

PRACTICA No. 1 TERMÓMETRO DIGITAL.

Objetivo:

Diseñar y construir un termómetro digital basado en circuitería TTL.

Material:

- Convertidor Analógico a Digital ADC0804
- Oscilador LM555
- Amplificador operacional LM741
- Sensor de temperatura LM35
- 8 Diodos emisores de luz (LEDS)
- 1 resistor variable de 1 Kohms
- 1 resistor de 16 Kohms, 2 Kohms, 75 ohms, 10 Kohms
- 2 resistores de 500 Kohms
- 8 resistores de 330 ohms
- 2 capacitores 0.1 μ F, 1 μ F
- 1 capacitor de 150 pF
- Protoboard
- Fuente de poder de ± 5 volts
- Alambres de diferentes longitudes

Procedimiento:

1. El alumno armará el siguiente circuito en su protoboard y realizará el despliegue de la temperatura ambiente en los diodos emisores de luz (LEDS).
2. El alumno agregará el bloque de conversión de código binario a BCD para que el valor de la temperatura pueda ser visualizado en dos display's de siete segmentos

Diagrama a bloques



Figura 1) Diagrama a bloques del termómetro digital.

Nota: La polarización de los circuitos integrados puede no estar visible en el esquema.

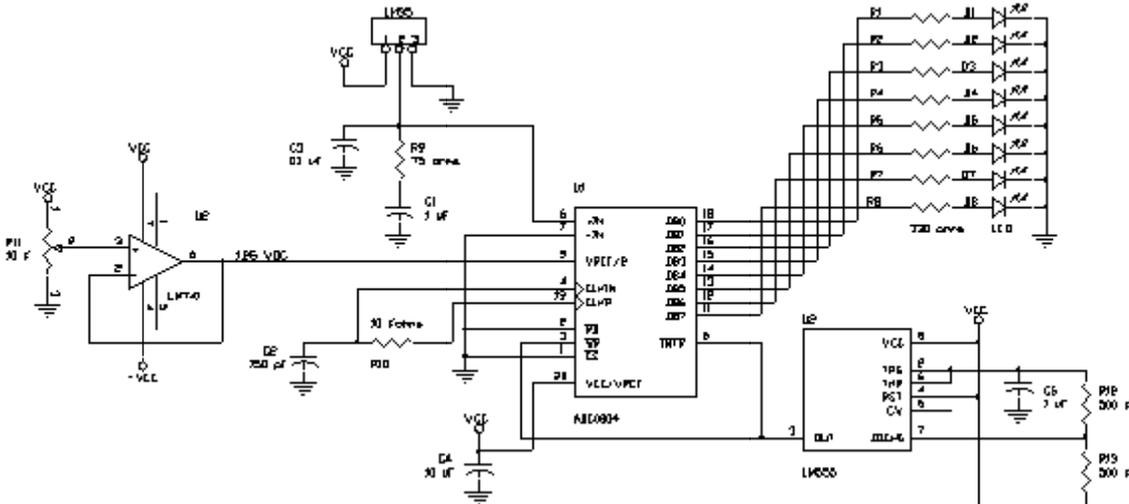


Figura 2) Diagrama esquemático (parcial) del termómetro digital.

PRACTICA No. 2 CONTROL ON/OFF DE TEMPERATURA.

Objetivo:

Diseñar y construir un circuito digital basado en circuitería TTL para controlar temperatura cuando esta exceda de su límite superior.

Material:

- Convertidor Analógico a Digital ADC0804
- Oscilador LM555
- Amplificador operacional LM741
- Amplificador operacional (comparador de precisión)
- Sensor de temperatura LM35
- Ventilador
- Relays, u optoacopladores
- TRIAC's
- 8 Diodos emisores de luz (LEDS)
- 1 resistor variable de 1 Kohms
- 1 resistor de 16 Kohms, 2 Kohms, 75 ohms, 10 Kohms
- 2 resistores de 500 Kohms
- 8 resistores de 330 ohms
- 2 capacitores 0.1 μ F, 1 μ F
- 1 capacitor de 150 pF
- Protoboard
- Fuente de poder de ± 5 volts
- Alambres de diferentes longitudes

Procedimiento:

1. El alumno armará el siguiente circuito en su protoboard y realizará el despliegue de la temperatura ambiente en los diodos emisores de luz (LEDS).
2. El alumno agregará el bloque de conversión de código binario a BCD para que el valor de la temperatura pueda ser visualizado en display de siete segmentos
3. El alumno agregará el bloque de comparación (analógica o digital) y etapa de potencia para habilitar el dispositivo eléctrico de enfriamiento.

Diagrama a bloques



Figura 1) Diagrama a bloques del termómetro digital.

Nota: Al diagrama a bloques le hace falta agregar los bloques de comparación (analógica o digital), etapa de potencia y carga (actuador).

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #3

Nombre: Sistema mínimo basado en microcontrolador 16F84A.

Objetivo: Implementar el circuito mínimo basado en microcontrolador 16F84 para el desarrollo de varias aplicaciones didácticas. Además deberá leer un dato de 5 bits por el puerto A y enviarlo a través del puerto B.

Material:

- 1 microcontrolador 16F84
- 1 XTAL de 4MHz
- 1 R de 10K Ω
- 10 R de 470 Ω o 330 Ω
- 2 C de 22pF
- 1 C de 100 pF
- 1 push button
- 8 Led's
- 1 dip swtich de 8 líneas
- 1 Fuente de voltaje 5 Volts
- 1 Programador de microcontroladores
- 1 software MPLAB

PROCEDIMIENTO

Armar el siguiente circuito, además interconectar un dip switch en puerto A, así como 8 LED's en el puerto B con sus respectivas resistencias limitadoras de corriente.

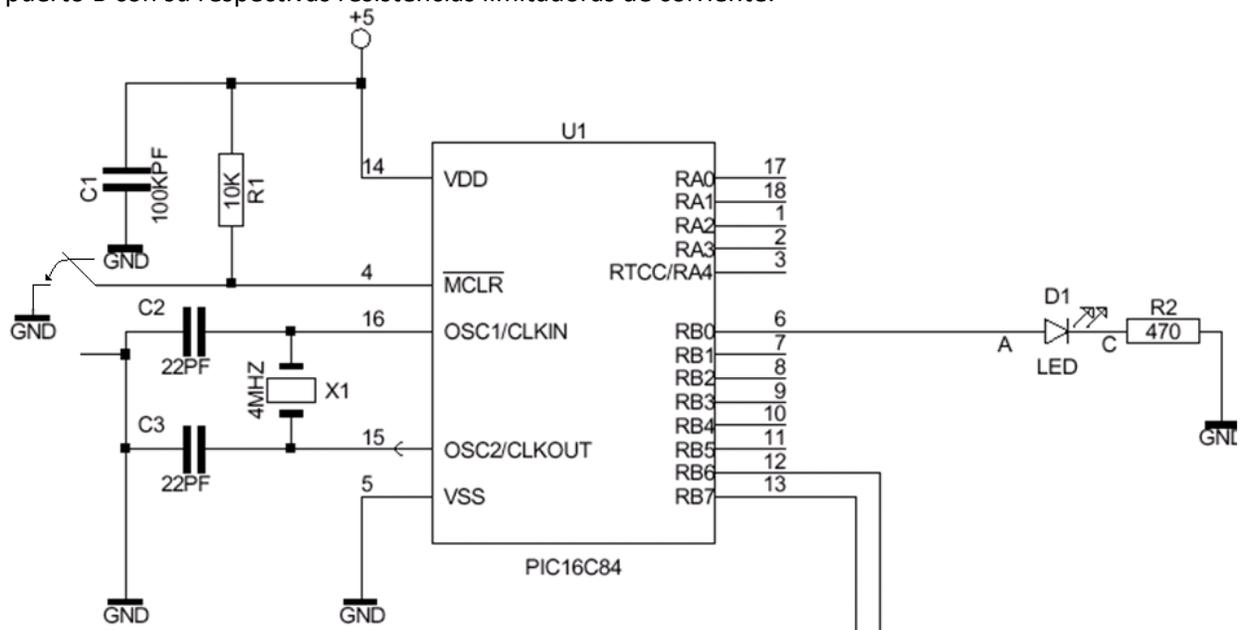


Figura 1) Diagrama eléctrico del sistema mínimo basado en microcontrolador 16F84A.

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #4

Nombre: Operaciones aritméticas con el microcontrolador 16F84A.

Objetivo: Desarrollar dos programas para el microcontrolador PIC 16F84A para que realice las siguientes operaciones aritméticas:

$$PB = (4PA + 8) / 2$$
$$PA = \sqrt{PB}$$

Donde, PA = Port A, PB = Port B del microcontrolador PIC 16F84A.

Material:

- 1 microcontrolador 16F84A
- 1 XTAL de 4MHz
- 1 R de 10KΩ
- 10 R de 470Ω o 330Ω
- 2 C de 22pF
- 1 C de 100 pF
- 1 push button
- 8 Led's
- 1 dip switch de 8 líneas
- 1 Fuente de voltaje 5 Volts
- 1 Programador de microcontroladores
- 1 software MPLAB v7.40 o superior

Diagrama a bloques

Paso # 1

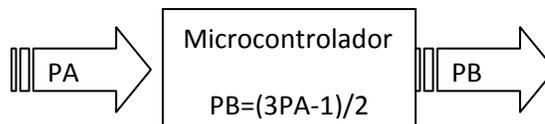


Figura 1) Diagrama a bloques del sistema mínimo para realizar la operación aritmética $PB = (3PA - 1) / 2$.

PASO #2

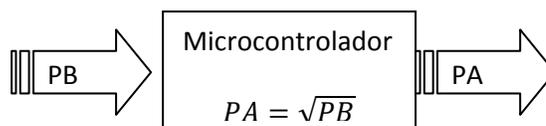


Figura 2) Diagrama a bloques del sistema mínimo para realizar la operación aritmética $PA = \sqrt{PB}$.

Nota: En las entradas de datos conectar un dip switch y en la salida LED's.

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #5

Nombre: Comparación de registros con el microcontrolador 16F84A.

Objetivo: Desarrollar un programa para el microcontrolador PIC 16F84A, para que esté leyendo los datos de entrada a través del puerto A y realice las siguientes tareas:

- a) Cuando el dato en el puerto A sea mayor que 27, únicamente deberá activar en ALTO la salida RB0.
- b) Cuando el dato en el puerto A, sea igual 27, únicamente deberá activar en ALTO la salida RB1.
- c) Cuando el dato en el puerto A, sea menor que 27, únicamente deberá activar en ALTO la salida RB2.

Nota: Observe que solamente deberá estar una salida activa en ALTO, dependiendo del resultado de la comparación.

Material:

- 1 Microcontrolador 16F84A
- 1 XTAL de 4MHz
- 1 R de 10KΩ
- 10 R de 470Ω o 330Ω
- 2 C de 22pF
- 1 C de 100 pF
- 1 push button
- 8 Led's
- 1 dip switch de 8 líneas
- 1 Fuente de voltaje 5 Volts
- 1 Programador de microcontroladores
- 1 software MPLAB

Diagrama a bloques



Figura 1) Diagrama a bloques del sistema mínimo para realizar comparaciones.

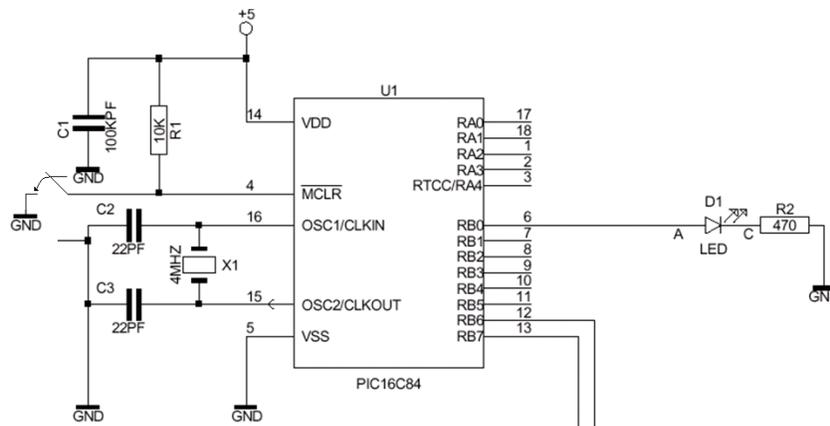


Figura 2) Diagrama eléctrico del sistema mínimo basado en microcontrolador 16F84A.

Elaboró: M.C. Everardo Inzunza González

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #6

Nombre: Generación de retardos y secuencias con el microcontrolador 16F84A.

Objetivo: Desarrolle un programa para que encienda 8 motores de CD en forma secuencial con un retardo entre secuencias de medio segundo.

El sistema deberá tener un botón de arranque para que inicie el encendido secuencial y un botón de Paro de Emergencia para apagar inmediatamente todos los motores. Considere que el botón de arranque está conectado en el pin RA0 y el de Paro en el pin RA1. Los motores deberán conectarse en el puerto B por medio de una etapa de potencia.

Material:

- 1 Microcontrolador 16F84
- 1 XTAL de 4MHz
- 1 R de 10K Ω
- 10 R de 470 Ω o 330 Ω
- 2 C de 22pF
- 1 C de 100 pF
- 2 push button
- 8 Led's
- 1 dip switch de 8 líneas
- 1 Fuente de voltaje 5 Volts
- 1 Programador de microcontroladores
- 1 software MPLAB
- 8 motores de CD

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #7

Nombre: DISEÑO DE UN SEMÁFORO DIGITAL

Objetivo: Desarrollar un sistema basado en microcontrolador PIC 16F84A para controlar un semáforo de cruce de una calle principal con una avenida y que considere el cruce de peatones.

Requerimientos generales del sistema

Se requiere diseñar un semáforo digital para controlar el tráfico vehicular en la intersección de una calle de tráfico muy denso con una calle de tráfico moderado.

La calle principal deberá tener una luz verde encendida durante un tiempo de 30 segundos. Antes de agotarse el tiempo de 30 segundos, la luz verde deberá parpadear 3 veces y posteriormente apagarse. La calle lateral debe tener la luz verde encendida durante 20 segundos, de igual manera antes de agotarse los 20 segundos la luz verde deberá parpadear 3 veces. La luz ámbar de precaución de ambos semáforos tiene que durar 5 segundos y es activada cuando se apaga la luz verde de cada semáforo. La luz roja de ambos semáforos es encendida una vez agotado el tiempo de la luz ámbar de cada semáforo. **Cuando se detecte la activación del botón de cruce peatonal**, se deberá esperar que termine el tiempo verde y ámbar de la calle principal o avenida y posteriormente activar la luz de cruce peatonal (activar luz roja de calle principal y avenida) durante 30 segundos y la alerta sonora para personas invidentes por el mismo tiempo (30 seg.).

Favor de construir una base para cada semáforo.

Material:

- 1 Microcontrolador 16F84
- 1 Relevadores u optoacopladores y TRIAC's
- 2 focos verdes, 2 focos ámbar y 2 focos rojos de 127 Vca @ 60Watts.
- 1 XTAL de 4MHz
- 1 R de 10K Ω
- 6 R de 470 Ω o 330 Ω
- 2 C de 22pF
- 1 C de 100 pF
- 2 push button
- 1 Fuente de voltaje 5 Volts
- 1 Programador de microcontroladores
- 1 software MPLAB

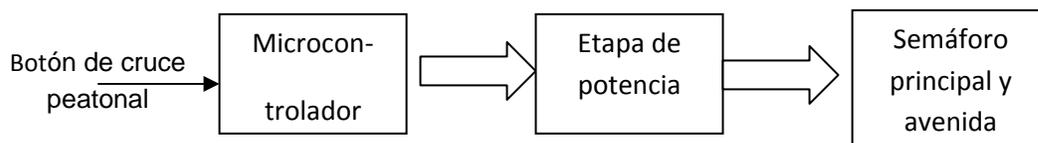


Figura 1) Diagrama a bloques del semáforo digital basado en microcontrolador PIC 16F84A.

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #8

Nombre: ALARMA DIGITAL

Objetivo: Diseñar y construir una alarma digital utilizando el microcontrolador 16F84A.

Requerimientos generales del sistema

El sistema tendrá como entrada 4 sensores del tipo ON/OFF y una entrada para ACTIVAR/DESACTIVAR la alarma. Estas 5 entradas serán a través del puerto A y son activas en ALTO. Cuando la alarma esté activada y cualquiera de los sensores se active, la alarma deberá encender los bits 0, 1 y 2 del puerto B, los cuales encenderán una bocina (de 12 Vcd o 24Vcd o 17vca), una luz de emergencia (12vcd o 127 vca) y un indicador led.

Material:

- 1 Microcontrolador 16F84
- 1 Luz de emergencia (torreta)
- 1 Bocina de alarma (12V)
- 4 sensores para alarma.
- Dispositivos de potencia (Relays, optoacopladores, TRIAC's)
- 1 XTAL de 4MHz
- 1 R de 10K Ω
- 1 LED
- 2 C de 22pF
- 1 C de 100 pF
- 2 push button
- 1 Fuente de voltaje 5 Volts
- 1 Programador de microcontroladores
- 1 software MPLAB

ALARMA DIGITAL

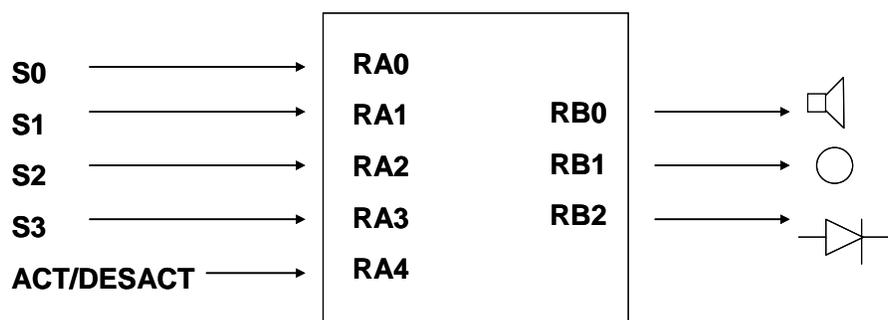


Figura 1) Diagrama a bloques de la alarma digital.

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #9

Nombre: CONTROL DE UNA BANDA TRANSPORTADORA

Objetivo: Desarrollar un programa para el microcontrolador 16F84A de tal forma que controle una Banda Transportadora de recipientes, para que sean llenados automáticamente con agua. El sistema deberá tener un botón de arranque, para que encienda al motor y la banda comience a girar, cuando se detecte que llegue el primer recipiente, la banda deberá detenerse y se encenderá una bomba hidráulica para que comience a llenar con agua el recipiente. Una vez llenado este recipiente, se apaga la bomba y se enciende el motor para que gire la banda y acerque el siguiente recipiente a ser llenado. El sistema deberá llevar el conteo de cuantos recipientes se han llenado en una jornada de trabajo, el despliegue consiste en 3 display's de siete segmentos o de un LCD. En caso de algún accidente, considerar un botón de paro de emergencia.

Material:

- 1 Microcontrolador 16F84
- 1 XTAL de 4MHz
- 1 R de 10KΩ
- 2 C de 22pF
- 1 C de 100 pF
- 2 push button
- 3 Display's de 7 segmentos (ánodo común o cátodo común) o un LCD 2X16
- 1 motor de CD
- 1 Bomba Hidráulica (puede ser de pecera ó wiper's)
- 2 relays
- 2 transistores de potencia
- 1 Fuente de voltaje para el motor y la bomba
- 1 Láser, LED infrarrojo y un foto detector.
- 1 Fuente de voltaje 5 Volts
- 1 Programador de microcontroladores
- 1 software MPLAB

Diagrama a bloques

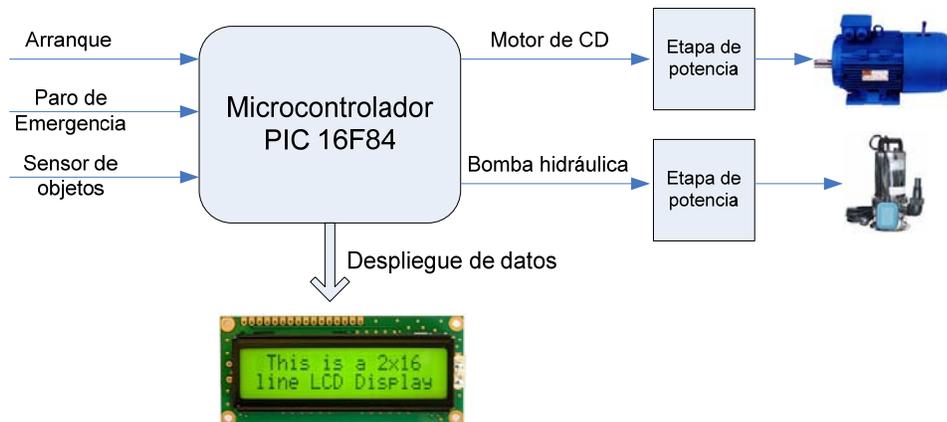


Figura 1) Diagrama a bloques del sistema automático de llenado de recipientes.

Elaboró: M.C. Everardo Inzunza González

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #10

Nombre: CONFIGURACION Y USO DEL CONVERTIDOR DE A/D DEL MICROCONTROLADOR 16F877.

Objetivo: Configurar y utilizar el convertidor analógico a digital del microcontrolador 16F877

Requerimientos generales del sistema

El circuito tendrá las siguientes características: entrada por el canal AN0, se utilizará como voltaje de referencia el interno (Vdd), resolución de 10 bits, frecuencia de muestreo $f_{osc}/32$, los datos justificados a derecha para que sean enviados a través de los puertos B y C.

Material:

- 1 Microcontrolador 16F877
- 1 potenciómetro de precisión de 5 KOhms o 10 KOhms.
- 1 XTAL de 4MHz
- 1 R de 10K Ω
- 10 LED's
- 10 R de 330 Ohms
- 2 C de 22pF
- 1 C de 100 pF
- 1 push buttom
- 1 Fuente de voltaje 5 Volts
- 1 Programador de microcontroladores
- 1 software MPLAB

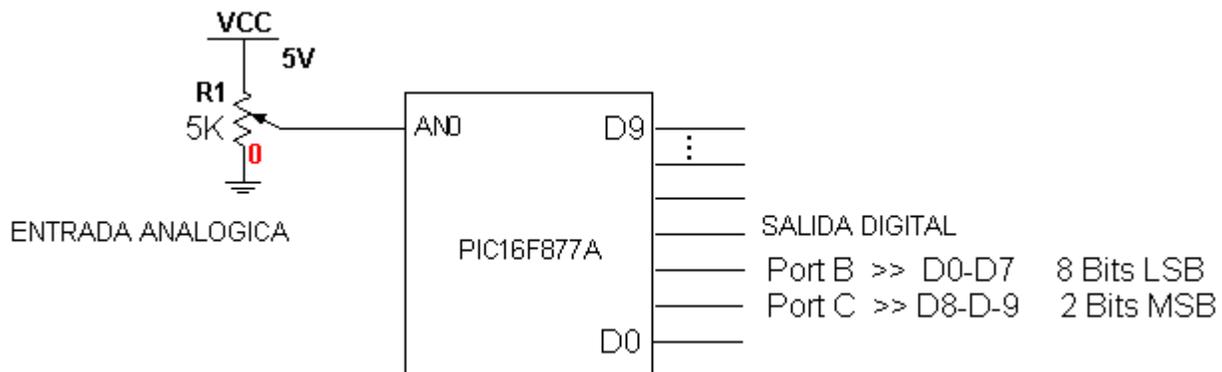


Figura 1) Diagrama a bloques del convertidor analógico a digital basado en el microcontrolador 16F877A.

**UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #11**

Nombre: TERMOMETRO DIGITAL BASADO EN MICROCONTROLADOR

Objetivo: Diseñe y construya un termómetro digital basado en microcontrolador de la familia 16F87X.

Requerimientos generales del sistema

El termómetro deberá medir la temperatura desde 0°C hasta 150°. Deberá desplegar en un display de cristal líquido (LCD) la siguiente información: UABC-INGENIERIA, La temp= xxx °C.

Material:

- 1 Microcontrolador 16F877
- 1 potenciómetro de precisión de 5 KOhms o 10 KOhms.
- 1 XTAL de 4MHz
- 1 R de 10KΩ
- 10 LED's
- 10 R de 330 Ohms
- 2 C de 22pF
- 1 C de 100 pF
- 1 push button
- 1 Fuente de voltaje 5 Volts
- 1 Sensor de temperatura
- 1 op-amp
- 1 LCD de 2X16.
- 1 Programador de microcontroladores
- 1 software MPLAB

PROCEDIMIENTO

Continúe con el circuito de la práctica del ADC con microcontrolador, implemente la conversión de código binario a código ASCII con el ensamblador y la subrutina de despliegue para el LCD, interconecte el sensor y el acondicionamiento de señal al microcontrolador, interconecte el LCD. Realice las pruebas y haga una calibración del instrumento para que la medición sea más exacta.

**UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERIA ENSENADA
CIRCUITOS DIGITALES III
PRACTICA #12**

Nombre: COMUNICACIÓN SERIAL CON EL MICROCONTROLADOR

Objetivo: Diseñe y construya un termómetro digital basado en microcontrolador de la familia 16F87 con comunicación RS-232 hacia una computadora personal.

Requerimientos generales del sistema

El termómetro de deberá medir la temperatura desde 0°C hasta 150°. Deberá desplegar en un display de cristal líquido (LCD) la siguiente información: UABC-INGENIERIA, La temp= xxx °C. El dato binario deberá ser enviado a una PC por medio del puerto USART del microcontrolador hacia una PC. La PC deberá desplegar el contenido del dato en código binario, sistema decimal y en gráfica de barra de la temperatura.

Material:

- 1 Microcontrolador 16F877
- 1 IC Max232.
- 1 XTAL de 4MHz
- 1 R de 10KΩ
- 10 LED's
- 10 R de 330 Ohms
- 2 C de 22pF
- 1 C de 100 pF
- 1 push button
- 1 Cable serial null modem con conector DB-9 (RS-232)
- 1 Fuente de voltaje 5 Volts
- 1 Sensor de temperatura
- 1 op-amp
- 1 LCD de 2X16.
- 1 Programador de microcontroladores
- 1 software MPLAB

INTRODUCCIÓN:

Cuando se ha logrado implementar cualquier protocolo de comunicación entre una PC y un microcontrolador se puede decir que una gran cantidad de aplicaciones están a la puerta, entre las cuales se pueden nombrar monitoreo, control digital, impresión de datos en papel, impresión en pantalla de datos inmediatos, etc. Los protocolos más comúnmente implementados en microcontroladores y PC son: RS232, IP, Paralelo, etc.

El microcontrolador PIC16F877A tiene incluido dentro de su estructura interna un modulo USART (Transmisor receptor universal sincrono asíncrono) que es la abreviación de “universal synchronous asynchronous receiver transmitter” en inglés también es conocido como SCI (“serial communications interface”) y es uno de los mas comúnmente usados para la implementación de puertos seriales, la configuración asíncrona full dúplex es una de las mas populares de la USART pues es usada para interfaces con el puerto serie de las computadoras personales o PC usando el protocolo RS-232.

La función principal del modulo USART es enviar y recibir datos de forma serial, esta operación puede ser dividida en dos categorías transmisión síncrona y transmisión asíncrona. La operación síncrona usa una línea de datos y una de reloj mientras que la operación asíncrona solo usa la línea de datos. Esta es la principal diferencia entre estos dos modos de operación.

Modo Asíncrono:

La comunicación asíncrona es la que generalmente se utiliza para la comunicación entre un microcontrolador y una PC. Para la comunicación serie con la PC se utiliza el estándar “no retorno a cero” (NRZ) en la forma conocida como 8-N-1 que quiere decir 8 bits de datos, sin paridad y un bit de parada tal como se muestra en la figura 1.

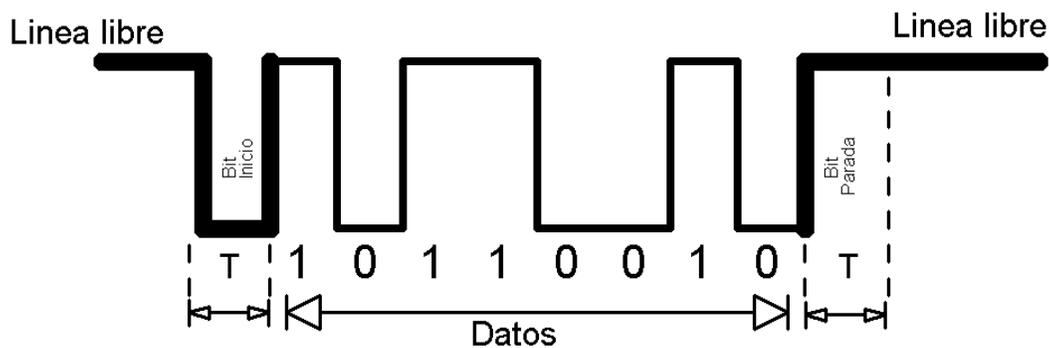


Figura 1. Cronograma del protocolo RS-232.

Línea libre se define como un estado lógico alto o uno. El inicio de la transmisión de datos (Bit de inicio) se define como un estado lógico bajo o cero. Los bits de datos son enviados enseguida del Bit

de inicio, enviando en primer lugar el Bit menos significativo. Después de los Bits de datos se envía el Bit de parada que es un estado lógico alto o uno. El periodo T depende de la velocidad de transmisión y es ajustado de acuerdo a las necesidades de transmisión. Para la velocidad de 9600 baudios, T es de 104 uS. En la figura 2 se muestra un conector RS-232 y en la tabla 1 muestra los nombres de cada pin.

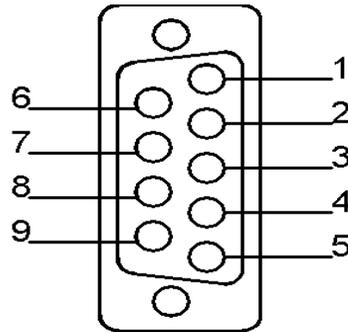


Figura 2. Conector DB9 estándar.

Tabla 3. Nombres de las patas del conector DB9.

Pin	Nombre	Descripción
1	CD	Acarreo detectado
2	RXD	Recepción de datos
3	TXD	Transmisión de datos
4	DTR	Terminal de datos lista
5	GND	Tierra
6	DSR	Grupo de datos listo
7	RTS	Petición de envío
8	CTS	Libre para enviar
9	RI	Indicador de Ring

Los pasos para la configuración del modulo USART como un puerto asíncrono serie, con la configuración no retorno a cero en la forma 8-N-1 son los siguientes:

Para la transmisión de datos

1. Inicializar el Registro generador de Baud Rate (SPBRG) con el valor apropiado si se piensa usar una velocidad alta el Bit BGRH debe ser puesto a uno.
2. Se habilita el puerto serie asíncrono limpiado o poniendo a cero el Bit (SYNC) y se pone a uno el Bit SPEN (Serial port enable)
3. Cargar el dato en el registro TXREG (después de esto se inicializa la transmisión)
4. Habilitar la transmisión poniendo a uno el Bit TXE

Para la recepción de datos

1. Inicializar el Registro generador de Baud Rate (SPBRG) con el valor apropiado si se piensa usar una velocidad alta el bit BGRH debe ser puesto a uno.
2. Se habilita el puerto asíncrono serie limpiado o poniendo a cero el Bit (SYNC) y se pone a uno el bit SPEN (Serial port enable)
3. Habilitar la transmisión poniendo a uno el Bit CREN
4. Leer el registro RCREG para conocer los datos recibidos

Para conocer el valor que se requiere introducir en el registro SPBRG para una determinada velocidad de transferencia asíncrona se usan la siguiente fórmula:

$$SPBRG = \frac{frec_clock_entrada}{64 * Velocidad_deseada} - 1$$

El valor obtenido por esta fórmula se redondea al entero inferior más cercano y se introduce en el registro SPBRG para fijar la velocidad de transferencia deseada. En la tabla 4 se muestran algunos valores para SPBRG para diferentes velocidades y un cristal externo de 4 MHz

Tabla 2. Valores para fijar diferentes velocidades de transferencia.

Baud Rate (K)	Kbaud	%Error	Valor de SPBRG (Decimal)
0.3	0.300	0	207
1.2	1.202	0.17	51
2.4	2.404	0.17	25
9.6	8.929	6.99	6
19.2	20.833	8.51	2
28.8	31.250	8.51	1
33.6	-	-	-
57.6	62.500	8.51	0

La interfase física que se utiliza comúnmente para conectar un microcontrolador con la PC se muestra en la figura 6. El circuito MAX232 realiza la tarea de generar de acuerdo a los valores lógicos entregados por el microcontrolador los niveles necesarios de voltaje para la comunicación RS-232.

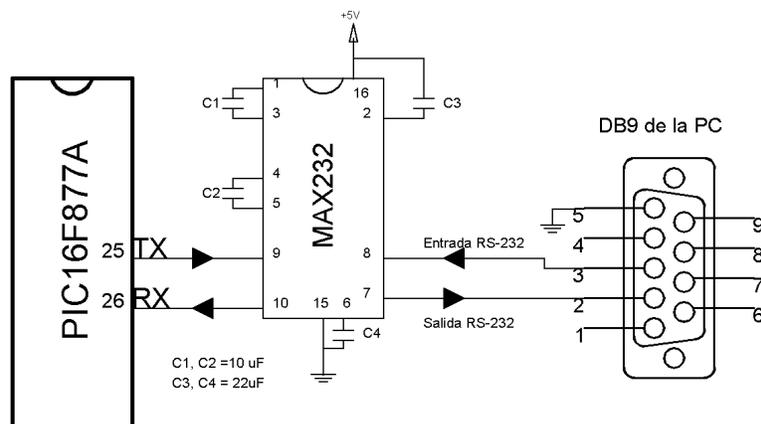


Figura 3. Conexión física para la comunicación RS-232.

DESARROLLO DE LA COMUNICACIÓN RS-232:

Para implementar la comunicación serie entre una PC y un microcontrolador en la etapa experimental se suele utilizar el programa HyperTerminal figura 4.



Figura 4. Pantalla de inicio del Programa HyperTerminal.

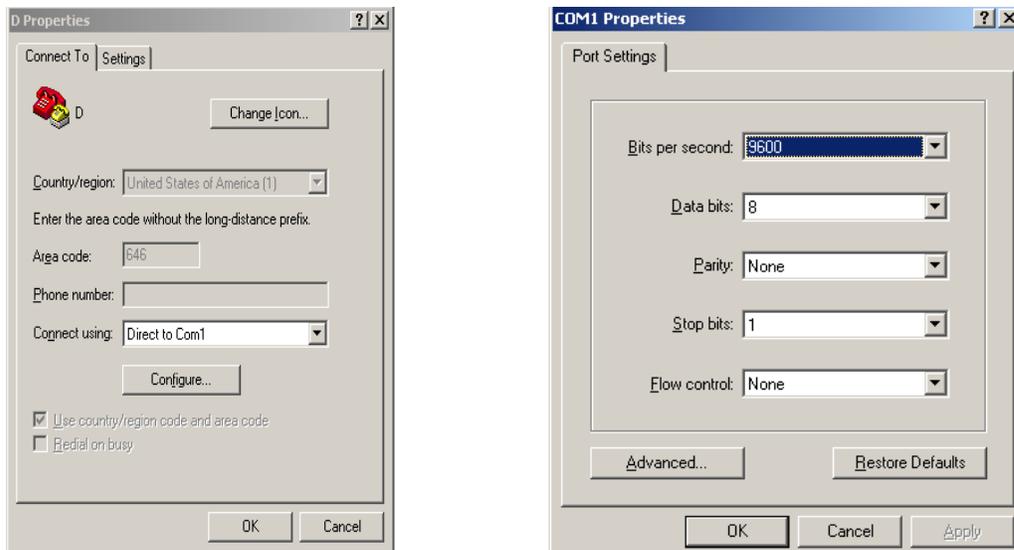
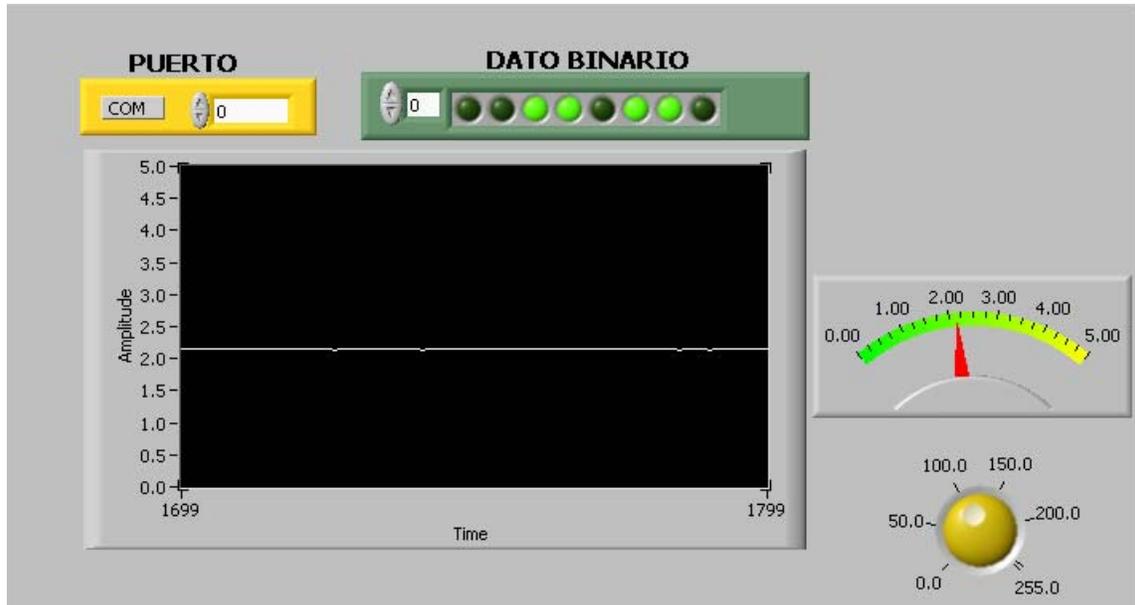


Figura 5. Configuración y elección del puerto disponible a utilizar.

Una vez que se ha establecido la comunicación entre HyperTerminal y el microcontrolador se puede pasar a programar la aplicación que se desee en el paquete que se desee MATLAB, LABVIEW, VISUAL BASIC, etc. En la presente práctica se optó por LABVIEW, del cual se presenta pantalla principal en la figura 6.



PROCEDIMIENTO

- Continúe con el circuito de la práctica del ADC con microcontrolador.
- Implemente la conversión de código binario a código ASCII con el ensamblador y la subrutina de despliegue para el LCD.
- Interconecte el sensor y el acondicionamiento de señal al microcontrolador.
- Interconecte el LCD.
- Escribir la subrutina de código para transmitir los datos por el puerto USART del microcontrolador.
- Interconectar el circuito MAXIM 232 al microcontrolador para convertir los niveles de voltaje TTL a RS-232.
- Realice el programa de la PC para la captura y graficado de datos en LABVIEW.
- Realice las pruebas y haga una calibración del instrumento para que la medición sea más exacta.

REFERENCIAS:

- [1] Anónimo, "*PICmicro Mid-Range MCU Family Reference Manual*", Microchip rev/DS33023A, December 1997.
- [2]. Anónimo, "*PIC16F87X Data sheet*" Microchip rev/30292C, 2001.
- [3]. Anónimo, "*MAX232 Data sheet*", Texas Instruments, 2003.