****

**Escuela:** Centro de enseñanza técnica industrial – Colomos  
**Carrera:** Desarrollo de software  
**Materia:** Sistemas embebidos.  
**Tema:** Act10 Prác6 Comunicación Serial-ADC. **Nombre del alumno:** Carlos Daniel Lozano Vázquez **Registro:** 18300249. **Grupo y grado:** 6ºB1. **Fecha:** 09/03/2021

**Actividad 10: Práctica #6 Comunicación Serial-ADC.**

**2.- Objetivo**

Objetivo de la práctica: Utilizar la comunicación serial y el ADC de un microcontrolador RISC para realizar una aplicación básica demostrativa utilizando programación en lenguaje c.

**3.- Descripción.**

a) Utilizar dos PIC16F877A conectarlos entre sí para comunicación serial. A uno conectarle un dip sw(8), este debe funcionar como trasmisor (emisor) y al otro PIC conectarle 8 LEDs, este debe funcionar como receptor. Lo que se escriba en el dip sw de lado del emisor se debe mostrar en los LEDs del receptor.  
  
b) A un PIC16F877A conectarle un sensor (temperatura, luz, etc.) o un potenciomentro, para introducir valores analógicos al PIC. Se debe establecer un valor limite; si el valor que se lea del sensor (entrada) supera ese limite, se debe mostrar en una LCD el mensaje “PASO EL LIMITE” de lo contrario mostrar “BAJO EL LIMITE”. En todo momento se debe mostrar en la LCD el valor leído del sensor.  
  
En una Parte (Juntar las anteriores)  
  
a) En lugar de utilizar dip swicth para introducir el dato, se puede utilizar el sensor, leer el dato del sensor, mandarlo al otro PIC que tendrá conectado una LCD como salida. Se debe establecer un valor limite; si el valor que se lea del sensor (entrada) supera ese limite, se debe mostrar en una LCD el mensaje “PASO EL LIMITE” de lo contrario mostrar “BAJO EL LIMITE”. En todo momento se debe mostrar en la LCD el valor leído del sensor.  
Nota:  
Implementar para este caso se debe entender como: Realizar los programas (emisor y receptor), programar los PICs y comprobar su funcionamiento en un circuito en protoboard o en simulador.

**4.- Resumen del tema de la práctica.**

El PIC 16F877A es un microcontrolador, el cual se refiere a un dispositivo programable capaz de realizar diferentes actividades que requieran del procesamiento de datos digitales y del control y comunicación digital de diferentes dispositivos. Este microcontrolador es fabricado por la familia MicroChip, a la cual se le denomina PIC.

Este contiene una memoria interna que almacena dos tipos de datos; las instrucciones, que corresponden al programa que se ejecuta, y los registros, es decir, los datos que el usuario maneja, así como registros especiales para el control de las diferentes funciones del microcontrolador.

El PIC que se estará utilizando es de arquitectura RISC, lo que quiere decir que es más rápido, pero tiene menos instrucciones. Los elementos principales del microcontrolador son la ALU (Unidad Lógico Aritmética), memoria del programa, memoria de registros, y pines I/O (entrada y/0 salida). La ALU es la encargada de procesar los datos dependiendo de las instrucciones que se ejecuten (ADD, OR, AND), mientras que los pines son los que se encargan de comunicar al microcontrolador con el medio externo; la función de los pines puede ser de transmisión de datos, alimentación de corriente para el funcionamiento de este o pines de control específico.

Las características principales del PIC son:

* Soporta modo de comunicación serial, posee dos pines para ello.
* Amplia memoria para datos y programa.
* Memoria reprogramable: La memoria en este PIC es la que se denomina FLASH, este tipo de memoria se puede borrar electrónicamente.
* Set de instrucciones reducido (tipo RISC), pero con las instrucciones necesarias para facilitar su manejo.

LCD 16x2

Es un pequeño dispositivo con pantalla de cristal líquido que cuenta con dos filas, de dieciséis caracteres cada una, que se utiliza para mostrar información, por lo general alfanumérica. También es empleado para la visualización de contenidos o información de una forma gráfica, mediante caracteres, símbolos o pequeños dibujos dependiendo del modelo. Está gobernado por un microcontrolador el cual dirige todo su funcionamiento.

Es una pantalla plana basada en el uso de una sustancia líquida atrapada entre dos placas de vidrio, haciendo pasar por este una corriente eléctrica a una zona específica, para que así esta se vuelva opaca, y además cuenta (generalmente) con iluminación trasera.

Las pantallas LCD de color, cada pixel individual se divide en tres cédulas o sub pixeles con los colores RGB (Rojo, Verde y Azul) respectivamente. Y así cada píxel puede controlarse para  producir una gran variedad de colores distintos.

Entre las ventajas que se tienen al utilizar este tipo de dispositivos están el mínimo consumo de energía o corriente eléctrica, la programación es sumamente sencilla.

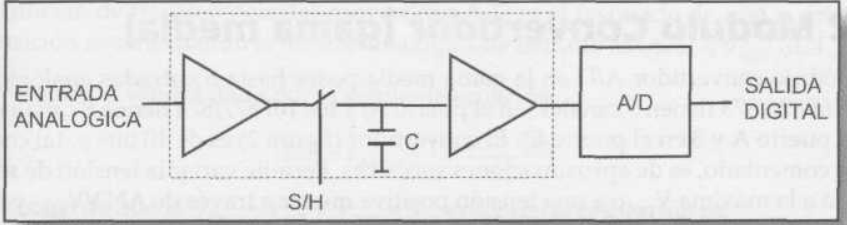
Comúnmente, el total de caracteres que se pueden representar en dispositivo es de 256, de los cuales los primeros 240 son los símbolos alfanuméricos comunes (letras mayúsculas y minúsculas, signos de puntuación, algunos caracteres especiales, comunes y números) y el resto son caracteres que pueden ser introducidos o definidos por el usuario.

Debido a la retroiluminación con la que cuentan estos dispositivos, es posible lograr un alto grado de contraste de colores y por lo tanto tener gran claridad en el mensaje mostrado, la pantalla es sumamente delgada, en consecuencia su tamaño es pequeño y su vida útil es bastante más extensa que muchos otros displays.

Quizás la única desventaja de este tipo de pantallas es que no se tiene una diversificación muy fina o exacta de los colores, por lo que en ambientes con una iluminación poco favorable no es posible visualizar de la mejor manera los mensajes mostrados en la pantalla.

ADC en el PIC16f877

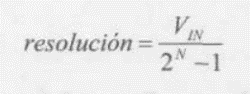
Los microcontroladores PIC pueden incorporar un módulo de conversión de señal analógica a señal digital. Los módulos AD que utiliza Microchip hacen un muestreo y retención (sample and hold) con un condesador y después utiliza el módulo de conversión, EL módulo de conversión A/D es del tipo de aproximaciones sucesivas.



El convertidor de aproximaciones sucesivas se utiliza en aplicaciones en donde se necesita altas velocidades de conversión. Se basa en realizar secesivas comparaciones de forma ascendete o descendete hasta encontrar un valor digital que iguale la tensión cargada por el convesor D/A y la tensión de entrada.

Durante la fase de muestreo el interruptor se cierra y el condensador se carga a la tensión de entrada (el tiempo que el interruptor permanece cerrado es fundamental para la correcta carga del condensador). Una vez abierto el interruptor, el condensador mantendrá teóricamente la tensión de entrada mientras el módulo A/D realiza la conversión.

El módulo de conversión se caracteriza por parámetros como los siguientes.

* Rango de entrada
* Número de bits
* Resolución
* Tensión de fondo de escala
* Error de conversión

El módulo que utilizan los PIC de gama media tiene un numero de 10 bits por lo que su resolución es :

Donde Vin es la tensión de entrada y N el número de bits del convertidor.

Los PICS permiten cambiar la tensión de referencia en un valor absoluto de 0 a +Vref o en un margen de -Vref a + Vref.

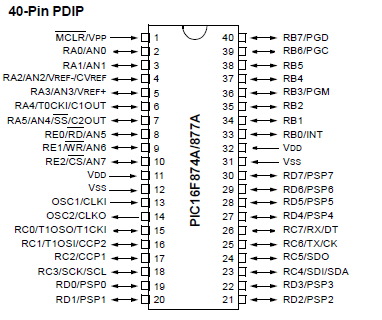
Si el ADC, está situado a la salida de un sensor (que habitualmente aporta una señal de amplitud débil) es esencial que en la etapa de conversión no se genere un nivel de ruido que impida la conversión real de la señal de entrada.

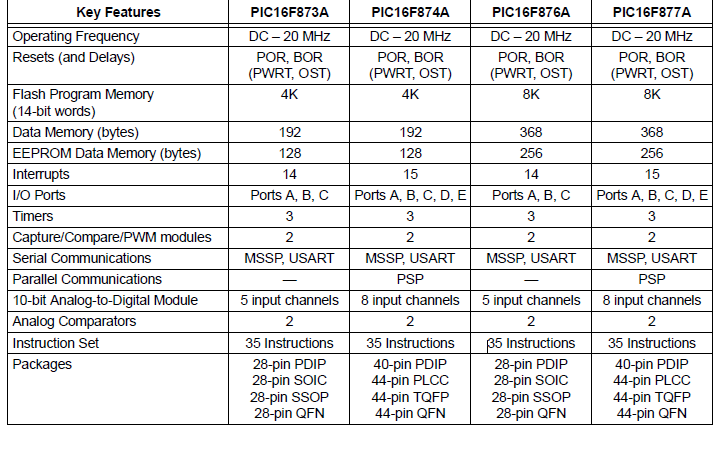
La arquitectura más extendida entre los ADC es la basada en el método de las aproximaciones sucesivas. Su éxito se fundamenta en conseguir tanto una resolución como una velocidad aceptable para una gran variedad de aplicaciones. Normalmente se trata de redes resistivas conectadas a los bits de entrada, con cada valor de resistencia ajustado al valor del bit de entrada, como estructura básica.

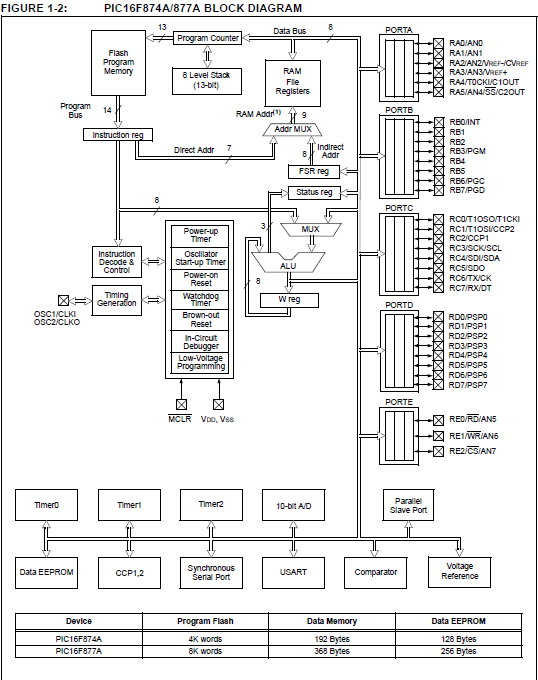
Los conversores se han enfrentado siempre a la dualidad velocidad y resolución, las diversas estructuras desarrolladas y disponibles comercialmente permiten adaptar un modelo para cada aplicación. Las configuraciones más frecuentes, atendiendo a criterios de velocidad, son: conversores lentos (de 1 a 100ms), que incluyen dispositivos de rampa y de escalera; los conversores medios (de 1μs a 1ms) abarcan los denominados aproximaciones sucesivas.

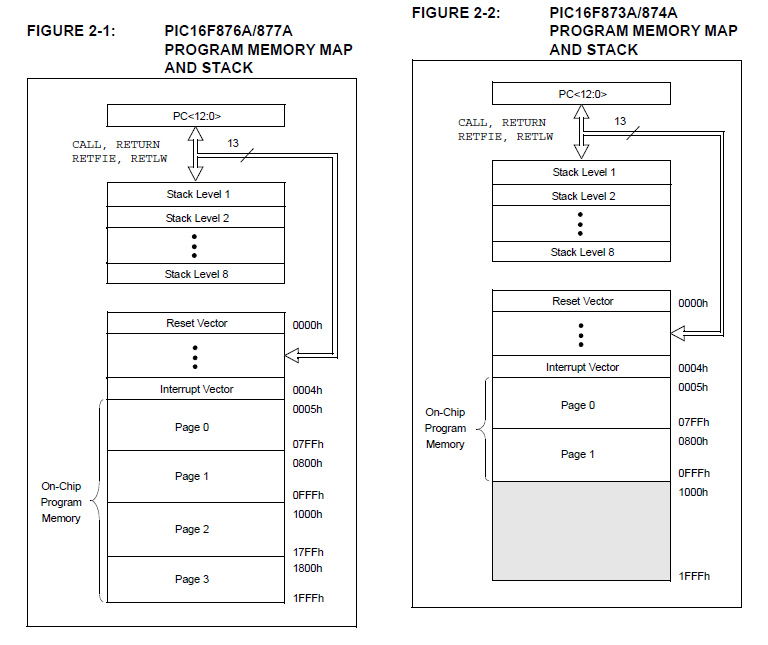
El microcontrolador PIC16F877 de Microchip tiene la capacidad de realizar diferentes funciones, entre ellas la conversión analógico-digital. Los sonidos que rodean nuestro medio son de tipo continuo al cual se le pueden tomar infinitos valores a lo largo del tiempo.

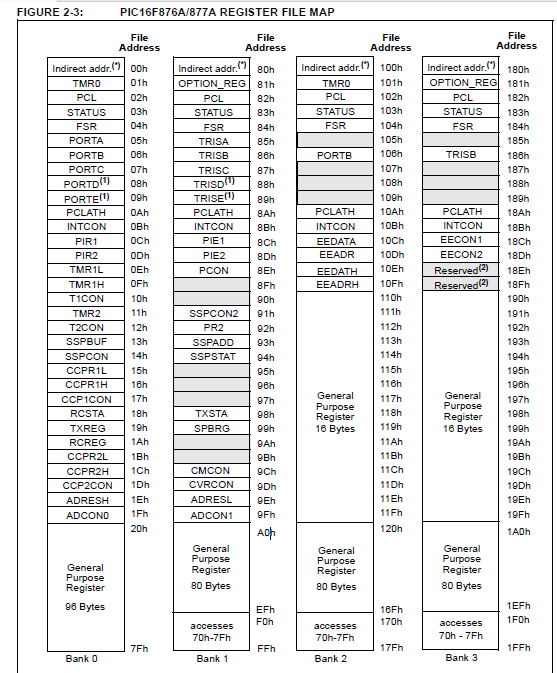
El módulo de A/D tiene cuatro registros.

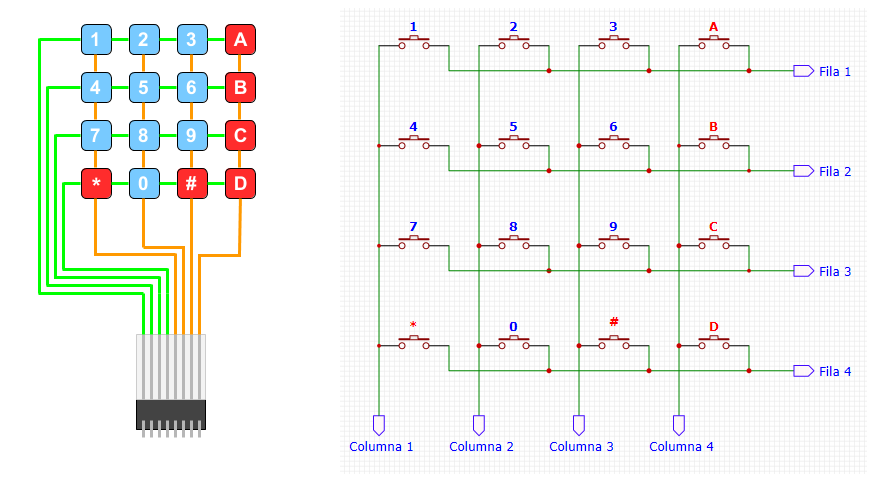


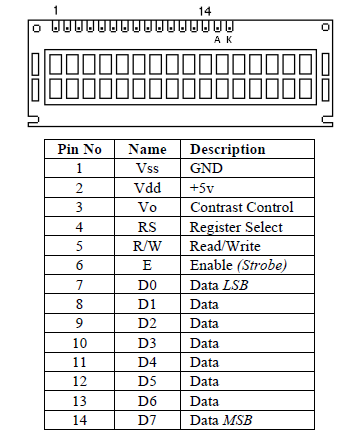


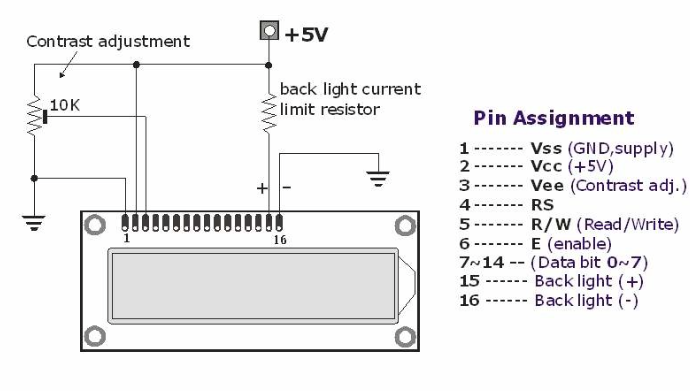










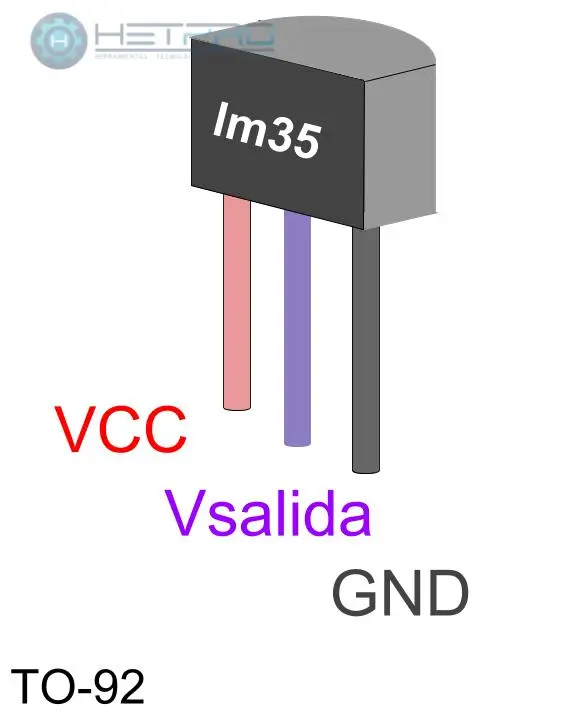


LM35

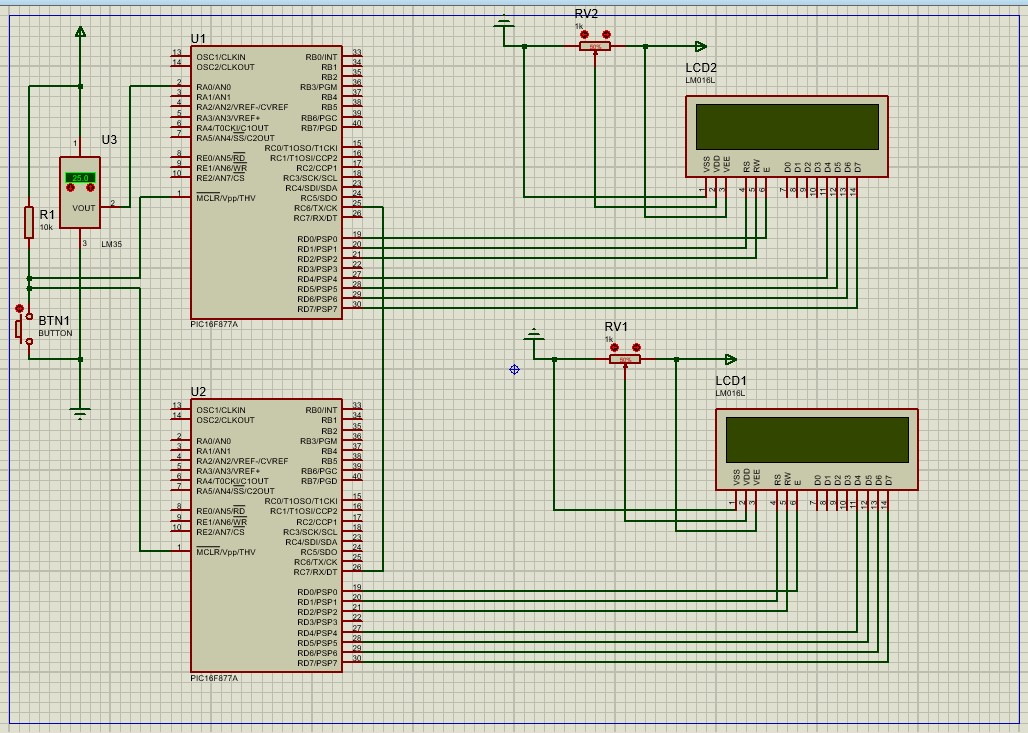
El LM35 es un circuito electrónico sensor que puede medir temperatura. Su salida es analógica, es decir, te proporciona un voltaje proporcional a la temperatura. Puede medir temperaturas desde los -55 hasta los +150 grados centígrados, por lo que cuenta con aplicabilidad para distintos ambientes, y gracias a sus dimensiones puede ser instalado en casi cualquier lugar. Su salida es analógica, pero por medio de simples ajustes en un Arduino es posible interpretar las variaciones de voltaje de salida en una pantalla digital con mucha precisión y poca cantidad de dinero.

LM35 y sus características principales

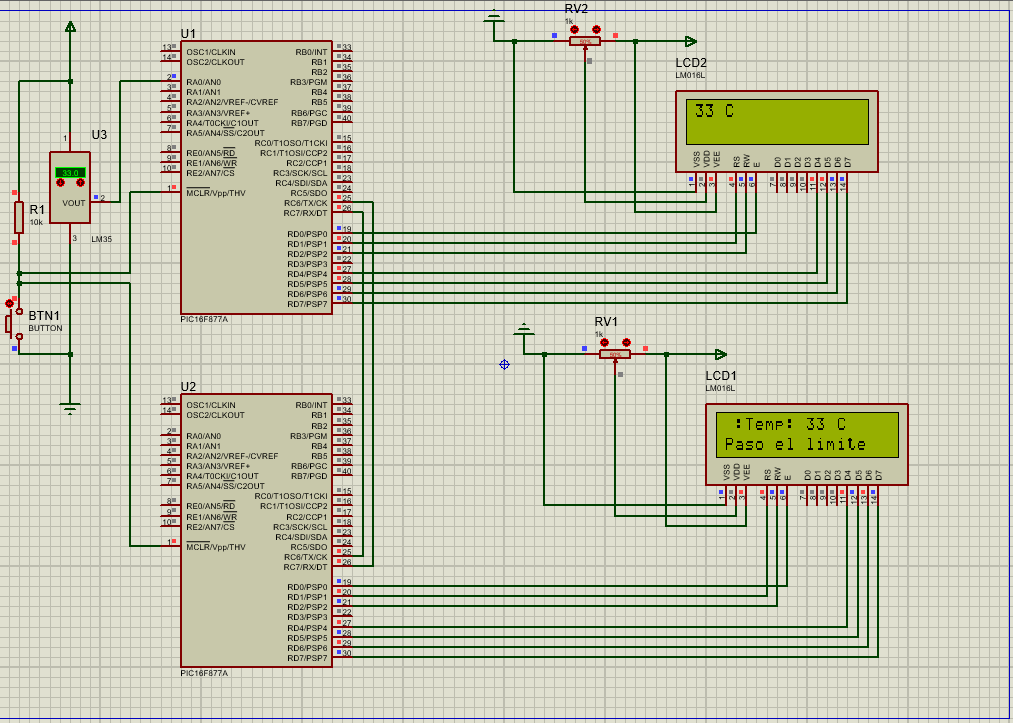
* Resolución: 10mV por cada grado centígrado.
* Voltaje de alimentación. Por ejemplo, esté sensor se puede alimentar desde 4Vdc hasta 20Vdc.
* Tipo de medición. Salida analógica.
* Numero de pines: 3 pines, GND, VCC y VSalida.
* No requiere calibración



**5.- Diagrama eléctrico o circuito.**



**6.- Captura de pantalla del circuito funcionando en el simulador.**



**7.- Código del programa.**

PIC1

#include <16F877A.h>

#fuses NOWDT, HS, NOLVP

#device ADC = 10

#use delay(clock = 4M)

#byte TRISA = 0x85

#byte PORTA = 0x05

#byte TRISC = 0x87

#byte PORTC = 0x07

#byte TRISD = 0x88

#byte PORTD = 0x08

#use RS232(BAUD = 62500, XMIT = PIN\_C6, RCV = PIN\_C7, bits = 8) //baudios 62500 bits por seg, Tx= c6 Rx = C7

#include <lcd.c>

#include <string.h>

//#use fast\_io(A)

//#use fast\_io(C)

//#use fast\_io(D)

void main()

{

int Celsius;

setup\_adc(ADC\_CLOCK\_INTERNAL); // ADC Module uses its internal oscillator

setup\_adc\_ports(AN0); // Configure AN0 pin as analog

set\_adc\_channel(0); // Select channel 0 (AN0)

lcd\_init(); // Initialize LCD module

lcd\_putc('\f'); // Clear LCD

while (TRUE) {

delay\_ms(1000);

lcd\_putc('\f');

Celsius = read\_adc() \* 0.489; // Read analog voltage and convert it to Kelvin (0.489 = 500/1023)

printf(lcd\_putc, "%d C", Celsius);

putc(Celsius);

}

}

PIC2

#include <16F877A.h>

#fuses NOWDT, XT

#use delay(clock = 4M)

#byte TRISC = 0x87

#byte PORTC = 0x07

#byte PORTD = 0x08

#byte TRISD = 0x88

#include <lcd.c>

#include <string.h>

#use RS232(BAUD = 62500, XMIT = PIN\_C6, RCV = PIN\_C7, bits = 8)   
//baudios 62500 bits por seg, Tx= c6 Rx = C7

int Celsius1;

#INT\_RDA

void rutinaSerial()

{

Celsius1 = getc();

}

void main()

{

lcd\_init();

enable\_interrupts(INT\_RDA); //se activa la interrupcion por recibo de datos den el puerto Rx

enable\_interrupts(GLOBAL);

while (true)

{

lcd\_putc('\f');

lcd\_gotoxy(2, 1);

printf(lcd\_putc, ":Temp: %d C", Celsius1);

lcd\_gotoxy(1, 2);

if (Celsius1 > 25)

{

printf(lcd\_putc, "Paso el limite");

}

else

printf(lcd\_putc, "Bajo el limite");

delay\_ms(100);

}

}

**8.- Explicación del circuito o del programa.**

El programa 1 manda el valor obtenido del sensor de temperatura al PIC 2 que esta concetado de manera serial y se muestra en una pantalla LCD.

**9.- Observaciones.**

Se me complico que el circuito funcionará ya que estaba usando un PIC mas antiguo.

**10.- Conclusión.**

Fue interesante hacer la practica y analizar como funciona la conexión serial entre PICS.