (33)	48	Static
1/2 (COM1: COM0)	30	48	84	64/ (66)	96	1/2 or 1/3
1/3 (COM2: COM0)	45	72	126	96/ (99)	144	1/2 or 1/3
1/4 (COM3: COM0)	60	96	168	128/ (132)	192	1/3

TIP #3 Resistor Ladder for Low Current

Las tensiones necesarias para la polarización se generan por medio de un "resistor ladder". Como éste está conectado entre V_{DD} y V_{SS}, va a fluir una corriente inversamente proporcional a la resistencia. En otras palabras, cuanto mayor sea la resistencia, menor va a ser la corriente que va a fluir por el "resistor ladder". Si se usan resistencias de 10K y V_{DD}=5V, el "resistor ladder" va a entregar 166µA. Esto es bastante corriente para algunas aplicaciones alimentadas con baterías.

Figura 3-1: Resistor Ladder

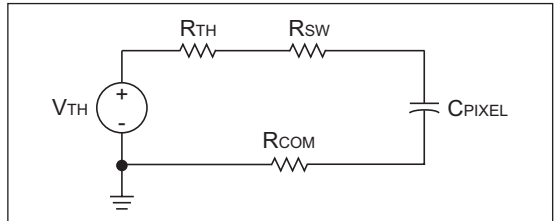


Cómo maximizar la resistencia sin afectar adversamente la calidad del display? Algunas ideas básicas de análisis de circuitos van a ayudar a determinar hasta qué valor se pueden aumentar las resistencias del "resistor ladder".

El módulo LCD es básicamente un multiplexor analógico que conecta de manera alternada las tensiones del LCD a los distintos segmentos y pines comunes que conectan a los pixeles del LCD. Estos pixeles pueden ser modelados como un capacitor. Cada punto de conexión del "resistor ladder" puede ser modelado con un circuito equivalente de Thevenin. La resistencia de Thevenin es 0 para VLCD3 y VLCD0, así que concentrémosnos en los dos casos en donde no es cero, VLCD2 y VLCD1.

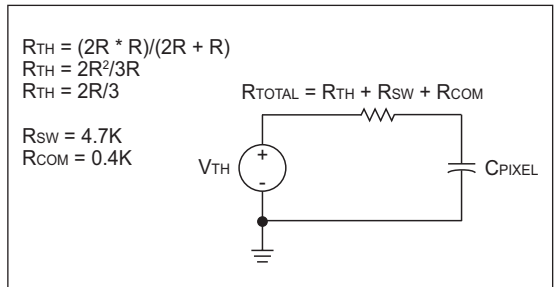
El circuito puede ser simplificado como se muestra en la Figura 3-2. R_{SW} es la resistencia del "segment multiplex switch" y R_{COM} es la resistencia del "common multiplex switch".

Figura 3-2: Circuito LCD Simplificado



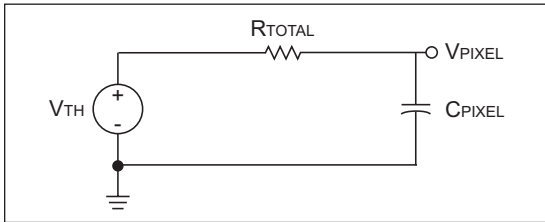
El voltaje de Thevenin es igual a 2/3 V_{DD} o 1/3 V_{DD}, en los casos en que la resistencia de Thevenin no es cero. La resistencia de Thevenin es igual a la resistencia paralela de la parte superior e inferior del "resistor ladder".

Figura 3-3: Estimación de la Resistencia del Circuito LCD



Como puede ver, se puede modelar la alimentación de un sólo pixel como un circuito RC, donde la tensión varía entre 0V hasta V_{th}, por ejemplo. Para los microcontroladores LCD PIC, se puede estimar las resistencias del segmento y comunes como 4.7K y 0.4K, respectivamente. Se puede ver que el tiempo que le lleva a la tensión del pixel cambiar de 0 a V_{th} va a depender de la capacidad del pixel y la resistencia total, en la cual la resistencia de Thevenin del "resistor ladder" resulta la más significativa.

Figura 3-4: Cambio de Tensión en un Pixel



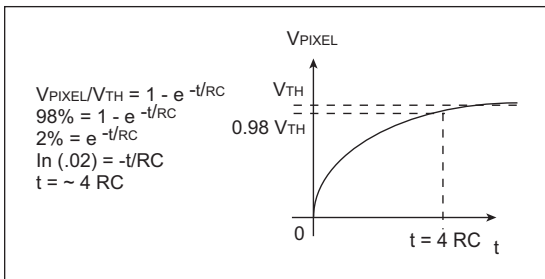
La respuesta al escalón de la caída de tensión a través de un pixel responde a la siguiente ecuación:

Ecuación 3-1

$$V_{PIXEL} = V_{TH} (1 - e^{-t/RC})$$

Manipulando la ecuación, se puede ver que lleva un tiempo de 4 veces la constante de tiempo para la tensión del pixel alcanzar el 98% de la tensión de polarización.

Figura 3-5: Respuesta al Escalón



Ahora necesitamos estimar la capacidad. La capacidad es proporcional al área del pixel. Ésta se puede medir y estimar la capacidad como se muestra a continuación. Obviamente, un display mayor, como un reloj de pared digital, va a tener pixeles más grandes con capacidades mayores.

Ecuación 3-2

$$C_{PIXEL} = 1500 \text{ pF/cm}^2$$

$$AREA_{PIXEL} = 1 \text{ mm} * 3 \text{ mm} = .03 \text{ cm}^2$$

$$C_{PIXEL} = 45 \text{ pF}$$

Se busca que la constante de tiempo sea mucho menor al período de la forma de onda del LCD, de manera que el redondeo de la forma de onda del LCD sea mínima. Si queremos que el RC sea igual a $100\mu\text{s}$, entonces la resistencia total se puede calcular como se muestra a continuación:

Ecuación 3-3

$$R_{TOTAL} = 100 \mu\text{S}/45 \text{ pF} = 2.22 \text{ m}\Omega$$

$$R_{TH} = 2.2\text{M} - 5.1\text{K} = 2.2\text{M}$$

La resistencia de los circuitos "switching" dentro del módulo LCD es muy pequeña comparada con esta resistencia, por lo que la resistencia de Thevenind del "resistor ladder" en Vlcd2 y Vlcd1 pueden ser tratados como la Rtotal. Se puede calcular el valor de R que nos de el valor adecuado de la resistencia de Thevenin.

Ecuación 3-4

$$R = 3 R_{TH}/2 = 3.3\text{M}$$

Ahora se puede calcular la corriente que fluye por el "resistor ladder" si se usan resistencias de 3.3 mΩ.

Ecuación 3-5

$$R_{LADDER} = 9.9\text{M},$$

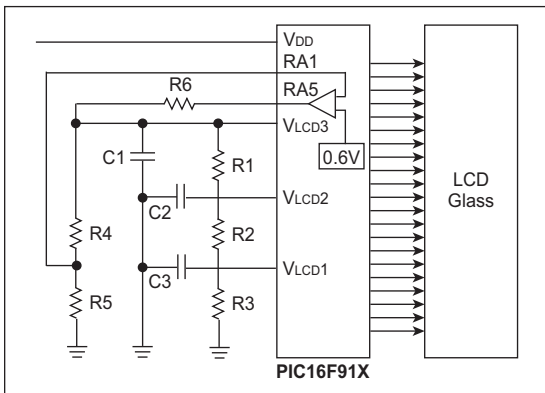
$$I_{LADDER} = 5\text{V}/9.9\text{M} = 0.5 \mu\text{A}$$

Use este proceso para estimar la resistencia máxima de su "resistor ladder" y va a poder reducir el consumo de potencia para su aplicación con LCD de manera drástica. No se olvide de observar el display en las condiciones de operación (como temperatura, humedad, tensión, etc.) para asegurarse que el contraste del display sea de buena calidad.

TIP #4: Contrast Control with a Buck Regulator

El control de contraste en cualquiera de los LCD PIC® MCUs es llevada a cabo controlando las tensiones aplicadas a las entradas V_{Lcd}. El generador más simple de voltaje para el contraste es ubicar un divisor resistivo en estos tres pines. El circuito se muestra a continuación. El método de "resistor ladder" es muy útil para muchas aplicaciones, pero no funciona en casos donde el contraste debe mantenerse dentro de un rango de V_{DDs}. La solución es usar un regulador de tensión. Éste puede ser externo al dispositivo, o puede ser construido usando un comparador interno al microcontrolador LCD PIC.

Figura 4-1: Voltage Generator con Resistor Divider

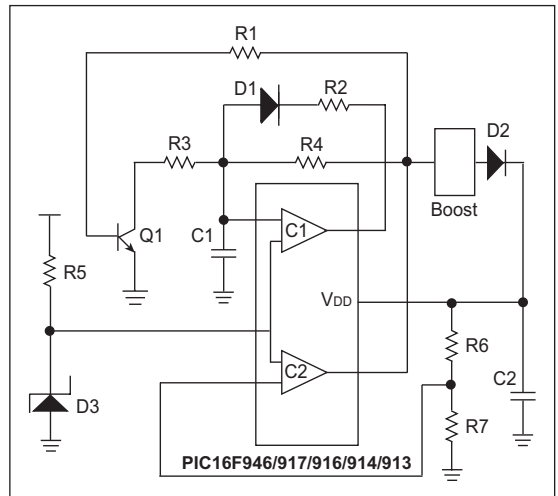


Los dispositivos PIC16F946/917/916/914/912 tienen un modo especial de "Comparator" que brinda una tensión de referencia de 0.6V. El circuito que muestra la Figura 4-1 hace uso de esta referencia para brindar una tensión regulada de contraste. En este circuito, R1, R2 y R3 brindan las tensiones de control de contraste. La tensión en V_{Lcd3} es comparada con la referencia interna dividiendo la tensión en V_{Lcd3} en R4 y R5 y aplicando una tensión reducida al comparador interno. Cuando la tensión en V_{Lcd3} está cerca de la tensión deseada, la salida del comparador va a comenzar a oscilar. Estas oscilaciones son filtradas por una tensión DC por medio de R6 y C1. C2 y C3 son pequeños capacitores que aseguran que las tensiones en V_{Lcd1} y V_{Lcd2} sean estables.

TIP #5: Contrast Control Using a Boost Regulator

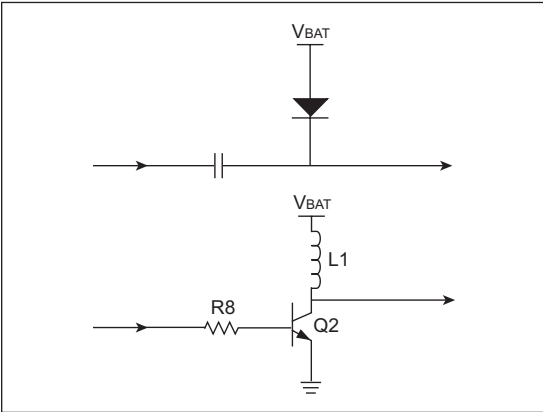
En el Tip #4 del LCD, se creó un "buck converter" usando un comparador. El circuito funciona perfecto cuando V_{DD} es mayor que la tensión del LCD. El microcontrolador PIC puede operar todo el rango hasta los 2.0V, mientras que casi todos los LCD de baja tensión solo operan hasta los 3V. En una aplicación donde se usa una batería, es importante que se mantenga en operación el máximo tiempo posible. Por eso, se requiere de un "boost converter" de 2.0 V a 3.0 V para el LCD. La figura debajo muestra un circuito para lograr esto.

Figura 5-1: Boost Converter



En este circuito, se usan los dos comparadores. La tensión es determinada por el valor del diodo Zener D3 y la tensión en R6:R7. El resto del circuito crea un simple "multivibrador" para estimular al circuito estimulador. Este circuito puede ser basado en un inductor o un capacitor. Cuando la tensión de salida es muy baja, el "multivibrador" oscila y causa que C2 se cargue. Como la tensión en C2 incrementa, el "multivibrador" empieza a operar esporádicamente para mantener la tensión deseada en C2.

Figura 5-2: Dos tipos de "Boost Converter"

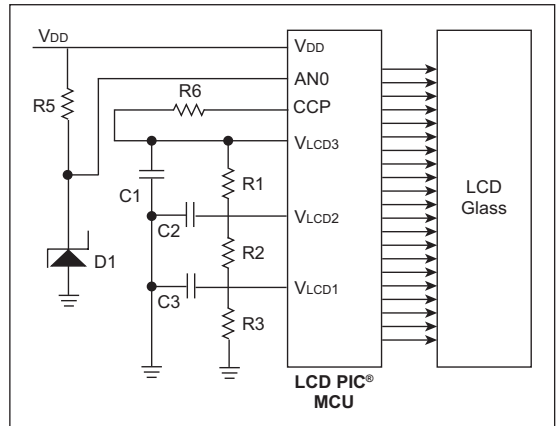


Los dos métodos de producir un "boos converter" se muestran en la figura de arriba. El primer circuito es simplemente un circuito del tipo "switched capacitor". El segundo es un circuito del tipo "standard inductor boost". Estos circuitos funcionan levantando V_{dd}. Esto permite la tensión en V_{lcd} exceder V_{dd}.

TIP #6: Software Controlled Contrast with PWM for LCD Contrast Control

En los circuitos anteriores para el control del contraste, la tensión de salida estaba seteada por una referencia fija. En algunos casos, el contraste debe ser variable para que responda a diferentes tipos de condiciones de operación. El módulo CCP, disponible en el dispositivo LCD, permite usar una señal PWM para controlar el contraste. En la figura 6-1, se puede ver el circuito modificado conectando la entrada a RA6 a un pin CCP. El divisor resistivo creado por R4 y R5 en el diseño anterior no es necesario. Una entrada a el ADC es usado para brindar realimentación pero esto puede ser opcional. Si se usa el feedback del ADC, fijese que es usado para monitorear el suministro V_{dd}. El PWM va a ser usado para compensar las variaciones de tensión.

Figura 6-1: Software Controlled Voltage Generator



TIP #7 Driving Common Backlights

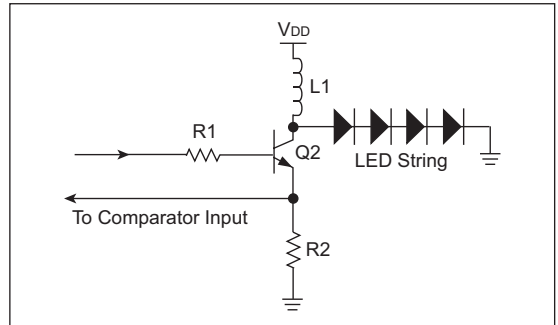
Cualquier aplicación que opere en condiciones de poca luz requiere de un "backlight". La mayoría de las aplicaciones de bajo costo usan una de las siguientes opciones como "backlight":

- 1) Electroluminescente (EL)
- 2) LEDs en serie
- 3) LEDs en paralelo

Otras tecnologías de "backlight", como CCFL, son usadas en paneles gráficos de alta luminosidad como esos encontrados en laptops. El uso de LEDs blancos es común en LCDs de color, donde una luz blanca es necesaria para crear los colores.

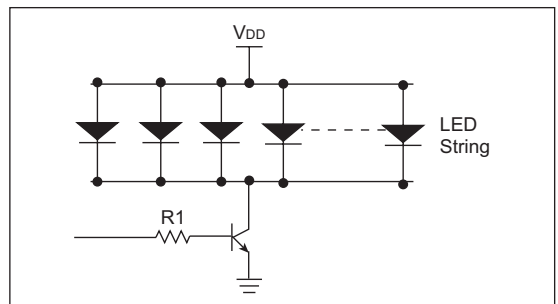
Manejar un panel EL simplemente requiere de una señal AC. Puede generar esta señal usando un segmento no utilizado del controlador del LCD. La señal también puede ser generada por el módulo CCP o por medio de software. La señal de AC debe pasar por un transformador para lograr una ganancia de tensión para generar la tensión requerida. Los LEDs en serie pueden ser utilizados con un "boost power supply". En el siguiente diagrama, se muestra un simple "boost supply". En este circuito, se le aplica un pulso al transistor. La duración del pulso es controlado por la corriente de R2. Cuando el pulso es apagado, la corriente almacenada en el inductor es transferida a los LEDs. La tensión va a aumentar al nivel requerido para suministrar la corriente necesaria a los LEDs. La tensión de quiebre del transistor debe ser igual a la tensión de directa de los LEDs multiplicada por el número de LEDs. La referencia del comparador puede ser ajustado por software para cambiar el nivel de la salida de los LEDs.

Figura 7-1: Simple Boost Supply



Si los LEDs están en paralelo, es más simple aún. En este caso, un transistor es usado como sumidero de la corriente de los LEDs en paralelo. El transistor puede ser modulado por un PWM para adquirir el nivel de salida deseado. Si V_{DD} es mayor que la máxima tensión de directa, se puede agregar una resistencia para controlar la corriente, o bien, se puede ajustar el "duty cycle" del PWM del transistor, para asegurarse que los LEDs cumplan con sus especificaciones.

Figura 7-2: LEDs en Paralelo



TIP #8 In-Circuit Debug (ICD)

Hay dos cuestiones potenciales en el uso del ICD para depurar las aplicaciones LCD. Primero, el controlador LCD puede congelarse cuando se detiene el dispositivo. Segundo, los pines ICD son compartidos con los segmentos en los PIC16F946/917/916/914/913 MCUs.

En el proceso de depuración, el dispositivo es detenido en los "breakpoints" y por el usuario cuando aprieta el botón de pausa. Si el ICD está configurado para detener los periféricos junto con el dispositivo, el control del LCD se va a detener, y va a aplicar tensiones DC al vidrio del LCD. Con el tiempo, estos niveles de DC pueden dañar el vidrio. Sin embargo, para la mayoría de los procesos de depuración, esto debe ser una consideración. El PIC18F LCD MCUs tienen una característica que permite al módulo LCD continuar operando mientras el dispositivo sea frenado durante el proceso de depuración. Esto es importante para chequear la imagen del display mientras que el dispositivo sea frenado y prevenir que el vidrio sea dañado si el dispositivo es detenido por un largo tiempo.

El PIC16F946/917/916/914/913 multiplexa los pines ICSP e ICD con los pines compartidos con los segmentos 6 y 7 del LCD. Si un LCD es agregado a estos pines, el dispositivo puede ser depurado con el ICD; sin embargo, todos los segmentos controlados por esos dos pines van a parpadear y no podrán ser controlados. Ni bien se detiene el proceso de depuración y el dispositivo es programado con el modo "Debug" deshabilitado, estos segmentos vuelven a ser controlados correctamente.

TIP #9 LCD in Sleep Mode

Si se tiene una aplicación con recursos de potencia limitados que debe mostrar datos en un display continuamente, el microcontrolador LCD PIC puede ser puesto en modo "Sleep" mientras el módulo LCD continúa manejando el display.

Para operar el LCD en modo "Sleep", se requieren sólo dos pasos. Primero, se requiere una fuente de tiempo adicional al oscilador principal para que sea la fuente de clock del LCD, ya que durante el "Sleep", el oscilador principal es detenido. A continuación se muestran varias opciones.

Tabla 9-1: Opciones para el LCD en Sleep Mode

Part	LCD Clock Source	Use in Sleep?
PIC16C925/926	Fosc/256	No
	T1OSC	Yes
	Internal RC Oscillator	Yes
PIC16F946/917/916/914/913	Fosc/8192	No
	T1OSC/32	Yes
	LFINTOSC/32	Yes
PIC18F6X90	(Fosc/4)/8192	No
PIC18F8X90	T1OSC	Yes
PIC18F6XJ90	INTRC/32	Yes
PIC18F8XJ90		

Segundo, el bit "Sleep Enable" (SLPEN) debe ser limpiado. El LCD va a continuar mostrando datos mientras el dispositivo está en modo "Sleep". Es así de simple!

Cuando debe elegir el oscilador interno RC (o LFINTOSC) por sobre el oscilador del Timer1? Depende de si la aplicación es lo suficientemente sensible al tiempo para requerir la precisión del cristal del Timer1 o no. Si se tiene una aplicación de cronometraje, entonces probablemente necesite un oscilador con cristal de 32kHz conectado al Timer1.

Como el Timer1 continúa operando durante el Sleep, no hay problema en usar el Timer1 como fuente de clock del LCD. Si no necesita usar el oscilador externo en el Timer1, entonces el oscilador interno RC (INTRC o LFINTOSC) es más que suficiente para usar como fuente de clock para el LCD y no requiere de componentes externos.

TIP #10 How to Update LCD Data Through Firmware

Para actualizar el LCD, el contenido de los registros LCDDATA es modificado para prender o apagar cada pixel del display LCD. La aplicación generalmente va a modificar las variables del "buffer" que son creadas para corresponder con los elementos del display, como posiciones de los caracteres, gráfico de barras, datos de la batería, etc.

Cuando la aplicación llama a un refresco del display, los valores guardados en el "buffer" deben ser convertidos para el seteo correcto de los bits de los pixeles, ubicados en los registros LCDDATA.

Para forma de ondas del tipo A, los registros de LCD Data pueden ser escritos en cualquier momento sin efecto de relleno. Sin embargo, las formas de onda del tipo B, los registros de LCD Data pueden ser escritos en cualquier otro LCD "frame" para asegurarse que dos "frames" del tipo B se complementen el uno al otro. De lo contrario, se puede presentar una DC en el LCD.

Los registros de LCD Data deben ser escritos cuando una escritura es permitida, o cual se indica con el bit WA en '1' en el registro LCDCON.

En el PIC16C926, no existe el bit WA. La escritura de pixeles de datos puede ser coordinada con una interrupción del LCD. Esta interrupción es generada únicamente cuando se elige una forma de onda del tipo B multiplexada.

TIP #11 Blinking LCD

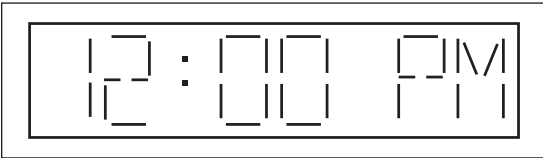
La información puede ser mostrada de diferentes maneras en un LCD externo. Por ejemplo, cómo puede ser llamada la atención del usuario a una porción específica del panel LCD? Una manera que no requiere nada adicional es el efecto de parpadeo.

Fíjese en una aplicación común como lo es un reloj. El ":" entre las horas y los minutos se hace parpadear una vez por segundo (prendido por medio segundo, apagado por medio segundo). Esto muestra que el reloj está funcionando frente a la ausencia del sonido de las manecillas.

Si hay un corte de alimentación, es común hacer que el reloj entero parpadee. Esto le da al usuario una indicación inmediata de que el reloj no está mostrando la hora correcta.

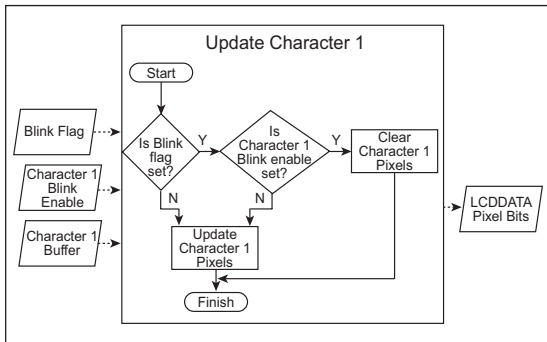
Cuando el usuario fija la hora, el parpadeo se usa para mostrar que se ha entrado en un nuevo modo, como el parpadeo de las horas para indentificar que la hora esta siendo modificada, o parpadeando los minutos para indicar que ahora son éstos lo que están siendo modificados. En un simple reloj, el parpadeo es usado para diferentes propósitos. Sin este efecto, el reloj dejaría de ser tan fácil de usar para el usuario.

Figura 11-1: Aplicación de un Reloj Común



Afortunadamente, el efecto de parpadeo es fácil de implementar. Hay muchas maneras de implementarlo por software. Cualquier evento regular puede ser usado para actualizar el contador del parpadeo. Un "flag" de parpadeo puede ser alternado cada vez que el período del parpadeo termine. Cada caracter o elemento del display que quiera hacer parpadear puede ser asignado a un flag de parpadeo correspondiente.

Figura 11-2: Flowchart del Refresco del Display

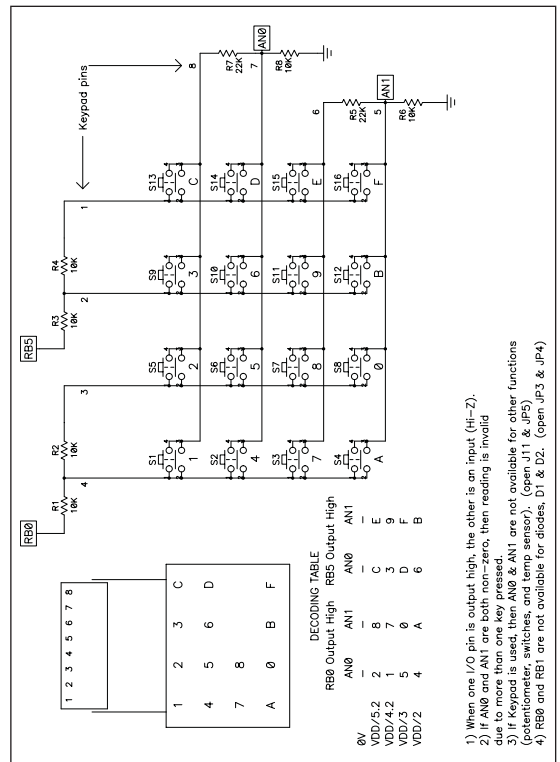


TIP #12 4 x 4 Keypad Interface that Conserves Pins for LCD Segment Drivers

Una interfaz digital típica de un teclado de 4X4 usa 8 pines digitales I/O. Pero usar 8 pines como I/Os digitales deja menos pines disponibles para usar como interfaz para un LCD.

Usando 2 pines digitales I/O y 2 pines analógicos, es posible agregar un teclado de 4X4 al microcontrolador sin sacrificar pines para el LCD. El esquemático para el teclado "hook-up" se muestra en la Figura 12-1. Este ejemplo usa un PIC18F8490, pero la técnica puede ser usada en cualquier LCD PIC® MCUs.

Figura 12-1: Esquemático Keypad Hook-up



- 1) When one I/O pin is output high, the other is an input (Hi-Z).
- 2) If AN0 and AN1 are both non-zero, then reading is invalid due to more than one key pressed.
- 3) If Keypad is used, then AN0 & AN1 are not available for other functions (potentiometer, switches, and temp sensor). (open J11 & JPS)
- 4) RB8 and RB1 are not available for diodes, D1 & D2. (open JP3 & JP4)

Los dos pines digitales I/O usados son el RB0 y RB5, y en cuanto a los analógicos cualquiera puede funcionar. En este caso se usan los pines AN0 y AN1.

Para leer el teclado siga los siguientes pasos:

1. Primero, configure el RB0 como salida "high" y RB5 como entrada. (para presentar alta impedancia)
2. Haga dos conversiones A/D sucesivas, primero en AN0 y luego en AN1.
3. Guarde los resultados en sus variables respectivas; por ejemplo, RB0_AN0_Result y RB0_AN1_Result.
4. Luego, establezca RB5 como salida "high" y RB0 como entrada. (para presentar alta impedancia)
5. Haga dos conversiones A/D sucesivas, primero en AN0 y luego en AN1.
6. Guarde los resultados en sus variables respectivas; por ejemplo, RB5_AN0_Result y RB5_AN1_Result.
7. Ahora hay 4 variables que representan una tecla presionada del teclado 4X4:
 - RB0_AN0_Result
denota que fue apretado el 1,2,4 o 5
 - RB0_AN1_Result
denota que fue apretado el 7,8,A o 0
 - RB5_AN0_Result
denota que fue apretado el 3,C,6 o D
 - RB5_AN1_Result
denota que fue apretado el 9,E,B o F

8. Finalmente, chequee cada valor contra la columna que coincida de la Tabla 12-1. Si está dentro del $\pm 10\%$ de un valor, luego puede ser tomado como que la tecla correspondiente ha sido apretada.

Tabla 12-1: Valores del Teclado

Value $\pm 10\%$	RB0_AN0	RB0_AN1	RB5_AN0	RB5_AN1
$<V_{DD}/10$	–	–	–	–
$V_{DD}/5.2$	2	8	C	E
$V_{DD}/4.2$	1	7	3	9
$V_{DD}/3$	5	0	D	F
$V_{DD}/2$	4	A	6	B

9. Este loop debe ser repetido una vez cada 20 ms aproximadamente.

No se olvide de hacer una rutina anti rebote. Por ejemplo, requiera que los pasos anteriores (con un intervalo de 20ms) retornen el mismo valor dos veces en una fila para considerar que la tecla fue presionada. También, requiera que se devuelva una señal de que no se ha presionado ninguna tecla dos veces antes de buscar la próxima tecla presionada.

Cuando las teclas en el mismo cuadrante son presionadas simultáneamente, tensiones diferentes a la tabla pueden ser generadas. Estos niveles pueden ser ignorados, o si quiere hacer uso de teclas simultáneas para habilitar ciertas funciones, puede agregar una decodificación para esos niveles también.

Referencias a notas de aplicación

- AN220, “*Watt-Hour Meter Using PIC16C923 and CS5460*” (DS00220)
- AN582, “*Low-Power Real-Time Clock*” (DS00582)
- AN587, “*Interfacing PIC® MCUs to an LCD Module*” (DS00587)
- AN649, “*Yet Another Clock Featuring the PIC16C924*” (DS00649)
- AN658, “*LCD Fundamentals Using PIC16C92X Microcontrollers*” (DS00658)
- TB084, “*Contrast Control Circuits for the PIC16F91X*” (DS91084)

Las notas de aplicación pueden ser encontradas en la página web de Microchip: www.microchip.com

CAPITULO 7

Diseño de fuentes inteligentes Soluciones

Tabla de contenido

INTRODUCCION

TIP #1:	Soft-Start Using a PIC10F200.....	7-2
TIP #2:	A Start-Up Sequencer	7-3
TIP #3:	A Tracking and Proportional Soft-Start of Two Power Supplies.....	7-4
TIP #4:	Creating a Dithered PWM Clock	7-5
TIP #5:	Using a PIC® Microcontroller as a Clock Source for a SMPS PWM Generator	7-6
TIP #6:	Current Limiting Using the MCP1630	7-7
TIP #7:	Using a PIC® Microcontroller for Power Factor Correction	7-8
TIP #8:	Transformerless Power Supplies.....	7-9
TIP #9:	An IR Remote Control Actuated AC Switch for Linear Power Supply Designs	7-10
TIP #10:	Driving High Side FETs	7-11
TIP #11:	Generating a Reference Voltage with a PWM Output	7-12
TIP #12:	Using Auto-Shutdown CCP	7-13
TIP #13:	Generating a Two-Phase Control Signal	7-14
TIP #14:	Brushless DC Fan Speed Control	7-15
TIP #15:	High Current Delta-Sigma Based Current Measurement Using a Slotted Ferrite and Hall Effect Device	7-16
TIP #16:	Implementing a PID Feedback Control in a PIC12F683-Based SMPS Design	7-17
TIP #17:	An Error Detection and Restart Controller.....	7-18
TIP #18:	Data-Indexed Software State Machine	7-19
TIP #19:	Execution Indexed Software State Machine	7-20
TIP #20:	Compensating Sensors Digitally	7-21
TIP #21:	Using Output Voltage Monitoring to Create a Self-Calibration Function	7-22

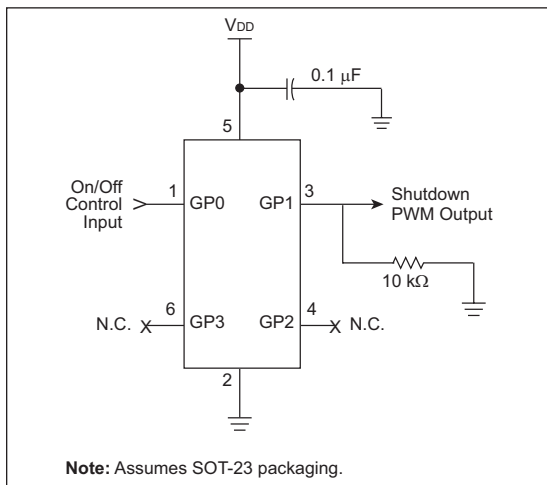
INTRODUCCIÓN

Microchip continúa presentando sus productos innovadores que son más pequeños, rápidos, fáciles de usar y confiables. Los "8-pin Flash microcontrollers (MCU)" son usados en PIC un amplio rango de productos cotidianos, desde cepillos de dientes y secadores de pelo, hasta productos industriales y de medicina. Mientras algunos diseños como "Switch Mode Power Supplies" (SMPS) son tradicionalmente implementados usando un control puramente analógico, estos diseños pueden ser beneficiosos por su posibilidad de configuración e inteligencia que puede brindarle el hecho de agregar un microcontrolador. Este documento hace hincapié en varios ejemplos en los cuales el microcontrolador PIC puede ser usado para incrementar la funcionalidad del diseño con un pequeño incremento en el costo. Varios de los tips proveen ejemplos de software o referencias a otros documentos para más información. El software y las referencias pueden encontrarse en la página web de Microchip: www.microchip.com/tipsntricks.

TIP #1 Soft-Start Using a PIC10F200

Casi todos los controladores de fuente de alimentación están equipados con entradas "shutdown" que pueden ser utilizadas para deshabilitar las salidas del "driver" del MOSFET. Usando PWM, la cantidad de tiempo de operación de la fuente de alimentación puede ser incrementado levemente para permitir que la tensión de salida suba lentamente de 0% a 100%.

Figura 1-1: Esquemático Circuito Soft-Start

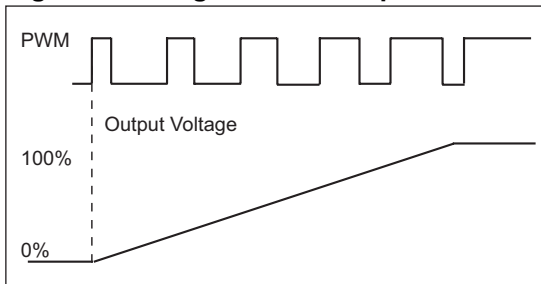


Esta técnica es llamada "arranque lento" y es usado para prevenir una gran irrupción de corrientes asociadas al "start-up" de una fuente de alimentación.

GP0 en el PIC® MCU se usa para habilitar o deshabilitar el "arranque lento". Una vez habilitado, el tiempo de "on" de la señal PWM que maneja la salida de "shutdown" va a incrementarse cada ciclo hasta que la fuente de alimentación esté completamente prendida.

Durante el "Power-on Reset" del PIC® MCU, la salida del PWM (GP1) está inicialmente en el estado de alta impedancia. Una resistencia de "pull-down" en la salida del PWM asegura que la fuente de alimentación no empiece a operar inesperadamente.

Figura 1-2: Diagrama de Tiempos



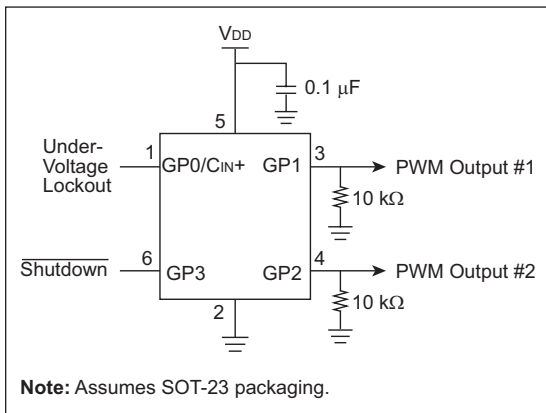
Es importante notar que este tipo de "soft-start" puede ser usado para reguladores de "switching" que respondan rápidamente a cambios en los pines de "shutdown" (como esos que hacen la limitación "cycle-by-cycle"). Algunos reguladores lineales tienen entradas de "shutdown" active-low. Sin embargo, estos reguladores no responden lo suficientemente rápido a los cambios en los pines de "shutdown" como para realizar un "soft-start". Se brinda software de ejemplo para el PIC10F200 que fue sacado de "TB081".

Por favor refiérase a TB081, "Soft-Start Controller For Switching Power Supplies" (DS91081) para mayor información.

TIP #2 A Start-Up Sequencer

Algunos dispositivos nuevos tiene requerimientos múltiples de tensión (ej. "core voltages", "I/O voltages", etc.). La secuencia en que estas tensiones suben y bajan puede ser importante. Extendiendo el tip anterior, un secuenciador de "start-up" puede crearse para controlar los dos voltajes de salida. Dos salidas PWM son generados para controlar los pines de "shutdown" de dos controladores SMPS. Nuevamente, este tipo de contro funciona únicamente con controladores que responden rápidamente a cambios en los pines de "shutdown".

Figura 2-1: Multiple PWM Output Soft-Start Controller



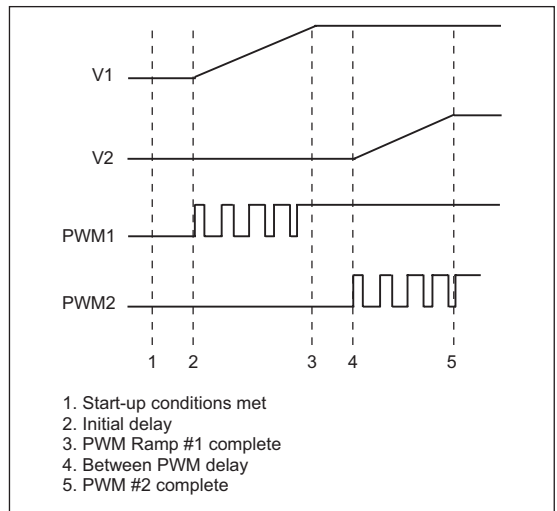
Este diseño utiliza el comparador del PIC® MCU para implementar un bloqueo de bajo voltaje. La entrada en el pin GP0/Cin+ debe estar por encima de la referencia interna de 0.6V para que comience el "soft-start", como se muestra en la Figura 2-2.

Se deben cumplir dos condiciones para que comience la secuencia de "soft-start":

1. El pin de "shutdown" debe mantenerse en Vdd ("logic high").
2. La tensión en GP0 debe estar por encima de 0.6V.

Una vez que las dos condiciones se cumplan, las secuencias van a demorar y el PWM #1 va a ir del 0% al 100%. Un segundo retraso permite estabilizar la primer tensión antes que el PWM #2 vaya de 0% a 100%. Todos los retrasos y tiempos están controlados por software y pueden ser modificados para aplicaciones específicas. Si alguna de las dos condiciones no se cumple, el circuito va a apagar los controladores del SMPS.

Figure 2-2: Timing Diagram



Se brindan ejemplos de software para el PIC10F200 que fue sacado de TB093, "Multiple PWM Output Soft-Start Controller for Switching Power Supplies" (DS91093).

TIP #3 A Tracking and Proportional Soft-Start of Two Power Supplies

Extendiendo el tipo anterior, se puede usar un PIC® MCU para asegurarse que dos tensiones en un sistema aumenten juntas o aumenten en proporción, como se muestra en la Figura 3-1. Este tipo de "start-up" es usado frecuentemente en aplicaciones con dispositivos que requieren de voltajes múltiples.

Como los dos tips anteriores, este tip está diseñado para controlar el pin de "shutdown" en el controlador de SMPS y va a trabajar únicamente con controladores que respondan rápidamente a cambios en el pin de "shutdown".

Figura 3-1: Diagrama de Tiempos

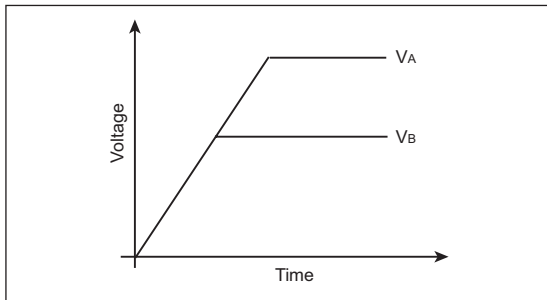
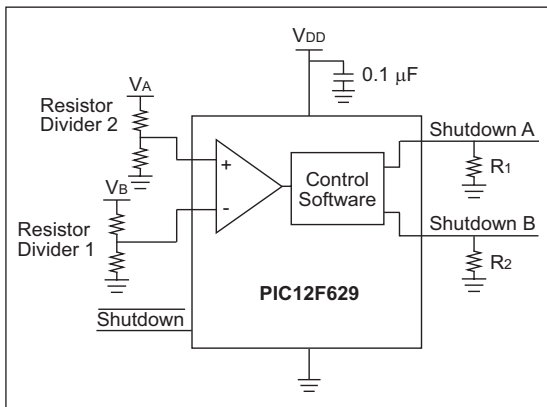


Figura 3-2: Ejemplo Esquemático



El comparador del PIC® MCU es usado para determinar cual de los voltajes es mayor e incrementa el "on-time" de la otra salida según corresponda. La lógica para los pines de "shutdown" se muestra en la Tabla 3-1.

Tabla 3-1: Shutdown Pin Logic

Case	Shutdown A	Shutdown B
$V_A > V_B$	Low	High
$V_B > V_A$	High	Low
$V_B > \text{Internal Reference}$	High	High

Para determinar si se ha llegado al voltaje deseado, Vb se compara con la referencia interna. Si Vb es mayor, ambas salidas de "shutdown" se mantienen altas.

El "Resistor Divider 1" debe ser diseñado de manera tal que el potenciómetro de salida es levemente mayor que la referencia del comparador cuando Vb haya llegado al máximo voltaje.

La razón de las resistencias en el "Resistor Divider 2" puede ser variado para cambiar la pendiente en la cual varía Va.

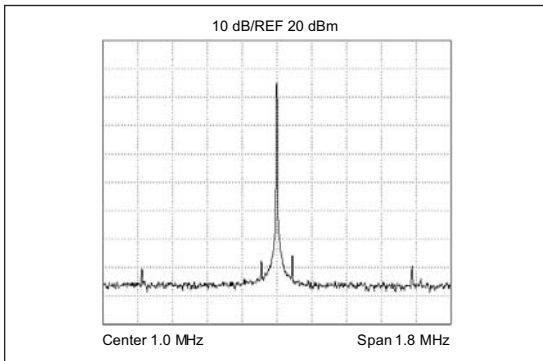
Resistencias de "pull-down" aseguran que las fuentes de alimentación no operen inesperadamente cuando el PIC esté siendo reseteado.

TIP #4 Creating a Dithered PWM Clock

Con el fin de cumplir con los requerimientos de emisiones decretado por el FCC y otras organizaciones regulatorias, la frecuencia de "switching" de una fuente de alimentación puede ser variada. Alternar a una frecuencia fija produce energía en esa frecuencia. Variando la frecuencia de "switching", la energía se esparce dentro de un amplio rango y la magnitud de la energía emitida en cada frecuencia es menor.

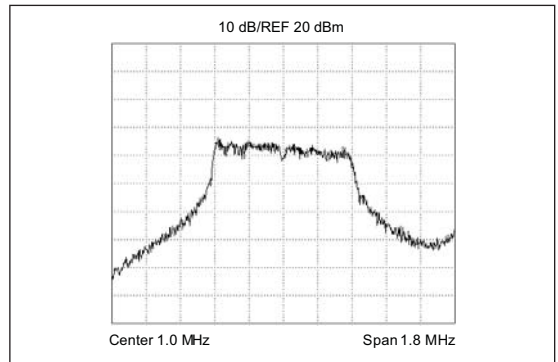
El PIC10F200 tiene un oscilador interno de 4MHz. Una versión escalada del oscilador puede conseguirse en el pin (Fosc/4). Esta salida es un 1/4 de la frecuencia de oscilación (1 Mhz) y va a tener siempre un 50% del "duty cycle". La Figura 4-1 muestra un analizador de espectro de la salida Fosc/4.

Figura 4-1: Spectrum of Clock Output Before Dithering



El PIC10F200 provee un registro "Oscillator Calibration"(OSCCAL) que se usa para calibrar la frecuencia de oscilación. Variando el valor de OSCCAL, la frecuencia del clock puede ser variada. Una secuencia "pseudo-random" fue usada para variar OSCCAL, permitiéndolo frecuencias de aproximadamente 600kHz hasta 1.2MHz. El espectro resultante se muestra en la Figura 4-2.

Figura 4-2: Spectrum of Clock Output After Dithering



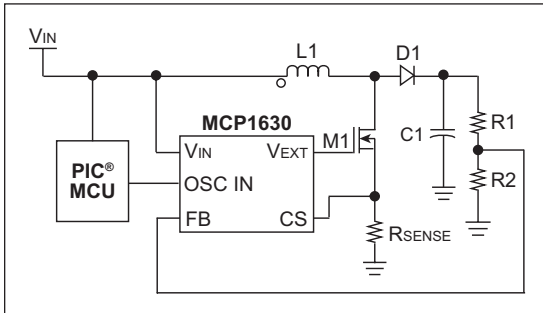
Dispersando la energía sobre un rango más ancho, se puede conseguir una caída de más de 20dB.

Se brinda software de ejemplo para el PIC10F200 que lleva a cabo una secuencia "pseudo-random" para cargar el registro OSCCAL.

TIP #5 Using a PIC® Microcontroller as a Clock Source for a SMPS PWM Generator

Un PIC® MCU puede ser usado como fuente de clock para un generador PWM como el MCP1630.

Figura 5-1: PIC® MCU y MCP1630 Ejemplo Boost Application



El MCP1630 empieza el ciclo cuando las transiciones de la fuente de clock/oscilador de "high-to-low", causando la salida PWM pasar al estado "high". El pulso PWM puede ser terminado de las siguientes 3 maneras:

1. La corriente sensada en el dispositivo magnético alcanza 1/3 de la salida del amplificador de error.
2. La tensión en el pin de "feed-back"(FB) es mayor que la referencia (Vref).
3. Las transiciones de la fuente de clock/oscilador de "low-to-high".

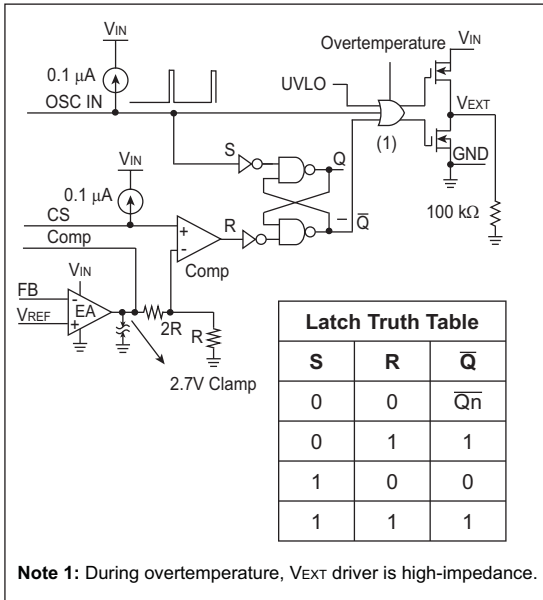
La frecuencia de "Switching" del MCP1630 puede ser ajustada cambiando la frecuencia de la fuente de clock. El máximo "on-timer" del MCP1620 PWM puede ser ajustado cambiando el "duty cycle" de la fuente de clock.

El PIC® MCU tiene varias opciones con respecto a la fuente de clock:

- El pin Fosc/4 puede ser habilitado. Esto va a producir un 50% de "duty cycle", onda cuadrada, con una frecuencia de 1/4 de la frecuencia de oscilación. El Tip #4 brinda ejemplos de software e información de clock usando la salida Fosc/4.
- Para los PIC® MCUs equipados con los módulos CCP o ECCP, una frecuencia variable, se puede crear una señal con "duty cycle" con poco software. Esta señal PWM es hecha por software y permite características avanzadas, como "soft-start".
- Para los que no tienen el módulo CCP o ECCP, un software de PWM puede ser creado. Tips #1 y #2 usan PWM hecho por software para el "soft-start".

TIP #6 Current Limiting Using the MCP1630

Figura 6-1: MCP1630 High-Speed PWM



El diagrama en bloques para el "MCP1630 high-speed PWM driver" se muestra en la Figura 6-1. Una de las características del MCP1630 es la habilidad de realizar corrientes limitantes. Como se muestra en la parte de abajo a la izquierda del diagrama, la salida del "Error Amplifier" (EA) esta limita a 2.7V. Por consiguiente, sin importar el error actual, la entrada al terminal negativo del comparador (Comp) esta limitada a 2.7V / 3 o 0.9V.

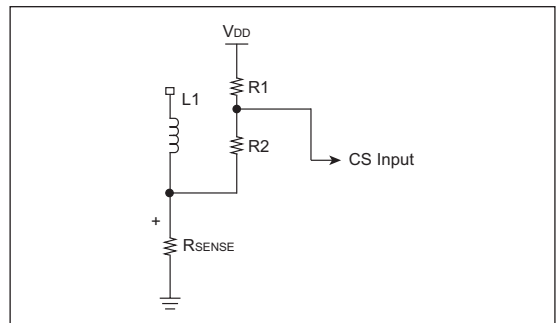
Es posible implementar un limitador de corriente usando una simple resistencia de sentido. En este caso la máxima corriente esta dado por la Ecuación 6-1.

Ecuación 6-1

$$I_{MAX} = (0.9V) / R_{SENSE}$$

Para aplicaciones que requieran altas corrientes, este método puede ser aceptable. Cuando se requieren limitadores de bajas corriente, el valor de la resistencia de sentido, R_{sense}, debe ser incrementado. Esto va a causar una disipación mayor de potencia. Una alternativa es el método mostrado en la Figura 6-2.

Figura 6-2: Límites de Corrientes Bajas



En este caso, la entrada de corriente de sentido (CS) del MCP1630 está polarizada por medio del divisor resistivo R1/R2. La Ecuación 6-2 muestra la nueva corriente limitante.

Ecuación 6-2

$$0.9V = \frac{(V_{DD} - I_{MAX} \cdot R_{SENSE}) \cdot R2}{R1 + R2}$$

Se puede resolver la Ecuación 6-2 para determinar los valores de R1 y R2 que proveen la corriente limitante deseada.

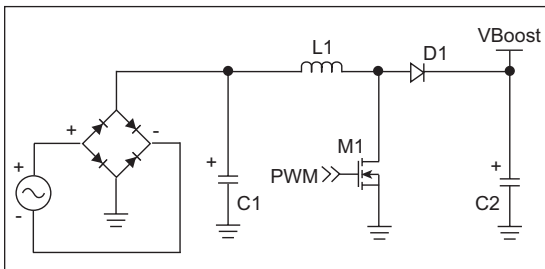
TIP #7 Using a PIC® Microcontroller for Power Factor Correction

En sistemas de AC, el término "Power Factor" (PF) es usado para describir la fracción de la potencia usada actualmente por la carga comparada con la potencia total aparente suministrada.

El "Power Factor Correction"(PFC) es usa para incrementar la eficiencia de la potencia suministrada maximizando el PF.

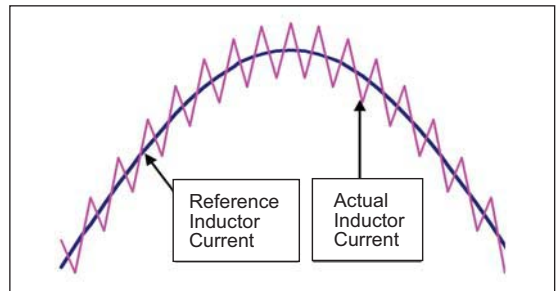
La base para casi todos los circuitos PFC activos es un circuito "boost" mostrado en la Figura 7-1.

Figura 7-1: Typical Power Factor Correction Boost Supply



La tensión de AC es rectificadada e impulsada a tensiones de hasta 400Vdc. La única característica del circuito PFC es que la corriente del inductor es regulada para mantener un cierto PF. Se genera una corriente de referencia de onda senoidal que está en fase con la tensión de línea. La magnitud de la onda sinusoidal es inversamente proporcional a la tensión de Vboost. Una vez que la señal de referencia se establezca, la corriente del inductor es regulada para seguirla, como se muestra en la Figura 7-2.

Figura 7-2: Corrientes Deseadas y Actuales del Inductor



El PIC® MCU tiene varias características que permiten hacer una corrección del factor de potencia:

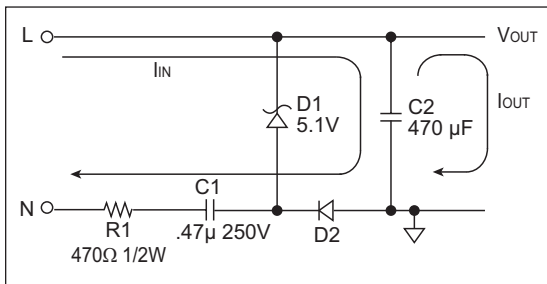
- El módulo CCP del PIC® MCU puede ser usado para generar una señal PWM que, una vez filtrada, puede ser usada para generar la señal de referencia sinusoidal.
- El conversor A/D del PIC puede ser usado para sensar el Vboost y la onda de referencia puede ser ajustada por software.
- El uso de los pines de "interrupt-on-change" permiten al PIC® MCU sincronizar la señal sinusoidal con la tensión de línea detectando los pasos por cero.
- Los comparadores "on-chip" pueden ser usados para manejar el "boost" MOSFET usando la señal de referencia sinusoidal del PWM como una entrada y la corriente actual del inductor como la otra.

TIP #8 Transformerless Power Supplies

Cuando se usa un microcontrolador en una aplicación alimentado de la línea, como el Control Remoto IR descrito en el Tip #9, el costo de construir un transformador de AC/DC puede ser significativo. Sin embargo, hay alternativas sin transformador que se describen a continuación.

Capacitive Transformerless Power Supply

Figura 8-1: Capacitive Power Supply



La Figura 8-1 muestra las bases de una fuente de alimentación capacitiva. El diodo Zener es polarizado en inversa para crear la tensión deseada. La corriente que fluye por el Zener está limitada por R1 y la impedancia C1.

Ventajas:

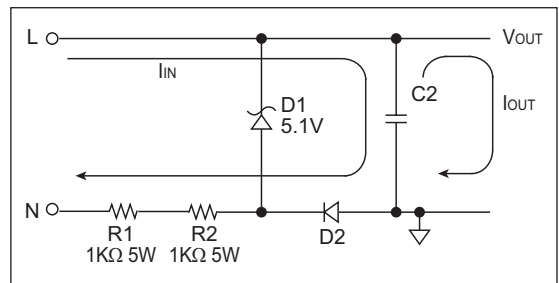
- Es mucho menor que una fuente de alimentación con transformador
- Menor costo que una fuente de alimentación con transformador
- El fuente de alimentación es más eficiente que una fuente de alimentación resistiva.

Desventajas:

- No esta aislada de la tensión AC de la línea lo cual presenta cuestiones de seguridad que deben ser tenidas en cuenta.
- Es más costoso que una fuente de alimentación resistiva porque se requieren capacitores X2.

Resistive Power Supply

Figura 8-2: Resistive Power Supply



La fuente de alimentación resistiva trabaja de modo similar que la capacitiva usando un diodo polarizado en reversa para producir la tensión deseada. Sin embargo, R1 es más grande y es el único elemento limitante de corriente.

Ventajas:

- Significativamente menor que un fuente de alimentación con transformador.
- Menor costo que una fuente de alimentación con transformador.
- Menor costo que una fuente de alimentación capacitiva

Desventajas:

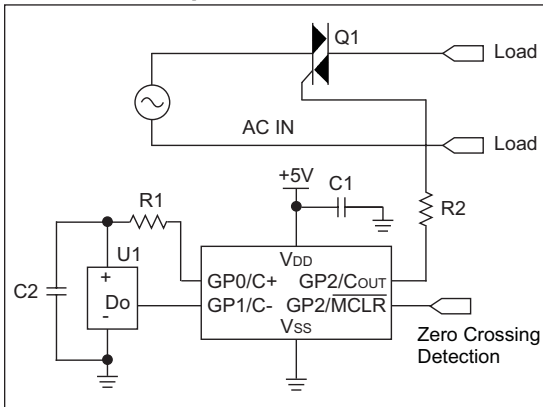
- No está aislada de la línea de tensión AC lo cual presenta cuestiones de seguridad que deben ser tenidas en cuenta.
- La fuente de alimentación es menos eficiente que la fuente de alimentación capacitiva.
- Se disipa más energía en forma de calor en R1.

Se puede conseguir mayor información acerca de estas soluciones, incluyendo ecuaciones en AN954 "Transformerless Power Supplies: Resistive and Capacitive"(DS00954) o en TB008, "Transformerless Power Supply" (DS91008).

TIP #9 An IR Remote Control Actuated AC Switch for Linear Power Supply Designs

Muchas aplicaciones alimentadas de la línea (amplificadores de audio, televisores, etc.) pueden ser prendidos o apagados usando un control remoto infrarrojo. Esto requiere que algunos componentes estén alimentados para recibir las señales del control remoto, aún cuando el dispositivo esté apagado. En esta aplicación conviene usar los microcontroladores "low current PIC". La Figura 9-1 muestra un ejemplo.

Figura 9-1: PIC® MCU Infrared Receiver Esquemático



El PIC10F200 tiene varias características que lo hacen ideal para este tipo de aplicaciones:

- Corrientes de operación y de stand-by extremadamente bajas (350 μ A operación, o.1 μ A en modo sleep).
- Pines de entrada/salida con pull-ups configurables y posibilidad de usar reset-on-change.
- Alta habilidad de "sink/source" (\pm 25mA) que permite manejar dispositivos externos como el receptor IR, directamente desde el pin I/O.
- Habilidad para usar fuentes de alimentación resistiva de bajo costo.
- Pequeño factor de forma (SOT-23).

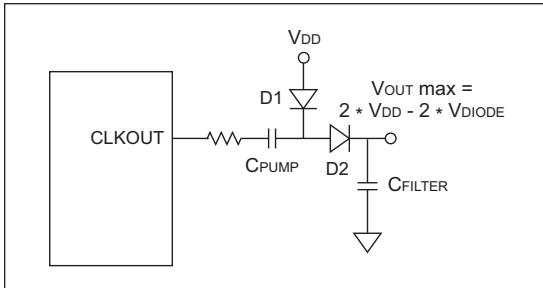
El TB094, "Dimming AC Incandescent Lamps Using A PIC10F200" (DS91094) brinda ejemplos de hardware y software de control infrarrojo.

TIP #10 Driving High Side FETs

En aplicaciones donde se necesita manejar un "high side N channel FET", hay varios métodos para generar una tensión alta para manejarlo. Un método muy simple es usar un "doubling charge pump" como se muestra en la Figura 10-1.

Método 1

Figura 10-1: Change Pump Típico

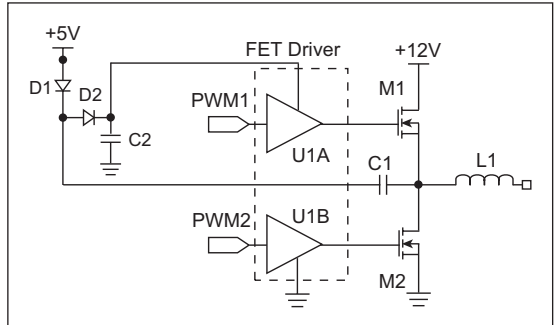


El pin de CLKOUT del PIC® MCU alterna a 1/4 de la frecuencia de oscilación. Cuando el CLKOUT esta en "low", D1 es polarizado en directa y conduce corriente y carga el Cpump. Luego, cuando el CLKOUT está en "alto", D2 se polariza en directo, moviendo la carga a Cfilter. El resultado es una tensión igual al doble de Vdd menos dos caídas de diodo. Esto puede ser usado con un PWM o con cualquier otro pin I/O que alterne.

En la Figura 10-2, se usa un "driver" de FET estándar para manejar los dos FETs usando un diodo y un capacitor.

Método 2

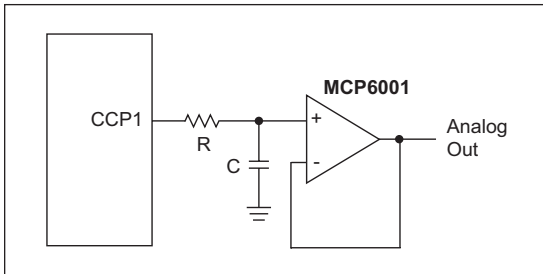
Figura 10-2: Esquemático



Los +5V son usados para alimentar al microcontrolador. Usando esta disposición, el "driver" del FET va a tener aproximadamente $12 + (5 - V_{diode}) - V_{diode}$ como alimentación y es capaz de impulsar los dos FETs. El circuito de arriba funciona cargando C1 por medio de D1 a $(5V - V_{diode})$ mientras M2 esta encendido, conectando efectivamente C1 a tierra. Cuando M2 se apaga y M1 se enciende, un lado de C1 está a 12V y el otro está a $12V + (5V - V_{diode})$. El D2 se enciende y la tensión suministrada al "driver" del FET es $12V + (5V - V_{diode}) - V_{diode}$.

TIP #11 Generating a Reference Voltage with a PWM Output

Figura 11-1: Filtro Pasa-Bajos



Una señal PWM puede ser usada para crear un conversor (DAC) con unos pocos componentes externos. Las conversiones de señales de PWM a señales analógicas involucran el uso de un filtro pasa-bajos analógico. Para eliminar las armónicas no deseadas causadas por la señal de PWM, la frecuencia del PWM (F_{pwm}) debe ser significativamente mayor que el ancho de banda (F_{bw}) de la señal analógica deseada. La Ecuación 11-1 muestra esta relación.

Ecuación 11-1

$$F_{PWM} = K \cdot F_{BW}$$

Las armónicas decrecen a medida que K incrementa

R y C se elijen de acuerdo a la siguiente ecuación:

Ecuación 11-2

$$RC = 1/(2 \cdot \pi \cdot F_{BW})$$

Las armónicas decrecen a medida que K incrementa

Elija el valor de R basado en la capacidad del "drive" y luego calcule el valor de C requerido. La atenuación de la frecuencia del PWM para un filtro RC dado se muestra en la Ecuación 11-3.

Ecuación 11-3

$$Att(dB) = -10 \cdot \log [1 + (2 \pi \cdot F_{PWM} \cdot RC)^2]$$

Si la atenuación calculada en la Ecuación 11-3 no es suficiente, entonces debe aumentarse el K de la Ecuación 11-1.

De manera de atenuar suficientemente las armónicas, puede ser necesario usar un pequeño capacitor o una resistencia muy grande. Cualquier corriente va a afectar la tensión en el capacitor. Agregando un amplificador operacional permite que la tensión analógica sea "buffered" y cualquier corriente va a ser suministrada por el AO y no por el capacitor.

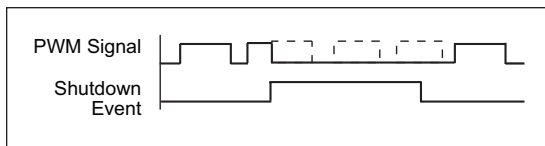
Para más información acerca del uso de señales PWM para generar una salida analógica recurra a AN538, "Using PWM to Generate Analog Output" (DS00538).

TIP #12 Using Auto-Shutdown CCP

PWM Auto-Shutdown

Varios microcontroladores PIC® MCU, como el PIC16F684, PIC16F685 y PIC16F690, tienen la característica de auto-shutdown del PWM. Cuando el "auto-shutdown" está habilitado, un evento puede terminar con el pulso PWM actual y prevenir pulsos siguientes hasta que el evento no sea limpiado. El ECCP puede ser configurado para empezar a generar pulsos automáticamente una vez que el evento sea limpiado.

Figure 12-1: PWM Auto-Shutdown Timing

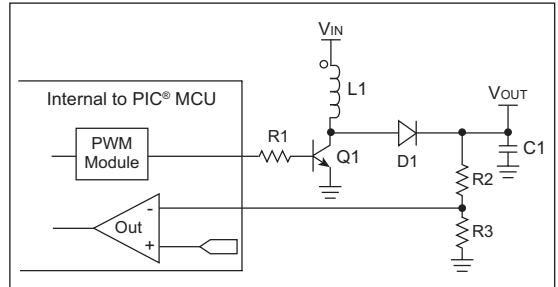


La Figura 12-1 muestra un ejemplo de los tiempos del "auto-shutdown" del PWM. Cuando ocurre el evento del "shutdown" el pulso es terminado de inmediato. En este ejemplo, los dos pulsos siguientes también son terminados porque el evento del "shutdown" no había sido limpiado cuando comenzó el período del pulso. Una vez que el evento haya sido limpiado, los pulsos comienzan de nuevo, pero solamente al comienzo del período.

Usando Auto-Shutdown para crear un Boost Supply

Usando el "auto-shutdown", se puede crear un simple SMPS. La figura 12-2 muestra un ejemplo.

Figura 12-2: Boost Power Supply



Esta configuración de la fuente de alimentación tiene características únicas:

1. La frecuencia de "switching" es determinada por la frecuencia del PWM, por ende, puede ser cambiada en cualquier momento.
2. El máximo de tiempo "on" es determinado por el "duty cycle", y por ende, puede ser cambiado en cualquier momento. Esto provee una manera muy fácil de implementar un "soft-start"
3. En los PIC® MCU que tienen un módulo de referencia programable, la tensión de salida puede ser configurada en cualquier momento.

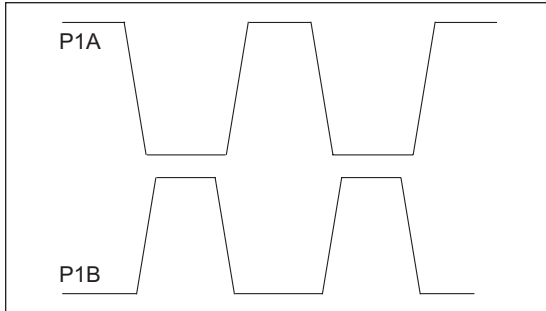
La topología puede ser re-diseñada para crear otro tipo de alimentaciones.

Se brinda software de ejemplo para el PIC16F685 (pero puede adaptarse a cualquier PIC con el módulo ECCP). Este software configura los módulos PWM y comparador como se muestra en la Figura 12-2.

TIP #13 Generating a Two-Phase Control Signal

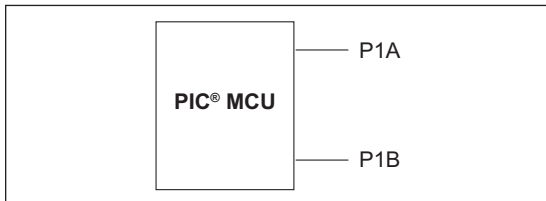
Las fuentes de alimentación que usan una topología "push-pull" o con múltiples componentes "switching" requieren de una señal de control de dos fases como se muestra en la Figura 13-1.

Figura 13-1: Señal de control de dos fases



Es posible producir este tipo de señal de control con dos ondas cuadradas fuera de fase usando el módulo ECCP del PIC® MCU.

Figure 13-2: Two-Phase Control Signal Schematic



Para configurar el ECCP para que produzca este tipo de salida:

1. Configure el ECCP en "half H-bridge PWM pulse" con ambas salidas en "active-high".
2. Setee el registro del "duty cycle" (CCPR1L) con el máximo duty cycle de 50%.
3. Cambie el generador programable de "dead-time" para reducir el ancho de pulso al valor deseado.

El generador programable de "dead-time" tiene una resolución de 7 bits, por ende, los pulsos resultantes van a tener una resolución de 7 bits. Cada pulso va a tener un 50% de "duty cycle", menor al "dead time".

Usando un clock interno de 4MHz se producen pulsos de salida de 31kHz, y usando un cristal del 20MHz va a producir una salida de 156kHz. La frecuencia de salida puede ser incrementada con una pérdida en la resolución.

Se brindan ejemplos de software para el PIC16F684, pero este tipo se aplica a todos los PICs MCUs que tengan el módulo ECCP.

TIP #14 Brushless DC Fan Speed Control

Hay varios métodos para controlar la velocidad de un ventilador "DC brushless". Este tipo de ventilador, el consumo de potencia permitido y el tipo de control deseado son factores a tener en cuenta para elegir el tipo apropiado.

Figura14-1: Low-Side PWM Drive

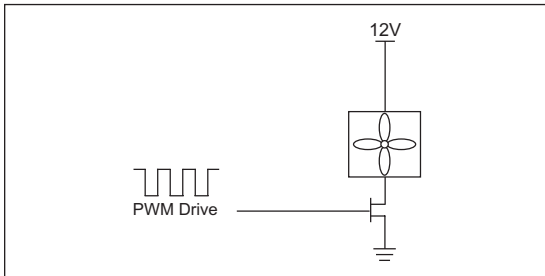
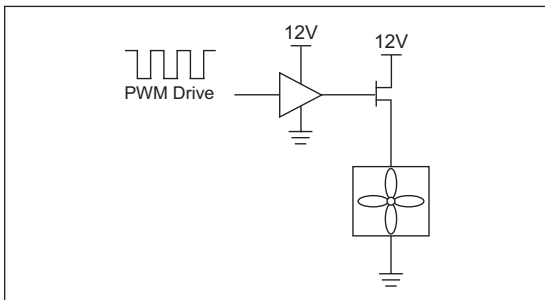


Figura 14-2: High-Side PWM Drive



Método 1 – Modulación Pulse-Width

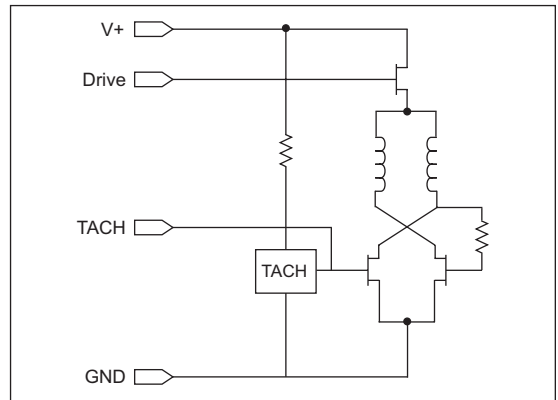
Como se muestra en la Figura 14-1 y 14-2, un simple drive PWM puede ser usado para encender o apagar un ventilador "two-wire". Mientras que es posible usar el circuito de la Figura 14-1 sin un driver con MOSFET, algunos de los fabricantes establecen que el "switching" en el "low-side" del ventilador caducaría la garantía.

Debido a esto, es necesario alternar el "high side" del ventilador para controlar la velocidad. La manera más fácil de controlar la velocidad es encendiendo o apagando. Sin embargo, si se requiere de un grado de control mayor, se puede usar un PWM para variar la velocidad del ventilador.

Para ventiladores "3-wire", la salida del tacómetro no va a ser precisa si se usa el PWM. El sensor que provee la salida del tacómetro en un ventilador "3-wire" es alimentado de la misma fuente que las bobinas del ventilador, en consecuencia, usar un PWM para controlar la velocidad del ventilador transformaría el tacómetro impreciso.

Una solución es usar un ventilador "4-wire" que incluye tanto el tacómetro como la entrada para el "drive". La figura 14-3 muestra un diagrama.

Figura 14-3: Típico 4-Wire Fan

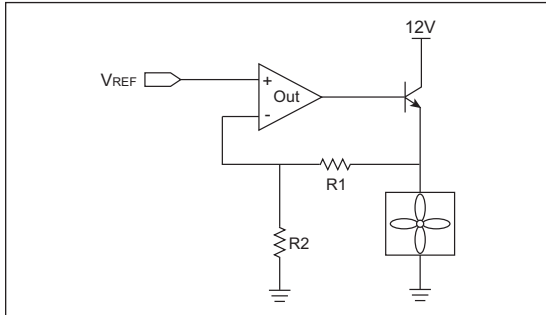


Un ventilador "4-wire" permite que la velocidad sea controlado usando PWM via "Drive line". Como la potencia del sensor del tacómetro no es interrumpida, va a continuar entregando la velocidad correcta.

Método 2 – Control Lineal

Cuando se usa un PWM, la tensión va a variar entre el máximo y el mínimo, sin embargo, es posible usar un método lineal para controlar la velocidad del ventilador como se muestra en la Figura 14-4.

Figura 14-4: Linear Control Drive

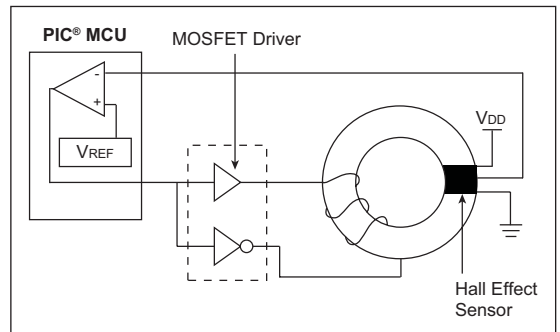


La tensión aplicada en el terminal no inversor del amplificador operacional es usado para variar la tensión a través del OA. La tensión del terminal no inversor puede ser producida por un convertidor (DAC) o por el método mostrado en el Tip #11. Cuando se usa este método, se debe tener cuidado para asegurarse que la tensión del ventilador no sea baja o el ventilador dejará de girar. Una ventaja de este método sobre PWM es que la salida del tacómetro va a funcionar correctamente en ventiladores "3-wires". La desventaja, sin embargo, es que usualmente ofrecen menos velocidad en el control. Por ejemplo, un ventilador de 12V no va a girar debajo de los 8V, así que un rango de solo 4V está disponible para el control. Un ventilador de 5V no va a girar por debajo de los 4V, así que el rango de control se reduce a 1V, que es inaceptable. Otra desventaja es el consumo del circuito. El transistor va a disipar más potencia que en el método del PWM.

TIP #15 High Current Delta-Sigma Based Current Measurement Using a Slotted Ferrite and Hall Effect Device

Muchos sensores de corriente se basan en núcleos de ferrito. La alinealidad de este último puede llevar a resultados poco precisos, especialmente con corrientes altas. Una manera de evitar las alinealidades es mantener el flujo neto en el ferrito cerca de cero. Considere el circuito de la Figura 15-1.

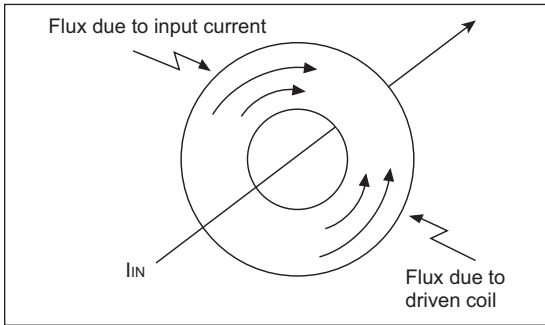
Figura 15-1: Hall Effect Current Measurement Schematic



La salida de los sensores de efecto "Hall" son proporcionales a la corriente que se mide. Cuando $I_{in}=0$ amps, la salida del sensor va a ser de $V_{DD}/2$. Una corriente pasando por el sensor en una dirección va a incrementar la salida del sensor, y una corriente en la otra dirección va a decrementar la salida del sensor.

La salida del comparador es usado para impulsar una bobina de alambre alrededor del núcleo de ferrito. Esta bobina va a ser usada para crear un flujo en dirección contraria al flujo impuesto en el núcleo.

Figura 15-2: Direcciones del flujo



El flujo neto en el núcleo debe ser aproximadamente cero. Como el flujo va a estar siempre cerca de cero, el núcleo va a ser bastante lineal en un rango pequeño.

Cuando $I_{in}=0$, la salida del comparador va a tener un 50% de "duty cycle" aproximadamente. A medida que la corriente se mueve en una dirección, el "duty cycle" va a incrementar. A medida que la corriente se mueva en la otra dirección, el "duty cycle" va a decrementar. Midiendo el "duty cycle" de la salida del comparador, se puede determinar el valor de I_{in} . Finalmente, un "Delta-Sigma" ADC puede ser usado para llevar a cabo mediciones.

Características como el $\Delta\Sigma$ del comparador y la compuerta del Timer1 permiten a la conversión "Delta-Sigma" sea cuidada enteramente por hardware. Sacando $65536(2^{16})$ muestras y contando el número de muestras que la salida del comparador estuvo "low" o "high", se puede obtener un resultado de 16 bits del A/D.

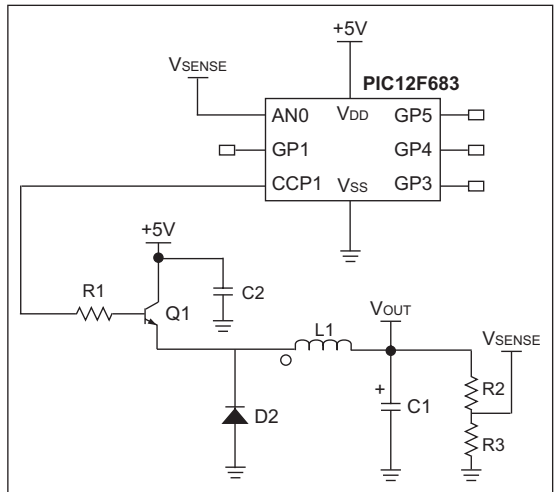
Se brindan esquemáticos y software para el PIC12F683 para C y Assembler.

Para mas información de como usar un PIC® MCU para implementar un conversor "Delta-Sigma" vea AN700, "Make a Delta-Sigma Converter Using a Microcontroller's Analog Comparator Module" (DS00700), que incluye ejemplo de software.

TIP #16 Implementing a PID Feedback Control in a PIC12F683-Based SMPS Design

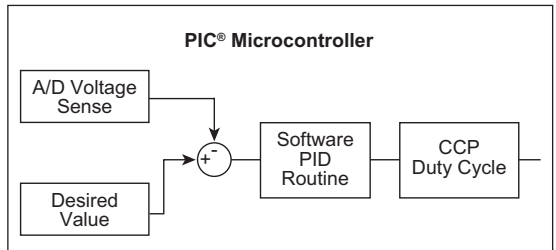
Las fuentes "switching" simples de alimentación pueden ser controladas digitalmente usando un algoritmo Proporcional Integrador Derivador (PID) en vez de un amplificador de error analógico sensando la tensión usando el conversor ADC.

Figura 16-1: Simple PID Power Supply



El diseño de la Figura 16-1 utiliza un PIC12F683 MCU de 8 pines en una topología "buck". El PIC12F683 tiene los bloques básicos necesarios para implementar este tipo de fuente de alimentación; un conversor A/D y un módulo CCP.

Figure 16-2: PID Block Diagram



El convertor A/D es usado para sensar la salida de una aplicación en particular. V_{dd} es usado como referencia para el convertor A/D. Si se quiere, se puede usar una referencia más precisa. La tensión de salida se resta al valor deseado, creando un valor de error. Este error se convierte en la entrada de la rutina del PID. Ésta usa el error para determinar el "duty cycle" apropiado para el "drive". Las constantes del PID son pesadas de manera que la mayor parte del control sea proporcional e integrativa. El componente diferencial no es esencial a este sistema y no es usado. Además, las constantes PID pueden ser optimizadas si se requiere una respuesta transitoria en particular, o si una carga transitoria conocida va a ser conectada.

Finalmente, el módulo CCP es usado para crear la señal de PWM en la frecuencia elegida con el "duty cycle" apropiado.

Se brindan ejemplos de software para el PIC12F683 usando el esquemático de la Figura 16-1.

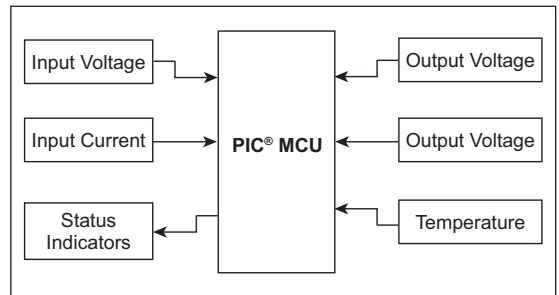
Las siguientes notas de aplicación se relacionan con el control PID y todas incluyen ejemplos de software:

- AN258, "Low Cost USB Microcontroller Programmer The Building of the PICKit® 1 Flash Starter Kit" (DS00258)
- AN937, "Implementing a PID Controller Using a PIC18 MCU" (DS00937)
- AN964, "Software PID Control of an Inverted Pendulum Using the PIC16F684" (DS00937)

TIP #17 An Error Detection and Restart Controller

Una controlador de detección de error y "restart" puede ser creado combinando el Tip #18 y el Tip #19. El controlador usa el ADC del microcontrolador para llevar a cabo las mediciones de tensión y corriente. La tensión de entrada, corriente de entrada, tensión de salida, corriente de salida, temperatura y más pueden ser medidos usando el convertor A/D. Los comparadores "on-boards" son usados para monitorear señales rápidas, como la corriente de salida, asegurando que no excedan un nivel máximo permitido. Muchos PIC® MCUs tienen referencias programables internas para el comparador, que simplifican el circuito.

Figura 17-1: Diagrama en Bloque



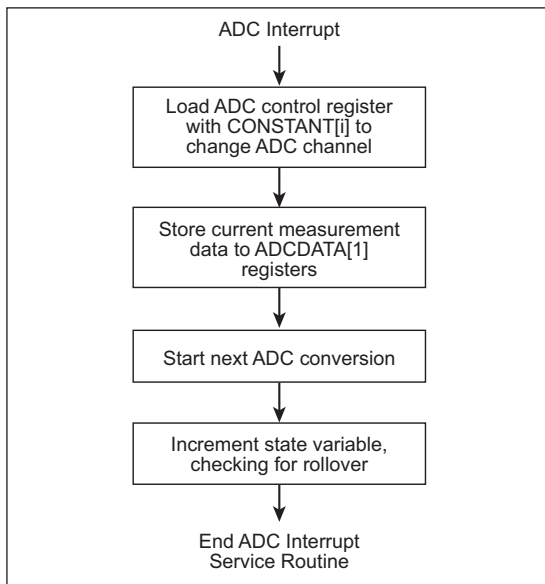
Usando un controlador PIC® MCU permite un nivel mayor de inteligencia en el monitoreo del sistema. En vez de que un sólo evento cause un "shutdown", pueden ser varios los que lo hagan. Un cierto número de eventos en un cierto tiempo o posiblemente una secuencia de eventos pueden ser responsables de un "shutdown". El PIC® MCU tienen la habilidad de volver a arrancar la fuente de alimentación basado en un evento de "shutdown". Algunos eventos (como sobrecorriente) pueden solicitar un "restart" inmediato, mientras que otros eventos (como la sobretemperatura) puede requerir de un retraso antes de realizar el "restart", quizá monitoreando otros parámetros o usando esos para determinar cuando realizar el "restart". También es posible crear este tipo de controlador de detección de error y "restart" con varios de los tipos listados en esta guía.

TIP #18 Data-Indexed Software State Machine

Una máquina de estados puede ser usada para simplificar una tarea dividiendo la tarea en segmentos más pequeños. Basado en la variable de estado, la tarea llevada a cabo o la data usada por la máquina de estado puede ser cambiada. Hay tres tipos básicos de máquinas de estado: "data-indexed", "execution indexed" y un híbrido de las dos. Este tip se va a focalizar en la maquina de estado "data-indexed".

Esta máquina de estado es ideal para el monitoreo de múltiples entradas analógicas con el conversor A/D. Las variables de estado en estas máquinas de estado determinan en que dato se está actuando. En este caso, las tareas de cambiar el canal del ADC, guardar el valor de corriente y comenzar una nueva conversión son siempre las mismas. Un diagrama de flujo simple para una máquina de estado "data-indexed" se muestra en la Figura 18-1.

Figura 18-1: Data-Indexed State Machine Flowchart



Como se muestra en la Figura 18-1, un arreglo constante (CONSTANT[i]) puede ser creado para guardar los valores que deben ser cargados en el registro de control del ADC para cambiar el canal del ADC. Además, un arreglo de datos (ADCDATA[i]) puede ser usado para guardar los resultados de la conversión ADC. Finalmente, la siguiente conversión comienza y la lógica requerida para incrementar y ligar la variable de estado es ejecutada.

Este ejemplo en particular usa la interrupción del ADC para señalar cuando la conversión ha sido completada, y va a intentar tomar las mediciones lo más rápido posible. Se puede crea una subrutina para llevar a cabo la misma tarea, permitiendo al usuario llamar a la subrutina cuando sea necesario.

Se brinda software de ejemplo usando el PIC16F676 y RS-232 para monitorear varios canales del ADC.

TIP #19 Execution-Indexed Software State Machine

Otro tipo de común de máquinas de estado es la "execution-indexed". Este tipo usa una variable de estado para determinar lo que está siendo ejecutado. En C, puede ser pensado como una estructura "switch" como se muestra en el Ejemplo 19-1.

Ejemplo 19-1: Ejemplo Usando Switch

```
SWITCH (State)
{
  CASE 0: IF (in_key()==5) THEN state = 1;
           Break;
  CASE 1: IF (in_key()==8) THEN State = 2;
           Else State = 0;
           Break;
  CASE 2: IF (in_key()==3) THEN State = 3;
           Else State = 0;
           Break;
  CASE 3: IF (in_key()==2) THEN UNLOCK();
           Else State = 0;
           Break;
}
```

Cada vez que el software corre por el loop, la acción llevada a cabo por la máquina de estados cambia con el valor de la variable de estado. Permite a la máquina de estado controlar su propia variable de estado, agregar memoria, o historia, debido a que el estado actual va a estar basado en los estados anteriores. El microcontrolador es capaz de tomar decisiones basadas en entradas y datos anteriores. En assembler, una máquina de estados "execution-indexed" puede ser implementada usando una "jump table".

Ejemplo 19-2: Ejemplo Usando Jump Table

MOVF	state	;load state into w
ADDWF	PCL,f	;jump to state
		;number
GOTO	state0	;state 0
GOTO	state1	;state 1
GOTO	state2	;state 2
GOTO	state3	;state 3
GOTO	state4	;state 4
GOTO	state5	;state 5

En el Ejemplo 19-2, el loop del programa va a saltar al GOTO basado en la variable de estado. El GOTO va a mandar al programa a la rama adecuada. Se debe tener cuidado en asegurar que la variable no va a ser más extensa que lo que debe. Por ejemplo, seis estados (000 al 101) requieren de una variable de estado de tres bits. Puede que la variable de estado se setee en un valor de un estado indefinido (110 al 111), y el comportamiento del programa se vuelve impredecible.

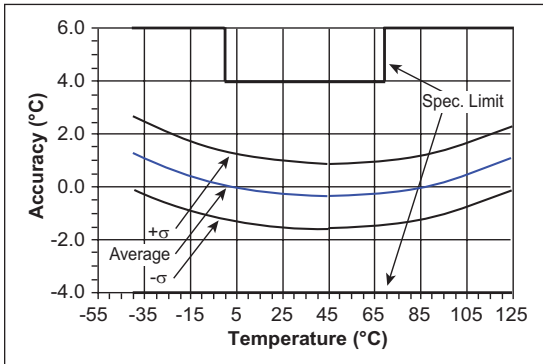
Para evitar este problema incluya:

- Un "mask off" de cualquier bit no usado por la variable. En el ejemplo de arriba, `ANDLW b'00000111'` va a asegurar que solo los 3 bits menos significativos contengan el valor.
- Agregue casos extras para asegurar que siempre haya un "jump" conocido. Por ejemplo en este caso, dos estados extras deben ser agregados y usados como estados de error o "reset".

TIP #20 Compensating Sensors Digitally

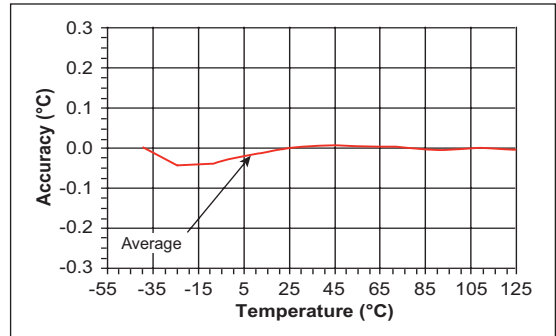
Muchos sensores y referencias tienden a desviarse con la temperatura. Por ejemplo, el MCP9700 especifica que el valor típico es $\pm 0.5^\circ\text{C}$ y el máximo error es $\pm 4^\circ\text{C}$.

Figura 20-1: MCP9700 Accuracy



La Figura 20-1 muestra la precisión de unas 100 muestras de los sensores de temperatura MCP9700. A pesar que el error del sensor es alineal, un microcontrolador PIC puede ser usado para compensar la lectura del sensor. Se pueden ajustar polinomios al error promedio del sensor. Cada vez que se recibe la lectura de la temperatura, el PIC® MCU puede usar el resultado medido y los polinomios de compensación de error para determinar cual es la verdadera temperatura.

Figura 20-2: MCP9700 Average Accuracy After Compensation



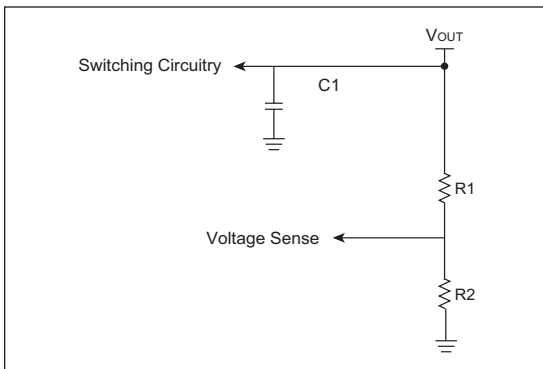
La Figura 20-2 muestra el promedio de precisión para las 100 muestra de los sensores de temperatura MCP9700 después de la compensación. El error promedio fue reducido dentro de todo el rango de temperaturas. También es posible compensar para los errores de las referencias de tensión usando este método. Para mayor información de cómo compensar un sensor digital de temperatura refiérase a:

AN1001, "IC Temperature Sensor Accuracy Compensation with a PIC Microcontroller" (DS01001).

TIP #21 Using Output Voltage Monitoring to Create a Self-Calibration Function

Un microcontrolador PIC puede ser usado para crear una fuente de alimentación "switching" controlada por un loop PID (como se mencionó en el Tip #6). Este tipo de fuentes de alimentación sensa su salida digitalmente, compara esa tensión con la de referencia deseada y modifica el "duty cycle" según corresponda. Sin calibración, es sensible a las tolerancias de los componentes.

Figura 21-1: Typical Power Supply Output Stage



La etapa de salida de muchas fuentes de alimentación es similar a la Figura 21-1. R1 y Rr son usadas para fijar la proporción de tensión que es sensada comparada con la de referencia.

Un simple método de calibrar este tipo de fuentes de alimentación se detalla a continuación:

1. Suministre una referencia de tensión conocida a la salida de la fuente.
2. Ubique la fuente en modo de "Calibration" y permita sensar esa referencia de tensión.

Brindando a la fuente con la tensión de salida que debe producir, puede sensar luego la tensión en el divisor resistivo y guarde el valor sensado. Sin importar la tolerancia de las resistencias, el valor sensado va a corresponder al valor correcto de la salida para esa fuente particular.

Además, esta disposición puede ser combinada con el Tip #20 para calibrar a diferentes temperaturas. También puede usarse para crear una fuente de alimentación programable cambiando la referencia suministrada y el divisor resistivo para un "feedback" de tensión.

CAPITULO 8

Soluciones para 3V

Tabla de contenidos

INTRODUCCION

TIP #1:	Powering 3.3V Systems From 5V Using an LDO Regulator	8-3	TIP #8:	3.3V → 5V Using A Voltage Comparator	8-8
TIP #2:	Low-Cost Alternative Power System Using a Zener Diode	8-4	TIP #9:	5V → 3.3V Direct Connect	8-9
TIP #3:	Lower Cost Alternative Power System Using 3 Rectifier Diodes	8-4	TIP #10:	5V → 3.3V With Diode Clamp	8-9
TIP #4:	Powering 3.3V Systems From 5V Using Switching Regulators	8-5	TIP #11:	5V → 3.3V Active Clamp	8-10
TIP #5:	3.3V → 5V Direct Connect	8-6	TIP #12:	5V → 3.3V Resistor Divider.....	8-10
TIP #6:	3.3V → 5V Using a MOSFET Translator	8-6	TIP #13:	3.3V → 5V Level Translators.....	8-12
TIP #7:	3.3V → 5V Using A Diode Offset.....	8-7	TIP #14:	3.3V → 5V Analog Gain Block.....	8-13
			TIP #15:	3.3V → 5V Analog Offset Block.....	8-13
			TIP #16:	5V → 3.3V Active Analog Attenuator ..	8-14
			TIP #17:	5V → 3V Analog Limiter	8-15
			TIP #18:	Driving Bipolar Transistors	8-16
			TIP #19:	Driving N-Channel MOSFET Transistors.....	8-18

TIPS 'N TRICKS INTRODUCTION

Overview - Conexión 3.3 Volt a 5 Volt

Uno de los subproductos de nuestra creciente necesidad de velocidad de procedimiento es la continua reducción en el tamaño de los transistores usada para construir los microcontroladores. La integración a bajo costo también llevan a la necesidad de geometrías más pequeñas. Con tamaños reducidos vienen las reducciones en la tensión de ruptura de los transistores, y últimamente, una reducción en la fuente de alimentación cuando la tensión de ruptura cae debajo de la tensión de suministro. Entonces, a medida que la velocidad aumentan la velocidad y la complejidad, es inevitable que las fuentes de tensión caigan desde 5V a 3.3V, o aún a 1.8V para dispositivos de alta densidad. Los microcontroladores de Microchip han llegado a un nivel suficiente de velocidad y complejidad que también están haciendo la transición a tensiones sub-5V. El desafío es que muchos de los dispositivos de interface siguen siendo diseñados para 5V. Esto significa que los diseñadores tienen que afrontar la tarea de adaptar sistemas de 3.3V y 5V. Además, la tarea no sólo incluye un nivel lógico sino también de alimentar sistemas de 3.3V y trasladando señales analógicas por sobre la barrera 3.3V/5V.

Este folleto enfrenta este desafío con una colección de bloques constructivos de fuentes de alimentación, bloques de adaptación de niveles digitales y además bloques de adaptación analógica. A lo largo del folleto, se presentan múltiples opciones para cada traslado. Se presentan todos los bloques necesarios para un diseñador para manejar el desafío de los 3.3V.

Se puede encontrar información adicional en Microchip web site at www.microchip.com/3volts.

Nota:

Fuentes de Alimentación

Uno de los primeros desafíos de los 3.3V es generar una fuente de alimentación de 3.3V. Dado que se discute la interfaz entre sistemas de 5V y 3.3V, se asume que nuestra V_{dc} es de 5V. Esta sección presenta soluciones para reguladores de tensión diseñados para el traslado de 5V a 3.3V. Un diseño con un modestos requerimientos de corriente puede usar un simple regulador lineal. Corrientes mayores pueden requerir de una solución con fuentes "switching". Aplicaciones sensibles al costo pueden necesitar la simplicidad de un regulador discreto de diodo. Ejemplos de cada una de estas áreas se incluyen aquí, con la información necesaria para adaptar una gran variedad de aplicaciones.

Tabla 1: Comparación Fuentes de Alimentación

Method	V _{REG}	I _q	Eff.	Size	Cost	Transient Response
Zener Shunt Reg.	10% Typ	5 mA	60%	Sm	Low	Poor
Series Linear Reg.	0.4% Typ	1 μ A to 100 μ A	60%	Sm	Med	Excellent
Switching Buck Reg.	0.4% Typ	30 μ A to 2 mA	93%	Med to Lrg	High	Good

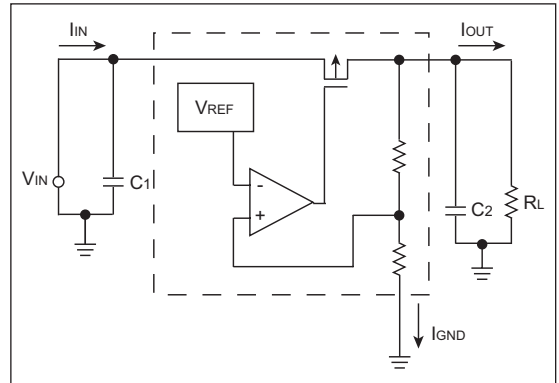
TIP #1 Powering 3.3V Systems From 5V Using an LDO Regulator

La tensión que entrega un regulador lineal estándar de 3 terminales es típicamente 2.0-3.0V. Esto lo imposibilita de ser usado para convertir 5V a 3.3V de forma segura. Reguladores "Low Dropout" (LDO) con una salida en el rango unos pocos cientos de milli volts, son útiles para este tipo de aplicaciones. La Figura 1-1 contiene un diagrama en bloques de un sistema básico LDO con los elementos de corrientes apropiados. De esta figura se puede ver que el LDO consiste de 4 elementos principales:

1. Transistor de paso
2. Referencia "bandgap"
3. Amplificador operacional
4. Feedback de divisor resistivo

Cuando se selecciona un LDO, es importante saber qué distingue un LDO de otro. La corriente de inactividad, el tamaño y el tipo son parámetros importantes. Evaluar cada parámetro para una aplicación específica lleva a un diseño óptimo.

Figura 1-1: LDO Regulador de Tensión

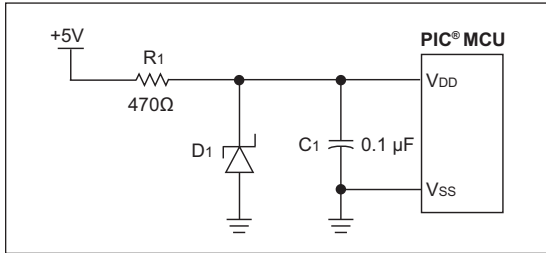


La corriente de inactividad o reposo, I_q , es la corriente de tierra, I_{gnd} , mientras el dispositivo está operando sin carga. I_{gnd} es la corriente usada por el LDO para llevar a cabo la regulación. La eficiencia del LDO puede ser aproximada como la tensión de salida dividida por la tensión de entrada cuando $I_{out} \gg I_q$. Sin embargo, cuando la carga es alta, el I_q debe ser tenido en cuenta cuando se calcula la eficiencia. Un LDO con menor I_q va a tener una mayor eficiencia con poca carga. Este es un efecto negativo en la eficiencia del LDO. Mayor corrientes de reposo del LDO son capaces de responder de manera más rápida a cambios repentinos de la línea y la carga.

TIP #2 Low-Cost Alternative Power System Using a Zener Diode

Detalles de un regulador de bajo costo alternativo usando un diodo Zener.

Figura 2-1: Alimentación con Zener

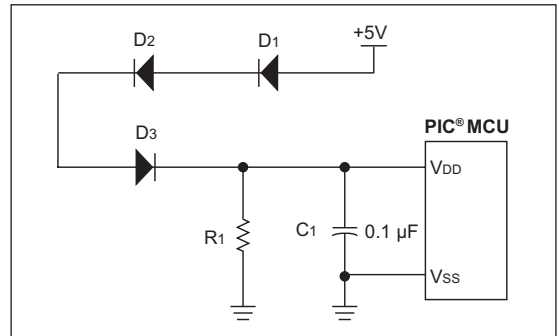


Un regulador de 3.3V simple de bajo costo puede ser construido a partir de un diodo Zener y una resistencia como se muestra en la Figura 2-1. En muchas aplicaciones, este circuito puede ser una alternativa de buen precio a usar un regulador LDO. Sin embargo, este regulador es más sensible a la carga que el regulador LDO. Además, es menos eficiente, ya que siempre se disipa potencia en R1 y D1.

R1 limita la corriente de D1 y la del PIC® MCU de manera que Vdd se mantenga dentro de un rango permitido. Como la tensión de reversa del Zener varía con la corriente que pasa por él, el valor de R1 debe ser cuidadosamente elegido. R1 debe ser elegido de manera que a máxima carga, generalmente cuando el PIC® MCU esta corriendo y su salidas están en "high", la caída de tensión en R1 es lo suficientemente baja para que el PIC® MCU tenga suficiente potencia para operar. Además, R1 tiene que ser elegida para que a mínima carga, generalmente cuando el PIC® MCU esta en "Reset", la Vdd no exceda el valor del diodo de Zener ni el máximo Vdd del PIC® MCU.

TIP #3 Lower Cost Alternative Power System Using 3 Rectifier Diodes

Figura 3-1: Alimentación con diodo



Se puede usar también la caída de directa de una serie de diodos para reducir la tensión que le llega al PIC® MCU. Esto puede ser aún más efectivo con el costo que el regulador con Zener. La corriente entregada por este diseño es menor que usando el circuito con Zener.

El número de diodos necesarios varía según la tensión de directa de los diodos elegidos. La caída entre los diodos D1-D3 es función de la corriente que fluye por ellos. R1 está presente para prevenir que la tensión de Vdd del pin del PIC® MCU exceda el máximo Vdd a cargas mínimas. Dependiendo del circuito conectado a Vdd, esta resistencia puede tener que ser incrementada o posiblemente eliminada. Los diodos D1-D3 deben elegirse de manera que a máxima carga, la caída de tensión en D1-D3 sea lo suficientemente baja para cumplir con los requerimientos del mínimo tolerado por los PIC® MCUs.

TIP #4 Powering 3.3V Systems From 5V Using Switching Regulators

Un regulador "buck switching", mostrado en la Figura 4-1, es un conversor basado en un inductor usado para bajar la tensión de entrada a una magnitud menor de salida. La regulación de la salida se consigue controlando el tiempo de ON del MOSFET Q1. Como el MOSFET esta en estado de "lower resistive" o "high resistive" (ON o OFF respectivamente), una alta tensión puede ser convertida a una menor salida muy eficientemente.

La relación entre la entrada y la salida puede ser establecida balanceando el "volt-time" del inductor durante los dos estados de Q1.

Ecuación 4-1

$$(V_s - V_o) * t_{on} = V_o * (T - t_{on})$$

Donde: $T \equiv t_{on}/Duty_Cycle$

Entonces cumple que el MOSFET Q1:

Ecuación 4-2

$$Duty_Cycle_{Q1} = V_o/V_s$$

Quando se elija el valor del inductor, un buen punto de comienzo es seleccionar el valor para producir una corriente máxima pico a pico de ripple en el inductor igual al 10% de la máxima corriente en la carga.

Ecuación 4-3

$$V = L * (di/dt)$$

$$L = (V_s - V_o) * (t_{on}/I_o * 0.10)$$

Quando se elija un valor para el capacitor de salida, un buen punto de comienzo es setear las impedancia característica del filtro igual a la resistencia de la carga. Esto produce un "overshoot" aceptable cuando se opere con la carga y se remueva abruptamente.

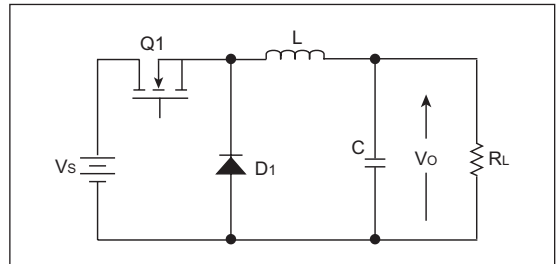
Ecuación 4-4

$$Z_o = \sqrt{L/C}$$

$$C = L/R^2 = (I_o^2 * L)/V_o^2$$

Quando se elija el diodo D1, elija un dispositivo que pueda manejar la corriente del inductor durante el ciclo de descarga.

Figura 4-1: Regulador Buck



Interface Digital

Quando se hace una interface entre dos dispositivos que operan en diferentes voltajes, es importante saber los umbrales de salida y entrada de los dos dispositivos. Una vez que se sepan estos valores, se puede seleccionar una técnica para hacer la interface basada en los otros requerimientos de la aplicación. La Tabla 4-1 contiene los umbrales de entrada y salida que vana a ser usados a lo largo de este documento. Quando se diseñe una interface, asegurese de referirse a los "datasheets" del fabricante correspondiente para averiguar los umbrales.

Tabla 4-1: Umbrales Entrada/Salida

	VOH min	VoL max	VIH min	VIH max
5V TTL	2.4V	0.5V	2.0V	0.8V
3.3V LVTTTL	2.4V	0.4V	2.0V	0.8V
5V CMOS	4.7V (Vcc-0.3V)	0.5V	3.5V (0.7xVcc)	1.5V (0.3xVcc)
3.3V LVCMOS	3.0V (Vcc-0.3V)	0.5V	2.3V (0.7xVcc)	1.0V (0.3xVcc)

TIP #5 3.3V → 5V Direct Connect

La manera más simple y deseada de conectar un salida de 3.3V a una entrada de 5V es por medio de una conexión directa. Esto puede ser llevado a cabo sólo si se cumplen los siguientes 2 requerimientos:

- El valor de V_{oh} de la salida de 3.3V es mayor que el valor de V_{ih} de la entrada de 5V.
- El valor de V_{ol} de la salida de 3.3V es menor que el valor de V_{il} de la entrada de 5V.

Un ejemplo de cuando usar esta técnica puede ser cuando se hace una interface entre una salida 3.3V LVCMOS y una entrada 5V TTL. De los valores que figuran en la Tabla 4-1, se puede ver claramente que se cumplen los 2 requerimientos.

El V_{oh} de 3.3V LVCMOS de 3.0V es mayor que el V_{ih} de 5V TTL de 2.0V, y

El V_{ol} de 3.3V LVCMOS de 0.5V es menor que el V_{il} de 5V TTL de 0.8V.

Cuando ambos de los requerimientos no se cumplen, se necesita de un circuito adicional para hacer de interface entre las dos partes. Vea los Tip #6, #7, #8 y #13 para posibles soluciones.

TIP #6 3.3V → 5V Using a MOSFET Translator

Para manejar cualquier entrada de 5V que tenga un mayor valor de V_{ih} que el V_{oh} de un 3.3V CMOS, se necesita de un circuito adicional. Una solución con dos componentes de bajo costo se muestra en la Figura 6-1.

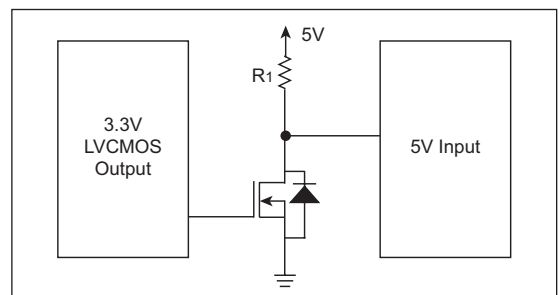
Para seleccionar el valor de la R_1 , hay dos parámetros que deben ser considerados; la velocidad de "switching" de la entrada y el consumo de corriente de R_1 . Cuando se alterna la entrada desde '0' a '1', hay que tener en cuenta el tiempo que tarda la entrada en hacer la transición debido a la constante de tiempo RC formada por R_1 y la capacidad de entrada de la entrada de 5V sumado a cualquier otra capacidad que pueda introducir la placa. La velocidad a la cual se puede alternar la entrada cumple la siguiente ecuación:

Ecuación 6-1

$$T_{SW} = 3 \times R_1 \times (C_{IN} + C_S)$$

Como la capacidad de entrada y la capacidad que pueda introducir la placa son fijas, la única manera de acelerar la velocidad de "switching" es disminuyendo la resistencia R_1 . La contraparte de bajar R_1 es el incremento en la corriente cuando la entrada de 5V permanece en "low". El cambio de '1' a '0' va a ser más rápido ya que la resistencia de ON del "N-channel MOSFET" va a ser mucho menor que R_1 . Además, cuando se seleccione el "N-channel FET", opte por un FET que tenga un umbral V_{gs} menor que el V_{oh} de la salida de 3.3V.

Figura 6-1: MOSFET Translator



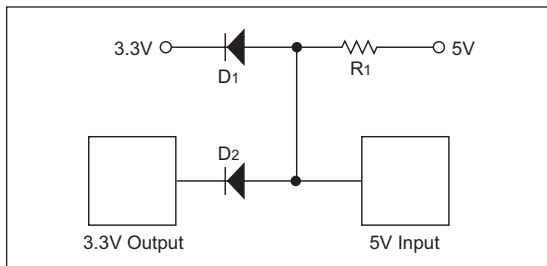
TIP #7 3.3V → 5V Using a Diode Offset

Los umbrales de entrada para un 5V CMOS y las tensiones de salida para un 3.3 LVTTTL y un LVC MOS se muestran en la Tabla 7-1.

Tabla 7-1: Umbrales de Entrada/Salida

	5V CMOS Input	3.3V LVTTTL Output	3.3V LVC MOS Output
High Threshold	> 3.5V	> 2.4V	> 3.0V
Low Threshold	< 1.5V	< 0.4V	< 0.5V

Fíjese que los dos umbrales de entrada ("low" y "high") para un 5V CMOS son un volt mayor que la salida del 3.3V. Por ende, aún si la salida del sistema de 3.3V no puede ser compensada, no va a haber casi ningún margen para ruido o tolerancias de los componentes. Lo que se necesita es un circuito que compense las salidas e incremente la diferencia entre las salidas "low" y "high".

Figura 7-1: Compensación con Diodos

Cuando se determinan las especificaciones de las tensiones de salida, se asume que la salida maneja una carga entre la salida y la tierra para la salida "high", y una carga entre 3.3V y la salida para la salida "low". Si la carga para el umbral superior está entre la salida y 3.3V, entonces la tensión de salida es mucho mayor ya que la resistencia es la que esta llevando a la salida hacia arriba, en vez de que sea el transistor de salida.

Si se crea un circuito compensador con diodos (vea la Figura 7-1) la salida "low" se incrementa una caída de tensión del diodo D1, normalmente de 0.7V, creando una tensión baja en la entrada de 5V CMOS de 1.1V a 1.2V. Esto esta bien dentro del umbral inferior de entrada para el 5V CMOS. La salida "high" está seteada por la resistencia de "pull-up" y el diodo D2, atado al suministro de 3.3V. Esto pone a la salida a aproximadamente 0.7V arriba del suministro de 3.3V, o a 4.0V a 4.1V, que está por encima del umbral de entrada del 5V CMOS que es de 3.5V.

Nota:

Para que el circuito funcione bien, la resistencia de "pull-up" debe ser mucho menor que la resistencia de entrada del 5V CMOS, para prevenir una reducción en la tensión de salida debido al efecto del divisor resistivo a la entrada. La resistencia de "pull-up" debe ser, también, lo suficientemente grande para mantener la corriente de salida del 3.3V dentro de las especificaciones del dispositivo.

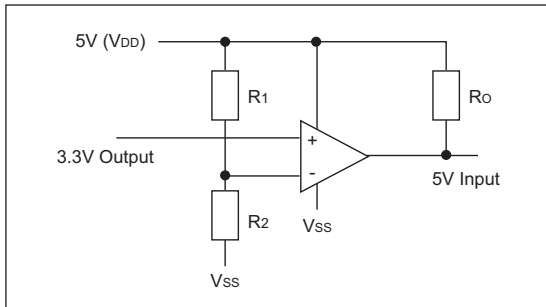
TIP #8 3.3V → 5V Using a Voltage Comparator

La operación básica de un comparador se explica a continuación:

- Cuando la tensión de la entrada inversora (-) es mayor que la tensión de la entrada no inversora (+), la salida del comparador cambia a Vss.
- Cuando la tensión en la entrada no inversora (+) es mayor que la tensión de entrada inversora (-), la salida del comparador está en estado "high".

Para preservar la polaridad de la salida de 3.3V, la salida de 3.3V debe estar conectada a la entrada no inversora del comparador. La entrada inversora se conecta a la tensión de referencia determinada por R1 y R2, como se muestra en la Figura 8-1.

Figura 8-1: Comparator Translator



Calculando R1 y R2

La razón entre R1 y R2 depende de los niveles lógicos de la señal de entrada. La entrada inversora debe estar fijada en una tensión en la mitad de Vol y Voh para la salida de 3.3V. Para una salida LVCMOS, la tensión es:

Ecuación 8-1:

$$1.75V = \frac{(3.0V + .5V)}{2}$$

Dado que R1 y R2 se relacionan por medio de los niveles lógicos.

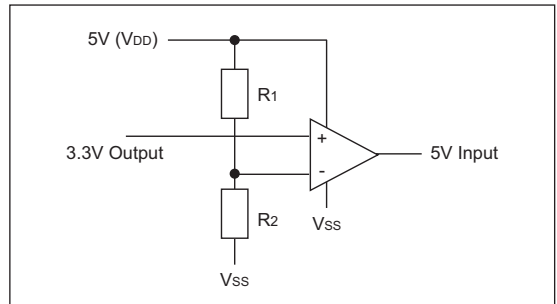
Ecuación 8-2:

$$R1 = R2 \left(\frac{5V}{1.75V} - 1 \right)$$

asumiendo un valor de 1K para R2 y 1.8K para R1. Un AP conectado como un comparador puede ser usado para convertir una señal entrada de 3.3V en una señal de salida de 5V. Esto se lleva a cabo usando una propiedad del comparador que fuerza la salida a cambiar a "high"(Vdd) o "low"(Vss), dependiendo de la magnitud de la diferencia de tensión entre las entradas inversora y no inversora.

Nota: Para que un AO trabaje de manera adecuada cuando se alimente con 5V, la salida de ser capaz de manejar "rail-to-rail"

Figura 8-2: El AO como Comparador

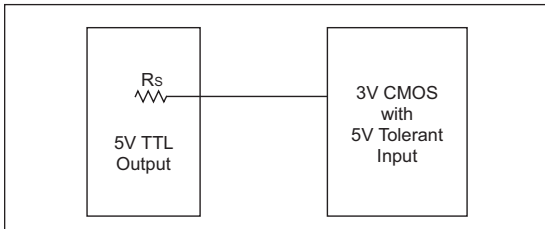


TIP #9 5V → 3.3V Direct Connect

Las salidas de 5V tienen un V_{oh} típico de 4.7V y un V_{ol} de 0.4V y una entrada de 3.3V LVCMOS tienen un valor típico de V_{ih} de $0.7 \times V_{dd}$ y un V_{il} de $0.2 \times V_{dd}$.

Cuando la salida de 5V está en "low", no hay problemas ya que la salida de 0.4V es menor que el umbral de entrada de 0.8V. Cuando la salida de 5V está en "high", el V_{oh} de 4.7V es mayor que los 2.1V del V_{ih} , por ende, se puede conectar directamente los dos pines sin ningún problema si la entrada del 3.3V CMOS es tolerante a 5V.

Figura 9-1: Entrada Tolerante 5V



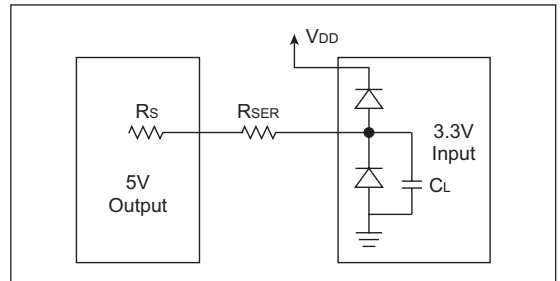
Si la entrada del 3.3V CMOS no es tolerante a los 5V, entonces sería un problema ya que se estaría excediendo la máxima tensión especificada.

Vea los Tips #10 y #13 para posibles soluciones.

TIP #10 5V → 3.3V With Diode Clamp

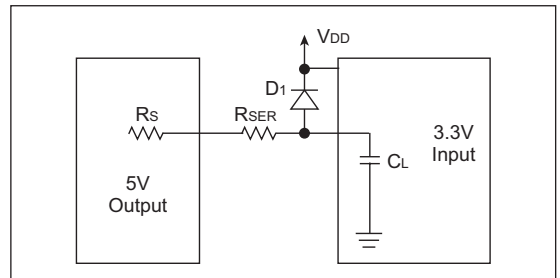
Muchos fabricantes protegen los pines I/O de que se supere la máxima tensión permitida usando diodos "clamping". Estos diodos evitan que la tensión del pin caiga más que una caída de diodo por debajo de V_{ss} y una caída de diodo por encima de V_{dd} . Para usar diodos "clamping" para proteger la entrada, se debe tener en cuenta la corriente que fluya por ese diodo. La corriente que atraviesa el diodo "clamping" debe mantenerse pequeña (en el orden de los μA). Si la corriente llegara a ser muy grande se corre el riesgo de arruinarlos. Como la resistencia de la fuente de 5V está aproximadamente alrededor de los 10Ω , se necesitan resistencias adicionales en serie para limitar la corriente en los diodos "clamping" como se muestra en la Figura 10-1. La consecuencia de usar estas resistencias es que reducen la velocidad de "switching" de la entrada debido a la constante de tiempo RC presente en la entrada.

Figura 10-1: Entrada con Diodos Clamping



Si los diodos "clamping" no están presentes, se puede agregar un simple diodo externo como se muestra en la Figura 10-2.

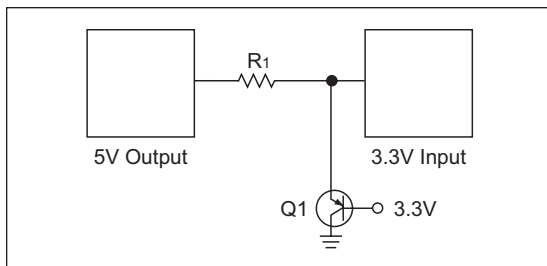
Figura 10-2: Sin Diodos Clamping



TIP #11 5V → 3.3V Active Clamp

Un problema de usar diodos "clamping" es que inyectan corriente en la fuente de alimentación de 3.3V. En diseños donde la salida de 5V tienen una corriente alta y las fuentes de alimentación de 3.3V están poco cargadas, esta corriente inyectada puede hacer la tensión de 3.3V suba. Para prevenir este problemas, se puede utilizar un transistor el cual dirige el exceso de corriente a tierra en vez de que vaya a la fuente de 3.3V. La Figura 11-1 muestra el circuito.

Figure 11-1: Transistor Clamp

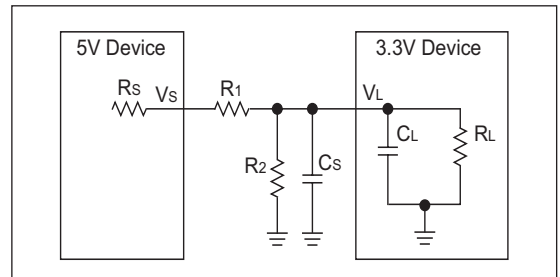


La juntura base emisor de Q1 lleva a cabo la misma función que el diodo "clamping". La diferencia es que sólo un porcentaje de la corriente de emisor sale de la base del transistor hacia los 3.3V, la mayoría de la corriente es dirigida hacia el colector donde se dirige directamente a tierra sin provocar daños. La razón de corriente de base sobre la corriente de colector se dicta por la ganancia de corriente del transistor, normalmente de 10-400, dependiendo de que transistor se use.

TIP #12 5V → 3.3V Resistor Divider

Se puede usar un simple divisor resistivo para reducir la salida del dispositivo de 5V a niveles apropiados para una entrada a un dispositivo de 3.3V. Un circuito equivalente con esta interface se muestra en la Figura 12-1.

Figura 12-1: Circuito de Interface Resistivo Equivalente



Típicamente, la resistencia de la fuente, R_s , es muy pequeña (menos de 10Ω) y su efecto en R_1 va a ser despreciable siempre y cuando se elija R_1 mucho mayor que R_s . La resistencia de carga, R_L , es bastante grande (mayor que $500k\Omega$) y su efecto en R_2 va a ser despreciable siempre y cuando R_2 se elija mucho menor que R_L .

Existe un compromiso entre la disipación de potencia y los tiempo de transición. Para mantener los requerimientos de potencia del circuito de interface al mínimo, la resistencias en serie R_1 y R_2 debe ser lo más grande posible. Sin embargo, la capacidad e carga, que es la combinación de la capacidad de la placa, C_s , y la capacidad de entrada al dispositivo de 3.3V, C_L , puede afectar el "rise time" y "fall time" de la señal de entrada. Estos tiempos pueden ser inaceptablemente grandes si R_1 y R_2 son muy grandes.

Despreciando los efectos de R_s y R_L , la fórmula para determinar los valores de R_1 y R_2 está dada por la Ecuación 12-1.

Ecuación 12-1: Divider Values

$$\frac{V_s}{R_1 + R_2} = \frac{V_L}{R_2} \quad ; \text{ General relationship}$$

$$R_1 = \frac{(V_s - V_L) \cdot R_2}{V_L} \quad ; \text{ Solving for } R_1$$

$$R_1 = 0.515 \cdot R_2 \quad ; \text{ Substituting voltages}$$

La fórmula para determinar los tiempos de "rise" y "fall" está dada por la Ecuación 12-2. Para un análisis de circuito, se usa el equivalente de Thevenin para determinar la tensión aplicada, V_a , y la resistencia en serie R . El equivalente de Thevenin se define como un circuito abierto dividido por un corto circuito. El equivalente de Thevenin, R , se determina haciendo $0.66 \cdot R_1$ y el valor de V_a se determina haciendo $0.66 \cdot V_s$ para el circuito que se muestra en la Figura 12-2 de acuerdo a las limitaciones impuestas en la Ecuación 12-2.

Ecuación 12-2: Rise/Fall Time

$$t = - \left[R \cdot C \cdot \ln \left(\frac{V_F - V_A}{V_I - V_A} \right) \right]$$

Where:

- t = Rise or Fall time
- R = $0.66 \cdot R_1$
- C = $C_s + C_L$
- V_i = Initial voltage on C (V_L)
- V_f = Final voltage on C (V_L)
- V_a = Applied voltage ($0.66 \cdot V_s$)

Como ejemplo suponga los siguientes valores:

- Stray capacitance = 30 pF
- Load capacitance = 5 pF
- Máximo "rise time" desde 0.3V to $3V \leq 1 \mu S$
- Tensión $V_s = 5V$

El cálculo para determina las resistencias máximas se muestra en la Ecuación 12-3.

Equation 12-3: Example Calculation

Resuelva la Ecuación 12-2 para R :

$$R = - \left[\frac{t}{C \cdot \ln \left(\frac{V_F - V_A}{V_I - V_A} \right)} \right]$$

Reemplace los valores:

$$R = - \left[\frac{10 \cdot 10^{-7}}{35 \cdot 10^{-12} \cdot \ln \left(\frac{3 - (0.66 \cdot 5)}{0.3 - (0.66 \cdot 5)} \right)} \right]$$

Equivalente Thevenin máximo R :

$$R = 12408$$

Resuelva para el máximo R_1 y R_2 :

$$R_1 = 0.66 \cdot R \qquad R_2 = \frac{R_1}{0.515}$$

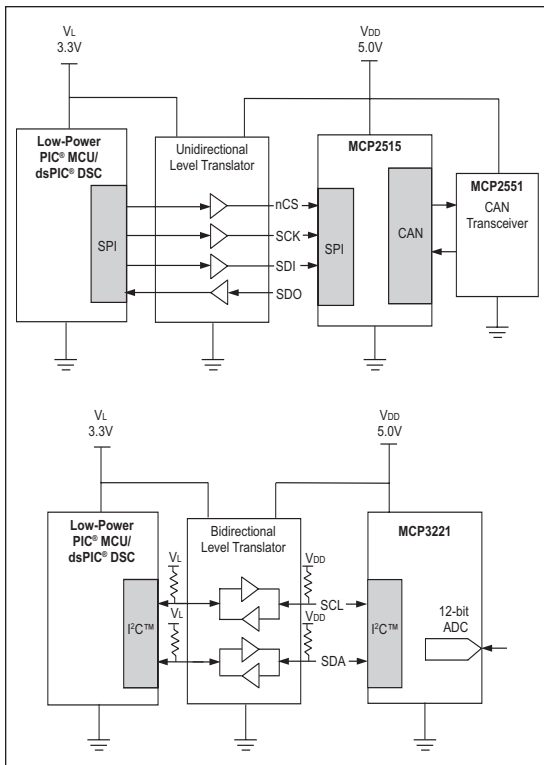
$$R_1 = 8190 \qquad R_2 = 15902$$

TIP #13 3.3V → 5V Level Translators

Mientras el traspaso de niveles se puede hacer de forma discrta, se prefiere usar una solución integrada. "Level translators" están disponibles en un amplio rango de capacidades. Hay unidireccionales y bidireccionales, diferentes tensiones y velocidades, que permiten al usuario optar la mejor solución.

La comunicacion a nivel de placa entre dispositivos (ej. MCU hacia periférico) se efectúa por SPI o I2C. Para SPI, puede ser apropiado usar un "level translator" unidireccional y para I2C conviene uno bidireccional. La Figura 13-1 muestran ambas soluciones.

Figura 13-1: Level Translator



Analógico

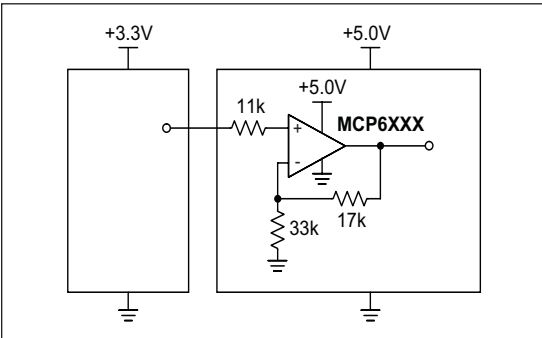
El desafío final de la interface entre 3.3V y 5V es la adaptación de señales analógicas a través de la barrera de la fuente de alimentación. Mientras señales "low" no van a requerir de circuitos externos, las señales entre sistemas de 3.3V y 5V van a estar afectadas por la fuente de alimentación. Por ejemplo, una señal de 1V de pico analógica convertida por el ADC en un sistema de 3.3V va a tener una resolución mejor que en el ADC de un sistema de 5V, simplemente porque se usa un rango mayor del ADC para convertir la señal en el 3.3V ADC.

Alternativamente, la amplitud de una señal mayor en un sistema de 3.3V puede tener problemas con las limitaciones de tensión de modo común. Es por esta razón, que se necesita de un circuito de interface para compensar las diferencias. Esta sección discute los circuitos que pueden ayudar a aliviar estos problemas cuando la señal hace transiciones entre diferentes fuentes de alimentación.

TIP #14 3.3V → 5V Analog Gain Block

Para adaptar la tensión analógica cuando sale del sistema 3.3V a 5V, las resistencias de 33kΩ y 17kΩ fijan la ganancia del AO para que use un rango completo hacia los dos lados. La resistencia de 11kΩ limita la corriente que vuelve al circuito de 3.3V.

Figura 14-1: Bloque de Ganancia Analógica

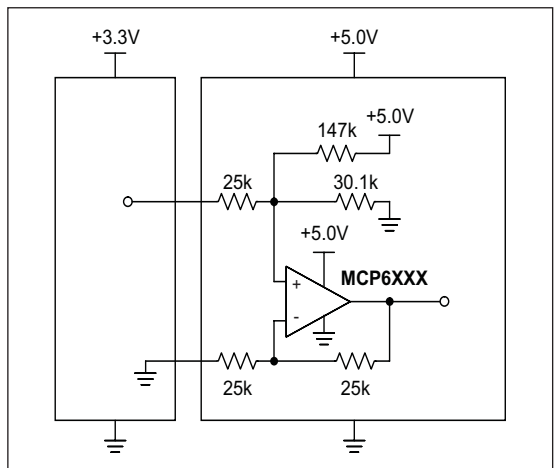


TIP #15 3.3V → 5V Analog Offset Block

Compensando una tensión analógica para el traspaso entre 3.3V y 5V.

Cambie la tensión analógica de 3.3V a 5V. La R de 147kΩ y de 30.1kΩ de arriba a la derecha y los +5V de suministro son equivalentes a la fuente de 0.85V en serie con la resistencia de 25kΩ. Esta resistencia equivalente de 25kΩ, las tres resistencias de 25kΩ, y el AP forman un amplificador de diferencia de 1V/V. La fuente de 0.85V equivalente sube a cualquier señal vista a la entrada en la misma cantidad; las señales centradas en $3.3V/2=1.65V$ van a estar centradas en $5.0V/2=2.50V$. La resistencia de arriba a la izquierda limita la corriente por el circuito de 5V.

Figure 15-1: Bloque Offset Analógico



TIP #16 5V → 3.3V Active Analog Attenuator

Reducir la amplitud de una señal desde un sistema de 5V a 3.3V usando un AO.
El método más simple de convertir una señal analógica de 5V a una señal de 3.3V es usando un divisor resistivo con una razón de $R1:R2$ de 1.7:3.3. Sin embargo, existen algunos problemas con esto:

1. El atenuador puede estar alimentando una carga capacitiva, creando un filtro pasa bajos sin intención.
2. El circuito atenuador puede necesitar manejar una carga de baja impedancia hasta una alta impedancia.

En cualquiera de estas condiciones, se necesita un AO para hacer de "buffer".
El circuito AO necesario es un seguidor de ganancia unitaria (ver Figura 16-1).

Figura 16-1: Ganancia Unitaria

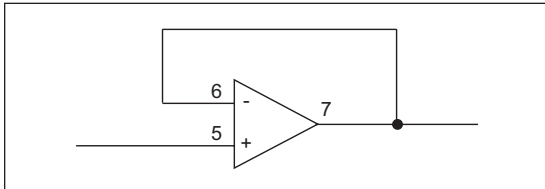
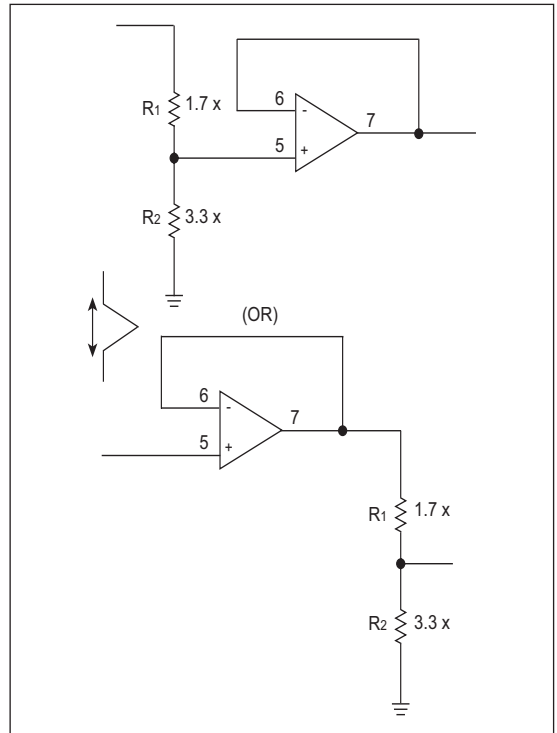


Figura 16-2: AO Atenuadores



Si el divisor resistivo está antes del seguidor de ganancia unitaria, entonces se provee la impedancia más baja posible para los circuitos de 3.3V. Además, el AO puede ser alimentado desde 3.3V, ahorrando potencia. Si la X es muy grande, se puede minimizar la potencia consumida por los 5V.

Si el atenuador se agrega después del seguidor de ganancia unitaria, entonces se presenta la mayor impedancia posible a la fuente de 5V. El AO debe ser alimentado desde los 5V y la impedancia en los 3V va a depender del valor de $R1 || R2$.

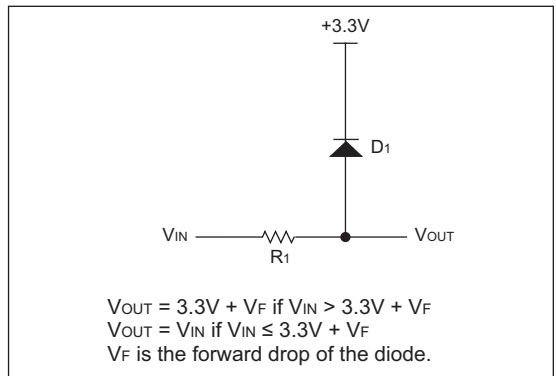
TIP #17 5V → 3V Analog Limiter

Cuando se mueve una señal de 5V hacia un sistema de 3.3V, a veces es posible usar un atenuador de ganancia. Si la señal deseada es menor que 5V, mandar esa señal a un conversor A/D de 3.3V va a resultar en valores de conversión más grandes. El peligro es cuando la señal se acerca a los 5V. Se requiere, entonces, un método para controlar las tensiones fuera de rango sin afectar las tensiones dentro del rango. Las tres maneras de lograr esto se discuten a continuación.

1. Usando un diodo para fijar la sobretensión al suministro de 3.3V.
2. Usando un diodo Zener para fijar la tensión en cualquier limite deseado.
3. Usando un AO con un diodo para fijar la tensión con mayor precisión.

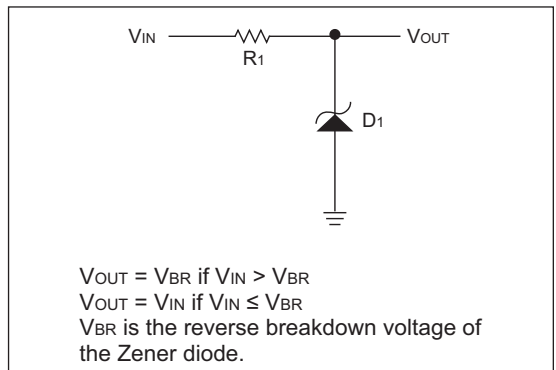
El método más simple para llevar a cabo esto es idéntico a método simple de hacer una interface para una señal digital de 5V a una de 3.3V. Se usan un diodo y una resistencia para direccionar el exceso de corriente a la fuente de 3.3V. La resistencia debe tener el valor adecuado para proteger al diodo y la fuente de 3.3V y que no afecte la rendimiento analógico. Si la impedancia de la fuente de 3.3 es muy baja, entonces este tipo de fijación puede causar que la tensión de 3.3V aumente. Aún si el suministro de 3.3V tiene una buena baja impedancia, este tipo de fijación va a permitir que la señal de entrada agregue ruido a la fuente de 3.3V cuando el diodo está conduciendo y si la frecuencia es lo suficientemente alta, aún cuando el diodo no esté conduciendo debido a la capacidad parásita en el diodo.

Figure 17-1: Diode Clamp



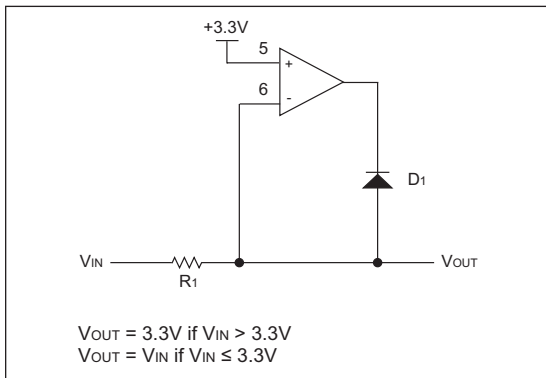
Para prevenir que la señal de entrada afecte el suministro o para hacer la entrada más robusta a cambios transitorios, una variante es usar un diodo Zener. El diodo Zener es más lento que el diodo usado en el primer circuito. Sin embargo, son generalmente más robustos y no dependen de las características de la fuente de alimentación para llevar a cabo esta fijación. Ésta última va a depender de la corriente que fluya por el diodo. Esto se setea con el valor de R1. R1 no es necesaria si la impedancia de salida de la fuente de Vin es lo suficientemente grande.

Figure 17-2: Zener Clamp



Si se requiere una fijación de sobretensión más precisa que no dependa del suministro, entonces se puede usar un AO para crear un diodo de precisión. En la Figura 17-3, se muestra un circuito de este estilo. El AO compensa la caída de directa en el diodo y causa que la tensión se fije en la tensión exacta de la entrada no inversora del AO. El AO puede ser alimentado desde 3.3V si es "rail-to-rail".

Figura 17-3: Precision Diode Clamp

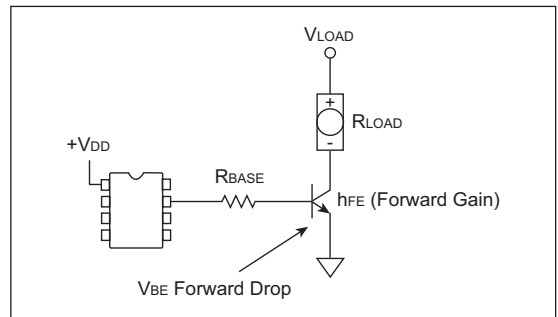


Como la fijación es llevada a cabo por el AO, no hay efectos de la fuente de alimentación. La impedancia presente en el circuito de baja tensión no es mejorada or el AO, se mantiene R1 sumado a la impedancia de la fuente.

TIP #18 Driving Bipolar Transistors

Cuando se manejan transistores bipolares, la cantidad de corriente de la base y la ganancia de corriente (B/hFE) van a determinar cuanto corriente puede fluir. Cuando se maneja por medio de un puerto I/O de un MCU, la corriente de base es calculada usando los límites de tensión y corriente del puerto(normalmente 20mA). Cuando se usa una tecnología de 3.3V, se deben usar resistencias menores de limitación de corriente de base para asegurar la corriente suficiente para saturar el transistor.

Figura 18-1: Manejando Transistores Bipolares Usando Puertos I/O de un MCU



El valor de Rbase va a depender de la tensión de suministro del microcontrolador. La Ecuación 18-1 describe como calcular Rbase.

Tabla 18-1: Especificaciones de Transistores Bipolares DC

Characteristic	Sym	Min	Max	Unit	Test Condition
OFF CHARACTERISTICS					
Collector-Base Breakdown Voltage	$V_{(BR)CBO}$	60	–	V	$I_C = 50 \mu A$, $I_E = 0$
Collector-Emitter Breakdown Voltage	$V_{(BR)CEO}$	50	–	V	$I_C = 1.0$ mA, $I_B = 0$
Emitter-Base Breakdown Voltage	$V_{(BR)EBO}$	7.0	–	V	$I_E = 50 \mu A$, $I_C = 0$
Collector Cutoff Current	I_{CBO}	–	100	nA	$V_{CB} = 60V$
Emitter Cutoff Current	I_{EBO}	–	100	nA	$V_{EB} = 7.0V$
ON CHARACTERISTICS					
DC Current Gain	h_{FE}	120 180 270	270 390 560	–	$V_{CE} = 6.0V$, $I_C = 1.0$ mA
Collector-Emitter Saturation Voltage	$V_{CE(SAT)}$	–	0.4	V	$I_C = 50$ mA, $I_B = 5.0$ mA

Quando se usan transistores bipolares como interruptores para encender y apagar cargas controladas por el pin del puerto de un MCU, use el mínimo HFE y asegúrese de que llegue a la saturación completa.

Ecuación 18-1: Calculando la resistencia de Base

$$R_{BASE} = \frac{(V_{DD} - V_{BE}) \times h_{FE} \times R_{LOAD}}{V_{LOAD}}$$

Ejemplo Tecnología 3V

$V_{DD} = +3V$, $V_{LOAD} = +40V$, $R_{LOAD} = 400\Omega$,
 $h_{FE} \text{ min.} = 180$, $V_{BE} = 0.7V$

$R_{BASE} = 4.14 \text{ k}\Omega$, I/O port current = 556 μA

Ejemplo Tecnología 5V

$V_{DD} = +5V$, $V_{LOAD} = +40V$, $R_{LOAD} = 400\Omega$,
 $h_{FE} \text{ min.} = 180$, $V_{BE} = 0.7V$

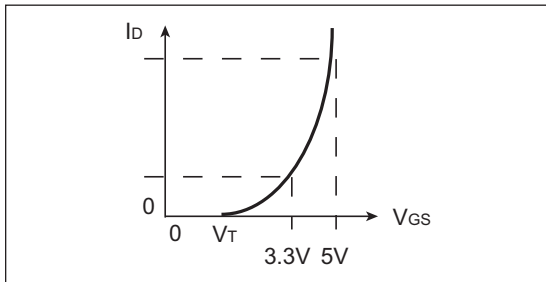
$R_{BASE} = 7.74 \text{ k}\Omega$, I/O port current = 556 μA

Para ambos ejemplos, es una buena práctica incrementar la corriente de base hasta el margen. Manejando la base con 1mA hasta 2mA va asegura la saturación a costas de un aumento en el consumo de potencia.

TIP #19 Driving N-Channel MOSFET Transistors

Se debe tener cuidado cuando se elije el "N-Channel MOSFET" para usar con un microcontrolador de 3.3V. El umbral del MOSFET es una indicacion de la capacidad del dispositivo para llegar a la saturación completa. Para aplicaciones de 3.3V, elija un MOSFET con una "ON resistance rating for gate drive" de 3V o menor. Por ejemplo, un FET de 250µA de corriente con 1V aplicado desde "gate-to-source" no va a entregar necesariamente resultados satisfactorios para 100mA cargados con un "drive" de 3.3V. Cuando se cambia de una tecnología de 5V a una de 3V, revise el umbral de "gate-to-source" y las características de "ON resistance" cuidadosamente como se muestra en la Figura 19-1. Un pequeño decremento en la tension del "gate drive" puede reducir la corriente significativamente

Figura 19-1: Drain Current Capability Versus Gate to Source Voltage



Umbral inferiores existen comunmente para dispositivos MOSFET con tensiones de "drain-to-source" menores a 30V. Las tensiones de MOSFET "drain-to-source" por encima de 30V normalmente tienen umbrales mayores (VT).

Tabla 19-1: Rds(ON) y Vgs(th) Especificaciones para IRF7467

Rds(on)	Static Drain-to-Source On-Resistance	-	9.4	12	mΩ	V _{GS} = 10V, I _D = 11A
		-	10.6	13.5		V _{GS} = 4.5V, I _D = 9.0A
		-	17	35		V _{GS} = 2.8V, I _D = 5.5A
Vgs(th)	Gate Threshold Voltage	0.6	-	2.0	V	V _{DS} = V _{GS} , I _D = 250 □A

Como se muestra en la Tabla 19-1, la tensión umbral para estos 30V N-Channel MOSFET es 0.6V. La resistencia para el MOSFET es 35 mΩ con 2.8V "applied gate", como resultado, este dispositivo es adecuado para aplicaciones de 3.3V

Tabla 19-2: Rds(ON) y Vgs(th) Especificaciones para IRF7201

Rds(on)	Static Drain-to-Source On-Resistance	-	-	0.030	Ω	V _{GS} = 10V, I _D = 7.3A
		-	-	0.050		V _{GS} = 4.5V, I _D = 3.7A
Vgs(th)	Gate Threshold Voltage	1.0	-	-	V	V _{DS} = V _{GS} , I _D = 250 □A

Para las especificaciones en los "datasheets" del IRF7201, la tensión de "gate threshold" es especificada como 1V mínimo. Esto no significa que el dispositivo pueda ser usado para alternar corriente con 1.0V de tension "gate-to-source" como no hay especificaciones de RDS(ON) para Vgs(th) menores a 4.5V. Este dispositivo no se recomienda para manejar aplicaciones de 3.3V que requieran de "low switch resistance", pero puede ser usados para manejar aplicaciones de 5V.

NOTES:

Sales Office Listing

AMERICAS

Atlanta

Tel: 678-957-9614

Boston

Tel: 774-760-0087

Chicago

Tel: 630-285-0071

Cleveland

Tel: 216-447-0464

Dallas

Tel: 972-818-7423

Detroit

Tel: 248-538-2250

Kokomo

Tel: 765-864-8360

Los Angeles

Tel: 949-462-9523

Santa Clara

Tel: 408-961-6444

Toronto

Mississauga, Ontario

Tel: 905-673-0699

EUROPE

Austria - Wels

Tel: 43-7242-2244-39

Denmark - Copenhagen

Tel: 45-4450-2828

France - Paris

Tel: 33-1-69-53-63-20

Germany - Munich

Tel: 49-89-627-144-0

Italy - Milan

Tel: 39-0331-742611

Netherlands - Drunen

Tel: 31-416-690399

Spain - Madrid

Tel: 34-91-708-08-90

UK - Wokingham

Tel: 44-118-921-5869

ASIA/PACIFIC

Australia - Sydney

Tel: 61-2-9868-6733

China - Beijing

Tel: 86-10-8528-2100

China - Chengdu

Tel: 86-28-8665-5511

China - Hong Kong SAR

Tel: 852-2401-1200

China - Nanjing

Tel: 86-25-8473-2460

China - Qingdao

Tel: 86-532-8502-7355

China - Shanghai

Tel: 86-21-5407-5533

China - Shenyang

Tel: 86-24-2334-2829

China - Shenzhen

Tel: 86-755-8203-2660

China - Wuhan

Tel: 86-27-5980-5300

China - Xiamen

Tel: 86-592-2388138

China - Xian

Tel: 86-29-8833-7252

China - Zhuhai

Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore

Tel: 91-80-4182-8400

India - New Delhi

Tel: 91-11-4160-8631

India - Pune

Tel: 91-20-2566-1512

Japan - Yokohama

Tel: 81-45-471- 6166

Korea - Daegu

Tel: 82-53-744-4301

Korea - Seoul

Tel: 82-2-554-7200

Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857

Malaysia - Penang

Tel: 60-4-227-8870

Philippines - Manila

Tel: 63-2-634-9065

Singapore

Tel: 65-6334-8870

Taiwan - Hsin Chu

Tel: 886-3-572-9526

Taiwan - Kaohsiung

Tel: 886-7-536-4818

Taiwan - Taipei

Tel: 886-2-2500-6610

Thailand - Bangkok

Tel: 66-2-694-1351

1/30/07



MICROCHIP
www.microchip.com

Microchip Technology Inc. • 2355 W. Chandler Blvd. • Chandler, AZ 85224-6199

Microcontrollers • Digital Signal Controllers • Analog • Serial EEPROMs

Information subject to change. The Microchip name and logo, the Microchip logo, dsPIC, MPLAB, PIC, PICmicro and PICSTART are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. FilterLab, MXDEV and MXLAB are registered trademarks of Microchip Technology Incorporated in the U.S.A. ICSP, MPASM, MPLIB, MPLINK, PICkit, PICDEM, PICDEM.net and PICtail are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. SQTP is a service mark of Microchip Technology Incorporated in the U.S.A. All other trademarks mentioned herein are property of their respective companies. © 2008, Microchip Technology Incorporated. All Rights Reserved. Printed in the U.S.A. 2/08