



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
**FACULTAD DE INGENIERÍA**



**LABORATORIO DE MICROCOMPUTADORAS**

**PRÁCTICA 5:**

**PUERTOS PARALELOS IV(LECTURA DE SENSORES ÓPTICOS)**

**GRUPO: 12**

**PROFESORA: M.I LOURDES ANGÉLICA QUIÑONES JUAREZ**

**ALUMNO: CHÁVEZ DELGADO JORGE LUIS**

**Nº DE CUENTA: 312217493**

**FECHA DE ASIGNACIÓN:**

*10/03/17*

**FECHA DE ENTREGA:**

*14/03/17*

## PRÁCTICA 5: Puertos Paralelos IV (Lectura de sensores ópticos)

**OBJETIVO:** Emplear los puertos paralelos, para hacer lecturas de señales externas (sensores reflectivos) con el microcontrolador y realizar operaciones de acuerdo a los valores recibidos.

**Ejercicio 1:** En este ejercicio, realizamos un reflejo con ayuda de la instrucción “*andwf*”, es decir que lo que entra en el puerto A(en este caso los sensores infrarrojos) se refleja en la salida puerto B (en este caso los led's). Utilizamos la máscara para poder realizar este reflejo y obtener el resultado de la siguiente tabla:

Entradas			Salidas			
Sensor Izquierdo PA2	Sensor Central PA1	Sensor Derecho PA0	PB3	PB2	PB1	PB0
N	N	N	0	0	0	0
N	N	B	0	0	0	1
N	B	N	0	0	1	0
N	B	B	0	0	1	1
B	N	N	0	1	0	0
B	N	B	0	1	0	1
B	B	N	0	1	1	0
B	B	B	0	1	1	1

Donde N equivale a un 0 y B equivale a un 1.

Lo que vimos en la placa fue que cuando cubríamos un sensor se iluminaba el igual pero en los led's, es decir si cubríamos el sensor 1 se prendía el led 1.

## **Código:**

```
;Programa 1
    processor 16f877
    include <pl6f877.inc>
inicio org 0                ;Carga al vector de RESET la dirección de
    goto inicio
usuario org 05              ;Dirección de inicio del programa del
inicio:
    bsf STATUS,RP0 ; Nos cambiamos de banco
    bcf STATUS,RP1
    movlw H'07'
    movwf ADCON1    ;Configuramos el registro como
entrada/salida
    movlw H'FF' ;Movemos un 255 a w
    movwf TRISA ;Movemos w al registro TRISA
    movlw H'00' ;Movemos un 0 a w
    movwf TRISB ;Movemos w al registro TRISB
    bcf STATUS,RP0 ;Regresamos al banco 0
AND:
    movlw H'07' ;Movemos un 7 a w
    andwf PORTA,w ;Aplicamos un and a lo que hay en el puerto
B
    movwf PORTB ;Mostramos el reflejo de lo que hay
en w en el puerto B
    goto AND ;Regresamos a la subrutina AND
end
```

## Ejercicio 2:

En este ejercicio también utilizamos los sensores, pero nuestras salidas en esta ocasión fueron los motores, debido a una falla de la tarjeta ya no pudimos visualizar el comportamiento de los sensores, pero estos resultados se pueden ver en el código, de acuerdo con la siguiente tabla:

ENTRADAS			ACCIÓN	
Sensor Izquierdo	Sensor Central	Sensor Derecho	MOTOR IZQUIERDO	MOTOR DERECHO
B	N	N	ATRÁS	ADELANTE
N	B	N	ADELANTE	ADELANTE
N	N	B	ADELANTE	ATRÁS
N	N	N	PARO	PARO

Utilizando la tabla de la práctica anterior para recordar como se comportaban los motores.

PORTA	PORTB			
	EN1	DIR1	EN2	DIR2
0100	1	0	1	1
1111	1	1	1	1
0001	1	1	1	0
0000	0	0	0	0

Donde si:

EN1=1 ; Se enciende el motor 1

DIR1=1 ; Gira el motor en sentido horario

EN2=0 ; Se apaga el motor 2

DIR2=0 ; Gira el motor en sentido antihorario

Pudimos haber realizado el código sin necesidad de utilizar la instrucción **andwf** sin embargo, la implementamos ya que fue la instrucción utilizada durante el primer ejercicio de la práctica.

### Código:

;Programa 2

processor 16f877

include <pl6f877.inc>

A equ H'24'

org 0 ;Carga al vector de RESET la dirección de inicio

goto inicio

org 05 ;Dirección de inicio del programa del usuario

inicio:

clrf PORTA ;Limpiamos lo que hay en el puerto A

bsf STATUS,RP0 ; Nos cambiamos de banco

bcf STATUS,RP1

movlw H'07'

movwf ADCON1 ;Configuramos el registro como entrada/salida

movlw H'FF' ;Movemos un 255 a w

movwf TRISA ;Movemos w al registro TRISA

movlw H'00' ;Movemos un 0 a w

movwf TRISB ;Movemos w al registro TRISB

bcf STATUS,RP0 ;Regresamos al banco 0

INFRA: ;Le asignamos el nombre INFRA a la subrutina

movf PORTA,0 ; Leemos lo que hay en puerto A

movwf A ;Lo que contiene w lo movemos a A

movwf H'07';Movemos un 7 a w

andwf A,f ; Realizamos la operación lógica and entre w y A

movfw A ; Movemos el resultado anterior a A

xorlw H'00' ; Realizamos una operación lógica xor entre la literal 0 y w

btfsc STATUS,Z ;Preguntamos por la bandera, salta si es cero

```
call paso1 ;Hacemos una llamada a paso1
```

```
movf PORTA,0 ; Leemos lo que hay en puerto A
```

```
xorlw H'01' ;Realizamos operación xor entre w y 1
```

```
btfsc STATUS,Z ;Preguntamos por la bandera, salta si es cero
```

```
call paso2 ;Hacemos una llamada a paso2
```

```
movf PORTA,0 ; Leemos lo que hay en puerto A
```

```
xorlw H'02' ;Realizamos la operación xor entre w y 2
```

```
btfsc STATUS,Z ;Preguntamos por la bandera, salta si es cero
```

```
call paso3 ;Hacemos una llamada a paso3
```

```
movf PORTA,0 ; Leemos lo que hay en puerto A
```

```
xorlw H'04' ;Realizamos la operación entre w y 4
```

```
btfsc STATUS,Z ;Preguntamos por la bandera, salta si es cero
```

```
call paso4 ;Hacemos una llamada a paso4
```

```
goto inicio ;Regresamos a inicio
```

```
paso1:
```

```
    movlw      b'00001011'; El motor 1 gira hacia atrás y el 2 hacia adelante
```

```
    movwf      PORTB      ; Movemos el resultado anterior al puerto B que son los motores
```

```
    return
```

```
paso2:
```

```
    movlw      b'11111111' ;Ambos motores giran hacia adelante
```

```
    movwf      PORTB      ; Movemos el resultado anterior al puerto B que son los motores
```

```
    return
```

```
paso3:
```

```
    movlw      b'00001110' ;El motor 1 gira hacia adelante y el segundo hacia atrás
```

```

        movwf      PORTB      ; Movemos el resultado anterior al puerto B que
son los motores
        return

paso4:

        movlw      b'00000000' ;Ambos motores se encuentran en paro cuando
no se cubre ningún sensor

        movwf      PORTB      ; Movemos el resultado anterior al puerto B que
son los motores

        return

        end      ;fin del programa

```

## Conclusiones:

Podemos concluir que el objetivo de la práctica se cumplió ya que pudimos realizar el ejercicio 1 con éxito y visualizar su ejecución en la tarjeta. Vimos que los sensores son muy sensibles ya que con un leve acercamiento se activaba; y aunque no pudimos ver la ejecución del segundo código, supimos como funcionaba internamente en el código. También cabe destacar que aprendimos el funcionamiento de la máscara para poder realizar el reflejo de un puerto a otro con la ayuda de la instrucción **andwf**, por otro lado también pudimos reforzar el uso de la instrucción **xorlw** para configurar el encendido y dirección de los motores aplicando nuevamente el uso de las tablas dandonos cuenta nuevamente que es de gran ayuda primero realizar la tabla antes de proceder a programar.