

UNIVERSIDAD MAYOR DE SAN ANDRES

FACULTAD TECNICA

CARRERA: ELECTRÓNICA Y TELECOMUNICACIONES



NIVEL LICENCIATURA

EXAMEN DE GRADO

TRABAJO DE APLICACIÓN

**“SISTEMA REMOTO DE CONTROL DE TEMPERATURA
APLICADO A UN INVERNADERO CON EL PIC16F877A”**

Postulante: Soledad Sonia Choque Jaillita

La Paz – Bolivia

2012

DEDICATORIA

A mi familia

Dedico este trabajo de aplicación a toda mi familia, especialmente a mi esposo Rudy por sus valiosos intercambios de opinión y consejos, como también a mi hijo Ian Ethan que me ha dado la fuerza para seguir adelante.

También para mi madre ya que gracias a su esfuerzo pude seguir mis estudios en la Universidad.

AGRADECIMIENTOS

A mis Docentes

Quisiera reflejar en estas líneas mi agradecimiento a todos mis docentes de la Universidad Mayor de San Andrés, de la Facultad Técnica por las enseñanzas durante todo el camino de mi carrera profesional .

INDICE GENERAL

DEDICATORIA

AGRADECIMIENTOS

CAPITULO I

INTRODUCCION

1.1.- RESUMEN	1
1.2.- PLANTEAMIENTO DEL PROBLEMA	3
1.3.- JUSTIFICACION.....	4
1.4.- OBJETIVOS	4
1.4.1. - OBJETIVO GENERAL.....	4
1.4.2. - OBJETIVOS ESPECÍFICOS	4

CAPITULO II

MARCO TEORICO

2.1.-CARACTERISTICAS DE INVERNADEROS	5
2.1.1 Ventajas del empleo de invernaderos	5
2.2. – CARACTERISTICAS DE LOS CULTIVOS	6
2.2.1. - VENTILACION	6
2.2.2. - TEMPERATURA.....	6
2.3. – MICROONTROLADOR PIC16F877A.....	7
2.3.1. – CARACTERISTICAS GENERALES DEL PIC 16F877A.....	7
2.3.2. - OSCILADOR	9
2.3.3. – DIAGRAMA DE BLOQUES PIC 16F877A	9
2.3.4. – ORGANIZACIÓN DE LA MEMORIA.....	10
2.4. – SENSOR DE TEMPERATURA LM35.....	10
2.7.1. – CARACTERISTICAS	10
2.4.2. - ENCAPSULADO.....	12
2.4.3. – DATOS TECNICOS DEL LM35	12

2.4.4. – CARACTERISTICAS TIPICAS	13
2.5. – DISPLAY LIQUIDO LCD.....	13
2.5.1. – CARACTERISTICAS	14
2.5.2. – LA MEMORIA DEL LCD.....	14
2.5.2.1. –DDRAM (Display Data RAM).....	14
2.5.2.2. – LA CG RAM (Character Generator RAM).....	16
2.5.3. – INTERFAZ HARDWARE	17
2.5.3.1. – Interfaz del display.....	18
2.5.4. – COMANDOS DEL LCD.....	18
2.5.2.1. – Resumen de comandos.....	19
2.5.5. –LCD CON EL COMPILADOR CCS	20
2.6. – EL RELE	20
2.6.1. – ESTRUCTURA	21
2.6.2. – FUNCIONAMIENTO	21
2.6.3. – CARACTERISTICAS	21
2.6.3.1. – Parte electromagnética.....	21
2.6.3.2. – Contactos o parte mecánicas.....	22
2.6.4. – TIPOS DE RELES.....	22
2.6.4.1. – Relé de armadura	22
2.6.4.1. – Relé de núcleo móvil	23
2.6.4.1. – Relé tipo Reed o de Lengüeta.....	23
2.6.4.1. – Relés polarizados	24
2.6.4.1. – Relé de estado sólido SSR	24
2.7. – COOLERS.....	25
2.7.1. – PARTES CONSTITUTIVAS DE UN EQUIPO EÓLICO.....	26
2.8. – TRANSISTORES.....	27
2.8.1. – BC548	27
2.8.1.1. - Descripción.....	27
2.8.1.2 – Principales características	27
2.8.2. – BC547	27
2.8.2.1. - Características.....	28
2.8.3. – BC559	28
2.8.3.1. – Características generales.....	29

2.8.3.2. - Tensión de ruptura Colector-Emisor	29
2.8.3.3 – Ganancia de corriente Estática y Dinámica	29
2.9. – BOMBILLAS (FOCOS)	30
2.10. – CRISTAL DE CUARZO	31
2.10.1. – CARACTERISTICAS DEL CUARZO	31
2.10.2. – CARACTERISTICAS DEL OSCILADOR DE CRISTAL	32
2.11. – NORMA RS232.....	34
2.11.1. – VELOCIDAD DE TRANSMISION	34
2.11.2. – LINEAS O CANALES DE COMUNICACION	35
2.11.2.1. – Simplex	35
2.11.2.2. – Semi duplex	35
2.11.2.3. – Full duplex	35
2.11.3. – MODOS DE TRANSMISION	36
2.12. – CONECTOR DB9	36
2.12.1. – DESCRIPCION DEL CONECTOR DB9	36
2.13. – CIRCUITO INTEGRADO MAX232	38
2.13.1. – FUNCIONAMIENTO	38
2.13.2. – USOS	39
2.13.3. – CARACTERÍSTICAS A +5V, CONDENSADORES DE 100 nF	40
2.13.4. – CONEXIÓN DE UN μ C AL PUERTO SERIE DEL PC	40
2.13.5. – CONVERTOR RS232 A TTL SIN MAX232	41
2.12. – CABLE DE CONEXION	42

CAPÍTULO III

MARCO PRÁCTICO

3.1. – HERRAMIENTAS SOFTWARE	44
3.1.1. – COMPILADOR CCS PCWH V.4068	44
3.1.1.1. – Características del lenguaje C para este compilador.....	45
3.1.1.2. – Utilidades adicionales	45
3.1.2. – VISUAL BASIC 6.0	46
3.1.2.1. – Características	47
3.1.2.2. – Entorno de desarrollo	47

3.1.3. – PROTEUS	49
3.1.3.1. – El modulo VSM.....	50
3.1.4. – VIRTUAL SERIAL PORT DRIVER	50
3.1.5. – SHOW MY PC	51
3.1.5.1. – Pasos a seguir para realizar esta conexión.....	51
3.2. – DISEÑO DEL SOFTWARE.....	52
3.2.1. – SISTEMA DE CONTROL DE TEMPERATURA	52
3.2.2. – INTERFAZ DE USUARIO	57
3.2. – DISEÑO DEL HARDWARE.....	59
3.3.1. – DISEÑO DE PLACAS	59
3.3.2. – QUEMADO DE PLACAS	60
3.3.2. – ACABADO DE PLACAS	61
3.2. – DISEÑO DE LA MAQUETA	62

CAPITULO V

4.1. – CONCLUSIONES	63
----------------------------------	-----------

BIBLIOGRAFÍA

CAPITULO I

INTRODUCCION

1.1 RESUMEN.-

El presente trabajo de aplicación tiene como objetivo principal controlar remotamente los parámetros de temperatura para un invernadero, y de esta manera comparar los parámetros de temperatura con la temperatura ambiente del invernadero, para luego controlar los diferentes dispositivos que compensarán los cambios de temperatura que se requieran dentro de éste.

Para desarrollar el control de temperatura se utilizó el microcontrolador PIC16F877A, el PIC realizará la comparación de los parámetros de temperatura y la temperatura ambiente del invernadero, y a través de este controlar los dispositivos. Para tomar la medición de temperatura se utilizó el sensor LM35, la salida es muy lineal y cada grado centígrado equivale a 10mV en la salida, no requiere de circuitos adicionales para su calibración externa, opera en un rango de temperatura desde -55 a +150 °C y debido a su baja corriente de alimentación (60uA) entre otras.

Debido a que es sistema en el cual podemos programar los parámetros de temperatura, no se realizó un cálculo exacto del rango de temperaturas que permite obtener una óptima productividad. Por ejemplo existe un rango de temperaturas óptimas diferenciado para la germinación de las semillas y para la vida de las plantas:

- Para la germinación, las temperaturas oscilan, generalmente, entre los 28 °C y los 32 °C.
- Para el normal crecimiento de las plantas, entre los 22 °C y los 28 °C.

Esto nos indica que para la germinación podremos configurar los rangos de temperatura para que se mantenga entre los 28°C y los 32°C, y para el crecimiento normal se configurará, entre los 22°C y los 28°C.

El circuito de control de temperatura contará con la posibilidad de configurar los parámetros de temperatura manualmente, y enviará los datos correspondientes a la PC a través del puerto serie.

Se desarrolló una interfaz de usuario, el cual fue programado en Visual Basic para realizar la comunicación serial entre la PC y el microcontrolador, de manera que también a través de esta interfaz se podrá programar los parámetros de temperatura para el control de dispositivos, y se podrá observar de forma gráfica el estado de los dispositivos que ya estarían instalados. De esta forma se podrá actuar sobre el sistema de una manera rápida y fácil.

Para el control remoto del sistema, se utilizará el software especializado **show my PC**, este software no requiere **ningún IP** ya que como medida de seguridad el programa asigna una **contraseña** al activar la opción “**Show My PC Now**”, esta contraseña deberá ser introducida en la otra PC que se encargará de controlar el sistema. Aunque se podría aplicar un sistema de seguridad ante los hackers, no se lo realizó debido a que encarecería el costo del trabajo de aplicación.



1.2 PLANTEAMIENTO DEL PROBLEMA.

La alimentación siempre ha sido y siempre será un recurso que el ser humano necesita y necesitará sin importar el paso del tiempo, el agotamiento de nuestros recursos naturales es un tema inquietante en la humanidad, aunque ya existen los alimentos transgénicos y otras variedades, no se compara con el alimento natural, y debido a esto se necesita realizar grandes plantaciones, pero en todos los países del mundo se está sufriendo cambios climáticos, lluvias torrenciales, granizadas de gran magnitud, tornados, cambios bruscos de temperatura de frío a calor o de calor a frío, etc.

Todos estos cambios producen que el “tiempo de cosecha”, que existe en diferentes poblaciones de nuestro país ya no sea exacto, lo cual produce la pérdida de los productos tales como la papa, el camote, etc. Y así también los productores pierden toda su inversión.

Además las cultivaciones de diferentes productos, se las hace dependiendo a la región en el que se encuentran, por ejemplo, en la altiplanicie de La Paz solo se puede plantar alimentos que pueden crecer a las temperaturas entre 0° a 28° y no a mayores temperaturas.



Fig.1 Cosechas pérdidas

1.3 JUSTIFICACION.

Con este proyecto se desea solucionar pérdidas de cultivos, que son producidos por el cambio climático por esto se desea implementar este sistema, para evitar todas estas problemáticas, respecto a la variación del tiempo, y además que se podría conseguir que los productos ya no sean exclusivamente de regiones.

Entonces con este sistema podríamos cultivar todo tipo de productos, sin poner mucho énfasis con respecto a las variaciones que existe en cada región. Puesto que los sistemas de control de temperatura, humedad podrían solucionar este problema. Y además que existiría mayor seguridad en que los productos no se perdieran, y así no perder el ingreso de capital invertido.

1.4 OBJETIVOS.-

1.4.1 OBJETIVO GENERAL.-

- Desarrollar un sistema de control de temperatura y además controlarlo remotamente.

1.4.2 OBJETIVOS ESPECIFICOS.-

- Diseñar e implementar un sistema de control de temperatura, utilizando el PIC16F877A.
- Implementar al circuito de control el sensor LM35 para realizar las mediciones de la temperatura ambiente.
- Diseñar e Implementar el circuito de control para configurar manualmente los rangos de temperatura.
- Diseñar una interfaz de usuario a través del programa Visual Basic para la comunicación serial entre la PC y el microcontrolador.
- Utilizar el software especializado show my PC para controlar remotamente el sistema de control de temperatura.

CAPITULO II

MARCO TEORICO

2.1 CARACTERÍSTICAS DE INVERNADEROS.-

Un invernadero es un ámbito cerrado, cubierto por materiales transparentes, dentro del cual es posible obtener un microclima apropiado para cultivar plantas fuera de estación en condiciones óptimas. Partiendo de un estudio técnico de ambientación climática, es necesario obtener en él, la temperatura, la ventilación apropiada, el control de riego, etc. para alcanzar alta productividad a bajo costo, en menos tiempo, sin daño ambiental, protegiendo al cultivo de lluvias, granizo, heladas, insectos o excesos de viento perjudiciales.



Fig.2 Invernadero

2.1.1 Ventajas del empleo de invernaderos.-

- Producción fuera de época.
- Posibilidad de obtener más de un ciclo de cultivo al año.
- Aumento de la calidad y del rendimiento.
- Precocidad en los frutos.
- Ahorro de agua y fertilizantes.
- Mejora del control de insectos y enfermedades.

2.2 CARACTERÍSTICAS DE LOS CULTIVOS.-

Cuando una planta no resulta productiva es porque ha tenido problemas de exceso o falta de humedad, temperatura, ventilación, y/o falta de luminosidad. Si consideramos que las plantas son seres vivos, no podemos dejar a la improvisación un aspecto tan importante como su microclima. Cada planta tiene un rango de temperaturas y de humedad relativa dentro del cual producen eficientemente; por debajo o por encima de este rango, las plantas se estresan y su productividad declina. Los parámetros que se deben tener en cuenta para realizar una producción eficiente son los siguientes:

2.2.1 VENTILACIÓN.-

El viento cumple la función de regular las temperaturas y humedad relativa en un ambiente natural. Éste, a la vez que desempeña una función vital en la polinización, expulsa los excedentes de humedad y reduce los excesos de temperatura.

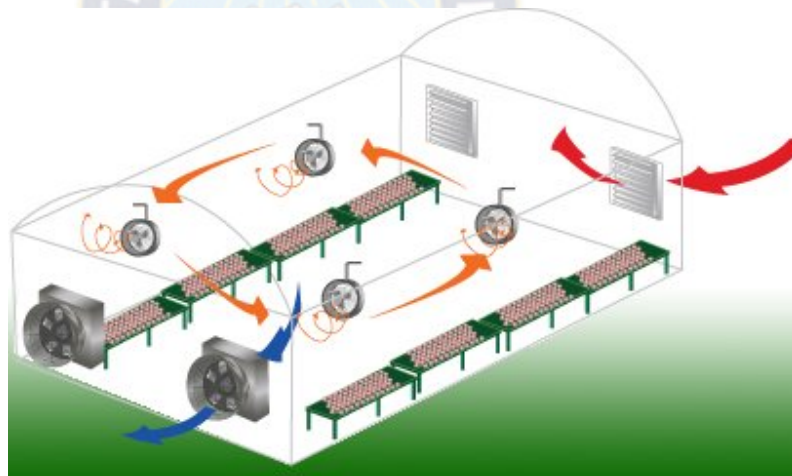


Fig. 3 Ventilación en un invernadero

2.2.2 TEMPERATURA.-

En general, el rango de temperaturas que permite obtener una óptima productividad se encuentra entre los 12 °C y los 32 °C de temperatura ambiente. Ahora bien, por debajo de los 10 °C, las plantas suelen detener sus procesos para entrar en un período de latencia o hibernación, proceso que se revierte cuando la temperatura ambiente vuelve a ser estable por encima de los 12 °C, lo que les permite continuar con su crecimiento.

En el otro extremo del rango también podemos tener inconvenientes; por encima de los 35 °C de temperatura, las plantas reducen su actividad para evitar la deshidratación producida por el excesivo calor. Existe un rango de temperaturas óptimas diferenciado para la germinación de las semillas y para la vida de las plantas:

- Para la germinación, las temperaturas oscilan, generalmente, entre los 28 °C y los 32 °C.
- Para el normal crecimiento de las plantas, entre los 22 °C y los 28 °C.

2.3 MICROCONTROLADOR: PIC 16F877A.-

El microcontrolador PIC16F877A pertenece a una gran familia de microcontroladores de 8 bits (bus de datos).



Fig.4 Microcontrolador PIC16F877

2.3.1 CARACTERISTICAS GENERALES DEL PIC16F877.-

- CPU RISC
- Frecuencia de operación de 0 a 20 MHz
- Hasta 8k x 14 bits de memoria Flash de programa
- Hasta 368 bytes de memoria de datos (RAM)
- Hasta 256 bytes de memoria de datos EEPROM
- 4 fuentes de interrupción.
- Timer de encendido (PWRT)
- Timer de arranque del oscilador (OST)
- Sistema de vigilancia Watchdog timer.
- Modo SLEEP de bajo consumo de energía
- Bajo consumo de potencia.

- Dos módulos de Captura, Comparación y PWM
- Convertidor Analógico/Digital: de 10 bits, hasta 8 canales
- Puerto Serie Síncrono (SSP)
- Puerto Serie Universal (USART/SCI).
- Puerto Paralelo Esclavo (PSP): de 8 bits con líneas de protocolo.

Estas características se conjugan para lograr un dispositivo altamente eficiente en el uso de la memoria de datos y programa y por lo tanto en la velocidad de ejecución. La memoria de los datos se divide en 4 bancos que se seleccionan con los bits RP1 y RP0 que se encuentran en el registro STATUS.

En cada banco hay 7Fh posiciones de memoria. Las posiciones más bajas están, reservadas para los Registros de Funciones Especiales. Por encima de los Registros de Funciones Especiales se encuentran los Registros de Propósito General, que se utilizan como posiciones de memoria RAM estática. Todos están estructurados en bancos. Algunos Registros de Funciones Especiales están reflejados en varios bancos para reducir el código y tener un acceso más rápido.



Fig.5 Diagrama de pines del PIC16F877A

2.3.2 OSCILADOR.-

Los modos de operación se muestran en la siguiente lista:

- LP Baja frecuencia (y bajo consumo de potencia)
- XT Cristal / Resonador cerámico externos, (Media frecuencia)
- HS Alta velocidad (y alta potencia) Cristal/resonador
- RC Resistencia / capacitor externos (mismo que EXTRC con CLKOUT)

Los tres modos LP, XT y HS usan un cristal o resonador externo, la diferencia sin embargo es la ganancia de los drivers internos, lo cual se ve reflejado en el rango de frecuencia admitido y la potencia consumida.

2.3.3 DIAGRAMA DE BLOQUES PIC16F877.-

En la siguiente figura se muestra a manera de bloques la organización Interna del PIC16F877, Se muestra también junto a este diagrama su diagrama de patitas, para tener una visión conjunta del interior y exterior del Chip.

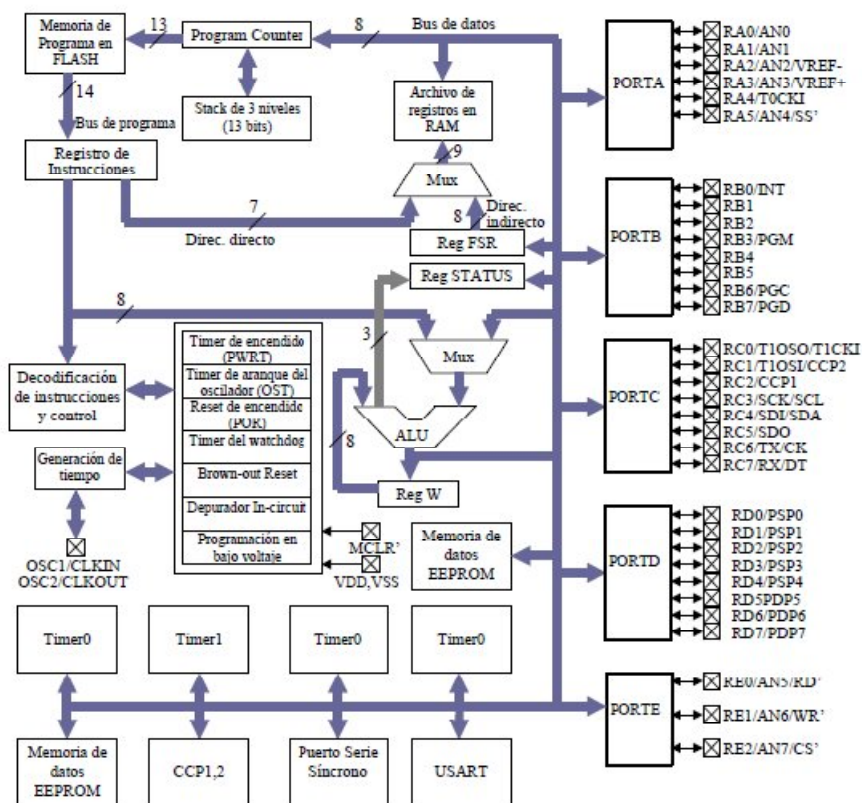


Fig. 6 Diagrama de bloques del PIC16F877A

2.3.4 ORGANIZACIÓN DE LA MEMORIA DEL PIC.-

Los PIC tienen dos tipos de memoria: Memoria de Datos y Memoria de programa, cada bloque con su propio bus: Bus de datos y Bus de programa; por lo cual cada bloque puede ser accedido durante un mismo ciclo de oscilación. La Memoria de datos a su vez se divide en:

- **Memoria de programa.** En sus 8192 posiciones contiene el programa con las instrucciones que gobiernan la aplicación. Es del tipo no volátil.
- **Memoria de datos (RAM).** Se destina a guardar las variables y datos. Es volátil, es decir los datos almacenados se borran cuando desaparece la alimentación
- **Memoria EEPROM de datos.** Es una pequeña área de memoria de datos de lectura y escritura no volátil.

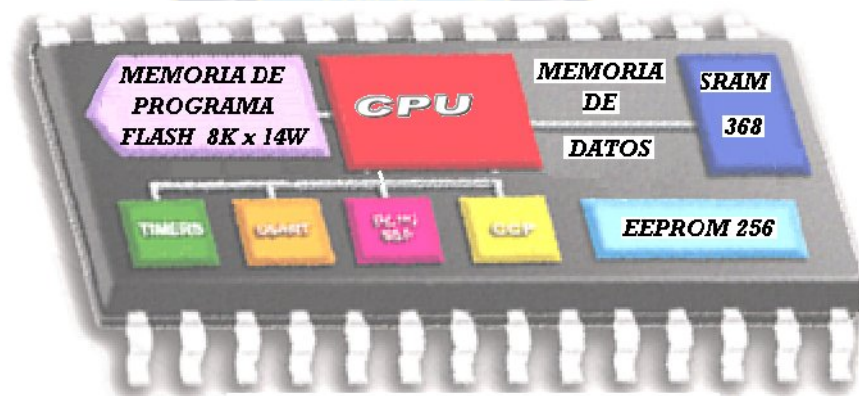


Fig.7 Organización de la memoria del PIC16F877A

2.4 SENSOR DE TEMPERATURA: LM35.-

El LM35 es un sensor de temperatura con una precisión calibrada de 1°C. Puede medir temperaturas en el rango que abarca desde -55° a + 150°C. La salida es muy lineal y cada grado centígrado equivale a 10 mV en la salida.

2.4.1 CARACTERÍSTICAS.-

- Esta calibrado directamente en grados Celsius.
- La tensión de salida es proporcional a la temperatura.
- Tiene una precisión garantizada de 0.5°C a 25°C.

- Opera entre 4 y 30 volts de alimentación.
- Baja corriente de alimentación (60uA).
- Amplio rango de funcionamiento (desde -55° a + 150°C).

Su tensión de salida es proporcional a la temperatura, en la escala Celsius. No necesita calibración externa y es de bajo costo. Funciona en el rango de alimentación comprendido entre 4 y 30 voltios.

Como ventaja adicional, el LM35 no requiere de circuitos adicionales para su calibración externa cuando se desea obtener una precisión del orden de ± 0.25 °C a temperatura ambiente, y ± 0.75 °C en un rango de temperatura desde 55 a 150 °C.

La baja impedancia de salida, su salida lineal y su precisa calibración inherente hace posible una fácil instalación en un circuito de control.

Debido a su baja corriente de alimentación (60uA), se produce un efecto de autocalentamiento reducido, menos de 0.1 °C en situación de aire estacionario



Fig. 8 Sensor LM35

2.4.2.- ENCAPSULADO.-

El sensor se presenta en diferentes encapsulados pero el mas común es el to-92 de igual forma que un típico transistor con 3 patas, dos de ellas para alimentarlo y la tercera nos entrega un valor de tensión proporcional a la temperatura medida por el dispositivo

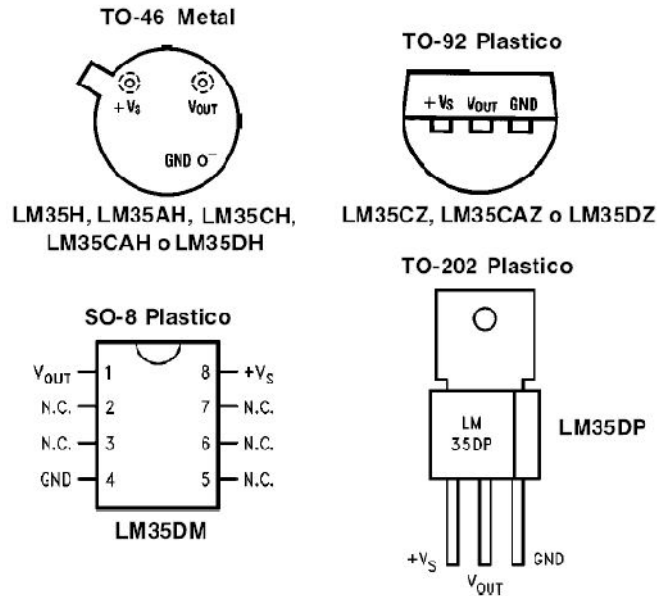


Fig. 9 Tipos de encapsulado del sensor LM35

2.4.3 DATOS TECNICOS DEL LM35.-

PARAMETRO	VALOR
Temperatura mínima	-40°C , -55 °C
Temperatura máxima	100°C, 110 °C, 150 °C
Precisión (+/-) °C	1, 5
Mínimo voltaje de alimentación	4 Voltios
Máximo voltaje de alimentación	30 Voltios
Ganancia del sensor	10mV/ °C
Salida de impedancia	5, 3
Package	TO220
Número de pines	3

2.4.4 CARACTERISTICAS TIPICAS.-

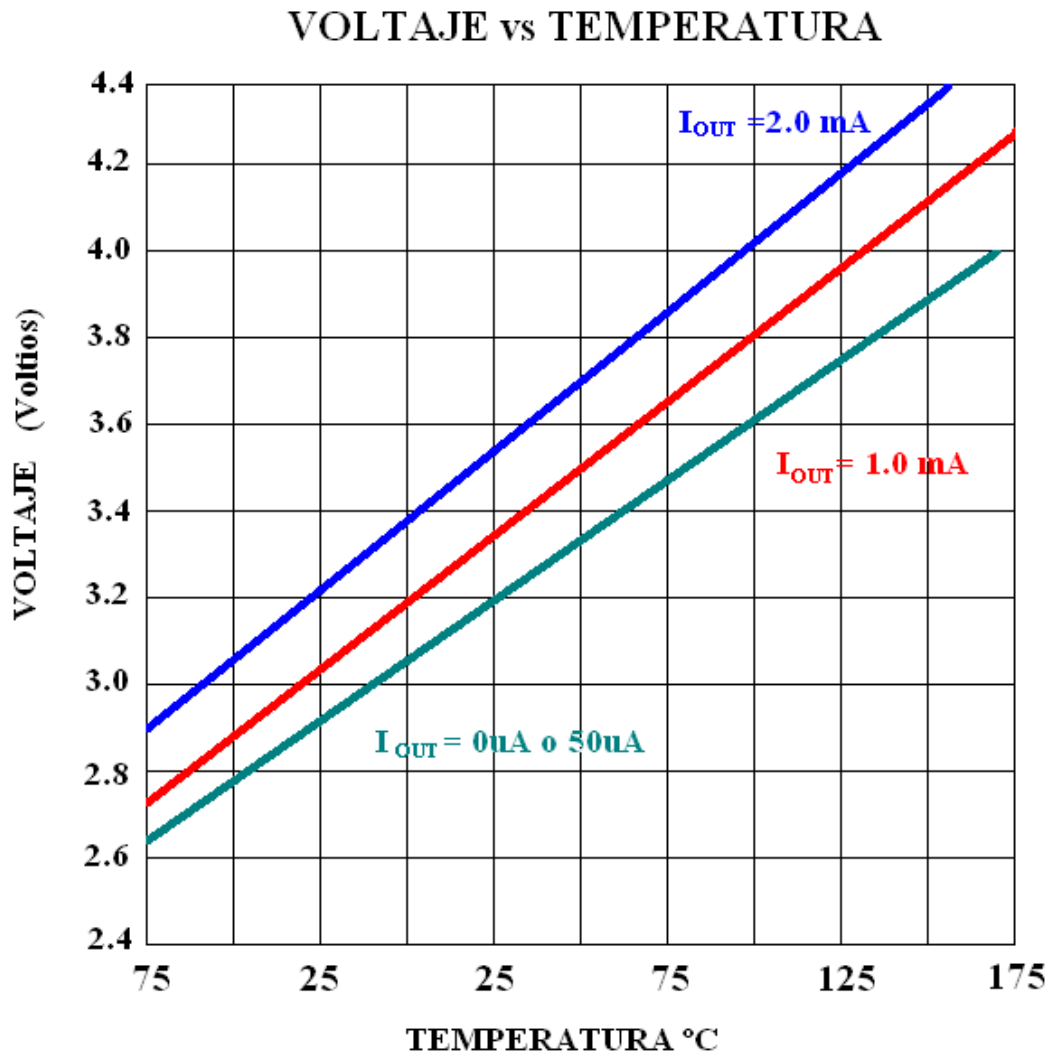


Fig. 10 Curva característica del sensor LM35: Voltaje vs. Temperatura

2.5 DISPLAY LÍQUIDO LCD.-

Se trata de un módulo microcontrolador capaz de representar 2 líneas de 16 caracteres cada una. A través de 8 líneas de datos se le envía el carácter ASCII que se desea visualizar así como ciertos códigos de control que permiten realizar diferentes efectos de visualización. Igualmente mediante estas líneas de datos el módulo devuelve información de su estado interno. Con otras tres señales adicionales se controla el flujo de información entre el módulo LCD y el equipo informático que lo gestiona.



Fig. 11 Display líquido LCD

2.5.1 CARACTERÍSTICAS.-

El LCD está constituido por un circuito impreso en el que están integrados los controladores del display y los pines para la conexión del display. Sobre el circuito impreso se encuentra el LCD en sí, rodeado por una estructura metálica que lo protege.

En total se pueden visualizar 2 líneas de 16 caracteres cada una, es decir, $2 \times 16 = 32$ caracteres. A pesar de que el display sólo puede visualizar 16 caracteres por línea, puede almacenar en total 40 por línea. La tensión nominal de alimentación es de 5V, con un consumo menor de 5mA.

2.5.2 LA MEMORIA DEL LCD.-

El LCD dispone de dos tipos de memorias independientes: la **DD RAM** y la **CG RAM**

2.5.2.1.- DD RAM (Display Data Ram).-

En esta memoria se almacenan los caracteres que están siendo visualizados o que se encuentran en posiciones no visibles. El display almacena en esta memoria dos líneas de 40 caracteres pero sólo se visualizan 2 líneas de 16 caracteres. Por ello la **DD RAM** tiene un tamaño de $2 \times 40 = 80$ Bytes.

Debido a esta peculiar disposición de la **DD RAM** se puede pensar en el display como un *display virtual* constituido por dos líneas de 40 caracteres cada una.

La posición situada más a la izquierda de cada línea es la **posición 1** y la situada más a la derecha es la **posición 40**. Para localizar los elementos dentro del *display virtual* se va a

utilizar un par de coordenadas (x,y) donde x representa la posición horizontal (comprendida entre 1-40) e y representa la línea (1-2).

El display real es una *ventana* en la que se visualizan dos líneas de 16 caracteres. Es lo que el usuario está viendo. En el ejemplo de la figura se encuentra almacenado en la línea 1 del display virtual el mensaje: “ESTO ES UNA PRUEBA DE UN MENSAJE”. Sin embargo, en este ejemplo, el usuario sólo verá en el display el mensaje “PRUEBA DE UN MEN” que tiene exactamente 16 caracteres de longitud.

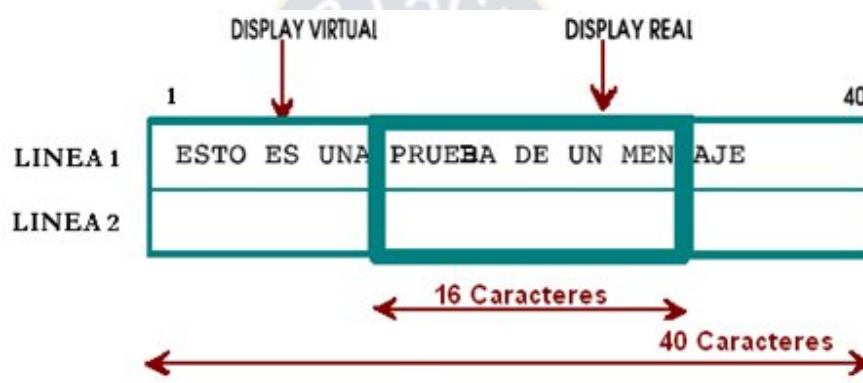


Fig.12 Display virtual y display real

El mapa de memoria de la **DD RAM** está constituido por dos bloques de 40 bytes. El primer bloque se corresponde con los 40 caracteres de la línea 1 del display virtual. El segundo bloque con la segunda línea. En la figura se han representado las direcciones en hexadecimal. Así, las direcciones \$00-\$27 están asociadas a las posiciones (1,1)-(40,1) del display virtual y las direcciones \$40-\$67 a las posiciones (1,2)-(40,2). En el ejemplo, en la dirección \$40 de la DD RAM se encuentra almacenado el carácter H, que se corresponde con la posición (1,2) del display virtual. En la dirección \$02 se encuentra el carácter S, posición (3,1) del display virtual. Nótese que **los bloques de memoria asociados a la línea 1 y 2 no son contiguos.**

Las operaciones de escritura en el display, en realidad son operaciones de escritura en la memoria DD RAM. Según en la posición de esta memoria en la que se escriba el carácter, aparecerá en una posición u otra en el display real. Los caracteres enviados al display pueden

ser visibles si se encuentran en posiciones que caen dentro del display real o pueden ser no visibles. En la figura, las posiciones (1,1)-(16,1) y (1,2)-(16,2) son visibles. Todos los caracteres enviados a esas posiciones serán visibles. Si se envía un carácter a cualquiera de las otras posiciones no será visible.

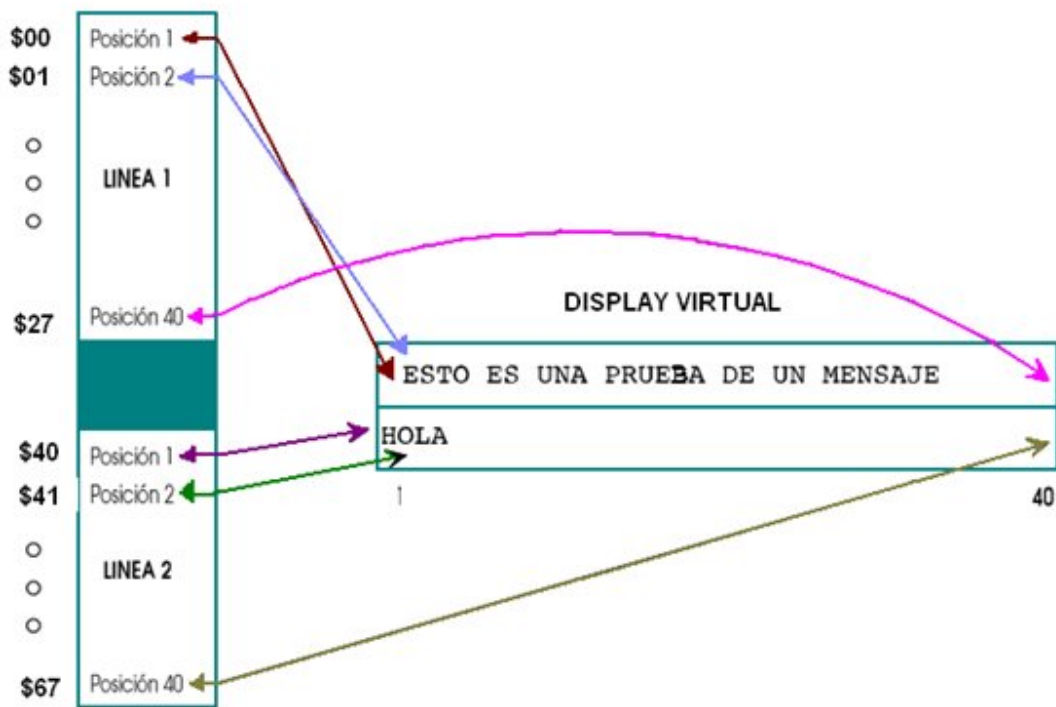


Fig. 13 Mapa de la memoria de la DDRAM y sus posiciones en el display virtual

2.5.2.2.- LA CG RAM (Character Generator RAM).-

La CG RAM es la memoria que contiene los caracteres definibles por el usuario. Está formada por 64 posiciones, con direcciones \$00-\$3F. Cada posición es de 5 bits. La memoria está dividida en 8 bloques, correspondiendo cada bloque a un carácter definible por el usuario. Por ello el usuario puede definir como máximo 8 caracteres, cuyos códigos van del 0 al 7. En la figura 14 se ha representado la CG RAM. Todas las direcciones están en hexadecimal.

Como se representa en la figura 14, cada carácter está constituido por una matriz de 5 columnas x 8 filas. Para definir un carácter y asignarle por ejemplo el código 0 habrá que

almacenar en las posiciones \$00-\$07 los valores binarios de las 8 filas del carácter del usuario. Un bit con valor 1 representa un punto encendido. Un bit con valor 0 es un punto apagado.



Fig. 14 Carácter definido por el usuario y los valores almacenados en la CGRAM

2.5.3 INTERFAZ HARDWARE.-

Asignación de pines:

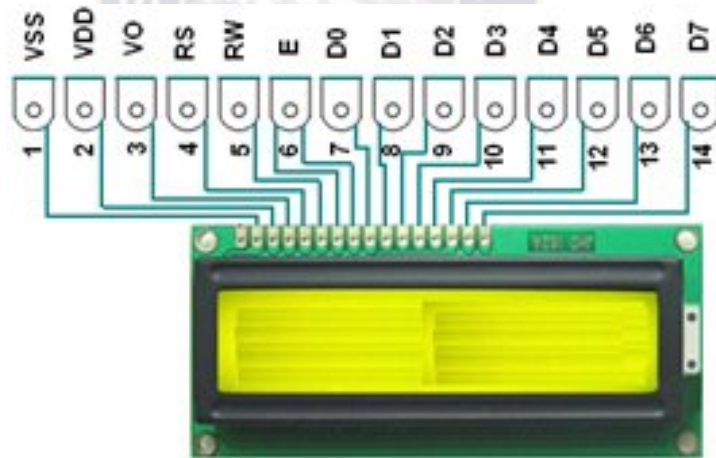


Fig. 15 Asignación de pines del LCD

Nº de PIN	Símbolo	Descripción
1	VSS	Tierra
2	VDD	Alimentación
3	VO	Voltaje de ajuste de contraste
4	RS	Selección de registro
5	R/W	Lectura/Escritura
6	E	Enable
7 – 14	D0 – D7	Bit de datos

2.5.3.1 Interfaz del display.-

Los datos se transmiten por un bus de datos de 8 bits de anchura (El display también ofrece la posibilidad de trabajar con este bus multiplexado en dos grupos de 4 bits). Para el control del display son necesarios 3 bits: una señal de **Enable (E)**, una para indicar **lectura/escritura (R/W)** y otra para seleccionar uno de los dos registros internos (**RS**). Por ello, en el caso peor, el sistema de control del display necesitará utilizar $8+3=11$ bits.

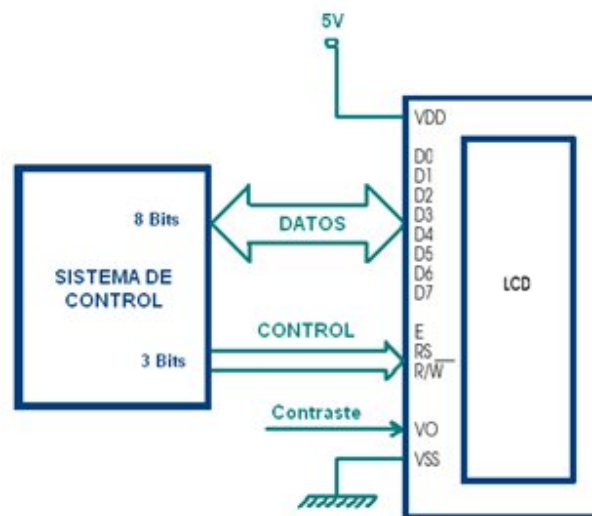


Fig.16 Interfaz del LCD con un sistema de control

2.5.4 COMANDOS DEL LCD.-

El LCD se controla mediante comandos que se envían al registro de control del LCD, seleccionado al poner la señal RS a nivel bajo (0). Cuando lo que se quiere es imprimir caracteres en el display o enviar información a la CG RAM para definir caracteres se selecciona el registro de datos poniendo RS a nivel alto (1).

Existe un *contador de direcciones* para la DD RAM y otro para la CG RAM, el cual contiene la dirección a la que se va a acceder. Modificando el *contador de direcciones* es posible acceder a cualquier posición tanto de la CG RAM como de la DD RAM. Con ello se consigue por ejemplo imprimir caracteres en cualquier posición del LCD. Cada vez que se realiza un acceso a memoria, el contador de direcciones se incrementa o decrementa automáticamente, según cómo se haya configurado el LCD.

Al LCD le lleva un cierto tiempo procesar cada comando enviado. Por ello, para que se ejecute el comando especificado es necesario asegurarse de que el comando anterior ha finalizado. Existen dos estrategias para realizar esto. La primera se basa en leer del display el **bit de ocupado**. Si este bit se encuentra a 1 quiere decir que el LCD está ocupado procesando el comando anterior y por tanto no puede procesar nuevos comandos. La segunda estrategia, menos elegante pero más cómoda de implementar, consiste en realizar una pausa antes de volver a enviar el siguiente comando. Los tiempos máximos que tarda el display en procesar los comandos están especificados por el fabricante y tienen un valor típico de 40 μ s. Si se realiza una pausa mayor o igual a esta se tiene garantía de que el display ha terminado de ejecutar el comando.

2.5.4.1 Resumen de comandos.-

	RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0
Borrar Display	0	0	0	0	0	0	0	0	0	0
Cursor o Home	0	0	0	0	0	0	0	0	1	.
Establecer modo de funcionamiento	0	0	0	0	0	0	0	1	I/D	S
Control ON/OFF	0	0	0	0	0	0	1	D	C	E
Desplazamiento del cursor/display	0	0	0	0	0	1	S/C	R/L	.	.
Modo de transferencia	0	0	0	0	1	DL	1	0	.	.
Acceso a la memoria CGRAM	0	0	0	1	Dirección de la CGRAM					
Acceso a la memoria DDRAM	0	0	1	Dirección de la DDRAM						
Lectura de dirección y de flag de ocupado	0	.	BF	Contador de dirección						
Escritura de datos en la CGRAM y DDRAM	1	0	Dato a escribir							
Lectura de datos en la CGRAM y DDRAM	1	.	Dato leído							

I/D = 1: Incrementar contador de direcciones

I/D = 0: Decrementar contador de direcciones

S = 1: Desplazamiento del display

S = 0: Display quieto

D = 1: Display ON

D = 0: Display OFF

C = 1: Cursor ON

C = 0: Cursor OFF

B = 1: Parpadeo del carácter en la posición del cursor	B = 0: No hay parpadeo
S/C = 1: Desplazar el display	S/C = 0: Desplaza el cursor
R/L = 1: Desplazamiento a la derecha	R/L = 0: Desplazamiento a la izquierda
DL = 1: Configura display a 8 bits	DL = 0: Configura display a 4 bits
BF = 1: Display ocupado	BF = 0: Display listo para ejecutar otra operación

2.5.5 LCD CON EL COMPILADOR CCS.-

El compilador C incluye un fichero (driver) que permite trabajar con un LCD. El archivo es *LCD.C* y debe llamarse con un *#include*. Este archivo dispone de varias funciones ya definidas.

- **Lcd_init();** Es la primera función que debe ser llamada. Borra el LCD y lo configura en el formato de 4 bits, con dos líneas y caracteres de 5x8 puntos, en modo encendido, cursor apagado y sin parpadeo. Configura el LCD con un autoincremento de puntero de direcciones y sin desplazamiento del display real.
- **Lcd_gotoxy(byte x, byte y);** Indica la posición de acceso al LCD. Por ejemplo (1,1) indica la primera posición de la primera línea y (1,2) indica la primera posición de la segunda línea.
- **Lcd_getc(byte x, byte y);** Lee el carácter de la posición (x,y).
- **Lcd_putc(char s);** S es una variable tipo *char*. Esta función escribe la posición correspondiente. Si además se indica:

\f se limpia el LCD.

\n el cursor va a la posición (1,2)

\b el cursor retrocede una posición

2.6 EL RELE.-

Es un dispositivo que consta de dos circuitos diferentes: un circuito electromagnético (electroimán) y un circuito de contactos, al cual aplicaremos el circuito que queremos controlar. En la siguiente figura se puede ver su simbología así como su constitución (relé de armadura).

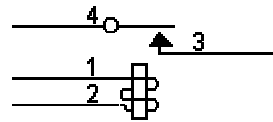
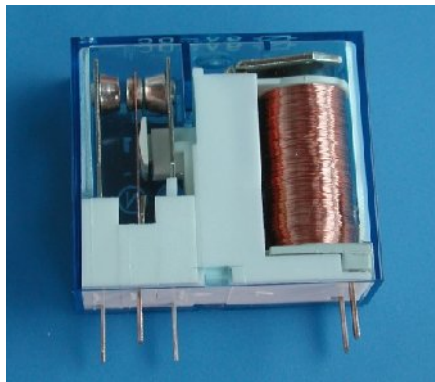


Fig. 17 Relé físico y Símbolo del relé

2.6.1 ESTRUCTURA.-

En general, podemos distinguir en el esquema general de un relé los siguientes bloques:

- Circuito de entrada, control o excitación.
- Circuito de acoplamiento.
- Circuito de salida, carga o maniobra, constituido por:
 - ✓ circuito excitador.
 - ✓ dispositivo conmutador de frecuencia.
 - ✓ protecciones.

2.6.2 FUNCIONAMIENTO.-

Su funcionamiento se basa en el fenómeno electromagnético. Cuando la corriente atraviesa la bobina, produce un campo magnético que magnetiza un núcleo de hierro dulce (ferrita). Este atrae al inducido que fuerza a los contactos a tocarse. Cuando la corriente se desconecta vuelven a separarse.

2.6.3 CARACTERÍSTICAS.-

2.6.3.1 Parte electromagnética.-

- Corriente de excitación.- Intensidad, que circula por la bobina, necesaria para activar el relé.
- Tensión nominal.- Tensión de trabajo para la cual el relé se activa.

- Tensión de trabajo.- Margen entre la tensión mínima y máxima, garantizando el funcionamiento correcto del dispositivo.
- Consumo nominal de la bobina.- Potencia que consume la bobina cuando el relé está excitado con la tensión nominal a 20°C.

2.6.3.2 Contactos ó Parte mecánica.-

- Tensión de conexión.- Tensión entre contactos antes de cerrar o después de abrir.
- Intensidad de conexión.- Intensidad máxima que un relé puede conectar o desconectarlo.
- Intensidad máxima de trabajo.- Intensidad máxima que puede circular por los contactos cuando se han cerrado.

Los materiales con los que se fabrican los contactos son: plata y aleaciones de plata que pueden ser con cobre, níquel u óxido de cadmio. El uso del material que se elija en su fabricación dependerá de su aplicación y vida útil necesaria de los mismos.

2.6.4 TIPOS DE RELÉS.-

2.6.4.1 Relé de armadura.-

El electroimán hace vascular la armadura al ser excitada, cerrando los contactos dependiendo de si es normalmente abierto o normalmente cerrado.

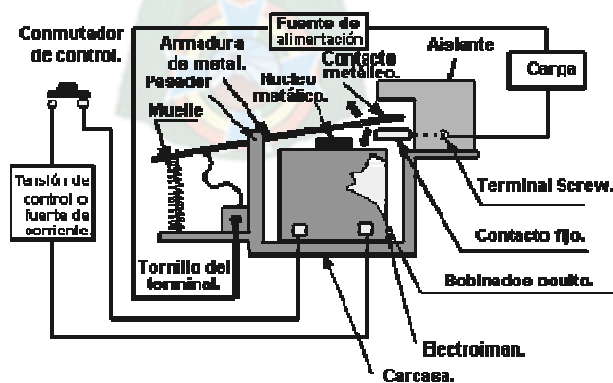


Fig.18 Partes de un relé de armadura

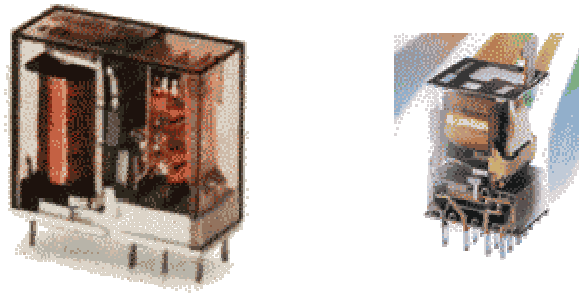


Fig. 19 Relés de armadura

2.6.4.2 Relé de núcleo móvil.-

Tienen un émbolo en lugar de la armadura. Se utiliza un solenoide para cerrar los contactos. Se suele aplicar cuando hay que manejar grandes intensidades.

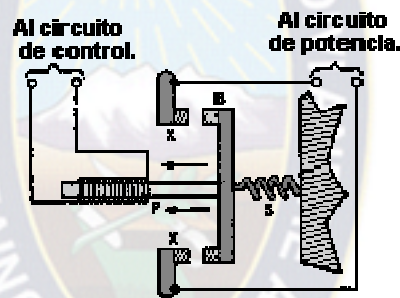


Fig. 20 Relé de núcleo móvil

Las aplicaciones de este tipo de componentes son múltiples: en electricidad, en automatismos eléctricos, control de motores industriales; en electrónica: sirven básicamente para manejar tensiones y corrientes superiores a los del circuito propiamente dicho, se utilizan como interfaces para PC, en interruptores crepusculares, en alarmas, en amplificadores, etc.

2.6.4.3 Relé tipo Reed o de Lengüeta.-

Formados por una ampolla de vidrio, en cuyo interior están situados los contactos (pueden ser múltiples) montados sobre delgadas láminas metálicas. Dichos contactos se cierran por medio de la excitación de una bobina, que está situada alrededor de dicha ampolla.

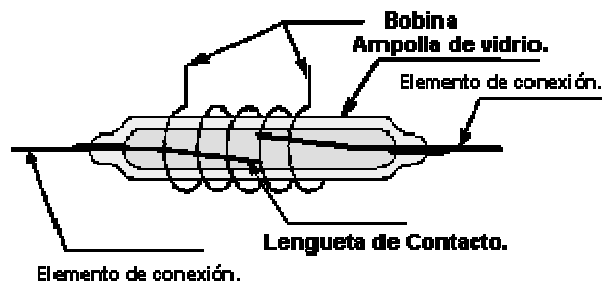


Fig.21 Relé tipo Reed o de lengüeta

2.6.4.4 Relés Polarizados.-

Llevan una pequeña armadura, solidaria a un imán permanente. El extremo inferior puede girar dentro de los polos de un electroimán y el otro lleva una cabeza de contacto. Si se excita al electroimán, se mueve la armadura y cierra los contactos. Si la polaridad es la opuesta girará en sentido contrario, abriendo los contactos ó cerrando otro circuito (ó varios)



Fig.22 Relés de estado sólido

2.6.4.5 Relé de estado sólido SSR.-

Un relé de estado sólido SSR (Solid State Relay), es un circuito electrónico que contiene en su interior un circuito disparado por nivel, acoplado a un interruptor semiconductor, un transistor o un tiristor. Por SSR se entenderá un producto construido y comprobado en una fábrica, no un dispositivo formado por componentes independientes que se han montado sobre una placa de circuito impreso.

Estructura del SSR:

➤ ***Circuito de Entrada o de Control:***

Control por tensión continua: el circuito de entrada suele ser un LED (Fotodiodo), solo o con una resistencia en serie, también podemos encontrarlo con un diodo en antiparalelo para evitar la inversión de la polaridad por accidente. Los niveles de entrada son compatibles con TTL, CMOS, y otros valores normalizados (12V, 24V, etc.).

Control por tensión Alterna: El circuito de entrada suele ser como el anterior incorporando un puente rectificador integrado y una fuente de corriente continua para polarizar el diodo LED.

➤ ***Acoplamiento:***

El acoplamiento con el circuito se realiza por medio de un optoacoplador o por medio de un transformador que se encuentra acoplado de forma magnética con el circuito de disparo del Triac.

➤ ***Circuito de Conmutación o de salida:***

El circuito de salida contiene los dispositivos semiconductores de potencia con su correspondiente circuito excitador. Este circuito será diferente según queramos conmutar CC, CA.

2.7 COOLERS.-

El Sistema de ventilación soluciona las situaciones de alta temperatura, humedad, olores, etc. en sistemas fabriles o depósitos, sin costo operativo ni de mantenimiento. Sus características principales son:

- No consume energía eléctrica
- Fácil instalación en el techo
- Alabes de aluminio

- No produce ruidos
- No necesita instalaciones especiales
- No necesita mantenimiento
- Funcionamiento ininterrumpido
- Rodamientos blindados autolubricados

2.7.1 PARTES CONSTITUTIVAS DE UN EQUIPO EÓLICO.-

Base: Adapta el extractor a cualquier tipo de techo: chapa, fibrocemento, loza, claraboyas, etc. Parabólico, con pendiente, auto portantes, salidas laterales. Está construida en chapa galvanizada, grafada y estañada de excelente terminación, con nervadura superior para encastre del extractor.

Extractor Eólico: Consta de parte giratoria accionada por el viento y parte fija que lo asegura a la base. Álabes de aluminio con nervaduras longitudinales y micro nervaduras que le otorgan mayor rigidez. Ejes de Acero con rulemanes blindados

Opcional: Capeta de cierre de paso de aire para invierno o para regular la salida de aire del interior. Disco y conducto galvanizado para alojamiento de capeta. Bulonería y eje con tratamiento anticorrosión (zincado). Prensa, Grower y autofrenantes para darle mayor robustez. Se acciona desde el interior por cadenas o desde el exterior por llave.



Fig.23 Cooler

2.8 TRANSISTORES.-

2.8.1 BC548.-

2.8.1.1 Descripción.-

Es un transistor bipolar NPN, eléctricamente es similar al transistor 2N3904 de fabricación estadounidense y al 2SC1815 japonés, aunque las asignaciones de los pines de conexión es distinta. El dispositivo viene integrado en un encapsulado plástico tipo TO-92. El orden de los pines mirando la parte plana del encapsulado de derecha a izquierda es emisor, base y colector.

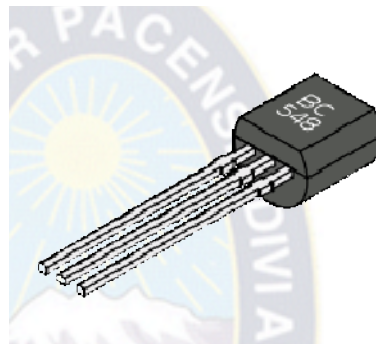


Fig. 24 Transistor BC548

2.8.1.2 Principales características.-

Voltaje colector emisor en corte 30V (V_{ce0}). Voltaje colector emisor en saturación 30V (V_{ces}). Voltaje emisor base en corte 5V (V_{ebo}). Corriente de colector constante 100mA (I_c). Potencia total disipada 500 mW (P_d). Encapsulado de plástico TO-92. Su par complementario PNP es el Transistor BC558.

2.8.2 BC547.-

Es un transistor bipolar, compuesto por silicio, es un transistor amplificador de audio y VHF Freq. Driver con una corriente máxima de colector de 0.6 ampere, en su composición posee una placa de semiconductor con tres regiones consecutivas de diferente conductibilidad eléctrica los cuales forman dos uniones n-p-n, las dos regiones extremas tienen un mismo tipo de conductibilidad, la intermedia, conductibilidad de otro tipo, estas son llamadas emisor, colector y base.

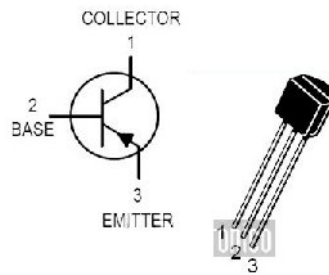


Fig. 25 Transistor BC547 y simbología

2.8.2.1 Características.-

- Polaridad (N-P-N)
- Amplificador, audio a VHF Freq. Driver
- Corriente máxima de colector (I_c) 0.6 Ampere
- De colector a base (CBO) 75 Voltios
- De colector a emisor (CEO) 40 Voltios
- De emisor a base (EBO) 6 Voltios
- Ganancia típica de la corriente directa (h_{fe}) 200 Min
- Máxima disipación de potencia en colector (P_d) 0.625 (Watts)
- Frecuencia en (MHz) 300 Min

2.8.3 BC559.-

BC559 es una familia de transistores de silicio PNP de propósito general.



Fig. 26 Transistor BC559

Existen cuatro tipos de transistores: BC556, BC557, BC558 y BC559. Cada tipo de transistor a su vez se subdivide en tres grupos: A, B y C. El tipo de transistor denota sus

características eléctricas, mientras que el grupo indica la ganancia de corriente. No todos los tipos están disponibles para todos los grupos. El siguiente cuadro resume lo ante-dicho:

Mayor ganancia de corriente
→

↑ Mayor tensión de ruptura

	A	B	C
BC556	X	X	
BC557	X	X	X
BC558	X	X	X
BC559	X	X	X

2.8.3.1 Características generales.-

- Encapsulado TO-92
- Corriente máxima de colector 100 mA
- Corriente de pico máxima de colector 200 mA
- Corriente de pico máxima de emisor 200 mA
- Corriente de pico máxima de base 200 mA
- Potencia disipada máxima a 25°C 500 mW

2.8.3.2 Tensión de ruptura Colector-Emisor.-

- BC546 60 V
- BC547 45 V
- BC548 30V
- BC549 30V

2.8.3.3 Ganancia de corriente Estática y Dinámica.-

- La ganancia de corriente denota cuantas veces la corriente de colector será mayor que la corriente de base.
- La ganancia estática se especifica para corriente continua y se designa con la abreviatura hfe (mayúsculas)

- La ganancia dinámica se especifica para corriente alterna (es decir, señal) y se designa con la abreviatura h_{fe} (minúsculas).
- Estas ganancias no son constantes y en un transistor dado dependen de la corriente de colector

2.9. **BOMBILLAS (FOCOS).**-

La forma, tamaño, material y acabado de las bombillas varían según las necesidades de su aplicación. Las formas van de tubulares a esféricas y de parabólicas a forma de llama. Las bombillas se designan con una letra que se refiere a la forma y por un número que es el diámetro máximo en octavos de pulgada; por ejemplo, A-19 designa una bombilla en forma de A con un diámetro de $19/8$ o $23/8$ pulgadas.

La mayor parte de las bombillas están hechas de vidrio suave de plomo ocal, aunque también se usa vidrio duro de gran resistencia al calor en aplicaciones de alta temperatura, esmerilados en su interior para difusión moderada de la luz sin reducción apreciable en la salida de luz. Las bombillas transparentes y sin esmerilar se usan cuando se requiere control preciso de luz desde un punto o fuente de línea. Para otras lámparas se utiliza vidrio de cuarzo fundido o de alto sílice.



Fig. 27 Bombilla (foco)

2.10 CRISTAL DE CUARZO.-

2.10.1 CARACTERISTICAS DEL CUARZO.-

Muchas son las veces que hemos oído hablar del cristal de cuarzo como elemento imprescindible en gran variedad de aparatos electrónicos. Así, por ejemplo, raro es encontrarse un reloj que no lleve en su interior tan preciado cristal. La razón de la utilización masiva del cuarzo radica en una propiedad electromecánica, conocida como efecto "piezoeléctrico", la cual es, como veremos, de una gran utilidad en los osciladores. El cuarzo tiene la propiedad de deformarse mecánicamente, es decir, aumentar o disminuir su volumen, cuando se le aplica una diferencia de potencial entre sus extremos. Además, este efecto piezoeléctrico es reversible, por lo que, si de alguna forma somos capaces de oprimir un cristal de cuarzo, podríamos observar cómo, durante el tiempo en que el cristal está reduciendo su tamaño, produciría una diferencia de potencial entre sus caras opuestas. Este efecto reversible es parecido al de un motor eléctrico, el cual, si le aplicamos una diferencia de polaridad comienza a girar pero si, por el contrario, lo hacemos girar manualmente, se produciría una diferencia de potencial entre sus dos conexiones.



Fig. 28 Mineral Cuarzo

El cuarzo es uno de los minerales más abundantes en la naturaleza formado por anhídrido de silicio. Se encuentra en la naturaleza en diferentes formas, principalmente como "cuarzo a", que se obtiene a alta temperatura y es hexagonal, y como "cuarzo b", que existe a temperatura ordinaria. Sin embargo, para su utilización en circuitos, la única variedad que nos interesa es la formada por cristales prismáticos hexagonales.

Volviendo al efecto piezoeléctrico, diremos que un cristal de cuarzo tiene una frecuencia natural de oscilación. Supongamos que conectemos un cristal de cuarzo a una diferencia de potencial provocando, por tanto, que este se deforme; si, a continuación, dejamos de aplicarle la diferencia de potencial, el cristal tenderá a su forma original ya que ha cesado la causa que lo deformaba. Durante su "vuelta" al estado original, el cristal, comienza a oscilar aumentando y disminuyendo su tamaño hasta que, al cabo de cierto tiempo, se detendrá definitivamente. Este aumento y disminución de tamaño son oscilaciones propias del cristal y a una frecuencia fija que depende exclusivamente del cristal y es lo que llamamos frecuencia natural de oscilación.

Para comprender mejor esta oscilación del cristal de cuarzo, pensemos en el clásico globo inflado de aire. Supongamos que cogemos de un extremo del globo y lo estiramos cierta cantidad sin llegar a explotarlo. El globo se deforma. Pues bien, si, a continuación, lo soltamos, el globo evidentemente, va a volver a su posición original. Pero esta "vuelta" a su posición original no es instantánea sino que, aunque apenas se aprecie debido a la velocidad con que ocurre, el globo, una vez que hemos dejado de estirarlo, vuelve a su posición oscilando, es decir, primero se hace más pequeño que inicialmente, luego más grande, de nuevo más pequeño y así sucesivamente hasta que termina por adoptar su tamaño original. Esto lo hace en un tiempo que podría ser del orden de 0,2 segundos y depende del material con que esté hecho el globo. Para hacernos una idea aproximada de las oscilaciones del cristal de cuarzo pensemos que este puede oscilar con frecuencias del orden de MHz, es decir, de millones de veces por segundo.

2.10.2 CARACTERISTICAS DEL OSCILADOR DE CRISTAL.-

El oscilador de cristal se caracteriza por su estabilidad de frecuencia y pureza de fase, dada por el resonador.

La frecuencia es estable frente a variaciones de la tensión de alimentación. La dependencia con la temperatura depende del resonador, pero un valor típico para cristales de cuarzo es de 0'005% del valor a 25 °C, en el margen de 0 a 70 °C.

Estos osciladores admiten un pequeño ajuste de frecuencia, con un condensador en serie con el resonador, que aproxima la frecuencia de este, de la resonancia serie a la paralela. Este ajuste se puede utilizar en los VCO para modular su salida.



Fig. 29 Cristal de cuarzo

Con lo visto sobre el efecto piezoeléctrico parece lógico poder aplicar las propiedades de este material, el cuarzo, para producir oscilaciones. En efecto, si a un cristal de cuarzo le aplicamos sobre sus caras opuestas una diferencia de potencial, y el dispositivo está montado adecuadamente, comenzarían a producirse fuerzas en las cargas del interior del cristal. Estas fuerzas entre sus cargas provocarían deformaciones en el cristal y darían lugar a un sistema electromecánico que comenzaría a oscilar. Sin embargo, vuelve a ocurrir lo mismo que en los circuitos formados por un condensador y por una inductancia. Esto es, las oscilaciones del cristal no duran indefinidamente, ya que se producen rozamientos en la estructura interna que hacen que se vayan amortiguando hasta llegar a desaparecer. Por tanto, necesita de un circuito externo que mantenga las oscilaciones, compensando las pérdidas producidas por el rozamiento.

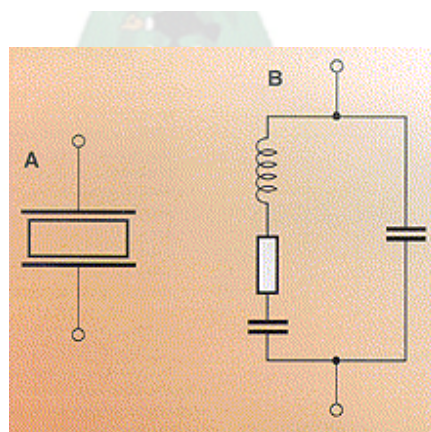


Fig. 30 A) representación de un cristal de cuarzo.

B) equivalente eléctrico de un oscilador piezoeléctrico

El comportamiento eléctrico del cuarzo se puede asemejar al de una inductancia, una resistencia y un condensador conectados en paralelo con otro condensador. Por lo tanto, es equivalente colocar un circuito con estos componentes que poner un cristal de cuarzo.

2.11 NORMA RS232.-

Es una de las normas más populares empleadas en la comunicación serie (su inserción en el PC incremento su popularidad). Fue desarrollada en la década de los 60 para gobernar la interconexión de terminales y MODEM. Está patrocinada por la EIA (Asociación de Industrias Eléctricas).

2.11.1 VELOCIDAD DE TRANSMISIÓN.-

La velocidad de transmisión de datos es expresada en bits por segundo o baudios. El baudio es un concepto más general que bit por segundo. El primero queda definido como el número de estados de la señal por segundo, si sólo existe dos estados (que pueden ser representados por un bit, que identifica dos unidades de información) entonces baudio es equivalente a bit por segundo. Baudio y bit por segundo se diferencian cuando es necesario más de un bit para representar más de dos estados de la señal.

La velocidad de transmisión queda limitada por el ancho de banda, potencia de señal y ruido en el conductor de señal.

- 75 baudios
- 110 baudios
- 150 baudios
- 300 baudios
- 600 baudios
- 1200 baudios
- 2400 baudios
- 4800 baudios
- 9600 baudios
- 19200 baudios

2.11.2 LÍNEAS O CANALES DE COMUNICACIÓN.-

Se pueden establecer canales para la comunicación de acuerdo a tres técnicas, siempre tomando al microprocesador o microcontrolador como referencia (transmisor) y al periférico como destino (receptor):

- Simplex
- Semi duplex (Half duplex)
- Totalmente duplex (Full duplex)

2.11.2.1 Simplex:

En ella la comunicación serie usa una dirección y una línea de comunicación. Siempre existirá un transmisor y un receptor, no ambos. La ventaja de este sistema consiste en que es necesario sólo un enlace a dos hilos. La desventaja radica en que el extremo receptor no tiene ninguna forma de avisar al extremo transmisor sobre su estado y sobre la calidad de la información que se recibe. Esta es la razón por la cual, generalmente, no se utiliza.

2.11.2.2 Semi duplex:

La comunicación serie se establece a través de una sola línea, pero en ambos sentidos. En un momento el transmisor enviará información y en otro recibirá, por lo que no se puede transferir información en ambos sentidos de forma simultánea. Este modo permite la transmisión desde el extremo receptor de la información, sobre el estado de dicho receptor y sobre la calidad de la información recibida por lo que permite así la realización de procedimientos de detección y corrección de errores.

2.11.2.3 Full duplex:

Se utilizan dos líneas (una transmisora y otra receptora) y se transfiere información en ambos sentidos. La ventaja de este método es que se puede transmitir y recibir información de manera simultánea.

La mayoría de los dispositivos especializados para la comunicación pueden transferir información tanto en full duplex como en half duplex (el modo simplex es un caso especial dentro de half duplex).

2.11.3 MODOS DE TRANSMISIÓN.-

Existen dos modos básicos para realizar la transmisión de datos y son:

- Modo asíncrono.
- Modo síncrono.

Las transmisiones asíncronas son aquellas en que los bits que constituyen el código de un carácter se emiten con la ayuda de impulsos suplementarios que permiten mantener en sincronismo los dos extremos. En las transmisiones síncronas los caracteres se transmiten consecutivamente, no existiendo ni bit de inicio ni bit de parada entre los caracteres, estando dividida la corriente de caracteres en bloques, enviándose una secuencia de sincronización al inicio de cada bloque.

2.12 CONECTORES DB9.-

DB9 macho y hembra: Ambos conectores son totalmente compatibles entre sí y existen adaptadores para pasar de un conector a otro:



Fig. 31 Conector DB9 hembra y macho

2.12.1 DESCRIPCIÓN DE TERMINALES DEL CONECTOR DB9.-

- **TXD** (*Transmit Data, transmisión de datos*): Señales de datos que se transmiten.
- **RXD** (*Receive Data, recepción de datos*): Señales de datos transmitidos.

- **DTR** (*Data Terminal Ready, terminal de datos preparado*): Señal que indica que está conectado, generalmente en "0" indica que esta listo para transmitir o recibir.
- **DSR** (*Data Set Ready, dispositivo preparado*): Señal que indica que el dispositivo está en modo de transmisión de datos.
- **RTS** (*Request To Send, petición de envío*): Señal que notifica la disposición de datos para enviar. Se emplea en líneas semiduplex para controlar la dirección de transmisión.
- **CTS** (*Clear To Send, preparado para transmitir*): Señal indicando que puede transmitirle datos.
- **CD** (*Carrier Detect, detección de portadora*): Señal que ha detectado la señal portadora enviado por un modem remoto o que la línea telefónica está abierta.
- **RI** (*Ring Indicator, timbre o indicador de llamada entrante*): Señal indicando que está recibiendo una llamada por un canal conmutado.
- **GND** (*System Ground ó Signal Ground, masa de señal*): Señal de tierra

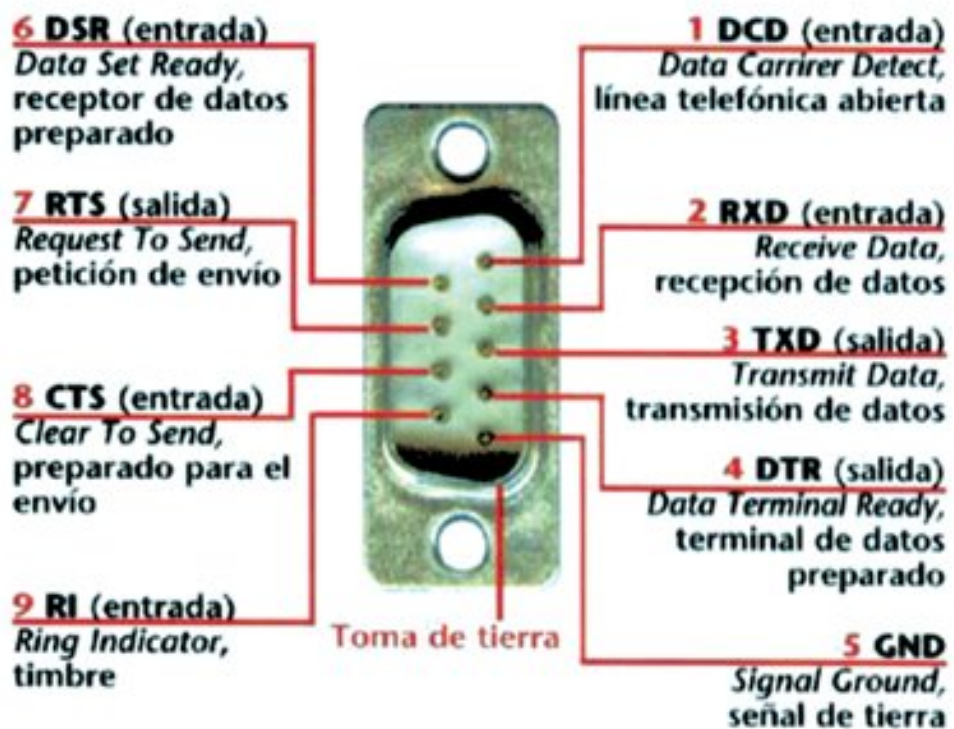


Fig. 32 Diagrama de pines del conector DB9

2.13 CIRCUITO INTEGRADO MAX232.-

Este chip permite adaptar los niveles RS232 y TTL, permitiendo conectar un PC con un microcontrolador. Sólo es necesario este chip y 4 condensadores electrolíticos. El esquema es el siguiente:

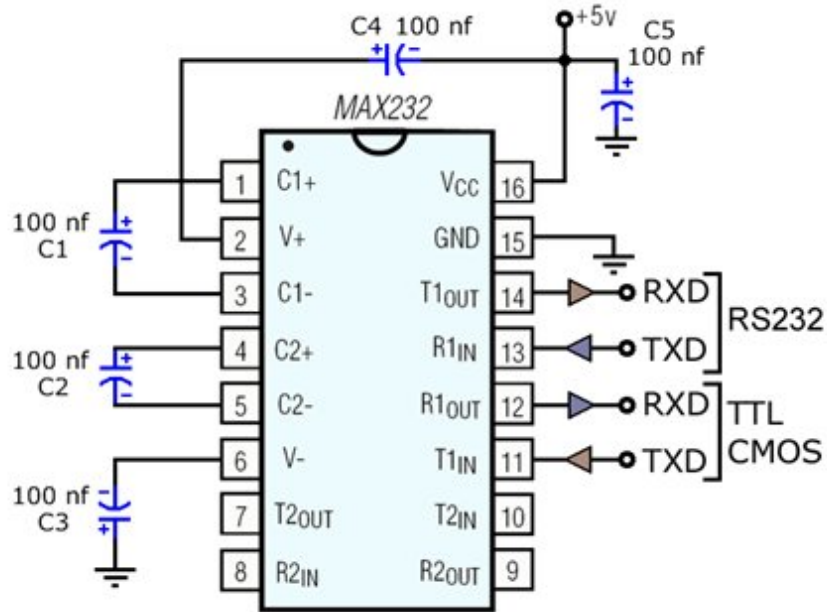


Fig. 33 Conexión mínima RS232/TTL con un MAX232

2.13.1 FUNCIONAMIENTO.-

El circuito integrado lleva internamente 2 convertidores de nivel de TTL a RS232 y otros 2 de RS232 a TTL con lo que en total podremos manejar 4 señales del puerto serie del PC, por lo general las mas usadas son; TXD, RXD, RTS, CTS, estas dos últimas son las usadas para el protocolo handshaking pero no es imprescindible su uso. Para que el MAX232 funcione correctamente debemos poner unos condensadores externos, todo esto lo podemos ver en la siguiente figura en la que solo se han cableado las líneas TXD y RXD que son las más usualmente usadas para casi cualquier aplicación.

En el MAX232 todos los condensadores deben ser de 1 microfaradio para llegar hasta 120 Kbps o de 100 nanofaradios para llegar hasta 64 Kbps. Para el MAX232A los condensadores han de ser de 100 nanofaradios y se consiguen hasta 200 Kbps.

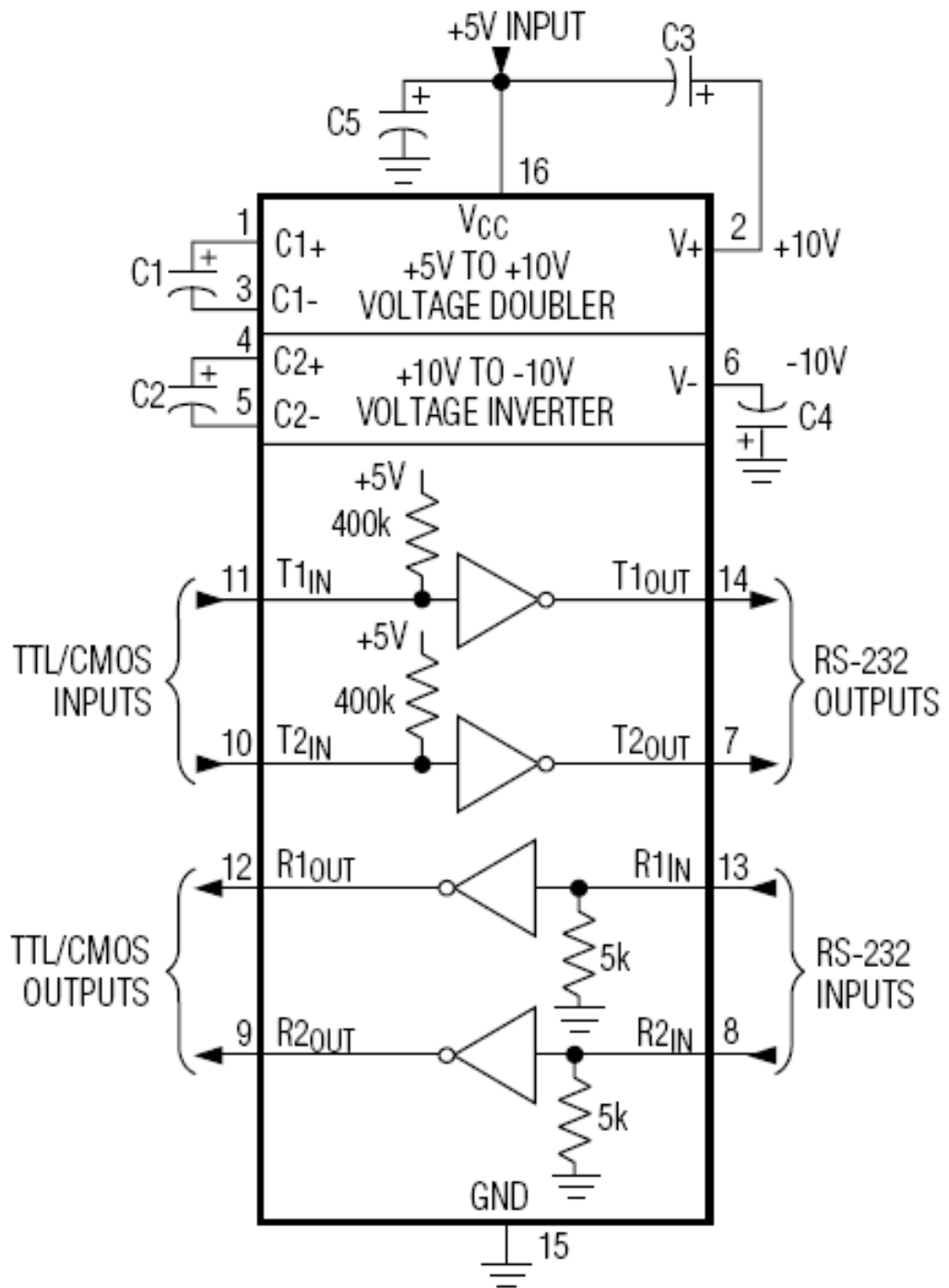


Fig. 34 Circuitería interna del chip MAX232

2.13.2 USOS.-

Este integrado es usado para comunicar un microcontrolador o sistema digital con un PC o sistema basado en el estándar RS232.

2.13.3 CARACTERÍSTICAS A +5V, CONDENSADORES DE 100 nF.-

- **Vcc:** de 4,5v a 5,5v.
- **Consumo:** 4 mA (15 mA con carga a la salida de 3 Kohm).
- **Entradas compatibles TTL y CMOS.**
- **Tensión de entrada máxima RS232:** +/- 30v.
- **Tensión de Salida RS232:** +/- 15v.
 - Tensión de salida típica de +/-8v con carga nominal de 5 Kohm en RS232.
- **Resistencia entrada RS232:** 5 Kohm (a masa).
- **Resistencia entrada TTL/CMOS:** 400 Kohm (a positivo).
- **Las entradas se pueden dejar al aire.**
 - Entrada TTL al aire, se considera un "0" al invertirse en la salida.
 - Entrada RS232 al aire, se considera un "1" al invertirse en la salida.
- **Salidas cortocircuitables continuamente:**
 - Salida RS232: +/- 22 mA.
 - Salida TTL/CMOS: a masa -10 mA, a positivo +30 mA.
- **Data Rate:** 200 Kbps (mín. 116 Kbps).

2.13.4 CONEXIÓN DE UN MICROCONTROLADOR AL PUERTO SERIE DEL PC.-

Para conectar el PC a un microcontrolador por el puerto serie se utilizan las señales TXD, RXD y GND. El PC utiliza la norma RS232, por lo que los niveles de tensión de las patillas están comprendidos entre +15 y -15 voltios. Los microcontroladores normalmente trabajan con niveles TTL (0-5v). Es necesario por tanto intercalar un circuito que adapte los niveles:

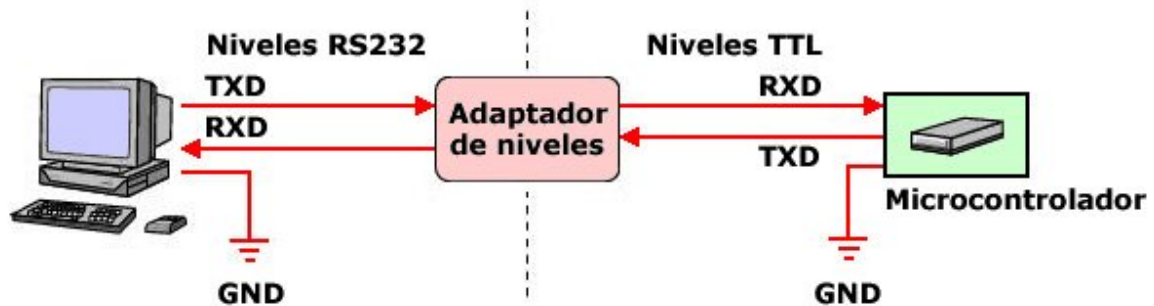


Fig. 35 Conexión entre la PC y el microcontrolador

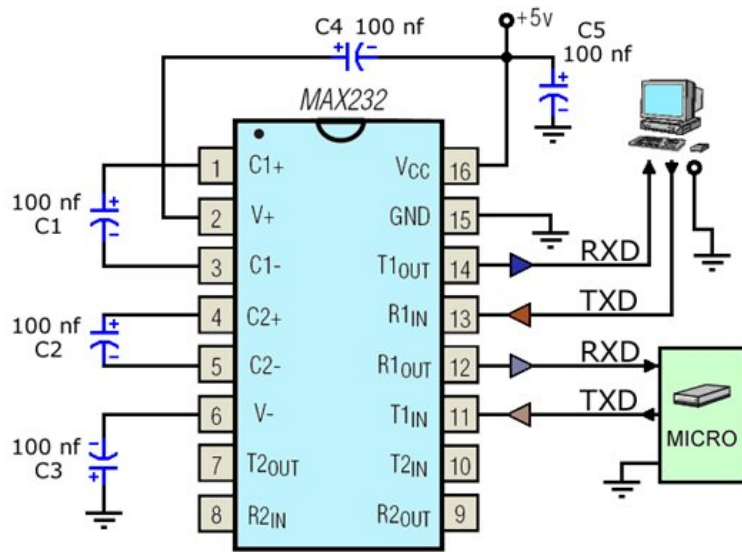


Fig. 36 Conexión de un microcontrolador con un PC mediante MAX232

2.13.5 CONVERSOR RS232 A TTL SIN MAX232.-

Todos sabemos que a la hora de conectar un PIC a la PC utilizar el MAX232 es el paso mas directo y fácil. Sólo cinco capacitores y éste integrado estará funcionando bien. Pero tomando en cuenta que hoy por hoy un MAX232 puede salirnos incluso mas caro que el mismísimo PIC micro decidimos desarrollar una alternativa válida que lo reemplace.

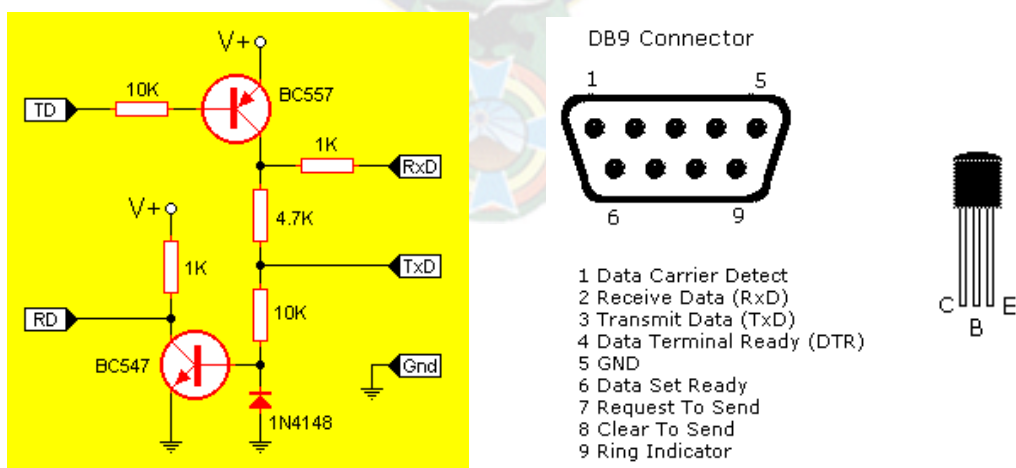


Fig. 37 Conversor RS232 a TTL

El circuito de arriba utiliza la propia corriente del puerto COM de la PC para generar los símbolos del RS232. Los pines marcados como TxD, RxD y Gnd corresponden al conector RS232 de la PC (ver conexionado) mientras que los pines marcados como RD y TD van directamente a microcontrolador. Podemos observar el pinout de los transistores.

2.14 CABLE DE CONEXIÓN.-

Para realizar la conexión entre el PC y un microcontrolador circuito podemos usar diferentes alternativas. Una manera es utilizar un cable serie macho-hembra no cruzado, y en el circuito un conector hembra DB9 para circuito impreso.



Fig. 38 Cable serie macho-hembra

En la placa de circuito impreso donde se encuentra el PIC y donde se colocará el conector DB9 hembra sería conveniente realizar la interconexión entre patillas que se describe en la siguiente figura.

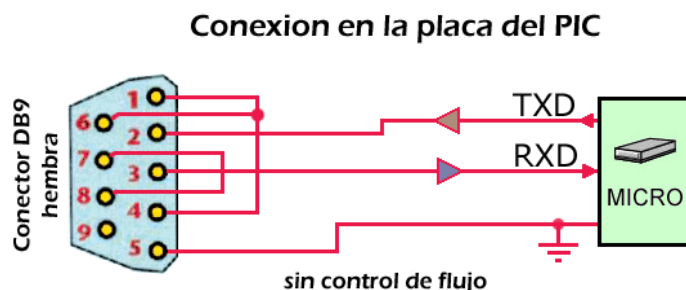


Fig. 39 Conexión del conector DB9 hembra con el PIC

Las conexiones que presenta la figura garantizan que cualquier programa de comunicación acepte la transmisión del PIC, si bien se realizará sin control de flujo. La salida DTR (patilla 4, Terminal de Datos Preparado) entrega señal a la entrada DCD (patilla 1, Detección de Portadora) y a la entrada DSR (patilla 6, Dispositivo Preparado). Por otro lado la salida RTS (patilla 7, Petición de Envío), entrega señal a la entrada CTS (patilla 8, Preparado para el Envío). Esta configuración no es necesaria ni para Hyperterminal de Windows ni para TerminalTOB



CAPÍTULO III

MARCO PRÁCTICO

3.1 HERRAMIENTAS SOFTWARE.-

3.1.1 COMPILADOR: CCS PCWH V4.068.-

Algunas características que presenta el compilador PCW CCS y que hacen de él una buena opción para elegirlo como compilador de C para programar Microcontroladores PIC. Algunas de esas características son:

- Al compilar genera un código máquina muy compacto y eficiente.
- Se integra perfectamente con MPLAB y otros simuladores/emuladores como PROTEUS para el proceso de depuración.
- Incluye una biblioteca muy completa de funciones pre compiladas para el acceso al hardware de los dispositivos (entrada/salida, temporizaciones, conversor A/D, transmisión RS-232, bus I2C...., etc.
- Incorpora drivers para dispositivos externos, tales como pantallas LCD, teclados numéricos, memorias EEPROM, conversores A/D, relojes en tiempo real, etc.(los drivers son pequeños programas que sirven de interfaz entre los dispositivos hardware y nuestro programa).
- Permite insertar partes de código directamente en Ensamblador, manteniendo otras partes del programa en C.



Fig. 40 Icono del compilador PIC C Compiler

3.1.1.1 Características del lenguaje C para este compilador.-

El lenguaje C estándar es independiente de cualquier plataforma. Sin embargo, para la programación de microcontroladores es necesario disponer de determinados comandos que se refieran a partes específicas de su hardware, como el acceso a memoria, temporizadores, etc. Por este motivo, además de los comandos, funciones y datos del lenguaje ANSI C, el compilador PCW incluye bibliotecas que incorporan determinados comandos que no son estándar, sino específicos para la familia de microcontroladores PIC. Éstos son básicamente de dos tipos: directivas del preprocesador y funciones pre compiladas.

3.1.1.2 Utilidades adicionales.-

El entorno PCW incluye, además del IDE y del compilador, una serie de utilidades adicionales con las que se amplían las posibilidades de éste, y que se encuentran en los menús View y Tools de la barra de menús, veamos algunas de ellas:

- Monitor del puerto serie: Consiste en un terminal que monitoriza la entrada y la salida del puerto serie del computador.
- Selección de dispositivos (Device Selection Tool): Esta utilidad consta de una base de datos con los dispositivos que puede programar el compilador, incluyendo todas sus características hardware, de manera que se puede emplear para buscar aquellos dispositivos que cumplan una serie de propiedades comunes.
- Editor de dispositivos (Device Editor): Este programa también emplea la base de datos de dispositivos, y permite editar los componentes para modificar sus características hardware, así como añadir nuevos dispositivos o eliminar algunos de ellos.
- Conversor numérico: Esta utilidad realiza conversiones entre los tipos de datos unsigned, signed, hex y float.
- Extracción de datos de calibrado: Esta opción permite leer los datos de calibración existentes en la memoria de programa de un determinado dispositivo. Estos datos contienen información particular de cada microcontrolador a su salida de fábrica, y se refieren a posibles problemas especiales que pudieran haber tenido lugar durante el

desarrollo y fabricación. Mediante esta opción es posible leer estos datos y grabarlos en un fichero .H o .C que incorporará una directiva #ROM para dicho dispositivo, con lo que cada vez que se programe el microcontrolador se incluirán estos datos de calibrado.

- Desensamblador: Esta opción lee un fichero en Código máquina y lo traduce a su equivalente en Ensamblador, con lo que se podría insertar este código en un programa en C, mediante las directivas #ASM y #ENDASM.

3.1.2 VISUAL BASIC 6.0.-

Visual Basic es un lenguaje de programación dirigido por eventos, desarrollado por el alemán Alan Cooper para Microsoft. Este lenguaje de programación es un dialecto de BASIC, con importantes agregados. Su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitara la creación de interfaces gráficas y, en cierta medida, también la programación misma.

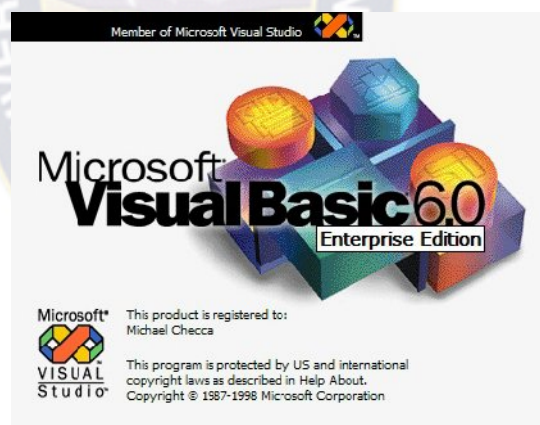


Fig. 41 Microsoft Visual Basic 6.0

En 2001 Microsoft propuso abandonar el desarrollo basado en la API Win32 y pasar a un framework o marco común de librerías, independiente de la versión del sistema operativo, .NET Framework, a través de Visual Basic .NET (y otros lenguajes como C Sharp (C#) de fácil transición de código entre ellos); fue el sucesor de Visual Basic 6. Si bien Visual Basic es de propósito general, también permite el desarrollo de aplicaciones de bases de datos usando Data Access Objects, Remote Data Objects, o ActiveX Data Objects.

Visual Basic (Visual Studio) contiene un entorno de desarrollo integrado o IDE que incluye un editor de textos para edición del código, un depurador, un compilador (y enlazador) y un constructor de interfaz gráfica o GUI.

3.1.2.1 CARACTERISTICAS.-

El compilador de Visual Basic 6.0 genera código que requiere librerías de enlace dinámico DLL para que funcione, en algunos casos llamada MSVBVMxy.DLL (acrónimo de "MicroSoft Visual Basic Virtual Machine x.y", donde x.y es la versión) y en otros VBRUNXXX.DLL ("Visual Basic Runtime X.XX"). Estas DLL proveen las funciones implementadas en el lenguaje, conteniendo rutinas en código ejecutable que son cargadas bajo demanda. Además existe un gran número de bibliotecas DLL, que facilitan el acceso a la mayoría de las funciones del sistema operativo y también la integración con otras aplicaciones.

En el IDE de Visual Basic se puede ejecutar el programa en desarrollo, "al vuelo" o en el modo intérprete (en realidad pseudo-compila el programa muy rápidamente y luego lo ejecuta), y también se permite la generación del programa en código ejecutable (exe). Tal programa generado en disco puede luego ser ejecutado fuera del ambiente de programación (incluso en modo stand alone, dependiendo de los requisitos de DLL's), aunque será necesario que las librerías DLL requeridas se encuentren instaladas en el sistema para su apropiada ejecución.

Visual Basic provee soporte para empaquetado y distribución, es decir, permite generar un módulo instalador que contiene el programa ejecutable y las bibliotecas DLL necesarias para él. Con ese módulo la aplicación generada se distribuye y puede ser instalada en cualquier equipo (con sistema compatible).

Así como bibliotecas DLL, hay numerosas aplicaciones de terceros que disponen de variadas funciones y mejoras para Visual Basic, incluyendo también para empaquetado y distribución.

3.1.2.2 ENTORNO DE DESARROLLO.-

Existe un único entorno de desarrollo para Visual Basic, desarrollado por Microsoft: **Microsoft Visual Basic** x.0 para versiones desde la 1.0 hasta la 6.0, (con las diferencias entre

las versiones desde la 1.0 (MS-DOS/Windows 3.1) hasta la 3.0 (16 bits, Windows 3.1/95) y las de la 4.0 (16/32 bits, Windows 3.1/95/NT) hasta la 6.0 (32 bits, Windows 9x/Me/NT/2000/XP/2003 server).

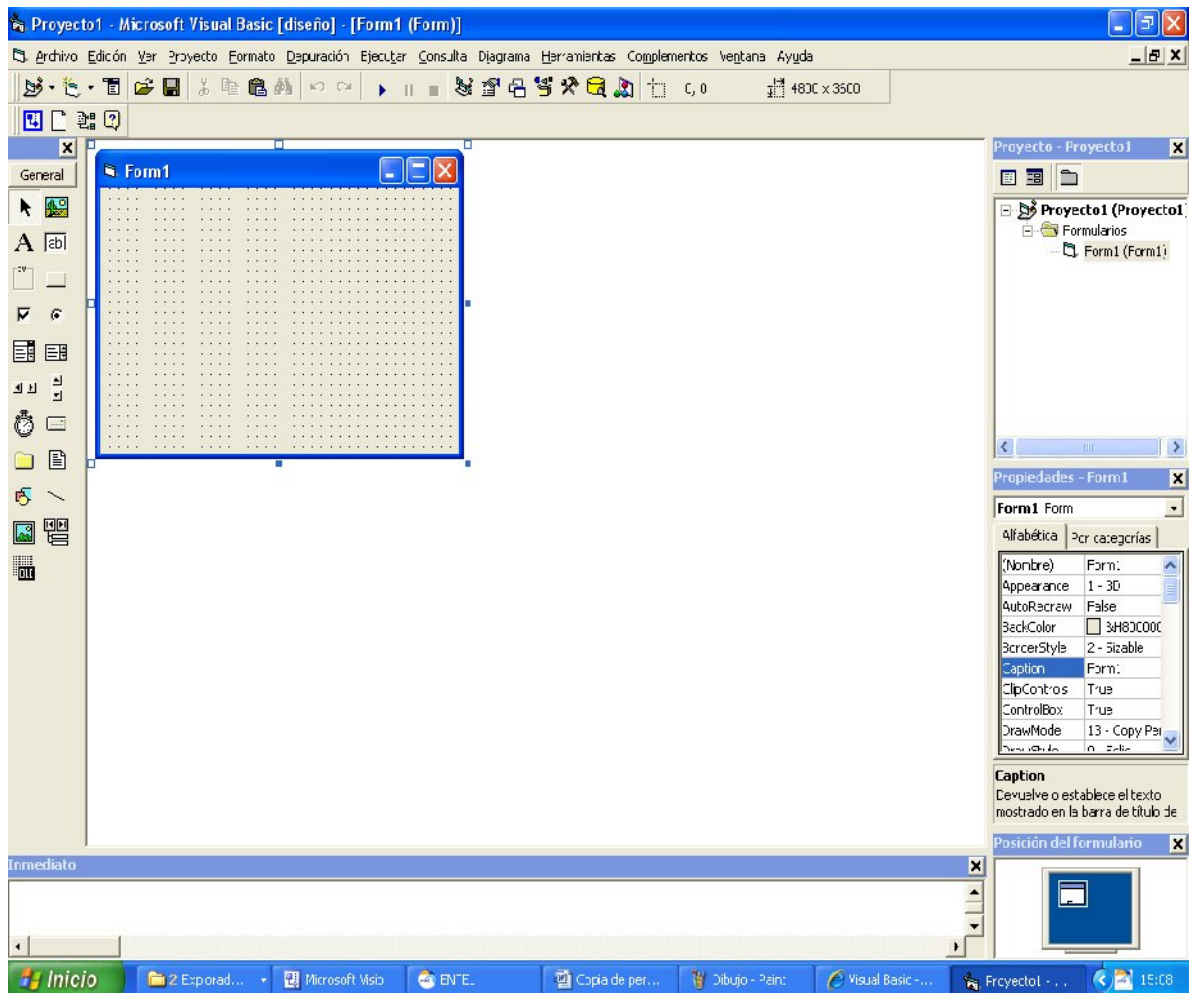


Fig.42 Entorno de trabajo del Visual Basic 6.0

Su entorno de desarrollo es muy similar al de otros lenguajes e IDE's.

- Se compone principalmente de una barra de herramientas y menú, que se pueden personalizar con prácticamente la totalidad de los comandos del IDE, a necesidad.
- El *espacio de trabajo* incluye y muestra todas las ventanas del proyecto, las vistas del código de los módulos y objetos, y los controles con los que se compondrán las ventanas de la aplicación.

Se pueden agregar todo tipo de controles de terceros, y hay una gran cantidad de ellos que se proveen con la instalación de Visual Basic 6.0, vienen embebidos dentro de archivos de extensión OCX.

- El panel lateral derecho contiene dos vistas principales:
 - El **Explorador de Proyectos**, que muestra todos los elementos que componen el proyecto o grupos de proyectos (formularios, interfaz de controles, módulos de código, módulos de clase, etc.)
 - El **Panel de propiedades**, donde se muestran todos los atributos de los controles de los formularios o la información de módulos clase y formularios, entre muchos otros.
- La **Ventana inmediato**, por defecto se encuentra en la parte inferior, aunque puede no estar visible; utilizar Ctrl+G para mostrar la ventana. Esta ventana resulta una herramienta muy útil a la hora de depurar el programa o incluso de realizar pruebas rápidas, ya que permite imprimir mensajes de texto desde el código y ejecutar sentencias y comandos simples (solo sentencias que se puedan representar, en una sola línea).

3.1.3 PROTEUS.-

Proteus es una compilación de programas de diseño y simulación electrónica, desarrollado por Labcenter Electronics que consta de los dos programas principales: Ares e Isis, y los módulos VSM y Electra.



Fig. 44 Icono del programa ISIS 7 Profesional

El Programa ISIS, **Intelligent Schematic Input System** (*Sistema de Enrutado de Esquemas Inteligente*) permite diseñar el plano eléctrico del circuito que se desea realizar con

componentes muy variados, desde simples resistencias, hasta alguno que otro microprocesador o microcontrolador, incluyendo fuentes de alimentación, generadores de señales y muchos otros componentes con prestaciones diferentes. Los diseños realizados en Isis pueden ser simulados en tiempo real, mediante el módulo VSM, asociado directamente con ISIS.

3.1.3.1 EL MÓDULO VSM.-

Una de las prestaciones de Proteus, integrada con ISIS, es **VSM**, el **Virtual System Modeling** (*Sistema Virtual de Modelado*), una extensión integrada con ISIS, con la cual se puede simular, en tiempo real, con posibilidad de más rapidez; todas las características de varias familias de microcontroladores, introduciendo nosotros mismos el programa que controlará el microcontrolador y cada una de sus salidas, y a la vez, simulando las tareas que queramos que lleve a cabo con el programa. Se pueden simular circuitos con microcontroladores conectados a distintos dispositivos, como motores, lcd's, teclados en matriz, etc. Incluye, entre otras, las familias de PIC's PIC10, PIC12, PIC16, PIC18, PIC24 y dsPIC33. ISIS es el corazón del entorno integrado PROTEUS.

3.1.4 VIRTUAL SERIAL PORT DRIVER.-

El conductor virtual de los puertos seriales (VSPD) es un driver de dispositivo de Windows. Usándolo se crea dos puertos virtuales de COM del cuento por entregas en su sistema que sea conectado virtualmente el uno al otro. Por ejemplo, después de VSDP de instalación, se tendrá el puerto serial virtual COM5 y puerto serial virtual COM6 agregado al sistema, y todos los datos enviados por otros usos a COM6 llegarán a COM5. Así pues, el uso, que COM5 que escucha conseguirá todos los datos, envió por el otro uso a COM6 y viceversa.



Fig. 45 Programa virtual serial port driver

3.1.5 SHOW MY PC.-

Es una **aplicación open source**; que permite **controlar un ordenador remotamente**, de una manera **sencilla** y sin necesidad de configurar ni instalar nada (es **portable**).



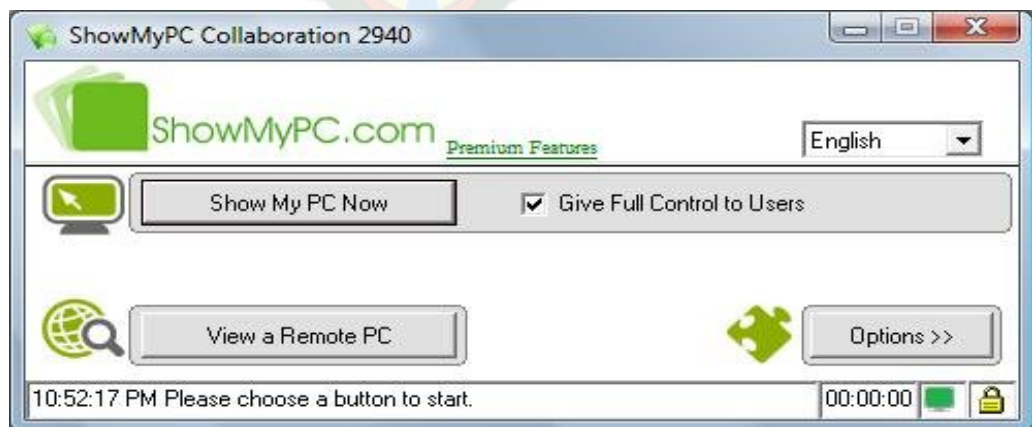
Fig. 46 Icono del programa showmyPC

Por ejemplo, si deseamos acceder a un **ordenador** vía **Internet**, entonces con solo ejecutar el programa en los dos **ordenadores** tienes control total de la **PC remota**.

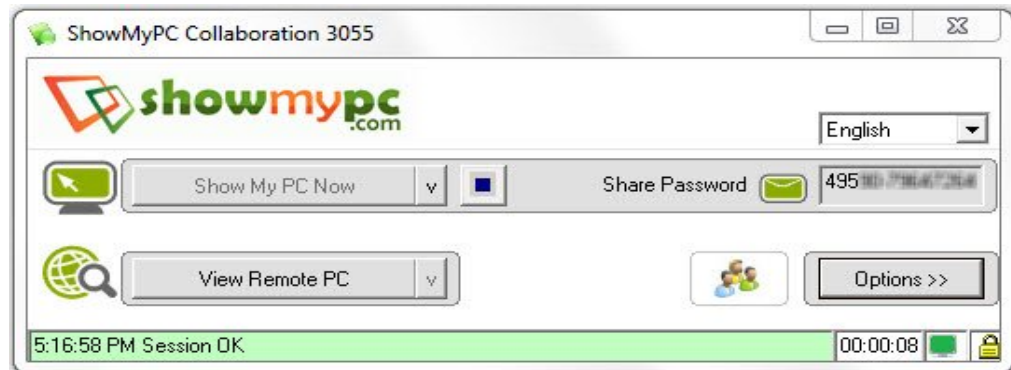
No se requiere **ningún IP**, ya que el **programa** les asigna una **contraseña** a cada uno al activar la opción “Mostrar Mi PC ahora” en cada sesión (Esta contraseña cambia cada vez que inicias **Windows**). Ahora cuando deseemos reparar un **ordenador remotamente** a alguien, podremos dar asistencia con sólo enviarle un archivo a la otra persona (de solo 1 MB!!), y ya podremos conectarnos.

3.1.5.1 Pasos a seguir para realizar la conexión.-

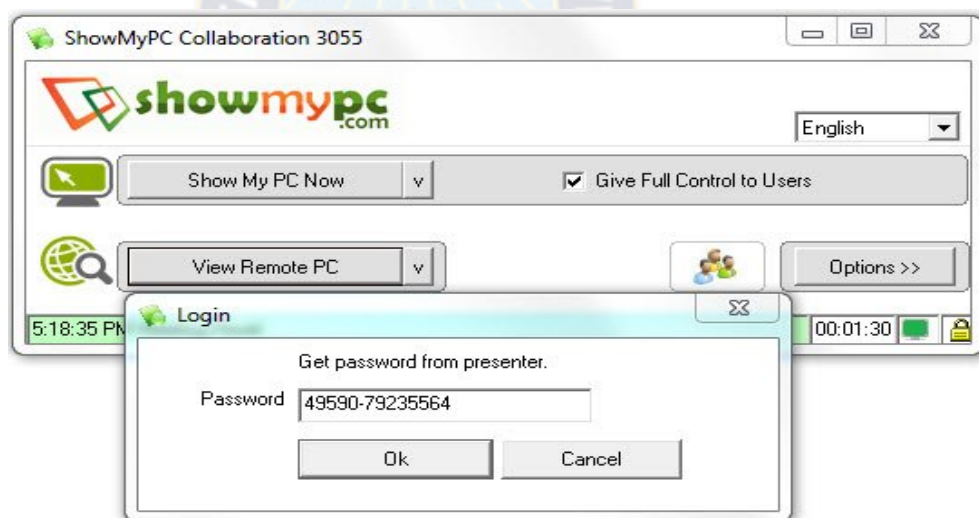
- **Primero:** *En la computadora que se va a compartir el escritorio:* para compartir el escritorio, ejecuta ShowMyPCSSH.exe Tarda unos segundos y aparece la siguiente pantalla:



- **Segundo:** Se hace un click en el botón "Show My PC to remote users" para generar un password y hace tu computadora accesible remotamente.



- **Tercero:** En la computadora que va a ver el escritorio remoto, ejecuta el mismo programa y se realiza click donde dice "View remote PC", pide un password que es el que genero el que va a compartir su escritorio:

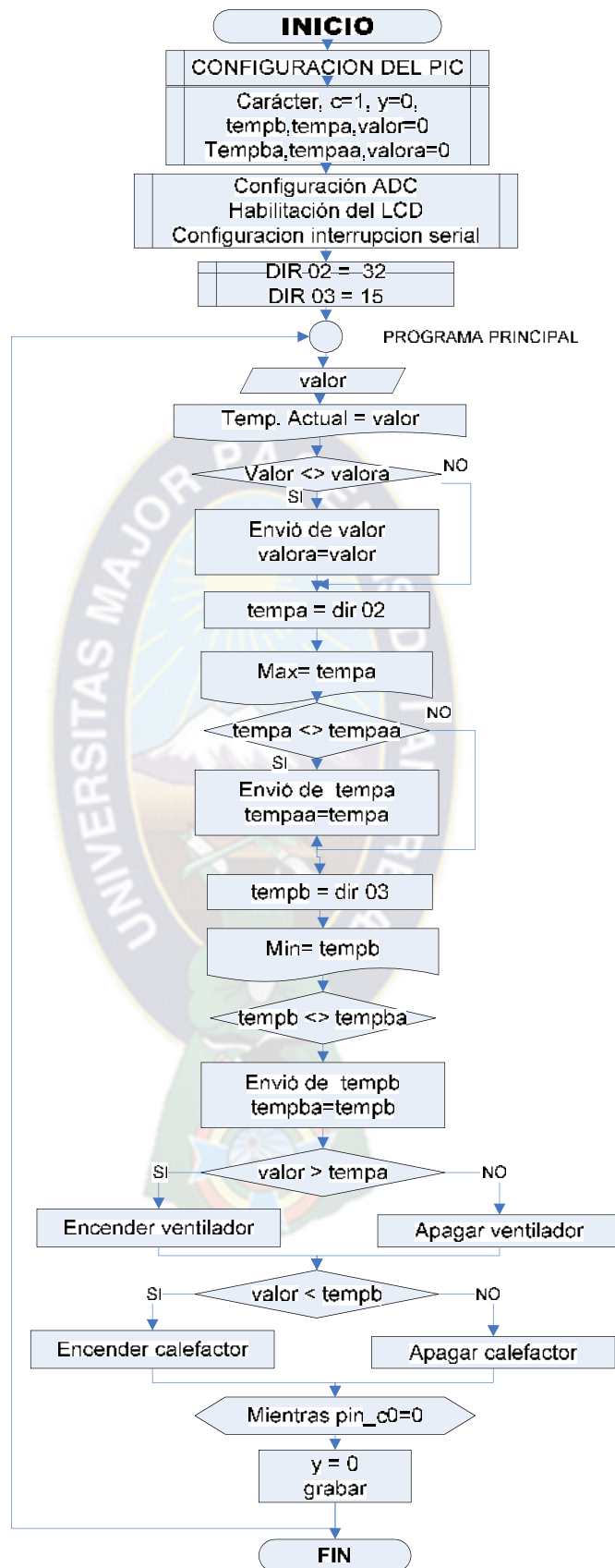


3.2 DISEÑO DEL SOFTWARE.-

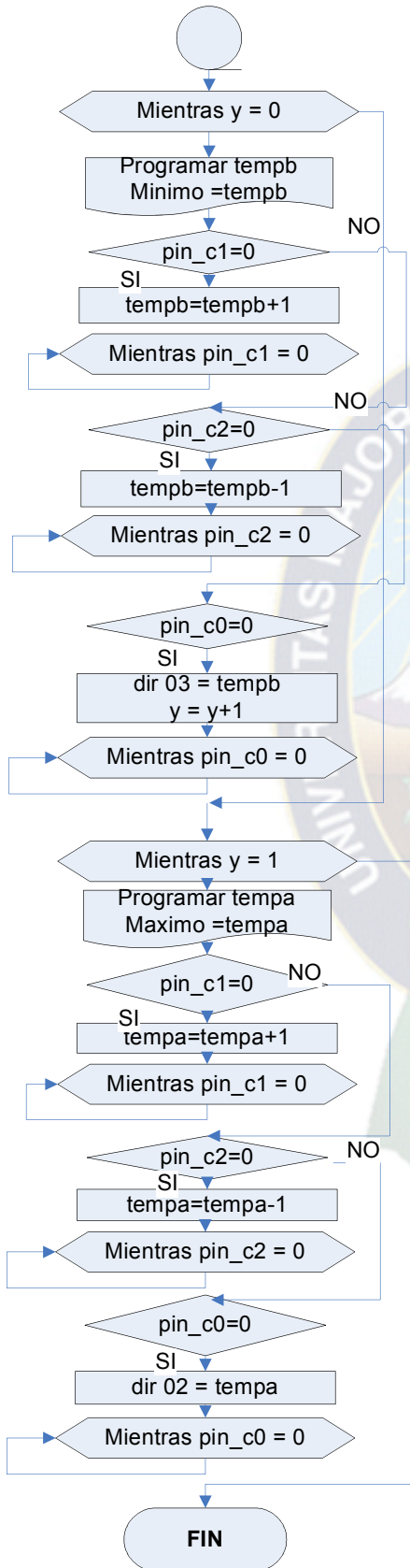
3.2.1 SISTEMA DE CONTROL DE TEMPERATURA.-

El programa del microcontrolador para realizar el sistema de control de temperatura se lo realizó en el compilador CCS PCWH V4.068.

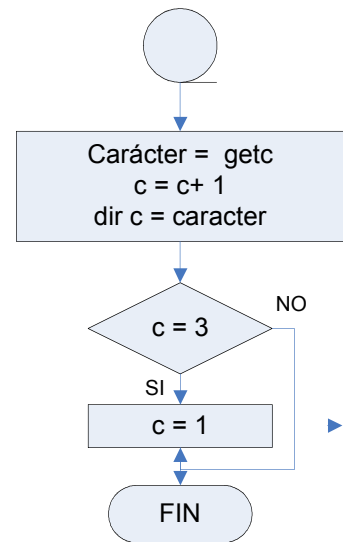
Para entender con mayor claridad el programa del PIC16F877A se realizaron los diagramas de flujos correspondientes los cuales son los siguientes:



SUBROUTINA GRABAR



INTERRUPCIÓN SERIAL



Y luego contamos con el código fuente realizado con el compilador de PIC C:

<pre> #include <16f877a.h> #device adc=10 #FUSES hs,nowdt #use delay (clock =11059200) #use RS232(BAUD=9600,BITS=8,PARITY=N, XMIT=PIN_C6,RCV=PIN_C7) #include <lcd.c> char character; int c=1; int y=0; int16 tempb,tempa,res=0,valor=0; int16 tempba,tempaa,valora=0; #int_RDA void RDA_isr(void) { if(kbhit()) { character=getc(); c=c+1; write_eeprom(c,character); if (c==3) { c=1; } output_e(character); delay_ms(500); } } void grabar(void) { while(y==0) { lcd_putc("\fPROGRAMAR TEMP\n"); printf(lcd_putc,"Minimo=%2ld oC",tempb); if (input(pin_c1)==0) { tempb++; delay_ms(25); while (input(pin_c1)==0){} } if (input(pin_c2)==0) { tempb--; delay_ms(25); while (input(pin_c2)==0){} } delay_ms(25); if (input(pin_c0)==0) { </pre>	<pre> write_eeprom(03,tempb); delay_ms(25); y++; while (input(pin_c0)==0){} } } while(y==1) { delay_ms(100); lcd_putc("\fPROGRAMAR TEMP\n"); printf(lcd_putc,"Maximo=%2ld oC",tempa); if (input(pin_c1)==0) { tempa++; delay_ms(25); while (input(pin_c1)==0){} } if (input(pin_c2)==0) { tempa--; delay_ms(25); while (input(pin_c2)==0){} } if (input(pin_c0)==0) { write_eeprom(02,tempa); delay_ms(100); y++; while (input(pin_c0)==0){} } } lcd_putc("\f"); } void main() { setup_adc_ports(AN0); setup_adc(ADC_CLOCK_INTERNAL); lcd_init(); enable_interrupts(GLOBAL); enable_interrupts(INT_RDA); res=read_eeprom(00); if (res==0) { write_eeprom(00,01); write_eeprom(02,32); write_eeprom(03,15); } } </pre>
---	---

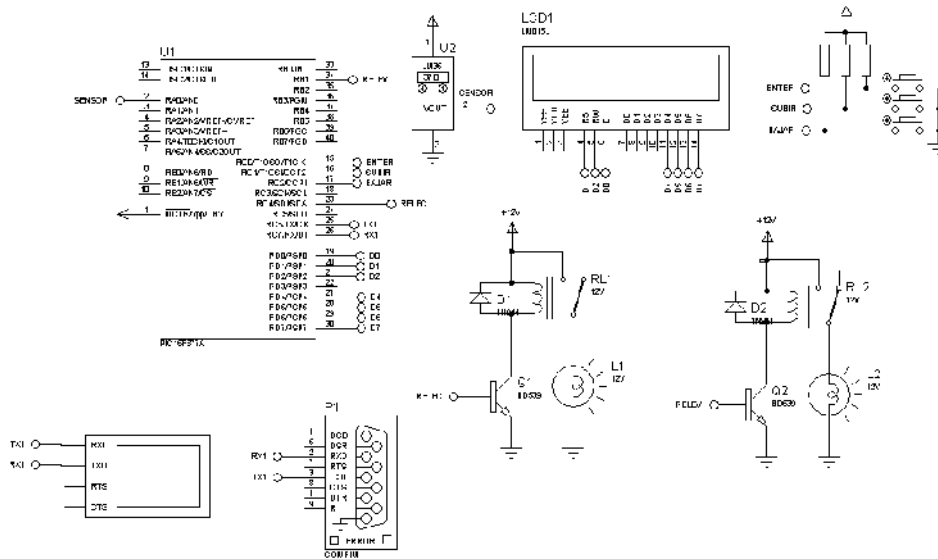
```

for (;;)
{
    set_adc_channel(0);
    delay_ms(10);
    valor=read_adc();
    valor=valor/2;
    lcd_gotoxy(1,1);
    printf(lcd_putc,"T.actual=%2ld oC",valor);
    if (valor!=valora){
printf("%c", valor);
        delay_ms(500);
        valora=valor;}
    tempa=read_eeprom(02);
    lcd_gotoxy(1,2);
    printf(lcd_putc,"Max=%2ld",tempa);
    if (tempa!=tempaa)
    { printf("%c",2);
        delay_ms(500);
        printf("%c",tempa)
        delay_ms(500);
        tempaa=tempa;}
    tempb=read_eeprom(03);
    lcd_gotoxy(9,2);
    printf(lcd_putc,"Min=%2ld",tempb);
    if (tempb!=tempba){
        printf("%c",3);
        delay_ms(500);
printf("%c",tempb);
        delay_ms(500);
        tempba=tempb; }
    if(valor>tempa){
        output_high(pin_c4);
        printf("%c",4);
        delay_ms(500); }
    else{
        output_low(pin_c4);
        printf("%c",5);
        delay_ms(500);}
    if(valor<tempb) {
        output_high(pin_B1);
        printf("%c",6);
        delay_ms(500); }
    else{
        output_low(pin_B1);
        printf("%c",7);
        delay_ms(500); }
    if (input(pin_c0)==0) {
        while (input(pin_c0)==0){}
        y=0;
        grabar();
    }
}
}
}

```

La simulación del programa se lo realizó usando el programa Proteus versión 7.7 SP2, y para realizar la simulación de comunicación serial entre el PIC 16F877A y la interfaz de usuario se utilizó el programa Virtual Serial Port Driver. El esquema es el siguiente:

CONTROL DE TEMPERATURA CON PIC16F877A



3.2.2 INTERFAZ DE USUARIO.-

La interfaz de usuario se realizó el diseño utilizando el programa de Visual Basic 6.0 y su código es el siguiente:

<pre> Dim c As Integer Dim d As Integer Dim str As String Private Sub CMD1_Click() End End Sub Private Sub CMD2_Click() txtx.SelStart = 0 txtx.Text = "" txtx.SetFocus End Sub </pre>	<pre> Private Sub Command1_Click() n = Val(Text4.Text) mserial.Output = Chr(n) Call tiempo n = Val(Text5.Text) mserial.Output = Chr(n) Call tiempo Text4.Text = "" Text5.Text = "" End Sub </pre>
--	---

<pre> Private Sub Form_Load() msserial.PortOpen = True c = 0 d = 0 End Sub Private Sub Form_Unload(Cancel As Integer) msserial.PortOpen = False End Sub Private Sub msserial_OnComm() If msserial.CommEvent = comEvReceive And msserial.InBufferCount <> 0 Then str = msserial.Input If Asc(str) = 1 Then c = 1 End If If Asc(str) = 2 Then c = 2 End If If Asc(str) = 3 Then c = 3 End If If Asc(str) = 4 Then Picture3.Visible = True Picture2.Visible = False c = 0 End If </pre>	<pre> If Asc(str) = 5 Then Picture3.Visible = False Picture2.Visible = True c = 0 End If if Asc(str) = 6 Then Picture4.Visible = True Picture5.Visible = False c = 0 End If If Asc(str) = 7 Then Picture4.Visible = False Picture5.Visible = True c = 0 End If Select Case c Case 1 txtrx.Text = Asc(str) Case 2 Text1.Text = Asc(str) Case 3 Text2.Text = Asc(str) End Select End If End Sub Private Sub tiempo() For x = 1 To 1000 For y = 1 To 100 DoEvents Next y </pre>
---	--

El diseño de interfaz de usuario es el siguiente:



3.3 DISEÑO DEL HARDWARE.-

3.3.1. DISEÑO DE PLACAS.-

Para la realización del diseño de placas se utilizó el programa ARES y los esquemas son los siguientes:

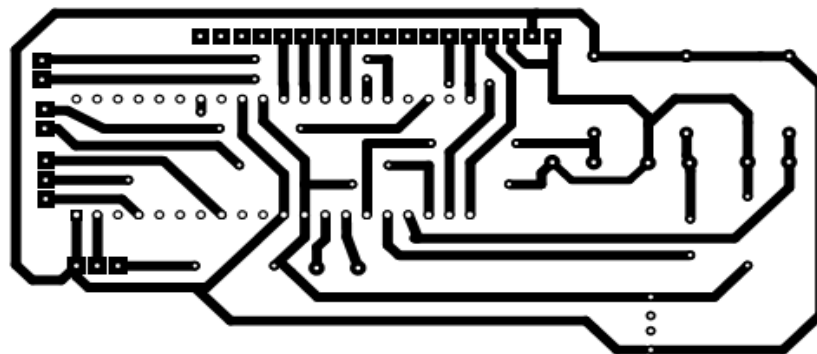


Fig. 47 Diseño para el quemado de placa para el control de temperatura

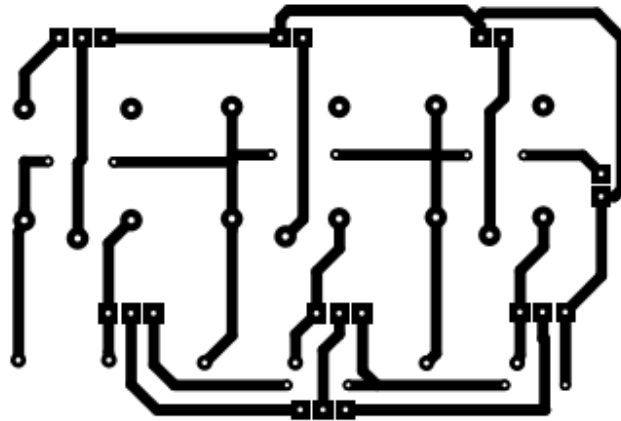


Fig. 48 Diseño para el quemado de placa del control de potencia

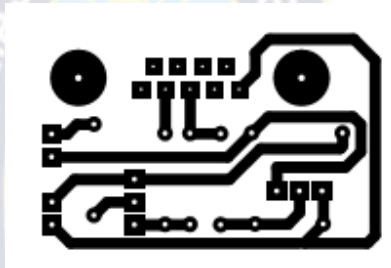


Fig. 49 Diseño para el quemado de placa para la comunicación serial

3.3.2 QUEMADO DE PLACAS.-

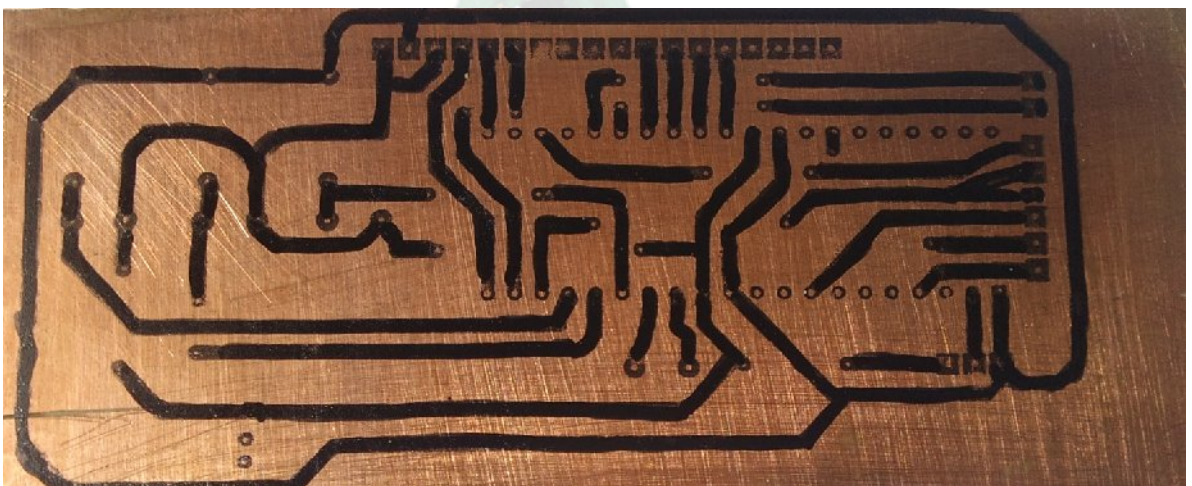


Fig. 50 Circuito impreso para el control de temperatura

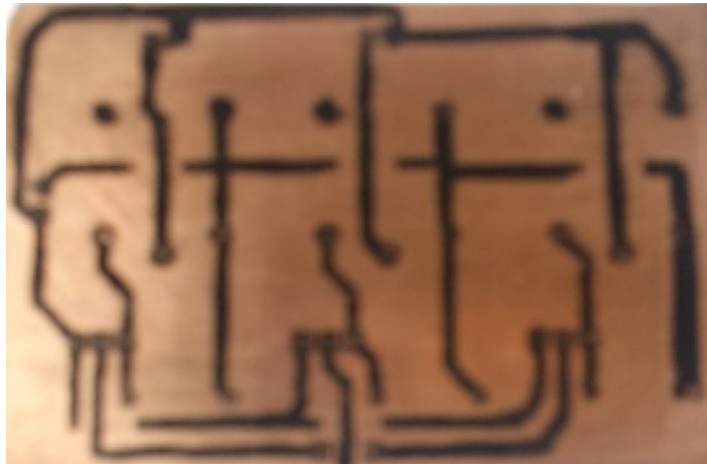


Fig. 51 Circuito impreso para el control de potencia

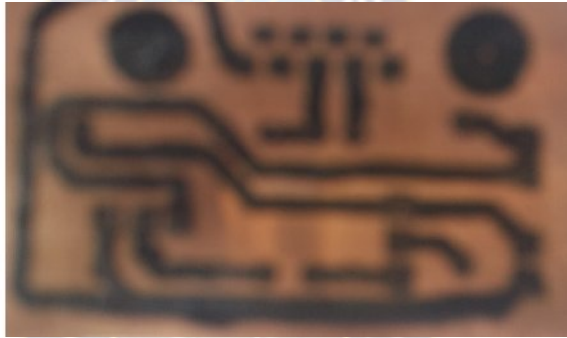


Fig. 52 Circuito impreso para la comunicación serial

3.3.3.- ACABADO DE PLACAS.-

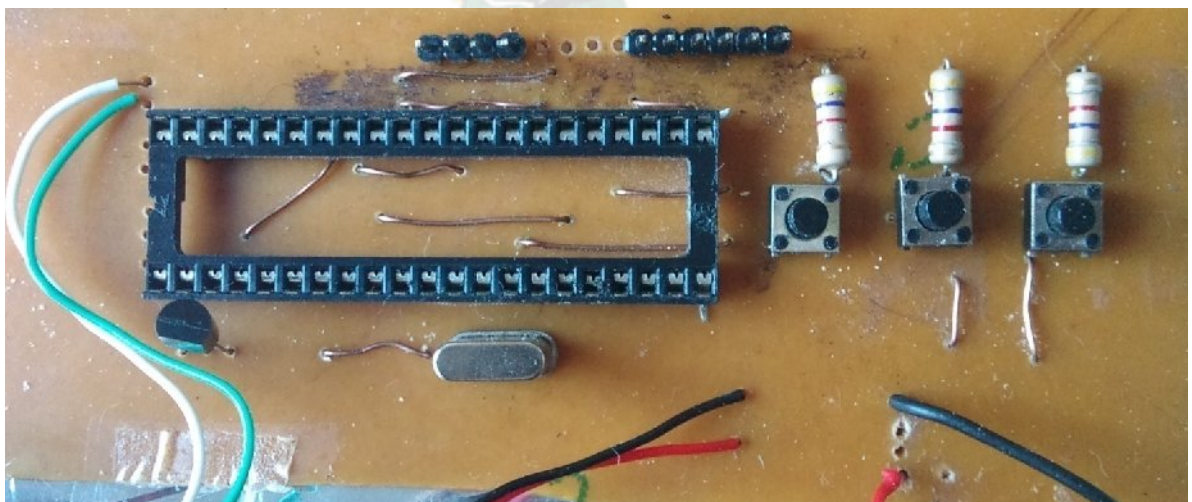


Fig. 53 Circuito para el control de temperatura

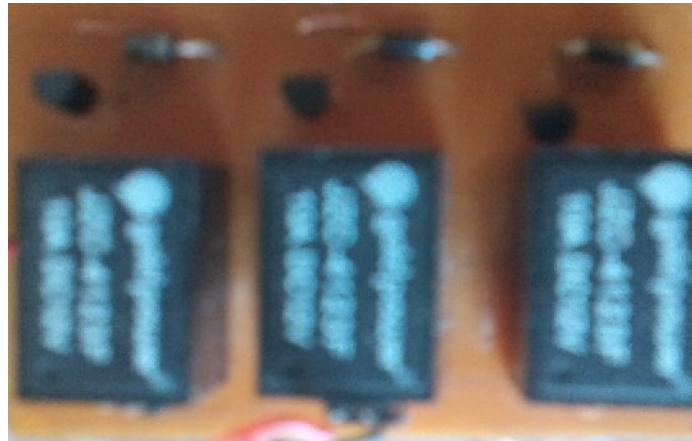


Fig. 54 Circuito para el control de potencia

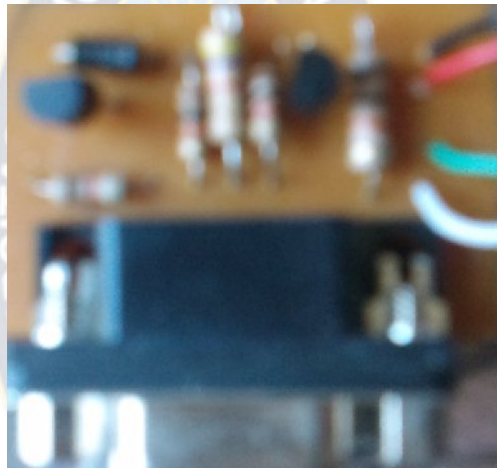
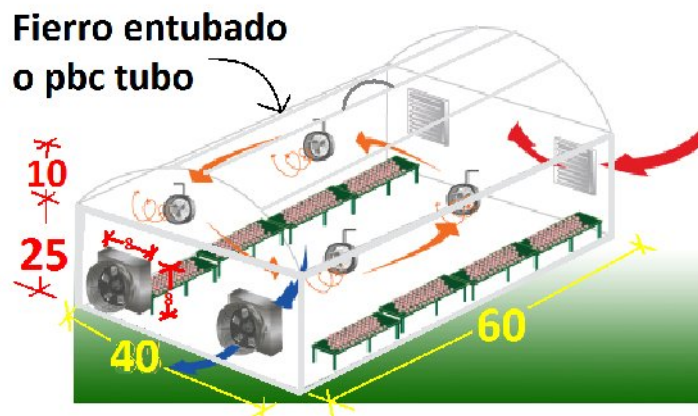


Fig. 55 Circuito para la comunicación serial

3.3.4 DISEÑO DE LA MAQUETA.-



CAPITULO V

4.1 CONCLUSIONES.-

- Para las mediciones de la temperatura ambiente se utilizó un solo sensor, el sensor LM35.
- Para realizar el control de temperatura se utilizó el PIC16F877A por ser un microcontrolador que posee entradas analógicas.
- La visualización para realizar la configuración manual de los rangos de temperatura, se utilizó un LCD.
- Se utilizó el compilador CCS PCWH V4.068, para realizar el código fuente y generar el archivo .hex para el microcontrolador.
- Se programo el microcontrolador PIC16F877A utilizando el WinPic800.
- Para realizar la interfaz de usuario se utilizó el programa Visual Basic, y de esta manera realizar la comunicación serial entre la PC y el microcontrolador.
- Debido a las características del sensor de temperatura, el sistema operará de acuerdo a los rangos establecidos.
- Para controlar los dispositivos encargados de mantener el invernadero entre los rangos de temperatura, se utilizó un circuito de potencia.
- Para simular las condiciones de un invernadero y para probar los dispositivos a controlar, se realizo una maqueta en forma de invernadero.
- Se utilizó el software especializado show my PC y de esta manera se controló remotamente el control de temperatura.

BIBLIOGRAFIA.

- Colección Serie “Conocimientos científico-tecnológicos para el desarrollo de equipos e instrumentos”
Instituto Nacional de Educación Tecnológica
Centro Nacional de Educación Tecnológica CeNET
- Curso Básico de PIC 16F877A: Autores: Raúl Peralta Meza y Carlos Quiñones Quispe
- Microcontroladores PIC 16F877A Diseño práctico de aplicaciones Autores: José M. Angulo Usategui, Susana Romero Yesa e Ignacio Angulo Martínez.
- Compilador C CCS y simulador Proteus para microcontroladores PIC de Eduardo Garcia Breijo.
- Visual Basic 6.0 como de ser...! de Juan Jose Castañeda León.

Páginas web consultadas

- <http://www.cambio climatico\Causas y Efectos del Efecto Invernadero Calentamiento Global.htm>
- <http://www.coolers\Extractores Eólicos - Características, Rendimientos.htm>
- <http://www.ds1307\El Reloj de Tiempo Real DS1307.htm>
- <http://www.características de los termistores.htm>
- <http://www.Compilador PCW CCS.htm>
- [http://www.programas\Proteus \(electrónica\).Wikipedia, la enciclopedia libre.htm](http://www.programas\Proteus (electrónica).Wikipedia, la enciclopedia libre.htm)
- <http://www.ShowMyPC Programa para control remoto\Reparacion de ordenadores.mht>