



UNIVERSIDAD
NACIONAL AUTÓNOMA DE
MÉXICO



FACULTAD DE INGENIERÍA

Laboratorio de microcomputadoras

PRACTICA #3

“PUERTOS PARALELOS 2: CONTROL DE ACCIONES”

EQUIPO:

- BUSTOS RAMÍREZ LUIS ENRIQUE
- EGUIARTE MORETT LUIS ANDRÉS

SEMESTRE: 2017-2

Desarrollo.

Para cada uno de los siguientes ejercicios, realizar los programas solicitados y comprobar el funcionamiento de ellos.

1. Empleando dos puertos paralelos del microcontrolador PIC, uno de ellos configurado como entrada y el otro como salida; realizar un programa que de acuerdo al valor del bit menos significativo del puerto **A**, se genere la acción indicada en el puerto **B**.

Valor PA0	Acción puerto B
0	00000000
1	11111111

Código:

```
processor 16f877
include <p16f877.inc>
```

```
org 0
goto inicio
org 5
inicio:
    clrf PORTA
    clrf PORTB
    BSF STATUS, RP0
    BCF STATUS, RP1
    movlw 06h
    movwf ADCON1
    movlw B'00111111'
    movwf TRISA
    movlw B'00000000'
    movwf TRISB
    BCF STATUS, RP0
ciclo:
    btfss PORTA, 0
    goto apagar
    goto prender
prender:
    movlw H'FF'
    movwf TRISB
    goto ciclo
apagar:
    movlw H'00'
    movwf TRISB
    goto ciclo
end
```

Algoritmo:

```
Inicio
    Limpia PORTA
    Limpia PORTB
    Cambia a BANCO1
    Establecer como digital
```

```

TRISA ≈ B'00111111'
TRISB ≈ B'00000000'
Cambia a BANCO0
Ciclo:
    Si PORTA = 0:
        TRISB ≈ H'00'
    Sino:
        TRISB ≈ H'FF'
    Ir a CICLO

```

Fin

- Realizar un programa, el cuál realice las siguientes acciones de control, para lo cuál requiere trabajar un puerto de entrada y otro puerto de salida, usar los sugeridos en el ejercicio anterior; generar retardos de ½ seg., en las secuencias que lo requieran.

DATO	ACCION	Ejecución
\$00	Todos los leds apagados	00000000
\$01	Todos los leds encendidos	11111111
\$02	Corrimiento del bit más significativo hacia la derecha	10000000 01000000 00100000 00000001
\$03	Corrimiento del bit menos significativo hacia la izquierda	00000001 00000010 00000100 10000000
\$04	Corrimiento del bit más significativo hacia la derecha y a la izquierda	10000000 01000000 00000001 00000010 10000000
\$05	Apagar y encender todos los bits.	00000000 11111111

Código:

```

processor 16f877
include <p16f877.inc>

```

```

tempo equ h'40'

```

```

valor1 equ h'21'

```

valor2 equ h'22'

valor3 equ h'23'

cte1 equ 20h

cte2 equ 50h

cte3 equ 60h

org 0

goto inicio

org 5

inicio:

 clrf PORTA

 clrf PORTB

 BSF STATUS, RP0

 BCF STATUS, RP1

 movlw 06h

 movwf ADCON1

 movlw B'00111111'

 movwf TRISA

 movlw B'00000000'

 movwf TRISB

 BCF STATUS, RP0

ciclo:

 movf PORTA,W ; W = PORTA

 XORLW H'00'

 btfsc STATUS, Z

 goto op0

 movf PORTA,W ; W = PORTA

 XORLW H'01'

 btfsc STATUS, Z

```

goto op1
movf PORTA,W ; W = PORTA
XORLW H'02'
btfsc STATUS, Z
goto op2
movf PORTA,W ; W = PORTA
XORLW H'03'
btfsc STATUS, Z
goto op3
movf PORTA,W ; W = PORTA
XORLW H'04'
btfsc STATUS, Z
goto op4
movf PORTA,W ; W = PORTA
XORLW H'05'
btfsc STATUS, Z
goto op5
clrf PORTA
goto ciclo

```

op0:

```

movlw h'00'
movwf PORTB
goto ciclo

```

op1:

```

movlw h'FF'
movwf PORTB
goto ciclo

```

op2:

```

movlw h'00' ; W = 0x00
movwf PORTB ; no se prende ningun led

```

call retardo
goto izquierda

izquierda:

rlf W ; desplazamiento de Carry a la izquierda
btfsc W,7 ; (el carry no esta en el bit 7)
goto ciclo ; En caso contrario
movwf PORTB; PORTB = W -> se prende el led donde este el carry
call retardo
goto izquierda ; regresa al ciclo de desplazamiento a la izquierda

op3:

movlw h'80';
movwf PORTB; se prende el bit 7
call retardo
goto derecha

derecha:

rrf W; desplazamiento del carry a la derecha
btfsc W,0 ; (el carry no esta en el bit 0)
goto ciclo ; En caso contrario
movwf PORTB; se prende el led donde este el carry
call retardo
goto derecha ; regresa al ciclo derecha

op4:

movlw h'80'
movwf PORTB; se prende el bit 7
call retardo
goto sig

sig:

btfsc W,0 ; el carry no esta en el bit 0
goto sag ; en caso contrario
rrf W; desplazamiento a la derecha

movwf PORTB

call retardo

goto sig

sag:

btfsc W,7 ; el carry no esta en el bit 7

goto ciclo ; en caso contrario

rif W; desplazamiento a la izquierda

movwf PORTB

call retardo

goto sag

op5:

movlw h'00'

movwf PORTB

call retardo

movlw h'FF'

movwf PORTB

call retardo

goto ciclo

retardo:

movlw cte1

movwf valor1

tres movlw cte2

 movwf valor2

dos movlw cte3

 movwf valor3

uno decfsz valor3

 goto uno

 decfsz valor2

 goto dos

```

        decfsz valor1
        goto tres
        return
end

```

Algoritmo:

Inicio

```

valor1 ≈ h'21'
valor2 ≈ h'22'
valor3 ≈ h'23'
cte3 ≈ 60h
cte2 ≈ 50h
cte1 ≈ 20h
LIMPIA PORTA
LIMPIA PORTB
Cambia a BANCO1
ADCON1 ≈ 06H ;configurar como digital
TRISA ≈ B'00111111'
TRISB ≈ B'00000000'
Cambia a BANCO0
ciclo:
    caso(PORTA):
        PORTA=H'00':
            Ir a op0
        PORTA=H'01':
            Ir a op1
        PORTA=H'02':
            Ir a op2
        PORTA=H'03':
            Ir a op3
        PORTA=H'04':
            Ir a op4
        PORTA=H'05':
            Ir a op5
    LIMPIA PORTA
    Ir a ciclo
op0:
    PORTB ≈ H'00'
    Ir a ciclo
op1:
    PORTB ≈ H'FF'
    Ir a ciclo
op2:
    PORTB ≈ H'0'
    Llama RETARDO

```



```

Ir a izquierda
izquierda:
    Rbi PORTB
    Si PORTB(7)= 1:
        Ir a ciclo
        Llama RETARDO
    Ir a Izquierda

```

```

op3:
    PORTB ≈ H'80'
    Llama RETARDO
    Ir a Derecha
    Derecha:
        Rbd PORTB
        Si PORTB(0)= 1:
            Ir a ciclo
            Llama RETARDO
        Ir a Derecha

```

```

op4:
    PORTB ≈ H'80'
    Llama Retardo
    Ir a sig
    Sig:
        Si PORTB(0)=1:
            Ir a sag
            Rbd PORTB
            Llama RETARDO
            Ir a Sig
    Sag:
        Si PORTB(7)=1:
            Ir a ciclo
            Rbi PORTB
            Llama RETARDO
            Ir a Sag

```

```

RETARDO:
    valor1 ≈ cte1
    TRES: valor2 ≈ cte2
    DOS: valor3 ≈ cte3
    UNO: valor3- ≈ 1
        Si valor3!=0:
            IR A UNO
        valor2- ≈ 1
        Si valor2!=0:
            IR A DOS
        valor1- ≈ 1
        Si valor1!=0:
            IR A TRES
    REGRESA

```

Fin

Conclusiones.

Bustos Ramírez Luis Enrique:

Eguiarte Morett Luis Andrés: Con esta práctica me fue posible familiarizarme más a fondo con los puertos paralelos del PIC16F877, con su uso como entradas y como adaptar las respuestas del PIC basados en estas entradas por medio de otro puerto paralelo configurado como salida.