



INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS

UPIITA

Trabajo Terminal

**BRAZO MANIPULADOR PARA
APLICACIONES DIDÁCTICAS**

Que para obtener el Título de

S. E. P.

“Ingeniero en Mecatrónica”

UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
BIBLIOTECA

Presentan:

Noé Martínez Rodríguez
Oscar Murillo Ramírez
Juan Carlos Pineda Pérez

Asesores:

M. en C. Cecilia Fernández Nava
Dr. José Alfredo Rosas Flores



INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA EN
INGENIERÍA Y TECNOLOGÍAS AVANZADAS

UPIITA

Trabajo Terminal

BRAZO MANIPULADOR PARA
APLICACIONES DIDÁCTICAS

Que para obtener el Título de
“Ingeniero en Mecatrónica”

INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS
BIBLIOTECA

Presentan:

Noé Martínez Rodríguez
Oscar Murillo Ramírez
Juan Carlos Pineda Pérez

Asesores:

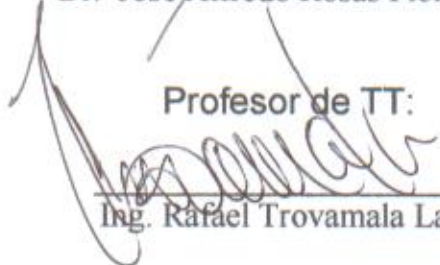

M. en C. Cecilia Fernández Nava


Dr. José Alfredo Rosas Flores

Presidente del Jurado:


M. en C. Gastón Hugo Salazar Silva

Profesor de TT:


Ing. Rafael Trovamala Landa

AGRADECIMIENTOS

A mis padres por todo el apoyo y la comprensión que me han brindado durante toda la carrera, ya que ellos han sido mi principal pilar a lo largo de toda mi formación.

A mis hermano por su apoyo y por ser un gran ejemplo.

A mis amigos de la vocacional y de la UPIITA por todo el tiempo de convivencia que compartimos durante todos estos años.

A mi amiga y compañera por todo el apoyo y cariño que me ha brindado.

Noé Martínez Rodríguez

AGRADECIMIENTOS

A mis padres por su invaluable y constante sacrificio en pos de mi formación y educación, sin ellos nada de lo que he logrado sería posible.

A mis hermanos por sus consejos, su gran apoyo y por ser un buen ejemplo a seguir.

A mi pareja por el apoyo brindado, por el tiempo que le he robado y por su comprensión.

A todos mis amigos por el tiempo que hemos compartido a lo largo de mi vida.

Oscar Murillo Ramírez

AGRADECIMIENTOS

Primero que nada quiero darte las gracias a ti DIOS por haberme permitido terminar la Ingeniería en Mecatrónica, por no dejarme solo en los momentos más difíciles y por ser mi guía y fortaleza a lo largo de estos cinco años, yo siempre he dicho que si DIOS tienes consentidos yo debo de ser uno de ellos por que solo así puedo explicarme todas las bendiciones que me regala día a día; De todo corazón muchas gracias Papá Chuchito.

También quiero darle las gracias a mis padres por el apoyo que me han brindado, no solo durante este tiempo sino a lo largo de toda mi vida, a mi papá por sus consejos, por su apoyo y por que no decirlo por sus regaños, a mi mamá por su paciencia, por su confianza y por todos los desvelos que la he hecho pasar, a mis hermanas por escucharme por apoyarme y por ser siempre amigas incondicionales.

Quiero agradecerles a todos mis amigos, por que con ellos he compartido amistad, confianza, alegrías, tristezas y hasta las respuestas de los exámenes, a todos y a cada uno de ellos muchas gracias.

Finalmente quiero darle las gracias a una personita muy especial a mi "Princesa de KraMeLitO", por todo el tiempo que ha estado a mi lado, por las tardes que pasamos juntos en la biblioteca pensando en el diseño del manipulador al lado de mis compañeros, por acompañarme en el taller mientras maquinaba las piezas y en el CIM cuando se realizó el ensamble, por estar conmigo en los exámenes profesionales, por la confianza que me ha brindado, por ser la niña más linda que he conocido por entenderme y comprenderme, por apoyarme en los momentos en los que parecía que las cosas no resultarían como se planearon, por enseñarme a hacer travesuras, por quererme y por amarme y por hacerme muy muy feliz, muchas gracias Angie jamás lo voy a olvidar.

Juan Carlos Pineda Pérez

ÍNDICE

RESUMEN	I
OBJETIVOS	II
INTRODUCCIÓN	III
1 MARCO TEÓRICO	1
1.1 Reseña histórica	1
1.1.1 Primer uso de la palabra robot.....	1
1.1.2 ¿Qué es un robot?.....	1
1.1.3 Clasificación de los robots industriales en generaciones.....	2
1.1.4 Primer uso de la palabra robótica.....	2
1.1.6 ¿Que es la robótica?.....	2
1.1.7 Como funcionan los robots.....	3
1.1.8 Uso de los robots en la actualidad.....	3
1.1.9 Impacto de los robots.....	4
1.1.10 Tecnologías del futuro.....	4
1.2 Aplicación de robots manipuladores en la educación	5
1.3 Estructura mecánica de un robot manipulador	8
1.3.1 Robot cartesiano.....	9
1.3.2 Robot articulado.....	9
1.3.3 Robot cilíndrico.....	10
1.3.4 Robot esférico.....	10
1.3.5 Robot SCARA.....	11
1.4 Transmisión de potencia y movimiento	12
1.4.1 Ejes.....	12
1.4.2 Engranés.....	14
1.4.3 Transmisión por bandas y cadenas.....	16
1.4.4 Acoplamientos.....	17
1.4.5 Cojinetes o rodamientos.....	18
1.5 Actuadores	22
1.5.1 Actuadores Neumáticos.....	23
1.5.1 Actuadores Hidráulicos.....	23
1.5.1 Actuadores Eléctricos.....	24
1.6 Sensores	26
1.6.1 Sensores internos.....	26
1.6.2 Sensores externos.....	30

1.7 Control de los manipuladores	31
1.7.1 Descripción de la orientación y posición	31
1.7.2 Cinemática directa de los manipuladores	32
1.7.3 Cinemática inversa de los manipuladores	34
1.7.4 Velocidades, fuerzas estáticas y singularidades	35
1.7.5 Dinámica	35
1.7.6 Generación de trayectoria	36
1.8 Lenguajes de programación para robot	37
1.9 Control de Motores	38
1.9.1 Control PI y PID	38
1.9.2 Sintonización de las constantes K_P , T_I y T_D	39
1.9.3 Tipos de control de potencia	41
2 DISEÑO DEL PROTOTIPO	43
2.1 Diseño mecánico	43
2.1.1 Selección de la configuración del manipulador	43
2.1.2 Diseño estructural (preliminar) del manipulador	43
2.1.3 Comparación y selección de materiales	44
2.1.4 Definición de la transmisión a utilizar	45
2.1.5 Selección de los rodamientos	45
2.1.6 Análisis estático a través de software	45
2.1.7 Análisis dinámico a través de software	47
2.1.8 Análisis matemático de la transmisión por engranes	53
2.2 Análisis cinemático del manipulador	55
2.2.1 Cinemática Directa	55
2.2.2 Cinemática Inversa	57
2.2.3 Jacobiano	59
2.3 Diseño electrónico	61
2.3.1 Descripción del sistema de control	61
2.3.2 Interfaz PC - Robot	62
2.3.3 Breve descripción del LM629	63
2.3.4 Filtro Proporcional Integral y Proporcional Integral Derivativo	64
2.3.5 Descripción de terminales del LM629	66
2.3.6 Arquitectura del LM629	67
2.3.7 Sensores requeridos por el LM629	68
2.3.8 Operaciones de lectura y escritura	69
2.3.9 Salidas al motor	69
2.3.10 Etapa de potencia (puente H)	70
2.3.11 Características del puente H	70
2.3.12 Descripción de terminales del puente H	71
2.3.13 Encoders	73

2.4 Interfaz Usuario-Manipulador	74
2.4.1 Diagrama de flujo.....	74
2.4.2 Selección del lenguaje de programación.....	76
2.4.3 Presentación preliminar del ambiente grafico.....	77
2.4.4 Comunicación PC-Manipulador.....	78
2.4.5 Puerto serie de la PC.....	79
3 CONSTRUCCIÓN DEL PROTOTIPO	81
3.1 Construcción mecánica	81
3.2 Construcción electrónica	85
3.2.1 Tarjeta modulo de comunicaciones y control general.....	86
3.2.2 Tarjeta modulo de control PID.....	87
3.2.3 Descripción del funcionamiento del sistema.....	87
3.2.4 El LM629N: ventajas y desventajas de usarlo.....	88
3.2.5 Reglas de sintonización para controladores PID.....	89
3.2.6 Fuente de alimentación.....	91
3.3 Desarrollo de la interfaz gráfica (Programación)	92
3.3.1 Presentación de BMAD.....	92
3.3.2 Menú.....	92
3.3.3 Botones principales.....	94
3.3.4 Otros controles.....	98
4 CONCLUSIONES	103
APÉNDICE A Planos estructurales del manipulador	106
APÉNDICE B Diagramas electrónicos del control del manipulador	117
APÉNDICE C Especificaciones de los circuitos integrados	121
APÉNDICE D Motor SWF de ángulo recto	125
APÉNDICE E Tablas de precios de componentes	126
APÉNDICE F Propiedades del aluminio	128
APÉNDICE G Código de la interfaz máquina - usuario	129
APÉNDICE H Protocolo de comunicación	133
APÉNDICE I Recepción y transmisión de datos mediante la UART del AVR AT90S8515	139
BIBLIOGRAFÍA	142

BRAZO MANIPULADOR PARA APLICACIONES DIDÁCTICAS (BMAD)

Palabras claves: Manipulador, robot, control PID, microcontrolador, programación, motores, maquinado, engranes.

Abstract: The purpose of this project consists of the designing and the construction of a three degrees of freedom Manipulator that is able to follow trajectories using a PI or PID control in order to compare the advantages and disadvantages. The design and the construction of the robot system are under a didactical approach.

Resumen: Tomando en cuenta que los brazos manipuladores son difíciles de adquirir, debido a su elevado costo, resulta atractiva la idea de desarrollar un brazo manipulador de tres grados de libertad capaz de seguir trayectorias previamente establecidas, es decir, programadas. Esto se logra empleando diferentes tipos de control tales como PI (Proporcional Integral) y PID (Proporcional Integral Derivativo) a fin de observar su comportamiento, las ventajas y desventajas de cada uno de los métodos empleados. Por supuesto, el brazo manipulador debe de ser útil, económico, versátil y de fácil adquisición para cualquier institución educativa que esté interesada en las ramas de la ingeniería tales como: la mecánica, la programación, el control, la automatización o la electrónica. De esta manera el brazo manipulador puede ser utilizado en centros educativos y de investigación.

A continuación se describe más detalladamente en que consiste el manipulador didáctico:

En lo referente a la parte mecánica, el acoplamiento entre actuadores y uniones es a través de engranes con lo cual se alcanza un grado de precisión confiable, los actuadores serán motores de CD (corriente directa) combinados con electrónica de potencia. La etapa de control del manipulador brazo, se realizará mediante un microcontrolador.

La computadora personal y el microcontrolador se encuentran conectados mediante comunicación serial, la PC lleva a cabo el cálculo de la cinemática directa e inversa a fin de generar la trayectoria seleccionada, los datos son decodificados y procesados por el microcontrolador para realizar la etapa de control.

El microcontrolador se encarga de enviar los datos, señales de control, así como los tiempos que se requieren para llevar a cabo los diferentes tipos de control. La seguridad está controlada mediante las señales que emiten los sensores de límite las cuales son introducidas al microcontrolador y éstas a su vez provocan las interrupciones. Por otra parte la medición de la posición y la velocidad se realiza utilizando encoders incrementales (codificadores de posición).

La forma de comunicación entre el usuario y el robot se efectúa por medio de una interfaz gráfica ejecutable en ambiente Windows desarrollada en el lenguaje de programación C++.

Objetivo general:

Desarrollar un brazo manipulador para aplicaciones didácticas en pruebas de trayectorias predefinidas, empleando las leyes de control PI y PID.

Objetivos particulares:

- ✓ Diseñar y construir mecánicamente un manipulador para la prueba de trayectorias.
- ✓ Desarrollar tarjetas electrónicas para el control PID de tres motores.
- ✓ Desarrollar la electrónica de potencia para los motores.
- ✓ Programar una interfaz gráfica donde se lleve a cabo la cinemática del manipulador para que el usuario pueda interactuar con dicho manipulador.
- ✓ Comunicar la PC con la electrónica de control.
- ✓ Integrar sinérgicamente los elementos del proyecto para obtener como resultado final un prototipo que refleje los campos de acción de la ingeniería Mecatrónica.

Introducción

En el primer capítulo se da una breve introducción acerca de los fundamentos teóricos que se consideran necesarios para desarrollar un brazo manipulador, también se da una referencia de los principales componentes mecánicos y electrónicos que se pueden utilizar en la construcción de dicho manipulador. Lo que se pretende al dar este marco teórico, es proporcionar al lector los conocimientos teóricos necesarios para que pueda comprender mejor el proyecto a desarrollar y además, de ésta información teórica puede obtener una pauta para que cada lector opte por los elementos que en determinado momento él considere necesarios para llevar a cabo el desarrollo de su proyecto particular.

En el segundo capítulo se desarrolla el diseño del manipulador comenzando por la parte mecánica en donde se presenta la transmisión seleccionada, los ejes, rodamientos, motor y los análisis estáticos y dinámicos realizados a la estructura del manipulador. Posteriormente se continúa con la parte electrónica en donde se incluye una descripción general de los circuitos integrados que se emplearán para llevar a cabo el control del brazo manipulador (electrónica de control y electrónica de potencia). Por último se proporciona una propuesta de la interfaz que se desarrollará para que usuario pueda interactuar con el brazo manipulador.

El tercer capítulo describe la construcción mecánica del manipulador; los logros obtenidos, problemas y las soluciones que se dieron, planteando también algunas mejoras. Se describe el desarrollo de las tarjetas electrónicas, igualmente problemas y soluciones en la sincronización y movimiento de los motores. En la parte final de este capítulo se describe detalladamente los controles y ventanas de la interfaz gráfica desarrollada con la finalidad de que el usuario interactúe con el manipulador.

CAPÍTULO 1. MARCO TEÓRICO

1.1 Reseña histórica

1.1.1 Primer uso de la palabra "robot"

El escritor checo Karel Capek (1890-1938) usó por primera vez la palabra "robot", de la palabra checa siervo o esclavo. El uso de la palabra Robot fue introducido en su obra R.U.R. (*Rossum's Universal Robots*) la cual se estrenó en Europa en enero de 1921. En R.U.R., Capek posee un paraíso, donde las máquinas inicialmente traen muchos beneficios pero al final traen una cantidad igual de males en la forma de desempleo y malestar social.

La obra fue un enorme éxito y otras producciones rápidamente estrenaron el tema de R.U.R. a lo largo de Europa y EU, en parte, el éxito fue la deshumanización del hombre en una civilización tecnológica [6].

1.1.2 ¿Qué es un robot?

No existe una definición precisa, pero por acuerdo general un robot es una maquina programable que imita las acciones o apariencia de una criatura inteligente, usualmente un humano. Para calificarse como un robot, una máquina tiene que ser capaz de hacer dos cosas: 1) Obtener información del medio que lo rodea, y 2) Hacer algo físico como mover o manipular objetos. [7].

Hoy la palabra robot tiene diferentes significados:

- La del *Robot Institute of América* (RIA - 1979): "un manipulador multifuncional y reprogramable, diseñado para mover materiales, piezas, herramientas o dispositivos especiales, mediante movimientos programables y variables que permitan llevar a cabo diversas tareas". [2].
- Esta definición, ligeramente modificada, ha sido adoptada por la *Organización Internacional de Estándares* (ISO) que define al robot industrial como: Manipulador multifuncional reprogramable con varios grados de libertad, capaz de manipular materiales, piezas, herramientas o dispositivos especiales según trayectorias variables programadas para realizar tareas diversas. [2].
- Se incluye en esta definición la necesidad de que el robot tenga varios grados de libertad. Una definición más completa es la establecida por la *Asociación Francesa de Normalización* (AFNOR) que define primero el manipulador y, basándose en dicha definición, el robot: [2].

Manipulador: mecanismo formado generalmente por elementos en serie, articulados entre sí, destinado al agarre y desplazamiento de objetos. Es multifuncional y puede ser gobernado directamente por un operador humano o mediante dispositivo lógico.

Robot: manipulador automático servocontrolado, reprogramable, polivalente, capaz de posicionar y orientar piezas, herramientas o dispositivos especiales, siguiendo trayectorias variables reprogramables, para la ejecución de tareas variadas. Normalmente tiene la forma de

uno o varios brazos terminados en una muñeca. Su unidad de control incluye un dispositivo de memoria y ocasionalmente de percepción del entorno. Normalmente su uso es el de realizar una tarea de manera cíclica, pudiéndose adaptar a otra sin cambios permanentes en su configuración.

- Por último, la *Federación Internacional de Robótica* (IFR) en su informe técnico ISO/TR 83737 (septiembre 1988) distingue entre robot industrial de manipulación y otros robots:

Por robot industrial de manipulación se entiende a una máquina de manipulación automática, reprogramable y multifuncional con tres o más ejes que pueden posicionar y orientar materias, piezas, herramientas o dispositivos especiales para la ejecución de trabajos diversos en las diferentes etapas de la producción industrial, ya sea en una posición fija o en movimiento.

1.1.3 Clasificación de los robots industriales en generaciones

1ª Generación. Repite la tarea programada secuencialmente.

No toma en cuenta las posibles alteraciones de su entorno.

2ª Generación. Adquiere información limitada de su entorno y actúa en consecuencia.

Puede localizar, clasificar (visión), detectar esfuerzos y adaptar sus movimientos en consecuencia.

3ª Generación. Su programación se realiza mediante el empleo de un lenguaje natural. Posee capacidad para la planificación automática de tareas.

En esta definición se debe entender que la reprogramabilidad y multifunción se consiguen sin modificaciones físicas del robot.

Común a todas las definiciones anteriores es la aceptación del robot industrial como un brazo mecánico con capacidad de manipulación y que incorpora un control más o menos complejo. Un sistema robotizado, en cambio, es un concepto más amplio. Engloba todos aquellos dispositivos que realizan tareas de forma automática en sustitución de un ser humano y que pueden incorporar uno o varios robots, siendo esto último lo más frecuente. [2].

1.1.4 Primer uso de la palabra "Robótica"

La palabra "Robótica" fue usada por primera vez en *Runaround*, una pequeña historia publicada en 1942, por Isaac Asimov (1920 – 1992). *Yo, robot*, es una colección de varias de estas historias, la cual fue publicada en 1950.

1.1.5 ¿Qué es la robótica?

Como se mencionó anteriormente el término "Robótica" fue acuñado por Asimov para describir la tecnología de los robots en donde predijo el aumento de una poderosa industria robótica. Esta predicción se ha hecho realidad, recientemente ha habido una explosión en el desarrollo y uso industrial de los robots tal que se ha llegado al punto de hablar de "revolución de los robots" y "era de los robots". El término robótica puede ser definido desde diversos puntos de vista [5]:

Actualmente las actividades que entrañan gran peligro para las personas, como la localización de barcos hundidos, la búsqueda de depósitos minerales submarinos o la exploración de volcanes activos, son especialmente apropiadas para emplear robots.

Ya se emplean robots para ayudar a los cirujanos a instalar caderas artificiales, y ciertos robots especializados de altísima precisión pueden ayudar en operaciones quirúrgicas delicadas en los ojos. La investigación en tele cirugía emplea robots controlados de forma remota por cirujanos expertos; estos robots podrían algún día efectuar operaciones en campos de batalla distantes.

1.1.8 Impacto de los robots

Los manipuladores robóticos crean productos manufacturados de mayor calidad y menor coste. Sin embargo, también pueden provocar la pérdida de empleos, especialmente en cadenas de montaje industriales. Aunque crean trabajos en los sectores de soporte lógico y desarrollo de sensores, en la instalación y mantenimiento de robots y en la conversión de fábricas antiguas y el diseño de fábricas nuevas, estos nuevos empleos exigen mayores niveles de capacidad y formación. Las sociedades orientadas hacia la tecnología deben enfrentarse a la tarea de volver a formar a los trabajadores que pierden su empleo debido a la automatización y enseñarles nuevas capacidades para que puedan tener un puesto de trabajo en las industrias del siglo XXI.

1.1.9 Tecnologías del futuro

Las máquinas automatizadas ayudarán cada vez más a los humanos en la fabricación de nuevos productos, el mantenimiento de las infraestructuras y el cuidado de hogares y empresas. Los robots podrán fabricar nuevas autopistas, construir estructuras de acero para edificios, limpiar conducciones subterráneas o cortar el césped. Ya existen prototipos que realizan todas esas tareas. Una tendencia importante es el desarrollo de sistemas microelectromecánicos, cuyo tamaño va desde centímetros hasta milímetros. Estos robots minúsculos podrían emplearse para avanzar por vasos sanguíneos con el fin de suministrar medicamentos o eliminar bloqueos arteriales. También podrían trabajar en el interior de grandes máquinas para diagnosticar con antelación posibles problemas mecánicos.

Puede que los cambios más espectaculares en los robots del futuro provengan de su capacidad de razonamiento cada vez mayor. El campo de la inteligencia artificial está pasando rápidamente de los laboratorios universitarios a la aplicación práctica en la industria, y se están desarrollando máquinas capaces de realizar tareas cognitivas como la planificación estratégica o el aprendizaje por experiencia. El diagnóstico de fallos en aviones o satélites, el mando en un campo de batalla o el control de grandes fábricas correrán cada vez más a cargo de ordenadores inteligentes [8].

1.2 Aplicación de robots manipuladores en la educación

A continuación se mencionan las principales empresas que se dedican ya sea al desarrollo o distribución de manipuladores con fines didácticos.

a) ROBODACTA

Es un portal administrado y respaldado por la empresa INTECSYS S.A. de C.V. Empresa mexicana dedicada al diseño y fabricación de controles electrónicos para equipo médico e industrial con siete años de experiencia con sede en la ciudad de México.

En ROBODACTA se pueden encontrar sistemas didácticos para construir robots. Desde kits básicos hasta componentes y accesorios para armar diseños propios. Esta empresa distribuye productos de las siguientes marcas: LEGO Mindstorms, OWI Kits, CYBUGs, Dynalloy, Solarbotics, etc. Un ejemplo de un brazo robótico es el que se muestra en la figura 1.2.1 [10]:



Figura 1.2.1 OWI-007 Robotic Arm Trainer (Brazo Robótico Entrenador).

Descripción:

- Este brazo mecánico de cuatro grados de libertad y la pinza, es dirigido mediante un control alámbrico que activa los 5 motores eléctricos de DC.
- Su sistema de engranaje puede observarse a través de su estructura transparente.
- Este brazo puede abrir y cerrar las pinzas, girar la muñeca, mover el codo, mover el hombro y girar su base.
- Recomendado de 14 años en adelante.

Tiene cinco ejes de movimiento:

1. Base giratoria de derecha e izquierda (350 grados aprox.).
2. Hombro (120 grados aprox.).
3. Codo (135 grados aprox.).
4. Muñeca (340 grados aprox.).
5. Pinza (5cm).

Precio \$1,650.00 pesos IVA Incluido

b) LEGO

La compañía LEGO es una organización que se ha comprometido en el proyecto de la IA (Inteligencia Artificial), y la ha aplicado en sus productos con el fin de revolucionar el juguete y las formas de entretenimiento de los niños y jóvenes de la nueva era.

Legó construyó la empresa LEGO MINDSTORM, una división que tiene como fin conjuntar el mundo de las computadoras con la IA y los juguetes de construcción. Los productos de estas empresas permiten que los niños diseñen y construyan sus propios robots LEGO, y además, pueden darles vida a través de una PC.

Los juguetes de LEGO permiten que el niño construya según su nivel de conocimiento. El primero es el de aprendiz, en el que hay que seguir el manual de instrucciones para que el juguete funcione; y el nivel más avanzado de todos, en el cual el niño puede fabricar su propio robot [9].

c) CRYA (Control Robótica y Automatización).

Como lo menciona la propia compañía “los kits de robótica son muy importantes en el aprendizaje de esta tecnología, no hay nada mejor que entender algo armándolo para después verlo funcionar”. Con diseños 100% mexicanos y con patente en trámite sus robots intentan responder a ciertas necesidades de los estudiantes que quieran empezar, profundizar y seguir armando robots con distintos grados de dificultad. En la figura 1.2.2 se muestra un manipulador desarrollado por esta compañía.



Figura 1.2.2 Robot brazo manipulador microcontrolado.

Este proyecto robot didáctico para el aprendizaje de mecanismos, electrónica y programación, cuenta con 3 grados de libertad, base, brazo y pinza, este brazo se puede controlar por medio de una computadora o también incluye la opción de controlarse mediante el PIC 16F84A para que trabaje de manera autónoma.

Precio: \$2000 C/U

d) Otros robots didácticos

La industria de los robots hágalo usted mismo, es un negocio de millones de dólares en Japón. Un ejemplo de estos robots para armar se muestra en la figura 1.2.3:



Figura 1.2.3 Brazo Robot Móvil (MR-999).

Representa todo un reto el armar este robot, a pesar de haber sido diseñado como una de las versiones más sencillas para la mecánica y la electrónica. El brazo robot incluye instrucciones bastante decentes para armarse, para que puedas entender cómo se arman sus 17 articulaciones, y requiere hacerle unos cuantos ajustes para que logre desempeñarse a toda su capacidad.

Precio: 36 dólares.

1.3 Estructura mecánica de un robot manipulador (Configuración Cinemática)

Una vez definido lo que se desea realizar, el primer paso es desglosar cada parte que conforma un manipulador, empezando por su estructura mecánica.

Mecánicamente, un robot es una serie de elementos o eslabones unidos mediante articulaciones que permiten el movimiento relativo entre cada par de eslabones consecutivos. El movimiento de cada articulación puede ser de desplazamiento, de giro, o de una combinación de ambos. De este modo son posibles los seis tipos diferentes de articulaciones (figura 1.3.1), aunque en la práctica en los robots solo se emplean la rotación y la prismática [2].

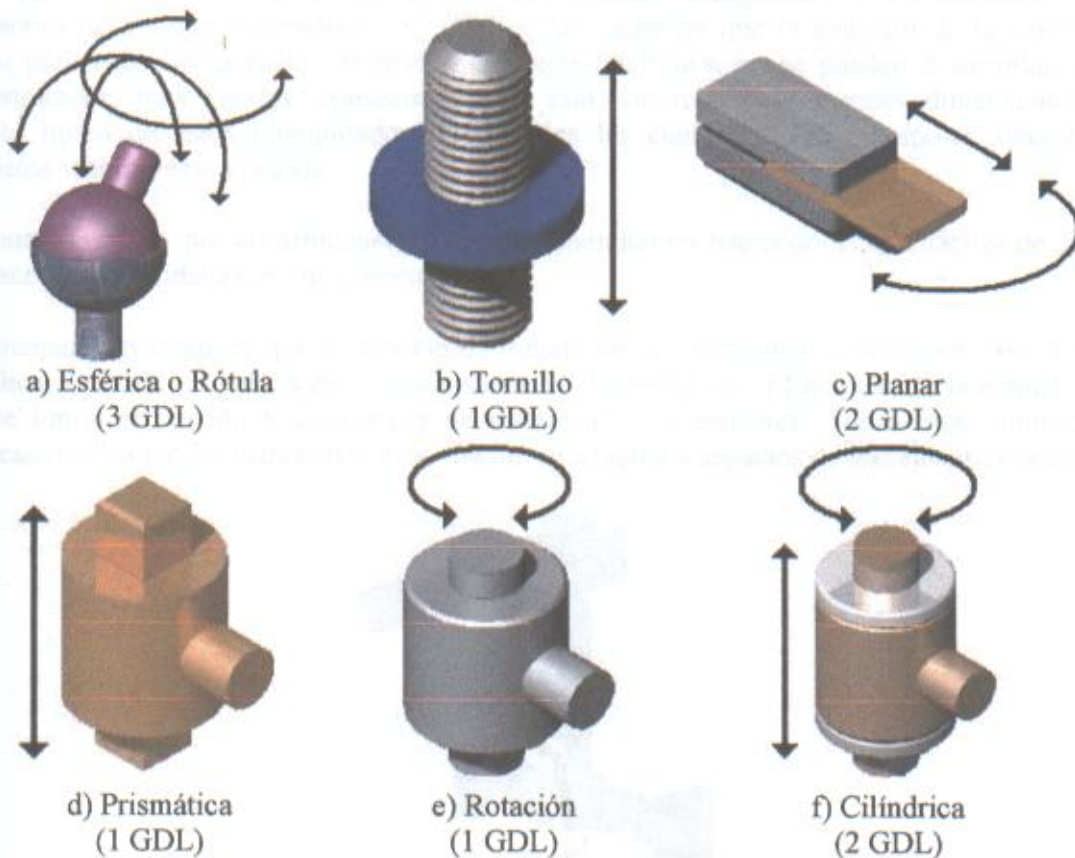


Figura 1.3.1 Tipos de articulaciones en robots.

Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior, se denomina grado de libertad (GDL). Ya que las articulaciones empleadas son únicamente las de rotación y prismática con un solo GDL cada una, generalmente el número de grados de libertad suele coincidir con el número de articulaciones del manipulador. Al combinar las diferentes articulaciones en un robot se obtienen configuraciones diferentes con características que considerar tanto en el diseño y construcción como en la aplicación del manipulador. Las configuraciones más usuales se describen a posteriormente, donde se muestran solamente los primeros tres GDL del robot, los cuales son los principales para posicionar su extremo en un punto en el espacio.

Hay otros tres ejes secundarios ubicados en la “muñeca” del manipulador. Estos ejes secundarios de movimiento son cabeceo, guiñada y giro. El cabeceo es el movimiento de arriba hacia abajo, la guiñada es el movimiento lateral de lado a lado y el giro es el movimiento de rotación.

La combinación de los ejes principales y secundarios proporciona al robot seis grados de libertad, que son los grados de libertad mínimos necesarios para replicar los movimientos básicos del brazo y muñeca humanos.

1.3.1 Robot Cartesiano

Probablemente un manipulador cartesiano tiene la configuración más directa. Como se muestra en la figura 1.3.2, sus articulaciones son mutuamente ortogonales y corresponden a las direcciones de los ejes cartesianos “X”, “Y” y “Z”, además que la solución de la cinemática inversa para esta configuración es trivial. Con esta configuración se pueden desarrollar robots con estructuras muy rígidas, consecuencia de esto son robots de grandes dimensiones. Un ejemplo típico de estos manipuladores son grúas las cuales pueden manipular automóviles completos y herramienta pesada.

Una ventaja más es que sus articulaciones son independientes haciéndolos más fáciles de diseñar y evitando singularidades en sus primeros 3 GDL.

Su principal desventaja es que su espacio de trabajo se encuentra dentro del robot. Por lo tanto las aplicaciones de los robots cartesianos son muy dependientes. El tamaño de la estructura de soporte limita el tamaño y colocación de instalaciones y sensores. Todas estas limitaciones provocan que los robots cartesianos difícilmente se adapten a espacios de trabajos existentes.

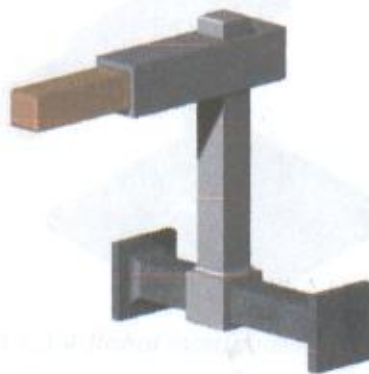


Figura 1.3.2 Robot manipulador cartesiano.

1.3.2 Robot articulado

En la figura 1.3.3 se muestra un robot articulado algunas veces llamado manipulador antropomorfo. Este manipulador consiste de tres articulaciones rotacionales, la primera es una rotación sobre el eje vertical, la segunda una elevación sobre el plano horizontal y otro más de elevación generalmente paralelo a la articulación anterior.

Debido a la configuración de los robots articulados la intrusión de su estructura mecánica es la menor de todos los manipuladores, siendo capaces de acceder a lugares reducidos. Otra ventaja es que necesitan una estructura total mucho menor que los robots cartesianos haciéndolos menos costosos.



Figura 1.3.3 Robot manipulador articulado.

1.3.4 Robot cilíndrico

Los manipuladores cilíndricos constan de tres articulaciones, la primera de traslación a lo largo de la cual corre el brazo verticalmente, la segunda es una articulación de rotación sobre el eje vertical y finalmente la tercera es una unión de traslación ortogonal al eje de rotación, al final de éste último eslabón se puede posicionar un efector final (figura 1.3.4).



Figura 1.3.4 Robot manipulador cilíndrico.

1.3.4 Robot esférico

Un manipulador que posee una configuración esférica tiene varias similitudes al manipulador articulado, con la diferencia de que la articulación del codo es reemplazada por una unión traslacional. En algunas aplicaciones esta configuración es más adecuada que la articular ya que el último eslabón puede sobresalir o retraerse según la necesidad (figura 1.3.5).



Figura 1.3.5 Robot manipulador esférico.

1.3.5 Robot SCARA (Selective Compliance Assembly Arm Robot, siglas en inglés)

La configuración SCARA, la cual se muestra en la figura 1.3.6, tiene dos o tres articulaciones de revolución paralelas, permitiendo así el movimiento y orientación en un plano con una cuarta articulación prismática para mover el efector final normal a dicho plano. Su principal ventaja es que sus primeras articulaciones no tienen que soportar, ni el peso del propio robot ni de su carga, otra ventaja es que los actuadores pueden ser muy grandes provocando un movimiento rápido del manipulador. Esta configuración es la más adecuada para tareas sobre un plano.



Figura 1.3.6 Robot SCARA.

1.4 Transmisión de potencia y movimiento

El siguiente aspecto a considerar es la transmisión, ya que de ella depende en buena parte que el manipulador pueda llevar a buen fin la tarea para la cual fue diseñado.

Sin lugar a duda los ejes juegan un papel muy importante en el diseño de cualquier mecanismo. En el proceso de transmitir potencia a una velocidad de giro o velocidad rotacional específica, el eje se sujeta, de manera inherente a un momento de torsión o par.

Es importante puntualizar que existen distintas formas de transmitir movimiento, pero dentro de esta investigación solo se describen las tres siguientes:

- Engranajes
- Bandas
- Cadenas

Con el propósito de elegir la que mejor se adecue a las necesidades del manipulador.

El siguiente elemento dentro de la transmisión que es importante mencionar son los acoplamientos. Los acoplamientos se usan para conectar secciones de ejes o para conectar el eje de una máquina conductora con el de la máquina conducida. Esto permite una conexión permanente, en contraste con los embragues, los cuales producen acoplamiento o desacoplamiento, a voluntad.

Finalmente pero sin menor grado de importancia se encuentran los cojinetes los cuales se encargan de soportar el eje y minimizar la fricción de éste cuando se encuentra en movimiento.

A continuación se da una descripción de los elementos mencionados en las líneas anteriores, a fin de que el lector tenga una visión más clara del proyecto a desarrollar y los fundamentos teóricos en los cuales está basado.

1.4.1 Ejes

Una flecha o eje es el componente de los dispositivos mecánicos que transmite energía rotacional y potencia. Es parte integral de dispositivos o artefactos como reductores de velocidad tipo engrane, impulsores de banda o cadena, transportadores, bombas, ventiladores, agitadores y muchos tipos de equipo para automatización. Como se muestra en la figura 1.4.4 el eje soporta componentes transmisores de potencia como engranes, poleas acanaladas para bombas o ruedas dentadas de cadena, que ejercen fuerzas sobre el eje en sentido transversal, es decir perpendicular a su eje. Estas fuerzas transversales provocan que se generen momentos de flexión en el eje, ello requiere de un análisis de tensión debida a la flexión.



Figura 1.4.4 Eje soportando componentes de transmisión.

Procedimiento de diseño de flechas o ejes

Debido a la aparición simultánea de tensiones por esfuerzo de corte por torsión y tensiones normales que se deben a la flexión, el análisis de una flecha o eje virtualmente implica siempre el uso de un enfoque combinado para el aspecto de las tensiones.

El método que se sugiere para el diseño de ejes es el de la teoría de la *Falla por distorsión de la energía*. En ocasiones, se presentan también tensiones por esfuerzo de corte vertical y tensiones normales directas que se deben a cargas axiales, sin embargo, su efecto es, por lo regular, mínimo a tal grado que es válido omitirlas. En ejes extremadamente cortos o en partes de ellos en los que no se generan torsión o flexión, es probable que predominen tales tensiones. Se sugiere el procedimiento siguiente para el diseño de un eje.

1. Determinar la velocidad de giro del eje o flecha.
2. Calcular la potencia o el par que va a transmitir el eje.
3. Determinar el diseño de los componentes transmisores de potencia u otros dispositivos que se pretenda montar en la flecha y especifique la ubicación que se necesita dar a cada dispositivo.
4. Precisar la ubicación de los cojinetes en los que se apoya el eje. Se supone que las reacciones en los cojinetes que soportan cargas radiales ejercen acción en el punto medio del cojinete. Si en el eje existen cargas de empuje, ósea axiales, se debe especificar qué cojinete debe diseñarse para que reaccione en contra de la carga de empuje. Por consiguiente se permite que el cojinete que no ejerce resistencia contra el empuje se desplace un poco en el sentido axial para asegurar que no se ejerza carga axial indeseable e inesperada sobre ese cojinete.

Otro concepto importante es que casi siempre se utilizan dos cojinetes para dar soporte a una flecha. Además, la longitud total de la flecha debe ser mínima para mantener las deflexiones en un nivel aceptable.

5. Proponer la forma general de la geometría para el eje o flecha, considerando de qué manera se mantiene en posición axialmente y como se lleva a cabo la transmisión de potencia a partir de cada elemento hacia el eje.

6. Calcular la magnitud del par que se observa en todos los puntos del eje. Se sugiere elaborar una gráfica de par.
7. Calcular las fuerzas que ejercen acción sobre el eje, tanto radial como axialmente.
8. Determinar las fuerzas radiales en componentes en sentidos perpendiculares, por lo regular tanto vertical como horizontalmente.
9. Calcular las reacciones en todos los cojinetes de soporte en cada plano.
10. Elaborar las gráficas completas de fuerza de corte y de momento de flexión para determinar la distribución de los momentos de flexión sobre el eje.
11. Elegir el material con que se va a fabricar el eje y especifique su condición: extruído en frío, con tratamiento térmico y demás.
12. Calcular una tensión de diseño adecuada, considerando la manera en que se aplica la carga, suave, de choque, sucesiva e inversa o de otro tipo.
13. Analizar cada punto crítico del eje para determinar el diámetro mínimo aceptable del eje para verificar la seguridad bajo aplicación de carga en cada punto. En general, los puntos críticos son numerosos e incluyen aquellos donde tiene lugar un cambio de diámetro, donde se generan los valores más altos de par y de momento de flexión y donde se presentan concentraciones de tensión.
14. Especificar las dimensiones finales para cada punto en el eje. Por lo regular, se utilizan los resultados del paso 13 a manera de parámetro, después se eligen los valores convenientes. Deben especificarse, a su vez, detalles como tolerancias, radio de los chaflanes, altura de los hombros, y dimensiones de los cuñeros. A veces, el tamaño y la tolerancia para el diámetro de un eje son dictados por el elemento que va a montarse ahí.

1.4.2 Engranés

Los engranes se pueden clasificar de muy diversas formas, una de ellas es la que utiliza Hall Allen, en su libro: "Teoría y problemas de elementos de máquinas". La clasificación es la siguiente:

a) Engranés rectos (cilíndricos)

Constituyen un medio conveniente para transmitir potencia entre ejes paralelos, con una relación de velocidades angulares constantes. El valor de la relación de velocidades es el mismo que se obtendría con dos cilindros imaginarios presionados uno contra el otro y girando sin deslizar en su línea de contacto (figura 1.4.1a).

b) Engranés helicoidales

Los engranes helicoidales se diferencian de los cilíndricos de dientes rectos en que tienen los dientes tallados en forma de una hélice alrededor de los cilindros primitivos, en lugar de ser paralelos al eje de rotación. Los engranes helicoidales se pueden utilizar para conectar ejes paralelos o no paralelos (figura 1.4.1b).

c) Engranés cónicos

Los engranes cónicos se utilizan generalmente para conectar ejes que se cortan y se pueden clasificar de acuerdo con la amplitud de su ángulo primitivo. Los que tienen el ángulo primitivo α menor que 90° son engranes cónicos externos. Los que tienen el ángulo primitivo de 90° se llaman coronas dentadas. Los que tienen el ángulo primitivo α mayor que 90° son los engranes cónicos internos. La suma de los ángulos primitivos de dos engranes cónicos acoplados es igual al ángulo formado por los dos ejes. Con respecto a un elemento del cono, los dientes pueden ser rectos o espirales (figura 1.4.1c).

d) Engranés con tornillos sin fin

Los engranes con tornillo sin fin se utilizan ampliamente para transmitir potencia con altas relaciones de velocidades entre ejes que no cortan, los cuales usualmente forman un ángulo recto. La transmisión por tornillo sin fin consta de un tornillo sin fin acoplado a un engrane, denominado a veces rueda dentada del tornillo sin fin. El tornillo puede tener uno, dos, tres o más filetes. El paso axial del tornillo sin fin es igual al paso circular del engrane. El avance es la distancia que avanza la hélice del tornillo sin fin a lo largo del eje, por cada revolución (figura 1.4.1d).



a) Engranés rectos (cilíndricos)



b) Engranés helicoidales



c) Engranés cónicos



d) Engranés con tornillos sin fin

Figura 1.4.1 Tipos de Engranés.

1.4.3 Transmisión por bandas y cadenas

Las bandas y cadenas representan los tipos principales de elementos flexibles para transmitir potencia. A diferencia de los impulsores de engranes, que requieren de distancias centrales espaciadas en forma reducida y precisa, los impulsores de banda y cadena son capaces de transmitir potencia entre flechas que se encuentran muy separadas. Además, la distancia central es inherentemente ajustable y no necesita ser tan precisa como para las transmisiones por engranes.

En general los impulsores de banda se utilizan donde las velocidades de giro son relativamente altas, como en la primera fase de reducción de un motor. La velocidad lineal de una banda es por lo regular de 2500 a 7000 pies por minuto. A velocidades más bajas, la tensión en la banda es demasiado alta para secciones transversales típicas de bandas. A velocidades más considerables, los efectos dinámicos como las fuerzas centrífugas, el chicoteo de la banda así como la vibración reducen la eficiencia del impulsor y su vida útil. Por lo general, una velocidad de 4000 pies/min resulta ideal.

Los impulsores de cadena se emplean casi siempre a velocidades más bajas, con los consecuentes pares de mayor magnitud. Los eslabones de cadenas de acero tienen una alta resistencia a esfuerzos de tracción para que sean capaces de soportar las considerables fuerzas que resultan en un par de alta magnitud. No obstante, a velocidades altas, el ruido, el impacto entre los eslabones de la cadena y los dientes de la rueda dentada así como la dificultad para brindar una lubricación adecuada se convierten en problemas severos.

a) Bandas

Una banda es un elemento flexible capaz de transmitir potencia que asienta en forma ajustada sobre un conjunto de poleas o poleas acanaladas (figura 1.4.2). Cuando se utiliza para reducción de velocidad, que es el caso más común, la polea acanalada más pequeña se monta en la flecha de alta velocidad como la flecha de un motor eléctrico. La polea de mayor tamaño se monta en la máquina que es impulsada. La banda se diseña de manera que gire alrededor de las dos poleas sin deslizarse.

En el mercado se disponen muchos tipos de bandas: *planas*, *dentadas*, *en V*, *bandas dobles en V* y *bandas múltiples en V*.

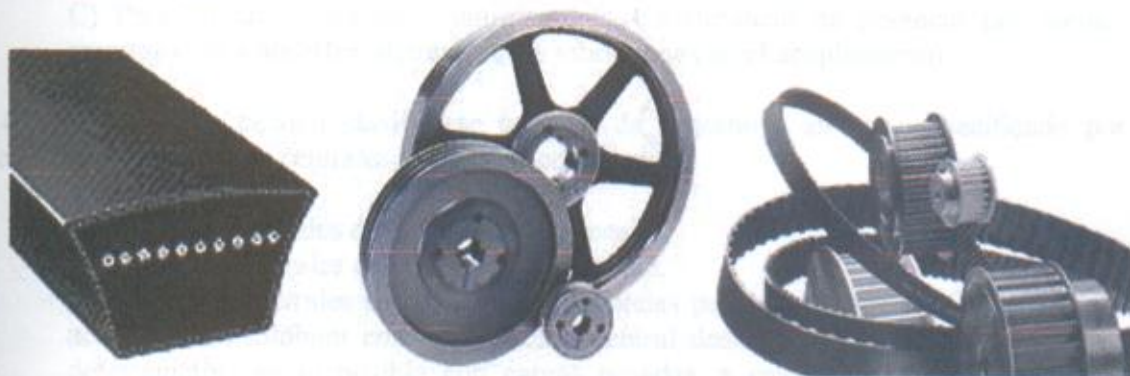


Figura 1.4.2 Poleas y bandas.

b) Cadenas

Una cadena es un elemento de transmisión de potencia que se fabrica como una serie de eslabones que se unen mediante pernos. El diseño proporciona flexibilidad mientras permite que la cadena transmita fuerzas de tracción cuya magnitud es considerable (figura 1.4.3). Cuando se transmite potencia entre flechas o ejes que giran, la cadena activa las ruedas dentadas que se enlazan, se conocen como catarinas o sprockets.



Figura 1.4.3 Cadenas y catarinas (sprockets).

1.4.4 Acoplamiento

La clasificación de los acoplamiento puede hacerse sobre la base de que los diseños sean rígidos o flexibles.

- 1) Acoplamiento rígidos: Como un acoplamiento de bridas, uno de compresión o un acoplamiento cónico. Este tipo de acoplamiento es recomendable para velocidades bajas en ejes alineados con precisión.
- 2) Acoplamiento flexibles: Como un acoplamiento flexible Falk, un acoplamiento Oldham, un acoplamiento tipo engranaje, un acoplamiento de rodillos o de cadena silenciosa, etc.

Los acoplamiento flexibles se utilizan:

- A) Para tener en cuenta pequeñas cantidades de desalineamiento no intencionales.
- B) Para suministrar un "extremo flotante" esto es, movimiento axial del eje.
- C) Para aliviar el choque, suministrando transferencia de potencia por medio de resortes o para absorber algunas de las vibraciones en el acoplamiento.

Los acoplamiento pueden clasificarse también de acuerdo a su uso, especificado por la relación entre las líneas centrales de los ejes conectados:

- a) Las líneas centrales de los ejes son colineales.
- b) Las líneas centrales de los ejes se intersectan.
- c) Las líneas centrales de los ejes son paralelas pero no colineales. (Puede usarse un acoplamiento Oldham con su elemento central deslizante este tipo de acoplamiento debe evitarse en lo posible con cargas pesadas, a causa del rozamiento debido al deslizamiento).

Como los acoplamientos rígidos pueden transmitir flexión en un eje, esto origina esfuerzos que pueden producir falla por fatiga. Es deseable, por consiguiente, proporcionar buen alineamiento y localizar el acoplamiento en donde el momento de flexión sea prácticamente nulo. Así tanto los acoplamientos rígidos como los flexibles, se analizan generalmente sólo por torsión. Aún cuando pueden compararse acoplamientos normalizados por los fabricantes, el análisis y el proporcionamiento de las diferentes partes ilustra el procedimiento de diseño aplicado a un elemento particular de una máquina.

1.4.5 Cojinetes o rodamientos

El fin que debe cumplir un cojinete es soportar una carga en tanto permite el movimiento relativo entre dos piezas de una máquina. Existe una extensa variedad de cojinetes en los que se pueden usar bolas o algún tipo de rodamiento entre las piezas fijas y las móviles. El tipo más común de cojinetes soporta una flecha giratoria, que resiste cargas radiales simples o una combinación de cargas radiales y axiales. Los componentes de un cojinete son el anillo o pista de rodamientos interno, el anillo o pista de rodamiento externo y las piezas giratorias. Por lo regular, la pista de rodamientos externo es fijo y se sostiene mediante la carcasa de la máquina. La pista interna de rodamientos es presionada contra la flecha giratoria y por tanto gira junto con ésta. Así, las bolas giran entre el anillo interno y el externo. La trayectoria de la carga es a partir de la flecha, hacia la pista de rodamientos interno, hacia las bolas, hacia la pista externa y finalmente hacia la carcasa. La presencia de las bolas permite un giro muy suave, con baja fricción de la flecha. La presencia de sellos, lubricante en exceso o cargas poco comunes hace que el coeficiente de fricción se incremente. La aplicación de cojinetes de rodamientos implica una selección, montaje, lubricación y protecciones adecuadas, con el fin de que el funcionamiento de los cojinetes sea satisfactorio. La selección de un cojinete de rodamientos se hace con base en catálogos. Desgraciadamente los catálogos de los diferentes fabricantes no emplean los mismos métodos de selección de rodamientos.

Tipos de rodamientos

En la tabla 1.4.1 se muestran los tipos de cojinetes, cada uno de ellos posee variantes tanto en el diseño como en la capacidad de carga radial, capacidad carga axial y capacidad de desalineación.

Tabla 1.4.1 Tipos de rodamientos [3]

Tipo de cojinete	Capacidad de carga radial	Capacidad de carga axial	Capacidad desalineación
Bola de hilera única	Buena	Aceptable	Aceptable
Bola de doble hilera	Excelente	Buena	Aceptable
Contacto angular	Buena	Excelente	Pobre
Rodamiento cilíndrico	Excelente	Pobre	Aceptable
Aguja	Excelente	Pobre	Pobre
Rodamiento esférico	Excelente	Aceptable / Buena	Excelente
Rodamiento ahusado	Excelente	Excelente	Pobre

a) Cojinete de bola de hilera única

En estos rodamientos la pista interna se presiona contra la flecha en el asiento del cojinete con un ajuste de interferencia pequeño para asegurar que gire junto con el eje o flecha. Las piezas esféricas giratorias, o bolas, giran dentro de una ranura profunda tanto en los anillos externos como en los internos.

El espaciamiento de las bolas se mantiene mediante dispositivos de retención. La ranura permite que se soporte una carga axial considerable porque se diseña con capacidad para soportar carga radial. La carga axial se aplica a un lado de la pista de rodamientos interna mediante un hombro en la flecha o eje. La carga pasa a lo largo de la ranura, a través de la bola, hacia el lado opuesto del anillo de bolas externo, y después hacia la carcasa. El radio de la bola es un poco más pequeño que el radio de la ranura para permitir el rodamiento libre de las bolas. En teoría el contacto entre una bola y la pista se da en un punto pero en realidad es un área circular pequeña debido a la deformación de las piezas. Como la carga es soportada en un área circular pequeña se presentan tensiones debidas al contacto muy altas a nivel local (figura 1.4.4a).

b) Cojinete de doble hilera

Si se le compara con el diseño de una sola hilera, agregar una segunda hilera de bolas incrementa la capacidad para soportar carga radial debido a que es mayor el número de bolas que soportan la carga. Por tanto, una mayor carga puede ser soportada por el mismo espacio o una carga específica puede ser soportada en un espacio más pequeño. El espesor más grande de los cojinetes de doble hilera suele afectar de manera adversa la capacidad de desalineación (figura 1.4.4b).

c) Cojinete de bolas de contacto angular

Si se le compara con el cojinete estándar de hilera única y ranura profunda, en un cojinete de contacto angular, un lado de cada pista de rodamientos es más alto para permitir su adaptación a cargas axiales considerables. En la figura 1.4.4c se muestra el ángulo que se prefiere de la fuerza resultante con cojinetes disponibles en el mercado de entre 15° y 45° .

d) Cojinetes de rodamientos cilíndricos

Sustituir las bolas esféricas con rodamientos cilíndricos con los cambios correspondientes en el diseño de los pistas de bolas, proporciona una mayor capacidad de carga radial. El patrón de contacto entre un rodamiento y su pista o collar es, en teoría una línea y adopta forma rectangular conforme las piezas se deforman bajo el efecto de una carga. Los niveles de tensión debida al contacto son más bajos que los que corresponden a cojinetes de bola de un tamaño equivalente, lo que permite que cojinetes más pequeños soporten una carga particular o que un cojinete de un tamaño específico soporte una carga mayor. La capacidad para soportar carga axial es pobre porque al lado de los rodamientos se le aplica cualquier carga axial y ello provoca frotamiento y no un verdadero movimiento giratorio (figura 1.4.4d).

e) Cojinete de aguja

Los cojinetes de aguja son en realidad cojinetes de rodamientos, pero el diámetro es mucho menor, como puede observarse en la figura 1.4.4e. Por lo común para que los cojinetes de aguja soporten una carga específica se requiere un espacio radial más pequeño que el que se necesita para cualquier otro tipo de cojinetes de contacto giratorio. Al igual que en otros cojinetes de rodamientos, la capacidad de carga axial y desalineación es pobre.

f) Cojinete de rodamiento esférico

El cojinete de rodamiento esférico es una forma de cojinete autoalineado se denomina así porque existe rotación relativa real de la pista de rodamientos externa con relación a los rodamientos y la pista de rodamientos interna cuando se presenta desalineación angular. Esto proporciona excelente especificación de la capacidad de desalineación en tanto se conservan las mismas especificaciones de capacidad de carga radial (figura 1.4.4f).

g) Cojinete de rodamiento ahusado

Los cojinetes de rodamientos ahusados están diseñados para soportar cargas de empuje sustanciales con cargas radiales altas, lo cual da por resultado excelentes especificaciones en ambos. Se utilizan con frecuencia en cojinetes de rueda para vehículos y equipo móvil en maquinaria de trabajo pesado a la que le son inherentes cargas de empuje altas (figura 1.4.4g).

h) Cojinetes para carga axial

Los cojinetes que se han mencionado hasta el momento se han diseñado para que soporten cargas radiales o una combinación de cargas radiales y axiales. Muchas aplicaciones de diseño mecánico exigen un cojinete que solo resista cargas axiales y en el mercado están disponibles numerosos tipos de cojinetes estándar de carga axial. Se utilizan las mismas piezas giratorias: bolas, rodamientos cilíndricos y rodamientos ahusados (figura 1.4.4h).



a) Cojinete de bola de hilera única



b) Cojinete de bola de doble hilera



c) Cojinete de bolas de contacto angular

Figura 1.4.4 Tipos de cojinetes.



d) Cojinetes de rodamientos cilíndricos



e) Cojinete de aguja



f) Cojinete de rodamiento Esférico



g) Cojinete de rodamiento ahusado



h) Cojinetes para carga axial

Figura 1.4.4 Tipos de cojinetes (continuación).

1.5 Actuadores

Hasta este momento se conocen las distintas configuraciones de los manipuladores y también las distintas formas en que se podría transmitir movimiento, ¿Pero quién proporciona este movimiento?, para contestar esta pregunta a continuación se describen los principales actuadores utilizados en robótica con el propósito de elegir el que mejor se adecue a las necesidades del manipulador.

Los actuadores tienen por misión generar el movimiento de los elementos del robot según las órdenes dadas por la unidad de control. A continuación se muestran los tipos de actuadores utilizados en robótica y una tabla 1.5.1 comparativa con sus características:

- a) *Neumáticos (cilindros y motores)*
- b) *Hidráulicos (cilindros y motores)*
- c) *Eléctricos (motores de DC, AC y paso a paso)*

Tabla 1.5.1 Características de actuadores para robots.

	<i>Neumático</i>	<i>Hidráulico</i>	<i>Eléctrico</i>
<i>Energía</i>	• Aire a presión (5 – 10 bar)	• Aceite mineral (50 – 100 bar)	• Corriente eléctrica (5-24 volt)
<i>Opciones</i>	• Cilindros • Motor de paletas • Motor de pistón	• Cilindros • Motor de paletas • Motor de pistón axiales	• Corriente continua • Corriente alterna • Motor paso a paso
<i>Ventajas</i>	• Baratos • Rápidos • Sencillos • Robustos	• Rápidos • Alta relación potencia-peso • Autolubrificantes • Estabilidad frente a cargas estáticas	• Precisos • Confiables • Fácil control • Sencilla instalación • Silenciosos
<i>Desventajas</i>	• Dificultad de control continuo • Instalación especial (compresor, filtros) • Ruidoso	• Difícil mantenimiento • Instalación especial (filtros, eliminación de aire) • Frecuentes fugas • Caros	• Potencia limitada

Algunas propiedades importantes de estos actuadores son:

- *Potencia*
- *Controlabilidad*
- *Peso y volumen*
- *Precisión*
- *Velocidad*
- *Mantenimiento*
- *Coste*

1.5.1 Actuadores neumáticos

A continuación se presentan las características más importantes de este tipo de actuadores:

- *Cilindros neumáticos*
 - Movimiento lineal de un émbolo debido a diferencia de presión
 - Pueden ser de simple efecto ó doble efecto (figura 1.5.1)
 - Posicionamiento en los extremos y dificultad en un posicionamiento continuo
 - Funcionamiento simple y un mantenimiento económico
 - Repetitividad inferior a otros tipos de actuadores
 - Adecuados para la manipulación de piezas pequeñas



Figura 1.5.1 Pistón simple efecto (izquierda) y pistón de doble efecto (derecha).

- *Motores rotativos*
 - Ligeros y compactos
 - Arranque y paradas muy rápidas
 - Velocidad y par variables
 - Control simple
 - Difícil control de posición
 - Motores de aletas rotativas
 - Simples
 - Dispositivos reductores para aumentar el par
 - Motores de pistones
 - Menor velocidad que los de aletas
 - Bajo nivel de vibración
 - Par elevado a bajas velocidades

1.5.2 Actuadores hidráulicos

A continuación se presentan las características más importantes de este tipo de actuadores:

- Funcionalmente similares a los neumáticos
- En cuanto al grado de compresibilidad, el del aceite es muy inferior al del aire lo que implica una mayor precisión en este tipo de sistemas
- Elevadas fuerzas y pares: cargas hasta de 200 kg.
- Mantenimiento no muy complejo
- Buena repetitividad (2.3 y 0.2 mm)
- No presentan problemas de refrigeración
- Cilindros: Iguales a los neumáticos

- Motores de aletas rotativas
 - Elevado par de arranque y rendimiento (90%)
 - Relativamente económicos
- Motores de pistones
 - Cilindrada variable
 - Construcción compleja

1.5.3 Actuadores eléctricos

Son los más usados en los robots industriales actuales debido a las siguientes características:

- Fáciles de controlar
- Sencillos pero de construcción delicada
- Precisos
- Alta repetitividad
- Más pesados que los neumáticos y los hidráulicos a igualdad de potencia
- Motores de corriente directa (CD)
 - *Controlados por inducido*
 - *Controlados por excitación*
- Motores de corriente alterna (CA)
 - *Síncronos*
 - *Asíncronos*
- Motores paso a paso

a) Principio de funcionamiento de motores de CD (figura 1.5.2)

- En una bobina cerrada, por la que circula corriente, sometida a un campo magnético se induce una fuerza que la hace girar.
- Para mantener la rotación en el mismo sentido es necesario conmutar el sentido de la corriente
- El campo magnético se produce en el estator
- Las bobinas se encuentran en el rotor
- Control por inducido: campo magnético constante y corriente variable
- Control por excitación: campo magnético variable y corriente constante
- Entrada → Tensión; Salida → Velocidad

Mejoras de los motores de CD

- Campos de excitación mediante imanes permanentes (con lo que se evitan fluctuaciones)
- Bobinado del rotor mediante espiras serigrafiadas (lo que implica una disminución de la inercia)
- Eliminación de escobillas (*brushless*): (lo que implica menos mantenimiento)
- *Servomotor*: Al cual se le aplica una entrada en tensión, obteniendo una salida de posición angular

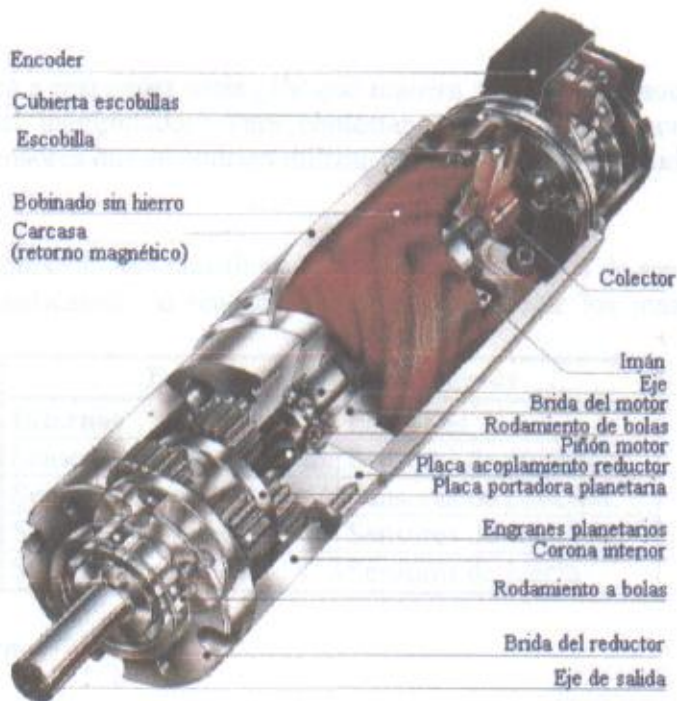


Figura 1.5.2 Esquema de motor CD seccionado con indicación de sus distintos componentes.

b) Motores de corriente alterna (CA)

A continuación se presentan las características más importantes de este tipo de actuadores:

- Inductor en el rotor: imanes permanentes
- Inducido en el estator: 3 devanados desfasados 120° eléctricos y alimentados con un sistema trifásico de tensiones
- Control de velocidad mediante variación de la frecuencia de la tensión de inducido → convertidor de frecuencia o variador de velocidad
- Sensor de posición continuo que detecta la posición del rotor
- No presentan problemas de mantenimiento ni de calentamiento
- Mayor potencia, a igualdad de peso, que los de corriente continua

c) Motores paso a paso

A continuación se presentan las características más importantes de este tipo de actuadores:

- El rotor, con polarización magnética constante, gira para orientar sus polos con el estator
- La polaridad del estator es variable, controlado por trenes de pulsos
- Por cada pulso, el rotor gira un número discreto de grados
- Aumento de la frecuencia de los pulsos progresivamente
- Ligeros y fiables
- Potencia y precisión bajas
- Giro de pinzas y mesas posicionadoras

1.6 Sensores

La siguiente pregunta a responder sería ¿De que manera se pueden conocer las posiciones de las articulaciones del manipulador? Para contestar a ella a continuación se presenta una descripción de los sensores que se podrían utilizar de acuerdo a las necesidades que presente el manipulador.

En la tabla 1.6.1 se presenta una clasificación de los distintos tipos de sensores que existen en el mercado, esta clasificación se realiza a fin de poder elegir los mas adecuados para el prototipo.

Internos	Externos
Sensores de presencia	Sensores de presencia
Sensores de posición	Sensores de posición
Sensores de velocidad	Sensores de velocidad
Sensores de fuerza	Sensores de fuerza

1.6.1 Sensores internos

- *Sensores de desplazamiento angular*
 - Con contacto
 - Codificadores de contacto
 - Potenciómetros
 - Sin contacto
 - Resolvers
 - Codificadores ópticos, magnéticos, inductivos y capacitivos
- *Sensores de desplazamiento lineal*
 - Con contacto
 - Regletas de contactos
 - Potenciómetros de desplazamiento lineal
 - Sin contacto
 - LVDT
 - Reglas magnéticas (*inductosyn*) y reglas ópticas
- *Sensores de posición angular*

A continuación una tabla comparativa donde se muestran los principales sensores de posición.

	Robustez mecánica	Rango dinámico	Resolución	Estabilidad térmica
<i>Encoder</i>	<i>Mala</i>	<i>media</i>	<i>Buena</i>	<i>buena</i>
<i>Resolver</i>	<i>Buena</i>	<i>buena</i>	<i>Buena</i>	<i>buena</i>
<i>Potenciómetro</i>	<i>Regular</i>	<i>mala</i>	<i>Mala</i>	<i>mala</i>

Potenciómetros.

Las desventajas que se tienen al utilizar potenciómetros son: Desgaste, ruido eléctrico y velocidad limitada

Codificadores ópticos (Encoders):

- Convierte movimiento en una secuencia de pulsos digitales
- Contando un solo bit o decodificando un conjunto de ellos, los pulsos se pueden convertir en medidas de posición relativas o absolutas
- Configuración lineal o rotativa (siendo ésta última la más utilizada)
- Los rotativos se componen de un disco plástico (transparente), al cual se acopla el eje cuya posición se quiere medir, con una serie de marcas (opacas) colocadas radialmente y equidistantes entre sí; de elementos emisores de luz y fotorreceptores
- A medida que el eje gira, el haz de luz se interrumpe con las marcas y se generan pulsos en el receptor. Contando estos pulsos es posible conocer la posición del eje.
- Problema de detección del sentido del giro

Encoders incrementales (figura 1.6.1 y figura 1.6.2):

- Miden posiciones relativas
- Dos trenes de pulsos desfasados 90° para determinar el sentido de giro
- Señal de paso por cero
- Hasta 100,000 pulsos por vuelta
- Electrónica adicional

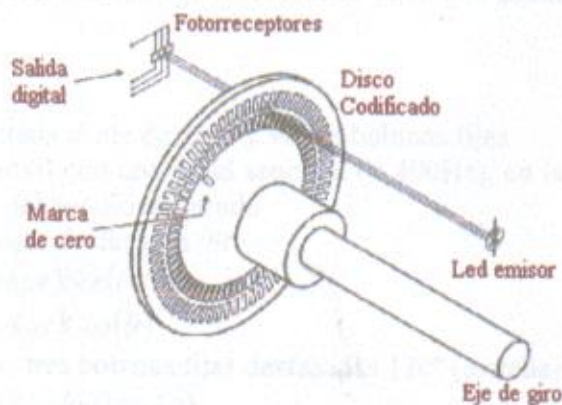


Figura 1.6.1 Funcionamiento de un encoder incremental.

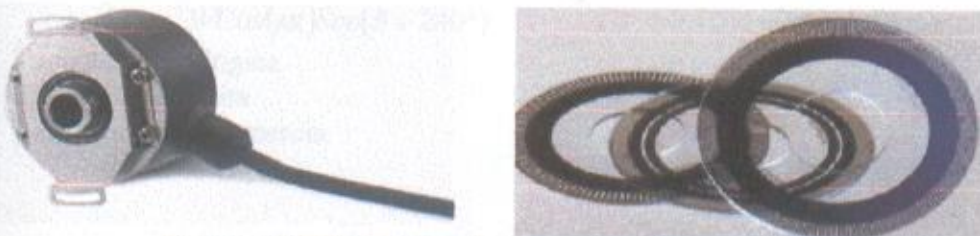


Figura 1.6.2 Imagen real de encoders incrementales.

Encoders absolutos (figura 1.6.3 y figura 1.6.4):

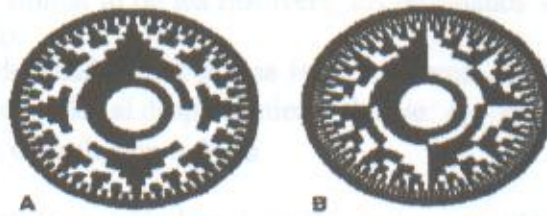


Figura 1.6.3 Encoders absolutos; A – Código gray, B – Código binario.

- Miden posiciones absolutas
- El disco transparente se divide en sectores
- Cada sector posee un código binario único
- Resolución fija

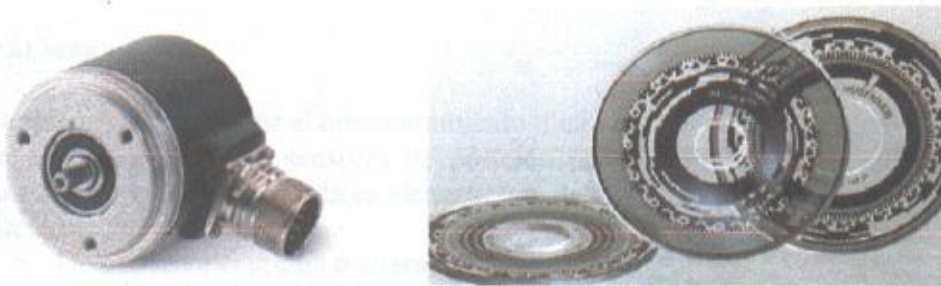


Figura 1.6.4 Imágenes reales de encoders absolutos.

Resolvers / Sincroresolvers

Una bobina móvil conectada al eje de giro, y varias bobinas fijas

Al excitar una bobina móvil con una señal senoidal ($\approx 400\text{Hz}$), en las bobinas fijas se inducen tensiones que dependen del ángulo de girado

Resolvers: dos bobinas fijas desfasadas 90° :

- $V_1 = V \text{Sen}(\omega t) \text{Sen}(\theta)$
- $V_2 = V \text{Sen}(\omega t) \text{Cos}(\theta)$

➤ **Sincroresolvers:** tres bobinas fijas desfasadas 120° (estrella):

- $V_{13} = \sqrt{3}V \text{Cos}(\omega t) \text{Sen}(\theta)$
- $V_{32} = \sqrt{3}V \text{Cos}(\omega t) \text{Sen}(\theta + 120^\circ)$
- $V_{21} = \sqrt{3}V \text{Cos}(\omega t) \text{Sen}(\theta + 240^\circ)$

- Tecnología analógica
- Resolución infinita
- Bajo momento de inercia

Reglas Magnéticas (Inductosyn):

- Funcionamiento similar al de los resolvers; los devanados secundarios son móviles y el primario es fijo
- Al alimentar el devanado fijo con una tensión alterna, en los secundarios se inducen tensiones proporcionales al desplazamiento del eje: $e_1 = E \text{Sen}(at)$; $e_2 = E \text{Sen}(at + d)$
- Bajo rozamiento y resolución infinita

Transformador diferencial de posición lineal (LVDT, siglas en inglés):

- Núcleo ferromagnético, unido al eje cuyo desplazamiento se puede medir, entre un devanado primario y dos devanados secundarios
- Al mover el núcleo, la tensión alterna del primario induce tensiones diferentes en los secundarios (aumenta en uno y disminuye en el otro), que son proporcionales al desplazamiento de eje
- Alta linealidad y repetitividad, resolución infinita y bajo rozamiento

Medida de la velocidad:

- Necesarios para mejorar el comportamiento dinámico de los actuadores
- Se pueden utilizar los sensores de posición para medir la velocidad, teniendo en cuenta el tiempo que se tarda en alcanzar una determinada posición
- *Tacómetros*
 - De corriente continua o alterna
 - El rotor, dotado de imán permanente y unido al eje del cual se quiere medir la velocidad, induce una tensión en el estator que es proporcional a la velocidad de giro
 - Resoluciones de 10mV/rpm
- *Convertidores frecuencia-tensión:*
 - Tecnología electrónica
 - Producen una tensión de salida proporcional a la frecuencia de un tren de pulsos, que pueden provenir de un sensor de posición digital, o bien de un sensor de posición analógico a través de una conversión analógico-digital

En la tabla 1.6.2 se resumen los tipos de sensores internos usados en robots.

Tabla 1.6.2 Tipos de sensores internos de robots.

Presencia	Posición		Velocidad
<ul style="list-style-type: none"> • Inductivo • Capacitivo • Presencia • Efecto hall • Célula Reed • Óptico • Ultrasonido • Contacto 	<i>Analógicos</i>	<i>Digitales</i>	<ul style="list-style-type: none"> • Tacogeneratriz
	<ul style="list-style-type: none"> • Potenciómetros • Resolver • Síncro • Inductosyn • LVDT 	<ul style="list-style-type: none"> • Encoders absolutos • Encoders incrementales • Regla óptica 	

1.6.2 Sensores externos

Sensores de presencia

- **Mecánicos:** fines de carrera
 - Problemas de mantenimiento: mecánico y desgaste de contactos
- **Ópticos**
 - Emisor y receptor en un único encapsulado
 - Emisor y receptor independientes
- **Inductivo**
 - Detección de variación de consumo debido a corrientes de Foucault
 - Sólo sirve para materiales metálicos
- **Capacitivo**
 - Medida de variación de capacidad
 - Materiales metálicos y no metálicos

Sensores de proximidad

- **Efecto Hall**
 - Mide variación de campo magnético
 - Sólo materiales ferromagnéticos
- **Célula Reed (láminas)**
- **Ultrasonidos**

Medida de distancias

- **Ultrasonidos**
 - Rango de 15cm a 20m
 - Poca precisión:
 - Dependiendo de las condiciones ambientales
 - Incertidumbre de la procedencia del eco
- **Láser**
 - Triangulación
 - Cortas distancias (mm a 2m)
 - Precisión no uniforme en el rango de medida
 - Medida del tiempo de propagación
 - Medida del tiempo de vuelo
 - Grandes distancias (>50m)
 - Medida de desfase
 - Distancias cortas y medias (20cm a 100m)

1.7 Control de los manipuladores

Como requisitos para poder llevar el manipulador de una posición a otra se deben tomar en cuenta los siguientes aspectos.

1.7.1 Descripción de la orientación y posición

Una preocupación constante en el estudio de la robótica es la localización de los objetos en el espacio tridimensional. Estos objetos son los eslabones del manipulador, las partes y herramientas con las cuales trata, y otros objetos del ambiente del manipulador. Estos objetos están descritos por solo dos atributos: Su posición y su orientación. Naturalmente, un tema de inmediato interés es la manera en que se representan estas cantidades y manipularlas matemáticamente.

Para describir la posición y orientación de un cuerpo en el espacio siempre se sujeta un sistema de coordenadas, o marco, rígidamente al objeto. Entonces se precede a describir su posición y orientación de este marco con respecto a algún sistema de referencia (figura 1.7.1).

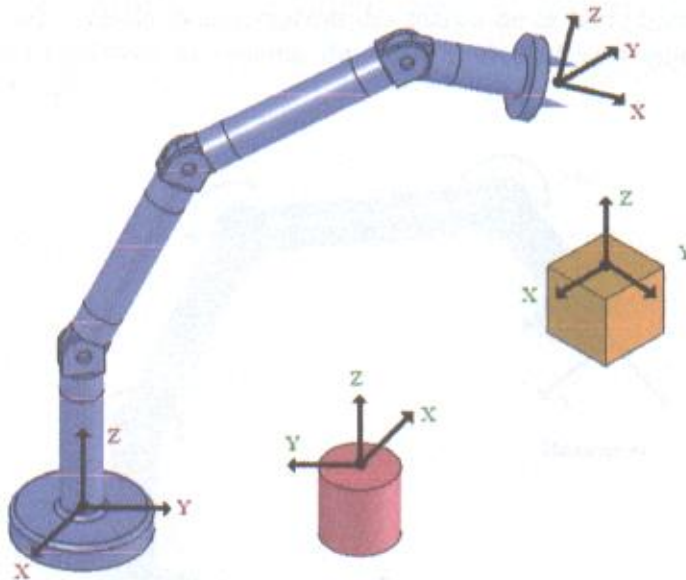


Figura 1.7.1 Los sistemas de coordenadas o marcos están referenciados al manipulador y a los objetos en el ambiente de trabajo.

Ya que cualquier cuadro puede servir como sistema de referencia en cual expresar la posición y orientación de un cuerpo, frecuentemente se piensa en la *transformación o cambio* de estos atributos de un cuerpo de un sistema a otro. Estas particularidades se estudian en lo que se conoce como descripción y transformación espacial.

1.7.2 Cinemática directa de los manipuladores

La cinemática es la ciencia del movimiento la cual trata al movimiento sin considerar las fuerzas que los provocan. En la ciencia de la cinemática se estudia la posición, velocidad, aceleración y todas las derivadas de orden mayor de las variables de posición. Por lo tanto el estudio de la cinemática de los manipuladores se refiere a todas las propiedades geométricas y en el tiempo del movimiento. Los manipuladores consisten en eslabones casi rígidos los cuales están conectados con uniones (articulaciones) que permiten un movimiento relativo de los eslabones vecinos. Estas articulaciones están usualmente instrumentadas con sensores de posición los cuales permiten medir la posición relativa de los eslabones.

El número de grados de libertad (GDL) que posee un manipulador es el número de variables de posición independientes que debería tener para localizar todas las partes del mecanismo. Este es un término general usado para cualquier mecanismo.

En el final de la cadena de eslabones, de los cuales está compuesto el manipulador, se encuentra el efector final. Dependiendo de la aplicación del robot el efector final pueden ser unas pinzas, herramientas para soldadura, un electro magneto, u otro dispositivo. Generalmente se describe la posición del manipulador dando la descripción del marco de la herramienta, la cual está fija al efector final, y ésta es relativa al sistema de la base el cual está sujeto a la base fija del manipulador (figura 1.7.2).

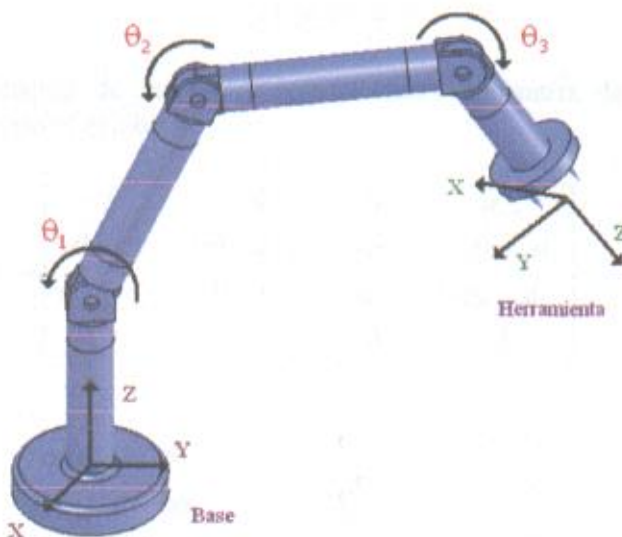


Figura 1.7.2 Las ecuaciones de la cinemática describen el marco de la herramienta relativo al sistema de la base en función de las variables de unión.

Un problema básico en el estudio de la manipulación mecánica es la cinemática directa. Esto es, el problema geométrico estático de calcular la posición y orientación del efector final del manipulador. Específicamente, dados un conjunto de ángulos de las articulaciones, el problema de la cinemática directa es calcular la posición y orientación del marco de la herramienta relativo al marco base.

Según la representación de Denavit - Hartenberg (DH), escogiendo adecuadamente los sistemas de coordenadas asociados a cada eslabón, es posible pasar de uno al siguiente mediante cuatro transformaciones básicas que dependen exclusivamente de las características geométricas de los eslabones, en la siguiente figura 1.7.3 se muestra una asignación de estos sistemas de coordenadas.

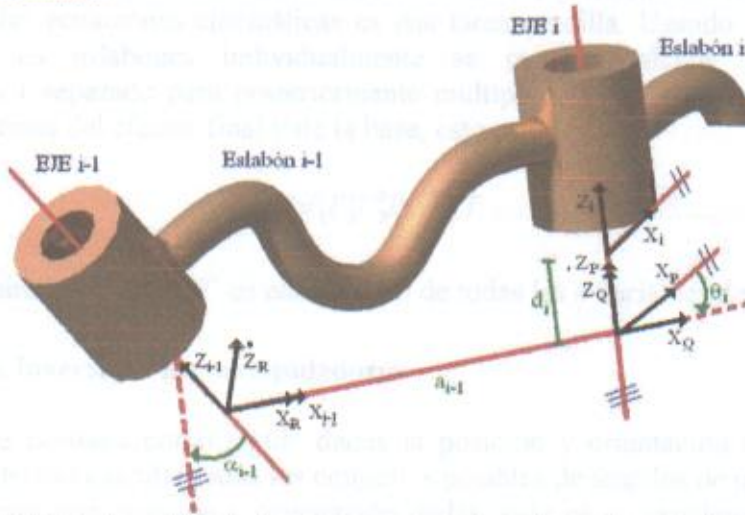


Figura 1.7.3 Asignación de los sistemas coordenados a los eslabones i-1 e i.

$${}^{i-1}T = {}^{i-1}T_R {}^R T_Q {}^Q T_P {}^P T_i \dots\dots\dots(7.1)$$

Realizando la multiplicación de las matrices se obtiene la matriz de transformación que relaciona el eslabón i-1 con el eslabón i.

$${}^{i-1}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots\dots\dots(7.2)$$

en donde:

$$\begin{aligned} \cos(\theta_i) &= c\theta_i & \sin(\theta_i) &= s\theta_i \\ \cos(\alpha_i) &= c\alpha_i & \sin(\alpha_i) &= s\alpha_i \end{aligned}$$

a_i = la distancia de Z_i a Z_{i+1} medida a lo largo de X_i .

α_i = el ángulo entre Z_i y Z_{i+1} medido sobre X_i .

d_i = la distancia de X_{i-1} a X_i medida a lo largo de Z_i .

θ_i = el ángulo entre X_{i-1} y X_i medido sobre Z_i .

Por lo tanto un robot puede ser descrito cinemáticamente al dar los valores de cuatro cantidades para cada eslabón que lo compone. Dos de estos valores describe el eslabón por sí mismo y los dos restantes su conexión con un eslabón vecino. En el caso de una articulación

de rotación, θ_i es llamada la variable de articulación, y las tres cantidades restantes son parámetros fijos del eslabón. Para articulaciones prismáticas d_i es la variable de la articulación y por lo tanto las tres cantidades restantes son parámetros fijos.

Una vez que se han establecido los sistemas de coordenadas y encontrado los parámetros DH, el desarrollo de las ecuaciones cinemáticas es una tarea sencilla. Usando los valores de los parámetros de los eslabones individualmente se pueden calcular las matrices de transformación por separado para posteriormente multiplicarlas y encontrar una matriz que relacione los sistemas del efector final y de la base, esto es:

$${}^0_N T = {}^0_1 T {}^1_2 T {}^2_3 T \dots {}^{N-1}_N T \dots \dots \dots (7.3)$$

Esta matriz de transformación ${}^0_N T$ es una función de todas las n variables del manipulador.

1.7.3 Cinemática inversa de los manipuladores

Este problema se presenta como sigue: dadas la posición y orientación del efector final del manipulador, se desean calcular todos los conjuntos posibles de ángulos de unión que podrían ser usados para obtener esta posición y orientación dadas. Este es un problema fundamental en el uso práctico de los manipuladores (figura 1.7.4).

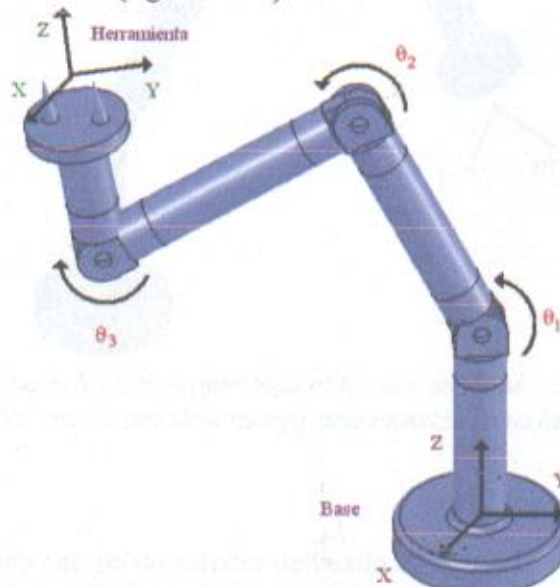


Figura 1.7.4 Para una posición y orientación del marco de la herramienta, los valores de las variables de unión pueden ser calculados usando la cinemática inversa.

El problema de la cinemática inversa no es tan simple como el de la cinemática directa. Debido a que las ecuaciones cinemáticas son no lineales, su solución no siempre es fácil o aún posible en una forma cerrada. También, surgen preguntas de la existencia de una solución o incluso de soluciones múltiples.

La existencia o no existencia de una solución cinemática define el espacio de trabajo del manipulador. La falta de una solución significa que el manipulador no puede alcanzar la posición y orientación deseadas porque se encuentra fuera de su espacio de trabajo.

1.7.4 Velocidades, fuerzas estáticas y singularidades

Además de tratar con los problemas de la posición estática, podríamos desear el analizar al manipulador en movimiento. Frecuentemente al realizar el análisis de la velocidad de un mecanismo es conveniente definir una matriz llamada el **Jacobiano** del manipulador. El Jacobiano especifica un mapeo de las velocidades en el espacio de las articulaciones hacia el espacio cartesiano (figura 1.7.5). La naturaleza de este mapeo cambia de acuerdo a la configuración del manipulador. En ciertos puntos, llamados singularidades, el mapeo es no invertible. El entendimiento de este fenómeno es importante para diseñadores y usuarios de manipuladores. Los manipuladores no siempre se mueve a través del espacio, en algunas ocasiones también requieren tener contacto con una pieza de trabajo o una superficie y aplicar una fuerza estática. En este caso surge un problema: ¿Dados un momento y fuerza de contacto deseados, que conjunto de pares se requieren en las articulaciones para generarlos? Una vez más, la matriz Jacobiana del manipulador es aplicada en la solución de este problema.

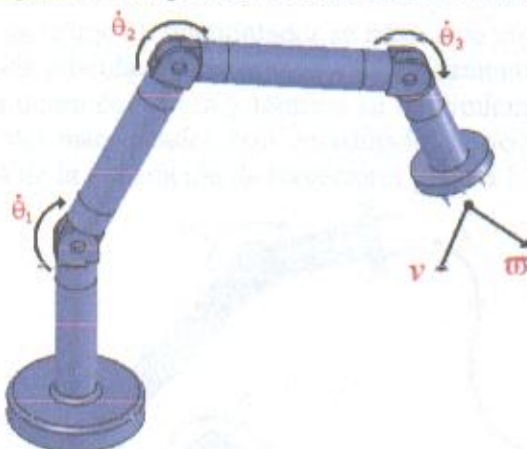


Figura 1.7.5 La relación geométrica entre los ángulos y la velocidad del efector final puede ser descrita en una matriz llamada Jacobiana.

1.7.5 Dinámica

La dinámica es un inmenso campo de estudio dedicado al estudio de las fuerzas requeridas para llevar a cabo un movimiento. Para acelerar un manipulador de su posición de descanso, un conjunto complejo de pares deben ser aplicados por los actuadores en las articulaciones. La forma exacta de las funciones requeridas del par del actuador dependen de atributos espaciales y temporales de la ruta tomada por el efector final, como también del peso de los eslabones y la carga, fricción en las uniones, etc. Un método para controlar un manipulador para que siga una ruta deseada involucra calcular estos pares de los actuadores usando ecuaciones dinámicas del movimiento del manipulador (figura 1.7.6). Un segundo uso de las ecuaciones de movimiento dinámicas es en simulación. Mediante la reformulación de las ecuaciones dinámicas la aceleración es calculada como una función del par del actuador, es posible simular como se podría mover un manipulador bajo la aplicación de un conjunto de pares.

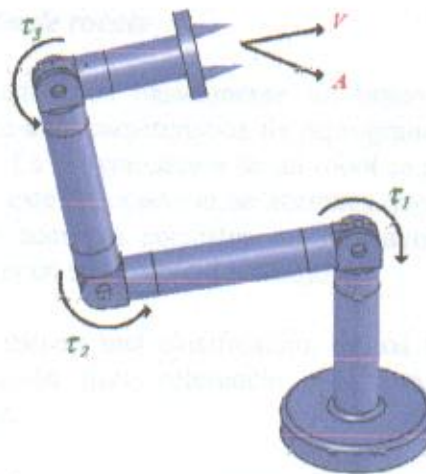


Figura 1.7.6 La relación entre los pares aplicados por los actuadores y el movimiento resultante del manipulador están descritos en las ecuaciones dinámicas del movimiento.

1.7.6 Generación de trayectoria

Una manera común de provocar que un manipulador se mueva de un punto a otro de una manera controlada es hacer que cada articulación se mueva específicamente mediante una función del tiempo. Comúnmente, cada unión comienza y termina su movimiento al mismo tiempo, así que parece que el movimiento del manipulador está coordinado. Exactamente como calcular estos movimientos es el problema de la generación de trayectoria (figura 1.7.7).

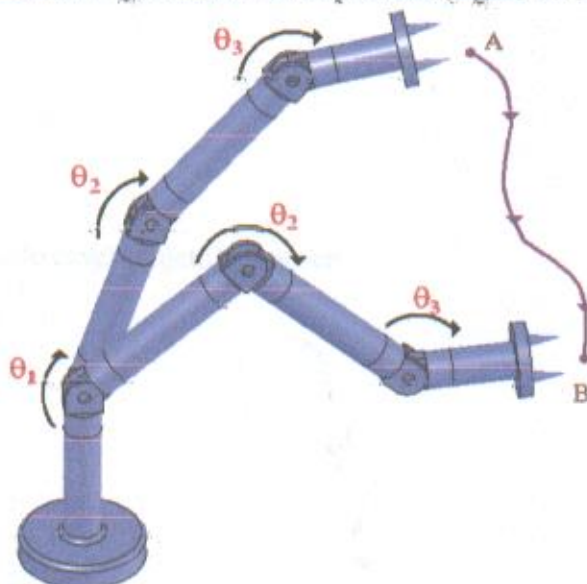


Figura 1.7.7 Para mover el efector final a través del espacio de un punto A al punto B se debe calcular una trayectoria a seguir para cada unión.

1.8 Lenguajes de programación de robots

Debido a que un robot industrial es básicamente un brazo manipulador multifuncional reprogramable, es precisamente esta característica de reprogramabilidad lo que le permite la realización de múltiples tareas. La programación de un robot se puede definir como el proceso mediante el cual se le indica a éste la secuencia de acciones que debe llevar a cabo durante la realización de su tarea. Estas acciones consisten en su mayor parte el moverse a puntos establecidos y manipular objetos en su espacio de trabajo [2].

Existen varios criterios para realizar una clasificación de los métodos de programación de robots. La siguiente clasificación hace referencia al sistema empleado para indicar la secuencia de acciones a realizar.

a) Programación por guiado

La programación por guiado o aprendizaje consiste en hacer que el robot realice la tarea al tiempo que se registran las configuraciones adoptadas, para su posterior repetición de manera automática.

b) Programación textual

Este método permite indicar la tarea al robot mediante el uso de un lenguaje de programación específico. La programación textual puede ser clasificada en tres niveles: robot, objeto y tarea.

Entre los lenguajes del primer nivel de programación (robot) se pueden encontrar:

- AL
- AML
- LM
- VAL II
- V+
- RAPID

Como ejemplos del segundo nivel (objeto) se tienen:

- LAMA
- AUTOPASS
- RAPT

1.9 Control de motores

De acuerdo a la información presentada en páginas anteriores se ha decidido utilizar motores de corriente continua cuyas características son:

- Acoplamiento reductor de tornillo sin fin y corona.
- Ángulo recto.

Y además en ellos es posible realizar el control PI y PID, el cual se describe a continuación

1.9.1 Control PI y PID

El controlador es un instrumento que sustituye al operador humano en su tarea de mantener una variable del proceso en torno a un valor deseado, mediante actuaciones sobre otra variable denominada "manipulada", para el desarrollo de este proyecto se ha optado por emplear el control PI y PID. En la figura 1.9.1 se muestra un diagrama a bloques del control PI.

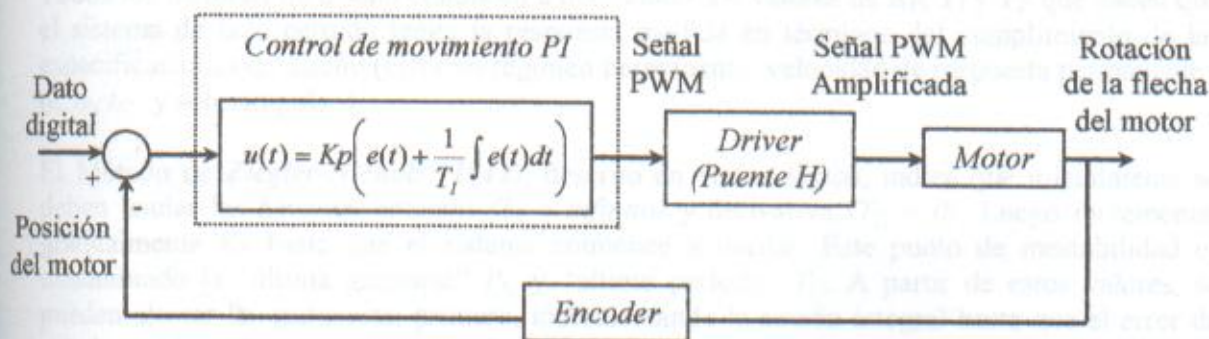


Figura 1.9.1 Diagrama a bloques para un control PI

Los controladores normalmente permiten la operación en modo "manual" dejando al operador exterior elegir el valor de la variable manipulada, y también pueden operar en modo automático efectuando modificaciones sobre la variable manipulada de acuerdo con una leyes de actuación. En este caso las leyes de control que se aplican son del tipo PI y PID. En la figura 1.9.2 se muestra un diagrama a bloques de un control PID.

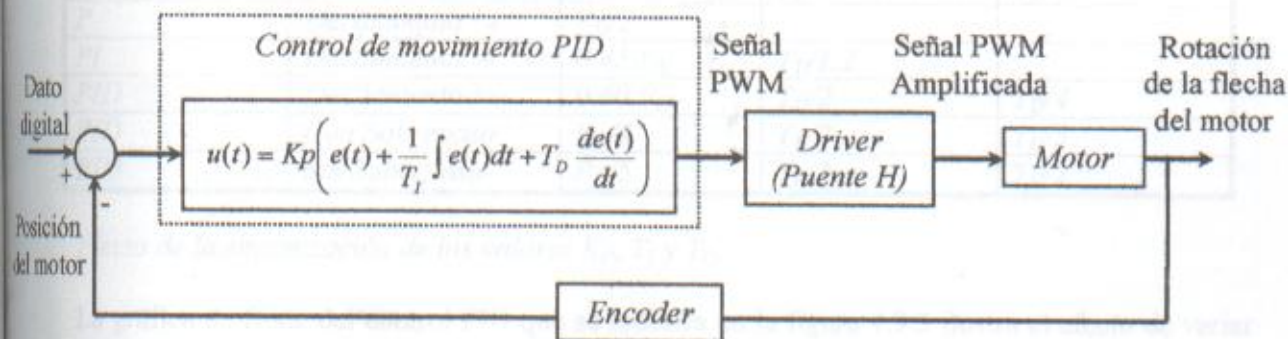


Figura 1.9.2 Diagrama a bloques para un control PID

1.9.2 Sintonización de las constantes K_P , T_I y T_D

Se asume un sistema realimentado en forma canónica, es decir, con realimentación unitaria. Uno de las estructuras de controladores más populares se denomina Controlador PID debido a que utiliza acciones de tipo Proporcional, Integral y Derivativo. La relación temporal entre la variable manipulada $u(t)$ y el error $e(t)$, para un controlador PID viene dada por:

$$u(t) = K_P \left(e(t) + \frac{1}{T_I} \int e(t) dt + T_D \frac{de(t)}{dt} \right) \dots\dots\dots (9.1)$$

O bien por su equivalente en el plano s

$$U(s) = K_P \left(1 + \frac{1}{T_I s} + T_D s \right) E(s) \dots\dots\dots (9.2)$$

Todos los métodos de diseño conducen a determinar los valores de K_P , T_I y T_D que hacen que el sistema de lazo cerrado tenga la respuesta medida en términos del cumplimiento de las especificaciones de diseño (error en régimen permanente, velocidad de respuesta permanente - t_P , t_R , t_S - y sobreimpulso).

El Método de **Ziegler-Nichols (1942)**, descrito en varios textos, indica que inicialmente se deben anular las acciones integral ($T_I = \text{infinito}$) y derivativa ($T_D = 0$). Luego incrementar gradualmente K_P hasta que el sistema comience a oscilar. Este punto de inestabilidad es denominado la "última ganancia" P_U y "último período" T_U . A partir de estos valores, se pueden ajustar las ganancias; primero, incrementando la acción integral hasta que el error de estado estable (Desviación) se anule, posteriormente se incrementa la acción derivativa lo suficiente para que la salida (Valor Medido) no muestre signos de oscilación. Por último se reduce el porcentaje de banda proporcional hasta obtener la mejor respuesta (menor sobrepaso y oscilación). En la tabla 1.9.1 se muestran los cálculos que se deben realizar para cada una de las diversas modalidades de control:

Tabla 1.9.1 Sintonización de los valores K_P , T_I y T_D .

Acción	Actuación	K_P	T_I	T_D
P	Decaimiento $1/4$	$P_U/2$		
PI	Decaimiento $1/4$	$0.45 P_U$	$T_U/1.2$	
PID	Decaimiento $1/4$	$0.60 P_U$	$T_U/2$	$T_U/4$
PID	Con Sobrepaso	$P_U/3$	$T_U/2$	$T_U/3$
PID	Sin Sobrepaso	$P_U/5$	$T_U/3$	$T_U/2$

Efecto de la sintonización de los valores K_P , T_I y T_D .

La gráfica de Bode del control PID que se muestra en la figura 1.9.3 ilustra el efecto de variar únicamente el factor proporcional (K_P). Como se puede observar, cuando aumenta K_P no solamente se aumenta el factor proporcional sino que también se baja la frecuencia en la cual

el factor de la integración cesa su eficacia y se levanta la frecuencia en la cual el factor diferencial comienza a incrementar su efecto.

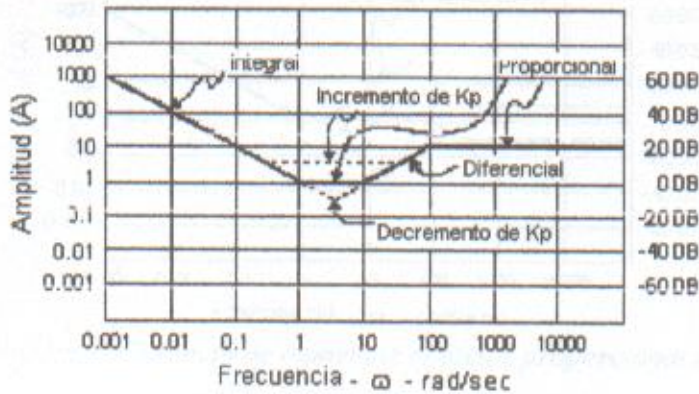


Figura 1.9.3 Efecto de aumentar el factor proporcional K_p .

Ajustar inadecuadamente el factor de la integración (K_I , donde $K_I = K_p/T_I$) afecta no sólo el aumento de la frecuencia baja, sino que cambia la frecuencia en la cual el factor proporcional (K_p) llega a ser eficaz. Sería lo más deseable levantar K_I tan arriba como sea posible, pero cuanto más alto sea, se introduce un desplazamiento negativo de fase en el motor. Esto reduce el margen de la fase del motor. En la figura 1.9.4 se ilustra el efecto que provoca variar el factor de integración K_I .

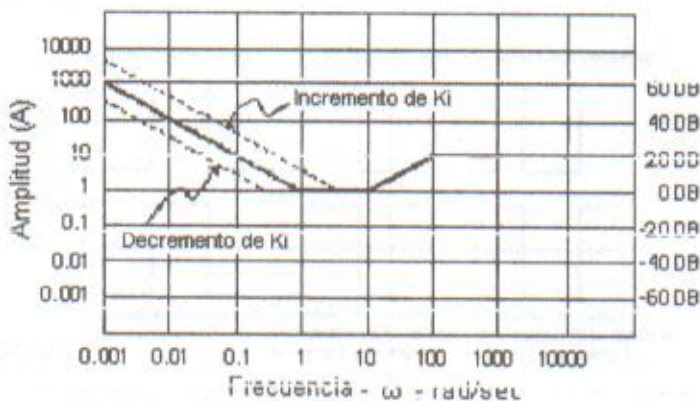


Figura 1.9.4 Efecto de aumentar el factor proporcional K_I .

Al aumentar el factor diferencial (K_D , donde $K_D = K_p T_D$) baja la frecuencia en la cual tiene más impacto el factor proporcional (K_p). Esto introduce el desplazamiento positivo de fase en el factor proporcional, al mismo tiempo que compensa el desplazamiento de fase negativo del factor K_I de la integración, de tal modo que se mejora el margen de la fase. Sin embargo, tiene el efecto indeseable de aumentar la frecuencia alta, de hacer el motor más ruidoso y sensible a efectos indeseables de resonancias naturales. En la figura 1.9.5 se ilustra el efecto que provoca variar el factor de diferencial K_D .

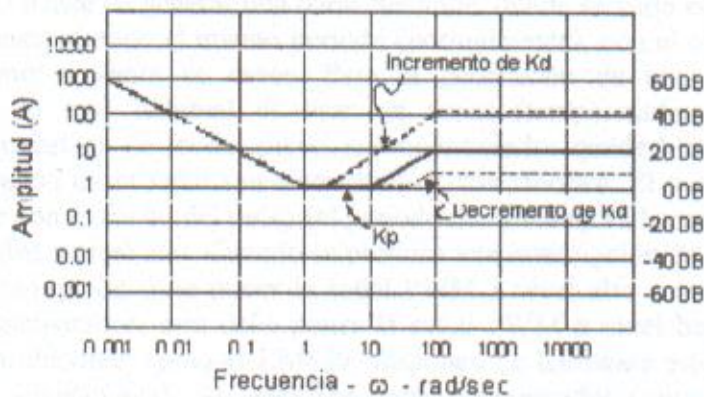


Figura 1.9.5 Efecto de aumentar el factor proporcional K_D .

1.9.3 Tipos de control de potencia

Para realizar el control de la potencia del motor existen básicamente tres grandes métodos:

- *Modulación por ancho de pulso* (PWM - Pulse Width Modulation), en corriente continua (ver figura 1.9.6)
- *Modulación de la frecuencia del pulso* (PFM - Pulse Frequency Modulation), en corriente continua (ver figura 1.9.6)
- *Rectificación controlado por silicio* (SCR - Silicon Controlled Rectifier), en corriente alterna

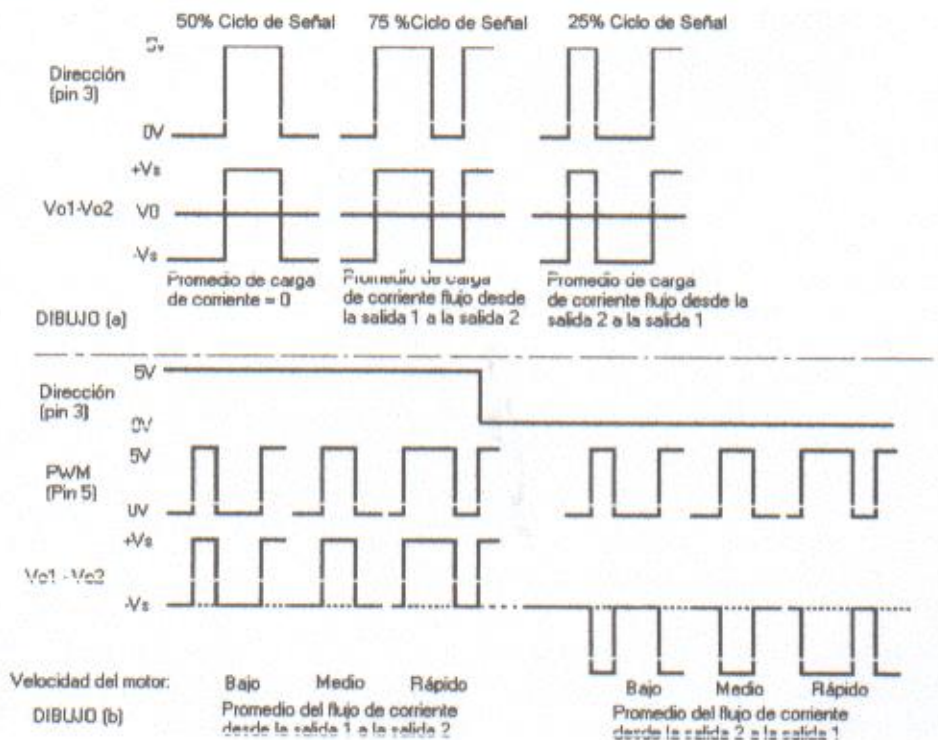


Figura 1.9.6 Forma de onda PWM.

El sistema PWM consiste en generar una onda cuadrada, donde se varía el tiempo que el pulso está a nivel alto, manteniendo el mismo periodo (normalmente), con el objetivo de modificar la posición del motor según se desee. Para la generación de una onda PWM en un microcontrolador, lo más habitual es usar un timer (temporizador) y un comparador (interrupciones asociadas), de modo que el microcontrolador quede libre para realizar otras tareas, y la generación de la señal sea automática y más efectiva. El mecanismo consiste en programar el timer con el ancho del pulso (el periodo de la señal) y al comparador con el valor de duración del pulso a nivel alto. Cuando se produce una interrupción de sobreflujo del timer, la subrutina de interrupción debe poner la señal PWM a nivel alto y cuando se produzca la interrupción del comparador, esta debe poner la señal PWM a nivel bajo. En la actualidad muchos microcontroladores, como el LM629, disponen de hardware específico para realizar esta tarea, eso sí, consumiendo los recursos antes mencionados (timer y comparador). El control de velocidad mediante PWM consiste en utilizar una señal modulada para la entrada del puente H que controla el motor. Lo que se hace es poner la señal de activación a nivel activo durante un tiempo, que es el ciclo de trabajo, e inactiva durante el tiempo que reste para completar un periodo fijo, por ejemplo de 5ms, si la señal está activa durante los 5ms, entonces no está inactiva nunca y el motor va al 100% de su velocidad. Lo ideal para los motores es usar un PWM con una frecuencia de más de 1KHz (así no se descarga la bobina y los picos de corriente son menores, es por tanto más efectivo), pero el problema es que a esta frecuencia los motores pitan, pues la frecuencia está dentro de la banda audible. No pasa nada porque piten, pero es molesto. Por ello se suelen utilizar frecuencias de más de 20KHz.

3.1.3 Definición de los ejes de movimiento del manipulador

El diseño preliminar del manipulador se muestra en la Figura 3.1.1.



Figura 3.1.1. Vista preliminar del Manipulador

En la figura anterior se muestran los tres grados de libertad, con los cuales cuenta el manipulador. Para poder definir las medidas del manipulador se han considerado las dimensiones de un robot humano desde el hombro, es decir, se ha dividido la distancia de la siguiente forma, dos quintos partes de la medida total corresponden al segmento establos y las dos quintas partes restantes corresponden al brazo establos.

CAPÍTULO 2. DISEÑO DEL PROTOTIPO

2.1 Diseño mecánico

En comparación con los robots didácticos existentes en el mercado, el diseño que se propone realizar pretende contar con las siguientes características

- Medidas semejantes a las de un brazo humano.
- Que sea una herramienta en la enseñanza de la robótica y no solo un simple juguete didáctico.
- Que cuente con un error en la posición del efector final máximo de 10mm.

2.2.1 Selección de la configuración

Después de haberse revisado las distintas configuraciones que existen para los manipuladores se ha decidido elegir la configuración articular sobre las otras presentadas anteriormente, la razón por la cual se ha elegido es la siguiente:

- Es la configuración que más se asemeja a un brazo humano.
- Su estructura mecánica es la que interviene menos en el área de trabajo.

2.1.2 Diseño estructural (preliminar) del manipulador

El diseño preliminar del manipulador se muestra en la figura 2.1.1



Figura 2.1.1 Vista preliminar del Manipulador.

En la figura anterior se muestran los tres grados de libertad, con los cuales cuenta el manipulador. Para poder definir las medidas del manipulador se han considerado las dimensiones de un brazo humano desde el hombro hasta la muñeca. Se ha dividido la distancia de la siguiente forma: tres quintas partes de la medida total corresponden al segundo eslabón y las dos quintas partes restantes corresponden al tercer eslabón.

Cada pieza fue diseñada en el programa Solid Works 2003, se ha elegido este programa sobre Mechanical Desktop 6.0 y AutoCAD 2002 dado que sus características de diseño son superiores a las de los programas anteriores, entre las que destacan un ambiente de trabajo más amigable, herramientas de renderizado y texturas, facilidad de ensamblaje de piezas, herramienta para realizar análisis de elemento finito (COSMOSXpress), además de ser un programa diseñado específicamente para la realización de piezas mecánicas.

2.1.3 Comparación y selección de materiales

Para la realización del manipulador se deben elegir los materiales más adecuados de tal forma que su peso y su rigidez sean las más adecuadas. Para poder hacer esto se utilizó una tabla comparativa (tabla 2.1.1) en la cual se presentan una serie de materiales y sus diferentes propiedades.

Tabla 2.1.1 Propiedades típicas de materiales de ingeniería.

PROPIEDAD	UNIDADES	MATERIAL			
		Nylon	Bronce	Acero	Aluminio
Densidad	g/cm ³	1.15	8,8	7,84	2,7
Resistencia a la Tracción	Mpa	83	152	248	207
Módulo de elasticidad	Mpa	2,75*10 ³	1,1*10 ⁵	2*10 ⁵	2,9*10 ⁴
Resistencia relativa al peso	Acero=1	2,27	0,54	1	2,41
Coefficiente de expansión térmica lineal	mm/mm/K	100*10 ⁻⁶	20*10 ⁻⁶	12*10 ⁻⁶	24*10 ⁻⁶

<http://www.prodigyweb.net.mx/degcorp/Quimica/Aluminio.htm>

Después de revisar la tabla 2.1.1 se ha decidido utilizar Aluminio 6063 T5 [ver apéndice F] para la construcción de los eslabones, los ejes, los acoplamientos y la base del manipulador por las siguientes razones.

- Es fácil de obtener.
- Es muy ligero.
- Su precio es menor que el del Nylamid.
- Se puede maquinar fácilmente.
- Un volumen dado de aluminio pesa menos que 1/3 del mismo volumen de acero.
- El aluminio es un material que no se oxida.

2.1.4 Definición de la transmisión a utilizar.

La transmisión que se utiliza en el manipulador es por medio de engranes cilíndricos rectos, se ha elegido esta transmisión por las siguientes características:

- No presenta deslizamiento en comparación con la transmisión por medio de bandas planas.
- No requiere de un mantenimiento costoso como lo requeriría una transmisión de cadenas.
- Es una transmisión silenciosa.
- Presenta mayor tiempo de vida en comparación con la transmisión por banda.
- Es la transmisión que ocupa menos espacio.

2.1.5 Selección de los rodamientos

Los cojinetes que se han decidido utilizar son de carga radial, de bola de hilera única, para los ejes de los eslabones que se encuentran en las articulaciones dos y tres dado que la carga que deben soportar es únicamente radial, mientras que para la primera articulación se utiliza un rodamiento de carga axial en la parte inferior que soporta el peso del manipulador y un rodamiento de carga radial en la parte superior para mantener la alineación del eje y eliminar la fricción entre el eje y la base del manipulador.

2.1.6 Análisis estático a través de software

Este análisis se llevo a cabo en el software de simulación llamado Visual Nastran 2001. Se ha elegido este paquete de simulación ya que en él es posible realizar un estudio que muestre a través de imágenes el comportamiento estático y dinámico del manipulador.

El primer análisis estático se realizó únicamente para los ejes a fin de conocer los esfuerzos que sufren estos y determinar las características del motor a utilizar.

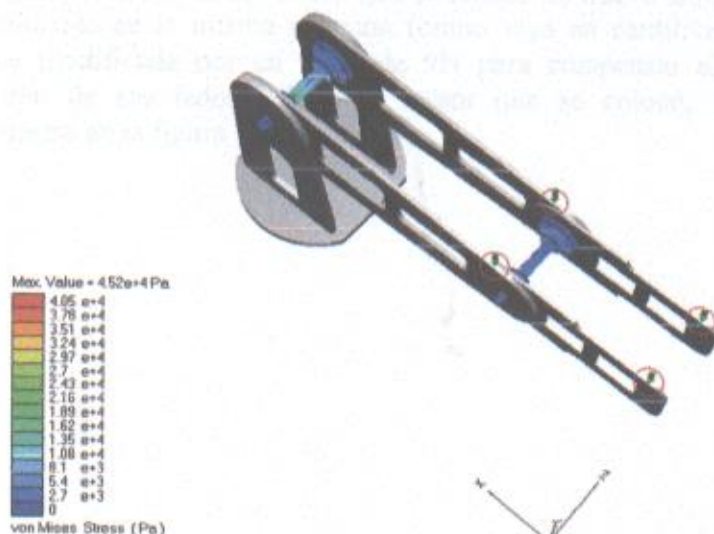


Figura 2.1.2 Análisis estático de los ejes del manipulador.

En la figura 2.1.2 se puede observar al manipulador extendido horizontalmente, simulando una viga en cantiliver, las fuerzas que se le aplicaron al manipulador fueron de 5N cada una (en el extremo del eslabón 2) y de 9N (en el extremo del eslabón 3).

El esfuerzo de trabajo máximo que presentan los ejes es de 45.2 kPa, pero como se observa a través de la imagen la mayor parte del eje no alcanza siquiera el valor de 18.9 kPa (color verde). En la figura 2.1.3 se observa la deformación que sufrirían los ejes.

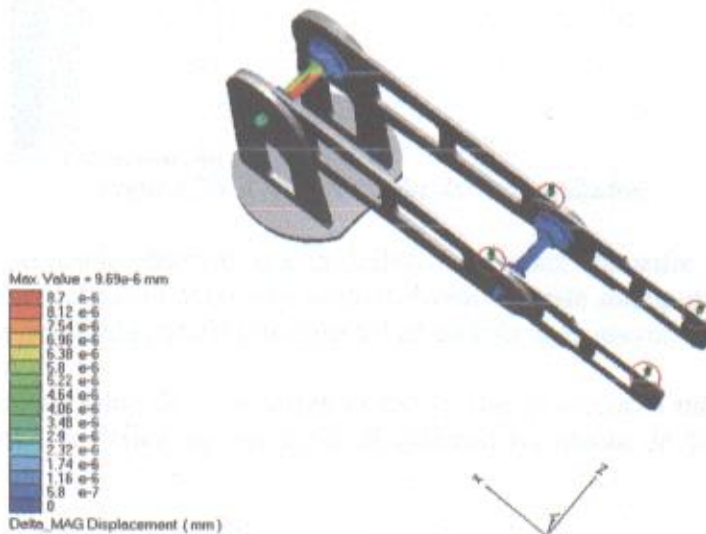


Figura 2.1.3 Deformación de los ejes del manipulador.

A partir de los resultados obtenidos en los análisis anteriores se ha decidido utilizar el motor 404 774 de la serie 0225 marca Valeo [ver apéndice D].

Después de realizar el análisis estático de los ejes se realizó un nuevo análisis estático de todo el manipulador colocado en la misma posición (como viga en cantiliver), pero una de las fuerzas de 5N fue modificada por un valor de 9N para compensar el peso que tiene el manipulador en uno de sus lados debido al motor que se colocó, la deformación del manipulador se muestra en la figura 2.1.4.

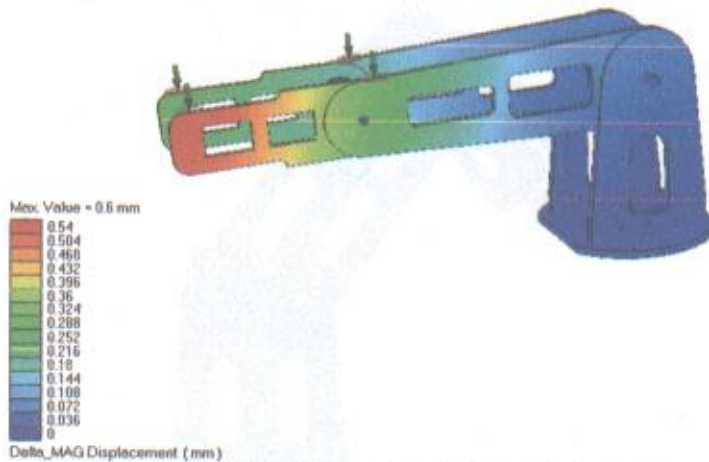


Figura 2.1.4 Deformación del manipulador.

En la figura 2.1.4 se puede observar, que la deflexión máxima que sufre el manipulador es de 0.6 mm, que es una deflexión aceptable para el diseño de este manipulador debido a que la tolerancia máxima permitida para la posición del efector final es mayor que ésta.

Además el esfuerzo máximo de Von-Mises es menor que el esfuerzo máximo permisible del material, de ahí que la deflexión que sufre el material no afecta de forma considerable el diseño.

2.1.7 Análisis dinámico a través de software

Este análisis también fue llevado a cabo en el programa de simulación Visual Nastran 2001, las fuerzas que se aplicaron al manipulador fueron mayores a las que se pretenden pueda soportar, esto con el fin de garantizar el desempeño del manipulador.

Como ya se mencionó anteriormente las piezas fueron diseñadas en Solid Works y para poder utilizarlas en Visual Nastran se requiere guardar cada una de ellas con la extensión .igs (IGES), posteriormente se vuelven a ensamblar dentro de Visual Nastran y se realiza el análisis dinámico.

Este análisis se llevó a cabo de la siguiente forma:

- Análisis estructural del Manipulador.
- Análisis de ejes.

Análisis estructural del Manipulador.

La estructura se restringió en la base en los tres ejes (x,y,z), como se muestra en la figura 2.1.5



Figura 2.1.5 Manipulador con restricciones en la base.

Posteriormente se realizó un mallado (Figura 2.1.6) cuya medida oscilo entre: 8.16 mm y 10.6 mm con un ángulo de 45° , esta variación en el tamaño del mallado se debe a la geometría de cada pieza.



Figura 2.1.6 Mallado del manipulador.

Posteriormente se realiza el análisis del manipulador sin someterlo a ninguna carga puntual, solo considerando las características del material.

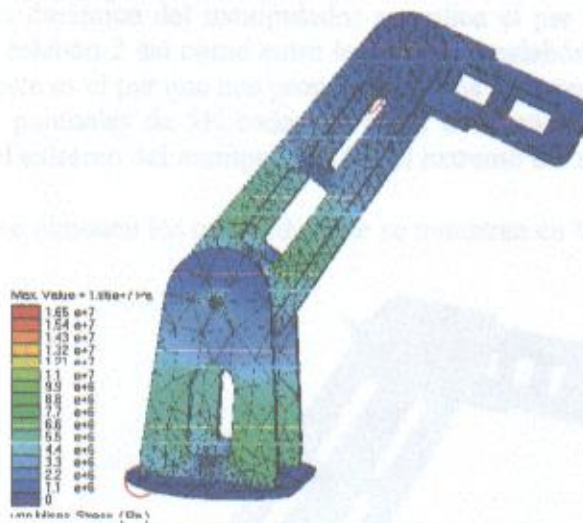


Figura 2.1.7 Análisis dinámico del manipulador sin carga.

La deformación del manipulador se muestra en la figura 2.1.8.

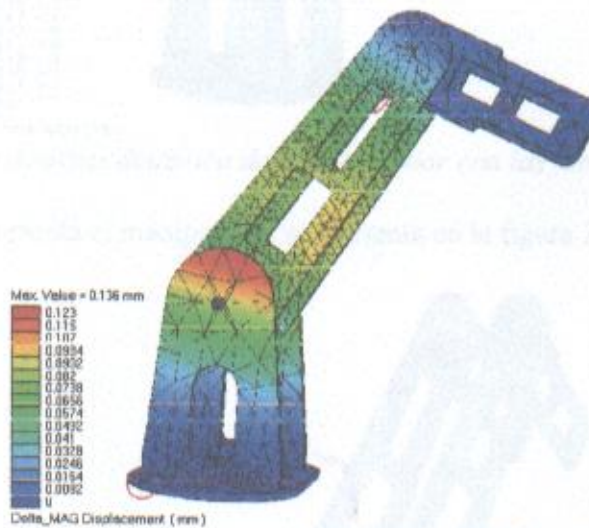


Figura 2.1.8 Deformación del manipulador sin aplicar cargas.

De la figura 2.1.7 se puede observar que el esfuerzo máximo de Von-Mises es de 18.6MPa, el esfuerzo máximo de cedencia del aluminio que se empleara es de: 55148500 Pa. Para calcular el factor de seguridad se divide el esfuerzo máximo de cedencia del material entre el esfuerzo máximo de trabajo (en este caso se puede utilizar el esfuerzo de Von-Mises) y se obtiene:

$$N = \frac{\sigma_Y}{\sigma_M} = \frac{55148500 \text{ Pa}}{18.6 \times 10^6 \text{ Pa}} = 2.965 \dots\dots\dots(10.1)$$

Para realizar el análisis dinámico del manipulador se aplica el par de un motor en la unión entre el eslabón 1 y el eslabón 2 así como entre la unión del eslabón 2 y el eslabón 3, ambos pares de 1300 N mm (este es el par que nos proporcionan los motores seleccionados). Además se aplican dos fuerzas puntuales de 5N cada una en el extremo del segundo eslabón y dos fuerzas más de 9N en el extremo del manipulador (en el extremo del eslabón 3).

Se realiza el análisis y se obtienen los resultados que se muestran en la figura 2.1.9.

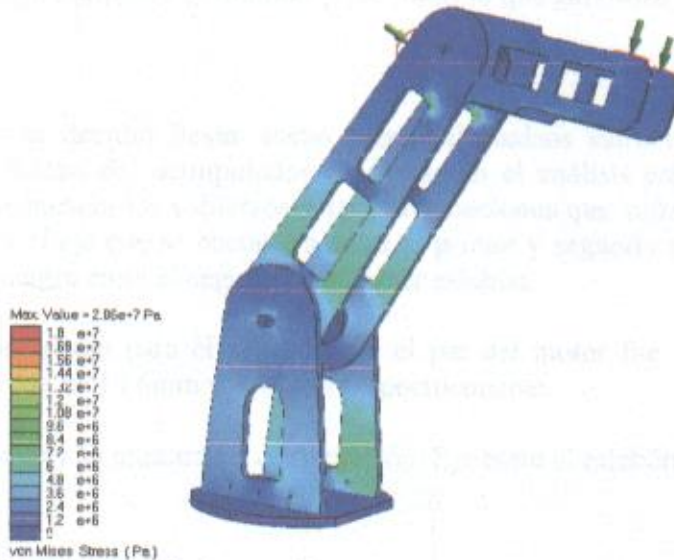


Figura 2.1.9 Análisis dinámico del manipulador con las fuerzas aplicadas.

La deformación que presenta el manipulador se presenta en la figura 2.1.10.

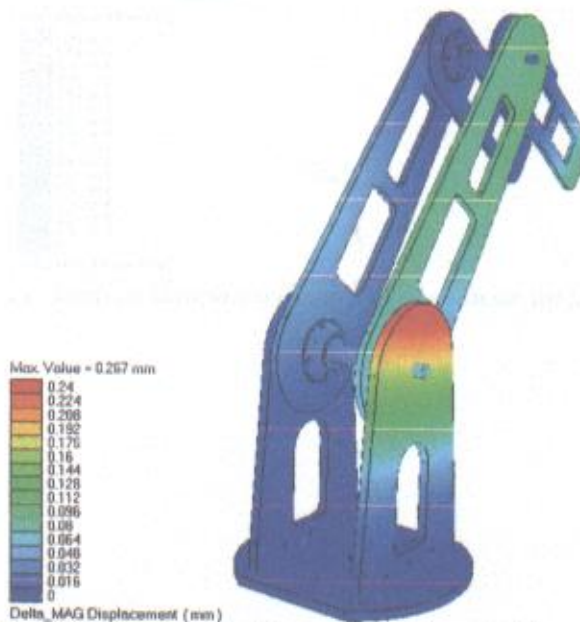


Figura 2.1.10 Deformación del manipulador al aplicar las fuerzas.

Para poder garantizar que el diseño es aceptable se puede recurrir nuevamente a la fórmula de factor de seguridad y comprobar que el esfuerzo máximo de trabajo es menor que el esfuerzo máximo de cedencia.

$$N = \frac{\sigma_Y}{\sigma_M} = \frac{55148500 \text{ Pa}}{20.6 \times 10^6 \text{ Pa}} = 2.677 \dots\dots\dots(10.2)$$

Como se puede ver el factor de seguridad es bastante aceptable ya que se está diseñando un manipulador y el desplazamiento es mínimo (0.267mm) lo que garantiza un buen diseño.

Análisis de ejes

El análisis de ejes se decidió llevar a cabo fuera del análisis estructural ya que es parte fundamental en el diseño del manipulador y además en el análisis estructural no se puede apreciar de la mejor manera los esfuerzos y las deformaciones que sufren estos. Este análisis se realizó tanto para el eje que se encuentra entre el primer y segundo eslabón, como para el segundo que se encuentra entre el segundo y el tercer eslabón.

Tanto para el primer como para el segundo eje el par del motor fue de 1300 N mm, y el tamaño del mallado fue de 10.6mm y 8.47mm respectivamente.

Los resultados obtenidos se muestran a continuación. Eje entre el eslabón 1 y 2.

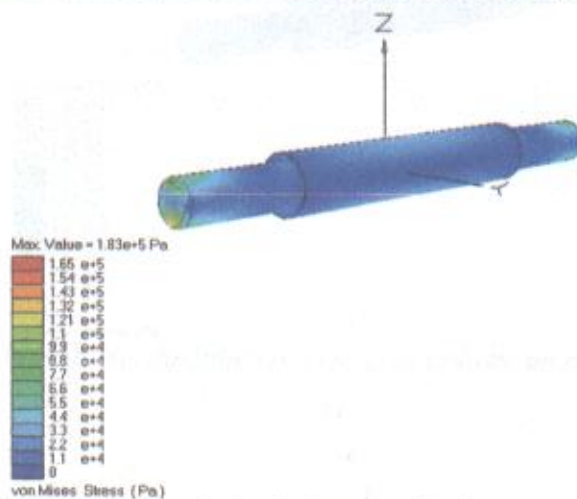


Figura 2.1.11 Análisis dinámico del eje 1, al aplicar un par de 1.3N m.

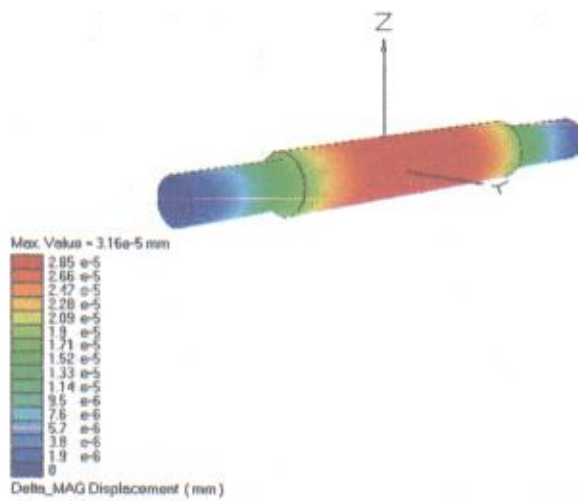


Figura 2.1.12 Deformación del eje 1, al aplicar un par de 1.3N m.

Eje entre el eslabón 2 y 3

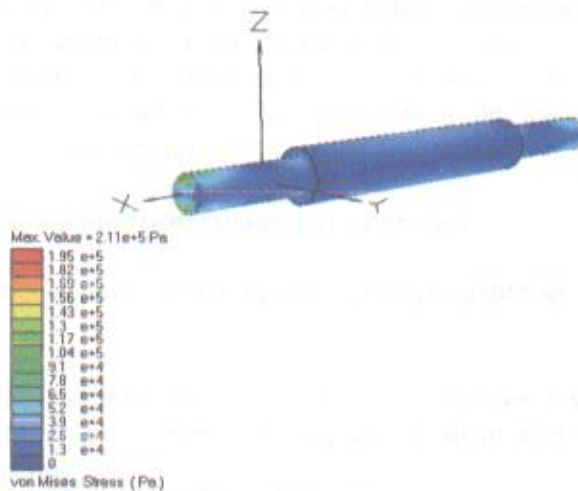


Figura 2.1.13 Análisis dinámico del eje 2, al aplicar un par de 1.3N m.

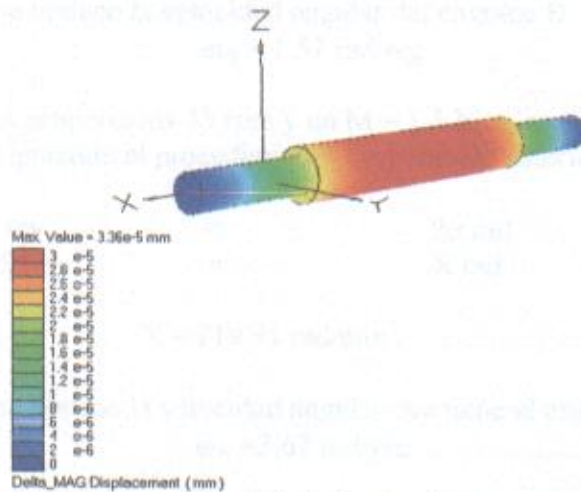


Figura 2.1.14 Deformación del eje 2, al aplicar un par de 1.3N m.

A partir de los resultados mostrados en las figuras 2.1.11, 2.1.12, 2.1.13 y 2.1.14, se puede observar claramente que las medidas propuestas cumplen satisfactoriamente con el diseño, ya que el factor de seguridad calculado tiene un valor de 261, que es un valor bastante elevado para lo que requiere el manipulador. Pero se ha decidido mantener estas medidas por que de otra manera los diámetros interiores de los engranes y de los rodamientos serian poco convencionales (serian muy pequeños).

2.1.8 Análisis matemático de la transmisión por engranes

Para poder llevar a cabo el análisis del sistema, primeramente se debe definir la siguiente condición:

Se requiere una velocidad máxima de 15 rpm en los eslabones 2 y 3 del manipulador, esta velocidad es adecuada para que el manipulador pueda realizar su tarea de forma rápida pero sin poner en riesgo al operador.

Para resolver este problema se puede suponer en un principio un engrane (que funge como piñón) al cual se le llama A y que cuenta con las siguientes características:

$$m_1 = 25 \times 10^{-3} \text{ kg} \qquad r_{1int} = 5 \text{ mm} \qquad r_{1ext} = 14 \text{ mm} \dots\dots\dots(10.3)$$

La velocidad que se desea en el engrane conducido (llamado engrane B) es de 15 rpm por lo tanto se tiene la siguiente relación:

$$\begin{matrix} 1 \text{ rev} & = & 2\pi \text{ rad} \\ 15 \text{ rev} & = & X \text{ rad} \end{matrix}$$

$$X = 94.27 \text{ rad/min} \dots\dots\dots(10.4)$$

Dividiendo entre 60 s, se obtiene la velocidad angular del engrane B:

$$\omega_B = 1.57 \text{ rad/seg} \dots\dots\dots (10.5)$$

Si sabemos que el motor proporciona 35 rpm y un $M = 1.3 \text{ N m}$, se puede conocer la velocidad angular del engrane A siguiendo el procedimiento mencionado anteriormente:

$$\begin{array}{l} 1 \text{ rev} = 2\pi \text{ rad} \\ 35 \text{ rev} = X \text{ rad} \end{array}$$

$$X = 219.91 \text{ rad/min} \dots\dots\dots (10.6)$$

Dividiendo entre 60 s, se obtiene la velocidad angular que tiene el engrane A.

$$\omega_A = 3.67 \text{ rad/seg} \dots\dots\dots (10.7)$$

Por lo tanto para calcular el diámetro del engrane B se toma la siguiente relación.

$$V_A = V_B$$

pero:

$$V_A = r_{Aext} \omega_A$$

y

$$V_B = r_{Bext} \omega_B$$

igualando:

$$r_{Aext} \omega_A = r_{Bext} \omega_B \dots\dots\dots (10.8)$$

de la ecuación anterior podemos despejar r_{Bext} y obtendremos lo siguiente:

$$r_B = \frac{\omega_A r_{Aext}}{\omega_{Bext}} \dots\dots\dots (10.9)$$

Sustituyendo valores se obtiene que el radio del engrane B es de:

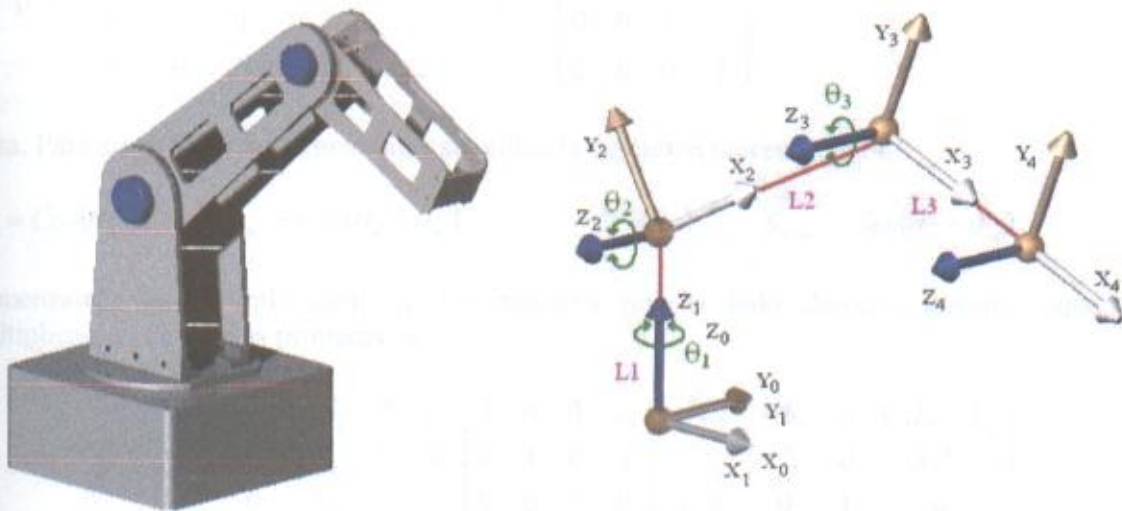
$$r_B = 32.67 \text{ mm} \dots\dots\dots (10.10)$$

2.2 Análisis cinemático del manipulador

2.2.1 Cinemática directa

Cinemática del manipulador propuesto

En el caso del robot manipulador de 3DGL que se ha propuesto se establecen los sistemas de coordenadas a cada eslabón como se muestra en la figura 2.2.1b y se obtienen los parámetros de Denavit – Hartenberg como se muestra en la tabla 2.2.1.



a) Vista preliminar del manipulador b) Definición de los sistemas de coordenadas
 Figura 2.2.1 Robot Manipulador de 3DGL.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	-90	0	0	θ_2
3	0	L2	0	θ_3
4	0	L3	0	0

Tabla 2.2.1 Parámetros de Denavit – Hartenberg para el manipulador de 3DGL.

Para encontrar matriz de transformación del efector final relativo al sistema de coordenadas de la base se deben concatenar las matrices de la siguiente manera:

$${}^0T_4 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 \dots \dots \dots (11.1)$$

en donde:

$$\begin{aligned}
 {}^0_1T &= \begin{pmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & {}^1_2T &= \begin{pmatrix} C_2 & -S_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -S_2 & -C_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & \dots\dots\dots(11.2) \\
 {}^2_3T &= \begin{pmatrix} C_3 & -S_3 & 0 & L_2 \\ S_3 & C_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & {}^3_4T &= \begin{pmatrix} 1 & 0 & 0 & L_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
 \end{aligned}$$

Nota: Para simplificar las expresiones se utiliza la siguiente representación:

$$C_n = \text{Cos}(\theta_n) \quad C_{n+m} = \text{Cos}(\theta_n + \theta_m) \quad S_n = \text{Sen}(\theta_n) \quad S_{n+m} = \text{Sen}(\theta_n + \theta_m)$$

Comenzando la multiplicación de las matrices por el lado derecho tenemos, que la multiplicación de las dos primeras es:

$${}^2_4T = {}^2_3T {}^3_4T = \begin{pmatrix} C_3 & -S_3 & 0 & L_2 \\ S_3 & C_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & L_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} C_3 & -S_3 & 0 & C_3L_3 + L_2 \\ S_3 & C_3 & 0 & S_3L_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^1_4T = {}^1_2T {}^2_4T = \begin{pmatrix} C_2 & -S_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -S_2 & -C_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C_3 & -S_3 & 0 & C_3L_3 + L_2 \\ S_3 & C_3 & 0 & S_3L_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} C_{2+3} & -S_{2+3} & 0 & L_3C_{2+3} + L_2C_2 \\ 0 & 0 & 1 & 0 \\ -S_{2+3} & -C_{2+3} & 0 & -L_3S_{2+3} - S_2L_2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^0_4T = {}^0_1T {}^1_4T = \begin{pmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C_{2+3} & -S_{2+3} & 0 & L_3C_{2+3} + L_2C_2 \\ 0 & 0 & 1 & 0 \\ -S_{2+3} & -C_{2+3} & 0 & -L_3S_{2+3} - S_2L_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \dots(11.3)$$

Lo cual resulta en la matriz de transformación que describe la posición del efector final con relación al sistema de coordenadas base del manipulador:

$${}^0_4T = \begin{pmatrix} C_1 C_{2+3} & -C_1 S_{2+3} & -S_1 & C_1(L_3 C_{2+3} + L_2 C_2) \\ S_1 C_{2+3} & -S_1 S_{2+3} & C_1 & S_1(L_3 C_{2+3} + L_2 C_2) \\ -S_{2+3} & -C_{2+3} & 0 & -L_3 S_{2+3} - S_2 L_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \dots\dots\dots (11.4)$$

2.2.2 Cinemática inversa

Ahora para calcular los ángulos de las articulaciones cuando se tiene la posición del efector final se recurre al siguiente proceso matemático:

Se plantea una matriz de transformación general (ecuación 11.5) que describe la posición y orientación del efector final.

$$M = \begin{pmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \dots\dots\dots (11.5)$$

Se obtiene las inversas de las matrices de transformación de cada par de eslabones vecinos.

$${}^{(0_1T)^{-1}} = \begin{pmatrix} C_1 & S_1 & 0 & 0 \\ -S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & -L_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad {}^{(1_2T)^{-1}} = \begin{pmatrix} C_2 & 0 & -S_2 & 0 \\ -S_2 & 0 & -C_2 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \dots\dots\dots (11.6)$$

Dado que el hecho de multiplicar la matriz de transformación obtenida en (11.4) por la inversa de la matriz de transformación que relaciona el primer par de sistemas de coordenadas, genera una matriz de transformación que relaciona θ_1 con el vector de posición de (11.5) se puede despejar dicha variable de articulación.

$${}^{(0_1T)^{-1}} M = \begin{pmatrix} C_1 & S_1 & 0 & 0 \\ -S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & L_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^{(0_1T)^{-1}} M = \begin{pmatrix} C_1 r_{11} + S_1 r_{21} & C_1 r_{12} + S_1 r_{22} & C_1 r_{13} + S_1 r_{23} & C_1 p_x + S_1 p_y \\ -S_1 r_{11} + C_1 r_{21} & -S_1 r_{12} + C_1 r_{22} & -S_1 r_{13} + C_1 r_{23} & -S_1 p_x + C_1 p_y \\ r_{31} & r_{32} & r_{33} & p_z - L_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^0_1T^{-1} M = {}^1_4T \dots\dots\dots(11.7)$$

$$\begin{pmatrix} C_1r_{11} + S_1r_{21} & C_1r_{12} + S_1r_{22} & C_1r_{13} + S_1r_{23} & C_1p_x + S_1p_y \\ -S_1r_{11} + C_1r_{21} & -S_1r_{12} + C_1r_{22} & -C_1r_{13} + C_1r_{23} & -S_1p_x + C_1p_y \\ r_{31} & r_{32} & r_{33} & p_z - L_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} C_{2+3} & -S_{2+3} & 0 & L_3C_{2+3} + C_2L_2 \\ 0 & 0 & 1 & 0 \\ -S_{2+3} & -C_{2+3} & 0 & -L_3S_{2+3} - S_2L_2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

De los elementos (2,4) de cada matriz se obtiene la ecuación 11.8

$$\begin{aligned} S_1p_x - C_1p_y &= 0 \\ \Rightarrow \theta_1 &= \arctan\left(\frac{p_y}{p_x}\right) \dots\dots\dots(11.8) \end{aligned}$$

Continuando con la multiplicación de la matrices inversas y seleccionando adecuadamente los elementos se pueden obtener sucesivamente expresiones que relacionen las variables articulares con respecto al vector de posición (X, Y ,Z), pero debido a la complejidad de las ecuaciones no lineales que resultan de esta relación es muy común utilizar un método geométrico para encontrar expresiones que relacionen los ángulos de las articulaciones con las variables cartesianas.

Las expresiones para θ_2 y θ_3 son:

$$\begin{aligned} A &= \left(\frac{p_x^2 + p_y^2 + p_z^2 - L_2^2 - L_3^2}{2L_2L_3} \right) \\ \theta_3 &= \arctan\left(\frac{\sqrt{1 - A^2}}{A} \right) \dots\dots\dots(11.9) \\ \theta_2 &= \arctan\left(\frac{p_z}{\sqrt{p_x^2 + p_y^2}} \right) - \arctan\left(\frac{L_3 \sin(\theta_3)}{L_2 + L_3 \cos(\theta_3)} \right) \end{aligned}$$

2.2.3 Jacobiano

Para establecer cuales son las velocidades angulares que se deben aplicar en cada articulación para que el extremo final del manipulador desarrolle una movimiento concreto a una determinada velocidad, se debe disponer de alguna herramienta matemática que relacione estas variables. La relación entre ambos vectores de velocidad se puede obtener a través de la matriz jacobiana.

Tomando el vector de posición de la matriz de transformación mostrado en la ecuación 11.4 tenemos las ecuaciones;

$$\begin{aligned} X &= \text{Cos}(\theta_1)(L_3\text{Cos}(\theta_2 + \theta_3) + \text{Cos}(\theta_2)L_2) \\ Y &= \text{Sen}(\theta_1)(L_3\text{Cos}(\theta_2 + \theta_3) + \text{Cos}(\theta_2)L_2) \dots\dots\dots (11.10) \\ Z &= -L_3\text{Sen}(\theta_2 + \theta_3) - L_2\text{Sen}(\theta_2) \end{aligned}$$

Ahora se obtiene el jacobiano de estas ecuaciones mediante la ecuación 11.11.

$$J = \begin{pmatrix} \frac{\partial x}{\partial \theta_1} & \frac{\partial x}{\partial \theta_2} & \frac{\partial x}{\partial \theta_3} \\ \frac{\partial y}{\partial \theta_1} & \frac{\partial y}{\partial \theta_2} & \frac{\partial y}{\partial \theta_3} \\ \frac{\partial z}{\partial \theta_1} & \frac{\partial z}{\partial \theta_2} & \frac{\partial z}{\partial \theta_3} \end{pmatrix} \dots\dots\dots (11.11)$$

Lo cual resulta en la matriz jacobiana que se describe a continuación (ecuación 11.12).

$$J = \begin{pmatrix} -\text{Sen}(\theta_1)(L_3\text{Cos}(\theta_2 + \theta_3) + \text{Cos}(\theta_2)L_2) & \text{Cos}(\theta_1)(-L_3\text{Sen}(\theta_2 + \theta_3) - \text{Sen}(\theta_2)L_2) & \text{Cos}(\theta_1)L_3\text{Sen}(\theta_2 + \theta_3) \\ \text{Cos}(\theta_1)(L_3\text{Cos}(\theta_2 + \theta_3) + L_2\text{Cos}(\theta_2)) & \text{Sen}(\theta_1)(-L_3\text{Sen}(\theta_2 + \theta_3) - \text{Sen}(\theta_2)L_2) & \text{Sen}(\theta_1)L_3\text{Sen}(\theta_2 + \theta_3) \\ 0 & -L_3\text{Cos}(\theta_2 + \theta_3) - \text{Cos}(\theta_2)L_2 & -L_3\text{Cos}(\theta_2 + \theta_3) \end{pmatrix} \dots\dots\dots (11.12)$$

Esta matriz jacobiana permite obtener la velocidad instantánea del efector final de acuerdo a las velocidades angulares de las articulaciones.

Posteriormente se obtiene el determinante de la matriz jacobiana con la intención de encontrar las singularidades del manipulador, es decir, geoméricamente representan las zonas en donde se ha alcanzado el límite de trabajo del manipulador o en donde se produce un alineamiento de dos o más eslabones.

$$|J| = \frac{1}{2} [L_3^2 L_2 \text{Sen}(\theta_2 + 2\theta_3) - L_2 L_3^2 \text{Sen}(\theta_2) + L_2^2 L_3 \text{Sen}(-\theta_2 + \theta_3) + L_2^2 L_3 \text{Sen}(\theta_2 + \theta_3)] \dots\dots\dots (11.13)$$

Como se puede observar evaluando $\theta_2 = 0$ y $\theta_3 = 0$ ó $\theta_2 = \pi$ y $\theta_3 = 0$ se anula el determinante del jacobiano, lo cual quiere decir que en estas condiciones se produce una singularidad.

Existen otros dos puntos los cuales matemáticamente se produciría una singularidad, pero que físicamente nunca presentan debido a las restricciones mecánicas del manipulador. Estos puntos singulares son:

$$\theta_2 = \pi \quad \text{y} \quad \theta_3 = \pi$$

$$\theta_2 = 0 \quad \text{y} \quad \theta_3 = \pi$$

2.3 Diseño Electrónico

2.3.1 Descripción del sistema de control

La computadora personal y el microcontrolador se encuentran conectados mediante comunicación serial, los datos son decodificados y procesados por el microcontrolador. Posteriormente los envía a la correspondiente etapa de control, para realizar la etapa de control. La PC lleva a cabo el cálculo de la cinemática directa e inversa a fin de generar la trayectoria seleccionada.

El microcontrolador se encarga de enviar los datos, señales de control, así como los tiempos que se requieren para llevar a cabo los diferentes tipos de control. La seguridad está controlada mediante las interrupciones externas que emiten los sensores de límite con los que cuenta el brazo manipulador. Por otra parte el monitoreo de la posición y la velocidad está a cargo de encoders (codificadores de posición) cuya resolución es de 1024 pulsos por revolución.

La forma de comunicación entre el usuario y el robot se realiza por medio de una interfaz gráfica desarrollada en un lenguaje de programación ejecutable en ambiente Windows. El usuario tiene la opción de elegir el tipo de control que desee aplicar para posteriormente modificar los parámetros principales de la ley de control, de esta manera se obtiene el desempeño del robot con lo cual se puede observar el comportamiento de los tipos de control y elegir el más adecuado a la necesidad a resolver.

A continuación se presenta un diagrama a bloques en donde se muestra a grandes rasgos lo que se realizó (Figura 2.3.1):

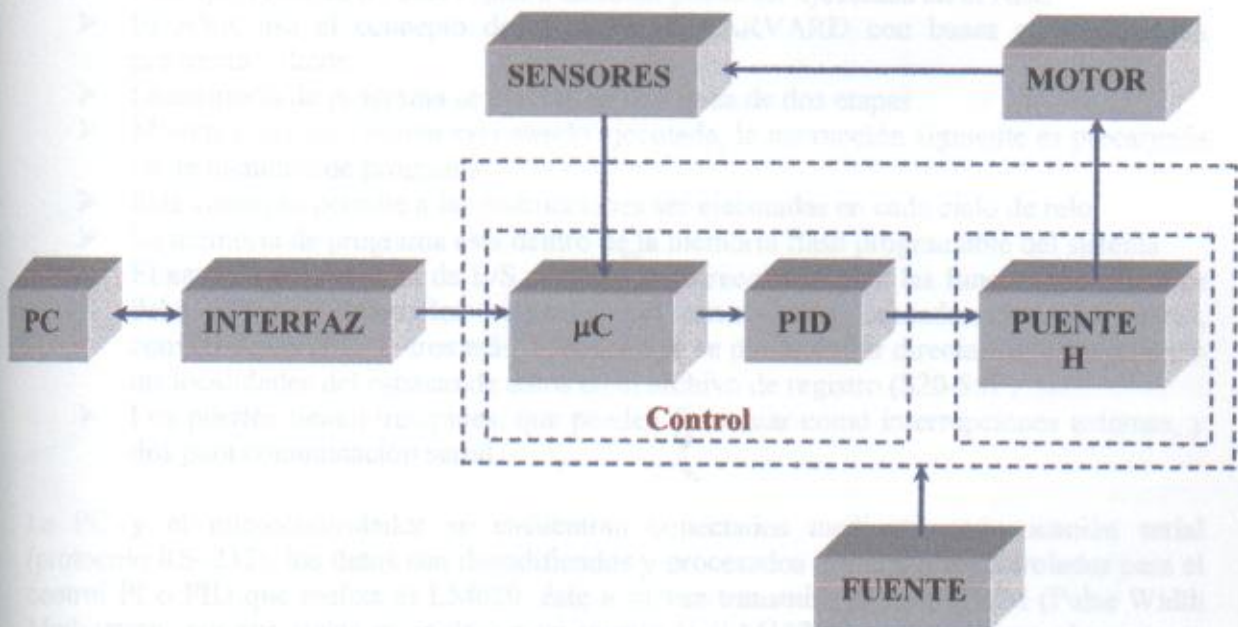


Figura 2.3.1 Diagrama a bloques de la etapa de control para un grado de libertad.

2.3.2 Interfaz PC - Robot

La electrónica controladora del robot inicia con un microcontrolador AVR funcionando como interfaz entre la PC y el LM629. Básicamente el AVR sirve como un búfer entre la PC y los LM629, pero se utiliza este enfoque para aprovechar la considerable cantidad de entradas y salidas digitales de manera que el sistema de control en conjunto sea extensible a robots de más grados de libertad sin necesidad de reconfiguraciones mayores. La programabilidad del AVR también nos permite implementar sistemas más sofisticados con detección de errores y conexión "plug&play" de más chips LM629. La comunicación con la PC se realiza utilizando la UART incorporada al AVR.

La comunicación entre la PC y la etapa de control del brazo manipulador, se realiza mediante un microcontrolador AVR ATMega16L, se ha elegido este AVR debido a las características que posee, tales como:

- Tecnología RISC
- Un puerto de entrada (para un convertidor analógico digital A/D, ya que también puede ser utilizado como puerto de salida) y tres puertos de entrada/salida (E/S)
- Archivo de registro constituido por 32 registros de 8 bits de acceso rápido
- Seis de estos registros pueden ser usados como punteros de registro de 16 bits para el espacio de direcciones
- Los diferentes tipos de direccionamiento de memoria también se encuentran disponibles.
- Una ALU (Arithmetic Logic Unit) que soporta funciones aritméticas y lógicas entre registros o entre constantes y registros
- Una operación de un solo registro también puede ser ejecutada en la ALU
- El AVR usa el concepto de arquitectura HARVARD con buses separados para programa y datos
- La memoria de programa se ejecuta en una línea de dos etapas
- Mientras una instrucción esta siendo ejecutada, la instrucción siguiente es precargada de la memoria de programa
- Este concepto permite a las instrucciones ser ejecutadas en cada ciclo de reloj
- La memoria de programa esta dentro de la memoria flash programable del sistema
- El espacio de memoria de E/S contiene 64 direcciones para las funciones perifericas del CPU tal como los registros de control, los contadores/temporizadores, convertidores A/D o otros más. Este espacio se puede entrar directamente o siguiendo las localidades del espacio de datos en el archivo de registro (\$20-\$5F)
- Los puertos tienen tres pines, que pueden funcionar como interrupciones externas, y dos para comunicación serial

La PC y el microcontrolador se encuentran conectados mediante comunicación serial (protocolo RS-232), los datos son decodificados y procesados por el microcontrolador para el control PI o PID que realiza el LM629, éste a su vez transmite la señal PWM (Pulse Width Modulation, por sus siglas en inglés) a un puente H (LM18200), a éste último elemento se encuentra conectado el motor. Cabe mencionar que a pesar que el LM629 está diseñado para realizar la función de un PID, se puede emplear también como un PI, mandando a cero la ganancia diferencial.

Lo anterior describe la etapa correspondiente al control y electrónica de potencia, para llevar a cabo el control del brazo manipulador. La PC realiza el cálculo de la cinemática directa e inversa a fin de generar la trayectoria seleccionada. La seguridad está controlada mediante las interrupciones externas generadas por los sensores de límite y las señales de error del LM629.

Es importante aclarar que el LM629 no representa la única forma en la que se puede realizar un filtro PI o PID. Este tipo de filtros se pueden realizar mediante el uso de amplificadores operacionales, o bien programándolo en el mismo microcontrolador que se está usando, ya sea AVR, PIC, entre otros. Se ha optado por emplear el LM629 debido a la precisión que ofrece, a la dualidad del uso del filtro (PI o PID) con el mismo circuito y a la simplificación en el código de control, entre otras características que se mencionan más adelante.

2.3.3 Breve descripción del LM629 [ver apéndice C]

El circuito integrado LM629 es un procesador dedicado al control de movimiento de un motor de corriente continua, servomotores sin escobillas (brushless) o cualquier tipo que pueda emitir una señal incremental en cuadratura como realimentación de su estado. En el robot de este proyecto, esta señal viene directamente de un encoder incremental acoplado al eje del motor. Para este sistema hay tres LM629, uno por cada motor, disponiendo cada motor de su propio encoder. La salida de cada uno de los LM629 es una señal PWM, ésta señal actúa directamente sobre la etapa de potencia de los motores.

Características generales del LM629

- 3 registros de 32 bits para la posición, velocidad y aceleración de los motores
- Filtro PID programable con coeficientes de 16 bits
- Intervalo de muestreo del control derivativo programable por el usuario
- Generador interno de patrones trapezoidales de velocidad
- La velocidad, la posición objetivo y los parámetros del filtro se pueden cambiar durante el movimiento del motor
- Funcionamiento en modo posición o modo velocidad
- Servidor de interrupciones programables en tiempo real
- Bus de datos paralelo asíncrono de 8 bits
- Interface para un encoder incremental con señal de índice
- Empaquetamiento de 28 pines formato DIP

El AVR se comunica con el LM629 a través de su bus de datos para facilitar la programación del perfil trapezoidal de velocidad y del filtro PID. Para iniciar el movimiento, el AVR carga los valores de aceleración, velocidad y posición deseada en el generador de trayectorias del LM629. En cada periodo de muestreo estos valores son usados para calcular la nueva posición deseada, a la que se le resta la posición real del eje del motor, obtenida por el encoder incremental. Así se origina la señal de error, que es la entrada al filtro compensador PID. El convertidor digital/análogo convierte la señal producida a otra aplicable al motor.

Los bloques más importantes en el LM629 son:

- a) El generador de trayectorias

- b) La realimentación en posición
- c) El filtro digital PID
- d) La salida PWM

Para este proyecto el corazón del sistema es el microcontrolador ATMega16L que envía todos los comandos al LM629. Al LM629 llega primero el valor de la señal de inicio, es decir, cuando el motor inicia su movimiento, posteriormente se retroalimenta con el valor de la señal del encoder, esta señal es decodificada dentro del LM629 para determinar cuál es la posición del motor; con la señal que se detecta a la salida del sistema (con la ayuda del encoder) y la de entrada se genera una diferencia de error que actúa directamente en el filtro PI o PID según sea el caso, al mismo tiempo dicho filtro (implementado en el LM629) se encarga de corregir los valores que llegan al puente H, el LM18200. Gracias a la señal de retroalimentación que recibe el LM629, se corrige la alimentación de corriente del motor, en la siguiente figura se observa el control básico de un motor empleando este tipo de circuito. En la figura 2.3.2 se muestra una descripción general del control de un grado de libertad del manipulador.

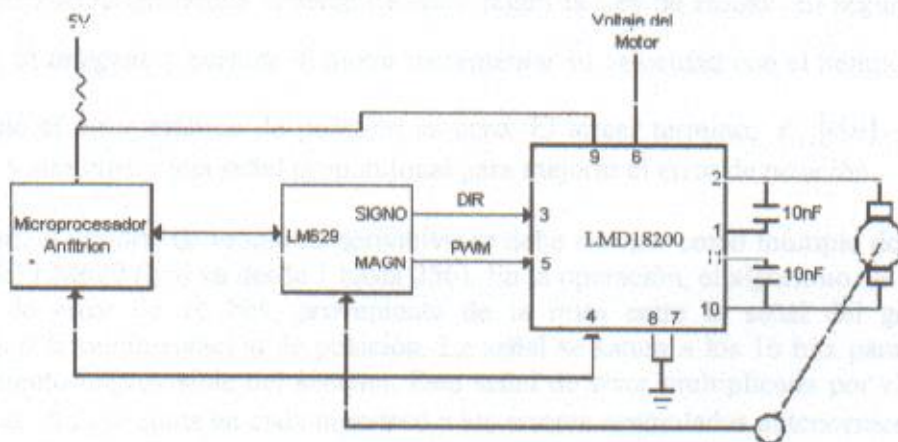


Figura 2.3.2 Descripción general del control del brazo manipulador.

Con el proceso antes descrito se logra variar la velocidad del motor, y si esto se repite un alto número de veces entonces el control ejercido sobre el sistema es efectivo e independiente de si existe carga o no en el motor. El LM629 es uno de los dispositivos de control de motores de CD y servomecanismos, que ocupan señales de retroalimentación de cuadratura incremental. En la siguiente figura se muestra el sistema de control de los motores mediante el dispositivo LM629.

2.3.4 Filtro Proporcional Integral (PI) y Proporcional Integral Derivativo (PID)

El LM629, puede ser empleado como un filtro PI (*Proporcional Integral*) o como un filtro PID (*Proporcional, Integral y Derivativo*) para compensar el lazo cerrado de control. En el caso del filtro PI el motor se sostiene en la posición deseada al aplicar una fuerza restauradora, que es: proporcional al error de posición sumada al error de integración. La siguiente ecuación en tiempo discreto ilustra el control ejecutado por el LM629:

$$u(n) = K_p e(n) + K_I \sum_{n=0}^n e(n) \dots\dots\dots (13.1)$$

En el caso del filtro PID el motor se sostiene en la posición deseada al aplicar una fuerza restauradora, que es: proporcional al error de posición sumada al error de integración, y al error derivativo. La siguiente ecuación en tiempo discreto ilustra el control ejecutado por el LM629:

$$u(n) = K_p e(n) + K_I \sum_{n=0}^n e(n) + K_D [e(n) - e(n-1)] \dots\dots\dots (13.2)$$

Donde $u(n)$ es la señal de salida de control en el tiempo "n" de muestreo, $e(n)$ es el error de posición en el tiempo "n" de muestreo, "n" indica la muestra de la derivada de muestreo, y K_p , K_I , y K_D son los parámetros cargados (ajustados por el usuario).

El primer término, $K_p e(n)$, el **proporcional**, es el llamado término proporcional, y transmite al motor una fuerza ajustada a la señal de error según la Ley de Hooke. El segundo término, $K_I \sum_{n=0}^n e(n)$, el **integral**, y permite al motor incrementar su velocidad con el tiempo, además de asegurar que el error estático de posición es cero. El tercer término, $K_D [e(n) - e(n-1)]$, el **derivativo**, y suministra una señal proporcional para mejorar el error de posición.

En este caso, el periodo de muestreo derivativo se debe escoger como múltiplo del periodo de muestreo del LM629 (que va desde 1 hasta 256). En la operación, el algoritmo del filtro recibe una señal de error de 16 bits, proveniente de la resta entre la señal del generador de trayectorias y la realimentación de posición. La señal se satura a los 16 bits para impedir un comportamiento imprevisible del sistema. Esta señal de error multiplicada por el coeficiente proporcional, K_p , se suma en cada muestreo a los errores acumulados anteriormente en forma de suma integral (K_I), para conseguir una señal de 24 bits (aunque sólo los 16 más significativos son usados posteriormente para multiplicarlos con el coeficiente derivativo K_D).

Los 16 bits menos significativos de cada uno de los tres productos (puesto que cada producto es un registro de 32 bits), se suman para producir la señal de control. De estos 16 bits, los 8 más significativos forman la salida, que tras ser debidamente tratada, decodificada y amplificada por la etapa de potencia, alimenta al motor para conducirlo a la posición designada.

Como ya se mencionó, la señal integral se mantiene en 24 bits, sin embargo, solamente se utiliza la parte alta de los 16 bits. Esta es la técnica de escalar un rango útil de los valores coeficientes de K_I . Los 16 bits se desplazan 8 posiciones a la derecha y son multiplicados por el coeficiente K_I del filtro para formar el término desde el cual se contribuye a la salida del motor. La magnitud absoluta de este producto es comparada con los coeficientes K_I , y el valor más pequeño en magnitud se apropia y se asigna entonces a la señal del motor.

La señal de derivación se multiplica por el coeficiente K_D a cada intervalo de muestreo derivativo. Este producto contribuye a la salida del motor en cada intervalo de muestreo, independientemente del intervalo de muestreo derivativo escogido por el usuario. El K_p , K_i limitado y K_D en productos, son sumados para formar una cantidad de 16 bits, y así obtener la salida de control del motor.

2.3.5 Descripción de terminales del LM629

A continuación se muestra una descripción general de las terminales del LM629.

Terminal 1

Entrada de índice: entrada del índice del encoder, si el encoder no cuenta con una, la terminal no se conecta.

Terminal 2 y 3

Señales A y B, respectivamente, del encoder: reciben las dos señales en cuadratura que indican la posición en la que se encuentra el motor.

Terminal 4 a 11

Bus de datos: se trata de un bus bidireccional que se puede conectar a un microcontrolador o directamente a la computadora.

Terminal 12

\overline{CS} (*Selección de Chip*): se debe activar esta señal para realizar operaciones de lectura o escritura.

Terminal 13

\overline{RD} (*Lectura*): debe activarse para realizar operaciones de lectura.

Terminal 14

GND: tierra.

Terminal 15

\overline{WR} (*Escritura*): debe activarse para realizar operaciones de escritura.

Terminal 16

\overline{PS} (*Selección de puerto*): se emplea para seleccionar internamente el puerto de comandos (nivel bajo) o el de datos (nivel alto) del LM629.

Terminal 17

HI: salida de interrupciones.

Terminal 18

PWM SIGN: señal de sentido de giro, va directamente a la etapa de potencia (LMD18200).

Terminal 19

PWM: señal PWM, va directamente a la etapa de potencia (LMD18200).

Terminal 20 a 25

NC: estas terminales no se conectan.

Terminal 26

CLK: señal de reloj.

Terminal 27

RST: reset.

Terminal 28

V_{DD}: alimentación.

En la siguiente figura 2.3.3 se observa la PC, el LM629, el puente H, el motor, y el codificador óptico (encoder), que forman el lazo de control PI ó PID de estudio (según sea el caso).

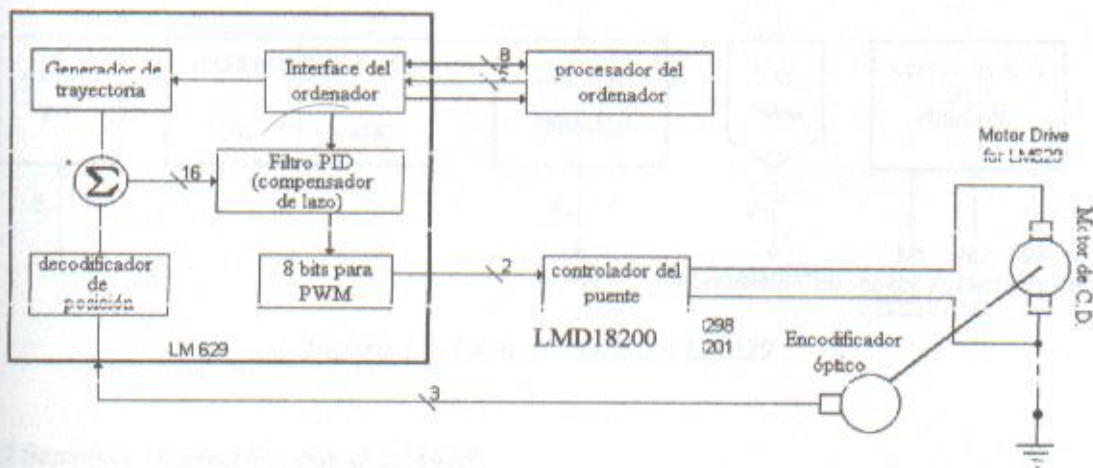


Figura 2.3.3 Diagrama a bloques del control de un motor con el LM629.

A pesar de que el LM629 tiene un bus periférico, y que este a su vez puede ser programado por una computadora. Esta aplicación no es recomendable por presentar un punto de iniciación concreto para la programación de estos controladores de movimiento de precisión, además de la posible falta de sincronía entre la PC y el LM629 y de las impedancias a considerar.

2.3.6 Arquitectura del LM629

A continuación se presentan algunas de las características del LM629 (figura 2.3.4):

- Posee una **ROM** de 1 KB x 16 bits que usa instrucciones de 16 bits, donde el algoritmo de control es almacenado
- Se cuenta con un seguidor de secuencias **PLA** que decodifica esas instrucciones y da transferencia de señales de datos cronometradas para un bus de datos de 16 bits

- Tiene una **RAM** donde son almacenados los parámetros de trayectoria de doble palabra (32 bits). Cuenta con un registro de posición de 32 bits de suficiente rango dinámico, el cual también es usado para valores de velocidad y aceleración
- Por último, hay una **ALU** de 32 bits que se usa para soportar multiplicaciones de 16 x 16 bits para valores del error y coeficiente del filtro PID.

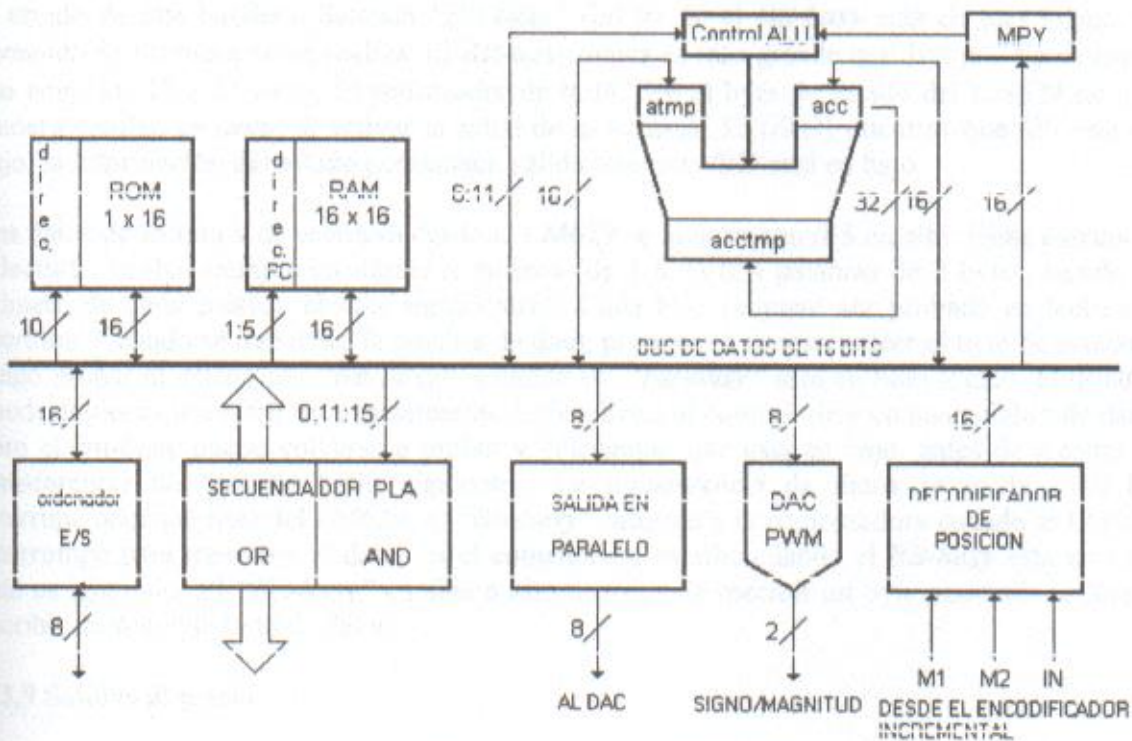


Figura 2.3.4 Arquitectura del LM629.

2.3.7 Sensores requeridos por el LM629

El LM629 tiene una interfaz para capturar tres señales de la posición (mediante el encoder). Dos señales cuadradas de entrada, A y B son decodificadas para obtener la información de la posición y la dirección, y una tercera señal de entrada opcional IN usada para capturar la posición una vez por revolución. Los cuatro estados de la señal cuadrada de posición son decodificados por el LM629. Las señales de entrada tienen un incremento de cuatro tiempos para la resolución de la posición en el número de líneas del encoder. Las "N" líneas del encoder pueden ser decodificadas como una posición de "4N" cuentas, en dicho dispositivo. El secuenciador **PLA** decodifica la base de las señales A, B e IN y determina si el eje del motor se mueve hacia adelante o hacia atrás, o permanece inmóvil. Dentro del secuenciador se tiene un contador ascendente de 16 bits que mantiene la posición actual del eje del motor una vez por revolución, cuando las tres señales A, B e IN están en bajo.

2.3.8 Operaciones de lectura y escritura

El microcontrolador envía los comandos al LM629 vía el puerto de entrada/salida, cuando la terminal de entrada 16 (Port Select /PS) está en bajo lógico. El código del comando deseado se aplica a la línea del puerto paralelo, cuando la terminal de entrada 15 (/WR) es activada. El byte de comando pasa los valores al LM629, al subir la señal de entrada /WR. Cuando el byte del comando de escritura es necesario, primero se lee el byte de estado y enseguida se prueba el estado de una bandera, llamada "*Bit-busy*" (bit 0). Si el *Bit-busy* está en uno lógico, el comando de escritura no se realiza. El *Bit-busy* nunca es más grande que 100 μ s, típicamente cae entre los 15 a 25 μ seg. El procesador de la PC lee el byte de estado del LM629 de una manera similar, es decir, al activar la señal de la terminal 13 (/RD) mientras que /PS está en bajo, la información del estado permanece válida mientras /RD esté en bajo.

Los datos de lectura y de escritura desde el LM629 se aplican con /PS en alto. Estas escrituras y lecturas se dan siempre en números enteros (de 1 a 7) con palabras de 2 bytes, siendo el primero de cada palabra el más significativo. Cada byte requiere ser probado en lectura o escritura. Cuando se transfiere la palabra de dato, primero es necesario leer el byte de estado y luego probar el estado del "*Bit-busy*". Cuando el "*Bit-busy*" esta en bajo lógico, el usuario puede entonces transferir secuencialmente ambos bytes al comprimirse en una palabra de dato, pero el *Bit-busy* puede volverse a probar y determinar que está en bajo, antes de atender la transferencia del par de bytes siguientes. La transferencia de datos se envía vía las interrupciones internas del LM629, el "*Bit-busy*" informa a la computadora cuando el LM629 interrumpe para transferir el dato, si el comando se escribe cuando el *Bit-busy* esta en alto, este es ignorado. El "*Bit-busy*" cambia a alto después de escribir un byte comando, o leer o escribir un segundo byte de datos.

2.3.9 Salidas al motor

El LM629 provee 8 bits de señales de salida PWM de signo y magnitud (figura 2.3.5), las cuales llegan directamente al puente H. Por otra parte la elección de los motores es un punto de suma importancia en el desarrollo de este proyecto ya que vienen a determinar parámetros en el análisis estructural tales como par y peso, a demás de que son los generadores de la fuerza del manipulador. Los motores que se han elegido son del tipo *SWF de ángulo recto* producidos por *VALEO*, tienen un consumo de voltaje de 24 Volts y de 0.5 A sin que la carga intervenga considerablemente en estos parámetros debido a que se trata de motoredutores, un peso de 0.570 Kg y un par de 1.3 N-m. La velocidad de este tipo de motor es de 35 rpm. Como se puede observar, es ligero, posee el par suficiente para mover los eslabones, brinda la ventaja de la flecha con el ángulo lo cual facilita la instalación de los mismos en el manipulador.

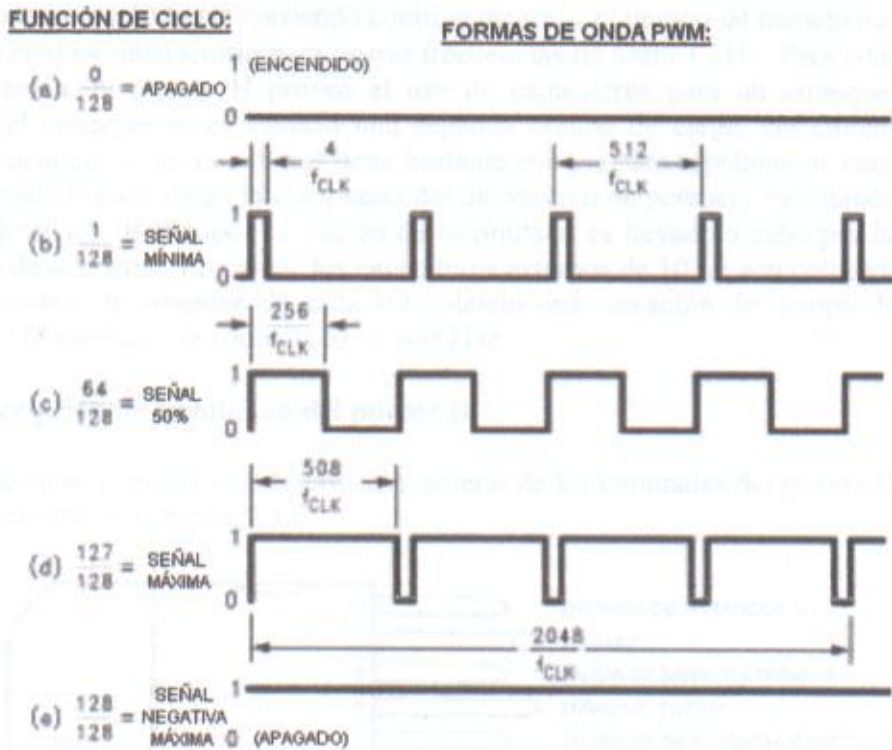


Figura 2.3.5 Señal PWM.

2.3.10 Etapa de Potencia (puente H)

Para la etapa de potencia se emplea el circuito integrado LMD18200, que es un puente H el cual se describe a continuación.

2.3.11 Características del puente H

- Maneja 3 A continuos de salida.
- Opera voltajes de alimentación de 12 V hasta 55 V.
- Resistencia característica baja 0.33 Ω por switch.
- Compatible con entradas TTL y CMOS.
- No tiene corrientes de sobretiro.
- Bandera de advertencia térmica en la salida a 145 °C.
- Cierre térmico a 170 °C.
- Diodos de alerta interna.
- Protección contra cortos en la carga.
- Bomba de carga interna con capacidad externa de arranque.

La compuerta de este dispositivo puede manejar aproximadamente 8 V, más grande que el voltaje de suministro. Para alcanzar esto una bomba de carga interna, es usada para dar el manejo de voltaje a la compuerta un capacitor interno, el cual es alternadamente aterrizado y es cargado sobre 14 V, que es entonces prendido para la terminal V_S ; sin embargo, da un manejo de voltaje grande de la compuerta de la misma, esta acción de prendido es controlada

para un oscilador de 300 kHz corriendo continuamente, y el tiempo de manejo de voltaje es de 20 μ seg, el cual es satisfactorio para operar frecuencias de hasta 1 kHz. Para altas frecuencias de conmutación, el puente H provee el uso de capacitores para un arranque externo. El principio del arranque es en esencia una segunda bomba de carga, sin embargo, un valor grande de capacitor es usado, el cual tiene bastante energía para rápidamente cargar la entrada de la capacitancia parásita en la compuerta del dispositivo de potencia, resultando mucho más rápida la elevación de tiempo. La acción de conmutado es llevado a cabo por la potencia de conmutado de sí mismo. Entonces, los capacitores externos de 10 nF son colocados a la salida de las terminales de arranque de cada lado, dando una elevación de tiempo de 100nseg, y permitiendo frecuencias de conmutado de 500 kHz.

2.3.12 Descripción de terminales del puente H

A continuación se presenta una descripción general de las terminales del puente H LMD18200 el cual se muestra en la figura 2.3.6.



Figura 2.3.6 Circuito integrado LMD18200.

Terminal 1

Entrada de arranque 1: Es la terminal del capacitor de arranque para la mitad del puente H de arranque 1. El capacitor que se recomienda es de 10 nF y se conecta entre los terminales 1 y 2.

Terminal 2

Salida 1: Salida de la mitad del puente H de arranque 1.

Terminal 3

Entrada de Dirección: Esta entrada controla la dirección del flujo de corriente entre la salida 1 y la salida 2, y por lo tanto, también la dirección de un motor con carga (Ver tabla 2.3.1).

Terminal 4

Entrada para frenado: Esta entrada se utiliza para frenar un motor a través de sus terminales. Cuando se desea frenar, esta entrada toma un nivel lógico alto, y esto es también necesario para aplicar un alto lógico a la entrada PWM (Ver tabla 2.3.1).

Terminal 5

Entrada PWM: Esta entrada se utiliza de acuerdo al formato de la señal PWM. (Ver tabla 2.3.1).

Terminal 6

Es alimentación de voltaje para la terminal V_s .

Terminal 7

Conexión de potencia para tierra / sensor: Esta terminal es el retorno a tierra para el voltaje de los transistores DMOS del puente H. La corriente a través del puente H puede ser medida por medio de una pequeña resistencia de 0.1Ω , desde dicha terminal hasta a la tierra del suministro de alimentación.

Terminal 8

Señal de tierra: Esta es el retorno a tierra para la circuitería interna lógica, usada para el control de conmutado de PWM desde el puente H.

Terminal 9

Salida para bandera térmica: Esta terminal provee la señal de advertencia de la bandera de la temperatura, y dicha terminal se va a bajo activo a los 145°C .

Terminal 10

Salida 2: Salida número 2 de la mitad del puente H de arranque 2.

Terminal 11

Entrada de arranque 2: Esta es la terminal del capacitor de arranque para la mitad del arranque 2 del puente H.

Tabla 2.3.1. Tabla de verdad del LM18200.

PWM	DIRECCIÓN	PARO	ACTIVACIÓN DE SALIDAS DEL MANEJADOR
H	H	L	FUENTE 1, SUMIDERO 2
H	L	L	SUMIDERO 1, FUENTE 2
L	X	L	FUENTE 1, FUENTE 2
H	H	H	FUENTE 1, FUENTE 2
H	L	H	SUMIDERO 1, SUMIDERO 2
L	X	H	NINGUNO

Bandera de advertencia térmica. La salida del terminal 9 es un transistor de colector abierto. Este permite un alambrado de una OR de una bandera de salida de cuidado térmico desde el puente H, y permite al usuario colocar los niveles lógicos altos del balance de la señal de salida, para emparejar los requerimientos del sistema. Esta salida típicamente maneja la interrupción de salida de un controlador del sistema. Y la rutina del servicio de interrupción esta diseñada para tomar apropiadamente los pasos, tales como reducir las corrientes de carga o inicializar un sistema ordenado de cierre.

2.3.13 Encoders

En el caso del LM629N utiliza las señales de encoders como retroalimentación. A continuación se muestra un poco de información de este tipo de encoders (figura 2.3.7).

El encoder es de tipo incremental, entrega dos señales de cuadratura y una de índice, las señales de cuadratura informan de la posición y sentido de giro del motor, la señal de índice toma el nivel bajo una vez por revolución. La figura muestra la gráfica de estas señales.

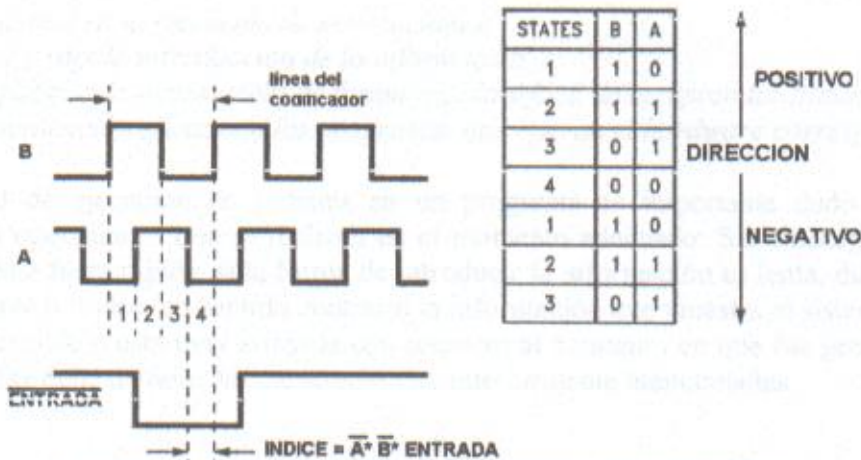


Figura 2.3.7 Señal generada por un encoder incremental.

Los encoder entregan un número determinado de cuentas por vuelta, 100, 600, 700, 1200 cuentas por vuelta. Se ha decidido utilizar encoder's del tipo incremental debido a que son los que requiere el LM629 incluso aunque nos se utilice este circuito es necesario un encoder del tipo incremental debido a que para el control se necesita una retroalimentación que proporcione la posición y la dirección del motor. No se utilizó potenciómetros por su inexactitud, ni resolver's, LVDT o Inductosym por la dificultad y costo de su adquisición.

El encoder que se utiliza en este proyecto es un encoder de tipo incremental ES5, con una resolución de 1024 pulsos por revolución, distribuido por US Digital.

2.4 Interfaz Usuario-Manipulador

Una de las partes dentro de cualquier sistema que recibe órdenes de un ser humano para llevar a cabo una función de cualquier índole, es la interfaz de comunicación que existe entre ellos (*Usuario – Sistema*). Para ser más precisos se entenderá como interfaz un programa de computadora a través del cual se le proporciona al sistema la información necesaria para llevar a cabo sus funciones.

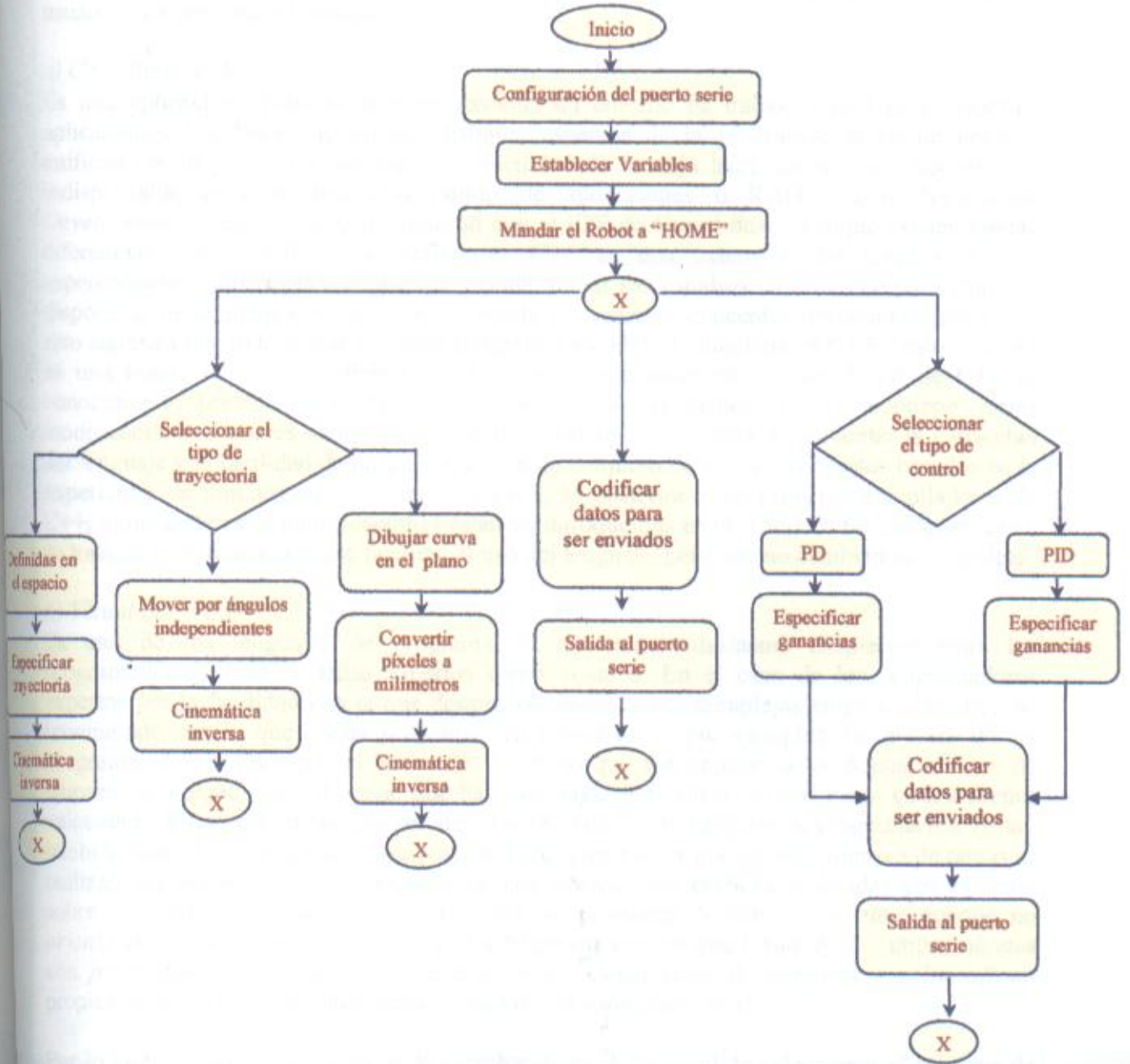
Las características de la interfaz son:

- *Velocidad en la ejecución de instrucciones*
- *Fácil y rápida introducción de la información*
- *Despliegue de información de forma rápida y fácil de comprender (intuitiva)*
- *Comunicación eficaz con los programas que operan el hardware correspondiente*

La velocidad de ejecución de órdenes en un programa es importante dado que de ello dependen las operaciones que se realicen en el momento adecuado. Sin embargo, no tendría sentido que ésta fuera rápida, si la forma de introducir la información es lenta, difícil y con un formato complejo o visto en sentido contrario la información que muestra el sistema al usuario no es comprensible o esta muy atrasada con respecto al momento en que fue generada, por lo tanto la interfaz debe de tener las características anteriormente mencionadas.

2.4.1 Diagrama de flujo

El siguiente diagrama de flujo muestra la secuencia que lleva a cabo la interfaz, el cual se realiza por medio de eventos que captura Windows y los envía al programa.



2.4.2 Selección del lenguaje de programación

A continuación se describen brevemente las características de dos lenguajes de programación comerciales, con la intención de seleccionar uno en el cual desarrollar la interfaz gráfica del usuario para controlar el manipulador.

a) C++ Builder 5.0

Es una aplicación Windows que proporciona un entorno de trabajo visual para construir aplicaciones Windows que integra distintos aspectos de la programación en un entorno unificado o integrado. La integración y facilidad de manejo hace que sea una herramienta indispensable para el desarrollo rápido de aplicaciones o RAD (*Rapid Application Development*). Guarda una gran similitud con el IDE de *Visual Basic*, aunque existen ciertas diferencias. De acuerdo a su definición, C++ es una extensión del lenguaje C, un superconjunto, esto implica para quién decide iniciar sus estudios en C++ que para empezar dispone de un lenguaje completo como soporte a los nuevos conceptos introducidos por C++, esto significa que todo lo que se puede programar en ANSI-C funciona en C++ y esto a su vez es una buena razón para afirmar la importancia que tiene en el aprendizaje de C++ el conocimiento previo del lenguaje C. Como ya se ha dicho, el C++ contiene varias modificaciones menores sobre el C original. Normalmente se trata de aumentar la capacidad del lenguaje y la facilidad de programación en un conjunto de detalles concretos basados en la experiencia de muchos años. Como el ANSI C es posterior a los primeros compiladores de C++, algunas de estas modificaciones están ya introducidas en el ANSI C. En cualquier caso, se trata de modificaciones que facilitan el uso del lenguaje, pero que no cambian su naturaleza.

b) Visual Basic 6.0

Es uno de los lenguajes de programación que más entusiasmo despiertan entre los programadores de PC's, tanto expertos como novatos. En el caso de los programadores expertos por la facilidad con la que desarrollan aplicaciones complejas en poquísimo tiempo (comparado con lo que cuesta programar en *Visual C++*, por ejemplo). En el caso de los programadores novatos por el hecho de ver de lo que son capaces a los pocos minutos de empezar su aprendizaje. El precio que hay que pagar por utilizar *Visual Basic* es una menor velocidad o eficiencia en las aplicaciones. *Visual Basic* es un lenguaje de programación visual, también llamado lenguaje de 4ª generación. Esto quiere decir que un gran número de tareas se realizan sin escribir código, simplemente con operaciones gráficas realizadas con el ratón sobre la pantalla. *Visual Basic* es también un programa basado en objetos, aunque no orientado a objetos como C++ o Java. La diferencia está en que *Visual Basic* utiliza objetos con propiedades y métodos, pero carece de los mecanismos de herencia y polimorfismo propios de los verdaderos lenguajes orientados a objetos como Java y C++.

Por lo tanto tomando en cuenta estas características se ha decidido seleccionar el lenguaje de programación C++ Builder 5.0 como el más indicado para desarrollar la interfaz gráfica. La decisión se tomó principalmente por los siguientes puntos:

- Pese a que se afirma que el desarrollo de programas es más sencillo en Visual Basic; ya que se ha con ambos lenguajes, desarrollar programas en C++ Builder no resulta una tarea más complicada que en Visual Basic y en este aspecto no se observa una ventaja en ninguno de los ambientes.

- C++ Builder trabaja con C++, el cual es totalmente compatible con C, y tiene destacadas ventajas con el Basic usado en Visual Basic en el aspecto de que el primero es un lenguaje orientado a objetos el cual nos proporciona ciertos mecanismos muy importantes en la programación actual (herencia y polimorfismo).
- Aunada a las anteriores la principal razón por la que se decidió utilizar C++ Builder es que se tiene mucha más experiencia en el manejo de C y C++, además de que es un lenguaje de programación con un ambiente agradable y por el cual se tiene más empatía a diferencia de Visual Basic.

A continuación se mencionan algunas de las características con que cuenta C++ Builder.

- Entorno de Trabajo. "IDE" (Entorno de Desarrollo Integrado)
- Sistema de Ayuda.
- Componentes
- La VCL (Biblioteca de Componentes Visuales)
- Programación Orientada a Objetos en C++
- Tratamiento de excepciones

2.4.3 Presentación preliminar del ambiente gráfico

En la figura 2.4.1 se presenta una propuesta de cómo se conforma la interfaz gráfica usuario - PC para el control y manejo del brazo robot manipulador.

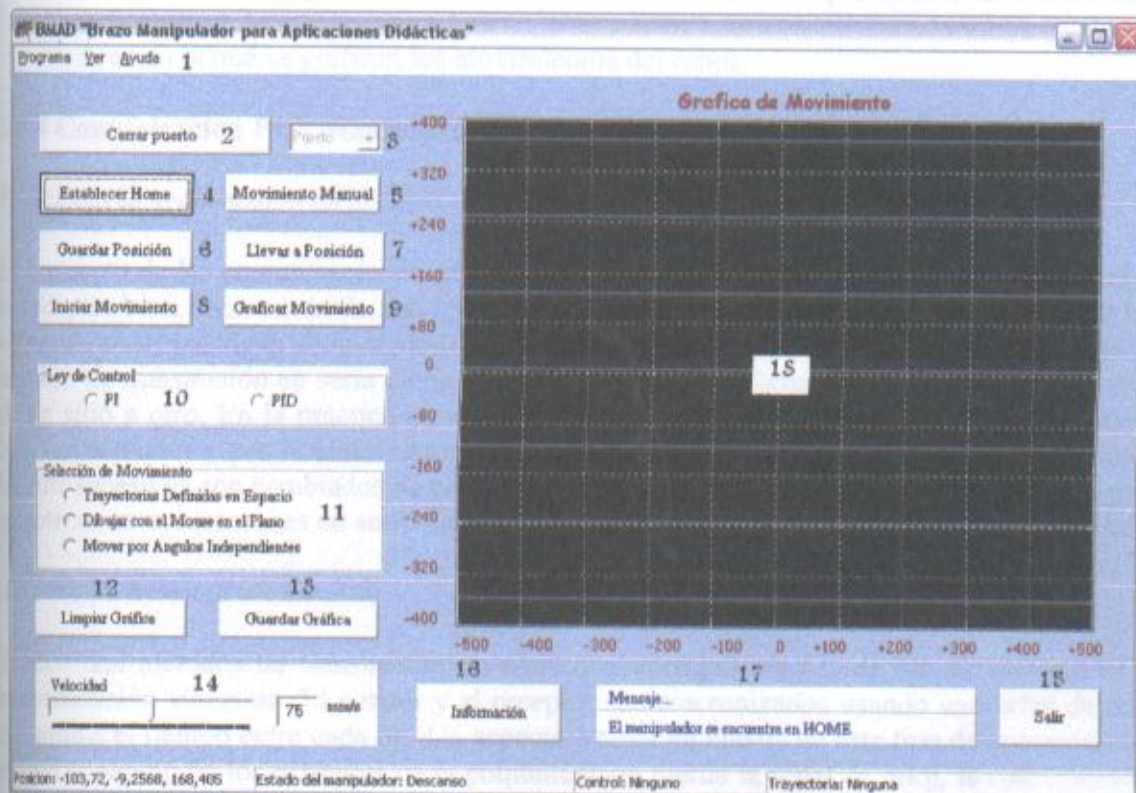


Figura 2.4.1 Presentación preliminar de la interfaz gráfica.

La descripción general de cada control con el que cuenta la interfaz es la siguiente.

1. Menú desplegable de la interfaz.
2. Establece comunicación con el puerto serie de la computadora, en caso de que haya una comunicación cierra el puerto.
3. Selección del puerto serie de la computadora.
4. Mueve el brazo a su posición de descanso "HOME".
5. Despliega una nueva ventana en donde se permite el movimiento manual del manipulador.
6. Permite guardar en uno de los registros la posición actual del brazo.
7. Lleva el brazo a una posición previamente guardada en uno de los registros destinados para esto.
8. Se inicia el movimiento del brazo de acuerdo a los parámetros previamente establecidos, los cuales son: ley de control, parámetros, trayectoria y velocidad.
9. Se grafica la trayectoria deseada el manipulador o los puntos de la trayectoria generada.
10. Permite modificar las constantes del control o motores.
11. Selección del tipo de trayectoria a seguir.
12. Limpia la zona de gráficas.
13. Guarda la gráfica presente en un archivo de imagen (*.bmp, *.jpg, *.gif).
14. Permite modificar la velocidad lineal del efector final del manipulador.
15. Salir del programa.
16. Muestra información acerca de la posición y estado actual del manipulador.
17. Ventana de mensajes que muestra el estado de la función que se esta realizando en el momento.
18. Área en la que se grafican los movimientos del robot.

2.4.4 Comunicación PC - Manipulador

Principios de transmisión en serie

El concepto de las comunicaciones en serie es el siguiente: Los datos son enviados del emisor al receptor bit a bit a través de una línea única o circuito. El puerto serie toma 8, 16 o 32 bits en paralelo del bus de la computadora y los convierte en un flujo de 8, 16 o 32 bits en serie. El nombre de transmisión en serie viene de aquí, cada bit de información es transmitido en serie de un sitio a otro. En la práctica en una transmisión en serie algunos bits pueden perderse o bien ser añadidos y por lo tanto alterar el resultado final. Si un bit se pierde o se añade, todos los bits sucesivos son cambiados de posición y se obtiene un resultado incorrecto. Así que para establecer comunicaciones en serie fiables se deben corregir posibles errores en la transmisión.

Métodos de transmisión en serie

Existen dos métodos de transmisión en serie que corrigen errores de bit: El primero es la **comunicación síncrona**. El emisor y el receptor son sincronizados usando una señal de reloj que indica el tiempo entre cada bit. Un aspecto a tener en cuenta en este tipo de comunicación es que si alguno de los extremos de la comunicación pierde la señal de reloj, la comunicación finaliza. El método alternativo, conocido como **comunicación asíncrona**, es añadir

marcadores dentro del flujo de bits para seguir la pista a cada bit de datos. Si se introduce un bit de comienzo que indica el inicio de un bloque de bits, la posición de cada bit puede ser determinada temporizando los bits en periodos regulares. Al utilizar bloques de pequeño tamaño (de 5 a 8 bits) no hay tiempo para que el temporizador se desincronice. El único factor importante es que receptor y emisor tengan configurada la misma velocidad en el puerto. Algunas veces se añade un bit de paridad entre el último bit de la palabra (conjunto de bits) y el bit de parada para comprobar la integridad de los datos.

Estándar RS-232 (Tabla 2.4.1)

Los niveles de voltaje descritos en el estándar son los siguientes:

<i>Tabla 2.4.1 Características eléctricas del estándar RS-232.</i>			
Señales de datos	"0"	"1"	
Emisor (necesario)	de 5 a 15	de -5 a -15	Volts
Receptor (esperado)	de 3 a 25	de -3 a -25	Volts
Señales de control	"Off"	"On"	
Emisor (necesario)	de -5 a -15	de 5 a 15	Volts
Receptor (esperado)	de -3 a -25	de 3 a 25	Volts

Puede verse que los voltajes del emisor y el receptor son diferentes. Esta definición de los niveles de voltaje compensa las pérdidas de voltaje a través del cable. Las señales son atenuadas y distorsionadas a lo largo del cable.

Características mecánicas (Tabla 2.4.2)

En el estándar no se hace referencia al tipo de conector que debe usarse. Sin embargo los conectores más comunes son el DB-25 (25 pines) y el DB-9 (9 pines).

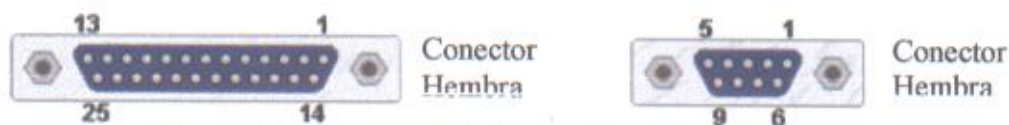


Figura 2.4.2 Diagrama de los conectores DB-25 y DB-9.

2.4.5 Puerto serie de la PC

El puerto serie de la PC es un dispositivo de entrada / salida compatible con el estándar RS-232-C. Casi todos las PC's tienen uno o dos puertos serie. Cada uno de estos puertos tiene un conector de nueve pines (algunas veces veinticinco) en la parte trasera de la computadora, de esta manera se pueden enviar y recibir datos a través del puerto serie.

La UART (Universal Asynchronous Receiver Transmitter)

La mayor parte de la electrónica del puerto serie esta implementada en un chip denominado UART. La misión principal de la UART es convertir los bytes del bus paralelo del PC a un flujo de bits en serie. Además cambia el voltaje utilizado en la señal para representar los bits y añade o extrae los bits de inicio y parada.

1.1.4.1. Descripción

La UART es un chip de microprocesador que se encarga de convertir los bytes del bus paralelo del PC a un flujo de bits en serie. Además cambia el voltaje utilizado en la señal para representar los bits y añade o extrae los bits de inicio y parada.

El chip UART es un chip de microprocesador que se encarga de convertir los bytes del bus paralelo del PC a un flujo de bits en serie. Además cambia el voltaje utilizado en la señal para representar los bits y añade o extrae los bits de inicio y parada.

La descripción de este chip de microprocesador se encuentra en el manual de usuario de este chip de microprocesador. Este manual de usuario describe el funcionamiento de este chip de microprocesador.

1) Descripción

Este chip de microprocesador se encarga de convertir los bytes del bus paralelo del PC a un flujo de bits en serie. Además cambia el voltaje utilizado en la señal para representar los bits y añade o extrae los bits de inicio y parada.

1) Descripción

Descripción

Este chip de microprocesador se encarga de convertir los bytes del bus paralelo del PC a un flujo de bits en serie. Además cambia el voltaje utilizado en la señal para representar los bits y añade o extrae los bits de inicio y parada.

2) Descripción

Descripción

Este chip de microprocesador se encarga de convertir los bytes del bus paralelo del PC a un flujo de bits en serie. Además cambia el voltaje utilizado en la señal para representar los bits y añade o extrae los bits de inicio y parada.

CAPÍTULO 3. CONSTRUCCIÓN DEL PROTOTIPO

La parte de construcción del manipulador se divide en tres partes principales:

- Mecánica.
- Electrónica.
- Programación.

Esto con el fin de poder describir de la mejor manera posible todo lo ocurrido durante la construcción del manipulador.

3.1 Construcción Mecánica

La construcción del manipulador en lo que respecta a la parte mecánica se llevó a cabo según se había planeado en el cronograma de la asignatura de TT2. Cabe señalar que los diseños de los componentes del manipulador no sufrieron cambios trascendentes, a lo más se agregaron algunos barrenos para poder acoplar algún elemento electrónico.

El maquinado de todas las piezas se llevó a cabo en el taller de máquinas de la UPIITA, para ello se utilizaron máquinas y herramientas convencionales (torno y fresa manuales) y la máquina de CNC (Control Numérico Computarizado).

La descripción de la parte mecánica se divide a través de seis puntos principales, con ello se pretende tener una mejor asimilación y comprensión de ésta, los puntos son: problemas, solución, cambios, adaptaciones, mejoras y finalmente el ensamble final.

1) Problemas

Entre los principales problemas que se presentaron durante la construcción del manipulador se pueden señalar los siguientes:

- 1) El ensamble del primer eslabón con la base giratoria.

Descripción:

Cuando se realizó el diseño se planeo ensamblar el primer eslabón con la base giratoria a través de un perfil L de $\frac{3}{4}$ " x $\frac{3}{4}$ " de aluminio, utilizando tornillos allen de cabeza plana de $\frac{1}{4}$ x $\frac{1}{2}$ " pero desafortunadamente, al realizar el avellanado en el perfil L la cantidad de material que resultaba después de realizarlo ya no era suficiente para poder realizar el ensamble.

- 2) El ensamble de la base giratoria con el eje de transmisión.

Descripción:

En el diseño se decidió utilizar una pieza que ensamblara la base giratoria con el eje de transmisión, pero una vez teniendo la pieza no fue posible realizar el ensamble ya que se

requería realizar un avellanado en está para la colocación de los tornillos de sujeción pero por las dimensiones de la pieza esto resultaba imposible.

2) Solución

A partir de los dos problemas que se presentaron en la parte mecánica se decidió diseñar una pieza nueva que permitiera realizar el ensamble del primer eslabón con la base giratoria y que a su vez permitiera a la base giratoria el ensamble con el eje de transmisión, esta pieza puede observarse en la figura 3.1.1. La pieza diseñada permite el ensamble correcto de los elementos y además reduce esfuerzos en los mismos.



Figura 3.1.1 Acoplamiento entre base giratoria y el eje de la base fija.

3) Cambios

Se cambió el material de los ejes de los motores, de aluminio a latón, ya que se requería una barra hexagonal para ser acoplada al motor (figura 3.1.2) pero en las casas de materiales esta barra no es muy comercial (en aluminio), por ello se tuvo que utilizar la barra hexagonal de latón, como se muestra en la figura 3.1.3, al utilizar la barra de latón pudimos darnos cuenta que las propiedades mecánicas no se ven afectadas ya que se realizó un análisis estático y la barra de latón presentó un buen comportamiento, además el desgaste de la barra de latón es menor que el que presenta la barra de aluminio.



Figura 3.1.2 Flecha del motor

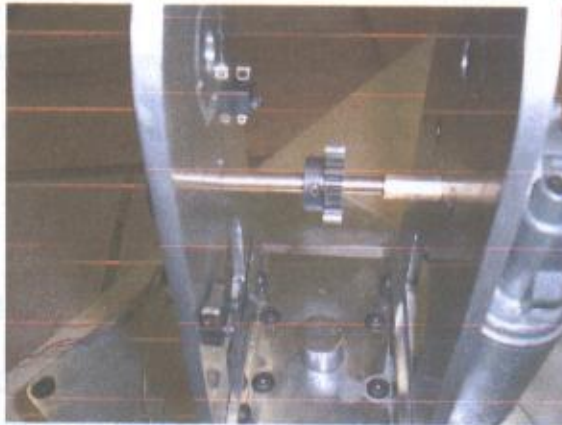


Figura 3.1.3 El material de la flecha del motor es latón.

4) Adaptaciones

La adaptación de componentes al manipulador fue tanto de elementos mecánicos como de elementos electrónicos y esto se enuncian a continuación:

1. Limit Switch (Sensores de límite).

Se adaptaron dos sensores de límite en vez de uno solo en los eslabones del manipulador, esto para delimitar los inicios y finales de carrera de los motores, tal y como se muestra en la figura 3.1.4.

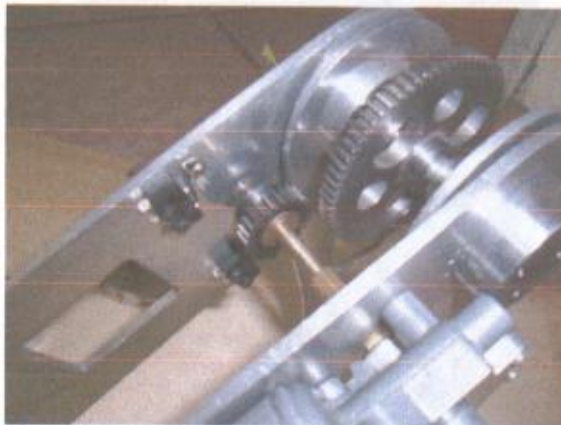


Figura 3.1.4 Acoplamiento de los sensores de límite.

2. Efecto final.

Se adaptó un efecto final de tenazas en el extremo del tercer eslabón del manipulador.

3. Caja para circuitos impresos.

Se adaptó una caja cuya función es contener las placas electrónicas.

5) Mejoras

Las mejoras que se han incluido al manipulador son:

Efector final: Se adaptó un efector final al manipulador que le permite sujetar piezas pequeñas (no mayores de 10mm x 10mm x 10mm) en donde las mordazas pueden ser intercambiables.

6) Ensamble

El ensamble del manipulador es como se muestra en la figura 3.1.5, incluye:

- ◆ 3 motores de CD de 1.3 N/m
- ◆ 3 engranes rectos de 72 dientes paso diametral 24
- ◆ 3 engranes rectos de 24 dientes paso diametral 24
- ◆ 1 balero de carga axial.
- ◆ 2 baleros de carga radial R12
- ◆ 2 baleros de carga radial R10
- ◆ 1 balero de carga axial.
- ◆ Tornillos allen de cabeza plana de diferentes medidas para realizar el ensamble.
- ◆ Tornillos allen de cabeza de botón (varias medidas)
- ◆ Peso aproximado de 12.5 kg.



Figura 3.1.5 Ensamble físico del manipulador.

3.2 Construcción electrónica

En cuanto al desarrollo de la electrónica, se han realizado algunos cambios en comparación con los diseños propuestos para la asignatura de TT1, esto es debido a problemas que se presentaron durante las pruebas que se realizaron antes de llevar a cabo la construcción de las placas electrónicas.

Uno de los primeros problemas que se presentó y que fue la causa por la que se decidió cambiar el microcontrolador AVR ATmega16L por un AVR AT90S8515 fue la comunicación PC – Microcontrolador. El ATmega16L hace uso de una USART (Universal Synchronous and Asynchronous serial Receiver and Transmitter) para llevar a cabo la comunicación entre él y la PC, el problema con este AVR, es que dos registros comparten la misma localidad de memoria. El registro UBRRH que es usado para configurar la velocidad a la cual se reciben y transmiten los datos del AVR a la PC ocupa la misma localidad de memoria que el registro UCSRC que sirve para configurar el formato en el cual se envían los datos, es decir, cantidad de bits, bit(s) de paridad, bit(s) de parada, etc. Existe un bit (URSEL) que se ocupa para definir en cual de los dos registros se va escribir, sin embargo no funcionó correctamente, únicamente se enviaba o recibía basura, esto debido a que entraban en conflicto estos dos registros, ya que uno modificaba al otro.

Después de hacer múltiples pruebas para tratar de que respondiera adecuadamente el ATmega16L se llegó a la conclusión de que era demasiado ya el tiempo invertido y que era mejor cambiar el microcontrolador debido a que no se había obtenido una respuesta satisfactoria.

El cambio fue por un AVR AT90S8515 (el cual hace uso de una UART), se realizó un rediseño de las placas y si bien es cierto que se invirtió un tiempo en los diagramas esquemáticos y los diagramas para las placas fenólicas, también es cierto que en poco tiempo se logró realizar la comunicación PC – AVR.

Cabe resaltar que en comparación con el ATmega16L, el AT90S8515 es menos susceptible a dañarse, debido a que no cuenta con un ADC, además dicho aditamento no era necesario para este proyecto; incluso es un poco más económico.

Una vez resuelto el problema anterior se pudieron llevar a cabo los siguientes objetivos:

- Programación de la recepción de datos en el microcontrolador
- Diseño del programa para la comunicación PC–AVR
- Diseño del programa para la interpretación de los datos

El problema a resolver se tenía bien definido, así como los pasos a seguir para resolverlo:

- Elaborar el programa en lenguaje ensamblador que permita la recepción de datos en el microcontrolador, de forma secuencial.
- Una vez que se obtenga un programa sin errores se precede a probar con la PC, transfiriendo datos a través del puerto serie en forma bidireccional.

- 3.2 ➤ Cuando se tenga una comunicación correcta entre PC y Microcontrolador el siguiente paso es elaborar código que permita interpretar los datos enviados por la PC.

Una vez que se llevó a cabo el rediseño de las placas se procedió a hacer pruebas para comunicar el AT90S8515 con el LM629N. Nuevamente se encontraron problemas debido a que se invirtió demasiado tiempo en comunicar el AT90S8515 con el LM629N, esto debido a que la programación de los algoritmos y sincronizar bien los dos elementos es tarea difícil de realizar.

Cabe mencionar que aunque el LM629 tiene un bus periférico, y puede ser programado por una computadora (procesador). Esta aplicación no es recomendable por la diferencia de velocidades de procesamiento que existe entre la PC y el LM629N, además de otros factores como lo es la impedancia, el ruido, la atenuación de las señales de control, entre otros.

3.2.1 Tarjeta modulo de comunicaciones y control general.

Hardware Interfaz de comunicaciones

Para el diseño de la interfaz y el control de los LM629N se seleccionaron los siguientes componentes:

- MAX232
- AT90S8515 (Microcontrolador)

Puerto Serie

Es el elemento a partir del cual se reciben los datos de la computadora, utilizando el protocolo RS232C, con una velocidad de 9600 baudios.

MAX232 (Transceiver)

Es el transceiver encargado de adaptar la señal proveniente del puerto (12V) a un nivel lógico aceptable para la lógica de los microcontroladores (5V).

AT90S8515 (Microcontrolador)

Es el elemento encargado de manejar las comunicaciones entre el puerto de la computadora y el hardware, se encarga de recibir las señales, provenientes del puerto a través de la UART full-duplex que viene integrada dentro del microcontrolador, administrar los datos de aceleración, velocidad, posición y las constantes K_P , K_I y K_D , además de direccional dichos datos a cada uno de los LM629N para llevar a cabo el movimiento del manipulador.

3.2.2 Tarjeta modulo de control PID

Hardware Modulo de Posicionamiento

Como se ha especificado se cuenta con un sistema de posicionamiento que debe de alcanzar y mantener la posición para completar satisfactoriamente la trayectoria que se va a describir, para tal propósito se diseño el módulo de posicionamiento para el cual se han elegido los siguientes componentes:

- LM629N (Controlador de movimiento de precisión)
- LMD18200T (Puente H)

LM629N (Controlador de posición de Precisión)

Este dispositivo es un microcontrolador de control de posición, diseñado para controlar los servomotores de CD que provean una señal de retroalimentación de posición en cuadratura incremental; recibe las consignas de posición provenientes de microcontrolador a través de un código especificado por el fabricante, y se encarga de generar una curva de velocidades que es la que determinara el comportamiento del sistema.

LMD18200T (Controlador Puente H)

Este dispositivo es un puente H de alto voltaje y alta corriente, diseñado para aceptar entradas lógicas TTL estándar y manejar cargas inductivas como son solenoides, relevadores motores a pasos y de CD, y es el encargado de suministra corriente controlada a los motores de posicionamiento.

3.2.3 Descripción del funcionamiento del sistema

Se inicia el sistema con un pulso de reset proveniente de la tarjeta de comunicaciones, el cual hace un reset por hardware al microcontrolador principal y después un reset por software a los controladores de posición (LM629N).

Una vez inicializado el sistema se inicia un recorrido para llevar al manipulador a la posición de HOME, a partir de ésta posición se realizan los movimientos necesarios; una vez que la trayectoria ha comenzado se verifica si esta se ha alcanzado, en caso afirmativo, ponemos la bandera de estado a 1 para indicar que el sistema esta listo para recibir la siguiente posición, para lo cual escanea el puerto de comunicaciones en espera de datos.

Si se tienen datos en el puerto, la bandera de estado se pone a cero indicando que se esta en proceso, después de esto los datos se reciben, se cargan los parámetros del filtro PID, y finalmente la trayectoria descrita por los datos recibidos es cargada a los controladores de posición. Con lo anterior realizado se procede a esperar que la posición se haya alcanzado, y si esto es así, se pone la bandera de estado a 1 nuevamente para poder recibir una nueva posición.

3.2.4 El LM629N: ventajas y desventajas de usarlo

El encapsulado LM629N es un microcontrolador PID programable (MPID), que incorpora en un solo dispositivo las funciones de adquisición de datos y sistema de control de movimiento.

Usando este elemento de control se ahorra mucho trabajo y hace que el control de posición o velocidad de un actuador sea mucho más fácil que con varios elementos discretos en un sistema de control.

Las características por las que se ha de emplear este dispositivo para controlar la velocidad del robot, son las siguientes:

- Decodificador de posición integrado
- Filtro de compensación digital PDI
- Generador de perfiles de trayectoria
- Salida PWM de magnitud variable con bit de signo

El dispositivo programable LM629N se comunica con el microcontrolador AT90S8515 por medio de 8 bits de datos bidireccionales y seis líneas de control en las que están incluidas la petición de interrupción y el reset de hardware. En operación, para iniciar un movimiento, el microcontrolador AT90S8515 descarga datos de aceleración, velocidad y posición deseada al generador de trayectorias del dispositivo LM629N. A cada intervalo de muestreo se obtienen los valores actuales de posición y velocidad, los cuales son usados para calcular los nuevos puntos. La posición actual del motor se detecta mediante la utilización de un encoder tipo incremental.

Así se puede calcular la señal de error, cuando se subtrae la información obtenida del encoder incremental y la posición o velocidad requerida, esta señal es enviada al filtro compensador del lazo de control. El compensador esta formado por tres términos, filtro PID (proporcional, Integrador y Derivativo), el cual se implementa digitalmente.

a) Ventajas

De entre las ventajas que tiene el microcontrolador LM629N se pueden enumerar las siguientes:

- *Modularidad del microcontrolador*
Rapidez en el cambio de las constantes K_P , K_I y K_D para la sintonización del PID
- *Manejo de retroalimentación del sistema*
El uso de menos dispositivos electrónicos para la implementación de un PID digital
- *Reducción en el uso de espacio físico*

b) Desventajas

- *Difícil comprensión de la programación*
 Contar con una etapa de potencia sensible que responda a las pequeñas señales de control (PWM) que envía el microcontrolador.
- *Comprender los modos de operación*

3.2.5 Reglas de sintonización para controladores PID

Si se puede obtener un modelo matemático de la planta, es posible aplicar diversas técnicas de diseño con el fin de determinar los parámetros del controlador que cumpla con las especificaciones en estado transitorio y en estado estable del sistema de lazo cerrado. Sin embargo si la planta es tan complicada que no es fácil obtener su modelo matemático, tampoco es posible un enfoque analítico para el diseño de un controlador PID. En este caso, debemos recurrir a los enfoques experimentales para la sintonización de los controladores PID.

El proceso de seleccionar los parámetros del controlador que cumplan con las especificaciones de desempeño se conoce como sintonización del controlador.

Sintonización del filtro PID del LM629N

El LM629N es un controlador digital PID. El filtro del rizo de compensación de un controlador PID es usualmente sintonizado experimentalmente, especialmente si la dinámica del sistema no es bien conocida o definida.

La meta de sintonizar el filtro PID es llevar al sistema a una forma críticamente amortiguado.

La respuesta del filtro PID es la suma de tres términos, el término proporcional, el integral y el derivativo. Cinco variables intervienen en la respuesta. Estas cinco variables incluyendo los tres coeficientes de ganancia (K_P , K_I y K_D), el coeficiente límite de integración (i_I), y el coeficiente derivativo de muestreo (d_s). Al poner en marcha la sintonización del filtro para determinar los valores de estos coeficientes variables, siempre trataremos de encontrar valores que amortigüen críticamente el sistema de control.

Los valores de los coeficientes son determinados de una mejor forma por una aproximación experimental de dos pasos. En el primer paso, los valores de K_P , K_I y K_D (junto con i_I y d_s) son sistemáticamente variados hasta obtener una respuesta razonable. Métodos visuales y manuales son utilizados para evaluar el efecto de cada coeficiente en el comportamiento del sistema. En el segundo paso, una grafica de la respuesta al escalón del sistema, arrojará información detallada del amortiguamiento del sistema y los coeficientes del filtro obtenidos en el primer paso, son modificados para críticamente amortiguar el sistema.

Método Manual y Visual

En el primer paso, los valores de K_p , K_I y K_D (junto con i_l y d_s) son sistemáticamente variados hasta obtener una respuesta razonable. Métodos manuales y visuales son utilizados para evaluar los efectos de cada uno de los coeficientes en el comportamiento del sistema.

1.- Preparar el Sistema. La sección de inicialización del programa de sintonización del filtro es ejecutado para preparar el sistema para esta operación. Esta sección inicializa el sistema, prefiere los parámetros del filtro (K_p , K_I , $i_l = 0$, $K_D = 2$, $d_s = 1$) y comanda el rizo de control para mantener la flecha en la posición actual. Después de ejecutar el programa de inicialización de sintonización del filtro, los valores de la posición actual y deseada de la flecha son iguales a cero; la flecha está estacionaria. Cualquier desplazamiento de la flecha constituye un error de posición, pero con ambos coeficientes K_p y K_I en cero, el ciclo de control no puede corregir este error.

2.- Determinación del Coeficiente de ganancia Derivativa. El termino derivativo del filtro provee amortiguamiento para eliminar la oscilación, minimizar el sobre impulso, las oscilaciones amortiguadas y estabiliza el sistema. El amortiguamiento es proporcionado como una fuerza proporcional a la tasa de cambio de la posición de error y la constante de proporcionalidad es $K_D \times d_s$. Los coeficientes K_D y d_s con determinados en un proceso iterativo. El coeficiente K_D es sistemáticamente incrementado hasta que la flecha presente oscilaciones de alta frecuencia. El coeficiente d_s al presentarse las oscilaciones es incrementado en uno. El proceso entero es repetido hasta que d_s alcance un valor apropiado para el sistema.

El periodo de muestreo del sistema selecciona el intervalo de tiempo entre las actualizaciones de la posición de error. El intervalo de muestreo derivativo es un múltiplo entero del periodo de muestreo del sistema. Selecciona el intervalo de tiempo entre muestras de posiciones de error sucesivas usadas en el término derivativo y, por lo tanto, directamente afecta el amortiguamiento del sistema. El intervalo de muestreo derivativo podría ser cinco o diez veces más pequeño que constante mecánica de tiempo, esto significa que muchos sistemas requerirán producto más largo de $K_D \times d_s$ que mantenga bajo las vibraciones del motor.

3.- Determinación del Coeficiente de ganancia Proporcional. Cargas inerciales causan los siguientes errores, error de posición asociado con un eje móvil. Perturbaciones externas y cargas de torque causan errores de desplazamiento, posición de error ahocicado con un eje estacionario. El término proporcional del filtro provee una restauradora proporcional a la posición de error e incrementa linealmente conforme incrementa el error. El coeficiente de ganancia proporcional K_p , es la constante de proporcionalidad.

El coeficiente K_p es determinado con un proceso iterativo. El valor de K_p es incrementado y el amortiguamiento del sistema es evaluado. Esto se repite hasta que el sistema esta críticamente amortiguado. El amortiguamiento del sistema es evaluado manualmente.

Calibrando manualmente la flecha se revela cada incremento de K_p , incrementando la rigidez de la flecha.

3.2.6 Fuente de alimentación.

Las corrientes y voltajes manejados en los motores, en suma tiene un total de máximo: 3A a 12V.

Haciendo un balance entre una fuente de PC y una armada, se obtuvo una diferencia de costos no muy significativa, aparte de que la fuente de PC no presenta un ruido considerable y dispone de varias salidas entre ellas 5V y 12V.

Debido a lo anterior la fuente de alimentación adquirida es de las utilizadas en gabinetes de PC, dicha fuente cubre los requerimientos de potencia para alimentar los motores, la electrónica de potencia y la electrónica digital; gracias a que posee voltajes de 5V y 12 V que son los únicos empleados.

Pruebas de la fuente de alimentación:

Las pruebas realizadas a la fuente de alimentación simplemente consistieron en conectar los tres motores simultáneamente a la fuente de 12 V aplicando una carga a cada uno de ellos, dicha carga fue similar a la que tienen que soportar una vez que ya se haya ensamblado el manipulador, y al mismo tiempo se alimentaron los microcontroladores con una tensión eléctrica de 5V. A pesar de que se trata de una aproximación, al ver que ni las caídas de voltaje ni el ruido eran factores significativos, se decidió elegir de ésta fuente.

3.3 Desarrollo de la interfaz gráfica (Programación)

3.3.1 Presentación de BMAD

Brazo Manipulador para Aplicaciones Didácticas (BMAD), es una aplicación creada junto con el prototipo del Trabajo Terminal: Brazo Manipulador para Aplicaciones Didácticas. Este programa permite el manejo y control de dicho prototipo, mediante la implementación de la cinemática directa, cinemática inversa y el jacobiano del manipulador. Con esto es posible controlar su movimiento y generar ciertas trayectorias en su espacio de trabajo. Cierta parte importante del código se muestra en el apéndice G, lo que se refiere a la comunicación con el puerto, manejo de archivos, etc.

BMAD permite el manejo sencillo del manipulador en un ambiente agradable y de apoyo didáctico en el área de la robótica.

La interfaz que se muestra a continuación (figura 3.3.1) contiene una serie de botones y controles que realizan las operaciones para el control adecuado del manipulador, estos controles se describen más detalladamente en las siguientes secciones.

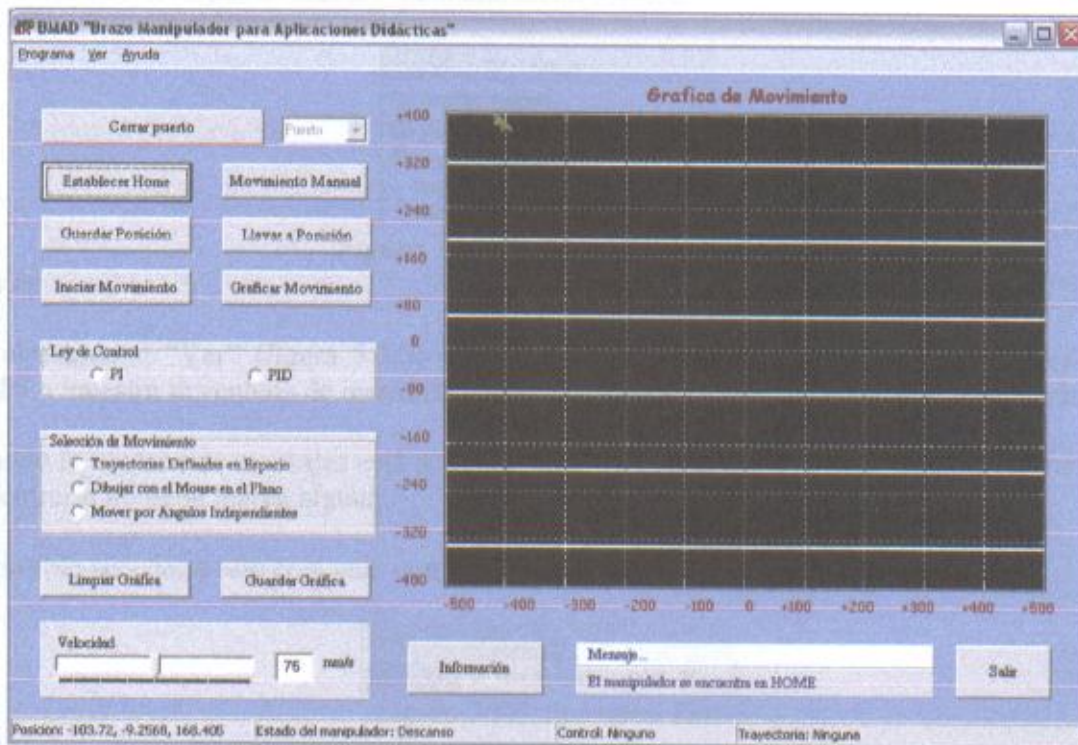


Figura 3.3.1 Presentación del panel principal.

3.3.2 Menú

La aplicación cuenta con una barra principal del menú, el cual contiene tres submenús:

a) Programa

En el submenú "Programa" se encuentran las funciones "Guardar / Llevar a Posición" que permiten guardar y recuperar una posición de un archivo que hace las veces de registro con 100 localidades. Con lo cual es posible llevar el manipulador a cierta posición el número de veces que uno desee.

También es posible controlar el manipulador manualmente con la función "Movimiento Manual" con lo que se abre una ventana en la que es posible rotar cada articulación independientemente.

El ítem "Opciones" permite modificar la apariencia de la interfaz, desde el tipo de letra de botones y etiquetas, colores de fondo y gráfica.

El comando "Salir" solicita al usuario que confirme la operación de lo contrario se continuará con ejecución del programa.

En la figura 3.3.2 se observa una imagen del submenú "Programa", cabe destacar que en ciertas funciones se puede acceder mediante teclas rápidas.

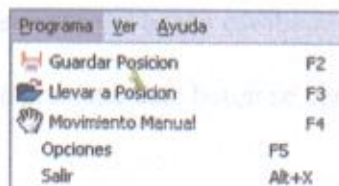


Figura 3.3.2 Submenú Programa.

b) Ver

En el submenú "Ver" (figura 3.3.3) se encuentra el comando "Ventana de mensajes" que oculta o muestra la ventana de mensajes que se encuentra en el panel principal del programa.

Cuando la ventana de mensajes está activa se observa que la función se encuentra marcada, de lo contrario no tiene marca alguna.

Es posible acceder a este comando con la tecla rápida F6.

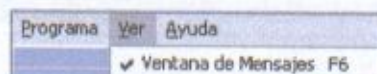


Figura 3.3.3. Submenú Ver.

c) Ayuda

En el submenú "Ayuda" (figura 3.3.4) se encuentran principalmente varios libros de ayuda de los diferentes temas involucrados en el desarrollo de este prototipo, los temas abarcan desde los fundamentos de robótica, control de motores, tutoriales de los programas usados para desarrollar el manipulador, ayuda de BMAD y un cuadro de diálogo final referente a este programa.

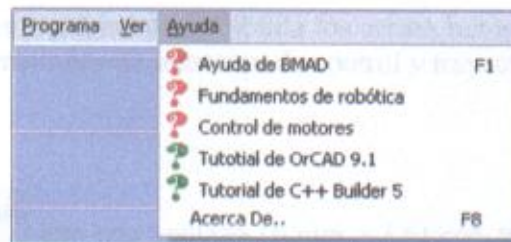


Figura 3.3.4 Submenú Ayuda.

3.3.3 Botones principales

a) Comunicación con el puerto

El botón **Comunicación con el puerto** establece la comunicación con el puerto serial establecido en el control de selección con la etiqueta **Puerto**.

Si el puerto en cuestión está disponible y se logra una comunicación exitosa, una función implementada habilitará los demás controles de la interfaz para llevar un seguimiento adecuado del programa; además el texto del botón cambiará a **Cerrar puerto**.

Por lo tanto la próxima vez que se presione este botón se cerrará el puerto con seguridad.

b) Establecer HOME

Antes de comenzar, el manipulador debe establecer una posición inicial que sirve de referencia para realizar sus movimientos posteriores, además que se evitarán colisiones entre sus eslabones.

El botón **Establecer Home** envía los comandos necesarios al microcontrolador para que lleve a cabo una rutina para determinar la posición HOME, un indicador muestra el progreso de la operación (figura 3.3.5); una vez concluida esta operación el microcontrolador le indica al programa la finalización de la tarea.

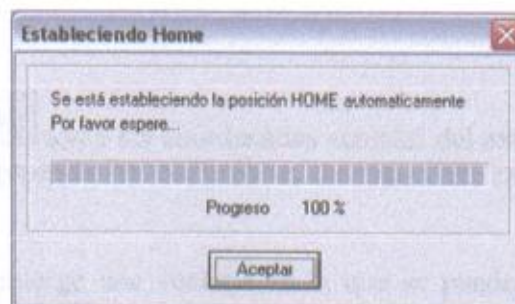


Figura 3.3.5 Indicador del progreso de la operación para establecer HOME.

Posteriormente una función implementada habilita los demás botones y controles de la interfaz para continuar con las operaciones de selección de control y trayectoria.

c) Mover Manualmente

El botón **Movimiento Manual** abre una ventana (figura 3.3.6) con la cual se puede mover cada articulación del manipulador independientemente, en los botones existentes en esta ventana se indica el nombre de la articulación y mediante un sistema de ejes se muestra como es su movimiento físicamente.



Figura 3.3.6 Ventana para mover manualmente el manipulador.

d) Guardar Posición

El botón **Guardar Posición** toma las coordenadas actuales del extremo final del manipulador y las guarda como una posición en un archivo de registros. Este archivo cuenta con 100 localidades disponibles.

Al presionar este botón emerge una ventana en la que se pueden seleccionar un registro y observar su contenido actual (figura 3.3.7); si el usuario selecciona un registro ocupado, la posición actual reemplaza a la anterior.

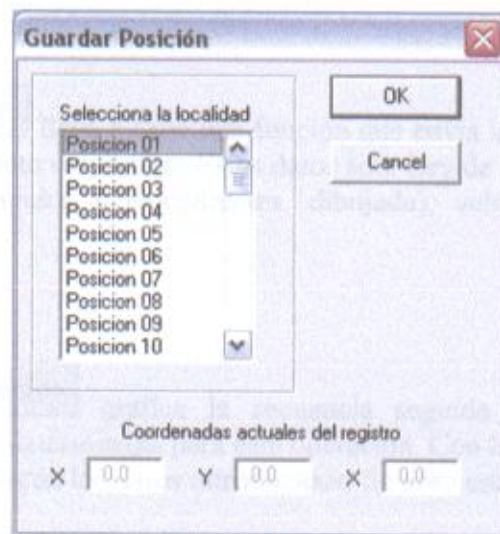


Figura 3.3.7 Ventana para guardar las coordenadas actuales.

e) Llevar a Posición

El botón **Llevar a Posición** abre una ventana en la que se pueden seleccionar una localidad del registro de posiciones y llevar el manipulador a las coordenadas previamente guardadas en dicha localidad.

La ventana emergente (figura 3.3.8) muestra el número de la localidad seleccionada y las coordenadas del extremo final del manipulador previamente guardadas..

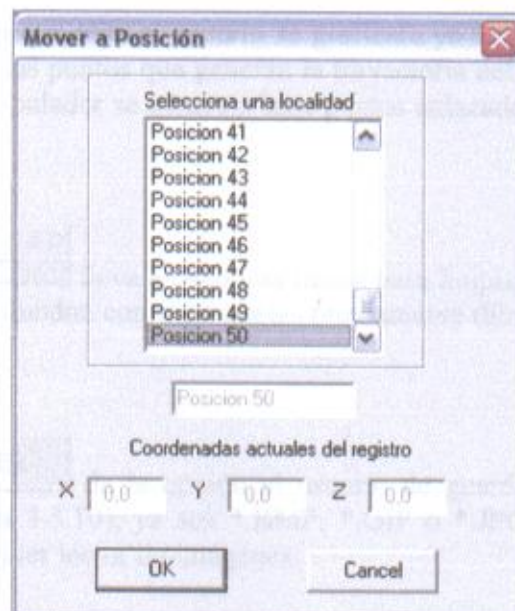


Figura 3.3.8 Ventana para mover el manipulador a una posición guardada.

f) Iniciar Movimiento

El botón **Iniciar Movimiento** lleva a cabo una función que envía los datos al microcontrolador para que inicie el movimiento respectivo. Estos datos son: Ley de control (constantes PI, PID), Trayectoria (Definida, ángulos independientes, dibujada), velocidad y posición de cada articulación.

g) Graficar Movimiento

Este botón **Graficar Movimiento** grafica la secuencia seguida por el extremo final del manipulador en la zona predeterminada para esta operación. Con la intención de que el usuario observe el comportamiento con los datos introducidos. Se abre una ventana donde se elige una opción (figura 3.3.9).

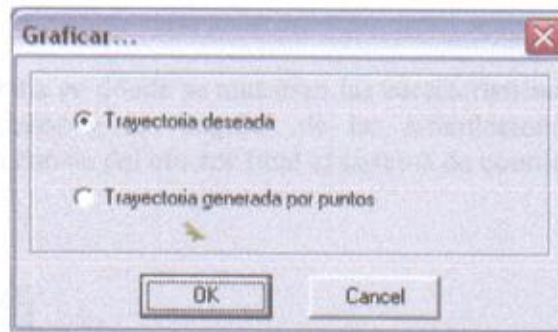


Figura 3.3.9 Selección de la opción a graficar.

Dependiendo de la opción que elija el usuario se graficará ya sea el movimiento deseado del manipulador o solamente los puntos que generan la trayectoria del movimiento. Cabe destacar que la trayectoria del manipulador se genera a base puntos enlazados por líneas rectas.

h) Limpiar Gráfica

Este botón **Limpiar Gráfica** lleva a cabo una rutina para limpiar la zona de gráficas, con la intención de que no se confundan con trayectorias previamente dibujadas.

i) Guardar Gráfica

Este botón **Guardar Gráfica** da la opción al usuario de guardar la gráfica actual en un archivo de imagen (figura 3.3.10), ya sea *.BMP, *.GIF ó *.JPG. Con la intención de que pueda ser abierto en cualquier lector de imágenes.

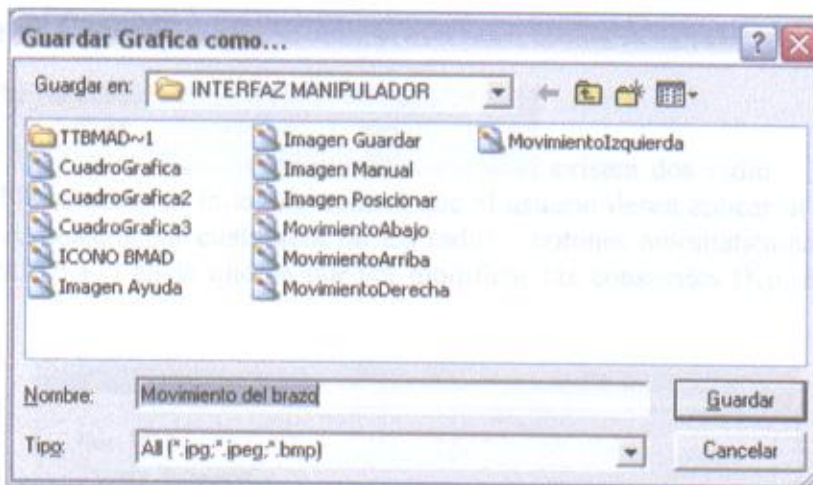



Figura 3.3.10 Cuadro de dialogo para guardar la gráfica actual.

j) Información

Este botón abre una ventana en donde se muestran las características del manipulador, es decir sus coordenadas cartesianas y los ángulos de las articulaciones, velocidad del último movimiento realizado, distancia del efector final al sistema de coordenadas origen, etc.

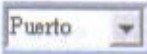
k) Salir

Salir

El botón  abre un cuadro de diálogo para confirmar la operación, de lo contrario se regresa al programa.

3.3.4 Otros controles

a) Elemento de selección "Puerto"

La casilla de selección  establece cual puerto serie de comunicaciones se usará para enviar y recibir los datos entre la computadora y el microcontrolador.

Existe la opción de seleccionar los puertos seriales "COM1", "COM2", "COM3" ó "COM4", esto con la finalidad de que el programa se adapte a la configuración de cada computadora en la que sea instalado.

De no seleccionarse algún puerto antes de establecer comunicación con el puerto se da por entendido que el usuario desea establecer comunicación con el puerto serie "COM1".

b) Panel “Ley de Control”



En el panel existen dos radio – botones con los cuales es posible seleccionar la ley de control que el usuario desea aplicar al movimiento del manipulador. Al seleccionar cualquiera de los radio – botones automáticamente se abre una ventana (figura 3.3.11) en la que se pueden modificar las constantes (K_p , K_i , K_d) de cada motor.

Motor	Termino Proporcional	K_p	Termino Integral	K_i	Termino Derivativo	K_d
Motor 1 (Cintura)	25	25	35	50		
Motor 2 (Hombro)	15	15	25	40		
Motor 3 (Codo)	10	10	20	30		

Figura 3.3.11 Ventana para establecer las constantes del control PID.

c) Panel “Selección de Movimiento”

En el panel para la selección de trayectoria (figura 3.3.12) existen tres radio – botones con los cuales es posible seleccionar ya sea una trayectoria previamente definida en el espacio, una trayectoria dibujada en un plano o un movimiento por ángulos independientes.

Con cada uno de las opciones anteriores emerge una ventana referente a la elección tomada en donde se introducirán los valores adecuados para generar una trayectoria aceptable.

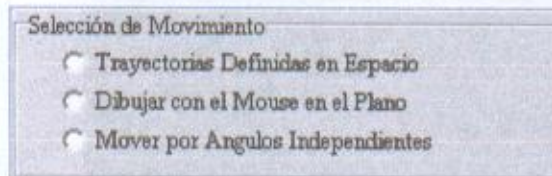


Figura 3.3.12 Panel para elegir una trayectoria.

En el caso de las trayectorias definidas se deben introducir los valores de las coordenadas y longitudes solicitadas por la ventana emergente (figura 3.3.13) y se selecciona el plano sobre el cual se desea realizar el movimiento.

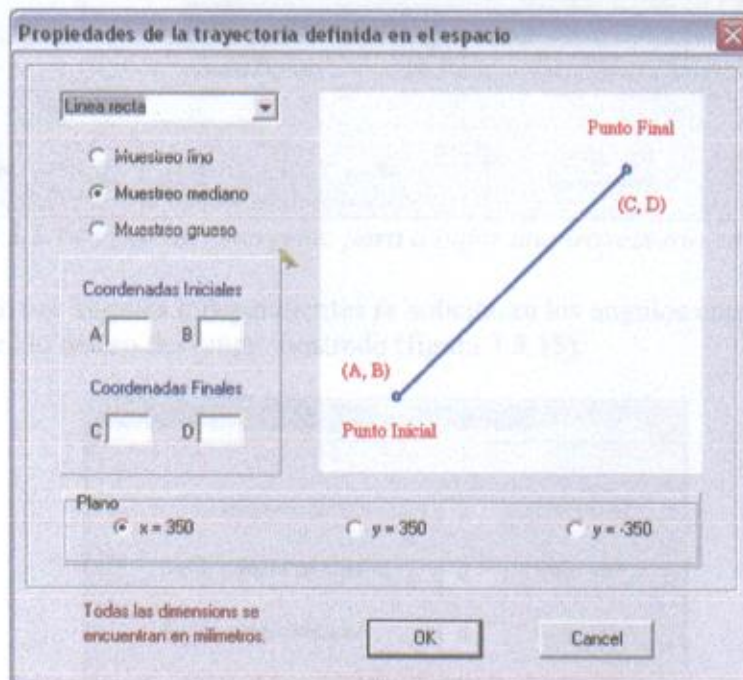


Figura 3.3.13 Ventana emergente para la selección de una trayectoria definida.

Para realizar una trayectoria en el plano basta con mover el puntero del mouse sobre la zona de gráficos mientras se mantiene presionado un botón del mouse. Unas casillas de edición muestran las coordenadas y el número de puntos de la trayectoria (figura 3.3.14), también se pueden seleccionar el plano de trabajo del manipulador.

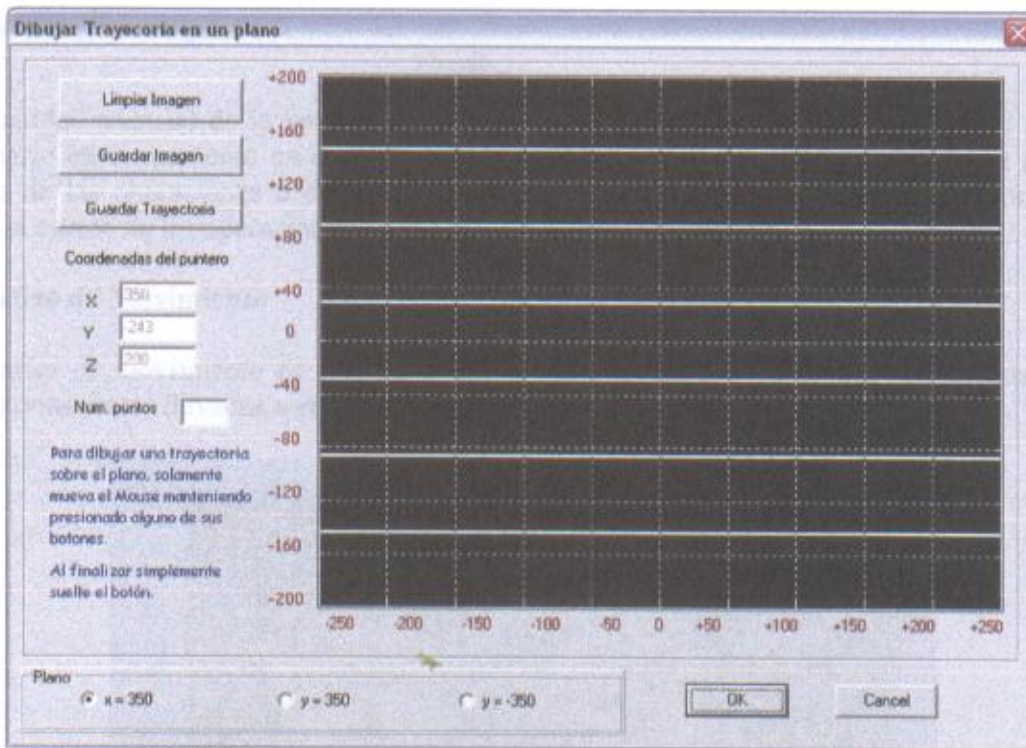


Figura 3.3.14 Ventana emergente para dibujar una trayectoria en el plano.

En el movimiento por ángulos independientes se solicitarán los ángulos correspondientes al movimiento deseado dentro del rango mostrado (figura 3.3.15).

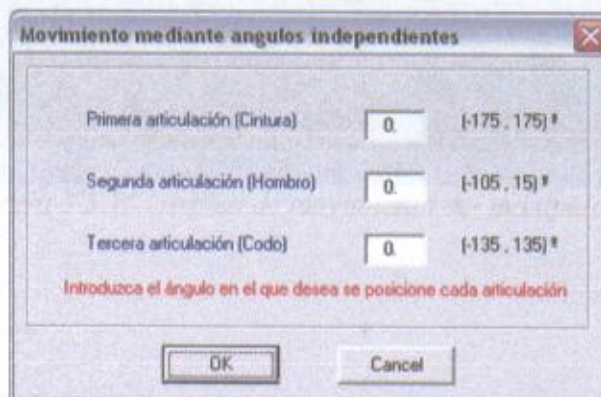
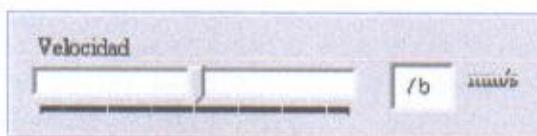


Figura 3.3.15 Ventana emergente para establecer los ángulos de los eslabones.

d) Panel "Velocidad"



El panel permite variar la velocidad lineal del extremo final del manipulador, ya sea mediante una barra o una casilla de edición.

e) Ventana “Mensaje...”

Mensaje...

El manipulador se encuentra en HOME

La finalidad esencial de la ventana es auxiliar al usuario en el momento en que se esté manejando el programa, ya que esta barra informa acerca de las operaciones o datos que se requieren y en ciertos momentos da información sobre el estado de las operaciones que se estén o se hayan realizado.

f) Gráfica de Movimiento

La gráfica de movimiento es la zona donde se dibujan tanto las trayectorias deseadas y las trayectorias reales llevadas a cabo por el manipulador (figura 3.3.16).

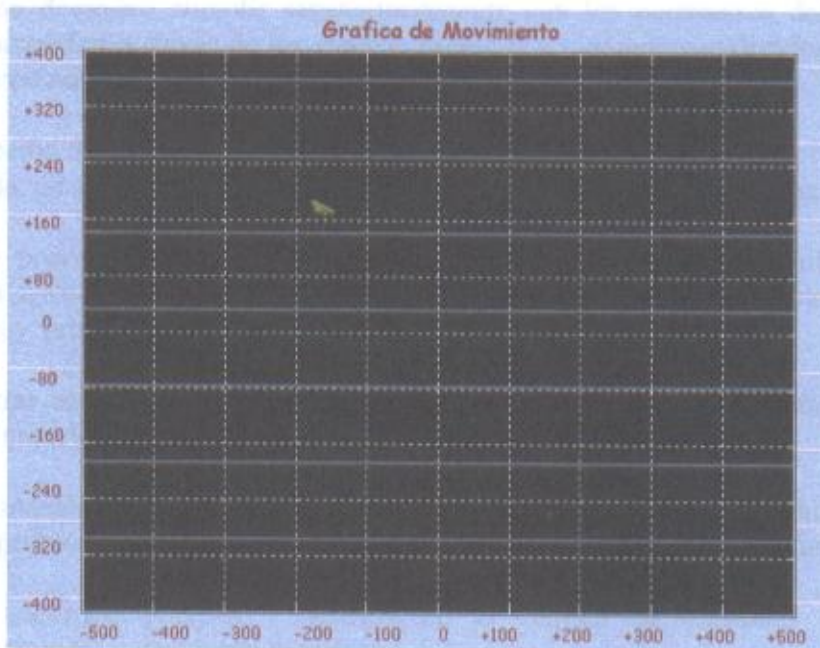


Figura 3.3.16 Gráfica de movimiento del manipulador.

CAPÍTULO 4. CONCLUSIONES

Una vez realizado y presentado el Trabajo Terminal se ha observado la importancia de realizar un estudio y una investigación de calidad, que además este bien sustentada en conocimientos teóricos y prácticos, ya que esto ha significado el llevar con éxito y a buen término la construcción del prototipo.

Los puntos más relevantes son:

- Se llevó a cabo la construcción de un brazo manipulador para aplicaciones didácticas de tres grados de libertad de acuerdo con los objetivos propuestos.
- El diseño demostró cumplir satisfactoriamente con los parámetros definidos, esto se pudo comprobar a través de los análisis estáticos y dinámicos realizados a la estructura del manipulador.
- La selección de los materiales usados en la construcción del manipulador demostró ser la adecuada, de acuerdo con el análisis estructural y el comportamiento final del prototipo.
- En este proyecto el uso de del lenguaje de programación C++ Builder 5 presentó ventajas frente a Visual Basic 6, específicamente en lo que respecta al manejo de archivos y velocidad de procesamiento.
- La interfaz desarrollada en C++ Builder 5 presentó un ambiente de trabajo agradable y de fácil manejo para el usuario final del Brazo Manipulador.
- La transmisión empleada es adecuada para la carga que se desea mover ya que no existen problemas en cuanto al par necesario para mover los eslabones del manipulador.
- Se ha adquirido práctica y destreza en el empleo de máquinas y herramientas (convencionales y de control numérico).
- Se ha adquirido un mejor dominio en el manejo de software de dibujo (Solid Works 2003, AutoCAD 2002 y Mechanical Desktop 6) que no forma parte del curriculum actual de la carrera.
- Considerando que el circuito integrado LM629 multiplica 4 veces la resolución del encoder usado, se tiene una resolución final de 4096 pulsos por revolución lo cual implica que se tiene un error máximo de la señal de realimentación de 0.087° , que representa una ventaja adicional al utilizar este circuito.
- Los motores seleccionados a partir de la carga que se desea mover y de los resultados obtenidos del análisis estático, resultaron correctos y responden adecuadamente a las necesidades del prototipo.

- El prototipo es aceptable en función de los parámetros definidos y esto se puede comprobar a través de las pruebas realizadas.

A continuación se describe el desempeño observado en el manipulador utilizando el control PID y el control PI.

Control PID. Cuando se hizo el ajuste de las constantes se llegó a la conclusión de que las variaciones que se realizaron no representaban un cambio significativo en el comportamiento del manipulador en cuanto al control de posición del efector final. Las constantes con las que el manipulador tuvo un buen desempeño fueron aquellas que se encontraban dentro de los siguientes rangos: $K_p = 200 - 240$, $K_I = 95 - 125$ y $K_D = 45 - 95$. Durante estas pruebas en los motores se presentaba poco ruido audible, esto debido a la presencia del término derivativo.

Control PI. Al igual que en el caso anterior, no se presentaron cambios significativos en el comportamiento del manipulador cuando se variaron las constantes, las constantes con las que el manipulador tuvo un buen desempeño se encontraban dentro de los siguientes rangos: $K_p = 200 - 240$ y $K_I = 95 - 125$. Durante estas pruebas también se presentaba ruido audible, aunque se esperaba que éste no existiera debido a la falta del término derivativo.

Además se observó que los controles PI y PID presentaban un error de posición y de velocidad de respuesta semejantes, consideramos que es debido al tipo de motores usados ya que tienen una transmisión mecánica interna de tornillo - corona lo cual limita el control que se aplicó; la velocidad máxima de dichos motores es de 35 rpm lo cual reduce mucho el rango de control por velocidad.

Haciendo un análisis comparativo entre el control PID y el control PI, se concluye que ambas leyes resultaron adecuadas para el control de posición y de velocidad del efector final, sin embargo en pruebas de seguimiento de trayectorias no se pudo obtener una buena respuesta debido a que el ajuste mecánico presentaba problemas que repercutían en vibraciones del manipulador, además de que en la parte de la comunicación PC - AVR en ocasiones se presentaban pérdidas de datos que impedían continuar con el siguiente punto de la trayectoria, ya que se enviaba un mensaje de error y se detenía la transmisión de datos.

MEJORAS Y TRABAJO FUTURO

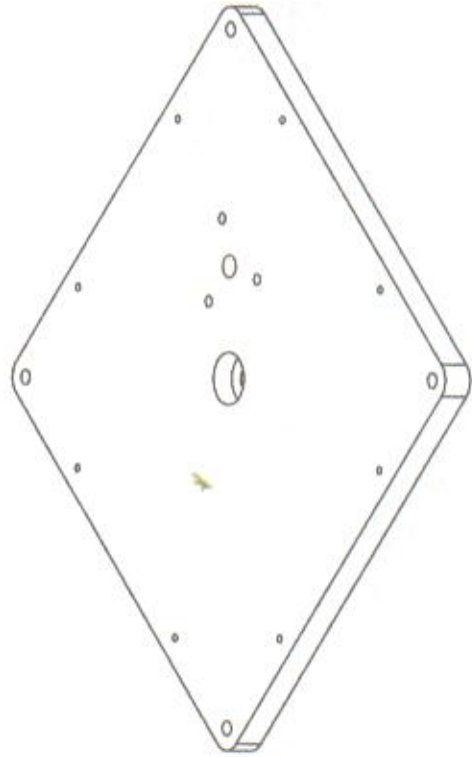
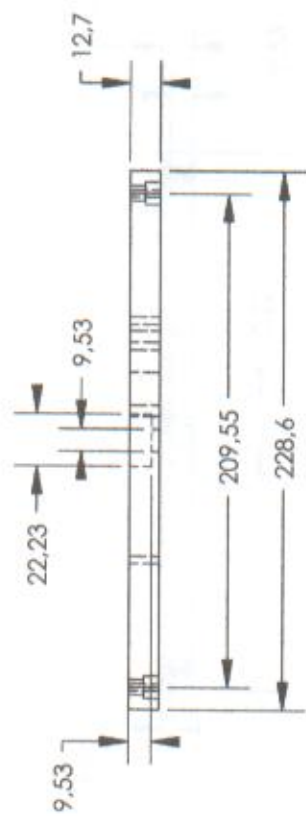
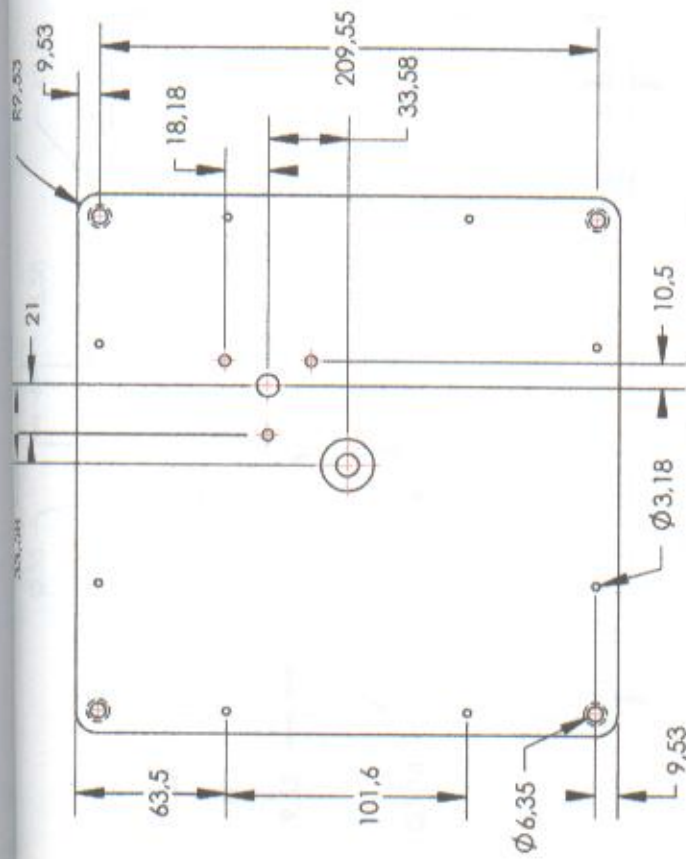
- Continuar la investigación sobre manipuladores con el fin de desarrollar diseños más estéticos y funcionales para la realización de diseños posteriores.
- Se ha considerado agregar unas pinzas de dos grados de libertad en vez del efector final que se presenta en este momento, con la finalidad de que el manipulador pueda ser más versátil.
- Corregir errores en la comunicación PC - AVR.
- Mejorar diseño mecánico y construcción del manipulador para evitar problemas de vibraciones o desajustes.

APÉNDICE A: Planos estructurales del manipulador

A continuación se presentan algunos de los planos estructurales empleados para la construcción mecánica del manipulador, cabe mencionar que el resto de los diagramas se incluyen dentro del CD que se anexa con ésta tesis.



Figura I. Presentación final del manipulador

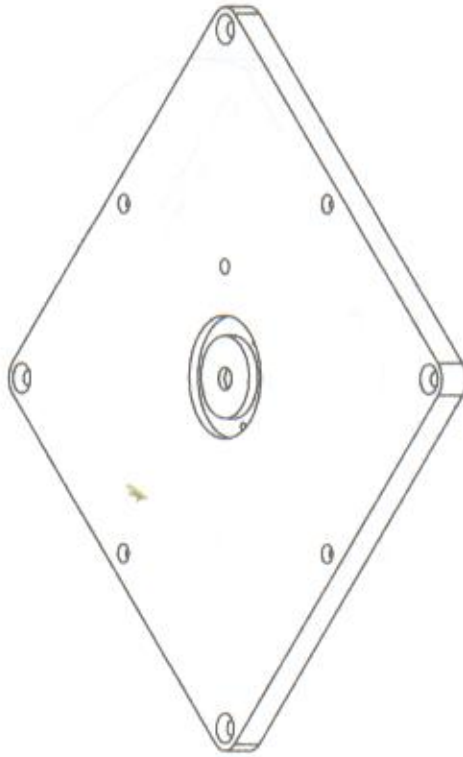
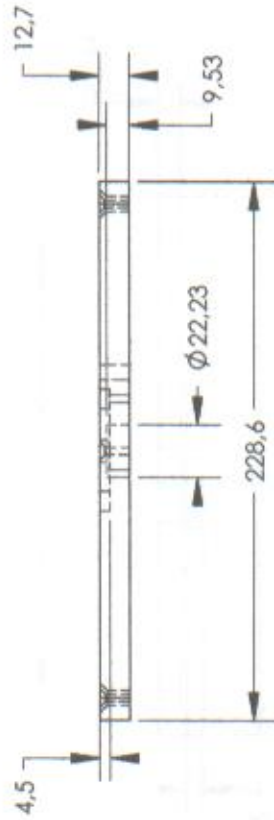
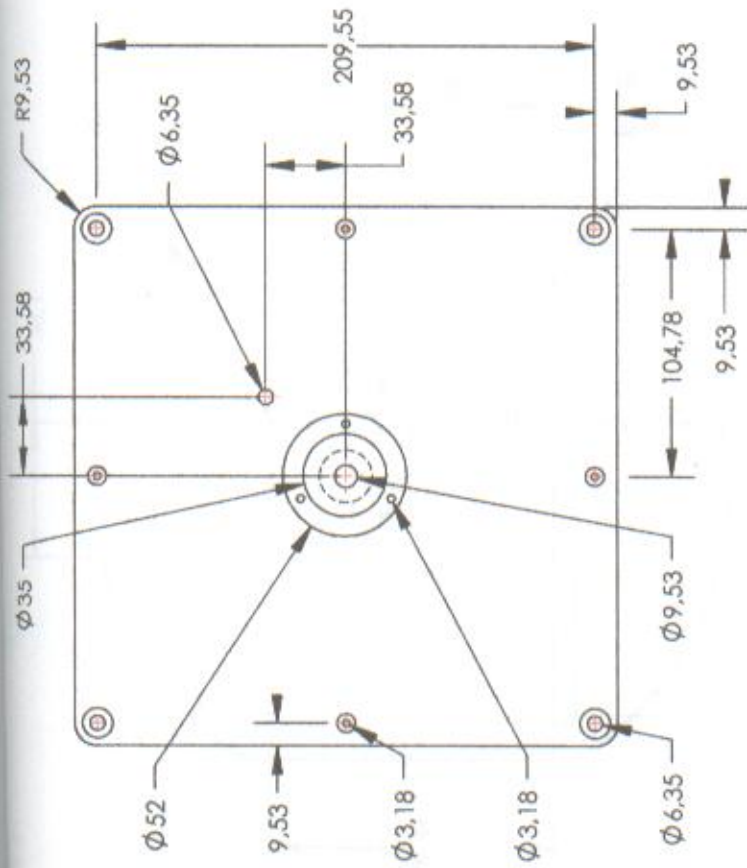


IPN



UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGIAS AVANZADAS

Diseñado por: Martinez Rodriguez Noé	Título: Base fija
Aprobado por: Fernández Nava Cecilia	Dibujo No. Rev
Comentarios:	Escala: 1:3 Acot: Milímetros



UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGIAS AVANZADAS

Diseñado por:
Martinez Rodriguez Noé
Aprobado por:
Fernández Nava Cecilia
Comentarios:

Título:

Base fija 2

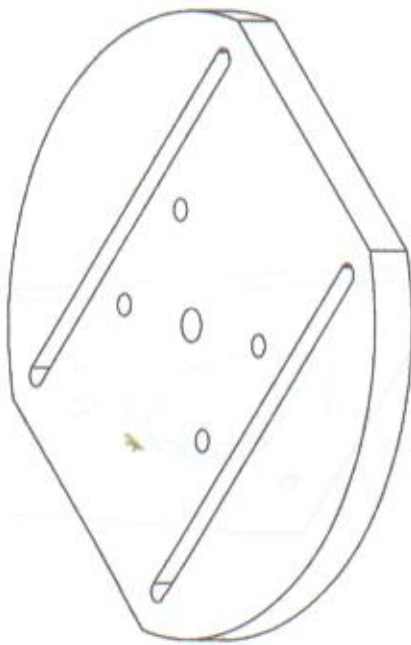
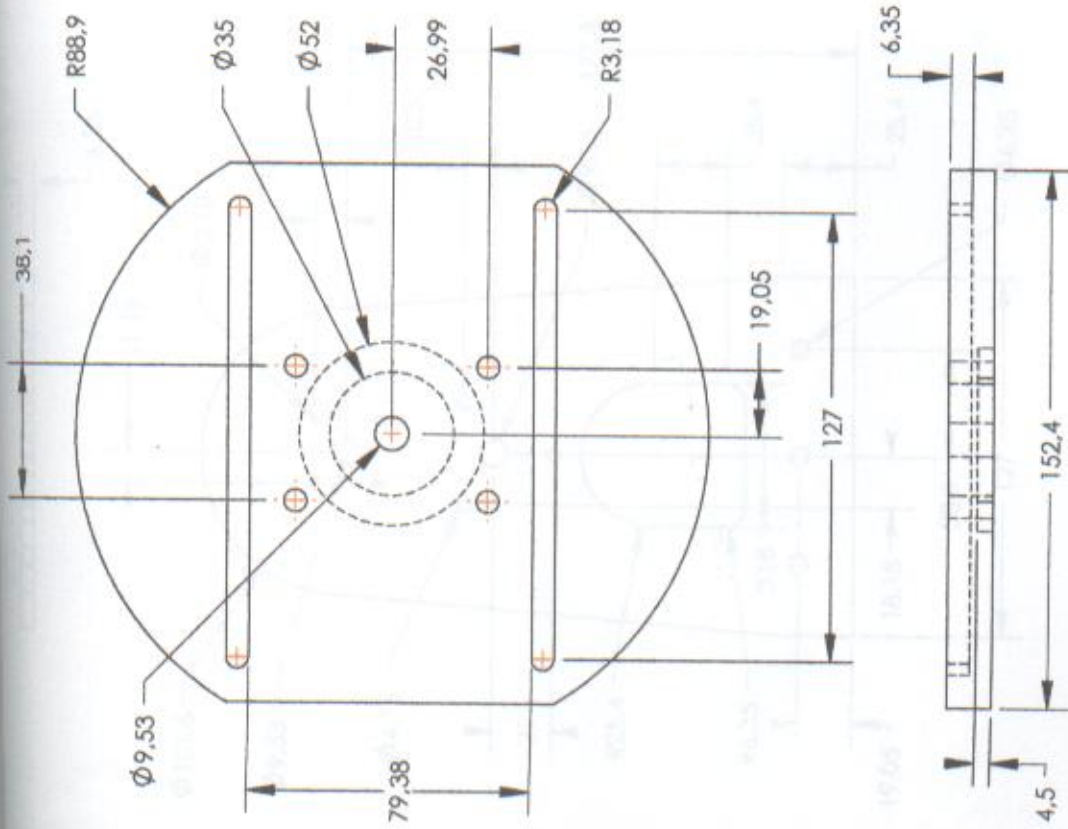
Dibujo No. Rev

Escala: 1:3

Acot: Milímetros

IPN



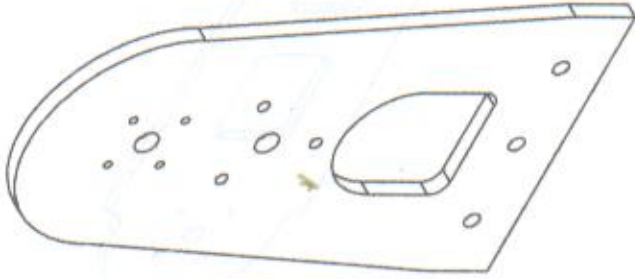
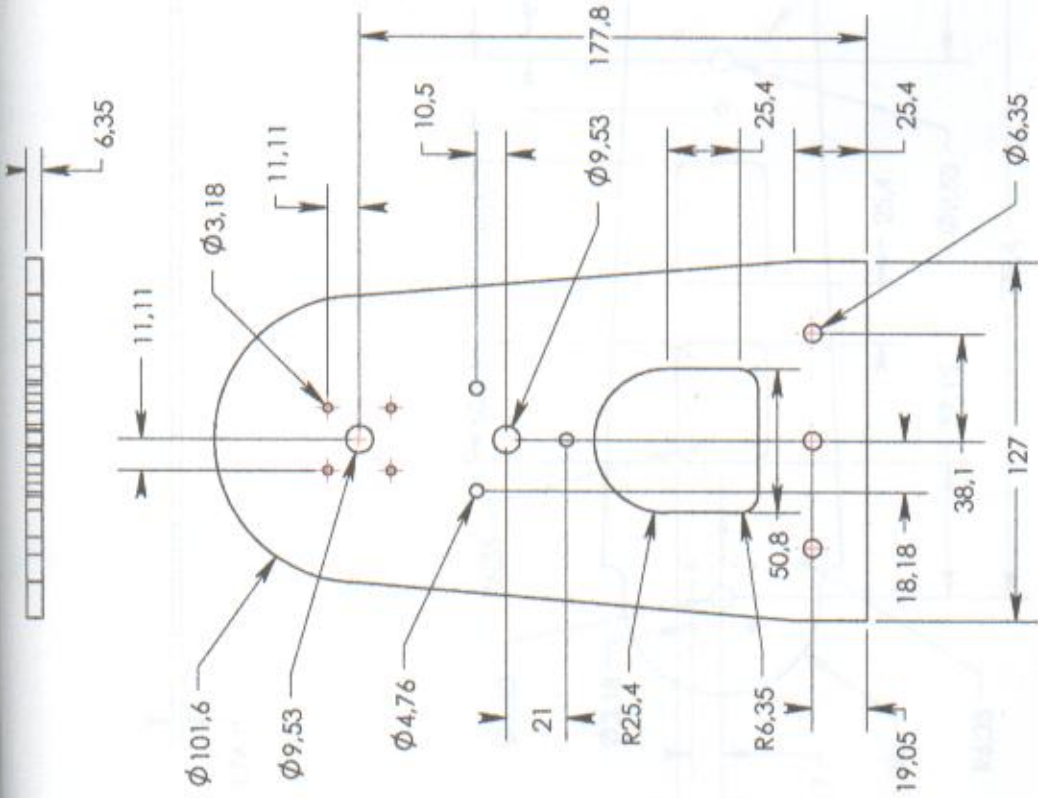


IPN



UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGIAS AVANZADAS

Diseñado por: Martínez Rodríguez Noé	Título: Base Giratoria
Aprobado por: Fernández Nava Cecilia	Dibujo No. _____ Rev _____
Comentarios:	Escala: 1:2 Acot: Milímetros



IPN



UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGIAS AVANZADAS

Diseñado por:
Pineda Pérez Juan Carlos

Aprobado por:
Fernández Nava Cecilia

Comentarios:

Título:

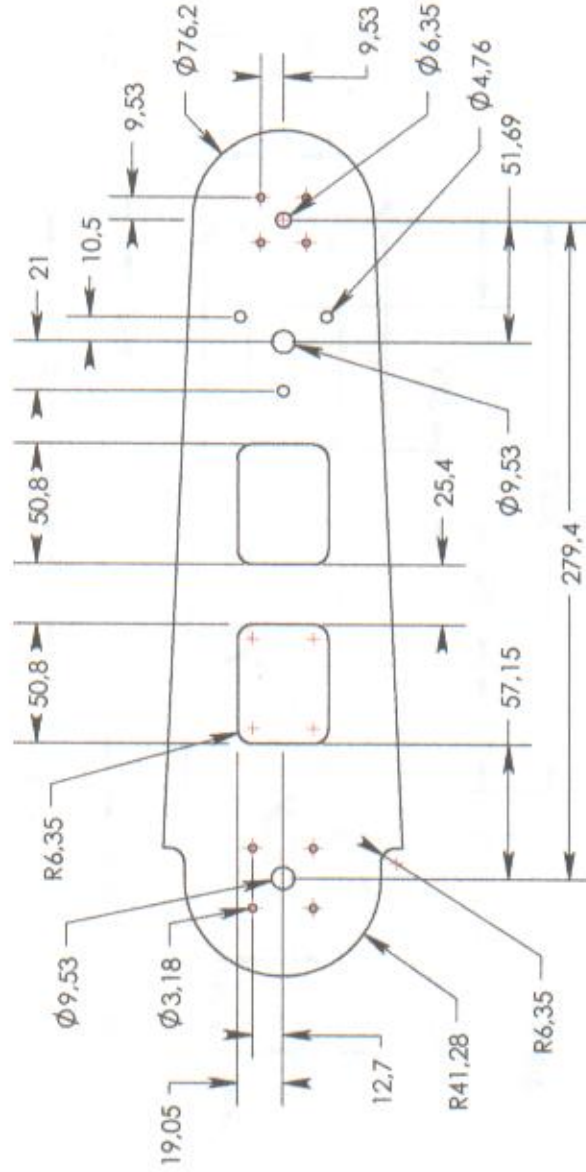
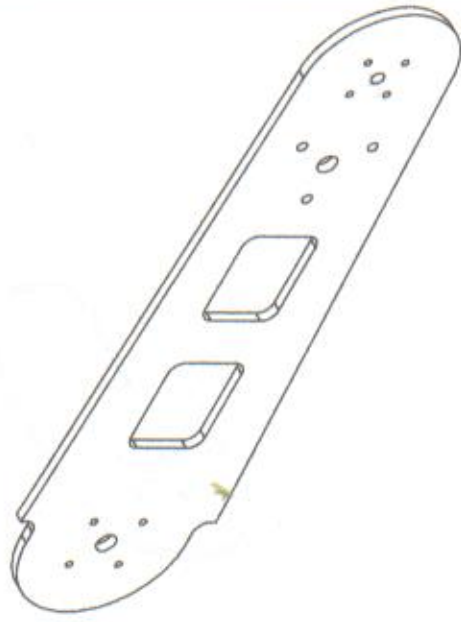
Estabón 1 derecho

Dibujo No.

Rev

Escala: 1 : 3

Acot: Milímetros

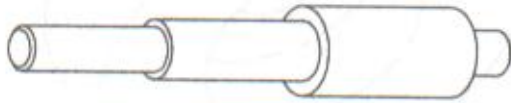
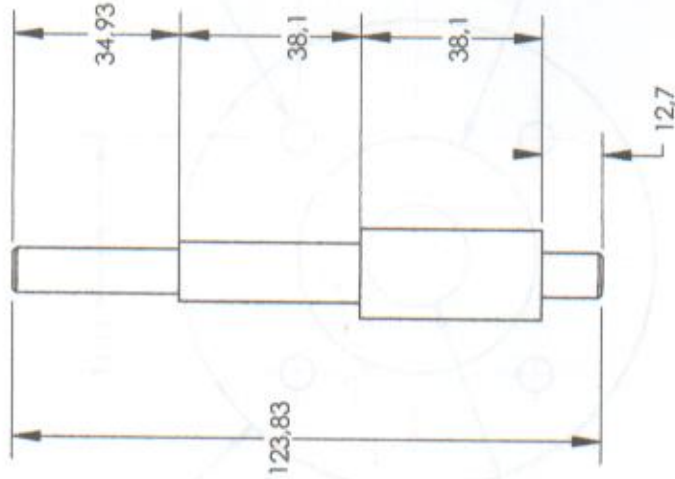


**UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGIAS AVANZADAS**

Diseñado por: Frieda Pérez Juan Carlos	Titulo: Eslabón 2 derecho	
Aprobado por: Fernández Nava Cecilia	Dibujo No.	Rev
Comentarios:	Escala: 1:3	Acot: Milímetros

IPN





IPN



UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGIAS AVANZADAS

Diseñado por:
Martínez Rodríguez Noé

Aprobado por:
Fernández Nava Cecilia

Comentarios:

Título:

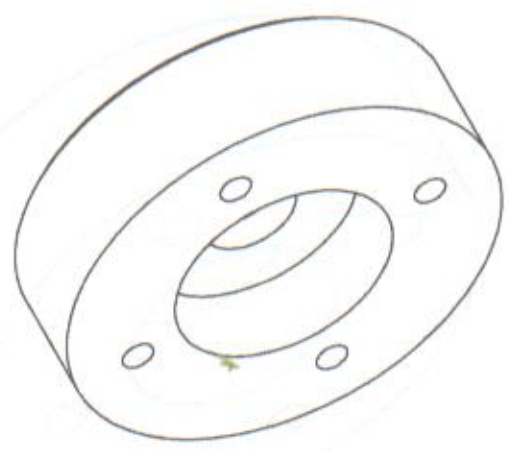
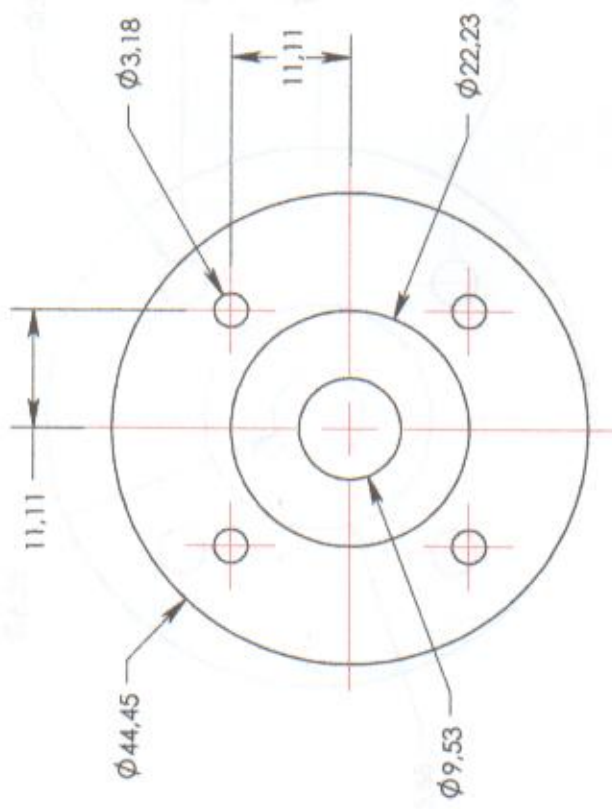
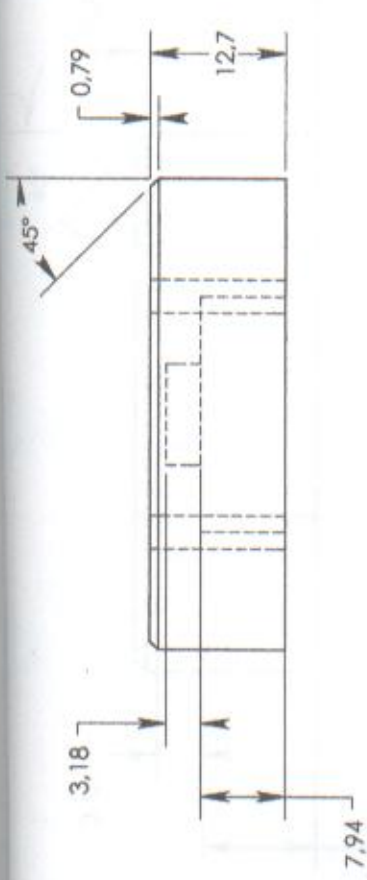
Eje base

Dibujo No.

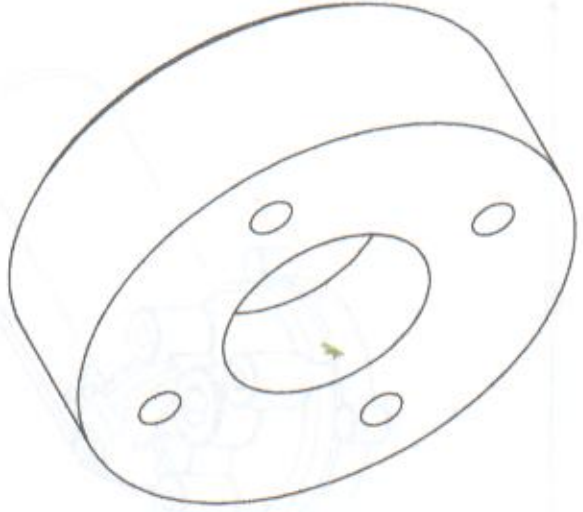
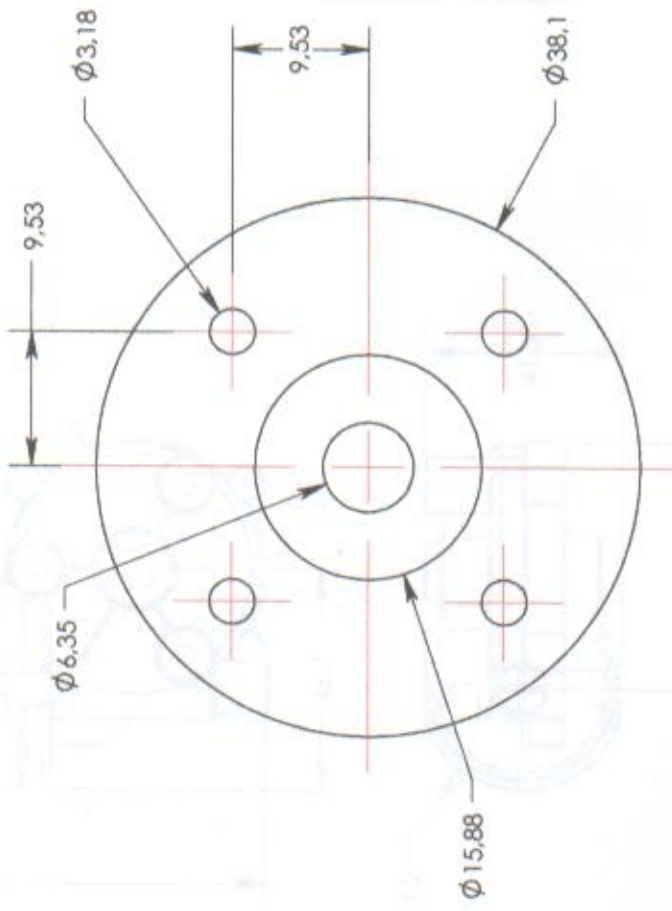
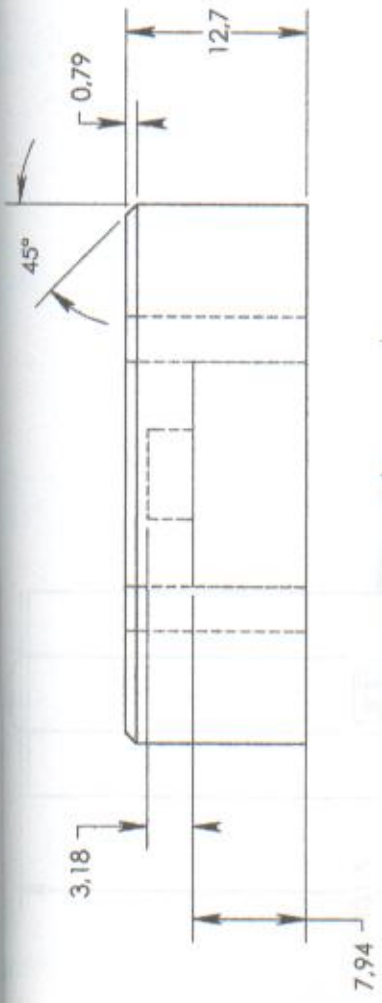
Rev

Escala: 1:2

Acot: Milímetros



IPN 	UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERIA Y TECNOLOGIAS AVANZADAS	
	Diseñado por: Murtillo Ramírez Oscar	Título: Tapa para balero eslabón 1
Aprobado por: Fernández Nava Cecilia	Dibujo No.	Rev
Comentarios:	Escala: 1.5 : 1	Acot: Milímetros

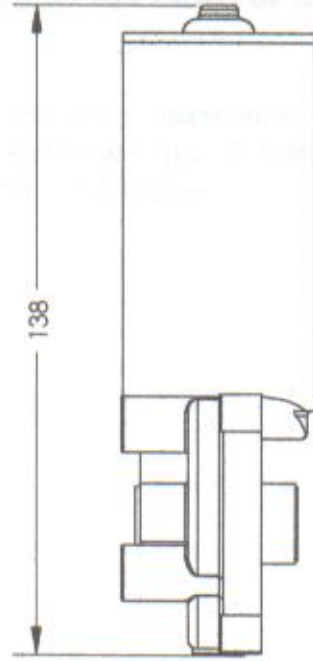
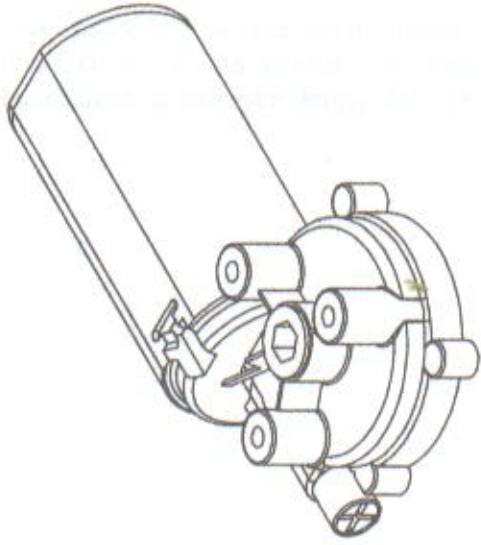
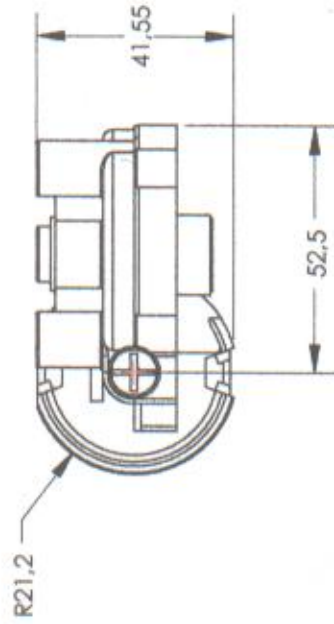
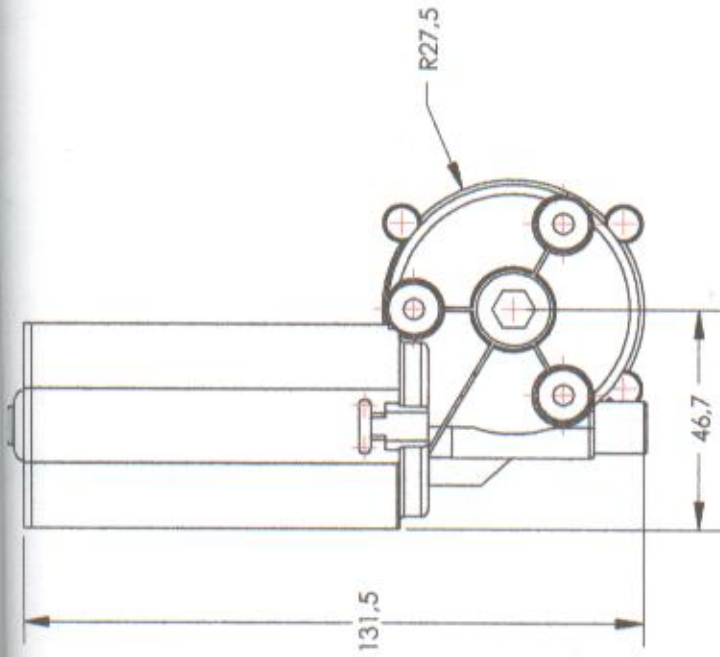


IPN



UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGIAS AVANZADAS

Diseñado por: Muello Ramírez Oscar	Título: Tapa para balero eslabón 2	
Aprobado por: Fermández Nava Cecilia	Dibujo No.	Rev
Comentaríot:	Escala: 2:1	Acot: Milímetros



IPN



UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGIAS AVANZADAS

Diseñado por:
Pineda Pérez Juan Carlos

Aprobado por:
Fernández Nava Cecilia

Comentarios:
Las cotas solo muestran las
principales dimensiones:
alto, largo y ancho.

Título:

Motor de 24 VCD

Dibujo No.

Rev

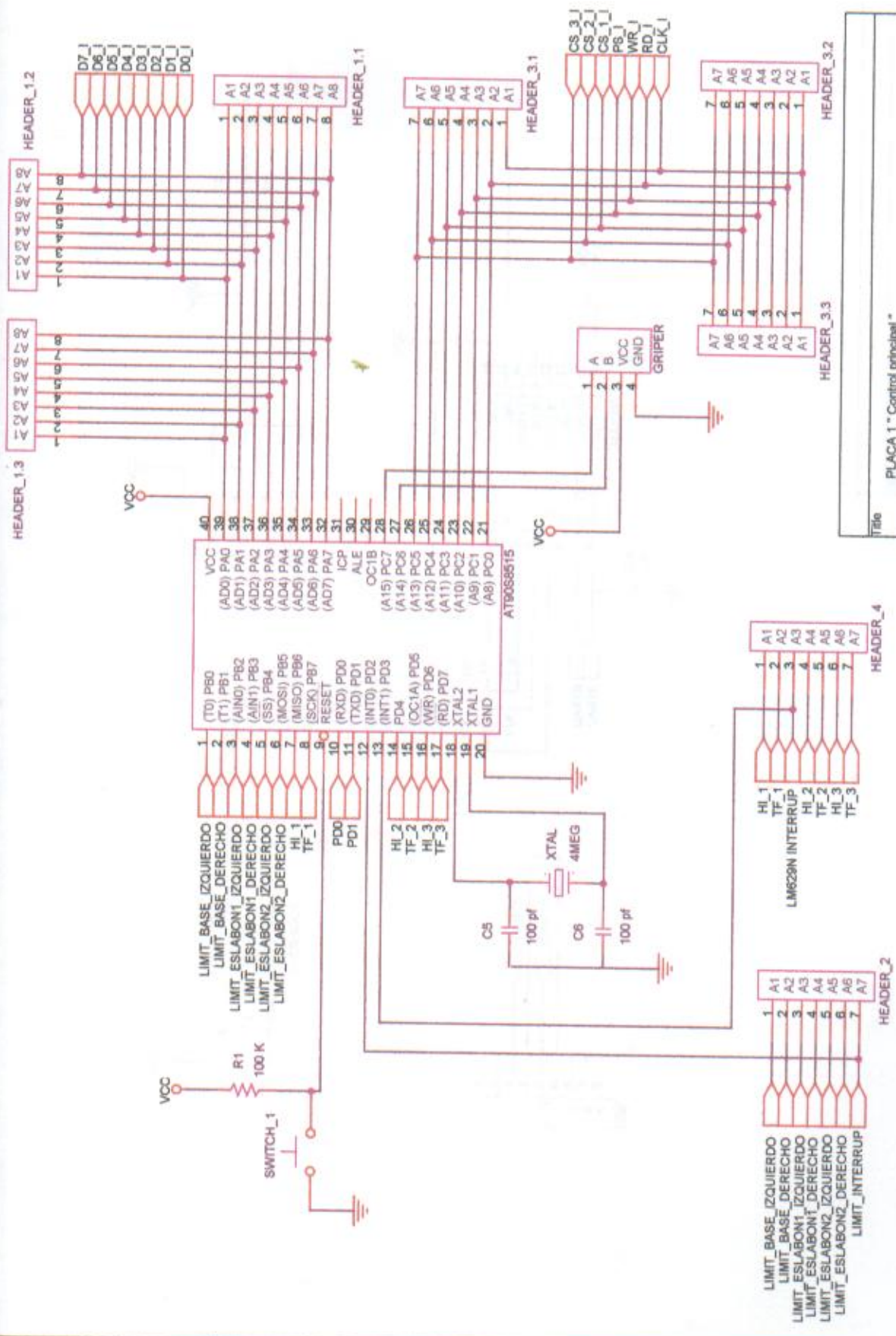
Escala:
1:2

Acot:

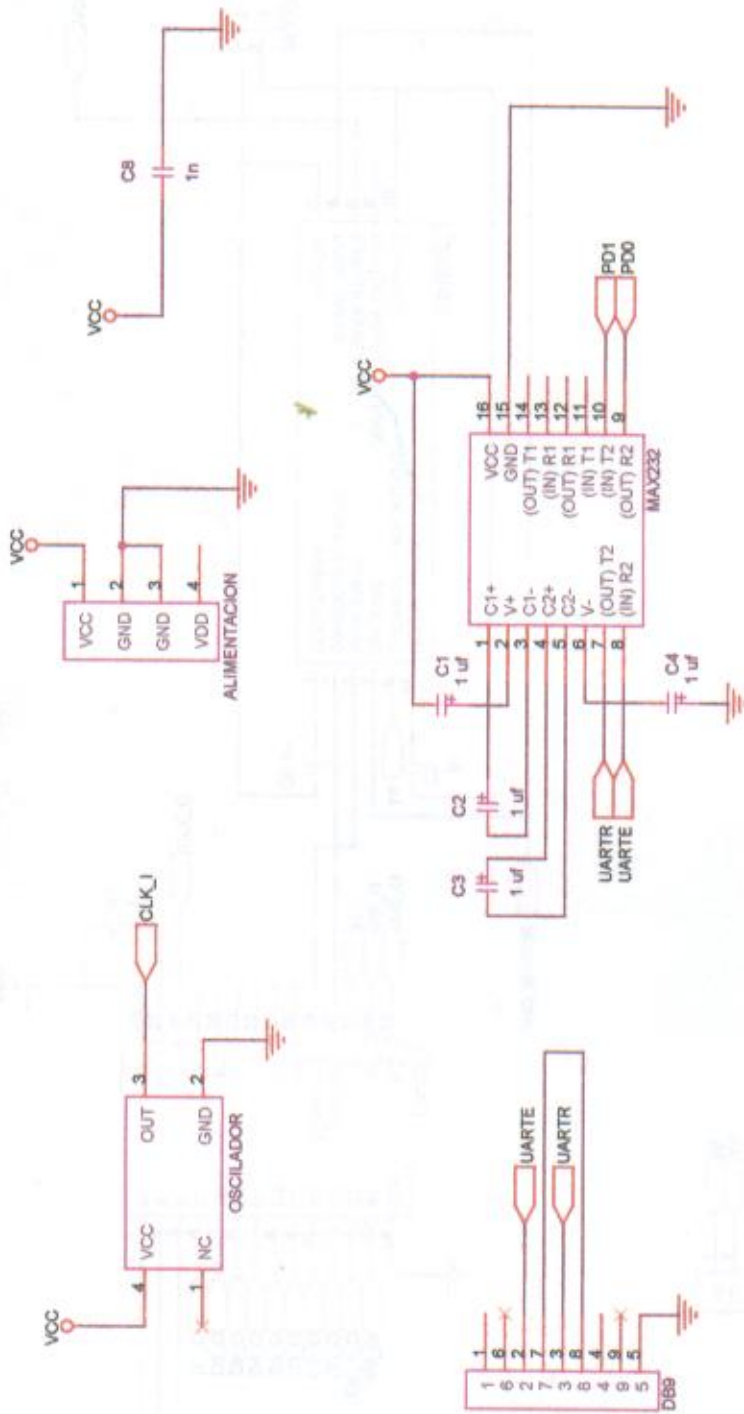
Milímetros

APÉNDICE B: Diagramas electrónicos del control del manipulador

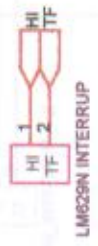
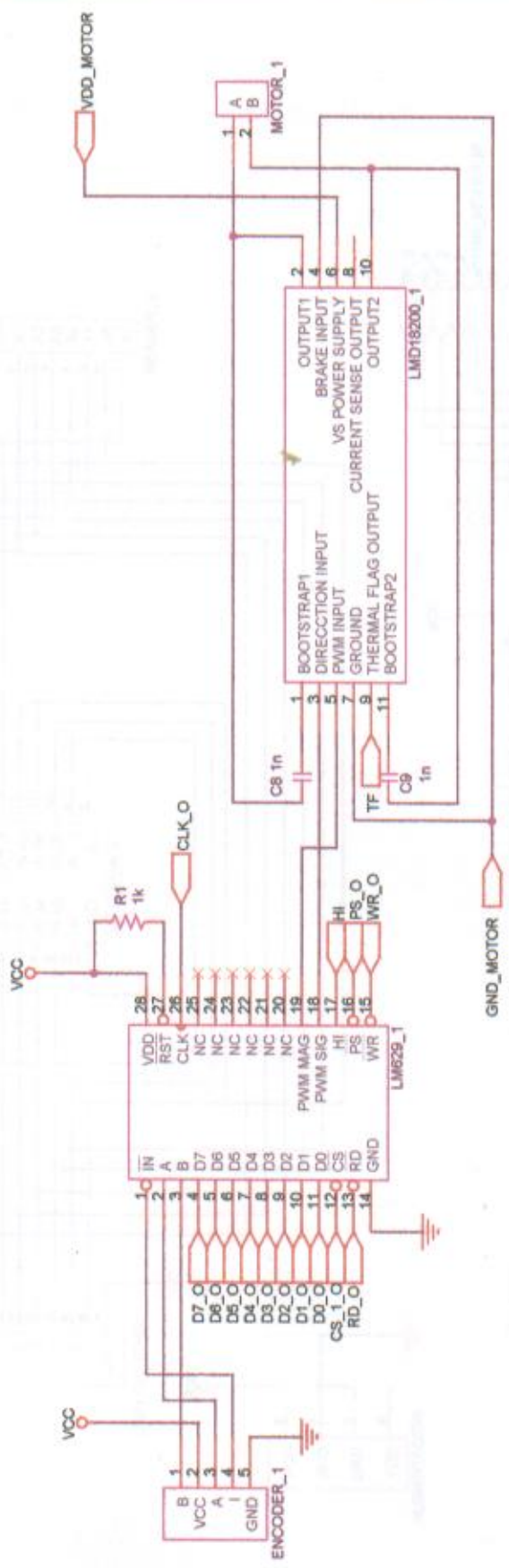
A continuación se presentan algunos de los diagramas electrónicos empleados para la construcción de las placas electrónicas, cabe mencionar que el resto de los diagramas electrónicos se incluyen dentro del CD que se anexa con ésta tesis.



Title		PLACA 1 " Control principal "	
Size	A	Document Number	1
Date:	Sunday, May 29, 2005	Sheet	1 of 1



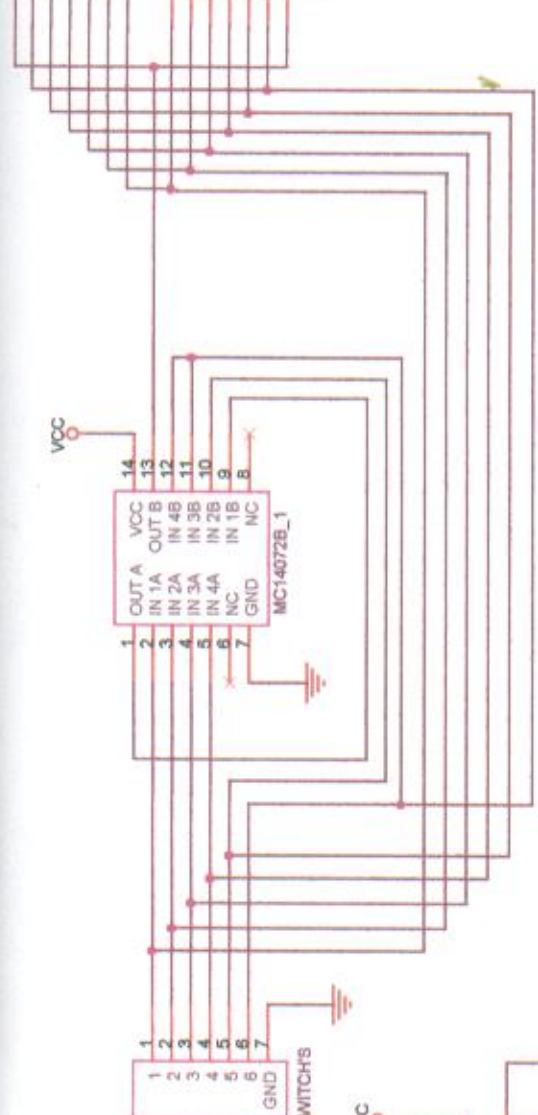
Title		PLACA 1 "MAX232 y alimentaciones"	
Size	Document Number	Rev	Rev
A	1	1	1
Date:	Sunday, May 29, 2005	Sheet	1 of 1



Title		PLACA 2 "LM629 y puente H LM18200"	
Size	Document Number		Rev
A	2		1
Date:	Sunday, May 29, 2005	Sheet	1 of 1

LIMIT_INTERRUPT
 LIMIT_ESLABON2_DERECHO
 LIMIT_ESLABON2_IZQUIERDO
 LIMIT_ESLABON1_DERECHO
 LIMIT_ESLABON1_IZQUIERDO
 LIMIT_BASE_DERECHO
 LIMIT_BASE_IZQUIERDO

1 A1
 2 A2
 3 A3
 4 A4
 5 A5
 6 A6
 7 A7
 HEADER_1



LIMIT SWITCHES

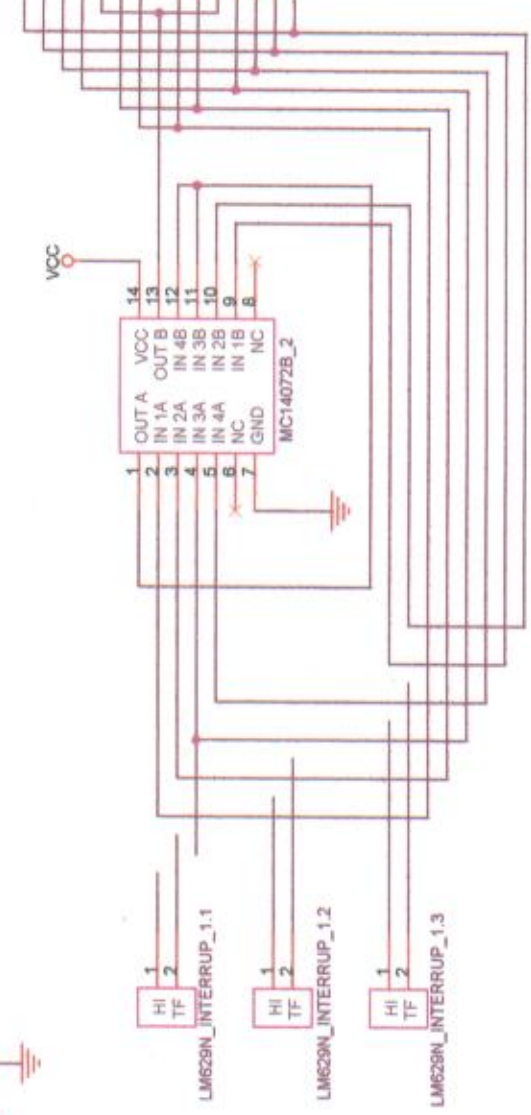
VCC



ALIMENTACION

TF_3
 HI_3
 TF_2
 HI_2
 LM629N_INTERRUPT
 TF_1
 HI_1

1 A1
 2 A2
 3 A3
 4 A4
 5 A5
 6 A6
 7 A7
 HEADER_2

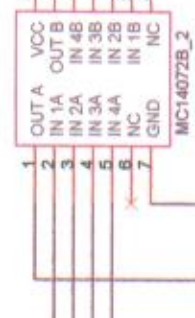


LM629N_INTERRUPT_1.1

LM629N_INTERRUPT_1.2

LM629N_INTERRUPT_1.3

VCC



Title: PLACA 6 "Control de Interrupciones"

Size: A
 Document Number: 6

Date: Sunday, May 29, 2005
 Sheet: 1 of 1

APÉNDICE C: Especificaciones de los circuitos integrados

A continuación se presenta parte de las especificaciones eléctricas de algunos de los circuitos electrónicos empleados, al igual que en el apéndice anterior el resto de las especificaciones se incluyen dentro del CD que se anexa con ésta tesis.



Figura 5. Circuito de control de velocidad



Figura 6. Circuito de control de velocidad, control de posición y control de fuerza

LM628/LM629 Precision Motion Controller

General Description

The LM628/LM629 are dedicated motion-control processors designed for use with a variety of DC and brushless DC servo motors, and other servomechanisms which provide a quadrature incremental position feedback signal. The parts perform the intensive, real-time computational tasks required for high performance digital motion control. The host control software interface is facilitated by a high-level command set. The LM628 has an 8-bit output which can drive either an 8-bit or a 12-bit DAC. The components required to build a servo system are reduced to the DC motor/actuator, an incremental encoder, a DAC, a power amplifier, and the LM628. An LM629-based system is similar, except that it provides an 8-bit PWM output for directly driving H-switches. The parts are fabricated in NMOS and packaged in a 28-pin dual in-line package or a 24-pin surface mount package (LM629 only). Both 6 MHz and 8 MHz maximum frequency versions are available with the suffixes -6 and -8, respectively, used to designate the versions. They incorporate an SDA core processor and cells designed by SDA.

Features

- 32-bit position, velocity, and acceleration registers
- Programmable digital PID filter with 16-bit coefficients
- Programmable derivative sampling interval
- 8- or 12-bit DAC output data (LM628)
- 8-bit sign-magnitude PWM output data (LM629)
- Internal trapezoidal velocity profile generator
- Velocity, target position, and filter parameters may be changed during motion
- Position and velocity modes of operation
- Real-time programmable host interrupts
- 8-bit parallel asynchronous host interface
- Quadrature incremental encoder interface with index pulse input
- Available in a 28-pin dual in-line package or a 24-pin surface mount package (LM629 only)

TRI-STATE™ is a registered trademark of National Semiconductor Corporation.

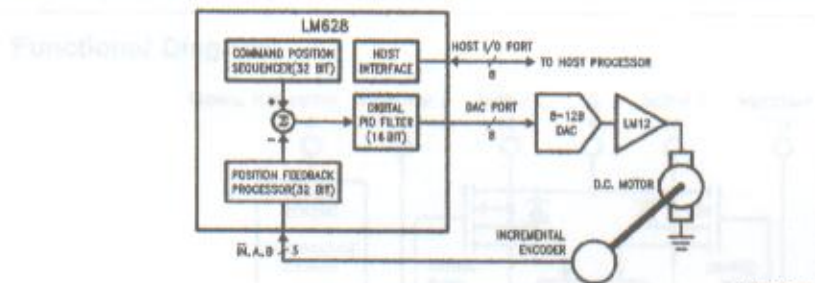
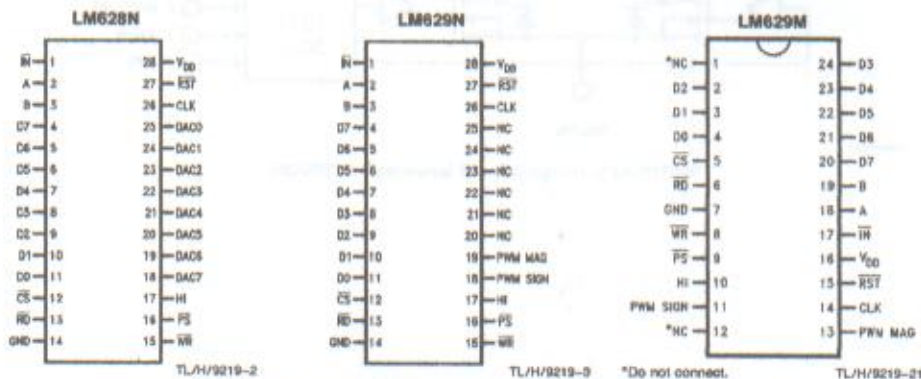


FIGURE 1. Typical System Block Diagram

TL/H/9219-1

Connection Diagrams



Order Number LM629M-6, LM629M-8, LM628N-6, LM628N-8, LM629N-6 or LM629N-8
See NS Package Number M24B or N28B

LMD18200 3A, 55V H-Bridge

General Description

The LMD18200 is a 3A H-Bridge designed for motion control applications. The device is built using a multi-technology process which combines bipolar and CMOS control circuitry with DMOS power devices on the same monolithic structure. Ideal for driving DC and stepper motors; the LMD18200 accommodates peak output currents up to 6A. An innovative circuit which facilitates low-loss sensing of the output current has been implemented.

Features

- Delivers up to 3A continuous output
- Operates at supply voltages up to 55V
- Low $R_{ON}(ON)$ typically 0.3 Ω per switch
- TTL and CMOS compatible inputs

- No "shoot-through" current
- Thermal warning flag output at 145°C
- Thermal shutdown (outputs off) at 170°C
- Internal clamp diodes
- Shorted load protection
- Internal charge pump with external bootstrap capability

Applications

- DC and stepper motor drives
- Position and velocity servomechanisms
- Factory automation robots
- Numerically controlled machinery
- Computer printers and plotters

Functional Diagram

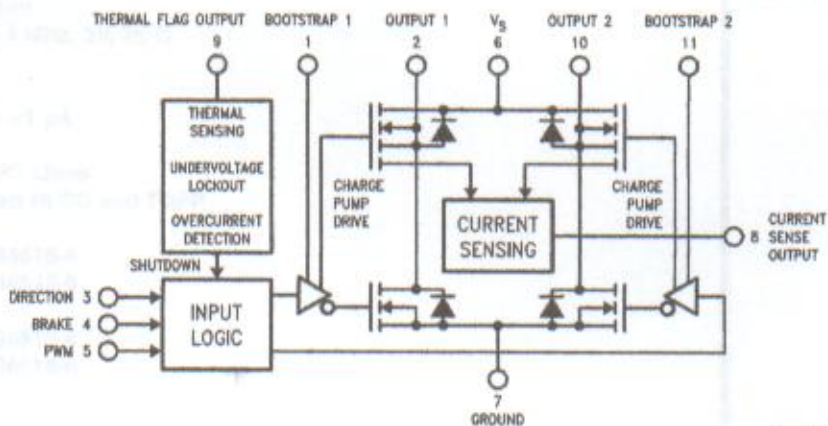


FIGURE 1. Functional Block Diagram of LMD18200

DS010568-1

Features

Utilizes the AVR[®] RISC Architecture

AVR – High-performance and Low-power RISC Architecture

- 118 Powerful Instructions – Most Single Clock Cycle Execution
- 32 x 8 General-purpose Working Registers
- Up to 8 MIPS Throughput at 8 MHz

Data and Nonvolatile Program Memory

- 8K Bytes of In-System Programmable Flash
Endurance: 1,000 Write/Erase Cycles

- 512 Bytes of SRAM

- 512 Bytes of In-System Programmable EEPROM

- Endurance: 100,000 Write/Erase Cycles
- Programming Lock for Flash Program and EEPROM Data Security

Peripheral Features

- One 8-bit Timer/Counter with Separate Prescaler
- One 16-bit Timer/Counter with Separate Prescaler
Compare, Capture Modes and Dual 8-, 9-, or 10-bit PWM
- On-chip Analog Comparator
- Programmable Watchdog Timer with On-chip Oscillator
- Programmable Serial UART
- Master/Slave SPI Serial Interface

Special Microcontroller Features

- Low-power Idle and Power-down Modes
- External and Internal Interrupt Sources

Specifications

- Low-power, High-speed CMOS Process Technology
- Fully Static Operation

Power Consumption at 4 MHz, 3V, 25°C

- Active: 3.0 mA
- Idle Mode: 1.0 mA
- Power-down Mode: <1 µA

I/O and Packages

- 32 Programmable I/O Lines
- 40-lead PDIP, 44-lead PLCC and TQFP

Operating Voltages

- 2.7 - 6.0V for AT90S8515-4
- 4.0 - 6.0V for AT90S8515-8

Speed Grades

- 0 - 4 MHz for AT90S8515-4
- 0 - 8 MHz for AT90S8515-8



8-bit AVR[®]
Microcontroller
with 8K Bytes
In-System
Programmable
Flash

AT90S8515



APÉNDICE D: Motor SWF de ángulo recto

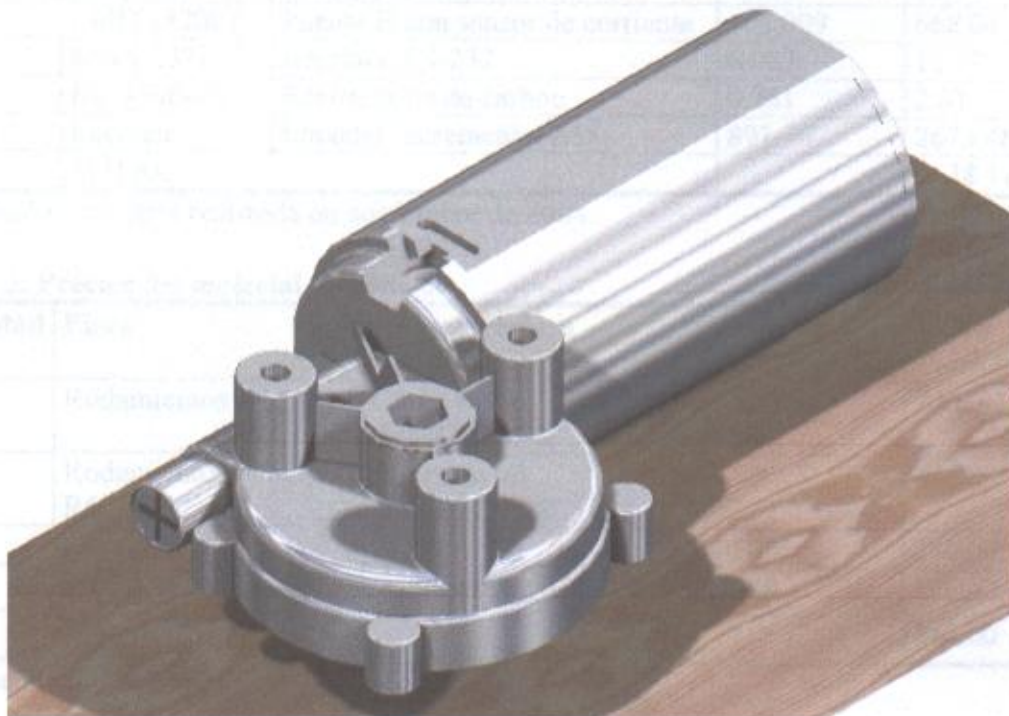
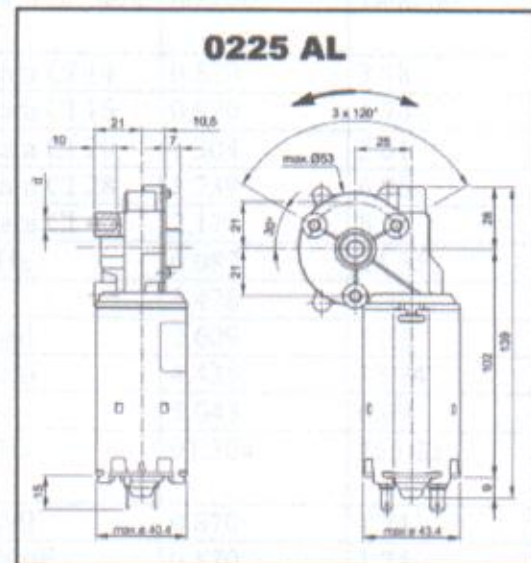
Series 0225 (GMPPG)

Motor type 404 774

Valeo

MOTORS &
ACTUATORS

Technical Data			
Rated voltage	U_N	[Volt]	24
No-load speed	n_0	[min ⁻¹]	35
Nominal torque	M_N	[Nm]	1,30
ON time	%		10
ON		[min]	
Starting torque	M_A	[Nm]	7,50
Gear ratio	i		72/1
Armature resistance, 2 bars	R	[mΩ]	5670,00
4 bars	R	[mΩ]	4200,00
Armature inductance, 2 bars	L	[mH]	5,65
4 bars	L	[mH]	3,90
Armature load inertia	J_R	[kgm ²] · 10 ⁻⁴	
Gear wheel material			plastic
Hall IC			
Pulses/rev. drive shaft			
Output channels			
Remarks	d = for self-tapping screw	M 5	
Enclosure class			IP 30
Weight		[kg]	0,570



APÉNDICE E: Tablas de precios de componentes

Tabla 1. Precios del material electrónico:

Cantidad	Pieza	Características	Precio unitario (\$)	Total (\$)
3	Motores de CD	Motor Valeo tipo 404 774 Serie 0225 (GMPG)	600.00	1800.00
4	14 P	Base doble contacto para CI 14	0.870	3.48
2	16 P	Base doble contacto para CI 16	0.870	1.74
2	20 P	Base doble contacto para CI 20	1.304	2.61
4	28 P	Base doble contacto para CI 28	1.739	6.96
4	40 P	Base doble contacto para CI 40	2.174	8.70
4	4 MHz	Cristal de cuarzo 4 MHz	6.087	24.35
2	74HC14	Inversor	3.478	6.96
2	74LS00	Compuerta NAND Cuad	2.609	5.22
4	74LS245	Transceptor de bus Octal	4.435	17.74
2	74LS32	Compuerta OR Cuad	3.043	6.09
4	ATMEGA 16-16PT	AVR MCU 16K 16MHz	91.304	365.22
2	CC 0.1/50V	Capacitor cerámico .1MF	0.870	1.74
2	CC 100/500V	Capacitor cerámico 100pF	0.870	1.74
2	CC 1/50V	Condensador electrolítico	0.870	1.74
2	DB 9	Conector DB 9 Hembra	5.217	10.43
4	HEADER 2	Tira con 36 terminales 2 líneas	4.348	17.39
4	LMD 18200T	Puente H con sensor de corriente	167.009	668.04
2	MAX 232N	Interface RS-232	6.087	12.17
1	RC 330E- 1/2	Resistencias de carbón	0.261	2.61
3	Encoder	Encoder Incremental E5S	891.36	2673.48
	TOTAL			5638.41

Cotización y compra realizada en noviembre de 2004.

Tabla 2. Precios del material mecánico:

Cantidad	Pieza	Características	Costo unitario (\$)	Costo Total (\$)
2	Rodamientos R4.2Z	Carga radial d=3/8", D=5/8"	12.00	24.00
2	Rodamientos R6.2Z	Carga radial d=1/4", D=7/8"	12.00	24.00
2	Engranés rectos	Piñón N=24	100.00	200.00
2	Engranés rectos	Engrane N=66	150.00	300.00
	TOTAL			548.00

Cotización y compra realizada en noviembre de 2004.

Tabla 2. Precios del material mecánico (continuación):

Cantidad (Kg)	Pieza	Descripción Dimensiones (mm)	Precio unitario (\$)	Total (\$)
0.70	Canal de aluminio	3.2*76.2*12.7*3660	53.87	37.71
0.70	Placa de aluminio	3.40*914*2440	51.24	35.87
1.20	Placa de aluminio	4.19*914*2440	52.82	63.38
2.10	Placa de aluminio	6.05*914*2440	51.24	107.60
7.50	Placa de aluminio	12.7*914*2440	53.87	387.45
0.10	Barra circular de aluminio	9.5*3660	53.87	63.00
0.20	Barra circular de aluminio	12.7*3660	53.87	5.39
0.30	Barra circular de aluminio	19.0*3660	53.87	10.77
0.50	Barra circular de aluminio	38.1*3660	53.87	16.16
0.60	Barra circular de aluminio	44.4*3660	53.87	26.94
0.20	Angulo de aluminio	6.4*25.4*25.4*3660	53.87	32.32
0.30	Barra hexagonal de latón	7.9	45.20	18.08
	TOTAL			804.67

Cotización y compra realizada en noviembre de 2004.

Tabla 3. Costos totales

TOTAL DE MATERIAL ELECTRÓNICO	\$ 5638.41
TOTAL DE MATERIAL MECÁNICO	\$ 1352.67
TOTAL	\$ 6991.08

Cotización y compra realizada en noviembre de 2004.

Densidad	2.71 g/cm ³
Módulo de elasticidad	7.00x10 ¹⁰ N/m ²
Módulo de torsión	2.65x10 ¹⁰ N/m ²
Temperatura de fusión	660 °C
Calor específico a 100 °C	0.21 cal/g°C
Coefficiente de expansión térmica	23x10 ⁻⁶ %
Conductividad térmica a 125 °C	0.48 cal/cm ² °C
Resistividad eléctrica (30 °C)	0.033 ohm cm ² /m

APÉNDICE F: Propiedades del aluminio

ALEACIONES DE ALUMINIO 6063 T5

Se parte de aluminio primario al cual se le adicionan materiales que le aportan características especiales, según el uso que se le vaya a dar al metal.

Los principales aleantes son: cobre, silicio, magnesio, zinc y manganeso.

Las aleaciones de aluminio que permiten ser extruídas se clasifican en el grupo del 6000, cuya composición depende del mineral o elemento que se le adicione en mayor proporción.

TEMPLE

El temple es una condición que se produce en el metal o aleación por efecto del tratamiento mecánico o térmico impartándole estructuras y propiedades mecánicas características. Los procedimientos térmicos que aumentan la resistencia a estas aleaciones son el tratamiento térmico en solución y el envejecimiento.

El tratamiento térmico en solución requiere que se caliente la aleación hasta una temperatura por debajo del punto de fusión por un periodo de tiempo específico, seguido de disminución rápida de dicha temperatura.

El envejecimiento es un tratamiento térmico a relativa baja temperatura que produce endurecimiento adicional al material tratado en solución.

TIPOS DE TRATAMIENTO

T- Se aplica a los productos cuyas propiedades mecánicas se obtienen mediante tratamientos térmicos. Esta letra se coloca después del grupo de aleación y antes del número del tratamiento específico.

Ej: AA6063 T5 (Aleación 6063 temple cinco).

T4- Enfriamiento por aire después de extruír y envejecimiento natural.

T5- Enfriamiento por aire después de extruír y envejecimiento artificial.

T6- Enfriamiento por agua después de extruír y envejecimiento artificial.

TABLA F. CARACTERÍSTICAS DEL ALUMINIO 6063 T5.

Peso específico	2.71 gr./cm³
Módulo de elasticidad	7.000 Kg/mm²
Módulo de rigidez	2.650 Kg/mm²
Temperatura de fusión	660 °C
Calor específico (0a 100 °C)	0.21 cal/g°C
Coefficiente de expansión lineal	23 x 10⁻⁶ °C
Conductividad térmica (25 °C)	0.48 cal/cms °C
Resistividad eléctrica (20 °C)	0.033 ohms mm²/m

APÉNDICE G: Código de la interfaz usuario-máquina

A continuación se muestra el código de ciertas funciones transcendentales para el control del puerto serie y del control manipulador, en el CD anexo se puede encontrar el código completo.

//Funcion creada para abrir el puerto.

```
bool __fastcall TMain::AbrirPuerto()
```

```
{
    DCB dcbPuerto;    //DCB define una estructura para propiedades
                    //de un dispositivo de comunicacion serial.
```

// En las siguiente lineas se abre el puerto.

```
hComm = CreateFile(nombrePuerto,
    GENERIC_READ | GENERIC_WRITE,
    0,
    NULL,
    OPEN_EXISTING,
    0,
    NULL);
```

//Si el puerto no se puede abrir se regresa una valor false.

```
if(hComm == INVALID_HANDLE_VALUE)
```

```
return false;
```

```
else
```

```
{
```

//Se obtienen las propiedades actuales de la estructura COMMTIMEOUTS.

```
GetCommTimeouts(hComm,&ctmoOld);
```

//Se colocan las propiedades de la estructura COMMTIMEOUTS.

```
ctmoNew.ReadTotalTimeoutConstant = 100;
```

```
ctmoNew.ReadTotalTimeoutMultiplier = 0;
```

```
ctmoNew.WriteTotalTimeoutMultiplier = 0;
```

```
ctmoNew.WriteTotalTimeoutConstant = 0;
```

```
SetCommTimeouts(hComm, &ctmoNew);
```

//Se introducen las propiedades para la comunicacion:

//Baud Rate, Paridad, Longitud de Byte, Bits de parada.

```
dcbPuerto.DCBlength = sizeof(DCB);
```

```
GetCommState(hComm, &dcbPuerto);
```

```
BuildCommDCB("9600,NOPARITY,8,ONESTOPBIT", &dcbPuerto);
```

```
SetCommState(hComm, &dcbPuerto);
```

```
if(!EnviarDato(0xF0)) //Se envía el dato 0xF0 " Reset de los LM629's "
```

```
Application->MessageBox("El puerto se abrio correctamente pero no se ha podido
establecer comunicación con el manipulador, revise que se encuentre conectado en el puerto
correcto", "Mensaje de TT BMAD", MB_OK|MB_ICONINFORMATION);
```



```

return true;
}
}

//Funcion creada para cerrar el puerto.
void __fastcall TMain::CerrarPuerto()
{
    if(PuertoAbierto==true)
    {
        PurgeComm(hComm, PURGE_RXABORT);
        SetCommTimeouts(hComm, &ctmoOld);
        CloseHandle(hComm);
    }
}

//Función creada para realizar la cinemática directa
void __fastcall TMain::CinemáticaDirecta(void)
{
    x=cos(t1)*(L3*cos(t2+t3)+L2*cos(t2));
    y=sin(t1)*(L3*cos(t2+t3)+L2*cos(t2));
    z=-L3*sin(t2+t3)-sin(t2)*L2;
}

//Función creada para realizar la cinemática inversa
void __fastcall TMain::CinemáticaInversa(void)
{
    t1=atan2(y,x);
    temp=(x*x+y*y+z*z-L2*L2-L3*L3)/(2*L2*L3);
    t3=atan2(sqrt(1-temp*temp),temp);
    t2=atan2(z,sqrt(x*x+y*y))-atan2(L3*sin(t3),L2+L3*cos(t3));

    if(Jacobiano()==0)
        Application->MessageBoxA("Existen singularidades en la trayectoria o se ha llegado a limite
del espacio de trabajo","Mensaje!!!",MB_OK);
}

//Función creada para determinar si existen singularidades
bool __fastcall TMain::Jacobiano()
{
    DetJ=(L2*L2*L3*sin(t2+t3)-L2*L3*L3*sin(t2)+L2*L3*L3*sin(t2+2*t3)+L2*L2*L3*sin(t3-
t2))/2;
    return DetJ;
}

//Función creada para enviar los datos codificados vía puerto serie,
//donde se muestra el manejo de archivos.
void __fastcall TMain::ButtonIniciarClick(TObject *Sender)

```

```

{
if(EnviarDato(0xF7)) //0xF7
{
if ((MovimientosCod= fopen("MovimientosCod.bmad", "rt")) == NULL)
{
printf(stderr, "No se abrió el archivo de los Movimientos.\n");
Application->MessageBox("No se pudo abrir el registro de los Movimientos.",
"MENSAJE", MB_OK | MB_ICONINFORMATION);
}

if ((VelocidadesCod= fopen("VelocidadesCod.bmad", "rt")) == NULL)
{
printf(stderr, "No se abrió el archivo de las Velocidades.\n");
Application->MessageBox("No se pudo abrir el registro de las Velocidades.",
"MENSAJE", MB_OK | MB_ICONINFORMATION);
}

fseek(MovimientosCod,-5,SEEK_END);
fgets(NumPuntos.c_str(),5,MovimientosCod);
NumPuntosint=atoi(NumPuntos.c_str());

Bytes(NumPuntosint);
if(!EnviarDato(dato1)) ErrorEnvio();
if(!EnviarDato(dato2)) ErrorEnvio();

rewind(MovimientosCod);
rewind(VelocidadesCod);
i=0;

do
{
fgets(Revo1.c_str(),6,MovimientosCod);
fseek(MovimientosCod,2,SEEK_CUR);
RevCod1=atof(Revo1.c_str());
fgets(Revo2.c_str(),6,MovimientosCod);
fseek(MovimientosCod,2,SEEK_CUR);
RevCod2=atof(Revo2.c_str());
fgets(Revo3.c_str(),6,MovimientosCod);
fseek(MovimientosCod,4,SEEK_CUR);
RevCod3=atof(Revo3.c_str());

fgets(Velo1.c_str(),6,VelocidadesCod);
fseek(VelocidadesCod,2,SEEK_CUR);
VelCod1=atof(Velo1.c_str());
fgets(Velo2.c_str(),6,VelocidadesCod);
fseek(VelocidadesCod,2,SEEK_CUR);

```



```

VelCod1=atof(Velo2.c_str());
fgets(Velo3.c_str(),6,VelocidadesCod);
fseek(VelocidadesCod,4,SEEK_CUR);
VelCod1=atof(Velo3.c_str());

Bytes(RevCod1);
if(!EnviarDato(dato1)) ErrorEnvio();
if(!EnviarDato(dato2)) ErrorEnvio();

Bytes(VelCod1);
if(!EnviarDato(dato1)) ErrorEnvio();
if(!EnviarDato(dato2)) ErrorEnvio();

Bytes(RevCod2);
if(!EnviarDato(dato1)) ErrorEnvio();
if(!EnviarDato(dato2)) ErrorEnvio();

Bytes(VelCod2);
if(!EnviarDato(dato1)) ErrorEnvio();
if(!EnviarDato(dato2)) ErrorEnvio();

Bytes(RevCod3);
if(!EnviarDato(dato1)) ErrorEnvio();
if(!EnviarDato(dato2)) ErrorEnvio();

Bytes(VelCod3);
if(!EnviarDato(dato1)) ErrorEnvio();
if(!EnviarDato(dato2)) ErrorEnvio();

i++;
}
while(i<NumPuntosint);

fclose(MovimientosCod);
fclose(VelocidadesCod);
}
else
ErrorEnvio();

StatusBar1->Panels->Items[1]->Text="Estado del manipulador: Descanso";
}
//-----

```

APÉNDICE H: Protocolo de comunicación

En este apéndice se proporcionan los comandos, es decir, los números hexadecimales que deben transmitirse de la PC al AVR y viceversa para realizar una operación.

Inicialización del sistema

Una vez que el sistema de control ha sido encendido el primer comando que se debe de enviar de la PC al AVR es $0xF0_H$, una vez que el comando ha sido enviado al AVR, éste debe de regresar el dato $0xF0_H$ a la PC para confirmar que existe una correcta comunicación entre ellos.

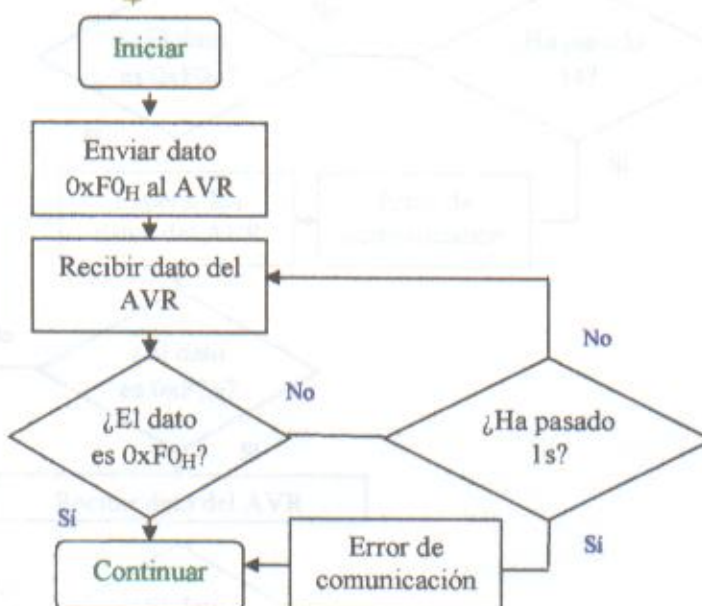


Figura II. Diagrama de flujo para la inicialización del sistema.

Establecimiento de HOME

Una vez que se ha confirmado la comunicación entre la PC y el AVR, se debe de enviar el comando para que el manipulador establezca HOME automáticamente, dicho comando debe ser $0xF1_H$; cuando el AVR recibe este dato empieza a mover uno a uno los eslabones. Primero se mueve la base hasta activar el sensor de límite correspondiente y el AVR envía a la PC el comando $0xF2_H$ (por lo que la PC debe de estar preparada para recibir dicho comando), éste comando indica que se ha completado el establecimiento del HOME con referencia a la base, después se mueve el primer eslabón y al activarse el sensor de límite correspondiente se detiene el primer eslabón por lo tanto el AVR envía a la PC el comando $0xF3_H$, éste comando indica que se ha completado el establecimiento del HOME con referencia al primer eslabón, de igual manera se mueve el segundo eslabón y al activarse el sensor de límite correspondiente se para el movimiento para que el AVR envíe el comando $0xF4_H$ a la PC, éste comando indica que se ha completado el establecimiento del HOME con referencia al segundo eslabón y a su vez se ha finalizado la tarea de establecer HOME del manipulador.

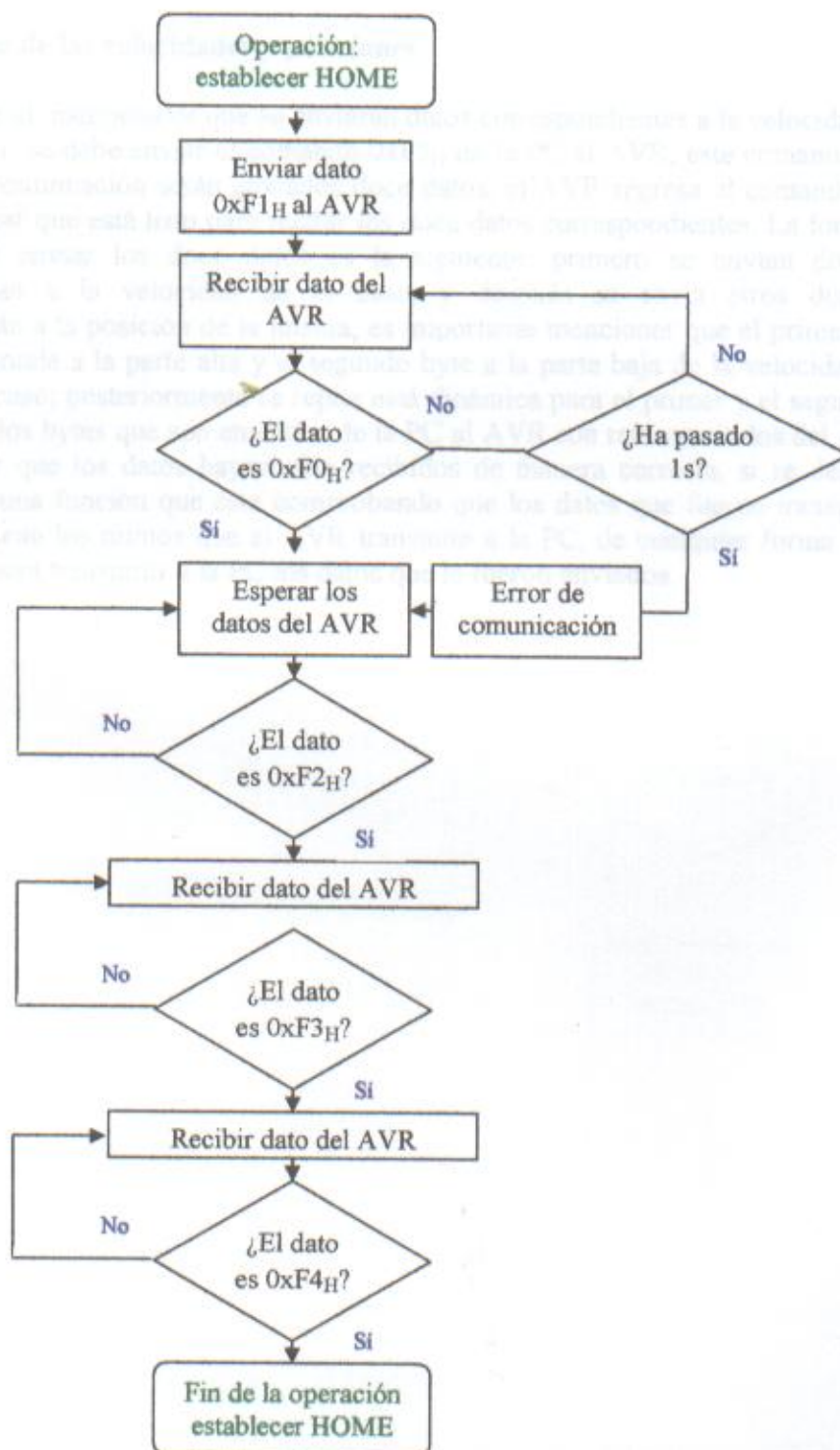


Figura III. Diagrama de flujo para el establecimiento del HOME.

Actualización de las velocidades y posiciones

Para indicarle al manipulador que se enviarán datos correspondientes a la velocidad y posición de cada motor, se debe enviar el comando $0xF5_H$ de la PC al AVR, este comando le indica al AVR que a continuación serán enviados doce datos, el AVR regresa el comando $0xF5_H$ a la PC para indicar que está listo para recibir los doce datos correspondientes. La forma en la que se deben de enviar los doce datos es la siguiente: primero se envían dos bytes que corresponderán a la velocidad de la base, y después se envía otros dos bytes que corresponderán a la posición de la misma, es importante mencionar que el primer byte que se envía corresponde a la parte alta y el segundo byte a la parte baja de la velocidad o posición según sea el caso; posteriormente se repite esta dinámica para el primer y el segundo eslabón. Cada uno de los bytes que son enviados de la PC al AVR son retransmitidos del AVR a la PC para asegurar que los datos hayan sido recibidos de manera correcta, si se desea se puede implementar una función que este comprobando que los datos que fueron transmitidos de la PC al AVR sean los mismos que el AVR transmite a la PC, de cualquier forma el AVR está programado para transmitir a la PC los datos que le fueron enviados.



Figura 11. Diagrama de flujo para la actualización de las posiciones y velocidades.

Actualización de las constantes para el control PID

Al igual al que se explicó en el capítulo anterior, se actualizan las constantes K_{p1} , K_{i1} y K_{d1} de cada uno de los eslabones, el comando que se debe de enviar de la PC al AVR es $0xF6_H$, con este comando se indica al AVR que se actualizarán los datos, es decir, una vez que se haya enviado el comando $0xF6_H$ el AVR estará en espera de nuevos datos que correspondieren a las tres constantes de cada uno de los eslabones, primero se envía la K_{p1} , K_{i1} y K_{d1} de la base, después



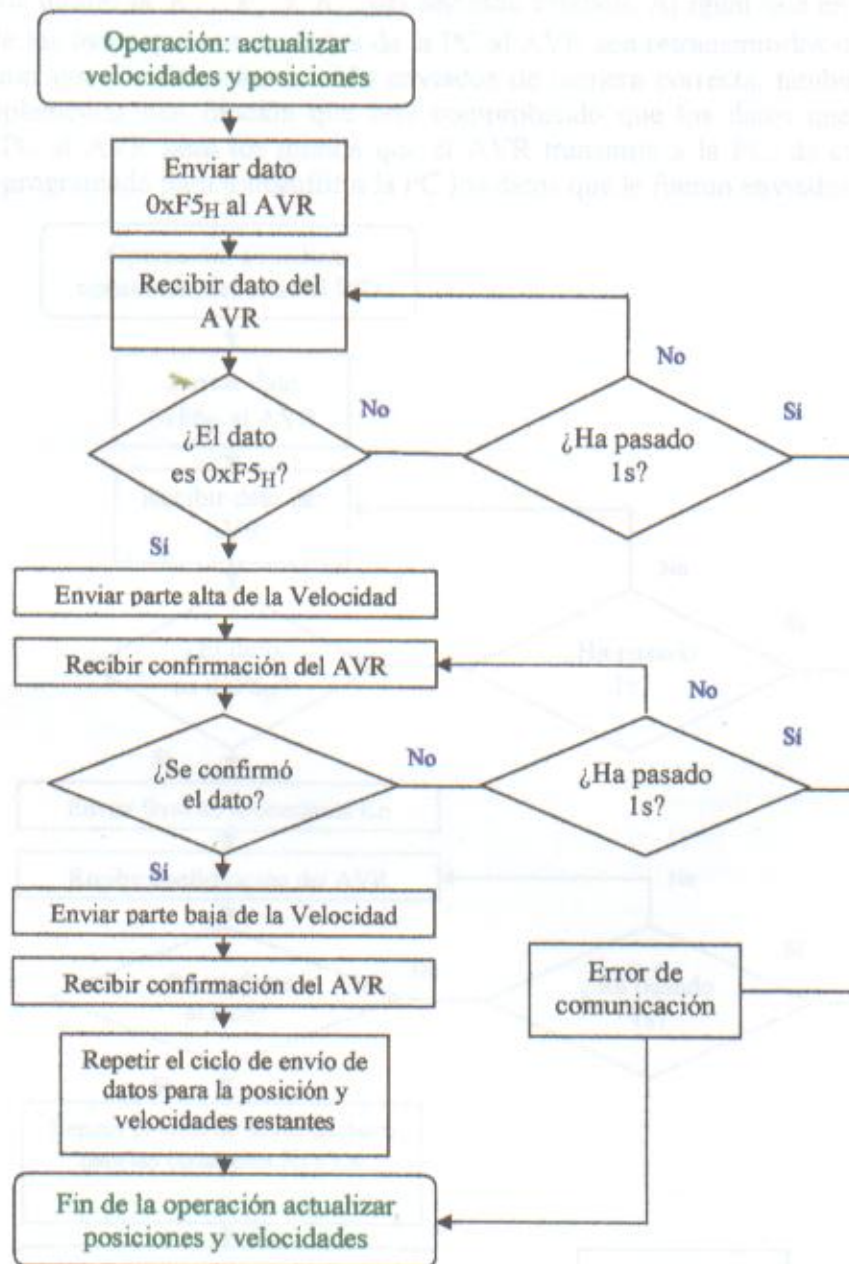


Figura IV. Diagrama de flujo para la actualización de las posiciones y velocidades.

Actualización de las constantes para el control PID

Si lo que se quiere es modificar las constantes K_p , K_i y K_d de cada uno de los eslabones, el comando que se debe de enviar de la PC al AVR es $0xF6_H$, con este comando se indica al AVR que se actualizarán los filtros, es decir, una vez que se haya enviado el comando $0xF6_H$ el AVR estará en espera de nueve datos que corresponderán a las tres constantes de cada uno de los eslabones, primero se envía la K_p , K_i y K_d de la base, después la K_p , K_i y K_d del

primer eslabón y por último la K_p , K_i y K_d del segundo eslabón. Al igual que en el caso anterior cada uno de los bytes que son enviados de la PC al AVR son retransmitidos del AVR a la PC para asegurar que los datos hayan sido enviados de manera correcta, también si se desea se puede implementar una función que este comprobando que los datos que fueron transmitidos de la PC al AVR sean los mismos que el AVR transmite a la PC, de cualquier forma el AVR está programado para transmitir a la PC los datos que le fueron enviados.

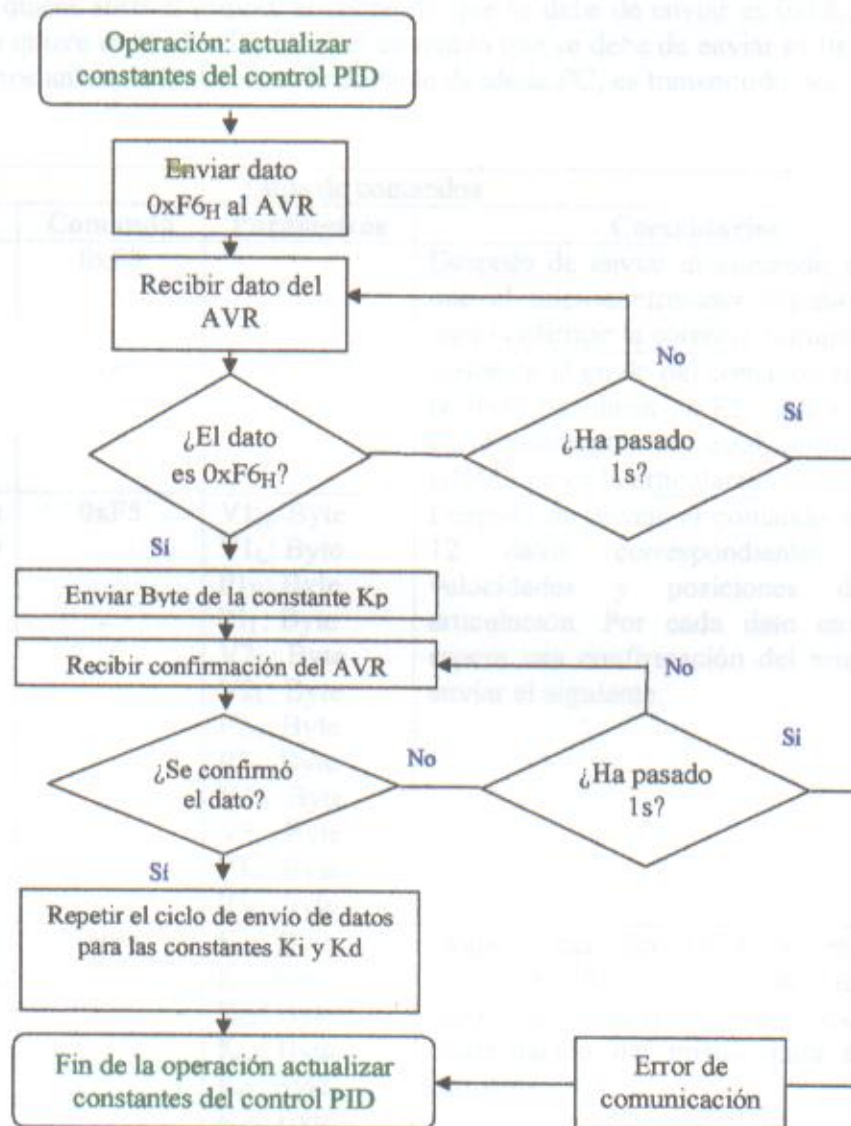


Figura V. Diagrama de flujo para la actualización de las constantes del control PID.

Inicialización del movimiento

Una vez que ya se han cargado las velocidades y las posiciones a cada uno de los eslabones se puede llevar a cabo el inicio del movimiento, esto se logra con el comando 0xF7H. Cabe mencionar que no es necesario cargar los filtros (comando 0xF6H), debido a que se cuenta con

ciertas constantes predefinidas para cada uno de los eslabones (K_p , K_i y K_d). Una vez que se ha completado el movimiento de los eslabones, el AVR transmite a la PC el comando $0xF7_H$, esto le indica a la PC que ya ha sido completada la trayectoria y que puede entonces recibir otras velocidades y posiciones (mediante el comando $0xF5_H$) para ejecutar otro movimiento (mediante el comando $0xF7_H$).

En caso de que se quiera abrir el gripper el comando que se debe de enviar es $0xF8_H$, si por contrario, lo que se quiere es cerrar el gripper el comando que se debe de enviar es $0xF9_H$. Al igual que en los casos anteriores, el comando enviado desde la PC, es transmitido por el AVR a la PC.

Tabla de comandos

Operación	Comando	Parámetros	Comentarios
Inicializar sistema	$0xF0$		Después de enviar el comando se espera que el microcontrolador regrese el dato para confirmar la correcta comunicación.
Establecimiento del HOME	$0xF1$		Posterior al envío del comando se esperan recibir los datos $0xF2$, $0xF3$ y $0xF4$ correspondientes al establecimiento del HOME de cada articulación.
Actualización de las posiciones y velocidades	$0xF5$	V _{1H} : Byte V _{1L} : Byte P _{1H} : Byte P _{1L} : Byte V _{2H} : Byte V _{2L} : Byte P _{2H} : Byte P _{2L} : Byte V _{3H} : Byte V _{3L} : Byte P _{3H} : Byte P _{3L} : Byte	Después de enviar el comando se envían 12 datos correspondientes a las velocidades y posiciones de cada articulación. Por cada dato enviado se espera una confirmación del mismo para enviar el siguiente.
Actualización de las constantes para el control PID	$0xF6$	K _{p1} : Byte K _{i1} : Byte K _{d1} : Byte K _{p2} : Byte K _{i2} : Byte K _{d2} : Byte K _{p3} : Byte K _{i3} : Byte K _{d3} : Byte	Después del dato $0xF6$ se envían las constantes PID de cada articulación. Por cada dato enviado se espera una confirmación del mismo para enviar el siguiente.
Inicialización del movimiento	$0xF7$		En este caso solo se espera la confirmación del dato.
Abrir gripper	$0xF8$		Se espera la confirmación del dato.
Cerrar gripper	$0xF9$		Se espera la confirmación del dato.

APÉNDICE I: Recepción y transmisión de datos mediante la UART del AVR AT90S8515

El siguiente código sirve para recibir y transmitir datos en forma serial a través de la UART del AVR AT90S8515 utilizando el protocolo RS-232, la configuración es la siguiente: la longitud del dato debe de ser de 8 bits, no paridad, 1 bit de parada, la velocidad de transmisión y recepción de 9600 baudios (aunque está puede ser modificada) cuando se utiliza un cristal de 4 Mhz, de la misma forma cuando se transmite del AVR a la PC, ésta debe conservar la misma configuración.

```
; Archivo: uart8515.asm
; Proyecto: Brazo Manipulador Para Aplicaciones Didácticas
; Descripción: Recepción y transmisión con el uso de la UART del AVR AT90S8515
; Fecha de creación: 12/03/05

;----- Inclusión de Archivos -----
.include "8515def.inc"
;*****

;----- Definición de variables -----
.def T = r16 ; Registro de trabajo
.def dato = r17 ; Registro donde se almacena el dato recibido/enviado

; /*-----*/
; /*- Código de segmento -*/
; /*-----*/

.cseg

; /*-----*/
; /*- Dirección del inicio del programa -*/
; /*-----*/

.org 0x0000
        rjmp INICIO

; /*-----*/
; /*- Dirección de las interrupciones empleadas -*/
; /*-----*/

.org 0x0009
        rjmp USART_RXC
.org 0x000B
        rjmp USART_TXC

; /*-----*/
; /*- Inicio de programa -*/
; /*-----*/

INICIO:

; /*-----*/
; /*- Se inicializa la Pila -*/
; /*-----*/

        ldi T,low(RAMEND)
        out SPL,T
        ldi T,high(RAMEND)
        out SPH,T

; /*-----*/
; /*- Se limpia los puertos de E/S -*/
; /*-----*/

        ser T ;Carga 0xFF en el registro "temp"
```



```

        out ddra,T           ;Puerto A como salida
        out ddrb,T           ;Puerto B como salida
        out ddrc,T           ;Puerto C como salida
        out ddrd,T           ;Puerto D como salida

        clr T                 ;Carga 0x00 en temp
        out porta,T          ;Limpia el puerto A
        out portb,T          ;Limpia el puerto B
        out portc,T          ;Limpia el puerto C
        out portd,T          ;Limpia el puerto D

; /*-----*/
; /*- Configura la velocidad a 9600 Baudios -*/
; /*- con un cristal de 4 MHz -*/
; /*-----*/

        ldi T,0x19
        out UBRR,T

; /*-----*/
; /*- Habilita la recepción y transmisión de datos -*/
; /*-----*/

        ldi T, (1<<RXEN) | (1<<RXCIE) | (1<<TXEN)
        out UCR,T

; /*-----*/
; /*- Configura: Modo asíncrono, 2 bits de parada -*/
; /*- sin paridad, 8 bits de datos -*/
; /*-----*/

ESPERA:

; /*-----*/
; /*- Habilitamos las interrupciones globales -*/
; /*-----*/

        sei

; /*-----*/
; /*- Espera recibir un dato -*/
; /*-----*/

        rjmp ESPERA

; /*-----*/
; /*- Subrutina de interrupción para recibir datos -*/
; /*-----*/

USART_RXC:

; /*-----*/
; /*- Almacena el dato enviado de la PC y lo muestra en el Puerto C -*/
; /*-----*/

        in dato,UDR
        out portc,dato

; /*-----*/
; /*- Manda a llamar a la subrutina de interrupción para transmitir datos -*/
; /*- para determinar si es posible enviar datos a la PC -*/
; /*-----*/

        rcall USART_TXC

; /*-----*/
; /*- Cuando se habilita la transmisión de datos se -*/
; /*- transmite el dato recibido de la PC a la PC -*/
; /*-----*/

```

```
        out UDR, dato
        reti

; /*-----*/
; /*- Subrutina de interrupción para enviar datos a la PC -*/
; /*-----*/

USART_TXC:

; /*-----*/
; /*- Espera a que el dato se haya enviado -*/
; /*-----*/

        sbis USR, UDRE
        rjmp USART_TXC
        ret
```

El programa anterior realiza lo que se conoce como “*eco*”, es decir, el dato que se transmite de la PC al microcontrolador (en este caso un AVR) es recibido por el microcontrolador y retransmitido del microcontrolador a la PC.

BIBLIOGRAFIA:

- [1]
John J. Craig, Introduction to Robotics, second edition, Addison Wesley Publishing Company, 1989
- [2]
Barrientos Antonio, Fundamentos de Robótica, Mc Graw Hill, 1999
- [3]
Mott Robert L., Diseño de Elementos de Máquinas, Prentice Hall, 1995
- [4]
Allen S. Hall, Teoría y problemas de diseño de máquinas, Primera edición; McGraw-Hill. México.1990. pp. 131.
- [5]
La Asociación para la Práctica y Enseñanza de la Robótica
<http://usuarios.bitmailer.com/aperobot/>
- [6]
The University of Texas at Austin
<http://www.robotics.utexas.edu/>
- [7]
The Tech Museum of Innovation
<http://www.thetech.org/robotics/>
- [8]
Enrayala, Sistemas de cómputo
<http://prodigyweb.net.mx/enrayala>
- [9]
LEGO
<http://mindstorms.lego.com/>
- [10]
ROBODACTA
<http://www.robodacta.com/>