



**UNIVERSIDAD ESTADAL
PENINSULA DE SANTA ELENA**

**FACULTAD SISTEMAS Y
TELECOMUNICACIONES**

CARRERA ELECTRÓNICA Y TELECOMUNICACIONES

TEMA

**“IMPLEMENTACIÓN DE UN PROTOTIPO DE UN ROBOT
MÓVIL TIPO HEXÁPODO TELEDIRIGIDO PARA LA
CONTRIBUCIÓN DE LA LIMPIEZA APLICANDO TECNOLOGÍA
OPEN SOURCE”**

TRABAJO DE TITULACIÓN

Previa a la obtención del Título de:

**INGENIERA EN ELECTRÓNICA Y
TELECOMUNICACIONES**

AUTORA

ADRIANA CRISTINA HIDALGO PANCHANA

TUTOR

ING. SENDEY AGUSTIN VERA GONZÁLEZ

LA LIBERTAD - ECUADOR

2015

La Libertad, 27 de Julio del 2015

APROBACION DEL TUTOR

En mi calidad de Tutor del trabajo de investigación, **“IMPLEMENTACIÓN DE UN PROTOTIPO DE UN ROBOT MÓVIL TIPO HEXÁPODO TELEDIRIGIDO PARA LA CONSTRIBUCIÓN EN LA LIMPIEZA APLICANDO TECNOLOGÍA OPEN SOURCE”** elaborado por Adriana Hidalgo Panchana, egresada de la Carrera de Electrónica y Telecomunicaciones, Escuela de Electrónica y Telecomunicaciones, Facultad de Sistemas y Telecomunicaciones de la Universidad Estatal Península de Santa Elena, previo a la obtención del Título de Ingeniero (a) en Electrónica y Telecomunicaciones, me permito declarar que luego de haber orientado, estudiado y revisado, apruebo en todas sus partes y autorizo al estudiante para que inicie los trámites legales correspondientes.

Atentamente

.....
Ing. Sendey Vera González
TUTOR

DEDICATORIA

Este trabajo de titulación va dedicado a mi Dios que supo guiarme, darme fuerzas para seguir adelante y no desmayar en los problemas que se presentaban, así enseñándome a enfrentar adversidades.

A mi familia que con sus palabras me motivaban a seguir adelante, a mi pareja Edgar Vergara que me brindo siempre su apoyo incondicional, a mi hijo Edgar Vergara Hidalgo que es el motivo de inspiración en cada paso de mi vida.

A mi madre que aunque no esté a mi lado, hasta su último aliento me dio palabras de apoyo y su ejemplo de nunca desistir, formando de mí una persona luchadora y con ganas de salir siempre vencedora en los problemas de la vida.

Adriana Cristina Hidalgo Panchana

AGRADECIMIENTO

Como agradecimiento especial a Dios que me permitió estar donde estoy, que me lleno de inspiración y entusiasmo cuando quería decaer.

A mi familia, que de alguna u otra manera me apoyaron incondicionalmente, gracias por su compañía, sus consejos, sus ánimos en los momentos más difíciles de la etapa estudiantil, gracias porque me tuvieron paciencia en todo este proceso de titulación.

A mí tutor, que fue un amigo durante todo este proceso de desarrollo de la tesis, el cual me brindó sus conocimientos, experiencia, paciencia y asesoría en las necesidades que requería el proyecto.

A la Universidad Estatal Península de Santa Elena, que me abrió las puertas para poder culminar esta etapa profesional.

Adriana Cristina Hidalgo Panchana

TRIBUNAL DE GRADO

Ing. Walter Orozco Iguasnia, MSc.
Decano de Facultad

Ing. Washington Torres Guin, MSc.
Director de Carrera

Ing. Sendey Agustín Vera González
Profesor Tutor

Ing. Samuel Baldomero Bustos Gaibor
Profesor Área

Ab. Joe Espinoza Ayala
Secretario General

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
ESCUELA DE ELECTRÓNICA Y TELECOMUNICACIONES

“Implementación de un prototipo de un robot móvil tipo hexápodo teledirigido para la contribución de la limpieza aplicando tecnología open source “

RESUMEN

El proyecto está basado en un robot hexápodo al que se le realiza modificaciones para agregarle una pinza robótica que recoja objetos, para su codificación se utilizó software libre como Arduino que organiza las funciones y declaraciones de todos los movimientos, también el programa Proteus Professional en donde se simuló las conexiones de los elementos. Para la implementación del prototipo se expone las bases teóricas que contienen información detallada sobre la estructura de un robot hexápodo, sus características, número de articulaciones y grados de libertad del mecanismo para su correcto funcionamiento, así como información del tipo de método para recolectar objetos. Se determinan rutas de trayectoria, como avanzar-recoger el objeto y girar- avanzar- recoger el objeto, la elección de una de ellas depende de la comparación de las medidas del exterior con las medidas de la codificación para la recolección del objeto.

Se realizó los análisis respectivos técnico, económico y operativo, indicando los software y hardware a utilizar, así mismo los costos de la implementación que fueron respectivamente accesibles, en el análisis operativo se utilizaron técnicas que fueron las encuestas a los estudiantes que reflejan la aceptabilidad de la implementación del prototipo. La elaboración de experimentos con sus pruebas y resultados respectivamente de los sistemas de control y de fuerza tienen una parte fundamental al éxito de la implementación del prototipo del robot hexápodo ya que éstos mostrarán los errores y el porcentaje de éxito sobre la funcionalidad del robot.

ÍNDICE GENERAL

Contenido	Pág.
APROBACION DEL TUTOR	II
DEDICATORIA	III
AGRADECIMIENTO	IV
TRIBUNAL DE GRADO.....	V
RESUMEN	VI
ÍNDICE DE FIGURAS.....	X
ÍNDICE DE TABLAS.....	XII
ÍNDICE DE ANEXOS.....	XIII
INTRODUCCIÓN.....	1
CAPÍTULO 1	3
MARCO REFERENCIAL	3
1. MARCO REFERENCIAL	3
1.1. Identificación del problema	3
1.2. Situación actual del problema	4
1.3. Justificación del problema	4
1.4. Objetivos de la investigación	5
1.4.1. Objetivo General.....	5
1.4.2. Objetivos Específicos.....	5
1.5. Hipótesis	6
1.6. Resultados esperados	6
CAPÍTULO 2	8
MARCO TEÓRICO	8
2. MARCO TEÓRICO	8
2.1. Antecedentes.....	8
2.1.1. Antecedentes Históricos de la robótica	8
2.1.2. Antecedentes Históricos de los robots de limpieza.....	10
2.2. Bases teóricas.....	11
2.2.1. Estructura de un robot manipulador	11
2.2.2. Formas de estructura que tiene un robot Manipulador.....	16
2.2.3. Ejemplos de Robots hexápodos	17
2.2.4. Tipos de movimiento básico de un Robot Hexápodo	18

2.2.5.	Motor.....	21
2.2.6.	Servomotores	22
2.2.7.	Control	24
2.2.8.	Sensores	26
2.3.	Variables	30
2.4.	Métodos e Instrumentos de Investigación.....	31
2.4.1.	Métodos de Investigación	32
2.4.2.	Técnicas de Investigación	32
2.5.	Términos básicos	33
CAPÍTULO 3	35
ANÁLISIS	35
3.	ANÁLISIS.....	35
3.1.	Diagrama de proceso.....	35
3.1.1.	Descripción funcional de los procesos.....	37
3.2.	Identificación de requerimientos.....	39
3.3.	Análisis del Sistema	40
3.3.1.	Análisis Técnico.....	40
3.3.2.	Análisis Económico	42
3.3.3.	Análisis Operativo.....	43
DISEÑO	45
4.	Diseño	45
4.1.	Arquitectura de la solución	46
4.1.1.	Desarrollo del Sistema de Control	46
4.1.2.	Desarrollo del Sistema de Fuerza.....	67
4.2.	Diseño de Interfaz Gráfica	77
4.2.1.	Monitor Serial	77
CAPÍTULO 5	80
IMPLEMENTACIÓN	80
5.	IMPLEMENTACIÓN Y DEMOSTRACIÓN DE HIPÓTESIS	80
5.1.	Construcción	80
5.1.1.	Construcción Mecánica.....	81
5.1.2.	Construcción Eléctrica.....	85
5.2.	Pruebas.....	93

5.2.1. Pruebas y resultados.....	93
5.3. Documentación	116
5.4. Demostración de la Hipótesis	116
CONCLUSIONES.....	117
RECOMENDACIONES.....	119
BIBLIOGRAFÍA.....	120
ANEXOS	I

ÍNDICE DE FIGURAS

	Pág.
Figura 1: Robot Hexápodo Rise.....	9
Figura 2: Izquierda Robot Aspirador, Derecha IRobot Roomba	10
Figura 3: Tipo de Estructura	11
Figura 4: Robot Manipulador con grados de libertad	12
Figura 5: Ubicación de parámetros	14
Figura 6: Cinemática Directa de una extremidad del robot	14
Figura 7: Cinemática inversa de una extremidad del robot.....	15
Figura 8 : Robot Hexápodo.....	17
Figura 9: Robot Spidermaut.....	17
Figura 10: Robot Limpiador de Playa.....	18
Figura 11: Movimiento Cuadrúpedo.....	19
Figura 12: Movimiento Trípode.....	20
Figura 13: Servomotor	22
Figura 14: Posiciones del Motor por Ancho de pulso.....	23
Figura 15: Circuito del Servomotor	24
Figura 16: Placas Controladoras	25
Figura 17: Diagrama de Procesos del Sistema.....	35
Figura 18: Diagrama del Sistema de Control.....	36
Figura 19: Diagrama del Sistema de Fuerza	37
Figura 20: Diagrama de bloques Sistema de control de Lazo Abierto.....	38
Figura 21: Diagrama de Bloques	45
Figura 22: Sistema de Control	46
Figura 23: Circuito impreso de la Tarjeta Adaptable de Control.....	47
Figura 24: Vista física de Arduino.....	48
Figura 25: Mega Arduino 2560.....	49
Figura 26: Simulación Arduino y Sensor Ultrasónico	51
Figura 27: Simulación de la conexión de Arduino con Servomotores.....	52
Figura 28: Diagrama de Compilación de Códigos en Arduino.....	55
Figura 29: Sensor Ultrasónico	56
Figura 30: Diagrama de flujos del Detección	57
Figura 31: a) Avance hasta el objeto b) Giro hacia el objeto.....	58
Figura 32: Trazado de trayectoria	59
Figura 33: Diagrama de flujo del trazado de la trayectoria.....	60
Figura 34: Posicionamiento, movimiento hacia la derecha	62
Figura 35: Posicionamiento, movimiento de avance	63
Figura 36: Posicionamiento, movimiento hacia izquierda.....	63
Figura 37: Pinza Robótica.....	65
Figura 38: Diagrama de flujos de la recolección del objeto.....	66
Figura 39: Estructura de un servomotor.....	68

Figura 40: Modelo Bidireccional	69
Figura 41: Grados de extremidad de un extremidad	70
Figura 42: a) Paralelogramo deformable b) Actuación In-situ	71
Figura 43: Cuerpo del Hexápodo	72
Figura 44: Articulaciones de una extremidad	73
Figura 45: Ubicación de las patas y extremidades	74
Figura 46: Configuración del triángulo de apoyo	75
Figura 47: Software de interfaz del robot hexápodo	77
Figura 48: Interfaz Grafica.....	78
Figura 49: Pantalla monitor Serial	79
Figura 50: Estructura del Hexápodo	81
Figura 51: Numero de patas en la estructura.....	82
Figura 52: a) Elementos sin ensamblar, b) Una pata ensamblada.....	83
Figura 53: Cuerpo del robot	84
Figura 54: Estructura y servomotores colocados en la estructura.....	84
Figura 55: Simulación en Isis Proteus.....	85
Figura 56: a) Simulación en PBC Proteus, b) Simulación en 3D visualizer.....	86
Figura 57: Tarjeta Adaptable Diseñada	86
Figura 58: Ubicación del Mega Arduino sobre el Robot	87
Figura 59: a) Engranaje b) Pinza Robótica	89
Figura 60: Pinza para el servomotor	90
Figura 61: Pieza para ajustar la pinza sobre el cuerpo del robot.....	90
Figura 62: Sensor Ultrasónico con el servomotor.....	91
Figura 63: Ubicación Sensor, Pinza y pieza metálica	92
Figura 64: Prototipo de robot Hexápodo.....	92
Figura 65: Hexapodo Caminando hacia adelante	100
Figura 66: Robot Hexápodo Girando a la izquierda	102
Figura 67: Robot Hexápodo Girando hacia la derecha	104
Figura 68: Trazado de trayectoria de la posición 0 a la 5	109
Figura 69: Trazado de una trayectoria de la posición 0 a la 2.....	111
Figura 70: Prueba Final del robot hexápodo	115

ÍNDICE DE TABLAS

	Pág.
Tabla 1: Tipo de Sensores.....	28
Tabla 2: Variable Independiente	30
Tabla 3: Variable Dependiente	30
Tabla 4: Hardware a Utilizar.....	40
Tabla 5: Software a Utilizar	41
Tabla 6: Software a Implementar.....	41
Tabla 7: Costo de Hardware a utilizar	42
Tabla 8: Costo de software a utilizar	42
Tabla 9: Costo de Hardware a Implementar	42
Tabla 10: Costo Materiales a Utilizar	43
Tabla 11: Costos Totales.....	43
Tabla 12: Características del Mega Aduino.....	50
Tabla 13: Características de un Sensor Ultrasónico.....	56
Tabla 14: Características del Servomotor	57
Tabla 15: Posiciones y Grados.....	61
Tabla 16: Posiciones y Movimientos correspondientes	61
Tabla 17: Posiciones, Movimiento y Siguiente paso de la estructura.....	62
Tabla 18: Características del Servomotor	66
Tabla 19: Distancia del Sensor Ultrasónico para recoger el objeto	67
Tabla 21: Descripción de los Servomotores	73
Tabla 21: Corriente Suministrada por los elementos	76
Tabla 22: Designación de cada pin en la tarjeta Mega Arduino	88
Tabla 23: Tiempo vs Velocidad de desplazamiento	94
Tabla 24: El tiempo correspondiente a un paso	95
Tabla 25: Distancia de un paso	95
Tabla 26: Valores de referencia de ángulos para las extremidades	96
Tabla 27: Estados del prototipo de robot en pie.....	97
Tabla 28: Resultados de robot cuando camina hacia delante.....	99
Tabla 29: Resultados cuando robot camina hacia izquierda	101
Tabla 30: Resultados cuando robot gira a la derecha.....	103
Tabla 31: Resultados del valor de detección de la posición 0.....	106
Tabla 32: Resultados del valor de detección de la posición 4.....	108
Tabla 33: Resultados del Prueba final.....	114

ÍNDICE DE ANEXOS

N.	Descripción
	ANEXO 1: CERTIFICADO GRAMATICAL
	ANEXO 2: ENCUESTAS
	ANEXO 3: LIBRERIAS
	ANEXO 4: PROGRAMACIÓN FINAL
	ANEXO 5: SIMULACIÓN DEL CIRCUITO
	ANEXO 6: CARACTERÍSTICAS TÉCNICAS DEL ARDUINO
	ANEXO 7: CARACTERÍSTICAS TÉCNICAS DEL SERVOMOTOR
	ANEXO 8: CARACTERÍSTICAS TÉCNICAS DEL SENSOR ULTRASÓNICO
	ANEXO 9: MANUAL DE USUARIO

INTRODUCCIÓN

La tecnología es el conjunto de conocimientos sistematizados para la creación de medios a beneficio de satisfacer necesidades al ser humano. El estudio de las ramas de la robótica, informática y electrónica son ciencias que estudian las técnicas y procesos a seguir para la creación de robots y maquinas que contribuyen al trabajo del ser humano.

Un robot móvil realiza su trayectoria sobre un territorio firme por medio de patas, llantas u orugas para su desplazamiento y trayectoria. En este proyecto se realiza la implementación de un robot móvil tipo hexápodo, el objetivo principal es que contribuya a la limpieza del laboratorio de la carrera electrónica y telecomunicaciones mediante un software de control con tecnología open source, la culminación del prototipo cumplirá con el objetivo y servirá como un antecedente o guía para investigaciones futuras en el campo tecnológico.

Para el sistema de control se utilizará la tarjeta mega arduino que contiene pines de entrada/salida suficientes, para conectar los servomotores y un sensor ultrasónico, el software de arduino tendrá una interacción con la tarjeta y mediante la adquisición de datos enviará una señal para que la estructura del robot hexápodo se desplace y realice la trayectoria, la interfaz gráfica se la realiza mediante la plataforma de arduino y su monitor serial. La detección, trazado de trayectoria y recolección, en conjunto cumplirán la función de detectará al objeto a recoger del suelo, analizando valores de distancia y tomando los grados de movimiento del servomotor que sostiene al sensor ultrasónico.

En el desarrollo de esta investigación se especifica todo el trabajo dividiéndolo en: 5 capítulos y 6 Anexos.

En el capítulo 1: se detalla el marco referencial especificando la identificación, situación y justificación del problema a resolver, así como los objetivos que se elaborarán para la creación del robot hexápodo.

En el capítulo 2: se expone el marco teórico y bases teóricas, que contienen información de antecedentes históricos y actuales de los robots hexápodos y robots de limpieza, información esencial para la implementación del robot hexápodo.

En el capítulo 3: se explica el análisis acerca de los requerimientos técnicos, económicos y operativos que se necesitaran ante la implementación del robot hexápodo; así como también la descripción de los procesos de la funciones del robot hexápodo.

En el capítulo 4: se detalla la arquitectura de la solución del tema y el desarrollo de los sistemas de control, locomoción y trayectoria; como el control de la interfaz gráfica mediante el monitor serial.

En el capítulo 5: se desarrolla específicamente los sistemas de la implementación del robot móvil tipo hexápodo, la elaboración de las pruebas para sus correcciones; adicional a estos capítulos se desarrollan las conclusiones y recomendaciones del proyecto.

CAPÍTULO 1

MARCO REFERENCIAL

1. MARCO REFERENCIAL

En este primer capítulo se detallará el desarrollo del marco referencial el análisis principal para reconocer cual es el problema y solución al mismo.

1.1. Identificación del problema

Los desechos en el planeta se incrementan de forma geométrica conforme aumenta la población, existen diferentes métodos para recolectar y reciclar basura, unas de las opciones para recolectar basura es aplicando tecnología con sistemas automáticos, para dicha actividad existen los robot domésticos y un tipo de estos robots son aplicables a la recolección de polvo y desechos; son elementos tecnológicos que contribuyen a la limpieza del ambiente.

En la actualidad existen muchos robots móviles recolectores de basura, unos utilizan el tipo de locomoción con ruedas los cuales tienen un sistema de aspiración o de barrido para recoger basuras como polvo, migas y pelusas; otro tipo son los robots orugas diseñado por su fácil acceso a territorios que no son totalmente planos poseen el sistema de recolección tipo tenazas y son mayormente utilizados en investigaciones científicas en el espacio exterior.

Existe una variedad de modelos que recolectan objetos pero del robot tipo hexápodo existen muy pocos, uno de los encontrados es el Lynxmotion-A-Pod tipo hormiga capaz de recoger objetos que no son exactamente basura. Estos tipos de robots no han sido muy investigados y puestos en uso, más solo se los han creado en pequeños proyectos ya que se prefiere mayormente los robot con ruedas. También han sido

diseñados robots limpiadores para las calles como el DustBot que recogen basura en lugares públicos que acudirán al lugar con realizar una llamada telefónica. Otro tipo de robot llamado DustClean está equipado con utensilios de limpieza y con un olfato electrónico capaz de detectar olores contaminantes atmosféricos. Estos robots todavía están en pruebas y sus costos de adquisición son elevados.

1.2. Situación actual del problema

En la Universidad Estatal Península de Santa Elena, en el laboratorio de la carrera de Electrónica y Telecomunicaciones se observó que los estudiantes que asisten al laboratorio tienen poco cuidado con el aseo ya que después de cada hora de clase se encontraban en el suelo desperdicios los cuales no han sido depositados en el tacho de la basura. En el caso del laboratorio existe poco personal encargado específicamente a la limpieza, es por aquello que para el laboratorio se quiso implementar un robot capaz de recolectar objetos que contribuya a la limpieza del aula.

Se optó por examinar este tipo de robot móvil ya que se pretende utilizar el modelo manipulado anteriormente en clases de robótica, en lo cual solo se llegó a coordinar las patas del hexápodo y se le quiere dar un poco más de evolución a este tipo de estructura añadiendo a éste una función adicional, adquiriendo una estructura semejante a la de un insecto que para este caso se asemeja a una araña que recolecta desechos.

1.3. Justificación del problema

Los robots que se crean para ayudar a la vida humana por lo general están representados por una figura humana, crear un humanoide genera grandes gastos. La elaboración de un robot para actividad de limpieza supondrá la apertura de una nueva línea en el campo investigativo en carrera de Electrónica y

Telecomunicaciones, específicamente en el área de la Robótica, Física, Cálculo, Informática y demás área de conocimiento que este necesite sobre robots móviles, la extensión de proyectos como estos ayudaría a consolidar conocimientos científicos- técnicos en los estudiantes de la carrera.

Tras la necesidad de mantener limpio el laboratorio de Electrónica y Telecomunicaciones después de cada clase y buscando una manera económica de hacer la limpieza, se abre un campo de investigación en esta área utilizando la tecnología robótica y automatismo.

1.4. Objetivos de la investigación

1.4.1. Objetivo General

Implementar un prototipo de un robot móvil tipo hexápodo para la contribución de la limpieza en el laboratorio de la carrera Electrónica y Telecomunicaciones, mediante el uso de tecnología Open Source.

1.4.2. Objetivos Específicos

- Estudio de la tecnología Open Source aplicada a los robots hexápodos para la selección del algoritmo de control.
- Diseño del control semiautónomo para la coordinación de movimientos y calculo cinemático del robot hexápodo.
- Desarrollar un algoritmo de control para detectar y recolectar un objeto.
- Implementar un prototipo de un robot móvil tipo hexápodo mediante el uso de arduino, actuadores y sensores.

- Sintetizar los resultados experimentales con el robot hexápodo para evaluar y verificar su funcionamiento.

1.5. Hipótesis

La implementación de un prototipo de un robot móvil tipo hexápodo teledirigido tendrá una eficiencia del 80 % en la identificación y recolección del objeto.

1.6. Resultados esperados

- Con la elaboración del robot hexápodo que contribuirá a la limpieza mediante la recolección de un objeto-basura se supone que al utilizarlo como un instrumento de estudio ayudara a consolidar conocimientos científicos- técnicos en los estudiantes de la carrera Electrónica y Telecomunicaciones.
- Mediante la implementación de este prototipo de robot móvil tipo hexápodo utilizando tecnología open source se supone que la limpieza del laboratorio será automatizada.
- Con el estudio realizado acerca de la tecnología open source aplicada a los robots hexápodos se espera que al momento de utilizar la codificación sea el adecuado y facilite el manejo del robot hexápodo.
- Con el diseño de control semiautónomo y el cálculo cinemático se espera que el robot reciba las señales requeridas para realizar desplazamientos y movilización de la pinza robótica.

- Con la codificación de la parte mecánica de la pinza se espera que el robot tome una ruta y sea capaz de levantar un objeto.
- Con la implementación de los diferentes elementos electrónicos y la estructura física se espera que el robot realice la contribución de la limpieza en el laboratorio correctamente.
- Con las pruebas experimentales realizadas se espera tomar datos para demostrar la eficiencia sobre el control del robot para que ejecute la actividad de coordinación de movimientos y de recolección.

CAPÍTULO 2

MARCO TEÓRICO

2. MARCO TEÓRICO

Se presentarán conceptos necesarios para comprender el desarrollo sobre el tema de la Implementación del Robot Móvil tipo Hexápodo, su historia, antecedentes y conceptos de los elementos electrónicos relacionados a la construcción del robot.

2.1. Antecedentes

En este capítulo se va describir la historia de los robots hexápodos y la historia de robots de limpieza.

2.1.1. Antecedentes Históricos de la robótica

A continuación se detallará de manera breve la evolución de la robótica, de manera especial de los robots hexápodos.

En el año 1921 Karel Carpek utiliza la palabra robot en su obra “Rossum’s Universal Robots, de origen checo robota, que significa dedicado a trabajos forzados. Así mismo la palabra robótica fue añadida por Issac Asimov para explicar la tecnología y la ciencia de los robots, también creó las tres leyes de la robótica. En aquellos tiempos la imaginación de algunos era que los robots se construyeran para elaborar labores domésticas y visitar lugares inhóspitos.

El primer robot hexápodo OSU programable a través de un ordenador fue construido en 1977, robot inspirado en un insecto. Desde esta fecha las investigaciones para controlar a robot de seis patas a través de algoritmos

comenzaron. Este robot se convirtió en el punto de enfoque para pruebas experimentales con resultados para el control de robots por medio de algoritmos.

En 1983 se da a conocer el robot hexápodo ODEX, de la corporación Odetics. Dicha compañía construyó una versión mejorada en 1990 para que realice la tarea de inspeccionar una planta nuclear.

Koyachi en el año de 1996 desarrolla el robot hexápodo Melmantis, con la características de alzar dos de sus patas y manipular una pelota mientras las otras cuatro son utilizadas para mantenerse de pie, al momento de desplazarse utilizaba las seis patas.

Para el 2000 se construyó el robot humanoide con la capacidad de desplazarse de forma bípeda e interactuar con las personas por ASIMO Honda Motor Co. Ltd.

En el 2005 Kim desarrolla el robot hexápodo Rise como se muestra en la figura (1) con la función específica de trepar paredes ya que cuenta con diminutas espinas en las puntas de las patas que le permiten adherirse a superficies de paredes de materiales como concreto ladrillo, cemento madera, etc.



Figura 1: Robot Hexápodo Rise

Fuente: Evolución de la Robótica.com

2.1.2. Antecedentes Históricos de los robots de limpieza

Tomy Dustbot

Primer Robot Aspirador japonés creado en el año 1985, diseñado para poseer una aspiradora incorporada. DustBot tenía ojos grandes, luces rojas y dos pequeños brazos que realizan movimientos de barrido y aspiraba cosas como polvo, migajas, trozos de papel, también detectaba obstáculos y cambiaba su trayectoria.

Robot aspirador IRobot Roomba

Este es uno de los últimos robots creados de tipo aspirador. Ya en el 2002 la empresa iRobot coloca a la venta el robot aspirador Roomba. Este robot posee un conjunto de sensores que le ayudan a mejorar su función como las de esquivar obstáculos, detectar manchas sucias en el suelo, detectar espacios largos de caídas por ejemplo el espacio que existe en una escalera.

En la figura (2) se muestra a la izquierda la parte física del primer robot Tomy Dustbot y a la derecha uno de los últimos el Robot IRobot Roomba.



Figura 2: Izquierda Robot Aspirador, Derecha IRobot Roomba

Fuente: Robots aspiradores.com

2.2. Bases teóricas

En esta parte del capítulo II se definirá conceptos básicos y tecnológicos para desarrollar esta tesis, tales como características, estructura de un manipulador, sistemas de movimientos de los hexápodos también lo relacionado a elementos placa de control, actuadores, sensores, fuente de energía, etc.

2.2.1. Estructura de un robot manipulador

El conocimiento de la parte teórica de la estructura de un robot manipulador es indispensable para el desarrollo del tema.

La estructura mecánica

La estructura mecánica de un brazo robot está conformada por una enlace cinemático de elementos o eslabones unidos por las articulaciones. Físicamente un brazo robot es comparado con las extremidades de un brazo humano.

En la figura (3) se muestra un tipo de estructura mecánica.

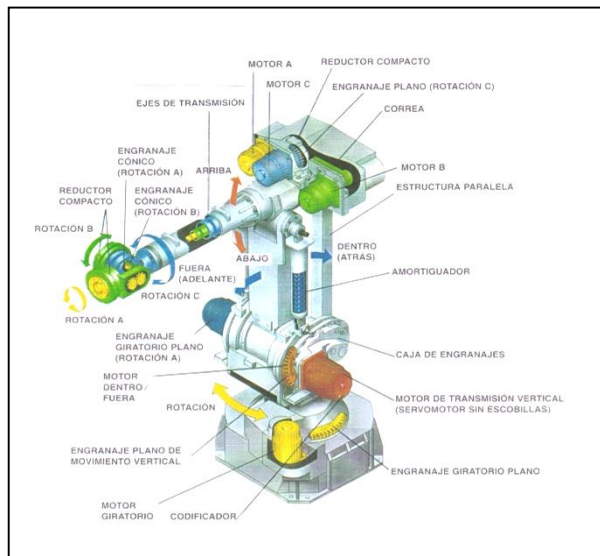


Figura 3: Tipo de Estructura

Fuente: A. Barrientos

Grados de libertad

Los grados de libertad son los movimientos individuales que realiza una articulación. El total de grados de libertad de un robot casi siempre coincide con el número de las articulaciones. 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 él la pieza o herramienta manipulada de cualquier modo en el espacio, se precisarán de seis GDL.

Al momento de construir un robot se tiene que tomar muy en cuenta los cambios que se puede realizar en las articulaciones ya que esto influye mucho en la programación del robot. En la figura (4) se muestra los grados de libertad.

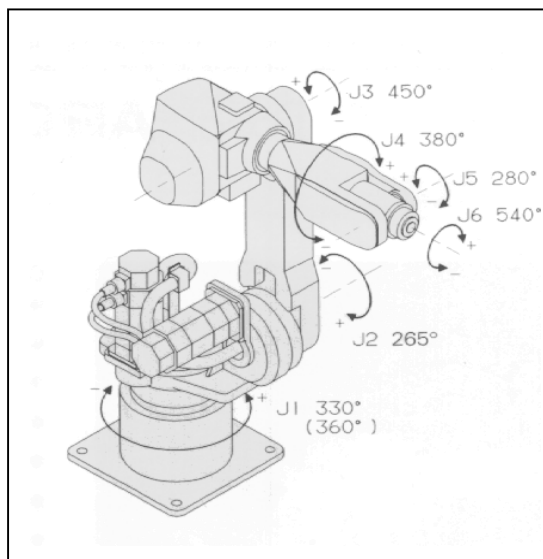


Figura 4: Robot Manipulador con grados de libertad

Fuente: A. Barrientos

Eslabones

Los eslabones se unen a una articulación, es decir, dos eslabones contiguos con una articulación producen un movimiento. Los eslabones deben ser ligeros pero así mismo ser tan rígidos como fuera posible ya que estos albergaran un peso adicional.

Articulaciones

Es la que permite el movimiento entre elementos en contacto.

El tipo de articulación definirá como se moverá el eslabón. Existen dos tipos de articulaciones la prismática y la giratoria. La prismática permite deslizar en línea recta a otra articulación y la giratoria toma la forma de una bisagra entre un eslabón y otro. Por lo general los robots están conformados por un conjunto de eslabones rígidos los cuales se conectan con articulaciones.

Tipos de movimiento en articulaciones

El movimiento de cada articulación puede ser de desplazamiento, de giro, o una combinación de ambos.

Cinemática de un robot hexápodo

Para el análisis del modelo cinemático se tiene las siguientes limitaciones:

- El robot tiene que desplazarse en una superficie plana.
- Debe de ser hecho de un material firme, no flexible.
- Que cada pata tenga un direccionamiento al suelo.
- Que la punta de las patas tengan movilidad y no presenten algún defecto para desplazarse.

Una extremidad consta de tres ejes móviles uno de estos en contacto con el suelo, definiendo tres grados de libertad, la dirección del eje Z es el que va a admitir que la pata se mueva hacia adelante o atrás para que avance o camine, la dirección del eje Y indica que el robot este más erguido hacia lo alto o más encogido y la dirección del eje X permite que al estar en contacto con el suelo los otros ejes Z y X giren y avancen mientras este eje se encuentra firme y se intercale los movimientos y el proceso se repita con todas las patas para realizar el avance.

Cada eslabón está constituido por las longitudes de cada segmento que conforman la extremidad y cada articulación por los puntos 1, 2 y 3. Se obtiene a través de esto la posición el extremo de la extremidad $p(X, Y, Z)$ en función de los ángulos en cada articulación en función de la posición de la extremidad. En la figura (5) se

muestra una pata del robot hexápodo con tres articulaciones, tres grados de libertad y sus ejes móviles x , y , z .

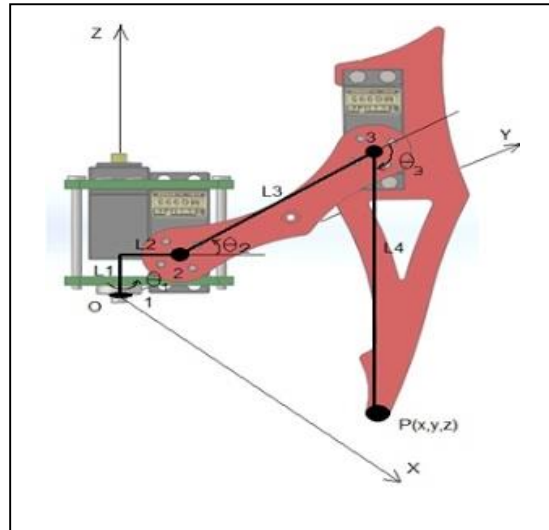


Figura 5: Ubicación de parámetros

Fuente: A. Barrientos

Cinemática Directa

La cinemática directa se adquiere mediante la posición y orientación del extremo de la extremidad de robot tomando como referencia los valores articulares y los parámetros geométricos. Como se hace referencia en la figura (6).

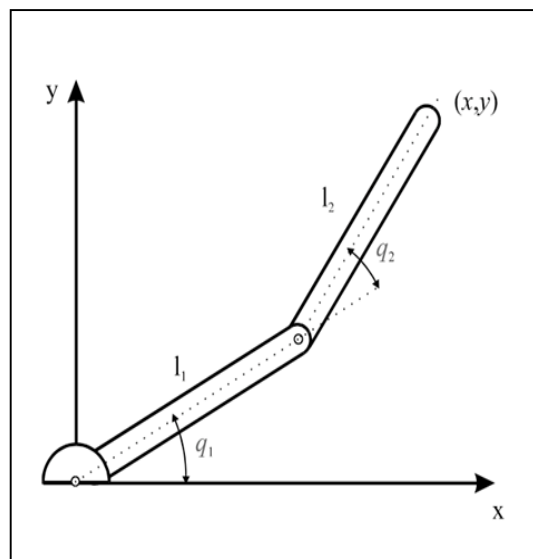


Figura 6: Cinemática Directa de una extremidad del robot

Fuente: A. Barrientos

Dada la posición del efector final y la longitud de cada enlace, encontrar los ángulos de las articulaciones. El objetivo consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot $q = (q_1, q_2, \dots, q_n)$ para que su extremo se posicione y oriente según una determinada localización espacial. El procedimiento de obtención de las ecuaciones es dependiente de la configuración del robot.

Cinemática Inversa

La cinemática inversa mediante el valor de las coordenadas articulares de posición y orientación conocidas del extremo de una pata como se muestra en la figura (7).

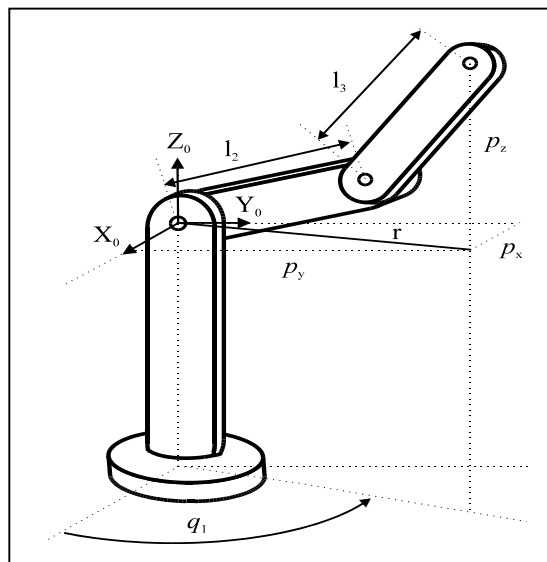


Figura 7: Cinemática inversa de una extremidad del robot

Fuente: A. Barrientos

Para mostrar el procedimiento a seguir se va a aplicar el método a un robot con 3DOF de rotación (estructura típica articular). El dato de partida son las coordenadas (P_x, P_y, P_z) referidas en las que se requiere posicionar su extremo. Como se ve este robot posee una estructura planar, quedando este plano definido por el ángulo de la primera variable articular q_1 .

2.2.2. Formas de estructura que tiene un robot Manipulador

El cuerpo del robot es el chasis que se utiliza y es el que va soportar el peso de los elementos que se agregan, tiene que tener firmeza y estabilidad, este actúa en conjunto con los motores para realizar un determinado movimiento. A continuación se nombra algunas formas de estructura que tiene un robot:

- Poli articulados
- Móviles, Androides
- Zoomórficos: Caminadores y no Caminadores.

Detallaremos los Zoomórficos que son el tipo de estructura que utilizaremos en el desarrollo del tema.

Zoomórficos

Podría agregarse también a los androides porque también imitan a los diferentes seres vivos según su sistema de locomoción. No siempre un robot zoomórfico tiene la forma de un animal, suficiente es que realice la misma trayectoria de un animal para permanecer a esta clasificación.

Zoomórficos Caminadores

Un hexápodo es un robot zoomórfico mecánico con la figura de un insecto que transita sobre sus seis patas. Caracterizado por su flexibilidad al desplazarse, ya que si alguna extremidad llegara a inmovilizarse el robot igual seguiría su camino. Al realizar robots hexápodos se comprueba teorías biológicas acerca de la locomoción de insectos. Un robot para estar de pie necesita solo tres patas las tres patas adicionales sirven para que el robot mantenga nuevas posiciones y avance.

En la figura (8) se muestra el robot Hexápodo de casa comercial.



Figura 8 : Robot Hexápodo

Fuente: Arqhys

2.2.3. Ejemplos de Robots hexápodos

Robot Spidernaut

En la figura (9) se muestra el robot Spidernaut creado por la NASA.



Figura 9: Robot Spidernaut

Fuente: Robótica: manipuladores y robots móviles.

Diseñado para atender naves espaciales en mal estado. Estos tipos de robots están siendo muy investigados para utilizarlos en el futuro como medio de transporte

autónomo o teledirigido, siendo su trayectoria en medios muy dificultosos con obstáculos. Son muy aplicados en las investigaciones espaciales y en estudios volcánicos.

Robot Limpiador de Playa

Tesis de un Robot Hexápodo Autónomo presentada por estudiantes de ingeniería de la Universidad Autónoma de México, robot capaz de recolectar objetos en la playa, el modo de operación del robot es mediante una tarjeta controladora y con visión artificial. La araña comercial fue modificada para agregarle la parte de la pinza y el control. Se muestra en la figura (10) la estructura mecánica del robot limpiador de playa.



Figura 10: Robot Limpiador de Playa

Fuente: Tesis Robot limpiador de playa

2.2.4. Tipos de movimiento básico de un Robot Hexápodo

Desde siempre la mira del hombre al crear un robot ha sido inspirado mirando la naturaleza, y esta clase de robot son un ejemplo claro. Los insectos desde el inicio de la vida han ido evolucionando su sistema de tracción se han establecido desde un par de patas hasta medio centenar de patas.

Los tipos de movimiento básicos de los robots hexápodos son dos: Movimiento Cuadrúpedo y Movimiento Trípede.

Ciclo de Movimiento Cuadrúpedo

Las características principales de este movimiento es que el robot como se muestra en la figura (11) siempre mantiene cuatro patas en el suelo, consiguiendo un grado elevado de equilibrio; siendo las dos restantes las que se levantan la estructura del robot.

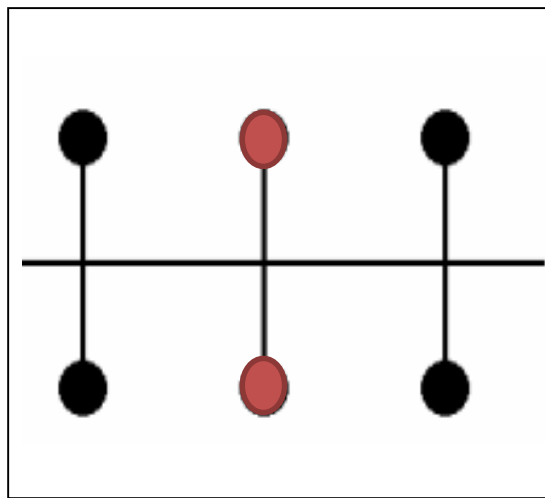


Figura 11: Movimiento Cuadrúpedo

Fuente: Alcabot

Ciclo de movimiento Trípede

Según la teoría del triángulo de apoyo como consecuencia de movimientos levanta tres patas mientras las otras tres se mantienen en el suelo. Cuando las tres patas levantadas ya estén de nuevo en el suelo las otras tres avanzan así continuamente.

Este sistema mantiene siempre el sistema de equilibrio. En la figura (12 a) se muestra el primer paso después de que el robot se encuentre de pie y la figura (12) b) se muestra el siguiente paso después de avanzar las tres patas hacia delante las otras tres prosiguen y repiten el mismo proceso.

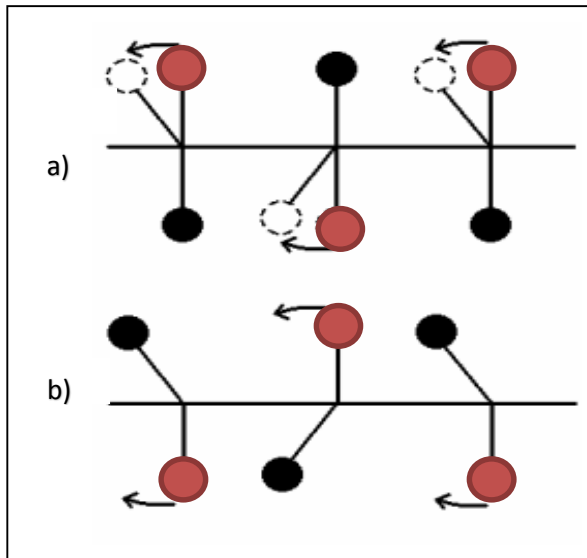


Figura 12: Movimiento Trípode

Fuente: Alcabot

El movimiento con un robot de seis patas tiene más sencillez que el de cuatro patas ya que la de cuatro tiene que levantar una a la vez y las otras tres mantener el equilibrio en el suelo. Si analizamos el robot hexápodo se verá que siempre serán tres patas las que permanezcan en el suelo, con lo cual estará estable y se podrá hacer que las otras tres patas restantes se movilicen paralelamente, es decir, que todas se encuentren conectadas a un solo mecanismo, facilitando la sencillez de su codificación. Si las particularidades del robot poseen menos complejidad, siempre brindaran agilidad y velocidad. Si se proyecta elaborar un robot todo terreno con la capacidad de subir obstáculos grandes un robot hexápodo es perfecto. Si por lo contrario se requiere un robot sencillo el hexápodo sobresale a otro tipo de robot zoomórfico por su tracción en velocidad. La gran ventaja entre las patas y las ruedas es la adaptación a diferentes lugares, por ejemplo un auto que posee ruedas no puede subir escaleras, comparando con un insecto que por lo general superan obstáculos y problemas. Si con estos robot hexápodos se desea escalar obstáculos se requiere una programación potente, por ejemplo si el hexápodo esta frente a un obstáculo lo primero que realiza es detectarlo, sea por sistemas establecidos en bumpers o reflexivos, así convirtiéndose en un desafío al momento de la programación.

Un Ejemplos claro es robot hexápodo Aleman Lauron III del FZI de Karlsruhe, todo terreno, con sistema de presión en cada punta del pie, y sus extremidades totalmente articuladas capaces de subir y bajar pendientes y evadir obstáculos.

2.2.5. Motor

Los actuadores son dispositivos que son capaces de transformar energía hidráulica, neumática o eléctrica en la activación de un proceso con el fin de generar un efecto sobre un control automatizado. Este recibe la orden de un regulador o controlador y en función a ella genera la orden para activar un elemento final de control. Un actuador es un dispositivo totalmente mecánico cuya función es proporcionar fuerza para mover otro dispositivo mecánico.

Existen varios tipos de Motores los térmicos, motores de combustión interna, motores de combustión externa y motores Eléctricos.

Motores Eléctricos

Se llaman motores eléctricos ya que usan la energía eléctrica para que el robot ejecute sus movimientos. Las características de control, sencillez y precisión de los accionamientos eléctricos han hecho que sean los más usados en los robots industriales actuales.

Motores de Corriente Continua

El motor de corriente continua es una máquina que convierte la energía eléctrica en mecánica, principalmente el movimiento rotatorio. Esta máquina de corriente continua es una de las más versátiles en la industria. Su fácil control de posición, paro y velocidad la convierten en la mejor opción en aplicaciones de control y automatización de procesos. Estos motores se clasifican en:

- Motores paso a paso
- Servomotores

- Motor sin Núcleo

2.2.6. Servomotores

Un servomotor posee la característica de ubicar se en cualquier posición dentro de su rango de operación y se mantiene estable en dicha posición. Conformado por un motor, una caja reductora y un circuito de control, utilizados frecuentemente en sistemas de radiocontrol y en robótica.

Un pequeño elemento electrónico que contiene en su interior un motor diminuto que reduce la velocidad y multiplica la fuerza, así contiene un circuito controlador que permite su ángulo de giro hasta 180 grados. Los servomotores son los encargados de dar movimiento al robot.

La duración y frecuencia son los que determinan el control sobre el servomotor. Todos los servomotores disponen de tres cables, dos para alimentación Vcc y Gnd (4.8 a 6 [V]) y un tercero para aplicar el tren de pulsos de control, que hace que el circuito de control diferencial interno ponga el servomotor en la posición indicada, dependiendo del ancho del pulso.

En la figura (13) se muestra el estado físico el servomotor.



Figura 13: Servomotor

Fuente: Maquinas Eléctricas

Funcionamiento del servomotor. Control PWM

Modulación por ancho de pulso, PWM sistema empleado para controlar los servomotores, el sistema genera una onda cuadrada en la cual varía el tiempo que el pulso se mantiene en nivel alto, manteniéndose periodo para modificar la posición del servomotor. Se utiliza un *timer* y un comparador para generar la onda PWM en un microcontrolador, hará que la señal sea generada automática y efectiva. El *timer* se programa con el ancho de pulso y el comparador con el valor de duración del pulso a nivel alto. El pulso entre 1ms y 2ms de anchura son los valores generales que dejan al motor entre 0° y 180° en los extremos. La posición de 1.5 ms indicará la posición central, es decir 90°. Cuando existen valores superiores a 1ms y a 2ms, fuera del rango indicado el motor emitirá un zumbido indicando que se cambie la longitud del pulso. En la figura (14) se muestra las posiciones de los pulsos más comunes, fuera de este rango el motor emite un zumbido.

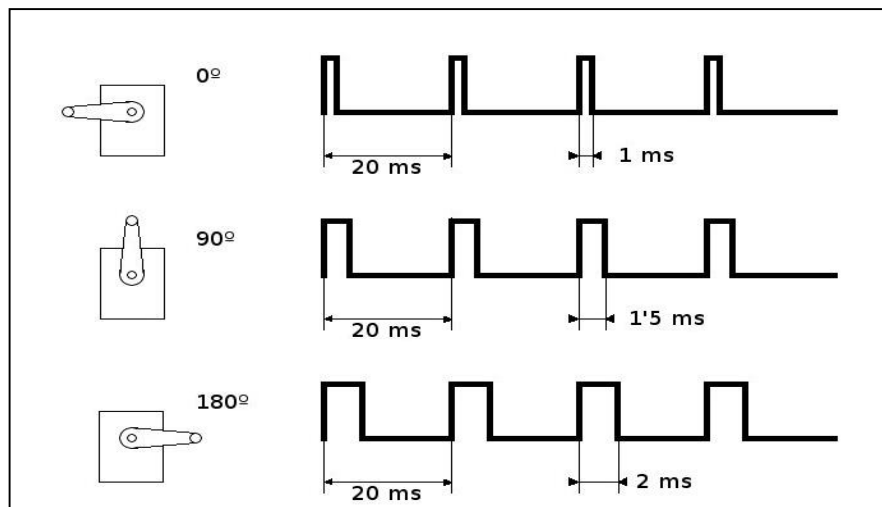


Figura 14: Posiciones del Motor por Ancho de pulso

Fuente: Maquinas Eléctricas

Circuito del Servomotor

El circuito que se presenta en la figura (15) sirve para controlar un servomotor típico de modelismo con una resistencia variable. Al mover la resistencia variable cambiara en la misma proporción la posición del servomotor.

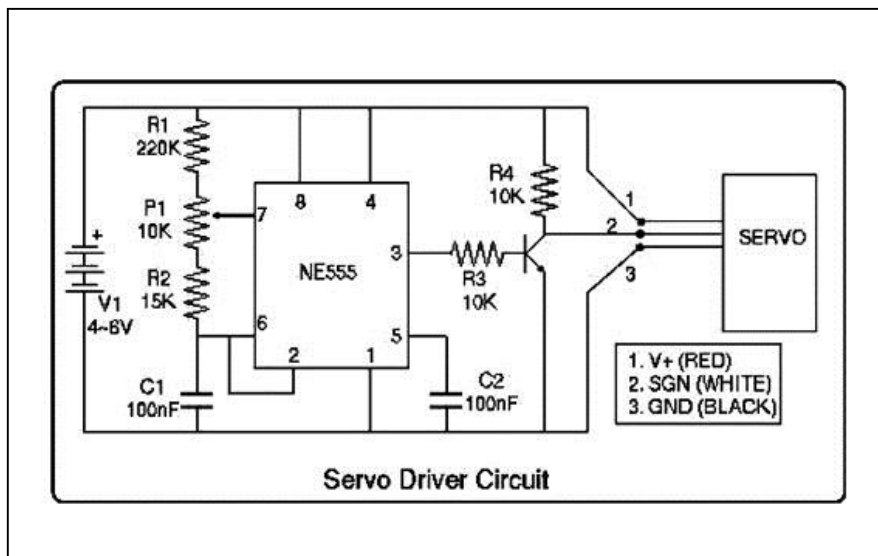


Figura 15: Circuito del Servomotor

Fuente: Maquinas Eléctricas

2.2.7. Control

La parte de control es como el cerebro del robot, conformada por elementos electrónicos complejos, esta placa de control se selecciona según los requerimientos que va utilizar el robot, es decir el número de entrada digitales, el voltaje y amperaje que soportarán los elementos que se le van agregar al robot, cantidad de canales para PWM, la clase de conexión a la fuente de energía y otras características que necesite el robot al armar.

Placas de circuito comerciales

En el mercado existen innumerables placas de circuito impreso que desempeñan tareas relacionadas con la robótica, siendo las más atractivas para este campo las controladoras de motores y servomotores.

- **La placa SSC32** permite el control de hasta 32 servomotores simultáneamente a través de un puerto serie RS232.
- **La placa ELECTAN** permite el control de hasta 28 servomotores simultáneamente a través de un puerto USB o serie (RS232). Además contiene un puerto sd card.

- **La placa de la I2C** permite el control de hasta 21 servomotores simultáneamente a través de un bus I2C12.
- **La placa Mega Arduino** posee 54 pines, 14 de estos tiene salida PWM para posibles conexiones siendo esta la tarjeta de mayor pines. También se llegó a comparar con la tarjeta arduino Uno que contiene 14 puertos de entrada, la cual no son suficientes para el requerimiento del prototipo.

A continuación en la figura (16) se muestra algunas de estas placas controladoras de servomotores, sus características dentro del lenguaje de códigos C++.

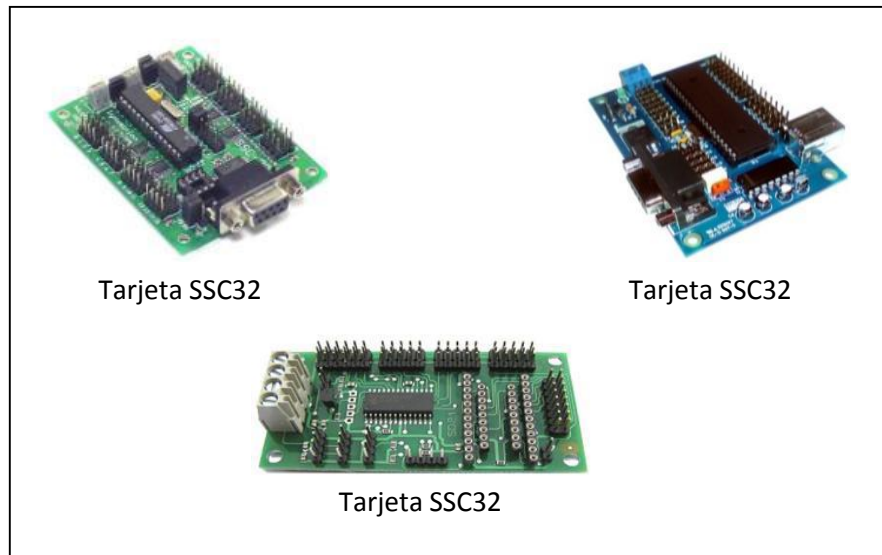


Figura 16: Placas Controladoras

Fuente: Páginas Web

Estas placas suelen ser utilizadas para la sincronización de múltiples servomotores en proyectos de robótica móvil con patas y en manipuladores (brazos robóticos).

Placas de control

Las principales familias de micro controladores existentes en el mercado actualmente son las siguientes:

- Atmel²: AVR mega, AVR tiny, 89Sxxxx, AVR32 y AT91SAM.
- Microchip³: PIC10F, PIC16F, PIC18F, PIC24F, PIC30F, PIC32, dsPIC30F y dsPIC33F.
- Freescale⁴ (antes Motorola): 68HCxx, 68k, PowerPC y HCS08.
- Intel: 8048, 8051, MCS96 y MXS296.
- Renesas⁵: H8, H8 tiny, H8S, R8C, M16C, H8SX, R32C, M32C y SHx.
- ST⁶: STMSS, ST6, ST7, uPSD y ST10

Todas estas contienen dispositivos muy similares, pudiendo ser programados en Ensamblador, Visual Basic, C, o incluso C++.

2.2.8. Sensores

Al momento de utilizar un robot para que realice una tarea adecuada es necesario que el robot conozca su propio estado así como el entorno que lo rodea. Un sensor es aquel dispositivo eléctrico y mecánico capaz de detectar magnitudes físicas llamadas variables de instrumentación y transformarlas en variables eléctricas estas son valores que pueden ser medidos. Las variables de instrumentación pueden ser:

Temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza torsión, humedad, movimiento, pH, etc.

Una magnitud eléctrica pueden ser: resistencia eléctrica (como en una RTD), una capacidad eléctrica (como en un sensor de humedad), una tensión eléctrica (como en un termopar), una corriente eléctrica (como en un fototransistor), etc.

Características de un Sensor

Entre las características técnicas de un sensor destacan las siguientes:

- **Rango de medida:** dominio en la magnitud medida en el que puede aplicarse el sensor.
- **Precisión:** es el error de medida máximo esperado.
- **Offset o desviación de cero:** valor de la variable de salida cuando la variable de entrada es nula. Si el rango de medida no llega a valores nulos de la variable de entrada, habitualmente se establece otro punto de referencia para definir el offset.
- **Linealidad o correlación lineal.**
- **Sensibilidad de un sensor:** relación entre la variación de la magnitud de salida y la variación de la magnitud de entrada.
- **Resolución:** mínima variación de la magnitud de entrada que puede apreciarse a la salida.
- **Rapidez de respuesta:** puede ser un tiempo fijo o depender de cuanto varíe la magnitud a medir. Depende de la capacidad del sistema para seguir las variaciones de la magnitud de entrada.
- **Derivas:** son otras magnitudes, aparte de la medida como magnitud de entrada, que influyen en la variable de salida. Por ejemplo, pueden ser condiciones ambientales, como la humedad, la temperatura u otras como el envejecimiento del sensor.
- **Repetitividad:** error esperado al repetir varias veces la misma medida.

Áreas de Aplicación:

- Robótica,
- Industria automotriz,
- Industria aeroespacial,
- medicina.

Tipos de Sensores

En la tabla (1) se indican algunos tipos y ejemplos de sensores electrónicos.

Magnitud	Transductor	Característica
Posición lineal y angular	Potenciómetro	Analógica
	Encoder	Digital
	Sensor Hall	Digital
Desplazamiento y deformación	Transformador diferencial de variación lineal	Analógica
	Galga extensiométrica	Analógica
	Magnetoestrictivos	A/D
	LVDT	Analógica
Velocidad lineal y angular	Dinamo tacométrica	Analógica
	Encoder	Digital
	Detector inductivo	Digital
	Servo-inclinómetros	A/D
	RVDT	Analógica
	Giróscopo	
Aceleración	Acelerómetro	Analógico
	Servo-acelerómetros	
Fuerza y par (deformación)	Galga extensiométrica	Analógico
	Triaxiales	A/D
Presión	Membranas	Analógica
	Piezoeléctricos	Analógica
	Manómetros Digitales	Digital
Temperatura	Termopar	Analógica
	RTD	Analógica
	Termistor NTC	Analógica
	Termistor PTC	Analógica
	[Bimetal - Termostato	I/O
Sensores de presencia	Inductivos	I/O
	Capacitivos	I/O
	Ópticos	I/O y Analógica
Sensores táctiles	Matriz de contactos	I/O
	Piel artificial	Analógica
Visión artificial	Cámaras de video	Procesamiento digital
	Cámaras CCD o CMOS	Procesamiento digital
Sensor de proximidad	Sensor final de carrera	
	Sensor capacitivo	Analógica
	Sensor inductivo	Analógica
	Sensor fotoeléctrico	Analógica
Sensor acústico (presión sonora)	micrófono	Analógica
Sensores de acidez	ISFET	
Sensor de luz	fotodiodo	Analógica
	Fotorresistencia	Analógica
	Fototransistor	Analógica
	Célula fotoeléctrica	Analógica
Sensores captura de movimiento	Sensores inerciales	

Tabla 1: Tipo de Sensores

Fuente: Introducción a los Sensores

2.2.9. Lenguaje de Programación

Es un lenguaje formal diseñado para expresar procesos que pueden ser utilizados para definir una secuencia de instrucciones para su procesamiento por un ordenador. Usados para crear programas que controlen comportamientos físico y lógico de una máquina, para expresarlo mediante un código fuente. Al código fuente a utilizar se prueba, se depura y se compila.

El lenguaje de Codificación de la plataforma de arduino de software libre, utiliza un lenguaje claro y fácil de codificar, para la ampliación de la codificación se utiliza librerías de C++, se representan algoritmos lógicos necesarios para el control de los movimientos mecánicos de robots, también de los procesos que se llevaran a cabo para establecer rutas de la trayectorias para cumplir objetivos de funcionamiento. Esta plataforma contiene como interfaz de usuario al monitor serial perteneciente al software de arduino.

2.3. Variables

En la tabla (2) se muestra la variable independiente del proyecto con sus respectivas características: definición, dimensiones, indicadores, instrumento.

TEMA: IMPLEMENTACIÓN DE UN PROTOTIPO DE UN ROBOT MÓVIL TIPO HEXÁPODO TELEDIRIGIDO PARA LA CONTRIBUCIÓN DE LA LIMPIEZA APLICANDO TECNOLOGÍA OPEN SOURCE					
Variable	Definición	Dimensiones	Indicadores	Instrumento	
Implementación de un prototipo de un robot móvil tipo hexápodo teledirigido	Robot Manipulador Zoomórfico tipo hexápodo que tendrá adicional un pinza robótica para coordinar movimientos para la obtención de un objeto.	Construcción Mecánica	Estructura de robot hexápodo: Estados del Desplazamiento	Experimento: Estado caminar hacia adelante	
				Experimento: Estado girar hacia la derecha	
				Experimento: Estado girar hacia la izquierda	
		Construcción Eléctrica	Pinza robótica	Tarjeta adaptable	Construcción de la tarjeta adaptable
				Distribución de pines	Tabla demostrativa
				Pinza robótica	Construcción de pinza robótica con servomotor
Construcción de servomotor con sensor ultrasónico					

Tabla 2: Variable Independiente

Fuente: Adriana Hidalgo Panchana

En la tabla (3) se muestra la variable dependiente del proyecto con sus respectivas características: definición, dimensiones, indicadores, instrumento.

TEMA: IMPLEMENTACIÓN DE UN PROTOTIPO DE UN ROBOT MÓVIL TIPO HEXÁPODO TELEDIRIGIDO PARA LA CONTRIBUCIÓN EN LA LIMPIEZA APLICANDO TECNOLOGÍA OPEN SOURCE				
Variable	Definición	Dimensiones	Indicadores	Instrumento
Tendrá una eficiencia del 80 % en la identificación y recolección del objeto	La tecnología open source aplicada a un robot que efectúa la recolección de objetos desperdiciados en el laboratorio de electrónica y telecomunicaciones	Sistema de control	Software de control	Pruebas de Detección
		Sistema de Fuerza	Pinza Robótica	Prueba Final de robot hexápodo
			Desplazamiento	Calculo tiempo vs Velocidad
				Calculo Tiempo vs 1 Paso
Calculo Distancia vs 1 Paso				

Tabla 3: Variable Dependiente

Fuente: Adriana Hidalgo Panchana

2.4. Métodos e Instrumentos de Investigación.

El desarrollo para adquirir los datos para la “implementación de un prototipo de un robot móvil tipo hexápodo teledirigido para la contribución en la limpieza aplicando tecnologías open source”, para el uso en el laboratorio de electrónica y telecomunicaciones, está caracterizado por cuatro etapas en donde se definirán métodos y técnicas de investigación para la obtención de información. Los métodos detallados a continuación serán utilizados en las etapas de obtención de datos.

Las siguientes etapas para el desarrollo de adquisición de datos utilizadas son:

Etapas 1

Antecedentes de Investigación.- En esta etapa es buscar información preliminar del proyecto tanto teórica como práctica sobre robots manipuladores, aplicación de robots hexápodos y robot de limpieza para realizar el desarrollo del tema.

Etapas 2

Definición de imposiciones del sistema.- En esta etapa se pretende describir las características, análisis técnico, económico y operativo de elementos de software y hardware del prototipo. Aquí se desarrollan las encuestas para conocer la aceptación del proyecto por parte de los estudiantes de la carrera Electrónica y telecomunicaciones.

Etapas 3

Diseño Técnico.- Se define qué tipo de estructura y de que material se plasmaría el robot hexápodo y cuál sería la interfaz gráfica y el desarrollo de esta.

Etapas 4

Codificación y pruebas.- Se plantea los experimentos y pruebas necesarias para corregir errores y tener resultados registrados para garantizar el funcionamiento adecuado del robot hexápodo para la contribución en la limpieza del laboratorio de Electrónica y Telecomunicaciones.

2.4.1. Métodos de Investigación

Método Empírico

Método de Investigación Científica que tiene su fundamento en la lógica empírica, el más utilizado en el área de las ciencias descriptivas y ciencias sociales. El método empírico se deriva de un griego antiguo ΕΜΝΕΙΡΊΑ que significa de experiencias y pruebas. Cuando se realice un experimento, por consecuencia los resultados son obtenidos de las pruebas y sus errores. Dichos métodos revelan las relaciones esenciales y las características fundamentales del tema de estudio, mediante procesos prácticos con el tema y diversos medios de estudio. Método empleado en la etapa 1 y 2.

Método Sistemático

Método que se radica en identificar reglas, sucesos y estándares que permitirán comprender sistemáticamente el contexto de un tema. Método empleado en la etapa 3 y 4.

2.4.2. Técnicas de Investigación

Las técnicas de investigación son un conjunto de procedimientos y medios que se utilizan para que el método sea activo, las técnicas que se aplicaron a este proyecto fueron: la observación, análisis de contenidos y encuestas realizadas a los estudiantes de la carrera Electrónica y Telecomunicaciones.

La observación

En la observación se instauró el uso constante de nuestros sentidos en la búsqueda de información que se necesita para poseer más información acerca del tema, es decir, consiste en conocer el contexto mediante visualización del funcionamiento de otros robots hexápodos para determinar el sistema de movimiento que hay que dar para la recolección de basura.

Encuestas

Es la búsqueda sistemática de información mediante preguntas, en este caso, una muestra de la población de estudiantes de la carrera Electrónica y Telecomunicaciones para la obtención de datos con respecto a la elaboración del robot hexápodo.

Estudio de contenidos

El estudio de contenidos es una parte primordial como técnica al desarrollo del proyecto “Implementación de un prototipo de un robot móvil tipo hexápodo teledirigido para la contribución en la limpieza aplicando tecnologías open source”, ya que esta información será la base al momento de la ejecución con respecto a cálculos, parámetros, información cuantitativa. El análisis documental se llevó a cabo mediante la extracción y comparación de información de libros, tesis, páginas web, videos tutoriales.

2.5. Términos básicos

Adquisición de Datos: tomar datos o señales del medio físico para transformarlas en movimientos a través de un robot.

GDL: siglas de Grados de Libertad.

Inteligencia Artificial: programación que realiza tareas similares que solo la inteligencia humana lo podría realizar.

Locomoción: es la estructura del cuerpo para realizar movimientos.

Teledirigido: robot que es controlado a distancia.

PWM: modulación por Ancho de Pulso en cuando se cambia el ciclo de la señal.

Microcontrolador: es un elemento electrónico, un circuito de memoria programable.

Mhz: abreviación de megahercio, utilizado para medir la frecuencia.

CA: Corriente Alterna.

CC: Corriente Continua.

Conversor A/D: convierte señal analógica a señal digital.

Sobre Oscilación: Se define como la amplitud de la primera oscilación en porcentaje sobre el valor final de referencia, en nuestra gráfica se ve representada por las líneas verdes y se denomina SO.

Rectificador: convierte corriente alterna en corriente continua.

Fusibles: elemento electrónico que se instala en circuitos para proteger a los elementos cuando existe un cortocircuito o sobre carga.

Termopar: es un transductor formado por la unión de dos metales distintos que produce una diferencia de potencial muy pequeña.

Energía Hidráulica: aquella que se obtiene del aprovechamiento de las energías cinética y potencial de la corriente del agua.

Energía Neumática: es la tecnología que emplea el aire comprimido como modo de transmisión de la energía necesaria para mover y hacer funcionar mecanismos.

CAPÍTULO 3

ANÁLISIS

3. ANÁLISIS

Como paso consecutivo al desarrollo del tema se presentará los procesos, especificaciones técnicas, tablas de viabilidad correspondientes para llevar la ejecución la implementación del prototipo.

3.1. Diagrama de proceso

En la figura (17) se expresa los diferentes sistemas que se asignaron para la elaboración del proyecto.

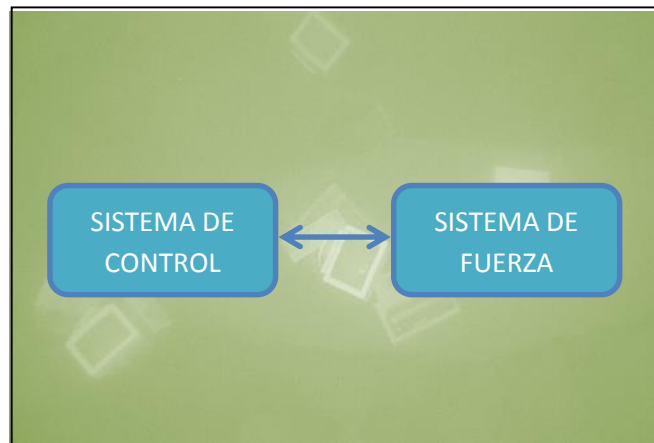


Figura 17: Diagrama de Procesos del Sistema

Fuente: Adriana Hidalgo Panchana

Sistema de control

El sistema de control es la parte que recibe acciones externas o variables de entrada, y la respuesta a estas acciones externas se las denomina variable de salida.

Como se muestra en la figura (18) las acciones recibidas en la variable de entrada tienen variables de control que se puede manipular y perturbaciones que no se controlan.



Figura 18: Diagrama del Sistema de Control

Fuente: Adriana Hidalgo Panchana

El sistema de control caracterizado porque posee un número determinado de elementos que intervienen en el manejo de señales emitidas para realizar una función; a diferencia del sistema de fuerza que será el encargado del ensamblaje de la estructura física y elementos necesarios para lograr un movimiento específico ordenado por el sistema de control.

El objetivo principal del sistema en general es que mediante la implementación y manipulación de los elementos de los sistemas de control y de fuerza se obtenga una respuesta de funcionamiento del robot.

Sistema de Fuerza

El Sistema de fuerza describe a los diferentes elementos y estructura que son necesarios para que el robot logre moverse.

En la figura (19) se muestra el sistema de fuerza y los elementos que contiene.

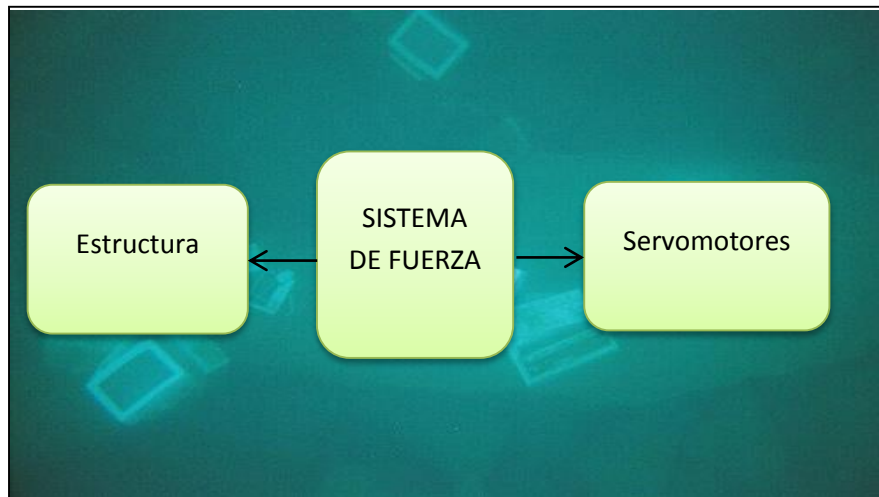


Figura 19: Diagrama del Sistema de Fuerza

Fuente: Adriana Hidalgo Panchana

El sistema de fuerza comprende elementos que servirán para manifestar la respuesta de la señal de salida, teniendo como resultado el desplazamiento del robot hexápodo, los causantes del movimiento son las articulaciones de la estructura y los servomotores, que darán la posición para realizar un estado establecido en la codificación.

3.1.1. Descripción funcional de los procesos

SISTEMA DE CONTROL

Es el encargado de recibir y enviar órdenes para su ejecución cuando se haya comparado los resultados del exterior con la codificación. Así mismo es el encargado de controlar a través de la codificación las funciones y la coordinación de los movimientos que va a realizar el robot. La parte de codificación se la realiza por medio del lenguaje de la plataforma arduino, por tal motivo, la tarjeta de control es importante ya que tiene la facultad de cambiar la función del robot.

También planifica las órdenes en trayectorias para que el robot, avance, gire a la derecha o izquierda. El procedimiento de trayectoria comienza con el movimiento del sensor de distancia que se encuentra colocado sobre un servomotor, el sensor

indica la distancia y el servomotor brinda el ángulo de dirección donde fue localizado el objeto, para que luego el robot se desplace y agarre el objeto.

Las Trayectorias determinadas son:

Girar hacia la izquierda o hacia la derecha.

Avanzar hacia adelante y recoger el objeto.

Sin el sistema de control el robot no obtendría movimientos coordinados porque a pesar de estar alimentado correctamente con la fuente energía sin un código de control previo sobre el arduino, no efectuaría algún movimiento útil, ya que el objetivo de mover un robot es que efectúe una actividad en concordancia y sin el sistema control el objetivo no se cumpliría.

Sistema de Control de Lazo Abierto

En la siguiente figura (20) se muestra el procedimiento a seguir en el sistema de control de Lazo Abierto correspondiente al proyecto.

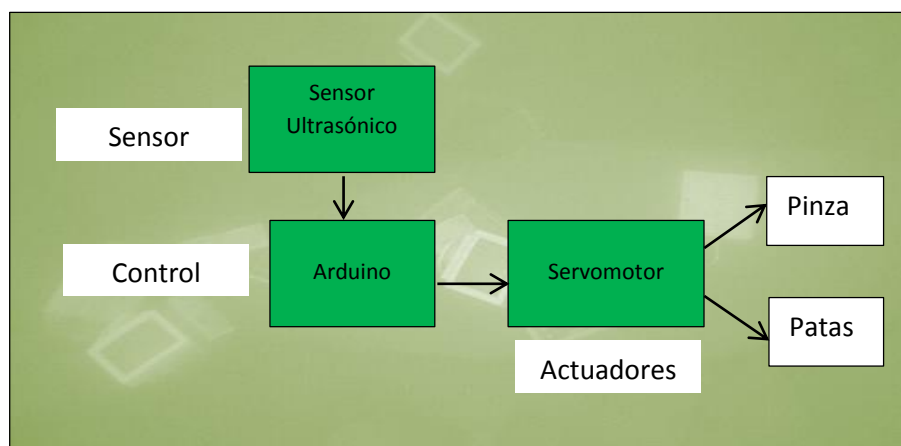


Figura 20: Diagrama de bloques Sistema de control de Lazo Abierto

Fuente: Adriana Hidalgo Panchana

Llamado sistema de lazo abierto o sistema sin realimentación. Es utilizado en algunas implementaciones robóticas para llevar a cabo procesos en un determinado periodo de tiempo. El robot hexápodo luego de realizar la tarea de identificar y recoger el objeto dentro este detiene el proceso

SISTEMA DE FUERZA

Parte mecánica que conforman la estructura del robot. Los elementos que proveen la movilidad a las articulaciones son los actuadores situados conjuntamente con la estructura las cuales son: el cuerpo, las patas y la pinza mecánica ubicada en la parte frontal para que realice la tarea de recoger el objeto.

El cuerpo está diseñado para que las patas mantengan una cierta distancia para evitar que se choquen entre sí, ya que el cuerpo tiene que soportar el sobrepeso de los otros elementos adicionales sobre él.

Las patas son resistentes estas poseen articulaciones conformadas por los actuadores, cada pata sostiene a tres servomotores. Las dimensiones son exactas ya que mantienen la distancia para poder movilizarse. Los actuadores ayudan al movimiento del robot, el correcto desplazamiento del robot depende de una buena codificación para el control de los servomotores.

3.2. Identificación de requerimientos

Los requerimientos al momento de implementar el trabajo son:

- Al seleccionar la placa de control se debe escoger la que contenga los puertos de entrada/salida suficiente para conectar los distintos elementos electrónicos que se implementaran en el prototipo del robot hexápodo.
- El servomotor debe tener la suficiente fuerza de torque para soportar el peso de los elementos electrónicos y hacer que el robot hexápodo se ponga de pie.
- Al momento de determinar un objeto se utilizará un sensor ultrasónico HC-SR04.
- La alimentación de energía, debe basarse en los requerimientos técnicos de cada dispositivo, es decir para añadir la batería con la potencia

necesaria se necesita conocer la cantidad de amperaje total de los elementos del prototipo a implementar.

- Para la interfaz gráfica se utilizara un programa fácil de manejar y visualizar en este caso la interfaz del usuario será el software de arduino.

3.3. Análisis del Sistema

El análisis técnico, económico y operativo de la investigación de “Implementación de un prototipo de un robot móvil tipo hexápodo teledirigido para la contribución de la limpieza aplicando tecnología open source” es recomendado ya que no solo colaborará con la limpieza del laboratorio sino también se lo podrá utilizar como un instrumento tecnológico hacia el aprendizaje de la robótica.

3.3.1. Análisis Técnico

Se analizará los elementos tanto para programación, simulación, hardware y software a utilizar con sus respectivas descripciones.

Cantidad	Hardware	Descripción
1	CPU(ordenador)	Procesador Inter(R) Core™ i3-3240 cpu@ 3.40 Ghz, 4096 MB de memoria ram, 931 GB de Disco Duro
1	Impresora	Canon MG2400 series Printer

Tabla 4: Hardware a Utilizar

Fuente: Adriana Hidalgo Panchana

Cantidad	Software	Descripción
1	Sistema Operativo	Windows 8.1 Single Languaje 64 bits
1	Microsoft Office	Microsoft Office Professional Plus 2010
1	Proteus	Proteus 8 Profesional
1	Software para Arduino	1.0.6

Tabla 5: Software a Utilizar

Fuente: Adriana Hidalgo Panchana

Cantidad	Hardware	Descripción
1	Tarjeta de control	Arduino Mega 2560
1	Tarjeta adaptable de control	Tarjeta con pines para conectar de la tarjeta arduino a los servomotores
1	Pinza Robótica	Pinza mecánica para recoger el objeto.
1	Estructura metálica	soporte metálico
1	Servomotores para patas	de 15 kilogramos de fuerza
1	Servomotor para la búsqueda	de 2 kilogramos de fuerza
1	Servomotor para el agarre	de 10 kilogramos de fuerza
1	Sensor de distancia	SRF05 medirá la distancia
1	Fuente de poder	batería tipo lipo de 5 voltios
1	Cable USB	que conecta el arduino con el ordenador
1	Multímetro	medidor de voltaje y de corriente
1	Cables	conectar un elemento con otro
	Cinta aislante	utilizada para proteger en cierto lugares del circuito

Tabla 6: Software a Implementar

Fuente: Adriana Hidalgo Panchana

3.3.2. Análisis Económico

En este apartado se describirán los precios de cada uno de los elementos de la implementación hardware, software y elementos adicionales para realizar la presentación del proyecto.

Cantidad	Hardware	Valor Unit.	Total
1	CPU(ordenador)	\$ 1.300,00	\$ 1.300,00
1	Impresora	\$ 65,00	\$ 65,00
TOTAL			\$ 1.365,00

Tabla 7: Costo de Hardware a utilizar

Fuente: Adriana Hidalgo Pachana

Cantidad	Software	Valor	Subtotal
1	Sistema Operativo	\$ 20,00	\$ 20,00
TOTAL			\$ 20,00

Tabla 8: Costo de software a utilizar

Fuente: Adriana Hidalgo Panchana

Cantidad	Elementos	Valor	Subtotal
1	Tarjeta de control	\$ 75,00	\$ 75,00
1	Tarjeta adaptable de control	\$ 50,00	\$ 50,00
1	Pinza robótica	\$ 44,00	\$ 44,00
1	Estructura metálica	\$ 400,00	\$ 400,00
18	Servomotores para patas	\$ 29,00	\$ 522,00
1	Servomotor para la búsqueda	\$ 11,00	\$ 11,00
1	Servomotor para el agarre	\$ 20,00	\$ 20,00
1	Sensor de distancia	\$ 7,50	\$ 7,50
1	Fuente de poder	\$ 40,00	\$ 40,00
1	Cables	\$ 5,00	\$ 5,00
1	Cinta aislante	\$ 1,00	\$ 1,00
TOTAL			\$ 1.175,50

Tabla 9: Costo de Hardware a Implementar

Fuente: Adriana Hidalgo Panchana

Cantidad	Materiales	Valor	Subtotal
1	Materiales adicionales	\$ 70,00	\$ 70,00
1	Internet	\$ 30,00	\$ 30,00
1	Tinta de impresora	\$ 25,00	\$ 25,00
TOTAL			\$ 125,00

Tabla 10: Costo Materiales a Utilizar

Fuente: Adriana Hidalgo Panchana

Cantidad	SOFTWARE	Valor	Subtotal
1	HARDWARE	\$ 1.365,00	\$ 1.365,00
1	SOFTWARE	\$ 20,00	\$ 20,00
1	ELEMENTOS PARA LA IMPLEMENTACION	\$ 1.175,50	\$ 1.175,50
1	ELEMENTOS ADICIONALES	\$ 125,00	\$ 125,00
TOTAL			\$ 2.685,50

Tabla 11: Costos Totales

Fuente: Adriana Hidalgo Panchana

Este proyecto es factible para utilizarlo en el laboratorio; ya que al realizar la comparación con otros robots de limpieza como los de ruedas de tipo succión, la inversión es más elevada que la implementación del robot hexápodo, también la inversión en este robot de limpieza servirá como instrumento de estudio que contribuirá al aprendizaje de los estudiantes.

3.3.3. Análisis Operativo

Con los análisis anteriores técnicos y económicos veremos si el proyecto es viable realizarlo y qué beneficios se obtendrían si se llegara a implementar.

Se utilizará una técnica de investigación como son las encuestas, para verificar que la mayoría de las personas están de acuerdo en que el robot contribuirá a la limpieza

del laboratorio de Electrónica y Telecomunicaciones. Las preguntas, tabulación y resultados de las encuestas se exponen en el Anexo 2. Es necesario conocer la muestra que representa a la población que se utilizó como estudio, por tal motivo se utilizó las siguientes fórmulas.

$$n = \frac{N}{e^2(N - 1) + 1}$$

Dónde:

N= tamaño de la población, 40

e= Error de estimación, 5%

n= Tamaño de la muestra

$$n = \frac{40}{0,05^2(40 - 1) + 1}$$

$$n = \frac{2940}{1,0875}$$

$$n = 36 \text{ Estudiantes}$$

El total de estudiantes encuestados fue de 36 estudiantes, este es el tamaño de la muestra del cual se tomará sus respuestas como guía para verificar si el prototipo del robot hexápodo es realizado para beneficio.

CAPÍTULO 4

DISEÑO

4. Diseño

En este capítulo se describe el funcionamiento del robot tanto del sistema de control y sistema de fuerza, también la descripción de las características de los elementos a implementar y el desarrollo de partes del código fuente que controlan los movimientos del robot. En la figura (21) se describe el diagrama de bloques de los sistemas a desarrollar e implementar que permitirán cumplir con el objetivo del tema de investigación.

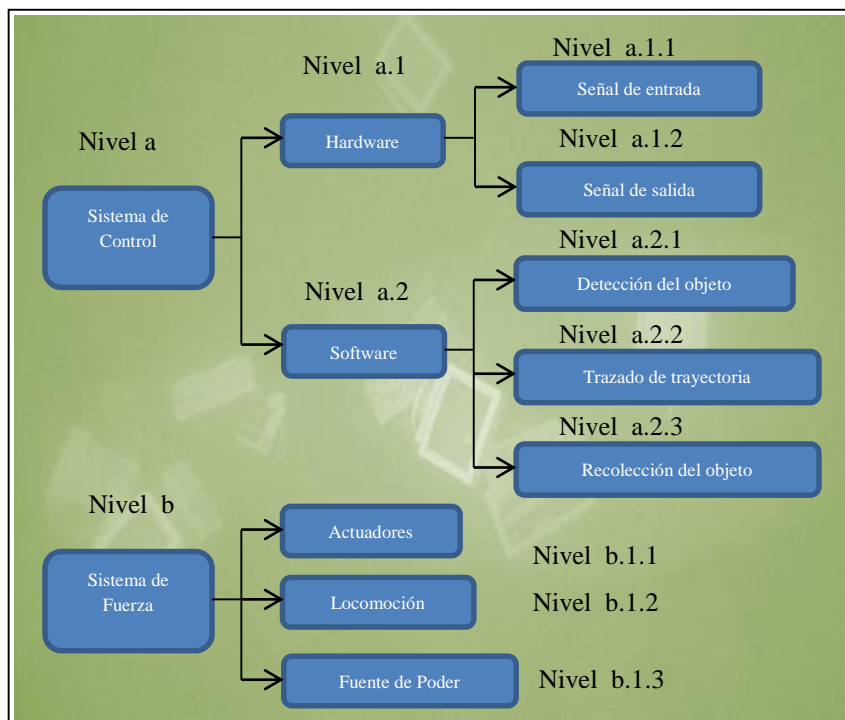


Figura 21: Diagrama de Bloques

Fuente: Adriana Hidalgo Panchana

4.1. Arquitectura de la solución

El sistema principal es la implementación del prototipo del robot móvil tipo hexápodo teledirigido para contribuir a la limpieza, para que trabaje en un territorio firme, plano y sin obstáculos, el proceso tiene como partes sistema de control y sistema de fuerza.

4.1.1. Desarrollo del Sistema de Control

Nivel a.1 Hardware

En la figura (22) se muestra el conjunto de elementos que conforman el hardware del sistema de control.

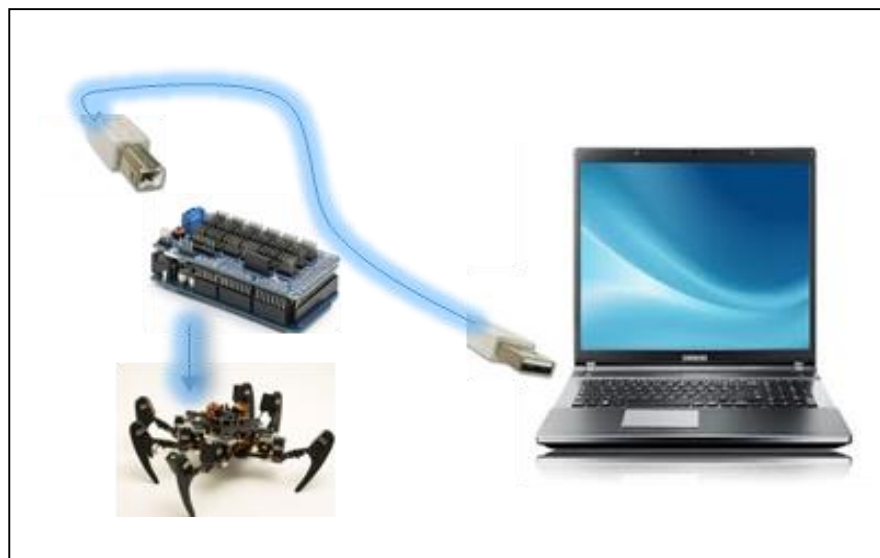


Figura 22: Sistema de Control

Fuente: Adriana Hidalgo Panchana

Ordenador Externo

El ordenador es el medio donde se ejecuta una capa de procedimientos de control desde él envió de órdenes simples como el posicionamiento de servomotores hasta la coordinación de la trayectoria del robot.

Interfaz

La interfaz en la que va a monitorear las órdenes entre el ordenador y la tarjeta de control, este monitoreo es visualizada por el usuario.

Tarjeta Adaptable

Ubicada sobre la tarjeta de control. Creada con la necesidad de tener un medio donde conectar todos los pines que corresponden al sensor y a los servomotores, de esta manera distribuir adecuadamente el suministro de energía proveniente de la tarjeta de control y de la fuente externa. Esta Placa está conformada por una bornera y un conjunto de espadines, los cuales encajen debidamente sobre la tarjeta mega arduino. Las mediciones de esta tarjeta fueron de espadín a espadín con valores milimetrados, para su simulación se utilizó el Simulador Proteus 8 Professional. En la figura (23) se muestra el circuito con cada una de las dimensiones de cómo va a quedar la impresión en la placa.

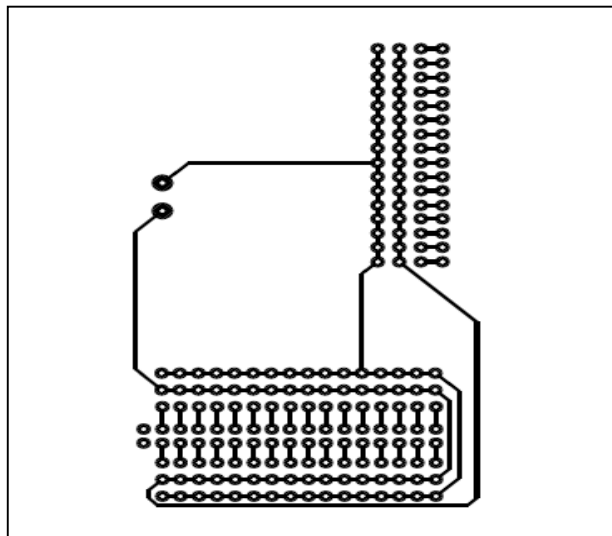


Figura 23: Circuito impreso de la Tarjeta Adaptable de Control

Fuente: Adriana Hidalgo Panchana (Autor)

Tarjeta de control

Situada sobre el cuerpo del robot, genera señales que sirven para el posicionamiento de los servomotores a través de los datos enviados desde el ordenador.

En la figura (24) se muestra la tarjeta mega arduino con cada una de la ubicación de los pines.

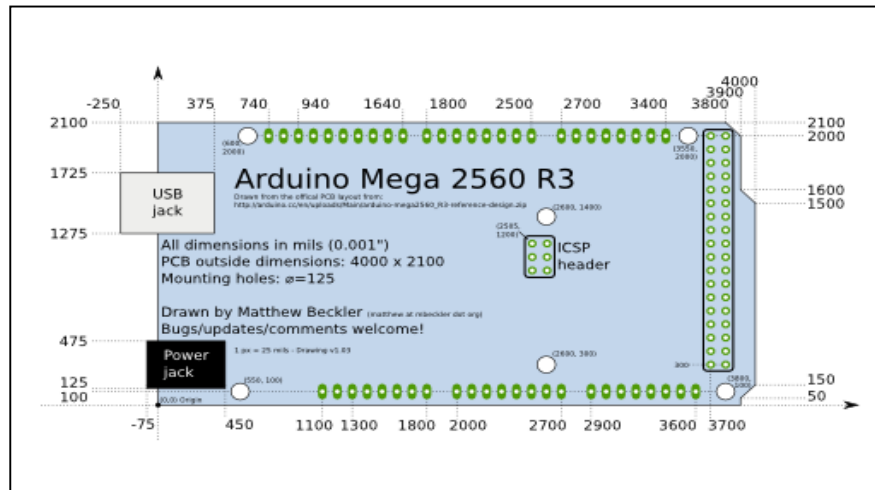


Figura 24: Vista física de Arduino

Fuente: Arduino.cc

Esta placa es la encargada de la inicialización o reseteo, la adquisición de datos y ejecución.

Inicialización Identifica todos los elementos, sensor y servomotores que son conectados a la placa.

Adquisición de datos toma los datos en la memoria o código y los compara con los datos receptados desde el exterior.

Ejecución Analiza y envía una señal de salida hacia los servomotores para que se muevan y así desplazar el cuerpo del robot.

Mega Arduino

Para este proyecto se ha elegido la placa mega arduino 2560, el elemento principal es el microcontrolador Atmega 2560, un chip sencillo y de bajo coste que permite el desarrollo de múltiples diseños. Dentro de la diversidad existente, es la placa más grande en la gama de arduino, con el mayor número de pines digitales y analógicos. La característica fundamental por la que se eligió la placa mega arduino es el número de puertos disponibles, con un total de 54 pines, de los cuales se van a utilizar 22. A pesar de la gran cantidad de pines, el arduino mega tiene unas

dimensiones compatibles con la finalidad del proyecto, que será su incorporación a un robot de servicios. Estas dimensiones son de 10,16 cm x 5,33 cm. Algunas características de la tarjeta, aunque no se usen en este proyecto, se tienen en cuenta para futuras ampliaciones, ya que la tarjeta mega arduino ha sido manipulada anteriormente en prácticas de laboratorio.

En la figura (25) se muestra la tarjeta mega arduino. Esta placa es la encargada de la recepción y transmisión de datos para que el robot hexápodo realice las funciones determinadas.



Figura 25: Mega Arduino 2560

Fuente: Páginas Web

La placa controladora es la encargada de todas las funciones de bajo nivel del robot, desde las más básicas como la generación de señales PWM para el control de los servomotores hasta las más complejas como: la generación de secuencias de pasos, generación de posturas del robot y lectura del sensor de distancias por ultrasonidos, entre otros. En el Anexo 6 se muestra las características técnicas de la tarjeta mega Arduino.

Características

En la tabla (12) se muestra las características del mega arduino 2560.

Microcontrolador	ATmega2560
Voltaje de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límite)	6-20V
Pines E/S digitales	54 (15 proporcionan salida PWM)
Pines de entrada analógica	16
Intensidad por pin	40mA
Intensidad en pin 3.3V	50mA
Memoria Flash	256 KB de las cuales 8 KB las usa el gestor de arranque(bootloader)
SRAM	8 KB
EEPROM	4KB
Velocidad de reloj	16 MHz
Puertos serie	4

Tabla 12: Características del Mega Aduino

Fuente: Adriana Hidalgo Panchana

Nivel a.1.1 Señal de Entrada:

Para conseguir que el robot realice la recolección del objeto de manera correcta, primero tiene que detectar una señal sobre lo exterior mediante el sensor ultrasónico. Este emitirá impulsos que se reflejan sobre el objeto, el sensor recibe el eco producido y lo convierte en señal eléctrica.

El dispositivo de la señal de entrada introduce datos a la tarjeta de control comparándolos con los códigos ingresados en la codificación.

Los movimientos del sensor serán realizados por el servomotor que lo contiene y el elemento que controla al servomotor es la tarjeta de control. En la siguiente figura (26) se muestra en Proteus la simulación de los elementos necesarios para que la señal de entrada se efectúe.

Los elementos utilizados son:

- 1 sensor ultrasónico
- 1 servomotor
- 1 Tarjeta mega arduino.

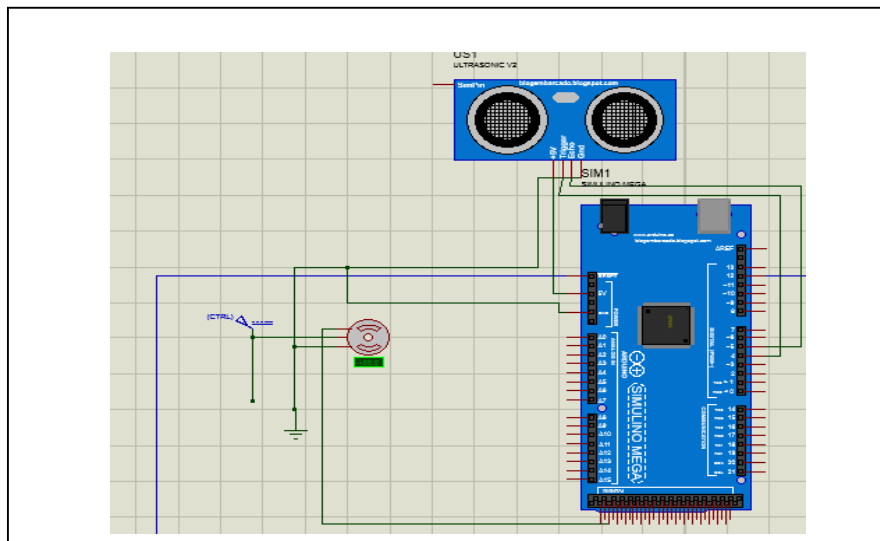


Figura 26: Simulación Arduino y Sensor Ultrasónico

Fuente: Adriana Hidalgo Panchana

Nivel a. 1.2 Señal de salida

Un objeto en el exterior es medido por el sensor y tiene como salida una señal eléctrica. Una vez que los elementos de la señal de entrada hayan tomado una medida y se hayan comparado con los códigos de programación de la tarjeta, este envía una señal de salida a los servomotores para que estos se desplacen según los estados que se le asigne.

El desplazamiento del robot es realizado por la estructura: cuerpo y extremidades que contiene en cada articulación un servomotor, el elemento que controla a los servomotores es la tarjeta arduino, que mediante los valores de la señal de entrada que recibe provocaran el movimiento de la estructura.

En la siguiente figura (27) se muestra la simulación en Proteus del circuito referente a la implementación de los elementos para que la señal de salida se lleve a cabo.

Los elementos utilizados son:

1 Tarjeta arduino mega 2560

18 servomotores que se utilizaran para el desplazamiento.

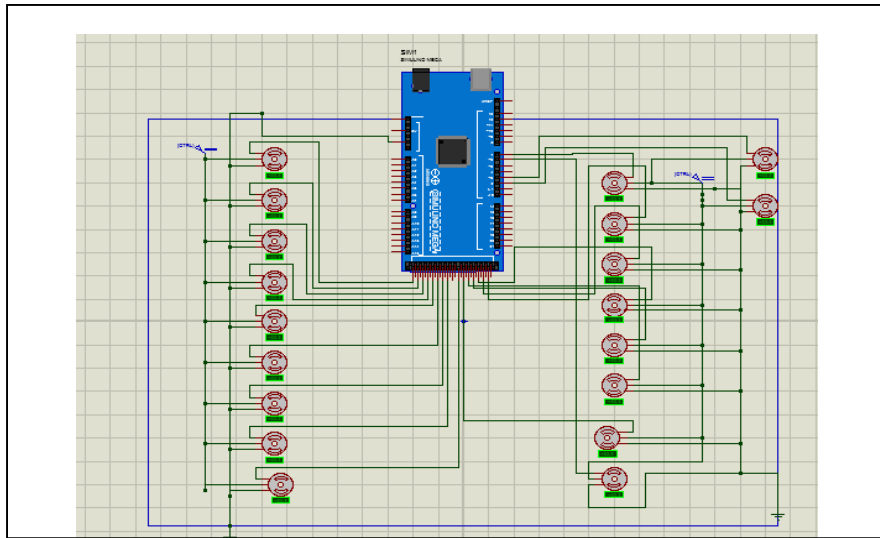


Figura 27: Simulación de la conexión de Arduino con Servomotores

Fuente: Adriana Hidalgo Panchana

Control de los actuadores

El control de los actuadores como respuesta de la señal de entrada será realizado mediante el código en arduino, primero se procede a declarar los pines de la tarjeta designado a cada uno de los 20 servomotores.

```

servo1.attach(22);      inicial 1           // declaración de pata 1
servo2.attach(24);      medio 1           // declaración de pata 1
servo3.attach(26);      final 1           // declaración de pata 1
servo4.attach(2);       inicio 2          // declaración de pata 2
servo5.attach(3);       medio 2          // declaración de pata 2
servo6.attach(4);       final 2 // declaración de pata 2
servo7.attach(5);       inicio3          // declaración de pata 3
servo8.attach(6);       medio 3          // declaración de pata 3
servo9.attach(7);       final 3 // declaración de pata 3
servo10.attach(28);     inicio 4         // declaración de pata 4
servo11.attach(30);     medio4          // declaración de pata 4
servo12.attach(32);     final 4         // declaración de pata 4
servo13.attach(46);     inicio 5        // declaración de pata 5
servo14.attach(48);     medio 5        // declaración de pata 5
servo15.attach(34);     final 5        // declaración de pata 5
servo16.attach(40);     inicio 6        // declaración de pata 6

```

```

servo17.attach(42);    medio 6           // declaración de pata 6
servo18.attach(44);    final 6          // declaración de pata 6
servo19.attach(36);           //servo de búsqueda
servo20.attach(47);           //servo de agarre

```

Luego de declarar cada servomotor, se procede a configurar comandos lógicos para el control de los movimientos utilizando el software de arduino. En los códigos de control se dan las funciones necesarias para realizar los movimientos de más alto nivel: levantamiento, avance, giros, reposo. Además, incluye todo lo relacionado con los "triángulos de apoyo". Las funciones declaradas para los movimientos individuales del robot son los siguientes:

```

void paro()
void adelante()
void derecha()
void izquierda()
void agache()
void servo_Move(int val)
void buscador()
void garra()

```

Cada una de estas funciones contiene algoritmos que al momento de llamarlos en la codificación, se ejecutarán.

A continuación se realiza la descripción de un estado de la estructura, la función “paro” la cual es utilizada cuando se requiere que el robot hexápodo se coloque de pie esto significa que todas las seis patas se estabilicen en estado vertical al mismo tiempo.

La instrucción servo. Write(a) genera una señal de pulso PWM que va hacer que gire el servomotor desde cualquier posición que se encuentre hacia la posición (a) que representa a un ángulo inicial, por lo tanto para que esta función “paro” funcione correctamente se asignan los valores de ángulos variables para cada servomotor y con esto se buscara una postura inicial para todos los servomotores, dando como resultado que el robot quedara en estado de espera y en pie. En el

código se utiliza tres servomotores por cada pata. A continuación se muestra la codificación para que el robot se encuentre en estado de pie:

```
void paro()  
{//pata 1  
servo1.write(60);  
servo2.write(120);  
servo3.write(120);  
//pata 2  
servo4.write(160); //giro izq  
servo5.write(130); //codo  
servo6.write(170); // final  
//pata 3  
servo7.write(160);  
servo8.write(130);  
servo9.write(120);  
//pata 4  
servo10.write(110); // izq  
servo11.write(35);  
servo12.write(70);  
//pata 5  
servo13.write(90);  
servo14.write(65);  
servo15.write(65);  
//pata 6  
servo16.write(0);  
servo17.write(65);  
servo18.write(65);  
}
```

El programa completo que controla todos los movimientos del robot se presenta en el anexo 4.

Nivel a.2 Software

En este nivel se describirá el desarrollo de los niveles de detección, trazado de trayectoria y reconocimiento del objeto.

A continuación se presenta el diagrama de flujo de la lógica de la compilación para los códigos aplicados en el software de arduino, se muestra en la figura (28).

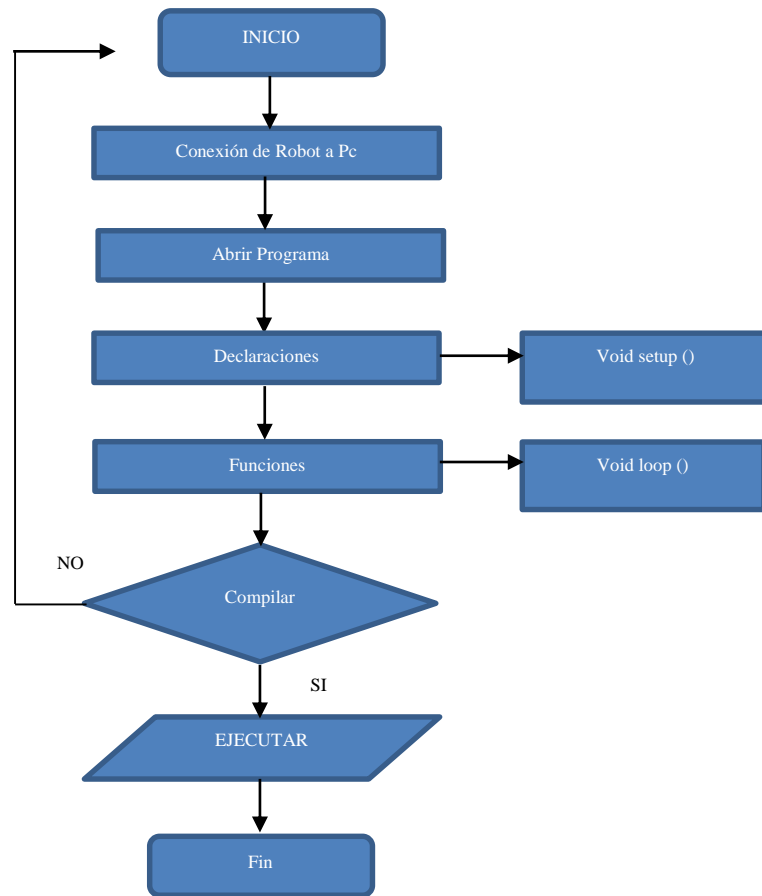


Figura 28: Diagrama de Compilación de Códigos en Arduino

Fuente: Adriana Hidalgo Panchana

Nivel a 2.1 Detección del objeto

En la parte frontal del robot se encontrara ubicado el conjunto del sensor ultrasónico y servomotor, que serán encargados de la detección. La codificación en arduino para el control de búsqueda se encargara de dar las órdenes para la detección del objeto.

Sensor Ultrasónico HC-SR04

Sensor que contiene cuatro pines de conexión: Trig, Echo, Gnd y Vcc. Trig es igual a la entrada de sensor, recibe un punto alto en 5v durante 10us, este valor

indica que se inicie y se transmita 8 ciclos de ráfagas de ultrasonidos a 40 KHz y se espera por la ráfaga reflejada. Cuando se detecta el ultrasonido en el receptor, el pin Echo se pondrá en alto de 5v y se pausará en un periodo proporcional a la distancia medida. En la figura (29) se muestra al sensor ultrasónico. Posee un gran rango de medición y de alta precisión al momento de detectar un objeto.



Figura 29: Sensor Ultrasónico

Fuente: Adriana Hidalgo Panchana

El sensor ultrasónico es el encargado de indicar la distancia en que la se encuentra algún objeto, en este caso el de la basura que será detectada hasta una distancia máxima de 50 cm y el sensor envía una señal a la tarjeta arduino para que la estructura del robot siga una ruta hacia el objeto.. A continuación en la tabla (13) se muestra las características del sensor ultrasónico.

Características del sensor ultrasónico:	
Alimentación:	5v
Corriente en Reposo:	< 2mA.
Corriente de trabajo:	15 mA.
Angulo eficaz:	< 15 grados.
Rango de distancia:	2cm – 400 cm/1” - 13ft.
Resolución:	0.3 cm.
Angulo de pulso del disparo de entrada:	10 us.
Dimensiones:	45 mm x 20 mm x 15 mm.

Tabla 13: Características de un Sensor Ultrasónico

Fuente: Geekfactory

En el Anexo 8 se encuentra las características técnicas del Sensor Ultrasónico. Las características del servomotor que contiene al sensor ultrasónico se detallan en la tabla (14):

Movimiento	Servomotor	Torque
Búsqueda	MG90S Micro Servo	2 Kg / cm

Tabla 14: Características del Servomotor

Fuente: Adriana Hidalgo Panchana

A continuación en la figura (30) se presenta el diagrama de flujos para la detección de un objeto.

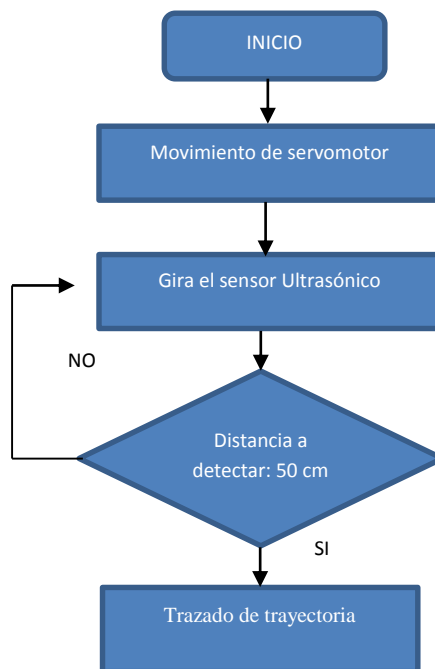


Figura 30: Diagrama de flujos del Detección

Fuente: Adriana Hidalgo Panchana

El diagrama de flujo se explica a continuación:

La detección del objeto se va a realizar por medio del sensor ultrasónico hasta una distancia máxima de 50 cm, el sensor gira a causa del movimiento del servomotor en las posiciones determinadas entre 40° y 130°, obteniendo 9 posiciones de posibles detecciones y que cumplirá la función de buscar el objeto y guardara la

información del grado de giro donde hubo la detección, el siguiente paso es dirigir hacia la izquierda, adelante o derecha del robot según las posiciones explicadas anteriormente.

En la figura (31) se muestra la dirección del sensor: caso a) el sensor ultrasónico detecta un objeto en la posición 4 para luego proceder a avanzar hacia adelante y b) el sensor ultrasónico detecta un objeto ubicado en las posiciones no céntricas, entonces el robot procede a girar en el sentido donde se encuentre el objeto.

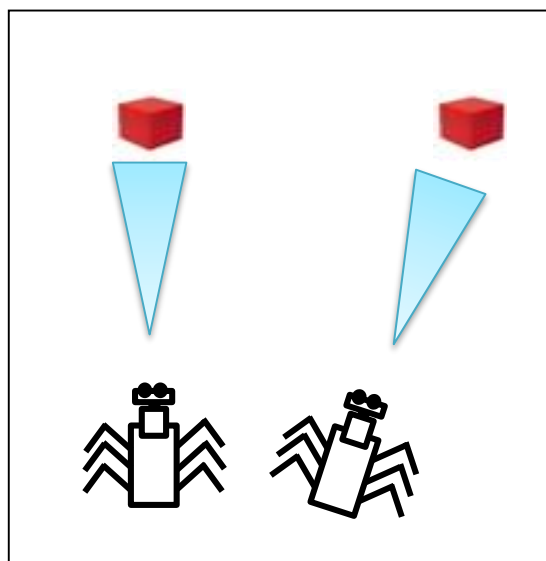


Figura 31: a) Avance hasta el objeto b) Giro hacia el objeto

Fuente: Adriana Hidalgo Panchana

Nivel a.2.2 Trazado de trayectoria

El ángulo donde se detecta el objeto es hacia donde se trasladará la estructura del robot, este proceso de desplazamiento se lleva a cabo cuando se envía la señal a la tarjeta de control, en donde comparará la distancia del exterior con la distancia designada en la codificación, seguido de esto se determina si el robot avanza hacia adelante, izquierda o derecha. El desarrollo del trazado de trayectoria, indica las rutas de selección que el robot hexápodo escogerá para desplazarse y determinar la trayectoria a seguir para recoger el objeto.

El movimiento del servomotor que estará ubicado en la parte frontal indicara la dirección de las posibles detecciones del lugar donde se encuentre el objeto, en conjunto con el sensor ultrasónico va a indicar la distancia a recorrer hacia el objeto, la información tomada del exterior la compara con las medidas en la codificación y si es la detección del objeto indicará cerrar la pinza y recoger el objeto. Para el trazado de trayectoria se muestra en la figura (32) la ubicación del sensor ultrasónico, la pinza y el trayecto que existirá hacia el objeto.

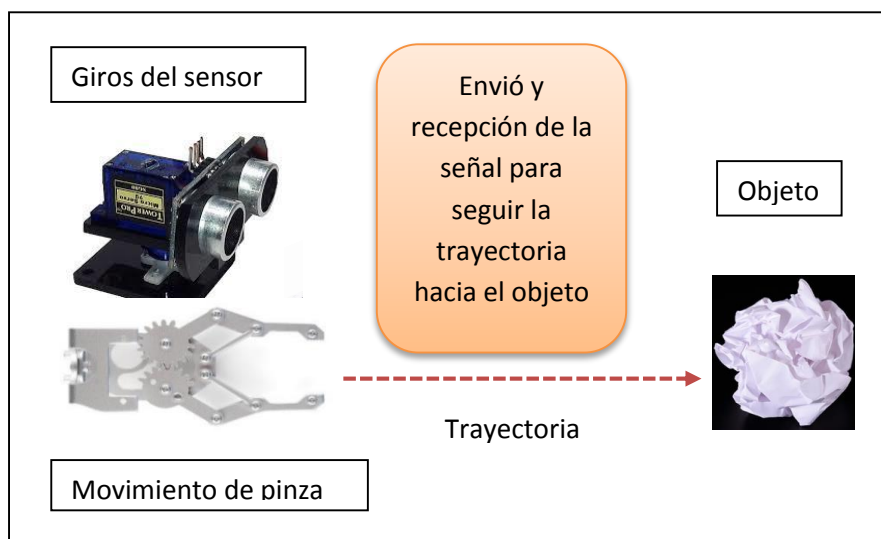


Figura 32: Trazado de trayectoria

Fuente: Adriana Hidalgo Panchana

A continuación se presenta el diagrama de flujo que contiene el trazado de la trayectoria hacia el objeto, en la figura (33) se muestra el procedimiento a seguir.

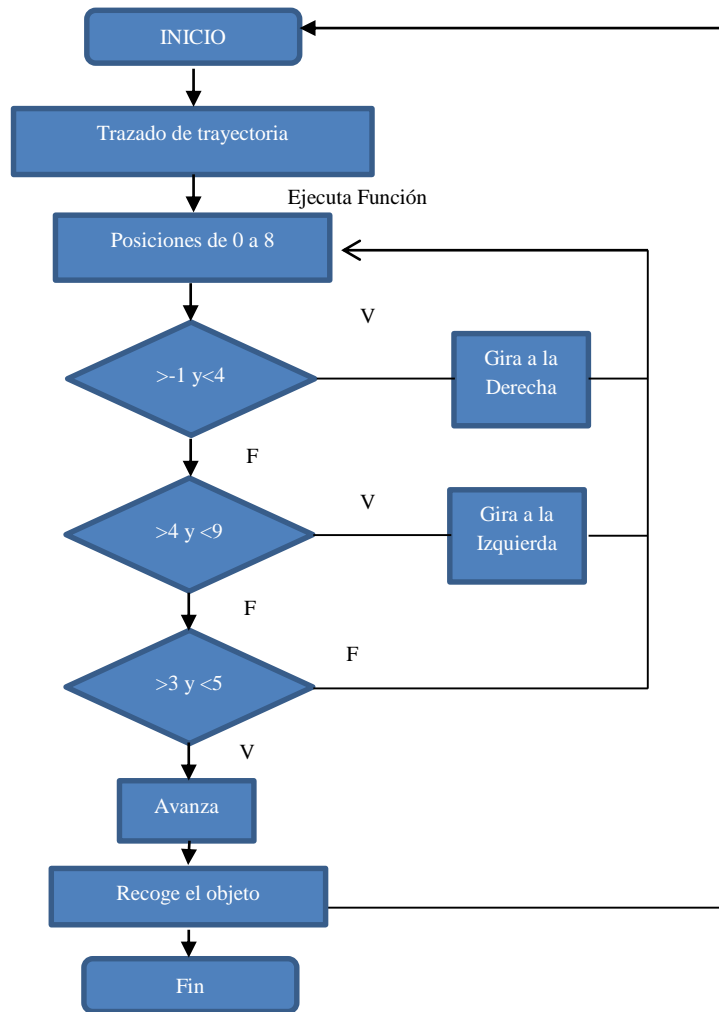


Figura 33: Diagrama de flujo del trazado de la trayectoria

Fuente: Adriana Hidalgo Panchana

El diagrama de flujo se explica a continuación:

Movimiento del servomotor para localización de objeto

En la parte frontal del robot se encontrará ubicado el sensor ultrasónico que está montado sobre un servomotor que servirá para ubicar la posición de un objeto, el sensor ultrasónico gira por acción del servomotor; el servomotor gira 180 grados y se determinan 9 posiciones específicas que no interfieran con la ubicación de las patas en movimiento, es decir la parte mecánica de detección que mueve al sensor no debe detectar el movimiento de las extremidades del robot.

Posiciones

Se plantea utilizar 9 posiciones que son las direcciones de giro del servomotor cada una con un ángulo específico para el movimiento del servomotor, las direcciones de posición van de 0 hasta 8, como se muestra en la tabla (15) las posiciones y grados respectivamente.

Posiciones	Grados
Posición 0:	40 °
Posición 1:	50 °
Posición 2:	60 °
Posición 3:	80 °
Posición 4:	100 °
Posición 5:	110 °
Posición 6:	120 °
Posición 7:	125 °
Posición 8:	130 °

Tabla 15: Posiciones y Grados

Fuente: Adriana Hidalgo Panchana

En el instante que el servomotor está girando por cada posición de ángulo, el sensor busca al objeto-basura de manera que registra en cada posición de giro un valor de distancia. Este valor de distancia se compara con las opciones de posiciones en la codificación para que de acuerdo a esa comparación tome la acción correspondiente girar o avanzar. De acuerdo a la posición donde se encontró el objeto el robot se desplazará considerando la siguiente tabla (16):

Posición	Movimiento
>4 y <9	Hacia la izquierda
> -1 y <4	Hacia la derecha
>3 y < 5	Avanza

Tabla 16: Posiciones y Movimientos correspondientes

Fuente: Adriana Hidalgo Panchana

Cuando el robot no detecte nada dentro del rango de los 50 cm, seguirá girando a la izquierda o derecha hasta localizar el objeto dirigido hacia la parte central del robot,

una vez que localice al objeto tendrá las siguientes opciones presentes en la tabla (17):

Posiciones	Movimiento	Siguiente Paso
0,1,2,3,5,6,7,8	Giro	Giro
4	Avanza	Avanza

Tabla 17: Posiciones, Movimiento y Siguiente paso de la estructura

Fuente: Adriana Hidalgo Panchana

En la figura (34) se muestra las posiciones que tendrá el robot hexápodo hacia el lado derecho. Se muestra las posiciones con su respectivo valor en grados, los valores de posicionamiento de 0 a 3 comprendidos entre números > -1 y < 4 , con los movimientos de la estructura hacia la derecha con sus grados correspondientes.

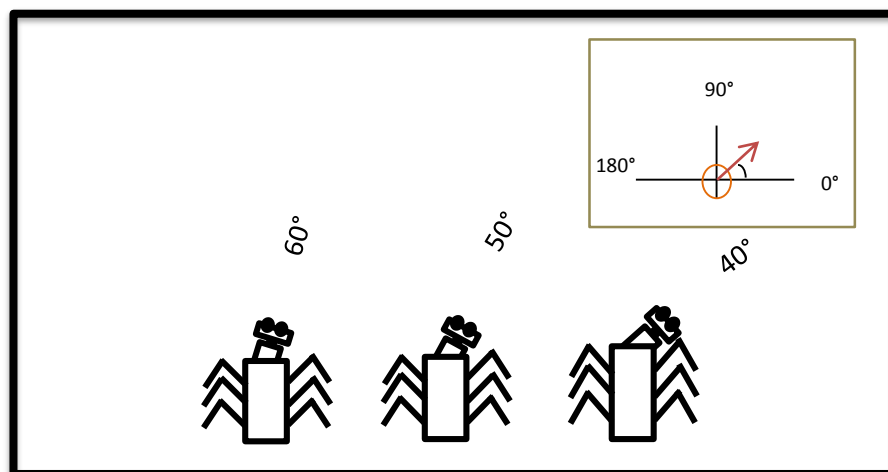


Figura 34: Posicionamiento, movimiento hacia la derecha

Fuente: Adriana Hidalgo Panchana

En la figura (35) se muestra las posiciones que tendrá el robot hexápodo al momento de avanzar con su respectivo valor en grados, el valor de posicionamiento es de 4 comprendido por el número > 3 y < 5 significa que la trayectoria es hacia adelante.

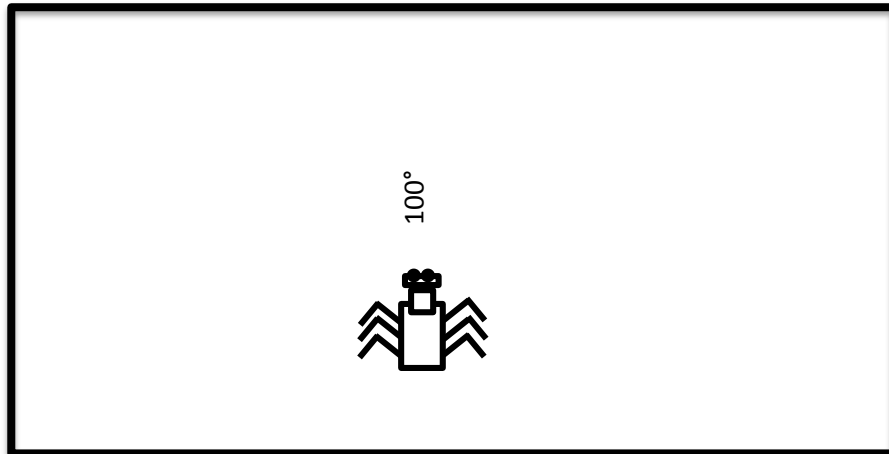


Figura 35: Posicionamiento, movimiento de avance

Fuente: Adriana Hidalgo Panchana

En la figura (36) se muestra las posiciones que tendrá el robot hexápodo hacia el lado izquierdo con su respectivo valor en grados, los valores de posicionamiento son de 5, 6, 7, 8 comprendidos entre números >4 y <9 los movimientos son hacia la izquierda con sus grados correspondientes.

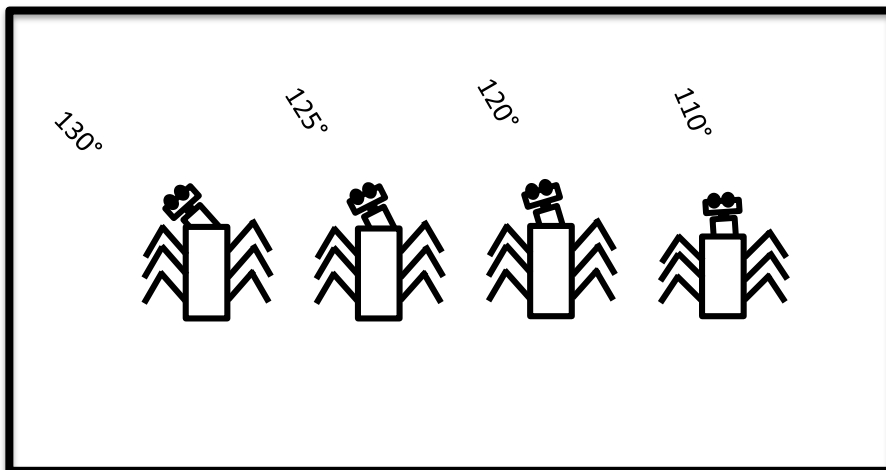


Figura 36: Posicionamiento, movimiento hacia izquierda

Fuente: Adriana Hidalgo Panchana

Función del trazado de la trayectoria

Una función `loop()` que ejecuta la función consecutivamente y sin interrupciones. Dentro de este función `loop` se encuentra la ruta de elección de trayectorias girar-

avanzar o solo avanzar dentro de las 9 posiciones, así mismo dentro de esta se determina la distancia mínima hacia el objeto para que se proceda a recoger. Cuando se termina de ejecutar la función. Se debe reiniciar el programa.

```

void loop (){
  ultra2 = ultrasonic2.Ranging(CM);
  for(j=0;j<9;j++)
  { servo19.write(PosValue[j]);
    delay(300);
    ultra2 = ultrasonic2.Ranging(CM);
    stringThree = stringOne+ultra3+' '+ultra2+' '+ultra1;
    Serial.println(stringThree);
    GP2[j]=ultra2;}
  MIN_Point = GET_Point();
  Serial.println(MIN_Point);
  delay(100);
  paro();
  while(sendey==1)
  { paro();
    leer=Serial.read();
    if (leer=='s'){
      sendey=0;}}
    if ((MIN_Point>-1)&&(MIN_Point<4))
  {derecha();
    sendey=0;
    Serial.println(MIN_Point);}
  else
  { if ((MIN_Point>4)&&(MIN_Point<9))
    { izquierda();
      sendey=0;
      Serial.println(MIN_Point);}
    else
  { while(MIN_Point==4)
    { adelante();
      Serial.println(MIN_Point);
      if (ultra2<8)
      {paro();// ha detectado avanzado hasta el objeto salir del bucle
        agache();
        garra();

```

Servo búsqueda de objeto

Entre >-1 y <4= derecha

Entre >4 y <9= Izquierda

Entre >3 y <5= adelante

Valor de 8 cm recoge

Pinza se cierra

```
sendey=1;  
MIN_Point=0; } } }
```

Nivel 2.3 Recolección del objeto

La Pinza Robótica

La pinza robótica es un complemento que se le añadió a la araña con la finalidad de que realice el trabajo de recolectar el objeto-basura. El movimiento mecánico de la pinza es realizado por el servomotor y un engranaje adaptable que permitirán abrir y cerrar la pinza. En la figura (37) se muestra la parte física vista desde arriba y lateral.

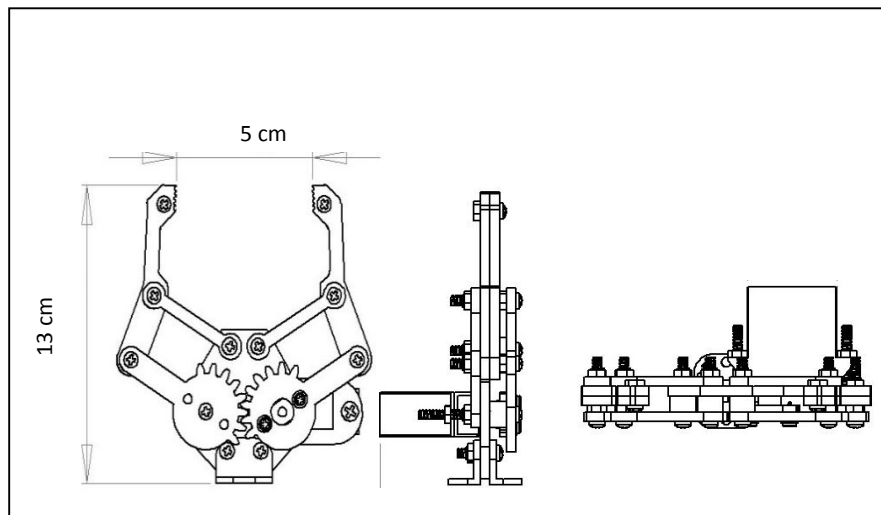


Figura 37: Pinza Robótica

Fuente: PinzaRobotica.com

La pinza tiene las siguientes dimensiones y se las detalla a continuación: de largo tiene 13 centímetros, de ancho 5,20 centímetros y sus dedos se abren alrededor de 5 cm. Se seleccionó este modelo de pinza en el cual se abren las dos tenazas al mismo tiempo para un mejor agarre y firmeza al momento de sostener el objeto. Dependiendo de la fuerza del servomotor que se le adapte este podrá agarrar los elementos, se pretende utilizar un servomotor de 10 kg caracterizado por tener una fuerza relativa. Como se muestra en la tabla (18) las características del servomotor que abre y cierra la pinza son:

Movimiento	Servomotor	Torque
Agarre	HexTronikHX12K	10 Kg / cm

Tabla 18: Características del Servomotor

Fuente: Adriana Hidalgo Panchana

En la figura (38) se muestra el diagrama de flujo para el proceso de recolección del objeto. Correspondiente a la pinza robótica, sensor, giro del servomotor y distancia al objeto.

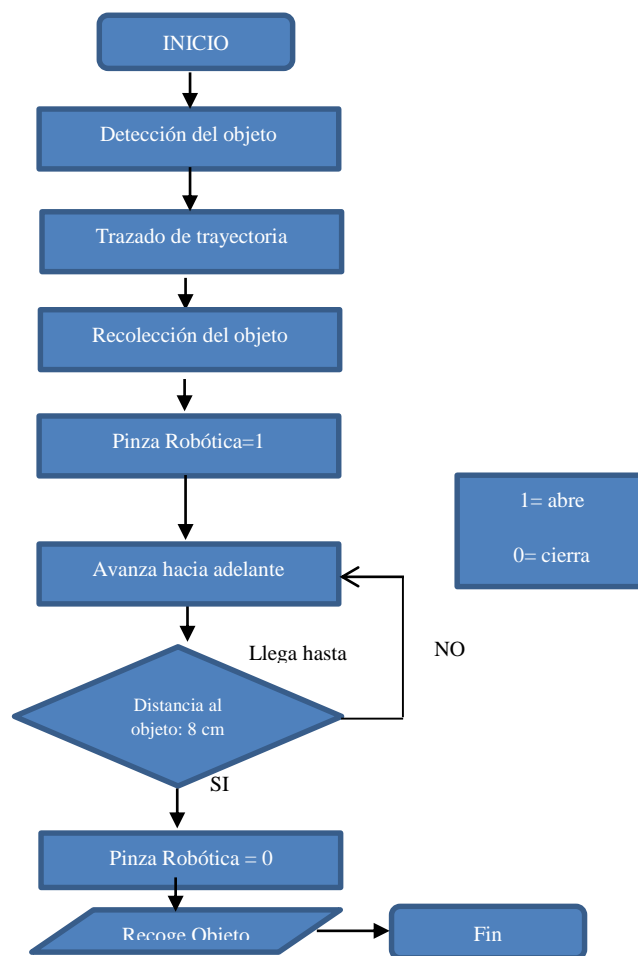


Figura 38: Diagrama de flujos de la recolección del objeto

Fuente: Adriana Hidalgo Panchana

A continuación se explica el proceso del diagrama de flujo:

Una vez detectado el objeto y ya elegido la ruta a seguir hacia el objeto como parte siguiente es la recolección del objeto, cuando se encuentre en esta parte la pinza estará abierta, en el diagrama tenemos representado con un número digital:

1 para abrir la pinza

0 para cerrar la pinza.

Mientras la pinza se encuentre abierta el robot hexápodo avanza hacia adelante hasta encontrar la distancia apropiada para tomar el objeto.

Distancia apropiada para recoger el objeto.

Cuando la estructura solo tenga la ruta de camino hacia adelante esta procede a obtener las siguientes opciones que se muestran en la tabla (19).

Avanza la estructura	Ultra2 > 8cm	Seguirá Avanzando
	Ultra2 <= 8 cm	Recoge el objeto

Tabla 19: Distancia del Sensor Ultrasónico para recoger el objeto

Fuente: Adriana Hidalgo Panchana

La recolección del objeto se realizará mediante una pinza robótica que consiste en un servomotor montado sobre un mecanismo de piñones que finaliza con una pinza o tenaza, la cual recibe las órdenes enviadas desde la tarjeta de control a través de la codificación, la pinza se encontrará abierta antes de empezar la trayectoria hacia el objeto, cuando el robot se encuentre a 8 cm del objeto basura, la pinza empezará a cerrarse en el diagrama representado con el número digital del valor 0 y de esta manera agarrará el objeto, el procedimiento se detiene y para que pueda recoger otro objeto, se debe reiniciar el sistema.

4.1.2. Desarrollo del Sistema de Fuerza

Nivel b.1.1 Los Actuadores

Los actuadores son los servomotores que estarán ubicados en las articulaciones y permitirán el posicionamiento mediante un bucle interno de control.

Las características de los servomotores son las siguientes:

- Poseen diferentes tamaños y pesos.
- A diferencia de los motores paso a paso, los servomotores son reguladores que ejercen una fuerza para llegar a una posición.
- La fuerza que se ejerce es alta.
- Relativamente de bajo costo.
- Los potenciómetros que se encuentran en su interior permiten la lectura de su posición.
- Giro de precisión hasta 180 grados y otros giros completos.

En la figura (39) se muestra la estructura de un servomotor.

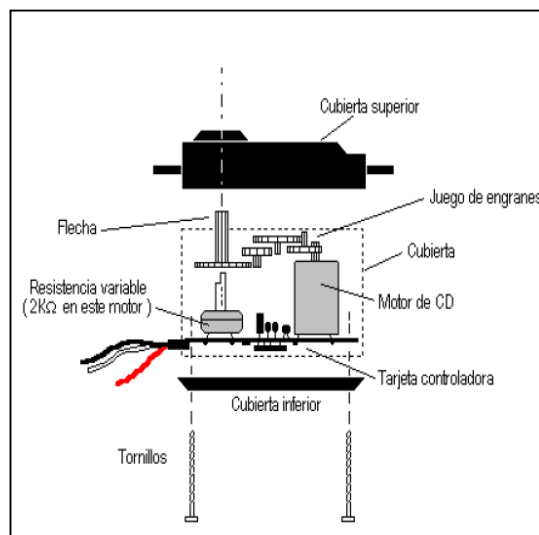


Figura 39: Estructura de un servomotor

Fuente: Maquinas Eléctricas

Nivel b.1.2 Locomoción Del Robot

El robot tendrá la apariencia de una araña de seis patas, cuya locomoción es conformada por un cuerpo y seis extremidades, cada pata cuenta con 3 articulaciones y servomotores situados sobre cada articulación por consecuencia en total 3 servomotores por cada pata, la implementación total del robot son 18 servomotores que le dan movimiento a la estructura de la araña.

Configuración del cuerpo

El cuerpo del hexápodo está diseñado de manera que las extremidades o patas que se coloquen alrededor del cuerpo no tengan conflictos mecánicos, el tipo de configuración que se utilizó es la Configuración Bilateral.

La configuración Bilateral

Se tomó como referencia este modelo para dar un frente y un espacio adicional al hexápodo ya que en la parte frontal del robot se colocará adicional la pinza robótica. Esta configuración presenta un gran equilibrio en movimientos que van de atrás hacia adelante y viceversa. Al momento de la codificación, este presenta una ventaja ya que los movimientos se realizan igualmente en todas las direcciones al eje de simetría. El principal problema que presenta este diseño es que al instante de querer cambiar de dirección toda la codificación cambia también. En la figura (40) se muestra la distribución de las patas.

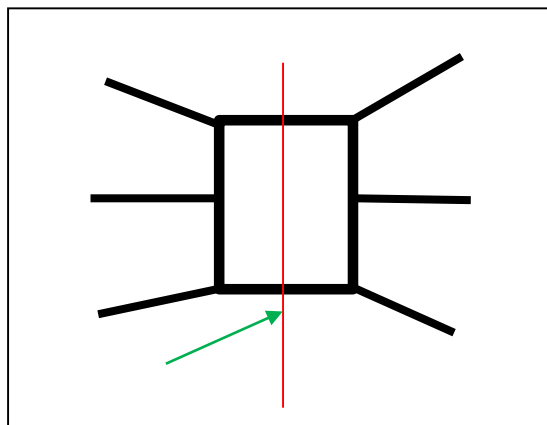


Figura 40: Modelo Bidireccional

Fuente: Adriana Hidalgo Panchana

Estructura de una extremidad

Se optó por tres grados de libertad para proporcionar un mayor soporte, movilidad y capacidad para girar dentro de los ángulos en un área establecida. Este consta de seis patas cada una cuenta con tres articulaciones y tres grados de libertad el conjunto con los servomotores ubicados de una correcta manera proporcionan una movilidad y orientación del robot.

En la figura (41) se muestra los tres grados de libertad que se asignaron al robot hexápodo para su orientación.

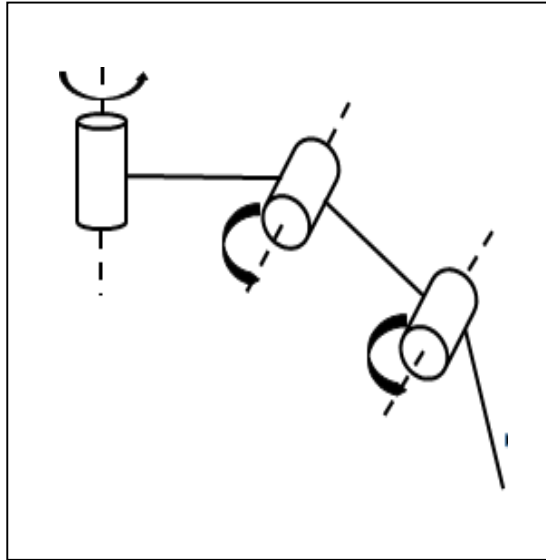


Figura 41: Grados de extremidad de un extremidad

Fuente: Adriana Hidalgo Panchana

Existen dos tipos de ubicación de los servomotores que son:

- El modo de *Actuación In.situ*
- El modo *Paralelogramo Deformable*

Al colocar los servomotores se debe buscar la manera de ubicarlos lo más cerca de las articulaciones.

El modo de *Actuación In-Situ* es la ubicación directamente en cada articulación. En cambio el modo *Paralelogramo Deformable* es colocar otro tipo diferente de pequeña estructura adicional en cada pieza para sostenerlo y de ahí colocarlo en la pieza correspondiente. Por la ubicación de los servomotores se designara el método actuación In-Situ.

En la figura (42) se muestra los dos tipos de ubicación de los motores.

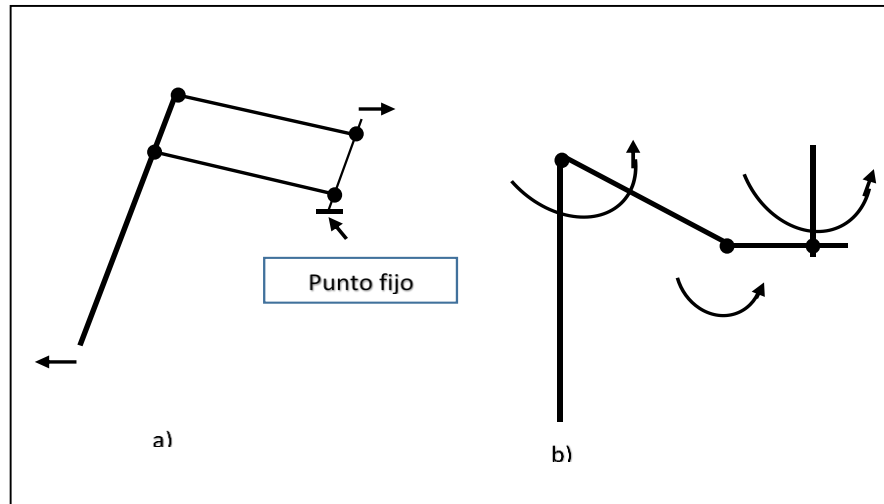


Figura 42: a) Paralelogramo deformable b) Actuación In-situ

Fuente: Adriana Hidalgo Panchana

La estructura mecánica

Se detallarán los elementos mecánicos que hacen que la estructura este completa. Entre los más destacados están: el cuerpo, las patas y los actuadores con sus respectivas características y dimensiones.

Como referencia se tomó de ejemplo el robot hexápodo que se realizó en cursos anteriores en donde la estructura era de aluminio de 0,1 mm de grosor, al completar la estructura de aluminio más los elementos se realizaron pruebas de torsión y se llegó a la conclusión de que este tipo de aluminio era un material flexible y fácil de doblar con el exceso de peso aplicado. Por lo que se optó como material de la estructura al aluminio 6061 de 0,25 mm de grosor, de material duro, resistente y no es fácil de doblar.

El cuerpo

El robot Hexápodo tiene un aspecto natural, las piernas y el cuerpo más articulados. Los tres DOF (grado de libertad) que contiene cada pierna significa que el este

robot puede caminar en cualquier dirección. El robot ha sido diseñado para utilizar 18 servomotores TowerPro Mg995 para las piernas.

El cuerpo se implementará para mantener más espacio entre las extremidades para evitar roces entre las extremidades. El cuerpo del hexápodo tiene las siguientes medidas:

De ancho 16 cm.

De largo 18 cm.

En la figura (43) se muestra la estructura del cuerpo.

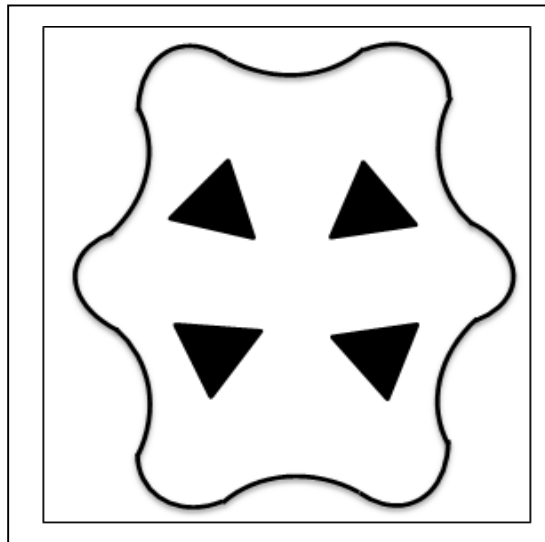


Figura 43: Cuerpo del Hexápodo

Fuente: Adriana Hidalgo Panchana

Patas

Cada pata tendrá 3 grados de libertad y soportará tres servomotores por cada extremidad, que permitirá la eficiencia de sus movimientos individuales por pata, constan de las siguientes medidas:

- La articulación inicial tiene 6 cm x 3.5 cm.
- La articulación media tiene 3 cm x 10cm.
- La articulación final tiene 9 cm x 3cm.

En la figura (44) se muestra la ubicación de las articulaciones de una pata.

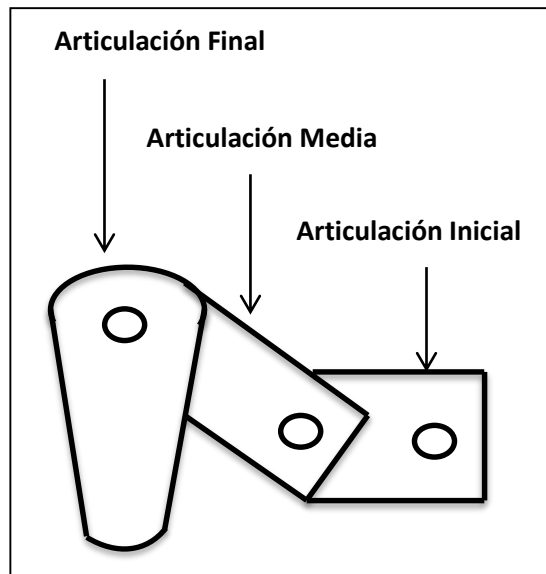


Figura 44: Articulaciones de una extremidad

Fuente: Adriana Hidalgo Panchana

La distribución de los servomotores es:

- Seis servomotores para las articulaciones iniciales.
- Seis servomotores para las articulaciones medias.
- Seis servomotores para las articulaciones finales.

Los servomotores que se utilizan en las extremidades poseen las siguientes características que se detallan a continuación en la tabla (21):

Articulación	Servomotor	Torque
Inicial	Tower Pro995 (180°)	15 Kg / cm
Media	Tower Pro995 (180°)	15 Kg / cm
Final	Tower Pro995 (180°)	15 Kg / cm

Tabla 20: Descripción de los Servomotores

Fuente: Adriana Hidalgo Panchana

En la siguiente figura (45) se muestra la ubicación de cada una de las patas y se señala la articulación correspondiente que se asignó en la codificación.

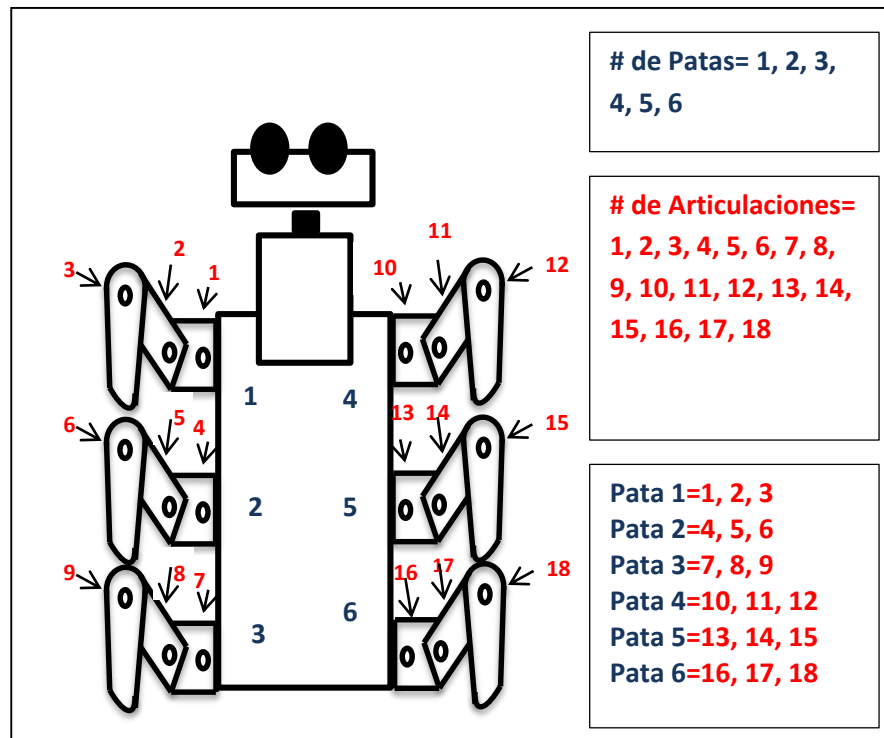


Figura 45: Ubicación de las patas y extremidades

Fuente: Adriana Hidalgo Panchana

Tipo de movimiento: Ciclo de movimiento Trípode

Se utilizó este ciclo porque se analizó que sería el más adecuado y entendible en la codificación. El triángulo de apoyo el cual está formado por los extremos de tres patas es decir son dos triángulos, que se utilizan como guía para seguir un orden de las patas al momento de coordinar los movimientos en la codificación. Se comienza con el primer triángulo que incluye: la pata tres, la pata cinco y la pata uno; el segundo triángulo está conformado por la pata seis, la pata dos y la pata cuatro.

Como consecuencia de movimientos levanta tres patas mientras las otras tres se mantienen en el suelo. Cuando las tres patas levantadas ya estén de nuevo en el suelo, las otras tres avanzan así continuamente. Este sistema mantiene siempre la estructura en equilibrio.

En la figura (46) se muestra los dos triángulos de apoyo conformados por tres patas.

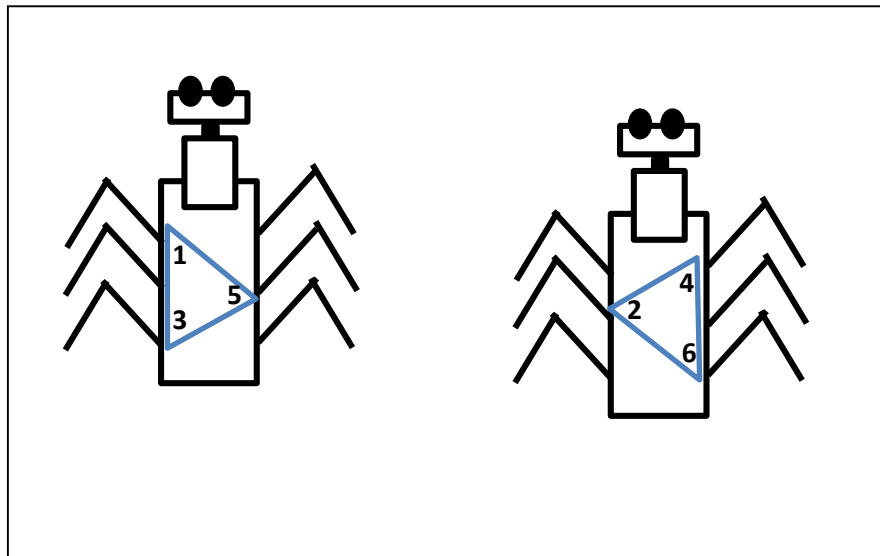


Figura 46: Configuración del triángulo de apoyo

Fuente: Adriana Hidalgo Panchana

Ajuste de los ángulos presentes en cada articulación de la extremidad

Para conocer la dirección del ángulo que tiene cada servomotor, se utiliza en cada uno de los 18 servomotores la posición y ángulo de referencia, colocándolos en cada articulación individual suministrando de energía al servomotor, y así ajustar la dirección en la que mueve, según este proceso se corrige los ángulos y se procede a armar la pata, este proceso se necesita para posicionar todas las extremidades sobre el cuerpo del robot hexápodo.

Nivel b.1.3 Fuente De Poder

Se utilizará una fuente conmutada de 5 voltios extraída de un computador, esta suministrara de energía a todo el robot, conectada en la bornera que tendrá la placa adaptable. Esta fuente de alimentación contiene la cantidad de voltaje y amperaje que necesita la estructura así el funcionamiento del robot será correcto.

En la siguiente tabla (21) se muestra los parámetros de amperaje totales de los elementos necesarios, los valores de potencia que necesitara el robot son obtenidos del consumo de amperaje del sistema.

Elemento	Cantidad	Corriente Individual	Corriente Total en mA
Mega Arduino	1	2170	2170
Servomotor Tower Pro 995- 15kg	18	100	1800
Servomotor HexTronikHX12K de 10KG	1	100	100
Servomotor MG 90S Micro servo	1	100	100
Sensor Ultrasónico HC-SR04	1	15	15
Corriente total			4185mA

Tabla 21: Corriente Suministrada por los elementos

Fuente: Adriana Hidalgo Panchana

4.2. Diseño de Interfaz Gráfica

Existen muchas aplicaciones pero se seleccionó el lenguaje de la plataforma de arduino, ya que contiene un conjunto de herramientas para aplicaciones de control, en este caso lo utilizaremos como interfaz entre la comunicación de la plataforma robótica y el lenguaje de programación. En la figura (47) se muestra la comunicación existente entre el robot y el ordenador.

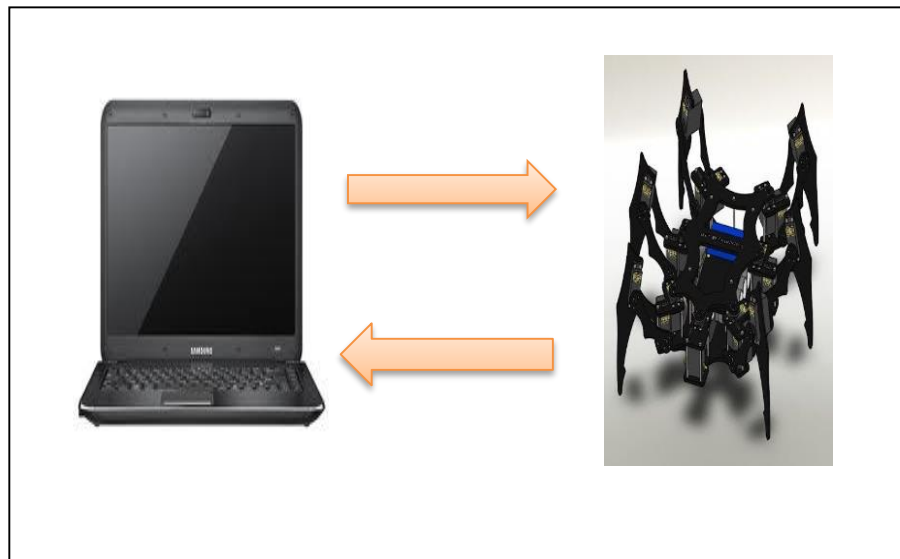


Figura 47: Software de interfaz del robot hexápodo

Fuente: Adriana Hidalgo Panchana

El software arduino integra todas las funciones en un solo programa mediante la interfaz gráfica del monitor serial se monitorea los valores de los sensores y realiza el control de movimiento, el monitor serial presentará las posiciones del sensor mostradas en centímetros específicamente la detección del objeto a recoger.

4.2.1. Monitor Serial

En la parte superior izquierda de la ventana principal de software arduino, se encuentra la barra de menú: archivos, editar, sketch, herramientas y de ayuda; en las barra de herramientas se podrán elegir el tipo de tarjeta a utilizar, como también

la asignación del número del puerto COM serial. En el área de mensajes se muestra cuando la compilación de la codificación ha finalizado, así mismo en el área de la consola se mostrará una línea de mensajes indicando la correcta compilación o los errores. En el área de botones de acceso se encuentra un icono de una lupa que hace referencia al acceso al monitor serial. En la figura (48) se muestra la ventana de monitor serial de arduino.

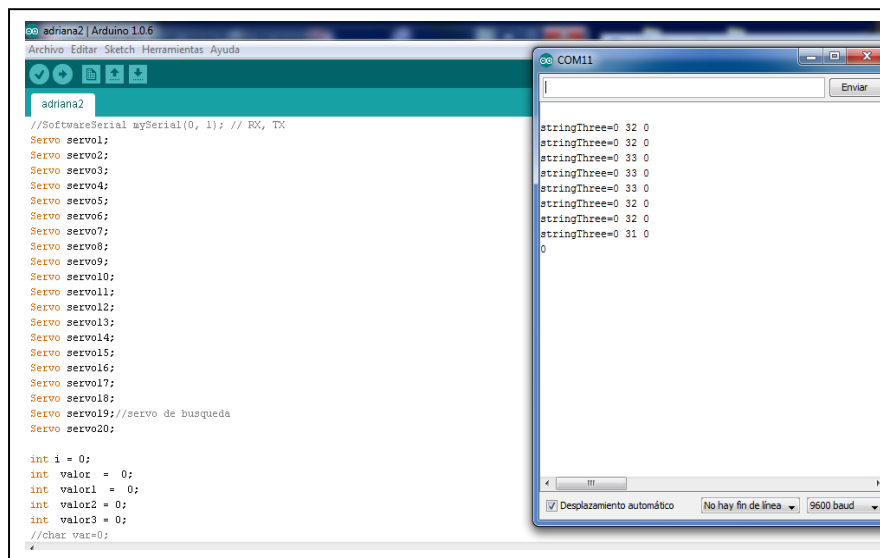


Figura 48: Interfaz Grafica

Fuente: Adriana Hidalgo Panchana

Para cargar el código de control en la tarjeta arduino se debe compilar con éxito es decir sin ningún error, luego se suministra de energía al robot, cuando tengamos estos dos paramentos listos procedemos a iniciar el monitor serial. Mediante esta ventana visualizamos nueve posiciones con su respectiva distancia hasta un máximo de 50 cm, a medida que el robot se mueva rastreará nueve valores y escogerá la distancia mínima comprendida entre todas posiciones, para que luego se trace la trayectoria hacia dónde dirigir la estructura del hexápodo. La codificación deberá estar configurada de manera que el desplazamiento del robot sea forma automática esto es para cuando detecte la posición del objeto en línea recta este comenzara a desplazarse. Posiciones y su movimiento hacia donde está el objeto:

- Si son >3 y <5 La dirección es hacia adelante.
- Si son >-1 y <4 el robot hexápodo se dirige hacia la derecha.
- Si son >4 y <9 el robot hexápodo se dirige hacia la izquierda.

En la figura (49) se muestra la ventana del monitor serial.

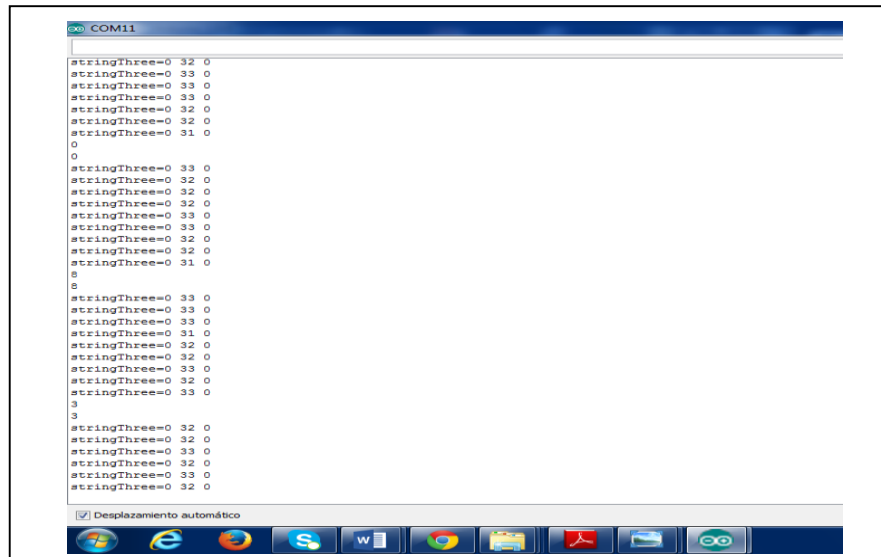


Figura 49: Pantalla monitor Serial

Fuente: Adriana Hidalgo Panchana

CAPÍTULO 5

IMPLEMENTACIÓN

5. IMPLEMENTACIÓN Y DEMOSTRACIÓN DE HIPÓTESIS

Se presenta las gráficas y tablas de los procesos de la implementación del robot hexápodo, mostrando paso a paso el ensamblaje del prototipo del robot hexápodo; utilizando como base los parámetros detallados en el capítulo anterior. También se analizarán las pruebas experimentales del prototipo para su correcto funcionamiento.

5.1. Construcción

En este apartado se detalla la construcción de las partes del prototipo del robot hexápodo para la identificación y recolección del objeto, el cual está dividido en:

Construcción Mecánica

- Estructura del robot hexápodo
 - Ensamblaje de las articulaciones
 - Ensamblaje del cuerpo del robot.

Construcción Eléctrica

- Construcción de la tarjeta adaptable
- Distribución de los pines de los elementos a la tarjeta arduino
- Procedimiento para implementar la pinza robótica:

- Implementación de la Pinza robótica
- Soporte entre la pinza y el cuerpo del hexápodo
- Implementación de la parte de detección

5.1.1. Construcción Mecánica

- **Estructura del robot hexápodo**

En el capítulo anterior se mencionó las dimensiones y forma de la estructura del robot, finalmente se utilizó una estructura de un hexápodo comercial que es de un material resistente utilizado como esqueleto y sostenimiento del robot. En la figura (50) se muestra las partes de la estructura sin armar.



Figura 50: Estructura del Hexápodo

Fuente: Adriana Hidalgo Panchana

Se adquirió 18 servomotores para ensamblar las articulaciones de las extremidades de la estructura del prototipo del robot hexápodo, con estos actuadores procedemos a ubicar las extremidades del robot hexápodo en el cuerpo, ya habiendo realizado el cálculo de la orientación de los ángulos para que se dirijan todos en una misma dirección.

La distribución del número de patas se presenta en la figura (51).

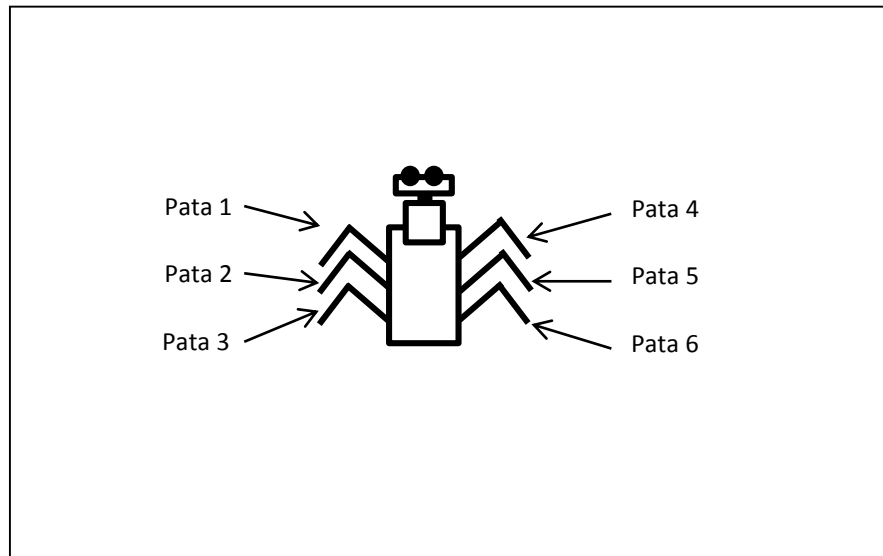


Figura 51: Numero de patas en la estructura

Fuente: Adriana Hidalgo Panchana

- **Ensamblaje de las articulaciones**

Para proceder a ensamblar una pata se toma cada una de las articulaciones con un servomotor correspondiente y los tornillos para ajustarlos, así como sus engranajes.

En la figura (52) a) se observa las articulaciones sin ensamblar de una pata, como se indica a continuación las partes y elementos a utilizar para una extremidad son:

1 Articulación Inicial- Servomotor- engranaje- tornillos.

1 Articulación Media- Servomotor- engranaje- tornillos.

1 Articulación Final- Servomotor- engranaje- tornillos.

En la figura (52) b) se muestra la extremidad ya armada.

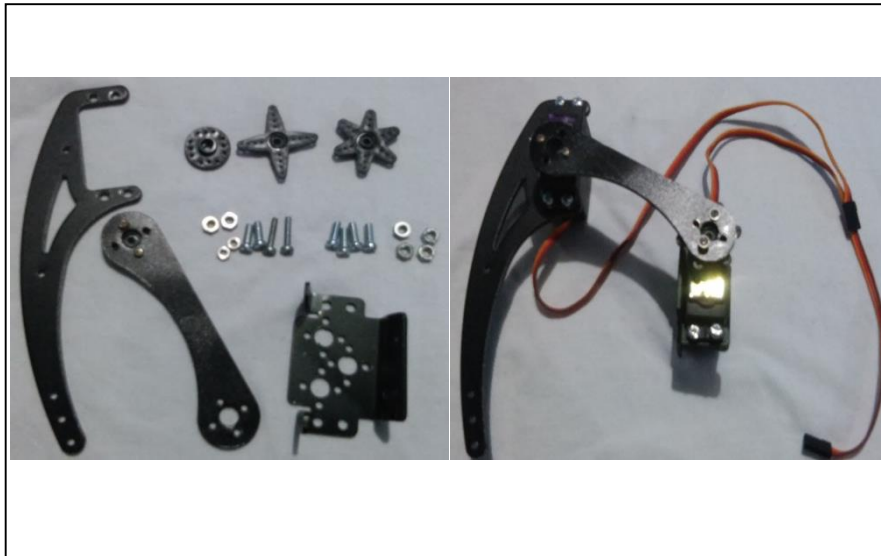


Figura 52: a) Elementos sin ensamblar, b) Una pata ensamblada

Fuente: Adriana Hidalgo Panchana

Este proceso se repite con las cinco patas restantes, luego de ensamblar las patas se procede a armar el cuerpo.

- **Ensamblaje del Cuerpo**

Para ensamblar las partes del cuerpo se necesita los dos lados del cuerpo y los servomotores que se colocaran para controlar el movimiento inicial. Los elementos necesarios para formar el cuerpo del hexápodo son:

2 Partes del cuerpo

6 Servomotores

6 Articulaciones iniciales

Engranajes para los servomotores y tornillos.

En la siguiente figura (53) se muestra el cuerpo del robot ya ensamblada.

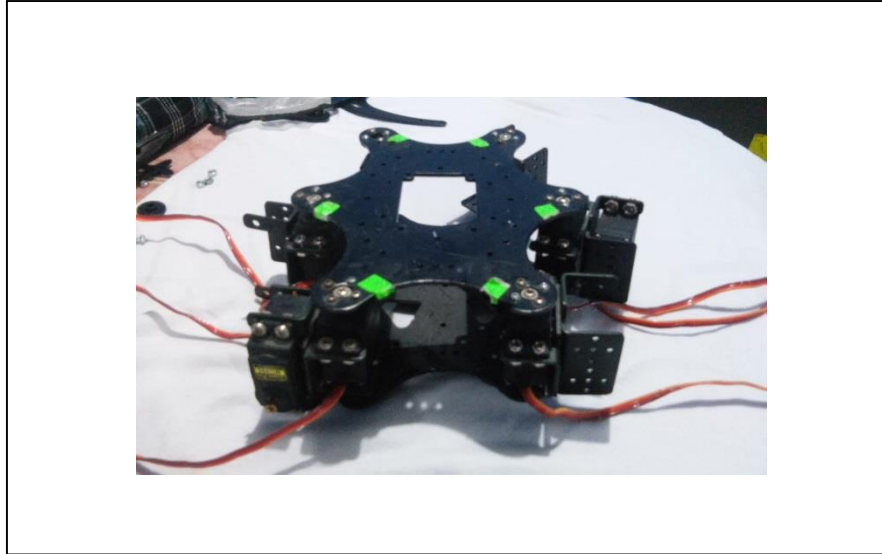


Figura 53: Cuerpo del robot

Fuente: Adriana Hidalgo Panchana

En la figura (54) se muestra la estructura del hexápodo ya armada vista desde arriba, en la parte del cuerpo del robot hexápodo se ha colocado números que diferencie a cada pata, estos números servirán de guía para la programación de control.

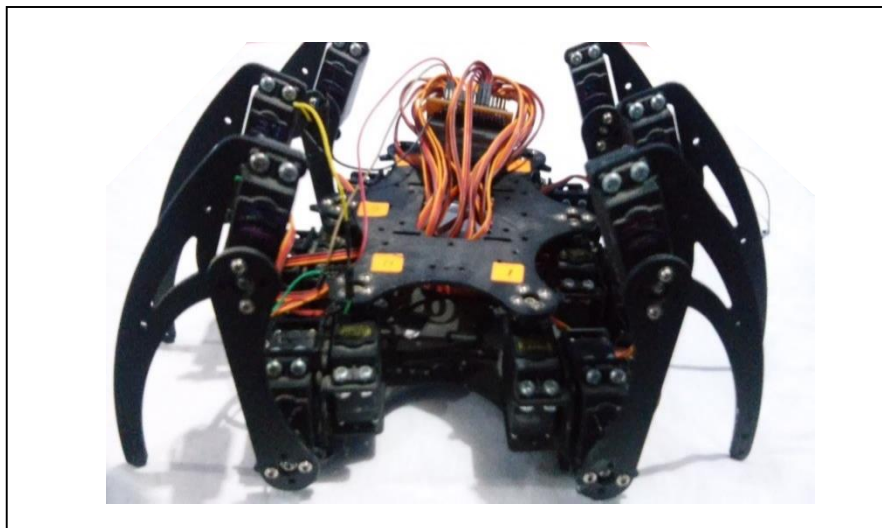


Figura 54: Estructura y servomotores colocados en la estructura

Fuente: Adriana Hidalgo Panchana

5.1.2. Construcción Eléctrica

- **Construcción de la tarjeta adaptable**

Como parte de la construcción eléctrica está la construcción de la placa adaptable, diseñada con el propósito de organizar y conectar los pines de todos los servomotores y el sensor ultrasónico correspondiente. La adaptación de la placa es colocada sobre la placa mega arduino, la placa tiene el objetivo de distribuir equitativamente la corriente y el voltaje hacia todos los elementos. A continuación se presenta en la figura (55) la simulación del circuito en Proteus en ISIS permite diseñar el plano eléctrico del circuito de la tarjeta adaptable con los componentes necesarios.

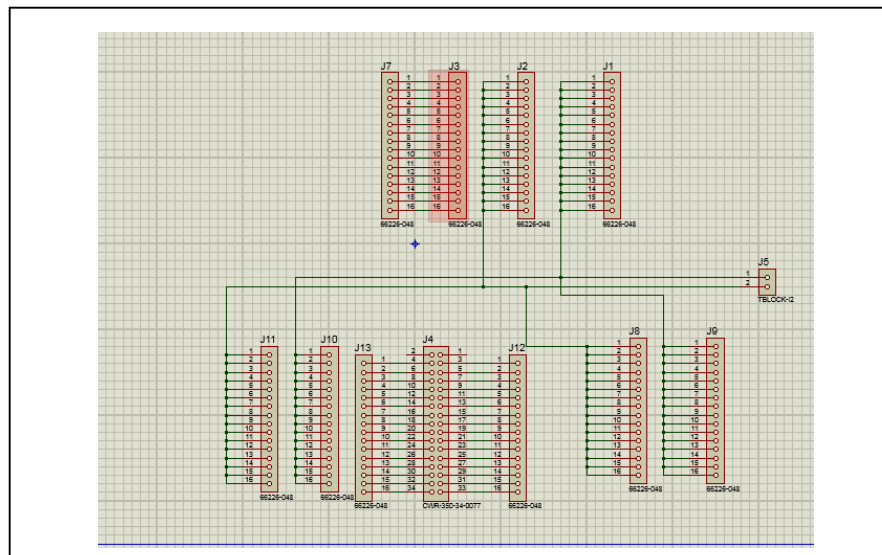


Figura 55: Simulación en Isis Proteus

Fuente: Adriana Hidalgo Panchana

En la figura (56) a) se muestra la simulación del circuito realizada en Proteus en PBC, simula las dimensiones y el trazado de la pista en donde serán colocados los elementos a colocar en la placa. En la figura (56) b) se muestra la simulación del circuito realizada en Proteus en el Diagrama 3D Visualizer, esta parte adicional del programa de Proteus es la vista tridimensional del circuito electrónico, nos muestra el ejemplo de cómo quedara la parte frontal de la placa adicional a diseñar.

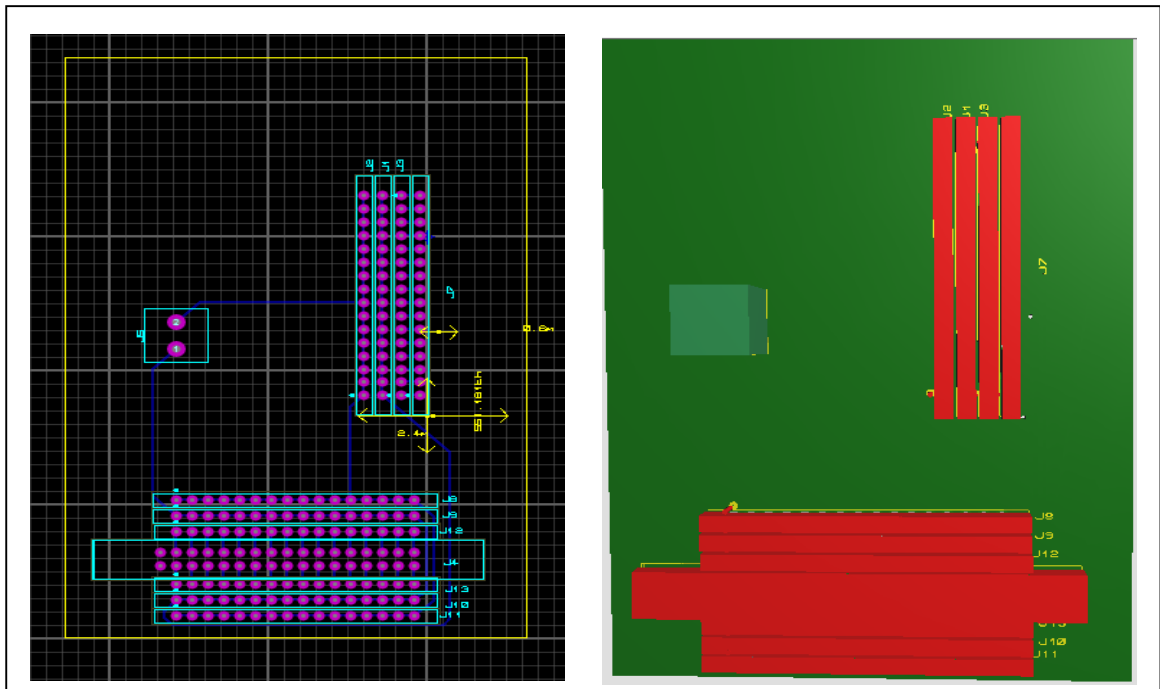


Figura 56: a) Simulación en PBC Proteus, b) Simulación en 3D visualizer

Fuente: Adriana Hidalgo Panchana

En la figura (57) se muestra la adaptación de placa en la parte física sobre la tarjeta mega arduino.

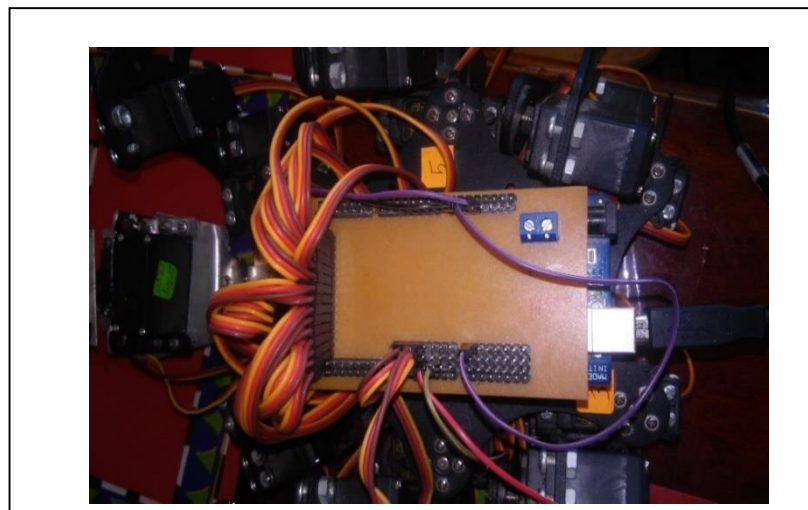


Figura 57: Tarjeta Adaptable Diseñada

Fuente: Adriana Hidalgo Panchana

En la siguiente figura (58) se muestra la ubicación en un lugar estratégico para la tarjeta arduino sobre la estructura.

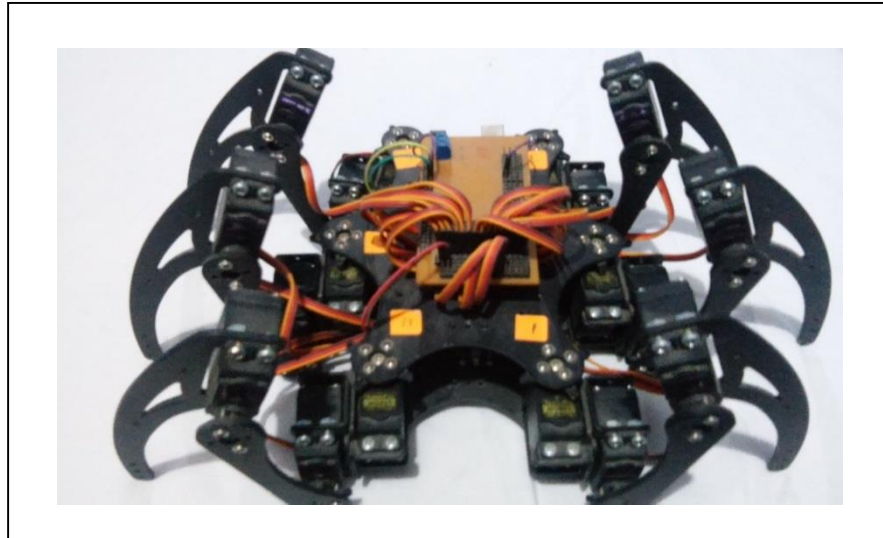


Figura 58: Ubicación del Mega Arduino sobre el Robot

Fuente: Adriana Hidalgo Panchana

La tarjeta mega arduino se ubica sobre la estructura luego de haber colocado todas las extremidades en el cuerpo. La placa adaptable y la tarjeta de control deben estar distanciadas del metal del cuerpo del hexápodo ya que este se puede convertir en medio de conducción energética y puede provocar que se dañen otros elementos, por tal motivo se utilizó unas pequeñas elementos mecánicos aislantes que sirven para distanciar el cuerpo de las tarjetas.

- **Distribución de los pines de los elementos con la tarjeta arduino**

A continuación en la tabla (22) se detalla la distribución de conexión de los mecanismos del sensor ultrasónico y los servomotores hacia los pines de la tarjeta mega arduino:

#	Pin de Mega Arduino	Elemento	Articulaciones	# de Pata
1	Pin 22	servomotor	Inicial 1	Pata 1
2	Pin 24	servomotor	Medio 1	
3	Pin 26	servomotor	Final 1	
4	Pin 2	servomotor	Inicial 2	Pata 2
5	Pin 3	servomotor	Medio 2	
6	Pin 4	servomotor	Final 2	
7	Pin 5	servomotor	Inicial 3	Pata 3
8	Pin 6	servomotor	Medio 3	
9	Pin 7	servomotor	Final 3	
10	Pin 28	servomotor	Inicial 4	Pata 4
11	Pin 30	servomotor	Medio 4	
12	Pin 32	servomotor	Final 4	
13	Pin 46	servomotor	Inicial 5	Pata 5
14	Pin 48	servomotor	Medio 5	
15	Pin 34	servomotor	Final 5	
16	Pin 40	servomotor	Inicial 6	Pata 6
17	Pin 42	servomotor	Medio 6	
18	Pin 44	servomotor	Final 6	
19	Pin 36	servomotor	Búsqueda	
20	Pin 47	servomotor	Agarre	
21	Pin GND	De la tarjeta	Tierra Común	
22	Pin Vcc	De la tarjeta	Voltaje 5v	
23	Pin 49	Sensor U.	Trig	
24	Pin 51	Sensor U.	Echo	

Tabla 22: Designación de cada pin en la tarjeta Mega Arduino

Fuente: Adriana Hidalgo Panchana

En esta tabla se indica los pines que fueron asignados para los 18 servomotores y los pines que se utilizaran para conectar el sensor ultrasónico.

La tabla de la distribución de los pines servirá como una guía al momento de la programación en el software arduino y para cualquier modificación que se necesite realizar en el futuro.

- **Procedimiento para implementar la pinza robótica**
 - **Implementación de la Pinza**

En la figura (59) se muestra a) engranaje para la pinza, estas dos pequeñas piezas de engranaje son indispensables para el movimiento de la pinza y b) pinza robótica.



Figura 59: a) Engranaje b) Pinza Robótica

Fuente: Adriana Hidalgo Panchana

Al adquirir la pinza se recibió la estructura mecánica, al colocar el servomotor no se ajustaba porque no tenía ningún engranaje que le permitiera orientar a las tenazas, por tal motivo se investigó cual sería el engranaje adecuado y se llegó a la conclusión de adquirir un engranaje metálico circular de estructura ligera que posee de radio 1,5 cm, de grosor de 1 cm.

Una vez obtenido el engranaje se procede a agrupar las piezas de la pinza robótica con el servomotor, si el engranaje no es el colocado correctamente este se saltará y no abrirá ni cerrará la pinza. Luego de este proceso la pinza robótica será ubicada en la parte frontal del robot. En la figura (60) se muestra hacia la izquierda la pinza abierta; y hacia la derecha la pinza cerrada.

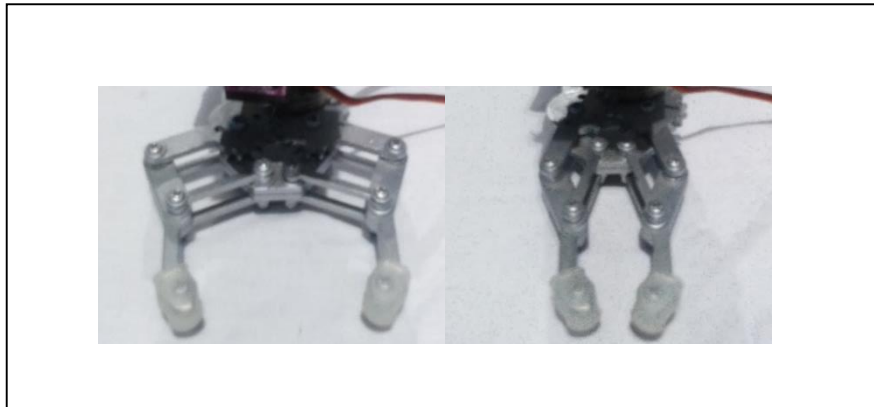


Figura 60: Pinza para el servomotor

Fuente: Adriana Hidalgo Panchana

Soporte entre la pinza y el cuerpo del hexápodo

Al elaborar el soporte entre la unión del cuerpo del robot y la pinza robótica, se probaron distintos tipos de material para dicha pieza, en la figura (61) la que se muestra hacia la izquierda es una lámina de Polietileno resistente de 3 cm de grosor y fácil de cortar.



Figura 61: Pieza para ajustar la pinza sobre el cuerpo del robot

Fuente: Adriana Hidalgo Panchana

En la figura (61) de la derecha es de metálica con un grosor de 1 ml, dichas piezas se ubicaron y se realizaron las pruebas respectivas y se llegó a la conclusión que tenían mucho peso y hacían que el robot se balanceara para adelante.

Como otra opción se realizó esta pieza de aluminio aunque este material es liviano y flexible, se le agregó una lámina pequeña de un elemento liviano ajustándola con el aluminio, se hizo la prueba colocándola sobre el robot, así se comprobó que ya no existía mayor peso hacia adelante.

- **Implementación de la parte de detección**

Para implementar la parte de detección se necesitó:

1 Sensor Ultrasónico

1 Servomotor de 1,5 kg

1 pieza que sostiene al sensor ultrasónico

1 pieza que sostiene al servomotor

En la siguiente figura (62) a) en la izquierda tenemos la parte anterior de la vista del sensor y en la figura (62) b) en la derecha tenemos la parte posterior de la pequeña estructura que se consideró para la ubicación del sensor ultrasónico sobre el servomotor.

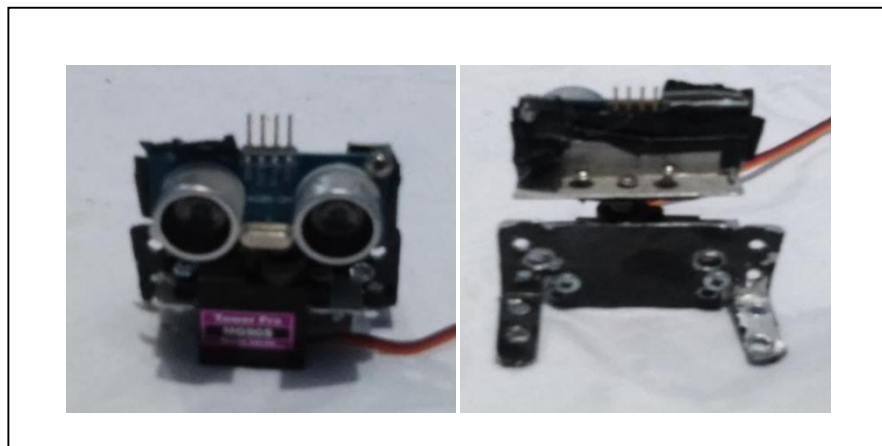


Figura 62: Sensor Ultrasónico con el servomotor

Fuente: Adriana Hidalgo Panchana

En la figura (63) se muestra la pieza ubicada con la pinza robótica, el servomotor y el sensor ultrasónico.

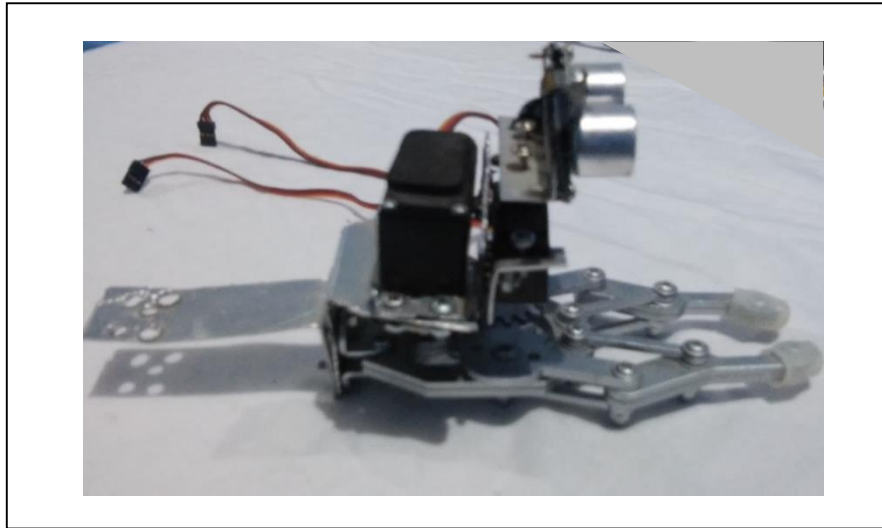


Figura 63: Ubicación Sensor, Pinza y pieza metálica

Fuente: Adriana Hidalgo Panchana

Se terminó de ensamblar todas las partes físicas tanto del desarrollo del sistema de control, locomoción y trayectoria del hexápodo que ayudaran a la contribución de la limpieza. En la figura (64) se muestra la implementación final del prototipo del robot hexápodo.

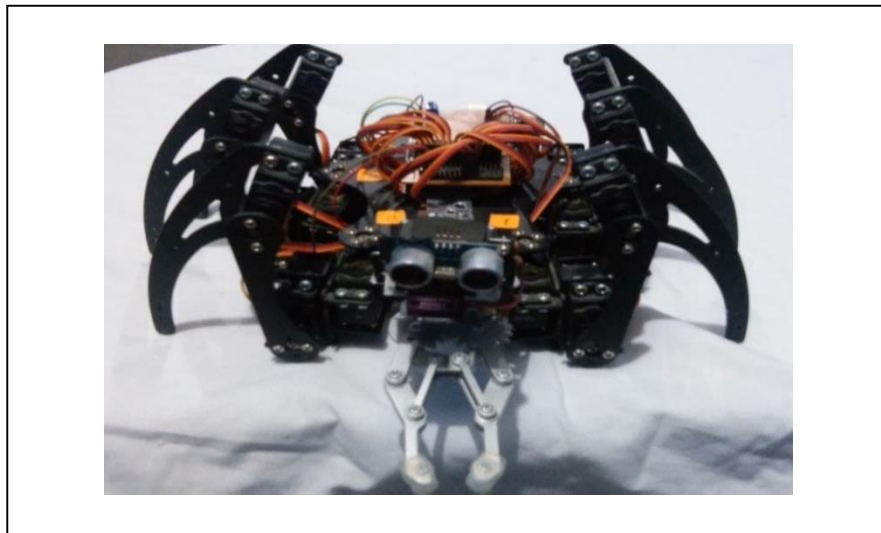


Figura 64: Prototipo de robot Hexápodo

Fuente: Adriana Hidalgo Panchana

5.2. Pruebas

A continuación se presentan los datos obtenidos de las pruebas experimentales, acerca de los diferentes procesos realizados en el hexápodo y la presentación de resultados como los indicadores de información para la implementación. Estas pruebas se realizan para corregir los errores y que exista un correcto funcionamiento.

5.2.1. Pruebas y resultados

Las pruebas se dividen en tres partes:

Pruebas con el sistema de Fuerza.

- Caminar hacia adelante y calcular la velocidad.
- Girar hacia la izquierda
- Girar hacia la derecha

Pruebas con el sistema de Control: Software parte Detección

- Posición 1 detectado en el mismo ángulo tres veces.
- Posición 2 detectado en el mismo ángulo tres veces.

Prueba Completa del Sistema del Robot Hexápodo.

- Detección y Recolección del objeto.
- Calculo de tiempo
- Calculo de velocidad

Antes de comenzar con las pruebas de avance de la estructura se hizo unos cálculos de velocidad con la que debería moverse el robot.

Velocidad con la que se moverá el robot hexápodo

En la programación del control se procedió a experimentar con diferentes valores de tiempo como 1000 ms de tiempo para que el robot realice la acción de caminar hacia adelante, entonces se obtuvo una velocidad de avance $<0,4$ m/s, así mismo se dio otro valor de tiempo de 10 ms con el cual se obtuvo que el movimiento fue $>0,4$ m/s. Finalmente se realizó la prueba con un valor de 100 ms para el movimiento entre patas y se obtuvo una velocidad de 0,4 m/s relativamente aceptable para el control de servomotores que generan el movimiento del robot.

Según el tiempo agregado este prolonga la velocidad en la que se demora la estructura del robot en desplazarse. En la tabla (23) muestra el tiempo y la velocidad con la que se desplaza el robot.

Tiempo de avance entre patas	Desplazamiento
1000 ms	$< 12.8 \times 10^{-3}$ m/s
≤ 100 ms	12.8×10^{-3} m/s
10 ms	$>12.8 \times 10^{-3}$ m/s

Tabla 23: Tiempo vs Velocidad de desplazamiento

Fuente: Adriana Hidalgo Panchana

Tiempo en el que da un paso

Para tomar el tiempo que se demora el robot en dar un paso, se cuenta los segundos mientras se enciende el robot, en el mismo instante se verifica en la pantalla del monitor serial cuando ya haya dado un paso, entonces se procede a detener el proceso.

Un total de 520 ms se demora cada pata para regresar a su estado inicial, calculado 520 ms por cada pata, el total de tiempo de las seis patas al desplazarse de 3120 ms, que es el tiempo que la estructura del robot tarda en dar un paso, como se muestra en la tabla (24).

Tiempo en Milisegundo	Tiempo en Segundos	# de pasos
3120 ms	3,12 segundos	Un paso

Tabla 24: El tiempo correspondiente a un paso

Fuente: Adriana Hidalgo Panchana

Distancia en avanzar cuando da un paso

Para conocer cuánto equivale la distancia en un paso se prueba encendiendo el robot mientras este camina se verifica en el monitor serial cuando ya haya dado un paso se detiene el proceso. Se mide la distancia tomando en cuenta el punto de partida en el que comenzó a caminar hasta donde paro el proceso.

En la tabla (25) se especifica a cuantos centímetros corresponde un paso:

Segundo y # Paso	Distancia en centímetros
1 paso =3,12 segundos	4cm de avance

Tabla 25: Distancia de un paso

Fuente: Adriana Hidalgo Panchana

Para comenzar a programar los servomotores se aplicó un método experimental el cual consiste en ubicar la posición y grados del servomotor puesto en las patas en función de la posición inicial requerida del robot, de manera que los movimientos de las patas del robot no interfieran al momento de detectar y también para que no choque con la ubicación de la pinza.

VALORES DE REFERENCIA

Los valores de referencia de los ángulos de los 18 servomotores utilizados para las extremidades del robot hexápodo determinaran la orientación de las articulaciones para luego organizar los diferentes estados del robot para su desplazamiento.

A continuación en esta tabla (26) se detallan los valores de ángulos que se tendrán como referencia al momento de coordinar movimientos de las extremidades.

VALORES DE REFERENCIA					
# de extremidad	# de servomotor	# de articulaciones	Incremento	Referencia	Decremento
Pata 1	servomotor 1	Inicial 1	50°	60°	105°
	servomotor 2	Medio 1	150°	120°	-
	servomotor 3	Final 1	-	120°	-
Pata 2	servomotor 4	Inicial 2	150°	160°	170°
	servomotor 5	Medio 2	160°	130°	-
	servomotor 6	Final 2	-	170°	-
Pata 3	servomotor 7	Inicial 3	150°	160°	170°
	servomotor 8	Medio 3	160°	130°	-
	servomotor 9	Final 3	-	120°	-
Pata 4	servomotor 10	Inicial 4	120°	110°	100°
	servomotor 11	Medio 4	5°	35°	-
	servomotor 12	Final 4	-	70°	-
Pata 5	servomotor 13	Inicial 5	100°	90°	80°
	servomotor 14	Medio 5	35°	65°	-
	servomotor 15	Final 5	-	65°	-
Pata 6	servomotor 16	Inicial 6	-	0°	65°
	servomotor 17	Medio 6	35°	65°	-
	servomotor 18	Final 6	-	65°	-

Tabla 26: Valores de referencia de ángulos para las extremidades

Fuente: Adriana Hidalgo Panchana

Cada extremidad contiene tres servomotores y tres articulaciones.

Patatas enumerada de: 1 a la 6

Servomotores de 1 a la 18

Articulaciones: Iniciales 1 a la 6, Medias 1 a la 6, Finales 1 a la 6.

El primer estado de Pie

Los ángulos asignados para que el robot hexápodo ubique sus extremidades de pie y en estado de paro, son los mostrados en la siguiente tabla (27), estos grados fueron cambiados varias veces para que la estructura del robot se mantenga estable al momento de estar de pie.

ESTADO DE PIE DEL ROBOT HEXÁPODO			
# de extremidad	# de servomotor	# de Servos	Ángulos
Pata 1	servomotor 1	Inicial 1	60°
	servomotor 2	Medio 1	120°
	servomotor 3	Final 1	120°
Pata 2	servomotor 4	Inicial 2	160°
	servomotor 5	Medio 2	130°
	servomotor 6	Final 2	170°
Pata 3	servomotor 7	Inicial 3	160°
	servomotor 8	Medio 3	130 °
	servomotor 9	Final 3	120°
Pata 4	servomotor 10	Inicial 4	110°
	servomotor 11	Medio 4	35°
	servomotor 12	Final 4	70°
Pata 5	servomotor 13	Inicial 5	90°
	servomotor 14	Medio 5	65°
	servomotor 15	Final 5	65°
Pata 6	servomotor 16	Inicial 6	0°
	servomotor 17	Medio 6	65°
	servomotor 18	Final 6	65°

Tabla 27: Estados del prototipo de robot en pie

Fuente: Adriana Hidalgo Panchana

EXPERIMENTO 1*

Caminar hacia adelante

Objetivo

El robot hexápodo móvil realizara las acciones de caminar hacia adelante debido a los movimientos coordinados en la codificación.

Criterios de Éxito

Si el robot se desplaza correctamente hacia adelante, durante el periodo de 5 pasos = 15,60 segundos, si durante este periodo el robot llega a desplazarse entre 20 a 21 cm, se considera que el experimento es exitoso.

Experimento

El experimento inicia ubicando al robot hexápodo en un lugar firme y con el objeto a recoger dentro de un área de 50 x 50 cm, luego de esto se suministra de energía al robot y se procede a cargar la codificación en la tarjeta y se controlará mediante la interfaz de usuario.

Para comparar las medidas luego de realizadas las diez pruebas físicas se marcaba con un lápiz en la superficie plana con números, para luego comparar cada uno de los diez resultados y completar la tabla que se mostrará posteriormente.

El cálculo de la velocidad se llevará a cabo luego de realizar las 10 pruebas y conocer la distancia recorrida en 5 pasos con estos dos valores procedemos a calcular la velocidad mediante esta ecuación:

$$V = \frac{d}{t}$$

En la siguiente tabla (28) se muestra las 10 pruebas realizadas, manifestando así si el resultado es exitoso o no.

# Pruebas	Tiempo= 5 pasos	Distancia En cm recorrida	Velocidad calculada	% Avance	Resultado
1	15,60 s.	20,5 cm=0,205m	13,1 x 10 ⁻³ m/s	100%	Exitoso
2	15,60 s.	21,1cm=0,211 m	13,5 x 10 ⁻³ m/s	100%	Exitoso
3	15,60 s.	20,2 cm= 0,202 m	12,9 x 10 ⁻³ m/s	100%	Exitoso
4	15,60 s.	22 cm= 0,22 m	14,1 x 10 ⁻³ m/s	70%	No Exitoso
5	15,60 s.	21,1cm=0,211 m	13,5 x 10 ⁻³ m/s	100%	Exitoso
6	15,60 s.	20,5 cm= 0,205 m	13.1x 10 ⁻³ m/s	100%	Exitoso
7	15,60 s.	20 cm= 0,20 m	12,8 x 10 ⁻³ m/s	100%	Exitoso
8	15,60 s.	22,3 cm=0,223 m	14,2 x 10 ⁻³ m/s	70%	No Exitoso
9	15,60 s.	20,2 cm=0,202 m	12,9 x 10 ⁻³ m/s	100%	Exitoso
10	15,60 s.	22,3 cm= 0,223 m	14,2 x 10 ⁻³ m/s	70%	No Exitoso

Tabla 28: Resultados de robot cuando camina hacia delante

Fuente: Adriana Hidalgo Panchana

El resultado se considera exitoso debido al porcentaje obtenido de las 10 pruebas realizadas esto es un 70% de cumplimiento.

En la siguiente figura (65) se muestra la distancia que el robot hexápodo recorrerá.

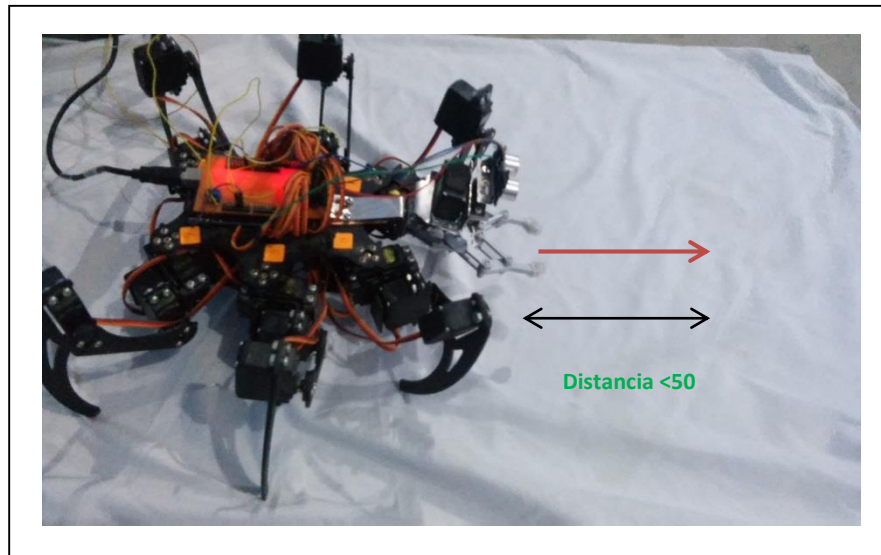


Figura 65: Hexapodo Caminando hacia adelante

Fuente: Adriana Hidalgo Panchana

****EXPERIMENTO 2**** **Girar hacia la izquierda**

Objetivo

El robot hexápodo móvil realizará las acciones de girar hacia la izquierda debido a los movimientos coordinados enviados desde la tarjeta de control.

Criterios de Éxito

Si el robot gira hacia la izquierda, con cada uno de las acciones sin caerse o quedarse inmóvil durante 5 pasos=15,60 segundos, dentro del rango comprendido entre 10 a 15 grados se considera que el experimento es exitoso.

Experimento

El experimento inicia ubicando al robot hexápodo en un lugar firme y con el objeto a recoger dentro de un área de 50 x 50 cm, luego de esto se suministra de energía al robot y se procede a cargar la codificación en la tarjeta se controla mediante la interfaz de usuario.

En la siguiente tabla (29) se tendrá los resultados cuando el robot hexápodo gira hacia la izquierda.

# Pruebas	Tiempo	Grados de Giro	Porcentaje de avanzar	Resultado
1	5 pasos= 15,60 segundos	17°	50%	No Exitoso
2	5 pasos= 15,60 segundos	10°	100%	No exitoso
3	5 pasos= 15,60 segundos	10°	100%	Exitoso
4	5 pasos= 15,60 segundos	20°	50%	No Exitoso
5	5 pasos= 15,60 segundos	20°	50%	No Exitoso
6	5 pasos= 15,60 segundos	15°	100%	Exitoso
7	5 pasos= 15,60 segundos	10°	100%	Exitoso
8	5 pasos= 15,60 segundos	10°	100%	Exitoso
9	5 pasos= 15,60 segundos	15°	100%	Exitoso
10	5 pasos= 15,60 segundos	22°	50%	No Exitoso

Tabla 29: Resultados cuando robot camina hacia izquierda

Fuente: Adriana Hidalgo Panchana

Como se muestra en la tabla las pruebas dentro del rango establecido de giro fue entre 10 a 15 grados, como resultado final el porcentaje ha sido de un 60 % realizado con éxito.

En la siguiente figura (66) se muestra cuando el robot hexápodo gira hacia la izquierda.

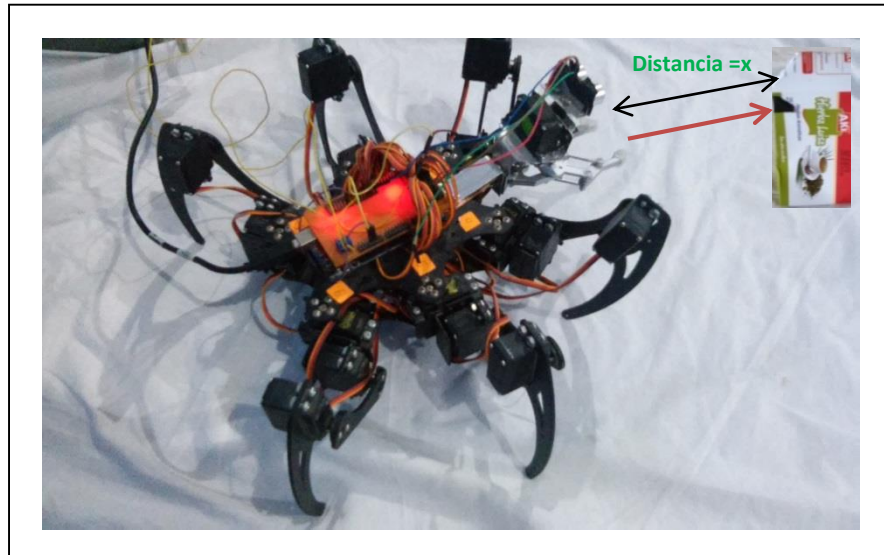


Figura 66: Robot Hexápodo Girando a la izquierda

Fuente: Adriana Hidalgo Panchana

****EXPERIMENTO 3****

Gira hacia la derecha

Objetivo

El robot hexápodo móvil realizará la acción de girar hacia la derecha según los ángulos asignados enviados desde la tarjeta de control.

Criterios de Éxito

Si el robot gira hacia la derecha, con cada uno de las acciones sin caerse o quedarse inmóvil durante 5 pasos=15,60 segundos, dentro del rango comprendido entre 10 a 15 grados se considera que el experimento es exitoso.

Experimento

El experimento inicia ubicando al robot hexápodo en un lugar firme y con el objeto a recoger dentro de un área de 50 x 50 cm, luego de esto se suministra de energía al robot y se procede a cargar la codificación en la tarjeta, se controla mediante la interfaz de usuario.

En la siguiente tabla (30) se tendrá los resultados cuando el robot hexápodo gira hacia la derecha.

# Pruebas	tiempo	Grados de Giro	Porcent aje de avanzar	Resultado
1	5 pasos= 15,60 segundos	10°	100%	Exitoso
2	5 pasos= 15,60 segundos	12°	50%	No Exitoso
3	5 pasos= 15,60 segundos	15°	100%	Exitoso
4	5 pasos= 15,60 segundos	15°	100%	Exitoso
5	5 pasos= 15,60 segundos	20°	50%	No Exitoso
6	5 pasos= 15,60 segundos	15°	100%	Exitoso
7	5 pasos= 15,60 segundos	14°	100%	Exitoso
8	5 pasos= 15,60 segundos	10°	100%	Exitoso
9	5 pasos= 15,60 segundos	14°	100%	Exitoso
10	5 pasos= 15,60 segundos	5°	100%	Exitoso

Tabla 30: Resultados cuando robot gira a la derecha

Fuente: Adriana Hidalgo Panchana

Como se muestra en la tabla las pruebas dentro del rango establecido de giro fue entre 10 a 15 grados, como resultado final el porcentaje ha sido de un 80 % realizado con éxito.

En la siguiente figura (67) se muestra cuando el robot hexápodo gira hacia la izquierda.

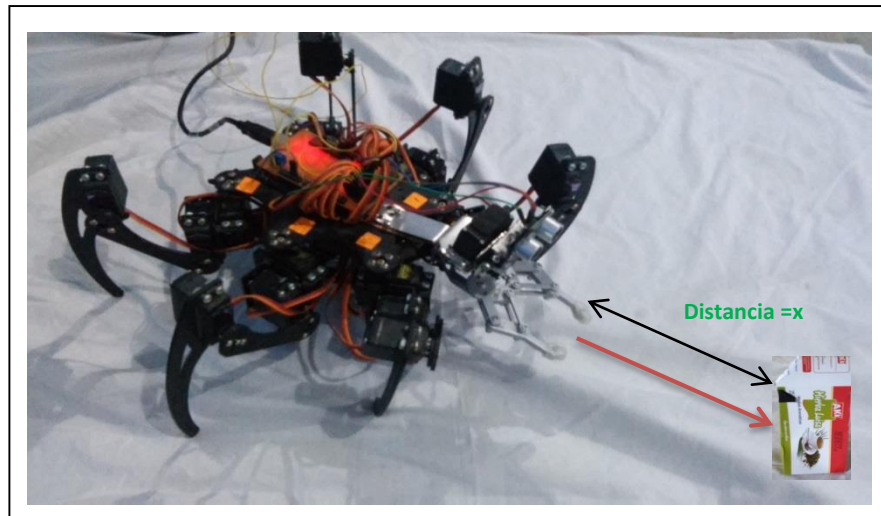


Figura 67: Robot Hexápodo Girando hacia la derecha

Fuente: Adriana Hidalgo Panchana

****EXPERIMENTO 4****

Pruebas con el Sistema de Detección

Para realizar los experimentos con respecto a la detección tomaremos como muestra tres valores detectados por el giro del sensor a través del servomotor. El número de pruebas que se realizarán por experimento son diez, la elaboración de estas pruebas son muy importantes ya que de estas depende la detección del objeto por el robot. Se comprueba cada experimento si son válidas según la repetición de los resultados.

Posición 1: detectado en el mismo ángulo tres veces.

Objetivo

El sensor ultrasónico detectará un objeto y el valor mostrado en la interfaz Arduino-Monitor Serial será el mismo repetido tres veces.

Criterios de Éxito

Al momento de manipular el sensor y que detecte el mismo valor de distancia tres veces en el mismo ángulo de movimientos consecutivas significa que el experimento es correcto, si por lo contrario solo repite dos valores se le da el nombre de valor cercano.

Experimento

Al tener armada la estructura de la pinza robótica con los servomotores y el sensor ultrasónico listo para proceder a detectar dentro de las posiciones de giro establecidas, se suministrará de energía al robot hexápodo y se realiza la pruebas dentro de un área de detección de 50 x 50 cm, con un solo objeto cerca para detectar.

Cuando el sensor llega a medir la distancia en la posición 0 que tiene el ángulo de 40 grados, debe ser el mismo detectado en tres repeticiones consecutivas este valor es visualizado en la interfaz arduino. Tras las 10 pruebas realizadas analizaremos los resultados obtenidos.

Al colocar estos nombres en el resultado significa:

Correcto: tres valores fueron correctos.

Valor cercano: dos valores fueron correctos.

Valor lejano: un valor fue correcto.

En la siguiente tabla (31) se muestra el resultado del experimento.

# Pruebas	Posiciones			Resultado
	1er valor	2er valor	3er valor	
1	1	0	2	Valor cercano
2	0	0	2	Valor lejano
3	0	0	0	Correcto
4	0	0	0	Correcto
5	1	0	0	Valor lejano
6	0	0	0	Correcto
7	0	0	0	Correcto
8	0	0	0	Correcto
9	0	1	0	Valor lejano
10	1	0	0	Valor lejano

Tabla 31: Resultados del valor de detección de la posición 0

Fuente: Adriana Hidalgo Panchana

Como resultado final el porcentaje:

Correcto: 5 valores, 50%

Valor cercano: 1 valores, 10%

Valor lejano: 4 valor, 40%

****EXPERIMENTO 5****

Posición 2 detectado en el mismo ángulo tres veces.

Objetivo

El sensor ultrasónico detectará un objeto y el valor mostrado en la interfaz Arduino-Monitor Serial será el mismo repetido tres veces.

Criterios de Éxito

Al momento de manipular el sensor y que detecte el mismo valor de distancia tres veces, en el mismo ángulo de movimientos, consecutivos significa que el experimento es correcto.

Experimento

Al tener armada la estructura de la pinza robótica con los servomotores y el sensor ultrasónico listo para proceder a detectar dentro de las posiciones de giro establecidas, se suministrar de energía al robot hexápodo y se realiza la pruebas dentro de un área de detección de 50 x 50 cm, con un solo objeto cerca para detectar. Cuando el sensor llega a medir la distancia en la posición 3 que tiene el ángulo de 80 grados, debe ser el mismo detectado en tres repeticiones consecutivas este valor es visualizado en la interfaz arduino. Tras las 10 pruebas realizadas analizaremos los resultados obtenidos.

Al colocar estos nombres en el resultado significa:

Correcto: tres valores fueron correctos.

Valor cercano: dos valores fueron correctos.

Valor lejano: un valor fue correcto.

En la tabla (32) se muestra los resultados de este experimento donde tuvo como respuestas las 10 pruebas la mayoría fue certero.

# Pruebas	Posiciones			Resultado
	1er valor	2er valor	3er valor	
1	2	3	3	Valor Cercano
2	3	3	2	Valor Cercano
3	2	2	3	Valor lejano
4	3	3	3	Correcto
5	3	3	3	Correcto
6	3	3	3	Correcto
7	3	3	3	Correcto
8	3	3	3	Correcto
9	3	3	3	Correcto
10	3	3	3	Correcto

Tabla 32: Resultados del valor de detección de la posición 4

Fuente: Adriana Hidalgo Panchana

Como resultado final el porcentaje:

Correcto: 7 valores, 70%

Valor cercano: 2 valores, 20%

Valor lejano: 1 valor, 10%

Ejemplos

Antes de empezar a realizar el experimento # 6: Prueba Final del Robot hexápodo. Se mostrara unos ejemplos sobre el sistema de control: detección del objeto, trazado de trayectoria y recolección del objeto, realizados para entender mejor la función a cumplir por el hexápodo este contiene: cálculo del tiempo, cálculo de la velocidad y ángulo de giro.

Ejemplos #1

Si el objeto se encuentra 30 cm de distancia y el robot avanza de la posición 0 a la posición 5, calcule el tiempo, la velocidad, y ángulo de giro si sabemos que cada paso demora = 3,12 segundos. En la figura (68) se muestra la trayectoria a seguir por el robot.

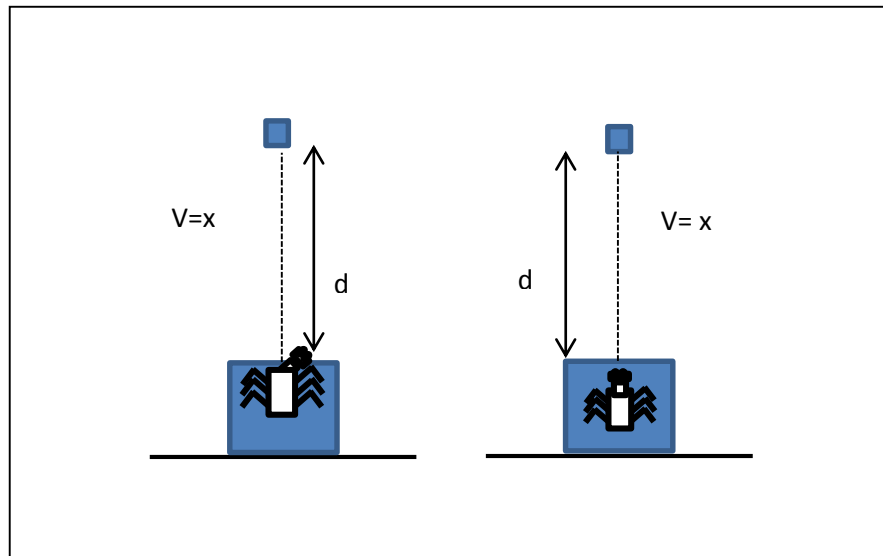


Figura 68: Trazado de trayectoria de la posición 0 a la 5

Fuente: Adriana Hidalgo Panchana

Datos

$t = 7,5$ pasos 23,4 segundos

$d = 30\text{cm} = 0,3$ m

$V=x$

Cálculo del tiempo

$$\begin{aligned} 1 \text{ paso} &= 4 \text{ cm} & 1 \text{ paso} &= 3,12 \text{ s.} \\ x &= 30 \text{ cm} & 7,5 \text{ pasos} &= t \\ x &= \frac{1 \text{ paso} * 30 \text{ cm}}{4 \text{ cm}} & t &= \frac{7,5 \text{ pasos} * 3,12 \text{ s}}{1 \text{ paso}} \\ x &= 7,5 \text{ pasos} & t &= 23,4 \text{ segundos} \end{aligned}$$

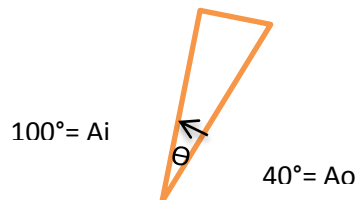
Cálculo de la velocidad

$$V = \frac{d}{t}$$

$$V = \frac{0,3 \text{ m}}{23,4 \text{ s}}$$

$$V = 12,82 \times 10^{-3} \text{ m/s} \quad //$$

Cálculo ángulo de giro



$$\emptyset = A_o - A_i$$

$$\emptyset = 40^\circ - 100^\circ$$

$$\emptyset = -60^\circ \text{ De desplazamiento}$$

A_i = Posición inicial de medición.

A_o = Posición donde se encontró al objeto.

Ejemplo #2

Si el objeto se encuentra 40 cm de distancia y el robot avanza de la posición 0 a la posición 2, calcule el tiempo, la velocidad y ángulo de giro si sabemos que cada paso demora = 3,12 segundos. En la figura (69) se muestra la trayectoria a seguir por el robot.

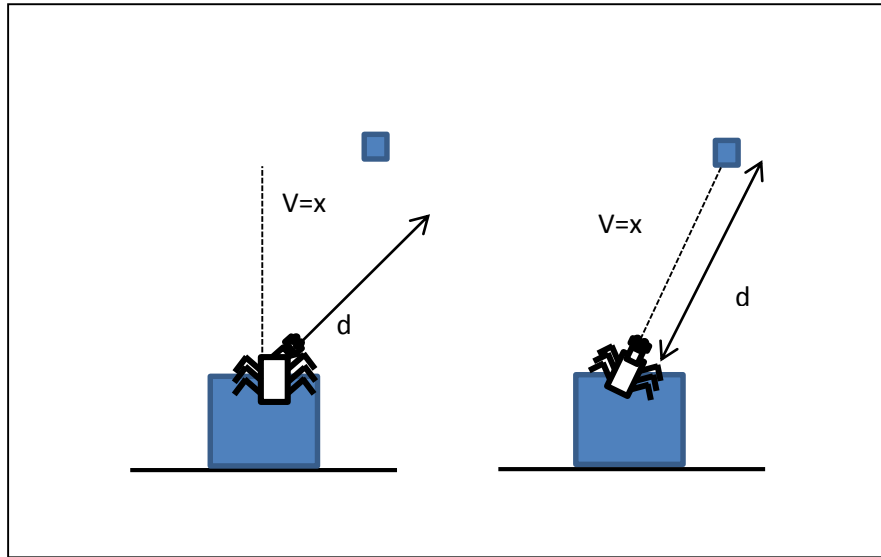


Figura 69: Trazado de una trayectoria de la posición 0 a la 2

Fuente: Adriana Hidalgo Panchana

Datos

$t = 10 \text{ pasos} = 31,2 \text{ segundos}$

$V = x$

$d = 40 \text{ cm} = 0,4 \text{ m}$

Cálculo del tiempo

$$\begin{aligned} 1 \text{ paso} &= 4 \text{ cm} \\ x &= 40 \text{ cm} \\ x &= \frac{1 \text{ paso} * 40 \text{ cm}}{4 \text{ cm}} \\ x &= 10 \text{ pasos} \end{aligned}$$

$$\begin{aligned} 1 \text{ paso} &= 3,12 \text{ s.} \\ 10 \text{ pasos} &= t \\ t &= \frac{10 \text{ pasos} * 3,12 \text{ s}}{1 \text{ paso}} \\ t &= 31,2 \text{ segundos} \end{aligned}$$

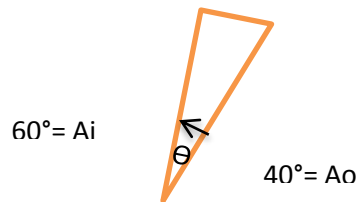
Cálculo de velocidad

$$V = \frac{d}{t}$$

$$V = \frac{0,4m}{31,20 s}$$

$$V = 12,82 \times 10^{-3} m/s \quad //$$

Cálculo ángulo de giro



$$\phi = A_o - A_i$$

$$\phi = 40^\circ - 60^\circ$$

$$\phi = -20^\circ \text{ De desplazamiento}$$

Ai= Posición inicial de medición.

Ao= Posición donde se encontró al objeto.

Experimento 6

Prueba final del Robot hexápodo

Objetivo

La Implementación del prototipo robot hexápodo que escogerá la trayectoria hacia el objeto detectado, buscando la dirección correcta, giros y avances hasta llegar al punto mínimo de distancia, cuando el robot llegue a este punto se detendrá y la pinza robótica se cerrará.

Criterios de Éxito

La prueba es exitosa si al momento de colocar un objeto a una distancia mínima de 50 cm de trayectoria, el sensor de distancia lo detecta y escoge el ángulo donde fue ubicado para dirigirse hacia el objeto, enviando una señal a la parte de control, como el objeto fue detectado ahora se sigue ordenando a las extremidades de que estas se muevan y realicen la acción de avanzar, girar, toma la posición de reposo, agarrar y luego regresa a su estado inicial, el estado de pie. En caso contrario que el objeto no haya sido detectado el robot hexápodo seguirá girando hasta encontrar algo que detectar.

Experimento

La prueba del prototipo se realiza en un área de 50x50, este lugar debe estar libre de obstáculos, las circunstancias del ambiente deben ser un suelo aplanado y firme, que no exista nada más que el objeto a detectar.

El robot hexápodo comienza girando de derecha a izquierda tratando de ubicar algo en su alrededor, cuando detecte algo, el robot se ubicará y avanzará hasta una cierta distancia mínima de 8 cm para que la pinza recoja al objeto. También se tomara otros valores de esta prueba como los segundos por cada paso, velocidad, la

posición de donde se detecta el objeto y posición donde se lo recoge. Esta prueba se la va a realizar utilizando la fuente de alimentación de 5 voltios.

En la tabla (33) se encuentran las pruebas de los resultados finales.

# Pruebas	Distancia	# Pasos=segundos	Velocidad	Posición donde se detectó el Obj.	%	Resultado: Si o No
1	8 cm	2 =6,24 s	$3,04 \times 10^{-3} \text{m/s}$	3	100%	Exitoso
2	10 cm	2,5=7,8 s	$1,28 \times 10^{-2} \text{m/s}$	2	100%	Exitoso
3	15cm	3,5 = 10,92s	$1,37 \times 10^{-2} \text{m/s}$	2	100%	No Exitoso
4	20 cm	5 =15,60 s	$1,28 \times 10^{-2} \text{m/s}$	1	100%	Exitoso
5	25 cm	6,1=18,72 s	$1,33 \times 10^{-2} \text{m/s}$	4	100%	Exitoso
6	30 cm	7,2=21,84 s	$1,37 \times 10^{-2} \text{m/s}$	3	100%	Exitoso
7	35 cm	8,5=26,52 s	$1,32 \times 10^{-2} \text{m/s}$	1	0%	No Exitoso
8	40 cm	10= 31,20	$1,28 \times 10^{-2} \text{m/s}$	4	100%	Exitoso
9	45 cm	11=34,32 s	$1,31 \times 10^{-2} \text{m/s}$	5	0%	No Exitoso
10	50 cm	12= 37,44 s	$1,33 \times 10^{-2} \text{m/s}$	6	0%	No Exitoso

Tabla 33: Resultados del Prueba final

Fuente: Adriana Hidalgo Panchana

Como resultado final el prototipo de robot hexápodo tiene un 70 % realizado con éxito. En el Anexo #5 se mostrará la simulación final en Proteus 8 Profesional del circuito.

En la figura (70) se muestra el trayecto que tiene el prototipo del robot hexápodo hacia el objeto.

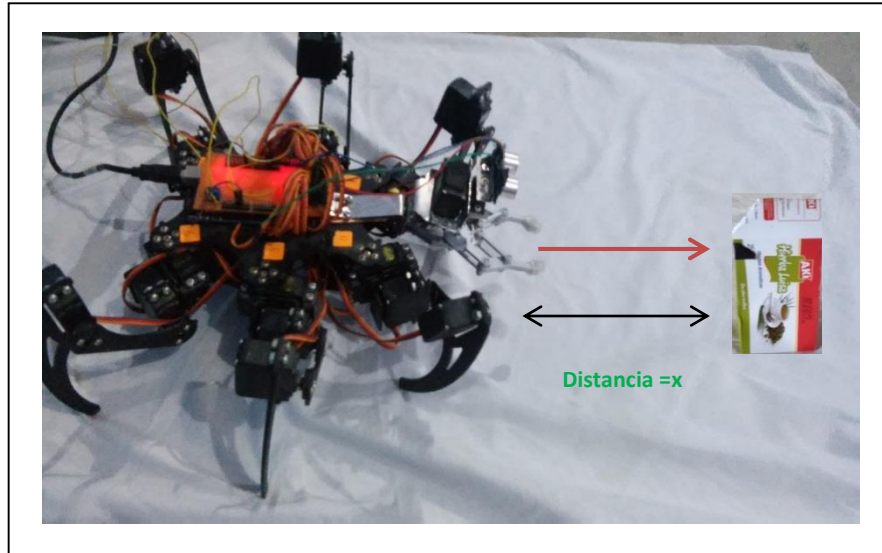


Figura 70: Prueba Final del robot hexápodo

Fuente: Adriana Hidalgo Panchana

5.3. Documentación

Se describirá el manual de usuario para facilitar el funcionamiento y mantenimiento del Robot Hexápodo para la contribución de la limpieza. El manual permitirá al usuario tener una guía de información para: descargar las librerías, cargar la codificación en la tarjeta arduino, realizar el control por la interfaz de arduino, etc. El documento del Manual Usuario lo encontraremos en el Anexo 9.

5.4. Demostración de la Hipótesis

Finalizadas las pruebas del prototipo del robot móvil hexápodo se realizó la evaluación de los resultados del control de movimientos del robot, verificando así que el robot se encuentre preparado para la identificación y recolección del objeto; la identificación del objeto se verificó realizando el experimento # 4 con un 60% de eficiencia, el experimento # 5 con un 90% de eficiencia y en cuanto a la recolección del objeto el experimento # 6 con un valor de 70% de eficiencia al momento de recolectar el objeto.

CONCLUSIONES

De acuerdo al desarrollo del tema se ha concluido en los siguientes puntos:

- El algoritmo de control que manipula los movimientos del robot fue debidamente comprobado bajo los resultados obtenidos mediante los 6 experimentos de prueba y que por lo menos 4 de estos cumplieron correctamente su ejecución, de tal manera que se puede decir que la estructura del robot y la señales de control de robot han sido debidamente calibradas y acopladas.
- El estudio del movimiento cinemático de pata es similar a un brazo robot con tres grados de libertad, los cálculos obtenidos de acuerdo a la cinemática del sistema del robot son comparados con las pruebas obtenida de los experimentos 1, 2, 3, estos experimentos dan como resultado: el experimento con un 70 % de éxito, el experimento 2 con 60% de éxito y el experimento 3 con un 80% de éxito; lo cual demuestra que existe la coordinación adecuada del robot y la lógica secuencial y ordenada del código de control con la estructura.
- El resultado de velocidad de 12.8×10^{-3} m/s obtenido en este prototipo de robot, fue determinado mediante el cálculo del uso de distancia recorrida y tiempo de desplazamiento, en base los ensayos de prueba del experimento 1, se menciona que la velocidad del robot hexápodo tiene un retardo mucho menor de rapidez al paso de una persona q es de 1.11 m/s como no es importante la velocidad para el proceso, se considera útil esta velocidad para el movimiento del robot
- El algoritmo de control para la detección del objeto fue analizado por los siguientes procedimientos como: posiciones de giro, movimiento del servomotor, la detección y recolección del objeto-basura; estas funciones contienen parámetros de posición dentro de un rango de valores de distancia a

detectar menores a 50 cm y mayor a 8 cm. Estos ensayos fueron realizados en varios experimentos; en el experimento 4 y 5, posiciones de giro de la detección, obteniendo como resultado un 60 y 90% respectivamente y la recolección y detección del objeto en el experimento 6 obteniendo un resultado de un 70% por lo tanto se comprueba que la detección de objeto es efectuada de manera correcta.

- La implementación del robot hexápodo que incluye la función de recoger objetos-basura para la contribución de la limpieza, ha sido acoplado y calibrado para su correcto funcionamiento y variación. Estos procedimientos involucran una nueva innovación para el robot.

RECOMENDACIONES

- Si se desea agregar otros elementos a la estructura de robot hexápodo se recomienda cambiar los servomotores con unos de mayor torque.
- Al momento de agregar baterías, se debe tener en cuenta la cantidad de corriente necesaria en todo el prototipo, ya que si solo nos fijamos en aumentar el voltaje sin tener en cuenta la corriente, este tendría como resultado el daño de los equipos.
- Con referencia a la tarjeta adaptable, por motivo de que algunos pines puedan estar dañados, se recomienda verificar el fluido de corriente proveniente de la tarjeta mega arduino hacia cada uno de los pines de los servomotores.
- La tierra de la tarjeta mega arduino esté alimentando a la tierra de los servomotores, sin esta conexión el robot tendrá movimientos descoordinados.
- El territorio donde se realicen las demostraciones debe ser plano y sin obstáculos.

BIBLIOGRAFÍA

1. Barrientos A. (2007). *Fundamentos de Robótica*. México: Mc Graw Hill.
2. Baturone, Anibal Ollero. (2005). *Robótica: manipuladores y robots móviles*. Marcombo.
3. Cortes Pichardo, Gamero María. (2009). *Tesis: Robot Limpiador de Playa. Previo a la obtención del título de Ingeniero eléctrico y Electrónico e Ingeniero en Computación*. México.
4. De la Rosa, Nelson. (2014). *Tesis "Brazo robot, para el reconocimiento y manipulación de objetos, controlado mediante inteligencia artificial, como complemento de tecnología robótica en la carrera de electrónica y telecomunicaciones de la universidad estatal península de santa Elena. Tesis previa a la obtención de Ingeniero Electrónica y Telecomunicaciones*. Santa Elena, Ecuador.
5. J.Craig, John. (1989). *Introducción a la Robótica, Mecanismos y Control*. México: Person educación. *Robótica 3ra Edición*.
6. Jorge Escobar, Juan silva. (2011). *Diseño de Sistema Electrónico Jerárquico mediante comandos par robot Hexápodo. Previo a la obtención de Ingeniero Electrónico*. Colombia, Bogotá: Pontificia Universidad Javeriana.
7. Juan Fernández. (2012). *Diseño y Construcción de un Robot Hexápodo, hardware y Software de Control. Previo a la obtención del título de Ingeniero en Sistemas y Automática*. Universidad de Málaga.
8. Marticorena, José Molina. (2006). *Tecnología de Control*. Argentina.
9. Martín, Juan Carlos. (2012). *Maquinas Eléctricas*. editex, 2012.
10. SAHA, SUBIR KUMAR. (2010). *Introducción a la Robótica*. Mexico, D.F.: McGraw- HILL.
11. Consejo Superior de Investigaciones Científicas Y Centro Para El Desarrollo Tecnológico Industrial. (1987). *Introducción a los Sensores*. Editorial Alcala De Henares Artes Graficas, S. A.
12. *Robots al Detalle*. (17 de Julio de 2014). *Obtenido de Historia delos Robots Aspiradores: <http://www.robotsaldetalle.es/aspiradores/noticias/historia-de-los-robots-aspiradores-ii#.VYguFPntmko>*
13. Rosique Gómez, A. (s.f.). *ALCABOT*. *Obtenido de Seminario de Diseño y construcción*

deMicrobots:<http://www.alcabot.com/alcabot/seminario2006/Trabajos/AlejandroRosiqueGomez.pdf>

- 14.** *Automodelismo.* (9 de 7 de 2013). *Obtenido de Baterías:*
<http://automodelismo.com/nicd.htm>
- 15.** *Arqhys.* (s.f.). *Obtenido de* <http://www.arqhys.com/articulos/robot-hexapodo.html>
- 16.** *Themes, Elegant.* (s.f.). <http://www.geekfactory.mx/> *Geekfactory.* *Obtenido de* <http://www.geekfactory.mx/tienda/sensores/hc-sr04-sensor-de-distancia-ultrasonico/>

ANEXOS

ANEXO 1: CERTIFICADO GRAMATICAL

CERTIFICACIÓN GRAMATICAL Y ORTOGRÁFICA

En mi calidad de Licenciada de la especialidad de Lengua y Literatura, luego de haber revisado y corregido la tesis **“IMPLEMENTACIÓN DE UN PROTOTIPO DE UN ROBOT MÓVIL TIPO HEXÁPODO TELEDIRIGIDO PARA LA CONSTRUCCIÓN DE LA LIMPIEZA APLICANDO TECNOLOGÍA OPEN SOURCE”**, previa la obtención del Título de ingeniero en electrónica y telecomunicaciones, de la Carrera de Electrónica y Telecomunicaciones, Escuela de Electrónica y Telecomunicaciones, Facultad de Sistemas y Telecomunicaciones de la Universidad Estatal Península de Santa Elena, de la estudiante Adriana Hidalgo Panchana, certifico que está habilitada con el correcto manejo del lenguaje, claridad en la expresión, coherencia en los conceptos, adecuado empleo de la sinonimia, corrección ortográfica y gramatical.

Es cuanto puedo decir en honor a la verdad.

La Libertad, 28 de Agosto del 2015

Lcda. Isabel Camacho

LICENCIADA EN LITERATURA Y PEDAGOGÍA

ANEXO 2: ENCUESTAS

UNIVERSIDAD ESTATAL PENINSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

“IMPLEMENTACION DE UN PROTOTIPO DE UN ROBOT MÓVIL TIPO HEXÁPODO
TELEDIRIGIDO PARA LA CONTRIBUCIÓN EN LA LIMPIEZA APLICANDO
TECNOLOGÍA OPEN SOURCE” previo a la obtención del título de INGENIERA EN
ELECTRÓNICA Y TELECOMUNICACIONES.

ENCUESTA A ESTUDIANTES DE CARRERA DE ELECTRÓNICA Y TELECOMUNICACIONES

1. ¿Conoce usted robots móviles tipo hexápodos que realicen la actividad de recolectar objetos?

Si

No

2. Al momento de ingresar al laboratorio usted lo encuentra totalmente limpio

Si

No

A veces

3. La limpieza es automatizada en el laboratorio de la Carrera de Electrónica y Telecomunicaciones.

Si

No

4. ¿Cree que la implementación de un prototipo de un robot móvil tipo hexápodo teledirigido contribuirá a la limpieza del laboratorio de Electrónica y Telecomunicaciones?

Si

No

5. ¿La implementación del prototipo robot hexápodo utilizando elementos como: la tarjeta de control, servomotores, sensores y programas como Arduino, Proteus es un ejemplo claro de la Robótica y su aplicación?

Si

No

6. Conoce usted acerca del funcionamiento y la capacidad de desplazamiento de los robots hexápodos.

Si

No

7. ¿Piensa usted que al tomar el robot hexápodo teledirigido como instrumento de estudio servirá para incrementar conocimientos tecnológicos a los estudiantes?

Si

No

8. La implementación de un robot hexápodo que contribuya a la limpieza apoyara a disminuir el trabajo del ser humano.

Si

No

9. ¿con los conocimientos previos adquiridos en la carrera usted se siente capaz de implementar robots que ayuden a mejorar el aspecto de la Universidad?

Si

No

10. En el futuro con el avance de la tecnología ¿piensa usted que esta clase de robot serían los únicos encargados de la limpieza?

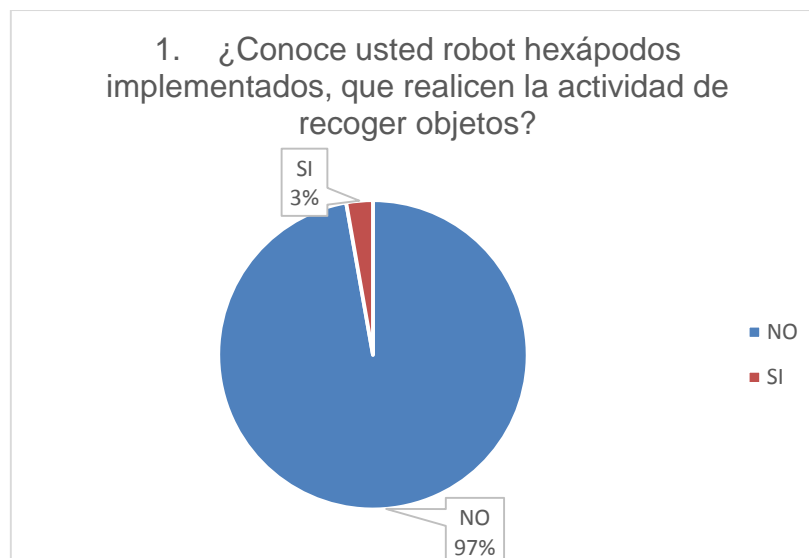
Si

No

RESULTADOS DE LAS ENCUESTAS

1. ¿Conoce usted robots móviles tipo hexápodos que realicen la actividad de recolectar objetos?

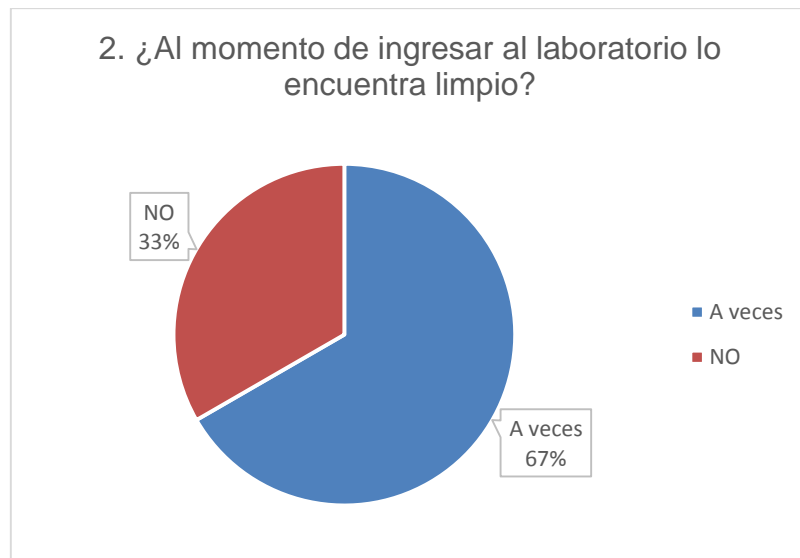
¿Conoce usted robots móviles tipo hexápodos que realicen la actividad de recolectar objetos?		
Alternativas	Resultado	Porcentaje
NO	34	97%
SI	1	3%
Total general	36	100%



Esta pregunta ayuda a conocer si servirá como modelo de estudio a los estudiantes de la carrera al momento de la implementación de este elemento demostrando que existe un total de 97% estudiantes que dicen que NO conocen robot hexápodos que realicen la función de recoger objeto y tan el 3% dice que sí ha visto robots hexápodos que recojan objetos.

2. Al momento de ingresar al laboratorio usted lo encuentra totalmente limpio

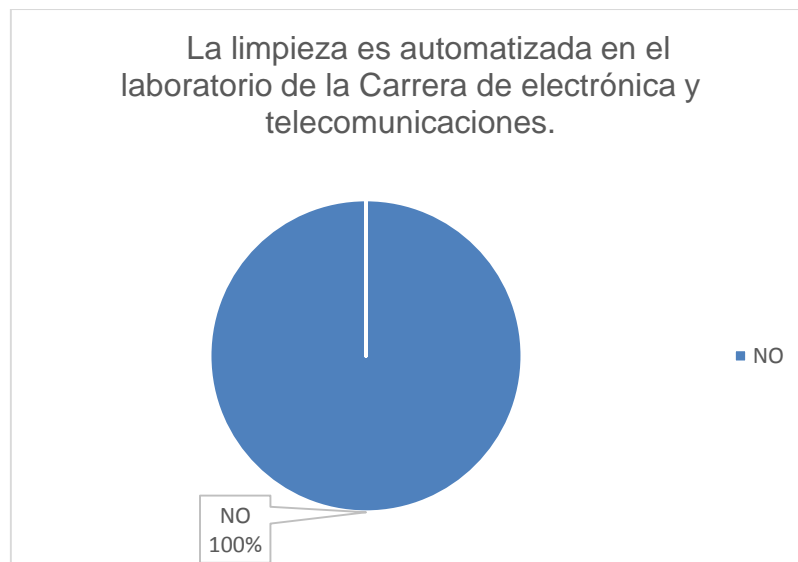
¿Al momento de ingresar al laboratorio lo encuentra limpio?		
Alternativas	Resultado	Porcentaje
A veces	24	67%
NO	12	33%
Total general	36	100%



Esta pregunta hace referencia al objetivo del tema el cual indica que el robot hexápodo se utilizará para limpiar el laboratorio de Electrónica y Telecomunicaciones, en este caso se analizó y se llegó a la conclusión de que si será útil el robot ya que el 67% indica que a veces lo encuentra limpio y el 33% indica que no lo encuentra limpio.

3. La limpieza es automatizada en el laboratorio de la Carrera de Electrónica y Telecomunicaciones.

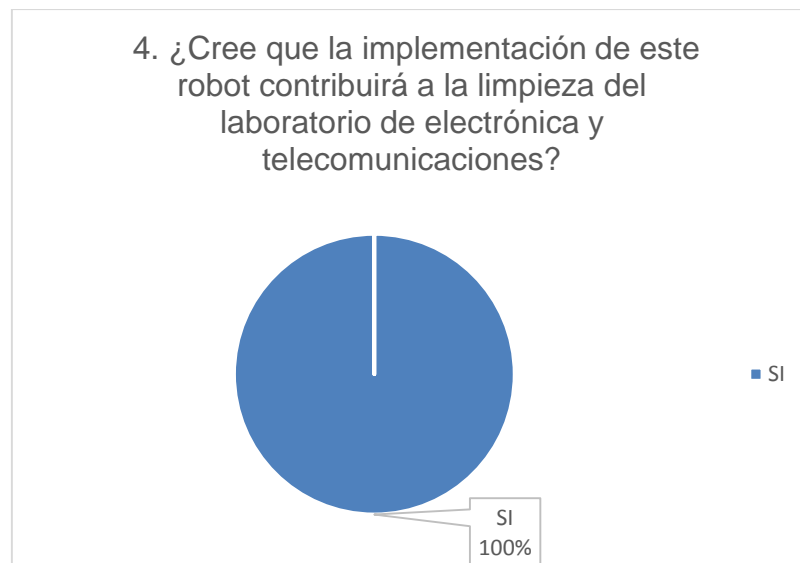
La limpieza es automatizada en el laboratorio de la Carrera de electrónica y telecomunicaciones.		
Alternativas	Resultado	Porcentaje
NO	36	100%
Total general	36	100%



La respuesta de esta pregunta fue un 100% No, ya que la limpieza del laboratorio no es automática sino que la realiza una persona. El robot será útil ya que no existe robot de limpieza en el laboratorio.

4. ¿Cree que la implementación de un prototipo de un robot móvil tipo hexápodo teledirigido contribuirá a la limpieza del laboratorio de electrónica y telecomunicaciones?

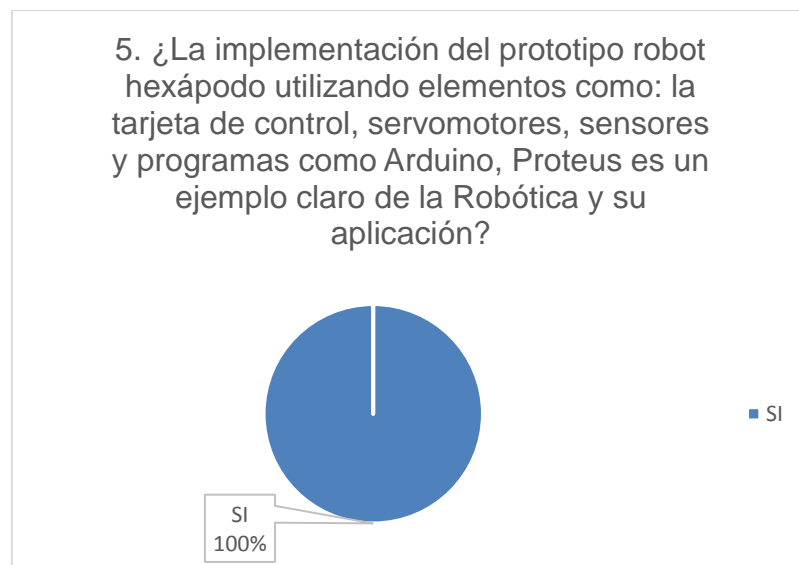
¿Cree que la implementación de un prototipo de un robot móvil tipo hexápodo teledirigido contribuirá a la limpieza del laboratorio de electrónica y telecomunicaciones?		
Alternativas	Resultado	Porcentaje
SI	36	100%
Total general	36	100%



El 100% de las personas indican que el robot si contribuirá a la limpieza del laboratorio. Ya que la finalidad del robot hexápodo es colaborar con la limpieza.

5. ¿La implementación del prototipo robot hexápodo utilizando elementos como: la tarjeta de control, servomotores, sensores y programas como Arduino, Proteus es un ejemplo claro de la Robótica y su aplicación?

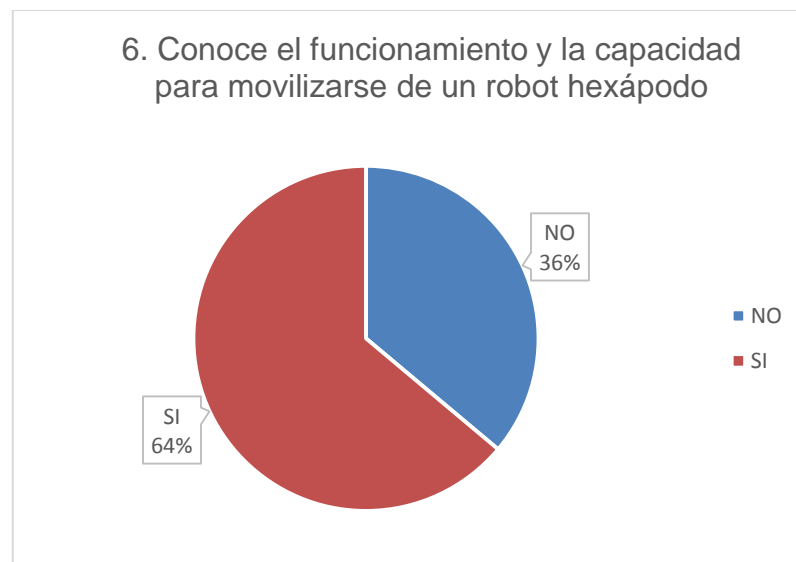
¿La implementación del prototipo robot hexápodo utilizando elementos como: la tarjeta de control, servomotores, sensores y programas como Arduino, Proteus es un ejemplo claro de la Robótica y su aplicación?		
Alternativas	Resultado	Porcentaje
SI	36	100%
Total general	36	100%



Esta pregunta con el 100% aseguran que sí servirá como complemento al estudiante como ejemplo de aplicación.

6. Conoce usted acerca del funcionamiento y la capacidad de desplazamiento de los robots hexápodos

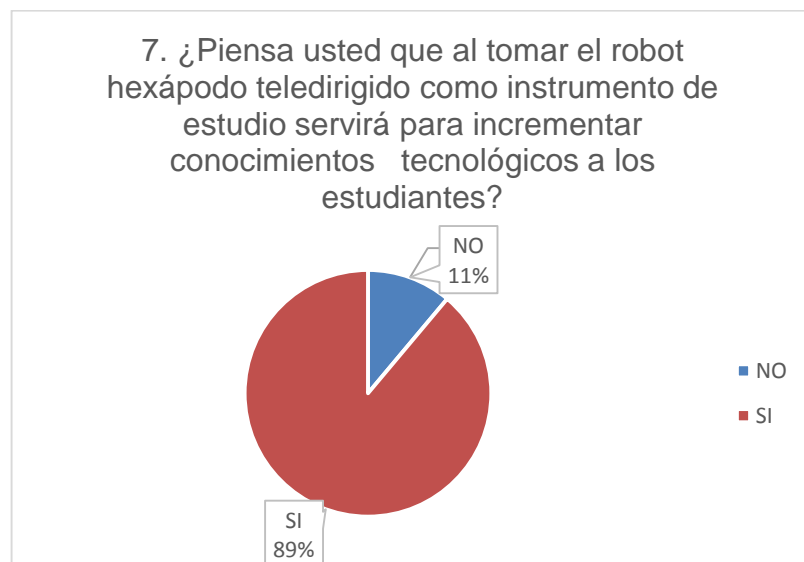
Conoce usted acerca del funcionamiento y la capacidad de desplazamiento de los robots hexápodos		
Alternativas	Resultado	Porcentaje
NO	13	36%
SI	23	64%
Total general	36	100%



Esta pregunta tiene que ver con el funcionamiento cinemático del robot, el 36% indica que no conoce la capacidad de movilidad del robot hexápodo y el 64% indica que sí conoce como funciona, tomando como instrumento la implementación robot hexápodo ayudará a reforzar la teoría aprendida en clases con la práctica.

7. ¿Piensa usted que al tomar el robot hexápodo teledirigido como instrumento de estudio servirá para incrementar conocimientos tecnológicos a los estudiantes?

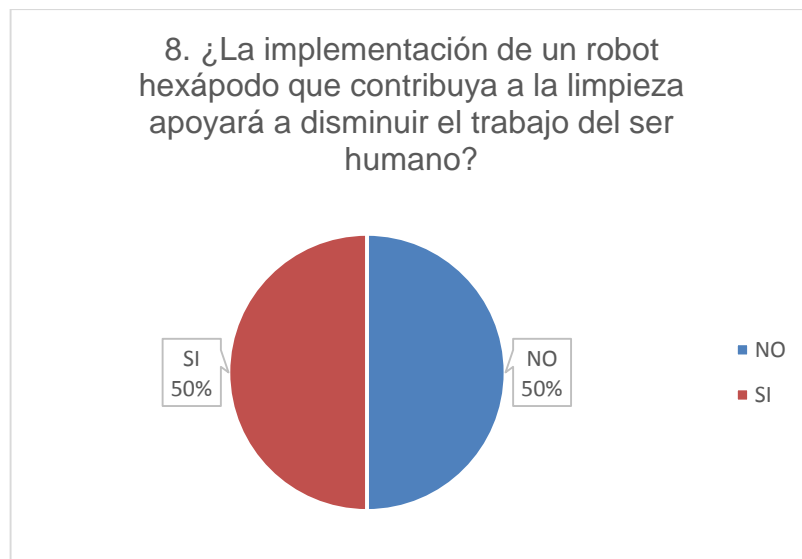
¿Piensa usted que al tomar el robot hexápodo teledirigido como instrumento de estudio servirá para incrementar conocimientos tecnológicos a los estudiantes?		
Alternativas	Resultado	Porcentaje
NO	4	11%
SI	32	89%
Total general	36	100%



El 89% indica que si servirá como instrumento ya que este prototipo demuestra la parte teórica aprendida en la materia de la robótica y les expresa cómo la tecnología es aplicada en robots, mientras el 11% indica que no servirá como instrumento de estudio.

8. ¿La implementación de un robot hexápodo que contribuya a la limpieza, apoyará a disminuir el trabajo del ser humano?

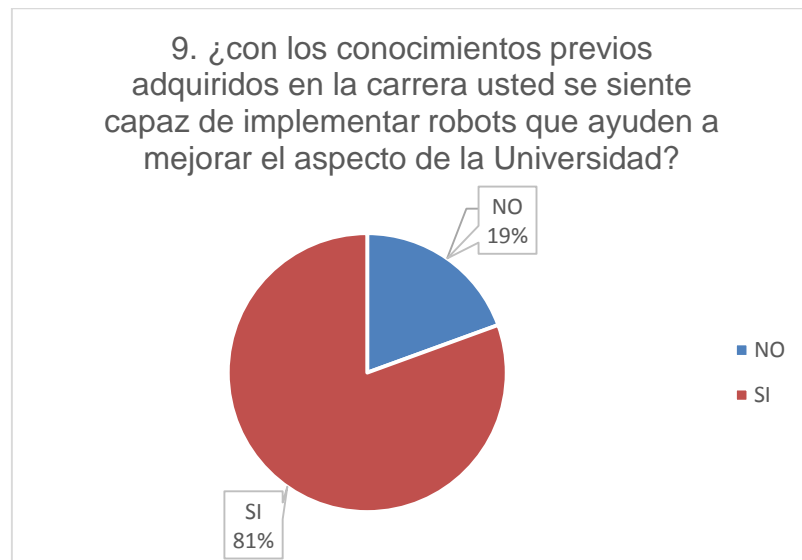
¿La implementación de un robot hexápodo que contribuya a la limpieza apoyará a disminuir el trabajo del ser humano?		
Alternativas	Resultado	Porcentaje
NO	18	50%
SI	18	50%
Total general	36	100%



El 50% de los encuestados aseguran que sí contribuirá ya que este realizará parte de la labor designada al ser humano y para el 50% le parece que no ayudará en nada.

9. ¿Con los conocimientos previos adquiridos en la carrera usted se siente capaz de implementar robots que ayuden a mejorar el aspecto de la Universidad?

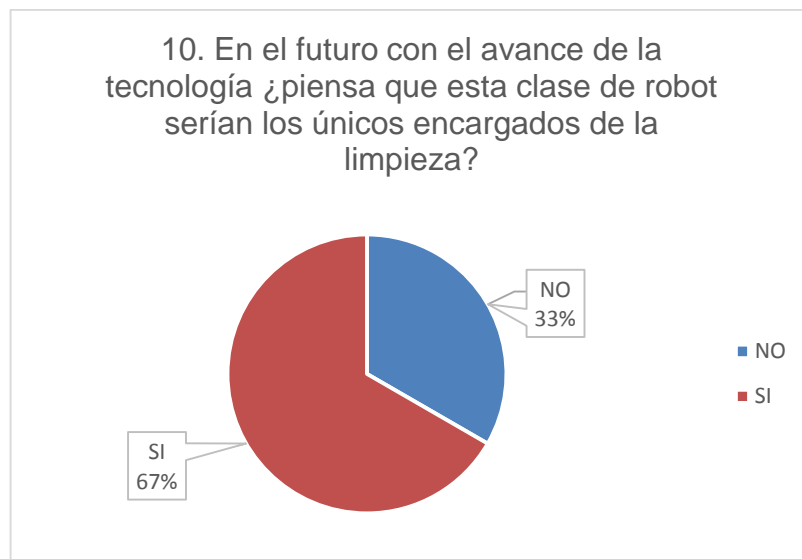
¿Con los conocimientos previos adquiridos en la carrera usted se siente capaz de implementar robots que ayuden a mejorar el aspecto de la Universidad?		
Alternativas	Resultado	Porcentaje
NO	7	19%
SI	29	81%
Total general	36	100%



El 89% indica que sí está capacitado para crear elementos que contribuyan al bienestar Universitario y el 19% no se cree capaz de ayudar a mejorar la calidad de vida de la universidad mediante robots. Sirviendo como motivación la implementación de este robot a los estudiantes.

10. En el futuro con el avance de la tecnología ¿piensa usted que esta clase de robot serían los únicos encargados de la limpieza?

En el futuro con el avance de la tecnología ¿piensa usted que esta clase de robot serían los únicos encargados de la limpieza?		
Alternativas	Resultado	Porcentaje
NO	12	33%
SI	24	67%
Total general	36	100%



El 33% indica que a pesar de todo el trabajo del ser humano será necesario y el 67% confirma que en la futura generación los robots serán los únicos encargados de la limpieza. Esta pregunta se la formula pensando en los robots con la tecnología del futuro.

ANEXO 3: LIBRERÍAS

Librería para codificar en Arduino un Servomotor

/* Servo.h - Interrupt driven Servo library for Arduino using 16 bit timers- Version 2 Copyright (c) 2009 Michael Margolis. All right reserved.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

*/

/* A servo is activated by creating an instance of the Servo class passing the desired pin to the attach() method.

The servos are pulsed in the background using the value most recently written using the write() method

Note that analogWrite of PWM on pins associated with the timer are disabled when the first servo is attached.

Timers are seized as needed in groups of 12 servos - 24 servos use two timers, 48 servos will use four.

The sequence used to seize timers is defined in timers.h

The methods are:

Servo - Class for manipulating servo motors connected to Arduino pins.

attach(pin) - Attaches a servo motor to an i/o pin.

attach(pin, min, max) - Attaches to a pin setting min and max values in microseconds default min is 544, max is 2400

write() - Sets the servo angle in degrees. (invalid angle that is valid as pulse in microseconds is treated as microseconds)

writeMicroseconds() - Sets the servo pulse width in microseconds

read() - Gets the last written servo pulse width as an angle between 0 and 180.

readMicroseconds() - Gets the last written servo pulse width in microseconds. (was read_us() in first release)

attached() - Returns true if there is a servo attached.

detach() - Stops an attached servos from pulsing its i/o pin.

*/

```
#ifndef Servo_h
```

```
#define Servo_h
```

```
#include <inttypes.h>
```

```
/* * Defines for 16 bit timers used with Servo library
```

```
/* * If _useTimerX is defined then TimerX is a 16 bit timer on the current board
```

```
/* timer16_Sequence_t enumerates the sequence that the timers should be allocated
```

```
/* _Nbr_16timers indicates how many 16 bit timers are available.
```

```
/*
```

```
*/ * Say which 16 bit timers can be used and in what order
```

```
#if defined(__AVR_ATmega1280__) || defined(__AVR_ATmega2560__)
```

```

#define _useTimer5
#define _useTimer1
#define _useTimer3
#define _useTimer4
typedef enum { _timer5, _timer1, _timer3, _timer4, _Nbr_16timers } timer16_Sequence_t ;

#elif defined(__AVR_ATmega32U4__)
#define _useTimer1
typedef enum { _timer1, _Nbr_16timers } timer16_Sequence_t ;

#elif defined(__AVR_AT90USB646__) || defined(__AVR_AT90USB1286__)
#define _useTimer3
#define _useTimer1
typedef enum { _timer3, _timer1, _Nbr_16timers } timer16_Sequence_t ;

#elif defined(__AVR_ATmega128__)
||defined(__AVR_ATmega1281__)||defined(__AVR_ATmega2561__)
#define _useTimer3
#define _useTimer1
typedef enum { _timer3, _timer1, _Nbr_16timers } timer16_Sequence_t ;

#else // everything else
#define _useTimer1
typedef enum { _timer1, _Nbr_16timers } timer16_Sequence_t ;
#endif

#define Servo_VERSION      2    // software version of this library

#define MIN_PULSE_WIDTH    544  // the shortest pulse sent to a servo
#define MAX_PULSE_WIDTH    2400 // the longest pulse sent to a servo
#define DEFAULT_PULSE_WIDTH 1500 // default pulse width when servo is attached
#define REFRESH_INTERVAL   20000 // mininum time to refresh servos in microseconds

#define SERVOS_PER_TIMER   12   // the maximum number of servos controlled by one
timer
#define MAX_SERVOS    (_Nbr_16timers * SERVOS_PER_TIMER)
#define INVALID_SERVO    255    // flag indicating an invalid servo index

typedef struct {
  uint8_t nbr      :6 ;          // a pin number from 0 to 63
  uint8_t isActive :1 ;          // true if this channel is enabled, pin not pulsed if false
} ServoPin_t ;
typedef struct {
  ServoPin_t Pin;
  unsigned int ticks;
} servo_t;
class Servo
{public:
  Servo();
  uint8_t attach(int pin);       // attach the given pin to the next free channel, sets pinMode, returns
channel number or 0 if failure
  uint8_t attach(int pin, int min, int max); // as above but also sets min and max values for writes.
  void detach();
  void write(int value);         // if value is < 200 its treated as an angle, otherwise as pulse width
in microseconds
  void writeMicroseconds(int value); // Write pulse width in microseconds
  int read();                    // returns current pulse width as an angle between 0 and 180 degrees

```

```

    int readMicroseconds();      // returns current pulse width in microseconds for this servo (was
read_us() in first release)
    bool attached();           // return true if this servo is attached, otherwise false
private:
    uint8_t servoIndex;        // index into the channel data for this servo
    int8_t min;                // minimum is this value times 4 added to MIN_PULSE_WIDTH
    int8_t max;                // maximum is this value times 4 added to MAX_PULSE_WIDTH
};
#endif

```

Librería para codificar en Arduino un Sensor Ultrasónico

```

// Ultrasonic - Library for HR-SC04 Ultrasonic Ranging Module.
// GitHub: https://github.com/JRodrigoTech/Ultrasonic-HC-SR04
// ##### LICENSE #####
// This code is licensed under Creative Commons Share alike
// and Attribution by J.Rodrigo ( http://www.jrodrigo.net ).

```

```

#ifndef Ultrasonic_h
#define Ultrasonic_h

#if ARDUINO >= 100
    #include "Arduino.h"
#else
    #include "WProgram.h"
#endif

#define CM 1
#define INC 0

class Ultrasonic
{
public:
    Ultrasonic(int TP, int EP);
    Ultrasonic(int TP, int EP, long TO);
    long Timing();
    long Ranging(int sys);

private:
    int Trig_pin;
    int Echo_pin;
    long Time_out;
    long duration,distacne_cm,distance_inc;
};

#endif

```

ANEXO 4: PROGRAMACIÓN FINAL

```
#include <Servo.h>
#include <Ultrasonic.h>
// #include <SoftwareSerial.h>

// Ultrasonic ultrasonic1(2,3); // (Trig
PIN,Echo PIN)
Ultrasonic ultrasonic2(49,51); // (Trig
PIN,Echo PIN)
// Ultrasonic ultrasonic3(6,7); // (Trig
PIN,Echo PIN)
int s1,s2,s3,ultra1,ultra2,ultra3;
int var;
int PosValue[] =
{40,50,60,80,100,110,120,125,130}; //valor
de la posicion del servo
int GP2[9];
int j,Minimo,MIN_Point,sendey;
char leer;

String stringOne,stringTwo,stringThree;
//SoftwareSerial mySerial(0, 1); // RX, TX
Servo servo1;
Servo servo2;
Servo servo3;
Servo servo4;
Servo servo5;
Servo servo6;
Servo servo7;
Servo servo8;
Servo servo9;
Servo servo10;
Servo servo11;
Servo servo12;
Servo servo13;
Servo servo14;
Servo servo15;
Servo servo16;
Servo servo17;
Servo servo18;
Servo servo19; //servo de busqueda
Servo servo20;

int i = 0;
int valor = 0;
int valor1 = 0;
int valor2 = 0;
int valor3 = 0;
//char var=0;
int dist=0;
int pos = 0;

void setup ()
{
  Serial.begin(9600);
  //Serial.println("ok");
  // izquierda derecha
  // 49 51 53 ( pata 1) 35 33 31(pata
4)
  // 43 45 47 (pata 2) 2 3 4 (pata 5)
  // 37 39 41 (pata 3) 8 9 10 (pata 6)

  stringOne = String("stringThree=");
  stringTwo = String("this string");
  stringThree = String();
  servo1.attach(22); //declaracion de pata 1
  servo2.attach(24);
  servo3.attach(26); //
  servo4.attach(2); // pata 2
  servo5.attach(3);
  servo6.attach(4);
  servo7.attach(5); // pata 3
  servo8.attach(6);
  servo9.attach(7);
  servo10.attach(28); // pata 4
  servo11.attach(30);
  servo12.attach(32);
  servo13.attach(46); // pata 5
  servo14.attach(48);
  servo15.attach(34);
  servo16.attach(40); //pata 6
  servo17.attach(42);
  servo18.attach(44);

  servo19.attach(36); //servo de busqueda
  servo20.attach(47); //servo de agarre
  paro();

  delay (50);
}

void paro()
{
  //pata 1
  servo1.write(60);
  servo2.write(120);
  servo3.write(120);

  // //pata 2
  servo4.write(160); //giro izq
```



```

servo5.write(130);//codo
servo6.write(170);// final

//pata 3
servo7.write(160);
servo8.write(130);
servo9.write(120);

//pata 4
servo10.write(110);// izq
servo11.write(35);
servo12.write(70);

// //pata 5
servo13.write(90);
servo14.write(65);
servo15.write(65);

//pata 6
servo16.write(0);
servo17.write(65);
servo18.write(65);
}

void adelante()
{
servo19.write(70);
//          i=0;
//          for (int i=0; i
<= 1; i++)
//          {
//          Serial.println(i); //presentacionde veces
del for
//
if(Serial.available())//detecciondeletra
tecleada
//
{
//
var=Serial.read();//almacenamiento del
caracter ascii en la variable var
//
}
//
if (var == 'p') // pregunta si la variable es
menor de 100
//
{
//
paro();
//
break;
//
}
}

// prueba pata2
//poaicion inicial

//pata 4
// ultra1 = ultrasonic1.Ranging(CM);
ultra2 = ultrasonic2.Ranging(CM);
// ultra3 = ultrasonic3.Ranging(CM);
delay(90);

stringThree = stringOne+ultra3+' '+ultra2+'
'+ultra1;

Serial.println(stringThree);

//pata 3
servo8.write(150);
servo7.write(150);
delay(60);
servo8.write(130);
delay(100);

servo7.write(149);
delay(60);
servo13.write(110);
delay(60);
servo1.write(41);
delay(60);
servo16.write(6);
delay(60);
servo4.write(155);
delay(60);
servo10.write(125);
delay(60);

//pata 5
servo14.write(30);
servo13.write(100);
delay(60);
servo14.write(60);
delay(100);

servo7.write(147);
delay(60);
servo13.write(101);
delay(60);
servo1.write(40);
delay(60);
servo16.write(4);
delay(60);
servo4.write(153);
delay(60);
servo10.write(127);
delay(60);

//          //pata 1
servo2.write(150);
servo1.write(50);
delay(60);

```

```

servo2.write(120);
delay(100);
servo7.write(145);
delay(60);
servo13.write(103);
delay(60);
servo1.write(49);
delay(60);
servo16.write(0);
delay(60);
servo4.write(151);
delay(60);
servo10.write(128);
delay(60);

//          //pata 6
servo17.write(35);
servo16.write(15);
delay(60);
servo17.write(65);
delay(100);

servo7.write(143);
delay(60);
servo13.write(105);
delay(60);
servo1.write(47);
delay(60);
servo16.write(13);
delay(60);
servo4.write(150);
delay(60);
servo10.write(129);
delay(60);
//

//          //pata 2
servo5.write(150);//codo
servo4.write(160);
delay(60);
servo5.write(130);//codo
delay(100);

//
servo7.write(141);
delay(60);
servo13.write(107);
delay(60);
servo1.write(45);
delay(60);
servo16.write(10);
delay(60);
servo4.write(159);
delay(60);
servo10.write(130);
delay(60);
//

servo2.write(120);
delay(100);
//pata 4
servo11.write(5);
servo10.write(120);
delay(60);
servo11.write(35);
delay(100);

servo7.write(140);
delay(60);
servo13.write(109);
delay(60);
servo1.write(43);
delay(60);
servo16.write(8);
delay(60);
servo4.write(157);
delay(60);
servo10.write(123);
delay(60);
//
*****
}
//}

void atras()
{
servo19.write(70);
//i=0;
// for (int i=0; i <= 5; i++)
//{
// Serial.println(i); //presentacionde
veces del for
// if(Serial.available())//detecciondeletra
tecleada
//{
// var=Serial.read();//almacenamiento
del caracter ascii en la variable var
//}
// if (var == 'p') // pregunta si la
variable es menor de 100
// {
// paro();
// break;
// }
// prueba pata2
//poaicion inicial
//pata 3
servo8.write(160);
servo7.write(110);
delay(60);
servo8.write(130);
delay(100);

servo7.write(115);

```

```

delay(60);
servo13.write(90);
delay(60);
servo1.write(70);
delay(60);
servo16.write(105);
delay(60);
servo4.write(135);
delay(60);
servo10.write(130);
delay(60);
//pata 5
servo14.write(30);
servo13.write(120);
delay(60);
servo14.write(65);
delay(100);
servo7.write(120);
delay(60);
servo13.write(115);
delay(60);
servo1.write(75);
delay(60);
servo16.write(100);
delay(60);
servo4.write(140);
delay(60);
servo10.write(125);
delay(60);
// //pata 1
servo2.write(150);
servo1.write(45);
delay(60);
servo2.write(120);
delay(100);
servo7.write(125);
delay(60);
servo13.write(110);
delay(60);
servo1.write(50);
delay(60);
servo16.write(95);
delay(60);
servo4.write(145);
delay(60);
servo10.write(120);
delay(60);
//
// //pata 6
servo17.write(25);
servo16.write(125);
delay(60);
servo17.write(65);
delay(100);

servo7.write(130);
delay(60);
servo13.write(105);
delay(60);
servo1.write(55);
delay(60);
servo16.write(120);
delay(60);
servo4.write(150);
delay(60);
servo10.write(115);
delay(60);
//
// //pata 2
servo5.write(160);//codo
servo4.write(120);
delay(60);
servo5.write(130);//codo
delay(100);
//
servo7.write(135);
delay(60);
servo13.write(100);
delay(60);
servo1.write(60);
delay(60);
servo16.write(115);
delay(60);
servo4.write(125);
delay(60);
servo10.write(110);
delay(60);
//
// //pata 4
servo11.write(10);
servo10.write(140);
delay(60);
servo11.write(35);
delay(100);
servo7.write(140);
delay(60);
servo13.write(95);
delay(60);
servo1.write(65);
delay(60);
servo16.write(110);
delay(60);
servo4.write(130);
delay(60);
servo10.write(135);
delay(60);
}

```

```

void agache()
{
  //*****
  //pata 1
  servo1.write(60);
  servo2.write(145);
  servo3.write(120);

  //          //pata 2
  servo4.write(70); //giro izq
  servo5.write(155); //codo
  servo6.write(170); // final

  //pata 3
  servo7.write(160);
  servo8.write(155);
  servo9.write(120);

  //pata 4
  servo10.write(110); // izq
  servo11.write(10);
  servo12.write(70);

  // //pata 5
  servo13.write(90);
  servo14.write(40);
  servo15.write(65);

  //pata 6
  servo16.write(0);
  servo17.write(40);
  servo18.write(65);
}

void izquierda()
{
  //i=0;
  // for (int i=0; i <= 5; i++)
  // {
  //   Serial.println(i); //presentacion de
  //   veces del for
  //   if(Serial.available()) //deteccion de letra
  //   tecleada
  //   {
  //     var=Serial.read(); //almacenamiento
  //     del caracter ascii en la variable var
  //   }
  //   if (var == 'p') // pregunta si la
  //   variable es menor de 100
  //   {
  //     paro();
  //     break;
  //   }
  // prueba pata2
  //poaicion inicial

  //*****
  //*****
  //pata 4
  servo11.write(10);
  servo10.write(100);
  delay(60);
  servo11.write(35);
  delay(100);

  servo10.write(103);
  delay(60);
  servo4.write(150);
  delay(60);
  servo16.write(13);
  delay(60);
  servo1.write(72);
  delay(60);
  servo13.write(87);
  delay(60);
  servo7.write(135);
  delay(60);

  //          //pata 2
  servo5.write(160); //codo
  servo4.write(140);
  delay(60);
  servo5.write(130); //codo
  delay(100);

  //
  servo10.write(105);
  delay(60);
  servo4.write(143);
  delay(60);
  servo16.write(10);
  delay(60);
  servo1.write(73);
  delay(60);
  servo13.write(88);
  delay(60);
  servo7.write(137);
  delay(60);
  //

  //          //pata 6
  servo17.write(25);
  servo16.write(25);
  delay(60);
  servo17.write(65);
  delay(100);

  servo10.write(107);
  delay(60);
  servo4.write(145);
  delay(60);
  servo16.write(23);
  delay(60);
}

```

```

servo1.write(75);
delay(60);
servo13.write(89);
delay(60);
servo7.write(138);
delay(60);
//          //pata 1
servo2.write(150);
servo1.write(65);
delay(60);
servo2.write(120);
delay(100);
servo10.write(108);
delay(60);
servo4.write(147);
delay(60);
servo16.write(20);
delay(60);
servo1.write(67);
delay(60);
servo13.write(90);
delay(60);
servo7.write(139);
delay(60);
//          //pata 5
servo14.write(30);
servo13.write(80);
delay(60);
servo14.write(65);
delay(100);
servo10.write(109);
delay(60);
servo4.write(148);
delay(60);
servo16.write(18);
delay(60);
servo1.write(69);
delay(60);
servo13.write(83);
delay(60);
servo7.write(140);
delay(60);
//          //pata 3
servo8.write(160);
servo7.write(130);
delay(60);
servo8.write(130);
delay(100);

servo10.write(110);
delay(60);
servo4.write(149);
delay(60);
servo16.write(16);
delay(60);

servo1.write(70);
delay(60);
servo13.write(85);
delay(60);
servo7.write(133);
delay(60);

}
void derecha()
{
//i=0;
// for (int i=0; i <= 5; i++)
//{
// Serial.println(i); //presentacionde
veces del for
// if(Serial.available())//detecciondeletra
tecleada
//{
// var=Serial.read();//almacenamiento
del caracter ascii en la variable var
//}
// if (var == 'p') // pregunta si la
variable es menor de 100
// {
//     paro();
//     break;
// }
// prueba pata2
//poaicion inicial

//pata 4

//*****
***
//pata 4
servo11.write(10);
servo10.write(120);
delay(60);
servo11.write(35);
delay(100);

servo10.write(119);
delay(60);
servo4.write(150);
delay(60);
servo16.write(3);
delay(60);
servo1.write(79);
delay(60);
servo13.write(93);
delay(60);
servo7.write(147);
delay(60);

```

```

//          //pata 2
servo5.write(160);//codo
servo4.write(160);
delay(60);
servo5.write(130);//codo
delay(100);

//
servo10.write(118);
delay(60);
servo4.write(159);
delay(60);
servo16.write(0);
delay(60);
servo1.write(77);
delay(60);
servo13.write(92);
delay(60);
servo7.write(145);
delay(60);

//          //pata 6
servo17.write(25);
servo16.write(15);
delay(60);
servo17.write(65);
delay(100);

servo10.write(117);
delay(60);
servo4.write(157);
delay(60);
servo16.write(13);
delay(60);
servo1.write(75);
delay(60);
servo13.write(91);
delay(60);
servo7.write(143);
delay(60);

//          //pata 1
servo2.write(150);
servo1.write(85);
delay(60);
servo2.write(120);
delay(100);

servo10.write(115);
delay(60);
servo4.write(155);
delay(60);
servo16.write(9);
delay(60);
servo1.write(83);
delay(60);
servo13.write(90);
delay(60);
servo7.write(142);

delay(60);
//          //pata 5
servo14.write(30);
servo13.write(100);
delay(60);
servo14.write(65);
delay(100);

servo10.write(113);
delay(60);
servo4.write(153);
delay(60);
servo16.write(7);
delay(60);
servo1.write(81);
delay(60);
servo13.write(97);
delay(60);
servo7.write(140);
delay(60);

//          //pata 3
servo8.write(160);
servo7.write(150);
delay(60);
servo8.write(130);
delay(100);

servo10.write(110);
delay(60);
servo4.write(151);
delay(60);
servo16.write(5);
delay(60);
servo1.write(80);
delay(60);
servo13.write(95);
delay(60);
servo7.write(149);
delay(60);

}
void Servo_Move(int val)
{
    int i;
    for(i=0;i<180;i++)
    {
        digitalWrite(7, HIGH);
        delayMicroseconds(val);
        digitalWrite(7, LOW);
        delay(20);
    }
}

int GET_Point()
{
    int i,Old=69,min_;

```

```

for(i=0;i<9;i++)
{
  if (GP2[i]<Old)
  {
    Old=GP2[i];
    min_=i;
  }
}
Minimo=Old;
return(min_);
}

void buscador()
{
  for(j=0;j<9;j++)
  {
    servo19.write(PosValue[j]);
    delay(500);
    ultra2 = ultrasonic2.Ranging(CM);
    stringThree = stringOne+ultra3+'
'+ultra2+' '+ultra1;
    Serial.println(stringThree);
    GP2[j]=ultra2;
  }
  MIN_Point = GET_Point();
  Serial.println(MIN_Point);
  delay(200);
}

void bajar()
{//pata 1
  servo1.write(70);
  servo2.write(150);
  servo3.write(120);

//          //pata 2
          servo4.write(70); //giro izq
          servo5.write(160); //codo
          servo6.write(170); // final

//pata 3
servo7.write(140);
servo8.write(160);
servo9.write(120);

//pata 4
servo10.write(110); // izq
servo11.write(20);
servo12.write(65);

// //pata 5
servo13.write(90);
servo14.write(35);
servo15.write(65);

//pata 6
servo16.write(100);
servo17.write(35);
servo18.write(50);
}

void garra()
{
  // for(pos = 0; pos <= 180; pos += 1) // goes
  // from 0 degrees to 180 degrees
  // {
  //           // in steps of 1
  // degree
  servo20.write(0);
  // servo20.write(180);
  // servo20.write(10); // tell servo
  // to go to position in variable 'pos'
  delay(15); // waits 15ms for
  // the servo to reach the position
  // }
  // for(pos = 180; pos >= 0; pos -= 1) // goes
  // from 180 degrees to 0 degrees
  // {
  // servo20.write(pos); // tell servo
  // to go to position in variable 'pos'
  // delay(15); // waits 15ms for
  // the servo to reach the position
  // }
}

void loop () { // Loop through motion
  tests

//paro();
// ultra1 = ultrasonic1.Ranging(CM);
// ultra2 = ultrasonic2.Ranging(CM);
// ultra3 = ultrasonic3.Ranging(CM);

  for(j=0;j<9;j++)
  {
    servo19.write(PosValue[j]);
    delay(300);
    ultra2 = ultrasonic2.Ranging(CM);
    stringThree = stringOne+ultra3+'
'+ultra2+' '+ultra1;
    Serial.println(stringThree);
    GP2[j]=ultra2;
  }
  MIN_Point = GET_Point();
  Serial.println(MIN_Point);
  delay(100);
  paro();

  while(sendey==1)
  {
    paro();
    leer=Serial.read();
    if (leer=='s'){

```

```

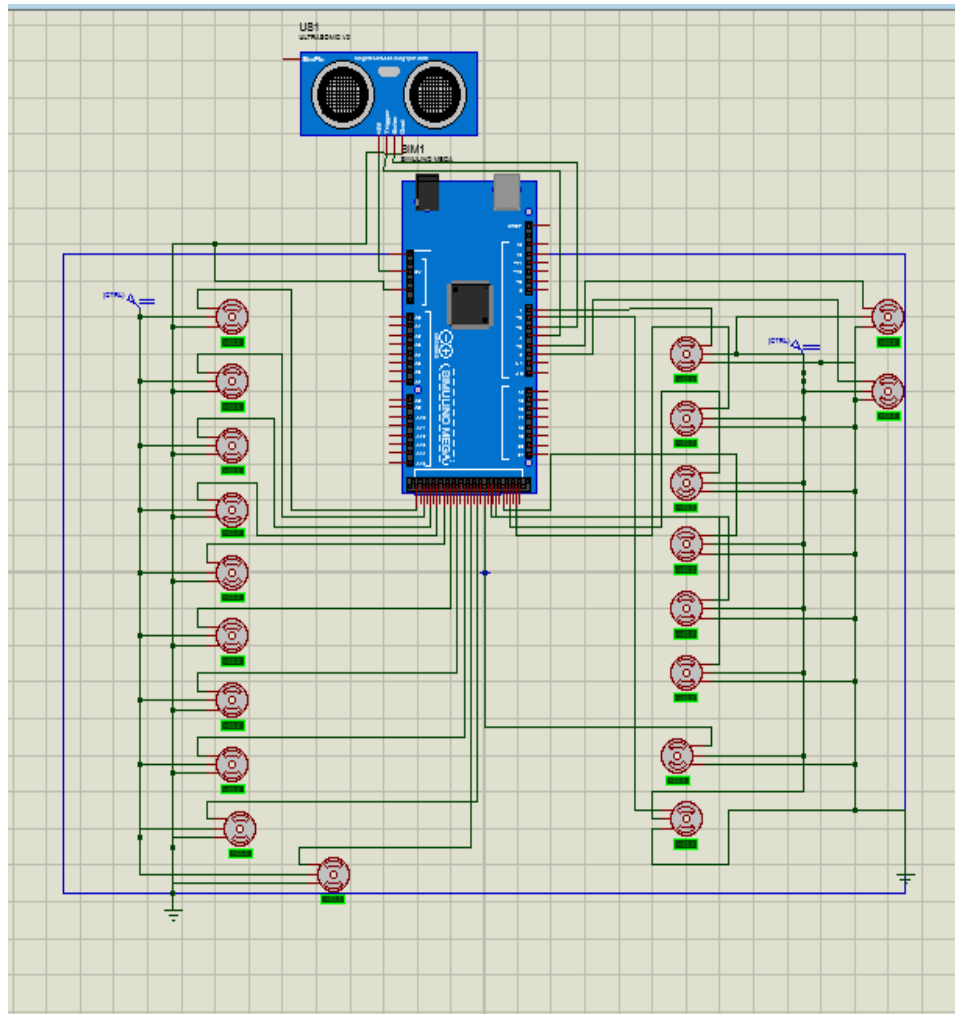
    sendey=0;
  }}
  if ((MIN_Point>-1)&&(MIN_Point<4))

  {derecha();
  sendey=0;
  Serial.println(MIN_Point);}
  else
  {
    if ((MIN_Point>4)&&(MIN_Point<9))
    { izquierda();
    sendey=0;
    Serial.println(MIN_Point);}
    else
    {
      while(MIN_Point==4)
      //
      while((MIN_Point>2)&&(MIN_Point<5))
      { adelante(); // atra es adelante
      Serial.println(MIN_Point);
      if (ultra2<6)
      {paro();// ha detectado avanzado
      hasta el objeto salir del bucle
      agache();
      garra();
      sendey=1;
      MIN_Point=0;
      }
      }
    }
  }
}
}

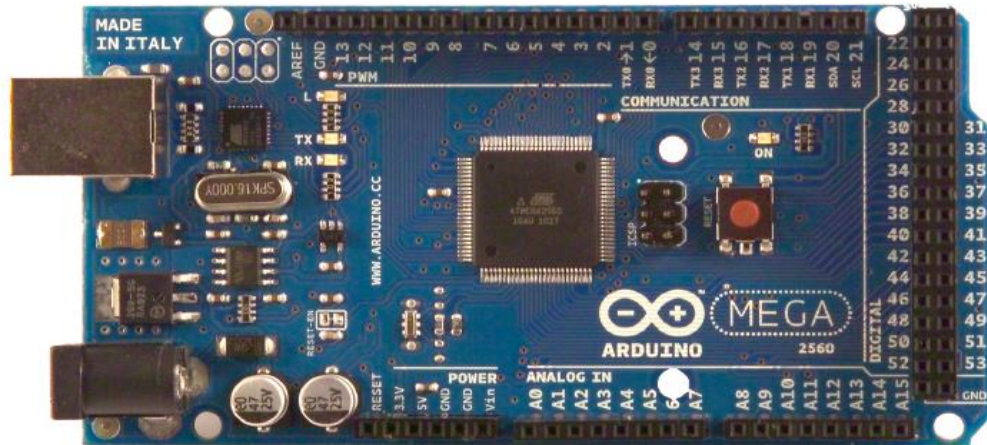
```


ANEXO 5: SIMULACIÓN DEL CIRCUITO

SIMULACIÓN TOTAL DEL CIRCUITO EN PROTEUS



ANEXO 6: CARACTERÍSTICAS TÉCNICAS DEL ARDUINO

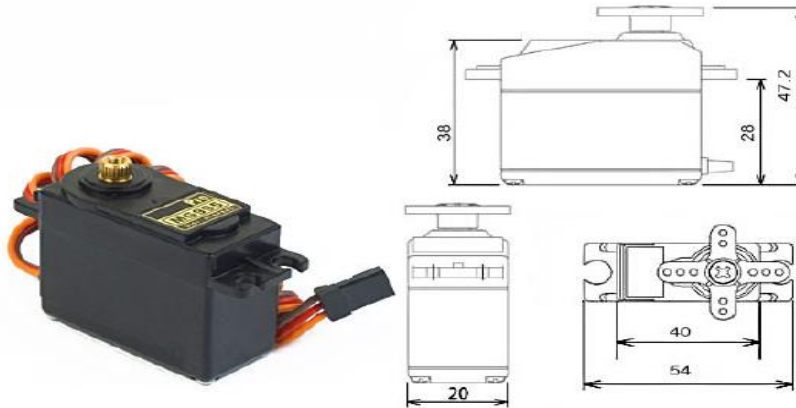


Características

Microcontroladores:	Atmega2560
Voltaje de funcionamiento:	5V
Voltaje de entrada (recomendado):	7-12V
Voltaje de entrada (límites):	6-20V
E / S digital pernos: (PWM)	54 (de los cuales 14 proporcionan salida)
Entrada analógica pernos:	16
Corriente DC por E / S Pin:	40 mA
Corriente DC de 3.3V Pin:	50 mA
Memoria Flash: por el gestor de arranque	256 KB de los cuales 8 KB utilizado
SRAM:	8 KB
EEPROM:	4 KB
Velocidad del reloj:	16 MHz

ANEXO 7: CARACTERÍSTICAS TÉCNICAS DEL SERVOMOTOR

TowerPro MG995 Alta velocidad y Torque, Sevo Digital



MG995 Servo Digital de Alta Velocidad y Torque

Voltaje recomendable de entrada: 3.5 - 8.4 Volt, nominal 5 voltios.

Datos Técnicos

Dimensiones: 1.57" x 0.79" x 1.44" (40 x 20 x 36.5mm)

Peso: 1.78oz (48g)

Velocidad de Operación

(4.8V sin carga): 0.17seg / 60 grados

Velocidad de Operación

(6V sin carga): 0.13seg / 60 grados

Torque de parada (4.8V): (13kg/cm)

Torque de Parada (6.0V): (15kg/cm)

Peso: 55.2g

Rango de Temperatura: -30 to +60 grados C

ANEXO 8: CARACTERÍSTICAS TÉCNICAS DEL SENSOR ULTRASÓNICO



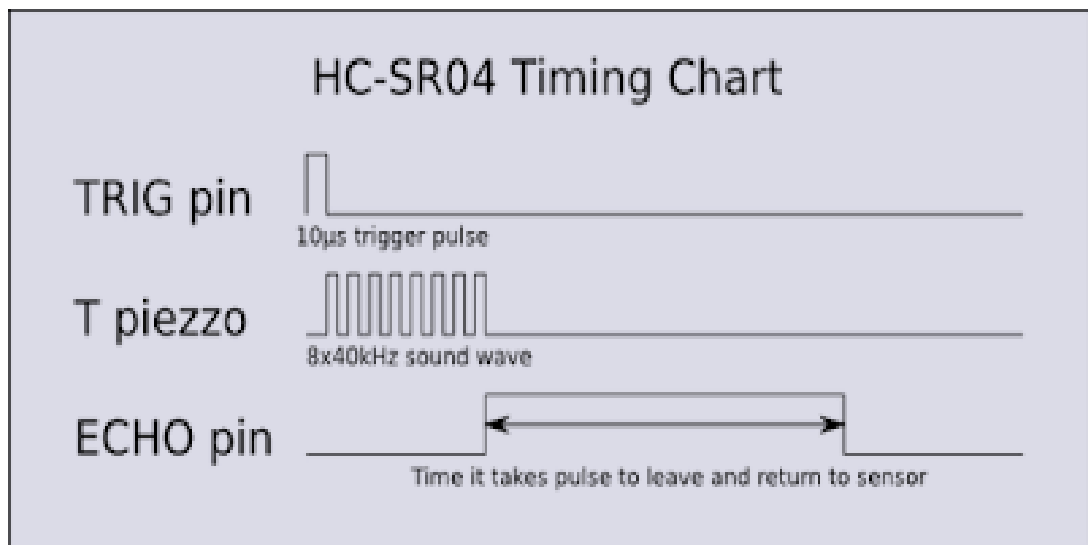
- El HC-SR04 es un sensor ultrasónico de bajo costo que no sólo puede detectar si un objeto se presenta, como un sensor PIR (Passive Infrared Sensor), sino que también puede sentir y transmitir la distancia al objeto.
- Tienen dos transductores, básicamente, un altavoz y un micrófono.
- Ofrece una excelente detección sin contacto con elevada precisión y lecturas estables en un formato fácil de usar.
- El funcionamiento no se ve afectado por la luz solar o el material negro como telémetros ópticos (aunque acústicamente materiales suaves como telas pueden ser difíciles de detectar).
- La velocidad del sonido en el aire (a una temperatura de 20 °C) es de 343 m/s. (por cada grado centígrado que sube la temperatura, la velocidad del sonido aumenta en 0,6 m/s).

Características técnicas:

- Los módulos incluyen transmisores ultrasónicos, el receptor y el circuito de control.

Número de pines:

- VCC: Alimentación +5V (4.5V min – 5.5V max)
- TRIG: Trigger entrada (input) del sensor (TTL)
- ECHO: Echo salida (output) del Sensor (TTL)
- GND
- Corriente de reposo: < 2mA
- Corriente de trabajo: 15mA
- Ángulo de medición: 30°
- Ángulo de medición efectivo: < 15°
- Detección de 2cm a 400cm o 1" a 13 pies.
- “Resolución” La precisión puede variar entre los 3mm o 0.3cm.
- Dimensiones: 45mm x 20mm x 15mm
- Frecuencia de trabajo: 40KHz



ANEXO 9: MANUAL DE USUARIO

En el manual de usuario se expondrá de manera detallada paso a paso como utilizar el robot para que pueda ser comprendido por la persona que deseen manipularlo. La división del manual está comprendido en:

Sistema de Fuerza y;

Sistema de Control

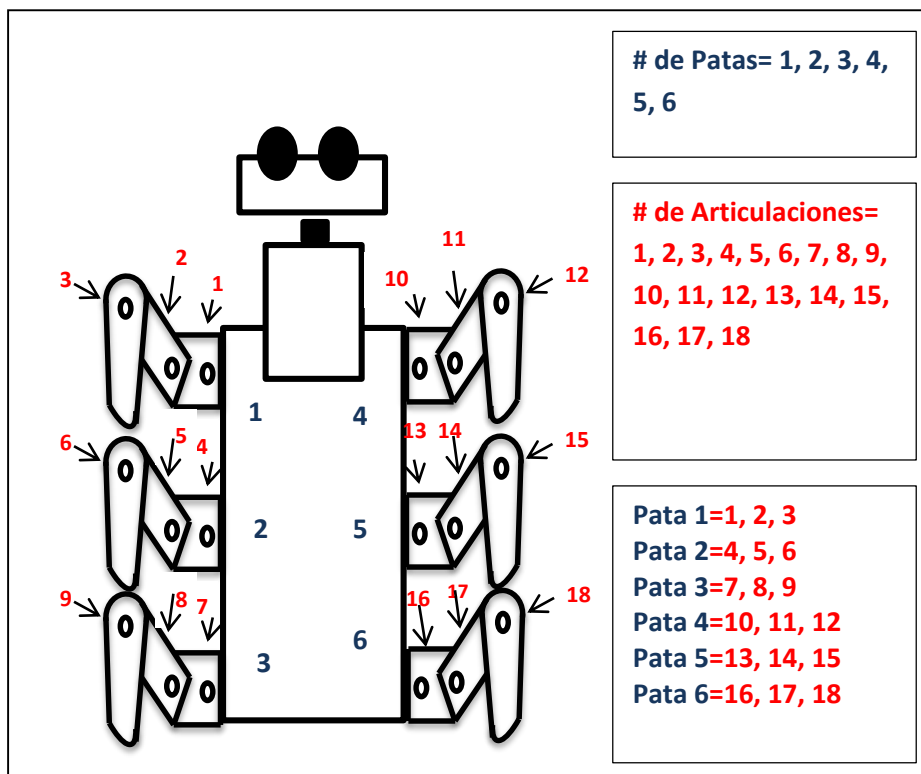
Manejo de la interfaz de Usuario

A continuación se explicara las instrucciones con respecto al Sistema de Fuerza.

SISTEMA DE FUERZA

La estructura mecánica

Para manipular el robot la persona se debe relacionar con el robot y con los valores asignados en la siguiente figura en la cual se identifica cada parte del cuerpo como son: número de patas, numero de articulaciones y que articulación corresponde a cada pata.



La interpretación correcta de estos valores servirá para posteriores modificaciones con la estructura en la codificación.

Así mismo se presenta el cuadro detallado de la obtención de los valores de referencia de los ángulos para las articulaciones de las extremidades, junto al número de pata y articulación correspondiente.

VALORES DE REFERENCIA					
# de extremidad	# de servomotor	# de articulaciones	Incremento	Referencia	Decremento
Pata 1	servomotor 1	Inicial 1	40°	60°	70°
	servomotor 2	Medio 1	130°	120°	-
	servomotor 3	Final 1	-	120°	-
Pata 2	servomotor 4	Inicial 2	150°	160°	170°
	servomotor 5	Medio 2	140°	130°	-
	servomotor 6	Final 2	-	170°	-
Pata 3	servomotor 7	Inicial 3	150°	160°	170°
	servomotor 8	Medio 3	140°	130°	-
	servomotor 9	Final 3	-	120°	-
Pata 4	servomotor 10	Inicial 4	120°	110°	100°
	servomotor 11	Medio 4	25°	35°	-
	servomotor 12	Final 4	-	70°	-
Pata 5	servomotor 13	Inicial 5	100°	90°	80°
	servomotor 14	Medio 5	55°	65°	-
	servomotor 15	Final 5	-	65°	-
Pata 6	servomotor 16	Inicial 6	-	0°	10°
	servomotor 17	Medio 6	55°	65°	-
	servomotor 18	Final 6	-	65°	-

Este cuadro especifica detalladamente el número de pata con sus respectivas articulaciones, nombre que se asignó en la programación y los grados calculados que serán utilizados como referencia al momento de la codificación de datos. Si en algún momento se desea cambiar un servomotor los ángulos cambiarían de referencia, ya que cada servomotor según su fabricante tiene el inicio en 0 grados en una dirección diferente.

Distribución de los pines de los elementos con la tarjeta arduino

La tabla de distribución de los pines servirá como una guía al momento de la programación en el software arduino, también como guía para la conexión física de los elementos. La distribución de los pines de conexión correspondientes al sensor ultrasónico y de los servomotores hacia los pines de la tarjeta mega arduino son:

#	Pin de Mega Arduino	Elemento	Articulaciones	# de Pata
1	Pin 22	servomotor	Inicial 1	Pata 1
2	Pin 24	servomotor	Medio 1	
3	Pin 26	servomotor	Final 1	
4	Pin 2	servomotor	Inicial 2	Pata 2
5	Pin 3	servomotor	Medio 2	
6	Pin 4	servomotor	Final 2	
7	Pin 5	servomotor	Inicial 3	Pata 3
8	Pin 6	servomotor	Medio 3	
9	Pin 7	servomotor	Final 3	
10	Pin 28	servomotor	Inicial 4	Pata 4
11	Pin 30	servomotor	Medio 4	
12	Pin 32	servomotor	Final 4	
13	Pin 46	servomotor	Inicial 5	Pata 5
14	Pin 48	servomotor	Medio 5	
15	Pin 34	servomotor	Final 5	
16	Pin 40	servomotor	Inicial 6	Pata 6
17	Pin 42	servomotor	Medio 6	
18	Pin 44	servomotor	Final 6	
19	Pin 36	servomotor	Búsqueda	
20	Pin 47	servomotor	Agarre	
21	Pin GND	De la tarjeta	Tierra Común	
22	Pin Vcc	De la tarjeta	Voltaje 5v	
23	Pin 49	Sensor U.	Trig	
24	Pin 51	Sensor U.	Echo	

En esta tabla indica los pines que fueron asignados para los 18 servomotores y los pines que se utilizaran para conectar el sensor ultrasónico.

Los elementos utilizados para la estructura son:

Cuerpo del robot.- Implementado para mantener más espacio entre las extremidades y así evitar roces.

Articulaciones Iniciales.- Sostiene al servomotor que está ubicado en uno de los orificios de la estructura del cuerpo, juntos dan giros de avance o retroceso.

Son 6 articulaciones iniciales.

Articulaciones Medias.- Sostiene al servomotor, esta articulación permite que la estructura se eleve o se baje, esta estructura se encuentra adjunta a la articulación inicial.

Son 6 articulaciones medias.

Articulaciones Finales.- Sostiene al servomotor, esta articulación permite que el robot eleve una de sus patas, esta estructura se encuentra adjunta a la articulación media.

Son 6 articulaciones finales.

Servomotores Tower Pro995 de 15 Kg.- El robot hexápodo ha sido diseñado para utilizar este tipo de actuadores, en total son 18 servomotores uno por cada articulación; la unión de tres articulaciones 1 inicial, 1 media y 1 final en conjunto con 3 servomotores forman una pata.

Con referente a la estructura se debe revisar lo siguiente:

- Revisar que los servomotores estén bien atornillados.
- Que los cables no estén ajustados o retenidos por algún objeto externo.
- Que el engranaje de la pinza esté dando giro y no esté obstruido.
- Chequear cada uno de los cables de los servomotores y el sensor ultrasónico estén bien conectados en su respectivo pin.
- Para empezar a movilizar el robot se debe tener conectada la fuente de alimentación de 5 voltios esta proveerá de energía a todos los elementos.
- Revisar bien la polaridad en cada prueba ya que una mala conexión podría hacer que los elementos se dañen.

- Antes de encender la fuente revisar que todos los elementos estén bien conectados para su correcto funcionamiento.

Una vez familiarizándose con la estructura mecánica se procede a continuar con las instrucciones con respecto al Sistema de Control.

SISTEMA DE CONTROL

Para lograr manipular el control del robot hexápodo se debe conocer los elementos que intervendrán en este sistema:

Para la parte de control los elementos utilizados son:

- **Tarjeta Mega arduino.-** Ubicada sobre la estructura del robot hexápodo, contiene un total de 54 pines de los cuales solo se utilizaran 22 pines. Placa encargada de la recepción y transmisión de datos para que el robot hexápodo realice las funciones determinadas.
- **Tarjeta adaptable.-** Ubicada sobre la placa mega arduino, dicha tarjeta es la encargada de la distribución de los pines y distribuir equitativamente la corriente y el voltaje hacia los elementos.

Para la parte de detección los elementos que intervienen son:

- **Sensor Ultrasónico HC-SR04.-** Es el encargado de indicar la distancia en la que se encuentra algún objeto a detectar.
- **Micro Servo MG90S de 2kg.-** Provocara el movimiento del sensor ultrasónico.

Para la recolección del objeto los elementos que intervienen son:

- **Pinza metálica.-** añadida con la finalidad de que realice el trabajo de recolectar el objeto-basura.

- **Servomotor HexTronikHX12K de 10kg.-** el movimiento mecánico de la pinza es realizado por el servomotor y un engranaje adaptable, en conjunto permitirán abrir y cerrar la pinza.

Ahora se describen algunas funciones necesarias que se deben conocer acerca de la detección, trazado de trayectoria y recolección del objeto.

Movimiento del servomotor para localización de objeto

En la parte frontal del robot se encontrará ubicado el sensor ultrasónico que está montado sobre un servomotor que servirá para ubicar la posición de un objeto, el sensor ultrasónico gira por acción del servomotor; el servomotor gira 180 grados y se determinan 9 posiciones específicas que no interfieran con la ubicación de las patas en movimiento, es decir la parte mecánica de detección que mueve al sensor no debe detectar el movimiento de las extremidades del robot.

Posiciones

Se plantea utilizar 9 posiciones que son las direcciones de giro del servomotor cada una con un ángulo específico para el movimiento del servomotor, las direcciones de posición van de 0 hasta 8, como se muestra en la tabla las posiciones y grados respectivamente.

Posiciones	Grados
Posición 0:	40 °
Posición 1:	50 °
Posición 2:	60 °
Posición 3:	80 °
Posición 4:	100 °
Posición 5:	110 °
Posición 6:	120 °
Posición 7:	125 °
Posición 8:	130 °

Posiciones y movimientos correspondientes

En el instante que el servomotor está girando por cada posición de ángulo, el sensor busca al objeto-basura de manera que registra en cada posición de giro un valor de distancia. Este valor de distancia se compara con las opciones de posiciones en la codificación para que de acuerdo a esa comparación tome la acción correspondiente girar o avanzar. De acuerdo a la posición donde se encontró el objeto el robot se desplazará considerando la siguiente tabla:

Posición	Movimiento
>4 y <9	Hacia la izquierda
> -1 y <4	Hacia la derecha
>3 y <5	Avanza

Posición, Movimiento y siguiente Paso

Cuando el robot no detecte nada dentro del rango de los 50 cm, seguirá girando a la izquierda o derecha hasta localizar el objeto dirigido hacia la parte central del robot, una vez que localice al objeto tendrá las siguientes opciones presentes en la tabla:

Posiciones	Movimiento	Siguiente Paso
0,1,2,3,5,6,7,8	Giro	Giro
4	Avanza	Avanza

Pinza Robótica

Pinza Robótica ubicada en la parte frontal de la estructura, esta pinza tiene la función de recoger el objeto mediante la apertura de sus pinzas, luego de recoger el objeto-basura la pinza se cierra. Mediante la interfaz de usuario se monitorea los valores de los sensores para visualizar donde se encuentra el objeto para que luego se proceda a cerrar la pinza.

Sensor que contiene al Servomotor

El sensor HC-SR04 es el encargado de indicar la distancia en la que se encuentre el objeto que será detectado a una distancia máxima de 50 cm, el sensor enviara una señal a la tarjeta arduino para que la estructura del robot escoja una ruta de trayectoria hacia el objeto. En siguiente tabla se presenta la ubicación del pin del Sensor Ultrasónico con el pin asignado a la placa arduino.

Sensor ultrasónico	Pin de la placa
Trigg	49
Eco	51
Vcc	Vcc
GND	Gnd

Sin las respectivas conexiones específicas tanto en el código y en la estructura del robot hexápodo no funcionara adecuadamente.

Distancia apropiada para recoger el objeto.

Cuando la estructura solo tenga la ruta de camino hacia adelante esta procede a obtener las siguientes opciones que se muestran en la tabla.

Avanza la estructura	Ultra2 > 8cm	Seguirá Avanzando
	Ultra2 <= 8 cm	Recoge el objeto

La recolección del objeto se realizará mediante una pinza robótica que consiste en un servomotor montado sobre un mecanismo de piñones que finaliza con una pinza o tenaza, la cual recibe las órdenes enviadas desde la tarjeta de control a través de la codificación, la pinza se encontrará abierta antes de empezar la trayectoria hacia el objeto, cuando el robot se encuentre a 8 cm del objeto basura, la pinza empezará a cerrarse en el diagrama representado con el numero digital del valor 0 y de esta manera agarrará el objeto, el procedimiento se detiene y para que pueda recoger otro objeto, se debe reiniciar el sistema.

INTERFAZ DE USUARIO

Como utilizar la interfaz de usuario

La codificación se la realiza a través del ordenador y el software de arduino mediante la conexión del cable USB, el lenguaje de programación es similar al lenguaje C++, para realizar modificaciones en la codificación son necesarios los conocimientos y el manejo de Arduino, una vez cargados los códigos se procede a compilar la codificación.

La parte de control se la realiza mediante la placa Mega Arduino 2560, está ubicada sobre la estructura del robot hexápodo. Se necesita tener en cuenta ciertos parámetros:

- Revisar que la tarjeta adaptable encaje correctamente al Mega Arduino ya que esta podría no estar haciendo contacto y el envío de señal puede ser fallido.
- Revisar que todos los pines estén bien ajustados.
- Verificar que la tarjeta Mega Arduino encienda cuando esté conectada a cable Usb de la Pc.

Programas necesarios

Para el manejo de la codificación y simulación del robot hexápodo deben ser instalados en el servidor los siguientes programas:

- Arduino 1.0.6
- Proteus 8 Profesional

La instalación de librerías es necesaria, porque la codificación puede ser correcta pero sin las librerías específicas no se compilara el código. Las librerías utilizadas en Arduino para poder compilar la codificación respecto al Sensor ultrasónico son los siguientes archivos:

KEYWORD.TXT

ULTRASONIC.CPP

ULTRASONIC.H

El link de descarga para la librería del sensor es ultrasónico es: <http://www.ardublog.com/library-for-arduino-ultrasonic-ranging-hc-sr04/>

Las librerías necesarias para trabajar en Proteus 8 Profesional para la simulación de la tarjeta controladora específicamente de la placa mega Arduino son los siguientes archivos:

ARDUINO.LIB

ARDUINO.IDX

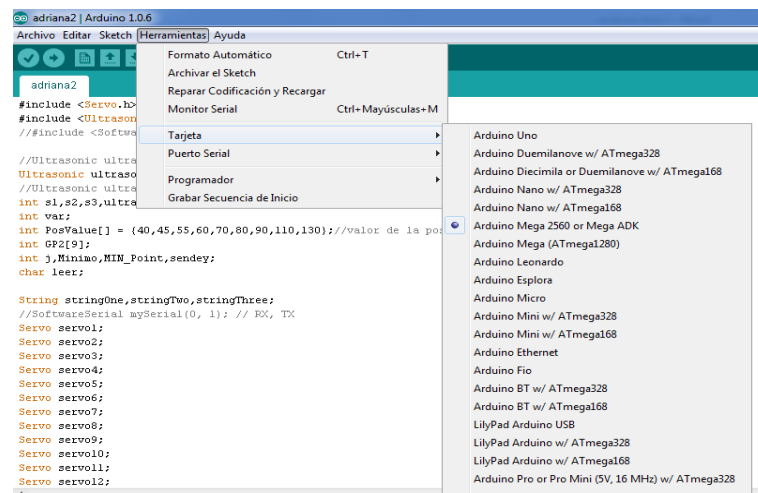
El link de descarga para la librería de la tarjeta de control en Proteus es: <http://ingerick.weebly.com/arduino/libreria-arduino-para-proteus-simulino-v3>

Luego de instalar los programas y las librerías se podrá cargar la codificación a la plataforma de arduino; y en el programa Proteus Professional la simulación del circuito.

Uso de Arduino

Seleccionar la Tarjeta

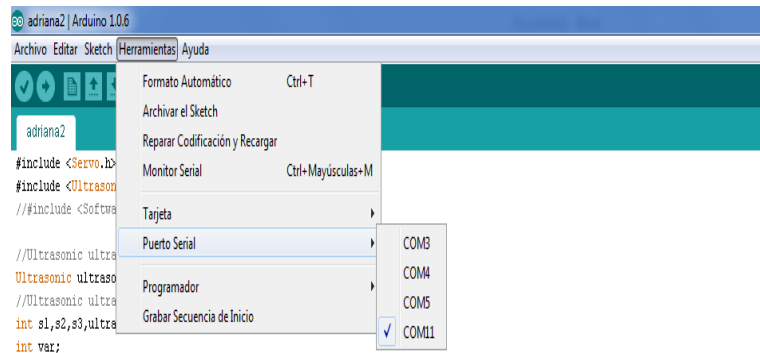
Se selecciona Herramientas - Tarjeta y aquí nos aparece una serie de placas controladoras pertenecientes a Arduino y se elige la tarjeta Mega Arduino 2560.



Seleccionar el puerto COM

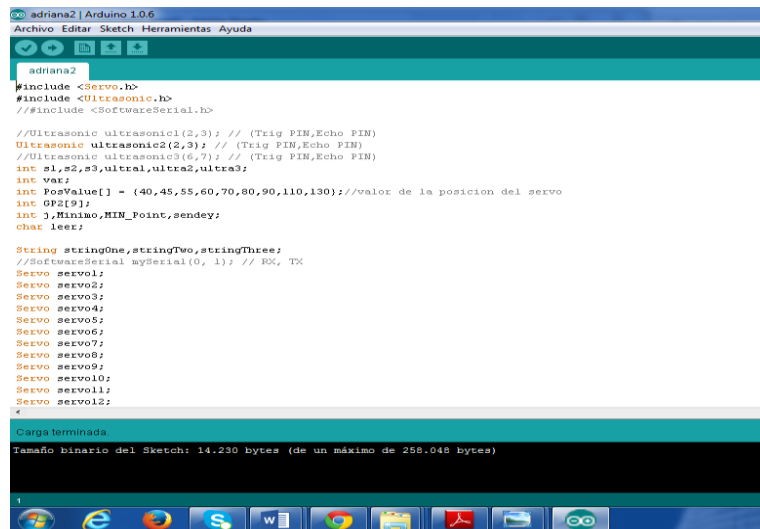
Para cargar la información a la tarjeta se requiere la comunicación del puerto serie.

Se selecciona el puerto COM que este asignado para la entrada USB convirtiéndolo en puerto serial. Herramientas – Puerto Serial y selecciona el puerto COM de destino en este caso el Puerto COM 11.



Abrir programación

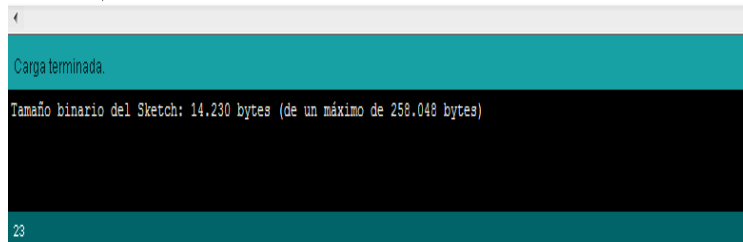
Nos dirigimos a Archivo- Abrir y elegimos la dirección donde está ubicado el archivo y se procede a cargar la programación que se ha diseñado.



Compilación de la programación

Después de tener la programación se compila el archivo dirigiéndose a la barra de herramientas – botón Cargar clic y espera a que se compile la información se demorar unos segundos. En la barra de estado en la parte inferior de la ventana se mostrará el estado de la compilación.

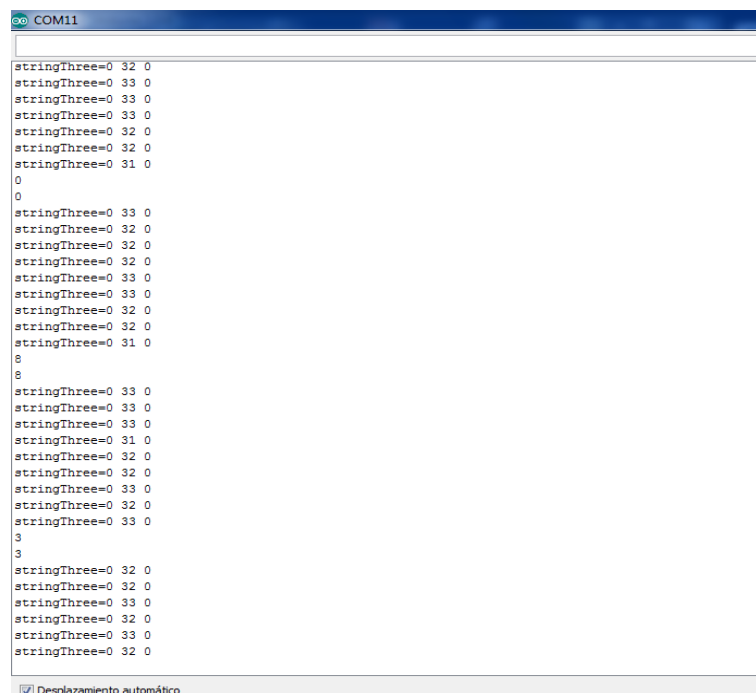

```
Servo servo8;  
Servo servo9;  
Servo servo10;  
Servo servo11;  
Servo servo12;
```



Monitor Serial

Luego de la compilación con éxito se provee de energía al robot, cuando tengamos estos dos parámetros listos procedemos a abrir el icono Monitor serial con este se abre una ventana. Mediante esta ventana se muestra los nueve estados de grados del servomotor cada uno representado por una distancia máxima a detectar de 50 cm ya que lo detecte avanza hasta cuando el objeto se encuentre a unos 8 cm de distancia este programa indicara a la estructura que se proceda a cerrar la pinza.

El programa analiza los nueve valores y escoge la distancia más cercana. Esta será la detección del objeto, con esta ventana veremos también cual fue el valor de la distancia que eligió para su trayectoria. A continuación se presenta la ventana del monitor serial.



Luego de leer estas instrucciones podrá utilizar el prototipo del robot hexápodo como elemento de contribución de la limpieza.

