

Informe 7

PIC usando PORTB

Laboratorio de Arquitectura del Software

Elaborado por:

Tomás Guzmán, 21615008

1 MARCO TEÓRICO

Los PIC son circuitos integrados programables, bien sea utilizando C o su propio conjunto de instrucciones en ensamblador.

El hecho de permitir esta programación hace que los PIC puedan resolver infinidades de problemas a nivel de ingeniería, además de permitir crear códigos donde el límite es la creatividad.

Es una de las computadoras más pequeñas que tenemos disponibles, ya que consta de:

- Memoria, donde se guardan los datos (variables) y el programa en sí
- Unidad de procesamiento

Para este informe nos resultará de interés lo siguiente:

1.1 REGISTROS

- PORTB, o el puerto B, el cual es un registro de 8 bits el cual puede ser de entrada o de salida. Cada bit del mismo está asociado a sus pines RB0 al RB7 (8 pines claramente). Su dirección es 06H y es uno de los varios registros de función especial (SFR) que posee el PIC16F84A.
- PORTB es configurable mediante un registro ubicado en su misma dirección (06H) conocido como TRISB.
- El PIC permite acceder a TRISB o a PORTB según se establezca un banco de registros. Este microchip cuenta con dos bancos (BANK0 y BANK1) los cuales pueden activarse (como si de una palanca mecánica se refiriese) utilizando el registro especial STATUS.
- STATUS es otro registro de 8 bits el cual presta varias funciones como determinar si la última operación resultó en cero.
- Para fines de este informe nos interesa el bit 5. Señalado en su manual como STATUS<5>, su nombre exacto es Register Bank Select y su nombre es RP0. Si este

bit está activado se usarán los registros del BANK1, de otra forma se usarán los de BANK0.

- Note que PORTB y TRISB tienen la misma dirección, 06H. Si RP0 está activado, entonces se está en BANK1 y se podrá configurar TRISB. Si RP0, está desactivado se usará PORTB, dado que se está en BANK0.
- El PIC cuenta con un acumulador, un registro de uso general y temporal para realizar operaciones, conocido como W. Es de 8 bits como la mayoría de los registros del PIC.
- Se pueden cargar valores a W, sumarlos, restarlos, compararlos, moverlos a memoria, etc.

1.2 INSTRUCCIONES

- EQU se utiliza en asignaciones mnemotécnicas para los registros de uso especial.
- ORG se utiliza para interrupciones y ubicar al programa en memoria
- #DEFINE **no** es una instrucción, sino una directiva que se comporta como EQU, pero genera una funcionalidad parecida a un macro en programación.
- CLRF pone todos los bits de un registro en 0
- BSF o Bit Set f, recibe un registro y un bit. Como su nombre indica, hace que el bit en ese registro se active (sea 1).
- BCF es lo contrario, es Bit Clear f, recibe un registro y un bit y lo apaga (será 0).

2 IMPLEMENTACIÓN

```
LIST P=16F84

;=====
; DATA
;=====

; Registros de función especial
STATUS EQU 03H
TRISB EQU 06H
PORTB EQU 06H

; Definiciones mnemotécnicas
#define RP0 STATUS, 5
#define BANK1 BSF RP0
#define BANK0 BCF RP0

;=====
```

```

; TEXT
;=====

    ORG 0
INICIO
    BANK1          ; Cambiar al banco 1
    CLRF    TRISB   ; Configura a PORTB como salida
    BANK0          ; De vuelta al banco 0

    MOVLW    B'10101010' ; Se carga el literal 10101010 en W
    MOVWF    PORTB      ; Se mueve el valor de W a PORTB

END

```

Note como hacer uso de #DEFINE hace que el código se apegue lo más posible a sus definiciones más verbosas y asociadas al lenguaje humano presentadas en su manual.

Vamos por línea por línea explicando qué sucede en cada parte

- #DEFINE RP0 STATUS, 5, le indica al PIC que, de ahora en adelante STATUS, 5 (es decir, el bit 5 de STATUS) será llamado como RP0. Esto evita escribir STATUS, 5 en próximas instrucciones.
- #DEFINE BANK1 BSF RP0 le indica al PIC que de ahora en adelante BSF RP0 será ejecutado con solo escribir BANK1. Esta instrucción cambia al BANK1 de los registros. Note que sin el #DEFINE anterior esta instrucción sería BSF STATUS, 5
- Lo mismo sucede con la próxima línea.

Ahora en el segmento del programa, señalado por TEXT

- ORG 0 se utiliza para indicar en dónde comienza el programa en memoria.
- BANK1 ya se sabe qué hace, cambia al BANK1 de los registros de funciones especiales
- CLRF TRISB hace que todos los bits de TRISB sean 0, es decir TRISB = 00000000, lo cual hace que ahora todos los pines de PORTB sean una salida. Luego se usa BANK0 para activar este banco y poder utilizar a PORTB.
- Posteriormente se mueve la palabra 10101010 al acumulador W. Note que la cifra más significativa estará en RB7.
- Luego se mueve esta palabra a PORTB.
- Se finaliza el programa.

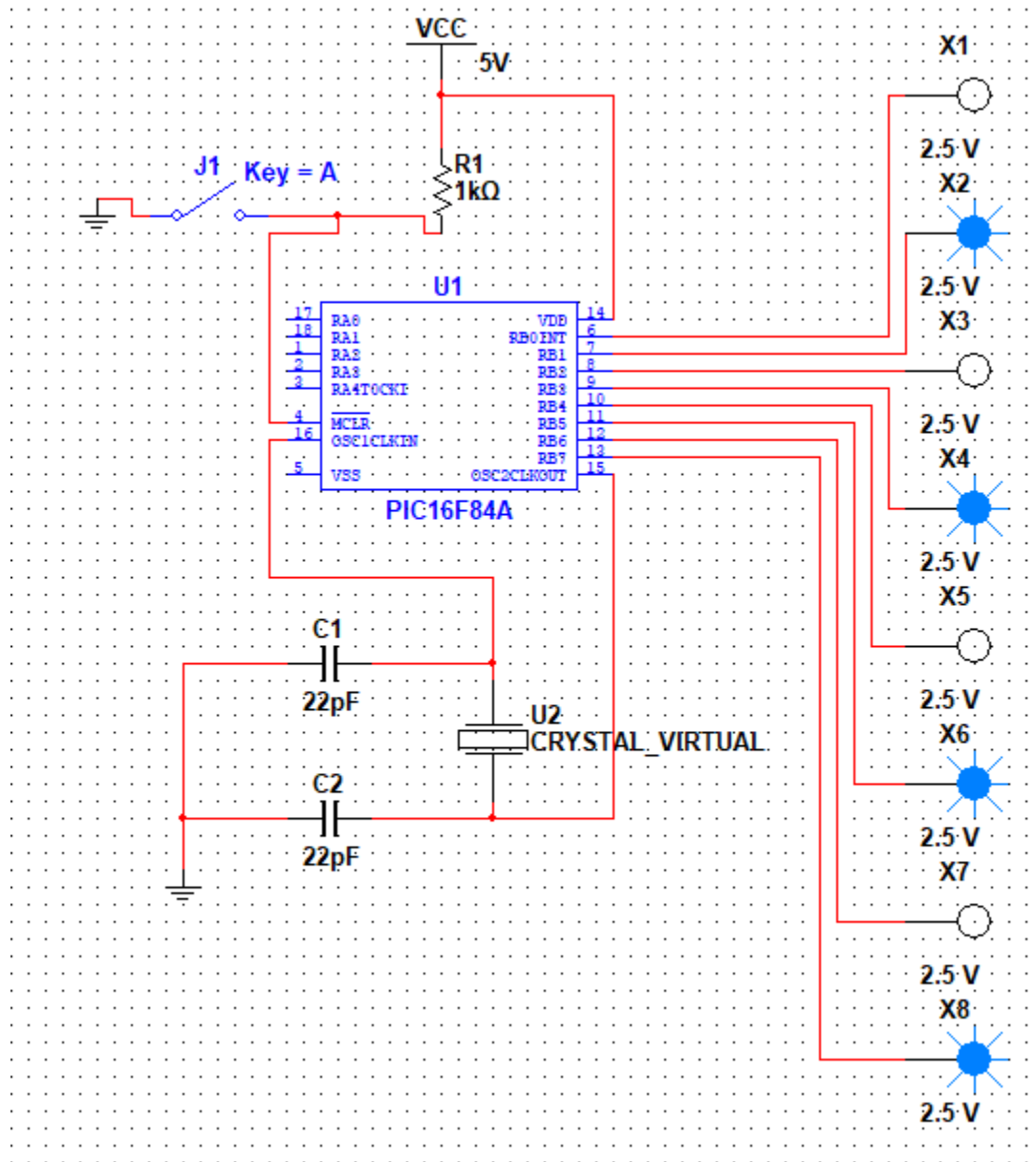
Note como el uso de escribir los macros BANK0 y BANK1 permite hacer el código realmente legible

BANK0	Mayor nivel de abstracción. Lectura sencilla
BCF RP0	Menor nivel. Si uno conoce bien la instrucción, sabe que se está cambiando a BANK0 dado que RP0 es el bit 5 de STATUS y encargado de esta operación
BCF STATUS, 5	Como la anterior, pero con más letras y sin señalar RP0
BCF 06H, 5	Mínimo nivel de abstracción. Produce código complicado de leer.

Este programa es pequeño y quizá escribir BCF STATUS, 5 no sea complicado por estos momentos, sin embargo, cuando los códigos llegan a cientos de líneas sí lo será.

Ahora para verificar el funcionamiento de este pequeño programa se necesita utilizar un simulador de circuitos el cual posea este PIC. Multisim es uno de estos.

3 SIMULACIÓN



Esta simulación realizada en Multisim, permite ver como el registro PORTB está trabajando como se espera. La secuencia luces azules representa al literal 10101010 que se cargó en W y luego de movió a PORTB, el cual fue configurado como puerto de salida. Note como RB7 está en azul.

4 CONCLUSIONES

- El PIC es un microchip programable el cual ofrece una cantidad de funciones útiles para hacer operaciones sobre circuitos
- Posee dos bancos de registros, BANK0 y BANK1
- Cada uno de estos bancos posee una cantidad de registros especiales destinados a distintas funciones y usos