**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**FIEC**

**Laboratorio de Microcontroladores**

**Proyecto 5**

**Ejercicio de potenciómetro y juego 21**

**Nombre del estudiante**

**Néstor Naula**

**Paralelo: # 9**

**Fecha de presentación:**

**22-08-2012**

**2012 – 1° TÉRMINO**

**1.- Enunciado del proyecto**

Vamos a mostrar dos juegos con el PIC 16F887, el cual se hizo utilización de la pantalla LCD ya que implementarlo en MicroC pro con ayuda de proteus es fácil y se comprende de mejor manera, se hizo un pequeño menú el cual, con la utilización del teclado 4x4 elegía la opción que deseaba el usuario ya sea ir al ejercicio del potenciómetro, o ir al juego opcional que era 21. Con esto saber usar sus diferentes comandos para poder mostrar mensajes en pantalla, como letras o números, ya después de haber elegido la opción la pantalla LCD se borraba y se insertaba la nueva información.

Si se elige la opción 1 se iba al juego del potenciómetro el cual, se puso la patita Re1 como entrada analógica del potenciómetro analógico, y al momento de variar la resistencia. Lo que hacía era refrescar la información que se muestra en la pantalla ubicando, en valor en binario y el valor en decimal del porcentaje de resistencia que se tiene en ese instante y se salía al menú presionando la opción 3.

Si se elige la opción 3 se iba al juego 21 el cual, se puso la generación de la carta aleatorio del jugador y de la mesa al presionar y soltar el botón que se encuentra en la patita Re3, este sumaba el valor acumulado de la carta del jugador y de la mesa, y mostraba en el LCD dicha información, esto refrescaba la información cada vez que se presionaba y soltaba el botón, al llegar un puntaje mayor a 21 para cualquiera de las manos, este verificaba quien era en ganador y encendía un led de la patita Rb6 y se salía al menú presionando la opción 3.

**2.- Diagrama de Bloques**

Teclado 4x4

Microcontrolador

PIC 16F887

Salida

Leds

Botón Jugador

LCD

2 Display

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

Definir variables globales, encera los puertos

Seleccionar puertos de I/O

Inicializar el teclado y el LCD

Definir variables locales

Limpiar el LCD y escribir el menú con Lcd\_Out(Fila,clumn,info);

Espera a que se presione el teclado opci= Keypad\_Key\_Click();

Opci=2

Opci=1

0

0

1

1

 Limpiar el LCD y Escribir el ejercicio2 con Lcd\_Out(Fila,clumn,info);

 Limpiar el LCD y Escribir el ejercicio1 con Lcd\_Out(Fila,clumn,info);

0

Valor de Re1 ADC\_Read(5); cambia y tem3=0

 Generar números aleatorios

RE2\_bit=0

0

1

tem3!=temp1

RE2\_bit=1

1

0

Convierte a binario y decimal y tem3=tem1

Acumula el valor de las cartas, y compara la mano de la mesa contra el jugador

Mostar el resultado en el LCD

Mostar el resultado en el LCD

0

1

0

Opci=3

Opci=3

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

1. Se configuran los puertos de entrada o salida, e inmediatamente se Inicializa el LC
2. Inicializamos la variable cnt=0, el teclado 4x4 y limpiamos pantalla del LCD
3. Se procede a escribir el menú, para poder elegir la opción.
4. Inicializamos la variable opci y este toma el valor que presionemos en el teclado4x4, mientras esta variable se mantenga en 0, indica que no ha sido presionada ninguna tecla por lo tanto queda encerrado en un lazo while
5. Cuando se presiona el uno en teclado, este realiza el ejercicio del potenciómetro, limpiando el LCD y procediendo a poner la información correspondiente.
6. Luego se lee la patita Re1 el cual se encuentra el potenciómetro analógico, y este valor se lo convierte a decimal y binario y después se procede a refrescar estos valores colocándolos en la LCD, esto se lo hace cada vez que varié la patita Re1, y al momento de presionar la tecla 3, este hace que vuelva al menú principal.
7. Cuando se presiona el dos en teclado, este realiza el ejercicio del juego21, limpiando el LCD y procediendo a poner la información correspondiente.
8. Luego se lee la patita Re3 el cual se encuentra un pulsador, que al momento de presionarlo y soltarlo, se genera un aleatorio, repartiendo el valor de la carta al jugador y a la mesa, transformamos el dato numérico del jugador y de la mesa a texto con el comando WordToStr para después mostrar el valor en la pantalla LCD.
9. Si se vuelve a presionar y soltar el pulsador, esta suma el valor anterior y así sigue acumulando, refrescando estos valores colocándolos en la LCD, y validándolo de tal forma que si la mesa o el jugador se pasa o llega al número 21, decide si hay un ganador procediendo a encender el led respetivo, si es que gano o perdió el jugador y al momento de presionar la tecla 3, este hace que vuelva al menú principal.

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

/\*

 \* Nombre del Proyecto:

 P2\_2.c

 \* Nombre del Autor:

 Néstor Naula López

 \* Description:

 Este proyecto deberá poder ejecutar dos ejercicios:

1. Ejercicio en lenguaje C propuesto por el profesor que consiste en la lectura de los valores analógicos de un potenciómetro colocado en patita RE0/AN5 del PIC16F887, presentando las variaciones de valores en binario y su equivalente decimal en una pantalla LCD 4x20.

2. Ejercicio en lenguaje C de un JUEGO PROPUESTO POR EL ESTUDIANTE haciendo uso de: teclado 4x4, pantalla LCD, display de 7 segmentos y LEDs.

 Funcionamiento:

a. Al encenderse el circuito deberán aparecer un menú inicial en la pantalla LCD 4x20.

b. Sus iniciales (de nombre y apellido) en displays de 7 segmentos.

c. Al presionar una tecla 1 se inicia el ejercicio 1, el cual mide el valor del potenciomentro, moestrando en la pantalla lcd 4x16 su equivalente en binario y decimal

d. Al presionar otra tecla 2 se inicia el ejercicio 2, el cual juega 21 el jugador vs la computadora, mostrandole si gano o no.

e. Al presionar otra tecla 3 sale del ejercicio 1 y 2, cunado se esta dentro de cualquiera de ellos.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

 \* Test configuration:

 MCU: PIC16F887

 Oscillator: HS, 08.0000 MHz

 SW: mikroC PRO for PIC

 \* NOTES:

\*/

unsigned short kp, cnt, opci, oldstate = 0,j,k,controlE=0;

char txt[6];

char txt2[1];

unsigned int temp\_res,temp1,temp2,temp3,temp4,i,num1,num2,num3,puntos;

// Keypad module connections

char keypadPort at PORTD;

// End Keypad module connections

// LCD module connections

sbit LCD\_RS at RB4\_bit;

sbit LCD\_EN at RB5\_bit;

sbit LCD\_D4 at RB0\_bit;

sbit LCD\_D5 at RB1\_bit;

sbit LCD\_D6 at RB2\_bit;

sbit LCD\_D7 at RB3\_bit;

sbit LCD\_RS\_Direction at TRISB4\_bit;

sbit LCD\_EN\_Direction at TRISB5\_bit;

sbit LCD\_D4\_Direction at TRISB0\_bit;

sbit LCD\_D5\_Direction at TRISB1\_bit;

sbit LCD\_D6\_Direction at TRISB2\_bit;

sbit LCD\_D7\_Direction at TRISB3\_bit;

// End LCD module connections

void main() {

 PORTB = 0xFF;

 PORTC = 0;

 TRISC = 0;

 cnt = 0; // Reset counter

 ANSEL = 0x32; // Configure AN5 pin as analog

 ANSELH = 0; // Configure other AN pins as digital I/O

 C1ON\_bit = 0; // Disable comparators

 C2ON\_bit = 0;

 TRISE = 0x04; // PORTE is input

 TRISA = 0; // PORTA is output

 TRISB = 0; // PORTB is output

 temp3=0;

 Keypad\_Init(); // Initialize Keypad

 Lcd\_Init(); // Initialize Lcd

 RB6\_bit =0;

 opci = 0;

 do {

 RB6\_bit =0;

 Lcd\_Cmd(\_LCD\_CLEAR); // Clear display

 Lcd\_Cmd(\_LCD\_CURSOR\_OFF); // Cursor off

 Lcd\_Out(1, 1, "Elija el juego :"); // Write message text on Lcd

 Lcd\_Out(2, 1, "1) Potenciomentro");

 Lcd\_Out(3, 1, "2) Juego 21");

 PORTC=0x48;

 PORTA=0x48;

 do {

 opci = Keypad\_Key\_Click(); // Store key code in kp variable

 delay\_ms(100) ;

 }

 while (!opci);

 if(opci == 1)

 Lcd\_Out(1, 18, " 1");

 if(opci == 2)

 Lcd\_Out(1, 18, " 2");

 delay\_ms(500) ;

 if(opci==1)

 {

 Lcd\_Cmd(\_LCD\_CLEAR);

 Lcd\_Cmd(\_LCD\_CURSOR\_OFF);

 do {

 PORTC=0x48;

 PORTA=0x48;

 temp\_res = ADC\_Read(5); // Get 10-bit results of AD conversion

 temp1 = (temp\_res)\*(100)/(1024);

 temp4=temp1;

 if(temp3!=temp1){

 for (i=0; i<8;i++){

 temp3=temp1;

 temp2 = temp4 % 2;

 temp4= temp4/2;

 if (temp2<=1) {

 WordToStr(temp2, txt);

 Lcd\_Out(2, 14-i, txt);

 }

 }

 }

 WordToStr(temp1, txt); // Transform counter value to string

 Lcd\_Out(3, 10, txt);

 Lcd\_Out(1, 2, "Juego Potenciometro"); // Write message text on Lcd

 Lcd\_Out(2, 1, "Binario :");

 Lcd\_Out(3, 1, "Decimal :");

 opci = Keypad\_Key\_Click();

 if(opci==3)

 break;

 } while(1);

 }

 if(opci==2 ){

 controlE=0;

 do{

 kp = 0;

 num2 =0;

 num3 =0;

 puntos=0;

 Lcd\_Cmd(\_LCD\_CLEAR); // Clear display

 Lcd\_Cmd(\_LCD\_CURSOR\_OFF);

 Lcd\_Out(1, 2, "Juego 21"); // Write message text on Lcd

 Lcd\_Out(2, 1, "CartaBan:"); // Write message text on Lcd

 Lcd\_Out(3, 1, "CartaJug:"); // Write message text on Lcd

 PORTC=0x01;

 PORTA=0x01;

 RC7\_bit =0;

 do{

 PORTC=0x48;

 PORTA=0x48;

 do{

 while(!RE2\_bit){

 puntos=0;

 j=(rand()%10)+1 ;

 opci = Keypad\_Key\_Click();

 delay\_ms(100) ;

 if(opci==3)

 controlE=1;

 };

 k=(rand()%10)+1;

 delay\_ms(100) ;

 puntos= 1;

 if(controlE==1){

 break;

 }

 }while(RE2\_bit);

 if(controlE==1)

 break;

 num2=num2+j;

 WordToStr(num2, txt2); // Transform counter value to string

 Lcd\_Out(2, 12, txt2);

 delay\_ms(500) ;

 num3=num3+k;

 WordToStr(num3, txt2); // Transform counter value to string

 Lcd\_Out(3, 12, txt2);

 delay\_ms(800) ;

 if(num3>21 || num2==21){

 Lcd\_Out(4, 2, "!Perdiste!"); // Write message text on Lcd

 PORTC=0xC8;

 delay\_ms(2500) ;

 break;

 } ;

 if(num2>21 || num3==21){

 Lcd\_Out(4, 2, "!Ganaste!");

 RB6\_bit =1;

 delay\_ms(300) ;

 RB6\_bit =0;

 delay\_ms(300) ;

 RB6\_bit =1;

 delay\_ms(300) ;

 RB6\_bit =0;

 delay\_ms(300) ;

 RB6\_bit =1;

 delay\_ms(300) ;

 RB6\_bit =0;

 break;

 } ;

 }while(1);

 if(controlE==1)

 break;

 }while(1);

 }

 } while (1);

 }

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









**7.- Conclusiones**

* Se utilizo funciones y procedimientos que nos otorga MicroC pro, para poder utilizar la pantalla LCD y el teclado 4X4, ya que son necesarios ciertos comandos para poder hacer uso de estos implementos, con la ayuda de los dos, se pudo dar la libertad al usuario de elegir entre la opción 1 y la opción 2 o volver al menú que es la opción 3, y para saber si lo que hicimos en MicroC pro tuvo un resultado favorable, se procedió a simular en proteus.
* Se pudo interactuar mas con el usuario o la persona que tenga algún requerimiento gracias al proyecto, ya que con ayuda de todo estos implementos de visualización nos permitió saber lo que la personaba necesitaba y le dio la opción a elegir y asi poder mostrarle de una mejor manera, esto se lo puso simular con el programa proteus el cual cuenta con pantalla LCD y con teclado, y así poder conectarlo al PIC y ver como sale de una forma más aproximada a la realidad.
* Se aprendió y se hizo uso de los comandos para la pantalla LCD y el teclado 4X4, los cuales nos fue de gran ayuda ya que nos permitió, mostrar textos como el menú , también mostrar las conversiones a decimal y binario del potenciómetro analógico, de forma numérica para ser convertido a texto y ser insertado en el LCD establecida en el PIC, no olvidando que con uno de los pulsadores del teclado podía retornar al menú principal, esto se debe gracias a que el, teclado 4x4 es muy práctico y fácil de usar.

**8.- Recomendaciones**

* Se recomienda que al momento de implementar dos o más ejercicios ya sea juegos en el PIC, estos ejercicios deben estar probados de manera independiente, para saber si funciona o no, y después de haber hecho esto, ahí si se los puede probar conjuntamente.
* No se debe olvidar de encerar los puertos, y definir si estos son de entrada o salida ya que si no se lo hace al momento de hacer cualquier práctica no, nos va a salir lo que nosotros requerimos y esto traería complicaciones al momento de implementarlo en proteus.
* Como ahora se trabaja con un teclado no se debe olvidar hacer un Delay cuando se presiona y suelta cualquier botón ya que como nosotros no somos tan rápidos al presionarlo y soltarlo esto puede traer problemas y no salir lo que nosotros implementamos en el código.

.