**UNIVERSIDAD PÚBLICA DE EL ALTO INGENIERÍA DE SISTEMAS**



**INGENIERÍA DE CONTROL LABORATORIO No. 6**

**1. OBJETIVO**

**DISPOSITIVOS DE ENTRADA Y SALIDA DE PROCESADOR PIC**

✓ Conocer a usar las librerías Keypad y LCD de MikroC para realizar la interface entre el PIC18F4550, un teclado matricial de 4 filas por 4 columnas y una tarjeta de pantalla de cristal líquido (LCD) de 2 filas y

16 columnas.

✓ Comprobar el funcionamiento del conversor ADC, de 10 bits y multiplexado en 13 canales

✓ Manejar la Ayuda de las librerías incorporadas en el IDE MikroC

✓ Mostrar en simulación Virtual y también en Hardware su funcionamiento de los ejemplos propuestos.

**2. EQUIPOS Y MATERIALES**

✓ PC equipado con software MiKroC y Proteus

✓ Grabador Bootloader, Hardware

✓ Guía de laboratorio de Ingeniería de Control

✓ Teclado matricial de 4 filas por 4 columnas, Pantalla de cristal líquido (LCD)

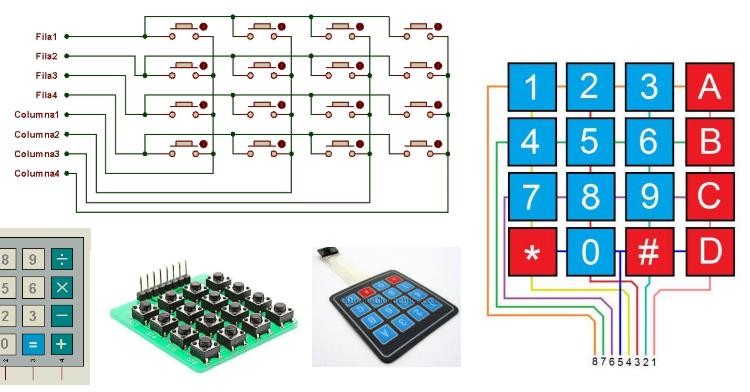
✓ Otros

**3. MARCO TEÓRICO**

También conocidos como puertos de E/S, generalmente agrupadas en puertos de 8 bits de longitud, permiten leer datos del exterior o escribir en ellos desde el interior del microcontrolador, el destino habitual es el trabajo con dispositivos simples como relés, LED, o cualquier otra cosa que se le ocurra al programador. Algunos puertos de E/S tienen características especiales que le permiten manejar salidas con determinados requerimientos de corriente, o incorporan mecanismos especiales de interrupción para el procesador. Típicamente cualquier pin de E/S puede ser considerada E/S de propósito general, p ero como los microcontroladores no pueden tener infinitos pines, ni siquiera todos los pines que queramos, las E/S de propósito general comparten los pines con otros periféricos. Para usar un pin con cualquiera de las características a él asignadas debemos configurarlo mediante los registros destinados a ellos.

**3.1. TECLADO MATRICIAL**

Un teclado es un arreglo de pulsadores conectados por filas y columnas.



**3.2. LIBRERÍA KEYPAD EN MIKROC**

MikroC provee una librería para trabajar con un teclado matricial de 4x4. Las rutinas también pueden ser usadas para teclados de 4x1, 4x2 o 4x3. Keypad\_Init: Inicializa el puerto del microcontrolador para trabajar

con el teclado Keypad\_Key\_Press: Función que devuelve el código de tecla del keypad cuando la tecla es

presionada. Si no hay tecla presionada, la función devuelve Cero (0). Keypad\_Key\_Click: Función que devuelve el código de tecla del keypad cuando la tecla es presionada y liberada. Si no hay tecla presionada, la

función devuelve Cero (0). Códigos de Tecla devueltos por las funciones Keypad\_Key\_Press y

Keypad\_Key\_Click:

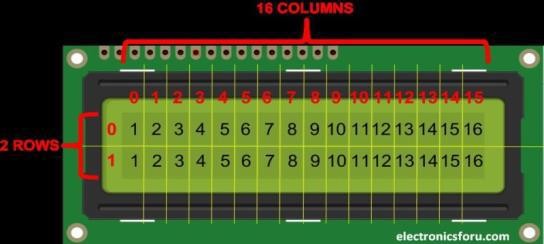
Filas

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 5 | 9 | 13 |
| 2 | 6 | 10 | 14 |
| 3 | 7 | 11 | 15 |
| 4 | 8 | 12 | 16 |

Columnas

**3.3. TARJETA DE PANTALLA DE CRISTAL LÍQUIDO LCD**

- Se utiliza para visualizar los diferentes mensajes (en código ASCII) en una pantalla LCD. Está basad o en el microcontrolador HD44780 (Hitachi) integrado y permite visualizar todas las letras de alfabeto (mayúsculas y minúsculas), letras de alfabeto griego, signos de puntuación, símbolos matemáticos etc. También es posible visualizar símbolos creados por el usuario



- La tarjeta dispone de 16 pines enumerados (14 si el LCD no posee luz de iluminación de fondo)

- La función de los pines se muestra en la siguiente tabla

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FUNCIÓN | No. PIN | NOMBRE | VALOR | DESCRIPCION |
| Tierra | 1 | Vss | - | GND |
| Alimentación | 2 | Vdd | - | +5V |
| Contraste | 3 | Vee | - | 0 - Vdd |
| Control  De  Funcionamiento | 4 | RS |  | Registrar Select (Selección de Registro)  D0 – D7 considerados como comandos/datos |
| 5 | R/W | 0  1 | Read/Write (Lectura/Escritura) |
| 6 | E | Transición  De 1 a 0 | Enable (Habilitación) |
| Datos  / Comandos | 7 | D0 | 0/1 | Bit 0 LSB |
| 8 | D1 | 0/1 | Bit 1 |
| 9 | D2 | 0/1 | Bit 2 |
| 10 | D3 | 0/1 | Bit 3 |
| 11 | D4 | 0/1 | Bit 4 |
| 12 | D5 | 0/1 | Bit 5 |
| 13 | D6 | 0/1 | Bit 6 |
| 14 | D7 | 0/1 | Bit 7 MSB |
|  | 15 | A | - | Ánodo de luz de fondo |
|  | 16 | K | - | Cátodo de luz de fondo |

- El contraste del visualizador depende del voltaje de alimentación y de si los mensajes se visualizan en una o dos líneas. Por esta razón, un voltaje variable 0-Vdd se aplica al pin marcado como Vee. Un potenciómetro se utiliza con frecuencia para este propósito.

- La tarjeta de pantalla LCD dispone de luz de fondo incorporada (diodos LED azules o verdes). Para utilizarla se debe de conectar el Ánodo (A) a +5V y el Cátodo (K) a tierra.

**3.4. LIBRERÍA LCD LIBRARY EN MIKROC**

- El lenguaje MikroC provee la librería LCD Library para trabajar con pantallas de cristal líquido capaces de desplegar caracteres ASCII alfanuméricos.

- En el modo de conexión de 4-bit, se requieren 6 líneas de conexión entre el microcontrolador PIC y el

LCD: 2 líneas de Control (RS: Register Select, EN: Enable) y 4 líneas de Datos (D4, D5, D6, D7 )

- Para especificar la conexión física entre el microcontrolador PIC y el LCD, en el programa se declaran

variables que indican los bits de los puertos del PIC correspondientes a cada una de las 6 líneas de conexión del LCD. Estos bits pueden modificarse de acuerdo a las necesidades del diseño.

- Previo al uso del LCD, se debe inicializar (una sola vez) con la instrucción: LCD\_Init(); Se pueden



enviar mensajes al LCD indicando la fila (1 o 2), columna (1 al 16) y el texto (ASCII entre comillas) con

la instrucción: LCD\_Out(fila,columna,”Texto”);

- En caso de imprimir a partir de la posición actual del cursor del LCD se debe utilizar la función: Lcd\_Out\_Cp("Texto");

- Para imprimir un solo carácter se debe utilizar: Lcd\_Chr(2, 3, 'i'); o Lcd\_Chr\_Cp(2, 3, 'i'); para imprimir

el carácter en la posición actual del cursor.

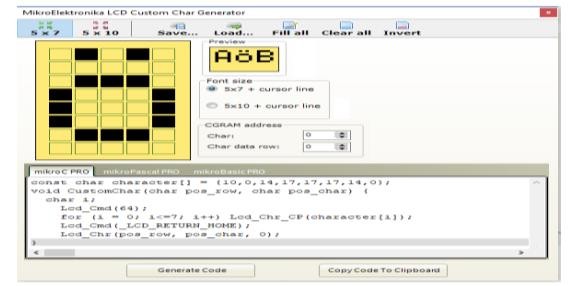
- Si se requiere mostrar Variables (byte, integer, float, etc.) en el LCD, se deben utilizar funciones de conversión de la librería de conversiones (Conversions Library): ByteToStr, WordToStr, IntToStr, FloatToStr, etc. (Ver Ejemplo B)

- También se pueden enviar comandos al LCD para su control con la instrucción: LCD\_Cmd(Comando

LCD); La siguiente tabla muestra los comandos disponibles:

|  |  |
| --- | --- |
| COMANDO LCD | PROPÓSITO |
| \_LCD\_FIRST\_ROW | Mover cursor a la primera fila |
| \_LCD\_SECOND\_ROW | Mover cursor a la segunda fila |
| LCD\_THIRD\_ROW | Mover cursor a la tercera fila |
| \_LCD\_FOURTH\_ROW | Mover cursor a la cuarta fila |
| \_LCD\_CLEAR | Limpiar pantalla |
| \_LCD\_RETURN\_HOME | Retornar el cursor a la fila 1, columna 1 |
| \_LCD\_CURSOR\_OFF | Apagar cursor |
| \_LCD\_UNDERLINE\_ON | Activar cursor ‘subrayado’ |
| \_LCD\_BLINK\_CURSOR\_ON | Activar cursor ‘parpadeante’ |
| \_LCD\_MOVE\_CURSOR\_LEFT | Mover cursor a la izquierda |
| \_LCD\_MOVE\_CURSOR\_RIGHT | Mover cursor a la derecha |
| \_LCD\_TURN\_ON | Encender la pantalla del LCD |
| \_LCD\_TURN\_OFF | Apagar la pantalla del LCD |
| \_LCD\_SHIFT\_LEFT | Desplazar la pantalla a la izquierda |
| \_LCD \_SHIFT\_RIGHT | Desplazar la pantalla a la derecha |

**3.5. CARACTERES PERSONALIZADOS CON LA HERRAMIENTA LCD CUSTOM CHARACTER**



✓ Si se requiere desplegar caracteres que no están incluidos en la memoria CGROM de la tarjeta del display

LCD, se pueden diseñar hasta 8 caracteres personalizados que se almacenan en la memoria CGRAM.

✓ Para facilitar la creación de caracteres personalizados, MikroC provee la herramienta LCD Custom

Character con la cual es muy sencillo diseñar y generar el código para el carácter.

✓ El código generado es una función (cuyo nombre debe ser renombrado para cada carácter generado) que tiene como parámetros de entrada, la fila y la columna donde se requiere visualizar al carácter en el LCD.

✓ Para abrir la herramienta, seleccionar del menú principal de MikroC, la opción ‘Tools – LCD Custom

Character’

**3.6. LIBRERÍAS INCORPORADAS EN MIKROC**

Una librería es un conjunto de variables y funciones, previamente escritas y compilad a, que permiten

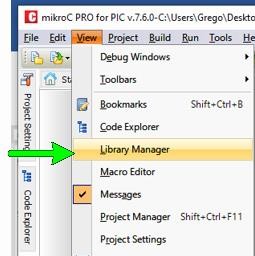
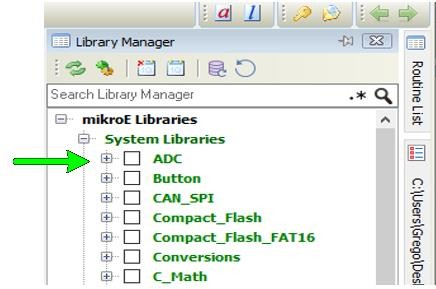
desarrollar tareas comunes de manera más rápida y sencilla.

MikroC incorpora una amplia variedad de librerías incorporadas disponibles al usuario.

Antes de utilizar alguna de ellas en el programa, es necesario habilitarlas seleccionándolas de la lista de librerías de MikroC existentes. Si el compilador encuentra una función desconocida durante la ejecución de

programa, primero buscará su declaración en las librerías seleccionadas.

✓ Para habilitar una librería ir al menú **View** - **Library Manager** y en ventana emergente elegir **mikroE Libraries** - **System Libraries**. De la lista seleccionar las librerías que se utilizarán en su proyecto.



Usted probablemente ha notado que en los ejemplos anteriores hemos utilizado algunas funciones como son

'Delay\_ms', 'LCD\_out', 'LCD\_cmd' etc. Estas funciones están definidas en las librerías contenidas en el compilador *mikroC*. Una librería representa un código compilado, anteriormente escrito en mikroC, que contiene un conjunto de variables y funciones. Cada librería tiene un propósito específico. Por ejemplo, la

librería LCD contiene funciones de visualización de la pantalla LCD, mientras que *C\_math* proporciona algunas funciones matemáticas. Antes de utilizar alguna de ellas en el programa, es necesario comunicárselo al compilador al marcarlas en la lista de las librerías del compilador existentes. Si el compilador encuentra una función desconocida durante la ejecución de programa, primero va a buscar su declaración en las librerías marcadas.

Aparte de las librerías existentes, es posible crear las librerías y luego utilizarlas en el programa. El procedimiento de cómo crear librerías se describe en detalles en Help (*Ayuda*) del compilador. El compilador mikroC incluye **tres** tipos de librerías:

✓ **LIBRERÍAS ANSI C ESTÁNDAR:**

|  |  |
| --- | --- |
| **LIBRARÍA** | **DESCRIPCIÓN** |
| ANSI C Ctype Library | Utilizada principalmente para probar o para convertir los datos |
| ANSI C Math Library | Utilizada para las operaciones matemáticas de punto flotante |
| ANSI C Stdlib Library | Contiene las funciones de librerías estándar |
| ANSI C String Library | Utilizada para realizar las operaciones de cadenas y de manipulación de memoria |

✓ **LIBRERÍAS MISCELÁNEAS:**

|  |  |
| --- | --- |
| **LIBRARÍA** | **DESCRIPCIÓN** |
| Button Library | Utilizada para desarrollar los proyectos |
| Conversion Library | Utilizada para la conversión de tipos de datos |
| Sprint Library | Utilizada para formatear los datos con facilidad |
| PrintOut Library | Utilizada para formatear los datos e imprimirlos |

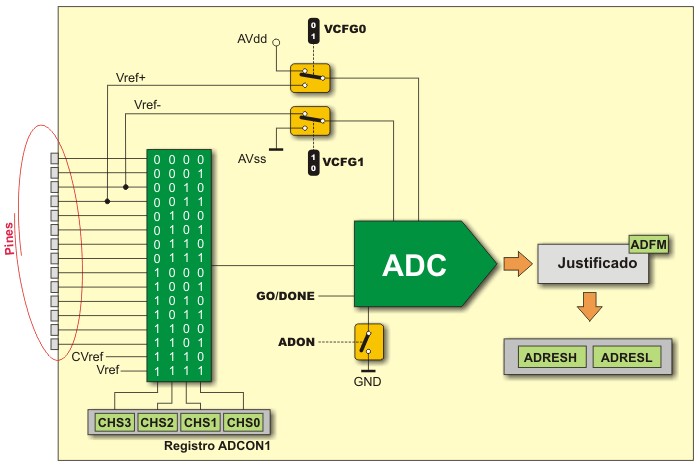
|  |  |
| --- | --- |
| Time Library | Utilizada para cálculos de tiempo (formato UNIX time) |
| Trigonometry Library | Utilizada para la implementación de funciones trigonométricas fundamentales |
| Setjmp Library | Utilizada para los saltos de programa |

✓ **LIBRERÍAS PARA EL HARDWARE:**

|  |  |
| --- | --- |
| **LIBRARÍA** | **DESCRIPCIÓN** |
| ADC Library | Utilizada para el funcionamiento del convertidor A/D |
| CAN Library | Utilizada para las operaciones con el módulo CAN |
| CANSPI Library | Utilizada para las operaciones con el módulo CAN externo (MCP2515 o MCP2510) |
| Compact Flash Library | Utilizada para las operaciones con las tarjetas de memoria *Compact Flash* |
| EEPROM Library | Utilizada para las operaciones con la memoria EEPROM incorporada |
| EthernetPIC18FxxJ60 Library | Utilizada para las operaciones con el módulo Ethernet incorporado |
| Flash Memory Library | Utilizada para las operaciones con la memoria Flash incorporada |
| Graphic Lcd Library | Utilizada para las operaciones con el módulo LCD gráfico con resolución 128x64 |
| I2C Library | Utilizada para las operaciones con el módulo de comunicación serial I2C incorporado |
| Keypad Library | Utilizada para las operaciones con el teclado (botones de presión 4x4) |
| Lcd Library | Utilizada para las operaciones con el LCD (de 2x16 caracteres) |
| Manchester Code Library | Utilizada para la comunicación utilizando el código *Manchester* |
| Multi Media Card Library | Utilizada para las operaciones con las tarjetas multimedia MMC flash |
| One Wire Library | Utilizada para las operaciones con los circuitos utilizando la comunicación serial *One*  *Wire* |
| Port Expander Library | Utilizada para las operaciones con el extensor de puertos MCP23S17 |
| PS/2 Library | Utilizada para las operaciones con el teclado estándar PS/2 |
| PWM Library | Utilizada para las operaciones con el módulo PWM incorporado |
| RS-485 Library | Utilizada para las operaciones con los módulos utilizando la comunicación serial  RS485 |
| Software I2C Library | Utilizada para simular la comunicación I2C con software |
| Software SPI Library | Utilizada para simular la comunicación SPI con software |
| Software UART Library | Utilizada para simular la comunicación UART con software |
| Sound Library | Utilizada para generar las señales de audio |
| SPI Library | Utilizada para las operaciones con el módulo SPI incorporado |
| SPI Ethernet Library | Utilizada para la comunicación SPI con el módulo ETHERNET (ENC28J60) |
| SPI Graphic Lcd Library | Utilizada para la comunicación SPI de 4 bits con el LCD gráfico |
| SPI LCD Library | Utilizada para la comunicación SPI de 4 bits con el LCD (de 2x16 caracteres) |
| SPI Lcd8 Library | Utilizada para la comunicación SPI de 8 bits con el LCD |
| SPI T6963C Graphic Lcd  Library | Utilizada para la comunicación SPI con el LCD gráfico |
| UART Library | Utilizada para las operaciones con el módulo UART incorporado |
| USB Hid Library | Utilizada para las operaciones con el módulo USB incorporado |

**3.7. CONVERSOR ANÁLOGO DIGITAL EN EL PIC18F4550**

**DIAGRAMA EN BLOQUES**



El convertidor analógico digital, ADC, es capaz de convertir una señal de voltaje variable en valores digitales a una resolución de 8 o 10 bits. Además, la frecuencia de muestreo puede llegar hasta a 50 kHz.

A continuación, se muestra las características y se describe los registros involucrados con el convertidor

ADC.

- Resolución programable por software en modo de 8 y 10 bit.

- El resultado de la conversión se almacena en los registros ADRESH y ADRESL

- 13 canales de entradas multiplexada desde AN0 hasta AN12 distribuidas en los puertos RA (AN0 a

AN4), RE (AN5 a AN7) y RC (AN8 a AN12)

- Rango del voltaje de entrada para el ADC desde 0 Voltios hasta +5 Voltios

- Referencia de voltaje interna o externa

- Modo de conversión ADC (continua o individual)

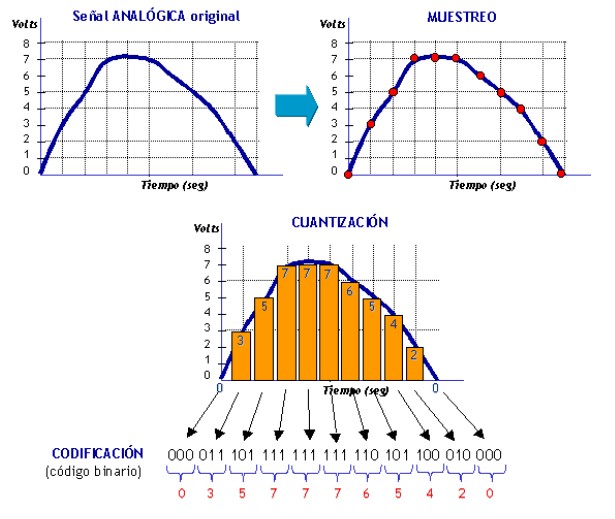
- Inicio de conversión ADC por Auto-Disparo en fuentes de Interrupción

- Interrupción ADC al completar la conversión

**CODIFICACIÓN DE LA SEÑAL EN CÓDIGO BINARIO**

Después de realizada la cuantificación, los valores de las tomas de voltajes se representan numéricamente por medio de códigos y estándares previamente establecidos. Lo más común es codificar la señal digital en

código numérico binario.



**REGISTROS DE CONTROL**

El módulo A/D tiene cinco registros:

1. Registro de control A/D ADCON0

2. Registro de control A/D ADCON1

3. Registro de control A/D ADCON2

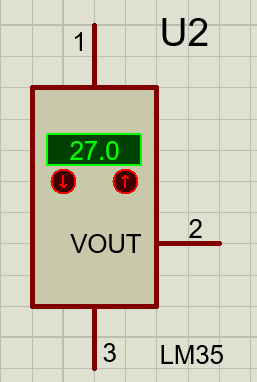
4. Registro de resultados alto (HIGH) A/D ADRESH

5. Registro de resultados bajo (HIGH) A/D ADRESL

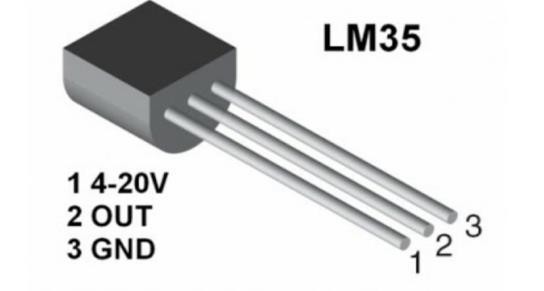
**3.8. SENSOR DE TEMPERATURA LM35**

El sensor LM35, es un dispositivo activo de 3 terminales que permite adquirir la temperatura ambiente en

rangos de -55 a 150 grados Celsius o centígrados. Este dispositivo es de fácil implementación dado que solo cuenta con dos terminales de polarización, y una salida de voltaje directamente proporcional a la temperatura. La apariencia física del sensor y su distribución de pines, así como la vista en ISIS, son las que se pueden ver en las siguientes figuras:



Este sensor puede ser polarizado de 4 a 30 voltios y tiene una salida de 10m voltios por cada grado Celsius. Para el caso de este sensor, se verá definido por las siguientes relaciones:



1023

=

5

𝑅����

������

Donde Radc es el resultado binario de la conversión ADC.

(5��) ∗ (𝑅����)

������ =

1023

Trabajando con la relación del sensor que es: 10mVoltios por cada 1º Celsius.

10�𝑉

=

1°𝐶

������

�°𝐶

Donde n es la temperatura en grados Celsius, que esta registrado el sensor y tenemos la temperatura n es:

nºC = (1°��)(��𝑎��) = 500 ��𝑎��

(10𝑚��)

1023

**4. PRE INFORME**

4.1. Objetivos de laboratorio.

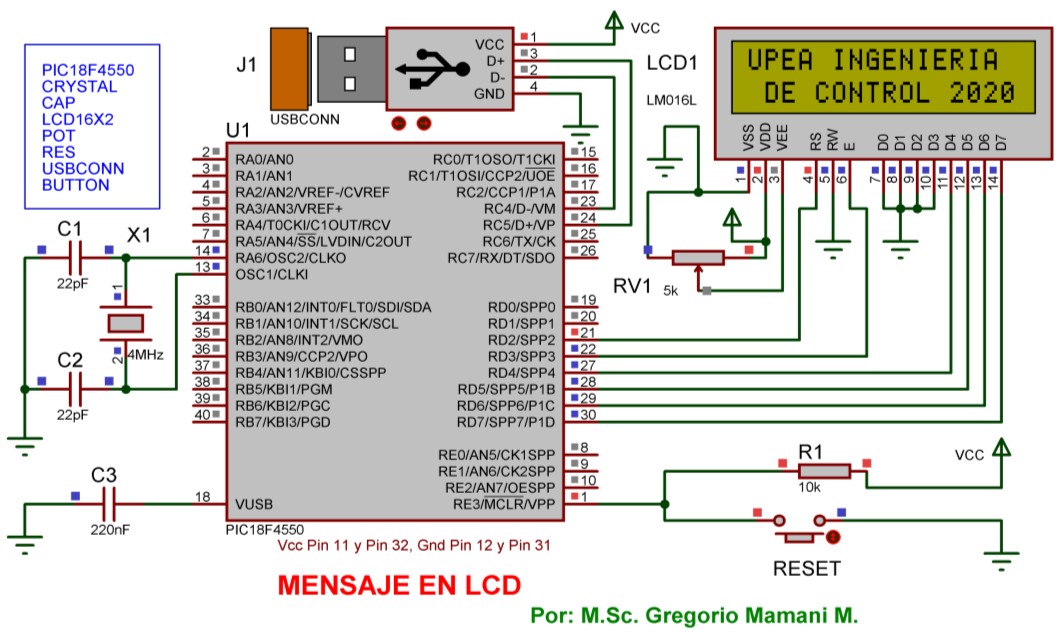
4.2. Características de LM35

4.3. Circuito de protección de Pic18F4550 para activar: Ventilador, Foco 220VAC y chapa eléctrica.

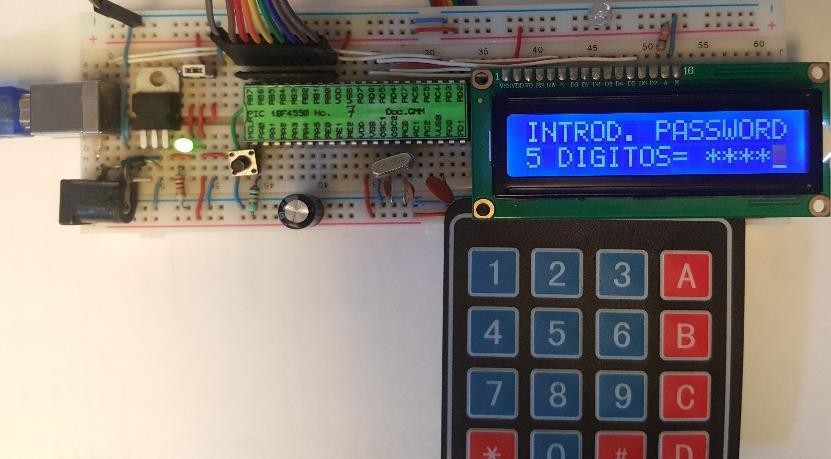
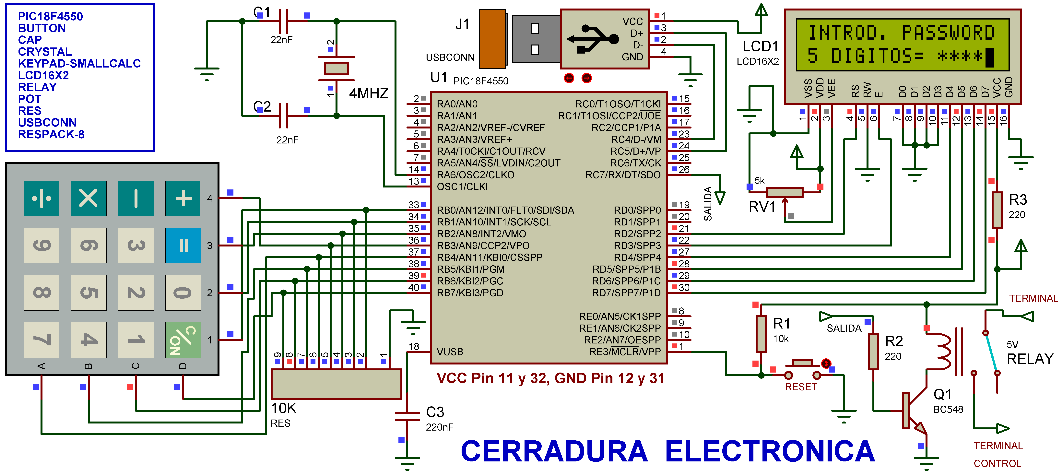
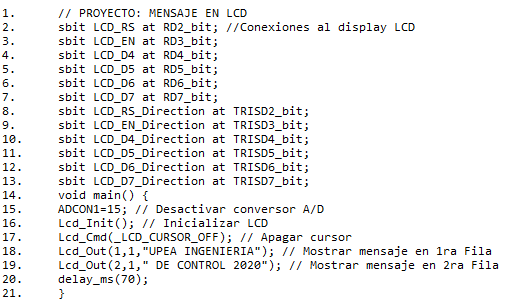
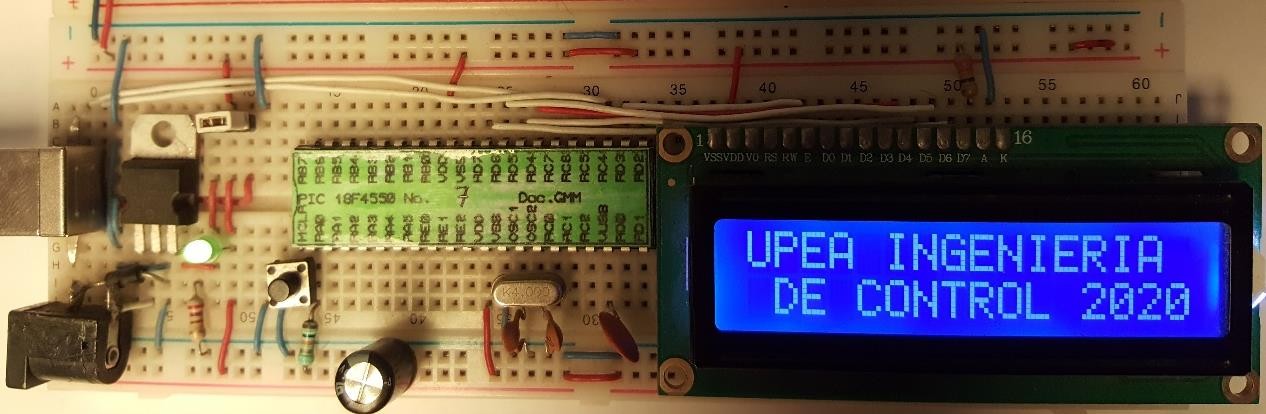
**5. PROCEDIMIENTO**

5.1. Diseño de programa en MikroC mensaje que muestra y que se desplace de derecha a izquierda “UPEA INGENIERIA DE CONTROL 2020” y demostrar su funcionamiento en Plataforma de simulación

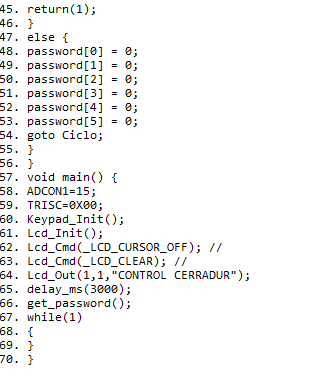
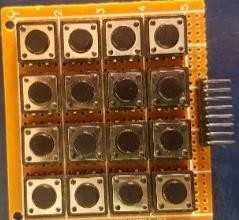
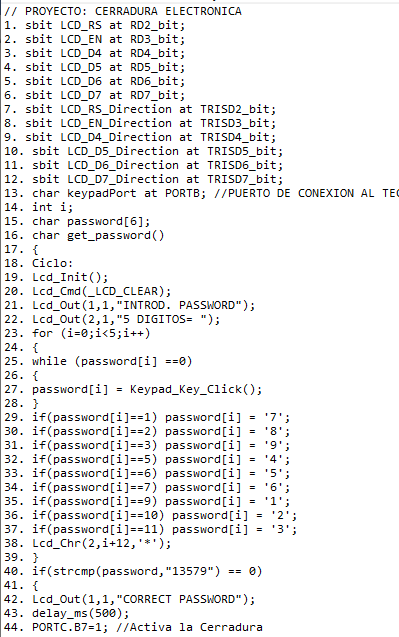
Virtual y hardware.



5.2. Diseño de programa en MikroC “CERRADURA ELECTRÓNICA CON CONTROL DE ACCESO POR TECLADO” password "13579". Modificar, demostrar su funcionamiento en Simulación Virtual y Hardware.



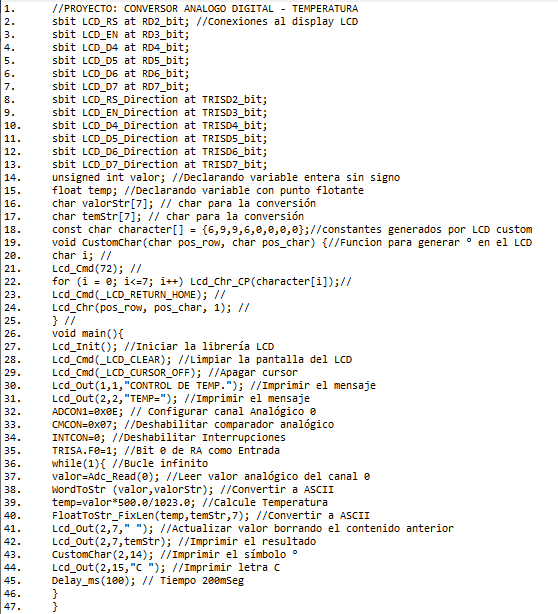
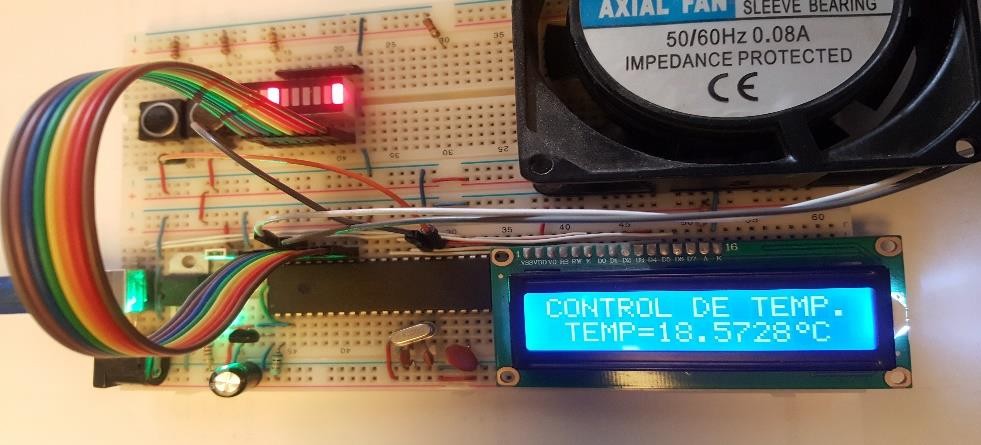
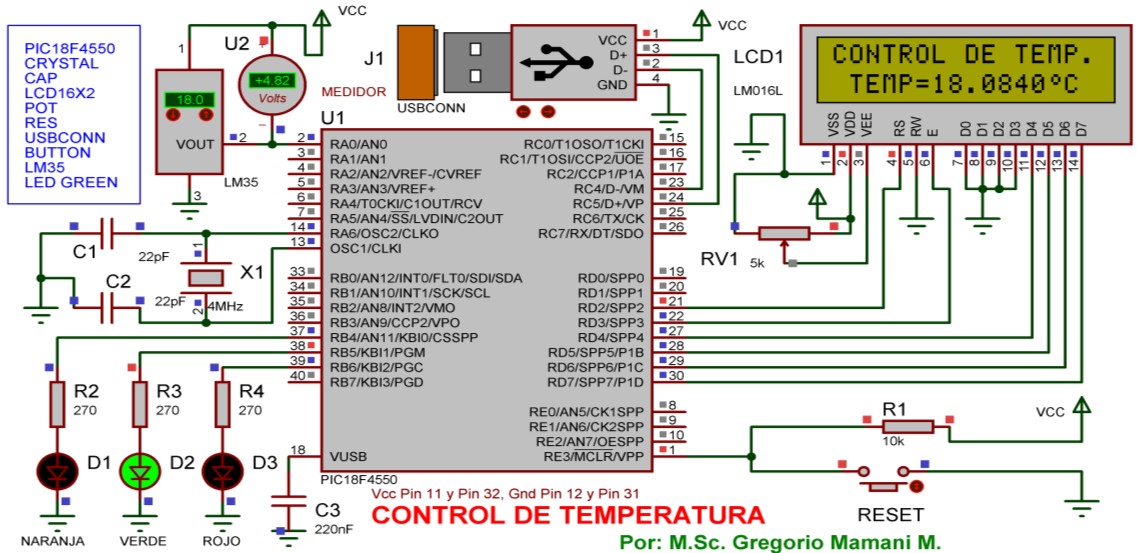
Utilice Teclado con pulsadores físicos



5.3. Diseño de “CONTROL DE TEMPERATURA” con rango de funcionamiento de: menores a 17°C active led naranja y calefón (foco de 60W 220 VAC), a mayores a 25°C active led rojo y ventilador (ventilador

de fuente de 12VDC), a temperatura igual a 17 ó 25 active led verde en estado aceptable. Demostrar su

funcionamiento en Simulación Virtual y en Hardware



5.4. Diseñe una aplicación relacionado con el presente laboratorio, tema de proyecto a elección del estudiante.

**6. CUESTIONARIO**

6.1. Explicar en detalle los tres bloques de memoria que posee el display LCD: Memoria DDRAM, CGRAM

y CGROM

6.2. Conjunto de caracteres ASCII que puede desplegar un display LCD, mostrar en una tabla

7. **CONCLUSIONES**