

**RADAC-ROBOT DE APLICACIÓN DIDÁCTICA PARA EL APRENDIZAJE
CONCEPTUAL DE LA AUTOMATIZACIÓN Y CONTROL EN LOS
LABORATORIOS DE INGENIERÍA ELECTRÓNICA DE LA UNIVERSIDAD DE
NARIÑO**

CRISTIAN ANDRÉS PATIÑO

**UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
PASTO. COLOMBIA
2012**

**RADAC-ROBOT DE APLICACIÓN DIDÁCTICA PARA EL APRENDIZAJE
CONCEPTUAL DE LA AUTOMATIZACIÓN Y CONTROL EN LOS
LABORATORIOS DE INGENIERÍA ELECTRÓNICA DE LA UNIVERSIDAD DE
NARIÑO**

CRISTIAN ANDRÉS PATIÑO

Trabajo de grado presentado para optar al título de Ingeniero Electrónico

Director del proyecto

ROLANDO BARAHONA

**UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
PASTO. COLOMBIA
2012**

Las ideas y conclusiones aportadas en este trabajo de grado son responsabilidad exclusiva del autor. Artículo 1° del acuerdo n°324 de octubre 11 de 1966 emanado del honorable consejo directivo de la Universidad de Nariño

Nota de aceptación

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Pasto, septiembre 20 de 2012

AGRADECIMIENTOS

Quiero expresar mi agradecimiento a mi familia, por su apoyo, a mi madre, que en paz descanse, quien con su ejemplo me ha inspirado; a los ingenieros Rolando Barahona y Wilson Achicanoy, por su generosa colaboración en el desarrollo de este proyecto.

RESUMEN

Este proyecto está enfocado a complementar los elementos didácticos que actualmente poseen los laboratorios de Ingeniería Electrónica de la Universidad de Nariño. En sí este trabajo de grado propone la alternativa de diseñar y construir elementos didácticos enfocados a reforzar y facilitar los procesos de enseñanza que se dan a lugar dentro del programa de ingeniería electrónica de la Universidad de Nariño. En particular, este proyecto propone el diseño y construcción de un robót de aplicación didáctica.

ABSTRACT

This project has the purpose of the building of a didactic robot, whose uses include the teaching of robotics into the laboratories of the Electronic Engineering program of the Universidad de Nariño.

CONTENIDO

	Pág.
INTRODUCCIÓN	18
1. TEMA	19
1.1. TÍTULO	19
1.2. MODALIDAD	19
1.3. LÍNEA DE INVESTIGACIÓN	19
1.4. ALCANCE Y DELIMITACIÓN	19
2. DEFINICIÓN DEL PROBLEMA	20
2.1. PLANTEAMIENTO DEL PROBLEMA	20
2.2. FORMULACIÓN DEL PROBLEMA	20
3. OBJETIVOS	21
3.1 OBJETIVO GENERAL	21
3.2 OBJETIVOS ESPECÍFICOS	21
4. JUSTIFICACIÓN	22
5. MARCO REFERENCIAL	23
5.1. AUTOMATIZACIÓN Y ROBÓTICA	23
5.1.1. La automatización fija	24
5.1.2. La automatización programable	24
5.1.3 La automatización flexible	25

	Pág.
5.2. MORFOLOGÍA DEL ROBOT Y SUS PERIFÉRICOS	26
5.2.1. Estructura Mecánica de un Robot.	26
5.2.2. Sensores	32
5.2.3. Actuadores	32
5.2.4. Efector final (pinzas y ejes de manos gripper)	33
5.3. PROGRAMACIÓN Y CONTROL DE ROBOTS	34
5.3.1 Unidades de control	34
6. MARCO REFERENCIAL	36
6.1. MARCO REFERENCIAL DE LA ENTIDAD	36
6.2. MARCO REFERENCIAL DEL PROYECTO	36
7. METODOLOGÍA	37
7.1. DESCRIPCIÓN METODOLÓGICA	37
7.2. MATERIALES Y HERRAMIENTAS METODOLÓGICAS	38
7.2.1. El manipulador	39
7.2.2. Circuitos de interfaz del robot	40
7.2.3. Estación de trabajo	41
7.2.4. Circuitos de alimentación del robot	41

	Pág.
8. FASE DE DESARROLLO	43
8.1. SELECCION DEL BRAZO ROBÓTICO	43
8.1.1 Ecuaciones cinemáticas y dinámicas Para las soluciones directa e inversa del robot	45
8.1.2 Diseño del software del robot	56
8.1.3 Construcción del manipulador	59
8.1.4. Fabricación de servomotores	61
8.1.5 Construcción circuito de interfaz por el microcontrolador 16F887	61
8.1.6. Construcción circuito de interfaz por el circuito integrado L293B	61
8.1.7. Elaboración Manual de usuario	63
9. RESULTADOS Y ANÁLISIS DE RESULTADOS	64
10. RECURSOS	66
10.1. RECURSOS HUMANOS	66
10.2. RECURSOS TECNOLÓGICOS	66
10.3. RECURSOS MATERIALES	66
10.4. RECURSOS FINANCIEROS	66
10.5. RECURSO OPERATIVO	67
11. CONCLUSIONES	68

	Pág.
RECOMENDACIONES	69
BIBLIOGRAFÍA	70
ANEXOS	72

LISTA DE TABLAS

	Pág.
Tabla 1. Actividades planificadas	38
Tabla 2. Distribución de los sistemas de referencia	49
Tabla 3. Relaciones entre articulaciones y motores escogidos	60

LISTA DE FIGURAS

	Pág.
Figura 1. Componentes básicos de un robot fijo	27
Figura 2. Tipos de posicionamiento	29
Figura 3. Punto focal de un robot	29
Figura 4. Espacio de trabajo del robot	30
Figura 5. Coordenadas de un robot	32
Figura 6. Pinza de tres puntos combinada con una unidad Lineal /giratoria	33
Figura 7. La mano humana puede ejecutar movimientos con 6 grados de libertad	34
Figura 8. Proceso de diseño	37
Figura 9. Robot didáctico	39
Figura 10. Manipulador	40
Figura 11. Circuitos de interfaz	41
Figura 12. Modelo utilizado en la construcción del robot	44
Figura 13. Modelo cinemático del robot	45
Figura 14. Proyección de los elementos del robot sobre el plano x-y	46
Figura 15. Sistemas de referencia asignados	48
Figura 16. Distribución de los sistemas de referencia asignados	49
Figura 17. Centro de masas y ejes de giro de los elementos del robot	52
Figura 18. Diagrama de bloques general del programa	56
Figura 19. Diagrama de bloques programa controlador dinámico	57

	Pág.
Figura 20. Diagrama de bloques programa controlador cinemático	57
Figura 21. Ventana de simulación gráfica	59
Figura 22. Construcción de Manipulador prototipo	59
Figura 23. Construcción de Manipulador	60
Figura 24. Motor reductor utilizado como servomotor	61
Figura 25. Circuito de controlador de motores paso a paso por PIC 16F887	61
Figura 26. Prototipo de implementación del controlador de potencia por C.I. L293B	62
Figura 27. Implementación del controlador por C.I. L293B en Circuito Impreso	62

LISTA DE ANEXOS

	Pág.
Anexo A. Planos manipulador robot vista isométrica	
Anexo B. Planos manipulador robot vista frontal	
Anexo C. Planos manipulador robot vista lateral	
Anexo D. Planos manipulador robot vista superior	
Anexo E. Software programación textual	73
Anexo F. Diagrama esquemático interfaz de potencia	74
Anexo G. Lista de Materiales interfaz de potencia	75
Anexo H. Programa para el control de motores paso a paso por 16F887	77
Anexo I. Manual del Usuario	

GLOSARIO

A/D: acrónimo de Análogo/Digital.

ADC: acrónimo de Analog to Digital Converter (Convertidor de análogo a digital)

ALGORITMO: procedimiento o conjunto de procedimientos que describen una asociación de datos lógicos destinados a la resolución de un problema. Los algoritmos permiten automatizar tareas. Secuencia codificada de instrucciones para manipulación de símbolos. En general, el proceso consiste en el acatamiento a una estructura única, ramificada, o recursiva, que se va desarrollando en serie-paralelo, con rutinas y subrutinas que se llaman al ser necesarias. Se discute si realmente el proceso intelectual de comprensión de la realidad es un proceso algorítmico o lo supera (postura, por ejemplo, de Roger Penrose). A veces se contraponen los algoritmos a las redes neuronales artificiales, pese a que el aprendizaje de estas últimas se realiza con algoritmos. La voz algoritmo deriva del apellido al-Jwarizmi, matemático árabe del siglo IX quien en sus textos presentaba este tipo de cálculos formalizados

ANDROIDE: palabra procedente del griego (andros = hombre, eidos = forma). Se aplica a las máquinas con forma humana.

ASINCRÓNICO: transmisión no relacionada con ningún tipo de sincronización temporal entre el emisor y el receptor.

AUTÓMATA: palabra que procede del griego (automatos = actuar por sí mismo). Es un mecanismo artificial que imita comportamientos de la vida real.

AUTOMÁTICA: ciencia que estudia los sistemas cibernéticos en los que ha sido introducido conscientemente algún elemento que realiza funciones de control.

AUTOMATIZACIÓN: aplicación de sistemas mecánicos, eléctricos o electrónicos, dentro de un sistema auto-gobernado, a tareas normalmente realizadas por un ser humano o que no pueden ser realizadas por él. Ejecución automática de tareas industriales, administrativas o científicas haciendo más ágil y efectivo el trabajo y ayudando al ser humano.

RAM: acrónimo de Random Access Memory (Memoria de acceso aleatorio). Memoria de acceso aleatorio que se utiliza tanto para lectura como para escritura. Es la memoria que permite al ordenador mantener el programa en curso de ejecución. Los contenidos de la memoria RAM se pueden cambiar, mientras que los del ROM no pueden cambiarse bajo el control del software. Memoria que puede ser utilizada por las aplicaciones y que desempeña otras tareas necesarias mientras la computadora funcione. Cuando se apaga la computadora, toda la información almacenada en RAM se perderá. Vea también Memoria ROM. Memoria de acceso directo, memoria principal del ordenador, donde residen los programas y datos.

ROBOT: manipulador mecánico, reprogramable y de uso general. Un dispositivo mecánico que realiza acciones basadas en movimientos. Las acciones más comunes de un robot incluyen su movimiento autónomo (caminar sin recibir instrucciones no contenidas en el mismo robot), soldar y reparar accidentes nucleares, otras manipulaciones demasiado precisas, pesadas, repetitivas (aburridas) o riesgosas para el ser humano.

ROBÓTICA: diseño, fabricación y utilización de máquinas automáticas programables con el fin de realizar tareas repetitivas como el ensamble de automóviles, aparatos, etc. y otras actividades. Es una de las tecnologías más cercanas a lo que es el núcleo de la bioingeniería del conocimiento. Se trata de una tecnología mecatrónica (mecánica más electrónica más cibernética) que diseña máquinas que realizan acciones que normalmente se hacen manualmente, esto es, con la mano y el brazo. El aprendizaje de los movimientos de un robot que recoja un vaso con líquido y que lo desplace sin derrames es una tarea monumental.

ROBUSTO: fuerte, vigoroso, bien fabricado, difícil de romper. En control, poco sensible a errores o incertidumbres de modelado. En la actividad computacional, tanto una máquina como un algoritmo se consideran robustos si son capaces de actuar y trascender a pesar de dificultades que accidentarían a máquinas o algoritmos más simples o menos flexibles. La tolerancia a las fallas o fallos se considera sinónimo de robustez.

ROM: acrónimo de Read Only Memory (Memoria de sólo lectura). Memoria que puede leerse pero no modificarse. Conserva instrucciones y programas indelebles, que no se pierden al apagar el ordenador.

INTRODUCCIÓN

La robótica industrial es un campo que compete a los Ingenieros Electrónicos y que se encuentra en pleno desarrollo debido a la rápida impulsión tecnológica de las últimas décadas. La definición de qué es un robot varía según la institución científica que se refiera a ella, por ello se suele asociar este concepto con la maquinaria automatizada. Debido a estas situaciones se hace cada día más imprescindible el aprendizaje de estas tecnologías dentro de las universidades, lo cual implica generalmente, la implementación de robots de coste muy elevado y de manejo, mantenimiento y reparación muy compleja y costosa.

El objetivo de este proyecto es presentar como solución alternativa, la construcción completa de un robot de aplicación didáctica a partir de elementos de muy bajo coste y fácil adquisición en el mercado, donde además este robot contará con características similares a los robots de aplicación industrial. Las aplicaciones de este autómeta buscan solventar las necesidades de formación, investigación y evaluación de aplicaciones presentes dentro de los laboratorios de ingeniería electrónica de la Universidad de Nariño.

1. TEMA

1.1.TÍTULO

**RADAC-ROBOT DE APLICACIÓN DIDÁCTICA PARA EL APRENDIZAJE
CONCEPTUAL DE LA AUTOMATIZACIÓN Y CONTROL EN LOS
LABORATORIOS DE INGENIERÍA ELECTRÓNICA DE LA UNIVERSIDAD
DE NARIÑO**

1.2.MODALIDAD

Trabajo de investigación.

1.3.LINEA DE INVESTIGACIÓN

Automatización y control

1.4 ALCANCE Y DELIMITACIÓN

El proyecto terminado consiste de la estructura elemental de un robot servocontrolado estacionario de tipo cilíndrico de tres grados de libertad y herramienta gripper, programable por computador, por medio de una consola que posea interfaces de potencia del robot.

Este tipo de autómatas tiene una finalidad didáctica dentro del marco del aprendizaje de los conceptos referentes a la automatización, el control y la robótica, previstos en la línea de automatización y control del programa de ingeniería electrónica de la Universidad de Nariño.

2. DEFINICIÓN DEL PROBLEMA

2.1. PLANTEAMIENTO DEL PROBLEMA

Pese a las múltiples herramientas de aprendizaje de la automatización y control que actualmente se encuentran en los laboratorios de ingeniería electrónica de la Universidad de Nariño, existe aún la necesidad de establecer nuevos elementos didácticos adicionales, como por ejemplo los controladores lógicos programables PLC's, elementos de sensórica, sistemas neumáticos, hidráulicos, electromecánicos, híbridos, de lógica cableada y robots de aplicación didáctica. La carencia de dichos dispositivos puede ser considerado como una seria limitante en el proceso de enseñanza de la electrónica aplicada, especialmente en el área de la robótica, cuyo estudio se ha venido diversificando y potenciando en el marco de las universidades y centros de investigación mundial.

La anterior situación conlleva al planteamiento de soluciones eficientes y eficaces dentro de los parámetros de precisión, costos, flexibilidad y fiabilidad del robot. Una de estas posibles soluciones se refiere al diseño y fabricación de uno de estos dispositivos a través de un esfuerzo realizado dentro del programa de ingeniería electrónica de la Universidad de Nariño.

2.2 FORMULACIÓN DEL PROBLEMA

¿Es posible realizar el diseño y construcción de un robot fijo, de aplicación didáctica, de bajo costo pero con características de precisión, fiabilidad y flexibilidad necesarias para la ejecución de prácticas dentro la línea de profundización de automatización y control en los laboratorios de ingeniería de la Universidad de Nariño?

3. OBJETIVOS

3.1. OBJETIVO GENERAL

Diseñar y construir un robot servocontrolado estacionario de tipo cilíndrico de tres grados de libertad, programable por computador, los que en conjunto permitirán realizar prácticas dentro la línea de profundización de automatización y control, en los laboratorios de ingeniería de la Universidad de Nariño, con características de precisión y fiabilidad aceptables.

3.2. OBJETIVOS ESPECÍFICOS

- Diseñar y construir la estructura mecánica, transmisión mecánica, sistemas de accionamiento, sistemas de control y elementos terminales del robot mediante la evaluación de alternativas susceptibles de implementarse de acuerdo con los requerimientos de bajo costo, precisión, fiabilidad, flexibilidad y adaptabilidad, en los procesos en los que el robot se verá involucrado.
- Diseñar y elaborar un software de programación, emulación y simulación de manipuladores robóticos.
- Diseñar y construir una interfaz electrónica que interactúe con el sistema de control, sensores internos y actuadores del robot, y que sea susceptible de reemplazar a dicho sistema de control basado en microcontroladores, por otros sistemas alternativos (control por P.C, microprocesadores, controles numéricos y PLC's.).

4. JUSTIFICACIÓN

La importancia de la robótica didáctica radica en la necesidad de implementar, dentro del marco de la automatización y el control, una disciplina, que tiene por objeto la generación de ambientes de aprendizaje basados fundamentalmente en la actividad de los estudiantes. Es decir, se pueden concebir, desarrollar y poner en práctica diferentes proyectos que permiten resolver problemas y facilitan al mismo tiempo, ciertos aprendizajes los cuales son representaciones en pequeña escala de la realidad tecnológica circundante, tanto en Nariño como en la industria nacional y extranjera.

La puesta en marcha de un proyecto de construcción de un robot didáctico requiere del conocimiento de diversas áreas, las cuales pueden resumirse en la aplicación de sistemas mecánicos y electrónicos integrados con sistemas computarizados para operación y control de la ejecución. Estos elementos técnicos y conceptuales se presentan actualmente o son fácilmente alcanzables dentro de los laboratorios de ingeniería electrónica de la Universidad de Nariño, haciendo posible la realización de este tipo de proyectos, así como un primer acercamiento a la robótica didáctica.

Por último, debe señalarse que el diseño y construcción de un robot fijo de aplicación didáctica, de bajo costo con características de precisión, fiabilidad y flexibilidad necesarias para la ejecución de prácticas dentro la línea de profundización de automatización y control conllevará a la ejecución de procesos de automatización más exactos, fiables y eficientes. Las consecuencias directas de estas ventajas redundarán en un ahorro económico sustancial en comparación con la adquisición de estos autómatas en el mercado, siendo también una contribución a los desarrollos científicos, investigativos y educativos que se dan a lugar en la Universidad de Nariño.

5. MARCO REFERENCIAL

La robótica es un concepto de dominio público. La mayor parte de la gente tiene una idea de lo que es la robótica, sabe sus aplicaciones y el potencial que tiene; sin embargo, no conocen el origen de la palabra robot, ni tampoco tienen idea del origen de las aplicaciones útiles de la robótica como ciencia.

La robótica como hoy en día se conoce, tiene sus orígenes hace miles de años. Desde el principio de los tiempos, el hombre se ha empeñado en crear máquinas herramientas que realicen sus tareas repetitivas, tareas pesadas o difíciles de realizar por un ser humano. Los primeros autómatas utilizaban, principalmente, la fuerza bruta para poder realizar sus movimientos. A las primeras máquinas herramientas que ayudaron al hombre a facilitar su trabajo no se les daba el nombre de autómatas, sino más bien se les reconocía como artefactos o simples máquinas.

5.1. AUTOMATIZACIÓN Y ROBÓTICA

La historia de la automatización industrial está caracterizada por períodos de constantes innovaciones tecnológicas. En términos generales, la automatización y la robótica son disciplinas surgidas en diferentes épocas. La robótica nace en décadas recientes para complementarse con la automatización, aportándole como elemento innovador cierto grado de inteligencia. En el contexto industrial, la automatización es como una tecnología que está relacionada con el empleo de sistemas mecánicos, electrónicos y basados en la informática en la operación y control de la producción. Este concepto, para ser actualizado, debe incluir el uso de robots.

El robot industrial forma parte del progresivo desarrollo de la automatización industrial, favorecido notablemente por el avance de las técnicas de control por computadora, y contribuye de manera decisiva a la automatización en los procesos de fabricación de series de mediana y pequeña escala.

Se entiende por Robot Industrial a un dispositivo de maniobra destinado a ser utilizado en la industria y dotado de uno o varios brazos, fácilmente programable para cumplir operaciones diversas con varios grados de libertad y destinado a sustituir la actividad física del hombre en las tareas repetitivas, monótonas,

desagradables o peligrosas. El RIA (Robot Institute of America) define al Robot como "Un manipulador multifuncional reprogramable, diseñado para mover materiales, partes, herramientas o dispositivos especializados a través de movimientos variables programados para el desempeño de una variedad de labores"¹.

En la actualidad el uso de los robots industriales está concentrado en operaciones muy simples, como tareas repetitivas que no requieren tanta precisión. Los análisis de mercado en cuanto a fabricación predicen que en ésta década y en las posteriores los robots industriales incrementaran su campo de aplicación, esto debido a los avances tecnológicos en sensórica, los cuales permitirán tareas mas sofisticadas como el ensamble de materiales.

Como se ha observado la automatización y la robótica son dos disciplinas estrechamente relacionadas. En un contexto industrial se puede definir la automatización como una tecnología que está relacionada con el empleo de sistemas mecánicos-eléctricos basados en computadoras para la operación y control de la producción. En consecuencia la robótica es una forma de automatización industrial.

Según Dueñas², hay tres clases muy amplias de automatización industrial: automatización fija, automatización programable, y automatización flexible.

5.1.1. La automatización fija. Se utiliza cuando el volumen de producción es muy alto, y por tanto se puede justificar económicamente el alto costo del diseño de equipo especializado para procesar el producto, con un rendimiento alto y tasas de producción elevadas. El inconveniente de la automatización fija es su ciclo de vida que va de acuerdo a la vigencia del producto en el mercado.

5.1.2. La automatización programable. Se emplea cuando el volumen de producción es relativamente bajo y hay una diversidad de producción a obtener. En este caso el equipo de producción es diseñado para adaptarse a la variaciones de conción del producto; ésta adaptación se realiza por medio de un programa (Software).

1 GROOVER, Mikell. Robótica industrial- tecnología programación y aplicaciones. Madrid. Mc Graw-Hill. 1989. p. 6

2 DUEÑAS, Francisco. La Robótica. En [http:// www.monografias.com/robótica](http://www.monografias.com/robótica). 2005

5.1.3. La automatización flexible. Es más adecuada para un rango de producción medio. Estos sistemas flexibles poseen características de la automatización fija y de la automatización programada.

Los sistemas flexibles suelen estar constituidos por una serie de estaciones de trabajo interconectadas entre si por sistemas de almacenamiento y manipulación de materiales, controlados en su conjunto por una computadora.

De los tres tipos de automatización, la robótica coincide más estrechamente con la automatización programable.

El control numérico y la telequerica son dos tecnologías importantes en el desarrollo de la robótica. El control numérico (NC) se desarrolló para máquinas herramienta a finales de los años 40 y principios de los 50's. Como su nombre lo indica, el control numérico implica el control de acciones de una máquina-herramienta por medio de números. Está basado en el trabajo original de Jhon Parsons, que concibió el empleo de tarjetas perforadas, que contienen datos de posiciones, para controlar los ejes de una máquina-herramienta.

A veces denominado teleoperador, el operador remoto es un dispositivo mecánico que traduce los movimientos del operador humano en movimientos correspondientes en una posición remota. A Goertz se le acredita el desarrollo de la telequérica. En 1948 construyó un mecanismo manipulador bilateral maestro-esclavo en el Argonne National Laboratory. El empleo más frecuente de los teleoperadores se encuentra en la manipulación de sustancias radiactivas, o peligrosas para el ser humano.

La combinación del control numérico y la telequerica es la base que constituye al robot modelo. Hay dos individuos que merecen el reconocimiento de la confluencia de éstas dos tecnologías y el personal que podía ofrecer en las aplicaciones industriales. El primero fue un inventor británico llamado Cyril Walter Kenward, que solicitó una patente británica para un dispositivo robótico en marzo de 1954.

La segunda persona citada es George C. Devol, inventor americano, al que deben atribuirse dos invenciones que llevaron al desarrollo de los robots cuyos principios se aplican hasta la actualidad. La primera invención consistía en un dispositivo para grabar magnéticamente señales eléctricas y reproducirlas para controlar una máquina. La segunda invención se denominaba Transferencia de Artículos Programada.

Un robot industrial es una máquina programable de uso general que tiene algunas características antropomórficas o "humanoides". Las características humanoides más típicas de los robots actuales es la de sus brazos móviles, los que se desplazarán por medio de secuencias de movimientos que son programados para la ejecución de tareas de utilidad.

Se espera en un futuro no muy lejano que la tecnología en robótica se desplace en una dirección que sea capaz de proporcionar a éstas máquinas capacidades más similares a las humanas.

5.2. MORFOLOGÍA DEL ROBOT Y SUS PERIFÉRICOS

5.2.1. Estructura Mecánica de un Robot.

Un robot está formado por los siguientes elementos: estructura mecánica, transmisiones mecánicas, sistema de accionamiento, sistema sensorial, sistema de control y elementos terminales. Aunque los elementos empleados en los robots no son exclusivos de estos (maquinas herramientas y otras muchas maquinas emplean tecnologías semejantes), las altas prestaciones que se exigen a los robots han motivado que en ellos se empleen elementos con características específicas.

Mecánicamente, un robot está formado por una serie de elementos o eslabones unidos mediante articulaciones que permiten un movimiento relativo entre cada dos eslabones consecutivos. La constitución física de la mayor parte de los robots industriales guarda cierta similitud con la anatomía del brazo humano, por lo que en ocasiones, para hacer referencia a los distintos elementos que componen el robot, se usan términos como cuerpo, brazo, codo y muñeca.

Sistemas de Robots básicos. Los componentes básicos de un robot son:

La estructura - la estructura mecánica (los eslabones, base, etc). Esto exige mucha masa, para proporcionar la rigidez estructural para asegurar la exactitud mínima bajo las cargas útiles variadas.

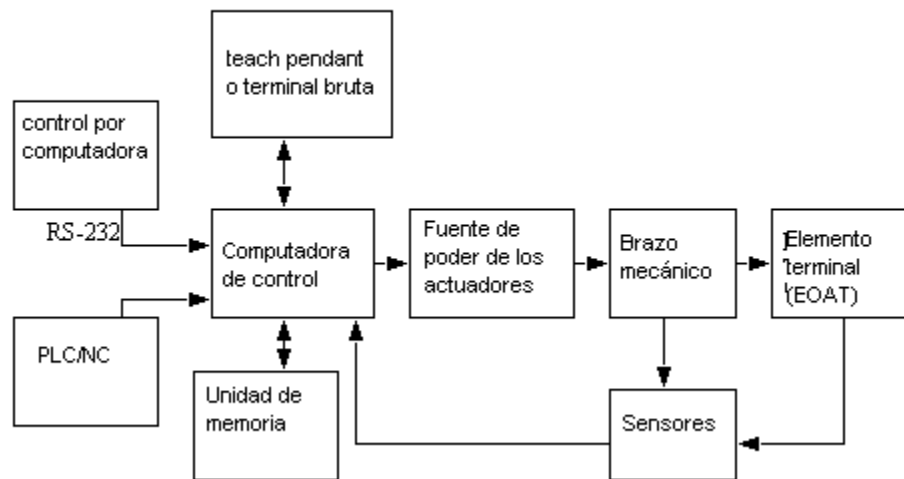
Actuadores - Los motores, los cilindros, etc., las juntas del robot. Esto también podría incluir los mecanismos para una transmisión, etc.,

Control a la Computadora - Esta computadora une con el usuario, y a su vez los mandos las juntas del robot.

El extremo de Brazo que labora con herramienta (EOAT) - La programación que proporciona el usuario se diseña para las tareas específicas.

Teach pendant- Un método popular para programar el robot. consiste en que una mano pequeña contiene un dispositivo que puede dirigir movimiento del robot, los puntos de registro en las sucesiones de movimiento, y comienza la repetición de sucesiones (ver figura 1).

Figura 1. Componentes básicos de un robot fijo



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.

Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior, se denomina grado de libertad³.

El número de grados de libertad del robot viene dado por la suma de los grados de libertad de las articulaciones que lo componen. Puesto que, como se ha indicado, las articulaciones empleadas son únicamente las de rotación y prismática con un

³ Estructura Mecánica de un Robot. En <http://proton.ucting.udg.mx/materias/robotica>

solo con grado de libertad cada una, el número de grados de libertad del robot suele coincidir con el número de articulaciones de que se compone.

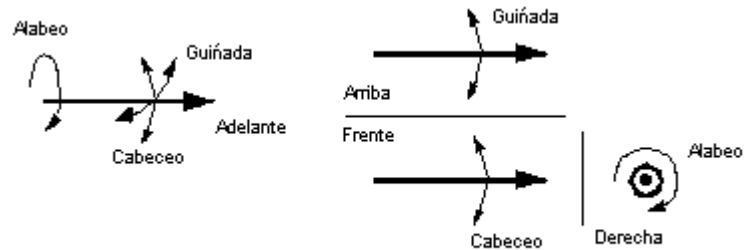
El empleo de diferentes combinaciones de articulaciones en un robot, da lugar a diferentes configuraciones, con características a tener en cuenta tanto en el diseño y construcción del robot como en su aplicación. Las combinaciones más frecuentes son las que poseen tres articulaciones, y son las más importantes a la hora de posicionar su extremo en un punto en el espacio.

Puesto que para posicionar y orientar un cuerpo de cualquier manera en el espacio son necesarios seis parámetros, tres para definir la posición y tres para la orientación, si se pretende que un robot posicione y oriente su extremo (y con el la pieza o herramienta manipulada) de cualquier modo en el espacio, se precisara al menos seis grados de libertad. En la práctica, a pesar de ser necesarios los seis grados de libertad comentados para tener total libertad en el posicionado y orientación del extremo del robot, muchos robots industriales cuentan con solo cuatro o cinco grados de libertad, por ser estos suficientes para llevar a cabo las tareas que se encomiendan. Existen también casos opuestos, en los que se precisan más de seis grados de libertad para que el robot pueda tener acceso a todos los puntos de su entorno. Así, si se trabaja en un entorno con obstáculos, el dotar al robot de grados de libertad adicionales le permitirá acceder a posiciones y orientaciones de su extremo a las que, como consecuencia de los obstáculos, no hubieran llegado con seis grados de libertad. Otra situación frecuente es dotar al robot de un grado de libertad adicional que le permita desplazarse a lo largo de un carril aumentando así el volumen de su espacio al que puede acceder. Cuando el número de grados de libertad del robot es mayor que los necesarios para realizar una determinada tarea se dicen que el robot es redundante.

Condiciones básicas. Los eslabones y Junturas - los Eslabones son los miembros estructurales sólidos de un robot, y las juntas son los acoplamientos móviles entre ellos.

El grado de Libertad (gdl) - Cada junta en el robot introduce un grado de libertad. Cada gdl pueden ser un deslizador, el tipo rotatorio, u otro actuador. Los robots tienen 5 o 6 grados de libertad típicamente. 3 de los grados de libertad permiten el posicionamiento en el espacio 3D, mientras el otro se usan 2 o 3 para la orientación del efector del extremo. 6 grados de libertad es bastante para permitir al robot alcanzar todas las posiciones y orientaciones en 3D. 5 gdl requiere una restricción a 2D espacio, el resto limita las orientaciones. Normalmente se usan 5 gdl por ocuparse de herramientas como los soldadores de arco. La orientación Eslabón - Básicamente, si la herramienta se sostiene a una posición fija, la orientación determina qué dirección puede apuntarse. El rollo, diapasón y guiñada (ver Figura 2) son los elementos de la orientación comunes usadas.

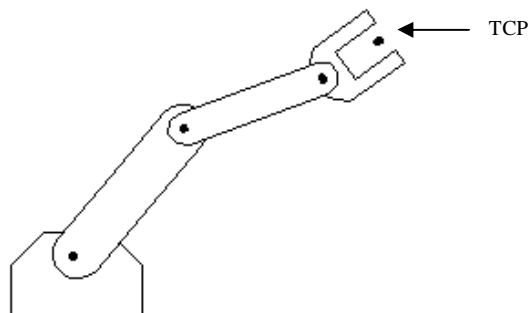
Figura 2. Tipos de posicionamiento



Los elementos de la posición - La herramienta, sin tener en cuenta la orientación, puede moverse a varias posiciones en el espacio. Se satisfacen las varias geometrías del robot a las geometrías de trabajo diferentes.

El Punto de Centro de herramienta (TCP) - El punto de centro de herramienta se localiza en el robot, la herramienta. Típicamente el TCP se usa al referirse a la posición de los robots, así como el punto focal de la herramienta. (Por ejemplo el TCP podría estar en la punta de una antorcha de la soldadura) El TCP puede especificarse en el cartesiano, cilíndrico, esférico, etc., coordenadas que dependen del robot (ver Figura 3).

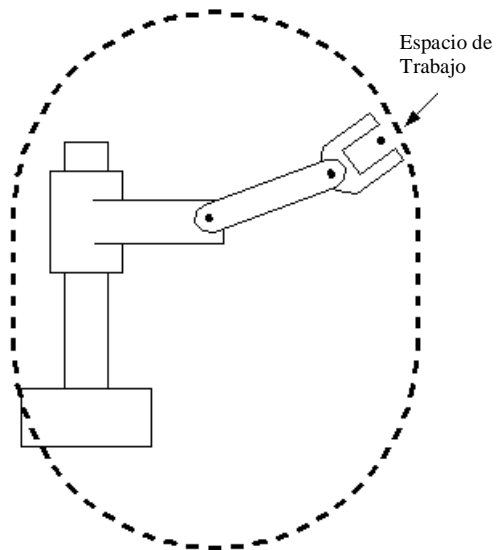
Figura 3. Punto focal de un robot



Tomado de fundamentos de robótica

El espacio de trabajo - El robot tiende a tener una geometría fija, y limitada. El espacio de trabajo es el límite de posiciones en el espacio que el robot puede alcanzar. Para un robot cartesiano los espacios de trabajo podrían ser un cuadrado, para los robots más sofisticados los espacios podrían ser de una forma esférica (ver Figura 4).

Figura 4. Espacio de trabajo del robot



Tomado de fundamentos de robótica

La velocidad - se refiere a la velocidad máxima que es logvable por el TCP, o por las juntas individuales. Este número no es exacto en la mayoría de los robots, y variará encima del espacio de trabajo como la geometría del robot cambia (y de los efectos dinámicos). El número reflejará a menudo la velocidad más segura máxima posible.

La carga útil - La carga útil indica la masa máxima que el robot puede levantar antes de cualquier fracaso de los robots, o pérdida dramática de exactitud. Es posible exceder la carga útil máxima, y el robot todavía tiene que operar, pero esto no se aconseja. Cuando el robot está acelerando rápidamente, la carga útil debe ser menos que la masa máxima. Esto es afectado por la habilidad de agarrar la parte firmemente, así como la estructura del robot, y el actuador. El extremo de brazo al laborar con herramienta debe ser considerado parte de la carga útil.

- La carga útil: La carga útil siempre se especifica como un valor máximo, esto puede estar antes del fracaso, o más normalmente, antes de la pérdida de la actuación seria.
- Las consideraciones Estáticas:
 - La gravedad que efectúa cause desviación descendente del brazo y sistemas de apoyo
 - Manejo a menudo de cubiertas, las cuales pueden traer cantidades notables de lentitud (la repercusión negativa) esa causa que posiciona los errores

- El trabajo de la junta - cuando se usan miembros rotatorios largos en un sistema de esfuerzos se tuercen bajo la carga
- Los efectos térmicos - la temperatura modifica las dimensiones en el manipulador.

Repetibilidad - El mecanismo del robot tendrá alguna variación natural en él. Esto significa que cuando el robot se devuelve al mismo punto repetidamente, no siempre detendrá a la misma posición. Se considera que Repetibilidad es ± 3 veces la desviación normal de la posición, o donde 99.5% de toda la caída de dimensiones de repetibilidad. Esta figura variará encima del espacio, especialmente cerca de los límites del espacio de trabajo, pero los fabricantes darán un solo valor en las especificaciones.

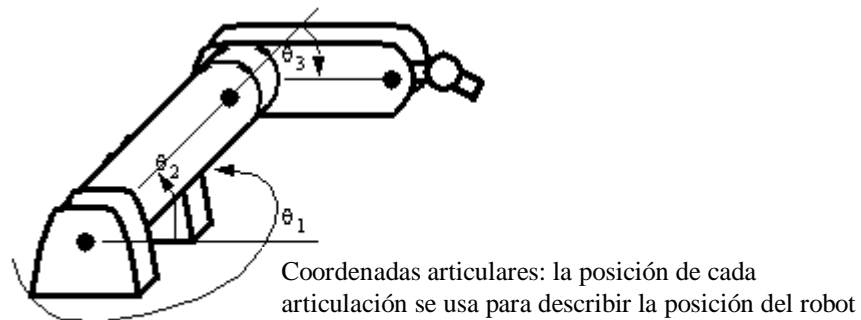
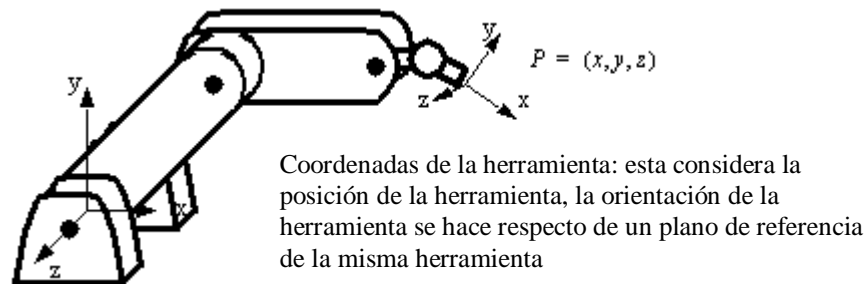
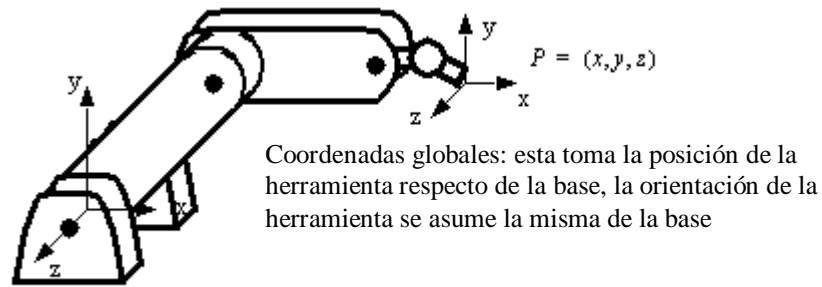
La exactitud – se refiere la capacidad de un robot para situar el extremo de su muñeca en un punto deseado dentro del espacio de trabajo.

Tiempo de establecimiento - Durante un movimiento, el robot se mueve rápidamente, pero a medida que el robot se acerca a la posición final se reduce la velocidad. El tiempo de establecimiento es el tiempo requerido para el robot, para estar dentro de una distancia dada de la última posición.

Control de la Resolución - Éste es el cambio más pequeño que puede medirse por los sensores de la articulación, a causa del actuador. Si una junta rotatoria tiene un encoder que mide cada 0.01 grado de rotación, y un motor de servo de paso directo se usa para manejar la junta, con una resolución de 0.5 grados, entonces la resolución del mando es aproximadamente 0.5 grados (el peor caso puede ser $0.5+0.01$).

Las coordenadas - El robot se puede mover, por consiguiente es necesario definir las posiciones. Nótese que las coordenadas son una combinación de la posición del origen y orientación de los eslabones (ver Figura 5).

Figura 5. Coordenadas de un robot



Tomado de fundamentos de robótica

5.2.2. Sensores. se dividen básicamente en dos tipos:

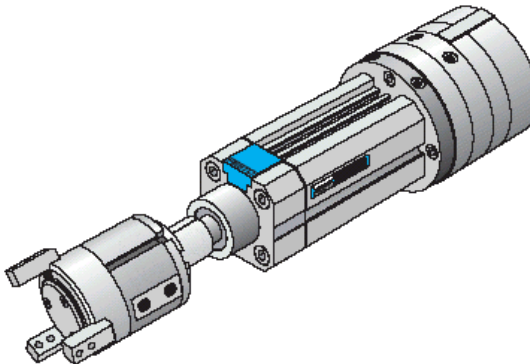
Sensores internos o propioceptivos: Miden el estado interno del robot.

Sensores externos o exteroceptivos: Miden el estado del entorno e interacciones con el robot.

5.2.3. Actuadores. Los actuadores son las partes del robot que se encargan de transformar algún tipo inicial de energía en variaciones de energía cinética, es decir, en movimientos o paradas del robot.

5.2.4. Efecto final (pinzas y ejes de manos gripper). Las pinzas son dispositivos de sujeción; esta es su principal función. Para obtener un determinado efecto, hay que disponer las pinzas de forma que puedan moverse en las tres dimensiones. Esto no es especialmente complicado, p. ej. En una aplicación con un robot de 6 ejes, ya que el propio robot dispone de la suficiente movilidad. En aplicaciones donde los costes sean críticos y no se exijan grandes velocidades, vale la pena considerar los ejes de manos, disponibles a menudo como módulos flexibles que pueden reemplazar a un robot. Esto genera soluciones que pueden instalarse rápidamente y a costes razonables. Dependiendo de la aplicación, un eje de mano puede ser interesante para los siguientes movimientos. Rotación $> 360^\circ$, giro parcial $< 360^\circ$, movimientos de fuerza (generalmente con carreras cortas) y movimientos de atornillado, particularmente para inserción de tornillos. Sin embargo, la aplicación más típica es el giro parcial, por lo que los fabricantes de pinzas casi siempre ofrecen unidades compatibles con el giro. La Figura 6 muestra un módulo de dos ejes que puede girar entre 0 y 270° y permite una carrera lineal de empuje de hasta 100 mm .

Figura 6. Pinza de tres puntos combinada con una unidad lineal/giratoria



Tomado de www.robotica2.galeon.com/index.html

Una pinza puede tener un máximo de 6 grados de libertad, expresados como 3 movimientos lineales en las tres dimensiones de los ejes x , y , z , y 3 movimientos giratorios α_1 , α_2 , α_3 sobre los ejes x , y , z . A propósito, las máquinas manipuladoras pueden tener más de 6 grados de libertad. Entonces se habla de grados de libertad mecánica o de libertad de movimientos.

Los movimientos de empuje (Figura 7) se describen como sigue:

- 1 Vertical, arriba/abajo
- 2 Frontal, avance/retroceso
- 3 Lateral, izquierdo/derecho

En donde los movimientos giratorios, siguiendo la terminología de la aviación, se designan como sigue:

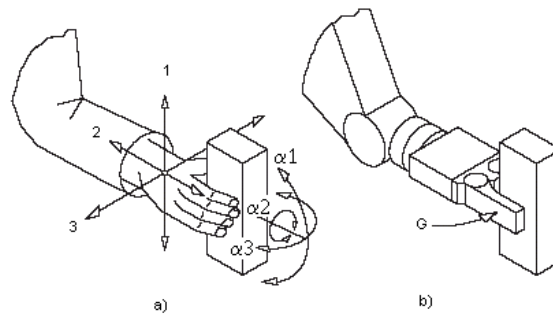
- α_1 Cabeceo, inclinación
- α_2 Balanceo, torsión
- α_3 Guiñada, giro.

El movimiento de los dedos de la pinza no se considera un grado de libertad, ya que este movimiento no influye en la trayectoria de la pinza.

Figura 7. La mano humana puede ejecutar movimientos con 6 grados de libertad

a) Biológicos

b) Técnicos



Tomado de Robótica industrial

5.3. PROGRAMACIÓN Y CONTROL DE ROBOTS

5.3.1. Unidades de control. Las unidades de control se encargan de decidir (controlar) las acciones del robot a partir de ordenes externas y las señales provenientes de los sensores. Sus señales de salida se dirigen a los actuadores.

A continuación se dan algunas consideraciones sobre estos elementos del robot:

Puede existir una única unidad central, o una distribución de unidades más o menos independientes. La definición o separación de cada una puede ser función de su situación física o de las tareas a su cargo.

Su situación física puede ser sobre el esqueleto del robot (robots móviles), o externa a éste (manipuladores).

Generalmente estarán formadas por ordenadores (armarios, PCs) y/o placas microcontroladoras.

En el caso de utilizar técnicas de inteligencia artificial, se consideran como la parte que "almacena" la inteligencia del robot, su cerebro, sistema nervioso, etc.

En el caso de dependencia entre ellas, la distribución suele ser jerárquica. La distribución jerárquica es tanto espacial como temporal, debido a una división de tareas en subtareas. La capacidad de decisión del conjunto definirá su autonomía de acción.

Grados de libertad y ligaduras. Los sistemas dinámicos se caracterizan mediante un conjunto de variables (x_i) denominadas coordenadas de configuración, que representan en cada instante la configuración del sistema. La cinemática se encarga del estudio de estas variables, así como de su variación el tiempo, es decir, las velocidades. La dinámica se encarga de estudiar el cambio de estas variables y velocidades debido a la acción de fuerzas externas o internas que actúan sobre el sistema.

La existencia de ligaduras o restricciones entre las coordenadas de configuración del tipo: $f(x_i, t) = 0$ nos permite reducir el número de coordenadas de configuración necesarias a un conjunto mínimo de variables linealmente independientes, denominadas coordenadas generalizadas, y que se representará por q_i . Estas coordenadas representan los grados de libertad reales del sistema.

6. MARCO REFERENCIAL

6.1. MARCO REFERENCIAL DE LA ENTIDAD

Por Decreto No. 049 de 7 de noviembre de 1904, se fundó la Universidad de Nariño en la capital del departamento para impartir Instrucción Secundaria y Profesional. Según esta norma, pertenecían a la Institución las Facultades de Pasto y Barbacoas y otras que pudiesen surgir como oficiales en el resto del Departamento. Se estableció la autonomía de la Universidad, se declaró la validez de los títulos, certificados y grados. Igualmente, se señaló el pensum que debía adoptarse en cada una de las Facultades y se inventariaron los bienes y rentas del establecimiento. El 20 de noviembre de 1905, quedó la Universidad establecida con las facultades de Derecho y Ciencias Políticas, Matemáticas e Ingeniería y la Clase de Comercio. En la década actual, la presión por el cambio y la reestructuración de los procesos académicos y administrativos se hicieron más evidentes y posibilitaron: la conformación de nuevas facultades; la diversificación de programas; la regionalización mediante el establecimiento de sedes en diferentes municipios de Nariño y Putumayo; la ampliación de la cobertura educativa; la vinculación de la Universidad mediante convenios, con instituciones nacionales e internacionales y la inserción en las redes mundiales del conocimiento. Al llegar la Universidad a sus 105 años de existencia, podemos afirmar que el "Alma Mater" a través del mejoramiento permanente, impulsa con tenacidad los campos de la investigación, la docencia y proyección social, acordes con los retos que la modernidad le impone⁴.

6.2. MARCO REFERENCIAL DEL PROYECTO

En la Universidad de Nariño y particularmente la línea de investigación de automatización y control existe en la actualidad, documentación y antecedentes de proyectos realizados con el mismo objetivo o similar.

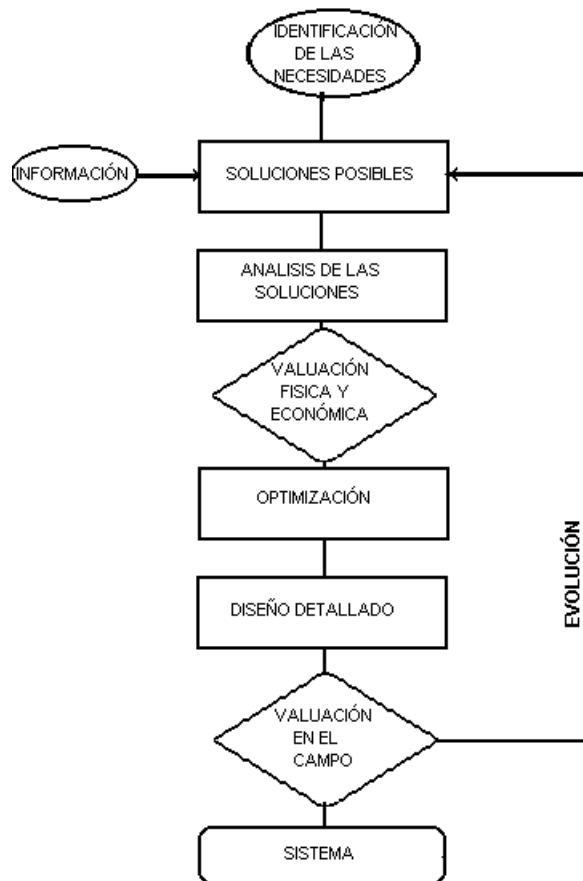
⁴ www.Udenar.edu.co/nu200.htm

7. METODOLOGÍA

7.1. DESCRIPCIÓN METODOLÓGICA

La metodología implementada utilizó las estrategias, métodos y medios definidos en el desarrollo de proyectos de ingeniería propuesto por Corzo⁵. Estos procedimientos se describen a continuación:

Figura 8. Proceso de diseño



⁵ CORZO, Miguel. Introducción a los proyectos de ingeniería. México D.F.. Limusa. 1982. p. 44

Por tanto, esta metodología se refirió a la evaluación y selección de los posibles modelos de solución. Finalmente, estas actividades se muestran a continuación en la siguiente tabla de secuencias y se describen en el marco del uso de los materiales y herramientas metodológicas a emplearse:

Tabla 1. Actividades planificadas

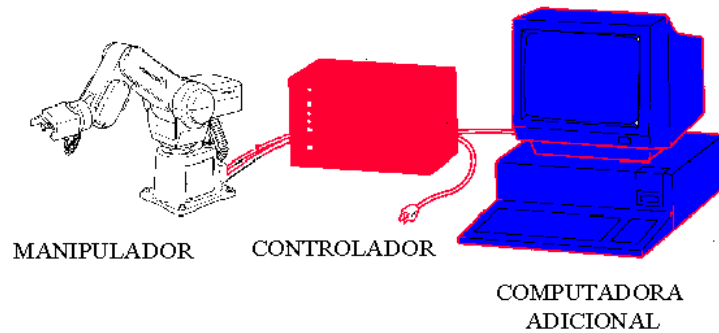
Actividades lógicas anteriores	Actividades planificadas			Actividades lógicas posteriores
	Orden	Detalle	Duración en semanas	
--	A	Estudio Explorativo	2	B
A	B	Análisis del Tema	4	C
B	C	Definición del Problema	2	D
A, B, C	D	Investigación Procesos Existentes	3	E
C, D	E	Diseño Conceptual de Prototipos	8	F, G
C, E	F	Determinación del Prototipo Final	2	H
F	G	Diseño Preliminar del Prototipo Escogido	5	I
G	H	Diseño Detallado del Prototipo Escogido	15	J
H	I	Desarrollo del Prototipo	20	K
I	J	Seguimiento Prototipo	3	L
I, J	K	Mejoramiento del Prototipo	1	M
K	L	Diseño Final	2	N
L	M	Comunicación y Presentación Final del Diseño	1	--
--	N	Documentación	16	--

7.2. MATERIALES Y HERRAMIENTAS METODOLÓGICAS

De acuerdo con la metodología implementado, la instrumentación del Robot prototipo consistió de:

- Manipulador
- Circuitos de interfaz y control (controlador programador).
- Computador de programación y visualización.

Figura 9. Robot didáctico

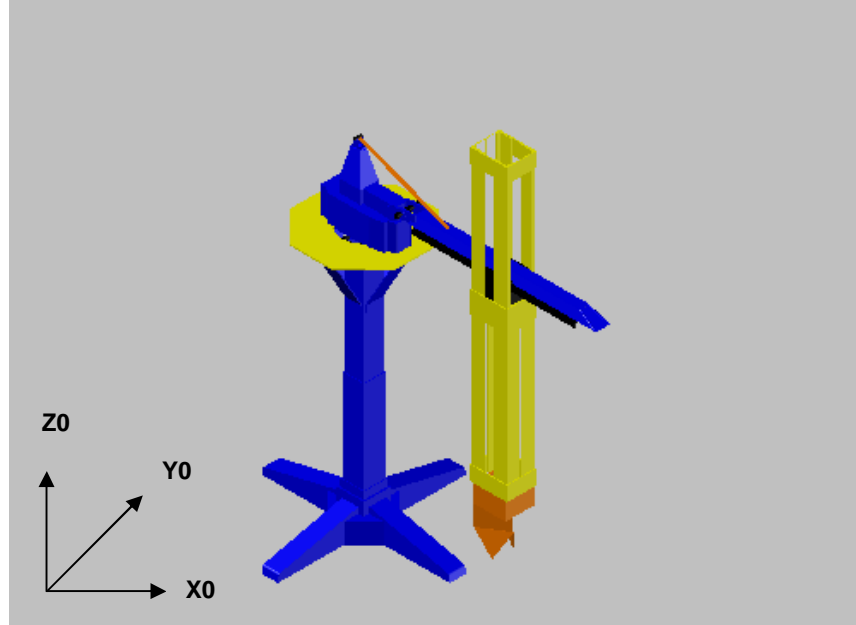


Tomado de <http://robotica2.galeon.com/index.html>, Op. cit.

7.2.1. El manipulador. El componente principal lo constituye el manipulador, el cual consta de varias articulaciones y sus elementos. En este proyecto se parte de una estructura (ver Figura 10), posee las siguientes características:

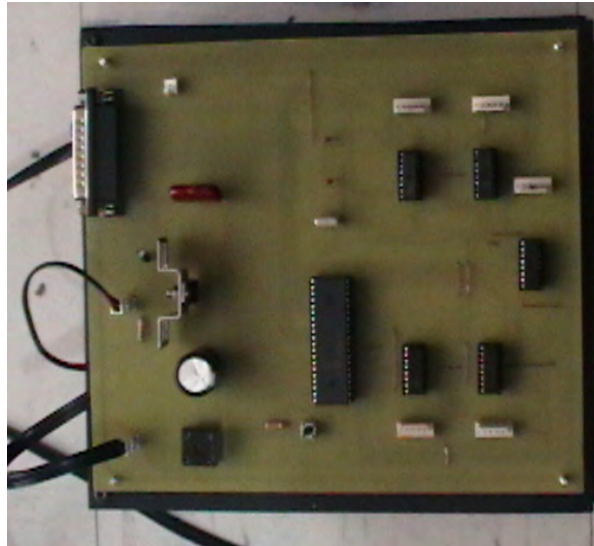
- Altura total: 45 cm, longitud del brazo: 24 cm, longitud efectiva: 16 cm
- Sistemas de coordenadas: Cilíndricas
- Tres grados de libertad (giro azimutal de 360°, elevación vertical de 10 cm y desplazamiento horizontal de 10 cm) y un gripper de dos posiciones (abierto-cerrado).
- Capacidad de carga: baja ($<0,1$ kg) sólo para aplicaciones didácticas, demostrativas, de inspección mediante sensores y elementos de poco peso. Posee como actuadores, motores de pasos.
- Gripper: tipo pinza, de tres dedos dispuestos a 120 ° entre sí para un agarre completo, de movimiento por pivotaje. Además este efector cuenta con el peso y la robustez adecuadas.
- Repetibilidad: alta pero susceptible a cambios debido a la inclusión de estructuras adicionales que alteren la inercia; posee enclavamiento en todos sus desplazamientos (las articulaciones no se mueven una vez retirada la energía de los actuadores).

Figura 10. Manipulador



7.2.2. Circuitos de interfaz del robot. Estos emplean las señales de entrada y salida se que obtienen de la tarjeta electrónica instalada en el controlador del robot, y de los sensores internos y actuadores situados en el manipulador, las cuales además le permiten tener comunicación con otras máquinas-herramientas. En las celdas de manufactura, por ejemplo, se utilizan estas tarjetas para comunicar al robot. Estas tarjetas se componen principalmente de conmutadores electrónicos, los cuales mandan señales eléctricas que después son interpretadas en un programa de control, estas señales nos permiten controlar cuando debe entrar el robot a cargar una pieza a la máquina, cuando debe empezar a funcionar la máquina o el robot, etc. (ver Figura 11 y Anexo F).

Figura 11. Circuitos de interfaz



7.2.3. Estación de trabajo. Esta consiste de dos controladores definidos por la estación de trabajo virtual como controlador principal, y un controlador secundario constituido en sí por la estación física, todo esto programado por P.C.

Circuitos de interfaz y control (controlador programador). El controlador del robot, como su nombre lo indica, es el que controla cada uno de los movimientos del manipulador y guarda sus posiciones. El controlador recibe y envía señales a otras máquinas-herramientas (por medio de señales de entrada / salida) y almacena programas. Este controlador consistirá en los soportes y disposiciones físicas, eléctricas y electrónicas del robot.

Computador de programación y visualización. Esta estación hace uso de un P.C, el cual posee un software que a su vez es una interfaz virtual amable con el usuario, la cual permite un control y monitoreo de los procesos a realizarse en el robot.

Otra característica del computador de programación y visualización, es la de establecer comunicación con los circuitos de interfaz y control (a la que se referirá también como controlador) a través del uso del puerto de impresora LPT1.

7.2.4. Circuitos de alimentación del robot. Se diseñaron los circuitos de alimentación para los diversos actuadores del robot, controlador esclavo y circuitos

de interfaz, tomando en cuenta las características y necesidades de voltaje y corriente de cada uno, así como las debidas protecciones eléctricas derivadas del uso de estos elementos.

8. FASE DE DESARROLLO

8.1. SELECCION DEL BRAZO ROBÓTICO

El paso inicial para la construcción del manipulador o brazo robot, consistió en definir el modelo o tipo de robot, de acuerdo con ciertas características deseables para una determinada aplicación. La lista de características fue la siguiente:

- Características geométricas
- Características cinemáticas
- Características dinámicas
- Tipos de movimientos
- Modo de programación
- Tipos de accionamientos
- Comunicaciones
- Servicio
- Coste

Ahora bien, de acuerdo con Usategui, “una grúa-juguete puede considerarse como una aproximación al micro-robot”⁶. Teniendo en cuenta lo anterior, se decidió basarse en las características geométricas, cinemática y dinámicas de un modelo de grúa-juguete, el cual aparece en la figura 12.

⁶ USATEGUI, José y GONZALEZ, Raúl. Curso de Robótica. Madrid: Paraninfo. 1988.

Figura 12. Modelo utilizado en la construcción del robot



Esta elección se hizo debido a su sencilla configuración cilíndrica y a sus características cinemáticas y dinámicas, las cuales se describirán más adelante. Este estudio se particularizó a un modelo a escala provisto de los sistemas eléctricos, mecánicos, de transmisión y reducción de los movimientos a partir de un sistema de engranes plásticos y tornillos sin fin, los cuales proporcionan enclavamiento y ventaja mecánica al reductor. El tipo de actuadores utilizados son motores D.C de baja potencia (400 mA/5V).

Una vez elegido el modelo, se determinó los siguientes inconvenientes del modelo a escala:

- Materiales usados poco robustos (polietileno plástico)
- Errores de precisión debido al uso de engranes
- Errores de posicionamiento vertical ,debido a la inclinación de la cabina
- Estructuras poco robustas e imprecisas
- Pares bajos de los motores

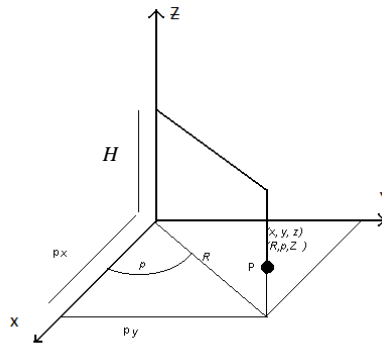
Estas características motivaron al diseño de un primer prototipo de manipulador, en donde se llegó al modelo plasmado en los planos (ver anexo A, B, C y D).

Para el desarrollo de los sistemas de transmisión, se tuvieron en cuenta las características de peso de las estructuras de las articulaciones. No obstante cabe anotar que para la presentación y puesta a punto de este prototipo se utilizaron algunos de los sistemas originales del modelo.

8.1.1 Ecuaciones cinemáticas y dinámicas para las soluciones directa e inversa del robot. Para describir la cinemática de RADAC, es preciso recordar sus tres grados de libertad (ver figura 13), los cuales son:

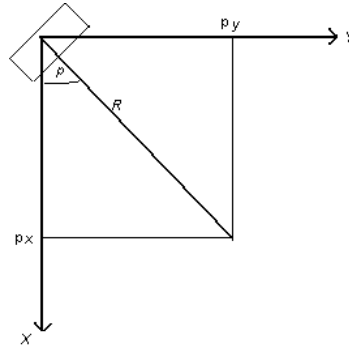
- Giro alrededor del eje Z , en un ángulo ρ .
- Traslación horizontal R , debido a su configuración, esta traslación horizontal se realiza a 30 centímetros, teniendo un recorrido máximo de 13 centímetros.
- Traslación vertical Z , que define la altura del gripper.

Figura 13. Modelo cinemático del robot



Cinemática directa. Consistirá en hallar las coordenadas del punto P de la figura 13, en función de las coordenadas x , y , z . Como el posicionamiento del robot viene definido por los valores que toman los parámetros ρ , R y Z , las coordenadas (p_x, p_y, p_z) vendrán en función de dichas variables. La figura 14 muestra la proyección de los elementos de la figura 13 sobre el plano x - y .

Figura 14. Proyección de los elementos del robot sobre el plano x-y



En donde ρ representa el recorrido angular del brazo medido desde el eje X en sentido anti horario; R representa la distancia horizontal recorrida por el antebrazo medida desde el origen de coordenadas, y Z es la distancia vertical que recorre el brazo desde una altura H que representa la altura a la que está suspendido el brazo.

De la figura se deduce:

$$\begin{aligned} p_x &= R \cdot \cos \rho \\ p_y &= R \cdot \sin \rho \end{aligned} \quad (1)$$

La coordenada Z se obtiene de la figura 13:

$$P_z = H - Z \quad (2)$$

Cinemática inversa. Se determinó el valor de los parámetros ρ , R y Z , conocida la posición del punto terminal P , mediante las coordenadas (p_x, p_y, p_z) . De la figura 14 se deduce

$$\begin{aligned} R &= \sqrt{p_x^2 + p_y^2} \\ \rho &= \arctan(p_y / p_x) \end{aligned} \quad (3)$$

De la figura 13, se deduce que:

$$Z = H - p_z \quad (4)$$

Modelado Dinámico. Debido a que el peso del manipulador es considerable y por tanto las inercias que generan influyen en el posicionamiento del robot, se ve la necesidad de establecer las relaciones dinámicas o modelo dinámico del mismo. A propósito del modelo dinámico, Barrientos⁷ dice:

“La dinámica se ocupa de la relación de las fuerzas que actúan sobre un cuerpo y el movimiento que en él se origina. Por tanto, el modelo dinámico de un robot tiene por objetivo conocer la relación entre el movimiento del robot y las fuerzas implicadas en el mismo.”

En resumen, el modelo dinámico relaciona matemáticamente:

- 1 La localización del robot definida por sus variables articulares o por las coordenadas de localización de su extremo, y sus derivadas: velocidad y aceleración
- 2 Las fuerzas y pares aplicados en las articulaciones (o en el extremo del robot).
- 3 Los parámetros dimensionales del robot como, longitud, masas e inercias de sus elementos.

Dichos argumentos motivaron a la obtención del modelo dinámico del robot mediante el algoritmo de Lagrange Euler. No obstante este análisis no contempla el análisis de fuerzas de Coriolis debido a los enclavamientos de cada articulación, de manera que se excluirán los pasos 7 y 8 de este algoritmo. A continuación se indica el algoritmo de Lagrange Euler.

L-E 1. Se asignó un sistema de referencia a cada eslabón de la siguiente manera:

⁷ BARRIENTOS, Antonio et al. Fundamentos de robótica. Madrid: Mc Graw- Hill, 1998. 326 p.

Figura 15. Sistemas de referencia asignados

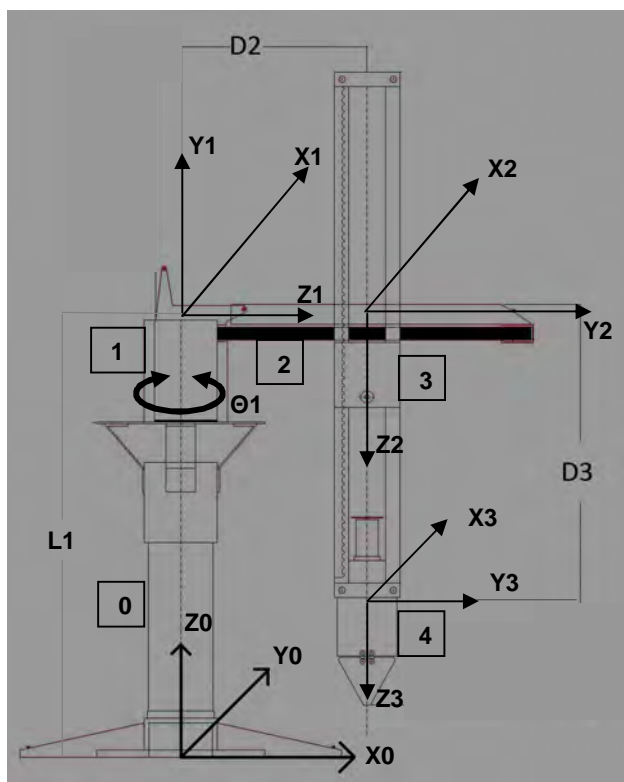


Figura 16. Distribución de los Sistemas de referencia asignados

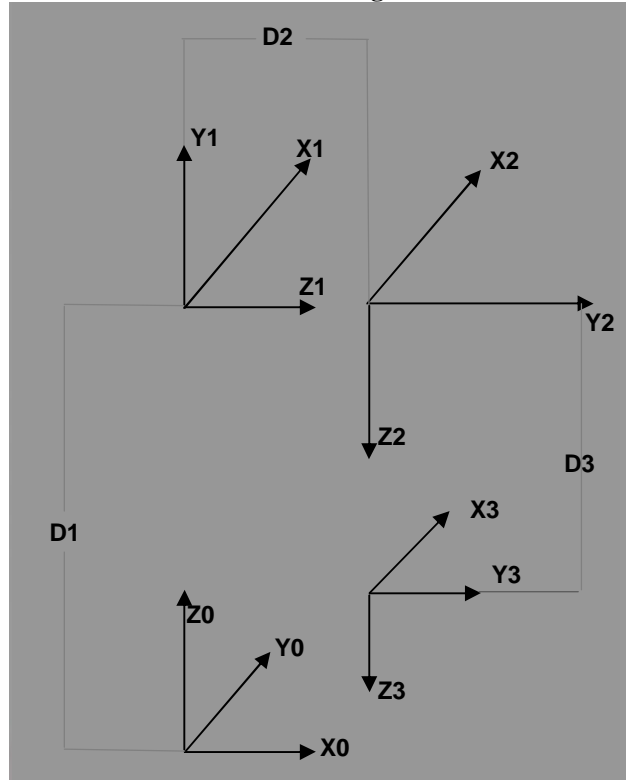


Tabla 2. Distribución de los sistemas de referencia

Articulación	θ_i	D_i	a_i	α_i
1	$\Theta_1 + 90^\circ$	L1	0	90
2	0	D2	0	90
3	0	D3	0	0

(5)

En donde θ_i , D_i , a_i y α_i son los parámetros Denavit – Hartenberg del eslabón i ; $\theta_1 = \Theta_1 + 90^\circ$

L-E 2. Se obtuvo las matrices de transformación 0A_i para cada elemento i . Es importante tener en cuenta la siguiente notación:

$\text{Sen}(\theta_1) = S1$, $\text{Cos}(\theta_1) = C1$; $L4$ = distancia del centro de masas al eje de giro de la articulación, $R = D2$; $\rho = \theta_1$; $L1$ = distancia a la que la articulación 1, se eleva del suelo.

$${}^0A_1 = \begin{bmatrix} C1 & 0 & S1 & 0 \\ S1 & 0 & -C1 & 0 \\ 0 & 1 & 0 & L1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6); \quad {}^1A_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & D2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

$${}^2A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & D3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8);$$

$${}^0A_2 = \begin{bmatrix} C1 & S1 & 0 & D2*S1 \\ S1 & -C1 & 1 & -C1*D2 \\ 0 & 0 & 1 & L1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9); \quad {}^0A_3 = \begin{bmatrix} C1 & S1 & 0 & S1*D2 \\ S1 & -C1 & 0 & -C1*D2 \\ 0 & 0 & 1 & D3+L1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10);$$

Esta última es una matriz homogénea de la forma

$${}^2A_3 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix} \quad (11)$$

De aquí, si se toma la última columna:

$$\begin{bmatrix} D2S1 \\ -D2C1 \\ D3+L1 \end{bmatrix} = \begin{bmatrix} px \\ py \\ pz \end{bmatrix} \Rightarrow \frac{py}{px} = -tg1 \Rightarrow$$

$$\theta1 = -arctg\left(\frac{py}{px}\right); \quad pz - L1 = D3$$

$$px^2 = (D2S1)^2; \quad py^2 = (D2C1)^2 \Rightarrow$$

$$px^2 + py^2 = D2^2 \Rightarrow D2 = \sqrt{px^2 + py^2} \quad (12)$$

Los anteriores resultados coinciden con el análisis cinemático anterior (ecuaciones 3 y 4)

L-E 3. Se obtuvo la matriz de energía (U cinética y potencial) del robot:

$$U_{ij} = \frac{\partial^0 A_i}{\partial \theta_j} \quad (13)$$

$$U_{11} = \frac{\partial^0 A_1}{\partial \theta_1} = \begin{bmatrix} -S1 & 0 & C1 & 0 \\ C1 & 0 & S1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (14);$$

$$U_{12} = \frac{\partial^0 A_1}{\partial D2} = [0] = U_{13} = \frac{\partial^0 A_1}{\partial D3}$$

$$U_{21} = \frac{\partial^0 A_2}{\partial \theta_1} = \begin{bmatrix} -S1 & C1 & 0 & D2C1 \\ C1 & S1 & 0 & D2S1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = U_{31} = \frac{\partial^0 A_3}{\partial \theta_1} \quad (15);$$

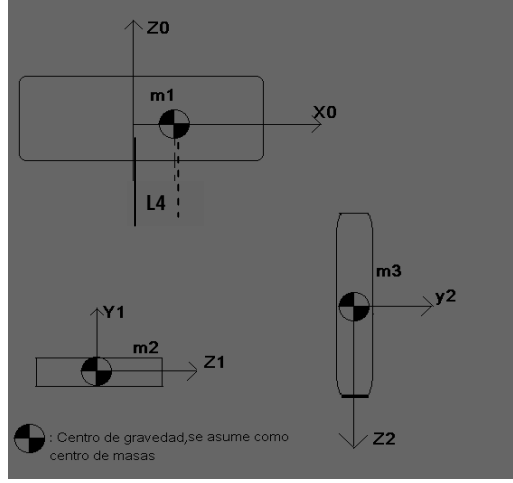
$$U_{22} = \frac{\partial^0 A_2}{\partial D2} = \begin{bmatrix} 0 & 0 & 0 & S1 \\ 0 & 0 & 0 & -C1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = U_{32} = \frac{\partial^0 A_3}{\partial D2} \quad (16);$$

$$U_{23} = \frac{\partial^0 A_2}{\partial D3} = [0] \quad (17);$$

$$U_{33} = \frac{\partial^0 A_3}{\partial D3} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (18)$$

L-E 5. Se obtuvo la matriz de pseudoinercias \mathbf{J}_i de cada elemento del robot

Figura 17. Centro de masas y ejes de giro de los elementos del robot



$$J_i = \begin{bmatrix} \int x_i^2 dm & \int x_i y_i dm & \int x_i z_i dm & \int x_i dm \\ \int x_i y_i dm & \int y_i^2 dm & \int y_i z_i dm & \int y_i dm \\ \int z_i x_i dm & \int z_i y_i dm & \int z_i^2 dm & \int z_i dm \\ \int x_i dm & \int y_i dm & \int z_i dm & \int dm \end{bmatrix} \quad (19)$$

Elemento 1

$$\begin{aligned} \int x_1^2 dm &= m_1 L^2; & \int x_1 y_1 dm &= 0; & \int x_1 z_1 dm &= 0; & \int x_1 dm &= 0; \\ \int x_1 y_1 dm &= 0; & \int y_1^2 dm &= 0; & \int y_1 z_1 dm &= 0; & \int y_1 dm &= 0; \\ \int z_1 x_1 dm &= 0; & \int z_1 y_1 dm &= 0; & \int z_1^2 dm &= 0; & \int z_1 dm &= 0; \\ \int x_1 dm &= L m_1; & \int y_1 dm &= 0; & \int z_1 dm &= 0; & \int dm &= m_1; \end{aligned} \Rightarrow J_1 = \begin{bmatrix} m_1 L^2 & 0 & 0 & m_1 L \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ m_1 L & 0 & 0 & m_1 \end{bmatrix}$$

(20)

Elemento 2

$$J_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_2 \end{bmatrix} \quad (21)$$

Elemento 3

$$J3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m3 \end{bmatrix} \quad (22)$$

L-E 6. Se obtuvo la matriz de inercias $D = [d_{ij}] = \begin{bmatrix} d11 & d12 & d13 \\ d21 & d22 & d23 \\ d31 & d32 & d33 \end{bmatrix}$; (23)

$$d_{ij} = \sum_{k=maz(i,j)}^n \text{Traza}(U_{kj} U_k^T U_{ki}^T) \Rightarrow U_{11}^T = \begin{bmatrix} -S1 & C1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ C1 & S1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (24)$$

$$U_{12}^T = [0] \quad (17); \quad U_{13}^T = [0] \quad (18); \quad U_{21}^T = \begin{bmatrix} -S1 & C1 & 0 & 0 \\ C1 & S1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ D2 * C1 & D2 * S1 & 0 & 0 \end{bmatrix} = U_{31}^T \quad (25);$$

$$U_{22}^T = U_{32}^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ S1 & -C1 & 0 & 0 \end{bmatrix} \quad (26); \quad U_{33}^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (27);$$

$$\begin{aligned} d_{11} &= \sum_{k=maz(1,1)}^3 \text{Traza}(U_{k1} J_k U_{k1}^T) = \text{Tr}(U_{11} J_1 U_{11}^T) + \text{Tr}(U_{21} J_2 U_{21}^T) + \text{Tr}(U_{31} J_3 U_{31}^T) \\ &= \text{Tr} \begin{bmatrix} S1^2 m1 L4^2 & -S1 C1 m1 L4^2 & 0 & 0 \\ 0 & C1^2 m1 L4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \text{Tr} \begin{bmatrix} D2^2 C1^2 m2 & D2 C1 m1 S1 & 0 & 0 \\ D2^2 S1 C1 m2 & D2^2 S1^2 m2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ &\quad + \text{Tr} \begin{bmatrix} D2^2 C1^2 m3 & D2 C1 m3 S1 & 0 & 0 \\ D2^2 S1 C1 m3 & D2^2 S1^2 m3 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ &= m1 L4^2 (S1^2 + C1^2) + m2 D2^2 (S1^2 + C1^2) + m3 D2^2 (S1^2 + C1^2) = m1 L4^2 + (m2 + m3) D2^2 \quad (28) \end{aligned}$$

$$\begin{aligned}
d_{12} &= \sum_{k=maz(1,2)}^3 \text{Traza}(U_{k2} J_k U_{k1}^T) = \text{Tr}(U_{22} J_2 U_{21}^T) + \text{Tr}(U_{32} J_3 U_{31}^T) \\
&= \text{Tr} \begin{bmatrix} D2S1C1m2 & D2S1^2m2 & 0 & 0 \\ -D2C1^2m2 & -D2S1C1m2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \text{Tr} \begin{bmatrix} D2S1C1m3 & D2S1^2m3 & 0 & 0 \\ -D2C1^2m3 & -D2S1C1m3 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
&= D2S1C1m2 - D2S1C1m2 + 0 + 0 + D2S1C1m3 - D2S1C1m3 + 0 + 0 = 0 \quad (29)
\end{aligned}$$

$$\begin{aligned}
d_{13} &= \sum_{k=maz(1,3)}^3 \text{Traza}(U_{k3} J_k U_{k1}^T) = \text{Tr}(U_{33} J_3 U_{31}^T) \\
&= \text{Tr} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ C1D2m3 & S1D2m3 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0 \quad (30)
\end{aligned}$$

$$\begin{aligned}
d_{21} &= \sum_{k=maz(2,1)}^3 \text{Traza}(U_{k1} J_k U_{k2}^T) = \text{Tr}(U_{21} J_2 U_{22}^T) + \text{Tr}(U_{31} J_3 U_{32}^T) \\
&= \text{Tr} \begin{bmatrix} D2S1C1m2 & -D2C1^2m2 & 0 & 0 \\ D2S1^2m2 & -D2S1C1m2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \text{Tr} \begin{bmatrix} D2S1C1m3 & -D2C1^2m3 & 0 & 0 \\ D2S1^2m3 & -D2S1C1m3 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0 \quad (31)
\end{aligned}$$

$$\begin{aligned}
d_{22} &= \sum_{k=maz(2,2)}^3 \text{Traza}(U_{k2} J_k U_{k2}^T) = \text{Tr}(U_{22} J_2 U_{22}^T) + \text{Tr}(U_{32} J_3 U_{32}^T) \\
&= \text{Tr} \begin{bmatrix} S1^2m2 & -S1C1m2 & 0 & 0 \\ -S1C1m2 & C1^2m2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \text{Tr} \begin{bmatrix} S1^2m3 & -S1C1m3 & 0 & 0 \\ -S1C1m3 & C1^2m3 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
&= (S1^2 + C1^2)m2 + (S1^2 + C1^2)m3 = m2 + m3 \quad (27)
\end{aligned}$$

$$\begin{aligned}
d_{23} &= \sum_{k=maz(2,3)}^3 \text{Traza}(U_{k3} J_k U_{k2}^T) = \text{Tr}(U_{33} J_3 U_{32}^T) = \text{Tr} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ m3S1 & -C1m3 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0 \quad (32)
\end{aligned}$$

$$\begin{aligned}
d_{31} &= \sum_{k=maz(3,1)}^3 \text{Traza}(U_{k1} J_k U_{k2}^T) = \text{Tr}(U_{31} J_3 U_{33}^T) = \text{Tr} \begin{bmatrix} 0 & 0 & D2C1m3 & 0 \\ 0 & 0 & D2S1m3 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0 \quad (33)
\end{aligned}$$

$$d_{32} = \sum_{k=maz(3,2)}^3 \text{Traza}(U_{k2} J_k U_{k3}^T) = \text{Tr}(U_{32} J_3 U_{33}^T) = \text{Tr} \begin{bmatrix} 0 & 0 & S1m3 & 0 \\ 0 & 0 & -C1m3 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0 \quad (34)$$

$$d_{33} = \sum_{k=maz(3,3)}^3 \text{Traza}(U_{k3} J_k U_{k3}^T) = \text{Tr}(U_{33} J_3 U_{33}^T) = \text{Tr} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & m3 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0 + 0 + m3 + 0 = m3 \quad (35)$$

Lo cual genera la siguiente matriz de inercias

$$D = \begin{bmatrix} m1L4^2 + D2^2(m2 + m3) & 0 & 0 \\ 0 & m2 + m3 & 0 \\ 0 & 0 & m3 \end{bmatrix} \quad (36)$$

L-E 9. Se obtuvo la matriz de fuerzas de gravedad

$$C = [Ci]^T; \quad Ci = \sum_{j=1}^3 (-m_j \bar{g} U_{ji}^j r_j);$$

$$\bar{g} = [0 \quad 0 \quad -g \quad 0]$$

$$C = \begin{bmatrix} 0 \\ 0 \\ -m3g \end{bmatrix} \quad (37)$$

L-E 10. Finalmente se logró el modelo dinámico descrito a través de las siguientes ecuaciones:

Ecuación 17: Resumen de los valores para cada articulación

$$\tau = D\ddot{q} + C = \begin{bmatrix} T1 \\ F2 \\ F3 \end{bmatrix} =$$

$$\begin{bmatrix} m1L4^2 + D2^2(m2 + m3) & 0 & 0 \\ 0 & m2 + m3 & 0 \\ 0 & 0 & m3 \end{bmatrix} \begin{bmatrix} \ddot{\theta}1 \\ \ddot{D}2 \\ \ddot{D}3 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ m3g \end{bmatrix} = \begin{bmatrix} 2400 + D2^2(450) & 0 & 0 \\ 0 & 450 & 0 \\ 0 & 0 & 250 \end{bmatrix} \begin{bmatrix} \ddot{\theta}1 \\ \ddot{D}2 \\ \ddot{D}3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 245000 \end{bmatrix} \quad (38)$$

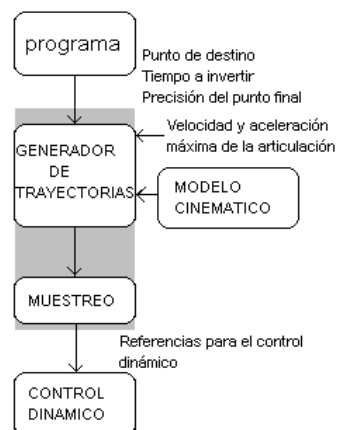
Donde $m1= 150$ gramos, $m2= 200$ gramos, $m3= 250$ gramos, $L1=25$ cm, $L4 = 4$ Centímetros.

8.1.2 Diseño del software del robot. Una vez logrado el modelado cinemático y dinámico del brazo, se estableció el sistema de control del mismo, para llevar a cabo las tareas de carga y descarga propuestas para este robot en particular.

Para el desarrollo del software se utilizó el software de aplicación LabVIEW V.7.0, cuya programación se realizó en los laboratorios de Ingeniería Electrónica de la Universidad de Nariño.

Descripción del lenguaje del robot. Según Usategui “el lenguaje de programación es el encargado de gobernar el robot desde el panel de programación”⁸. La estructura del programa, la cual incluye el generador de trayectorias basadas en el modelo cinemático, un proceso de muestreo de la trayectoria, y un controlador dinámico, aparece en la figura 18.

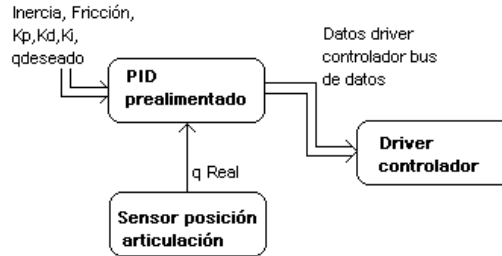
Figura 18. Diagrama de bloques general del programa



El control dinámico. Para el control de todas las articulaciones incluido el efector, se definió el control por PID con prealimentación, cuya estructura se muestra en la figura 19, y cuyo algoritmo aparece en el anexo E.

⁸ USATEGUI, José y GONZALEZ, Raúl. Curso de Robótica. Madrid: Paraninfo. 1988.

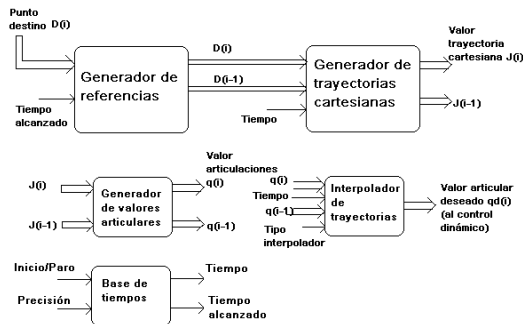
Figura 19. Diagrama de bloques programa controlador dinámico



Una vez desarrollado el programa, se construyó la interfaz de potencia, cuyo diagrama esquemático aparece en el anexo F.

El controlador cinemático. Para la generación de los valores articulares, tipo de trayectoria, precisión de movimiento y tiempo de ejecución de cada articulación incluyendo el efector, se desarrolló el controlador cinemático, cuya estructura se muestra en la figura 20, y cuyo algoritmo aparece en el anexo E.

Figura 20. Diagrama de bloques programa controlador cinemático



La interfaz de programación. En la fase inicial del proyecto, se había previsto el desarrollo de una programación textual a nivel robot, es decir, en base a instrucciones explícitas a desarrollar. Teniendo en cuenta lo anterior, la estructura de la interfaz de programación obedece a una lógica similar a la de la creación de una matriz de instrucciones, y a una posterior lectura de las mismas. En lo relacionado con la respuesta a los requerimientos de programación, podemos afirmar lo siguiente:

- En cuanto al entorno de programación, este programa ofrece la facilidad de analizar y depurar cada instrucción, a través del controlador numérico situado en el panel frontal y la ventana de simulación gráfica, la cual permite visualizar la ejecución del programa del robot sin la necesidad de su presencia física.

- Respecto al modelado del entorno, este modelo de programación atiende a objetos independientes. No obstante, debido a la flexibilidad que ofrece la programación en LabVIEW, es posible analizar otro tipo de modelos relacionales de los objetos.
- Los tipos de datos de entrada son los correspondientes a coordenadas cartesianas para posición, y ángulos de Euler para la orientación del robot, éstos últimos vienen definidos en grados. En cuanto al gripper, su representación viene dada por el porcentaje de apertura del mismo (1-abierto, 0-cerrado).
- Esta programación no admite originalmente comunicación con máquinas o procesos ajenos, no obstante es posible utilizar aplicaciones adicionales que puedan satisfacer esta necesidad, esto debido a las herramientas que este software posee.
- El control de movimiento incluye una fácil modificación del valor de precisión del movimiento del robot (en su panel frontal, la trayectoria se divide en el número de puntos indicado, si no se especifica, su valor es de 100), así como el tipo de interpolador a emplear (en su panel frontal, existe el interpolador lineal o cúbico, si no se especifica, el interpolador es lineal). También se puede modificar el valor de paso del encoder si se desea un cambio en la posición del mismo respecto al actuador.
- Finalmente el control de flujo de programa no admite cambios en la estructura de las instrucciones a realizar, no obstante esto puede lograrse introduciendo instrucciones adicionales a las del programa y modificando el puntero del mismo según las instrucciones externas.

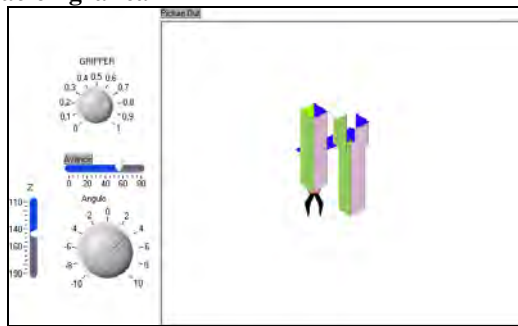
El simulador tridimensional. En el panel frontal del programa se encuentra una ventana de simulación gráfica, el cual posee un modelo sólido en 3D del robot, una botonera consistente en un botón de ejecución y de paro, y un conjunto de control-indicador de las posiciones actuales y futuras del robot, así como un control del tiempo de ejecución de cada instrucción, el tipo de interpolador a utilizar, y los controles correspondientes al control dinámico y los posibles valores de inercia (ver anexo E). Adicionalmente, existe un control de apertura y cierre del gripper del robot, susceptible a manipularse dentro de la ejecución de cada instrucción.

Mediante los tres controles de posición final, el usuario puede programar los movimientos del robot, los cuales están encaminados a alcanzar dicha posición programada en x, y, y z, medida en centímetros, así como tres controles de los ángulos de Euler medidos en grados. Existe una tecla de control de apertura del gripper, la cual está programada para asumir dos posiciones extremas (abierto o cerrado). Adicionalmente existen dos botones que indican el comienzo y el final de la ejecución, *EJECUTAR* y *STOP*. En conjunto con esto, existen los respectivos

indicadores de la posición de partida del robot, estos valores son necesarios para el adecuado posicionamiento del robot.

La ventana de programación gráfica. Para el desarrollo de la ventana de programación gráfica, se utilizaron varios sub-VI (instrumentos virtuales) de construcción y manipulación gráfica de objetos tridimensionales en el espacio, junto con un algoritmo el cual genera las ecuaciones para todas las posiciones a mostrarse en el modelo sólido en 3D del manipulador, derivadas de las ecuaciones cinemáticas anteriormente obtenidas. Este desarrollo se muestra en la figura 21.

Figura 21. Ventana de simulación gráfica



8.1.3 Construcción del manipulador. Con base en los planos del manipulador y en las entrevistas realizadas, se ha construido en acrílico las partes correspondientes a la articulación giratoria y a la de desplazamiento horizontal. Dicho modelo se muestra en la figura 22.

Figura 22. Construcción del Manipulador Prototipo



Seguido a esto, se estudiaron una serie de motores D.C. se determinó inicialmente que el motor de taladradora de 600wattios/2200 r.p.m sería viable, pero debido a

su falta de estructura de soporte de sus elementos y elevado coste de construcción del mismo, se desechó.

Figura 23. Construcción de Manipulador



Seguido a esto, se estudió un grupo de motores reductores D.C y motores de pasos para cada una de las articulaciones, de acuerdo con las necesidades del tipo de movimiento del robot, carga a manipular, velocidad y torque de los respectivos motores de pasos, de los cuales se escogieron los que a continuación aparecen:

Tabla 3. Relaciones entre articulaciones y motores escogidos

ARTICULACIÓN	MOTOR REDUCTOR	VELOCIDAD (R.P.M)
$\theta 1$	Brother Ref: BP203518-X	20
D2	Brother Ref: BP203518	60
D3	Brother Ref: BP203518	60
GRIPPER	Minebea ref: PM25L – 024 – NBJ3	150

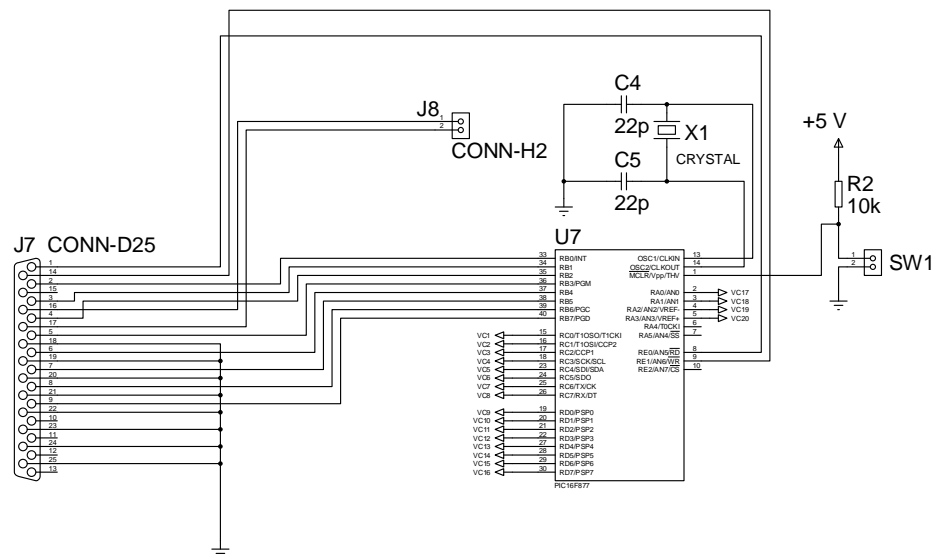
8.1.4. Fabricación de servomotores. Utilizando la caja de reducción de los motores reductores escogidos, y acoplándolos a los motores de pasos, (ver figura 24) se desarrollaron los servomotores destinados a realizar el movimiento de cada articulación del robot

Figura 24. Motor reductor utilizado como servomotor



8.1.5. Construcción circuito de interfaz por el microcontrolador 16F887. Dentro del diseño de la interfaz, se utilizó el microcontrolador 16F887 con el objeto de aprovechar las señales del puerto paralelo del computador, de esta forma solo se utilizarán dos bits del puerto para control las cuatro terminales de las bobinas de un motor de pasos en particular. El circuito que utiliza el microcontrolador se muestra en la figura 25. El programa realizado en ensamblador aparece en el anexo H.

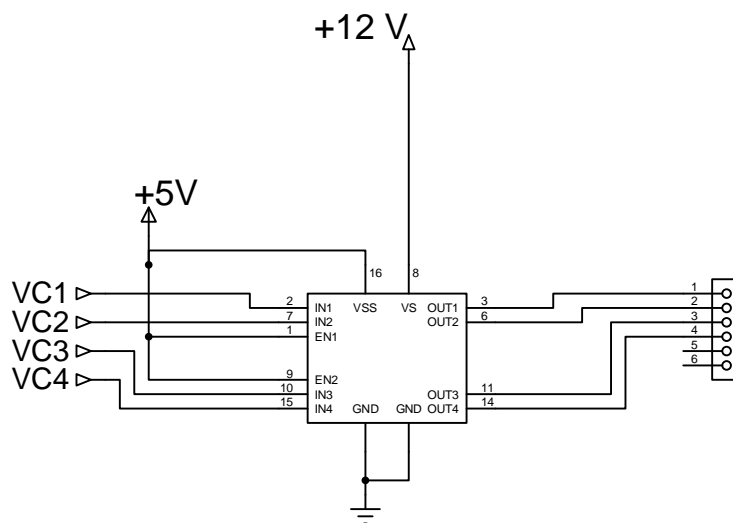
Figura 25. Circuito de controlador de motores paso a paso por PIC 16F887



8.1.6. Construcción circuito de interfaz por el circuito integrado L293B. Se desarrollaron diversos prototipos de servo amplificadores, no obstante se redujeron al mínimo el número de componentes de tal manera que se pudo

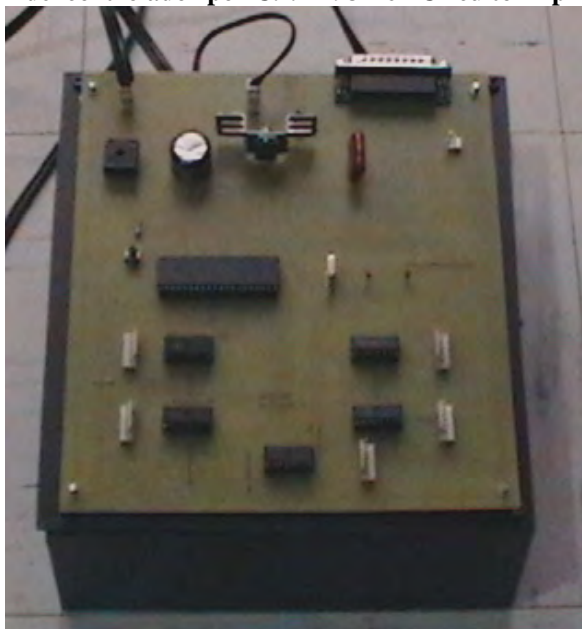
implementar un circuito simple por medio del circuito integrado L293B, y que aparece en la figura 26.

Figura 26. Prototipo de implementación del controlador de potencia por C.I. L293B



El circuito completo, plasmado en circuito impreso se muestra en la figura 27.

Figura 27. Implementación del controlador por C.I. L293B en Circuito Impreso



8.1.7. Elaboración Manual de usuario. Como fase final, una vez se realizó la fase de seguimiento del prototipo final, se elaboró un manual de usuario, con sus respectivos diagramas esquemáticos de los circuitos, uso del programa textual, así como ejemplos de uso y programación del robot. Dicho manual corresponde al Anexo I.

9. RESULTADOS Y ANÁLISIS DE RESULTADOS

Al finalizar este proyecto se obtuvo como resultado la construcción un robot servocontrolado estacionario de tipo cilíndrico de tres grados de libertad, programable por computador, que además posee las siguientes características:

- Altura total: 45 cm, longitud del brazo: 24 cm, longitud efectiva: 16 cm
- Sistemas de coordenadas: Cilíndricas
- Tres grados de libertad, (giro azimutal de 360°, elevación vertical de 10 cm y desplazamiento horizontal de cm) y un gripper de dos posiciones (abierto-cerrado).
- Capacidad de carga: baja (<0,1 kg) sólo para aplicaciones didácticas, demostrativas, de inspección mediante sensores y elementos de poco peso. Posee como actuadores, motores de Pasos
- Gripper: tipo pinza, de tres dedos dispuestos a 120 ° entre sí para un agarre completo, de movimiento por pivotaje. Además este efector cuenta con el peso y la robustez adecuadas.
- Repetibilidad: alta pero susceptible a cambios debido a la inclusión de estructuras adicionales que alteren la inercia; posee enclavamiento en todos sus desplazamientos (las articulaciones no se mueven una vez retirada la energía de los actuadores).
- Alimentación: 120 Voltios A.C. (Toma de corriente doméstica).
- Software de programación basado en LabVIEW versión 7.0 con una ventana de simulación gráfica, la cual posee un modelo sólido en 3D del robot y la capacidad de simular programas variados.
- Un módulo de control electrónico de potencia, especialmente útil para el manejo de motores de paso de 12 Voltios C.C. y/o servomotores de potencia.

Estos resultados pueden ser tomados para desarrollos mayores, esto es debido a que el diseño original puede utilizar hasta 6 grados de libertad y dos efectores finales ubicados en los extremos de las muñecas del robot. Es incluso posible aumentar más grados de libertad o incluso manejar más de un robot simultáneamente utilizando otro tipo de puerto paralelo, los cuales pese a que no son normalmente visibles, existen dentro de un P.C convencional. Esto es posible tanto en la programación textual, electrónica de potencia en las conexiones y disposiciones físicas del robot.

Por último, es importante anotar que debido al diseño del robot, cada una de las partes de éste (manipulador, control de potencia y la programación en general)

son módulos interdependientes, es decir pueden funcionar tanto en conjunto como individualmente en otro tipo de usos, lo cual puede proporcionar una gran cantidad de aplicaciones en el estudio de la automatización y el control dentro de la Universidad de Nariño (por ejemplo, aplicación de otro tipo de controladores de potencia para el manipulador, controles de motores de paso de potencia por medio de la interfaz, simulación de movimiento de un robot a través de la interfaz virtual del programa, etc.).

10. RECURSOS

10.1. RECURSOS HUMANOS

- **Asesor de la Investigación:** Ingeniero Rolando Barahona
- **Investigador:**

Programa de Ingeniería Electrónica:

- Cristian Andrés Patiño.

10.2. RECURSOS TECNOLÓGICOS

- Laboratorio de Ingeniería Electrónica Universidad de Nariño.
- Computador.
- Software de simulación, programación y control (Multisim 2000, Circuit Maker 2000, TLE. Step 7, Proteus v 7.0, C/C++ LabVIEW)
- Instrumentos de medición y análisis electrónico.
- Ficheros.
- Libros.

10.3. RECURSOS MATERIALES

- Medios de almacenamiento óptico y magnético (Diskettes, C.D's, discos duros).
- Medios de impresión (impresoras, plotters).
- Papelería en general.

10.4. RECURSOS FINANCIEROS

A continuación se indican los respectivos conceptos para la realización de este presupuesto

Presupuesto robot de aplicación didáctica para el aprendizaje conceptual de la automatización y control en los laboratorios de ingeniería electrónica de la Universidad de Nariño	
Concepto	Costo
Gastos investigación y consulta	\$ 500.000,00
Diseño de prototipos (incluye fases de documentación)	\$ 100.000,00
Fabricación de prototipos	\$ 400.000,00
Diseño producto terminado	\$ 200.000,00
Fabricación de manipulador terminado (incluye Gripper)	\$ 700.000,00
Otros	\$ 200.000,00
Recurso Humano	\$ 6'000.000,00
Total	\$ 8'100.000,00

Es importante anotar que los costos de fabricación fueron asumidos por el investigador.

10.5. RECURSO OPERATIVO

Los diseños, programas y documentación del prototipo de robot a implantar, harán parte de los equipos utilizados dentro de los laboratorios de ingeniería electrónica de la Universidad de Nariño.

El personal capacitado en el uso de este prototipo, requerirá una fase de capacitación mínima respecto al manejo de este dispositivo, así como para su posterior mantenimiento.

11. CONCLUSIONES

Se demostró la factibilidad de fabricar un modelo de robot cilíndrico de características semi-industriales con materiales comunes y con características de precisión, fiabilidad y flexibilidad necesarias para la ejecución de prácticas dentro la línea de profundización de automatización y control en los laboratorios de ingeniería de la Universidad de Nariño.

Se obtuvo el desarrollo de nuevos elementos de programación de robots en Labview y un nuevo tipo de manipulador cilíndrico. Estos elementos pueden ser destinados a su uso en robótica, automatización, electrónica de potencia, etc. debido a sus características modulares las cuales permiten un uso conjunto o individual es estos elementos.

Se comprobó que LabVIEW es un lenguaje muy versátil en el cual se pueden desarrollar cualquier tipo de aplicaciones en robótica, ya que entre otras prestaciones, se ofrece una programación de tipo gráfico, lo cual es una ventaja desde el punto de vista del programador.

Se desarrolló un manual de uso del robot RADAC, el cual permitirá el conocimiento del mismo par su adecuada manipulación, así como el desarrollo de prácticas dentro de los laboratorios de ingeniería electrónica de la Universidad de Nariño.

RECOMENDACIONES

Se recomienda enfáticamente leer cabalmente el contenido del Manual de usuario de este robot antes de realizar cualquier práctica o procedimiento de mantenimiento, reparación o ajuste. El Manual de usuario del robot RADAC versión 1.0 está incluido en el C.D denominado "ANEXOS" este manual corresponde al Anexo I.

BIBLIOGRAFÍA

BARRIENTOS, Antonio et al. Fundamentos de robótica. Madrid: Mc Graw- Hill, 1998. 326 p.

CARVAJAL, Jaime H. Automatización y mecatrónica en la educación. En: Revista de la Universidad de La Salle. Bogotá. No 39, Enero junio 2005; p 2

Cho Hyungsuck y Kim Young. Optomechatronic Technology: The characteristics and perspectives. En IEEE transactions on industrial electronics. Vol 52 No 4 (Agosto 2005): p. 933-938

CORZO, Miguel. Introducción a la ingeniería de proyectos. 6 ed. México: Limusa, 1982. 226 p.

DUEÑAS, Francisco. La Robótica. [Online] [http:// www.monografias.com/robótica](http://www.monografias.com/robótica). 2005

DYM, Clive y LITTLE. Patrick. El proceso de diseño en ingeniería. New york, Limusa, 2002. 330 p

Estructura Mecánica de un Robot. [Online] <http://proton.ucting.udg.mx/materias/robotica>. 2005

GROOVER, Mikell et al. Robótica industrial. Madrid: Mc Graw- Hill. 1989. 600 p.

HENAO, David. Introducción a los microcontroladores [online] <http://www.monografias.com>. 2005
HISPAVISTA. <http://robotica2.galeon.com/index.html>. 2005

INSTITUTO COLOMBIANO DE NORMAS TÉCNICAS. Normas Colombianas para la presentación de trabajos de investigación. Quinta actualización. Santafé de Bogotá D.C.: ICONTEC, 2006. 112 p. NTC 1486.

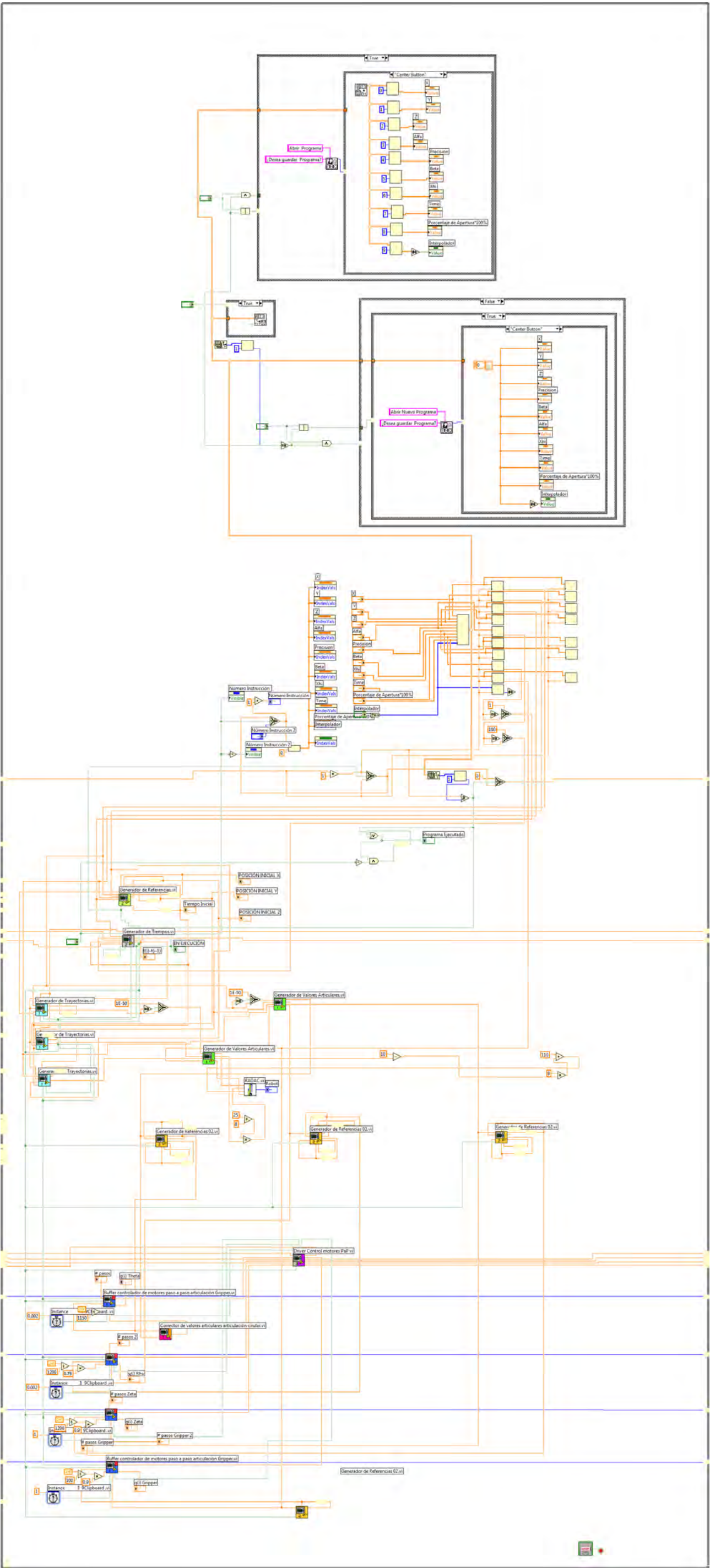
Kofman, Jonnathan et al. Teleoperation of a Robot Manipulator Using a Vision-Based Human–Robot Interface. En IEEE transactions on industrial electronics. Vol 52 No 5 (Octubre 2005): p. 1206-1207

PIEDRAFITA, Ramón. Ingeniería de la automatización industrial. Bogotá: RA-MA, 2001. 570 p

Sheng W, Chen H, Xi N, Chen Y. Tool path planning for compound surfaces in spray forming processes. En IEEE transactions on automation science and engineering. Vol 2 No 3 (Julio 2005): p. 240-242

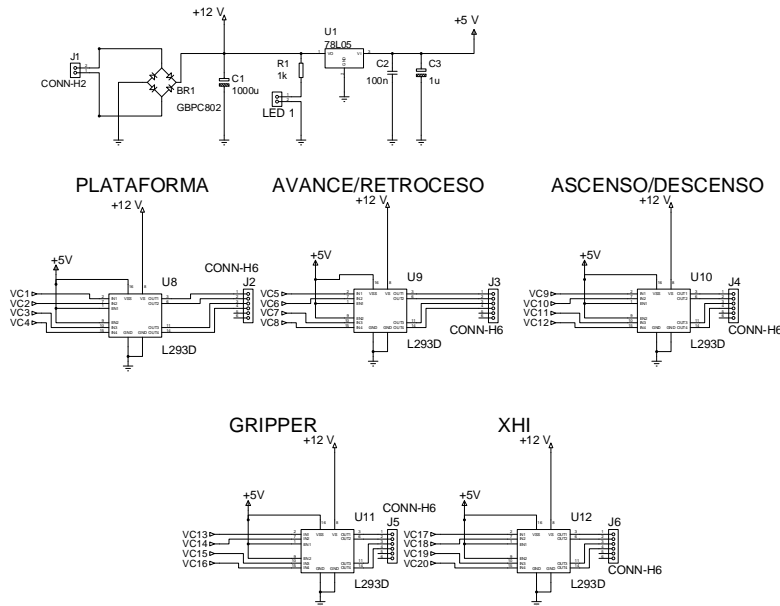
USATEGUI, José y GONZALEZ, Raúl. Curso de Robótica. Madrid: Paraninfo. 1988.

ANEXOS

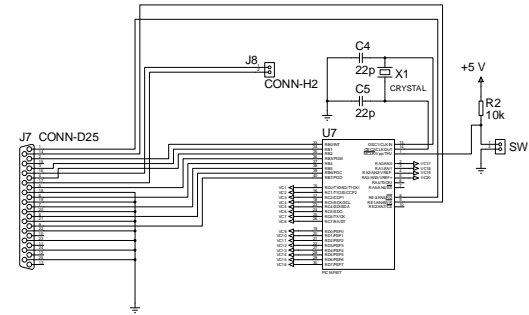


Anexo F. Diagrama esquemático interfaz de potencia

INTERFACE DE POTENCIA DE SALIDA



CONEXION DB-25



UNIVERSIDAD DE NARIÑO

DIBUJANTE
CRISTIAN ANDRES PATIÑO

CIRCUITOS DE INTERFACE DE POTENCIA

ROBOT DE APLICACION DIDACTICA PARA
AUTOMATIZACION Y CONTROL

REVISADO POR

FECHA: MAYO DE 2012

SERIE TOTAL

HOJA # 1

APROBADO POR

ESCALA:

TOTAL HOJAS

REF

Anexo G. Lista de Materiales interfaz de potencia

Componentes del circuito Impreso

Resistencias 5% 1/4 W

R1	1k Ω
R2	10k Ω

Condensadores

C1	2200uF/25V Electrolítico
C2	100nF Poliéster
C3	1u /25V Electrolítico
C4, C5	22pF
Poliéster	

Semiconductores

BR1	Puente rectificador de 4 diodos de 6 Amperios
U1	LM78L05
U7	PIC16F877
U8-U12	L293D

Varios

J1, J8, LED1	Conector Molex Macho dos terminales
J2, J3, J4, J5, J6	Conector Mólex Macho de seis terminales
J7	Conector Macho DB – 25 Para Circuito Impreso
SW1	Pulsador miniatura SPST
X1	Cristal de cuarzo de 4 Mhz
5 Zócalos DIL 16	
1 Zócalo DIL 40	
1 Disipador Para TO 220	
Placa PCB fotosensible positiva	

Componentes para Instalación en caja

- 1 Diodo LED verde 5 mm (LED1)
- 1 Portaled de 5 mm
- 1 Portafusible para panel
- 1 Fusible de 3A
- 1 Interruptor de red
- 1 Pasamuros de goma
- 1 Cable de red con enchufe macho

- 1 Transformador 110 V / 7,5 V 5 A
- 4 Tornillos 5 cm 3 mm de diámetro
- 4 Arandelas 3 mm
- 12 Tuercas 3 mm
- 1 Lámina acrílica transparente 16 por 19.5 cm
- 1 Caja acrílica

Anexo H. Programa para el control de motores paso a paso por 16F887

```
*****
;
*
; Filename: Control de Cinco Motores Unipolares Paso a Paso por puerto DB - 25
Versión 2.0
.*
;
; Descripción: Este programa pretende manejar un microcontrolador 16F887 de
modo tal que pueda manejar 5 motores paso a paso del tipo
; Unipolar por CI L293B o similares. Esto se hace para aprovechar al máximo las
señales de entrada de un puerto de un computador
; U otro dispositivo que ingrese al puerto B del PIC. Para controlar un motor se
necesita sólo dos bits. RB0 y RB1 controlan el Motor 1,
; RB2 y RB3 controlan el Motor 2, RB4 y RB5 controlan el Motor 3, RB6 y RB7
controlan el Motor 4, RE0 y RE1 controlan el Motor 5. Las
; señales que van a los puertos B y E provienen del puerto DB-25 o de cualquier
fuente que maneje 10 bits. Los bits de salida son por
; Los puertos A, C y D. En esta versión, se establece una nueva secuencia de
salida a los motores con dos bobinas: Clock wise: 9 - 10 - 6 - 5; Contra
; Clock wise: 5 - 6 - 9 - 10
;
; Comentarios: se utilizó un cristal de 4 MHZ, el cual funcionó en la práctica, de
modo que las transiciones que ingresan al
; Puerto B del PIC, no se pierdan, perdiéndose información que se convertirá en
las señales a los motores paso a paso.
;
; El programa para la "quema" del microcontrolador utilizado es PIC KIT
.*
;
; Date: 28/12/11
; File Version: 1.0 *
.*
;
; Author: Christian Andrés Patiño
; Company: Universidad de Nariño
.*
;
.*
*****
```

INCLUDE "P16F887.INC"

```
__CONFIG __CONFIG1, _XT_OSC & _WDT_OFF & _PWRTE_OFF &
_MCLRE_ON & _CP_OFF & _CPD_OFF & _BOR_OFF & _IESO_ON &
_FCMEN_ON & _LVP_OFF
__CONFIG __CONFIG2, _WRT_OFF & _BOR40V
```

cblock 20

```

;
;
;
                *PROGRAMA PRINCIPAL*
;*****
;
;*****
reset org    0        ;el vector de reset es la direccion 00
            goto inicio      ;se salta al inicio del programa
ORG         05

;*****
;
;Subrutina de inicialización del Controlador:
;*****
inicio CLRF STATUS      ;Selecciona Banco 0
           CLRF PORTA          ;Inicializa latches de datos de
PORTA
           CLRF PORTC          ;Inicializa latches de datos de
PORTC
           CLRF PORTD          ;Inicializa latches de datos de PORTD

           BSF STATUS,RP0       ;Selecciona Banco 1
           BANKSEL PORTA ;
           CLRF PORTA
           BANKSEL ANSEL
           CLRF ANSEL ;digital I/O
           BANKSEL TRISA
           MOVLW    0x00          ;configura RA0...RA5
           MOVWF    TRISA          ;como salidas

           BANKSEL PORTB ;
           CLRF PORTB ;Init PORTB
           BANKSEL ANSELH ;
           CLRF ANSELH ;digital I/O
           BANKSEL TRISB ;
           MOVLW    0xFF          ;configura RB7,...,RB0 como entradas
           MOVWF    TRISB          ;
           BCF     OPTION_REG,7 ;Conecta todas las resistencias Pull-Up

           BANKSEL PORTC ;
           CLRF PORTC ;Init PORTC

```

```

BANKSEL TRISC
movlw00                ;configura RC0...RC7 como salidas
MOVWF    TRISC
BANKSEL PORTD ;
CLRF PORTD ;Init PORTD
BANKSEL TRISD
movlw00                ;configura RD0...RD7 como salidas
MOVWF    TRISD

BANKSEL PORTE ;
CLRF PORTE ;Init PORTE
BANKSEL ANSEL ;
CLRF ANSEL ;digital I/O
BANKSEL TRISE
movlw03                ;configura RE0...RE1 como entradas
MOVWF    TRISE
BCF  STATUS,RP0        ;regresa al Banco 0
CLRF PORTC              ;Apagar Motor 1 y 2
CLRF PORTD              ;Apagar Motor 3 y 4
CLRF PORTA              ;Apagar Motor 5
CLRF CONT1              ;borrar contador 1
CLRF CONT2              ;borrar contador 2
CLRF CONT3              ;borrar contador 3
CLRF CONT4              ;borrar contador 4
CLRF CONT5              ;borrar contador 5
CLRF MOTOR1             ;borrar contenido motor 1
CLRF MOTOR2             ;borrar contenido motor 2
CLRF MOTOR3             ;borrar contenido motor 3
CLRF MOTOR4             ;borrar contenido motor 4
CLRF MOTOR5             ;borrar contenido motor 5
MOVLW    0x99           ;inicializar motores
MOVWF    PORTC           ;inicializar motor 1 y 2
MOVWF    PORTD           ;inicializar motor 3 y 4
MOVLW    0x09           ;inicializar motores
MOVWF    PORTA           ;inicializar motor 3 y 4; todo esto se
hace en caso de tener que resetear el PIC

.*****
; Rutina de reconocimiento del código para controlar cada motor: RBX | RBY |
PASO
;
|      C      0      |      0

```


[illegible]

* SECUENCIA PASO 1 *

PASO1MOTOR1

PASO2MOTOR1

```

        BTFSC     STATUS, Z           ;si el código de RB0 y RB1
es "0",      GOTOCODIGOP2M1         ;ejecute el paso 3

```

PASO3MOTOR1

```

        BTFSC PORTB, 0           ;detecte el estado de RB0
        BSF     CONT1, 0         ;escriba el bit detectado

```

```

        BTFSC PORTB, 1           ;detecte el estado de RB0
        BSF     CONT1, 1         ;escriba el bit detectado

```

RB0yRB1igual2

```

        MOVLW     0x02           ;Código del paso
        XORWF     CONT1, 0       ;compare el bit
detectado con la posición anterior
        BTFSC     STATUS, Z     ;si el código de RB0 y RB1
es "0",      GOTOCODIGOP3M1     ;ejecute el paso 4

```

PASO4MOTOR1

```

        BTFSC PORTB, 0           ;detecte el estado de RB0
        BSF     CONT1, 0         ;escriba el bit detectado

```

```

        BTFSC PORTB, 1           ;detecte el estado de RB1
        BSF     CONT1, 1         ;escriba el bit detectado

```

RB0yRB1igual3

```

        MOVLW     0x03           ;Código del paso
        XORWF     CONT1, 0       ;compare el bit
detectado con la posición anterior
        BTFSC     STATUS, Z     ;si el código de RB0 y RB1
es "0",      GOTOCODIGOP4M1     ;ejecute el paso 1
              GOTOPASO1MOTOR2   ;vaya a paso la secuencia 2

```

; A continuación, la rutina para el movimiento de un motor paso a paso usando el nibble que utiliza los bits 0 a 3

```

;*****
;

```

CODIGOP1M1

```
                MOVLW 0x09                                ;código del primer
paso 1          MOVWF  MOTOR1                            ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 1
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT1                                ;limpiar CONT1 para
habilitar el contador del siguiente paso
                GOTOPASO1MOTOR2                          ;vaya a paso 2
```

CODIGOP2M1

```
                MOVLW 0x0A                                ;código del primer
paso           MOVWF  MOTOR1                            ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 1
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT1                                ;limpiar CONT1 para
habilitar el contador del siguiente paso
                GOTOPASO1MOTOR2                          ;vaya a paso 3
```

CODIGOP3M1

```
                MOVLW 0x06                                ;código del primer
paso           MOVWF  MOTOR1                            ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 1
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT1                                ;limpiar CONT1 para
habilitar el contador del siguiente paso
                GOTOPASO1MOTOR2                          ;vaya a paso 4
```

CODIGOP4M1

```
                MOVLW 0x05                                ;código del primer
paso           MOVWF  MOTOR1                            ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 1
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT1                                ;limpiar CONT1 para
habilitar el contador del siguiente paso
                GOTOPASO1MOTOR2                          ;vaya a paso la secuencia 2
```

,*****
,

```

*****
;
;
;

```

* SECUENCIA PASO 2 *

```

*****
;

```

PASO1MOTOR2

```

BTFSC PORTB, 2          ;detecte el estado de RB2
BSF  CONT2, 0           ;escriba el bit detectado

```

```

BTFSC PORTB, 3          ;detecte el estado de RB3
BSF  CONT2, 1           ;escriba el bit detectado

```

RB2yRB3igual a 0

```

MOVLW 0x00              ;Código del paso
XORWF  CONT2, 0         ;compare el bit
detectado con la posición anterior
BTFSC  STATUS, Z        ;si el código de RB2 y RB3
es "0",
GOTOCODIGOP1M2          ;ejecute el paso 1

```

PASO2MOTOR2

```

BTFSC PORTB, 2          ;detecte el estado de RB2
BSF  CONT2, 0           ;escriba el bit detectado

```

```

BTFSC PORTB, 3          ;detecte el estado de RB3
BSF  CONT2, 1           ;escriba el bit detectado

```

RB2yRB3igual a 1

```

MOVLW 0x01              ;Código del paso
XORWF  CONT2, 0         ;compare el bit
detectado con la posición anterior
BTFSC  STATUS, Z        ;si el código de RB2 y RB3
es "1",
GOTOCODIGOP2M2          ;ejecute el paso 2

```

PASO3MOTOR2

```

BTFSC PORTB, 2          ;detecte el estado de RB2
BSF  CONT2, 0           ;escriba el bit detectado

```

BTFSC PORTB, 3	;detecte el estado de RB3
BSF CONT2, 1	;escriba el bit detectado

RB2yRB3igual2

MOVLW 0x02	;Código del paso
XORWF CONT2, 0	;compare el bit
detectado con la posición anterior	
BTFSC STATUS, Z	;si el código de RB2 y RB3
es "2",	
GOTOCODIGOP3M2	;ejecute el paso 3

PASO4MOTOR2

BTFSC PORTB, 2	;detecte el estado de RB2
BSF CONT2, 0	;escriba el bit detectado
BTFSC PORTB, 3	;detecte el estado de RB3
BSF CONT2, 1	;escriba el bit detectado

RB2yRB3igual3

MOVLW 0x03	;Código del paso
XORWF CONT2, 0	;compare el bit
detectado con la posición anterior	
BTFSC STATUS, Z	;si el código de RB2 y RB3
es "3",	
GOTOCODIGOP4M2	;ejecute el paso 4
GOTOCAMBIONIBBLEM2	;vaya a
CAMBIONIBBLEM2	

; A continuación, la rutina para el movimiento de un motor paso a paso usando el nibble que utiliza los bits 0 a 3
 ,*****

CODIGOP1M2

MOVLW 0x09	;código del primer
paso 1	
MOVWF MOTOR2	;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 2	
CLRF STATUS	;limpiar la bandera de cero
CLRF CONT2	;limpiar CONT2 para
habilitar el contador del siguiente paso	

```

                                GOTOCAMBIONIBBLEM2                ;vaya a paso 2

CODIGOP2M2

                                MOVLW 0x0A                        ;código del primer
paso
                                MOVWF MOTOR2                      ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 2
                                CLRF STATUS                      ;limpiar la bandera de cero
                                CLRF CONT2                        ;limpiar CONT2 para
habilitar el contador del siguiente paso
                                GOTOCAMBIONIBBLEM2                ;vaya a paso 3

```

```

CODIGOP3M2

                                MOVLW 0x06                        ;código del primer
paso
                                MOVWF MOTOR2                      ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 2
                                CLRF STATUS                      ;limpiar la bandera de cero
                                CLRF CONT2                        ;limpiar CONT2 para
habilitar el contador del siguiente paso
                                GOTOCAMBIONIBBLEM2                ;vaya a paso 4

```

```

CODIGOP4M2

                                MOVLW 0x05                        ;código del primer
paso
                                MOVWF MOTOR2                      ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 2
                                CLRF STATUS                      ;limpiar la bandera de cero;
envíe los datos por el puerto C
                                CLRF CONT2                        ;limpiar CONT2 para
habilitar el contador del siguiente paso
.*****
;
;
;
*          ENVÍO SECUENCIAS MOTOR 1 Y 2 POR PUERTO C          *
.*****
;

```

```

CAMBIONIBBLEM2

                                SWAPF MOTOR2, 1                  ;Mueva los nibbles menos
significativos de MOTOR 2
                                MOVFMOTOR2, 0                    ;Mueva MOTOR 2 al acumulador

```

MOVERAPUERTOC

```
ADDWF    MOTOR1, 0      ;Sumar W + MOTOR 1 y
llevar el resultado al acumulador
MOVWF    PORTC           ;Mover los bits al
puerto C
```

*****,
,

* SECUENCIA PASO 3 *

*****,
,

PASO1MOTOR3

```
BTFSC PORTB, 4           ;detecte el estado de RB4
BSF  CONT3, 0             ;escriba el bit detectado
```

```
BTFSC PORTB, 5           ;detecte el estado de RB5
BSF  CONT3, 1             ;escriba el bit detectado
```

RB4yRB5igual a 0

```
MOVLW    0x00             ;Código del paso
XORWF    CONT3, 0          ;compare el bit
detectado con la posición anterior
BTFSC    STATUS, Z         ;si el código de RB4 y RB5
es "0",
GOTOCODIGOP1M3            ;ejecute el paso 1
```

PASO2MOTOR3

```
BTFSC PORTB, 4           ;detecte el estado de RB4
BSF  CONT3, 0             ;escriba el bit detectado
```

```
BTFSC PORTB, 5           ;detecte el estado de RB5
BSF  CONT3, 1             ;escriba el bit detectado
```

RB4yRB5igual a 1

```
MOVLW    0x01             ;Código del paso
XORWF    CONT3, 0          ;compare el bit
detectado con la posición anterior
BTFSC    STATUS, Z         ;si el código de RB4 y RB5
es "1",
```

GOTOCODIGOP2M3

;ejecute el paso 2

PASO3MOTOR3

BTFSC PORTB, 4
BSF CONT3, 0

;detecte el estado de RB4
;escriba el bit detectado

BTFSC PORTB, 5
BSF CONT3, 1

;detecte el estado de RB5
;escriba el bit detectado

RB4yRB5igual2

MOVLW 0x02
XORWF CONT3, 0
detectado con la posición anterior
BTFSC STATUS, Z
es "2",

;Código del paso
;compare el bit

;si el código de RB4 y RB5

GOTOCODIGOP3M3

;ejecute el paso 3

PASO4MOTOR3

BTFSC PORTB, 4
BSF CONT3, 0

;detecte el estado de RB4
;escriba el bit detectado

BTFSC PORTB, 5
BSF CONT3, 1

;detecte el estado de RB5
;escriba el bit detectado

RB4yRB5igual3

MOVLW 0x03
XORWF CONT3, 0
detectado con la posición anterior
BTFSC STATUS, Z
es "3",

;Código del paso
;compare el bit

;si el código de RB0 y RB1

GOTOCODIGOP4M3

;ejecute el paso 4

; A continuación, la rutina para el movimiento de un motor paso a paso usando el nibble que utiliza los bits 0 a 3

.*****
,

CODIGOP1M3


```

                MOVLW 0x09                                ;código del primer
paso 1
                MOVWF MOTOR3                              ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 3
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT3                                ;limpiar CONT3 para
habilitar el contador del siguiente paso
                GOTOPASO1MOTOR4                            ;vaya a paso 2

```

CODIGOP2M3

```

                MOVLW 0x0A                                ;código del primer
paso
                MOVWF MOTOR3                              ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 3
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT3                                ;limpiar CONT3 para
habilitar el contador del siguiente paso
                GOTOPASO1MOTOR4                            ;vaya a paso 3

```

CODIGOP3M3

```

                MOVLW 0x06                                ;código del primer
paso
                MOVWF MOTOR3                              ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 3
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT3                                ;limpiar CONT3 para
habilitar el contador del siguiente paso
                GOTOPASO1MOTOR4                            ;vaya a paso 4

```

CODIGOP4M3

```

                MOVLW 0x05                                ;código del primer
paso
                MOVWF MOTOR3                              ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 3
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT3                                ;limpiar CONT3 para
habilitar el contador del siguiente paso

```

```

,*****
,

```

```

,*****
,

```

BTFSC PORTB, 6	;detecte el estado de RB6
BSF CONT4, 0	;escriba el bit detectado
BTFSC PORTB, 7	;detecte el estado de RB7
BSF CONT4, 1	;escriba el bit detectado

	MOVLW	0x00		;Código del paso
	XORWF	CONT4, 0		;compare el bit
detectado con la posición anterior				
	BTFSC	STATUS, Z		;si el código de RB6 y RB7
es "0",				
	GOTOCODIGOP1M4			;ejecute el paso 1

```

BTFSC PORTB, 6           ;detecte el estado de RB6
BSF  CONT4, 0             ;escriba el bit detectado

BTFSC PORTB, 7           ;detecte el estado de RB7
BSF  CONT4, 1             ;escriba el bit detectado

```

	MOVLW	0x01		;Código del paso
	XORWF	CONT4, 0		;compare el bit
detectado con la posición anterior				
	BTFSC	STATUS, Z		;si el código de RB6 y RB7
es "1",				
	GOTOCODIGOP2M4			;ejecute el paso 2

```

BTFSC PORTB, 6           ;detecte el estado de RB6
BSF  CONT1, 0             ;escriba el bit detectado

BTFSC PORTB, 7           ;detecte el estado de RB7
BSF  CONT4, 1             ;escriba el bit detectado

```

RB6yRB7igual2

```

                MOVLW    0x02                ;Código del paso
                XORWF    CONT4, 0            ;compare el bit
detectado con la posición anterior
                BTFSC    STATUS, Z          ;si el código de RB6 y RB7
es "2",
                GOTOCODIGOP3M4              ;ejecute el paso 3

```

PASO4MOTOR4

```

                BTFSC PORTB, 6                ;detecte el estado de RB6
                BSF     CONT4, 0              ;escriba el bit detectado

                BTFSC PORTB, 7                ;detecte el estado de RB7
                BSF     CONT4, 1              ;escriba el bit detectado

```

RB6yRB7igual3

```

                MOVLW    0x03                ;Código del paso
                XORWF    CONT4, 0            ;compare el bit
detectado con la posición anterior
                BTFSC    STATUS, Z          ;si el código de RB6 y RB7
es "3",
                GOTOCODIGOP4M4              ;ejecute el paso 4

```

; A continuación, la rutina para el movimiento de un motor paso a paso usando el nibble que utiliza los bits 0 a 3

,*****

CODIGOP1M4

```

                MOVLW    0x09                ;código del primer
paso 1
                MOVWF    MOTOR4              ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 4
                CLRF    STATUS                ;limpiar la bandera de cero
                CLRF    CONT4                ;limpiar CONT4 para
habilitar el contador del siguiente paso
                GOTOCAMBIONIBBLEM4          ;vaya a paso 2

```

CODIGOP2M4

```

                MOVLW 0x0A                                ;código del primer
paso
                MOVWF MOTOR4                              ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 4
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT4                                ;limpiar CONT4 para
habilitar el contador del siguiente paso
                GOTOCAMBIONIBBLEM4                        ;vaya a paso 3

```

CODIGOP3M4

```

                MOVLW 0x06                                ;código del primer
paso
                MOVWF MOTOR4                              ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 4
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT4                                ;limpiar CONT4 para
habilitar el contador del siguiente paso
                GOTOCAMBIONIBBLEM4                        ;vaya a paso 4

```

CODIGOP4M4

```

                MOVLW 0x05                                ;código del primer
paso
                MOVWF MOTOR4                              ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 4
                CLRF STATUS                                ;limpiar la bandera de cero;
envíe los datos por el puerto D
                CLRF CONT4                                ;limpiar CONT2 para
habilitar el contador del siguiente paso

```

```

*****
;
*****
;
;
*          ENVÍO SECUENCIAS MOTOR 3 Y 4 POR PUERTO D          *
*****
;

```

CAMBIONIBBLEM4

```

                SWAPF MOTOR4, 1                            ;Mueva los nibbles menos
significativos de MOTOR 2
                MOVFMOTOR4, 0                              ;Mueva MOTOR 2 al acumulador

```

MOVERAPUERTOD

```
ADDWF    MOTOR3, 0      ;Sumar W + MOTOR 1 y
llevar el resultado al acumulador
MOVWF    PORTD           ;Mover los bits al
puerto C
```

```
*****
;
;
*          SECUENCIA PASO 5          *
*****
PASO1MOTOR5
```

```
BTFSC PORTE, 0           ;detecte el estado de RE0
BSF  CONT5, 0            ;escriba el bit detectado

BTFSC PORTE, 1           ;detecte el estado de RE1
BSF  CONT5, 1            ;escriba el bit detectado
```

RE0yRE1igual a 0

```
MOVLW    0x00            ;Código del paso
XORWF    CONT5, 0         ;compare el bit
detectado con la posición anterior
BTFSC    STATUS, Z        ;si el código de RE0 y RE1
es "0",
GOTOCODIGOP1M5           ;ejecute el paso 1
```

PASO2MOTOR5

```
BTFSC PORTE, 0           ;detecte el estado de RE0
BSF  CONT5, 0            ;escriba el bit detectado

BTFSC PORTE, 1           ;detecte el estado de RE1
BSF  CONT5, 1            ;escriba el bit detectado
```

RE0yRE1igual a 1

```
MOVLW    0x01            ;Código del paso
XORWF    CONT5, 0         ;compare el bit
detectado con la posición anterior
```

es "1",	BTFSC STATUS, Z	;si el código de RE0 y RE1
	GOTOCODIGOP2M5	;ejecute el paso 2

PASO3MOTOR5

BTFSC PORTE, 0	;detecte el estado de RE0
BSF CONT5, 0	;escriba el bit detectado

BTFSC PORTE, 1	;detecte el estado de RE1
BSF CONT5, 1	;escriba el bit detectado

RE0yRE1igual2

	MOVLW 0x02	;Código del paso
	XORWF CONT5, 0	;compare el bit
detectado con la posición anterior		
	BTFSC STATUS, Z	;si el código de RE0 y RE1
es "2",	GOTOCODIGOP3M5	;ejecute el paso 3

PASO4MOTOR5

BTFSC PORTE, 0	;detecte el estado de RE0
BSF CONT5, 0	;escriba el bit detectado

BTFSC PORTE, 1	;detecte el estado de RE1
BSF CONT5, 1	;escriba el bit detectado

RE0yRE1igual3

	MOVLW 0x03	;Código del paso
	XORWF CONT5, 0	;compare el bit
detectado con la posición anterior		
	BTFSC STATUS, Z	;si el código de RE0 y RE1
es "3",	GOTOCODIGOP4M5	;ejecute el paso 4

; A continuación, la rutina para el movimiento de un motor paso a paso usando el nibble que utiliza los bits 0 a 3

CODIGOP1M5

```

                MOVLW 0x09                                ;código del primer
paso 1          MOVWF  MOTOR5                            ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 5
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT5                                ;limpiar CONT5 para
habilitar el contador del siguiente paso
                GOTOMOVERAPUERTO A                        ;vaya a paso 2

```

CODIGOP2M5

```

                MOVLW 0x0A                                ;código del primer
paso           MOVWF  MOTOR5                            ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 5
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT5                                ;limpiar CONT5 para
habilitar el contador del siguiente paso
                GOTOMOVERAPUERTO A                        ;vaya a paso 3

```

CODIGOP3M5

```

                MOVLW 0x06                                ;código del primer
paso           MOVWF  MOTOR5                            ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 5
                CLRF STATUS                                ;limpiar la bandera de cero
                CLRF CONT5                                ;limpiar CONT5 para
habilitar el contador del siguiente paso
                GOTOMOVERAPUERTO A                        ;vaya a paso 4

```

CODIGOP4M5

```

                MOVLW 0x05                                ;código del primer
paso           MOVWF  MOTOR5                            ;mueva el contenido
del acumulador que contiene el paso a seguir en el motor 5
                CLRF STATUS                                ;limpiar la bandera de cero;
envíe datos por el puerto E
                CLRF CONT5                                ;limpiar CONT5 para
habilitar el contador del siguiente paso

```

```

,*****
,

```

```

*****
;
*
;
*          *      ENVÍO SECUENCIAS MOTOR 5 POR PUERTO E      *
*****
;

MOVERAPUERTO A

                MOVFMOTOR5, 0                ;Sumar W + MOTOR 1 y llevar el
resultado al acumulador
                MOVWF    PORTA                ;Mover los bits al
puerto A

CERRARCICLO

                GOTOPASO1MOTOR1              ;Cierra el ciclo

*****
;
                END
*****
;

```