

CURSO DE MICROCONTROLADORES PIC16F87X (...y IV)

Fernando Remiro Domínguez
Profesor de Sistemas Electrónicos
IES. Juan de la Cierva
www.terra.es/personal/fremiro

LOS REGISTROS PCL y PCLATH

El contador del programa (PC) está formado por 13 bit que sirven para direccionar la memoria de código, estos bits se encuentran en dos registros específicos. El byte bajo viene del registro de PCL que puede ser leído y escrito. Los bits superiores (PC<12:8>). Están alojados en el registro PCH, sobre el que no se puede leer ni escribir, pero se puede acceder a él indirectamente a través del registro PCLATH.

Las instrucciones de salto CALL y GOTO sólo proporcionan 11 bits de la dirección a saltar. Esto limita el salto dentro de cada Banco de 2K. Cuando se desea salir del Banco actual hay que programar

correctamente los bits PCLATH <4:3> que seleccionan el Banco. Es labor del programador modificar el valor de dichos bits en las instrucciones CALL y GOTO.

La familia de los PIC 16F87X dispone de una pila de 8 niveles de profundidad para un tamaño de PC de 13 bits. Esta pila es transparente al programador, es decir, funciona automáticamente y no dispone de instrucciones para guardar o sacar de ella información.

Con la Instrucción CALL y con las interrupciones el valor de PC se salva en el nivel superior. Con las instrucciones RETURN, RETLW y RETFIE el valor contenido en el valor superior de la pila se carga en el PC.

La pila funciona como un buffer circular. Esto significa que después de que se han guardado 8 valores en ella, el noveno borra el valor que se guardó en primer lugar, el décimo borra el que se guardó en segundo lugar, etc.

Paginación de la Memoria de Programa

Los dispositivos de la familia PIC 16F87X son capaces de direccionar un bloque de hasta 8K de memoria continua. Las instrucciones CALL y GOTO proporcionan solo 11 bits de la dirección de memoria, lo que permite un salto de bifurcación dentro de una página de 2K de la memoria de programa.

Al hacer un CALL o una instrucción GOTO, dos 2 bit superiores de la dirección son proporcionados por el PCLATH <4:3>. Al hacer un CALL o un GOTO, el usuario debe asegurarse de que la página se han seleccionados con los bit correspondientes a la página que se desea llamar. Hay que tener precaución en los retornos de subrutinas y retornos de interrupción para no salirse con el valor almacenado en la PILA.

Por consiguiente, manipulación del PCLATH <4:3> no se requieren los bits para las instrucciones del retorno. Este es el caso del ejemplo Inter.asm, en el que el programa principal, que consiste en encender y apagar un LED, conectado a través de una

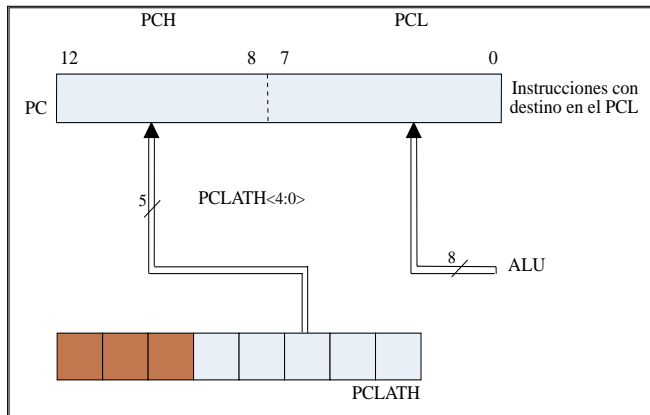


Figura 1.- Estructura del Contador de Programa

```

;*****
; Programa      INTER.asm                      Fecha : 24- Enero-2003
; Este programa genera una intermitencia en un IED conectado a la línea RB0. El programa principal
; está en la pagina 0 de memoria de programa y la subrutina en la página 1 de programa
; Revisión 0.0                                Programa para PIC16F87X
; Velocidad del Reloj: 4 MHz                  Reloj Instrucción: 1 MHz = 1 uS
; Perro Guardián :deshabilitado              Tipo de Reloj : XT
; Protección del código : OFF
;*****
List      p=16F876          ;Tipo de procesador
include  "P16F876.INC"     ;Definiciones de registros internos
;__config__CP_OFF & _DEBUG_OFF & _WRT_ENABLE_ON & _CPD_OFF & _LVP_OFF & _XT_OSC & _WDT_OFF
;
; org      0x00             ;Vector de Reset
; goto    INICIO
CONTA1   EQU  0x20
CONTA2   EQU  0x21
PCLATH_TEMP EQU  0x22

INICIO   org      0x05          ;Salva el vector de interrupción
         clrf    PORTB         ;Borra los latch de salida
         bsf    STATUS,RP0     ;Selecciona banco 1
         movlw  b'00000110'
         movwf  ADCON1         ;Puerta A/E/S digitales
         clrf   TRISB          ;Puerta B se configura como salida
         bcf    STATUS,RP0     ;Selecciona banco 0

BUCLE   org      0x05          ;Salva el vector de interrupción
         bsf    PORTB,0
         movf   PCLATH,W       ;Salvamos valor de PCLARH en un registro temporal
         movwf  PCLATH_TEMP
         bsf    PCLATH,3       ;Selecciona la Página 1
         bcf    PCLATH,4
         call   TEMPO
         bcf    PORTB,0
         movf   PCLATH,W
         movwf  PCLATH_TEMP
         bsf    PCLATH,3       ;Selecciona la Página 1
         bcf    PCLATH,4
         call   TEMPO
         goto   BUCLE

TEMPO   org      0x0800        ;Dirección de Inicio de la Página 1
         clrf   CONTA1        ;Borra el contenido de counter1
         clrf   CONTA2        ;Borra el contenido de counter2
BUCLE_1 decfsz  CONTA1,f      ;resta 1 al contenido de CONTA1.si CONTA1 llega a
                                ;cero: salta la instrucción GOTO BUCLE1
                                ;Si counter1 no llega a cero: ejecuta la siguiente instrucción
         goto   BUCLE_1      ;Cierra el primer bucle de retardo
         decfsz CONTA2,f      ;Lo mismo que el caso anterior, pero esta vez aplicado a CONTA2
         goto   BUCLE_1      ;Cierra el segundo bucle de retardo
         movf   PCLATH_TEMP,W ;Repone el valor de PCLATH antes de la subrutina
         movwf  PCLATH
         return                ;Retorno de subrutina

org      0x1F00
         bcf    PCLATH,4
         bcf    PCLATH,3      ;Selecciona la página 0
         goto   INICIO
         end                    ;Fin del programa fuente
                                Programa Inter1.asm

```

resistencia de 330 W a la línea RB0, cada cierto tiempo, el programa principal se ejecuta en la página cero y la subrutina de temporización se encuentra en la página 1 de memoria de programa.

EL STACK

La familia de los PIC 16F87X dispone de una pila de 8 niveles de profundidad para un tamaño de PC de 13 bits. Esta pila es transparente al programador, es decir, funciona automáticamente y no dispone de instrucciones para guardar o sacar de ella información.

Con la Instrucción CALL y con las interrupciones el valor de PC se salva en el nivel superior. Con las instrucciones RETURN, RETLW y RETFIE el valor contenido en el valor superior de la pila se carga en el PC.

La pila funciona como un buffer circular. Esto significa que después de que se han guardado 8 valores en ella, el noveno borra el valor que se guardó en primer lugar. El décimo borra el que se guardó en segundo lugar, etc.

Paginación de la Memoria de Programa

Los dispositivos de la familia PIC 16F87X son capaces de direccionar un bloque de hasta 8K de memoria continua. Las

instrucciones CALL y GOTO proporcionan solo 11 bits de la dirección de memoria, lo que permite un salto de bifurcación dentro de una página de 2K de la memoria de programa.

Al hacer un CALL o una instrucción GOTO, dos 2 bit superiores de la dirección son proporcionados por el PCLATH <4:3>. Al hacer un CALL o un GOTO, el usuario debe asegurarse de que la página se han seleccionados con los bit correspondientes a la página que se desea llamar. Hay que tener precaución en los retornos de subrutinas y retornos de interrupción para no salirse con el valor almacenado en la PILA.

Por consiguiente, manipulación del PCLATH <4:3> no se requieren los bits para las instrucciones del retorno. Este es el caso del ejemplo Inter.asm, en el que el programa principal, que consiste en encender y apagar un LED, conectado a través de una resistencia de 330 Ω a la línea RB0, cada cierto tiempo, el programa principal se ejecuta en la página cero y la subrutina de temporización se encuentra en la página 1 de memoria de programa.

El programa está diseñado para aquellos que utilicen el bootloader con el circuito que presentamos en el número anterior

o un programador con el IC-PROG, tal y como está es válido para los primeros y si se quitan los ";", se puede cargarse directamente utilizando cualquiera de los demás programadores, estas líneas de programa se han marcado en azul. En rojo se han marcado las líneas de programa utilizadas para salvar el valor de registro PCLATH en un registro temporal, antes de manipular PCLATH <4:3>, cargando en el bit 4 y 3 del mismo, respectivamente los valores 0 y 1, que seleccionan la página 1 de memoria de programa.

Por último, en la subrutina que empieza en la página 1, en la dirección 8000h, se ha marcado en verde las instrucciones que reponen el valor que tenía el PCLATH antes de saltar a la subrutina.

Direccionamiento indirecto, los registros INDF y FSR

El registro INDF no es un registro físico, que ocupa la dirección 0 en todos los Bancos, este registro se utiliza cuando queremos realizar un direccionamiento indirecto. Cuando se hace referencia al registro, se accede realmente a la dirección de un Banco de registro especificada con los 7 bits de menor peso del registro FSR. El bit de más peso del FSR junto al

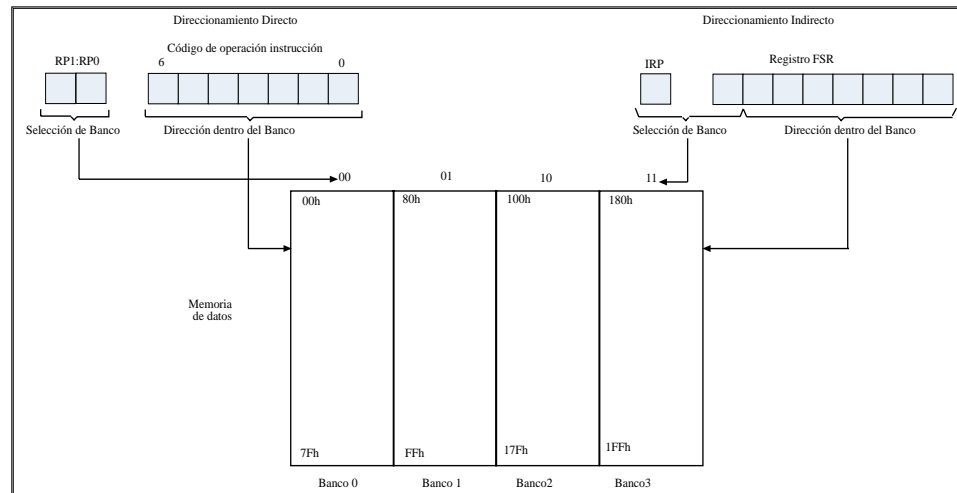


Figura 2.-Forma de seleccionar el Banco y la dirección de la RAM en los direccionamientos directo e indirecto.

bit IRP de registro de STATUS se encarga de seleccionar el Banco a acceder, mientras que los 7 bits de menos peso apuntan a la posición.

LOS PUERTOS DE E/S Y RECURSOS ESPECIALES DEL PROCESADOR

Alguno de los pines de los puertos de E/S están multiplexados con una o unas funciones alternativas para la utilización de los periféricos internos. En general, cuando un periférico se habilita el pin correspondiente no puede utilizarse como pin de E/S.

Los microcontroladores PIC16F87X que están encapsulados en 28 pines, disponen de tres puertos (PORTA, B y C) mientras que los microcontroladores de 40 pines disponen de cinco puertos (PORTA, B, C, D y E).

PORTA y el registro TRISA

Este puerto dispone de 6 pines bidireccionales denominados RA0 - RA5 cuyo sentido está controlado por el registro TRISA, cuando se pone a 1 el bit correspondiente del registro TRISA, la línea correspondiente del PORTA se comporta como una entrada, mientras que si se pone a 0 se comporta como salida.

Al leer el registro del PORTA se está leyendo el estado de los pines, que es, el que se halla escrito en el biestable de datos, La función de escritura implica la operación de lectura, modificación y escritura. Es decir, le lee el pin, se modifica su valor y por último se escribe en el biestable el dato. Todos los pines del PORTA entregan niveles TTL.

El pin RA4 está multiplexado con el módulo de reloj TIMER0, por lo tanto el pin RA4/TOCKI actúa como E/S digital y como

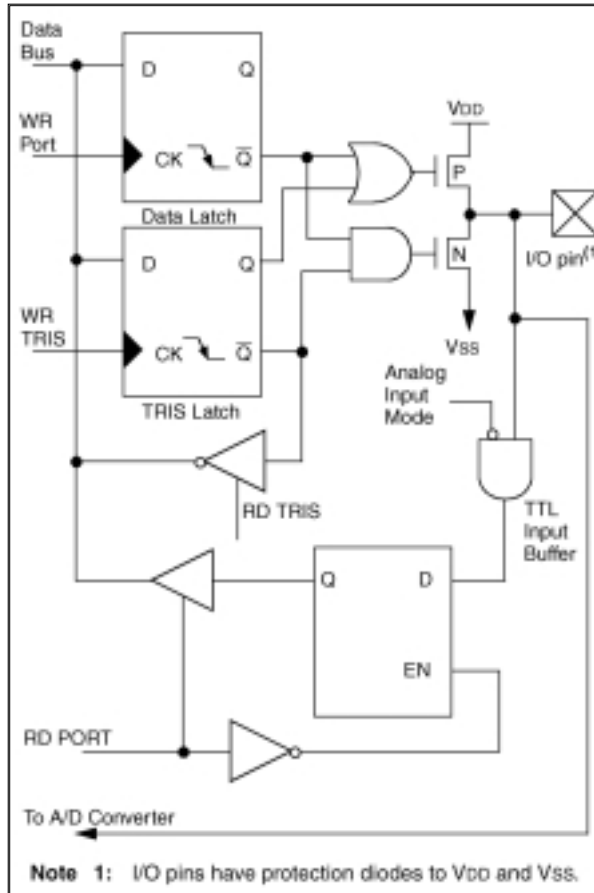


Figura 3.-Diagrama de conexión de los pines RA0-RA3 y RA5 que multiplexan la función de E/S digital con la del canal de entrada de una señal analógica para el Convertor A/D

entrada de reloj para el TIMER0.

Los pines RA0/AN0, RA1/AN1, RA2/AN2, además de E/S digital también pueden actuar como los canales 0, 1 y 2 del convertidor A/D.

El pin RA3/AN3/VREF+ tiene multiplexadas tres funciones: E/S digital, entrada del canal 3

del convertidor A/D y entrada de la tensión de referencia para los periféricos que la necesitan.

El pin RA5/AN4/SS# tiene multiplexada tres funciones: E/S digital, entrada del canal 4 del convertidor A/D y selección del modo esclavo cuando trabaja con la comunicación serie síncrona. Cuando se produce un Reset al

RELACIÓN DE REGISTROS RELACIONADOS CON EL PORTA

Dirección	Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Valor en POR,BOR	Valor en el resto de Reset
05 h	PORTA	--	--	RA5	RA4	RA3	RA2	RA1	RA0	--0x 0000	--0u 0000
85 h	TRISA	--	--	Registro de direccionamiento del PORTA						--11 1111	--11 1111
97 h	ADDCON1	ADFM	--	--	--	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000

conectar la tensión de alimentación (POR: Power-on Reset) todos los pines del PORTA quedan configurados como canales de entrada para el convertidor A/D y son leídas como 0.

Para seleccionar el modo de funcionamiento como E/S digitales o analógicas para el convertidor A/D hay que configurar el registro ADCON1 que se explicará con detalle más adelante. Si

se quiere configurar el PORTA como E/S digitales bastará con escribir en los 4 bit menos significativos del registro ADCON1 el valor 011x.

El registro TRISA que configura los pines deL PORTA, debe configurarse como entrada cuando se utilizan las entradas analógicas.

Ejemplo de inicialización del PORTA

```

bcf STATUS,RP0
bcf STATUS,RP1 ;Selecciona el Banco 0 de memoria
clrf PORTA ;inicializa el PORTB, borrando
;todas las basculas de datos

bsf STATUS,RP0 ;Selecciona el Banco 1 de memoria
movlw b'00001110' ;Configura el PORTA como
movwf ADCON1 ;E/S digitales
movlw b'00001111' ;Direcciona las líneas de- RA0-RA3
movwf TRISA ;como salida RA4-RAS como entrada
    
```

EL PORTB y el registro TRISB

El PortB es está formado por 8 líneas bidireccionales controladas por el registro de direccionamiento TRISB, poniendo un "1" en un bit del registro TRISB se direcciona la línea correspondiente como entrada, mientras que los bit que se ponen a "0" en el registro de direccionamiento TRISB, se configuran como salida.

Los PIC16F87X pueden ser programados en modo bajo voltaje o alto voltaje, en el primero de los casos se puede programar con VDD=5V. El modo alto voltaje es el modo que utiliza el PIC16F84, en este caso se introduce por el pin MCLR#Vpp una tensión entre 12 y 14V. En el modo de programación de bajo voltaje se pone el pin MCLR#Vpp 5V y por los pines RB3/PGM se pone a nivel alto. Como la programación en cualquiera de los casos se realiza sincronamente en serie, por el pin RB6/PGC se introducen los impulsos de reloj y por RB7/PGD los bit de datos serie.

Las líneas RB7-RB4 pueden programarse para generar una interrupción cuando una de ellas cambia de estado. Se deben configurarse como entradas y el valor que se introduce por ellas se compara con el anterior de tal forma que si no coinciden se genere una interrupción, siempre que esta esté autorizada. De todas forma el bit RBIF, que es el bit 0 del registro INTCON, se pondrá a 1.

El pin RB0/INT también puede programarse como petición de interrupción externa, si se autoriza con el correspondiente bit de permiso de ininterrupción, que corresponde con el bit (4) INTE del registro INTCON y con el bit INTEDG del registro OPTION_REG se programa el flanco de disparo.

Los pines del PORTB tienen una resistencia de pull-up a positivo de alimentación programable, esta se conecta cuando el bit 7 del OPTION REG, es decir RBPU# tiene un cero. La resistencia de pull-up realmente es un transistor CMOS tipo P, tal y

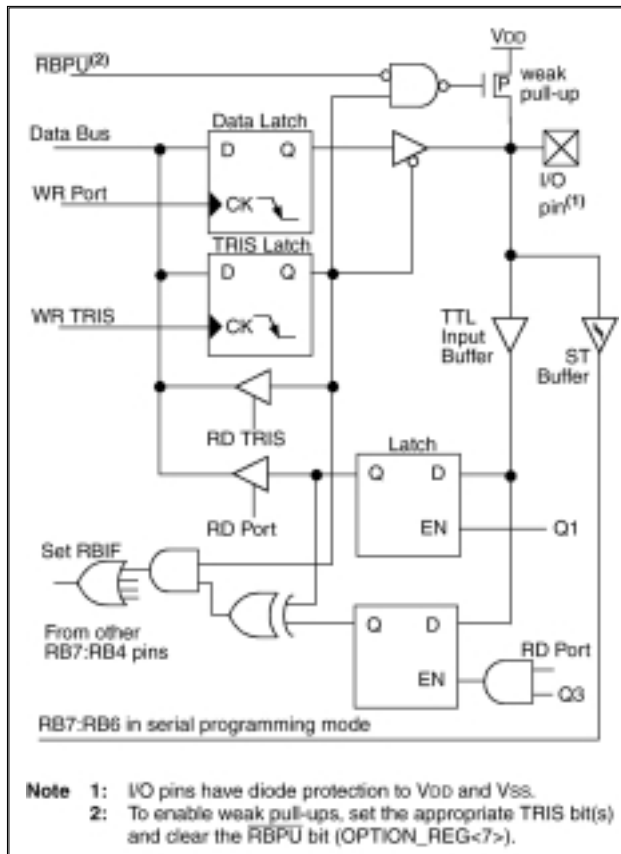


Figura 4.- Diagrama interno de las líneas RB7-RB4 del PORTB

RELACIÓN DE REGISTROS RELACIONADOS CON EL PORTB											
Dirección	Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Valor en POR,BOR	Valor en el resto de Reset
06 h 106 h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86 h 186 h	TRISB	Registro de direccionamiento del PORTB								1111 1111	1111 1111
81 h 181 h	OPTION_REG	RBPU#	INTEDG	TOSC	TOSC	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Leyenda
x = desconocido, u = inalterado. Las celdas sombreadas no son usadas por el PORTB

como puede verse en la Figura 4, que se conecta siempre que la línea está configurada como salida. Cuando se produce un Reset por conexión de alimentación (POP) se desconectan todas las resistencias de pull-up.

El PORTC y el Registro TRISC

El PORTC está formado por 8 líneas bidireccionales controladas por el registro de direccionamiento TRISC. Poniendo un 1 en un bit del registro TRISC se direcciona la línea correspondiente como entrada. Mientras que los bit que se ponen a 0 en el registro de direccionamiento TRISC, se configuran como salida.

Las líneas del PORTC están multiplexadas con varias funciones de los periféricos. En la Tabla 1, se muestra los pines del PORTC y las distintas funciones que tienen.

El PORTD y el registro TRISD

Este puerto solo existe en los microcontroladores de la familia 16F87X de 40 pines. El PORTD está formado por 8 líneas bidireccionales denominadas RD0/PSP0-RD7/PSP7 controladas por el registro de direccionamiento TRISD. Poniendo un 1 en un bit del registro TRISD se direcciona la línea correspondiente como entrada. Mientras que los bit que se ponen a 0 en el registro de direccionamiento TRISD, se configuran como salida. En este puerto todas las entradas disponen de un Trigger Schmitt.

Además los pines de E/S digitales de este puerto, pueden multiplexarse como una puerta paralela de 8 líneas (PSP), que sirve para permitir la comunicación en paralelo con otros elementos del sistema.

Para que trabaje como puerto de comunicación esclavo, hay que poner a 1 el bit PSPMODE, que se encuentra en el registro TRISE.

El PORTE y el registro TRISE

Este puerto solo existe en los microcontroladores de la familia 16F87X de 40 pines. Está formado por 3 pines multifunción R E 0 / R D # / A N 5 , R E 1 / W R # / A N 6 , y RE2/CS#/AN7, que se configuran individualmente como entrada o salida a través de los tres bit de menor peso del registro TRISE (TRISE <2:0>).

Las E/S del PORTE se convierten en entradas de control (RD#, WR# y CS#) para el puerto PSP cuando el bit PSPMODE que corresponde con el bit 4 del registro TRISE. Los pines de E/S digital del PORTD están multiplexa-

Nombre	bit	Función
RC0/T1OSO/TICK1	bit 0	E/S digital. Salida Timer 1. Entrada de impulsos para el Timer1
RC1/T1OSI/CCP2	bit 1	E/S digital, entrada al oscilador del Timer1. Entrada del módulo de Captura 2. Salida del Comparador 2; Salida del PWM2
RC2/CCP1	bit 2	E/S digital. Entrada Captura 1; Salida Comparador 1; Salida PWM 1
RC3/SCK/SCL	bit 3	E/S digital. Señal de reloj en modo SPI. Señal de reloj en modo I2C
RC4/SDI/SDA	bit 4	E/S digital. Entrada de datos en modo SPI. Línea de datos en modo I2C
RC5/SDO	bit 5	E/S digital. Salida de datos en modo SPI
RC6/TX/CK	bit 6	E/S digital. Línea de transmisión en USART. Señal de reloj síncrona en transmisión serie.
RC7/RX/DT	bit 7	E/S digital. Línea de transmisión en USART. Línea de datos en transmisión serie síncrona.

Tabla 1

RELACIÓN DE REGISTROS RELACIONADOS CON EL PORTC											
Dirección	Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Valor en POR,BOR	Valor en el resto de Reset
07 h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
87 h	TRISC	Registro de direccionamiento del PORTC								1111 1111	1111 1111

Leyenda: x = desconocido, u = inalterado

INFORMÁTICA
 61
 RESISTOR

62
RESISTOR
MICROCONTROLADORES

RELACIÓN DE REGISTROS RELACIONADOS CON EL PORTD											
Dirección	Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Valor en POR,BOR	Valor en el resto de Reset
08 h	PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	uuuu uuuu
88 h	TRISC	Registro de direccionamiento del PORTC								1111 1111	1111 1111
89 h	TRISE	IBF	OBF	IBOV	PSPMODE	--	Registro de direccionamiento del PORTE			0000 -111	0000 -111

Leyenda: x = desconocido, u = inalterado; Las celdas sombreadas no son usadas por el PORTD

RELACIÓN DE REGISTROS RELACIONADOS CON EL PORTE												
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0			Valor en POR,BOR	Valor en el resto de Reset	
09 h	PORTE	---	---	---	---	---	RE2	RE1	RE0	---- -xxx	---- -uuu	
89 h	TRISE	IBF	OBF	IBOV	PSPMODE	---	Registro de direccionamiento del PORTE					
97 h	ADCON1	ADFM	---	---	---	PCFG3	PCFG1	PCFG0		--0- 0000	--0- 0000	

Leyenda: x = desconocido, u = inalterado; - = no implementado se lee como '0'. Las celdas sombreadas no son usadas por el PORTE

das con las entradas digitales. Cuando se seleccionan las entradas analógicas los bit de control del TRISE deben ponerse a 0.

Nota.- Después de un Power-on-Reset, estos pines quedan configurados como entradas analógicas. Las funciones multiplexadas de los bit del PORTD son las siguientes:
Para aclarar el concepto del manejo de los puertos vamos a

Nombre	Bit	Función
RE0#/RD#/AN5	Bit 0	E/S digital. Señal de lectura en el modo de puerta paralelo esclava. Canal 5 del Conversor A/D
RE1#/WR#/AN6	Bit 1	E/S digital. Señal de escritura PSP. Canal 6 del Conversor A/D
RE2#/CS#/AN7	Bit 2	E/S digital. Señal de Chip Select en el modo PSP. Canal 7 del Conversor A/D

Descripción del registro TRISE							
R-0x	R-0	R/W-0	R/W-0	U-1	R/W-1	R/W-1	R/W-1
IBF	OBF	IBOV	PSPMODE	---	Bit2	Bit1	Bit0
Bit 7							Bit 0
bit 7:	IBF: Flag de buffer de entrada lleno 1 = Una palabra se ha recibido y esta esperando a ser leída por la CPU 0 = No se ha recibido ninguna palabra						
bit 6:	OBF: Flag de buffer salida lleno 1 = El buffer de salida todavía tiene la palabra previamente escrita 0 = El buffer de salida ha sido leído						
bit 5:	IBOV: La entrada de desbordamiento de buffer de entrada detecta bit (en modo microprocesador). 1 = Se ha producido una escritura antes de que la palabra de entrada se haya leído (debe ponerse a cero por software). 0 = No ha ocurrido desbordamiento						
bit 4:	PSPMODE: bit de selección de modo de Puerto Paralelo Esclavo 1 = Seleccionado el modo puerto paralelo esclavo 0 = Seleccionado el modo de E/S digital de propósito general						
bit 3:	:No implementado se lee como "0"						
bit 2:	:Bit2: bit de selección del modo de direccionamiento del pin RE2#/CS#/AN7 1 = Entrada 0 = Salida						
bit 1:	Bit1: bit de selección del modo de direccionamiento del pin RE1 /WR#/AN6 1 = Entrada 0 = Salida						
bit 0:	Bit0: bit de selección del modo de direccionamiento del pin RE0#/RD#/AN5 1 = Entrada 0 = Salida						

```

*****
Programa      Display.asm                      Fecha : 27- Enero-2003
Este programa lee el estado de cuatro interruptores conectados al PORTA <RA3:RA0>
y representa sobre un display conectado al PORTB, el valor hexadecimal correspondiente
Revisión 0.0                                  Programa para PIC16F87X
Velocidad del Reloj: 4 MHz                    Reloj Instrucción: 1 MHz = 1 uS
Perro Guardián :deshabilitado                Tipo de Reloj : XT
Protección del código : OFF
*****
List         p=16F876                          :Tipo de procesador
include     "P16F876.INC"                    :Definiciones de registros internos
;_config _CP_OFF & _DEBUG_OFF & _WRT_ENABLE_ON & _CPD_OFF & _LVP_OFF & _XT_OSC & _WDT_OFF
;Vector de Reset
org         0x00
;Inicio
org         0x05
INICIO      clrf PORTB                        :Borra los latch de salida
           bsf STATUS,RP0                    :Selecciona banco 1
           clrf TRISB                         :Puerta B se configura como salida
           movlw b'00000110'                 :Configura PORTA como E/S Digitales
           movwf ADCON1
           movlw b'00001111'                 :Configura RA3:RA0 como entradas
           movwf TRISA
           bcf STATUS,RP0                    :Selecciona banco 0

BUCLE      movf PORTA,W                       :Lee el código de RA0-RA3
           andlw b'00001111'                 :Convierte a 7 segmentos
           call Tabla                         :Visualiza sobre el display
           movwf PORTB
           goto BUCLE
*****
Tabla:     addwf PCL,F                        :Desplazamiento sobre la tabla
           retlw b'00111111'                 :Dígito 0
           retlw b'00000110'                 :Dígito 1
           retlw b'01011011'                 :Dígito 2
           retlw b'01001111'                 :Dígito 3
           retlw b'01100110'                 :Dígito 4
           retlw b'01101101'                 :Dígito 5
           retlw b'01111101'                 :Dígito 6
           retlw b'00000111'                 :Dígito 7
           retlw b'01111111'                 :Dígito 8
           retlw b'01100111'                 :Dígito 9
           retlw b'01110111'                 :Dígito A
           retlw b'01111100'                 :Dígito b
           retlw b'00111001'                 :Dígito C
           retlw b'01011110'                 :Dígito d
           retlw b'01111001'                 :Dígito E
           retlw b'01110001'                 :Dígito F
*****
ORG         0x1F00
bcf         PCLATH,4
bcf         PCLATH,3                          ;Selecciona la página 0
goto       INICIO
end
    
```

realizar un sencillo ejercicio que consiste en la lectura de unos interruptores conectados al PORTA y la representación de dicho valor binario en hexadecimal sobre un display de 7 segmentos conectado en el PORTB.

El circuito de aplicación es el que se muestra en la Figura 5. El programa está diseñado para aquellos que utilicen el bootloader con el circuito que presentamos en el número anterior o un programador con el IC-PROG,

tal y como está es válido para los primeros y si se quitan los "n", se puede cargarse directamente utilizando cualquiera de los demás programadores, estas líneas las hemos marcado en azul.

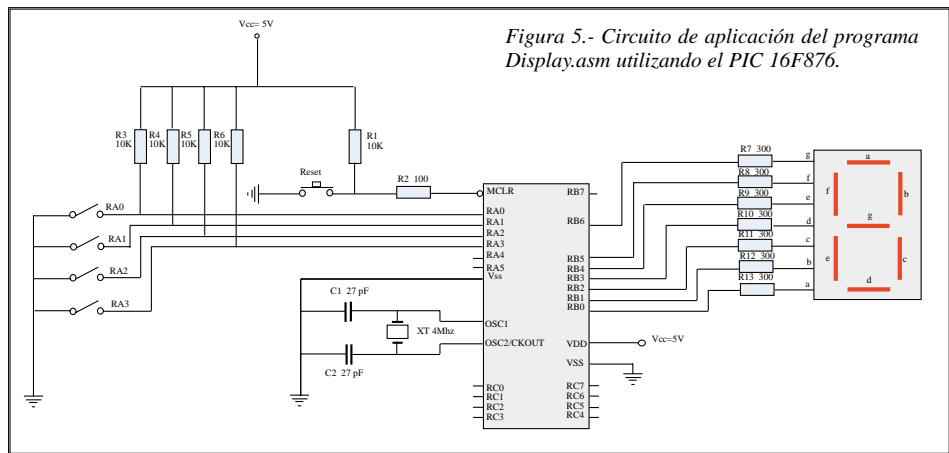


Figura 5.- Circuito de aplicación del programa Display.asm utilizando el PIC 16F876.