



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

**FACULTAD DE ESTUDIOS SUPERIORES CUAUTILÁN
INGENIERÍA MECÁNICA ELÉCTRICA**

**PROYECTO Y CONSTRUCCIÓN DE UN MODULO DE
POTENCIA Y UN BRAZO ROBÓTICO AUTOMATIZADO POR
COMPUTADORA Y MICROCONTROLADORES PICAXE.
Como ejemplo de las capacidades de automatización que se pueden
obtener en diversos entornos.**

TESIS

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO MECÁNICO ELECTRICISTA

PRESENTA: DANIEL SÁENZ NISHIMURA

ASESOR: ING. JAVIER HERNÁNDEZ VEGA

CUAUTILÁN IZCALLI, ESTADO DE MÉXICO 2013



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.



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

**FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN
UNIDAD DE ADMINISTRACIÓN ESCOLAR
DEPARTAMENTO DE EXÁMENES PROFESIONALES**

Con base en el Reglamento General de Exámenes, y la Dirección de la Facultad, autoriza al alumno:

DANIEL SÁENZ NISHIMURA

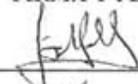
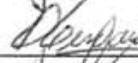
Con número de cuenta: **40209571-8** a presentar LA TESIS:

“Proyecto y Construcción de un Módulo de Potencia y un Brazo Robótico Automatizado por Computadora y Microcontroladores PICAXE. Como Ejemplo de las Capacidades de Automatización que se Pueden Obtener en Diversos Entornos”.

Bajo la asesoría del: **Ing. Javier Hernández Vega**

Para obtener el título de: **Ingeniero Mecánico Electricista**

PROFESORES QUE INTEGRAN EL JURADO

	NOMBRE	FIRMA Y FECHA
PRESIDENTE	<u>Ing. Javier Hernández Vega</u>	 3/05/13
VOCAL	<u>Ing. Oscar Cervantes Torres</u>	 9/05/13
SECRETARIO	<u>Ing. Angel Isaias Lima Gómez</u>	 07/05/13
1er SUPLENTE	<u>Ing. Ernesto Alfonso Ramírez Orozco</u>	 9/05/13
2do SUPLENTE	<u>Ing. Gilberto Chavarría Ortiz</u>	7/mayo/2013

Atentamente notificamos su participación en la revisión y evaluación del trabajo para que en un plazo no mayor a 30 días hábiles emita su VOTO APROBATORIO.

ATENTAMENTE
“POR MI RAZA HABLARÁ EL ESPÍRITU”
 Cuautitlán Izcalli, Méx. a 30 de Abril de 2013.


L.A. ARACELI HERRERA HERNÁNDEZ
JEFA DEL DEPARTAMENTO

NOTA: los sinodales suplentes están obligados a presentarse el día y hora del Examen Profesional (art. 127).

HHA/Vc

Agradecimientos

A Dios

Por permitirme concluir esta etapa de mi vida con salud para mis seres queridos y para mí.

A la Universidad Nacional Autónoma de México

Por brindarme el honor de ser Universitario en la Máxima Casa de Estudios, la que en todo momento se ha asegurado de hacerme sentir bienvenido en cualquiera de sus recintos que he visitado. Por mi Raza Hablará el Espíritu.

A la Facultad de Estudios Superiores Cuautitlán

Por abrirme las puertas de su comunidad dándome la oportunidad de aprender y forjarme como profesionalista en sus aulas, haciendo de mis años de estudio una de las mejores etapas de mi vida.

A mis padres Belinda y Jaime

Por su cariño y cuidado, siempre me faltarán palabras para lograr agradecerles todo lo que han hecho por mí. Me han guiado a lo largo de mi vida y siempre estaré orgulloso de ser su hijo. Este trabajo se los dedico especialmente a ustedes que me lo han dado todo.

A mis Hermanos Verónica y Alejandro

Porque como mis hermanos mayores siempre me han cuidado y brindado su apoyo en todo momento. Les agradezco por su comprensión y su paciencia.

A mis Amigos

Hazael, Sergio, Orlando, Brenda y Mary; por compartir sus vidas conmigo y permitirme ser parte de las suyas, gracias por las risas y por su sinceridad, simplemente gracias por su amistad. Estoy agradecido con dios y con la vida por haberlos conocido.

A mi Asesor

Por todo el tiempo y la paciencia que me brindó. Lo admiro por su vocación de enseñanza, ya que es un profesor que deja huella en sus estudiantes y le agradezco porque sin usted esta tesis no hubiera sido posible.

A mis sinodales

Les agradezco por sus valiosas aportaciones y comentarios, sobre todo por el apoyo e interés que me mostraron al ayudarme a finalizar este trabajo.

A mis Profesores de la FESC

Por su completa dedicación como académicos de la Máxima Casa de Estudios, por compartir conmigo todo su conocimiento y experiencia. Les agradezco por la formación que me dieron como profesionista.

Índice.

	Pág.
Marco conceptual y de referencia	
i. CONCEPTOS.....	1
ii. PLANTEAMIENTO DEL PROBLEMA.....	1
iii. OBJETIVO GENERAL.....	2
iv. OBJETIVOS ESPECÍFICOS.....	2
v. JUSTIFICACIÓN.....	3
Introducción.....	4
Capítulo 1	
MODULO DE CONTROL Y POTENCIA.....	7
1.1 EL SCR.....	7
1.1.1 Características eléctricas.....	8
1.2 CARACTERÍSTICAS DE DISPARO Y TIPOS DE DISPARO.....	10
1.2.1 Características de voltaje – corriente en la Compuerta.....	10
1.2.2 Disparo por corriente directa.....	11
1.2.3 Disparo por corriente alterna.....	12
1.2.4 Disparo por unijunction transistor (transistor de unijuntura).....	13
1.2.5 Disparo por cambio de fase (CA).....	17
1.3 LA INTERFACE DEL MODULO DE POTENCIA.....	26
1.3.1 La transmisión paralela y el puerto paralelo.....	28
Capítulo 2	
FUNDAMENTOS BÁSICOS DE ROBÓTICA INDUSTRIAL.....	33
2.1 RESEÑA DE LA ROBÓTICA.....	33
2.2 PARTES DEL ROBOT.....	34
2.2.1 El controlador.....	34
2.2.2 Manipulador.....	36
2.2.3 El Efecto Final.....	37
2.2.4 Fuente de Poder.....	37
2.2.5 Formas de programación.....	37

2.2.6	Actuadores.....	38
2.3	TIPOS DE ACTUADORES.....	38
2.3.1	Actuadores Hidráulicos.	38
2.3.2	Actuadores Neumáticos.	39
2.3.3	Actuadores Eléctricos.....	40
2.4	RESEÑA DE LOS MOTORES DE CORRIENTE DIRECTA.....	41
2.4.1	Motores shunt (paralelo).	41
2.4.2	Motores Serie.....	42
2.4.3	Motor Compound (compuesto).....	42
2.4.4	Motor de Imanes permanentes.	43
2.5	GENERALIDADES DE LOS ACOPLAMIENTOS POR ENGRANES	44
2.5.1	Engranés rectos.....	44
2.5.2	Engranés Helicoidales.....	44
2.5.3	Engranés cónicos.....	45
2.5.4	Engrane de piñón-cremallera.....	46
2.5.5	Engranés corona sin fin. (Worm gear pair).....	46
2.6	TIPOS DE ROBOTS CON RESPECTO A SU VOLUMEN DE TRABAJO.	47
2.6.1	Configuración de Brazo articulado.	48
2.6.2	Configuración de robot cartesiano.	51
2.6.3	Configuración cilíndrica.....	51
2.6.4	Configuración esférica (polar).....	52
Capítulo 3		
	PROYECTO Y CONSTRUCCIÓN DEL BRAZO ROBÓTICO.....	54
3.1	SELECCIÓN DEL BRAZO ROBÓTICO.....	54
3.1.1	Características Básicas.....	54
3.1.2	Construcción.	55
3.1.3	Sensores.....	58
3.1.4	Rediseño de la base del brazo robótico.	60
Capítulo 4		
	PROGRAMACIÓN DEL BRAZO ROBÓTICO.....	66
4.1	RESEÑA DE LA PROGRAMACIÓN DE UN ROBOT.....	66
4.1.1	Control por computadora.....	68
Capítulo 5		
	CONTROL DEL MODULO DE POTENCIA POR MEDIO DE	74

5.1	EL MICRO CONTROLADOR.	74
5.2	EL SISTEMA PICAXE	75
5.3	SISTEMA PICAXE IMPLEMENTADO EN ESTE PROYECTO.....	78
5.3.1	Picaxe1	79
5.3.2	Características del sistema de transmisión serial.	80
5.3.3	Programación del Chip 1.....	83
5.3.4	Circuito electrónico del Picaxe1.....	87
5.3.5	Picaxe2.	89
5.3.6	Programación del Chip 2.....	91
5.3.7	Picaxe3.	93
5.3.8	Programación del Chip 3.....	95
5.3.9	Circuito electrónico Picaxe2 y Picaxe3.....	99
	Conclusiones	101
	Bibliografía.....	103

Marco conceptual y de referencia.

i. CONCEPTOS.

▲ Automatización.

La automatización es el cumplimiento independiente de una tarea o función, realizada por un sistema o dispositivo que previamente fue programada por un humano.

▲ Automatización por PC.

De forma conjunta con el software, la PC en diferentes formas es el núcleo de un amplio rango de diversas tareas de automatización como el control de maquinaria en la industria. Para tareas básicas de automatización, la tecnología de control basada en PC ofrece una excelente escalabilidad y flexibilidad.

▲ Microcontrolador.

Un microcontrolador es una pequeña computadora dentro de un circuito integrado, el cual contiene un núcleo de procesamiento, memoria y periféricos programables de entrada/salida.

ii. PLANTEAMIENTO DEL PROBLEMA.

Los aspectos más relevantes de la problemática son:

▲ El limitado acceso que tienen las pequeñas y medianas industrias en nuestro país, a las nuevas tecnologías de automatización.

▲ La necesidad de aplicaciones específicas en el área de la automatización de forma sencilla y económica.

- ⤴ El grado de dificultad para introducirse a la programación y manejo de sistemas automatizados.

iii. OBJETIVO GENERAL.

Presentar un ejemplo desde mi perspectiva acerca de las capacidades de automatización que se pueden obtener en diversos entornos con herramientas que se encuentran a nuestro alcance, entornos como el área industrial o la vida cotidiana.

El ejemplo se basa en el proyecto y construcción de un modulo de potencia, así como de un brazo robótico, el cual es controlado por computadora y microcontroladores picaxe.

iv. OBJETIVOS ESPECÍFICOS.

- ⤴ Diseño y construcción de un modulo de control y potencia que se encargue de energizar los motores de c.d. que conforman al brazo robótico. Así mismo el modulo contará con la interface que interpretará las señales provenientes de la PC, las cuales a su vez activaran los SCR's (Rectificadores controlados de silicio) para controlar los motores de corriente directa. Adicionalmente el modulo de potencia albergará un sistema de microcontroladores picaxe.
- ⤴ Diseño y construcción de un brazo robótico conformado por motores de c.d. equipados con sensores magnéticos tipo reed switch. El cual será energizado por el modulo de potencia y controlado por medio de la computadora y microcontroladores picaxe.
- ⤴ Diseño y elaboración de un programa en lenguaje Q-basic, que a través del puerto paralelo de la computadora y por medio de la interface del modulo de potencia, controle los movimientos del brazo robótico.
- ⤴ Diseño y construcción de un sistema de microcontroladores picaxe incorporados en el modulo de potencia, los cuales controlen los

movimientos del brazo robótico con ayuda de los sensores magnéticos. Haciendo uso de su característica de simulación de tareas en paralelo para controlar cada uno de los motores, registrando la información de los sensores simultáneamente.

- ▲ Diseño y construcción de un modulo independiente, el cuál a través de un microcontrolador picaxe se envíen instrucciones vía serial (usando el protocolo RS-232) por medio de cable coaxial, al sistema de microcontroladores que se encuentran en el modulo de potencia y control.

v. JUSTIFICACIÓN.

El presente trabajo está dirigido a ingenieros, estudiantes y técnicos que estén familiarizados o deseen introducirse al control y automatización.

Este tipo de sistema es un conjunto de herramientas que se encuentran a nuestro alcance y es fácil de aplicar. Tomando en cuenta la lenta migración a nuevas tecnologías en la industria mexicana, principalmente por factores económicos, considero que la aplicación de este tipo de herramientas en el área de la automatización en nuestro país sería eficiente y económica.

Introducción.

El término programador siempre nos ha evocado a un profesional de la computación en la interpretación y creación de programas para diferentes aplicaciones en diversos lenguajes de programación. Sin embargo la acelerada evolución de la computación en los últimos años ha permitido que la computadora se haya vuelto una herramienta universal a la que cada vez más personas tienen acceso. Por lo que para muchas personas se nos ha facilitado hacernos partícipes del papel como programadores, lo que nos ha permitido aplicar conocimientos propios de nuestras áreas de formación profesional, haciendo de la computadora una herramienta fundamental.

En el tercer semestre de mi carrera participé en el 1er Concurso de Innovación Tecnológica, donde afortunadamente obtuve el primer lugar con el trabajo “Brazo robótico controlado por computadora”, con publicaciones en la “Gaceta UNAM”¹ y “Gaceta UNAM Comunidad Cuautitlán”.²

Con la elaboración de este proyecto, el cual he mejorado constantemente desde entonces, he tenido como propósito presentar una idea sobre mi perspectiva de la automatización en nuestro país.

No cabe duda que en las industrias, cada vez hay más procesos y tareas automatizadas que son operados por controladores lógicos programables PLC's, pero que en muchas ocasiones por la interconexión de varios de ellos y la complejidad propia de una industria, son controlados a su vez por una computadora central, que está al tanto de todas las tareas automatizadas y que de hecho es capaz de modificar a distancia las programaciones independientes de cada uno de los procesos.

Sin embargo, es común que muchos no nos percatemos que tenemos a nuestro alcance ese nivel de tecnología, si bien no contamos con la

¹ “Gaceta UNAM” #3607. Febrero 2003. P.6

<http://www.dgcs.unam.mx/gacetaweb/2003/06feb03.pdf>

² Gaceta UNAM Comunidad Cuautitlán Enero 2003. P.4

infraestructura tecnológica que les caracteriza a las compañías profesionales de automatización, como lo son las normas y certificaciones con la que cumplen sus productos. Pero en el aspecto de la programación es donde la computadora demuestra ser una poderosa herramienta a nuestro alcance, ya que al contar con puertos de comunicación de datos, nos resta diseñar interfaces de control entre la computadora y el medio que deseamos automatizar.

Por otra parte la computadora no es la única herramienta que tenemos a nuestro alcance en el área de la automatización, la tecnología de los microcontroladores se encuentra en constante evolución y ha alcanzado capacidades tan altas como lo han demostrado tecnologías como PICAXE y ARDUINO.

La tecnología a la que poco a poco me he adentrado es PICAXE. Y es fácil percatarse que en su sencillez radica su poder, son una nueva generación de microcontroladores que pueden ser programados una y otra vez hasta 100,000 veces por medio de una computadora sin la necesidad de un circuito programador independiente. PICAXE intenta romper la barrera de la programación clásica al incluir diferentes métodos de programación en varios software que permiten programar por diagramas de flujo, lenguaje de programación Basic y por contactos (“ladder” o escalera usualmente utilizado para programar PLC’s).

Cuentan con una amplia lista de aplicaciones, de entre ellas la transmisión de datos por un solo hilo (Serial), la cual implementé en este proyecto permitiendo un control a largas distancias sin la necesidad de múltiples cables de conexión, ahorrando espacio y presupuesto.

La conectividad entre la PC y los chips PICAXE es sencilla y fácil de usar, por lo que la combinación de ambos nos proporciona grandes capacidades en automatización, remarcando que el bajo costo de ambas las pone a nuestro alcance.

Respecto a lo anterior y para dar un ejemplo de las capacidades de automatización que podemos obtener utilizando este tipo de herramientas diseñé y construí un sistema conformado por un modulo de potencia, que por medio del disparo de SCR’s (Rectificadores controlados de silicio), el uso de la computadora y microcontroladores PICAXE, automatizo un grupo de motores de corriente directa.

Conjuntamente y con el fin de la demostración de este sistema de automatización llevé a cabo la construcción de un brazo robótico conformado por los motores de corriente directa reutilizados de partes automotrices, haciendo uso de reductores y acoplamientos mecánicos, los cuales a su vez están unidos por una estructura de acero.

Por medio de la descripción detallada en el proceso de diseño y construcción de este proyecto, deseo presentar mi perspectiva del nivel de automatización que podemos alcanzar con herramientas que se encuentran a nuestro alcance, el cual es aplicable tanto al área industrial como al de la vida cotidiana, debido a los considerables bajos costos que ofrece este tipo de tecnología. A su vez, se espera que el lector de esta tesis pueda encontrar este trabajo como un material de apoyo que despierte ideas frescas, principalmente en la comunidad estudiantil, en la creación de nuevos y mejores sistemas automatizados.

Capítulo 1.

MODULO DE CONTROL Y POTENCIA

1.1 EL SCR.

La parte central del modulo de control y potencia es operada por SCR's (Rectificadores controlados de silicio), los cuales son disparados por fase y accionan los motores de c.d.

El SCR fue desarrollado y manufacturado por General Electric Company a principios de 1958 en base al trabajo publicado por los laboratorios Bell en 1956. El SCR es un dispositivo semiconductor de tres terminales, es muy similar a un rectificador de silicio ordinario, el SCR bloquea su flujo normal de corriente hasta que una pequeña señal de corriente es aplicada en la terminal de la compuerta.

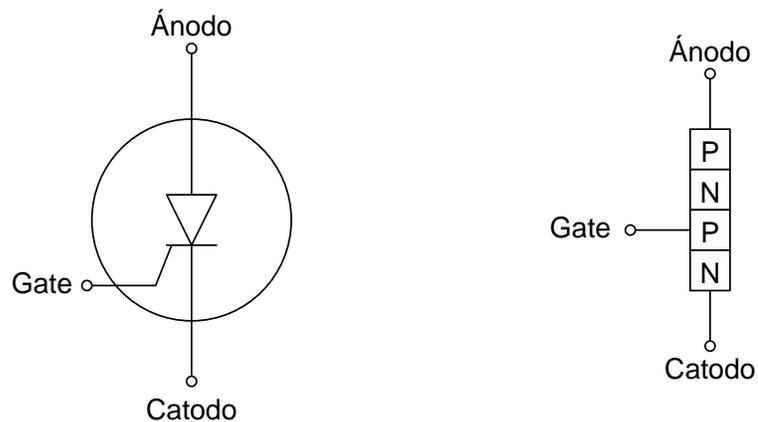


Figura 1.1 Símbolo eléctrico del SCR y su estructura PNPN (de izquierda a derecha).

Después de que la señal en la compuerta es aplicada, el SCR conducirá el flujo de corriente de forma normal muy similar a como lo haría un diodo rectificador ordinario, sin embargo el SCR continuará su flujo incluso después de que la señal en la compuerta sea removida.

El Rectificador Controlado de Silicio tiene una estructura PNP con un ánodo, cátodo y una compuerta. Su símbolo eléctrico, así como su estructura se muestran en la figura 1.1.

Al igual que un transistor, el rectificador controlado de silicio puede actuar como una válvula electrónica que controla el flujo de corriente. Y al igual que un rectificador, deja fluir la corriente en una sola dirección. Con el SCR, la corriente alterna puede ser convertida en corriente directa y, al mismo tiempo, variar la cantidad de corriente.

1.1.1 Características eléctricas.

Las características de polarización directa e inversa de un SCR sin corriente aplicada en la compuerta se muestran en la figura 1.2.

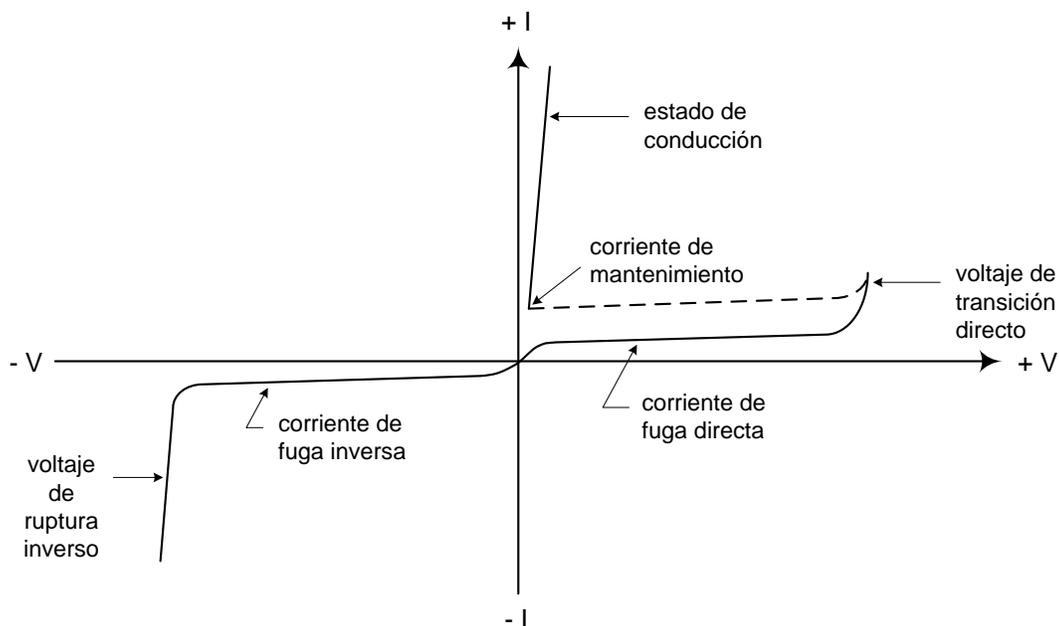


Figura 1.2 Características eléctricas del SCR.

En la polarización inversa, el SCR tiene una característica muy similar a un rectificador normal. Por lo que tiene una alta impedancia hasta que el voltaje inverso alcance un punto de voltaje de ruptura inverso (ruptura de avalancha inversa).

La polarización directa del SCR es diferente a un rectificador normal. El SCR bloquea el flujo directo (también conocido como "corriente de fuga directa") hasta que el voltaje alcance el valor conocido como el voltaje de transición conductora (voltaje de ruptura directo). En este punto, el SCR cambia a un estado de conducción, similar a la condición de conducción de un rectificador.

El SCR se mantendrá en este estado de conducción mientras que la corriente se mantenga por encima de un cierto valor mínimo conocido como corriente de mantenimiento. Si la corriente desciende por debajo de este valor mínimo, el SCR regresará al estado de bloqueo de flujo directo y permanecerá así hasta que una vez más el voltaje alcance el valor de ruptura directa.

La figura 1.3 muestra los cambios en las características de la polarización directa mientras se incrementa el valor en la corriente que se aplica en la compuerta.

La curva $I_{G0}=0$ (corriente en compuerta igual a 0) es la misma que muestra en la figura 1.2, pero ampliada para mostrarse a detalle, cuando se aplica un valor bajo de corriente en la compuerta (I_{G1}), el voltaje donde ocurre la ruptura directa, se reduce. De igual forma, el valor de la corriente de mantenimiento también se reduce. Como se observa en la figura 1.3, mientras que el valor de la corriente en la compuerta aumenta, el voltaje de ruptura directa decrece y se reduce el valor de la corriente de mantenimiento. Cuando la corriente de la compuerta alcanza un valor lo suficientemente alto (I_{G4}), el SCR muestra características casi idénticas a las de un diodo rectificador convencional.

En un estado de operación típico, se usa un SCR en una zona muy por debajo del voltaje de ruptura directo, para que de esta forma el SCR pueda ser disparado inyectando una corriente en su compuerta. El SCR se apagará al reducir la corriente de conducción por debajo del valor de la corriente de sostenimiento. Esta es la única forma de apagar el SCR una

vez que la compuerta ya no tiene el control desde que el SCR ha sido disparado.

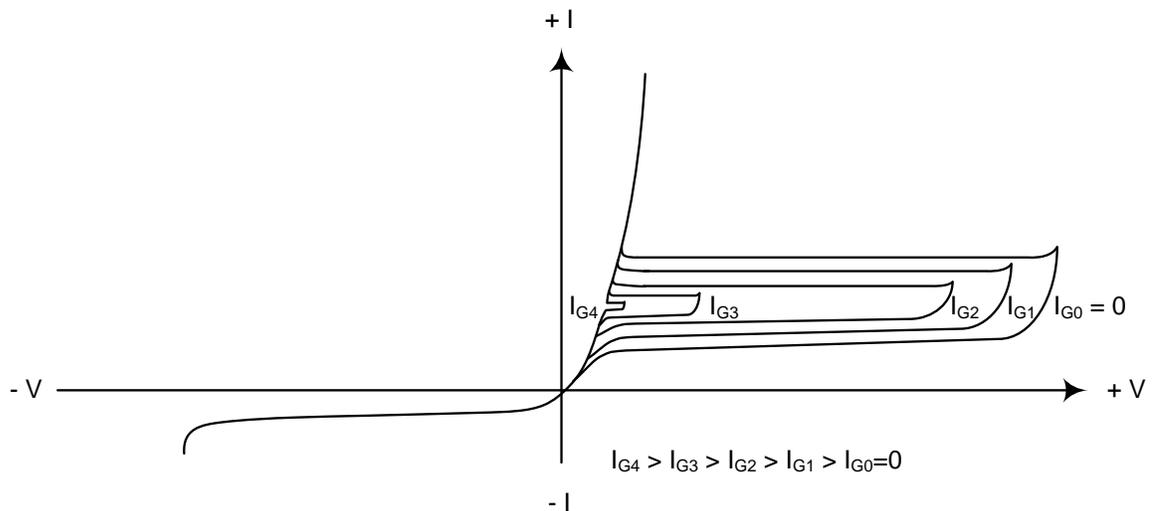


Figura 1.3 Efecto de la corriente en la compuerta del SCR

1.2 CARACTERÍSTICAS DE DISPARO Y TIPOS DE DISPARO.

El diseño de circuitos de disparo para los SCR's envuelve tres consideraciones importantes: a) la selección de un circuito apropiado para suministrar la señal de disparo, dependiendo de las aplicaciones requeridas, b) determinar la corriente y voltaje máximos que el circuito de disparo suministrara a la compuerta del SCR para asegurarse que estas no sean excedidas, y c) determinar la corriente y voltaje mínimo que el circuito de disparo suministrara a la compuerta del SCR para asegurarse de que el SCR se disparará de forma confiable bajo cualquier condición.

1.2.1 Características de voltaje – corriente en la Compuerta.

Las características de la compuerta y de su disparo generalmente se presentan en forma de una gráfica similar al de la figura 1.4. Este tipo de gráficas indican los rangos de corriente y voltaje en la compuerta necesarios para disparar el SCR. También indican los valores límite de voltaje-corriente entre la compuerta y el cátodo.

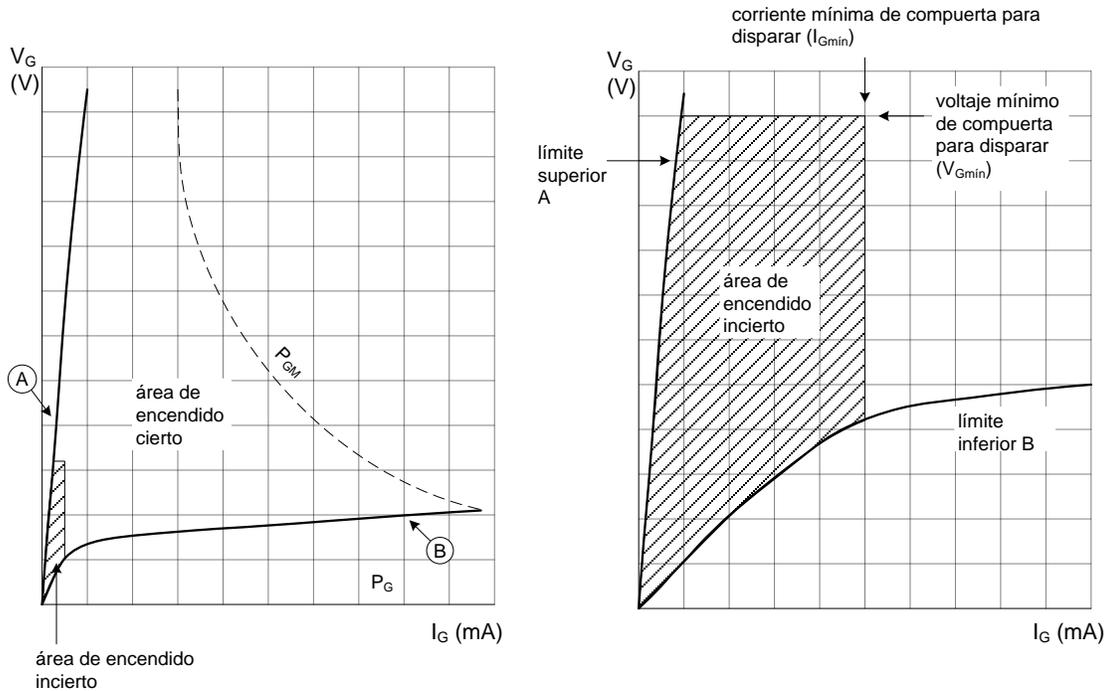


Figura 1.4 Características del disparo en la compuerta del SCR.

1.2.2 Disparo por corriente directa.

Cada gama de familia de SCR's cuentan con sus parámetros y graficas de operación contenidas en las hojas de especificación del fabricante, los cuales muestran los límites máximos y mínimos concernientes a la compuerta de disparo, entre ellos se encuentran: el voltaje de c.d. máximo y mínimo de la compuerta; la potencia pico de la compuerta y la corriente de la compuerta.

En el siguiente diagrama se muestra un disparo básico por corriente directa de un SCR. En el diseño del siguiente circuito sólo se toma en cuenta cumplir con los requerimientos básicos que marcan las hojas de especificación del fabricante para disparar confiablemente el SCR. Inclusive gran parte de los fabricantes incluyen en las hojas de especificación, un circuito básico de operación similar al que se muestra en la figura 1.5.

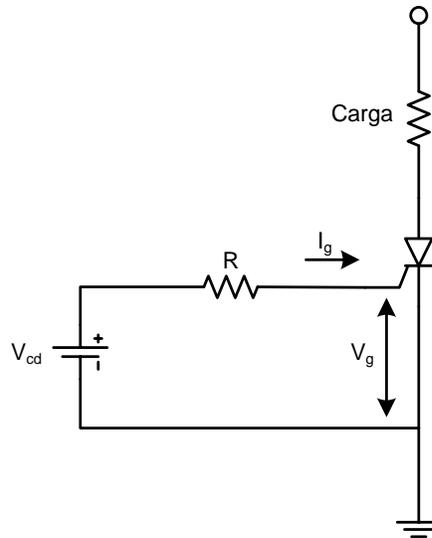


Figura 1.5 Disparo básico de un SCR por corriente directa.

1.2.3 Disparo por corriente alterna.

La figura 1.6 muestra un método simple de obtener corriente en la compuerta para disparar el SCR desde la fuente de corriente alterna, siempre y cuando el ánodo sea positivo con respecto al cátodo. Tan pronto como el SCR se ha disparado, el voltaje del ánodo decrece al estado de conducción y la corriente en la compuerta decrece. La resistencia R limita el pico de corriente en la compuerta (se puede hacer un ejemplo con respecto a los valores de un SCR). El diodo en el circuito de la compuerta previene un voltaje inverso entre el cátodo y la compuerta durante la parte inversa del ciclo.

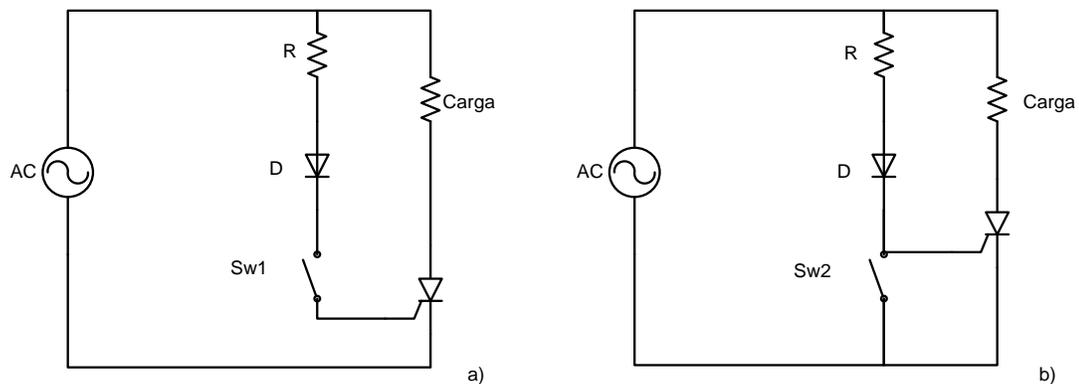


Figura 1.6 Disparo de compuerta por medio de switch.

La conducción comienza al cerrar el switch Sw1 en la figura (1.6 a), o al abrir el switch SW2 en la figura (1.6 b). Para interrumpir la corriente de carga solo se abre el Sw1 o se cierra el Sw2, esto sucede debido a que la misma alternancia del ciclo desactiva al SCR.

1.2.4 Disparo por unijunction transistor (transistor de unijuntura).

El transistor de unijuntura (UJT por sus siglas en inglés) es un semiconductor de tres terminales, emisor (E), base-uno (B_1) y base-dos (B_2). Entre B_1 y B_2 la unijuntura tiene las características de una resistencia ordinaria.

Su principio de operación se muestra en la figura 1.7. Si el voltaje de emisor (V_E), es menor que el voltaje pico (V_P), el emisor se encontrará inversamente polarizado y no habrá conducción. Cuando V_E es igual a V_P , el emisor se polarizará directamente y la resistencia entre el emisor y la base-uno decrece a un nivel muy bajo por lo tanto comienza el estado de conducción del UJT.

El UJT presenta características tales como: (1) tener un voltaje de disparo estable (V_P), (2) requiere un bajo nivel de corriente para su disparo, (3) puede proporcionar pulsos altos de corriente y (4) tiene un bajo costo.

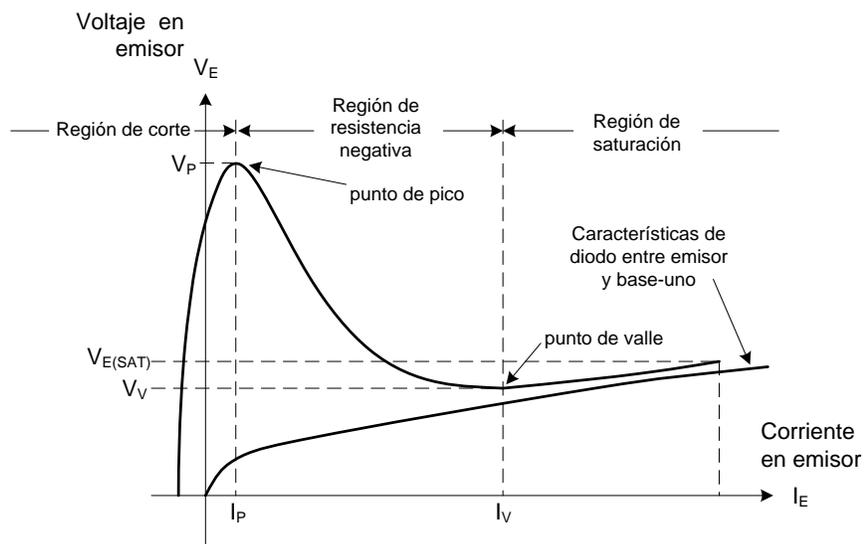


Figura 1.7 Principio de conducción del UJT.

Como oscilador, el UJT proporciona un método muy eficiente para el disparo del SCR por control de fase. El circuito básico de oscilación se

muestra en la figura 1.8 b, al principio del ciclo de operación el emisor se encuentra inversamente polarizado por lo que no hay conducción. Mientras que el capacitor C se carga a través de la resistencia R_T , el voltaje del emisor se incrementa exponencialmente, cuando el voltaje del emisor alcanza un voltaje pico V_p el emisor se polariza directamente y la resistencia dinámica entre el emisor y la base-uno cae a un valor bajo. Por lo que el capacitor C se descarga a través del emisor y el UJT entra en estado de conducción. Cuando el voltaje del emisor cae a un voltaje mínimo $V_{E(\min)}$ (por la propia descarga del capacitor C), el emisor detiene su conducción y se repite el ciclo nuevamente como se observa en la figura 1.8 c.

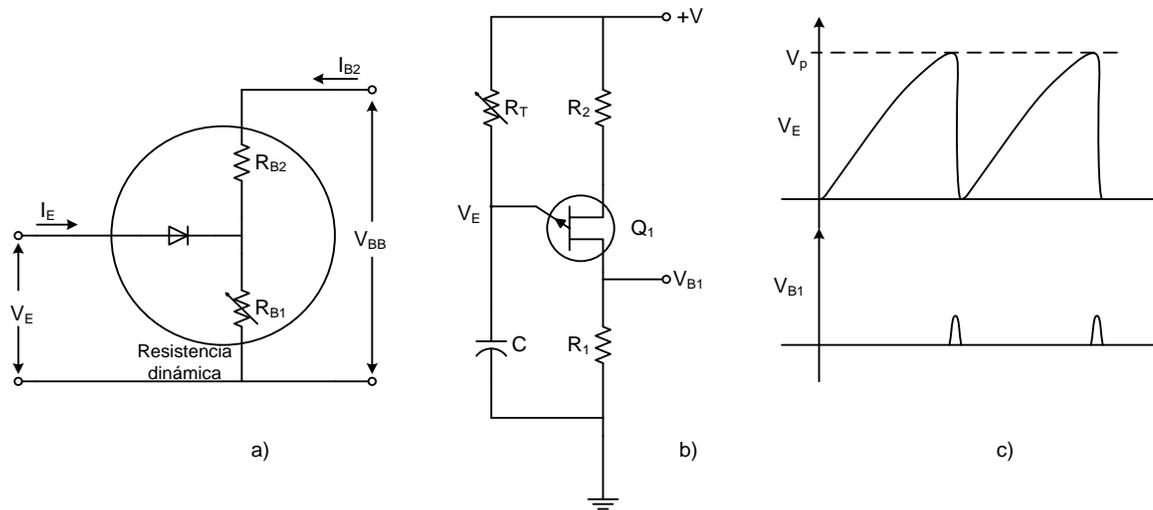


Figura 1.8 Pulso de salida del UJT como oscilador.

El circuito de disparo del SCR por medio del UJT es considerado uno de los mejores sistemas de disparo en los SCR's por ser eficiente, sencillo, compacto, con un bajo consumo de potencia, económico y por proporcionar un amplio control de fase.

El circuito de la figura 1.9 proporciona un control de fase de 180° para onda completa en CA, el cual fue puesto a prueba como dimmer para el control de una lámpara incandescente. Una señal rectificada de onda completa (obtenida del puente rectificador), es usada tanto para suministrar energía al SCR (un solo SCR para el control de una onda completa), como para sincronizar la señal del circuito de disparo. El diodo zener D_z se usa para recortar y regular los picos de la señal de corriente alterna como se muestra en la figura 1.10. En este caso el circuito oscilador funciona ligeramente

diferente al de la figura 1.8 b, al final de cada semiciclo el voltaje en la base del UJT Q_1 cae a cero, causando que el UJT se dispare y provocando que el capacitor se descargue al principio de cada semiciclo.

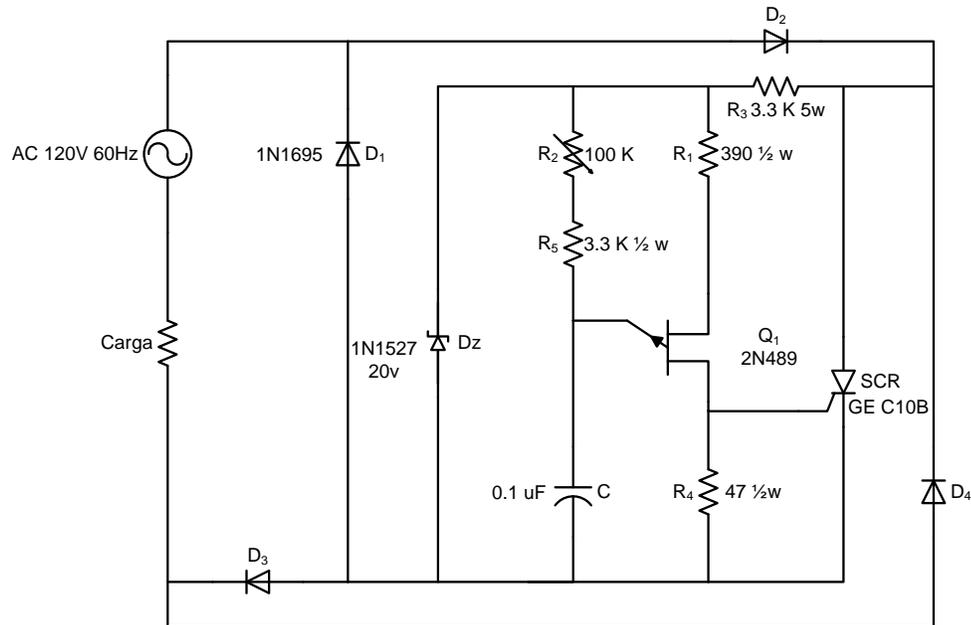


Figura 1.9 Circuito controlador de fase de onda completa en CA, (propuesto por General Electric).³

³ D.R. Grafham , "SCR Manual General Electric" (New York: General Electric, 1972), p. 266.

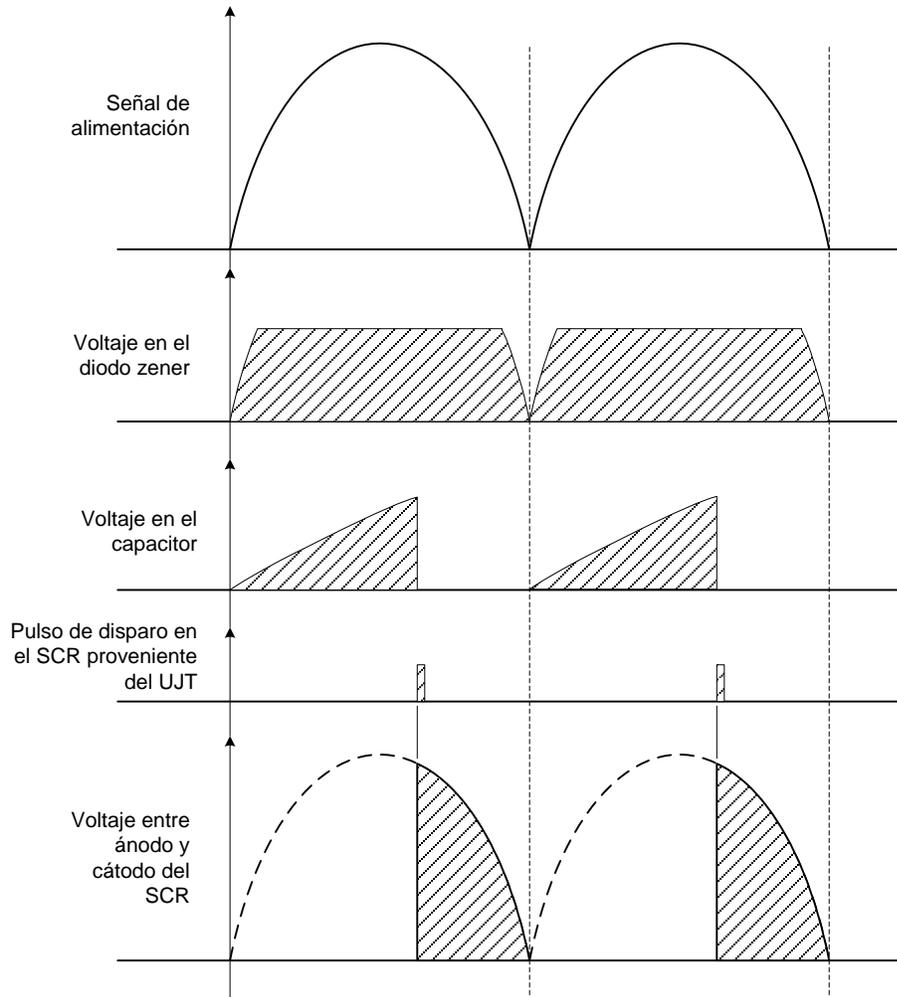


Figura 1.10 Diagrama de disparo del SCR de la figura 1.6

Complementariamente al circuito anterior, en la figura 1.11 se ilustra el disparo de un par de SCR's controlados por fase a través de un UJT, para CD. Al variar R_5 , el voltaje de CD en la carga puede ser ajustado desde su máximo valor hasta cero. Al igual que en el anterior switch de control de fase para CA del circuito 1.9, se usa un solo UJT para suministrar un pulso de disparo en ambos SCR's en ambos semiciclos. El UJT funciona como oscilador de relajación para proporcionar el pulso de disparo, determinado por la constante de tiempo originada por R_2 , R_5 y C. Nuevamente en este caso el ángulo de disparo es ajustado por el valor del potenciómetro R_5 , a 60 Hz, el ángulo de disparo de este circuito puede variar aproximadamente

de 10° a 180° . Este circuito es una variante propia con respecto al propuesto por General Electric.⁴

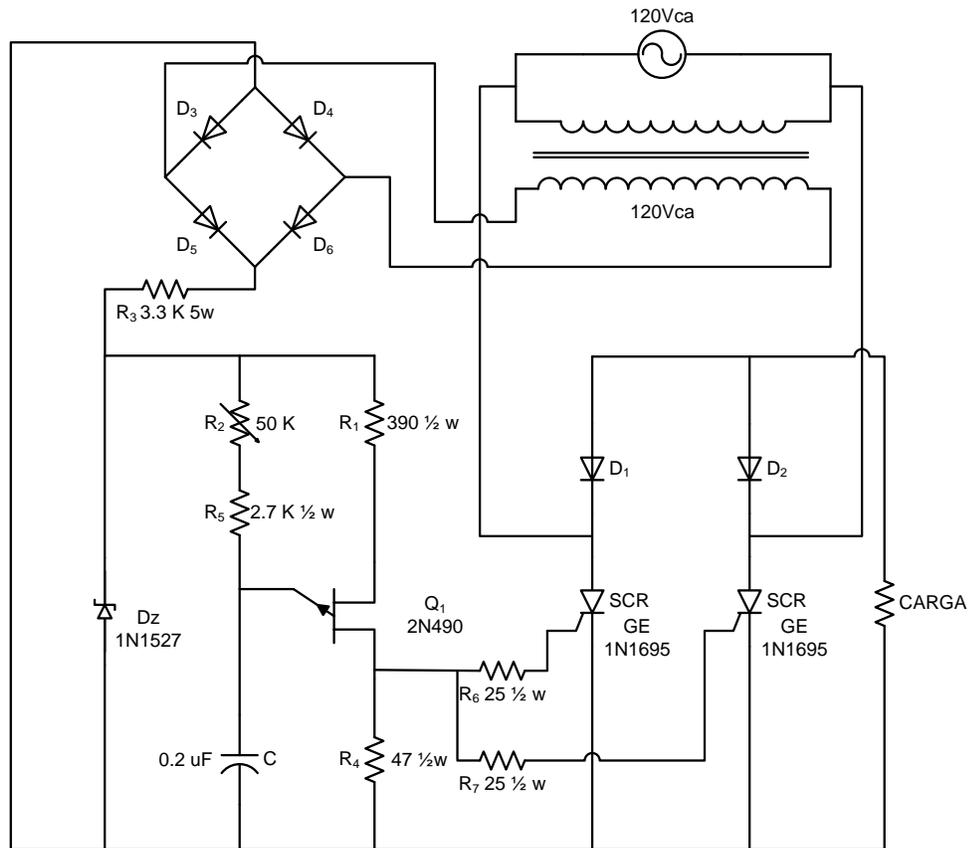


Figura 1.11 Circuito de control de fase en CD disparado por UJT.

1.2.5 Disparo por cambio de fase (CA).

En este tipo de disparo, el ángulo de disparo (o punto durante el semiciclo en donde el SCR es activado) determina la cantidad de energía que fluirá a través del ánodo y cátodo. Actúa como un switch de alta velocidad el cual generalmente se encuentra abierto al principio del semiciclo y después se cierra para permitir el flujo de energía después de que el pulso de disparo es aplicado.

La figura 1.12 muestra una señal de corriente alterna mientras se dispara la compuerta de un SCR a 45° . En un ciclo eléctrico hay 360° , 180° representan la mitad del ciclo. El número de grados que hay desde que comienza el ciclo hasta que el SCR es disparado se refiere al ángulo de

⁴ D.R. Grafham, "SCR Manual General Electric" (New York: General Electric, 1972), p. 273.

disparo y el número de grados que el SCR permanece conduciendo se refiere como ángulo de conducción.

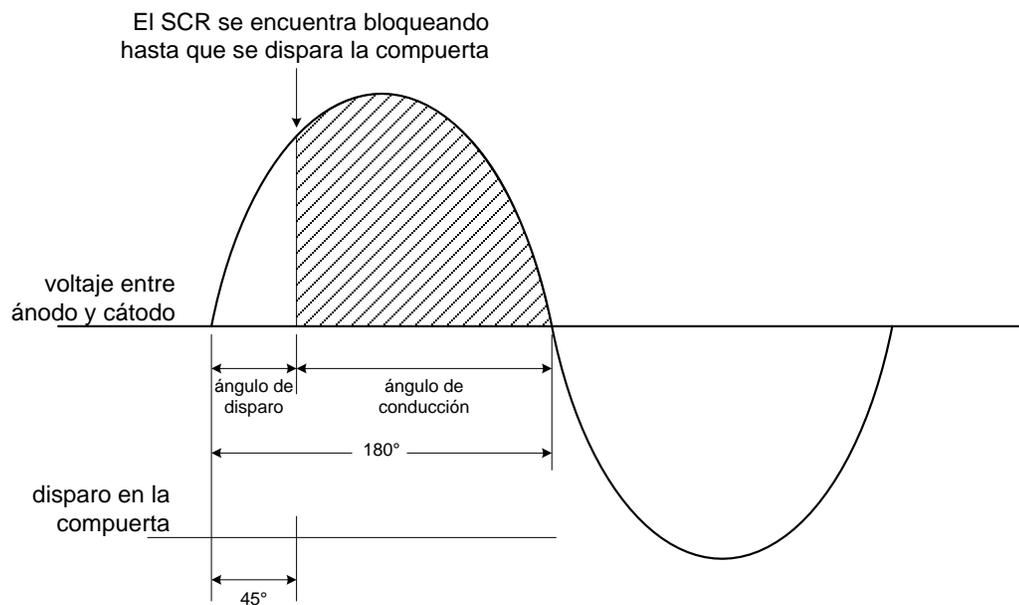


Figura 1.12 Ángulo de disparo en la compuerta del SCR.

Mientras más rápido se dispare el SCR, mayor será la cantidad de voltaje que se aplicará a la carga. La figura 1.13 muestra la comparación entre un SCR disparado a 30° y un SCR que es disparado a 90° . Se puede notar la diferencia entre las cantidades de voltaje que se suministra a la carga.

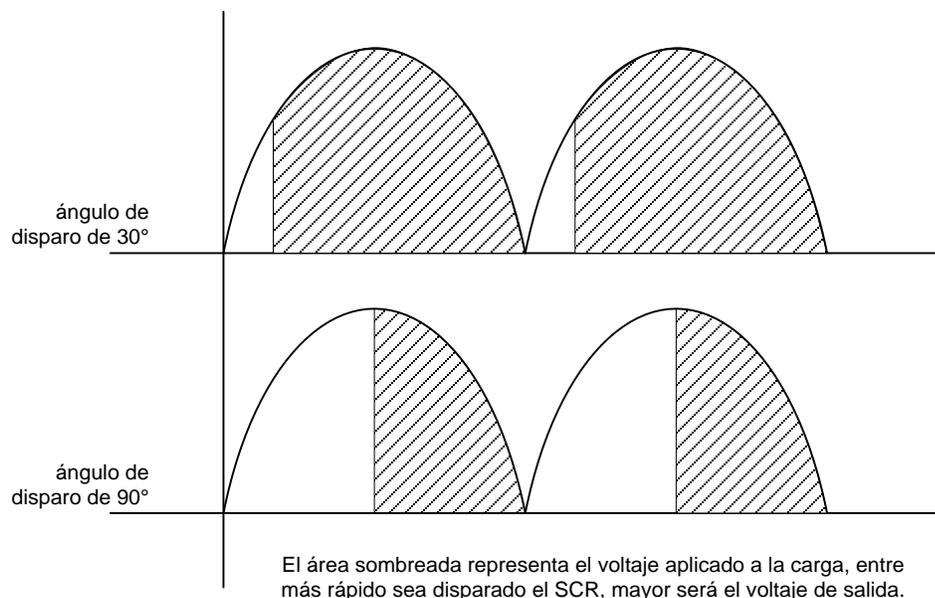


Figura 1.13 Voltaje aplicado a la carga con respecto al ángulo de disparo en el SCR.

Existe una gran variedad de circuitos para controlar la fase de disparo en un SCR, y no cabe duda que cada uno de ellos empleado en las circunstancias apropiadas tiene sus respectivas ventajas por sobre los demás.

El circuito de la figura 1.14 es un sencillo diseño teórico para el control de fase entre los puntos a y b, es un circuito RC que trabaja en corriente alterna alimentado por un transformador con derivación en el centro (12v cada derivación).

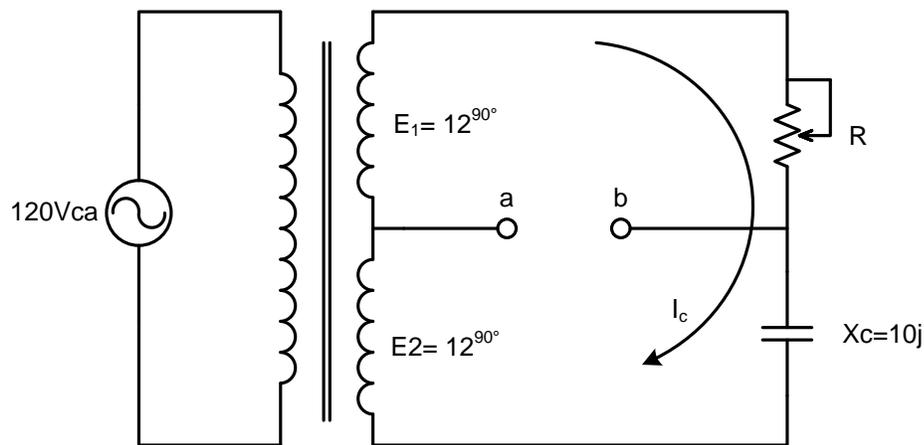


Figura 1.14 Circuito RC para el control de la fase.

Este circuito se encuentra en un par de textos, los cuales indican que solo pueden controlar hasta 170° aprox. de la señal, sin embargo no encontré el por qué el control de fase solo es de 170°, por lo que desarrollé el circuito vectorialmente por números complejos para observar su comportamiento a diferentes valores de resistencia en el potenciómetro, a 0Ω, 10Ω y 100Ω. Obteniendo los siguientes resultados:

1. Calculando el voltaje entre a y b para una resistencia con valor de 0 Ω.

$$I_c = \frac{E_1 + E_2}{X_c} = \frac{24^{90^\circ}}{-10j} = \frac{24^{90^\circ}}{10^{-90^\circ}} = 2.4^{180^\circ}$$

$$V_{X_c} = I_c \times X_c = 2.4^{180^\circ} \times 10^{-90^\circ} = 24^{90^\circ}$$

$$V_{ab} = E_2 - V_{Xc} = 12^{90^\circ} - 24^{90^\circ} = 12j - 24j = -12j$$

$$V_{ab} = 12 \angle -90^\circ$$

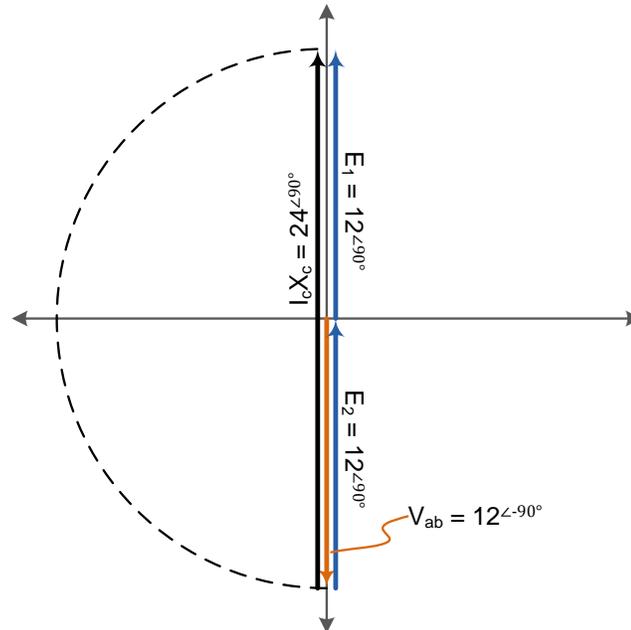


Figura 1.15 Comportamiento de la fase en el Voltaje entre los puntos a y b para un resistencia de 0 Ω .

Del primer resultado se puede señalar que el voltaje de la reactancia capacitiva refleja el voltaje de la suma de ambas derivaciones del transformador (24^{90° v), esto debido a que el valor de la resistencia es cero y no hay ninguna caída de tensión.

2. Calculado el voltaje entre a y b para una resistencia con valor de 10 Ω .

$$I_c = \frac{E_1 + E_2}{R - Xc} = \frac{24^{90^\circ}}{10 - 10j} = \frac{24^{90^\circ}}{14.1421^{-45^\circ}} = 1.6970^{135^\circ}$$

$$V_R = I_c \times R = 1.6970^{135^\circ} \times 10 = 16.9705^{135^\circ}$$

$$V_{Xc} = I_c \times X_c = 1.6970^{135^\circ} \times 10^{-90^\circ} = 16.9705^{45^\circ}$$

$$V_{ab} = -E_1 + V_R = V_R - E_1 = 16.9705^{135^\circ} - 12^{90^\circ}$$

$$V_{ab} = 12^{180^\circ}$$

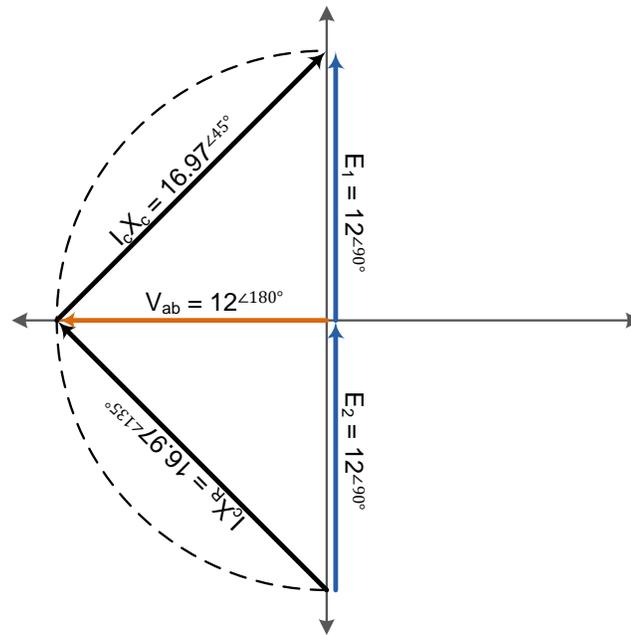


Figura 1.16 Comportamiento de la fase en el Voltaje entre los puntos a y b para un resistencia de 10 Ω .

En el segundo resultado se observa que el voltaje de la resistencia y el voltaje de la reactancia capacitiva se equilibran para alcanzar un corrimiento en la fase de 90° con respecto del primer resultado.

3. Calculado el voltaje entre a y b para una resistencia con valor de 100 Ω

$$I_c = \frac{E_1 + E_2}{R - X_c} = \frac{24^{90^\circ}}{100 - 10j} = \frac{24^{90^\circ}}{100.4987^{-5.71^\circ}} = 0.2388^{95.71^\circ}$$

$$V_R = I_c \times R = 0.2388^{95.71^\circ} \times 100 = 23.8808^{95.71^\circ}$$

$$V_{X_c} = I_c \times X_c = 0.2388^{95.71^\circ} \times 10^{-90^\circ} = 2.3880^{5.71^\circ}$$

$$V_{ab} = E_2 - V_{Xc} = 12\angle 90^\circ - 2.3880\angle 5.71^\circ$$

$$V_{ab} = 12\angle 101.42^\circ$$

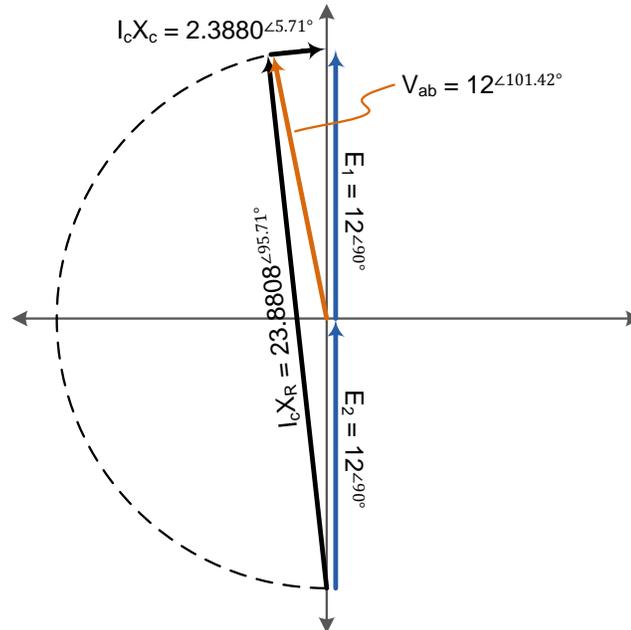


Figura 1.17 Comportamiento de la fase en el Voltaje entre los puntos a y b para un resistencia de 100 Ω .

El tercer resultado muestra que a diferencia del primer resultado, donde debido a que el valor de la resistencia es cero no hay ninguna caída de tensión, en este resultado hay un pequeño voltaje en la reactancia capacitiva sin importar que el potenciómetro este ajustado en su máximo valor, por lo que se necesitaría una resistencia infinita para reducir el voltaje de la reactancia capacitiva a cero. Esta conclusión demuestra él porque el ángulo de control de fase no es de 180° y solo alcanza los 168° en este ejemplo teórico.

Como se puede observar en la figura 1.18, el circuito de cambio de fase de la figura 1.14, solo puede controlar la fase en un rango de 168° (de -90° a 101.42° en este ejemplo). Pero como se puede observar, sigue siendo un circuito muy eficiente para controlar la fase, ya que controla el 93% de corrimiento de la misma.

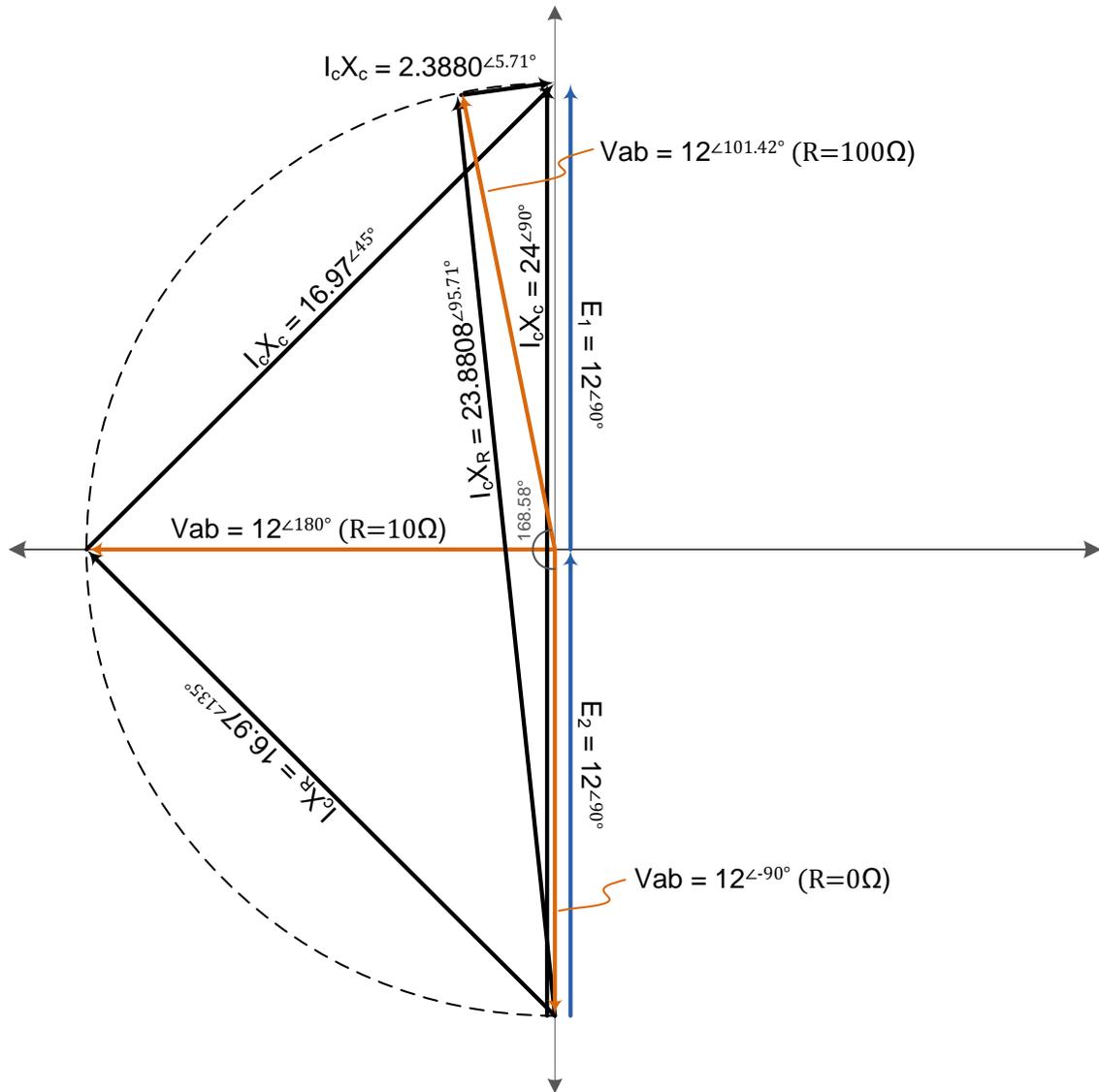


Figura 1.18 Suma de resultados de los últimos tres diagramas del comportamiento de la fase en el Voltaje entre los puntos a y b.

Al percatarme que teóricamente el circuito funcionaba con resultados eficientes, quise llevarlo a la práctica y comprobarlo en una aplicación real, por lo que monté el circuito controlador de fase en el disparo de un SCR para regular la intensidad de una lámpara (circuito dimmer). Obteniendo como resultado el circuito de la figura 1.19.

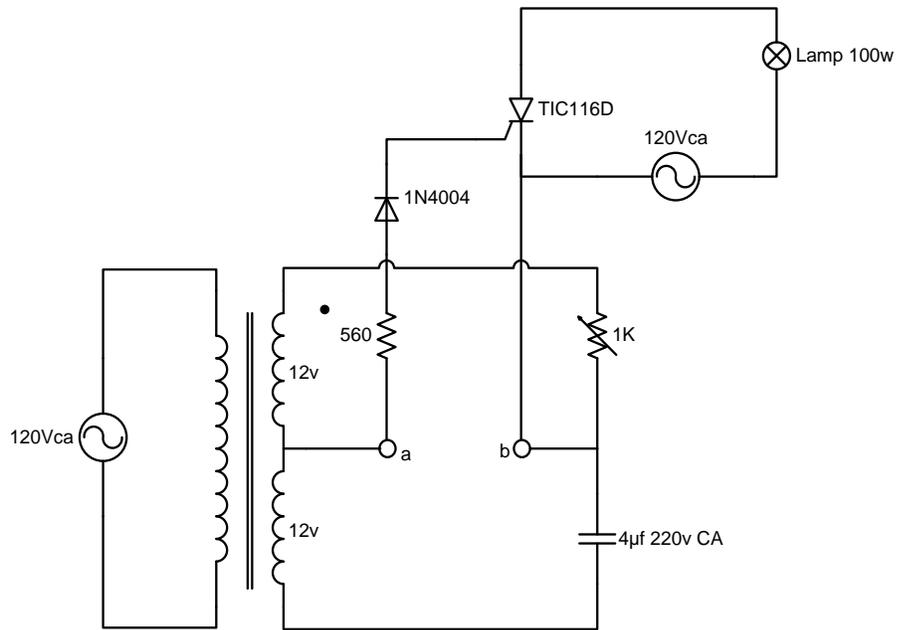


Figura 1.19 Circuito para el disparo de un SCR por control de fase de media onda.

En este circuito utilizo un SCR similar al que uso en el modulo de potencia, es el TIC116D, un capacitor de c.a. de $4\mu\text{F}$ a 220v y un potenciómetro de alambre de $1\text{K}\Omega$.

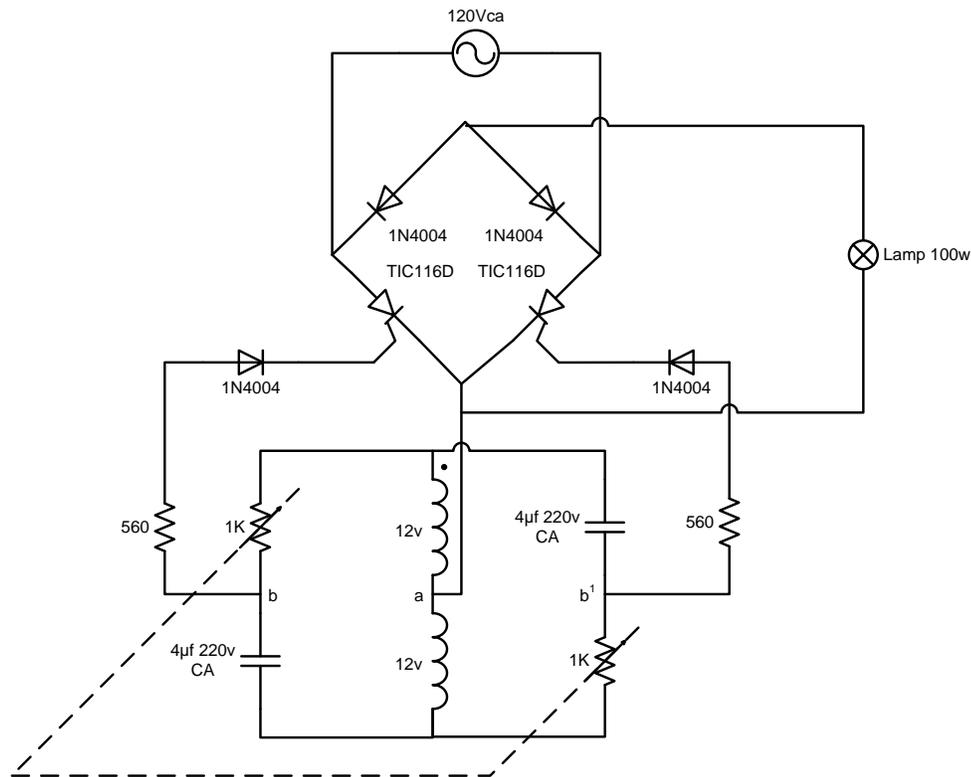
El circuito opera a media onda de la señal de alimentación, y como se esperaba, la lámpara muestra un cambio en su intensidad, resultado del control de la fase al variar el potenciómetro, como se muestra en la tabla 1.1. Sin embargo como puede observarse en la tabla el rango de intensidad que se puede obtener en la lámpara está limitado a la mitad debido a que es un circuito de media onda.

Ajuste del Potenciómetro	V_{a-b}	$V_{\text{lámpara}}$
$1\text{K}\Omega$ (max)	23v	55v
0Ω (mín)	17.5v	2.15v

Tabla 1.1 Comportamiento del cambio de fase en el circuito 1.17 al ajustar el potenciómetro.

No podía considerar completa la comprobación del circuito para controlar la fase de la figura 1.14 sin obtener su comportamiento al trabajar en onda completa. Por lo que simplemente dupliqué el circuito utilizando dos SCR's, dos capacitores de c.a. y un potenciómetro de alambre en tándem (para controlar ambos circuitos simultáneamente y por lo tanto controlar el

corrimiento de fase de ambos semiciclos). El circuito resultante es el de la figura 1.20 y a pesar de que tiene un arreglo diferente del anterior que trabaja a media onda, se puede distinguir la configuración inicial del circuito controlador de fase que se ha analizado desde la figura 1.14.



1.20 Circuito para el disparo de un SCR por control de fase de onda completa.

Finalmente en este circuito de onda completa se aprecia el rango completo de control de fase que se refleja en el cambio de intensidad luminosa en la lámpara como se observa en la tabla 1.2. Hay que recordar que teóricamente el voltaje entre los puntos a y b (V_{a-b}) debería mantener el voltaje de las derivaciones del transformador (12v teóricamente, pero depende del voltaje de alimentación del transformador), independientemente del ajuste en el potenciómetro, tal y como se observó en los diagramas 1.15 – 1.18 obtenidos de los cálculos correspondientes (ya que lo que se controla es únicamente la fase), sin embargo esos cálculos se obtuvieron sin considerar carga alguna entre los puntos a y b. Y como se observa en las tablas 1.1 y 1.2, hay una ligera variación en el voltaje V_{a-b}

debido a la carga aplicada a dichos puntos debida a los circuitos de disparo de los SCR's.

Cabe señalar que este tipo de circuitos son muy costosos para utilizarlos en una aplicación como lo es un control dimmable de una lámpara, sin embargo es un método sencillo para demostrar la precisión de control que se puede obtener en cualquier otra aplicación.

Potenciómetro	V_{a-b}	$V_{lámpara}$
1K Ω (max)	17.5v	110v
0 Ω (mín)	9.6v	9.6v

Tabla 1.2 Comportamiento del cambio de fase en el circuito 1.18 al ajustar el potenciómetro.

1.3 LA INTERFACE DEL MODULO DE POTENCIA.

Tomando en cuenta que los motores que tenía a mi disposición eran los motores de corriente directa de 12V, la forma más apropiada para controlarlos fueron los SCR's, debido a sus cualidades como controlar grandes cantidades de corriente entre su ánodo y cátodo al aplicar una pequeña cantidad de corriente en su compuerta, cuentan con una gran variedad de formas de dispararlos, son muy económicos, entre otras características.

Aprovechando la fuente de alimentación de los motores, utilicé un disparo por fase en los SCR's como se muestra en la figura 1.21, este tipo de disparo permite ajustar hasta cierto limite el ángulo de conducción del SCR debido a la variación de la corriente de compuerta del SCR por medio de la variación de la resistencia R de acuerdo a los niveles máximo y mínimo de corriente en la compuerta permitidas dentro de las características del fabricante. Sin embargo la finalidad del control de los motores de C.D. es el máximo aprovechamiento de la energía proporcionada por la fuente. Por lo que el valor de la resistencia se fijó de acuerdo al pronto disparo de la compuerta del SCR (recordando el aprovechamiento de energía que se muestra en la figura 1.12).

Este tipo de disparo por fase cuenta con varias ventajas:

- No requiere de una fuente independiente para el disparo de la compuerta del SCR (se dispara con la misma fuente de alimentación de los motores).
- Se asegura la desactivación del SCR al interrumpir la señal de disparo en la compuerta por medio del optoacoplador, el cual funciona como un switch. Provocando que el SCR se desactive una vez que el semiciclo entre ánodo y cátodo llegue a cero.
- Asegura el propio disparo del SCR, ya que si existe alguna variación de voltaje en la fuente de los motores (de donde también proviene la alimentación para la compuerta del SCR), seguirá disparando la compuerta mientras alcance su voltaje requerido de 1.5v, por lo que es un disparo confiable, el disparo del SCR se muestra en la figura 1.22.

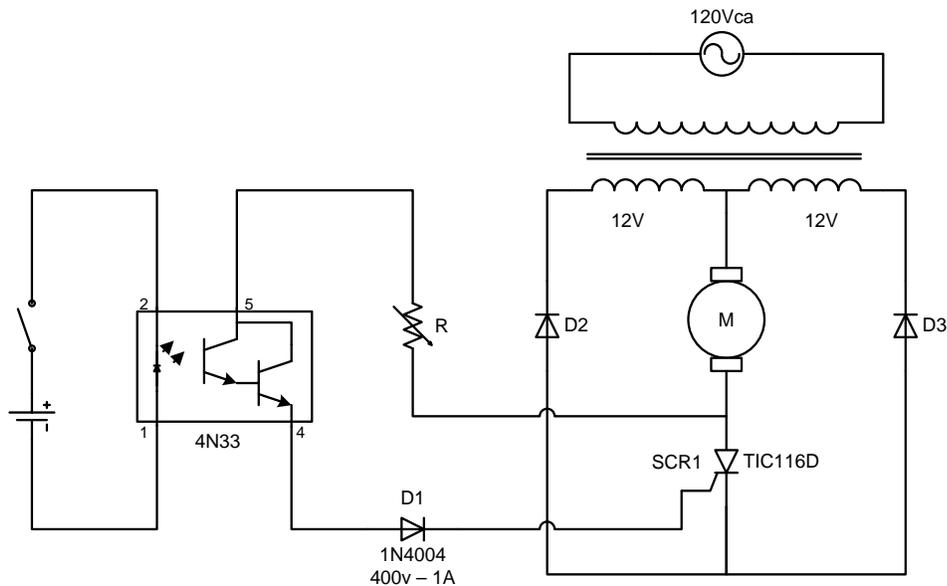


Figura 1.21 Principio del circuito de disparo por fase de los SCR's del modulo.

Teniendo en cuenta que el circuito tendría conexión con la PC añadí un optoacoplador, el cual es un dispositivo que permite la transmisión de señales entre dos circuitos, mientras que mantiene a los dos circuitos eléctricamente aislados uno del otro. Consta de un foto emisor como puede ser un diodo led o un diodo infrarrojo, y de un foto receptor como un fototransistor o un foto triac. Es por ello que por características de aislamiento es ideal para formar parte de la interface a la computadora.

Este modulo de potencia fue originalmente diseñado para operar al recibir los pulsos de mando provenientes del puerto paralelo de la PC controlados por el software correspondiente (tomando en cuenta que ahora también cuenta con la opción de ser controlado por medio de microcontroladores Picaxe como se explica en el capítulo 5. Por lo que el diseño del circuito implica las características eléctricas de salida del puerto paralelo, como lo son los niveles de voltaje de salida V_{OH} y V_{OL} (“voltaje de salida alto” y “voltaje de salida bajo” respectivamente).

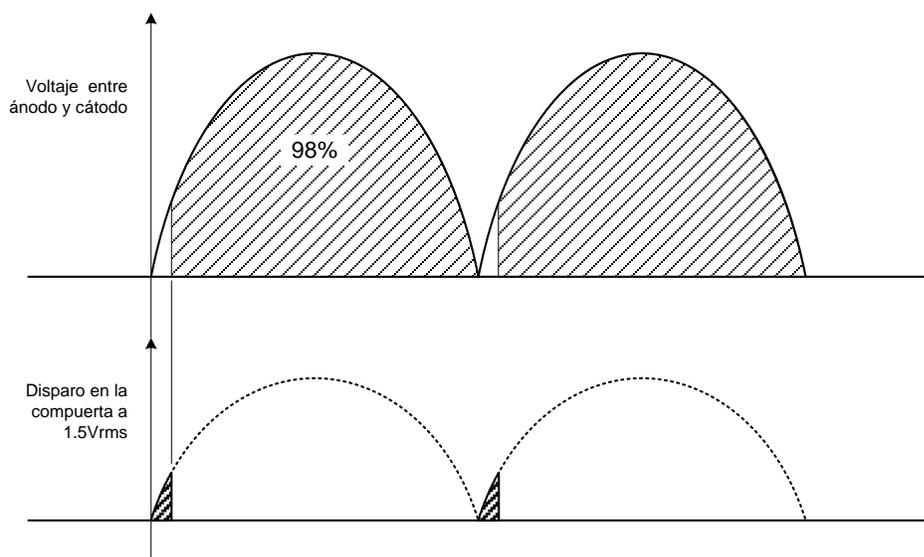


Figura 1.22 Disparo por fase del SCR.

1.3.1 La transmisión paralela y el puerto paralelo.

En la transmisión paralela, todos los bits son transmitidos al mismo tiempo vía paralela en un número apropiado de líneas de transmisión. La aplicación de este tipo de transmisión es para cortas distancias, donde se requieren altas tasas de transmisión de datos sin la necesidad de métodos de conversión que requieran un gran número de componentes. La transmisión de un byte (8 bits) requiere un mínimo de ocho líneas y una referencia de potencial como se muestra en la figura 1.23. La instalación es cara por lo que es recomendable para cortas distancias.

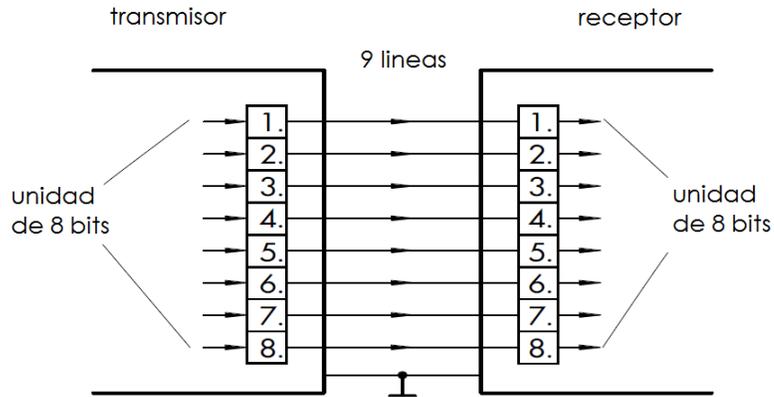


Figura 1.23 Transmisión de datos vía paralela.

El puerto paralelo generalmente era utilizado como interface entre la PC y las impresoras a través de un cable conector DB25 de 25 pines como se muestra en la figura 1.24, viendo el conector de frente y con la parte que tiene más pines hacia arriba, se enumera de menor a mayor de derecha a izquierda en ambas filas de pines (1 a 13 arriba y 14 a 25 abajo). Sin embargo se fueron añadiendo otro tipo de dispositivos como escáners, cámaras web, unidades lectoras de CD, etc. Pero con el paso del tiempo y debido a las características del puerto, se utilizó como interface entre la PC y circuitos diseñados por particulares para el control por medio de la PC.

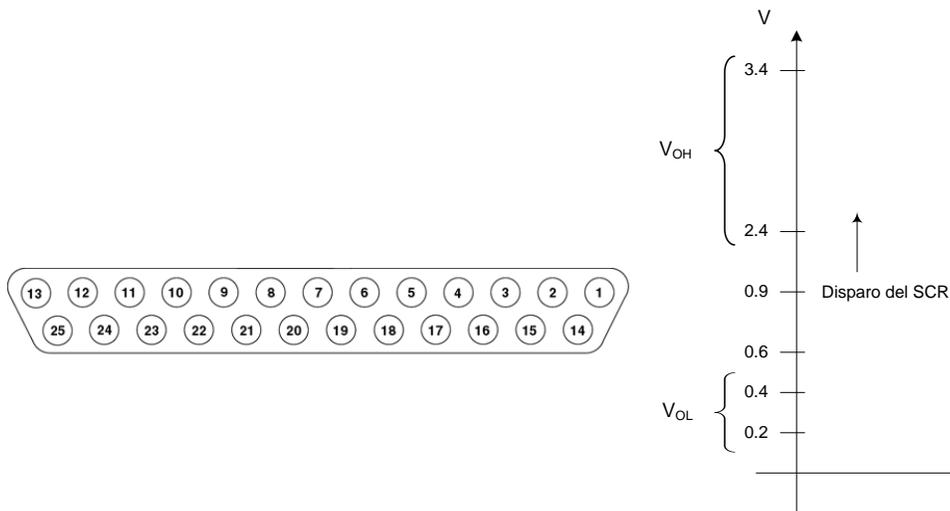


Figura 1.24 Niveles de voltaje de salida del puerto paralelo de la PC y numeración de sus pines

Los pines que se utilizan para la transmisión de datos son los numerados del 2 al 9 y responden a la lógica TTL, es decir que un

voltaje alto representa un “1” lógico y un voltaje bajo representa un “0” lógico. En la figura 1.24 se muestran los niveles interpretados por el puerto paralelo.

Una vez que se tomaron en cuenta las características eléctricas de salida del puerto paralelo, es posible completar el circuito de interface para el disparo de los SCR's como se muestra en la figura 1.25. Recordando que el puerto paralelo proporciona de 2.2v a 3.6v como niveles de voltaje de salida altos (“1” lógico), el transistor Q_1 (alimentado por una fuente regulada de 5Vcd) tipo NPN con emisor a tierra y conectado a través del diodo D_2 de protección al pin del puerto paralelo, se encarga de disparar 2 diodos led, de los cuales uno sirve como señalización visual de funcionamiento y el otro es el diodo led interno del optoacoplador 4N33. A su vez, el fototransistor tipo Darlington de dicho optoacoplador opera como switch para disparar la compuerta del SCR, el cual como lo mencionamos anteriormente es disparado por fase.

Como se puede observar en el circuito de la figura 1.25, el circuito de interface representado funciona para disparar un solo SCR y por lo tanto para un solo sentido de giro del motor, por lo que se requiere de dos circuitos de interface para disparar ambos SCR's que controlen ambos sentidos de giro de un motor. Esto significa que para operar los motores que integran el brazo robótico, así como su efector, son necesarios ocho circuitos de interface como el de la figura 1.25 (los cuales están incorporados dentro del modulo de potencia).

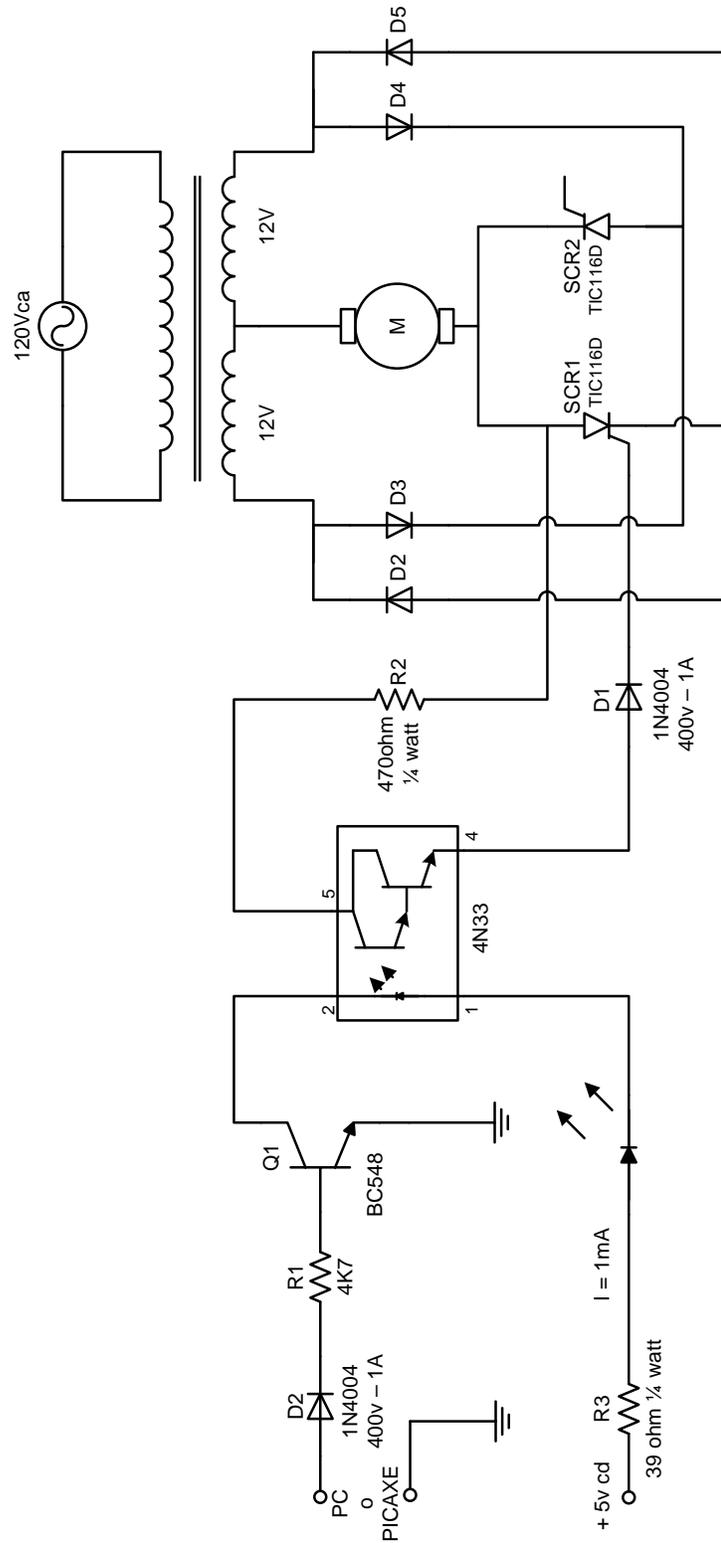


Figura 1.25 Circuito de interface para el disparo de los SCR's desde la PC.

Cada motor cuenta con su fuente de corriente directa integrada por un transformador de 24v con derivación en el centro, proporcionando dos fuentes de corriente directa de 12v, una para cada sentido de giro del motor. El hecho de que cada motor cuente con su fuente de alimentación independiente a los demás motores, permite que los mandos del brazo sean efectuados de manera simultánea.



Figura 1.26 Modulo de potencia del Proyecto.

Capítulo 2.

FUNDAMENTOS BÁSICOS DE ROBÓTICA INDUSTRIAL.

2.1 RESEÑA DE LA ROBÓTICA.

En los últimos años nuestro concepto y uso de los robots ha evolucionado de la ciencia ficción a la realidad de dispositivos electromecánicos controlados por computadora dentro de una amplia variedad de ambientes. Se ha vuelto convencional ver robots manipulados ser usados para soldar y pintar estructuras de automóviles en líneas de ensamblaje, insertando circuitos integrados en tarjetas de circuitos impresos, robots en el área militar como vehículos de reconocimiento no tripulados, robots que realizan operaciones quirúrgicas en los seres humanos e incluso robots que cosechan frutas en el campo de la agricultura y en el área de la automatización industrial en general. Por otra parte hay una gran variedad de robots antropomórficos, debido al constante intento por construir robots humanoides, es por ello que generalmente las personas tienen el concepto de los robots, como replicas electromecánicas de nosotros mismos. Mientras que es muy probable que pasen años en que veamos perfeccionados este tipo de robots, es justo decir que se ha hecho un gran progreso en la introducción de robots con formas amorfas ante la vista común, pero que cumplen con éxito tareas requeridas en diferentes circunstancias.

Se puede decir que la impaciencia que el ser humano ha tenido por acelerar la investigación y construcción de nuevos y más sofisticados robots, es por la comparación inherente que hacemos con el increíble funcionamiento de nuestros propios cuerpos como sistemas biológicos.

2.2 PARTES DEL ROBOT

En la industria existen robots de diferentes formas y tamaños. Los robots consisten en cierto número de componentes que trabajan en un conjunto: el controlador, el manipulador, un efector final, fuente de poder y medios de programación, la relación de estos componentes se muestra en la figura 2.1.

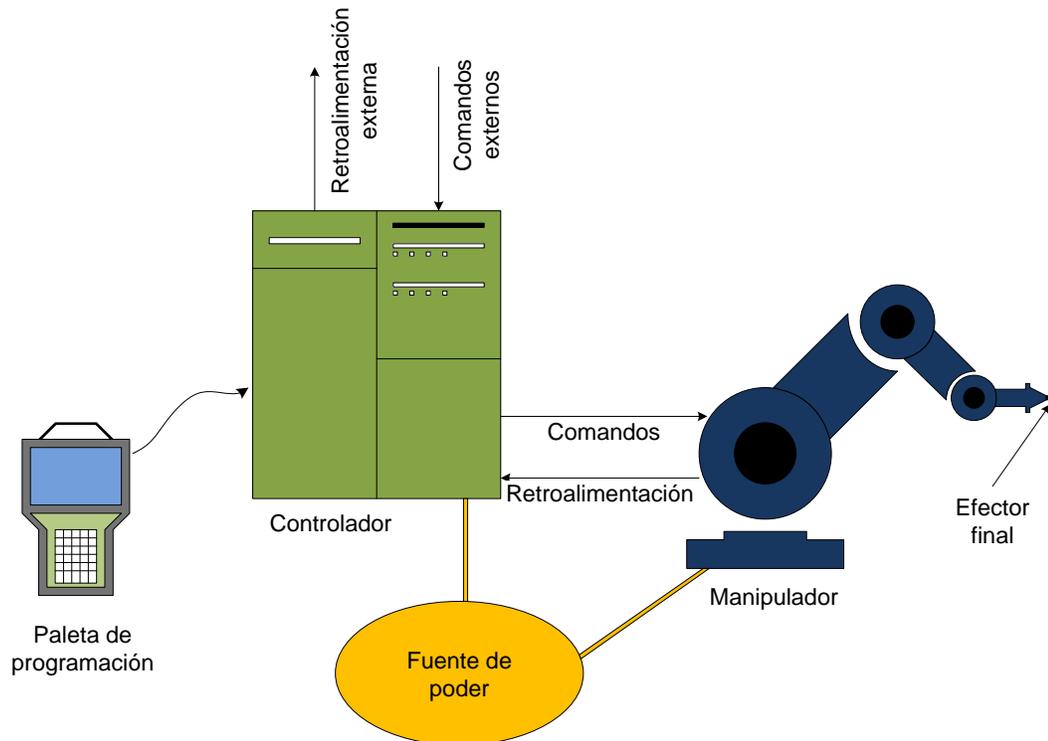


Figura 2.1 Componentes que generalmente conforman a un robot industrial.

2.2.1 El controlador.

El control es la parte del robot que coordina todos los movimientos del sistema mecánico, también recibe las entradas de los sensores que controlan el ambiente inmediato del robot. El corazón del controlador del robot es generalmente la unidad de procesamiento unida a las entradas/salidas y a los dispositivos de monitoreo. Los comandos emitidos por el controlador activan los mecanismos de control de movimiento y de potencia, que pueden consistir en otros controladores, amplificadores y actuadores. Un actuador es un motor o válvula, que convierten la potencia en movimientos del robot. Estos movimientos son iniciados por una serie de instrucciones, llamados programas, los cuales se almacena en la memoria del controlador.

El controlador contiene tres niveles de control jerárquico. El control jerárquico asigna niveles de organización a los controladores dentro de un sistema robótico. Cada nivel envía señales de control al nivel inferior y envía señales de retroalimentación al nivel superior. Los niveles se vuelven más elementales mientras se aproximan más al actuador. Cada nivel depende de instrucciones de un nivel por encima de él figura 2.2.

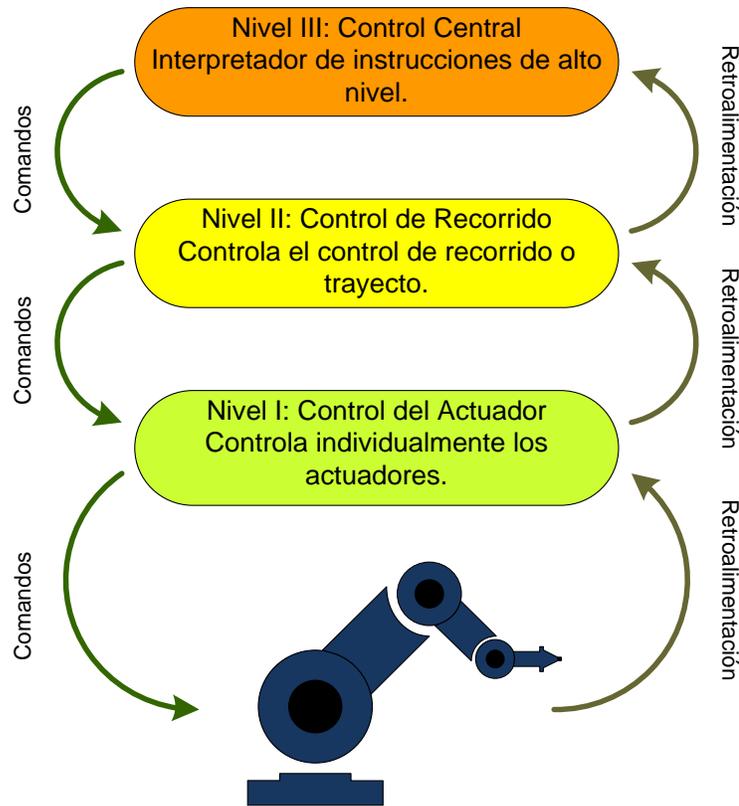


Figura 2.2 Los tres niveles jerárquicos básicos de control.

Nivel I – Control del Actuador. Es el nivel más elemental en donde se separan los movimientos del robot a través de diferentes planos, como el plano X, Y y Z.

Nivel II – Control de Recorrido o trayecto. El nivel del control de recorrido (nivel intermedio) coordina los movimientos separados en los diferentes planos a los que se refiere el nivel I y los interpreta en la trayectoria deseada (recorrido).

Nivel III – Control Central. La función primaria del más alto nivel de control es interpretar las instrucciones escritas por el programador humano con relación a las tareas requeridas. Las instrucciones se combinan diferentes señales provenientes de las retroalimentaciones de los sensores y son

traducidas por el controlador en instrucciones más elementales que el nivel II pueda entender.

2.2.2 Manipulador.

El manipulador consiste en segmentos articulados que producen una serie de movimientos, permitiendo que el robot cumpla con su tarea. El manipulador es el brazo del robot que debe mover materiales, controlar herramientas, etc. Un manipulador puede identificarse por el método de control, fuente de poder, los actuadores de las articulaciones y otros factores. Estos factores ayudan a identificar el mejor tipo de robot para la tarea requerida. Por ejemplo, uno no usaría un robot con actuadores eléctricos en un ambiente donde existan vaporizaciones de combustibles con el posible riesgo de generar una chispa que cause una explosión.

El manipulador está conformado por una serie de segmentos y articulaciones, que en muchas ocasiones tienen bastante semejanza con el brazo humano. Una articulación une dos segmentos y genera un movimiento entre los dos. Las articulaciones proveen de movimientos lineales (en línea recta) o rotatorios (circular) como se muestra en la figura 2.3.

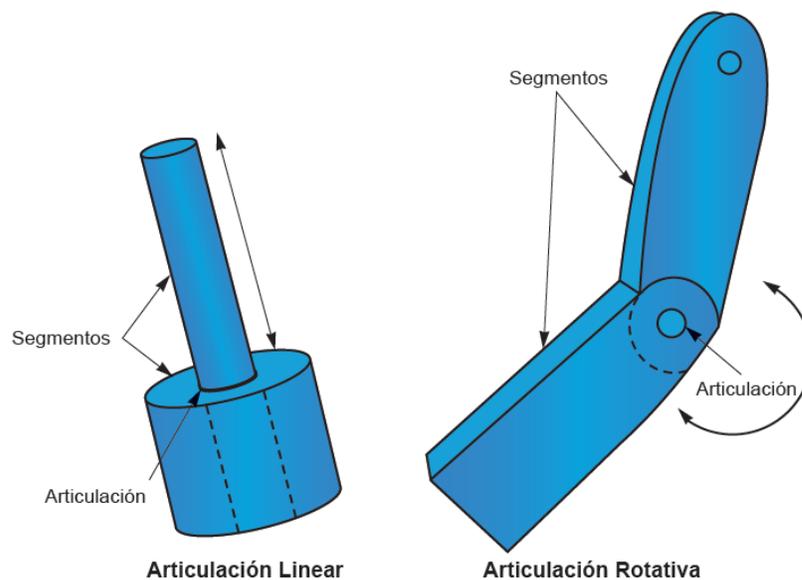


Figura 2.3 Articulaciones lineales y rotativas, las más comúnmente encontradas en los robots.

Los músculos del cuerpo humano suministran la fuerza que mueve las diferentes articulaciones. De forma similar, un robot usa actuadores para

mover su brazo a través de trayectorias programadas y mantiene rígidas sus articulaciones una vez que se alcanza la posición correcta.

Por lo tanto existen dos tipos de movimientos que los actuadores pueden proporcionar: lineal y rotatorio. Los actuadores lineales proveen de movimientos a través de una línea recta, extienden o retraen las cargas adheridas a ellos. Los actuadores rotatorios proveen de rotación, moviendo sus cargas en un arco o circularmente.

2.2.3 El Efecto Final.

El efecto final es la mano del robot, o la herramienta al final del brazo. Es un dispositivo ajustado a la muñeca del manipulador con el propósito de sujetar, transportar, maniobrar o realizar operaciones en una pieza de trabajo (barrenar, soldar, etc.). El efecto final es uno de los componentes más importantes de un sistema robótico. El espacio dentro del alcance del efecto final del robot es llamado volumen de trabajo.

2.2.4 Fuente de Poder.

La fuente de poder se encarga de suministrar energía al controlador y a los actuadores. Puede convertir voltaje C.A. a C.D., o puede ser una bomba o un compresor para proveer potencia hidráulica o neumática. Las fuentes de energía más comunes son la eléctrica, la hidráulica y la neumática.

2.2.5 Formas de programación.

El término de programación es usado como la grabación de movimientos en la memoria del controlador. Un robot puede ser programado usando alguno de los diferentes métodos disponibles. Los controladores pueden ser programados por medio de otros computadores a distancia, o por medio de paneles portátiles integrados a los controladores (paletas programadoras), usados para reprogramar secuencias de movimiento dentro de la memoria del robot (enseñarle movimientos al robot).

Este tema se abarca más adelante en los capítulos 4 y 5.

2.2.6 Actuadores.

Un actuador es un mecanismo que está diseñado para convertir energía de algún tipo, en movimiento o fuerza mecánica de forma controlada. En la robótica, los actuadores son los componentes que causan que un robot se mueva (motores, pistones, etc.). Típicamente los actuadores son de desplazamiento lineal o rotatorio y pueden ser eléctricos, neumáticos o hidráulicos.

Los actuadores son usados para accionar y controlar las articulaciones del manipulador. Sin embargo algunas articulaciones no son accionadas por ningún actuador, es decir que son pasivas. Por ejemplo, algunas articulaciones pueden ser bisagras que solo articulan cuando por causa de otra articulación con actuador producen el movimiento de la primera al estar unida por un segmento, tal es el caso de los manipuladores hidráulicos (figura 2.4).

2.3 TIPOS DE ACTUADORES.

Un método común de clasificar a los robots es por el tipo de actuadores que utilizan:

- ▲ Actuadores hidráulicos
- ▲ Actuadores neumáticos
- ▲ Actuadores eléctricos

2.3.1 Actuadores Hidráulicos.

Muchos de los primeros robots creados utilizaban sistemas hidráulicos como actuadores. Estos sistemas hidráulicos usan fluidos a presión y consisten en una bomba conectada a un tanque de almacenamiento, válvulas de control y un actuador hidráulico. Los sistemas hidráulicos proporcionan tanto movimientos lineales como rotacionales utilizando arreglos mucho más simples que los usados por los sistemas eléctricos convencionales (motores con sus reductores).

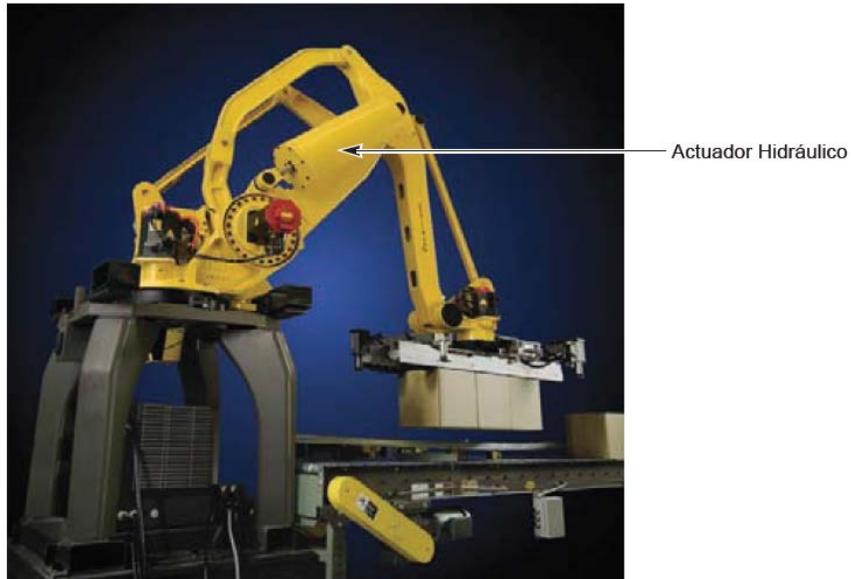


Figura 2.4 Un largo actuador hidráulico proporciona un movimiento de arriba – abajo al brazo de este robot industrial (FANUC Robotics).

Los actuadores hidráulicos cuentan con varias ventajas. Proporcionan un control preciso de movimiento en un amplio rango de velocidades. Pueden soportar grandes cargas al final del brazo del manipulador, se pueden utilizar en un ambiente con materiales inflamables o explosivos. No se dañan fácilmente cuando se detienen súbitamente mientras transportan cargas pesadas (contragolpe por inercia). Por otra parte los sistemas hidráulicos como actuadores son costosos, así como también en su mantenimiento. No son eficientes con respecto a la energía de alimentación, son más ruidosos que los actuadores eléctricos y no se recomiendan en lugares que requieran de un medio libre de contaminantes, esto debido a la posibilidad de fugas de líquido hidráulico.

2.3.2 Actuadores Neumáticos.

Los sistemas neumáticos utilizan actuadores controlados por aire a presión. Y como se considera que el aire es un fluido, muchas de los principios que se tienen para los sistemas hidráulicos, aplican para los actuadores neumáticos. Los motores neumáticos e hidráulicos son muy similares y la mayoría de las plantas industriales cuentan con sistemas de aire comprimido, el aire es económico y una fuente de energía segura. Esto hace que la instalación de robots que usan actuadores neumáticos sean

más sencillos de controlar y menos costosos que los robots con actuadores hidráulicos. Para aplicaciones de recoger – colocar pesos ligeros que requieren de velocidad y precisión, los robots neumáticos son ideales.

Los actuadores neumáticos trabajan a altas velocidades y trabajan con cargas pequeñas y medianas. Son económicos en su operación y mantenimiento, también pueden trabajar en atmosferas explosivas o inflamables. Sin embargo, debido a que el aire es compresible, el posicionamiento preciso requiere de componentes adicionales para igualar el suave control de los sistemas hidráulicos. Los robots que usan actuadores neumáticos son ruidosos y vibran cuando los cilindros de aire y los motores se detienen, también es muy difícil mantener el aire limpio y seco como lo requiere el sistema de control.

2.3.3 Actuadores Eléctricos.

Tres tipos de motores son los que se usan comúnmente como actuadores eléctricos: servo-motores de ca, servo-motores de c.d. y motores de pasos. Los dos primeros son utilizados para controlar posiciones exactas en los robots. Muchos de los últimos modelos de robots utilizan servo-motores en vez de sistemas hidráulicos o neumáticos. Los robots de pequeña y mediana escala utilizan servo-motores de c.d. Debido a su alto torque, los servo-motores de c.a. son usados en robots de trabajo pesado.



Figura 2.5 Este robot soldador industrial de alto desempeño utiliza dos servo motores de corriente alterna para proporcionar movimiento al brazo. Motoman Robotics Europ AB.

Los robots que utilizan actuadores eléctricos requieren de menor superficie en el suelo para su operación, son silenciosos, simples y pueden ser usados en lugares donde se pretenda preservar el aire libre de contaminantes. Sin embargo, sus engranajes convencionales causan problemas como contragolpe por inercia, fricción y desgaste. Estos problemas pueden llegar a provocar inexactitud, una respuesta dinámica pobre, realizar mantenimiento con regularidad y limitaciones de velocidad. Las cargas que son lo suficientemente grandes como para parar el motor pueden causar grandes daños.

La mayoría de los movimientos rotatorios de los actuadores eléctricos requieren de un reductor para proporcionar al manipulador la velocidad y torque que requiere.

Para el ensamble del brazo robótico de este proyecto utilicé como actuadores, motores de corriente directa, por lo que a continuación se describen brevemente algunos de los motores de corriente directa más usados como actuadores para robots.

2.4 RESEÑA DE LOS MOTORES DE CORRIENTE DIRECTA.

Los motores de corriente directa son extensamente usados en aplicaciones donde se requiere un ajuste variable de velocidad y en aplicaciones donde se requiere un par grande.

Existen varios tipos de motores de corriente directa, de entre los más comunes se encuentran:

2.4.1 Motores shunt (paralelo).

Los motores shunt se emplean cuando se requiere una velocidad prácticamente constante, como en los ventiladores, taladros y bombas. No hay ningún tipo de motor que se acomode mejor que el shunt a la variación regulada de la velocidad. Por consiguiente se emplea también en los casos en que se requiere una velocidad variable a voluntad. Si se requiere ajustar la velocidad se la puede llevar al valor que se desee y luego mantenerla invariable, con este valor al variar la carga. El torno mecánico es un excelente ejemplo de funcionamiento con velocidad regulable; para cada clase de trabajo se requieren velocidades diferentes.

2.4.2 Motores Serie.

Los motores serie se utilizan para los casos en que se exige un gran par de arranque, como en los tranvías, trolebuses y grúas. Además del gran par de arranque, tienen otra característica que los hace especialmente adecuados para la tracción. Supongamos que se usa un motor serie en un tranvía, cuando este asciende por una rampa, el motor serie va despacio debido a la mayor intensidad de corriente demandada, y por lo tanto desarrolla un par mayor a velocidad menor. La reducción de la velocidad permite al motor desarrollar un gran par con un aumento moderado de potencia.

2.4.3 Motor Compound (compuesto).

Se puede decir que el sistema compound es una combinación del motor serie y shunt.

El motor compound acumulativo desarrolla un par elevado cuando se aumenta súbitamente la carga. Tiene también una velocidad definida cuando funciona en vacío y no peligra que se desboque cuando se suprime la carga. La aplicación del motor compound acumulativo se encuentra principalmente en el accionamiento de máquinas destinadas a soportar súbitamente cargas intensas, como sucede con las prensas, grúas tracción y laminadores. Este tipo de motor se emplea también cuando se requiere un par de arranque elevado, y no se puede utilizar adecuadamente un motor serie normal. Otra ventaja del motor tipo compound acumulativo sucede cuando se presentan cargas bruscas, automáticamente experimenta una sensible disminución de velocidad.

El motor compound diferencial mantiene una velocidad prácticamente constante o con tendencia a aumentar cuando la carga se incrementa. Esta característica de velocidad se obtiene debido a la disminución en el aumento del par al aumentar la carga. Estos motores se emplean en los casos en que se desee una velocidad muy constante. Como los motores shunt ya tienen una velocidad prácticamente constante, se presentan pocas ocasiones de emplear el compound diferencial.

2.4.4 Motor de Imanes permanentes.

La configuración predominante utilizada en aplicaciones alta demanda (arranque – paro) es el motor de imán permanente. Ya que el campo del estator es generado por los imanes permanentes, no se requiere energía alguna para generar su campo. Los imanes proveen de un constante flujo de campo a cualquier velocidad, por lo que las curvas de torque-velocidad son lineales.

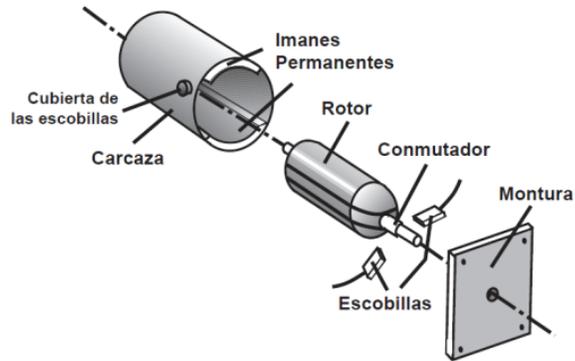


Figura 2.9 Diagrama del motor de corriente directa de imanes permanentes.

Este motor tiene un comportamiento lineal y predecible, tiene una estructura más pequeña y ligera comparada con otro tipo de motores, proporcionando un rápido posicionamiento.

Los motores utilizados en este proyecto son motores de imanes permanentes de 12v c.d. que originalmente fungían como motores para limpiaparabrisas de automóviles, similares al de la figura 2.10. Estos motores fueron los que pude adquirir con mayor facilidad debido a su bajo costo y abundancia en centros de autopartes.



Figura 2.10 Motorreductor de c.d. con reductor corona sin fin.

Este tipo de motor esta acoplado a un reductor tipo corona sin fin para reducir su velocidad y así proporcionar una velocidad moderada.

2.5 GENERALIDADES DE LOS ACOPLAMIENTOS POR ENGRANES

2.5.1 Engranés rectos

El engrane recto es el tipo de engranes generalmente más usados como reductores y también los más conocidos. Se caracterizan por tener posicionados los dientes del engrane en forma paralela al eje, son usados en aplicaciones de baja a mediana velocidad. Cuentan con una eficiencia del 98 – 99%.

Ventajas: Son fáciles de conseguir, eficientes y nos son costosos.

Desventajas: Generan ruido, la disposición de los engranes es muy limitada

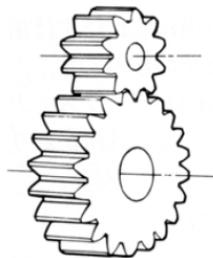


Figura 2.11 Engranés rectos.

2.5.2 Engranés Helicoidales

Los engranes helicoidales son similares a los engranes rectos, excepto que sus dientes están maquinados en un ángulo de inclinación con respecto al eje, en vez de ser dientes paralelos al eje como el engrane recto.

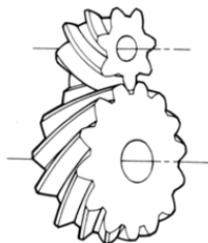


Figura 2.12 Engranés helicoidales.

El resultado es que el área de contacto que tiene el engrane es mayor debido a que los dientes son más largos (por su inclinación) a diferencia de

un engrane recto del mismo diámetro. Provocando que: La fuerza del diente es mayor debido a la longitud del mismo diente.

Debido a que tiene una mayor superficie de contacto un engrane helicoidal puede soportar una mayor carga que un engrane recto.

Los engranes helicoidales pueden ser usados para engranar dos ejes que no son paralelos, sin embargo se usan principalmente en ejes paralelos. Una aplicación especial de los engranes helicoidales es en engranajes cruzados, en donde los dos ejes se encuentran perpendicularmente uno del otro. Cuentan con una eficiencia del 98 – 99%.

Ventajas: El área de contacto en los dientes es muy eficiente, pueden ser usados no solo en ejes paralelos, sino en distintos ángulos e inclusive en ejes perpendiculares. Soportan mayores cargas que los engranes rectos.

Desventajas: La mayor desventaja de los engranes helicoidales es su alto costo, dependiendo de la disposición del ángulo de los engranes se pueden generar fuerzas axiales en él mismo forzando a colocar rodamientos que absorban dichas fuerzas.

2.5.3 Engranes cónicos.

Se usan para transmitir potencia entre ejes concurrentes (ejes de intersección perpendicular). Los dientes del engrane están dispuestos de forma cónica en la superficie. Pueden operar a altas velocidades con altas cargas. Una de sus principales y más común de las aplicaciones es como sistema diferencial en vehículos automotores. Cuentan con una eficiencia del 96 – 98%.

Ventajas: Son una buena opción en sistemas de ejes concurrentes.

Desventajas: No se pueden usar en ejes paralelos. Se vuelven ruidosos a altas velocidades.

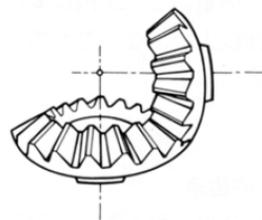


Figura 2.13 Engranes cónicos.

2.5.4 Engrane de piñón-cremallera.

La transmisión proporcionada por este tipo de engranes convierte el movimiento rotacional del piñón en un movimiento rectilíneo en la cremallera o de forma inversa, sin embargo el primero es el más usado. El piñón es una rueda con dientes rectos o helicoidales y la cremallera es una pieza recta dentada de la misma forma. Cuentan con una eficiencia del 98 – 99%.

Ventajas: Es el único engranaje capaz de convertir movimientos rotacionales en movimientos longitudinales

Desventajas: Aplicaciones limitadas.

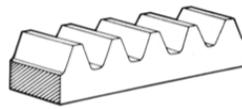


Figura 2.14 Engrane de cremallera.

2.5.5 Engranajes corona sin fin. (Worm gear pair)

Se compone de un engrane especial que se asemeja a un tornillo (piñón) y de una rueda con dentado recto o helicoidal (corona). Se usan en ejes perpendiculares y a diferencia de otros engranes, la transmisión de potencia ocurre cuando el filete (cuerda) del piñón se desliza a través de los dientes de la corona, pero no es posible mover el piñón al intentar rotar la corona. Por lo que lo hace irreversible, proporcionándole la ventaja de auto bloquearse.

El efecto de deslizamiento entre el piñón y los dientes de la corona provocan un efecto de calentamiento por fricción. La eficiencia de este tipo de engranes puede ir del 40% al 80% dependiendo de varios factores como la velocidad y la lubricación, sin embargo es considerado como un engranaje de baja eficiencia.

La lubricación ayuda a reducir los efectos de calentamiento por fricción y mejora la eficiencia. Una de las grandes ventajas de este tipo de engranaje es el reducido espacio que se requiere para transmitir potencia de un eje a otro con una alta relación de velocidad.

Limitaciones: Baja eficiencia. El piñón transmite la potencia a la corona por medio de movimientos de deslizamiento, por lo tanto hay pérdidas por fricción.

Ventajas: Tolera grandes cargas y puede reducir la velocidad considerablemente. El engranaje se auto bloquea.

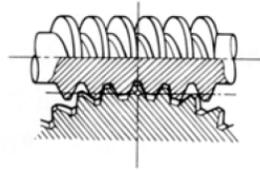


Figura 2.15 Engrane corona sin fin.

La disposición del reductor corona sin fin de los motores usados como actuadores en el brazo robótico del proyecto es similar al de la figura 2.16, con la excepción de que el conmutador en los motores que usé se encuentra en el otro extremo del eje.



Figura 2.16 Corte transversal de un motorreductor corona sin fin de corriente directa (ERMEC company)

2.6 TIPOS DE ROBOTS CON RESPECTO A SU VOLUMEN DE TRABAJO.

Existen robots de diferentes formas y tamaños. El tipo de sistema de coordenadas del manipulador también varía. El tipo de sistemas de coordenadas, el arreglo de las articulaciones, el largo de los segmentos del manipulador, etc. Todo lo anterior ayuda a determinar la forma del volumen de trabajo.

El volumen de trabajo varía de un fabricante a otro, dependiendo del diseño del manipulador. Combinando diferentes configuraciones en un solo robot puede resultar en otro tipo de volumen de trabajo. Antes de escoger la

configuración de un robot en particular, la aplicación debe ser estudiada cuidadosamente para determinar el volumen exacto de trabajo requerido.

Existen volúmenes de trabajo que tienen formas geométricas y otras que son irregulares. Un método para clasificar un robot es por la configuración de su volumen de trabajo.

Pueden existir robots equipados con más de una configuración, pero las principales configuraciones de robots son: brazo articulado, cartesiano, cilíndrico y esférico. Cada configuración está diseñada para diferentes aplicaciones.

2.6.1 Configuración de Brazo articulado.

Esta configuración es la más común. Estos tipos de robot generalmente son referidos como antropomórficos debido a que sus movimientos están basados en el brazo del cuerpo humano. Una serie de segmentos rígidos forman el brazo y el antebrazo. Varias articulaciones imitan el movimiento de la muñeca, codo y hombro.

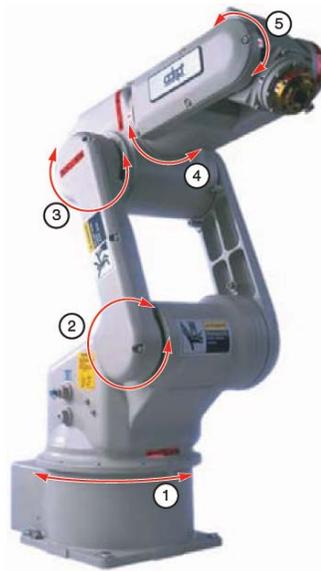


Figura 2.17 Brazo con cinco articulaciones rotatorias con la configuración de un brazo articulado verticalmente (Adept Technology, Inc.)

Un robot de brazo articulado se desenvuelve en un volumen de trabajo irregular. Existen aparte dos configuraciones de brazos articulados: verticalmente articulado y horizontalmente articulado.

El robot de configuración verticalmente articulado, mostrado en la figura 2.17, cuenta con cinco articulaciones rotatorias. La configuración vertical se describe en la figura 2.18 b).

El robot de brazo articulado verticalmente es práctico para tarea de pintado debido al largo alcance que su configuración permite.

La configuración horizontalmente articulada generalmente está formada por una articulación vertical (linear) y dos articulaciones rotatorias. También es conocido como configuración SCARA (selective compliance assembly robot arm), fue diseñado por el profesor Makino de la Universidad de Yamanashi Japón. El principal objetivo fue crear una configuración que contara con un buen rendimiento en movimientos horizontales y con un movimiento vertical rígido.

La configuración básica del SCARA es una adaptación de la configuración cilíndrica. El robot SCARA mostrado en la figura 2.19 está diseñado para aplicaciones en ambientes limpios, como el manejo de discos en la industria electrónica.

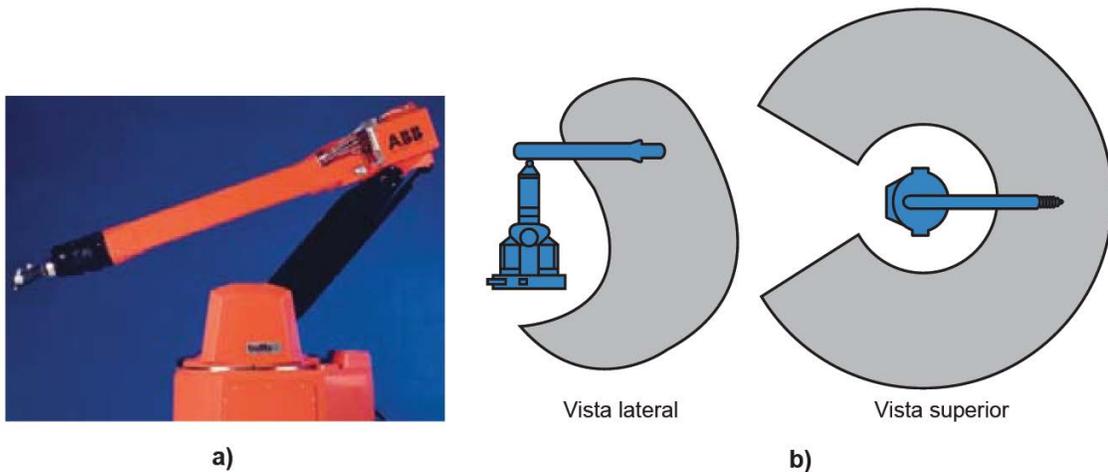


Figura 2.18 a) – Robot para tareas de pintado, verticalmente articulado. (ABB Graco Robotics, Inc.) b) – El área sombreada representa la vista del volumen de trabajo del robot.

Los robots SCARA son ideales para operaciones en donde el movimiento vertical requerido es menor comparado con los movimientos horizontales requeridos, como en aplicaciones de trabajo de ensamble donde se recogen piezas verticalmente y se sostiene mientras se transportan a un lugar cercano donde una unidad está siendo ensamblada.



Figura 2.19 Configuración de robot SCARA. Resaltando sus dos articulaciones rotatorias y una sola articulación vertical en esta configuración horizontal. (Adept Technology, Inc.)

La configuración del brazo articulado cuenta con varias ventajas. Es por mucho, la configuración más versátil y provee de un volumen de trabajo más grande que la de las configuraciones cartesiana, cilíndrica y esférica. Ofrece un alcance más flexible, por lo que es ideal para operaciones de soldado y pintura por spray.

Sin embargo también cuenta con desventajas. Requiere de un controlador muy sofisticado y la programación es más compleja que la de las otras tres configuraciones.

Esta configuración se vuelve menos estable a medida que el brazo se aproxima al límite de su alcance.

Las siguientes son algunas aplicaciones típicas de la configuración del brazo articulado:

- ▲ Ensamble automatizado en líneas de producción.
- ▲ Manejo y transporte de partes o materiales.
- ▲ Aplicación de soldadura
- ▲ Aplicación de pintura.
- ▲ Corte de material
- ▲ Manejo de materiales peligrosos
- ▲ Cirugías en seres humanos

2.6.2 Configuración de robot cartesiano.

El movimiento de un robot usando la configuración cartesiana, se puede describir como la intersección perpendicular de tres líneas rectas, referidas como los ejes X, Y y Z (figura 2.20). El movimiento puede empezar y detener simultáneamente. Esto permite que el robot se desplace directamente al lugar deseado, en lugar de seguir trayectorias paralelas en cada eje.

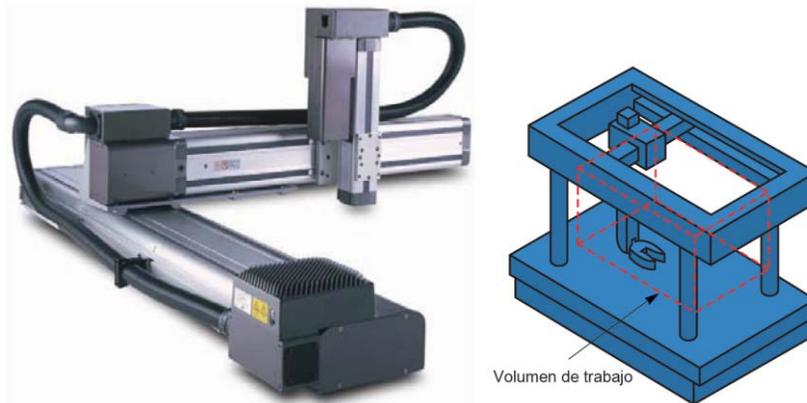


Figura 2.20 Robot de configuración cartesiana para trabajos de alta precisión.
(Adept Technology, Inc.)

Una ventaja de los robots con configuración cartesiana es que todos sus movimientos son lineales, por lo que controlarlos se vuelve más sencillo, figura 2.20. Tienen un alto grado de rigidez mecánica y precisión. Pueden llevar cargas pesadas y esta capacidad no varía en cualquier posición dentro del volumen de trabajo. Como desventaja, los robots cartesianos generalmente están limitados a un pequeño volumen rectangular de trabajo.

Aplicaciones típicas de los robots cartesianos:

- ▲ Transporte de materiales de un punto a otro
- ▲ Ensamblaje
- ▲ Cortadoras a base de chorro de agua
- ▲ Aplicación de soldadura

2.6.3 Configuración cilíndrica.

La configuración cilíndrica consiste en dos deslizamientos ortogonales a 90°, montados en un eje rotatorio, figura 2.21. El alcance final se realiza cuando el segmento final se mueve hacia adentro y hacia afuera. Para el

movimiento vertical, el segmento final (o brazo) se mueve de arriba-abajo en un poste estacionario, el movimiento a través de estos tres ejes traza un cilindro (volumen de trabajo).

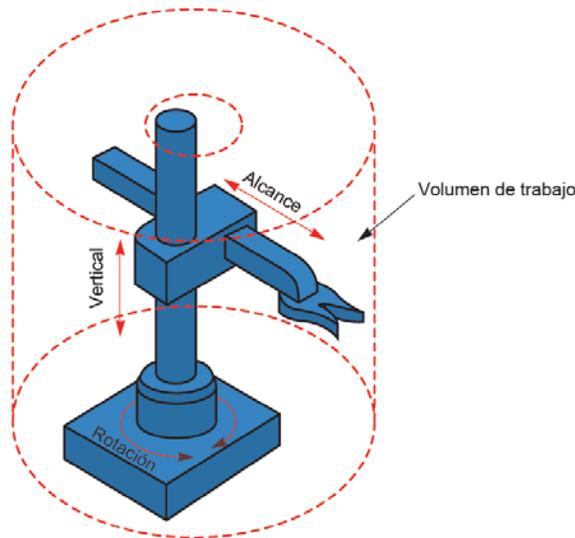


Figura 2.21 Volumen de trabajo de la configuración cilíndrica formada por el movimiento de sus tres ejes.

Generalmente una configuración cartesiana resulta tener un volumen de trabajo mayor que la configuración cilíndrica. Estos robots son ideales para operaciones de recoger y depositar objetos. Sin embargo también cuentan con algunas desventajas. Su capacidad de rigidez mecánica se ve reducida debido a que los robots con ejes rotatorios deben contrarrestar la inercia del objeto que están transportando mientras el robot rota sobre su eje. La configuración cilíndrica requiere de un sistema de control más complicado que la configuración cartesiana.

Aplicaciones típicas de la configuración cilíndrica:

- ▲ Manejo y transporte de partes o materiales.
- ▲ Máquina de carga y descarga.
- ▲ Moldeado por inyección.

2.6.4 Configuración esférica (polar)

La configuración esférica, algunas veces referida como la configuración polar, se asemeja al funcionamiento de un telescopio, un punto pivote le proporciona el movimiento vertical, el cambio telescópico de las lentes le da alcance. El movimiento rotatorio se encuentra en un eje perpendicular a la

base. La figura 2.22 ilustra el volumen de trabajo de la configuración típica de un robot de configuración esférica.

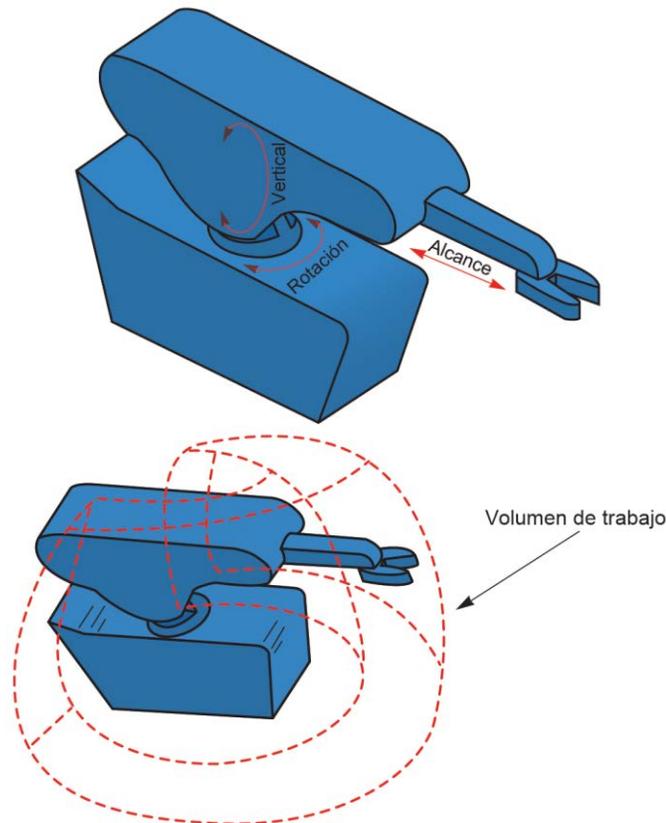


Figura 2.22 Configuración esférica formada por el movimiento vertical generado desde el punto pivote y por la rotación de la base. Volumen de trabajo del robot esférico.

Generalmente brinda un mayor volumen de trabajo que la configuración esférica. El diseño es sencillo y posee buenas capacidades en el transporte de carga. Esta configuración se ajusta en donde no hay espacio para varios movimientos verticales. Sus desventajas incluyen la reducción de rigidez mecánica, tiene limitaciones de movimientos verticales y tiene la necesidad de un sistema de control más complicado que las configuraciones cilíndrica y cartesiana. También con los problemas de la inercia y precisión.

Aplicaciones típicas de la configuración esférica:

- ▲ Manejo y transporte de partes o materiales.
- ▲ Aplicación de soldadura
- ▲ Manejo de materiales peligrosos
- ▲ Montaje por inserción

Capítulo 3.

PROYECTO Y CONSTRUCCIÓN DEL BRAZO ROBÓTICO.

3.1 SELECCIÓN DEL BRAZO ROBÓTICO.

Como mencioné anteriormente y debido a que la automatización ha sido un tema de gran interés para mí y en general un tema muy importante en la actualidad, me di a la tarea de buscar la forma de expresar mis ideas de forma práctica y que pudiera apreciarse fácilmente. Consideré que la mejor manera de demostrarlo era mediante un brazo robótico, debido a que es un icono de la automatización en diversas áreas pero principalmente en la industrial.

Para demostrar algunas de las capacidades de automatización que puede proporcionar el diseño del modulo de control y potencia descrito en el capítulo 1, diseñé y construí un brazo robótico a partir de motorreductores reutilizados, figura 3.1.

3.1.1 Características Básicas.

Para su construcción utilice los recursos que tuve a mi disposición, en el caso de los actuadores, los motorreductores de corriente directa que conseguí a muy bajo costo, fueron el elemento determinante para el diseño del brazo robótico. Por otra parte y como lo mencioné en el capítulo 1, la selección de los motorreductores también determinaron el diseño electrónico del modulo de control y potencia.

Básicamente está conformado por tres motorreductores corona sin fin de 12v corriente directa, el motor 1 de la base realiza los movimientos de izquierda – derecha de todo el brazo en un ángulo de 200° de libertad, el

motor 2 ubicado en la punta del primer segmento se encarga de realizar los movimientos de arriba – abajo con 160° de libertad y el motor 3 ubicado al principio del segundo segmento (muy cercano al motor 2) se encarga de proporcionar un giro a distancia al efector de izquierda – derecha con 360° de libertad. Por último cuenta con un solenoide de 12v que funge como efector del brazo robótico para sujetar y soltar determinados objetos.



Figura 3.1 Brazo robótico del proyecto, semejante a la configuración esférica o polar.

3.1.2 Construcción.

Inicialmente construí la base del motor conformada por el motor 1 con el eje del reductor posicionado hacia arriba (el cual proporciona el movimiento izquierda – derecha al brazo) y por un tripie conformado por tres secciones iguales de perfil de ángulo de acero de aletas iguales de $1'' \times 1/8''$ con una longitud de $7''$, los cuales están sujetos a la placa del motor por tornillos de $1/4''$.

El brazo está sujeto al eje de $3/16''$ de diámetro con cuerda del motorreductor 1 (como se observa en la figura 3.2), el brazo es un perfil ángulo de $1'' \times 1/8''$ con una altura de $10 \frac{1}{2}''$. En la punta, el brazo aloja dos motorreductores, el motor 2 para proporcionar al ante brazo movimientos de arriba – abajo y el motor 3 que se encarga de proporcionar movimiento rotatorio al efector final.



Figura 3.2 Unión de la base entre el motor 1, el tripie y el brazo.

El antebrazo es un perfil ángulo de acero de 1"x1/8" con una longitud de 9 1/2", en su punta se encuentra el solenoide de 12v que funge como efector final del brazo. Tomando en cuenta el solenoide, el ante brazo alcanza una longitud total de 17".

El giro del efector producido por el motor 3 es proporcionado por medio de un tornillo esparrago de 3/8" de diámetro ajustado al eje del motorreductor 3 por medio de un cople flexible ubicado en paralelo y a lo largo del segmento 2 (antebrazo) hasta el cuerpo del efector (solenoide).

Este giro representaba un problema en cuando a la alimentación de energía del propio efector (debido al giro del mismo) y para evitar el enrollamiento de cables por el giro del efector, el efector es energizado por medio de un conmutador con escobillas que se encuentra acoplado entre el tornillo esparrago y el propio efector, como se muestra en la figura 3.3.



Figura 3.3 Alimentación del solenoide por medio de escobillas de conmutación.

Este brazo robótico tienen una configuración y volumen de trabajo similar a la configuración esférica o polar, a excepción del segmento telescópico, pero como lo comenté anteriormente cumple con el fin de demostrar físicamente las capacidades de automatización del módulo de control y potencia.

El diseño original del brazo robótico determinaba su posición por medio de relaciones de tiempo provenientes del programa de la computadora a través del módulo de control y potencia, lo que provocaba la posición deseada al final del efector no fuera exacta. Por este motivo añadí un sistema de sensores a los motorreductores que enviaran señales de retroalimentación al programa informando el número de vueltas exactas que ha dado el motor y de esa manera determinar de una forma mucho más exacta la posición en la que se encuentra cada motor y por lo tanto también la del efector.

En la figura 3.4 se observa el conjunto completo del proyecto que consta del módulo de potencia con los microcontroladores Picaxe integrados; el brazo robótico, la botonera de 4x4 del Picaxe 1 que se encarga de enviar instrucciones de forma serial al módulo de potencia y por último la PC.

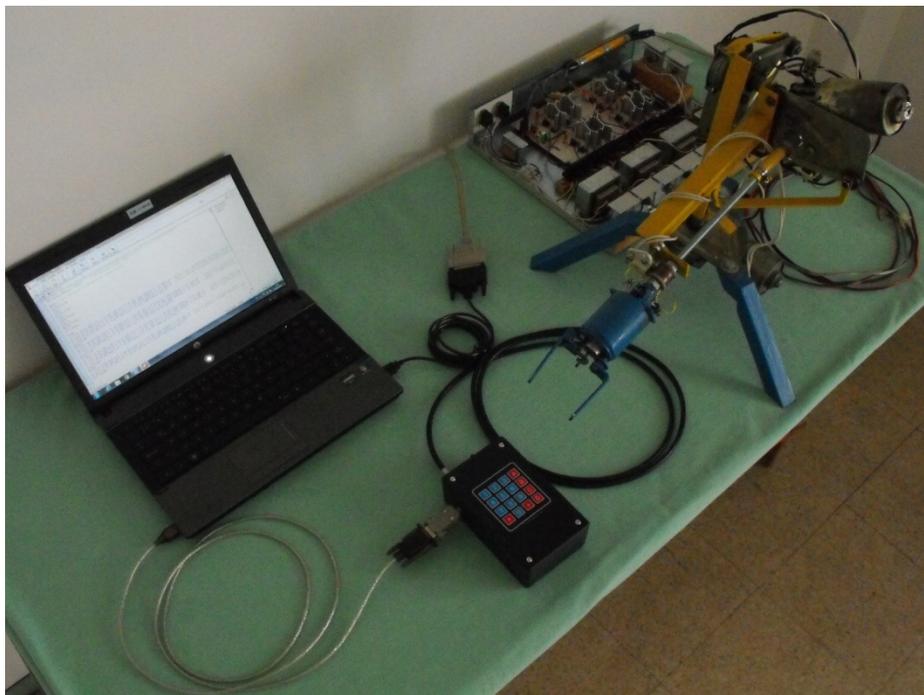


Figura 3.4 Conjunto completo del Proyecto.

Posteriormente en la figura 3.5 se muestra brevemente el funcionamiento por medio de un diagrama a bloques del conjunto del proyecto.

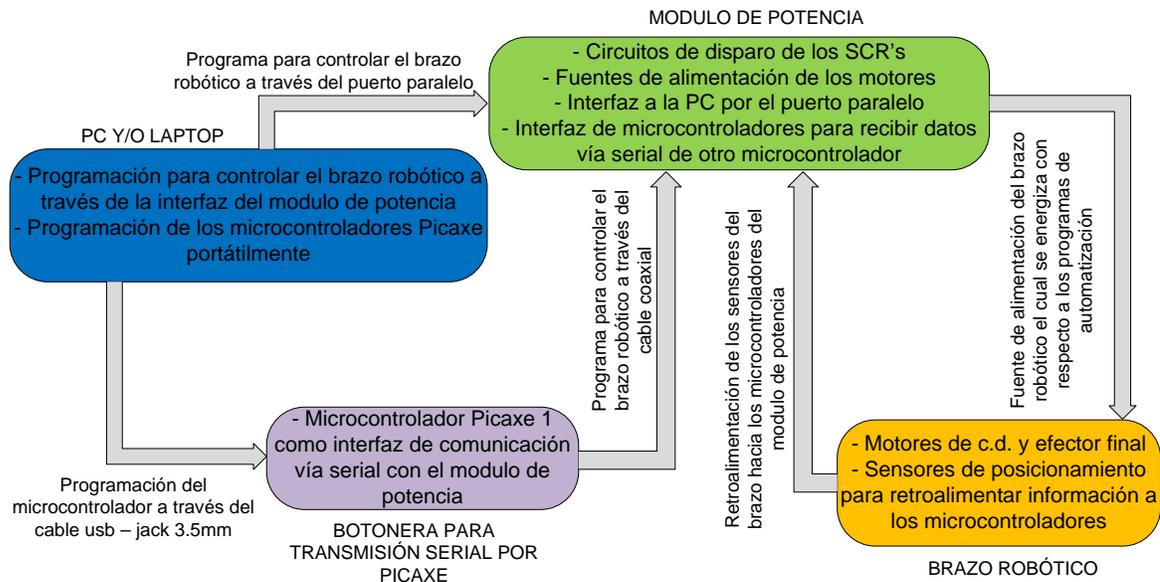


Figura 3.5 Diagrama a bloques del funcionamiento del conjunto del proyecto.

3.1.3 Sensores.

Un sensor es un convertidor que mide cantidades físicas y las convierte en señales que pueden ser interpretadas por otro instrumento, generalmente y como en este caso, un instrumento electrónico.

Para el correcto desempeño en el amplio rango de aplicaciones de los robots, se requieren diferentes tipos de sensores para realizar diferentes tipos de tareas sensitivas. Se han desarrollado diferentes tipos de dispositivos para cumplir varias necesidades sensitivas como lo son: orientación, desplazamiento, velocidad, aceleración, fuerza, etc.

Para ya no controlar los motores por medio de lapsos de tiempo, sino por número de vueltas en el eje del motor, coloqué un reed switch (switch de lengüeta, también conocido como relevador magnético) en la carcasa de cada motor.

Un reed switch es un pequeño dispositivo electromecánico (figura 3.6) que contiene dos lengüetas ferro-magnéticas herméticamente selladas en una capsula de vidrio rellena de un gas inerte (generalmente nitrógeno) o encapsuladas en vacío. Cuando se acerca a un campo magnético, las

lengüetas ferro-magnéticas se cierran, creando la función de switcheo. El reed switch es pequeño, rápido y seguro.



Figura 3.6 Reed switch o relevador magnético RDS-1.

Para activar los reed switches se requería de una fuente de campo magnético que con respecto al movimiento del eje del motor activara una vez por vuelta el reed switch. Debido a que este brazo robótico es un modo de ejemplificar el funcionamiento del modulo de control y potencia, modifiqué los motorreductores al extender el eje por fuera de la carcasa para que con una determinada disposición y con la colocación de un pequeño imán en el extremo de la extensión, activara el reed switch como se muestra en la figura 3.7.

Con la adición del reed switch es posible alcanzar un control más exacto de la posición de los motorreductores del brazo robótico.

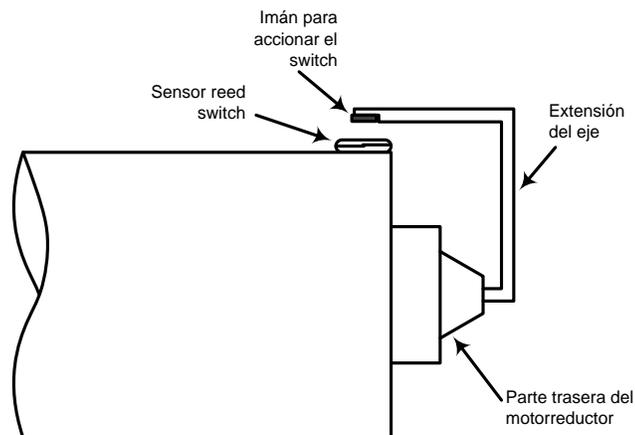


Figura 3.7 Disposición del sensor reed switch en la parte trasera del motorreductor.

Debido a que durante el control por computadora la conexión se hace por medio del puerto paralelo no existe una gran flexibilidad para conectar los sensores magnéticos, de igual forma por las propias limitaciones del programa de control. Sin embargo son capaces de operar por medio de los micro controladores picaxe integrados al modulo de control y potencia, en donde tengo más posibilidades de entrada y salidas debido a los propios microcontroladores.

3.1.4 Rediseño de la base del brazo robótico.

La construcción de este brazo robótico cumplió con la finalidad de mi propósito para demostrar el funcionamiento de este sistema de automatización, no obstante al finalizar y probar todo el conjunto, decidí mejorar el diseño de mi brazo robótico al construir otra base que brindara un mayor soporte para el mismo.

Decidí rediseñar la base del brazo debido a que en la construcción del primero monté la estructura en la flecha del reductor del primer motor que se encuentra en el tripie. Aunque ha resistido todas las pruebas a las que lo he sometido, la unión ideal sería un acoplamiento entre el motorreductor y el resto de la estructura.

Nuevamente me di a la tarea de buscar un acoplamiento comercial, ya que uno a la medida resultaría muy costoso. Pensé en que un sistema de engranaje sería muy buena opción, de esta forma y recorriendo la calle “artículo 123” del centro de la ciudad del D.F. encontré y compré un kit de reparación para la caja de transmisión de una lavadora marca Koblenz (figura 3.8), que consta de un engrane helicoidal, su piñón y el cople flexible que se acopla al eje motriz de la lavadora.

No me fue posible usar la caja de transmisión de la lavadora, que a pesar de contar con las distancias entre los centros del engrane y del piñón fijas, la caja está diseñada para dar un movimiento de retorno al eje motriz para la función del lavado. Lo que me orillo a construir una caja de engranaje que estaría unida a la base nueva del brazo.



Figura 3.8 Kit de reparación de engrane y piñón para lavadora Kovlenz.

El diámetro interior del engrane es de $\text{Ø } 1''$ y el diámetro del eje del piñón es de $\text{Ø } 1/2''$. Por lo que la idea principal es que el motor de la base se acople al eje del piñón y que el engrane lleve en su diámetro interior el eje que soporte el resto de la estructura del brazo. Como eje del engrane utilicé un tubo galvanizado de pared gruesa de $\text{Ø } 3/4''$ C-40 x 40cm. Para la elaboración de la caja de engranes usé dos placas de lámina negra de $4'' \times 8'' \times 1/8''$, que estarán separadas por cuatro secciones de tubo galvanizado de $\text{Ø } 3/8''$ C-40 colocados en las esquinas de las placas y atornilladas con 4 tornillos de $\text{Ø } 1/2'' \times 3/4''$, aumentando el diámetro interno de las secciones de tubo a $\text{Ø } 1/2''$ para que los tornillos también los atraviesen, pero debido a que las secciones de tubo son de pared gruesa mantendrán separadas las dos placas formando la caja de engranaje.

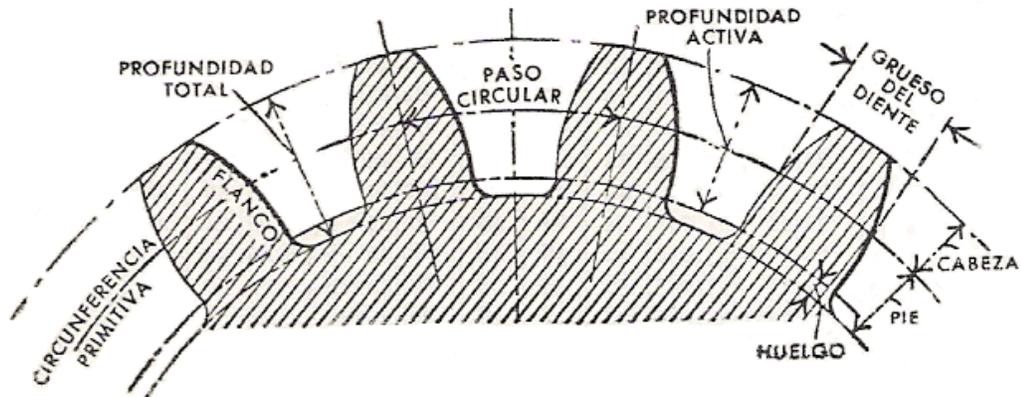
Sin embargo me encontré con el problema de encontrar la correcta localización del engrane y del piñón para que sus dientes hagan contacto en la zona correcta de sus flancos.

El Manual Universal de la Técnica Mecánica dispone una serie de reglas y formulas de donde se puede obtener la distancia entre los centros del engrane y el piñón.⁵ De acuerdo a dichas disposiciones obtenemos lo siguiente:

- ▲ Módulo: Razón del diámetro primitivo al número de dientes. En el sistema internacional, en milímetros.
- ▲ Diámetro Primitivo: Diámetro de la circunferencia primitiva. Se sobreentiende generalmente como el módulo por el número de dientes.

⁵ Erik Oberg y F.D. Jones, "Manual Universal de la Técnica Mecánica" (New York: The Industrial Press, 1973), pp. 766-777.

- ▲ Circunferencia Primitiva: Una circunferencia cuyo radio es igual a la distancia del eje del engranaje hasta el punto de tangencia.
- ▲ Punto de tangencia: El punto de tangencia de las circunferencias primitivas o el punto donde la línea de centros de engranajes apareados corta las circunferencias primitivas (línea central donde los dientes de ambos engranes hacen contacto).



3.9 Partes del diente de engranaje.⁶

Para obtener la distancia entre centros: Súmese los diámetros primitivos del piñón y del engranaje y divídase entre dos. Donde D_a y D_b son los diámetros primitivos.

$$C = \frac{D_a + D_b}{2} \quad \dots (1)$$

Para obtener el diámetro primitivo: Multiplique el número de dientes por el módulo métrico. Donde “N” es el número de dientes y “M” es el módulo.

$$D = N \times M \quad \dots (2)$$

De acuerdo a las reglas para el sistema de engranajes de modulo, el diámetro exterior se obtiene al añadir “dos” al número de dientes del engrane y la suma se multiplica por el módulo.

$$D_e = (N + 2) \times M \quad \dots (3)$$

⁶ Erik Oberg y F.D. Jones, “Manual Universal de la Técnica Mecánica” (New York: The Industrial Press, 1973), p. 706.

Usando el vernier se obtiene el Diámetro exterior del engrane y del piñón, que es igual a 96.4 mm y 22.1 mm respectivamente. Por lo tanto se puede sustituir y despejar en la ecuación (3) de la siguiente forma:

Para el Engrane

$N = 58$ dientes

$De = 96.4$ mm

$$M = \frac{De}{(N + 2)} = \frac{96.4 \text{ mm}}{60} = 1.6066$$

Para el Piñón

$N = 12$ dientes

$De = 22.1$ mm

$$M = \frac{De}{(N + 2)} = \frac{22.1 \text{ mm}}{14} = 1.5758$$

De esta forma se puede sustituir en la ecuación (2) para obtener los diámetros primitivos del engrane y del piñón.

Para el engrane:

$$D = N \times M = 58 \times 1.6066 = 93.1828 \text{ mm}$$

Para el Piñón:

$$D = N \times M = 12 \times 1.5785 = 18.9420 \text{ mm}$$

Finalmente con estos datos se puede obtener la distancia entre los centros del engrane y del piñón al sustituir en la ecuación (1).

$$C = \frac{Da + Db}{2} = \frac{93.1828 \text{ mm} + 18.9420 \text{ mm}}{2} = 56.0624 \text{ mm}$$

Contando con la distancia entre centros del engrane y el piñón, continúe por trazar sobre las placas las zonas cuadradas donde deben ir circunscritos los

diámetros de los barrenos para el eje del piñón, el eje del engrane y los cuatro tornillos que atraviesan las placas, así como las secciones de tubo. Una vez marcadas las placas, llevé los materiales a un taller mecánico para que perforaran con precisión los barrenos de las placas; para que ampliaran el diámetro interno de las cuatro secciones de tubo a $\varnothing 1/2"$; para recortar la flecha del piñón a la medida (barrenar la sección recortada para posteriormente yo machuelear con respecto a la cuerda de la punta de la flecha del motorreductor, ya que este punto es donde se utilizaría el cople flexible para transmitir movimiento al piñón) y por último que maquinaran una sección de 10 cm del tubo de $\varnothing 3/4"$ C-40 x 40cm (ya que su diámetro externo es de $\varnothing 1.050"$ y rebasa el diámetro interno del engrane por 0.050").

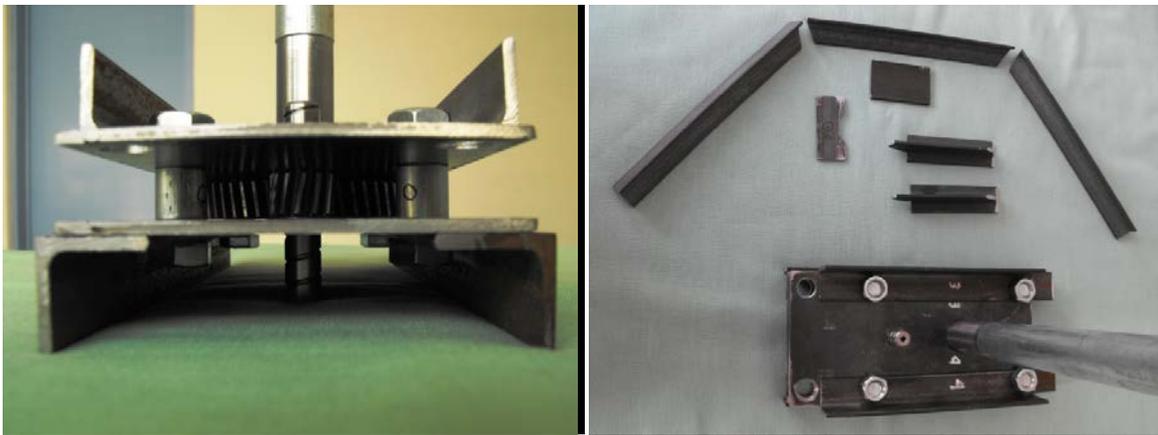


Figura 3.10 Vista frontal y superior de la caja de engranaje.

Una vez que las placas están perforadas y listas para armarse se le acoplaron cuatro perfiles ángulo de 1" a lo largo y por los extremos de las placas, dos en la parte inferior y dos en la parte superior, con el propósito de conformar una estructura para sustentar el tripie en el que la base se sustentara, el tripie está formado por tres secciones de perfil T de 1", 2 frontales de 20 cm y una trasera de 16 cm (como se puede observar en la figura 3.10).

La distancia entre los soportes frontales del tripie es de 45 cm, mientras que la distancia entre cualquiera de los soportes frontales al soporte trasero es de 42 cm, ofreciendo una altura a toda la estructura de 55 cm (de los soportes a la punta de la flecha de $\varnothing 3/4"$ C-40 x 40 cm).

Para sujetar el motor en la base y acoplarlo al piñón de el engranaje, uní una estructura formada de perfil T de 1" y solera de 1" por debajo de la base, sujetando y atornillando la carcasa del motor a la solera en sus tres puntos de apoyo y aislado dieléctricamente el motor del resto de la estructura por medio de cilindros de hule. La estructura que sostiene al motor se ajustó con respecto a la unión del de la flecha del motorreductor con la flecha del piñón por medio del cople flexible que era parte del kit de reparación de la lavadora.

Esta base evita que la flecha del motorreductor reciba el peso de la estructura, permitiendo que por otra parte sea el sistema de engranaje quien reciba el peso. La base con todas sus partes ensambladas se muestra en la figura 3.11

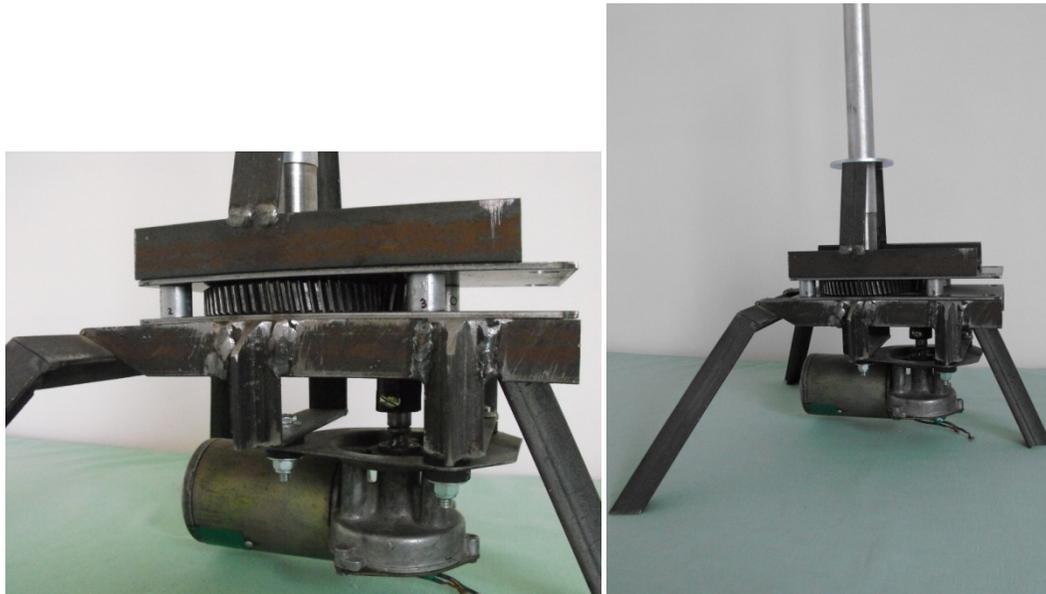


Figura 3.11 Acoplamiento del motorreductor con el piñón y vista del tripie armado.

Capítulo 4.

PROGRAMACIÓN DEL BRAZO ROBÓTICO

4.1 RESEÑA DE LA PROGRAMACIÓN DE UN ROBOT

El lenguaje de programación de un robot sirve como interface entre el usuario humano y el robot industrial. De donde emergen algunas preguntas: ¿Cómo puede el programador describir fácilmente los movimientos del robot dentro de su espacio de trabajo? ¿Cómo se programan múltiples manipuladores para que trabajen de forma paralela? ¿Cómo se describen en el programa los movimientos basados en la información obtenida por los sensores?

Los robots programables se diferencian de una automatización fija, por ser “flexibles” que significa que sean programables. No solo los movimientos del robot pueden ser programables, sino que por medio de sensores y por comunicación con otras áreas de procesos de automatización, los robots se puedan adaptar a variaciones de todo un sistema.

En los sistemas típicos de robots programables, hay una forma más sencilla y eficiente en que el usuario humano instruya al robot que camino debe seguir. Generalmente como punto de partida, el usuario especifica un punto especial del efector (una mano o una herramienta) como un punto operacional, algunas veces también llamado TCP (tool center point o punto central de la herramienta). Los movimientos del robot se describen por el usuario en términos del destino deseado por el punto operacional, el cual comúnmente se especifica por el usuario como un sistema de coordenadas. A menudo, la trayectoria se programa al especificar una secuencia o una vía de puntos. La vía de puntos se especifica en relación al sistema de coordenadas y denota locaciones a través de la trayectoria por la que el

punto operacional (efector) debe pasar y efectuar alguna operación. Paralelamente a la secuencia de vía de puntos, el usuario también podría indicar que el robot puede usar diferentes velocidades para transportar al efector por varias partes del trayecto. En algunas ocasiones y si la infraestructura del robot lo permite, se puede modificar la suavidad del movimiento del robot, como por ejemplo, cuando el efector se va a acercarse a una pieza delicada, puede llevar una curva de velocidad al aproximarse. De los puntos anteriores, el programa que genera la trayectoria debe planear los detalles de los movimientos: diferentes velocidades en los actuadores, tiempo de duración del movimiento, etc.

La sofisticación de la interface de programación del usuario se ha vuelto extremadamente importante a medida que los robots programables u otros dispositivos automatizables se instalan en aplicaciones industriales cada vez más demandantes.

Dentro de la programación de robots o dispositivos automatizables, es importante remarcar que típicamente son solo una parte mínima de un proceso automatizado. El término “célula de trabajo” es usado para describir un conjunto dispositivos locales, que puede incluir uno o más robots programables, más otro tipo de sistemas o dispositivos automatizados o no automatizados. En un siguiente nivel, las células de trabajo pueden estar interconectadas por redes dentro de la misma industria de manera que una computadora central pueda llevar el flujo de control de todas las células de trabajo.

Los robots industriales pueden programarse utilizando varias técnicas o métodos. Para sincronizar en secuencia los movimientos de los robots, ha sido aplicado el uso de controladores lógicos programables (PLC's) y el uso de computadores que controlan a su vez grupos de células de trabajo automatizadas.

Se tratan todos ellos de sistemas de control, que utilizan retroalimentación proveniente de interruptores limitadores, detectores de proximidad, etc. Esto permite el empleo de técnicas complejas en y fuera de línea de programación.

Programación en línea. Aquí se utiliza el brazo del robot mismo durante la operación directa de programación. Cuando la tarea que se está

programando es compleja el tiempo de programación puede resultar exageradamente largo. Básicamente existen dos métodos de programación en línea: aprendizaje mediante conducción guiada a través de la tarea y aprendizaje mediante paleta de programación. También es posible programar directamente desde una terminal de computadora conectada al controlador, lo que en principio es similar al método de aprendizaje por paleta de programación.

Programación fuera de línea. Programar fuera de línea significa la creación de un programa para una tarea robótica. Sin la necesidad de físicamente tener una conexión con el robot, ni cerca de él. De hecho, cuando se combina con técnicas de simulación, la programación fuera de línea puede llevarse a cabo antes de decidir qué robot adquirir para una aplicación específica. La simulación gráfica del robot y de su entorno ofrece muchos beneficios al programador. Los software de simulación tienen la capacidad de modelar el robot de manera cinemática y a veces de manera dinámica, para dar una representación animada, visual y en tiempo real de cómo funcionará el robot bajo condiciones programadas en su entorno de trabajo. La simulación se está convirtiendo en una herramienta de programación de robots de popularidad creciente en la industria, en la educación y en la investigación.

4.1.1 Control por computadora.

Actualmente muchas operaciones se automatizan por medio de la computadora. El sistema donde una sola computadora controla todas las operaciones en una industria es conocido como "sistema de manufactura flexible" (flexible manufacturing system FMS). Solo una computadora es necesaria para controlar diferentes procesos industriales y es capaz de programar o modificar cada aspecto de los procesos automatizados que tiene bajo su control.

Existe una amplia variedad de lenguajes de programación y métodos de programación disponibles como se mencionó anteriormente. El lenguaje de programación que usé para controlar los movimientos del brazo robótico fue el Q-basic.

El lenguaje de programación QuickBasic (Q-basic) fue desarrollado por Microsoft para emplearse en el sistema operativo MS-DOS. Siendo su predecesor el BASIC (Beginners All-Purpose Symbolic Instruction Code o Código simbólico de instrucciones todo-propósito para principiantes, en español). Q-Basic es un lenguaje diseñado para programadores principiantes ya que cuenta con comandos intuitivos, una estructura simple y por su versatilidad. Existe mucha información sobre él y pueden encontrarse gran variedad de tutoriales y ejemplos de programación en libros o en el internet.

El Q-basic es un lenguaje que interpreta línea por línea (de la misma forma en cómo los microcontroladores PICAXE ejecutan su programación).

El tipo de lenguaje interpretador de línea por línea es un lenguaje que ejecuta una línea de programación a la vez, de la misma forma en cómo aparece en el editor del programa. Casi todos los lenguajes de computadora modernos compilan el código del editor del programa. Esto significa que el programa desarrollado en el editor se traduce (compila) en un lenguaje de máquina. Este código compilado es el que ejecuta finalmente la computadora.

El programa que proporciona las secuencias de operación del brazo robótico está direccionado por el puerto paralelo de la computadora DB-25. A continuación se muestra el sencillo programa que controla el brazo robótico por medio de la computadora.

```
160 CLEAR 500: DEF SEG = C007: KEY OFF: CLS : OUT &H378, 0
180 POKE &HC008, 0: POKE &HC009, 0
190 G% = 0: g1% = 0: g2% = 0
200 REM IDENTIFICACION DE LOS MOVIMIENTOS
205 CLS
210 ID$(1) = "MOVER A LA DERECHA"
220 ID$(2) = "MOVER A LA IZQUIERDA"
230 ID$(3) = "MOVER HACIA ABAJO"
240 ID$(4) = "MOVER HACIA ARRIBA"
250 ID$(5) = "METER EL BRAZO"
260 ID$(6) = "SACAR EL BRAZO"
270 ID$(7) = "CERRAR EL EFECTOR"
```

```

280 ID$(8) = "ABRIR EL EFECTOR"
290 CLS
294 COLOR 1
295 PRINT SPC(15);
*****
300 PRINT SPC(15); "* UNIVERSIDAD NACIONAL AUTONOMA DE
MÉXICO      *"
310 PRINT SPC(15); "* FACULTAD DE ESTUDIOS SUPERIORES
CUAUTITLÁN  *"
315 PRINT SPC(15); "**                BRAZO ROBÓTICO
*"
320 PRINT SPC(15);
*****
330 PRINT : PRINT
340 COLOR 7
350 PRINT SPC(20); "I = INICIAR LA MEMORIZACION DE
MOVIMIENTOS": PRINT
360 PRINT SPC(20); "F = FINALIZAR LA MEMORIZACION": PRINT
370 PRINT SPC(20); "R = REPETIR LA MEMORIZACION": PRINT
380 PRINT SPC(20); "L = LIMPIAR LA MEMORIA": PRINT
395 PRINT SPC(20); "O = REGRESAR AL MENU PRINCIPAL": PRINT
400 PRINT SPC(20); "E = CIERRA EL PROGRAMA": PRINT
410 LOCATE 3, 20
415 PRINT : PRINT :
PRINT
416 PRINT : PRINT :
PRINT
420 PRINT SPC(25); "TECLADO NUMERICO MOVIMIENTOS"
430 PRINT SPC(25); "ESC CANCELA LOS MOVIMIENTOS"
436 COLOR 7
440 a$ = INKEY$: IF a$ = "" THEN 440
460 IF a$ = "6" THEN m% = 0: GOSUB 590: GOTO 440
470 IF a$ = "4" THEN m% = 1: GOSUB 590: GOTO 440
480 IF a$ = "2" THEN m% = 2: GOSUB 590: GOTO 440
490 IF a$ = "8" THEN m% = 3: GOSUB 590: GOTO 440
491 IF a$ = "7" THEN m% = 4: GOSUB 590: GOTO 440
492 IF a$ = "9" THEN m% = 5: GOSUB 590: GOTO 440

```

```

493 IF a$ = "1" THEN m% = 6: GOSUB 590: GOTO 440
494 IF a$ = "3" THEN m% = 7: GOSUB 590: GOTO 440
520 IF a$ = "l" OR a$ = "i" THEN GOSUB 1120: GOTO 440
530 IF a$ = "F" OR a$ = "f" THEN GOSUB 1190: GOTO 440
540 IF a$ = "R" OR a$ = "r" THEN GOSUB 880: GOTO 440
550 IF a$ = "L" OR a$ = "l" THEN GOSUB 1260: GOTO 440
565 IF a$ = "O" OR a$ = "o" THEN 290
570 IF a$ = "E" OR a$ = "e" THEN CLS ol: STOP
580 GOTO 440
590 REM ACCIONA MOTOR
620 IF G% = 0 THEN 680
630 POKE (&HC010 + g2%), m%
640 g2% = g2% + 1
650 IF g2% = 3000 THEN G% = 0: BEEP: BEEP
660 POKE &HC008, INT(g2% / 256)
670 POKE &HC009, g2% - (INT(g2% / 256) * 256)
680 N% = 2 ^ m%
690 GOSUB 1330
700 PRINT SPC(2); "NRO="; ID$(m% + 1); g2%; "M"; m%
720 OUT &H378, N%
730 FOR x = 0 TO 75: NEXT x: OUT &H378, 0
740 RETURN

880 REM REPITE MOVIMIENTOS
910 GOSUB 1330
920 PRINT "PAUSA ENTRE PASOS <"; w%; ">";
930 INPUT w%
940 g1% = PEEK(&HC009) + (PEEK(&HC008) * 256)
950 GOSUB 1330
960 PRINT "REPETICION DE MOVIMIENTOS INICIADA"
970 FOR g2% = 0 TO (g1% - 1)
980 m% = PEEK(&HC010 + g2%)
990 GOSUB 590: FOR x = 0 TO w%: NEXT x
1080 NEXT g2%: g2% = 0
1090 GOSUB 1330
1100 PRINT "REPETICION DE MOVIMIENTOS CERRADA": PRINT
1105 PRINT SPC(20); "(O) REGRESA AL MENU PRINCIPAL"

```

```

1110 RETURN
1120 REM MEMORIZACION
1130 REM * MEMORIZACION INICIADA *
1150 GOSUB 1330
1160 PRINT "INTRODUCIR MOVIMIENTOS A MEMORIZAR"
1170 G% = 1: g2% = 0
1180 RETURN
1190 REM FIN DE MEMORIZACION
1220 GOSUB 1330
1230 PRINT "MEMORIZACION COMPLETA": PRINT
1232 PRINT SPC(20); "(R) REPETIR LA MEMORIZACION": PRINT
1235 PRINT SPC(20); "(O) REGRESAR AL MENU PRINCIPAL"
1240 G% = 0
1250 RETURN
1260 REM LIMPIA MEMORIZACION
1290 GOSUB 1330
1300 PRINT "MEMORIZACION REINICIADA": PRINT
1305 PRINT SPC(20); "(O) REGRESA AL MENU PRINCIPAL"
1317 POKE &HC008, 0: POKE &HC009, 0
1320 RETURN
1330 REM LIMPIA AREA MENSAJES
1335 CLS
1360 LOCATE 10, 20: PRINT SPC(40);
1370 LOCATE 10, 20: RETURN

```

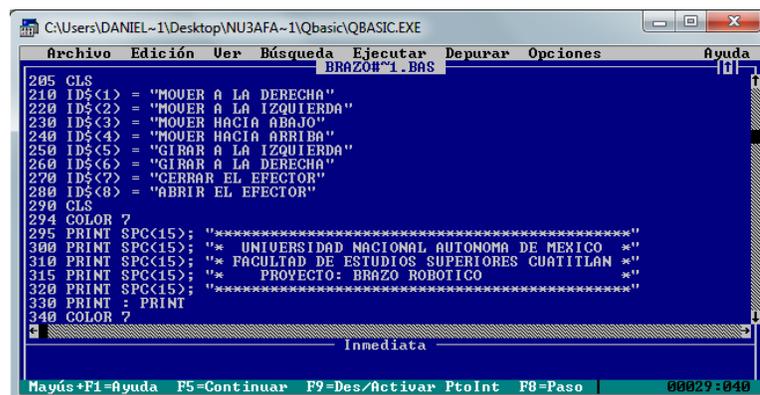
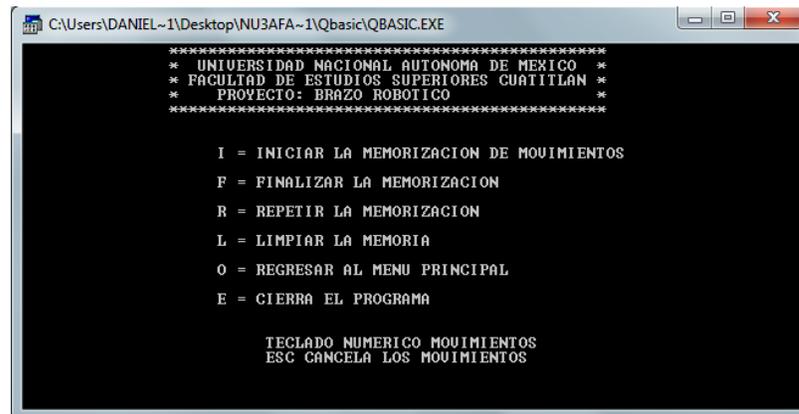


Figura 4.1 Ambiente del editor de Q-basic.

Básicamente el programa cuenta con dos tipos de operaciones, la primera permite almacenar en memoria una secuencia de movimientos ingresada

por el usuario, que involucra a los tres motorreductores y al efector. Para posteriormente terminada la grabación de movimientos, se reproduzca la secuencia de movimientos almacenada en memoria. La segunda operación con que cuenta el programa, es el accionamiento libre de los movimientos del robot por medio del teclado.



```
C:\Users\DANIEL~1\Desktop\NU3AFA~1\Qbasic\QBASIC.EXE
*****
* UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO *
* FACULTAD DE ESTUDIOS SUPERIORES CUATITLAN *
* PROYECTO: BRAZO ROBOTICO *
*****
I = INICIAR LA MEMORIZACION DE MOVIMIENTOS
F = FINALIZAR LA MEMORIZACION
R = REPETIR LA MEMORIZACION
L = LIMPIAR LA MEMORIA
O = REGESAR AL MENU PRINCIPAL
E = CIERRA EL PROGRAMA

TECLADO NUMERICO MOVIMIENTOS
ESC CANCELA LOS MOVIMIENTOS
```

Figura 4.2 Programa del brazo robótico en funcionamiento.

Adicionalmente cuenta con las funciones básicas de un menú de opciones, como regresar al menú principal, finalizar la memorización de movimientos, cerrar el programa, etc.

Capítulo 5.

CONTROL DEL MODULO DE POTENCIA POR MEDIO DE MICROCONTROLADORES PICAXE

5.1 EL MICRO CONTROLADOR.

Un microcontrolador es precisamente lo que sugiere su nombre, un “micro” – “controlador”, es una pequeña computadora dentro de un circuito integrado, el cual contiene un núcleo de procesamiento, memoria y periféricos programables de entrada/salida.

Hoy en día se pueden encontrar en diferentes dispositivos, como pequeños reproductores de música, maquinas de lavado e inclusive en los autos. Son programables, requieren poca energía, son baratos y existe una gran variedad de ellos para cada necesidad.

Como se mencionó, un microcontrolador es una computadora en un solo chip. Todas las computadoras tienen varias cosas en común, principalmente:

- ▲ Una unidad de procesamiento central (CPU) que ejecuta programas.
- ▲ Memoria de acceso aleatorio (RAM) donde puede almacenar información temporalmente..
- ▲ Memoria de solo lectura (ROM) donde se almacenan los programas que serán ejecutados.
- ▲ Dispositivos de entrada y salida (I/O) que permiten se establezca comunicación con el entorno, por ejemplo, dispositivos como el teclado, mouse, monitores y otros periféricos.

Por otra parte, hay características que pueden definir a los microcontroladores:

- ▲ “Embedded System” (Sistema Embebido o Integrado), es un sistema computarizado diseñado para controlar funciones específicas. Su principal característica es estar enfocado al control de una tarea en particular (a diferencia de una PC), con funciones limitadas de cómputo, confiable y a un bajo costo. Estos sistemas los podemos encontrar a nuestro alrededor como en reproductores DVD, algunos controles remotos, semáforos, cajas registradoras, juguetes, equipo médico, etc.
- ▲ Trabajan sobre un programa específico ya que están enfocados en una tarea en particular. Generalmente el programa es almacenado en una memoria ROM.
- ▲ Son dispositivos que consumen baja potencia.

Un microcontrolador puede tomar una señal de entrada del dispositivo que este controlando y responder mandando señales a diferentes componentes en el dispositivo. Usualmente los microcontroladores son pequeños y de bajo costo.

5.2 EL SISTEMA PICAXE

PICAXE explota las características únicas de la nueva generación de microcontroladores basados en memoria “FLASH”, a un bajo costo. Por lo que estos microcontroladores pueden ser programados una y otra vez hasta 100,000 veces, sin la necesidad de un costoso programador de PIC's.

Para su programación PICAXE puede utilizar lenguaje Basic, Diagramas de Flujo ó programación por escalera (los dos primeros son software gratuitos descargables en la página electrónica de PICAXE), programas relativamente más sencillos que los programas convencionalmente utilizados para los microcontroladores, como lo son el lenguaje C ó el ensamblador.

El poder del sistema PICAXE radica en su simplicidad, ya que no se requiere de algún sistema programador o borrador independientes a la tablilla de operación, esto debido a que los chips PICAXE son programados por medio de tres conexiones hacia el puerto serial de la computadora o por medio de un cable usb. Esto representa una gran ventaja, ya que cada circuito cuenta con una conexión integrada para la programación del chip PICAXE, lo cual se traduce en reducir considerablemente el número de veces que se requiere retirar el chip del circuito para programarlo y evitar de esta forma dañar algún pin del chip.

El software editor de programas PICAXE es gratuito y descargable desde su página electrónica (www.picaxe.co.uk), en esta página se encuentran diversos tipos de software gratuitos y de paga, que maximizan las funciones y facilidad para programar un chip PICAXE. También existe una gran variedad de circuitos prototipo pre-ensamblados (listos para usar), sensores, actuadores, cables conectores y la gama completa de chips PICAXE.

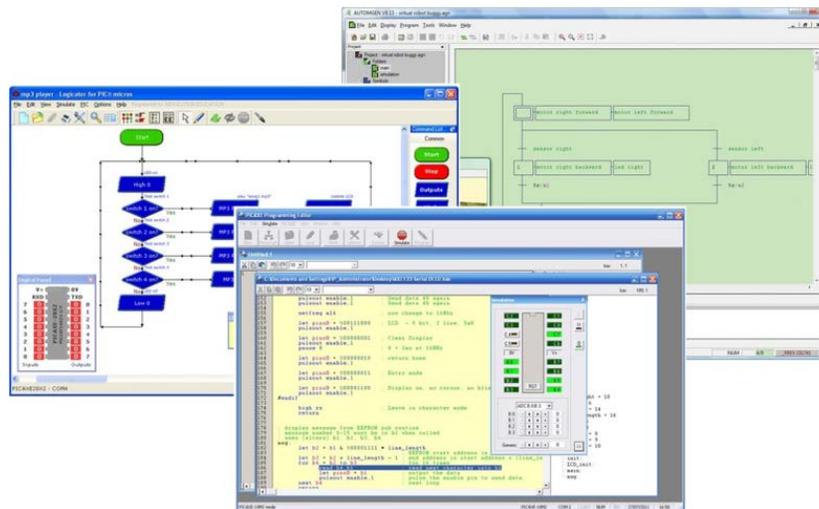


Figura 5.1 Diferentes tipos de software para programar los microcontroladores PICAXE.

Un microcontrolador PICAXE es un microcontrolador Microchip PICmicro™ estándar, el cual ha sido pre-programado con el código bootstrap de PICAXE (“bootstrap” es un pequeño programa que se activa inmediatamente después de que el microcontrolador ha sido energizado, con el propósito de cargar y ejecutar otro programa).

El código bootstrap de PICAXE permite realizar diferentes tipos de tareas como, reprogramar el chip directamente a través de la conexión con la computadora (eliminando la necesidad de un programador convencional), contiene diferentes rutinas sencillas (como pausas, retardos o sonidos) para que al momento de descargar un programa no se pierda tiempo descargando rutinas básicas.



Figura 5.2 Microcontrolador PICAXE 18M2+

Como los microcontroladores que se usan para los sistemas PICAXE, se adquieren en grandes cantidades, es posible programarlos con el código bootstrap de PICAXE y el fabricante los venda muy cercano al de un microcontrolador sin programar, por lo que el precio al consumidor es muy económico.

Últimamente la tecnología PICAXE (de la compañía "Microchip") ha ganado gran fama por llevar al siguiente paso a los micro controladores. Existe una gran variedad de dimensiones, desde 8 hasta 40 pines y están disponibles tanto en montura superficial, como en montura de socket. Su lenguaje de programación es sencillo, sin embargo es muy completo y suficiente para la mayoría de las necesidades en automatización, debido a que es compatible con una amplia gama de sensores y actuadores.

La conectividad de los chips PICAXE a la PC es extremadamente sencilla y fácil de usar, es difícil pensar que con las características antes mencionadas y muchas más por mencionar el costo de los chips PICAXE sea considerablemente más bajo que otras marcas en el área de micro controladores.

5.3 SISTEMA PICAXE IMPLEMENTADO EN ESTE PROYECTO.

El sistema automatizado por microcontroladores PICAXE funciona a base de 3 chips 18M2+ conectados entre sí (Picaxe1, Picaxe2 y Picaxe3), se implementaron 3 chips debido a las diferentes capacidades y funciones que se pueden obtener de la combinación de los mismos.

La primera característica del sistema es programar una serie de movimientos predeterminados por el usuario (programa de automatización), los cuales se podrán cambiar en cualquier momento de acuerdo a las necesidades, requiriendo para ello únicamente una PC portátil y el cable de conexión al modulo (usb-jack3.5mm).



Figura 5.3 Cables-conexión con la PC.

La segunda característica implementada en el sistema es la transmisión de datos seriales (por un solo hilo conductor), esta característica expande las capacidades en automatización implementadas en este modulo, debido a que abarca el control a distancia del equipo, este tipo de transmisión permitiría el control de diferentes módulos por un solo hilo conductor, ya que cada instrucción lleva una llave la cual especifica a que modulo hace referencia dicha instrucción.

La tercera característica radica en la simulación de tareas en paralelo, con esta gran ventaja es posible controlar cada motor independientemente de los demás, así como los sensores magnéticos. Esto es posible gracias a la última generación de chips PICAXE M2.

5.3.1 Picaxe1

Este chip tiene la función de transmitir los mandos al modulo de potencia, el cual cuenta con los chips Picaxe2 y Picaxe3. Sin embargo cuenta con la particularidad de enviar datos de forma serial gracias a las funciones propias de los chips Picaxe M2.

La tecnología de transmisión serial es usada cada vez más para la transmisión de información digital. Hoy en día, un gran número de redes de comunicación utilizan la transmisión serial. Entre las numerosas aplicaciones se incluyen redes computacionales para oficinas, internet, redes digitales de servicios integrados (RDSI ó ISDN en inglés, que proporcionan servicios de voz y datos, como lo son los servicios de telefonía, cable e internet), finalmente, automatización.

La transmisión de datos vía serial implica que un bit es enviado tras otro en una sola línea de transmisión. Generalmente los procesadores de los dispositivos, procesan datos de forma paralela, el transmisor realiza una conversión paralela a serial, mientras que el receptor realiza una conversión serial a paralela. Esto es llevado a cabo por módulos transmisores y receptores especiales, los cuales son comercialmente disponibles para diferentes tipos de redes.

Hoy en día existe una alta velocidad de transferencia de datos, por lo que el tiempo de transferencia de datos para este tipo de tecnología es muy aceptable en muchos casos. También hay una reducción de costos y esfuerzo de instalación, así como un sistema más sencillo para el usuario final.

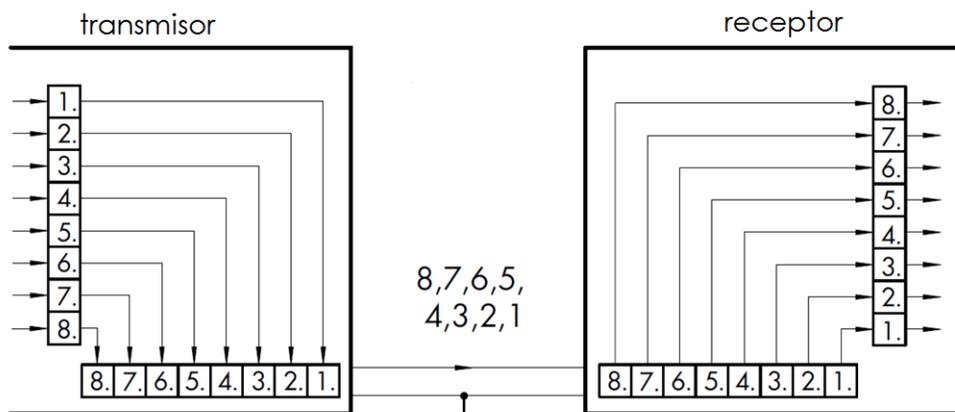


Figura 5.4 Ejemplo de transmisión de datos vía serial.

5.3.2 Características del sistema de transmisión serial.

La transmisión de datos vía serial se ajusta a la comunicación entre dos participantes, así como la comunicación entre varios participantes. Las principales características del sistema de transmisión serial son la dirección del flujo de datos y la cadencia o tasa máxima de datos.

- Dirección del flujo de datos.

Básicamente existen tres tipos de comunicación:

- ▲ Simplex: El envío de datos es en una sola dirección,
- ▲ Half-duplex: Las estaciones toman turnos para transmitir datos y una transmisión a la vez
- ▲ Full-duplex: Los datos se pueden intercambiar en ambas direcciones simultáneamente
- ▲

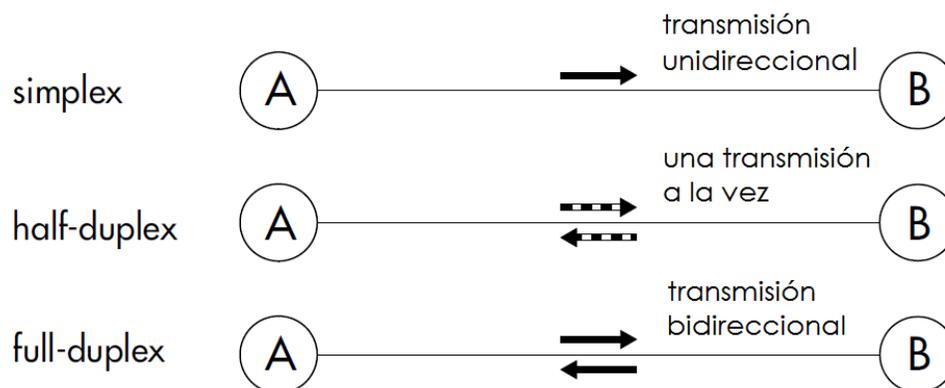


Figura 5.5 Técnicas de transmisión.

- Conexiones punto a punto.

En las conexiones punto a punto, el transmisor y receptor pueden conectarse a través de dos líneas separadas (la suma de dos canales simplex), la línea de transmisión de un participante es la línea de recepción del otro.

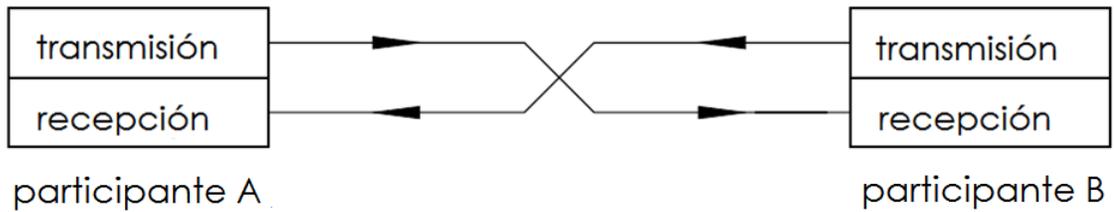


Figura 5.6 Conexión punto a punto

La serie de chips Picaxe M2 manejan conexión similar a la de punto a punto para la transmisión de datos seriales, pero con la diferencia de que en cada chip solo se puede recibir o transmitir un dato a la vez, debido a que la función de simulación de tareas en paralelo no opera con funciones especiales, como lo es en este caso la transmisión de datos vía serial, como se comentara más adelante.

- Redes de comunicación.

En redes de comunicación con varios participantes, el medio de transmisión usualmente es una sola línea para transmitir y recibir datos al mismo tiempo (full duplex). La secuencia de comunicación es coordinada por transmisión de información de control adicional, el cual se define como el comúnmente llamado protocolo de transmisión. Esta información de control ayuda a identificar la información, así como la fuente y la dirección de destino tras cada mensaje transmitido.

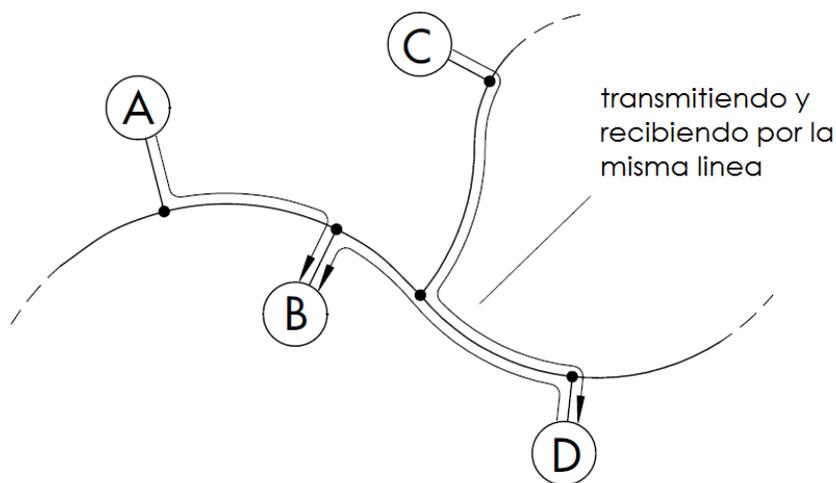


Figura 5.7 Red de comunicación con varios participantes.

- Velocidad de transmisión de datos

Un criterio esencial para determinar la capacidad en líneas de comunicación es la tasa de datos, es decir, la velocidad a la cual la información será transmitida. La tasa de datos se caracteriza por el número de bits transmitidos cada segundo, medido en bps (bits por segundo). Debido hoy en día la tasa de datos es extremadamente alta, unidades como, “kilobit por segundo (kbit/s)” y “megabit por segundo (Mbit/s)”, no son inusuales. Cuando cada bit es codificado y transmitido individualmente, la línea de transmisión debe ser capaz de transmitir frecuencias que correspondan a la mitad de la tasa de transmisión de bits. Ejemplo:

Tasa de transmisión de bits:	100 kbit/s
Frecuencia de transmisión:	50 kHz

Cuando es necesario alcanzar una alta tasa de datos, aún cuando el ancho de banda es limitado, varios bits pueden ser agrupados y codificados juntos. La figura 5.8 muestra como cuatro niveles diferentes de voltaje pueden ser usados para transmitir dos bits al mismo tiempo. Este método recorta los cambios de estado en la señal de la línea a la mitad y reduce la frecuencia de transmisión.

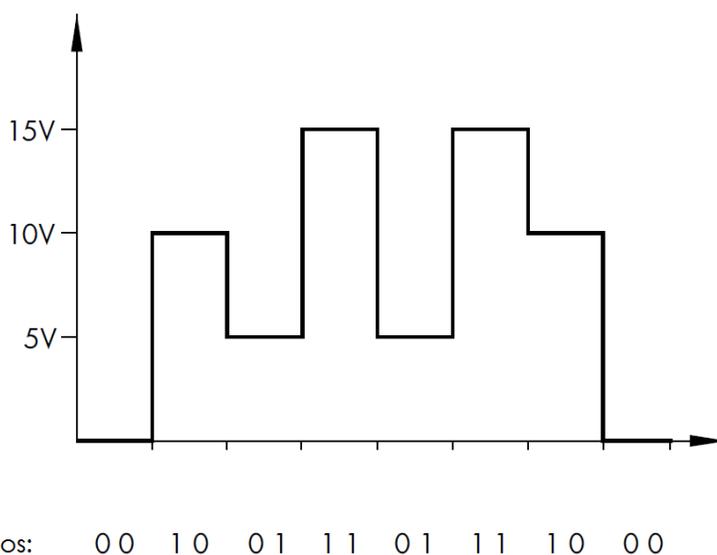


Figura 5.8 Una codificación más compleja reduce la frecuencia de transmisión

Para medir la velocidad de switcheo, es decir, “el número de cambios de voltaje o frecuencia por unidad de tiempo”, se utiliza el llamado “tasa de baudios”. Cuando solo un bit es transmitido por unidad de transmisión, la tasa de baudios es idéntica a la tasa de datos “bit por segundo” (bps).

La capacidad de comunicación en una línea no puede ser suficientemente definida solo por la tasa de baudios. Los siguientes parámetros, especialmente para redes con varios participantes, son importantes:

- ▲ Periodo de tiempo hasta que la línea está lista para transmitir y
- ▲ El numero de datos a ser transmitidos en adición al mensaje apropiado, tal como la dirección del dispositivo, información de control y así sucesivamente (entre más compleja sea la red se requieren de más parámetros).

5.3.3 Programación del Chip 1

Los comandos que requiere el chip 18M2 para transmitir datos vía serial son tres básicamente, pero en el caso del chip Picaxe1 de este proyecto requiere de dos de los comandos básicos: hsersetup y hserout.

El comando “hsersetup” se encarga de configurar dos pines del chip, los cuales se encargaran de la transmisión y recepción de datos. Y consta de la siguiente estructura:

HSERSETUP baud_setup, mode

- ▲ baud_setup es una variable/constante que especifica la tasa de baudios

B300_X	donde X =	B19200_X	40 for 40 MHz
B600_X	4 for 4MHz	B31250_X	64 for 64MHz
B1200_X	8 for 8 MHZ	B38400_X	
B2400_X	16 for 16MHz	B57600_X	
B4800_X	20 for 20MHZ	B115200_X	
B9600_X	32 for 32MHx		

▲ mode es una variable/constante en el que cada uno de sus bits cuenta con una función especial (no todos los chips cuentan con estas funciones)

Bit 0 – Recibe datos vía serial en segundo plano (background) a la memoria scratchpad (memoria de acceso rápido denominada bloc de notas). (No aplica para los chips M2)

Bit 1 – Invierte los datos de salida (0 = “T”, 1=“N”). (de los términos: “T”rue non-inverted y “N”egative inverted)

Bit 2 – Invierte los datos de entrada (0 = “T”, 1=“N”). (Solo aplica para chips X2)

Bit 3 – Deshabilita el pin hserout (datos de salida). (1 = hserout pin normal entrada/salida)

Bit 4 – Deshabilita el pin hserin (datos de entrada). (1 = hserin pin normal entrada/salida)

Detalles del comando:

La tasa de baudios se determina por el valor de baud_setup. Por conveniencia hay un número de valores predeterminados para este comando, por ejemplo, “B9600_4” para una tasa de baudios de 9600,n,8,1 a operando a 4 Mhz (refiriéndose “9600” a la tasa de baudios, “n” a que no hay bit de paridad, “8” a los bits de datos y “1” a un bit de paro), esta es la configuración convencional para la comunicación serial de la PC.

Cuando el bit 1 es 0, la polaridad de salida no está invertida (“T”), lo que significa que el estado normal del pin de salida es un “nivel alto” y realiza pulsos con un “nivel bajo”. Este es el estado normal usado por el protocolo RS232 (Estándar recomendado 232, también conocido como RS-232), el cual utiliza niveles de voltaje de +12v para el 0 lógico y -12v para el 1 lógico. Por lo que para una conexión directa a la PC, es necesario un chip “adaptador de niveles”, el cual se encarga de convertir los niveles de voltaje TTL-RS232 y de RS232-TTL, uno de los chips más utilizado es el MAX232, debido a que utiliza un voltaje de alimentación de 5v.

Cuando el bit 1 es 1, la polaridad de la salida es invertida (“N”), significado que el estado normal del pin de salida es un “nivel bajo” y realiza pulsos con un “nivel alto”. Este estado es ideal para conectar directamente un display LCD ó un circuito TTL con lógica positiva.

El segundo comando que requiere el chip Picaxe1 de este proyecto es el comando “hserout”, el cual se encarga de transmitir datos vía serial a través del pin de salida, como se comentó en el comando “hsersetup”, permite 8 bits, sin bit de paridad y un bit de paro (n,8,1). Consta de la siguiente estructura:

HSEROUT break, ({#}data,{#}data...)

- ▲ Break – es una variable/constante (0 ó 1) que indica sí enviar ó no, una señal de corte (break o wake-up) antes de enviar datos. Para el caso de 8 bits de datos, la señal de corte envía 10 bits “0” consecutivos y es considerada como una señal de interrupción.
- ▲ Data – son variables/constantes (0-255), que esencialmente es la información de salida. El símbolo “#” se refiere a los datos de salida en forma decimal, para obtener enviar datos en código ASCII (Código estándar estadounidense para el intercambio de información, pronunciado [áski]), se encierra el texto con comillas (“Picaxe1”).

Detalles del comando:

La polaridad y la tasa de baudios se definen por el comando “hsersetup”, el cual debe ser configurado antes de usar este comando.

Para transmitir los datos existentes en una localidad, se inserta el símbolo “#” antes de la localidad, ejemplo, si el byte “b1” contiene el dato “123”, se denota como #b para obtener el dato “1” “2” “3”, por otro lado si se omite el símbolo se obtendrá el carácter correspondiente de 123 en código ASCII “{”.

Teniendo en cuenta los comandos anteriores, así como los comandos básicos de programación Basic de PICAXE, diseñé el siguiente programa

de transmisión vía serial del Picaxe1 al Picaxe2, para el envío de los mandos del brazo robótico a través del modulo de potencia.

;Picaxe1. Este programa envia los mandos a través del protocolo serial a los Picaxe2 y Picaxe3

hsersetup B9600_4, %00000 ; Tasa de baudios 9600 a 4Mhz (n,8,1)

main: ; Aquí se inicia la secuencia de pulsos cíclica.

high 0

gosub fila1

low 0

high 1

gosub fila2

low 1

high 3

gosub fila3

low 3

high 4

gosub fila4

low 4

goto main

fila1:

if pin0 is on and pinC.1 is off and pinC.6 is off then hserout 0,(1) end if

;Botón 1 mov izq antebrazo evitando presionar boton 3 mov der antebrazo

if pin1 is on and pinC.0 is off and pinC.6 is off then hserout 0,(2) end if

;Botón 2 mov arri evitando presionar boton 8 mov aba

if pin6 is on and pinC.0 is off and pinC.1 is off then hserout 0,(3) end if

;Botón 3 mov der antebrazo evitando presionar boton 1 mov izq antebrazo

if pinC.0 is off and pinC.1 is off and pinC.6 is off then return end if

goto fila1

fila2:

if pin0 is on and pinB.1 is on and pinC.1 is off and pinC.6 is off then hserout 0,(4) end if ;Botón 4 mov izq evitando presionar boton 6 mov der

if pin1 is on and pinB.1 is on and pinC.0 is off and pinC.6 is off then hserout 0,(5) end if ;Botón 5 movimientos libres del brazo con el teclado

```

if pin6 is on and pinB.1 is on and pinC.0 is off and pinC.1 is off then hserout
0,(6) end if ;Botón 6 mov der evitando presionar boton 4 mov 1zq
if pinC.0 is off and pinC.1 is off and pinC.6 is off then return end if
goto fila2
fila3:
if pin0 is on and pinB.3 is on and pinC.1 is off and pinC.6 is off then hserout
0,(7) end if ;Botón 7 cerrar efect evitando presionar boton 9 abrir efector
if pin1 is on and pinB.3 is on and pinC.0 is off and pinC.6 is off then hserout
0,(8) end if ;Botón 8 mov aba evitando presionar boton 2 mov arri
if pin6 is on and pinB.3 is on and pinC.0 is off and pinC.1 is off then hserout
0,(9) end if ;Botón 9 abrir efect evitando presionar boton 7 cerrar efector
if pinC.0 is off and pinC.1 is off and pinC.6 is off then return end if
goto fila3
fila4:
if pin0 is on and pinC.1 is off and pinC.6 is off then hserout 0,(10) end if
;Botón * inicia el programa del brazo
if pin1 is on and pinC.0 is off and pinC.6 is off then hserout 0,(11) end if
;Botón 0 detener y resetear cualquier operación
if pinC.0 is off and pinC.1 is off and pinC.6 is off then return end if
goto fila4

```

Notas del programa:

El programa está diseñado con el propósito de enviar mandos al chip Picaxe2, para operar el brazo robótico de forma libre (por control del usuario), a través de la botonera del chip Picaxe1. Por otra parte envía mandos a Picaxe2 para que este a su vez ordene al chip Picaxe3 comenzar con los procesos programados del brazo robótico.

5.3.4 Circuito electrónico del Picaxe1.

Con respecto a las especificaciones técnicas de los manuales de Microchip para los dispositivos Picaxe y con respecto al programa anterior, diseñé el circuito de la figura 5.9.

Es importante recordar que este circuito se encuentra separado del sistema general de control (Modulo de potencia, Picaxe2 y Picaxe3), ya que el

diseñar una cadena cíclica de pulsos positivos en cuatro de las salidas del microcontrolador que se conecten a las terminales de las filas o las columnas de la botonera (en este caso fueron las filas las que se conectaron al microcontrolador), para que de esta forma las columnas se conecten a cuatro de las entradas del microcontrolador, los cuales servirán como la referencia de cada tecla. De esta forma el microcontrolador reconoce en forma de coordenada que tecla fue la que se oprimió. Su diagrama se observa en la figura 5.11 y la botonera completa se puede observar en la figura 5.10.

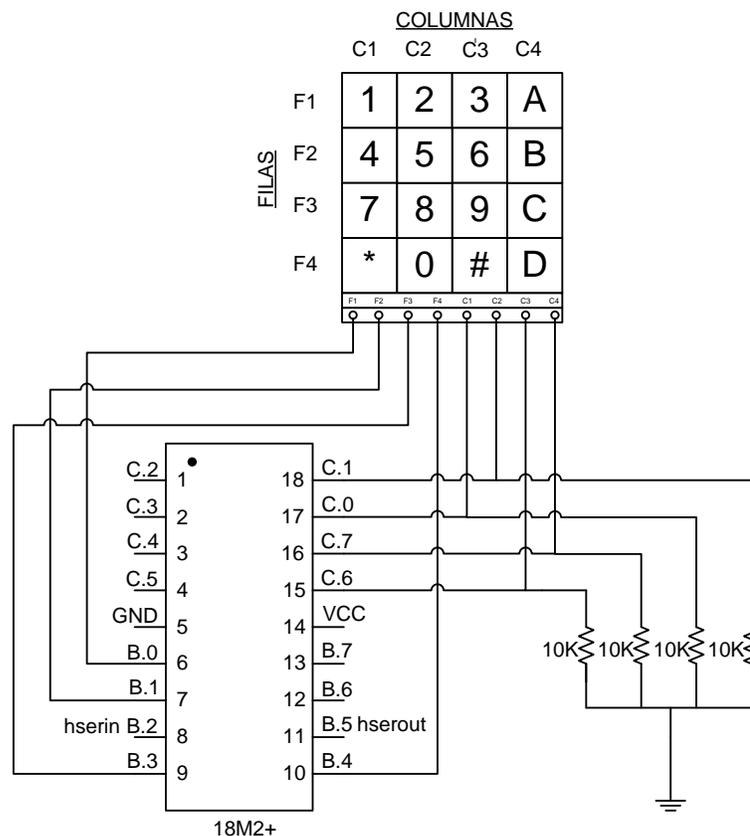


Figura 5.11 Diagrama de operación de la botonera del Picaxe 1.

5.3.5 Picaxe2.

Este chip se encarga de recibir la información vía serial proveniente del chip Picaxe1, para posteriormente enviar mandos al chip Picaxe3 (la conexión entre los chips Picaxe2 y Picaxe3 “no es vía serial”). Simultáneamente, este chip está encargado de realizar los mandos libres del brazo robótico por medio de la botonera de mandos incorporada al Picaxe1.

Para la conexión vía serial entre los dos primeros chips, existen diferentes medios de transmisión disponibles. Las señales pueden ser transmitidas, ya sea eléctricamente ó por pulsos de luz. Al seleccionar el medio apropiado, existen varios factores que deben tomarse en cuenta:

- Costos y esfuerzo de instalación.
- Seguridad de transmisión, susceptibilidad a interferencias, probabilidad de error, etc.
- Tasa de datos permitida.
- Distancias y posición topológica de los participantes, etc

Ningún medio cuenta con todas las características óptimas. La figura 5.12, muestra una sencilla comparación de las características entre los medios alámbricos.

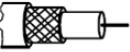
	par trenzado	cable coaxial	fibra óptica
diseño			
preparación para instalación	muy sencilla	sencilla	compleja
facilidad de instalación	muy sencilla	sencilla	sencilla, pero con dificultad en las esquinas
suceptibilidad a la interferencia	alta si no hay blindaje	baja	casi inexistente

Figura 5.12 Propiedades de los medios de transmisión alámbricos

En este caso en particular empleé cable coaxial RG-58U (impedancia de 50 Ω y conductor calibre 20AWG) por su nivel de confiabilidad para la transmisión de datos, así como su bajo costo, también tomando en cuenta que requerí de un par de metros para la conexión, sin embargo hay que tomar en cuenta que en un escenario real, la distancia requerida de cable coaxial para conectar a los participantes de la transmisión serial, no es

gravosa en cuanto a la atenuación de la señal, ya que generalmente el cable coaxial se emplea para la transmisión de datos en grandes distancias y para altas tasas de baudios, como es el caso de la TV por cable.

5.3.6 Programación del Chip 2

Como se comentó anteriormente se requieren de tres comandos principales para la transmisión vía serial, sin embargo siendo que el Picaxe2 funge como el receptor, requiere solo de dos comandos, “hsersetup” (el cual se explico anteriormente) y de “hserin.

El comando “hserin” el cual se encarga de recibir los datos vía serial a través del pin de entrada, como se comentó en el comando “hsersetup”, permite 8 bits, sin bit de paridad y un bit de paro (n,8,1). Consta de la siguiente estructura:

HSERIN var

- ▲ Var – Es la variable que va a almacenar ó recibir el byte de datos

Detalles del comando:

La polaridad y la tasa de baudios se definen por el comando “hsersetup”, el cual debe ser configurado antes de usar este comando.

Como se mencionó este como se encarga de recibir los bytes de entrada para almacenarlos en una variable, sin embargo los bytes pueden recibirse en cualquier momento durante la ejecución del programa Picaxe (no solo cuando el comando “hserin” lo requiera), esto debido a que los bytes son recibidos en 2 buffers FIFO (“first in – first out” ó primero en entrar – primero en salir, es una unidad de memoria que almacena información y la retiene hasta que el receptor está listo para procesar la información).

Tomando en cuenta los comandos principales necesarios para la transmisión de datos, así como de los demás comandos requeridos para realizar las tareas deseadas y tomando en cuenta que este chip se encarga de realizar movimientos libres del brazo (a petición del usuario), así como

enviar los mandos al siguiente chip para ejecutar los movimientos programados, diseñe el siguiente programa.

;Picaxe2. Este programa recibe los mandos del Picaxe1 por protocolo serial, realiza los movimientos libres del brazo y transmite el mando para la ejecución del programa al Picaxe3

```
hsersetup B9600_4, %00 ;Tasa de baudios 9600 a 4Mhz (n,8,1)
```

```
let dirsC = %00000011 ;Permite convertir C.0 y C.1 en salidas
```

```
daniel:
```

```
w1=0
```

```
hserin w1 ;hserin revisa la entrada de algún dato en la localidad w1, se dividen en tres opciones con el propósito de hacer más rápida la respuesta
```

```
if w1 = 10 then gosub paro ;Botón 0 de paro general (llama subrutina)
```

```
if w1 = 9 then gosub movlibre ;Botón 5 del teclado para utilizar los movimientos libres del brazo
```

```
if w1 = 11 then gosub programa ;Botón * inicia el programa del brazo
```

```
w1=0
```

```
goto daniel
```

```
movlibre:
```

```
hserin w1 ;Un solo comando hserin w1 en todo este grupo para que los pulsos a la salida no salgan retardados
```

```
if w1 =1 then high 0 else low 0 end if ;Motor 1 base mov izquierda y previene que se mantenga presionado
```

```
if w1 =2 then high 1 else low 1 end if ;Motor 1 base mov derecha
```

```
if w1 =3 then high 3 else low 3 end if ;Motor 2 mov arriba
```

```
if w1 =4 then high 4 else low 4 end if ;Motor 2 mov abajo
```

```
if w1 =5 then high 6 else low 6 end if ;Motor 3 antebrazo mov izquierda
```

```
if w1 =6 then high 7 else low 7 end if ;Motor 3 antebrazo mov derecha
```

```
if w1 =7 then high C.6 else low C.6 end if ;Efactor cerrar
```

```
if w1 =8 then high C.7 else low C.7 end if ;Efactor abrir
```

```
if w1 =10 then daniel ;Botón 0 detener cualquier operación
```

w1=0 ;Se asegura que que el registro w1 no guarde información no deseada
goto movlibre

paro:

```
let pinsc = %00000001 ;Botón 0 que enciende C.1 paro general
pause 300 ;Un lapso de tiempo para asegurar el paro
let pinsc = %00000000
return ;Regresa a donde se llamó a la subrutina
```

programa:

```
let pinsc = %00000010
pause 300
let pinsc = %00000000
return
```

5.3.7 Picaxe3.

Este chip se encarga de recibir los mandos directamente del chip Picaxe2 (no es vía serial), mandos que originalmente provienen del chip Picaxe1. De igual forma contiene el programa que realiza una secuencia de movimientos preestablecidos para el brazo robótico, la parte que quisiera remarcar, es que dicha secuencia de movimientos permite actuar al mismo tiempo todos los motores y el efector que conforman las articulaciones del brazo, pero recordando que paralelamente recibe los pulsos de los sensores magnéticos instalados en cada motor (reed swich RDS-1), esto con el propósito de realizar movimientos más exactos en los motores y sean capaces de regresar a la misma posición que tenían antes de efectuar el programa.

Esta situación representaba un reto en cuanto a la programación del chip se refiere, ya que se presento el problema de contar los pulsos de los tres sensores magnéticos y al mismo tiempo, realizar los movimientos de los motores a la espera del conteo final de los sensores que representarían el paro de los motores. Por lo que realizar un programa como este en un solo chip resultaba problemático, sin embargo la tecnología Picaxe cuenta con

una solución muy apropiada para este tipo de situaciones, y es el “proceso de tareas en paralelo”.

Es sabido que un microcontrolador Picaxe, es un controlador de un solo núcleo, por lo que solo puede procesar una sola instrucción a la vez. Pero existe una excepción a esta regla, se debe a que cuenta con un periférico separado dentro del chip, que funciona de forma independiente al núcleo central. Este genera pulsos completamente independiente a las tareas de procesamiento principal, por lo que no requiere ningún procesamiento por parte del núcleo para operar de forma alterna.

Tomando en cuenta lo anterior, el microcontrolador Picaxe (solo los chips M2) ahora puede “simular procesamiento en paralelo” al saltar repetidamente entre las tareas a una alta velocidad. Esto es posible debido al incremento de la velocidad de operación de los chips más recientes, por ejemplo, procesar cuatro tareas a 16 Mhz, es aproximadamente igual a procesar una tarea a 4 Mhz, por lo que todas las tareas parecieran estar siendo procesadas de forma paralela.

¿Cómo operan las tareas en paralelo?

Todos los chips M2 son capaces de operar tanto en el modo de una sola tarea como en el modo de tareas en paralelo. En el modo de una sola tarea, el chip M2 se comporta como tradicionalmente lo haría un microcontrolador Picaxe, de forma secuencial (línea por línea) como es de esperarse.

Al incluir las etiquetas “start” dentro del programa, el compilador del software automáticamente descarga el programa al chip en el modo de tareas en paralelo.

Los comandos dentro del chip son procesados de forma circular, por ejemplo, en el funcionamiento de dos tareas en paralelo el primer comando de “task0” se procesa inicialmente, después se procesa el primer comando de “task1”, posteriormente se procesa el segundo comando de “task0”, y así sucesivamente. Durante el tiempo de procesamiento de tareas “muertas”, como lo son los comandos de pausas o retardos, el núcleo se da cuenta automáticamente de que no hay ninguna tarea que procesar en ese momento por lo que salta a la

siguiente tarea. De esta manera las demás tareas no son afectadas por una pausa o retardo.

Todas las variables, memoria RAM y memoria EEPROM se comparten entre todas las tareas, por lo tanto, las tareas pueden interactuar unas con otras al transferir información entre ellas.

Suspender tareas.

Es posible suspender tareas durante la ejecución del programa a través del comando “suspend”. También se puede restablecer la tarea suspendida con el comando “resume”. Únicamente se debe tener la precaución de no suspender todas las tareas al mismo tiempo, ya que no habrá procesamiento alguno.

Limitaciones.

Las tareas en paralelo no están diseñadas para tareas complejas, donde cada tarea intenta usar diferentes características avanzadas simultáneamente como lo son, la transmisión vía infrarroja, protocolos de comunicación serial, etc. En estas situaciones es necesario usar un chip independiente para procesar cada característica avanzada.

Así mismo, si un programa requiere de una alta exactitud en el reloj, es recomendable utilizar una sola tarea.

5.3.8 Programación del Chip 3

El uso de un tercer chip en este proyecto (Picaxe3) para el proceso independiente del programa de movimientos del brazo, es debido a las limitaciones de operación que conllevan las tareas en paralelo. Pero tomando en cuenta que el tamaño de estos microcontroladores es muy reducido (18 pines) y que también tienen un costo bastante accesible, fue viable utilizar este tercer chip considerando las grandes ventajas que le proporcionaría a todo el sistema por la sola implementación de un chip extra.

Considerando el tipo de transmisión serial entre los dos primeros chips, los mandos que estos mismos envían a este chip e implementando las características de las tareas en paralelo, diseñé el siguiente programa para cumplir con las últimas funciones del sistema automatizado a base de los

microcontroladores Picaxe para el control del brazo robótico a través del modulo de potencia.

;Picaxe3. Este programa recibe el mando desde el Picaxe1 a través del Picaxe2 para ejecutar el programa del brazo

start0:

esperar:

if pin0 = 1 then reset end if ;Paro general proveniente del botón cero

if pin2 = 1 then continuar

goto esperar

continuar:

if pin2 = 1 then continuar

resume 1: resume 2:resume 3:resume 4

goto start0

start1: ; Motor 1 izquierda y derecha con sensor 1

suspend 1

low 1 ;Se asegura de que no estén activados los 2 sentidos al mismo tiempo

high 0

b0=0 ;motor 0 izquierda

fornext1: ;Se usa el mismo bucle for-next para ambos sentidos del motor

for b0 = 1 to 5 ; define el bucle para 5 tiempos

ciclo11:

if pin0 = 1 then ciclo12

goto ciclo11

ciclo12:

if pin0 = 1 then ciclo12

next b0 ; end of loop

low 0

if pinb.1 = 1 then apaga1d

pause 4000 ;espera 4 segundos (recordar que en este punto en vez de regresar a start, damos el sentido contrario al motor junto con su ciclo)

b0=0

high 1

goto fornext1

```
apaga1d:
low 1
low 0
pause 4000
goto start1 ; el bucle regresa a start1
```

```
start2: ; Motor 2 izquierda y derecha con sensor 2
suspend 2
low 3 ;Se asegura de que no estén activados los 2 sentidos al mismo
tiempo
high 2
b1=0 ;motor 0 izquierda
fornext2: ;Se utiliza el mismo loop for-next para ambos sentidos del motor
for b1 = 1 to 5 ; define el bucle para 5 tiempos
ciclo21:
if pin7 = 1 then ciclo22
goto ciclo21
ciclo22:
if pin7 = 1 then ciclo22
next b1 ; fin del bucle
low 2
if pinb.3 = 1 then apaga2d
pause 4000 ; espera por 4 segundos (recordar que en este punto en vez de
regresar a start, damos el sentido contrario al motor junto con su ciclo)
b1=0
high 3
goto fornext2
apaga2d:
low 3
low 2
pause 4000
goto start2 ; el bucle regresa a start2
```

```
start3: ; Motor 3 izquierda y derecha con sensor 3
suspend 3
low 5 ;Se asegura de que no estén activados los 2 sentidos al
mismo tiempo
```

```

high 4
b2=0 ;motor 0 izquierda
fornext3: ;Se utiliza el mismo loop for-next para ambos sentidos del motor
for b2 = 1 to 5 ; define el bucle para 5 tiempos
ciclo31:
if pin6 = 1 then ciclo32
goto ciclo31
ciclo32:
if pin6 = 1 then ciclo32
next b2 ; fin del bucle
low 4
if pinb.5 = 1 then apaga3d
pause 4000 ;espera 4 segundos (recordar que en este punto en vez de
regresar a start, damos el sentido contrario al motor junto con su ciclo)
b2=0
high 5
goto fornext3
apaga3d:
low 5
low 4
pause 4000
goto start3 ; bucle regresa a start3

start4: ;Efector abrir y cerrar (No se necesita el start3, el efector ira
dentro de start2 en el antebrazo porque él determina cuando se puede abrir
el efector, es una programación más sencilla porque el efector no cuenta
con sensor)
suspend 4
high 6
pause 3000
low 6
pause 4000
high 7
pause 3000
low 7
pause 4000
goto start4

```

5.3.9 Circuito electrónico Picaxe2 y Picaxe3.

Este circuito se encuentra incorporado dentro del modulo de potencia y lo diseñé considerando las especificaciones técnicas de Picaxe, así como las funciones principales, como lo son la recepción de datos seriales y la programación de los movimientos del brazo robótico, las cuales se encuentran resumidas dentro de los programas del Picaxe2 y Picaxe3.

Lo anterior recordando que el modulo de potencia cuenta con la opción tanto de operar directamente a la PC por medio del puerto paralelo, como operar con el sistema Picaxe. El circuito se puede observar en la figura 5.13.

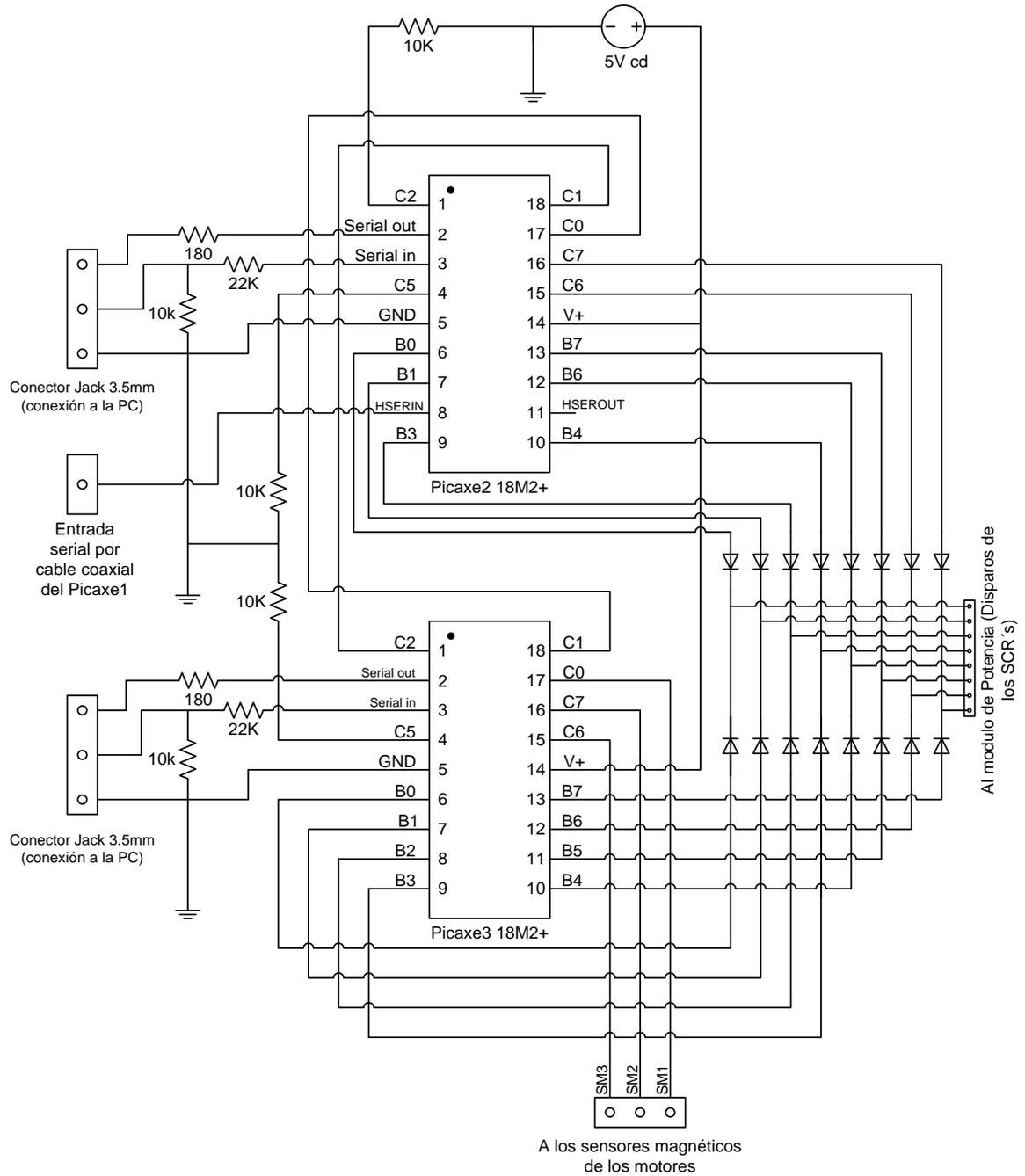


Figura 5.13 diagrama de conexión de Picaxe1 y Picaxe 2

Conclusiones

Un hecho implacable es que la tecnología en nuestra era evoluciona aceleradamente en comparación a décadas anteriores. Tecnologías aplicables a todo tipo de escenarios y necesidades se han hecho presentes en lugares en donde antes no lo imaginábamos. Un ejemplo de ello es la notable forma en cómo la automatización ha llegado a nuestros hogares cuando antes era un concepto propio de una industria.

No cabe duda que el PLC, no solo proporciona una solución, sino una infraestructura completa que la respalda, desde la certificación de una extensa lista de normas internacionales, hasta la asesoría técnica especializada en soluciones, ya no solo industriales sino a muchos otros niveles también, todo lo anterior a un determinado costo que en la mayoría de las ocasiones solo las grandes empresas son capaces solventar.

Sin embargo no nos dimos cuenta en qué momento fue que tecnologías como los microcontroladores o la computadora misma llegaron a niveles tan tecnológicamente avanzados y a costos realmente accesibles, que posteriormente se convertirían en herramientas a nuestro alcance e incluso de uso cotidiano para muchas personas como la computadora misma.

El objetivo general de este proyecto es “presentar un ejemplo desde mi perspectiva acerca de las capacidades de automatización que se pueden obtener en diversos entornos con herramientas que se encuentran a nuestro alcance, entornos como el área industrial o la vida cotidiana”, este proyecto no solo presenta un ejemplo, sino también una aplicación concretada. De esta forma es más fácil visualizar que si bien, en este caso se está controlando un pequeño grupo de motores, solo resta modificar la idea de cambiar los motores por otro tipo de maquinaria industrial. Inclusive se puede aplicar en una alarma residencial, un programador de riego o hasta un sistema de control para una casa inteligente. Todo lo anterior, a un precio bastante accesible.

Precisamente debido al bajo costo del sistema picaxe, así como su interface amigable, resultaría muy adecuado para implementarlo en los laboratorios de clase de esta Facultad, evitando así el riesgo de usar de forma principiante equipos costosos como los PLC's. De esta manera se puede desarrollar una serie de prácticas para un laboratorio de "introducción a la automatización" y que requeriría equipo electrónico con el que actualmente cuentan los laboratorios de electrónica de la Facultad. Dicho laboratorio sería de gran ayuda para la formación del alumno en la automatización.

Cabe señalar que la teoría necesaria para la realización de mi proyecto reúne diferentes asignaturas que cursé durante mi carrera. Electrónica de Potencia y Diseño Lógico en el desarrollo del Modulo de Potencia; Programación y Electrónica Digital en el desarrollo de la interfaz y programación de la PC; Laboratorio de Equipo Eléctrico y Transformadores y Motores de Inducción para el desarrollo un brazo robótico; Sistemas de Comunicaciones, Comunicaciones Digitales y Microprocesadores para el desarrollo del sistema de microcontroladores. Aunado al curso de PLC-I impartido también en la Facultad.

Por lo anterior opino que incluir un laboratorio de automatización en el plan curricular, incrementaría las capacidades y el perfil del egresado notablemente. Ampliando de esta forma, su visión en la aplicación de los conocimientos ya adquiridos en el transcurso de su formación académica.

Finalmente, considero que de la misma forma en que he adaptado un conjunto de elementos para una función específica como lo fue la automatización de este brazo robótico, es posible hacerlo en una gran diversidad de aplicaciones y entornos. Tomando en cuenta que el límite es nuestro propio ingenio e imaginación.

Bibliografía.

Grafham D.R. "SCR Manual General Electric" (Fifth edition). General Electric. 1972.

Belke R. E., Cleary J. F., Davidsohn U. S., Giorgis J., Gottlieb E. "General Electric Transistor Manual" (Seventh edition). J. F. Cleary Editions. 1970.

International General Electric Company "Electric equipment for Rod Mill". General Electric. 1965

Rashid H. Muhammad. "Electrónica de Potencia", (Tercera Edición). Pearson / Prentice Hall, 2004.

Fratschner O. "Elementos de Máquinas" (3ra edición). Gustavo Gili. 1980.

Dawes Chester L. "Tratado de Electricidad Corriente Continua". Gustavo Gili, 1962.

Murphy R. Robin. "Introduction to AI Robotics". Brandford Book - The MIT Press. 2000.

Addison Darrick. "Introduction to Robotics Technology IBM". IBM Editions. 2001.

Pires Norberto J. "Industrial Robots Programming: Building Aplications for the Factories of the Future". Springer Science+Business Media L.L.C. 2007.

"SIEMENS PLM Software's Robotics Simulation". Arc Advisory Group. 2008.

Microchip Technology Inc. "PIC Microchip DC Motor Control Tips & Tricks". 2004.

PICAXE Manual 1. Revolution Education Ltd., 2012. (Consulta: 8 junio 2012)
Disponible en: http://www.picaxe.com/docs/picaxe_manual1.pdf

PICAXE Manual 2. Revolution Education Ltd., 2012. (Consulta: 8 junio 2012)
Disponible en: http://www.picaxe.com/docs/picaxe_manual2.pdf

PICAXE Manual 3. Revolution Education Ltd., 2012. (Consulta: 8 junio 2012)
Disponible en: http://www.picaxe.com/docs/picaxe_manual3.pdf

PICAXE Programming Editor. Revolution Education Ltd., 2012. (Consulta: 14 mayo 2013) Disponible en:
<http://www.picaxe.com/Software/PICAXE/PICAXE-Programming-Editor/>

Cortés Manuel E., “León Iglesias Miriam. Generalidades sobre la metodología de la investigación”. Universidad Autónoma del Carmen Ciudad del Carmen. 2004.