



UNIVERSIDAD AUTÓNOMA DEL ESTADO
DE HIDALGO

INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍA

**ROBOT CARTESIANO: SEGUIMIENTO DE
TRAYECTORIAS IRREGULARES ARBITRARIAS
MEDIANTE COMPUTADORA**

T E S I S
QUE PARA OBTENER EL GRADO DE
INGENIERO EN ELECTRÓNICA Y
TELECOMUNICACIONES

P R E S E N T A

JOSÉ LUIS LÓPEZ SEGOVIA

MISAEAL ALAMILLA SANTIAGO

JUAN FRANCISCO DOMÍNGUEZ VÁZQUEZ

DIRECTOR DE TESIS
DR. OMAR DOMÍNGUEZ RAMÍREZ

Vo. Bo:

DR. OMAR DOMÍNGUEZ RAMÍREZ
DIRECTOR DEL PROYECTO DE TESIS

COASESORES DE TESIS
M. EN C. JESÚS PATRICIO ORDAZ OLIVER
M. EN C. MAYEN GICELA DÍAZ MONTIEL

PACHUCA DE SOTO, HIDALGO. NOVIEMBRE DE 2007

Agradecimientos

Gracias a nuestros padres por habernos apoyado en lo moral y económico, a nuestros hermanos por haber alentado a terminar este proyecto y gracias a nuestro asesor que es parte fundamental de este mismo.

Notación empleada en esta tesis

Notación	Significado
<i>CNC</i>	Máquinas de control numérico.
<i>LISP</i>	Lenguaje de programación, Acrónimo de LIStProcessing.
<i>PCB's</i>	Placas de circuito impreso.
<i>DXF</i>	Formato de gráfico vectorial.
<i>PPP</i>	Configuración Prismática, Prismática, Prismática.
<i>RPP</i>	Configuración Rotacional, Prismática, Prismática.
<i>PRP</i>	Configuración Prismática, Rotacional, Prismática.
<i>RRP</i>	Configuración Rotacional, Rotacional, Prismática.
<i>PTP</i>	Robots tipo Planar, Tornillo, Prismático.
<i>CP</i>	Robots Cilíndrico, Planar.
<i>PAP</i>	Abreviatura de referencia a motores paso a paso.
R_c	Resistencia En el colector de transistor.
V_{ce}	Tensión del colector al emisor
$C0 - C3$	Pines de disposición del 0 al 3 en el registro de control
<i>c.c.</i>	Corriente continua
<i>c.a.</i>	Corriente alterna

Índice general

1. Introducción	1
1.1. Antecedentes	1
1.2. Estado del arte	2
1.2.1. Mesa de posicionamiento X-Y	2
1.2.2. Construcción de una máquina para corte térmico de siluetas metálicas asistida por computador	3
1.2.3. Interprete de CAD y aplicación para optimización de parámetros de control de un robot industrial	5
1.2.4. Cortador ultrasónico de disco SPI	5
1.2.5. Mejora de la detección y caracterización de materiales con un sistema automático de ultrasonidos	6
1.2.6. Fischerscope: Análisis de materiales y medición del espesor de recubrimientos	8
1.2.7. Inspección y clasificación automática de pasta de papel TCF y EFC	9
1.3. Justificación	11
1.4. Planteamiento del problema	11
1.5. Solución propuesta	12
1.6. Objetivo general	12
1.7. Objetivos específicos	12
1.8. Contribución de la tesis	13
1.9. Organización del trabajo	13
2. Principios de robótica	15
2.1. Antecedentes	15
2.1.1. Las generaciones de los robots	17
2.1.2. Las leyes de la robótica	18
2.2. Robots manipuladores	18
2.2.1. Sistema mecánico	19
2.2.2. Actuadores	21
2.2.3. Sensores y sistemas de control	21
2.3. Robots móviles	22
2.4. Definiciones básicas en un robot	23
2.4.1. Conceptos de partes que conforman a un robot	23

2.4.2.	Tipos de articulaciones	25
2.5.	Configuraciones de robots	26
2.5.1.	Robot cartesiano	26
2.5.2.	Robot cilíndrico	27
2.5.3.	Configuración esférico o polar	27
2.5.4.	Configuración angular	28
2.5.5.	Configuración ESCARA	28
2.5.6.	Configuración paralela	28
2.6.	Sistemas de un robot	28
2.6.1.	La estructura	29
2.6.2.	Sistema de accionamiento	29
2.6.3.	Sistema de transmisión	29
2.6.4.	Sistema sensorial	29
2.6.5.	Sistema de control	29
2.6.6.	Sistemas de comunicación	29
2.7.	Cinemática de los robots	30
2.8.	Tipos de trayectorias	30
2.9.	Conclusiones	34
3.	Estudio de los robots: estructura y sistemas	35
3.1.	Introducción	35
3.2.	Anatomía del robot	36
3.2.1.	Configuraciones básicas de robots	36
3.2.2.	Movimientos del robot	37
3.2.3.	Volumen de trabajo	39
3.3.	Sistemas de impulsión de robot	40
3.3.1.	Velocidad	41
3.3.2.	Capacidad de carga	41
3.4.	Sistema de control	42
3.4.1.	Cuatro tipos de controles	42
3.4.2.	Velocidad de respuesta y estabilidad	43
3.5.	Precisión de movimiento	44
3.5.1.	Resolución espacial	45
3.5.2.	Exactitud	46
3.5.3.	Repetibilidad	47
3.5.4.	Sensores robóticos	48
3.6.	Sistemas de control y componentes	48
3.7.	Actuadores y componentes de realimentación	48
3.7.1.	Sensores de posición	49
3.7.2.	Codificadores	51
3.7.3.	Sensores de velocidad	52
3.7.4.	Actuadores	52

3.7.5.	Servomotores de c.a y otros tipos	58
3.8.	Sistemas de transmisión de potencia	58
3.8.1.	Engranés	58
3.8.2.	Tornillos de potencia	59
3.8.3.	Otros sistemas de transmisión	60
3.9.	Conclusiones	61
4.	Estructura mecánica del robot cartesiano	63
4.1.	Acero	63
4.1.1.	Clasificación del acero	64
4.2.	Aluminio	65
4.2.1.	Características físicas	66
4.2.2.	Características mecánicas	66
4.2.3.	Aplicaciones del aluminio	67
4.3.	Hule	67
4.3.1.	El hule natural	67
4.4.	Procedimientos de elaboración del prototipo	69
4.5.	Robot cartesiano	73
4.5.1.	Placa coll-roll	74
4.5.2.	Barras guías eje Y	74
4.5.3.	Barra guía eje X	75
4.5.4.	Soportes de engranes	75
4.5.5.	Soportes 416	76
4.5.6.	Soportes longitudinales	77
4.5.7.	Barra del motor principal	77
4.5.8.	Carro de desplazamiento	78
4.5.9.	Engranés	78
4.5.10.	Motor de movimiento en X	79
4.5.11.	Soporte de eje de motor principal	79
4.5.12.	Perno o cremallera	80
4.6.	Conclusiones	81
5.	Etapas de potencia	83
5.1.	Motores a paso	83
5.1.1.	Tipos de motores a pasos	84
5.1.2.	Parámetros de los motores a pasos	86
5.1.3.	Funcionamiento del motor bipolar	88
5.1.4.	Funcionamiento de motores unipolares	89
5.2.	Puentes H	92
5.2.1.	Funcionamiento	92
5.2.2.	Transistores	95
5.3.	Optoacopladores	97

5.3.1. Funcionamiento	97
5.4. Conclusiones	98
6. Programación y control	99
6.1. Visual Basic	99
6.1.1. Derivados	100
6.1.2. Ventajas	100
6.1.3. Inconvenientes	100
6.2. Ficheros GERBER y diseño de las PCB's	101
6.2.1. Configurando los archivos de salida	104
6.2.2. Documentación	110
6.3. Manejo del puerto paralelo	111
6.4. Manejo de Visual Basic	113
6.4.1. Programación orientada a objetos	114
6.4.2. Módulos de Visual Basic	116
6.4.3. Ratón	116
6.4.4. Matrices	117
6.4.5. Librería de E/S para Visual Basic	117
6.5. Funcionamiento del robot cartesiano	118
6.5.1. Interfaz	119
6.6. Conclusiones	128
7. Conclusiones	129
. Bibliografía	131
A. Glosario	135
B. Puerto paralelo	137
B.1. Puerto paralelo	137
B.1.1. Puerto paralelo en la Pc	139
B.2. Entradas y salidas por el puerto paralelo	142
B.2.1. Características E/S	142
C. Programa del robot	145
C.1. Declaración de variables y de la librería IO.dll.	145
C.1.1. Inicio de la aplicación.	145
C.1.2. Uso del eventoMouseDown de visual basic.	146
C.1.3. Uso del eventoMouseMove de visual basic.	147
C.1.4. Uso del evento MouseUp de visual basic	148
C.1.5. Función Reproducir() de la interfaz grafica.	148
C.1.6. Función Borrar() de la interfaz grafica	161
C.1.7. Función Salir() de la interfaz grafica.	161

Índice de figuras

1.1. Proyecto de mesa automatizada X-Y.[7]	3
1.2. Esquema del sistema para el corte térmico asistido por computador.[5] .	4
1.3. Lápiz utilizado en el extremo del robot sobre la mesa X-Y.[6]	5
1.4. Cortador Ultrásónico de Disco SPI.[8]	6
1.5. Esquema del sistema automático de medida.[9]	8
1.6. Microscopio de análisis de material.[25]	9
1.7. Inspección de impurezas de hojas en una mesa XY.[10]	10
2.1. Brazo robot industrial, de la marca ASEA).[29]	19
2.2. Segmentos y ejes del brazo robot RX60 Los distintos elementos del brazo son: La base (A), el hombro (B), el brazo (C), el codo (D), el antebrazo (E) y la muñeca (F) [30]	20
2.3. Ejemplo de robot móvil pequeño (microbot TRITTON).[4]	23
2.4. Diferentes tipos de articulaciones usadas en robots.[13]	26
2.5. Ejemplos de trayectorias continuas.[19]	31
2.6. Interpoladores lineales.[19]	32
2.7. Interpoladores a tramos[19].	33
3.1. Ejemplos de robots cartesianos.[28]	36
3.2. Ejemplos de configuraciones roboticas.[26]	37
3.3. tipos de articulaciones.[27]	38
3.4. Espacio de trabajo de brazo articulado y de robot cartesiano.[14]	39
3.5. Espacio de trabajo de los robots cilíndrico, polar angular y Scara.[14] .	40
3.6. Velocidad y estabilidad en robótica: (b) baja amortiguación: respuesta rápida y (a) alta amortiguación: respuesta lenta.[16]	44
3.7. Exactitud y resolución espacial en donde las inexactitudes mecánicas se representan por una distribución estadística.[16]	46
3.8. Repetibilidad y exactitud.[16]	47
3.9. Potenciómetro.[16]	50
3.10. Cilindro y pistón.[16]	54
3.11. Figura diagrama a bloques de un motor de c.c.[16]	56
3.12. Esquema de un motor a pasos.[23]	57
3.13. Motor a pasos de 4 hilos[23]	57
3.14. Tren de engranajes rectos[16]	59

4.1. Herramienta de maquila(Torno paralelo).[18]	70
4.2. Herramienta de perforación(Taladro).[20]	70
4.3. Herramienta de medición(Vernier).[22]	71
4.4. Fresadora universal.[21]	72
4.5. Robot Cartesiano	73
4.6. Placa de acero Coll-Roll	74
4.7. Barras guías del eje Y	75
4.8. Barras guías del eje X	75
4.9. Soportes de engranes	76
4.10. Soportes principales de las guías	76
4.11. Soportes longitudinales	77
4.12. Barra del motor principal	77
4.13. Carro de deslizamiento	78
4.14. Engranes del robot	78
4.15. Motor de movimiento del eje X	79
4.16. Soporte de barra de motor principal	79
4.17. Soporte de barra de motor principal	80
5.1. Funcionamiento de un motor paso a paso de reluctancia variable.[33]	85
5.2. Funcionamiento de un motor unipolar.[33]	86
5.3. Funcionamiento de un motor bipolar.[33]	86
5.4. Grados angulares por cada impulsión del motor[24]	87
5.5. Secuencia en un motor bipolar.[24]	88
5.6. Secuencia normal de un motor a pasos.[24]	89
5.7. Secuencia wave drive.[24]	90
5.8. Secuencia de medio paso.[24]	91
5.9. Puente H formado por transistores PNP y NPN.[36]	92
5.10. Circuito para avance del motor a pasos.[36]	93
5.11. Circuito para retroceso del motor a pasos.[36]	93
5.12. Circuito interlock que evita daños en los transistores.[36]	94
5.13. Circuito con diodos.[36]	95
5.14. Símbolos de transistores PNP y NPN	95
5.15. Esquema de un optoacoplador	97
6.1. Base de datos Post Process.[40]	102
6.2. visualización de la cara de componentes (TOP).[40]	103
6.3. Visualización de la cara TOP después de hacer visible el contorno de la placa.[40]	104
6.4. Archivos de salida.[40]	104
6.5. Cara de componentes (TOP)[40].	106
6.6. Cara de soldadura (BOTTOM)[40]	106
6.7. Plano GND[40]	107
6.8. Plano VCC[40]	107

6.9. Cara de la reserva de soldadura[40]	108
6.10. Plano de serigrafía(cara de componentes)[40]	108
6.11. Plano de ensamblaje (cara de componentes)[40]	109
6.12. Plano de taladro)[40]	109
6.13. Generación de informes que documenten el diseño.[40]	110
6.14. Archivo de taladro donde se muestran las diferentes coordenadas.[40]	111
6.15. Conectores DB25 hembra y macho.[47]	112
6.16. Disposición de pines de los tres registros[47].	113
6.17. Envío y recibimiento de datos manipulados por los métodos del objeto.[17]	115
6.18. Ejemplo de una matriz que contiene un conjunto de elementos[17]	117
6.19. Área de trabajo del Robot Cartesiano	118
6.20. Presentación de la interfaz gráfica del Robot Cartesiano	119
6.21. Interfaz gráfica del Robot Cartesiano	120
6.22. Diagrama de Clases del Robot Cartesiano.	121
6.23. Diagrama de procesos para crear la trayectoria.	122
6.24. Creación de la trayectoria	123
6.25. Creación de la trayectoria	124
6.26. Diagrama de procesos para reproducir la trayectoria	125
6.27. Ejemplo de una trayectoria con líneas rectas	126
6.28. Reproducción de trayectoria de líneas rectas por el Robot Cartesiano	126
6.29. Ejemplo de una trayectoria con líneas curvas	127
6.30. Reproducción de trayectoria de líneas curvas por el Robot Cartesiano	127
7.1. Robot Cartesiano	129
B.1. Descripción de los pines del Puerto paralelo.	138
B.2. Registro de datos y los pines correspondientes al conector DB-25.	139
B.3. Registro de estado y los pines correspondientes al conector DB-25.	140
B.4. Registro de control y los pines correspondientes al conector DB-25.	141
B.5. Líneas del conector DB-25 del puerto paralelo.	143
B.6. Niveles lógicos y de tensión del puerto paralelo.	144

Resumen

En consecuencia del porque de tomar este proyecto se hizo un estudio general de todo lo que se refiere a robótica y robots, desde sus sistemas en los que abarcan los actuadores, sensores, sistemas de transmisión de potencia en los que se encuentran los engranes, tornillos etc. además del estudio de las diferentes configuraciones de robots como sus tipos. Se estudia a grandes rasgos los tipos de motores, así como de los diferentes tipos de articulaciones que son movidas por los actuadores. En esta tesis se da a conocer desde el porque de tomar esta configuración de robot para el trazado de líneas y un bosquejo general de lo que es un robot: sus diferentes configuraciones, el espacio de trabajo, dependiendo de su tipo de configuración, los diferentes sistemas que lo componen, así como el funcionamiento de cada uno de estos. Podemos ver desde el cómo se se diseño y se maquinó cada uno de los componentes del robot cartesiano así como el diseño de la etapa de potencia para los movimientos de desplazamiento en los ejes X-Y. Este proyecto de robot tiene la finalidad de reproducir una trayectoria irregular arbitraria, que en muchas ocasiones este tipo de configuraciones robóticas solo son utilizadas en la ubicación de un punto en específico o para trazar solo líneas rectas. El principio que se manejo fue el de generación de comandos GERBER, que ayudan a la fabricación de PCB's (Placas de Circuito Impreso, por sus siglas en inglés). Para hacer que el robot tenga un control de sus movimientos hacia su área de trabajo se utiliza el puerto paralelo controlado a través de una interfaz grafica creada en Visual Basic que manda el código de pulsos hacia los motores de las articulaciones de este robot. En la interfaz grafica se cuenta con una pantalla de trazado, que con la ayuda del ratón podemos realizar un trazo curvo o lineal para posteriormente ser trazado, así como otra que indica el seguimiento del trazo de la trayectoria que siguen las diferentes líneas. A la vez que se ve la reproducción de la trayectoria en la segunda pantalla, el robot también recibe los datos y comienza a dibujar el trazo dando un dibujo similar al trazado en la primer pantalla.

Capítulo 1

Introducción

La robótica cada vez avanza a pasos agigantados donde podemos ver que casi cada medio año se muestra un nuevo robot el cual tiene movimientos muy autónomos en su funcionamiento propio, más sin embargo es necesario tener en cuenta los proyectos de robótica a los cuales se les puede considerar básicos, o bien si se les puede dar alguna otra aplicación. Es necesario tener en cuenta que tanto investigadores como alumnos de las diferentes universidades se vean interesados en modelos robóticos conocidos para hacer nueva investigación e innovar en métodos ya existentes.

En el uso de robots cartesianos se ha observado que son utilizados comunmente para apilamiento de estructuras de volumen uniforme, así como para aplicaciones dentro de la industria ya sea para tareas de soldadura, pintura, perforación de placas de acero, etc. En el presente trabajo se trata de utilizar una tecnica la cual no solo identifique una coordenada en un plano X-Y, sino que además sea capaz de seguir una trayectoria, la cual es formada por diferentes puntos en diferentes coordenadas.

Teniendo en cuenta que para el deslizamiento sobre los ejes X-Y, se usaron motores a pasos para dar una mayor exactitud de estos puntos. El control viene dado a través de un software programado en un lenguaje de alto nivel (Visual Basic 6), que manda pulsos de control por el puerto paralelo de un computador para llevar un control de movimiento de los dos ejes.

Es necesario hacer saber a los nuevos profesionistas que la robótica no va de algo que sea nuevo en su totalidad, sino que todo lo que conlleva a un nuevo proyecto tiene sus bases desde la robótica básica y para poder innovar es necesario estudiar lo ya creado.

1.1. Antecedentes

Los robots cartesianos en sus varias modalidades son utilizados en diferentes tareas dentro de la industria y la tecnología para lo cual estos son usados en ubicar una posición dentro del área de trabajo y colocar algún objeto en específico.

Por ejemplo algunos son utilizados en la ubicación de cajas, en los cuales primero se desplazan hacia una coordenada localizada en un eje X y posteriormente a un eje Y ,

teniendo ubicado el punto, el robot descarga el objeto mediante un efector final ubicado dentro de un eje de coordenadas Z .

Otro ejemplo de robot de este tipo son los utilizados en los famosos *tornos* para la perforación o taladrar orificios de placas de acero en un determinado punto. Cabe destacar que estas máquinas usadas en los tornos son manipuladas de manera manual. Dentro del estudio a nivel molecular una aplicación parecida a este se encuentra en los microscopios electrónicos que para ubicar una mejor visión se va desplazando hacia donde está el objeto o sustancia de estudio.

1.2. Estado del arte

Teniendo en cuenta que los robots cartesianos existentes, de tipo mesa X - Y , solo hacen la localización de un punto en específico para actuar sobre él, de manera que van actuando en un desplazamiento en el eje X para posteriormente desplazarse en un eje Y y en ocasiones en un eje Z , el robot solo hace el seguimiento de una trayectoria lineal.

Dentro de la realización de esta tesis lo que se pretende es que el robot pueda seguir cualquier tipo de trayectoria, para que vaya uniendo una infinidad de puntos continuos para dibujar una trayectoria visualizada desde el computador.

Para esto empezaremos por el estudio de este tipo de robots y sus diferentes aplicaciones tanto en la industria como en la investigación.

1.2.1. Mesa de posicionamiento X - Y

Este proyecto obedece a una configuración cartesiana con la finalidad de solucionar los problemas que se tienen en la precisión que se tienen en máquinas convencionales en el fresado y torno de herramientas, donde estas máquinas son manuales y totalmente mecánicas.

Este proyecto tiene la ventaja de adaptarse a las máquinas ya existentes. Este robot está pensado para ser una alternativa más económica frente a la adquisición de una máquina CNC puesto que no se estaría desechando las máquinas ya existentes en un taller.

Lo que propone este proyecto es reducir los tiempos en el maquinado de las piezas, es decir, evitar los tiempos muertos en los que los utilizamos para medir en un proceso no automatizado.

La solución planteada es un robot cartesiano de posicionamiento X - Y que permitirá que los procesos de fabricación puedan ser realizados de una manera cada vez más rápida y eficiente, en sus diversas aplicaciones se da el caso de un fresa convencional: al colocar el robot como mesa o bancada de la fresa, se obtuvo un sistema robusto y con

flexibilidad en trayectoria, pudiendo esta hacer los recorridos y figuras más estrictas y de alta precisión[7].



Figura 1.1: Proyecto de mesa automatizada X-Y.[7]

1.2.2. Construcción de una máquina para corte térmico de siluetas metálicas asistida por computador

En este apartado se analiza el corte de siluetas metálicas apoyándose de un software de diseño de siluetas para posteriormente cortar placas metálicas sin tener que trazar la figura de forma manual, tener equipo de protección y poder ahorrar el troquelado que en ocasiones es muy costoso.

Este sistema esta compuesto por un computador convencional con un sistema de software, un control y una mesa XY , para el desplazamiento del efector final que realiza el corte.

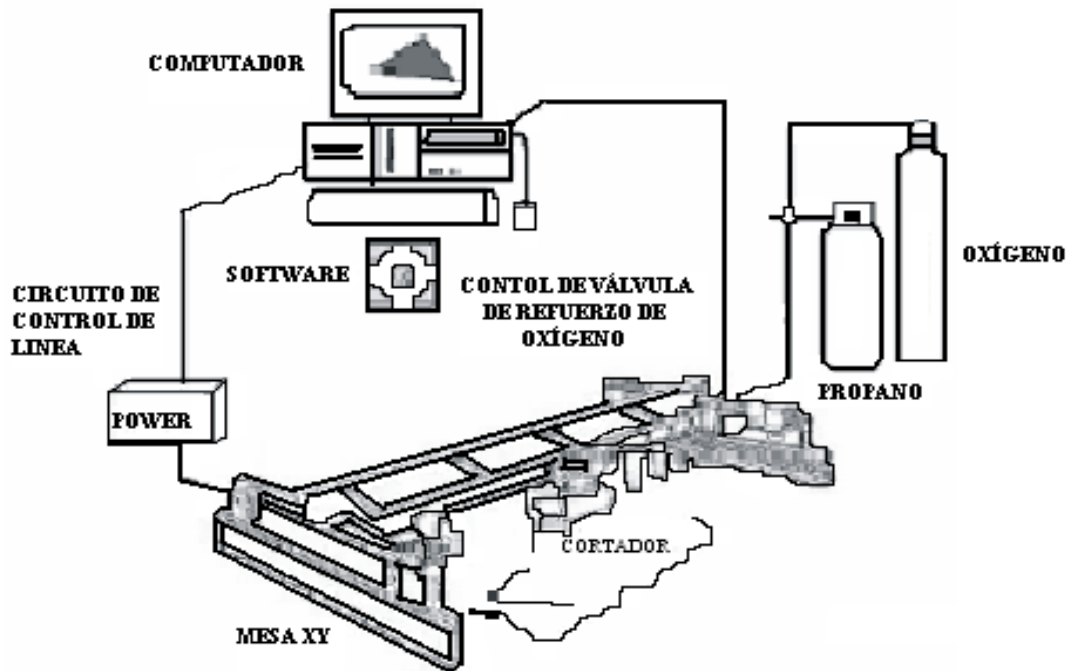


Figura 1.2: Esquema del sistema para el corte térmico asistido por computador.[5]

Uno de los subsistemas, llamado *Esbozar*, que conforma este proyecto realiza el dibujo en una interfaz gráfica dentro de un computador convencional, dentro de las herramientas que tiene, es poder dibujar la figura deseada y también diseñar a su vez el área de trabajo y darle color a la figura dibujada.

El subsistema de *procesamiento* reduce el tamaño de la figura poniendo el color de esta en un tono gris que dada la función $f(x,y)$ se obtenga una función $g(x)$ suavizada.

El de *segmentación* hace que el corte de la lámina se obtenga de una manera perfecta sin errores en el corte de las láminas.

En el subsistema de *descripción*, El software manda los datos del dibujo para que sea procesado en una mesa xy en el cual el cortador realizara el corte correspondiente. La ruta que el software mande hacia la mesa xy es determinada por los motores de los ejes[5].

1.2.3. Interprete de CAD y aplicación para optimización de parámetros de control de un robot industrial

En este proyecto se realiza el control de un robot de la marca ACMA, donado por la empresa Renault Argentina que pretende implementar un sistema de control para la generación de trayectorias desde un ambiente gráfico, en donde se opta por utilizar un ordenador, una PC, que además de ser un simple ordenador debe de mandar los datos correctos para que el control actúe de manera que la trayectoria sea la deseada.

Para la generación de estas trayectorias se utiliza una interfaz gráfica en un software de diseño de AutoCAD donde se implementa un lenguaje traductor que en este caso es el lenguaje Visual Basic en el que se pueden interpretar los datos obtenidos de la herramienta de autocad denominada Auto LISP.

Para lo anterior lo que se realice en AutoCAD se debe salvar con la extensión DXF para poder manipular el diseño en auto LISP y poder interpretar en Visual Basic. Para obtener la parte de interés, la trayectoria, se implemento en una mesa X-Y un papel para poder observar la trayectoria que se desea obtener a partir de la interfaz gráfica[6].

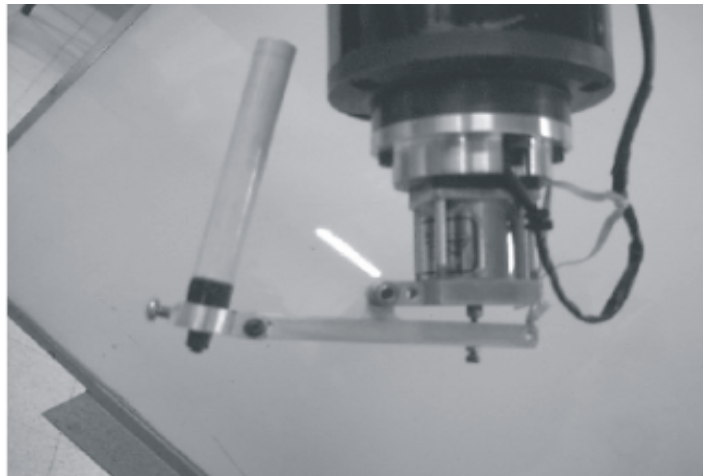


Figura 1.3: Lápiz utilizado en el extremo del robot sobre la mesa X-Y.[6]

1.2.4. Cortador ultrasónico de disco SPI

El Cortador ultrasónico de disco ha sido diseñado específicamente para cortar las pequeñas muestras que utilizan los microscopistas electrónicos. Requiere de un espacio pequeño en la mesa de trabajo y es capaz de cortar discos a partir de wafers tan delgados como $40\mu m$ y aún es suficientemente poderoso como para cortar en pocos minutos cilindros de 5 mm a partir de muestras más grandes[8].

Este cortador ultrasónico cuenta con una mesa posicionadora de tipo X-Y, donde son colocadas las muestras para posteriormente ser cortadas.

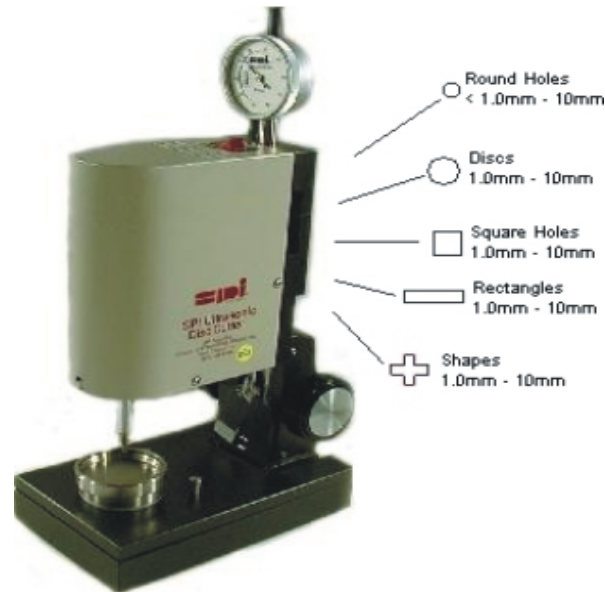


Figura 1.4: Cortador Ultrasónico de Disco SPI.[8]

La muestra se adhiere a la mesa X-Y, misma que está colocada sobre una platina magnética abajo del microscopio. La mesa X-Y se mueve sobre la platina magnética hasta que la porción de muestra con la que se necesita trabajar está centrada en el campo de visión del microscopio. La base magnética y el espécimen entonces son trasladados al cortador ultrasónico de disco y presionados firmemente contra dos tornillos posicionadores en la plataforma de carga, de tal forma que el eje del cortador corresponda exactamente con el eje óptico del microscopio. La atracción magnética de la base a la plataforma del espécimen asegura que el espécimen no se mueva y que la sección de interés se mantenga en el centro del disco de corte.

1.2.5. Mejora de la detección y caracterización de materiales con un sistema automático de ultrasonidos

Se presenta un sistema automático de inspección de materiales por ultrasonidos basado en el plano cartesiano XY. Con este se pretende mejorar el rendimiento en el procesamiento y la precisión en variedad de operaciones de inspección por ultrasonidos. Los campos principales de aplicación de máquinas de medidas en un plano coordenado XY son: inspección en control de calidad, aplicaciones en medidas precisas, inspección masiva de productos, ingeniería reversa, y procesos de medición automáticos para reducir errores y costos de operación humanos.

Toda automatización en un sistema ultrasónica se encuentra con el problema del acoplamiento entre el sensor ultrasónico y la pieza o material a inspeccionar. Uno de los sistemas más utilizados es el acoplamiento directo entre el sensor y el material, utilizando un gel como elemento de adaptación de impedancias. Este sistema ofrece la máxima transmisión de energía ultrasónica, pero obliga en un sistema automático a controlar la aproximación del sensor a la pieza e introduce incertidumbre entre medidas. Esto lo podemos encontrar en la bibliografía[9].

Por otro lado, existe el acoplamiento en inmersión donde la pieza a analizar y el sensor o sensores ultrasónicos son sumergidos en un tanque de agua. El sensor no está en contacto directo con el material, sirviendo el agua como acoplante o vehículo de guiado del haz ultrasónico. Esta técnica ofrece ciertas ventajas frente a la anterior como puede ser una mayor facilidad de automatización y evita que la pieza tenga que presentar una superficie plana regular. Como contrapartida presenta un nivel de energía inyectado menor debido a la reflexión que se produce entre la interfaz agua-material.

El sistema esta compuesto por diferentes módulos que son: una mesa XY, un computador, un equipo de ultrasonidos, un equipo de digitalización y el control electrónico de la mesa.

La mesa XY desplaza un cabezal en dos direcciones que corresponden a los ejes X y Y. En este cabezal se instala un transductor de ultrasonidos. el cual viene a ser el efector final en con una coordenada en Z, el cual puede ser desplazado manualmente.

En la figura 1.5, puede apreciarse el tanque de inmersión donde se colocan las probetas de material que van a ser inspeccionadas. La película de agua que queda entre el sensor y el material sirve como medio de acoplamiento para la transmisión de los ultrasonidos desde el sensor al material.

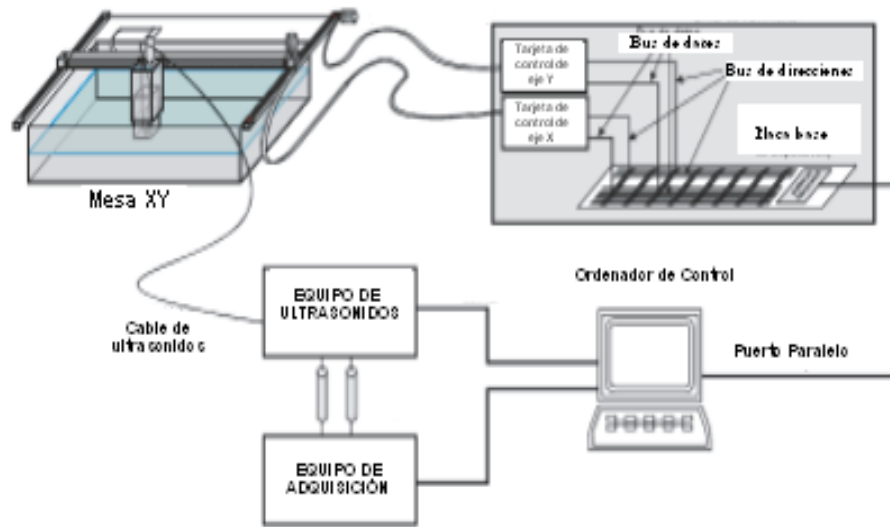


Figura 1.5: Esquema del sistema automático de medida.[9]

1.2.6. Fischerscope: Análisis de materiales y medición del espesor de recubrimientos

Este tipo de dispositivo además de analizar las sustancias de los materiales también en el estudio de las mismas verifica la dureza y algunas propiedades como la elasticidad, la permeabilidad y algunas otras. Permite estudiar el recubrimiento de los materiales y colocar piezas de tamaño pequeño.

Dentro del análisis de las sustancias se ayuda de una pequeña mesa X-Y que desplaza el objeto hacia el punto de enfoque y estos datos obtenidos los transite hacia un computador para ser analizados de manera grafica. En cuanto el análisis también pueden hacer modificaciones en el objeto, las cuales algunas pueden ser como el recubrimiento de este.



Figura 1.6: Microscopio de análisis de material.[25]

1.2.7. Inspección y clasificación automática de pasta de papel TCF y EFC

El proyecto desarrollado automatiza el procedimiento manual usado hasta ahora para la detección y medición de impurezas en hojas de pasta de papel. Para la detección de dichas impurezas se analiza cada una de las caras de la hoja, las cuales son iluminadas a contraluz según la norma ENCE 404, etiquetándose los defectos y definiendo la limpieza de la pasta. Esta normativa es más restrictiva que la norma internacional TAPI sobre limpieza en pasta de papel. Mayor información en la bibliografía[10].

En el sistema de inspección automática consta de las siguientes partes:

- Sistema mecánico: Mesa X-Y que desplaza el papel ante el sistema de adquisición de imágenes.
- Sistema de iluminación: 4 Lámparas de halogenuros metálicos de PHILIPS, modelo MASTERCOLOR PAR30 grados L, con pantallas lumínicas.
- Sistema de adquisición de imágenes: Iluminación 4 cámaras matriciales entrelazadas WATEC 902A y tarjeta de adquisición PULSAR-LC de Matrox.
- Sistema informático: Aplicación MMX.

Dentro del sistema mecánico, que engloba a la mesa XY. La adquisición de los datos se toma sobre la cara iluminada con mesa XY parada. Una vez almacenadas las imágenes digitales, la mesa se desplaza a la siguiente posición y se aplica simultáneamente la ejecución del algoritmo de detección de los defectos. El algoritmo de inspección tarda 41 segundos por cada cara.

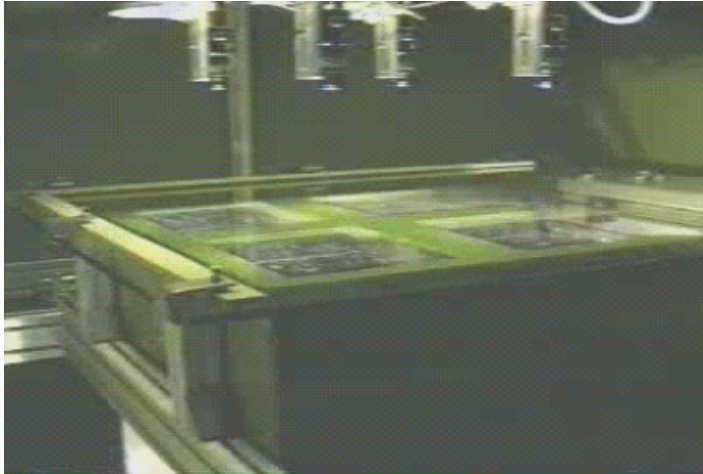


Figura 1.7: Inspección de impurezas de hojas en una mesa XY.[10]

La detección de los defectos se basa en filtros paso bajos y segmentación mediante modelos de la función de densidad del fondo. Los defectos se clasifican por su tamaño y acabado el algoritmo de inspección se determina la limpieza en pasta a partir del gramaje de la misma.

1.3. Justificación

Generalmente, una tarea de planificación suele tener tres tipos de resultados los cuales son: rutas, caminos y trayectorias.

Una ruta es un conjunto ordenado de configuraciones que deben ser alcanzadas por el robot. Por camino se entiende la discretización de una función continua que interpola las configuraciones definidas en una ruta. Finalmente, cuando se habla de trayectoria temporal (por omisión simplemente trayectoria) se está haciendo referencia a un camino que tiene asociado un perfil cinemático; es decir, a cada configuración perteneciente al camino se le asocia una velocidad. El resultado de la planificación se utiliza en sistemas de navegación planificada[1].

Una característica que arroja a todos es que se debe de contar con un controlador, en tiempo real, que permita que el sistema siga la trayectoria previamente obtenida. Todo dependerá del tipo de controlador para planificar caminos o trayectorias. En esta tesis, se entiende indistintamente por planificar, obtener un camino o una trayectoria, cuya ruta está definida por una configuración inicial y una configuración final.

Cuando se trata de generar trayectorias o caminos en sistemas no holónomos, hay una característica relevante que diferencia éste de otros problemas de planificación. Existe una amplia bibliografía que detalla las propiedades más significativas de estos sistemas [2]. Quizás la característica más sobresaliente esté vinculada al tipo de trayectorias que pueden seguir estos sistemas. En efecto, una configuración inicial y una final no pueden unirse mediante cualquier trayectoria. Las restricciones cinemáticas del sistema imponen unas condiciones que sólo algunos caminos cumplirán. En dichos casos, el problema de planificación consiste en encontrar un camino que conecte la configuración inicial con la configuración final y que, además, a lo largo del mismo se satisfagan las restricciones no holónomas[1].

En el proyecto que nosotros planteamos servirá para futuros trabajos en los cuales se desee implementar alguna aplicación industrial que comunmente se les asignan a brazos robóticos y vean la eficacia de estas configuraciones robóticas.

1.4. Planteamiento del problema

El Robot Cartesiano es utilizado para tareas en la industria tanto en ensamble, pintura, soldadura y paletización de objetos. Este tipo de robots manejan planos x, y, z y hasta llegan a manejar planos x, y, z, r . De acuerdo con el uso de los robots X-Y, solo pueden ser utilizados para ubicar un punto en determinadas coordenadas en un plano. La mayoría de los robots solo actúan para desplazarse en puntos específicos pero la mayoría no sigue un desplazamiento de forma paralela en ambos ejes para formar trayectorias curvas, solo se desplazan primero en un eje y posteriormente en el otro.

Teniendo en cuenta esto la interrogativa llevará a si un robot cartesiano X-Y, puede seguir y reproducir una trayectoria irregular arbitraria. Dependiendo del control apropiado que se aplique a los actuadores se obtendrán los resultados.

1.5. Solución propuesta

La solución que se propone para hacer un seguimiento de una trayectoria por medio del uso de un robot de configuración cartesiana, siguiendo el principio de comandos Gerber para la impresión de placas de circuitos. La solución es utilizar motores a pasos los cuales son necesarios para tener menos fricción y un control menos complicado. Además los motores a pasos pueden ser controlados desde el puerto paralelo de la computadora a través de una interfaz gráfica creada en Visual Basic.

El control de el seguimiento de las coordenadas se hace con la programación de un tren de pulsos que obedece a los diferentes puntos ubicados en la trayectoria y teniendo la localización del primer punto y el último a trazar, para que al terminar una trayectoria pueda localizar el próximo punto a trazar.

1.6. Objetivo general

Diseñar la estructura del robot cartesiano que sea capaz de recibir datos desde una computadora a través del puerto paralelo por medio de una interfaz gráfica que a su vez sea diseñada en el lenguaje de programación Visual Basic. Una vez recibiendo los datos el robot comenzará a trazar la trayectoria dibujada en el interfaz gráfica. De manera que la trayectoria dibujada coincidirá con la trazada en la computadora.

1.7. Objetivos específicos

- Realizar una etapa de potencia la cual sea confiable para el funcionamiento óptimo de los motores a pasos utilizados en el robot.
- Diseñar un sistema mecánico del robot lo mayor preciso para evitar fricciones en las barras de desplazamiento así como el dentaje que se le dé a los engranes para el desplazamiento en ayuda con las bandas acopladas a los dos ejes.
- Hacer que la interfaz gráfica pueda mandar y recibir datos de la trayectoria para poder visualizar la reproducción de esta misma.

1.8. Contribución de la tesis

Esta tesis servirá para aquellos estudiantes y profesores de las diferentes universidades y preparatorias que deseen participar en el mejoramiento de un robot de configuración cartesiana, haciendo en ellos la implementación de nuevas aplicaciones dentro de la industria, casa u oficina.

El proyecto pretende tener un sistema de control hecho solamente en un lenguaje de programación que ayudará a aquellos programadores que sepan manejar el lenguaje de programación Visual Basic. De tal manera que tendrá un control desde una computadora solo utilizando un tren de pulsos configurado en la salida del puerto paralelo.

El manejo de este robot ayuda a no depender de programas de trazados de líneas o vectores tales como el CAE, CAD, CAM, etc. En el trazado de trayectorias se ha venido utilizando estos robots solo para trayectorias lineales el cual nuestro proyecto viene a dar una innovación en cuanto al trazado de líneas curvas además de poder ubicar una coordenada inicial y una final además de poder localizar una segunda trayectoria dentro del mismo plano para ser trazada.

En la etapa de potencia ayudará a tener un mejor control de motores con el uso de puentes H en cada bobina para dar un desplazamiento correcto por cada pulso. De esta manera el circuito estará protegido con diodos para evitar cortos circuito y afecte a nuestro computador.

En general este proyecto viene a hacer parte de una manera de hacer trazos de líneas, ya sea curvas o rectas, con un robot de configuración X-Y. Este proyecto ayudará a estudiantes que se vean interesados en seguir y buscar una manera de hacer trazos más eficientes con este tipo de robot además de que se le podrá dar una aplicación diferente a las que hasta ahora se les ha dado. Una posible aplicación de este es el cortado de figuras artísticas en lámina, vidrio o cualquier otro material en el que se desee dibujar o trazar, utilizando ya en adición un tipo de láser que ayude a estas tareas.

1.9. Organización del trabajo

La organización de la tesis es la siguiente:

En el Capítulo 2, se presenta la historia de la robótica dentro del mundo de la tecnología. Los diferentes tipos de robots, topologías, se presentan a grandes rasgos los sistemas por los que están conformados y una descripción breve de ellos. Definiciones básicas en terminos de robótica.

Dentro del Capítulo 3 se muestra una descripción más general de los sistemas que conforman a los robots, sus topologías, así como la descripción de cada parte que conforma a cada sistema. Además el análisis de los movimientos que interviene en los robots y los parámetros que influyen en el proceso de movimiento.

En el capítulo 4, se muestra cada una de las partes de la estructura del robot, así como el material que fue utilizado para la elaboración de cada una de las piezas, no olvidando la parte del maquilado de las mismas.

En el capítulo 5, analizamos la etapa de potencia del circuito que controla a la parte mecánica. Donde se mencionan el funcionamiento de cada componente así como la función que tiene dentro de los puentes H, para el control de los motores a pasos.

En el capítulo 6, se da el detalle de como se programó los movimientos del Robot Cartesiano a través de Visual Basic en el manejo del puerto paralelo en conjunto con la etapa de potencia y la estructura mecánica del robot.

En el capítulo 7, se da a conocer las aportaciones de este proyecto, además de una solución en posibles trabajos a futuro con este tipo de robot.

Capítulo 2

Principios de robótica

En este capítulo veremos las partes fundamentales que componen a un robot tales como su sistema de control, sistema electromecánico y su sistema de programación.

En los siguientes subcapítulos veremos desde la definición de la palabra *robot* y desde que época se empezó a dar un conocimiento de estas máquinas tecnológicas.

Así también estudiaremos brevemente la morfología de ellos, la clasificación de cada uno y las diferentes aplicaciones que tienen dentro de la industria.

2.1. Antecedentes

La palabra *Robótica* fue acuñada por Isaac Asimov para describir a las nuevas máquinas llamadas robots. Recientemente se ha explotado al máximo a los robots que ya se ha llegado a hablar de la era de los robots.

El término robótica es definido desde diversos puntos de vista:

- Con independencia respecto a la definición de robot: La Robótica es la conexión inteligente de la percepción a la acción[11].
- En base a su objetivo: La Robótica consiste en el diseño de sistemas. Actuadores de locomoción, manipuladores, sistemas de control, sensores, fuentes de energía, software de calidad, todos estos subsistemas tienen que ser diseñados para trabajar conjuntamente en la consecución de la tarea del robot[12].

¹

Varias películas han seguido mostrando a los robots como máquinas que pueden dañar y ser una amenaza para el mundo en un futuro. Sin embargo, películas recientes, como la saga de *La Guerra de las Galaxias* desde 1977, retratan a robots como ayudantes del hombre.

¹ La palabra robot fue usada por primera vez en el año 1921, cuando el escritor checo Karel Capek (1890 - 1938) estrena en el teatro nacional de Praga su obra *Rossum's Universal Robot* (R.U.R.). Su origen es de la palabra eslava *robot*, que se refiere al trabajo realizado de manera forzada[32]

Algunos de los expertos en Robótica han estudiado los términos y no llegan a la conclusión de utilizar un termino universal entonces utilizan las siguientes definiciones:

- Ingenio mecánico controlado electrónicamente, capaz de moverse y ejecutar de forma automática acciones diversas, siguiendo un programa establecido.
- Máquina que en apariencia o comportamiento imita a las personas o a sus acciones como, por ejemplo, en el movimiento de sus extremidades.
- Un robot es una máquina que hace algo automáticamente en respuesta a su entorno.
- Un robot es un puñado de motores controlados por un programa de ordenador.
- Un robot es un ordenador con músculos.
- Un robot, es un agente artificial mecánico o virtual. Es una máquina usada para realizar un trabajo automáticamente.

Es difícil tener una definición exacta de la palabra robot sin embargo todos los términos anteriores apuntan a lo mismo. Quizás, Joseph Engelberg (padre de la robótica industrial) lo resumió inmejorablemente cuando dijo: Puede que no se capaz de definirlo, pero sé cuándo veo uno.

La imagen del robot como una máquina a semejanza del ser humano, aparece en el hombre desde hace muchos siglos, existiendo diversas realizaciones con este fin.

Uno de los avances más conocidos y utilizados es sin duda el computador, el cual nos ayuda a las diferentes tareas que tenemos, ya sea en el hogar, en la oficina o en la propia escuela. Éste se ha introducido hoy en día en su versión personal en multitud de hogares, y el ciudadano medio va conociendo en creciente proporción, además de su existencia, su modo de uso y buena parte de sus posibilidades. Pero dejando de lado esta verdadera revolución social.

Pero el robot industrial, que se conoce y emplea en nuestros días, no surge como consecuencia de la tendencia o afición de reproducir seres vivientes, sino de la necesidad de realizar tareas de gran esfuerzo sin llegar tener un desgaste físico a lo que los humanos podemos tener después de largas horas de trabajo, "estrés". Fue la necesidad la que dio origen a la agricultura, el pastoreo, la caza, la pesca, etc. Más adelante, la necesidad provoca la primera revolución industrial con el descubrimiento de la máquina de vapor de Watt y, actualmente, la necesidad ha cubierto de ordenadores la faz de la tierra.

Inmersos en la era de la informatización, la imperiosa necesidad de aumentar la productividad y mejorar la calidad de los productos, ha hecho insuficiente la automatización industrial rígida, dominante en las primeras décadas del siglo XX, que estaba

destinada a la fabricación de grandes series de una restringida gama de productos. Hoy día, más de la mitad de los productos que se fabrican corresponden a lotes de pocas unidades.

Al enfocarse la producción industrial moderna hacia la automatización global y flexible, han quedado en desuso las herramientas que hasta hace poco eran habituales:

- Forja, prensa y fundición
- Esmaltado
- Corte
- Encolado
- Desbardado
- Pulido.

Finalmente, el resto de los robots instalados en 1979 se dedicaban al montaje y labores de inspección. En dicho año, la industria del automóvil ocupaba el 58 por ciento del parque mundial, siguiendo en importancia las empresas constructoras de maquinaria eléctrica y electrónica. En 1997 el parque mundial de robots alcanzó la cifra de aproximadamente 830.000 unidades, de los cuales la mitad se localizaba en Japón[32].

2.1.1. Las generaciones de los robots

Se ha entrado en debate en cuanto a la aparición de la Robótica dentro de las etapas de la tecnología, pues algunos determinan que esta consta de tres generaciones y para otros como T.M Knasel destacan cinco generaciones de robots.

Hacia los años de 1950 se dieron los primeros pasos para la innovación de robots con la construcción de tele operadores ó sistemas dirigidos remotamente y surgen los manipuladores programables con control. La innovación de los primeros robots industriales, con ellos aparece la primera generación de robots hacia el año de 1960, los cuales eran capaces de repetir una secuencia independientemente de que variará el entorno exterior, son los robots manipuladores programables y servo controlados. Veáse bibliografía[32].

La segunda generación de robots se caracteriza por la capacidad de interacción con el entorno de trabajo. Tienen la capacidad de realizar diferentes operaciones como procesamiento de la información captada por sensores de visión y tacto y tomar alguna decisión de ejecución, son los robots del presente.

Los robots del futuro, los cuales se están diseñando y construyendo en la actualidad, con capacidad de adaptarse al entorno, generar sus propios planes de acción, interactuar con el entorno, además contará con un sistema de aprendizaje que le permite superar situaciones imprevistas, es el denominado robots de la tercera generación.

La T.M Knasel clasifica a los robots en cinco generaciones las cuales inician con la primer generación en 1982 con el nombre de *Pick and Place*, que tenían un control de final de carrera, que fue implementado en el servicio de las máquinas industriales. La segunda generación se caracterizó por contar con controles definidos como el servocontrol, un programa con parámetros y de trayectoria continua y capacidad de desplazarse sobre una vía. Estos fueron empleados en trabajos de soldadura y pintura; y aparece en el mercado en 1984.

En el año de 1989 aparece la tercera generación, con un control más avanzado que en las anteriores generaciones, en donde se implementa los servos de precisión, los sensores de visión y tacto, la programación *off-line*, guiado por vía AGV. Este empleado en el ensamblado en algunas empresas. La cuarta generación empieza desde el año 2000. Es montado sobre ruedas ó con piernas artificiales, posee sensores inteligentes, lleva como nombre móvil y se emplea en la industria de la construcción y en algunos procesos de mantenimiento en las empresas. Los inovadores pretenden mostrar al mundo la quinta generación de robots, los cuales tendrán controladores basados en inteligencia artificial, se conocerá como un androide saltarín, el cual será empleado en la industria militar.

2.1.2. Las leyes de la robótica

En Octubre del año de 1945 el escritor ruso Isaac Asimov escribe las tres leyes de la robótica y las enuncia en la revista *Galaxy science Fiction* definidas así:

1. Un robot no puede perjudicar a un ser humano, ni con inacción permitir que un ser humano sufra daño.
2. Un robot ha de obedecer las órdenes de un ser humano, excepto si tales ordenes entran en conflicto con la primera ley.
3. Un robot debe proteger su propia existencia, mientras tal protección no entre en conflicto con la primera ley ó la segunda ley.

2.2. Robots manipuladores

La mayoría de los robots manipuladores son de tipo de brazo articulado. De acuerdo con la RIA(Robot Institute of America), un robot industrial es robot manipulador programable multifuncional diseñado para mover materiales, piezas, herramientas o dispositivos especiales, mediante variados movimientos en la ejecución de distintas tareas.

El concepto *programable* dentro de la robótica es básico, ya que la realización de funciones de control por medio de programa ofrece una mayor flexibilidad y la posibilidad de implantar funciones complejas necesarias para controlar un manipulador[3].



Figura 2.1: Brazo robot industrial, de la marca ASEA).[29]

2.2.1. Sistema mecánico

El sistema mecánico está compuesto por varias articulaciones. Normalmente se distingue entre el brazo y el órgano terminal o efector final, el cual puede ser intercambiable por pinzas o fingers, para distintas tareas. El aumento de un mayor número de articulaciones ayuda a tener mayor maniobrabilidad del brazo pero dificulta el problema de control, obteniéndose normalmente menos precisión. Muchos de los robots industriales cuentan con menos de los seis grados de libertad que deben tener en la rotación o traslación para posicionar el órgano terminal en su punto final de trayectoria. Sin embargo, también se desarrollan robots industriales con múltiples y redundantes articulaciones para aplicaciones en áreas de trabajo difíciles de acceder. Entre estos cabe destacar los famosos robots tipo serpiente. De la misma manera, se investiga en robots flexibles que tengan mayor alcance con un peso mínimo.

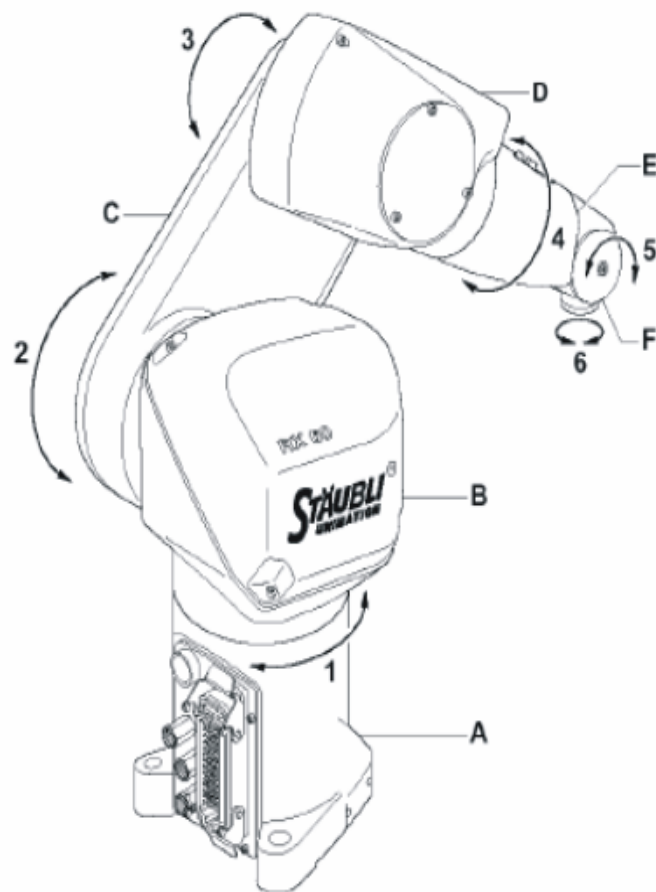


Figura 2.2: Segmentos y ejes del brazo robot RX60 Los distintos elementos del brazo son: La base (A), el hombro (B), el brazo (C), el codo (D), el antebrazo (E) y la muñeca (F) [30]

Cabe destacar que las ecuaciones que describen el movimiento del brazo articulado son ecuaciones diferenciales no lineales y acopladas, para las que en lo general, resulta difícil obtener soluciones analíticas. Los terminos de acoplamiento físicamente son los pares gravitacionales que dependen de la posición de las articulaciones, pares de reacción ocasionados por la aceleración de otras articulaciones, pares debidos a la aceleración de Coriolis y fuerzas centrífugas. La magnitud de estas interacciones depende de las características del brazo y de la carga.

2.2.2. Actuadores

la función de los actuadores es generar los pares necesarios para mover la estructura mecánica. En estos se utilizan tecnologías hidráulicas, neumáticas y en la actualidad se utilizan motores eléctricos, en particular los de corriente continua servocontrolados, también en algunos casos motores a pasos y otros actuadores electromecánicos sin escobillas. Existen también robots industriales de accionamiento directo que permiten eliminar problemas mecánicos inherentes al empleo de engranajes y otras transmisiones. En la actualidad se investiga sobre actuadores que disminuyan la inercia y suministren un par elevado, aumente la precisión, originen menos ruido magnético y sean ligeros y de consumo mínimo.

2.2.3. Sensores y sistemas de control

Dentro del sistema de control se descompone de manera jerárquica según su estructura. En el nivel inferior se realizan las funciones de servocontrol y supervisión de articulaciones. La mayor parte de los robots industriales utilizan servomecanismos convencionales con realimentación de posición y velocidad para generar señales de control en los actuadores de las articulaciones. Los parámetros del controlador son fijos aunque varíen las condiciones de trabajo y de la carga con el propio movimiento. Los problemas aumentan cuando se le aplica mayor velocidad. Debido a lo anterior, la mayoría de los robots industriales actuales utilizan una velocidad de operación pequeña.

El segundo nivel de control realiza la generación de trayectorias, captando la evolución del órgano terminal cuando se desplaza de una posición a otra. El generador de trayectorias debe proporcionar a los servomecanismos las referencias adecuadas para obtener la evolución del órgano terminal a partir de las especificaciones del movimiento deseado en el espacio de trabajo. Para obtener las referencias adecuadas para las articulaciones, es necesario resolver el modelo geométrico inverso el cual es no lineal. Los niveles superiores se ocupan de la comunicación con el usuario, interpretando programas y datos de los sensores.

Los primeros robots industriales eran programados por guiado manual, el cual almacenaba la secuencia de posiciones en memoria digital. La interacción con la tarea específica se limitaba a la apertura o cierre del órgano terminal, indicándolo a un equipo externo, o esperando una señal de sincronización. Las especificaciones que se tenían eran de *pick and place*, tales como la carga y descarga de máquinas con movimientos definidos y fijos. Es decir, se primaba la repetibilidad sobre la adaptación.

En la robótica industrial se ha integrado los avances de control por computador de telemanipuladores. En 1958 Shannon y Minsky propusieron un dispositivo, *sensor-controlled robot*, el cual consistía en un teleoperador conformado por varios sensores conectados a un computador que administraba la información necesaria para decidir

las acciones necesarias para alcanzar el objetivo especificado. Este dispositivo no llegó a realizarse debido a las investigaciones de otros, como Ernst que en su tesis doctoral, en el MIT, construyó un robot con sensores de tacto en la mano que podía ser programado para realizar tareas tales como localización, agarre, transporte y descarga. Esto mediante programas parecido al lenguaje ensamblador, incorporando ordenes relativas a los sensores de tacto.

En 1963 Roberts hizo el hincapie en el uso de percepción, los cuales tenían la posibilidad de procesar una imagen digitalizada capaz de procesarla y obtener una descripción matemática de objetos situados en la escena, expresando su orientación y posición mediante transformaciones homogéneas.

Wichman en 1967 presentó en Stanford un equipo con cámara de televisión conectada a un computador que en tiempo real identificaba objetos y su posición. Cabe mencionar que los movimientos se deben obtener mediante coordenadas cartesianas y no en desplazamientos angulares de las articulaciones. Pieper en 1968 obtuvo una solución al problema utilizando resultados teóricos de la cinemática. En los años setenta, se investiga en robots con sensores de visión, resolviendo en tiempo real problemas de manipulación con visión en color y ensamblado de estructuras de bloques.

A principios de los ochenta se trabaja en lenguajes de programación orientados a proporcionar inteligencia artificial para la generación automática de planes. Hoy en día se sigue utilizando la visión para el control de manipuladores en la industria con diferentes métodos para resolver la planificación de movimiento en tiempo real. todo lo anterior lo vemos en la bibliografía[3].

2.3. Robots móviles

En el estudio de los robots móviles se extiende su estudio ya que los robots dejan de estar fijos a uno de sus puntos de sus extremos y ahora se les debe dar una mayor inteligencia en la autonomía de sus movimientos. En la autonomía de estos robots utilizan dispositivos electromecánicos, tales como los denominados micro-mouse, para desarrollar funciones inteligentes tales como describir caminos en laberintos. En estos se tiene que cuentan con la percepción de objetos y la evasión de los mismos para seguir una trayectoria sobre cualquier terreno.

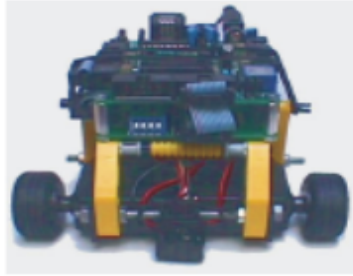


Figura 2.3: Ejemplo de robot móvil pequeño (microbot TRITTON).[4]

2.4. Definiciones básicas en un robot

Un Robot es una máquina inteligente que cuenta con una capacidad de relizar tareas que en ocasiones para el hombre son complicadas de realizar ya que son de un requerimiento de gran esfuerzo y además se consume un tiempo considerable en la realización de estas.

En nuestros tiempos ya es normal ver máquinas inteligentes que nos facilitan muchos trabajos de la vida cotidiana y ya es mas común ver a los humanoides que cuentan con una inteligencia escepcional y que casi pueden ser autónomos, pero debemos tener en cuenta que las máquinas no llegaran a reemplazar al hombre y dentro de este concepto el robot tiene que obedecer a ciertas normas y la principal de estas es el no dañar al hombre físicamente[13].

2.4.1. Conceptos de partes que conforman a un robot

Dentro de los conceptos básicos abarca hasta los que componen los robots manipuladores que son llamados comúnmente industriales ya que son utilizados mayormente en el ramo de la industria.

Mecanismos de eslabones articulados

Es un conjunto de sólidos (eslabones) articulados entre sí de tal manera que conduzcan al órgano terminal, colocado en el extremo de la estructura; a las posiciones y orientaciones necesarias para la realización de una tarea dada.

Articulaciones

Las articulaciones de un robot pueden ser de revolución o prismática, la primera se refiere a un movimiento rotacional generado por el movimiento de la flecha de un motor, y la articulación prismática implica la conversión de un movimiento de revolución a un movimiento de traslación.

Organo terminal

Dispositivo que se instala en el extremo del manipulador, el cual interactúa con los objetos de su entorno al efectuar una tarea. Consiste en un sistema mecánico capaz de prender y soltar objetos (pinza mecánica) o herramienta capaz de realizar alguna operación (antorcha o pistola de soldadura, etc.).

Actuadores

Son los dispositivos que producen la potencia mecánica que se requiere para mover los eslabones del manipulador. Pueden ser motores eléctricos, pistones y motores neumáticos e hidráulicos.

Transmisiones de potencia

Permiten la conducción del movimiento generado en los actuadores hasta cada eslabón del manipulador. Pueden ser del tipo de bandas dentadas, engranes, cadenas, etc.

Grados de libertad(GDL) de un robot

Corresponde al número mínimo de variables independientes que es necesario definir para determinar la situación en el espacio de todos los eslabones del manipulador. La regla de K. Kutzbach permite determinar el número de grados de libertad de un robot con mecanismos especiales en términos del número de eslabones y de articulaciones.

$$G = 6 * (e - 1) - 5 * a \quad (2.1)$$

Donde:

G=Grados de libertad.
e=Número de eslabones.
a=Número de articulaciones.

Coordenadas operacionales

Representadas por la posición en y la orientación del órgano terminal en el espacio. En la acción de control, permiten conocer la posición instantánea que el robot adquiere con su movimiento.

Espacio de configuraciones admisibles

Ó espacio de trabajo del robot, es la zona del espacio físico que puede ser alcanzada por un punto del órgano terminal.

2.4.2. Tipos de articulaciones

Los movimientos de articulaciones individuales asociados con las categorías de movimientos de brazo y cuerpo y movimientos de la muñeca se denominan, a veces por el término grado de libertad, y un robot industrial típico está dotado de cuatro a seis grados de libertad.

Los movimientos del robot se realizan por medio de articulaciones accionadas. Tres articulaciones suelen estar asociadas con la acción del brazo y del cuerpo, y dos o tres articulaciones se suelen emplear para accionar la muñeca.

Las articulaciones utilizadas en el diseño de robots industriales suelen implicar un movimiento relativo de las uniones contiguas, movimiento que es lineal o rotacional. Las articulaciones lineales implican un movimiento deslizante o de traslación de las uniones de conexión. Nos referimos a la articulación lineal como una articulación de tipo L (Lineal). El término de articulación prismática se utiliza, a veces, en la documentación sobre robótica en lugar de articulación lineal[13].

Hay, como mínimo, tres tipos de articulación giratoria que pueden distinguirse en los manipuladores de robots. Denominaremos al primero como una articulación tipo R (Rotacional). En la articulación de tipo R el eje de rotación es perpendicular a los ejes de las dos uniones. El segundo tipo de articulación giratoria implica un movimiento de torsión entre las uniones de entrada y salida. El eje de rotación de la articulación de torsión es paralelo a los ejes de ambas uniones. Llamaremos a ésta articulación de tipo T (Torsión). El tercer tipo de articulación giratoria es una articulación de revolución en la que la unión de entrada es paralela al eje de rotación y la de salida es perpendicular a dicho eje. Esencialmente, la unión de salida gira alrededor de la de entrada como si estuviera en órbita. Esta articulación es designada como tipo V (revolución).

Las articulaciones del brazo y del cuerpo están diseñadas para permitir al robot desplazar su efector final a una posición deseada dentro de los límites del tamaño del robot y de los movimientos de las articulaciones. Los diferentes tipos de articulaciones se pueden ver en la figura 2.4.

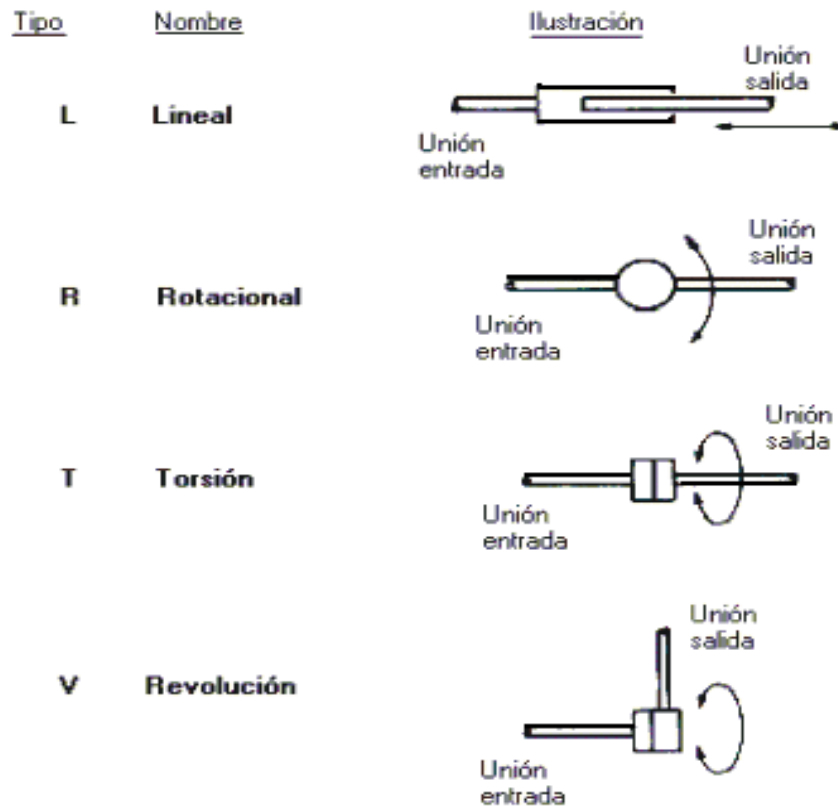


Figura 2.4: Diferentes tipos de articulaciones usadas en robots.[13]

2.5. Configuraciones de robots

La ciencia ficción se ha inspirado en el humanoide, para presentar sus narraciones ficticias y lo ha considerado como un robot, ya que la palabra proviene del checo e inventada por Karel capek, el cual era un dramaturgo y la utilizó en su obra, *Los robots Universales de Rossum*, publicada en 1917 y popularizada en 1921 como un melodrama; sin embargo muchos siglos antes el hombre ya pensaba en algunos aparatos y máquinas que tuvieran relación con los seres vivos y la función laboral.

2.5.1. Robot cartesiano

Los robots presentan una estructura articulada, es decir que se encuentra conformado por una serie de elementos ó eslabones que facilitan el movimiento, el cual puede ser de translación vertical, horizontal y transversal , los cuales se pueden combinar con movimiento de giro sobre el mismo eje[32].

En todo robot se debe tener en cuenta los grados de libertad definidos por la sigla **GDL** que significa que se debe estudiar a cada uno de los movimientos en una forma independiente, que en un espacio tridimensional puede realizar una articulación. Es por ello que el número de grados de libertad de los robots, viene dada por la suma de los grados de libertad de las articulaciones que lo componen.

- Tiene una estructura PPP.
- La especificación del punto se efectúa mediante coordenadas cartesianas (x,y,z) .
- La precisión es uniforme en todo el espacio de trabajo.
- Apto para seguir una trayectoria previamente especificada.
- Construcción rígida: la distribución de cargas no presenta problemas especiales.
- No resulta adecuada para puntos situados en espacios cerrados.

2.5.2. Robot cilíndrico

La estructura determinada para el robot cilíndrico facilita un desplazamiento vertical, transversal y giratorio sobre el eje que gira sobre su propia base. Se ha de reconocer también la forma de la estructura del robot *SCARA* que facilita movimientos y desplazamiento vertical, giratorio sobre su propio eje de la base, como el de la articulación, lo cual facilita cualquier tipo de ensamble de un producto, que se relacione con el mundo moderno. Se encuentra otros tipos de estructuras como la esférica ó polar, la angular ó antropomórfico y el cilíndrico.

- Tiene una estructura RPP y PRP.
- La posición del punto en el espacio se encuentra en las coordenadas cilíndricas (a,r,z) .
- Ofrecen ventajas cuando las tareas o máquinas servidas se encuentran radialmente al Robot.

2.5.3. Configuración esférico o polar

Este Robot son de los primeros robots y tiene las siguientes características.

- Tiene una estructura RRP.
- La posición del punto en el espacio se realiza mediante las coordenadas (a,b,r) .
- Configuración utilizada para los primeros Robots.

2.5.4. Configuración angular

A continuación veremos las principales características de este tipo de Robots.

- Tiene una estructura RRR.
- La posición del punto se ubica en el espacio en las coordenadas angulares (a,b,g).
- Soluciona en cierta forma, el acceso a espacios cerrado.
- Sin embargo, obliga a un esfuerzo suplementario en el sistema de control para el seguimiento de trayectorias rectilíneas.

2.5.5. Configuración ESCARA

- Tiene una estructura RRP o PRR.
- Especialmente para realizar tareas de ensamblado electrónico, y en general de manipulación vertical.
- La mayoría de los fabricantes incluye este tipo en su oferta.

2.5.6. Configuración paralela

- Elemento que se encuentra conectado a la base, por lo menos, a dos cadenas cinemáticas independientes.
- Inicialmente utilizada en los simuladores de vuelo.
- La carga se reparte entre los eslabones.
- La rigidez de los eslabones asegura con mayor precisión el punto deseado.
- Bajo coste relativo y montaje preciso. Lo anterior lo podemos ver en[14].

2.6. Sistemas de un robot

Un robot industrial se encuentra constituido por una serie de sistemas que en conjunto facilitan con gran precisión los movimientos y procesos que debe ejecutar, como es su estructura, los sistemas de accionamiento, los mecanismos de transmisión, los sistemas sensoriales y la unidad de control. Este tema y los demás subtemas podemos verlo en la bibliografía[32].

Los sistemas de un robot dan cada uno la parte fundamental del sistema y cada uno es importante para el trabajo que desarrolle este mismo en las mejores condiciones de realización.

2.6.1. La estructura

Este sistema esta conformado por soportes articulados los cuales permiten el giro o traslación de los ejes al ejecutar un proceso de movimiento en el plano de trabajo, pues cada articulación permite un solo movimiento. Su diseño es fundamental porque debe tener en cuenta que tipo de operadores mecánicos se deben incluir ó que dispositivos debe soportar para realizar las tareas en el robot.

2.6.2. Sistema de accionamiento

En principio los accionamientos de los robots fueron hidr hulicos y en algunos casos neum ticos pero en la actualidad se componen de controles de potencia electr nicos, encargados de controlar la energ a a los generadores en movimiento y motores los cuales pueden ser de corriente continua o corriente alterna.

2.6.3. Sistema de transmisi n

El Sistema de transmisi n por lo general consta de un sistema de reductores los cuales reducen altas velocidades de los ejes en los motores y aumentan el torque. En ocasiones constan de pi ones, eslabones, cadena de pi on, de husillo, los cuales se encargan de transferir y canalizar el giro de los motores hasta los elementos m viles del robot.

2.6.4. Sistema sensorial

El sistema sensorial lo compone todos aquellos sensores que se pueden acoplar internamente a las articulaciones   tambi n en la parte externa los cuales pueden ser sensores de posici n, de velocidad, de proximidad, de visi n; y se encargan de captar la situaci n en que se encuentra el robot, cualquier informaci n que se emita por uno de ellos es captada por la unidad de control y optimizar el trabajo del sistema general para el buen uso del robot.

2.6.5. Sistema de control

El control de un robot puede estar compuesto de diversas maneras como empleando temporizadores, coprocesadores especializados, contadores, unidades de entrada y salida, convertidores d/a   de a/d los cuales seran programados desde uno   varios microprocesadores.

2.6.6. Sistemas de comunicaci n

Son los dispositivos los cuales ayudan a el recibimiento y env o de datos para el funciainamiento de sistemas, que pueden ser desde una computadora hasta diferentes

dispositivos de radiocomunicación y procesamiento.

2.7. Cinemática de los robots

Cinemática es la parte de la física que estudia el movimiento de los cuerpos desde un punto de vista geométrico, sin considerar las causas que lo producen. La descripción del movimiento de un cuerpo se realiza de acuerdo con un sistema de referencia, este sistema de referencia puede ser fijo o móvil.

En estos terminos, se considera, además del problema puramente geométrico y estático, las variaciones en el tiempo de las posiciones y orientaciones; es decir, las velocidades. En los modelos geométricos y cinemáticos se involucra esencialmente el estudio de las relaciones existentes entre el espacio de las variables articulares y el espacio de trabajo, o espacio operacional, que suele ser un espacio cartesiano.

2.8. Tipos de trayectorias

Dentro de las trayectorias se pueden clasificar en tres tipos. La primera es trayectoria punto a punto en la cual se derivan dos clasificaciones las de movimiento eje a eje y las de movimiento simultaneo de ejes, las de trayectorias coordinadas o isócronas y las de trayectorias continuas.

Trayectorias punto a punto

Cada articulación evoluciona desde su posición inicial a la final sin considerar el estado o evolución de las demás articulaciones.

- Movimiento eje a eje: sólo se mueve un eje cada vez (aumento del tiempo de ciclo).
- Movimiento simultáneo de ejes: los ejes se mueven a la vez acabando el movimiento cuando acabe el eje que más tarde (altos requerimientos inútiles)

Este tipo de trayectorias se utiliza solo en robots muy simples o con unidad de control muy limitada.

Trayectorias coordinadas o isocronas

Los ejes se mueven simultáneamente ralentizando las articulaciones más rápidas de forma que todos los ejes acaben a la vez. Esto hace que el tiempo total se reduzca y sea el menor posible y se evita el exigirle a las articulaciones mayor velocidad y aceleración[15].

Trayectorias continuas

En este tipo de trayectorias, a la trayectoria del extremo se le conoce como cartesiana y puede ser de dos tipos: de línea recta y de arco de círculo. A continuación se muestran los tres tipos de trayectorias ya mencionadas en la figura 2.5.

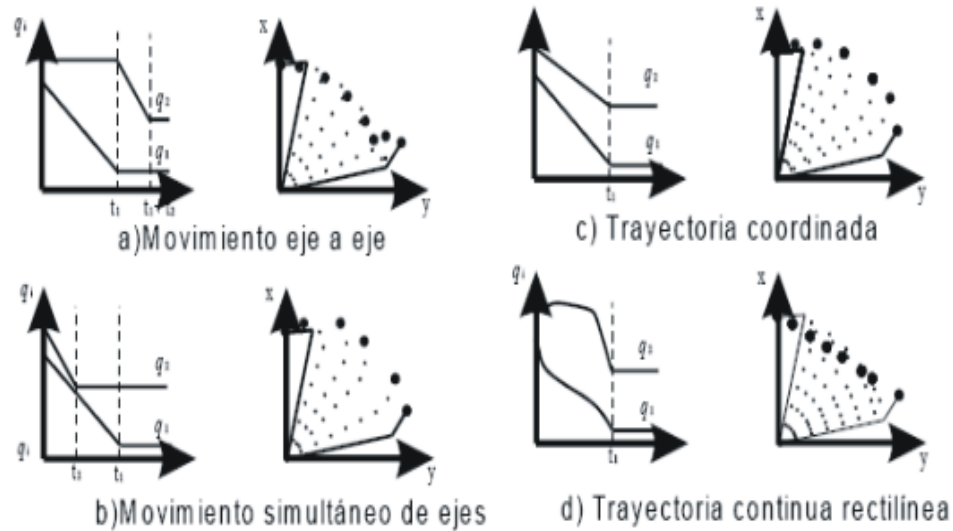


Figura 2.5: Ejemplos de trayectorias continuas.[19]

Interpolación de trayectorias

La Interpolación de trayectorias define la unión de una sucesión de puntos en el espacio articular por los que han de pasar las articulaciones del robot en un instante determinado. Debido a esto hay que considerar las restricciones que se tengan en el sistema. Para estas restricciones se utilizan funciones polinómicas que se ajustan a estas restricciones.

Existen diferentes interpoladores utilizados:

- Interpoladores lineales
- Interpoladores cúbicos (splines)
- Interpoladores a tramos
- Otros interpoladores

Los interpoladores lineales se comportan de la siguiente forma como se muestra en la figura 2.6.

$$q(t) = (q^i - q^{i-1}) \frac{t - t^{i-1}}{T} + q^{i-1} \quad t^{i-1} < t < t^i$$

$$T = t^i - t^{i-1}$$

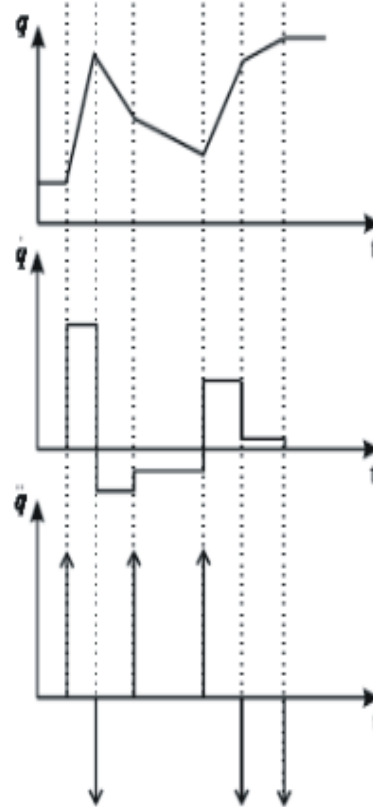
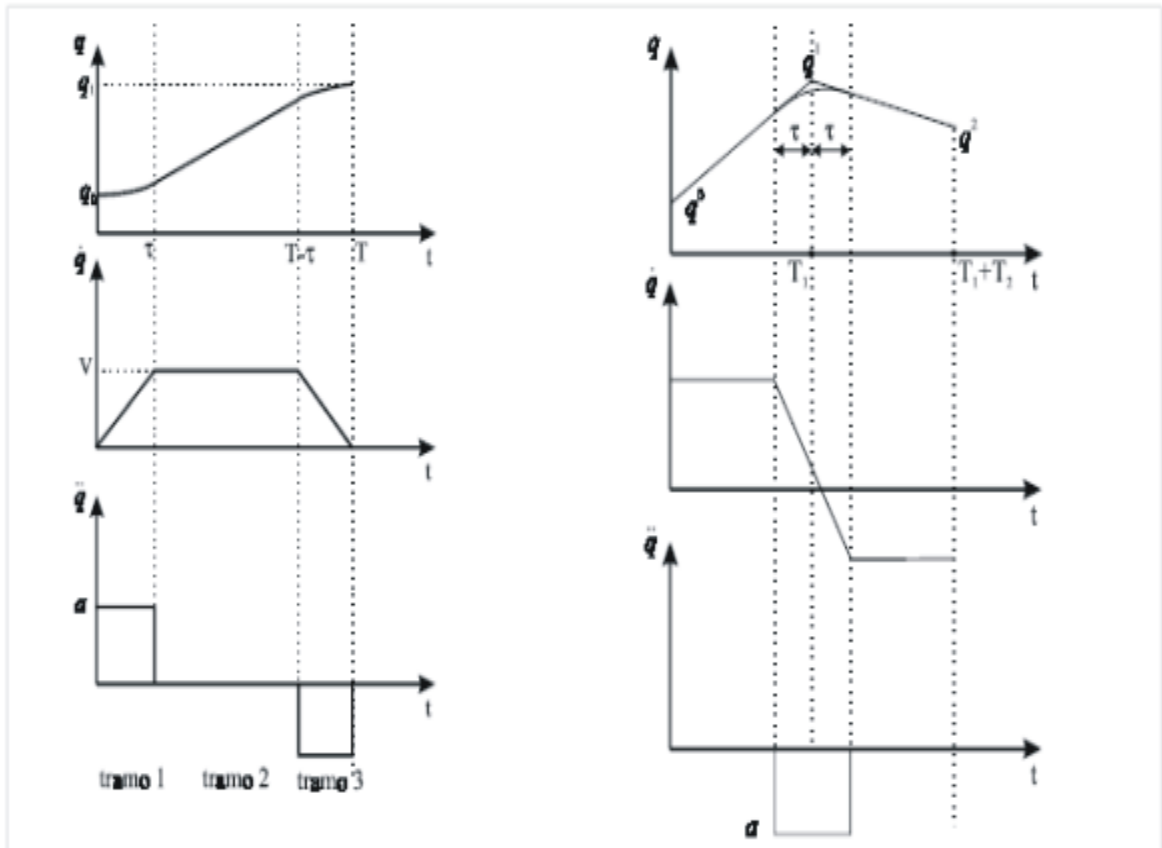


Figura 2.6: Interpoladores lineales.[19]

En los interpoladores cúbicos se utilizan polinomios de grado 3 para unir cada pareja de puntos, se tiene la posibilidad de implementar a cuatro condiciones de contorno al usar cuatro parámetros (dos de posición y dos de velocidad). Según sea la trayectoria será igual a la serie de polinomios cúbicos concatenados escogidos de forma que exista continuidad en posición y velocidad, denominados splines.

Los interpoladores a tramos se clasifican de dos formas los de interpoladores de tres tramos y los interpoladores con ajuste parabólico. Esto lo podemos encontrar en la bibliografía[15]



Interpolador con 3 tramos

Interpolador con ajuste parabólico

Figura 2.7: Interpoladores a tramos[19].

2.9. Conclusiones

En este capítulo se ayuda a entender el comportamiento que debe tener un robot, en el cual damos el concepto básico dependiendo de su comportamiento así como las diferentes etapas que tuvo para su creación.

Da una perspectiva de los diferentes tipos de robots, donde se abarcan los manipuladores, móviles, etc. De esta clasificación vemos a grandes rasgos las funcionalidades que se le dan a cada uno y una descripción de sus manejos, además de como realiza sus operaciones y así como de sus diferentes subsistemas.

Estos robots también en su género se clasifican según la funcionalidad y herramienta que utilizan para moverse, en el caso de los robots móviles. Es necesario conocer el concepto así como de su manera de operación de los diferentes robots para poder ver en que se puede innovar en cada uno de ellos.

Capítulo 3

Estudio de los robots: estructura y sistemas

3.1. Introducción

En nuestros tiempos podemos ver que los robots forman una parte importante dentro de las industrias, ya que su funcionamiento en ellos por lo general reduce el tiempo requerido en el proceso de producción y algunos son utilizados en tareas las cuales se harían complicadas para el hombre.

Dentro de los robots industriales se mencionan los robots tipo cartesiano los cuales generalmente trabajan en una área de trabajo situada en un plano X, Y y Z . Este tipo de robots también son utilizados en tareas de soldadura, pintura, ensamble y en la carga y descarga de objetos, (paletización), y hacen el empilamiento de cajas de un peso considerable ya que manejan un espacio de trabajo muy grande.

Como se mencionaba anteriormente ocupan un espacio de trabajo muy grande, posee tres movimientos lineales, es decir, tiene tres grados de libertad que corresponden a los movimientos lineales en las coordenadas X, Y, Z . Los movimientos de este robot los hace mediante interpolaciones lineales cuando se desplaza de un punto a otro. A la trayectoria realizada en línea recta se le conoce como interpolación lineal y a la trayectoria hecha de acuerdo con el tipo de movimientos que tienen sus articulaciones se le llama interpolación por articulación.

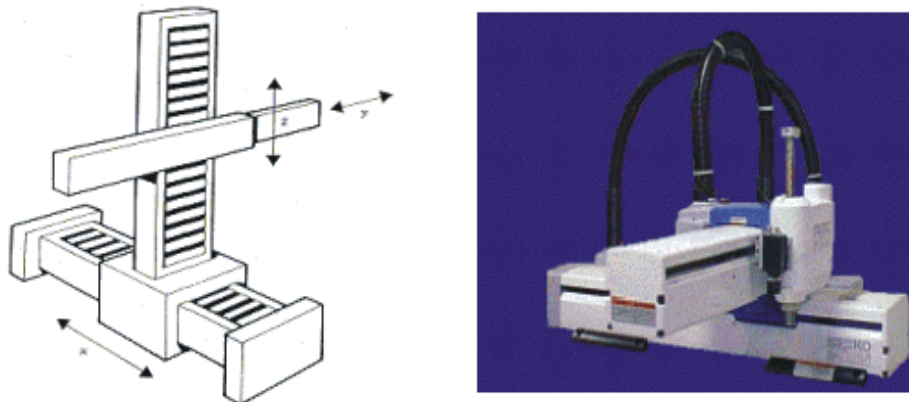


Figura 3.1: Ejemplos de robots cartesianos.[28]

La configuración cartesiana generalmente cuenta con tres articulaciones en **3D** o estructura *PPP*. Los valores que deben tomar las variables articulares corresponden a las coordenadas que toma el extremo del brazo. Esta configuración no es adecuada para puntos situados en espacios de trabajo cerrados.

3.2. Anatomía del robot

La anatomía del robot se refiere a la construcción física del cuerpo. La mayoría de los robots utilizados en las industrias se encuentran sobre una base fija al suelo. En el caso de los brazos robóticos el cuerpo está unido a la base y el conjunto del brazo está unido al cuerpo y al final del brazo está la muñeca que está compuesta por diversos componentes que le permiten orientarse en diferentes posiciones. Los movimientos hechos en el brazo obedecen a las articulaciones que generan giros o deslizamientos[16].

Unida a la muñeca del robot va una mano, que en términos técnicos se conoce como efector final, que es el que termina el trabajo de posicionarse en un punto específico y hacer el contacto con el objeto ubicado en ese punto.

3.2.1. Configuraciones básicas de robots

Estas configuraciones básicas existen dentro de la industria en diferentes tamaños y en la actualidad los comercializadores los venden en diferentes formas y configuraciones físicas. Las configuraciones básicas de robots industriales son las siguientes:

- Configuración polar.
- Configuración cilíndrica.

- Configuración cartesiana.
- Configuración SCARA.
- Configuración articulada.
- Configuración paralela.

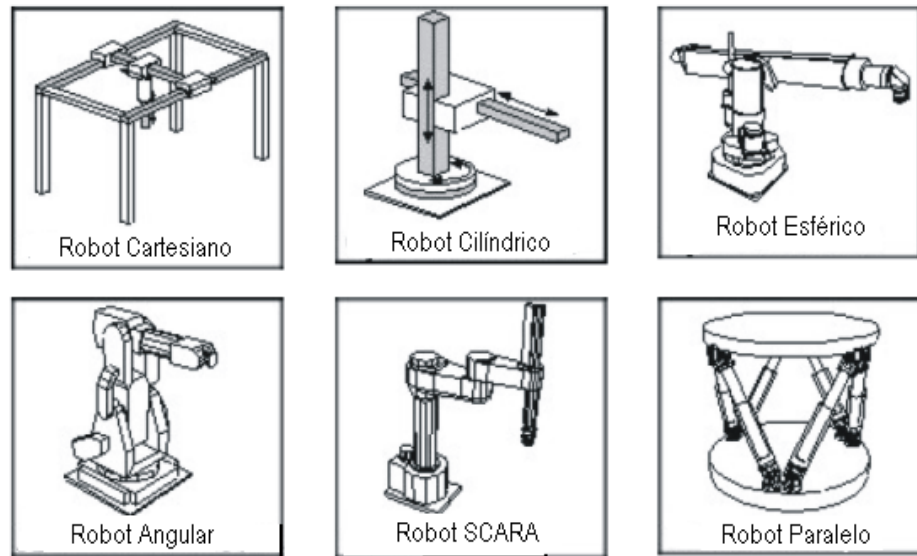


Figura 3.2: Ejemplos de configuraciones roboticas.[26]

El robot de coordenadas cartesianas tiene tres eslabones perpendiculares para construir los ejes x,y,z . Algunas veces los robots de este tipo de configuración también reciben el nombre de robots X,Y,Z o robots rectilíneos. Un ejemplo de esta configuración es el robot **IBM RS-1** (actualmente denominado modelo 7565). Debido a su apariencia y construcción se le denomina tipo caja. El robot tipo pórtico es otro nombre usado para los robots cartesianos que generalmente son grandes y tiene la apariencia a una grúa del tipo de pórtico.

3.2.2. Movimientos del robot

En los robots industriales los movimientos se dividen en tres categorías, los movimientos del cuerpo, el brazo y los movimientos de la muñeca.

Los movimientos del robot se realizan mediante las articulaciones accionadas. Tres articulaciones están asociadas con la acción del brazo y del cuerpo y dos o tres para el

accionamiento de la muñeca. Para la unión de estas articulaciones se emplean elementos rígidos llamados uniones y algunas veces llamados eslabones.

Las articulaciones en el diseño de robots industriales poseen uniones las cuales hacen movimientos lineales y rotacionales. Los movimientos lineales implican movimientos deslizantes o de traslación los cuales pueden obtenerse mediante un pistón, mecanismo telescópico y el movimiento relativo a lo largo de un carril. En este se menciona una articulación lineal L .

Existen tres tipos de articulaciones giratorias básicas, la primera es de tipo rotacional R . Esta articulación es perpendicular a los ejes de las dos uniones. El segundo tipo de articulación giratoria implica un movimiento de torsión entre las uniones ya que el eje de rotación de la articulación de torsión es paralelo al eje de ambas uniones y se le conoce como tipo T , y el tercer tipo de articulación giratoria es de tipo V , debido a que la articulación es de revolución en la que una de las uniones es paralela al eje de rotación y la otra es perpendicular a dicho eje.

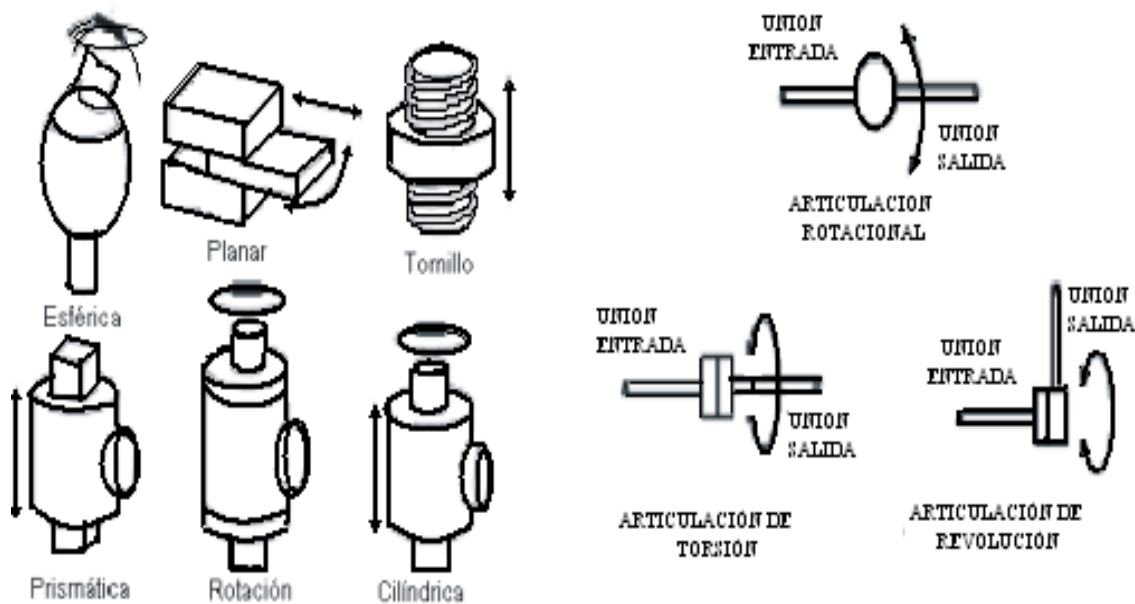


Figura 3.3: tipos de articulaciones.[27]

3.2.3. Volumen de trabajo

El volumen de trabajo es el espacio donde el robot podrá manipular el efector final y esto es debido a que el efector final define el espacio de trabajo. El volumen de trabajo viene determinado por las siguientes características físicas del robot:

- las configuraciones físicas del robot.
- Los tamaños de los componentes (cuerpo, brazo y muñeca).
- Los límites de los movimientos del robot.

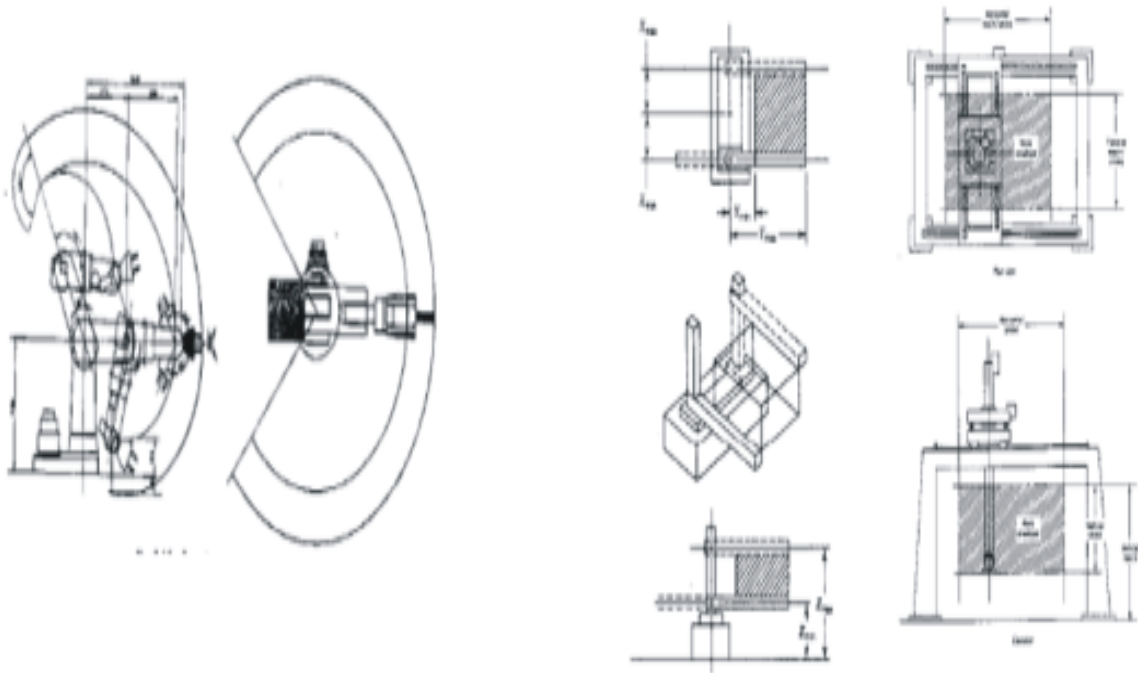


Figura 3.4: Espacio de trabajo de brazo articulado y de robot cartesiano.[14]

La estructura física del robot determina el volumen de trabajo del mismo, por ejemplo el volumen de trabajo de un robot de coordenadas polares, forma una esfera parcial, un robot de coordenadas cilíndricas tiene una envolvente de trabajo en forma de cilindro, un robot de coordenadas cartesianas tiene un espacio de trabajo de forma rectangular y un robot de brazo articulado tiene un volumen de trabajo aproximadamente en forma esférica. El volumen de trabajo está determinado por las dimensiones del brazo y los límites de los movimientos de sus articulaciones.

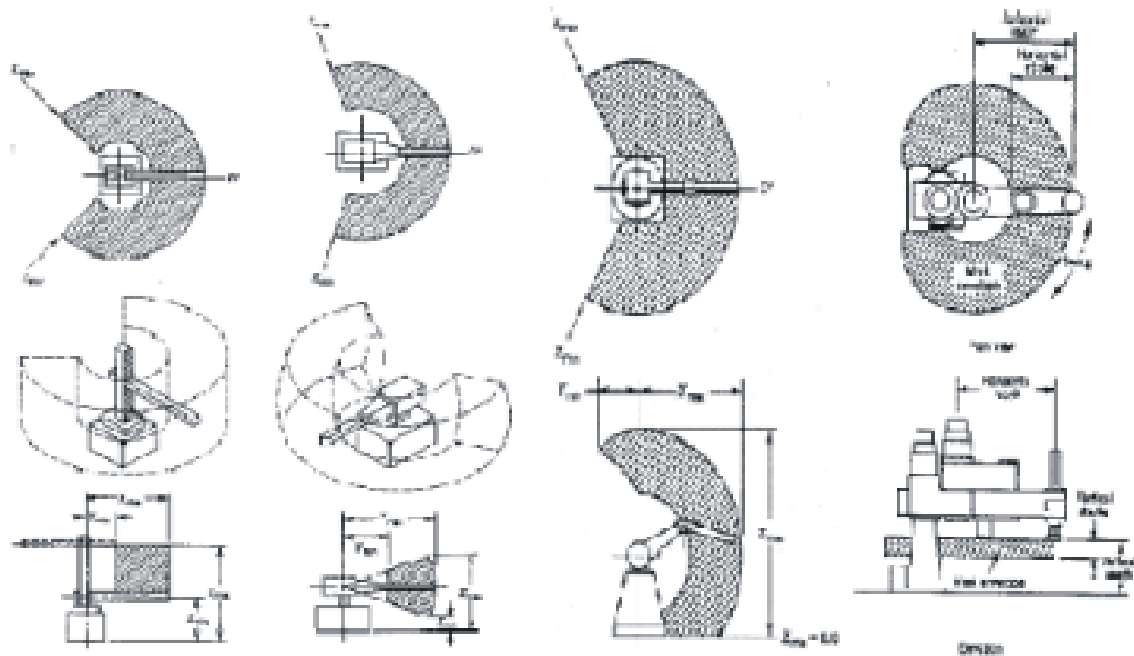


Figura 3.5: Espacio de trabajo de los robots cilíndrico, polar angular y Scara.[14]

3.3. Sistemas de impulsión de robot

El desplazamiento de el brazo, cuerpo y muñeca en un robot industrial obedece a un sistema de impulsión que acciona el robot. Este sistema determina la velocidad de los movimientos, la resistencia mecánica y el rendimiento mecánico. Además determina las aplicaciones que puede realizar el robot.

Los Robots industriales que se encuentran en el mercado están accionados por tres sistemas de impulsión que son:

- Impulsión hidráulica.
- Impulsión neumática.
- Impulsión eléctrica.

En los robots actuales se utilizan generalmente los sistemas de impulsión hidráulica y eléctrica. La impulsión hidráulica proporciona al robot mayor velocidad y resistencia mecánica. La desventaja de este sistemas es que se añaden mas necesidades en el espacio de trabajo y en ocasiones tienen fugas de aceite. Estos sistemas se implementan para articulaciones que desplazamientos rotacionales o lineales, se emplean ocasionalmente actuadores de paletas giratorias para el desplazamiento de rotación y un pistón hidráulico para obtener el desplazamiento lineal.

Los robots con sistemas de impulsión neumática suelen ser pequeños ya que tienen menos grados de libertad. Generalmente las aplicaciones de estos están limitadas a coger y situar un objeto en periodos de tiempo muy cortos. La potencia neumática se puede adaptar a la actuación de dispositivos de pistón para proporcionar un movimiento de traslación de articulaciones deslizantes y también puede ser utilizado para accionar actuadores giratorios para articulaciones de rotación.

En los sistemas de impulsión eléctrica ofrecen menos velocidad o potencia que los sistemas hidráulicos, pero la ventaja es que tiene mejor exactitud y repetibilidad en el robot y por consecuencia los robots de sistemas de impulsión eléctrica suelen ser más pequeños y su trabajo tiene mayor precisión. Este tipo de robots tienen accionamiento a partir de motores paso a paso o servomotores de corriente continua, que generalmente accionan a un sistema de engranajes y trenes impulsores.

3.3.1. Velocidad

Las velocidades más altas se pueden obtener por robots grandes con el brazo extendido a su máxima distancia del eje vertical del robot. En consecuencia la velocidad determina la rapidez en que el robot logra realizar un ciclo de trabajo. Esta suele desearse para hacer mínimo el ciclo de trabajo de un área determinada. Casi todos los robots cuentan con medios para ajustar la velocidad. Reducen el tiempo de ciclo de trabajo para aumentar la producción pero también dependen de otros factores como:

- La exactitud con la que se sitúa el efector final.
- El peso del objeto que se está manipulando.
- La distancia a recorrer.

Existe una relación inversa entre la velocidad y la exactitud de los robots ya que si se aumenta la exactitud se requiere más tiempo con el fin de reducir errores de localización en las diferentes articulaciones para conseguir la posición final deseada. El peso del objeto desplazado también influye en la velocidad. El peso de los objetos aumenta la inercia y la cantidad de movimiento y el robot tiene que actuar con lentitud para tratar de llegar al punto deseado. La distancia a recorrer influye en la velocidad ya que entre mayor sea la distancia se reduce el tiempo de ciclo y por consiguiente las distancias pequeñas por requerimiento de exactitud deben tener un ciclo mayor de tiempo.

3.3.2. Capacidad de carga

El tamaño, la configuración y el sistema de impulsión determinan la capacidad de carga del robot. La capacidad de transporte de carga se determina por la parte más débil del brazo del robot. En los casos de las configuraciones cilíndricas, polar y brazo articulado, en estos casos el brazo del robot se encuentra en su extensión máxima. Un

ejemplo muy sencillo es el de un humano que le es más fácil levantar un objeto pesado si tiene las manos pegadas al cuerpo que si las tiene extendidas. En un robot el peso de un objeto se debe considerar conjuntamente con el peso del efector final, pues ya que el efector final es una parte pequeña y en este no caería toda la capacidad de carga en el peso total.

3.4. Sistema de control

El funcionamiento de los robots debe tener un sistema el cual controle el sistema de impulsión y tener la regulación adecuada de los movimientos. El control en los robots es indispensable para los movimientos y posiciones correctas.

3.4.1. Cuatro tipos de controles

Los robots existentes en las industrias se pueden clasificar en cuatro categorías en el sistema de control que son:

- Robot de secuencia limitada.
- Robots de reproducción con control punto a punto.
- Robot de reproducción con control de recorrido continuo.
- Robots inteligentes.

Los robots de secuencia limitada representan el nivel más bajo y los robots inteligentes el más sofisticado. Los robots de secuencia limitada no utilizan un servo control para indicar las posiciones de las articulaciones; estos se controlan a través de interruptores de fin de carrera o topes mecánicos para establecer los puntos finales de desplazamiento. Este método solo ayuda al control del robot a que solo se pueda desplazar en límites de desplazamiento extremos.

La secuencia en la que se produce el ciclo de movimiento se define por un conmutador paso a paso, una placa de clavijas u otro dispositivo de secuencia. Este dispositivo señala a cada uno de los actuadores en que secuencia debe ir. Este tipo de robots no cuentan con dispositivos de realimentación para verificar si alcanzó la posición deseada. Para este tipo de robots se suele utilizar con mayor frecuencia el sistema de impulsión neumática aunque puede utilizarse cualquier otro.

Los robots de reproducción cuentan con una unidad de control más completa, en la que una serie de posiciones y movimientos son programados al robot y son almacenados en una unidad de memoria para posteriormente ser reproducidos en la aplicación requerida. Estos robots cuentan con un tipo de servocontrol para verificar si las posiciones

y movimientos realizados fueron los que se le programaron. Se pueden clasificar en dos categorías un robot punto a punto (*PTP*) y robot de trayectoria continua (*CP*). Los robots *PTP* realizan ciclos de movimiento que consisten en la localización de una serie de puntos deseados y acciones a fines. Al robot se le programa cada punto y se almacenan en la unidad de control y se reproduce posteriormente en la secuencia correcta. El robot controla la trayectoria correcta al pasar de un punto a otro. Los robots *CP* realizan ciclos de movimiento en la que se controla la trayectoria seguida por el robot. Esto se realiza mediante el desplazamiento del robot en una serie de puntos próximos. Los puntos individuales los define la unidad de control y no el programador. Un ejemplo es cuando se define el punto inicial y final por el programador y la unidad de control calcula la secuencia de puntos individuales que permitan seguir una trayectoria de línea recta. Algunos robots tiene la capacidad de seguir una línea curva suave definida por el programador que desplaza el brazo manualmente a través del ciclo de movimiento deseado. Para conseguir un control de trayectoria mas allá de los límites de extensión es necesario que la unidad de control pueda almacenar un gran número de posiciones y puntos individuales que definan la trayectoria curva compuesta, esto hoy en día se hace mediante un computador que actua como unidad de control.

Los robots inteligentes constituyen una clase cada vez más vista dentro de las industrias no solo por reproducir una serie de ciclos de movimiento programados sino también por interactuar con su entorno. Su controlador consiste en un computador o un controlador programable. Estos robots pueden modificar su ciclo programado según las condiciones de su volumen de trabajo, toman decisiones lógicas de acuerdo a los datos obtenidos de los sensores, se comunican con los humanos y con otros dispositivos basados en computadora durante la realización de un ciclo de trabajo. Estos robots se programan mediante un lenguaje de alto nivel para realizar actividades complejas y sofisticadas.

3.4.2. Velocidad de respuesta y estabilidad

Estos términos son importantes en la dinámica para sistemas de control dentro de un robot. La velocidad de respuesta refiere a la capacidad del robot de desplazarse a la posición siguiente en un período de tiempo corto. La estabilidad se define como la medida de oscilaciones que se producen en el robot(brazo) durante el movimiento desde una posición a la siguiente. Un robot con buena estabilidad presenta pocas o ninguna de las oscilaciones producidas por el movimiento. En el diseño de sistemas de control se requiere que se tenga una buena estabilidad y un tiempo de respuesta corto; sin embargo estos terminos son incompatibles pues la estabilidad puede controlarse, en cierta medida, incorporando sistemas amortiguadores en el diseño del robot. Un alto nivel de amortiguación aumenta la estabilidad del robot. El problema con la alta amortiguación es que reduce la velocidad de respuesta, debido a este problema se debe buscar una solución la cual reduzca las oscilaciones y aumente la velocidad de respuesta.

En los siguientes diagramas se muestra una relación en donde según el nivel de amortiguamiento se obtiene la velocidad de respuesta. En un alto nivel de amortiguamiento no se tiene oscilaciones pero obtenemos una velocidad de respuesta lenta. Cuando se reduce el nivel de amortiguamiento la velocidad de respuesta es rápida pero el sistema tiene oscilaciones.

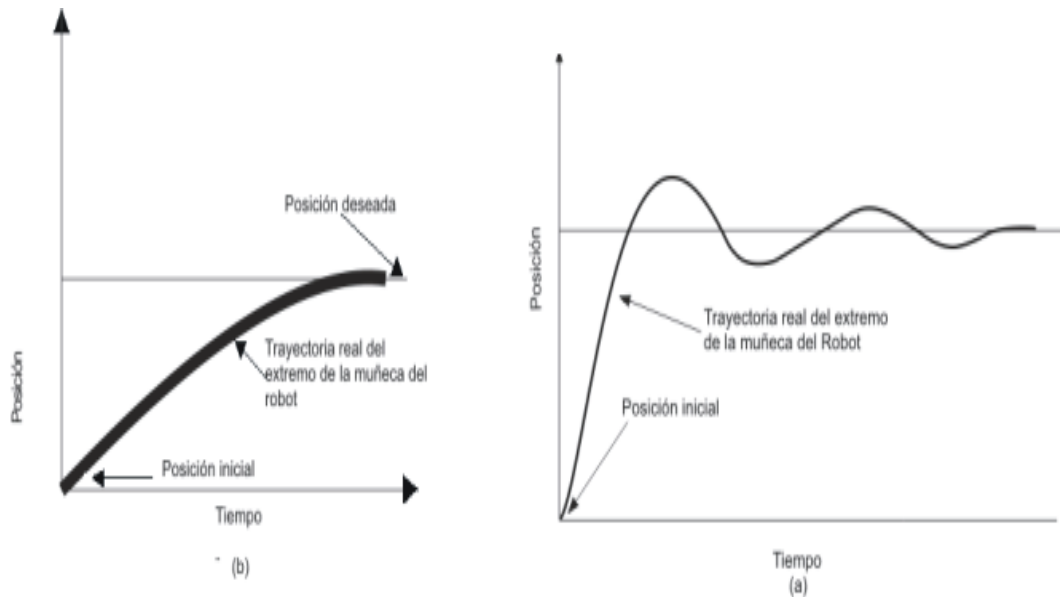


Figura 3.6: Velocidad y estabilidad en robótica: (b) baja amortiguación: respuesta rápida y (a) alta amortiguación: respuesta lenta.[16]

3.5. Precisión de movimiento

Este término lo definiremos a partir de tres características primordiales dentro del funcionamiento del robot.

1. Resolución espacial.
2. Exactitud.
3. Repetibilidad.

Estos terminos aplicarán al extremo de la muñeca de un brazo robótico y también aplicará donde las condiciones del robot sean las más desfavorables donde la precisión del robot es la peor. También se definiran en un robot PTP.

3.5.1. Resolución espacial

Este termino se define como el más pequeño incremento de movimiento en el que el robot puede medir su volumen de trabajo y esto depende de dos factores importantes. El primero de ellos es la resolución de control del sistema que viene determinada por el sistema de control de posición del robot y su sistema de medida de realimentación. Es la capacidad para dividir el margen total de movimiento para cada articulación en incrementos individuales que regula el controlador. Esto de dividir en incrementos individuales depende de la capacidad de almacenamiento en bits que tenga la memoria de control. El número de incrementos identificables separados (puntos direccionables) para un eje particular viene dado por:

$$2^n \tag{3.1}$$

Donde n = números de bits en la memoria de control.

Por ejemplo, un robot con 8 bits de almacenamiento puede dividir el margen en 256 posiciones distintas. La resolución de control se definiría como el margen de movimiento total dividido por el número de incrementos.

Otro ejemplo más claro es que si tenemos un robot con un grado de libertad, suponemos que tiene una articulación deslizante con un rango completo de 1m. La memoria de control del robot tiene una capacidad de almacenamiento de 12 bits. El problema consiste en determinar la resolución de control para este eje de movimiento.

El número de incrementos de control puede detrerminarse como el siguiente:

$$\text{Número de incrementos} = 2^{12} = 4096$$

El rango total de 1 m se divide en 4096 incrementos. Cada posición estará seperada por:

$$1 \text{ m}/4096 = 0.000244\text{m} \text{ ó } 0.224\text{mm}$$

Este ejemplo solo se refiere a una sola articulación. Un robot con varios grados de libertad tendría una resolución de control para cada articulación de movimiento. para obtener una resolución de control del robot completo, las resoluciones componentes para cada articulación deberán de sumarse en forma vectorial.

Las inexactitudes mecánicas en los comoponentes de las uniones y de las articulaciones que influye en la resolución espacial. Esto procede de la desviación elástica de los componentes estructurales: holgura de los engranes, tensión en bandas de poleas, fugas

de fluidos hidráulicos, etc. Estas suelen ser de mayor presencia en los robots de gran tamaño. Estas inexactitudes se ven afectadas por factores como la carga de manipulación, la velocidad de desplazamiento y las condiciones de mantenimiento del robot. La resolución espacial se ve degradada por inexactitudes mecánicas. La resolución espacial se ve mejorada con el incremento en la capacidad de memoria de control.

3.5.2. Exactitud

La exactitud es la capacidad de que tiene el robot para situar el efector final en el punto final deseado dentro del volumen de trabajo. También se puede definir en términos de resolución espacial, pues la capacidad de alcanzar un punto destino determinado depende de que tan próximos pueda el robot definir los incrementos de control para cada uno de sus movimientos de articulaciones. En el caso más desfavorable el punto deseado estaría en medio de dos incrementos de control adyacentes.

En el caso más desfavorable podemos definir a la exactitud, como la mitad de la resolución de control. Esta definición se muestra en la figura 3.7.

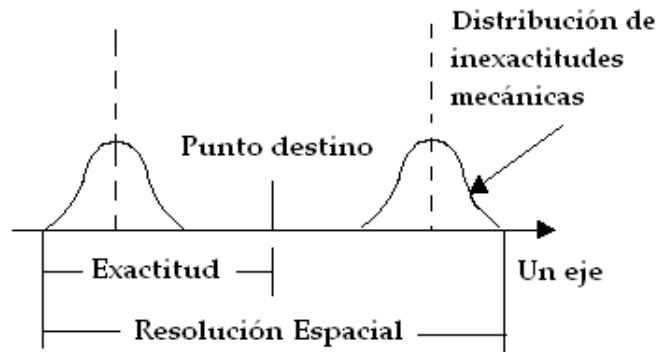


Figura 3.7: Exactitud y resolución espacial en donde las inexactitudes mecánicas se representan por una distribución estadística.[16]

En este caso donde el punto destino se encuentra entre dos puntos de control es cuando caemos en el caso más desfavorable. La exactitud en un robot se encuentra afectada por varios factores. Como primer lugar, la exactitud varía en el volumen de trabajo, donde es peor cuando el brazo se encuentra en el rango exterior del volumen de trabajo y se mejora cuando el brazo del robot se encuentra completamente extendido.

En segundo lugar la exactitud se mejora cuando el ciclo de trabajo esta restringido a un margen de trabajo limitado. Los errores mecánicos se reducen a medida de que el robot se restringe a un rango de movimientos. Cuando el robot alcanza un punto de referencia particular se le denomina exactitud local y cuando la exactitud se evalúa dentro del volumen de trabajo se le denomina exactitud global. Un tercer factor que afecta es la carga que se transporta, ya que entre mayor sea la carga se produce una mayor desviación entre las uniones mecánicas del robot.

3.5.3. Repetibilidad

Este termino se refiere a la capacidad que tiene el robot de situar el efecto final en un punto en particular el cual fue programado con anterioridad. La repetibilidad no solo se refiere a que se alcance el punto deseado sino además de que lo haga cuando se le ordene.

Esto se puede ver en la figura (3.8), donde el punto destino deseado se le asigna la letra T . En la programación se le ordena al robot que llegue al punto T , pero debido a las limitaciones que causa la exactitud, la posición programada se realiza en el punto P . La distancia entre los puntos T y P es la manifestación de la exactitud del robot. Posteriormente se da instrucción de regresar al punto programado P , pero retorna a otro punto el cual se le denomina punto de retorno R . El espacio entre los Puntos P y R es el resultado de las de las limitaciones de la repetibilidad del robot. El robot no siempre retornara al punto R , en ocasiones retornara a posiciones que se encuentran a los lados de los puntos R y P .

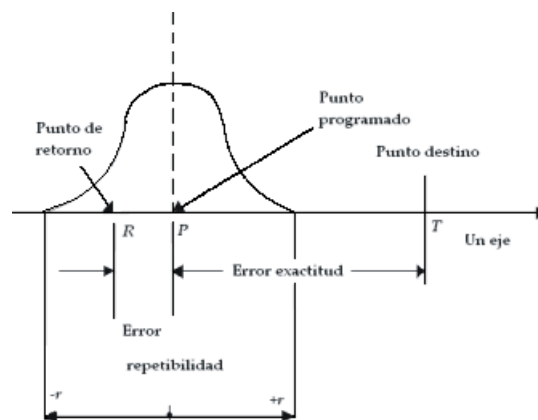


Figura 3.8: Repetibilidad y exactitud.[16]

3.5.4. Sensores robóticos

Los sensores empleados como periféricos en la robótica incluyen tipos simples como los interruptores de fin de carrera y tan sofisticados como los sistemas de visión. También los sensores se utilizan como componentes integrales del sistema de control en la realimentación de posiciones del robot. Dentro de los sensores se dividen en las siguientes categorías.

- Sensores táctiles. Este tipo de sensores responden a fuerzas de contacto con algún otro objeto. Estos dispositivos son capaces de medir la fuerza aplicada.
- Sensores de proximidad y alcance. Estos dispositivos indican la proximidad de un objeto antes de que se produzca el contacto. Cuando se logra determinar la distancia entre los dos objetos se les denomina sensores de alcance.
- Sensores diversos. En este grupo se mencionan algunos en los que se utilizan en sistemas más complicados, tales como los de temperatura, presión y otras variables.
- Sensores de visión. Estos son capaces de visualizar el volumen de trabajo e interpretar los datos que reciben de las imágenes. Estos robots son utilizados en tareas de inspección reconocimiento de piezas y otras.

Los sensores son una parte importante dentro de la célula de trabajo y en los sistemas de vigilancia actuales.

3.6. Sistemas de control y componentes

El robot es un sistema mecánico el cual debe ser capaz realizar una tarea de utilidad para el hombre. En los robots industriales la aplicación implican movimientos. Por lo tanto, la primer función del sistema de control posicionar y orientar con una velocidad y precisión específicas el efector final. En las siguientes secciones hablaremos de aspectos básicos de sistemas control y de la aplicación de este en las articulaciones del robot.

3.7. Actuadores y componentes de realimentación

En esta sección se analizará algunos dispositivos usados con frecuencia como componentes de los sistemas de control. Estos dispositivos se clasifican en:

- Sensores de posición
- Sensores de velocidad
- Actuadores

- Dispositivos de transmisión de energía

Los sensores de velocidad y posición son utilizados como dispositivos de realimentación, en tanto que los actuadores y los mecanismos de transmisión de energía son utilizados para realizar acciones de control. Los sensores de posición proporcionan los datos para determinar si las articulaciones se desplazaron para rectificar la posición lineal o rotacional, para obtener la posición deseada y la orientación requerida por el efector final. La velocidad es otra variable la cual requiere ser controlada. Algunos robots utilizan sistemas de realimentación para asegurar el control de la velocidad. Esto para los casos de desarrollo de control sofisticado con el objeto de depurar las prestaciones dinámicas durante la aceleración y desaceleración.

Los dispositivos de transmisión de energía y actuadores proporcionan la fuerza requerida para mover los eslabones.

Los actuadores contemplan dispositivos hidráulicos, eléctricos y neumáticos. La potencia producida por estos actuadores debe transmitirse desde el actuador a la articulación por medio del dispositivo de transmisión de energía, excepto cuando el actuador esta acoplado directamente a la articulación del robot. Algunos dispositivos de transmisión de energía son:

1. poleas
2. engranes
3. tornillos.

3.7.1. Sensores de posición

En la mayoría de los casos lo que se requiere controlar en la robótica es la posición. Existen diversos dispositivos para determinar la posición. Examinaremos los siguientes: potenciómetros, unidades de resolución y codificadores.

Potenciómetros

Los potenciómetros son dispositivos analógicos cuya tensión de salida es proporcional a la posición del cursor. En la figura 3.9 se muestra un potenciómetro común. En este se aplica una tensión en el elemento resistivo. La tensión entre el cursor y la masa es proporcional a la relación entre la resistencia en un lado del cursor y la resistencia total en el elemento resistente.

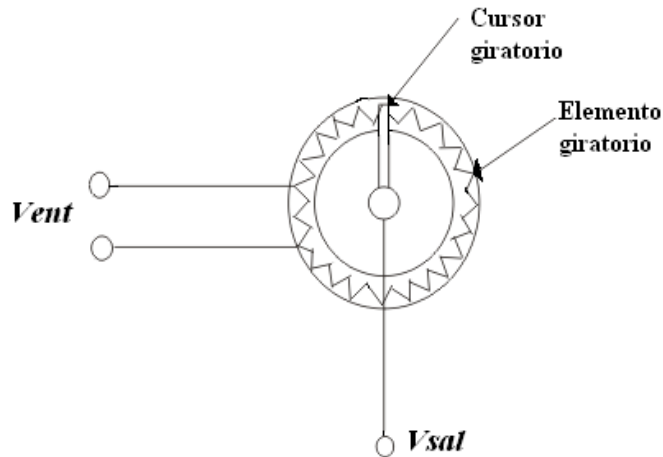


Figura 3.9: Potenciómetro.[16]

El potenciómetro actúa como una red de condición de tensión. Es decir, la energía que cruza el elemento resistente es dividida en dos partes por el cursor. La medición de esta energía proporciona el posicionamiento del cursor. La función del potenciómetro se puede representar mediante la siguiente función:

$$V_o(t) = K_p \theta(t) \quad (3.2)$$

Donde $V_o(t)$ es la tensión de salida, K_p es la constante de tensión del potenciómetro en volts por radianes (o volts por centímetros en el caso del potenciómetro lineal) y $\theta(t)$ es la situación del potenciómetro en radianes (o centímetros). Debido a que un potenciómetro requiere una tensión de excitación, con el objeto de calcular V_o se puede utilizar la fórmula:

$$V_o = V_{ex} \frac{\theta_{act}}{\theta_{tot}} \quad (3.3)$$

Donde V_{ex} es la tensión de excitación, θ_{tot} es el avance total disponible del cursor y θ_{act} es la posición actual del cursor.

Unidades de resolución

Es otro tipo de dispositivos analógicos cuya salida es proporcional al ángulo de un elemento de rotación con respecto a un elemento fijo. En la forma más simple, cuenta con un devanado sencillo en su rotor y un par de devanados en su estator. Los devanados

del estator tienen una separación de 90° . Si la excitación del rotor esta dada por una señal de tipo $A \sin(Wt)$, la tensión a través de los pares de terminales del estator será:

$$V_{s1}t = A \sin(Wt) \sin \theta \quad (3.4)$$

$$V_{s2}t = A \sin(Wt) \sin \theta \quad (3.5)$$

Donde θ es el ángulo de rotor con respecto al estator. Esta señal se puede analizar directamente o se puede convertir en forma digital empleando un dispositivo conocido como *resolver-digital*. Puesto que la unidad de resolución, es básicamente un transformador de rotación, es importante recordar que una señal de *c.a.* se debe utilizar para proporcionar la excitación. Si se utiliza una señal de *c.c.* no habría señal de salida.

3.7.2. Codificadores

Debido a que la mayoría de los sistemas se controlan mediante computadoras y otros dispositivos similares, se incrementa el uso de codificadores de posición digitales se requiere un estudio preciso de su funcionamiento. Los codificadores se suministran en dos tipos básicos: incremental y absoluto. Esto referencia a la manera en que se suministran los datos al codificador. Existen diferentes codificadores, pero solo nos enfocaremos a los que son utilizados con mayor frecuencia en la robótica.

Un codificador incremental consta de un disco de cristal marcado con bandas transparentes y opacas alternadas, radialmente alineadas. Un fototransmisor se encuentra situado en el lado del disco y un fotoreceptor en el otro. A medida de que gira el disco, el haz de luz emitido, por el fototransmisor, se corta en forma alternativa. La salida desde el fotoreceptor es un tren de impulso que es proporcional a la velocidad del disco. Contiene, en un codificado común, dos tipos de juegos de fototransmisores y receptores alineados con un defasamiento de 90° . El ajuste de fase produce información sobre el sentido de giro, es decir, si la señal A se adelanta en fase a la señal B en 90° , el disco codificador girará un sentido. Si B se adelanta en fase a A , entonces el disco codificador girará en sentido contrario. Mediante el recuento de los impulsos y sumando o restando según el signo, es posible utilizar el codificador para proporcionar información sobre la situación con respecto a una posición inicial conocida.

En algunos casos es desable conocer la posición de un objeto enterminos absolutos; es decir, no con respecto a una posición de arranque. Para hacerlo se podría utilizar un codificador absoluto. Los codificadores absolutos emplean la misma construcción básica que los incrementales, pero en estos existen más pistas de bandas, con su número correspondiente de receptores y transmisores. Las bandas proporcionan un número binario proporcional a algún ángulo de eje. La primera pista debera tener dos bandas, la segunda cuatro, la tercera ocho y así sucesivamente. De esta manera el ángulo se puede

leer de forma directa. la resolución de un codificador depende del número de pistas. Esta resolución está dada por:

$$\text{resolucin} = 2^n \quad (3.6)$$

Donde n es el número de pistas del disco.

3.7.3. Sensores de velocidad

Uno de los sensores utilizados con más frecuencia en la realimentación de la información de la velocidad es el tacómetro *c.c.* que proporciona una tensión de salida proporcional a la velocidad angular del inducido. Este se puede describir mediante la siguiente relación:

$$V_o(t) = K_t(t)w \quad (3.7)$$

Donde $V_o(t)$ es la tensión de salida del tacómetro en voltios, $K_t(t)$ es la constante del tacómetro, con frecuencia en V/rad/seg., y w es la velocidad angular en radianes por segundo.

Los tacómetros se suelen utilizar para proporcionar al controlador la información sobre la velocidad. Se puede utilizar para controlar la velocidad de un dispositivo o, en muchos casos, para incrementar el valor de K_d en su sistema por medio del perfeccionamiento de la estabilidad del sistema y su respuesta a las magnitudes perturbadoras.

Los tacómetros de corriente continua proporcionan una salida de tensión proporcional a la velocidad rotacional del inducido. Existe un equivalente digital del tacómetro de *c.c.* que proporciona una salida de tren de impulsos con una frecuencia la cual es proporcional a la velocidad angular.

3.7.4. Actuadores

Estos dispositivos son los que accionan a las articulaciones. Los actuadores suelen obtener su energía a partir de tres fuentes básicas: aire comprimido, fluido por presión o electricidad. Estos son neumáticos, hidráulicos y eléctricos, respectivamente. En Los próximos subcapítulos estudiaremos cada uno de ellos.

Actuadores neumáticos e hidráulicos

Este tipo de actuadores obtienen energía a partir del movimiento de fluidos. Los primeros, que son los actuadores neumáticos, el fluido que utilizan es el aire comprimido y los actuadores hidráulico utilizan, generalmente, aceite a presión. Comúnmente el funcionamiento de estos dos tipos de actuadores es similar, excepto en su capacidad para contener la presión del fluido. Pues los actuadores neumáticos operan a unas cien libras por pulgada cuadrada y los sistemas hidráulicos de 1000 a 3000 libras por pulgada cuadrada.

En los actuadores neumáticos se comprime el aire que abastecido por un compresor, el cual circula a través de mangueras. Los robots pequeños están diseñados para funcionar con actuadores neumáticos y esto los limita acciones tales como tomar y situar en un lado ciertos elementos. La exactitud puede aumentarse mediante paros mecánicos. Dentro de estos se clasifican cilindros y pistones neumáticos:

1. **Cilindro de doble vástago.** Posee vástago en ambos extremos del compartimiento.
2. **Cilindro Tándem.** Son dos cilindros acoplados mecánicamente, de modo que la fuerza resultante es la suma de la fuerza de cada cilindro.
3. **Cilindro multiposicional.** También son dos cilindros acoplados mecánicamente con longitudes distintas que arrojan a obtener cuatro posiciones distintas con dos señales de control.
4. **Cilindro de impacto.** Es un cilindro con dos cámaras de aire las cuales una de ellas contiene presión que se libera rápidamente hacia la otra cámara que contiene el émbolo. Esta velocidad del vástago se convierte en un fuerte impacto.
5. **Cilindro de giro.** Poseen un cople mecánico, que convierten el desplazamiento lineal de un vástago en giro sobre una pieza circular externa. Estos pueden ser de efecto simple o doble. El de efecto simple, son los que para su regreso, luego de eliminar la presión de aire, utilizan resortes que empujan al eje para regresar a su posición de reposo. Los de efecto doble son los que utilizan el aire a presión para las dos funciones la de ida y la de regreso.

La figura 3.10 muestra un dibujo del dispositivo más sencillo de potencia de fluido. Este dispositivo podría accionar una articulación lineal por medio de un pistón móvil. Este ejemplo recibe el nombre de cilindro de extremo único. Si el émbolo del pistón sale del cilindro en un solo extremo. Otro tipo de cilindros son los de doble extremo y los de cilindro sin émbolo.

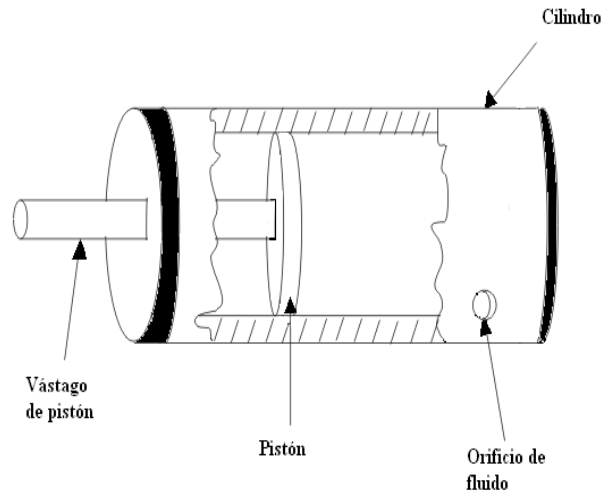


Figura 3.10: Cilindro y pistón.[16]

Existen dos relaciones importantes en este estudio: la velocidad del actuador con respecto a la potencia de entrada y la fuerza del actuador con respecto a la potencia de entrada. Para el actuador tipo cilindro la relación está dada por:

$$V(t) = \frac{f(t)}{A} \quad (3.8)$$

$$F(t) = P(t)A \quad (3.9)$$

Donde $V(t)$ es la velocidad del pistón, $f(t)$ es el caudal del fluido (volumétrico), $F(t)$ es la fuerza, $P(t)$ es la presión del fluido y A es el área del pistón. Debido a los requerimientos de un robot para llevar una carga de compensación a una velocidad dada, se puede utilizar las relaciones descritas para elegir el actuador apropiado.

Motores eléctricos

En los diseños de robots los motores eléctricos han tomado gran importancia. Estos motores proporcionan una gran controlabilidad con un requerimiento mínimo de mantenimiento. Existen diversos motores eléctricos utilizados en los robots. Los más usados son los servomotores de *c.c.*, los servomotores de *c.a.* y los motores de fase.

Los principales componentes del servomotor de *c.c.* son el rotor y el estator. El rotor incluye el inducido y el conjunto del conmutador y el estator está compuesto por los conjuntos de escobillas y los imanes permanentes. Cuando la corriente circula a través

de los devanados del inducido, establece un campo magnético opuesto al campo que producen los imanes. Esto proporciona un par motor en el rotor. Cuando gira el rotor, la escobilla y los conjuntos del conmutador desvían la corriente del inducido, por lo que el campo permanece opuesto a alguno de los campos establecidos por los imanes. En esta posición el par de torsión proporcionado por el rotor es constante en toda rotación. Puesto que la intensidad del campo del rotor es una función de la corriente que circula por él, el par motor, para un servomotor se puede expresar como:

$$T_m(t) = K_m I_a(t) \quad (3.10)$$

Donde T_m es el par motor, I_a es el flujo de la corriente que pasa a lo largo del inducido y K_m es la constante del par de torsión.

Otro efecto asociado a un servomotor de *c.c.* es la fuerza contraelectromotriz. Un motor de *c.c.* es similar a un generador *c.c.* o a un tacómetro. El giro del inducido en la presencia de un campo magnético, proporciona una tensión a través de los terminales del inducido. Esta tensión es proporcional a la velocidad angular del motor:

$$e_b(t) = K_b w(t) \quad (3.11)$$

Donde e_b es la fuerza electromotriz, k_b es la constante de tensión y w es la velocidad angular. La función de la fuerza contraelectromotriz es actuar como amortiguación viscosa para el motor, de manera que la velocidad es proporcional al amortiguamiento. Si se suministra una tensión a través de los terminales del motor de V_{in} y la resistencia del inducido es R_a , por lo tanto la corriente a través del inducido es V_{in}/R_a . Esta corriente proporciona un par motor en el rotor que hace girar al motor. Cuando el inducido gira, la fuerza generada por la fuerza contraelectromotriz igual a $K_b w(t)$ o $e_b(t)$. Esta tensión se debe restar de V_{in} con el objeto de calcular la corriente del inducido. La corriente relativa del inducido, por lo tanto, es:

$$I_a(t) = \frac{V_{in}(t) - e_b(t)}{R_a} \quad (3.12)$$

De manera que se incrementa la velocidad en el motor, la tensión contraelectromotriz aumenta, disminuye la corriente disponible para el inducido. La disminución de la corriente reduce el par de torsión generado por el rotor. Cuando el par motor se reduce, la aceleración se reduce también. Por lo tanto el punto en que $e_b = V_{in}$ el motor mantiene una velocidad de régimen permanente. El siguiente diagrama a bloques ilustra los efectos de la constante del par de torsión y la fuerza contraelectromotriz en el modelo de un motor. Se ha de tener en cuenta que este modo simplificado no contiene los efectos tales como rozamientos o autoinducciones de los devanados de los inducidos.

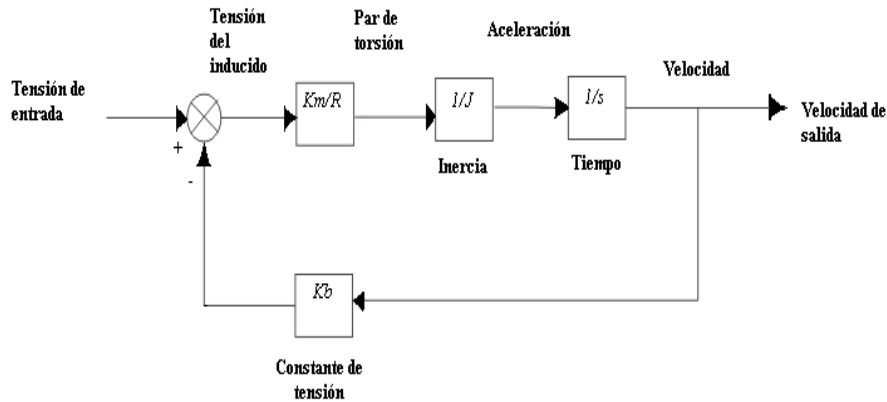


Figura 3.11: Figura diagrama a bloques de un motor de c.c.[16]

Motores a pasos

Estos motores, llamados también motores de velocidad gradual son un tipo de actuador único en su género y se utilizan en periféricos de computadora. Estos arrojan a una señal de salida en forma de incrementos discretos de movimiento angular. Estos son accionados mediante una serie de pulsos eléctricos discretos. para cada pulso hay una rotación de paso único en el eje del motor. Estos en la robótica son utilizados en tareas relativamente ligeras. Emplean sistemas de bñcle abierto.

Como se observa en la figura 3.12, se tiene un estator compuesto por cuatro polos y el rotor es un imán permanente de dos polos. Si los polos electromagnéticos se activan de tal manera que el polo 3 sea el norte y el polo 1 el sur, el rotor estará alineado como se ilustra. Si el estator se excita de tal manera que el polo 4 sea norte y el 2, dará un giro de 90° . El giro de estos motores se obtiene mediante la ayuda de dispositivos electrónicos que podemos obtener un giro continuo.

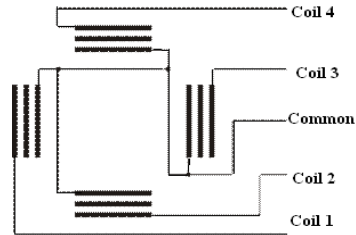


Figura 3.12: Esquema de un motor a pasos.[23]

El número de polos en el estator del motor determina la resolución. La resolución de un motor paso a paso y su ángulo de paso viene dada por:

$$n = A/360^\circ \quad (3.13)$$

Donde n es la resolución y A es el ángulo de paso.

A diferencia de un motor de *c.c.*, la relación entre la velocidad y el par de un motor a pasos no es necesariamente una línea recta. Dada su construcción el par es también una función del ángulo entre los polos del estator y del rotor. El par motor es más grande cuando los polos están alineados. Este par máximo se le conoce par de retención del motor. Es posible aumentar la resolución del motor a pasos utilizando una técnica conocida como semiescalamiento o microescalamiento. Aplicando corriente a más de un juego de devanados inductores es posible hacer que el rotor busque una posición media, pero reduciría el par de retención.

El control de un motor a pasos depende de la capacidad de el sistema eléctrico en su capacidad de conmutar los devanados en el momento adecuado. Si los devanados se conmutan de manera rápida es posible que el motor no capte las señales de mando y funcionará de manera errónea. La relación de velocidad par motor se degrada en algunas frecuencias de funcionamiento y se debe evitar que los motores funcionen a otras frecuencias.



Figura 3.13: Motor a pasos de 4 hilos[23]

3.7.5. Servomotores de c.a y otros tipos

Estos motores tiene la ventaja de ser más económicos en la fabricación que los de corriente continua, no tiene escobillas y poseen una alta potencia de salida. Con el adecuado conjunto electrónico pueden actuar como un motor de corriente continua.

Otro tipo es el de motor de *c.c.* sin escobillas. Esta construido como un motor de adentro - afuera, lo que quiere decir que tiene un rotor de imán permanente y un estator electromagnético, y la conmutación lo hace por medio de medios electrónicos con un codificador para informar al sistema electrónico de las posiciones relativas del estator y del rotor. También existen motores eléctricos y lineales. Su construcción es similar a un servomotor de corriente continua que haya sido cortado, abierto y aplanado.

3.8. Sistemas de transmisión de potencia

En muchas ocasiones es frecuente que el actuador no cumpla con las característica de velocidad-fuerza que requiere la articulación. Por tal motivo se utilizan sistemas de transmisión de potencia. Estas cumplen con dos funciones: transmitir potencia a una distancia y actuar como un transformador de potencia. Existen varias formas de suministrar esta potencia en las que se encuentran los engranes, poleas, cadenas, ejes de transmisión y tornillos.

3.8.1. Engranés

El empleo de engranes es muy frecuente dentro del diseño de robots. Esta forma de transmisión de potencia puede realizarse entre ejes paralelos, ejes de intersección o ejes sesgados. Los tipos más simples son para la transmisión entre ejes paralelos y se conocen como engranajes rectos.

En la figura 3.14 se muestra un tren de engranes rectos con dos engranes simples. El engrane impulsor, el más pequeño en este caso, se le conoce como el piñón y el otro engrane como como el engrane conducido. Por ejemplo, si el piñón es de $1/4$ del tamaño del engrane conducido girará solamente un cuarto de revolución. Este tren de engranes se conoce como reductor de velocidad. El par motor aplicado por el piñón se multiplica por 4 en el eje del engranaje. Puesto a que la velocidad se reduce en $1/4$ y el par motor se multiplica por 4, la potencia de salida del tren de engranes sigue siendo la misma.

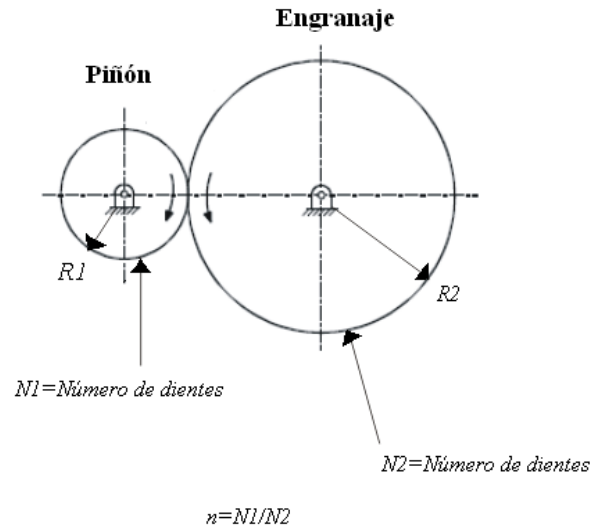


Figura 3.14: Tren de engranajes rectos[16]

El número de dientes en un engrane es proporcional a su diámetro. Si suponemos que el número de dientes en un piñón es N_1 y los dientes en el engrane conducido es N_2 , la relación de engranes viene dada por:

$$n = \frac{N_1}{N_2} \quad (3.14)$$

y la velocidad de salida con respecto a la entrada es:

$$W_o = nW_m \quad (3.15)$$

En donde W_o es la velocidad de salida y W_m es la velocidad de entrada. El par motor de salida es:

$$T_o = \frac{T_m}{n} \quad (3.16)$$

3.8.2. Tornillos de potencia

Los tornillos de potencia se suelen utilizar para convertir un movimiento giratorio en un movimiento lineal. El parámetro de un tornillo, que es análogo a una relación de engranes, es el paso del tornillo, p , que se suele denominar también intervalo[16]. El paso define la distancia que el tornillo recorre en una rotación única. La conversión de rotación angular del tornillo en un movimiento lineal viene dada por:

$$v(t) = pw(t) \quad (3.17)$$

en donde $v(t)$ es la velocidad lineal en pulgadas por minuto, $w(t)$ es la velocidad angular en revoluciones por minuto y p es el paso del tornillo expresado en pulgadas por rotación. Por lo general, el tornillo está girando y una tuerca se desplaza a lo largo del tornillo. La conversión del par de torsión T aplicado al tornillo en una fuerza F sobre la tuerca se obtiene por la siguiente ecuación

$$F = \frac{2T}{d_m} \frac{\pi d_m - \mu p \sec \beta}{d_m p + \mu \pi d_m \sec \beta} \quad (3.18)$$

en donde μ es el coeficiente de rozamiento entre las roscas del tornillo, β es el ángulo de rosca para una rosca Acme or Undefined y d_m el diámetro medio del tornillo.

Esta ecuación se aplica para roscas Acme or Undefined, en las que existe un ángulo de rosca β . Para roscas cuadradas, el valor de β es cero y los terminos son iguales a 1 en la ecuación.

3.8.3. Otros sistemas de transmisión

Algunos sistemas de transmisión de potencia incluyen sistemas de poleas, transmisiones de cadena y transmisiones armónicas. Los sistemas de poleas se suelen utilizar para transmitir potencia desde actuadores situados en la base del robot. En algunos casos, el cable o la cuerda puede estar constituido por fibras de acero o por materiales sintéticos, tales como nylon. Las articulaciones rotacionales pueden conectarse a una polea, que es impulsada por un cable unido a un actuador giratorio. De manera análoga puede utilizarse cables para activar las articulaciones lineales. En una u otra aplicación, el cable sufre una flexión continua durante la operación. Si el cable está mal dimensionado para la aplicación, está puede dar lugar a un estiramiento, incluso a romperse. Si el cable se estira, se produce una degradación de la exactitud del robot.

Las transmisiones de cadena operan con una relación constante. Debido a la interacción positiva entre la cadena y las ruedas dentadas, no se produce deslizamiento alguno. El paso de una cadena es la distancia entre centros de rodillos contiguos. La rueda dentada impulsora y la rueda conducida tienen cada una dientes diseñados para la adaptación del tamaño y del paso de la cadena. La transmisión de la velocidad de rotación y de la de potencia entre las ruedas dentadas sigue relaciones similares a las desarrolladas para engranes, que examinamos con anterioridad. La lubricación es importante en el mantenimiento de las transmisiones de cadena, ya que una cadena bien lubricada puede tener la duración de 100 veces más que la de una mal lubricada.

Las transmisiones armónicas son un producto patentado de USM, Inc. Pueden utilizarse elevadores o reductores de velocidad. Los ejes de entrada y de salida pueden estar dispuestos a lo largo del eje geométrico, de tal modo que una transmisión armónica puede montarse en la parte frontal de un motor con el eje de salida saliendo por el mismo extremo. Las transmisiones pueden proporcionar cualquier relación de reducción, desde 1:1: infinito:1, aunque suelen emplear el rango de 100:1. Estas transmisiones

requieren poco mantenimiento y pueden funcionar sin ningún desgaste en su vida útil. Sin embargo son menos eficientes que los trenes de engranes bien diseñados. La mayoría de este capítulo lo podemos ver en la bibliografía[16].

3.9. Conclusiones

En este capítulo se analizó de una manera más profunda cada una de las partes que conforman los sistemas, en este caso abarcó gran parte de los actuadores y sensores, donde se da a conocer los diferentes tipos sus usos y el desgaste así como la duración que tiene cada uno de ellos en cada una de sus aplicaciones. En cuanto a los sensores se analiza principalmente a los sensores de velocidad, resolución y codificadores ópticos que nos ayudan a tener un sistema de realimentación en cuanto al desplazamiento lineal o circular de las articulaciones de los robots. Además se da una introducción en los diferentes aspectos que intervienen en el funcionamiento de un robot como es la velocidad, la capacidad de carga, y la exactitud que son aspectos muy importantes en la consideración del buen funcionamiento de los robots.

Capítulo 4

Estructura mecánica del robot cartesiano

En la estructura de cualquier configuración robótica, es necesario tener un buen diseño mecánico de todos los componentes para tener un buen funcionamiento de este. Es de gran importancia que cada parte del robot sea diseñada de manera perfecta para no tener problemas tales como fricciones, retardos, desgastes, etc. De la misma manera saber que material es el ideal para el diseño de cada pieza.

4.1. Acero

El acero es una aleación, es decir, un metal mezclado que se logra derritiendo y uniendo diferentes materiales. Actualmente existen más de 2 500 clases de acero estándar en todo el mundo. Todos ellos están hechos principalmente con lingotes de hierro que, a su vez, están conformados por el elemento hierro, más un tres por ciento de carbón. El lingote de hierro es extraído del hierro mineral en los altos hornos de las fundiciones. Luego es procesado en la acería para obtener un acero con menos del dos por ciento de carbón. Esta baja proporción suaviza el material, haciéndolo más fácil de procesar[41].

El desarrollo del horno de alta temperatura en el Siglo XIV lo hizo posible: el hierro podía ser calentado hasta que tomara forma líquida. Pero la tecnología sólo madura gradualmente: mientras que en el Siglo XVII aún se necesitaban ocho toneladas de carbón para obtener dos toneladas de lingotes de hierro, actualmente necesitamos sólo alrededor de media tonelada de coque para producir 10 000 toneladas de lingotes de hierro por día.

En 1912, científicos del grupo alemán de trabajadores siderúrgicos Krupp, descubrieron accidentalmente cómo fabricar acero a prueba de óxido. También conocido como *V2A* o acero inoxidable. Este material está compuesto por hierro, cromo y níquel, y se utiliza, por ejemplo, en tecnología médica.

Actualmente, el acero se ha convertido en un material de alta tecnología. Por ejemplo, el acero conocido como HDS (de alta fuerza y ductilidad) hace posible las *zonas de*

pliegue inteligentes: la idea es que este material, que se deforma fácilmente, se vuelve más resistente luego de una colisión debido a transformaciones estructurales, por lo que brinda una mayor protección. Las carrocerías de vehículos confeccionadas con este *acero deformante*, no sólo incrementarían la seguridad: también serían particularmente livianas, lo que contribuiría a disminuir el consumo de energía[42].

4.1.1. Clasificación del acero

Los diferentes tipos de acero se clasifican de acuerdo a los elementos de aleación que producen distintos efectos en el Acero.

Aceros al carbono

Más del 90 por ciento de todos los aceros son aceros al carbono. Estos aceros contienen diversas cantidades de carbono y menos del 1.65 por ciento de manganeso, el 0.60 por ciento de silicio y el 0.60 por ciento de cobre. Entre los productos fabricados con aceros al carbono figuran máquinas, carrocerías de automóvil, la mayor parte de las estructuras de construcción de acero, cascos de buques, somieres y horquillas. Véase[43].

Aceros aleados

Estos aceros contienen una proporción determinada de vanadio, molibdeno y otros elementos, además de cantidades mayores de manganeso, silicio y cobre que los aceros al carbono normales. Estos aceros de aleación se pueden subclasificar en :

- Estructurales: son aquellos aceros que se emplean para diversas partes de máquinas, tales como engranajes, ejes y palancas. Además se utilizan en las estructuras de edificios, construcción de chasis de automóviles, puentes, barcos y semejantes. El contenido de la aleación varía desde 0.25 por ciento a un 6 por ciento .
- Para herramientas: aceros de alta calidad que se emplean en herramientas para cortar y modelar metales y no-metales. Por lo tanto, son materiales empleados para cortar y construir herramientas tales como taladros, escariadores, fresas, terrajas y machos de roscar.
- Especiales: Los Aceros de aleación especiales son los aceros inoxidables y aquellos con un contenido de cromo generalmente superior al 12 por ciento. Estos aceros de gran dureza y alta resistencia a las altas temperaturas y a la corrosión, se emplean en turbinas de vapor, engranajes, ejes y rodamientos.

Aceros de baja aleación ultraresistentes

Esta familia es la más reciente de las cuatro grandes clases de acero. Los aceros de baja aleación son más baratos que los aceros aleados convencionales ya que contienen

cantidades menores de los costosos elementos de aleación. Sin embargo, reciben un tratamiento especial que les da una resistencia mucho mayor que la del acero al carbono. Por ejemplo, los vagones de mercancías fabricados con aceros de baja aleación pueden transportar cargas más grandes porque sus paredes son más delgadas que lo que sería necesario en caso de emplear acero al carbono. Además, como los vagones de acero de baja aleación pesan menos, las cargas pueden ser más pesadas. En la actualidad se construyen muchos edificios con estructuras de aceros de baja aleación. Las vigas pueden ser más delgadas sin disminuir su resistencia, logrando un mayor espacio interior en los edificios.

Aceros inoxidables

Los aceros inoxidables contienen cromo, níquel y otros elementos de aleación, que los mantienen brillantes y resistentes a la herrumbre y oxidación a pesar de la acción de la humedad o de ácidos y gases corrosivos. Algunos aceros inoxidables son muy duros; otros son muy resistentes y mantienen esa resistencia durante largos periodos a temperaturas extremas. Debido a sus superficies brillantes, en arquitectura se emplean muchas veces con fines decorativos. El acero inoxidable se utiliza para las tuberías y tanques de refinerías de petróleo o plantas químicas, para los fuselajes de los aviones o para cápsulas espaciales. También se usa para fabricar instrumentos y equipos quirúrgicos, o para fijar o sustituir huesos rotos, ya que resiste a la acción de los fluidos corporales. En cocinas y zonas de preparación de alimentos los utensilios son a menudo de acero inoxidable, ya que no oscurece los alimentos y pueden limpiarse con facilidad[43].

4.2. Aluminio

El aluminio metálico fue obtenido en forma pura, por primera vez, en 1825, por Oersted, quien calentó el cloruro de aluminio con una amalgama de potasio y mercurio. En 1854, Henri Sainte- Claire Deville produjo aluminio a partir de cloruro de Na-Al, calentándolo con sodio metálico. El proceso funcionó durante unos 35 años, y el metal se vendía a 220 dólares el *Kg*. Ya para 1886 el precio se había reducido a 17 dólares el *Kg*. En 1886, Charles Hall produjo el primer aluminio por el proceso actual, a gran escala; Esto es la electrólisis de la alúmina en un baño de criolita fundida. En el mismo año Paul Heroult obtuvo una patente francesa por un proceso similar al de Hall. En 1893, la producción de aluminio había aumentado ya tan rápidamente por el método de Hall que el precio se había desplomado hasta 4.40 dólares el *Kg*. La industria creció en forma segura, basada firmemente en los mercados nuevos y en aumento creados, sobre todo, por sus propios estudios sobre las propiedades del aluminio y las rutas de consumo económico de este nuevo metal. El Aluminio es el metal más abundante en el mundo. Constituye el 8 por ciento de la porción sólida de la corteza terrestre. Todos los países poseen grandes existencias de materiales que contienen aluminio, pero los procesos

para obtener aluminio metálico a partir de la mayor parte de estos compuestos no son económicos todavía[44].

Generalmente se reconoce a Friedrich Wöhler el aislamiento del aluminio en 1827. Aún así, el metal fue obtenido, impuro, dos años antes por el físico y químico danés Hans Christian Orsted. En 1807, Humphrey Davy propuso el nombre aluminum para este metal aún no descubierto, pero más tarde decidió cambiarlo por *aluminium* por coherencia con la mayoría de los nombres de elementos, que usan el sufijo *-ium*.

4.2.1. Características físicas

Entre las características físicas del aluminio se tienen las siguientes:

- Es un metal ligero, cuya densidad o peso específico es de 2700 kg/m^3
- Tiene un punto de fusión bajo 660°C (933K) *El peso atómico del aluminio es de 26.9815*
- Es de color blanco brillante.
- Buen conductor del calor y de la electricidad.
- Resistente a la corrosión.
- Material abundante en la Naturaleza
- Material fácil y barato de reciclar

4.2.2. Características mecánicas

En las características mecánicas del aluminio están las siguientes:

- De fácil mecanizado.
- Muy maleable, permite la producción de láminas muy delgadas.
- Bastante dúctil, permite la fabricación de cables eléctricos.
- Material blando (Escala de Mohs *2-3*). Límite de resistencia en tracción $160-200 \text{ N/mm}^2$ [$160-200 \text{ MPa}$] en estado puro, en estado aleado el rango es de $1400-6000 \text{ N/mm}^2$. El duraluminio es una aleación particularmente resistente.
- Material que forma aleaciones con otros metales para mejorar las propiedades mecánicas.
- Permite la fabricación de piezas por fundición y moldeo.
- Material soldable

4.2.3. Aplicaciones del aluminio

Un volumen dado de aluminio pesa menos que $1/3$ del mismo volumen de acero. Los únicos metales más ligeros son el litio, el berilio y el magnesio. Debido a su elevada proporción resistencia-peso es muy útil para construir aviones, vagones ferroviarios y automóviles, y para otras aplicaciones en las que es importante la movilidad y la conservación de energía. Por su elevada conductividad del calor, el aluminio se emplea en utensilios de cocina y en pistones de motores de combustión interna. Solamente presenta un 63 de la conductividad eléctrica del cobre para alambres de un tamaño dado, pero pesa menos de la mitad. Un alambre de aluminio de conductividad comparable a un alambre de cobre es más grueso, pero sigue siendo más ligero que el de cobre. El peso tiene mucha importancia en la transmisión de electricidad de alto voltaje a larga distancia, y actualmente se usan conductores de aluminio para transmitir electricidad a 700.000 voltios o más[45].

El metal es cada vez más importante en arquitectura, tanto con propósitos estructurales como ornamentales. Las tablas, las contraventanas y las láminas de aluminio constituyen excelentes aislantes. El metal se utiliza también en reactores nucleares a baja temperatura porque absorbe relativamente pocos neutrones. Con el frío, el aluminio se hace más resistente, por lo que se usa a temperaturas criogénicas. El papel de aluminio de 0.018 *cm* de espesor, actualmente muy utilizado en usos domésticos, protege los alimentos y otros productos perecederos. Debido a su poco peso, a que se moldea fácilmente y a su compatibilidad con comidas y bebidas, el aluminio se usa mucho en contenedores, envoltorios flexibles, y botellas y latas de fácil apertura. El reciclado de dichos recipientes es una medida de conservación de la energía cada vez más importante. La resistencia a la corrosión al agua del mar del aluminio también lo hace útil para fabricar cascos de barco y otros mecanismos acuáticos.

Puede prepararse una amplia gama de aleaciones recubridoras y aleaciones forjadas que proporcionen al metal más fuerza y resistencia a la corrosión o a las temperaturas elevadas. Algunas de las nuevas aleaciones pueden utilizarse como planchas de blindaje para tanques y otros vehículos militares.

4.3. Hule

El hule (palabra náhuatl) es un polímero natural o sintético, en el primer caso echo de la savia de plantas específicas, como por ejemplo la Castilla elástica y el guayule.

4.3.1. El hule natural

La posibilidad de diseñar las propiedades de los plásticos se inició con los intentos de los químicos por imitar a la naturaleza en la síntesis del hule natural, cuyas propiedades elásticas se conocen desde hace aproximadamente cinco siglos. Originario

de América, el hule era utilizado por los primeros habitantes de México, quienes lo extraían del arbusto del guayule para impermeabilizar utensilios y fabricar las bolas del juego de pelota. Aunque los colonizadores españoles también lo usaron, durante los siguientes 300 años, sólo se empleó para fabricar pelotas y otras curiosidades[46].

La primera fábrica que usó hule natural fue establecida en París en 1803. En un principio producía ligas y tirantes, pero después incorporó hule en las gabardinas para hacerlas impermeables. Estos productos tenían algunos inconvenientes, pues el hule se reblandecía y se hacía pegajoso en verano mientras que en invierno era rígido y quebradizo. La solución llegó en 1839, cuando Charles Goodyear descubrió que al calentar hule natural en presencia de azufre se obtenía un producto mucho más elástico y resistente. A este proceso se le llamó vulcanización. Los usos del hule vulcanizado aumentaron y se diversificaron; pronto las plantaciones en América resultaron insuficientes para cubrir la demanda, y la posibilidad de controlar este nuevo mercado impulsó a los ingleses a establecer enormes plantaciones en Ceilán, Malasia y Singapur, con semillas del árbol *Hevea brasiliensis* llevadas de contrabando desde Brasil. Durante la Segunda Guerra Mundial la imposibilidad de importar hule natural para las llantas de los aviones propició que en los laboratorios de investigación, estadounidenses y alemanes, se iniciara la búsqueda por descubrir, reproducir y mejorar los secretos de la estructura química del hule.

Plástico o polímero

Hoy sabemos que el hule natural, al igual que el pelo, la lana, las telarañas, los capullos de las polillas y el gusano de seda, así como muchas otras sustancias provenientes de organismos vivos, son polímeros. Los polímeros son moléculas gigantes que resultan de la unión de miles de moléculas más pequeñas a las que se llama monómeros. La estructura básica de estas sustancias es similar a la de una larga cadena (el polímero) formada por pequeños eslabones (los monómeros). En algunos polímeros las cadenas lineales pueden unirse entre sí a través de otras cadenas, dando lugar a redes que pueden formar arreglos tridimensionales. Los eslabones pueden ser todos iguales, lo que se denomina homopolímero, o pueden estar constituidos por unidades diferentes y alternarse siguiendo un patrón específico formando lo que se conoce como un copolímero.

Las sorprendentes y variadas propiedades que pueden darse a estas sustancias dependen precisamente del tipo de monómero, la longitud de las cadenas y la forma en que éstas se acomodan en el material. Aunque generalmente se utilizan los términos *plástico* y *polímero* como sinónimos, plástico hace referencia a cualquier material que puede moldearse fácilmente, mientras que polímero clasifica a una sustancia por su estructura molecular. A pesar de que todos los plásticos comerciales conocidos son polímeros, algunas de las biomoléculas más importantes los plásticos.

Materiales asombrosos

Aunque algunos plásticos y sus propiedades nos son muy familiares, como el polietileno en las bolsas del supermercado, el poliestireno en los vasos desechables, el poliuretano en el hule espuma, el PVC en las tuberías y el nailon en la ropa, las posibilidades en el diseño de estos materiales han llegado mucho más lejos. Un ejemplo es el Kevlar, material elaborado con un polímero llamado parafenileno tereftalamida, el cual es hasta cinco veces más fuerte que el acero, pero mucho más liviano, por lo que ha resultado ideal para construir partes de aviones y naves espaciales ya que la relación fortaleza/peso es de gran importancia en dichos vehículos (se requiere de mucho menos combustible para mover un objeto ligero). El Kevlar es usado para la fabricación de la cola de algunos de los aviones más grandes, las puntas de algunos cohetes, chalecos y en discos y bastones para hockey[46]. Gracias a este material, la posibilidad de disponer de automóviles más ligeros y resistentes parece muy cercana. Cinco veces más fuerte que el acero, el Kevlar es ideal para la fabricación de algunos cohetes. Árbol de hule.

Desde la aparición en el mercado de los primeros plásticos se pensó que todos estos materiales eran aislantes (es decir, que no conducían la electricidad). Aunque esto es cierto en la mayoría de los casos, se han desarrollado los llamados *metales sintéticos* que son plásticos conductores que semejan propiedades metálicas a pesar de que no contienen ni un átomo metálico. Estos pocos ejemplos muestran que las posibilidades de aplicación de los plásticos son muy grandes. Además, estos materiales sintéticos se pueden producir en su totalidad a partir del petróleo.

A diario utilizamos en los diferentes productos diferentes tipos de plásticos que son utilizados tanto en el hogar, la construcción o para el envasado de diferentes productos. Algunos de estos plásticos son Politereftalato de etileno (*PET*), Polietileno de alta densidad (*HDPE*), Policloruro de vinilo (*PVC*), Polietileno de baja densidad (*LDPE*), polipropileno (*PP*), poliestireno (*PS*) y espuma de poliestireno(*EPS*).

4.4. Procedimientos de elaboración del prototipo

El primer paso que se tomó fue la obtención de la placa de acero collroll *1018* con dimensiones de 51 x 50 cm, para el prototipo, posteriormente se realizó la delimitación del área de trabajo, esto con el fin de saber en donde se pondrían los soportes de las barras de acero *12L14* cromado y perfectamente alineadas, para esto se tomo las bases de la barras guías las cuales estarían a una distancia de 0.34m. Cada par estan en paralelo a otro par con una distancia de 0.30m todo esto se realizó con herramienta precisa para que fuesen correctas las medidas. En el maquinado de esto se utilizó un torno paralelo con buriles especiales para este tipo de material esto con la finalidad de obtener un corte fino.



Figura 4.1: Herramienta de maquila(Torno paralelo).[18]

Para los orificios de las bases se utilizó también un taladro de columna mediano con diferentes brocas, también en la parte de la base se realizaron perforaciones para poder sujetar a las bases con tornillos de $1/4$ se colocaron 2 en cada base y se utilizó un machuelo de esta medida para realizar la cuerda de los orificios. El motor se colocó en una esquina del prototipo en paralelo con un soporte que detiene a una barra de acero $12L14$ cromado que transfiere el torque hacia el otro extremo del modelo con una base de aluminio que soporta a la misma.



Figura 4.2: Herramienta de perforación(Taladro).[20]

Para la elaboración de los soportes de las barras se compró aluminio *No. 416*, cuadrado de una pulgada, y todos se cortaron de 5 cm cada uno y se les realizó unos orificios en a 3.5 cm de altura para colocar las barras de soporte transversales. Para esto fue necesario utilizar un taladro de columna, el vernier fue una herramienta indispensable para la exactitud en los trazos de los orificios a cada uno de los soportes. Se realizó una perforación para ponerles opresores allem para mantener fijas las barras y se utilizó un machuelo de $3/16\text{ in}$ para poder realizar la cuerda para colocar los opresores.



Figura 4.3: Herramienta de medición(Vernier).[22]

En la parte inferior y por la parte frontal superior se hicieron los orificios para las barras de soporte del carro esto a cada una de las piezas a demás se colocó una guía de soportes de solera de $1/4$ de pulgada acerada para evitar el desvalanceo de los soportes todo esto con ayuda de herramienta necesaria como arco de segueta, tornillo de banco, vernier, taladro y brocas.

En la elaboración del carro se utilizó aluminio *No 416* rectangular de 1×2 pulgadas y se le hicieron los orificios para introducir las barras de soporte a si mismo se le colocaron bujes de bronce SAE62 para evitar la fricción y mejorar el deslizamiento de el carro y a su vez en este se puso la base para el motor que moverá a el lápiz. Esto fue hecho con una pequeña placa de aluminio y un motor pequeño de CC con un engrane.

Para el lápiz se utilizó acero *12L14* de $7/16$ de pulgada, cromado perforado casi en su totalidad para reducir el peso del mismo, se realizó la cremallera con paso de 1.5 n/d 36 y los engranes fueron de aluminio *No 416* cilíndrico de 1 pulgada con paso de 1.5 n/d 36. Esto fue posible de realizar con una fresadora universal *No 4* con plato divisor de $40/1$ con un cortador de paso 1.5 .



Figura 4.4: Fresadora universal.[21]

Para el desplazamiento de las barras se utilizaron bandas de caucho con un ancho de 0.3 cm con un paso de 1.5 o también conocidas de $1/4$ de pulgada. Estas van acopladas en los diferentes engranes de las diferentes posiciones de los ejes.

4.5. Robot cartesiano

En la realización de la estructura del robot cartesiano se tornearon cada una de las piezas en un taller, las cuales fueron echas por nosotros mismos cuidando cada detalle en la tecnica de torneado y fresado de piezas mecánicas. Básicamente el robot funciona con el principio de las copiadoras las cuales utilizan motores a pasos. Para poder armar el robot se utilizaron:

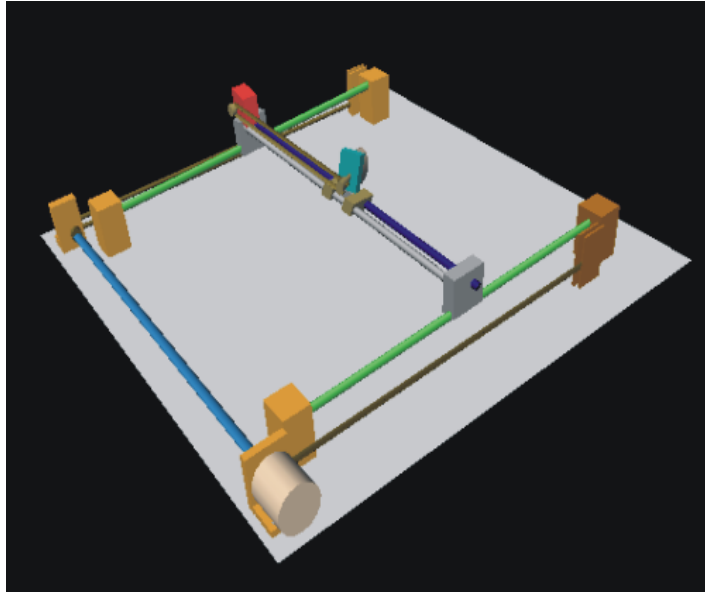


Figura 4.5: Robot Cartesiano

- Placa coll-roll 1018.
- Barras guías de acero cromado 12L14.
- Soportes de aluminio No. 416.
- Carro de aluminio con bujes de bronce: SAE62.
- Engranés de aluminio: P/D 1.5, N/D 36.
- Soportes longitudinales, aluminio, No. 416 con bujes de bronce SAE62.
- Perno cremallera: P/D 1.5.
- Guía de soportes de solera acerada de 1/4.

4.5.1. Placa coll-roll

El proyecto de robot cartesiano consta de una placa de acero, llamada placa de coll-roll 1018(ver figura 4.6), como base y área de trabajo del mismo, tiene medidas de 51 *cm* de largo por 50 *cm* de ancho, con un espesor de 0.5 *cm*. El área de trabajo dentro de la propia placa abarca 12*cm* X 14*cm*). Esta base tiene un peso de 7 *Kg* esto ayuda a que la el robot tenga una seguridad de fijación en el lugar de trabajo sin estar estático y pueda ser movable para su traslado.

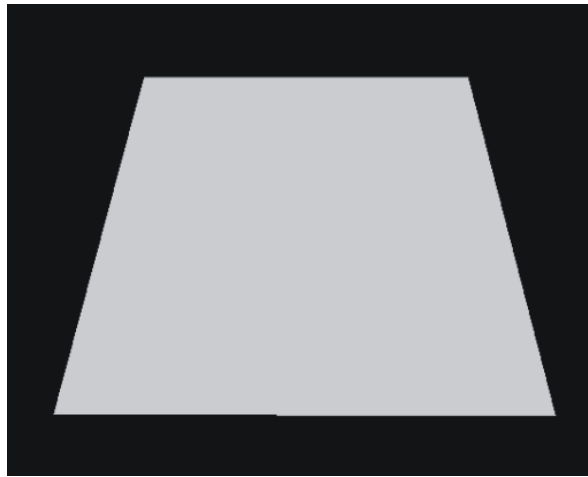


Figura 4.6: Placa de acero Coll-Roll

4.5.2. Barras guías eje Y

Las barras de deslizamiento en el eje *Y*, que son las barras guías de acero cromado las cuales se encuentran fijas cada una en dos de los cuatro soportes ubicados en las esquinas de la placa. Los soportes de aluminio 416, son echos de acero inoxidable y sin ningún tipo de cuerda a fin evitar alguna fricción. Las medidas de estas barras ayudan a darle un mayor espacio en la realizacion de trayectorias al robot cartesiano, las cuales son dimensiones de 42 *cm* de longitud y con 8 *mm* de diámetro(figura 4.7).

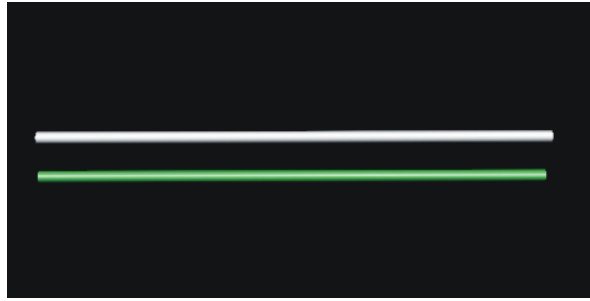


Figura 4.7: Barras guías del eje Y

4.5.3. Barra guía eje X

Esta barra se encuentra situada en el engrane de mayor tamaño y a su vez en un soporte pequeño el cual contiene un buje para su movimiento de forma circular lo cual ayuda a generar las trayectorias en las coordenadas en Y(figura 4.8). Debido a que esta barra esta manipulada a través de dos bandas que a su vez se sitúan dos soportes pequeños respectivamente, y hace que el perno/cremallera se desplace en el eje Y.

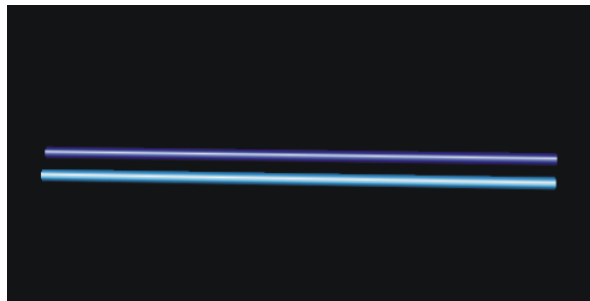


Figura 4.8: Barras guías del eje X

4.5.4. Soportes de engranes

Estos sujetan a los engranes situados en los extremos del eje Y cada uno ubicado uno en el motor principal y otro en un soporte. Estos soportes ayudan a que las bandas esten situadas en un lugar fijo y aparte de los soportes principales además de dar la estabilidad en el trazo sobre el eje Y(figura 4.9).

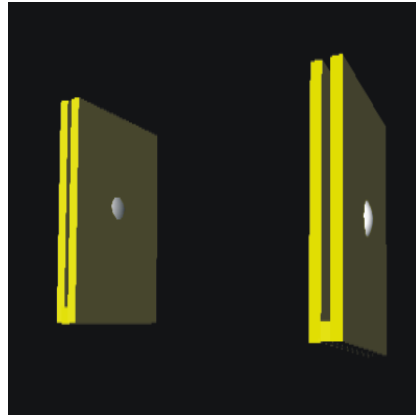


Figura 4.9: Soportes de engranes

4.5.5. Soportes 416

Estos están situados en las cuatro esquinas de la placa a fin de las barras guías del eje Y (ver figura 4.10). Estan debidamente calibrados, es decir que tienen una misma altura de fijación de las barras a fin de no causar fricción en el caso de que no estuviesen en un mismo nivel. Lo cual a su vez ayudan a buen deslizamiento de la barra perpendicular donde los soportes longitudinales ayudan a realizar la tarea.

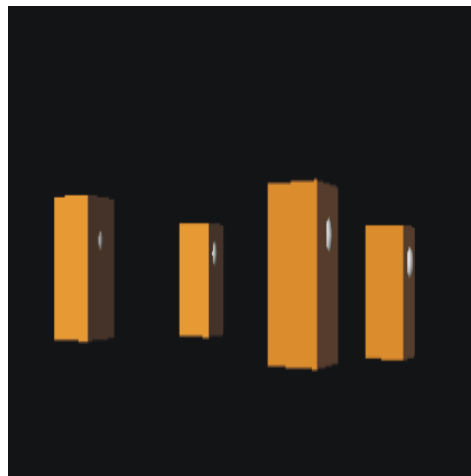


Figura 4.10: Soportes principales de las guías

4.5.6. Soportes longitudinales

Estos contienen dos perforaciones de manera paralela a fin de situar la barra guía en el eje X (figura 4.11). En los costados contienen una perforación a fin de que se deslice en ese orificio la barra guía del eje Y en cada uno de los soportes, además de que están separadas por una guía de solera acerada.

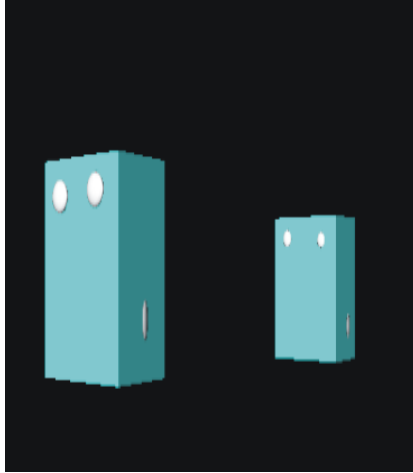


Figura 4.11: Soportes longitudinales

4.5.7. Barra del motor principal

Esta barra depende del movimiento del motor principal y hace que las barras guía del eje X tengan un movimiento (figura 4.12), debido a que los engranes que se encuentran en el motor y otro en un soporte de guía actúan sobre las bandas y a su vez estas tienen el movimiento, para desplazar los ejes en un desplazamiento en el eje Y .

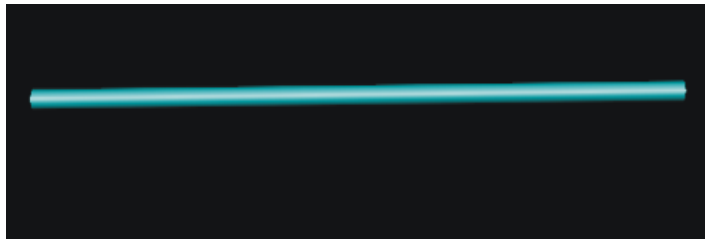


Figura 4.12: Barra del motor principal

4.5.8. Carro de desplazamiento

Hecho de aluminio este tiene la función de llevar el perno cremallera (figura 4.13), para que debido a los desplazamientos que se tengan en los dos ejes este junto con el perno se encarguen de trazar la trayectoria en la superficie que se haya asignado en el área específica, trabajando en esto en un material plano y blando (papel). Este también tiene un motor incrustado en la parte superior el cual mantendrá al perno inactivo hasta encontrar la primera coordenada a trazar.

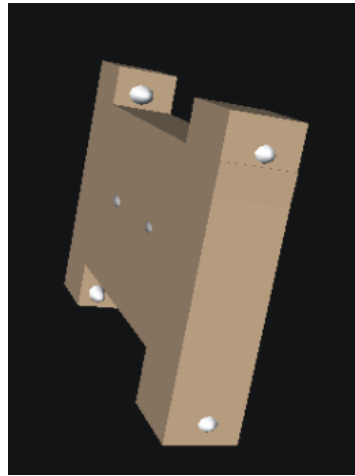


Figura 4.13: Carro de deslizamiento

4.5.9. Engranajes

Hechos de aluminio y tiene especificaciones de paso por diámetro de 1.5 y teniendo 36 dientes en su perímetro (figura 4.14). Estos están acoplados al eje principal y a su vez a un soporte y de manera conjunta trabajan con las bandas para el desplazamiento en el eje Y.



Figura 4.14: Engranajes del robot

4.5.10. Motor de movimiento en X

Este es un motor en forma rectangular el cual esta situado en uno de los soportes longitudinales y actua de manera conjunta con una segunda banda ajustada en un motor de *c.d.* y que hace el movimiento de desplazamiento en el eje X desplazando el carro que sostiene a l motor de *c.d.* y a su vez al lápiz que traza la trayectoria(figura 4.15).

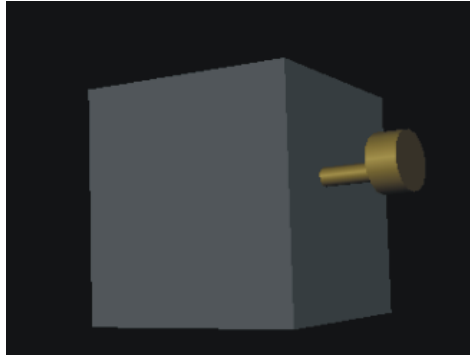


Figura 4.15: Motor de movimiento del eje X

4.5.11. Soporte de eje de motor principal

Este está acoplado al eje del motor principal el cual tiene un engrane acoplado al eje que a su vez tiene una de las bandas en el eje Y y de manera conjunta hacen el desplazamiento en el eje Y(figura 4.16). Este esta hecho de aluminio el cual ayuda a que de forma conjunta con las demás piezas sea más ligero el robot.

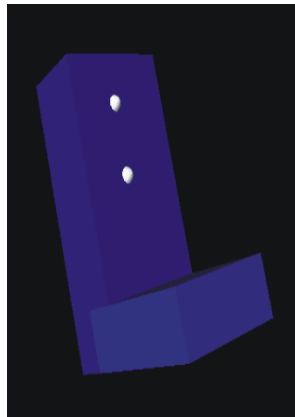


Figura 4.16: Soporte de barra de motor principal

4.5.12. Perno o cremallera

Este fue realizado a través de la fresadora para darle una forma de cuerda de 1.5, para que pudiese ser movido por el motor con el que tiene contacto. Tiene la función de trazar la trayectoria asignada en la pantalla de trazado, obedeciendo a un pulso que viene de la Pc a través del puerto paralelo para activar un puente H y dar el sentido de giro del motor para que el perno pueda subir o bajar según sea la activación de giro de este(ver figura 4.17).

Este dispositivo es una especie de cilindro hueco, a manera de evitar fricción en el motor, con un diámetro aproximado de 8 mm y una longitud de 10 cm, teniendo el orificio a manera de poder insertar cualquier material que pueda trazar en el área especificada.

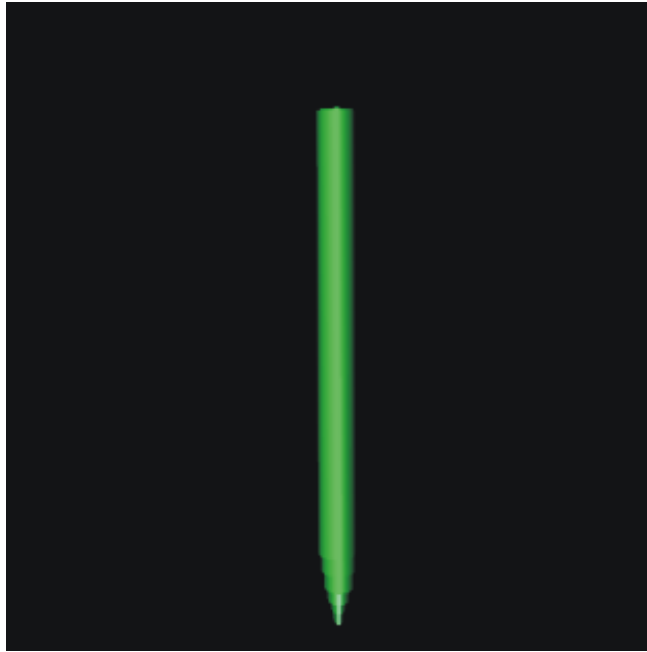


Figura 4.17: Soporte de barra de motor principal

4.6. Conclusiones

El estudio de los diferentes materiales utilizados no solo sirve para saber como darle el uso a este robot sino también para saber el tipo de material para escoger en el caso de hacer una modificación al proyecto, o en el mejor de los casos hacer un nuevo robot. Esto ayuda a saber que material nos conviene para la elaboración de piezas de robots.

Además podemos observar que nos da una breve explicación de como se maquinó cada pieza, las medidas de estas y el funcionamiento que tiene cada una de ellas en nuestro proyecto. Para el maquinado de las piezas fue muy importante el saber el uso de las herramientas de elaboración en las que abarca: el torno paralelo, taladro, fresa, vernier, etc., ya que si no se sabe como manejarlas se puede hacer un desperdicio de material que hoy en dia no es nada barato.

Capítulo 5

Etapa de potencia

La parte de la electrónica en cualquier prototipo automatizado es una de las partes fundamentales ya que a través de esta nuestro sistema puede ser o no ser óptimo en las tareas de automatización que realice. Dependiendo de la función que el sistema automatizado desee realizar, serán los componentes electrónicos que conlleven a su diseño, ya sean transistores, diodos, resistencias o circuitos integrados entre otros. En la elaboración de esta parte en el proyecto del robot cartesiano se consideran transistores para ganancias de voltaje, diodos para una mayor seguridad en el manejo de corrientes, optoacopladores que al igual que los diodos sirven de protección, además de tomar en cuenta la parte de control que se realiza a través del puerto paralelo de la Pc.

5.1. Motores a paso

A diferencia de los demás motores, los motores a pasos no son de corriente de c.d. ni de c.a., sino que su funcionamiento obedece a un tren de pulsos previamente definidos, energizando a cada una de las bobinas en el estator. Cada pulso energiza una de las bobinas dando un paso en un sentido de motor en el cual puede ser desde los 90° hasta los 0.9° , esto hace que nosotros controlemos sus pasos con mayor o menor precisión dependiendo de la longitud del paso. También dependiendo de la frecuencia con que apliquemos el tren de pulsos podemos obtener un control de la velocidad del motor y aplicando un orden diferente en el tren de pulsos se obtiene un sentido de giro diferente. Este tema y los demás subtemas lo podemos ver en la bibliografía[33].

En la actualidad estos motores son utilizados en manipulación de discos duros, robots, impresoras, unidades de CD ROM, DVD o impresoras. Una de las desventajas que presentan los motores a pasos es que se debe encontrar su tiempo de equilibrio, es decir, que si no le damos el tiempo necesario en realizar su primer paso puede que este funcione de manera de vaivén, moverse en sentido contrario o simplemente no avanzar.

5.1.1. Tipos de motores a pasos

Una forma de conseguir motores PAP de paso mas reducido, es la de aumentar el número de bobinas del estator, pero ello llevaría a un aumento del coste y del volumen y a pérdidas muy considerable en el rendimiento del motor, por lo que esta situación no es viable. Hasta ahora y para conseguir la solución más idónea, se recurre a la mecanización de los núcleos de las bobinas y el rotor en forma de hendiduras o dientes, creándose así micropolos magnéticos, tantos como dientes y estableciendo las situaciones de equilibrio magnéticos con avances angulares mucho menores, siendo posible conseguir motores de hasta de 500 pasos.

Dentro de la clasificación de los motores a pasos se obtienen los siguientes:

De imán permanente

Es el modelo en el cual el rotor es un imán permanente en el que se mecanizan un número de dientes limitado por su estructura física. Ofrece como principal ventaja que su posicionamiento no varía aún sin excitación y en régimen de carga.

De reluctancia variable

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

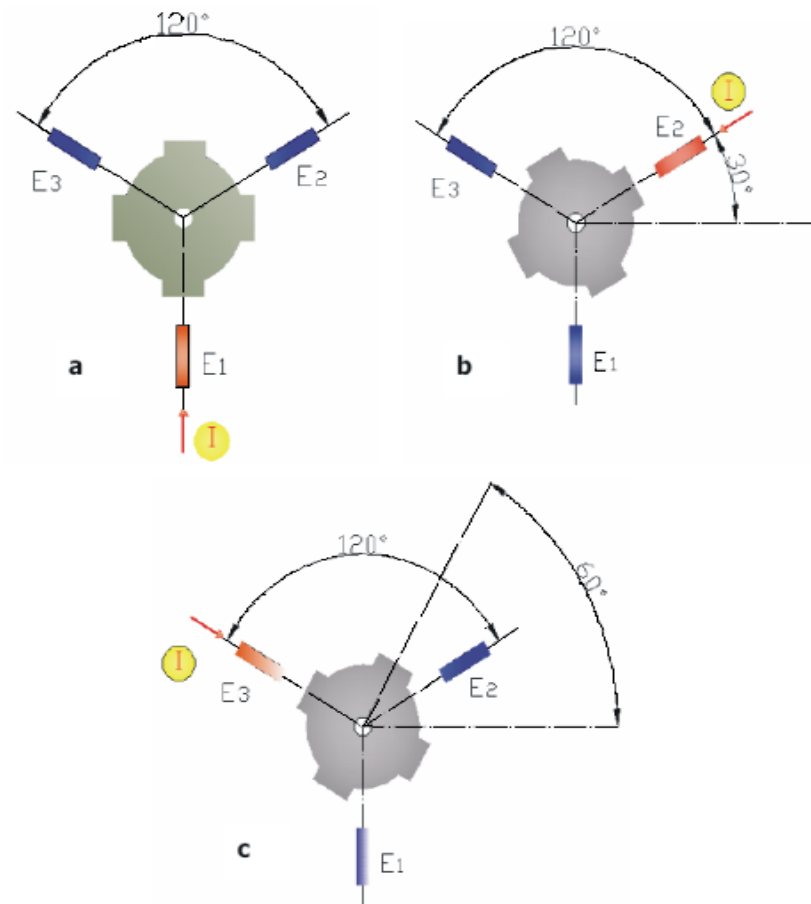


Figura 5.1: Funcionamiento de un motor paso a paso de reluctancia variable.[33]

Híbridos

Son combinación de los dos tipos anteriores; el rotor suele estar constituido por anillos de acero dulce dentado en un número ligeramente distinto al del estator y dichos anillos montados sobre un imán permanente dispuesto axialmente.

Motores unipolares

En este tipo de motores, todas las bobinas del estator están conectadas en serie formando cuatro grupos. Esto a su vez, se conectan dos a dos, también en serie, y se montan sobre dos estatores diferentes, tal y como se aprecia en la figura 5.2. Según puede apreciarse en dicha figura, del motor paso a paso salen dos grupos de tres cables, uno de los cuales es común a dos bobinados. Los seis terminales que parten del motor, deben ser conectados al circuito de control, el cual, se comporta como cuatro conmutadores

electrónicos que, al ser activados o desactivados, producen la alimentación de los cuatro grupos de bobinas con que está formado el estator. Si generamos una secuencia adecuada de funcionamiento de estos interruptores, se pueden producir saltos de un paso en el número y sentido que se desee.

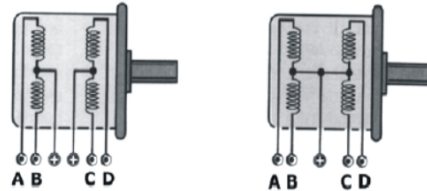


Figura 5.2: Funcionamiento de un motor unipolar.[33]

Motores bipolares

En este tipo de motores las bobinas del estator se conectan en serie formando solamente dos grupos, que se montan sobre dos estatores, tal y como se muestra en la figura 5.3. Según se observa en el esquema de este motor salen cuatro hilos que se conectan, al circuito de control, que realiza la función de cuatro interruptores electrónicos dobles, que nos permiten variar la polaridad de la alimentación de las bobinas. Con la activación y desactivación adecuada de dichos interruptores dobles, podemos obtener las secuencias adecuadas para que el motor pueda girar en un sentido o en otro.

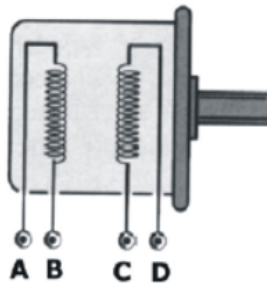


Figura 5.3: Funcionamiento de un motor bipolar.[33]

5.1.2. Parámetros de los motores a pasos

Desde el punto de vista mecánico y eléctrico, es conveniente conocer el significado de algunas de las principales características y parámetros que se definen sobre un motor

paso a paso.

- Par dinámico de trabajo: depende de sus características dinámicas y es el momento máximo que el motor es capaz de desarrollar sin perder paso, es decir, sin dejar de responder a algún impulso de excitación del estator y dependiendo, evidentemente, de la carga.
- Par de mantenimiento: es el par requerido para desviar, en régimen de excitación, un paso el rotor cuando la posición anterior es estable ; es mayor que el par dinámico y actúa como freno para mantener el rotor en una posición estable dada.
- Par de detención: es un par de freno que siendo propio de los motores de imán permanente, es debida a la acción del rotor cuando los devanados del estator están desactivados.
- Angulo de paso: se define como el avance angular que se produce en el motor por cada impulso de excitación[34]. Se mide en grados, siendo los pasos estándar más importantes los siguientes:

Grados por impulso de excitación	Nº de pasos por vuelta
0,72°	500
1,8°	200
3,75°	96
7,5°	48
15°	24

Figura 5.4: Grados angulares por cada impulsión del motor[24]

- Número de pasos por vuelta: Es la cantidad de pasos que ha de efectuar el rotor para realizar una revolución completa; evidentemente es:

$$NP = \frac{360^\circ}{\alpha} \quad (5.1)$$

donde NP es el número de pasos y α el ángulo de paso.

- Frecuencia de paso máximo (Maximum pull-in/out: define como el máximo número de pasos por segundo que puede recibir el motor funcionando adecuadamente.
- Momento de inercia del rotor: Es su momento de inercia asociado que se expresa en gramos por centímetro cuadrado.

- Par de mantenimiento, de detención y dinámico: Definidos anteriormente y expresados en miliNewton por metro.

5.1.3. Funcionamiento del motor bipolar

Este tipo de motores esta compuesto por dos bobinas o polos, en las cuales estará circulando la corriente en un sentido u en otro según la secuencia definida.

Mientras no circule ninguna corriente en ninguna de las bobinas el rotor quedará en reposo en una posición cualquiera. Si aplicamos corrientes en las dos bobinas el rotor girará en un sentido, si por el contrario se invierten las corrientes el sentido seguirá un mismo sentido en el siguiente paso en el cual se da un desplazamiento de 90° . De manera que invirtiendo las polaridades de las corrientes se obtiene un giro completo del motor.

Como cada vez que se aplica un pulso distinto a la entrada del motor, éste gira un paso completo se dice que está funcionando en modo paso completo. En este caso el paso es de 90 grados, demasiado grande para poder realizar ningún tipo de control.

Para aumentar la resolución, se tienen varias opciones. La más sencilla de todas y que no necesita un cambio constructivo del motor, consiste en cambiar la secuencia de alimentación. En modo de funcionamiento de paso completo, las bobinas nunca quedan sin alimentación. Pues bien, si entre cada cambio en la tensión de alimentación de una bobina, esta se deja sin alimentar, podemos conseguir una posición del rotor intermedia entre dos pasos. A esta forma de funcionamiento se le denomina medio paso.

PASO	TERMINALES			
	A	B	C	D
1	+V	-V	+V	-V
2	+V	-V	-V	+V
3	-V	+V	-V	+V
4	-V	+V	+V	-V

Figura 5.5: Secuencia en un motor bipolar.[24]

5.1.4. Funcionamiento de motores unipolares

Recordemos que un motor unipolar está compuesto por dos bobinas con una toma intermedia cada una, y su principal característica es que la intensidad que circula por cada bobina siempre lo hace en la misma dirección. Al igual que en los motores bipolares, encontramos dos modos de paso completo y medio paso.

Existen tres secuencias posibles para este tipo de motores, las cuales se detallan a continuación. Todas las secuencias comienzan nuevamente por el paso 1 una vez alcanzado el paso final (4 u 8). Para revertir el sentido de giro, simplemente se deben ejecutar las secuencias en modo inverso.

Secuencia normal

Esta es la secuencia más usada y la que generalmente recomienda el fabricante. Con esta secuencia el motor avanza un paso por vez y debido a que siempre hay al menos dos bobinas activadas, se obtiene un alto torque de paso y de retención[35].



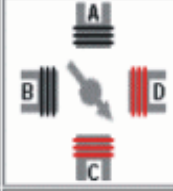
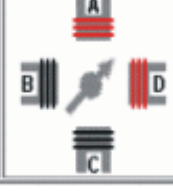
PASO	Bobina A	Bobina B	Bobina C	Bobina D	
1	ON	ON	OFF	OFF	
2	OFF	ON	ON	OFF	
3	OFF	OFF	ON	ON	
4	ON	OFF	OFF	ON	

Figura 5.6: Secuencia normal de un motor a pasos.[24]

Secuencia del tipo wave drive

En esta secuencia se activa solo una bobina a la vez. En algunos motores esto brinda un funcionamiento mas suave. La contrapartida es que al estar solo una bobina activada, el torque de paso y retención es menor.


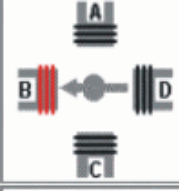
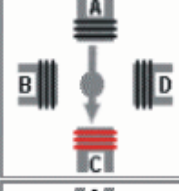
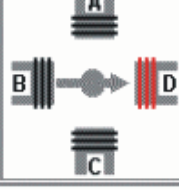
PASO	Bobina A	Bobina B	Bobina C	Bobina D	
1	ON	OFF	OFF	OFF	
2	OFF	ON	OFF	OFF	
3	OFF	OFF	ON	OFF	
4	OFF	OFF	OFF	ON	

Figura 5.7: Secuencia wave drive.[24]

Secuencia del tipo medio paso

En esta secuencia se activan las bobinas de tal forma de brindar un movimiento igual a la mitad del paso real. Para ello se activan primero 2 bobinas y luego solo 1 y así sucesivamente. Como vemos en la tabla la secuencia completa consta de 8 movimientos en lugar de 4.

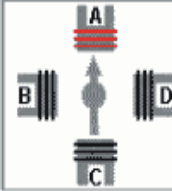
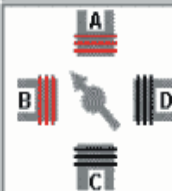
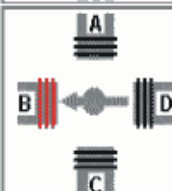
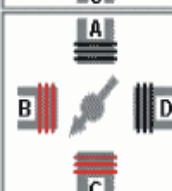
PASO	Bobina A	Bobina B	Bobina C	Bobina D	
1	ON	OFF	OFF	OFF	
2	ON	ON	OFF	OFF	
3	OFF	ON	OFF	OFF	
4	OFF	ON	ON	OFF	

Figura 5.8: Secuencia de medio paso.[24]

5.2. Puentes H

En el circuito vemos un Puente H de transistores, nombre que surge, obviamente, de la posición de los transistores, en una distribución que recuerda la letra H. Esta configuración es una de las más utilizadas en el control de motores de CC, cuando es necesario que se pueda invertir el sentido de giro del motor[36].

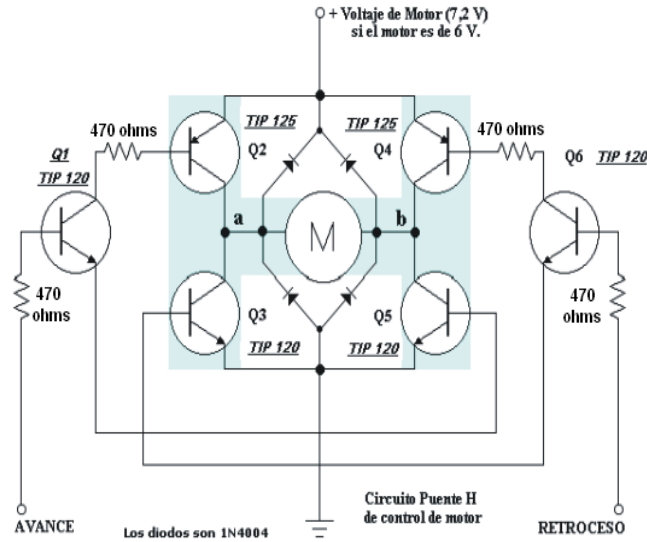


Figura 5.9: Puente H formado por transistores PNP y NPN.[36]

Dependiendo del pulso de la activación en el primer transistor opuesto se activará el motor de CC en un sentido al momento se queda activado en un solo sentido y se tiene que dar un pulso en el sentido opuesto para que gire en el sentido opuesto.

5.2.1. Funcionamiento

Aplicando una señal positiva en la entrada marcada *AVANCE* se hace conducir al transistor *Q1*. La corriente de *Q1* circula por las bases, de *Q2* y *Q5*, haciendo que el Terminal *a* del motor reciba un positivo y el Terminal *b* el negativo (tierra).

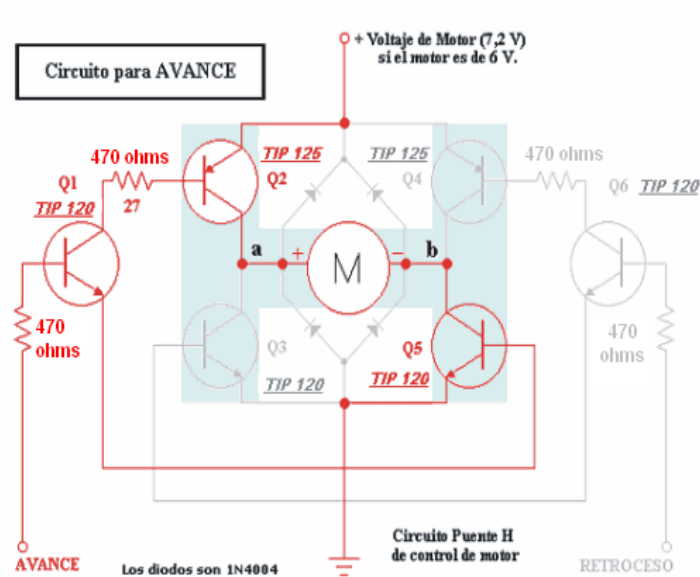


Figura 5.10: Circuito para avance del motor a pasos.[36]

Si en cambio se aplica señal en la entrada RETROCESO, se hace conducir al transistor Q6, que cierra su corriente por las bases, el positivo al terminal b del motor y el negativo (tierra) al terminal a del motor.

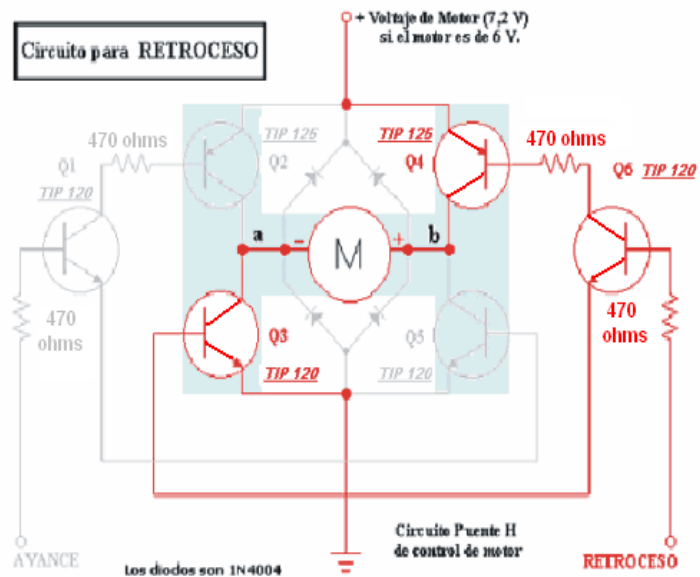


Figura 5.11: Circuito para retroceso del motor a pasos.[36]

Una de las cosas muy importantes que se deben tener en cuenta en el control de

este circuito es que las señales AVANCE y RETROCESO jamás deben coincidir. Si esto ocurre los transistores, $Q2$, $Q3$, $Q4$ y $Q5$ cerrarán circuito directamente entre el positivo de la fuente de alimentación y tierra, sin pasar por el motor, de modo que es seguro que se excederá la capacidad de corriente Emisor-Colector y los transistores, se dañarán para siempre. Y si la fuente no posee protección, también podrá sufrir importantes daños. Al efecto existen varias formas de asegurarse de esto, utilizando circuitos que impiden esta situación (llamados de interlock), generalmente digitales, basados en compuertas lógicas. Abajo mostramos un ejemplo en la figura 5.12.

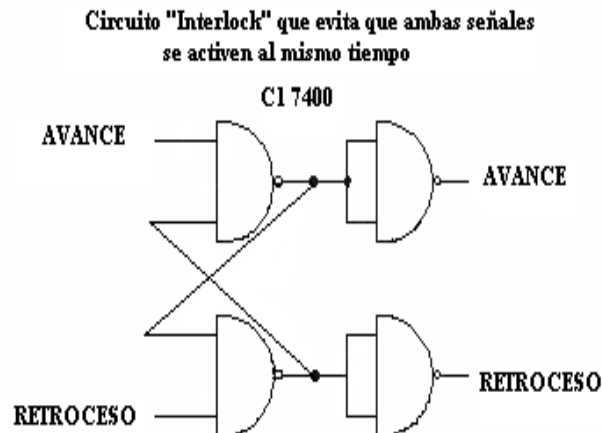


Figura 5.12: Circuito interlock que evita daños en los transistores.[36]

He aquí otra opción de Puente H y circuito de interlock, con la ventaja de que utiliza menos transistores, (tipo Darlington en este caso) y de tener un circuito de interlock aún más seguro. En el circuito anterior, si se presentan las dos señales activas simultáneamente se habilita uno de los sentidos de marcha, sin que se pueda prever cuál será. Si las señales llegan con una leve diferencia de tiempo, se habilita la orden que ha llegado primero, pero si ambas señales llegan al mismo tiempo no se puede prever cuál comando (AVANCE o RETROCESO) será habilitado. En este segundo circuito no se habilita ninguno:

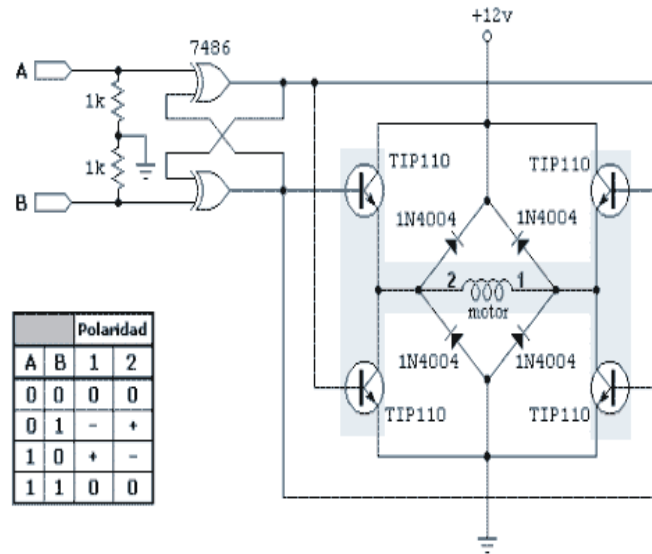


Figura 5.13: Circuito con diodos.[36]

5.2.2. Transistores

El transistor se puede considerar como la unión de dos diodos, por lo que debe de tener dos uniones PN ó NP.

Un transistor tiene por tanto tres zonas de dopado, en un transistor **PNP**, por ejemplo, existen tres zonas de dopado, diferenciadas entre sí y con diferentes cualidades. En un transistor PNP, la base es la zona N y las otras dos zonas P se denominan colector y emisor y viceversa. El emisor y la base de un transistor es como si fueran un diodo (una unión **PN**) y la base y el colector forman la otra unión **PN**. A efectos prácticos, un transistor no funciona como la unión de dos diodos.

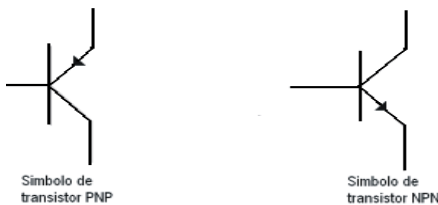


Figura 5.14: Símbolos de transistores PNP y NPN

Funcionamiento del transistor

En el transistor, el emisor es el encargado de inyectar electrones en la base, la cual se encarga de gobernar dichos electrones y mandarlos finalmente al colector[37].

La fabricación del transistor se realiza de forma que la base es la zona más pequeña, después el emisor, siendo el colector el más grande en tamaño.

Polarización del transistor

Un transistor cuenta con dos uniones **PN**, por lo que necesita ser polarizado correctamente. La unión emisor debe estar polarizada directamente y la unión colector debe de estar polarizada inversamente.

Por ejemplo, en un transistor **NPN**, dispondremos de dos baterías, una tendrá conectado a su polo positivo el colector **N** del transistor y la otra tendrá conectado a su polo negativo el emisor **N** del transistor, quedando así polarizado el transistor, circulando así una corriente del emisor a la base y de esta al colector, también circula una pequeña Intensidad de base, la cual es muy pequeña comparada con la intensidad de colector, que se puede tomar en la práctica casi idéntica a la intensidad de emisor, aunque la intensidad de emisor sea igual a la intensidad de colector más la intensidad de base.

$$I_E = I_C + I_B \quad (5.2)$$

Parámetros fundamentales de un transistor.

Para una pequeña tensión de entrada, se obtiene una elevada tensión de salida.

Para una pequeña corriente de entrada, se obtiene una elevada corriente de salida.

Para una pequeña potencia de entrada, se obtiene una elevada potencia de salida.

Según esto, se obtienen tres parámetros de un transistor:

- Ganancia de corriente (A_i)
- Ganancia de tensión (A_v)
- Ganancia de potencia (A_p)

Recta de carga de un transistor

Esta es la recta que refleja todos los posibles puntos de funcionamiento que pueden darse para unos valores determinados de R_c y tensión de alimentación. Esta recta se debe de hacer sobre las curvas características de un transistor, tomando como un punto **P1** una intensidad de colector que debe de ser por debajo de la mínima y como un punto **P2**, una tensión de alimentación.

Si nos situamos en un punto alto de esta recta, estamos cerca de la intensidad máxima de colector, lo cual crea un situación de saturación.

Si nos situamos en un punto demasiado bajo de esta recta, estamos más cerca de la V_{ce} máxima, con la cual se crea un situación de Corte.

Llamamos Punto de trabajo de un transistor a un punto de la recta de carga que determine el valor de la tensión colector-emisor y de las corrientes de colector y de base.

5.3. Optoacopladores

Un optoacoplador combina un dispositivo semiconductor formado por un fotoemisor, un fotoreceptor y entre ambos hay un camino por donde se transmite la luz. Todos estos elementos se encuentran dentro de un encapsulado que por lo general es del tipo DIP[38].

5.3.1. Funcionamiento

La señal de entrada es aplicada al fotoemisor y la salida es tomada del fotoreceptor. Los optoacopladores son capaces de convertir una señal eléctrica en una señal luminosa modulada y volver a convertirla en una señal eléctrica. La gran ventaja de un optoacoplador reside en el aislamiento eléctrico que puede establecerse entre los circuitos de entrada y salida.

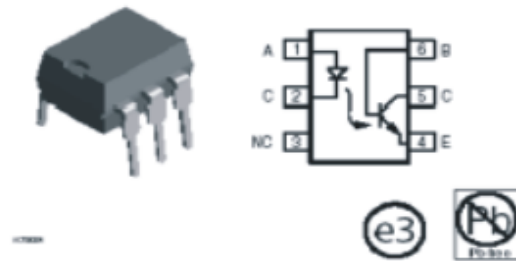


Figura 5.15: Esquema de un optoacoplador

Los fotoemisores que se emplean en los optoacopladores de potencia son diodos que emiten rayos infrarrojos (IRED) y los fotoreceptores pueden ser tiristores o transistores.

Cuando aparece una tensión sobre los terminales del diodo IRED, este emite un haz de rayos infrarrojo que transmite a través de una pequeña guía-ondas de plástico o cristal hacia el fotorreceptor. La energía luminosa que incide sobre el fotorreceptor

hace que este genere una tensión eléctrica a su salida. Este responde a las señales de entrada, que podrían ser pulsos de tensión.

5.4. Conclusiones

En este capítulo se analizó cada uno de los componentes electrónicos y eléctrico que hacen que la estructura mecánica tenga movimiento. es necesario saber el funcionamiento de cada uno de ellos para saber como es que va a suministrar la energía para poder mover a los actuadores, que en este caso son los motores a pasos.

En este se analiza cada una de las secuencias que se le puede dar a un motor a pasos y ver cual es la ideal dependiendo a los requerimientos de la aplicación del robot. Es necesario saber que cuando nosotros aplicamos el tren de pulsos correctos al motor, es decir según sean la disposición de bobinas y el tiempo necesario entre cada pulso nuestros motores funcionaran de manera correcta.

En los puentes H, es necesario utilizar diodo para la protección de regresiones de corriente ya que como se manejan pulsos, si se mandan dos pulsos para activar el motor en ambos sentidos se llega a quemar el circuito y la fuente de suministro de voltaje.

Capítulo 6

Programación y control

En este capítulo se abarcará la parte de la programación del robot así como su interfaz de control, que en este caso fue a través del puerto paralelo de la PC. El control que se hace es siguiendo el principio de la creación de comandos *Gerber* y teniendo una programación a través del lenguaje VisualBasic.

6.1. Visual Basic

Visual Basic es un lenguaje de programación desarrollado por Alan Cooper para Microsoft. El lenguaje de programación es un dialecto de BASIC, con importantes añadidos. Su primera versión fue presentada en 1991 con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitara la creación de interfaces gráficas y en cierta medida también la programación misma. Visual Basic fue discontinuado por Microsoft hace ya varios años. Muchos programadores están migrando a Real Basic o hacia Delphi que es un entorno de desarrollo más avanzado y mucho más actual, (última edición en 2007 y con planes anunciados hasta el 2009). Microsoft propone abandonar el desarrollo en Win32 y pasar al Visual Basic.NET que presenta serias incompatibilidades con el código Visual Basic existente[39].

Es un lenguaje de fácil aprendizaje pensado tanto para programadores principiantes como expertos, guiado por eventos, y centrado en un motor de formularios que facilita el rápido desarrollo de aplicaciones gráficas. Su sintaxis, derivada del antiguo BASIC, ha sido ampliada con el tiempo al agregarse las características típicas de los lenguajes estructurados modernos. Se ha agregado una implementación limitada de la programación orientada a objetos (los propios formularios y controles son objetos), aunque sí admite el polimorfismo mediante el uso de los Interfaces, no admite la herencia. No requiere de manejo de punteros y posee un manejo muy sencillo de cadenas de caracteres. Posee varias bibliotecas para manejo de bases de datos, pudiendo conectar con cualquier base de datos a través de ODBC (Informix, DBase, Access, MySQL, SQL Server, PostgreSQL ,etc) a través de ADO.

6.1.1. Derivados

- Su derivado VBScript es el lenguaje predeterminado para Active Server Pages (ASP).
- Además, una extensión propia del lenguaje llamada Visual Basic for Applications (VBA) permite codificar módulos (a veces llamados macros) para las aplicaciones de Microsoft Office.
- Especialmente a partir de la versión 6 del lenguaje, se permite la interacción y generación de objetos remotos que puedan ser invocados desde páginas de scripts (concretamente las ASP, aunque no es imposible hacer un enlace empleando JSP), alojándolos en servidores de web.
- Visual Basic.NET, parte de .NET, es un lenguaje prácticamente equivalente en funcionalidades a C (por ejemplo, no admite pseudo-punteros), añadiendo capacidades de POO como herencia, polimorfismo.

6.1.2. Ventajas

- La facilidad del lenguaje permite crear aplicaciones para Windows en muy poco tiempo. En otras palabras, permite un desarrollo eficaz y menor inversión en tiempo que con otros lenguajes.
- Permite generar librerías dinámicas (DLL) ActiveX de forma nativa y Win32 (no ActiveX, sin interfaz COM) mediante una reconfiguración de su enlazador en el proceso de compilación.
- Permite la utilización de formularios (Forms) tanto a partir de recursos (como en otros lenguajes) como utilizando un IDE para diseñarlos.
- Posibilidad de desarrollar y ejecutar aplicaciones de Visual Basic 6.0 en Windows Vista sin realizar cambios en la mayoría de los casos pero no se logra aprovechar al máximo las características de este sistema como permite hacerlo Visual Basic 2005 o el próximo Visual Basic 9 de Visual Studio Orcas (Noviembre 2007).

6.1.3. Inconvenientes

En este tipo de lenguaje tiene los siguientes inconvenientes:

- En Visual Basic 6.0 y anteriores sólo existe un compilador e IDE, llamado igual que el lenguaje.
- No existe forma alguna de exportar el código a otras plataformas fuera de Windows.

- Los ejecutables generados son relativamente lentos en Visual Basic 6.0 y anteriores al ser código pseudo-interpretado.
- Por defecto permite la programación sin declaración de variables. (que puede ser sencillamente corregida escribiendo la frase `Option Explicit` en el encabezado de cada formulario, en cuyo caso será menester declarar todas las variables a utilizar, lo que a la postre genera código más puro).
- Su sintaxis no es *case sensitive* (no distingue entre minúsculas y mayúsculas como C++ por ejemplo)
- NO permite programación a bajo nivel ni incrustar secciones de código en ASM.
- Sólo permite el uso de funciones de librerías dinámicas (DLL) `stdcall`.
- Algunas funciones están indocumentadas.
- La escasa implementación de POO en Visual Basic 6.0 y anteriores no permite sacar el máximo provecho de este modelo de programación.
- NO permite el manejo de memoria dinámica, punteros, etc. como parte del lenguaje.

Estos son solo algunas de las desventajas que se tiene con este lenguaje de alto nivel.

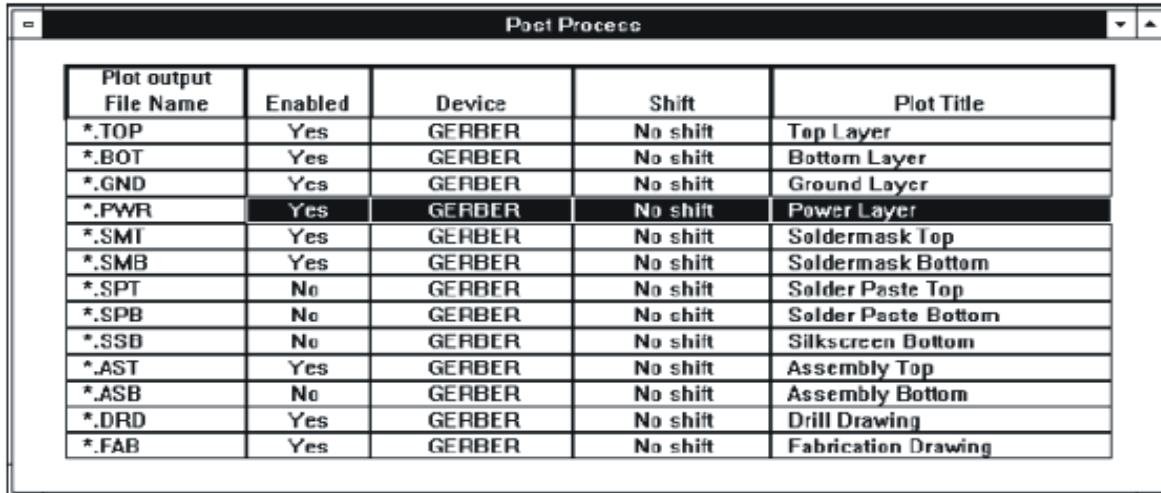
6.2. Ficheros GERBER y diseño de las PCB's

Los ficheros tipo GERBER son utilizados en el diseño de las PCB's. Fichero de fabricación (GERBER). Contiene toda la información necesaria para que se puedan fabricar los PCB's en la industria. Este es un fichero para las máquinas. (Sería algo similar a lo que es el formato *Postscript* para las impresoras)[40].

Una vez finalizado el diseño de la PCB, se entra en un proceso denominado *Post Processing*. *Post procesing* es el proceso por el cual se crean archivos o documentos los cuales sean comprensibles tanto para las máquinas como para el hombre, y se puedan utilizar en la fabricación del circuito impreso. Estos tipos de documentos se pueden crear en tipos de formatos GERBER, formatos DXF y otros tipos de formatos de ficheros.

Layout dispone de la base de datos *Post Process* que nos permitirá añadir o eliminar archivos a la salida que se crearan.

Para abrir esta base de datos, se da clic en el botón *Post Processign* de la barra de herramientas y de la lista desplegada elegir el comando *Setup Batch*.



Plot output File Name	Enabled	Device	Shift	Plot Title
*.TOP	Yes	GERBER	No shift	Top Layer
*.BOT	Yes	GERBER	No shift	Bottom Layer
*.GND	Yes	GERBER	No shift	Ground Layer
*.PWR	Yes	GERBER	No shift	Power Layer
*.SMT	Yes	GERBER	No shift	Soldermask Top
*.SMB	Yes	GERBER	No shift	Soldermask Bottom
*.SPT	No	GERBER	No shift	Solder Paste Top
*.SPB	No	GERBER	No shift	Solder Paste Bottom
*.SSB	No	GERBER	No shift	Silkscreen Bottom
*.AST	Yes	GERBER	No shift	Assembly Top
*.ASB	No	GERBER	No shift	Assembly Bottom
*.DRD	Yes	GERBER	No shift	Drill Drawing
*.FAB	Yes	GERBER	No shift	Fabrication Drawing

Figura 6.1: Base de datos Post Process.[40]

La base de datos nos informa que archivos serán creados, de los nombres de los archivos de salida y del formato en que serán creados estos.

Antes de procesar el diseño para obtener los distintos archivos de salida, se recomienda visualizar todas las capas para asegurarse que todos los elementos necesarios están presentes y visibles en el film que se envía al fabricante de la placa. Si un objeto es visible en la pantalla en modo previsualización (*preview*), este aparecerá en la salida Gerber o DXF. Si un objeto es invisible en la pantalla, no aparecerá en la salida.

Para visualizar una capa efectuar lo siguiente:

1. Hacer click en el botón Post Proc. de la paleta y seleccionar el comando Setup Batch de la lista desplegable que aparece.
2. Elegir el comando *Tile* del menú Window de forma que se puedan ver al mismo tiempo la base de datos Post Process y la ventana de diseño (*design window*).
3. En la base de datos Post Process, seleccionar la capa que se quiere visualizar haciendo clic en la celda correspondiente dentro de la columna *Plot output File Name*.
4. Del menú pop-up, elegir el comando *Preview*. Layout representará automáticamente la capa en la ventana de diseño.

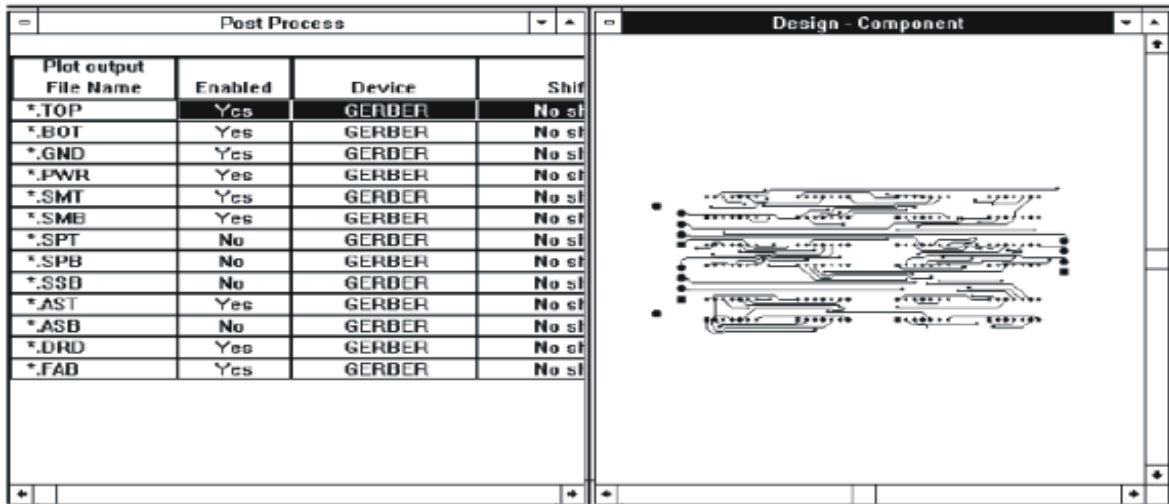


Figura 6.2: visualización de la cara de componentes (TOP).[40]

Supongamos que queremos que el borde de la placa aparezca en la visualización de la cara Top. Para ello haremos lo siguiente:

1. Hacer click en el botón *Initialize Color* de la barra de herramientas para abrir la base de datos *Color (Color spreadsheet)*.
2. En la base de datos *Color*, seleccionar la celda correspondiente al borde de la placa (*board outline*).
3. Del menú pop-up de la base de datos *Color*, elegir el comando *Invisible*.
4. En la base de datos *Post Process*, elegir el comando *Save Colors* del menú pop-up para guardar esta configuración y elegir el comando *Preview* del menú pop-up para volver a dibujar la pantalla. El borde será ahora visible en la capa.

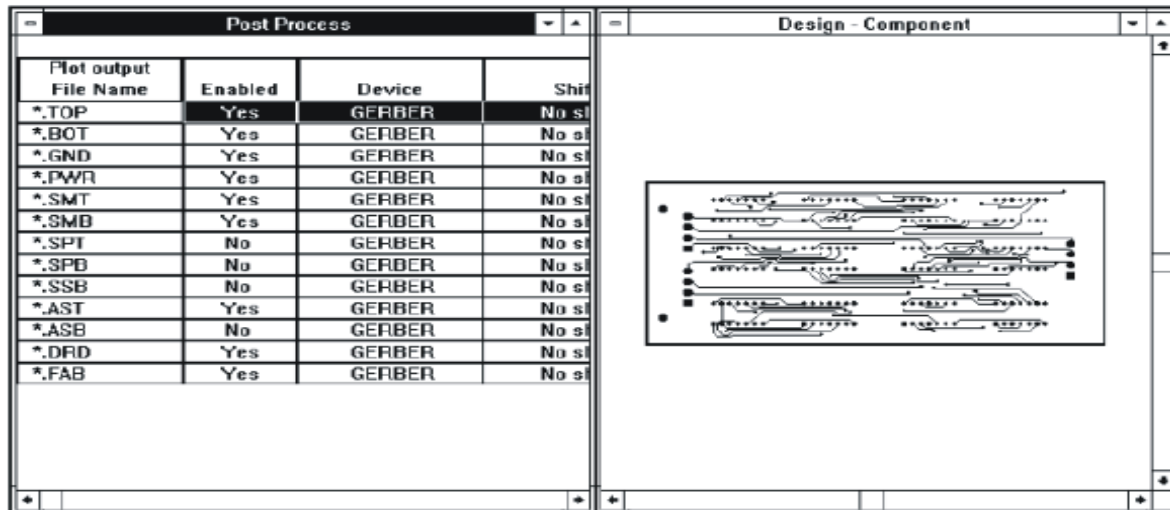


Figura 6.3: Visualización de la cara TOP después de hacer visible el contorno de la placa.[40]

6.2.1. Configurando los archivos de salida

Sirviéndonos de la base de datos Post Process, podremos añadir archivos a la salida o modificar el formato de alguno de estos. En nuestro diseño, crearemos en formato Gerber los archivos habilitados en la base de datos Post Process.

Plot output File Name	Enabled	Device	Shift	Plot Title
*.TOP	Yes	GERBER	No shift	Top Layer
*.BOT	Yes	GERBER	No shift	Bottom Layer
*.GND	Yes	GERBER	No shift	Ground Layer
*.PWR	Yes	GERBER	No shift	Power Layer
*.SMT	Yes	GERBER	No shift	Soldermask Top
*.SMB	Yes	GERBER	No shift	Soldermask Bottom
*.SST	Yes	GERBER	No shift	Silkscreen Top
*.AST	Yes	GERBER	No shift	Assembly Top
*.DRD	Yes	GERBER	No shift	Drill Drawing

Figura 6.4: Archivos de salida.[40]

Utilizando el cuadro de diálogo *Post Process Setup* que se despliega al elegir el comando *Modificar* del menú pop-up podríamos modificar las siguientes características

del ploteado:

- Desplazar el ploteado con respecto al datum de la placa.
- Rotar el ploteado.
- Escalar el ploteado.
- Reflejar el ploteado.

Para crear los archivos de salida especificados en la base de datos Post Process, elegir el comando *Run Batch* del menú que se despliega al hacer click sobre el botón *Post Proc.* de la barra de herramientas.

Además Layout crea dos ficheros adicionales: nombreplaca.**APP** (fichero de aperturas) y nombreplaca.**GTD** (fichero de diseño Gerber).

A continuación se muestra la apariencia de algunos de los archivos creados por la herramienta *Run Batch* y que se enviarán al fabricante.

Aparte de los archivos de salida antes mencionados, debemos crear el archivo de taladros en formato Excellon, que Layout sitúa en el directorio de trabajo. Durante el proceso de fabricación, este será el archivo utilizado por la máquina de taladrado para determinar el tamaño y la posición de los agujeros en la placa.

El archivo creado por Layout tiene por nombre **THRUHOLE.TAP** y toma las coordenadas que aparecen en la ventana de diseño, a no ser que hallamos desplazado la salida en el cuadro de diálogo *Post Process Setup*. Todo lo referente a este tema lo podemos ver en la bibliografía[40].

En las siguientes figuras de 6.5 - 6.12, se muestran las diferentes etapas por las que pasa la placa de circuito impreso.

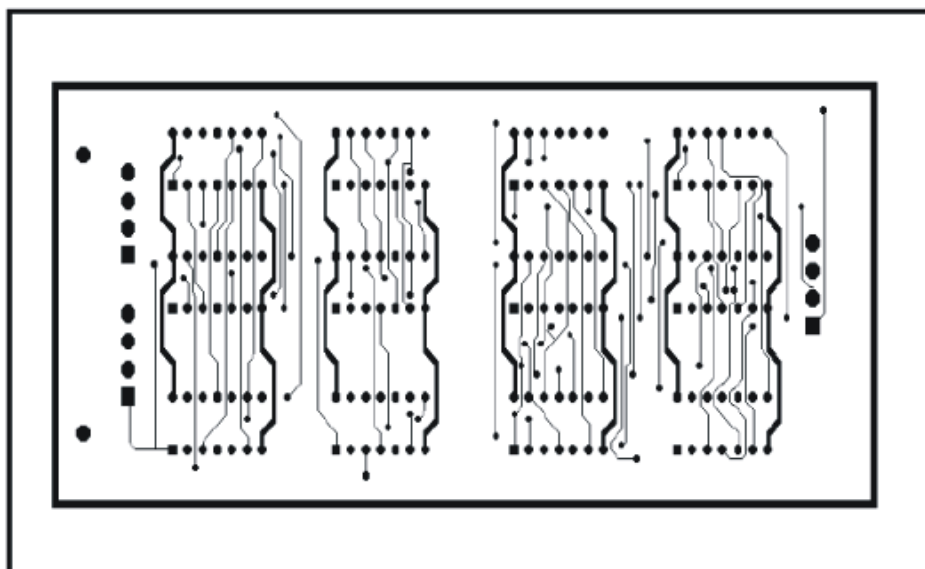


Figura 6.5: Cara de componentes (TOP)[40].

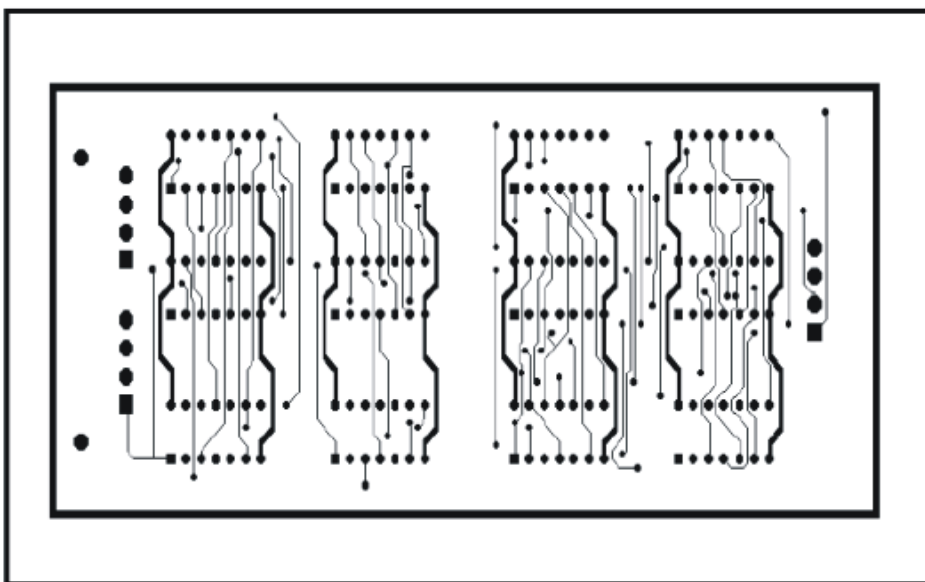


Figura 6.6: Cara de soldadura (BOTTOM)[40]

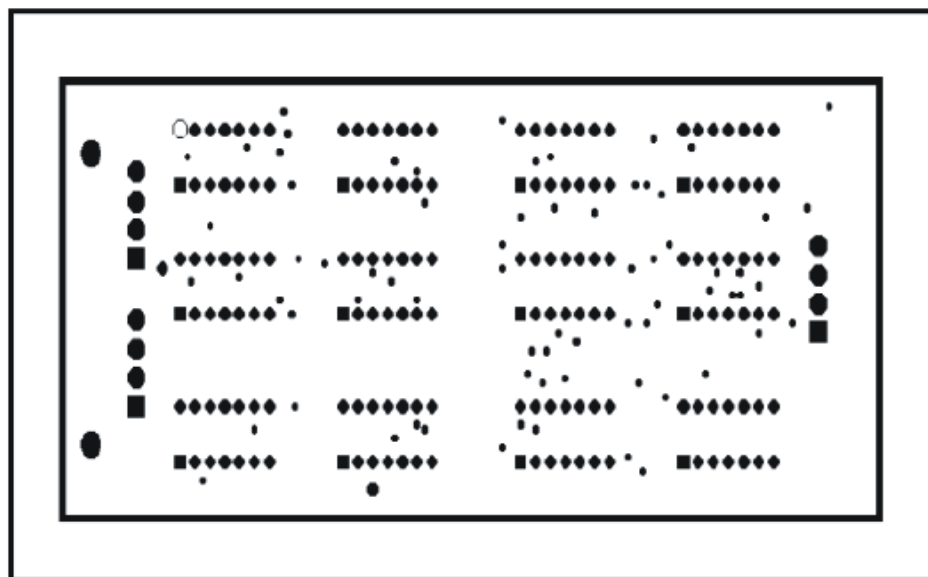


Figura 6.7: Plano GND[40]

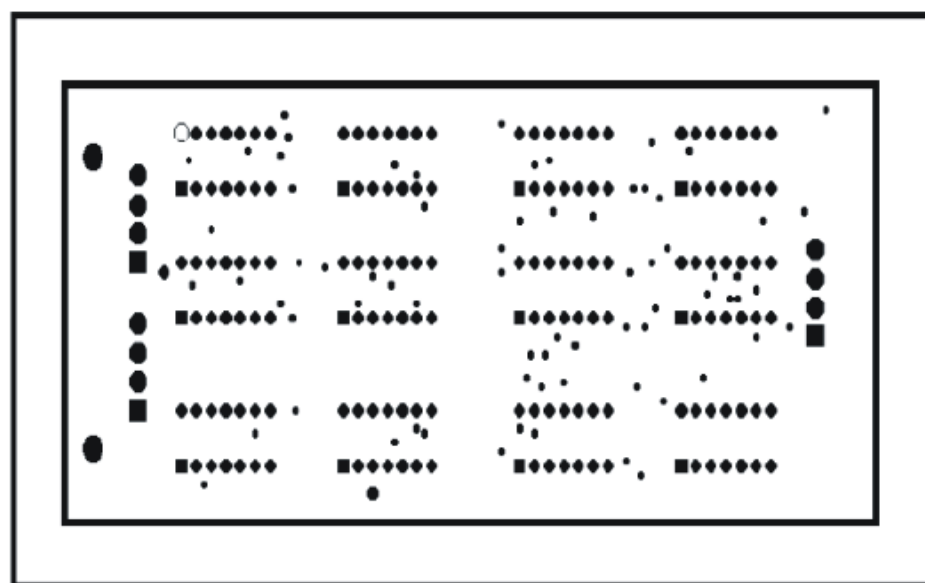


Figura 6.8: Plano VCC[40]

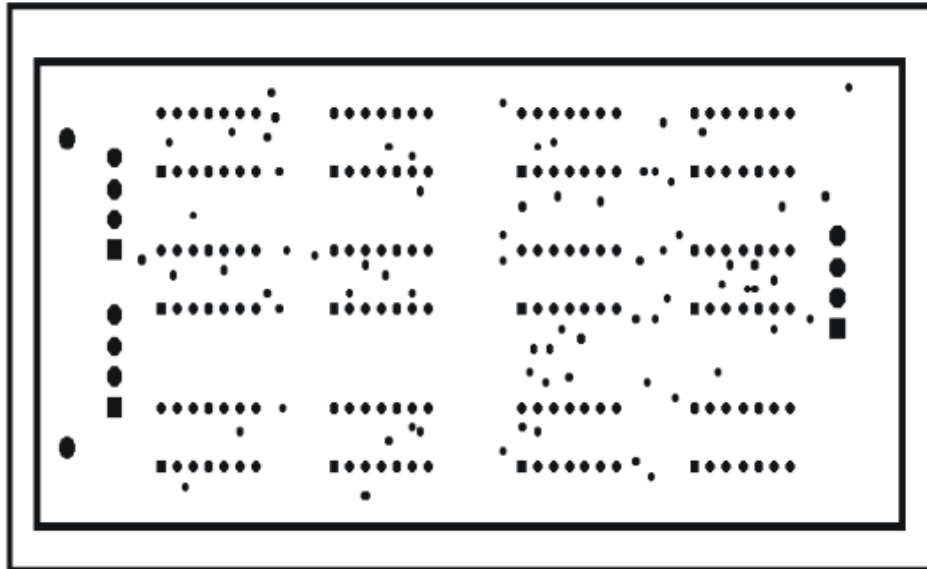


Figura 6.9: Cara de la reserva de soldadura[40]

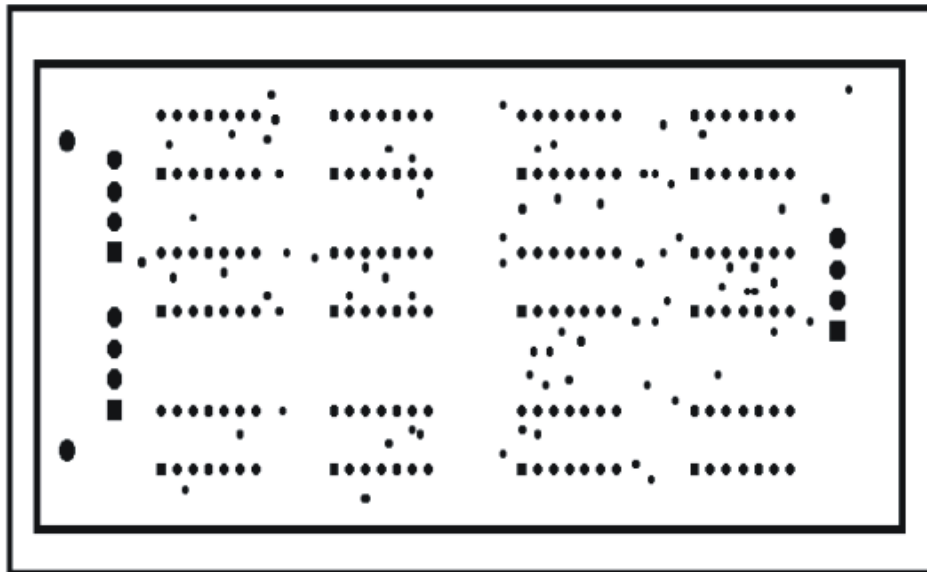


Figura 6.10: Plano de serigrafía(cara de componentes)[40]

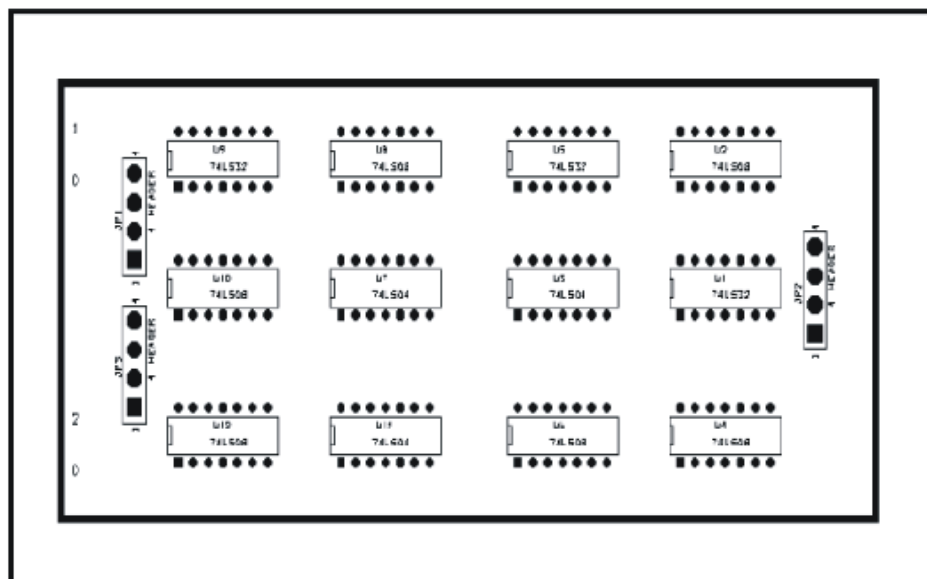


Figura 6.11: Plano de ensamblaje (cara de componentes)[40]

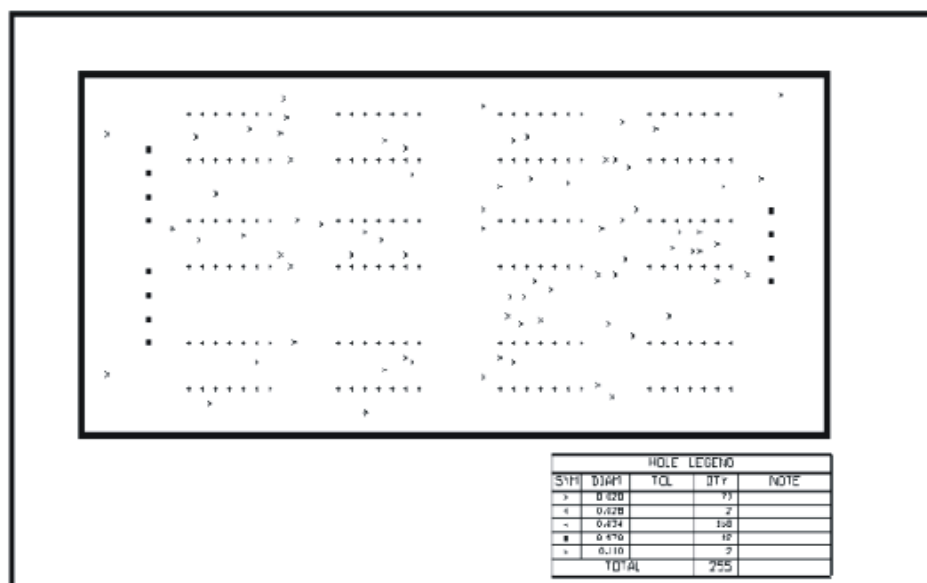


Figura 6.12: Plano de taladro)[40]

6.2.2. Documentación

Para documentar el diseño se pueden generar una gran variedad de informes que incluyen listas de partes, listas de componentes, de conexiones, informes de cambio de nombres, listas de taladros comprensibles para el usuario, etc.

Para especificar que informes se quieren crear, se realiza lo siguiente:

1. Hacer click en el botón *Post Proc.* de la barra de herramientas.
2. Del menú que se despliega, seleccionar el comando *Reports*.
3. En el cuadro de diálogo *Generate Reports* seleccionar los informes que se quieran generar haciendo clic sobre ellos.

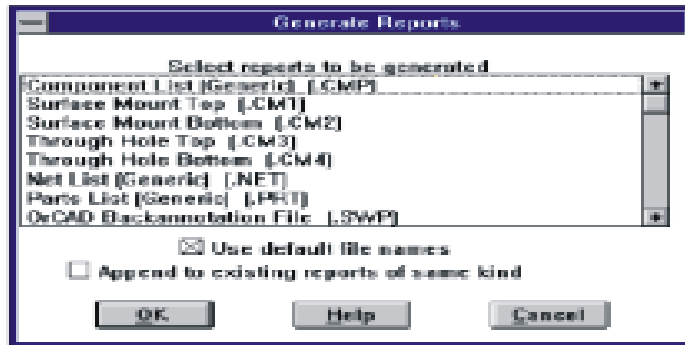


Figura 6.13: Generación de informes que documenten el diseño.[40]

Como ejemplo se incluye parte del archivo de taladros (drill list (.DRL)), creado por Layout.

```

*****
*
* Component List (Generic)
*
* C:\ORCADWIN\LAYOUT\SAMPLES\CURSO99\PRACTICA_SE\DOCUMENTA\TUTORIAL_2.MAX *
* Wed Dec 15 10:54:23 1999
*
*****

```

REF DES	PART NAME	FOOTPRINT	X LOC	Y LOC	ROTATION
1	0	MTHOLE1_1	100.00 m	100.00 m	0
2	0	MTHOLE1_2	100.00 m	2300.0..	0
3	0	MTHOLE1_3	5400.0..	2300.0..	0
4	0	MTHOLE1_4	5400.0..	100.00 m	0
JP1	4_HEADER	SIP/TM/L_400/4..	200.00 m	300.00 m	90
JP2	4_HEADER	SIP/TM/L_400/4..	200.00 m	1000.0..	90
JP3	4_HEADER	SIP/TM/L_400/4..	200.00 m	1600.0..	90
U1	74LS08	DIP.100/14/W.3..	4100.0..	1700.0..	0
U2	74LS04	DIP.100/14/W.3..	1700.0..	1700.0..	0
U3	74LS32	DIP.100/14/W.3..	500.00 m	1700.0..	0
U4	74LS08	DIP.100/14/W.3..	2900.0..	1700.0..	0
U5	74LS08	DIP.100/14/W.3..	500.00 m	1000.0..	0
U6	74LS04	DIP.100/14/W.3..	4100.0..	300.00 m	0
U7	74LS08	DIP.100/14/W.3..	4100.0..	1000.0..	0
U8	74LS32	DIP.100/14/W.3..	1700.0..	300.00 m	0
U9	74LS08	DIP.100/14/W.3..	1700.0..	1000.0..	0
U10	74LS04	DIP.100/14/W.3..	2900.0..	1000.0..	0
U11	74LS08	DIP.100/14/W.3..	2900.0..	300.00 m	0
U12	74LS32	DIP.100/14/W.3..	500.00 m	300.00 m	0

Figura 6.14: Archivo de taladro donde se muestran las diferentes coordenadas.[40]

6.3. Manejo del puerto paralelo

El manejo del puerto paralelo es importante en el uso de la comunicación de datos ya sea a un dispositivo de red, llámese impresora, escáner, etc. El conector que usa se le conoce como conector DB25 debido a los 25 pines que tiene a su disposición para los registros de datos, estado y de control.

En los conectores DB25 se conocen dos clasificaciones:

1. Conector DB25 macho.
2. Conector DB25 hembra.

Debemos de tener en cuenta la disposición de pines para cada uno de los DB25 porque si llegamos a conectar erróneamente podría darnos problemas en la transmisión y recepción de datos. Cada uno de los registros tiene un byte de disposición en sus pines.

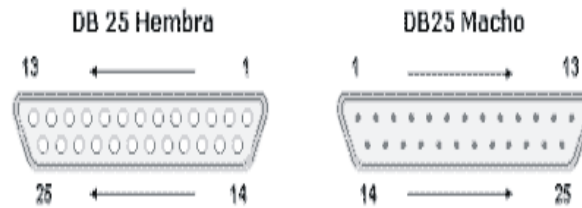


Figura 6.15: Conectores DB25 hembra y macho.[47]

La tensión de trabajo del puerto es de 5 voltios, por lo que necesitamos una fuente estabilizada o regulada de tensión, esto es importante tenerlo en cuenta, ya que estaremos enviando señales al puerto. Por otro lado, si bien puedes utilizar la PC para enviar señales al exterior sin necesidad de una fuente externa, es recomendable utilizarla y así no exiges demasiado al puerto y te evitas problemas. si nosotros mandamos un bit o 5V por un pin estaremos mandando un 1 no dejara de hacerlo hasta que lo pongamos en cero[47].

En este proyecto se esta manejando la disposición de pines de la siguiente manera para cada registro.

- Registro de datos: dispone de los pines del 2 al 9. Es el PORT 888 y es de solo escritura, por este registro enviaremos los datos al exterior de la pc. Se debe tener cuidado al no enviar señales eléctricas al ordenador por estos pines.
- Registro de estado: dispone de los pines 15, 13, 12, 10 y 11. Es el PORT 889 y es de solo lectura, por aquí enviaremos señales eléctricas al ordenador, de este registro solo se utilizan los cinco bits de más peso, que son el bit 7, 6, 5, 4 y 3 teniendo en cuenta que el bit 7 funciona en modo invertido.
- Registro de control (Pin 1, 14, 16 y 17): Es el correspondiente al PORT 890, y es de lectura/escritura, es decir, podremos enviar o recibir señales eléctricas, según nuestras necesidades. De los 8 bits de este registro solo se utilizan los cuatro de menor peso o sea el 0, 1, 2 y 3, con un pequeño detalle, los bits 0, 1, y 3 están invertidos.

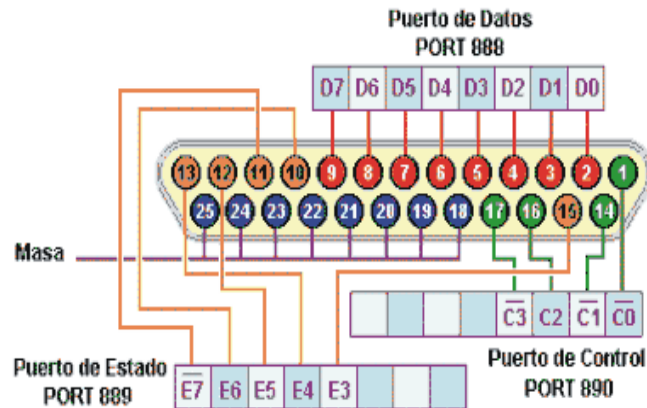


Figura 6.16: Disposición de pines de los tres registros[47].

Para este proyecto cabe mencionar que se están utilizando el registro de datos para dar un pulso de excitación a cada una de las bobinas de los motores a pasos en sus dos sentidos de giro, ya que debemos saber que por cada bobina se le asigna dos pulsos en los dos sentidos de giro. Para el registro estado se desprecia ya que no se están almacenando datos a la computadora. Para el registro de control solo se habilitan dos pines necesarios para activar el motor de c.d. que levanta y baja a l perno/cremallera para que inicie el trazado de líneas.

6.4. Manejo de Visual Basic

Visual es hoy el lenguaje de programación más popular del mundo. Es el sueño del programador de aplicaciones. Es un producto con una interfaz grafica de usuario para crear aplicaciones para Windows basado en el lenguaje Basic, QBasic o QuickBasic, y en la programación orientada a objetos. Cuantas veces trabajando con Windows nos hemos preguntado: cómo haría yo de una forma sencilla una interfaz grafica para mi aplicación, similar a esas ventanas de Windows. Hoy tenemos la respuesta Visual Basic. Todo lo referente a este tema lo podemos ver en[17].

La palabra *Visual* hace referencia al método que se utiliza para crear la interfaz grafica de usuario En lugar de escribir numerosas líneas de código para implementar una interfaz, utilice el ratón arrastrar y colocar los objetos prefabricados al lugar deseado dentro de un formulario.

La palabra *Basic* hace referencia al lenguaje BASIC (Beginners All-Purpose Symbolic Instruction Code), un lenguaje utilizado por más programadores que ningún otro lenguaje en la historia de la informática. Visual Basic ha evolucionado a partir del lenguaje BASIC original y ahora contiene centenares de instrucciones, funciones y pa-

labras clave, muchas de las cuales están directamente relacionadas con la interfaz gráfica de Windows.

6.4.1. Programación orientada a objetos

La programación orientada a objetos (POO) es una forma de programación que utiliza objetos, ligados mediante mensajes, para la solución de problemas. Puede considerarse como una extensión natural de la programación estructurada en un intento de potenciar los conceptos de modularidad y reutilización del código.

Mecanismos básicos de la POO

Los mecanismos básicos de la programación orientada a objetos son: objetos, mensajes, métodos y clases.

Objetos

Un programa tradicional se compone de procedimientos y datos. Un programa orientado a objetos consiste solamente en objetos. Un objeto es una encapsulación genérica de datos y de los procedimientos para manipularlos. Dicho de otra forma, un objeto es una entidad que tiene unos atributos particulares, las propiedades, y unas formas de operar sobre ellos, los métodos. Por lo tanto, un objeto contiene, por una parte, operaciones que definen su comportamiento, y por otra, variables manipuladas por esas operaciones que definen su estado.

Por ejemplo, una ventana del sistema operativo Windows es un objeto. El color de fondo de la ventana, el ancho, el alto, etc. son propiedades. Las rutinas lógicamente transparentes al usuario, que permiten maximizar la ventana, minimizar, etc. son métodos.

Mensajes

Cuando se ejecuta un programa orientado a objetos, los objetos están recibiendo interpretando y respondiendo a mensajes de otros objetos. Esto marca una clara diferencia con respecto a los elementos de datos pasivos de los sistemas tradicionales. Por ejemplo, en Visual Basic un mensaje está asociado con un procedimiento, de tal forma que cuando un objeto recibe un mensaje la respuesta a ese mensaje es ejecutar el procedimiento asociado. Esta función o procedimiento recibe el nombre de método.

Otro ejemplo; cuando un usuario quiere maximizar una ventana Windows, lo que hace simplemente es pulsar el botón de la misma que realiza esa acción. Eso, provoca que Windows envíe un mensaje a la ventana para indicar que tiene que maximizarse. Como respuesta a este mensaje se ejecutará el método programado para ese fin.

Método

Un método se implementa en una clase de objetos y determina como tiene que actuar el objeto cuando recibe un mensaje. En adición, las propiedades permitirán almacenar información para dicho objeto. Un método puede también enviar mensajes a otros objetos solicitando una acción o información.

La estructura más interna de un objeto esta oculta para otros usuarios y la única conexión que tiene con el exterior son los mensajes. Los datos que están dentro de un objeto solamente pueden ser manipulados por los métodos asociados al propio objeto.

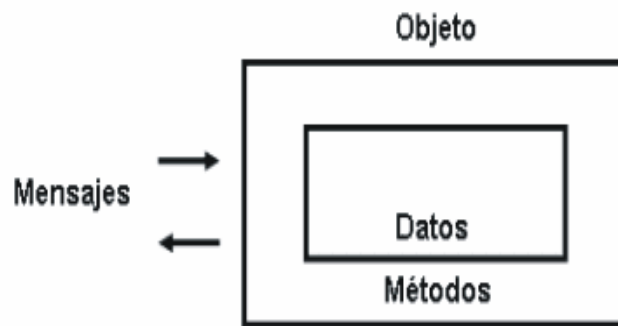


Figura 6.17: Envío y recibimiento de datos manipulados por los métodos del objeto.[17]

La ejecución de un programa orientado a objetos realiza fundamentalmente tres cosas:

1. Crea los objetos necesarios.
2. Los mensajes enviados a unos y a otros objetos dan lugar a que se procese internamente la información.
3. Finalmente, cuando los objetos no son necesarios, son borrado, liberándose la memoria ocupada por los mismos.

Clases

Una clase es un tipo de objetos definidos por el usuario. Una clase equivale a la generalización de un tipo específico de objetos. Por ejemplo, piense en un molde para hacer flanes; el molde es la clase y los flanes los objetos.

Formulario

Es la ventana sobre la que colocaremos los controles de la interfaz grafica que el usuario utilizara para comunicarse con la aplicación.

6.4.2. Módulos de Visual Basic

El código de una aplicación Visual Basic se edita en módulos. Los módulos tienen incorporadas funciones automáticas de formato y comprobación de sintaxis. Hay tres tipos de módulos: de formulario, estándar y de clase.

Módulo de formulario

Es un fichero con extensión `.frm` que puede contener descripciones gráficas de un formulario, sus controles y sus valores de propiedades, declaraciones de constantes, variables y procedimientos externos a nivel del formulario, así como procedimientos conducidos por eventos y procedimientos generales.

Módulo estándar

Es un fichero con extensión `.bas` que solo contiene declaraciones y definiciones de procedimientos, tipos y datos a los que tienen acceso otros módulos de la aplicación.

Módulo de clase

Es un fichero con extensión `.cls` que contiene la definición de una clase de objetos; esto es, la definición de cada una de sus propiedades y métodos.

6.4.3. Ratón

Los formularios y la mayoría de los controles reconocen tres eventos producidos por el ratón:

- **MouseDown**. Reconocido cuando el usuario pulsa cualquier botón.
- **MouseUp**. Reconocido cuando el usuario suelta cualquier botón.
- **MouseMove**. Reconocido cada vez que el usuario mueve el puntero del ratón a una nueva posición.

Un formulario reconoce un evento del ratón cuando el puntero del mismo está en una zona en la que no hay ningún control, y un control reconoce un evento del ratón cuando el puntero del ratón está sobre el propio control.

Cuando el usuario hace clic sobre un objeto y mantiene pulsado el botón del ratón, después de que el objeto haya reconocido el evento `MouseDown`, aunque mueva el puntero fuera del objeto, este continúa reconociendo el evento `MouseMove`, y el evento `MouseUp` cuando suelte el botón del ratón.

Argumentos `X`, `Y`. Los argumentos `X` e `Y` se corresponden con las coordenadas del puntero del ratón referidas al objeto para el que se da el evento.

6.4.4. Matrices

Una matriz es un conjunto de elementos, todos del mismo tipo, que comparten un nombre común, a los que se puede acceder por la posición que ocupa cada uno de ellos dentro de la matriz. Cada elemento es una variable que puede contener un dato numérico o una cadena de caracteres, dependiendo esto del tipo de la matriz.



Figura 6.18: Ejemplo de una matriz que contiene un conjunto de elementos[17]

La representación de las matrices se hace mediante variables suscritas o de subíndices y pueden tener una o varias dimensiones.

6.4.5. Librería de E/S para Visual Basic

Con la llegada de Windows XP llegaron nuevas medidas de seguridad y restricciones, una de ellas es el acceso al puerto paralelo, desde un programa hecho por cualquier usuario[31]. Para poder acceder al puerto paralelo se empleó una librería diseñada para ese fin, dicha librería es IO.dll, la cual contiene una serie de funciones que permiten manipular el puerto con Visual Basic, las cuales son las siguientes:

- **PortOut:** Envía un byte al puerto especificado
- **PortWordOut:** Envía una palabra (de 16 bits) al puerto especificado
- **PortIn:** Envía una palabra doble (de 32 bits) al puerto especificado
- **PortWordIn:** Lee un byte del puerto especificado
- **PortDWordIn:** Lee una palabra (16 bits) del puerto especificado
- **SetPortBit:** Lee un palabra doble (32 bits) del puerto especificado
- **ClrPortBit:** Pone en 1 el bit especificado del puerto indicado
- **NotPortBit:** Pone en 0 el bit especificado del puerto indicado
- **GetPortBit:** Invierte el estado del bit especificado del puerto indicado
- **RightPortShift:** Obtiene el estado del bit especificado

- **LeftPortShift:** Rota a la derecha el contenido del puerto haciendo que LSB pase a MSB
- **IsDriverInstalled:** Rota a la izquierda el contenido del puerto haciendo que MSB pase a LSB
- **IsDriverInstalled:** Devuelve un valor diferente a 0 si el driver se encuentra cargado en memoria

Nota: En la manipulación del Robot cartesiano se utilizaron los siguientes puertos:

1. PortOut
2. SetPortBit
3. ClrPortBit

6.5. Funcionamiento del robot cartesiano

El área Robot cartesiano esta limitada a una superficie de $12\text{cm} \times 14\text{cm}$, que será el área de trabajo para los trazos a realizar.

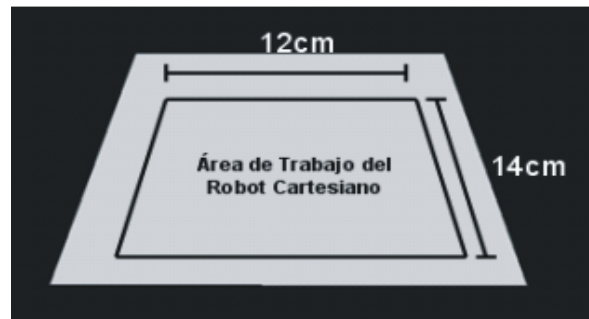


Figura 6.19: Área de trabajo del Robot Cartesiano

6.5.1. Interfaz

La pantalla siguiente es la principal de la interfaz grafica en ella se menciona que es la interfaz para el Robot Cartesiano y el nombre de los autores.



Figura 6.20: Presentación de la interfaz gráfica del Robot Cartesiano

La ventana se cierra después de pasar algunos segundos para posteriormente pasar al área de trabajo a las pantallas del área de trabajo.

Área de trabajo

Al cerrarse la ventana de presentación, se abre la pantalla de área de trabajo, la cual cuenta con las siguientes partes:

1. Área de Dibujo. Área reservada para que el usuario trace una trayectoria irregular arbitraria.
2. Área de Trayectoria del Robot. Área reservada para la trayectoria del Robot.
3. Funciones. Área que contiene funciones como: Reproducir, Borrar y Salir.
4. Reproducir. Función empleada para activar el Robot una vez que el usuario haya trazado el dibujo a reproducir. Cuando el trazo esta hecho se pulsa la opción de

Reproducir y el Robot recibirá los pulsos que activan los motores a través de la interfaz de potencia, al momento que avanza el Robot en el área de *Trayectoria del Robot* se sigue la evolución del dibujo reproducido por el prototipo.

5. Borrar. Función empleada para borrar la trayectoria dibujada. Si el usuario se equivoca al trazar un dibujo puede borrarlo con esta opción de igual manera borrara los dibujos antes trazados.
6. Salir. Cierra la aplicación.



Figura 6.21: Interfaz gráfica del Robot Cartesiano

Diagrama de clases del robot cartesiano

La figura (6.22) muestra el diagrama de clases del robot cartesiano. Dicha figura muestra la estructura y la relación de los objetos de la interfaz grafica, el proceso empieza cuando se quiere crear una trayectoria, se obtienen las coordenadas que serán evaluadas para ver si se encuentran dentro del área de dibujo, posteriormente si las coordenadas son validas, entonces serán guardadas en las matrices X e Y. Una vez creada y guardada la trayectoria, esta será reproducida, el primer paso para reproducirla es obtener las coordenadas de las matrices, evaluar una a una las coordenadas para evaluar el movimiento de los motores (izquierda, derecha, arriba, abajo) y por ultimo llamar al archivo DLL para acceder al puerto y mover el robot.

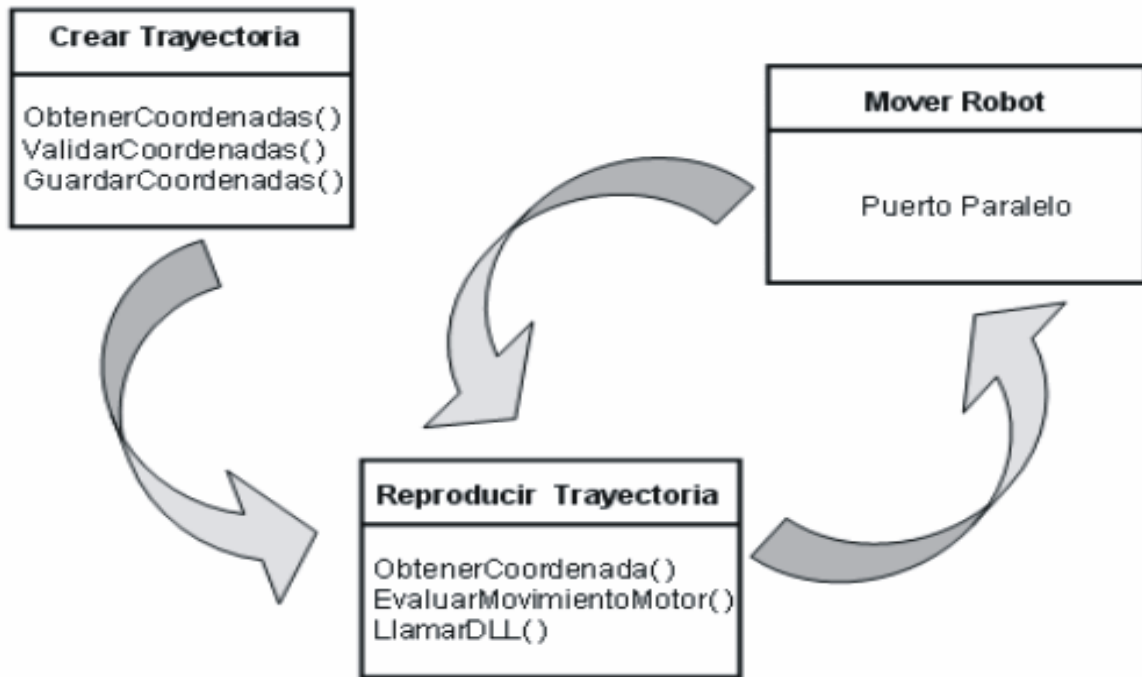


Figura 6.22: Diagrama de Clases del Robot Cartesiano.

Diagramas de procesos del robot cartesiano

La figura (6.23) muestra el diagrama de clases del robot cartesiano. Dicha figura muestra la estructura y la relación de los objetos de la interfaz grafica, el proceso empieza cuando se quiere crear una trayectoria, se obtienen las coordenadas que serán evaluadas para ver si se encuentran dentro del área de dibujo, posteriormente si las coordenadas son validas, entonces serán guardadas en las matrices X e Y. Una vez creada y guardada la trayectoria, esta será reproducida, el primer paso para reproducirla es obtener las coordenadas de las matrices, evaluar una a una las coordenadas para evaluar el movimiento de los motores (izquierda, derecha, arriba, abajo) y por ultimo llamar al archivo DLL para acceder al puerto y mover el robot.

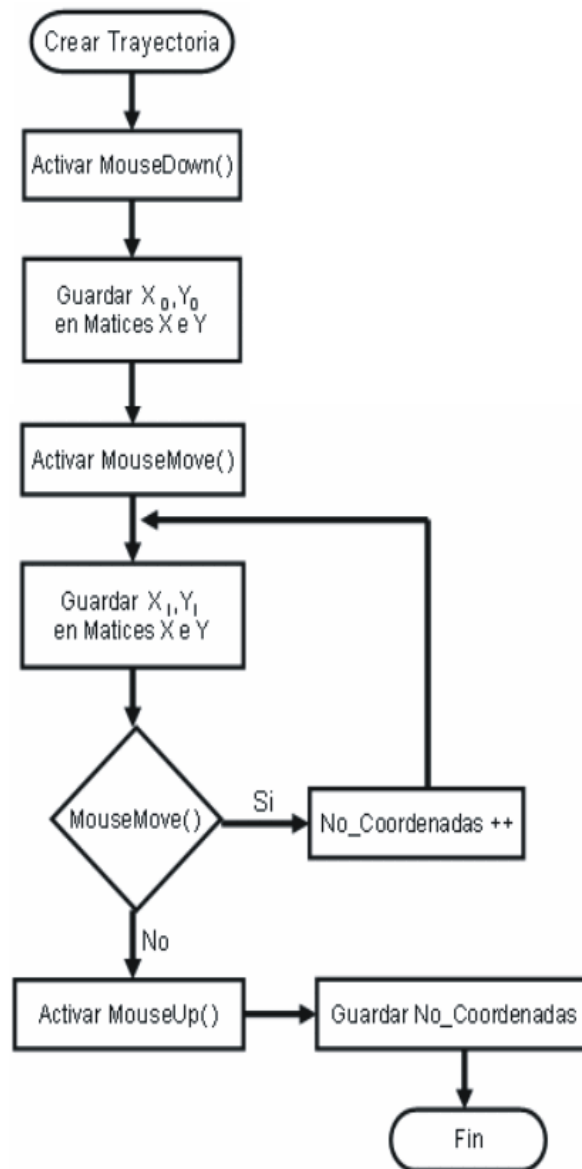


Figura 6.23: Diagrama de procesos para crear la trayectoria.

Creación de la trayectoria

Al hacer clic con el Mouse en el área de Dibujo de la interfaz grafica, se activa el evento del ratón MouseDown, lo que hace que se guarde la posición actual del ratón, en una Matriz la coordenada X y en otra la coordenada Y, dichas matrices son MatX y MatY respectivamente.

Si se mantiene presionado el botón del ratón, y este se mueve a través del área de

dibujo de la interfaz grafica, entonces se activa el evento MouseMove, lo que hace que se vayan guardando todas las coordenadas por donde pasa el Mouse en las matrices para X e Y. Una vez que el usuario deja de pulsar el botón del ratón, se genera el evento MouseUp del ratón lo que finaliza el proceso de guardar las coordenadas del dibujo trazado.

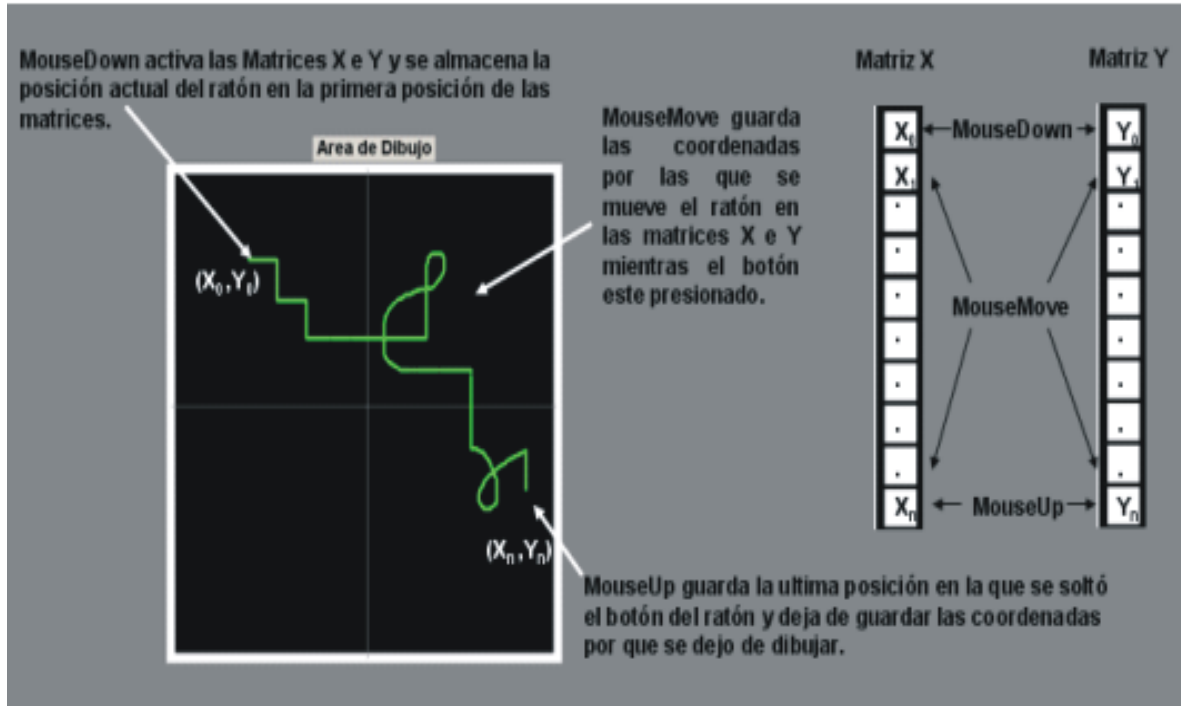


Figura 6.24: Creación de la trayectoria

Reproducción de la trayectoria

La figura (6.25) muestra el diagrama de procesos para reproducir la trayectoria. El proceso empieza al obtener el número de coordenadas de una trayectoria previamente creada y guardada, una vez obtenido el número total de coordenadas se inicializa un contador que tiene como función recuperar cada una de las coordenadas de la trayectoria, esto mediante un bucle for, al recuperar una coordenada se hace una evaluación de la coordenada actual y la coordenada anterior, para determinar hacia donde se debe mover el robot. Al terminarse las coordenadas de las matrices X e Y correspondientes a la trayectoria guardada, se finaliza el trazado del dibujo.

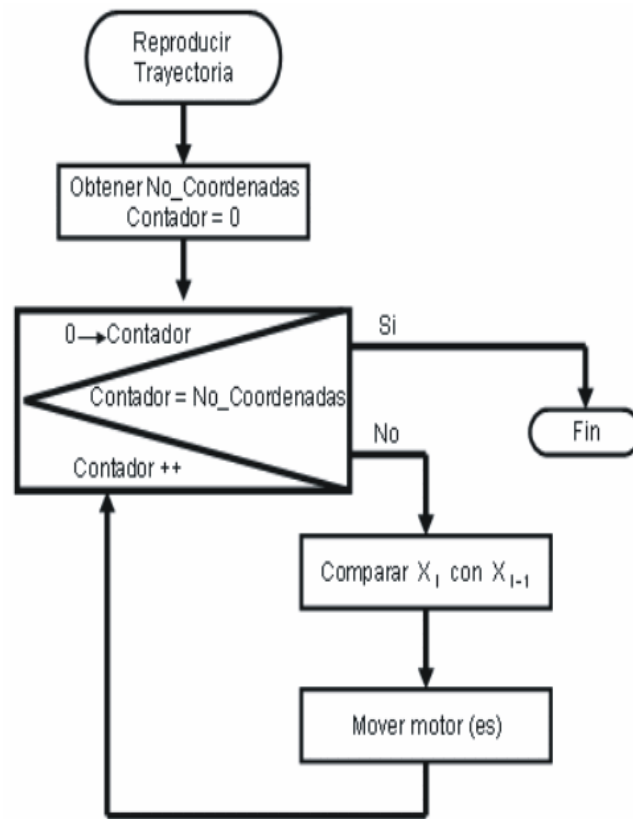


Figura 6.25: Creación de la trayectoria

Reproducir la trayectoria guardada

Para poder mover los motores, a través del puerto paralelo, se empleará la librería *IO.dll* que nos permite dicha acción, se guarda la librería en el mismo directorio del programa de la interfaz gráfica y se agrega un modulo tipo *.bas*, en el cual se hace la declaración de las funciones de acceso al puerto, de igual forma se agrega otro modulo *.bas* para generar una pausa, esto se hace de manera sencilla a través de dos bucles anidados, de tal forma que, el sistema al llamar a la pausa empezará un conteo y esto nos producirá una pausa, ya que se tardará un lapso de 1 segundo al realizar el conteo definido. La pausa generada por los bucles anidados, nos sirve para que haya un lapso de tiempo para activar una a una las bobinas de los motores a pasos y así no perder la secuencia.

Una vez creada la trayectoria y guardadas las coordenadas en las matrices, el siguiente paso será activar el botón Reproducir de la interfaz gráfica. Dicho botón lo que hace es iniciar un contador(i) para ir recuperando las coordenadas de las matrices X e Y, se recupera la primera coordenada (X_0, Y_0) y al recuperar la siguiente coordenada(X_i, Y_i) se toma una decisión, si la siguiente coordenada(X_i, Y_i) es mayor a la anterior(X_{i-1}, Y_{i-1}),

entonces se mueve el lápiz en el eje X a la derecha y en el eje Y hacia arriba, pero si la coordenada siguiente es menor a la anterior, se mueve el lápiz en el eje X hacia la derecha y en el eje Y hacia abajo. También se hace una comparación con la coordenada (X_i, Y_i) y la coordenada (X_{i-1}, Y_{i-1}) , al mismo tiempo X e Y; si ambas coordenadas cambian, ya sea que las coordenadas siguientes sean mayores, menores, X mayor y Y menor o X menor y Y mayor, se activan ambos motores el de X y el de Y, para evitar lo mayor posible que se haga un trazo con esquinas y tratar de que se haga curvo lo mas posible que se pueda.

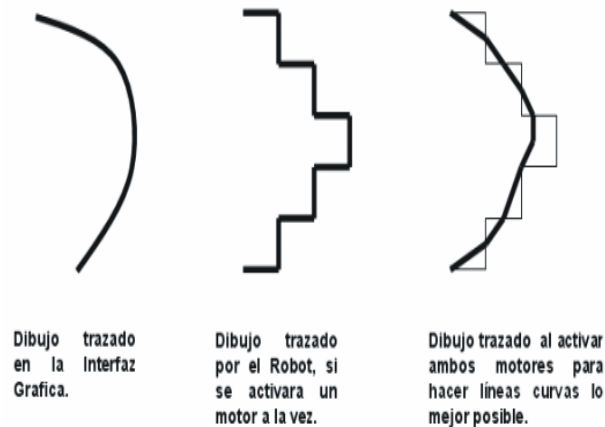


Figura 6.26: Diagrama de procesos para reproducir la trayectoria

En la figura anterior se muestra que el Robot puede trazar una línea curva lo mejor posible si se mueven ambos motores, el motor del eje X y el motor del eje Y, esto al tomar la decisión con la coordenada siguiente y la coordenada anterior para X e Y, como se menciono anteriormente.

Pruebas realizadas con el robot cartesiano

La siguiente imagen es un ejemplo de cómo trabajan juntos la PC y el Robot. En la imagen se muestra una trayectoria con líneas rectas definida por el usuario y posteriormente en la parte derecha, se muestra el avance del robot cartesiano.

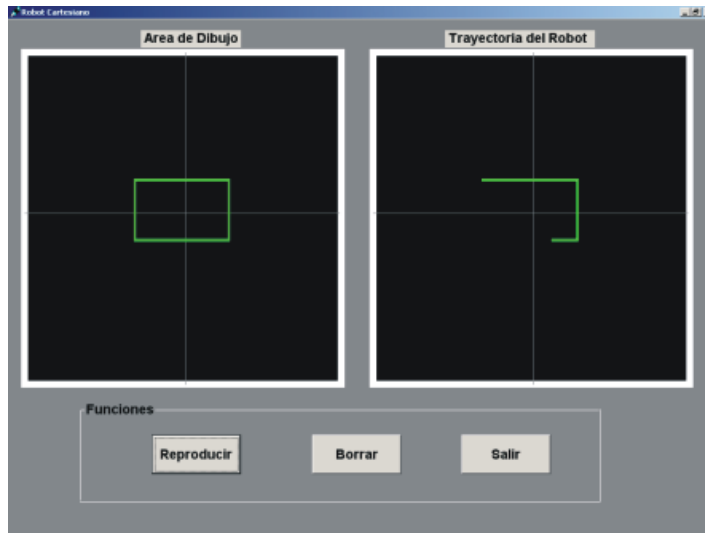


Figura 6.27: Ejemplo de una trayectoria con líneas rectas

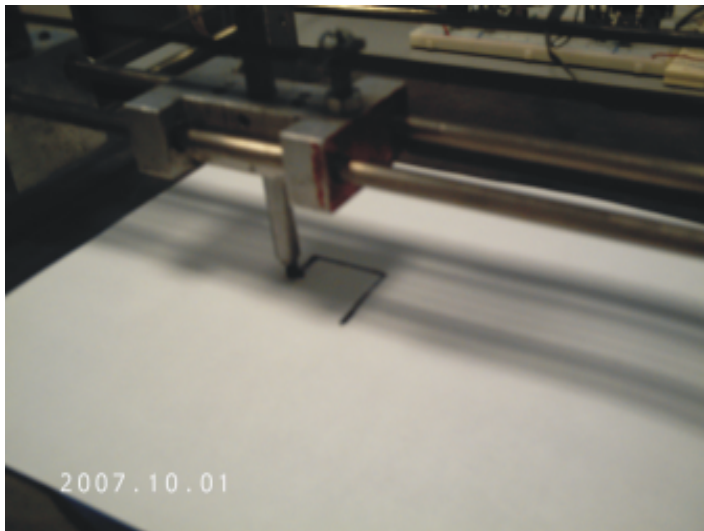


Figura 6.28: Reproducción de trayectoria de líneas rectas por el Robot Cartesiano

En la siguiente imagen se muestra el trazo de una trayectoria mixta, con líneas curvas y rectas, así como la evolución del trazo del Robot.

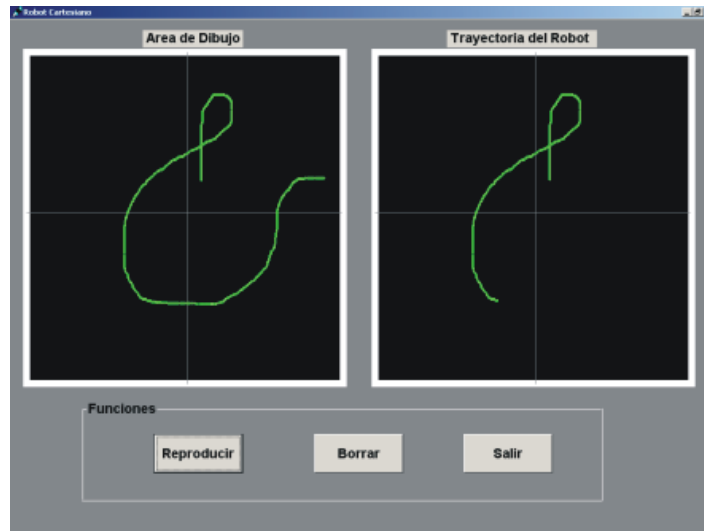


Figura 6.29: Ejemplo de una trayectoria con líneas curvas

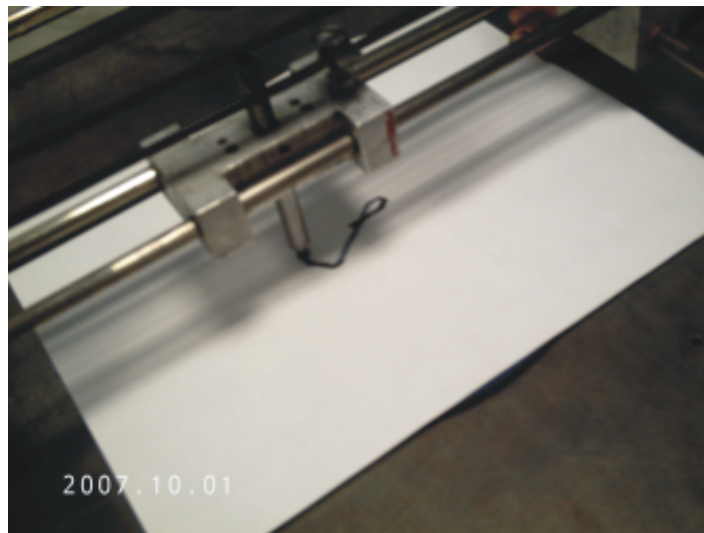


Figura 6.30: Reproducción de trayectoria de líneas curvas por el Robot Cartesiano

6.6. Conclusiones

Como pudimos observar en las imágenes la trayectoria que se traza en nuestra interfaz gráfica coincide con la trayectoria que traza nuestro robot. De esta manera podemos ver que el uso tanto del lenguaje de programación Visual Basic así como el manejo de puerto paralelo son herramientas aun útiles en el automatizado de máquinas robóticas.

Es necesario también conocer cada herramienta que se utiliza en nuestro lenguaje de programación para saber de que estamos hablando y como podemos manejarlo. En la disposición de pines del puerto paralelo de igual manera se debe de conocer la función de cada uno de los registros para no realizar operaciones erróneas en nuestro sistema que deseamos manipular desde nuestra computadora ya que podemos llegar a dañarla de igual manera que todo nuestro sistema electromecánico.

Capítulo 7

Conclusiones

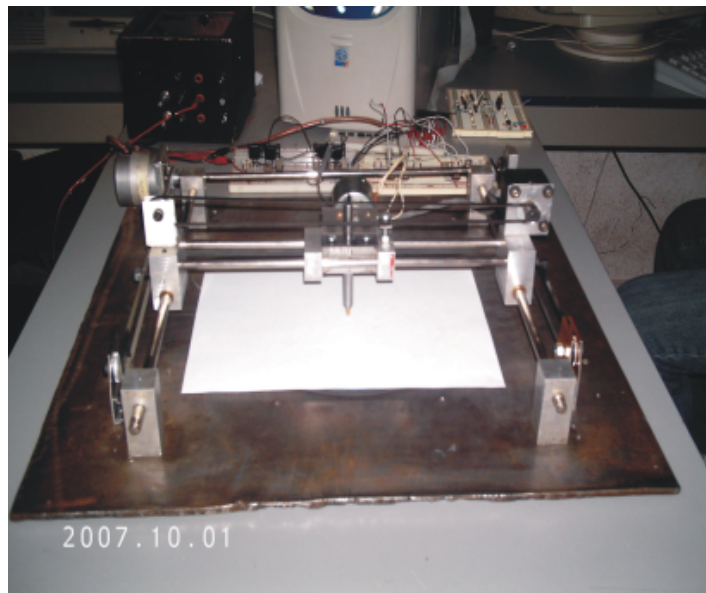


Figura 7.1: Robot Cartesiano

En este proyecto de tesis, uno de los inconvenientes que se tuvo en el trazado de líneas es que Visual Basic no es muy preciso por que la resolución que se maneja abarca en un rango de 5 pixeles por coordenada, en otros programas, para lo que en este solo lo toma como 1, y de tal manera que si se traza en este rango una pequeña curva lo toma como una lectura de una recta.

Una de las soluciones de este problema puede resolverse con programas requieren mayor proceso en el trazo de líneas como el CAD, CAM, y CAE y se usa las ordenes a través de lenguaje ensamblador programando en un PIC para que el robot reciba los movimientos programados en este.

Se sabe que estas operaciones con este tipo de robot se ha asignado por medio de lenguaje de AutoLISP, que es una herramienta de Autocad, la cual tiene una interfaz

que se puede interpretar en lenguaje Visual Basic, pero la desventaja es que el proceso de trazado abarca mucho tiempo en el programado y después ser interpretado para realizar el trazo.

El Robot Cartesiano resuelve en parte el trazo de líneas curvas que se requieran, ya que la mayoría de estos robots solo trazan líneas rectas que se utilizan mayormente en el up/down de herramientas, autopartes, cajas y en aplicaciones de localización de puntos para tareas de soldadura. En las configuraciones pequeñas es muy frecuente el trazado de pistas de PCB's a lo que el Robot cartesiano dará una mayor versatilidad en el trazado, no solo de pistas si no de cualquier tipo de trazos para varias aplicaciones.

Este proyecto se decidió hacerlo de manera que pudiese controlarse desde una interfaz grafica, debido a que también se puede hacer a través de WebCam, pero esto requiere de el procesamiento digital de señales que debe tener un control muy preciso debido a que es muy vulnerable a interferencias. La interfaz grafica ayuda a darle un control desde el software además de poder visualizar el trazado que se efectará y de una manera poco más sencilla.

Para trabajos a futuro se deja en las manos de los estudiantes que deseen mejorar el proyecto en cuanto al trazado de líneas curvas que como se aprecia tiene un margen de error en la exactitud de trazado ya que como se mencionaba con anterioridad en Visual Basic solo toma como una línea recta en un rango de 5 pixeles y esto hace que por ejemplo 3 coordenadas que se encuentren en especie de curva lo trace como una recta.

Una de las mejoras que se le puede hacer en cuanto a la estructura es el uso de un motor de torque para el levantamiento del perno/cremallera, ya que el uso de un motor de c.d. se tendría que mantener energizado mientras está levantado ya que el perno es muy pesado y puede quemarse el circuito electrónico de manipulación además si en caso de que no fuese así produciría un desgaste en los carbones del motor al mantenerlo energizado.

Por el contrario si se aplica un motor de torque a la hora de desactivar el giro del motor, su sistema de engranaje interno mantendría al perno levantado hasta activarlo nuevamente con otro pulso para un nuevo trazo.

Otra perspectiva a futuro que se le puede dar a este proyecto es en el uso de cortado de vidrio o lámina haciendo la implementación de un cortador o láser ya que si se utiliza el principio de nuestro programa diseñado reduciría tiempos muertos en el programado y carga en la memoria de algún microcontrolador para posteriormente relizar las tareas de cortado. Esto haría el aumento de la producción en el cortado de figuras de vidrio o lámina de una manera automatizada y segura.

Bibliografía

- [1] F. Gómez-Bravo*, F. Cuesta y A. Ollero, 'PLANIFICACIÓN DE TRAYECTORIAS EN ROBOTS MÓVILES BASADA EN TÉCNICAS DE CONTROL DE SISTEMAS NO HOLÓNOMO', *Depto. Ingeniería de Sistemas y Automática. Escuela Superior de Ingenieros, Universidad de Sevilla*, (2003).
- [2] Latombe, 1991; Ting-Yung, 1997
- [3] Aníbal Ollero Baturone, 'Robótica. Manipuladores y Robots móviles'(1998).
- [4] J.L. Guzmán, M. Berenguel, F. Rodríguez, Herramienta Interactiva para Robótica Móvil *Dpto. Lenguajes y Computación, Universidad de Almería*.
- [5] Mery Yaneth Sarmiento Saavedra and Jesús Pedroza Rojas, 'Construction of Machine for Thermal cut of Metallic Silhouettes Attended by Computer' *Revista Colombiana de Tecnologías de Avanzadas*, Vol. 1, No. 7 (2006).
- [6] Gustavo González, Guillermo Gutiérrez and Carlos Cagliolo, 'INTERPRETE DE CAD Y SU APLICACIÓN PARA OPTIMIZACIÓN DE PARÁMETROS DE CONTROL DE UN ROBOT INDUSTRIAL', *CUDAR, UTN-FRC*, Córdoba, Argentina, 2000.
- [7] 'Mesa de Posicionamiento XY', *Centro de Desarrollo e Investigación de Mecatrónica*, Grupos de Proyectos, 2002.
- [8] Structure Probe, Inc., 'Cortador Ultrasónico de Disco SPI', 1996 - 2007. *Structure Probe, Inc.*, 2007.
- [9] J.Gosálbez, A.Salazar, R.Miralles, I. Bosch, L. Vergara, 'MEJORA DE LA DETECCIÓN Y CARACTERIZACIÓN DE MATERIALES CON UN SISTEMA AUTOMÁTICO DE ULTRASONIDOS', *Departamento de Comunicaciones*, Universidad Politécnica de Valencia 2005.
- [10] Cesar Platero, 'Inspección y clasificación automática de pasta de papel TCF y EFC', *Visión Artificial con aportaciones al Reconocimiento de Formas*, UPM, 1998.
- [11] Michael Brady and Richard Paul, 'The First International Symposium' *Robotics Research*, The MIT Press, Cambridge MA, 1984.

- [12] Joseph L. Jones and Anita M. Flynn. 'Inspirations to implementation' *Mobile robots*, A K Peters Ltd, 1993.
- [13] Jonathan Carrera Díaz and Fernando Garrido Reséndiz, 'Software de Simulación del Robot PUMA Unimation 560 Basado en el Modelo Cinemático Directo e Inverso de Posición', *Diseño e implementación del software de simulación del robot PUMA 560*, University of State Hidalgo, 2000.
- [14] Rafael Sanz Domínguez, Aplicaciones industriales de la robótica, *Curso de verano: Robots y Humanos :hacia la convivencia*, Universidad de Santiago de Compostela, 19-22 de Julio del 2004.
- [15] Robótica Industrial, Ingeniería de Sistemas y automática, *Control de Robots y Sistemas sensoriales*, Control Cinemático.
- [16] Mikell P. Goover, Mitchell Weiss, Roger N. Nagel y Nicholas G. Odrey, *ROBOTICA INDUSTRIAL Tecnología, Programación y Aplicaciones*, Ed. McGraw-Hill 1990.
- [17] Fco. Javier Ceballos, Visual Basic, *versión 5, Curso de Programación*, Ed. Alfaomega 1997.
- [18] [Herramientas]
www.cancelarich.com.ar/Dalian1000x3000C.JPG
- [19] [Tipos de trayectorias]
<http://icaro.eii.us.es/descargas/RI/20transparencias/20tema/205/20Planif/20Tray/204porpagM.pdf>
- [20] [Herramientas]
www.infomaquinaria.com/.../006.jpg
- [21] [Herramientas]
www.empremaq.com.br/180.jpg
- [22] [Herramientas]
www.innovationplus.com/image/vernier_calipers...
- [23] [Control de motores a pasos por el puerto paralelo]
<http://www.infolaser.net/franpr/tecnica/mpap/mpap.html>
- [24] [Motores a pasos]
<http://www.monografias.com/trabajos37/motores/motores2.shtml>
- [25] [Medición de espesor de recubrimientos]
http://www.helmut-fischer.com/globalfiles/Segelboot_spanisch_04.pdf

- [26] [Morfología de los robots]
<http://www2.isa.uma.es/personal/antonio/Robotica/Tema2/20-/20Mecanica.pdf>
- [27] [Morfología y tipos de robots]
[http://icaro.eii.us.es/descargas/RI/202005/20transparencias/20tema/202/20prim/20parte/20\(4/20por/20pag\).pdf](http://icaro.eii.us.es/descargas/RI/202005/20transparencias/20tema/202/20prim/20parte/20(4/20por/20pag).pdf)
- [28] [Estructura de un Robot Industrial]
http://cfievalladolid2.net/tecno/cyr_01/robotica/sistema/morfologia.htm
- [29] [DISEÑO Y CONSTRUCCIÓN DE UN ROBOT ARTICULADO QUE EMULA MODELOS ANIMALES: APLICACIÓN A UN GUSANO]
http://www.learobotics.com/personal/juan/proyectos/cube-2.0/doc/cube_1.pdf
- [30] [CIRUGÍA ASISTIDA POR ORDENADOR. CONTROL DE UN BRAZO ROBOT EN APLICACIONES QUIRURGICAS]
http://www.depeca.uah.es/docencia/doctorado/cursos04_05/83190/Documentos/BrazoRobotTFC.pdf
- [31] [Librería de E/S para VB (funciona en NT/2000/XT)]
<http://www.pablin.com.ar/computer/programa/vb/iod11.htm>
- [32] [La robótica]
http://www.redacademica.edu.co/redacad/export/REDACADEMICA/beducadora/web_colegios/08/alqueria_fragua/robotica_insedaf/index.html
- [33] [Motores paso a paso]
http://www.sebyc.com/crr/descargas/motores_pap.pdf
- [34] [Control de motor a pasos]
http://www2.brazcubas.br/professores1/arquivos/20_franklin/T7037A/Stepper_Motor_PIC_Control_-_Espanhol.doc
- [35] [Motores de paso]
<http://www.todorobot.com.ar/informacion/tutorial/20stepper/stepper-tutorial.htm>
- [36] [Control de motores de cc]
http://robots-argentina.com.ar/MotorCC_PuenteH.htm
- [37] [Transistores]
http://html.rincondelvago.com/transistores_1.html

- [38] [Optoacopladores]
http://www.elprisma.com/apuntes/ingenieria_electrica_y_electronica/electronicadepotencia/default5.asp
- [39] [Microsoft Visual Basic]
http://es.wikipedia.org/wiki/Visual_Basic
- [40] [Generación de archivos GERBER]
<http://www.depeca.uah.es/wwwnueva/Extranet/20Ingles/docencia/ITI-EI/lte2/Practica7.pdf>
- [41] [Acero y Hierro]
<http://www.monografias.com/trabajos45/hierro-y-acero/hierro-y-acero.shtml>
- [42] [Acero]
<http://es.wikipedia.org/wiki/Acero>
- [43] [Qué es el acero]
http://www.infoacero.cl/acero/que_es.htm
- [44] [Aluminio]
<http://www.textoscientificos.com/quimica/aluminio>
- [45] [Aluminio]
<http://es.wikipedia.org/wiki/Aluminio>
- [46] [LOS PLÁSTICOS: MATERIALES A LA MEDIDA]
http://www.cientec.or.cr/ambiente/pdf/plasticos_materiales2003-CIENTEC.pdf
- [47] [Conectores del Puerto Paralelo (DB25)]
<http://r-luis.xbot.es/puerto/port01.html>

Apéndice A

Glosario

- **Sistemas no Holónomos:** definición en el estudio de geometría de sistemas mecánicos.
- **Actuadores:** son los dispositivos que producen la potencia mecánica que se requiere para mover los eslabones del manipulador. Pueden ser motores eléctricos, pistones y motores neumáticos e hidráulicos[3].
- **Mecanismos** de eslabones articulados: es un conjunto de sólidos (eslabones) articulados entre sí de tal manera que conduzcan al órgano terminal, colocado en el extremo de la estructura; a las posiciones y orientaciones necesarias para la realización de una tarea dada[3].
- **Articulaciones:** articulaciones de un robot pueden ser de revolución o prismática, la primera se refiere a un movimiento rotacional generado por el movimiento de la flecha de un motor, y la articulación prismática implica la conversión de un movimiento de revolución a un movimiento de traslación[13].
- **Organo terminal:** dispositivo que se instala en el extremo del manipulador, el cual interactúa con los objetos de su entorno al efectuar una tarea. Consiste en un sistema mecánico capaz de prender y soltar objetos (pinza mecánica) o herramienta capaz de realizar alguna operación (antorcha o pistola de soldadura, etc.)
- **Transmisiones de potencia:** permiten la conducción del movimiento generado en los actuadores hasta cada eslabón del manipulador. Pueden ser del tipo de bandas dentadas, engranes, cadenas, etc[16].
- **Grados de libertad(GDL):** corresponde al número mínimo de variables independientes que es necesario definir para determinar la situación en el espacio de todos los eslabones del manipulador. La regla de K. Kutzbach permite determinar el número de grados de libertad de un robot con mecanismos especiales en términos del número de eslabones y de articulaciones[16].

- **Espacio de trabajo:** es la zona del espacio físico que puede ser alcanzada por un punto del órgano terminal[14][3].
- **Robot cartesiano:** presentan una estructura articulada, es decir que se encuentra conformado por una serie de elementos ó eslabones que facilitan el movimiento, el cual puede ser de translación vertical, horizontal y transversal , los cuales se pueden combinar con movimiento de giro sobre el mismo eje[28].
- **Sistema mecánico:** conformado por soportes articulados los cuales permiten el giro o traslación de los ejes al ejecutar un proceso de movimiento en el plano de trabajo[32].
- **Sistema de accionamiento:** encargados de controlar la energía a los generadores en movimiento y motores los cuales pueden ser de corriente continua o corriente alterna[32].
- **Sistema de transmisión:** por lo general consta de un sistema de reductores los cuales reducen altas velocidades de los ejes en los motores y aumentan el torque[32].
- **Sistema sensorial:** lo compone todos aquellos sensores que se pueden acoplar internamente a las articulaciones ó también en la parte externa los cuales pueden captar la situación en que se encuentra el robot[32].
- **Sistema de control:** el control de un robot puede estar compuesto de diversas maneras como empleando temporizadores, coprocesadores especializados, contadores, unidades de entrada y salida, convertidores d/a ó de a/d [32].
- **Plano r:** es aquel que se refiere a la parte real de la posición de un objeto respecto a un ángulo formado entre los planos x,y,z.

Apéndice B

Puerto paralelo

B.1. Puerto paralelo

Las comunicaciones en paralelo se realizan mediante la transferencia simultánea de todos los bits que constituyen el dato (byte o palabra). Presentan la ventaja de que la transmisión puede ser más rápida. Sin embargo, las comunicaciones en paralelo no pueden ser implementadas para grandes distancias debido a que no es viable la conexión física de todas las líneas necesarias. Las comunicaciones en paralelo propiamente dichas no han sido normalizadas, lo que sí se reconoce es la norma Centronic;, para la conexión del PC a la impresora, mediante el envío simultáneo de 8 bits de datos (un byte), además de un conjunto de líneas de protocolo (handshake o intercambio). La operación más frecuente en la que interviene el puerto paralelo del PC es en el envío de datos a la impresora.

Los antiguos circuitos integrados que se incluían en las tarjetas de interface del puerto paralelo no permitían la recepción de datos, sólo estaban diseñados para el envío de información al exterior. Las versiones recientes de estas tarjetas de interface de puertos paralelo sí permiten la recepción de datos y dan la posibilidad, por ejemplo, de intercambiar información entre PC a través del puerto paralelo, siempre que se utilice el software adecuado.

N.º de pin		Señal	Sentido	Descripción
DB-25	DB-36		PC-PRN	
1	1	/STR	→	STROBE. Validación del dato (activa a nivel bajo). Un nivel L indica a la impresora que el dato es válido.
2	2	D0	→	Bit 0 de datos.
3	3	D1	→	Bit 1 de datos.
4	4	D2	→	Bit 2 de datos.
5	5	D3	→	Bit 3 de datos.
6	6	D4	→	Bit 4 de datos.
7	7	D5	→	Bit 5 de datos.
8	8	D6	→	Bit 6 de datos.
9	9	D7	→	Bit 7 de datos.
10	10	/ACK	←	ACKNOWLEDGE. (Activa a nivel bajo). Un nivel L indica que la impresora está en disposición de recibir un nuevo dato.
11	11	BSY	←	BUSY. Ocupada. Un nivel H indica que la impresora está ocupada y no puede recibir datos.
12	12	PAP	←	PAPER END. Sin papel. Un nivel H indica que la impresora se ha quedado sin papel.
13	13	OFON	←	ON LINE. Conectada. Un nivel H indica que la impresora está conectada y en línea.
14	14	/ALF	→	AUTO LINE FEED. Cambio de línea automático (Activa a nivel bajo). Un nivel L indica a la impresora que cuando reciba un retorno de carro debe hacer también un cambio de línea automáticamente.
15	32	/ERR	←	ERROR. (Activa a nivel bajo). Un nivel L indica que se ha producido un error en la impresora (buffer lleno, impresora fuera de línea, etc.).
16	31	/INI	→	INITIALIZE PRINTER. Inicialización (Activa a nivel bajo). Un nivel L inicializa o provoca un reset en la impresora (si la impresora lo admite).
17	36	/DSL	→	SELECT. (Activa a nivel bajo). Un nivel L selecciona o pone on line la impresora (si la impresora lo admite).
18 - 25	19-30 33	Masa		Referencia de tensión para las señales.
—	16	0V		—
—	17	Chasis		Conexión al chasis del equipo.
—	18	+VCC		Tensión de +5 V.
—	34,35	—		No utilizada.

Figura B.1: Descripción de los pines del Puerto paralelo.

B.1.1. Puerto paralelo en la Pc

Todos los ordenadores tipo PC están equipados, al menos, con una tarjeta de interface paralelo, frecuentemente junto a un interface serie. Como sistema operativo, el DOS puede gestionar hasta cuatro interfaces de puertos paralelo, LPT1 , LPT2 , LPT 3 y LPT4, además, reserva las siglas PRN como sinónimo del LPT 1, de modo que puede ser tratado como un archivo genérico. En el byte 0040:0011 del BIOS almacena el número de interfaces de puertos paralelo que se hayan instalado en el equipo. La dirección de entrada/salida de cada uno de los puertos paralelo y el número de puertos instalados en un PC se muestra en la pantalla inicial de arranque del equipo es frecuente, casi estandard que las direcciones de los dos primeros puertos paralelo sean las siguientes:

- LPT1 = 0x378 Hexadecimal
- LPT2 = 0x278 Hexadecimal

El puerto paralelo cuenta con tres registros los cuales estan, el de **control**, el de **datos** y el de **estado**. Las señales que interviene en el puerto paralelo tienen un bit asociado a cada uno de los registros.

El registro de datos

Es de tipo latch de 8 bits, que puede ser leído y escrito desde el procesador. Es el registro donde el procesador, en operaciones de salida (OUT), pone el dato que se quiere enviar a la impresora y su dirección coincide con la dirección base del puerto paralelo. En la figura siguiente muestra la disposición de los bits asociado a cada uno de los pines del **DB-25**.

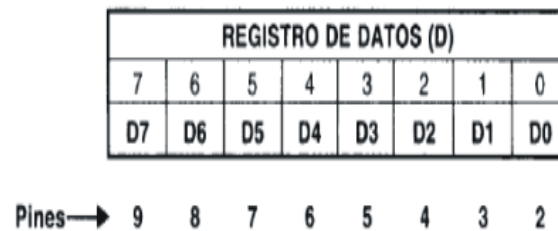


Figura B.2: Registro de datos y los pines correspondientes al conector DB-25.

El registro de estado

El registro de estado indica la situación actual de la impresora conectada al puerto, de acuerdo con los niveles de tensión que tengan las líneas *ACK*, *BSY*, *PAP* y *OF/ON*, lo que permite controlar el comportamiento de la impresora.

Se trata de un registro de entrada (Lectura) de información, su dirección se obtiene sumando 1 a la dirección base del puerto (**0x379** en LPT1).



Figura B.3: Registro de estado y los pines correspondientes al conector DB-25.

El símbolo \ll / \gg delante del nombre del bit indica que es activo a nivel bajo. Pero el bit 7 además (/ BSY) del registro de estado (bit 7) es invertido por el hardware y, por tanto, la línea tiene un nivel complementado al que aparece en ese bit.

El significado que tienen los bits de este registro es el siguiente:

- Si el bit 7 (/BSY → Busy) está a 0, significa que la impresora está ocupada, en este caso el buffer de impresión marca: lleno, procesando información, pendiente de inicializar, etc.
- El bit 6 (/ACK → Acknowledge) indica que se ha producido una transferencia correcta: cuando del puerto paralelo se transfiere un byte a la impresora, la impresora activa la línea ACK de reconocimiento del carácter y, como consecuencia, el bit ACK del registro de estado pasa a nivel bajo; cuando el bit ACK está a nivel alto, significa que la impresora está ocupada y no se pueden realizar envíos.
- El bit 5 (PAP → Paper) si está a 1, señala que la impresora no dispone de papel.
- El bit 4 (OF/ON → Line Off) indica cuando está a 1, que la impresora no está en línea.
- El bit 3 (ERR) si está a 0, indica que se ha producido un error de impresora (mal funcionamiento, falta de papel, impresora fuera de línea ...).

Nota: Los bits 0, 1 y 2 no se utilizan.

El registro de control

El registro de control permite controlar las transferencias de información con la impresora, y puede ser escrito y leído desde el microprocesador. Es un registro de entrada/salida cuya dirección se obtiene sumando 2 a la dirección base del puerto (**0x37A** en LPT1). El símbolo \ll / \gg delante del nombre del bit indica que es activo a nivel bajo.

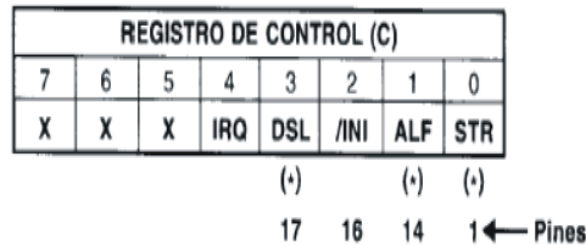


Figura B.4: Registro de control y los pines correspondientes al conector DB-25.

El símbolo * indica que los bits STR, ALF y OSL del registro de control son invertidos por el hardware con relación a las líneas correspondientes al cable de conexión, por lo que el nivel de los bits 0,1 y 3 del registro es complementado con relación a las líneas correspondientes.

El significado que tienen los bits de este registro es el siguiente:

- El bit 4 (IRQ) es el que permite controlar la generación de interrupciones de tipo hardware desde el puerto paralelo. Si este bit está a 1, el interface paralelo puede generar la petición de interrupción IRQ7 (en LPT1), que se corresponden con las interrupción 0x0Fh respectivamente del procesador 80X86. Esta petición de interrupción se produce cuando se da una transición H→L en la línea ACK.
- El bit 3 (DSL) : La mayoría de las impresoras paralelo IBM-compables, no utilizan esta línea y son activadas con un pulsador de on-line. El bit 2 (INI) produce una inicialización de la impresora (es poco utilizado).
- Si el bit 1 (ALF) está a nivel alto, la impresora produce automáticamente un cambio de línea (LF) cada vez que recibe un retorno de carro (CR).
- El bit 0 (STR) controla la línea que permite validar el dato existente en el registro de datos. La puesta a 1 del bit STR genera un impulso corto que indica a la impresora que el carácter del registro de datos es válido y debe ser aceptado. Así pues, cada vez que se precise enviar un carácter, no basta con ponerlo en el registro de datos, sino que hay que hacer un reset en el bit STR del registro de control y validar el dato volviendo a poner un 1 en ese bit.

- Los bits 5, 6 y 7 no se utilizan.

B.2. Entradas y salidas por el puerto paralelo

Al hablar de operaciones de entrada y salida por el puerto paralelo no debe olvidarse que, inicialmente, este elemento se desarrolló de acuerdo con el estándar Centronics con el fin, casi exclusivo, de que el PC pudiese enviar datos en paralelo a la impresora conectada, no se pensó en la posibilidad inversa: que el PC pudiese recibir datos a través de ese puerto. Las operaciones de entrada y salida de información a través del puerto paralelo en el PC las realizaremos gestionando el puerto paralelo en el nivel de registros, es decir, programando directamente los circuitos integrados o chips que constituyen la tarjeta de interface, lo cual permitirá aprovechar al máximo todas las posibilidades que ofrezca realmente el hardware de la tarjeta de interface.

B.2.1. Características E/S

Cuando usamos el puerto paralelo para otro cometido distinto al original, solo podemos hablar de 12 líneas de salida de información desde el ordenador:

- pines del 2 al 9 → registro de datos
- pines 1,14, 16 y 17 → registro de control

15 líneas de entrada al mismo:

- pines 2-9 → registro de datos
- pines 10,11,12,13 y 15 → registro de estado

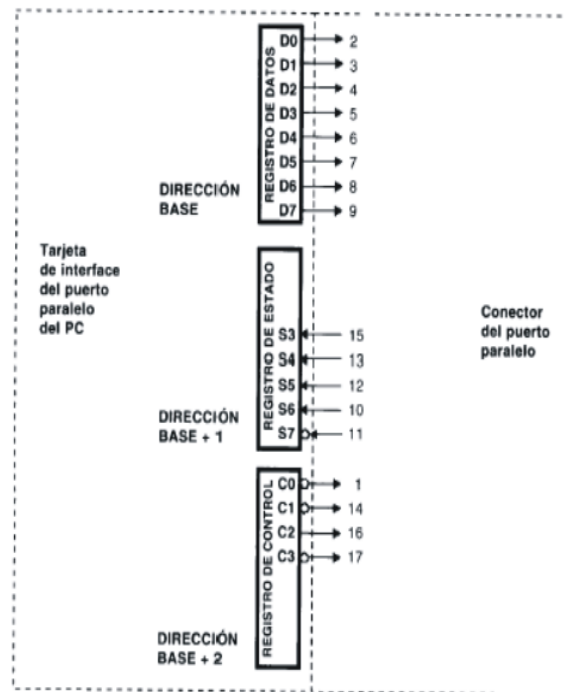


Figura B.5: Líneas del conector DB-25 del puerto paralelo.

Esto hace del puerto paralelo un interface de comunicación con el exterior bastante flexible. El registro de estado es de sólo lectura. Cuando se lee este registro, lo que se recibe es el estado lógico de los pines 10, 11, 12, 13 y 15 del conector DB-25 (el bit S 7 contiene el complemento del estado de la línea). Los tres bits de menor peso (S0-S2) no se utilizan y, habitualmente, se encuentran a nivel alto.

El registro de control es parcialmente bidireccional. Cuando se escribe en los cuatro bits de menor peso, C0 - C3, lo que se hace es establecer el nivel lógico de los pines C2 de forma directa y C0, C1 y C3 de forma complementada. Los tres bits de mayor peso C5 a C7 no se utilizan.

De forma experimental, se ha podido comprobar que, sólo en algunas tarjetas de interface paralelo, el bit C6 del registro de control influye en la configuración, de modo que si C6=0 las líneas de datos se configuran como ENTRADAS y si C6=1 las líneas de datos se configuran como SALIDAS. Otras tarjetas, sobre todo si son bidireccionales, no permiten el cambio de nivel de ese bit.

El registro de datos es de tipo latch de lectura y de escritura, de modo que cuando se realiza una operación de escritura (OUT) el dato se carga en los bits correspondientes y las líneas asociadas del conector tienden a alcanzar la tensión correspondiente a ese estado.

En algunas ocasiones las líneas de datos de la tarjeta de interface paralelo (Centronics) son bidireccionales, pero la etapa de salida se ha construido mediante buffers con transistores en colector abierto. En este caso, el hecho de que las operaciones de entrada y salida se hagan por las mismas líneas, condiciona notablemente el proceso de lectura, ya que con esa configuración electrónica de las líneas de datos (D0-D7), los valores lógicos leídos dependerán del nivel lógico presente en el registro y del valor de tensión en la línea (que no tienen por qué coincidir).

Nivel lógico de los bits en el registro de datos	Nivel de tensión en los pines del conector	Nivel lógico leído en una operación de lectura del puerto paralelo
L	L	L
L	H	L
H	L	L
H	H	H

Figura B.6: Niveles lógicos y de tensión del puerto paralelo.

Lo que se deduce en la figura es que, si se va a realizar una operación de lectura sobre el puerto paralelo, lo que se va a leer realmente es la operación AND lógica entre el nivel lógico del registro y el nivel lógico de la línea, lo que implica que, si se desea realizar una lectura real del estado de las líneas, deberá escribirse antes el dato 0xFF en el registro de datos del puerto paralelo.

Apéndice C

Programa del robot

En esta sección haremos referencias a algunas líneas de código que se utilizó para la programación de los movimientos del robot así mismo de la localización de las diferentes coordenadas que ayudan al trazo de las diferentes líneas.

C.1. Declaración de variables y de la librería IO.dll.

En las líneas siguientes se definen las funciones a utilizar de la librería IO.dll. Se hace referencia al nombre de la función y los argumentos de la misma.

```
Private Declare Sub ClrPortBit Lib "io.dll" (ByVal Port As Integer,
ByVal Bit As Byte)
Private Declare Sub PortOut Lib "io.dll" (ByVal Port As Integer,
ByVal Data As Byte)
Private Declare Sub SetPortBit Lib "io.dll" (ByVal Port As Integer,
ByVal Bit As Byte)
```

Se declaran las variables tales como matrices, contadores, el tipo de cada una y las constantes a utilizar en el programa.

```
Dim pintar As Boolean, VectMat(30) As Integer, Matx(0 To 30, 0 To
5000), Maty(0 To 30, 0 To 5000) As Double, contador As Integer,
sec1, sec2, sec3, sec4, sec5, sec6, sec7, sec8, x, y, doblepaso,
coordenada, No_Trayectoria, No_matriz As Integer

Const puertodatos = 888 /*Constante*/
```

C.1.1. Inicio de la aplicación.

Al iniciar la interfaz gráfica se hace una serie de configuraciones gráficas y de hardware tales como el ancho del píxel, limpiar los puertos de control y datos, color del píxel y el inicio del contador que llevara el control del número de trayectorias que se trazaran.

```

Private Sub Form_Load()          /* Inicio de la subrutina */
Private Sub Form_Load()          /* Inicio de la subrutina*/ Call
PortOut(puertodatos + 2, 11)     /* Se envía 11 al Puerto de control
                                para borrar los bits invertidos */
DrawWidth = 3                    /* Se configura el tamaño del píxel
                                a 3 */
ForeColor = RGB(0, 255, 0)       /* Color del píxel verde */
No_Trayectoria = 0               /* Se inicia el contador de control
                                de las trayectorias a 0 */
End Sub                          /* Fin de la subrutina */

```

C.1.2. Uso del evento MouseDown de visual basic.

En el siguiente código se limita el área de dibujo y se inician las matrices que guardaran las coordenadas de las trayectorias dibujadas.

```

/* Inicio del evento MouseDown */
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, x As
Single, y As Single)

/* Se limita el área de dibujo mediante una decisión */
If x < 450 Or x > 7110 Or y < 810 Or y > 7740 Then

/* Se muestra un mensaje si se intenta dibujar fuera del area
definida */
MsgBox "Coordenadas Fuera del Limite. ą Imposible Completar la
Trayectoria !", vbCritical,
      "Coordenadas Incorrectas"

/* Si el trazo se realiza en el área correcta, se activa el lápiz
para iniciar el dibujo */
Else
pintar = True

/* El contador que llevara el control del numero de trazos
realizados se activa incrementándose de 1 en 1 por cada trazo que se
haga */
No_Trayectoria = No_Trayectoria + 1

/* Se dibuja un píxel en la coordenada donde se pulso el botón del
Mouse */

```

```
PSet (x, y)
```

```
/* El contador que llevara el control del numero de coordenadas de
cada trayectoria se activa y se incrementara de 1 en 1 por cada
píxel o coordenada por la que pase el mouse en el evento MouseMove
*/
```

```
contador = 0
```

```
/* Se almacena la primera coordenada del dibujo en las matrices X e
Y respectivamente */
```

```
Matx(No_Trayectoria, contador) = x
```

```
Maty(No_Trayectoria, contador) = y
```

```
End If End Sub
```

C.1.3. Uso del evento MouseMove de visual basic.

La siguiente parte muestra es para guardar las coordenadas del evento MouseMove en las matrices X e Y.

```
/* Inicio del evento MouseMove */
```

```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, x As
Single, y As Single)
```

```
/* Al pulsar el botón del mouse se activa el lápiz para empezar a
dibujar la trayectoria, siempre y cuando esto se haga en el área
correcta. */
```

```
If pintar = True Then
```

```
/* Se evalúan los limites del área de dibujo */
```

```
If x > 449 Or x < 7109 Or y > 809 Or y < 7739 Then PSet (x, y)
```

```
/* El contador que controla el numero de coordenadas se incrementa
de 1 en 1 cada vez que el mouse cambia de coordenada */
```

```
contador = contador + 1
```

```
/* Se guardan las coordenadas por las que pasa el mouse en las
matrices X e Y durante el evento MouseMove */
```

```
Matx(No_Trayectoria, contador) = x
```

```
Maty(No_Trayectoria, contador) = y
```

```
End If
```

```
/* Si al realizar el trazo este sale del área permitida para el
```

```

dibujo se muestra un mensaje de error */
If pintar = True And (x < 449 Or x > 7109 Or y < 809 Or y > 7739)
Then MsgBox "Coordenadas Fuera del Limite. ą Imposible Completar la
Trayectoria !", vbCritical,
"Coordenadas Incorrectas"

/* Se desactiva el lápiz */
pintar = False

/* Se borra la pantalla */
Form1.Cls
End If
End If

/* Fin del evento MouseMove */
End Sub

```

C.1.4. Uso del evento MouseUp de visual basic

Esta subrutina pertenece al evento MouseUp y nos sirve para finalizar la trayectoria que se dibujo, así como guardar los contadores que nos ayudaran a recuperarla al momento de dibujarla con el robot cartesiano.

```

/* Inicio del evento MouseUp */
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, x As
Single, y As Single)

/* Al dejar de pulsar el botón del mouse se desactiva el lápiz y se
deja de dibujar */
pintar = False

/* Se almacena el total de coordenadas de la trayectoria en una
matriz o vector para posteriormente reproducirla con el robot.*/
VectMat(No_Trayectoria) = contador

/* Fin del evento MouseUp */
End Sub

```

C.1.5. Función Reproducir() de la interfaz grafica.

En esta parte se recupera la o las trayectorias dibujadas y se activa el robot para que las dibuje.

```
/* Inicio de la subrutina */
Private Sub Reproducir_Click()
```

Se emplean 8 variables para evitar que se pierda la secuencia de los motores al momento de hacerlos girar. Son 8 para cada una de las combinaciones tales son:

1. X adelante, Y no cambia
2. X atrás, Y no cambia
3. X no cambia, Y arriba
4. X no cambia, Y abajo
5. X adelante, Y arriba
6. X adelante Y abajo
7. X atrás, Y arriba
8. X atrás, Y abajo

Dichas variables sirven para mover un motor o dos al mismo tiempo y no perder la secuencia del o de los motores.

```
sec1 = 1
sec2 = 1
sec3 = 1
sec4 = 1
sec5 = 1
sec6 = 1
sec7 = 1
sec8 = 1
```

```
/* Se baja el lapiz del robot cartesiano */
Call PortOut(puertodatos + 2, 1)
```

Se inician bucles anidados para recuperar la o las trayectorias trazadas en el area de dibujo. Un bucle es para tomar el numero de trayectoria y otro para tomar el total de coordenadas de cada trayectoria

```
/* Se toma una a una hasta terminar con el numero de trayectorias */
For No_matriz = 0 To No_Trayectoria Step 1
```

```
/* Se toma una a una las coordenadas hasta terminar con total de
coordenadas de cada trayectoria. */
```

```

For coordenada = 1 To VectMat(No_matriz) Step 1

/* Se hace una comparación entre la coordenada actual y la siguiente
para ver que motor se tiene que mover y hacia que lado. Esto se hace
de igual manera en el resto de la subrutina Reproducir para cada
caso antes mencionados en las variables Sec1...Sec8. */
If Matx(No_matriz, coordenada) > Matx(No_matriz, coordenada - 1) And
Maty(No_matriz, coordenada) = Maty(No_matriz, coordenada - 1) Then

/* Se dibuja un píxel en cada coordenada de la trayectoria */
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz,
coordenada))

/* Se ponen a 1 las variables para evitar que se pierda la secuencia
de los motores */
sec2 = 1
sec3 = 1
sec4 = 1
sec5 = 1
sec6 = 1
sec7 = 1
sec8 = 1

/* Se evalúa la variable Sec1 Ę. Sec 8 que controla la secuencia
para ver en que paso se debe continuar para cada motor */
Select Case sec1
Case 1
Call SetPortBit(puertodatos, 0

/* Se llama una pausa para permitir el paso del motor y no perder la
secuencia */
Call Pausa

/* Se borra el bit activado para mover el motor */ Call
ClrPortBit(puertodatos, 0)
Call Pausa
sec1 = sec1 + 1

Case 2
Call SetPortBit(puertodatos, 3)
Call Pausa
ClrPortBit(puertodatos, 3)

```



```
Call Pausa
sec1 = sec1 + 1

Case 3
Call SetPortBit(puertodatos, 1)
Call Pausa
Call ClrPortBit(puertodatos, 1)
Call Pausa
sec1 = sec1 + 1

Case 4
Call SetPortBit(puertodatos, 2)
Call Pausa Call
ClrPortBit(puertodatos, 2)
Call Pausa
sec1 = 1

End Select
End If

If Matx(No_matriz, coordenada) < Matx(No_matriz, coordenada - 1) And
Maty(No_matriz, coordenada) = Maty(No_matriz, coordenada - 1) Then
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz, coordenada))
sec1 = 1
sec3 = 1
sec4 = 1
sec5 = 1
sec6 = 1
sec7 = 1
sec8 = 1

Select Case sec2

Case 1
Call SetPortBit(puertodatos, 3)
Call Pausa Call
ClrPortBit(puertodatos, 3)
Call Pausa
sec2 = sec2 + 1

Case 2
Call SetPortBit(puertodatos, 0)
```

```
Call Pausa
Call ClrPortBit(puertodatos, 0)
Call Pausa
sec2 = sec2 + 1

Case 3
Call SetPortBit(puertodatos, 2)
Call Pausa
Call ClrPortBit(puertodatos, 2)
Call Pausa
sec2 = sec2 + 1

Case 4
Call SetPortBit(puertodatos, 1)
Call Pausa
Call ClrPortBit(puertodatos, 1)
Call Pausa
sec2 = 1

End Select
End If

If Maty(No_matriz, coordenada) > Maty(No_matriz, coordenada - 1) And
Matx(No_matriz, coordenada) = Matx(No_matriz, coordenada - 1) Then
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz,
coordenada))
sec1 = 1
sec2 = 1
sec4 = 1
sec5 = 1
sec6 = 1
sec7 = 1
sec8 = 1

For doblepaso = 0 To 1 Step 1

Select Case sec3

Case 1
Call SetPortBit(puertodatos, 4)
Call Pausa
Call ClrPortBit(puertodatos, 4)
```

```
Call Pausa
```

```
sec3 = sec3 + 1
```

```
Case 2
```

```
Call SetPortBit(puertodatos, 7)
```

```
Call Pausa
```

```
Call ClrPortBit(puertodatos, 7)
```

```
Call Pausa
```

```
sec3 = sec3 + 1
```

```
Case 3
```

```
Call SetPortBit(puertodatos, 5)
```

```
Call Pausa
```

```
Call ClrPortBit(puertodatos, 5)
```

```
Call Pausa
```

```
sec3 = sec3 + 1
```

```
Case 4
```

```
Call SetPortBit(puertodatos, 6)
```

```
Call Pausa
```

```
Call ClrPortBit(puertodatos, 6)
```

```
Call Pausa
```

```
sec3 = 1
```

```
End Select
```

```
Next doblepaso
```

```
End If
```

```
If Maty(No_matriz, coordenada) < Maty(No_matriz, coordenada - 1) And  
Matx(No_matriz, coordenada) = Matx(No_matriz, coordenada - 1) Then  
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz, coordenada))
```

```
sec1 = 1
```

```
sec2 = 1
```

```
sec3 = 1
```

```
sec5 = 1
```

```
sec6 = 1
```

```
sec7 = 1
```

```
sec8 = 1
```

```
For doblepaso = 0 To 1 Step 1
```

```
Select Case sec4
```

```
Case 1
```

```
Call SetPortBit(puertodatos, 7)
```

```
Call Pausa
```

```
Call ClrPortBit(puertodatos, 7)
```

```
Call Pausa
```

```
sec4 = sec4 + 1
```

```
Case 2
```

```
Call SetPortBit(puertodatos, 4)
```

```
Call Pausa
```

```
Call ClrPortBit(puertodatos, 4)
```

```
Call Pausa
```

```
sec4 = sec4 + 1
```

```
Case 3
```

```
Call SetPortBit(puertodatos, 6)
```

```
Call Pausa
```

```
Call ClrPortBit(puertodatos, 6)
```

```
Call Pausa
```

```
sec4 = sec4 + 1
```

```
Case 4
```

```
Call SetPortBit(puertodatos, 5)
```

```
Call Pausa Call
```

```
Clr PortBit(puertodatos, 5)
```

```
Call Pausa
```

```
sec4 = 1
```

```
End Select
```

```
Next doblepaso
```

```
End If
```

```
If Matx(No_matriz, coordenada) > Matx(No_matriz, coordenada - 1) And
```

```
Maty(No_matriz, coordenada) > Maty(No_matriz, coordenada - 1) Then
```

```
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz,
```

```
coordenada))
```

```
sec1 = 1
```

```
sec2 = 1
```

```
sec3 = 1
```

```
sec4 = 1
```

```
sec6 = 1
```

```
sec7 = 1
sec8 = 1

Select Case sec5

Case 1
Call PortOut(puertodatos, 17)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos,7)
Call Pausa
Call ClrPortBit(puertodatos, 7)
Call Pausa
sec5 = sec5 + 1

Case 2
Call PortOut(puertodatos, 40)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 6)
Call Pausa
Call ClrPortBit(puertodatos, 6)
Call Pausa
sec5 = sec5 + 1

Case 3
Call PortOut(puertodatos, 18)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 7)
Call Pausa
Call ClrPortBit(puertodatos, 7)
Call Pausa
sec5 = sec5 + 1

Case 4
Call PortOut(puertodatos, 36)
Call Pausa
Call PortOut(puertodatos, 0)
```

```
Call Pausa
Call SetPortBit(puertodatos, 6)
Call Pausa
Call ClrPortBit(puertodatos, 6)
Call Pausa
sec5 = 1

End Select
End If

If Matx(No_matriz, coordenada) < Matx(No_matriz, coordenada - 1) And
Maty(No_matriz, coordenada) < Maty(No_matriz, coordenada - 1) Then
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz, coordenada))
sec1 = 1
sec2 = 1
sec3 = 1
sec4 = 1
sec5 = 1
sec7 = 1
sec8 = 1

End Select
End If

If Matx(No_matriz, coordenada) < Matx(No_matriz, coordenada - 1) And
Maty(No_matriz, coordenada) < Maty(No_matriz, coordenada - 1) Then
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz, coordenada))
sec1 = 1
sec2 = 1
sec3 = 1
sec4 = 1
sec5 = 1
sec7 = 1
sec8 = 1

Select Case sec6

Case 1
Call PortOut(puertodatos, 136)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
```

```
Call SetPortBit(puertodatos, 4)
Call Pausa
Call ClrPortBit(puertodatos, 4)
Call Pausa
sec6 = sec6 + 1
```

```
Case 2
Call PortOut(puertodatos, 65)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 5)
Call Pausa
Call ClrPortBit(puertodatos, 5)
Call Pausa
sec6 = sec6 + 1
```

```
Case 3
Call PortOut(puertodatos, 132)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 4)
Call Pausa
Call ClrPortBit(puertodatos, 4)
Call Pausa
sec6 = sec6 + 1
```

```
Case 4
Call PortOut(puertodatos, 66)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 5)
Call Pausa
Call ClrPortBit(puertodatos, 5)
Call Pausa
sec6 = 1
```

```
End Select
End If
```

```
If Matx(No_matriz, coordenada) > Matx(No_matriz, coordenada - 1) And
Maty(No_matriz, coordenada) < Maty(No_matriz, coordenada - 1) Then
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz, coordenada))
sec1 = 1
sec2 = 1
sec3 = 1
sec4 = 1
sec5 = 1
sec6 = 1
sec8 = 1
```

```
Select Case sec7
```

```
Case 1
```

```
Call PortOut(puertodatos, 129)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 4)
Call Pausa
Call ClrPortBit(puertodatos, 4)
Call Pausa
sec7 = sec7 + 1
```

```
Case 2
```

```
Call PortOut(puertodatos, 72)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 5)
Call Pausa
Call ClrPortBit(puertodatos, 5)
Call Pausa
sec7 = sec7 + 1
```

```
Case 3
```

```
Call PortOut(puertodatos, 130)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 4)
Call Pausa
```



```
Call ClrPortBit(puertodatos, 4)
```

```
Call Pausa
```

```
sec7 = sec7 + 1
```

```
Case 4
```

```
Call PortOut(puertodatos, 68)
```

```
Call Pausa
```

```
Call PortOut(puertodatos, 0)
```

```
Call Pausa
```

```
Call SetPortBit(puertodatos, 5)
```

```
Call Pausa
```

```
Call ClrPortBit(puertodatos, 5)
```

```
Call Pausa
```

```
sec7 = 1
```

```
End Select
```

```
End If
```

```
If Matx(No_matriz, coordenada) < Matx(No_matriz, coordenada - 1) And
```

```
Maty(No_matriz, coordenada) > Maty(No_matriz, coordenada - 1) Then
```

```
PSet (Matx(No_matriz, coordenada) + 7560, Maty(No_matriz, coordenada))
```

```
sec1 = 1
```

```
sec2 = 1
```

```
sec3 = 1
```

```
sec4 = 1
```

```
sec5 = 1
```

```
sec6 = 1
```

```
sec7 = 1
```

```
Select Case sec8
```

```
Case 1
```

```
Call PortOut(puertodatos, 24)
```

```
Call Pausa
```

```
Call PortOut(puertodatos, 0)
```

```
Call Pausa
```

```
Call SetPortBit(puertodatos, 7)
```

```
Call Pausa
```

```
Call ClrPortBit(puertodatos, 7)
```

```
Call Pausa
```

```
sec8 = sec8 + 1
```

```
Case 2
Call PortOut(puertodatos, 33)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 6)
Call Pausa
Call ClrPortBit(puertodatos, 6)
Call Pausa
sec8 = sec8 + 1

Case 3
Call PortOut(puertodatos, 20)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 7)
Call Pausa
Call ClrPortBit(puertodatos, 7)
Call Pausa
sec8 = sec8 + 1

Case 4
Call PortOut(puertodatos, 34)
Call Pausa
Call PortOut(puertodatos, 0)
Call Pausa
Call SetPortBit(puertodatos, 6)
Call Pausa
Call ClrPortBit(puertodatos, 6)
Call Pausa
sec8 = 1

End Select
End If

Next coordenada

Next No_matriz
Call PortOut(puertodatos + 2, 1) /* Se sube el lápiz del robot */
```

End Sub

C.1.6. Función Borrar() de la interfaz grafica

El siguiente código pertenece a la subrutina Borrar() y es empleada para borrar las trayectorias de la pantalla de la interfaz grafica.

```
Private Sub Borrar_Click() /* Inicio de la subrutina Borrar() */
Form1.Cls /* Borrarnos lo que se encuentra en la
           pantalla */
No_Trayectoria = 0 /* Iniciamos el contador No_Trayectoria
                   a 0 para empezar a contar las trayectorias que se t
End Sub /* Fin de la subrutina Borrar() */
```

C.1.7. Función Salir() de la interfaz grafica.

```
Private Sub Salir_Click() /* Inicio de la subrutina Salir() */
End /* Se termina y cierra la interfaz
    grafica */
End Sub /* Fin de la subrutina Borrar() */
```