

**DISEÑO Y CONSTRUCCION DE UN PROTOTIPO EXPERIMENTAL DE  
ROBOT ESCALADOR SEMIAUTONOMO PARA INSPECCION EN UNA  
PALMA**

**JAVIER ARMANDO GONZALEZ LIZCANO  
OSCAR YESID ZAPATA LÓPEZ**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERIAS FISICOMECÁNICAS  
ESCUELA DE INGENIERIA MECÁNICA  
BUCARAMANGA**

**2013**

**DISEÑO Y CONSTRUCCION DE UN PROTOTIPO EXPERIMENTAL DE  
ROBOT ESCALADOR SEMIAUTONOMO PARA INSPECCION EN UNA  
PALMA**

**JAVIER ARMANDO GONZALEZ LIZCANO  
OSCAR YESID ZAPATA LÓPEZ**

**Trabajo de grado para optar el título de  
INGENIERO MECÁNICO**

**Director  
CARLOS BORRÁS PINILLA  
Ingeniero Mecánico**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERIAS FISICOMECÁNICAS  
ESCUELA DE INGENIERIA MECÁNICA  
BUCARAMANGA**

**2013**

## DEDICATORIA

A Dios por darme salud y fortaleza para afrontar este proyecto y permitir sacarlo adelante.

A mis padres por ser un apoyo constante e incondicional a lo largo de toda mi carrera universitaria, sin ellos este proyecto no hubiese sido posible.

A mi hermano que quiero mucho y al que le deseo lo mejor en el comienzo de sus estudios en ingeniería mecánica.

A mi Lucero que siempre me apoyó y estuvo pendiente de la realización del proyecto, deseando siempre los más grandes éxitos.

Oscar Yesid.

## DEDICATORIA

*A mi Dios que me dio el hermoso regalo de la vida y la sabiduría para  
afrentarla con valor, amor y respeto.*

*A mis padres y hermanos que me han apoyado y guiado durante mi vida por el  
camino de la unión familiar*

*Javier Armando.*

## **AGRADECIMIENTOS**

A Gonzalo Calderón Caballero por su colaboración en el taller de la escuela de ingeniería mecánica con las actividades de torneado y taladrado de las piezas realizadas.

A Henry Cruz Mejía por su colaboración en las labores de doblado y corte en el taller de la escuela de ingeniería mecánica.

A nuestros compañeros de Ingeniería Mecánica por creer en nuestra capacidad y no perder la fe en nuestro proyecto.

Al profesor Carlos Borrás Pinilla por su guía a través del desarrollo del proyecto.

## TABLA DE CONTENIDO

	<b>Pág.</b>
<b>Contenido</b>	
INTRODUCCION .....	24
1. OBJETIVOS.....	26
1.1 OBJETIVO GENERAL .....	26
1.2 OBJETIVOS ESPECIFICOS.....	26
2. ROBOTS CAMINANTES .....	28
2.1 QUADRUPEDO O HEXAPODO .....	29
2.2 VENTAJAS DE LA LOCOMOCION DE LOS ROBOTS CAMINANTES.....	31
2.2.1 Movilidad.....	31
2.2.2 Evasión de obstáculos.....	31
2.3 ESTABILIDAD DE LOS ROBOTS CAMINANTES .....	33
2.3.1 Estabilidad estática en robot caminantes (plano horizontal).....	33
2.3.2 Criterio de estabilidad dinámica.....	36
2.4 MODOS DE CAMINAR (GAITS).....	39
2.4.1 Generación de gaits.....	40
2.4.1.1 Gaits continuos.....	41
2.4.1.2 Gaits discontinuos.....	44
2.4.1.3 Fases de los ciclos.....	45
2.5 MECANISMOS PARA ROBOTS CAMINANTES .....	46
2.5.1 Mecanismo de Hoekens.....	50
2.5.2 Mecanismo de Theo Jansen.....	51

2.5.3 Mecanismo de Hrones-Nelson.....	53
3. ROBOTS TREPADORES O ESCALADORES.....	55
3.1 SCREENBOT.....	55
3.2 STICKYBOT.....	56
3.3CLIMBING PARALLEL ROBOT (CPR).....	57
3.4GECKOBOT.....	59
3.5ROCR WALL CLIMBING ROBOT.....	60
4. ACTUADORES EN ROBOTICA MOVIL .....	63
4.1 SERVO MOTOR.....	63
4.1.1 Conexiones y funcionamiento.....	64
4.2 SERVO TRUCADO.....	66
4.3 MICROMOTORES.....	68
4.3.1 Aspectos a tener en cuenta con los integrados con arquitectura puente H.....	74
4.3.1.1 Transistores.....	74
4.4 MOTORES PASO A PASO.....	77
4.4.1 Bipolar.....	79
4.4.2 Unipolar.....	81
4.4.3 Secuencias de movimiento de los motores paso a paso.....	81
4.4.3.1 Secuencia Normal.....	81
4.4.3.2 Secuencia ondular.....	82
4.4.3.3 Secuencia tipo medio paso.....	83
5. SENSORES EN ROBOTICA MOVIL .....	85
5.1 PROPIEDADES A TENER EN CUENTA EN LA SELECCIÓN DE SENSORES .....	85

5.1.1 Velocidad de operación.....	85
5.1.2 Costo.....	86
5.1.3 Robustez.....	86
5.1.4 Peso y tamaño. ....	86
5.2SENSORES DE POSICIÓN.....	86
5.2.1 Potenciómetro.....	86
5.2.2 Codificador rotatorio o encoder.....	89
5.2.2.1 Codificador rotatorio absoluto. ....	89
5.2.2.2 Codificador rotatorio relativo. ....	91
5.2.3 IMU o unidad inercial de medición. ....	93
6. PREPROTOTIPO .....	95
6.1DISEÑO MECÁNICO DEL PRE PROTOTIPO DEL ROBOT ESCALADOR.....	95
6.1.1DISEÑO MECÁNICO DE LA PATA DEL ROBOT ESCALADOR.....	96
6.1.2 Diseño del pie o mecanismo de apoyo. ....	106
6.1.3 Diseño mecanismo de suspensión. ....	111
6.1.4 Diseño del sistema hombro.....	111
6.1.5 Sistemas de actuación.....	112
6.1.6 Análisis estático para el movimiento del pre prototipo en trípodes alternados. .....	115
6.1.7 Diseño de estructura de soporte o cuerpo del robot. ....	126
6.1.8 materiales y proceso de manufactura usado para el pre prototipo. ....	131
6.1.8.1 Compra de lámina de acrílico. ....	132
6.1.8.2 Generación de planos y corte de piezas. ....	132
6.1.8.3 Corte de piezas.....	133

6.1.8.4	Ensamble mecanismo pata.....	133
6.1.8.5	Ensamble mecanismos de suspensión.....	134
6.1.8.6	Ensamble estructura base.....	136
6.1.8.7	Ensamble sistema hombro.....	136
6.1.1.8	Ensamble trenes de engranajes y acople sistemas de actuación.....	137
6.1	DISEÑO DEL SISTEMA ELECTRÓNICO Y DE CONTROL.....	139
6.2.1	Captura de tecla presionada a través de librería Tkinter de Python.....	141
6.2.2	Interpretación de los datos por arduino.....	146
6.2.3	Control inasistido.....	151
6.2.4	Resistencia al flujo de corriente eléctrica de materiales.....	155
7.	EXPERIENCIAS DEL PREPROTOTIPO Y REDISEÑO DEL MISMO.....	157
7.1	PRUEBAS REALIZADAS AL PREPROTOTIPO.....	158
7.1.1	Pruebas en el tronco.....	158
7.2	DISEÑO CAD EN SOLIDWORKS.....	162
7.2.1	Selección de platinas para mecanismo de Hoekens.....	163
7.2.1.1	Calculo de la fuerza en el punto de agarre.....	164
8.	DETERMINACION DE EL TORQUE NECESARIO PARA ESCALAR Y SELECCIÓN DE MOTORES.....	170
8.1	ANALISIS ESTATICO PARA LA CONDICION CRÍTICA.....	171
8.2	CARACTERIZACION DE LOS MICROMOTORES.....	176
9.	DISEÑO DEL SISTEMA ELECTRONICO Y DE POTENCIA DEL PROTOTIPO FINAL.....	178
10.	PROCESO DE CONSTRUCCION Y ENSAMBLE DEL PROTOTIPO FINAL.....	181
10.1	PIEZAS ELABORADAS.....	181
10.1.1	Estructura.....	181

10.1.2 Barras del mecanismo de Hoekens. ....	181
10.1.3 Acoples para ejes de motores y potenciómetros. ....	181
10.1.4 Hombros. ....	181
10.1.5 Apoyos para los hombros (Potenciómetro). ....	182
10.1.6 Apoyos para los hombros (motor). ....	182
10.1.7 Base para micromotor del hombro. ....	182
10.1.8 Placa base potenciómetro. ....	182
10.2 ENSAMBLE DE COMPONENTES .....	183
11. CONTROL PROTOTIPO ROBOT ESCALADOR.....	187
11.1 SENSORES USADOS.....	192
11.2 PROGRAMACION EN ARDUINO DE LA SECUENCIA DE ESCALADA ....	193
11.2.1 Caracterización de actuadores y sensores. ....	193
11.2.2 Nomenclatura Usada. ....	194
11.2.3 Declaración de variables. ....	195
11.2.3 Multiplexor.....	197
11.2.4 Configuración inicial de los pines de la Arduino. ....	198
11.2.5 Lazo principal del código Arduino. ....	199
11.2.6 Función subir hombros.....	201
11.2.7 Función posicionarpatas. ....	203
11.2.8 Función asignarvector.....	212
11.2.9 Función desenergizar. ....	212
11.2.10 Función darfuerza. ....	213
11.2.11 Función leerpot. ....	214
12. RESULTADOS.....	218

12.1 PRUEBA DE TORQUES EN CONDICIÓN CRÍTICA.....	222
13. CONCLUSIONES .....	225
BIBLIOGRAFÍA.....	227
ANEXOS.....	229

## LISTA DE FIGURAS

	<b>Pág.</b>
Figura 1. Márgenes de estabilidad para vehículos caminantes .....	35
Figura 2. Diagrama geométrico para el cálculo del SESM.....	36
Figura 3. Fuerzas y momentos que actúan sobre el robot.....	38
Figura 4. Gaits que aseguran la estabilidad de un cuadrúpedo .....	41
Figura 5. Definiciones geométricas para gaits continuos.....	43
Figura 6. Diagrama de gaits para cuadrúpedos en creeping gait (wave) .....	44
Figura 7. Secuencia de avance discontinuo de dos fases para un cuadrúpedo ....	46
Figura 8. Partes de un mecanismo caminante.....	48
Figura 9. Fases de la trayectoria del pie .....	49
Figura 10. Mecanismo de Hoekens .....	51
Figura 11. Mecanismo de Theo Jansen.....	52
Figura 12. Trayectorias mecanismo Theo Jansen .....	52
Figura 13. Mecanismo Hrones-Nelson.....	53
Figura 14. SCREENBOT del CWRU biorobotics lab.....	56
Figura 15. StickyBot de la Universidad de Standford.....	57
Figura16. Parallel Robot .....	58
Figura 17. Geckobot .....	60
Figura 18. ROCR Wall climbing robot.....	61
Figura 19. Movimiento del ROCR .....	62
Figura 20. Partes de un servomotor.....	65
Figura 21. Pulsos de posicionamiento del servo.....	66
Figura 22. Trucado de un Servomotor. ....	68
Figura 23. Micromotor relación 298:1 .....	69
Figura 24. Arquitectura del puente H. ....	70
Figura 25. Integrado L293D NE package.....	71
Figura 26. Conexión típica para control de dos motores.....	72
Figura 27. Generador de pulsos basado en el 555 IC .....	73

Figura 28. Diagramas de NPN y PNP.....	75
Figura 29. Rotor y estator de un motor Paso a Paso. ....	78
Figura 30. Esquema de conexión usando un integrado L293.....	80
Figura 31. Secuencia de control motor bipolar.....	80
Figura 32. Conexión motor paso a paso unipolar usando ULN2803.....	81
Figura 33. Características de un potenciómetro. ....	88
Figura 34. Diferentes configuraciones de potenciómetros. ....	89
Figura 35. Encoder Absoluto .....	90
Figura 36. Codificador rotativo de tres contactos.....	91
Figura 37. Esquema de conjunto encoder incremental.....	92
Figura 38. Diseño CAD del preprototipo Solidworks. ....	95
Figura 39. Mecanismo Barras Hoekens .....	97
Figura 40. Angulo barra balancín con respecto a la horizontal en función del ángulo de la barra motora. ....	101
Figura 41. Angulo barra biela con respecto a la horizontal en función del ángulo de la barra motora.....	102
Figura 42. Nomenclatura usada mecanismo Hoekens. ....	103
Figura43. Posición x e y del punto de interés P, cada punto indica una posición de la manivela durante un ciclo. ....	104
Figura44. Posición en y del punto de interés P vs Angulo de la manivela. ....	104
Figura45. Posición en x del punto de interés P vs Angulo de la manivela. ....	105
Figura 46. Representación gráfica de los requerimientos del sistema pie. ....	107
Figura 47. Representación posición de barras mecanismo Hoekens durante el avance del sistema. ....	107
Figura 48. Sistema pie rotula propuesto. ....	108
Figura 49. Diseño sistema pie propuesto tipo dactyl.....	109
Figura 50. Diseño sistema pie propuesto tipo dactyl en etapa de compresión de resorte.....	110
Figura 51. Fuerzas generadas en el sistema de agarre tipo dactyl en etapa final. ....	110

Figura 52. Configuración sistema hombro y posición mecanismo Hoekens sobre este.....	112
Figura 53. Servomotores tentativos para usar en el pre prototipo. ....	113
Tabla 8. Tabla de pesos del preprototipo.....	114
Figura 54. Diagrama de cuerpo libre conjunto total del robot con trípode 1,4,5 apoyados. ....	116
Figura 55. Vector de momento aplicado por el actuador de la pierna sobre el módulo 1 .....	119
Figura 56.Reacciones en la barra manivela.....	121
Figura 57. Reacciones en la barra biela. ....	123
Figura 58. Rango de movimiento del hombro. ....	126
Figura 59. Rango de interés del punto A para diferentes inclinaciones de Hombro. ....	127
Figura 60. Ubicación deseada de sistema hombro-servomotor para una dupla de patas. ....	128
Figura 61. Distancias entre patas .....	129
Figura 62. Acople de barras usando arandelas entre estas.....	134
Figura 63. Cartón insertado entre acople de barras.....	134
Figura 64. Configuración barras en mecanismo de suspensión. ....	135
Figura 65. Configuración acople sistema suspensión y mecanismo Hoekens.....	135
Figura 66. Vista ensamble modulo hombro. ....	137
Figura 67. Acople engranaje final-eje hombro .....	139
Figura 68. Representación sistema control asistido pre prototipo.....	140
Figura 69. Inicialización tkinter y enlace de eventos a funciones:.....	142
Figura 70. Función llamada al presionar una tecla. ....	144
Figura 71. Clase servo comunicación serial Arduino - Python.....	145
Figura 72. Procesamiento datos entrada serial.....	147
Figura 73. Acciones tomadas al tener procesados los datos del puerto serial ....	148
Figura 74. Nomenclatura conexión servos.....	149
Figura 75. Diagrama de bloques de secuencia de escalada .....	153

Figura 76. Nomenclatura resistividad material .....	156
Figura 77. Preprototipo finalizado .....	157
Figura 78. Preprototipo configuración tripode .....	159
Figura 79. Diseño prototipo final .....	163
Figura 80. Distancia para cálculo de fuerza critica.....	164
Figura 81. Reacciones en barra de avance .....	165
Figura 82. Propiedades físicas del diseño en solidworks.....	171
Figura 83. Esquema de la estática del trípode de avance .....	173
Figura 84. Ángulos y aislamiento de la barra biela .....	174
Figura 85. Grafica de torque requerido para equilibrio vs ángulo de la manivela del mecanismo de hoekens .....	175
Figura 86. Pruebas de caracterización de micromotor.....	177
Figura 87. Caminos de conexión de las PCBs.....	178
Figura 88. Ensamble de PCBs.....	180
Figura 89. Piezas elaboradas .....	183
Figura 90. Estructura con Apoyos de hombro.....	184
Figura 91. Ensamblaje de módulo .....	185
Figura 92. Prototipo finalizado. ....	186
Figura 93. Diagrama flujo tarea de avance en trípodes. ....	187
Figura 94. Ciclos for para emular dos acciones en paralelo. ....	191
Figura 95. Nomenclatura robot escalador.....	195
Figura 96. Grafica de corriente y Ángulo de manivela durante un tiempo de prueba .....	219
Figura 97. Comportamiento en secuencia de avance en vertical. ....	220
Figura 98. Zoom en el avance .....	221
Figura 99. Comportamiento de la Corriente durante ensayo de criticidad en varios ciclos.....	223
Figura 100. Aproximación línea de corriente durante un ciclo de trabajo. ....	224

## LISTA DE TABLAS

	<b>Pág.</b>
Tabla1. Diferencias entre robots caminantes y robots accionados por ruedas .....	32
Tabla 2. Secuencia Normal motor paso a paso unipolar .....	82
Tabla 3. Secuencias Tipo ondular.....	83
Tabla 4. Secuencia tipo medio paso .....	84
Tabla 5. Tabla comparativa de los sensores de posición angular. ....	93
Tabla 6. Diferentes relaciones de barras en mecanismo Hoekens, para optimización de velocidad y de linealidad del punto de movimiento de interés. ....	98
Tabla 7. Longitudes características mecanismo Hoekens elegido.....	100
Tabla 9. Propiedades del polímero de metil metacrilato: .....	131
Tabla 10. Valores ejemplo propiedad usbport librería servo.....	146
Tabla 11. Pines digitales arduino, teclas y funciones asignados a cada servomotor .....	150
Tabla 12. Modificaciones del preprototipo. ....	160
Tabla 13. Caracterización de micromotor 298:1 .....	176

## LISTA DE ANEXOS

ANEXO A. PLANOS CLIMBOT.....	230
ANEXO B. CIRCUITO IMPRESO .....	244

## RESUMEN

**TITULO: DISEÑO Y CONSTRUCCION DE UN PROTOTIPO EXPERIMENTAL DE ROBOT ESCALADOR SEMIAUTONOMO PARA INSPECCION EN UNA PALMA\***

**AUTORES: JAVIER ARMANDO GONZALEZ LIZCANO, OSCAR YESID ZAPATA LOPEZ \*\***

**PALABRAS CLAVES:** Robot, escalador, arboles, hexápodo, semiautónomo

### DESCRIPCIÓN

Este proyecto surge de la necesidad de generar conocimiento en el campo de la robótica móvil, más específicamente en robots con mecanismos caminantes escaladores, cumpliendo de esta manera con una de las misiones de la Universidad Industrial de Santander en el aporte de material científico a la sociedad.

Para este propósito se investigó el estado del arte en esta área del conocimiento y se diseñó y construyó un prototipo de robot escalador de árboles capaz de realizar secuencias de avance de manera semiautónoma. El diseño consiste en un hexápodo inspirado en algunos insectos, cuyas patas poseen dos grados de libertad y son accionadas por actuadores rotativos.

El proceso de construcción del prototipo se lleva a cabo en su totalidad en la Universidad Industrial de Santander mediante el uso de las máquinas- herramientas existentes en el taller de mecánica de la escuela de ingeniería mecánica. Para permitir una buena relación potencia- peso que permita al prototipo su ascenso vertical en la superficie de prueba, se diseñó el prototipo con materiales livianos como el aluminio y el acrílico ya que son materiales de fácil consecución en el área de Santander.

El circuito de control fue diseñado de manera que este se acople al prototipo permitiendo una mayor robustez y confiabilidad mientras el robot se encuentra en operación.

---

\* Trabajo de grado

\*\* Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería Mecánica, Ing. Carlos Borrás Pinilla

## ABSTRACT

**TITLE: DESIGN AND CONSTRUCTION OF AN EXPERIMENTAL PROTOTYPE OF A SEMIAUTONOMOUS CLIMBING ROBOT FOR INSPECTING A PALM TREE\***

**AUTHORS:** JAVIER ARMANDO GONZALEZ LIZCANO, OSCAR YESID ZAPATA LOPEZ, \*\*

**KEYWORDS:** Robot, climbing, trees, hexapod, semiautonomous

### DESCRIPTION

This project arises from the need of generating knowledge in the mobile robotic field, more specifically in walking and climbing robots, fulfilling one of the missions of the Universidad Industrial de Santander providing scientific material to the society.

For this purpose the state of the art was investigated and an experimental prototype of a semiautonomous climbing robot capable to do semiautonomous advance sequences was designed and built. The design consists in an insect-inspired hexapod, which have two degrees of freedom per leg and actuated by rotary actuators.

The whole construction process was done at the Universidad Industrial de Santander by using the machine tools of the mechanical engineering school. In order to have a good weight – power rate so the prototype can vertically climb the test surface, it was designed with light materials such as aluminum and acrylic which are easily founded in Santander

The control circuit was designed to be joined to the prototype; it allows a big sturdiness and reliability while the robot is working.

---

\* Undergraduated thesis

\*\* Physical Mechanical Engineering Faculty, Mechanical engineering school, Eng. Carlos Borrás Pinilla

## INTRODUCCION

La escuela de ingeniería mecánica de la universidad industrial de Santander, se ha caracterizado por formar profesionales con alto grado de competencias matemáticas, analíticas, técnicas y más, siendo la generación de conocimiento una de las tareas primordiales por parte de los grupos de investigación que la conforman.

Los sistemas robóticos se han consolidado como una solución fundamental a las problemáticas de la industria, atendiendo en trabajos tediosos, complicados, con requerimientos de precisión y muchas veces difíciles de realizar por operadores humanos.

Debido a esto, la investigación de sistemas robóticos es actualmente un campo de estudio importante que vale la pena dar más jerarquía a nivel regional, logrando con esto producir sistemas eficaces que satisfagan los requerimientos de la industria en pro del desarrollo tecnológico.

Alrededor de esta temática, y ya que se presenta la oportunidad de aplicación de un sistema robótico a una problemática en la industria colombiana, más exactamente en la industria del cultivo de palma de cera, es válido plantearse ciertas inquietudes acerca de los principales sistemas robóticos actuales.

Una alternativa aceptada para solucionar el problema de la industria palmera y al mismo tiempo abordar esta temática investigativa la constituye plantear un proyecto de investigación acerca de robots exploradores, analizando sus modos de caminar, sistemas de control, el estudio de su estabilidad, el procesamiento y muestreo de información, entre otras acciones que garantizan su función.

Con el aumento en la investigación en robótica en Colombia, es posible desarrollar sistemas que automaticen diferentes tareas que en la actualidad son realizadas por operadores humanos y que debido a las limitaciones normales, impiden un mayor aprovechamiento de los recursos y por consecuencia limitan el desarrollo económico del país.

El diseño y construcción de un prototipo de robot escalador, libera en gran medida de uno de los trabajos difíciles en el proceso de recolección de frutos de palma africana tal como es la tarea de inspección, visualización de plagas, entre otras tareas que son tediosas cuando las plantaciones han alcanzado una elevada altura.

La liberación de trabajo tedioso para el ser humano es uno de los objetivos de la tecnología, es por esto que se busca solucionar el problema.

En el presente trabajo de grado, denominado “DISEÑO Y CONSTRUCCION DE UN PROTOTIPO EXPERIMENTAL DE ROBOT ESCALADOR SEMIAUTONOMO PARA INSPECCION EN UNA PALMA” se recopila la información obtenida del proceso de diseño y construcción del prototipo y los análisis realizados para su desarrollo.

## 1. OBJETIVOS

### 1.1 OBJETIVO GENERAL

- Contribuir con la Universidad Industrial de Santander en el desarrollo de material científico y de investigación en el área de la robótica por medio del diseño, propuesta y construcción de un prototipo experimental capaz de escalar superficies de palmas de manera semiautónoma, estable y eficiente.

### 1.2 OBJETIVOS ESPECIFICOS

- Investigar el estado del arte actual en cuanto a alternativas de robots escaladores con el fin de generar material científico regional.

- Proponer, diseñar y construir el sistema mecánico para un prototipo de robot experimental capaz de escalar un árbol o palma con los siguientes requerimientos:

- Altura mínima a escalar: 2 metros
- Envergadura máxima del sistema: 35 cm
- Largo máximo: 60 cm

- Proponer, diseñar y construir un sistema de sujeción para un robot experimental que permita sostenerlo en la superficie de una palma al escalar.

-Diseñar e implementar un sistema de locomoción que garantice estabilidad y fluidez del movimiento de escalada del robot experimental con un rango de velocidad de avance promedio entre  $0,3 \frac{m}{min}$  y  $1,2 \frac{m}{min}$ .

- Diseñar e implementar un sistema de control que asegure el buen funcionamiento y autonomía del robot escalador.

- Valorar los resultados obtenidos de pruebas de velocidad y estabilidad del prototipo de robot en funcionamiento variando parámetros de la superficie a escalar como rugosidad e inclinación a través de una plataforma de pruebas previamente diseñada.

## 2. ROBOTS CAMINANTES

La inspección y reconocimiento de terrenos es una práctica que ha venido teniendo un auge en el campo de la robótica, la aplicación de sistemas autónomos en labores de reconocimiento ha centrado sus esfuerzos en robots móviles a base de ruedas (llantas) que permiten un ágil avance en superficies principalmente uniformes.

Los robots caminantes presentan un tipo de locomoción diferente a los vehículos convencionales, ya que presentan diseños inspirados en los modos de desplazamiento de algunos animales como insectos, mamíferos entre otros.

Este tipo de locomoción permite realizar tareas un poco más complicadas, debido a su mayor capacidad de adaptación a diversos terrenos.

La creación de robots caminantes implica tareas como:

- **Diseño mecánico:** implica el diseño estructural, los materiales usados así como también los grados de libertad que tendrá el robot y los mecanismos que permitirán la locomoción del mismo de una manera característica.
- **Diseño electrónico:** implica la utilización de los instrumentos de medición y captación de señales necesarios para que el robot cumpla adecuadamente la tarea propuesta, además incluye el desarrollo de los algoritmos que definirán la manera de caminar del robot, la estabilidad, la velocidad, la distribución de fuerza y esfuerzos en los componentes del robot.

Como es de esperarse, cada aplicación determina las características mecánicas (diseño mecánico) y los algoritmos de control (diseño electrónico) empleados para el cumplimiento de la tarea.

Las aplicaciones pueden ser diversas y estarán determinadas principalmente por el tipo de terreno de operación. Así podemos encontrar aplicaciones para terrenos

rocosos con zonas prohibidas, terrenos practicables (que ofrecen puntos de apoyo definidos.), terrenos llanos, superficies inclinadas entre otros.

Al hablar de robots caminantes, necesariamente se debe hacer relación a las patas (elementos mecánicos que proporcionan la sustentación y permiten el desplazamiento del robot), las patas de un robot pueden tener diversos diseños dependiendo de la aplicación, su número puede variar dependiendo del grado de estabilidad que se desee y de la forma de locomoción en la que se inspiró.

Los tres tipos de robots caminantes más comúnmente usados son los bípedos, cuadrúpedos y hexápodos; en este libro se tomara en consideración los dos últimos debido a que el desarrollo de este libro va encaminado al diseño y construcción de un prototipo experimental de robot escalador, por lo que se hace más viable el uso de un robot de cuatro o seis patas.

## **2.1 QUADRUPEDO O HEXAPODO**

Cuando un robot caminante está siendo desarrollado, el primer aspecto que se debe definir es el número de patas. El número de patas es el resultado de sopesar los diferentes aspectos que el robot tendrá.

Como se mencionó anteriormente, estos aspectos están relacionados con la estabilidad, confiabilidad, velocidad, peso y precio.

Por ejemplo los hexápodos poseen mayor estabilidad estática que los cuadrúpedos debido a que pueden soportar el peso del cuerpo en cinco de sus patas mientras que la otra restante se moviliza para avanzar, por otro lado los cuadrúpedos solo tienen tres de sus patas soportando el peso mientras la otra se está movilizand.

En términos de velocidad de los robots caminantes, los estudios del profesor Kenneth Waldron y sus colegas han demostrado que la velocidad depende de la carrera de la pata,  $R$ , el tiempo de regreso,  $\tau$  y un factor de servicio,  $\beta$ , que depende directamente del número de patas.

$$V = \frac{R}{\tau} \left( \frac{1 - \beta}{\beta} \right)$$

El mínimo factor de servicio de un robot con  $n$  patas está dado por  $\beta_n = \frac{3}{n}$  lo cual reemplazando para  $n=4$  y  $6$  patas se obtiene la velocidad en términos de la carrera y el tiempo de regreso. Si comparamos la velocidad entre un hexápodo y un cuadrúpedo para un mismo mecanismo de locomoción (carrera y tiempo de regreso) se obtiene los siguientes resultados.

$$V_4 = 0.333 (R/\tau) \text{ (velocidad para un cuadrúpedo).}$$

$$V_6 = R/\tau \text{ (velocidad para un hexápodo)}$$

Con esto se puede observar claramente que la velocidad es mayor en hexápodos que en cuadrúpedos, y a medida que el número de patas aumenta, también lo hace su velocidad.

Algunos investigadores a lo largo de la historia, han señalado que un hexápodo puede continuar caminando después de presentar fallo en dos patas (una de cada lado), lo que indica que los hexápodos son estáticamente redundantes en cuanto a estabilidad se refiere.

Esta redundancia implica desventajas en cuanto a confiabilidad debido al mayor número de piezas presentes y a la mayor complejidad de los circuitos de control; además debido a que el peso del cuerpo es en la mayoría de aplicaciones

independiente al número de patas, un mayor número de patas supondría un peso total mayor.

## **2.2 VENTAJAS DE LA LOCOMOCION DE LOS ROBOTS CAMINANTES**

A parte de querer imitar la naturaleza, los investigadores ven en los robots caminantes muchas otras ventajas sobre los vehículos convencionales basados en ruedas y pistas, algunas de estas ventajas son las siguientes:

**2.2.1 Movilidad.** Los robots caminantes pueden cambiar de dirección de una manera más fácil por medio del cambio de los puntos de apoyo de sus patas, mientras que un vehículo con ruedas tendrá que hacer maniobras y giros para re direccionar su cuerpo.

En cuanto a grados de libertad, los robots caminantes poseen mayor cantidad de grados de libertad, ya que pueden direccionar su cuerpo cambiando la extensión de sus patas.

**2.2.2 Evasión de obstáculos.** La evasión de obstáculos se mejora notablemente en los robots articulados ya que estos pueden ubicar sus patas sobre los obstáculos utilizándolos como superficie de apoyo lo que brinda una mayor flexibilidad en los terrenos, los robots con ruedas convencionales, en cambio, pueden evadir obstáculos rodeándolos o pasando sobre ellos pero cuando estos no superan una altura mayor que el radio de la rueda.

Otras ventajas son mostradas en la siguiente tabla por medio de dibujos, en esta tabla se puede observar que los robots caminantes:

- Se adaptan mejor a terrenos discontinuos.

- Presentan mayor eficiencia sobre terrenos blandos que los robots a base de ruedas.
- Son más amigables con el medio ambiente debido a que no generan daños significativos sobre el terreno.
- Poseen una velocidad promedio en terrenos no uniformes.

Las diferencias entre los robots caminantes y los robots accionados por ruedas se ilustran en la tabla 1.

**Tabla1. Diferencias entre robots caminantes y robots accionados por ruedas**

Mobility	<p>Legged robot Wheeled robot Obstacle Obstacle</p>
	<p>Positioning Heading Levelling</p>
Obstacle surmounting	
Active suspension	
Natural terrain (non-continuous terrain)	
Slippage and jamming	<p>Resistance Motion</p>
Environmental damage	<p>Legs Wheels</p>
Average speed	

**Fuente:**GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Advantages of Walking Locomotion. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. p. 17-22.

## **2.3 ESTABILIDAD DE LOS ROBOTS CAMINANTES**

Las investigaciones en la estabilidad de robots caminantes comenzaron en la mitad de 1960 cuando McGhee y Frank (1968) definieron la estabilidad estática de un robot caminante. Siguiendo su definición, un robot ideal es estáticamente estable si la proyección horizontal de su centro de gravedad cae dentro del patrón de soporte.

El robot ideal supone patas con masa despreciable y no se toman en cuenta las componentes dinámicas. La idea de estabilidad estática fue inspirada en los insectos, los cuales usan sus patas para soportar su cuerpo y a la vez proporcionar propulsión. Con la inspiración de los insectos, rápidamente se empezaron a crear mecanismos que emularan la locomoción de los insectos pero el control de estos mecanismos era un poco complicado si se tenía en cuenta las componentes dinámicas y los efectos de inercia del cuerpo, por lo que se optó por limitar el movimiento a velocidades bajas y constantes para facilitar su control.

La manera de mejorar la velocidad de los robots caminantes es teniendo en cuenta los componentes dinámicos, por lo que se generaron mecanismos sencillos con pocos grados de libertad, pero que por su propia simpleza eran prácticamente inútiles en las aplicaciones que se requerían.

**2.3.1 Estabilidad estática en robot caminantes (plano horizontal).**El primer criterio de estabilidad a velocidad constante en una dirección sobre una superficie plana fue propuesto por R.B McGhee y A.A. Frank de la universidad del sur de

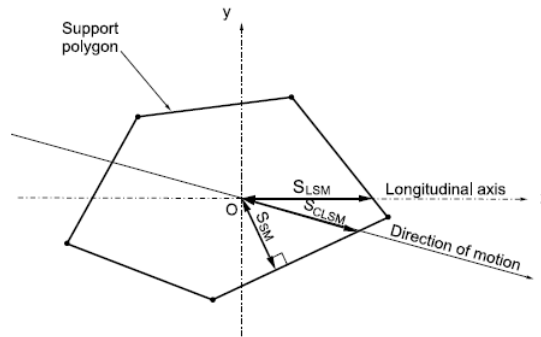
california, Estados Unidos. El método de la proyección del centro de gravedad dice que si la proyección horizontal del centro de gravedad recae dentro del polígono de soporte (polígono formado con la unión de las patas en apoyo), el vehículo será estáticamente estable.

Este criterio fue extendido posteriormente a terrenos disparejos redefiniendo el polígono de soporte como la proyección horizontal del polígono de contacto. Para el análisis de la estabilidad dentro del polígono, se definió el margen de estabilidad estática (SSM por sus siglas en inglés) como la mínima distancia desde el centro de gravedad a uno de los bordes del polígono de estabilidad. Debido a que este margen es en la mayoría de los casos difícil de calcular, Chang-De Zhang y Shin-Min Song propusieron el margen de estabilidad estática longitudinal (SLSM), definido como la más pequeña distancia desde la proyección del centro de gravedad hasta el borde frontal y trasero del polígono de estabilidad a lo largo del eje longitudinal del vehículo. El SLSM es una buena aproximación del SSM y es en muchas veces más fácil de calcular.

Para aplicaciones donde se tiene en cuenta los efectos inerciales y de aceleración, es más conveniente utilizar el margen de estabilidad estática longitudinal del escarabajo (SCSLM) el cual fue definido por Chang-De Zhang y Shin-Min Song como la más pequeña distancia desde la proyección del centro de gravedad hasta el borde del polígono de estabilidad en dirección del movimiento.

En la figura 1 se presentaran los márgenes de estabilidad

**Figura 1. Márgenes de estabilidad para vehículos caminantes**



Fuente: GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Static Stability Criteria. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. p. 34-36.

En la imagen se puede observar un polígono de estabilidad para un hexápodo donde 5 de sus patas están soportando el peso del cuerpo.

Una mejor medida de la estabilidad fue propuesta por Messuri en 1985 donde definió el margen de estabilidad estática basada en la energía (SESM) donde se toma el valor más pequeño de la energía potencial requerida para volcar el robot sobre uno de los bordes del polígono de estabilidad. Matemáticamente se tiene:

$$S_{ESM} = \min(mgh_i)$$

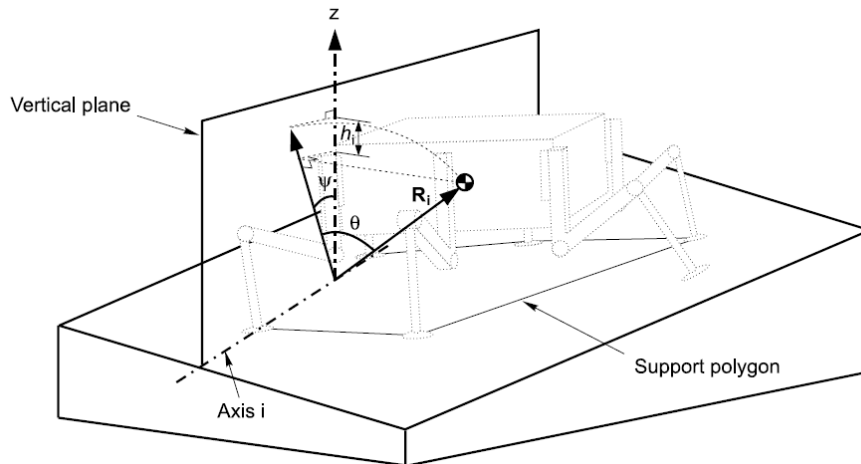
Donde  $i$  es el segmento del polígono considerado en rotación y  $h_i$  es la variación del centro de gravedad durante la volcadura del vehículo, la cual se obtiene de:

$$h_i = |R_i|(1 - \cos\theta)\cos\psi$$

Donde  $R_i$  es el vector desde el eje de rotación hasta el centro de gravedad,  $\theta$  es el ángulo que forma  $R_i$  con el plano vertical y  $\psi$  es el ángulo de inclinación del eje de rotación (borde del polígono de estabilidad) con el plano horizontal.

En la figura 2 se observa un diagrama geométrico para el cálculo del SESM

**Figura 2. Diagrama geométrico para el cálculo del SESM.**



Fuente: GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Static Stability Criteria. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. p. 34-36.

En algunas ocasiones es más eficiente establecer el margen de estabilidad estática normalizado basado en energía o SNESM propuesto por Hirose et al. (1998).

Este considera el peso del vehículo de la siguiente forma:

$$S_{NESM} = \frac{S_{ESM}}{mg} = \min(h_i).$$

**2.3.2 Criterio de estabilidad dinámica.** Como se mencionó anteriormente, los robots caminantes están limitados en cuanto a su velocidad de avance siendo está lo suficientemente baja como para ser analizados como un movimiento cuasi

estático donde las fuerzas generadas por la inercia no son tomadas en cuenta y no influyen en la estabilidad del robot.

Para aumentar el rendimiento y la velocidad de los robots caminantes es necesario tener en cuenta los efectos inerciales tanto de fuerzas como de momentos que se generan durante el movimiento del robot para de esta manera poder desarrollar un criterio de estabilidad más robusto.

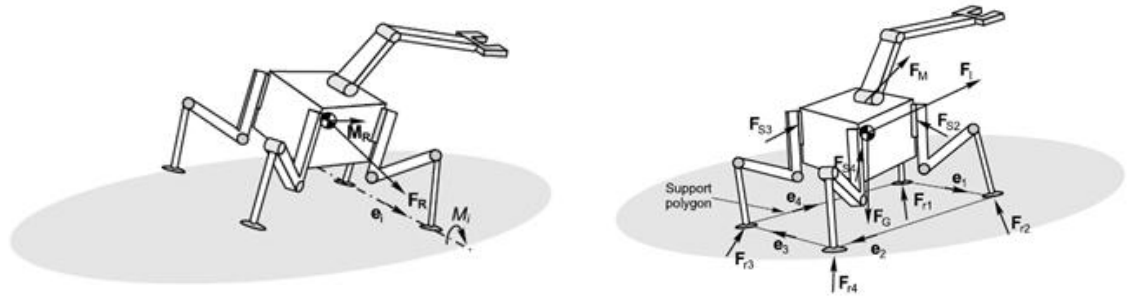
Dado un robot caminante con un sistema de manipulación, (figura 3), las fuerzas y los momentos que actúan en el centro de gravedad pueden desestabilizarlo y hacerlo caer.

El equilibrio dinámico requiere:

$$F_{inercia} = F_{soporte} + F_{gravitacional} + F_{manipulacion}$$
$$M_{inercia} = M_{soporte} + M_{gravitacional} + M_{manipulacion}$$

En la figura 3 se puede observar las fuerzas y momentos que actúan sobre el robot y el efecto de desestabilización que estas generan haciendo rotar el robot sobre uno de los ejes del polígono de soporte

**Figura 3. Fuerzas y momentos que actúan sobre el robot**



Fuente: GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Dynamic Stability Criteria. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. p. 37-41.

Cuando el robot se inclina sobre el eje del polígono, la fuerza y el momento resultante, generan un momento  $M_i$  sobre el eje  $e_i$  que tiende a hacer caer el robot, por lo que para mantener la estabilidad del mismo es necesario un momento de magnitud  $M_i$  pero en sentido contrario.

Basado en esto, Ben Shen Lin y Shin Min Song en 1993 redefinieron el margen de estabilidad dinámica (SDSM) como el momento más pequeño  $M_i$  para cada eje de rotación en el polígono de soporte, normalizado para el peso del sistema esto es:

$$S_{DSM} = \min\left(\frac{M_i}{mg}\right) = \min\left(\frac{e_i * (F_R \times P_i + M_R)}{mg}\right)$$

Donde  $P_i$  es el vector posición desde el centro de gravedad hasta el  $i$ ésimo pie de soporte y  $e_i$  es el vector unitario del eje del polígono de soporte que va en dirección de las manecillas del reloj.

## 2.4 MODOS DE CAMINAR (GAITS)

Las primeras investigaciones realizadas en este campo empezaron con la observación de los distintos modos de caminar que se encontraban en la naturaleza. Estos modos de caminar se intentaron comprender para posteriormente formularse matemáticamente, se observó que muchas de las formas de caminar eran periódicas y continuas.

Periódicas porque la secuencia de pasos se repetía después de un determinado tiempo y continuas porque el cuerpo se encuentra en constante movimiento mientras las patas se mueven simultáneamente.

Cuando las irregularidades del terreno son muy grandes como para mantener un gait continuo, los mamíferos y los insectos pueden cambiar su forma de caminar a una más segura. El cuerpo es propulsado hacia arriba y hacia abajo con todas sus patas ubicadas con seguridad en el piso, luego las patas son transferidas a una nueva posición de avance (puede ser una por una o en trípodas) mientras el cuerpo permanece estático.

Esta forma de caminar se conoce como gait discontinuo debido a que el movimiento del cuerpo se da de manera intermitente. Un gait discontinuo presenta muchas ventajas respecto de los continuos, a continuación se nombraran las ventajas de los gait discontinuos:

- Mayor facilidad de implementación.
- Ofrecen mayor margen de estabilidad longitudinal.
- Pueden alcanzar mayor velocidad de movimiento.
- Se adaptan mejor a los terrenos irregulares debido a que la transferencia de la pata se realiza a menor velocidad hasta que este posicionada para avanzar, lugar donde hace contacto con el piso.

Algunas aplicaciones de robots caminantes requieren que la secuencia de pasos pueda ser cambiada dependiendo de las condiciones del terreno, por ejemplo un robot que deba evitar zonas de peligro como minas antipersonas, debe poder cambiar de una secuencia periódica de movimiento a una secuencia que se adapte a las condiciones del terreno que puede ser cualquiera. Este tipo de gaits son llamados no-periódicos o también conocidos como secuencia libre.

**2.4.1 Generación de gaits.** La palabra gait viene del inglés y es definida por Song y Waldron en 1989 como:

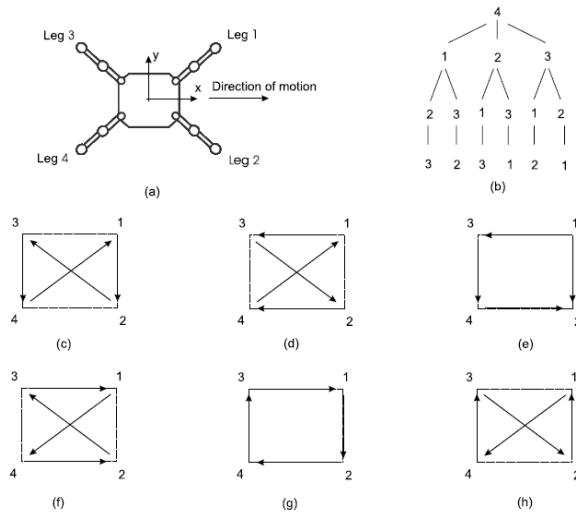
“un gait es definido por el tiempo y localización de ubicación y levantamiento de cada pie, coordinado con el movimiento del cuerpo en sus seis grados de libertad, con el fin de mover el cuerpo desde un lugar a otro.”

Los primeros modelos matemáticos para los gaits fueron desarrollados por McGhee y Frank en 1968 y se centraron principalmente en los cuadrúpedos estos investigadores lograron encontrar que gaits podían mantener la estabilidad del robot, para ello crearon una notación llamada la secuencia de eventos, en donde en un robot con  $n$  patas, el posicionamiento de la pata  $i$  se denota como el evento  $i$  mientras el levantamiento de la pata  $i$  se denota como el evento  $i+n$ .

Para que un cuadrúpedo mantenga la estabilidad, debe mantener al menos tres patas en apoyo, es decir una sola pata en transferencia, esto implica que después del levantamiento de la pata  $i$ , el apoyo de la pata  $i$  debe ocurrir. Esto reduce los posibles gaits que mantienen la estabilidad a  $N = (n - 1)!$ , que en el caso de los cuadrúpedos,  $n = 4$  por lo tanto  $N = 6$ .

Estos seis gaits son mostrados en la figura 4.

**Figura 4. Gaits que aseguran la estabilidad de un cuadrúpedo.**



Fuente: GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Generation of Periodic Gaits. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. p. 57-63

Todos los gaits mostrados anteriormente se dice que son creeping gaits debido a que todos los patrones de soporte involucran al menos  $n-1$  puntos de apoyo, para el caso de los cuadrúpedos ( $n=4$ ) involucra el trípede de estabilidad como patrón de apoyo.

A continuación se mostraran las principales características que tienen los tipos de gaits continuos y discontinuos.

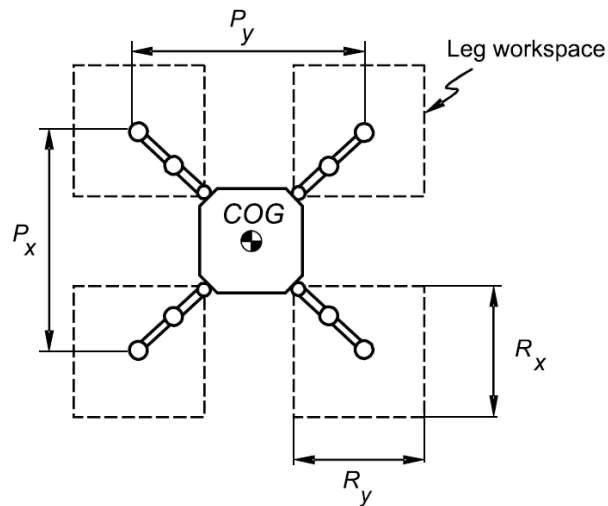
**2.4.1.1 Gaits continuos.** Este modo de caminar, es el que se ve más ampliamente tanto en cuadrúpedos naturales como en artificiales. Es necesario tener en cuenta las siguientes definiciones que se tienen para este tipo de movimientos en robots caminantes.

- **Factor de deber.** Este factor se denota como  $B_i$ , siendo este el factor de la  $i$ -ésima pata. Hace referencia a la fracción del ciclo para el cual la  $i$ -ésima pata se encuentra en la tierra (apoyo).
- **Fase de la pata.** Denotada como  $\phi_i$  (para la  $i$ -ésima pata). Es el tiempo normalizado en el que se da la ubicación de la pata  $i$  en el piso después de la ubicación de la pata 1 (la pata 1 se toma como referencia de la fase).
- **Alcance de la pata.** Denotado por la letra  $R$ . Es la distancia a la que se encuentra el pie relativo al cuerpo durante la fase de apoyo,  $R$  debe estar dentro del espacio de trabajo de la pata definido por  $R_x$  y  $R_y$ .
- **Distancia entre pies.** Denotada por la letra  $P$ . es la distancia entre los centros de patas adyacentes, existe la distancia colateral  $P_x$  y la distancia contra lateral  $P_y$ .
- **Longitud de avance.** Denotada por la letra  $\lambda$ . Es la distancia viajada por el centro de gravedad del robot caminante en un ciclo de locomoción. Si el modo de caminar es periódico, la longitud se calcula así:

$$\lambda = \frac{R}{\beta}$$

Para una mejor visualización de las definiciones anteriores, estas serán mostradas en la figura 5.

**Figura 5. Definiciones geométricas para gaites continuos.**



Fuente: GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Generation of Periodic Gaits. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. p. 57-63

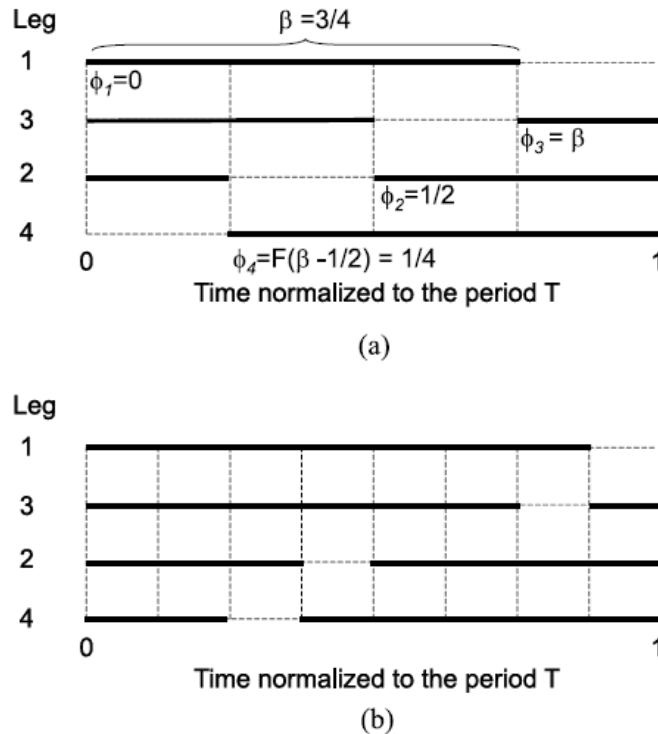
Como se mencionó anteriormente, los gaites continuos se caracterizan por hacer que el centro de gravedad del robot permanezca en constante movimiento durante la secuencia. La secuencia de pasos puede ser representado por medio de un diagrama de gaites, en términos del el factor de deber y la fase de cada pata.

En la figura 6 se ilustran dos gráficas para dos factores diferentes de deber o de apoyo.

Las líneas oscuras representan el estado de apoyo de una pata mientras las líneas punteadas representan el estado de transferencia de las mismas, en la figura se puede observar las fracciones de periodo en las que cada una de las patas se encuentran en apoyo o transferencia según sea el modo de caminar.

Para esta figura solo una pata se encuentra en transferencia mientras las demás están en apoyo por lo que es un creeping gait como se mencionó anteriormente.

**Figura 6. Diagrama de gaits para cuadrúpedos en creeping gait (wave)**



Fuente: GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Generation of Periodic Gaits. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. p. 57-63

**2.4.1.2 Gaits discontinuos.** Este tipo de secuencias se caracterizan porque una de las patas se encuentra en transferencia mientras las demás están sujetas y estáticas, la propulsión del cuerpo del robot se da con la totalidad de las patas simultáneamente una vez todas ya están debidamente posicionadas.

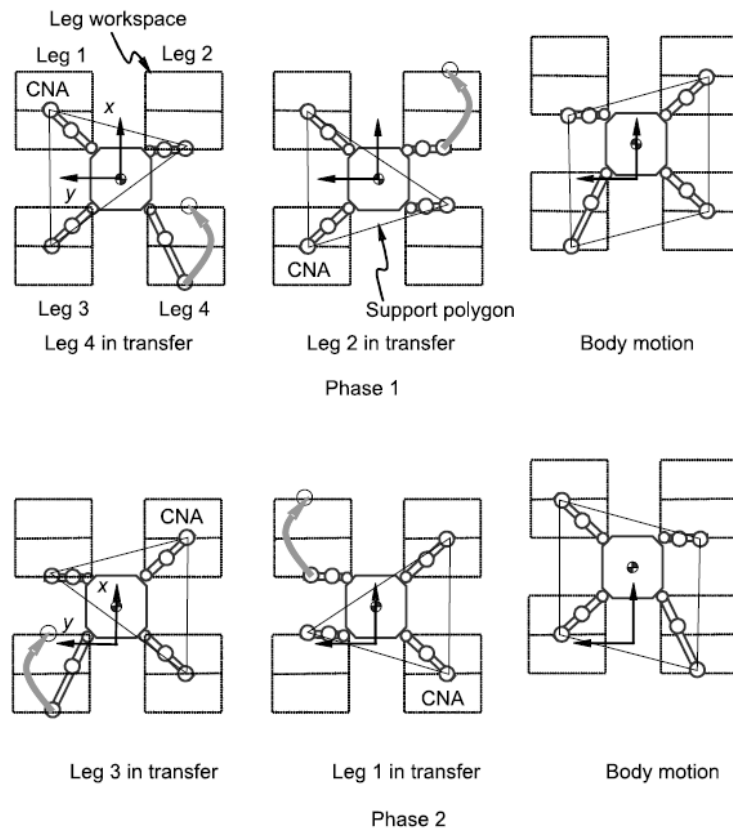
Para generar secuencias discontinuas periódicas se deben tener en cuenta algunos aspectos importantes:

- Si una de las patas alcanza el límite trasero mientras está en fase de apoyo, esta debe ser transferida a su posición límite frontal.
- Como el movimiento se realiza con todas las patas al mismo tiempo, una vez terminado el movimiento de avance, al menos una de las patas debe encontrarse en el límite cinemático trasero.
- Se debe asegurar que a la hora de que la pata en transferencia entre en apoyo, esta pata forme, por lo menos, un trípode de estabilidad con el resto de las patas sin contar la pata que entrara en transferencia seguidamente.

**2.4.1.3 Fases de los ciclos.** Las fases en un ciclo discontinuo, se conocen como la cantidad de avances que se obtienen en un ciclo de movimiento de las patas.

En la figura 7 se muestra un esquema de un avance discontinuo de dos fases.

**Figura 7. Secuencia de avance discontinuo de dos fases para un cuadrúpedo.**



**Fuente:**GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Generation of Periodic Gaits. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. p. 57-65.

## 2.5 MECANISMOS PARA ROBOTS CAMINANTES

A la hora de diseñar un robot caminante, un aspecto muy importante que se debe tener en cuenta, es el tipo de mecanismo o configuración de motores que comprenderán las articulaciones de las patas del robot.

A la configuración de motores que comprenden las articulaciones de las patas se le denomina SCARA, que proviene del acrónimo en inglés Selective Compliant Articulated Robot Arm.

Los mecanismos basados en un componente que gira 360 grados se denominan mecanismos de barras con manivela, estos mecanismos por lo general son de cuatro barras unidas de tal manera que se genera una trayectoria determinada. Esta trayectoria depende en gran medida de las longitudes de las barras y de los puntos de unión de las mismas.

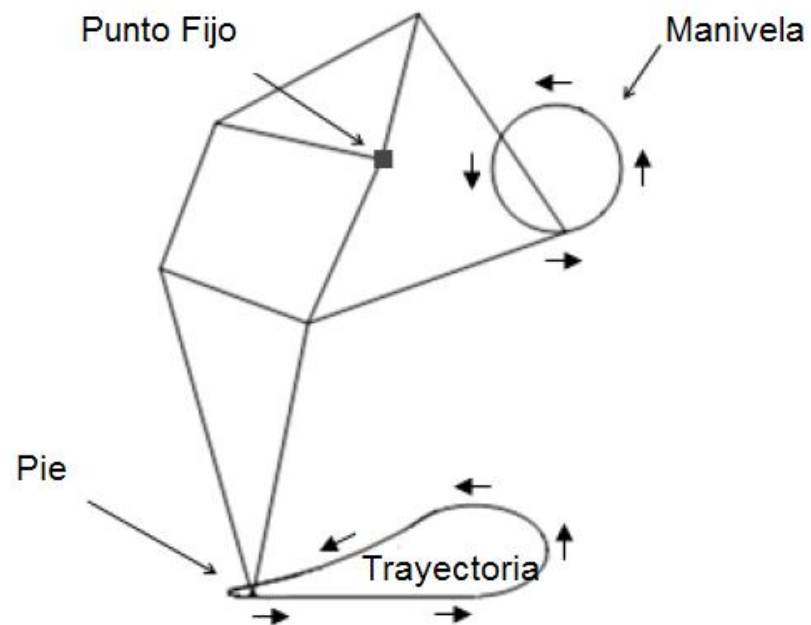
El pie de un mecanismo caminante es la parte del mecanismo que está en contacto directo con el piso y es el encargado de transmitir el movimiento al robot por medio de la reacción con el piso, el pie debe describir una trayectoria cíclica a medida que la manivela va girando, de esta manera se aseguran las fases del pie que son las siguientes:

- Apoyo
- Avance
- Levantamiento
- Posicionamiento

Estas fases son necesarias en cualquier mecanismo caminante y deben cumplir con algunos requisitos dependiendo de la aplicación que se desee, por ejemplo la velocidad de avance, posición de apoyo, posición de levantamiento, longitud de avance, entre otros.

En la figura 8 se puede observar un mecanismo caminante con sus partes.

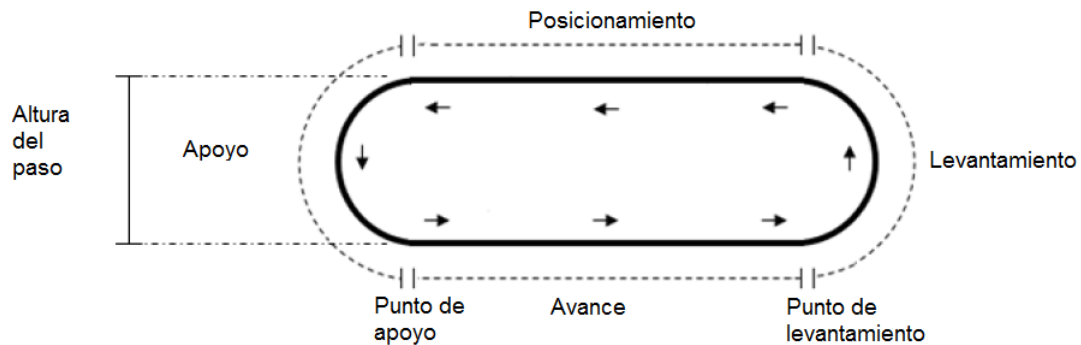
**Figura 8. Partes de un mecanismo caminante**



Fuente: GHASSAEI, Amanda; CHOI, Phil y WHITAKER, Dwight. The Design and Optimization of a Crank-Based Leg Mechanism. Pomona College Department of Physics and Astronomy, 2011. 172p

Las fases de la trayectoria del pie pueden ser observadas en la figura 9, donde teóricamente el avance se da mientras el pie se encuentra en apoyo con el suelo y las flechas indican la dirección del movimiento del pie.

**Figura 9. Fases de la trayectoria del pie.**



Fuente: GHASSAEI, Amanda; CHOI, Phil y WHITAKER, Dwight. The Design and Optimization of a Crank-Based Leg Mechanism. Pomona College Department of Physics and Astronomy, 2011. 172p

El diseño de los mecanismos caminantes debe encaminarse a la eficiencia energética y a la estabilidad del movimiento, estos factores son de gran importancia a la hora de implementar estos mecanismos a aplicaciones robóticas móviles. Para lograr esto se debe tener en cuenta los siguientes aspectos:

- La trayectoria de avance debe tener velocidad uniforme.

Mientras la aceleración relativa al suelo del pie sea cero, favorece el avance del mecanismo debido a que reduce las posibilidades de arrastrar el pie por el suelo sin lograr avance alguno o muy poco.

- La barra manivela debe tener velocidad angular constante durante el ciclo completo de locomoción.

Esto es requerido ya que de esta forma se obtiene una velocidad uniforme en cada una de las patas del robot y además asegura que la trayectoria de velocidad constante del mecanismo no se vea afectada por variaciones en la rotación de la manivela.

- Las fuerzas inerciales y los torques inerciales deben estar balanceados durante la totalidad del ciclo de locomoción.
- El rango de duración del pie en fase de avance con relación al ciclo completo debe ser mayor o igual a 0.5.

Para garantizar una locomoción estáticamente estable, es beneficioso maximizar el tiempo en el que el pie permanece en apoyo con el suelo.

A continuación se mostraran los mecanismos caminantes que se han desarrollado para los diferentes propósitos de la robótica.

**2.5.1 Mecanismo de Hoekens.** Consiste básicamente en un mecanismo biela-manivela (ver figura 10). Con este mecanismo se consigue convertir el movimiento circular que nos aporta el motor en un movimiento recto, la mitad del ciclo, y elíptico la otra mitad. El movimiento recto del punto de referencia corresponderá con el período durante el cual el pie este totalmente apoyado, mientras que el otro pie se encontrará en el período de movimiento elíptico.

El movimiento elíptico del pie que se encuentre en el aire lo desplazará hacia delante hasta que se encuentre totalmente apoyado, momento en cual comenzará a soportar todo el peso del robot y comenzará el período de movimiento rectilíneo de su punto de referencia y, en consecuencia, el período de movimiento elíptico del otro pie.

**Figura 10. Mecanismo de Hoekens**

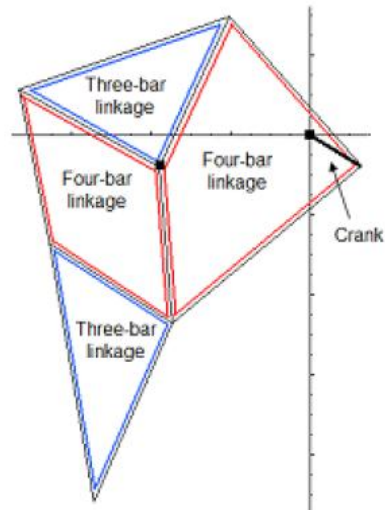


Fuente: ESPANTOSO MIRANDA, Víctor. Modelado y simulación del robot PASIBOT. Estudio de la rigidez y mejora en la prevención del vuelco lateral. Madrid: Universidad Carlos III de Madrid. 83 p.

**2.5.2 Mecanismo de Theo Jansen.** Este mecanismo está compuesto por dos mecanismos de cuatro barras unidas entre sí en serie con solo uno de los mecanismos de cuatro barras manejado directamente por la manivela, los mecanismos de cuatro barras comparten una barra en común y son separados por una estructura triangular rígida en un lado, adicionalmente otro triángulo rígido es unido a la unión para crear el pie del mecanismo.

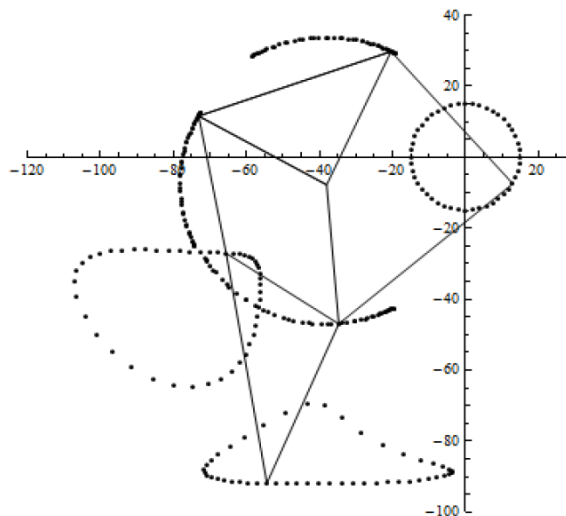
En la figura 11 puede observarse lo anteriormente descrito para el mecanismo de Theo Jansen mientras que en la figura 12 se puede observar las trayectorias.

**Figura 11. Mecanismo de Theo Jansen.**



Fuente: GHASSAEI, Amanda; CHOI, Phil y WHITAKER, Dwight. The Design and Optimization of a Crank-Based Leg Mechanism. Pomona College Department of Physics and Astronomy, 2011. 172p

**Figura 12. Trayectorias mecanismo Theo Jansen**



Fuente: GHASSAEI, Amanda; CHOI, Phil y WHITAKER, Dwight. The Design and Optimization of a Crank-Based Leg Mechanism. Pomona College Department of Physics and Astronomy, 2011. 172p

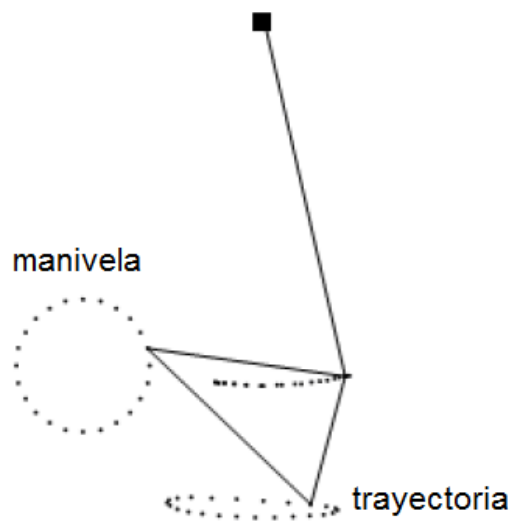
Las trayectorias de estos mecanismos pueden ser optimizadas para ciertas aplicaciones variando las dimensiones de las barras de los mecanismos de cuatro barras y los triángulos rígidos.

**2.5.3 Mecanismo de Hrones-Nelson.** Este mecanismo está compuesto de una estructura triangular unida a un mecanismo de cuatro barras el cual es movido directamente por una manivela.

La trayectoria de este mecanismo no es tan recta durante la fase de avance como en el caso de Theo Jansen, presentando así una forma de lágrima.

Un esquema de este mecanismo puede observarse en la figura 13.

**Figura 13. Mecanismo Hrones-Nelson**



Fuente: GHASSAEI, Amanda; CHOI, Phil y WHITAKER, Dwight. The Design and Optimization of a Crank-Based Leg Mechanism. Pomona College Department of Physics and Astronomy, 2011. 172p

Estos conceptos preliminares sobre robots caminantes, mecanismos, ventajas y características sirven como base para el estudio de robots capaces de trepar superficies rugosas como árboles y demás, en los siguientes capítulos se abordará el tema de robots trepadores con análisis estáticos de estabilidad, sujeción a la superficie entre otros.

### **3. ROBOTS TREPADORES O ESCALADORES**

En la actualidad, han sido varias las aplicaciones que requieren de robots que puedan escalar superficies verticales o inclinadas. El desarrollo de robots escaladores ha tomado mucho auge y han sido varios los intentos realizados por suplir esta necesidad y los mecanismos implementados para adherirse a la superficie de trabajo van desde elementos de vacío, fijación electromagnética, adhesivos, garras, fuerzas intermoleculares entre otras, muchos de estos han obtenido resultados favorables y otros no tanto pero todos han colaborado con la creación de conocimiento en esta área de la robótica.

Para empezar este capítulo, se empezará con un breve repaso de los robots más simbólicos de este campo, sus características y aspectos que llevaron a cada uno de estos a generar conocimiento.

#### **3.1 SCREENBOT**

Es un hexápodo que puede escalar verticalmente e invertido en una pantalla de malla utilizando espinas. SCREENBOT fue diseñado para mostrar la factibilidad de la sujeción interior distribuida o DIG (Distributed inward Gripping) en una plataforma simple.

Utilizando un grado de libertad el mecanismo de patas mueve estas en una secuencia predeterminada que hace que cada pie hale hacia el interior una vez ha hecho contacto con la pantalla.

Cada una de las patas son manejadas por un motor a través de cadenas y engranajes, debido a que este robot solo posee un grado de libertad, limita su movimiento en línea recta y este movimiento es radio controlado debido a que no

posee una tarjeta de procesamiento en su interior. La velocidad de este robot es de 0.08 veces el largo de su cuerpo por segundo.

En la figura 14 se puede observar una foto del SCREENBOT durante un ascenso vertical.

**Figura 14. SCREENBOT del CWRU biorobotics lab.**



Fuente: WILE, Gregory D, et al. Screenbot: Walking Inverted Using Distributed Inward Gripping. En: International Conference on Intelligent Robots and Systems, 22,Febrero, 2008. 6p.

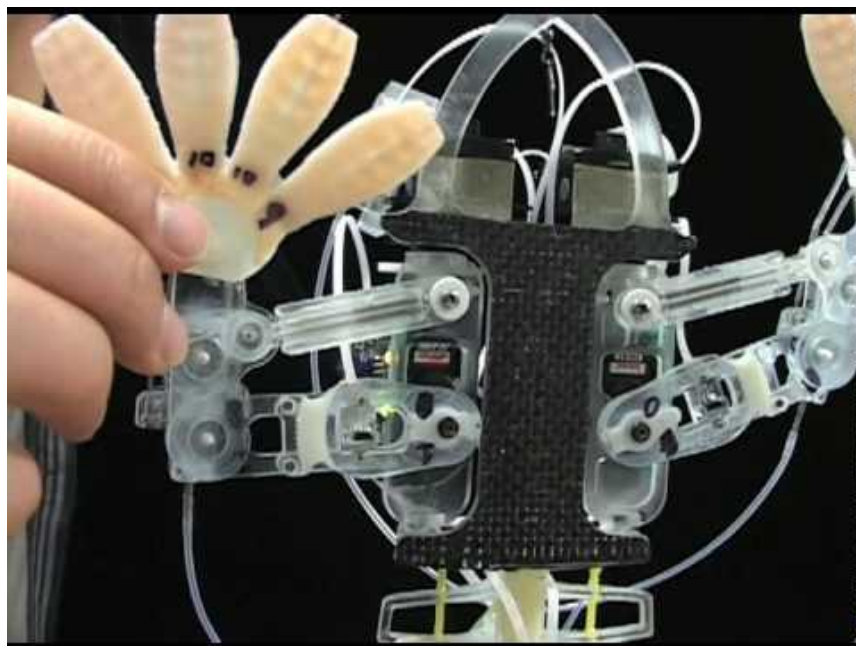
### **3.2 STICKYBOT**

Es un robot hexápodo escalador desarrollado por la universidad de Stanford que usa almohadillas de adhesivos direccionales lo que le permite escalar en superficies lisas tales como el vidrio y la baldosa.

El principal objetivo de este proyecto fue la creación de una superficie que pueda adherirse direccionalmente a superficies lisas para obtener un mejor desempeño. STICKYBOT escala a velocidades lentas de aproximadamente 0.05 longitudes del cuerpo por segundo.

En la figura 15 se puede observar una imagen del STICKYBOT .

**Figura 15. StickyBot de la Universidad de Standford**



Fuente: SPENKO,M, et al. Smooth vertical surface climbing with directional adhesion. Robotics, IEEE Transactions on, 2008, Vol.24.

### **3.3CLIMBING PARALLEL ROBOT (CPR)**

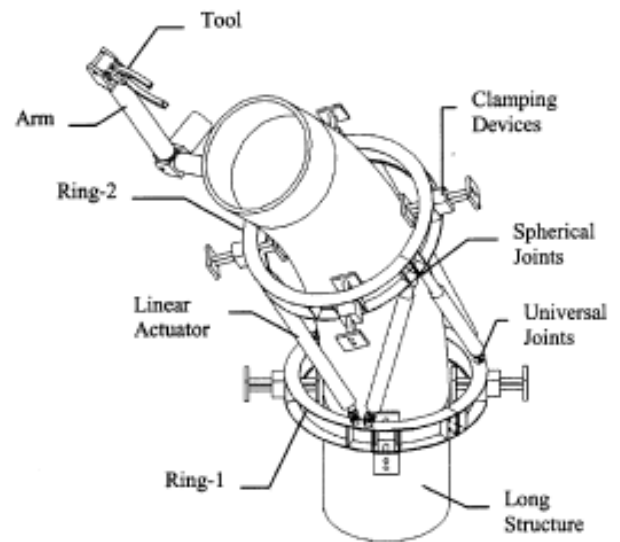
Este robot tiene una estructura basada en la plataforma de Stewart con seis grados de libertad. Dos anillos son unidos mediante seis actuadores lineares, estos actuadores son unidos a la base mediante juntas universales y unidas a la plataforma superior mediante uniones esféricas.

Este robot es capaz de escalar en trayectorias curvas debido a los grados de libertad que ofrecen los seis actuadores. Además de estos actuadores, posee otros actuadores en cada uno de los anillos perpendiculares al eje del anillo los cuales permiten el agarre a la superficie de trabajo.

Este robot fue propuesto para cumplir tareas de fumigación, mantenimiento de tubos de aceite, mantenimiento de postes de luz entre otras.

En la figura 16 se puede observar la estructura del PARALLEL ROBOT.

**Figura16. Parallel Robot**



Fuente: ALMONACID,M, et al.. Motion planning of a climbing parallel robot. IEEE transactions on robotics and automation, 2003. Vol.19.

### **3.4GECKOBOT**

Este es un robot inspirado en el sistema de agarre que presentan los geckos, usa un PIC como su controlador principal y es capaz de escalar sobre superficies planas, este robot posee una cola la cual permite ejercer una fuerza normal sobre las patas delanteras, asegurando un mejor agarre y una mayor estabilidad durante su movimiento.

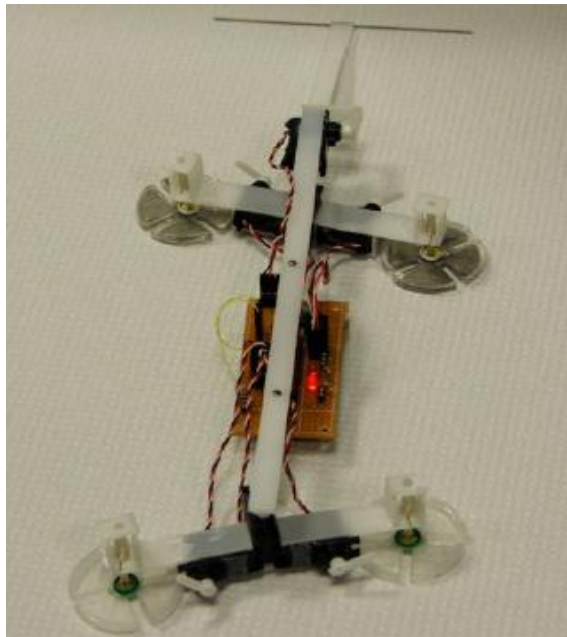
El mecanismo de movimiento está compuesto por un mecanismo de cuatro barras que mueve dos patas diagonales opuestas mediante servomotores, esto permite el avance estable del robot.

El robot también permite dirigir el cuerpo a diferentes ángulos, para esto presiona fuertemente con la cola de manera que las patas traseras se levantan levemente permitiendo el giro con las patas delanteras mientras la cola se va deslizando.

Este robot fue elaborado en la Universidad de Carnegie Mellon Estados Unidos.

En la figura 17 se puede apreciar el geckobot durante su movimiento.

**Figura 17. Geckobot**



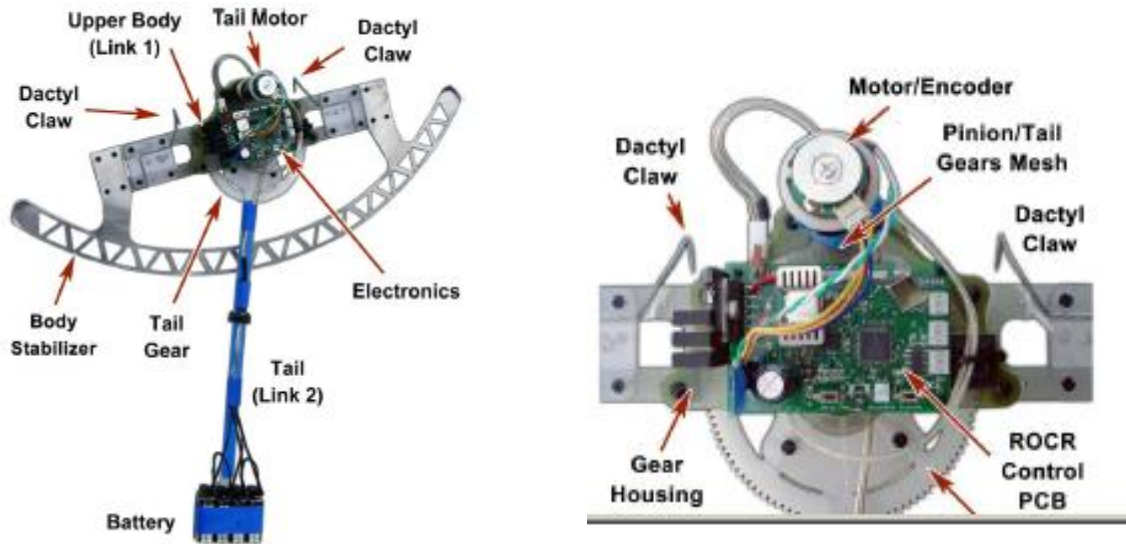
Fuente: UNVER,Ozgur; UNERI, Ali y SITTI, Metin. Geckobot: A Gecko Inspired Climbing Robot Using Elastomer Adhesives. IEEE International Conference on Robotics and Automation, 2006.

### **3.5 ROCR WALL CLIMBING ROBOT**

ROCR es un robot pendular de dos barras, el cual posee una cola que pivota unida al centro del primer eslabón. Posee dos garras como mecanismo de agarre las cuales se encuentran ubicadas a lo ancho de la parte superior del cuerpo, las garras están hechas con acero para resortes de 1 milímetro de diámetro doblado de tal forma que produzca adhesión direccional.

Al final de la cola se encuentran las baterías las cuales además de proveer la potencia al motor, también sirven como masa del péndulo que ayudara al avance del robot mediante el balanceo de la cola (ver figura 18).

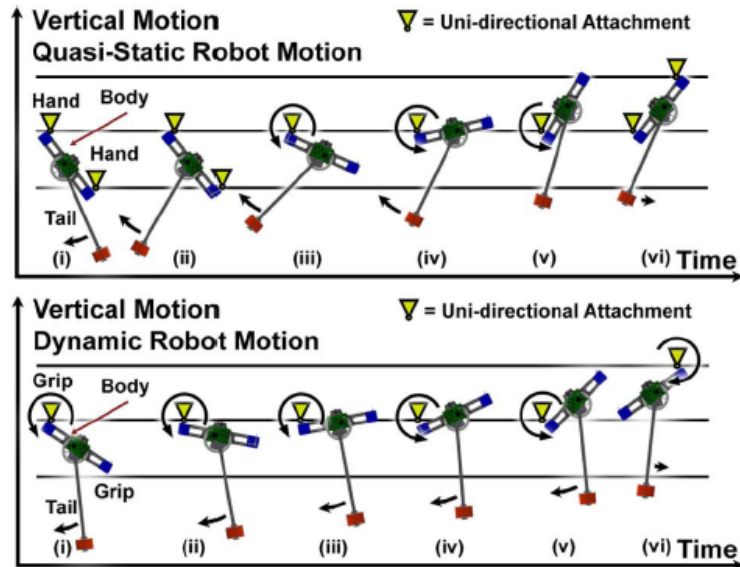
**Figura 18. ROCR Wall climbing robot**



Fuente: PROVANCHER, William R, et al. An energy efficient dynamic wall-climbing robot. IEEE/ASME transactions on mechatronics, 2011. Vol.16.

El movimiento mostrado en la figura 19 es proporcionado por la cola y por los momentos que genera su balanceo.

Figura 19. Movimiento del ROCR



Fuente: PROVANCHER, William R, et al. An energy efficient dynamic wall-climbing robot. IEEE/ASME transactions on mechatronics, 2011. Vol.16.

Como se puede observar, los mecanismos desarrollados para robots escaladores son diversos y muy ingeniosos, se utiliza la biomimética como fuente primaria de información junto con la aplicación de las ciencias como la física.

Mediante el uso de la estática se buscara establecer criterios para garantizar la estabilidad en robots escaladores, para esto se partirá de la suposición de diferentes casos en el cuales se analizaran las fuerzas generadas para cada caso y su influencia en la estabilidad del robot.

## **4. ACTUADORES EN ROBOTICA MOVIL**

En capítulos anteriores se habló sobre mecanismo que permiten a los robots caminantes su avance por diferentes terrenos y hacen posible la locomoción del robot. Estos mecanismo deben ser accionados por algún elemento que les suministre la potencia y genere el movimiento, estos elementos son llamados actuadores y los hay de diversos tipos.

- Hidráulicos
- Neumáticos
- Eléctricos

Los actuadores más ampliamente utilizados en robótica son los eléctricos debido a que son muy precisos y fiables, además no producen mucho ruido y tanto el control como la instalación son sencillos. Los actuadores eléctricos son usados en aplicaciones de robótica donde la potencia requerida no es muy alta ya que estos actuadores poseen una potencia limitada.

### **4.1 SERVO MOTOR**

Este es un motor similar a un motor de corriente continua convencional, con la diferencia de que este posee un circuito de control que le permite ubicarse en cualquier posición dentro del rango de operación del mismo y mantenerse en esta posición de manera estable.

El servomotor está conformado por un motor de corriente continua cuyo eje se encuentra acoplado a una caja de engranajes los cuales aumentan su torque, al final de la caja de engranajes se encuentra un potenciómetro acoplado al eje de estos el cual cumple la función de sensar la posición a la que se encuentra la salida del motor para controlar su ubicación. Este potenciómetro está conectado a

la tarjeta de control que posee el servo en donde se realiza la acción de control necesaria para estabilizar el servo.

**4.1.1 Conexiones y funcionamiento.** Los servomotores tienen tres cables de conexión, el primer cable es generalmente la tierra del servomotor y se encuentra en la mayoría de los casos en color café, el cable del medio es la alimentación (en la mayoría de casos 5 voltios) y se encuentra de color rojo, por último está el cable de control que se encuentra de color amarillo.

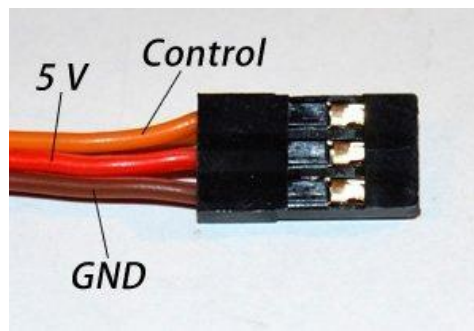
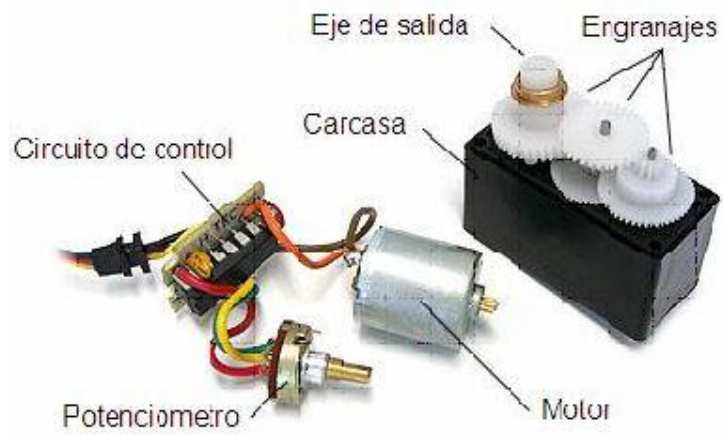
Para posicionar el servomotor, se envía el ángulo al que se quiere posicionar por medio de modulación de ancho de pulsos, la duración del pulso dictamina la posición a la que se quiere tener el servo. Un pulso de 1.5 ms., por ejemplo, hará que el motor se torne a la posición de 90 grados (llamado la posición neutra). Si el pulso es menor de 1.5 ms., entonces el motor se acercará a los 0 grados. Si el pulso es mayor de 1.5ms, el eje se acercará a los 180 grados.

Esta señal es enviada a la tarjeta interna del servo, la cual comparará la posición actual del servo, dada por el valor del potenciómetro, y la posición a la que se desea tener el servo, dada por los pulsos enviados. Sobre el servo se aplica un control teniendo en cuenta el error, el cual aplicará más voltaje si el error es grande por lo tanto el servomotor se acercará rápidamente a la posición requerida y menos voltaje a medida que se acerca a su Setpoint.

Una característica importante de los servomotores es que trabajan en un rango de operación y no dan vueltas completas esto debido a que los potenciómetros se comportan de esta manera, es por esto que los servomotores son equipados con topes mecánicos para que no dañen los potenciómetros y por ende el servomotor.

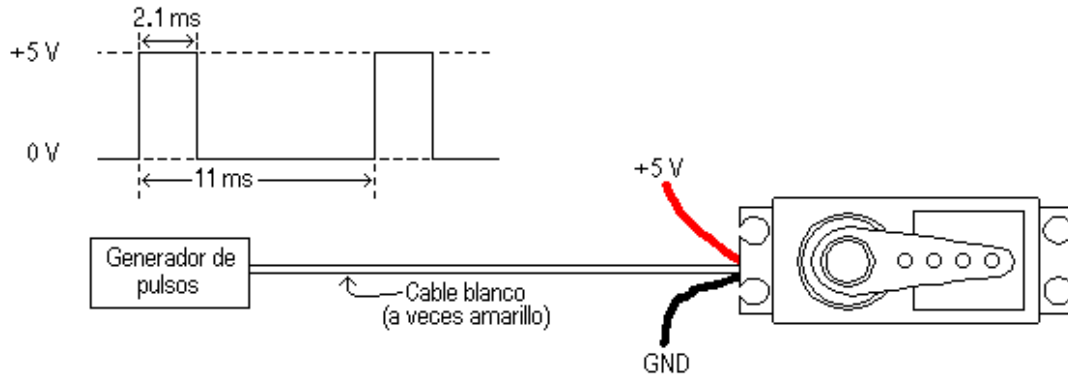
A continuación en las figuras 20 y 21 se mostraran imágenes de servomotores típicos para poder apreciar sus partes.

**Figura 20. Partes de un servomotor.**



Fuente: Anonimo.Estructura del sistema embebido[Online], Enero 2009 [Cited 19 January 2013]. Available from internet: <http://cybertesis.upc.edu.pe/upc>

**Figura 21. Pulsos de posicionamiento del servo**



Fuente:Escuela PolitecnicaSuperior de Albacete.El servomotor [Online], [Cited 6 April 2013]. Available from internet: <http://www.info-ab.uclm.es/labelec/solar/electronica/elementos/servomotor.htm>.

#### **4.2 SERVO TRUCADO**

Los Servomotores trucados son servomotores a los cuales se les ha aislado la retroalimentación de la posición del servo y se les ha cambiado la conexión de la tarjeta de control al potenciómetro interno por una resistencia fija.

La idea de trucar un servomotor es que este pueda girar 360 grados libremente, para esto es necesario remover los topes mecánicos tanto del potenciómetro como el de los engranajes. Al permanecer con la tarjeta de control pero ahora sin una retroalimentación, el servomotor se convertirá en un motor de rotación continua con control de velocidad y dirección de giro.

Si la resistencia fija que se puso en el lugar del potenciómetro, equivale a la resistencia que marcaría el potenciómetro en la posición de 90 grados por ejemplo, al mandar por el cable de control unos pulsos equivalentes a la posición de 90 grados, el servomotor permanecerá detenido debido a que la tarjeta de

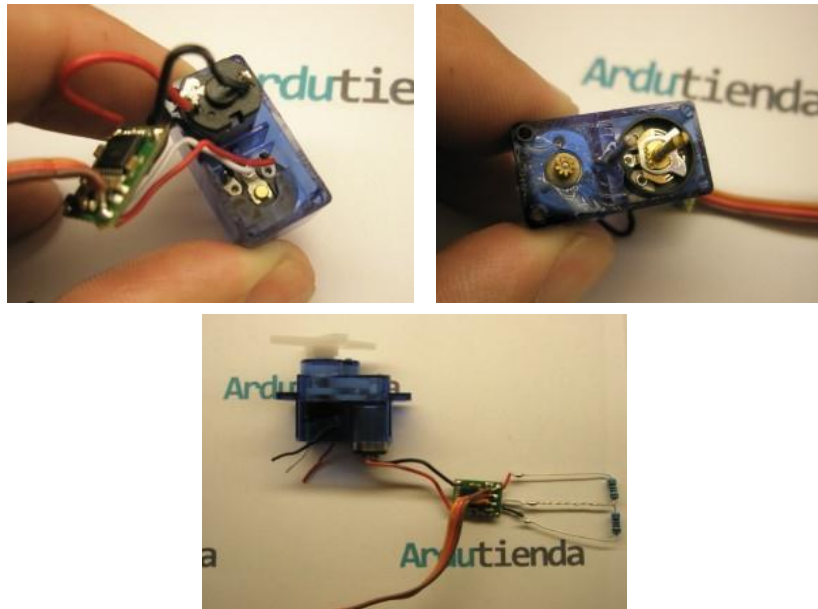
control interpretará que se ha llegado satisfactoriamente al Setpoint establecido por la duración de los pulsos del cable de control.

Si los pulsos enviados por el cable de control equivalen a una posición de 95 grados por ejemplo, el servomotor se moverá lentamente en dirección hacia donde aumente la posición según estaba configurado el potenciómetro, en este caso la velocidad del servo será pequeña debido a que 95 grados no está muy lejos de la posición a la que la tarjeta percibe que esta el motor (90 grados debido al valor de resistencia fija que se introdujo al sistema) y al no existir retroalimentación, el servomotor girará de forma continua intentando buscar la posición de 95 grados.

De igual forma si los pulsos enviados equivalen a una posición que está lejana de la posición de 90 grados, el servomotor se moverá rápidamente según sea el caso hacia un lado para valores mayores que 90 y hacia el otro para valores menores que 90.

De esta forma se consigue controlar la velocidad y dirección de giro del servomotor variando el ancho de pulsos de control. Ver en la figura 22 el proceso de trucado.

**Figura 22. Trucado de un Servomotor.**



Fuente: ArduTienda. Trucar servo a rotación continua [Online], [Cited 10 April 2013]. Available from internet: <http://www.ardumania.es/trucar-servo-a-rotacion-continua/>.

Este mismo resultado se puede obtener mediante la realización de un control de velocidad para un motor eléctrico convencional de corriente directa.

### **4.3 MICROMOTORES**

Una opción en cuanto a actuadores en robótica son los llamados micro motores, estos son motores de corriente continua de alta potencia que vienen con una caja de reducción hasta de 1000 a 1 con un torque máximo hasta de 9 kg-cm y girando a unas 35 RPM (Ver figura 23).

**Figura 23. Micromotor relación 298:1**



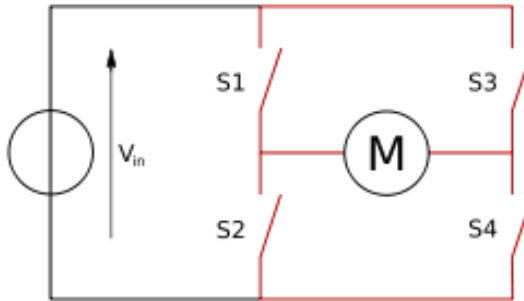
Fuente: Dynamo electronics.Micromotor [Online], [Cited 10 April 2013]. Available from internet: <http://www.dynamoelectronics.com/>.

Estos motores se caracterizan por ser compactos y livianos, que son factores importantes en robots escaladores donde el peso es un aspecto limitante en el diseño. Estos motores no contienen ningún tipo de tarjeta de control por lo que es necesario hacer un circuito de control para estos motores.

El control de los motores de corriente continua es sencillo y de bajo costo, se utiliza un puente H para controlar la dirección de giro del motor y por medio de modulación de ancho de pulso se puede controlar además la velocidad a la que gira el motor.

Estos puentes H se pueden construir pero puede dar un poco de trabajo, por lo que en robótica pequeña se suele usar modelos que se encuentran en circuitos integrados. El nombre de puente H viene dado debido a que su arquitectura se asemeja a la letra H como se puede observar en la figura 24.

**Figura 24. Arquitectura del puente H.**



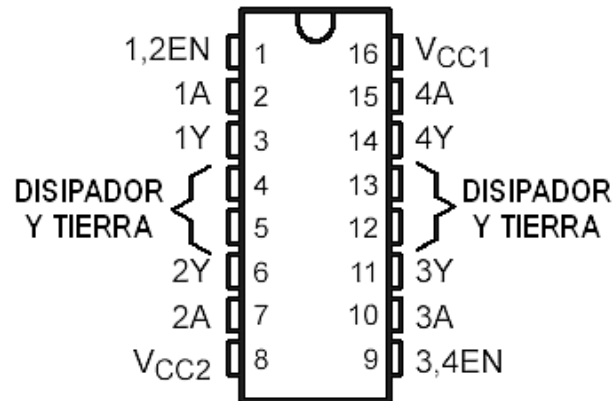
Fuente: Cyril BUTTAY. Structure of an H-bridge, Junio 2006 [Cited April 15 2013]. Available from internet: [http://es.wikipedia.org/wiki/Puente\\_H\\_\(electronica\)](http://es.wikipedia.org/wiki/Puente_H_(electronica)).

Un puente H se construye con cuatro interruptores (pueden ser mecánicos o transistores). Cuando los interruptores S1 y S4 están cerrados y S3 con S2 están abiertos (ver figura 24), se aplica una tensión positiva en el motor haciéndolo girar en un sentido. Cuando se abren los interruptores S1 y S4 cerrando los dos restantes, el voltaje se invierte, permitiendo así el giro en sentido contrario.

Hay diferentes circuitos integrados con la arquitectura de un puente H, los más sencillo son los L293D y SN754410. Aunque es recomendable mirar el datasheet suelen dar unos 1200mA y mediante cada circuito integrado se puede controlar 2 motores (600mA cada uno).

En la figura 25 se puede ver como los pines de un lado (1, 2, 3, 4, 5, 6 y 7) son para controlar un motor y los pines del otro lado (9, 10, 11, 12, 13, 14 y 15) son para controlar un segundo motor. El pin 8 hay que conectarlo a la fuente de alimentación con la que se alimenta el motor y el pin 16 es la alimentación propia del chip. Recuerda que si se usan más de una fuente de alimentación (pila motor y pila chip) tienen que tener las tierras comunes (4, 5, 12 y 13) para que tengan la misma referencia.

**Figura 25. Integrado L293D NE package**

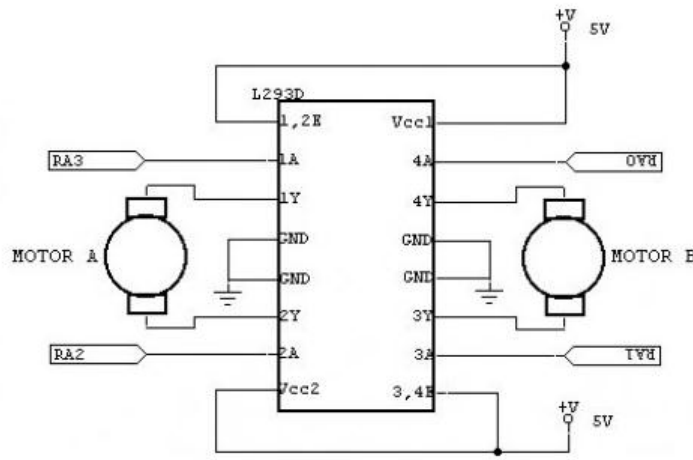


Fuente: L293D Dathasheet.

Una conexión básica para el control de dos motores es presentada en la figura 26, en esta imagen, tanto el integrado como los motores son alimentados a 5 voltios. Por los pines 1A, 2A ,3A y 4A se conectará una señal digital de alta o de baja según se desee controlar el sentido de giro de cada motor.

En la figura 26 se parecían las conexiones típicas para este tipo de integrado.

Figura 26. Conexión típica para control de dos motores.



Fuente: L293D Dathasheet.

De esta forma es posible controlar la dirección de cada motor según la siguiente tabla de verdad:

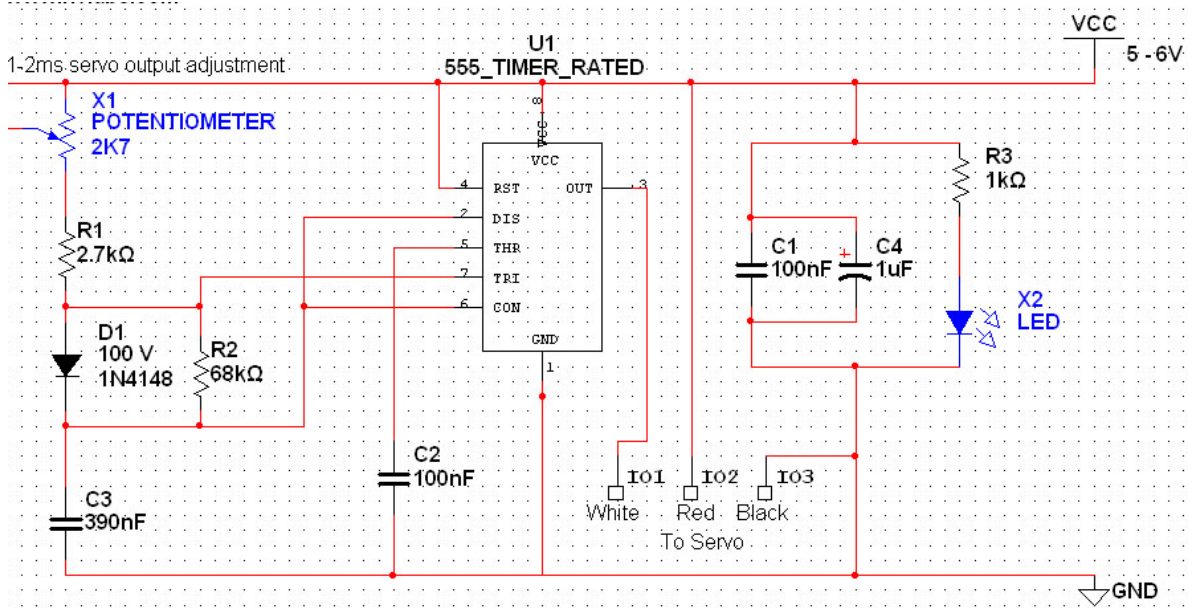
EN	1A	2A	FUNCION
H	L	H	GIRA DERECHA
H	H	L	GIRA IZQUIERDA
H	L	L	DETIENE MOTOR
H	H	H	DETIENE MOTOR
L	X	X	DETIENE MOTOR

H= HIGH
L= LOW
X= INDEPENDIENTE

Para controlar la velocidad a la que gira el motor, es necesario utilizar una señal PWM que nos dará un voltaje neto dependiendo de la duración de los pulsos. Esta señal PWM puede ser adquirida por un circuito usando un Timer como el 555 o puede obtenerse de una tarjeta como la Arduino.

Un circuito basado en el timer 555 el cual genera pulsos los cuales son ajustados variando un potenciómetro puede observarse en la figura 27.

**Figura 27. Generador de pulsos basado en el 555 IC**



Fuente: HUNT, Oliver. 555 Based Simple Servo Controller [Online], Enero 2006 [Cited 16 April 2013]. Available from internet: <http://www.hvllabs.com/555servo.html>

En la figura 27, la salida es usada para controlar un servo motor (cable blanco).

Esta señal de PWM debe conectarse al pin Enable del L293D y dependiendo de estos pulsos, el motor girara a mayor o menor velocidad dependiendo de la duración de los pulsos.

En muchas ocasiones resulta más sencillo obtener la señal de PWM por medio de arduino utilizando la función analogWrite (#) donde # indicara la duración de los pulsos donde su máximo valor será 255.

Una de las desventajas de controlar un micromotor como el que se mencionó anteriormente por medio de arquitectura puente H es la cantidad de pines

necesarios, lo cual en aplicaciones donde se requieren muchos motores, la tarjeta arduino uno no alcanza a proporcionar toda la cantidad de pines analógicos y digitales.

Una solución a este problema se encuentra en la tarjeta arduino mega, la cual posee mayor cantidad de pines analógicos y digitales permitiendo el control de una mayor cantidad de motores.

**4.3.1 Aspectos a tener en cuenta con los integrados con arquitectura puente H.**A la hora de utilizar un circuito integrado con arquitectura de puente H se debe tener en cuenta ciertos aspectos que son propios de estos integrados y que afectan directamente el comportamiento de la aplicación que se esté desarrollando y de los actuadores que se estén controlando.

Para empezar se debe saber que estos circuitos integrados se basan en dispositivos electrónicos llamados transistores los cuales son usados normalmente como switches aunque también pueden ser interruptores proporcionales debido a que son dispositivos de estado sólido (amplificadores).

Estos dispositivos tienen ciertas características que serán explicadas a continuación.

**4.3.1.1 Transistores.**Los transistores son usados para permitir el paso de una corriente o una parte de esta a través de él mediante el accionamiento por medio de una corriente mucho menor. De esta manera se puede diferenciar entre la corriente de control y la corriente de potencia.

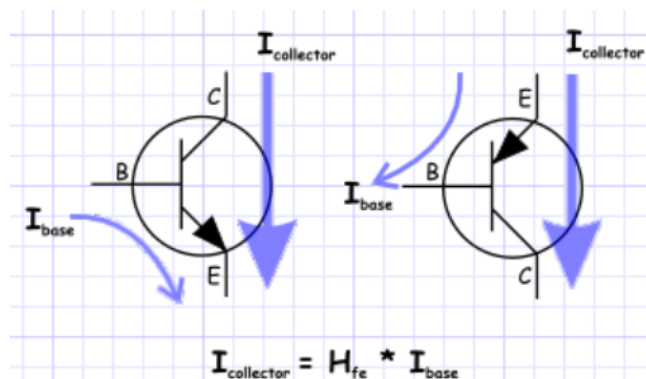
La propiedad que define a los transistores es llamada ganancia, a la salida de un transistor se puede obtener una salida igual a la entrada multiplicada por el factor de ganancia.

- **Transistor de unión bipolar.** Este tipo de transistores se basa en el contacto entre dos semiconductores para su operación y se pueden diferenciar tres regiones: Emisor, colector y base, dependiendo de la forma en la cual la corriente fluye con respecto a la base se pueden diferenciar dos tipos de transistores de unión bipolar.

En los transistores NPN la corriente fluye hacia la base lo que resulta en un flujo de corriente más grande entre el colector y el emisor. Por otro lado, en un transistor PNP la corriente fluye desde la base permitiendo un flujo de corriente entre el emisor y el colector

En la figura 28 se puede observar el diagrama de cada uno de los tipos de transistores NPN y PNP con sus respectivos flujos de corriente.

**Figura 28. Diagramas de NPN y PNP**



*C = Colector, B = Base y E = Emisor*

Fuente: MCMANIS, Chuck. Bipolar Junction (BJT) H-Bridges [Online], Noviembre 2003 [Cited 16 April 2013]. Available from internet:

<http://www.mcmanis.com/chuck/robotics/tutorial/h-bridge/bjt-bridge.html>

Como se puede observar, la principal diferencia entre un NPN y un PNP es la dirección del flujo de la corriente de potencia, por lo que mientras que en uno es

positiva en el otro será negativa, esta relación de polaridad se vuelve importante a la hora de construir un puente H.

Una de las cosas que se debe tener en cuenta es que a través del transistor solo puede fluir una cantidad de corriente que está determinada por el voltaje de la fuente de poder y la resistencia que está conectada en serie con el transistor, independientemente de la ganancia y de la corriente a la que fue activado el transistor.

Esto quiere decir que si la multiplicación entre la corriente de accionamiento y la ganancia del transistor, es mayor que la corriente máxima que puede circular a través del circuito (calculada mediante la ley de ohm) se dice que el transistor está saturado y este funcionará como un switch que estará completamente abierto permitiendo el paso de la totalidad de la corriente del circuito.

Sin lugar a dudas el factor que más se debe tener en cuenta en los transistores y por ende en los integrados con arquitectura de puente H es el hecho de que a través de un transistor siempre habrá una caída de voltaje que será propia del transistor, esto es debido a que el transistor no es un conductor perfecto. Esta caída de voltaje se da incluso cuando el transistor se encuentra saturado y es presentada en los datasheets como  $V_{ce}(sat)$  y  $V_{be}$  dependiendo de dónde se quiere saber la caída de voltaje.

Esto es de gran importancia ya que al momento de realizar una aplicación en la que los actuadores (motores eléctricos DC) deben ser alimentados a un voltaje específico según los requerimientos de velocidad y de carga, se debe tener en cuenta esta caída de voltaje la cual es típicamente 0.2 V a 1V y en ocasiones mayor dependiendo del transistor.

Esta caída de voltaje es la misma causa de la pérdida de potencia en calor debido a que al perderse una cantidad de voltaje a una corriente determinada se obtiene una potencia de pérdida dada por la ecuación  $P=I*V$  donde P es la potencia e I y V son la corriente y el voltaje respectivamente.

#### **4.4 MOTORES PASO A PASO**

Los motores paso a paso se pueden ver como motores eléctricos sin escobillas. Es típico que todos los bobinados del motor sean parte del estator, y el rotor puede ser un imán permanente o, en el caso de los motores de reluctancia variable, un cilindro sólido con un mecanizado en forma de dientes (similar a un engranaje), construido con un material magnéticamente "blando" (como el hierro dulce).

Los motores paso a paso son ideales para la construcción de mecanismos en donde se requieren movimientos muy precisos.

La característica principal de estos motores es el hecho de poder moverlos un paso a la vez por cada pulso que se le aplique. Este paso puede variar desde  $90^\circ$  hasta pequeños movimientos de tan solo  $1.8^\circ$ , es decir, que se necesitarán 4 pasos en el primer caso ( $90^\circ$ ) y 200 para el segundo caso ( $1.8^\circ$ ), para completar un giro completo de  $360^\circ$ .

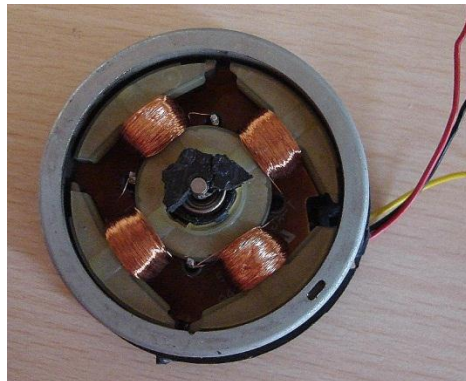
Básicamente estos motores están constituidos normalmente por un rotor sobre el que van aplicados distintos imanes permanentes y por un cierto número de bobinas excitadoras bobinadas en su estator. Las bobinas son parte del estator y el rotor es un imán permanente. Toda la conmutación (o excitación de las bobinas) debe ser externamente manejada por un controlador.

Típicamente, los motores y sus controladores se diseñan de manera que el motor se pueda mantener en una posición fija y también para que se lo pueda hacer girar en un sentido y en el otro.

La mayoría de los motores paso a paso conocidos se pueden hacer avanzar a frecuencias de audio, lo que les permite girar muy velozmente. Con un controlador apropiado, se los puede hacer arrancar y detenerse en un instante en posiciones controladas.

En la figura 29 puede observarse el rotor y el estator de un motor paso a paso.

**Figura 29. Rotor y estator de un motor Paso a Paso.**



Alma de hierro de motor Paso a Paso



Rotor motor Paso a Paso

Fuente: Alma de hierro. Motor paso a paso [Online], Septiembre 2012 [Cited 16 April 2013]. Available from internet:

<http://almadeherrero.blogspot.com/2012/09/motor-paso-paso.html>

Los motores paso a paso tienen una característica adicional: el torque de detención (que se puede ver mencionado también como "par de detención", e incluso par/torque "de mantenimiento"), que no existe en los motores de CC. El torque de detención hace que un motor paso a paso se mantenga firmemente en

su posición cuando no está girando. Esta característica es muy útil cuando el motor deja de moverse y, mientras está detenido, la fuerza de carga permanece aplicada a su eje. Se elimina así la necesidad de un mecanismo de freno.

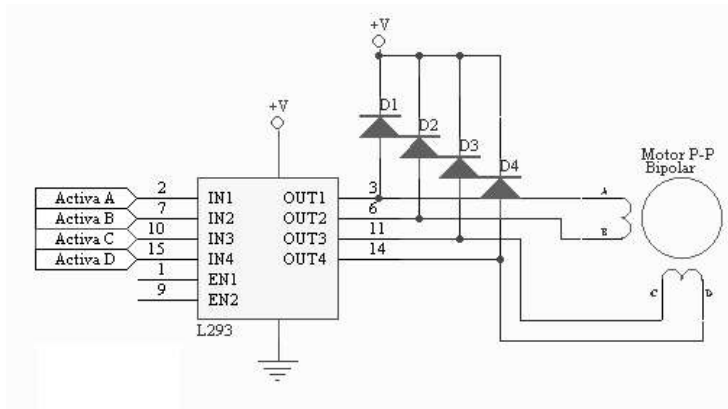
Los motores paso a paso más comunes son los de imán permanente, y estos los hay de dos tipos:

- Bipolar
- Unipolar

**4.4.1 Bipolar.** Estos tienen generalmente cuatro cables de salida. Necesitan ciertos trucos para ser controlados, debido a que requieren del cambio de dirección del flujo de corriente a través de las bobinas en la secuencia apropiada para realizar un movimiento. Para esto es necesario un H-Bridge por cada bobina del motor, es decir que para controlar un motor Paso a Paso de 4 cables (dos bobinas), se necesita usar dos H-Bridges iguales. En general es recomendable el uso de H-Bridge integrados como son los casos del L293B.

A continuación se muestra en la figura 30 un diagrama ilustrativo de la conexión de este tipo de motores.

Figura 30. Esquema de conexión usando un integrado L293.



Fuente: Docentes. Motores paso a paso [Online], [Cited 17 April 2013]. Available from internet:

<http://www.docentes.unal.edu.co/hfvelascop/docs/CLASES/DIGITALES2/LABORATORIO/Motor Paso aPaso.pdf>

La secuencia necesaria para controlar motores paso a paso bipolares se representa en la Figura 31.

Figura 31. Secuencia de control motor bipolar

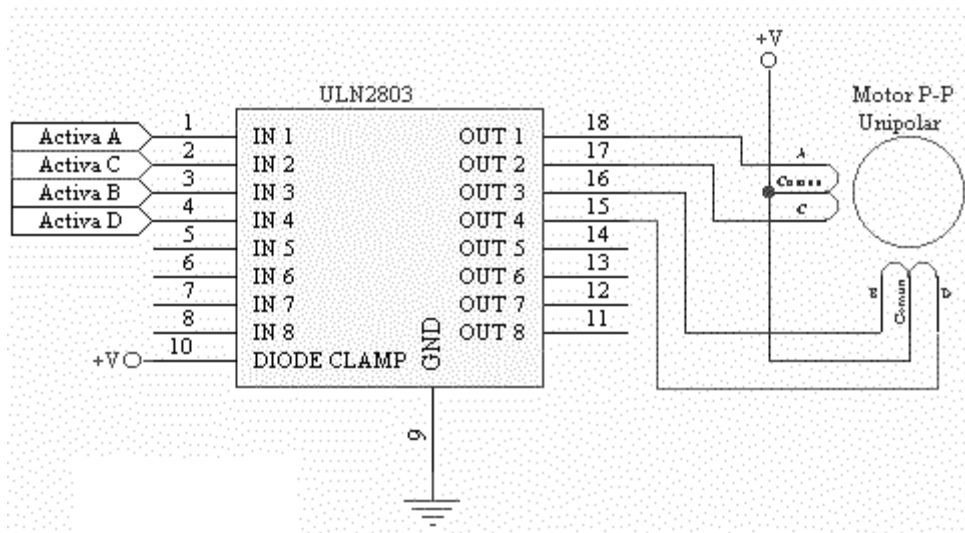
PASO	TERMINALES			
	A	B	C	D
1	+V	-V	+V	-V
2	+V	-V	-V	+V
3	-V	+V	-V	+V
4	-V	+V	+V	-V

Fuente: Docentes. Motores paso a paso [Online], [Cited 17 April 2013]. Available from internet:

<http://www.docentes.unal.edu.co/hfvelascop/docs/CLASES/DIGITALES2/LABORATORIO/Motor Paso aPaso.pdf>

**4.4.2 Unipolar.** Estos motores suelen tener 6 o 5 cables de salida, dependiendo de su conexión interna. Este tipo se caracteriza por ser más simple de controlar. En la figura 32 se puede apreciar un ejemplo de conexión para controlar un motor paso a paso unipolar mediante el uso de un ULN2803, el cual es un arreglo de 8 transistores tipo Darlington capaces de manejar cargas de hasta 500mA. Las entradas de activación (Activa A, B, C y D) pueden ser directamente activadas por un microcontrolador.

**Figura 32. Conexión motor paso a paso unipolar usando ULN2803.**



Fuente: Docentes. Motores paso a paso [Online], [Cited 17 April 2013]. Available from internet:

<http://www.docentes.unal.edu.co/hfvelascop/docs/CLASES/DIGITALES2/LABORATORIO/Motor Paso aPaso.pdf>

**4.4.3 Secuencias de movimiento de los motores paso a paso.** Los motores unipolares pueden ser movidos usando varias secuencias.

**4.4.3.1 Secuencia Normal.** Esta es la secuencia más usada y la que generalmente recomienda el fabricante. Con esta secuencia el motor avanza un paso por vez y

debido a que siempre hay al menos dos bobinas activadas, se obtiene un alto torque de paso y de retención.

En la tabla 2 se puede observar la secuencia que se debe seguir en la conmutación de las bobinas para hacer girar el motor en una secuencia normal de movimiento.

**Tabla 2. Secuencia Normal motor paso a paso unipolar**

Paso	Bobina A	Bobina B	Bobina C	Bobina D	Ilustracion
1	ON	ON	OFF	OFF	
2	OFF	ON	ON	OFF	
3	OFF	OFF	ON	ON	
4	ON	OFF	OFF	ON	

Fuente: Docentes. Motores paso a paso [Online], [Cited 17 April 2013]. Available from internet:

[http://www.docentes.unal.edu.co/hfvelascop/docs/CLASES/DIGITALES2/LABORATORIO/Motor Paso aPaso.pdf](http://www.docentes.unal.edu.co/hfvelascop/docs/CLASES/DIGITALES2/LABORATORIO/Motor%20Paso%20aPaso.pdf)

**4.4.3.2 Secuencia ondular.** En esta secuencia se activa solo una bobina a la vez.

En algunos motores esto brinda un funcionamiento más suave. Pero al estar solo un bobina activada, el torque de paso y retención es menor.

En la tabla 3 se puede observar las secuencias de conmutación para el tipo ondular.

**Tabla 3. Secuencias Tipo ondular**

Paso	Bobina A	Bobina B	Bobina C	Bobina D	Ilustracion
1	on	off	off	off	
2	off	on	off	off	
3	off	off	on	off	
4	off	off	off	on	

Fuente: Docentes. Motores paso a paso [Online], [Cited 17 April 2013]. Available from internet:

<http://www.docentes.unal.edu.co/hfvelascop/docs/CLASES/DIGITALES2/LABORATORIO/Motor Paso aPaso.pdf>.

**4.4.3.3 Secuencia tipo medio paso.** En esta secuencia se activan las bobinas de tal formade brindar un movimiento igual a la mitad del paso real. Para ello se activan primero 2bobinas y luego solo 1 y así sucesivamente. Como se puede ver en la tabla4 la secuenciacompleta consta de 8 movimientos en lugar de 4.

**Tabla 4. Secuencia tipo medio paso**

Paso	Bobina A	Bobina B	Bobina C	Bobina D	Ilustracion
1	on	off	off	off	
2	on	on	off	off	
3	off	on	off	off	
4	off	on	on	off	
5	off	off	on	off	
6	off	off	on	on	
7	off	off	off	on	
8	on	off	off	on	

Fuente: Docentes. Motores paso a paso [Online], [Cited 17 April 2013]. Available from internet:

[http://www.docentes.unal.edu.co/hfvelascop/docs/CLASES/DIGITALES2/LABORATORIO/Motor Paso aPaso.pdf](http://www.docentes.unal.edu.co/hfvelascop/docs/CLASES/DIGITALES2/LABORATORIO/Motor%20Paso%20aPaso.pdf)

## 5. SENSORES EN ROBOTICA MOVIL

En las aplicaciones de robótica móvil, es necesario contar con sensores que adquieran los datos y las señales que se obtienen del medio de trabajo con el fin de realizar las funciones para las que se ha creado el robot.

Estos sensores deben ser pequeños y de bajo peso para que no interfieran con la movilidad del robot en desarrollo.

Los sensores se pueden clasificar de muchas maneras, pero una de las más comunes para robots móviles es:

- **SENSORES PROPRIOCEPTIVOS:** Se refiere a sensores capaces de detectar el estado interno del robot como por ejemplo la carga de las baterías, la posición del robot, corriente consumida entre otros.
- **SENSORES EXTEROCEPTIVOS:** Se refiere a sensores capaces de detectar el estado externo del robot (entorno) como por ejemplo la temperatura, la presión, localización de obstáculos, cercanía de objetos entre otros.

### 5.1 PROPIEDADES A TENER EN CUENTA EN LA SELECCIÓN DE SENSORES

**5.1.1 Velocidad de operación.** Es la velocidad con la que un sensor puede actualizar la información que está percibiendo del entorno, algunos sensores son capaces de trabajar en tiempo real mientras que otros son usados para puntos o intervalos específicos de tiempo.

**5.1.2 Costo.** Es uno de los principales factores ya que en la mayoría de casos el costo de los sensores puede limitar el diseño de nuestro robot.

**5.1.3 Robustez.** Esta propiedad se refiere a la capacidad del sensor a no variar sus medidas de forma irregular con el cambio del medio de funcionamiento.

**5.1.4 Peso y tamaño.** Estos factores influyen directamente en la movilidad y autonomía del robot.

## **5.2 SENSORES DE POSICIÓN**

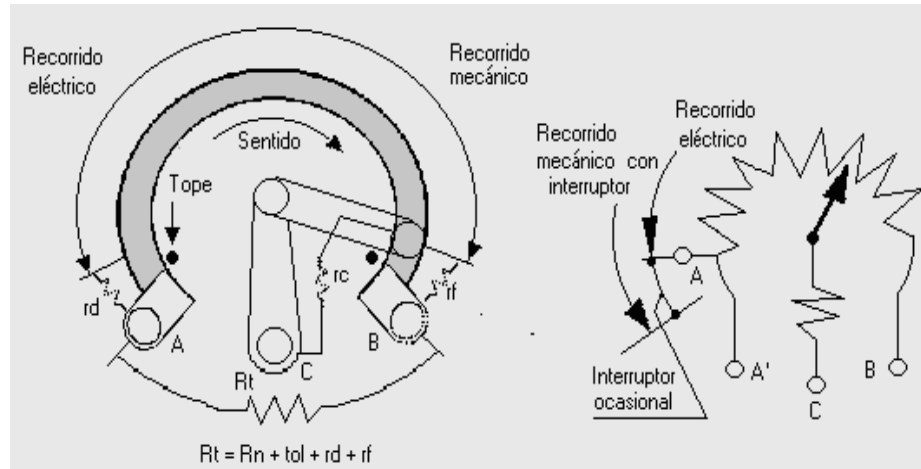
Los dispositivos que cambian alguna propiedad eléctrica con el cambio de posición pueden ser usados como sensores de posición, un ejemplo de ello son los potenciómetros, los cuales varían la resistencia interna de acuerdo con la posición angular del eje que estos poseen.

**5.2.1 Potenciómetro.** Son dispositivos capaces de variar su resistencia mediante el giro de un eje el cual está conectado a un material resistivo. El eje mueve un contacto que se desliza sobre el material resistivo que junto a uno de los apoyos fijos marca una resistencia determinada por la longitud del arco del material resistivo que hay entre estos dos apoyos. Los potenciómetros tienen ciertas características:

- **Recorrido mecánico:** Este es el recorrido que puede efectuar el eje del potenciómetro sin que su movimiento se vea interrumpido por algún tipo de tope mecánico.
- **Recorrido eléctrico:** Es la parte del deslizamiento que proporciona cambios en la resistencia medida. Suele coincidir con el recorrido mecánico.
- **Resistencia nominal ( $R_n$ ):** valor esperado de resistencia variable entre los límites del recorrido eléctrico.

- Resistencia residual de fin de pista ( $r_f$ ): resistencia comprendida entre el límite superior del recorrido eléctrico del cursor y el contacto B (ver figura 33).
- Resistencia residual de principio de pista ( $r_d$ ): valor de resistencia comprendida entre límite inferior del recorrido eléctrico y el contacto A (ver figura).
- Resistencia total ( $R_t$ ): resistencia entre los terminales fijos A o A' y B, sin tener en cuenta la conexión del cursor e incluyendo la tolerancia. Aunque a efectos prácticos se considera igual al valor nominal ( $R_t=R_n$ ).
- Resistencia de contacto ( $r_c$ ): resistencia que presenta el cursor entre su terminal de conexión externo y el punto de contacto interno (suele despreciarse, al igual que  $r_d$  y  $r_f$ ).
- Temperatura nominal de funcionamiento ( $T_n$ ): es la temperatura ambiente a la cual se define la disipación nominal.
- Temperatura máxima de funcionamiento ( $T_{max}$ ): máxima temperatura ambiente en la que puede ser utilizada la resistencia.
- Potencia nominal ( $P_n$ ): máxima potencia que puede disipar el dispositivo en servicio continuo y a la temperatura nominal de funcionamiento.
- Tensión máxima de funcionamiento ( $V_{max}$ ): máxima tensión continua ( o alterna eficaz) que se puede aplicar a la resistencia entre los terminales extremos en servicio continuo, a la temperatura nominal de funcionamiento.
- Resolución: cantidad mínima de resistencia que se puede obtener entre el cursor y un extremo al desplazar (o girar) el cursor. Suele expresarse en % en tensión, en resistencia, o resolución angular.

**Figura 33. Características de un potenciómetro.**



Fuente: Universidad Complutense de Madrid. Resistencias variables[Online], [Cited 17 April 2013]. Available from internet:

<http://pendientedemigracion.ucm.es/info/electron/laboratorio/componentes/codigos/resisva.htm>

Para poder sensor la posición angular con ayuda de los potenciómetros, es necesario acoplar el eje del potenciómetro con el componente que se desee controlar, de esta forma si el componente gira, también lo hará el eje del potenciómetro con lo que variara su resistencia.

El potenciómetro debe estar alimentado con una fuente de voltaje con el fin de sacar una señal de voltaje la cual será leída por un controlador para efectuar el proceso de sensado. Es necesario realizar una caracterización para poder determinar que voltaje representa que posición, esta labor es un poco más sencilla en el caso de utilizar potenciómetros lineales, los cuales varían su resistencia de forma lineal dependiendo del ángulo en el que se encuentren.

**Figura 34. Diferentes configuraciones de potenciómetros.**



Fuente: EA1AV. Componentes [Online], [Cited 17 April 2013]. Available from internet: <http://ea1av.ure.es/componentes%20antiguos/cambios.htm>

**5.2.2 Codificador rotatorio o encoder.** Estos codificadores son dispositivos electromecánicos utilizados para convertir una posición angular en un código digital por lo que se pueden usar como transductores o sensores de posición angular. Estos codificadores los hay de varios tipos.

**5.2.2.1 Codificador rotatorio absoluto.** Estos poseen una hoja de metal puesta en un disco aislador que está fijado al eje, también tiene una fila de contactos deslizantes a lo largo del radio del disco los cuales generan un código binario a medida que el eje va girando. De esta manera para cada posición del eje existe un código binario diferente que permite conocer la posición absoluta del eje.

Algunos codificadores de este tipo pueden sacar hasta 16384 (14 bits) posiciones únicas por vuelta. Esto representa una resolución angular de  $0.022^\circ$ , después de una revolución completa, la codificación empieza nuevamente en la posición de inicio.

**Figura 35. Encoder Absoluto**

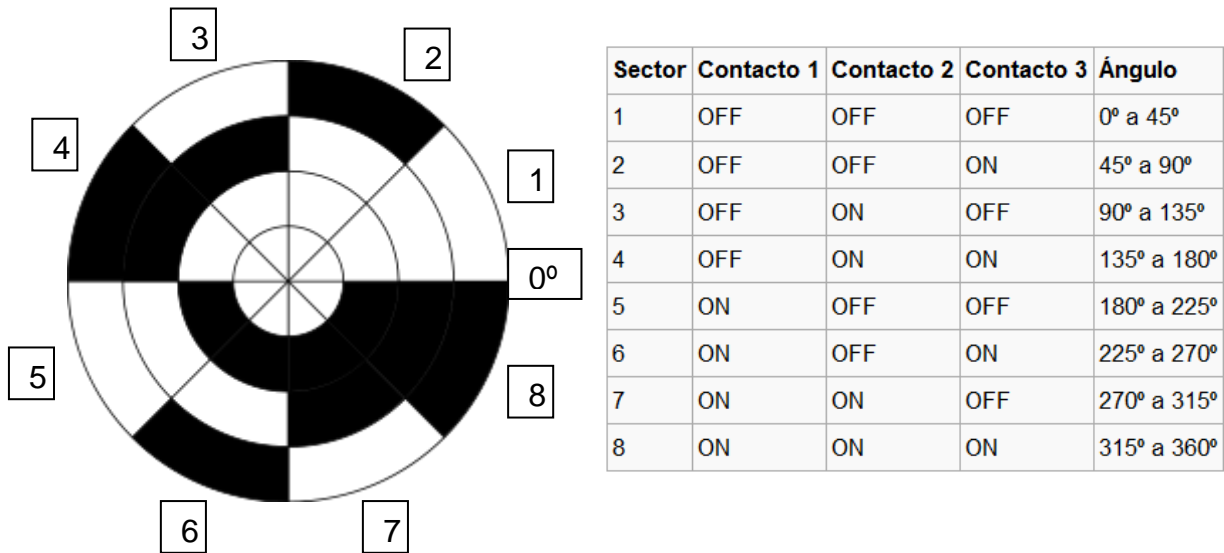


Fuente: SCANCON.Codificador rotativo absoluto antideflagrante [Online], [Cited April 17 2013]. Available from internet:  
<http://www.directindustry.es/prod/scancon/codificadores-rotativos-absolutos-antideflagrantes-23341-57171.html>

Las posiciones distintas del eje dependen de la cantidad de contactos que se tenga, así si hay  $n$  contactos, el número de posiciones diferentes será  $2^n$ .

De esta forma la resolución estará dada por el número de contactos. A continuación se mostrará en la figura 36 un codificador rotativo de 3 contactos lo que da 8 posiciones diferentes para el eje, con una resolución de 45 grados que equivale a dividir los 360 grados del giro completo entre el número de posiciones diferentes o sectores de la codificación.

**Figura 36. Codificador rotativo de tres contactos.**



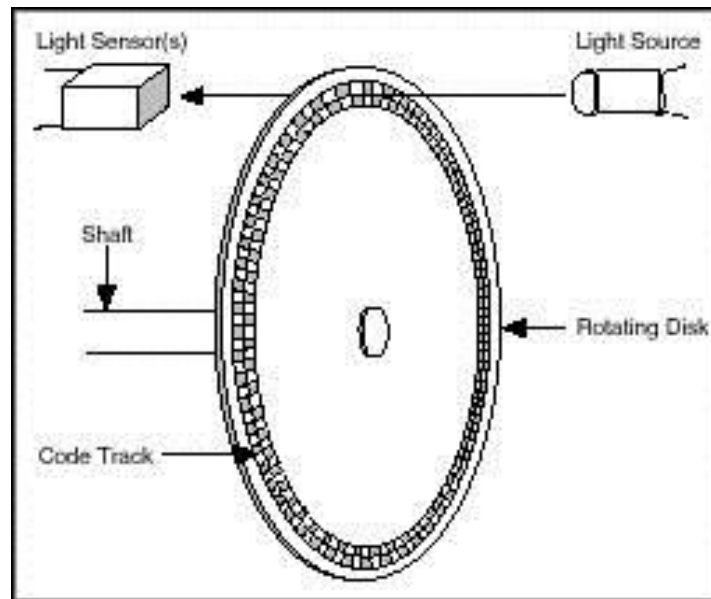
Fuente: jjbeard. A Rotary Encoder Disc with a 3-Bit Binary Reflected Gray Code (BRGC) [Online], Mayo 2006 [Cited April 18 2013]. Available from internet: [http://es.wikipedia.org/wiki/Codificador\\_rotatorio](http://es.wikipedia.org/wiki/Codificador_rotatorio)

**5.2.2.2 Codificador rotatorio relativo.** El codificador rotatorio relativo (también llamado codificador incremental) se utiliza cuando los métodos de codificación absolutos sean demasiado incómodos (debido al tamaño del disco modelado). Este método también utiliza un disco unido al eje, pero este disco es mucho más pequeño marcado con una gran cantidad de líneas de la parte radial como los radios de una rueda. El interruptor óptico, parecido a un fotodiodo, genera un pulso eléctrico cada vez que una de las líneas pase a través de su campo visual. Un circuito de control electrónico cuenta los pulsos para determinar el ángulo con el cual el eje da vuelta.

En la figura 37 se puede observar el principio básico de funcionamiento donde se tiene un interruptor fotoeléctrico y un disco ranurado el cual a medida que va

girando, va generando pulsos digitales los cuales son contados para que con el espaciamiento entre ranuras se pueda obtener los grados avanzados en el tiempo.

**Figura 37. Esquema de conjunto encoder incremental.**



Fuente: MCDUGALL, Archie. Incremental Encoders [Online], Enero 2008 [Cited 18 April 2013]. Available from internet: [http://www.pc-control.co.uk/incremental\\_encoders.htm](http://www.pc-control.co.uk/incremental_encoders.htm)

A continuación se comparara cada uno de los sensores mencionados anteriormente con sus respectivas ventajas y desventajas para la aplicación de un robot escalador (Ver tabla 5).

**Tabla 5. Tabla comparativa de los sensores de posición angular.**

<b>SENSOR</b>	<b>VENTAJAS</b>	<b>DESVENTAJAS</b>
<b>POTENCIOMETRO</b>	Es fácilmente conseguido en el mercado y está en precios muy económicos, además de su variedad de tamaños en los que se puede encontrar algunos muy pequeños, son sensores análogos..	Con el uso pierden confiabilidad, la lectura es a veces inestable, no permiten sensado para aplicaciones de giro continuo y la temperatura afecta de manera significativa su lectura.
<b>ENCODER INCREMENTAL</b>	Bajo costo, permite aplicaciones de giro continuo	Se debe hacer un tratamiento complejo de las señales adquiridas para determinar la posición angular correctamente.
<b>ENCODER ABSOLUTO</b>	Alta confiabilidad en el tiempo de la posición actual, alta robustez	Alto precio, elevados tamaños, alto peso

Fuente: Autores.

**5.2.3 IMU o unidad inercial de medición.** En aplicaciones de robótica móvil es de vital importancia conocer la orientación, velocidad y fuerzas gravitacionales que están presentes en el robot con el fin de obtener información del estado del cuerpo rígido.

Para ello se utiliza una combinación de acelerómetros y giróscopos que se encargan de medir la aceleración con respecto a la gravedad en cada uno de sus ejes (acelerómetro) y de medir la posición angular (giróscopo).

Aunque estas unidades de medición trabajan de forma similar a como los seres humanos detectan su posición con respecto a una referencia, estas poseen un error que se va acumulando a medida que se van efectuando las mediciones.

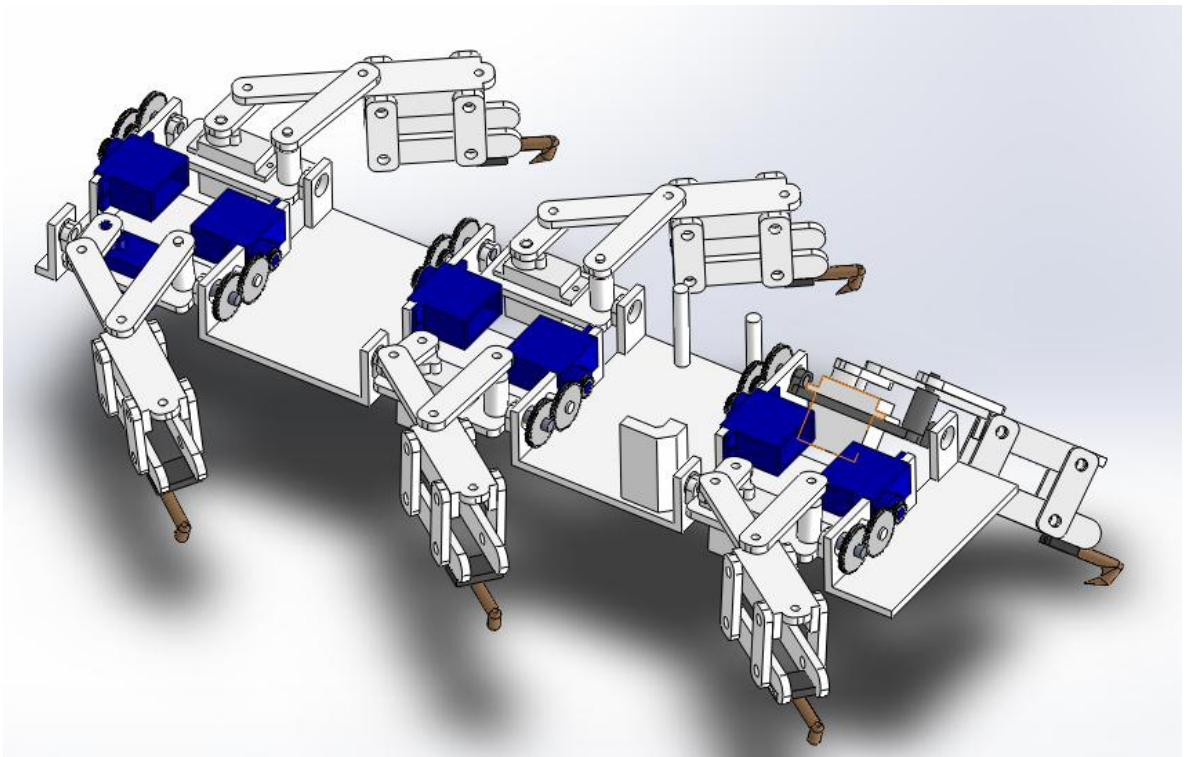
## 6. PREPROTOTIPO

### 6.1 DISEÑO MECÁNICO DEL PRE PROTOTIPO DEL ROBOT ESCALADOR

El diseño de un pre prototipo del diseño final del robot escalador es necesario para saber a grandes rasgos los inconvenientes que se pueden presentar en el proceso de manufactura y en la acción de escalar del prototipo final.

En la figura 38 se muestra el resultado del diseño del preprototipo con el cual se realizaran las observaciones pertinentes para desarrollar un mejor diseño del prototipo final.

**Figura 38. Diseño CAD del preprototipo Solidworks.**



Fuente: Autores.

A continuación se presentará como se realizó el diseño de este preprototipo y que consideraciones se tuvieron en cuenta para el diseño y su respectiva construcción.

El diseño mecánico del robot escalador comprende varios sistemas que se plantearon a la par en un modo de diseño concurrente, de esta forma se logró tener en cuenta tanto los procesos de manufactura, los dispositivos y componentes disponibles en el mercado y la interacción que debía suceder entre estos sistemas.

Como prototipo inicial, se pretendió diseñar el sistema mecánico del robot con materiales y dispositivos accesibles y asequibles para evitar gastos elevados, pero a la vez aproximando este diseño al objetivo final que es el proceso de escalada en vertical sobre superficies de palma y así obtener información de interés para involucrarla en un diseño final de mayor robustez.

El diseño mecánico del robot se dividió en los siguientes sistemas:

1. Diseño de la pata
2. Diseño del pie o mecanismo de apoyo
3. Diseño de sistema hombro
4. Selección de actuadores
5. Diseño de estructura de soporte o cuerpo del robot

### **6.1.1 DISEÑO MECÁNICO DE LA PATA DEL ROBOT ESCALADOR**

Las patas del robot son las encargadas principales de realizar el movimiento de avance y posicionamiento del sistema, a su vez son las encargadas de transmitir la fuerza de agarre dada por el sistema hombro.

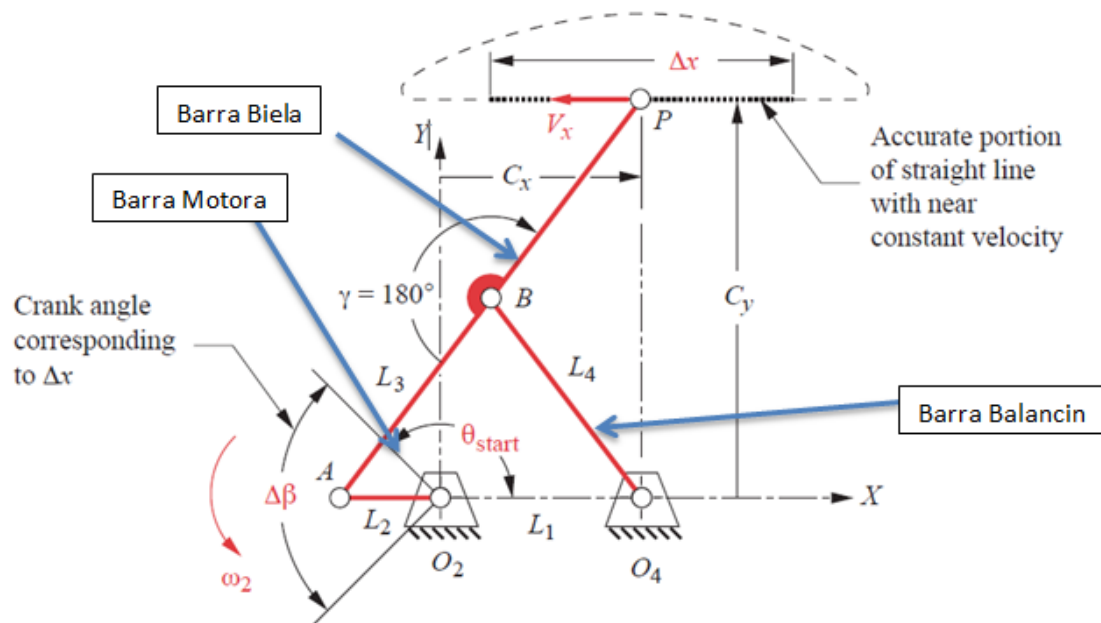
El diseño del sistema pata fue dirigido a obtener un mecanismo que pueda generar en un punto extremo, un movimiento de línea recta o semirrecta con el fin

de brindar la carrera de avance del sistema. A diferencia de un sistema articulado del tipo SCARA, el mecanismo de barras requiere menos sistemas de actuación, con la consecuente disminución del peso total de robot.

Después de un estudio de diferentes mecanismos de barras, se opta por diseñar el sistema pata mediante un mecanismo de barras de Hoekens.

Este es un mecanismo de 4 barras articulado, el cual maneja unas proporciones entre tamaño de barras determinado para generar una trayectoria aproximadamente recta en el punto final de la barra biela, a continuación, en la Figura 39 se muestra la configuración general de este tipo de mecanismo:

**Figura 39. Mecanismo Barras Hoekens**



Fuente: NORTON, Robert. Design of machinery. 2 ed. Boston: McGraw-Hill, 1999. P 119-121.

Con este mecanismo, obtenemos una conversión del movimiento rotacional del motor, en un movimiento de línea recta con velocidad aproximadamente constante, ideal para el avance del sistema sin aparición de aceleraciones y el consecuente desequilibrio inercial que puede producirse en el robot.

La tabla 6 muestra las relaciones de barras para diferentes rangos de ángulos de la manivela que generan la porción de línea recta, este rango es llamado  $\Delta\beta$ .

**Tabla 6. Diferentes relaciones de barras en mecanismo Hoekens, para optimización de velocidad y de linealidad del punto de movimiento de interés.**

Range of Motion			Optimized for Straightness						Optimized for Constant Velocity					
$\Delta\beta$ (deg)	$\theta_{start}$ (deg)	% of cycle	Maximum $\Delta C_y$ %	$\Delta V$ %	$\frac{V_x}{(L_2 \omega_2)}$	Link Ratios			Maximum $\Delta V_x$ %	$\Delta C_y$ %	$\frac{V_x}{(L_2 \omega_2)}$	Link Ratios		
						$L_1 / L_2$	$L_3 / L_2$	$\Delta x / L_2$				$L_1 / L_2$	$L_3 / L_2$	$\Delta x / L_2$
20	170	5.6%	0.00001%	0.38%	1.725	2.975	3.963	0.601	0.006%	0.137%	1.374	2.075	2.613	0.480
40	160	11.1%	0.00004%	1.53%	1.717	2.950	3.925	1.193	0.038%	0.274%	1.361	2.050	2.575	0.950
60	150	16.7%	0.00027%	3.48%	1.702	2.900	3.850	1.763	0.106%	0.387%	1.347	2.025	2.538	1.411
80	140	22.2%	0.001%	6.27%	1.679	2.825	3.738	2.299	0.340%	0.503%	1.319	1.975	2.463	1.845
100	130	27.8%	0.004%	9.90%	1.646	2.725	3.588	2.790	0.910%	0.640%	1.275	1.900	2.350	2.237
120	120	33.3%	0.010%	14.68%	1.611	2.625	3.438	3.238	1.885%	0.752%	1.229	1.825	2.238	2.600
140	110	38.9%	0.023%	20.48%	1.565	2.500	3.250	3.623	3.327%	0.888%	1.178	1.750	2.125	2.932
160	100	44.4%	0.047%	27.15%	1.504	2.350	3.025	3.933	5.878%	1.067%	1.124	1.675	2.013	3.232
180	90	50.0%	0.096%	35.31%	1.436	2.200	2.800	4.181	9.299%	1.446%	1.045	1.575	1.863	3.456

Fuente: NORTON, Robert. Design of machinery. 2 ed. Boston: McGraw-Hill, 1999. P 119-121.

Como podemos apreciar, una optimización de la linealidad y de velocidad constante al mismo tiempo no puede ser obtenida con la configuración del tamaño de las barras.

Para nuestro sistema, es necesario obtener una trayectoria recta lo más larga posible para brindar al robot una carrera de avance grande, a la vez, es

conveniente una velocidad aproximadamente constante en el movimiento para evitar aceleraciones excesivas y por lo tanto movimientos inerciales.

Ya que la velocidad puede ser controlada también con la velocidad de giro del motor, elegiremos la relación del tamaño de las barras optimizado para linealidad y también para una longitud de carrera máxima.

Como criterio principal, se diseñará por carrera lineal máxima, analizando el último renglón de la tabla de la columna optimizado para linealidad, se obtiene para un tamaño de barra motora dado, una longitud de carrera lineal alta:

Range of Motion			Optimized for Straightness						Optimized for Constant Velocity							
$\Delta\beta$ (deg)	$\theta_{start}$ (deg)	% of cycle	Maximum $\Delta C_y$ %	$\Delta V$ %	$\frac{V_x}{(L_2 \omega_2)}$	Link Ratios				Maximum $\Delta V_x$ %	$\Delta C_y$ %	$\frac{V_x}{(L_2 \omega_2)}$	Link Ratios			
						$L_1 / L_2$	$L_3 / L_2$	$\Delta x / L_2$				$L_1 / L_2$	$L_3 / L_2$	$\Delta x / L_2$		
20	170	5.6%	0.00001%	0.38%	1.725	2.975	3.963	0.601	0.006%	0.137%	1.374	2.075	2.613	0.480		
40	160	11.1%	0.00004%	1.53%	1.717	2.950	3.925	1.193	0.038%	0.274%	1.361	2.050	2.575	0.950		
60	150	16.7%	0.00027%	3.48%	1.702	2.900	3.850	1.763	0.106%	0.387%	1.347	2.025	2.538	1.411		
80	140	22.2%	0.001%	6.27%	1.679	2.825	3.738	2.299	0.340%	0.503%	1.319	1.975	2.463	1.845		
100	130	27.8%	0.004%	9.90%	1.646	2.725	3.588	2.790	0.910%	0.640%	1.275	1.900	2.350	2.237		
120	120	33.3%	0.010%	14.68%	1.611	2.625	3.438	3.238	1.885%	0.752%	1.229	1.825	2.238	2.600		
140	110	38.9%	0.023%	20.48%	1.565	2.500	3.250	3.623	3.327%	0.888%	1.178	1.750	2.125	2.932		
160	100	44.4%	0.047%	27.15%	1.504	2.350	3.025	3.933	5.878%	1.067%	1.124	1.675	2.013	3.232		
180	90	50.0%	0.096%	35.31%	1.436	2.200	2.800	4.181	9.299%	1.446%	1.045	1.575	1.863	3.456		

A la vez, la desviación de la linealidad varia muy poco, 0.096%, lo cual podríamos tratarlo como aceptable teniendo en cuenta que el tamaño de las barras será elegido lo más compacto posible.

Teniendo en cuenta este criterio, se procede a definir el tamaño de la barra motriz, el cual se elige teniendo en cuenta el tamaño de los agujeros para pernos de unión y el mínimo tamaño que facilite su manufactura.

Se eligió una longitud de barra motriz de 15 mm (entre centro de agujeros de unión), la cual es lo suficientemente corta para garantizar un tamaño del robot adecuado, y a la vez su manufactura no se hace tan complicada.

Con este tamaño de barra motriz, y la tabla 6, calculamos las diferentes longitudes del mecanismo:

Basándonos en la Figura 39 y en la nomenclatura:

$$L_2 = O_2A; L_3 = AB; L_4 = O_4B; L_1 = O_2O_4; L_3 = L_4$$

Las longitudes en milímetros se presentan en la tabla 7.

**Tabla 7. Longitudes características mecanismo Hoekens elegido**

Barra motriz L2 [mm]	Barra Balancín L4 [mm]	Barra biela L3[mm]	Longitud apoyos L1 [mm]	Longitud lineal $\Delta x$ [mm]
15	42	42	33	62,72

Fuente: Autores

Teniendo en cuenta las relaciones de distancia vertical y horizontal entre el punto B y el punto O2 de la Figura 39, medidos con referencia al punto O2 se obtienen las siguientes ecuaciones que relacionan los ángulos del mecanismo:

Relación distancia vertical entre O2 y B

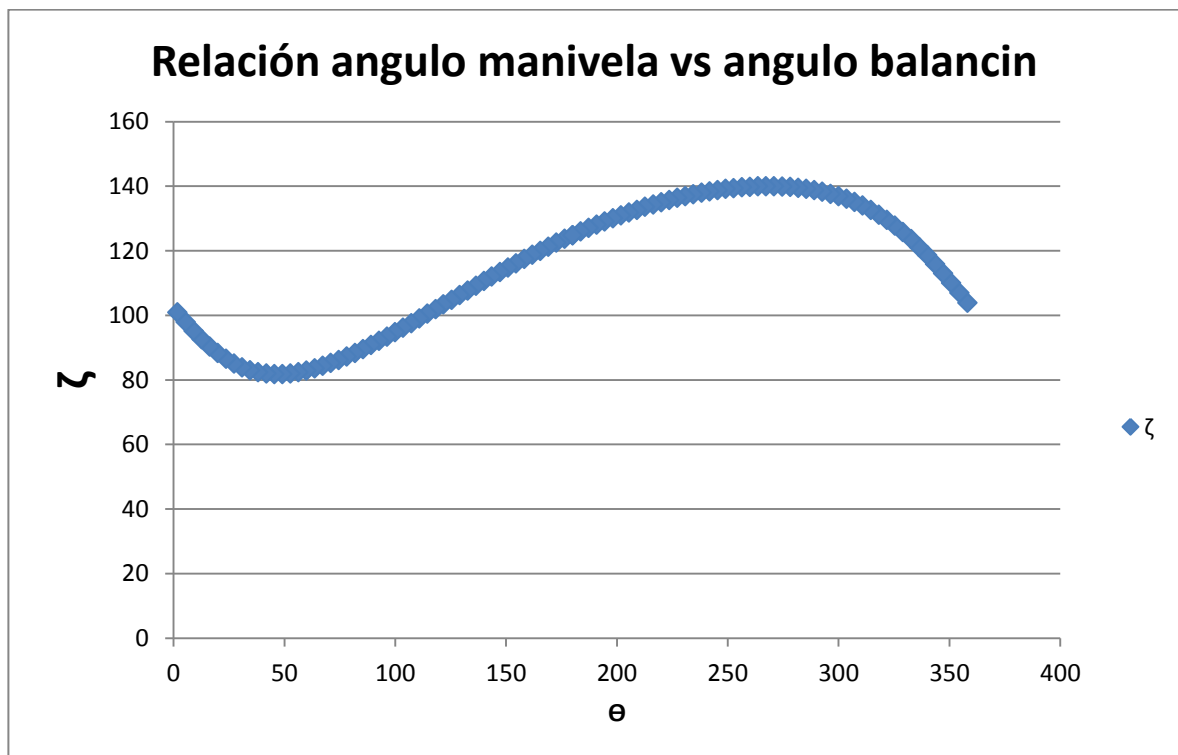
$$L_2 * \sin(180 - \theta) + L_3 * \sin(\alpha) = L_4 * \sin(180 - \gamma)$$

Relación distancia horizontal entre O2 y O4

$$L3 * \cos(\alpha) - L1 * \cos(180 - \theta) + L4 * \cos(180 - \gamma) = L1$$

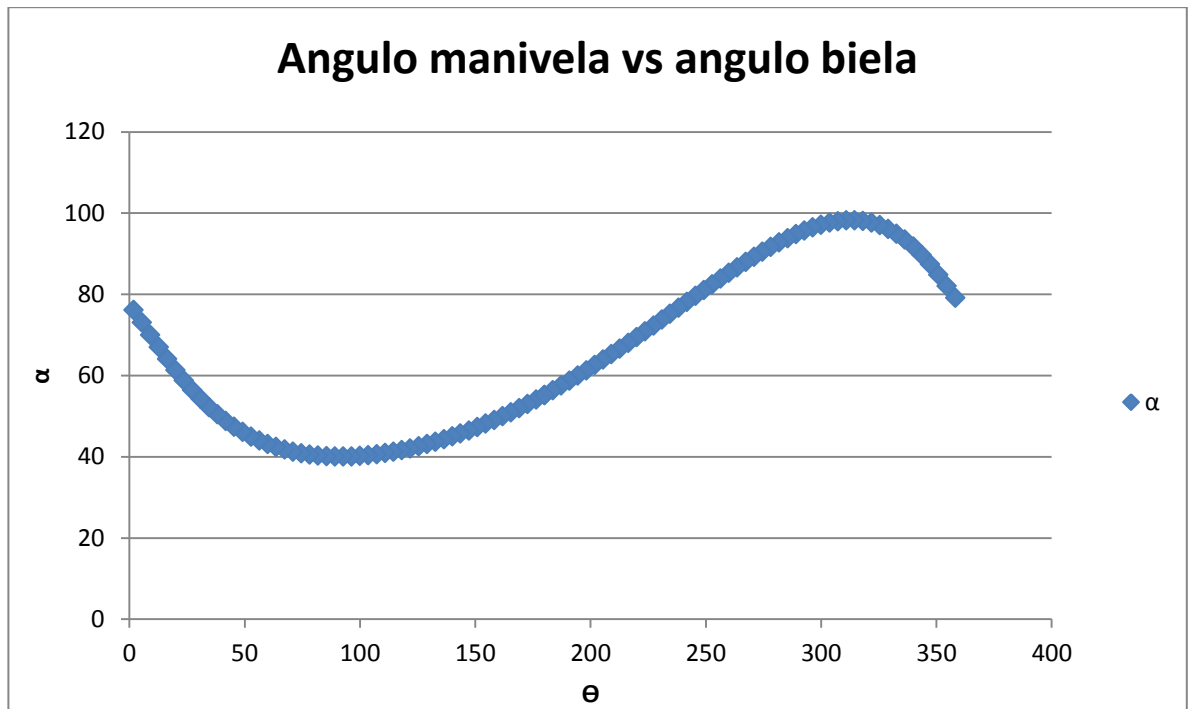
En las figuras 40 y 41 se relaciona el ángulo de cada barra con respecto a la horizontal para diferentes valores de ángulo de la barra motriz

**Figura 40. Angulo barra balancín con respecto a la horizontal en función del ángulo de la barra motora.**



Fuente: Autores

**Figura 41. Angulo barra biela con respecto a la horizontal en función del ángulo de la barra motora.**

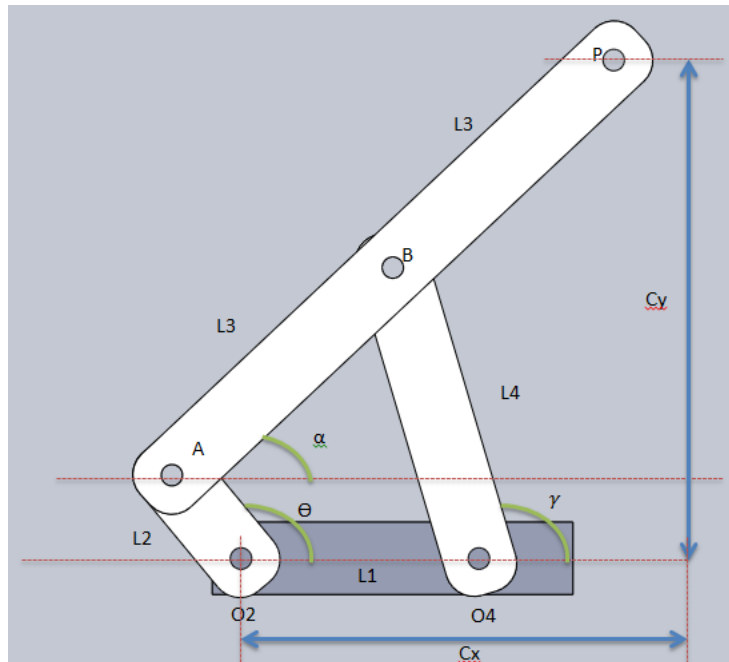


Fuente: Autores.

Es de importancia también, el conocimiento del comportamiento del punto de interés, punto P de la Figura 39, tanto su comportamiento al variar el ángulo de la manivela como también la trayectoria de este punto durante un ciclo de trabajo.

La figura 42 muestra la nomenclatura que se usará para referirse a las variables que determinan el mecanismo, tales como ángulo de inclinación y longitud de cada barra, posición del punto de interés P representada por  $C_x$  y  $C_y$ , entre otros.

**Figura 42. Nomenclatura usada mecanismo Hoekens.**



Fuente: autores.

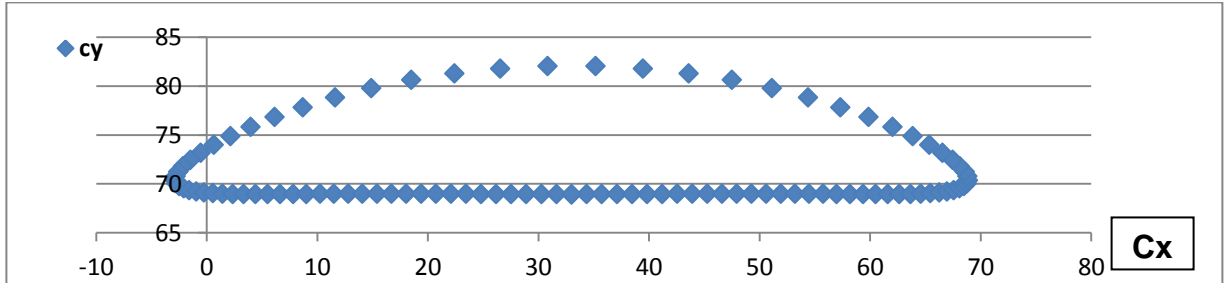
Teniendo en cuenta la figura 42 podemos relacionar la posición del punto P con respecto a O2 así:

$$Cx = L1 - L4 * \cos(180 - \gamma) + L3 * \cos(\alpha)$$

$$Cy = L2 * \sin(180 - \gamma) + 2 * L3 * \sin(\alpha)$$

La gráfica siguiente (figura 43) nos muestra el valor de la posición relativa de Cx y Cy para distintos ángulos de la manivela:

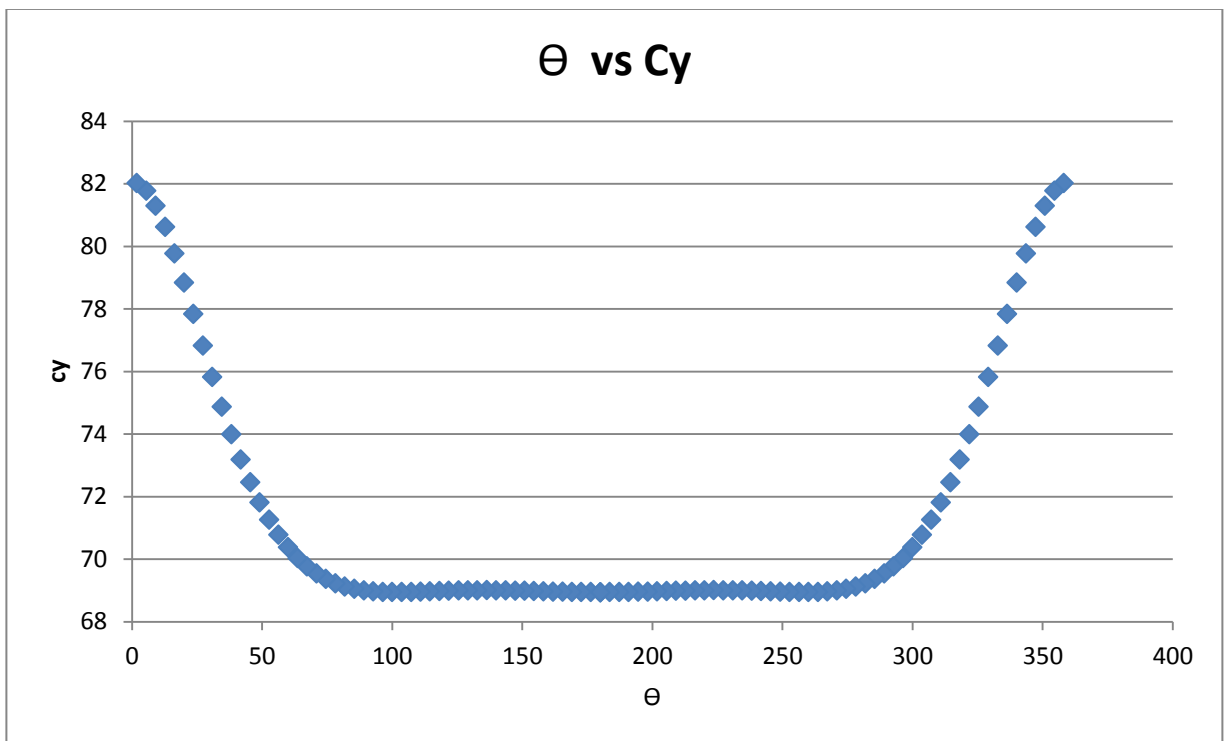
**Figura43. Posición x e y del punto de interés P, cada punto indica una posición de la manivela durante un ciclo.**



Fuente: Autores

La figura 44 muestra la posición en Y del punto de interés para un ciclo de la manivela:

**Figura44. Posición en y del punto de interés P vs Angulo de la manivela.**

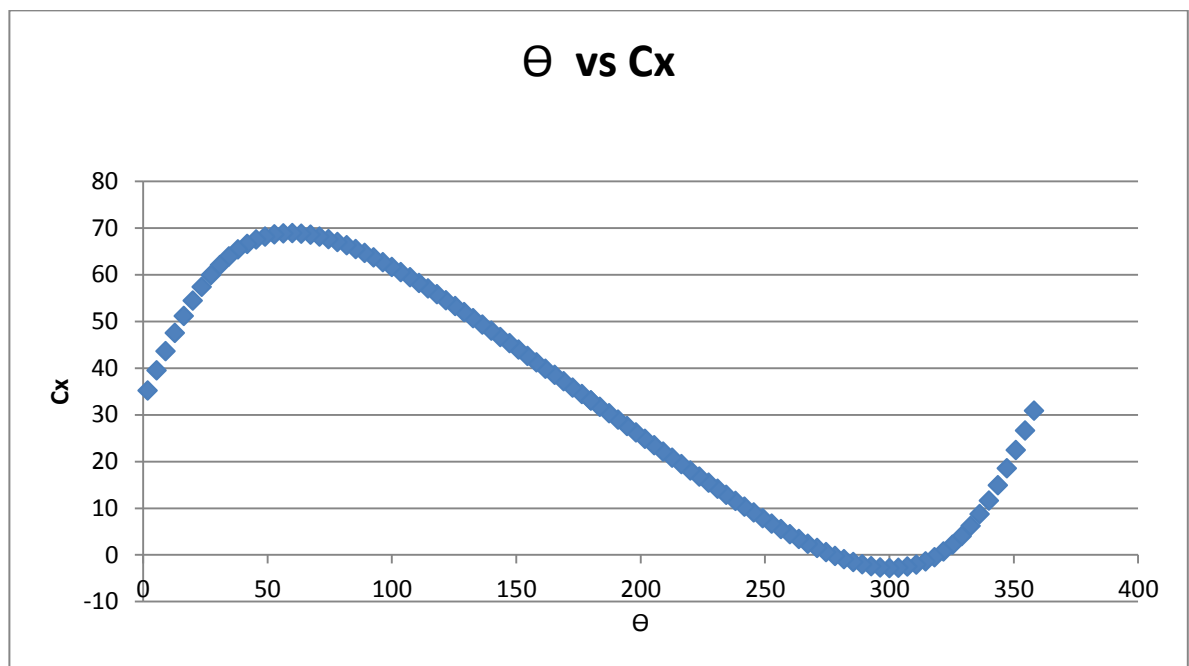


Fuente: Autores

Podemos apreciar en la gráfica la zona en la que se puede considerar como una línea recta el movimiento del punto de interés P, el cual está en un rango aproximado entre 90° y 270°.

La figura 45 muestra la distancia en x del punto de interés con respecto al punto O2 durante un ciclo de giro de la manivela.

**Figura45. Posición en x del punto de interés P vs Angulo de la manivela.**



Fuente: Autores

Podemos notar la porción en la que esta relación es lineal, la cual coincide con la porción del movimiento lineal, de esta grafica podríamos obtener la carrera aproximada rectilínea:

$$C_x(90^\circ) = 64,36 \text{ mm}$$

$$C_x(270^\circ) = 1,64 \text{ mm}$$

Carrera aproximada de línea recta:

$$\Delta x = 64,36 - 1,64 = 62,72$$

Esta carrera coincide exactamente con el valor dado en la tabla 6 para las relaciones de longitud de barras dadas.

**6.1.2 Diseño del pie o mecanismo de apoyo.** El pie o mecanismo de apoyo del robot es un sistema de suma importancia para asegurar la función principal del robot, la cual es, la escalada en vertical, el pie permite asegurar cada pata del robot a la superficie.

Este sistema involucra tanto la configuración del pie, sus grados de libertad, el mecanismo de suspensión, como también el medio con el cual se apoyará el robot sobre la superficie a trepar.

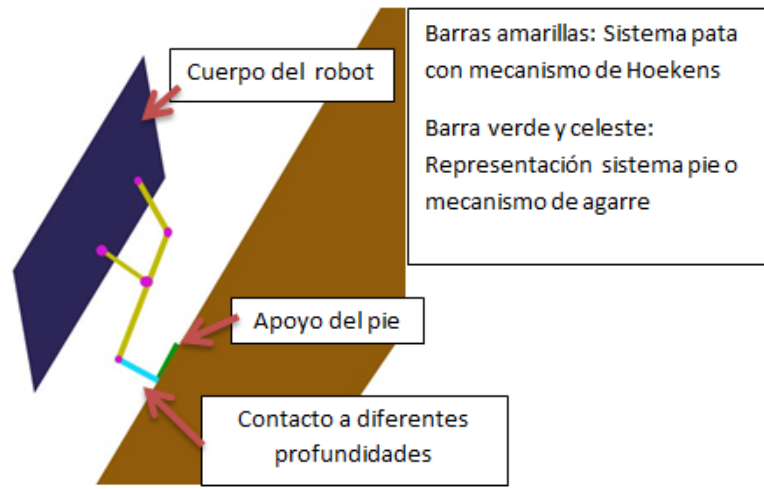
Para el diseño del pie, se toman en cuenta ciertos parámetros necesarios, a continuación se listan:

**Requerimientos:**

- Debe permitir apoyo del pie sobre superficies con diferentes configuraciones (inclinaciones, protuberancias) de tal manera que el sistema de agarre pueda realizar su función.
- Debe asegurar contacto a diferentes profundidades del terreno.
- Debe permitir libre avance del sistema mientras la pata está en apoyo, teniendo en cuenta la posición angular del pie mientras el sistema avanza.

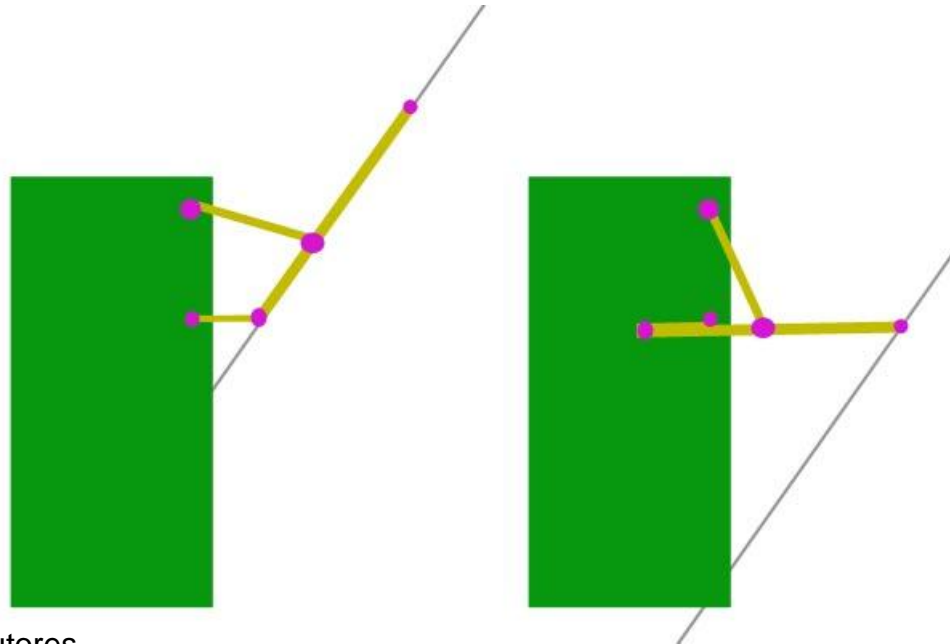
De manera más entendible, estos requerimientos se muestran en la figura 46 y 47.

**Figura 46. Representación gráfica de los requerimientos del sistema pie.**



Fuente: Autores

**Figura 47. Representación posición de barras mecanismo Hoekens durante el avance del sistema.**



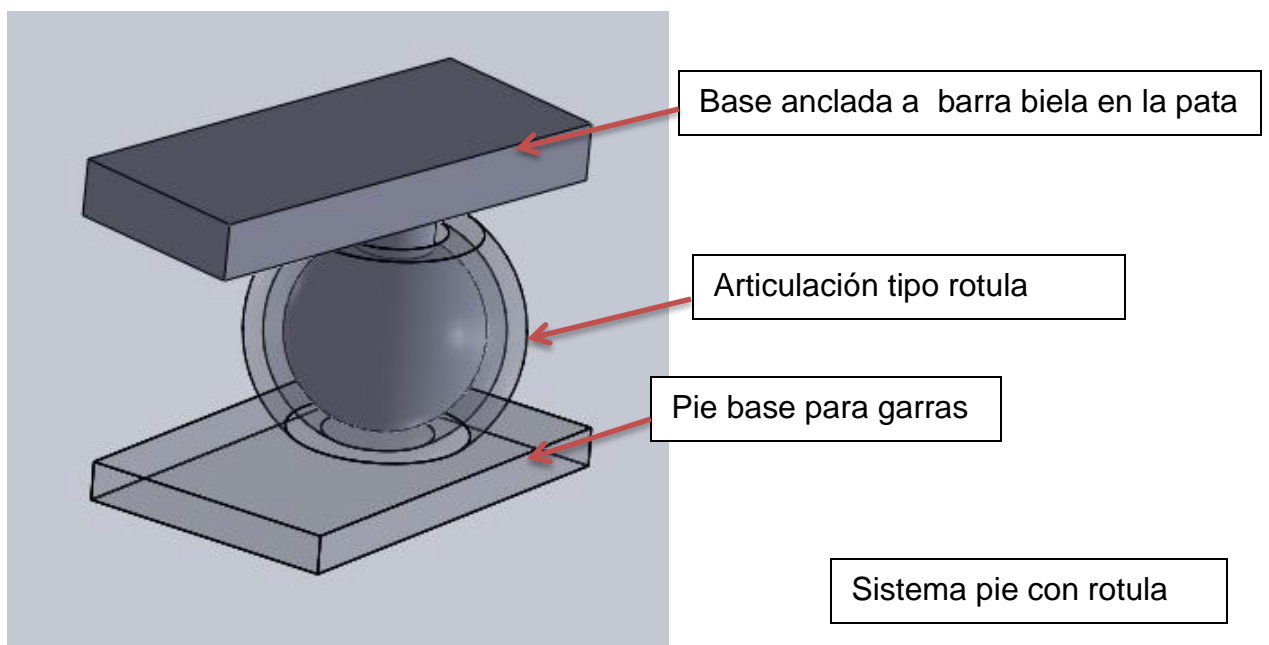
Fuente: Autores.

Como se puede apreciar en la figura 41, el mecanismo de avance del robot posee diversas posiciones angulares de las barras durante la fase de avance, en la cual el pie se supone anclado a la superficie a escalar.

Es de importancia saber la orientación de la barra biela, ya que de esta se sostendrá el sistema pie, esta orientación se puede obtener para diferentes puntos de la trayectoria rectilínea o de avance por medio de la gráfica 38.

A continuación se muestran algunos diseños propuestos (ver figura 48 y 49):

**Figura 48. Sistema pie rotula propuesto.**

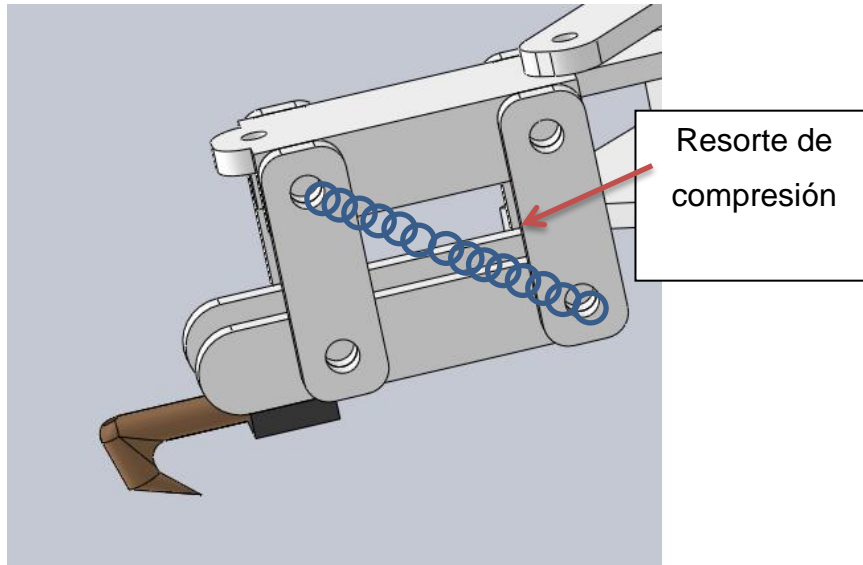


Fuente: Autores.

Este diseño permite una adecuación casi asegurada del pie en diferentes inclinaciones y configuraciones del terreno, ya que la articulación tipo rotula tiene los grados de libertad necesarios para dar esta característica. Este diseño fue descartado para el pre prototipo ya que su construcción es complicada, costosa y

suponía el diseño de un sistema de reposicionamiento del pie rotulado al estado inicial en cada periodo de avance de la pata en agarre.

**Figura 49. Diseño sistema pie propuesto tipo dactyl.**



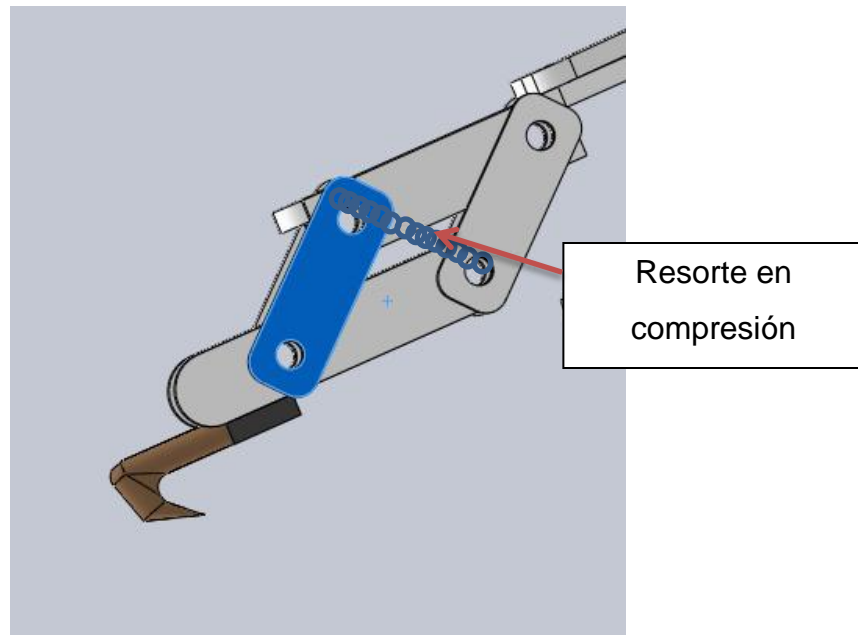
Fuente: Autores.

Este diseño de sistema de agarre usa un mecanismo de suspensión tipo tijera el cual equipado con dos resortes de compresión a cada lado de la tijera proporciona un almacenamiento de la fuerza aplicada por el hombro a medida que la tijera se cierra. El sistema de agarre está equipado de una sola garra tipo dactyl que se ayuda del movimiento de restitución del resorte y la fuerza del hombro para arraigarse a la superficie a escalar.

Fases de movimiento:

1. El sistema antes de entrar en contacto con la superficie se encuentra en su etapa de reposo, en el cual el resorte no ha sufrido deformación alguna.
2. Al entrar en contacto con la superficie, la tijera empieza a cerrarse y el resorte a comprimirse, a su vez la garra empieza a afirmarse sobre la superficie.

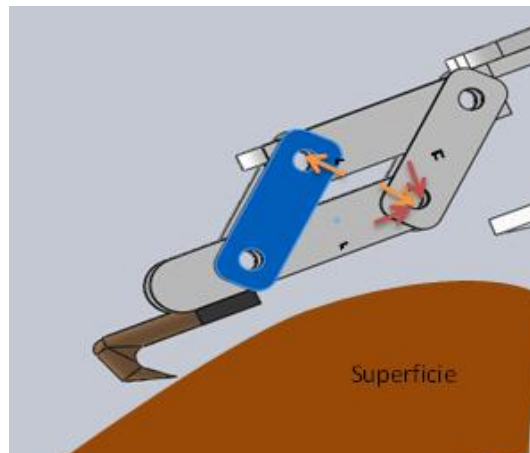
**Figura 50. Diseño sistema pie propuesto tipo dactyl en etapa de compresión de resorte.**



Fuente: Autores.

3. La tijera finaliza su etapa de cerrado una vez el servo del hombro deje de bajar la pata, la fuerza almacenada en el resorte obliga a la garra a rasgar la superficie y penetrar en ella gracias a la configuración de la garra.

**Figura 51. Fuerzas generadas en el sistema de agarre tipo dactyl en etapa final.**



Fuente: Autores.

La configuración del sistema pie propuesto tipo dactyl con sistema de suspensión tipo tijera, proporciona características que cumplen con los requerimientos necesarios para asegurar el agarre del pre prototipo. A continuación numeramos algunas ventajas de este sistema:

- El sistema de suspensión evita cargas excesivas sobre elementos rígidos tales como parras y pernos.
- El almacenamiento de la fuerza en el resorte es reusada para otras labores, la fuerza transversal ayuda a asegurar la garra en la superficie y la fuerza longitudinal ayuda a penetrar la superficie por medio del movimiento de la punta de la garra sobre el árbol.
- Este sistema es relativamente fácil de fabricar e implementar.

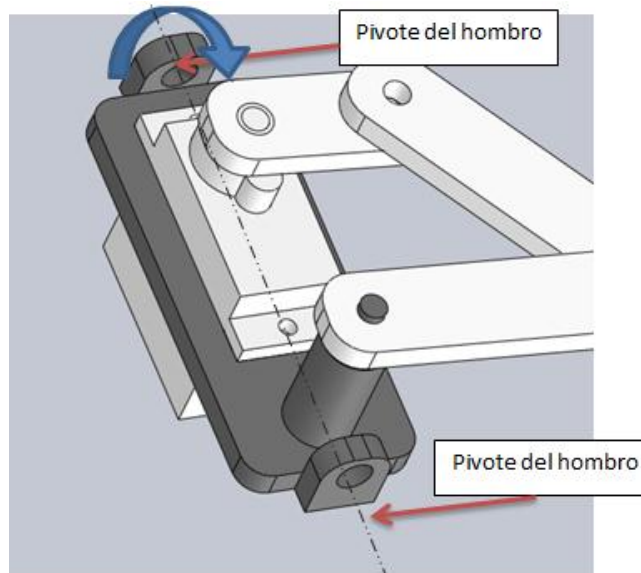
**6.1.3 Diseño mecanismo de suspensión.** Este diseño involucra el tamaño de las barras del mecanismo de suspensión y la configuración del resorte de tal manera que asegure una fuerza de desgarramiento y una fuerza normal a la superficie.

Ya que desconocemos estas fuerzas, se diseñara este sistema de tal manera que asegure el contacto de la garra con una superficie horizontal cuando el hombro se mantiene paralelo a la superficie. Con este mecanismo se experimentará sobre diferentes superficies con el fin de obtener estos datos que serán parámetros de diseño para el prototipo final.

**6.1.4 Diseño del sistema hombro.** Para asegurar fijación de las patas a diferentes profundidades de la superficie del árbol, el robot cuenta con el sistema hombro. Este posee un grado de libertad que permite girar todo el sistema pata alrededor del eje que atraviesa los puntos O2 - O4 del mecanismo de Hoekens visto.

En el diseño de este sistema, se ubicaron los puntos O2 y O4 del mecanismo de Hoekens en una plataforma tipo placa pivotada en sus extremos tal como se muestra en la figura 52

**Figura 52. Configuración sistema hombro y posición mecanismo Hoekens sobre este.**



Fuente: Autores

El diseño de esta placa se realizó teniendo en cuenta el tipo de actuador que sería usado en el movimiento del mecanismo de Hoekens, el cual debe girar junto con el sistema hombro y también está basado en las distancias de los puntos fijos O2 y O4 que garantizan la configuración dada en el sistema pata.

**6.1.5 Sistemas de actuación.** El pre prototipo realizado requería dos actuadores por módulo de pata, uno para accionar el mecanismo de Hoekens y el otro para realizar la labor de giro dada por el hombro.

Tentativamente, se eligieron para este pre prototipo, servomotores 9g marca hxt900 (Figura 53) ya que para los fines de investigación y búsqueda de

información para el prototipo final, brindaban el torque necesario para avanzar el robot en vertical y en trípodes (sin el sistema de control abordo y sin el resto de patas), además son servos económicos, pequeños y livianos.

**Figura 53. Servomotores tentativos para usar en el pre prototipo.**



Fuente: Autores

Para asegurar que estos actuadores proporcionaran el par necesario para avanzar el robot en trípodes, se diseñó una estructura del robot teniendo en cuenta el tamaño y configuración para este tipo de servomotores y se calculó un peso total estimado del robot para poder así verificar por medio de un análisis estático la elección.

A continuación se tabulan los datos de pesos de cada componente del preprototipo en la tabla 8.

**Tabla 8. Tabla de pesos del preprototipo.**

Cant.	Nombre	Alto	ancho	grosor	radio	Volumen [cm <sup>3</sup> ]	Densidad [g/cm <sup>3</sup> ]	Peso Unitario [kg]	Peso Unitario Gramos	Peso Total [kg]	
1	Placa base estructura	330	75	3		74,25	1,19	0,0883575	88,3575	0,0883575	
6	apoyos hombros	20	12	3		0,72	1,19	0,0008568	0,8568	0,0051408	
6	Apoyo hombros y motor	23	49,5	3		3,4155	1,19	0,00406445	4,064445	0,02438667	
12	Servomotores Hxt900							0,009	9	0,108	
12	Engranaje hombro grande			2	8,5	0,45396014	1,14	0,00051751	0,51751456	0,00621017	
6	Engranajes servo hombro			2	3,75	0,08835729	1,14	0,00010073	0,10072731	0,00060436	
1	Placa de desarrollo							0,01	10	0,01	
3	<u>Baterias</u>							0,206	206	0,618	
Total										0,86069951	
									Factor de seguridad	1,5	1,29104926

Fuente: Autores

El peso total que se debería tener en cuenta en la estática sería de 1,29 Kg, sin embargo, debido a que con cálculos realizados, este valor requiere exceder el torque máximo suministrado por los servomotores, el sistema se trabajará con baterías outboard, asimismo la placa de desarrollo, calculando un peso total en el robot de:

$$W_{total} = 0,364$$

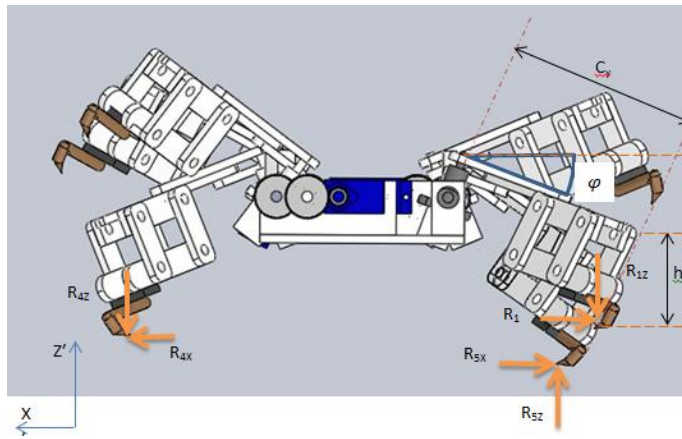
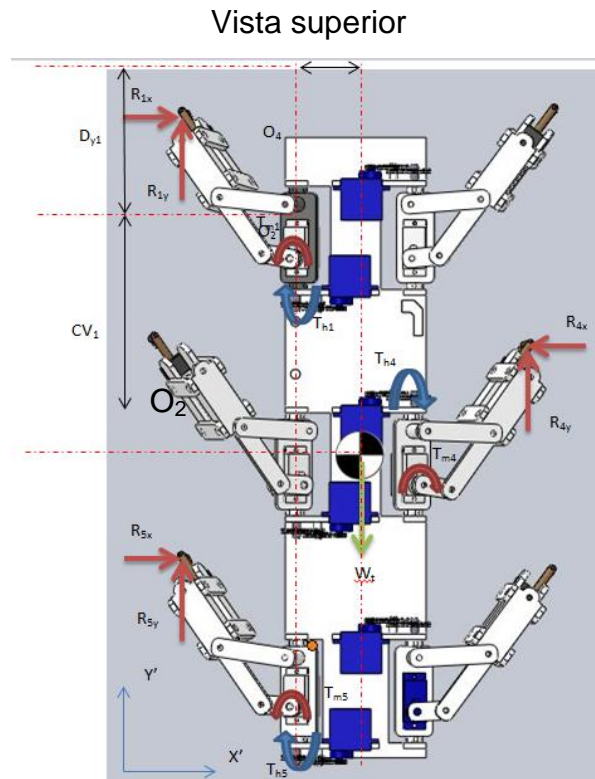
**6.1.6 Análisis estático para el movimiento del pre prototipo en trípodes alternados.** Es de importancia conocer el valor de las reacciones presentes en los elementos del robot en diferentes configuraciones de agarre en trípodes, esto es, diferentes valores de inclinación de hombros y barras, para conocer con certeza la posibilidad de un equilibrio estático del robot.

A continuación se realizará el análisis estático del pre prototipo con el fin de establecer criterios de estabilidad.

Se realizará este análisis inicial, suponiendo al robot escalando en posición totalmente vertical y sin inclinación alguna en sus ejes longitudinal, transversal y eje de guiñada, y manteniendo en apoyo los trípodes 1, 4, 5.

Se tratará como análisis inicial el conjunto total del robot como cuerpo rígido, con tres módulos de pata apoyados a diferente profundidad o diferente ángulo de hombro, considerando también las reacciones que se generan en la transmisión de los motores como suficientes para mantener estático el sistema, la posición del centro de gravedad se considerará constante y ubicada en la mitad del cuerpo del robot, tal como se muestra en la figura 51.

Figura 54. Diagrama de cuerpo libre conjunto total del robot con trípode 1,4,5 apoyados.



Fuente: Autores

Se representarán las fuerzas y brazos de momento a modo vectorial para facilitar el análisis:

Fuerzas en los puntos de apoyo:

$$\overrightarrow{F1} = Rx1(i) + Ry1(j) - Rz1(k)$$

$$\overrightarrow{F4} = -Rx4(i) + Ry4(j) - Rz4(k)$$

$$\overrightarrow{F5} = Rx5(i) + Ry5(j) + Rz5(k)$$

Vectores brazos de momento con respecto a centro de gravedad:

$$\overrightarrow{R1CG} = -(Cycos(\varphi1) + Ch)(i) + (Dy1 + Cv1)(j) - (Cysin(\varphi1) + h)(k)$$

$$\overrightarrow{R4CG} = (Cycos(\varphi4) + Ch)(i) + (Dy4 - Cv4)(j) - (Cysin(\varphi4) + h)(k)$$

$$\overrightarrow{R5CG} = -(Cycos(\varphi5) + Ch)(i) + (Cv5 - Dy5)(j) - (Cysin(\varphi5) + h)(k)$$

Cálculo de momentos con respecto al Centro de gravedad:

Debido a la fuerza del apoyo 1:

$$\overrightarrow{M1cg} = \begin{vmatrix} i & -j & k \\ -(Cycos(\varphi1) + Ch) & (Dy1 + Cv1) & -(Cysin(\varphi1) + h) \\ Rx1 & Ry1 & -Rz1 \end{vmatrix}$$

$$\begin{aligned}\overline{M1_{cg}} = & i\{[(Dy1 + Cv1)(-Rz1)] - [(Ry1)(-Cysin(\varphi1) + h)]\} \\ & - j\{[(-Cycos(\varphi1) + Ch)(-Rz1)] - [(Rx1)(-(Cysin(\varphi1) + h))]\} \\ & + k\{[(-Cycos(\varphi1) + Ch)(Ry1)] - [(Rx1)(Dy1 + Cv1)]\}\end{aligned}$$

Debido a la fuerza del apoyo 4:

$$\overline{M4_{cg}} = \begin{vmatrix} i & -j & k \\ (Cycos(\varphi4) + Ch) & (Dy4 - Cv4) & -(Cysin(\varphi4) + h) \\ Rx4 & Ry4 & -Rz4 \end{vmatrix}$$

$$\begin{aligned}\overline{M4_{cg}} = & i\{[(Dy4 - Cv4)(-Rz4)] - [(Ry4)(-Cysin(\varphi4) + h)]\} \\ & - j\{[(Cycos(\varphi4) + Ch)(-Rz4)] - [(Rx4)(-(Cysin(\varphi4) + h))]\} \\ & + k\{[(Cycos(\varphi4) + Ch)(Ry4)] - [(Rx4)(Dy4 - Cv4)]\}\end{aligned}$$

Debido a la fuerza del apoyo 5:

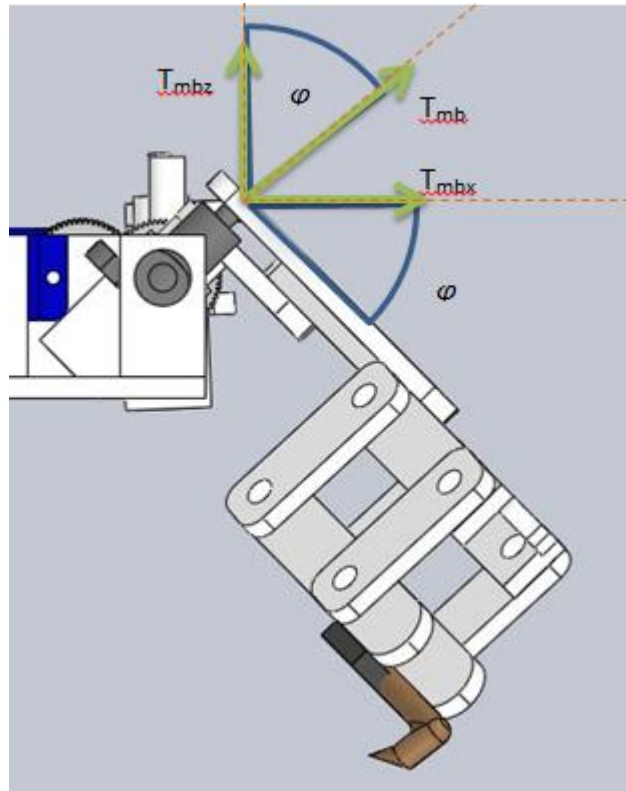
$$\overline{M5_{cg}} = \begin{vmatrix} i & -j & k \\ -(Cycos(\varphi5) + Ch) & (Cv5 - Dy5) & -(Cysin(\varphi5) + h) \\ Rx5 & Ry5 & Rz5 \end{vmatrix}$$

$$\begin{aligned}\overline{M5_{cg}} = & i\{[(Cv5 - Dy5)(Rz5)] - [(Ry5)(-(Cysin(\varphi5) + h))]\} \\ & - j\{[(-Cycos(\varphi5) + Ch)(Rz5)] - [(Rx5)(-(Cysin(\varphi5) + h))]\} \\ & + k\{[-(Cycos(\varphi5) + Ch)(Ry5)] - [(Rx5)(Cv5 - Dy5)]\}\end{aligned}$$

**Momentos debido a los torques aplicados por los actuadores, tanto de hombro como de pierna:**

En la figura 55 se muestra la descomposición del momento del motor de la pierna del módulo 1.

**Figura 55. Vector de momento aplicado por el actuador de la pierna sobre el módulo 1.**



Fuente. Autores

Torque debido a los actuadores, estos momentos son aplicados siempre en la dirección Y del sistema de referencia del robot.

$$\overline{M}_{th} = (-Th1 + Th4 - Th5)(j)$$

Los torques debido a los actuadores del mecanismo de Hoekens, se descomponen tal como se ve en la figura.

$$\overline{T}_{mb} = T_{mb} \sin(\varphi) (i) + T_{mb} \cos(\varphi)(k)$$

La dirección de sus componentes dependerá del módulo a analizar, siendo en dirección (k) positivos y en dirección (i) positivos para el módulo 5 y negativo para el módulo 1 y 4.

$$\overrightarrow{M_{tbrazos}} = (-T_{mb1} \sin(\varphi1) - T_{mb4} \sin(\varphi4) + T_{mb5} \cos(\varphi5))(i) + (T_{mb1} \cos(\varphi1) + T_{mb4} \cos(\varphi4) + T_{mb5} \cos(\varphi5))(k)$$

Cálculo estático en componentes i, j, k:

$$\sum fi = 0$$

$$Rx1 - Rx4 + Rx5 = 0$$

$$\sum fj = 0$$

$$Ry1 + Ry4 + Ry5 - W = 0$$

$$\sum fk = 0$$

$$-Rz1 - Rz4 + Rz5 = 0$$

$$\sum M(i) = 0$$

$$\begin{aligned} &(-T_{mb1} \sin(\varphi1) - T_{mb4} \sin(\varphi4) + T_{mb5} \cos(\varphi5)) \\ &+ \{[(Cv5 - Dy5)(Rz5)] - [(Ry5)(-Cysin(\varphi5) + h)]\} \\ &+ \{[(Dy4 - Cv4)(-Rz4)] - [(Ry4)(-Cysin(\varphi4) + h)]\} \\ &+ \{[(Dy1 + Cv1)(-Rz1)] - [(Ry1)(-Cysin(\varphi1) + h)]\} = 0 \end{aligned}$$

$$\sum M(j) = 0$$

$$\begin{aligned} &(-Th1 + Th4 - Th5) - \{[(-Cycos(\varphi5) + Ch)(Rz5)] - [(Rx5)(-Cysin(\varphi5) + h)]\} \\ &- \{[(Cycos(\varphi4) + Ch)(-Rz4)] - [(Rx4)(-Cysin(\varphi4) + h)]\} \\ &- \{[(-Cycos(\varphi1) + Ch)(-Rz1)] - [(Rx1)(-Cysin(\varphi1) + h)]\} = 0 \end{aligned}$$

$$\sum M(k) = 0$$

$$(T_{mb1} \cos(\varphi1) + T_{mb4} \cos(\varphi4) + T_{mb5} \cos(\varphi5))$$

$$+ \{[-(Cycos(\varphi5) + Ch)(Ry5)] - [(Rx5)(Cv5 - Dy5)]\}$$

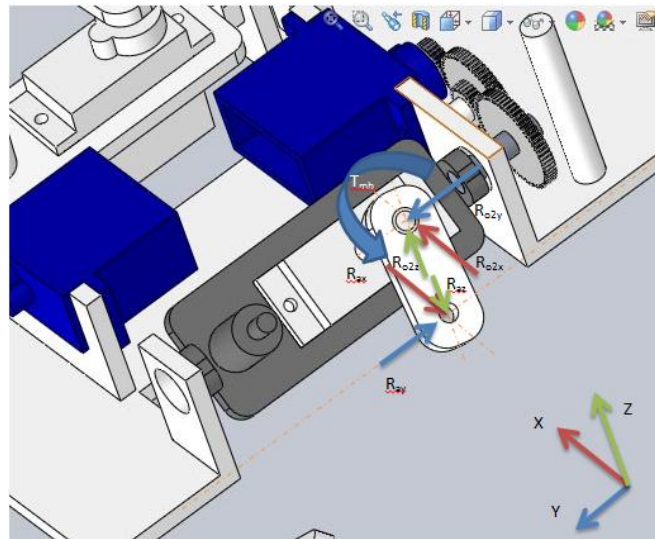
$$+ \{[(Cycos(\varphi4) + Ch)(Ry4)] - [(Rx4)(Dy4 - Cv4)]\}$$

$$+ \{[-(Cycos(\varphi1) + Ch)(Ry1)] - [(Rx1)(Dy1 + Cv1)]\} = 0$$

**Análisis de barras A-P y barra motriz O2- A, tomando como referencia el módulo 1.** Se realizará un análisis estático de las barras AP y O2-A del mecanismo de Hoekens para cada uno de los módulos en apoyo, tomando como referencia el módulo No. 1, variando estos cálculos tan solo en algunas direcciones para los demás módulos.

Las reacciones en la barra manivela se muestran en la figura 56.

**Figura 56. Reacciones en la barra manivela**



Fuente: Autores

Para el análisis de esta barra se realizarán momentos sobre el punto O2 para evitar integrar más incógnitas al sistema.

Brazo de momento O2 – A:

$$\overline{O2A} = \overline{O2A}\{\sin(\theta)\cos(\varphi)(-i) + [\cos(\theta)](j) - [\sin(\theta)\sin(\varphi)](k)\}$$

Fuerza punto A:

$$\overline{Fa} = Rax(-i) - Ray(j) - Raz(k)$$

Cálculo momentos respecto punto O2:

$$\overline{M_{FaO2}} = \begin{vmatrix} i & -j & k \\ -\overline{O2A}[\sin(\theta)\cos(\varphi)] & \overline{O2A}[\cos(\theta)] & -\overline{O2A}[\sin(\theta)\sin(\varphi)] \\ -Rax & -Ray & -Raz \end{vmatrix}$$

$$\begin{aligned} \overline{M_{FaO2}} = & i\{(-\overline{O2A}[\cos(\theta)]Raz) - (\overline{O2A}[\sin(\theta)\sin(\varphi)]Ray)\} \\ & - j\{(\overline{O2A}[\sin(\theta)\cos(\varphi)]Raz) - (\overline{O2A}[\sin(\theta)\sin(\varphi)]Rax)\} \\ & + k\{(\overline{O2A}[\sin(\theta)\cos(\varphi)]Ray) + (\overline{O2A}[\cos(\theta)]Rax)\} \end{aligned}$$

Momento debido a los actuadores del mecanismo de Hoekens:

$$\overline{T_{mb}} = T_{mb}\sin(\varphi)(-i) + T_{mb}\cos(\varphi)(k)$$

La dirección del momento en dirección (i) depende del módulo de pata en el cual se plantean las ecuaciones, siendo dirección negativa en i para los módulos 1,3, 5 y positiva para los módulos 2,4, 6 cuando el robot escala a un Angulo  $0 < \varphi < 90$ .

Cálculo estático de momentos componentes i, j, k:

$$\sum M(i) = 0$$

$$-T_{mb1} \sin(\varphi) - (\overline{O2A}[\cos(\theta)]Raz) - (\overline{O2A}[\sin(\theta) \sin(\varphi)]Ray) = 0$$

$$\sum M(j) = 0$$

$$-\{(\overline{O2A}[\sin(\theta) \cos(\varphi)]Raz) - (\overline{O2A}[\sin(\theta) \sin(\varphi)]Rax)\} = 0$$

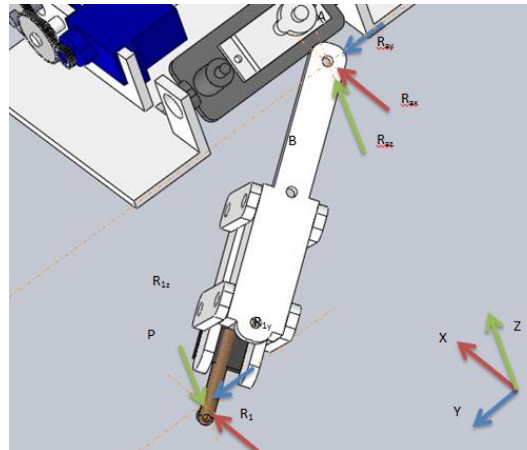
$$(\sin(\theta) \{ \cos(\varphi)Raz \} - (\sin(\varphi)Rax)) = 0$$

$$\sum M(k) = 0$$

$$T_{mb} \cos(\varphi) + (\overline{O2A}[\sin(\theta) \cos(\varphi)]Ray) + (\overline{O2A}[\cos(\theta)]Rax) = 0$$

Las reacciones en la barra biela se muestran en la figura 57.

**Figura 57. Reacciones en la barra biela.**



Fuente: Autores

Para el análisis de esta barra se realizarán momentos sobre el punto B para evitar integrar más incógnitas al sistema y solo involucrar las incógnitas establecidas en los DCLs anteriores.

Brazo de momento B – A:

$$\overrightarrow{BA} = \overline{AB}\{\cos(\varphi)\sin(\alpha)(i) + \cos(\varphi)\cos(\alpha)(-j) + \sin(\varphi)(k)\}$$

Las direcciones dadas dependen del módulo a analizar, siendo direcciones contrarias en (i) para el módulo 4.

Brazo de momento B – P:

$$\overrightarrow{BP} = \overline{BP}\{\cos(\varphi)\sin(\alpha)(-i) + \cos(\varphi)\cos(\alpha)(j) + \sin(\varphi)(-k)\}$$

Al igual que el brazo de momento anterior, las direcciones de este dependen del módulo a analizar, siendo direcciones contrarias en (i) para el módulo 4.

Fuerza punto A:

$$\overrightarrow{Fa} = Rax(i) + Ray(j) + Raz(k)$$

Fuerza punto P:

$$\overrightarrow{Fp} = Rx(i) + Ry(j) + Rz(-k)$$

Cálculo momentos respecto punto B de la Fuerza en A:

$$\overrightarrow{M_{FaB}} = \begin{vmatrix} i & -j & k \\ \overline{AB}[\cos(\varphi)\sin(\alpha)] & -\overline{AB}[\cos(\varphi)\cos(\alpha)] & \overline{AB}[\sin(\varphi)] \\ Rax & Ray & Raz \end{vmatrix}$$

$$\begin{aligned}\overrightarrow{M_{FaB}} = & i\{(-\overline{AB}[\cos(\varphi)\cos(\alpha)]Raz) - (\overline{AB}[\sin(\varphi)]Ray)\} \\ & - j\{(\overline{AB}[\cos(\varphi)\sin(\alpha)]Raz) - (\overline{AB}[\sin(\varphi)]Rax)\} \\ & + k\{(\overline{AB}[\cos(\varphi)\sin(\alpha)]Ray) + (\overline{AB}[\cos(\varphi)\cos(\alpha)]Rax)\}\end{aligned}$$

Cálculo momentos respecto punto B de la Fuerza en P:

$$\overrightarrow{M_{FpB}} = \begin{vmatrix} i & -j & k \\ -\overline{BP}[\cos(\varphi)\sin(\alpha)] & \overline{BP}[\cos(\varphi)\cos(\alpha)] & -\overline{BP}[\sin(\varphi)] \\ Rx & Ry & -Rz \end{vmatrix}$$

$$\begin{aligned}\overrightarrow{M_{FpB}} = & i\{(-\overline{BP}[\cos(\varphi)\cos(\alpha)]Rz) + (\overline{BP}[\sin(\varphi)]Ry)\} \\ & - j\{(\overline{BP}[\cos(\varphi)\sin(\alpha)]Rz) + (\overline{BP}[\sin(\varphi)]Rax)\} \\ & + k\{(-\overline{BP}[\cos(\varphi)\sin(\alpha)]Ry) - (\overline{BP}[\cos(\varphi)\cos(\alpha)]Rax)\}\end{aligned}$$

Cálculo estático de momentos componentes i, j, k:

$$\begin{aligned}\sum M(i) = & 0 \\ \{(-\overline{AB}[\cos(\varphi)\cos(\alpha)]Raz) - (\overline{AB}[\sin(\varphi)]Ray)\} \\ & + \{(-\overline{BP}[\cos(\varphi)\cos(\alpha)]Rz) + (\overline{BP}[\sin(\varphi)]Ry)\} = 0\end{aligned}$$

$$\begin{aligned}\sum M(j) = & 0 \\ \{(\overline{AB}[\cos(\varphi)\sin(\alpha)]Raz) - (\overline{AB}[\sin(\varphi)]Rax)\} \\ & + \{(\overline{BP}[\cos(\varphi)\sin(\alpha)]Rz) + (\overline{BP}[\sin(\varphi)]Rax)\} = 0\end{aligned}$$

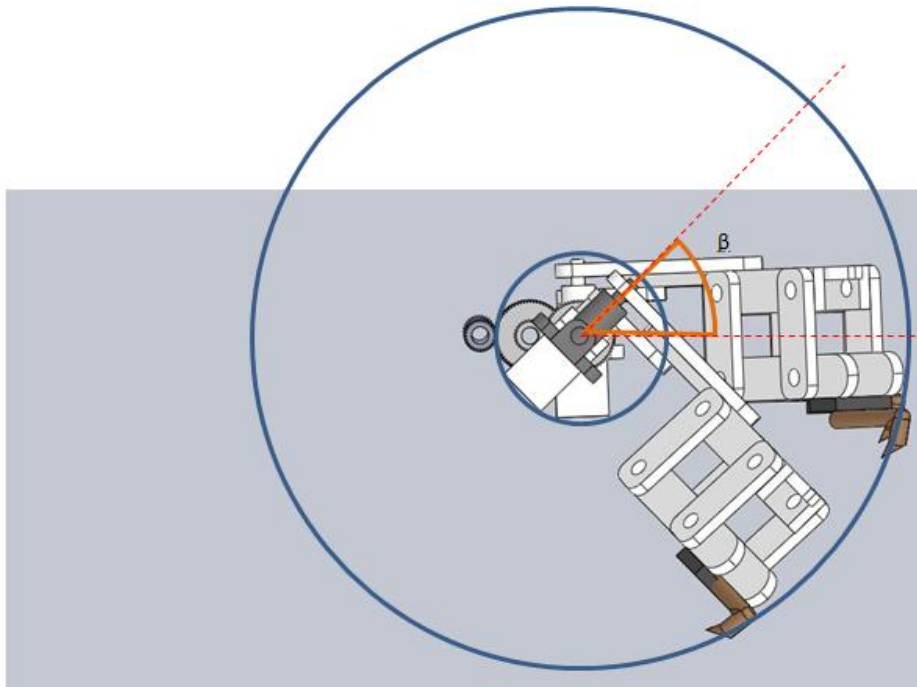
$$\sum M(k) = 0$$

$$\{(\overline{AB}[\cos(\varphi) \sin(\alpha)]Ray) + (\overline{AB}[\cos(\varphi) \cos(\alpha)]Rax)\} \\ + \{(-\overline{BP}[\cos(\varphi) \sin(\alpha)]Ry) - (\overline{BP}[\cos(\varphi) \cos(\alpha)]Rx)\} = 0$$

**6.1.7 Diseño de estructura de soporte o cuerpo del robot.** La estructura base sirve de soporte para los elementos del robot, su diseño se basa en garantizar la ubicación de cada elemento de tal manera que no interfieran entre ellos al realizar su movimiento y además ubicando estos con el fin de mantener una simetría del robot y así una estabilidad mejorada.

A continuación (Figura 58) se mostrarán algunos de los elementos del robot y sus rangos de movimiento:

**Figura 58. Rango de movimiento del hombro.**



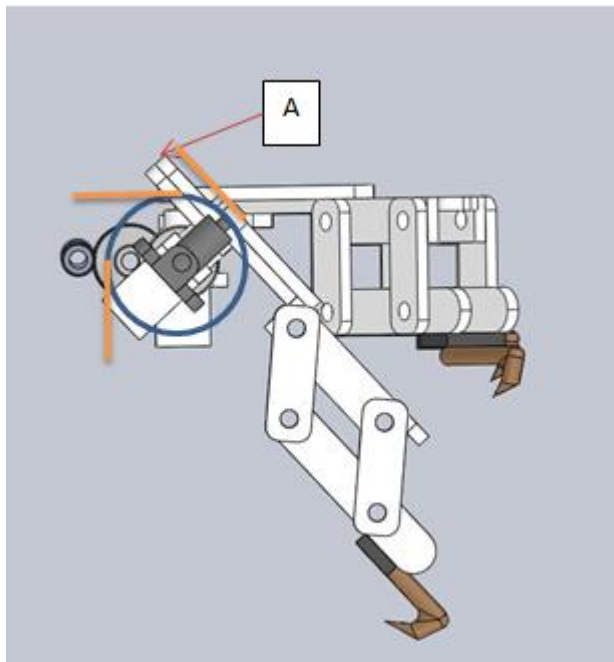
Fuente: autores

Como podemos apreciar en la figura 58, el modulo hombro realizará una rotación alrededor de su eje de giro. Las circunferencias azules nos muestran las posibles posiciones de los puntos extremos de algunos elementos de interés que podrían ser interferidos por otros elementos del robot si la estructura no es diseñada teniendo en cuenta estos movimientos.

Para fines de diseño, es importante limitar el rango de movimiento que tendrán los elementos móviles.

El rango de  $\beta$  deseado para un rango de apoyo y de levantamiento suficiente es  $[0^\circ, 180^\circ]$ , así para  $\beta=0^\circ$  el robot abrazará el árbol a escalar y para  $\beta=180^\circ$  el robot levantará sus patas a una posición máxima suficiente para evitar algún obstáculo que impida el posicionamiento de su pata.

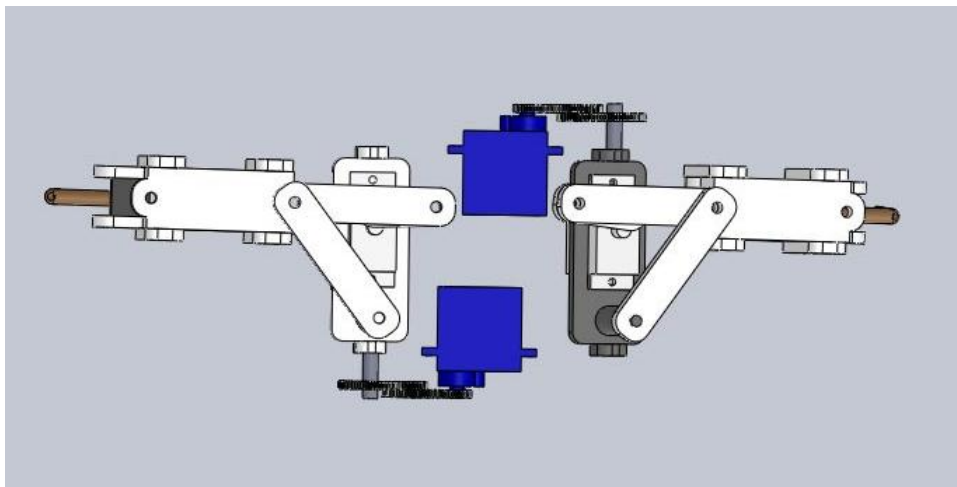
**Figura 59. Rango de interés del punto A para diferentes inclinaciones de Hombro.**



Fuente: Autores.

En la figura 59, se muestra el módulo hombro con un ángulo  $\Theta$  de manivela diferente, para  $\Theta=270^\circ$  se encuentra un punto de interés extremo A, el cual para diferentes ángulos  $\beta$ , se encuentra tangente a la circunferencia azul de la figura 56y a una distancia dada por la longitud de la barra O2A. Este punto podría interferir con el servomotor que se ubicará en la sección trasera del módulo hombro tal como muestra la figura 60.

**Figura 60. Ubicación deseada de sistema hombro-servomotor para una dupla de patas.**



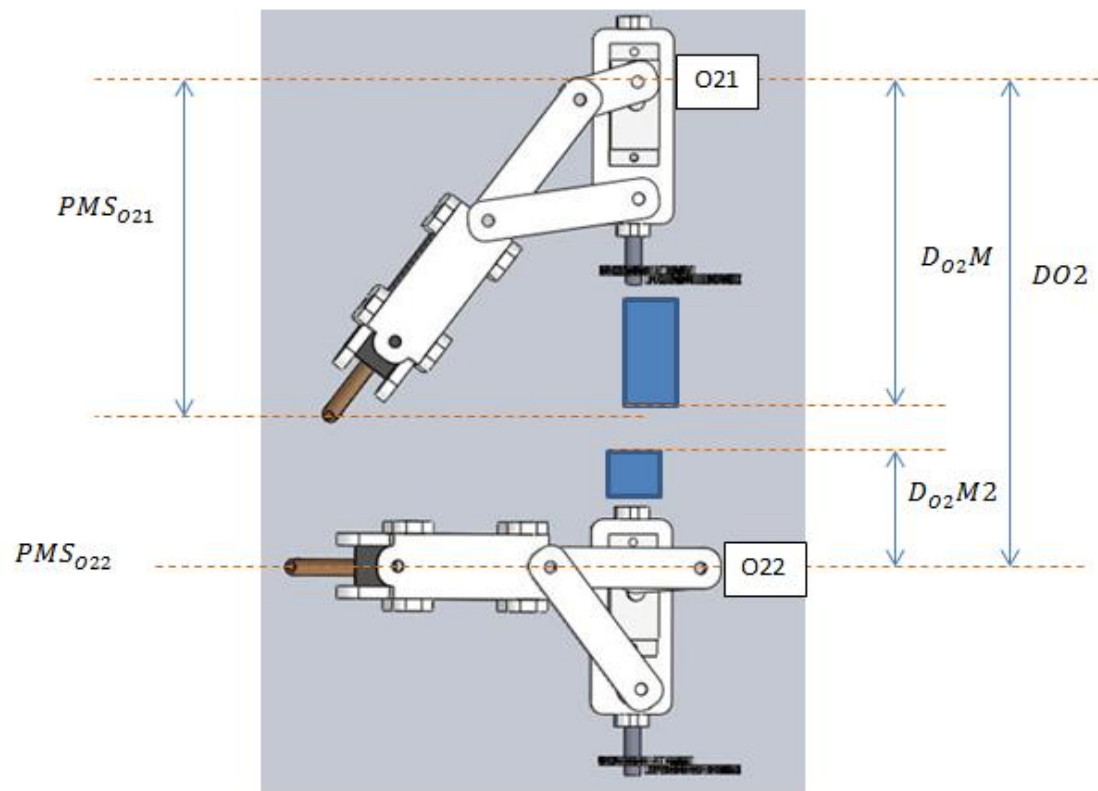
Fuente: Autores.

Esta ubicación del actuador del hombro se ha diseñado tanto para ahorrar espacio como para evitar un largo del sistema total mayor. Adicionalmente, esta ubicación favorece el diseño del tren de engranajes necesario para transmitir el torque al hombro, ya que el torque brindado por el actuador no era suficiente para penetrar las superficies de interés (ver cálculo fuerza de penetración), siendo necesario un tren multiplicador de torque, el cual fue diseñado como se ve en la figura 60.

Para ángulos  $\beta$  entre  $0^\circ$  y  $90^\circ$ , el saliente inferior del servomotor ubicado en la placa del hombro también podría interferir, es por esto que estas distancias se tomarán en cuenta para el diseño de la estructura base o cuerpo del robot.

La distancia para evitar la interferencia de patas en movimiento es mostrada en la figura 61.

**Figura 61. Distancias entre patas**



Fuente: Autores.

La distancia entre puntos de apoyo es definida por la configuración del mecanismo de avance del sistema, para una dupla de patas que pueden causar este tipo de interferencia, el punto crítico se forma cuando la pata trasera está en su punto máximo superior PMS y la pata delantera en su punto máximo inferior PMI.

En el mecanismo de Hoekens, estos puntos están dados aproximadamente con respecto al ángulo de la manivela así:

$$\text{PMI: } \theta = 270^\circ$$

$$\text{PMS: } \theta = 90^\circ$$

El PMS y el PMI se pueden calcular para el punto O2 de cada módulo, siendo la distancia mínima entre puntos O2:

$$DMO2 = PMS_{O21} - PMI_{O22}$$

Tomados desde un punto de origen en O2 y positivo hacia adelante de este en dirección y.

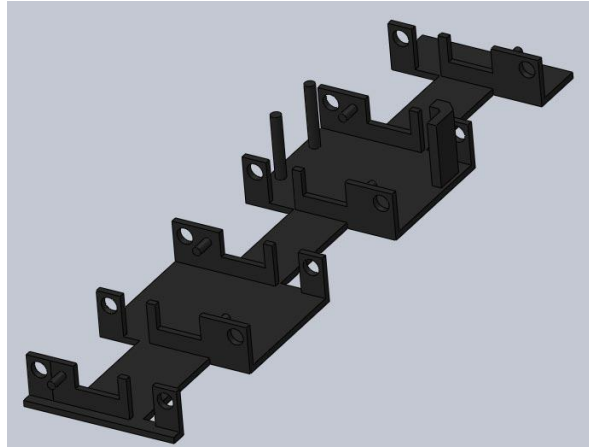
Esta distancia es la mínima entre puntos O2, sin embargo hay que tener en cuenta que debido a los elementos que van en medio de estos dos puntos (tren de engranajes, sensores de posición, entre otros) la distancia mínima puede que ya no dependa del PMI y del PMS sino de la distancia entre los elementos intermedios delante de O21 y los elementos intermedios detrás de O22 ver figura 61.

Así, la distancia mínima sería:

$$DMO2 = D_{O2M} + D_{O2M2}$$

Tomando en cuenta estos parámetros de funcionamiento, se diseñó la estructura o cuerpo del robot (Ver figura 62) que contendrá los módulos de pata y actuadores necesarios para el funcionamiento.

Figura 62. Base o cuerpo robot escalador



Fuente. Autores.

**6.1.8 materiales y proceso de manufactura usado para el pre prototipo.** Al hacer un análisis general de materiales y procesos de manufactura disponibles, se optó por fabricar el pre prototipo en acrílico por sus propiedades físicas (ver tabla 9) y la facilidad de corte de este a través de empresas de manufactura especializadas en el corte a laser del acrílico.

Adicionalmente, el costo de los materiales no es excesivo y se pueden realizar pegues fácilmente con un compuesto químico llamado cloruro de Metileno, el cual también es asequible en el mercado regional.

**Tabla 9. Propiedades del polímero de metil metacrilato:**

<b>Polimetilmetacrilato</b>	<b>Tracción</b>	<b>Flexión</b>	<b>Compresión</b>
<b>Punto de fusión:</b>	160 °C		
<b>Densidad:</b>	1,18 g/cm <sup>3</sup>		

<b>Fórmula:</b>	$(C_5O_2H_8)n$		
<b>Punto de ebullición:</b>	200 °C		
<b>Resistencia a la rotura</b>	562 – 773 kg/cm <sup>2</sup>	840 – 1300 kg /cm <sup>2</sup>	773 – 1.330 $\frac{kg}{cm^2}$
<b>Módulo de elasticidad</b>	24600 – 310000 kg /cm <sup>2</sup>	27500 – 33400 kg /cm <sup>2</sup>	27.500 – 33.300 $\frac{kg}{cm^2}$

Fuente: <http://www.famadacrilicos.com.ar/>

Proceso de manufactura y ensamble:

1. Compra de lámina de acrílico
2. Generación de planos de corte para cortadora laser.
3. Corte de piezas
4. Ensamble mecanismo de pata.
5. Ensamble mecanismos de suspensión.
6. Ensamble de estructura base
7. Ensamble sistema hombro
8. Ensamble trenes de engranajes y acople a servomotores.

**6.1.8.1 Compra de lámina de acrílico.** Como diseño para el pre prototipo, se escogió lámina de acrílico de 3 líneas, con el fin de evitar pesos excesivos en el sistema y además poder acoplar las barras entre ellas usando remaches tipo pop como pernos.

**6.1.8.2 Generación de planos y corte de piezas.** De los modelos CAD, se obtuvieron los planos de corte, teniendo en cuenta que para la cortadora laser, se requerían las líneas por donde el láser cortaría la lámina en longitud real.

Ya que algunas piezas se generaron enterizas, fue necesario el uso de software vectorial para realizar ciertos arreglos a los planos y así poder incorporar todas las piezas necesarias que serían cortadas.

Estos planos se entregaban en formato CDR de Corel draw. Ver anexos tal1.

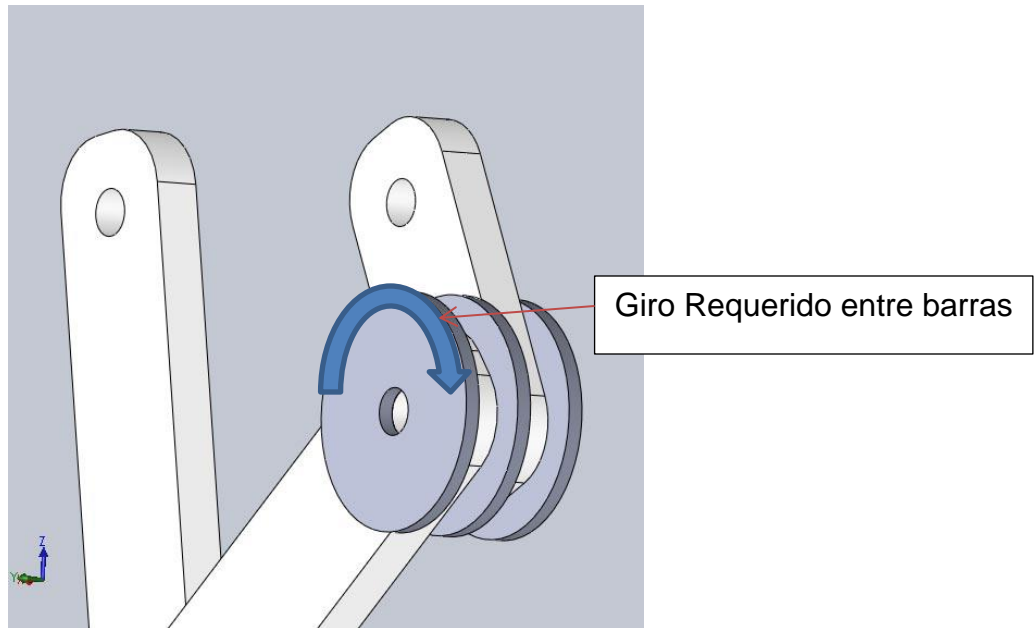
**6.1.8.3 Corte de piezas.**El corte de las piezas fue realizado por una máquina router de corte a laser automatizada, esta máquina corta siguiendo trayectorias dadas por los planos de corte en formato vectorial.

**6.1.8.4 Ensamble mecanismo pata.**El ensamble del mecanismo de Hoekens se realizó uniendo las barras con remaches tipo POP, la unión directa de las barras de acrílico con este tipo de remache no permitía el giro entre ellas, además debido a la fragilidad del acrílico, este se quebraba.

Como solución inicial se probó el acople adicionando arandelas entre barras tal como muestra la figura 62. Con esto, las barras no se quebraban pero aún no pivotaban, quedando estas totalmente rígidas.

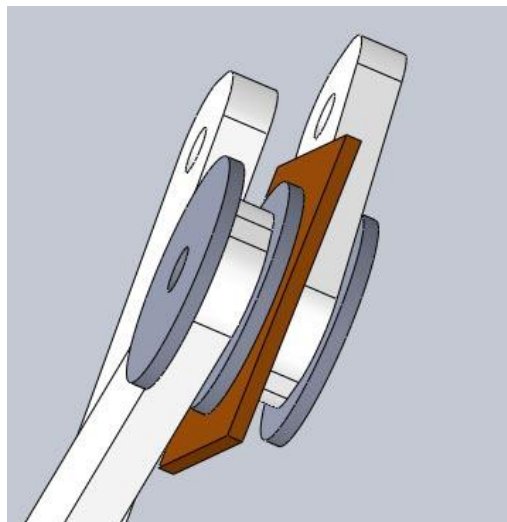
Para garantizar el giro entre estas era necesario dejar una tolerancia entre barra y arandela, para esto se insertó cartón a modo de arandela entre cada acople de barra para que después de ser remachado se retirara, generando la tolerancia requerida tal como muestra la figura 63.

**Figura 62. Acople de barras usando arandelas entre estas.**



Fuente: Autores

**Figura 63. Cartón insertado entre acople de barras.**

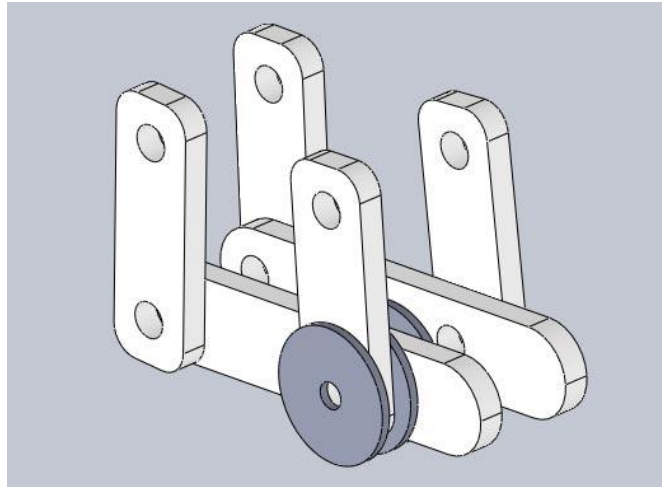


Fuente: Autores

**6.1.8.5 Ensamble mecanismos de suspensión.** Estos mecanismos se ensamblaron al igual que las patas para garantizar un giro entre barras y usando

remaches como pernos y ponchándolos con una arandela de cartón. La configuración de estas barras se muestra en la figura 64.

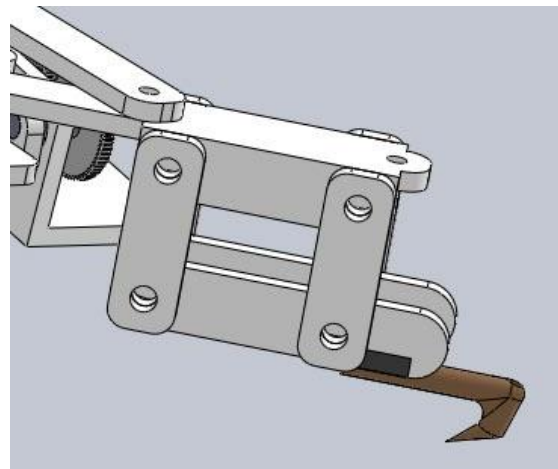
**Figura 64. Configuración barras en mecanismo de suspensión.**



Fuente: Autores.

Este mecanismo finalmente se acopla a la barra biela mediante remaches unidos a una placa vertical pegada con cloruro de metileno a la barra biela como se ve en la figura 65.

**Figura 65. Configuración acople sistema suspensión y mecanismo Hoekens.**



Fuente: Autores.

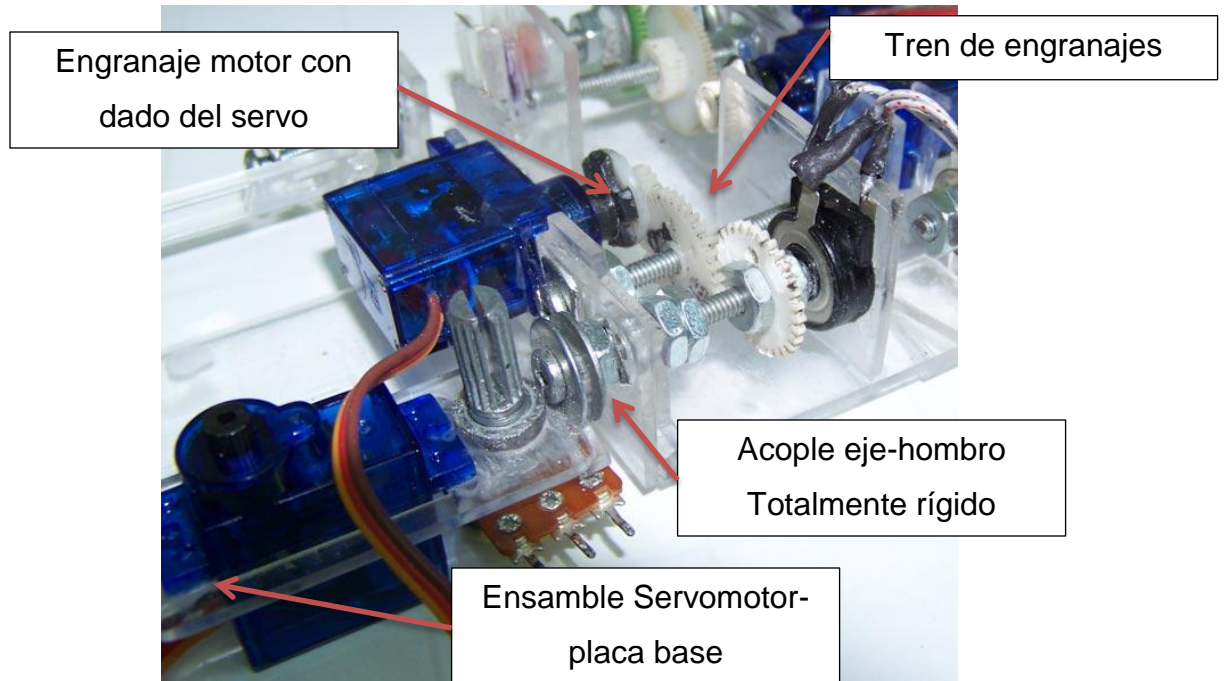
**6.1.8.6 Ensamble estructura base.** Con las piezas cortadas, base de apoyo ejes, base para eje de tren de engranajes y servomotor, se procede a pegarlas con cloruro. Se debe tener en cuenta la perpendicularidad necesaria de estas piezas con respecto a la placa base, de tal forma que los ejes a acoplar no tengan excentricidades.

La estructura base y sus componentes fue ensamblada de forma modular, es decir se ensamblaban los apoyos de cada hombro y a la vez se realizaba en ensamble de este, con el fin de evitar interferencias a la hora de trabajar.

**6.1.8.7 Ensamble sistema hombro.** Este sistema se ensambló uniendo las piezas de apoyo de eje del hombro a la placa base del hombro con cloruro, a estas piezas de apoyo una vez pegadas a la base se le acoplaron tornillos de  $\frac{1}{4}$ ", arandela y tuerca de tal manera que el tornillo hacía las veces de eje de rotación, ver figura 66.

Se ubican los elementos que van sobre la placa base, tales como el servomotor, el cual se sujeta a esta mediante cloruro y el potenciómetro que servirá como sensor de posición del mecanismo de Hoekens y a la vez como eje de la barra manivela, ver figura 66.

**Figura 66. Vista ensamble modulo hombro.**



Fuente: Autores

Cada uno de estos módulos es montado sobre los apoyos de la placa base mediante sus ejes, este diseño requería que a medida que se iban ensamblando los ejes se debían insertar estos a los apoyos, ya que de lo contrario, los apoyos fijos de la base no permitirían su ensamble.

Al ensamblar los ejes a los apoyos de la placa base, es necesario limitar su movimiento axial, por lo que se ubica un par de tuercas en el eje pero sin realizar un apriete contra el apoyo de la base sino entre ellos mismos para garantizar el giro del eje del hombro y también evitar que las tuercas se desenrosquen.

**6.1.1.8 Ensamble trenes de engranajes y acople sistemas de actuación.** La consecución y/o fabricación de engranajes que se adaptaran a los requerimientos de la caja de reducción para mover el hombro no era asequible y generaba gastos

elevados, por este motivo se buscaron engranajes de diferentes tamaños que se obtuvieron de otros sistemas y se trató de obtener una reducción parecida.

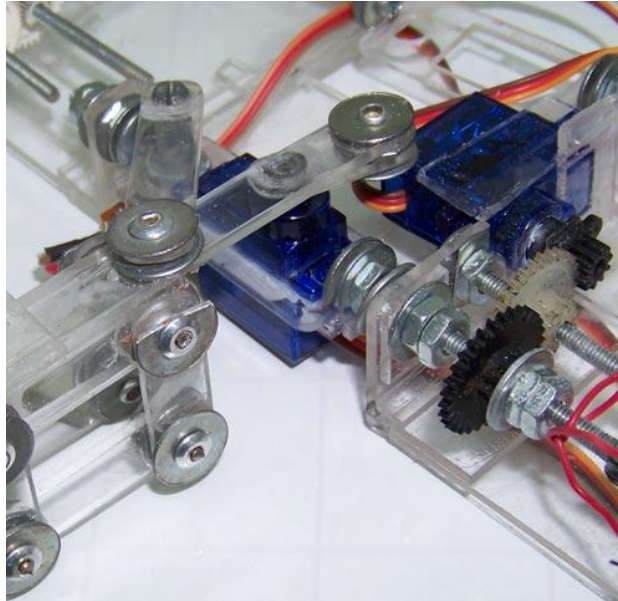
Ya que los engranajes requerían posición de los ejes diferente a las diseñadas, el ensamble de estos y la consecución de una transmisión sin interferencias fue difícil de conseguir.

El acople de los sistemas de actuación se obtuvo mediante el aprovechamiento de los dados de las barras que venían junto con los servomotores, estos dados se adaptaron al engranaje motriz con pegante industrial a base de cianocrilato.

En la figura 66 se aprecia el engranaje motor del servo del hombro acoplado con el dado que viene junto con el servomotor.

El engranaje final del tren de engranajes se acopla mediante apriete de tuercas al eje del hombro tal como muestra la figura 67.

**Figura 67. Acople engranaje final-eje hombro**



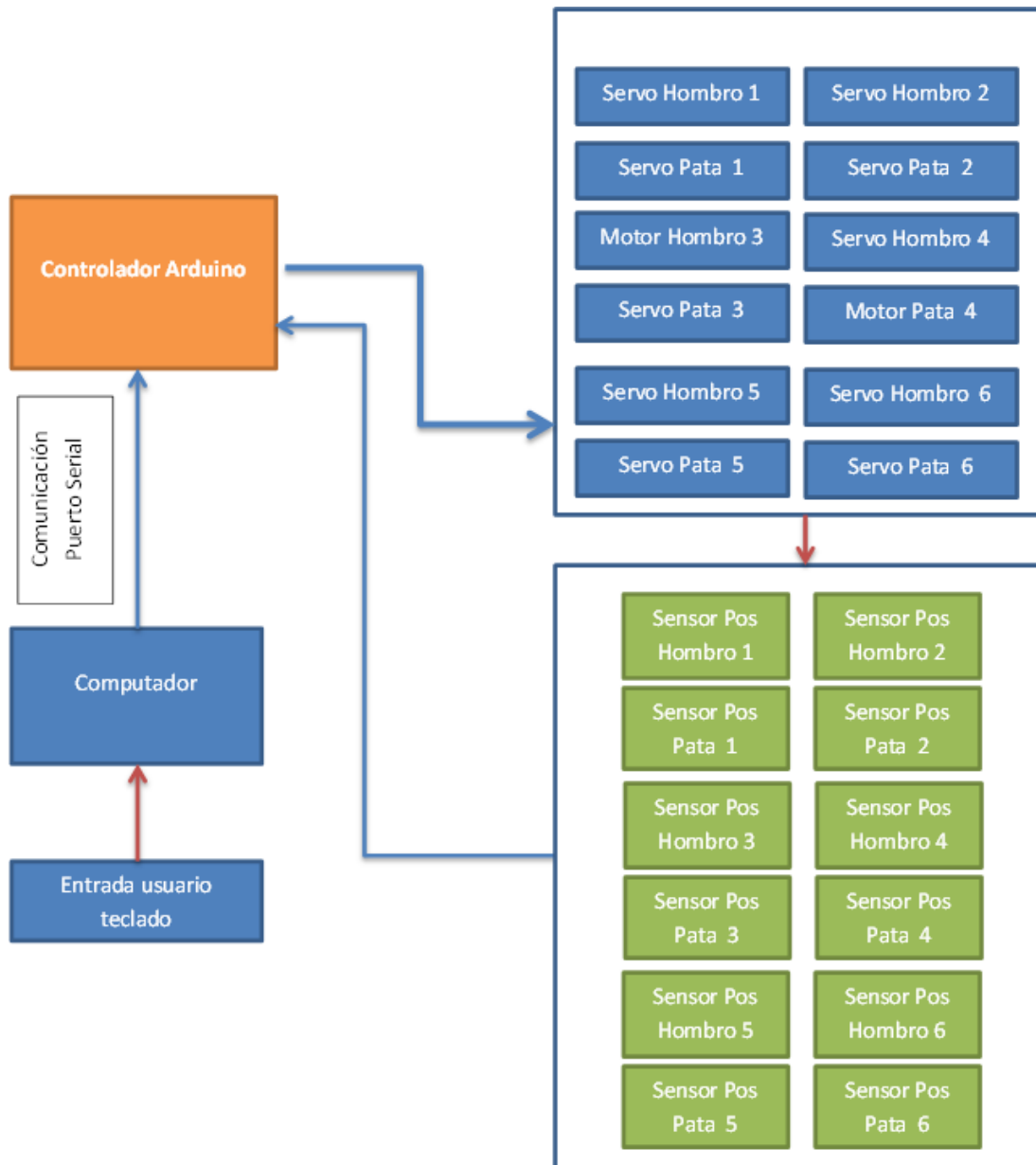
Fuente: Autores.

El acople del servomotor del mecanismo de Hoekens se realizó al igual que el del hombro con la diferencia que este dado se acopló directamente a la barra motriz.

## **6.1 DISEÑO DEL SISTEMA ELECTRÓNICO Y DE CONTROL**

En la figura 68 se representa a grandes rasgos el funcionamiento del control del pre prototipo del robot escalador:

**Figura 68. Representación sistema control asistido pre prototipo**



Fuente: Autores.

Para obtener un mayor control sobre los diferentes parámetros de desempeño del robot, el sistema de control del pre prototipo fue diseñado para trabajar en modo control remoto y en modo semiautónomo.

El modo de control remoto permite a través de la interfaz serial entre computador y Arduino la inserción de instrucciones específicas al robot, tales como movimiento de subida y bajada de hombros y avance y retroceso de patas concretas.

Este sistema de inserción de órdenes se basó en la librería Tkinter de Python, la cual permite capturar en tiempo real la tecla presionada en el teclado y según la tecla, se llama una función en python que envía datos por el puerto serial, estos datos finalmente son interpretados por arduino para mover cierto actuador en cierta dirección o detenerlo.

A continuación se muestra con más detalle este proceso:

**6.2.1 Captura de tecla presionada a través de librería Tkinter de Python.** El módulo Tkinter de python, es la interface estándar del toolkit Tk GUI, generalmente usado para generar interfaces usuario-programa.

El módulo Tkinter incluye módulos adicionales de python, constantes predefinidas y métodos que son invocados al realizar acciones de usuario, es allí donde encontramos el evento “key” y “KeyRelease”, el primero es invocado cuando se presiona una tecla y el segundo cuando se suelta.

Tkinter proporciona un poderoso mecanismo para enlazar un evento a una función, la sintaxis para esto es:

*widget.bind(event, handler)*

Donde “widget” es el frame de donde queremos capturar el evento, “event” es el evento al cual queremos enlazarle la función definida en “handler”.

Para nuestro caso, se enlazo el evento key y keyrelease a funciones que después analizarían la tecla presionada o soltada y dependiendo de su valor, invocarían otra función que envía los comandos necesarios por el puerto serial para ser procesado por arduino.

La figura 69 muestra el código realizado en python para inicializar la librería Tkinter y enlazar los eventos a las funciones propuestas:

**Figura 69. Inicialización tkinter y enlace de eventos a funciones:**

```
root = tk.Tk()
print "Presione escape para salir:"
root.bind_all('<Key>', keypress)
root.bind_all ('<KeyRelease>', keyrelease)
# Para no mostrar la ventana de Tk
root.withdraw()
root.mainloop()
```

Fuente: Autores.

La primera línea del código de la figura 67 inicializa el módulo Tkinter, la línea 2 imprime un mensaje en consola a modo explicativo, la línea 3 y 4 enlaza los eventos “key” y “keyrelease” a las funciones llamadas keypress y keyrelease respectivamente, la línea 7 evita que aparezca la ventana GUI de tkinter y finalmente la línea 8 inicializa el loop principal de la aplicación.

La función invocada en cada evento, recibe como parámetro un objeto con propiedades específicas dependiendo del tipo de evento. Para nuestro caso, la función llamada recibe un objeto con la propiedad que nos interesa que se llama “char” que contiene a modo cadena de texto el nombre de la tecla que hemos presionado o soltado.

La función llamada por el evento “key”, tecla presionada, se muestra en la figura 70.

Esta función evalúa el valor de la propiedad “char” del evento y dependiendo de ella, llama la función move de la clase servo escrita por Brian D. Wendt, la cual envía datos por el puerto serial para ser interpretados por arduino.

La clase servo es definida en un archivo aparte y es incluida al frame principal mediante importación.

La figura 71 muestra la clase servo con sus métodos y propiedades.

Figura 70. Función llamada al presionar una tecla.

```
def keypress(event):
    if event.keysym == 'Escape':
        root.destroy()
    x = event.char
    #teclas hombros:

    if x == "q": #mueva hombro 1 arriba
        print "Moviendo hombro 1 arriba"
        servo.move(1,0);

    elif x == "w": #mueva hombro 1 abajo
        print "moviendo hombro 1 abajo"
        servo.move(1,180);

    elif x == "e": #mueva hombro 2 arriba
        print "moviendo hombro 2 arriba"
        servo.move(3,0);

    elif x == "r": #mueva hombro 2 abajo
        print "moviendo hombro 2 abajo"
        servo.move(3,180);

    elif x == "t": #mueva hombro 3 arriba
        print "moviendo hombro 3 arriba"
        servo.move(5,0);

    elif x == "y": #mueva hombro 3 abajo
        print "moviendo hombro 3 abajo"
        servo.move(5,180);

    elif x == "u": #mueva hombro 4 arriba
        print "moviendo hombro 4 arriba"
        servo.move(7,0);

    elif x == "i": #mueva hombro 4 abajo
        print "moviendo hombro 4 abajo"
        servo.move(7,180);

    elif x == "o": #mueva hombro 5 arriba
        print "moviendo hombro 5 arriba"
```

Fuente: Autores

**Figura 71. Clase servo comunicación serial Arduino - Python**

```
import serial

usbport = 'COM10'

# Set up serial baud rate
ser = serial.Serial(usbport, 9600, timeout=1)

def move(servo, angle):
    |
    if (0 <= angle <= 180):
        ser.write(chr(255))
        ser.write(chr(servo))
        ser.write(chr(angle))
    else:
        print "El angulo del servo debe ser entre 0 y 180.\n"
```

Fuente: Brian D. Wendt

El método `move` de la clase `servo` recibe 2 parámetros, `servo` y `angle`. El primero indicará que `servo` se moverá y el segundo indicará a que ángulo deberá moverse. Si el ángulo del servo se encuentra entre  $0^\circ$  y  $180^\circ$  la función escribirá en el puerto serial el carácter que corresponda al código ASCII 255, No. de servo y No. de grados, esta conversión a carácter ASCII es necesaria ya que arduino lee los datos del puerto serial como valores numéricos, por ejemplo, el carácter "a", arduino lo leerá como el número 147, ya que este es su valor en código ASCII.

El envío del primer parámetro "255" se hace necesario para delimitar e informar a arduino que los datos enviados corresponden a instrucciones para movimiento de nuestros actuadores, es así como serán enviados paquetes de datos de tres en tres delimitándolos por el dato inicial 255. Este método evita la malinterpretación de datos cuando se haya perdido alguno de los valores, ya sea del número de servo a mover o del ángulo al cual debe moverse.

Para su funcionamiento, la clase servo debe ser configurada con su propiedad *usbport*, donde se debe escribir el nombre del puerto al que enviará los datos seriales. En la tabla 10, se muestran algunos valores de ejemplo comunes de configuración para diferentes plataformas.

**Tabla 10. Valores ejemplo propiedad usbport librería servo.**

	Windows	MacOSX	Linux
usbport	COM3	/dev/tty.usbserial- FTALLOK2	/dev/ttyUSB0

Fuente: Brian D. Wendt.

Este valor depende del puerto usado por arduino o el dispositivo conectado al puerto serial.

**6.2.2 Interpretación de los datos por arduino.** El programa en arduino se basa en la librería servo del propio arduino, la cual permite crear clases de tipo servo que contienen métodos y propiedades útiles para control preciso de servomotores.

Uno de los métodos de interés es “servo.write(angle)”, el cual mueve el servo a una posición dada en su parámetro “angle”.

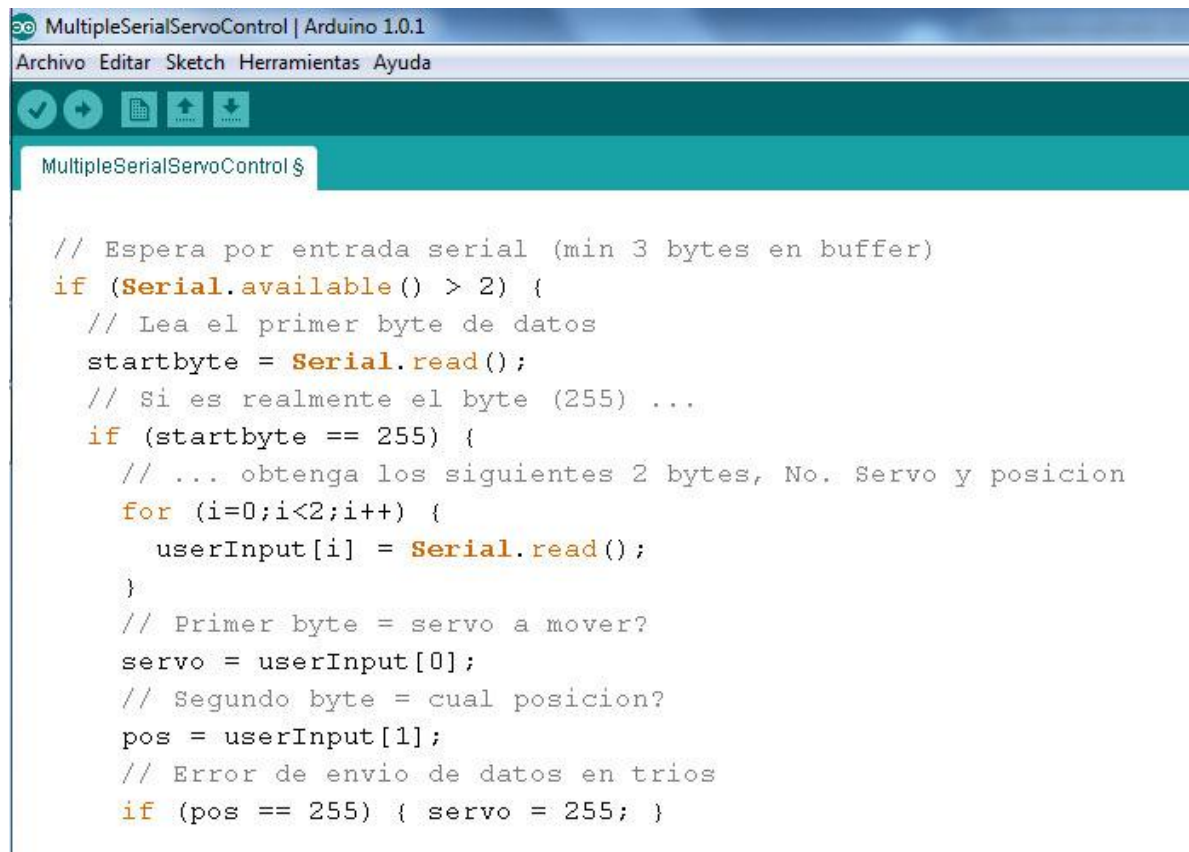
Cada servo puede ser instanciado creando una variable de tipo Servo, y se especifica el pin por el cual estará conectado a través del método “attach(pin)”, donde “pin” es el número del pin digital al cual se ha conectado el cable de control del servomotor.

El programa en arduino verifica la disponibilidad de más de dos bytes de datos en el puerto serial y valida que el primer carácter de esta secuencia de datos sea el

carácter ASCII 255 y así corroborar que los datos siguientes sean el valor del número de servo a mover y del ángulo a ubicar.

La figura 72 muestra el código usado para validación de datos del puerto serial. La acción tomada por estos datos se muestra en la figura 73, donde con una sentencia switch se sabe el servo a mover para enviarle el valor del ángulo deseado.

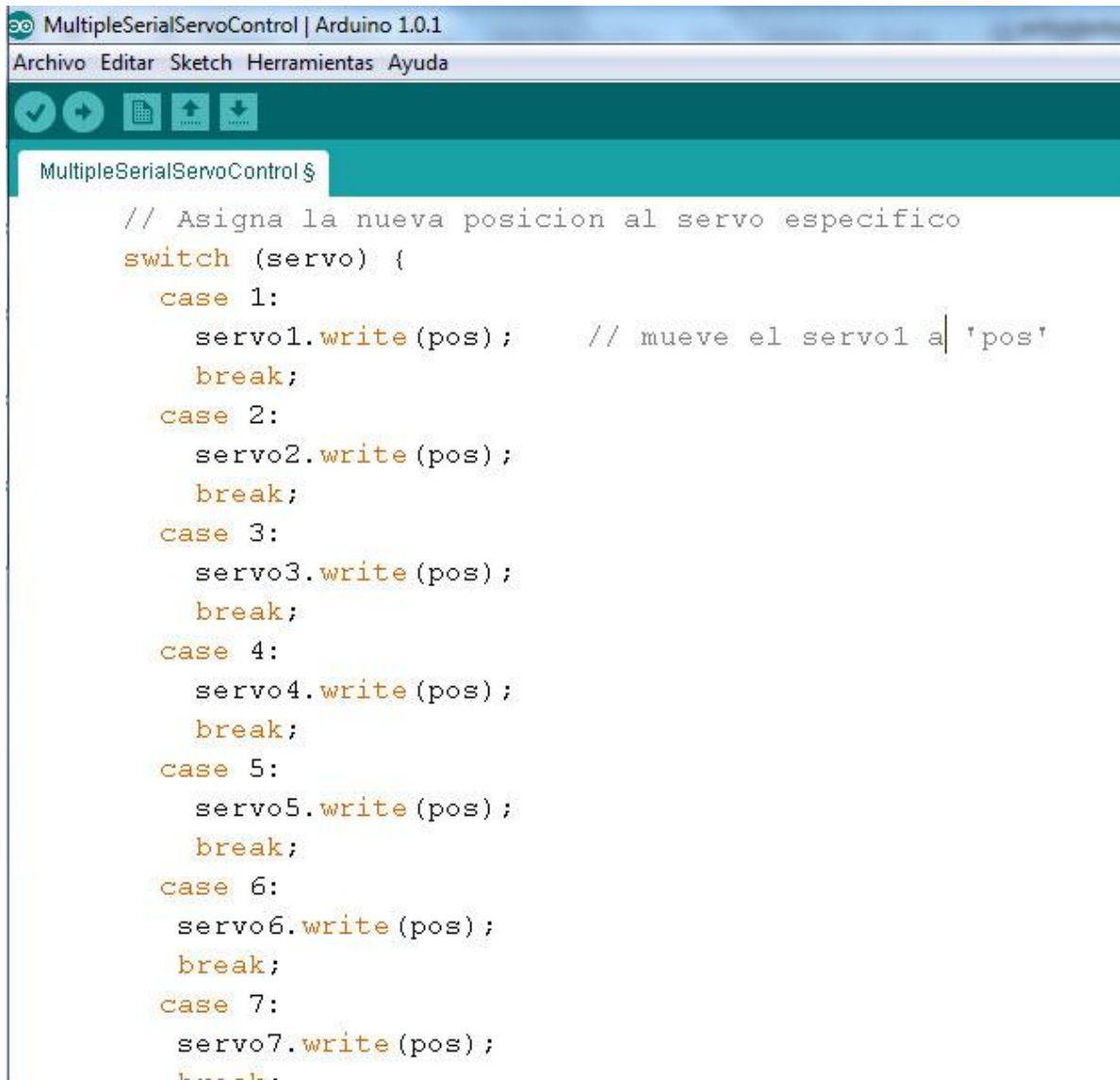
**Figura 72. Procesamiento datos entrada serial.**

The image shows a screenshot of the Arduino IDE interface. The title bar reads "MultipleSerialServoControl | Arduino 1.0.1". The menu bar includes "Archivo", "Editar", "Sketch", "Herramientas", and "Ayuda". Below the menu bar is a toolbar with icons for saving, running, and uploading. The main editor area shows the following code:

```
MultipleSerialServoControl $  
  
// Espera por entrada serial (min 3 bytes en buffer)  
if (Serial.available() > 2) {  
  // Lea el primer byte de datos  
  startbyte = Serial.read();  
  // Si es realmente el byte (255) ...  
  if (startbyte == 255) {  
    // ... obtenga los siguientes 2 bytes, No. Servo y posicion  
    for (i=0;i<2;i++) {  
      userInput[i] = Serial.read();  
    }  
    // Primer byte = servo a mover?  
    servo = userInput[0];  
    // Segundo byte = cual posicion?  
    pos = userInput[1];  
    // Error de envio de datos en trios  
    if (pos == 255) { servo = 255; }
```

Fuente: Autores.

**Figura 73. Acciones tomadas al tener procesados los datos del puerto serial**

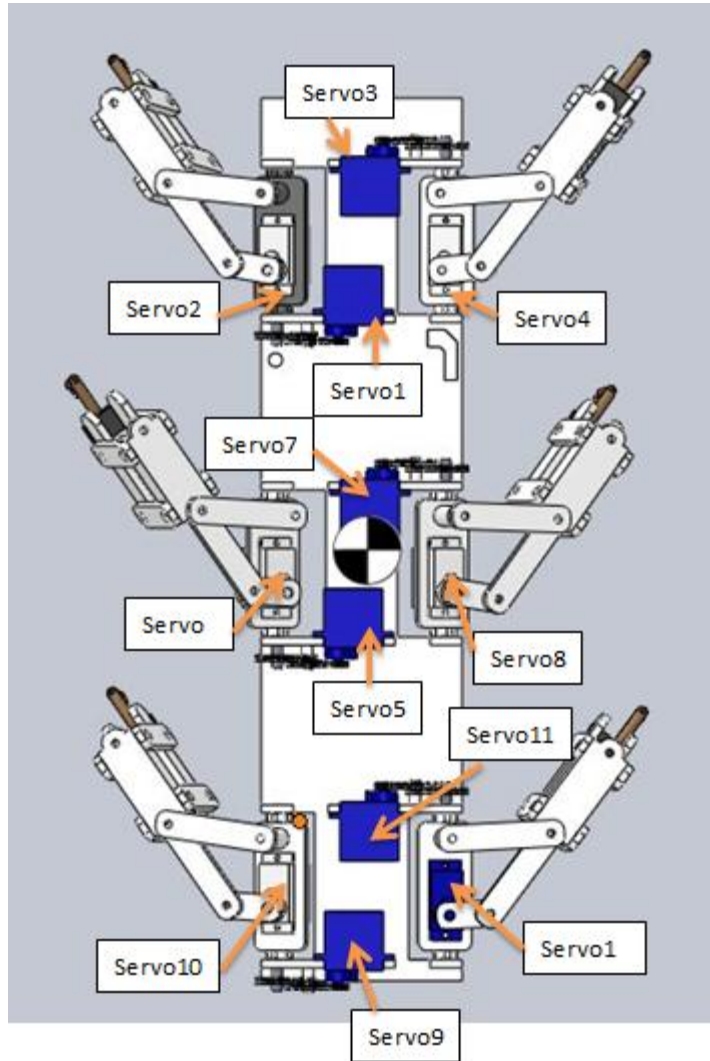


```
MultipleSerialServoControl | Arduino 1.0.1
Archivo Editar Sketch Herramientas Ayuda
MultipleSerialServoControl $
// Asigna la nueva posicion al servo especifico
switch (servo) {
  case 1:
    servo1.write(pos); // mueve el servo1 a 'pos'
    break;
  case 2:
    servo2.write(pos);
    break;
  case 3:
    servo3.write(pos);
    break;
  case 4:
    servo4.write(pos);
    break;
  case 5:
    servo5.write(pos);
    break;
  case 6:
    servo6.write(pos);
    break;
  case 7:
    servo7.write(pos);
    break;
}
```

Fuente: Autores.

La figura 74 muestra la asignación de número de servo en el pre prototipo, la tabla 11 da a conocer los pines digitales de control correspondientes a cada servo.

Figura 74. Nomenclatura conexión servos.



Fuente: Autores

**Tabla 11. Pines digitales arduino, teclas y funciones asignados a cada servomotor.**

Característica No. servo	1	2	3	4	5	6	7	8	9	10	11	12
Pin	1	2	3	4	5	6	7	8	9	10	11	12
Función	H	P	H	P	H	P	H	P	H	P	H	P
Tecla subida /avance	q	a	e	d	t	g	u	j	o	l	z	c

Tecla bajada/retroceso	w	s	r	f	y	h	i	k	p	{	x	v
---------------------------	---	---	---	---	---	---	---	---	---	---	---	---

H- servo de hombro            P – servo de Pata

Fuente: Autores

La configuración de actuadores del hombro para el pre prototipo hizo necesario convertir estos servomotores en actuadores de rotación continua con control de giro y velocidad, proceso llamado trucado de servos.

El trucado de servos consiste en eliminar los topes mecánicos que posee el eje de rotación del servomotor y reemplazar la retroalimentación de este (potenciómetro) por un par de resistencias fijas de valores cercanos a la resistencia total del potenciómetro. De este modo, el error tomado por el controlador del servomotor será el mismo y por lo tanto continuará su giro.

Cada servomotor de hombro fue trucado y caracterizado para hallar su centro. El centro se refiere al valor de ángulo dado por el método `servo.write(angle)` para que el eje se mantuviera quieto. Ya hallado este valor, un valor mayor o menor que este hará mover el eje en un sentido o en otro.

Por la ubicación de los servomotores de los hombros se verificó que un valor por encima de este centro hacia bajar el hombro y un valor por debajo lo haría subir.

Ya que el control propio del servomotor funciona como un control PD, un error mayor hará bajar o subir el hombro con mayor velocidad de giro. Con este concepto se obtuvo que al enviar al servo un ángulo de  $0^\circ$  y  $180^\circ$  se obtenía la máxima velocidad de rotación para un lado y para otro.

En primera instancia se realizó un control seco para posicionar los hombros, asignando para cada hombro una tecla de subida y otra de bajada y configurando con el evento "KeyRelease" para los servos trucados que al soltarla enviará el valor central haciendo que este se detenga.

**6.2.3 Control inasistido.** El control semiautónomo o inasistido para el pre prototipo se basó en sensores de posición angular de hombro y pata que retroalimentaban al controlador la posición de forma analógica.

Se eligieron potenciómetros logarítmicos como sensores para retroalimentar la posición angular tanto de hombro como de pata ya que este tipo de dispositivos son económicos y están disponibles en presentaciones ideales para acople con los ejes usados en el pre prototipo.

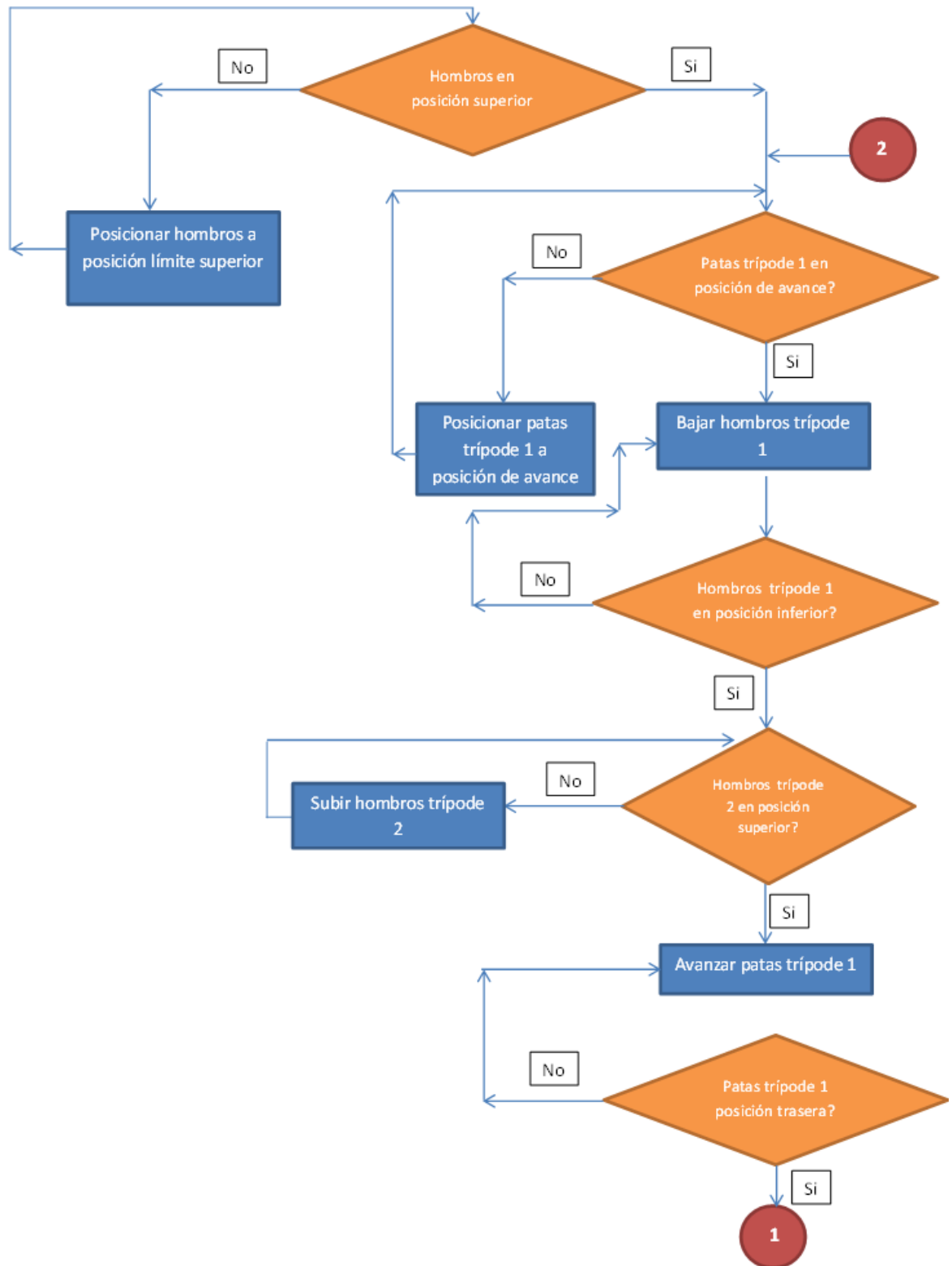
Este tipo de potenciómetros posee una relación logarítmica entre ángulo de eje vs resistencia y además posee un rango de giro limitado aproximadamente de  $180^\circ$ . Para el conocimiento de la posición del hombro este tipo de potenciómetros funciona correctamente acoplado directamente al eje de este, ya que el rango de giro del hombro del pre prototipo es  $[0^\circ, 180^\circ]$ .

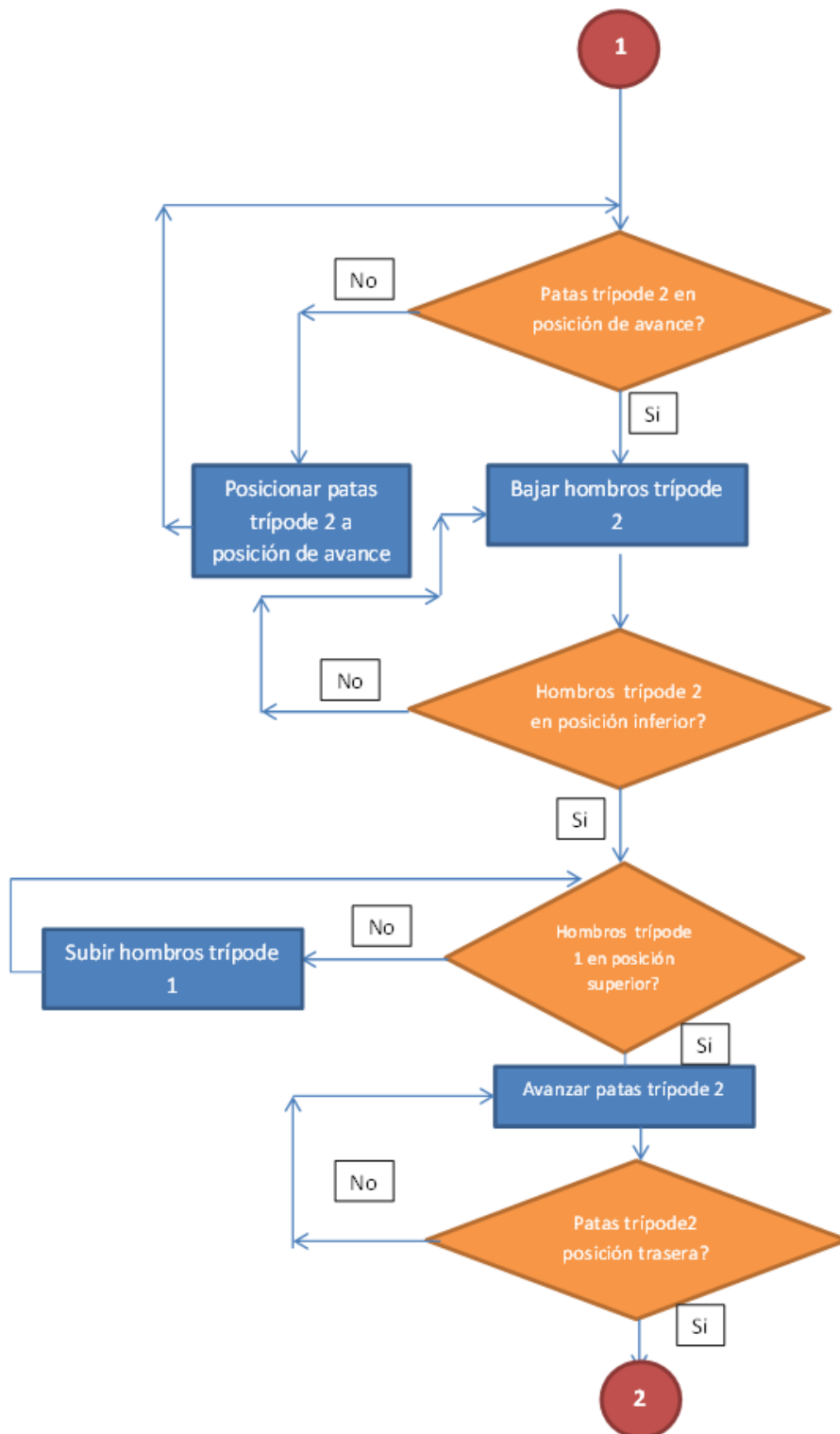
Un potenciómetro ubicado en el eje de la barra manivela era ideal para sensor la posición del punto P del mecanismo de Hoekens, sin embargo, ya que esta

manivela gira continuamente, este tipo de sensores no funcionaria. Para solucionar este inconveniente, se diseñó un sistema de sensado en el eje de la barra balancín que junto con la información del sentido de giro del motor podrían caracterizar en cualquier momento la posición del punto P.

Ya que el objetivo del pre prototipo es brindar información a grandes rasgos acerca de los problemas que puede presentar la planta en estudio cuando escala, el control inasistido fue programado para realizar una secuencia completa usando los sensores de posición de pata y hombro a modo final de carrera, usando los valores extremos como referencia para realizar las diversas tareas de la secuencia de subida, esta secuencia se explica en la figura 75.

Figura 75. Diagrama de bloques de secuencia de escalada





Fuente: Autores

**6.2.4 Resistencia al flujo de corriente eléctrica de materiales.** La sección transversal, longitud y temperatura tienen efectos sobre la cantidad de resistencia que un cable o material puede imponer al paso de la corriente eléctrica, a su vez, el material repercute en gran manera esta resistencia ya que algunos metales son más conductores eléctricos que otros.

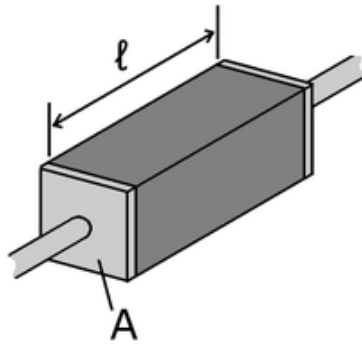
Para el cálculo de la resistencia que impone cierto material al flujo de electrones, nos basamos en el concepto de resistividad o resistencia eléctrica específica, la cual determina que tanto se opone dicho material al flujo de una corriente eléctrica.

Para un material con sección transversal constante y un flujo de corriente uniforme la resistividad eléctrica se define como:

$$\rho = R * \frac{A}{L}$$

Donde R es la resistencia eléctrica del material, A es el área de la sección transversal de dicho material y L es su longitud tal como se aprecia en la figura 76. Si la resistencia está dada en  $\Omega$ , el área en  $m^2$  y la longitud en  $m$ , la resistividad tendrá unidades de  $\Omega * m$

**Figura 76. Nomenclatura resistividad material**



Fuente: Omegatron. A somewhat cartoonish diagram of the geometry of the resistivity equation (Online), Febrero 2007 [Cited April 20 2013]. Available from internet: [http://en.wikipedia.org/wiki/Electrical\\_resistivity\\_and\\_conductivity](http://en.wikipedia.org/wiki/Electrical_resistivity_and_conductivity)

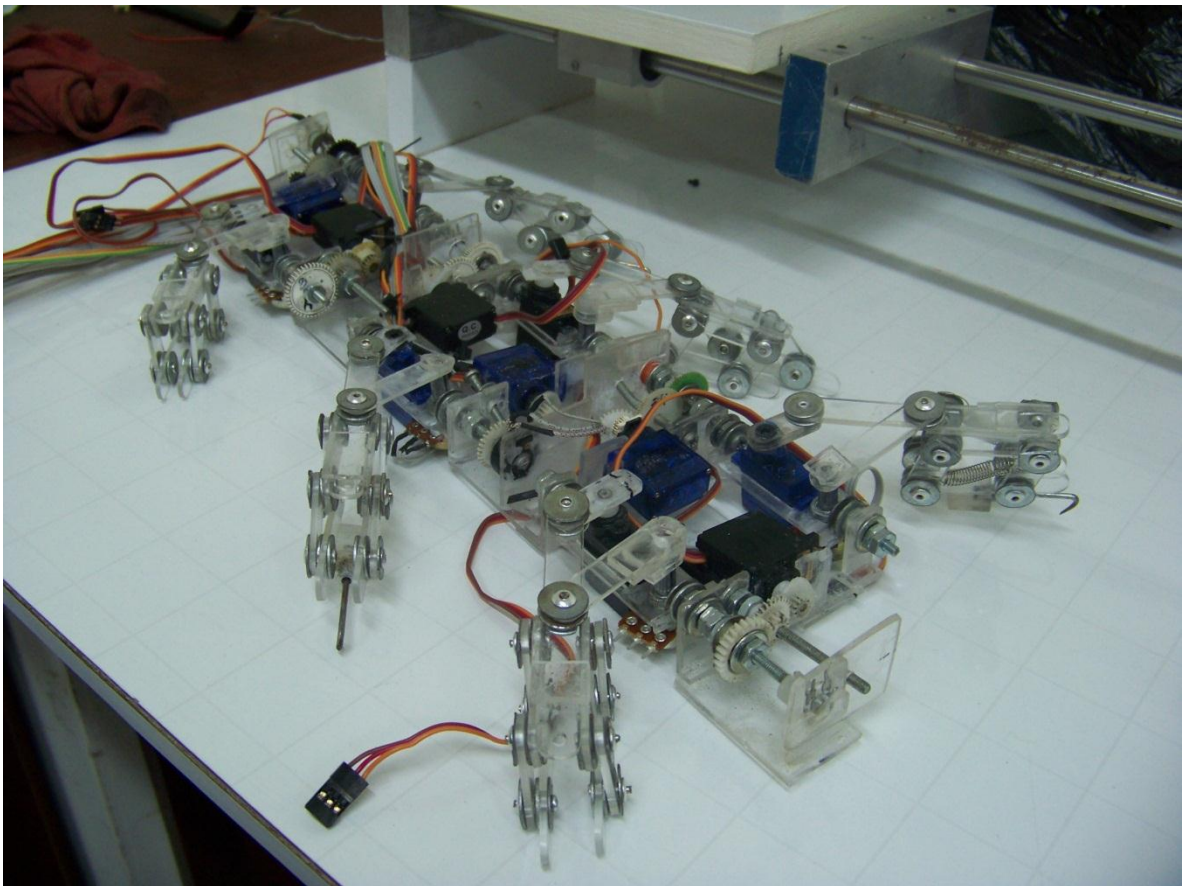
La razón de ser del concepto resistividad es que es una propiedad del material a diferencia de la resistencia.

## 7. EXPERIENCIAS DEL PREPROTOTIPO Y REDISEÑO DEL MISMO

Una vez construido el preprototipo, se pudo probar la planta por medio de control remoto usando el puerto serial como comunicación entre el computador y la arduino, el objetivo de las pruebas se logró a cabalidad ya que proporcionó información valiosa acerca del comportamiento de la planta y de los problemas que se presentan tanto en la construcción como en el ensamble.

En la figura 77 se puede observar el preprototipo finalizado con sus respectivos componentes.

**Figura 77. Preprototipo finalizado.**



Fuente: Autores

## 7.1 PRUEBAS REALIZADAS AL PREPROTOTIPO

Las pruebas que se le realizaron al preprototipo consistían principalmente en pruebas estáticas y de resistencia en las que se pudieron determinar los componentes críticos de una manera experimental.

Se pudo corroborar que los componentes críticos se encontraban en la barra balancín, los apoyos de los hombros y los engranajes (como era de esperarse). Los engranajes presentaron dos problemas significativos los cuales fueron, la ruptura de los dientes y la desalineación entre ellos lo que hacía que se desengranaran en ciertos puntos de la rotación, esto debido al proceso de ensamble y al presupuesto utilizado para el preprototipo.

**7.1.1 Pruebas en el tronco.** En el tronco se hicieron pruebas de posicionamiento de las patas, torque de los servomotores, agarre a la superficie, penetración de las garras entre otras.

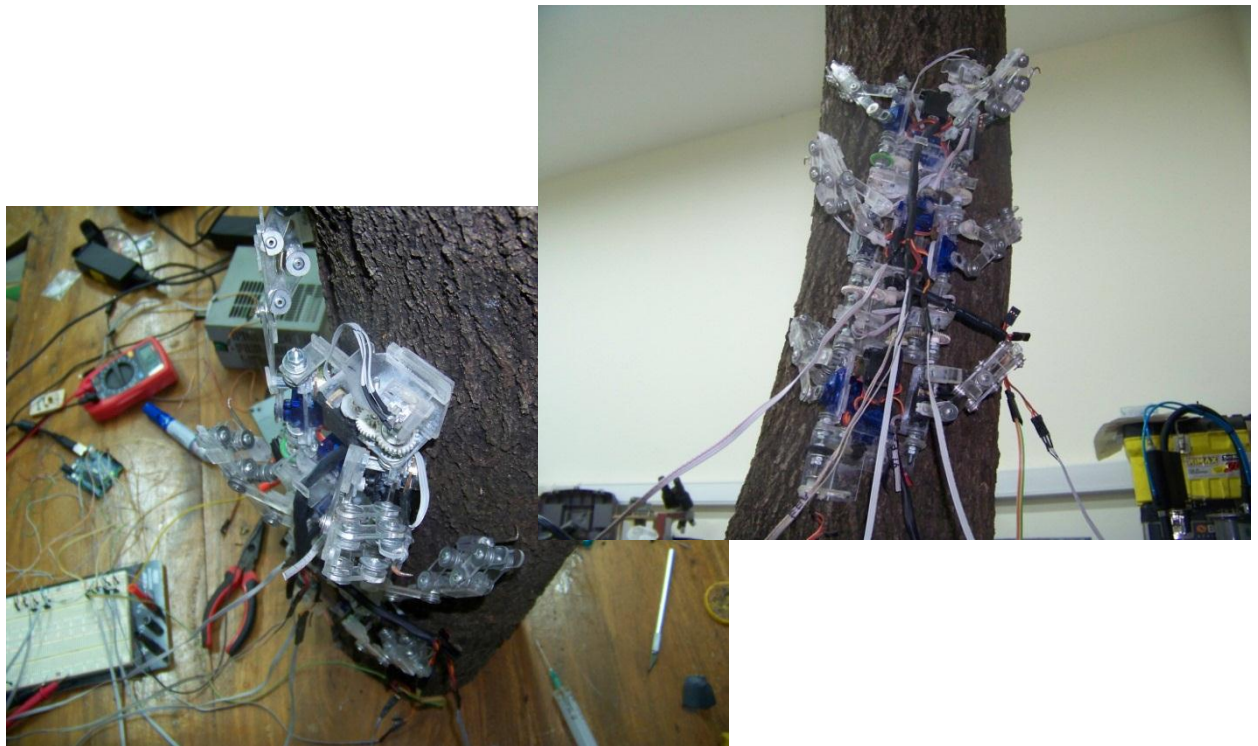
Debido a que los servomotores para este preprototipo no fueron elegidos por su torque si no por el bajo coste y tamaño para poder observar el comportamiento del diseño del preprototipo era de esperarse que no cumplieran con el torque necesario para avanzar, sin embargo proporciono información en cuanto al torque que se puede producir cuando el motor esta energizado y el torque cuando el motor esta desenergizado lo que hay que tener en cuenta a la hora de la elaboración del código en arduino.

En cuanto al movimiento de los hombros, se observó que debía ampliarse el rango de movimiento para que de esta forma todas las patas puedan hacer contacto con la superficie del árbol, ya que en diámetros pequeños de tronco esta labor se dificulta debido al poco rango mecánico de algunos hombros.

Mediante las pruebas en el tronco se pudo corroborar la estabilidad que puede tener el robot mientras se agarra al árbol en un trípode lo que permite seguir con la Generación de gaitos discontinuos con avances en trípodes como secuencia de locomoción principal.

A continuación en la figura 78 se observa el preprototipo agarrado al árbol en una configuración de trípode.

**Figura 78. Preprototipo configuración trípode**



Fuente: Autores.

De todas estas observaciones surgieron ideas para el rediseño del prototipo para su construcción final, en la tabla 12 se especificaran las dificultades que existían y sus respectivas modificaciones.

**Tabla 12. Modificaciones del preprototipo.**

Elemento o conjunto	Preprototipo	Prototipo
<b>Material</b>	Acrílico: este material posee un bajo peso lo cual lo hace idóneo para aplicaciones como las de escalar, pero debido a su alta fragilidad, los componentes del mecanismo se fracturan fácilmente al igual que los pegues se sueltan fácilmente	Aluminio: Aunque es un poco más pesado que el acrílico, su resistencia es mucho más elevada al igual que las uniones son más resistentes debido a la posibilidad de realizar soldaduras lo que permite absorber impactos sin que estos elementos fallen tempranamente
<b>Potenciómetros</b>	Logarítmicos: Estos potenciómetros no permiten un posicionamiento preciso debido a que su resistencia varia de forma no lineal lo que dificulta su sintonización.	Lineales: Se decidió emplear este tipo de potenciómetros debido a su mayor precisión de posicionamiento y fácil sintonización
<b>Acoples</b>	Cianocrilato de metilo: Los acoples entre potenciómetros y mecanismos se realizaron con pegamento y calor esto no permitía un buen alineamiento además se despegaban	Bujes y prisioneros: Para un mejor alineamiento se mecanizaron unos bujes con el diámetro interno igual al del elemento a acoplar y con machuelos se hizo la rosca para el tornillo prisionero que

	con facilidad y descalibraban la sintonización. No permiten desensamble	fijaría el acople con el potenciómetro.
<b>Conexiones</b>	<p>Protoboard: las conexiones se realizaron en una protoboard externa al robot, lo que implicaba cables largos que propiciaban errores de conexión y baja confiabilidad debido a los malos contactos que hacen los cables en la protoboard</p>	<p>PCB: Para evitar las conexiones tediosas y los malos contactos que limitan el buen funcionamiento del robot, se diseñaron 3 pcb las cuales se acoplan entre si y con la Arduino, proporcionando las conexiones necesarias para el funcionamiento del robot evitando problemas de desconexión.</p>
<b>Actuadores</b>	<p><b>Servomotores:</b> Se utilizaron servomotores de bajo presupuesto los cuales proporcionan un torque de 1.6 kg.cm y poseen una caja de reducción con engranajes de plástico, los cuales fueron trucados para permitir un giro de 360 grados que dificultaba su</p>	<p><b>Micromotores:</b> Estos son motores DC de alto torque debido a que poseen una reducción de 298:1, esta reducción es realizada con un tren de engranajes miniatura de dientes metálicos, se optó por estos motores ya que tienen un torque de 6 Kg.cm y pueden ser</p>

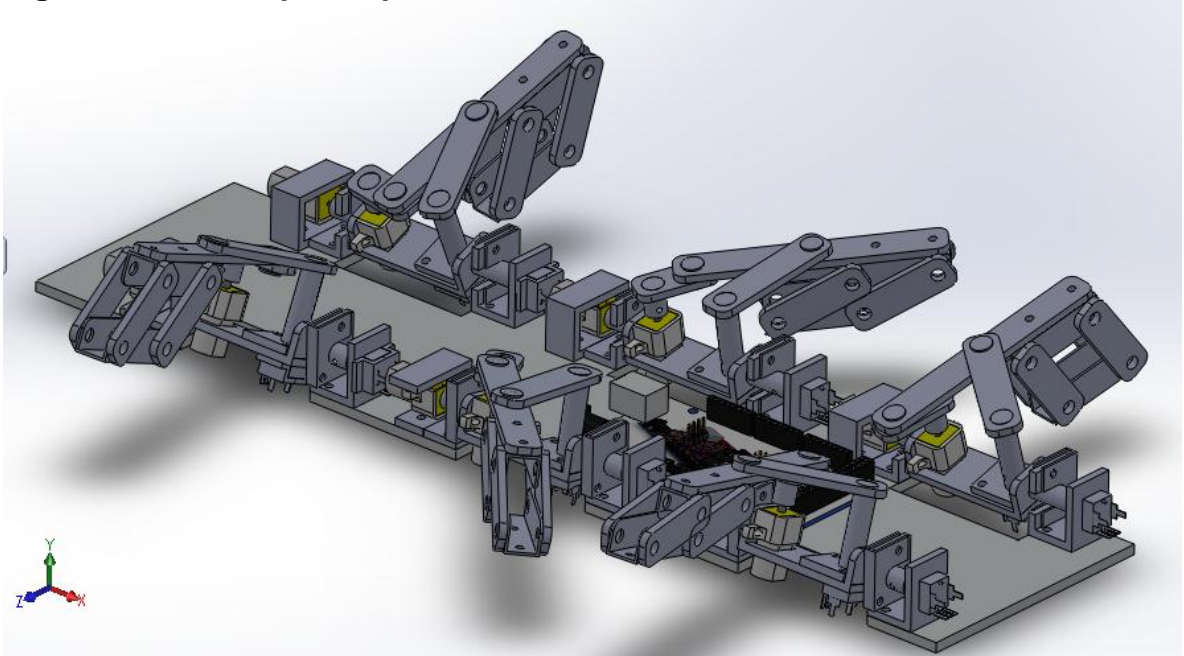
	detención ya que se requiere un valor de pulsos exacto para mantenerlo quieto, y estos a veces son alterados por corrientes inducidas.	fácilmente controlados con un IC tipo puente H y se elimina el problema de vibración cuando se encuentran quietos ya que no se hace por pulsos
<b>Diseño y Ensamble</b>	No desensamblable: en este diseño los servomotores iban pegados a la estructura de acrílico al igual que la mayoría de sus partes, los pasadores fueron hechos a partir de remaches y los acoples con pegamento, lo que hace imposible el mantenimiento y su desensamblable.	Desensamblable: Con la experiencia del preprototipo, se realizó un diseño que tenía en cuenta los aspectos para el desensamble, todos los acoples se realizan con tornillos y prisioneros y los apoyos tienen cortes que facilita sacar cada uno de los componentes.

Fuente: Autores

## 7.2 DISEÑO CAD EN SOLIDWORKS

Con las observaciones realizadas en el preprototipo y teniendo en cuenta los inconvenientes del mismo se hizo un nuevo diseño (Figura 79) que serviría para generar los planos de construcción que son anexados al final de este libro.

**Figura 79. Diseño prototipo final**



Fuente: Autores

**7.2.1 Selección de platinas para mecanismo de Hoekens.** Para el diseño de las barras del mecanismo de Hoekens, se eligieron unas platinas de aluminio de 13 milímetros de ancho por 3 milímetros de espesor, las cuales brinda el mercado local a un bajo costo. El ancho se seleccionó debido a que se querían hacer las barras con un ancho moderado con fines estéticos además que el peso juega factor importante para esta aplicación (se eligieron las platinas con el mínimo ancho ofrecido por el mercado).

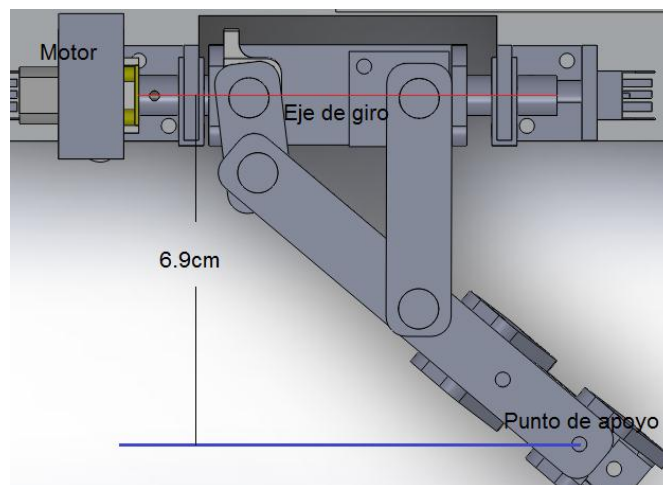
El cálculo del espesor de la platina se realizó teniendo en cuenta la carga máxima a la cual se sometería el mecanismo de barras la cual se da en el momento de agarre al tronco.

Los cálculos para la determinación del espesor serán mostrados a continuación con sus respectivas especificaciones.

**7.2.1.1 Cálculo de la fuerza en el punto de agarre.** Para poder calcular esta fuerza, se debe tomar la condición crítica que es cuando se está realizando el torque máximo sobre la superficie del árbol, este torque es proporcionado por el micromotor ubicado en el hombro, cuyo torque máximo es de 6 Kg.cm.

Para calcular la fuerza de reacción que actúa sobre la barra que hace apoyo con el árbol debe conocerse la distancia perpendicular entre el punto de apoyo y el eje en el que gira el motor. En la figura 80 se puede observar más detalladamente la distancia a la que se está haciendo referencia.

**Figura 80. Distancia para cálculo de fuerza crítica**



Fuente: Autores.

La distancia mostrada en la figura 80 permanece constante independientemente del ángulo en el que se encuentre el hombro, ya que este es un mecanismo que se encuentra en el mismo plano y todo este va girando a medida que el hombro lo hace. El mecanismo está diseñado para que siempre haga contacto dentro de la

línea recta de la trayectoria lo que también hace a esta distancia independiente del ángulo de giro de la barra manivela (dentro del rango de la línea recta).

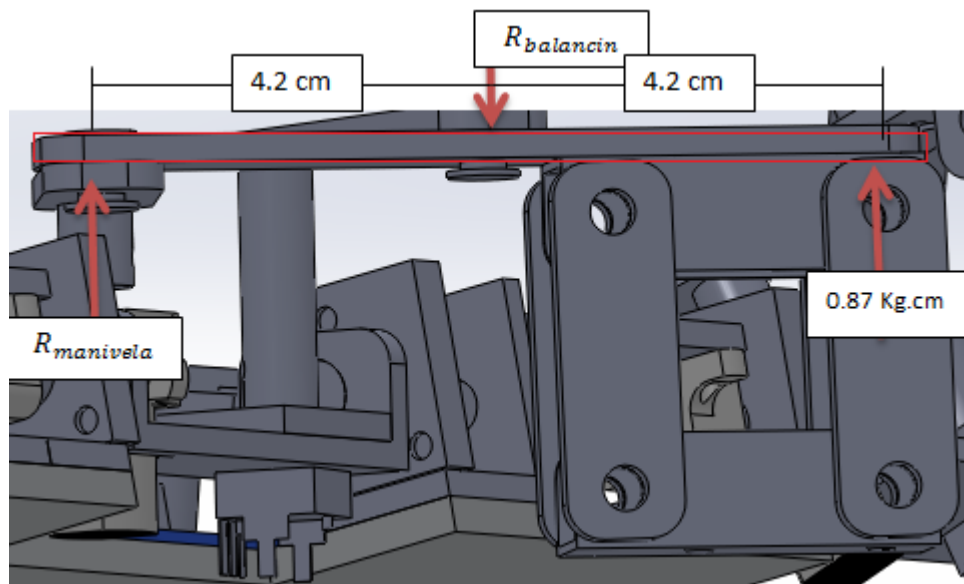
Para la fuerza de reacción tenemos:

$$T = F * d$$
$$6 [Kg.cm] = F * 6.9 [cm]$$
$$F = 0.87 Kg$$

Para el siguiente análisis de resistencia se toma que las barras del mecanismo se encuentran a flexión pura y se procede a encontrar las reacciones en cada una de las barras a analizar.

En la figura 81 se observan las reacciones presentes en la barra de avance.

**Figura 81. Reacciones en barra de avance**



Fuente: Autores

Aplicando estática se tiene:

$$\sum F_y = 0$$

$$R_{manivela} - R_{balancin} + 0.87 = 0$$

$$\sum M_{balancin} = 0$$

$$-R_{manivela} * 4.2 + 0.87 * 4.2 = 0$$

$$R_{manivela} = 0.87 \text{ Kg}$$

$$R_{balancin} = 1.74 \text{ Kg}$$

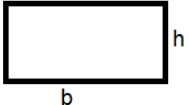
De las ecuaciones de resistencia según Berlusconi-Navier quien planteo la siguiente hipótesis: **“Las secciones planas y perpendiculares al eje de la viga antes de la deformación, siguen siendo planas y perpendiculares al eje de la viga después de la deformación”**.

Con la que fue posible deducir la siguiente ecuación:

$$\sigma_{max} = \frac{M}{W}$$

En donde M es el momento Flector máximo y W es el modulo resistente, el cual depende de la geometría de la viga (momento de inercia)

Para perfiles rectangulares El modulo resistente se calcula de la siguiente forma:



$$W = \frac{1}{6} b * h^2$$

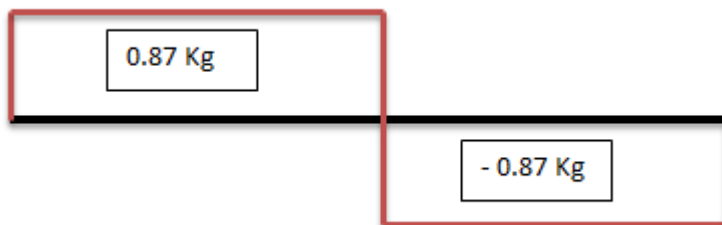
Donde b para nuestro caso es 13 milímetros ya fijo y se buscara el espesor de la platina para que soporte las cargas de trabajo.

- **Análisis barra de avance**

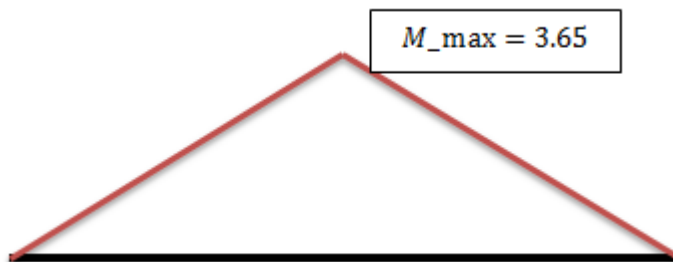
### Diagrama de fuerzas



### Diagrama de cortante



### Diagrama de momentos

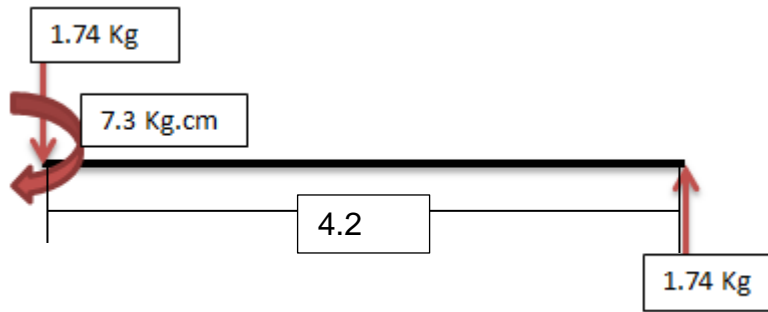


Una vez calculado el momento máximo, se obtiene el esfuerzo máximo para esta viga. Este esfuerzo máximo debe igualarse con el esfuerzo admisible del aluminio que son  $150 \text{ Mpa}$  los cuales equivalen a  $1530,6 \text{ Kg/cm}^2$ , de la formula se tiene:

$$1530,6 = 3.65 \text{ Kg.cm} * \frac{6}{h^2 * 1.3 \text{ cm}}$$

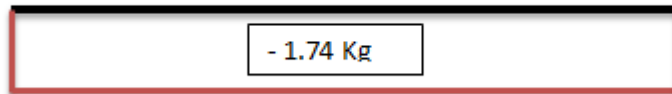
$h = 0,1$  aplicando factor de seguridad de 50%  $h = 0,15 \text{ cm}$

### ANALISIS DE LA BARRA BALANCIN

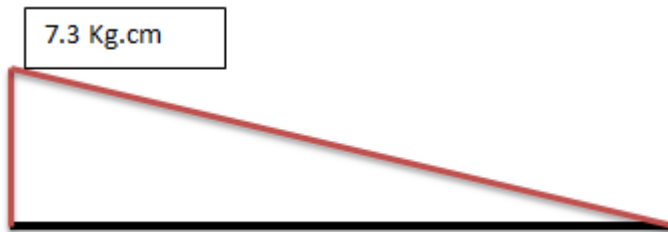


Los datos de la barra balancín salen de una estática sencilla haciendo sumatoria de fuerzas en y de momentos en el primer apoyo.

### Diagrama de cortante



### Diagrama de momento



Igual que en el caso anterior se calcula el esfuerzo máximo y se compara con el admisible

$$1530,6 = 7.3 \text{ Kg.cm} * \frac{6}{h^2 * 1.3 \text{ cm}}$$

$h = 0.15$  y aplicando un factor de seguridad de 50%  $h = 0.225 \text{ cm}$

Debido a que esta barra es más crítica que la anterior, se opta por tomar esta medida del espesor como patrón de diseño. En el mercado no se encuentra una medida exacta de 0.225cm por lo que se compra la más cercana que este en el mercado la cual es de 0.3cm de espesor.

La barra manivela al ser más corta que las demás y presentar una fuerza de reacción menor a la de la barra balancín, su análisis no es necesario.

## **8. DETERMINACION DE EL TORQUE NECESARIO PARA ESCALAR Y SELECCIÓN DE MOTORES**

Con ayuda de los datos cualitativos obtenidos del preprototipo, el cual tenía un peso de 700 gramos y unos servomotores de 1.6 Kg.cm se pudo observar que se requieren motores de mayor torque, pero debe tenerse en cuenta que el peso de estos motores también afectan el peso del robot, por lo que se debe hacer un proceso iterativo en cuanto al torque suministrado por el motor y el peso del robot.

También se debe tener en cuenta que el material del prototipo final es aluminio (más denso que el acrílico), además el prototipo final involucrará más componentes tales como la PCB y todos los elementos electrónicos necesarios para el sistema de control (integrados con arquitectura puente H, reguladores de voltaje, resistencias entre otros.).

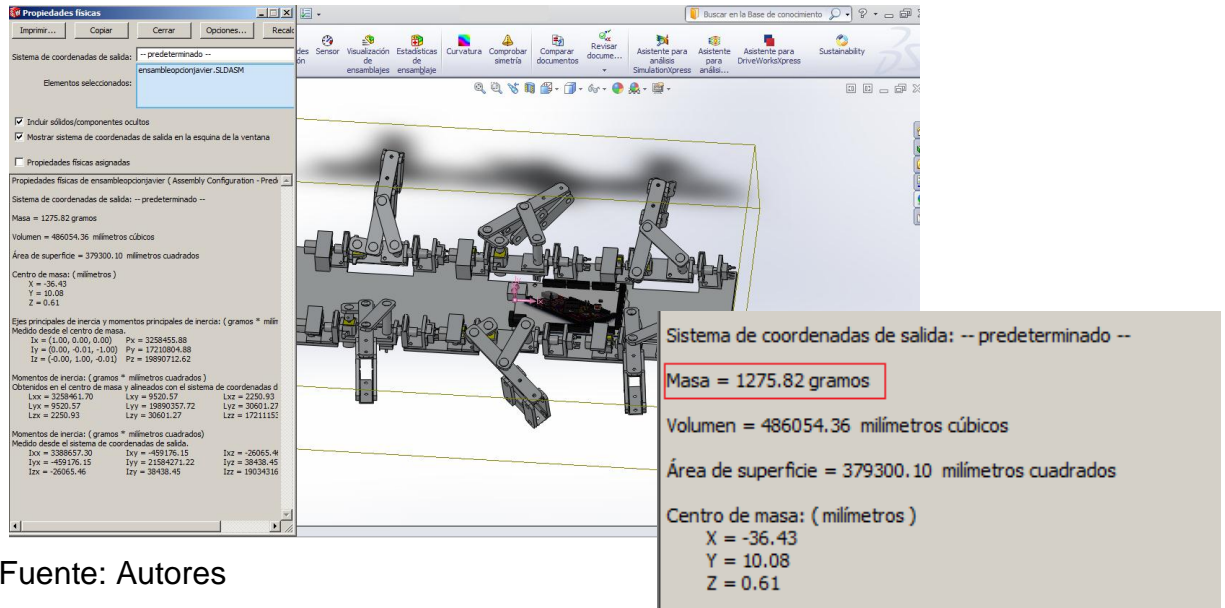
Todos estos factores suponen un valor de torque más elevado para el prototipo final, para realizar el diseño y cálculo del torque se buscó en el mercado el motor más pequeño con el más alto torque. Se encontraron unos micromotores los cuales consisten en un motor dc muy pequeño acoplado con una caja reductora de engranajes metalicos miniatura, estas cajas reductoras vienen con reducciones de 5:1, 30:1, 50:1, 100:1, 298:1 y 1000:1 con torques desde 0.1Kg.cm hasta 9Kg.cm.

Estos micromotores pesan en promedio 9 gramos. Se usará este peso promedio para tenerlo en cuenta en el peso del diseño que se realizó en solidworks.

Con ayuda de solidworks se procedio a calcular el peso del diseño, especificando en este los materiales del ensamble.

En la figura 82 se muestra una imagen con las propiedades físicas del ensamble.

**Figura 82. Propiedades físicas del diseño en solidworks**



Fuente: Autores

## 8.1 ANALISIS ESTATICO PARA LA CONDICION CRÍTICA

Para la condición crítica, debe tenerse en cuenta que el número mínimo de patas con las que avanzara el robot es 3 (trípodes), esto implica que las tres patas sean las encargadas de la estabilidad del robot repartiéndose entre ellas el valor del peso del robot.

La forma como se reparten estas fuerzas en los tres apoyos dependerá de la posición en las que las garras hagan contacto con el árbol.

Como se observó en el capítulo 6, donde se planteó la estática del preprototipo de forma general y para cualquier posible ubicación de las patas, determinar el valor de las reacciones de los tres apoyos en función de los ángulos del hombro y pata es una labor bastante compleja que requiere una mayor profundización en esta área.

A continuación se presentará un análisis estático de la posición crítica a la cual puede estar sometido el robot en base la figura 81.

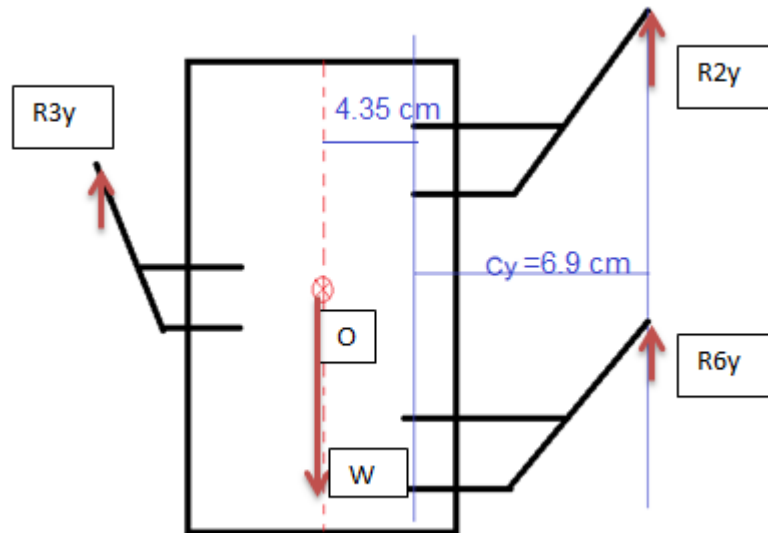
Partiendo de la premisa de que escalar a  $90^\circ$  demanda la mayor cantidad de torque por parte de los motores que proporcionarán el avance del robot, esto debido a que la totalidad del peso se está oponiendo al movimiento y no existen componentes del peso en otras direcciones.

En cada uno de los apoyos se generarán reacciones en los tres ejes, las cuales son desconocidas.

Para el análisis se tomará al robot como un cuerpo rígido ya que los motores deben ser capaces de mantenerlo estático (sin que este se descuelgue) por un tiempo mientras se realizan las fases de posicionamiento de patas propias de un gait discontinuo.

De la definición de cuerpo rígido: Cuerpo que no sufre deformaciones por efecto de fuerzas externas, es decir un sistema de partículas cuyas posiciones relativas no cambian, se deduce que las únicas reacciones presentes en el robot son las reacciones en el eje Y y en el eje Z debido a que no existen fuerzas externas que intenten desplazar al robot a lo largo del eje X.

**Figura 83. Esquema de la estática del trípode de avance**



Fuente: Autores

El momento crítico es cuando la pata que esta solitaria del trípode, se encuentra lo más cercana posible al centro de masas mientras que las otras dos patas están lo más alejadas posible del centro de masa (O).

De esta forma se tiene:

$$\sum F_y = 0$$

$$R_{2y} + R_{3y} + R_{6y} = 1.2 \text{ Kgf}$$

$$\sum M_{R_{6y}} = 0$$

$$1.2 * (4.35 + 6.9) - R_{3y} * (4.35 + 6.9 + 4,35) = 0$$

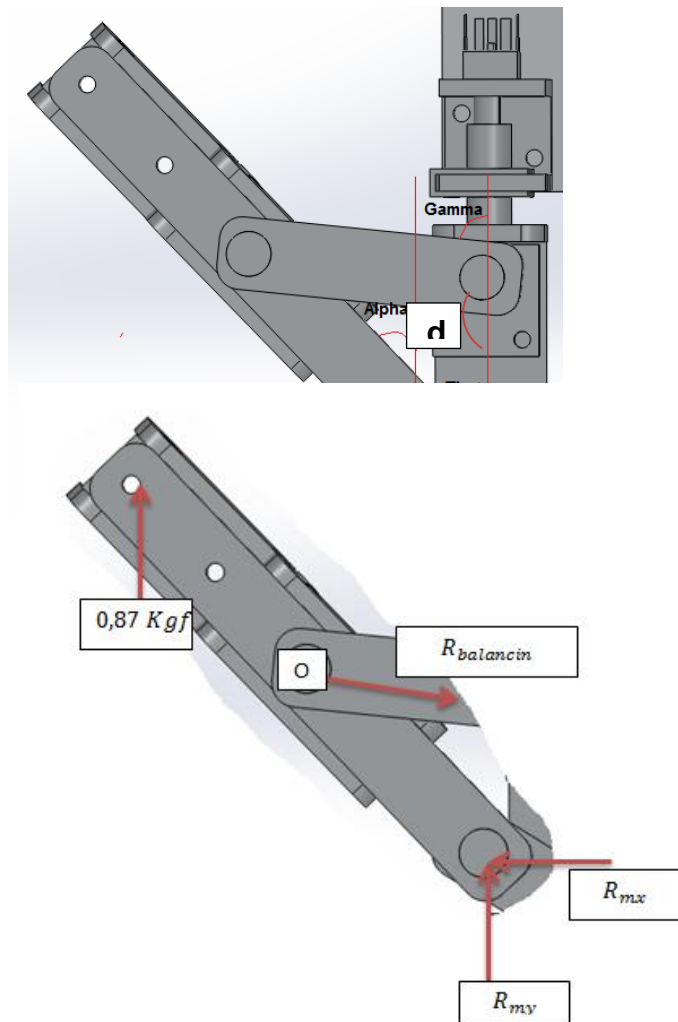
**Despejando  $R_{3y} = 0.87 \text{ Kgf}$**

Esta fuerza que está actuando en el punto de apoyo con el árbol debe transmitirse a lo largo del mecanismo de Hoekens para determinar su efecto torsional en el eje del motor de la pata.

Para esto se aislará la barra de avance (biela, Ver figura 84) y se encontrarán las reacciones de la barra manivela mediante un análisis estático, el cual toma como sistema de coordenadas un plano que gira con el hombro.

Consideraciones: Para este análisis se despreciara el peso de las patas

**Figura 84. Ángulos y aislamiento de la barra biela**



Fuente: Autores

$$\sum F_y = 0$$

$$0,87 + R_{my} - R_{balancin} * \cos(\gamma) = 0$$

$$\sum F_x = 0$$

$$R_{balancin} * \sin(\gamma) - R_{mx} = 0$$

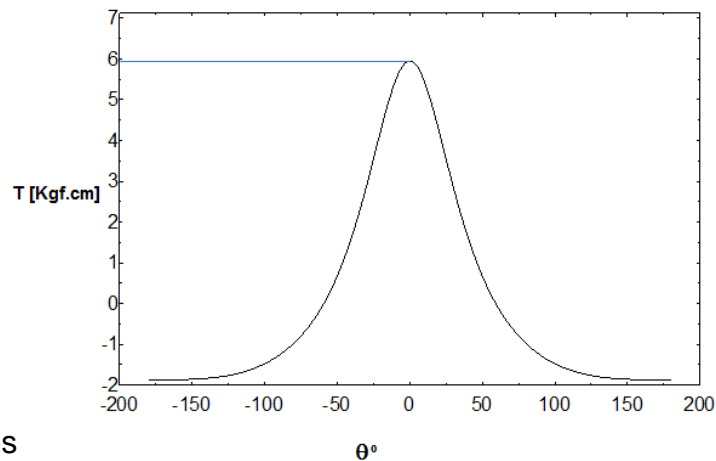
$$\sum M_o = 0$$

$$R_{my} * (4,2 * \sin(\alpha)) - R_{mx} * (4,2 * \cos(\alpha)) - 0,87 * (4,2 * \sin(\alpha)) = 0$$

Una vez planteadas las ecuaciones de estática y con ayuda de las relaciones de los ángulos del mecanismo de hoekens planteadas anteriormente, se procede a realizar una gráfica (Figura 85) mostrando los valores del torque generado sobre el eje del motor de la pata a medida que cambia el ángulo theta de la manivela.

Para la solución en EES fue necesario realizar un cambio de variable donde  $\theta = 180 - d$ , siendo d el ángulo suplementario de theta, debido a que los métodos de solución de EES no permitían dejar las ecuaciones en términos de theta directamente. El ángulo que se varió fue d por lo que theta adquiere valores positivos y negativos dependiendo de si d es mayor o menor de  $180^\circ$ .

**Figura 85. Grafica de torque requerido para equilibrio vs ángulo de la manivela del mecanismo de hoekens**



Fuente: Autores

Como se puede observar en la gráfica anterior, el torque máximo es de 6 Kgf.cm y se presenta cuando el ángulo de la manivela es de 0°. Por esta razón se decidió emplear los micromotores con relación de 298:1, los cuales según datos del fabricante proporcionan un máximo torque nominal de 6,4 Kgf.cm.

## 8.2 CARACTERIZACION DE LOS MICROMOTORES

Como es bien sabido, los valores nominales dados por los fabricantes difieren en la mayoría de los casos en gran medida a los valores proporcionados en la realidad.

Para la caracterización se tomó un micromotor al cual se le acopló una barra manivela de 1.5 cm. A esta manivela se iban agregando pesos y registrando valores de corriente y voltaje de la fuente cuando el motor se encontraba en stall.

En la tabla 13 se presentan los resultados de las pruebas de caracterización

**Tabla 13. Caracterización de micromotor 298:1**

Peso [Kgf}	Voltaje [V]	Corriente [A]
1	2,1	0,6
2	4,6	1
3,3	9	1,7

Fuente: Autores

En la figura 86 se observa los ensayos realizados para la toma de datos.

**Figura 86. Pruebas de caracterización de micromotor.**



Corriente



Voltaje

Fuente: Autores.

En esta prueba se obtuvo un peso máximo de 3.3 Kgf que al multiplicarlo por el brazo de 1.5cm da un torque de 4.95 Kg.cm pero con el inconveniente de que consume una corriente de 1.7 A siendo esta mayor que la corriente nominal de 1.6A, esto genera que el motor se recaliente rápidamente y se quemé.

Por seguridad de los motores es recomendable utilizarlos a una corriente máxima de 1A lo que genera un torque de 3Kgf.cm siendo este torque mucho menor que el nominal y además menor que el torque crítico requerido.

Sin embargo, los rangos de operación de la manivela se encuentran entre  $90^\circ$  y  $-90^\circ$  sin pasar por  $0^\circ$  y como se puede observar en la gráfica, los valores de torque requerido en este rango no superan los 2Kgf.cm por lo que los motores seleccionados sirven para la aplicación deseada.

## 9. DISEÑO DEL SISTEMA ELECTRONICO Y DE POTENCIA DEL PROTOTIPO FINAL

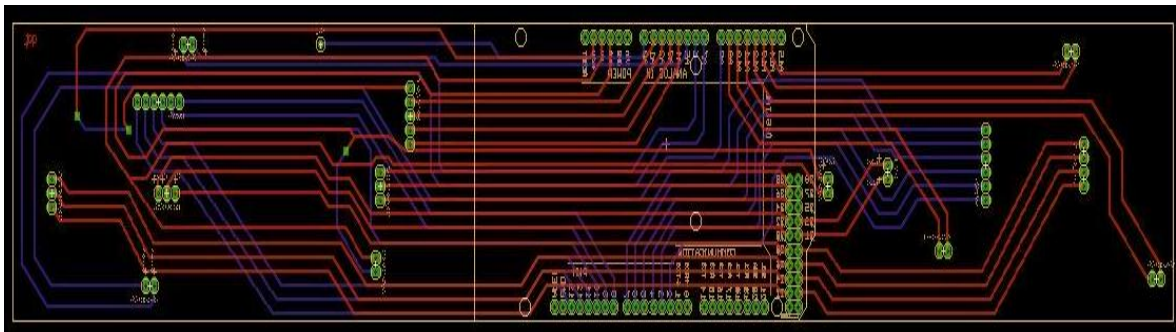
Debido que los requerimientos de esta aplicación suponen que los elementos de control se encuentren en el robot ya que este es una aplicación de robótica móvil, se vio la necesidad de diseñar 3 placas de circuitos impresos (PCB) de dos capas. Debido a la gran cantidad de elementos y de conexiones requeridas inicialmente era necesario placas de más de dos capas para suplir la falta de espacio en la PCB, sin embargo a causa de la limitante que supone el proceso de manufactura regional (solo posibilita 2 capas por placa) se vio la necesidad de fabricar más de una placa las cuales se acoplarán entre sí y con la Arduino, simulando las capas adicionales.

El diseño tuvo en cuenta los rangos de movimiento de las manivelas de cada una de las patas para que estos no interfirieran con el circuito.

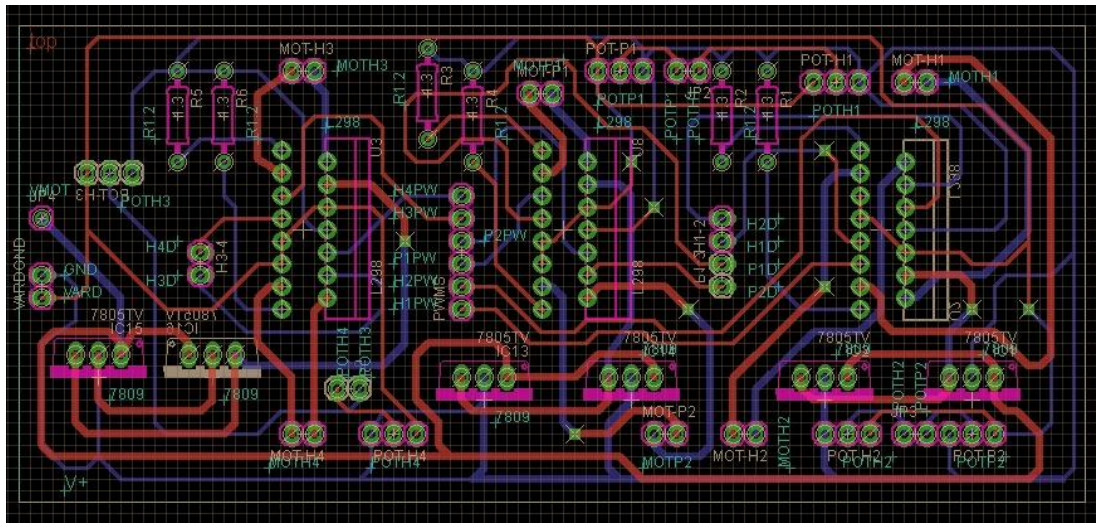
El diseño fue realizado con ayuda del software EAGLE en el cual se especificaron las conexiones de los integrados y se elaboraron los planos de impresión.

El circuito realizado en EAGLE se muestra en la figura 87 con sus respectivos caminos de conexión de las PCBs, en la sección de anexos se pueden ver con más detalle.

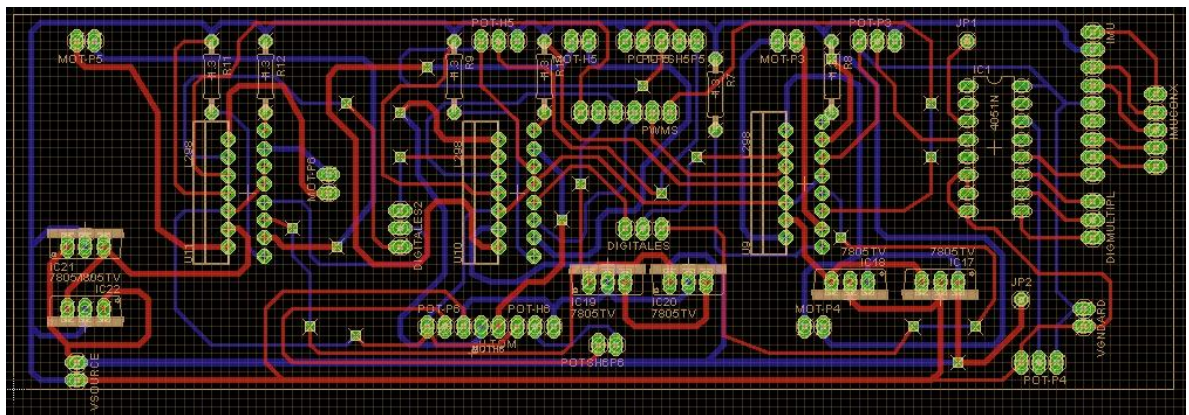
**Figura 87. Caminos de conexión de las PCBs**



Shield Arduino diseñado



Placa delantera con tres drivers de motores

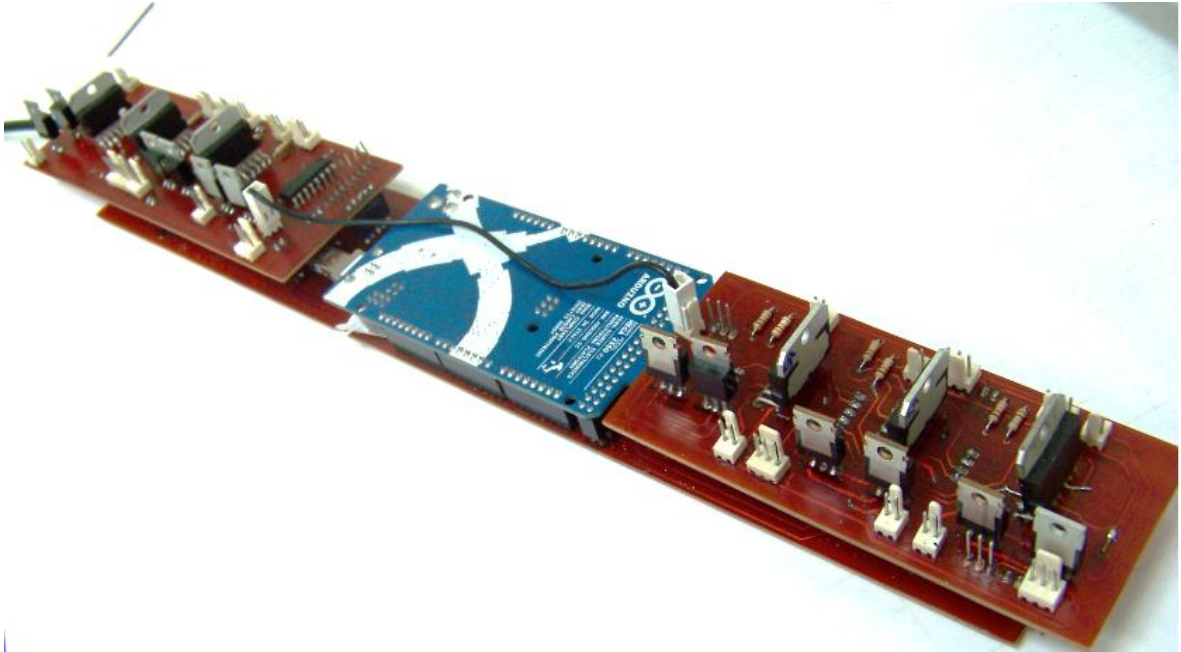


Placa trasera con 3 drivers de motores, multiplexor y conexiones para un IMU.

Fuente: Autores

Una vez realizados los circuitos de impresión, se fabricaron las placas y posteriormente se soldaron cada uno de los componentes manualmente, el ensamble de las placas de circuitos se puede observar en la figura 88.

**Figura 88. Ensamble de PCBs.**



Fuente: Autores.

Los componentes electrónicos que estarán presentes en las PCB son:

- L298N: Un integrado con arquitectura puente H que se encargara de controlar la velocidad y dirección de giro de 2 motores.
- 7809: Reguladores de voltaje para mantener constante la alimentación del L298N
- Resistencia de 1,2 Ohms: Resistencia para limitar la corriente que circula por cada canal del L298N para la protección de los motores.
- Multiplexor 4051: Utilizado para aumentar las entradas análogas de la Arduino
- Modulo inercial IMU: Utilizado para retroalimentar la posición del robot en cada instante.

## 10. PROCESO DE CONSTRUCCION Y ENSAMBLE DEL PROTOTIPO FINAL

El proceso de construcción fue realizado en el taller de metalmecánica de la escuela de ingeniería de la Universidad Industrial de Santander en el cual se utilizaron las siguientes máquinas y herramientas:

- Dobladora de placas
- Cizalladora
- Torno
- Taladro vertical
- Segueta

### 10.1PIEZAS ELABORADAS

**10.1.1 Estructura.**En una lámina de aluminio, se marca la forma de la estructura y se corta utilizando la cizalladora y un disco de corte fino.

**10.1.2 Barras del mecanismo de Hoekens.**Estas fueron medidas para luego ser cortadas con la cizalladora, posteriormente se realizaron los agujeros con el taladro vertical.

**10.1.3 Acoples para ejes de motores y potenciómetros.**Se compran cilindros de un diámetro un poco mayor que el diámetro del elemento a acoplar (con suficiente espesor para la hechura de la rosca del tornillo prisionero) y se realiza una cilindrada interna con ayuda del torno para realizar el agujero concéntrico.

**10.1.4 Hombros.**Se toma la platina del ancho del hombro y se dobla formando una U, luego se realizaron los taladrados y con ayuda de una lima se le da forma al agujero rectangular.

**10.1.5 Apoyos para los hombros (Potenciómetro).** Se toma la platina y se dobla en forma de L luego se pasa por la cizalladora y se recorta a la medida luego se taladra para hacer el agujero del eje y por ultimo con la segueta se recorta hacia afuera del agujero para dejar un espacio por donde ingresar el eje.

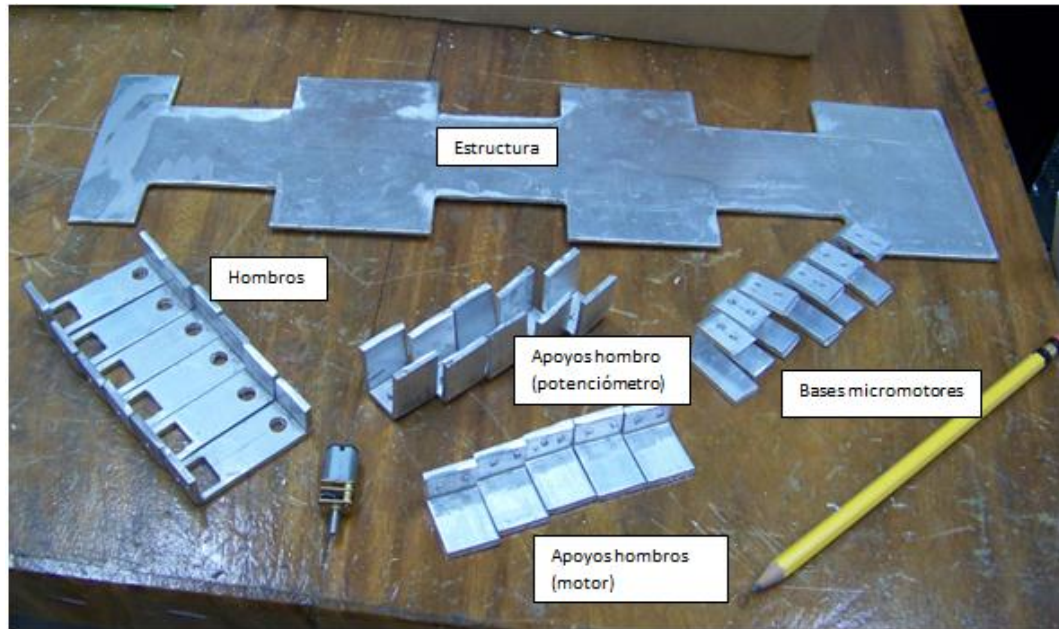
**10.1.6 Apoyos para los hombros (motor).** Se dobla la platina en forma de U y se taladra con dos brocas diferentes, una para el eje del hombro y otra para el eje del potenciómetro, luego con la segueta se saca un bocado para poder ingresar el eje del hombro.

**10.1.7 Base para micromotor del hombro.** Se dobla la platina en forma de U con la altura adecuada para que el eje del micromotor acople con el eje del hombro y luego se recorta con la cizalladora

**10.1.8 Placa base potenciómetro.** Se toma una platina, se corta a la medida con la cizalladora y se taladran los agujeros.

Algunas de las piezas fabricadas son mostradas en la figura 89, aunque hacen falta unos procesos de manufactura.

**Figura 89. Piezas elaboradas**



Fuente: Autores

## **10.2 ENSAMBLE DE COMPONENTES**

Para el ensamble, primero se remachan los apoyos de los hombros en sus respectivas posiciones según el diseño de Solidworks, teniendo especial cuidado en que deben quedar colineales los dos apoyos que sostienen el hombro. En la figura90 se puede observar la estructura con todos sus apoyos ya remachados.

**Figura 90. Estructura con Apoyos de hombro**



Fuente: Autores

Los acoples fabricados con el torno se soldaron a los hombros y a las barras manivela y balancín para que de esta forma se puedan acoplar tanto al micromotor como al potenciómetro a sus respectivos elementos.

El mecanismo de Hoekens se ensambla mediante pasadores de aluminio y se acopla al hombro mediante los tornillos prisioneros. Una vez el módulo (Hombro-mecanismo de Hoekens) está ensamblado, se procede a montarlo sobre los apoyos y acoplarlo con el micromotor del hombro.

En la figura 91 se puede ver la forma en que se ensambla un módulo y como queda al final.

**Figura 91. Ensamblaje de módulo**

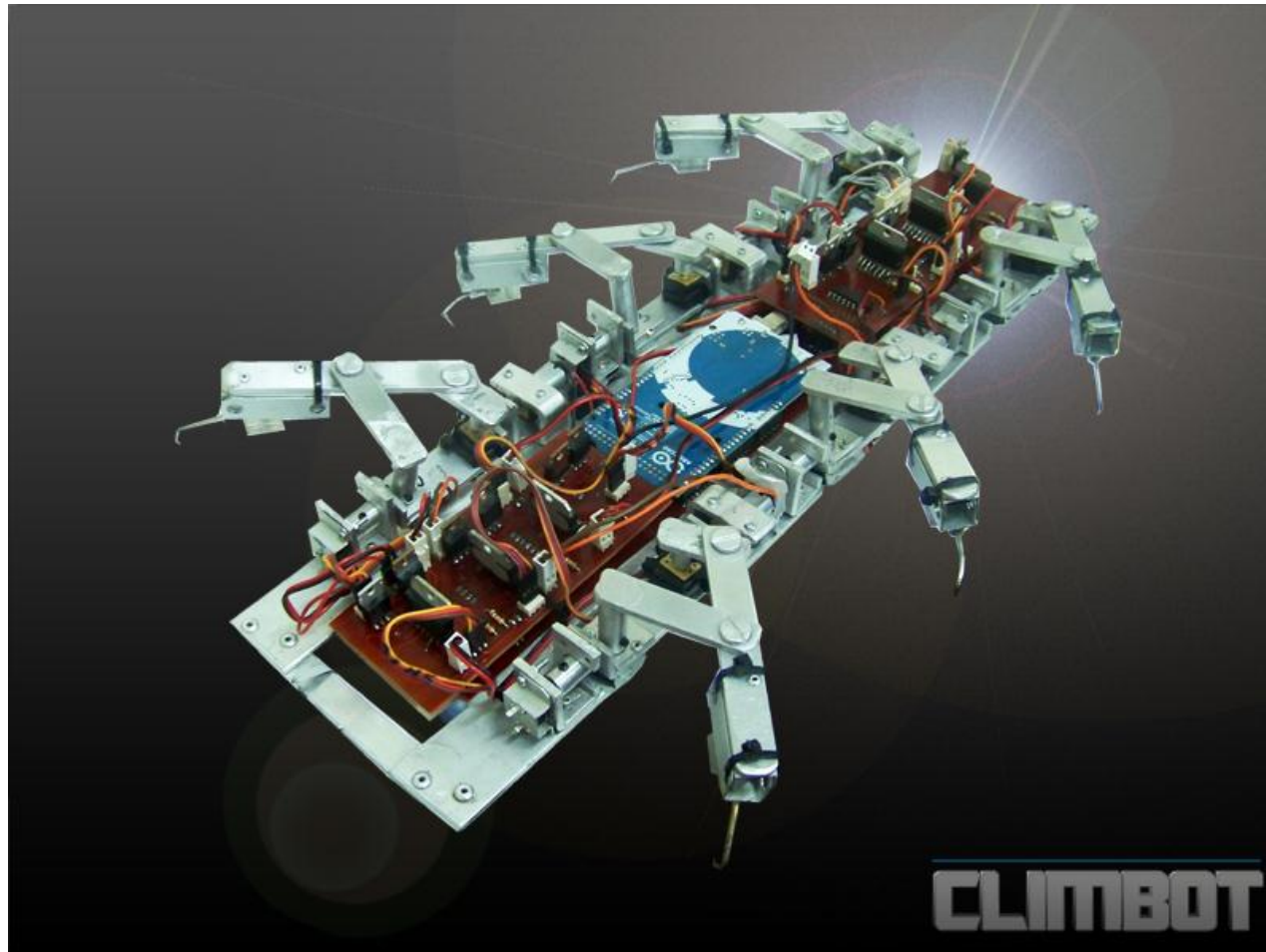


Fuente: Autores

De igual forma se realiza para todos los módulos y por último se monta la PCB en la estructura del robot.

En la figura 92 se puede observar el prototipo finalizado.

Figura 92. Prototipo finalizado.

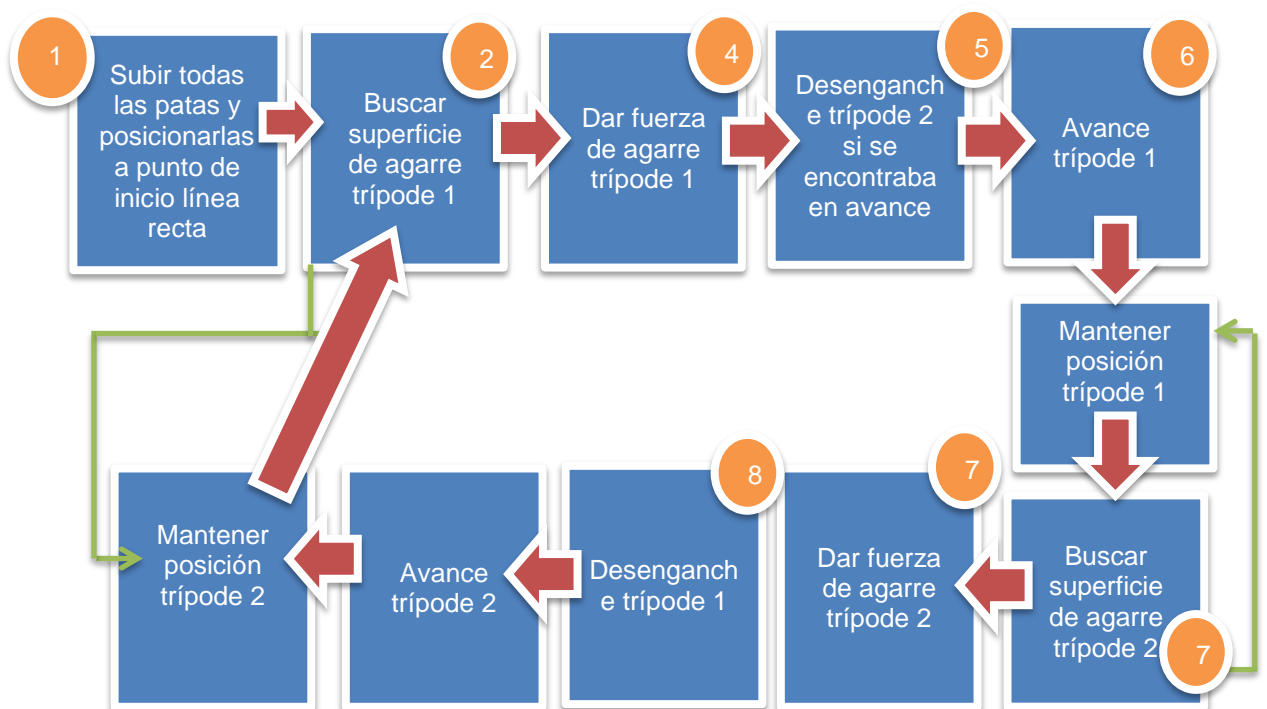


Fuente: Autores

## 11. CONTROL PROTOTIPO ROBOT ESCALADOR

La figura 93 muestra las tareas que debe realizar el robot escalador para asegurar la labor de escalada a trípodos:

**Figura 93. Diagrama flujo tarea de avance en trípodos.**



Fuente: Autores.

1. Subir todas las patas y ubicarlas en posición de avance.
2. Bajar patas de trípedo 1 buscando superficie de enganche
3. Encontrar superficie de enganche de todas las patas del trípedo 1.
4. Dar fuerza de enganche a hombros trípedo 1 que encontraron superficie
5. Si estaban las patas del trípedo 2 en avance, desengancharlas.

6. Realizar avance de patas trípode 1 por la línea recta teniendo en cuenta que se haga a velocidad constante y sin pasarse del punto máximo de línea recta.
7. Mantener patas trípode 1 en posición final de avance mientras se realiza enganche de trípode 2 (La labor de enganche se resume en los ítems 2 a 4).
8. Desenganchar trípode 1
9. Realizar tarea 6 con el trípode 2
10. Mantener posición patas trípode 2.
11. Repetir ciclo desde ítem No. 2.

A continuación se detallan cada una de las actividades que realiza el control:

**Subir patas.** Envía orden a actuadores de hombros para que posicionen estos en un punto superior caracterizado previamente y sensado por los potenciómetros de hombro.

La velocidad a la que suben los hombros es controlada mediante ancho de pulsos para prevenir que se pase del setpoint debido a la inercia

**Posicionar patas.** Envía órdenes a actuadores de patas para que por medio de un control proporcional, ubiquen las patas en cierta posición, generalmente una posición requerida para iniciar la labor de avance.

**Buscar superficie de agarre.** En esta actividad, el controlador envía señal de bajada a actuadores de hombros, sensando si en alguna posición este hombro se ha quedado quieto, verificando el cambio de la posición de este con respecto al tiempo. Si en dado caso no encuentra superficie de enganche en su rango de agarre, el hombro sube nuevamente y posiciona la pata en otra posición sobre la línea recta y realiza la labor de búsqueda.

**Dar fuerza de agarre.** En esta actividad, el controlador envía órdenes a actuadores de hombro para que por medio de una rampa, realicen fuerza sobre la superficie de enganche hasta el máximo valor.

Esta función solo se realiza una vez las patas que van a realizar el avance hayan detectado superficie de contacto para que de esta forma no se desestabilice el robot.

En esta función, la fuerza de agarre se da con una rampa para que los engranajes de los micromotores no sufran impactos y puedan producir fallos en los dientes de los mismos.

**Avance.** En esta instrucción, el controlador busca la manera óptima de mover el robot desde la posición donde esté, hasta la posición extrema inferior de avance, calculando en todo momento, el valor Pwm necesario para mantener una velocidad de avance constante para las distintas cargas que pueda presentar la pata en avance.

La mayoría de las tareas solicitadas para asegurar control de escalada requieren un procesamiento en paralelo ya que algunas actividades deben hacerse, manteniendo otras. Debido al controlador utilizado "Arduino", solo se cuenta con ejecución de instrucciones en serie, es decir una tras otra.

En el caso de control del robot escalador, se hizo necesario emular el procesamiento en paralelo por medio de artilugios de programación. Esta técnica brinda un grado medio de confiabilidad en cuanto a aseguramiento de ejecución de instrucciones mientras se realiza otra, jugando con los tiempos de ejecución. La velocidad de reacción que tendrá este tipo de programación dependerá del tiempo de ejecución entre tareas.

Para la aplicación de este proyecto de grado, lo ideal sería que las acciones se pudieran realizar al mismo tiempo lo cual requiere múltiples procesadores (procesamiento en paralelo).

Debido a la accesibilidad de la placa de desarrollo de Arduino, se optó por utilizar una Arduino Mega 2560 que contiene un microcontrolador (ATmega2560), por esta razón fue necesario crear un código que emulara las características de un procesamiento en paralelo por medio de ciclos while y for, de esta forma se van realizando las acciones una a la vez a la velocidad del procesador, descuidando el resto de las labores por un tiempo dado por la duración de un ciclo.

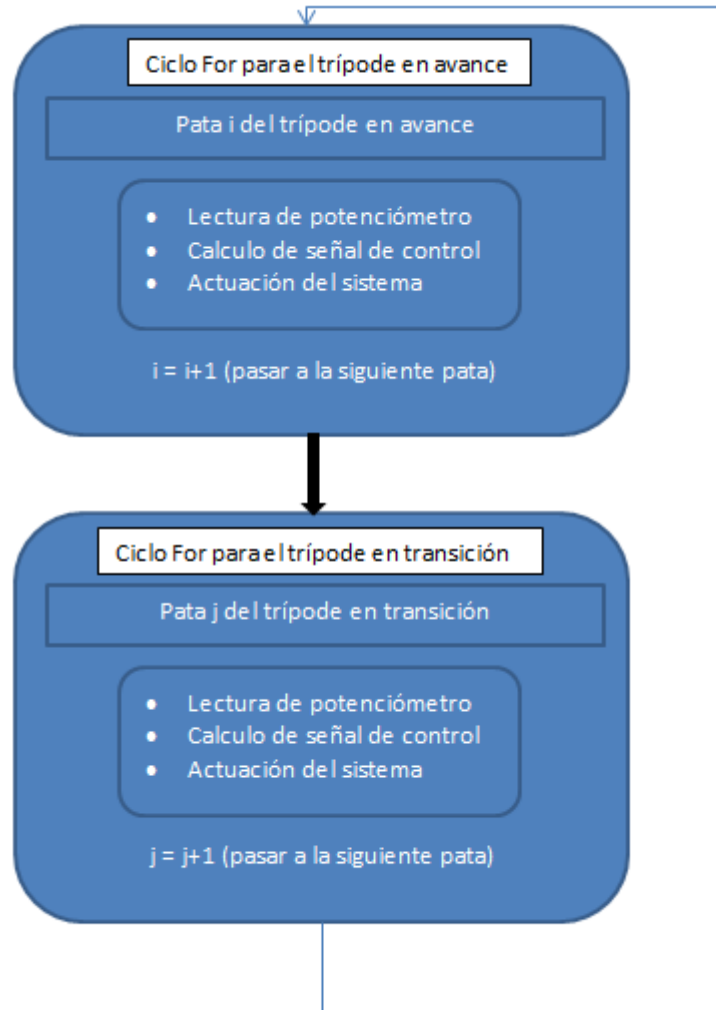
Se debe tratar de que este tiempo sea el mínimo posible para crear el efecto paralelo.

Un ejemplo de esta labor está en el caso donde se requiere mantener cada una de las patas del trípode en apoyo en la posición final de avance (las 3 a la vez haciendo fuerza sin pasarse del setpoint) y al mismo tiempo posicionar cada una de las patas del trípode en transferencia (las 3 a la vez hasta el setpoint), estas operaciones requieren un monitoreo permanente de las posiciones de los actuadores.

Para estos casos, la forma en la que se emulará el procesamiento en paralelo consiste en crear un Loop en el cual se realizaran estas operaciones para cada pata, en donde se tomaran los datos de los sensores, se calculara la señal que se dirigirá a los actuadores y se enviara esta señal, inmediatamente y sin esperar a que el actuador llegue a su setpoint se hará lo mismo con la pata siguiente.

En la Figura 94 se mostrara en un diagrama este proceso y los tiempos muertos en los que no se ejerce control en cada pata

**Figura 94. Ciclos for para emular dos acciones en paralelo.**



Fuente: Autores

En estos ciclos se está verificando una pata a la vez, pero gracias a que Arduino tiene una memoria en la que la última instrucción que se le fue enviada a cierto actuador permanece igual hasta que esta sea modificada por el código, todas las patas que se encuentren en los ciclos for se estarán moviendo a la vez, verificando cada una en intervalos de tiempos que dependen de la velocidad del procesador y de la cantidad de código que exista entre ciclos.

## 11.1 SENSORES USADOS

Para la aplicación de escalada fue necesario incorporar sensores que dieran conocimiento acerca de la posición de la pata y del hombro de manera continua.

Para sensar la posición del hombro se usó potenciómetros ubicados en el mismo eje de giro de este, y debido a que gira en un rango de aproximadamente  $180^\circ$ , un potenciómetro lineal de 1 vuelta suple esta necesidad.

Para obtener la posición de la pata, se podría pensar en ubicar el sensor en el eje de giro de la barra manivela del mecanismo de Hoekens (Ver figura 39), sin embargo, debido a que esta barra tendrá el sistema de actuación y debido a que en el mercado los sensores de posición especializados son pesados y costosos, se optó por implementar un potenciómetro lineal ubicado en el eje de giro de la barra balancín, caracterizando la posición de la manivela ayudados de la figura 40 del capítulo 6 y con la posición de la manivela se caracteriza la posición de la pata gracias a la gráfica de las figuras 44 y 45.

Analizando estas gráficas, podemos notar que en el rango de escalada, el cual es la línea recta donde el ángulo  $\theta$  de la manivela va de  $90^\circ$  a  $270^\circ$ , las relaciones entre ángulo del balancín y ángulo de la manivela es lineal, así mismo, la relación entre el ángulo de la manivela y el Cx de la figura 45 son lineales, con lo que podemos concluir que el sensado de la barra balancín en el rango de avance nos proporciona una relación lineal con el Cx o posición de la pata.

Es de importancia tener en cuenta que fuera de este rango de avance, estas relaciones no son lineales y además la barra balancín medirá valores que se repiten para distintas posiciones, lo cual causaría un problema grave en el sistema de control, siendo crítico el evitar que el sistema trabaje en rangos por fuera de la línea de avance en línea recta.

## 11.2 PROGRAMACION EN ARDUINO DE LA SECUENCIA DE ESCALADA

La programación de las secuencias de escalada se realizó en la plataforma Arduino debido a su bajo costo de hardware y por ser un software libre.

En este código se crearon funciones para cada parte de la secuencia, las cuales son llamadas en el Loop principal cada vez que son requeridas, estas funciones tienen varias entradas las cuales determinan el modo en el que la función actúa en el proceso de escalar. La finalidad de crear las funciones es la no repetición de código para ahorrar memoria y tiempo de procesamiento, ya que este es crítico a la hora de tratar de crear un código que emule las características del procesamiento en paralelo.

Antes de empezar con la explicación del código, se debe tener en cuenta la caracterización del sistema, ya que toda la programación se hará basada en estos parámetros.

**11.2.1 Caracterización de actuadores y sensores.** Para un ordenado y lógico sistema, se caracterizaron los motores y potenciómetros de tal forma que reaccionen de igual manera con los parámetros que se varían, es así como:

- Cuando la pata de un módulo se posiciona en el punto de avance superior, el potenciómetro marcará mayor valor que en el punto de llegada inferior.
- Cuando el hombro se encuentra en la posición arriba, marcará un valor de potenciómetro mayor que en la posición abajo.
- Cuando a un motor de hombros se le envían pulsos así:

Pin PWM – Valor alto

Pin Digital – Valor Bajo

El hombro subirá.

- Cuando a un motor de hombros se le envían pulsos así:

PWM – Valor bajo

Digital – Valor alto

El hombro bajará.

- Cuando a un motor de pata se le envían pulsos así:

Pin PWM – Valor alto

Pin Digital – Valor Bajo

La pata girará hacia adelante, horario para las patas 1,3 y 5, anti horario para las demás.

- Cuando a un motor de pata se le envían pulsos así:

PWM – Valor bajo

Digital – Valor alto

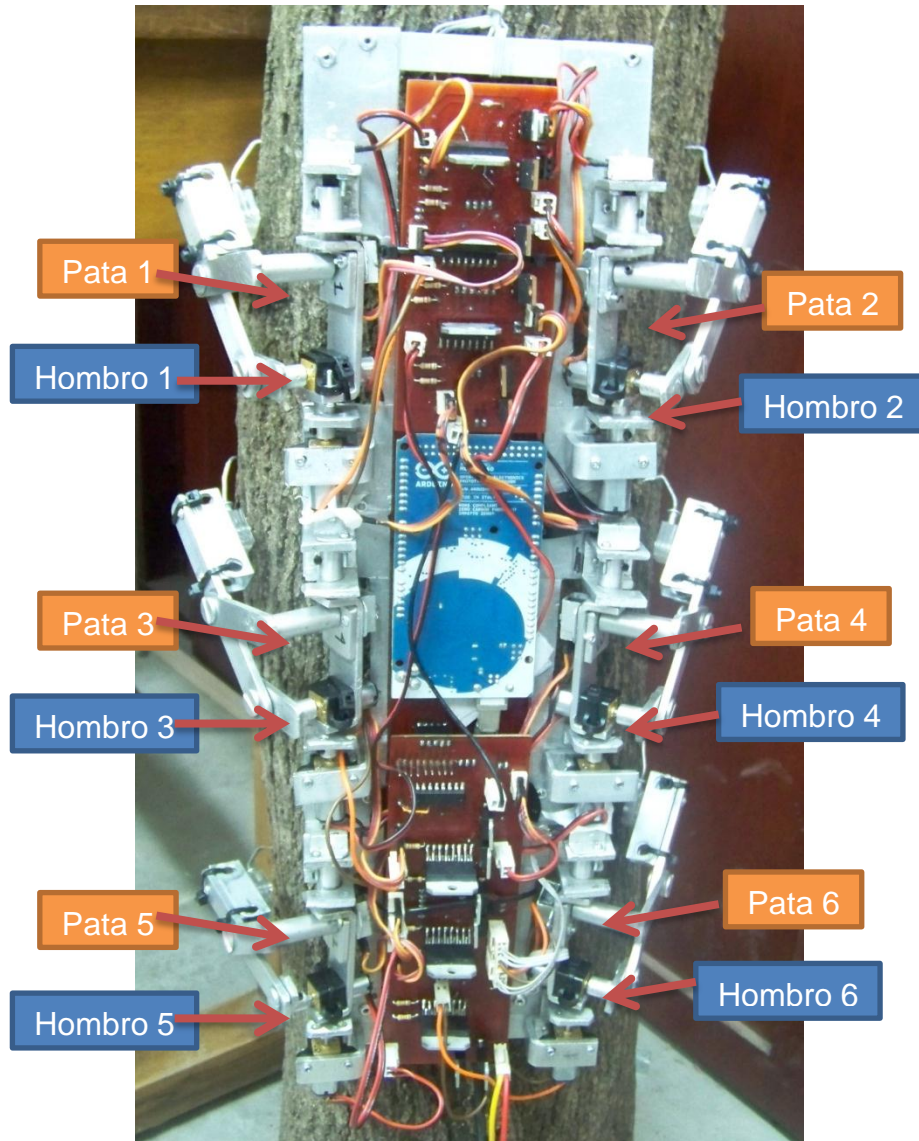
La pata girará hacia atrás, horario para las patas 2,4 y 6, anti horario para las demás.

**11.2.2 Nomenclatura Usada.** Para la correcta interpretación del código explicado, se formuló la siguiente nomenclatura, basados en la figura tal5:

**Trípode 1:** Módulos 1,4 y 5.

**Trípode 2:** Módulos 2,3 y 6

Figura 95. Nomenclatura robot escalador.



Fuente: Autores.

**11.2.3 Declaración de variables.** Al inicio del código deben ser declaradas las variables que se emplearán en la programación, se decidió trabajar con vectores para una mejor organización de las variables, la implementación de vectores también facilita la tarea de lectura y de accionamiento de los actuadores de una manera casi paralela, además en caso.

Dado de querer cambiar las secuencias y los módulos a utilizar, basta con cambiar los vectores declarados al inicio del programa sin necesidad de ir buscando por todo el código donde se encuentra cada instrucción de accionamiento para posteriormente cambiarla.

```
int poshombros[6]; // posiciones leídas del potenciómetro de cada hombro
int pospatas[6]; // posiciones leídas del potenciómetro de cada pata
int pinpotshomb[6] = {13,11,10,9,5,8}; // pines de sensor potenciómetro hombro
int pinpotspat[6] = {14,12,0,0,6,7}; // pines de sensor potenciómetro pata
int pinshpwm[6] = {4,3,6,7,12,13}; // pines pwm para cada hombro --
pinshpwm[0]=hombro1
int pinshdig[6] = {25,29,33,31,24,22}; // pines digitales para cada hombro
int pinsppwm[6] = {2,5,8,9,10,11}; // pines pwm para cada pata --
pinsppwm[0]=pata 1
int pinspdig[6] = {27,23,28,30,32,26}; // pines digitales para cada pata
int adelante[6] = {600, 601, 882, 915, 976, 736}; // valores de pulsos producto de
caracterización de la posición extrema de la pata adelante en línea recta
int atras[6] = {477, 473, 743, 762, 825, 605}; // valores de pulsos producto de
caracterización de la posición extrema de la pata atrás en línea recta
int arriba[6] = {420, 519, 480, 429, 559, 517};
int abajo[6] = {100, 195, 100, 100, 100, 100};
int todos[6] = {0, 1, 2, 3, 4, 5};
int tripode1[3] = {0, 3, 4}; // módulos del trípode 1
int tripode2[3] = {1, 2, 5}; // módulos trípode 2
int pminmov[6] = {40,40,40,40,40,40}; // pulsos mínimos de movimiento de patas
int hombroprueba[1] = {0};
int poshombrosant[6] = {1000,1000,1000,1000,1000,1000}; // vector posición
hombros declarado así para asegurar movimiento de bajada
```

```

int hasta[6]; //vector creado para actualizar los valores de posicionamiento de las
patas
int j;
float kp[6] ={0.4,0.4,0.4,0.4,0.4,0.4 }; //constant proporcional para la funcion de
posicionamiento
int tamano;
int nPuntosLinea = 3; // No de puntos a buscar en la linea recta si no encuentra
apoyo
int nDeltasa = 5;//para función de avance en deltas
int deltaA = 40;// deltas para avanzar
int pCorrida =0;
int pbajada[6]={180,180,180,180,180,180};
int psubida[6]={150,150,150,150,150,150}; //múltiplo de 10
//pines digitales multiplexor
int mpxA = 38;
int mpxB = 36;
int mpxC = 34;

```

En este código fue necesaria la utilización de un multiplexor análogo 4051 debido a que los pines análogos no eran suficientes para la cantidad de sensores presentes en el prototipo.

**11.2.3 Multiplexor.** Este dispositivo permite seleccionar entre 8 canales para leer uno solo a la vez, la selección del canal que se quiere leer se hace mediante una tabla de verdad de tres pines digitales, en la tabla 13 tomada del datasheet de este multiplexor se puede ver los pines seleccionados dependiendo de cada combinación entre los tres pines digitales del multiplexor.

**Tabla 13. Tabla de la verdad para el multiplexor 4051.**

Control Inputs				ON Switches
INHIBIT	C	B	A	
0	0	0	0	X0
0	0	0	1	X1
0	0	1	0	X2
0	0	1	1	X3
0	1	0	0	X4
0	1	0	1	X5
0	1	1	0	X6
0	1	1	1	X7
1	x	x	x	None

x = Don't Care

Fuente: 4051 multiplexor datasheet.

**11.2.4 Configuración inicial de los pines de la Arduino.** Un factor importante en la programación con Arduino y especialmente con los pines digitales, es que se debe configurar el tipo de pin, es decir, si el pin que se desea emplear en una salida (output) o una entrada (input), esta acción permitirá que los pines manden señales de HIGH y LOW de manera adecuada.

En la etapa del void setup () de Arduino, también se realiza una configuración de los motores para que estos permanezcan quietos a la hora de empezar la ejecución del código, para esto se envía a cada uno de los pines controladores del puente H una señal de HIGH para así detener los motores de un arranque fortuito.

A continuación se presenta el código para realizar las tareas anteriormente descritas.

```

void setup(){
for(j=0;j<6;j++){
pinMode(pinshdig[j],OUTPUT) ; //configuracion de los pines de los hombros como
salidas
pinMode(pinspdig[j],OUTPUT) ; //configuracion de los pines de las patas como
salidas

analogWrite(pinshpwm[j],255); //detener los motores de los hombros
digitalWrite(pinshdig[j],HIGH);

analogWrite(pinsppwm[j],255); //detener los motores de las patas
digitalWrite(pinspdig[j],HIGH);
}
pinMode(mpxA,OUTPUT) ; //configuracion de los pines del multiplexor
pinMode(mpxB,OUTPUT) ;
pinMode(mpxC,OUTPUT) ;
Serial.begin(9600); //iniciacion del Puerto serial
}

```

**11.2.5 Lazo principal del código Arduino.** En este loop se realizan las tareas principales del código, es el lazo que se repetirá siempre mientras este corriendo el código. En él están todas las funciones necesarias para la secuencia de escalada, cada una con sus respectivas entradas de acuerdo a la parte de la secuencia que se esté realizando.

A continuación se presenta el lazo principal de Arduino y sus funciones.

```

void loop(){
for(j=0;j<6;j++)
{

```

```
    poshombros[j] = leerPot(j,2); // posiciones leídas del potenciómetro de cada  
hombro
```

```
    pospatas[j]= leerPot(j,1);// posiciones leídas del potenciómetro de cada pata
```

```
Serial.print ("poshombros ");//impresión de valor de potenciómetros
```

```
Serial.println (j+1);
```

```
Serial.println (poshombros[j]);
```

```
Serial.print ("pospatas ");
```

```
Serial.println (j+1);
```

```
Serial.println (pospatas[j]);
```

```
}
```

```
if(pCorrida==0){
```

```
    subirhombros(todos,6);
```

```
    posicionarpatas(todos,6,adelante,1);
```

```
}
```

```
    bajarhombros(tripode1,3,2);//el 2 final para que no busque superficie y de fuerza  
sostener (tripode1,3);
```

```
    desenergizar(tripode2,3);
```

```
    desenganchar(tripode2,3);
```

```
    subirhombros(tripode2,3);
```

```
    posicionarpatas(tripode2,3,adelante,1);
```

```
    posicionarpatas(tripode1,3,atras,2);
```

```
    posicionarpatas(tripode2,3,adelante,1);
```

```
    bajarhombros(tripode2,3,2);
```

```
    sostener (tripode2,3);
```

```
    desenergizar(tripode1,3);
```

```

desenganchar(tripode1,3);
subirhombros(tripode1,3);
posicionarpatas(tripode1,3,adelante,1);
posicionarpatas(tripode2,3,atras,2); //ya tiene pulsos para dejarlo en esa
posicion quieto
posicionarpatas(tripode1,3,adelante,1);

Serial.println("-----Realizo un ciclo!-----");
pCorrida=1;

}

```

Las funciones empleadas en este lazo serán explicadas a continuación.

**11.2.6 Función subir hombros.** Esta función tiene como entradas un vector que tiene en sus elementos el número que determina los hombros que se desean subir y un entero que es la cantidad de hombros que se desean subir por ejemplo:

`subirhombros(tripode1,3);` : Se suben los hombros del trípode 1 el cual tiene tres hombros a subir.

Esta función sube todos los hombros del vector entrante mediante un for y se sale de esta función cuando todos los hombros del vector entrante hayan alcanzado el set point, este set point está dado por el valor de pulsos leídos de la entrada del potenciómetro acoplado al eje del hombro en su posición superior.

La labor de subida es realizada por medio de un control de entrada constante, enviando unos pulsos mínimos para asegurar que el hombro suba.

```
void subirhombros(int* hombros, int tamano){
```

```

int swiche[tamano];
int sumswitch =0;
int i;
if (pCorrida==1)//cuando inicia el funcionamiento del robot
{
darfuerza (hombros,tamano,2);//desengancha los hombros suavemente
}
//Combinación While y for para emular procesamiento en paralelo
while(sumswitch<tamano){
sumswitch =0;
for(i=0;i<tamano;i++){

if(poshombros[hombros[i]]<arriba[hombros[i]]){//si la posición del hombro está por
debajo de la posición de arriba, entonces súbalos
swiche[i] = 0;
analogWrite(pinshpwm[hombros[i]],psubida[hombros[i]]);//pulsos para subir
minimios, caracterizados previamente para cada hombro
digitalWrite(pinshdig[hombros[i]],LOW)

}
else{//si se pasó de la posición de arriba
analogWrite(pinshpwm[hombros[i]],255);//envía pulsos de frenado para eliminar
inercia
digitalWrite(pinshdig[hombros[i]],LOW);
delay(1);
swiche[i] = 1;
analogWrite(pinshpwm[hombros[i]],255); //envía pulsos para no actuar el motor

digitalWrite(pinshdig[hombros[i]],HIGH);
}
}
}

```

```
    poshombros[hombros[i]]= leerPot(hombros[i],2);
sumswitch = sumswitch + swiche[i];
}
}
}
```

**11.2.7 Función posicionarpatas.** Esta función puede considerarse la más importante de la secuencia de escalada, ya que en esta se encuentra el avance del robot.

Tiene cuatro entradas las cuales son:

- Patas a posicionar (vector con la numeración de las patas a posicionar)
- El número de patas que contiene el vector anterior.
- Posición a la cual quiere posicionar las patas (un vector con valores de pulsos que están relacionados con posición, caracterizados para cada pata).
- Acción, si quiere posicionar adelante o si quiere posicionar atrás (Es diferente para nuestro control posicionar adelante o atrás, ya que el posicionar adelante se realiza siempre cuando el motor se encuentra sin carga, la cual es la labor de posicionamiento en el punto de línea recta superior, por el contrario, la acción posicionar atrás es cuando se desea avanzar el robot, en este caso el sistema se encuentra sometido a cargas variables dependiendo del modo en que el robot agarró, la inclinación del robot, entre otros factores..

En la acción de posicionar adelante, se realizó un control proporcional al error, el cual mantiene la pata en un rango de posicionamiento aceptable al valor set.

Para la acción 2 (avanzar el robot, posicionando atrás), debido a que en esta condición la carga del robot maneja condiciones variables por lo mencionado anteriormente, se hace necesario realizar un control adaptativo para asegurar la labor de avanzada del robot, siendo esta a velocidad constante para cada pata para así evitar desviaciones angulares del robot por el avance más rápido de una pata u otra.

El control adaptativo supone la variación de parámetros de control con el tiempo, dependiente este cambio en el comportamiento del sistema en tiempo real.

El mecanismo de adaptación presenta una solución en tiempo real al problema de diseño para sistema con parámetros conocidos.

Según Landau 1981, un sistema adaptativo se diferencia de un sistema de control normal porque en el sistema adaptativo se tiene un bucle de control en el que se compara un índice de funcionamiento.

Existen muchos controladores y patrones de programación para realizar un sistema adaptativo, un ejemplo muy utilizado es la modificación de los parámetros del controlador a partir de una tabla que ha sido previamente calculada para distintos puntos de funcionamiento en función de una variable auxiliar.

La rapidez con que este sistema se adapte dependerá fuertemente en la rapidez con la que la variable auxiliar refleje el cambio de la dinámica del proceso.

Los controladores adaptativos se pueden dividir en dos grupos, Controladores adaptativos con modelado de referencia (MRAC) y reguladores autoajustables (STR).

Los controladores adaptativos con modelo de referencia(MRAC), intentan alcanzar para una señal de entrada definida, un comportamiento en bucle cerrado dado por un modelo de referencia.

Los reguladores adaptativos autoajustables(STR), tratan de alcanzar un control óptimo, sujeto a un tipo de controlador y a obtener información del proceso y sus señales.

Veamos el código de esta función para entender mejor el tipo de control adaptativo que utilizamos:

```
void posicionarpatas(int* patas, int tamano, int* posfin, int accion){  
  
float kpa[6]={0.2,0.2,0.5,0.5,0.2,0.2}; //Vector de constantes de proporcionalidad  
usadas en el control de avance  
float kda[6]={1,1.3,1,1,1,1};//Vector de constantes derivativas.  
float kia[6]={3,3,3,3,3,3}; //Vector de constantes integrales  
  
float erroracum[6] = {0,0,0,0,0,0};//vector que acumula en caso de detección de no  
movimiento  
int pulsmAv[6]= {100,100,100,100,100,100};//Vector de pulsos minimos para  
avance de cada pata  
int swiche[tamano];//vector alerta para detectar cuando una pata ha llegado a  
setpoint  
int sumswitch =0;//variable para salida de control  
int i;  
int deltatheta;//variable que almacena el cambio de ángulo que ha tenido la pata  
int pospatasant[tamano];// vector que almacena las posiciones anteriores de la  
pata  
float error;// variable que almacena la diferencia entre posición final y posición  
actual de la pata
```

```

float pulsos;//pulsos calculados y enviados a actuadores
int anterior;//variable almacena tiempo en el que empieza a correr el loop de
posicionamiento
int tposterior;// variable que almacena el tiempo en el instante que ha corrido cada
ciclo.
for(i=0;i<tamano;i++){
pospatasant[i] = 5000; //para que delta theta al principio no sea cero se inicializa
en valor alto
    pospatas[patas[i]]= leerPot(patas[i],1); // Se lee la posición de las patas
    swiche[i]=0; //Se setea la variable de aviso de llegada al setpoint de una pata
}

anterior=millis();//tiempo de inicio del bucle principal

while(sumswitch<tamano){
    sumswitch =0;//set de variable de salida en cero para cada inicio de bucle de
emulación paralelo for
for(i=0;i<tamano;i++){// bucle que emula acciones en paralelo
    error = posfin[patas[i]] - pospatas[patas[i]];//error de la pata actual

if(accion==1){//acción 1 posicionar las patas con control proporcional, patas deben
estar en condiciones de no carga
    pulsos=error*kp[patas[i]]; //pulsos proporcionales al error
if(error>5){// si la pata está muy atrás de la posición set
    swiche[i] = 0;// variable de alerta diciendo que no ha llegado la pata
if(pulsos<pminmov[patas[i]]){// si pulsos calculados son menores que los minimos
de movimiento en condición sin carga se mueve con estos.
analogWrite(pinsppwm[patas[i]],pminmov[patas[i]]);
}
else{

```

```

analogWrite(pinsppwm[patas[i]],pulsos);
}
digitalWrite(pinspdig[patas[i]],LOW);
}
else if(error<-5){// si la pata está por encima de la posición set
swiche[i] = 0;
if(pulsos>-pminmov[patas[i]]){
analogWrite(pinsppwm[patas[i]],255-pminmov[patas[i]]);
}
else{
analogWrite(pinsppwm[patas[i]],255+pulsos);
}
digitalWrite(pinspdig[patas[i]],HIGH);
}
else{// Si la pata ya llego, está en un rango de 5 por encima y por debajo de la
posición set.
    swiche[i] = 1; // variable de alerta diciendo llegó la pata
analogWrite(pinsppwm[patas[i]],255);//Pulsos de mantenimiento quieto
digitalWrite(pinspdig[patas[i]],HIGH);
}
    pospatas[patas[i]]= leerPot(patas[i],1);// lee de nuevo donde quedo esta pata
}
else if(accion==2){//acción 2, posicionar patas con carga variable hacia atrás, acá
se realiza el control adaptativo y avance del robot
    pospatasant[i]= pospatas[patas[i]];//posición anterior de la pata
    pospatas[patas[i]]= leerPot(patas[i],1);//posición actual de la pata
    deltatheta = pospatas[patas[i]] - pospatasant[i];//Cambio de la posición. En
este control se usa como variable auxiliar que detecta las condiciones de
velocidad de avance.

```

```

if(deltatetha>=-10){ //La pata se mueve muy lento o se esta descolgando
if(deltatetha==0){//suma al error acumulado para que suba pulsos y aumente
velocidad
    erroracum[patas[i]] += 30;
}
else{
    erroracum[patas[i]] += 30/abs(deltatetha); //suma al error acumulado
proporcional al deltatetha para que suba pulsos y aumente velocidad

}
}
else if(deltatetha < -10){ //Se mueve a velocidad alta
    erroracum[patas[i]] = 0;//resetea variable que aumenta pulsos al detectar no
movimiento
}

//cálculo de pulsos, proporcionales al error, al cambio de velocidad y al
erroracumulado por velocidad.
pulsos=pulsminAv[patas[i]] + kpa[patas[i]]*abs(error) + kda[patas[i]]*deltatetha
+ kia[patas[i]]*erroracum[patas[i]];

//seguridad para no enviar pulsos menores o mayores a los que la función
analogWrite acepta.
if(pulsos>255) pulsos=255;
else if(pulsos<0) pulsos = 0;

pulsos = 255-pulsos;

if(error>1){//se pasó de atrás un poco
    swiche[i] = 1;
}

```



```

delay(5);
analogWrite(pinsppwm[patas[i]],200); //sostenimiento patas
digitalWrite(pinspdig[patas[i]],HIGH);

}
}
}
}

```

Como podemos observar en el código de avance de robot, el avance se rige por los pulsos enviados por PWM:

```

pulsos=pulsminAv[patas[i]] + kpa[patas[i]]*abs(error) + kda[patas[i]]*deltatetha
+ kia[patas[i]]*erroracum[patas[i]];

```

En este se observa que estos, son proporcionales al error, al cambio del ángulo y al error que se acumula por no movimiento.

Estos pulsos mantienen un nivel mínimo de inicio, el cual está determinado por la variable pulsminAv y por el error inicial, esta variable pulsminAv es introducida para minimizar el tiempo que demora en posicionar, y setear un nivel mínimo de fuerza al comenzar.

La parte proporcional al error, disminuye estos pulsos de inicio a medida que se acerca al valor de set, evitando así, que la pata se pase demasiado del set.

La parte proporcional al deltatetha actúa de forma similar al sistema derivativo, ya que el cambio en la posición de la pata es lo mismo que el cambio de la posición con respecto al setpoint:

$$\frac{\Delta\theta}{\Delta t} = \frac{\Delta error}{\Delta t}$$

Sin embargo, debido a la configuración de sensores, una disminución del error conlleva un aumento en el  $\Delta\theta$ , siendo este negativo ya que la posición de atrás es menor que la posición adelante en cuanto a valores marcados por el sensor. Este aumento con signo negativo brinda en el cálculo de pulsos una sustracción de pulsos a medida que el  $\Delta\theta$  se hace mayor (Velocidad se incrementa), lo cual es deseable para mantener una velocidad constante.

La parte integral actúa proporcional al error acumulado por no movimiento, este error acumulado incrementa solo cuando la velocidad de la pata es baja y reseteado cuando la velocidad está en el rango aceptable o es muy alta. Es en esta parte del control donde se puede apreciar el control adaptativo, la cual es necesaria, ya que al variar las condiciones del sistemas, hay puntos en los cuales el control caracterizado inicialmente por los pulsos mínimos, la constante proporcional y la derivativa no es capaz de mover el robot, es allí donde empieza a actuar la parte de acumulación hasta que hace mover el sistemas, una vez en movimiento, este sumando desaparece ya que se restablece la variable error acumulado a cero.

Este tipo de control podríamos definirlo como un control normal en lazo cerrado pero que se adapta en circunstancias en las que el sistema no puede mover el robot, estableciendo un sumando que aumenta cada vez que detecta que el sistema no está moviéndose y desapareciendo una vez ha vencido el punto de freno para continuar con el control que llevaba.

La función posicionar patas en su labor de avance del robot, establece unos rangos en los cuales el sistema toma como finalizado el ciclo de la pata, estos son:

- Si la pata se pasa del setpoint.
- La pata está muy cercana al setpoint

Una vez detectado esto, el controlador envía a los actuadores una señal de frenado para evitar que se pase por más por inercia de movimiento y mantiene una señal de sostenimiento para evitar que el robot se descuelgue.

También, esta función por seguridad y por aseguramiento de escalada establece un tiempo máximo para realizar la tarea de avance, si dicho tiempo se excede, se envía a los actuadores señal de frenado y señal de mantenimiento, saliendo posteriormente del ciclo total.

**11.2.8 Función asignarvector.** Esta función asigna los elementos de un vector (vectorin) a otro vector (vectorout), esta función realiza prácticamente una copia de un vector en otro.

```
void asignarvector(int* vectorout, int* vectorin, int tam){
int i;
for(i=0;i<tam;i++){
vectorout[i] = vectorin[i];
}
}
```

**11.2.9 Función desenergizar.** Envía señal de desenergizado de actuador a patas, las cuales se envían como parámetro a la función.

```
void desenergizar (int* patas, int tamaño){
int i;
for(i=0;i<tamaño;i++){
analogWrite(pinsppwm[patas[i]],255);
}
```

```

digitalWrite(pinspdig[patas[i]],HIGH);
}
}

```

**11.2.10 Función darfuerza.** Esta función tiene dos modos de operar, uno en el que da agarre a los hombros hacia abajo a forma de rampa hasta el valor de máxima fuerza y otra en la que quita fuerza con proporción en forma de rampa para empezar labor de subida.

```

void darfuerza (int* hombros,int tamano,int accion)//accion=1 es para el agarre
gradual accion=2 es para soltar gradualmente
{
int i;
if (accion==1)//agarrar con rampa
{
j=pbajada[hombros[0]];//inicia dando fuerza con pulsos del primer hombro
// j es el valor de pulsos enviado
while(j>0){//da fuerza hasta máximo valor el cual es PWM Bajo y digital alto

for (i=0;i<tamano;i++)// ciclo para emular procesamiento en paralelo
{
analogWrite(pinshpwm[hombros[i]],j);
digitalWrite(pinshdig[hombros[i]],HIGH);
}
delay(20);// tiempo de espera para que la rampa sea suave
j-=2;
}
for (i=0;i<tamano;i++)
{
analogWrite(pinshpwm[hombros[i]],0);

```

```

digitalWrite(pinshdig[hombros[i]],HIGH);
    }
}
else // desenganchar suave , acción de desenganche
{
    j=0;
    while(j<255)
    {
        for(i=0;i<tamano;i++)
        {
            analogWrite(pinshpwm[hombros[i]],j);
            digitalWrite(pinshdig[hombros[i]],HIGH);
        }
        j+=10;
    }
}
}
}

```

**11.2.11 Función leerpot.** Esta función lee valor de sensor potenciómetro de pata o de hombro n veces para un instante dado y realiza un filtro promedio de movimiento, devolviendo un valor filtrado. La función a su vez determina si es necesario realizar el multiplexado del integrado 4051 para leer los sensores que quedaron con entrada al multiplexor.

El filtro de movimiento se realiza ya que las señales de sensores son distorsionadas en ciertos instantes por ruido, y puede ser difícil aislar estas señales, una forma de suavizar estas señales es por medio de un filtro promedio de movimiento.

El filtro funciona creando una nueva señal compuesta del promedio de los puntos de la señal original, la forma más fácil de implementarla es un promedio de movimiento de dos pasos.

Teniendo una señal A que es el vector de n datos:

$$A = [a_1, a_2, \dots, a_{n-1}, a_n]$$

Después de aplicar el filtro la señal B sería:

$$B = \left[ a_1, \frac{a_2 + a_1}{2}, \dots, \frac{a_{n-1} + a_{n-2}}{2}, \frac{a_n + a_{n-1}}{2} \right]$$

El primer dato en B es el mismo que en A, los siguientes se vuelven el promedio entre el valor de A en esa posición y el inmediatamente anterior.

En nuestro filtro se tomarán 6 valores casi instantáneos del sensor y se realizará un filtro promedio de movimiento de tres pasos, el primer paso será crear el vector A, el segundo paso se pasará del vector A al vector B, y en el tercero se realizará un promedio de los datos que contiene el vector B.

En el código Arduino se puede apreciar mejor esta transformación:

```
int leerPot(int pos, int tipo){
float potValue[6];
float corrValue[6];
float devolver;
int i;

if(tipo==1){//es una pata la que se quiere sensar
switch(pos){//la pata 3 y la 4 son las que están siendo multiplexadas
case 2: //pata 3 X7
```

```

digitalWrite(mpxA,HIGH);
digitalWrite(mpxB,HIGH);
digitalWrite(mpxC,HIGH);
break;
case 3: //pata 4 X1
digitalWrite(mpxA,HIGH);
digitalWrite(mpxB,LOW);
digitalWrite(mpxC,LOW);
break;
default:
break;
    }
for(i=0;i<6;i++){//realiza 6 lecturas del sensor
potValue[i] = analogRead(pinpotspat[pos]);
}

}
else{//es un hombro el que se quiere sensar, ninguno de estos va a multiplexor
for(i=0;i<6;i++){//realiza 6 lecturas del sensor
potValue[i] = analogRead(pinpotshomb[pos]);
}
}

//Aplicación de filtro promedio de movimiento
corrValue[0] = potValue[0];
for(i=1;i<6;i++){
    corrValue[i] =(potValue[i-1] + potValue[i])/2;
}
for(i=0;i<6;i++){
    devolver +=corrValue[i]/6;
}

```

```
//valor de vuelto por la función  
return (int) devolver;  
}
```

## 12. RESULTADOS

En esta etapa del proyecto se registraron valores de corriente en una de las patas intermedias con ayuda de un sensor de corriente de efecto hall, la idea de estas mediciones es visualizar la respuesta del control en diferentes secuencias, mostrando valores de corrientes y posición. Estos valores de corrientes están relacionados con el torque necesario en la pata medida.

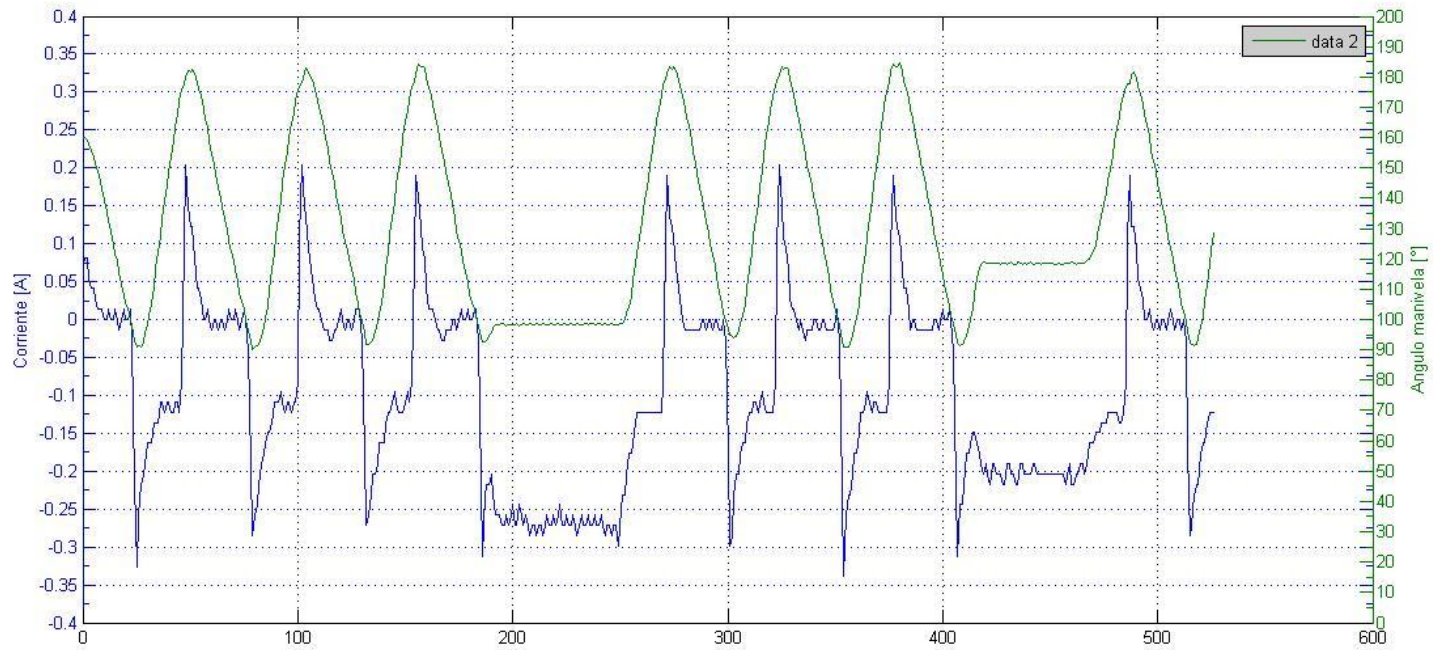
En la figura 96, se muestra la respuesta de control en un movimiento de posicionar patas adelante ( $90^\circ$ ) y atrás ( $180^\circ$ ) mediante un control Proporcional en carga libre (Únicamente cargas por fricción).

Como se puede observar en la gráfica 96, la corriente empieza en un valor alto para cuando la posición de la pata se encuentra lejos del Set point y se va disminuyendo a medida que se acerca a este. Cuando las patas llegan al set point de adelante y atrás existe un pico de corriente debido al cambio de dirección que imponía la prueba.

Este pico de corriente es el esperado debido a la carga inercial que existe por el cambio de sentido de giro.

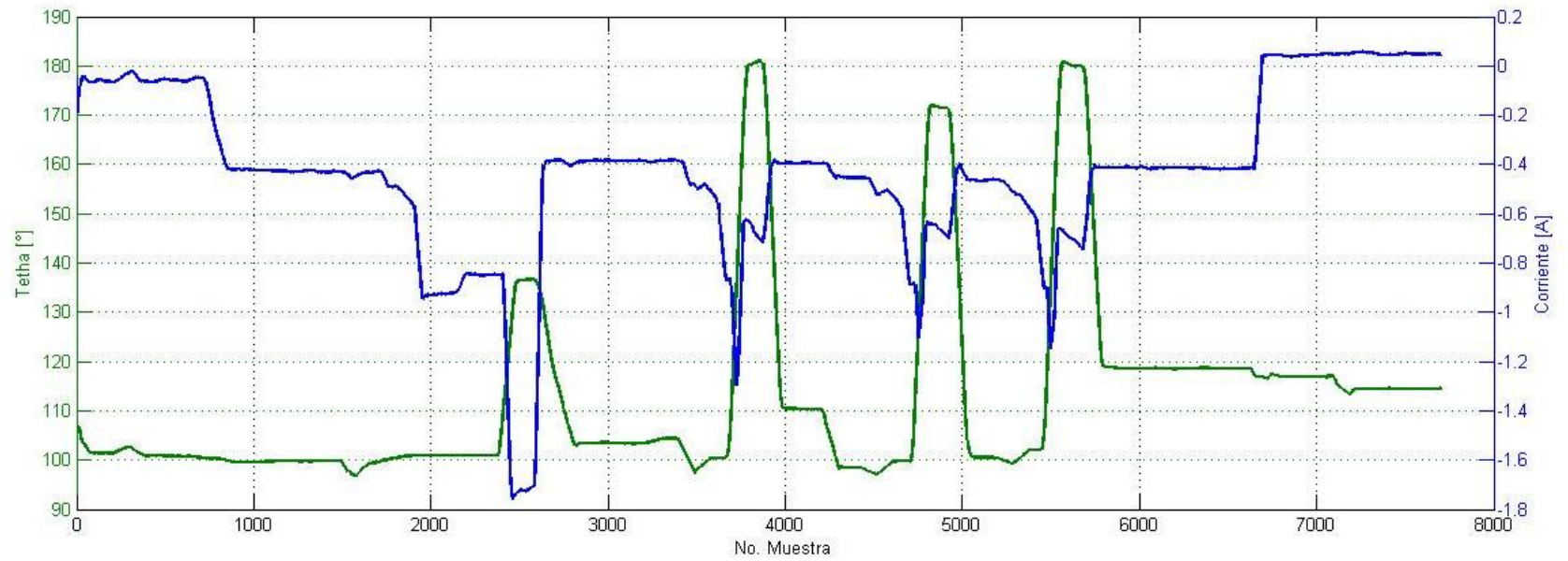
Se realizó una prueba para una secuencia de escalada en vertical y se graficaron los valores de corriente y ángulo de la manivela de la pata número tres, los resultados se muestran en la figura 97.

Figura 96. Grafica de corriente y Ángulo de manivela durante un tiempo de prueba



Fuente: Autores.

Figura 97. Comportamiento en secuencia de avance en vertical.



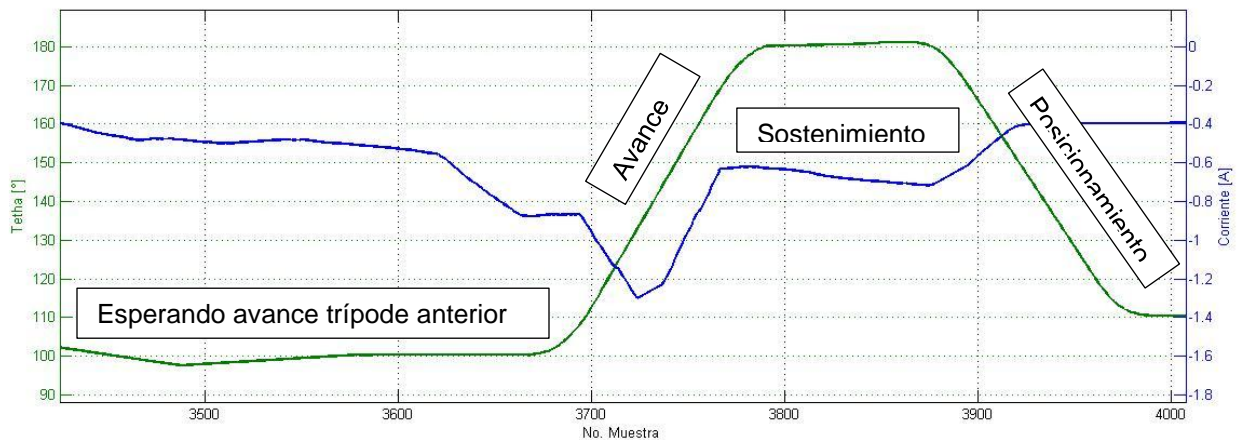
Fuente: Autores.

La figura 97 muestra el comportamiento del ángulo de la manivela y el valor de corriente con una frecuencia de muestreo de 43 muestras por segundo.

En esta figura se exponen cuatro avances del robot, cada avance se compone de un posicionamiento inicial sin carga, un posicionamiento con carga del robot a la posición 180°, y un sostenimiento de la posición mientras engancha el siguiente trípode. La labor de posicionamiento con carga posee un timer de salida, el cual interrumpe el avance después de 2 segundos, con el fin de evitar demoras en este ciclo e inicie la nueva secuencia con el trípode siguiente, proporcionando menos tiempo muerto entre secuencias, es por esto que se observan aproximadamente los mismos tiempos entre inicio de avance e inicio del posicionamiento o fin de sostenimiento (ver figura 98).

En la figura 98 se muestra un zoom realizado a la gráfica para detallar mejor lo explicado.

**Figura 98. Zoom en el avance.**



Fuente: Autores

## 12.1 PRUEBA DE TORQUES EN CONDICIÓN CRÍTICA

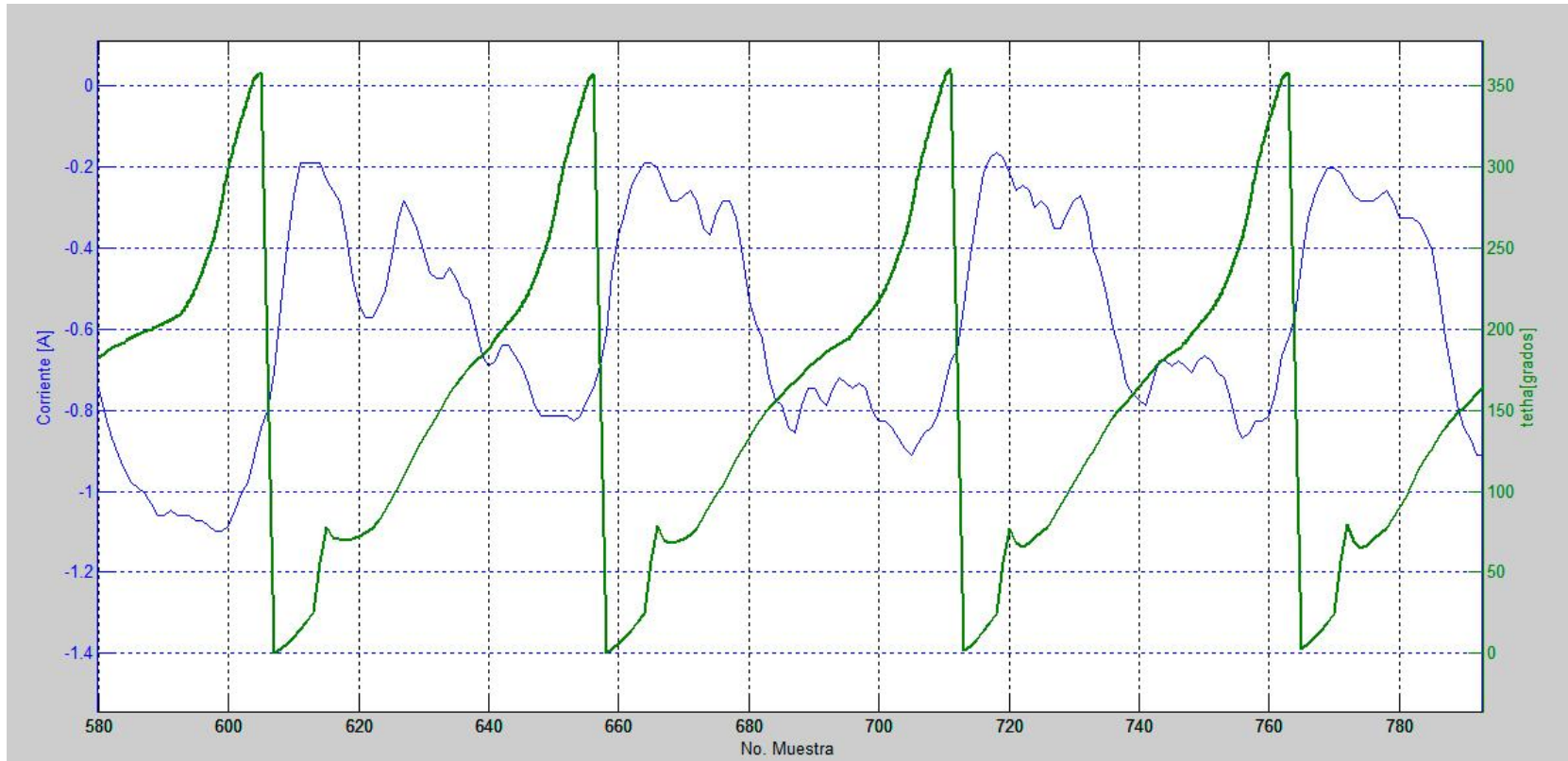
La figura 99 muestra el comportamiento de la corriente durante una prueba del sistema a condición crítica. Esta prueba se realizó configurando las patas de un trípode como se muestra en la figura 83 y enviando una señal de avance a los actuadores.

Esta prueba tuvo como fin verificar los cálculos propuestos en la sección 8.1 que detalla la condición crítica para la selección de motores.

Como se puede apreciar, durante un ciclo de movimiento (manivela de  $0^\circ$  a  $360^\circ$ ) se encuentra un mayor consumo de corriente en el punto de  $\theta = 0^\circ$ . El ensayo resulta en valores de corriente negativos, ya que el sensor de corriente está polarizado inversamente al flujo de electrones en la línea. El ensayo no muestra valores de corriente inversos a los valores centrales tal como lo muestra la teoría, ya que la prueba se hizo enviando una señal de actuación que daba un solo sentido de giro al motor, de tal manera que cuando el sistema no pide carga, la corriente disminuye pero no cambia de sentido.

La figura 100a muestra una línea de tendencia de la corriente durante un ciclo de prueba, la figura 100b es formada por los valores de ángulo y corriente para cada muestra del ciclo, esta línea de tendencia relaciona el ángulo de la manivela con el consumo de corriente en ese instante. La figura 100b se aproxima al comportamiento teórico obtenido en la figura 85 excepto por la porción desde  $30^\circ$  a  $180^\circ$  donde se aprecia un comportamiento de consumo en aumento, este es debido a que en el proceso de manufactura y ensamble este tramo queda con mayor fricción. A estas conclusiones se llega teniendo en cuenta que el torque es aproximadamente lineal con la corriente en las condiciones de operación.

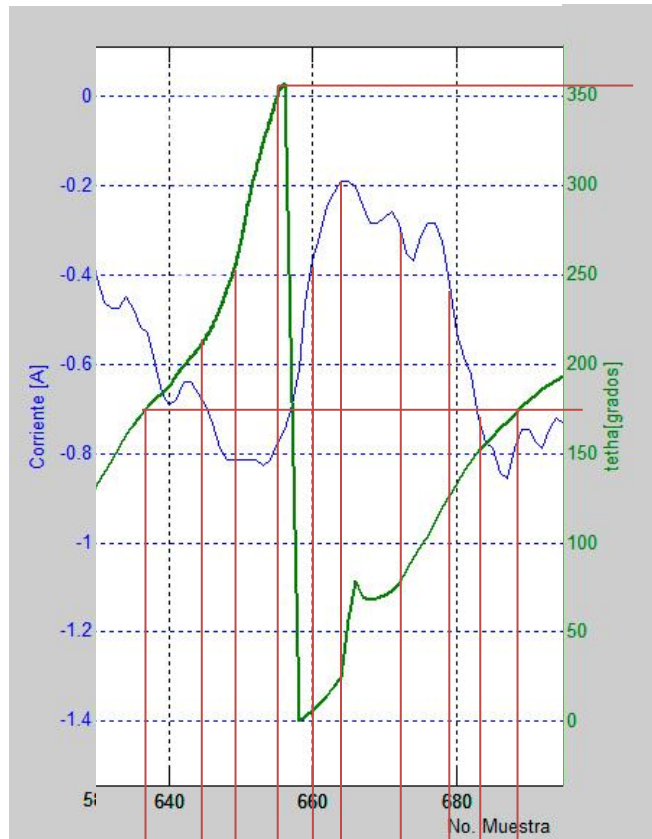
**Figura 99. Comportamiento de la Corriente durante ensayo de criticidad en varios ciclos.**



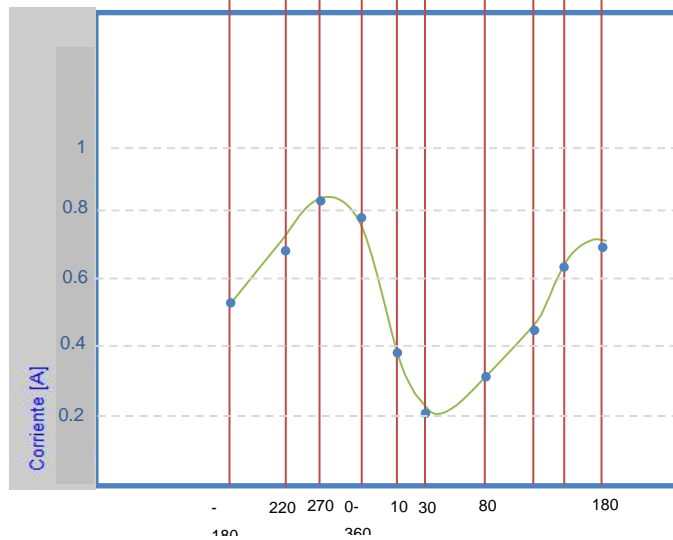
Fuente. Autores

Figura 100. Aproximación línea de corriente durante un ciclo de trabajo.

a)



b)



Fuente. Autores

### 13. CONCLUSIONES

Se generó material de investigación mediante la ejecución del presente proyecto, el cual sirve de punto de partida para futuros proyectos en el área de robots escaladores con mecanismos caminantes, contribuyendo de esta forma al desarrollo de la robótica en nuestro país.

Se logró diseñar y construir un prototipo experimental de robot escalador para superficies de árbol con una envergadura de 28 centímetros, un largo de 41,5 centímetros y un peso de 1kilogramo.

Se logró escalar un árbol inclinado 80 grados respecto a la horizontal hasta una distancia de 43 centímetros. Las pruebas arrojaron que se debe diseñar un mecanismo que asegure la cercanía de la parte delantera del robot con el árbol con el fin de que las patas no pierdan la superficie de contacto y así asegurar una mayor distancia escalada.

Se diseñó y construyó un sistema de sujeción para superficies de árbol mediante un sistema de pivote articulado y garra que asegura contacto y sujeción con la superficie.

Se diseñó e implementó un sistema de locomoción basado en el mecanismo de Hoekens y movimiento en trípodas alternados que proporcionó una velocidad promedio de escalada de 0.35 m/min.

Se diseñó e implementó un sistema de control semi adaptativo basado en un control PID, el cual permite al robot detectar las condiciones de no avance de manera autónoma.

Se creó un código en la plataforma Arduino que emula las características de un procesamiento en paralelo por medio de ciclos While y For, utilizando vectores para una lectura rápida de los sensores y un accionamiento veloz de los actuadores.

Es posible la creación de un prototipo de robot escalador utilizando únicamente materiales y herramientas que se encuentran fácilmente en la región y en la universidad industrial de Santander.

## BIBLIOGRAFÍA

ALMONACID,M, et al.. Motion planning of a climbing parallel robot. IEEE transactions on robotics and automation, 2003. Vol.19.

COBANO, José Antonio. Localización y seguimiento de trayectorias con robots caminantes en entornos naturales. Memoria para optar al grado de doctor. Madrid: Universidad complutense de Madrid. Facultad de ciencias físicas. Departamento de Arquitectura de Computadores y Automática, 2008. 235 p.

ESPANTOSO MIRANDA, Victor. Modelado y simulación del robot PASIBOT. Estudio de la rigidez y mejora en la prevención del vuelco lateral. Madrid: Universidad Carlos III de Madrid. 83 p.

GHASSAEI, Amanda; CHOI, Phil y WHITAKER, Dwight. The Design and Optimization of a Crank-Based Leg Mechanism. Pomona College Department of Physics and Astronomy, 2011. 172p

GONZALEZ, Pablo; GARCIA, Elena y ESTREMER, Joaquin. Advantages of Walking Locomotion. En: Quadrupedal locomotion an introduction to the control of four-legged robots. Madrid: Springer, 2005. 267 p.

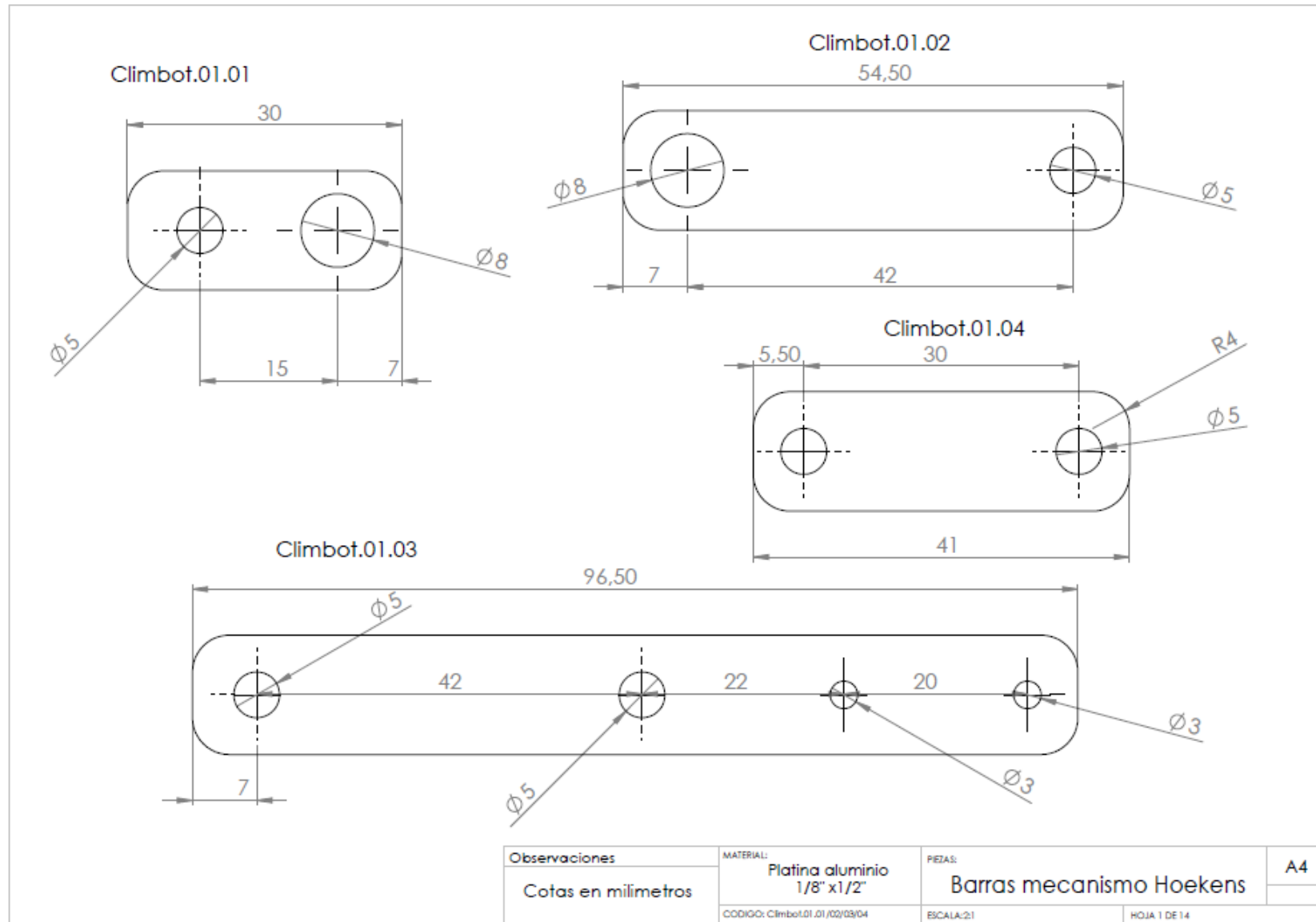
SPENKO,M, et al. Smooth vertical surface climbing with directional adhesion. Robotics, IEEE Transactions on, 2008, Vol.24.

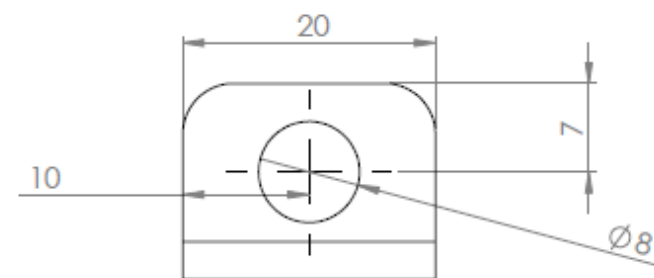
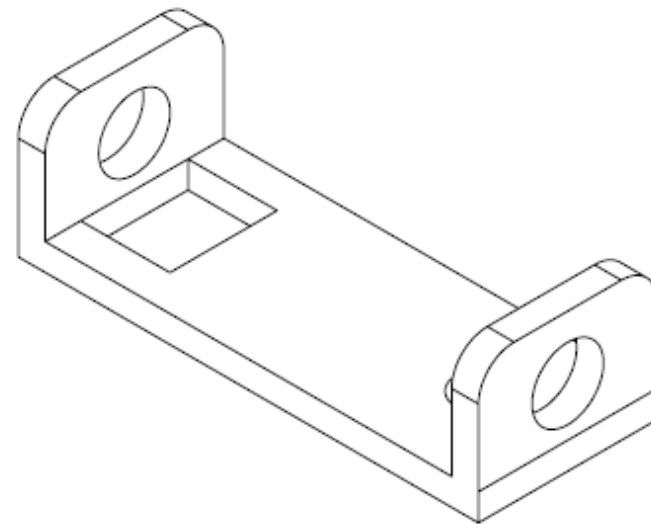
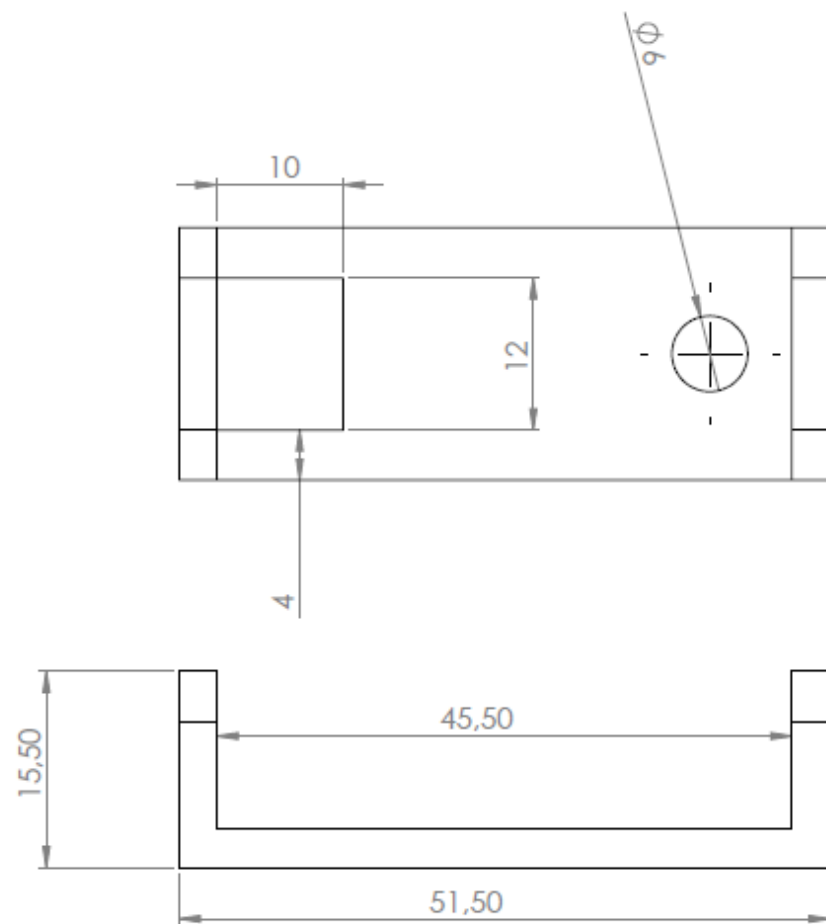
UNVER,Ozgun; UNERI, Ali y SITTI, Metin. Geckobot: A Gecko Inspired Climbing Robot Using Elastomer Adhesives. IEEE International Conference on Robotics and Automation, 2006.

WILE, Gregory D, et al. Screenbot: Walking Inverted Using Distributed Inward Gripping. En: International Conference on Intelligent Robots and Systems, 22,Febrero, 2008. 6p.

## **ANEXOS**

## ANEXO A. PLANOS CLIMBOT





Observaciones  
Fabricada con dobladora, tener en cuenta longitud real por doblez, cotas en mm

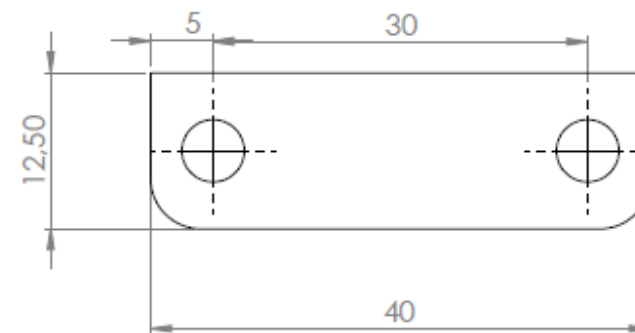
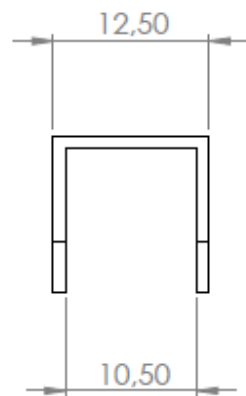
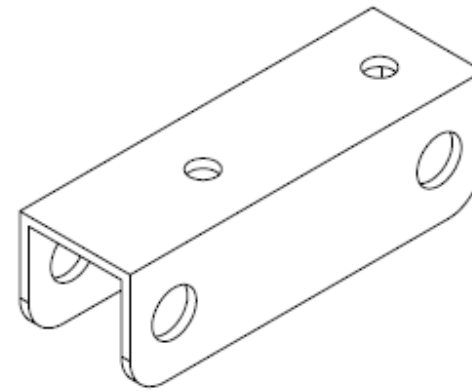
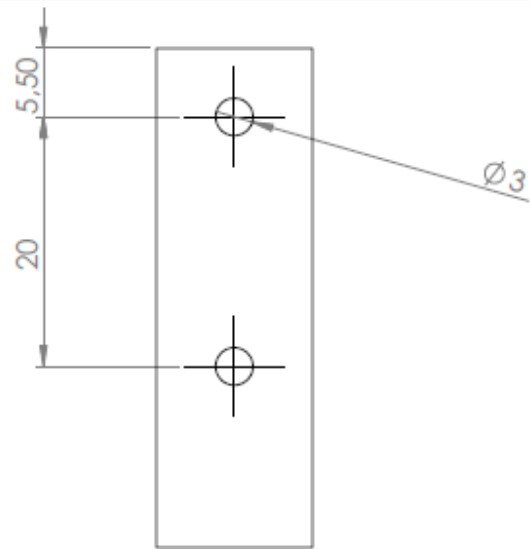
MATERIAL:  
Platina aluminio  
1/8" x 3/4"  
CODIGO: Clmbot.02.01

N.º DE DIBUJO  
ESCALA:2:1

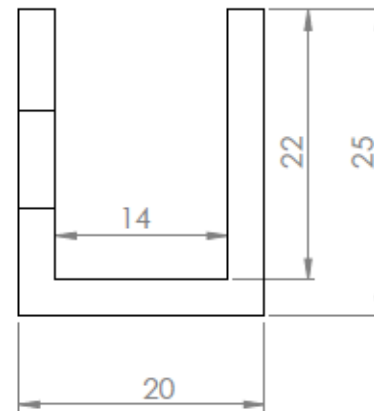
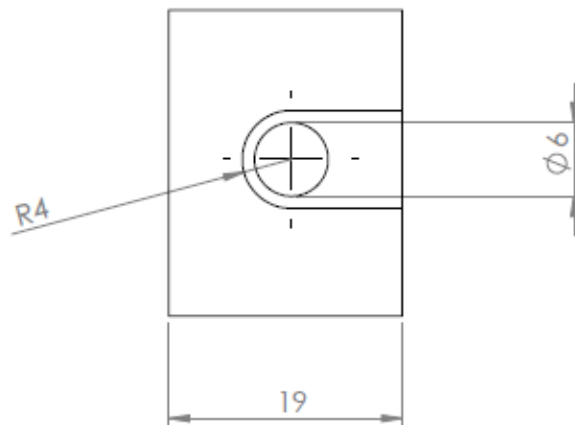
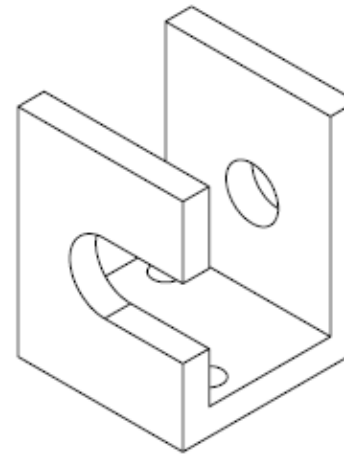
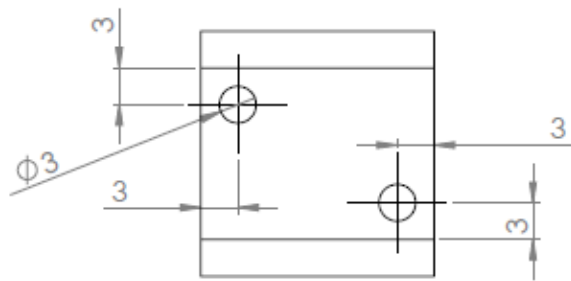
Hombro

A4

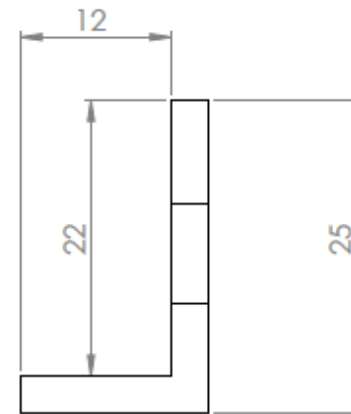
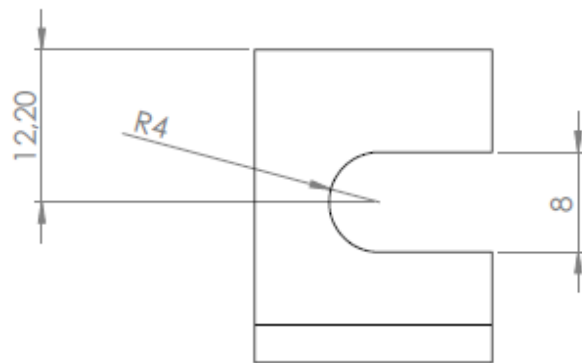
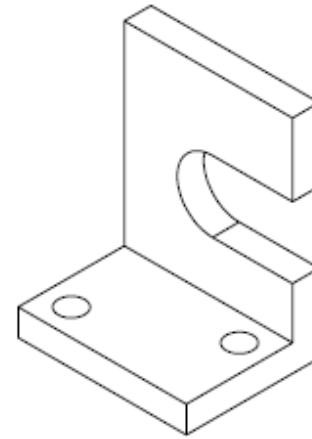
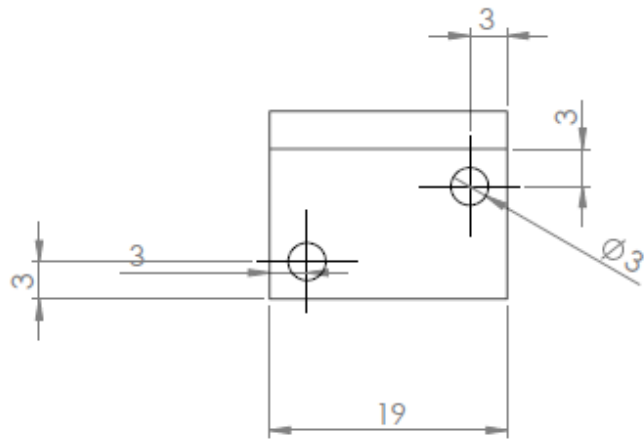
HOJA 3 DE 14



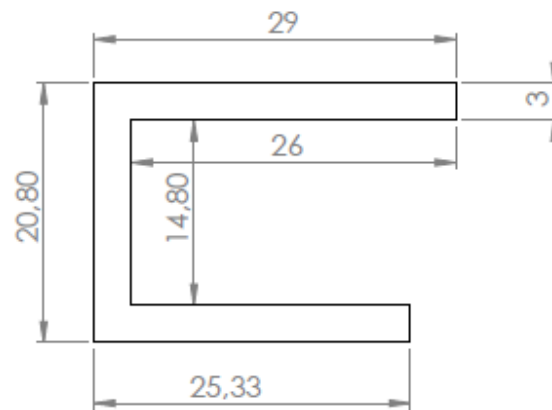
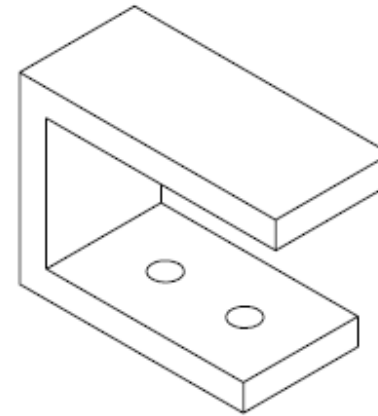
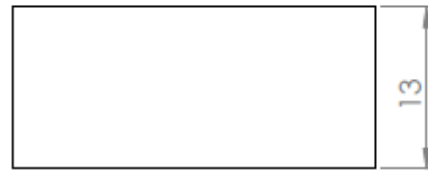
Observaciones	MATERIAL:	N.º DE DIBUJO	A4
Distancia entre centros debe ser precisa, cotas en milímetros	Perfil en U aluminio 1/32" x 1/2"	Base para suspension	
	CODIGO: Cimbot.01.05	ESCALA:2:1	HOJA 2 DE 14



<b>Observaciones</b> Pieza fabricada en dobladora, tener en cuenta longitud requerida por el proceso, cotas en mm	<b>MATERIAL:</b> Platina aluminio 1/8" x 3/4"	<b>N.º DE DIBUJO</b> Apoyo para hombro - pot	<b>A4</b>
	<b>CODIGO:</b> Cimbot.03.01	<b>ESCALA:</b> 2:1	

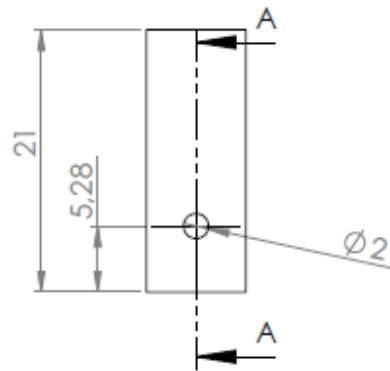


<b>Observaciones</b> Pieza fabricada en dobladora, tener en cuenta las longitudes requeridas por la doblada, cotas en mm	<b>MATERIAL:</b> Platina aluminio 1/8" x3/4"	<b>N.º DE DIBUJO</b> Apoyo para hombro - Motor	A4
	CODIGO: Cimbat.03.02	ESCALA: 2:1	

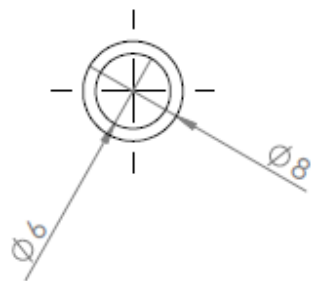
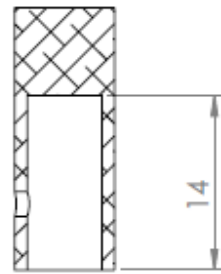


<b>Observaciones</b> Pieza fabricada en dobladora, tener en cuenta las longitudes requeridas por la doblada, cotas en mm	<b>MATERIAL:</b> Platina aluminio 1/8" x3/4"	<b>PIEZA:</b> Base para micromotor hombro	<b>A4</b>
	<b>CODIGO:</b> Cimbot.03.03	<b>ESCALA:</b> 2:1	

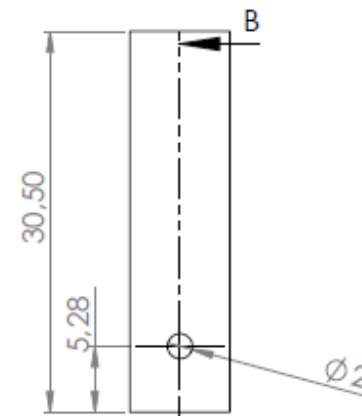
Climbot.02.02



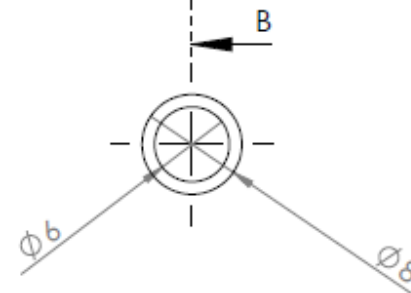
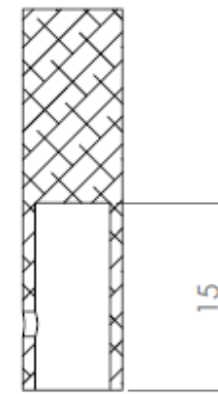
SECCIÓN A-A



Climbot.01.06

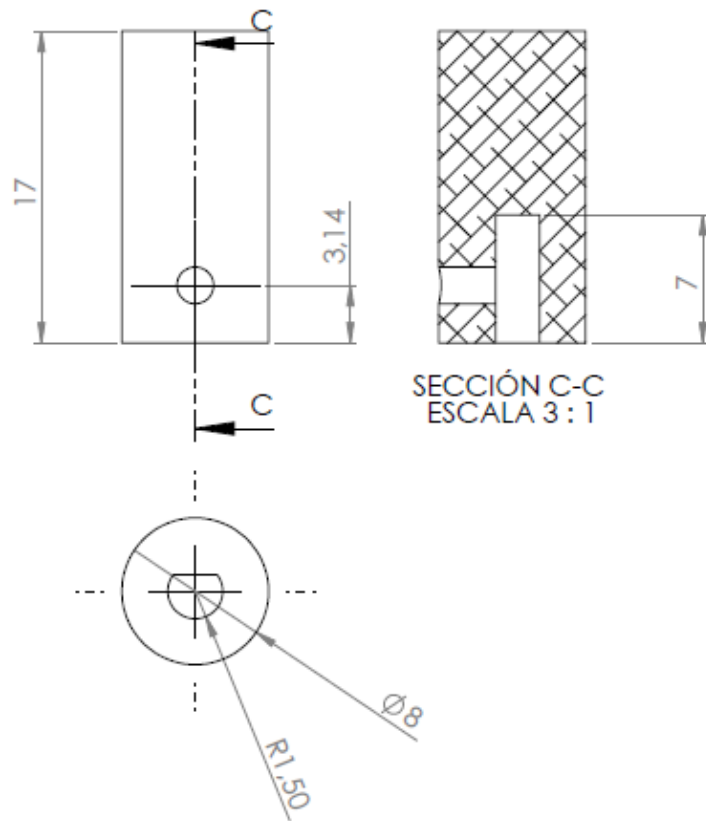


SECCIÓN B-B

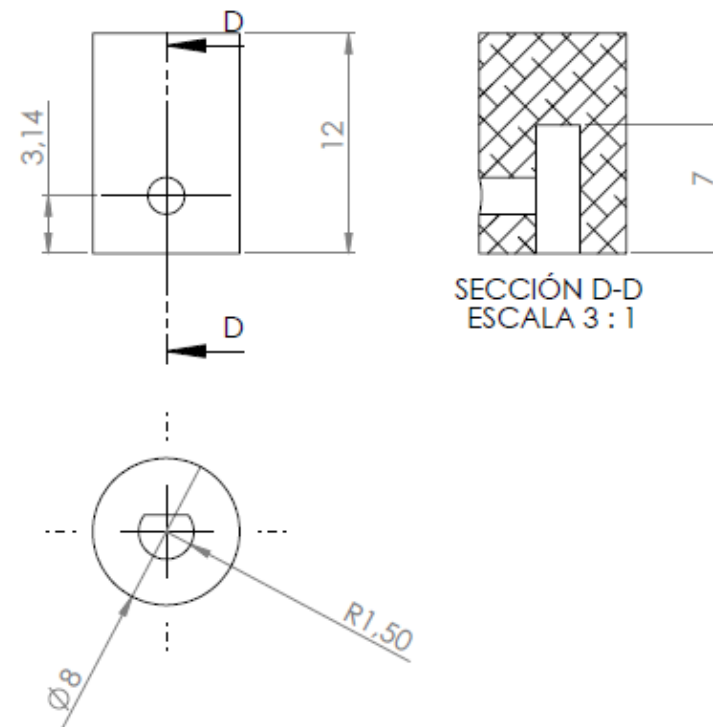


Observaciones	MATERIAL: Cilindro aluminio 5/16"	PIEZAS: Acoples Potenciómetros	A4
Cotas en milímetros	CODIGO: Climbot.01.06, Climbot.02.02	ESCALA: 2:1	HOJA 7 DE 14

Climbot.02.03

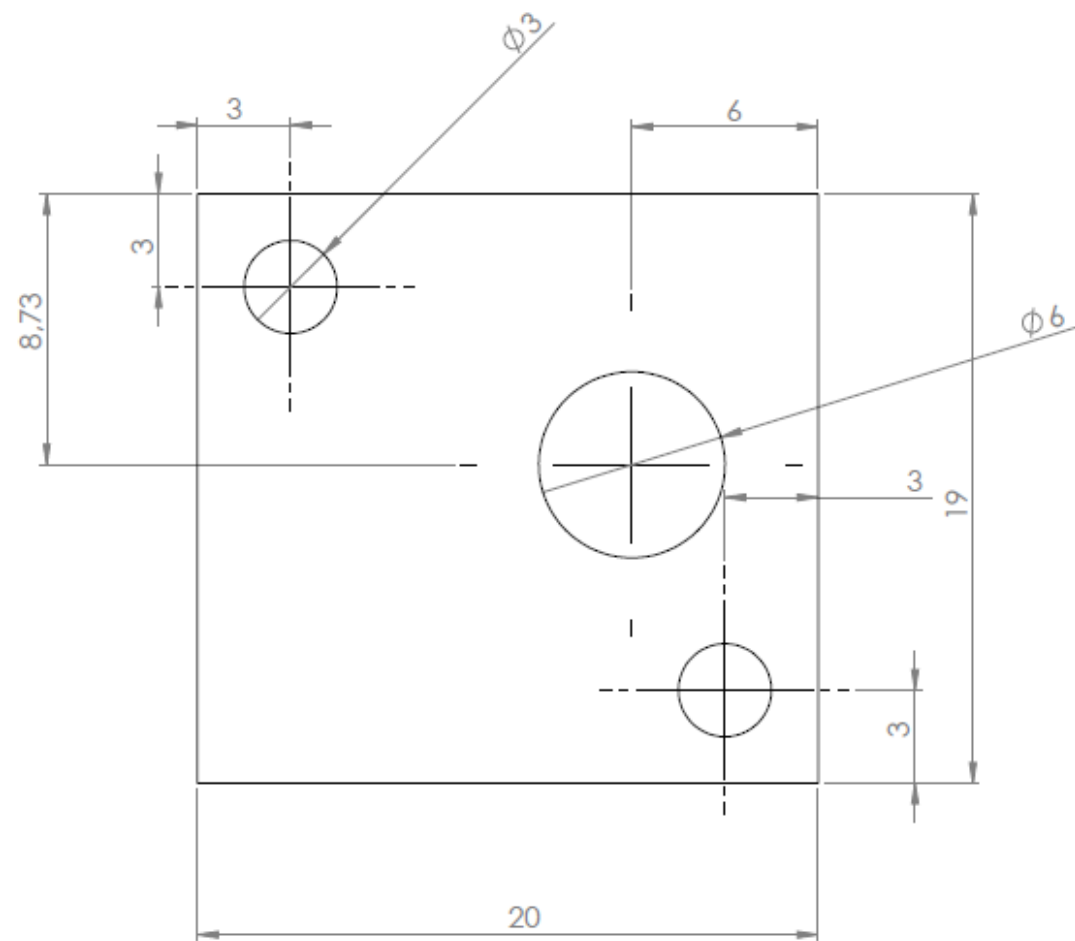


Climbot.01.07

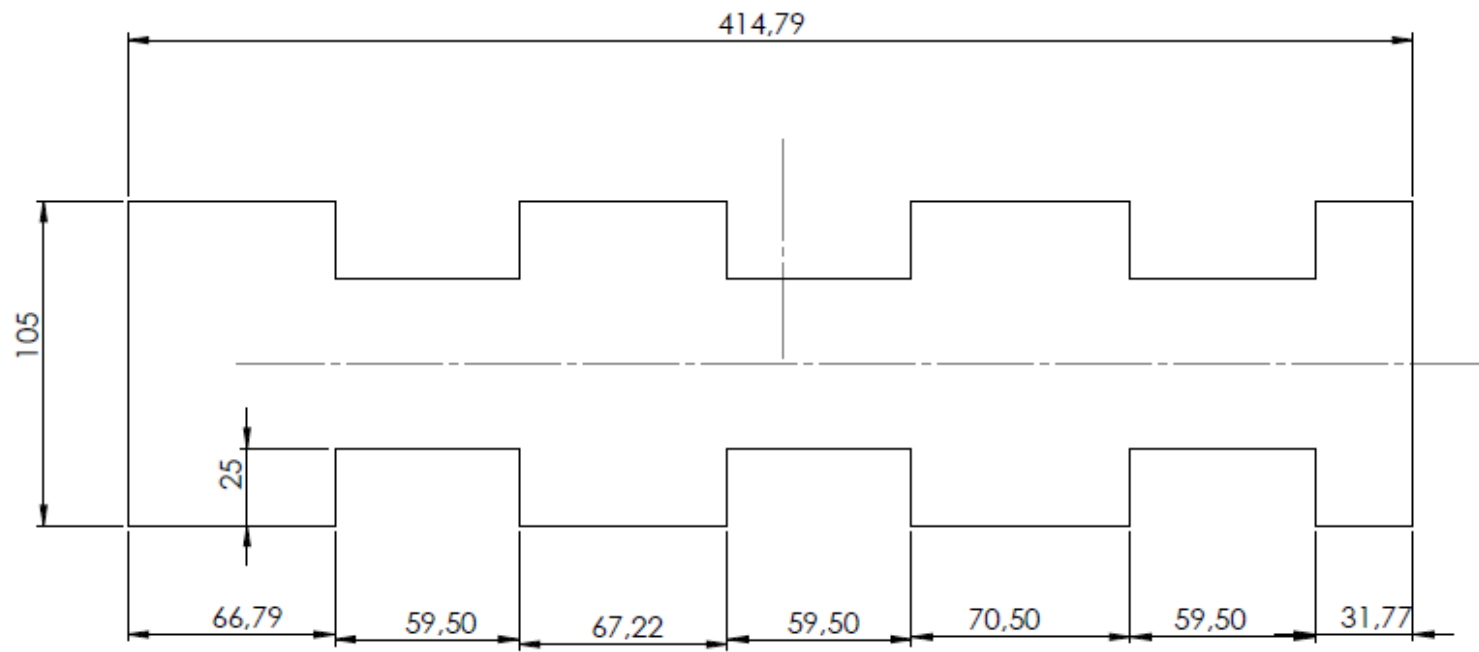


SECCIÓN D-D  
ESCALA 3 : 1

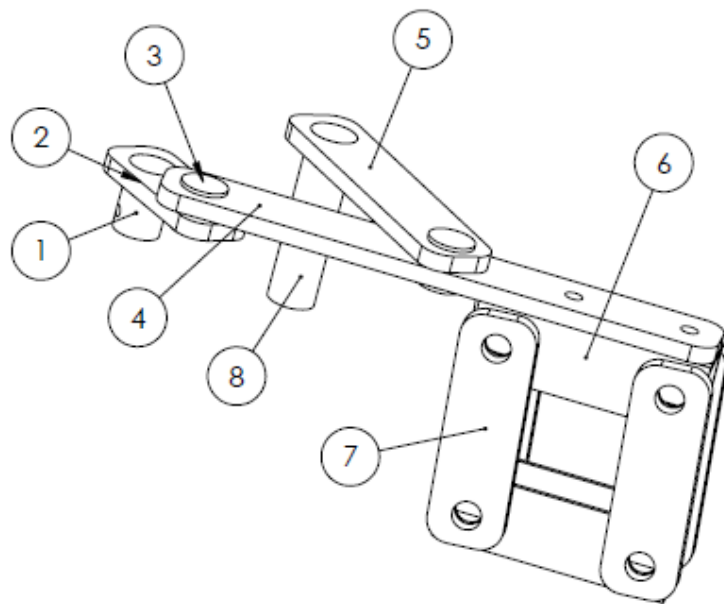
Observaciones	MATERIAL:	Pieza:	A4
Cotas en mm	Cilindro aluminio 5/16"	Acoples Micromotor	
	CODIGOS: Climbot.02.03, Climbot.01.07	ESCALA:3:1	HOJA 8 DE 14



Observaciones	MATERIAL:	Pieza:	A4
Cotas en milímetros	Platina aluminio 1/8" x3/4"	Placa base Potenciometro	
	CODIGO: Cimbot.02.04	ESCALA: 1:1	HOJA 9 DE 14



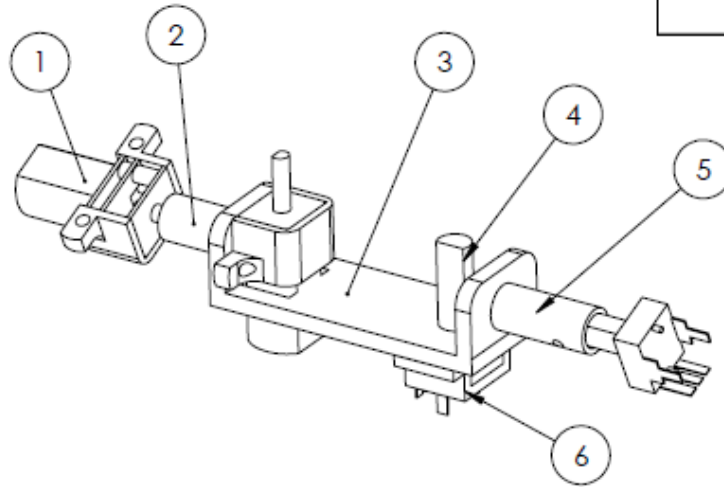
Observaciones	MATERIAL:	PIEZAS:	A4
Cotas en milímetros	Placa de aluminio de 1/8' de espesor	Estructura	
	CODIGO: Cimbot.03.04	ESCALA: 1:2	HOJA 10 DE 14



Numero	Pieza	Codigo
1	Acople micromotor-manivela	Climbot.01.07
2	Manivela	Climbot.01.01
3	Pasador	Climbot.01.08
4	biela	Climbot.01.03
5	balancin	Climbot.01.02
6	Base suspension	Climbot.01.05
7	Barra suspension	Climbot.01.04
8	Acople potenciometro balancin	Climbot.01.06

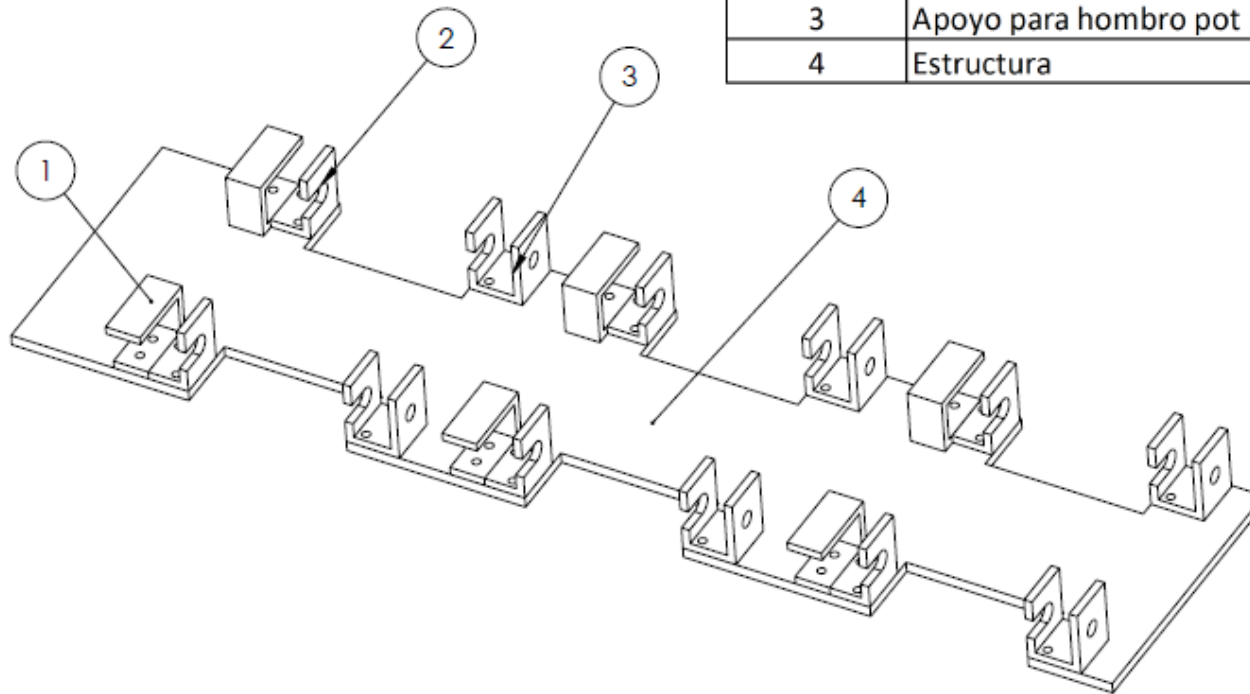
Observaciones	MATERIAL:	SUBENSAMBLE:	A4
Cotas en milímetros	Aluminio	Mecanismo Hoekens	
	CODIGO: Climbot.01.00	ESCALA:1:1	HOJA 11 DE 14

Numero	Pieza	Codigo
1	Micromotor	Climbot.02.05
2	Acople micromotor-Hombro	Climbot.02.03
3	Hombro	Climbot.02.01
4	Potenciometro	Climbot.02.06
5	Acople potenciometro Hombro	Climbot.02.02
6	Acople potenciometro	Climbot.02.07

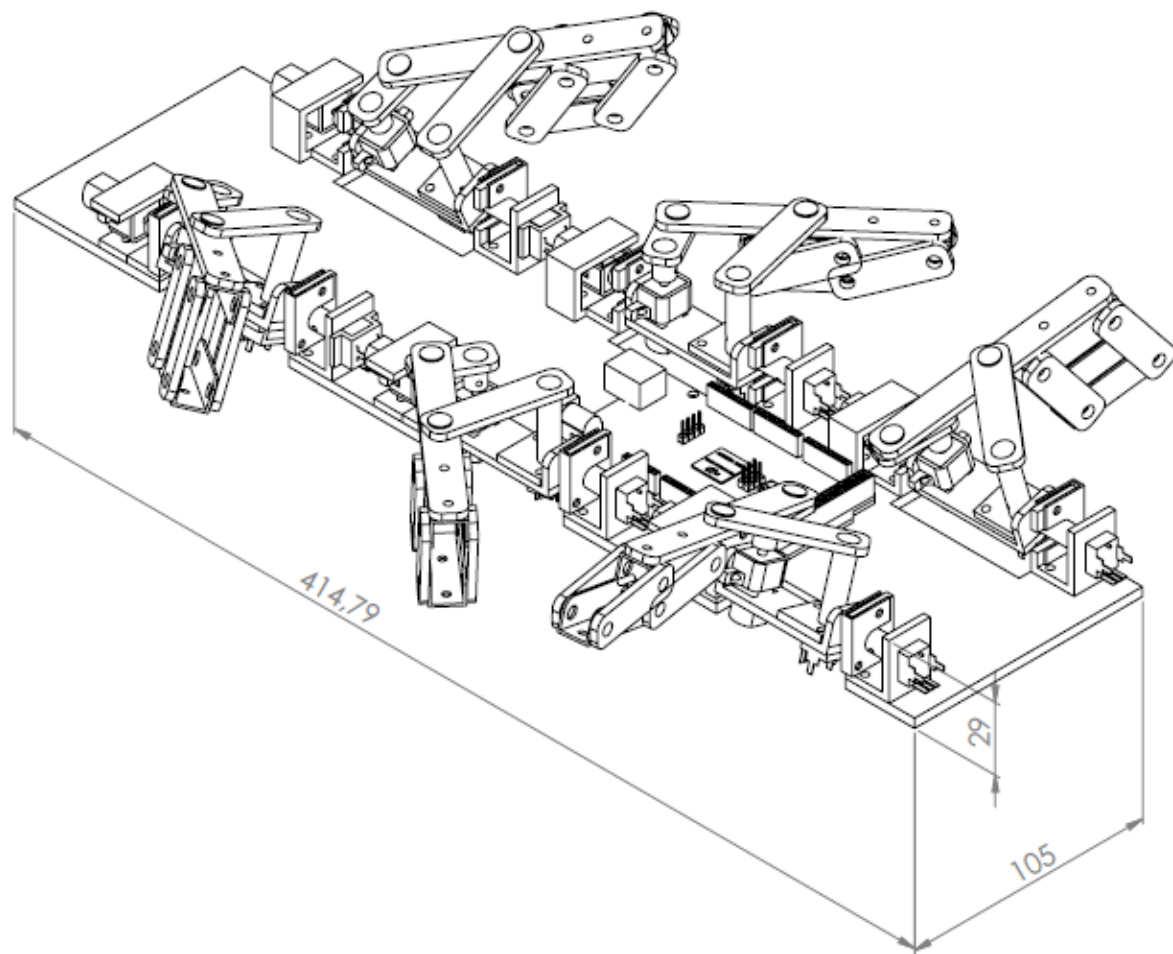


Observaciones	MATERIAL:	SUBENSAMBLE:	A4
Cotas en milímetros	Aluminio	Hombro	
	CODIGO: Climbot.02.00	ESCALA:1:1	HOJA 12 DE 14

Numero	Pieza	Codigo
1	Base para micromotor	Climbot.03.03
2	Apoyo para hombro motor	Climbot.03.02
3	Apoyo para hombro pot	Climbot.03.01
4	Estructura	Climbot.03.04



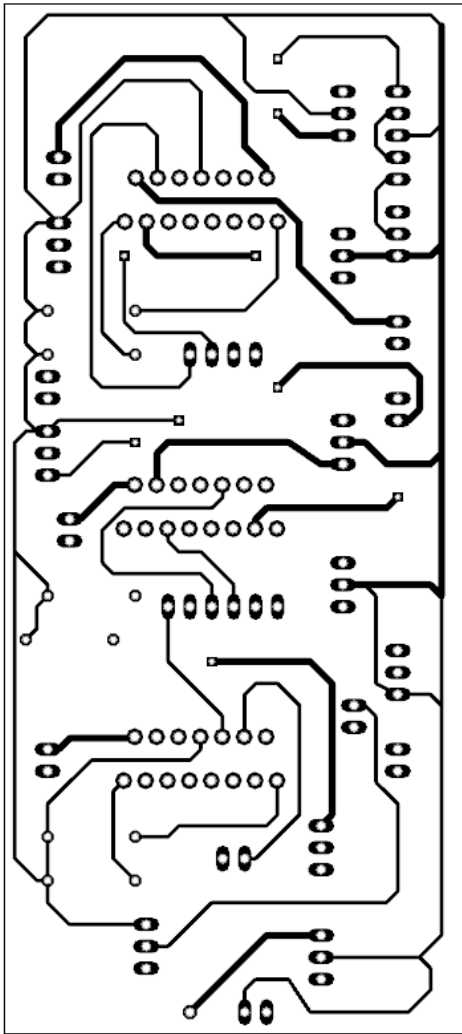
Observaciones	MATERIAL: ALUMINIO	SUBENSAMBLE	A4
Cotas en milímetros		Estructura base	
	CODIGO: Climbot.03.00	ESCALA:1:2	HOJA 13 DE 14



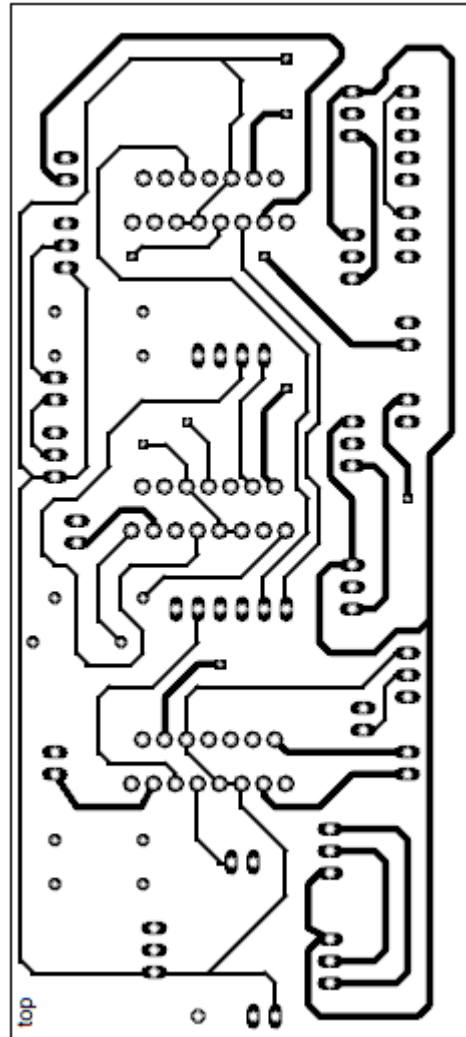
Observaciones	MATERIAL:	ENSAMBLE	CLIMBOT	A4
Cotas en milímetros	VARIOS			
	CODIGO: Climbot.00.00	ESCALA: 1:2	HOJA 14 DE 14	

# ANEXO B. CIRCUITO IMPRESO

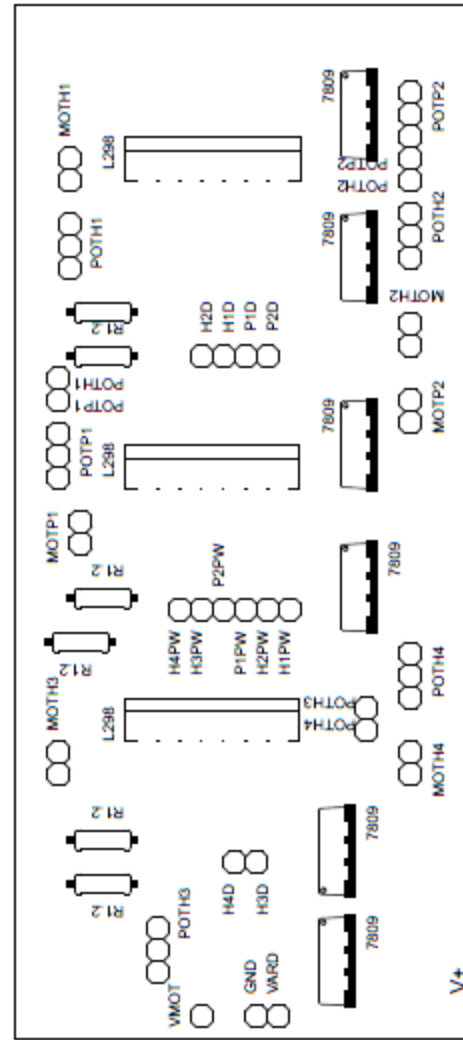
## PCB DELANTERA



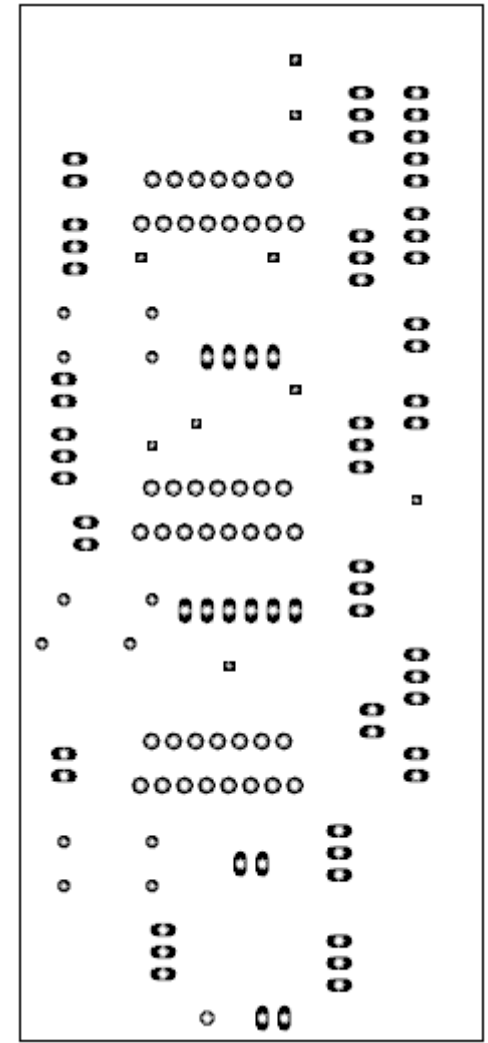
Capa Top



Capa Bottom



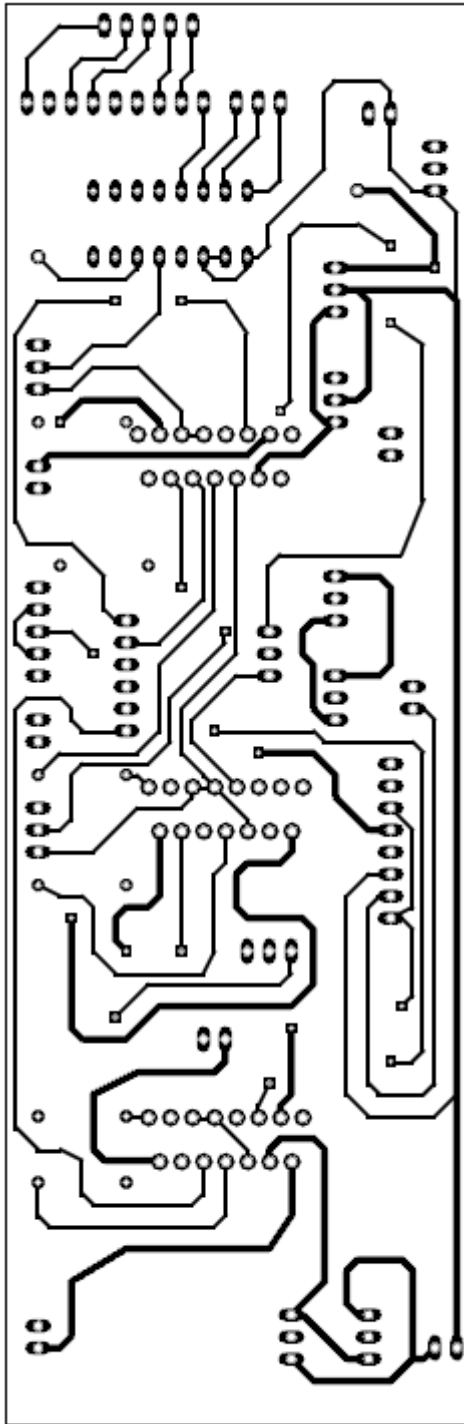
Capa textos



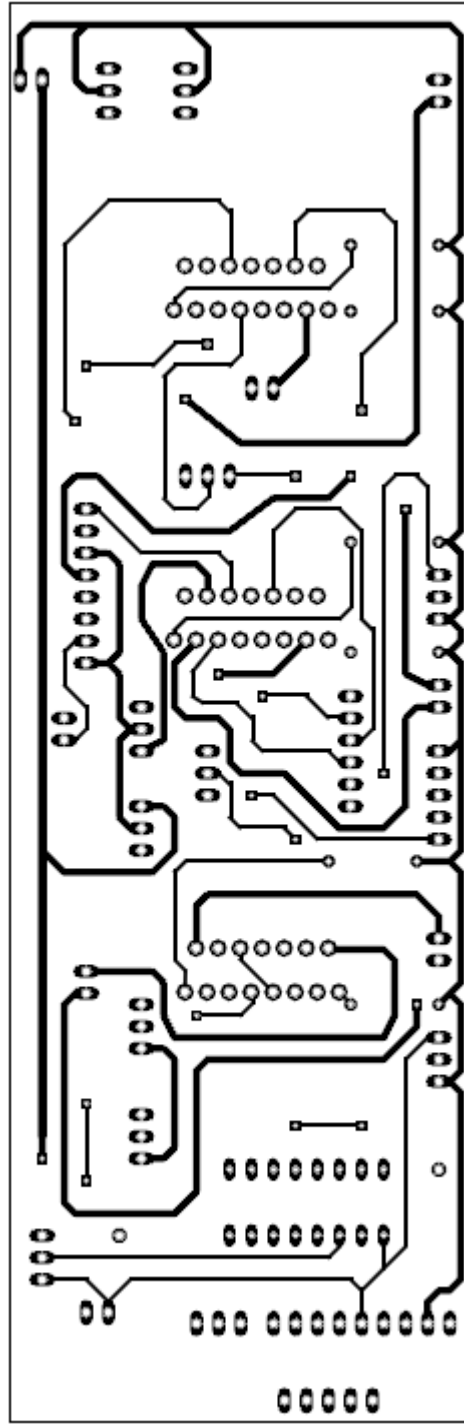
Capa pads Vias

# PCB TRASERA

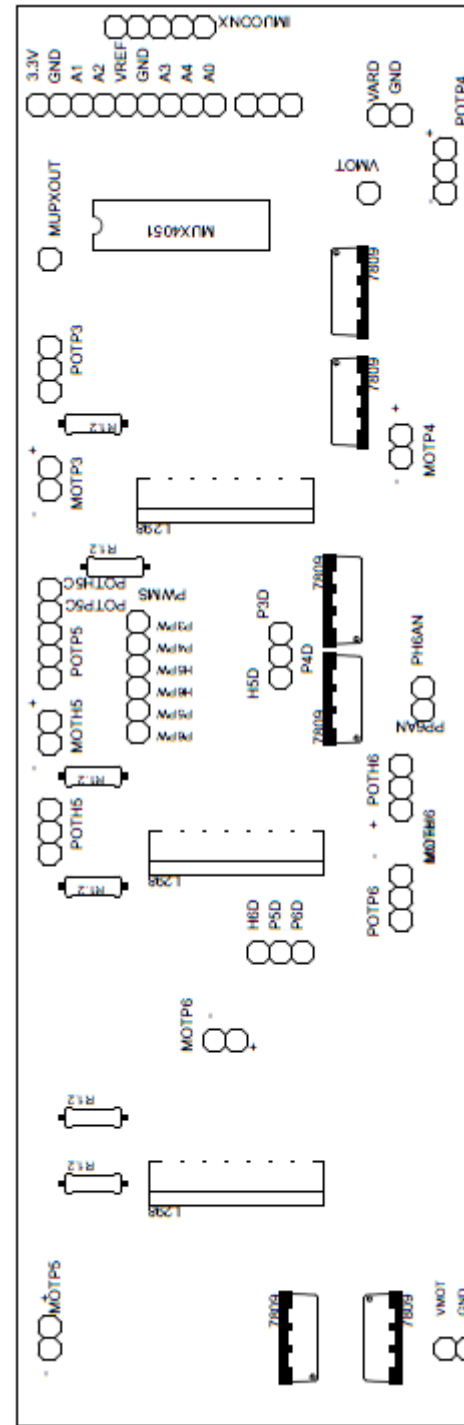
## Capa Top



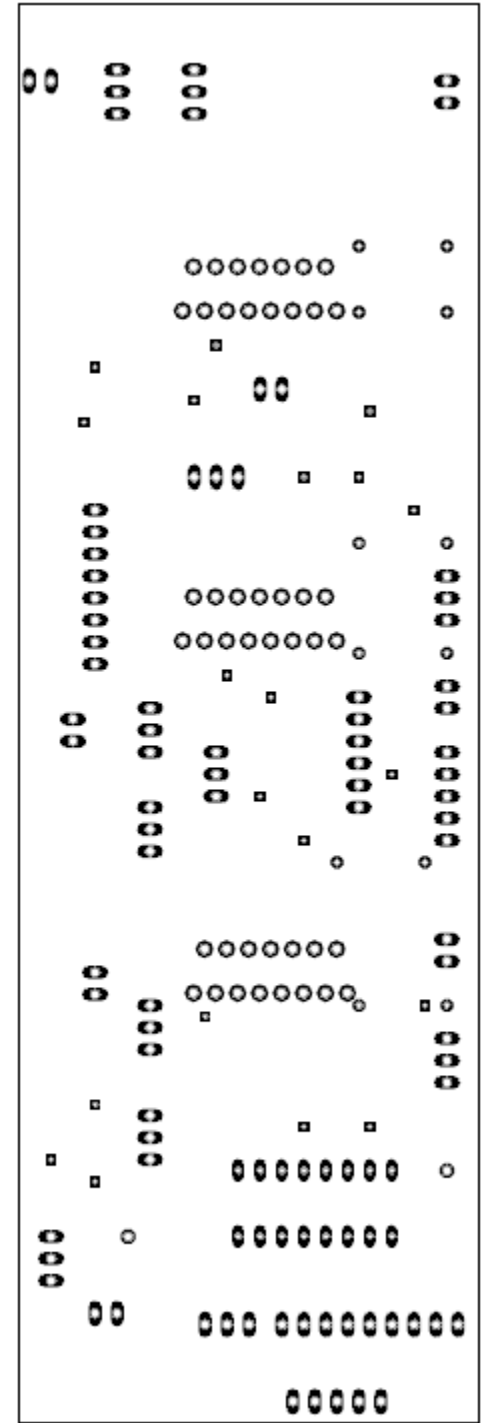
## Capa Bottom



## Capa textos

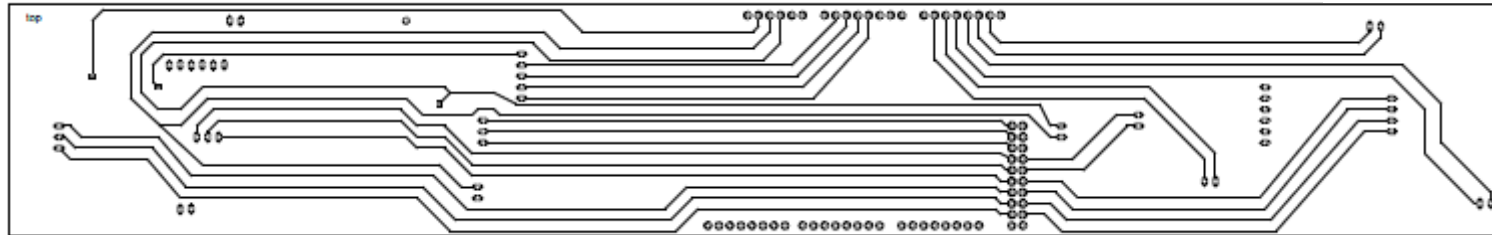


## Capa pads Vias

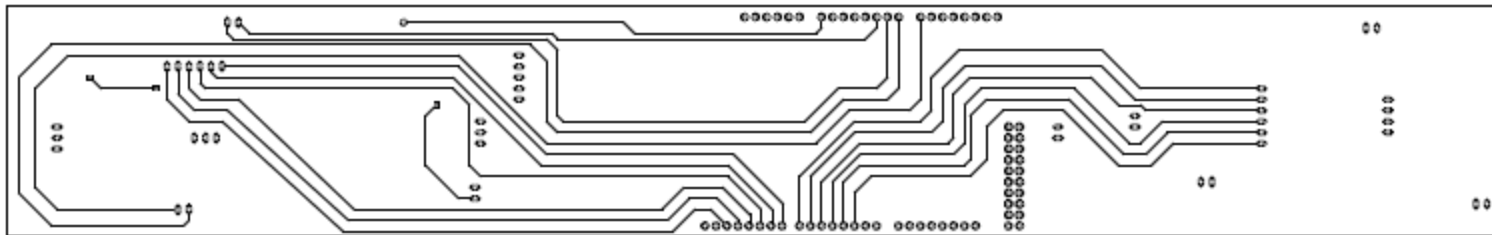


# PCB UNION

Capa Top



Capa Bottom



Capa pads Vias

