

**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**



FACULTAD DE INGENIERÍA

**“DISEÑO Y DESARROLLO DE UN
ELECTROPORADOR DÍGITAL”**

T E S I S

**PARA OBTENER EL TÍTULO DE:
ING. ELÉCTRICO ELECTRÓNICO**

**P R E S E N T A :
JORGE OCTAVIO GUTIÉRREZ RUIZ**

DIRECTOR: M.I. RAÚL RUVALCABA MORALES



México D.F

2010



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

"Hay una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad"

Albert Einstein

Es ahora, ya en los primeros años de este tercer milenio, cuando aquellos a quienes en algún momento se les llamo novatos – mis compañeros – son por fin ingenieros, terminan sus estudios, o al menos los de licenciatura, pues son muchos los que recorren el camino de la Ingeniería, pero son muy pocos los que la culminan, a ellos agradezco la muestra de personalidades tan variadas que me permitieron deslumbrar el tipo de persona que deseo tener a mi lado, a los teóricos, a los prácticos, a los irascibles, vengativos, a los que carecen de palabra y a los que están llenos de artificios; ó a los que me enseñaron el significado de lealtad, amistad y responsabilidad; en todos ellos he encontrado el significado del "hecho" y del "derecho" y por eso les doy las gracias.

Igualmente quiero agradecer sinceramente a aquellas personas que compartieron sus conocimientos conmigo aquellos que son quienes conforman el pilar del conocimiento, la sabiduría de la naturaleza y la problemática de la sociedad – Ingeniería – y son a ellos los que se han conformado como artífices de lo que se denominaría "*mi escuela*", en el sentido de que cada uno de sus miembros ha enseñado a miles de generaciones antes de mí el conocimiento de su profesión.

Gracias:

- M.C Edgar Baldemar
- Ing. Erick Castañeda
- Ing. Francisco Rodríguez
- Ing. Jaime Martínez
- Lic. Juan Pablo García
- M.I Raúl Ruvalcaba
- M.I Lauro Santiago

A mí tía Norma Marisela donde quieras que estés te digo que fue muy difícil vivir y amanecer recordando que te has ido pero siempre te voy a recordar por tu cariño, amabilidad y alegría y nunca me olvidare de los buenos ratos y recuerdos que pasé a tu lado en esa ciudad tan linda que es GUADALAJARA. A ti te quedo particularmente agradecido por las enseñanzas de vida que compartiste conmigo, por mostrarme lo que soy y lo que puedo llegar a ser.

A mis verdaderos amigos donde los conocí por brindarme su apoyo y mantener firme la voluntad de finalizar esta etapa de mi vida, por tener siempre un eco alentador en los malos y difíciles momentos, son a ellos a los que agradezco y les digo con la firme certeza de que los problemas son tan sólo la oportunidad para conocer nuestra fuerza interior.

Gracias:

- Javier de Jesús
- Gerardo Leon
- Rebeca Teutli
- Laura Martinez
- Mariana Camiña
- Jesica Fabila
- Denisse Delgado
- Eduardo Lugo

A mi familia les doy las gracias a todos aquellos que tuvieron un detalle de atención y preocupación por la integridad de mis estudios, por sus oraciones y su invaluable apoyo.

Y por ultimo quiero expresar mis deseos que este trabajo de Tesis es ante todo la evolución de mi experiencia dentro del camino de la Ingeniería, y que venga a ser como colocar un eslabón dentro de la cadena que conformara la columna vertebral de mi carácter como Ingeniero.

La inteligencia consiste no sólo en el conocimiento, sino también en la destreza de aplicar los conocimientos en la práctica.

Isaac Newton.

Índice

<u>Antecedentes</u>	3
---------------------------	---

<u>¿QUÉ HACER ?</u>	4
<u>SOLUCIÓN PROPUESTA.</u>	5

<u>Capítulo 1</u> <i>Introducción</i>	7
---	---

1.1 <u>OBSERVACIONES GENERALES DE SEGURIDAD</u>	8
1.2 <u>ESTRUCTURA GENERAL DEL SISTEMA.</u>	8

<u>Capítulo 2</u> <i>Unidad de Control</i>	10
--	----

2.1 <u>MICROCONTROLADOR</u>	10
2.2 <u>FAMILIA DE MICROCONTROLADORES</u>	11
2.3 <u>CARACTERÍSTICAS DEL MICROCONTROLADOR.</u>	12
2.3.1 <u>ARQUITECTURA INTERNA.</u>	13
2.3.2 <u>ARQUITECTURA HARVARD.</u>	13
2.4 <u>CARACTERÍSTICAS DEL MICROCONTROLADOR PIC18F452</u>	15
2.4.1. <u>MEMORIA EEPROM</u>	16
2.4.2. <u>ORGANIZACIÓN DE LA MEMORIA DE PROGRAMA (ROM</u> <u>FLASH)</u> 16	
2.4.3. <u>ORGANIZACIÓN DE LA MEMORIA RAM (RANDOM ACCESS</u> <u>MEMORY)</u>	17
2.4.4. <u>INTERRUPCIONES</u>	19
2.4.5. <u>CONVERTIDOR ANALÓGICO DIGITAL</u>	20

	2.4.6. TIMERS.....	20
2.5	<u>HERRAMIENTAS DE DESARROLLO DE SOFTWARE</u>	22
	2.5.1. EDITORES DE TEXTO.....	22
	2.5.2. ENSAMBLADORES Y COMPILADORES.....	22
	2.5.3. SIMULADORES.....	23
	2.5.4. ENTORNOS INTEGRADOS DE DESARROLLO (IDE).....	23
	2.5.5. COMPONENTES ADICIONALES PARA MPLAB IDE.....	24
2.6	<u>CICLO DE DESARROLLO</u>	25
2.7	<u>BUS I²C</u>	26
2.8	<u>MEMORIA EEPROM DE DATOS</u>	27
	2.8.1. TRANSFERENCIA DE LA INFORMACIÓN.....	28
2.9	<u>RELOJ-CALENDARIO DS1307</u>	30
	2.9.1. CONEXIÓN DEL DS1307 AL MICROCONTROLADOR.....	30
	2.9.2. REGISTROS DEL DS1307.....	31
	2.9.3. ESCRITURA EN EL DS1307.....	32
	2.9.4. LECTURA DEL DS1307.....	32

Capítulo 3 *Unidad de Potencia*.....33

3.1	<u>RECTIFICACIÓN</u>	32
	3.1.1. RECTIFICACIÓN DE MEDIA ONDA.....	33
	3.4.1. RECTIFICACIÓN DE ONDA COMPLETA.....	34
3.2	<u>FILTRADO</u>	34
3.3	<u>PORCENTAJE DE REGULACIÓN</u>	35
3.4	<u>REGULACIÓN</u>	35
	3.4.1. REGULADORES DE VOLTAJE LINEALES.....	36
	3.4.2. CARACTERÍSTICAS COMUNES.....	36
3.5	<u>DRIVER DE LA CORRIENTE DE SALIDA</u>	37



<i>2N2907</i>	96
<i>PN2222</i>	98
<i>MOC3011</i>	101
<i>IRFD110</i>	103
<i>ZENER 1N4733A</i>	106
<i>LCD</i>	107
<i>PIC18F452</i>	116
<i>MEMORIA 24C256</i>	123
<i>DS1307</i>	125



Antecedentes

A pesar de la capacidad regenerativa de la mayoría de los tejidos del cuerpo, existen algunas lesiones o enfermedades como el mal de Parkinson, la diabetes mellitus, las alteraciones cardíacas y las enfermedades osteodegenerativas y nerviosas, que provocan un daño irreversible en las células afectadas, haciendo que el correspondiente órgano o tejido pierda su funcionalidad^[1]. Las enfermedades osteodegenerativas como la osteoporosis, la artritis y las lesiones de los discos articulares generan grandes pérdidas de fragmentos óseos, y tienen un efecto devastador en el bienestar y la calidad de vida de los pacientes.

Los procedimientos clínicos disponibles, tienen un uso limitado debido a la movilidad del sitio donante, en el caso de los autoinjertos; o por el rechazo inmune o la baja capacidad regenerativa, en el caso de los aloinjertos o xenoinjertos^[2]. Todas estas limitaciones han llevado a la búsqueda de nuevas formas de restituir los procesos biológicos que han resultado afectados, bien mediante el aporte de precursores celulares sanos, en procesos conocidos como terapias celulares, o la aplicación de factores de crecimiento producidos normalmente por tales células^[3]. Investigadores del Departamento de Medicina Genómica y Toxicología Ambiental del Instituto de Investigaciones Biomédicas de la UNAM, encabezan una línea de investigación con modelos animales que permiten la manipulación experimental, cuyo objetivo es comprender los mecanismos moleculares de la morfogénesis de la mano y la formación de los dedos.

La morfogénesis (del griego morphê, forma, y génesis, creación) es uno de los tres aspectos fundamentales de la biología del desarrollo, junto con el control del crecimiento celular y de la diferenciación celular. La morfogénesis incluye la forma de los tejidos, de los órganos y de los organismos completos y las posiciones de varios tipos de células especializadas. El estudio de la morfogénesis, pretende comprender el proceso que controla la distribución organizada especialmente de las células, proceso que aparece a lo largo del desarrollo embrionario de un organismo y que da lugar a las formas características de los tejidos biológicos, de los órganos y de la anatomía corporal.

En este sentido, actualmente se estudian las malformaciones de las extremidades humanas como son: los dedos pegados (sindactilias); más de cinco dedos (polidactilias), y los enanismos (acndroplasia y acondrogénesis), entre otras. Entender, cómo se estructura los mecanismos que llevan a la configuración de los dedos y las moléculas involucradas en este proceso, será posible que en un futuro próximo se pueda regenerar un dedo humano o modificarse alguna patología en la mano del hombre. Lo que sería relevante, por ejemplo, en las enfermedades osteodegenerativas humanas.

¹ Mínguez A., Escamilla F. Terapia celular y otras estrategias neuroregenerativas en la enfermedad de Parkinson (I). *Revistaneurológica*, 41, 604-614, 2005.

² Estrada C., Paz A.C., López L.E. Ingeniería de tejido óseo: consideraciones básicas. *Revista EIA*, 5, 93-100, 2006.

³ Prósper F., Gaviria J.J., Herreros J., Rábago G., Luquin R., Moreno J., Robles J.E., Redondo P. Trasplante celular y terapia regenerativa con células. *Anales del sistema sanitario de Navarra*, 29, 219-234, 2006.

Sin embargo, el actual incremento de sofisticados instrumentos biológicos de diagnóstico, tratamiento y asistencia desarrollados en otros países, dejan a México rezagado en la búsqueda del desarrollo biológico.

Es por eso, que el objetivo del presente trabajo de Tesis es establecer una solución tecnológica propia y llevarla a los laboratorios de Investigación Biológica de la UNAM.

¿Qué hacer?

Varias son las propuestas (metodologías y desarrollos tecnológicos) para sanar la carencia de infraestructura en sistemas biológicos, entre las cuales, se pueden mencionar las siguientes:

- ✚ Desarrollar hardware para caracterización biológica basado en un μ C (por sus siglas en inglés, Micro Controller) o algún otro dispositivo programable, y CI (Circuitos Integrados) comerciales de propósito general.
- ✚ Usar laboratorios remotos a través de Internet.
- ✚ Acceder a facilidades industriales o a la de otras instituciones de educación.

La última propuesta se puede realizar, eventualmente, sin un convenio establecido entre las partes, en el que la buena voluntad permite, acceder a facilidades de simulación, de pruebas y de caracterización del sistema. Sin embargo, el acceso a facilidades industriales es viable sólo cuando hay una colaboración o proyecto conjunto en marcha. Si es el caso, en cuanto el proyecto concluye, el acceso a dichas facilidades también.

Por otro lado, los laboratorios remotos, por otro lado, merecen una atención especial. Las arquitecturas cliente-servidor ofrecen instrumentos de propósitos generales. Éstos se controlan vía Internet y están conectados directamente a una PC o a través de una interfase vía un bus de control. Toda vez que las aplicaciones son definidas en línea (on-line) se inicia un periodo de espera, es decir, el sistema espera por la solicitud o conexión del cliente, para así poder poner en operación los instrumentos solicitados. En esta arquitectura, el servidor actúa como una clase de instrumentos que el cliente accede. Una variación de la arquitectura servidor-cliente es la denominada DCOM (por sus siglas en inglés Distributed Component Object Model) en la cual es posible acceder a componentes remotos y obtener resultados adicionales. El software comercialmente disponible para controlar remotamente sistemas de medición, solicita aplicaciones específicas para ser instaladas en el equipo de cómputo del cliente^[4]. El software no es de distribución libre, y su costo es elevado para la economía de las instituciones de educación pública nacional.

Finalmente, el desarrollo de hardware para sistemas biológicos parece ser la opción más adecuada, dos son los dispositivos programables generalmente utilizados: μ C y FPGA (por sus siglas en inglés, Field Programmable Gate Array). Por las capacidades propias de un μ C, su uso se orienta principalmente a aplicaciones de control/supervisión de procesos y monitorización de parámetros de interés. La ventaja de usar dispositivos programables contra la del uso de aplicaciones no digitales es que se cuenta con memoria (RAM y Flash) para almacenamiento, conexión serial a PC y posibilidades de incorporar otros dispositivos para aumentar las capacidades de la plataforma bajo desarrollo.

⁴ De Capitán S., Ferrero A. y Asaron M., Mobile Agent Technology for Remote Measurements, en IEEE Transaction on Instrumentation and Measurements, vol. 55, núm. 5, pp. 1559-1564, octubre 2006.

Solución propuesta.

De lo anterior es viable desarrollar una plataforma de caracterización basada en un μC , incorporar los CI comerciales para incrementar las capacidades del μC seleccionado.

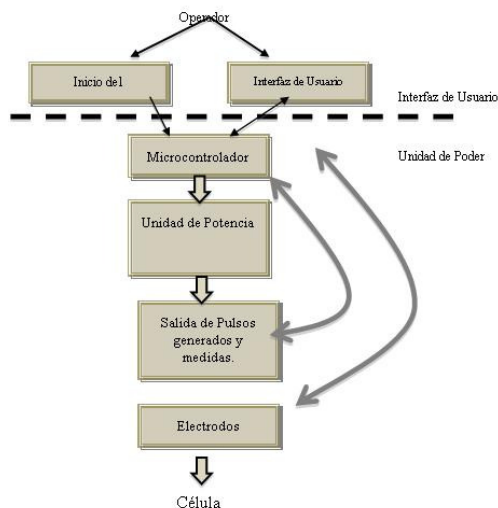


Figura A “Diagrama a bloques de la plataforma de caracterización.”

Considerando ventajas y/o desventajas de los dispositivos programables existentes en el mercado, la solución propuesta se muestra en la Figura A en la que el procesador es un dispositivo μC , Adecuación refiere CIs comerciales que acondicionan las señales que se envían y/o se reciben de esta etapa de adecuación. Esta característica es necesaria para aquellas situaciones en las que las variables de interés son de magnitud tan pequeña que, para conservar las capacidades del μC , se requiere aplicar una etapa de amplificación, por el lado contrario, se debe reducir la amplitud si ésta es de gran valor. Luego, para el bloque Despliegue, se propone un LCD (por sus siglas en inglés, Liquid Cristal Display) que permita al usuario dar seguimiento a la caracterización seleccionada, mientras que un teclado permite la interacción con el sistema, es decir, permite programar un evento y/o visualizar resultados, entre otras capacidades.

Pero ¿qué es lo que realmente hace el sistema y por que es necesaria la aplicación en un hardware para caracterización biológica?

El objetivo de este sistema de caracterización biológica es generar lo que se conoce como electroporación o electropermeabilización. La electroporación tiene sus inicios en los años ochenta del siglo pasado y consiste en la emisión de una onda electromagnética (con predominio del campo eléctrico) pulsada, atérmica y no ionizable con la finalidad de producir una alteración en el potencial de la membrana. Ésta alteración provoca una electrorotación de los lípidos con la consiguiente apertura momentánea de la membrana celular (poros) para el ingreso de sustancias hacia su interior. Este mecanismo tiene la particularidad de ser reversible y transitorio debido a la característica de la corriente; pulsada y no constante.

Cuando el voltaje que atraviesa una membrana plasmática excede su rigidez dieléctrica se forman poros. Si la fuerza del campo eléctrico aplicado y/o la duración de la exposición al mismo se eligen apropiadamente, los poros formados por el pulso eléctrico se sellan tras un corto periodo de tiempo, durante el cual los compuestos extracelulares tienen la oportunidad de entrar a la célula. Sin embargo, una exposición excesiva de células vivas, a campos eléctricos puede causar apoptosis y/o necrosis, procesos que provocan la muerte celular.

La electroporación es usada habitualmente en biología molecular para la transformación de bacterias, levaduras y protoplastos vegetales. También corrige genes defectuosos responsables del desarrollo de enfermedades, las proteínas que codifican los genes son capaces de realizar nuevamente las funciones dañadas por los desórdenes (trastornos genéticos).

Las membranas lipídicas, las bacterias también tienen una pared celular compuestas de peptidoglicano y sus derivados. Sin embargo, las paredes son porosas por naturaleza y sólo actúan como corazas que protegen a la célula de impactos ambientales severos.

La electroporación se lleva a cabo en un electroporador, sistema de caracterización biológica que crea la corriente eléctrica y la hace pasar a través de la suspensión celular (típicamente bacterias, aunque otros tipos de células pueden ser usadas, como se ha comentado anteriormente). La suspensión se micro-inyecta en una cubeta de plástico o vidrio con electrodos de aluminio en los costados.

El éxito de la electroporación depende en gran medida de la pureza de la solución con el plásmido, especialmente de su contenido en sal. Las soluciones impuras pueden causar una pequeña explosión (un arco eléctrico), en cuyo caso las células morirían. Si esto ocurre a menudo, una precipitación de las células podría ser necesaria antes de una nueva electroporación.

Las ventajas de utilizar esta tecnología son inmediatas para el biólogo, pues dispone de un mejor manejo de inserción de datos, así como un mejor despliegue de los mismos.

Capítulo 1

Introducción

Para realizar la electroporación es necesario utilizar un sistema diseñado específicamente para este uso. El diseño de un sistema de electroporación consiste en la entrega de pulsos de voltaje, la seguridad de la membrana celular y la seguridad del usuario son igualmente importantes. Pulsos con una amplitud máxima de 80 V de amplitud y 0.7 A de corriente máxima de acuerdo a la necesidad, que se genere dentro de la electroporación de la membrana celular.

La electroporación al tejido celular, se lleva a cabo por la generación de pulsos de voltaje, el campo eléctrico es por medio de electrodos insertados en la célula o tejido que son tratados. Dado que la amplitud y el gradiente del campo eléctrico dependerá del voltaje aplicado, así como de la distancia entre los electrodos, de igual forma la entrega de corriente depende de las características óhmicas de los tejidos y células en tratamiento y la distancia entre los electrodos.

Además, el tejido puede tener propiedades eléctricas (impedancias, capacitancias) alteradas durante el tratamiento, como consecuencia de las profundas modificaciones causadas a las células y a su ambiente exterior cuando se aplican altas corrientes; cabe mencionar de igual forma que los altos valores de la intensidad de campo eléctrico pueden también ser alcanzados por la organización de los electrodos más cerca los unos de los otros.

Los voltajes y corrientes utilizados representan los principales problemas que deben tenerse en cuenta a la hora de diseñar un sistema de electroporación. Se plantean cuestiones de seguridad debido a la alta energía y por las condiciones de trabajo presentes en el laboratorio.

Una peculiaridad del proceso de electroporación es que la resistencia de carga de un tejido biológico es muy variable y, por lo tanto, desconocido en el momento del tratamiento. Por otra parte, la resistencia depende de las propiedades físicas de los electrodos utilizados y hay una disminución de la resistencia durante el tratamiento debido a los cambios inducidos por el campo eléctrico en el tejido celular.

Por consiguiente, en el diseño del electroporador se tiene que mantener bajo control la corriente de fuga hacia la célula, un diseño confiable y fácil manipulación para la generación de pulsos de voltaje, respuesta en caso de cortocircuito, y proporcionar una interfase de uso fácil para los usuarios (esto para reducir al mínimo la probabilidad de error del usuario).

1.1 Observaciones generales de seguridad

La liberación de energía sobre la célula debe ser controlada y limitada; estrictamente relacionados con esta cuestión es la confiabilidad del sistema, la ausencia de medidas de protección adecuadas y la falla de algún componente crítico, puede llevar a una descarga de energía no deseada.

La limitación de la entrega de energía se refiere a condiciones de fallo. Se trata de una elección estratégica del control de riesgos en particular para los componentes críticos cuya falla puede dar lugar a una incontrolada entrega de energía hacia la célula, o mediante la aplicación de un sistema que impida la entrega de energía por encima de los valores nominales de trabajo, ambas soluciones tienen pros y contras. En cualquier caso, el sistema debe ser lo suficientemente sencillo para limitar esta falla, a tan solo un caso remoto.

La confiabilidad se destina en particular contra los cortocircuitos y las chispas que puedan producirse entre los electrodos, debido a la imprevisibilidad de la carga resistiva de los tejidos biológicos, la presencia de la conductividad de las soluciones y el error humano.

Como consecuencia de ello, es casi imposible hacer una predicción exacta de la carga sin una prueba preliminar y es más seguro considerar el escenario menos favorable. Por esta razón, el sistema diseñado considera el caso de falla con chispas y corto circuito de corriente y presenta una limitación para evitar lesiones a la célula y la falla del sistema. Las consideraciones pueden estimar que en el peor de los casos de corto-circuito, la carga no debe demandar más de 0.7 A, por lo que se supone que si durante el tratamiento establecido la corriente supera este límite, es probable que exista un cortocircuito.

1.2 Estructura General del Sistema.

El sistema de electroporación está conformado de la siguiente manera:

- a) Unidad de control (UC) calcula los parámetros de tratamiento sobre la base de datos introducida por el usuario, muestra y elabora los datos y señales de medida durante el tratamiento.
- b) Unidad de energía (UE) genera los pulsos, utilizando los parámetros proporcionados por la unidad de control.
- c) Interfase humana proporciona una navegación intuitiva al usuario, que le permite al usuario dar seguimiento al tratamiento biológico y desplegar los datos del mismo.
- d) Fuente de Alimentación proporciona la energía necesaria que la unidad de control y la interfase humana necesitan.

Una vez que los parámetros son establecidos, la UE realiza el tratamiento en tiempo real del sistema. La configuración permite un buen control sobre la energía entregada por la fuente, el ancho del pulso, la frecuencia de los pulsos y una reacción inmediata a los sistemas de seguridad.

El diagrama de bloques del sistema de electroporación se muestra en la figura 5.1. La UE se compone de un bloque de poder. El bloque de control incluye principalmente el microprocesador, un reloj de tiempo real y una memoria EEPROM que en un futuro le

permitirá al usuario crear una base de datos de todos los tratamientos que realice y una interfase para comunicarse con el usuario y se encarga de las operaciones de los demás bloques.

El voltaje de la fuente es establecida por el usuario. Una vez que los parámetros del tratamiento son establecidos el microprocesador espera una señal para proceder con el tratamiento. La señal es proporcionada directamente por el usuario por medio de una tecla de inicio.

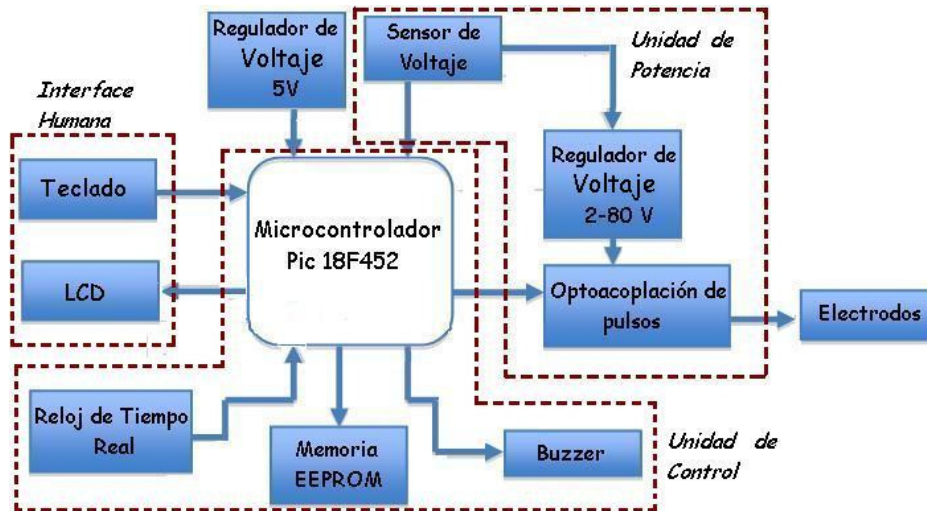


Figura 1.1 “Diagrama a bloques del sistema de electroporación”.

Capítulo 2

Unidad de Control

Actualmente se ha tenido un gran incremento en aplicaciones donde las computadoras digitales son utilizadas para controlar procesos de medición e instrumentación en experimentos complejos de laboratorio. Muchas de las cualidades de una computadora de escritorio, como son la facilidad de procesar la información y obtener resultados, pueden ser obtenidos de igual forma por un microcontrolador.

Casi todo producto electrónico a nuestro alcance tiene al menos un microcontrolador en un su interior. Las aplicaciones electrónicas frecuentemente los usan para motores y elementos de control como son las interfaces de operación, los teléfonos celulares no podrían existir sin un microcontrolador; dentro de las aplicaciones que se pueden nombrar en el uso de los microcontroladores, tenemos mandos a distancia, termómetros digitales, controles de acceso por puertas de seguridad, los sistemas ABS o EPS de los coches, control y sensores de maquinaria, automatización del hogar, microrrobótica, monederos electrónicos..., etc. La actual sociedad moderna tiene más de 5 billones de microcontroladores que son parte esencial de los productos electrónicos que compramos año con año.

Pero... ¿Qué es un Microcontrolador?

2.1 Microcontrolador

Un microcontrolador es un circuito integrado programable que contiene todos los componentes necesarios para controlar el funcionamiento de una tarea determinada. Un sistema con microcontrolador debe disponer de una memoria donde se almacena el programa que gobierna el funcionamiento del mismo, que una vez programado y configurado, sólo sirve para realizar la tarea asignada.

La utilización de un microcontrolador en el circuito reduce notablemente el tamaño y número de componentes y, en consecuencia, disminuye el número de fallas, el volumen y el peso del equipo.

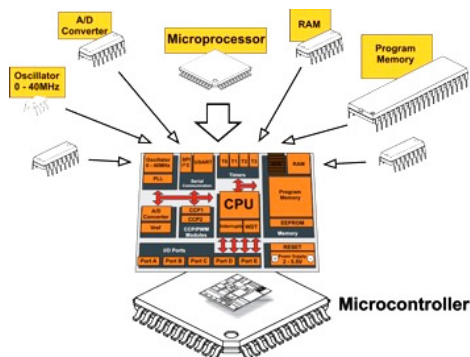


Figura 2.1 “Diagrama a bloques interno de un Microcontrolador.”

Uno de los microcontroladores más populares y fáciles de usar son los de la familia de Microchip Technology conocidos como PIC (por sus siglas en inglés "Peripheral Interface Controller") con cientos de características diferentes, cada uno se diseña para ser óptimo en diferentes aplicaciones. Estas variaciones consisten en su configuración de memoria, diferentes puertos de I/O "del inglés Input/Output", una amplia gama de recursos dedicados, encapsulados y disposición de periféricos. Este amplio intervalo de opciones para el dispositivo no es único de los microcontroladores PIC, existen muchos otros microcontroladores que pueden tener cualidades similares pero con opciones diferentes para el diseñador. La diversidad tiene el objetivo fundamental de reducir costos.

Los objetivos que se persiguen al usar un microcontrolador para controlar el proceso de electroporación incluyen:

- ✓ Eficiencia en la operación.
- ✓ Facilidad de operación.
- ✓ Seguridad.
- ✓ Producto de mayor calidad.
- ✓ Reducción del tiempo de ejecución.
- ✓ Control secuencial.

2.2 Familia de Microcontroladores

Actualmente existen muchos tipos de microcontroladores, hechos por numerosos y diferentes fabricantes. Los fabricantes construyen las familias de microcontroladores alrededor de un microprocesador (μ P), el que está formado por una Unidad de Procesamiento Central (CPU), una Unidad Lógica Aritmética (ALU) y dispositivos de Entrada/Salida (E/S). Los diferentes miembros de una familia son creados usando el mismo CPU, un CPU se combina con diferentes tipos de periféricos y tamaños de memoria creando así diferentes tipos de familia del microcontrolador. Esto es mostrado simbólicamente en la figura 2.2. Un CPU puede ser de 4 bits, son los más sencillos en todos los aspectos y de muy bajo costo, otro CPU es de 8 bits son los más usados actualmente por su gran diversidad y versatilidad, los CPU de 16 bits con aplicaciones típicas en el procesamiento digital de señales, y los CPU más sofisticados son de 32 bits que son más complejos y costosos con aplicaciones en inteligencia artificial, aplicaciones militares y almacenamiento masivo de datos. Existen cientos de microcontroladores dentro de cada familia cada uno con pequeñas diferencias y capacidades y algunos dirigidos hacia aplicaciones muy específicas.

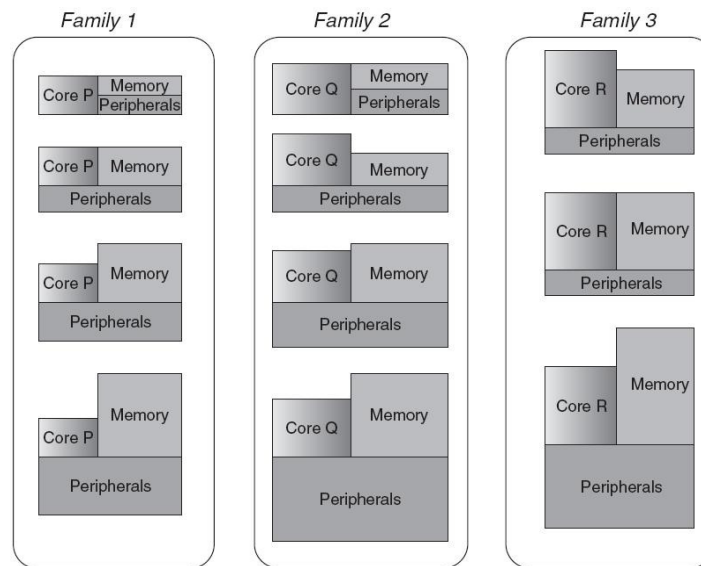


Figura 2.2 “Diferentes tipos de familia de microcontroladores”

2.3 Características del Microcontrolador.

La familia de la serie PIC16 (PIC16-series) son microcontroladores que han estado en el mercado por varios años. A pesar que son excelentes microcontroladores de propósito general, tienen ciertas limitaciones. Por ejemplo, la capacidad de la memoria de programa y de datos son limitadas, la pila es pequeña y la estructura de interrupciones es primitiva, y todas las fuentes de interrupción comparten el mismo vector de interrupción. La serie PIC16 de igual forma no posee el soporte para interfaces avanzadas como el USB, bus CAN, etc. Hacer la interfase con estos dispositivos no es del todo fácil. El conjunto de instrucciones para estos microprocesadores esta limitado, por ejemplo, no hay instrucciones de multiplicación o división, y los saltos son bastante simples, siendo una combinación de instrucciones “skip” y “goto”.

De ahí que Microchip Inc. haya desarrollado la serie de microcontroladores PIC18 para uso en aplicaciones complejas. Los microcontroladores PIC18 ofrecen soluciones eficiencia-costo para aplicaciones de propósito general escritas en C que usa un sistema operativo en tiempo real (RTOS) y requiere un complejo protocolo de comunicación con la pila, así como: TCP/IP, CAN, USB, o ZigBee. La serie PIC18F proveen una memoria de programa flash de un tamaño que va desde 8 hasta 128 kbytes y una memoria de programa desde los 256 hasta 4 kbytes, con un intervalo de operación desde los 2.0 V hasta los 5.0V, y velocidades desde 0 Hz hasta 40MHz.

Por éstas razones, el desarrollo del sistema de control para el sistema de electroporación se centra en la utilización del microcontrolador PIC 18F452. El cual se describirá para entender mejor la arquitectura de software que se implantará en él.

Las características básicas de PIC18F serie de microcontroladores, son:

- ✓ 77 instrucciones.
- ✓ Compatibilidad de código con PIC16.
- ✓ Memoria del programa mayor a 2 Mbytes.
- ✓ Memoria de datos mayor a 4 kbytes.

- ✓ Operación desde 0 Hz a 40 MHz.
- ✓ Multiplicadores de hardware de 8x8.
- ✓ Niveles de prioridad en interrupción.
- ✓ Instrucciones con ancho de palabra de 16-bit, datos con ancho de palabra de 8-bits.
- ✓ Hasta dos temporizadores/contadores de 8-bits.
- ✓ Hasta tres temporizadores/contadores de 16-bits.
- ✓ Hasta cuatro interrupciones externas.
- ✓ Terminales con suministro de corriente de 25 mA.
- ✓ Hasta cinco módulos de captura/compare/PWM.
- ✓ Master módulo de puerto serie síncrono (SPI y I²C modos).
- ✓ Hasta dos módulos USART.
- ✓ Puerto esclavo paralelo (PSP).
- ✓ Convertidor analógico-digital de 10-bits.
- ✓ Módulo de Programable de detección de bajo voltaje (LVD).
- ✓ Power-on reset (POR), el encendido del temporizador (PWRT), y empezar a oscilador-up Temporizador (OST).
- ✓ Con temporizador, perro guardián (Watchdog , WDT) con el oscilador RC.
- ✓ Programación In-circuit

Además, algunos microcontroladores de la serie PIC18F ofrecen las siguientes características especiales:

- ✓ Interfase directa bus CAN 2.0
- ✓ Interfase directa bus USB 2.0
- ✓ Interfase TCP/IP
- ✓ Interfase ZigBee

2.3.1 *Arquitectura Interna.*

Al igual que los demás miembros de su familia, el PIC18F452 se caracteriza por:

- ✓ Tener una arquitectura Harvard.
- ✓ Su procesador es segmentado ó Pipeline.
- ✓ Su procesador es tipo RISC.
- ✓ El formato de las instrucciones es ortogonal.
- ✓ La arquitectura está basada en banco de registros.

2.3.2 *Arquitectura Harvard.*

Esta arquitectura dispone de dos memorias independientes a las que se conecta mediante dos grupos de buses separados.

- ✓ Memoria de datos
- ✓ Memoria de programa.

Ambos buses son totalmente independientes y pueden ser de distintos anchos, esto permite que la ALU pueda acceder de forma independiente y simultánea a la memoria de datos y la de instrucciones, consiguiendo que las instrucciones se ejecuten en menos ciclos de reloj.

Esta distribución de la memoria permite la adecuación del tamaño de las palabras y los buses a los requerimientos específicos de las instrucciones y los datos. Se puede concluir que las principales ventajas de esta arquitectura son.

El tamaño de las instrucciones no está relacionado con el de los datos y por lo tanto, puede ser optimizado para que cualquier instrucción ocupe una sola posición de memoria de programa. Así se logra una mayor velocidad y una menor longitud de programa. El tiempo de acceso a las instrucciones puede superponerse con el de los datos, logrando una mayor velocidad de operación.

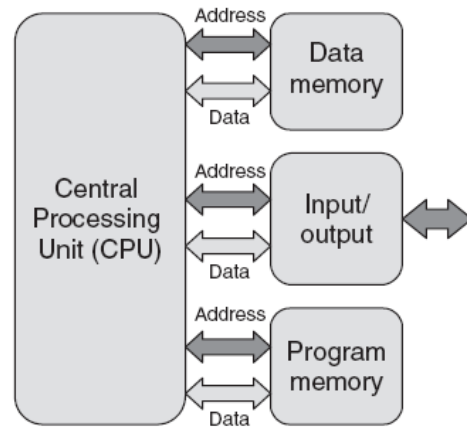


Figura 2.3 "Arquitectura Harvard"

2.4 Características del Microcontrolador PIC18F452

Dentro del circuito integrado que define al microcontrolador existen los recursos y prestaciones limitadas que lo definen.

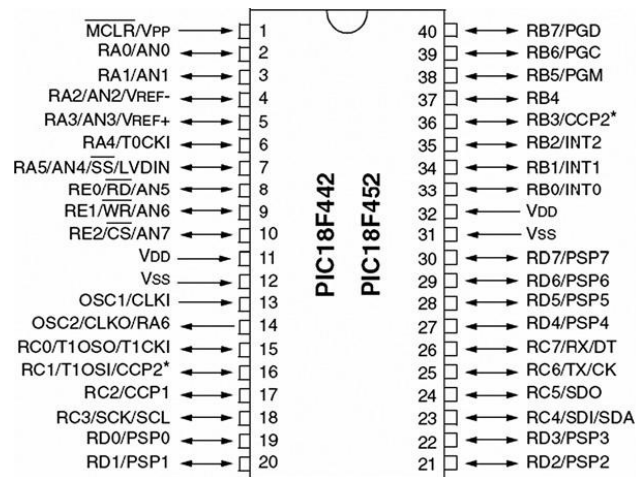


Figura 2.4 “Configuración de Terminales encapsulado DIP microcontrolador PIC 18F452”

Unidad de proceso:

Procesador.	
Memoria de programa.	32 kbytes
Memoria de datos	163864 bytes
Puertos de E/S	(A, B, C, D)

Periféricos Complementarios:

Temporizadores	(4)
Convertidores A/D	10 bits, 8 canales de entrada
Puertos de comunicación	MSSP, USART, PSP, I ² C
Módulos	de 2
Captura/Comparación/PWM	
Otros.	

Recursos Auxiliares:

- Circuito de reloj.
- Modos de bajo consumo.
- Perro guardián.
- Reset al conectar la alimentación.

Otros:

- Tres Terminales para interrupción externa.
- Temporizador 0: 8bit /16 bit/ contador con pre-escalador programable a 8bits.
- Temporizador 1: 16-bit/contador.
- Temporizador 2: 8-bit/ contador con 8-bit registros de periodo (base de tiempo para PWM).
- Temporizador 3: 16-bit/contador.
- Entrada de reloj por oscilación secundaria- Temporizador1 y Temporizador 3.

Dos módulos de Captura/Comparación/PWM (CCP). Las Terminales del CCP pueden ser configurados como:

1. Entrada de Captura: La captura es de 16-bits máximo, con resolución de 6.25 ns (TCY /16).
2. Comparación de 16-bits máximo resolución de 100ns (TCY).
- 2 Salida de PWM: el PWM alcanza una resolución de 1-10 bits, máximo La frecuencia del PWM es

8-bits de resolución = 156 kHz
10-bits de resolución = 39 kHz

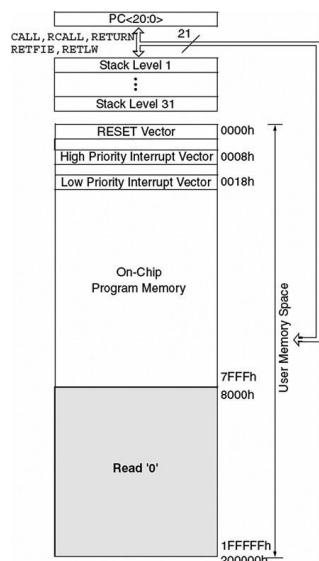
- 3 Modulo de Comunicación I²C Maestro/Esclavo

2.4.1. Memoria EEPROM

La EEPROM, (por sus siglas en inglés (Electrically Erasable Programmable Read Only Memory), es una memoria no volátil que se puede ser borrada y reprogramar usando un dispositivo de programación adecuado, las memorias EEPROM son utilizadas para guardar información de configuración, valores de variables, datos de identificación, etc. Algunos microcontroladores tienen incorporados una memoria EEPROM. Por ejemplo, el PIC18F452 contiene una memoria EEPROM de 256 bytes en el que cada byte puede ser programado y borrado directamente por software.

2.4.2. Organización de la memoria de programa (ROM Flash)

El microcontrolador está diseñado, para que en su memoria de programa se almacenen todas las instrucciones del programa de control. En sus 32,768 posiciones (32k) contiene el programa a ejecutar con las instrucciones que gobiernan la aplicación organizadas en palabras de 16 bits. Así pues, la memoria de programa comienza en la posición 0000h (posición inicial de reset) y llega hasta la 7FFFh. Está memoria es no volátil de tal forma que su grabado es de forma permanente. La información contenida en éstas memorias debe ser grabada previamente con un equipo físico denominado programador o grabador.



La figura 2.5 representa el espacio de la memoria de programa de la serie PIC18F, con las direcciones más importantes junto al Contador de Programa de 21 bits con la capacidad de direccionamiento de 2 Mbytes de espacio de memoria.

Las direcciones 0008H y 0018H son reservadas para vectores de interrupción de alta y baja prioridad respectivamente y es donde se debe de inicializar las rutinas de servicio de interrupción según sea el caso.

Figura 2.5 “Mapa de la memoria de programa del PIC 18F452”

2.4.3. Organización de la memoria RAM (Random Access Memory)

La memoria RAM o simplemente memoria (interna o principal) se utiliza para almacenar variables y datos; estos datos varían continuamente, por lo que esta memoria debe ser de lectura y escritura (volátil). La memoria RAM consta de un conjunto de celdas de memoria denominadas también palabras, el número de celdas de memoria suele ser de (1k, 2k, 4k, 8k, 16k, etc). Cada celda de memoria consta de un cierto número de bits. La unidad elemental de memoria se llama byte y está formado por un conjunto de unidades más pequeñas de almacenamiento denominadas bits, que son dígitos binarios (0 ó 1), por definición, se acepta que un byte contiene ocho bits.

Cada celda o byte tiene asociada una única dirección que indica su posición relativa en memoria y mediante la cual se puede acceder a la posición para almacenar o recuperar información. La información almacenada en una posición de memoria es su contenido. El contenido de estas direcciones o posiciones de memoria se llaman palabras.

En la memoria RAM se almacenan:

- ✚ Los datos enviados para procesarse desde los dispositivos de entrada.
- ✚ Los programas que realizarán los procesos.
- ✚ Los resultados obtenidos preparados para enviarse a un dispositivo de salida.

El mapa de la memoria RAM del PIC 18F452 se muestra en la figura 2.6 el bus de direcciones de la memoria de datos es de 12 bits con la capacidad de direccionamiento de 4 Mbytes. Un problema con cualquier espacio de memoria es el gran tamaño de direccionamiento del bus que debe tener. Una forma de evitar grandes buses de direcciones es dividir la memoria dentro de pequeños bloques llamados bancos de memoria cada uno de tamaño idéntico. Así un bus de direccionamiento de memoria más pequeño puede ser usado. La memoria en general del PIC18F452 consiste en 16 bancos, cada uno de 256 bits, donde solo 6 bancos son usados. El intercambio entre bancos ocurre automáticamente cuando se usan compiladores de alto nivel, de esta forma el programador no se debe preocupar por la selección del banco de memoria durante la programación.

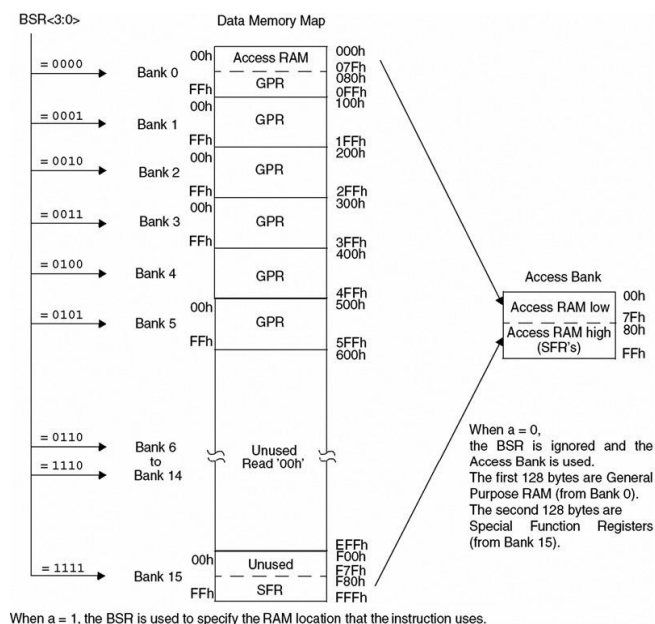


Figura 2.6 “Mapa de la memoria de datos del PIC 18F452”

La memoria de datos tiene posiciones implementadas en RAM y otras en EEPROM. En la sección RAM, se alojan los registros operativos fundamentales en el funcionamiento del procesador y en el manejo de todos sus periféricos, además de registros que el programador puede usar para información de trabajo propia de la aplicación. La RAM estática consta de 4 bancos con 128 bytes cada uno, en las posiciones iniciales de cada banco se ubican los registros de funciones especiales (SFR) y propósito general (GPR) que gobiernan al procesador y sus recursos.

Los registros del SFR están agrupados entre las direcciones 00h a FFh, mientras que el banco de registro de propósito general está formado por 68 posiciones de memoria, ya que sólo son operativos los del Banco 15 (direcciones desde la F80h hasta la FFFh).

Address	Name	Address	Name	Address	Name	Address	Name
FFFh	TOSU	FDfH	INDF2 ⁽³⁾	FBFh	CCPR1H	F9Fh	IPR1
FFEh	TOSH	FDEh	POSTINC2 ⁽³⁾	FBEh	CCPR1L	F9Eh	PIR1
FFDh	TOSL	FDDh	POSTDEC2 ⁽³⁾	FBDh	CCP1CON	F9Dh	PIE1
FFCh	STKPTR	FDCh	PREINC2 ⁽³⁾	FBCh	CCPR2H	F9Ch	—
FFBh	PCLATU	FDBh	PLUSW2 ⁽³⁾	FBBh	CCPR2L	F9Bh	—
FFAh	PCLATH	FDAh	FSR2H	FBAh	CCP2CON	F9Ah	—
FF9h	PCL	FD9h	FSR2L	FB9h	—	F99h	—
FF8h	TBLPTRU	FD8h	STATUS	FB8h	—	F98h	—
FF7h	TBLPTRH	FD7h	TMR0H	FB7h	—	F97h	—
FF6h	TBLPTRL	FD6h	TMR0L	FB6h	—	F96h	TRISE ⁽²⁾
FF5h	TABLAT	FD5h	T0CON	FB5h	—	F95h	TRISD ⁽²⁾
FF4h	PRODH	FD4h	—	FB4h	—	F94h	TRISC
FF3h	PRODL	FD3h	OSCCON	FB3h	TMR3H	F93h	TRISB
FF2h	INTCON	FD2h	LVDCON	FB2h	TMR3L	F92h	TRISA
FF1h	INTCON2	FD1h	WDTCON	FB1h	T3CON	F91h	—
FF0h	INTCON3	FD0h	RCON	FB0h	—	F90h	—
FEFh	INDF0 ⁽³⁾	FCFh	TMR1H	FAFh	SPBRG	F8Fh	—
FEeh	POSTINC0 ⁽³⁾	FCEh	TMR1L	FAEh	RCREG	F8Eh	—
FEDh	POSTDEC0 ⁽³⁾	FCDh	T1CON	FADh	TXREG	F8Dh	LATE ⁽²⁾
FECh	PREINC0 ⁽³⁾	FCCh	TMR2	FACH	TXSTA	F8Ch	LATD ⁽²⁾
FEbh	PLUSW0 ⁽³⁾	FCBh	PR2	FABh	RCSTA	F8Bh	LATC
FEAh	FSR0H	FCAh	T2CON	FAAh	—	F8Ah	LATB
FE9h	FSR0L	FC9h	SSPBUF	FA9h	EEADR	F89h	LATA
FE8h	WREG	FC8h	SSPADD	FA8h	EEDATA	F88h	—
FE7h	INDF1 ⁽³⁾	FC7h	SSPSTAT	FA7h	EECON2	F87h	—
FE6h	POSTINC1 ⁽³⁾	FC6h	SSPCON1	FA6h	EECON1	F86h	—
FE5h	POSTDEC1 ⁽³⁾	FC5h	SSPCON2	FA5h	—	F85h	—
FE4h	PREINC1 ⁽³⁾	FC4h	ADRESH	FA4h	—	F84h	PORTE ⁽²⁾
FE3h	PLUSW1 ⁽³⁾	FC3h	ADRESL	FA3h	—	F83h	PORTD ⁽²⁾
FE2h	FSR1H	FC2h	ADCON0	FA2h	IPR2	F82h	PORTC
FE1h	FSR1L	FC1h	ADCON1	FA1h	PIR2	F81h	PORTB
FE0h	BSR	FC0h	—	FA0h	PIE2	F80h	PORTA

Figura 2.7 “Registros SFR del PIC18F452”

Explicar a detalle el funcionamiento del CPU esta lejos del alcance de este proyecto, lo que es importante decir, es que: el CPU esta hecho con tecnología RISC y que la reducción del set de instrucciones le da a los microcontroladores PIC dos grandes ventajas. Además el CPU puede reconocer y ejecutar 35 instrucciones simples.

El tiempo de ejecución es casi el mismo para todas las instrucciones, la única excepción son los saltos y las ramas de la instrucción cuyo tiempo de ejecución es el doble de largo. Esto hace que la velocidad de operación del microcontrolador sea de 20 MHz, el tiempo de ejecución será de 20 ns, el programa se ejecutará a una velocidad de 5 millones de instrucciones por segundo.

La ALU (por sus siglas en inglés "Arithmetic Logic Unit") realiza operaciones aritméticas y lógicas, tales como suma, resta, multiplicación, división y comparaciones. La ALU del PIC18F452 es de 8 bits y también el registro de trabajo W, del que normalmente recibe un operando que puede ser cualquier registro, memoria, puerto de entrada/salida o el propio código de instrucción.

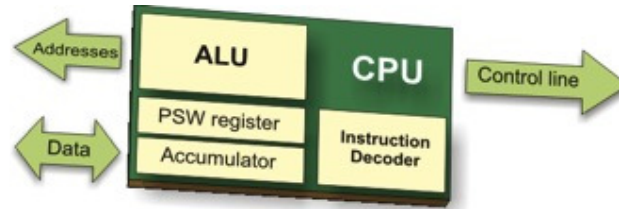


Figura 2.8 "Diagrama a bloques del µC"

Existen cinco puertos para la comunicación con el mundo exterior:

- PORTA de 7 bits <RA6:RA0>
- PORTB de 8 bits <RB7:RB0>
- PORTC de 8 bits <RC7:RC0>
- PORTD de 8 bits <RD7:RD0>
- PORTE de 3 bits <RE2:RE0>

Estos puertos de entrada, sirven para introducir datos (información) al microcontrolador para su proceso. Los datos se leen de los puertos de comunicación y se almacenan en la memoria central o interna (Flash). Los puertos de entrada convierten la información de entrada en señales eléctricas que se almacenan en la memoria central. Los puertos de salida permiten representar los resultados (salida) del proceso de los datos

El contador de programa coordina las actividades del microcontrolador y determina cuales operaciones se deben realizar y en qué orden, además, controla y sincroniza todo el proceso de la µC. Cuenta con 21 bits, lo que en teoría permitiría direccionar 2 Mbytes de palabras de memoria.

2.4.4. Interrupciones

Las interrupciones son un importante concepto de lo que es un microcontrolador. Una interrupción provoca que el microcontrolador responda a eventos tanto externos como internos. Cuando una interrupción ocurre, el µC abandona el programa o rutina que se ejecuta y salta a la subrutina de programa conocida como ISR (por sus siglas en inglés Interrupt Service Routine). El código dentro de la rutina ISR se ejecuta, al finalizar su ejecución la rutina de programa ISR, regresa el control del programa a la rutina interrumpida reanudando así el flujo normal del programa.

El ISR comienza desde una localidad de memoria de programa conocida normalmente como dirección del vector de interrupción. Algunos µC con múltiples interrupciones tienen tan solo una dirección de vector de interrupción, mientras que otros tienen diferentes vectores de interrupción, una dirección para cada fuente de interrupción. Una característica importante de un sistema con múltiples interrupciones es que para diferentes fuentes de interrupción se pueden asignar diferentes niveles de prioridad.

2.4.5. Convertidor Analógico Digital

El convertidor Analógico-Digital (A/D) es usado para convertir señales analógicas como puede ser un voltaje, la voz, señales de salida de sensores electrónicos, así como muchas de las señales que viajan a través de un medio guiado como un cable, o no guiado como es el aire que son de tipo continuo y pueden tomar valores infinitos a lo largo del tiempo. Con el fin de adaptar las señales a circuitos digitales para su lectura, almacenamiento o procesamiento de un μC es necesario digitalizar la señal analógica acotándola dentro un intervalo de tiempo (muestreo) y con valores de voltaje (niveles de cuantización). El muestreo implica que tenemos que tomar una muestra de la señal cada T segundos ya que no hay memoria suficiente capaz de almacenar los puntos infinitos de una señal en cualquier intervalo de tiempo.

La cuantificación implica adquirir una muestra de la amplitud de la señal redondeándola a valores fijos de amplitud. Estos valores van a depender del número de bits que se vayan a almacenar para cada muestra. Los convertidos A/D generalmente son de 8 a 10 bits teniendo niveles de cuantización desde los 256 hasta los 1024 niveles. Muchos de los μC PIC tienen múltiples entradas para una conversión A/D. Los convertidores A/D son inicializados desde el programa del usuario y le toma algunos microsegundos en completar la conversión. Los convertidores A/D generalmente originan interrupciones una vez que la conversión de la señal es completada.

2.4.6. Temporizadores

Los temporizadores (Timers) son parte importante de un microcontrolador. Un temporizador es básicamente un contador, el cual recibe impulsos de reloj ya sea de una fuente externa o del oscilador interno del propio μC . Un temporizador puede tener una longitud de 8 o 16 bits, la información puede ser cargada por el programa de control hacia el temporizador, el cual puede iniciar o detener su funcionamiento. La mayoría de los temporizadores son configurados para generar una interrupción cuando alcanzan cierto valor de cuenta (generalmente cuando el temporizador se desborda). El programa principal puede usar esta interrupción para cumplir con tiempos precisos de alguna operación dentro del microcontrolador. Los microcontroladores PIC18F tienen por lo menos tres temporizadores como en el caso del PIC 18F452. Algunos μC tienen incorporados módulos de captura y comparación, donde el valor del temporizador es leído cuando ocurre un evento externo ó el valor del temporizador es comparado con algún valor preestablecido generando una interrupción cuando la comparación es exitosa. Los PIC18F cuentan por lo menos con 2 módulos de este tipo.

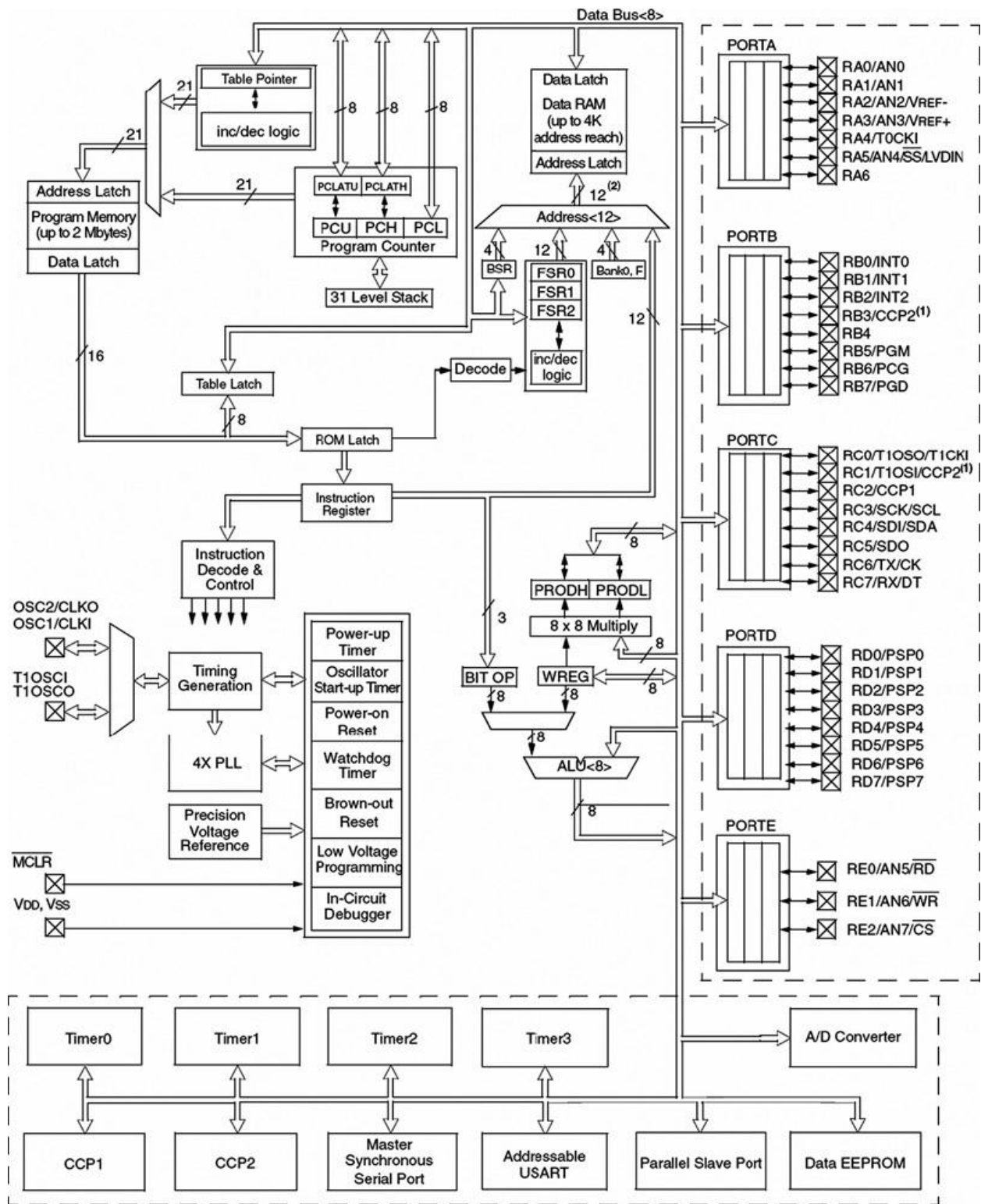


Figura 2.9 “Diagrama de bloques de un microcontrolador PIC 18F452”

2.5 Herramientas de desarrollo de software

Las herramientas de desarrollo de software son programas de computadora, que permiten al programador crear, modificar y probar las aplicaciones de nuevos programas, así mismo, ayudan a los programadores a desarrollar y probar sistemas complejos en un tiempo relativamente corto.

Algunas herramientas comunes de desarrollo de software son:

- 4 Editores de texto
- 5 Ensambladores / Compiladores
- 6 Simuladores
- 7 Simuladores en lenguaje de alto nivel
- 8 Entornos integrados de Desarrollo (IDE)

2.5.1. Editores de Texto

Un editor de texto es usado para crear o editar programas y archivos de texto. Windows posee un editor de texto llamado bloc de notas (Notepad). Usando bloc de notas, podemos crear un nuevo archivo de programa, modificar un archivo ya existente, o visualizar o imprimir el contenido de un archivo. La mayoría de los ensambladores y compiladores vienen con editores de texto ya instalados, haciendo posible la creación de programas y después ensamblarlos o compilarlos sin la necesidad de salir del editor.

2.5.2. Ensambladores y compiladores

Los lenguajes ensamblador, generan código maquina ejecutable, que puede ser cargado dentro de la memoria flash del microcontrolador.

Los compiladores, generan código maquina ejecutable desde un lenguaje de alto nivel. Los compiladores más usados para la serie PIC18 son BASIC, C y PASCAL.

Los lenguajes ensamblador son usados en aplicaciones donde la velocidad de procesamiento es crucial y el microcontrolador debe responder a interrupciones internas y externas en un tiempo sumamente pequeño, sin embargo, es difícil desarrollar programas complejos usando código ensamblador.

Los lenguajes de alto nivel, por otra parte, son fáciles de aprender y programas complejos pueden ser desarrollados y probados en un periodo de tiempo sumamente corto.

Existen variedad de compiladores de C disponibles para el desarrollo de microcontroladores PIC, algunos de los más populares son:

- CCS C ^[1]
- Hi-Tech C ^[2]
- C18 C ^[3]
- mikroC C ^[4]
- Wiz-C C ^[5]

Sin embargo, la mayoría de los compiladores C en esencia son lo mismo, cada uno posee sus propias características y modificaciones o se ha agregado algún tipo de ventajas al lenguaje estándar.

2.5.3. Simuladores

Un simulador es un programa de computadora que corre sobre una plataforma PC y es usado para probar el programa de aplicación sin la necesidad del hardware de aplicación. Simula el comportamiento de la aplicación en hardware interpretando las instrucciones del programa creado; provee ayuda en la revisión de correcciones de un algoritmo o un programa y pueden ser corregidos bastantes errores durante la simulación.

Los simuladores pueden visualizar el contenido de registros, memoria de datos y el estatus de los puertos de entrada y salida dentro del microcontrolador. De igual forma el programa puede ser ejecutado en modo paso a paso (single-step), de esta forma la memoria y los registros pueden ser examinados hasta que el programa trabaje perfectamente, el código máquina ejecutable es cargado al hardware de aplicación (microcontrolador) por medio de un programador.

Algunos programas en ensamblador, contienen simuladores como por ejemplo:

MPLAB IDE ^[3]

Oshon Software PIC18 simulator ^[6]

Forest Electronics PIC18 assembler ^[5]

2.5.4. Entornos integrados de Desarrollo (IDE)

Los entornos integrados de Desarrollo (IDE) son herramientas poderosas de programación sobre una plataforma PC, los cuales incluyen todo: editores, ensambladores, compiladores, administradores de proyecto (Linkers), simuladores, etc. Pueden depurar los programas y descargar el código máquina ejecutable hacia el microcontrolador usando un dispositivo de programación. Estos programas poseen entornos gráficos de desarrollo (GUI), donde el programador puede seleccionar varias opciones desde el programa sin la necesidad de salirse de él. Los IDE son útiles cuando se programan microcontroladores.

El entorno de programación usado en ésta tesis es el MPLAB IDE, que es un “paquete” que coordina todas las herramientas en una simple interfase gráfica automática. Por ejemplo, una vez que el código es escrito, puede ser convertido a instrucciones ejecutables y descargar el código en el microcontrolador para verificar su funcionamiento. MPLAB IDE tiene los siguientes componentes internos:

Project Manager: El administrador de proyectos provee integración y comunicación entre el IDE y las herramientas del lenguaje.

Editor: El editor es una herramienta completa de programación que también sirve como una ventana dentro del depurador.

Ensamblador y herramientas de lenguaje: El ensamblador puede ser usado por si solo para ensamblar un archivo simple, o puede ser usado con el linker para construir un proyecto

desde archivos fuente por separado, bibliotecas y objetos recompilados. El linker es el responsable de posicionar el compilador de código dentro de las áreas de memoria del microcontrolador a usar.

Debugger: El depurador de Microchip permite puntos de ruptura del programa (breakpoints), compilación paso a paso, y ventana de visualización de variables (watch window), y todas las características de compilación de los modernos IDE.

Simulación: Existe software de simulación en MPLAB IDE para todos los dispositivos y los μ C PIC y dsPIC. Estos simuladores usan el PC para simular las instrucciones del μ C y algunas funciones propias de los μ C PIC y dsPIC. De manera opcional tenemos emuladores In-circuit y depuradores in-circuit que están disponibles para censar el código mientras se ejecuta en la aplicación de hardware correspondiente.

2.5.5. Componentes adicionales para MPLAB IDE

Algunos componentes opcionales que pueden trabajar en conjuntos con MPLAB IDE son:

- **Lenguajes de Compilación:** MPLAB C18^[7] y MPLAB C30^[8] compiladores en C de Microchip proveen una amplia integración con el entorno de MPLAB IDE, optimizando código. En conjunto con los compiladores HI-TECH^[2,9], IAR^[10], microEngineering Labs^[11], CCs^[1] y Byte Craft^[12], son adjuntados por el administrador de proyecto de MPLAB IDE para compilar el código, esto hace que el código sea automáticamente cargado al μ C, compilado, depurado y verificado en un instante.
- **Programadores:** MPLAB PM3^[13], PICSTART Plus^[14], PICKit^[15] 1, 2 y 3, así como MPLAB ICD 2^[16] depurador y MPLAB REAL ICE^[17] in-circuit emuladores, pueden programar código dentro de los μ C ha utilizar.
- **In-Circuit Emuladores:** MPLAB REAL ICE y MPLAB ICE 2000 son sistemas emuladores in-circuit para μ C PIC y dsPIC. Son conectados a la PC y permiten un completo control sobre las aplicaciones del microcontrolador.

2.6 Ciclo de Desarrollo

El proceso de desarrollo de una aplicación es frecuentemente descrito, como ciclo de desarrollo, rara vez todos los pasos de diseño e implementación pueden ser hechos perfectamente a la primera vez, cada vez más código es escrito, probado y modificado de manera de producir una aplicación que trabaje correctamente. El ciclo de desarrollo empieza escribiendo el programa de aplicación usando un editor de texto, luego el programa es traducido a código máquina con la ayuda de ensambladores o compiladores. Si el programa posee muchos módulos, un organizador de proyectos (linker manager) es usado para combinarlos en una sola aplicación, cualquier error de sintaxis es detectado por el ensamblador o compilador y debe ser corregido antes que se genere el código ejecutable del microcontrolador.

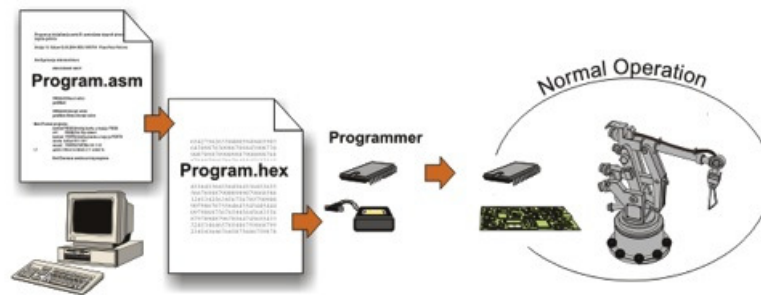


Figura 2.10“Diagrama esquemático del ciclo de desarrollo.”

2.7 Bus I²C

El bus I²C (por sus siglas en inglés Inter-Integrated Circuit) es un bus de comunicación serial síncrona desarrollado por Phillips Semiconductors a principios de los años 80's, con la principal intención de interconectar una cierta cantidad de circuitos integrados dentro de sus diversos productos electrónicos.

En el bus son utilizadas dos líneas de comunicaciones, una para los datos llamada SDA y otra para el reloj SCL. Cada dispositivo que se conecta al bus es direccionable por software, a través de una única e irreplicable dirección dentro del bus. La misma es determinada a través de una combinación de Hardware/Software, donde el fabricante define los bits más significativos.

El número de dispositivos que pueden ser conectados al bus I²C, está limitado por la capacitancia que estos representen al mismo, a un máximo de 400 pF. Las transferencias de información se realizan a través de paquetes de 8 bits bidireccionales, y pueden ser efectuadas a tres velocidades o modos: el modo normal (Standard) a unos 100 kbit/s, modo rápido a 400 kbit/s, o modo de alta velocidad 3.4 Mbit/s

Los dispositivos pueden clasificarse en maestro (master o principal) o esclavo (slave o secundario). El maestro es el que inicia la transferencia de datos y genera la señal de reloj. Cualquiera de los dispositivos seleccionado por un maestro se considera un esclavo. El I²C es un bus multi-maestro; puede haber más de un maestro conectados y controlando el bus.

Tanto las líneas SDA como SCL son líneas bidireccionales que se conectan a +V_{dd} mediante resistores de jalón (pull-up), tal y como se muestra en la figura 2.11.

Cuando el bus esta libre, ambas líneas están a un nivel lógico 1, los transistores de salida conectados a las líneas del bus I²C deben ser de colector abierto para que todos ellos se puedan conectar entre si formando una conexión tipo AND.

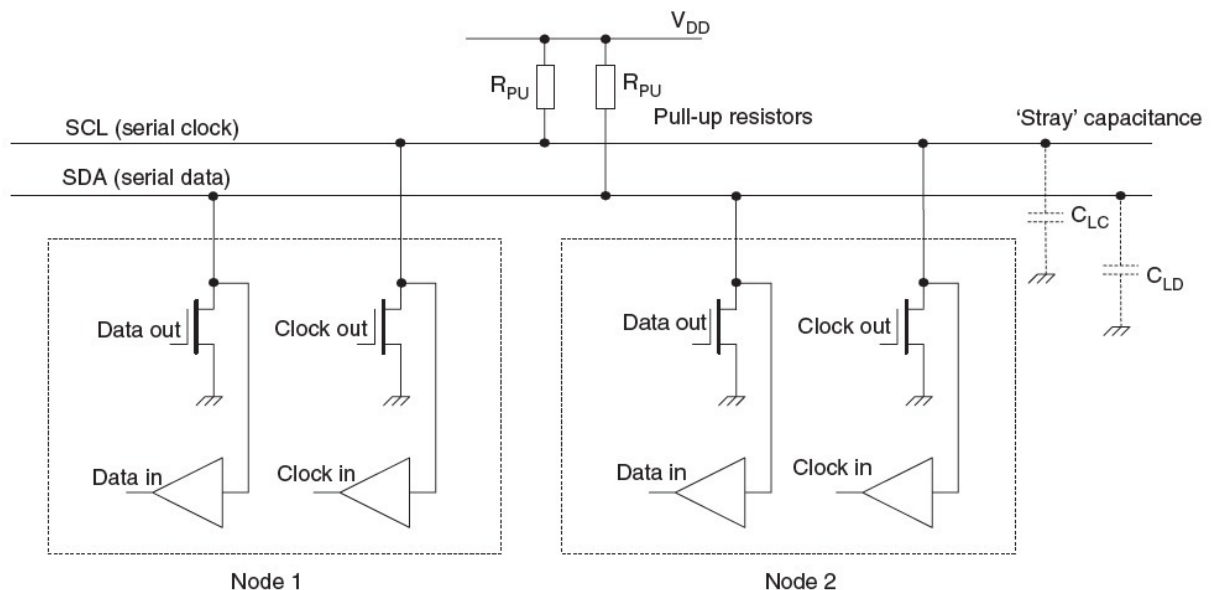


Figura 2.11 "Diagrama de interconexión I²C"

2.8 Memoria EEPROM de datos

Es una pequeña área de memoria de datos de lectura y escritura no volátil, gracias a la cual, un corte del suministro de la alimentación no ocasiona la pérdida de la información, que estará disponible al reinicializarse el programa.

Las memorias EEPROM seriales, poseen las siguientes características:

- ✓ Se pueden conectar fácilmente con microprocesadores o microcontroladores, inclusive algunos de ellos tienen Terminales dedicados para esta labor.
- ✓ Transferencia de datos de manera serial, lo que permite ahorrar Terminales del microcontrolador para dedicarlos a otras funciones.
- ✓ Ocupan la décima parte del espacio de las memorias que trabajan en paralelo, esto permite ahorrar dinero debido al menor tamaño del circuito impreso.
- ✓ El consumo de corriente es mucho menor que en las memorias en paralelo, esto las hace ideales para sistemas portátiles que funcionan con baterías.

Estas memoria usan la comunicación a 2 hilos empleando la interfase I²C cuyas referencias más conocidas son 24C64/256/512. La velocidad de transferencia de información para estos dispositivos es de 100-400 kHz (aunque el límite lo impone el protocolo I²C y no la tecnología del dispositivo). Como característica importante de este elemento se tiene la inmunidad al ruido, dado que este integrado tiene filtros en las Terminales de comunicación.

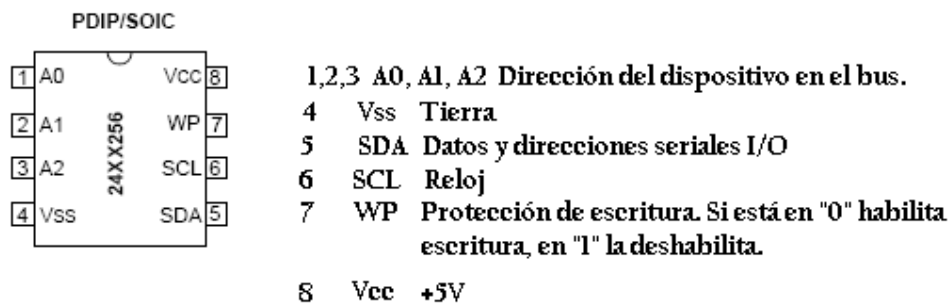


Figura 2.12 "Encapsulado dip 24Cxxx"

Estas memorias utilizan el bus de 2 hilos para comunicarse con otros dispositivos. Dado que cumplen con el protocolo I²C, tiene una terminal llamada SCL que recibe los pulsos generados por el dispositivo maestro (microcontrolador) y otro llamado SDA que maneja el flujo de datos de forma bidireccional (entrada/salida). En la figura 2.12 se muestra el diagrama de terminales correspondiente a estas memorias.

Este dispositivo no requiere de una terminal de habilitación o chip select, ya que en este esquema la transferencia de información solo se puede iniciar cuando el bus este libre. En este caso, como cada dispositivo tiene su dirección determinada mediante las terminales A0, A1 y A2, solamente responderá la memoria cuya dirección coincida con la dirección que va encabezando la trama de información.

2.8.1. Transferencia de la información

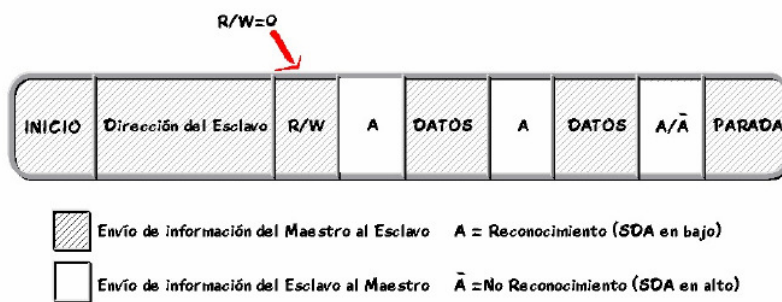
La figura 2.13 muestra la forma del como el microcontrolador entabla comunicación con la memoria, mostrando la serie de bits que debe enviarle a está. La información que el microcontrolador manda a la memoria es la siguiente:

- ✓ Se envía el bit de arranque o start bit
- ✓ El código 1010 (propio de estas memorias).
- ✓ La dirección del dispositivo (A_2, A_1, A_0).
- ✓ Un bit que indica que se desea escribir ("o") en la memoria.

Luego la memoria debe enviar un reconocimiento para informarle al microcontrolador que recibió la información, dicho reconocimiento, llamado ACK, consiste en poner el bus en un nivel bajo (lo hace la memoria). Después, el microcontrolador debe enviar los bits que corresponden a la posición de memoria que se quiere leer o escribir; nuevamente la memoria envía un reconocimiento. El paso siguiente depende de la operación que se vaya a ejecutar.



(a)



(b)

Figura 2.13 (a) "Direccionamiento de dispositivo I²C con dirección de 7 bits" (b) "Direccionamiento de dispositivo I²C con dirección de 10 bits"

Si se trata de un proceso de escritura, el microcontrolador solo debe enviar el dato a ser almacenado y esperar el reconocimiento por parte de la memoria para confirmar que llegó correctamente. Si se trata de una lectura, nuevamente se debe repetir los primeros cuatro pasos, solo que en lugar de un "0" que indica escritura, se debe enviar un "1" que indica lectura. Después se espera el reconocimiento y acto seguido se puede leer el byte con el dato que estaba en la posición de memoria que se indicó anteriormente. Cuando se termina la operación, el microcontrolador debe enviar una señal de parada o stop bit. En la figura 2.15 se muestra el diagrama de tiempos correspondiente a todo el proceso descrito anteriormente.

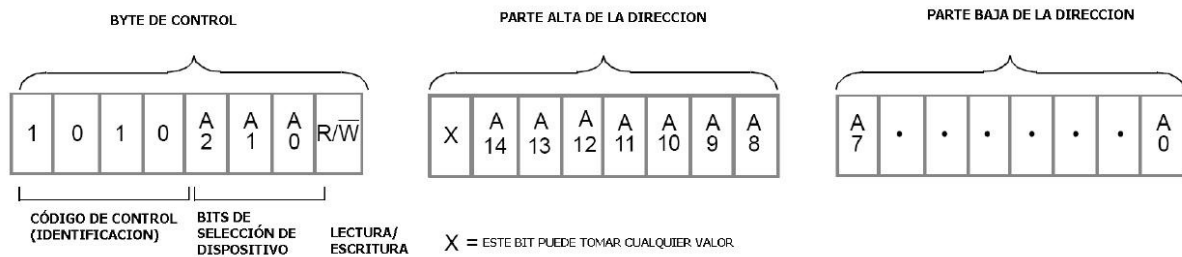


Figura 2.14 "Byte de control y direccionamiento de la memoria 24LC256"

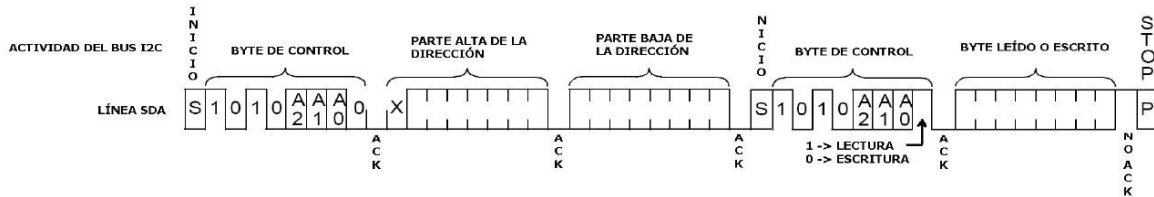


Figura 2.15 "Actividad del bus I²C durante la lectura/escritura de la memoria 24LC256"

2.9 Reloj~Calendario DS1307

El DS1307 es un reloj en tiempo real RTC (por sus siglas en inglés Real Time Clock) con líneas de conexión a un bus I²C. Este circuito integrado es un poderoso reloj y calendario de tiempo real, que cumple perfectamente con muchas de las necesidades normales en la adquisición y registro del tiempo. Sus características más destacadas son:

- ✓ Fabricado por Dallas Semiconductors^[18] en encapsulado de 8 terminales.
- ✓ El DS1307 es un reloj y calendario de tiempo real que cuenta los segundos, los minutos, las horas, los días de la semana, los días del mes, los meses y los años, válido hasta el año 2100.
- ✓ Almacena los datos en formato BCD para que se pueda trabajar directamente con ellos.
- ✓ Tiene 56 bytes RAM no volátil para almacenamiento de datos.
- ✓ En su terminal SQW/OUT proporciona una onda cuadrada programable.
- ✓ Tiene una circuitería interna de “respaldo” para alimentación en caso de fallo de la alimentación principal, por tanto, es capaz de mantener el tiempo y la fecha actualizados aún cuando el sistema esta apagado.
- ✓ Se puede alimentar entre 4.5 a 5.5V, siendo su valor típico 5V.
- ✓ Posee un bajo consumo de corriente menor a 500 nA en el modo de respaldo. Utiliza un cristal de cuarzo propio de 32.768 Hz para lograr tiempos exactos y no depender del microcontrolador.
- ✓ El último día del mes es automáticamente ajustado a 28, 29, 30 ó 31 días según corresponda, tiene en cuenta los años bisiestos.
- ✓ Puede trabajar en formato europeo de 24 horas o el americano de 12 horas con indicador de AM/PM.
- ✓ Se activa cuando recibe la dirección válida indicada en la figura.



Figura 2.17 “Encapsulado DIP DS1307”

2.9.1. *Conexión del DS1307 al microcontrolador*

La figura 2.17 muestra las terminales del reloj DS1307. Las terminales SDA y SCL del DS1307 se conectan a las líneas del μ C que conformen el bus I²C. Los resistores de jalón tienen el valor característico de 4.7 k Ω .

Si falla la alimentación principal, el DS1307 se alimenta de la batería de respaldo, manteniendo la información del tiempo y la fecha mientras la alimentación principal se mantenga apagada. Si esta batería no se utiliza, los datos de calendario del DS1307 no se

actualizarán en caso de fallo de la alimentación. El valor debe estar comprendido entre los 2 y 3.5V. puede usarse cualquier tipo de batería, aunque el fabricante recomienda una batería de litio de al menos 48 mAh que garantiza una conservación de la información en el DS1307 para más de 10 años.

2.9.2. Registros del DS1307

00h	SEGUNDOS
01h	MINUTOS
02h	HORAS
03h	DÍA
04h	FECHA
05h	MES
06h	AÑO
07h	CONTROL
08h	RAM 56X8
.	
.	
.	
3Fh	

En la tabla se muestra el mapa de direcciones de la memoria RAM del DS1307, los registros del calendario se localizan en las direcciones 00h hasta la 07h. Desde la 08h hasta la 3Fh hay 56 posiciones de memoria RAM que pueden ser utilizadas para almacenar datos.

El valor del tiempo y calendario se obtiene mediante la lectura de los registros apropiados. La figura 2.18 muestra los registros del reloj y calendario:

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE
00h	CH	10 Seconds			Seconds			Seconds		00-59
01h	0	10 Minutes			Minutes			Minutes		00-59
02h	0	12	10 Hour	10 Hour	Hours			Hours	1-12 +AM/PM 00-23	
		24	PM/ AM							
03h	0	0	0	0	0	DAY		Day	01-07	
04h	0	0	10 Date		Date			Date	01-31	
05h	0	0	0	10 Month	Month			Month	01-12	
06h	10 Year			Year			Year		00-99	
07h	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08h-3Fh									RAM 56 x 8	00h-FFh

Figura 2.18 “Registros del DS1307”

El formato de todos los datos está en BCD.

El bit 7 del registro 00h es el bit de puesta en marcha Clock Halt (CH):

- ✓ Si CH=0, pone en marcha el reloj.
- ✓ Si CH=1, impide el funcionamiento del reloj, el cual permanecerá parado.

El formato de las horas puede seguir el modelo americano o el europeo controlado por el bit 6 del registro 02h.

- ✓ Si (Bit 12/24)=0, elige el modo europeo de 24 horas.
- ✓ Si (Bit 12/24)=1, elige el modo americano de 12 horas.

En el modo americano de 12 horas, el bit 5 (A/P) del registro 02h es el bit que determina si la hora es AM (A/P=0) ó PM (A/P=1).

En el modo europeo de 24 horas, el bit 5 es el de mayor peso de las decenas de hora. El contenido del registro 07h controla la señal cuadrada de la terminal SQW/OUT.

2.9.3. Escritura en el DS1307

La transferencia de datos desde el μC al DS1307 sigue el procedimiento de escritura del maestro al esclavo de un bus I²C esquematizado en la figura 2.19. La transferencia de datos se efectúa de la siguiente manera:

- ✓ Primero el μC maestro envía la condición de "Start".
- ✓ Luego envía la dirección del DS1307 (Slave Address) en modo escritura que es la b'11010000' ó Doh.
- ✓ A continuación el maestro envía un puntero con la primera dirección del registro a escribir (Word Address).
- ✓ Después se transmiten los datos a escribir. La dirección del registro a escribir se incrementa automáticamente.
- ✓ Cuando termina de escribir el μC maestro envía la condición de "Stop".

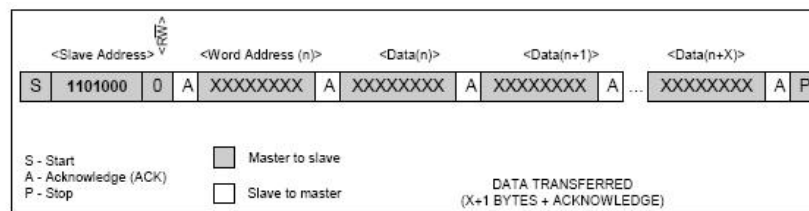


Figura 2.19 "Protocolo de escritura del DS1307"

2.9.4. Lectura del DS1307

La lectura de los datos del DS1307 por parte del microcontrolador sigue el procedimiento de lectura del esclavo por parte del maestro y como se muestra en la Figura 2.20. La transferencia de datos se efectúa de la siguiente manera:

- ✓ Primero el μC maestro envía la condición de "Start".
- ✓ Luego envía la dirección del DS1307 (Slave Address) en modo lectura que es la b'11010001' D1h.
- ✓ Después el maestro lee los datos de los registros. La dirección del registro a leer se incrementa automáticamente.
- ✓ Por último el μC maestro envía la condición de "Stop".

El primer registro leído será el señalado en la última operación anterior realizada. Si este dato se desconoce, primero habrá que realizar una operación de escritura para conocer exactamente el valor de este puntero.

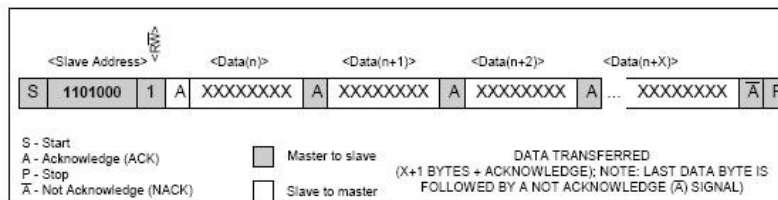


Figura 2.20 "Protocolo de lectura del DS1307"

Capítulo 3

Unidad de Potencia

Una fuente de alimentación de corriente directa cd convierte la energía de una línea de corriente alterna ca en corriente continua ó directa, que es una voltaje constante a un valor deseado. La energía de entrada de ca es rectificadora primeramente para proporcionar una cd pulsante y luego filtrada para producir una voltaje semiestable. Finalmente, el voltaje puede regularse para mantener un nivel de salida constante a pesar de las fluctuaciones de la tensión de línea o de la carga del circuito. La rectificación, el filtrado y la regulación en una fuente de alimentación de cd se muestra en la figura 3.1.



Figura 3.1 “Diagrama a bloque de una fuente regulada”

3.1 Rectificación

El circuito rectificador más adecuado para una aplicación depende de los requerimientos de voltaje y corriente de cd , de la magnitud de la ondulación que puede tolerar el circuito y del tipo de energía de ca disponible. La forma de onda ideal de voltaje debe ser lo más plana posible (acercándose a una cd casi pura), la rectificación se clasifica ya sea como de media onda o de onda completa.

3.1.1. Rectificación de media onda.

El rectificador de media onda generalmente es usado para aplicaciones de baja corriente, o de alta frecuencia, ya que requiere un filtrado mayor para mantener el voltaje de cd que un rectificador de onda completa. La figura 3.2 (a) muestra un circuito rectificador de media onda simple. Cuando el voltaje de entrada es positivo, el diodo está polarizado directamente y es posible sustituirlo por un corto circuito (suponiendo que el diodo sea ideal), por el lado contrario, si el voltaje de entrada es negativo, el diodo está polarizado en forma inversa, y puede sustituirse por un circuito abierto.

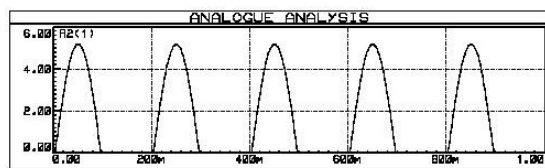
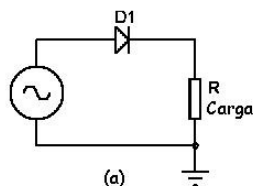
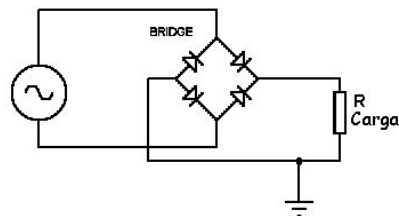


Figura 3.2 (a) “Rectificador de media onda”
(b) “Voltaje de salida de cd ”.

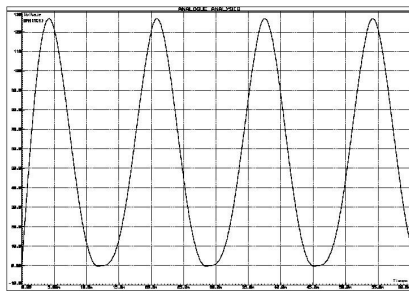
Como se muestra en la figura 3.2 (b) el rectificador de media onda simple no posee una buena aproximación a un voltaje de cd constante a la salida de su forma de onda; posee un factor de rizado significativamente elevado, lo que significa que tiene más componentes de tensión de ca en su salida que componentes de tensión de cd.

3.1.2. Rectificación de onda completa.

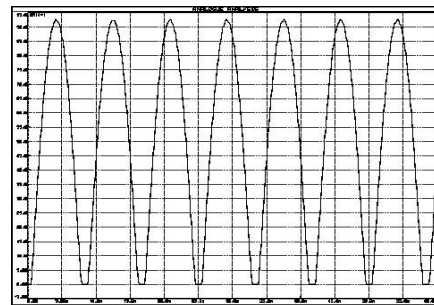
El circuito puente de onda completa transfiere energía de la entrada a la salida durante ambas mitades del ciclo de entrada, como se muestra en la figura 3.3(b) emplea cuatro diodos rectificadores. Es utilizado para exponer a cada diodo individual a la mitad de la tensión inversa de pico para la misma tensión de salida. A través de cada rectificador circula sólo el 50 % de la corriente total.



(a)



(b)



(c)

Figura 3.2.a "Voltaje de entrada de ca", 3.2.b "Voltaje de cd pulsante"

3.2 Filtrado

Las etapas de filtrado se utilizan generalmente para suavizar la ondulación de ca (voltaje de rizo) a la salida de un circuito rectificador está suavización de la onda se conoce como voltaje de rizo. Los filtros pueden ser de dos tipos básicos: de entrada a choke inductivo y de entrada capacitiva figura 3.3.

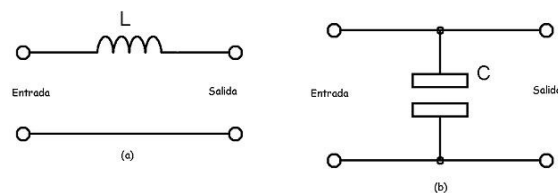


Figura 3.3 (a) "Entrada a choke inductivo" (b) "Entrada capacitiva"

El más simple de estas etapas de filtrado es la de entrada capacitiva, este tipo de filtrado se emplea comúnmente en circuitos de baja corriente en los cuales se puede tolerar un voltaje de rizo bastante grande. Una capacitancia grande reduce el voltaje de rizo pero se debe tener cuidado ya que el rectificador se calienta a causa de las excesivas corrientes eficaces y de pico, esto es, el capacitor se carga hasta aproximadamente el pico de la tensión de entrada en cada semiciclo de conducción del rectificador. La corriente que se dirige a la carga es suministrada entonces por el capacitor, y no por la fuente de alimentación, hasta que la tensión de entrada vuelva a ser igual a la tensión existente a través del capacitor en el semiciclo siguiente. En la figura 3.3 se muestra esta configuración.

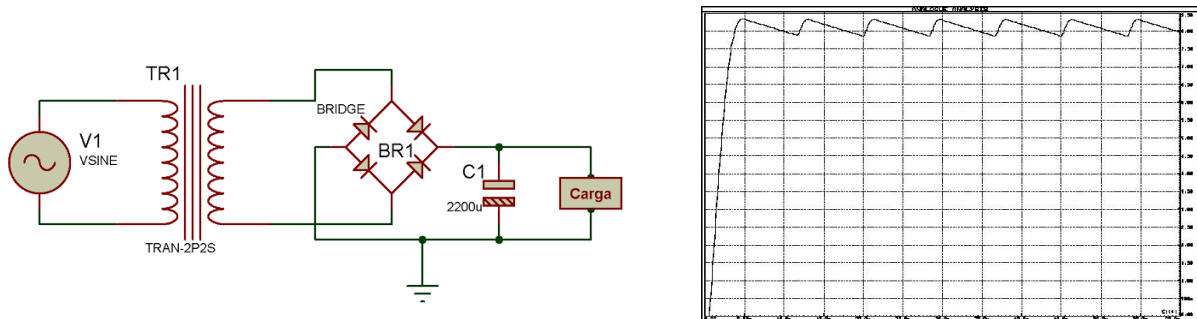


Figura 3.3.a “Fuente de alimentación de *cd* no regulada” 3.3.b “Variación del voltaje de carga de *cd* a plena carga”

3.3 Porcentaje de regulación

Una forma de caracterizar el funcionamiento de una fuente de alimentación de *cd* es mediante el porcentaje de regulación de voltaje para ello, se mide el voltaje sin carga y el voltaje a plena carga de la fuente.

$$\% \text{ de regulación de voltaje} = \frac{V_{cd \text{ NL}} - V_{cd \text{ FL}}}{V_{cd \text{ FL}}} \times 100 \quad (\text{Ec.3.1})$$

$V_{cd \text{ NL}}$: Voltaje de corriente directa sin carga.

$V_{cd \text{ FL}}$: Voltaje de corriente directa a plena carga.

La fuente de alimentación no regulada tiene dos características indeseables: al aumentar el consumo de corriente en la carga, el voltaje de *cd* disminuye mientras que el voltaje de rizo de *ca* aumenta. Para atenuar estas desventajas se añade una sección de regulación de voltaje y por lo tanto se convierte en una fuente regulada de voltaje.

3.4 Regulación

La regulación de una fuente de alimentación de *cd* se efectúa generalmente mediante algún tipo de circuito realimentado que detecta cualquier variación en la salida de *cd* y desarrolla una señal de control para anular esta variación. En consecuencia, la salida se mantiene casi constante.

3.4.1. Reguladores de voltaje Lineales

Los reguladores de voltaje integrados se clasifican de acuerdo con cuatro características:

1. Polaridad: negativa, positiva o bipolares.
2. Cantidad de terminales: tres terminales o varias terminales.
3. Voltaje de salida fijo o variable.
4. Corriente de salida: el valor de las corrientes de salida varia desde 0.1 hasta 3 A.

3.4.2. Características comunes

El voltaje instantáneo de entrada de un regulador integrado siempre debe exceder al voltaje de salida de CD, por una cantidad que por lo general va de 0.5 a 3 V como mínimo. A esta condición se la conoce como voltaje instantáneo mínimo de entrada- salida, voltaje de caída.

Aunque se puede obtener un $V_{cd FL}$ con un valor bastante alto para lograr así un margen amplio, hay que tener en cuenta que el calor disipado por el regulador en el peor de los casos es P . Un $V_{cd FL}$ mayor disipa más calor en el regulador.

$$P=I_{L FL}(V_{cd FL}- V_{O reg}) \quad (\text{Ec. 3.2})$$

$V_{cd FL}$: Voltaje de corriente directa a plena carga.

$I_{L FL}$: Corriente de carga a plena carga.

$V_{O reg}$: Voltaje de salida del regulador.

La parte de voltaje de referencia del regulador es en realidad lo que determina la calidad de la etapa de regulación. Cualquier inestabilidad, ruido o variación térmica de la referencia repercutirá directamente en la salida.

Los circuitos internos de estos dispositivos cuentan con diferentes etapas de seguridad que evitan la falla del regulador apagándolo. Al eliminar la falla, el regulador vuelve a trabajar.

1. **Limitación de corriente:** constituye el control contra cortos, si se rebasa un valor determinado de la corriente de carga, se limita automáticamente la corriente de salida hasta que se elimina la sobrecarga.
2. **Área de seguridad:** mide la diferencia del voltaje y de la corriente de entrada y de salida para comprobar que no exista ninguna combinación no permitida reaccionando ante un excesivo diferencial de entrada/salida.
3. **Límite térmico:** mide la temperatura para verificar que cuentan con una adecuada disipación de calor evitando temperaturas demasiado altas en la unión.

3.5 Driver de la corriente de salida

Muchas aplicaciones requieren suministros de corriente superiores a la capacidad de los reguladores de tres terminales. El empleo de transistores serie externos como los que se muestran en las figuras 3.5(a) y 3.5(b) permite incrementar sustancialmente la intensidad de corriente de salida.

La figura 3.5(a) muestra la configuración básica, mientras que en la figura 3.7(b) se añade una protección de cortocircuito para el transistor serie externo.

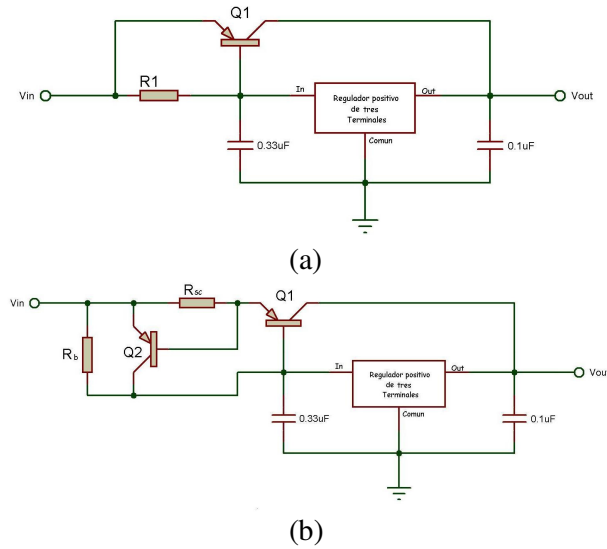


Figura 3.4 “Transistores externos para incrementar la corriente de salida. a) Transistor serie simple; b) elevador de corriente con protección de cortocircuito.”

El valor de R_b establece el punto en el que el transistor externo comienza a conducir y se obtiene de

$$i_{R_1} = i_{R_{e g}} - i_b \quad (\text{Ec. 3.3})$$

$$V_{BE_{Q_1}} = R_1 \cdot i_{R_1} \quad (\text{Ec. 3.4})$$

$$V_{BE_{Q_1}} = R_1 \cdot (i_{R_{e g}} - i_b) \quad (\text{Ec. 3.5})$$

$$i_c = \beta \cdot i_b \quad (\text{Ec. 3.6})$$

$$R_b = \frac{V_{BE(ON)_{Q_1}}}{I_{reg(max)} - \frac{I_{sal}}{\beta_{Q_1}}} \quad (\text{Ec. 3.7})$$

$$I_{sal(max)} = \beta_{Q_1} \left(I_{reg(max)} - \frac{V_{BE(ON)_{Q_1}}}{R_b} \right) \quad (\text{Ec. 3.8})$$

La intensidad de corriente de cortocircuito está determinada por R_{sc} y el voltaje base emisor de Q_2 .

$$R_{sc} = \frac{V_{BE(ON)_{Q_2}}}{I_{sc}} \quad (\text{Ec. 3.9})$$

3.6 Interruptor Electrónico

Una aplicación básica de los transistores es ser usados como interruptores electrónicos para el control de corriente en una carga externa. Hay dos formas de conectar la carga, como se muestra en la figura 3.5. Cuando la carga es colocada en derivación con el transistor, una resistencia R_c es conectada en serie entre la red paralela y la fuente de voltaje. Cuando el interruptor esta en corte (OFF), la corriente fluye a través de la carga y la resistencia R en serie. Cuando el transistor esta saturado (ON), la carga es cortocircuitada y la corriente esta dirigida desde la carga a través del transistor.

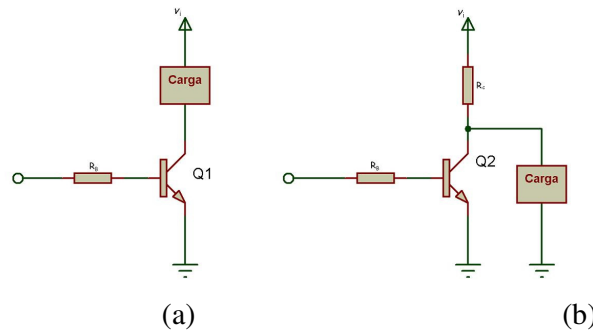


Figura 3.5.a “Carga en serie con el transistor”, 3.6.b “Carga en derivación con el transistor”

Cuando una carga externa esta en serie con el transistor, la corriente a través de la carga es la misma que la corriente de colector I_c .

Para entender mejor este funcionamiento del transistor TBJ como interruptor se debe considerar las características $I_c - V_{CE}$. Cuando un transistor se usa como interruptor electrónico trabaja solamente dentro de la región de saturación (ON) y de corte (OFF).

En el interruptor en derivación, el voltaje colector-emisor es:

$$V_{CE} = V_i - I_c R_c \quad (\text{Ec. 3.10})$$

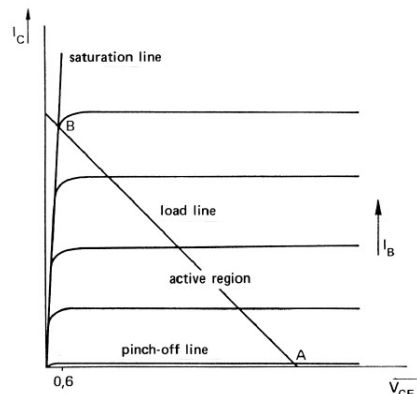
La ecuación 3.10 representa la recta de carga de la figura 3.6. Y es la corriente de base quien controla el interruptor. El transistor entra en la región de corte cuando $I_B = 0$, si la unión B-E esta en circuito abierto entonces $I_B = 0$. Con $I_B = 0$ no habrá I_c , con esto el voltaje a través de R_c será esencialmente cero, situando al transistor en la zona de corte. Entonces el V_{CE} será igual al voltaje de alimentación v_i .

$$v_i - V_{R_c} - V_{CE} = 0 \quad (\text{Ec. 3.11})$$

$$V_{CE} = v_i - V_{R_c} \quad (\text{Ec. 3.12})$$

$$V_{CE} = v_i - I_c R_c \quad (\text{Ec. 3.13})$$

Figura 3.6 “Transistor TBJ como interruptor, gráfica de $V_{CE} - I_c$ ”



Cuando la unión B-E es polarizada en directa fluye corriente en la base. Si el transistor esta en la región activa el resultado será la corriente de colector de la ecuación 3.6, si I_B crece, el valor de I_C crecerá hasta que sea igual al valor de saturación $I_{C(sat)}$. Al aplicar incrementos en I_B no producirá incrementos en I_C . Podemos usar el hecho que $V_{CE} \approx 0$ en la región de saturación para encontrar el valor de $I_{C(sat)}$. Como $V_{CE} \approx 0$, la fuente de voltaje V_i caerá a través de R_C así:

$$V_{CE} = v_i - I_C R_C \quad (\text{Ec. 3.14})$$

$$0 \approx v_i - I_C R_C \quad (\text{Ec. 3.15})$$

$$-I_C R_C \approx -v_i \quad (\text{Ec. 3.16})$$

$$I_{C(sat)} \approx \frac{v_i}{R_C} \quad (\text{Ec. 3.17})$$

Para un transistor dado, habrá un mínimo valor de corriente de base necesitado para saturar a la corriente de colector, este valor es llamado $I_{B(sat)}$. Cuando la corriente de base I_B que fluye es mayor que $I_{B(sat)}$, el transistor esta saturado con $I_C = I_{C(sat)}$. De otro forma el transistor esta en la región activa con $I_C = \beta I_B$.

3.7 Conexión a la carga

Si la carga externa requiere corrientes muy altas, estas pueden ser suministradas por un transistor que es controlado por una corriente-limite de base y una fuente de voltaje, es posible conectar transistores en como se muestra en la figura 3.7.

La unión base-emisor de Q2 es la carga derivativa de Q1. La corriente de base de Q2 depende de la fuente de voltaje V_i , el resistor R_C y $V_{BE(ON)Q2}$.

$$I_{B(Q2)} = \frac{v_i - V_{BE(ON)Q2}}{R_C} \quad (\text{Ec. 3.18})$$

Siempre que Q1 esta saturado la corriente de base de Q2 es prácticamente cero, por lo tanto Q2 esta en corte, y cuando Q1 esta en corte hay una corriente de base $I_{B(Q2)}$ para Q2. La corriente a través de la carga es igual que la corriente de colector $I_{C(Q2)}$. Si se escoge el resistor R_C de tal forma que la corriente de base de Q2 cause saturación, la corriente en la carga será:

$$I_{C(Q2)} = \beta \cdot \frac{v_i - V_{BE(ON)Q2}}{R_C} \quad (\text{Ec. 3.19})$$

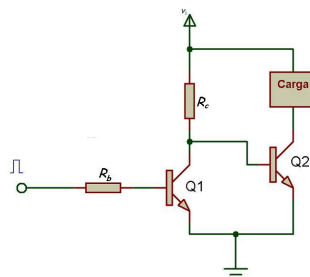


Figura 3.7 "Conexión de transistores en cascada"

Capítulo 4

Interfase Humana

Para poder interactuar con cualquier máquina con la cual se trabaje se necesita una interfase, a través de la cual el usuario puede saber cual es el status actual de la máquina recibiendo información del sistema, luego, en base a los deseos del usuario se interactúa por medio de la interfase para crear un cambio en el sistema.

El sistema de electroporación necesita leer señales, generar salidas e interactuar con el usuario haciendo posible la entrega de los pulsos a la célula. En este capítulo se verán algunos aspectos de la interfase con el sistema.

4.1 Teclado Matricial.

Los teclados matriciales son realmente una extensión del concepto de botón, son un simple arreglo de botones conectados en filas y columnas, figura "3.1", de modo que se pueden leer un gran número de botones de entrada con el número mínimo de terminales requeridos por el microcontrolador.

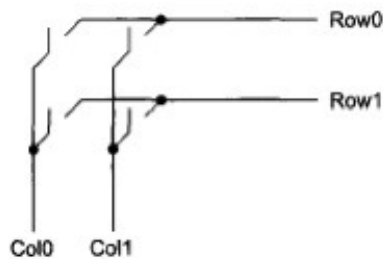


Figura 3.1 "Conexiones del Teclado"

Un teclado matricial 4×4 solamente ocupa 4 líneas de un puerto para las filas y otras 4 líneas para las columnas, de este modo se pueden leer 16 teclas utilizando solamente 8 líneas de un puerto del microcontrolador. Si se asume que todas las columnas y filas inicialmente están en alto (1 lógico), la activación de un botón se puede detectar al enviar por las líneas de salida (renglones) solo un (0 lógico) por vez y sondear cada columna en busca de un cero, si ninguna columna está en bajo entonces se rota el 0 de las filas secuencialmente de tal manera que solamente un cero se encuentre entre los renglones del teclado cuando se realiza las lecturas de las líneas de entrada (columnas). Cuando el 0 llegue a la fila más significativa del teclado, debe reingresar en la próxima ocasión por la menos significativa, reiniciando la exploración del teclado.

El puerto B del microcontrolador viene preparado especialmente para el control de un teclado matricial 4x4. Para tener siempre un valor de voltaje alto (lógico) en los renglones del teclado (parte alta del puerto B del μC) es necesario conectar los resistores de jalón figura "3.2", sin embargo el puerto B cuenta con resistores de jalón integradas, de ese modo es posible trabajar con un teclado matricial sin necesidad de ningún componente externo.

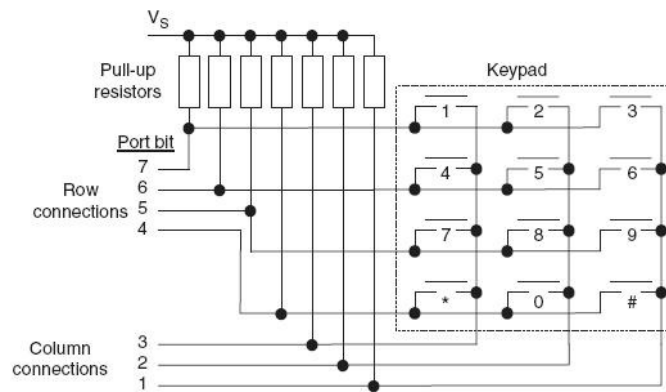


Figura 3.2 "Teclado, con resistores de Jalón"

Los resistores de jalón del puerto B se habilitan poniendo en 0 el bit RBPU del registro INTCON2. Al método expuesto para detectar la activación de una tecla en un teclado matricial se le conoce como muestreo secuencial. Existen otros, sin embargo este es tal vez el más sencillo.

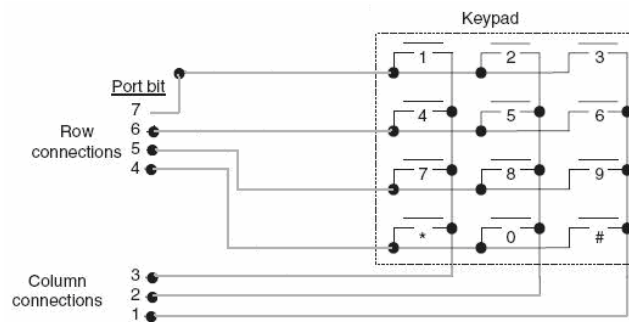


Figura 3.3 "Teclado, conectado al Puerto B resistores de jalón internas"

El programa principal es muy sencillo, simplemente configura la parte baja del puerto B como entradas y la parte alta del puerto B como salidas. Además de configurar el puerto también se habilitan los resistores de jalón internas con la línea bitclr (INTCON2,RBPU). Después de la configuración se llama a la rutina get_key () encargada de escanear el teclado y regresar el valor de la tecla pulsada a una variable para su posterior manejo. La rutina get_key () escanea el teclado matricial utilizando el método descrito.

```
void INI_keypad (void)
{
    bitclr (INTCON2,RBPU); //Se activan las resistencias de Pull-Up del puerto B
    TRISB=0X0F; // (B0:B3) -->Entradas (B4:B7) --> Salidas
    PORTB=0XF0; //Pone en alto las salidas.
}
```

Primero se carga un 0 a la primer fila, después se checa columna a columna si hay un 0, en caso de que no se detecte ningún 0, el 0 de las filas se recorre y se vuelven a revisar las columnas. Si no se detectó ningún 0 significa que no se presionó ninguna tecla y la rutina vuelve a comenzar. Si se detectó un 0 significa que se presionó una tecla y en ese momento se genera un retardo de tiempo en la lectura del teclado, de tal manera que se ignoren los contactos subsiguientes a los rebotes causados por la activación de la tecla. Después de eliminar los rebotes se decodifica la tecla pulsada utilizando una tabla de equivalencias y saber así, qué tecla fue pulsada.

```
char get_key (void)
{
    char renglon, columna, key;    //Variables para guardar la informacion de la Tecla
    char tecla;

    renglon=0xEF;
    PORTB=renglon;    //Inicio del testeo de las teclas por renglon
    while ((B0&&B1&&B2&&B3)!=0)
    {
        renglon<<=1;    //Pasa a revisar todas los renglones
        if (renglon == 0xF0)
        {
            key=0xFF;    //Devuelve que ninguna tecla se apreto
            return key;
        }
        else
        |
        PORTB=renglon;
    }

    Delay10KTCYx (20);    //Debounce de 20 ms
    columna=PORTB;    //Guarda en columna el contenido de todo el puerto.
    renglon&=0xF0;    //Se guarda en renglon el nibble alto de tecla.
    columna&=0x0F;    //Se guarda en columna el nibble bajo de tecla.
    tecla=renglon|columna;
}
```

La función get_key sólo retornará cuando un tecla haya sido oprimida. Esta rutina no permite los rebotes dentro de las teclas. El código espera a que el botón pulsado sea liberado y hace un retardo de tiempo de 20 ms para eliminar los posibles rebotes que genera la tecla al ser liberada.

```
void debounce (void)    //Pregunta si se sigue presionando la tecla.
{
    while ((B0&&B1&&B2&&B3)==0)
    { }
    Delay10KTCYx (20);    //Debounce de 20 ms
}
```

Este método es ideal para aplicaciones que no tienen interrupciones y no se tiene necesidad de alguna aplicación mientras que la tecla es pulsada. Para resolver este problema se puede localizar la rutina get_key dentro una rutina de interrupción que se genera cada cierto tiempo.

4.2 Display de Cristal Líquido o LCD

Antes de mostrar la forma de conectar estos displays con el microcontrolador, se hará un pequeño recuento de las principales características que ellos tienen, las cuales servirán para entender mejor los programas y los diagramas que se muestran más adelante.



Figura 3.4 "Display LCD 2x16"

El controlador de LCD más común es el Hitachi 44780, el cual provee una conexión entre el μC y el LCD muy fácil de usar, poseen diferentes presentaciones, por ejemplo (2 líneas por 16 caracteres), 2x20, 4x16, 4x20, 4x40 etc, son de bajo costo y se consiguen fácilmente en el comercio. La forma de utilizarlos y sus interfaces son similares, por eso, los conceptos que se verán pueden ser empleados en cualquiera de ellos. Para el desarrollo del electroporador, se trabajó con un display de 4x16, ya que su tamaño es suficiente para la aplicación de despliegue de parámetros. La explicación del funcionamiento será basado en un LCD 2x16.

Número de Terminal	Símbolo	Función
1	Vss	GND
2	Vdd	+3V ó +5V
3	Vo	Control de Ajuste
4	RS	H/L Registro de Selección
5	R/W	H/L Lectura/Escritura
6	E	Habilitación
7	DB0	H/L Línea de Datos
8	DB1	H/L Línea de Datos
9	DB2	H/L Línea de Datos
10	DB3	H/L Línea de Datos
11	DB4	H/L Línea de Datos
12	DB5	H/L Línea de Datos
13	DB6	H/L Línea de Datos
14	DB7	H/L Línea de Datos
15	A/Vee	Anodo
16	K	Catodo

Tabla 3.1 "Configuración de terminales del módulo LCD HD44780"

La tabla 3.1 muestra la configuración de terminales que se encuentran comúnmente en un LCD, aunque su ubicación cambia, conforme al LCD que se este utilizando, por ese motivo, es recomendable ver las hojas de datos del fabricante, aunque la mayoría de los LCD conservan las mismas funciones. Algunos LCD tienen luz posterior o “backlight”, para mejorar su visualización, ésta se maneja a través de dos terminales que normalmente se conectan a +5V y tierra por medio de una resistencia de 10 Ω para alimentar al positivo del backlight.

Las terminales de conexión de estos módulos incluyen un bus de datos de 8 bits, una terminal de habilitación (Enable), una Terminal de selección, que indica que el dato es una instrucción o un caracter del mensaje (Register Select) y una Terminal que indica si se va a escribir o leer en el módulo LCD (Read/Write).

Según la operación que se desee realizar sobre el LCD, las Terminales de control E, RS y R/W deben tener un estado determinado. Además, debe tener en el bus de datos un código que indique un caracter para mostrar en la pantalla o una instrucción de control. El módulo LCD responde a un conjunto especial de instrucciones, estas deben ser enviadas por el microcontrolador o sistema de control al display, según la operación que se requiera. En las tablas 3.2.a y 3.2.b se muestra el set de instrucciones y los bits de comandos respectivamente del LCD.

Instruction	Code										Description	Execution time
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear display	0	0	0	0	0	0	0	0	0	1	Clears display and returns cursor to the home position (address 0).	1.64ms
Cursor home	0	0	0	0	0	0	0	0	1	*	Returns cursor to home position (address 0). Also returns display being shifted to the original position. DDRAM contents remains unchanged.	1.64ms
Entry mode set	0	0	0	0	0	0	0	1	VD	S	Sets cursor move direction (VD), specifies to shift the display (S). These operations are performed during data read/write.	40us
Display On/Off control	0	0	0	0	0	0	1	D	C	B	Sets On/Off of all display (D), cursor On/Off (C) and blink of cursor position character (B).	40us
Cursor/display shift	0	0	0	0	0	1	S/C	R/L	*	*	Sets cursor-move or display-shift (S/C), shift direction (R/L). DDRAM contents remains unchanged.	40us

Instruction	Code										Description	Execution time
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Function set	0	0	0	0	1	DL	N	F	*	*	Sets interface data length (DL), number of display line (N) and character font(F).	40us
Set CGRAM address	0	0	0	1	CGRAM address						Sets the CGRAM address. CGRAM data is sent and received after this setting.	40us
Set DDRAM address	0	0	1	DDRAM address						Sets the DDRAM address. DDRAM data is sent and received after this setting.	40us	
Read busy-flag and address counter	0	1	BF	CGRAM / DDRAM address						Reads Busy-flag (BF) indicating internal operation is being performed and reads CGRAM or DDRAM address counter contents (depending on previous instruction).	0us	
Write to CGRAM or DDRAM	1	0	write data						Writes data to CGRAM or DDRAM.	40us		
Read from CGRAM or DDRAM	1	1	read data						Reads data from CGRAM or DDRAM.	40us		

Tabla 3.2.a "Set de instrucciones del HD44789"

Bit name	Setting / Status	
I/D	0 = Decrement cursor position	1 = Increment cursor position
S	0 = No display shift	1 = Display shift
D	0 = Display off	1 = Display on
C	0 = Cursor off	1 = Cursor on
B	0 = Cursor blink off	1 = Cursor blink on
S/C	0 = Move cursor	1 = Shift display
R/L	0 = Shift left	1 = Shift right
DL	0 = 4-bit interface	1 = 8-bit interface
N	0 = 1/8 or 1/11 Duty (1 line)	1 = 1/16 Duty (2 lines)
F	0 = 5x7 dots	1 = 5x10 dots
BF	0 = Can accept instruction	1 = Internal operation in progress

Figura 3.2.b "Bit de comandos del HD44780"

La interfase entre el microcontrolador y el LCD se puede realizar con el bus de datos a 4 u 8 bits. Las señales de control trabajan de la misma forma en cualquiera de los dos casos, la diferencia se establece en el momento de iniciar el sistema, ya que existe una instrucción que permite seleccionar dicha configuración. Estas conexiones se explican más adelante de forma detallada.

Los caracteres que se envían al display se almacenan en la memoria RAM del módulo. Existen posiciones de memoria RAM, cuyos datos son visibles en la pantalla y otras que no lo son, estas últimas se pueden utilizar para guardar caracteres que luego se desplazan hacia la parte visible. En la figura "3.5" se muestran las direcciones de memoria visibles y no visibles, que conforman las dos líneas de caracteres del módulo.

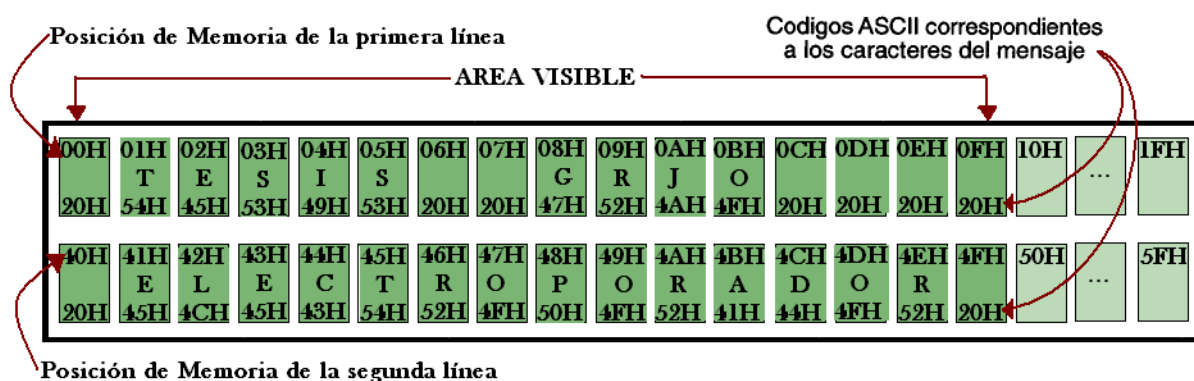


Figura 3.5 "Mapa de memoria del HD44780"

Es importante señalar que sólo se pueden mostrar caracteres ASCII de 7 bits, por lo tanto, algunos caracteres especiales no se pueden ver. Por otra parte, se tiene la opción de crear caracteres especiales (creados por el programador), y almacenarlos en la memoria RAM que posee el LCD como se muestra en la figura 3.6.

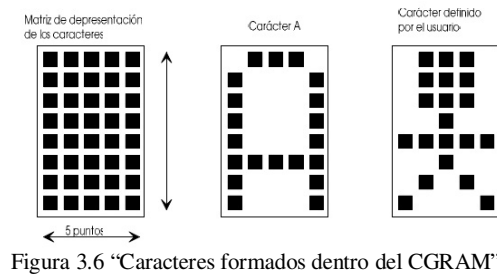


Figura 3.6 “Caracteres formados dentro del CGRAM”

4.2.1. Interfase con microcontrolador a LCD

La conexión entre el PIC y el módulo LCD se realizará por medio de un bus de datos de 4 bits, donde se utilizarán las 4 Terminales de mayor peso del puerto D (RD₄-RD₇) del microcontrolador. Las señales de control (R/W, RS y E), se generarán con las Terminales de menor peso del puerto D (RD₁, RD₂ y RD₃). La figura 3.7 muestra esta configuración.

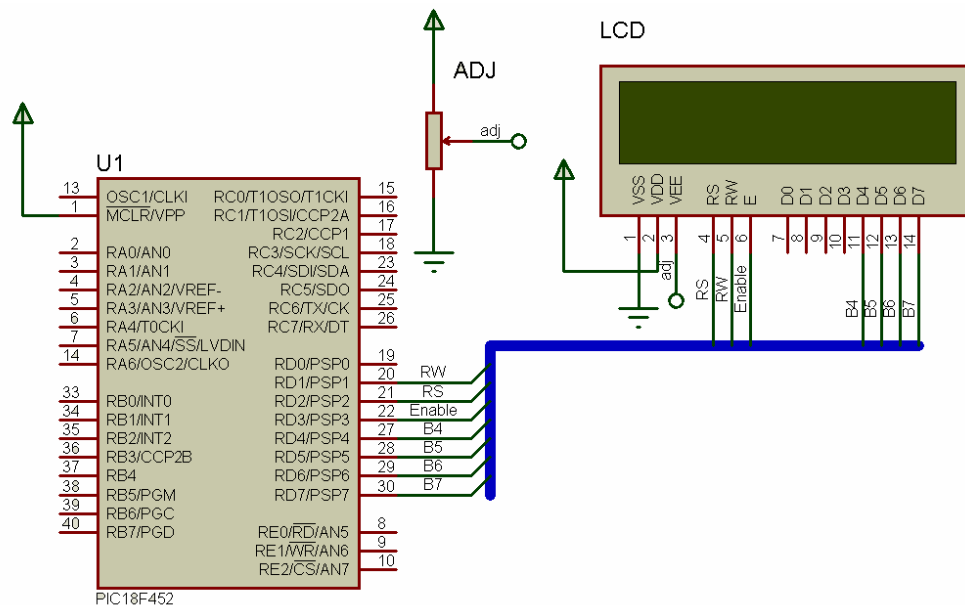


Figura 3.7 “Conexión a 4 bits del LCD con el microcontrolador”

El software que se implanta en el microcontrolador, se encarga de mostrar mensajes en un LCD de 4x16. Dichos mensajes ocuparan las 4 líneas de la pantalla y permanecen fijos, o cambiaran según sea necesario dentro del menú contextual en el cual se encuentre el usuario.

Los pasos más importantes para la inicialización del módulo LCD son:

- ✓ Se programan los puertos según el circuito.
- ✓ Se debe inicializar el módulo LCD. El primer dato que se envía (2C h) le indica al módulo que la comunicación se va a realizar a 4 bits, se emplearán todas las líneas de caracteres del LCD y el formato de la celda del LCD. El dato se puede deducir con la lista de las instrucciones que se muestra en la figura "3.5.a". La función de INI_LCD, se encarga de generar todas las señales de inicialización y los tiempos necesarios para que exista una correcta comunicación.

```
void INI_LCD(void)
{
    TRISD= 0;           //Se elige como salida el puerto D
    PORTD=0;           //Pone a cero el puerto D
    bitClr(PORTD,RS);  //Linea RS en bajo (PortD, bit2)
    Delay10KTCYx(100); //Retardo de 100ms
    PUERTO_LCD |= BIT4 | BIT5; //D7-D4 = 0011
    PUERTO_LCD &= ~BIT6 & ~BIT7;
    _E();              //Activa Enable
    Delay10KTCYx(10); //Retardo de 10ms
    _E();
    Delay10KTCYx(10);
    _E();
    Delay10KTCYx(10);
    PUERTO_LCD &= ~BIT4;
    _E();              //toggle E
    LCD_COM(LINES_5X7); //4 BITS , 2 LINEAS, 5X7
    LCD_COM(DISP_ON);
    LCD_COM(CLR_DISP); //ESTE COMANDO TARDA MAS QUE LOS DEMAS
    temp=PUERTO_LCD;
    Delay10KTCYx(100); //Delay 100ms
}
```

Las funciones LCD_COM y LCD_DAT que se muestran a continuación envían primero el nibble alto (4 bits de mayor peso) del dato y luego el nibble bajo (4 bits de menor peso). En el momento de enviar cada uno de los datos de 4 bits, las señales de control deben comportarse de la misma forma como si fuera un dato completo. Por eso en la rutina se ve que la señal RS conserva el nivel lógico adecuado y la señal E genera los dos pulsos que se requieren (el primero para el nibble alto y el segundo para el bajo).


```

void LCD_DAT ( char d)
{
    Delay1KTCYx (5);           //Retardo de 0.5ms 5000 ciclos@40 MHZ
    temp = d & 0xf0;           //ENVIA NIBBLE SUPERIOR
    PUERTO_LCD &= 0x0f;
    PUERTO_LCD |= temp;
    bitset(PORTD,RS);         //ENVIA A REGISTRO DE DATOS
    _E();                     //Activa Enable
    temp = d & 0x0f;
    temp = temp << 4;         //ENVIA NIBBLE INFERIOR
    PUERTO_LCD &= 0x0f;
    PUERTO_LCD |= temp;
    bitset(PORTD,RS);         //ENVIA A REGISTRO DE DATOS
    _E();                     //TOGGLE E
}

```

```

void LCD_COM ( char c)
{
    Delay10KTCYx (10);        //10ms , 100000CICLOS A 40 Mhz.
    temp = c & 0xf0;          //get upper nibble
    PUERTO_LCD &= 0x0f;
    PUERTO_LCD |= temp;      //send CMD to LCD
    bitclr(PORTD,RS);        //set LCD to CMD mode
    _E();                    //toggle E for LCD
    temp = c & 0x0f;
    temp <<= 4;              //get down nibble
    PUERTO_LCD &= 0x0f;
    PUERTO_LCD |= temp;
    bitclr(PORTD,RS);        //set LCD to CMD mode
    _E();                    //toggle E for LCD
}

```

Capítulo 5

Diseño e implementación

5.1 Diseño de la Unidad de Control

Sobre la base de la estructura general que se muestra en la figura 5.1 se diseña cada etapa que la compone.

El estricto control de tiempos y frecuencia de los pulsos es controlado por un microcontrolador PIC18F452 el cual es el corazón de la unidad de control, además de ser una magnífica solución para ahorrar espacio y costo en esta aplicación de instrumentación biológica. Las tareas del microcontrolador: son la obtención de los parámetros del tratamiento, leer el voltaje indicado por el usuario, generar los pulsos necesarios para el tratamiento y al finalizar el proceso emitirá una señal auditiva y mostrará los parámetros de tratamiento realizados.

Las especificaciones de diseño de los pulsos son las siguientes:

- ✓ Ancho de pulso desde 1 ms hasta 999 ms con resolución de 1 ms.
- ✓ Separación entre pulsos desde 50ms hasta 950 ms con resolución de 50 ms.
- ✓ Número de pulsos de 1 hasta 99 pulsos.

Para esto y de acuerdo a lo visto en los capítulos anteriores se desarrolló el hardware necesario y cuyo esquema eléctrico se muestra en la figura 5.1.

El PIC18F452 de 10 MIPS (Millones de Instrucciones Por Segundo) con 40 Terminales y tecnología CMOS, dispone de memoria flash y un convertor A/D de 10 bits, espacio para código de programa de 16kbytes, 32 kbytes de RAM, 256 bytes de memoria EEPROM, comunicación SPI™, I²C™ y AUSART con un consumo mínimo de corriente, funciones "power-on reset" (POR) y watchdog programable y un amplio número de herramientas de desarrollo y notas de aplicación. Todas estas características lo convierten en un sistema ideal para la instrumentación, monitorización y adquisición de datos biológicos.

Los conectores K1 y K7 permiten la conexión al display LCD la cual está conectada al puerto D del microcontrolador en modo de 4 bits, lo que requiere que los datos sean enviados dos veces usando el separador superior del puerto.

De igual forma los conectores K2 y K8 ofrecen el acceso al puerto B para la conexión de un teclado matricial de 4x4, se utilizan resistores de jalón internas del microcontrolador para establecer el nivel lógico de las teclas.

El microcontrolador puede ser programado en la propia tarjeta utilizando el conector K3 ICSP.

El conector K4 es la unión entre la unidad de control y la unidad de energía, el cual recibe el valor del voltaje censado y entrega los pulsos del tratamiento introducido.

Para la puesta en funcionamiento del bus I²C, que utiliza el reloj de tiempo real (DS1307) y la memoria EEPROM (24C256F) sólo son necesarios dos resistores de polarización. El valor de los resistores varía en función de la velocidad del bus que se desee alcanzar. En la tarjeta, el valor del resistores colocadas es de 4,7 k Ω , lo que permite alcanzar una velocidad teórica de transmisión de 100 kbits/s.

Con el fin de mejorar la eficiencia de la generación de pulso y facilitar así su uso de la temporización se optó por utilizar dos diferentes dominios de reloj el cual puede ser habilitado por programación, uno reservado sólo para la generación de los pulsos, para la cual se usa un reloj de 32.768 kHz y el segundo reloj para controlar la frecuencia de trabajo del microcontrolador con un reloj de 10 MHz.

El conector K6 esta conectado al puerto A que permite, además de las funciones de entradas/salidas digitales, definir convertidores ADC en caso de ser necesarios.

La programación del firmware del microcontrolador se hace con la ayuda del entorno de programación de desarrollo MPLAB de Microchip. Para la gama de microcontroladores de 18Fxxx, la programación en C se hace a través del compilador C18.

La programación de la tarjeta se efectúa con el programador PICKIT 2, sin embargo, también puede usarse otro programador compatible. Si todo transcurre correctamente, el programador sólo es necesario una sola vez, a partir de lo cual, el programa de arranque será el encargado de hacer la carga del programa al microprocesador.

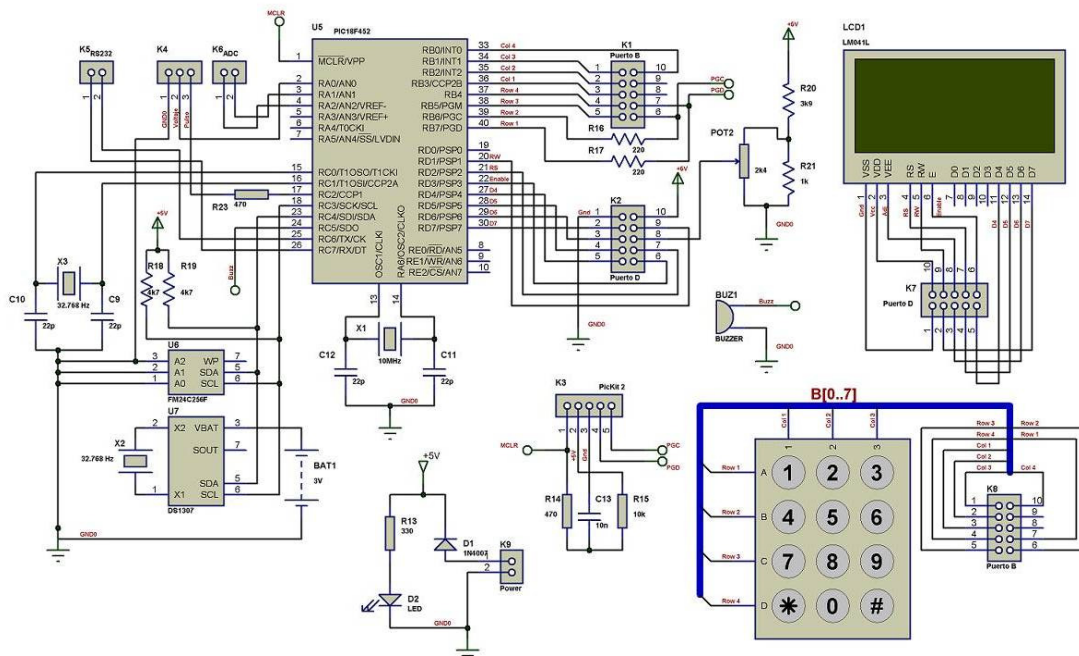


Figura 5.1 “Esquema eléctrico de la tarjeta base de la Unidad de Control.”

En las figuras 5.2 (a) y 5.2 (b) se muestra las caras inferior y superior de la tarjeta de control dedicada al sistema de electroporación y en la figura 5.3 se observa la tarjeta ya ensamblada lista para ser utilizada en el sistema.

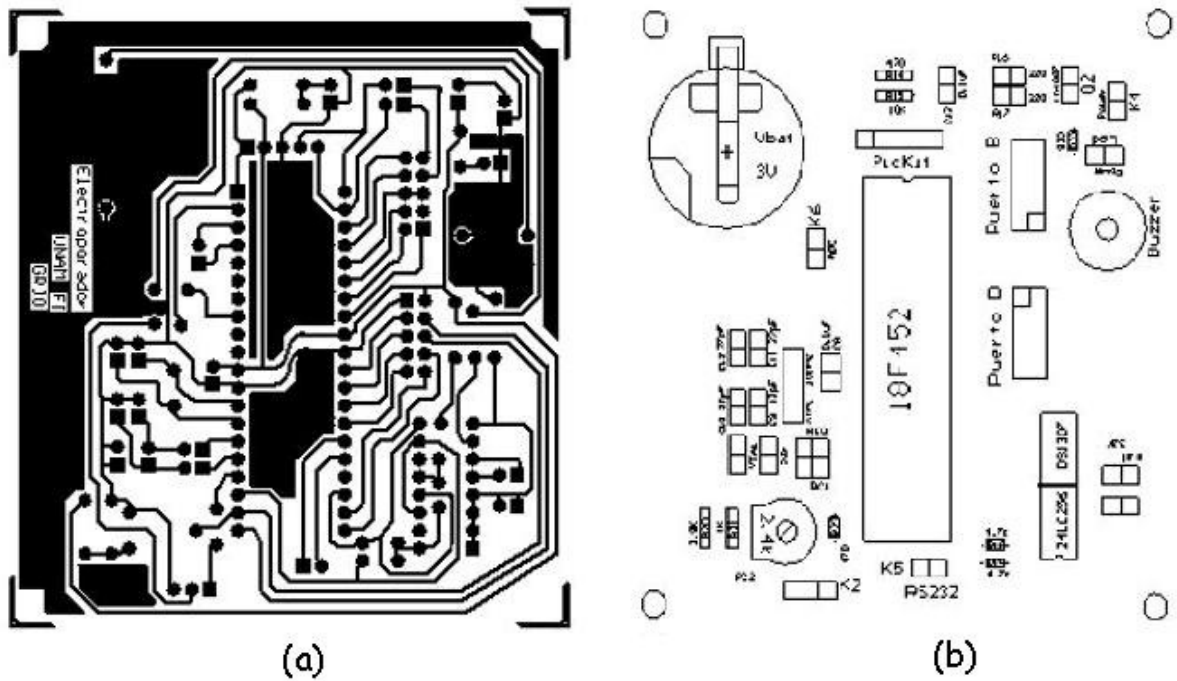


Figura 5.2 (a) "PCB de la placa de Control" (b) "Cara de componentes de la placa de Control"

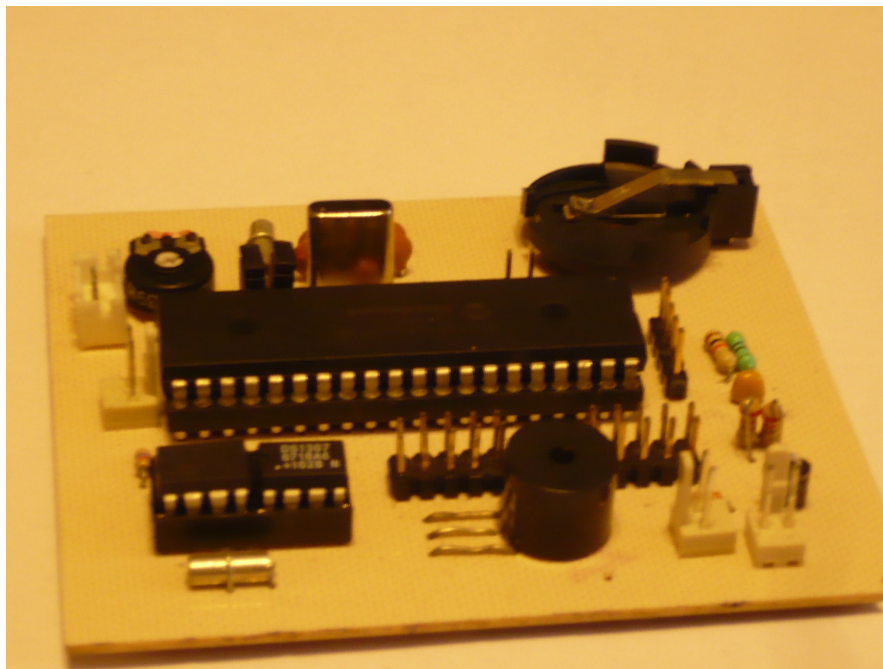


Figura 5.3 "Vista de la placa de control"

5.2 Diseño de la Unidad de Potencia

Muchas cuestiones se han planteado durante el proceso de diseño, principalmente por el nivel de energía que participa y debido a la necesidad de seguridad asociados con un instrumento de laboratorio. Gracias a la naturaleza de la electroporación reversible, la generación de pulsos cuadrados a 80 V es fácil de lograr, no así como lo requiere el diseño de uno de muy alto voltaje para un electroporador irreversible. El enfoque utilizado para el diseño de la unidad de potencia es a través de una fuente lineal regulada, cuyo componente principal es el CI TL783, por razones de seguridad en caso de falla se utiliza un transistor 2N2907 que limita la corriente máxima. Dado que las especificaciones permiten una corriente máxima de 0.7 A, se utilizará para el control de la corriente entregada a cada pulso un transistor de paso TIP 32C.

El sistema controla la generación de pulsos por medio de un dispositivo optoacoplador MOC 3031 y un transistor driver PN2222. Ambos dispositivos tienen diferentes propósitos, el primero está directamente manejado por el microprocesador acoplado el pulso TTL a la unidad de potencia y el segundo es utilizado para manejar un transistor MOSFET de potencia IRF110 que conmuta de estado de acuerdo a las ordenes del microcontrolador.

Para el diseño se consideró el uso de dispositivos de aislamiento para los circuitos que manejan altos voltajes; para controlar el voltaje de entrada se utilizó un transformador que aísla cualquier falla de la toma de corriente hacia la unidad de potencia. Para alimentar a la carga y a la circuitería interna del regulador, es necesario un voltaje de entrada no mayor a 84 volts.

En base que
$$V_p = \frac{V_{rms}}{\sqrt{2}} = \frac{84}{\sqrt{2}} = 59.396 \text{ Volts}$$

Así que debemos escoger un transformador a 60 V y 1 A de corriente.

A continuación se muestra el procedimiento y las ecuaciones básicas de diseño para el cálculo de una fuente práctica reguladora de voltaje y limitadora de corriente. Se supone que la fuente de potencia de ca rectificadas es la salida de un rectificador puente con una capacitancia de filtro de 2200 μ Farads; aislada de la línea por el transformador escogido.

Las condiciones de funcionamiento deseadas se eligieron así:

Condiciones de entrada

Voltaje de entrada:
$$V_i = 127\sqrt{2} \approx 180 \text{ V}$$

Variación en el voltaje de entrada: 123-127 VAC

Condiciones de Salida

Voltaje de salida $V_o = 2-80 \text{ VDC}$

Corriente de salida máxima $I_{o(max)} = 0.7 \text{ A}$

El suministro de corriente de salida será dado por un transistor serie externo de paso. Esto es para proteger al regulador de la disipación excesiva de potencia ocasionando que se dañe y donde la figura 5.8 representa su configuración.

Inicialmente el transistor debe satisfacer los siguientes requerimientos:

$$V_{CE} \geq V_{in(reg)} - V_{out(min_reg)} \quad (\text{Ec. 5.1})$$

$$I_{C(max)} \geq I_{o(max)} \quad (\text{Ec.5.2})$$

Donde $V_{in(reg)}$ es el voltaje de alimentación de entrada, $V_{out(min_reg)}$ es el voltaje de salida mínima del regulador, I_C el régimen máximo de corriente de colector e $I_{o(max)}$ la corriente de salida máxima. Se ha elegido un transistor TIP 32C como transistor de paso por que satisface estas condiciones

$$100 \geq (84-2=82) \text{ V}$$

$$3 \text{ A} \geq 0.7 \text{ A}$$

Se elige un transistor Q_L limitador de corriente, el régimen de voltaje máximo colector-emisor de este transistor V_{CEO} debe ser mayor que, $R_{SC(I_{o(max)})} > V_{BE(ON)Q2}$. El transistor 2N2907 es adecuado como dispositivo limitador de corriente porque satisface el siguiente requerimiento:

$$60 > 2.66$$

En base a la ecuación 3.8 del capítulo 3 la resistencia de sensado RSC es:

$$R_{SC} = R_2 = \frac{1.3}{0.7} = 1.857\Omega, 1\text{W}$$

Para asegurar el funcionamiento del regulador se escoge una $I_{reg(max)}=150\text{mA}$, el transistor de paso Q_1 empieza a funcionar cuando se alcanza su $V_{BE(ON)}$, y haciendo uso de las ecuaciones 3.6 y 3.7 del capítulo 3.

$$R_B = R_1 = \frac{1.8}{150\text{mA} - \frac{700\text{mA}}{25}} = 14.75 \Omega$$

$$I_{sal(max)} = 25(150\text{mA} - \frac{1.8\text{V}}{14.75\Omega}) = 0.69915 \text{ A}$$

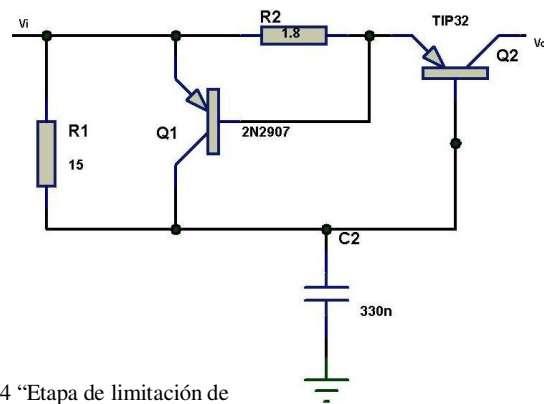


Figura 5.4 "Etapa de limitación de corriente y amplificación de corriente"

El CI TL783 es un regulador de alto voltaje ajustable de 3 Terminales de Texas Instruments, con un voltaje de salida desde (1.27-127) V. La referencia interna de voltaje (V_{ref}) es de 1.25V entre la Terminal de salida y el de ajuste, este voltaje se crea a través de R_1 y provoca una corriente constante a través de R_1 y el resistor de ajuste R_2 originando una salida de voltaje descrita por la siguiente ecuación:

$$V_o = 1.27 \left(1 + \frac{R_2}{R_1} \right) \quad (\text{Ec. 5.1})$$

De acuerdo a las condiciones de salida el voltaje máximo deseado en el sistema es de 80 volts, sustituyendo y despejando la relación de resistores de la ecuación 5.1 se tiene que:

$$\frac{R_2}{R_1} = \left(\frac{80}{1.27} - 1 \right) = 61.992$$

R2 es proporcional a R1 por un factor prácticamente de 62, para el ajuste del voltaje del regulador se propone que R2 sea un potenciómetro de alambre de 10 kΩ a 1W, debido a la potencia disipada entre sus bornes. De esta forma si R2=10 kΩ;

$$R_1 = \frac{R_2}{61.992} = \frac{10k}{61.992} = 161.311\Omega$$

No existe un valor comercial para este valor de resistencia por lo que se usará un arreglo de resistores en serie de tal forma que se alcance este valor. Entonces:

$$R_1 = R_3 + R_4 = 150 + 12 = 162 \Omega$$

De igual forma el voltaje mínimo deseado de salida es de 2 volts, así que sustituyendo y despejando la relación de resistores de la ecuación 5.1 se tiene que:

$$\frac{R_2}{R_1} = \left(\frac{2}{1.27} - 1 \right) = 0.5748$$

De esta forma el valor de $R_2 = 0.5748R_1$, R1 es el valor obtenido Dado que R1=161.311Ω

$$R_2 = 0.5748 \cdot (161.311\Omega) = 92.721\Omega$$

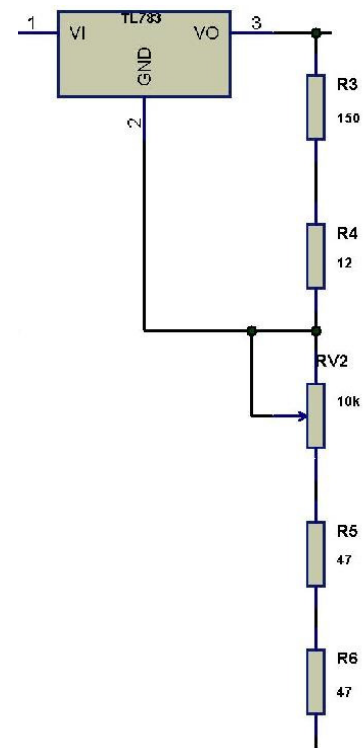


Figura 5.5 “Configuración del Regulador TL783 con un voltaje de salida de 2-80V”

De igual forma no existe un valor comercial para este valor por lo que se usara un arreglo de resistencia en serie de tal forma que alcancemos este valor, entonces:

$$R_1 = R_5 + R_6 = 47 + 47 = 94 \Omega$$

Por otra parte es necesario acondicionar la señal para que la UC lea los parámetros del voltaje, por lo que es necesaria una etapa de acondicionamiento de la señal. Para esto, se tomará una muestra del voltaje de salida a través de un divisor de voltaje, el cual reducirá el voltaje máximo de salida de 80 V al voltaje máximo permitido a la entrada del microcontrolador de 5 V.

$$V_o = \frac{R_2}{R_1 + R_2} V_i$$

$$\frac{80}{5} = \frac{R_2}{R_1 + R_2}$$

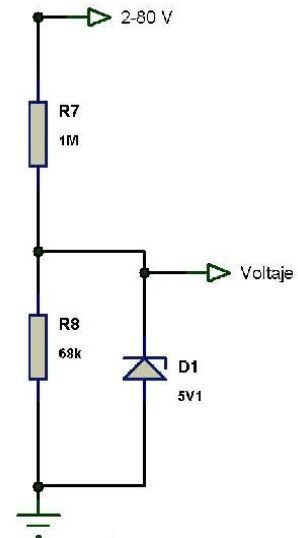
$$R_1 = R_2(16 - 1)$$

$$R_1 = R_2 \cdot 15$$

Considerando un $R_2 = 68 \text{ k}\Omega$, entonces:

$$R_1 = 68 \text{ k}\Omega \cdot 15 \approx 1 \text{ M}\Omega$$

Figura 5.6 "Adecuación del voltaje de salida hacia la UC"



En caso de un mal funcionamiento de la unidad de potencia el cual origine un voltaje excesivo a las consideraciones de diseño del divisor de voltaje, se añade un diodo zener de 5.1 V como medida de protección hacia el microcontrolador.

Con el fin de evitar la destrucción de la etapa de control como consecuencia de las sobretensiones que podrían aparecer sobre la línea de alimentación de AC, se ha decidido equipar este interfase con una alimentación de 5 V aislada (U2) y un optoacoplador (U3).

Dado que el transistor driver no demandará más de 10 mA la fuente se diseña con los siguientes datos

$$V_o = R_L I_o$$

$$R_L = \frac{V_o}{I_o} = \frac{5V}{10mA} = 500\Omega$$

Por medio de un divisor de voltaje

$$V_o = \frac{R_2 \parallel R_L}{R_1 + R_2 \parallel R_L} V_i$$

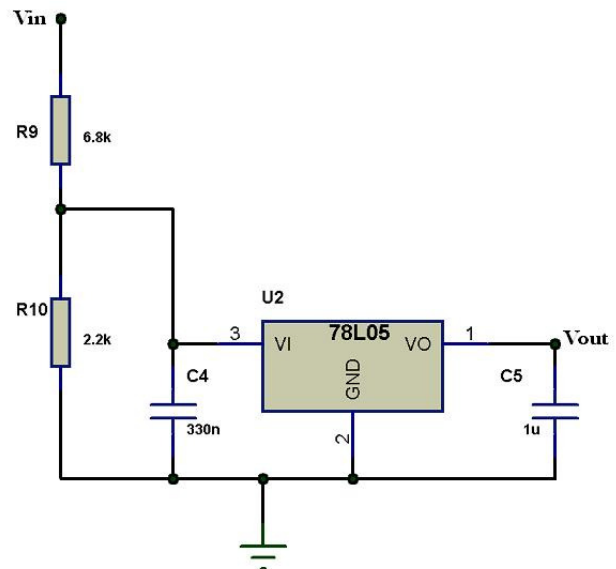


Figura 5.7 "Fuente de alimentación a 5V que alimenta la etapa de acoplación"

Despejando el denominador

$$R_1 + R_2 \parallel R_L = \frac{V_i}{V_o} R_2 \parallel R_L \quad \Rightarrow \quad R_1 = (R_2 \parallel R_L) \cdot \left(\frac{V_i}{V_o} - 1 \right)$$

$$R_1 = (R_2 \parallel 500\Omega) \cdot \left(\frac{84}{5} - 1 \right)$$

Si $R_2 = R_{10} = 2.2 \text{ k}\Omega$

$$R_1 = (2.2 \text{ k}\Omega \parallel 500\Omega) \cdot (16)$$

$$R_1 = R_9 = 6.518 \text{ k}\Omega, 1\text{W}$$

Por último el manejo de corriente al embrión será controlada por un transistor controlado por una corriente-límite de base y una fuente de voltaje. Como se explicó al inicio, un transistor driver PN2222 será directamente manejado por el microprocesador acoplado el pulso TTL a la unidad de potencia y un transistor MOSFET de potencia IRF110 conmutará el estado de la carga de acuerdo a las ordenes del microcontrolador.

Para esto, ambos transistores trabajarán en las zonas de corte y saturación que los definen como interruptores electrónicos.

Para ubicar al transistor Q_4 en la zona de saturación es necesario romper el voltaje de ruptura $V_{GS(TH)}$, haciendo uso de la ecuación 3.17 del capítulo 3.

$$R_C \approx \frac{v_i}{I_C} = \frac{4\text{V}}{2\text{mA}} = 2\text{k}\Omega$$

Como no existe valor comercial de $2 \text{ k}\Omega$ se utilizará el valor más cercano de $1.8 \text{ k}\Omega$.

Y una resistencia de polarización R_{11} de $150 \text{ k}\Omega$ como se muestra en la figura.

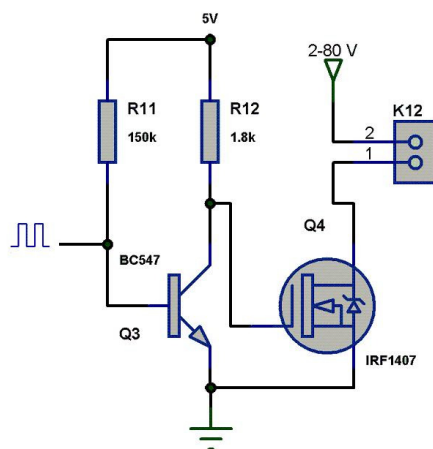


Figura 5.8 “Conexión a la carga”

A continuación en la figura 5.9 se muestra el diagrama electrónico correspondiente a la unidad de energía, así como, el PCB dedicado a la aplicación con sus vistas inferior y superior de la placa en la figura 5.10 (a), 5.10 (b) y en la figura 5.11 se muestra la tarjeta ya ensamblada lista para ser utilizada en el sistema de electroporación.

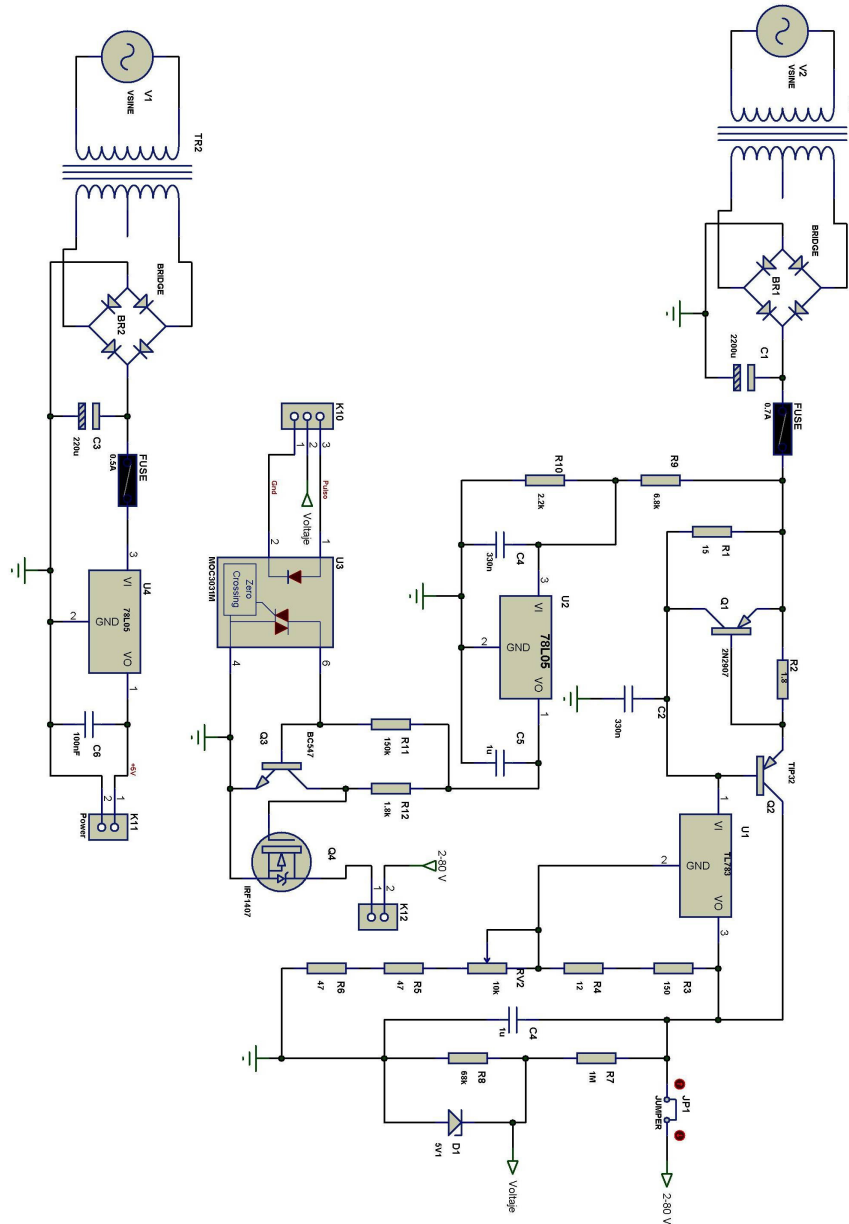
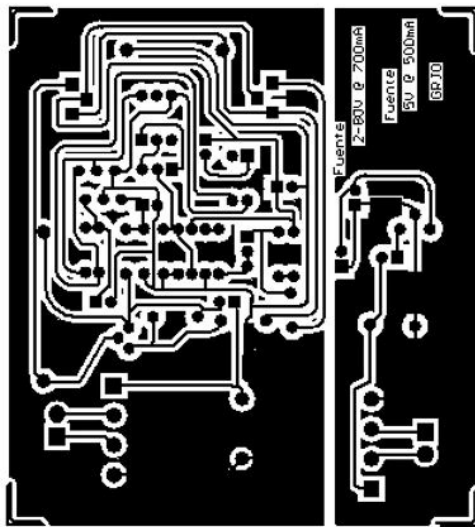
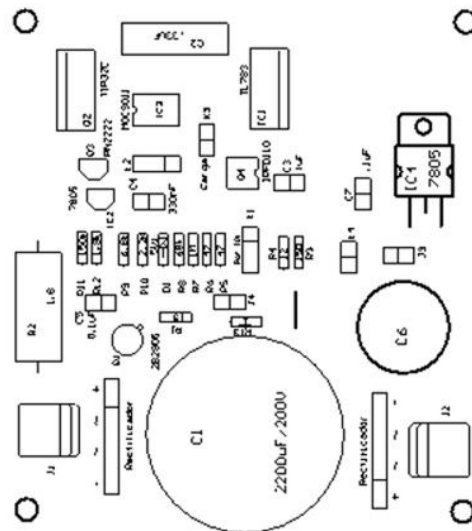


Figura 5.9”Esquema eléctrico y la placa base de la Unidad de Energía”



(a)



(b)

Figura 5.10 (a) “PCB placa de la Unidad de Energía” (b) “Cara de componentes del PCB de la UE”

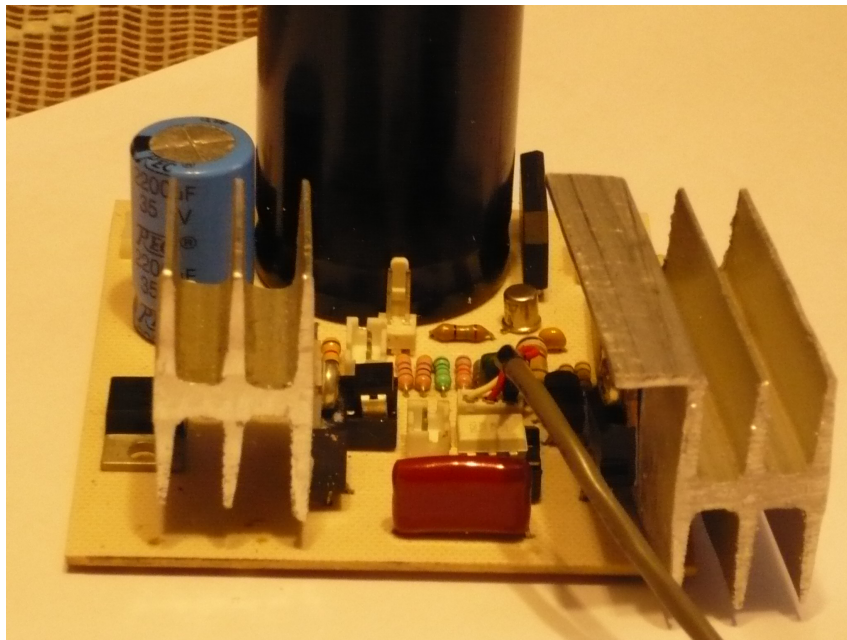


Figura 5.11 ” Vista de la unidad de potencia”

Por último las etapas de control, potencia e interfase humana son ensambladas como todo un sistema para lo cual se adecua una caja de prototipos de PROKIT como se muestra en las figuras 5.12.a y 5.12.b para ser presentado como el nuevo sistema de electroporación al Laboratorio de Medicina Gnómica y Toxicología Ambiental.



Figura 5.12.a “Sistema de electroporación ensamblado en una caja de proyectos PROKIT”

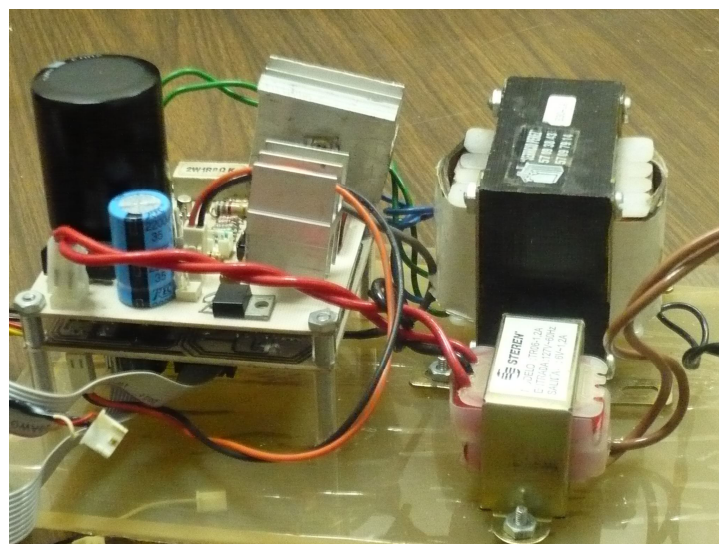


Figura 5.12.b “Sistema de electroporación ensamblado”

Capítulo 6

Arquitectura de Software

6.1 Descripción del Firmware

La programación en firme (del inglés firmware) desarrollado fue pensando para facilitar el uso del sistema y minimizar los errores que suele introducir el usuario dentro de un sistema, elimina por software las inestabilidades del equipo básico y evita el uso de metodologías de caracterización no adecuadas que algunos usuarios por su formación primera conservan como válidas.

La arquitectura principal del software de aplicación, es un menú de introducción de parámetros el cual inicia con la especificación del voltaje requerido y continua con la recepción de los parámetros de tratamiento deseados (ancho de pulso, tren de pulsos y número de pulsos), al finalizar la entrada de parámetros, el software pide la autorización al usuario para la ejecución del tratamiento, permitiendo de este modo que un tratamiento pueda ser abortado por el usuario en cualquier momento, regresando a la etapa de introducción de parámetros a la espera de nuevas aportaciones de tratamiento.

Por otro lado, una vez que los parámetros del tratamiento son aceptados, el usuario debe autorizar el tratamiento para que el sistema elabore todos los datos recibidos, despliegue los parámetros del tratamiento realizados y por último regrese a la etapa de introducción de nuevos parámetros.

De igual forma si se produce un error durante una operación ó los parámetros son incompatibles inmediatamente el software notifica al usuario del error desplegándolo en el display y generando una señal auditiva que le avisa del error.

Como se mencionó en el capítulo 5, la programación del firmware se hace en el entorno de programación de desarrollo MPLAB de microchip usando el compilador C18.

El código fuente se dividió en los siguientes módulos de acuerdo a su función:

- ✓ `electroporador.c` : Contiene el programa principal en donde se configuran los parámetros de control de electroporación, registro de eventos y datos y control global de errores.
- ✓ `electroporador.h` : Consta de todas las declaraciones de prototipos de función, declaración de los fuses del microcontrolador, declaración y asignación de variables a memoria del microcontrolador e inicializan todas las variables internas del programa.
- ✓ `LCD_PIC.c` : Modulo que especifica todas las funciones relacionadas con la escritura y borrado básico de caracteres a la pantalla LCD, además de establecer su inicialización.
- ✓ `LCD_PIC.h` : Modulo de cabecera que contiene todas las declaraciones y prototipos de función que manejan al LCD.

- ✓ teclado.c: Modulo que permite la introducción de parámetros a través de un teclado matricial.
- ✓ funciones.c: Modulo que especifica todas las funciones necesarias para el funcionamiento del programa principal "main" además de subrutinas de decisiones de control.
- ✓ boolean.h: función que define valores que pueden ser solo verdadero ó falso correspondiente a los dos estados lógicos de una variable digital (1 y 0).

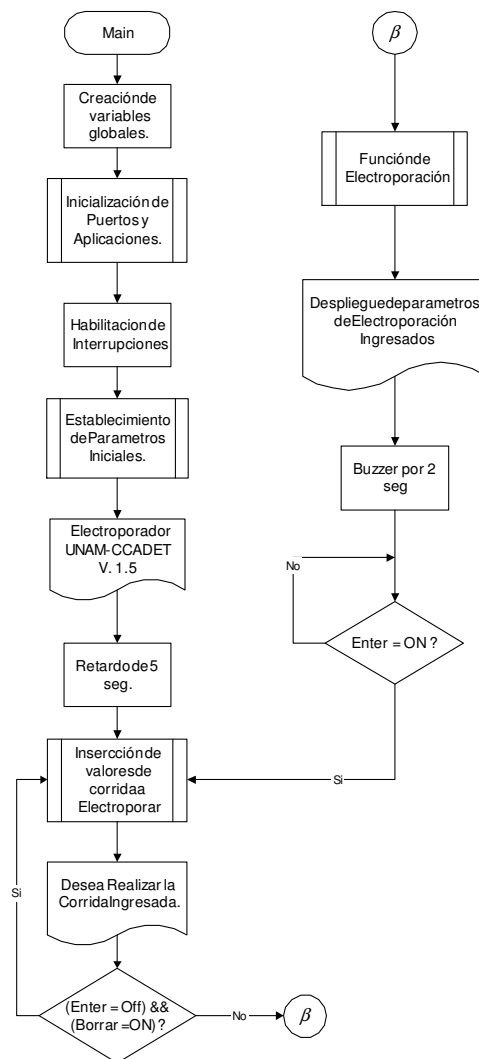
La programación en la tarjeta de control se efectúa con el programador PICKIT 2.

A continuación se muestra los diagramas de flujo que describen la arquitectura del firmware del microcontrolador.

6.2 Diagrama de bloques del Firmware

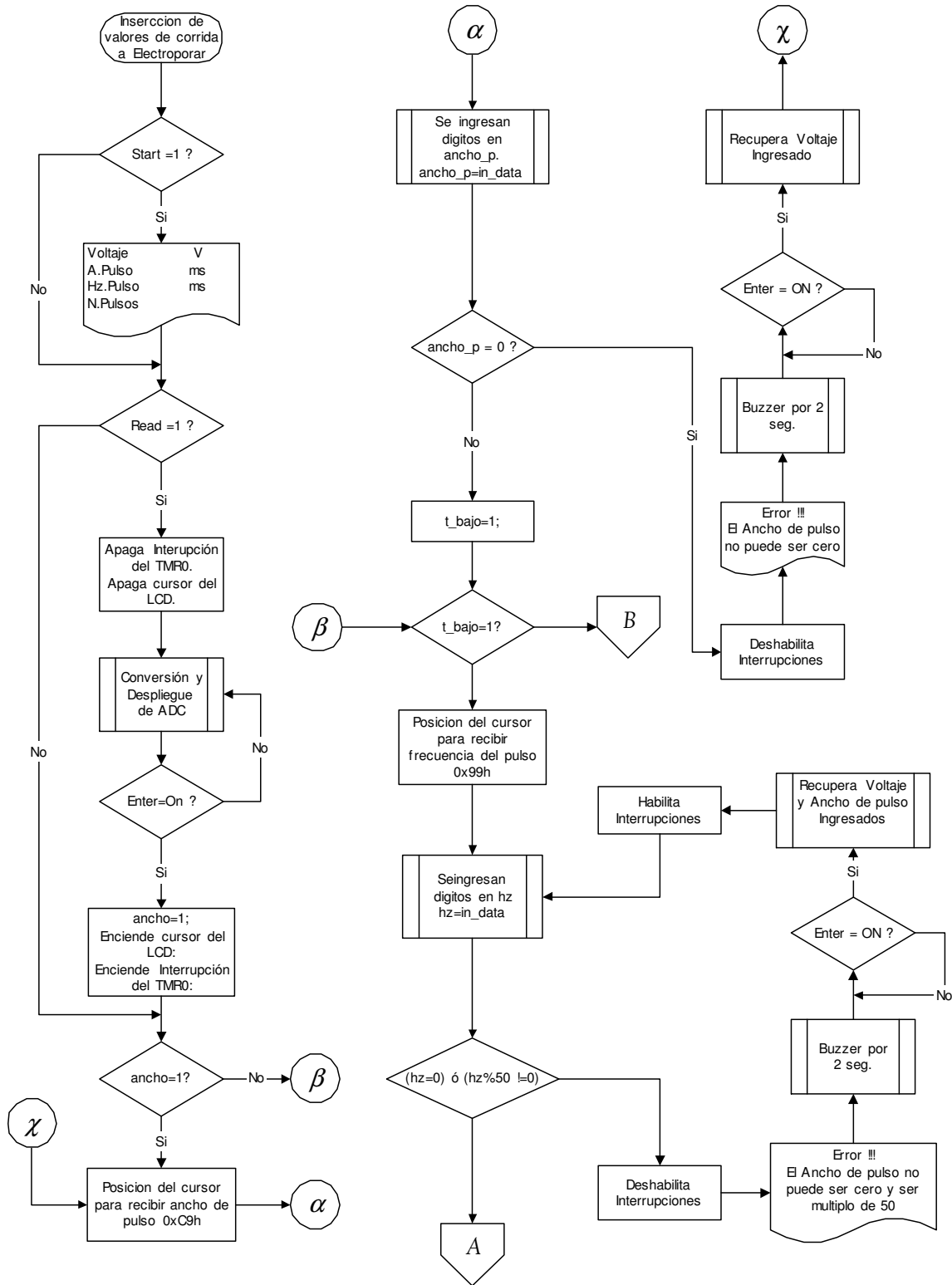
6.2.1. *Función main*

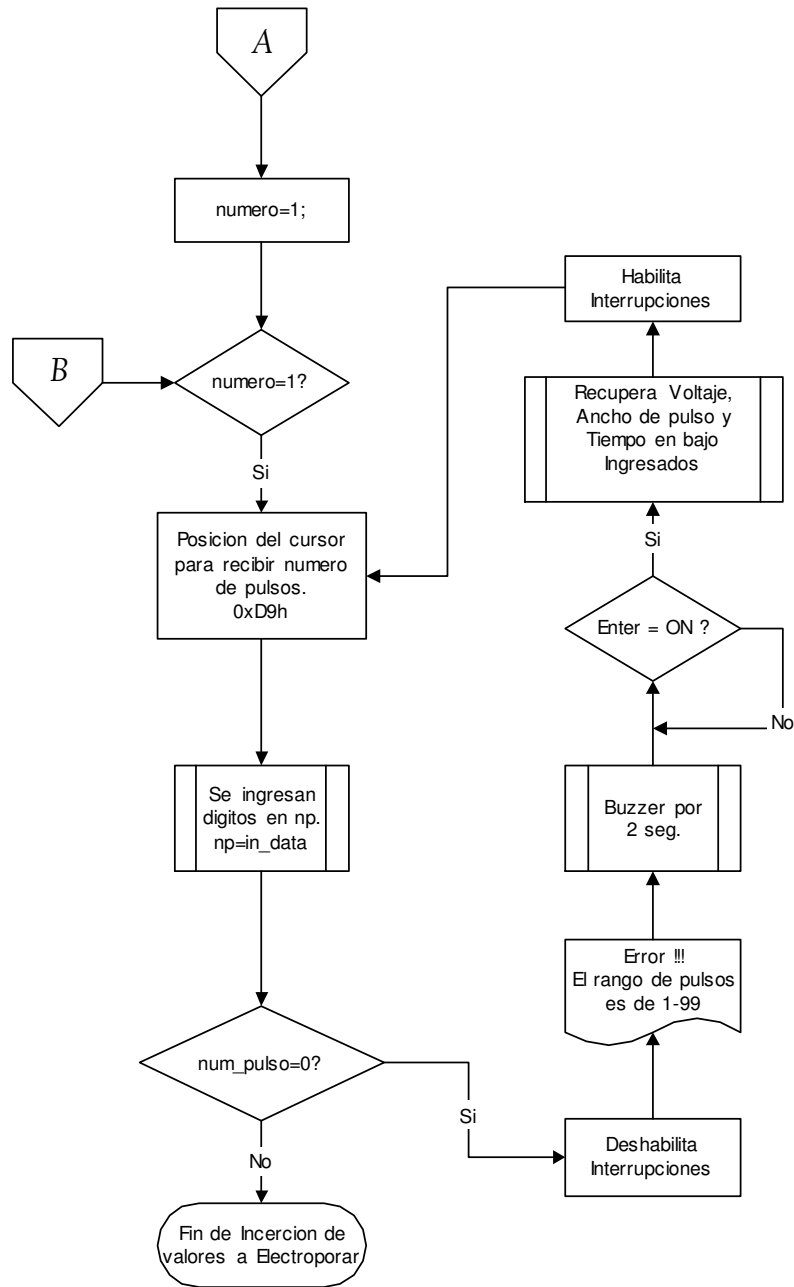
Función principal que especifica el comportamiento del sistema de electroporación.



6.2.2. Función de Inserción de Parámetros

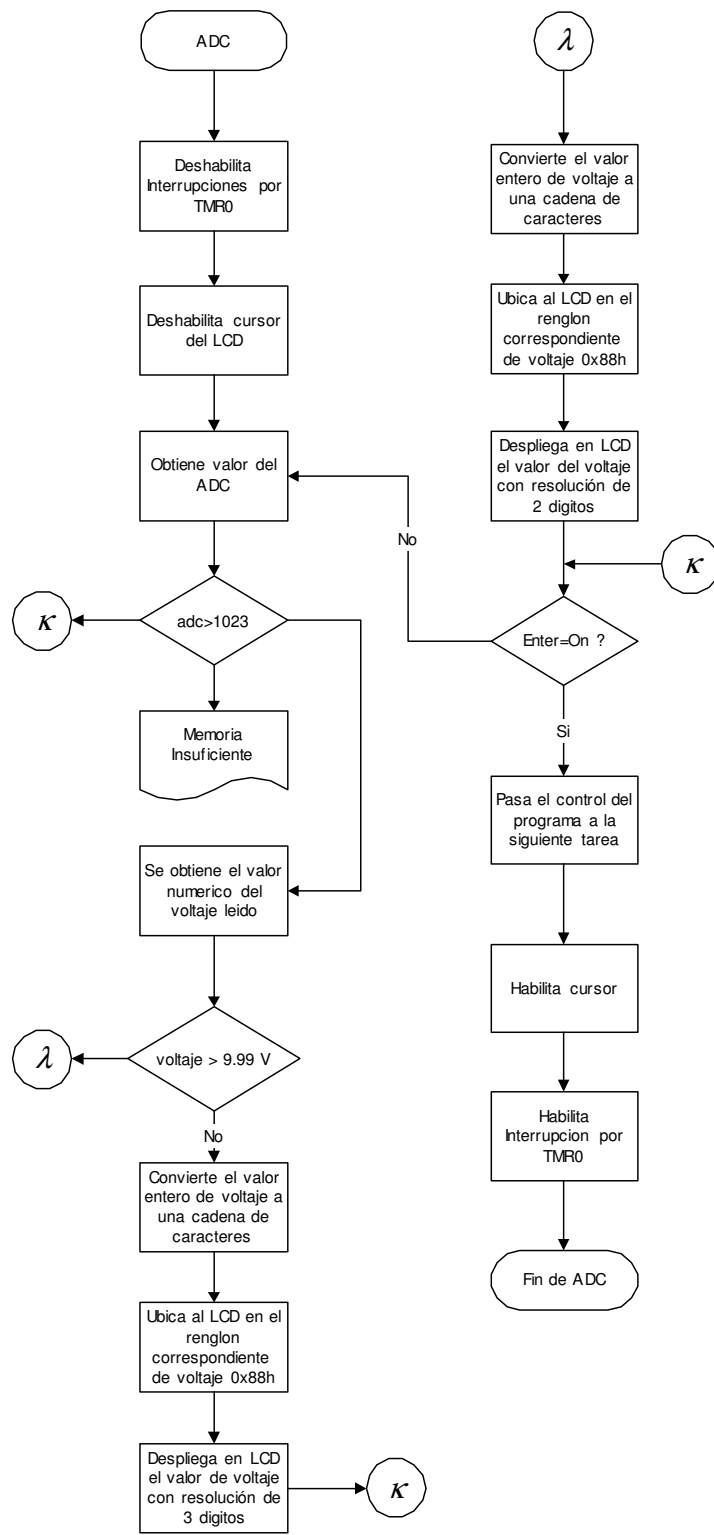
Esta subrutina se encarga de ingresar en distintas variables los valores del tratamiento ingresado para su posterior manejo y control dentro del proceso de electroporación.





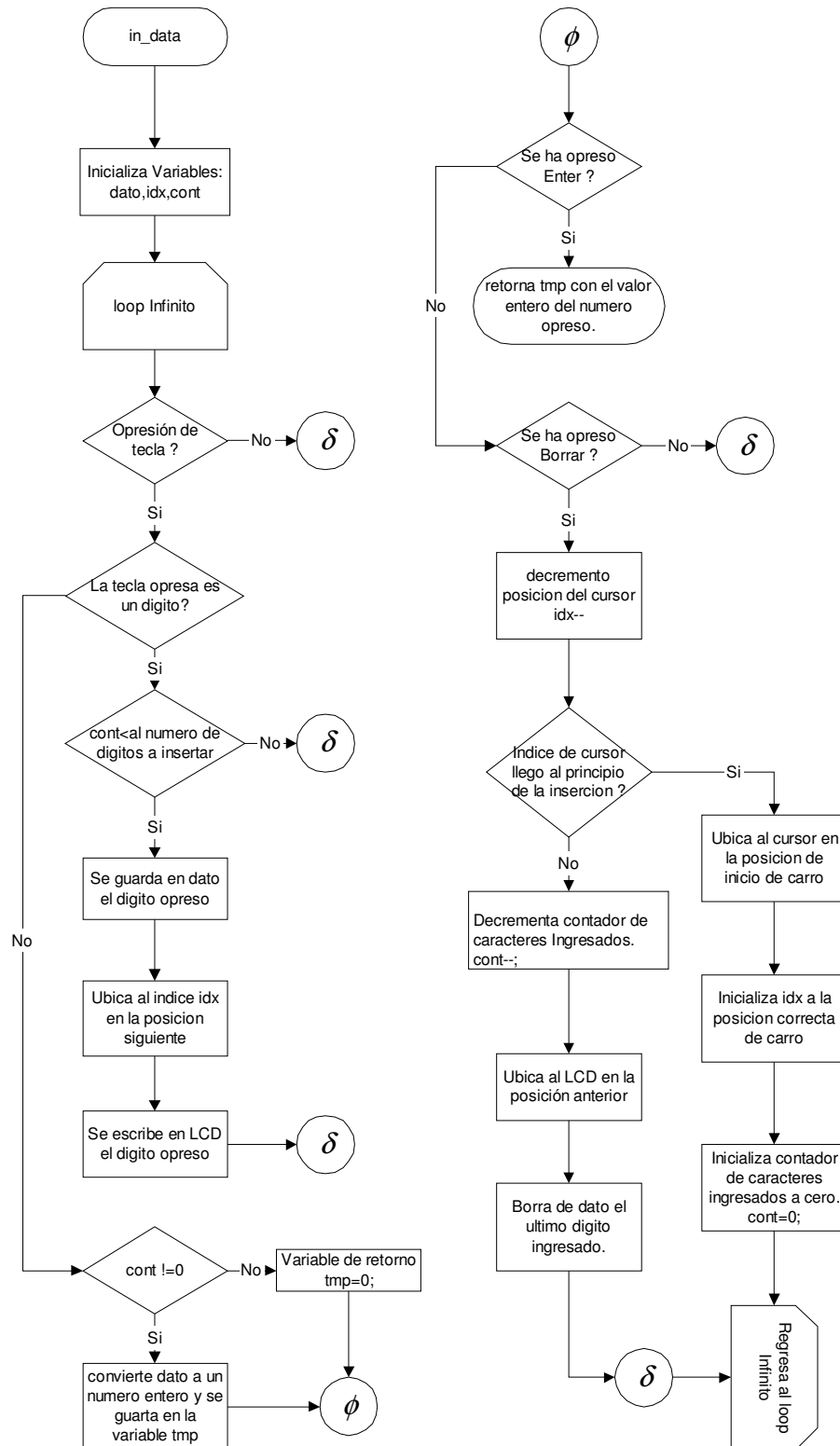
6.2.3. Función ADC

Esta función se encarga de censar el voltaje de la UE y mostrarlo en pantalla.



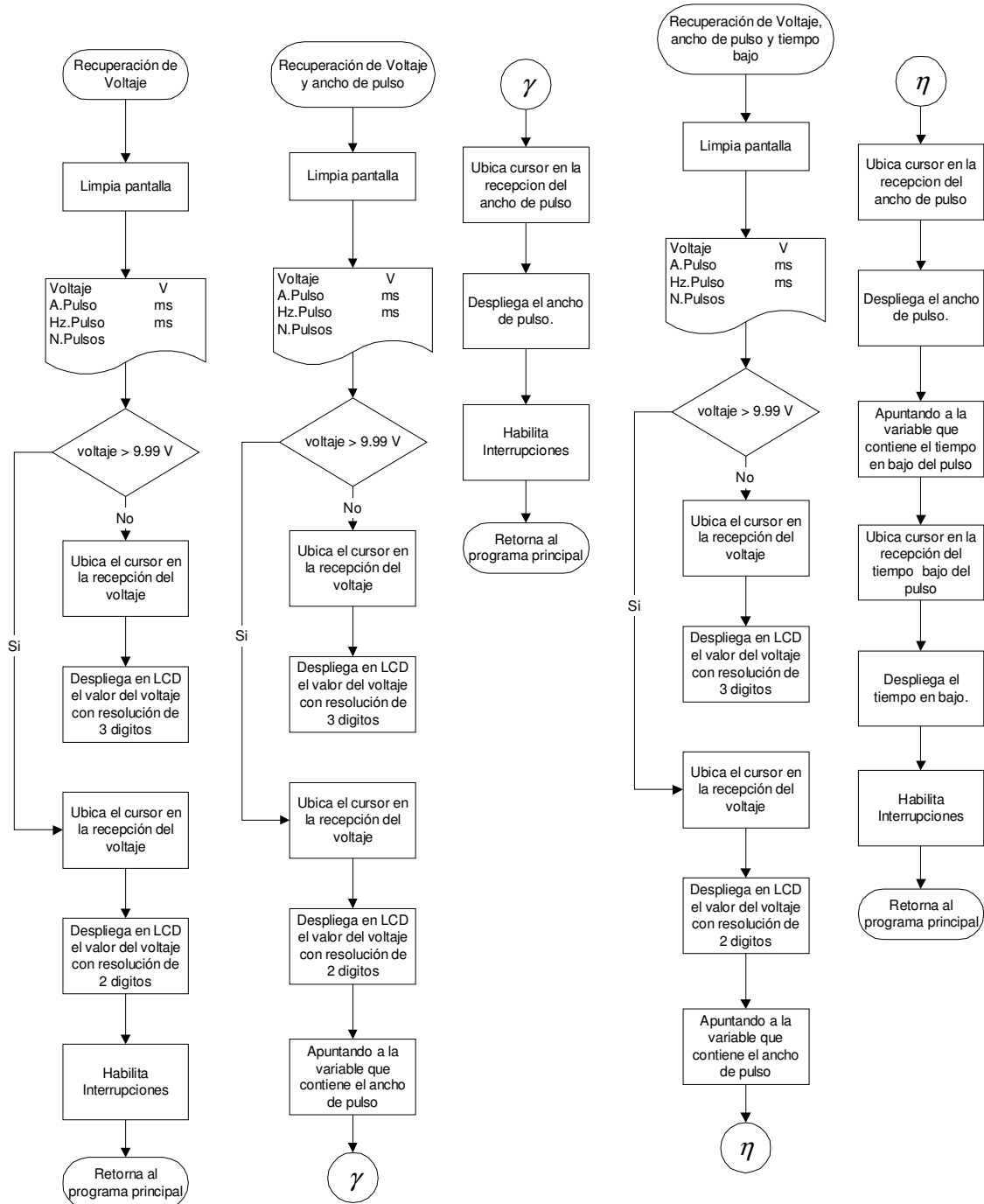
6.2.4. Función *in_data*

Esta función recolecta los dígitos que son tecleados, los despliega en LCD y guarda su valor como ASCII en una variable temporal.

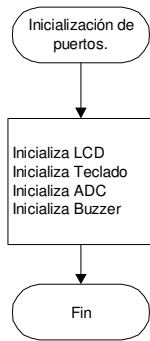


6.2.5. Funciones de Recuperación

Cada vez que se genera un error el sistema avisa del mismo, desplegando en pantalla dicho error, las funciones de recuperación despliegan en pantalla los valores ya ingresados por el usuario antes que se introdujera un error en el sistema.



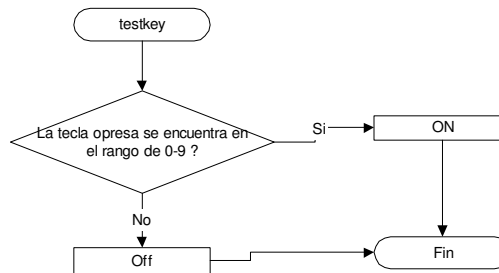
6.2.6. Función Inicialización de puertos



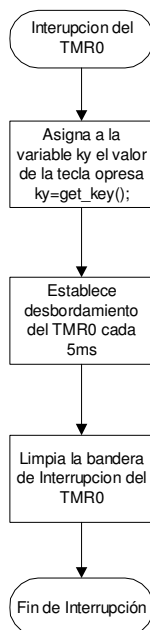
Esta función inicializa puertos, registros y funciones principales para el funcionamiento del sistema.

6.2.7. Función testkey

Esta función regresa un valor booleano dependiendo de la tecla opresa si es un numero retorna un 1 de lo contrario retorna un 0

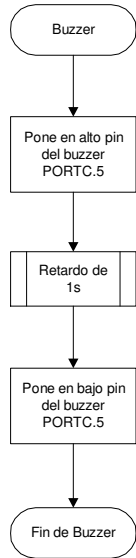


6.2.8. Función interrupción del TMR0



Esta función realiza interrupciones por hardware a través del TMR0 cada 5 ms para la lectura del teclado matricial.

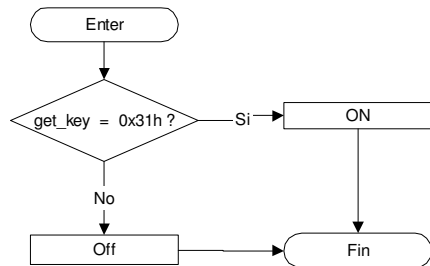
6.2.9. Función Buzzer



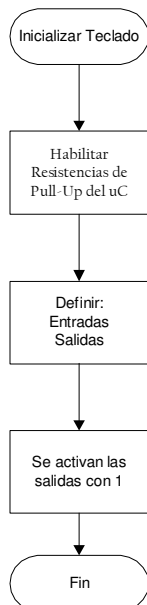
Esta función genera una señal auditiva que avisara al usuario de algún error y finalización de tratamiento.

6.2.10. Función Enter

Esta función regresa un valor booleano si la tecla Enter ha sido opresa.



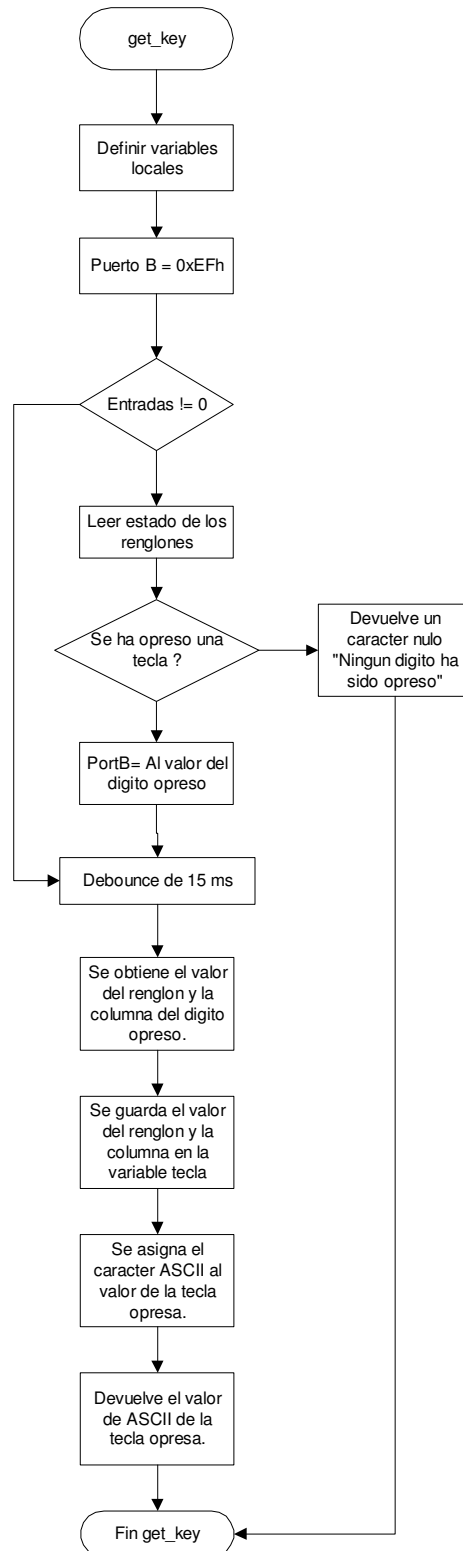
6.2.11. Funcion Inicializacin del Teclado



Esta función inicializa el puerto B, así como los resistores de jalón del puerto para la utilización del teclado matricial.

6.2.12. Función *get_key*

Esta función escanea el teclado y regresar el valor de la tecla pulsada a una variable para su posterior manejo



Capítulo 7

Resultados

El embrión de pollo es uno de los modelos animales utilizados en la investigación que permiten la manipulación experimental, los cuales ayudan a comprender mejor el proceso del desarrollo embrionario. Además de que este embrión proporciona accesibilidad y fácil manipulación durante los diferentes estadios embrionarios, lo que permite conocer la función de los genes.

Al producir la alteración en el potencial de la membrana con el sistema de electroporación se conseguirá una apertura momentánea en la zona de la membrana celular del embrión de pollo que esta siendo electroporada para el ingreso de un plásmido (ADN circular) hacia su interior, los poros formados por el pulso eléctrico se sellaran tras un corto periodo de tiempo, durante el cual el plásmido quedara insertado dentro de la célula.

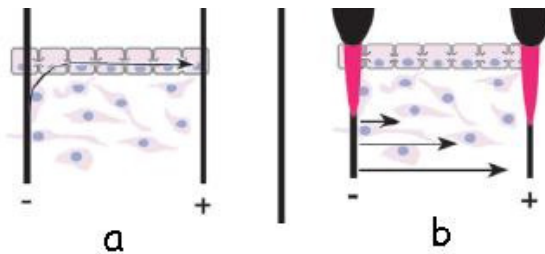


Figura 7.1 (a) “La corriente eléctrica afecta las células del ectodermo” (b) el aislamiento de los electrodos impide que se electroporesn células del ectodermo (Oberg, et al, 2002)

El plásmido se dosificara a través de una micro-aguja dentro de las celulas del interdigito del embrión de pollo con electrodos de tungsteno en los costados, estos electrodos deben ir aislados con pintura de acrílico, exceptuando la punta. La punta debe estar afilada de manera que atravesese el interdigitio del embrión sin traspasarlo, de esta forma se electroporarán tan solo células de mesénquima y no así del ectodermo figura 7.1 , el electrodo positivo tiene una forma de asa como base para sostener la extremidad (de flecha roja en la figura 7.2.b). Tanto los electrodos que se utilizaron como la micro-aguja que microinyectó el plásmido se muestra en la figura 7.2.a.

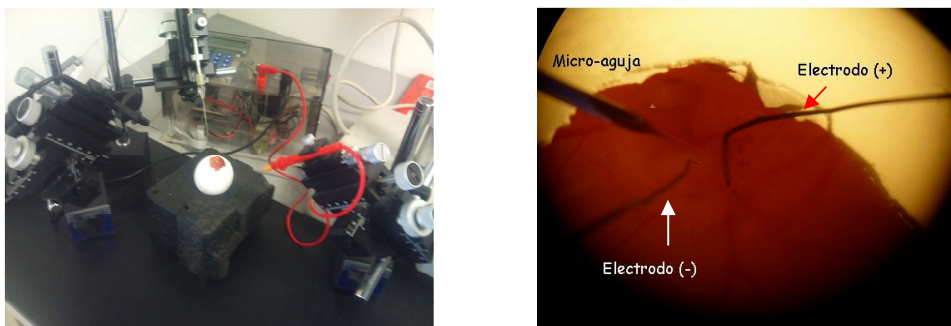


Figura 7.2 (a) “Electrodos apoyados en micromanipuladores (World Precision Instrument, inc)” (b) “Electrodos insertados en el interdigito y micro-aguja dosificando el plásmido”.

El plásmido utilizado es el GFP (proteína verde fluorescente); pcx-GFP (Figura 7.3).La construcción pex-GFP se encontró bajo el promotor de citomegalovirus y el promotor de β -Actina de pollo.

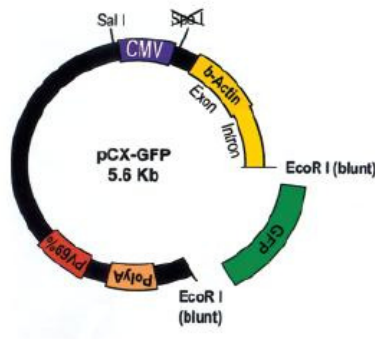


Figura 7.3 “ Construcción del plásmido pcx-GFP”

El éxito de la electroporación depende varios factores la pureza del plásmido, la concentración del mismo, y las condiciones del tratamiento de electroporación, para probar que dichos tratamientos no causen alteraciones al interdedo se llevaron a cabo diferentes condiciones de electroporación los cuales se muestran en la tabla comparativa embrionaria de pollo (Figura 7.4) siguiente.

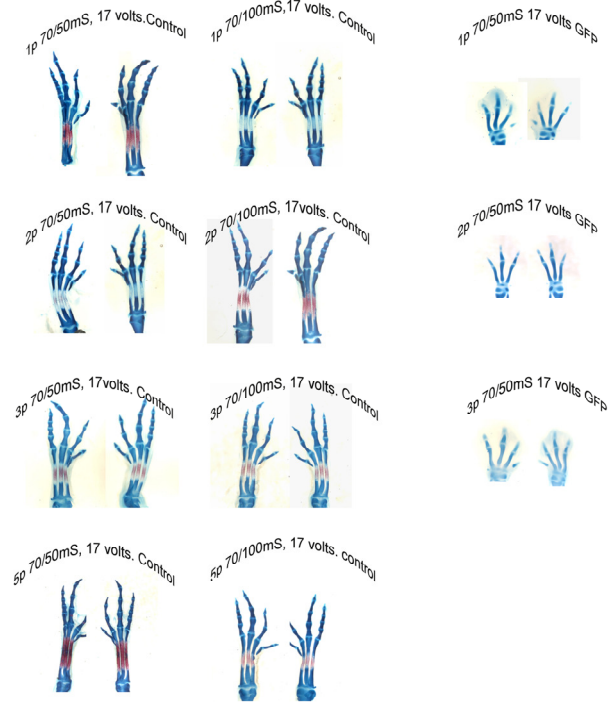


Figura 7.4 “Tabla comparativa de distintas condiciones de electroporación”

Y por ultimo en las figuras 7.5(a,b,c) se muestran las condiciones de electroporación optimas que se utilizan en el laboratorio del Departamento de Medicina Gnómica y Toxicología Ambiental.

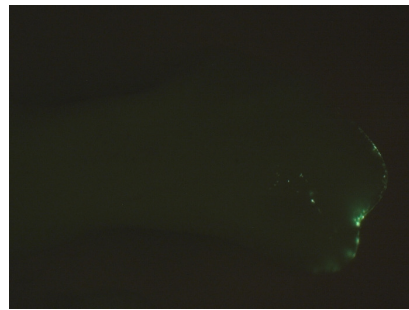
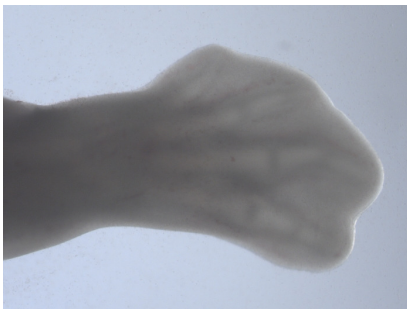


Figura 7.5.a Electroporación @ 13V, 70ms, 50ms, 2 pulsos

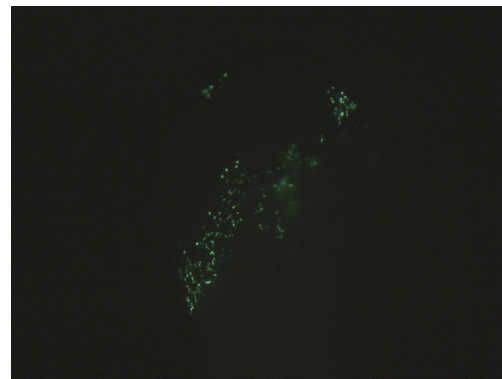


Figura 7.5.b Electroporación @ 13V, 70ms, 50ms, 2 pulsos

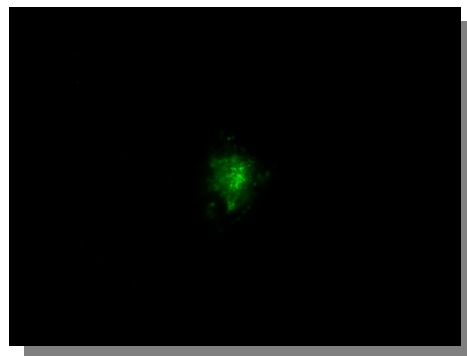
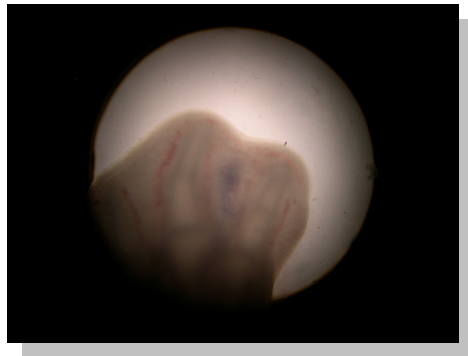


Figura 7.5.c Electroporación @ 17V, 70ms, 50ms, 1 pulsos

Con esto podemos concluir que las condiciones de electroporación descritas no afectan al fenotipo del interdígito de embrión de pollo.

Capítulo 8

Conclusiones

El objetivo de la tesis se enfocó en el desarrollo de un hardware de caracterización biológica. Se busco conformar la plataforma de caracterización basándose en un dispositivo programable, incorporar circuitos integrados comerciales para incrementar las capacidades del dispositivo, y habilitar al sistema para usar directamente una PC a través de una interfase vía un bus de control además de tener una interfase estándar de comunicación.

Esta primera etapa sólo corresponde a la puesta en marcha de un sistema de electroporación que deberá incluir mayores prestaciones. De éstas, la segunda etapa incluye incorporar técnicas de verificación y caracterización biológica, además de poder tener una base de datos correspondiente a todos los tratamientos realizados por los investigadores con el propósito final de desarrollar una bitácora que ayude al mejor entendimiento del proceso biológico de la morfogénesis.

Así mismo, el sistema de electroporación minimiza los errores que suele introducir el usuario, elimina las inestabilidades del equipo básico y evita el uso de metodologías de caracterización no adecuadas que algunos usuarios (por su formación primera) conservan como válidas.

Finalmente, es importante mencionar que este desarrollo es una solución parcial al equipamiento que debe tener el laboratorio de Investigaciones Biomédicas de la UNAM, ésta puesta en marcha del sistema tiene mayores prestaciones que el equipo básico utilizado, además de que resulta ser una herramienta útil de caracterización, la captura de información experimental es rápida y confiable y contar con una interfase para descargar datos de memoria a una PC hace que la representación gráfica sea inmediata. El propósito final es desarrollar un sistema completo de caracterización, de bajo costo que pueda ser usado en laboratorios de investigación biológica.

Bibliografía

- ✓ "MPLAB IDE Quick Chart" (DS51410)
- ✓ "MPLAB IDE Quick Start Guide" (DS51281)
- ✓ "MPLAB IDE User's Guide" (DS51519)
- ✓ "MPLAB C18 C Compiler Getting Started" (DS1295F)
- ✓ Programming 16-bit Microcontrollers in C "Learning to Fly the PIC 24" by Lucio Di Jasio.
- ✓ Designing Embedded Systems with PIC Microcontrollers "Principles and applications" by Tim Wilmshurst
- ✓ Advanced PIC Microcontroller Projects in C "From USB to ZIGBEE with the PIC 18F Series" by Dogan Ibrahim
- ✓ Microcontroladores PIC "Diseño práctico de aplicaciones 2a Parte por José María Angulo Usategui.
- ✓ Circuitos de Potencia e Estado Sólido. Manual para Proyectistas RCA Editorial Arbó S.A.C e I. Buenos Aires, Argentina. Depto. Técnico de la División Estado Sólido de RCA
- ✓ Dispositivos PLL de fuentes reguladas, temporizadores y de telecomunicaciones. Fundamentals of Electronic devices Ronald J. Tocci /mark E. Oliver
- ✓ Electrónica: Teoría de circuitos y dispositivos electrónicos. Robert L. Boylestad, Louis Nashelsky Pearson 8ª. Ed.
- ✓ Diseño Electrónico, Circuitos y Sistemas C.J. Savant Jr., Martin S. Roden, Gordon L. Carpenter. Pearson 3ª. Ed.
- ✓ How to use Intelligent L.C.D.s by Julián Klett Part One.
- ✓ How to use Intelligent L.C.D.s by Julián Klett Part Two.
- ✓ User's Manual ECM® 830 Electroporation System.
- ✓ Gene Pulser Xcell™ Electroporation System.
- ✓ Diseño y desarrollo de un electroporador, SOMI XXI

Referencias

- [1] <http://www.ccsinfo.com>
- [2] <http://htsoft.com>
- [3] <http://www.microchip.com>
- [4] <http://www.mikroe.com>
- [5] <http://www.fored.co.uk>
- [6] <http://www.oshonsoft.com>
- [7] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en010014
- [8] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en010065
- [9] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en542849
- [10] <http://www.iar.com/website1/1.0.1.0/68/1/>
- [11] <http://store.melabs.com/prod/boards/LABX1A.html>
- [12] http://www.bytecraft.com/MPC_C_Compiler_for_Microchip_PIC
- [13] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en010068
- [14] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en010020
- [15] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en010020
- [16] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en010046
- [17] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en028120
- [18] www.dalsemi.com

A P É N D I C E S

Lista de componentes

Tarjeta de Potencia

Resistores (5%, 250 mW, salvo indicación contraria)

R1= 15 Ω R7= 1M Ω R12=1.8 k Ω
R2= 1.8 Ω , 1W R8= 68 k Ω
R3= 150 Ω R9=6.8 k Ω , 1 W
R4= 12 Ω R10= 2.2 k Ω
R5,R6= 47 Ω R11= 150 k Ω
Pot 1 = Resistencia ajustable de alambre 10 k Ω

Condensadores

C1= 2200 μ F
C2,C4= 300 nF
C3= 2200 μ F
C5,C6,C7= 1 μ F

Semiconductores

U1= TL783 (Texas Instruments).
U2= 78L05
U3= MOC3031
U4= 7805
D1= Diodo Zener 1N4733A 5.1V/ 1W
Q1= 2N2907
Q2= TIP32C
Q3= PN2222
Q4= IRF1110

Varios

TR1= Transformador 60V/1A
TR2= Transformador 6V / 1A
BR1= Puente rectificador 1.2 A, 100V
BR2= Puente rectificador de Diodos.
F1= Fusible rápido 1A /125 V
F2= Fusible rápido 0.75 A /125 V
F3= Fusible rápido 0.5 A /125 V
K10= Conector horizontal 3 vías con paso de 100 mm .
K11= Conector horizontal 2 vías con paso de 100 mm.
K12= Conector horizontal 2 vías con paso de 100 mm.
K13= Conector horizontal 2 vías con paso de 150 mm.
K14= Conector horizontal 2 vías con paso de 150 mm.

Placa de Control

Resistores

R14,R22=470 Ω

R15= 10 k Ω

R16,R17= 220 Ω

R18,R19= 4.7 k Ω

R20=3.9 k Ω

R21=1 k Ω

POT 2 = Resistencia ajustable 2.4 k Ω horizontal.

Condensadores

C9,C10,C11,C12= 22 pF

C13= 10 nF

Semiconductores

U5= PIC18F452 (Microchip).

U6= 24C256F (Microchip).

U7=DS1307 (Maxim).

Varios

K1= Conector 10 terminales vertical.

K2= Conector 10 terminales vertical.

K3= Conector 6 vías con paso de 100 mm.

K4= Conector 3 vías con paso de 100 mm.

K5= Conector 2 vías con paso de 100 mm.

K6= Conector 2 vías con paso de 100 mm.

K7= Conector 10 terminales vertical

K8= Conector 10 terminales vertical

K9= Conector 2 vías con paso de 100 mm.

BUZ1 Buzzer

X1= Cristal de cuarzo de 10 Mhz

X2= Cristal de cuarzo de 32.768 Khz

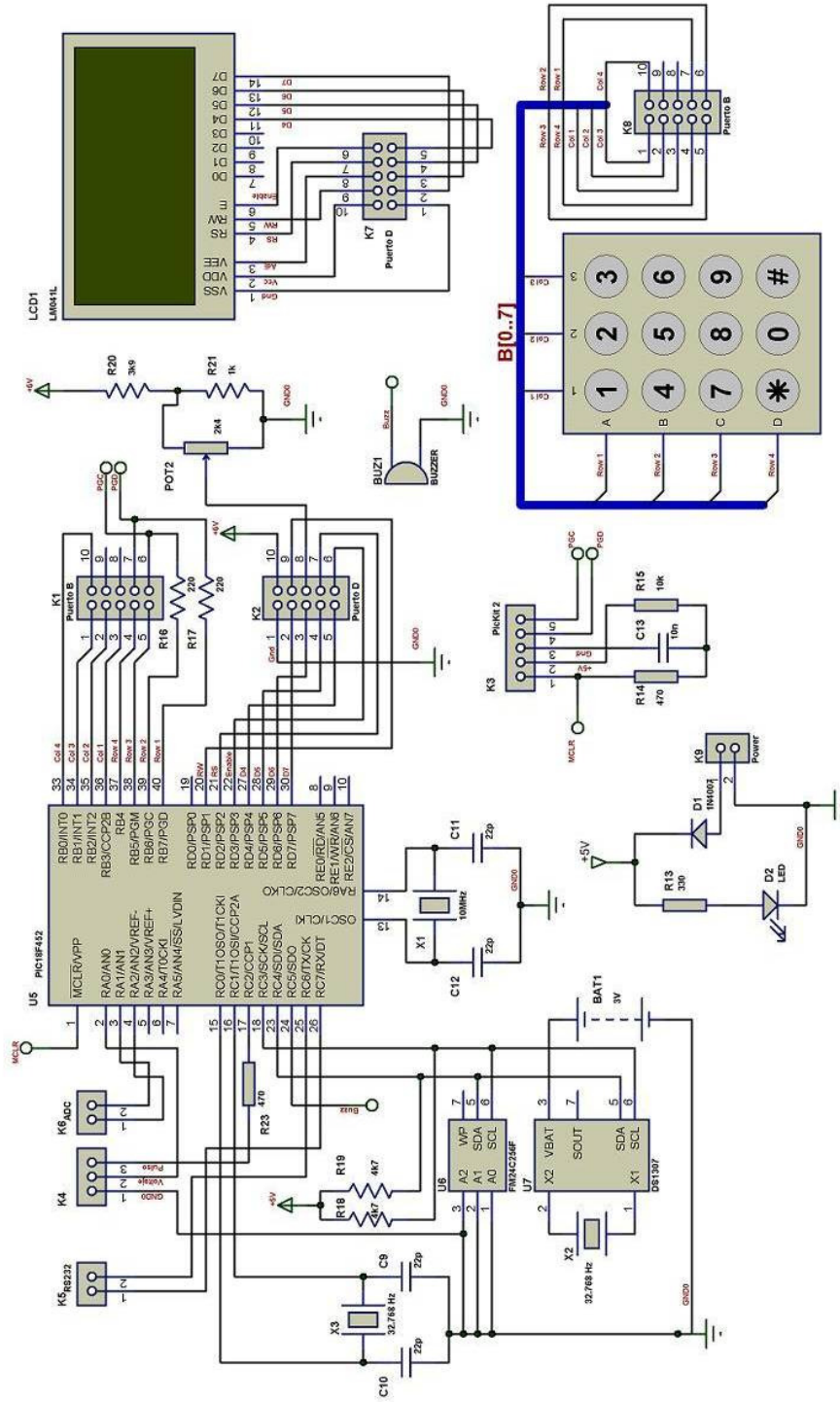
X3= Cristal de cuarzo de 32.768 Khz

BAT1=Zócalo para pila

LCD1 = Visualizador alfanumérico de 4 líneas x 16 caracteres (EPS 050176-73)

S1-S12 = Botón-pulsador, SPN

Esquema Electrónico de Control





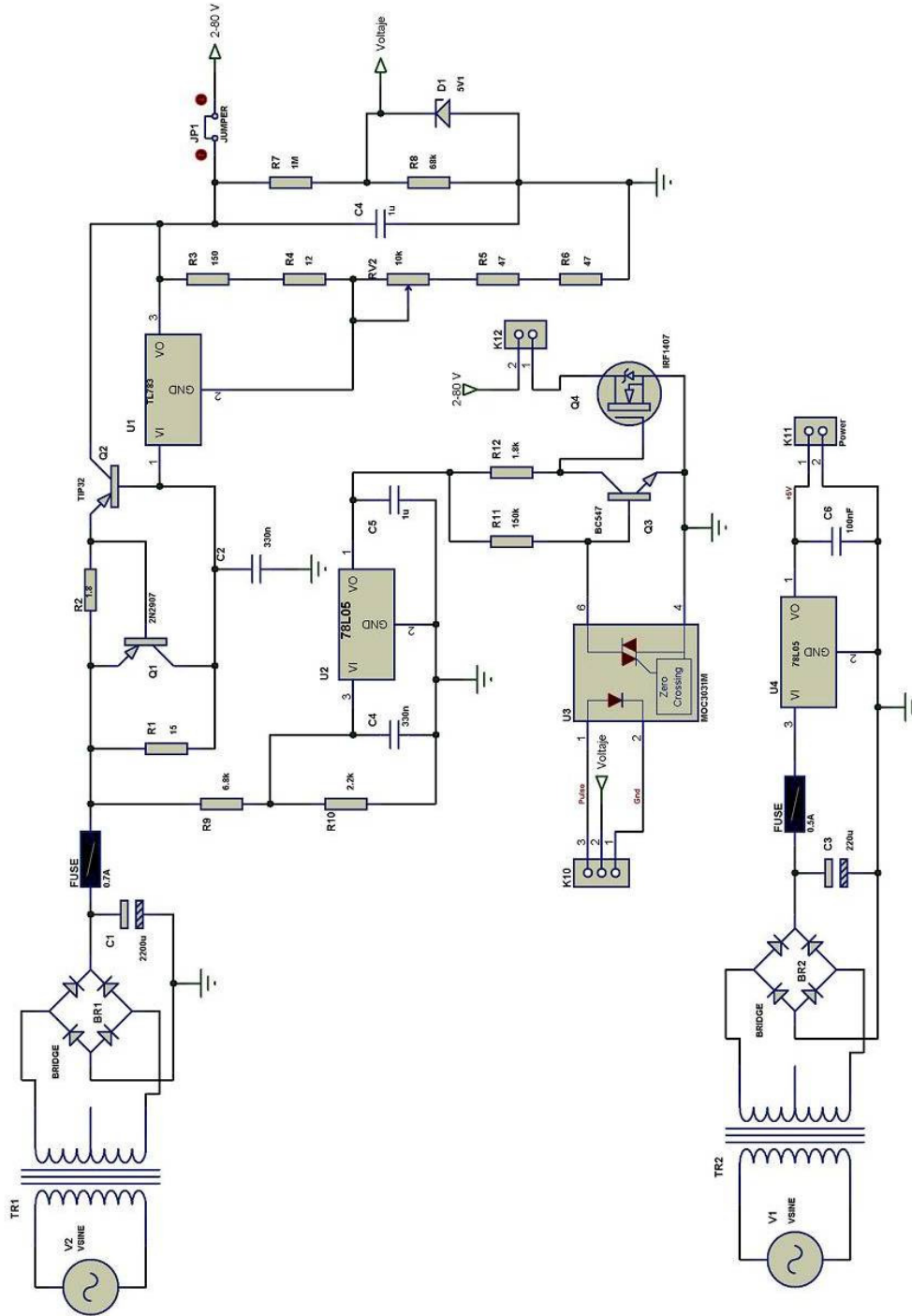
Diseño y Desarrollo de un Electroplador Digital

Unidad de Control

Autor: Gutiérrez Ruiz, Jorge Octavio



Esquema Electrónico de Potencia





Diseño y Desarrollo de un Electropotador Digital

Unidad de Potencia

Autor: Gutiérrez Ruiz Jorge Octavio



Firmware

Electroporador.c

C:\ProgramMicro\C compiler\Electroporador\electroporador.c

```

1  /*electroporador.c -Codigo principal de la Unidad de Control
2     que realiza todas las funciones del Electroporador
3     Creada por Jorge O. Gutiérrez Ruiz, Julio 13, 2010
4     Código liberado para el dominio público.
5  */
6  #include "LCD_PIC.c"      // Manejo del LCD
7  #include "electroporador.h" //Archivo de cabecera del electroporador
8  #include "teclado.c"     //Manejo del Teclado
9  #include "funciones.c"
10 #include <stdlib.h>      //
11 #include <p18cxxx.h>     //Libreria que especifica puertos del PIC
12 //Registro que maneja el control del programa
13 #define Read      regis_0.b0
14 #define ancho    regis_0.b1
15 #define t_bajo   regis_0.b2
16 #define numero   regis_0.b3
17 #define pulso    regis_0.b4
18 #define fin      regis_0.b5
19 #define wait     regis_0.b6
20 #define start    regis_0.b7
21 /*-----
22     Subroutine: Interrupt_High_Vector
23     Synopsys: Inicializa las Interrupciones de alta prioridad
24 -----*/
25 #pragma code Interrupt_High_Vector = 0x08 //Define la direccion=0008h
26 void Interrupt (void) //para high interruption
27 {
28     _asm
29     goto Interrupt_Handler // Salto a la rutina de Interrupción
30     _endasm
31 }
32 #pragma code
33 /*-----
34     Subroutine: main
35     Synopsys: Rutina principal
36 -----*/
37 void main (void)
38 {
39     INIT_PORTS (); //Inicializacion de puertos principales
40     regis_0._byte=0;
41     LCD_COM (CLR_DISP); //Limpia pantalla del LCD
42     INTCON=0b11100000; //Inicializa TMRO
43     TOCON=0b01000101;
44     T1CON=0b10100000;
45     TMR0L=61; //Desbordamiento del TMRO cada 5ms
46     LCD_write(1,1,"Electroporador"); //Mensaje de bienvenida al sistema
47     LCD_write(2,2,"UNAM-CCADET");
48     LCD_write(3,5,"V 1.5");
49     delay_5s();
50     start=1; //Primera etapa del programa
51
52     while (1)
53     {
54         if (start) //Mensajes del tratamiento
55         {
56             LCD_COM(CLR_DISP); //Limpia pantalla del LCD
57             LCD_write (1,0,"Voltaje V"); //Mensajes al LCD
58             LCD_write (2,0,"Ancho ms");

```

```

59     LCD_write (3,0,"Separacion  ms");
60     LCD_write (4,0,"No. pulsos  ");
61     regis_0._byte=0x01;          //Pasa el control del programa a Read
62 }
63 if (Read) //Lectura del ADC
64 {
65     TOCON&=0X7F;                //Deshabilita interrupcion TMRO
66     LCD_COM (BLINK_OFF);        //Deshabilita cursor del LCD
67     do
68     { read_voltage (); }        //Fin de do
69     while (get_key()!='');      //Espera el enter del voltaje deseado
70     regis_0._byte= 0x02;        //Pasa el control del programa a ancho
71     LCD_COM (BLINK_ON);         //Habilita parpadeo del cursor
72     TOCON^=0X80;               //Habilita Interrupcion de TMRO
73 }//Fin de Read
74 if (ancho) // Recepción del ancho del pulso Tiempo en Alto
75 {
76     do
77     {
78         cursor=0xCB;
79         LCD_COM(cursor);        //Cursor listo para recibir Ancho de pulso
80         ancho_p=in_data(3);     //Se ingresan tres digitos para el ancho de pulso
81         if (ancho_p==0)         //El ancho de pulso no puede ser igual a cero
82         {
83             LCD_COM(CLR_DISP);  //Limpia pantalla del LCD
84             INTCONbits.GIE =0;  //Deshabilitando Interrupciones globa
85             LCD_write (1,0,"El ancho de "); //Error
86             LCD_write (2,0,"pulso no puede");
87             LCD_write (3,0,"ser cero.");
88             buzzer ();          //Aviso auditivo para el usuario
89             while (get_key()!='') //Espera a que se oprima el Enter
90             /* Nothing */ ;
91             LCD_COM(CLR_DISP);  //Limpia pantalla del LCD
92             LCD_write (1,0,"Voltaje  V"); //Mensajes al LCD
93             LCD_write (2,0,"Ancho  ms");
94             LCD_write (3,0,"Separacion  ms");
95             LCD_write (4,0,"No. pulsos  ");
96             if (voltaje>9999)   //Escribe el voltaje ya seleccionado
97             {
98                 LCD_COM (0X88);
99                 LCD_DAT(displ_val[0]); //Escribe en LCD la parte entera
100                LCD_DAT(displ_val[1]); //Escribe en LCD la parte entera
101                LCD_DAT(0X2E);        //Escribe le punto decimal
102                LCD_ascii (val_frac); //Escribe la parte fraccionaria en LCD
103            }
104            else
105            {
106                LCD_COM (0X88);
107                LCD_DAT(displ_val[0]); //Escribe en LCD la parte entera
108                LCD_DAT(0X2E);        //Escribe le punto decimal
109                LCD_ascii (val_frac); //Escribe la parte fraccionaria en LCD
110            }
111            INTCONbits.GIE =1;        //Habilitando Interrupciones globales.
112        }
113    }
114    while(ancho_p==0);              //Regresa a pedir ancho de pulso
115    regis_0._byte= 0x04;           //Pasa el control del programa a t_bajo
116    ky=0xff;                       //Evita que la siguiente etapa lea el enter anterior.

```

```

117 }//Fin de if(ancho)
118 if (t_bajo) // Recepción del tren del pulso Tiempo en Bajo
119 {
120     do
121     {
122         cursor=0x9B;
123         LCD_COM(cursor); //Cursor listo para recibir frecuencia
124         hz=in_data(3); //Se ingresan dos digitos para la frecuencia del pulso
125         if ((hz ==0) || (hz%50!=0)) //El ancho de pulso no puede ser igual a cero
126         {
127             //y debe ser divisible entre 50
128             INTCONbits.GIE =0; //Deshabilitando Interrupciones globales.
129             LCD_COM(CLR_DISP); //Limpia pantalla del LCD
130             LCD_write (1,0,"Solo multiples ");
131             LCD_write (2,0,"de 50.");
132             buzzer (); //Aviso auditivo para el usuario
133             while (get_key()!=' '); //Espera a que se oprima el Enter
134             LCD_COM(CLR_DISP); //Limpia pantalla del LCD
135             LCD_write (1,0,"Voltaje V"); //Mensajes al LCD
136             LCD_write (2,0,"Ancho ms");
137             LCD_write (3,0,"Separacion ms");
138             LCD_write (4,0,"No. pulsos ");
139             if (voltaje>9999) //Voltaje mayor a 9.999V
140             {
141                 //Escribe el voltaje ya seleccionado
142                 LCD_COM (0X88);
143                 LCD_DAT(displ_val[0]); //Escribe en LCD la parte entera
144                 LCD_DAT(displ_val[1]); //Escribe en LCD la parte entera
145                 LCD_DAT(0X2E); //Escribe el punto decimal
146                 LCD_ascii (val_frac); //Escribe la parte fraccionaria en LCD
147             }
148             else
149             {
150                 LCD_COM (0X88);
151                 LCD_DAT(displ_val[0]); //Escribe en LCD la parte entera
152                 LCD_DAT(0X2E); //Escribe el punto decimal
153                 LCD_ascii (val_frac); //Escribe la parte fraccionaria en LCD
154             }
155             disp_val_ptr=&data[0]; //Escribe el ancho ya seleccionado
156             itoa(ancho_p,disp_val_ptr); //Convierte el varlo de ancho a un string
157             LCD_COM(0xCB); //Ubica al LCD en el renglon de A.pulso
158             LCD_ascii(disp_val_ptr); //Escribe en LCD el ancho de pulso
159             INTCONbits.GIE =1; //Habilita interrupciones
160         }
161     }
162     while((hz==0) || (hz%50!=0)); //Regresa a pedir la frecuencia del pulso
163     regis_0_byte=0x08 ; //Pasa el control del programa a pedir el No. de pulsos
164     ky=0xff; //Evita que la siguiente etapa lea el enter anterior.
165 }//Fin de if(t_bajo)
166 if (numero) //Recepción del Numero de pulsos
167 {
168     do
169     {
170         cursor=0xDB;
171         LCD_COM(cursor); //Cursor listo para recibir frecuencia
172         num_pulso=in_data(2); //Se ingresan dos digitos para la frecuencia del pulso
173         if (num_pulso==0) //El ancho de pulso no puede ser igual a cero
174         {
175             //y debe ser divisible entre 50
176             INTCONbits.GIE =0; //Habilitando Interrupciones globales.
177             LCD_COM(CLR_DISP); //Limpia pantalla del LCD

```

```

175 LCD_write (1,0,"El intervalo de");
176 LCD_write (2,0,"pulsos es de ");
177 LCD_write (3,0,"1-99.");
178 buzzer (); //Aviso auditivo para el usuario
179 while (get_key()!=' ');
180 //Espera a que se oprima el Enter
181 LCD_COM(CLR_DISP); //Limpia pantalla del LCD
182 LCD_write (1,0,"Voltaje V"); //Mensajes al LCD
183 LCD_write (2,0,"Ancho ms");
184 LCD_write (3,0,"Separacion ms");
185 LCD_write (4,0,"No. pulsos ");
186 if (voltaje>9999)
187 { //Escribe el voltaje ya seleccionado
188 LCD_COM (0X88);
189 LCD_DAT(displ_val[0]); //Escribe en LCD la parte entera
190 LCD_DAT(displ_val[1]); //Escribe en LCD la parte entera
191 LCD_DAT(0X2E); //Escribe le punto decimal
192 LCD_ascii (val_frac); //Escribe la parte fraccionaria en LCD
193 }
194 else
195 {
196 LCD_COM (0X88);
197 LCD_DAT(displ_val[0]); //Escribe en LCD la parte entera
198 LCD_DAT(0X2E); //Escribe le punto decimal
199 LCD_ascii (val_frac); //Escribe la parte fraccionaria en LCD
200 }
201 displ_val_ptr=&data[0];
202 itoa(ancho_p,displ_val_ptr); //Escribe el ancho ya seleccionado
203 LCD_COM(0xCB);
204 LCD_ascii(displ_val_ptr); //Escribe en LCD la parte entera
205 displ_val_ptr=&data[0];
206 itoa(hz,displ_val_ptr); //Escribe el tren de pulso ya seleccionado
207 LCD_COM(0x9B);
208 LCD_ascii(displ_val_ptr); //Escribe en LCD la parte entera
209 INTCONbits.GIE =1; //Habilitando Interrupciones globales.
210 } //Fin de if (num_pulso==0)
211 } // Fin de do
212 while(num_pulso==0); //Regresa a pedir No. de pulsos
213 } // Fin de if(numero)
214
215 LCD_COM(CLR_DISP); //Limpia pantalla del LCD
216 LCD_write (1,0,"Los parametros"); //Permiso de realizar el tratamiento
217 LCD_write (2,0,"son correctos ?");
218 LCD_write (3,0,"SI: Enter");
219 LCD_write (4,0,"NO: Borrar");
220 while((ky!=' ')&&(ky!='*'));
221 if (ky==' ')
222 {
223 INTCONbits.GIE =0; //Deshabilitando Interrupciones globales.
224 do
225 {
226 LCD_COM(CLR_DISP); //Limpia pantalla del LCD
227 for (cont2=1; cont2<=num_pulso;cont2++)
228 temp1s(); //Realiza el tratamiento ingresado
229 buzzer (); //Aviso auditivo Fin del Tratamiento
230 INTCONbits.GIE =1; //Deshabilitando Interrupciones globales.
231 LCD_write (1,0,"Voltaje V"); //Mensajes al LCD
232 LCD_write (2,0,"Ancho ms");

```

```

233         LCD_write (3,0,"Separacion  ms");
234         LCD_write (4,0,"No. pulsos  ");
235         if (voltaje>9999)          // Escribe los parametros del tratamiento
236         {
237             LCD_COM (0X88);
238             LCD_DAT(disp_val[0]); //Escribe en LCD la parte entera
239             LCD_DAT(disp_val[1]); //Escribe en LCD la parte entera
240             LCD_DAT(0X2E);       //Escribe le punto decimal
241             LCD_ascii (val_frac); //Escribe la parte fraccionaria en LCD
242         }
243         else
244         {
245             LCD_COM (0X88);
246             LCD_DAT(disp_val[0]); //Escribe en LCD la parte entera
247             LCD_DAT(0X2E);       //Escribe le punto decimal
248             LCD_ascii (val_frac); //Escribe la parte fraccionaria en LCD
249         }
250         disp_val_ptr=&data[0];    //Variable que guarda parametros
251         itoa(ancho_p,disp_val_ptr); //Se guarda A.pulso para desplegar a LCD
252         LCD_COM(0xCB);
253         LCD_ascii(disp_val_ptr);  //Escribe en LCD A.Pulso
254         disp_val_ptr=&data[0];
255         itoa(hz,disp_val_ptr);    //Se guarda tiempo en bajo para desplegar a LCD
256         LCD_COM(0x9B);
257         LCD_ascii(disp_val_ptr); //Escribe en LCD tren de pulsos
258         disp_val_ptr=&data[0];
259         itoa(num_pulso,disp_val_ptr); //Se guarda num_pulso para desplegar a LCD
260         LCD_COM(0xDB);
261         LCD_ascii(disp_val_ptr);  //Escribe en LCD numero de pulsos
262         while (ky!='');           //Espera a que se oprima el Enter
263         LCD_COM(CLR_DISP);        //Limpia pantalla del LCD
264         LCD_write (1,0,"Repetir ?"); //Permiso de realizar el tratamiento
265         LCD_write (3,0,"Si: Enter");
266         LCD_write (4,0,"No: Borrar");
267         while((ky!='')&&(ky!='*'));
268     }
269     while (ky=='');
270     regis_0_byte=0x80 ; //Pasa el control del programa a start
271 }
272 else // Se ha opreso Borrar
273 { regis_0_byte=0x80 ; //Pasa el control del programa a start
274   INTCONbits.GIE =1; //Habilita Interrupciones globales
275 }
276
277 } // Fin de while (1)
278 } // Fin de main
279
280 /*-----*/
281     Subroutine: Interrupt_Handler
282     Synopsys: Maneja todas las interrupciones del programa
283     /*-----*/
284 #pragma interrupt Interrupt_Handler //La funcion ADCVoltage se especifica como high-priority ISR
285 void Interrupt_Handler (void) //No transfiere parametros por ser ISR
286 {
287     if (INTCONbits.TMROIF)
288     {
289         ky=get_key(); //Obtiene la tecla pulsada
290         TMR0L=61; //Desbordamiento del TMR0 cada 5 ms

```

```

291     INTCONbits.TMR0IF =0;    //Limpia bandera de interrupcion del TMRO
292 }
293 } //Final de Interrupt Handler
294
295
296 /*-----
297     Subroutine: INIT_PORTS
298     Synopsys: Inicializa todos los registros y funciones asociados con la aplicacion.
299 -----*/
300
301 void INIT_PORTS (void)
302 {
303     INI_ADC ();    //Inicializa el ADC
304     INI_LCD ();    //Inicializa el LCD
305     INI_keypad ();    //Inicializa el teclado
306     INI_BUZZER ();//Inicializa el buzzer
307     bitclr(TRISC,2);    //Inicializa la salida del pin del tratamiento
308     bitclr(PORTC,2);
309 }
310

```

Electroporador.h

C:\ProgramMicro\C compiler\Electroporador\electroporador.h

```
1  /*electroporador.h - Libreria de cabecera para el manejo del Electroporador
2   Creada por Jorge O. Gutiérrez Ruiz, Julio 13, 2010
3   Código liberado para el dominio público.
4  */
5
6  #include "boolean.h"    /* funciones booleanas */
7  #include <adc.h>        //Libreria de conversión AD
8  #include <delays.h>     //Libreria de retardos por software
9  #include <stdlib.h>     //
10 #include <timers.h>     //Libreria de temporizador
11 #include <p18cxxx.h>    //Libreria que especifica puertos del PIC
12
13 // Configuración de fuses del PIC
14 #pragma config OSC = OFF
15 #pragma config OSC = HS    // - set HSPLL oscilador
16 #pragma config WDT = OFF  // -disable watchdog timer
17 #pragma config LVP = OFF  // -disabel low voltage programming
18 #pragma config DEBUG = ON  // -enable background debugging
19
20 #define bitset(Var,bitno) ((Var) |= 1 << (bitno))
21 #define bitclr(Var,bitno) ((Var) &= ~(1 << (bitno)))
22 #define ON    1          //Definicion de booleanos
23 #define OFF   0
24 #define BASE  4.169      // Voltaje Max a la entrada del AD (4.176/1023)
25
26
27
28 //Definicion de Prototipos de Funcion
29 void INIT_PORTS (void);    //Inializa todos los puertos principales y aplicaciones
30 void delay_5s (void);     //Retardo de 5 segundos por software
31 void delay_1s (void);     //Retardo de 1 segundo por software
32 void INI_ADC (void);      //Inicializa ADC
33 void INI_BUZZER (void);   //Inicializa Buzzer
34 unsigned int x_read(int no); //Variable con Oversampling del ADC
35 unsigned int ADC_READ (void); //Activación y Lectura del ADC
36 void buzzer (void);       //Aviso auditivo del buzzer
37 int in_data ( int n);     //Recepción de parametros del tratamiento
38 void temp1s (void);       //Ejecución del tratamiento
39 void Interrupt_Handler (void); //Maneja todas las Interrupciones del PIC
40 void read_voltage (void); //Lee el voltaje del ADC y despliega en pantalla
41 // Creacion de un Registro de 8 bits
42 typedef union _BYTE
43 {
44     unsigned char _byte;
45     struct
46     {
47         unsigned b0:1;
48         unsigned b1:1;
49         unsigned b2:1;
50         unsigned b3:1;
51         unsigned b4:1;
52         unsigned b5:1;
53         unsigned b6:1;
54         unsigned b7:1;
55     };
56 } BYTE;
57
58 //Definicion de Variables globales          //near:indica que la variable es puesta en Access RAM
```



```

59 int multi1=1000; //volatile:El contenido de la variable puede cambiar
60 char *disp_val_ptr;
61 #pragma udata access var_glob //udata: locates uninitialised user variables in data memory
62 near volatile char addr,val_high,val_low,data[3], enter,ky;
63 near volatile char cont, cont2, i, disp_val [6], val_frac[4];
64 near volatile int tmp,ADC,integer,frac,cursor,ADC_current,idx;
65 near volatile long int voltaje, x;
66 near volatile int ancho_p, hz,bajo;
67 near volatile int num_pulso;
68 near BYTE regis_0,regis_temp;
69 #pragma udata
70

```

Hojas de Especificaciones

TL783



TL783

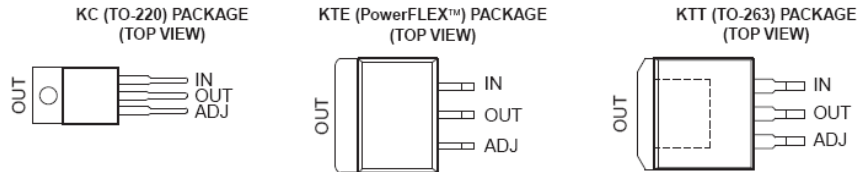
www.ti.com

SLVS036M – SEPTEMBER 1981 – REVISED APRIL 2008

HIGH-VOLTAGE ADJUSTABLE REGULATOR

FEATURES

- Output Adjustable From 1.25 V to 125 V When Used With an External Resistor Divider
- 700-mA Output Current
- Full Short-Circuit, Safe-Operating-Area, and Thermal-Shutdown Protection
- 0.001%/V Typical Input Voltage Regulation
- 0.15% Typical Output Voltage Regulation
- 76-dB Typical Ripple Rejection



DESCRIPTION/ORDERING INFORMATION

The TL783 is an adjustable three-terminal high-voltage regulator with an output range of 1.25 V to 125 V and a DMOS output transistor capable of sourcing more than 700 mA. It is designed for use in high-voltage applications where standard bipolar regulators cannot be used. Excellent performance specifications, superior to those of most bipolar regulators, are achieved through circuit design and advanced layout techniques.

As a state-of-the-art regulator, the TL783 combines standard bipolar circuitry with high-voltage double-diffused MOS transistors on one chip, to yield a device capable of withstanding voltages far higher than standard bipolar integrated circuits. Because of its lack of secondary-breakdown and thermal-runaway characteristics usually associated with bipolar outputs, the TL783 maintains full overload protection while operating at up to 125 V from input to output. Other features of the device include current limiting, safe-operating-area (SOA) protection, and thermal shutdown. Even if ADJ is disconnected inadvertently, the protection circuitry remains functional.

Only two external resistors are required to program the output voltage. An input bypass capacitor is necessary only when the regulator is situated far from the input filter. An output capacitor, although not required, improves transient response and protection from instantaneous output short circuits. Excellent ripple rejection can be achieved without a bypass capacitor at the adjustment terminal.

ORDERING INFORMATION⁽¹⁾

T _J	PACKAGE ⁽²⁾		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 125°C	PowerFLEX™ – KTE	Reel of 2000	TL783CKTER	TL783
	TO-263 – KTT	Reel of 500	TL783CKTTR	TL783C
	TO-220 – KC	Tube of 50	TL783CKC	TL783C

(1) For the most current package and ordering information, see the Package Option Addendum at the end of this document, or see the TI web site at www.ti.com.

(2) Package drawings, thermal data, and symbolization are available at www.ti.com/packaging.



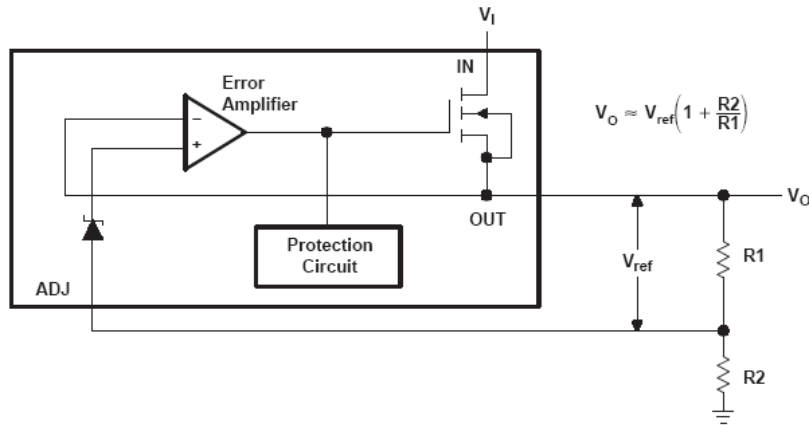
Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PowerFLEX, PowerPAD are trademarks of Texas Instruments.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of the Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1981–2008, Texas Instruments Incorporated

FUNCTIONAL BLOCK DIAGRAM



Absolute Maximum Ratings⁽¹⁾

over operating temperature range (unless otherwise noted)

		MIN	MAX	UNIT
$V_I - V_O$	Input-to-output differential voltage		125	V
T_J	Operating virtual junction temperature		150	°C
T_{stg}	Storage temperature range	-65	150	°C

(1) Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

Package Thermal Data⁽¹⁾

PACKAGE	BOARD	θ_{JC}	$\theta_{JP}^{(2)}$	θ_{JA}
PowerFLEX (KTE)	High K, JESD 51-5		2.7°C/W	23°C/W
TO-263 (KTT)	High K, JESD 51-5	18°C/W	1.94°C/W	25.3°C/W
TO-220 (KC)	High K, JESD 51-5	17°C/W	3°C/W	19°C/W

- (1) Maximum power dissipation is a function of $T_{J(max)}$, θ_{JA} , and T_A . The maximum allowable power dissipation at any allowable ambient temperature is $P_D = (T_{J(max)} - T_A)/\theta_{JA}$. Operating at the absolute maximum T_J of 150°C can affect reliability. Due to variations in individual device electrical characteristics and thermal resistance, the built-in thermal overload protection may be activated at power levels slightly above or below the rated dissipation.
- (2) For packages with exposed thermal pads, such as QFN, PowerPAD™, or PowerFLEX, θ_{JP} is defined as the thermal resistance between the die junction and the bottom of the exposed pad.

Recommended Operating Conditions

		MIN	MAX	UNIT
$V_I - V_O$	Input-to-output differential voltage		125	V
I_O	Output current	15	700	mA
T_J	Operating virtual junction temperature	0	125	°C

Electrical Characteristics
 $V_I - V_O = 25\text{ V}$, $I_O = 0.5\text{ A}$, $T_J = 0^\circ\text{C}$ to 125°C (unless otherwise noted)

PARAMETER	TEST CONDITIONS ⁽¹⁾	MIN	TYP	MAX	UNIT
Input voltage regulation ⁽²⁾	$V_I - V_O = 20\text{ V}$ to 125 V , $P \leq$ rated dissipation	$T_J = 25^\circ\text{C}$	0.001	0.01	%/ V
		$T_J = 0^\circ\text{C}$ to 125°C	0.004	0.02	
Ripple rejection	$\Delta V_{I(PP)} = 10\text{ V}$, $V_O = 10\text{ V}$, $f = 120\text{ Hz}$	66	76		dB
Output voltage regulation	$I_O = 15\text{ mA}$ to 700 mA , $T_J = 25^\circ\text{C}$	$V_O \leq 5\text{ V}$	7.5	25	mV
		$V_O \geq 5\text{ V}$	0.15	0.5	%
	$I_O = 15\text{ mA}$ to 700 mA , $P \leq$ rated dissipation	$V_O \leq 5\text{ V}$	20	70	mV
		$V_O \geq 5\text{ V}$	0.3	1.5	%
Output voltage change with temperature			0.4		%
Output voltage long-term drift	1000 hours at $T_J = 125^\circ\text{C}$, $V_I - V_O = 125\text{ V}$		0.2		%
Output noise voltage	$f = 10\text{ Hz}$ to 10 kHz , $T_J = 25^\circ\text{C}$		0.003		%
Minimum output current to maintain regulation	$V_I - V_O = 125\text{ V}$			15	mA
Peak output current	$V_I - V_O = 25\text{ V}$, $t = 1\text{ ms}$		1100		mA
	$V_I - V_O = 15\text{ V}$, $t = 30\text{ ms}$		715		
	$V_I - V_O = 25\text{ V}$, $t = 30\text{ ms}$	700	900		
	$V_I - V_O = 125\text{ V}$, $t = 30\text{ ms}$	100	250		
ADJ input current			83	110	μA
Change in ADJ input current	$V_I - V_O = 15\text{ V}$ to 125 V , $I_O = 15\text{ mA}$ to 700 mA , $P \leq$ rated dissipation		0.5	5	μA
Reference voltage (OUT to ADJ) ⁽³⁾	$V_I - V_O = 10\text{ V}$ to 125 V , $I_O = 15\text{ mA}$ to 700 mA , $P \leq$ rated dissipation	1.2	1.27	1.3	V

- (1) Pulse-testing techniques maintain the junction temperature as close to the ambient temperature as possible. Thermal effects must be taken into account separately.
- (2) Input voltage regulation is expressed here as the percentage change in output voltage per 1-V change at the input
- (3) Due to the dropout voltage and output current-limiting characteristics of this device, output current is limited to less than 700 mA at input-to-output voltage differentials of less than 25 V.



LM78LXX Series 3-Terminal Positive Regulators

General Description

The LM78LXX series of three terminal positive regulators is available with several fixed output voltages making them useful in a wide range of applications. When used as a zener diode/resistor combination replacement, the LM78LXX usually results in an effective output impedance improvement of two orders of magnitude, and lower quiescent current. These regulators can provide local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow the LM78LXX to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment.

The LM78LXX is available in the plastic TO-92 (Z) package, the plastic SO-8 (M) package and a chip sized package (8-Bump micro SMD) using National's micro SMD package technology. With adequate heat sinking the regulator can deliver 100 mA output current. Current limiting is included to limit the peak output current to a safe value. Safe area pro-

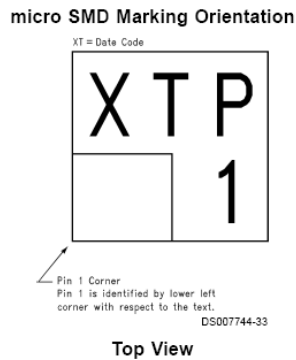
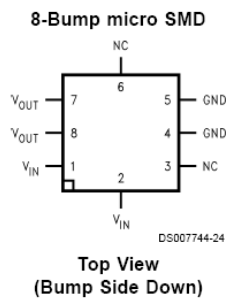
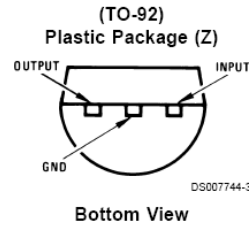
tection for the output transistors is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Features

- LM78L05 in micro SMD package
- Output voltage tolerances of $\pm 5\%$ over the temperature range
- Output current of 100 mA
- Internal thermal overload protection
- Output transistor safe area protection
- Internal short circuit current limit
- Available in plastic TO-92 and plastic SO-8 low profile packages
- No external components
- Output voltages of 5.0V, 6.2V, 8.2V, 9.0V, 12V, 15V

LM78LXX Series 3-Terminal Positive Regulators

Connection Diagrams



Absolute Maximum Ratings (Note 1)

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

Power Dissipation (Note 5)	Internally Limited
Input Voltage	35V
Storage Temperature	-65°C to +150°C

Operating Junction Temperature	SO-8	0°C to 125°C
	micro SMD	-40°C to 85°C
Soldering Information	Infrared or Convection (20 sec.)	235°C
	Wave Soldering (10 sec.)	260°C (lead time)
ESD Susceptibility (Note 2)		1kV

LM78LXX Electrical Characteristics Limits in standard typeface are for $T_J = 25^\circ\text{C}$, **Bold typeface applies over 0°C to 125°C for SO-8 package and -40°C to 85°C for micro SMD package.** Limits are guaranteed by production testing or correlation techniques using standard Statistical Quality Control (SQC) methods. Unless otherwise specified: $I_O = 40\text{ mA}$, $C_I = 0.33\ \mu\text{F}$, $C_O = 0.1\ \mu\text{F}$.

LM78L05

Unless otherwise specified, $V_{IN} = 10\text{V}$

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V_O	Output Voltage		4.8	5	5.2	V
		$7\text{V} \leq V_{IN} \leq 20\text{V}$ $1\text{ mA} \leq I_O \leq 40\text{ mA}$ (Note 3)	4.75		5.25	
		$1\text{ mA} \leq I_O \leq 70\text{ mA}$ (Note 3)	4.75		5.25	
ΔV_O	Line Regulation	$7\text{V} \leq V_{IN} \leq 20\text{V}$		18	75	mV
		$8\text{V} \leq V_{IN} \leq 20\text{V}$		10	54	
ΔV_O	Load Regulation	$1\text{ mA} \leq I_O \leq 100\text{ mA}$		20	60	mV
		$1\text{ mA} \leq I_O \leq 40\text{ mA}$		5	30	
I_Q	Quiescent Current			3	5	mA
ΔI_Q	Quiescent Current Change	$8\text{V} \leq V_{IN} \leq 20\text{V}$ $1\text{ mA} \leq I_O \leq 40\text{ mA}$			1.0 0.1	
V_n	Output Noise Voltage	$f = 10\text{ Hz to } 100\text{ kHz}$ (Note 4)		40		μV
$\frac{\Delta V_{IN}}{\Delta V_{OUT}}$	Ripple Rejection	$f = 120\text{ Hz}$ $8\text{V} \leq V_{IN} \leq 16\text{V}$	47	62		dB
I_{PK}	Peak Output Current			140		mA
$\frac{\Delta V_O}{\Delta T}$	Average Output Voltage Tempco	$I_O = 5\text{ mA}$		-0.65		mV/°C
$V_{IN}(\text{Min})$	Minimum Value of Input Voltage Required to Maintain Line Regulation			6.7	7	V
θ_{JA}	Thermal Resistance (8-Bump micro SMD)			230.9		°C/W



**TIP31A/31C
TIP32A/32B/32C**

**COMPLEMENTARY SILICON POWER
TRANSISTORS**

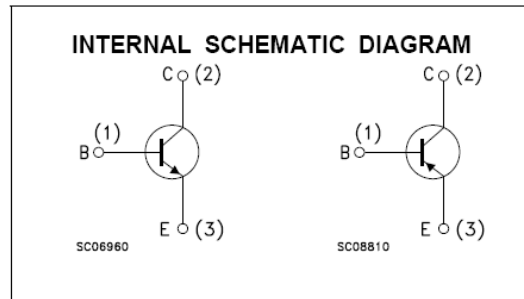
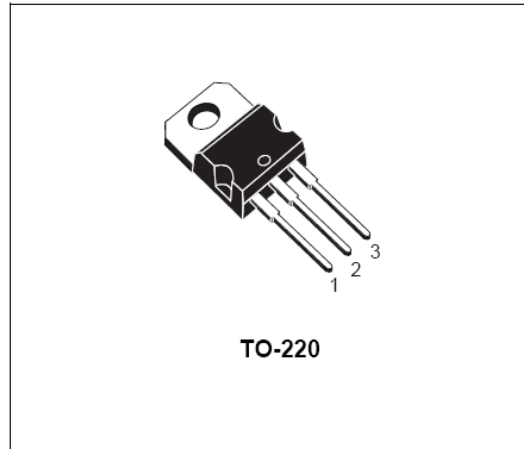
APPLICATION

- LINEAR AND SWITCHING INDUSTRIAL EQUIPMENT

DESCRIPTION

The TIP31A and TIP31C are silicon Epitaxial-Base NPN transistors mounted in Jedec TO-220 plastic package. They are intended for use in medium power linear and switching applications.

The complementary PNP types are TIP32A and TIP32C respectively. Also TIP32B is a PNP type.



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value			Unit	
		NPN	TIP31A	TIP31C		
		PNP	TIP32A	TIP32C		
V _{CB0}	Collector-Base Voltage (I _E = 0)		60	80	100	V
V _{CEO}	Collector-Emitter Voltage (I _B = 0)		60	80	100	V
V _{EBO}	Emitter-Base Voltage (I _C = 0)		5			V
I _C	Collector Current		3			A
I _{CM}	Collector Peak Current		5			A
I _B	Base Current		1			A
P _{tot}	Total Dissipation at T _{case} ≤ 25 °C T _{amb} ≤ 25 °C		40			W
			2			W
T _{stg}	Storage Temperature		-65 to 150			°C
T _j	Max. Operating Junction Temperature		150			°C

For PNP types voltage and current values are negative

TIP31A/TIP31C/TIP32A/TIP32B/TIP32C

THERMAL DATA

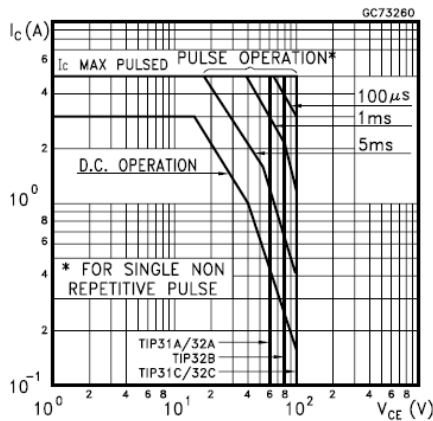
$R_{thj-case}$	Thermal Resistance Junction-case	Max	3.12	$^{\circ}C/W$
$R_{thj-amb}$	Thermal Resistance Junction-ambient	Max	62.5	$^{\circ}C/W$

ELECTRICAL CHARACTERISTICS ($T_{case} = 25^{\circ}C$ unless otherwise specified)

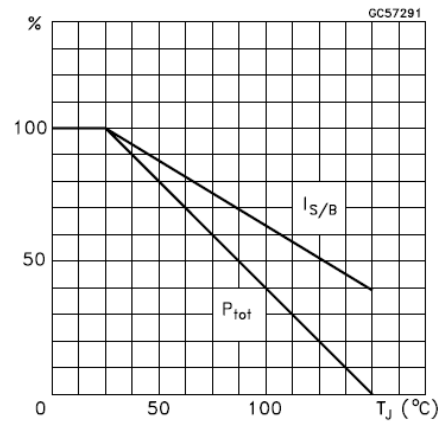
Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
I_{CEO}	Collector Cut-off Current ($I_B = 0$)	for TIP31A/32A $V_{CE} = 30 V$ for TIP31C/32B/32C $V_{CE} = 60 V$			0.3 0.3	mA mA
I_{CES}	Collector Cut-off Current ($V_{BE} = 0$)	for TIP31A/32A $V_{CE} = 60 V$ for TIP32B $V_{CE} = 80 V$ for TIP31C/32C $V_{CE} = 100 V$			0.2 0.2 0.2	mA mA mA
I_{EBO}	Emitter Cut-off Current ($I_C = 0$)	$V_{EB} = 5 V$			1	mA
$V_{CEO(sus)*}$	Collector-Emitter Sustaining Voltage ($I_B = 0$)	$I_C = 30 mA$ for TIP31A/32A for TIP32B for TIP31C/32C	60 80 100			V V V
$V_{CE(sat)*}$	Collector-Emitter Saturation Voltage	$I_C = 3 A$ $I_B = 375 mA$			1.2	V
$V_{BE(on)*}$	Base-Emitter Voltage	$I_C = 3 A$ $V_{CE} = 4 V$			1.8	V
h_{FE*}	DC Current Gain	$I_C = 1 A$ $V_{CE} = 4 V$ $I_C = 3 A$ $V_{CE} = 4 V$	25 10		50	
h_{fe}	Small Signal Current Gain	$I_C = 0.5 A$ $V_{CE} = 10 V$ $f = 1 KHz$ $I_C = 0.5 A$ $V_{CE} = 10 V$ $f = 1 MHz$	20 3			

* Pulsed : pulse duration = 300 μs , duty cycle $\leq 2\%$
For PNP types voltage and current values are negative.

Safe Operating Area




Derating Curves

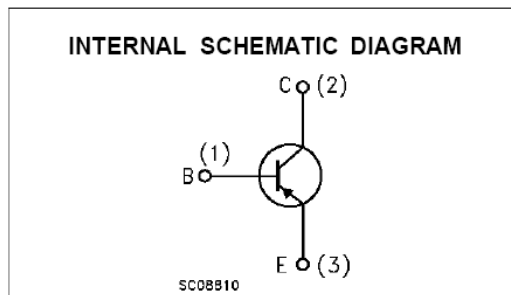
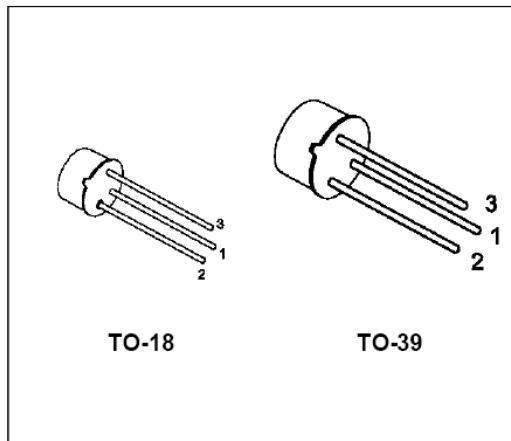


GENERAL PURPOSE AMPLIFIERS AND SWITCHES

DESCRIPTION

The 2N2905 and 2N2907 are silicon planar epitaxial PNP transistors in Jedec TO-39 (for 2N2905) and in Jedec TO-18 (for 2N2907) metal case. They are designed for high speed saturated switching and general purpose application.

 2N2905 approved to CECC 50002-102,
 2N2907 approved to CECC 50002-103
 available on request.



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_{CB0}	Collector-Base Voltage ($I_E = 0$)	-60	V
V_{CE0}	Collector-Emitter Voltage ($I_B = 0$)	-40	V
V_{EB0}	Emitter-Base Voltage ($I_C = 0$)	-5	V
I_C	Collector Current	-0.6	A
P_{tot}	Total Dissipation at $T_{amb} \leq 25^\circ\text{C}$ for 2N2905 for 2N2907 at $T_{case} \leq 25^\circ\text{C}$ for 2N2905 for 2N2907	0.6	W
		0.4	W
		3	W
		1.8	W
T_{stg}	Storage Temperature	-65 to 200	$^\circ\text{C}$
T_j	Max. Operating Junction Temperature	200	$^\circ\text{C}$

THERMAL DATA

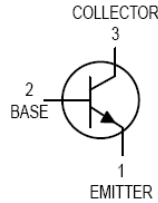
		TO-39	TO-18	
R _{thj-case}	Thermal Resistance Junction-Case	Max	58.3	97.3 °C/W
R _{thj-amb}	Thermal Resistance Junction-Ambient	Max	292	437.5 °C/W

ELECTRICAL CHARACTERISTICS (T_{case} = 25 °C unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
I _{CBO}	Collector Cut-off Current (I _E = 0)	V _{CB} = -50 V V _{CB} = -50 V T _{case} = 150 °C			-20 -20	nA μA
I _{CEX}	Collector Cut-off Current (V _{BE} = -0.5V)	V _{CE} = -30 V			-50	nA
I _{BEX}	Base Cut-off Current (V _{BE} = -0.5V)	V _{CE} = -30 V			-50	nA
V _{(BR)CBO} *	Collector-Base Breakdown Voltage (I _E = 0)	I _C = -10 μA	-60			V
V _{(BR)CEO} *	Collector-Emitter Breakdown Voltage (I _B = 0)	I _C = -10 mA	-40			V
V _{(BR)EBO} *	Emitter-Base Breakdown Voltage (I _C = 0)	I _E = -10 μA	-5			V
V _{CE(sat)} *	Collector-Emitter Saturation Voltage	I _C = -150 mA I _B = -15 mA I _C = -500 mA I _B = -50 mA			-0.4 -1.6	V V
V _{BE(sat)} *	Base-Emitter Saturation Voltage	I _C = -150 mA I _B = -15 mA I _C = -500 mA I _B = -50 mA			-1.3 -2.6	V V
h _{FE} *	DC Current Gain	I _C = -0.1 mA V _{CE} = -10 V I _C = -1 mA V _{CE} = -10 V I _C = -10 mA V _{CE} = -10 V I _C = -150 mA V _{CE} = -10 V I _C = -500 mA V _{CE} = -10 V	35 50 75 100 30		300	
f _T	Transition Frequency	V _{CE} = -20 V f = 100 MHz I _C = -50 mA	200			MHz
C _{EBO}	Emitter Base Capacitance	I _C = 0 V _{EB} = -2 V f = 1MHz			30	pF
C _{CBO}	Collector Base Capacitance	I _E = 0 V _{CB} = -10 V f = 1MHz			8	pF
t _d	Delay Time	V _{CC} = -30 V I _C = -150 mA I _{B1} = -15 mA			10	ns
t _r	Rise Time	V _{CC} = -30 V I _C = -150 mA I _{B1} = -15 mA			40	ns
t _s	Storage Time	V _{CC} = -6 V I _C = -150 mA I _{B1} = -I _{B2} = -15 mA			80	ns
t _f	Fall Time	V _{CC} = -6 V I _C = -150 mA I _{B1} = -I _{B2} = -15 mA			30	ns

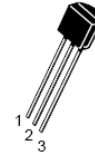
* Pulsed: Pulse duration = 300 μs, duty cycle ≤ 1%

General Purpose Transistors
NPN Silicon



MPS2222
MPS2222A*

*Motorola Preferred Device



CASE 29-04, STYLE 1
TO-92 (TO-226AA)

MAXIMUM RATINGS

Rating	Symbol	MPS2222	MPS2222A	Unit
Collector–Emitter Voltage	V_{CE0}	30	40	Vdc
Collector–Base Voltage	V_{CBO}	60	75	Vdc
Emitter–Base Voltage	V_{EBO}	5.0	6.0	Vdc
Collector Current — Continuous	I_C	600		mAdc
Total Device Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	625	5.0	mW mW/ $^\circ\text{C}$
Total Device Dissipation @ $T_C = 25^\circ\text{C}$ Derate above 25°C	P_D	1.5	12	Watts mW/ $^\circ\text{C}$
Operating and Storage Junction Temperature Range	T_J, T_{stg}	–55 to +150		$^\circ\text{C}$

THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	200	$^\circ\text{C/W}$
Thermal Resistance, Junction to Case	$R_{\theta JC}$	83.3	$^\circ\text{C/W}$

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
----------------	--------	-----	-----	------

OFF CHARACTERISTICS

Collector–Emitter Breakdown Voltage ($I_C = 10\text{ mAdc}, I_E = 0$)	MPS2222 MPS2222A	$V_{(BR)CEO}$	30 40	— —	Vdc
Collector–Base Breakdown Voltage ($I_C = 10\ \mu\text{Adc}, I_E = 0$)	MPS2222 MPS2222A	$V_{(BR)CBO}$	60 75	— —	Vdc
Emitter–Base Breakdown Voltage ($I_E = 10\ \mu\text{Adc}, I_C = 0$)	MPS2222 MPS2222A	$V_{(BR)EBO}$	5.0 6.0	— —	Vdc
Collector Cutoff Current ($V_{CE} = 60\text{ Vdc}, V_{EB(off)} = 3.0\text{ Vdc}$)	MPS2222A	I_{CEX}	—	10	nAdc
Collector Cutoff Current ($V_{CB} = 50\text{ Vdc}, I_E = 0$) ($V_{CB} = 60\text{ Vdc}, I_E = 0$) ($V_{CB} = 50\text{ Vdc}, I_E = 0, T_A = 125^\circ\text{C}$) ($V_{CB} = 50\text{ Vdc}, I_E = 0, T_A = 125^\circ\text{C}$)	MPS2222 MPS2222A MPS2222 MPS2222A	I_{CBO}	— — — —	0.01 0.01 10 10	μAdc
Emitter Cutoff Current ($V_{EB} = 3.0\text{ Vdc}, I_C = 0$)	MPS2222A	I_{EBO}	—	100	nAdc
Base Cutoff Current ($V_{CE} = 60\text{ Vdc}, V_{EB(off)} = 3.0\text{ Vdc}$)	MPS2222A	I_{BL}	—	20	nAdc

Preferred devices are Motorola recommended choices for future use and best overall value.

MPS2222 MPS2222A
ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted) (Continued)

Characteristic	Symbol	Min	Max	Unit
ON CHARACTERISTICS				
DC Current Gain ($I_C = 0.1\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $T_A = -55^\circ\text{C}$) ($I_C = 150\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) ⁽¹⁾ ($I_C = 150\text{ mAdc}$, $V_{CE} = 1.0\text{ Vdc}$) ⁽¹⁾ ($I_C = 500\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$) ⁽¹⁾	hFE MPS2222A only MPS2222 MPS2222A	35 50 75 35 100 50 30 40	— — — — 300 — — —	—
Collector–Emitter Saturation Voltage ⁽¹⁾ ($I_C = 150\text{ mAdc}$, $I_B = 15\text{ mAdc}$) ($I_C = 500\text{ mAdc}$, $I_B = 50\text{ mAdc}$)	MPS2222 MPS2222A MPS2222 MPS2222A	— — — —	0.4 0.3 1.6 1.0	Vdc
Base–Emitter Saturation Voltage ⁽¹⁾ ($I_C = 150\text{ mAdc}$, $I_B = 15\text{ mAdc}$) ($I_C = 500\text{ mAdc}$, $I_B = 50\text{ mAdc}$)	MPS2222 MPS2222A MPS2222 MPS2222A	— 0.6 — —	1.3 1.2 2.6 2.0	Vdc

SMALL–SIGNAL CHARACTERISTICS

Current–Gain — Bandwidth Product ⁽²⁾ ($I_C = 20\text{ mAdc}$, $V_{CE} = 20\text{ Vdc}$, $f = 100\text{ MHz}$)	MPS2222 MPS2222A	f_T	250 300	— —	MHz
Output Capacitance ($V_{CB} = 10\text{ Vdc}$, $I_E = 0$, $f = 1.0\text{ MHz}$)		C_{obo}	—	8.0	pF
Input Capacitance ($V_{EB} = 0.5\text{ Vdc}$, $I_C = 0$, $f = 1.0\text{ MHz}$)	MPS2222 MPS2222A	C_{ibo}	— —	30 25	pF
Input Impedance ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	MPS2222A MPS2222A	h_{ie}	2.0 0.25	8.0 1.25	k Ω
Voltage Feedback Ratio ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	MPS2222A MPS2222A	h_{re}	— —	8.0 4.0	$\times 10^{-4}$
Small–Signal Current Gain ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	MPS2222A MPS2222A	h_{fe}	50 75	300 375	—
Output Admittance ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$) ($I_C = 10\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	MPS2222A MPS2222A	h_{oe}	5.0 25	35 200	μmhos
Collector Base Time Constant ($I_E = 20\text{ mAdc}$, $V_{CB} = 20\text{ Vdc}$, $f = 31.8\text{ MHz}$)	MPS2222A	$r_b \cdot C_C$	—	150	ps
Noise Figure ($I_C = 100\text{ }\mu\text{Adc}$, $V_{CE} = 10\text{ Vdc}$, $R_S = 1.0\text{ k}\Omega$, $f = 1.0\text{ kHz}$)	MPS2222A	NF	—	4.0	dB

SWITCHING CHARACTERISTICS MPS2222A only

Delay Time	(V _{CC} = 30 Vdc, V _{BE(off)} = –0.5 Vdc, I _C = 150 mAdc, I _{B1} = 15 mAdc) (Figure 1)	t _d	—	10	ns
Rise Time		t _r	—	25	ns
Storage Time	(V _{CC} = 30 Vdc, I _C = 150 mAdc, I _{B1} = I _{B2} = 15 mAdc) (Figure 2)	t _s	—	225	ns
Fall Time		t _f	—	60	ns

1. Pulse Test: Pulse Width $\leq 300\text{ }\mu\text{s}$, Duty Cycle $\leq 2.0\%$.

2. f_T is defined as the frequency at which $|h_{fe}|$ extrapolates to unity.

SWITCHING TIME EQUIVALENT TEST CIRCUITS

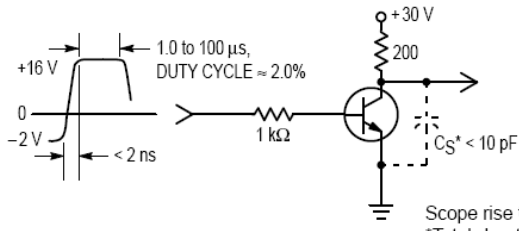


Figure 1. Turn-On Time

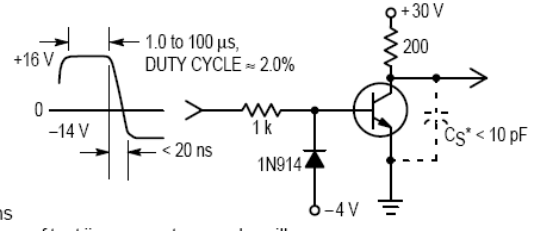


Figure 2. Turn-Off Time

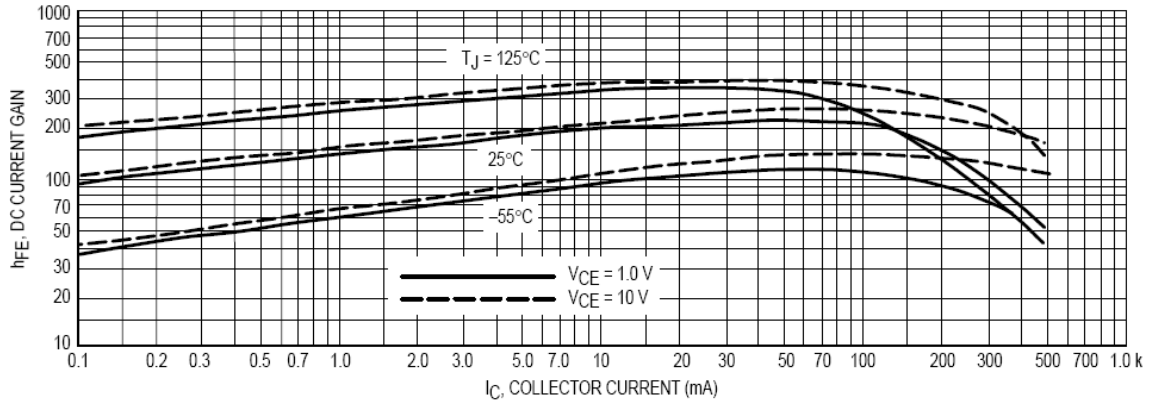


Figure 3. DC Current Gain

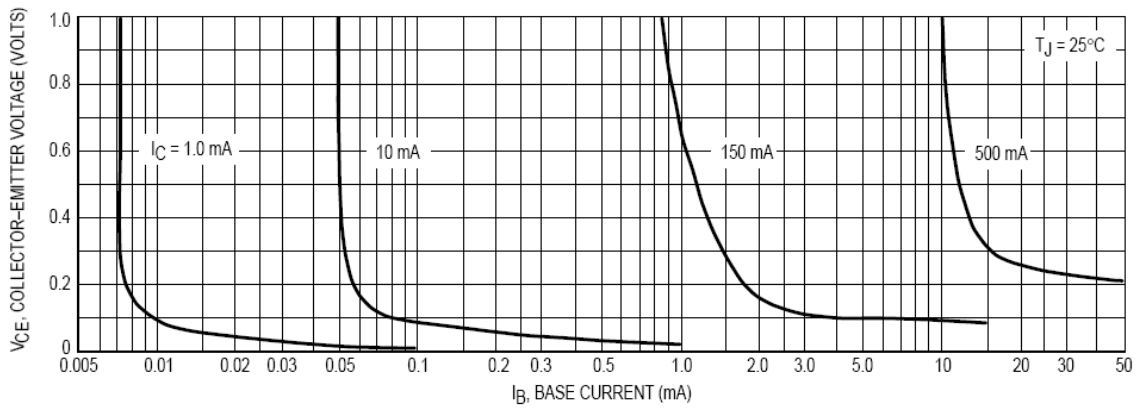


Figure 4. Collector Saturation Region



6-Pin DIP Random-Phase Optoisolators Triac Driver Output (250 Volts Peak)

The MOC3010 Series consists of gallium arsenide infrared emitting diodes, optically coupled to silicon bilateral switch and are designed for applications requiring isolated triac triggering, low-current isolated ac switching, high electrical isolation (to 7500 Vac peak), high detector standoff voltage, small size, and low cost.

- To order devices that are tested and marked per VDE 0884 requirements, the suffix "V" must be included at end of part number. VDE 0884 is a test option.

Recommended for 115 Vac(rms) Applications:

- Solenoid/Valve Controls
- Lamp Ballasts
- Interfacing Microprocessors to 115 Vac Peripherals
- Motor Controls
- Static ac Power Switch
- Solid State Relays
- Incandescent Lamp Dimmers

MAXIMUM RATINGS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Rating	Symbol	Value	Unit
--------	--------	-------	------

INFRARED EMITTING DIODE

Reverse Voltage	V_R	3	Volts
Forward Current — Continuous	I_F	60	mA
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Negligible Power in Transistor Derate above 25°C	P_D	100	mW
		1.33	mW/ $^\circ\text{C}$

OUTPUT DRIVER

Off-State Output Terminal Voltage	V_{DRM}	250	Volts
Peak Repetitive Surge Current (PW = 1 ms, 120 pps)	I_{TSM}	1	A
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	300	mW
		4	mW/ $^\circ\text{C}$

TOTAL DEVICE

Isolation Surge Voltage ⁽¹⁾ (Peak ac Voltage, 60 Hz, 1 Second Duration)	V_{ISO}	7500	Vac(pk)
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	330	mW
		4.4	mW/ $^\circ\text{C}$
Junction Temperature Range	T_J	-40 to +100	$^\circ\text{C}$
Ambient Operating Temperature Range ⁽²⁾	T_A	-40 to +85	$^\circ\text{C}$
Storage Temperature Range ⁽²⁾	T_{stg}	-40 to +150	$^\circ\text{C}$
Soldering Temperature (10 s)	T_L	260	$^\circ\text{C}$

1. Isolation surge voltage, V_{ISO} , is an internal device dielectric breakdown rating.

For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.

2. Refer to Quality and Reliability Section in Opto Data Book for information on test conditions.

Preferred devices are Motorola recommended choices for future use and best overall value.

GlobalOptoisolator is a trademark of Motorola, Inc.

(Replaces MOC3009/D)

MOC3010
[IFT = 15 mA Max]
MOC3011
[IFT = 10 mA Max]
MOC3012*
[IFT = 5 mA Max]

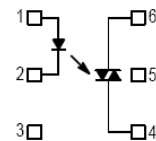
*Motorola Preferred Device

STYLE 6 PLASTIC



**STANDARD THRU HOLE
CASE 730A-04**

COUPLER SCHEMATIC



1. ANODE
2. CATHODE
3. NC
4. MAIN TERMINAL
5. SUBSTRATE
DO NOT CONNECT
6. MAIN TERMINAL

MOC3010 MOC3011 MOC3012

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
INPUT LED					
Reverse Leakage Current ($V_R = 3\text{ V}$)	I_R	—	0.05	100	μA
Forward Voltage ($I_F = 10\text{ mA}$)	V_F	—	1.15	1.5	Volts
OUTPUT DETECTOR ($I_F = 0$ unless otherwise noted)					
Peak Blocking Current, Either Direction (Rated $V_{DRM}^{(1)}$)	I_{DRM}	—	10	100	nA
Peak On-State Voltage, Either Direction ($I_{TM} = 100\text{ mA Peak}$)	V_{TM}	—	1.8	3	Volts
Critical Rate of Rise of Off-State Voltage (Figure 7, Note 2)	dv/dt	—	10	—	$\text{V}/\mu\text{s}$
COUPLED					
LED Trigger Current, Current Required to Latch Output (Main Terminal Voltage = $3\text{ V}^{(3)}$)	I_{FT}				mA
	MOC3010	—	8	15	
	MOC3011	—	5	10	
	MOC3012	—	3	5	
Holding Current, Either Direction	I_H	—	100	—	μA

1. Test voltage must be applied within dv/dt rating.
2. This is static dv/dt . See Figure 7 for test circuit. Commutating dv/dt is a function of the load-driving thyristor(s) only.
3. All devices are guaranteed to trigger at an I_F value less than or equal to max I_{FT} . Therefore, recommended operating I_F lies between max I_{FT} (15 mA for MOC3010, 10 mA for MOC3011, 5 mA for MOC3012) and absolute max I_F (60 mA).

TYPICAL ELECTRICAL CHARACTERISTICS $T_A = 25^\circ\text{C}$

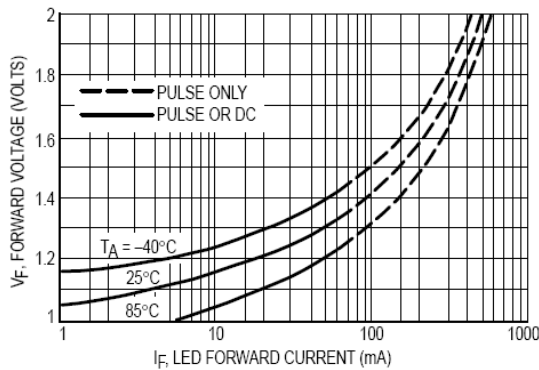


Figure 1. LED Forward Voltage versus Forward Current

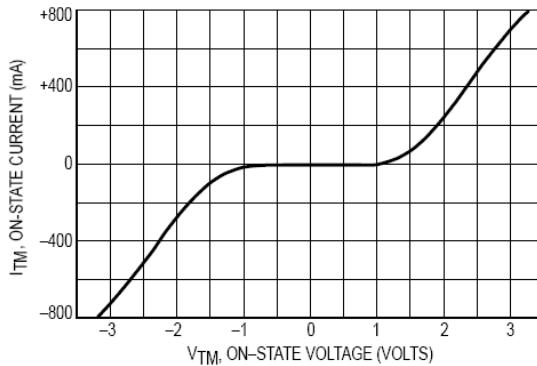


Figure 2. On-State Characteristics

1A, 100V, 0.600 Ohm, N-Channel Power MOSFET

This N-Channel enhancement mode silicon gate power field effect transistor is an advanced power MOSFET designed, tested, and guaranteed to withstand a specified level of energy in the breakdown avalanche mode of operation. All of these power MOSFETs are designed for applications such as switching regulators, switching convertors, motor drivers, relay drivers, and drivers for high power bipolar switching transistors requiring high speed and low gate drive power. These types can be operated directly from integrated circuits.

Formerly developmental type TA17441.

Ordering Information

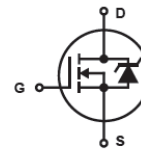
PART NUMBER	PACKAGE	BRAND
IRFD110	HEXDIP	IRFD110

NOTE: When ordering, use the entire part number.

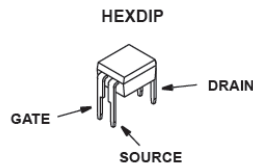
Features

- 1A, 100V
- $r_{DS(ON)} = 0.600\Omega$
- Single Pulse Avalanche Energy Rated
- SOA is Power Dissipation Limited
- Nanosecond Switching Speeds
- Linear Transfer Characteristics
- High Input Impedance
- Related Literature
 - TB334 "Guidelines for Soldering Surface Mount Components to PC Boards"

Symbol



Packaging



IRFD110

Absolute Maximum Ratings $T_C = 25^\circ\text{C}$, Unless Otherwise Specified

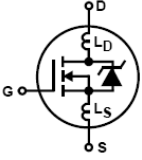
	IRFD110	UNITS
Drain to Source Breakdown Voltage (Note 1)	V_{DS}	100 V
Drain to Gate Voltage ($R_{GS} = 20k\Omega$) (Note 1)	V_{DGR}	100 V
Continuous Drain Current	I_D	1.0 A
Pulsed Drain Current (Note 3)	I_{DM}	8.0 A
Gate to Source Voltage	V_{GS}	± 20 V
Maximum Power Dissipation	P_D	1.0 W
Linear Derating Factor (See Figure 1)		0.008 W/ $^\circ\text{C}$
Single Pulse Avalanche Energy Rating (Note 3)	E_{AS}	19 mJ
Operating and Storage Temperature	T_J, T_{STG}	-55 to 150 $^\circ\text{C}$
Maximum Temperature for Soldering		
Leads at 0.063in (1.6mm) from Case for 10s.	T_L	300 $^\circ\text{C}$
Package Body for 10s, See Techbrief 334	T_{pkg}	260 $^\circ\text{C}$

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

- $T_J = 25^\circ\text{C}$ to 125°C .

Electrical Specifications $T_C = 25^\circ\text{C}$, Unless Otherwise Specified

PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNITS	
Drain to Source Breakdown Voltage	BV_{DSS}	$I_D = 250\mu\text{A}, V_{GS} = 0\text{V}$ (Figure 9)	100	-	-	V	
Gate Threshold Voltage	$V_{GS(TH)}$	$V_{GS} = V_{DS}, I_D = 250\mu\text{A}$	2.0	-	4.0	V	
Zero Gate Voltage Drain Current	I_{DSS}	$V_{DS} = \text{Rated } BV_{DSS}, V_{GS} = 0\text{V}$	-	-	25	μA	
		$V_{DS} = 0.8 \times \text{Rated } BV_{DSS}, V_{GS} = 0\text{V}, T_C = 125^\circ\text{C}$	-	-	250	μA	
On-State Drain Current (Note 2)	$I_{D(ON)}$	$V_{DS} > I_{D(ON)} \times r_{DS(ON)MAX}, V_{GS} = 10\text{V}$	1.0	-	-	A	
Gate to Source Leakage Current	I_{GSS}	$V_{GS} = \pm 20\text{V}$	-	-	± 100	nA	
Drain to Source On Resistance (Note 2)	$r_{DS(ON)}$	$I_D = 0.8\text{A}, V_{GS} = 10\text{V}$ (Figures 7, 8)	-	0.5	0.6	Ω	
Forward Transconductance (Note 2)	g_{fs}	$V_{DS} > I_{D(ON)} \times r_{DS(ON)MAX}, I_D = 0.8\text{A}$ (Figure 11)	0.8	1.2	-	S	
Turn-On Delay Time	$t_{d(ON)}$	$V_{DD} = 0.5 \times \text{Rated } BV_{DSS}, I_D \approx 1.0\text{A}, R_G = 9.1\Omega, R_L = 50\Omega$ MOSFET Switching Times are Essentially Independent of Operating Temperature	-	10	20	ns	
Rise Time	t_r		-	15	25	ns	
Turn-Off Delay Time	$t_{d(OFF)}$		-	15	25	ns	
Fall Time	t_f		-	10	20	ns	
Total Gate Charge (Gate to Source + Gate to Drain)	$Q_{g(TOT)}$	$V_{GS} = 10\text{V}, I_D \approx 1.0\text{A}, V_{DS} = 0.8 \times \text{Rated } BV_{DSS}, I_{g(REF)} = 1.5\text{mA}$ (Figure 13) Gate Charge is Essentially Independent of Operating Temperature	-	5.0	7.0	nC	
Gate to Source Charge	Q_{gs}		-	2.0	-	nC	
Gate to Drain "Miller" Charge	Q_{gd}		-	3.0	-	nC	
Input Capacitance	C_{ISS}		$V_{GS} = 0\text{V}, V_{DS} = 25\text{V}, f = 1\text{MHz}$ (Figure 10)	-	135	-	pF
Output Capacitance	C_{OSS}		-	80	-	pF	
Reverse Transfer Capacitance	C_{RSS}		-	20	-	pF	
Internal Drain Inductance	L_D	Measured from the Drain Lead, 2mm (0.08in) from Package to Center of Die	-	4.0	-	nH	
Internal Source Inductance	L_S	Measured from the Source Lead, 2mm (0.08in) from Header to Source Bonding Pad	-	6.0	-	nH	
							
Thermal Resistance Junction to Ambient	$R_{\theta JA}$	Free Air Operation	-	-	120	$^\circ\text{C/W}$	

IRFD110

Source to Drain Diode Specifications

PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNITS
Continuous Source to Drain Current	I_{SD}	Modified MOSFET Symbol Showing the Integral Reverse P-N Junction Diode	-	-	1.0	A
Pulse Source to Drain Current (Note 4)	I_{SDM}		-	-	8.0	A
Source to Drain Diode Voltage (Note 2)	V_{SD}	$T_J = 25^\circ\text{C}$, $I_{SD} = 1.0\text{A}$, $V_{GS} = 0\text{V}$ (Figure 12)	-	-	2.5	V
Reverse Recovery Time	t_{rr}	$T_J = 150^\circ\text{C}$, $I_{SD} = 1.0\text{A}$, $dI_{SD}/dt = 100\text{A}/\mu\text{s}$	-	100	-	ns
Reverse Recovery Charge	Q_{RR}	$T_J = 150^\circ\text{C}$, $I_{SD} = 1.0\text{A}$, $dI_{SD}/dt = 100\text{A}/\mu\text{s}$	-	0.2	-	μC

NOTES:

2. Pulse test: pulse width $\leq 300\mu\text{s}$, duty cycle $\leq 2\%$.
3. $V_{DD} = 25\text{V}$, starting $T_J = 25^\circ\text{C}$, $L = 28.5\text{mH}$, $R_G = 25\Omega$, peak $I_{AS} = 1.0\text{A}$.
4. Repetitive rating: pulse width limited by maximum junction temperature.

Typical Performance Curves Unless Otherwise Specified

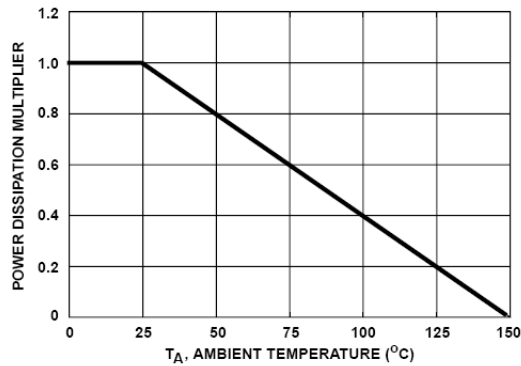


FIGURE 1. NORMALIZED POWER DISSIPATION vs AMBIENT TEMPERATURE

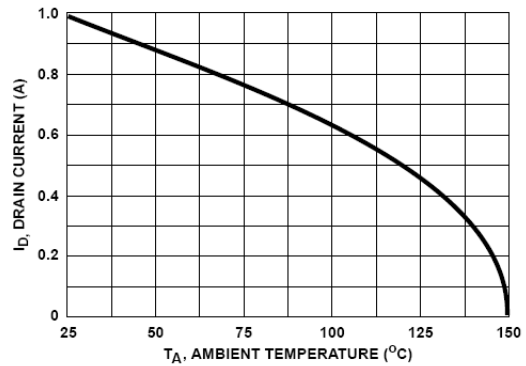


FIGURE 2. MAXIMUM CONTINUOUS DRAIN CURRENT vs AMBIENT TEMPERATURE

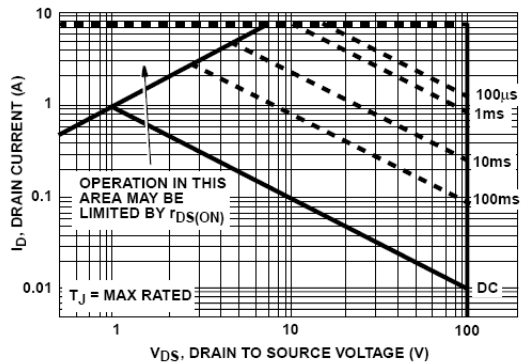


FIGURE 3. FORWARD BIAS SAFE OPERATING AREA

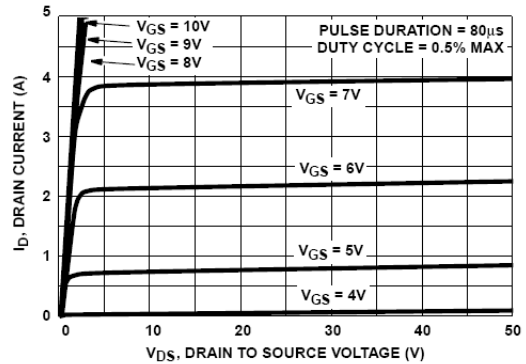


FIGURE 4. OUTPUT CHARACTERISTICS

Zener 1N4733A



1N4728A THRU 1N4764A
1W ZENER DIODE



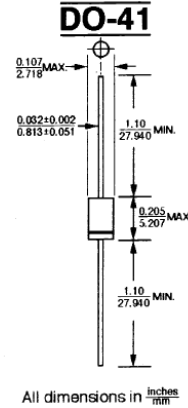
FEATURES

- * 3.3 thru 100 volt voltage range
 - * High surge current rating
 - * Higher voltages available, see 1EZ series
- MECHANICAL CHARACTERISTICS**
- * CASE: Molded encapsulation, axial lead package (DO - 41).
 - * FINISH: Corrosion resistant. Leads are solderable.
 - * THERMAL RESISTANCE: 45°C/Watt junction to lead at 0.375 inches from body.
 - * POLARITY: banded end is cathode.
 - * WEIGHT: 0.4 grams (Typical).

MAXIMUM RATINGS

Junction and Storage temperature: - 65°C to + 200°C
 DC Power Dissipation: 1 Watt
 Power Derating: 10mW/°C, from 100°C
 Forward Voltage @ 200mA: 1.2 Volts

VOLTAGE RANGE
3.3 to 100 Volts



* **ELECTRICAL CHARACTERISTICS @ 25°C**

JEDEC TYPE NUMBER (Note 1)	ZENER VOLTAGE (VZ) (Note 4)	TEST CURRENT IZT	MAXIMUM DYNAMIC IMPEDANCE ZZT @ IZT (Note 2)	MAXIMUM REVERSE CURRENT IR @ VR	TEST VOLTAGE (VR)	MAXIMUM REGULATOR CURRENT (IZM) TA = 50°C	MAXIMUM KNEE IMPEDANCE (ZK @ IZK) (Note 2)	TEST CURRENT (IZK)	MAXIMUM (SURGE) CURRENT (IS) (Note 3)
	VOLTS	mA	OHMS	µA	VOLTS	mA	OHMS	mA	mA
1N4728A	3.3	76	10	100	1	276	400	1.0	1380
1N4728A	3.6	69	10	100	1	252	400	1.0	1250
1N4730A	3.9	64	9	50	1	234	400	1.0	1190
1N4731A	4.3	58	9	10	1	217	400	1.0	1070
1N4732A	4.7	53	8	10	1	193	500	1.0	970
1N4733A	5.1	49	7	10	1	178	550	1.0	890
1N4734A	5.6	45	5	10	2	162	600	1.0	810
1N4735A	6.2	41	2	10	3	146	700	1.0	730
1N4736A	6.8	37	3.5	10	4	133	700	1.0	660
1N4737A	7.5	34	4.0	10	5	121	700	0.5	605
1N4738A	8.2	31	4.5	10	6	110	700	0.5	560
1N4739A	9.1	28	5.0	10	7	100	700	0.5	500
1N4740A	10	25	7	10	7.6	91	700	0.25	454
1N4741A	11	23	8	5	8.4	83	700	0.25	414
1N4742A	12	21	9	5	9.1	76	700	0.25	380
1N4743A	13	19	10	5	9.9	69	700	0.25	344
1N4744A	15	17	14	5	11.4	61	700	0.25	304
1N4745A	16	15.5	16	5	12.2	57	700	0.25	285
1N4746A	18	14	20	5	13.7	50	750	0.25	250
1N4747A	20	12.5	22	5	15.2	45	750	0.25	225
1N4748A	22	11.5	23	5	16.7	41	750	0.25	205
1N4749A	24	10.5	25	5	18.2	38	750	0.25	190
1N4750A	27	9.5	35	5	20.6	34	750	0.25	170
1N4751A	30	8.5	40	5	22.8	30	1000	0.25	150
1N4752A	33	7.5	45	5	25.1	27	1000	0.25	135
1N4753A	36	7.0	50	5	27.4	25	1000	0.25	125
1N4754A	39	6.5	60	5	29.7	23	1000	0.25	115
1N4755A	43	6.0	70	5	32.7	22	1500	0.25	110
1N4756A	47	5.5	80	5	35.8	19	1500	0.25	95
1N4757A	51	5.0	96	5	38.8	18	1500	0.25	90
1N4758A	56	4.5	110	5	42.6	16	2000	0.25	80
1N4759A	62	4.0	125	5	47.1	14	2000	0.25	70
1N4760A	68	3.7	150	5	51.7	13	2000	0.25	65
1N4761A	75	3.3	175	5	56.0	12	2000	0.25	60
1N4762A	82	3.0	200	5	62.2	11	3000	0.25	55
1N4763A	91	2.8	250	5	69.2	10	3000	0.25	50
1N4764A	100	2.5	350	5	76.0	9	3000	0.25	45

NOTE 1 The JEDEC type numbers shown have a 5% tolerance on nominal zener voltage. No suffix signifies a 10% tolerance, C signifies 2%, and D signifies 1% tolerance.

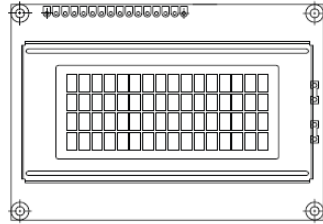
NOTE 2 The Zener impedance is derived from the 60 Hz ac voltage, which results when an ac current having an rms value equal to 10% of the DC Zener current (IZT or IZK) is superimposed on IZT or IZK. Zener impedance is measured at two points to insure a sharp knee on the breakdown curve and eliminate unstable units.

NOTE 3 The zener surge current is measured at 25°C ambient using a 1/2 square wave or equivalent sine wave pulse 1/120 second duration superimposed on IZT.

NOTE 4 Voltage measurements to be performed 90 seconds after application of DC current.



16 x 4 Character LCD



FEATURES

- 5 x 8 dots includes cursor
- Built-in controller (KS 0066 or Equivalent)
- + 5V power supply (Also available for + 3V)
- 1/16 duty cycle
- B/L to be driven by pin 1, pin 2 or pin 15, pin 16 or A and K (LED)
- N.V. optional for + 3V power supply

MECHANICAL DATA		
ITEM	STANDARD VALUE	UNIT
Module Dimension	87.0 x 60.0	mm
Viewing Area	62.0 x 26.0	mm
Dot Size	0.55 x 0.55	mm
Character Size	2.95 x 4.75	mm

ABSOLUTE MAXIMUM RATING					
ITEM	SYMBOL	STANDARD VALUE			UNIT
		MIN.	TYP.	MAX.	
Power Supply	VDD-VSS	- 0.3	-	7.0	V
Input Voltage	VI	- 0.3	-	VDD	V

NOTE: VSS = 0 Volt, VDD = 5.0 Volt

ELECTRICAL SPECIFICATIONS						
ITEM	SYMBOL	CONDITION	STANDARD VALUE			UNIT
			MIN.	TYP.	MAX.	
Input Voltage	VDD	VDD = + 5V	4.7	5.0	5.3	V
		VDD = + 3V	2.7	3.0	5.3	V
Supply Current	IDD	VDD = + 5V	-	1.0	1.2	mA
Recommended LC Driving Voltage for Normal Temp. Version Module	VDD - V0	- 20 °C	5.0	5.1	5.7	V
		0 °C	4.6	4.8	5.2	
		25 °C	4.1	4.5	4.7	
		50 °C	3.9	4.2	4.5	
		70 °C	3.7	3.9	4.3	
LED Forward Voltage	VF	25 °C	-	4.2	4.6	V
LED Forward Current	IF	25 °C	-	220	440	mA
EL Power Supply Current	IEL	Vel = 110VAC:400Hz	-	-	5.0	mA

DISPLAY CHARACTER ADDRESS CODE:																
Display Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DD RAM Address	00	01														0F
DD RAM Address	40	41														4F
DD RAM Address	10	11														1F
DD RAM Address	50	51														5F



Intelligent Alphanumeric Application Notes

Character Position and Character Address

For each device, the relationship between character position and character address is straightforward. A hexadecimal code for each character position in each device is given in the following charts. Character positions are numbered from left to right beginning in the top left corner as you view the device from the front.

AND671

Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F

AND491, 481, 471

Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Character Position	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	DD RAM (Hex) Add.	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

AND501

Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
Character Position	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
DD RAM (Hex) Add.	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53

AND731

Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Character Position	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	DD RAM (Hex) Add.	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
Character Position	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	Character Position	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
DD RAM (Hex) Add.	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	DD RAM (Hex) Add.	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F

AND771

Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17
Character Position	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
DD RAM (Hex) Add.	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57

AND591

Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0E	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
Character Position	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
DD RAM (Hex) Add.	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67

AND721

Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
Character Position	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
DD RAM (Hex) Add.	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53
Character Position	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
DD RAM (Hex) Add.	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
Character Position	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
DD RAM (Hex) Add.	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	6	62	63	64	65	66	67

AND791

Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0E	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
Character Position	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
DD RAM (Hex) Add.	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67
Character Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
DD RAM (Hex) Add.	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0E	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
Character Position	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
DD RAM (Hex) Add.	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67

Note: Address locations for Lines 1 & 2 are controlled by E1, and lines 3 & 4 are controlled by E2.



Intelligent Alphanumeric Application Notes

Timing

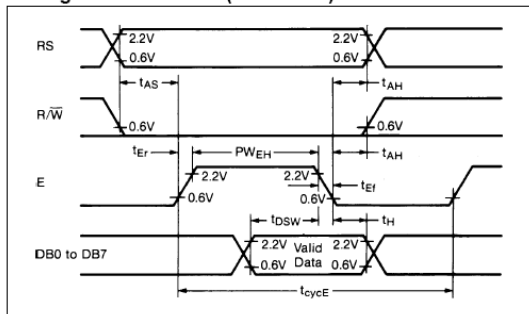
Timing Characteristics (TA = 25°C) Data Write

Item	Symbol	Value		Unit
		Min.	Max.	
Enable Cycle Time	t_{CYCE}	1000	-	
Enable Pulse Width	PW_{EH}	450		
Enable Rise/Fall Time	t_{Er}, t_{Ef}		25	
Set Up Time	t_{AS}	140		ns
Address Hold Time	t_{AH}	10		
Data Set Up Time	t_{DSW}	195		
Data Hold Time	t_H	10		

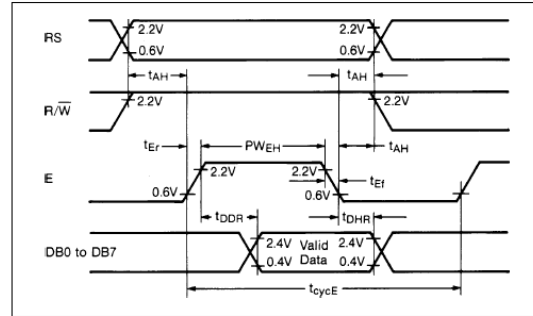
Data Read

Item	Symbol	Value		Unit
		Min.	Max.	
Enable Cycle Time	t_{CYCE}	1000		ns
Enable Pulse Width	PW_{EH}	450		
Enable Rise/Fall Time	t_{Er}, t_{Ef}		25	
Set Up Time	t_{AS}	140		
Address Hold Time	t_{AH}	10		
Data Delay Time	t_{DDR}		320	
Data Hold Time	t_{DHR}	20		

Timing Characteristics (Data Write)



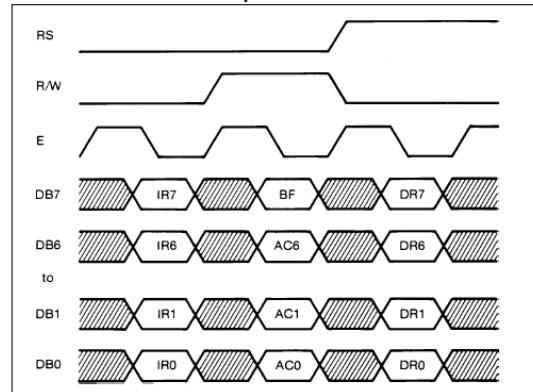
Timing Characteristics (Data Read)



Data Transfer Example

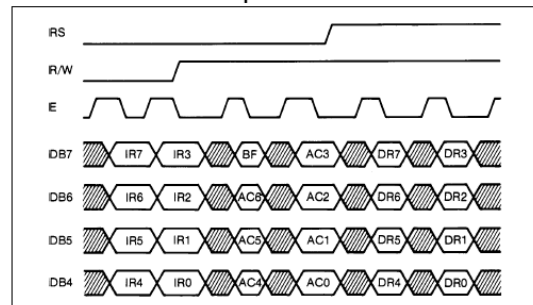
Data can be sent in either two cycles of 4-bit data or one cycle of 8-bit data, a scheme that suits 4-bit or 8-bit CPUs. Data that is 8-bits long is transferred using 8 data lines of DB0 to DB7.

8-Bit Data Transfer Example



Data that is 4 bits long is transferred by using only 4 lines of DB7 to DB4—DB3 to DB0 are not used. Data transfer between the module and a 4-bit CPU is completed when the high order 4 bits are transferred first, followed by the low order 4 bits.

4-Bit Data Transfer Example





Intelligent Alphanumeric Application Notes

Command List

Command	Command Code										Description	Execution Time (Max.) ⁽¹⁾	Execution Time (Max.) ⁽²⁾
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0			
Clear display ⁽³⁾	0	0	0	0	0	0	0	0	0	1	Clear display and return cursor to home position (Address 0).	1.64ms	4.9ms
Return Home	0	0	0	0	0	0	0	0	1	X	Return cursor to home position (Address 0). Also return display being shifted to original position. DD RAM contents remain the same.	1.64ms	4.8ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Set cursor move direction and specify whether to shift display. These operations are performed during data write.	40μs	120μs
Display ON/OFF Control	0	0	0	0	0	0	1	D	C	B	Set ON/OFF of entire display (D), dresser ON/OFF (C), and blinking of cursor position B	40μs	120μs
Cursor and Display Shift ⁽⁴⁾	0	0	0	0	0	1	S/C	R/L	X	X	Move cursor and shift display without changing DD RAM contents.	40μs	120μs
Function Set	0	0	0	0	1	DL	N	F	X	X	Set interface data length (DL) number of display lines (L) and character font (F).	40μs	120μs
Set RAM Address	0	0	0	1	ACG						Set CG RAM address, CG RAM data is sent and received after this setting.	40μs	120μs
Set DD RAM Address	0	0	1	ADD						Set DD RAM address, DD RAM data is sent and received after this setting.	40μs	120μs	
Read Busy Flag & Address	0	1	BF	AC						Read Busy flag (BF) indicating internal operation is being performed and reads address counter contents.	40μs	120μs	
Write Data to CG or DD RAM	1	0	Write Data						Write Data from DD RAM or CG RAM.	40μs	120μs		
Read Data to CG or DD RAM	1	1	Read Data						Read Data from DD RAM or CG RAM.	40μs	120μs		
	I/D = 1 : Increment I/D = 0 Decrement S = 1 : Accompanies display shift S/C = 1 : Display shift S/C = 0 Cursor move R/L = 1 : Shift to the right R/L = 0 : Shift to the left DL = 1 : 8 bits DL = 0: 4 bits N = 1 : 2 lines N = 0: 1 line F = 1 : 5 x 10 dots F = 0: 5 x 7 dots BF = 1 : Internally operating BF = 0 : Can accept instruction										DD RAM : Display Data RAM CG RAM : Character Gen RAM ACG : CG RAM Address ADD : DD RAM Address corresponds to Cursor Address AC : Address Counter used for DD and CG RAM Address.		

X = Don't Care

Notes:

1. Applies to AND491, AND481, AND491, and AND501.
2. Applies to AND591, AND731, AND721, AND771.
3. The repeat time interval of command Clear Display must be 13ms minimum (5 x 7 dot font) and 18ms minimum (5 x 10 dot font).
4. Commands "Cursor and Display Shift" are invalid for the AND671.



Function of Registers

The following paragraphs describe the function of the registers.

Instruction Register and Data Register

The built-in controller has two 8-bit registers, an Instruction Register (IR) and a Data Register (DR). The IR stores commands such as display clear and cursor shift, or address information of display data RAM (DD RAM), and character generator RAM (CG RAM). The IR can be written to by a CPU, but a CPU cannot read this register.

The DR temporarily stores data to be written into the DD RAM or the CG RAM. Data written into the DR is automatically sent to the DD RAM or the CG RAM as an internal operation. The DR is also used for data storage when reading data from the DD RAM or the CG RAM. When address information is written into the IR, data is transferred to the DR from the DD RAM or the CG RAM as an internal operation. Then, the CPU reads the DR and data transfer is completed. After the CPU reads the DR, data of the DD RAM or the CG RAM at the next address is sent to DR for the next reading. Register Selector (RS) signals select these two registers.

Register Selection

RS	R/W	Operation	Enable (E)
0	0	Write commands to IR.	
0	1	Read of a Busy Flag (DB7) and Address Counter (DB0 to DB6)	
1	0	DR Write as internal operations (DR → DD or CG RAM)	
1	1	DR Read as internal operations (DD or CG RAM → DR)	

Busy Flag (BF)

When the Busy Flag is "1", the LCD module is in the internal operation mode, and the next instruction is not accepted at this time. As shown in the "Command List" on page 5 of this section, the Busy Flag is shown in DB7 when RS = 0 and R/W = 1. The next instruction must be written after checking that the Busy Flag is "0".

Address Counter (AC)

The address counter (AC) assigns DD and CG RAM address. When an instruction for address setting is written in IR, the address information is sent from IR to AC.

Selection of either the DD or CG RAM is also determined by an instruction. After writing into (or reading from) DD or CG RAM display data, AC is automatically incremented by 1 (or decremented by 1). Data in address counters (AC) are in DB6 to DBO when RS = 0 and R/W = 1, as shown in the table entitled "Command List" on page 5 of this section.

Display Data RAM (DD RAM)

The display data RAM (DD RAM) stores display data represented in 8-bit character codes. The relationship between the DD RAM address and display position on the LCD Display is described by a series of tables under the paragraph "Character Position and Character Address" on page 3 of this section.

Commands

The command code is the signal through which the LCD module is accessed through the CPU. The LCD module begins operation upon receipt of the code input. Because the internal processing operation of the LCD module is started with a timing that does not affect the LCD display, the busy status continues longer than the CPU cycle time.

Under the busy status (when the busy flag is set to "1"), the LCD module does not execute any commands other than the busy flag read. Accordingly, the CPU has to verify that the busy flag is set to "0" prior to the input of the command code.

Clear Display

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	0	0	0	0	1

Write space code "20" (hexadecimal) into all the DD RAM addresses. The cursor returns to address "0" (DD RAM Address = "00H") and the display, if it has been shifted, returns to the original position. In other words, the display disappears and the cursor goes to the left edge of the first line.

Return Home

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	0	0	0	1	X

Return the cursor to character position 1 (DD RAM Address = "00H") and returns the display to the original position if it has been shifted (S in the instruction register is 1). The DD RAM contents remain unchanged.

X = Don't care

Entry Mode Set

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	0	0	1	I/D	S

I/D: Increment (I/D = 1) or decrement (I/D = 0) the DD RAM address by one upon writing a character code into the DD RAM or reading a character code from the DD RAM. The cursor moves to the right when I/D = 1, and to the left when I/D = 0.

S: When writing to the DD RAM, shift the entire display to the right (when I/D = 0, S = 1) or to the left (when I/D = 1, S = 1). Therefore, the cursor looks as if it stood still and only the display moves. Display is not shifted when reading from the DD RAM. Display is not shifted when S = 0.



Intelligent Alphanumeric Application Notes

Display ON/OFF Control

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	0	1	D	C	B

- D: Display is turned ON when D = 1 and OFF when D = 0. When display is turned off due to D = 0, the display data remains in the DD RAM and they can be displayed immediately by setting D = 1.
- C: The cursor is displayed when C = 1 and not displayed when C = 0. Even if the cursor disappears, the function of I/D does not change during "display data write." The cursor is displayed at the 8th line when the 5 x 7 dots character font is selected.
- B: The character at the cursor position blinks when B = 1. The blink is done by switching between all black dots and display characters at 0.4 second interval. The cursor and the blink can be set concurrently.

Cursor or Display Shift

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	1	S/C	R/L	X	X

Shift the cursor position or display position to the right or the left without writing or reading the display data. This function can be used for correction or search of display.

S/C	R/L	Function
0	0	Shift the cursor position to the left. (AC is decremented by one.)
0	1	Shift the cursor position to the right. (AC is incremented by one.)
1	0	Shift the entire display to the left. The cursor follows the display shift.
1	1	Shift the entire display to the right. The Cursor follows the display shift.

Function Set

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	0	1	DL	N	F	X	X

- DL: Sets the interface data length. Data is sent or received in 8-bit length (DB7 to DB0) when DL = 1 and 4-bit length (DB7 to DB4) when DL = 0. When 4-bit length is selected, data must be sent or received in two cycles.
- N: Set number of display lines.
- F: Set character font. The 5 x 7 dots character font is selected when F = 0. While 5 x 10 dots character font is selected when F = 1 and N = 0.

Module Type Number

N	F	No. of Display Lines	Character Font	Duty Ratio	AND Model No.
1	0	2	5 x 7 Dots	1/16	AND471, AND481, AND491, AND501, AND591, AND671, AND771
1	0	4	5 x 7 Dots	1/16	AND721, AND731, AND791

Set CG RAM Address

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	1	A	A	A	A	A	A

Set the CG RAM address to a binary number of AAAAAA in the address counter. After execution of this instruction, all the data from MPU is written into the CG RAM and all the data is read from CG RAM.

Set DD RAM Address

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	1	An	A	A	A	A	A	A

Set the DD RAM address to a binary number of AnAAAAA in the address counter (An = 0 for the first line, An = 1 for the second line). After execution of this instruction, all the data from MPU is written into the DD RAM and all the data is read from DD RAM.

Read Busy Flag and Address

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	1	BF	A	A	A	A	A	A	A

Read Busy Flag (BF) and the value of the address counter (AAAAAA). The condition BF = 1 indicates that an internal operation is going on and the next command is not accepted until BF becomes "0." You must check the BF status before the next write operation. The address counter generates the CG or DD RAM address.

Write Data to CG RAM or DD RAM

	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	1	D	D	D	D	D	D	D	D	D

Write binary 8-bit data DDDDDDDD to the CG RAM or the DD RAM. Whether the CG RAM or the DD RAM is to be written is determined by the previous designation (CG RAM address setting or DD RAM address setting). After writing, the address is automatically incremented or decremented by one according to entry mode. Display shift also follows the entry mode.



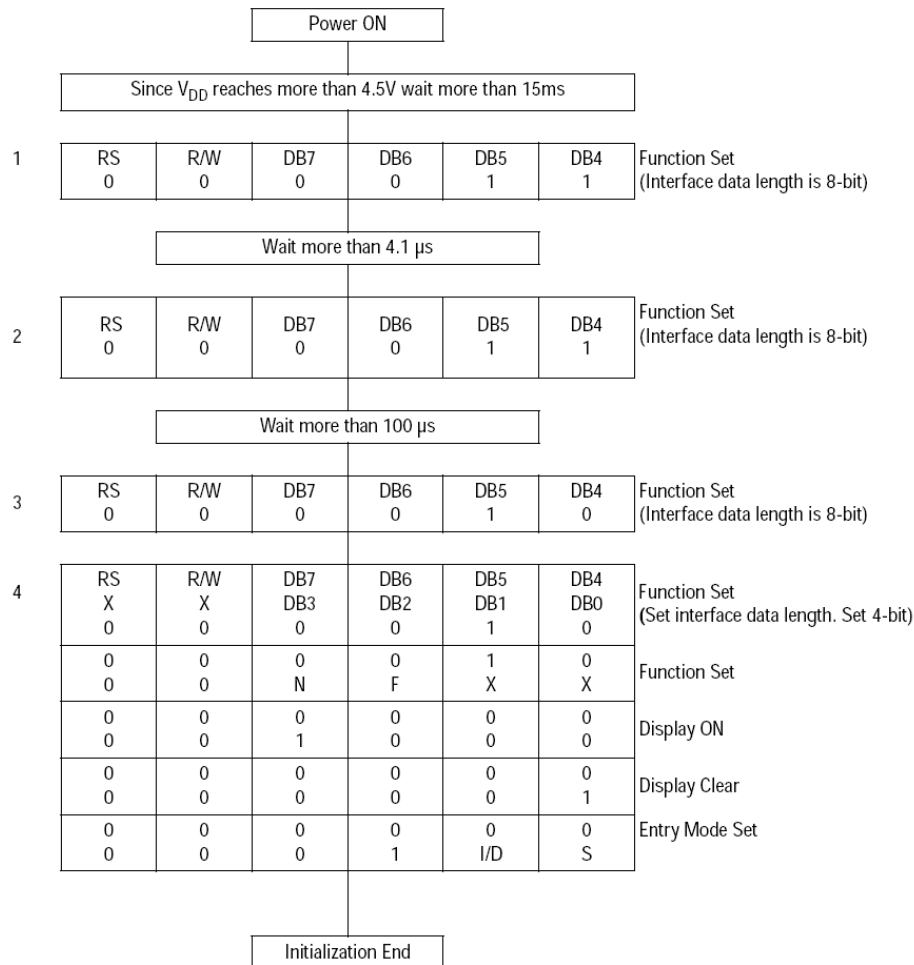
Character Pattern and Character Code

Upper 4 bit Lower 4 bit	0000	0010	0011	0100	0101	0110	0111	1010	1011	1100	1101	1110	1111
XXXX0000	CG RAM (1)		0	a	P	\	P	-	9	3	0	p	
XXXX0001	(2)	!	1	0	a	4	7	7	4	a	q		
XXXX0010	(3)	"	2	R	b	r	"	/	w	x	p	0	
XXXX0011	(4)	#	3	C	S	c	s	.	7	t	e	e	*
XXXX0100	(5)	\$	4	O	T	d	t	.	I	k	t	p	a
XXXX0101	(6)	%	5	E	U	e	u	.	7	+	1	e	0
XXXX0110	(7)	&	6	F	U	f	u	9	n	c	a	p	z
XXXX0111	(8)	'	7	G	W	g	w	7	7	7	9	g	n
XXXX1000	(1)	<	8	H	X	h	x	4	7	7	7	7	x
XXXX1001	(2)	>	9	I	Y	i	y	7	7	7	7	7	y
XXXX1010	(3)	*	:	J	Z	j	z	7	7	7	7	7	7
XXXX1011	(4)	+	:	K	L	k	l	7	7	7	7	7	7
XXXX1100	(5)	,	<	L	*	l	l	7	7	7	7	7	7
XXXX1101	(6)	-	=	M	I	m	i	7	7	7	7	7	7
XXXX1110	(7)	.	>	N	^	n	7	7	7	7	7	7	7
XXXX1111	(8)	/	?	0	_	0	7	7	7	7	7	7	7



Manual Initialization Procedure

The following diagram applies when the interface data length is 4-bits.



X = Don't care

Notes:

1. Before initialize step 1, 2, and 3, cannot check busy flag.
2. After initialize step 4, cannot change function set mode, number of display lines and character font.



PIC18FXX2

28/40-pin High Performance, Enhanced FLASH Microcontrollers with 10-Bit A/D

High Performance RISC CPU:

- C compiler optimized architecture/instruction set
 - Source code compatible with the PIC16 and PIC17 instruction sets
- Linear program memory addressing to 32 Kbytes
- Linear data memory addressing to 1.5 Kbytes

Device	On-Chip Program Memory		On-Chip RAM (bytes)	Data EEPROM (bytes)
	FLASH (bytes)	# Single Word Instructions		
PIC18F242	16K	8192	768	256
PIC18F252	32K	16384	1536	256
PIC18F442	16K	8192	768	256
PIC18F452	32K	16384	1536	256

- Up to 10 MIPs operation:
 - DC - 40 MHz osc./clock input
 - 4 MHz - 10 MHz osc./clock input with PLL active
- 16-bit wide instructions, 8-bit wide data path
- Priority levels for interrupts
- 8 x 8 Single Cycle Hardware Multiplier

Peripheral Features:

- High current sink/source 25 mA/25 mA
- Three external interrupt pins
- Timer0 module: 8-bit/16-bit timer/counter with 8-bit programmable prescaler
- Timer1 module: 16-bit timer/counter
- Timer2 module: 8-bit timer/counter with 8-bit period register (time-base for PWM)
- Timer3 module: 16-bit timer/counter
- Secondary oscillator clock option - Timer1/Timer3
- Two Capture/Compare/PWM (CCP) modules. CCP pins that can be configured as:
 - Capture input: capture is 16-bit, max. resolution 6.25 ns ($T_{CY}/16$)
 - Compare is 16-bit, max. resolution 100 ns (T_{CY})
 - PWM output: PWM resolution is 1- to 10-bit, max. PWM freq. @: 8-bit resolution = 156 kHz
10-bit resolution = 39 kHz
- Master Synchronous Serial Port (MSSP) module, Two modes of operation:
 - 3-wire SPI™ (supports all 4 SPI modes)
 - I²C™ Master and Slave mode

Peripheral Features (Continued):

- Addressable USART module:
 - Supports RS-485 and RS-232
- Parallel Slave Port (PSP) module

Analog Features:

- Compatible 10-bit Analog-to-Digital Converter module (A/D) with:
 - Fast sampling rate
 - Conversion available during SLEEP
 - Linearity ≤ 1 LSb
- Programmable Low Voltage Detection (PLVD)
 - Supports interrupt on-Low Voltage Detection
- Programmable Brown-out Reset (BOR)

Special Microcontroller Features:

- 100,000 erase/write cycle Enhanced FLASH program memory typical
- 1,000,000 erase/write cycle Data EEPROM memory
- FLASH/Data EEPROM Retention: > 40 years
- Self-reprogrammable under software control
- 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 including:
 - 4X Phase Lock Loop (of primary oscillator)
 - Secondary Oscillator (32 kHz) clock input
- Single supply 5V In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins

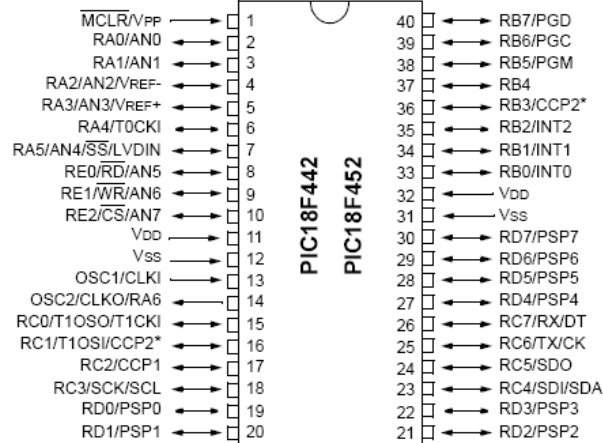
CMOS Technology:

- Low power, high speed FLASH/EEPROM technology
- Fully static design
- Wide operating voltage range (2.0V to 5.5V)
- Industrial and Extended temperature ranges
- Low power consumption:
 - < 1.6 mA typical @ 5V, 4 MHz
 - 25 μ A typical @ 3V, 32 kHz
 - < 0.2 μ A typical standby current

PIC18FXX2

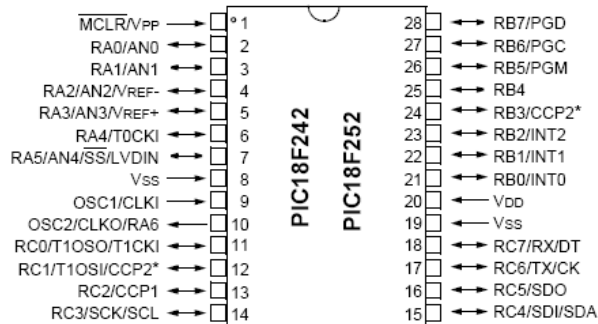
Pin Diagrams (Cont.'d)

DIP



Note: Pin compatible with 40-pin PIC16C7X devices.

DIP, SOIC



* RB3 is the alternate pin for the CCP2 pin multiplexing.

PIC18FXX2

1.0 DEVICE OVERVIEW

This document contains device specific information for the following devices:

- PIC18F242
- PIC18F252
- PIC18F442
- PIC18F452

These devices come in 28-pin and 40/44-pin packages. The 28-pin devices do not have a Parallel Slave Port (PSP) implemented and the number of Analog-to-Digital (A/D) converter input channels is reduced to 5. An overview of features is shown in Table 1-1.

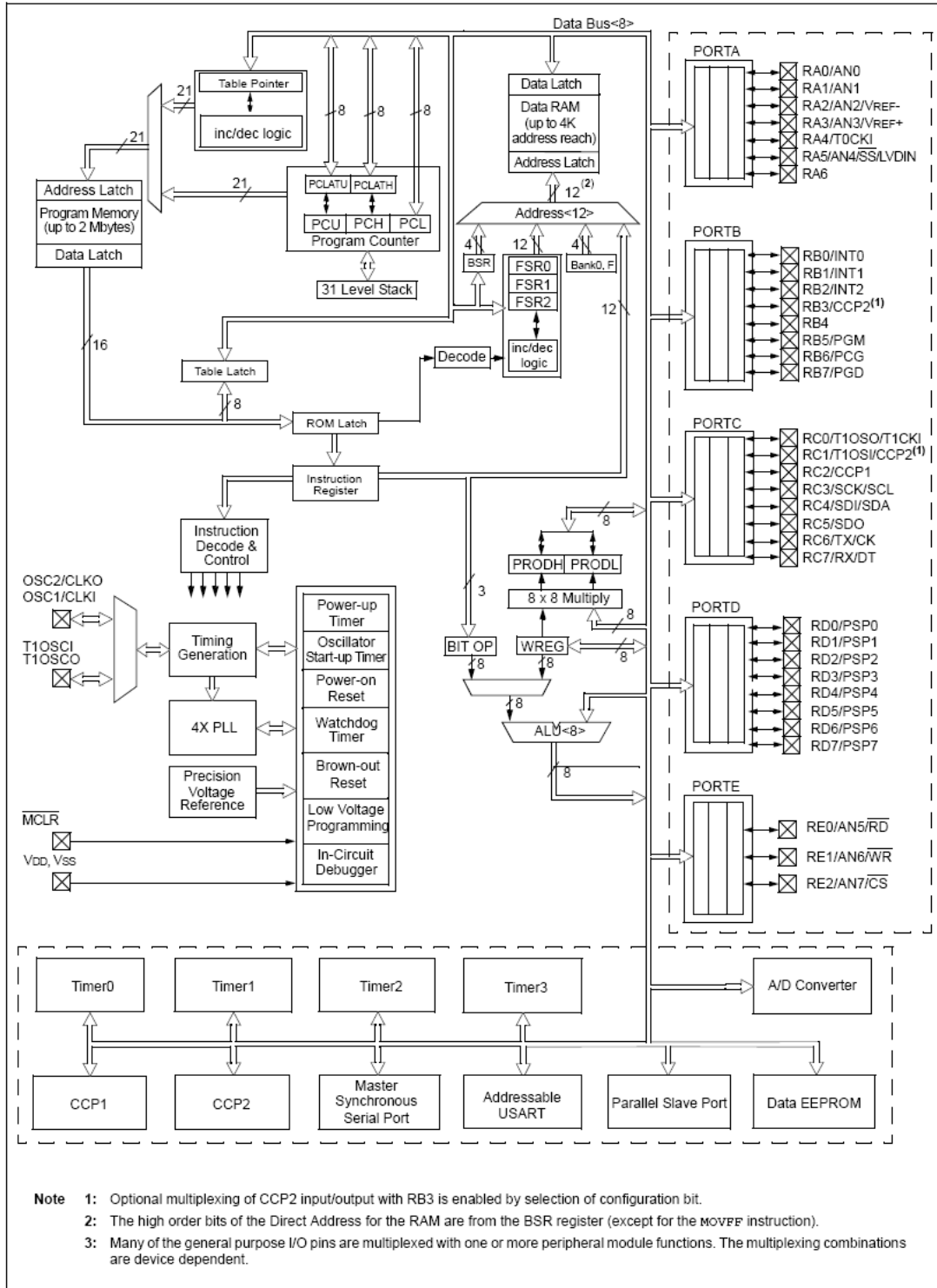
The following two figures are device block diagrams sorted by pin count: 28-pin for Figure 1-1 and 40/44-pin for Figure 1-2. The 28-pin and 40/44-pin pinouts are listed in Table 1-2 and Table 1-3, respectively.

TABLE 1-1: DEVICE FEATURES

Features	PIC18F242	PIC18F252	PIC18F442	PIC18F452
Operating Frequency	DC - 40 MHz	DC - 40 MHz	DC - 40 MHz	DC - 40 MHz
Program Memory (Bytes)	16K	32K	16K	32K
Program Memory (Instructions)	8192	16384	8192	16384
Data Memory (Bytes)	768	1536	768	1536
Data EEPROM Memory (Bytes)	256	256	256	256
Interrupt Sources	17	17	18	18
I/O Ports	Ports A, B, C	Ports A, B, C	Ports A, B, C, D, E	Ports A, B, C, D, E
Timers	4	4	4	4
Capture/Compare/PWM Modules	2	2	2	2
Serial Communications	MSSP, Addressable USART	MSSP, Addressable USART	MSSP, Addressable USART	MSSP, Addressable USART
Parallel Communications	—	—	PSP	PSP
10-bit Analog-to-Digital Module	5 input channels	5 input channels	8 input channels	8 input channels
RESETS (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST)
Programmable Low Voltage Detect	Yes	Yes	Yes	Yes
Programmable Brown-out Reset	Yes	Yes	Yes	Yes
Instruction Set	75 Instructions	75 Instructions	75 Instructions	75 Instructions
Packages	28-pin DIP 28-pin SOIC	28-pin DIP 28-pin SOIC	40-pin DIP 44-pin PLCC 44-pin TQFP	40-pin DIP 44-pin PLCC 44-pin TQFP

PIC18FXX2

FIGURE 1-2: PIC18F4X2 BLOCK DIAGRAM



PIC18FXX2

TABLE 1-3: PIC18F4X2 PINOUT I/O DESCRIPTIONS

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	DIP	PLCC	TQFP			
MCLR/VPP MCLR VPP	1	2	18	I I	ST ST	Master Clear (input) or high voltage ICSP programming enable pin. Master Clear (Reset) input. This pin is an active low RESET to the device. High voltage ICSP programming enable pin.
NC	—			—	—	These pins should be left unconnected.
OSC1/CLKI OSC1 CLKI	13	14	30	I I	ST CMOS	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. ST buffer when configured in RC mode, CMOS otherwise. External clock source input. Always associated with pin function OSC1. (See related OSC1/CLKI, OSC2/CLKO pins.)
OSC2/CLKO/RA6 OSC2 CLKO RA6	14	15	31	O O I/O	— — TTL	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate. General Purpose I/O pin.
RA0/AN0 RA0 AN0 RA1/AN1 RA1 AN1 RA2/AN2/VREF- RA2 AN2 VREF- RA3/AN3/VREF+ RA3 AN3 VREF+ RA4/T0CKI RA4 T0CKI RA5/AN4/SS/LVDIN RA5 AN4 SS LVDIN RA6	2	3	19	I/O I I/O I I/O I I I/O I I I/O I I I	TTL Analog TTL Analog TTL Analog Analog TTL Analog Analog ST/OD ST TTL Analog ST Analog	PORTA is a bi-directional I/O port. Digital I/O. Analog input 0. Digital I/O. Analog input 1. Digital I/O. Analog input 2. A/D Reference Voltage (Low) input. Digital I/O. Analog input 3. A/D Reference Voltage (High) input. Digital I/O. Open drain when configured as output. Timer0 external clock input. Digital I/O. Analog input 4. SPI Slave Select input. Low Voltage Detect Input. (See the OSC2/CLKO/RA6 pin.)

Legend: TTL = TTL compatible input
ST = Schmitt Trigger input with CMOS levels
O = Output
OD = Open Drain (no P diode to VDD)

CMOS = CMOS compatible input or output
I = Input
P = Power

PIC18FXX2

TABLE 1-3: PIC18F4X2 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	DIP	PLCC	TQFP			
RB0/INT0 RB0 INT0	33	36	8	I/O I	TTL ST	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs. Digital I/O. External Interrupt 0.
RB1/INT1 RB1 INT1	34	37	9	I/O I	TTL ST	External Interrupt 1.
RB2/INT2 RB2 INT2	35	38	10	I/O I	TTL ST	Digital I/O. External Interrupt 2.
RB3/CCP2 RB3 CCP2	36	39	11	I/O I/O	TTL ST	Digital I/O. Capture2 input, Compare2 output, PWM2 output.
RB4	37	41	14	I/O	TTL	Digital I/O. Interrupt-on-change pin.
RB5/PGM RB5 PGM	38	42	15	I/O I/O	TTL ST	Digital I/O. Interrupt-on-change pin. Low Voltage ICSP programming enable pin.
RB6/PGC RB6 PGC	39	43	16	I/O I/O	TTL ST	Digital I/O. Interrupt-on-change pin. In-Circuit Debugger and ICSP programming clock pin.
RB7/PGD RB7 PGD	40	44	17	I/O I/O	TTL ST	Digital I/O. Interrupt-on-change pin. In-Circuit Debugger and ICSP programming data pin.

Legend: TTL = TTL compatible input
 ST = Schmitt Trigger input with CMOS levels
 O = Output
 OD = Open Drain (no P diode to VDD)

CMOS = CMOS compatible input or output
 I = Input
 P = Power

PIC18FXX2

TABLE 1-3: PIC18F4X2 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	DIP	PLCC	TQFP			
RC0/T1OSO/T1CKI	15	16	32	I/O	ST	PORTC is a bi-directional I/O port. Digital I/O. Timer1 oscillator output. Timer1/Timer3 external clock input.
RC0				O	—	
T1OSO				I	ST	
T1CKI						
RC1/T1OSI/CCP2	16	18	35	I/O	ST	Digital I/O. Timer1 oscillator input. Capture2 input, Compare2 output, PWM2 output.
RC1				I	CMOS	
T1OSI CCP2				I/O	ST	
RC2/CCP1	17	19	36	I/O	ST	Digital I/O. Capture1 input/Compare1 output/PWM1 output.
RC2				I/O	ST	
CCP1						
RC3/SCK/SCL	18	20	37	I/O	ST	Digital I/O. Synchronous serial clock input/output for SPI mode. Synchronous serial clock input/output for I ² C mode.
RC3				I/O	ST	
SCK						
SCL				I/O	ST	
RC4/SDI/SDA	23	25	42	I/O	ST	Digital I/O. SPI Data In. I ² C Data I/O.
RC4				I	ST	
SDI SDA				I/O	ST	
RC5/SDO	24	26	43	I/O	ST	Digital I/O. SPI Data Out.
RC5				O	—	
RC6/TX/CK	25	27	44	I/O	ST	Digital I/O. USART Asynchronous Transmit. USART Synchronous Clock (see related RX/DT).
RC6				O	—	
TX CK				I/O	ST	
RC7/RX/DT	26	29	1	I/O	ST	Digital I/O. USART Asynchronous Receive. USART Synchronous Data (see related TX/CK).
RC7				I	ST	
RX DT				I/O	ST	

Legend: TTL = TTL compatible input
 ST = Schmitt Trigger input with CMOS levels
 O = Output
 OD = Open Drain (no P diode to VDD)

CMOS = CMOS compatible input or output
 I = Input
 P = Power



MICROCHIP 24AA256/24LC256/24FC256

256K I²C™ CMOS Serial EEPROM

Device Selection Table

Part Number	V _{CC} Range	Max. Clock Frequency	Temp. Ranges
24AA256	1.8-5.5V	400 kHz ⁽¹⁾	I
24LC256	2.5-5.5V	400 kHz	I, E
24FC256	1.8-5.5V	1 MHz ⁽²⁾	I

Note 1: 100 kHz for V_{CC} < 2.5V.

Note 2: 400 kHz for V_{CC} < 2.5V.

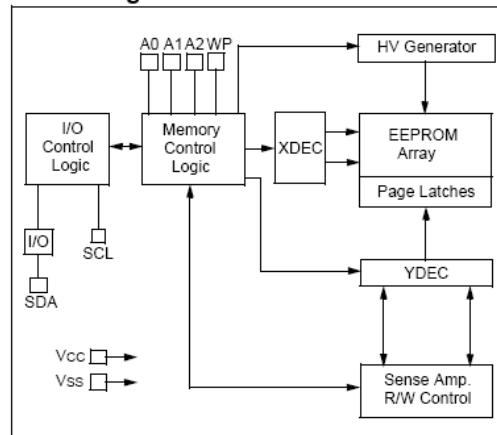
Features:

- Low-power CMOS technology:
 - Maximum write current 3 mA at 5.5V
 - Maximum read current 400 μA at 5.5V
 - Standby current 100 nA, typical at 5.5V
- 2-wire serial interface bus, I²C™ compatible
- Cascadable for up to eight devices
- Self-timed erase/write cycle
- 64-byte Page Write mode available
- 5 ms max. write cycle time
- Hardware write-protect for entire array
- Output slope control to eliminate ground bounce
- Schmitt Trigger inputs for noise suppression
- 1,000,000 erase/write cycles
- Electrostatic discharge protection > 4000V
- Data retention > 200 years
- 8-pin PDIP, SOIC, TSSOP, MSOP and DFN packages, 14-lead TSSOP package
- Pb-free finishes available
- Temperature ranges:
 - Industrial (I): -40°C to +85°C
 - Automotive (E): -40°C to +125°C

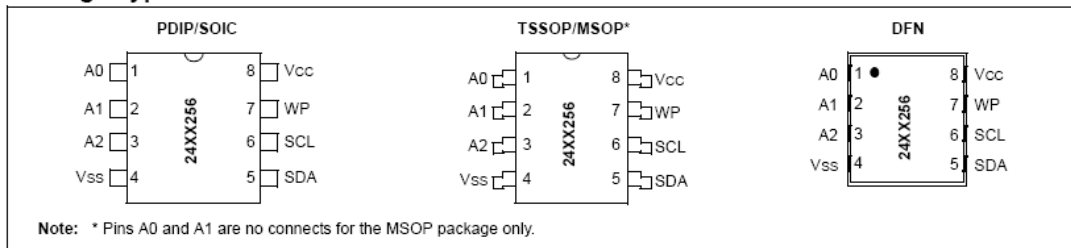
Description:

The Microchip Technology Inc. 24AA256/24LC256/24FC256 (24XX256*) is a 32K x 8 (256 Kbit) Serial Electrically Erasable PROM, capable of operation across a broad voltage range (1.8V to 5.5V). It has been developed for advanced, low-power applications such as personal communications or data acquisition. This device also has a page write capability of up to 64 bytes of data. This device is capable of both random and sequential reads up to the 256K boundary. Functional address lines allow up to eight devices on the same bus, for up to 2 Mbit address space. This device is available in the standard 8-pin plastic DIP, SOIC, TSSOP, MSOP and DFN packages.

Block Diagram



Package Types



*24XX256 is used in this document as a generic part number for the 24AA256/24LC256/24FC256 devices.

24AA256/24LC256/24FC256

1.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings^(†)

V _{CC}	6.5V
All inputs and outputs w.r.t. V _{SS}	-0.6V to V _{CC} +1.0V
Storage temperature.....	-65°C to +150°C
Ambient temperature with power applied.....	-40°C to +125°C
ESD protection on all pins.....	≥ 4 kV

† NOTICE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability.

TABLE 1-1: DC CHARACTERISTICS

DC CHARACTERISTICS			Electrical Characteristics:			
			Industrial (I): V _{CC} = +1.8V to 5.5V		T _A = -40°C to +85°C	
			Automotive (E): V _{CC} = +2.5V to 5.5V		T _A = -40°C to +125°C	
Param. No.	Sym.	Characteristic	Min.	Max.	Units	Conditions
D1	—	A0, A1, A2, SCL, SDA and WP pins:	—	—	—	—
D2	V _{IH}	High-level input voltage	0.7 V _{CC}	—	V	—
D3	V _{IL}	Low-level input voltage	—	0.3 V _{CC} 0.2 V _{CC}	V V	V _{CC} ≥ 2.5V V _{CC} < 2.5V
D4	V _{HYS}	Hysteresis of Schmitt Trigger inputs (SDA, SCL pins)	0.05 V _{CC}	—	V	V _{CC} ≥ 2.5V (Note)
D5	V _{OL}	Low-level output voltage	—	0.40	V	I _{OL} = 3.0 ma @ V _{CC} = 4.5V I _{OL} = 2.1 ma @ V _{CC} = 2.5V
D6	I _{LI}	Input leakage current	—	±1	μA	V _{IN} = V _{SS} or V _{CC} , WP = V _{SS} V _{IN} = V _{SS} or V _{CC} , WP = V _{CC}
D7	I _{LO}	Output leakage current	—	±1	μA	V _{OUT} = V _{SS} or V _{CC}
D8	C _{IN} , C _{OUT}	Pin capacitance (all inputs/outputs)	—	10	pF	V _{CC} = 5.0V (Note) T _A = 25°C, F _{CLK} = 1 MHz
D9	I _{CC} Read	Operating current	—	400	μA	V _{CC} = 5.5V, SCL = 400 kHz
	I _{CC} Write		—	3	mA	V _{CC} = 5.5V
D10	I _{CCS}	Standby current	—	1	μA	T _A = -40°C to +85°C SCL = SDA = V _{CC} = 5.5V A0, A1, A2, WP = V _{SS}
			—	5	μA	T _A = -40°C to +125°C SCL = SDA = V _{CC} = 5.5V A0, A1, A2, WP = V _{SS}

Note: This parameter is periodically sampled and not 100% tested.



DS1307 64 X 8 Serial Real Time Clock

www.dalsemi.com

FEATURES

- Real time clock counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap year compensation valid up to 2100
- 56 byte nonvolatile RAM for data storage
- 2-wire serial interface
- Programmable squarewave output signal
- Automatic power-fail detect and switch circuitry
- Consumes less than 500 nA in battery backup mode with oscillator running
- Optional industrial temperature range -40°C to +85°C
- Available in 8-pin DIP or SOIC
- Recognized by Underwriters Laboratory

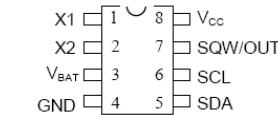
ORDERING INFORMATION

DS1307	8-Pin DIP
DS1307Z	8-Pin SOIC (150 mil)
DS1307N	8-Pin DIP (Industrial)
DS1307ZN	8-Pin SOIC (Industrial)

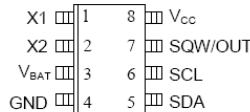
DESCRIPTION

The DS1307 Serial Real Time Clock is a low power, full BCD clock/calendar plus 56 bytes of nonvolatile SRAM. Address and data are transferred serially via a 2-wire bi-directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with less than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit which detects power failures and automatically switches to the battery supply.

PIN ASSIGNMENT



DS1307 8-Pin DIP (300 mil)



DS1307Z 8-Pin SOIC (150 mil)

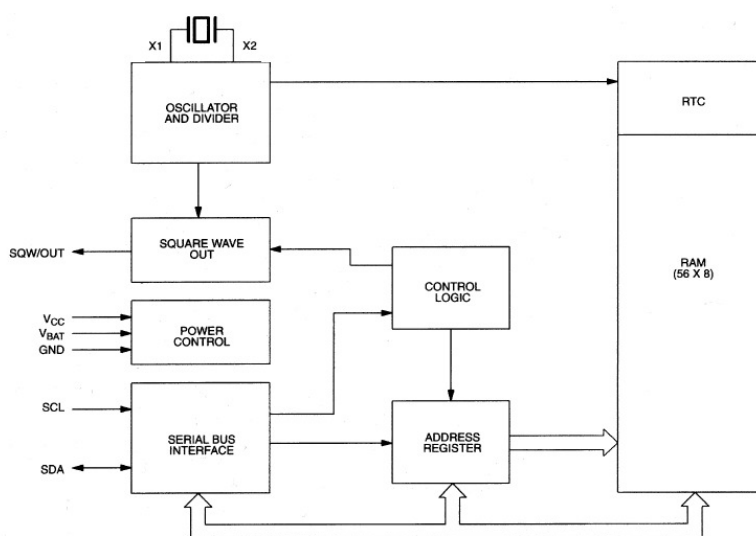
PIN DESCRIPTION

V _{CC}	- Primary Power Supply
X1, X2	- 32.768 kHz Crystal Connection
V _{BAT}	- +3V Battery Input
GND	- Ground
SDA	- Serial Data
SCL	- Serial Clock
SQW/OUT	- Square wave/Output Driver

OPERATION

The DS1307 operates as a slave device on the serial bus. Access is obtained by implementing a START condition and providing a device identification code followed by a register address. Subsequent registers can be accessed sequentially until a STOP condition is executed. When V_{CC} falls below $1.25 \times V_{BAT}$ the device terminates an access in progress and resets the device address counter. Inputs to the device will not be recognized at this time to prevent erroneous data from being written to the device from an out of tolerance system. When V_{CC} falls below V_{BAT} the device switches into a low current battery backup mode. Upon power up, the device switches from battery to V_{CC} when V_{CC} is greater than $V_{BAT} + 0.2V$ and recognizes inputs when V_{CC} is greater than $1.25 \times V_{BAT}$. The block diagram in Figure 1 shows the main elements of the Serial Real Time Clock.

DS1307 BLOCK DIAGRAM Figure 1



SIGNAL DESCRIPTIONS

V_{CC} , **GND** - DC power is provided to the device on these pins. V_{CC} is the +5 volt input. When 5 volts is applied within normal limits, the device is fully accessible and data can be written and read. When a 3-volt battery is connected to the device and V_{CC} is below $1.25 \times V_{BAT}$, reads and writes are inhibited. However, the Timekeeping function continues unaffected by the lower input voltage. As V_{CC} falls below V_{BAT} the RAM and timekeeper are switched over to the external power supply (nominal 3.0V DC) at V_{BAT} .

V_{BAT} - Battery input for any standard 3-volt lithium cell or other energy source. Battery voltage must be held between 2.0 and 3.5 volts for proper operation. The nominal write protect trip point voltage at which access to the real time clock and user RAM is denied is set by the internal circuitry as $1.25 \times V_{BAT}$ nominal. A lithium battery with 48 mAh or greater will back up the DS1307 for more than 10 years in the absence of power at 25 degrees C.

SCL (Serial Clock Input) - SCL is used to synchronize data movement on the serial interface.

SDA (Serial Data Input/Output) - SDA is the input/output pin for the 2-wire serial interface. The SDA pin is open drain which requires an external pullup resistor.

SQW/OUT (Square Wave/ Output Driver) - When enabled, the SQWE bit set to 1, the SQW/OUT pin outputs one of four square wave frequencies (1 Hz, 4 kHz, 8 kHz, 32 kHz). The SQW/OUT pin is open drain which requires an external pullup resistor. SQW/OUT will operate with either Vcc or Vbat applied.

X1, X2 - Connections for a standard 32.768 kHz quartz crystal. The internal oscillator circuitry is designed for operation with a crystal having a specified load capacitance (CL) of 12.5 pF.

For more information on crystal selection and crystal layout considerations, please consult Application Note 58, "Crystal Considerations with Dallas Real Time Clocks." The DS1307 can also be driven by an external 32.768 kHz oscillator. In this configuration, the X1 pin is connected to the external oscillator signal and the X2 pin is floated.

Please review Application Note 95, "Interfacing the DS1307 with a 8051-Compatible Microcontroller" for additional information.

RTC AND RAM ADDRESS MAP

The address map for the RTC and RAM registers of the DS1307 is shown in Figure 2. The real time clock registers are located in address locations 00h to 07h. The RAM registers are located in address locations 08h to 3Fh. During a multi-byte access, when the address pointer reaches 3Fh, the end of RAM space, it wraps around to location 00h, the beginning of the clock space.

DS1307 ADDRESS MAP Figure 2

00H	SECONDS
	MINUTES
	HOURS
	DAY
	DATE
	MONTH
	YEAR
07H	CONTROL
08H	RAM
3FH	56 x 8

Acronimos

ACK	A cknowledgement
ADC	A nalog-to- D igital Converter
ALU	A rithmetic L ogic U nit
ASCII	A merican S tandard C ode for I nformation I nterchange
BCD	B inary- C oded D ecima
CAN	C ontroller A rea N etwork
CCP	C apture- C ompare and P wm M ode
CI	C ircuito I ntegrado
CISC	C ompleted I nstruction S et C omputer
CPU	C entral P rocessing U nit
DCOM	D istributed C omponent O bject M odel
DNA	D eoxyribo N ucleic A cid
EEPROM	E lectrically E rasable P rogrammable R ead O nly M emory
FPGA	F ield P rogrammable G ate A rray
GFP	G reen F luorescent P rotein
GPR	G eneral P urpose R egisters
IDE	I ntegrated D evelopment E nviroment
ISR	I nterrupt S ervice R outine
I ² C	I nter- I ntegrated C ircuit
LCD	L iquid C ristal D isplay
LVD	L ow V oltage D etection
MSSP	M aster S ynchronous S erial P ort
μC	M icro C ontrolador
PCB	P rinted C ircuit B oard
PIC	P eripheral I nterface C ontroller
POR	P ower- O n R eset
PSP	P arallel S lave P ort
PWRT	P ower-up T imer
PWM	P ulse- W idth M odulation
RAM	R andom A ccess M emory
ROM	R ead- O nly M emory
RISC	R educed I nstruction S et C omputer
RTC	R eal T ime C lock
SQW	S quare W ave
SCL	S erial C lock
SPI	S erial P eripheral I nterface
SDA	S erial D ata
TCP/IP	T ransfer C ontrol P rotocol / I nternet P rotocol
TCY	T imes the processor C lock period
UART	U niversal A synchronous R eceiver- T ransmitter
USB	U niversal S erial B us
USART	U niversal S ynchronous/ A synchronous R eceiver T ransmitter
WDT	W atch D og T imer
WP	W ire P rotect