

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN TECNOLÓGICA

CONSTRUCCIÓN DE UN PROTOTIPO QUE CONTROLE LA TEMPERATURA, EL ALMACENAMIENTO Y EXTRACCIÓN DE LÍQUIDOS EN UN TANQUE

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN
ELECTRÓNICA Y TELECOMUNICACIONES**

**ROLANDO VLADIMIR CABRERA PEÑAHERRERA
SYLVANA MARITZA JÁCOME CADENA**

DIRECTOR: ING. ALCIVAR COSTALES

Quito, Agosto 2006

DECLARACIÓN

Nosotros, Sylvana Maritza Jácome Cadena y Rolando Vladimir Cabrera Peñaherrera, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad intelectual, por su Reglamento y por la normatividad institucional vigente.

Rolando Vladimir Cabrera Peñaherrera

Sylvana Maritza Jácome Cadena

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Rolando Vladimir Cabrera Peñaherrera y Sylvana Maritza Jácome Cadena, bajo mi supervisión.

Ing. Alcívar Costales
DIRECTOR DE PROYECTO

PRESENTACIÓN

La ejecución de ciertos procesos en forma manual o utilizando tecnologías ambiguas son en muchas empresas técnicas que si bien permiten el desenvolvimiento normal de la empresa no cumplen con ciertos parámetros de calidad y efectividad en el resultado final del producto o proceso.

Sin embargo, los avances tanto de la ciencia como de la tecnología han permitido implementar una gran variedad de herramientas y equipos que dan lugar a la automatización de muchas tareas aplicadas en muchos campos de grandes empresas, generando así, alternativas prácticas que permiten el ahorro de tiempo, dinero y con resultados excepcionales; lastimosamente al tratarse de tecnologías de punta son inversiones costosas que la pequeña empresa ecuatoriana no está dispuesta a afrontar, aunque ello signifique que no se pueda hacer uso de ese tipo de avances tecnológicos.

Así, en el control de la temperatura, almacenamiento y extracción de líquidos en un tanque mermado en un prototipo, no es más que la aplicación práctica de algunos de los recursos que la tecnología actual nos proporciona teniendo a nuestro favor muchas ventajas, proporcionando una alternativa a bajo costo y muy buenas prestaciones para el manejo y control de los líquidos.

La educación superior recibida en la Escuela de Formación Tecnológica de la Escuela Politécnica Nacional nos permite aplicar las técnicas necesarias para ejecución del prototipo para controlar los parámetros mencionados a través del uso de un dispositivo de control principal con sus respectivas adecuaciones implementadas de una forma óptima y contando con los recursos necesarios.

El prototipo está diseñado para controlar la temperatura, el almacenamiento y la extracción de líquidos en un tanque de una manera óptima y segura con un mínimo margen de error.

RESUMEN

En el **(Anexo A)** se encuentra el Plan de Proyecto donde se detallan las ideas que se llevaron a cabo para la realización de éste proyecto.

El Prototipo que se ha desarrollado para el control de la temperatura, el almacenamiento y la extracción de líquidos en un tanque, opera bajo el mando de un microcontrolador PIC (Peripheral Interface Controller)

La programación de este microcontrolador ha sido realizada mediante el diseño de diagramas de flujo en un entorno visual de alto nivel.

El programa desarrollado permite comandar a los sensores y actuadores que se hallan conectados a las líneas de entrada o salida del PIC respectivamente, los mismos que a través de los distintos circuitos permiten el control y monitoreo del nivel y temperatura del líquido en el tanque. Estos parámetros pueden ser visualizados desde un panel frontal en el que se encuentra una pantalla LCD destinada para dicho fin, además el panel frontal cuenta con un teclado que permite el ingreso de parámetros necesarios para el correcto funcionamiento del dispositivo.

Por medio de este dispositivo se puede obtener una medida correcta del nivel y temperatura del líquido y desde el MENÚ PRINCIPAL se puede extraer cantidades discretas del mismo, evitando la manipulación directa y sobre todo errores de medición, todos estos parámetros al ser controlados por medio de un panel brinda al usuario seguridad así como ahorro de tiempo y dinero.

CAPÍTULO 1

1.1 INTRODUCCIÓN

El medir el nivel de un líquido, conocer la temperatura a la que se encuentra, extraerlo de una manera segura y en cantidades discretas; no pueden ser tareas que se realicen utilizando métodos obsoletos, ya que al ser técnicas prácticamente manuales no brindan la precisión ni confiabilidad que muchos procesos requieren.

Este dispositivo es una buena opción sobretodo en aquellos procesos donde se utilicen líquidos de uso restringido o que sean peligrosos al momento de manipularlos y sobre todo donde se requiera de una buena resolución en las mediciones.

Muchas empresas a pesar de contar con instrumentos sofisticados, destinados para realizar el control de los parámetros inherentes a los líquidos, tienen el inconveniente de que estos son equipos costosos, de muy difícil reparación, y cuyos repuestos no son comunes en nuestro país. Afortunadamente contamos con las herramientas necesarias para proveer de una alternativa tecnológica que facilite dichos controles, a costos accesibles, de fácil reparación y rendimiento óptimo.

Por otro lado dentro del desarrollo de este modelo basado en la tecnología de los microcontroladores, hace que el circuito que se requiere para llevar a cabo el control de líquidos sea lo más pequeño posible optimizando espacio y ahorrando dinero, pero sobre todo facilitando su reparación, haciéndolo un instrumento tecnológico perfecto para aquellas empresas donde no cuenten con equipos que realicen este control y que estén especulando en la instalación de un instrumento con estas características.

1.2 ANTECEDENTES Y REFERENCIAS

Actualmente existen empresas ecuatorianas que utilizan métodos obsoletos o inadecuados para la manipulación de sustancias líquidas, y a pesar de existir varias alternativas en el mercado, por diversas razones no son asequibles para la pequeña empresa; así también, empresas que económicamente son solventes y capaces de implementar nuevas tecnologías buscan diferentes alternativas que les permitan optimizar sus recursos.

Así, es muy difícil encontrar en el mercado un solo dispositivo que se encargue de realizar las funciones de medición de nivel, de extracción de líquidos y de medición de la temperatura. Es mucho más común encontrar dispositivos individuales destinados únicamente a controlar uno de los parámetros antes mencionados. Esto hace que el sistema sea demasiado costoso ya que se debe comprar un dispositivo por cada parámetro a controlar, además de que el sistema se vuelve muy complejo de manejar, debido a que se tiene demasiados parámetros configurables que se debe recordar cada vez que se desee manejar los controles.

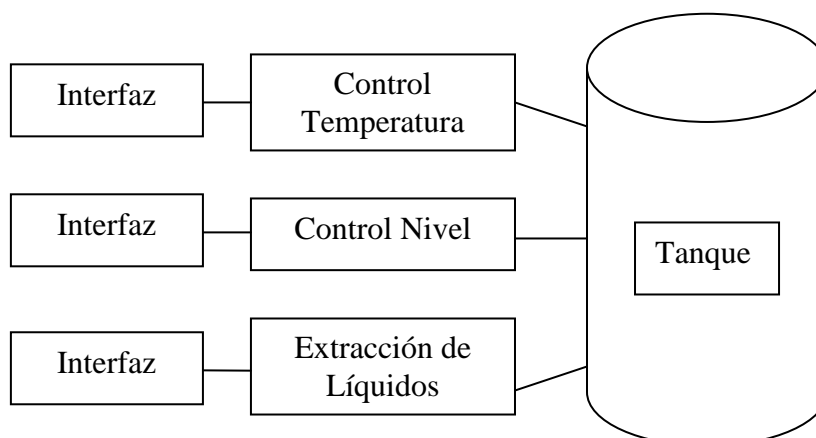


Figura 1.1.- Sistema de control individual a cada parámetro

Este tipo de sistemas son demasiado costosos y de muy difícil reparación porque contienen elementos que nos son comunes en el mercado y requiere que sean sistemas compatibles entre sí para lograr la perfecta integración en uno solo; además de que cada uno cuenta con un su propio grupo de circuitos haciéndolo un sistema demasiado complejo.

1.3 MARCO TEÓRICO

1.3.1 MICROCONTROLADOR PIC

1.3.1.1 Definición

Un microcontrolador PIC (Peripheral Interface Controller) es un computador completo, aunque de limitadas prestaciones, que está contenido en el chip de un circuito integrado y se designa a gobernar una sola tarea. En su memoria sólo reside un programa destinado a gobernar una aplicación determinada; sus líneas de entrada/salida soportan el conexionado de los sensores y actuadores del dispositivo a controlar. Una vez programado y configurado el microcontrolador solamente sirve para gobernar la tarea asignada.

1.3.1.2 Arquitectura

Un microcontrolador consta de varias secciones diferentes:

- La unidad aritmético-lógica: efectúa cálculos con números y toma decisiones lógicas
- Los registros: son zonas de memoria especiales para almacenar información temporalmente
- La unidad de control: decodifica los programas
- Los buses transportan información digital a través del chip
- La memoria local se emplea para los cálculos realizados en el mismo chip

1.3.2 SISTEMAS DE DESARROLLO

Se conoce como sistema de desarrollo a aquella herramienta de software que permite realizar el diseño del programa que controlara el comportamiento de algún dispositivo, en este caso el microcontrolador PIC.

Se han desarrollado todo tipo de lenguajes de programación para los microcontroladores, aunque los más usados son el ensamblador, Basic y el lenguaje C. Hemos encontrado una herramienta novedosa que permite programar

al PIC en un entorno completamente visual y de alto nivel. Esta herramienta toma el nombre de NIPLE.

1.3.2.1 Los Simuladores

Son programas soportados generalmente desde una PC (Computador Personal) que permiten la simulación de cualquier circuito virtualmente montado en su interfaz gráfica, permitiendo la respectiva ejecución y depuración del circuito.

1.3.2.2 El Grabador

Luego de que el programa que gobernará al PIC ha sido correctamente depurado se necesita de una herramienta que nos permita transportar el archivo HEX desde el computador hacia la memoria de programa del PIC físico. Esta herramienta se conoce como grabador, consta de una parte de software y otra de hardware.

1.3.3 CONCEPTOS BÁSICOS

1.3.3.1 Resistor

Cualquier elemento localizado en el paso de una corriente eléctrica sea esta corriente continua o corriente alterna y causa oposición a que ésta circule se llama resistencia o resistor.

Normalmente las resistencias se representan con la letra R y el valor de éstas se mide en Ohmios (Ω).



Figura 1.2.- Símbolo de una resistencia

1.3.3.2 Condensador

Los condensadores son componentes pasivos diseñados con el fin de almacenar energía electrostática o presentar una capacidad eléctrica determinada. Otra forma de definirlo sería la siguiente; componentes pasivos de dos terminales en

los que la intensidad que los atraviesa (aparentemente) es proporcional a la variación de tensión existente entre sus terminales respecto al tiempo. Su unidad de medida en el Sistema Internacional es el Faradio aunque por las limitaciones características de los mismos se usan distintos submúltiplos (micro, μ / nano, n / pico, p).

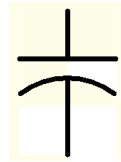


Figura 1.3.- Símbolo de un condensador

1.3.3.3 Diodo

Un diodo es un dispositivo que permite el paso de la corriente eléctrica en un solo sentido. De forma simplificada, la curva característica de un diodo (I-V) consta de dos regiones, por debajo de cierta diferencia de potencial, se comporta como un circuito abierto (no conduce), y por encima de ella como un circuito cerrado con muy pequeña resistencia eléctrica.

Debido a este comportamiento, se les suele denominar rectificadores, ya que son dispositivos capaces de convertir una corriente alterna en corriente continua.

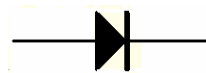


Figura 1.4.- Símbolo de un diodo

1.3.3.4 Diac

El DIAC es un dispositivo semiconductor de dos conexiones. Es un diodo bidireccional disparable que conduce la corriente sólo tras haberse superado su tensión de disparo, y mientras la corriente circulante no sea inferior al valor característico para ese dispositivo. El comportamiento es fundamentalmente el mismo para ambas direcciones de la corriente.



Figura 1.5.- Símbolo de un DIAC

1.3.3.5 Triac

El Triac es un dispositivo semiconductor que pertenece a la familia de los dispositivos de control por tiristores, es en esencia la conexión de dos tiristores en paralelo pero conectados en sentido opuesto y compartiendo la misma compuerta. El triac es un tiristor que tiene dos estados de funcionamiento: bloqueo y conducción. Conduce la corriente entre sus terminales principales en un sentido o en el inverso, por ello, es un dispositivo bidireccional.

Empieza a conducir entre los dos ánodos (MT1 y MT2) cuando se aplica una señal a la puerta (G).

Se puede considerar como dos tiristores en antiparalelo. Al igual que un tiristor, el paso de bloqueo al de conducción se realiza por la aplicación de un impulso de corriente en la puerta, y el paso del estado de conducción al de bloqueo por la disminución de la corriente por debajo de la intensidad de mantenimiento (I_H).

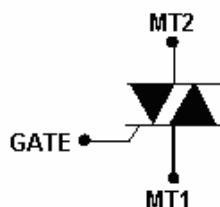


Figura 1.6.- Símbolo de un TRIAC

1.3.3.6 Optoacoplador

Un optoacoplador es un dispositivo que contiene una fuente de luz y un detector Fotografía-sensible separados una cierta distancia y sin contacto eléctrico entre ellos. La clave del funcionamiento de un optoacoplador está en el emisor que es un LED, y en el detector Fotografía-sensible a la salida.

El detector Fotografía-sensible puede ser de varios tipos, en este caso se utiliza un optoacoplador que tiene como detector Fotografía-sensible un FOTOGRAFÍA DIAC, que no es mas que el mismo DIAC pero activado por luz, es decir que este dispositivo solo conducirá una vez que se el elemento emisor (LED) sea activado.

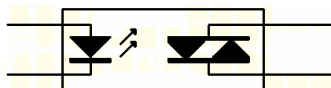


Figura 1.7.- Símbolo de un optoacoplador

1.3.3.7 El Amplificador Operacional

Es un dispositivo amplificador de la diferencia de sus dos entradas, con una alta ganancia, una impedancia de entrada muy alta, (mayor a $1\text{ M}\Omega$) y una baja impedancia de salida (de 8 a 20Ω), por lo que las corrientes de entrada son prácticamente nulas y que tiene la característica de poder entregar corriente relativamente alta según los datos del fabricante.

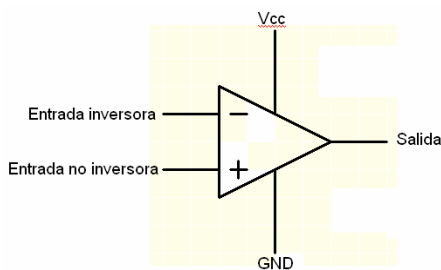


Figura 1.8.- Símbolo de un amplificador

1.3.4 NOCIONES BÁSICAS

1.3.4.1 Magnitudes físicas

Magnitud física es toda aquella propiedad de los sistemas físicos susceptible de ser medida y expresada mediante un número y una unidad, y con la cual se pueden establecer relaciones cuantitativas.

El Sistema Internacional de Unidades se basa en dos tipos de magnitudes físicas, fundamentales y derivadas.

La longitud, el tiempo; entre otras, pertenecen a las magnitudes físicas fundamentales, mientras que entre una de las magnitudes derivadas encontramos a la velocidad.

1.3.4.1.1 Distancia

Se denomina distancia a la longitud del camino más corto entre dos puntos, se la representa con la letra **d** y su unidad es el metro [m].

$$\text{distancia} = \text{velocidad} \times \text{tiempo}$$

Formula 1.1.- Fórmula para la distancia

1.3.4.1.2 Tiempo

Se denomina tiempo al periodo en el que tiene lugar una acción o acontecimiento.

1.3.4.1.3 Velocidad

En física, velocidad es la magnitud que expresa la variación de posición de un objeto por unidad de tiempo, o distancia recorrida por un objeto en la unidad de tiempo.

Para el cálculo de la velocidad con sus respectivas unidades tenemos:

$$v = \frac{d}{t} \left[\frac{m}{s} \right]$$

Formula 1.2.- Fórmula para calcular la velocidad

1.3.4.2 Velocidad del sonido en el aire

El sonido al viajar por un medio elástico que es el aire tiene una velocidad que responde a las variaciones del medio que lo trasmite, así tenemos la siguiente fórmula que permite conocer la velocidad del sonido en función de la temperatura:

$$\text{velocidad del sonido} = 331.6 \sqrt{1 + \frac{t(^{\circ}\text{C})}{273}} \left[\frac{m}{s} \right]$$

Formula 1.3.- Fórmula para la velocidad del sonido en el aire

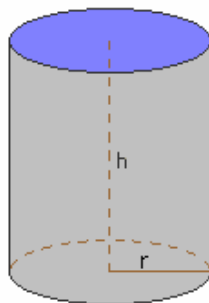
1.3.4.3 Cilindro Recto

1.3.4.3.1 Definición

Un cilindro recto es un cuerpo geométrico engendrado por un rectángulo al girar alrededor de uno de sus lados. El cilindro consta de bases, una superficie lateral que al abrirse da lugar al rectángulo.

1.3.4.3.2 Cálculo del volumen de un cilindro

Para calcular el volumen del cilindro recto se tiene la siguiente fórmula, cuyas unidades pueden ser metros cúbicos [m³] con sus respectivos múltiplos y submúltiplos, los cuales pueden ser transformados a litros [l] o galones



$$V = \pi(r^2)h$$

Formula 1.4.- Fórmula para el volumen de un cilindro

1.3.5 SENSORES

1.3.5.1 Definición

La medición de magnitudes mecánicas, térmicas, eléctricas y químicas se realiza empleando dispositivos denominados sensores y transductores. El sensor es un elemento sensible a los cambios de la magnitud a medir, como una temperatura, una posición o una concentración química. El transductor convierte estas mediciones en señales eléctricas, que pueden alimentar a instrumentos de lectura, registro o control de las magnitudes medidas.

Al dispositivo formado por los dos elementos un sensor y transductor es comúnmente designado como uno solo, ya sea como sensor o transductor dejando sin trascendencia su auténtica definición.

Los sensores y transductores pueden funcionar en ubicaciones alejadas del observador, así como en entornos inconvenientes o impracticables para los seres humanos.

Todos los instrumentos deben ser especificados de acuerdo a la aplicación, a un punto tal que aseguren la operación del proceso y que permita la estimación de sus costos.

1.3.6 ACTUADORES

1.3.6.1 Definición

Un actuador es un mecanismo por el cual un agente puede influir en su entorno. Dicho agente puede ser tanto artificial como otro tipo de ser autónomo (humano, animal, electrónico, etc.).

También un mecanismo que pone algo en acción autónoma es denominado actuador.

Algunos ejemplos de actuadores pueden ser:

- En humanos: Brazos, piernas, dedos...
- En robots: Motores de movimiento de brazos, piernas, etc.

En ingeniería, los actuadores son dispositivos que transforman una señal de entrada (eléctrica) por lo general en movimiento, aunque puede ser cualquier otro parámetro físico.

1.3.7 REGULADORES DE VOLTAJE

1.3.7.1 Definición

Los circuitos integrados reguladores, no son más que una fuente de alimentación estabilizada mediante un diodo zener, donde los típicos integrados reguladores de tensión tienen la entrada de la fuente de alimentación, tierra y la salida regulada.

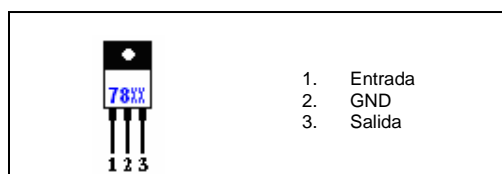


Figura 1.9.- Regulador de Voltaje 78XX

El "78", nos indica que es un regulador positivo; "XX" nos indica a que voltaje regulará.

Estos pueden ser; 7805, 7806, 7808, 7810, 7812, 7815, 7818, 7822 y 7824. Es importante también tomar en cuenta la corriente de cada regulador ya que algunos vienen para 0,1 [A] y otros hasta 3 [A], recalcando además que los reguladores 78XX trabajan con corrientes de hasta 1 [A].

CAPÍTULO 2

2.1 SOLUCIÓN DEL PROBLEMA

2.1.1. MICROCONTROLADOR Y HERRAMIENTAS EMPLEADAS

2.1.1.1 PIC 16F877

En el mercado existen diversas marcas y modelos de microcontroladores, tomando en cuenta las necesidades de este proyecto, así como la disponibilidad en el mercado, el costo, documentación, herramientas para la programación etcétera, se optó por trabajar con el microcontrolador PIC 16F877 del fabricante MICROCHIP.

Este modelo posee varias características que hacen a este microcontrolador un dispositivo muy versátil, eficiente y práctico para ser empleado en la aplicación que posteriormente será detallada.

Las características generales de este microcontrolador PIC se describen en la **Tabla 2.1**.

	Descripción	Características
Procesador	Núcleo	RISC, Arq. Harvard, 20 MHz. 5 MIPS
	Interrupciones	14 fuentes posibles de interrupción
	Reloj	Hasta 20 MHz
	Reset	Master Clear, Brown Out, Watchdog, Power On
	Instrucciones	35 instrucciones de 14 bits
Memoria	M. de programa FLASH	8 K palabras de 14 bits
	M. de datos RAM	368 registros de 8 bits
	M. de datos EEPROM	256 registros de 8 bits
	Pila	8 palabras de 13 bits
	M. de datos ext. EEPROM	Hasta 256 KBytes
Periféricos	Puertos programables de E/S	Hasta 33 bits, pueden ser usados por otros periféricos
	Timers/Counters	Dos de 8 bits y uno de 16 bits
	Puertos de captura/Comparación de datos	Dos de 8 bits
	Moduladores de ancho de pulso (PWM)	Dos de 8 bits
	Convertor Analógico/Digital de 10 bits	Con un MPX de 8 canales para 8 entradas diferentes
	Puerto serie síncronico	Configurable en modo SPI e IC ²
	USART	Para conexiones RS 232
	Parallel Slave Port	8 bits + 3 bits de control

Tabla 2.1.- Características generales del PIC 16F877

En el **Anexo B** encontraremos la hoja de datos del PIC 16F877 donde se puede observar con mayor detalle las características del mismo.

Teniendo como herramienta de control al PIC16F877 se ha pensado en integrar en un solo dispositivo, los elementos medidores de temperatura, medidor de nivel,

y uno que controle la extracción de líquidos, evitando así los problemas de interconexión entre varios sistemas, y a la vez optimizando el espacio y especialmente la cantidad de elementos que contiene todo el sistema, haciéndolo de más fácil reparación ya que cuenta con un solo elemento que controla todas las funciones del sistema.

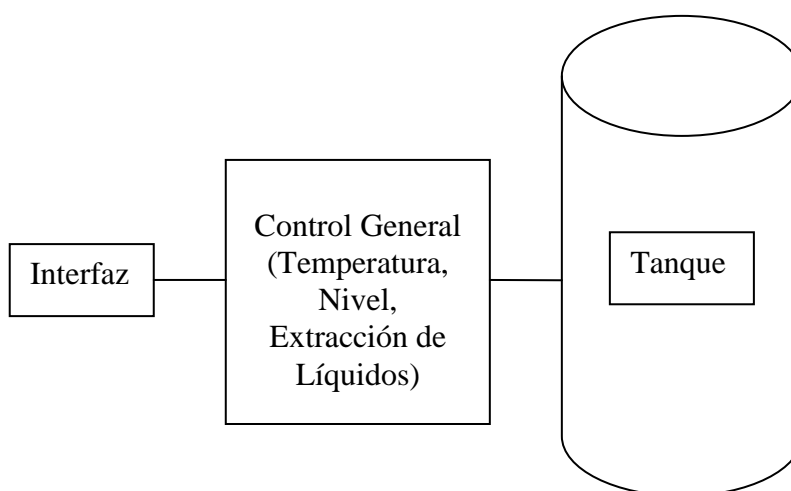


Figura 2.1.- Sistema de control general para todos los parámetros

Como se trata de integrar en uno solo sistema tenemos esencialmente 3 parámetros para controlar:

- El Nivel del líquido
- La Temperatura del líquido
- La Extracción segura del líquido

Los dispositivos externos como sensores, actuadores, entre otros; están conectados a través de diferentes circuiterías a las líneas de entrada/salida del PIC, la distribución de los mismos se la puede observar en la siguiente figura.

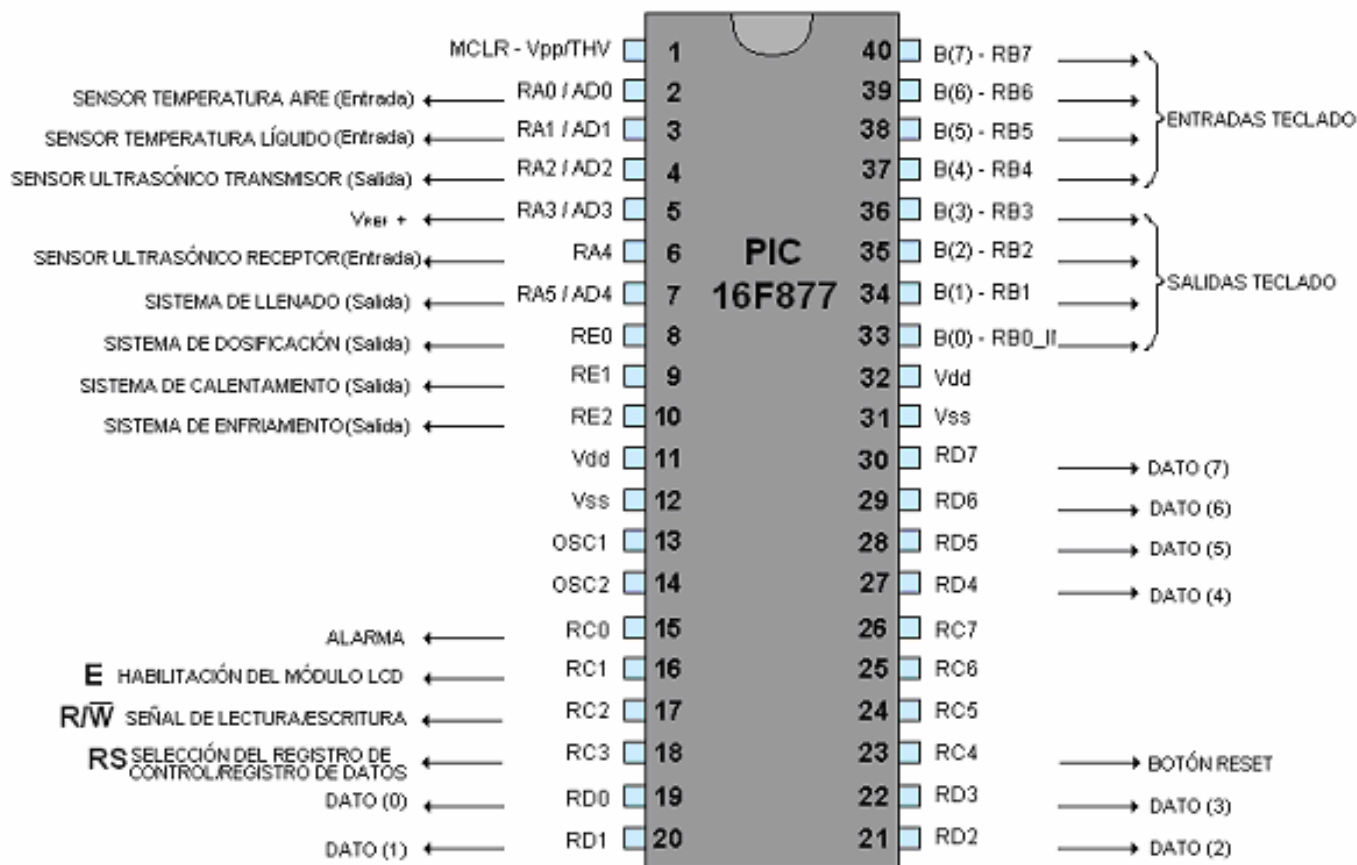


Figura 2.2.- Conexiones a las líneas de entrada y salida del PIC

2.1.1.2 Sistemas de Desarrollo NIPLE

Para la programación del dispositivo que controla todo el sistema se ha contado con una herramienta que debido a la gran cantidad de tareas que se requieren controlar facilitó mucho la tarea de programación, este software recibe el nombre de NIPLE.

NIPLE es un “*Entorno Visual de Desarrollo para Microcontroladores PIC*”¹, que permite programar los microcontroladores de la manera más rápida y sencilla a través del diseño de un diagrama de flujo, en un entorno completamente visual y de alto nivel, generando el código assembler adecuado con sólo hacer un clic.

¹ MANUAL DE USUARIO DE SOFTWARE “NIPLE”, <http://www.niplesoft..net>

En el **Anexo C** encontraremos una parte del manual que NIPLE entrega para el usuario, esto servirá para el mejor entendimiento del programa final que esta siendo destinado al control del dispositivo.

También es posible descargar una versión de demostración de este software desde el siguiente link, www.niplesoft.net, y se puede comprar en línea la versión completa.

2.1.1.3 Los Simuladores

En este proyecto usamos un simulador conocido como *ISIS*, que es capaz de simular la ejecución de programas para PIC's a partir de un diagrama donde se crea el circuito esquemático que hemos diseñado y carga el programa generado en *Assembler*. Gracias a la herramienta *MPWIN* integrada en esta misma herramienta, se genera el archivo codificado en hexadecimal (*.HEX) listo para grabar en el PIC.

Este simulador permite visualizar en el monitor de la PC el estado actual de todos los recursos que utiliza el PIC, además admite la ejecución paso a paso, puntos de parada, modificación de los contenidos de los registros, entre otras herramientas perfectas para la depuración del programa.

Este software tiene derechos reservados, y aunque su costo es un poco elevado se puede conseguir fácilmente en Internet versiones anteriores que ya han sido liberadas gratuitamente.

En la siguiente figura se puede apreciar el diagrama esquemático con el cual se hicieron las simulaciones para realizar las pruebas correspondientes.

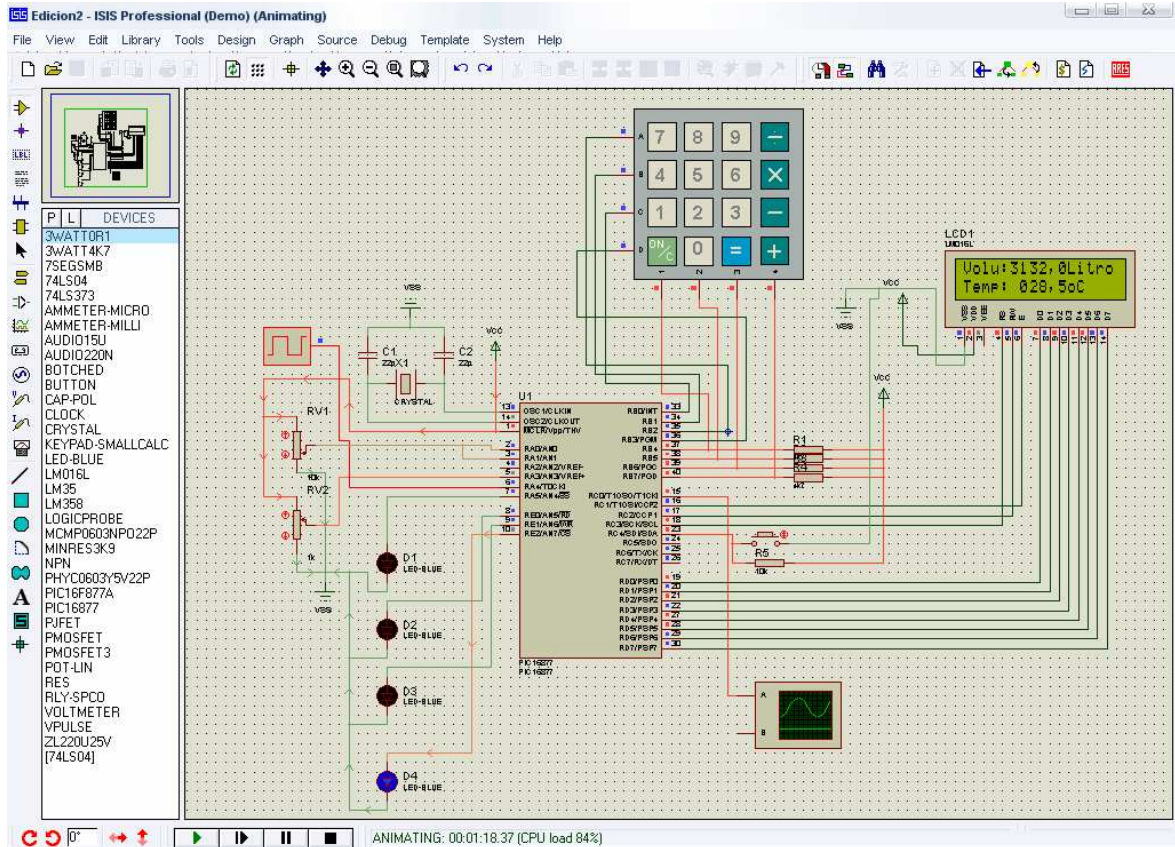


Figura 2.3.- Circuito del prototipo realizado en ISIS para las simulaciones

2.1.1.4 El Grabador

En este caso se utilizó el programa ICPROG y se adquirió una tarjeta que permite grabar fácilmente al PIC que utilizaremos.

Este programa ICPROG es de libre distribución así que no es difícil conseguirlo en Internet gratuitamente.

En las figuras siguientes se muestra el grabador de PIC's utilizado, así como una muestra de la pantalla principal del IC-Prog.

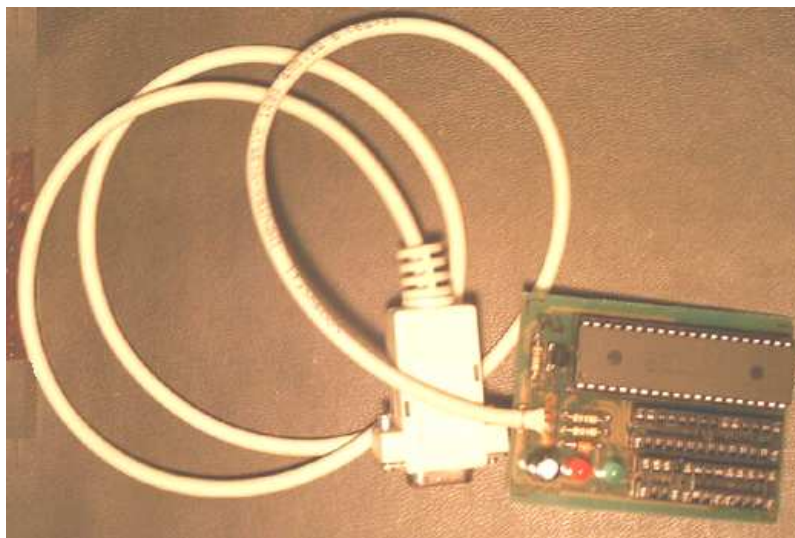


Figura 2.4.- Grabador de PIC's utilizado en este proyecto

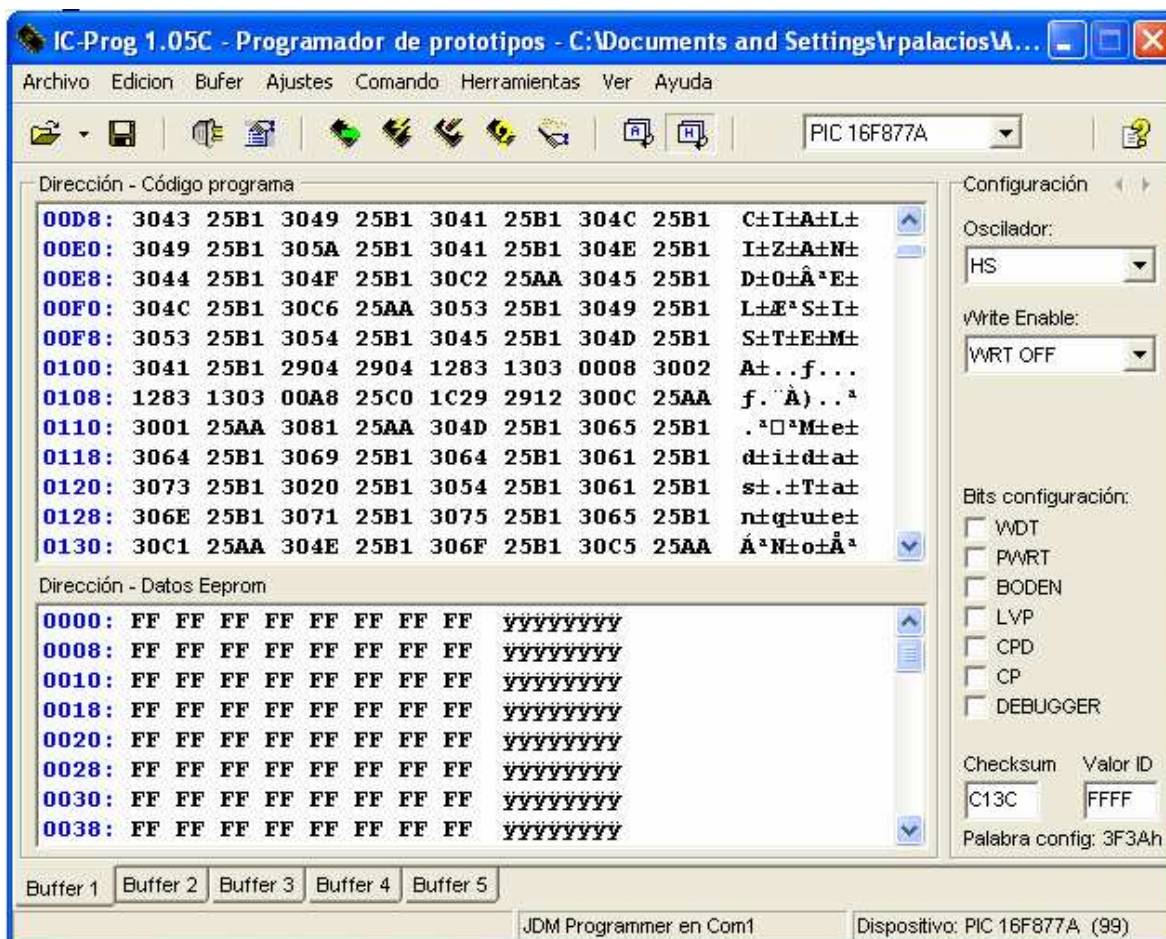


Figura 2.5.- Programa IC-Prog para grabar el programa ADC.HEX en el PIC

2.1.2. CONTROL DE NIVEL DE LÍQUIDO

Para realizar el control de este parámetro se tiene varias opciones, desde colocar una boya para que según el nivel de líquido en el tanque me proporcione una señal eléctrica que me indique que cantidad de líquido esta contenido en el tanque. La opción de este sistema tiene la desventaja de ser difícil de montar en cualquier tanque.

También tenemos la opción del uso de una varilla metálica introducida en el tanque, teniendo una variación de la capacitancia entre esta varilla y el tanque, de acuerdo a la cantidad de líquido que hay en el tanque. Esto se cumple siempre y cuando el tanque sea metálico, o a menos que utilicemos dos placas metálicas, haciendo de esta manera también difícil de instalar y no muy efectivo, ya que necesitaríamos de varillas que sean capaces de variar su longitud para cubrir todo el rango de altura del tanque.

Para evitar estos inconvenientes la opción que hemos escogido, por su fácil montaje y economía, es la utilización de sensores ultrasónicos, que sean fáciles de conseguir.

2.1.2.1 Sensores Ultrasónicos

Se define al ultrasonido como vibraciones del aire de la misma naturaleza que el sonido audible pero de una frecuencia mas elevada; su frecuencia parte desde los 20000 Hz hasta los 5×10^8 Hz, no audibles por el oído humano. Se alcanzan longitudes de onda que se aproximan a la luz visible.

Además se debe indicar que el ultrasonido es dependiente de la temperatura.

En este caso para la medición del nivel de líquido se utilizó los sensores de ultrasonido (136653CK), que está formado por dos unidades piezoeléctricas en donde una de ellas es el emisor y la otra el receptor de ondas de presión ultrasónicas.

Estos pequeños sensores son comunes en el mercado y sobre todo de bajo costo. Las características específicas de estos se las puede apreciar en el **Anexo D**.

2.1.2.1.1 *Funcionamiento*

Con la utilización de estos sensores, la circuitería adecuada (ver diagrama esquemático de conexión de los sensores ultrasónicos (**Figura 2.6 y 2.7**), y partiendo de la **Fórmula 1.1** para el cálculo de la distancia se puede llegar a conocer el nivel del líquido.

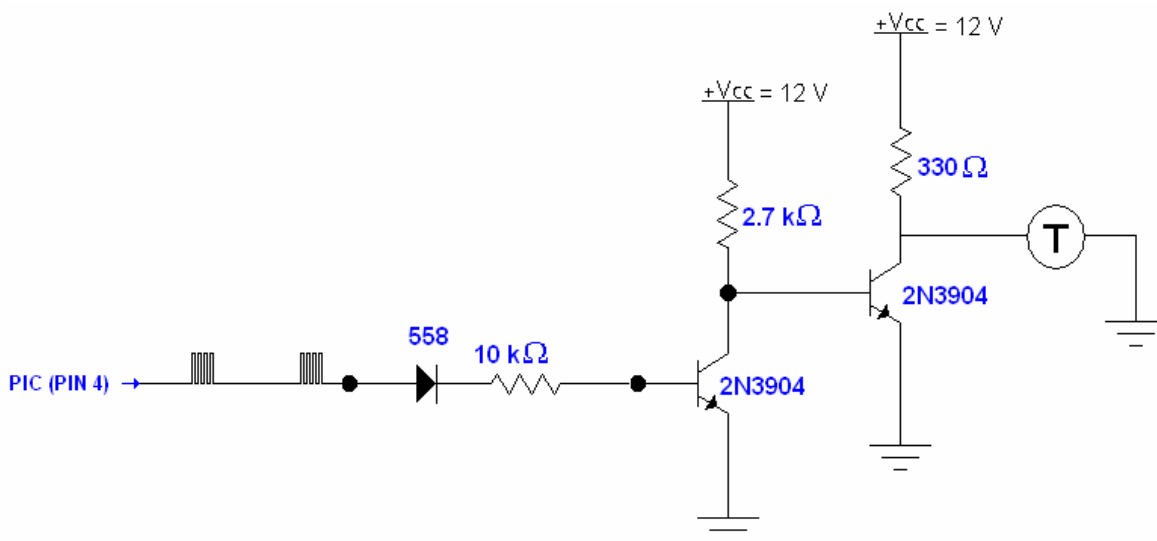


Figura 2.6.- Diagrama Esquemático Transmisor Ultrasónico

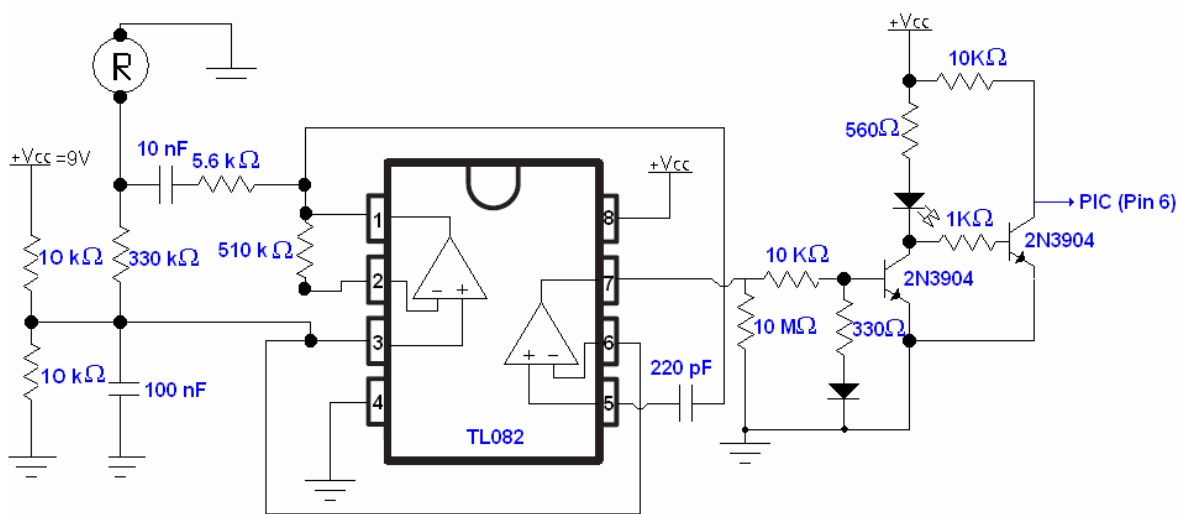


Figura 2.7.- Diagrama Esquemático Receptor Ultrasónico

La unidad emisora debe excitarse con una señal adecuada en amplitud y frecuencia, para que genere la señal ultrasónica adecuada; la señal ultrasónica se refleja sobre la superficie del líquido e inmediatamente la unidad receptora transformara en señales eléctricas todas aquellas ondas de presión ultrasónicas, que fueron generadas y reflejadas (ver características de receptor ultrasónico **Anexo D**).

El tiempo transcurrido entre el comienzo de la emisión y el comienzo de la recepción será proporcional a la distancia recorrida por las ondas ultrasónicas. De esta manera obtenemos el primer parámetro para resolver esta fórmula.

Debido a que la señal ultrasónica recorre un total de dos veces la distancia que existe entre el transmisor y el nivel del líquido, este tiempo se deberá dividir por dos para poder conocer dicha distancia necesaria para resolver la fórmula de partida.

El siguiente parámetro que debemos obtener es la velocidad de la señal ultrasónica, pero debemos conocer que la velocidad del ultrasonido especialmente varía de acuerdo a la temperatura del medio en el que se propaga (Ver **Fórmula 1.3**).

Pero como vemos antes de conocer la velocidad de propagación de la señal ultrasónica es necesario conseguir la temperatura del medio que se encuentra entre el transmisor ultrasónico y el líquido, que para este caso es el aire.

2.1.2.1 Sensor de Temperatura para el Ambiente

En el mercado existen numerosos tipos de sensores de temperatura o termómetros que, según la aplicación específica, pueden ser los más adecuados, a continuación presentamos los sensores utilizados para este proyecto, que son el circuito integrado LM35.

2.1.2.1.2 Circuito Integrado LM35

El LM35 es un circuito integrado de precisión (con una precisión de 1°C calibrado a temperatura ambiente 25°C) que actúa como un sensor de temperatura graduado directamente en grados centígrados ($^{\circ}\text{C}$) y un rango que abarca desde -55° a $+150^{\circ}\text{C}$. Su tensión de salida es lineal y proporcional a la escala Celsius de temperatura, teniendo una variación de 10mV por cada grado centígrado.

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

Este es un dispositivo muy común en el mercado y sobre todo de bajo costo.

En el **Anexo E** se puede observar en detalle las características de este sensor.

Mediante la utilización del integrado LM35, el circuito adecuado y un canal de conversión de analógico / digital que este libre en el PIC (ver diagrama esquemático de conexión del integrado LM35 **Figura 2.8**), es posible obtener el valor de la temperatura del aire que se encuentra entre el transmisor y el líquido contenido en el tanque, representado en forma digital para que pueda ser utilizado por el mismo PIC para los respectivos cálculos.

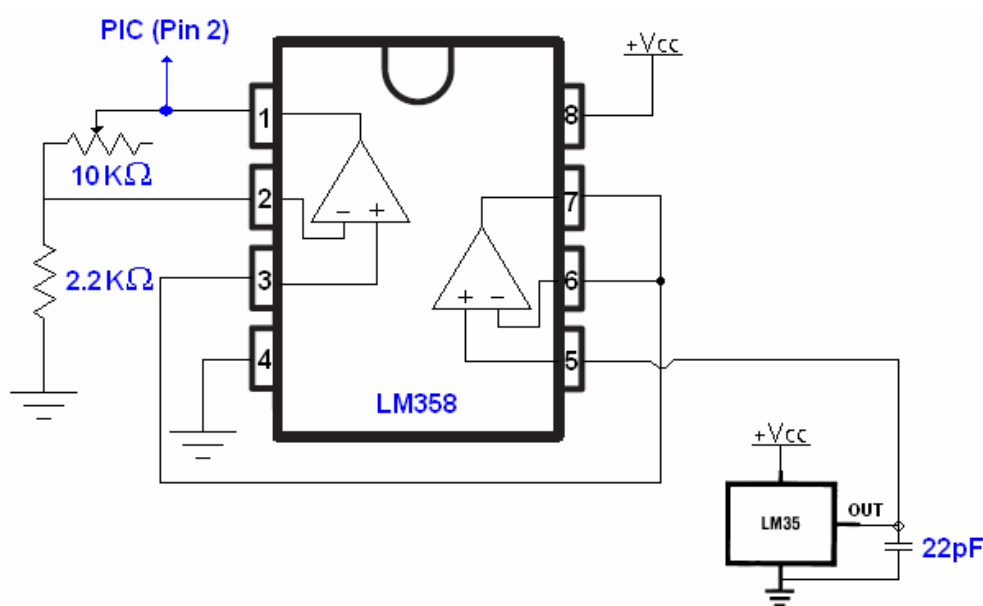


Figura 2.8.- Diagrama Esquemático para conexión del integrado LM35

De aquí es posible conseguir el segundo parámetro, que buscamos para obtener la distancia entre el sensor ultrasónico y el líquido, que es la velocidad de la señal ultrasónica.

Una vez conocido este último parámetro bastará con tener la altura total del tanque, para poder restar de este valor la distancia obtenida, y por lo tanto podremos dar un resultado sobre el nivel del líquido contenido en el tanque.

Luego bastará con conocer los límites máximo y mínimo del nivel de líquido que se quiere tener en el tanque, para de acuerdo a estos datos se active o no el sistema que actúa para llenar el tanque.

Cabe mencionar que según lo que se ha podido observar e indagar la gran mayor parte de gente dentro del país tienen presente como unidades para medir la longitud a los metros con sus múltiplos y submúltiplos, por lo que es precisamente este el sistema de unidades que se utiliza para la medida de la altura del tanque y el nivel del líquido.

Ya que conocemos el nivel del líquido contenido en el tanque, es posible conocer también el volumen del líquido, siempre y cuando sea conocido el radio del tanque, ya que con estos datos podemos resolver la **Fórmula 1.4:**

$$V = \pi(r^2)h$$

Formula 1.4.- Fórmula para el volumen de un cilindro

Donde:

V : Volumen del líquido en el tanque

r : Radio del tanque

h: Altura a la que se encuentra el líquido (nivel)

Cabe mencionar que según lo se que ha podido observar e indagar la gran mayor parte de gente dentro del país tienen presente como unidades para medir el volumen a los litros con sus múltiplos y submúltiplos, por lo que es precisamente

este el sistema de unidades que se utiliza para la medida del volumen de líquido en el tanque, aunque se puede tener el volumen del líquido en galones ya que también es muy utilizado en ciertos sectores.

2.1.3. CONTROL DE LA TEMPERATURA DEL LÍQUIDO

2.1.3.1 Termocupla



Figura 2.9.- Fotografía de Termocupla real

Para medir la temperatura a la que se encuentra un líquido no es recomendable utilizar circuitos integrados como el LM35 ya que no están precisamente diseñados para trabajar en esos medios, por lo que se optó por utilizar una termocupla, que es perfecta para la aplicación que se plantea.

Una termocupla es un dispositivo que traduce la temperatura en una señal eléctrica. Está constituida por dos alambres metálicos diferentes, que unidos desarrollan una diferencia de potencial eléctrica entre sus extremos libres, el mismo que es aproximadamente proporcional a la diferencia de temperaturas entre estas puntas y la unión. Se suelen fabricar con metales puros o aleaciones (caso más común) y la característica más notable es que son empleadas para medir temperaturas en un rango importantemente grande comparadas con otros elementos.

Para este proyecto únicamente se lo ha calibrado para que mida de 0° a 100° como temperaturas extremas mínima y máxima respectivamente, ya que son los puntos de congelamiento y ebullición del agua.

Es necesario recalcar que una termocupla, en rigor, mide diferencias de temperaturas y no temperaturas absolutas.

Nosotros utilizamos una termocupla en forma de tornillo que es muy práctica, económica y fácil de instalar; lastimosamente no contamos con las especificaciones técnicas exactas, únicamente se sabe que es del tipo K por lo que debe funcionar correctamente en el rango anteriormente especificado.

Este proceso del sistema es mucho más fácil de resolver ya que contamos con el elemento perfecto que nos permitirá conocer la temperatura a la que se encuentra el líquido.

La termocupla al estar formada por partes metálicas es perfecta para soportar tanto altas como bajas temperaturas.

Bastará con adecuar con la circuitería (**Figura 2.10**) adecuada a la señal que nos entrega la termocupla, para poder conectarlo a un canal de conversión analógico / digital que se encuentre libre del PIC, y del cuál mediante algunos cálculos podremos obtener en forma digital el valor de la temperatura a la que se encuentra el líquido, capaz de poder ser mostrado en la pantalla LCD.

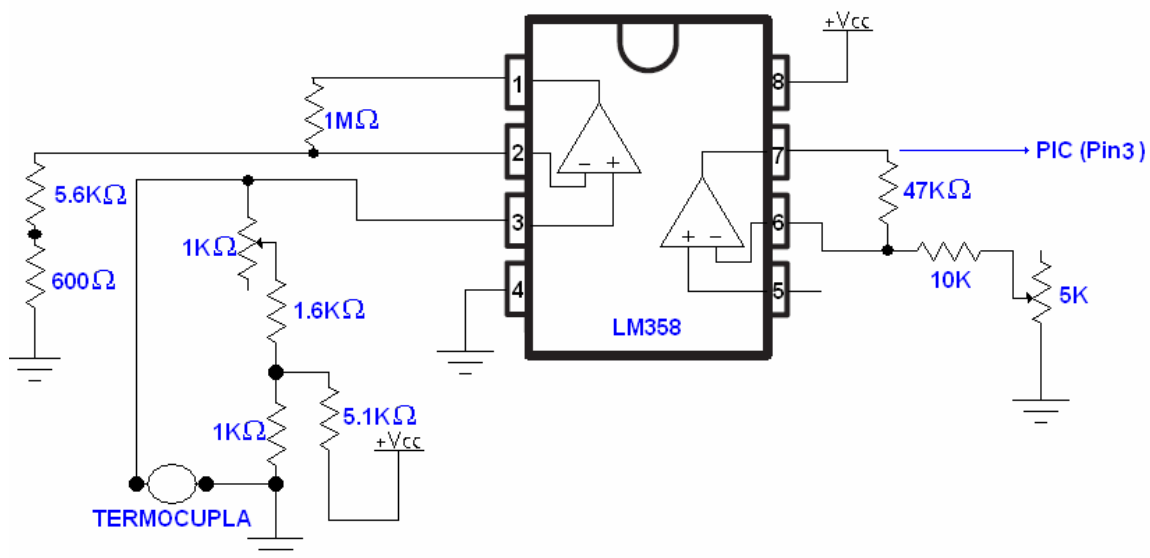


Figura 2.10.- Diagrama esquemático para el sensor de temperatura del líquido (Termocupla)

Cabe mencionar que según lo que se ha podido observar e indagar la gran mayor parte de gente dentro del país tienen presente como unidades para medir la temperatura a los grados centígrados, por lo que es precisamente este el sistema de unidades que se utiliza para la medida de temperatura del líquido.

2.1.4. EXTRACCIÓN Y LLENADO DE LÍQUIDO

2.1.4.1 Extracción segura de líquido

Esta parte de sistema se basa en que no debe existir manipulación alguna por parte del usuario para poder conseguir una cierta cantidad de líquido del tanque; sobre todo cuando se trata de líquidos que sean peligrosos o de uso restringido.

Para esta parte del sistema se plantea simplemente que el usuario pueda ingresar mediante teclado un valor de líquido a descargar, ya sea como nivel o volumen, y que el sistema automáticamente haga todo el trabajo.

Una vez que el sistema tiene la cantidad de líquido a descargar, se resta este valor del nivel actual del líquido, obteniendo como resultado el valor final de líquido en el tanque. Entonces se procede a abrir una válvula que permita la salida del líquido del tanque, hasta que el nivel actual del líquido alcance el valor final de líquido en el tanque

2.1.4.1.1 Electro - Válvula



Figura 2.11.- Fotografía de una Electro – válvula real

Para la extracción segura del líquido utilizamos una electro-válvula que no es más que un aparato mecánico con el cual se puede iniciar, detener o regular la circulación (paso) de líquidos o gases mediante una pieza movable que abre, cierra u obstruye en forma parcial o total uno o más orificios o conductos.

Estas válvulas se utilizan cuando la señal proviene de un temporizador eléctrico, un final de carrera eléctrico, reóstatos o mandos electrónicos. En general, se elige el accionamiento eléctrico para mandos con distancias extremadamente largas y cortos tiempos de conexión.

Lastimosamente se desconoce mayor detalle de la válvula utilizada para este proyecto ya que se cuenta solo con un parámetro (funciona con una tensión de 110V), pero cumple que con las necesidades del proyecto ya que se requiere de un dispositivo que permita u obstruya totalmente el paso de líquido cuando reciba ciertas órdenes del dispositivo de mando; además de ser económica y muy fácil de encontrar.

2.1.4.2 Llenado de líquido

De la misma manera esta parte de sistema se basa en que no debe existir manipulación alguna por parte del usuario para poder llenar el tanque con el líquido.

Para esta parte del sistema se plantea simplemente que el tanque se llene automáticamente cuando el nivel líquido no cumpla con el mínimo requerido y establecido en el ingreso de datos al iniciar el sistema.

2.1.4.2.1 Bombas



Figura 2.12.- Fotografía de una bomba real

Para llenar de líquido al tanque se utilizó una bomba que no es más que un convertidor de energía, o sea, transforma la energía mecánica en energía cinética, generando presión y velocidad en el fluido.

Al tratarse de un prototipo se requiere de pequeñas cantidades de líquido, con un flujo constante, y una altura máxima de 30 cm.; tomando en cuenta estos aspectos estamos utilizando una bomba que aunque se desconoce mayor detalle, ésta trabaja a 110V, 60 Hz, cumpliendo con los requerimientos de este sistema.

2.1.5. ELEMENTOS DE INTERFAZ

2.1.5.1 Teclado Matricial 4x4



Figura 2.13.- Fotografía de un teclado real

2.1.5.1.1 Descripción

El teclado matricial es un dispositivo de entrada de datos que en este caso consta de 16 teclas o pulsadores, dispuestos e interconectados en filas y columnas, además se puede observar los ocho pines que se corresponden con las cuatro filas y las cuatro columnas.

2.1.5.1.2 Funcionamiento

En el esquema de conexión de un teclado matricial (**Figura 2.14**) se puede observar que al digitar una tecla cualquiera se interconecta filas y columnas dependiendo de la tecla presionada, así el PIC detecta el dato enviado y realizará las instrucciones previamente programadas.

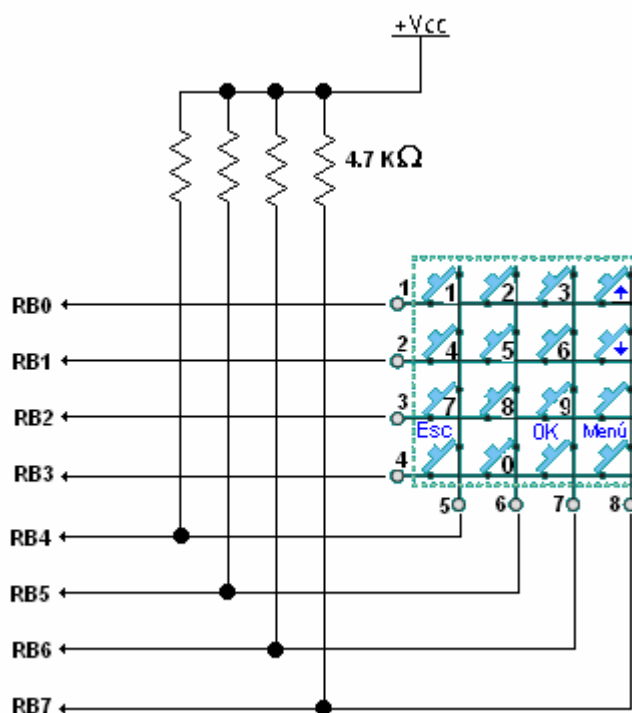


Figura 2.14.- Esquema de conexión de un teclado

2.1.5.2 Pantalla LCD 2x16

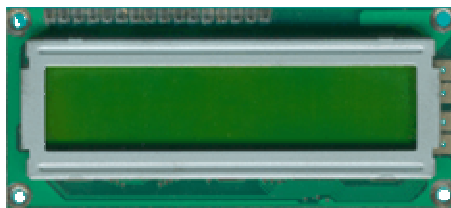


Figura 2.15.- Pantalla LCD 2 filas x 16 caracteres

2.1.5.2.1 Descripción

La pantalla de cristal liquido o LCD (Liquid Crystal Display) es un dispositivo microcontrolador de visualización grafica para la presentación de caracteres, símbolos o incluso dibujos (en algunos modelos).

En este caso dispone de 2 filas de 16 caracteres cada una, cada carácter dispone de una matriz de 5x7 puntos (pixeles), aunque los hay de diferente número de filas y caracteres. Este dispositivo esta gobernado internamente por un microcontrolador Hitachi 44780 y regula todos los parámetros de presentación.

Características principales:

- Pantalla de caracteres ASCII, además de los caracteres Kanji y Griegos.
- Desplazamiento de los caracteres hacia la izquierda o la derecha.
- Proporciona la dirección de la posición absoluta o relativa del caracter.
- Memoria de 40 caracteres por línea de pantalla.
- Movimiento del cursor y cambio de su aspecto.
- Conexión a un procesador usando un interfaz de 4 u 8 bits

2.1.5.2.2 Configuración

Para comunicarse con la pantalla LCD podemos hacerlo por medio de sus pines de entrada de dos maneras posibles, con bus de 4 bits o con bus de 8 bits.

En el **Anexo F** se pueden apreciar la distribución de pines del LCD.

2.2 DISEÑO DEL PROTOTIPO

El sistema cuenta con un MENÚ PRINCIPAL que ayuda al usuario para controlar el dispositivo. Este MENÚ se describe a continuación:

MENÚ PRINCIPAL

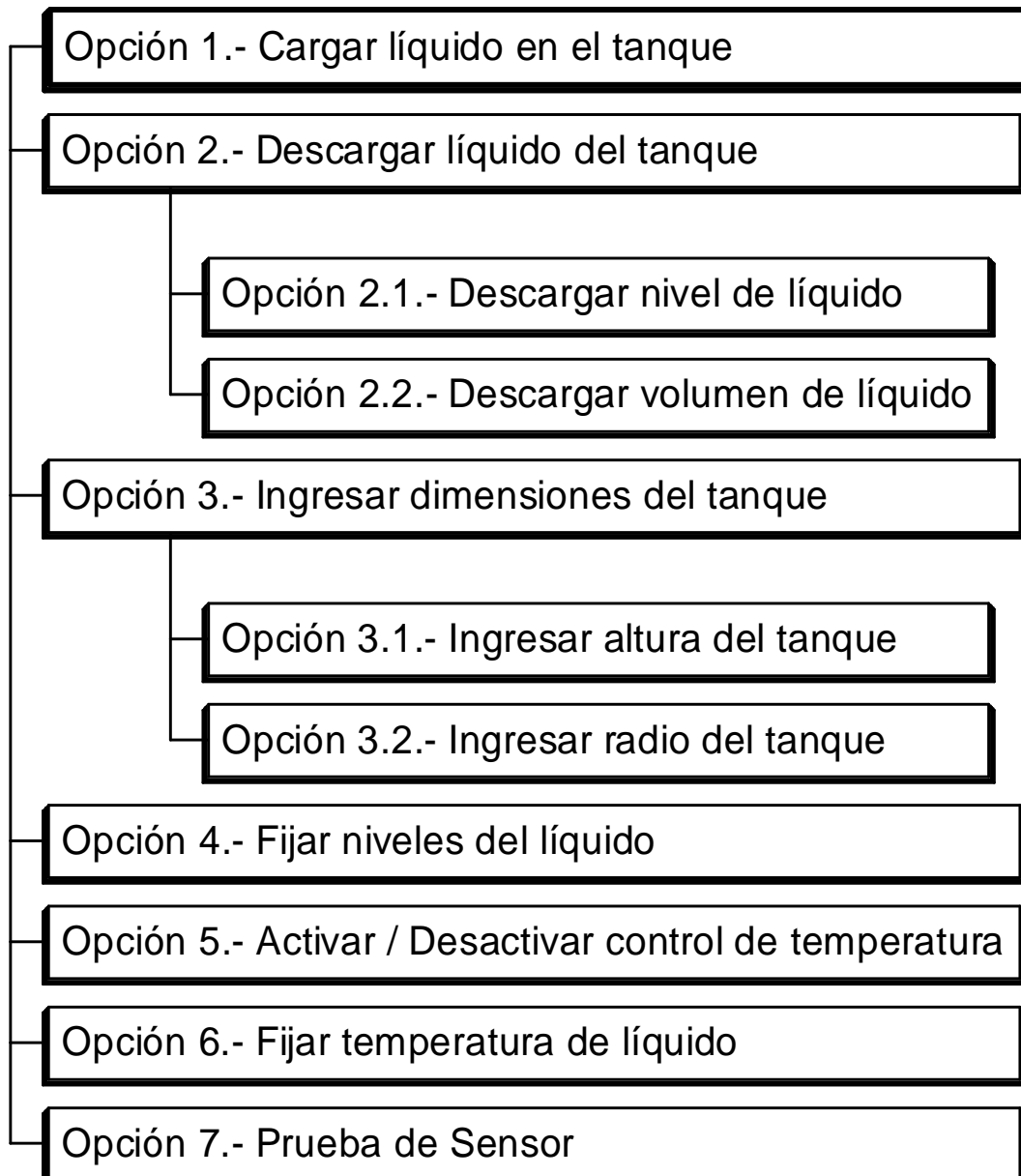


Diagrama 2.1.- Menú Principal

2.2.1 PROGRAMA PRINCIPAL

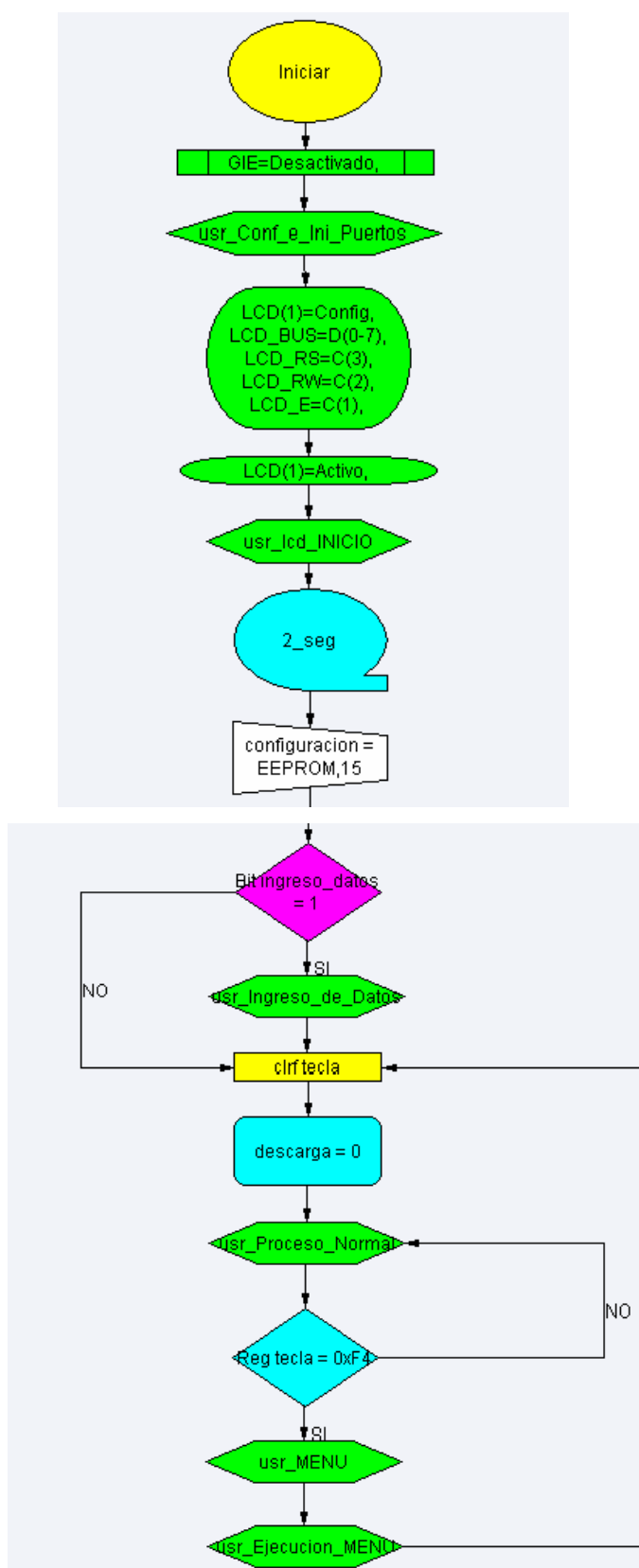


Figura 2.16.- Diagrama de flujo del programa principal

2.2.2 EXPLICACIÓN DEL DIAGRAMA

- Inicialmente empezamos por desactivar todas las interrupciones por medio del bit GIE (General Interruption Enable).
- Luego configuramos como entradas o salidas todos los puertos del PIC conforme al hardware que estamos utilizando, es decir de acuerdo a todos los periféricos que se esta conectando al PIC, de tal manera que se cumpla con el diagrama mostrado en la **Figura 2.2** donde consta la distribución de hardware del proyecto. Inmediatamente se debe inicializar dichos puertos.
- A continuación configuramos la pantalla LCD que se esta utilizando para mostrar toda información. Esta configuración debe realizarse de acuerdo al siguiente grafico.

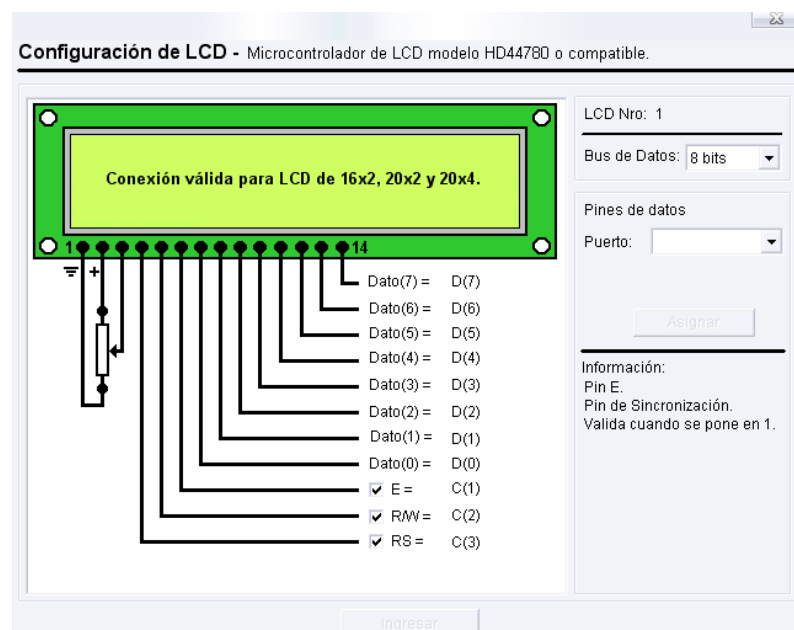


Figura 2.17.- Configuración de LCD

Donde tenemos todo el PUERTO D del PIC utilizado para enviar los datos a la pantalla LCD, mientras que los pines RC1 (Enable), RC2 (Read/Write) y RC3 (Register Select) sirven para enviar las respectivas señales de control a la pantalla.

- Luego se debe activar la pantalla LCD e inmediatamente pasar a mostrar el primer mensaje que será el mensaje de bienvenida. Se puede ingresar cualquier leyenda, pero la que se eligió para este proyecto es INICIALIZANDO EL SISTEMA.
- Este mensaje será mostrado en la pantalla por 2 segundos, por lo que se debe ingresar una temporización por bucles de indique este tiempo.
- Pasamos a leer la configuración del sistema que se encuentra guardada en la dirección 15 de la memoria EEPROM del PIC, hacia un registro que ha sido designado como **configuracion**.
- Dentro del registro **configuracion** tenemos el bit 7 que es llamado **ingreso_datos**. Pues ahora se debe comprobar su valor ya que si es uno (1) se debe pedir el ingreso de los datos del tanque ya que nos indica que es la primera vez que estamos encendiendo el sistema, pero en caso de que sea cero (0), nos esta indicando que los datos del tanque ya fueron ingresados y nos es necesario que se ingrese nuevamente esta información.

En caso sea necesario el ingreso de esta información, se muestra un mensaje con la leyenda MEDIDAS TANQUE NO INGRESADAS, por medio del cual se da ha entender que es necesario ingresar los datos del tanque contenedor de liquido sobre el cual estamos intentando hacer funcionar el sistema.

Los datos requerimos para el funcionamiento del sistema son:

- ALTURA DEL TANQUE.- Necesario para el cálculo del nivel de líquido contenido en el tanque. Esta información tiene el rango de 0.3 m a 1.8m que es lo que nos permite el sistema tal y cual fue desarrollado, pero con algunas modificaciones, sobre todo del programa, se podría tener mayor alcance; obviamente que esto influirá en la resolución que tenga el sistema de medición de nivel.
- RADIO DEL TANQUE.- Necesario para el cálculo del volumen de líquido contenido en el tanque. Esta información tiene el rango de 0.1 m a 1.0m que es lo que nos permite el sistema tal y cual fue desarrollado, pero con algunas modificaciones, sobre todo del programa, se podría tener mayores dimensiones; obviamente que esto influirá en la resolución que tenga el sistema de medición del volumen.
- NIVEL MÁXIMO.- Indica el nivel máximo que podrá alcanzar el líquido en el tanque.
- NIVEL MÍNIMO.- Indica el nivel mínimo que podrá alcanzar el líquido en el tanque.
- ACTIVAR O DESACTIVAR CONTROL DE TEMPERATURA.- Indica si se desea que el control de temperatura que posee el sistema estará activado o desactivado.
- TEMPERATURA MÁXIMA Y TEMPERATURA MÍNIMA.- Solo en caso de que el control de temperatura este activado se deberá establecer estos parámetros, que claramente se trata de la temperatura máxima y la temperatura mínima que tendrá el líquido en el tanque. Estas temperaturas pueden estar en el rango de 0°C a 100°C, con una diferencia de 5°C entre ellas.

- Se indica mediante el bit ***mensaje*** del mismo registro ***configuracion*** que el sistema debe mostrar por defecto el mensaje 0 en la pantalla cuando el sistema este en el proceso normal de funcionamiento.
- Como últimos pasos de este ingreso de datos del tanque se indica mediante el bit ***ingreso_datos*** que la información del tanque ya fue ingresada y se graba la configuración actual del sistema en la memoria EEPROM del PIC.
- Inmediatamente paso a mostrar por un segundo más el mensaje de inicio del sistema.
- Se procede a borrar el registro que contiene el valor de la tecla presionada.
- Colocamos a cero (0) el bit descarga que nos indica si el sistema se encuentra en una ejecución normal del sistema o si se encuentra en un proceso de descarga de liquido, en el primer caso se ejecutara el control de nivel máximo y nivel mínimo, pero en el segundo caso se desactiva esta opción y simplemente ejecutara la instrucción de descarga de liquido desde el tanque.
- El paso siguiente es la ejecución normal del sistema, donde tenemos los siguientes pasos:
 - Empezamos por activar la interrupción de teclado, ya que en caso de que una tecla sea presionada, este valor se guarde provisionalmente en el registro tecla, para luego poder realizar la función determinada para cada tecla dentro de la ejecución normal del programa.
 - El siguiente paso es la lectura de la temperatura del medio (aire) que tenemos entre el sensor ultrasónico y el líquido que contiene el tanque. Este se realiza por medio de la circuitería indicada para este

fin (**Figura 2.8**) y sobre todo por medio del conversor A/D (Analógico / Digital) que posee el PIC en el canal 0 (pin RA1), rigiéndose con los cálculos de temperatura para el sensor LM35 aplicando las fórmulas indicadas anteriormente.

- Seguidamente con este dato de la temperatura del aire pasamos al cálculo de la velocidad que tendrá la onda ultrasónica al propagarse por este medio (aire), enfocando la **Fórmula 1.3**
- Por medio de los circuitos indicados (**Figura 2.10**), y el conversor A/D pero esta vez por el canal 1 (pin RA2) obtenemos la temperatura del líquido contenido en el tanque.
- También los cálculos respectivos para la obtención de esta temperatura del líquido a partir de la conversión del PIC.
- Se realiza la revisión respectiva para ver si la temperatura del líquido se encuentra dentro de los límites máximo y mínimo de temperatura, para activar o desactivar los sistemas de calentamiento o enfriamiento y por medio de la circuitería indicada (**Figura 2.23**) encender o apagar los instrumentos con que cuente el sistema para dicho fin.
- Por medio de la subrutina SEÑAL ULTRASÓNICA, se procede a generar un tren de pulsos de 40KHz en el PIN RA2, el mismo que sirve para alimentar al sistema que lleva la señal indicada para que el transmisor ultrasónico funcione adecuadamente; para luego poder detectar por medio de receptor ultrasónico y la circuitería indicada la misma señal emitida pero desfasada en el tiempo, y por medio del PIN RA4 poder detener al TIMER 1 que fue encendido en el momento que se genero la señal ultrasónica. De tal manera que los respectivos registros del TIMER 1, contendrán el tiempo que demoro

la señal ultrasónica en recorrer la distancia que separa al transmisor – receptor ultrasónico del líquido que contiene el tanque.

- Se procederá a realizar el cálculo del nivel del líquido que esta contenido en el tanque, pues ya se cuenta con los datos necesarios para resolver la ecuación que los liga, que es:

$$d = v \cdot t$$

Formula 2.1.- Fórmula 1.1 de la distancia aplicada para conocer el nivel del líquido

DONDE

d: Distancia entre el transmisor – receptor ultrasónico y el líquido

v: Velocidad de la señal ultrasónica.

t: Tiempo de la señal ultrasónica en recorrer la distancia **d**

Para obtener el nivel del líquido bastara con restar del valor de la altura del tanque, que habríamos ingresado anteriormente, el valor de la distancia **d**.

- Enseguida realizamos el control de nivel liquido, comprobando que se encuentre entre los límites máximo o mínimo, siendo el caso si llega al nivel mínimo se enciende el sistema de llenado y si se llega al nivel máximo se desactiva dicho sistema, sea cual sea el elemento elementos que se destine a dicho propósito, en este caso podría ser una válvula que simplemente permita o bloquee completamente el paso del liquido.
- Seguidamente se procederá a realizar el cálculo de volumen del líquido contenido en el tanque. Ya que contamos con los datos necesarios será relativamente fácil realizar la resolución de la ecuación:

$$V = \pi \cdot (r^2) \cdot h$$

Formula 2.2.- Fórmula 1.4 aplicada para el volumen de un cilindro

DONDE

V: El volumen del líquido contenido en el tanque

r: Radio del tanque contenedor del líquido

h: Nivel del líquido contenido en el tanque

Cabe recalcar que este valor de volumen se encontrara en m³ por lo que también se realiza la división para mil (1000) para tener el valor del volumen en litros, que es lo que se quiere obtener.

- A continuación mostramos el mensaje indicado por el bit **mensaje** del registro **configuracion**. Este mensaje podrá variar entre el mensaje llamado **Mensaje_0** o le mensaje nombrado como **Mensaje_1**, que contienen la siguiente información.

MENSAJE_0

Línea 1: Volumen del líquido en litros

Línea 2: Temperatura del líquido en °C

MENSAJE_1

Línea 1: Volumen del líquido en galones

Línea 2: Nivel del líquido en milímetros

- De esta manera concluye la ejecución del programa en situación normal.
- El paso siguiente es la comprobación de que si se presionó o no la tecla **MENÚ**. En caso de haberse presionado dicha tecla se ejecutará la subrutina **MENÚ**, pero en caso de no presionarse esta tecla se seguirá la ejecución normal del programa de forma indefinida.

- En caso de haberse presionado la tecla **MENÚ**, tanto el sistema de llenado como el sistema de dosificación es detenido y se ejecuta las opciones del **MENÚ PRINCIPAL**, el mismo que es graficado en la **Figura 2.16**, donde se presenta todas las opciones con que cuenta el sistema y que podrán ser seleccionadas por la persona que este utilizando este sistema de control.
- En caso de que no se seleccione ninguna opción y pase veinte (20) segundos, el sistema automáticamente retorna a la ejecución normal del sistema.
- Al seleccionar cualquier opción del **MENÚ PRINCIPAL**, inmediatamente pasa a la ejecución de dicha opción, y este es el último paso que tenemos en el programa. Luego de ejecutar completamente la opción seleccionada se retorna al **PROCESO NORMAL**, cerrando así el ciclo de ejecución del programa.

2.2.3 ERRORES PRODUCIDOS EN LOS CÁLCULOS REALIZADOS

Los cálculos realizados tanto para conocer las magnitudes medidas como nivel, temperatura, etcétera requieren tienen pequeños errores producidos por las aproximaciones para introducir ciertos valores al PIC en los siguientes gráficos se puede apreciar dichos errores:

ERROR TEMPERATURA LM35

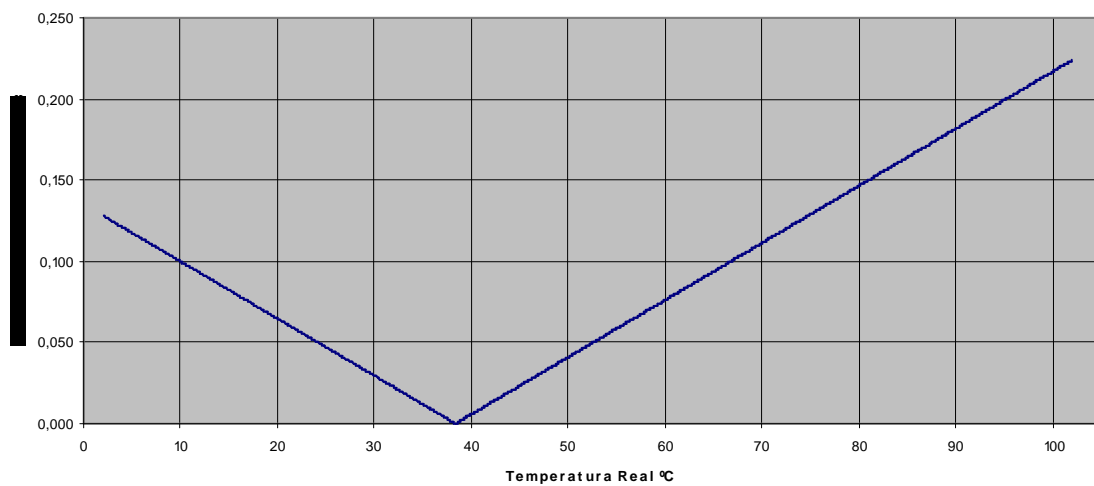


Figura 2.18.- Error Temperatura LM35

Error Termocupla

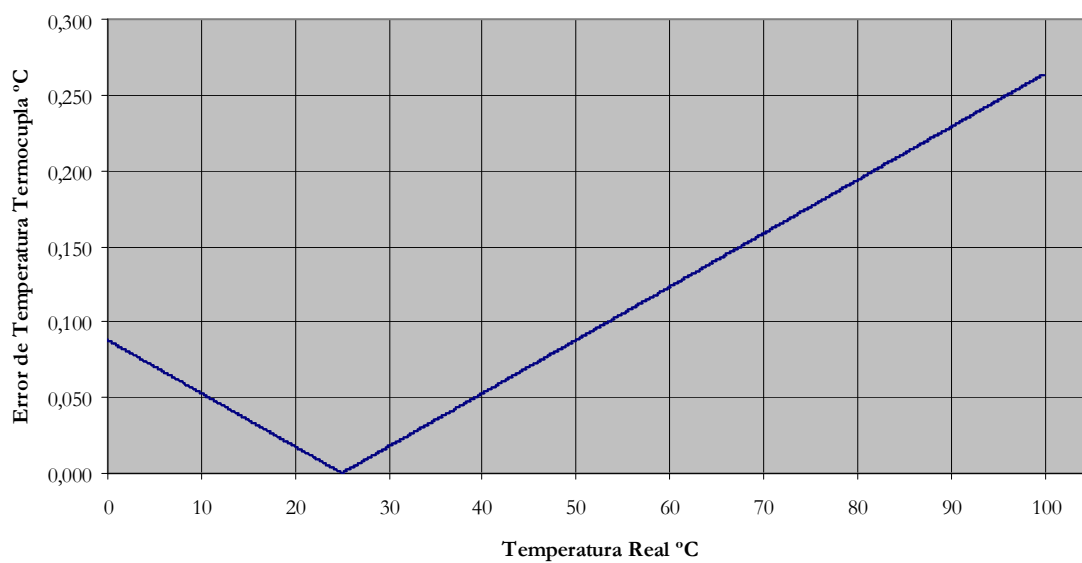


Figura 2.19.- Error Temperatura Termocupla

Error Velocidad - Temperatura

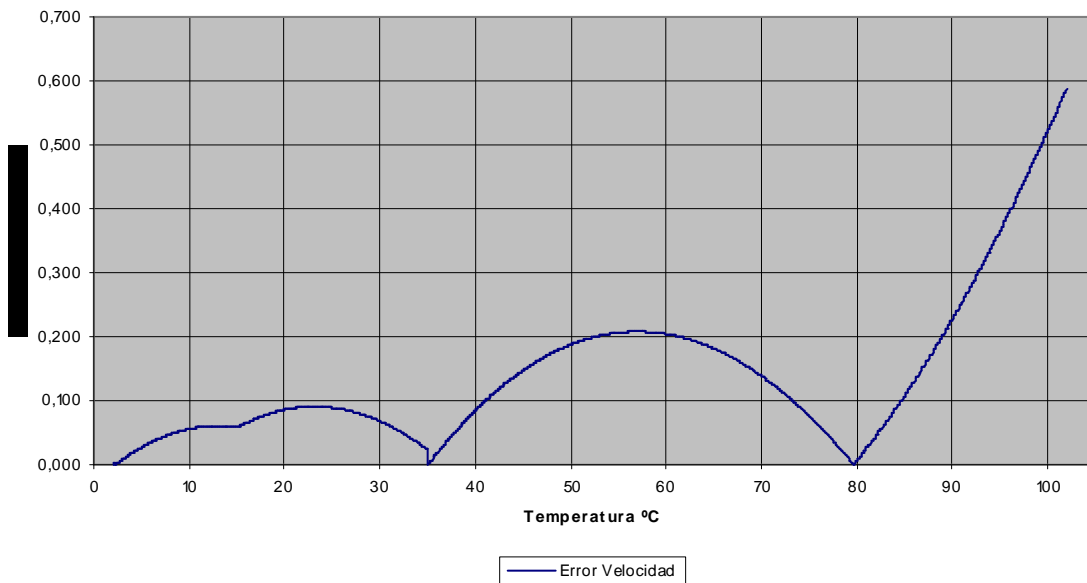


Figura 2.20.- Error Velocidad

VELOCIDAD - TEMPERATURA

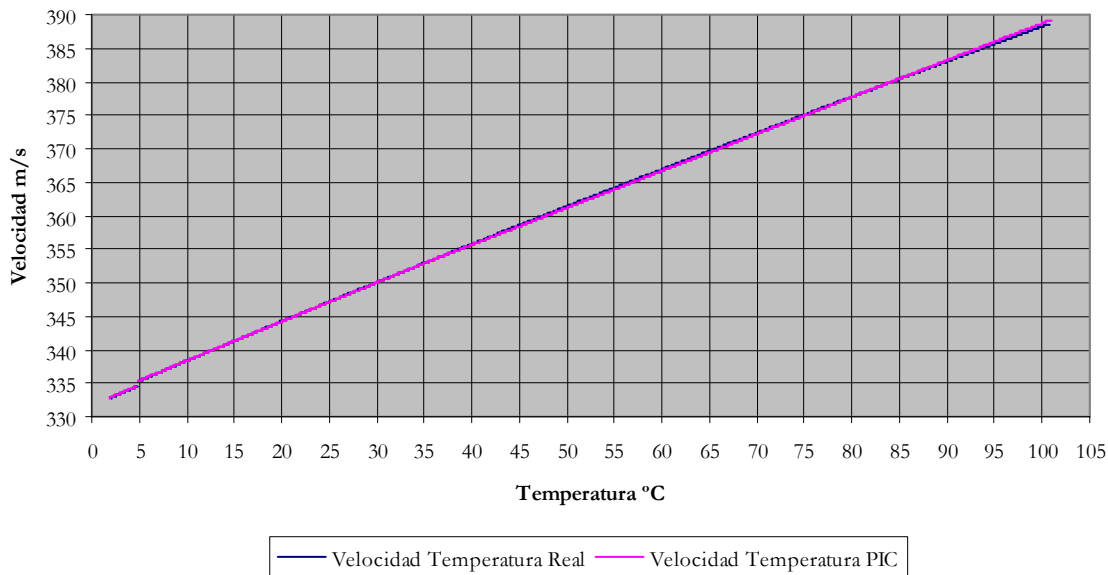


Figura 2.21.- Desviación de la Velocidad según la Temperatura

2.3 CONSTRUCCIÓN DEL PROTOTIPO

Para la construcción del prototipo se han implementado diferentes etapas cuyas circuiterías están dispuestas de acuerdo a la función que se va a realizar y al parámetro a controlar.

2.3.1 CIRCUITO PRINCIPAL

En el circuito principal el componente básico es el Microcontrolador PIC 16F877 que tiene sus respectivas conexiones de polarización así como un oscilador externo de 20 MHz configurado para trabajar en modo HS (High Speed Cristal/Resonador), este modo de trabajo soporta la frecuencia generada por el oscilador y requiere que los respectivos condensadores estén en el rango de 15 a 33 pF.

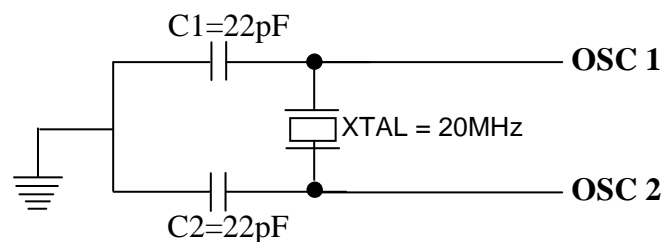


Figura 2.22.- Configuración Oscilador Modo HS

Además utilizamos una resistencia en el MCLR (Pin #1 del PIC) conectada directamente a cinco (5) voltios, ya que en este caso no queremos habilitar la opción de RESETEAR al PIC.

2.3.2 CONTROL DEL NIVEL DE LÍQUIDO

Para realizar el control del nivel del líquido utilizando los sensores de ultrasonido se genera desde el pin 4 del PIC una frecuencia de 40KHz hasta que la señal sea recibida por el receptor teniendo como obstáculo límite la base del tanque, es decir, existiría una distancia límite recorrida por las ondas de dos veces la altura total del tanque hasta que la señal sea receptada.

La señal enviada tendrá un tiempo máximo de duración de 13ms, dado el caso de que la señal no sea recibida al transcurrir dicho periodo se reconfigurarán los puertos y contadores para la generación de los 40kHz y decrementará en uno a un registro temporal cargado con 10, este proceso se repetirá hasta que el registro temporal llegue a 0; está puede ser una premisa de que los sensores no están trabajando correctamente y que los debe dar una revisión.

La señal generada por el PIC luego de pasar por un diodo rectificador nos asegura pulsos en nivel alto de 5V y en bajo nivel de 0V, manteniendo la frecuencia original, así al suministrar esta señal a la base de un transistor NPN configurado para operar en estado de corte y saturación originará una salida que también es pulsante y sincronizada.

Cuando en la base del transistor llegue un pulso se suministra una intensidad de corriente a través de la resistencia de base que activa al transistor, pasa a conducir y la salida quedará conectada tierra, es decir, tendremos 0V.

De igual manera cuando la señal originada se encuentra en 0V la base dicho transistor permanece conectado a tierra por lo que no existirá intensidad de corriente en la base y el transistor no se activa, por ende en la salida del mismo obtendremos el voltaje que atraviesa la resistencia del colector, es decir, 12V para nuestro caso.

Las señales obtenidas van conectadas a otro transistor NPN también configurado para trabajar fuera de la zona de trabajo, es decir, en corte y saturación de ésta manera obtendremos en su salida 12 V cuando existe un impulso desde el PIC y 0V cuando la señal del PIC también es 0, esta es la señal adecuada tanto en amplitud como en frecuencia a la que se excita el transmisor ultrasónico.

El diagrama circuital del transmisor ultrasónico se las puede apreciar en la **Figura 2.6.**

La señal que llega a la unidad receptora es amplificada a través de un amplificador operacional y acondicionada a través de dos transistores conectados también en corte y saturación para que llegue a la patilla número 6 del PIC el mismo que mediante la programación necesaria detectará el tiempo transcurrido entre el inicio de la emisión de las ondas ultrasónicas y el inicio de la recepción, este tiempo será proporcional a la distancia recorrida por las ondas.

El diagrama circuital del receptor ultrasónico se las puede apreciar en la **Figura 2.7**.

2.3.3 CONTROL DE LA TEMPERATURA DEL AIRE

Para medir la temperatura del aire utilizamos el sensor LM35 cuya salida de voltaje es lineal y proporcional a la escala de temperatura en grados centígrados.

Este sensor tiene un rango de trabajo de -55°C a $+150^{\circ}\text{C}$ para nuestro proyecto lo hemos calibrado para trabajar desde 2°C hasta 102°C ; mediante el uso de amplificadores operacionales adecuamos a la señal para obtener una salida de 0V a la temperatura mínima y 3.5V a la temperatura máxima.

El voltaje generado a la salida del amplificador operacional va conectado al pin 2 (RA0) del PIC que es un canal de conversión analógico / digital que recibe la señal y la procesa para los cálculos mencionados en el diseño del proyecto.

Es necesario recalcar que las calibraciones pertinentes se las pueden realizar utilizando el potenciómetro de precisión del circuito y utilizando un termómetro de mercurio de alta escala para comprobar los datos obtenidos.

El diagrama circuital que permite medir la temperatura del aire se lo puede apreciar en la **Figura 2.8**.

2.3.4 CONTROL DE LA TEMPERATURA DEL LÍQUIDO

Para medir la temperatura en un líquido el dispositivo apropiado por diversas razones ya mencionadas fue una termocupla tipo tornillo.

Mediante la circuitería adecuada logramos acondicionar la señal mediante el uso de amplificadores operacionales para que sea receptada y procesada por el PIC en el pin 3 (RA1) que es un canal de conversión analógico / digital.

Para utilizar este dispositivo fueron imprescindibles algunas pruebas ya que no se tenía mayor especificación de éste, el paso inicial fue la verificación de la variación del voltaje al alterar la temperatura, así al medir el voltaje de la termocupla a la salida del amplificador y verificar la temperatura del líquido utilizando un termómetro de de alta escala se pudo observar existía una variación de $35.6\text{mV}/^{\circ}\text{C}$, pero al realizar pruebas con diferentes temperaturas se observó un margen de error no muy alto pero no satisfacía nuestras necesidades.

Al realizar las pruebas de calibración se asume una variación de $35\text{mV}/^{\circ}\text{C}$ valor con el cual se producía el menor margen de error y tendríamos la siguiente relación:

0V	→	0°C
3500mV	→	100°C

Mediante éstos cálculos programados en el PIC podremos obtener en forma digital el valor de la temperatura a la que se encuentra el líquido.

Las calibraciones pertinentes se las pueden realizar con el potenciómetro de precisión de $5\text{K}\Omega$ y utilizando un termómetro de mercurio de alta escala para comprobar los datos obtenidos.

El diagrama del circuito se le puede encontrar en la **Figura 2.10**.

2.3.5 CONTROL DE ACTUADORES

Para el control de los actuadores como válvulas, bombas, etcétera, se ha implementado un sistema de control de potencia que permita aislar la tensión de la red (110V) del el sistema de control y de ésta manera queden físicamente separados del circuito.

Estos circuitos están compuestos por optoacopladores cuya salida controla al Triac y permite que se activen o desactiven los diferentes dispositivos de salida dependiendo de la señal que envíe el PIC.

El diagrama del circuito se le puede encontrar en la **Figura 2.23**

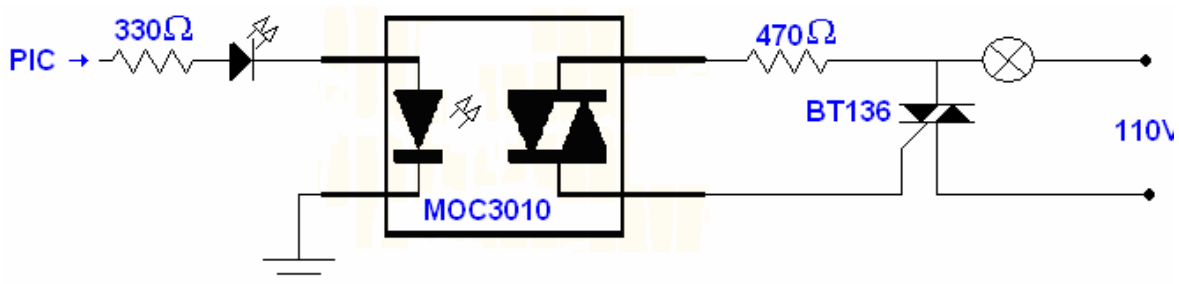


Figura 2.23.- Diagrama de conexión para el circuito de potencia

2.3.6 FUENTE DE ALIMENTACIÓN

Para el funcionamiento del prototipo se requieren tres voltajes 12V_{DC}, 9V_{DC} y 5V_{DC} y una corriente máxima de 85mA al utilizar todas las herramientas que el prototipo posee.

Para satisfacer con estas necesidades se ha decidido diseñar una fuente de alimentación que proporcione los voltajes y corriente necesarios, por lo que se ha utilizado un transformador de 110V a 24V con una corriente de 600mA, además de los reguladores de voltaje LM7812, LM7809 y LM7805 para obtener estabilidad en todo el sistema sobre todo en el PIC.

En la **Figura 2.24** encontraremos el diseño de la fuente de alimentación con la circuitería necesaria y en el **Anexo G** encontraremos las especificaciones de cada uno de los reguladores de voltaje.

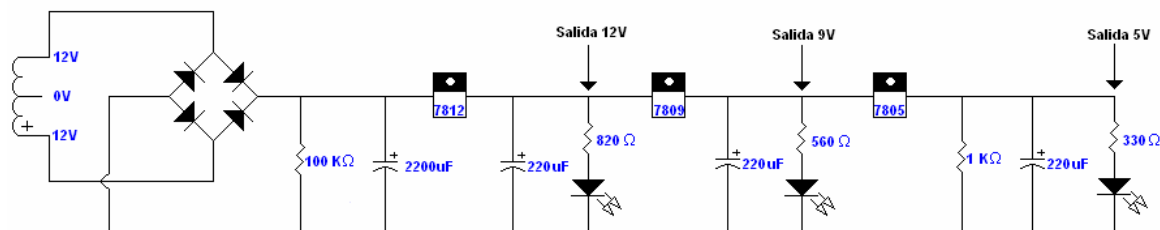


Figura 2.24.- Diagrama de conexión para la fuente de alimentación

2.3.7 ELABORACIÓN DE PLACAS

La elaboración de los circuitos impresos se basa en la consecución del circuito de pistas metálicas montadas sobre una placa de baquelita que sirve para la interconexión de los diferentes elementos que componen el proyecto que se este construyendo.

En este caso se utilizó la herramienta conocida como ARES para la elaboración de los circuitos electrónicos (pistas eléctricas) que luego serán quemados sobre una placa de baquelita.

Este programa viene junto con la herramienta ISIS descrita anteriormente y que a partir del circuito esquemático que construimos en ISIS nos da la posibilidad de la generar las pistas para el circuito impreso que estemos montando. Este programa es de gran utilidad ya que facilita en muchos aspectos este paso en la del proyecto.

Como dijimos anteriormente este software tiene derechos reservados, y aunque su costo es un poco elevado se puede conseguir fácilmente en Internet versiones anteriores que ya han sido liberadas gratuitamente

Los diagramas necesarios generados por el ARES son impresos sobre una la parte brillante de una lámina termotransferible, utilizando una impresora láser, éste paso es muy importante ya que este tipo de impresión se lo realiza a altísimas temperaturas y permite que el tóner se adhiera totalmente a la lámina.

Se prepara la baquelita limpiándola y lavándola utilizando un lustre limpio y jabón desengrasante luego se la seca utilizando servilletas desechable cuidando de no tocar con los dedos para que la lámina se adhiera correctamente y no se produzcan fallas.

Se corta el área de las pistas del circuito generado y se coloca sobre la baquelita (el lado impreso sobre la parte de cobre de la baquelita) y se plancha a una temperatura máxima utilizando un paño delgado aproximadamente por diez minutos, en vista de que las pistas se adhieren a la placa mediante la transferencia de calor es importante aplicar un poco de presión sobre la placa para obtener buenos resultados, después se coloca la placa en un recipiente con agua fría hasta que la lamina esté totalmente mojada y se pueda remover el papel con facilidad.

Luego de limpiar cuidadosamente y de revisar que no haya ningún residuo de la lámina en la placa, se realizan las perforaciones necesarias con un taladro de pedestal utilizando brocas cuyo grosor dependerá del diámetro a perforar.

Una vez que la imagen está adherida al cobre se introduce la placa en un recipiente plástico con cloruro férrico y se agita el recipiente de lado a lado para que el ácido pueda disolver mucho más rápido el cobre de la placa ya que el tiempo influye en gran medida a la calidad final de la placa.

Luego se procede a soldar los elementos utilizando un cautín de baja potencia para evitar que alguna pista se levante y también cuidándose de dejar sueldas frías.

2.3.7.1 Placas Realizadas

Todas las circuiterías detalladas se realizaron en diferentes etapas teniendo como resultado las placas que se muestran a continuación.

2.3.7.1.1 Circuito Principal

Esta placa es la mas compleja ya que tiene las pistas impresas a los dos lados de la baquelita debido existen muchos elementos y por ende muchas líneas de conexión por lo que su construcción requiere de mayor precisión y tiempo, esta fue diseñada pero no elaborada por nosotros a continuación se muestra la Fotografía real del este circuito.

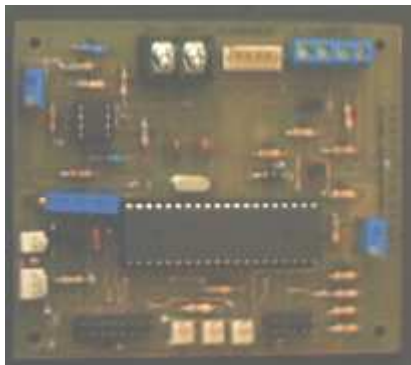


Figura 2.25.- Fotografía real del circuito principal

2.3.7.1.2 Circuito de Interfaz

En la **Figura 2.26** se puede apreciar las pistas generadas por el ARES para esta placa, en la **Figura 2.27** encontramos un esquema gráfico para la parte superior de la placa, y en la **Figura 2.28** tenemos la Fotografía real de la placa elaborada.

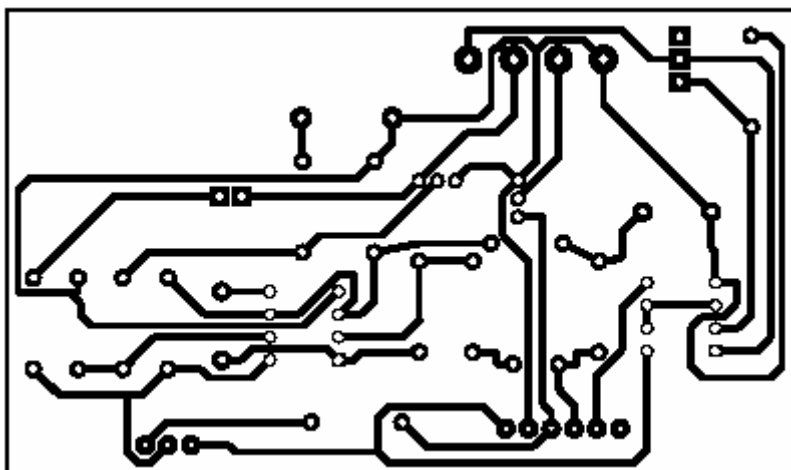


Figura 2.26.- Pistas del Circuito de Interfaz

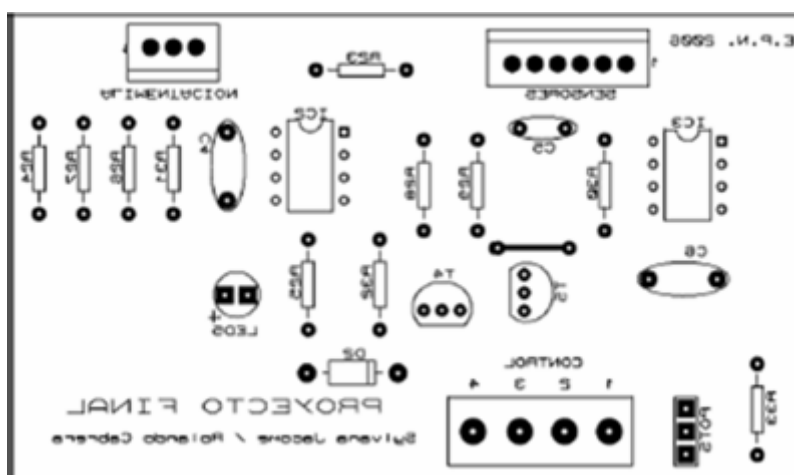


Figura 2.27.- Esquema gráfico de la parte superior del circuito Interfaz



Figura 2.28.- Fotografía real del circuito de interfaz

2.3.7.1.3 Circuito Transmisor - Receptor y Sensor temperatura ambiente

En la **Figura 2.29** se puede apreciar las pistas generadas por el ARES para esta placa, en la **Figura 2.30** encontramos un esquema gráfico para la parte superior de la placa, y en la **Figura 2.31** tenemos la Fotografía real de la placa elaborada.

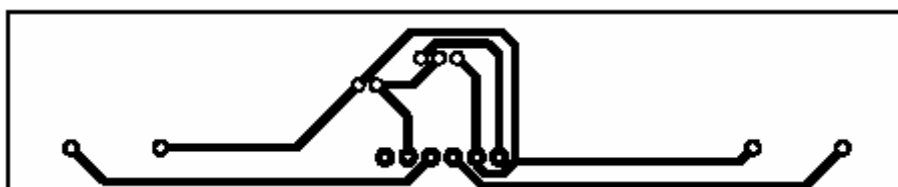


Figura 2.29.- Pistas del circuito ultrasónico

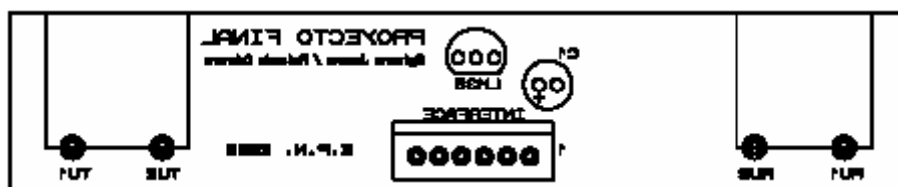


Figura 2.30.- Esquema gráfico de la parte superior del circuito ultrasónico



Figura 2.31.- Fotografía real del Circuito Ultrasónico

2.3.7.1.4 Circuito de Potencia

En la **Figura 2.32** se puede apreciar las pistas generadas por el ARES para esta placa, en la **Figura 2.33** encontramos un esquema gráfico para la parte superior de la placa, y en la **Figura 2.34** tenemos la Fotografía real de la placa elaborada.

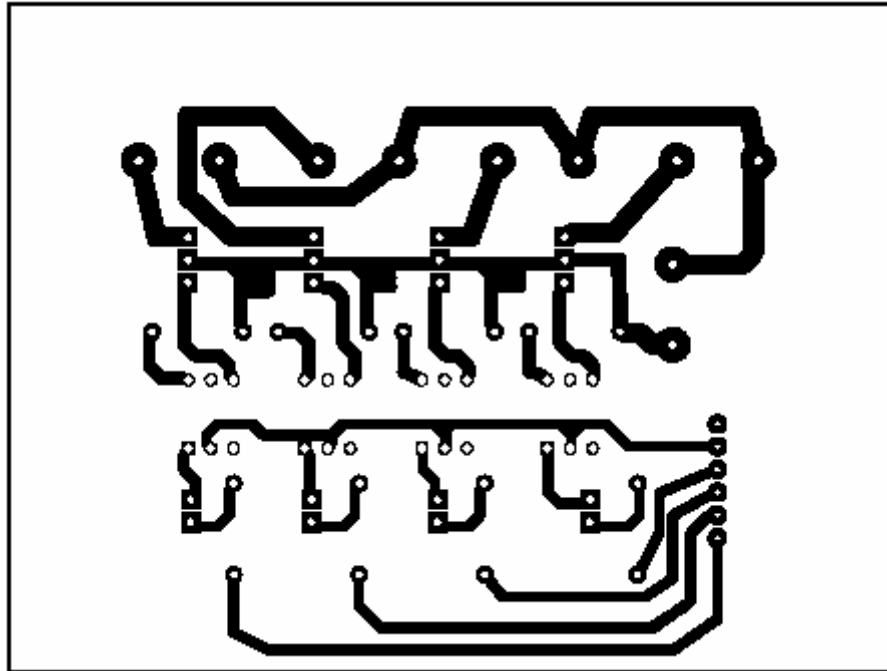


Figura 2.32.- Pistas del circuito de potencia

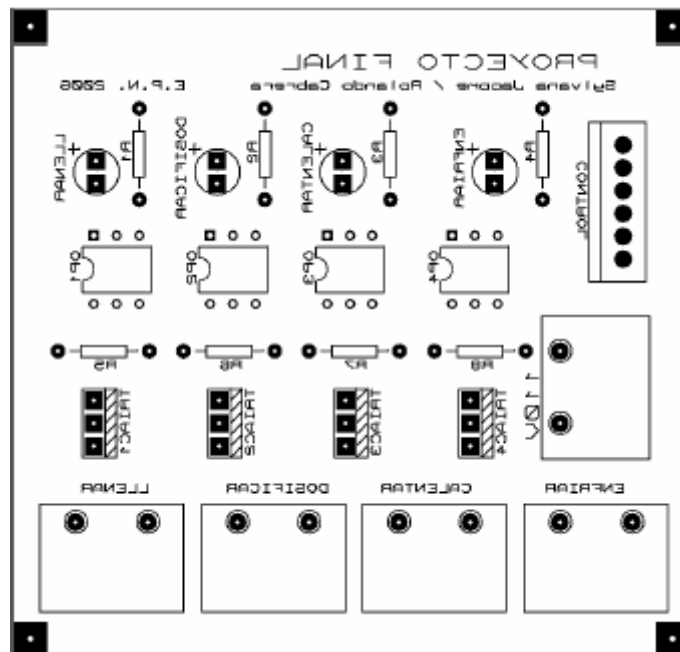


Figura 2.33.- Esquema gráfico de la parte superior del circuito de potencia



Figura 2.34.- Fotografía real del Circuito de Potencia

2.3.7.1.5 Circuito para la Fuente de Alimentación

En la **Figura 2.35** se puede apreciar las pistas generadas por el ARES para esta placa, en la **Figura 2.36** encontramos un esquema gráfico para la parte superior de la placa, y en la **Figura 2.37** tenemos la Fotografía real de la placa elaborada.

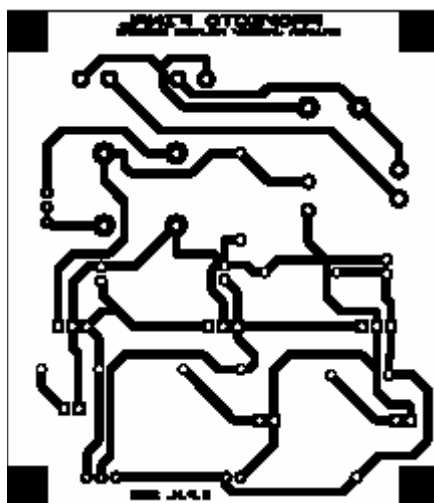


Figura 2.35.- Pistas de la Fuente de Alimentación

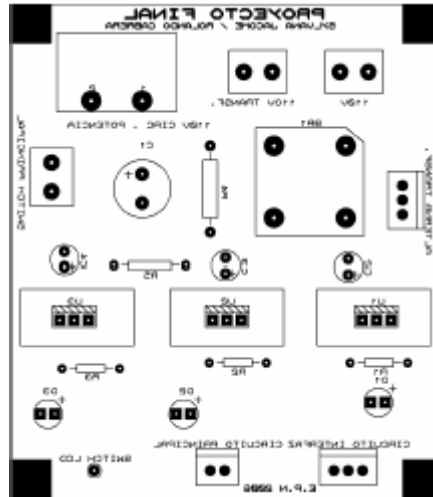


Figura 2.36.- Esquema gráfico de la parte superior de la fuente de alimentación



Figura 2.37.- Fotografía real del Circuito de la Fuente de Alimentación

2.3.8 MONTAJE

Se ha enviado a diseñar un pequeño cilindro transparente de 30cm. de altura por 20cm. de diámetro, en la parte superior del mismo se ha colocado una tapa con una pequeña adecuación en su centro, para colocar la tarjeta con los sensores de ultrasonido y temperatura del ambiente así como la tarjeta de interfaz.

En la parte frontal del tanque contamos con una regleta de 30 cm que nos servirá de guía para las pruebas realizadas, así también en las partes laterales hemos

colocado unos acoples para la conexión de las válvulas de llenado y vaciado del tanque.

A una altura de 3cm del cilindro se realiza una perforación para que la termocupla tipo tornillo pueda ser colocada, el diseño de la misma facilita mucho su instalación ya que únicamente se requiere de un perno en la parte interna del tanque y de unos empaques fuera y dentro del mismo para evitar cualquier fuga de líquido; así también en la parte posterior del cilindro se ha colocado un dispositivo para calentar el agua en el caso de que el usuario así lo requiera

En la figura siguiente se puede apreciar el cilindro con todas las adecuaciones realizadas.



Figura 2.38.- Fotografía real del Cilindro utilizado en el Proyecto

Se diseñó y envió a elaborar también una pequeña caja desde donde el usuario controla los parámetros del tanque usando el teclado y visualizando los datos a partir de la pantalla LCD, en el interior de ésta encontramos la fuente de alimentación, la tarjeta de control de los actuadores y la placa principal.

Desde la parte lateral derecha se pueden manipular y controlar factores como:

- Luz de encendido / apagado de la LCD
- Control de intensidad de la luz de la LCD
- Botón Reset
- Interruptor principal (encendido / apagado de todo el sistema)

En este mismo sector tenemos instalado un parlante que funciona como alarma del sistema.



Figura 2.39.- Fotografía real del Panel de Control

En la figura siguiente se puede apreciar la caja de control con sus respectivas instalaciones.

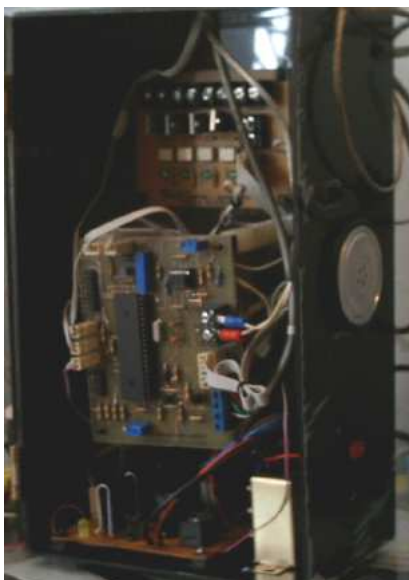


Figura 2.40.- Fotografía real de la Caja de Control con sus respectiva circuitería

CAPÍTULO 3

3.1 PRUEBAS FINALES Y CALIBRACIÓN

De acuerdo a todos los parámetros establecidos anteriormente tenemos una caja de control desde donde se manipulan todos los datos que servirán de información para el programa.

Para la calibración como ya se mencionó anteriormente el voltaje de referencia establecido es de 3.5V, por lo que el primer paso es comprobar que dicho valor se encuentre en el Pin 5 del PIC este valor se lo puede regular con el potenciómetro de precisión conectado al mismo.

Los dispositivos que requieren de calibración son los sensores de temperatura y antes de realizar cualquier prueba se debe regular que la temperatura tanto del ambiente como del líquido sean las correctas así utilizando termómetro de alta escala y manipulando los potenciómetros que regulan los voltajes de la termocupla y del LM35 se puede ajustar estos dispositivos para que midan la temperatura tanto del agua como del ambiente con un mínimo error.

Así, luego de verificar estos aspectos comprobaremos que el PIC cumpla con los parámetros programados.

3.1.1 INGRESO DE DATOS

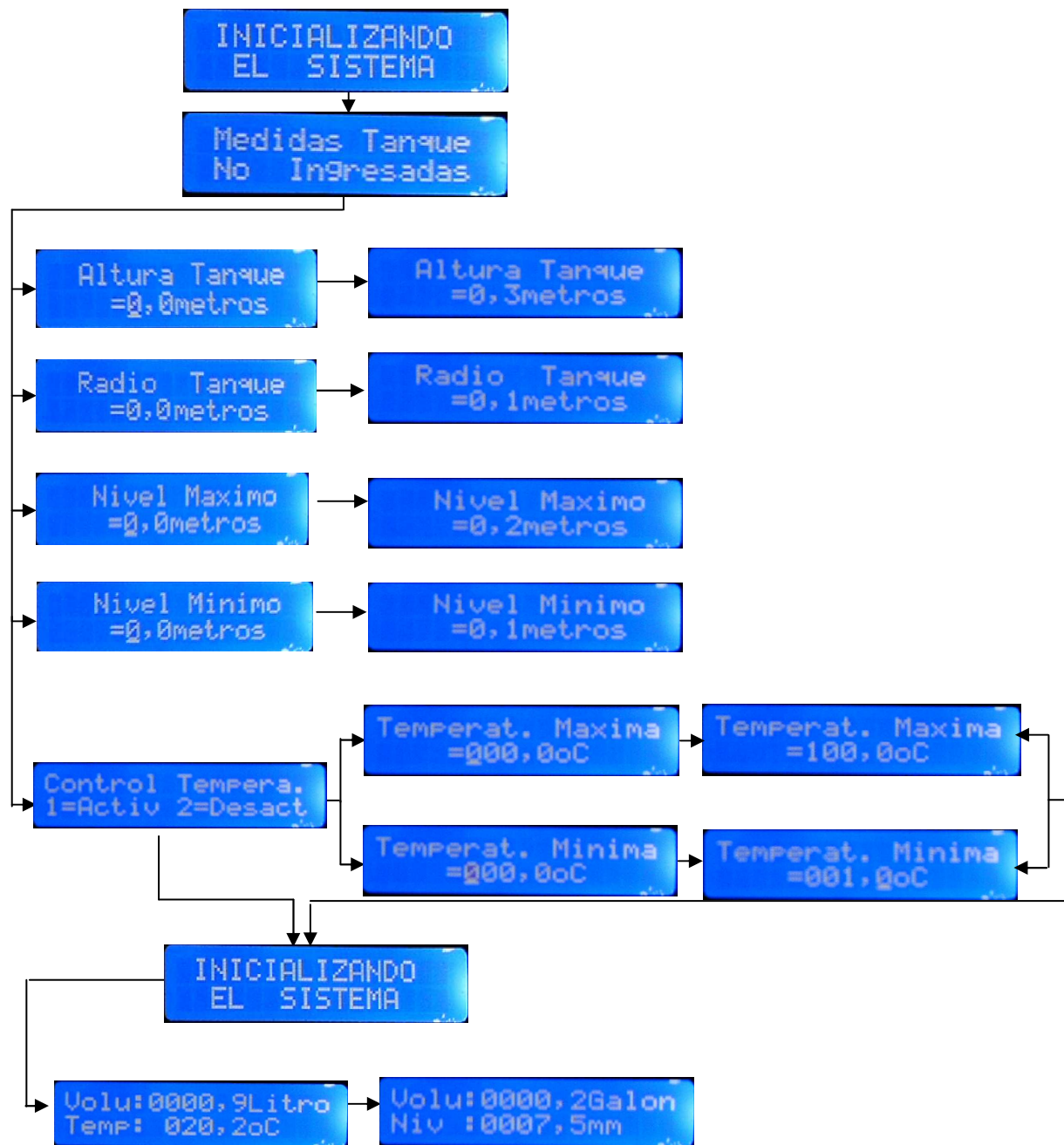


Figura 3.1.- Esquema real para el ingreso de datos

Como se puede observar en la **Figura 3.1** para el ingreso de datos inicialmente se ha programado la visualización de las pantallas **INICIALIZANDO EL SISTEMA** durante 3 segundos y **Medidas Tanque no ingresadas** durante 2 segundos, luego empieza la recepción de los datos donde es necesario ingresar las medidas del cilindro con el cual se va trabajar.

3.1.1.1 Altura del tanque

Para la altura del cilindro se pueden ingresar valores a partir de 0,3 metros hasta 1.8 metros para nuestro caso hemos diseñado un tanque con una altura del mínimo valor admisible, es decir, el valor ingresado es de 0.3 metros. En el caso de no cumplir con los requerimientos de la altura aparece una pantalla que indica que estamos ingresando una medida incorrecta, o sea, fuera del rango y muestra además el rango de valores aceptable, apareciendo nuevamente la pantalla donde se debe ingresar la altura del tanque; esto sucede hasta que la altura del tanque cumpla con las condiciones dadas.

En el **Figura 3.2** podemos observar la pantalla que aparecerá en el caso de sobrepasar los límites requeridos para el ingreso de los datos de la altura del tanque

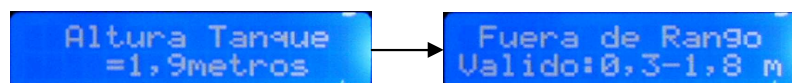


Figura 3.2.- Esquema real para datos que se encuentran fuera del rango

3.1.1.2 Radio del tanque

El siguiente dato a ingresar es el radio del tanque, en este caso el rango aceptable es de 0.1 metros a 1.0 metros; en nuestro caso el valor ingresado es de 0.1 metros que es la medida del radio del cilindro utilizado; en el caso de no estar dentro del rango permitido aparecerá una pantalla que indica que está fuera de los valores aceptables y le indicará los correctos; esto sucederá hasta que los datos ingresados sean los correctos.

En el **Figura 3.3** podemos observar la pantalla que aparecerá en el caso de sobrepasar los límites requeridos para el ingreso de los datos de la altura del tanque



Figura 3.3.- Esquema real para datos que se encuentran fuera del rango

3.1.1.3 Nivel Máximo y Mínimo de líquido

Luego, se debe ingresar tanto el valor del nivel máximo de líquido admitido así como el valor del nivel mínimo; estos valores dependerán de los datos ingresados anteriormente. En el caso de no ingresar los datos correctos se mostrará una pantalla donde le indican el rango de valores aceptable.

En nuestro caso el valor del nivel máximo ingresado es de 0.2 metros y el valor del nivel mínimo ingresado es de 0.1 metros.

En el **Figura 3.4** podemos observar la pantalla que aparecerá en el caso de sobrepasar los límites requeridos para el ingreso de los datos de la altura del tanque

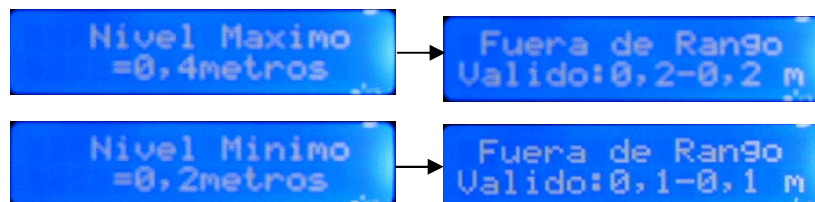


Figura 3.4.- Esquema real para datos que se encuentran fuera del rango

3.1.1.4 Control de la temperatura del líquido

El control de la temperatura del líquido se lo puede hacer también desde este panel, así la primera opción será determinar si se desea activar o no el control de la temperatura.

En el caso de activar el control de la temperatura el usuario deberá ingresar el valor máximo y mínimo al que se encontrará el líquido, estos valores pueden oscilar entre los 5°C hasta 100°C requiriendo de una mínima diferencia de 5°C entre el valor máximo y mínimo ingresado; de la misma manera que en los casos anteriores si no se ingresa los datos correctos aparecerá una pantalla con los valores permitidos hasta que los datos ingresados sean los indicados.

Este es el paso final del ingreso de datos luego de esto aparecerá en la pantalla el volumen del líquido en litros y la temperatura en °C y para visualizar más datos aplastaremos la letra **B** del teclado que cumple con la función de desplazar la información para observar el resto de la información que no es más que el

volumen en galones y el nivel en milímetros. En el caso de que la temperatura del líquido sea menor que la temperatura mínima aceptable se activará uno de los actuadores que permite calentar el líquido hasta que el valor de la temperatura esté dentro de los valores correctos, así también si la temperatura del líquido es mayor a la máxima aceptable se activará el actuador que permita enfriar el líquido hasta que se encuentre dentro del valor máximo permitido

En el **Figura 3.5** podemos observar la pantalla que aparecerá en el caso de sobrepasar los límites requeridos para el ingreso de los datos de la altura del tanque

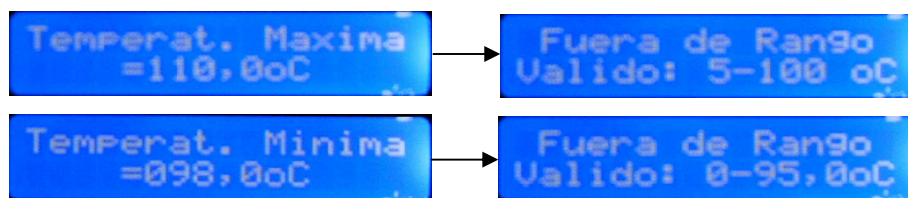


Figura 3.5.- Esquema real para datos que se encuentran fuera del rango

En el caso de optar por desactivar el control de la temperatura la información descrita anteriormente se mostrará automáticamente descartando la posibilidad de controlar la temperatura del líquido.

3.1.1.5 Menú Principal

Al digitar la letra **D** del teclado matricial observaremos una pantalla con el texto **MENU PRINCIPAL**, aquí nosotros podremos escoger entre las siguientes opciones:

- 1 Cargar Líquido en el tanque
- 2 Descargar Líquido del Tanque
- 3 Ingresar dimensiones de Tanque
- 4 Fijar Niveles de Líquido
- 5 Activar Desactivar Control Temperatura
- 6 Fijar Temperatura de Líquido
- 7 Prueba de Dispositivo

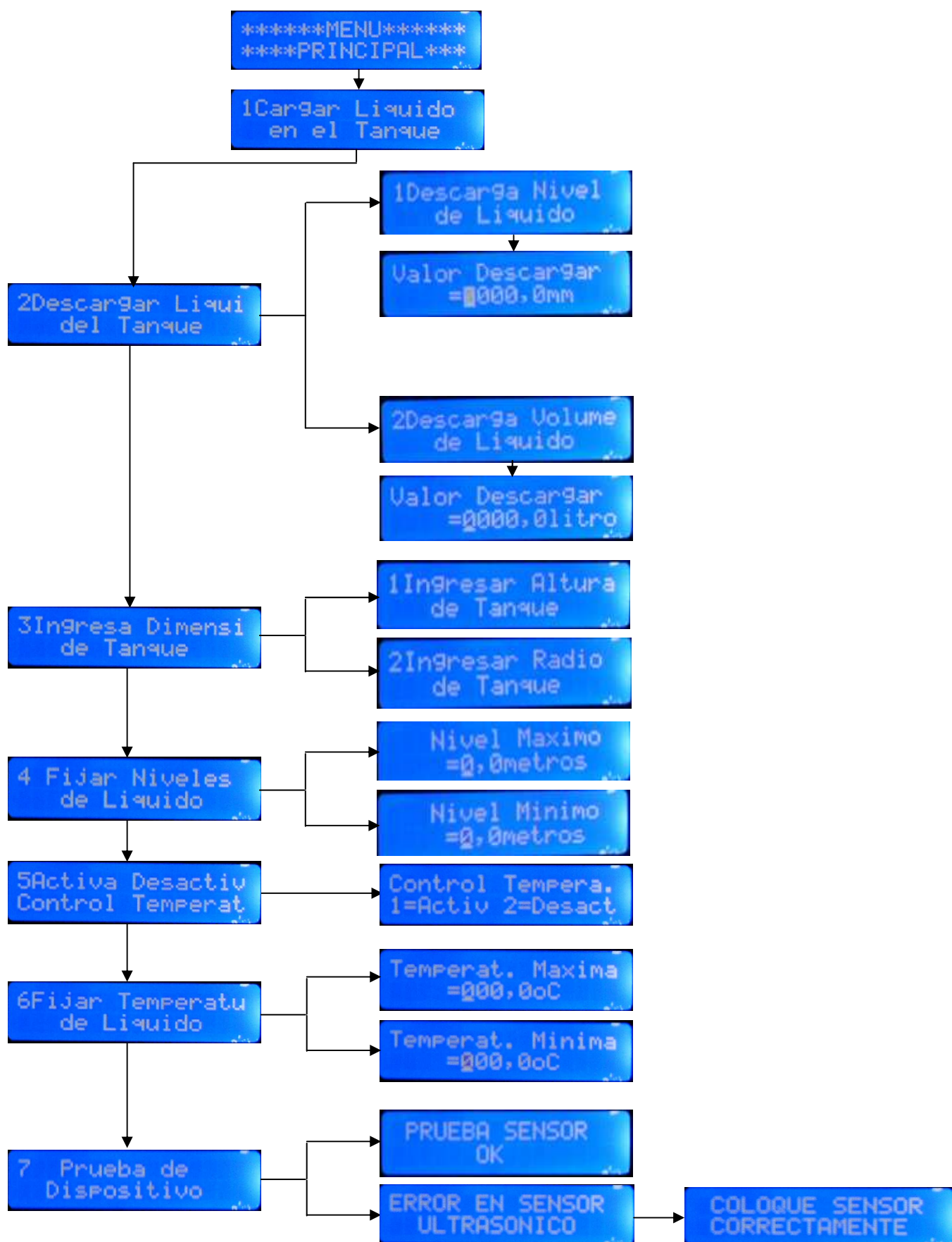


Figura 3.6.- Esquema real del MENÚ PRINCIPAL

En el **Figura 3.6** se puede observar el esquema real que se visualiza en el display al utilizar el Menú Principal

3.1.1.5.1 Cargar Líquido en el Tanque

Al escoger esta opción el PIC envía una señal al circuito de potencia que activa la bomba y permite cargar de líquido al tanque hasta llegar al valor del nivel máximo establecido.

3.1.1.5.2 Descargar Líquido del Tanque

Si se requiere descargar el líquido del tanque se lo puede realizar de dos maneras:

- 1 Descarga Nivel de Líquido
- 2 Descarga Volumen de Líquido

Al escoger Descarga Nivel de Líquido se introduce la cantidad a descargar en milímetros en cambio al escoger Descarga Volumen de Líquido la cantidad a descargar será en Litros.

3.1.1.5.3 Ingresar Dimensiones del Tanque

Esta opción nos permite reajustar los datos introducidos anteriormente ya que se cuenta con estas dos opciones:

- 1 Ingresar altura de Tanque
- 2 Ingresar radio de Tanque

3.1.1.5.4 Fijar Niveles de Líquido

Esta opción nos permite reajustar los datos introducidos anteriormente en cuanto a nivel máximo y mínimo del líquido.

3.1.1.5.5 Activar Desactivar Control Temperatura

Esta opción nos permite activar o desactivar el control de la temperatura del líquido o simplemente establecer nuevos rangos para el control de la misma.

3.1.1.5.6 Prueba de Dispositivo

Se realiza una prueba de los sensores ultrasónicos que muestra **PRUEBA SENSOR OK** cuando no existe ningún problema tanto en la posición como en el funcionamiento de los mismos, así también muestra **ERROR SENSOR ULTRASÓNICO** y **COLOQUE SENSOR CORRECTAMENTE** cuando existe algún problema con dichos sensores.

Al realizar estas tareas se ha comprobado el correcto funcionamiento de cada uno de los dispositivos así como del programa realizado.

Es necesario recalcar que los datos son ingresados únicamente al iniciar el sistema por primera vez o cuando se decida cambiar cualquier parámetro desde el menú principal, de ahí en adelante los datos se quedan grabados en la memoria EEPROM del PIC.

3.1.1.6 Reset

El RESET provoca la re-inicialización de todo el sistema, en este caso al utilizarlo aparecerá una pantalla donde se indica que **EL SISTEMA HA SIDO REINICIADO** y se solicita que el se **APAGUE EL SISTEMA** al usar este botón se borran los datos ingresados de la memoria y al encender el sistema es necesario ingresar todos los datos nuevamente. En la figura siguiente se puede apreciar las pantallas reales que se visualizan al resetear el sistema.

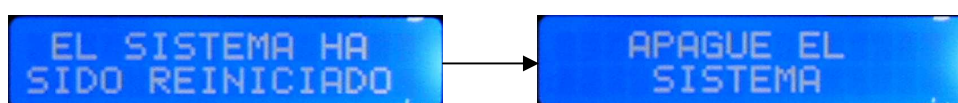


Figura 3.7.- Esquema real de las pantallas visualizadas al ejecutar RESET

3.1.1.7 Alarma

Se ha implementado una alarma que sirve para indicar que el nivel de líquido esta bajo el nivel mínimo designados por el usuario, el sonido emitido por el pequeño parlante se lo escuchará hasta que el nivel de líquido alcance o sobrepase el valor mínimo elegido.

3.2 RESULTADOS

Todas las pruebas realizadas han permitido efectuar ciertas correcciones en cuanto al funcionamiento de ciertos dispositivos e incluso del programa hasta obtener un resultado óptimo

Tanto la circuitería adecuada al prototipo como el programa grabado en el PIC funcionan correctamente y se puede conocer el volumen, temperatura, nivel del líquido de una forma fácil y certera. Así también se puede extraer el líquido en cantidades discretas sin tener que manipularlo.

Todos los resultados y errores de las pruebas realizadas pueden observarse de una manera gráfica en los ANEXO H: ERROR DE NIVEL DEL LÍQUIDO y ANEXO I: ERROR DE TEMPERATURA DEL LÍQUIDO

3.3 MANUAL DEL USUARIO

3.3.1 INTRODUCCIÓN

Con la siguiente información el usuario podrá manipular fácilmente el prototipo que controla la temperatura, el almacenamiento y extracción de líquidos; dependiendo del proceso que desee realizar el interesado estará en capacidad de realizar cualquier tarea desde el panel de control, siempre y cuando esté dentro de los márgenes establecidos.

3.3.2 INSTALACIÓN

- Coloque los tanques y panel de control en lugar firme tratando de evitar movimientos bruscos.
- Llene el TANQUE 1 con agua (**Figura 3.9**)
- Coloque correctamente la tapa del TANQUE 2, de tal manera que coincida la regleta de la tapa con la del recipiente (**Figura 3.10**)
- Conecte la caja de control a 110V (**Figura 3.8**)

3.3.3 INSTRUCCIONES

- Ingrese las dimensiones del tanque desde el teclado (la unidad de medida será el metro [m]), tomando en cuenta los valores mínimos y máximos requeridos.
 - Altura
 - Radio
 - Nivel Máximo
 - Nivel Mínimo

Si usted ingresó una medida errónea, usted el rango de medidas permitido y podrá ingresar nuevamente el valor.

- Tiene la posibilidad de habilitar o deshabilitar el control de la temperatura, en el caso de que esta opción sea habilitada es necesario que conecte el calentador de agua del TANQUE 2 a 110V y establezca los valores temperatura mínimo y máximo.
- Luego de ingresar las dimensiones del tanque, automáticamente el TANQUE 1 se llenará hasta el nivel máximo establecido por el usuario, de aquí en adelante usted podrá visualizar en la pantalla LCD la cantidad de líquido existente en dicho tanque, expresado en unidades de volumen

como son galones o litros, además podrá conocer el valor del nivel del líquido expresado en milímetros, así como también podrá visualizar la temperatura en grados centígrados, esto se logra con la ayuda de las flechas del teclado matricial como se puede observar en la **Figura 3.11**

- Para utilizar las demás opciones que presenta el prototipo puede hacerlo presionando la tecla MENÚ (Letra D del teclado) para acceder al MENÚ PRINCIPAL, desde donde usted podrá:
 1. Cargar Líquido en el tanque
 2. Descargar Líquido del Tanque
 3. Ingresar dimensiones de Tanque
 4. Fijar Niveles de Líquido
 5. Activar Desactivar Control Temperatura
 6. Fijar Temperatura de Líquido
 7. Prueba de Dispositivo

Utilizando la tecla ENTER se puede optar por cualquiera de éstas alternativas.

- La opción 1 le permite **Cargar Líquido en el Tanque (TANQUE 2)**, hasta el nivel máximo establecido por el usuario.
- La opción 2 le permite **Descargar Líquido del Tanque**, ya sea en Nivel o Volumen; esto depende de sus necesidades. Si escoge la opción DESCARGA POR NIVEL entonces el sistema muestra una pantalla donde se puede ingresar el valor de nivel en milímetros de líquido a descargar; por otro lado si se escoge la opción DESCARGA POR VOLUMEN entonces el sistema muestra una pantalla donde se puede ingresar el valor de volumen en litros a descargar.
- Con la opción 3 **Ingresar Dimensiones del Tanque** usted podrá cambiar las dimensiones del **TANQUE 2** (Altura, radio), de tal suerte que al variar el valor de la altura del tanque será necesario que se

ingrese nuevamente los valores del nivel máximo y nivel mínimo del tanque.

- Es posible variar el valor de los niveles máximo y mínimo del líquido en el **TANQUE 2** con la opción 4 **Fijar Niveles de Líquido**.
 - La opción 5 le permite activar o desactivar el control de la temperatura, en el caso de que el control de la temperatura esté desactivado no se mostrará la opción 6 **Fijar Temperatura de Líquido** que hace referencia a los rangos máximo y mínimo de temperatura.
 - Se puede realizar la prueba del dispositivo al escoger la opción 7 del **MENÚ PRINCIPAL Prueba de Dispositivo**.
- La caja de control tiene un pulsador **RESET (Figura 3.8)**, que reinicializa todo el sistema. Por seguridad se puede acceder a éste con un pequeño destornillador y luego de ser presionado es necesario apagar el equipo y volverlo a encender e ingresar todos los datos.
 - La pantalla LCD tiene una luz azul de fondo, usted puede activarla o desactivarla por medio del **Swich LCD (Figura 3.8)**, así también la intensidad de la luz de la pantalla LCD también se la puede controlar de acuerdo a sus necesidades por medio de un **Potenciómetro LCD (Figura 3.8)**.

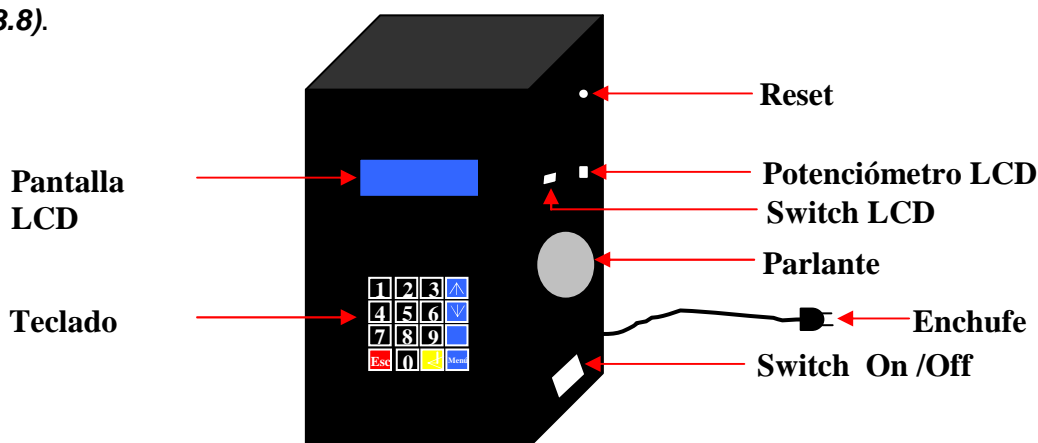


Figura 3.8.- Esquema y Partes de la Caja de Control

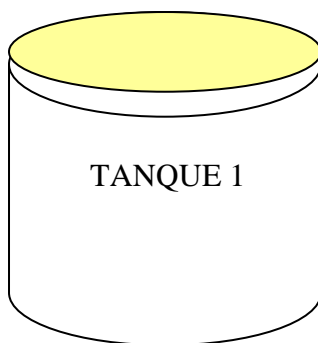


Figura 3.9.- Esquema básico del TANQUE 1

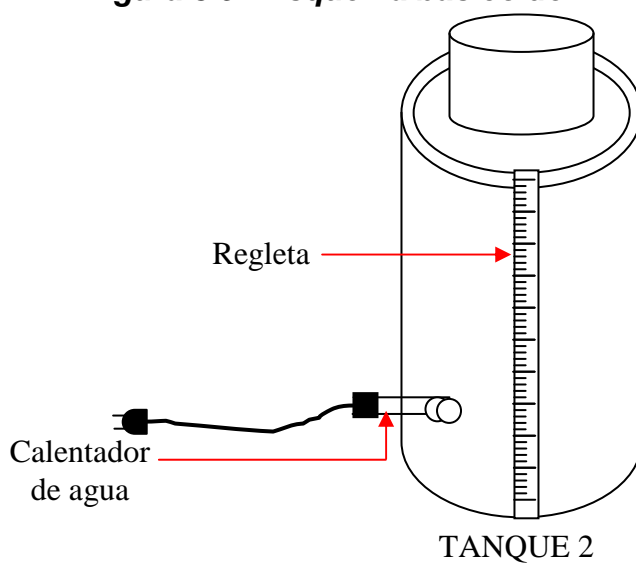


Figura 3.10.- Esquema básico del TANQUE 2

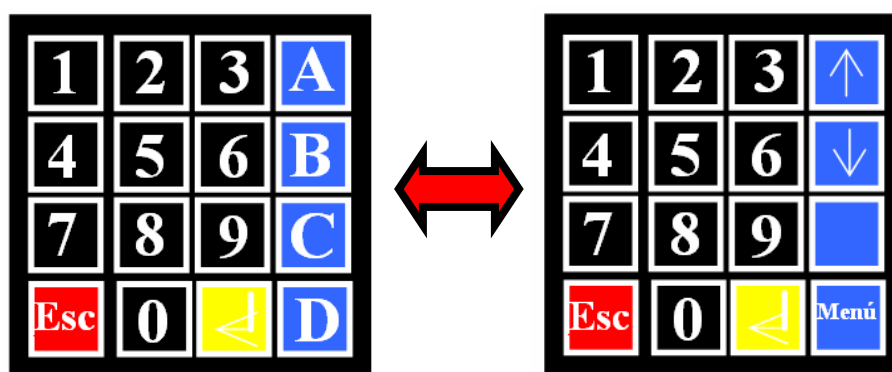


Figura 3.11.- Esquema Real del teclado y equivalencia para cada tecla

CONCLUSIONES Y RECOMENDACIONES

Al finalizar el diseño y construcción de este proyecto se pudo capturar las siguientes conclusiones y recomendaciones:

CONCLUSIONES

- Se pudo conseguir una herramienta tecnológica que al ser implementada en la vida real, puede resultar una muy buena inversión para aquellas empresas que deseen una alternativa a bajo costo y muy buenas prestaciones para el manejo y control de los líquidos.
- Con la utilización de este sistema se puede dejar a un lado la utilización de técnicas vetustas para realizar la labor de medición de nivel de líquidos ya que por medio de este sistema se obtiene de una forma rápida y certera el valor de nivel del líquido, así como su volumen y la temperatura a la que se encuentra el mismo.
- Fundamentalmente el sistema posibilita el manejo automático de líquidos, que puedan resultar peligrosos para el ser humano o que sean de uso restringido, minimizando así el riesgo de accidentes a causa del mal manejo de este tipo de sustancias.
- Todos los dispositivos electrónicos que forman parte del prototipo implementado, son comunes en el mercado nacional por lo que existe la posibilidad de que sus elementos sean reemplazados sin mayores inconvenientes y por ende pueda ser reparado con facilidad.
- La programación del dispositivo microcontrolador a través de diagramas de flujo permitió optimizar muchísimo tiempo, ya que para realizar todas las tareas planteadas en ésta tesis se requería proyectar un sin número de instrucciones; así NIPLE generó el código assembler adecuado para que el

dispositivo de control pueda realizar todas las tareas y aplicaciones necesarias para este proyecto.

- Al escoger el dispositivo de comando para este proyecto fue muy importante tomar en cuenta todas las prestaciones que requeríamos para implementar todos los dispositivos externos así como las características de memoria, velocidad, conversores, etcétera; así el microprocesador PIC16F877 posee las características que permitieron la correcta adecuación de todas las herramientas y dispositivos utilizados en el proyecto, siendo además muy versátil, eficiente y práctico.
- Al tratarse de un prototipo se instaló un sencillo y económico calentador de agua que al ser activado calienta mucho más rápido el líquido que se encuentra a su alrededor, por lo que su temperatura varía de acuerdo al lugar donde se tome la medida, así la termocupla al estar colocada en la pared del cilindro toma cierta medida que no es exactamente la temperatura promedio del líquido, para evitar que la temperatura del líquido varíe de acuerdo a la posición o lugar donde se realice la medida, se debería colocar más de una termocupla ubicadas en sitios estratégicos para poder tener una cuantas medidas y luego sacar el promedio de éstas, pero no se optó por ello ya que es un prototipo y sobre todo porque en las industrias no se utilizan este tipo de calentadores de agua, mas bien se utilizan métodos que activan o desactivan ciertas válvulas para el calentamiento homogéneo del líquido y nuestro proyecto le permite al usuario conectar y activar cualquier dispositivo siempre y cuando trabaje con el voltaje establecido de 110V y maneje una corriente máxima de 25A
- Es importante también recalcar que a pesar de que el control establecido para la temperatura se encuentra entre los 0°C a 100°C, en las pruebas resulto imposible llegar a éstos límites, tomando en cuenta también que en Quito el agua hierve a 91°C; así en las pruebas realizadas se trabajó con temperaturas desde los 10°C hasta los 40°C, teniendo buenos resultados.

RECOMENDACIONES

- Aunque Niple facilitó mucho la programación como lo demostramos al realizar el programa de control para este prototipo, la primera recomendación y observación que debemos indicar es que en la fase de desarrollo se pudo notar ciertas falencias que posee el software NIPLE, las mismas que fueron reportadas al desarrollador del mismo para su respectiva revisión y solución.
- Este tema fue muy interesante y hemos incrementado nuestros conocimientos con la ejecución del mismo, pero recomendamos a nuestros compañeros que al realizar su proyecto de titulación busquen un tema que no sea tan complejo, sobre todo por el tiempo ya que éste fue uno de nuestras principales desventajas en el diseño y construcción del prototipo.
- Este proyecto puede ser el punto de partida para futuras investigaciones, como puede ser el control de ciertas sustancias que por su esencia no puedan ser controladas por este dispositivo, como por ejemplo sustancias corrosivas u otras que puedan causar daños, sobretodo en los sensores ultrasónicos, elementos fundamentales en la implementación de este proyecto.
- Así mismo se puede pensar en la mejora del presente proyecto, implantando ciertas características, que otros dispositivos lo poseen. Como por ejemplo se puede plantear construir este dispositivo a prueba de explosiones; características que para ciertas aplicaciones puede ser necesaria y primordial.
- Indiscutiblemente la implementación de este sistema en la vida real podría llegar a ser un tema para la presentación de un nuevo proyecto, ya que al tratarse de un prototipo, este fue desarrollado en un esquema de requerimientos muy generales, sería interesante si se plantea la idea de

llevar a la realidad este sistema realizando un estudio sobre la viabilidad de montar este proyecto destacando las necesidades puntuales que pueda tener cualquier empresa.

BIBLIOGRAFÍA

REVISTAS:

- GOÑI FERNÁNDEZ, Miguel. “ELECTRÓNICA Microcontroladores y Microprocesadores”. MP Multipress

INTERNET:

- Elektronika dla Wszystkich. Projekty AVT. KIT 2643. Sonar dalmierz ultradźwiękowy. Lipiec 2002. (Archivo *.PDF)
- <http://www.national.com/pf/LM/LM35.html>
- http://www.unicrom.com/Tut_optoacoplador.asp
- http://www.unicrom.com/Tut_triac.asp
- <http://www.x-robotics.com/rutinas.htm>
- <http://www.monografias.com/trabajos18/descripcion-pic/descripcion-pic.shtml>
- http://autric.com/Microbotica%20y%20Mecatronica/descripcion_global.htm
- <http://www.monografias.com/trabajos14/dispositivos/dispositivos.shtml>
- <http://www.qsl.net/cx1ddr/7812.htm>
- http://www.unicrom.com/Tut_ElevacionTension78XX.asp
- <http://www.monografias.com/trabajos14/micros/micros.shtml>
- <http://www.monografias.com/trabajos15/actuadores/actuadores.shtml>
- <http://enciclopedia.us.es/index.php/Ultrasonido>
- http://www.interq.or.jp/japan/se-inoue/e_pic877.htm
- <http://www.niplesoft..net>
- <http://www.warburtech.com/microcontrollers/40.pin.pic.microcontrollers/pic16f877.microcontroller.htm>

CONTENIDO

	Página
PRESENTACIÓN	1
RESUMEN	5
CAPÍTULO 1	7
1.1 INTRODUCCIÓN	7
1.2 ANTECEDENTES Y REFERENCIAS	8
1.3 MARCO TEÓRICO	9
1.3.1 MICROCONTROLADOR PIC	9
1.3.1.1 Definición	9
1.3.1.2 Arquitectura	9
1.3.2 SISTEMAS DE DESARROLLO	9
1.3.2.1 Los Simuladores	10
1.3.2.2 El Grabador	10
1.3.3 CONCEPTOS BÁSICOS	10
1.3.3.1 Resistor	10
1.3.3.2 Condensador	10
1.3.3.3 Diodo	11
1.3.3.4 Diac	11
1.3.3.5 Triac	12
1.3.3.6 Optoacoplador	13
1.3.3.7 El Amplificador Operacional	13
1.3.4 NOCIONES BÁSICAS	14
1.3.4.1 Magnitudes físicas	14
1.3.4.1.1 Distancia	14
1.3.4.1.2 Tiempo	14
1.3.4.1.3 Velocidad	14
1.3.4.2 Velocidad del sonido en el aire	15
1.3.4.3 Cilindro Recto	15
1.3.4.3.1 Definición	15
1.3.4.3.2 Cálculo del volumen de un cilindro	15
1.3.5 SENSORES	16
1.3.5.1 Definición	16
1.3.6 ACTUADORES	16
1.3.6.1 Definición	16
1.3.7 REGULADORES DE VOLTAJE	17
1.3.7.1 Definición	17
CAPÍTULO 2	18
2.1 SOLUCIÓN DEL PROBLEMA	18
2.1.1. MICROCONTROLADOR Y HERRAMIENTAS EMPLEADAS	18
2.1.1.1 PIC 16F877	18
2.1.1.2 Sistemas de Desarrollo NIPLE	21
2.1.1.3 Los Simuladores	22
2.1.1.4 El Grabador	23
2.1.2. CONTROL DE NIVEL DE LÍQUIDO	25
2.1.2.1 Sensores Ultrasónicos	25
2.1.2.1.1 Funcionamiento	26
2.1.2.1 Sensor de Temperatura para el Ambiente	27
2.1.2.1.2 Circuito Integrado LM35	28
2.1.3. CONTROL DE LA TEMPERATURA DEL LÍQUIDO	30
2.1.3.1 Termocupla	30
2.1.4. EXTRACCIÓN Y LLENADO DE LÍQUIDO	32
2.1.4.1 Extracción segura de líquido	32
2.1.4.1.1 Electro - Válvula	32
2.1.4.2 Llenado de líquido	33
2.1.4.2.1 Bombas	34
2.1.5. ELEMENTOS DE INTERFAZ	34
2.1.5.1 Teclado Matricial 4x4	34

2.1.5.1.1	Descripción	35
2.1.5.1.2	Funcionamiento	35
2.1.5.2	Pantalla LCD 2x16	36
2.1.5.2.1	Descripción	36
2.1.5.2.2	Configuración	36
2.2	DISEÑO DEL PROTOTIPO	37
2.2.1	PROGRAMA PRINCIPAL	38
2.2.2	EXPLICACIÓN DEL DIAGRAMA	39
2.2.3	ERRORES PRODUCIDOS EN LOS CÁLCULOS REALIZADOS	46
2.3	CONSTRUCCIÓN DEL PROTOTIPO	49
2.3.1	CIRCUITO PRINCIPAL	49
2.3.2	CONTROL DEL NIVEL DE LÍQUIDO	49
2.3.3	CONTROL DE LA TEMPERATURA DEL AIRE	51
2.3.4	CONTROL DE LA TEMPERATURA DEL LÍQUIDO	52
2.3.5	CONTROL DE ACTUADORES	53
2.3.6	FUENTE DE ALIMENTACIÓN	53
2.3.7	ELABORACIÓN DE PLACAS	54
2.3.7.1	Placas Realizadas	56
2.3.7.1.1	Circuito Principal	56
2.3.7.1.2	Circuito de Interfaz	56
2.3.7.1.3	Circuito Transmisor - Receptor y Sensor temperatura ambiente	58
2.3.7.1.4	Circuito de Potencia	58
2.3.7.1.5	Circuito para la Fuente de Alimentación	60
2.3.8	MONTAJE	61
	CAPÍTULO 3	64
3.1	PRUEBAS FINALES Y CALIBRACIÓN	64
3.1.1	INGRESO DE DATOS	65
3.1.1.1	Altura del tanque	66
3.1.1.2	Radio del tanque	66
3.1.1.3	Nivel Máximo y Mínimo de líquido	67
3.1.1.4	Control de la temperatura del líquido	67
3.1.1.5	Menú Principal	68
3.1.1.5.1	Cargar Líquido en el Tanque	70
3.1.1.5.2	Descargar Líquido del Tanque	70
3.1.1.5.3	Ingresar Dimensiones del Tanque	70
3.1.1.5.4	Fijar Niveles de Líquido	70
3.1.1.5.5	Activar Desactivar Control Temperatura	71
3.1.1.5.6	Prueba de Dispositivo	71
3.1.1.6	Reset	71
3.1.1.7	Alarma	72
3.2	RESULTADOS	72
3.3	MANUAL DEL USUARIO	72
3.3.1	INTRODUCCIÓN	72
3.3.2	INSTALACIÓN	73
3.3.3	INSTRUCCIONES	73
	CONCLUSIONES Y RECOMENDACIONES	77
	BIBLIOGRAFÍA	81

ANEXOS

ANEXO A: PLAN DEL PROYECTO.....	I
ANEXO B: DATASHEET DEL PIC 16F877.....	X
ANEXO C: MANUAL DEL USUARIO DEL SOFTWARE "NIPLÉ"	XXVIII
ANEXO D: SENSORES DE ULTRASONIDO (136653CK)	XXXIV
ANEXO E: LM35.....	XXXV
ANEXO F: DISTRIBUCIÓN DE PINES DE LCD (2X16).....	XLI
ANEXO G: REGULADORES DE VOLTAJE	XLII
ANEXO H: ERROR DE NIVEL DEL LÍQUIDO	XLVI
ANEXO I: ERROR DE TEMPERATURA DEL LÍQUIDO	XLVII
ANEXO J: COSTOS REALES	XLVIII

ANEXO A: PLAN DEL PROYECTO

PROPUESTO POR: Jácome Cadena Sylvana Maritza Cabrera Peñaherrera Rolando Vladimir	ÁREA DEL TEMA: Electrónica e Introducción a la programación
AUSPICIADO POR:	FECHA:

1) Tema o Título del Proyecto

Construcción de un Prototipo que controle la temperatura, el almacenamiento y extracción de líquidos en un tanque.

2) Planteamiento del Problema

Las empresas ecuatorianas, con el fin de sobrevivir en un mercado cada vez más competitivo, tienen la obligación de implementar tecnologías que ayuden a mejorar y optimizar sus procesos a un bajo costo, con el único fin de conseguir productos de mejor calidad a un costo razonable.

Existen algunas empresas que aún usan métodos ambiguos y obsoletos para la manipulación y medición del nivel de líquidos, técnicas que por ser prácticamente manuales no brindan precisión, ni confiabilidad. De la misma manera, muchas otras empresas a pesar de contar con instrumentos sofisticados para el manejo y control de líquidos, tienen el inconveniente de que estos son dispositivos costosos y de muy difícil reparación.

Por tales razones se pretende construir el modelo de un sistema que controle el almacenamiento de líquidos en un tanque, en el cual se implementará un medidor de nivel, un medidor de temperatura y un dosificador. Todos estos dispositivos

serán monitoreados y controlados por un microcontrolador, el cual a la vez servirá de interfaz para la comunicación con un panel principal, donde se podrá conocer los datos del nivel del líquido, además de la temperatura a la que se encuentra el mismo y nos permitirá extraer el líquido en cantidades discretas.

Este prototipo puede convertirse en una alternativa para facilitar el desempeño de dichas empresas en cuanto a tiempo, trabajo y costo, parámetros que pueden estar siendo desaprovechados en la consecución manual de los datos, creándose así la alternativa de automatizar los procesos restantes que se realicen en la empresa.

Por otra parte al ser un sistema construido básicamente con dispositivos electrónicos comunes en el mercado nacional, existe la posibilidad de ser reparado con facilidad y que sus elementos sean reemplazados sin mayores inconvenientes; además los factores mencionados con anterioridad hacen de este un sistema de bajo costo en comparación con otros sistemas conocidos.

3) Sistematización del Problema

1. ¿Cuáles son los principales parámetros que se deben controlar?
2. ¿Qué características tendrá el sistema de medición de nivel, de temperatura y dosificador?
3. ¿Qué sensores se pueden adecuar a la aplicación para poder medir el nivel de los líquidos y la temperatura?
4. ¿Qué dispositivo puede aplicarse como dosificador del líquido?
5. ¿Qué valores de nivel necesitan ser medidos? Y ¿cuáles son los más frecuentes?
6. ¿Qué valores de temperatura necesitan ser medidos? Y ¿cuáles son los más frecuentes?
7. ¿Qué cantidades de líquido necesitan ser dosificados desde el tanque contenedor? Y ¿cuáles son los más frecuentes?
8. ¿Qué características tendrá el sistema de control?, y ¿de qué modo facilitará el uso de este sistema a los usuarios?

9. ¿Cuáles son las ventajas y desventajas que originaría el diseño de dicho dispositivo?

10. ¿Cuán eficaz resulta este sistema de control?

4) Objetivos de la Investigación

a) Objetivo General

Proveer de una alternativa tecnológica de bajo costo, de fácil de reparación y rendimiento óptimo, que facilite la obtención de los datos en forma rápida y automática permitiendo el control adecuado de los líquidos; mediante el diseño y modelación de un tanque contenedor de líquidos que cuente con un medidor de nivel, un dosificador de líquidos y un medidor de temperatura; todos monitoreados y controlados desde un Panel Principal.

b) Objetivos Específicos

1. Determinar los principales parámetros que intervienen en el sistema, para de esta manera evaluar las ventajas y desventajas del detector de nivel, dosificador de líquidos, y del medidor de temperatura e identificar las necesidades de medición que requieren los técnicos, y priorizarlas.
2. Mostrar los datos del nivel y temperatura a la que se encuentra el líquido, en una pantalla LCD, así como otros datos anexos dependientes de los requerimientos de los técnicos.
3. Controlar un dosificador de líquidos desde un panel diseñado con todos los parámetros necesarios para permitir de ésta manera la extracción del líquido contenido en el tanque en cantidades discretas.
4. Establecer las características y dificultades que tendrá el sistema.

5. Realizar las pruebas necesarias que permitan determinar la eficiencia del sistema.

5) Justificación Práctica.

El tanque contenedor de líquidos con medidor de nivel, de temperatura y un dosificador, constituye un proyecto muy conveniente para ciertos sectores empresariales, ya que la implementación de este sistema facilitará y automatizará ciertas funciones que se desempeñan de manera manual como es la extracción de sustancias y permitirá llevar un control del nivel y temperatura de ciertas sustancias.

A la vez este sistema permitirá optimizar tiempo y dinero apoyando al proceso de mejoramiento continuo y como resultado de ello incrementar la rentabilidad de la empresa ecuatoriana.

6) Marco Teórico Referencial.

1. Referencias Informativas.

1.1 Definición de conceptos.

MICROCONTROLADOR

Un microprocesador no es un ordenador completo. No contiene grandes cantidades de memoria ni es capaz de comunicarse con dispositivos de entrada. Un tipo diferente de circuito integrado llamado microcontrolador es de hecho una computadora completa situada en un único chip, que contiene todos los elementos del microprocesador básico además de otras funciones especializadas.

Un microcontrolador consta de varias secciones diferentes. La unidad aritmético-lógica efectúa cálculos con números y toma decisiones lógicas; los registros son zonas de memoria especiales para almacenar información temporalmente; la unidad

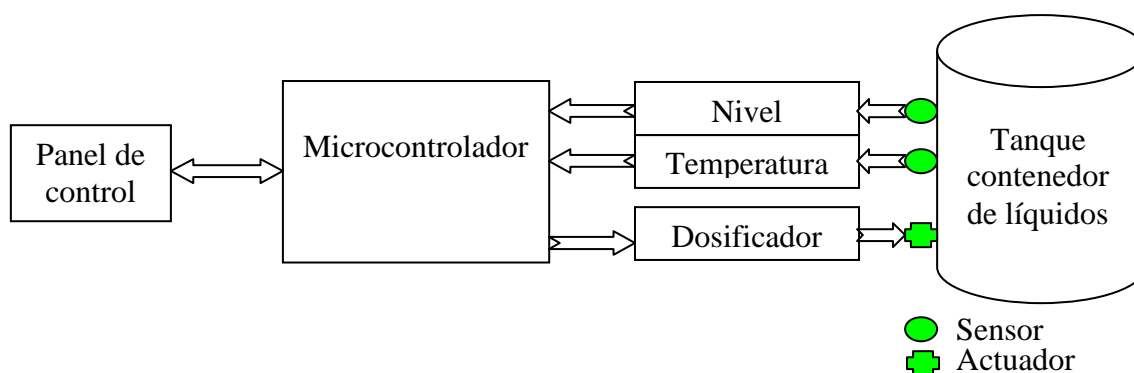
de control descodifica los programas; los buses transportan información digital a través del chip y de la computadora; la memoria local se emplea para los cálculos realizados en el mismo chip.

DISPOSITIVOS DE DETECCIÓN Y TRANSDUCTORES

La medición de magnitudes mecánicas, térmicas, eléctricas y químicas se realiza empleando dispositivos denominados sensores y transductores. El sensor es sensible a los cambios de la magnitud a medir, como una temperatura, una posición o una concentración química. El transductor convierte estas mediciones en señales eléctricas, que pueden alimentar a instrumentos de lectura, registro o control de las magnitudes medidas. Los sensores y transductores pueden funcionar en ubicaciones alejadas del observador, así como en entornos inadecuados o impracticables para los seres humanos.

Algunos dispositivos actúan de forma simultánea como sensor y transductor.

1.2 Esquema general del Prototipo



7) Ideas a Defender

La construcción de un prototipo que controle el almacenamiento de líquidos en un tanque desde un panel dará una nueva visión de desarrollo para ciertas empresas ecuatorianas ya que presenta alternativas para mejorar los procesos de extracción de líquidos, de detección de nivel y temperatura usando una tecnología que provea

de resultados óptimos y a un bajo costo.

Por medio de este dispositivo se podrá obtener una medida correcta del nivel y temperatura del líquido y se podrá extraer cantidades discretas de mismo evitando la manipulación directa y sobre todo errores de medición, todos estos parámetros al ser controlados por medio de un Panel brindará al usuario seguridad así como ahorro de tiempo y dinero.

8) Aspectos Metodológicos

- ✓ **Análisis Documental**, para poder recopilar información necesaria para el desarrollo del proyecto, de ésta manera evaluar ventajas y desventajas de las alternativas encontradas y poder escoger la opción que me permita desarrollar el proyecto con características óptimas.
- ✓ **Observación Científica**, para determinar las posibles falencias y fortalezas de sistemas utilizados para medición y extracción de líquidos en ciertas empresas ecuatorianas y tomar en cuenta esos parámetros para posibles aplicaciones en el proyecto.
- ✓ **Encuestas, entrevistas**; para conocer distintos puntos de vista para el desarrollo del proyecto, y poder determinar que dispositivos se utilizará para conseguir los datos del nivel del líquido y temperatura a la que se encuentra el mismo, los cuales serán mostrados en el panel principal junto con otros datos anexos y dependientes de los requerimientos de los técnicos. Estas encuestas y entrevistas van dirigidas a técnicos e ingenieros que conozcan del tema.
- ✓ **Apuntes teóricos y consultas prácticas**, para desarrollar la parte práctica del proyecto, que sea de fácil uso y se adapte a las necesidades de los usuarios del sistema.
- ✓ **Diseño de pruebas**, para realizar las pruebas necesarias que permitan

determinar la eficacia del sistema, en condiciones normales de uso.

9) Temario

CAPÍTULO I

1.1. RESUMEN

1.2. INTRODUCCIÓN

1.3. ANTECEDENTES Y REFERENCIAS MARCO TEÓRICO

CAPÍTULO II

2.3 SOLUCIÓN DEL PROBLEMA

2.4 DISEÑO DEL PROTOTIPO

2.5 CONSTRUCCIÓN DEL PROTOTIPO

CAPÍTULO III

3.3 PRUEBAS FINALES Y CALIBRACIÓN

3.4 RESULTADOS

Conclusiones

Bibliografía

Anexos

10) Bibliografía

HERNÁNDEZ SAMPIERI, ROBERTO. Metodología de la Investigación Científica. Colombia: Ed. Panamericanas Formas e Impresos S.A., 1997. 505 p.

<http://www.metodologia-unmsm.com/clases/8/>

http://pcazau.galeon.com/guia_red.htm

<http://cariari.ucr.ac.cr/~rodolfor/protesis.html>

Biblioteca de Consulta Microsoft ® Encarta ® 2005. © 1993-2004 Microsoft Corporation. Reservados todos los derechos.

11) Presupuesto y Fuentes de Financiamiento

El diseño construcción de este prototipo se realizará con el autofinanciamiento de los responsables de la investigación.

Actividad	Recursos	Unidad	Valor Unitario (\$)	Valor Total (\$)
Fase de Aprobación del Plan de Proyecto	Papel	7	0,01	0,07
	Cartucho	1	30,0	30,0

	Computador	10 horas	0,5	5,0
Fase de Recolección de la Información	Transporte	20 viajes	0,25	5,0
	Copias	50	0,02	1,00
	Internet	10 horas	1,0	10,0
	Varios		3,0	3,0
	Computador	10 horas	0,5	5,0
Fase de estudio justificativo	Transporte	30 viajes	0,25	7,5
	Hojas	20	0,01	0,2
	Copias	20	0,02	0,4
	Varios		5,0	5,0
	Computador	10 horas	0,5	5,0
Fase de diseño del proyecto	Hojas	100	0,01	1,0
	Computadora	50 horas	0,5	25,0
	Varios		5,0	5,0
Fase de construcción del Proyecto	Herramientas	-	-	100,0
	Elementos y material	-	-	250,0
Fase de presentación del Proyecto	Hojas	100	0,01	1,0
	Computador	10 horas	0,5	5,0
	Impresión y empastado	1	15,0	15,0
	Alquiler infocus	1	5,0	5,0
Total				486,17

ANEXO B: DATASHEET DEL PIC 16F877



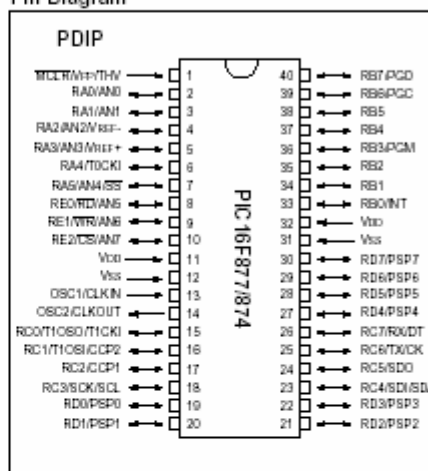
PIC16F87X

28/40-pin 8-Bit CMOS FLASH Microcontrollers

Microcontroller Core Features:

- High-performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,
Up to 368 x 8 bytes of Data Memory (RAM)
Up to 256 x 8 bytes of EEPROM data memory
- ★ Pinout compatible to the PIC16C73/74/76/77
- Interrupt capability (up to 14 internal/external interrupt sources)
- Eight level deep hardware stack
- Direct, indirect, and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code-protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low-power, high-speed CMOS FLASH/EEPROM technology
- Fully static design
- In-Circuit Serial Programming™ via two pins
- ★ Only single 5V source needed for programming
- ★ In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial and Industrial temperature ranges
- Low-power consumption:
 - < 2 mA typical @ 5V, 4 MHz
 - 20 µA typical @ 3V, 32 kHz
 - < 1 µA typical standby current

Pin Diagram



Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
- Capture is 16-bit, max. resolution is 12.5 ns, Compare is 16-bit, max. resolution is 200 ns, PWM max. resolution is 10-bit
- ★ 10-bit multi-channel Analog-to-Digital converter
- ★ Synchronous Serial Port (SSP) with SPI™ (Master Mode) and I²C™ (Master/Slave)
- ★ Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) 8-bits wide, with external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

PIC16F87X

Key Features PICmicro™ Mid-Range Reference Manual (DS33023)	PIC16F873	PIC16F874	PIC16F876	PIC16F877
Operating Frequency	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
FLASH Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory	128	128	256	256
Interrupts	13	14	13	14
I/O Ports	Ports A,B,C	Ports A,B,C,D,E	Ports A,B,C	Ports A,B,C,D,E
Timers	3	3	3	3
Capture/Compare/PWM modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Instruction Set	35 Instructions	35 Instructions	35 Instructions	35 Instructions

1.0 DEVICE OVERVIEW

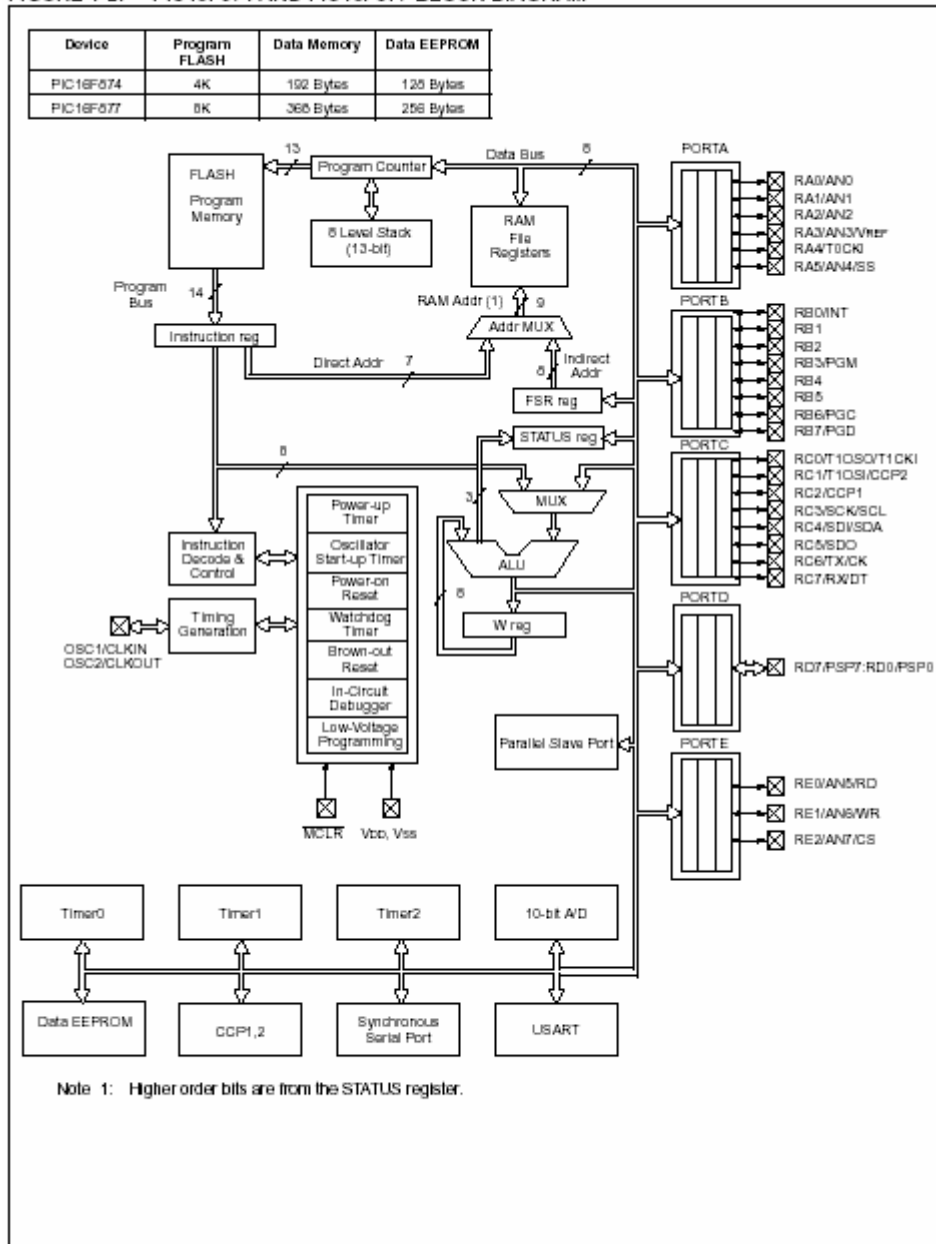
This document contains device-specific information. Additional information may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip website. The Reference Manual should be considered a complementary document to this data sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

There are four devices (PIC16F873, PIC16F874, PIC16F876, and PIC16F877) covered by this data sheet. The PIC16F876/873 devices come in 28-pin packages and the PIC16F877/874 devices come in 40-pin packages. The 28-pin devices do not have a Parallel Slave Port implemented.

The following two figures are device block diagrams sorted by pin number; 28-pin for [Figure 1-1](#) and 40-pin for [Figure 1-2](#). The 28-pin and 40-pin pinouts are listed in [Table 1-1](#) and [Table 1-2](#), respectively.

PIC16F87X

FIGURE 1-2: PIC16F874 AND PIC16F877 BLOCK DIAGRAM



PIC16F87X

TABLE 1-2 PIC16F874 AND PIC16F877 PINOUT DESCRIPTION

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS ⁽¹⁾	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
MCLR/PP/THV	1	2	18	IP	ST	Master clear (reset) input or programming voltage input or high voltage test mode control. This pin is an active low reset to the device.
RA0/AN0	2	3	19	I/O	TTL	PORTA is a bi-directional I/O port. RA0 can also be analog input0 RA1 can also be analog input1 RA2 can also be analog input2 or negative analog reference voltage RA3 can also be analog input3 or positive analog reference voltage RA4 can also be the clock input to the Timer0 timer/counter. Output is open drain type. RA5 can also be analog input4 or the slave select for the synchronous serial port.
RA1/AN1	3	4	20	I/O	TTL	
RA2/AN2/VREF-	4	5	21	I/O	TTL	
RA3/AN3/VREF+	5	6	22	I/O	TTL	
RA4/T0CK1	6	7	23	I/O	ST	
RA5/SS/AN4	7	8	24	I/O	TTL	
RB0/INT	33	36	8	I/O	TTL/ST ⁽¹⁾	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0 can also be the external interrupt pin. RB3 can also be the low voltage programming input Interrupt on change pin. Interrupt on change pin. Interrupt on change pin or In-Circuit Debugger pin. Serial programming clock. Interrupt on change pin or In-Circuit Debugger pin. Serial programming data.
RB1	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3/PGM	36	39	11	I/O	TTL	
RB4	37	41	14	I/O	TTL	
RB5	38	42	15	I/O	TTL	
RB6/PGC	39	43	16	I/O	TTL/ST ⁽²⁾	
RB7/PGD	40	44	17	I/O	TTL/ST ⁽²⁾	

Legend: I = input O = output I/O = input/output P = power
 — = Not used TTL = TTL input ST = Schmitt Trigger Input

- Note 1: This buffer is a Schmitt Trigger Input when configured as an external interrupt.
 2: This buffer is a Schmitt Trigger Input when used in serial programming mode.
 3: This buffer is a Schmitt Trigger Input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
 4: This buffer is a Schmitt Trigger Input when configured in RC oscillator mode and a CMOS input otherwise.

PIC16F87X

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
RC0/T10S0/T1CKI	15	16	32	I/O	ST	PORTC is a bi-directional I/O port. RC0 can also be the Timer1 oscillator output or a Timer1 clock input.
RC1/T10S1/CCP2	16	18	35	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	17	19	36	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	18	20	37	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I ² C modes.
RC4/SDI/SDA	23	25	42	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I ² C mode).
RC5/SDO	24	26	43	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	25	27	44	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	26	29	1	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data.
RD0/PSP0	19	21	38	I/O	ST/TTL ⁽³⁾	PORTD is a bi-directional I/O port or parallel slave port when interfacing to a microprocessor bus.
RD1/PSP1	20	22	39	I/O	ST/TTL ⁽³⁾	
RD2/PSP2	21	23	40	I/O	ST/TTL ⁽³⁾	
RD3/PSP3	22	24	41	I/O	ST/TTL ⁽³⁾	
RD4/PSP4	27	30	2	I/O	ST/TTL ⁽³⁾	
RD5/PSP5	28	31	3	I/O	ST/TTL ⁽³⁾	
RD6/PSP6	29	32	4	I/O	ST/TTL ⁽³⁾	
RD7/PSP7	30	33	5	I/O	ST/TTL ⁽³⁾	
RE0/RD/AN5	8	9	25	I/O	ST/TTL ⁽³⁾	PORTE is a bi-directional I/O port. RE0 can also be read control for the parallel slave port, or analog input5.
RE1/WR/AN6	9	10	26	I/O	ST/TTL ⁽³⁾	RE1 can also be write control for the parallel slave port, or analog input6.
RE2/CS/AN7	10	11	27	I/O	ST/TTL ⁽³⁾	RE2 can also be select control for the parallel slave port, or analog input7.
V _{SS}	12,31	13,34	6,29	P	—	Ground reference for logic and I/O pins.
V _{DD}	11,32	12,35	7,28	P	—	Positive supply for logic and I/O pins.
NC	—	1,17,28,40	12,13,33,34	—	—	These pins are not internally connected. These pins should be left unconnected.

Legend: I = input O = output IO = input/output P = power
 — = Not used TTL = TTL input ST = Schmitt Trigger Input

- Note: 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
 2: This buffer is a Schmitt Trigger input when used in serial programming mode.
 3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
 4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

PIC16F87X

2.0 MEMORY ORGANIZATION

There are three memory blocks in each of these PICmicros. The Program Memory and Data Memory have separate buses so that concurrent access can occur and is detailed in this section. The EEPROM data memory block is detailed in [Section 4.0](#).

Additional information on device memory may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

2.1 Program Memory Organization

The PIC16F87X PICmicros have a 13-bit program counter capable of addressing an 8K x 14 program memory space. The PIC16F877/876 devices have 8K x 14 words of FLASH program memory and the PIC16F873/874 devices have 4K x 14. Accessing a location above the physically implemented address will cause a wraparound.

The reset vector is at 0000h and the interrupt vector is at 0004h.

2.2 Data Memory Organization

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers. Bits RP1 and RP0 are the bank select bits.

RP1	RP0	(STATUS<6:5>)
= 00	→	Bank0
= 01	→	Bank1
= 10	→	Bank2
= 11	→	Bank3

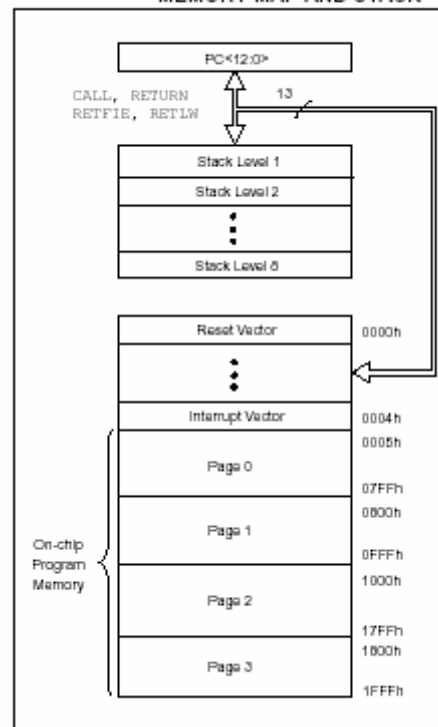
Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain special function registers. Some "high use" special function registers from one bank may be mirrored in another bank for code reduction and quicker access.

Note: EEPROM Data Memory description can be found in Section 7.0 of this Data Sheet

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly, or indirectly through the File Select Register FSR.

FIGURE 2-1: PIC16F877/876 PROGRAM MEMORY MAP AND STACK



PIC16F87X

FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP

Bank 0		Bank 1		Bank 2		Bank 3		
Indirect addr. ⁽¹⁾	00h	Indirect addr. ⁽¹⁾	80h	Indirect addr. ⁽¹⁾	100h	Indirect addr. ⁽¹⁾	180h	
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h	
PCL	02h	PCL	82h	PCL	102h	PCL	182h	
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h	
FSR	04h	FSR	84h	FSR	104h	FSR	184h	
PORTA	05h	TRISA	85h		105h		185h	
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h	
PORTC	07h	TRISC	87h		107h		187h	
PORTD ^(*)	08h	TRISD ^(*)	88h		108h		188h	
PORTE ^(*)	09h	TRISE ^(*)	89h		109h		189h	
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah	
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh	
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch	
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh	
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved ⁽²⁾	18Eh	
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reserved ⁽²⁾	18Fh	
T1CON	10h		90h		110h		190h	
TMR2	11h	SSPCON2	91h	General Purpose Register 16 Bytes	111h	General Purpose Register 16 Bytes	191h	
T2CON	12h	PR2	92h		112h		192h	
SSPBUF	13h	SSPADD	93h		113h		193h	
SSPCON	14h	SSPSTAT	94h		114h		194h	
CCPR1L	15h		95h		115h		195h	
CCPR1H	16h		96h		116h		196h	
CCP1CON	17h		97h		117h		197h	
RCSTA	18h	TXSTA	98h		118h		198h	
TXREG	19h	SPBRG	99h		119h		199h	
RCREG	1Ah		9Ah		11Ah		19Ah	
CCPR2L	1Bh		9Bh		11Bh		19Bh	
CCPR2H	1Ch		9Ch		11Ch		19Ch	
CCP2CON	1Dh		9Dh		11Dh		19Dh	
ADRESH	1Eh	ADRESL	9Eh		11Eh		19Eh	
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh	
	20h		A0h				120h	
General Purpose Register 96 Bytes	7Fh	General Purpose Register 80 Bytes	EFh	accesses 70h-7Fh	16Fh	General Purpose Register 80 Bytes	accesses 70h - 7Fh	
					170h			1F0h
					17Fh			1FFh

■ Unimplemented data memory locations, read as '0'.
 * Not a physical register.
 Note 1: These registers are not implemented on 28-pin devices.
 2: These registers are reserved, maintain these registers clear.

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and Peripheral Modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is given in Table 2-1.

The special function registers can be classified into two sets; core (CPU) and peripheral. Those registers associated with the core functions are described in detail in this section. Those related to the operation of the peripheral features are described in detail in that peripheral feature section.

PIC16F87X

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY

Address	Name	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	Value on: POR, BOR	Value on all other resets (2)		
Bank 0													
00h ⁽⁴⁾	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000		
01h	TMR0	Timer0 module's register								XXXX XXXX	UUUU UUUU		
02h ⁽⁴⁾	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000		
03h ⁽⁴⁾	STATUS	IRP	RP1	RP0	TO	PO	Z	DC	C	0001 1XXX	000q QUUU		
04h ⁽⁴⁾	FSR	Indirect data memory address pointer								XXXX XXXX	UUUU UUUU		
05h	PORTA	—	—	PORTA Data Latch when written; PORTA pins when read								--0X 0000	--0U 0000
06h	PORTB	PORTB Data Latch when written; PORTB pins when read								XXXX XXXX	UUUU UUUU		
07h	PORTC	PORTC Data Latch when written; PORTC pins when read								XXXX XXXX	UUUU UUUU		
08h ⁽⁴⁾	PORTD	PORTD Data Latch when written; PORTD pins when read								XXXX XXXX	UUUU UUUU		
09h ⁽⁴⁾	PORTE	—	—	—	—	—	RE2	RE1	RE0	---X -XXX	---U -UUU		
0Ah ^(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter							---0 0000	---0 0000
0Bh ⁽⁴⁾	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000X	0000 000U		
0Ch	PIR1	PSPIF ⁽³⁾	ADF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000		
0Dh	PIR2	—	(6)	—	EEIF	BCLIF	—	—	CCP2IF	-I-0 0--0	-I-U 0--0		
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								XXXX XXXX	UUUU UUUU		
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								XXXX XXXX	UUUU UUUU		
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1QSCEN	T1SYNC	TMR1CS	TMR1CN	--00 0000	--0U 0000		
11h	TMR2	Timer2 module's register								0000 0000	0000 0000		
12h	T2CON	—	TOU2PS3	TOU2PS2	TOU2PS1	TOU2PS0	TMR2CN	T2CKPS1	T2CKPS0	0000 0000	0000 0000		
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								XXXX XXXX	UUUU UUUU		
14h	SSPCON	WCOL	SSPCV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000		
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								XXXX XXXX	UUUU UUUU		
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								XXXX XXXX	UUUU UUUU		
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000		
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000X	0000 000U		
19h	TXREG	USART Transmit Data Register								0000 0000	0000 0000		
1Ah	RCREG	USART Receive Data Register								0000 0000	0000 0000		
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								XXXX XXXX	UUUU UUUU		
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								XXXX XXXX	UUUU UUUU		
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000		
1Eh	ADRESH	A/D Result Register High Byte								XXXX XXXX	UUUU UUUU		
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GOV DONE	—	ADON	0000 00-0	0000 00-0		

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved.
Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose

contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through MCLR and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices, always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices, always maintain these bits clear.

PIC16F87X

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY (Cont'd)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)	
Bank 1												
80h ⁽⁴⁾	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000	
81h	OPTION_REG	RBPD	INTEG	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111	
82h ⁽⁴⁾	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000	
83h ⁽⁴⁾	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu	
84h ⁽⁴⁾	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu	
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111	
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111	
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111	
88h ⁽⁴⁾	TRISD	PORTD Data Direction Register								1111 1111	1111 1111	
89h ⁽⁴⁾	TRISE	IBF	OBF	IOV	PSPMODE	—	PORTE Data Direction Bits				0000 -111	0000 -111
8Ah ^(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter						--0 0000	--0 0000
8Bh ⁽⁴⁾	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000x	
8Ch	PIE1	PSPIE ⁽³⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000	
8Dh	PIE2	—	(6)	—	EEIE	BCLIE	—	—	CCP2IE	-r-0 0--0	-r-0 0--0	
8Eh	POCON	—	—	—	—	—	—	FOR	SCR	---- --qq	---- --uu	
8Fh	—	Unimplemented								—	—	
90h	—	Unimplemented								—	—	
91h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000	
92h	PR2	Timer2 Period Register								1111 1111	1111 1111	
93h	SSPADDD	Synchronous Serial Port (I ² C mode) Address Register								0000 0000	0000 0000	
94h	SSPSTAT	SMP	CKE	D/A	P	S	R/W	UA	BF	0000 0000	0000 0000	
95h	—	Unimplemented								—	—	
96h	—	Unimplemented								—	—	
97h	—	Unimplemented								—	—	
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010	
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000	
9Ah	—	Unimplemented								—	—	
9Bh	—	Unimplemented								—	—	
9Ch	—	Unimplemented								—	—	
9Dh	—	Unimplemented								—	—	
9Eh	ADRESL	A/D Result Register Low Byte								xxxx xxxx	uuuu uuuu	
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000	

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved.
 Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose

contents are transferred to the upper byte of the program counter.

- 2: Other (non power-up) resets include external reset through MCLR and Watchdog Timer Reset.
- 3: Bits PSPIE and PSPIF are reserved on the 28-pin devices, always maintain these bits clear.
- 4: These registers can be addressed from any bank.
- 5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.
- 6: PIR2<6> and PIE2<6> are reserved on these devices, always maintain these bits clear.

PIC16F87X

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY (Cont'd)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 2											
100h ⁽⁴⁾	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
101h	TMRD	Timer0 module's register								XXXX XXXX	UUUU UUUU
102h ⁽⁴⁾	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
103h ⁽⁴⁾	STATUS	IRP	RP1	RP0	\overline{TO}	\overline{PD}	Z	DC	C	0001 1XXX	000q qUUU
104h ⁽⁴⁾	FSR	Indirect data memory address pointer								XXXX XXXX	UUUU UUUU
105h	—	Unimplemented								—	—
106h	PORTB	PORTB Data Latch when written; PORTB pins when read								XXXX XXXX	UUUU UUUU
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah ^(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
10Bh ⁽⁴⁾	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000X	0000 000X
10Ch	EEDATA	EEPROM data register								XXXX XXXX	UUUU UUUU
10Ch	EEADR	EEPROM address register								XXXX XXXX	UUUU UUUU
10Eh	EEDATH	—	—	EEPROM data register high byte						XXXX XXXX	UUUU UUUU
10Fh	EEADRH	—	—	EEPROM address register high byte						XXXX XXXX	UUUU UUUU
Bank 3											
180h ⁽⁴⁾	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
181h	OPTION_REG	RBP0	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
182h ⁽⁴⁾	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
183h ⁽⁴⁾	STATUS	IRP	RP1	RP0	\overline{TO}	\overline{PD}	Z	DC	C	0001 1XXX	000q qUUU
184h ⁽⁴⁾	FSR	Indirect data memory address pointer								XXXX XXXX	UUUU UUUU
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah ^(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
18Bh ⁽⁴⁾	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000X	0000 000X
18Ch	EEDCON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	X-- X000	X-- 0000
18Ch	EEDCON2	EEPROM control register2 (not a physical register)								----	----
18Eh	—	Reserved maintain clear								0000 0000	0000 0000
18Fh	—	Reserved maintain clear								0000 0000	0000 0000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved.
Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose

contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through MCLR and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices, always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices, always maintain these bits clear.

PIC16F87X

2.2.2.1 STATUS REGISTER

The STATUS register, shown in Figure 2-5, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

The STATUS register can be the destination for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the \overline{TO} and \overline{PD} bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper-three bits and set the Z bit. This leaves the STATUS register as `000u u1uu` (where u = unchanged).

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register because these instructions do not affect the Z, C or DC bits from the STATUS register. For other instructions, not affecting any status bits, see the "Instruction Set Summary."

Note 1: The C and DC bits operate as a borrow and digit borrow bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

FIGURE 2-5: STATUS REGISTER (ADDRESS 03h, 83h, 103h, 183h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	\overline{TO}	\overline{PD}	Z	DC	C
bit7							bit0
<p>bit 7: IRP: Register Bank Select bit (used for indirect addressing) 1 = Bank 2, 3 (100h - 1FFh) 0 = Bank 0, 1 (00h - FFh)</p> <p>bit 6-5: RP1:RP0: Register Bank Select bits (used for direct addressing) 11 = Bank 3 (180h - 1FFh) 10 = Bank 2 (100h - 17Fh) 01 = Bank 1 (80h - FFh) 00 = Bank 0 (00h - 7Fh) Each bank is 128 bytes</p> <p>bit 4: \overline{TO}: Time-out bit 1 = After power-up, <code>CLRWDT</code> instruction, or <code>SLEEP</code> instruction 0 = A WDT time-out occurred</p> <p>bit 3: \overline{PD}: Power-down bit 1 = After power-up or by the <code>CLRWDT</code> instruction 0 = By execution of the <code>SLEEP</code> instruction</p> <p>bit 2: Z: Zero bit 1 = The result of an arithmetic or logic operation is zero 0 = The result of an arithmetic or logic operation is not zero</p> <p>bit 1: DC: Digit carry/borrow bit (<code>ADDWF</code>, <code>ADDLW</code>, <code>SUBLW</code>, <code>SUBWF</code> instructions) (for borrow the polarity is reversed) 1 = A carry-out from the 4th low order bit of the result occurred 0 = No carry-out from the 4th low order bit of the result</p> <p>bit 0: C: Carry/borrow bit (<code>ADDWF</code>, <code>ADDLW</code>, <code>SUBLW</code>, <code>SUBWF</code> instructions) 1 = A carry-out from the most significant bit of the result occurred 0 = No carry-out from the most significant bit of the result occurred Note: For borrow the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (<code>RRF</code>, <code>RLF</code>) instructions, this bit is loaded with either the high or low order bit of the source register.</p>							

PIC16F87X

2.2.2.2 OPTION_REG REGISTER

The OPTION_REG register is a readable and writable register which contains various control bits to configure the TMR0 prescaler/WDT postscaler (single assignable register known also as the prescaler), the External INT Interrupt, TMR0, and the weak pull-ups on PORTB.

Note: To achieve a 1:1 prescaler assignment for the TMR0 register, assign the prescaler to the Watchdog Timer.

FIGURE 2-6: OPTION_REG REGISTER (ADDRESS 81h, 181h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBP0	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit7							bit0

R= Readable bit
 W= Writable bit
 U= Unimplemented bit, read as '0'
 - n= Value at POR reset

bit 7: **RBP0**: PORTB Pull-up Enable bit
 1 = PORTB pull-ups are disabled
 0 = PORTB pull-ups are enabled by individual port latch values

bit 6: **INTEDG**: Interrupt Edge Select bit
 1 = Interrupt on rising edge of RB0/INT pin
 0 = Interrupt on falling edge of RB0/INT pin

bit 5: **T0CS**: TMR0 Clock Source Select bit
 1 = Transition on RA4/T0CKI pin
 0 = Internal instruction cycle clock (CLKOUT)

bit 4: **T0SE**: TMR0 Source Edge Select bit
 1 = Increment on high-to-low transition on RA4/T0CKI pin
 0 = Increment on low-to-high transition on RA4/T0CKI pin

bit 3: **PSA**: Prescaler Assignment bit
 1 = Prescaler is assigned to the WDT
 0 = Prescaler is assigned to the Timer0 module

bit 2-0: **PS2:PS0**: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1:2	1:1
001	1:4	1:2
010	1:8	1:4
011	1:16	1:8
100	1:32	1:16
101	1:64	1:32
110	1:128	1:64
111	1:256	1:128

PIC16F87X

2.2.2.3 INTCON REGISTER

The INTCON Register is a readable and writable register which contains various enable and flag bits for the TMR0 register overflow, RB Port change and External RB0/INT pin interrupts.

Note: Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

FIGURE 2-7: INTCON REGISTER (ADDRESS 0Bh, 8Bh, 10Bh, 18Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit7							bit0
<p>R= Readable bit W= Writable bit U= Unimplemented bit, read as '0' - It= Value at POR reset</p>							
bit 7:	GIE: Global Interrupt Enable bit 1 = Enables all un-masked interrupts 0 = Disables all interrupts						
bit 6:	PEIE: Peripheral Interrupt Enable bit 1 = Enables all un-masked peripheral interrupts 0 = Disables all peripheral interrupts						
bit 5:	T0IE: TMR0 Overflow Interrupt Enable bit 1 = Enables the TMR0 interrupt 0 = Disables the TMR0 interrupt						
bit 4:	INTE: RB0/INT External Interrupt Enable bit 1 = Enables the RB0/INT external interrupt 0 = Disables the RB0/INT external interrupt						
bit 3:	RBIE: RB Port Change Interrupt Enable bit 1 = Enables the RB port change interrupt 0 = Disables the RB port change interrupt						
bit 2:	T0IF: TMR0 Overflow Interrupt Flag bit 1 = TMR0 register has overflowed (must be cleared in software) 0 = TMR0 register did not overflow						
bit 1:	INTF: RB0/INT External Interrupt Flag bit 1 = The RB0/INT external interrupt occurred (must be cleared in software) 0 = The RB0/INT external interrupt did not occur						
bit 0:	RBIF: RB Port Change Interrupt Flag bit 1 = At least one of the RB7:RB4 pins changed state (must be cleared in software) 0 = None of the RB7:RB4 pins have changed state						

PIC16F87X

2.2.2.4 PIE1 REGISTER

This register contains the individual enable bits for the peripheral interrupts.

Note: Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

FIGURE 2-8: PIE1 REGISTER (ADDRESS 8Ch)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
PSPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	
bit7								bit0
<p>R= Readable bit W= Writable bit U= Unimplemented bit, read as '0' - n= Value at POR reset</p>								
<p>bit 7: PSPIE⁽¹⁾: Parallel Slave Port Read/Write Interrupt Enable bit 1 = Enables the PSP read/write interrupt 0 = Disables the PSP read/write interrupt</p>								
<p>bit 6: ADIE: A/D Converter Interrupt Enable bit 1 = Enables the A/D converter interrupt 0 = Disables the A/D converter interrupt</p>								
<p>bit 5: RCIE: USART Receive Interrupt Enable bit 1 = Enables the USART receive interrupt 0 = Disables the USART receive interrupt</p>								
<p>bit 4: TXIE: USART Transmit Interrupt Enable bit 1 = Enables the USART transmit interrupt 0 = Disables the USART transmit interrupt</p>								
<p>bit 3: SSPIE: Synchronous Serial Port Interrupt Enable bit 1 = Enables the SSP interrupt 0 = Disables the SSP interrupt</p>								
<p>bit 2: CCP1IE: CCP1 Interrupt Enable bit 1 = Enables the CCP1 interrupt 0 = Disables the CCP1 interrupt</p>								
<p>bit 1: TMR2IE: TMR2 to PR2 Match Interrupt Enable bit 1 = Enables the TMR2 to PR2 match interrupt 0 = Disables the TMR2 to PR2 match interrupt</p>								
<p>bit 0: TMR1IE: TMR1 Overflow Interrupt Enable bit 1 = Enables the TMR1 overflow interrupt 0 = Disables the TMR1 overflow interrupt</p>								
<p>Note 1: PSPIE is reserved on 28-pin devices, always maintain this bit clear.</p>								

PIC16F87X

2.2.2.5 PIR1 REGISTER

This register contains the individual flag bits for the Peripheral interrupts.

Note: Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>).

FIGURE 2-9: PIR1 REGISTER (ADDRESS 0Ch)

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7				bit0			
<p>R= Readable bit W= Writable bit -n= Value at POR reset</p>							
<p>bit 7: PSPIF⁽¹⁾: Parallel Slave Port Read/Write Interrupt Flag bit 1 = A read or a write operation has taken place (must be cleared in software) 0 = No read or write has occurred</p> <p>bit 6: ADIF: A/D Converter Interrupt Flag bit 1 = An A/D conversion completed 0 = The A/D conversion is not complete</p> <p>bit 5: RCIF: USART Receive Interrupt Flag bit 1 = The USART receive buffer is full 0 = The USART receive buffer is empty</p> <p>bit 4: TXIF: USART Transmit Interrupt Flag bit 1 = The USART transmit buffer is empty 0 = The USART transmit buffer is full</p> <p>bit 7: SSPIF: Synchronous Serial Port (SSP) Interrupt Flag 1 = The SSP interrupt condition has occurred, and must be cleared in software before returning from the interrupt service routine. The conditions that will set this bit are: <u>SPI</u> A transmission/reception has taken place. <u>µC Slave</u> A transmission/reception has taken place. <u>µC Master</u> A transmission/reception has taken place. The initiated start condition was completed by the SSP module. The initiated stop condition was completed by the SSP module. The initiated restart condition was completed by the SSP module. The initiated acknowledge condition was completed by the SSP module. A start condition occurred while the SSP module was idle (Multimaster system). A stop condition occurred while the SSP module was idle (Multimaster system). 0 = No SSP interrupt condition has occurred.</p> <p>bit 2: CCP1IF: CCP1 Interrupt Flag bit <u>Capture Mode</u> 1 = A TMR1 register capture occurred (must be cleared in software) 0 = No TMR1 register capture occurred <u>Compare Mode</u> 1 = A TMR1 register compare match occurred (must be cleared in software) 0 = No TMR1 register compare match occurred <u>PWM Mode</u> Unused in this mode</p> <p>bit 1: TMR2IF: TMR2 to PR2 Match Interrupt Flag bit 1 = TMR2 to PR2 match occurred (must be cleared in software) 0 = No TMR2 to PR2 match occurred</p> <p>bit 0: TMR1IF: TMR1 Overflow Interrupt Flag bit 1 = TMR1 register overflowed (must be cleared in software) 0 = TMR1 register did not overflow</p> <p>Note 1: PSPIF is reserved on 28-pin devices, always maintain this bit clear.</p>							

PIC16F87X

2.2.2.6 PIE2 REGISTER

This register contains the individual enable bits for the CCP2 peripheral interrupt, the SSP bus collision interrupt, and the EEPROM write operation interrupt.

FIGURE 2-10: PIE2 REGISTER (ADDRESS 8Dh)

U-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	
—	(1)	—	EEIE	BCLIE	—	—	CCP2IE	
bit7								bit0

R= Readable bit
W= Writable bit
U= Unimplemented bit, read as '0'
-n= Value at POR reset

bit 7: Unimplemented: Read as '0'

bit 6: Reserved: Always maintain this bit clear

bit 5: Unimplemented: Read as '0'

bit 4: EEIE: EEPROM Write Operation Interrupt Enable
1 = Enable EE Write Interrupt
0 = Disable EE Write Interrupt

bit 3: BCLIE: Bus Collision Interrupt Enable
1 = Enable Bus Collision Interrupt
0 = Disable Bus Collision Interrupt

bit 2-1: Unimplemented: Read as '0'

bit 0: CCP2IE: CCP2 Interrupt Enable bit
1 = Enables the CCP2 interrupt
0 = Disables the CCP2 interrupt

Note 1: PIE2<6> is reserved, always maintain this bit clear.

PIC16F87X

2.2.2.7 PIR2 REGISTER

This register contains the flag bits for the CCP2 interrupt, the SSP bus collision interrupt and the EEPROM write operation interrupt.

Note: Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

FIGURE 2-11: PIR2 REGISTER (ADDRESS 0Dh)

U-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	(1)	—	EEIF	BCLIF	—	—	CCP2IF
bit7						bit0	

R= Readable bit
W= Writable bit
U= Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7: Unimplemented: Read as '0'

bit 6: Reserved: Always maintain this bit clear

bit 5: Unimplemented: Read as '0'

bit 4: EEIF: EEPROM Write Operation Interrupt Flag bit
1 = The write operation completed (must be cleared in software)
0 = The write operation is not complete or has not been started

bit 3: BCLIF: Bus Collision Interrupt Flag
1 = A bus collision has occurred in the SSP, when configured for I²C master mode
0 = No bus collision has occurred

bit 2-1: Unimplemented: Read as '0'

bit 0: CCP2IF: CCP2 Interrupt Flag bit
Capture Mode
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare Mode
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM Mode
Unused

Note 1: PIR2<6> is reserved, always maintain this bit clear.

2.2.2.8 PCON REGISTER

The Power Control (PCON) register contains a flag bit to allow differentiation between a Power-on Reset (POR) to an external MCLR Reset or WDT Reset. Those devices with brown-out detection circuitry contain an additional bit to differentiate a Brown-out Reset condition from a Power-on Reset condition.

Note: BOR is unknown on Power-on Reset if the BOR circuit is disabled by clearing the BODEN bit in Configuration Word. The BOR status bit is a don't care and is not necessarily predictable if the brown-out circuit is disabled.

FIGURE 2-12: PCON REGISTER (ADDRESS 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-1
—	—	—	—	—	—	POR	BOR
bit7						bit0	

R= Readable bit
W= Writable bit
U= Unimplemented bit, read as '0'
- n= Value at POR reset

bit 7-2: Unimplemented: Read as '0'

bit 1: **POR**: Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0: **BOR**: Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

PIC16F87X

2.3 PCL and PCLATH

The program counter (PC) specifies the address of the instruction to fetch for execution. The PC is 13 bits wide. The low byte is called the PCL register. This register is readable and writable. The high byte is called the PCH register. This register contains the PC<12:8> bits and is not directly readable or writable. All updates to the PCH register go through the PCLATH register.

2.3.1 STACK

The stack allows a combination of up to 8 program calls and interrupts to occur. The stack contains the return address from this branch in program execution.

Midrange devices have an 8 level deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not modified when the stack is PUSHed or POPed.

After the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

2.4 Program Memory Paging

PIC16F87X devices are capable of addressing a continuous 8K word block of program memory. The CALL and GOTO instructions provide only 11 bits of address to allow branching within any 2K program memory page. When doing a CALL or GOTO instruction the upper 2 bits of the address are provided by PCLATH<4:3>. When doing a CALL or GOTO instruction, the user must ensure that the page select bits are programmed so that the desired program memory page is addressed. If a return from a CALL instruction (or interrupt) is executed, the entire 13-bit PC is pushed onto the stack. Therefore, manipulation of the PCLATH<4:3> bits are not required for the return instructions (which POPs the address from the stack).

The INDF register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR register (FSR is a pointer). This is indirect addressing.

EXAMPLE 2-1: INDIRECT ADDRESSING

- Register file 05 contains the value 10h
- Register file 06 contains the value 0Ah
- Load the value 05 into the FSR register
- A read of the INDF register will return the value of 10h
- Increment the value of the FSR register by one (FSR = 06)
- A read of the INDF register now will return the value of 0Ah.

Reading INDF itself indirectly (FSR = 0) will produce 00h. Writing to the INDF register indirectly results in a no-operation (although STATUS bits may be affected).

ANEXO C: MANUAL DE USUARIO DE SOFTWARE “NIPLE”



INTRODUCCIÓN A NIPLE

Niple es un “Entorno Visual de Desarrollo para programación de microcontroladores PIC”.

Con Niple podrá programar los PIC a través del diseño de un diagrama de flujo de manera gráfica y de alto nivel de programación, sin necesidad de escribir código Assembler, o escribiendo la menor cantidad de código, convirtiéndose en la forma más rápida, sencilla y productiva de desarrollar proyectos con microcontroladores PIC.

Diseñe un diagrama de flujo de manera gráfica y en “lenguaje humano” mientras Niple realiza el trabajo pesado por Ud., controlando errores lógicos o de configuración, realizando cálculos, coordinando la interacción entre las distintas partes del proyecto y generando de manera automática el código Assembler.

Estas características, hacen de Niple la herramienta ideal tanto para quienes se inician en el mundo de los microcontroladores como para aquellos usuarios más experimentados que quieran aumentar su productividad.

Entre las principales prestaciones que Niple ofrece podemos destacar:

- Entono Visual de Alto Nivel de programación:
- Niple ofrece una metodología de programación totalmente visual y en lenguaje humano.
- Bloques estandarizados, lo cual facilita la interpretación del proyecto.
- Ud. se concentra en “Que Hacer” y Niple se encarga de “Como Hacerlo”.
- Control automático de errores lógicos.

- . Supervisa el ingreso de datos.
- . Control automático de registros y bits:
- . Controla la existencia de todos los registros y bits necesarios para el correcto funcionamiento
- del proyecto en el momento de generar el código Assembler.
- . Controla que no existan nombres de registros y bits duplicados.
- . Valida los nombres de los registros y los bits en el momento de ser declarados.
- . Corrección automática de errores lógicos
- . Controla la correcta configuración de los puertos.
- . Controla la configuración y activación de las interrupciones.
- . Supervisa la creación de vínculos entre bloques.
- . Generación del Código Assembler:
- . Control automático de la sintaxis del código generado.
- . Control automático del cambio de página.
- . Generación automática de etiquetas.
- . Modelos de PIC soportados por Niple:
- Niple para PIC 12F6xx
 - . 12F629
 - . 12F675
 - . Niple para PIC 16F84
 - . 16F84
 - . Niple para PIC 16F6xx
 - . 16F627
 - . 16F628
 - . 16F648
 - . Niple para PIC 16F87x
 - . 16F870
 - . 16F873
 - . 16F874
 - . 16F876
 - . 16F877

- Velocidades de reloj soportadas: 4 Mhz, 10 Mhz, 20 Mhz.

FUNCIONES DISPONIBLES

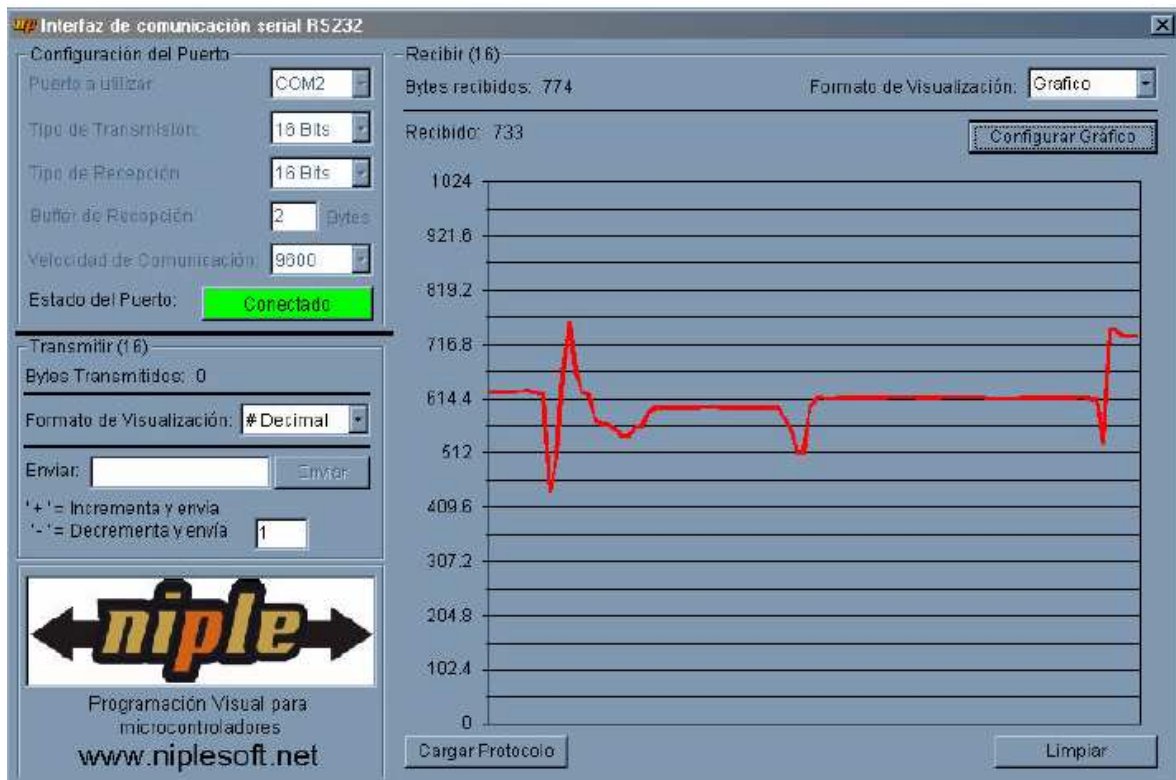
Las funciones disponibles dependen del modelo de microcontrolador utilizado pero en general podemos mencionar:

- Selección del cristal utilizado (4, 10 o 20 Mhz)
- Asignación de valores a registros y bits
- Configuración y manejo de interrupciones
- Manejo de rutinas de usuario
- Configuración, Lectura y Escritura de puertos
- Lectura / Escritura de EEPROM interna del PIC
- Funciones de comparación de registros y bits (condiciones =, >, <, >=, <=, <>)
- Manejo automático de tablas
- Temporizadores por bucle (de tiempo fijo o variable)
- Visualización de datos en dígitos 7 segmentos (Ánodo común y Cátodo común)
- Ingreso de datos por teclado matricial (3x3 y 3x4)
- Visualización de datos en pantallas LCD (16x2, 20x2 y 20x4)
- Manejo de memorias EEPROM 24Cxxx por comunicación I2C
- Manejo de memorias RAM PCF8570 por comunicación I2C.
- Manejo de reloj/calendario de tiempo real PCF8583, con control de alarmas por fecha.
- Configuración y manejo de temporizadores internos (TMR0, TMR1 y TMR2)
- Comunicaciones RS232 por código y por USART
- Comunicaciones RS485
- Comparadores analógicos
- Conversión A/D. La cantidad de canales depende del modelo de micro
- Módulos CCP (Compara, Captura y PWM)
- Cálculos matemáticos: (+, -, /, *, en 8 o 16 bits), conversión a BCD

- Escalado de valores de un rango a otro (a 8 y 16 bits)
- Manejo de registros de desplazamiento

HERRAMIENTAS ADICIONALES

□ Interfase de comunicación RS232. Comunicaciones a 8 o 16 bits con visualización en distintos formatos (decimal, hexadecimal, binario, texto, o modo gráfico).



METODOLOGÍA DE PROGRAMACIÓN CON NIPLE

La metodología de programación en Niple consiste en el diseño de un diagrama de flujo de manera totalmente visual y de alto nivel de programación.

Esto significa que Ud. no escribe código en Assembler adaptándose al “lenguaje” del micro, sino que, por el contrario, Ud. realiza el diseño del programa de manera gráfica, seleccionando opciones a través de pantallas claras e intuitivas, y donde el sistema se adapta al “lenguaje humano” y luego “traduce” a código Assembler .

El diagrama de flujo está formado por un conjunto de bloques “estandarizados”. Tanto la forma, como el color y el texto de cada bloque, le están presentando información y datos concretos acerca de cada parte del proceso, lo cual facilita aún más la interpretación del programa.

Además, el diagrama de flujo le permite tener una “visión general” del proyecto. El proceso de diseño del diagrama de flujo consiste en insertar bloques y vincularlos por medio de flechas.

Durante éste proceso, Niple controla y advierte de posibles errores, tanto de la lógica del programa como en la configuración del sistema (configuración de puertos, interrupciones, etc.).

En el margen derecho de la pantalla se encuentra el “Árbol de Herramientas”. Este consiste en una estructura de árbol donde se encuentran todas las funciones que pueden ser insertadas como bloques en el diagrama de flujo y que generan un código Assembler.

Los bloques se encuentran agrupados en distintos niveles organizados por categoría para facilitar la búsqueda de un bloque en particular.

En el primer nivel se organizan por tipo de función donde se encuentran los siguientes ítems:

Instrucciones: Esta categoría incluye todas instrucciones en Assembler.

A su vez se organizan por tipo de instrucción:

- . Instrucciones orientadas a Registros
- . Instrucciones orientadas a Bits
- . Instrucciones orientadas a Literales
- . Instrucciones orientadas a Control
- . Todas las instrucciones ordenadas alfabéticamente

El objetivo de Niple es utilizar lo menos posible los bloques de instrucciones ya que éstos bloques equivalen a programar en código Assembler.

Rutinas Básicas: Las rutinas básicas son módulos prediseñados y se encargan de resolver un proceso en particular (una configuración, una comparación, un cálculo, etc.). Ud. puede formar estructuras lógicas muy complejas de manera rápida y sencilla a partir de la combinación de rutinas básicas.

Utilice las rutinas básicas para programar dentro de cualquier “Entorno”.

Las rutinas básicas están organizadas por tipo de rutina:

- . Asignación de valores
- . Manejo de puertos
- . Manejo de EEPORM
- . Funciones Lógicas
- . Funciones matemáticas
- . Comunicaciones
- . Etc.

Interrupciones: En ésta categoría se encuentran todas las funciones de configuración y manejo de las interrupciones. Niple presenta las interrupciones disponibles de acuerdo al modelo de PIC que se está utilizando.

Rutinas de usuario: Ud. puede diseñar sus propias funciones o procedimientos para solucionar un problema en particular (siempre combinando distintas rutinas básicas) y luego llamarlos desde distintas partes del programa (subrutinas). Esto le permite por un lado, la creación de funciones o procedimientos personalizados y por otro lado, la optimización del código Assembler generado, ya que las rutinas de usuario sólo se escriben una vez y luego se insertan “llamadas” a las mismas.

Esta categoría incluye las funciones de declaración e inserción de llamadas a rutinas de usuario.

ANEXO D: SENSORES DE ULTRASONIDO (136653CK)

Ultrasonic Sensor Sets

- Use for motion or distance sensing
- Frequency: 40kHz \pm 1.0kHz
- Aluminum case

136653CK:

Transmit/Receive Set

- Capacitance: 2000pF \pm 30%
- Transmitter: bandwidth 4.0kHz @ 112dB, sound pressure level 119dB @ 40 \pm 1.0kHz
- Lead length/spacing: 0.39"
- Receiver: bandwidth 3.5kHz @ -71dB, min. sensitivity 65dB @ 40 \pm 1.0kHz (R=3.9k Ω)
- Case size: 0.47"H x 0.62" Dia.



136653CK

136653CK 40TR16F⊕ \$6.95

ANEXO E: LM 35



July 1999

LM35 Precision Centigrade Temperature Sensors

General Description

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in $^{\circ}$ Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of $\pm 1/4^{\circ}\text{C}$ at room temperature and $\pm 3/4^{\circ}\text{C}$ over a full -55 to $+150^{\circ}\text{C}$ temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only $60\ \mu\text{A}$ from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a -55° to $+150^{\circ}\text{C}$ temperature range, while the LM35C is rated for a -40° to $+110^{\circ}\text{C}$ range (-10° with improved accuracy). The LM35 series is available packaged in

hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-220 package.

Features

- Calibrated directly in $^{\circ}$ Celsius (Centigrade)
- Linear $+10.0\ \text{mV}/^{\circ}\text{C}$ scale factor
- 0.5°C accuracy guaranteeable (at $+25^{\circ}\text{C}$)
- Rated for full -55° to $+150^{\circ}\text{C}$ range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than $60\ \mu\text{A}$ current drain
- Low self-heating, 0.08°C in still air
- Nonlinearity only $\pm 1/4^{\circ}\text{C}$ typical
- Low impedance output, $0.1\ \Omega$ for 1 mA load

Typical Applications

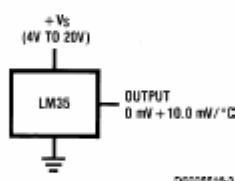
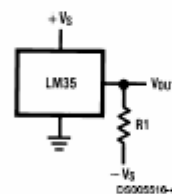


FIGURE 1. Basic Centigrade Temperature Sensor
($+2^{\circ}\text{C}$ to $+150^{\circ}\text{C}$)



Choose $R_1 = -V_{S2}/50\ \mu\text{A}$
 $V_{out} = +1,500\ \text{mV}$ at $+150^{\circ}\text{C}$
 $= +250\ \text{mV}$ at $+25^{\circ}\text{C}$
 $= -550\ \text{mV}$ at -55°C

FIGURE 2. Full-Range Centigrade Temperature Sensor

TRI-STATE[®] is a registered trademark of National Semiconductor Corporation.

Connection Diagrams

TO-46
Metal Can Package*



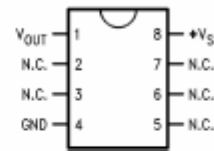
BOTTOM VIEW
DS000516-1

*Case is connected to negative pin (GND)

Order Number LM35H, LM35AH, LM35CH, LM35CAH or LM35DH

See NS Package Number H03H

SO-8
Small Outline Molded Package



DS000516-21

N.C. = No Connection

Top View
Order Number LM35DM
See NS Package Number M08A

TO-92
Plastic Package

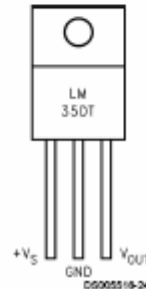


BOTTOM VIEW
DS000516-2

Order Number LM35CZ,
LM35CAZ or LM35DZ

See NS Package Number Z03A

TO-220
Plastic Package*



DS000516-24

*Tab is connected to the negative pin (GND).

Note: The LM35DT pinout is different than the discontinued LM35DP.

Order Number LM35DT
See NS Package Number TA03F

Absolute Maximum Ratings (Note 10)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage	+35V to -0.2V
Output Voltage	+6V to -1.0V
Output Current	10 mA
Storage Temp.:	
TO-46 Package,	-60°C to +180°C
TO-92 Package,	-60°C to +150°C
SO-8 Package,	-65°C to +150°C
TO-220 Package,	-65°C to +150°C
Lead Temp.:	
TO-46 Package,	
(Soldering, 10 seconds)	300°C

TO-92 and TO-220 Package, (Soldering, 10 seconds)	260°C
SO Package (Note 12)	
Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C
ESD Susceptibility (Note 11)	2500V
Specified Operating Temperature Range: T_{MIN} to T_{MAX} (Note 2)	
LM35, LM35A	-55°C to +150°C
LM35C, LM35CA	-40°C to +110°C
LM35D	0°C to +100°C

Electrical Characteristics

(Notes 1, 6)

Parameter	Conditions	LM35A			LM35CA			Units (Max.)
		Typical	Tested Limit (Note 4)	Design Limit (Note 5)	Typical	Tested Limit (Note 4)	Design Limit (Note 5)	
Accuracy (Note 7)	$T_A = +25^\circ\text{C}$	± 0.2	± 0.5		± 0.2	± 0.5	± 1.0	$^\circ\text{C}$
	$T_A = -10^\circ\text{C}$	± 0.3			± 0.3			$^\circ\text{C}$
	$T_A = T_{MAX}$	± 0.4	± 1.0		± 0.4	± 1.0		$^\circ\text{C}$
	$T_A = T_{MIN}$	± 0.4	± 1.0		± 0.4		± 1.5	$^\circ\text{C}$
Nonlinearity (Note 8)	$T_{MIN} \leq T_A \leq T_{MAX}$	± 0.18		± 0.35	± 0.15		± 0.3	$^\circ\text{C}$
Sensor Gain (Average Slope)	$T_{MIN} \leq T_A \leq T_{MAX}$	+10.0	+9.9, +10.1		+10.0		+9.9, +10.1	mV/ $^\circ\text{C}$
Load Regulation (Note 3) $0 \leq I_L \leq 1$ mA	$T_A = +25^\circ\text{C}$	± 0.4	± 1.0		± 0.4	± 1.0		mV/mA
	$T_{MIN} \leq T_A \leq T_{MAX}$	± 0.5		± 3.0	± 0.5		± 3.0	mV/mA
Line Regulation (Note 3)	$T_A = +25^\circ\text{C}$	± 0.01	± 0.05		± 0.01	± 0.05		mV/V
	$4V \leq V_S \leq 30V$	± 0.02		± 0.1	± 0.02		± 0.1	mV/V
Quiescent Current (Note 9)	$V_S = +5V, +25^\circ\text{C}$	56	67		56	67		μA
	$V_S = +5V$	105		131	91		114	μA
	$V_S = +30V, +25^\circ\text{C}$	56.2	68		56.2	68		μA
	$V_S = +30V$	105.5		133	91.5		116	μA
Change of Quiescent Current (Note 3)	$4V \leq V_S \leq 30V, +25^\circ\text{C}$	0.2	1.0		0.2	1.0		μA
	$4V \leq V_S \leq 30V$	0.5		2.0	0.5		2.0	μA
Temperature Coefficient of Quiescent Current		+0.39		+0.5	+0.39		+0.5	$\mu\text{A}/^\circ\text{C}$
Minimum Temperature for Rated Accuracy	In circuit of Figure 1, $I_L = 0$	+1.5		+2.0	+1.5		+2.0	$^\circ\text{C}$
Long Term Stability	$T_J = T_{MAX}$, for 1000 hours	± 0.08			± 0.08			$^\circ\text{C}$

Electrical Characteristics								
(Notes 1, 6)								
Parameter	Conditions	LM35			LM35C, LM35D			Units (Max.)
		Typical	Tested Limit (Note 4)	Design Limit (Note 5)	Typical	Tested Limit (Note 4)	Design Limit (Note 5)	
Accuracy, LM35, LM35C (Note 7)	$T_A = +25^\circ\text{C}$	± 0.4	± 1.0		± 0.4	± 1.0	± 1.5	$^\circ\text{C}$
	$T_A = -10^\circ\text{C}$	± 0.5			± 0.5		± 1.5	$^\circ\text{C}$
	$T_A = T_{\text{MAX}}$	± 0.8	± 1.5		± 0.8		± 1.5	$^\circ\text{C}$
	$T_A = T_{\text{MIN}}$	± 0.8		± 1.5	± 0.8		± 2.0	$^\circ\text{C}$
Accuracy, LM35D (Note 7)	$T_A = +25^\circ\text{C}$				± 0.8	± 1.5		$^\circ\text{C}$
	$T_A = T_{\text{MAX}}$				± 0.9		± 2.0	$^\circ\text{C}$
	$T_A = T_{\text{MIN}}$				± 0.9		± 2.0	$^\circ\text{C}$
Nonlinearity (Note 8)	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	± 0.3		± 0.5	± 0.2	± 0.5	$^\circ\text{C}$	
Sensor Gain (Average Slope)	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	$+10.0$	$+9.8$, $+10.2$		$+10.0$	$+9.8$, $+10.2$	$\text{mV}/^\circ\text{C}$	
Load Regulation (Note 3) $0 \leq I_L \leq 1 \text{ mA}$	$T_A = +25^\circ\text{C}$	± 0.4	± 2.0		± 0.4	± 2.0	mV/mA	
	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	± 0.5		± 5.0	± 0.5		± 5.0	mV/mA
Line Regulation (Note 3)	$T_A = +25^\circ\text{C}$	± 0.01	± 0.1		± 0.01	± 0.1	mV/V	
	$4\text{V} \leq V_B \leq 30\text{V}$	± 0.02		± 0.2	± 0.02		± 0.2	mV/V
Quiescent Current (Note 9)	$V_B = +5\text{V}$, $+25^\circ\text{C}$	56	80		56	80	μA	
	$V_B = +5\text{V}$	105		158	91		138	μA
	$V_B = +30\text{V}$, $+25^\circ\text{C}$	56.2	82		56.2	82	μA	
	$V_B = +30\text{V}$	105.5		161	91.5		141	μA
Change of Quiescent Current (Note 3)	$4\text{V} \leq V_B \leq 30\text{V}$, $+25^\circ\text{C}$	0.2	2.0		0.2	2.0	μA	
	$4\text{V} \leq V_B \leq 30\text{V}$	0.5		3.0	0.5		3.0	μA
Temperature Coefficient of Quiescent Current		$+0.39$		$+0.7$	$+0.39$		$\mu\text{A}/^\circ\text{C}$	
Minimum Temperature for Rated Accuracy	In circuit of Figure 1, $I_L = 0$	$+1.5$		$+2.0$	$+1.5$		$^\circ\text{C}$	
Long Term Stability	$T_J = T_{\text{MAX}}$, for 1000 hours	± 0.08			± 0.08		$^\circ\text{C}$	

Note 1: Unless otherwise noted, these specifications apply: $-55^\circ\text{C} \leq T_J \leq 150^\circ\text{C}$ for the LM35 and LM35A; $-40^\circ\text{C} \leq T_J \leq 110^\circ\text{C}$ for the LM35C and LM35CA; and $0^\circ\text{C} \leq T_J \leq 100^\circ\text{C}$ for the LM35D. $V_B = +5\text{Vdc}$ and $I_{\text{LOAD}} = 50 \mu\text{A}$, in the circuit of Figure 2. These specifications also apply from $+2^\circ\text{C}$ to T_{MAX} in the circuit of Figure 1. Specifications in boldface apply over the full rated temperature range.

Note 2: Thermal resistance of the TO-46 package is $400^\circ\text{C}/\text{W}$, junction to ambient, and $24^\circ\text{C}/\text{W}$ junction to case. Thermal resistance of the TO-92 package is $180^\circ\text{C}/\text{W}$ junction to ambient. Thermal resistance of the small outline molded package is $220^\circ\text{C}/\text{W}$ junction to ambient. Thermal resistance of the TO-220 package is $90^\circ\text{C}/\text{W}$ junction to ambient. For additional thermal resistance information see table in the Applications section.

Note 3: Regulation is measured at constant junction temperature, using pulse testing with a low duty cycle. Changes in output due to heating effects can be computed by multiplying the internal dissipation by the thermal resistance.

Note 4: Tested Limits are guaranteed and 100% tested in production.

Note 5: Design Limits are guaranteed (but not 100% production tested) over the indicated temperature and supply voltage ranges. These limits are not used to calculate outgoing quality levels.

Note 6: Specifications in boldface apply over the full rated temperature range.

Note 7: Accuracy is defined as the error between the output voltage and $10\text{mV}/^\circ\text{C}$ times the device's case temperature, at specified conditions of voltage, current, and temperature (expressed in $^\circ\text{C}$).

Note 8: Nonlinearity is defined as the deviation of the output-voltage-versus-temperature curve from the best-fit straight line, over the device's rated temperature range.

Note 9: Quiescent current is defined in the circuit of Figure 1.

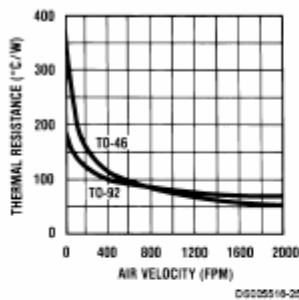
Note 10: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its rated operating conditions. See Note 1.

Note 11: Human body model, 100 pF discharged through a $1.5 \text{ k}\Omega$ resistor.

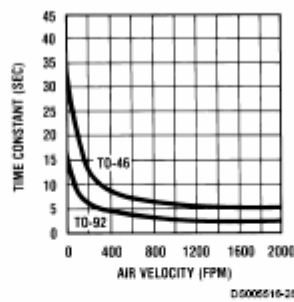
Note 12: See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" or the section titled "Surface Mount" found in a current National Semiconductor Linear Data Book for other methods of soldering surface mount devices.

Typical Performance Characteristics

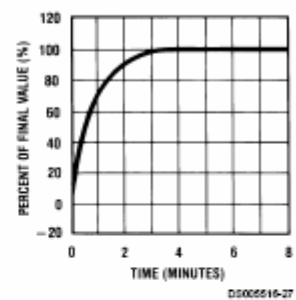
**Thermal Resistance
Junction to Air**



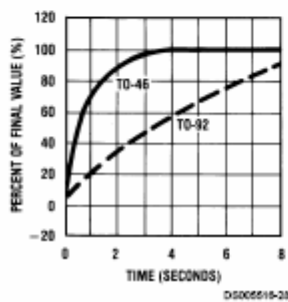
Thermal Time Constant



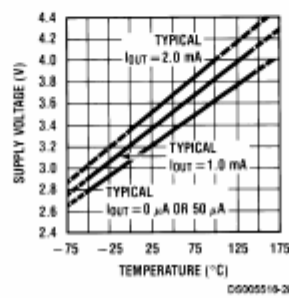
**Thermal Response
in Still Air**



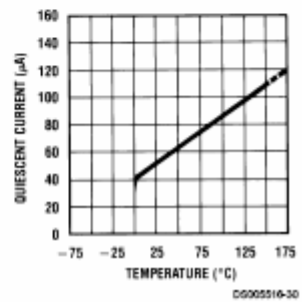
**Thermal Response in
Stirred Oil Bath**



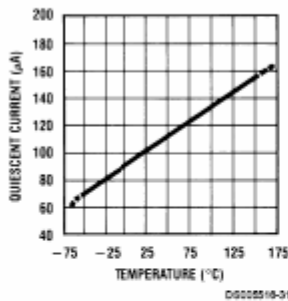
**Minimum Supply
Voltage vs. Temperature**



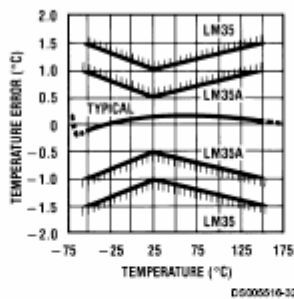
**Quiescent Current
vs. Temperature
(In Circuit of Figure 1.)**



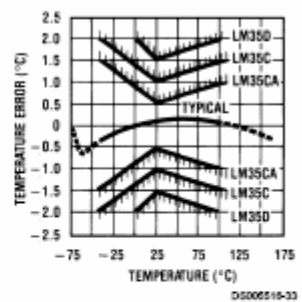
**Quiescent Current
vs. Temperature
(In Circuit of Figure 2.)**



**Accuracy vs. Temperature
(Guaranteed)**

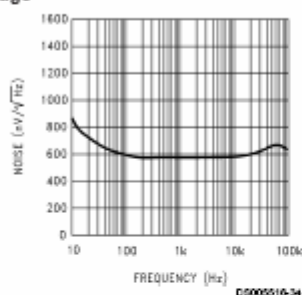


**Accuracy vs. Temperature
(Guaranteed)**

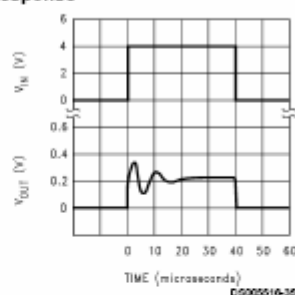


Typical Performance Characteristics (Continued)

Noise Voltage



Start-Up Response



Applications

The LM35 can be applied easily in the same way as other integrated-circuit temperature sensors. It can be glued or cemented to a surface and its temperature will be within about 0.01°C of the surface temperature.

This presumes that the ambient air temperature is almost the same as the surface temperature; if the air temperature were much higher or lower than the surface temperature, the actual temperature of the LM35 die would be at an intermediate temperature between the surface temperature and the air temperature. This is especially true for the TO-92 plastic package, where the copper leads are the principal thermal path to carry heat into the device, so its temperature might be closer to the air temperature than to the surface temperature.

To minimize this problem, be sure that the wiring to the LM35, as it leaves the device, is held at the same temperature as the surface of interest. The easiest way to do this is to cover up these wires with a bead of epoxy which will insure that the leads and wires are all at the same temperature as the surface, and that the LM35 die's temperature will not be affected by the air temperature.

The TO-46 metal package can also be soldered to a metal surface or pipe without damage. Of course, in that case the V- terminal of the circuit will be grounded to that metal. Alternatively, the LM35 can be mounted inside a sealed-end metal tube, and can then be dipped into a bath or screwed into a threaded hole in a tank. As with any IC, the LM35 and accompanying wiring and circuits must be kept insulated and dry, to avoid leakage and corrosion. This is especially true if the circuit may operate at cold temperatures where condensation can occur. Printed-circuit coatings and varnishes such as Humiseal and epoxy paints or dips are often used to insure that moisture cannot corrode the LM35 or its connections.

These devices are sometimes soldered to a small light-weight heat fin, to decrease the thermal time constant and speed up the response in slowly-moving air. On the other hand, a small thermal mass may be added to the sensor, to give the steadiest reading despite small deviations in the air temperature.

Temperature Rise of LM35 Due To Self-heating (Thermal Resistance, θ_{JA})

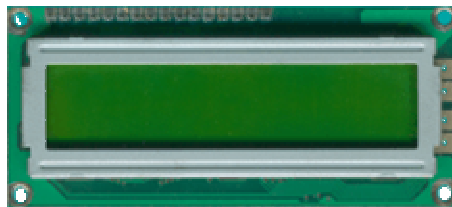
	TO-46, no heat sink	TO-46*, small heat fin	TO-92, no heat sink	TO-92**, small heat fin	80-8 no heat sink	80-8**, small heat fin	TO-220 no heat sink
Still air	400°C/W	100°C/W	180°C/W	140°C/W	220°C/W	110°C/W	90°C/W
Moving air	100°C/W	40°C/W	90°C/W	70°C/W	105°C/W	50°C/W	25°C/W
Still oil	100°C/W	40°C/W	90°C/W	70°C/W			
Stirred oil	50°C/W	30°C/W	45°C/W	40°C/W			
(Clamped to metal, infinite heat sink)		(24°C/W)				(55°C/W)	

*Wakefield type 201, or 1" disc of 0.020" sheet brass, soldered to case, or similar.

**TO-92 and 80-8 packages glued and leads soldered to 1" square of 1/16" printed circuit board with 2 oz. foil or similar.

ANEXO F: DISTRIBUCIÓN DE PINES DE LCD (2x16)

Pin1 Pin 14



PIN Nº	SIMBOLO	DESCRIPCION
1	Vss	Tierra de alimentación GND
2	Vdd	Alimentación de +5V CC
3	Vo	Contraste del cristal liquido. (0 a +5V)
4	RS	Selección del registro de control/registro de datos: RS=0 Selección registro de control RS=1 Selección registro de datos
5	R/W	Señal de lectura/escritura: R/W=0 Escritura (Write) R/W=1 Lectura (Read)
6	E	Habilitación del modulo: E=0 Módulo desconectado E=1 Módulo conectado
7-14	D0-D7	Bus de datos bidireccional.

ANEXO G: REGULADORES DE VOLTAJE



www.fairchildsemi.com

MC78XX/LM78XX/MC78XXA

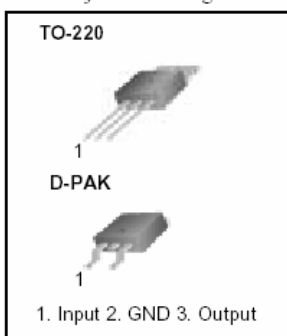
3-Terminal 1A Positive Voltage Regulator

Features

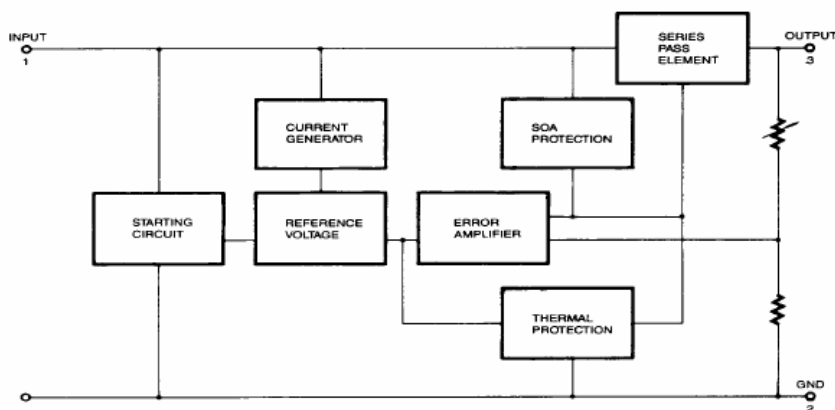
- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

Description

The MC78XX/LM78XX/MC78XXA series of three terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



Internal Block Diagram



Rev. 1.0.1

Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5V$ to $18V$) (for $V_O = 24V$)	V_I V_I	35 40	V V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	$^{\circ}C/W$
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	$^{\circ}C/W$
Operating Temperature Range	T_{OPR}	0 ~ +125	$^{\circ}C$
Storage Temperature Range	T_{STG}	-65 ~ +150	$^{\circ}C$

Electrical Characteristics (MC7805/LM7805)

(Refer to test circuit, $0^{\circ}C < T_J < 125^{\circ}C$, $I_O = 500mA$, $V_I = 10V$, $C_I = 0.33\mu F$, $C_O = 0.1\mu F$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7805/LM7805			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}C$	4.8	5.0	5.2	V	
		$5.0mA \leq I_O \leq 1.0A$, $P_O \leq 15W$ $V_I = 7V$ to $20V$	4.75	5.0	5.25		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}C$	$V_O = 7V$ to $25V$	-	4.0	100	mV
			$V_I = 8V$ to $12V$	-	1.6	50	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}C$	$I_O = 5.0mA$ to $1.5A$	-	9	100	mV
			$I_O = 250mA$ to $750mA$	-	4	50	
Quiescent Current	I_Q	$T_J = +25^{\circ}C$	-	5.0	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5mA$ to $1.0A$	-	0.03	0.5	mA	
		$V_I = 7V$ to $25V$	-	0.3	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5mA$	-	-0.8	-	mV/ $^{\circ}C$	
Output Noise Voltage	V_N	$f = 10Hz$ to $100KHz$, $T_A = +25^{\circ}C$	-	42	-	$\mu V/V_O$	
Ripple Rejection	RR	$f = 120Hz$ $V_O = 8V$ to $18V$	62	73	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1A$, $T_J = +25^{\circ}C$	-	2	-	V	
Output Resistance	r_O	$f = 1KHz$	-	15	-	m Ω	
Short Circuit Current	I_{SC}	$V_I = 35V$, $T_A = +25^{\circ}C$	-	230	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}C$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7809)

(Refer to test circuit, $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 15\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7809			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	8.65	9	9.35	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 11.5\text{V to } 24\text{V}$	8.6	9	9.4		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 11.5\text{V to } 25\text{V}$	-	6	180	mV
			$V_I = 12\text{V to } 17\text{V}$	-	2	90	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	12	180	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	4	90	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.0	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA to } 1.0\text{A}$		-	-	0.5	mA
		$V_I = 11.5\text{V to } 26\text{V}$		-	-	1.3	
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$		-	-1	-	mV/ $^{\circ}\text{C}$
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$, $T_A = +25^{\circ}\text{C}$		-	58	-	$\mu\text{V}/V_O$
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 13\text{V to } 23\text{V}$		56	71	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$		-	2	-	V
Output Resistance	r_O	$f = 1\text{KHz}$		-	17	-	$\text{m}\Omega$
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$		-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$		-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7812)

(Refer to test circuit, $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 19\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7812			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	11.5	12	12.5	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 14.5\text{V to } 27\text{V}$	11.4	12	12.6		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 14.5\text{V to } 30\text{V}$	-	10	240	mV
			$V_I = 16\text{V to } 22\text{V}$	-	3.0	120	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	11	240	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	5.0	120	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.1	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA to } 1.0\text{A}$	-	0.1	0.5	mA	
		$V_I = 14.5\text{V to } 30\text{V}$	-	0.5	1.0		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-1	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{kHz}$, $T_A = +25^{\circ}\text{C}$	-	76	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 15\text{V to } 25\text{V}$	55	71	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{kHz}$	-	18	-	$\text{m}\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	230	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

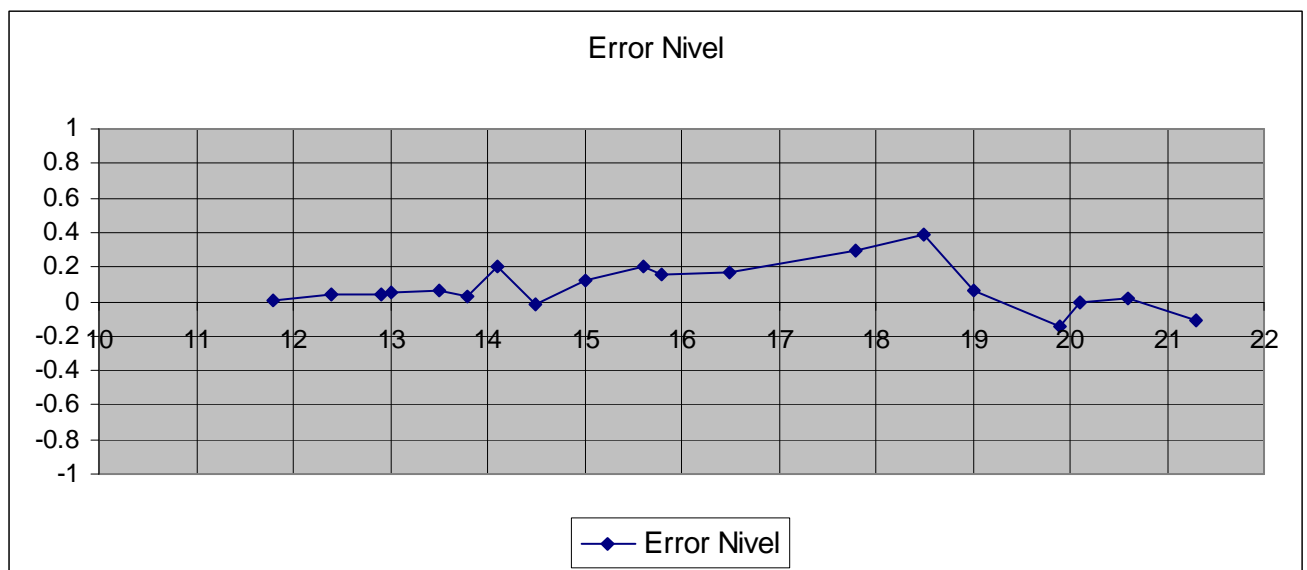
Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

ANEXO H: ERROR DE NIVEL DEL LÍQUIDO

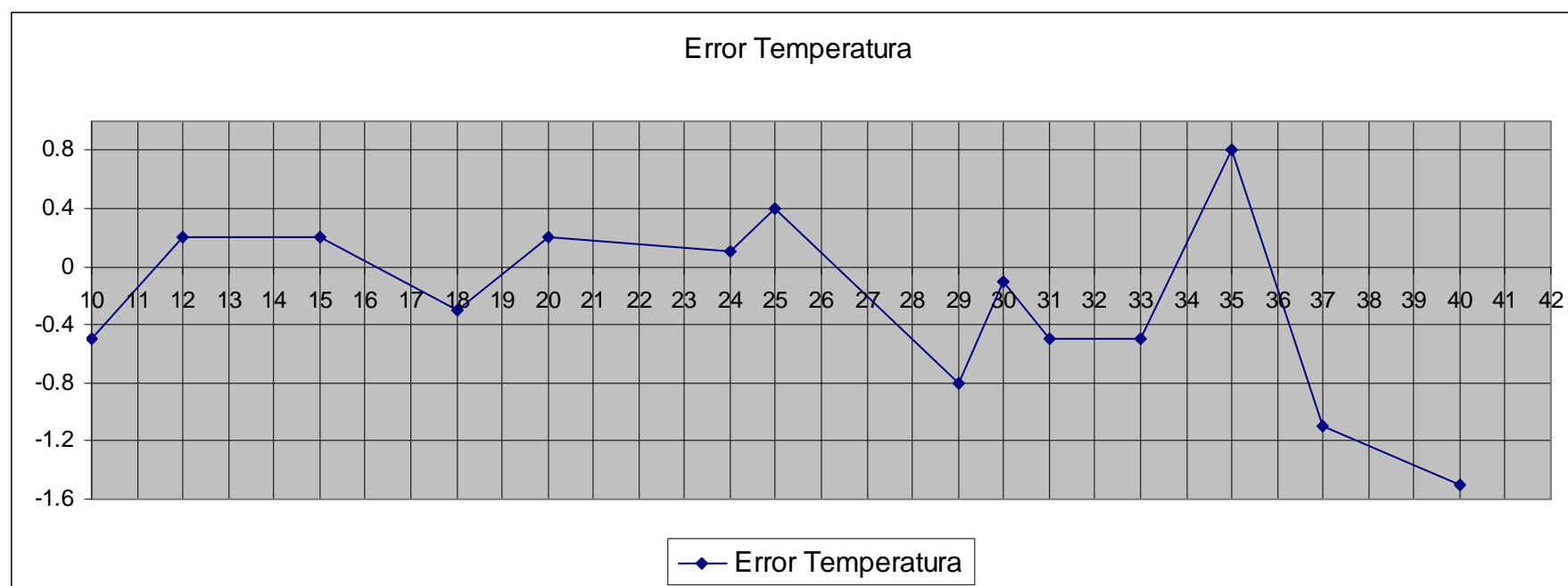
PRUEBAS DEL DISPOSITIVO

NIVEL (cm)		
REAL	MEDIDO	ERROR
11.80	11.79	0.01
12.40	12.36	0.04
12.90	12.86	0.04
13.00	12.95	0.05
13.50	13.44	0.06
13.80	13.77	0.03
14.10	13.90	0.20
14.50	14.52	-0.02
15.00	14.88	0.12
15.60	15.40	0.20
15.80	15.64	0.16
16.50	16.33	0.17
17.80	17.51	0.29
18.50	18.11	0.39
19.00	18.94	0.06
19.90	20.04	-0.14
20.10	20.11	-0.01
20.60	20.58	0.02
21.30	21.41	-0.11



ANEXO I: ERROR DE TEMPERATURA DEL LÍQUIDO

PRUEBAS DEL DISPOSITIVO		
TEMPERATURA (°C)		
REAL	MEDIDA	ERROR
10.00	10.50	-0.50
12.00	11.80	0.20
15.00	14.80	0.20
18.00	18.30	-0.30
20.00	19.80	0.20
24.00	23.90	0.10
25.00	24.60	0.40
29.00	29.80	-0.80
30.00	30.10	-0.10
31.00	31.50	-0.50
33.00	33.50	-0.50
35.00	34.20	0.80
37.00	38.10	-1.10
40.00	41.50	-1.50



ANEXO J: COSTOS REALES

Actividad	Recursos	Unidad	Valor Unitario (\$)	Valor Total (\$)
Fase de Aprobación del Plan de Proyecto	Papel	7	0,01	0,07
	Impresiones	7	0,10	0,70
	Computador	4 horas	0,5	2,00
Fase de Recolección de la Información	Transporte	20 viajes	0,25	5,00
	Copias	50	0,10	5,00
	Internet	10 horas	0,5	10,0
	Varios	---	---	10,0
	Computador	10 horas	0,5	5,00
Fase de estudio justificativo	Transporte	30 viajes	0,25	7,50
	Hojas	100	0,01	1,00
	Copias	20	0,10	2,00
	Varios		---	5,00
	Computador	10 horas	0,5	5,00
Fase de diseño del proyecto	Hojas	100	0,01	1,00
	Computadora	100 horas	0,5	50,00
	Varios	---	---	20,00
Fase de construcción del Proyecto	Herramientas	-	-	150,0
	Elementos y material	-	-	300,0
Fase de presentación del Proyecto	Hojas	100	0,01	1,00
	Computador	40 horas	0,5	20,0
	Impresión y empastado	1	---	30,0
	Alquiler infocus	1	5,0	5,00
Total				635,27