



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES
"ARAGÓN"

DISEÑO Y DESARROLLO DEL HARDWARE Y
SOFTWARE PARA UN EQUIPO MECÁNICO-ÓPTICO
PARA PRUEBAS DE FLEXIÓN INSITU EN MATERIALES

T E S I S

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO MECÁNICO ELECTRICISTA

[ÁREA: ELÉCTRICA - ELECTRÓNICA]

PRESENTA:

MARTIN ESTRADA ARCOS

ASESOR DE TESIS:

M. en I. ISRAEL NAVA BRAVO

México, 2011



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.

Dedicatoria.

Gracias a Dios que me permitió realizarme en lo que más me gusta.

A mis padres.

Especialmente a ti mamá, gracias por tu apoyo, pero sobre todo, gracias por tu amor. A ti papá, gracias por tu apoyo y tus consejos.

Antonio y Nancy.

Gracias por todo su apoyo y por su ejemplo.

Moni, Liz, Chechon, Yose y Luis.

Gracias por su motivación.

A ti Faby.

Porque también eres parte de este logro, gracias por tu apoyo.

*A toda mi familia y amigos.
Martin Estrada, Abril 2010*

Agradecimientos.

A la Universidad Nacional Autónoma de México.

A la Facultad de Estudios Superiores "Aragón"

Al M. en I. Israel Nava Bravo.

Gracias por su apoyo y por sus consejos, le agradezco mucho sus enseñanzas, pero sobre todo, su paciencia.

A mis sinodales y profesores de Aragón.

M. en I. Arturo Ocampo, M. en I. Alejandro Vega, Ing.

Juan Gastaldi, M. en C. Rafael Márquez, M. en C.

Ismael Díaz, Ing. Martín Meléndez, Ing. Pablo Luna,

Ing. Ignacio Mendoza, Ing. Margarito.

Por contribuir enormemente en mi formación académica y profesional

Dr. Jacinto Cortes y M. en I. Néstor García.

Por todo su apoyo y sus consejos, gracias por su motivación y su contribución en el desarrollo de este trabajo.

A todos mis compañeros de Aragón.

*Moy, Carlos, Sergio, Ángel, Mauricio, Filip, Ricardo,
Copado, Tolucos, Arubi, Aldo, Rukis, Chabelo, Pepepus, y
todos los compañeros del laboratorio de mecánica.*

M. en S. Lauro Santiago.

Mi corta estancia a su lado fue de total aprendizaje.

A todos los compañeros del Instituto de Ingeniería.

*Humberto, William, Anuar, David, M. en S. Mario, Rayu,
Cuauhtémoc.*

COMECYT

Por el apoyo otorgado a través del programa:

Beca Tesis de Licenciatura 2010.

SEP

Por el apoyo brindado a través del programa:

Becas de Titulación 2010

Un agradecimiento especial a la Dirección General de Asuntos del Personal Académico (DGAPA) de la UNAM, a través del Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica (PAPIIT).

Por el apoyo al proyecto: **“Estudio y desarrollo de aplicaciones empleando materiales con memoria de forma”** cuya clave es IN111310.

CONTENIDO

Introducción		1
Capítulo 1.	GENERALIDADES	3
I.1	Ciencia e ingeniería de los materiales	3
I.1.1	Clasificación de los materiales	4
I.1.1.1	Metales	4
I.1.1.2	Cerámicas	4
I.1.1.3	Polímeros	4
I.1.1.4	Materiales compuestos	4
I.1.1.5	Semiconductores	4
I.1.1.6	Nuevos materiales	5
I.1.2	La estructura de los sólidos cristalinos	5
I.1.2.1	Estructura cristalina	5
I.1.2.2	Celda unitaria	6
I.1.3	Estructura cristalina de los metales	6
I.1.3.1	Estructura cristalina cúbica centrada en las caras	6
I.1.3.2	Estructura cristalina cúbica centrada en el cuerpo	6
I.1.3.3	Estructura hexagonal compacta	7
I.1.4	Materiales cristalinos	8
I.1.4.1	Monocristales	8
I.1.4.2	Materiales Policristalinos	9
I.1.5	Observación microscópica	9
I.1.5.1	Microscopia	10
I.2	Caracterización y pruebas mecánicas a materiales	11
I.2.1	Equipo para el estudio de materiales	12
I.2.1.1	Mesas de posicionamiento	12
I.2.1.2	Software de procesamiento de imágenes	13
I.3	Equipo Mecánico-Óptico para Pruebas de Flexión In Situ en Materiales	13
I.4	Planteamiento del problema y propuesta de solución	16
I.4.1	Planteamiento del Problema	16
I.4.2	Propuesta de Solución	16
Capítulo 2.	FUNDAMENTOS TEÓRICOS	19
II.1	Fuentes de alimentación	19
II.1.1	Filtro de capacitor	20
II.1.2	Reguladores lineales integrados	21
II.1.3	Convertidores CD-CD	22
II.1.3.1	Principio de funcionamiento	22

II.1.4	Reguladores conmutados	23
	II.1.4.1 Regulador reductor (Buck o Step-Down)	24
	II.1.4.2 Regulador elevador (Boost o Step-Up)	25
II.2	Motores de pasos	26
	II.2.1 Tipos de motores paso a paso	26
	II.2.2 Motores bipolares	27
	II.2.2.1 Secuencias de funcionamiento	28
	II.2.3 Motores unipolares	29
	II.2.3.1 Secuencias de funcionamiento	30
II.3	Microcontroladores	31
	II.3.1 ¿Qué Es Un Microcontrolador?	31
	II.3.2 Características de los microcontroladores PIC	32
	II.3.3 El Microcontrolador PIC 18F 4550	34
II.4	Comunicación serie	35
	II.4.1 Especificaciones del estándar RS-232	35
	II.4.1.1 Flujo de datos	35
	II.4.1.2 Tensiones	36
	II.4.1.3 Cables y conectores	37
	II.4.2 Especificaciones del estándar USB	38
	II.4.2.1 Flujo de datos	38
	II.4.2.2 Topología	39
	II.4.2.3 Protocolo de comunicación	39
	II.4.2.4 Cables	40
	II.4.2.5 Conectores	41
	II.4.3 Migración de RS232 a USB	41
II.5	Herramientas de apoyo	42
	II.5.1 Entorno de programación en el Compilador C	42
	II.5.1.1 Entorno de trabajo	42
	II.5.1.2 Estructura de un programa	43
	II.5.2 Programación grafica en LabVIEW	43
	II.5.2.1 Ambiente de trabajo	44
	II.5.2.2 Componentes del programa	44
	II.5.3 Diseño Electrónico en PROTEUS VSM	41

Capítulo 3. DESARROLLO DEL HARDWARE 47

III.1	Descripción del hardware	47
	III.1.1 Selección de los motores	48
III.2	Fuente de alimentación	44
III.3	Control	49
	III.3.1 Conexiones al microcontrolador	50
	III.3.1.1 Alimentación	50
	III.3.1.2 Oscilador	50
	III.3.1.3 Reset	51
	III.3.1.4 Modulación por Ancho de Pulso (PWM)	51

	III.3.1.5 Iluminación	52
	III.3.1.6 Botones	52
	III.3.1.7 Indicadores	53
III.4	Potencia	53
	III.4.1 Driver	54
	III.4.2 Aislamiento	55
	III.4.3 Conexión de los motores	56
	III.4.4 Multiplexaje de los motores	56

Capítulo 4. DESARROLLO DEL CÓDIGO DEL MICROCONTROLADOR **59**

IV.1	Estructura del firmware	59
IV.2	Inicialización de puertos y periféricos	60
	IV.2.1 PWM	60
	IV.2.2 USB	61
	IV.2.3 Interrupciones externas	61
IV.3	Secuencias de giro	61
IV.4	Conexión USB	63
IV.5	Control manual	63
IV.6	Deshabilitación del control manual	65
IV.7	Control automatizado por computadora	65

Capítulo 5. DESARROLLO DEL SOFTWARE **67**

V.1	Estructura del programa de computadora	67
V.2	Interfaz de Usuario para el Control de los Motores	67
	V.2.1 Decodificación de datos, SubVI 1	69
	V.2.2 Envío de datos, SubVI 2	70
	V.2.3 Comparación de datos, SubVI 3	70
	V.2.4 Integración de los SubVI's	71
V.3	Programa para Manipulación de Imágenes	72
V.4	Programa para Comparación de Imágenes	75

Capítulo 6. PRUEBAS Y CONCLUSIONES **77**

VI.1	Integración del sistema	77
VI.2	Pruebas al sistema	80
	VI.2.1 Resultados del hardware	81
	VI.2.2 Resultados del software	81
VI.3	Conclusiones	81

Bibliografía	83
Apéndice A. Código del microcontrolador	85
Apéndice B. Hojas de especificaciones	91

Introducción

Probablemente la importancia de los materiales en nuestra cultura es mayor de lo que habitualmente se cree. Prácticamente cada segmento de nuestra vida cotidiana está influido en mayor o menor grado por los materiales, como por ejemplo: transporte, vivienda, vestimenta, comunicación, recreación y alimentación. En las sociedades más antiguas podemos encontrar gran variedad de usos de los materiales de acuerdo al conocimiento de cada época. A pesar de ello, no es sino hasta el siglo XVII cuando se dan avances significativos en el estudio del comportamiento de materiales, mismos que han ido incrementando de manera agigantada gracias a la evolución de tecnologías que permiten la existencia de nuevos materiales con nuevas propiedades y características.

De manera general los materiales se clasifican en metales, polímeros, cerámicos y compuestos, siendo los primeros de esta lista de gran importancia tanto desde el punto de vista social como científico. Para poder utilizar y aprovechar las propiedades de estos materiales es necesario poder predecir su comportamiento, es por ello que existen diversas áreas y técnicas que estudian los materiales, desde su obtención, su fabricación, conformado y caracterización, hasta detalles mismos en su estructura atómica. Una de las áreas de mayor interés en este sentido es la metalurgia mecánica, la cual estudia el comportamiento mecánico de los materiales metálicos bajo fuerzas y su relación con su micro estructura. Para encontrar dichas relaciones es necesario estudiar a nivel microscópico y atómico el comportamiento del material. Para dicho fin, existe diversidad de equipos dentro de los que se resaltan los microscopios ópticos y electrónicos. Con tales equipos y debido a sus diferentes características es posible realizar estudios a través de la observación, tomando micrografías de cierta región de la muestra del material, sin embargo, en su gran mayoría no cuentan con la posibilidad de dar seguimiento in situ en pruebas a diferentes arreglos mecánicos. Por tal razón, es de vital importancia contar con equipos que sean capaces de seguir paso a paso las pruebas en diferentes arreglos de carga mecánica como lo son: tensión, flexión, torsión y diferentes combinaciones de ellas, así como tomar micrografías in situ con programas de computadora que sirvan para manipular imágenes con el fin de poder comparar el comportamiento antes y después de aplicar las fuerzas.

Justificación

Actualmente en el mercado existen diversos equipos para arreglos de carga básicos que sirven para dicho fin, sin embargo tienen la particularidad de ser de importación, con especificaciones de cada tipo de fabricante, así como tener precios muy elevados. Es por ello que dentro del laboratorio de Mecánica Aplicada de la FES Aragón a través de equipos de trabajo multidisciplinarios se pretende resolver problemas concretos de ingeniería mecánica y desarrollar tecnología propia, ya sea con conocimiento de dominio público o con conocimientos desarrollados con el grupo de trabajo.

Una línea de investigación de este laboratorio es precisamente el estudio del comportamiento mecánico de materiales. Como parte de esta investigación se diseñó y fabricó un Equipo Mecánico-Óptico para Pruebas de Flexión In Situ en Materiales. Este equipo permite desarrollar pruebas de flexión a materiales, por medio de una mesa de posicionamiento, la cual posiciona un microscopio de forma manual sobre el material para tomar micrografías que son analizadas en la computadora. Sin embargo, recientemente fue propuesto un rediseño del equipo para que su funcionamiento fuese de forma automática, lo cual dio origen a la presente tesis.



Objetivo

En el presente trabajo se describe el diseño y la implementación de un sistema de control para el sistema de posicionamiento del equipo mecánico-óptico, dicho sistema tiene como objetivo el desarrollo del hardware y software necesario para controlar a la mesa, así como el desarrollo de un programa de computadora que permita manipular la posición del microscopio. También se desarrolla un programa de computadora que sirva para manipular las imágenes obtenidas por el microscopio. Con este trabajo se pretende complementar el estudio de los materiales con equipo y herramientas que permitan facilitar, agilizar y precisar las pruebas a dichos materiales.

Para lograr el objetivo, el presente trabajo de tesis se ha dividido en seis capítulos, en los cuales se encuentran antecedentes y generalidades necesarios para el desarrollo de este trabajo. Se plantearán las necesidades y características con que deba contar el sistema de control. Se detallará el software y hardware desarrollado, así como las pruebas que se realizaron al mismo.

Las características principales de cada capítulo son:

Capítulo 1. Se describirá de manera general una de las técnicas más utilizadas para el estudio de los materiales, la importancia del estudio y la descripción del equipo Mecánico-Óptico. Se planteará el problema y la propuesta de solución que originó el desarrollo del presente trabajo de tesis.

Capítulo 2. Se presentan los fundamentos teóricos básicos, se darán a conocer los principios de funcionamiento de la fuente y los motores seleccionados. Se especificarán las características del dispositivo que controlará a los motores y los ambientes de programación utilizados.

Para una mejor comprensión, el diseño y desarrollo del sistema se ha dividido en tres capítulos:

Capítulo 3. Este capítulo se dedica solamente al desarrollo de la electrónica que se utilizó para el control de los motores. Se dan a conocer los dispositivos electrónicos y sus respectivas configuraciones.

Capítulo 4. Se detalla el funcionamiento y la lógica del código del microcontrolador. Se establecen los modos de funcionamiento y sus respectivas funciones.

Capítulo 5. En este capítulo se muestra un programa de computadora que a su vez contiene tres programas, el primero sirve como interfaz gráfica de usuario para el posicionamiento coordinado de cada motor, el segundo es para la manipulación de las imágenes obtenidas del microscopio, y el tercero, es un programa que sirve para comparar imágenes, con este programa se puede obtener el campo de desplazamientos discretos, propuesta que surge del desarrollo de una nueva técnica para determinarlos, dicho programa es complemento del segundo.

Es importante mencionar y entender que, dado que la teoría para la aplicación del tercer programa de computadora es extensa y difiere en los intereses planteados en este trabajo de tesis; no se detallará dicha teoría y solo se describirá brevemente el uso y la aplicación de éste tercer programa en éste capítulo. Se invita al lector, en caso de requerir mayor información, referirse a la bibliografía correspondiente.

Capítulo 6. En este capítulo se muestra la integración del sistema en general, esto comprende el desarrollo de los circuitos impresos y su montaje dentro de un chasis. Se conectará el sistema desarrollado para realizar las correspondientes pruebas. Se muestran las conclusiones pertinentes.

Capítulo **1**

Generalidades

Objetivo particular. El presente capítulo presenta los conceptos básicos relacionados con la estructura atómica de los materiales, en particular la de los metales. Se definen los materiales monocristalinos y policristalinos y la importancia de su observación microscópica. En la parte final se describe el objetivo principal del presente trabajo de tesis.

I.1 CIENCIA E INGENIERÍA DE LOS MATERIALES

La disciplina **ciencia de los materiales** implica investigar la relación entre la estructura y las propiedades de los materiales. Por el contrario, la ingeniería de los materiales se fundamenta en las relaciones propiedades-estructura y diseña o proyecta la estructura de un material para conseguir un conjunto predeterminado de propiedades.

En general, el término estructura se refiere a la descripción del arreglo atómico de un material. La estructura de un material puede ser examinada en tres diferentes niveles:

1. **Estructura atómica:** Está formada por una red geométrica y una base. La base es el mínimo patrón repetición en el que se acomodan los átomos del material.
2. **Microestructura:** Estructura a escala microscópica, va desde algunos Angstroms a milímetros.
3. **Macroestructura:** Los elementos estructurales susceptibles de apreciarse a simple vista, hasta 1 metro

Debido a que un material en servicio, está expuesto a ciertos estímulos externos que provocan algún tipo de respuesta, las propiedades de un material se expresan en términos del tipo y magnitud de la respuesta a un estímulo específico impuesto. Las definiciones de las propiedades suelen ser independientes de la forma y del tamaño del material. Todas las propiedades importantes de los materiales sólidos se agrupan en seis categorías: mecánicas, eléctricas, térmicas, magnéticas, ópticas y químicas. Para cada categoría existe un tipo característico de estímulos capaz de provocar respuestas diferentes.



I.1.1. Clasificación de los materiales

Los materiales sólidos, en general, se clasifican en tres grupos: metales, cerámicas y polímeros. Este esquema se basa en la composición química y en la estructura atómica. Por lo general, la mayoría de los materiales encaja en un grupo u otro, aunque haya materiales intermedios. Además existen otros dos grupos de importantes materiales técnicos: materiales compuestos (composite) y semiconductores. Los materiales compuestos constan de dos o más materiales diferentes, mientras que los semiconductores se utilizan por sus extraordinarias características eléctricas. A continuación se describen brevemente los tipos de materiales y sus características más representativas.

I.1.1.1 Metales

Pueden ser puros o aleaciones. Tienen gran número de electrones libres que no pertenece a ningún átomo en concreto. La mayoría de las propiedades de los metales se atribuyen a los electrones libres. Los metales conducen perfectamente el calor y la electricidad, son opacos a la luz visible aunque la superficie metálica pulida tiene apariencia lustrosa. Además, los metales son resistentes, aunque deformables, lo que contribuye su principal ventaja en aplicaciones estructurales.

I.1.1.2 Cerámicas

Los compuestos químicos constituidos por metales y no metales (óxidos, nitruros y carburos), pertenecen al grupo de las cerámicas, que incluyen minerales de arcilla, cemento y vidrio. Por lo general se trata de materiales que son aislantes eléctricos y térmicos, y que a elevada temperatura y ambientes agresivos son más resistentes que los metales y los polímeros. Desde el punto de vista mecánico, las cerámicas son duras y muy frágiles.

I.1.1.3 Polímeros

Los polímeros comprenden materiales que van desde los familiares plásticos al caucho. Se tratan de compuestos orgánicos, basados en el carbono, hidrógeno y otros elementos no metálicos. Su "estructura molecular" consiste en grandes cadenas de "monómeros". Los polímeros poseen densidad, rigidez y resistencia bajas pero extraordinaria flexibilidad.

I.1.1.4 Materiales compuestos

Los materiales compuestos se forman por más de un tipo de material unidos química o mecánicamente. La fibra de vidrio, que es vidrio en forma filamentosos embebido dentro de un material polímero, es un ejemplo familiar. Los materiales compuestos están diseñados para alcanzar la mejor combinación de las características de cada componente. La fibra de vidrio es mecánicamente resistente debido al vidrio y flexible debido al polímero. La mayoría de los materiales desarrollados últimamente son materiales compuestos.

I.1.1.5 Semiconductores

Los semiconductores tienen propiedades eléctricas intermedias entre los conductores y los aislantes eléctricos. Las características eléctricas de los semiconductores, son extremadamente sensibles a la presencia de diminutas concentraciones de átomos de impurezas. Estas concentraciones de deben controlar en regiones especiales muy pequeñas. Los semiconductores facilitan la fabricación de los circuitos integrados que han revolucionado, en las últimas décadas, la industria electrónica.

I.1.1.6 Nuevos Materiales

Muchos de los avances de la tecnología actual, están fuertemente relacionados con el surgimiento de nuevos materiales los cuales, presentan un incremento considerable en sus propiedades o bien comportamientos diferentes a los convencionales. Un ejemplo de ellos son los denominados **Materiales con Memoria de Forma (MMF)**, los cuales tienen la capacidad de recuperar deformaciones aparentemente plásticas por simple calentamiento, lo que se denomina efecto memoria de forma simple.

I.1.2. La estructura de los sólidos cristalinos

A continuación se discute el siguiente nivel de la estructura de los materiales: la disposición adoptada por los átomos en el estado sólido. En este contexto se introducen los conceptos de cristalinidad en términos de la celda unitaria. Se detallan las tres estructuras cristalinas más frecuentes en los metales en función de planos y direcciones cristalográficas.

I.1.2.1 Estructura cristalina

Los materiales sólidos se pueden clasificar según la regularidad con que se sitúan, unos respecto de otros, los átomos o iones. En un material **cristalino**, los átomos se sitúan en una disposición repetitiva o periódica a lo largo de muchas distancias atómicas; es decir, existe un orden de largo alcance tal que, al solidificar el material, los átomos se sitúan según un patrón tridimensional repetitivo, en el cual cada átomo está enlazado con su vecino más próximo. Los metales, muchas cerámicas y ciertos polímeros adquieren estructuras cristalinas en condiciones normales de solidificación.

Algunas propiedades de los sólidos cristalinos dependen de la **estructura cristalina** del material; es decir, de la ordenación espacial de los átomos, iones y moléculas. Hay un número muy grande de estructuras cristalinas diferentes que tienen, todas ellas, orden atómico de largo alcance. Estas estructuras varían desde las relativamente simples de los metales a las excesivamente complejas de los materiales cerámicos y polímeros. Al describir la estructura cristalina se consideran a los átomos (o iones) como esferas sólidas con diámetros muy bien definidos. Es el denominado modelo atómico de las esferas rígidas, en el cual, las esferas representan átomos macizos en contacto. La figura 1.1c muestra un ejemplo de la disposición atómica de algunos de los metales elementales más corrientes mediante el modelo de las esferas rígidas. En este caso, todos los átomos son idénticos. En el contexto de las estructuras cristalinas a menudo se usa la palabra Red; en este sentido red significa disposición tridimensional de puntos coincidentes con las posiciones de los átomos (o centros de las esferas).

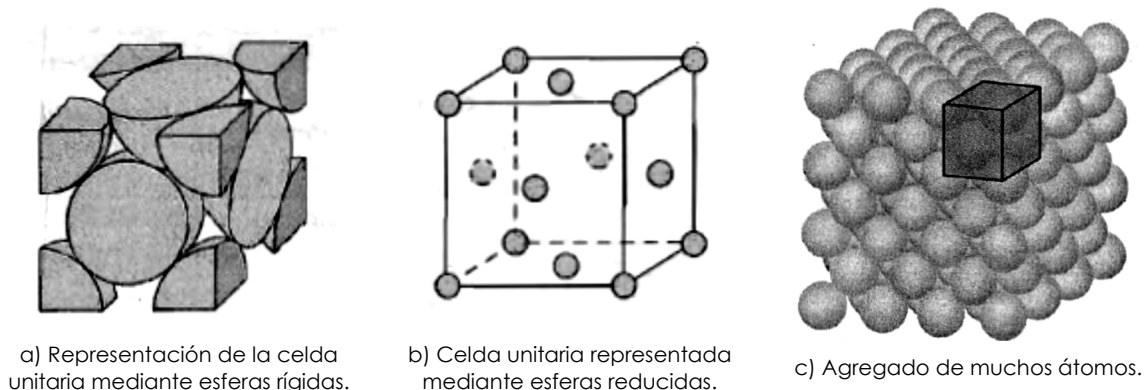


Figura 1.1 Estructura cristalina cúbica centrada en las caras.



I.1.2.2 Celda unitaria

El orden atómico de los sólidos cristalinos indica que grupos de pocos átomos forman un patrón que se repite en el espacio. Al describir la estructura cristalina conviene dividirla en las pequeñas entidades que se repiten, llamadas celdas unitarias. La **celda unitaria** de la mayoría de las estructuras cristalinas son paralelepípedos o prismas con tres conjuntos de caras paralelas. En la figura 1.1c se ha dibujado una celda, que en este caso es un cubo. La celdilla unidad se elige para representar la simetría de la estructura cristalina, de modo que las posiciones de los átomos en el cristal se puedan representar desplazando a distancias discretas la celda unitaria a lo largo de los ejes. De este modo, la celda unitaria *es la unidad estructural fundamental* y define la estructura cristalina mediante su geometría y por la disposición de los átomos dentro de ella. Ordinariamente, la claridad aconseja que los vértices del paralelepípedo coincidan con los centros de las esferas rígidas que representan los átomos. Para definir algunas estructuras cristalinas es necesario establecer más de una celda unitaria; sin embargo, generalmente se usa la celda unitaria que tiene el mayor nivel de simetría geométrica.

I.1.3. Estructura cristalina de los metales

I.1.3.1 Estructura cristalina cúbica centrada en las caras

Muchos metales dúctiles tienen una celda unitaria de geometría cúbica, con los átomos localizados en los vértices del cubo y en los centros de todas las caras del cubo. Es la denominada estructura cristalina **Cúbica Centrada en las Caras** (FCC). Cristalizan en esta estructura el cobre, aluminio, plata y oro. La figura 1.1a muestra el modelo de esferas rígidas de la celda unitaria FCC, mientras que la figura 1.1b representa los centros de los átomos mediante pequeños círculos, lo que origina una mejor perspectiva de las posiciones de los éstos. El agregado de la figura 1.1c representa una sección de cristal formada por un número de celda unitaria FCC. Estas esferas o cationes están en contacto entre sí a lo largo de la diagonal.

En los cristales de estructura FCC, cada átomo del vértice es compartido con ocho celdas unitarias, mientras que los átomos centrados en las caras sólo son compartidos con dos. Esto está representado en la figura 1.1a, donde sólo se han dibujado las partes de las esferas integradas dentro del cubo. La celda comprende el volumen del cubo generado desde los centros de los átomos de los vértices, como indica la citada figura.

Las posiciones de los vértices y de los centros de las caras son equivalentes, puesto que el desplazamiento de la celda unitaria desde el átomo del vértice original al átomo del centro de una cara no altera la estructura de la celda.

I.1.3.2 Estructura cristalina cúbica centrada en el cuerpo

Otra estructura cristalina común en los metales también es una celda unitaria cúbica que tiene átomos localizados en los ocho vértices y uno en el centro. Esta estructura cristalina se denomina **Cúbica Centrada en el Cuerpo** (BCC). En la figura 1.2c se muestra un conjunto de esferas que representan esta estructura cristalina, mientras que las figuras 1.2a y 1.2b son representaciones de celdas unitarias mediante modelos de esfera rígida y de esfera reducida, respectivamente. Los átomos del centro y de los vértices se tocan mutuamente a lo largo de las diagonales del cubo. El cromo, el hierro y el tungsteno presentan estructura BCC.

Cada celda unitaria BCC tiene asociado dos átomos: uno equivalente a un octavo de cada uno de los ocho átomos de los vértices, que son compartidos con otras celdas unitarias, y el del centro de la celda, que no es compartido. Además, las posiciones de los átomos del centro y del vértice son equivalentes.

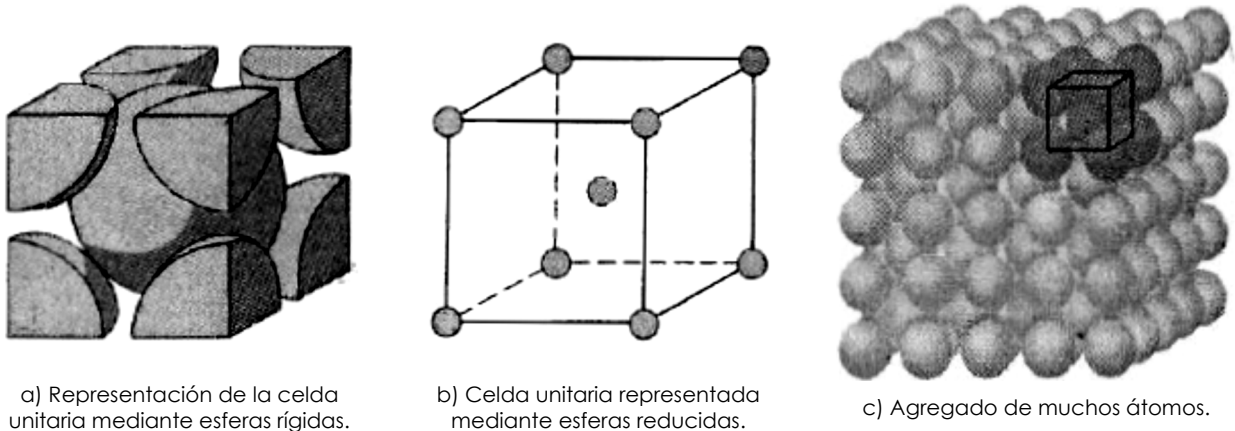


Figura 1.2 Estructura cristalina cúbica centrada en el cuerpo (BCC).

I.1.3.3 Estructura hexagonal compacta

No todos los metales tienen celda unitaria con simetría cúbica; la última estructura cristalina que se discute es la que tiene celda unitaria hexagonal. La figura 1.3a muestra la celda unitaria, en el modelo de esferas reducidas, de esta estructura denominada **Hexagonal Compacta** (HC). En la figura 1.3b se aprecian un conjunto de varias celdas unitarias HC. La base superior e inferior consiste en hexágonos regulares con sus átomos en los vértices y uno en el centro. Otro plano que provee de tres átomos adicionales a la celda unitaria está situado entre ambos planos. Cada celda unitaria equivale a seis átomos: cada átomo situado en los 12 vértices superiores e inferiores contribuye con la sexta parte, los 2 átomos del centro de los hexágonos contribuyen con la mitad y los 3 átomos del plano central contribuyen enteramente. En la figura 1.3a se aprecia que a y c representan las dimensiones corta y larga de la celda unitaria, respectivamente. La relación c/a debe ser 1.633, sin embargo, para algunos metales HC esta relación se desvía del valor ideal.

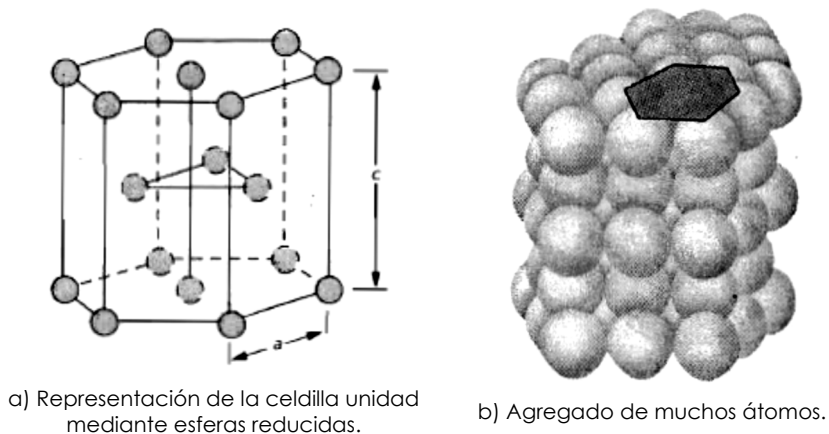


Figura 1.3 Estructura cristalina hexagonal compacta.

La tabla 1.1 indica una relación de distintos materiales metálicos en la que se indica su estructura cristalina y su radio atómico.

Tabla 1.1 Radios atómicos y estructuras cristalinas de 16 metales.

Metal	Estructura Cristalina ^a	Radio Atómico (nm) ^b	Metal	Estructura Cristalina ^a	Radio Atómico (nm) ^b
Aluminio	FCC	0.1431	Molibdeno	BCC	0.1363
Cadmio	HC	0.1490	Níquel	FCC	0.1246
Cromo	BCC	0.1249	Platino	FCC	0.1387
Cobalto	HC	0.1553	Plata	FCC	0.1445
Cobre	FCC	0.1278	Tántalo	BCC	0.1430
Oro	FCC	0.1442	Titanio (α)	HC	0.1445
Hierro	BCC	0.1241	Tungsteno	BCC	0.1371
Plomo	FCC	0.1750	Zinc	HC	0.1332

^a FCC = Cúbica Centrada en las Caras; HC = Hexagonal Compacta; BCC = Cúbica Centrada en el Cuerpo

^b Un nanómetro (nm) equivalente a 10^{-9} m.

I.1.4. Materiales cristalinos

I.1.4.1 Monocristales

Cuando la disposición atómica de un sólido cristalino es perfecta, sin interrupciones, a lo largo de toda la muestra, el resultado es un **monocristal**. Todas las celdas unitarias están entrelazadas o unidas del mismo modo y tienen la misma dirección. Los monocristales existen en la naturaleza, pero también se pueden generar artificialmente. El crecimiento de estos últimos es, por lo general, difícil, ya que el medio se debe controlar cuidadosamente.

Si los extremos de un monocristal crecen sin impedimentos externos, el cristal adquiere una forma geométrica regular con caras planas, como las que representan algunas piedras preciosas. La forma refleja la estructura cristalina. En la fotografía de la figura 1.4 se muestra un monocristal.



Figura 1.4 Aspecto externo de un monocristal de un mismo material.

En los últimos años los monocristales han adquirido extraordinaria importancia en la moderna tecnología, sobre todo en la electrónica, que emplea monocristales de silicio y de otros semiconductores. Sin embargo, es más frecuente el uso de los policristalinos debido a su bajo costo de fabricación.

I.1.4.2 Materiales policristalinos

La mayoría de los sólidos cristalinos son un conjunto de muchos cristales pequeños o granos. Este tipo de material se denomina policristalino. En la figura 1.5 se han esquematizado varias etapas de la solidificación de una muestra policristalina. Al principio, aparecen pequeños cristales o núcleos en distintas posiciones. Estas orientaciones cristalográficas son completamente al azar y están indicadas por cuadros. Los granos pequeños crecen por la sucesiva adición a la estructura de átomos del líquido subenfriado. Los extremos de granos adyacentes interaccionan entre sí al finalizar el proceso de solidificación. La figura 1.5 indica que la orientación cristalográfica varía de un grano a otro. También existe alguna irregularidad en la disposición atómica en la región donde se unen dos granos; esta área se denominada límite de grano.

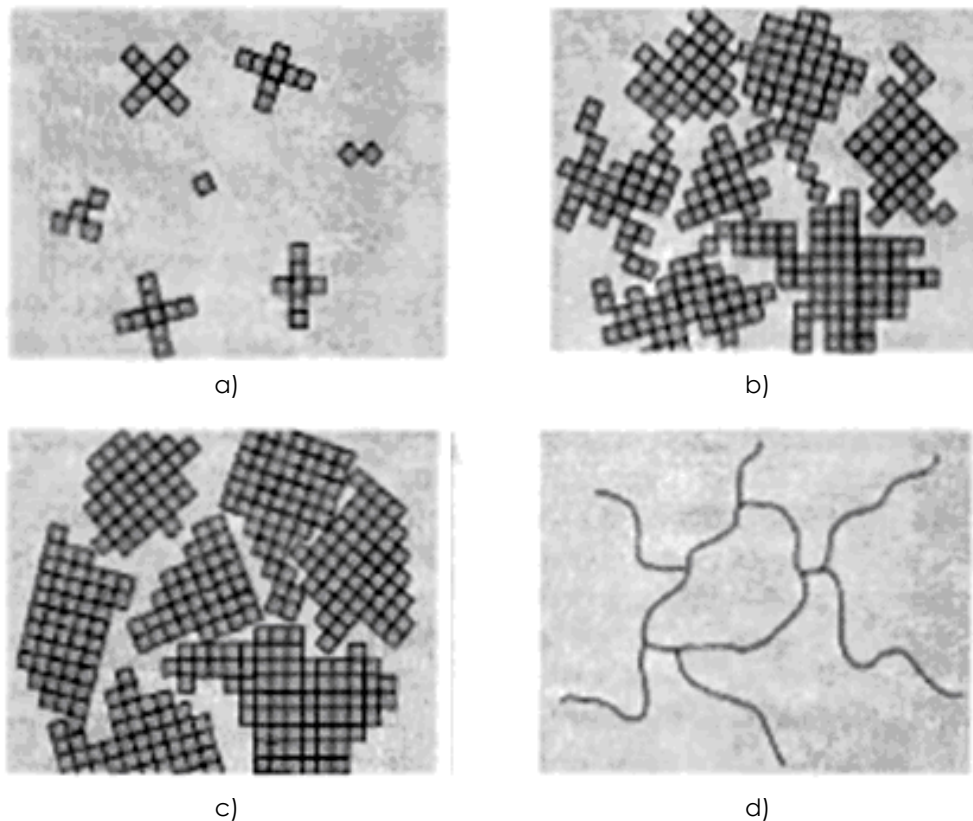


Figura 1.5 Esquema de varias etapas de la solidificación de un material policristalino; las cuadrículas representan celdas unitarias.

Es importante mencionar que el tamaño y forma del grano, así como la llamada morfologías de la fases, constituyen la llamada microestructura de un material. El tamaño del grano, y la misma forma del grano, puede variar entre milímetros y unas cuantas micras, dependiendo la rapidez con que solidificó el material y el trabajo en caliente efectuado sobre él entre otros parámetros.

I.1.5. Observación microscópica

A veces conviene examinar los elementos estructurales y defectos que modifican las propiedades de los materiales. La importancia de realizar estas investigaciones radica, en primer lugar, en asegurar la idónea comprensión de la asociación entre propiedades y estructura (defectos) y, en segundo lugar, en comprender las propiedades de los materiales

una vez establecidas estas relaciones. A continuación se discuten varias técnicas utilizadas en estas investigaciones. En la mayoría de los materiales los granos constituyentes son de dimensiones microscópicas, con diámetros del orden de micrómetros, y sus detalles se deben investigar mediante algún tipo de microscopio. Tamaño y forma de los granos son dos ejemplos de las características de la microestructura.

I.1.5.1 Microscopia

En microscopia normalmente se utilizan el microscopio óptico y el electrónico. Estos instrumentos contribuyen a la investigación de las características microestructurales de todo tipo de materiales. La mayoría de estas técnicas llevan acoplado un equipo fotográfico; la fotografía que recoge la imagen microscópica se denomina **micrografía**.

Para revelar los detalles importantes de la microestructura es necesario, generalmente, preparar cuidadosamente las superficies. La superficie debe lijarse y pulirse hasta que quede como un espejo. Esta condición se consigue usando papeles abrasivos y polvos cada vez más finos. Se revela la microestructura tratando la superficie con un reactivo químico apropiado en un procedimiento denominado **ataque**. La reactividad química de los granos de un material monofásico depende de la orientación cristalográfica. Por consiguiente, en una probeta policristalina, las características del ataque varían de un grano a otro. La figura 1.6b muestra cómo se refleja la luz que incide perpendicularmente sobre tres superficies de granos atacados y con diferentes orientaciones cristalográficas. La figura 1.6a describe la estructura superficial al ser observada con el microscopio, también se pueden apreciar tres figuras de geometría irregular las cuales son denominadas **grano**, el brillo de cada uno de ellos depende de sus propiedades de reflexión. La figura 1.6c muestra una micrografía de una probeta policristalina que representa estas propiedades.

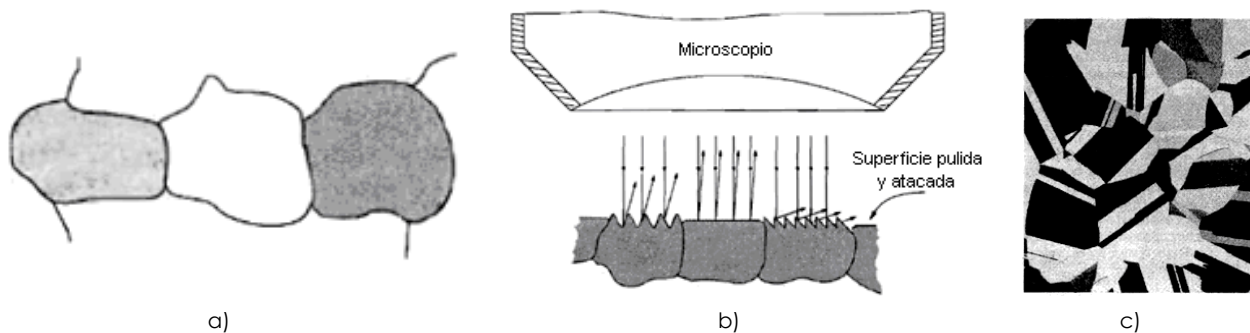


Figura 1.6 (a) Aspecto que ofrecen, al ser observados mediante microscopio óptico, los granos pulidos y atacados. (b) Sección transversal de estos granos mostrando la variación del ataque y de la textura superficial como consecuencia de las diferentes orientaciones cristalográficas. (c) micrografía de una probeta de latón policristalino x 60.

A lo largo de los límites de grano se forman pequeños surcos como consecuencia del ataque. Los átomos situados a lo largo de los límites del grano son más reactivos y se disuelven con mayor velocidad que los granos. Estos surcos se hacen visibles en la observación microscópica porque reflejan la luz según un ángulo distinto al de los granos; este efecto se muestra en la figura 1.7a. La figura 1.7b es una micrografía de una probeta policristalina cuyos surcos de los límites de grano se evidencian como líneas oscuras.

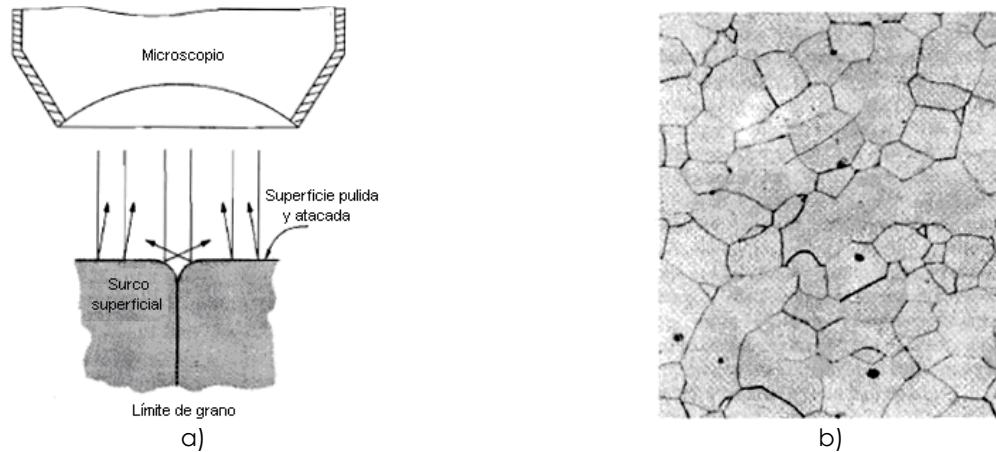


Figura 1.7 (a) Sección de un límite de grano y el surco superficial producido por ataque; se muestran las características de la luz reflejada en la vecindad del surco. (b) Micrografía de la superficie, pulida y atacada, de una probeta policristalina de una aleación de hierro-cromo, cuyos límites de grano aparecen oscuros x100.

1.2 CARACTERIZACIÓN Y PRUEBAS MECÁNICAS A MATERIALES

Un área de estudio de los materiales es la Metalurgia Mecánica, esta área determina el comportamiento de los materiales al ser sometidos a fuerzas externas. Una parte primordial y ya convencional de esta metodología es realizar pruebas en diferentes arreglos comunes, como son: tensión, flexión, torsión y/o combinaciones de éstas, para analizar su comportamiento. Uno de los arreglos mecánicos más utilizados para caracterizar materiales es la flexión, la cual consiste en tomar una muestra del material y someterla a una fuerza externa, como se muestra en la figura 1.8. La prueba de flexión requiere seguir el comportamiento de la muestra mediante micrografías durante la aplicación de la fuerza.

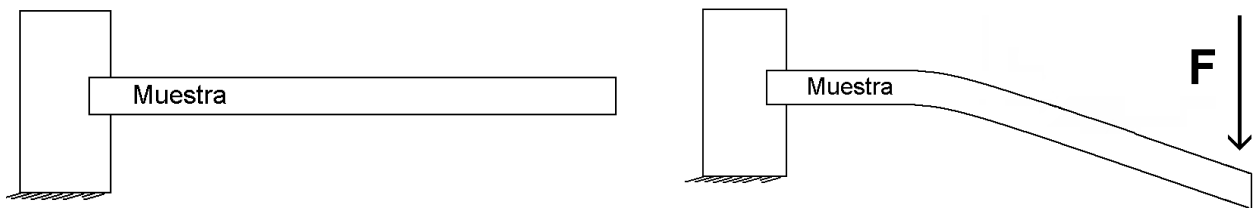


Figura 1.8 Arreglo mecánico en flexión.

El examen microscópico de los materiales constituye una herramienta extremadamente útil para el estudio y la caracterización de materiales. Del examen de la microestructura se determina la fractura mecánica, se prevén las propiedades mecánicas de las aleaciones, el correcto o anómalo tratamiento y el diseño de aleaciones con nuevas combinaciones de propiedades.

En la figura 1.9 se puede observar un ejemplo de una micrografía de un material policristalino, en este caso el material es una aleación de cobre, aluminio y berilio (Cu - Al %12 - Be %0.5), en dicha imagen se pueden observar los granos que se forman en esta aleación.

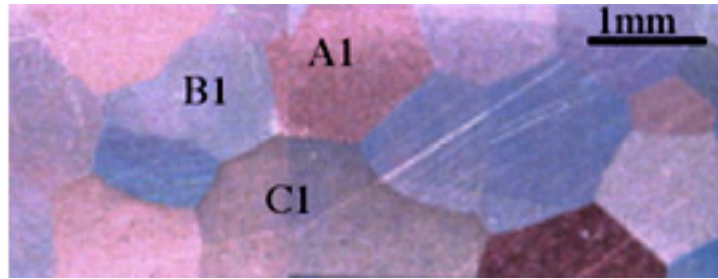


Figura 1.9 Micrografía de un material policristalino.

A partir de este momento nos centraremos en el estudio de los materiales policristalinos. El estudio es mediante el uso de *probetas*, estas probetas son pequeñas láminas de dimensiones y espesores regularmente del orden de cm, a las cuales se les llamará: **Muestras**.

1.2.1. Equipo para el estudio de materiales

Para realizar pruebas en los arreglos mecánicos mencionados en la sección anterior, es necesario contar con equipos que combinen principios de diferentes áreas de la ingeniería eléctrica, mecánica y en computación, para resolver inconvenientes técnicos al momento de realizar las pruebas como son: la aplicación de la fuerza, el sistema de sujeción de las muestras, la adquisición de datos e imágenes, etc. En la actualidad existen gran variedad de equipos en el mercado que resuelve éste tipo de inconvenientes para el estudio de materiales.

1.2.1.1 Mesas de posicionamiento

Las llamadas mesas de posicionamiento, como su nombre lo indica, nos permiten posicionar en un punto determinado algún componente que necesite ser llevado a un punto específico. Por ejemplo, en una prueba de flexión, se necesita tomar micrografías en diferentes puntos de la muestra, para ello se necesita contar con una mesa que posicione un microscopio en un punto determinado, dicho posicionamiento debe ser exacto y preciso ya que el área a fotografiar es del orden de milímetros. En la figura 1.10 se puede ver un ejemplo de una mesa de posicionamiento comercializada por el fabricante DeltaNu.



Figura 1.10 Mesas de posicionamiento comercial.

Estas mesas, dependiendo de la aplicación, manejan diferentes grados de libertad, que también suelen ser llamados 'ejes de movimiento'. Actualmente se construyen mesas de n ejes, y es importante resaltar que en la mayoría de estas mesas el control es meramente mecánico, es decir; los ejes se mueven de manera manual a partir de mecanismos basados en manivelas, volantes, engranes, etc., y su costo es de acuerdo a la cantidad de ejes de movimiento. Sin embargo, cada vez más equipos se diseñan para que su funcionamiento sea autónomo a través de la implementación de sistemas embebidos, dichos equipos se pueden dividir en dos partes: una parte es a nivel de hardware, es decir; equipo físico encargado de realizar el arreglo mecánico para tomar las micrografías correspondientes, y la otra parte a nivel de software, ésta se encarga de controlar automáticamente el manejo de la mesa para el posicionamiento.

1.2.1.2 Software de procesamiento de imágenes

También existen programas de computadora capaces de reconocer las imágenes de las micrografías y procesarlas automáticamente para la adquisición de datos, estos programas son de construcción robusta y debido a su complejidad y por ser software especializado no es de fácil adquisición. Esta herramienta es de gran utilidad ya que optimiza y acelera las pruebas, permitiendo al usuario dedicarle más tiempo al análisis matemático y/o mecánico de los datos obtenidos.

1.3 EQUIPO MECÁNICO-ÓPTICO PARA PRUEBAS DE FLEXIÓN IN SITU EN MATERIALES

En el laboratorio de Mecánica Aplicada del Centro Tecnológico de la Facultad de Estudios Superiores Aragón, se diseñó previamente un Equipo Mecánico-Óptico manual con el fin de realizar pruebas de flexión en materiales y tomar micrografías en la muestra al momento de realizar la prueba, figura 1.11.

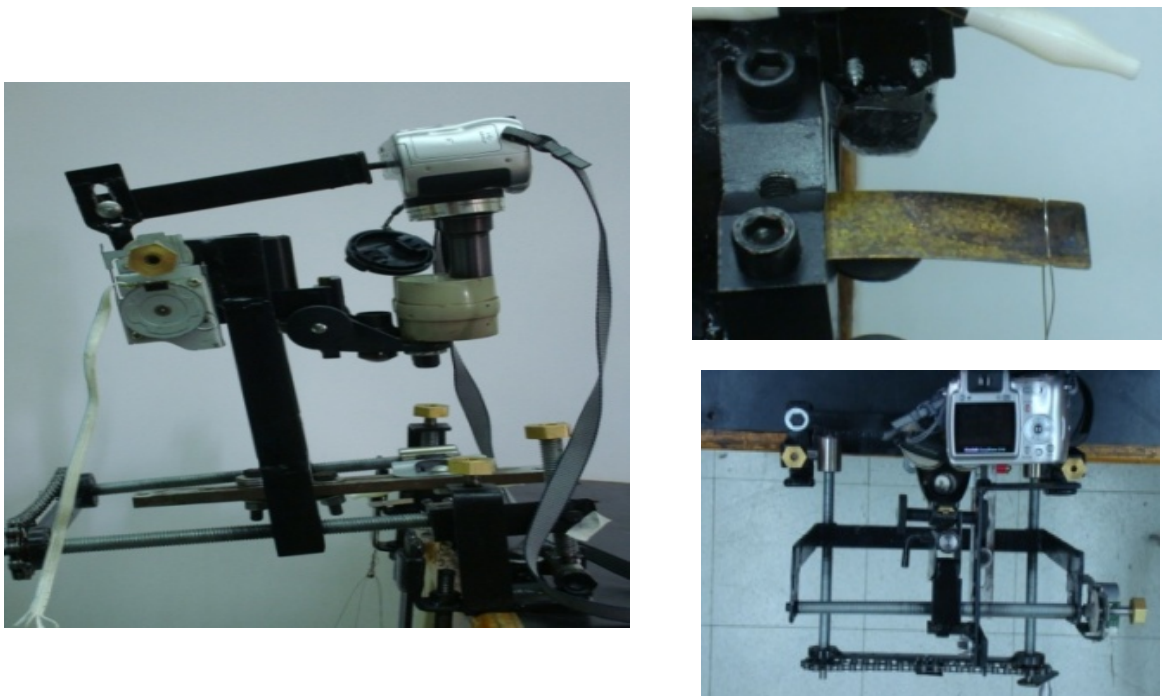


Figura 1.11 Primer diseño del Equipo Mecánico-Óptico.

Para llevar a cabo la prueba, se adaptó un microscopio y una cámara al equipo. Uno de los requerimientos principales que debe cubrir el equipo para realizar la prueba, es que el microscopio enfoque y siga a la muestra durante la aplicación de la fuerza, manteniendo su posición de manera perpendicular a ella, como lo muestra la figura 1.12. Además, se debe posicionar al microscopio en regiones donde sea posible dar seguimiento a ciertos granos de interés.

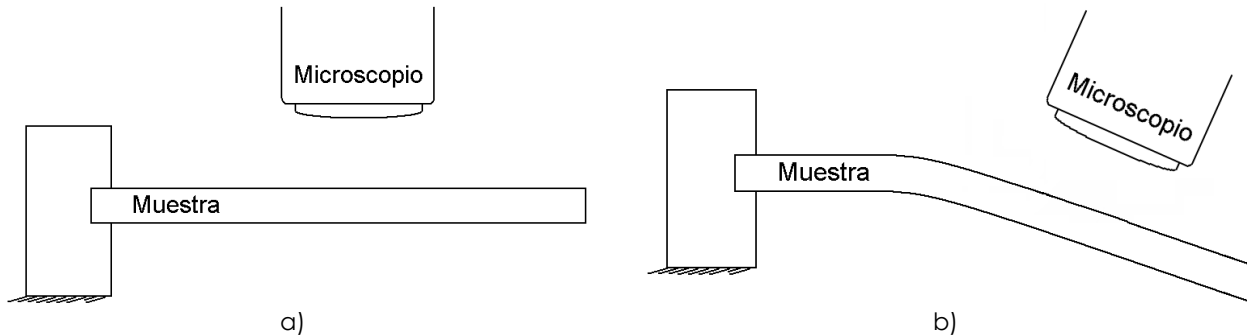


Figura 1.12 Posicionamiento de microscopio: (a) sin flexión, (b) en flexión.

Al realizar pruebas con este equipo, se concluyó que los resultados no eran apropiados, ya que no cumplía con los requerimientos necesarios, por lo cual; se planteó el rediseño del equipo, con el objetivo de cubrir en su totalidad los requerimientos.

Este nuevo diseño denominado '**Equipo Mecánico-Óptico para Pruebas de Flexión In Situ En Materiales**', se basa principalmente en una mesa de posicionamiento de cuatro grados de libertad para posicionar un microscopio electrónico, este equipo logra dar seguimiento perpendicular a la muestra para realizar un barrido de micrografías, figura 1.13.

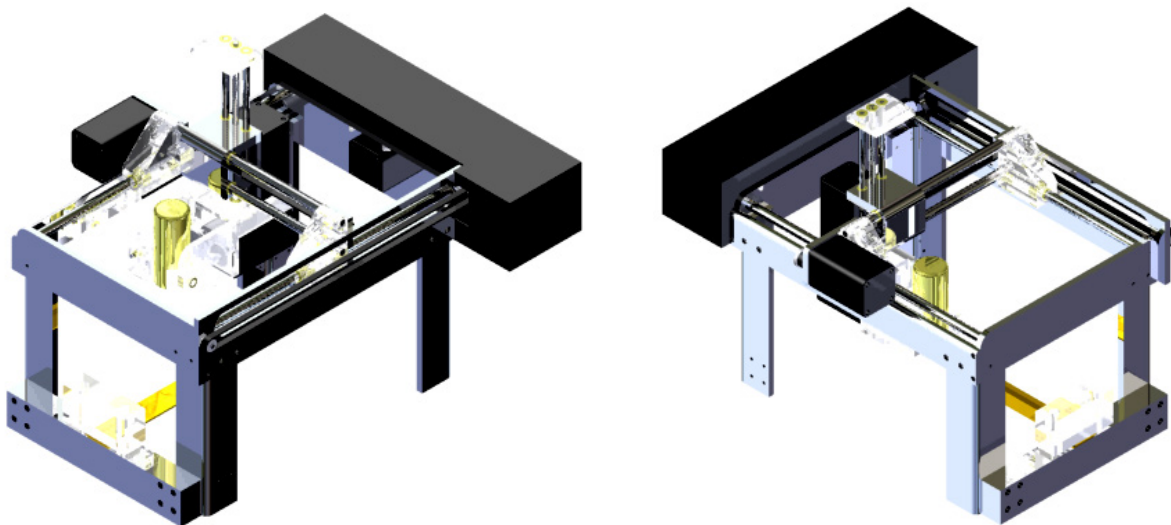


Figura 1.13 Equipo Mecánico-Óptico De Pruebas De Flexión In Situ En Materiales.

Este equipo fue diseñado y desarrollado para el estudio estructural a nivel microscópico de cualquier material, sin embargo; la principal aplicación es para obtener micrografías de

metales policristalinos, y como su nombre lo indica, las micrografías se adquieren en pruebas de flexión.

En la figura 1.14 se pueden ver los cuatro grados de libertad, cada grado corresponde a un eje de movimiento, estos ejes son: x , y , z y w . Donde los ejes x e y , se encargan de la posición del microscopio sobre la muestra, el eje z acerca o aleja el microscopio de la muestra, y por último, el eje w rota al microscopio.

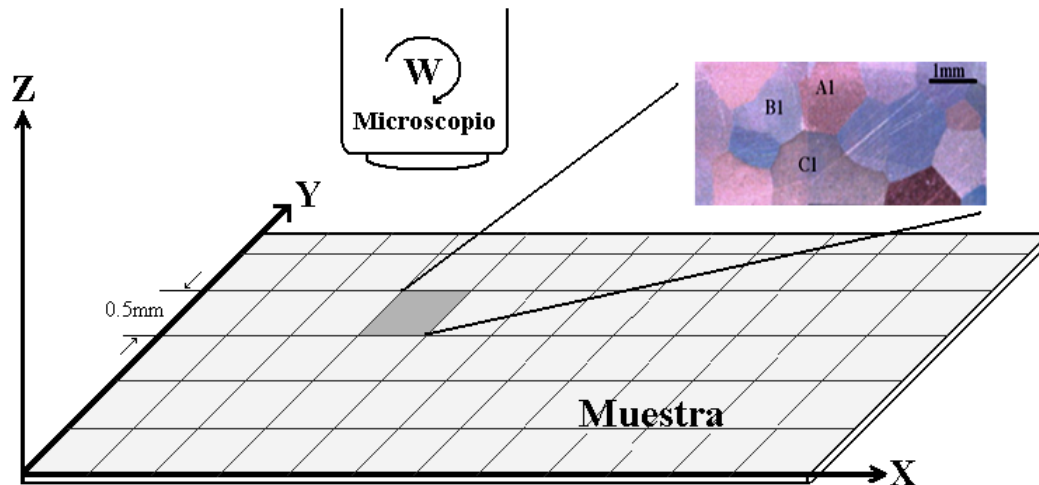


Figura 1.14 Ejes de movimiento del equipo Mecánico-Óptico.

Con este equipo se pueden tener desplazamientos lineales y/o rotacionales del orden de milímetros en cada uno de sus ejes, permitiendo así, un posicionamiento milimétrico del microscopio. Los cuatro ejes, pero principalmente los ejes x e y se manejan como ejes coordenados, donde la muestra es dividida virtualmente en n posiciones o coordenadas para poder tener un control de la ubicación del microscopio.

Dados los pequeños desplazamientos que debe realizar este microscopio para posicionarse en una u otra coordenada, es complicado manipular el equipo de manera manual a través de manivelas, volantes ó engranes como lo hacen la mayoría de las mesas de posicionamiento, ya que al realizarlo de esta manera es lento e impreciso, por lo tanto; el Equipo Mecánico-Óptico cuenta con un sistema de engranes acoplado a cada eje de movimiento para una futura integración de algún tipo de motor de corriente continua (el cual se explicará más adelante) que cubra como requerimiento principal un torque mínimo de $0.2 \text{ N}\cdot\text{m}$ para realizar los movimiento de cada uno de los ejes.

La relación de engranes que tiene cada eje convierte el movimiento angular de 1° a un movimiento lineal de 0.07 mm y para esta aplicación se indica que es suficiente tener desplazamientos de 0.5 mm , dado que el tamaño promedio de cada grano es superior a 1 mm .

El microscopio con el que cuenta el Equipo Mecánico-Óptico es del fabricante Dino-Lite, modelo AM413MT y tiene como características principales:

- Microscopio digital.
- 1.3 MPíxeles
- Resolución de 1280×1024
- Zoom continuo de $1 - 50x, 200x$
- Conexión USB 2.0



I.4 PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN

En esta aplicación se ha planteado la importancia y la forma de estudio de los materiales que, en particular, se basa en pruebas de flexión en metales policristalinos. También se ha descrito el equipo que se tiene para realizar la toma de las micrografías, sin embargo, hasta ahora este equipo carece de motores y de un sistema de control electrónico que permita controlarlos, además, se necesita de un programa de computadora que permita manipular las micrografías obtenidas, por consiguiente, se plantea las siguientes necesidades:

- Seleccionar el tipo de motor adecuado e implementar un sistema de control electrónico que permita manipularlos de forma fácil, rápida y precisa a través de una interfaz de usuario.
- Desarrollar un sistema de iluminación variable para el enfoque de las micrografías.
- Se requiere de un programa de computadora capaz de manipular las micrografías para obtener las coordenadas de la frontera de cada grano que a su vez deben ser almacenadas en la computadora.

I.4.1 Planteamiento del problema

En base a las necesidades anteriores se plantea el siguiente problema:

Diseñar y desarrollar el hardware y software necesario para controlar los 4 motores que en su momento sean seleccionados, implementando un sistema de posicionamiento coordinado para cada eje, teniendo como interfaz de usuario un programa de computadora. Adicionalmente se propone una aplicación en la cual se puedan manipular las micrografías obtenidas, dicha aplicación será capaz de almacenar los datos obtenidos en el formato requerido.

Dado el problema, se presentan los objetivos y requerimientos con que debe contar el hardware y software del sistema:

- a) Selección de los motores adecuados para el movimiento de cada eje.
- b) Diseño de la tarjeta electrónica para el control de los motores.
- c) Diseño del sistema de control y la comunicación con la computadora.
- d) Diseño de un sistema de iluminación.
- e) Diseño de la interfaz de usuario para el control de cada eje de movimiento.
- f) Diseño de un programa de computadora para la manipulación de las micrografías.

I.4.2 Propuesta de solución

Para cubrir las necesidades y requerimientos establecidos, se presenta la propuesta de solución con la que se cubrirá cada uno de los puntos anteriores. El sistema se ha dividido en dos partes, hardware y software, en este apartado sólo se mencionaran los componentes propuestos para

alcanzar los objetivos, posteriormente, en el capítulo 2 se detallarán los fundamentos teóricos y los principios de funcionamiento de cada uno. En el tercer capítulo se darán a conocer los fabricantes, su configuración y las características por las que fueron seleccionados dichos componentes.

Hardware. Esto contempla la parte física del sistema, la cual abarca la selección de los motores, el diseño y desarrollo de una tarjeta electrónica para el sistema de control, el medio de comunicación con la computadora y el sistema de iluminación.

- a) Considerando que la aplicación requiere de pequeños desplazamientos angulares para lograr posicionamientos del orden de milímetros y que se necesita cubrir con un mínimo de torque de 0.2 N•m dado que ya se tiene implementado un arreglo de engranes, se propone el uso de motores de pasos.
- b) La tarjeta electrónica para el control de los motores contempla el diseño de la fuente de alimentación, la etapa de potencia y el aislamiento óptico entre el sistema de control y los motores. Dada la naturaleza de los motores de funcionar con el flujo de corriente, se propone el uso de Transistores Bipolares de Unión como etapa de potencia y el desarrollo de una fuente de alimentación basada en un regulador conmutado. Para el aislamiento óptico se usarán optoacopladores con salida transistor.
- c) Para la etapa de control se usará un microcontrolador de 8 bits, ya que estos pueden ser programados fácilmente y presentan una alta velocidad de procesamiento e incluye un módulo que permite su comunicación con la computadora.
- d) El sistema de iluminación será por medio de diodos emisores de luz, ya que estos presentan bajo consumo energético y pueden variar su intensidad lumínica.

Software. Comprende el desarrollo del programa de computadora que servirá para controlar a los motores y manipular las micrografías obtenidas.

- e) La interfaz de usuario se desarrollará bajo el ambiente de programación de LabVIEW, ya que este ambiente brinda potencia y facilidad a la hora de programar. Dicha interfaz debe permitir al usuario el control de la posición de cada motor de una manera fácil, rápida y precisa.
- f) Para la manipulación de las micrografías se desarrollara un programa de computadora que permita obtener las coordenadas de la frontera de cada grano y almacenarlas en un archivo de texto con el formato requerido, dicho programa también será desarrollado bajo el ambiente de programación de LabVIEW.



Capítulo **2**

Fundamentos Teóricos

Objetivo particular. El presente capítulo tiene como finalidad conocer los fundamentos teóricos de los componentes y/o dispositivos utilizados para la tarjeta electrónica y el sistema de control, se describen los ambientes de programación usados.

II.1 FUENTES DE ALIMENTACIÓN

Una fuente de alimentación es el circuito, dentro de los equipos electrónicos, que convierte la tensión de entrada alterna en una tensión de salida continua casi perfecta. Incluye rectificadores, filtros y reguladores de voltaje. Hoy en día la tendencia consiste en diseñar fuentes de alimentación de bajo voltaje y alta corriente.

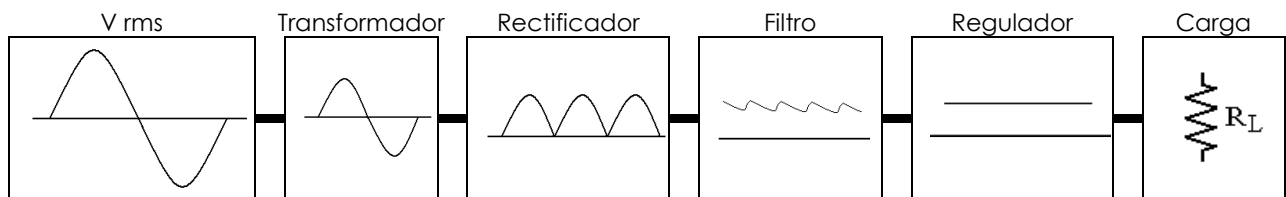


Figura 2.1 Diagrama de bloques que muestra las partes de una fuente de alimentación.

En la figura 2.1 se muestra un diagrama de bloques que contiene las etapas de una fuente de alimentación típica para alimentar una carga, donde se ejemplifica la forma de onda del voltaje correspondiente a su salida. El voltaje de Corriente Alterna (CA), de 127 Vrms, se conecta a un transformador, que normalmente disminuye el voltaje en una relación de 10:1. Luego, un rectificador de diodos proporciona un voltaje rectificado de onda completa que inicialmente se filtra mediante un capacitor. Este voltaje resultante cuenta con cierta oscilación (rizo) o variación de voltaje de CA.

Al utilizar un circuito rectificador para una señal de CA, se tiene como salida resultante un voltaje de Corriente Directa (CD) **pulsante**, el cual todavía no es apropiado para usarlo como fuente de alimentación en sistemas electrónicos digitales; se requiere además un circuito de filtrado que proporcione un voltaje de CD más estable.

Un circuito regulador recibe en su entrada un voltaje de CD pulsante con el objetivo de minimizar o eliminar el rizo, también trata de estabilizarlo incluso si el voltaje de CD a la entrada tiene pequeñas variaciones de voltaje o cambia la carga conectada.

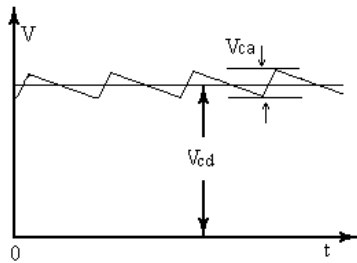


Figura 2.2 Forma de onda que muestra las componentes de CA y CD.

La figura 2.2 muestra las características de la forma de onda del voltaje de salida de un filtro conectado a una carga, esta señal cuenta con un valor de CD y cierta variación de CA (ruido). La eficiencia del circuito de filtro es mejor mientras más pequeña sea la variación de CA con respecto al nivel de CD.

Un factor a considerar en el diseño de fuentes de alimentación es la cantidad de voltaje de CD que cambia entre las condiciones con carga y sin carga, a este factor se denomina **regulación de voltaje**.

$$\% \text{ Regulación de Voltaje} = \frac{\text{Voltaje sin carga} - \text{Voltaje con carga completa}}{\text{Voltaje con carga completa}} \times 100\%$$

Si el valor del voltaje con carga completa es el mismo que el del voltaje sin carga, la regulación de voltaje será de 0%, mientras menor sea la regulación de voltaje, mejor será el circuito de la fuente de alimentación.

II.1.1. Filtro de capacitor

Un circuito de filtro muy funcional es el de capacitor mostrado en la figura 2.3a, en el que un capacitor se conecta a la salida de un rectificador para obtener un voltaje de CD más estable. La figura 2.3b indica el voltaje de salida de un rectificador antes de que la señal sea filtrada, y la figura 2.3c muestra la forma de onda resultante después de que el filtro de capacitor se conecta a la salida del rectificador.

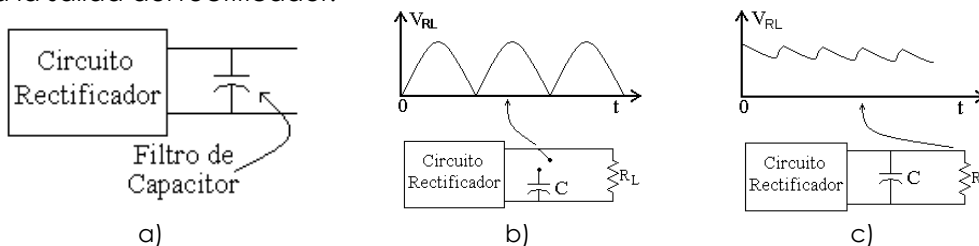


Figura 2.3 (a) Filtro simple de capacitor, (b) Voltaje rectificado de onda completa, (c) Voltaje de salida filtrado.

Cabe mencionar que cuando no se conecta carga a través del capacitor, la forma de onda de salida sería, idealmente, de un nivel de CD constante con valor igual al voltaje pico (V_p) del circuito rectificador.

En la figura 2.4 se muestran los periodos de la forma de onda de salida del circuito filtro de capacitor conectado a una carga. En esta figura el tiempo T_1 es el tiempo durante el cual el valor del voltaje DC aumenta mientras se carga el capacitor hasta el voltaje rectificado pico V_p . El tiempo T_2 es el intervalo de tiempo durante el cual el voltaje rectificado cae por debajo del voltaje pico, y el capacitor se descarga a través de la carga, dado que el ciclo de carga-descarga sucede para cada medio ciclo de una señal senoidal de periodo T .

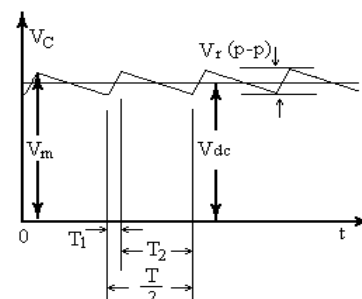


Figura 2.4 Forma de onda a través de un filtro de capacitor.

A partir de esto se puede concluir que para mejorar el desempeño del filtro de capacitor sólo es necesario incrementar el tamaño del capacitor o conectar más de uno. Sin embargo, el capacitor también afecta a la corriente pico extraída a través de los diodos rectificadores, por lo tanto, mientras mayor sea el valor del capacitor, mayor será el pico de la corriente que circula a través de los diodos.

Ahora, con este filtro de capacitor ya se tiene un voltaje adecuado para poder conectar los dispositivos electrónicos. Para algunos dispositivos que necesitan de un voltaje y corriente estable esta fuente no es la más eficiente. Para esto se necesita de algún circuito que regule el voltaje y la corriente de salida. A continuación se describirán brevemente estos circuitos reguladores.

II.1.2. Reguladores lineales integrados

Existe una gran variedad de Circuitos Integrados (CI) lineales para la regulación de tensión. Todos son reguladores serie dado que tienen un rendimiento mayor que los reguladores en derivación. Tienen como elementos principales un transistor de paso que regula la corriente y un comparador para mantener estable al voltaje de la salida (figura 2.5). La mayoría de los reguladores integrados son de una tensión de salida fija positiva, fija negativa o ajustable. Los reguladores integrados utilizados con mayor frecuencia son los que tienen únicamente tres terminales de conexión: tensión de entrada, tensión de salida y referencia. Estos CI's sólo necesitan de dos capacitores externos para su funcionamiento.

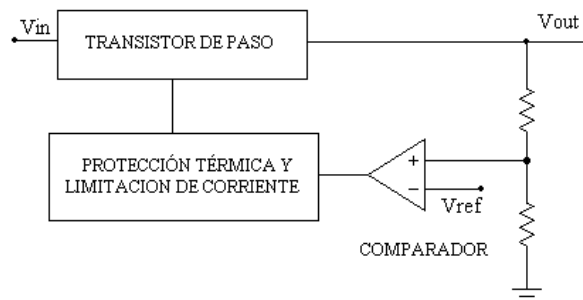


Figura 2.5 Diagrama de bloques funcional de un CI regulador de tres terminales.

A continuación se describen los aspectos más importantes de los CI's reguladores de voltaje de la serie 78XX, ya que estos satisfacen las necesidades de estabilidad y regulación de voltaje y corriente, además de su sencilla implementación y bajo costo. El número de parte de esta serie se complementa con algunos caracteres alfabéticos según se seleccione el fabricante, sin embargo los conceptos son los mismos.

La serie 78XX (donde XX = 05, 09, 12, 15, 18 o 24) está compuesta por reguladores de tensión de tres terminales. Estos reguladores de tensión fija positiva están fabricados para obtener diferentes tensiones fijas en un rango que va de los 5 a los 24V. La figura 2.6 muestra la conexión de un 78XX, donde: la terminal 1 es la entrada, la 3 es la salida y la 2 es la referencia.

Cuando un circuito integrado está conectado a unos cuantos centímetros del filtro capacitivo de la fuente de alimentación no regulada, la inductancia de las terminales de conexión puede producir oscilaciones dentro del integrado. Es por esta razón que los fabricantes recomiendan el uso de un capacitor de desacoplo C_1 , en la terminal 1 (figura 2.6), y para mejorar la respuesta transitoria de la tensión de salida regulada, se usa un capacitor de desacoplo C_2 , que finalmente funciona como un filtro paso bajas. Valores típicos para cualquier capacitor de desacoplo están entre 0.1 y $1\mu\text{F}$.

Cualquier dispositivo de la serie 78XX necesita una tensión de **dropout** de 2 ó 3 V, dependiendo de la tensión de salida. La tensión dropout quiere decir que la tensión de entrada debe ser al menos 2 ó 3 V mayor que la de salida, de otra manera, el circuito dejará de regular. Además, hay una tensión máxima debido a la disipación de potencia. La hoja de especificaciones proporciona las tensiones de entrada mínimas y máximas para las tensiones de salidas prefijadas.

Voltaje de salida, el fabricante especifica la tensión nominal de salida, pero también proporciona la tensión máxima y mínima en la que puede variar tensión dependiendo de la carga.

Corriente de salida, es la corriente máxima que puede regular este dispositivo.

Regulación de voltaje, normalmente dada en mV. Este valor especifica que la tensión de salida puede variar en un rango de $\pm mV$ a partir de su valor nominal.

Diferencia de voltaje, es la tensión de dropout.

Rechazo al rizado, dado en dB, este parámetro reducirá el rizado de la tensión de entrada.

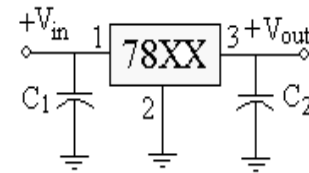


Figura 2.6 Conexión del regulador de voltaje.

Hay que tomar en cuenta que existen reguladores de tensión que ofrecen una salida negativa, los cuales pertenecen a la serie 79XX, y básicamente cumplen con las mismas características de la serie 78XX.

En este tipo de circuitos el dispositivo que se encuentra entre la entrada y la salida es un transistor de paso (figura 2.5) que realiza la función de regulador. Al existir una caída de tensión entre la entrada y la salida de estos reguladores, hay una pérdida de potencia en ellos.

Esta pérdida de potencia es: $P_P = (V_{ENTRADA} - V_{SALIDA}) \times I_{CARGA}$

Como hay pérdida de potencia, la eficiencia de estos reguladores es baja. Peor aun es si la diferencia de tensión entre la entrada y la salida es grande y existe un consumo de corriente elevado (ver la fórmula anterior). Normalmente esta eficiencia está entre el 50 y el 70 %.

En un regulador convencional, el transistor de paso trabaja en la región activa, en consecuencia hay una disipación de potencia constante.

$$P = V_{CE} \times I_C$$

Una opción a estos reguladores de baja eficiencia son los convertidores CD-CD. En estos, el transistor trabaja un tiempo en corte y otro en saturación en forma continua. Al trabajar de esta manera el consumo de potencia disminuye.

II.1.3. Convertidores CD-CD

II.1.3.1 Principio de funcionamiento

En un convertidor CD-CD sin regular típico, la entrada de corriente se aplica a un oscilador de onda cuadrada. El valor pico a pico de la señal cuadrada de salida es proporcional a la tensión de entrada. La señal cuadrada excita la bobina primaria de un transformador, como aparece en la figura 2.7. Cuanto mayor sea la frecuencia, menor será el transformador y los componentes del filtro. Para frecuencias muy altas es difícil producir una onda cuadrada con los flancos verticales, normalmente la frecuencia de la onda cuadrada están entre los 10 y 100 KHz.

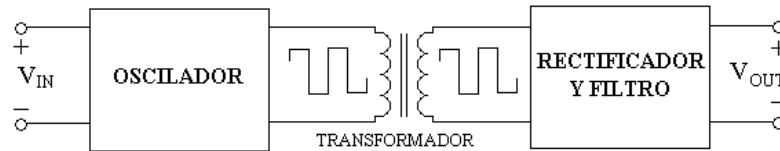


Figura 2.7 Diagrama de bloques funcional de un convertido CD-CD sin regular.

Para mejorar el rendimiento se utiliza un tipo especial de transformadores en los convertidores CD-CD más caros. Consiste en un núcleo toroide con un lazo de histéresis regular, esto produce una tensión secundaria en forma de onda cuadrada. Dicha tensión puede, posteriormente; ser rectificadas y filtradas para obtener una tensión de salida continua. Seleccionando diferentes radios de giro se puede modificar arriba o abajo la tensión en el secundario. De esta forma se puede construir un convertidor CD-CD que modifique la tensión de entrada, aumentándola o disminuyéndola.

Existen muchas formas de diseñar un convertidor CD-CD, dependiendo de si se usan transistores bipolares o FET, de la frecuencia de conmutación, o de que la tensión de entrada se convierta en mayor o menor tensión a la salida. En algunos convertidores CD-CD, la tensión de salida no está regulada, esto es típico en convertidores CD-CD económicos. Estos convertidores requieren de una tensión de entrada fija, ya que tampoco incluyen regulación de tensión a la entrada.

Los convertidores CD-CD en alta frecuencia son circuitos de potencia donde los semiconductores, o llaves de potencia, conmutan a una frecuencia mucho mayor que la de variación de las formas de onda de entrada y salida del convertidor. Esto permite emplear filtros paso-bajos para eliminar los componentes armónicos no deseados. No podemos usar la inversión de la forma de onda de entrada para bloquear los semiconductores, como se hace con los conmutados por la red. Los interruptores deberán entonces controlarse al conducir y bloquear. Los convertidores CD-CD forman parte de este grupo de convertidores, controlan el flujo de energía entre dos sistemas de corriente directa. Ejemplos de aplicación son: fuentes de alimentación CD (Switch Mode Power Supplies, SMPS) para equipamiento electrónico, control de máquinas eléctricas de corriente directa, etc.

II.1.4. Reguladores conmutados

Los reguladores conmutados provienen de la clase general de convertidores CD-CD, dado que transforman una tensión de entrada continua en otra de salida continua, superior o inferior a la de la entrada. Pero estos incluyen regulación de tensión, usando típicamente Modulación en Ancho de Pulsos (también llamado PWM, Pulse Width Modulation), controlando el corte y la conducción de un transistor. Con cambios en el ciclo de trabajo, los reguladores conmutados pueden mantener la tensión de salida constante bajo condiciones de variación de la red o la carga. Su frecuencia de conmutación va desde los 10 hasta los 200 KHz, incluso hasta los MHz, por esta razón incluyen protección contra interferencias de RF.

En este tipo de regulador, el transistor de paso está conmutando entre saturación y corte. Cuando el transistor está en corte, la potencia disipada es virtualmente cero. Cuando el transistor se satura, la potencia disipada permanece muy baja, ya que $V_{CE(sat)}$ es de bajo voltaje. Estos reguladores pueden alcanzar rendimientos de entre el 75 y 95 %. Por su gran rendimiento y pequeño tamaño, son extensamente usados. Existen diferentes topologías sobre los reguladores conmutados, la tabla 2.1 muestra los nombres de estas topologías y sus características principales. Las tres primeras son las más básicas, usan el menor número de componentes y pueden soportar potencias de carga del orden de 150W. Ya que su complejidad es baja, son comúnmente usadas con reguladores conmutados integrados.

Tabla 2.1 Reguladores conmutados.

Topología	Conversión	Transformador	Diodos	Transistores	Potencia (W)	Complejidad
Reductora	Reduce	No	1	1	0-150	Baja
Elevadora	Aumenta	No	1	1	0-150	Baja
Inversora	Ambas	No	1	1	0-150	Baja
Flyback	Ambas	Si	1	1	0-150	Media
Half-forward	Ambas	Si	1	1	0-150	Media
Contrafase	Ambas	Si	2	2	100-1000	Alta
Medio puente	Ambas	Si	4	4	100-500	Alta
Puente completo	Ambas	Si	4	4	400-2000	Muy alta

A continuación se describirá el funcionamiento básico de las dos primeras topologías, ya que son las más utilizadas.

II.1.4.1 Regulador reductor (Buck o Step-Down)

La figura 2.8a presenta un regulador reductor, este regulador siempre disminuye la tensión de entrada. Se usa como dispositivo de conmutación un transistor (Q), ya sea bipolar o FET. La señal rectangular de salida del modulador de pulso abre y cierra el conmutador. El comparador controla el ciclo de trabajo de los pulsos. Cuando el pulso está en nivel alto, el conmutador está cerrado. Esto polariza en inversa al diodo, por lo que toda la corriente de entrada circula a través de la inductancia. Esta corriente crea un campo magnético alrededor de dicha inductancia. La cantidad de energía almacenada en el campo magnético viene dada por:

$$\text{Energía} = 0.5 Li^2$$

La corriente que atraviesa la inductancia también carga al capacitor y proporciona corriente a la carga. Mientras el conmutador esté cerrado, la tensión en la inductancia tiene la polaridad como aparece en la figura 2.8b, como la corriente en la bobina aumenta, se va almacenando más energía en el campo magnético.

Cuando cae el pulso, se abre el conmutador. En este instante, el campo magnético en torno a la inductancia comienza a caer e induce una tensión inversa en la misma, como se muestra en la figura 2.8c. Esta tensión inversa se llama **golpe inductivo**. Debido a este golpe inductivo, el diodo se polariza en directa y la corriente que circula por la inductancia continúa haciéndolo en el mismo sentido. En este momento, la inductancia está devolviendo al circuito la energía almacenada. En otras palabras, la inductancia actúa como una fuente y continúa suministrando corriente a la carga. La corriente circulará por la bobina hasta que ésta entregue toda la energía al circuito o hasta que el conmutador se cierre de nuevo, con lo que se volverá a comenzar el ciclo. En cualquier caso, el capacitor proporcionará también corriente a la carga durante parte del tiempo que el conmutador esté abierto. De esta forma, se minimiza el rizado en la carga.

La tensión de salida del regulador depende del circuito PWM, ya que si se aumenta el ciclo de trabajo se aumentará la tensión de salida. Así mismo, el ciclo de trabajo de la señal PWM depende de la realimentación negativa en el comparador. De esta manera, cualquier intento de cambio en la tensión de salida, ya sea debido a cambios en la red o en la carga, será completamente compensado por la realimentación negativa. Por ejemplo, si la tensión de salida trata de aumentar, la tensión de realimentación disminuye la salida del comparador, esto disminuye el ciclo de trabajo y la tensión de salida disminuye.

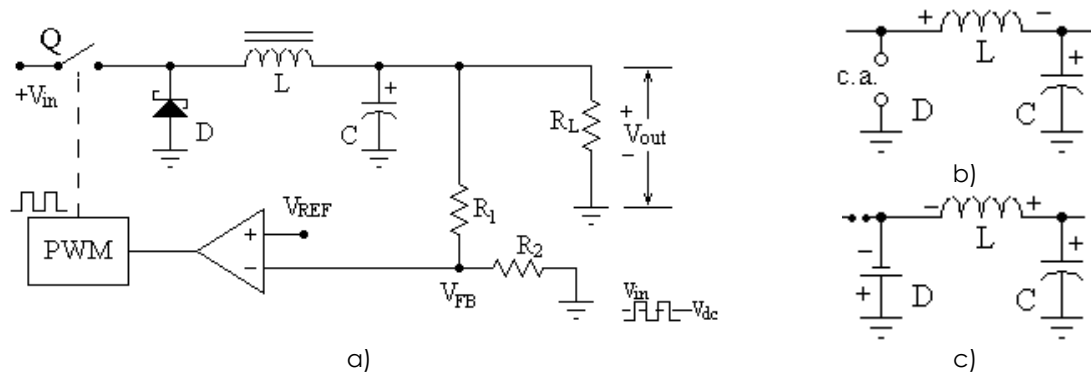


Figura 2.8 Regulador reductor: (a) Circuito simplificado, (b) Polaridad con el conmutador cerrado, (c) Polaridad con el conmutador abierto.

II.1.4.2 Regulador elevador (Boost o Step-up)

En la figura 2.9a se muestra un regulador elevador, este tipo de regulador siempre aumenta la tensión de salida. La teoría de su funcionamiento es similar a la del regulador reductor en algunos aspectos. Por ejemplo, cuando el pulso está en un nivel alto, el conmutador está cerrado y la energía se almacena en el campo magnético, como se vio anteriormente.

Cuando el pulso pasa a nivel bajo se abre el conmutador, de nuevo; el campo magnético inducido en la bobina comienza a disminuir y provoca una tensión inversa, como lo muestra la figura 2.9b. Observe que la tensión de entrada V_{in} se suma ahora al golpe inductivo. Esto significa que el pico de tensión en el extremo derecho de la inductancia es:

$$V_P = V_{in} + V_{golpe}$$

El golpe inductivo depende de la cantidad de energía que se haya almacenado en el campo magnético. Dicho de otra manera, V_{golpe} es proporcional al ciclo de trabajo. Con una tensión fija en la entrada, llegará una tensión de forma rectangular a la entrada del filtro con condensador de la figura 2.9c. De esta forma, la tensión de salida regulada es aproximadamente igual al pico de tensión dado por la ecuación anterior. Como V_{golpe} es siempre mayor que cero, V_P es siempre mayor que V_{in} . Es por esto por lo que un regulador elevador siempre aumenta la tensión de salida.

La regulación con topología elevadora es similar a la topología reductora. Debido a la alta ganancia del comparador, la realimentación es casi igual a la tensión de referencia. De esta forma, la tensión de salida regulada para ambas topologías viene dada por la siguiente ecuación:

$$V_{out} = \frac{R_1 + R_2}{R_1} V_{REF}$$

Si la tensión de salida trata de aumentar (V_{out}), hay menor realimentación, menor salida en el comparador, un ciclo de trabajo menor y un menor golpe inductivo. Esto disminuye el pico de tensión, lo que compensa el intento de aumento en la tensión de salida. Si la tensión de salida trata de disminuir, el aumento de la caída de realimentación aumentará el pico del mismo, lo que compensa este intento de disminución.

Es importante resaltar el uso de diodos Schottky en ambos reguladores conmutados para mejorar su rendimiento. Como el diodo Schottky tiene una barrera de potencial baja (aproximadamente 0.25 V), la disipación será menor. Además de su fácil conmutación entre polarizaciones directa e inversa a altas frecuencias, del orden de los Mhz.

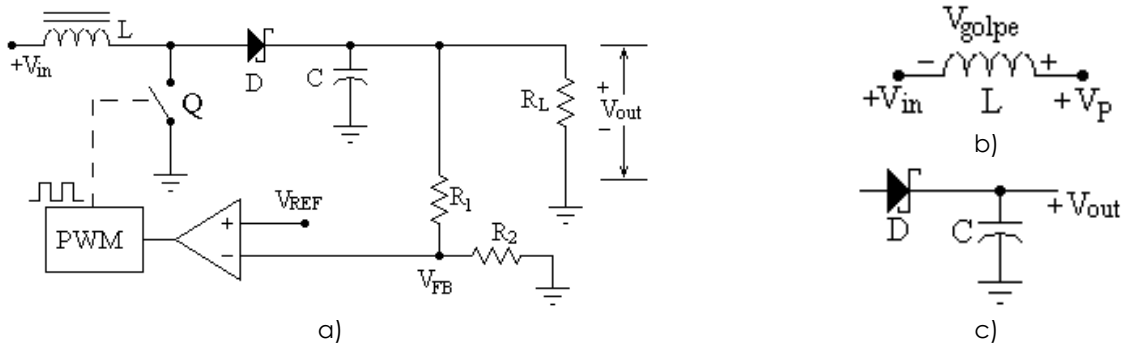


Figura 2.9 Regulador elevador: (a) Circuito simplificado, (b) La tensión de golpe se suma a la salida cuando Q está abierto, (c) Filtro con capacitor de entrada.

II.2 MOTORES DE PASOS

Los motores de pasos (Stepper Motors) son ideales para la construcción de mecanismos en donde se requieren movimientos con cierta precisión; tienen un comportamiento diferente al de los motores de corriente continua. En primer lugar, no giran libremente por sí mismos. Los motores de pasos, como lo indica su nombre, avanzan girando por pequeños pasos. Estos pasos corresponden a un tren de pulsos generados con una secuencia previamente definida aplicados a cada una de las bobinas que componen al estator. Cada vez que a alguna de estas bobinas se le aplica un pulso, el motor se desplaza un paso, y queda fijo en esa posición mientras se mantenga polarizada la bobina. Dependiendo de las características constructivas del motor, este paso puede ser desde 90° hasta 0.9° . Con este parámetro podemos obtener el número de pasos necesarios para que el motor realice una vuelta completa, si se divide 360° (una revolución) entre los grados por paso que tiene el motor. Por ejemplo, un motor con una resolución de 1.8° por paso, necesita de 200 pasos para realizar una vuelta.

Si se varía la frecuencia con la que se aplica el tren de pulsos, también estaremos variando la velocidad con que se mueve el motor, lo que nos permite realizar un control de velocidad en función de la frecuencia. Además, si se invierte el orden del tren de pulsos se estará realizando una inversión en el sentido de giro del motor. En los motores de pasos su mayor capacidad de torque se produce a baja velocidad. Sin embargo, una desventaja de estos motores de pasos es que presentan una velocidad angular limitada. Dicha limitación surge del tiempo requerido para alcanzar el equilibrio cuando gira un paso. Si dicho tiempo es demasiado corto, es decir, si la frecuencia de los pulsos es demasiado elevada, el motor puede no encontrar nunca la posición de equilibrio y se perdería el control sobre él, de esta forma el motor no se moverá apropiadamente o puede ser que la inercia se imponga a la frecuencia de los pulsos produciendo un zumbido y no se mueva, o puede ser que gire en sentido contrario al deseado.

II.2.1 Tipos de motores de pasos

Los motores de pasos se dividen en dos categorías principales: de imán permanente y de reluctancia variable. También existe una combinación de ambos, a estos se les llama híbridos.

De imán Permanente. Este tipo de motores son los más usados. El rotor está formado por un imán permanente, en forma de disco, y en cuya superficie se encuentran mecanizados un determinado número de dientes. El estator tiene forma cilíndrica, y en su interior se encuentran diversos bobinados, que al ser alimentados secuencialmente generan un campo magnético

giratorio. Como consecuencia de atracción-repulsión, el rotor se orientará dentro de este campo magnético giratorio, tratando de estabilizar sus polos con los generados en las bobinas.

Los motores paso a paso de imán permanente se dividen a su vez en distintos tipos, diferenciados por el tipo de bobinado. Existen entonces motores de pasos de imán permanente *Unipolares*, y motores de pasos de imán permanente *Bipolares*.

De reluctancia variable. Poseen un rotor de hierro dulce que, en condiciones de excitación del estator y bajo la acción de su campo magnético, ofrecen menor resistencia a ser atravesado por su flujo en la posición de equilibrio. Su mecanización es similar a los de imán permanente y su principal inconveniente radica en que en condiciones de reposo (sin excitación) el rotor queda en libertad de girar y, por lo tanto, su posicionamiento de régimen de carga dependerá de su inercia y no será posible predecir el punto exacto de reposo.

Híbridos. Este tipo de motores son una mezcla de los dos anteriores. El rotor está formado por una serie de anillos de acero dulce que tienen en su superficie un número de dientes ligeramente distinto a los del estator. Dichos anillos están montados sobre un eje que es un imán permanente.

En la figura 2.10 se puede ver el aspecto externo e interno de los motores de pasos de imán permanente.



Figura 2.10 Motores de pasos de imán permanente: (a) Aspecto externo (b) Rotor (c) Estator.

II.2.2 Motores bipolares

Estos motores están formados por dos bobinas (figura 2.11a), y se requiere del cambio de dirección del flujo de corriente a través de las bobinas (de ahí surge el nombre de bipolares), en la secuencia apropiada, para realizar un movimiento. Se pueden reconocer externamente porque presentan cuatro cables, uno para cada extremo de cada bobina (figura 2.11b). En la figura 2.11c se puede apreciar la conexión para el control de estos motores mediante el uso de un puente en H (H-Bridge). Como se aprecia, será necesario un puente H por cada bobina del motor, es decir que para controlar un motor bipolar, necesitaremos usar dos puentes H.

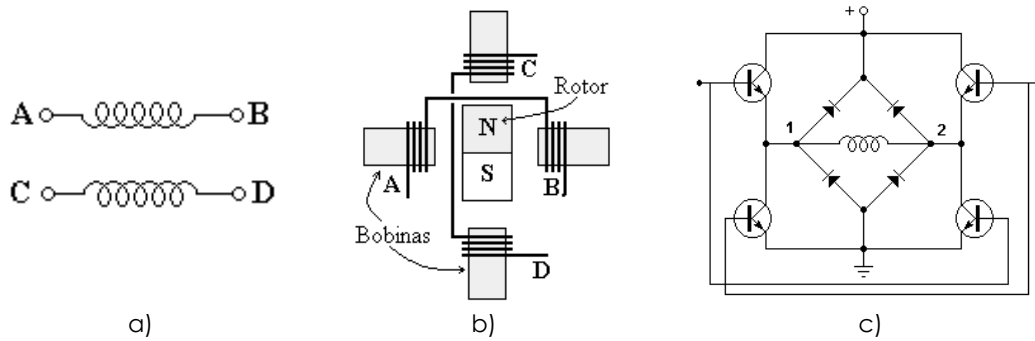


Figura 2.11 Motor Bipolar: (a) Bobinas, (b) Composición, (c) Control mediante un puente H.

II.2.2.1 Secuencias de funcionamiento

En modo de paso completo. En la tabla 2.2 se muestra la secuencia que se debe aplicar a este tipo de motores para su operación. Es importante mencionar que las bobinas A-B y C-D mostradas en la tabla 2.2 están conectadas como muestran las figuras 2.11 a y b. Como ventaja se puede resaltar un buen torque del motor, ya que siempre están polarizadas las dos bobinas y esto permite al estator mantenerse en su posición fijamente.

Tabla 2.2 Secuencia de excitación para un motor bipolar en modo de paso completo.

Paso	A	B	C	D	Posición
1	+V	-V	+V	-V	
2	+V	-V	-V	+V	
3	-V	+V	-V	+V	
4	-V	+V	+V	-V	

En modo de medio paso. Esta secuencia permite al motor realizar la mitad de un paso, mediante la activación de las dos bobinas y después sólo una, como se especifica en la tabla 2.3. Como ventaja se puede decir que se logran movimientos más suaves y exactos del motor, aunque se aumenta el número de pasos a realizar.

Para el motor ilustrado en la figura 2.11 b se necesita una secuencia de 4 pasos para completar una vuelta (en modo paso completo), es decir, esta secuencia es para un motor de 90° por paso. Para invertir el sentido de giro del motor solo basta con repetir la secuencia en orden inverso, del último al primer paso. La velocidad a la que gira el rotor está dada por la frecuencia de estos trenes de pulsos que recibe el controlador (puente H).

Tabla 2.3 Secuencia de excitación para un motor bipolar en modo de medio paso.

Paso	A	B	C	D	Posición
1	+V	-V	+V	-V	
2	+V	-V	0	0	
3	+V	-V	-V	+V	
4	0	0	-V	+V	
5	-V	+V	-V	+V	
6	-V	+V	0	0	
7	-V	+V	+V	-V	
8	0	0	+V	-V	

II.2.3 Motores unipolares

Los motores unipolares son relativamente más fáciles de controlar, gracias a que están compuestos por dos bobinas con una toma intermedia cada una (figura 2.12a), y su principal característica es que la intensidad de corriente que circula por cada bobina siempre lo hace en la misma dirección. Esta toma intermedia hace que se facilite el diseño del circuito de control, ya que permite el uso, en la parte de potencia, de un transistor único por cada una de las bobinas. En la figura 2.12c podemos apreciar un ejemplo de conexión para controlar al motor mediante el uso de transistores tipo Darlington. Estos motores suelen tener 5 o 6 cables de salida, dependiendo de su conexión interna. En los motores de 6 cables el común 1 y 2 de la figura 2.12a están separados, mientras que para los de 5 cables estos dos comunes están unidos a un solo cable, los otros 4 cables restantes corresponden a cada una de las bobinas.

El motor Unipolar presenta las mismas características que el Bipolar.

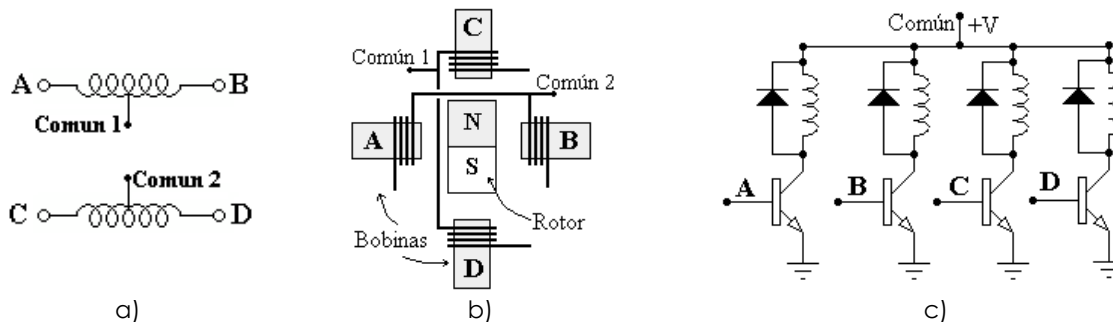


Figura 2.12 Motor Unipolar: (a) Bobinas, (b) Composición, (c) Control mediante transistores Darlington.

II.2.3.1 Secuencias de funcionamiento

En modo de paso completo. En la tabla 2.4 se muestra la secuencia que se debe aplicar a este tipo de motores para su operación, tomando en cuenta el circuito de la figura 2.12c. Las bobinas se van alimentando de dos en dos, siempre cuidando que no pueden estar alimentadas al mismo tiempo las dos partes de una misma bobina, es decir, A y B o C y D. Como ventaja también se puede resaltar un buen torque del motor, ya que siempre están polarizadas por lo menos dos bobinas y esto permite al estator mantenerse en su posición fijamente.

Tabla 2.4 Secuencia de excitación para un motor unipolar en modo de paso completo.

Paso	A	C	B	D	Posición
1	1	1	0	0	
2	0	1	1	0	
3	0	0	1	1	
4	1	0	0	1	

En modo de medio paso. En la tabla 2.5 se puede ver esta secuencia. Como ventaja se puede decir que se logran movimientos más suaves y exactos del motor, aunque se aumenta el número de pasos a realizar. Para invertir el sentido de giro del motor solo basta con repetir la secuencia en orden inverso, del último al primer paso. La velocidad a la que gira el rotor está dada por la frecuencia de estos trenes de pulsos que recibe cada transistor Darlington.

Tabla 2.5 Secuencia de excitación para un motor unipolar en modo de medio paso.

Paso	A	C	B	D	Posición
1	1	1	0	0	
2	0	1	0	0	
3	0	1	1	0	
4	0	0	1	0	
5	0	0	1	1	
6	0	0	0	1	
7	1	0	0	1	
8	1	0	0	0	

II.3 MICROCONTROLADORES

II.3.1 ¿Qué Es un microcontrolador?

Es un circuito integrado programable que contiene todos los componentes de una computadora. Se emplea para controlar el funcionamiento de una tarea determinada y, debido a su reducido tamaño, suele ir incorporado en el propio dispositivo al que gobierna. Esta última característica es la que le confiere la denominación de **controlador embebido** (embedded controller). En su memoria sólo reside un programa destinado a ejecutar un proceso determinado; a través de sus líneas de entrada/salida se pueden conectar sensores electrónicos o etapas de potencia que permitan controlar actuadores del dispositivo a controlar y todos los recursos complementarios disponibles pueden atender sus requerimientos.

Un microcontrolador (μC) es una computadora completa, aunque de limitadas prestaciones, que está contenido en un circuito integrado. El uso de un microcontrolador en un circuito

reduce notablemente el tamaño y número de componentes y, en consecuencia, disminuye el número de averías, el volumen y el peso de los equipos, entre otras ventajas.

El número de productos que funcionan con base en uno o varios microcontroladores aumenta de forma exponencial. No es aventurado pronosticar que habrá pocos elementos que carezcan de un microcontrolador. La industria Informática acapara gran parte de los microcontroladores que se fabrican. Casi todos los periféricos de la computadora, desde el ratón o el teclado hasta la impresora, son controlados por el programa de un microcontrolador. Los electrodomésticos (lavadoras, hornos, televisores, dvd's, aparatos musicales, etc.) incorporan numerosos microcontroladores. Igualmente, los sistemas de supervisión, vigilancia y alarma en los edificios utilizan estos dispositivos. También se emplean para optimizar el rendimiento de ascensores, calefacción, aire acondicionado, alarmas de incendio, robo, etc.

En los últimos años han tenido un gran auge los **microcontroladores PIC** fabricados por Microchip Technology Inc. Los PIC (Peripheral Interface Controller) son de bajo precio, reducido consumo energético, pequeño tamaño, gran calidad, fiabilidad y abundancia de documentación, que lo convierten en un dispositivo muy eficiente y fácil de usar.

II.3.2 Características de los microcontroladores PIC

1ª. La arquitectura del procesador sigue el modelo Harvard. En esta arquitectura, el CPU se conecta de forma independiente y con buses distintos con la memoria de instrucciones y con la de datos. Además, permite al CPU acceder simultáneamente a ambas, figura 2.13.

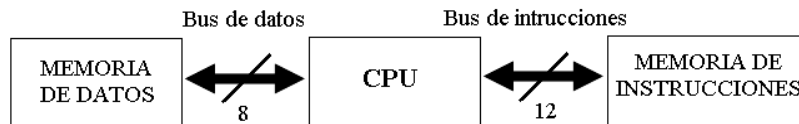


Figura 2.13 Arquitectura Harvard.

2ª. Se aplica la técnica de segmentación o Pipe-Line en la ejecución de las instrucciones. Un procesador segmentado realiza simultáneamente la ejecución de una instrucción y la búsqueda de código de la siguiente, de esta manera se puede ejecutar una instrucción en un ciclo. La figura 2.14 describe el funcionamiento donde cada instrucción se ejecuta en el denominado ciclo de instrucción, que está constituido por cuatro ciclos de reloj.

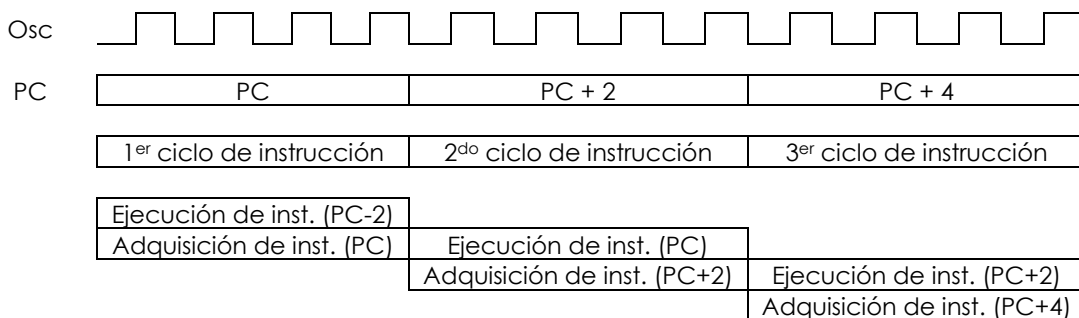


Figura 2.14 Ejemplo de funcionamiento de un procesador segmentado o Pipeline.

Este sistema, acompañado de una estructura Harvard, permite que los resultados de la ejecución de las instrucciones se obtengan a intervalos de un ciclo de reloj, salvo en el caso de saltos de programa que son dos. Lo que se hace internamente es que mientras se ejecuta la instrucción actual se carga la siguiente, alcanzando una alta velocidad de ejecución.

3º. El código máquina de todas las instrucciones tiene la misma longitud.

Todas las instrucciones de los microcontroladores de la gama baja tienen una longitud de 12 bits, los de la gama media tienen 14 y 16 bits los de la gama alta. Esta característica es de gran utilidad en la optimización de la memoria de instrucciones, y facilita enormemente la construcción de ensambladores y compiladores.

4º. Procesador RISC (Reduced Instruction Set Computer). Son microprocesadores con un conjunto de instrucciones reducido. Las instrucciones son muy simples y suelen ejecutarse en un ciclo de instrucción. Los procesadores RISC suelen tener una estructura Pipeline y procesar varias instrucciones al mismo tiempo. Los modelos de la gama baja disponen de un repertorio de 33 instrucciones, 35 los de la gama media y 77 los de la gama alta.

5º. Arquitectura ortogonal.

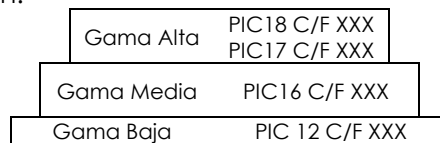
En los microcontroladores PIC la salida de la ALU va al registro W y también a la memoria de datos, así el resultado puede guardarse en cualquiera de los dos destinos. La gran ventaja de esta arquitectura es que permite un gran ahorro de instrucciones ya que el resultado de cualquier instrucción que opere con la memoria, puede dejarse en la misma posición de memoria o en el registro W.

6º. Herramientas de soporte potentes y económicas.

La empresa Microchip y otras que utilizan los PIC ponen a disposición de los usuarios numerosas herramientas para desarrollar hardware y software. Son muy abundantes los programadores, simuladores, emuladores en tiempo real, ensambladores, compiladores C, intérpretes, compiladores BASIC, etc.

La decisión más importante de cualquier proyecto es la elección del modelo de microcontrolador. Para adaptarse de forma óptima a las necesidades de los usuarios, Microchip ofrece tres gamas de microcontroladores de 8 bits. Con las tres gamas de PIC se dispone de gran diversidad de modelos y encapsulados, pudiendo seleccionar el que mejor se acople a las necesidades de acuerdo con la capacidad de memoria, el número de líneas de entrada/salida (E/S) y sus diversos módulos complementarios. Sin embargo, todas las versiones están construidas alrededor de una arquitectura común, un repertorio mínimo de instrucciones y un conjunto de opciones muy apreciadas, como el bajo consumo y el amplio margen del voltaje de alimentación.

Estas tres gamas principales son:



La familia de microcontroladores PIC de 8 bits (PIC 18Fxxx) reúne las características anteriores dentro de su gama y sale al mercado para situarse en un puesto relevante en los productos de elevadas exigencias. Esta familia mantiene su compatibilidad con las otras gamas, además de poder ser programados en forma muy eficiente con el lenguaje C, alcanzar rendimientos de 10 MIPS, disponer de un repertorio de 77 instrucciones de 16 bits, alcanzar un rango de direccionamiento de 20 Mbytes de memoria de programa, funcionar a una frecuencia de 48 MHz y ser programados en sistema (ISP) usando la interfaz serial SPI.

Una vez definidas las características de los microcontroladores PIC se ha optado por seleccionar el modelo que más convenga a la aplicación, el requerimiento que ha definido esta elección es la necesidad de contar con una comunicación USB hacia la computadora; por lo cual se optó por el modelo PIC 18F4550, ya que cuenta con un módulo interno que permite realizar la comunicación sin necesidad de dispositivos o componentes adicionales.

II.3.3 El microcontrolador PIC 18F 4550

Este μC posee todas las características mencionadas anteriormente. Es un μC de 8 bits con arquitectura Harvard, con procesador segmentado, procesador RISC y arquitectura ortogonal, trabaja a altas velocidades, gran cantidad de memoria de datos y gran eficiencia para ser programado con compiladores en lenguaje C.

Dentro de las características principales de este modelo, se tienen las siguientes:

- Alimentación con una tensión que va desde los 2 V hasta los 5.5 V.
- Bajo consumo de corriente ($<5\text{ mA}$) y la opción de pasar a modo Sleep ($<0.1\ \mu\text{A}$).
- Ofrece la opción de manejar un oscilador interno o externo (hasta 48MHz).
- 32 KBytes de memoria de programa FLASH.
- 256 Bytes de memoria de datos EEPROM.
- 2048 Bytes de memoria de datos SRAM.
- 5 Puertos de entrada/salida (Puerto A, B, C, D y E).
- 35 terminales de entrada/salida que consume/suministra hasta 25 mA cada terminal.
- Tres fuentes de interrupciones externas y diferentes interrupciones por software.
- Contador/temporizador (Timer), 1 de 8 bits y 2 de 16 bits.
- Convertidor analógico-digital de 13 canales y 10 bits de resolución.
- 2 comparadores de tensión.
- 2 módulos de Captura/Comparación/PWM (CCP).
- Puerto Serie Síncrono Maestro (MSSP), que incluye los modos: Interfaz Serie de Periféricos (SPI) e Interfaz Inter-Circuitos (I^2C).
- Transmisor-Receptor Universal Síncrono-Asíncrono (USART).
- Puerto Paralelo Maestro (SSP).
- Comunicación USB V2.0 a baja velocidad (1.5 Mb/s) y alta velocidad (12Mb/s).
- 100 000 ciclos de escritura/borrado de la memoria FLASH.
- 1 000 000 ciclos de escritura/borrado de la memoria EEPROM.
- El fabricante garantiza que el código grabado durará más de 40 años en este μC .

En la figura 2.15a se muestran de manera ilustrativa estos elementos dentro del microcontrolador, y en la figura 2.15 b y c se muestran los tipos más comunes de encapsulados.

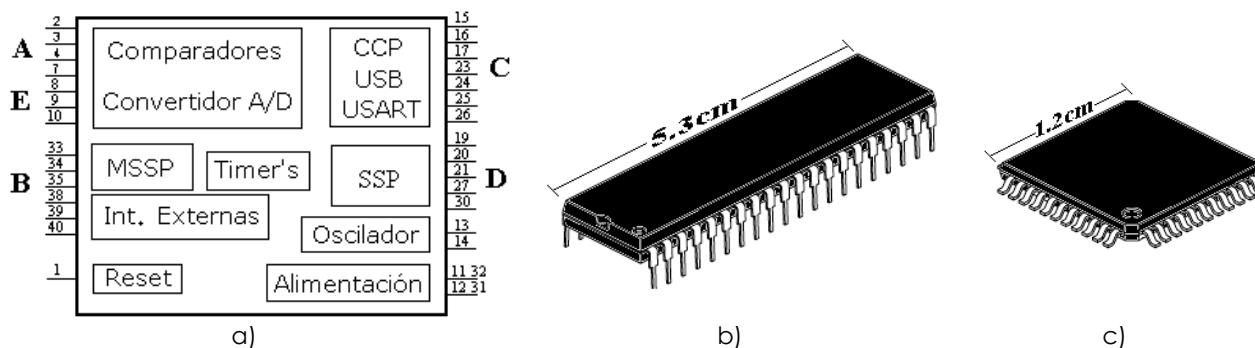


Figura 2.15 PIC 18F 4550: (a) Módulos dentro del μC , (b) Encapsulado DIP, (c) Encapsulado TQFP.

II.4 COMUNICACIÓN SERIE

La mejor manera de realizar una comunicación entre una Computadora Personal (PC, de las siglas en inglés Personal Computer) y un dispositivo electrónico es mediante la implementación de un protocolo de comunicación que se encuentren presentes en ambos. Un **Protocolo de Comunicación** es un conjunto de reglas estándar para la conexión y transferencia de información entre una computadora y un dispositivo. Este protocolo se implementa por software y hardware. Los procedimientos definidos por estos protocolos pueden ser de tipo síncrono o asíncrono.

Las comunicaciones que se pueden establecer entre varios dispositivos se pueden clasificar de acuerdo a muchos parámetros. Desde el punto de vista de la forma de transmitir los datos, se tienen a la Comunicación Serie (como RS-232, IEEE 1384 o USB) y la Comunicación en Paralelo (como GPIB, VXI o PXI). La transferencia de datos entre los dispositivos externos y la PC se realiza a través de una interfaz de entrada/salida denominada Puerto de Comunicaciones, el cual por simplicidad se le denominará puerto. Los sistemas serie, en comparación con los paralelos, tienen transmisión a mayor distancia, menor costo y son más sencillos en cuanto al hardware necesario. A menudo son comunicaciones punto a punto.

II.4.1. Especificaciones de la Recomendación Estándar RS-232

La Recomendación Estándar RS-232 (RS-232: Recommended Standard 232) está definida en las especificaciones del Instituto de Estándares Nacionales Americanos (ANSI: American National Standards Institute) como: «la interfaz entre un equipo terminal de datos (ETD) y un equipo de comunicación de datos (ECD) empleando una transmisión de datos binarios en modo serie». Esta es una de las recomendaciones estándares más viejas y comunes para la interfaz serie, la cual fue adoptada por la Asociación de Industrias Electrónicas (EIA: Electronics Industries Association) y ha sufrido seis modificaciones, la más reciente es la «F» hecha en 1997.

II.4.1.1 Flujo de datos

Un dato importante a tener en cuenta en cualquier comunicación es la velocidad de transmisión, que es la cantidad de información enviada por la línea de transmisión por la unidad de tiempo. Hay distintas unidades para expresar esta medida, la más utilizada es el **Baud**, que es proporcional a los Símbolos/segundo (Bps), definidos como el número de símbolos de información enviados en un segundo. La velocidad a la que pueden trabajar los puertos COM de una PC está normalizada a 75, 150, 300, 600, 1200, 2400, 4800, 9600 Bauds, etc.

Los bits son enviados con una frecuencia predefinida, el **Baud Rate**. Ambos, el transmisor y el receptor, deben ser programados para usar la misma frecuencia de transferencia de los bits. Después de recibir el primer bit, el receptor calcula en qué momento el siguiente bit de datos debe ser recibido. Éste revisará el valor de tensión de la línea de información en ese momento.

Para poder ser enviada la información de manera serial, ésta debe de ser separada en palabras de información. En la figura 2.17 se muestra la representación de un paquete de información de la Recomendación Estándar RS-232. La longitud de esta palabra es variable, en una PC puede ser seleccionada entre cinco y nueve bits. Para una transmisión apropiada son agregados ciertos bits con propósitos de sincronización y de detección de errores, estos son: el bit de inicio, el bit de paro y el bit de paridad. Es importante que tanto el receptor como el transmisor sean configurados con la misma cantidad de bits, de otra manera el mensaje no podrá ser interpretado de la manera correcta o incluso no ser reconocido.

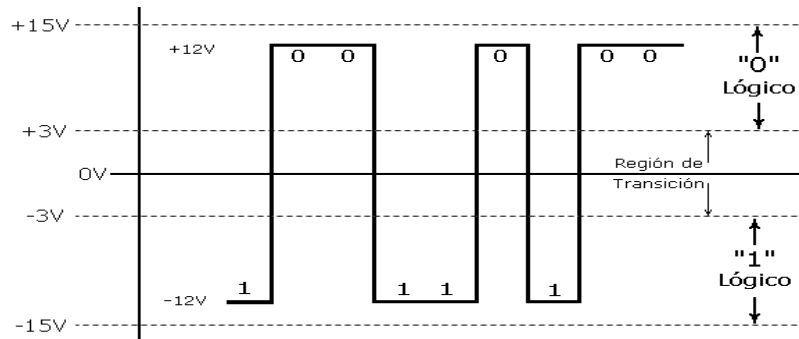


Figura 2.18 Valores de tensión del estándar RS-232.

Para garantizar un '0' lógico, una línea debe mantener una señal de voltaje entre +3 y +15 V. Del mismo modo para '1' lógico garantizado debe estar entre -3 y -15 V. Los voltajes usados en las computadoras son +12V para el '0' y -12 V para el '1'. Es importante resaltar que la banda muerta entre +3 y -3 V se conoce como la **región de transición**, donde los niveles lógicos no están definidos. Esto significa que cualquier valor entre +3 y -3 V puede interpretarse ambiguamente como '0' ó '1' lógico.

Estos valores de tensión proporcionan un amplio margen de seguridad, que es de gran utilidad cuando los cables deben pasar por zonas cercanas a elementos que generan interferencias electromagnéticas: motores, transformadores, equipos de comunicación, etc. Estos elementos, unidos a la longitud del cable, hacen disminuir la señal hasta en varios Volts, pero sin que afecte adversamente los niveles de voltaje. Si se aumenta la velocidad de transmisión, las señales de datos se vuelven más susceptibles a pérdidas de voltaje, causadas por la capacitancia, resistencia e inductancia del cable, estas pérdidas son conocidas como efectos de alta frecuencia y aumentan con la longitud del mismo.

II.4.1.3 Cables y conectores

Según la Recomendación Estándar RS-232, la distancia máxima del cable de conexión utilizado en la comunicación es aquella con una capacitancia menor o igual a 2500 pF. Utilizando un cable estándar, esta capacitancia se alcanza entre los 15 y 20 metros, pero si, por ejemplo, se utiliza un cable UTP CAT-5 con una capacitancia de 52 pF/m, esta distancia se incrementa considerablemente.

El conector originalmente definido para este estándar es el conector DB-25, con 25 terminales para la conexión serial. Debido al gran tamaño de este conector, en los equipos de cómputo que aún poseen este tipo de comunicación se suele encontrar el conector DB-9, el cual realiza una conexión con sólo nueve terminales y elimina los terminales del conector DB-25 que realizan funciones raramente usadas. En la figura 2.19 se muestra la distribución de las terminales de los conectores DB-9 y DB-25, y en la tabla 2.6 se muestra el nombre y dirección de dichas terminales.

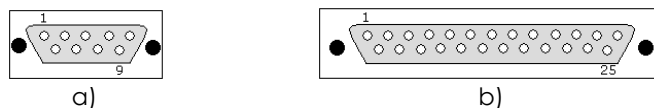


Figura 2.19 Conectores hembras para un ETD utilizados por la recomendación estándar RS-232: (a) DB-9, (b) DB-25.

Una comunicación RS-232 se puede realizar con sólo tres terminales, la de recepción de datos (RxD), la de transmisión de datos (TxD) y la referencia o común de las señales (GND), realizando una comunicación en modo full-duplex.



Tabla 2.6 Descripción de las señales del estándar RS-232.

Nombre del terminal	Conector DB-25	Conector DB-9	Dirección De la señal	Función
TxD	2	3	Salida	Datos Transmitidos
RxD	3	2	Entrada	Datos Recibidos
Gnd	7	5	---	Común o Tierra
RTS	4	7	Salida	Control de flujo en la transmisión de datos
CTS	5	8	Entrada	Permite realizar la transmisión de datos
DTR	20	4	Salida	Detector de línea recibida
DSR	6	6	Entrada	Equipo de datos preparado
RI	22	9	Entrada	Timbre indicador
DCD	8	1	Entrada	Detector de línea recibida

II.4.2. Especificaciones del estándar USB

El Bus Serie Universal (USB: Universal Serial Bus) es un puerto que sirve para conectar periféricos a una computadora. Fue creado en 1996 por siete empresas: IBM, Intel, Northern Telecom, Compaq, Microsoft, Digital Equipment Corporation y NEC.

Dentro de las características de este estándar están: el permitir a los dispositivos una conexión con la PC sin la necesidad de que ésta deba ser reiniciada, cuando se conecta un nuevo dispositivo, ésta lo enumera y agrega el software necesario para que pueda funcionar, y da la posibilidad de alimentar a los dispositivos conectados a través del mismo bus.

Los dispositivos USB se clasifican en cuanto a su velocidad de transferencia de datos (tabla 2.7):

Tabla 2.7 Velocidades de transferencias USB.

Estándar	Velocidad	Publicación	Bits / Seg	Bytes / Seg
USB 0.9	Primer borrador	1995		
USB 1.0	Baja velocidad	1996	1.5 M	192 K
USB 1.1	Velocidad completa	1998	12 M	1.5 M
USB 2.0	Alta velocidad	2000	480 M	60 M
USB 3.0	Súper velocidad	2008	4.8 G	600 M

Los tres componentes principales del estándar USB son: el **Anfitrión** (*Host*), los **Concentradores** (*Hubs*) y los **Dispositivos** (*Devices*). Los *dispositivos* son el punto final del sistema como ratones, teclados, impresoras, etc. Múltiples dispositivos pueden conectarse a un solo puerto USB a través de un hub. El dispositivo anfitrión es el centro de la comunicación USB, como una PC. El protocolo USB define las características necesarias para la interconexión de los dispositivos con el anfitrión USB y esto incluye los conectores, los cables, la topología y las capas de comunicación.

II.4.2.1 Flujo de datos

El estándar USB es síncrono y emplea para la transmisión de sus paquetes de información el algoritmo de codificación No Retorno a Cero Invertido (NRZI: Non Return to Zero Inverted). La codificación NRZI representa un '1' lógico manteniendo el voltaje a nivel de la línea, pero no hay retorno a cero entre bits, de forma que una serie de unos corresponde a un voltaje uniforme. En cambio, los '0' lógicos se marcan como cambios del nivel de tensión, de modo que una sucesión de ceros produce sucesivos cambios de tensión entre los conductores de señal.

II.4.2.2 Topología

La topología se basa en un modelo de conexión estrella escalonada, con el anfitrión en la punta de la pirámide y los hubs en el centro de cada conexión con los dispositivos, como se muestra en la figura 2.20.

El **anfitrión** se encarga de detectar cuando un dispositivo es conectado y desconectado, también maneja el control de flujo de datos y suministra energía a los dispositivos que así lo soliciten. Sólo puede existir un anfitrión en una red USB.

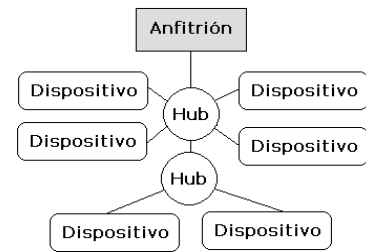


Figura 2.20 Topología estrella USB.

Por otra parte, los **dispositivos** pueden ser descritos de acuerdo a la *clase* a la cual pertenecen. Una **clase** es un grupo de dispositivos que tienen similar funcionalidad e implementación, por ejemplo: monitores, dispositivos de audio, de almacenamiento masivo, impresoras y dispositivos de interfaz humana (HID [Human Interfaz Devices], como ratones, teclados, etc.). Esta separación de clases permite el desarrollo de un controlador (*driver*) genérico para toda una clase de dispositivos y que el usuario no necesite instalar uno específico para cada dispositivo. Si un dispositivo tiene más funciones que las presentes en éste controlador genérico, se instalará uno específico para que pueda funcionar.

Todos los dispositivos USB tienen un conjunto de descriptores que incluye el nombre del proveedor del dispositivo, el identificador del protocolo y el número de configuraciones que posee el dispositivo. De esta manera el anfitrión conoce cuantos descriptores de configuración debe pedir. Los descriptores de configuración incluyen el número y el tipo de interfaz o funciones que tiene el dispositivo. Después de recibir todos los descriptores de configuración, el anfitrión pide los de interfaz. El descriptor de interfaz le dice al anfitrión el tipo y clase de características que tiene este dispositivo.

II.4.2.3 Protocolo de comunicación

El protocolo de comunicación usado es de intercambio de paquetes (**Tokens**). Puesto que todos los periféricos comparten el bus y pueden funcionar de forma simultánea, la información es *enviada en paquetes*. Cada paquete contiene una cabecera que indica el dispositivo al que va dirigido, como muestra la figura 2.21a. Existen cuatro tipos de paquetes distintos: **Token**, **Datos**, **Handshake** (ACK), y **Especial**; el máximo número de datos por paquete es de 8, 16, 32 y 64 Bytes. Se usa un sistema de detección y corrección de errores bastante robusto tipo CRC (Cyclical Redundancy Check).

El funcionamiento está centrado en el anfitrión, todas las transacciones se originan en él. Una **transacción** es un grupo de paquetes que realiza una comunicación y éstas son siempre iniciadas por el anfitrión. Cuando se detecta un nuevo dispositivo, es el anfitrión el encargado de cargar los drivers necesarios para su comunicación, también envía un paquete Token al dispositivo notificándole el número que le ha sido asignado. Cada transacción utiliza un mínimo de tres paquetes (ver parte superior derecha de la figura 2.21a). El primero es siempre un Token que avisa al dispositivo que puede iniciar la transmisión, o que el anfitrión desea realizar una acción sobre éste. De esta forma se puede entender a éste protocolo como un método de encuesta que utiliza el anfitrión, es decir, el anfitrión interroga al dispositivo y éste le responde. Por ejemplo, en la figura 2.21b, si una petición de información es enviada con el paquete Token, el dispositivo responde a ésta petición con un paquete de datos. De la misma manera el paquete Token puede indicar al dispositivo que espere por información. Si éste es el caso, un paquete de datos será enviado por el anfitrión, al que el dispositivo responderá con un mensaje de confirmación, ya sea que la operación haya tenido éxito o no.

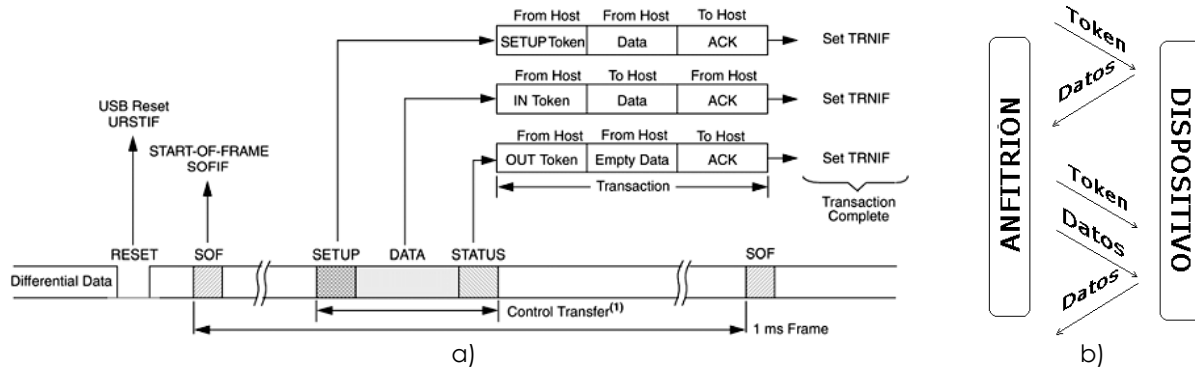


Figura 2.21 (a) Diagrama de tiempo de la comunicación USB, (b) Ejemplo de comunicación USB.

El protocolo USB 1.X dividía el tiempo en tramas de 1 mSeg, USB 2.0 define un tiempo de trama de 125 μ Seg, y USB 3.0 define un tiempo de trama de 1.5 μ Seg.

II.4.2.4 Cables

Los cables consisten en dos líneas para la alimentación y dos líneas para la transmisión de la información. Las señales del USB son transmitidas en un cable de par trenzado cuyos pares son llamados D+ y D-, como muestra la figura 2.22. Éstos colectivamente utilizan señalización diferencial en Half Dúplex, para combatir los efectos del ruido electromagnético en enlaces largos. Para transmitir en alta velocidad, el transmisor activa una fuente de corriente interna, derivada a partir de su fuente de alimentación positiva, y dirige dicha corriente hacia la línea D+ por medio de un conmutador de corriente de alta velocidad, y después hacia la línea D-, de esta manera el transmisor genera dos estados en el cable.

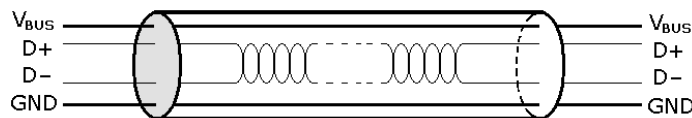


Figura 2.22 Cable de conexión USB.

Esta conmutación dinámica de corriente sobre ambas líneas de datos D+ y D- sigue las mismas reglas de codificación NRZI y de inserción de bit (*Bit-Stuffing*), ya utilizadas en alta y baja velocidad. El estado J se obtiene dirigiendo la corriente sobre la línea D+, mientras que el estado K se obtiene dirigiendo la corriente sobre la línea D-. En USB 2.0, tanto el transmisor como el receptor activan unas resistencias de terminación entre cada línea y la referencia ($45 \Omega \pm 10\%$), de forma que el valor nominal de la corriente (17,78 mA) produce un voltaje nominal en la línea de 400 mV. El voltaje diferencial (D+, D-) es de +400 mV para el estado J y de -400 mV para el estado K.

Es necesario definir correctamente el diámetro del hilo con el fin de que no se produzca una caída de tensión demasiado importante en el cable. Existen dos tipos de cable: blindado y sin blindar. En el primer caso el par de hilos de señal es trenzado; los de tierra y alimentación son rectos, y la cubierta de protección (blindaje) solo puede conectarse a tierra en el anfitrión. En el cable sin blindar todos los hilos son rectos.

Las conexiones a 15 Mbps y superiores exigen cable blindado. El estándar permite un largo máximo de 5 metros.

A nivel de alimentación, el cable proporciona una tensión nominal de 5 V (± 0.25 V) regulados por cada uno de sus puertos. Por lo tanto, dispositivos de bajo consumo de potencia puede obtener de ahí la corriente necesaria para su funcionamiento. Cada Hub puede conectar hasta 7 dispositivos, aunque lo normal es que sean de 4, y proporcionar 500 mA de alimentación (hasta 2.5 W) a cada uno de ellos.

Cuando un dispositivo es conectado le reporta al anfitrión cuanta potencia va a consumir. De esta manera el anfitrión lleva un registro de los requisitos de cada puerto y cuando un dispositivo se excede generalmente se apaga, cortando el suministro de corriente, de forma de no afectar al resto de los dispositivos.

II.4.2.5 Conectores

Existen dos tipos de conectores: **Estándar** y **Mini**, mostrados en la tabla 2.8. Los estándar vienen en dos subtipos: A y B. El subtipo A se encuentra del lado del anfitrión, mientras que el subtipo B se encuentra del lado del dispositivo. Todos los cables son machos, mientras que los conectores (ya sea en la PC o los dispositivos) son hembra. No existen intercambiadores de género puesto que no se puede conectar dispositivo con dispositivo y anfitrión con anfitrión. Los conectores mini siguen la misma política que los estándar, pero son utilizados para dispositivos pequeños como Palm y celulares.

Tabla 2.8 Conectores USB y descripción de terminales.









	Conector Macho (Cable)	Conector Hembra (Anfitrión)	Apariencia	Pin	Nombre
Tipo A				1	V _{CC}
Tipo B				2	Data -
Mini				3	Data +
				4	GND

Tabla 2.8 Conectores USB y descripción de terminales.

El estándar USB, a diferencia de otros estándares, también define tamaños para el área alrededor del conector de un dispositivo, lo que evita el bloqueo entre puertos adyacentes.

II.4.3. Migración de RS232 a USB

La interfaz serie RS-232 está desapareciendo prácticamente de las computadoras personales y esto supone un problema, ya que muchas de las aplicaciones con microcontroladores utilizan este bus para la comunicación con la PC. La solución ideal es emigrar a una interfaz USB y existen distintas formas de hacerlo. El método más sencillo es emular RS-232 con el USB, con la ventaja de que la PC verá la conexión USB como una conexión COM RS-232 y no requerirá de cambios en el software existente. Otra ventaja es que se usan bibliotecas (*drivers*) suministradas por el Sistema Operativo (por ejemplo *Windows*), por lo que no es necesario desarrollar uno propio, así la PC puede identificar automáticamente al dispositivo conectado. Además, puesto que el protocolo USB maneja comunicaciones de bajo nivel, los conceptos de baud rate, bit de paridad y control de flujo para el RS-232 ya no importan.



II.5 HERRAMIENTAS DE APOYO

En este trabajo se usaron tres programas de desarrollo como herramientas de apoyo, los cuales permitieron el desarrollo del sistema. A continuación se describirá cada uno de ellos de manera general, para mayor información se recomienda referirse al manual o la ayuda de cada programa.

II.5.1 Entorno de programación en el compilador C

El uso de los lenguajes más cercanos a la máquina (de bajo nivel) representan un considerable ahorro de código en la síntesis de los programas, esto es muy importante dada la estricta limitación de la capacidad de la memoria de instrucciones. Los programas bien realizados en lenguaje Ensamblador optimizan el tamaño de la memoria que ocupan y su ejecución es muy rápida, en cambio para aplicaciones con mayor complejidad este lenguaje ya no es tan eficiente, ya que se vuelve demasiado extenso y complicado, para evitar esto se recomienda el uso de lenguajes de alto nivel. Los lenguajes de alto nivel más empleados con microcontroladores son el C y el BASIC, de los que existen varias empresas que comercializan versiones de compiladores e intérpretes.

Entre los muchos programas para el desarrollo de sistemas con microcontroladores PIC destacan, el compilador C de Custom Computer Services Incorporated (CCS). El compilador C de CCS permite sintetizar código para PIC programando en lenguaje C con las ventajas que supone tener un lenguaje desarrollado específicamente para un microcontrolador específico. Su facilidad de uso, su amplia variedad de bibliotecas y la posibilidad de compilar en las tres gamas: baja, media y alta, proporcionan una versatilidad y potencia muy elevadas, por estas razones nos centraremos en la descripción de dicho compilador.

El Compilador C de CCS ha sido desarrollado específicamente para PIC MCU, obteniendo la máxima optimización del compilador con estos dispositivos. Dispone de una amplias bibliotecas de funciones predefinidas, comandos de preprocesado y ejemplos. Además, suministra los controladores (*drivers*) para diversos dispositivos como LCD, convertidores analógicos a digital, relojes en tiempo real, memorias EEPROM serie, etc. Las características generales de este compilador y más información adicional se pueden encontrar en la dirección <http://www.ccsinfo.com>

II.5.1.1 Entorno de trabajo

El CCS es C estándar y, además de las directivas estándar (*#include, etc*), suministra unas directivas específicas para PIC (*#device, etc*), además incluye funciones específicas (*bit_set(), etc*). Se suministra con un editor que permite controlar la sintaxis del programa.

CCS suministra tres compiladores básicos, el PCB, el PCM y el PCH. PCB se utiliza para PIC de 12 bits, el PCM para 14 bits y el PCH para 16 bits.

El entorno de trabajo del CCS en PCW y PCWH permite ensamblar y también suministra una gran variedad de herramientas auxiliares. En la figura 2.23 se muestra los distintos elementos básicos del entorno de trabajo.

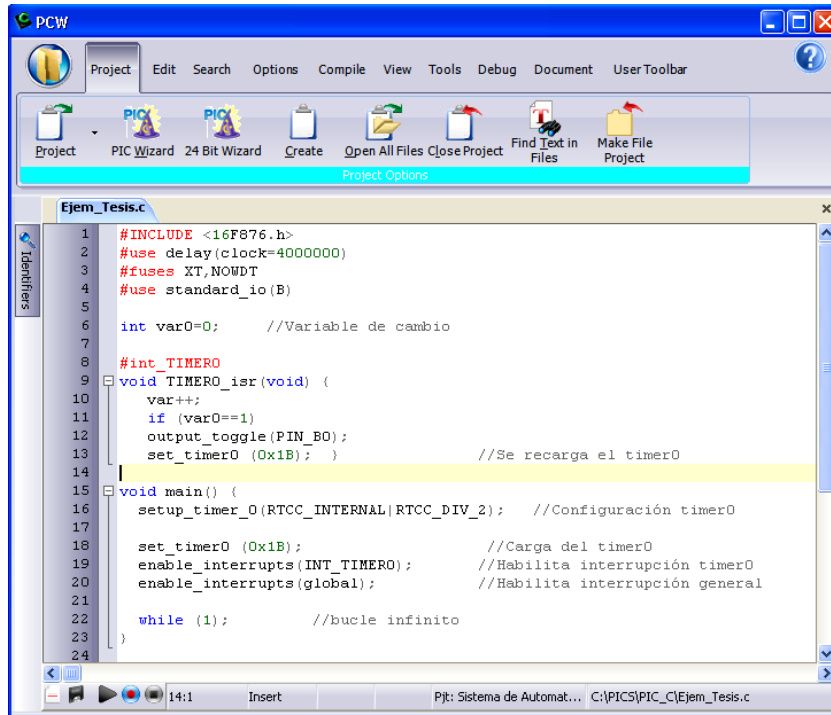


Figura 2.23 Entorno de trabajo.

II.5.1.2 Estructura de un programa

Para escribir un programa en C con el CCS C se debe tener en cuenta una serie de elementos básicos de su estructura.

- **DIRECTIVAS DE PREPROCESADO:** controlan la conversión del programa a código máquina por parte del compilador.
- **PROGRAMAS o FUNCIONES:** conjunto de instrucciones. Puede haber uno o varios, en cualquier caso siempre debe haber uno definido como principal mediante la inclusión de la directiva *main()*.
- **INSTRUCCIONES:** indican cómo se debe comportar el PIC en todo momento.
- **COMENTARIOS:** permiten documentar lo que significa cada línea del programa.

II.5.2 Programación gráfica en LabVIEW

LabVIEW es el acrónimo de *Laboratory Virtual Instrument Engineering Workbench*. Es un lenguaje y a la vez un ambiente de programación gráfica en el que se pueden crear aplicaciones visuales de una forma rápida y sencilla. National Instruments es la empresa desarrolladora y propietaria de LabVIEW, comenzó en 1976 en Austin, Texas. El lenguaje que usa se llama lenguaje G, donde la G simboliza que es lenguaje Gráfico.

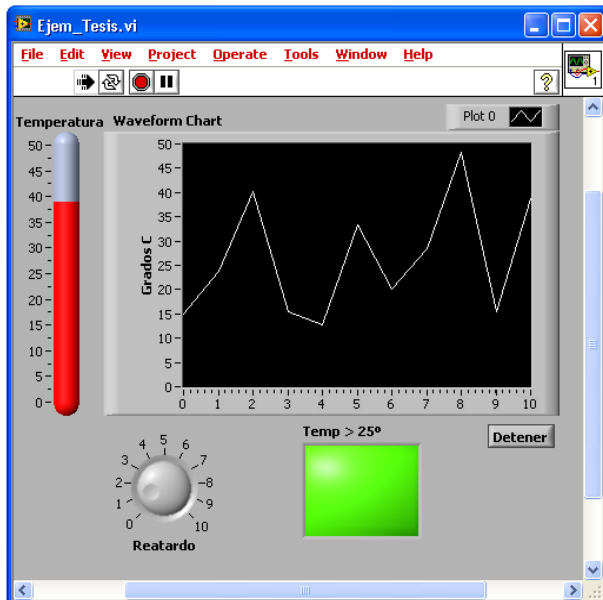
II.5.2.1 Ambiente de trabajo

Originalmente este programa estaba orientado a aplicaciones de control de instrumentos electrónicos usadas en el desarrollo de sistemas de instrumentación, lo que se conoce como instrumentación virtual. Por este motivo los programas creados en LabVIEW se guardan en ficheros llamados VI y con la misma extensión, que significa instrumento virtual (*Virtual Instrument*). También relacionado con este concepto se da el nombre a sus dos ventanas principales: un instrumento real tendrá un Panel Frontal donde estarán sus botones, pantallas, etc. y una circuitería interna. En LabVIEW estas partes reciben el nombre de Panel Frontal y Diagrama de Bloques respectivamente.

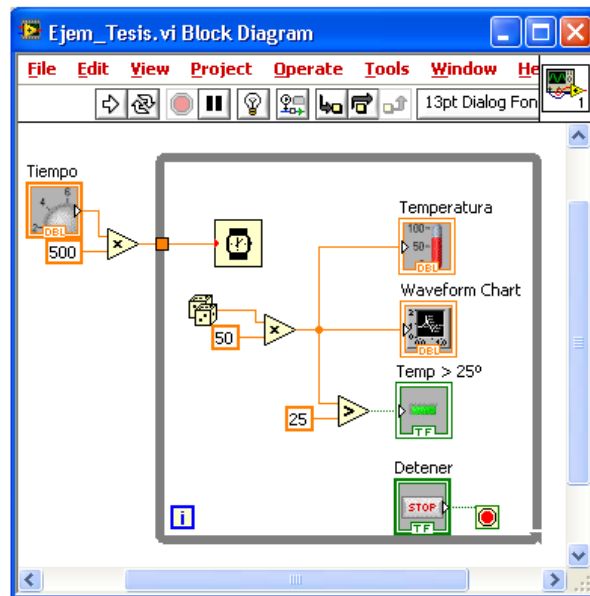
II.5.2.2 Componentes del programa

El panel frontal (Figura 2.24a) se construye a base de controles e indicadores, los cuales no son más que las terminales de entrada y salida, respectivamente, del VI. Como controles se puede tener *Knobs* y *dials* (botones rotatorios), *push buttons* (pulsadores) y otros dispositivos de entrada. Como indicadores se tienen gráficas, leds y otros visualizadores. Los controles simulan elementos de entrada al instrumento y proporcionan datos de bloques. Los indicadores simulan elementos de salida del instrumento y visualizan los datos que el diagrama de bloques adquiere o genera.

Una vez construido el panel frontal, se desarrolla el código fuente usando unas representaciones gráficas de funciones que controlarán los objetos de éste. Estos objetos aparecen como terminales en el diagrama de bloques (figura 2.24b), los cuales se conectan a bloques funcionales con entradas y salidas mediante cables ficticios por donde fluyen los datos.



a)



b)

Figura 2.24 Ejemplo de un VI: (a) Panel Frontal, (b) Diagrama a Bloques.

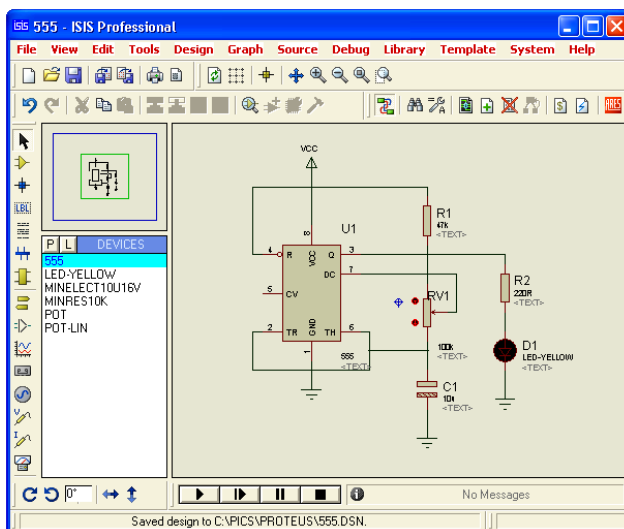
II.5.3 Diseño electrónico en PROTEUS VSM

Siempre que se diseña con circuitos integrados programables se precisan herramientas para la puesta a punto del hardware y del software. Con referencia al software, además de los compiladores o intérpretes de los lenguajes usados, es muy interesante disponer de simuladores software, que consisten en programas que simulan la ejecución de instrucciones representando el comportamiento interno del procesador y el estado de las líneas de entrada/salida (E/S). Como se simula al procesador por software, el comportamiento no es en tiempo real, ya que depende de las características físicas de la computadora (memoria RAM, etc.), aunque proporciona una aproximación aceptable, especialmente cuando no es esencial el trabajo en tiempo real.

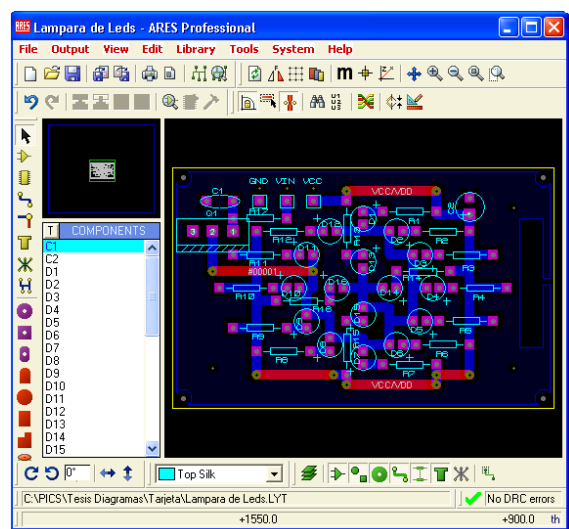
El programa PROTEUS VSM es una herramienta para la verificación vía software que permite comprobar, prácticamente en cualquier diseño, la eficacia del programa desarrollado. Su combinación de simulación de código de programación y simulación mixta SPICE permite verificaciones analógico-digitales de sistemas basados en microcontroladores.

El entorno de diseño electrónico PROTEUS VSM de LABCENTER ELECTRONICS, ofrece la posibilidad de simular el código de un microcontrolador de alto y bajo nivel. Esto permite el diseño tanto a nivel hardware como de software y realizar la simulación en un mismo y único entorno. Para ello, se suministran de tres potentes subentornos como son el ISIS para el diseño gráfico, VSM (*Virtual System Modelling*) para la simulación y el ARES para el diseño de los circuitos impresos.

ISIS es un potente programa de diseño electrónico que permite realizar esquemas que pueden ser simulados en el entorno VSM (Figura 2.25a) y pasados a un circuito impreso ya en el entorno ARES (Figura 2.25b). Posee una extensa colección de bibliotecas de modelos tanto para dibujar, simular o para las placas. Además, permite la creación de nuevos componentes, el modelo para su simulación e, incluso, la posibilidad de solicitar al fabricante (*Labcenter Electronics*) que cree un nuevo modelo.



a)



b)

Figura 2.25 PROTEUS (a) ISIS, diseño y simulación, (b) ARES, creación de PCB's.



Capítulo

3

Desarrollo del Hardware

Objetivo particular. Este capítulo se centra en los elementos físicos que componen al sistema, es decir, en el hardware. Únicamente se mostrará el desarrollo del mismo y contendrá los diagramas electrónicos de los diferentes módulos o etapas que lo componen. Se presentará cada uno de los componentes que fueron seleccionados y sus configuraciones.

III.1. DESCRIPCIÓN DEL HARDWARE

Como se estableció en el primer capítulo, la propuesta de solución (subtema 1.4.2) para el hardware consiste en la selección de los motores, el diseño y desarrollo de la tarjeta electrónica para el sistema de control, el medio de comunicación con la computadora y el sistema de iluminación, para dar solución a este punto se propone un sistema de control que pueda funcionar en dos modos:

Manual (modo por *default*): el control de los motores es mediante 3 botones, de los cuales uno funciona para girar libremente el motor en un sentido, otro para girarlo en sentido contrario y el tercero para seleccionar el número del motor a controlar.

Automatizado con computadora: en este modo se requiere de la conexión con la computadora, la cual, mediante un programa controlará la posición de los motores, esta posición podrá ser manipulada introduciendo el valor numérico de la posición a la que se desea mover el motor. En dicho programa también se podrá variar la intensidad lumínica del sistema de iluminación.

El diagrama de bloques del hardware se muestra en la figura 3.1.

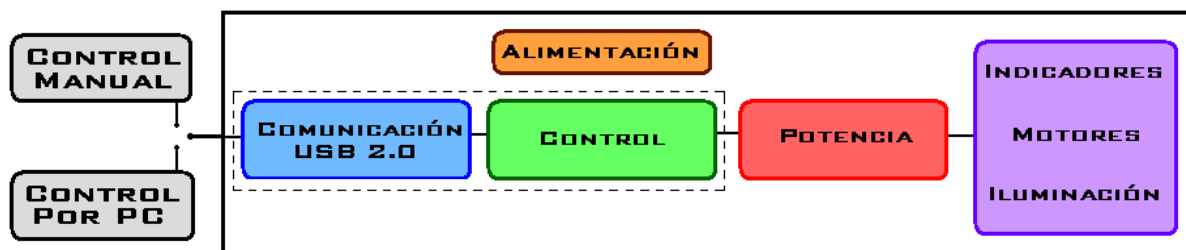


Figura 3.1 Diagrama a bloques funcional del hardware.



A continuación se describirá brevemente cada una de las etapas.

Fuente de Alimentación: esta etapa es de tipo convertidor CD-CD.

Comunicación: esta etapa es para poder realizar una conexión USB entre la computadora y el hardware, y así poder establecer el control.

Control: es la etapa más importante y la encargada de coordinar e integrar todas las funciones, esta etapa se basa en un microcontrolador de 8 bits que genera las señales para el control de los motores.

Potencia: es la etapa que dirige a los motores y es la conexión entre la parte lógica del sistema y los motores.

Iluminación: para esta etapa se construirá una pequeña lámpara de LEDs, la cual está constituida por 16 LEDs blancos superbrillantes y puede variar su intensidad de iluminación a través de la Computadora.

Indicadores y botones: se trata de botones y LEDs que sirven para el funcionamiento del sistema.

III.1.1 Selección de los motores

Para lograr el movimiento de los ejes del Equipo Mecánico-Óptico se debe seleccionar el tipo de motor adecuado que genere un torque mínimo de 0.2 N·m, dado que ya tiene implementado el sistema de engranes que necesita de por lo menos este torque; para cubrir ésta necesidad se eligieron los **motores de pasos** por tener esa o mayor capacidad de torque en cada uno de sus desplazamientos, también, estos se pueden adquirir por grados de movimiento angular entre cada paso, además, permiten realizar pequeños movimientos y mantener su posición. El tipo de motor seleccionado es: **motor de pasos de imán permanente unipolar de dos fases** (en el subtema III.4 se describirá a detalle los motivos que llevaron a ésta elección), una de las características por la que fue seleccionado este tipo de motor, es, que en determinado momento este mismo motor *unipolar*, se puede manejar como *bipolar*, lo que permite la comparación en cuanto al funcionamiento de ambos tipos para determinar el más óptimo.

Una vez establecido el tipo de motor, se han adquirido del fabricante SANYO con número de parte 103H5210-0440 de las siguientes características técnicas: 5 V_{DC}, 1.2 A/fase, 3.3 Ω/fase, 3.4 mH/fase, 1.8°/paso y 0.37 N·m. Se eligieron estos porque se adaptan al torque y grados que se necesitan para la aplicación, además el costo de adquisición es accesible.

III.2. FUENTE DE ALIMENTACIÓN

Como se vio en el capítulo 2, una fuente de alimentación es el circuito que convierte un voltaje de corriente alterna a su entrada en un voltaje de corriente directa en su salida. Para nuestro caso, dado que los motores de pasos consumen una considerable cantidad de corriente (1.2 A/fase), es necesaria la conexión permanente a una toma de voltaje de corriente alterna. También es importante notar que los voltajes de operación aunque son iguales tanto para el sistema de control como para los motores, se han dividido las tomas para evitar que el ruido electromagnético generado por las bobinas de los motores afecte al sistema de control. Dadas estas características se ha diseñado una fuente con tres voltajes de salida, 5V para alimentar a los motores, 5V para alimentar la parte sistema de control, y 12 V para alimentar a un ventilador. En la figura 3.2 podemos observar el diagrama de la fuente de alimentación.

La toma de $V_{CC} = 5\text{ V}$ se basa en un regulador conmutado en configuración reductor, es el L4960H del fabricante STMicroelectronic. Este regulador fue seleccionado con base en sus óptimas características: requiere de pocos componentes externos, amplio rango de voltaje de entrada (de 8 a $46V_{CD}$), voltaje de salida ajustable (de 5 a $40V_{CD}$), corriente de salida superior a la necesaria para alimentar a los motores (2.5A) y alta eficiencia (90%).

Dentro de sus componentes externos, tal vez en el que se deba tener más cuidado es en la inductancia, ya que durante la práctica se pudo observar que el voltaje de salida, depende en parte de la caída de tensión que tenga ésta, en un principio se había diseñado la bobina manualmente con la inductancia que marcaba el fabricante, sin embargo, el valor de la resistencia era muy grande, por consiguiente la caída de voltaje en sus terminales era mayor y no proporcionaba el voltaje adecuado. Para solucionar el problema se optó por conseguirla comercialmente. Las resistencias R5 y R6 son las encargadas de fijar al voltaje de salida.

Para las tomas restantes de $V_{DD} = 5\text{V}$ y 12V se utilizaron los ya conocidos reguladores lineales de la familia L78XX del fabricante STMicroelectronic, su selección se basó en la precisión y facilidad de uso de estos CIs, no se necesita de componentes externos más que un par de capacitores ayudando con esto a reducir el tamaño de la fuente, también se tomó en cuenta que su capacidad de corriente (1.5 A) es mayor a la necesaria para el funcionamiento de las etapas a las que van a alimentar, V_{DD} alimenta al sistema de control que requiere de aproximadamente 0.4 A , y 12V alimentará a un ventilador de 0.28A .

Con esta fuente se tiene: voltaje de alimentación de $127 \pm 10\%$, voltajes de salida de 5V a 2.5A para los motores, 5V a 1.5A para el sistema de control y 12V a 1.5A para el ventilador.

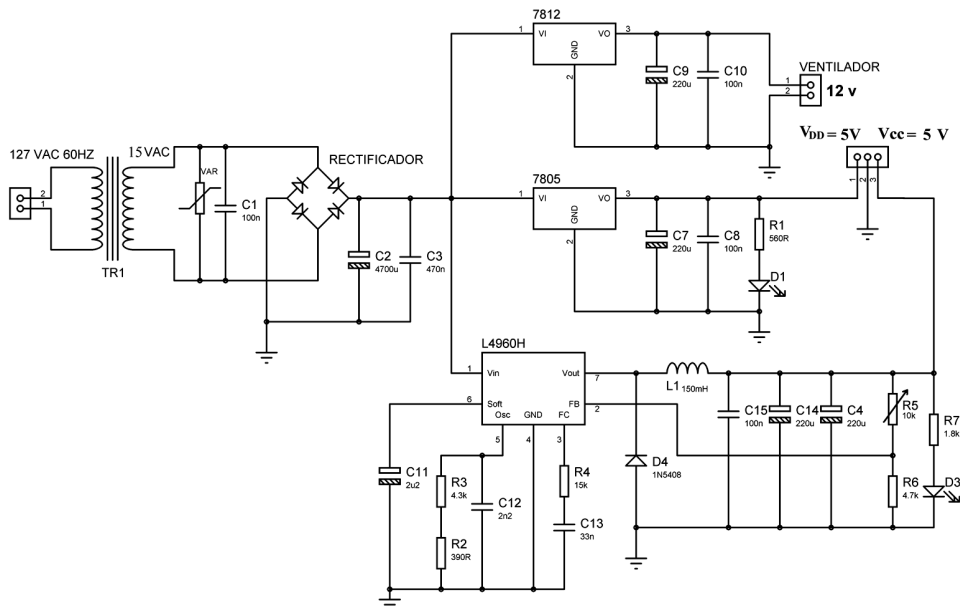


Figura 3.2 Diagrama de la fuente de alimentación.

III.3. CONTROL

Como ya se ha mencionado, esta etapa está basada en un microcontrolador de 8 bits, de la familia 18F, el PIC 18F 4550 del fabricante Microchip Technology Inc. Este μC es el encargado de realizar la comunicación USB y generar las señales para controlar a los motores. El código de este μC se explicará en el capítulo siguiente.



III.3.1 Conexiones del microcontrolador

El μC se comunica con el mundo exterior a través de los Puertos. Estos están constituidos por señales digitales de entrada/salida que trabajan entre 0 y 5 V (lógica TTL). Cada línea puede ser configurada como entrada o como salida, independientemente unas de otras, según se programe.

- El Puerto A con 6 líneas, pines RA0 a RA5.
- El Puerto B con 8 líneas, pines RB0 a RB7.
- El Puerto C con 7 líneas, pines RC0 a RC2 y de RC4 a RC7.
- El Puerto D con 8 líneas, pines RD0 a RD7.
- El Puerto E con 3 líneas, pines RE0 a RE2.

La máxima capacidad de corriente de cada línea es de 25 mA cuando consume corriente (modo sink), y soporta 25 mA cuando proporciona corriente (modo source). Sin embargo, la suma de las corrientes de todos los puertos no debe exceder a 200 mA, ya sea en modo sink o modo source.

III.3.1.1 Alimentación

Para que el μC pueda operar sólo requiere de ser alimentado, normalmente con una tensión de 5V aplicados entre los pines V_{DD} (Voltaje, pin 32 y 11) y V_{SS} (Referencia, pines 31 y 12). El consumo de corriente para el funcionamiento del microcontrolador depende de la tensión de alimentación, de la frecuencia de trabajo y de las cargas que soporten sus entradas/salidas, siendo del orden de unos pocos miliamperes.

El circuito de alimentación del microcontrolador debe tratarse como un dispositivo digital, debiendo conectarse un capacitor de desacoplo de 100 nF lo más cerca posible de los pines de alimentación. En la figura 3.3a se muestra como se conectó la alimentación a este μC .

III.3.1.2 Oscilador

El oscilador o señal de reloj genera una onda cuadrada de alta frecuencia que se utiliza como señal para sincronizar todas las operaciones del sistema. Este circuito es muy simple pero de vital importancia para el funcionamiento del sistema. Generalmente todos los componentes del reloj se encuentran integrados en el propio microcontrolador y sólo se requieren unos pocos componentes externos, como un cristal de cuarzo o una red RC, para definir la frecuencia de trabajo.

El PIC 18F 4550 incorpora un PLL (Circuito de Fase Enlazada), el cual permite multiplicar n veces la frecuencia de entrada. De esta manera se puede tener un oscilador externo de baja frecuencia y con la activación del PLL se podrá trabajar con una frecuencia interna superior a la de entrada. Este μC puede operar con 12 distintos modos de oscilador entre los que destacan:

- XTPLL. Cristal de cuarzo o circuito resonador con PLL activado.
- HS. Cristal de alta velocidad.
- HSPLL. Cristal de alta velocidad con PLL activado.
- EC. Señal de reloj externa.
- ECPLL. Señal de reloj externa con PLL activado.
- INT. El μC genera su propia señal de reloj interna sin necesidad de componentes adicionales.

En la figura 3.3b se muestra la configuración usada en este proyecto. Se usó un cristal de cuarzo de 20 Mhz como oscilador, y se activó la opción HSPLL para trabajar con una velocidad interna de 48 MHz. Es importante resaltar que el μC divide la Frecuencia del Oscilador (F_{OSC}) entre 4 para definir su Frecuencia de Instrucción (F_{INST}), con esta F_{INST} se puede calcular el Periodo de cada instrucción (T_{INST}) en el μC , es decir:

$$f_{INST} = \frac{f_{OSC}}{4} = \frac{48\text{MHz}}{4} = 12\text{MHz}, \quad T_{INST} = \frac{1}{f_{INST}} = \frac{1}{12\text{MHz}} = 83.3\text{nSeg}$$

III.3.1.3 Reset

El Reset en un μC activa el ciclo de máquina de reinicio. En este ciclo, la mayoría de los dispositivos internos del μC toman un estado predeterminado. En los μC se requiere una terminal de reset para reiniciar el funcionamiento del sistema cuando sea necesario. La terminal de reset en los PIC se denomina MCLR (Master Clear) y produce el reinicio cuando se le aplica un nivel lógico bajo. Para tener un control sobre la señal de reinicio del sistema, se puede conectar un pulsador tal como se muestra en la figura 3.3c, y conseguir un reset manual llevando momentáneamente la terminal MCLR a tierra cada vez que se presiona el pulsador. El reset permanecerá en estado bajo mientras esté el pulsador presionado y no comenzará la secuencia de arranque hasta que no se libere, suministrando así un nivel lógico '1' al MCLR.

Este μC también permite el llamado Power-On Reset (POR), que proporciona un reset al μC en el momento de conectar la fuente de alimentación. El PIC dispone de un temporizador denominado Reset PWRT (Power-Up Timer), que proporciona un retardo de 72 mseg desde el momento de la conexión a la alimentación. Un reset se mantiene durante este tiempo, garantizando que V_{DD} alcance un nivel aceptable de tensión para un arranque correcto del sistema y permite que todos los componentes se estabilicen después del ciclo de reinicio (reset).

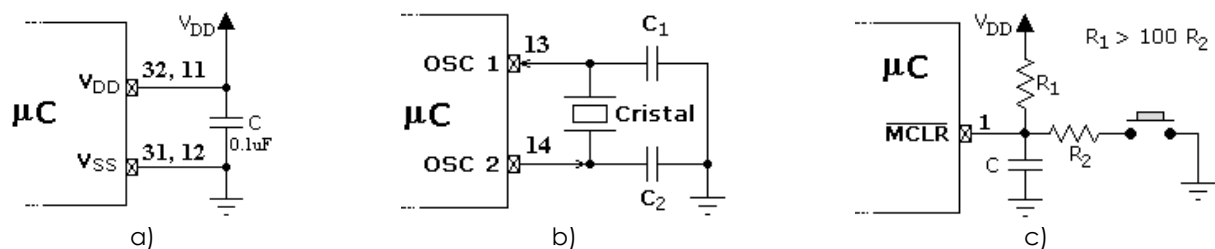


Figura 3.3 PIC 18F 4550: (a) Alimentación, (b) Oscilador, (c) Reset.

III.3.1.4 Modulación por ancho de pulso (PWM)

El módulo PWM o Modulación por Ancho de Pulso (Pulse Width Modulation) permite obtener en las terminales CCPx una señal periódica a la que se puede modificar su ciclo de trabajo de encendido en alto (Duty Cycle); ver la figure 3.4. De esta forma, la tensión aplicada a la carga es proporcional al T_{ON}/T controlando, por ejemplo, la velocidad de motores, luminosidad de lámparas, etc. La resolución de salida es de hasta 10 bits.

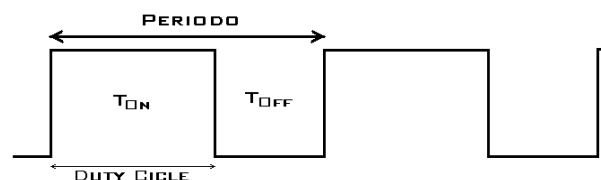


Figura 3.4 Señal PWM.

El periodo se especifica configurando un divisor de frecuencia, el registro PR2 y puede ser calculado usando la formula:

$$PR2 = \left[\frac{PWM_{PERIODO}}{4 \cdot T_{OSC} \cdot (Pr\ eescala_TMR2)} \right] - 1$$

El ciclo de encendido (T_{ON}) del módulo PWM se configura con los registros CCPR1L y CCP1CON. La siguiente ecuación se usa para calcular el ciclo de trabajo PWM en tiempo:

$$PWM_{DUTYCYCLE} = (CCPR1L : CCPCON) \cdot T_{OSC} \cdot (Pr\ eescala_TMR2)$$

Se propone un valor de ciclo de trabajo en segundos, y para conocer el valor que se debe cargar a CCPR1L y CCP1CON se despeja:

$$CCPR1L : CCPCON = \frac{PWM_{DUTYCYCLE}}{T_{OSC} \cdot (Pr\ eescala_TMR2)}$$

CCPR1L y CCP1CON pueden escribirse en cualquier momento, pero el valor del ciclo de trabajo no se carga en CCPR1H hasta después de que ocurra la coincidencia entre PR2 y TMR2 (un período completo).

III.3.1.5 Iluminación

Se agregó una lámpara de Diodos Emisores de Luz (LED's) para cubrir los requerimientos sobre la iluminación de las muestras a examinar. Esta lámpara está compuesta por 16 LED's blancos superbrillantes, el circuito se muestra en la figura 3.5 y se conecta al modulo PWM del μC para poder variar su intensidad. Es importante resaltar que el transistor TIP41 estará trabajando entre corte y saturación.

Con $V_{RC2} = 5V$, $V_{CC} = 5V$, $V_{BE} = 2V$, $V_{CE(sat)} = 1.5V$, $R_B = 10K\Omega$, $R_1 = 150\Omega$, $V_{DIODO} = 2.6V$, se tiene:

$$V_{CC} - I_{R1}R_1 - V_D - V_{CE(SAT)} = 0$$

$$I_R = \frac{V_{CC} - V_D - V_{CE(SAT)}}{R_1}$$

$$I_R = \frac{5 - 2.6 - 0.2}{150} = 14.66mAmp$$

$$I_C = 16(I_R) = 16(14.66 \times 10^{-3})$$

$$I_C = 234.66mAmp$$

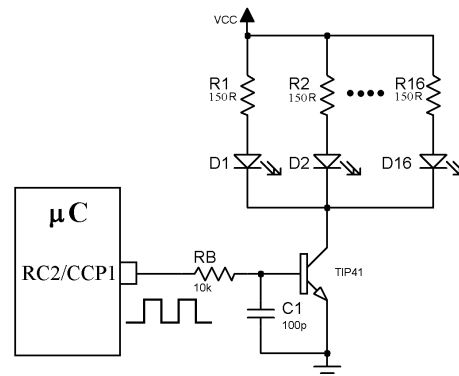


Figura 3.5 Lámpara de Leds.

III.3.1.6 Botones de control para el control manual

Para el control de los motores en modo manual se usaron 3 botones pulsadores (*push button*) normalmente abiertos, en la figura 3.6a se muestra la conexión que se uso para cada uno de los tres botones, las terminales de conexión al microcontrolador van desde RB0 a RB2, estos botones están conectados para obtener en las terminales del μC un nivel lógico '1' al no ser presionados, de esta forma el μC detectará la caída a nivel lógico '0' cada que se presiona un botón.

III.3.1.7 Indicadores

Se uso un LED tricolor para indicar al usuario el modo en el que se está trabajando, en color verde indica modo manual, color morado indica que el control es a través de la PC, en color rojo indica que el motor gira en sentido horario, en color azul indica que el motor gira en sentido anti horario. Su conexión se muestra en la figura 3.6b.

También se uso un display de 7 segmentos de ánodo común, el cual tiene la función de indicar al usuario el motor que está seleccionado. Este display se acopló a un decodificador de BCD a 7 segmentos (74LS247) con la intención de sólo utilizar 4 pines del μC para manejarlo, su conexión se muestra en la figura 3.6c.

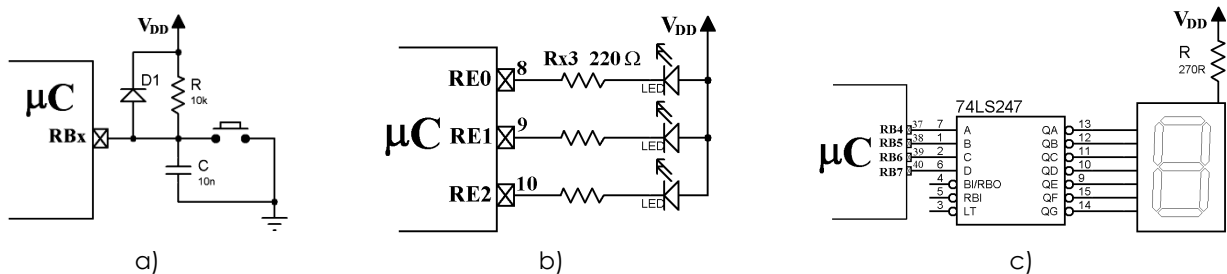


Figura 3.6 (a) Botones, (b) LED Tricolor, (c) Display 7 segmentos.

III.4. POTENCIA

La etapa de potencia es la encargada de suministrar el voltaje y la corriente necesaria para el funcionamiento de los motores, es la conexión entre la parte lógica del sistema y los motores. Al comienzo de este proyecto se consideró e implementó el sistema con motores de pasos de imán permanente de los tipos bipolares y unipolares de 2 fases, ya que éstos se encuentran con mayor disponibilidad en el mercado y su costo de adquisición es accesible.

En el caso de los motores de pasos **bipolares** se necesita de 4 terminales de conexión del μC y 2 puentes H (*driver*) para manejar a un motor, considerando que se requiere controlar 4 motores; el número de terminales del μC y los drivers crecen considerablemente haciendo que se necesite más de un μC y varios drivers para lograr el control de todos los motores, ocasionando el incremento del tamaño y por lo tanto el costo de la tarjeta.

Para la implementación de los motores de pasos **unipolares** se necesitan 4 pines de conexión del μC y 4 transistores Darlington (*driver*), además de una conexión directa al voltaje positivo de la fuente que polarizará a cada bobina de los motores, tomando las mismas consideraciones del caso anterior, se podría creer que se necesitan igual o más componentes que para los motores de pasos los bipolares, sin embargo; aprovechando la conexión en común que tienen los motores unipolares, se pudo disminuir la cantidad requerida tanto de terminales del μC como de drivers; con esta conexión en común de los motores se logró hacer un **multiplexor**, lo anterior se logró sustituyendo la mayoría de los drivers por Rectificadores Controlados de Silicio (SCR). De esta forma sólo se necesitó de un sólo driver para el control de todos los motores, sólo 4 pines de conexión del μC y una terminal junto con un SCR para cada motor que se deseara conectar.

III.4.1 Driver

Como se mencionó en el capítulo 2, los motores de pasos necesitan del flujo de corriente a través de sus bobinas en una secuencia determinada, para el caso de los motores unipolares se necesita como driver un transistor Darlington por cada bobina, como se muestra en la figura 3.7a, este transistor (Q) se ha sustituido por el circuito mostrado en la figura 3.7b, debido a la configuración del circuito que corresponde a un transistor Darlington, a partir de ahora se ilustrará con un inversor como se indica en esta figura. En el mercado existen diferentes CI que contienen varios transistores Darlington como el de la figura 3.7b en un solo encapsulado, como lo muestra la figura 3.7c.

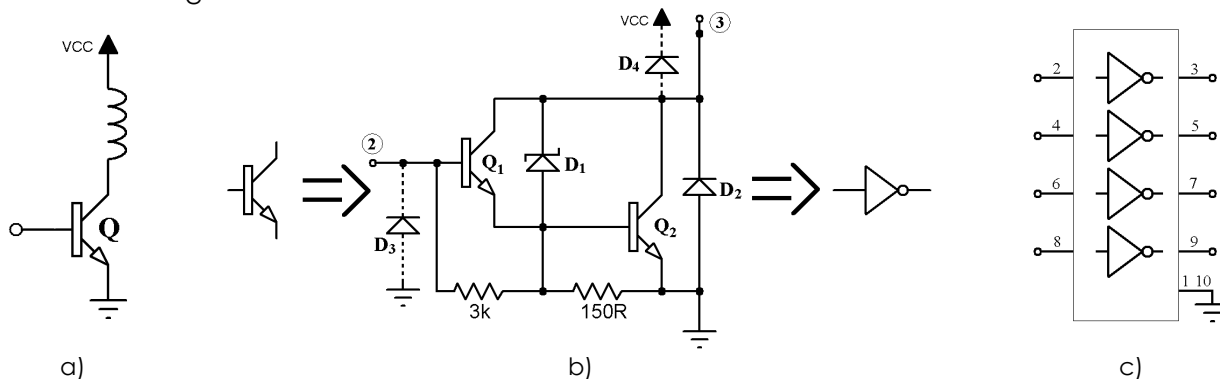


Figura 3.7 Driver para motor unipolar: (a) Representación, (b) Conexión Darlington, (c) CI STA401A.

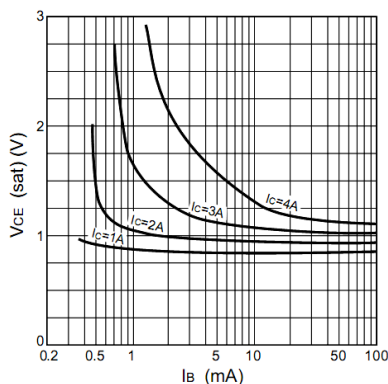


Figura 3.8 Características típicas de $I_c - V_{CE}$.

El driver que se utilizó en esta etapa es el STA401A, este CI se selecciono por contener en un solo encapsulado DIP a 4 transistores en conexión Darlington NPN como se muestra en la figura 3.7c, y por que puede soportar hasta 60 V_{CE} y una I_c de hasta 4A, con una ganancia (h_{fe}) mínima de 1000.

En la figura 3.8 se presenta un gráfica extraída de la hoja de especificaciones, en ella se observa que con un par de miliamperes en su Base se obtiene más de 1A a través de su Colector.

Es importante tener en cuenta que al tratarse de una conexión en configuración Darlington, el circuito de la figura 3.7b se analizará como un solo transistor, es decir; como el transistor Q de la figura 3.7a. También debe considerarse que el transistor trabaja en las zonas de corte y saturación.

Con $V_{CC} = 5V$, $R_{Bobina} = 3.3\Omega$, $V_{CE(SAT)} = 1V$, se tiene:

$$V_{CC} - I_{BOBINA} R_{BOBINA} - V_{CE(SAT)} = 0 \quad ; \quad I_{BOBINA} = \frac{V_{CC} - V_{CE(SAT)}}{R_{BOB}} = \frac{5-1}{3.3} = 1.21A$$

Este CI incluye dos diodos de protección en cada uno de sus transistores (ver figura 3.7b): D₁ que es un diodo Zener conocido como diodo *avalancha* y funciona para tratar de mantener estable el V_{CE} de Q₁ y V_{BE} de Q₂. El diodo D₂ sirve para asegurarse de que el V_{CE} de Q₂ no pase de 0.7V en caso de que ocurra una inversión en la polaridad del transistor. El diodo D₃ no se encuentra implementado dentro de este CI, pero se agregó externamente al circuito; este diodo cumple la misma función que D₂ pero en el V_{BE} de Q₁ y Q₂. El diodo D₃ es el 1N4007, y se eligió por soportar un voltaje de 1000V entre sus terminales y tener un voltaje en conducción (V_F) de 1.1V a 1A.

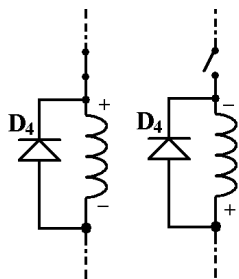


Figura 3.9 Protección contra impulsos inductivos.

El diodo D_4 de la figura 3.7b tampoco está implementado dentro de este CI, pero se agregó externamente al circuito a manera de que quede en paralelo a la bobina del motor, la razón es que, como se trabaja en cortos periodos de tiempo, las bobinas pueden generar un **impulso inductivo** cada vez que se realiza la conmutación, esto puede llegar a dañar a los elementos circundantes o al sistema mismo. Así, cuando la fuente de voltaje se desacopla rápidamente, la polaridad del voltaje a través del inductor es tal que encenderá el diodo y conducirá en la dirección indicada por éste en la figura 3.9.

Dado que la corriente que se estableció a través del inductor debe circular directamente a través del diodo, éste debe ser capaz de soportar el mismo nivel de corriente que circulaba a través del inductor antes de que la fuente se desacoplara. El ritmo al cual la corriente se colapsa la controlarán la resistencia del inductor y el diodo. El diodo colocado es el STPS5L40 del fabricante STMicroelectronic, el cual soporta una corriente de conducción I_f de 5 Amp.

III.4.2 Aislamiento

Para evitar que el ruido magnético generado por las bobinas de los motores afecte al funcionamiento lógico de la etapa de control, se ha implementado un aislamiento. Esta parte se construyó con 4 optoacopladores, como los mostrados en la figura 3.10a, el CI que se utilizó fue el LTV847 del fabricante Lite On, este CI contiene 4 optoacopladores en un solo encapsulado de 16 pines. En la figura 3.10b se muestra la conexión que se utilizó para este CI. Por simplicidad y legibilidad en los consecuentes diagramas, se representará como se muestra en la figura 3.10c, donde sólo se aprecian las entradas del lado izquierdo y las salidas del lado derecho; pero, teniendo siempre en cuenta que dentro de él existe la conexión mostrada en la figura 3.10b.

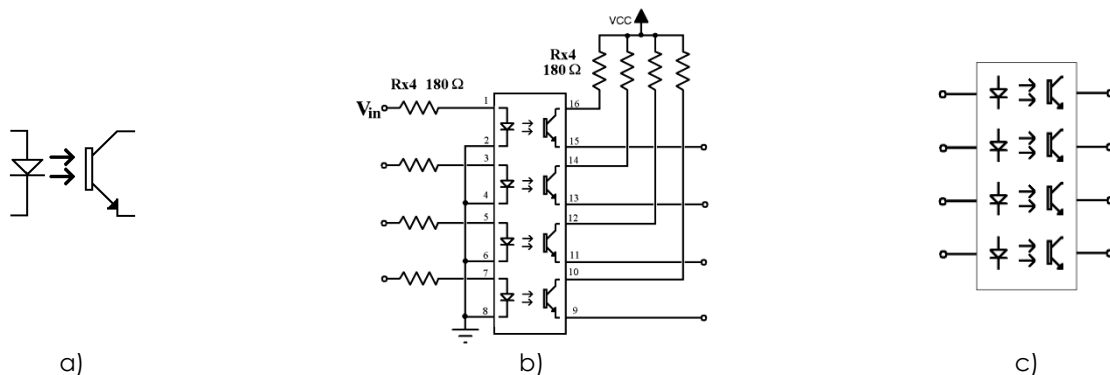


Figura 3.10 Aislamiento: (a) Opto acoplador, (b) Conexión del CI LTV847, (c) Representación del LTV847.

Con $V_{in} = 5V$, $V_F = 1.2V$, $V_{CC} = 5V$, $V_{CE(SAT)} = 0.2V$, $R = 180\Omega$, $V_{BE} = 1.5V$ (este último es del transistor Darlington de la figura 3.7a), se tiene:

$$V_{in} - I_{DIODO}R - V_F = 0 \Rightarrow I_{DIODO} = \frac{V_{in} - V_F}{R} = \frac{5 - 1.2}{180} = 21.11mAmp$$

$$V_{CC} - I_C R - V_{CE(SAT)} - V_{BE} = 0 \Rightarrow I_C = \frac{V_{CC} - V_{CE(SAT)} - V_{BE}}{180} = \frac{5 - 0.2 - 1.5}{180} = 18.33mAmp$$

III.4.3 Conexión de los Motores

En la figura 3.11 se puede observar la conexión del μC a un sólo motor (para fines prácticos se asume que el μC a través de sus terminales RA0 – RA3 nos brinda la secuencia necesaria para activar las bobinas de los motores con un voltaje en sus terminales de 5V, ya que el funcionamiento del μC se describirá más adelante), con esta conexión se tiene control sobre este motor, pero hay que recordar que se necesitan controlar 4 motores.

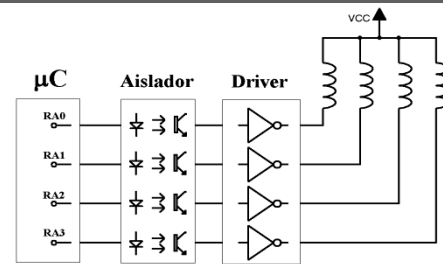


Figura 3.11 Conexión del μC a un sólo motor.

La figura 3.12 muestra el diagrama resultante de conectar dos motores, de seguir conectando motores de esta forma, resultaría que para controlar a 4 motores se necesitará de al menos 16 terminales del μC , 4 aisladores y 4 drivers, haciendo que, el tamaño y costo de la tarjeta se incremente considerablemente.

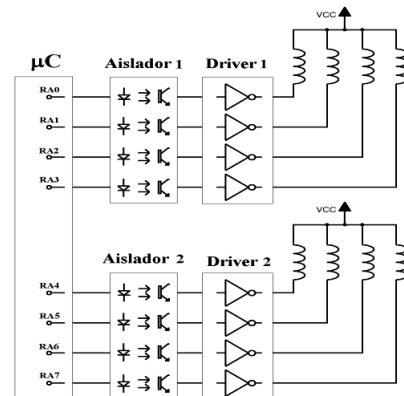


Figura 3.12 Conexión del μC a varios motores.

Tomando en cuenta que el μC solo cuenta con 32 terminales disponibles, no sería suficiente un solo μC . Siendo la restricción más importante el uso de uno sólo, ya resultaría deficiente utilizar dos. Por lo que se decidió realizar un multiplexaje de los motores.

III.4.4 Multiplexaje de los motores

La teoría básica del SCR dice que para dispararlo solo basta con aplicar un pequeño pulso de corriente (I_G) a su compuerta (G) entre 0.1 y 50 mA, con un voltaje superior a 0.7 V. Una vez disparado el SCR permanecerá conduciendo y no se podrá desactivar, a menos que; la corriente de ánodo a cátodo (I_A) caiga por debajo de un valor mínimo, llamada *corriente de retención* (normalmente de 1 mA) o que la corriente I_G sea mayor que I_A .

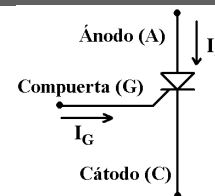


Figura 3.13 SCR.

Al colocar un SCR en la conexión común del motor como muestra la figura 3.14, se puede seleccionar al motor activando el SCR correspondiente. Se realizó una prueba con el circuito de la figura 3.14 y se pudo comprobar que el motor giraba sólo cuando el SCR estaba activado.

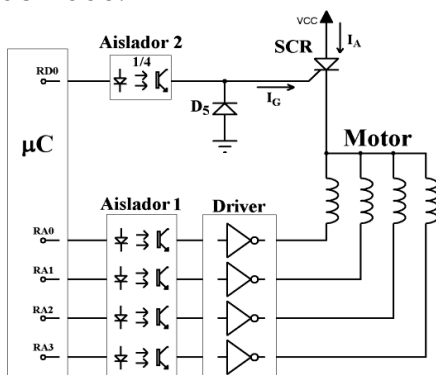


Figura 3.14 Conexión del SCR al motor.

El diodo D5 se agregó para protección, asegurando que el voltaje de la Compuerta (V_G) no exceda de 0.7V en caso de ocurrir una inversión en la polaridad. El SCR seleccionado es el C106D del fabricante On Semiconductor, y se seleccionó porque necesita de poca corriente en su compuerta para activarse ($I_G > 200\mu A$), soporta 200V y 2.5A.

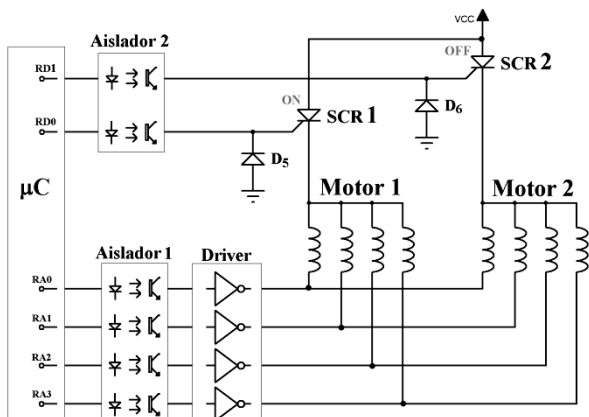


Figura 3.15 Conexión del dos SCR a dos motores.

La siguiente prueba se realizó con el circuito mostrado en la figura 3.15, la intención de este circuito es multiplexar a los motores, se disparó al SCR1 y se mantuvo desactivado al SCR2, después se mandó la secuencia de pulsos que harán que el motor gire y finalmente se desactivará al SCR1. Al realizar la prueba, el motor 1 giraba a la velocidad y sentido que debía hacerlo, pero el motor 2 también giraba, aunque de manera inestable, es decir; vibraba e intentaba girar igual que el motor 1, aun cuando el SCR2 estaba desactivado, incluso se desconectó físicamente la conexión entre el motor 2 y el SCR2; el resultado fue el mismo.

Tras varias pruebas se llegó a la conclusión de que al existir una conexión física entre las bobinas de los motores, la corriente circulaba como lo indica la figura 3.16a, ésta es la causa de que el motor no seleccionado vibrara e intentara girar. Para limitar el flujo de corriente en el motor no seleccionado se colocó un diodo en serie a cada bobina, como lo muestra la figura 3.16b, de ésta forma se asegura que la corriente solo circule como muestra dicha figura.

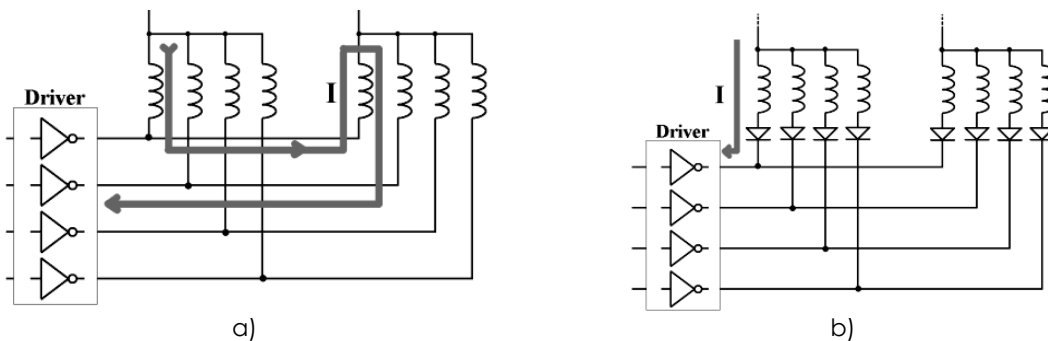


Figura 3.16 Sentido de la corriente a través de las bobinas, a) Sin diodo, b) Con un diodo en serie.

Los diodos seleccionados son los RL203 ya que soportan un voltaje en conducción (V_F) de 50 V y una corriente en conducción (I_F) de 2A, estos diodos se seleccionaron dado que al estar conectados en serie con las bobinas, los diodos deben soportar la misma o mayor cantidad de corriente que circula a través de ellas (1.2A).

Al estar terminado el multiplexaje de los motores, se concluyó que: sí en determinado momento se necesita implementar el control para más de 4 motores en la aplicación; sólo se necesita agregar dispositivos externos como: un aislador y un SCR con su respectiva terminal de conexión. Dado que no se requiere de gran número de componentes externos y se puede requerir el control de más de 4 motores, se determinó realizar la tarjeta electrónica para controlar 8 motores de pasos unipolares.

Una vez determinado el número de motores, entonces se hicieron pruebas con el circuito mostrado en la figura 3.17, que funciona de la siguiente manera:

1. Se dispara al SCR seleccionado, manteniendo la corriente I_G .
2. Se manda la secuencia de pulsos para girar al motor, como solo estará activado un sólo SCR los motores restantes permanecen inactivos.
3. Se desactiva al driver, esto causa que $I_A = 0$ A.
4. Se estable un tiempo de 50 mSeg para asegurar que $I_G > I_A$.
5. Se desactiva al SCR seleccionado haciendo que $I_G = 0$ A.

Con este circuito se tienen como ventajas que solo se necesite de un solo driver para el manejo de todos los motores, 4 terminales del μC para mandar la secuencia de control de giro, 1 terminal de conexión por cada motor que se desee controlar, en este caso son 8 terminales para 8 motores, es decir; para manejar 8 motores de pasos unipolares de 2 fases se necesitaron de 12 terminales del μC .

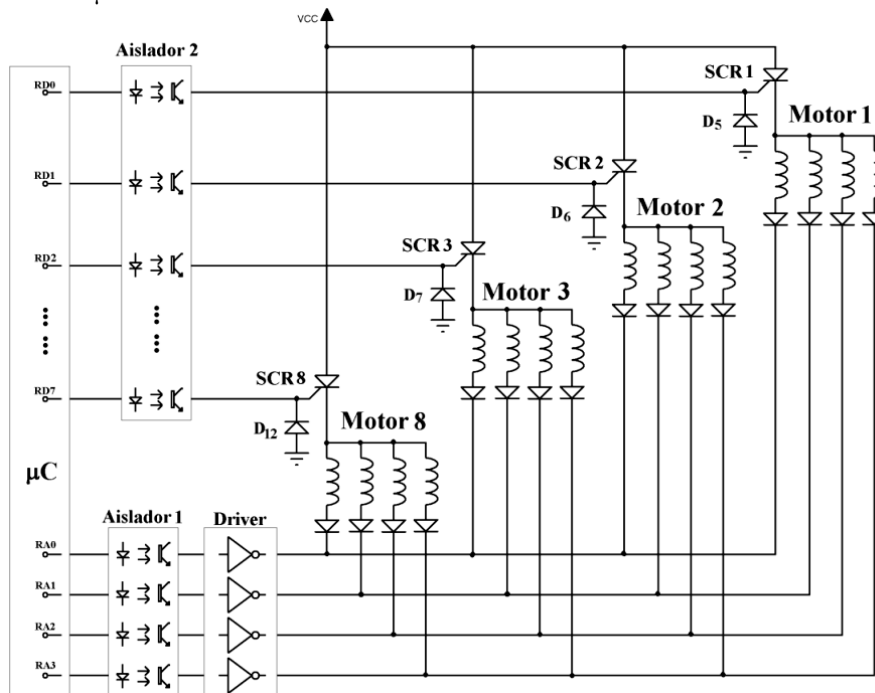


Figura 3.17 Circuito para el control de los motores.

El consumo energético máximo de este circuito ocurre cuando se activan dos bobinas, como nunca están activadas más de dos, se hace el cálculo para éste caso.

Tabla 3.1 Cálculo aproximado de consumo energético de la tarjeta electrónica.

V_{DD}		V_{CC}	
Consumo de corriente de:	Amperes	Consumo de corriente de:	Amperes
Microcontrolador	0.005	SCR	0.019
Botones	0.0045	Driver	0.019
Led indicador tricolor	0.011	Bobina 1	0.605
Display 7 segmentos	0.062	Bobina 2	0.605
Aislador 1	0.022		
Aislador 2	0.022		
Lámpara (a máxima intensidad)	0.235		
Total de consumo para V_{DD}:	0.3516	Total de consumo para V_{CC}:	1.248

Capítulo **4**

Desarrollo del Código del Microcontrolador

Objetivo particular. En este capítulo diseña y desarrolla el código que será programado en el microcontrolador, el *firmware*. Este firmware es el encargado de integrar las funciones de comunicación y control del hardware.

IV.1. ESTRUCTURA DEL FIRMWARE

El firmware fue desarrollado en lenguaje C con el compilador C de Custom Computer Services Incorporated (CCS). El compilador C de CCS permite desarrollar programas en C enfocado a μ C PIC con las ventajas que supone tener un lenguaje desarrollado específicamente para un microcontrolador específico. Su facilidad de uso, su entorno de trabajo y la posibilidad de compilar para la gama alta, le confieren una versatilidad y potencia muy elevadas, por estas razones se seleccionó dicho compilador.

En la figura 4.1 se muestra de forma general el diagrama de flujo del firmware, este diagrama muestra los 9 procesos principales, que, a su vez contienen otros diagramas de flujo que se irán desarrollando a lo largo del capítulo. Como se puede ver en esta figura, el firmware cuenta con dos ciclos principales, uno cuando se establece el control manual y el segundo para el control a través de la PC.

Para el manejo de los motores en el modo manual, se utilizan 3 botones pulsadores (*push button*) normalmente abiertos, los cuales están conectados a cada una de las 3 interrupciones externas (INT RBx) que soporta el μ C. Una vez inicializados todos los puertos, módulos e interrupciones a utilizar, el microcontrolador espera a ser conectado por medio de USB, al no ser conectado, se pueden atender cada una de estas tres interrupciones externas. Al momento de conectarse se deshabilitan las interrupciones y de esta forma se deshabilita el control manual.

Cuando se ha conectado al puerto USB de la computadora, se establece una comunicación punto a punto. La comunicación es por comandos que el programa de computadora (se explicará en el siguiente tema) le manda al μ C, éste al recibirlo lo decodifica o interpreta, una vez que lo hace, ejecuta la acción correspondiente al comando previamente establecido en la memoria de programa, el μ C responde al programa de computadora con el mismo comando para indicar que el comando recibido se ha ejecutado satisfactoriamente.

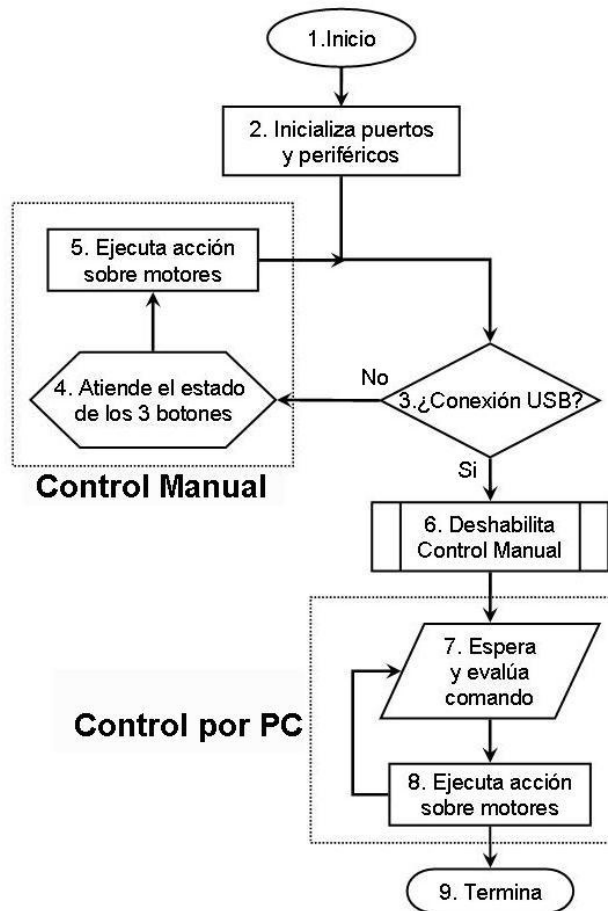


Figura 4.1 Diagrama de flujo del firmware del μ C.

A continuación se describe los procesos principales del firmware. Los extractos de código fuente equivalentes a cada diagrama de bloques se pueden consultar en el apéndice A.

IV.2. INICIALIZACIÓN DE PUERTOS Y PERIFÉRICOS

Este proceso configura los puertos del μ C como entrada o como salida, configura los módulos PWM, USB y las Interrupciones Externas RB0-2 del μ C.

IV.2.1 PWM

El módulo PWM se ha configurado para entregar una señal periódica de 2.9 KHz en el pin RC2, porque es el valor mínimo que se puede obtener con una frecuencia de oscilación de 48MHz; esta frecuencia resulta de la siguiente fórmula:

$$\left(\frac{1}{F_{osc}}\right) * (4) * (preescala_timer2) * (periodo + 1) = \left(\frac{1}{48 \times 10^6}\right) * (4) * (16) * (255) = 340\mu s \text{ o } 2.9\text{KHz}$$

Donde el ciclo de trabajo se puede variar solo con escribir al registro correspondiente un valor entre 0 y 1023.

IV.2.2 USB

El módulo USB está configurado para realizar una emulación del puerto serie (COM) a través de las especificaciones de la Clase de Dispositivo de Comunicación (CDC). Esta clase CDC se seleccionó por ser la más sencilla y rápida de implementar tanto en el μC como para el programa de computadora. Al emular un puerto serie se crea una conexión virtual de la recomendación Estándar RS232 entre la PC y el μC , pero el medio físico de comunicación es por el puerto USB. Una vez conectado el μC a la PC por medio del puerto USB, espera a que el Host enumere al dispositivo, entonces se creará temporalmente en la computadora un puerto virtual COM, el cual aparecerá en el administrador de dispositivos de la computadora.

IV.2.3 Interrupciones externas

Los botones de control manual serán conectados a las terminales de interrupción externa. Se habilitan las tres interrupciones externas que soporta este μC , una para cada botón del control manual. Para los dos botones que hacen girar al motor en uno u otro sentido se configura la interrupción para detectar los flancos de bajada, es decir; la transición de nivel alto a nivel bajo. Para el botón de selección de motor se configura la interrupción para detectar los flancos de subida. En la tabla 4.1 se muestra el estado de cada terminal del μC , las terminales que no aparecen se han definido como entrada.

Tabla 4.1 Estado de los pines del μC .

Pin	Estado	Se conecta
RA0 – RA3	Salida	Secuencia para motores
RB0 – RB3	Entrada	Botones
RB4 – RB7	Salida	Display 7 seg
RC2	Salida	Salida PWM

Pin	Estado	Se conecta
RC4 y RC5	Comunicación USB	
RC7	Entrada	Conexión USB
RD0 – RD7	Salida	Selección de motor
RE0 – RE2	Salida	Led tricolor

Antes de continuar con la descripción de los procesos de la figura 4.1 se explicarán las rutinas o funciones para el control de giro de cada motor. Se debe recordar que en el capítulo anterior ya se estableció el hardware para la conexión entre el μC y los motores.

IV.3. SECUENCIAS DE GIRO

Con las terminales RA0 hasta RA3 del μC se produce la secuencia de pulsos que activarán a los optoacopladores para que estos a su vez activen a los transistores Darlington y que al activarse estos transistores se genere el flujo de corriente a través de las bobinas de cada motor (ver figura 3.17). En la figura 4.2 se muestran las secuencias de pulsos para la excitación de un motor unipolar en modo de paso completo. El valor del ancho del pulso cuando esta a '1' lógico es aproximadamente 1 ms, se dice aproximado porque cuando se trabaja en un compilador C no se pueden precisar los tiempos de ejecución de cada instrucción, dado que el compilador agrega instrucciones, pero, para esta aplicación es aceptable margen de tiempo adicional que se tenga.

Tomando en cuenta que los motores son de 1.8° por paso, se tiene que, se necesitarán 200 pasos para completar una vuelta, así que al termino del paso 4 se volverá al paso 1 hasta completar los 200 pasos sí se desea completar una vuelta. También es importante resaltar que se estará trabajando alrededor de 1Kpulsos/segundo, con los cuales se tendrá un torque aproximado de 0.35 N·m, según la hoja de especificaciones del motor.

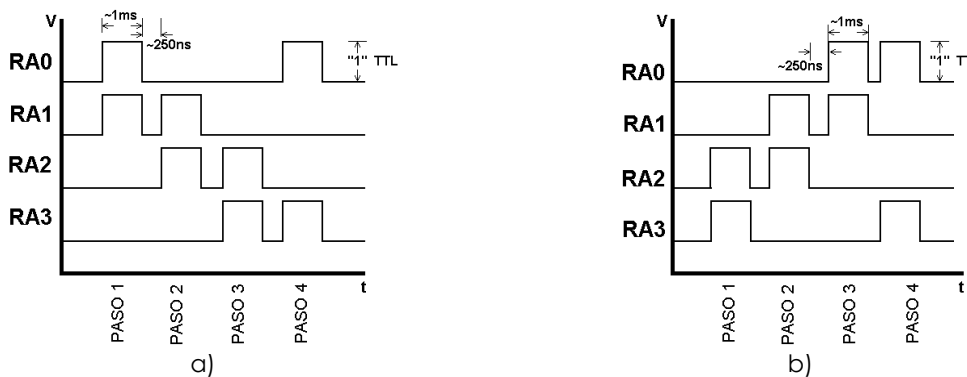


Figura 4.2 Secuencia de pulsos para giro del motor: (a) Giro a la izquierda, (b) Giro a la derecha.

Las figuras 4.2 a y b serán referidas como secuencia de **giro a la izquierda** y **giro a la derecha**, respectivamente.

En la figura 4.3 se muestra el diagrama de flujo de la función que realiza la secuencia de giro a la izquierda y a la derecha. Para realizar esta secuencia se necesitan de dos variables, una llamada Contador que sirve para realizar los cuatro pasos requeridos, y la otra variable llamada Puntero que sirve para apuntar el orden en que los pines RA0-3 se deban activar en nivel alto ó '1' lógico.

Las dos funciones son muy similares, lo único que las diferencia entre sí, es la inicialización del puntero y el corrimiento a la derecha o izquierda que deba realizar cada una.

Es importante tener en cuenta que cada vez que se mande a llamar a esta rutina o función, el motor seleccionado sólo avanzará cuatro pasos, esto tiene lugar a que la relación de engranes que estarán en cada eje de movimiento convertirá esta distancia angular de cuatro pasos en una distancia lineal de alrededor de 0.5 mm.

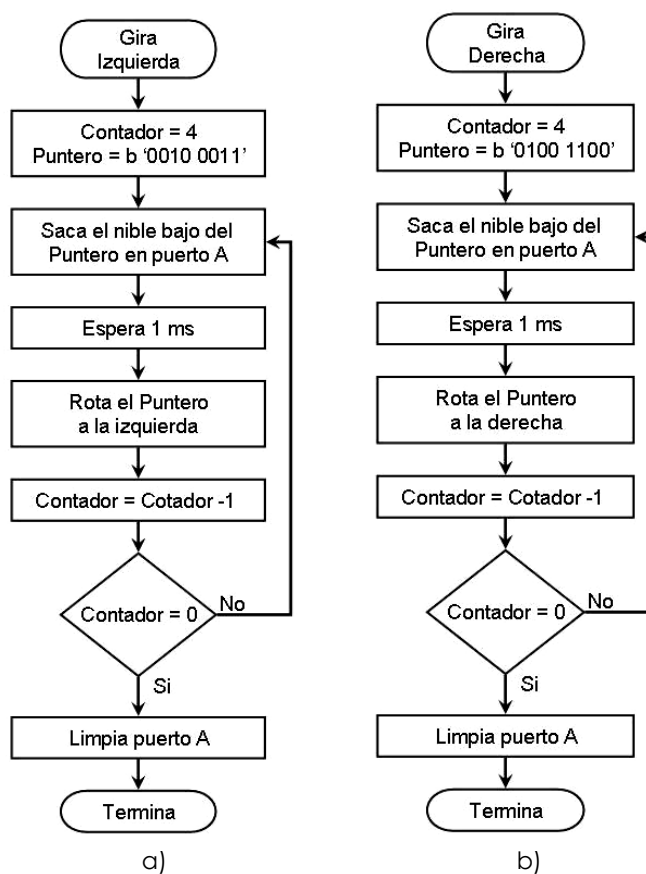


Figura 4.3 Diagrama de flujo para realizar la secuencia de giro: (a) izquierda, (b) derecha.

Para determinar el desplazamiento lineal de los ejes se considera lo siguiente:

Relación de engranes: 1° angular = 0.07 mm lineal (dato del fabricante del Equipo Mecánico-Óptico).

1 paso del motor = 1.8° (dato del fabricante del motor).

4 pasos del motor = 7.2°, por lo tanto:

(7.2) (0.07mm) = 0.504 mm de desplazamiento lineal de cada eje.

IV.4. CONEXIÓN USB

Este proceso es sumamente corto. En este bloque el firmware permanece en espera a ser conectado por medio de USB, esto se logra monitoreando el estado de la terminal RC7, el cual fue configurado para activar al módulo USB del μ C cuando detecta el voltaje del puerto USB de la computadora. Mientras no se detecta el voltaje se puede atender a las interrupciones externas previamente habilitadas.

IV.5. CONTROL MANUAL

Este subtema comprende los pasos 4 y 5 del diagrama de flujo de la figura 4.1. En realidad este bloque no se ejecuta de manera secuencial, es decir; no está monitoreando constantemente el estado de los botones del control manual.

Para realizar el control manual se atienden las interrupciones externas, el botón que hace girar al motor seleccionado hacia la izquierda fue asignado la Interrupción Externa 1 (será nombrado **Botón Izquierdo**), es decir la terminal RB0, para el botón que hace girar al motor seleccionado a la derecha fue asignado a la Interrupción Externa 3 (será nombrado **Botón Derecho**), es decir la terminal RB2, y para el botón que selecciona a un motor de los ocho disponibles fue asignado a la Interrupción Externa 2 (será nombrado **Botón Selección**), es decir la terminal RB1.

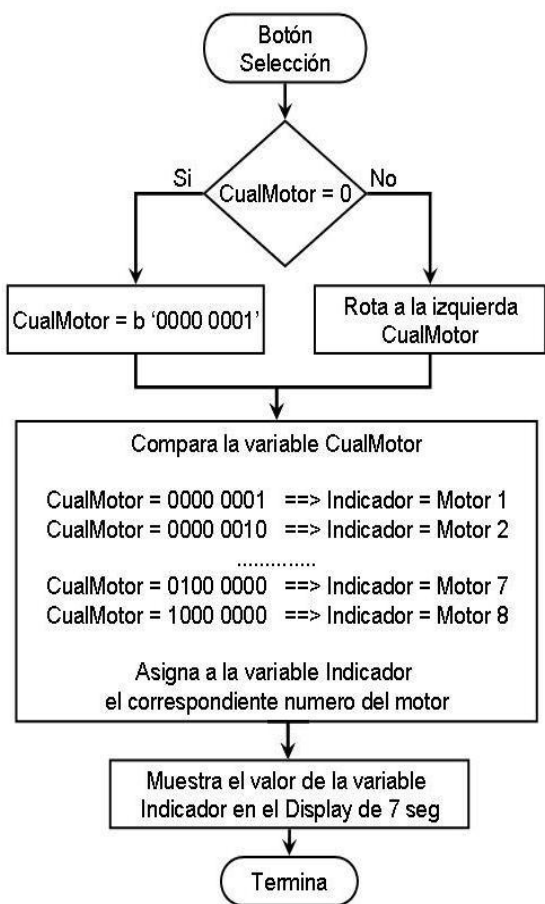


Figura 4.4 Diagrama de flujo del botón de selección.

En la figura 4.4 se observa el diagrama de flujo de la función que realiza la selección del motor. Esta función es llamada cada vez que se presiona el botón de selección. En esta función se usan dos variables, una que sirve para seleccionar al motor (*CualMotor*) y la otra para mostrar en un display de 7 segmentos el motor seleccionado (*Indicador*).

Al momento de energizar al hardware todas las variables del firmware toman un estado predeterminado, en esta función las variables *CualMotor* e *Indicador* toman el valor 0x00, con esto se asume que no estará seleccionado ningún motor y que el display de 7 segmentos mostrará el numero 0.

Una vez que se establecieron los valores de las variables, se espera a que se seleccione un motor, presionando el botón de selección. Cuando se presiona por primera vez el botón, lo que hace la función es asignarle el valor del primer motor (b '0000 0001'). Cuando se vuelva a presionar se correrá a la izquierda la variable *CualMotor* y entonces el motor seleccionado será el numero 2.

Una vez que se asignó el motor, también se asigna el valor del motor a la variable *Indicador* y se muestra en el display de 7 segmentos.

Se debe recordar que el display de 7 segmentos está conectado a un decodificador de BCD a 7 segmentos el cual, a su vez, está conectado al nibble alto del puerto B del μC (ver figura 3.6c). Cuando a la variable $\text{CualMotor} = \text{b}'0000\ 0001'$, se asignará al Indicador = $0x1F$, cuando la variable $\text{CualMotor} = \text{b}'0000\ 0010'$, se le asignará a Indicador = $0x2F$,... y así sucesivamente, para posteriormente mostrarlo por el puerto B. El valor del nibble bajo del Indicador en realidad no se usa, ya que el puerto B solo tiene implementados los terminales como salida a las terminales del nibble alto, que son las que le indicarán al decodificador que número mostrar en el display de 7 segmentos.

Una vez que se termina la función anterior, en la variable CualMotor queda almacenado el valor del motor seleccionado. Y a continuación se espera que se presione algún botón para hacer girar al motor.

En la figura 4.5 se observa el diagrama de flujo de la función que hace que el motor gire en uno u otro sentido. En este mismo diagrama se indican los dos botones, ya que el funcionamiento es prácticamente el mismo, lo único que cambia es la llamada de la función de gira a la izquierda o gira a la derecha. Las interrupciones detectan los flancos de bajada.

Recordemos que para activar a un motor, solo basta con mandar un pulso de un '1' lógico a través de los pines del puerto D del μC para activar el SCR, ver la figura 3.17.

Cuando se presiona el botón, lo primero que hace la función es disparar al SCR correspondiente al motor seleccionado, esto se logra tomando la variable CualMotor y sacándola directamente por el puerto D.

De esta forma sí la variable $\text{CualMotor} = \text{b}'0000\ 0001'$ el SCR activado será el correspondiente al motor 1, sí la variable $\text{CualMotor} = \text{b}'1000\ 0000'$ el SCR activado será el correspondiente al motor 8 (ver la figura 3.17).

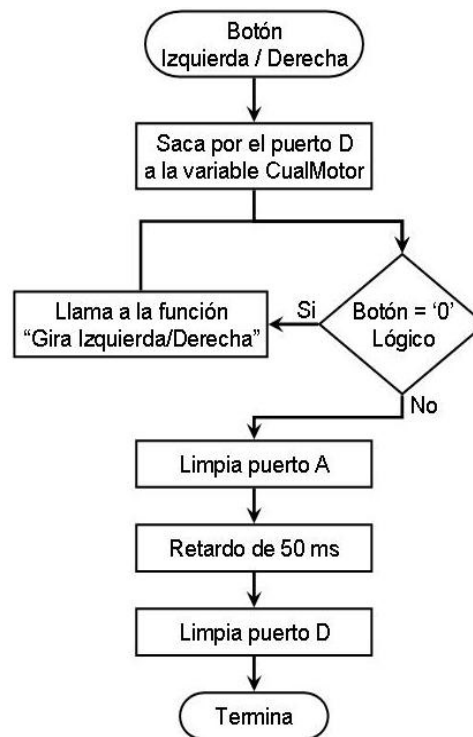


Figura 4.5 Diagrama de flujo de los botones izquierda/derecha.

Una vez que se ha disparado al SCR, se verifica que el botón siga presionado y si es así se llama a la función que hace que el motor gire a la izquierda o Derecha, dependiendo qué botón se haya presionado. Esta función que se manda a llamar corresponde a alguna secuencia de la figura 4.2.

Una vez que el botón se deja de presionar, se limpia el puerto A, ocasionando que el motor se detenga y con esto se logra que I_A del SCR sea $\approx 0A$. Se debe notar que el puerto D aun sigue manteniendo en alto el pin correspondiente al motor seleccionado, es decir, la variable CualMotor aún se está sacando por el puerto D, en este momento $I_G > I_A$ (ver figura 3.13), por lo tanto el SCR se desactivará, el siguiente paso es mantener un tiempo de de 50ms mediante un retardo para garantizar que el SCR se desactive, después se limpia al puerto D para que ningún SCR permanezca activado.

IV.6. DESHABILITACIÓN DEL CONTROL MANUAL

El control manual se puede operar por tiempo indefinido si así se desea, siempre que no se conecte el hardware por medio de USB. Una vez conectado, se desactiva el control manual, deshabilitando las interrupciones externas. Pasando al modo de control por computadora, en el que el firmware queda en espera de recibir comandos.

IV.7. CONTROL AUTOMATIZADO POR COMPUTADORA

Este subtema comprende los pasos 7 y 8 del diagrama de flujo de la figura 4.1. A partir de este modo, el firmware queda en un ciclo infinito del cual solo se podrá salir desconectando la alimentación del hardware. En la figura 4.6 se puede observar el diagrama de flujo del control automatizado por computadora.

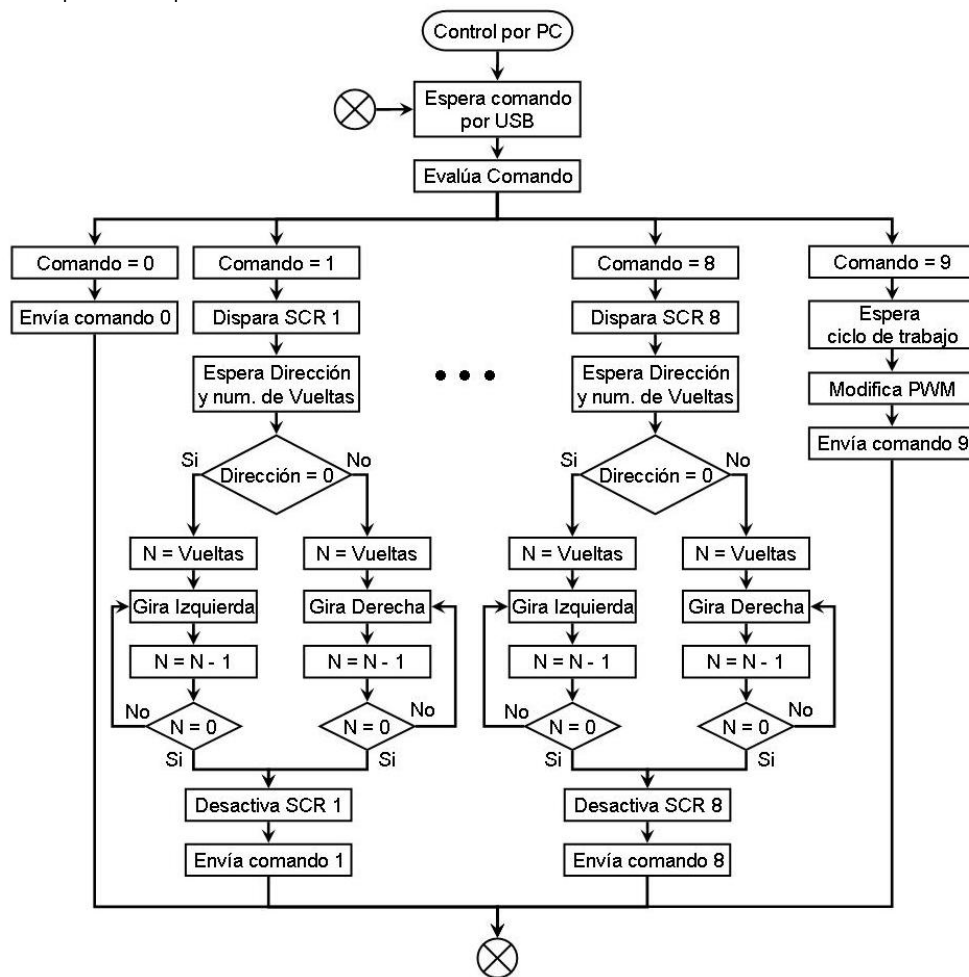


Figura 4.6 Diagrama de flujo del Control Automatizado por Computadora.

Como ya se ha comentado, el modo de trabajo del control por computadora es mediante comandos. En la memoria de programa del firmware se establecen las rutinas a seguir dependiendo el comando que se reciba, como se puede ver en la figura 4.6 los comandos están en decimal y van desde el 0 hasta el 9, en dicha figura solo se muestra el flujo de los comandos 0, 1, 8, y 9 pero el resto de los comandos siguen la misma estructura del comando 1; lo único que cambia es el SCR que se activa y el comando que reenvía a la computadora.

Para comprender mejor el funcionamiento de este diagrama de flujo se explica lo siguiente:

En la figura 4.7 se muestra la representación de un eje de movimiento de la mesa de posicionamiento, donde se puede ver que este eje cuenta con 500 posiciones, sobre las que se requiere realizar el posicionamiento. El programa de computadora se encargará de realizar el control de los motores, por ejemplo: este eje corresponde al motor 5, al inicio, se encuentra en la posición '0' como lo indica la letra **A**, se desea pasar a la posición '250' (letra **B**), el programa de computadora manda al μC el comando '5' para indicarle el motor a controlar; en la variable 'Dirección' el número '1' indicando que el motor debe girar a la derecha, y, en la variable 'Vueltas' manda el número 250, que son los giros que debe dar el motor para llegar a la posición deseada.

Sí se desea que se coloque en la posición 50 como lo indica la letra **C**, el programa de computadora calcula las variables y manda al μC el comando '5' para indicar que es el motor 5, en la variable 'Dirección' el número '0' indicando que el motor debe girar a la izquierda; en la variable 'Vueltas' manda el número 200, que son los giros que debe dar el motor para llegar a la posición deseada.

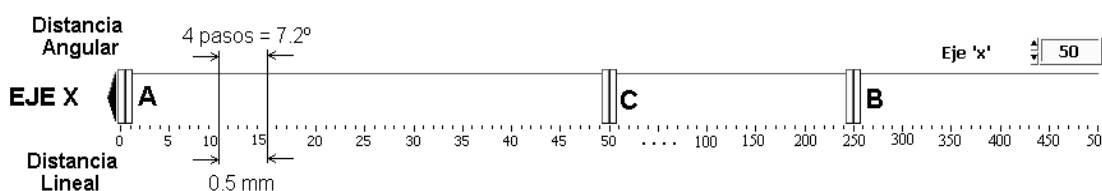


Figura 4.7 Representación de un eje de movimiento.

De esta forma el μC solo se dedica a controlar la dirección y número de giros que deba realizar cada motor. Regresando a la figura 4.6 y en la parte que se refiere al control de los motores, el programa de computadora mandará cuatro bytes al μC , donde, el primer byte es la variable Comando que puede ser un número entre 1 y 8, una vez identificado el comando, lo primero que hace el firmware es activar al SCR correspondiente para establecer una $I_G > 0A$. Después espera el segundo Byte, donde se establece la Dirección del giro, '0' para indicar que se debe girar a la izquierda ó '1' para indicar que se debe girar a la derecha. Con el tercer y cuarto Byte el firmware genera un registro de 16 bits tomando al tercer byte como los 8 bits más significativos, dicho registro contiene el número de Vueltas que realizará el motor seleccionado. Una vez que termina de dar los giros, se desactiva al SCR y se reenvía el mismo comando que fue recibido para indicar que el proceso ha terminado satisfactoriamente.

Para el control de la iluminación, el programa de computadora manda 3 bytes, en el primero se manda el Comando 9 para indicar que se va a manipular el ciclo de trabajo de la señal PWM y con el segundo y tercer Byte se conforma un registro de 16 bits para ajustar el valor del Ciclo de Trabajo de la señal PWM, al final se reenvía el mismo comando que fue recibido para indicar que el proceso ha terminado satisfactoriamente.

El comando '0' lo único que hace es reenviar el mismo comando a la computadora, este comando se reservó así, ya que, habrá ocasiones donde se requerirá que el μC no haga nada, y para el programa de computadora conviene este comando (se explicará más adelante), a este comando se le ha nombrado Comando Nulo.

Como se pudo ver en éste capítulo, el firmware es simple. Parte de la simplicidad se basa en que la tarea de controlar a los 8 motores de pasos se divide entre el firmware y el programa de computadora que se desarrollará en el siguiente capítulo.

Capítulo **5**

Desarrollo del Software

Objetivo particular. En este capítulo se presenta el desarrollo y la descripción del programa de computadora, que está conformando por tres aplicaciones: interfaz de usuario para el control de los motores, programa para manipulación de imágenes y programa de comparación de imágenes. Se mostrarán las ventanas principales y se detallará las secciones más importantes.

V.1. ESTRUCTURA DEL PROGRAMA DE COMPUTADORA

Este programa fue realizado en LabVIEW 8.5 de National Instruments, se seleccionó este entorno de programación por ser un entorno de gráfico que facilita y acelera la programación. Cuenta con una amplia gama de controles, indicadores, funciones, bibliotecas, etc. Adicionalmente, incorpora en la Ayuda, un extenso repertorio de códigos que pueden servir como apoyo en tiempo de programación.

A continuación se muestran las ventanas principales de las aplicaciones que conforman al programa de computadora.

V.2. INTERFAZ DE USUARIO PARA EL CONTROL DE MOTORES

Para definir el posicionamiento de cada eje se ha desarrollado una interfaz de usuario con la que el usuario puede decidir la posición de cada eje, la interfaz se presenta en la figura 5.1.

En el lado derecho de la figura 5.1 se pueden ver los ocho controles que sirven para manipular a cada motor, estos controles son idénticos a los de la figura 4.7. Para explicar el funcionamiento de la interfaz se describe el algoritmo que sigue cada control. Se usan 8 controles de tipo 'Slide' configurados como variables de punto flotante, uno para cada motor, en su escala se muestran los límites mínimos y máximos. La forma de usar estos controles es haciendo click directamente sobre la barra o introduciendo un número en el cuadro de texto que aparece en el lado superior derecho.

Cada uno de estos controles se ha colocado dentro una estructura 'Event Structure', de forma que cuando cambie el valor de cualquier control provocará un evento, los eventos que se puede entender como una interrupción dentro del ciclo principal, de esta forma, el programa espera a que un control cambie de valor para ejecutar su código correspondiente.

Cada control ha sido nombrado con el motor al que controlará, es decir, el control llamado Motor 1 manipulará al motor 1, el control Motor 2 manipulará al motor 2, y así sucesivamente.

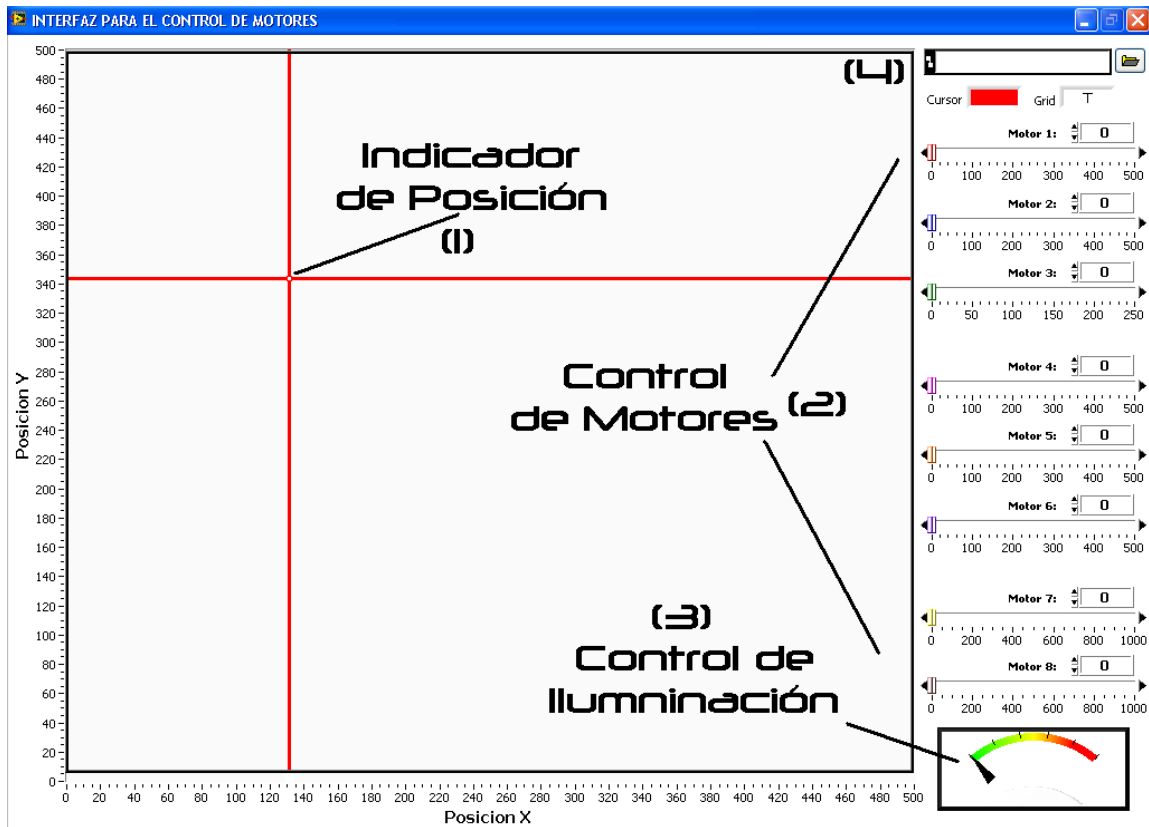


Figura 5.1 Interfaz de Usuario para el Control de los Motores.

Como ya se mencionó en el capítulo anterior, cuando se conecta por medio del puerto USB, el μC permanece en espera a que la interfaz de usuario le envíe cuatro Bytes de información, el motor a manipular, dirección del giro y el número de giros que deba realizar cada motor (éste último lo conforman el tercer y cuarto Byte). El algoritmo que se usa para obtener esta información se muestra en el diagrama de flujo de la figura 5.2.

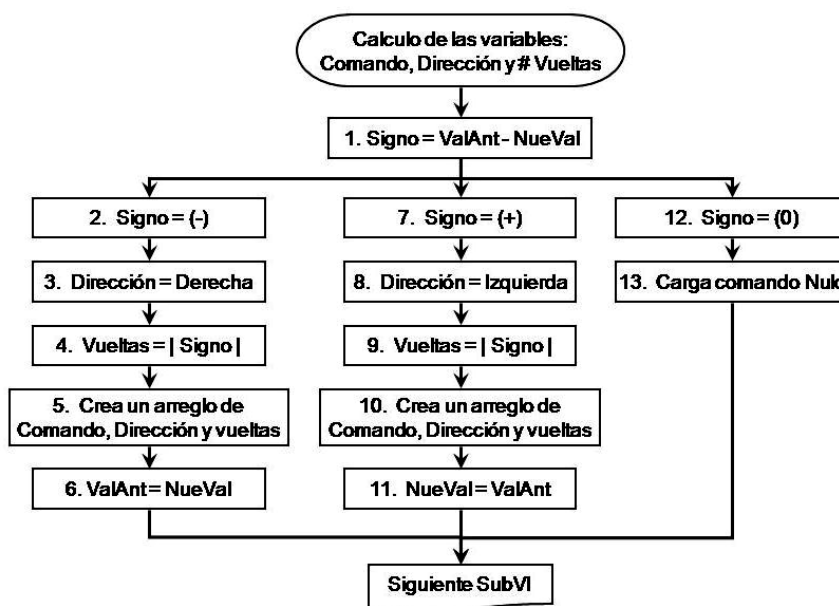


Figura 5.2 Diagrama de flujo para obtener las variables Comando, Dirección y Vueltas.

Para este algoritmo se usó una variable de tipo punto flotante, a la que se asigna el valor cero cada vez que se inicia el programa, ésta se le ha nombrado Valor Anterior (ValAnt). La variable NueVal corresponde a la nueva posición en la que se desea colocar el eje. El diagrama de la figura 5.2 se explicará con un ejemplo.

Al inicio del programa los controles se configuran con el valor = 0. Suponiendo que se desea controlar la posición del Motor 1, se consideran las posiciones A, B y C de la figura 4.7. Cuando se indique la posición B, se tendrá el valor del primer Byte = 1, por ser el motor 1 al que se va a manipular. Siguiendo el flujo de la figura 5.2 se tiene:

$$1. \text{Signo} = \text{ValAnt} - \text{NueVal} \rightarrow \text{Signo} = 0 - 250 \rightarrow \text{Signo} = -250$$

Se toma solo el signo para determinar en qué dirección debe girar el motor.

2. El signo es negativo (-).

3. Al ser signo negativo la dirección = derecha. El valor del segundo Byte = 1.

4. Se toma el valor absoluto de la variable signo y se asigna a la variable vueltas, para este caso Vueltas = 250. Después esta variable de punto flotante es convertida en un entero de 16 bits y separada en dos bytes, donde el primer byte corresponde al byte más significativo.

5. Se crea un arreglo de las cuatro variables que posteriormente se enviarán por el puerto serie.

6. Ahora el valor al que se movió el motor corresponde a la variable ValAnt.

En este momento se han mandado las cuatro variables al μC y éste hará que el motor 1 realice 250 giros (recordar que 1 giro = 4 pasos) a la derecha. Ahora, el motor 1 se encuentra en la posición 250. Para cambiar a la posición C de la figura 4.7, siguiendo el flujo de la figura 5.2, se tendrá que:

$$1. \text{Signo} = \text{ValAnt} - \text{NueVal} \rightarrow \text{Signo} = 250 - 50 \rightarrow \text{Signo} = 200$$

Se toma solo el signo para determinar en qué dirección debe girar el motor.

7. El signo es positivo (+).

8. Al ser signo positivo la dirección = izquierda. El valor del segundo Byte = 0.

9. Se toma el valor absoluto de la variable signo y se asigna a la variable vueltas, para este caso Vueltas = 200. Después esta variable de punto flotante es convertida en un entero de 16 bits y separada en dos bytes, donde el primer byte corresponde al byte más significativo.

10. Se crea un arreglo de las cuatro variables que posteriormente se enviarán por el puerto serie.

11. Ahora el valor al que se movió el eje del motor corresponde a la variable ValAnt.

Nuevamente se han enviado las cuatro variables al μC y éste hará que el motor 1 realice 200 giros a la izquierda para posicionar el eje del motor 1 en la posición 50.

Cuando se realiza la resta en el bloque 1 de la figura 5.2, el algoritmo detecta si el valor de la variable Signo = 0 (esta situación ocurre cuando se introduce la misma posición en la que se estaba), este caso se refiere al bloque 12 de la figura 5.2, cuando esto sucede, no se realiza ninguna operación y se manda el comando Nulo al μC , este comando está definido en el código del μC para no realizar ninguna operación.

V.2.1 Decodificación de datos, SubVI 1

El diagrama de bloques de la figura 5.2 se muestra en la figura 5.3. Este subVI sólo obtiene el valor de las variables a enviar y forma un arreglo de cuatro bytes, cuando termina, el arreglo que contiene esta información pasa a un segundo subVI, este arreglo está representado por el indicador llamado 'Dato 2' de la figura 5.3.

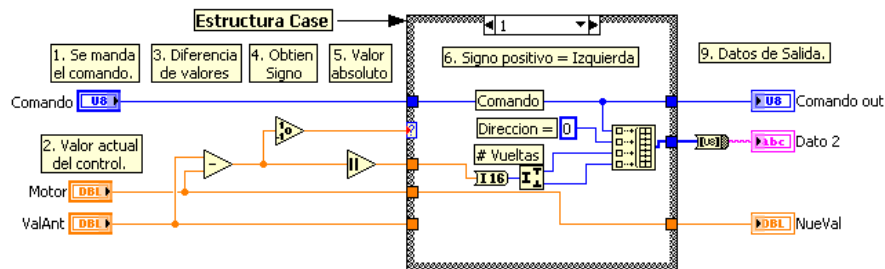


Figura 5.3 Diagrama de bloques para obtener las variables Comando, Dirección y Vueltas.

V.2.2 Envío de datos, SubVI 2

En la figura 5.4 se muestra el diagrama a bloques del código que envía datos por el puerto serie, únicamente la primera vez que se envía datos se configura el puerto serie, para esta aplicación es necesario indicarle el puerto (COM) que será usado y las características de comunicación, es decir, 8 bits de datos, 1 bit de paro y sin control de flujo. Después se limpian los datos que puedan estar en el buffer y se establece el tamaño del mismo (paso 2), en este caso el valor por default es 4096 bytes. En el paso 3 sólo se asigna un tiempo para permitir configurar el puerto. En el paso 4 se envían datos el puerto serie a través de la variable 'Dato', el bloque que escribe en el puerto serie (VISA write) indica si ocurrió un error, de esta forma se puede conectar a una estructura Case para saber cuando ocurre o no un error. Si no ocurre algún error, se ejecuta el paso 7, donde el puerto espera a recibir la cantidad de Bytes marcados en la variable 'Byte a recibir'. Después de recibir todos los Bytes, en el paso 8 se cierra el puerto serie. Para terminar este subVI se muestra en el indicador booleano si ocurrió algún error, en este caso indica Falso, y en el indicador 'Dato leído' muestra un arreglo de los Bytes recibidos. Para futuros envíos o recepción de datos por el puerto serie, solo se usará la parte de código que está dentro del cuadro de color negro.

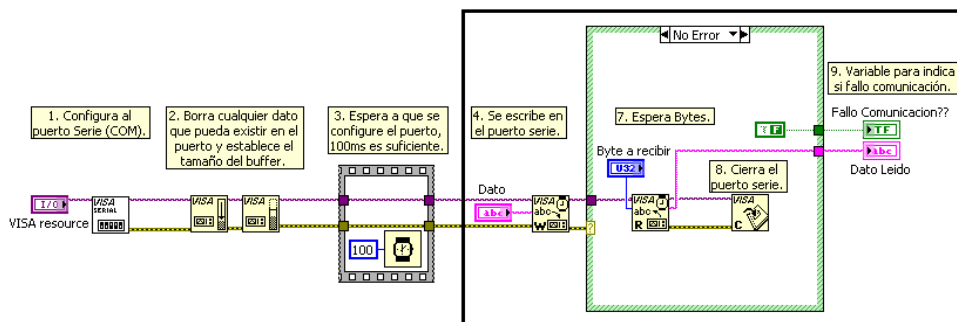


Figura 5.4 Diagrama de bloques para enviar/recibir datos por el puerto serie.

V.2.3 Comparación de datos, SubVI 3

Una vez que se han obtenido las variables que se han enviado al μC y este ha contestado, se comparan para saber si todo está correcto. Para saberlo se desarrolló un tercer subVI que se muestra en la figura 5.5. Lo que hace este código es tomar el comando que se envió y compararlo con el que se recibió. Si es el mismo le asigna el nuevo valor al control del motor que se había manipulado, de no ser el mismo, se deja el valor que se tenía antes de enviar los datos por el puerto serie.

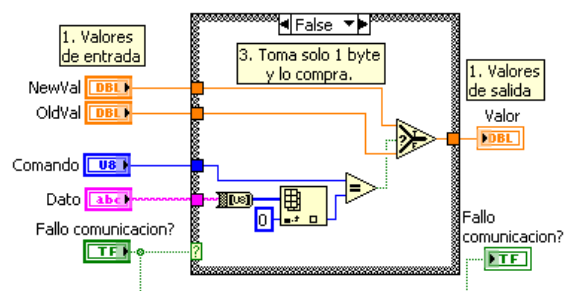


Figura 5.5 Diagrama de bloques para comparar comandos.

Lo anterior se explica de la siguiente manera: regresando al ejemplo de la figura 4.7, cuando se manipula el control del motor para que pase a la posición B (250), lo primero que hace la interfaz de usuario es calcular las variables (figura 5.3) para después mandarlas por el puerto serie (figura 5.4), una vez que el μC contesta se compara lo que se envió con lo que se recibió por el puerto serie (figura 5.5), si todo salió bien, entonces el control de la figura 4.7 marcará que ha pasado a la posición B, de ocurrir algún error (ya sea por conexión del puerto serie o algún otro), el control de la figura 4.7 se quedará en la posición A (0).

V.2.4 Integración de los SubVI's

En la figura 5.6 se muestra el diagrama de bloques que resulta de unir todos los subVI's anteriores, se tiene la misma estructura para cada uno de los 8 motores disponibles, lo único que cambia es el comando que corresponde al motor a manipular.

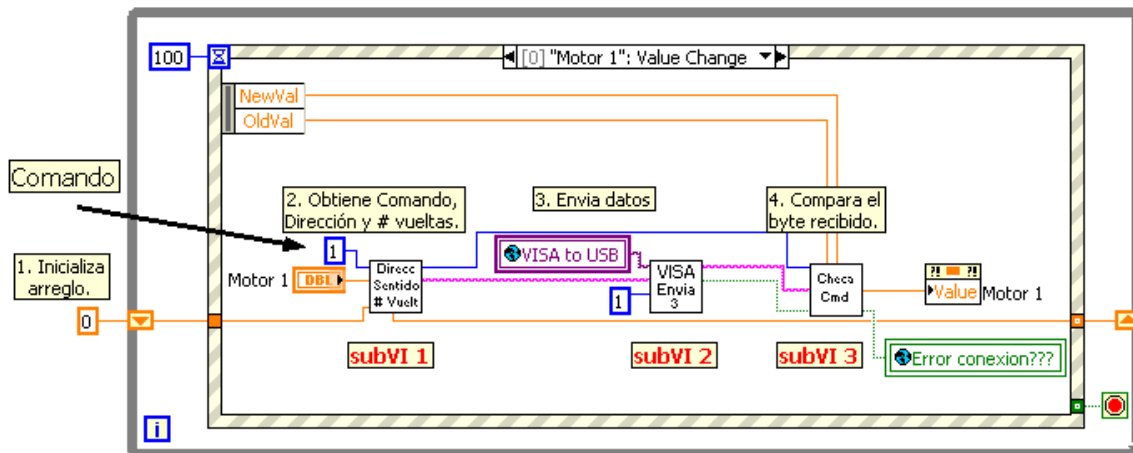


Figura 5.6 Diagrama de bloques para el control de los motores.

Por último, en la figura 5.7 se muestra el diagrama a bloques que se uso para controlar la intensidad de la lámpara de LED's, se recuerda que esta lámpara se controla por medio de PWM. Para realizar esta operación se cuenta con un control de tipo 'Meter' de punto flotante, este control se puede variar de 0 a 100, y está indicado con el número 3 de la figura 5.1.

El diagrama de bloques de la figura 5.7 sigue la misma estructura de la figura 5.6. Lo primero que se hace es ajustar el valor que será enviado para que el μC lo interprete como el Ciclo de Trabajo de la señal PWM, y lo ajusta multiplicando el valor del control por 10.23, se recuerda que el ciclo de trabajo va de 0 a 1023. Este dato se envía por el puerto serie junto con el comando correspondiente y ejecuta los subVI 2 y 3 descritos anteriormente.

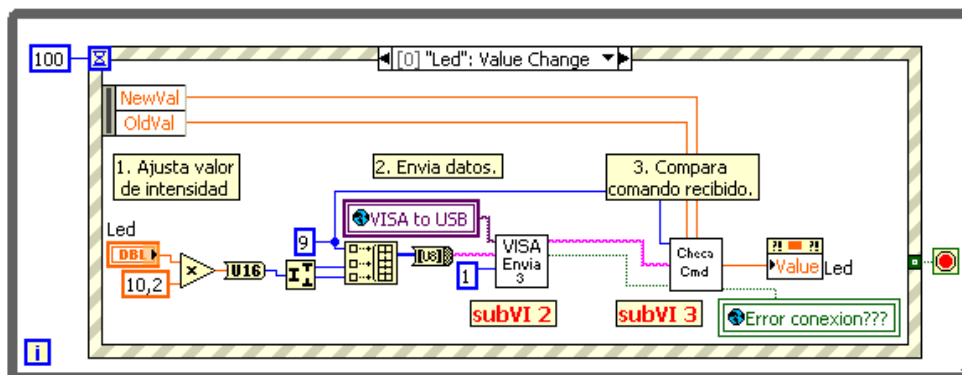


Figura 5.7 Diagrama de bloques para el control de la iluminación.



Hasta ahora se ha expuesto el hardware y software que se desarrolló para lograr un posicionamiento coordinado del microscopio electrónico sobre la muestra de cualquier tipo de material, en este momento ya se puede realizar un barrido de dicha muestra para poder tomar las micrografías correspondientes. Como se mencionó en el primer capítulo, el propósito de la mesa junto con el sistema electrónico de posicionamiento es poder tomar las micrografías de la muestra, para después manipularlas y poder obtener información acerca del comportamiento estructural del material a estudiar. Esta información estructural puede ser, entre otras; tamaño, forma y ubicación del grano dentro de la muestra.

Atendiendo a la necesidad de contar con un software con el que se pueda manipular éstas micrografías para obtener la ubicación del grano dentro de la muestra, se ha desarrollado una aplicación adicional a la interfaz de usuario para el control de motores, la cual se explica a continuación. Se mostrarán las ventanas principales y se describirá su funcionamiento.

V.3. PROGRAMA PARA MANIPULACIÓN DE IMÁGENES

En la figura 5.8 se muestra una micrografía de un material policristalino. Para conocer el comportamiento estructural de este material se necesita conocer la ubicación de los granos dentro de la muestra, es decir; se necesita conocer las coordenadas de la frontera de cada grano.

El objetivo consiste en manipular la imagen para poder obtener las coordenadas dentro de la ésta. En la figura 5.8 se puede ver esta manipulación y su funcionamiento es el siguiente:

1. Se debe localizar la imagen mediante el control marcado con el número ②.
2. Una vez que se ha visualizado la imagen (①), se debe seleccionar el color y tamaño del punto, después se activa la función de lápiz que aparece en el número ③.
3. Para recolectar las coordenadas sólo basta con dar un click izquierdo sobre la frontera del grano, cada vez que se da un click se agrega un punto de color sobre la imagen y la coordenada se adjuntará en el indicador marcado con el número ⑤.
En el punto número ④, aparecen 5 botones con los cuales se puede **eliminar** una coordenada no deseada o errónea, **graficar**, **guardar**, ó **limpiar** las coordenadas que aparecen en el número ⑤.
4. Una vez recolectadas todas las coordenadas como aparece en el número ①, se pueden graficar con el botón correspondiente, el resultado se mostrará en el punto ⑥.
5. Para guardar las coordenadas aparece una ventana como la mostrada en la figura 5.9a, donde se le asigna un nombre y la ubicación a guardar. Las coordenadas se guardarán con extensión txt, como se muestra en la figura 5.9b. El formato requerido con el cual se guardarán las coordenadas es " **{ x , y } , "**, dado que estos datos posteriormente pasarán a un programa de análisis matemático.

Con este programa se agiliza la forma de obtener las coordenadas de los granos haciendo que se le pueda dedicar más tiempo a la interpretación de los datos que a la recolección de los mismos.

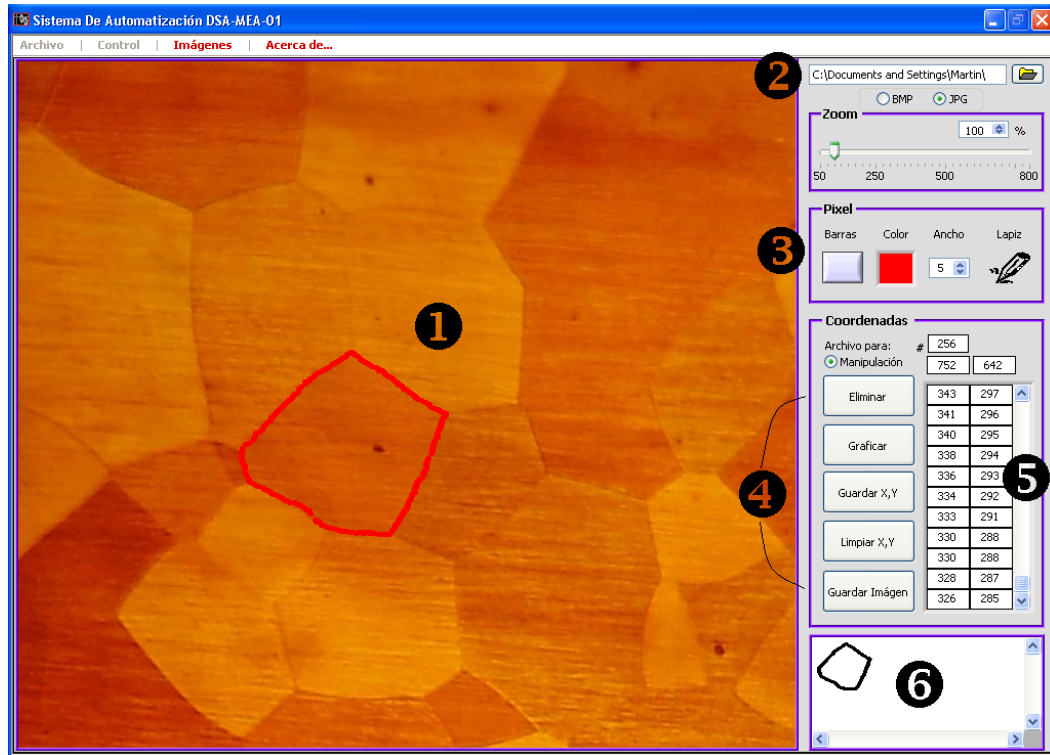
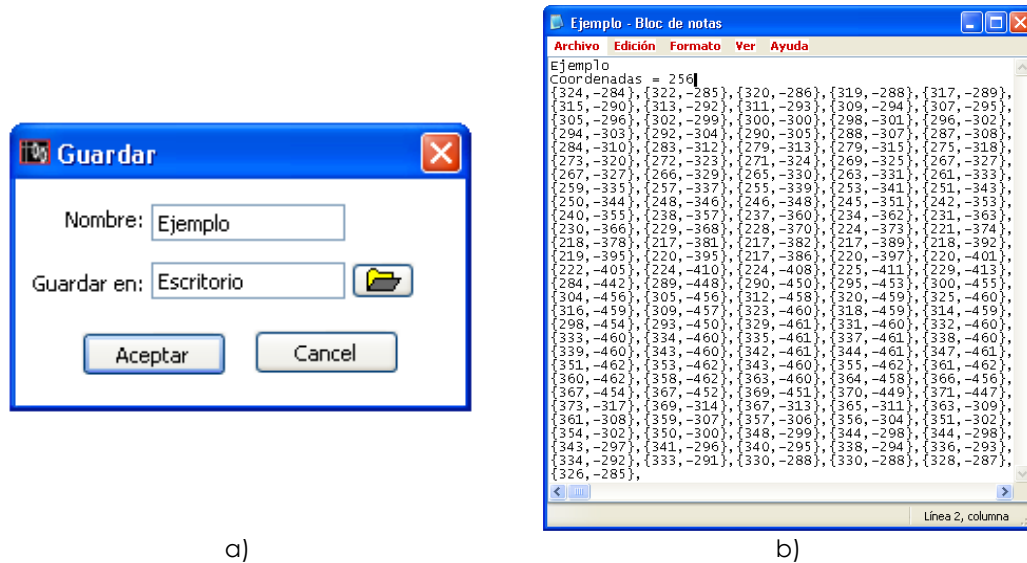


Figura 5.8 Software para manipulación de imágenes.



a) Ventana para guardar, b) Coordenadas guardadas.

A continuación se describe el desarrollo de un tercer programa de computadora, que surge como un complemento del segundo, éste programa no está planteado en los *objetivos* debido a que ésta aplicación surgió en tiempo de terminar la escritura del presente trabajo, dado lo anterior y como la teoría difiere de los objetivos planteados, sólo se describirán las necesidades que originaron su desarrollo y las características principales de dicho programa.

Actualmente se realiza un estudio que consiste en realizar pruebas experimentales de transmisión de carga de compresión en anillos plásticos confinados en medios elásticos de diferente rigidez.

Hasta ahora, la forma para determinar el comportamiento del anillo es mediante la instrumentación de galgas extensiométricas (*strain gauges*) en los anillos, estas galgas miden la deformación que sufre el material, sin embargo, el proceso de instrumentación es complicado y lento. Para determinar con precisión el comportamiento del anillo se deben realizar estudios teórico-experimentales más complejos que incluyan la aplicación de técnicas experimentales más eficientes y novedosas, como por ejemplo el análisis de imágenes, la cual se describe brevemente.

En la figura 5.10 se muestran dos fotografías de la prueba realizada, para este caso el material de confinamiento fue gravado con un mallado cuadrado. Durante la prueba, se toma una fotografía al inicio (figura 5.10a) y otra después de que se ha aplicado la carga máxima (figura 5.10b). Luego, para cada fotografía se obtienen las coordenadas de cada vértice de la malla, con estas coordenadas se puede obtener el desplazamiento que sufrió cada punto después de aplicar la carga. Con el procedimiento anterior, se obtienen campos de desplazamientos discretos que al ser ajustados permiten determinar el campo de deformaciones y conociendo las propiedades mecánicas del material, se puede determinar el **campo de esfuerzos**.

Calculando el esfuerzo radial en la interfaz entre el material de confinamiento y el anillo, se tendrá una estimación muy cercana de la presión que actúa sobre el anillo plástico, es decir, se puede conocer la carga que se aplica sobre el anillo. Conociendo esto es posible calcular las deformaciones tangenciales en la pared interior, la cual puede ser comparada con la que se mide con los extensómetros colocados en las muestras ensayadas.

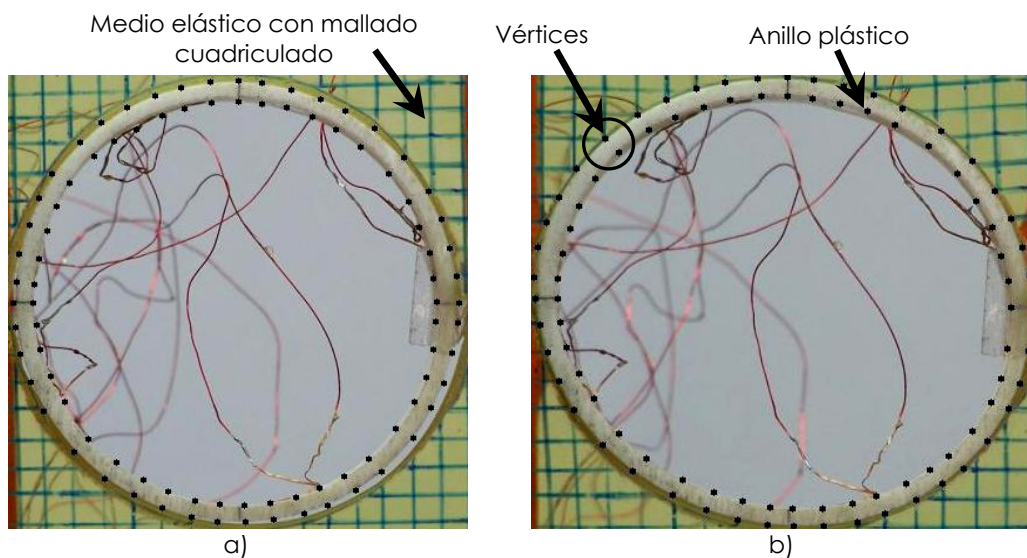


Figura 5.10 Anillo plástico: a) Sin carga, b) Con carga.

En la figura 5.11, se muestra un ejemplo del análisis descrito en los párrafos anteriores para el caso del anillo. Este análisis se realizó empleando un programa de computadora llamado **"FLOW MAP"** el cual funciona mediante el procesamiento digital de imágenes, dicho programa aplica un algoritmo que "filtra" las componentes de desplazamiento rígido y rotaciones rígidas mostradas en la siguiente figura.

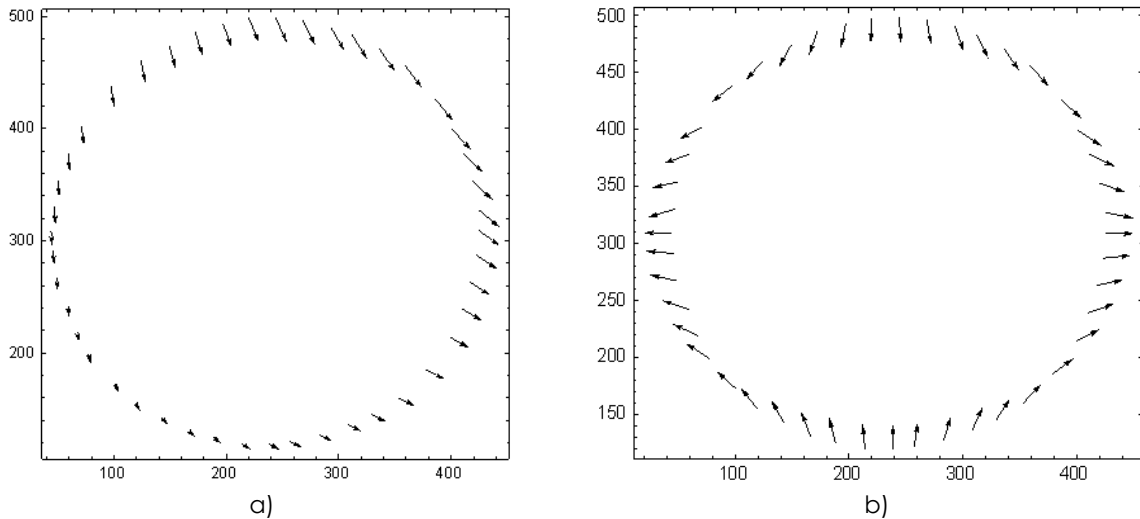


Figura 5.11 a) Campo de desplazamientos, b) Campo de desplazamientos sin las componentes de desplazamiento rígidas y rotación rígidas.

Es claro que aplicando este tipo de metodologías, es posible determinar campos de desplazamientos discretos que pueden ser sometidos a un ajuste no lineal que proporcione los campos de desplazamientos continuos que se necesitan para obtener la información sobre la forma en que se transmite la carga de compresión a través del medio elástico al anillo plástico.

Sin embargo, emplear el software FLOW MAP ha resultado un tanto complicado debido a que fue diseñado para determinar campos de velocidades en flujos con partículas reflejantes. Se ha observado que en el caso de sólidos, dicha condición se puede obtener proporcionando a la muestra cierta "textura óptica" que puede lograrse aplicando un atomizado muy fino de tinta oscura o bien iluminando la muestra con un láser multipuntual. Además de las dificultades que se tienen con el empleo del software FLOW MAP, se agrega la complejidad en su manejo y que no es de fácil adquisición por tratarse de software especializado. En atención a esta necesidad y ya que se cuenta con un programa en el cual se puede manipular imágenes, se plantea la solución a dicha necesidad con el desarrollo de otro programa con el que se pueda tener un valor aproximado del campo de desplazamiento, éste se desarrolla como un complemento del "Programa para Manipulación de Imágenes".

V.4. PROGRAMA PARA COMPARACIÓN DE IMÁGENES

Aprovechando que el microscopio de la mesa de posicionamiento puede tomar fotografías con acercamientos de 1x, es decir, sin aumento, se puede usar para la toma de las fotografías como se muestran en la figura 5.10. A éstas fotografías se les ha manipulado para obtener las coordenadas de la frontera del anillo con el Programa de Manipulación de Imágenes, en dicha figura se puede observar que las coordenadas que se obtuvieron son las mismas de los puntos que aparecen en los vértices del mallado de estas imágenes.

Una vez tomadas y almacenadas las coordenadas de los puntos de cada una de las imágenes de la figura 5.10, se debe realizar una comparación de éstas para obtener el campo de desplazamiento y así poder determinar el comportamiento del anillo cuando se somete a una carga.

El Programa de Comparación de Imágenes toma las coordenadas del anillo, antes y después de haber sido cargado, para posteriormente compararlas. La ventana principal se puede ver en la figura 5.12. Una vez que se ubica a los archivos donde se encuentran las coordenadas previamente guardadas con los controles etiquetados con los números ❶ y ❷ de ésta figura, se puede graficar pulsando el botón indicado con el número ❸.

En el cuadro etiquetado con el número ❺ se grafican las coordenadas obtenidas, y en el cuadro indicado con el número ❻ se grafica el campo de desplazamiento que presentó el anillo al ser sometido a la carga. Adicionalmente, se pueden guardar estas imágenes en formato JPG pulsando el botón etiquetado con el número ❹, donde aparecerá una ventana como la de la figura 5.9a.

Este programa muestra un aproximado del campo de desplazamiento, también es importante mencionar que la imagen que aparece etiquetada con el número ❻ no es exacta, ya que la naturaleza de las fotografías no lo permite, dichas fotografías no cuadran en origen, tamaño y posición, una con respecto a otra, sin embargo; al comparar ésta imagen con la mostrada en la figura 5.11a se puede decir que el resultado es aceptable y cumple con el objetivo de la aplicación.

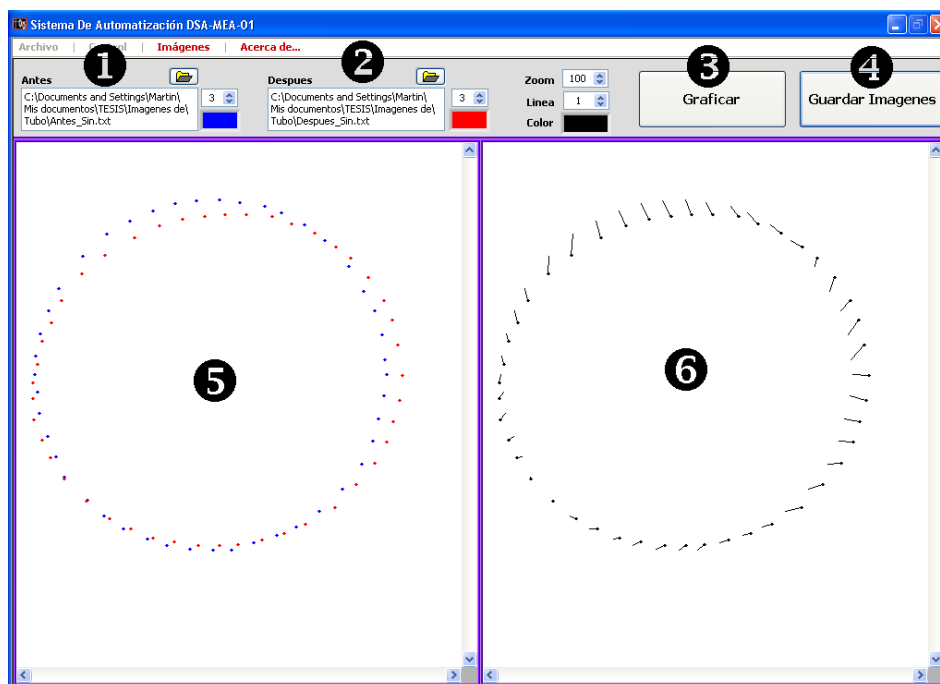


Figura 5.12 Software de Comparación de Imágenes.

Se realizaron pruebas al Programa de Comparación de Imágenes y se llegó a la conclusión de que para mejorar el campo de desplazamiento se deben tomar las fotografías de manera que el centro de ambas sea el mismo. Actualmente se encuentra en etapa de diseño un equipo con el que se pueda tomar fotografías que correspondan en origen, tamaño y posición del anillo. El Programa de Comparación de Imágenes está diseñado para adicionar futuras mejoras.

Capítulo **6**

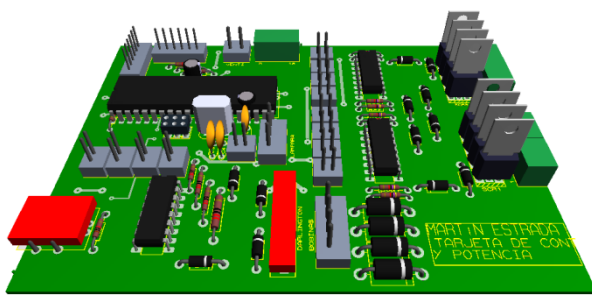
Pruebas y Conclusiones

Objetivo particular. En este último capítulo se presenta:

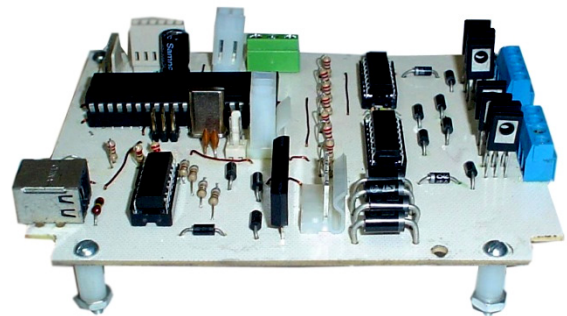
- El diseño y desarrollo de los circuitos impresos.
- El montaje de las tarjetas dentro de un chasis.
- La integración del sistema de control al Equipo Mecánico-Óptico y su puesta a punto.
- Las pruebas realizadas y sus correspondientes conclusiones.

VI.1 INTEGRACIÓN DEL SISTEMA

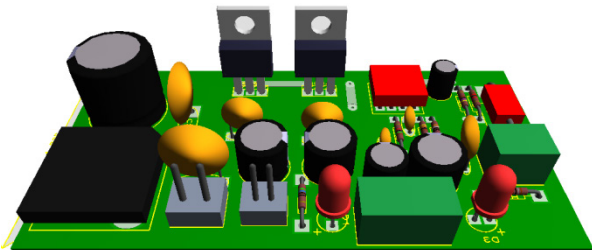
Para la integración del sistema se han diseñado y desarrollado los circuitos impresos (PCB) del hardware usando el software PROTEUS VSM y se muestran en la figura 6.1. En las figuras a y c se muestra el diseño electrónico de las tarjetas de control y de la fuente, respectivamente; en los incisos b y d se muestran la construcción de las mismas.



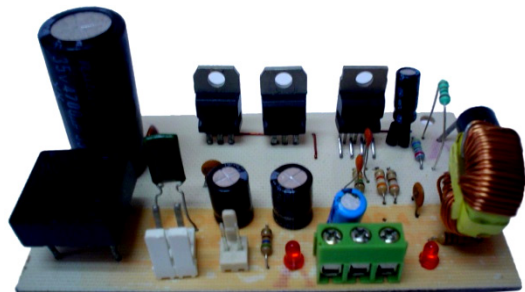
a)



b)



c)



d)

Figura 6.1 Diseño de los circuitos impresos.

Las tarjetas de la figura anterior fueron incorporadas a un chasis para brindar protección al sistema, en la figura 6.2 se muestra la conexión e integración de éste.

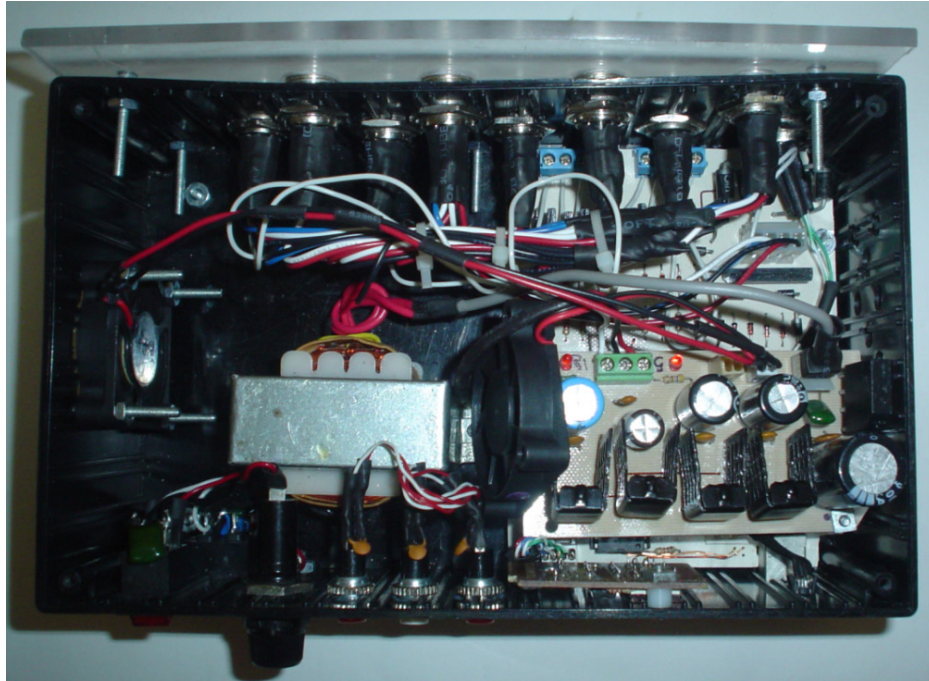


Figura 6.2 Integración de las tarjetas dentro del chasis.

Una vez integradas las tarjetas, se le ha asignado el nombre de “**Sistema de Automatización**”, esto con el fin de tener una mejor presentación. Las imágenes del Sistema de Automatización se muestran en la figura 6.3.



Figura 6.3 Sistema de Automatización.

En cuanto al software desarrollado, se nombró igual que al hardware, y con un recurso de LabVIEW se creó el archivo de instalación. Así, al contar con este archivo, este software se puede instalar en cualquier equipo y ejecutarse sin la necesidad de tener el programa LabVIEW. Una vez que se instala el programa, como cualquier otro, se puede localizar en el menú inicio como se muestra la figura 6.4.

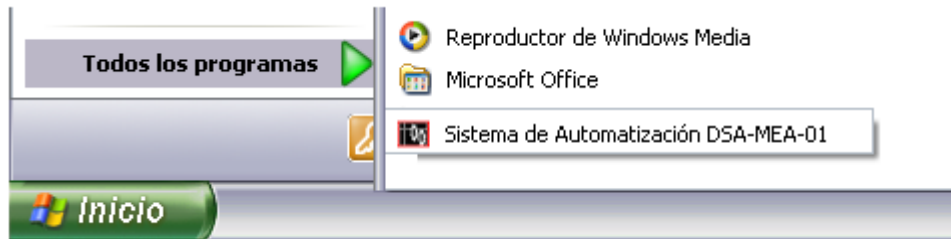


Figura 6.4 Localización del Software para el Sistema de Automatización una vez instalado.

En la figura 6.5 se muestra el panel principal del Sistema de Automatización.



Figura 6.5 Panel principal del Sistema de Automatización al ejecutarse.

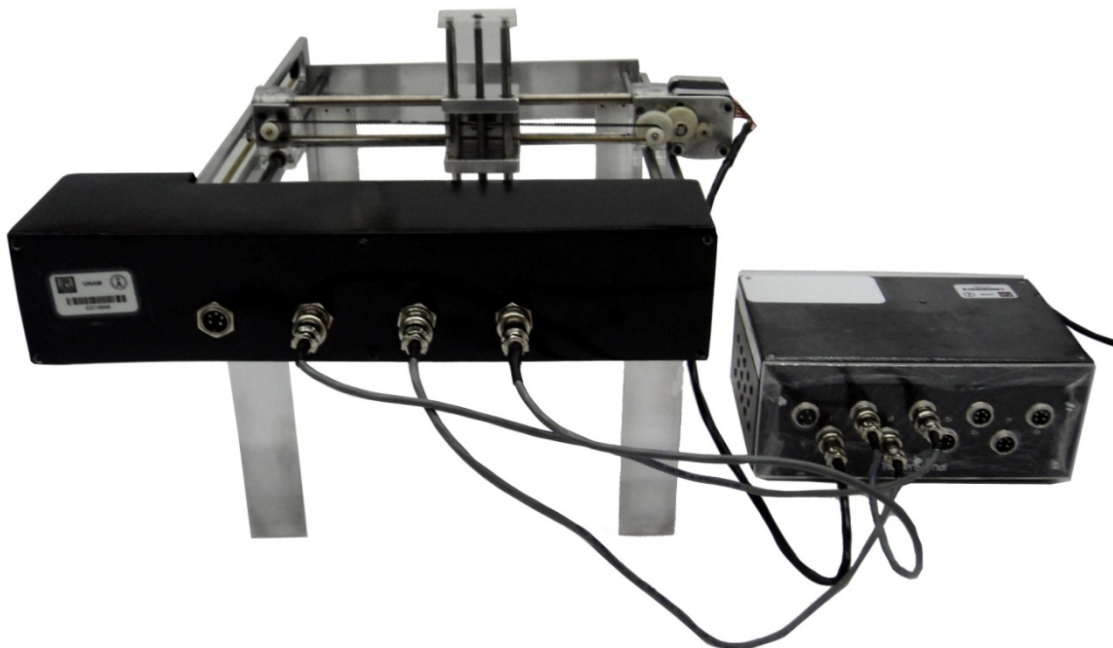
Una vez integrado todo el sistema, tanto hardware como software, se pueden realizar las pruebas pertinentes.

VI.2 PRUEBAS AL SISTEMA

En la figura 6.5 se muestran las imágenes donde se incorpora el Sistema de Automatización al equipo Mecánico-Óptico para las pruebas correspondientes obteniendo los siguientes resultados.



a)



b)

Figura 6.5 Pruebas al sistema.

VI.2.1 Resultados del hardware

- La fuente genera el voltaje y la corriente necesaria para el funcionamiento de los motores, con esto, los motores generan el torque suficiente para mover a su eje correspondiente.
- El código del microcontrolador funciona satisfactoriamente, esto significa que las secuencias de control para los giros a la derecha y a la izquierda, el sistema de multiplexaje, los botones e indicadores, así como la comunicación USB funcionan correctamente.
- El desplazamiento lineal de cada eje corresponde al calculado. La velocidad de desplazamiento es aceptable.
- El Sistema de Automatización presenta gran versatilidad al contar con dos modos de control.

VI.2.1 Resultados del software

- Los datos que procesa y manda la Interfaz de Usuario al microcontrolador son los indicados, permitiendo que con estos datos el posicionamiento sea coordinado preciso y exacto, lo que da como consecuencia un buen control sobre el equipo Mecánico-Óptico.
- Se realizaron pruebas con el Software para Manipulación de Imágenes y se compararon con datos previamente verificados. El resultado fue prácticamente el mismo en ambos casos.
- Las pruebas realizadas al Software de Comparación de Imágenes mostraron resultados aceptables.

VI.3 CONCLUSIONES

Tomando el objetivo planteado en este proyecto, se puede concluir que dicho objetivo ha sido cumplido. Se ha logrado complementar el estudio de los materiales implementando un Sistema de Automatización simple y eficiente que cuenta con:

- Una tarjeta electrónica que controla el doble de los motores planteados, requiriendo un mínimo de componentes.
- Una Interfaz de Usuario que precisa y agiliza la posición de los motores para la toma de las micrografías.
- Un Programa para la Manipulación de Imágenes que obtiene los datos requeridos de las micrografías y los almacena en un formato predefinido en la computadora.
- Un Programa que compara imágenes de forma simple y eficiente.

Es importante mencionar que el hardware desarrollado no sólo se puede aplicar al equipo Mecánico-Óptico, prácticamente se puede adaptar a cualquier estructura y/o mecanismo que involucre movimientos coordinados a través de motores de pasos unipolares, ya que durante el desarrollo del sistema se tuvo contemplada esta característica. Con respecto al software, igualmente se puede adaptar a cualquier proceso en el que se necesite la manipulación de imágenes para la obtención de coordenadas dentro de la misma.



Tal vez la mejor conclusión a la que se puede llegar, es la demostración de que en México se puede aplicar la ingeniería y/o tecnología necesaria para resolver los problemas en ciertos procesos, ya sea en la investigación o en la industrial, disminuyendo la dependencia de la tecnología extranjera.

También es importante mencionar que el hardware fue pensado para ser optimizado y/o mejorado en sus funciones sin tener que agregar o quitar dispositivos electrónicos a las tarjetas, solo se necesita editar el código del microcontrolador.

Este proyecto está sujeto a futuras mejoras del sistema, entre las que se deben destacar:

- Controlar más de un motor a la vez y aumentar la velocidad de giro de cada uno.
- Incorporar algún sensor que pueda indicar la posición de cada eje.
- Implementar un algoritmo dentro del código del microcontrolador y de la Interfaz de Usuario para la detección de errores, por mencionar alguno, recordar la posición de los motores cuando por alguna razón inesperada se desenergiza el sistema.
- Complementar el protocolo de comunicación, es decir, implementar USB en lugar de USB emulado.
- Visualizar en la pantalla de la Interfaz de Usuario la visión que tiene el microscopio cuando se encuentra en funcionamiento.
- Para el Programa de Manipulación de Imágenes se puede mejorar la forma de obtener las coordenadas de forma automática, es decir, a través del procesamiento digital de la imagen.
- Incorporar el procesamiento matemático que se requiere para el procesamiento de las coordenadas de las micrografías, así, una vez que se tienen los datos que se procesen en ese instante.
- Complementar y afinar el funcionamiento del Programa de Comparación de Imágenes.

Por último, personalmente, me encuentro muy satisfecho con la realización de este proyecto, ya que me permitió complementar mi formación profesional poniendo en práctica todos los conocimientos y habilidades adquiridas durante la carrera, además me ha brindado la oportunidad de interactuar con compañeros de una área distinta a la mía, aportándome nuevos conocimientos e ideas.

Bibliografía

Libros

- William D. Callister, Jr. Introducción a la ciencia e ingeniería de los materiales. Editorial Reverté. S.A.
- Donald R. Askeland. Ciencia e ingeniería de los materiales. International Thomson Editores. Tercera edición 1998.
- William F. Smith. Ciencia e ingeniería de los materiales. McGraw-Hill. Tercera edición 2004.
- Eduardo García briego. Compilador C CCS y simulador Proteus para microcontroladores PIC. Alfaomega. Primera edición 2008.
- José R. LajaraVizcaíno. José Pelegrí Sebastián. LabVIEW entorno gráfico de programación. Alfaomega marcombo, Primera edición 2007.
- Robert L. Boylestad, Louis Nashelsky. Electrónica: teoría de circuitos y dispositivos electrónicos. Pearson education, Octava edición.
- Robert L. Boylestad. Introducción al análisis de circuitos. Prentice Hall. Decima edición 2004.
- Albert P. Malvino. Principios de electrónica. McGraw-Hill. Sexta edición 1999.
- José M. Angulo Usategui, Susana R. Yesa, Ignacio Angulo Martínez. Microcontroladores PIC - Diseño Práctico de Aplicaciones 2da Parte 16F87x, segunda parte. McGraw-Hill. Primera edición.
- Enrique palacios, Fernando Remiro, Lucas J. López. Microcontrolador PIC16F84, desarrollo de proyectos. Alfaomega. Primera edición 2004.

Páginas de Internet

<http://es.wikipedia.org/wiki/Micrograf%C3%ADa>

<http://www.usb.org/home>

http://en.wikipedia.org/wiki/Universal_Serial_Bus<http://es.wikipedia.org/wiki/RS-232>

<http://www.labcenter.com/index.cfm>

<http://www.labcenter.com/products/pic18.cfm>

<http://www.ni.com/support/>

<http://www.microchip.com/search/searchapp/searchhome.aspx?id=2&q=usb>

<http://www.ccsinfo.com/content.php?page=compilers>

<http://www.microcontroladorespic.com/>

http://robots-argentina.com.ar/MotorPP_basico.htm



<http://www.x-robotics.com/motorizacion.htm>

<http://www.directindustry.es/fabricante-industrial/mesa-posicionamiento-62023.html>

Artículos y Publicaciones

- J. Cortés-Pérez, F. P. Delgado G., F. N. García Castillo, R. Cornejo T. y J. Raigosa V. Transmisión descarga a través de un medio elástico a un anillo circular confinado, Congreso anual de la SOMIM 2010.
- Euiti Eibar . Introducción a la electrónica de potencia.
- Christopher G. Relf. Image Acquisition and Processing with LabVIEW. National Instruments.

Apéndice

Código del Microcontrolador

```
#include <18F4550.h>

#fuses HSPLL, NOWDT, NOPROTECT, NOLVP, NODEBUG, USBDIV, PLL5, CPUDIV1, VREGEN
#use delay(clock=48000000) //configure a 20MHz crystal to operate at 48MHz

#define USB_CON_SENSE_PIN PIN_C6
#include <usb_cdc.h>

#use standard_io(A)
#use fixed_io(B_outputs=PIN_B4, PIN_B5, PIN_B6, PIN_B7)
#use standard_io(C)
#use standard_io(D)
#use fixed_io(E_outputs=PIN_E0, PIN_E1, PIN_E2)

    int8 Comando, Direccion, i, CualMotor, Motor1, PasoDer, PasoIzq, Hi, Lo;
    int16 Contador, Vueltas;

//////////////////////////////////////
//XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
//XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
//XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
//////////////////////////////////////

// Derecha 4 Pasos -----
void Derecha_4_Pasos() {
    Motor1 = 0b01001100;
    for (i=0; i<=3; ++i) {
        OUTPUT_A(Motor1); // 0b 0100 1100
        rotate_right( &Motor1, 1); // 0b 0010 0110
        delay_ms(2); // 0b 1000 1001
    } // 0b 1100 0100
    OUTPUT_A(0x00);
    delay_ms(2);
}

// Izquierda 4 Pasos -----
void Izquierda_4_Pasos() {
// BYTE i, Motor1;
    Motor1 = 0b00100011;
    for (i=0; i<=3; ++i) {
        OUTPUT_A(Motor1); // 0b 0010 0011
        rotate_left( &Motor1, 1); // 0b 1000 1100
        delay_ms(2); // 0b 0001 1001
    } // 0b 1100 0100
    OUTPUT_A(0x00);
    delay_ms(2);
}

// DecodCualMotor -----
void DecodCualMotor() {
    switch (CualMotor) {
        case 0: output_b(0xCF);
            break;
    }
}
```



```

//////////////////////////////////////////////////////////////////// CONFIGURACION DE INT EXT RB0-2 ////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////
port_b_pullups(FALSE); // Deshabilita Resistencias de pull-up

ext_int_edge( 0, H_TO_L); // Flanco de caida para RB0
ext_int_edge( 1, L_TO_H); // Flanco de subida para RB1
ext_int_edge( 2, H_TO_L); // Flanco de caida para RB2

clear_interrupt(int_ext);
clear_interrupt(int_ext1);
clear_interrupt(int_ext2);

enable_interrupts(int_ext); // Habilita INT EXT RB0
enable_interrupts(int_ext1); // Habilita INT EXT RB1
enable_interrupts(int_ext2); // Habilita INT EXT RB2

enable_interrupts(global); // Habilitacion global de interrupciones

//////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////// INICIALIZACIÓN DE USB, ESTA PARTE ES NECESARIA ////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////
usb_cdc_init();
usb_init();

// Configuración de PWM -----
setup_ccp1(CCP_PWM); // Configure CCP1 as a PWM
setup_timer_2(T2_DIV_BY_16, 255, 1);
set_pwm1_duty(1);

// Configuración de USB -----
while(!usb_cdc_connected()) {}
do {
    usb_task();
    if (usb_enumerated()) {
        output_e(0b00000011); // Si se conecta, enciende Led Azul-Rojo
        output_b(0x0F);
    }
}

// Deshabilita interrupciones -----
disable_interrupts(INT_EXT); // Habilita INT EXT RB0
disable_interrupts(INT_EXT1); // Habilita INT EXT RB1
disable_interrupts(INT_EXT2); // Habilita INT EXT RB2

//////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////// CÓDIGO PRINCIPAL DEL PROGRAMA ////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////
Comando = usb_cdc_getc(); // Obtiene comando de USB
switch(Comando){ // Evalua a este comando

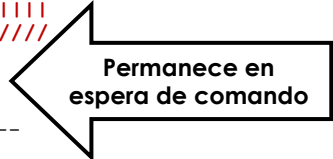
// NULO -----
case 0: OUTPUT_D(0x00);
        usb_cdc_putc(0x00);
        break;

// Motor 1 -----
case 1: output_high(PIN_D0); // Selecciona Motor 1
        OUTPUT_B(0x1F); // Indica 1 en display de 7 seg
        Direccion = usb_cdc_getc(); // Obtiene Direccion en Motor 1

        HI = usb_cdc_getc();
        LO = usb_cdc_getc();
        Vueltas = make16(HI,LO);

        switch(Direccion){ // Evalua la direccion para Motor 1
            case 0: for (Contador=1;Contador<=Vueltas;++Contador)
                    Izquierda_4_Pasos(); // Gira a la izq
                    break;
            case 1: for (Contador=1;Contador<=Vueltas;++Contador)
                    Derecha_4_Pasos(); // Gira a la der
                    break;
        }
        OUTPUT_D(0x00); // desactiva SCR
        OUTPUT_B(0x0F); // Indica 0 en display de 7 seg
        usb_cdc_putc(1); // regresa el comando recibido
}

```





```
break; //termina estructura case, regresa donde se encuentra la flecha

// Motor 2 -----
case 2:  output_high(PIN_D1);           // Selecciona Motor 2
        OUTPUT_B(0x2F);                // Indica 2 en display de 7 seg
        Direccion = usb_cdc_getc();    // Obtiene Direccion en Motor 2

        HI = usb_cdc_getc();
        LO = usb_cdc_getc();
        Vueltas = make16(HI,LO);

        switch(Direccion){              // Evalua la direccion para Motor 2
            case 0:  for (Contador=1;Contador<=Vueltas;++Contador)
                    Izquierda_4_Pasos(); // Gira a la izq
                    break;
            case 1:  for (Contador=1;Contador<=Vueltas;++Contador)
                    Derecha_4_Pasos();   // Gira a la der
                    break;
        }
        OUTPUT_D(0x00);                 // desactiva SCR
        OUTPUT_B(0x0F);                 // Indica 0 en display de 7 seg
        usb_cdc_putc(2);                 // regresa el comando recibido
        break; //termina estructura case, regresa donde se encuentra la flecha

// Motor 3 -----
case 3:  output_high(PIN_D2);           // Selecciona Motor 3
        OUTPUT_B(0x3F);                // Indica 3 en display de 7 seg
        Direccion = usb_cdc_getc();    // Obtiene Direccion en Motor 3

        HI = usb_cdc_getc();
        LO = usb_cdc_getc();
        Vueltas = make16(HI,LO);

        switch(Direccion){              // Evalua la direccion para Motor 3
            case 0:  for (Contador=1;Contador<=Vueltas;++Contador)
                    Izquierda_4_Pasos(); // Gira a la izq
                    break;
            case 1:  for (Contador=1;Contador<=Vueltas;++Contador)
                    Derecha_4_Pasos();   // Gira a la der
                    break;
        }
        OUTPUT_D(0x00);                 // desactiva SCR
        OUTPUT_B(0x0F);                 // Indica 0 en display de 7 seg
        usb_cdc_putc(3);                 // regresa el comando recibido
        break; //termina estructura case, regresa donde se encuentra la flecha

// Motor 4 -----
case 4:  output_high(PIN_D3);           // Selecciona Motor 4
        OUTPUT_B(0x4F);                // Indica 2 en display de 7 seg
        Direccion = usb_cdc_getc();    // Obtiene Direccion en Motor 4

        HI = usb_cdc_getc();
        LO = usb_cdc_getc();
        Vueltas = make16(HI,LO);

        switch(Direccion){              // Evalua la direccion para Motor 4
            case 0:  for (Contador=1;Contador<=Vueltas;++Contador)
                    Izquierda_4_Pasos(); // Gira a la izq
                    break;
            case 1:  for (Contador=1;Contador<=Vueltas;++Contador)
                    Derecha_4_Pasos();   // Gira a la der
                    break;
        }
        OUTPUT_D(0x00);                 // desactiva SCR
        OUTPUT_B(0x0F);                 // Indica 0 en display de 7 seg
        usb_cdc_putc(4);                 // regresa el comando recibido
        break; //termina estructura case, regresa donde se encuentra la flecha

// Motor 5 -----
case 5:  output_high(PIN_D4);           // Selecciona Motor 5
        OUTPUT_B(0x5F);                // Indica 5 en display de 7 seg
        Direccion = usb_cdc_getc();    // Obtiene Direccion en Motor 5
```

```

HI = usb_cdc_getc();
LO = usb_cdc_getc();
Vueltas = make16(HI,LO);

switch(Direccion){
    // Evalua la direccion para Motor 5
    case 0:    for (Contador=1;Contador<=Vueltas;++Contador)
                Izquierda_4_Pasos();    // Gira a la izq
                break;
    case 1:    for (Contador=1;Contador<=Vueltas;++Contador)
                Derecha_4_Pasos();      // Gira a la der
                break;
}
OUTPUT_D(0x00);          // desactiva SCR
OUTPUT_B(0x0F);         // Indica 0 en display de 7 seg
usb_cdc_putc(5);        // regresa el comando recibido
break; //termina estructura case, regresa donde se encuentra la flecha

// Motor 6 -----
case 6:    output_high(PIN_D5);          // Selecciona Motor 6
            OUTPUT_B(0x6F);             // Indica 6 en display de 7 seg
            Direccion = usb_cdc_getc();  // Obtiene Direccion en Motor 6

            HI = usb_cdc_getc();
            LO = usb_cdc_getc();
            Vueltas = make16(HI,LO);

            switch(Direccion){
                // Evalua la direccion para Motor 6
                case 0:    for (Contador=1;Contador<=Vueltas;++Contador)
                            Izquierda_4_Pasos();    // Gira a la izq
                            break;
                case 1:    for (Contador=1;Contador<=Vueltas;++Contador)
                            Derecha_4_Pasos();      // Gira a la der
                            break;
            }
            OUTPUT_D(0x00);          // desactiva SCR
            OUTPUT_B(0x0F);         // Indica 0 en display de 7 seg
            usb_cdc_putc(6);        // regresa el comando recibido
            break; //termina estructura case, regresa donde se encuentra la flecha

// Motor 7 -----
case 7:    output_high(PIN_D6);          // Selecciona Motor 7
            OUTPUT_B(0x7F);             // Indica 7 en display de 7 seg
            Direccion = usb_cdc_getc();  // Obtiene Direccion en Motor 7

            HI = usb_cdc_getc();
            LO = usb_cdc_getc();
            Vueltas = make16(HI,LO);

            switch(Direccion){
                // Evalua la direccion para Motor 7
                case 0:    for (Contador=1;Contador<=Vueltas;++Contador)
                            Izquierda_4_Pasos();    // Gira a la izq
                            break;
                case 1:    for (Contador=1;Contador<=Vueltas;++Contador)
                            Derecha_4_Pasos();      // Gira a la der
                            break;
            }
            OUTPUT_D(0x00);          // desactiva SCR
            OUTPUT_B(0x0F);         // Indica 0 en display de 7 seg
            usb_cdc_putc(7);        // regresa el comando recibido
            break; //termina estructura case, regresa donde se encuentra la flecha

// Motor 8 -----
case 8:    output_high(PIN_D7);          // Selecciona Motor 8
            OUTPUT_B(0x8F);             // Indica 8 en display de 7 seg
            Direccion = usb_cdc_getc();  // Obtiene Direccion en Motor 8

            HI = usb_cdc_getc();
            LO = usb_cdc_getc();
            Vueltas = make16(HI,LO);

            switch(Direccion){
                // Evalua la direccion para Motor 8
                case 0:    for (Contador=1;Contador<=Vueltas;++Contador)

```



```
        Izquierda_4_Pasos();    // Gira a la izq
        break;
    case 1:    for (Contador=1;Contador<=Vueltas;++Contador)
                Derecha_4_Pasos();    // Gira a la der
        break;
    }
    OUTPUT_D(0x00);                // desactiva SCR
    OUTPUT_B(0x0F);                // Indica 0 en display de 7 seg
    usb_cdc_putc(8);                // regresa el comando recibido
    break;    //termina estructura case, regresa donde se encuentra la flecha

// Control de iluminación -----
    case 9:    OUTPUT_B(0xCF);

                HI = usb_cdc_getc();
                LO = usb_cdc_getc();
                Vueltas = make16(HI,LO);

                set_pwm1_duty(Vueltas);    //coloca en Byte recibido en el ciclo de trabajo
                usb_cdc_putc(9);
                break;

// Inicia -----
    case 10:    usb_cdc_putc(79);                // 79 Dec == 4F Hex == 'O' en ascii
                usb_cdc_putc(107);            // 107 Dec == 6B Hex == 'k' en ascii
                break;

// Default -----
    default:    OUTPUT_D(0x00);
                usb_cdc_putc(0xff);
                break;

// Aqui termina el case //////////////////////////////////////
    } //Case principal
////////////////////////////////////
// Todo el programa debe ir dentro de de este {if}
} //if
////////////////////////////////////\////////////////////////////////////
// Si no enumera el HUB a el PIC, se apaga el led
else
    output_e(0b00000000);    // No enumerado, apaga el LED
////////////////////////////////////\////////////////////////////////////
// Fin de while y de programa Principal
} while (TRUE);
} // main
```

Apéndice **B**

Hojas de Especificaciones

- Microscopio AM413
- Motor de pasos 103H5210-0440
- Microcontrolador PIC18F4550
- Transistor Darlington STA401A
- Regulador de voltaje conmutado L4960
- Regulador de voltaje L7805
- Optoacoplado LTV-817
- Rectificador controlado de silicio c106-D
- Diodo rectificador STPS5L40
- Diodo Rectificador RL203

Dino-Lite Series

Portable, handheld Dino-Lite Digital Microscope / Camera



Model : AM413MT (Aluminum Alloy Housing)

- Handheld Metal Shell Digital Microscope
- 1.3M / Resolution 1280x1024
- 10~50X,200X Continuous Magnification
- Achieves effective magnification values of up to 330x w/ digital zoom (only available in VGA mode - 640*480)
- USB 2.0 Output
- LED on/off controlled by software
- Measurement with Calibration



Description			
Product Number	AM-413M	Magnification	10~50X, 200X Continuous (zoom in option - 330x VGA mode only)
Warranty Period	1 Year	Pixels	1.3M
Condition	New	Resolution	1280x1024
Packaging	Retail Box	Data Output	USB 2.0
Tech Support	Dealer or BigC.com	Sensor	1/3" Color CMOS
Regulatory Approval	CE, FCC, ROHS	Country of Origin	Taiwan
Unit Weight	132 gm	Unit Weight w/package	1.4 lbs
Unit Dimension	4" (H) x 1.26" (D)	Unit Box Dimension	8.5"x7"x 2.75"

Product Applications

AM413MT model to it's line of digital microscopes. The aerospace-grade machine tooled aluminum housing offers enhanced protection and durability even in the harshest working environments and the precision machined construction allows for smoother adjustment and operation and focusing. The AM413MT is the latest addition to the flagship "X" series of Dino-Lite digital microscopes offering true 1.3 mega pixel high resolution imaging. Model AM413MT features a specially designed removable nozzle with built in light diffuser to reduce glare as well as a spare standard nozzle for high light examination when required. The stylish and durable polished aluminum body and performance upgrades makes the AM413MT the first choice of professionals.

AM413MT has "Microtouch" shutter trigger allowing for convenient image capture capability while in hand-held mode OR rigid mount mode trigger feature is best when use in hand held mode.

Ideal for a wide variety of practical applications in the classroom, workplace and even the home. Can be used for science and engineering work and study, dermatology (skin/scalp) exam, detailed repair, assembly and quality control (i.e. electronics, mechanical, etc), hobbies & collecting (coin, stamp, watch, jewelry, etc.), law enforcement (counterfeit ID, crime lab, etc), entertainment or simply as a reading aid. Just use your imagination!



2-phase Stepping Motor

42mm sq.
(1.65inch sq.)

103H52 □□
1.8°/step

Recommendable Driver
Refer to the page 7,17,27 and 45.

Specifications

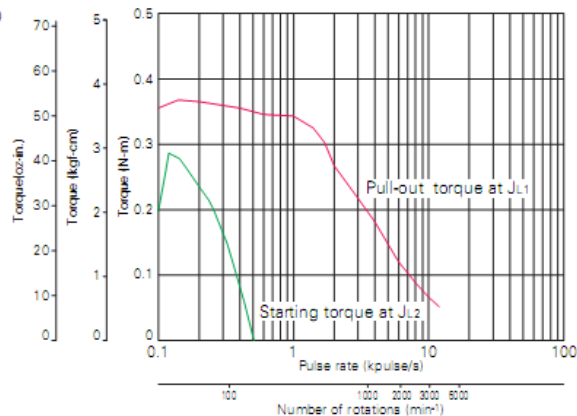
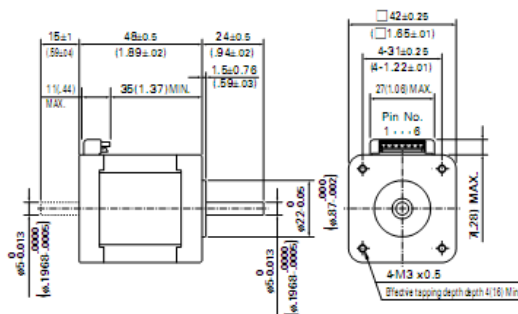
Unipolar winding

Model		Holding torque at 2-phase energization	Rated current	Resistance	Inductance	Rotor inertia	Mass(Weight)
Single shaft	Double shaft	N·m (oz·in)MIN.	A/phase	Ω/phase	mH/phase	$\times 10^{-4}$ kg·m ² (oz·in ²)	kg(lbs)
103H5205-0440	-0410	0.2(28.32)	1.2	2.4	2.3	0.036(0.20)	0.23(0.51)
103H5208-0440	-0410	0.3(42.48)	1.2	2.9	3.4	0.056(0.31)	0.29(0.64)
103H5209-0440	-0410	0.32(45.31)	1.2	3	3.9	0.062(0.34)	0.31(0.68)
103H5210-0440	-0410	0.37(52.39)	1.2	3.3	3.4	0.074(0.40)	0.37(0.82)

● 103H5210-0440

103H5210-0440 (Single shaft)
103H5210-0410 (Double shaft)

Applicable connector (J.S.T. MFG., CO.)
Connector: EHR-6
Terminal: SEH-001T-P0.6



Sanyo constant current circuit



MICROCHIP PIC18F2455/2550/4455/4550

28/40/44-Pin, High-Performance, Enhanced Flash, USB Microcontrollers with nanoWatt Technology

Universal Serial Bus Features:

- USB V2.0 Compliant
- Low Speed (1.5 Mb/s) and Full Speed (12 Mb/s)
- Supports Control, Interrupt, Isochronous and Bulk Transfers
- Supports up to 32 Endpoints (16 bidirectional)
- 1-Kbyte Dual Access RAM for USB
- On-Chip USB Transceiver with On-Chip Voltage Regulator
- Interface for Off-Chip USB Transceiver
- Streaming Parallel Port (SPP) for USB streaming transfers (40/44-pin devices only)

Power-Managed Modes:

- Run: CPU on, peripherals on
- Idle: CPU off, peripherals on
- Sleep: CPU off, peripherals off
- Idle mode currents down to 5.8 μ A typical
- Sleep mode currents down to 0.1 μ A typical
- Timer1 Oscillator: 1.1 μ A typical, 32 kHz, 2V
- Watchdog Timer: 2.1 μ A typical
- Two-Speed Oscillator Start-up

Flexible Oscillator Structure:

- Four Crystal modes, including High Precision PLL for USB
- Two External Clock modes, up to 48 MHz
- Internal Oscillator Block:
 - 8 user-selectable frequencies, from 31 kHz to 8 MHz
 - User-tunable to compensate for frequency drift
- Secondary Oscillator using Timer1 @ 32 kHz
- Dual Oscillator options allow microcontroller and USB module to run at different clock speeds
- Fail-Safe Clock Monitor:
 - Allows for safe shutdown if any clock stops

Peripheral Highlights:

- High-Current Sink/Source: 25 mA/25 mA
- Three External Interrupts
- Four Timer modules (Timer0 to Timer3)
- Up to 2 Capture/Compare/PWM (CCP) modules:
 - Capture is 16-bit, max. resolution 5.2 ns (Tcy/16)
 - Compare is 16-bit, max. resolution 83.3 ns (Tcy)
 - PWM output: PWM resolution is 1 to 10-bit
- Enhanced Capture/Compare/PWM (ECCP) module:
 - Multiple output modes
 - Selectable polarity
 - Programmable dead time
 - Auto-shutdown and auto-restart
- Enhanced USART module:
 - LIN bus support
- Master Synchronous Serial Port (MSSP) module supporting 3-wire SPI (all 4 modes) and I²C™ Master and Slave modes
- 10-bit, up to 13-channel Analog-to-Digital Converter module (A/D) with Programmable Acquisition Time
- Dual Analog Comparators with Input Multiplexing

Special Microcontroller Features:

- C Compiler Optimized Architecture with optional Extended Instruction Set
- 100,000 Erase/Write Cycle Enhanced Flash Program Memory typical
- 1,000,000 Erase/Write Cycle Data EEPROM Memory typical
- Flash/Data EEPROM Retention: > 40 years
- Self-Programmable under Software Control
- Priority Levels for Interrupts
- 8 x 8 Single-Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
 - Programmable period from 41 ms to 131s
- Programmable Code Protection
- Single-Supply 5V In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins
- Optional dedicated ICD/ICSP port (44-pin devices only)
- Wide Operating Voltage Range (2.0V to 5.5V)

Device	Program Memory		Data Memory		I/O	10-Bit A/D (ch)	CCP/ECCP (PWM)	SPP	MSSP		EUSART	Comparators	Timers 8/16-Bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SPI	Master I ² C™			
PIC18F2455	24K	12288	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F2550	32K	16384	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F4455	24K	12288	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3
PIC18F4550	32K	16384	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3

STA401A

NPN Darlington

With built-in avalanche diode

External dimensions (D) STA (10-pin)

Absolute maximum ratings

($T_c=25^\circ\text{C}$)

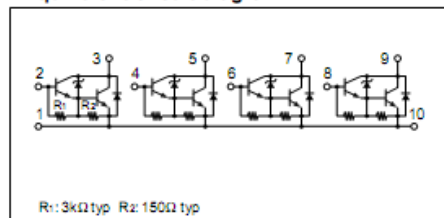
Symbol	Ratings	Unit
V_{ce0}	60 ± 10	V
V_{cso}	60 ± 10	V
V_{es0}	6	V
I_c	4	A
I_{cP}	8 (PW \leq 10ms, D \leq 50%)	A
P_T	4 ($T_a=25^\circ\text{C}$)	W
	20 ($T_c=25^\circ\text{C}$)	
T_j	150	$^\circ\text{C}$
T_{stg}	-40 to +150	$^\circ\text{C}$

Electrical characteristics

($T_c=25^\circ\text{C}$)

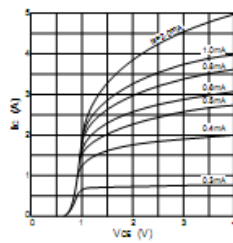
Symbol	Specification			Unit	Conditions
	min	typ	max		
I_{cso}			100	μA	$V_{ce}=50\text{V}$
I_{eso}			10	mA	$V_{es}=6\text{V}$
V_{ce0}	50	60	70	V	$I_c=10\text{mA}$
h_{FE}	1000				$V_{ce}=4\text{V}$, $I_c=3\text{A}$
$V_{ce(sat)}$			2.0	V	$I_c=3\text{A}$, $I_b=10\text{mA}$
t_{on}		1.0		μs	$V_{cc}=30\text{V}$, $I_c=3\text{A}$, $I_{b1}=I_{b2}=10\text{mA}$
t_{stg}		4.0		μs	
t_r		1.5		μs	

Equivalent circuit diagram

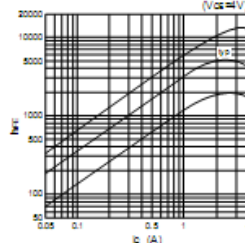


Characteristic curves

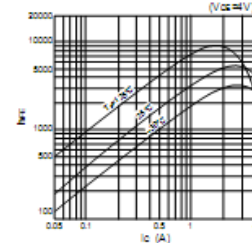
I_c - V_{ce} Characteristics (Typical)



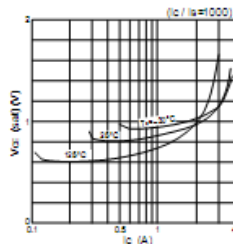
h_{FE} - I_c Characteristics (Typical)



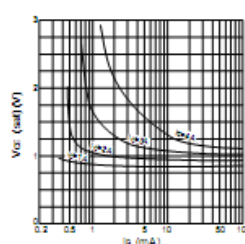
h_{FE} - I_c Temperature Characteristics (Typical)



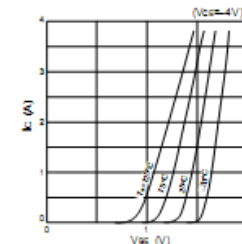
$V_{ce(sat)}$ - I_c Temperature Characteristics (Typical)



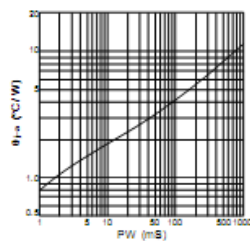
$V_{ce(sat)}$ - I_b Characteristics (Typical)



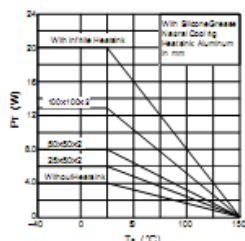
I_c - V_{ce} Temperature Characteristics (Typical)



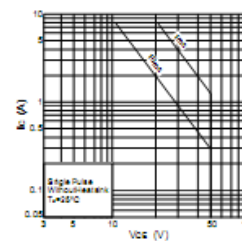
θ_{jw} -PW Characteristics



P_T - T_c Characteristics



Safe Operating Area (SOA)

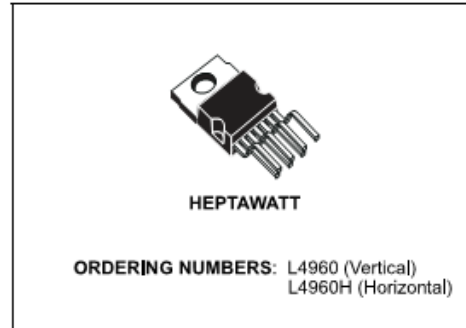




L4960

2.5A POWER SWITCHING REGULATOR

- 2.5A OUTPUT CURRENT
- 5.1V TO 40V OUTPUT VOLTAGE RANGE
- PRECISE ($\pm 2\%$) ON-CHIP REFERENCE
- HIGH SWITCHING FREQUENCY
- VERY HIGH EFFICIENCY (UP TO 90%)
- VERY FEW EXTERNAL COMPONENTS
- SOFT START
- INTERNAL LIMITING CURRENT
- THERMAL SHUTDOWN



DESCRIPTION

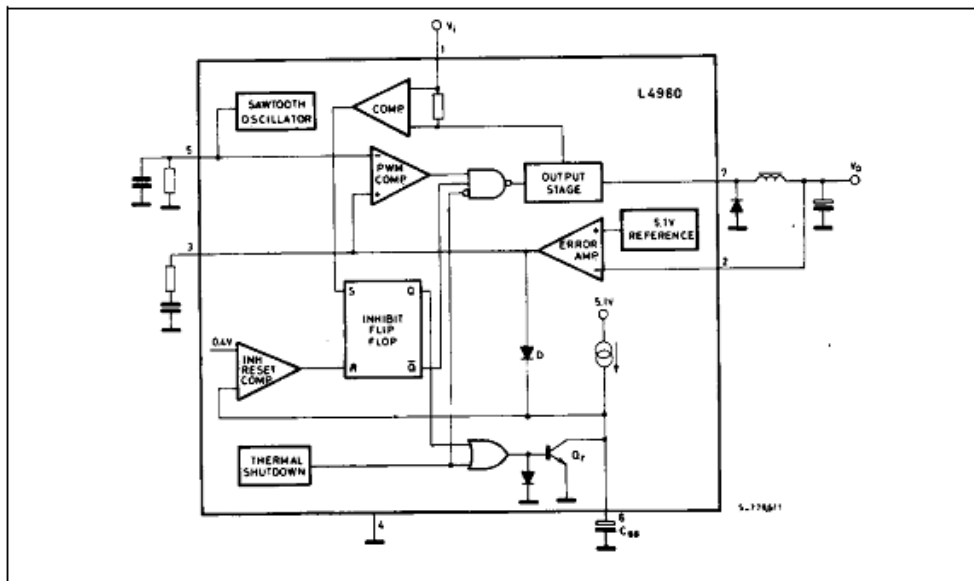
The L4960 is a monolithic power switching regulator delivering 2.5A at a voltage variable from 5V to 40V in step down configuration.

Features of the device include current limiting, soft start, thermal protection and 0 to 100% duty cycle for continuous operation mode.

The L4960 is mounted in a Heptawatt plastic power package and requires very few external components.

Efficient operation at switching frequencies up to 150KHz allows a reduction in the size and cost of external filter components.

BLOCK DIAGRAM





L7800 SERIES

POSITIVE VOLTAGE REGULATORS

- OUTPUT CURRENT TO 1.5A
- OUTPUT VOLTAGES OF 5; 5.2; 6; 8; 8.5; 9; 10; 12; 15; 18; 24V
- THERMAL OVERLOAD PROTECTION
- SHORT CIRCUIT PROTECTION
- OUTPUT TRANSITION SOA PROTECTION

DESCRIPTION

The L7800 series of three-terminal positive regulators is available in TO-220, TO-220FP, TO-220FM, TO-3 and D²PAK packages and several fixed output voltages, making it useful in a wide range of applications. These regulators can provide local on-card regulation, eliminating the distribution problems associated with single point regulation. Each type employs internal current limiting, thermal shut-down and safe area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltage and currents.

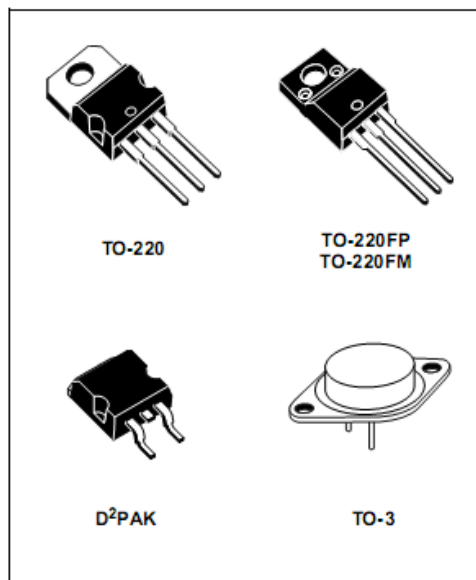
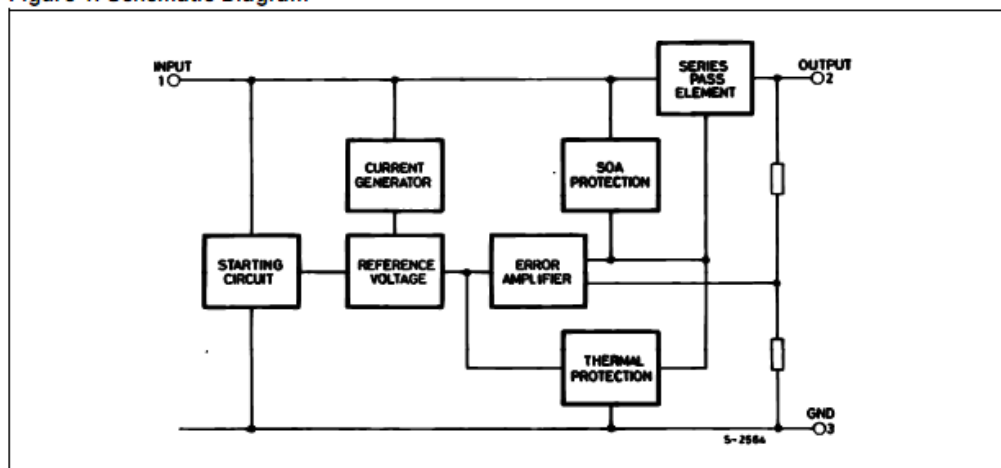


Figure 1: Schematic Diagram



November 2004

Rev. 12

1/34



High Density Mounting Type Photocoupler

LTV-817 Series

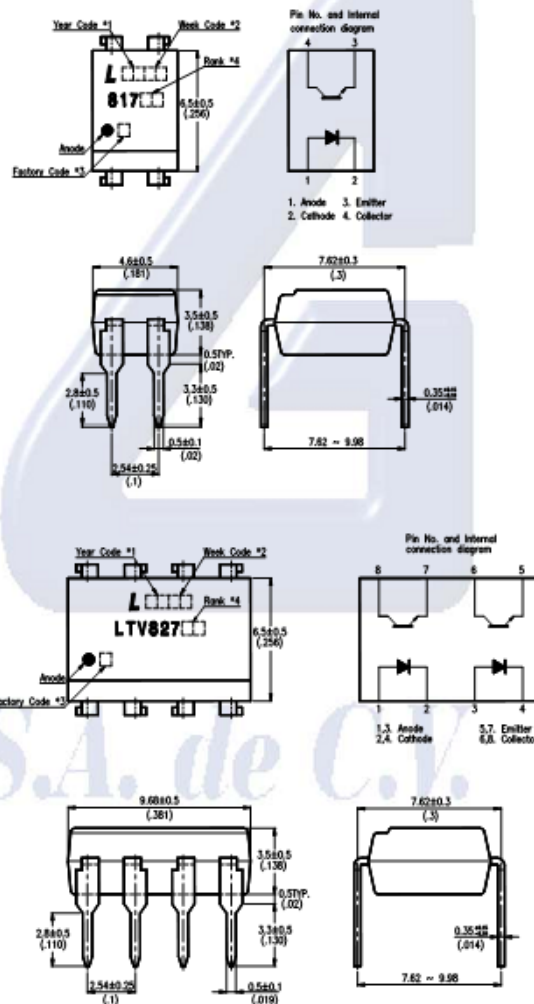
Features

- Current transfer ratio
(CTR : MIN. 50% at $I_F=5\text{mA}$, $V_{CE}=5\text{V}$)
- High input-output isolation voltage:
(Viso : 5,000V_{rms})
- Compact dual-in-line package
LTV-817 : 1-channel type
LTV-827 : 2-channel type
LTV-847 : 4-channel type
- UL approved (No. E113898)
- TUV approved (No. R9653630)
- CSA approved (No. CA91533-1)
- FIMKO approved (No. 202634)
- NEMKO approved (No. P98101945)
- DEMKO approved (No. 307857)
- SEMKO approved (No. 9832157/01-03)
- VDE approved (No. 094722)
- Options available :
 - Leads with 0.4"(10.16mm)spacing (M Type)
 - Leads bends for surface mounting(S Type)
 - Tape and Reel of Type I for SMD(Add"-TA"Suffix)
 - Tape and Reel of Type II for SMD(Add"-TA1"Suffix)
 - VDE 0884 approvals (Add"-V"Suffix)

Applications

1. Computer terminals.
2. System appliances, measuring instruments.
3. Registers, copiers, automatic vending machines.
4. Electric home appliances such as fan heaters, etc.
5. Signal transmission between circuits of different potentials and impedances.

Package Dimensions



C106 Series

Preferred Devices

Sensitive Gate Silicon Controlled Rectifiers

Reverse Blocking Thyristors

Glassivated PNP devices designed for high volume consumer applications such as temperature, light, and speed control; process and remote control, and warning systems where reliability of operation is important.

Features

- Glassivated Surface for Reliability and Uniformity
- Power Rated at Economical Prices
- Practical Level Triggering and Holding Characteristics
- Flat, Rugged, Thermopad Construction for Low Thermal Resistance, High Heat Dissipation and Durability
- Sensitive Gate Triggering
- Pb-Free Packages are Available*



ON Semiconductor®

<http://onsemi.com>

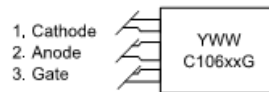
SCRs

4 A RMS, 200 – 600 Volts



TO-225AA
CASE 077
STYLE 2

MARKING DIAGRAM & PIN ASSIGNMENT



Y = Year
 WW = Work Week
 C106xx = Device Code
 xx = B, D, D1, M, M1
 G = Pb-Free Package

ORDERING INFORMATION

See detailed ordering and shipping information in the package dimensions section on page 2 of this data sheet.

*For additional information on our Pb-Free strategy and soldering details, please download the ON Semiconductor Soldering and Mounting Techniques Reference Manual, SOLDERRM/D.



STPS5L40

POWER SCHOTTKY RECTIFIER

MAIN PRODUCT CHARACTERISTICS

$I_{F(AV)}$	5 A
V_{RRM}	40 V
T_j (max)	150°C
V_F (max)	0.44 V

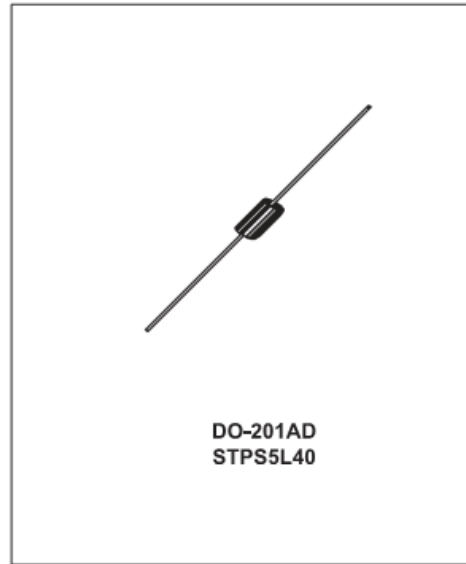
FEATURES AND BENEFITS

- NEGLIGIBLE SWITCHING LOSSES
- LOW FORWARD VOLTAGE DROP FOR HIGHER EFFICIENCY.
- LOW THERMAL RESISTANCE
- AVALANCHE CAPABILITY SPECIFIED

DESCRIPTION

Axial Power Schottky rectifier suited for Switch Mode Power Supplies and high frequency inverters.

Packaged in DO-201AD, this device is intended for use in low voltage output for small battery chargers & consumer SMPS such as DVD and Set-Top-Box..



ABSOLUTE RATINGS (limiting values)

Symbol	Parameter	Value	Unit
V_{RRM}	Repetitive peak reverse voltage	40	V
$I_{F(RMS)}$	RMS forward current	15	A
$I_{F(AV)}$	Average forward current	5	A
I_{FSM}	Surge non repetitive forward current	150	A
P_{ARM}	Repetitive peak avalanche power	2700	W
T_{stg}	Storage temperature range	-65 to +150	°C
T_j	Maximum operating junction temperature *	150	°C
dV/dt	Critical rate of rise of reverse voltage (rated V_R , $T_j = 25^\circ\text{C}$)	10000	V/ μs

* : $\frac{dP_{tot}}{dT_j} < \frac{1}{R_{th(j-a)}}$ thermal runaway condition for a diode on its own heatsink

RL201 THRU RL207



2.0 AMP SILICON RECTIFIERS



FEATURES

- * Low forward voltage drop
- * High current capability
- * High reliability
- * High surge current capability

MECHANICAL DATA

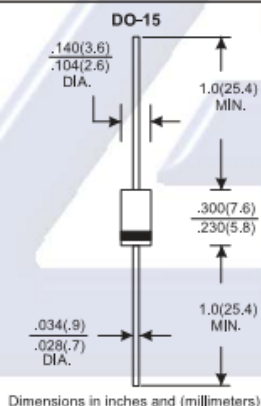
- * Case: Molded plastic
- * Epoxy: UL 94V-0 rate flame retardant
- * Lead: Axial leads, solderable per MIL-STD-202, method 208 guaranteed
- * Polarity: Color band denotes cathode end
- * Mounting position: Any
- * Weight: 0.40 grams

VOLTAGE RANGE

50 to 1000 Volts

CURRENT

2.0 Amperes



Dimensions in inches and (millimeters)

MAXIMUM RATINGS AND ELECTRICAL CHARACTERISTICS

Rating 25°C ambient temperature unless otherwise specified.
Single phase half wave, 60Hz, resistive or inductive load.
For capacitive load, derate current by 20%.

TYPE NUMBER	RL201	RL202	RL203	RL204	RL205	RL206	RL207	UNITS	
Maximum Recurrent Peak Reverse Voltage	50	100	200	400	600	800	1000	V	
Maximum RMS Voltage	35	70	140	280	420	560	700	V	
Maximum DC Blocking Voltage	50	100	200	400	600	800	1000	V	
Maximum Average Forward Rectified Current .375"(9.5mm) Lead Length at Ta=75°C								2.0	A
Peak Forward Surge Current, 8.3 ms single half sine-wave superimposed on rated load (JEDEC method)								70	A
Maximum Instantaneous Forward Voltage at 2.0A								1.0	V
Maximum DC Reverse Current Ta=25°C								5.0	µA
at Rated DC Blocking Voltage Ta=100°C								50	µA
Typical Junction Capacitance (Note 1)								20	pF
Typical Thermal Resistance RθJA (Note 2)								40	°C/W
Operating and Storage Temperature Range Tj, Tstg								-65 — +150	°C

NOTES:

1. Measured at 1MHz and applied reverse voltage of 4.0V D.C.
2. Thermal Resistance from Junction to Ambient .375" (9.5mm) lead length.