**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**

**LABORATORIO DE MICROCOMPUTADORAS**

PRÁCTICA 4:

**PUERTOS PARALELOS III (CONTROL DE MOTORES DE CD)**

**GRUPO:** 12

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

**ALUMNO:** CHÁVEZ DELGADO JORGE LUIS

**N° DE CUENTA:** 312217493

**FECHA DE ASIGNACIÓN: FECHA DE ENTREGA:**

*03/03/17* 07*/03/17*

PRÁCTICA 4: PUERTOS PARALELOS III(CONTROL DE MOTORES DE CD)

OBJETIVO: Emplear los puertos paralelos que contiene un microcontrolador, para controlar la operación de dos motores de corriente directa.

**Ejercicio 1:**

En este ejercicio tuvimos complicaciones debido a la tarjeta, sin embargo, pudimos lograr realizar lo que pedía el inciso 1. Primeramente construimos nuestra tabla para saber que es lo que iba a realizar cada selección del dipswitch, quedó de la siguiente manera:

|  |  |  |
| --- | --- | --- |
| **PORTA** | **PORTB** | |
|  | |  |  | | --- | --- | | **EN1** | **DIR1** | | |  |  | | --- | --- | | **EN2** | **DIR2** | |
| 00000000 | |  |  | | --- | --- | | 0 | 0 | | |  |  | | --- | --- | | 0 | 0 | |
| 00000010 | |  |  | | --- | --- | | 0 | 0 | | |  |  | | --- | --- | | 1 | 1 | |
| 00000100 | |  |  | | --- | --- | | 0 | 0 | | |  |  | | --- | --- | | 1 | 0 | |
| 00001000 | |  |  | | --- | --- | | 1 | 1 | | |  |  | | --- | --- | | 0 | 0 | |
| 00010000 | |  |  | | --- | --- | | 1 | 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

Proseguimos con la programación para comprobar si nuestra tabla estaba bien, en este ejercicio aplicamos la salida solo a un motor a la vez.

**Código:**

;Programa 1

processor 16f877

include <p16f877.inc>

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

goto inicio

org 05 | ;Dirección de inicio del programa del usuario

inicio:

; Configuraciones para tratar puertos DIGITALES

clrf PORTA ; Limpia el registro PORTA

bsf STATUS,RP0 ; Cambiando al banco 1

bcf STATUS,RP1

movlw H'07' ; Configura puertos A y E como digitales

movwf ADCON1

movlw H'FF' ; Configura el puerto A como entrada

movwf TRISA

movlw H'00' ; Configura el puerto B como salida

movwf TRISB

bcf STATUS,RP0 ; Regresa al banco cero

PUERTOA:

movf PORTA,0 ;leer lo que hay en el puerto A y cargarlo en w

xorlw H'00' ;Aplicamos la operación lógia xor a la literal y lo guardamos en w, se modifica la bandera Z

btfsc STATUS,Z ; Si Z es sero salta

call paso1 ; Llamamos al paso1

movf PORTA,0 ; Se realiza lo mismo para los pasos siguientes pero se cambia el valor de la literal

xorlw H'02'

btfsc STATUS,Z

call paso2

movf PORTA,0

xorlw H'04'

btfsc STATUS,Z

call paso3

movf PORTA,0

xorlw H'08'

btfsc STATUS,Z

call paso4

movf PORTA,0

xorlw H'10'

btfsc STATUS,Z

call paso5

goto PUERTOA

paso1:

movlw H'00' ;Los motores estan en paro 00000000

movwf PORTB ;Se mueve al puerto B la acción

return

paso2:

movlw H'03' ;El motor dos esta habilitado y gira en sentido horario

movwf PORTB ;Se mueve al puerto B la acción

return

paso3:

movlw H'02' ;El motor dos esta habilitado y gira en sentido antihorario

movwf PORTB ;Se mueve al puerto B la acción

return

paso4:

movlw H'0C' ;El motor 1 esta habilitado y gira en sentido horario

movwf PORTB ;Se mueve al puerto B la acción

return

paso5:

movlw b'00001000' ;El motor 1 esta habilitado y gira en sentido antihorario

movwf PORTB ;Se mueve al puerto B la acción

return

end

**Ejercicio 2:**

Del ejercicio anterior nos dimos cuenta de la importancia de analizar las opciones primero y no comenzar a programar sin saber a lo que vamos a llegar. Por lo que construimos esta segunda tabla:

|  |  |  |
| --- | --- | --- |
| **PORTA** | **PORTB** | |
|  | |  |  | | --- | --- | | **EN1** | **DIR1** | | |  |  | | --- | --- | | **EN2** | **DIR2** | |
| 00000000 | |  |  | | --- | --- | | 0 | 0 | | |  |  | | --- | --- | | 0 | 0 | |
| 00000001 | |  |  | | --- | --- | | 1 | 1 | | |  |  | | --- | --- | | 1 | 1 | |
| 00000010 | |  |  | | --- | --- | | 1 | 0 | | |  |  | | --- | --- | | 1 | 0 | |
| 00000011 | |  |  | | --- | --- | | 1 | 1 | | |  |  | | --- | --- | | 1 | 0 | |
| 00000100 | |  |  | | --- | --- | | 1 | 0 | | |  |  | | --- | --- | | 1 | 1 | |

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

Esta tabla nos ayudo con la programación de los motores, en este ejercicicio activamos ambos motores con sus giros posibles, incluyendo el paro.

**Código:**

**;Programa 1**

**processor 16f877**

**include <p16f877.inc>**

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

**goto inicio**

**org 05 ;Dirección de inicio del programa del usuario**

**inicio:**

**; Configuraciones para tratar puertos DIGITALES**

**clrf PORTA ; Limpia el registro PORTA**

**bsf STATUS,RP0 ; Cambiando al banco 1**

**bcf STATUS,RP1**

**movlw H'07' ; Configura puertos A y E como digitales**

**movwf ADCON1**

**movlw H'FF' ; Configura el puerto A como entrada**

**movwf TRISA**

**movlw H'00' ; Configura el puerto B como salida**

**movwf TRISB**

**bcf STATUS,RP0 ; Regresa al banco cero**

**PUERTOA: ;Configuramos las opciones para el dipswitch en bits**

**movf PORTA,0 ;leer lo que hay en el puerto A y cargarlo en w**

**xorlw b'00000000' ;Aplicamos una operación lógica a la literal y verificamos la bandera Z**

**btfsc STATUS,Z ;Si Z es cero, salta.**

**call paso1 ;Llamamos al paso 1**

**movf PORTA,0 ; Realizamos lo mismo para los pasos siguientes.**

**xorlw b'00000001'**

**btfsc STATUS,Z**

**call paso2**

**movf PORTA,0**

**xorlw b'00000010'**

**btfsc STATUS,Z**

**call paso3**

**movf PORTA,0**

**xorlw b'00000011'**

**btfsc STATUS,Z**

**call paso4**

**movf PORTA,0**

**xorlw b'00000100'**

**btfsc STATUS,Z**

**call paso5**

**goto PUERTOA**

**paso1:**

**movlw b'00000000' ;Ambos motores estan en paro**

**movwf PORTB ; Se mueve al puerto B la acción**

**return**

**paso2:**

**movlw b'11111111' ; Ambos motores estan habilitados y girando en sentido horario**

**movwf PORTB ; Se mueve al puerto B la acción**

**return**

**paso3:**

**movlw b'00001010' ;Ambos motores estan habilitados y girando en sentido antihorario**

**movwf PORTB ; Se mueve al puerto B la acción**

**return**

**paso4:**

**movlw b'00001110' ;Ambos motores estan habilitados pero el motor 1 gira en sentido horario y el 2 al revés**

**movwf PORTB ; Se mueve al puerto B la acción**

**return**

**paso5:**

**movlw b'00001011' ;Ambos motores estan habilitados pero el motor 2 gira en sentido horario y el 1 al revés.**

**movwf PORTB ; Se mueve al puerto B la acción**

**return**

**end**

**Conclusiones:**

Principalmtente podemos concluir que el objetivo de la práctica se cumplió, aunque hubo un poco de dificultad en cuanto al material, en este caso la tarjeta. No impidió que pudieramos observar el comportamiento de los ejercicios propuestos.

Durante la ejecución de esta práctica, nos dimos cuenta que es necesario alimentar tanto a la tarjeta como a los dos motores para que puedan funcionar los programas, además que el uso de la instrucción ***xorlw*** nos facilita el uso del puerto A como configurar el dipswitch para que en cada combinación realizará una acción reflejada en el puerto B, sin embargo no podemos olvidar que también es necesario saber hacerlo bit por bit.