**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**FIEC**

**Laboratorio de Microcontroladores**

**PROYECTO 1 PARCIAL**

**“Comparador de 2 números”**

**Alumno:**

**Jairo Rayo Cedillo**

**Paralelo: # 5**

**Grupo: # 5**

**Fecha de presentación:**

**11 - Julio - 2011**

**2011 – I TÉRMINO**

**1.- Enunciado del proyecto**

Este programa ingresa 2 números por switches, internamente los compara y luego los muestra en dos displays de 7 segmentos cual es el mayor en un display y el menor lo muestra en el otro display.

**2.-Diagrama de Bloques**

DISPLAYS

SWITCHES

PIC 16F887

**3.- Diagrama de Flujo funcional del Programa principal**

En el PORTA va la mayor y PORTB menor

No

Si

Carga

Ingreso de la variable INT2

Carga

Ingreso de la variable INT1

Inicialización de Variables y Registros

Inicio

Compara las dos variables INT1 INT2

**4.- Diagrama de Flujo funcional de las Subrutinas**

Se copia el valor de W en INT2

Subrutina

Se copia el valor de W en INT1

Subrutina

Copia PORTC en W

Copia PORTC en W

**5.- Descripción del algoritmo o estrategia utilizada.**

1. Inicialización de las variables
2. Carga del valor INT1
3. Carga del valor INT2
4. Resta del valores INT2 – INT2
5. Comparación de los valores a través del acarreo C del registro STATUS
6. Si C = 0 entonces INT2 es mayor que INT1
7. Se copia INT2 en PORTA e INT1 en PORTD
8. Si C = 1 entonces INT1 es mayor que INT2
9. Se copia INT1 en PORTA e INT1 en PORTD
10. Fin del Programa hasta el RESET

**6.- Listado del programa fuente en lenguaje ensamblador con comentarios en las líneas de código que considere fundamentales.**

 LIST p=16F887 ;Tipo de microcontrolador

 INCLUDE P16F887.INC ;Define los SFRs y bits del

 ;P16F887

 \_\_CONFIG \_CONFIG1, \_CP\_OFF&\_WDT\_OFF&\_XT\_OSC

 ;Setea parámetros de

 ;configuración

 errorlevel -302 ;Deshabilita mensajes de error

CBLOCK 0X20

AUXA,AUXB,AUXC,INT1,INT2

ENDC

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ORG 0X00

GOTO PRINCIPAL

;Configuración de registros del PIC16F887

CONFIG\_PIC

;BANCO 4

BANKSEL ANSEL

CLRF ANSEL

CLRF ANSELH

;BANCO 1

BANKSEL TRISA

MOVLW B'00000000'

MOVWF TRISA

MOVLW B'00000011'

MOVWF TRISB

MOVLW B'11111111'

MOVWF TRISC

MOVLW B'00000000'

MOVWF TRISD

MOVLW B'01101001'

MOVWF OSCCON

MOVLW .0

MOVWF INT1

MOVLW .0

MOVWF INT2

;BANCO 0

BANKSEL PORTA

CLRF PORTC

CLRF PORTB

CLRF PORTD

CLRF PORTA

RETURN

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;Rutina de retardo para visualización

RETARDO

MOVLW .1

MOVWF AUXA

RETAR3 MOVLW .2

 MOVWF AUXB

RETAR2 MOVLW .100

 MOVWF AUXC

RETAR1 DECFSZ AUXC,F

 GOTO RETAR1

 DECFSZ AUXB,F

 GOTO RETAR2

 DECFSZ AUXA,F

 GOTO RETAR3

 RETURN

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;Tabla para activación de segmentos en display de cátodo común

TABLA\_DISPLAY\_CC

 ADDWF PCL,F

 RETLW 0x3F ; Retorna con el código del 0

 RETLW 0x06 ; Retorna con el código del 1

 RETLW 0x5B ; Retorna con el código del 2

 RETLW 0x4F ; Retorna con el código del 3

 RETLW 0x66 ; Retorna con el código del 4

 RETLW 0x6D ; Retorna con el código del 5

 RETLW 0x7D ; Retorna con el código del 6

 RETLW 0x07 ; Retorna con el código del 7

 RETLW 0x7F ; Retorna con el código del 8

 RETLW 0x67 ; Retorna con el código del 9

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;Rutina para visualización de valor en display

VISUAL\_CC

MOVF INT1,W

CALL TABLA\_DISPLAY\_CC

MOVWF PORTA

CALL RETARDO

MOVF INT2,W

CALL TABLA\_DISPLAY\_CC

MOVWF PORTD

CALL RETARDO

RETURN

VISUAL\_C1

MOVF INT2,W

CALL TABLA\_DISPLAY\_CC

MOVWF PORTA

CALL RETARDO

MOVF INT1,W

CALL TABLA\_DISPLAY\_CC

MOVWF PORTD

CALL RETARDO

RETURN

;Copia el valor temp1

COPIAR\_1

MOVF PORTC,W

MOVWF INT1

RETURN

;Copia el valor al temp2

COPIAR\_2

MOVF PORTC,W

MOVWF INT2

RETURN

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;Programa principal

PRINCIPAL CALL CONFIG\_PIC

LOOP1 BTFSS PORTB,0

 GOTO LOOP1

 CALL COPIAR\_1

LOOP2 BTFSC PORTB,0

 GOTO LOOP2

LOOP3 BTFSS PORTB,0

 GOTO LOOP3

 CALL COPIAR\_2

LOOP4 BTFSC PORTB,0

 GOTO LOOP4

BUCLE MOVF INT1,W

 SUBWF INT2,W

 BTFSS STATUS,0

 CALL VISUAL\_CC

 BTFSC STATUS,0

 CALL VISUAL\_C1

 GOTO BUCLE

 END

**7.- Copia impresa del circuito armado en PROTEUS para la simulación en el momento de su ejecución.**

****



**8.- Conclusiones:**

-Se entendió y comprendió de una forma eficaz el uso del comando o directiva CBLOCK en el programa Mplab la cual nos permite seleccionar un grupo de variables las cuales se direccionaran en la memoria RAM a partir de la dirección asignada a la primera variable, la cual es la variable de referencia para todas las demás asignadas al bloque CBLOCK.

-Se logró la familiarización e interacción de 2 programas muy útiles en la vida profesional de cualquier ingeniero eléctrico como son el Mplab y Proteus, logrando de esta forma poder cargar nuestro código fuente programable realizada en Mplab en el simulador Proteus exactamente en el Isis y cargarlo en el Pic 16F887 el cual respondió de acuerdo a lo que nosotros programamos.

-Se aprendieron 2 herramientas muy útiles del Mplab como son el BREAKPOINT y el STOPWACH. El breakpoint es una herramienta la cual consiste en hacer un punto en el momento de la ejecución en cualquiera de las líneas del programa, en cambio el stopwach mide el tiempo desde el inicio de la ejecución hasta cuando se tarde en llegar al breakpoint.

-.Gracias a los displays digitales pudimos mostrar el resultado de la comparación de los dos números, en mi caso utilice solamente números BCD es decir del 0 al 9, en la parte izquierda mostré el número mayor y en la parte derecha mostré el número menor. Otra forma de haber podido mostrar los dos números hubiera sido mediante un arreglo de leds, y mostrar el encendido y apagado de los leds de forma binaria, es decir si el led esta encendido significa que es un 1 o apagado es un 0.

**9.- Recomendaciones:**

-Se recomienda utilizar la herramienta BREAKPOINT la cual nos permite realizar un seguimiento línea a línea el código del programa ya que de esta manera es más visible encontrar alguna falla en el funcionamiento del programa.

-No debemos de regirnos en cuanto a los simuladores utilizados ya que estos no funcionan del todo bien, y muchas veces puede que lo que observamos en el simulador no de el mismo resultado que si lo observaramos en la vida real es decir en el PCB.

-Se recomienda tener a la mano los datasheets de los elementos a utilizar en la práctica, en este caso se hizo uso del datasheet del Pic 16F887 ya que asi podemos comprender fácilmente la estructura interna del Pic y las funciones que posee.

**-ANEXOS**

-**Visualización en 3D del PCB:**

****