



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA



**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y  
MECÁNICA**

**CARRERA DE INGENIERÍA MECATRÓNICA**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN  
DEL TÍTULO DE INGENIERÍA MECATRÓNICA**

**AUTORES: BENALCÁZAR REDROBÁN, PAOLA ROCÍO  
NOBOA ENCALADA, ALEXIS RAÚL**

**TEMA: DISEÑO Y CONSTRUCCIÓN DE UNA  
PLATAFORMA ROBÓTICA ZOOMÓRFICA DE CUATRO  
EXTREMIDADES CON DOCE GRADOS DE LIBERTAD**

**DIRECTOR: ING. TOBAR QUEVEDO, JOHANNA BELÉN**

**SANGOLQUÍ**

**2017**



**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y  
MECÁNICA**

**CARRERA DE INGENIERÍA MECATRÓNICA**

**CERTIFICACIÓN**

Certifico que el trabajo de titulación, "**DISEÑO Y CONSTRUCCIÓN DE UNA PLATAFORMA ROBÓTICA ZOOMÓRFICA DE CUATRO EXTREMIDADES CON DOCE GRADOS DE LIBERTAD**" realizado por los señores **PAOLA ROCÍO BENALCÁZAR REDROBÁN** y **ALEXIS RAÚL NOBOA ENCALADA**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar a los señores **PAOLA ROCÍO BENALCÁZAR REDROBÁN** y **ALEXIS RAÚL NOBOA ENCALADA** para que lo sustenten públicamente.

**Sangolquí, 09 de marzo del 2017**

Ing. Johanna Belén Tobar Quevedo

**DIRECTORA**



**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA  
Y MECÁNICA**

**CARRERA DE INGENIERÍA MECATRÓNICA**

**AUTORÍA DE RESPONSABILIDAD**

Nosotros, **PAOLA ROCÍO BENALCÁZAR REDROBÁN**, con cédula de identidad N° 1725781882 y **ALEXIS RAÚL NOBOA ENCALADA**, con cédula de identidad N° 1719950261, declaramos que este trabajo de titulación "**DISEÑO Y CONSTRUCCIÓN DE UNA PLATAFORMA ROBÓTICA ZOOMÓRFICA DE CUATRO EXTREMIDADES CON DOCE GRADOS DE LIBERTAD**" ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaramos que este trabajo es de nuestra autoría, en virtud de ello nos declaramos responsables del contenido, veracidad y alcance de la investigación mencionada.

**Sangolquí, 09 de marzo del 2017**

Paola Rocío Benalcázar Redrobán  
C.C:1725781882

Alexis Raúl Noboa Encalada  
C.C:1719950261



DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y  
MECÁNICA

CARRERA DE INGENIERÍA MECATRÓNICA

**AUTORIZACIÓN**

Nosotros, **PAOLA ROCÍO BENALCÁZAR REDROBÁN** y **ALEXIS RAÚL NOBOA ENCALADA** autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación "**DISEÑO Y CONSTRUCCIÓN DE UNA PLATAFORMA ROBÓTICA ZOOMÓRFICA DE CUATRO EXTREMIDADES CON DOCE GRADOS DE LIBERTAD**" cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Sangolquí, 09 de marzo del 2017

Paola Rocío Benalcázar Redrobán  
C.C:1725781882

Alexis Raúl Noboa Encalada  
C.C: 1719950261

## DEDICATORIA

Dedico este proyecto a mi padre Julio, quien desde pequeña me ha enseñado el valor del conocimiento de la naturaleza que nos rodea, así como el respeto a cada ser viviente, encontrando inspiración y deleite en las cosas simples de la vida. A mi madre Rosario, quien ha sido mi soporte incondicional durante todo este tiempo, siendo también mi guía y fuente de amor, entrega y alegría para mí y toda mi familia. A mis hermanos, quienes me han brindado su apoyo y enseñanzas convirtiéndose en otros padres a los que también respeto y amo con todo mi corazón. Y a Marlon, quien con paciencia y amor me enseñó el camino de Dios.

*Paola Benalcázar*

Dedico este logro alcanzado a Dios, a mi madre Soraya que durante toda mi vida ha sido el soporte fundamental de todos mis estudios académicos y siempre me ha ayudado a salir adelante, a mi padre Milton quien siempre estuvo apoyándome y dándome todos los medios necesarios para mis estudios y proyectos a futuro. Menciono también a mi abuela Dolores que siempre estuvo alentándome para superarme y a mis amigos que desde distintas partes del mundo estuvieron animándome a culminar con éxito mi carrera universitaria.

*Raúl Noboa*

## **AGRADECIMIENTO**

Agradezco a Dios, por sostenerme en los momentos difíciles, por brindarme su completo amor y ternura, y por haberme dado una familia maravillosa la cual ha sido mi fuente de inspiración para salir adelante.

A mis padres, por su amor y apoyo constante, por haber creado un hogar en el que me pude sentir segura y capaz de cumplir mis metas.

A mis hermanos, quienes siempre me impulsaron a salir adelante y ser mejor cada día.

A mis amigos, por acompañarme en esta etapa de mi vida y llenar cada día de experiencias y alegría.

A la Ingeniera Johanna Tobar y al Ingeniero David Loza, por su apoyo y conocimientos impartidos que permitieron la realización de este proyecto.

A mis profesores, quienes durante toda mi vida universitaria me brindaron sus conocimientos.

*Paola Benalcázar*

En primer lugar agradezco a Dios que siempre me brindó la salud, a mi madre Soraya que estuvo en los momentos más difíciles, a mi padre Milton que siempre me apoyo en todo ámbito, a mis amigos que se encuentran dentro o fuera del país que siempre me brindaron ayuda académica o emocional para poder lograr mis metas.

A mi compañera de tesis Paola quien fue parte fundamental para resolver todos los problemas que se nos presentaron, y a nuestra directora de tesis la Ingeniera Johana Tobar y al Ingeniero David Loza que siempre nos guiaron y asistieron en partes complejas de nuestro proyecto.

*Raúl Noboa*

## ÍNDICE DE CONTENIDO

CERTIFICADO DE LA ELABORACIÓN DEL PROYECTO .....	i
AUTORÍA DE RESPONSABILIDAD .....	ii
AUTORIZACIÓN.....	iii
DEDICATORIA .....	iv
AGRADECIMIENTO .....	v
ÍNDICE.....	vi
ÍNDICE DE FIGURAS.....	xi
ÍNDICE DE TABLAS .....	xv
RESUMEN .....	xvii
ABSTRACT.....	xviii
CAPÍTULO 1. GENERALIDADES.....	1
1.1    PLANTEAMIENTO DEL PROBLEMA .....	1
1.2    ANTECEDENTES .....	2
1.2.1    DESARROLLO DE INVESTIGACIONES DE ROBOTS .....	2
HEXÁPODOS EN LA UNIVERSIDAD DE LAS FUERZAS ARMADAS....	2
ESPE.....	2
1.3    JUSTIFICACIÓN E IMPORTANCIA.....	3
1.4    ÁREA DE INFLUENCIA .....	4
1.5    OBJETIVOS .....	5
1.5.1    OBJETIVO GENERAL .....	5
1.5.2    OBJETIVOS ESPECÍFICOS.....	5
1.6    ALCANCE .....	6
1.7    ESTRUCTURA DEL DOCUMENTO .....	7
CAPÍTULO 2. ESTADO DEL ARTE .....	8
2.1    INTRODUCCIÓN .....	8

2.2	ROBOTS MÓVILES .....	8
2.2.1	PERSPECTIVA HISTÓRICA DE ROBOTS CAMINANTES .....	8
2.2.2	DESARROLLO TECNOLÓGICO DE ROBOTS CAMINANTES .....	12
2.3	ANÁLISIS DE PROYECTOS REALIZADOS EN EL PAÍS.....	15
2.4	LOCOMOCIÓN TERRESTRE DE INSECTOS .....	29
2.4.1	TIPOS DE PATAS SEGÚN EL MEDIO EN QUE SE .... DESARROLLAN .....	31
2.4.2	COEFICIENTES DE CORRELACIÓN DE LONGITUD ENTRE..... LOS SEGMENTOS DE LA PATA .....	32
2.5	ESTRUCTURAS DE CONTROL DEL ROBOT ZOOMÓRFICO.....	32
2.5.1	GENERADOR CENTRAL DE PATRONES.....	33
2.5.2	ALGORITMO EVOLUTIVO .....	34
2.6	CINEMÁTICA DE LAS EXTREMIDADES DEL ROBOT .... ZOOMÓRFICO .....	35
2.6.1	MÉTODOS DE ANÁLISIS CINEMÁTICO.....	35
2.6.2	CONSIDERACIONES INICIALES PARA EL ANÁLISIS.....	36
2.7	ANÁLISIS GLOBAL.....	38
CAPÍTULO 3. DISEÑO MECATRÓNICO .....		40
3.1	MODELAMIENTO ROBÓTICO .....	40
3.1.1	ANÁLISIS CINEMÁTICO.....	40
3.1.2	PARÁMETROS DEL ROBOT .....	45
3.1.3	ESPACIO DE TRABAJO.....	46
3.1.4	GENERACIÓN DE LOCOMOCIÓN DEL ROBOT.....	49
3.1.5	GENERACIÓN DE LA TRAYECTORIA DE LA EXTREMIDAD.....	57
3.1.6	ANÁLISIS DE ESTABILIDAD ESTÁTICA DEL CUADRÚPEDO .....	57
3.1.7	ANÁLISIS DINÁMICO DEI ROBOT.....	61

3.2	COMPONENTES MECÁNICOS.....	73
3.2.1	ACTUADORES .....	73
3.2.2	ESTRUCTURA.....	74
3.2.3	ANÁLISIS MECÁNICO DEL ROBOT .....	79
3.2.4	ANÁLISIS DE LAS PIEZAS.....	88
3.3	DISEÑO DEL SISTEMA DE POTENCIA.....	91
3.3.1	FUENTE DE ALIMENTACIÓN .....	91
3.3.2	DISEÑO DE LA PLACA PCB DE POTENCIA.....	93
3.4	DISEÑO DEL SISTEMA DE CONTROL .....	96
3.4.1	ARQUITECTURA DEL SISTEMA .....	96
3.4.2	TARJETA DE CONTROL.....	97
3.4.3	CAMARA IP WIRELESS .....	98
3.4.4	DISEÑO DE LA PLACA PCB DE CONTROL.....	100
3.5	DESARROLLO DE SOFTWARE.....	102
3.5.1	ROBOT OPERATING SYSTEM (ROS).....	102
3.5.2	ARQUITECTURA DEL SOFTWARE PROYECTO.....	111
3.6	DISEÑO DEL SISTEMA DE COMUNICACIÓN .....	114
3.6.1	MÓDULOS DE RADIOFRECUENCIA DE COMUNICACIÓN..... INALÁMBRICA.....	114
3.7	DISEÑO DE LA INTERFAZ HOMBRE MÁQUINA (HMI) .....	118
3.7.1	METODOLOGÍA EN EL DESARROLLO DEL PROYECTO .....	119
3.7.2	SOFTWARE QT CREATOR EN EL DESARROLLO DE LA ..... INTERFAZ .....	122
3.7.3	MODELACIÓN Y SIMULACIÓN .....	124
	CAPÍTULO 4. CONSTRUCCIÓN E IMPLEMENTACIÓN .....	133

4.1	IMPLEMENTACIÓN DE LA ESTRUCTURA Y COMPONENTES MECÁNICOS .....	133
4.1.1	TIEMPOS DE CORTE E IMPRESIÓN .....	133
4.1.2	TIEMPOS DE ELABORACIÓN .....	138
4.1.3	TIEMPOS DE MONTAJE .....	141
4.2	IMPLEMENTACIÓN DEL SISTEMA DE POTENCIA .....	141
4.2.1	TIEMPO DE ELABORACIÓN DE PUNTOS DE SOLDADURA.....	142
4.3	IMPLEMENTACIÓN DEL SISTEMA DE CONTROL.....	142
4.3.1	TIEMPO DE ELABORACIÓN DE PUNTOS DE SOLDADURA.....	143
4.4	IMPLEMENTACIÓN DEL SISTEMA DE COMUNICACIÓN .....	143
4.4.1	DIAGRAMA DE COMUNICACIÓN SERIAL NODO PYTHON .....	143
4.4.2	CONFIGURACIÓN DE LOS PARÁMETROS DE..... COMUNICACIÓN DE LA PLACA DE CONTROL.....	144
4.5	PROGRAMACIÓN DEL SISTEMA DE CONTROL Y HMI .....	145
4.5.1	PROGRAMACIÓN DEL SISTEMA DE CONTROL .....	145
4.5.2	PROGRAMACIÓN DEL HMI.....	146
	CAPÍTULO 5. PRUEBAS Y RESULTADOS .....	150
5.1	PRUEBAS MECÁNICAS.....	150
5.1.1	PRUEBAS DE LOS SERVOMOTORES .....	150
5.1.1.1	PROCEDIMIENTO .....	150
5.2	PRUEBAS DE LA PLACAS DE POTENCIA Y DE CONTROL.....	152
5.2.1	PROCEDIMIENTO.....	152
5.2.2	RESULTADOS.....	153
5.3	PRUEBAS DE COMUNICACIÓN ROS - MÓDULO RF – PLACA..... DE CONTROL .....	153
5.3.1	PROCEDIMIENTO.....	153

5.3.2	RESULTADO .....	154
5.4	PRUEBAS DE LA RESPUESTA CINEMÁTICA INVERSA..... DEL ROBOT .....	155
5.4.1	PROCEDIMIENTOS.....	155
5.4.2	RESULTADOS EXTREMIDAD 1.....	155
5.4.3	RESULTADOS EXTREMIDAD 2.....	156
5.4.4	RESULTADOS EXTREMIDAD 3.....	157
5.4.5	RESULTADOS EXTREMIDAD 4.....	158
5.5	PRUEBAS DE CAMINATA DISCONTINUA DEL ROBOT .....	159
	CONCLUSIONES .....	161
	RECOMENDACIONES.....	163
	BIBLIOGRAFÍA.....	164

## ÍNDICE DE FIGURAS

Figura 1. Mecanismo de diez barras de la extremidad .....	9
Figura 2. Mecanismo de Armadura de Caballero.....	9
Figura 3. Plataforma Caminante de Chebyshev .....	10
Figura 4. <i>The Mechanical Horse</i> .....	11
Figura 5 Cuadrúpedo de Bechtolsheim.....	11
Figura 6. Robot móvil terminado .....	16
Figura 7. Esquema de conexiones del robot móvil .....	16
Figura 8. Pruebas sobre terreno irregular de robot móvil.....	17
Figura 9. Caminata sobre superficie lisa - Robot hexápodo .....	18
Figura 10. Configuración inicial del cuadrúpedo .....	19
Figura 11. Secuencia de locomoción del cuadrúpedo .....	20
Figura 12. Robot cuadrúpedo en funcionamiento .....	21
Figura 13. Diseño de la pata del robot hexápodo .....	22
Figura 14. Conexiones al receptor R6006FS Futaba.....	23
Figura 15. Robot hexápodo-Universidad de las Fuerzas Armadas.....	23
Figura 16. Robot móvil Universidad Estatal Península de Santa Elena.....	24
Figura 17. Ensamblaje del robot hexápodo Universidad de Queensland ....	26
Figura 18. Robot hexápodo de la universidad de Santa Elena .....	27
Figura 19. Estructura del cuadrúpedo Universidad de Chimborazo .....	28
Figura 20. Robot móvil para la inspección de cultivos de haba .....	29
Figura 21. Tipos de uniones en artrópodos .....	30
Figura 22. Partes de una extremidad de un insecto.....	30
Figura 23. Tipo de patas de insectos .....	31
Figura 24. Arquitectura del sistema de control de la locomoción .....	33
Figura 25. Esquema de la Robótica Evolutiva .....	34
Figura 26. Distribución de uniones y eslabones de la pata.....	40
Figura 27. Parámetros de Denavit Hartenberg – Modelo real.....	41
Figura 28. Parámetros de Denavit Hartenberg de la extremidad del robot. .	41
Figura 29. Límites angulares mínimos de las uniones 2 y 3. ....	47
Figura 30. Límites angulares máximos de las uniones 2 y 3. ....	48

Figura 31. Espacio de trabajo Método Monte Carlo.....	49
Figura 32. Definiciones geométricas.....	51
Figura 33. Secuencias posibles de las extremidades .....	51
Figura 34. Caminata discontinua de dos fases .....	53
Figura 35. Caminata discontinua giratoria .....	56
Figura 36. Márgenes de estabilidad y polígono de estabilidad .....	58
Figura 37. Puntos de soporte que determinan la diagonal del SLSM .....	60
Figura 38. Esquema de una pata para análisis de energías.....	63
Figura 39. Placa inferior del cuerpo del robot .....	75
Figura 40. Placa superior del cuerpo del robot .....	75
Figura 41. Estructura de la coxa del cuadrúpedo.....	76
Figura 42. Bracket servo fémur-1.....	77
Figura 43. Bracket servo fémur-2.....	77
Figura 44. Bracket servo tibia .....	78
Figura 45. Pieza de soporte servo tibia.....	78
Figura 46. Estructura de la tibia del robot .....	79
Figura 47. Reacciones normales de cada pata.....	80
Figura 48. Posición inicial del robot .....	80
Figura 49. Sistema de referencia del robot .....	81
Figura 50. Diagrama de cuerpo libre del robot (Plano XZ).....	83
Figura 51. Diagrama de cuerpo libre del robot (Plano YZ).....	84
Figura 52. Diagrama de cuerpo libre del eslabón 3 .....	85
Figura 53. Diagrama de cuerpo libre del eslabón 2 .....	86
Figura 54. Diagrama del cuerpo libre del eslabón 1.....	87
Figura 55. Diagrama de esfuerzos del eslabón 3.....	88
Figura 56. Factor de seguridad en el eslabón 3.....	88
Figura 57. Diagrama de esfuerzos del eslabón 2.....	89
Figura 58. Factor de seguridad del eslabón 2.....	89
Figura 59. Diagrama de esfuerzos del eslabón 1.....	90
Figura 60. Factor de seguridad del eslabón 1.....	90
Figura 61. Gráfica Corriente vs Sección transversal para diseño de PCB...	94
Figura 62. Gráfica Ancho del conductor vs Sección transversal.....	94

Figura 63. Placa de potencia .....	95
Figura 64. Arquitectura del sistema de control.....	96
Figura 65. Conexión LAN de la cámara .....	99
Figura 66. Conexión WLAN de la cámara.....	99
Figura 67. Placa de control .....	101
Figura 68. Nivel del Sistema de Archivos de ROS .....	104
Figura 69. Nivel Gráfico Computacional .....	105
Figura 70. Nivel gráfico del funcionamiento de un robot.....	107
Figura 71. Mensaje de envío de datos del nodo <i>joystick</i> a ROS.....	109
Figura 72. Estructura del proyecto .....	111
Figura 73. Módulos de radiofrecuencia 3DR <i>Radio Telemetry</i> .....	114
Figura 74. Interfaz para configurar el 3DR <i>Radio Telemetry</i> .....	116
Figura 75. Conexiones del módulo esclavo .....	118
Figura 76. Metodología GEDIS .....	119
Figura 77. Distribución de botones del HMI .....	120
Figura 79. Configuración de un link .....	125
Figura 80. Configuración de una unión .....	126
Figura 81. Simulador RVIZ .....	127
Figura 82. Modelo del Robot RVIZ.....	128
Figura 83. Comando para cargar el modelo URDF al simulador .....	129
Figura 84. Ejecución de nodos para el simulador .....	130
Figura 85. Posición inicial. ....	130
Figura 86. Traslación hacia adelante. ....	131
Figura 87. Rotación hacia la derecha. ....	131
Figura 88. Rotación hacia la izquierda.....	132
Figura 89. Trayectoria aleatoria. ....	132
Figura 90. Tiempo de corte de la pieza: cuerpo pieza superior .....	134
Figura 91. Tiempo de corte de la pieza: cuerpo pieza inferior .....	134
Figura 92. Tiempo de impresión de la pieza: <i>bracket</i> servo coxa 1 .....	134
Figura 93. Tiempo de impresión de la pieza: <i>bracket</i> servo coxa 2 .....	135
Figura 94. Tiempo de corte de la pieza: <i>bracket</i> servo fémur 1 .....	135
Figura 95. Tiempo de corte de la pieza: <i>bracket</i> servo fémur 1 .....	135

Figura 96. Tiempo de corte de la pieza: <i>bracket</i> servo tibia.....	136
Figura 97. Tiempo de impresión de la pieza: soporte servo tibia .....	136
Figura 98. Tiempo de corte de la pieza: case batería 1 .....	136
Figura 99. Tiempo de corte de la pieza: case batería 2 .....	137
Figura 100. Tiempo de corte de la pieza: case batería 3 .....	137
Figura 101. Tiempo de corte de la pieza: case batería 4 .....	137
Figura 102. Diagrama de proceso de <i>bracket</i> servo coxa 1 .....	138
Figura 103. Diagrama de proceso de <i>bracket</i> servo coxa 2.....	139
Figura 104. Diagrama de proceso de soporte servo tibia .....	139
Figura 105. Programación del nodo <i>serial_node.py</i> .....	144
Figura 106. Diagrama de flujo de la programación del sistema de control	145
Figura 107. Posición inicial del cuadrúpedo.....	146
Figura 108. Posición arriba del cuadrúpedo .....	146
Figura 109. Diagrama de flujo de la programación del HMI .....	147
Figura 110. Diagrama de flujo del proceso de rotación.....	148
Figura 111. Diagrama de flujo del proceso de traslación .....	149

## ÍNDICE DE TABLAS

Tabla 1 Avance Tecnológico Cronológico de Robots Caminantes .....	12
Tabla 2 Parámetros de DH de la extremidad del robot cuadrúpedo .....	42
Tabla 3 Parámetros físicos generales del robot.....	45
Tabla 4 Definición de las longitudes del robot .....	46
Tabla 5 Valores tomados en cuenta en la generación de movimiento.....	52
Tabla 6 Márgenes de Estabilidad.....	60
Tabla 7 Características de los eslabones del robot .....	72
Tabla 8 Características del servo motor .....	74
Tabla 9. Elección del material .....	79
Tabla 10 Parámetros de peso y espacio de trabajo del robot.....	82
Tabla 11 Centro de masa de cada eslabón en cada extremidad.....	82
Tabla 12 Baterías principales disponibles en el mercado .....	91
Tabla 13 Parámetros de densidad de almacenamiento de baterías.....	91
Tabla 14 Matriz de selección de baterías .....	92
Tabla 15 Elementos electrónicos principales y consumos de energía.....	92
Tabla 16 Características Técnicas del Arduino MEGA 2560 .....	97
Tabla 17 Especificaciones técnicas de la cámara.....	98
Tabla 18 Terminales utilizados en la placa de control .....	100
Tabla 19 Comandos de verificación y funcionamiento del <i>joystick</i> .....	108
Tabla 20 Nodos de lanzamiento en el archivo <i>launch</i> .....	113
Tabla 21 Características técnicas de 3DR <i>Radio Telemetry</i> .....	115
Tabla 22 Comportamiento de las LEDS indicadores .....	118
Tabla 23 Colores utilizados en la interfaz .....	121
Tabla 24 Comandos de la HMI .....	121
Tabla 25 Funciones de los botones de la interfaz.....	123
Tabla 26 Tipos de uniones.....	127
Tabla 27 Funciones de <i>Robot Model</i> .....	129
Tabla 28 Material y operación de las piezas que conforman la estructura	133
Tabla 29 Tiempos de operaciones de las piezas de la estructura .....	138
Tabla 30 Tiempos de fabricación de cada pieza del robot.....	140

Tabla 31 Tiempo de taladrado de la pieza bracket servo accesorio .....	140
Tabla 32 Tiempos de montaje de las partes del robot .....	141
Tabla 33 Lista de elementos de la placa de potencia. ....	142
Tabla 34 Tiempos de elaboración de puntos de la placa de potencia .....	142
Tabla 35 Lista de elementos de la placa de control. ....	143
Tabla 36 Pruebas de los límites angulares de los servomotores sin carga	151
Tabla 37 Pruebas de los límites angulares de los servos corregidos .....	151
Tabla 38 Pruebas de los servomotores con el robot en movimiento.....	151
Tabla 39 Pruebas de los servos con el robot en movimiento corrección 1	152
Tabla 40 Pruebas de los servos con el robot en movimiento final .....	152
Tabla 41 Pruebas de potencia y control.....	153
Tabla 42 Resultados de las pruebas de comunicación iniciales .....	154
Tabla 43 Resultados de las pruebas de comunicación corregida. ....	154
Tabla 44 Datos teóricos del espacio de trabajo-límites.....	155
Tabla 45 Mediciones de la pata 1 .....	156
Tabla 46 Errores de cada límite en la pata 1 .....	156
Tabla 47 Mediciones de la pata 2 .....	156
Tabla 48 Errores de cada límite en la pata 2 .....	157
Tabla 49 Mediciones de la pata 3 .....	157
Tabla 50 Errores de cada límite en la pata 3 .....	157
Tabla 51 Mediciones de la pata 4 .....	158
Tabla 52 Errores de cada límite en la pata 4 .....	158
Tabla 53 Resultados de las pruebas de caminata discontinua traslación..	159
Tabla 54 Resultados de las pruebas de caminata de giro antihorario .....	160
Tabla 55 Resultados de caminata discontinua de giro horario .....	160

## RESUMEN

Este proyecto presenta el diseño y construcción de una plataforma zoomórfica con doce grados de libertad distribuidos de manera simétrica en sus cuatro extremidades, capaz de manejarse sobre superficies regulares de manera teleoperada mediante el control remoto desde un *joystick* o una HMI. Para esto inicialmente se realizó un estudio previo de los trabajos elaborados en el país, luego se desarrolló el modelamiento robótico mediante el análisis cinemático y dinámico, permitiendo que el cuadrúpedo desenvuelva la caminata discontinua de dos fases en puntos de apoyo estáticamente estables, garantizando que no existan fallos por sobrecarga en sus actuadores. El mecanismo consta de tres eslabones por pata, diseñados a partir de las proporciones longitudinales en insectos; y su fabricación se realizó con los materiales ABS y acrílico, obteniendo factores de seguridad aceptables para su implementación. El sistema de comunicación inalámbrico se estableció con módulos RF que permitieron enlazar grandes distancias desde los mandos de control al robot. El sistema de control fue desarrollado en el software de código abierto *Robot Operating System* (ROS), el cual es compatible con distintos dispositivos y posee una arquitectura modular que implica una independencia en la ejecución de procesos evitando fallos a nivel de sistema. En base a las pruebas realizadas, el funcionamiento del robot mostró una estabilidad apropiada para los diferentes tipos de movimiento, reflejando un porcentaje de exactitud promedio del 90% en la ejecución de la locomoción y cinemática inversa.

### PALABRAS CLAVE

- **CÓDIGO LIBRE**
- **LOCOMOCIÓN**
- **ROBOT**
- **ROBOT OPERATING SYSTEM**

## **ABSTRACT**

This project presents the design and construction of a zoomorphic platform with twelve degrees of freedom distributed symmetrically in its four legs. It is able to move over flat surfaces being teleoperated by a joystick or HMI. Firstly, a study was carried out about previous researches done in the country. After that, the kinematic and dynamic analysis was done for the robot modelling that allowed the quadruped to accomplish a stable two-phase discontinuous walking and ensuring that the damages in the actuators by overcharging do not exist. The three links per leg which conform the mechanism was designed on the longitudinal correlations of the insects and the implementation was made in ABS and acrylic materials reaching suitable security factors. The wireless communication system was settled down with RF modules which permitted to joint large distances from the control knobs. The control system was developed in the free software known as Robot Operated System (ROS) compatible with many devices and within a modular architecture which means an independence in the execution processes to avoid system fails. Based on the trials done, the functionality of the robot showed a proper stability for the different movements achieving a high accuracy with 90% of reliability in its locomotion and inverse kinematic.

### **KEYWORDS**

- **ROBOT**
- **FREE SOFTWARE**
- **LOCOMOTION**
- **ROBOT OPERATING SYSTEM**

## CAPÍTULO 1. GENERALIDADES

### 1.1 PLANTEAMIENTO DEL PROBLEMA

Los robots móviles han ido desarrollando una amplia gama de aplicaciones y se ha implementado nueva tecnología para adaptarse a las condiciones a las que están expuestos. Es así, que abarcan desde aplicaciones tan complejas como su intervención en procesos productivos, cirugía asistida por robots teledirigidos, robots espaciales, hasta aplicaciones sencillas como robots de compañía y entretenimiento.

Muchos de los robots móviles manejan su locomoción mediante ruedas, debido a que: su construcción y control es rápido en su implementación, tienen menor consumo de energía, son más robustos, soportan mayores cargas, son más rápidos, entre otras. Pero tienen la problemática de no poder ser manejados en terreno irregular.

Los robots diseñados con locomoción por patas, al tener puntos discretos de soporte de su cuerpo, pueden evadir obstáculos en el camino y manejarse en terrenos de difícil acceso. Por lo que, a pesar de su complejo diseño y construcción, presentan mayor versatilidad con respecto a los robots de locomoción por ruedas, debido a la flexibilidad de su movimiento.

Uno de los problemas comunes de la industria es acceder a lugares de difícil acceso, ya sea por razones como:

- Terreno irregular.
- Espacios pequeños de movilización para ejecutar el proceso.
- Zonas de riesgo por emanación de gases tóxicos e inflamables.
- Zonas peligrosas como: lugares de detonación de bombas y campos minados.
- Lugares sin atmósfera (exploraciones en el espacio).

El proyecto consta de la implementación de una plataforma robótica que permita exploraciones en lugares sin obstáculos de manera teleoperada. Dando paso a investigaciones como: inteligencia artificial para su locomoción autónoma, cambio del algoritmo de coordinación del movimiento para que pueda desarrollarse en terrenos hostiles con obstáculos, implementación de sensores de acuerdo a las aplicaciones en las que se desarrolle, etc. Además, se va a implementar una cámara que permita el seguimiento del terreno.

El robot permitirá contribuir de manera activa a futuras investigaciones motivando su construcción y uso en la exploración de lugares de difícil acceso.

## **1.2 ANTECEDENTES**

### **1.2.1 DESARROLLO DE INVESTIGACIONES DE ROBOTS HEXÁPODOS EN LA UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE**

El proyecto “Diseño e Implementación de un Robot Móvil tipo Hexápodo Teledirigido” implementado por Fuertes & Llumiquinga (2005), está compuesto de sensores que permiten una verificación del terreno tanto de obstáculos así como de agujeros en el suelo, además posee una cámara para monitorización y su forma de control es teleoperada. El segundo proyecto nombrado como “Diseño y Construcción de un Robot Hexápodo con Sistema de Articulaciones de Apéndice Flexible Diseñado para Lugares de Difícil Acceso, con Sistema de Control y Monitoreo en Tiempo Real” realizado por Mullo Mullo & Aymacaña Masapanta (2009) desarrolla un robot que tiene un sistema inalámbrico de gran alcance conjuntamente con una cámara para monitorización, pero sin ningún tipo de sensor para detección de obstáculos. Finalmente Andino Alberca & Rodríguez Sánchez (2016) presentan el diseño de varios controladores de inteligencia artificial aplicados a un robot comercial llamado *Phenix* tipo hexápodo, que posee varios sensores para su estabilidad y evasión de obstáculos.

Los proyectos mencionados tienen cierta similitud con el robot ha desarrollarse ya que presentan la misma forma de control teleoperada y la implementación de una cámara de monitorización. Pero así mismo tienen varias diferencias como el número de extremidades que en este caso son cuatro, la implementación de sensores y el tipo de control para el cual, se utilizará un software libre diseñado específicamente para aplicaciones robóticas llamado ROS.

### **1.3 JUSTIFICACIÓN E IMPORTANCIA**

El proyecto es de gran importancia para el desarrollo tecnológico, educativo y de innovación.

En la actualidad la Robótica es una de las ciencias que tienen más interés de desarrollo a nivel mundial. Es así que el Presidente de una potencia mundial, Estados Unidos, mencionó en su discurso en México en el 2013: “Juntos, tengamos presente que cada dólar y cada peso que invirtamos en las investigaciones y el desarrollo genera un rendimiento mucho mayor a nuestras economías, es decir, más empleos y más oportunidades. Así es que forjemos nuevas alianzas en áreas tales como la aeroespacial, TI, nanotecnología, biotecnología y robótica” (Terra, 2013).

El Gobierno Nacional de la República del Ecuador hace un énfasis exhaustivo en el desarrollo de nuevas tecnologías. El objetivo once del Plan Nacional del Buen Vivir cita “Asegurar la soberanía y eficiencia de los sectores estratégicos para la transformación industrial y tecnológica” (PNBV, 2003-2017). El cual va de la mano con la Participación Ciudadana en Ciencia y Tecnología: “Programas de Fortalecimiento y Articulación del Sistema Nacional de Ciencia Tecnología e Innovación” (PNBV, 2003-2017). El proyecto pretende alinearse a este plan ya que a futuro permitirá desarrollar una herramienta con infinidad de aplicaciones para beneficio de la sociedad. Siendo una de estas, la monitorización en lugares peligrosos y de difícil acceso de forma rápida y fácil sin comprometer la vida humana.

Además el proyecto aportará significativamente a nivel educativo y de investigación a toda la comunidad universitaria en el desarrollo de robots que cubran una amplia gama de aplicaciones y que se desenvuelvan en condiciones extremas. También permitirá generar trabajos futuros como: inteligencia artificial para su locomoción, un control de extremidades que permita al robot evadir obstáculos, etc.

Una característica importante del robot, será su implementación en la plataforma de programación de código abierto llamada *Robot Operating System* (ROS). El sistema mencionado permite facilitar la comunicación entre distintos dispositivos, manejar controladores y poseer una arquitectura basada en nodos. Dicha herramienta es muy útil para el desarrollo de robots y marca una innovación sobre los proyectos realizados anteriormente en este campo.

#### **1.4 ÁREA DE INFLUENCIA**

El proyecto desarrollado tendrá como fin la obtención de titulación de la carrera de Ingeniería Mecatrónica y comprenderá las siguientes líneas de investigación para aplicaciones de automatización y robótica:

- Automática y Control: en esta línea se abordará un control de movimiento, trabajando con controladores lineales para coordinar cada grado de libertad y tener movimientos conjuntos de acuerdo a los requerimientos.
- Sistemas electrónicos: en esta línea se trabajará la generación de sistemas que permitan la movilidad coordinada, y así mismo la transmisión de datos vía remota para el control de la plataforma móvil.

El área de influencia también se verá reflejada en la comunidad de estudiantes de la Universidad de la Fuerzas Armadas, ya que permitirá futuras investigaciones, como también la exploración de un nuevo medio de programación como lo es el sistema *Robotic Operating System*, descubriendo la versatilidad del mismo para la codificación y control de sistemas robóticos.

## **1.5 OBJETIVOS**

### **1.5.1 OBJETIVO GENERAL**

Diseñar e implementar un sistema robótico teleoperado de cuatro extremidades con doce grados de libertad.

### **1.5.2 OBJETIVOS ESPECÍFICOS**

Diseñar un mecanismo de tres grados de libertad para cada extremidad que permita la mejor locomoción y coordinación del tetrápodo.

Diseñar y construir el cuerpo del robot de manera que aloje todos los componentes eléctricos y electrónicos en su estructura, soporte su peso y mantenga una inercia distribuida para el equilibrio del robot.

Diseñar e implementar un sistema de potencia de manera que transmita la energía necesaria para alimentar los doce servomotores que permitirán la movilización del robot.

Diseñar un sistema de control que admita tener un tiempo de respuesta rápido, para conseguir un óptimo desempeño entre el módulo de comunicación y la placa de control.

Diseñar la placa electrónica del robot, de manera que las fases de: alimentación, control y comunicación, estén bien distribuidas y tengan los mínimos componentes necesarios para disminuir su peso y efecto en la inercia.

Implementar un sistema de comunicación inalámbrico mediante un mando de control que sea intuitivo para el usuario.

Desarrollar un HMI acorde con las funcionalidades que presentará el robot, mostrando las configuraciones disponibles para su control remoto mediante una computadora o un *joystick*.

Verificar la funcionalidad del prototipo mediante pruebas y proseguir con el refinamiento de la estructura mecánica o electrónica de ser necesario.

## **1.6 ALCANCE**

Mediante el desarrollo de este proyecto se pretende aportar a las investigaciones del Laboratorio de Instrumentación de la Universidad de las Fuerzas Armadas ESPE. Permitiendo generar trabajos relevantes que sean causa de más exploraciones en el campo de la robótica aplicada con el desarrollo de software libre. Posibilitando a su vez el crecimiento de la investigación realizada, ya sea modificando el programa para crear algoritmos más complejos, desarrollando más capacidades para el robot zoomórfico, etc.

El proyecto presentará un robot zoomórfico teleoperado con doce grados de libertad, tres por cada extremidad, que permitan su movimiento mediante servomotores. Constará de un módulo de comunicación que transmita de manera serial wireless los movimientos que se deseen, al microcontrolador que se encuentra en el robot. El receptor será acoplado a la placa de control mientras que el emisor se va a conectar a la computadora, la cual va a contar con un “*joystick*” o una interfaz como mandos de control.

La investigación tendrá una nueva perspectiva sobre otros proyectos realizados tanto local como nacional, ya que el software de programación de carácter libre es muy completo, de ahí su nombre de *Robot Operating System* ya que posee todas las particularidades de un sistema operativo. Dicha herramienta permite que la programación se construya en base a una plataforma robusta prometiendo un buen desempeño al momento de operar el robot.

## 1.7 ESTRUCTURA DEL DOCUMENTO

El documento correspondiente al desarrollo del proyecto de titulación constará de seis partes, organizadas en los siguientes capítulos:

Primero, se desarrollarán las generalidades del proyecto que consisten en las características principales que permitieron dar su origen, como: objetivos, planteamiento del problema y los antecedentes que permiten realizar una búsqueda superficial de proyectos similares desarrollados anteriormente nacional e internacionalmente para poder familiarizarse con el tema del proyecto planteado.

En el capítulo dos se realizará un análisis extenso del estado del arte pertinente al proyecto, de tal forma que se investigue, clasifique y analice la información y avances perpetrados en temas referentes al desarrollo de robots zoomórficos con apéndices móviles.

El capítulo tres se enfocará en el diseño mecatrónico del robot, abarcando todo los sistemas: mecánicos, electrónicos, de control y de comunicación que van a conformarlo para su funcionamiento.

En el capítulo cuatro se tiene la implementación del diseño completo elaborado en el capítulo tres. Consiguientemente, se muestran las pruebas y resultados de la implementación total del robot.

En la parte final del documento se realizarán conclusiones y recomendaciones, que permitirán tener una mejor visión de las características finales del proyecto y las particularidades que se presentaron a lo largo del mismo.

## **CAPÍTULO 2. ESTADO DEL ARTE**

### **2.1 INTRODUCCIÓN**

En este capítulo se desarrollará una breve investigación bibliográfica científica sobre robots caminantes. Para ello se aplicará la hermenéutica y la heurística como metodologías para el desarrollo del estado del arte. Lo cual permitirá analizar las ventajas, desventajas, limitaciones y avances de los trabajos relevantes realizados hasta la actualidad.

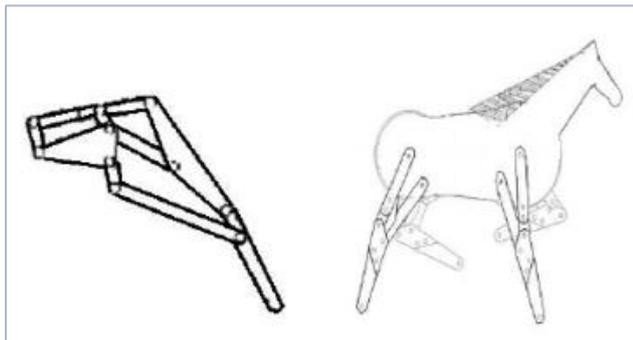
Aportando de esta manera con diversos enfoques que amplíen el horizonte de conocimiento sobre robots cuadrúpedos, como resultado del análisis bibliográfico. Contribuyendo de esta forma al desarrollo del presente proyecto.

### **2.2 ROBOTS MÓVILES**

#### **2.2.1 PERSPECTIVA HISTÓRICA DE ROBOTS CAMINANTES**

La primera vez en la historia en hablarse de robots caminantes es en el siglo III c. AD en la provincia Sichuan de China. El mecanismo se denominó *Mu Niu Lu Ma*, que significa: “*un dispositivo tan potente como un caballo y tan rápido como una vaca*”; fue creado para aplicaciones militares, su objetivo era trasladar provisiones con cargas de hasta 250 kilogramos. El caballo de madera era capaz de recorrer una distancia de diez kilómetros por día y fue construido bajo supervisión de Zhu Ge-Liang durante la guerra contra el reino de Wei, ubicado en el centro de China (Ceccarelli, 2004).

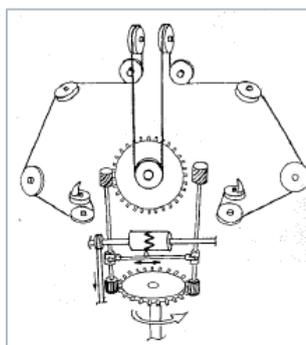
A pesar de que no existen datos exactos acerca de los detalles del diseño de *Mu Niu Lu Ma*, varios investigadores han realizado diferentes diseños sobre la reconstrucción del mismo. La más conocida es la reconstrucción hecha por Wang Jian, la cual se basa en un mecanismo complejo de diez barras (Ceccarelli, 2004). Ver Figura 1.



**Figura 1. Mecanismo de diez barras de la extremidad**

**Fuente: (Ceccarelli, 2004)**

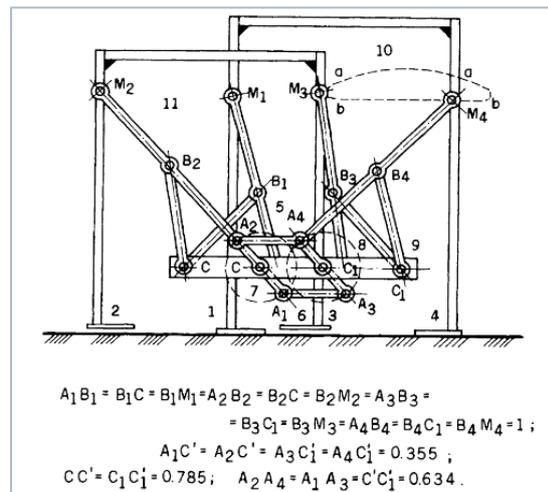
A pesar de que *Mu Niu Lu Ma* tuvo un gran impacto en la época que se desarrolló, no se efectuaron más mecanismos caminantes hasta la época del Renacimiento; en la cual hubo un gran crecimiento del arte y la ciencia. Entre los años de 1495 y 1497 Leonardo Da Vinci creó una armadura de caballero que tenía movilidad, considerada como el primer robot antropomórfico de la civilización oriental (Rosheim, 1997). El diseño de la armadura se lo puede observar en la Figura 2.



**Figura 2. Mecanismo de Armadura de Caballero**

**Fuente: (Rosheim, 1997)**

En el año de 1850, se desarrolló un modelo mecánico de un sistema de locomoción antropomórfico, el cual, mediante transmisiones mecánicas movía al cuerpo en una trayectoria horizontal, mientras que los pies se movían de arriba a abajo para efectuar pasos. El modelo lo realizó el ruso matemático Chebyshev y se basó en un mecanismo de cuatro barras (Raibert M. H., 1986).



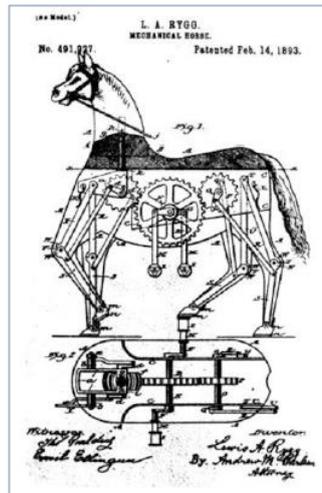
**Figura 3. Plataforma Caminante de Chebyshev**

**Fuente: (Sandler, 1999)**

La plataforma caminante de Chebyshev, mostrada en la Figura 3 era incapaz de realizar giros o desplazarse por terreno irregular. Sin embargo significó un gran avance en el desarrollo de robots caminantes.

Alrededor del año 1878 se realizaron investigaciones sobre el movimiento de las extremidades en la locomoción de caballos. Posteriormente se efectuaron investigaciones en más mamíferos y finalmente gracias a los resultados de las mismas (Irawan, Mohd Razali, & Nonami, 2014).

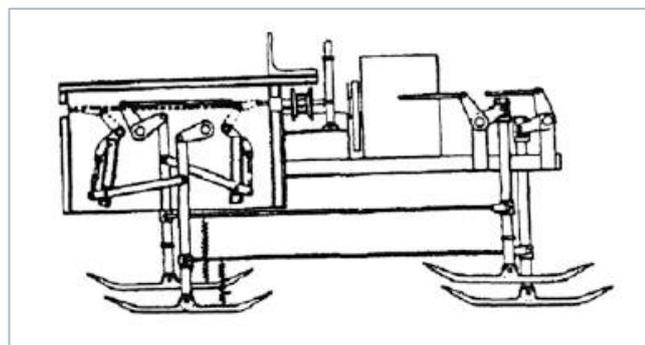
En 1893 L.A. Rygg crea la primera máquina cuadrúpeda denominada *The Mechanical Horse*. El mecanismo estaba diseñado con dos pedales a sus extremos laterales, que mediante una serie de engranajes transmitían el movimiento a los eslabones de las extremidades y permitía la caminata del caballo como se puede ver en la Figura 4, (Irawan, Mohd Razali, & Nonami, 2014).



**Figura 4. *The Mechanical Horse***

**Fuente: Irawan et al. (2014)**

El Barón Bechtolsheim en 1913 patenta una máquina caminante cuadrúpeda, el funcionamiento del mecanismo no está bien especificado pero se muestra el esquema de su estructura en la Figura 5 (Irawan et al., 2014).



**Figura 5 Cuadrúpedo de Bechtolsheim**

**Fuente: Irawan et al. (2014)**

A pesar de que los creadores de los mecanismos observados, desarrollaron las partes estructurales y funcionales de sus proyectos de forma empírica, el conocimiento generado por parte de sus investigaciones, fue la base del desarrollo de las investigaciones realizadas en robot caminantes futuras, a partir de que en 1921 el checoslovaco Karel Capek, mencionó por primera vez el término “Robot” y que en 1942 Isaac Asimov introdujo las Tres Leyes de la Robótica (Irawan et al.,2014).

Por otra parte, la Segunda Guerra Mundial (1939-1945) fue un detonante de desarrollo en varios campos de la ciencia. Los avances realizados en materiales, sistemas de control, programación, electrónica y tecnología, entre otros (Irawan et al.,2014). Éstos permitieron alcanzar un mejor desempeño en las investigaciones, proyectos e innovaciones realizados sobre robots caminantes, por lo que fue una época rica en cuánto a avances tecnológicos y científicos en este campo; dominando formas de caminata e incluso permitiendo el desarrollo de robots autónomos para tareas peligrosas o repetitivas que involucran un deterioro de salud en el ser humano.

## 2.2.2 DESARROLLO TECNOLÓGICO DE ROBOTS CAMINANTES

A partir del año 1950 el desarrollo y evolución de los robots caminantes fue considerable, por lo que se va a mostrar una visión resumida de los proyectos de mayor impacto y sus principales características en la Tabla 1.

Tabla 1

### Avance Tecnológico Cronológico de Robots Caminantes y sus Características

Proyecto	Autor (Año)	Características
<b>The Iron Mule Train (Morrison, 1968).</b>	<i>Space General Corporation</i> (1960)	Aplicaciones de vehículo lunar. Locomoción: mediante ocho extremidades. Movimiento mediante levas y transmisión por enlaces. <u>Limitaciones:</u> Incapaz de recorrer terrenos irregulares. <u>Observaciones:</u> Necesidad de más grados de libertad para su movimiento.
<b>General Electric Quadruped Irawan et al. (2014).</b>	Ralph Mosher (1960-1968)	Locomoción: Tres grados de libertad por pata. Actuadores: cilindros hidráulicos lineales. Dimensiones: 3.3m de alto, 3m de largo. Peso: 1400 kg. Funcionamiento: Motor de combustión interna Operado mediante palancas combinadas con pedales que controlaban cada extremidad. <u>Limitaciones:</u> Su control implicaba gran capacidad del operario. <u>Observaciones:</u> Primero en caminar en terreno irregular.
<b>Big Muskie, (Caterpillar, 2016).</b>	<i>Bucyrus Eire Company</i> (1966-1969)	Aplicación: excavadora de minas. Locomoción: 4 patas accionadas hidráulicamente. Propulsión hacia adelante con todas las patas en el piso. Peso: 19500 ton. <u>Limitaciones:</u> Coste energético alto: 13800 voltios, traducido a decenas de miles de dólares por hora. <u>Observaciones:</u> Cuadrúpedo más grande conocido. Removió un total de 465000000 m <sup>3</sup> durante su funcionamiento.

Continúa 

<b>Phony Pony (McGhee R. B., Robot locomotion, 1976).</b>	McGhee (1969)	<p>Locomoción: Dos grados de libertad por extremidad.          Actuadores: motores eléctricos.          Funcionamiento: Poseía sensores para determinar tres estados: bloqueo, adelante, atrás.          Primero en implementar un control mediante electrónica lógica basada en <i>flip-flops</i>.  <u>Limitaciones:</u> Energizado externamente. Incapaz de girar.  <u>Observaciones:</u> Estructura de la pata en forma de <i>T</i> invertida, le brindaba mayor estabilidad.</p>
<b>Hexápodo OSU, (McGhee R. B., 1977).</b>	Universidad Estatal de Ohio (1977)	<p>Locomoción: Tres grados de libertad por extremidad.          Puede caminar sobre superficies inclinadas, subir escaleras, sobrepasar obstáculos y girar.          Actuadores: motores de taladro eléctrico.          Control mediante un multi-procesador.          Dimensiones: 1.3m longitud, 1.4m ancho. Peso: 100kg.  <u>Limitaciones:</u> Velocidad pequeña (pulg/s).  <u>Observaciones:</u> Contaba con un giroscopio para su mejor desempeño.</p>
<b>Cuadrúpedo PV-II, (Hirose, y otros, 2009).</b>	Instituto Tecnológico de Tokio (1980)	<p>Tres grados de libertad por extremidad.          Longitud: 0.9m; peso: 10kg.          El segundo en aplicar el concepto GDA (<i>Gravitationally Decoupled Actuation</i>) para desarrollar un mecanismo basándose en la eficiencia de la energía del robot.          Posee las mismas capacidades de OSU.          Sensores de contacto para la detección de obstáculos.  <u>Limitaciones:</u> Alimentación externa. La comunicación con el control era cableada desde una computadora.  <u>Observaciones:</u> Usa un mecanismo de pantógrafo para disminuir el peso del robot y aumentar el espacio de trabajo de la pata.</p>
<b>TITAN III (Hirose, y otros, 2009).</b>	Instituto Tecnológico de Tokio (1984)	<p>Mejora del mecanismo de pantógrafo de las patas incrementando la movilidad del robot.          Equipado de sensores de postura y sensores de contacto en las extremidades.          Tenía un súper-sistema de control inteligente encargado de efectuar decisiones de acuerdo a la lectura de los sensores.  <u>Limitaciones:</u> Las mismas de PV-II.  <u>Observaciones:</u> Fue el comienzo de una serie de robots, terminando en TITAN IX en el 2001.</p>
<b>Robot Cuadrúpedo de Raibert (Raibert &amp; Chepponis, 1986)</b>	Instituto de Tecnología de MA. (1986)	<p>Locomoción: Dos grados de libertad por extremidad.          Implementó el principio "un-pie" según el cual solo un pie permanece en el suelo en un determinado tiempo.          Primero en desarrollar la modalidad de trote, moviendo un par de patas sincronizadamente.          Utilizó actuadores hidráulicos lineales.  <u>Limitaciones:</u> El proyecto descartó el análisis de estabilidad pasiva de las patas. Limitaciones de PV-II.  <u>Observaciones:</u> Primero en aplicar estabilidad dinámica en el análisis del movimiento.</p>

A partir de 1986, varios robots se diseñaron para distintas aplicaciones, permitiendo la innovación y mejoramiento de los robots detallados en la Tabla 1. Uno de ellos fue el robot humanoide HONDA, creado para cooperar en las actividades cotidianas de los seres humanos, además podía subir y bajar escaleras (Irawan et al., 2014).

Otro robot que aportó significativamente fue DANTE, creado originalmente en 1993 y con un diseño mejorado en 1994. En Irawan et al. (2014), se menciona que fue el primer robot diseñado para trabajar en un ambiente hostil, su funcionamiento se probó en un cráter en Alaska; la alimentación del robot era externa mediante un cable, se diseñó de dicha manera como medio de rescate, en caso de que el robot perdiera estabilidad y pueda caer hacia el interior del cráter, como sucedió con su primer diseño.

Las aplicaciones de los robots realizados después, se basaron en el desarrollo y perfeccionamiento de humanoides y cuadrúpedos. Los cuadrúpedos creados se enfocaron en la marcha de galope fundamentados en el estudio realizado por Raibert & Chepponis (1986).

El análisis de las características de los robots observadas en la Tabla 1 aparte de mostrar el desarrollo de los robots en la historia, permiten observar los puntos relevantes en el diseño de robots caminantes, como también, los puntos en que se debe focalizar la atención al momento de desarrollar el proyecto.

Los factores de diseño importantes que se rescatan para el mejor desempeño de los robots, de acuerdo a la experiencia obtenida de los proyectos mencionados en la Tabla 1 son:

- A mayor grados de libertad, mayor flexibilidad de movimiento.
- Selección correcta de actuadores en los enlaces del robot.
- Es de suma importancia incluir en el diseño inicial del robot, análisis de estabilidad estática y análisis de estabilidad dinámica para el mejor desempeño del robot.

- Realizar el número de pruebas necesarias sobre el funcionamiento del robot, para asegurar el desempeño esperado en campo.
- Es necesario la implementación de sensores que permitan tener una visión medible del ambiente en el que trabaja el robot, para desarrollar aplicaciones de movimiento sobre terreno irregular y superación de obstáculos.

### **2.3 ANÁLISIS DE PROYECTOS REALIZADOS EN EL PAÍS**

El desarrollo de robots cuadrúpedos en el país es muy escaso, debido a varios factores, uno de ellos es la necesidad de un análisis más profundo de la estabilidad en el movimiento, ya que en la transición de una extremidad solo tres patas van a permanecer en contacto con el terreno en que se desenvuelve el robot y entre otros constan el costo, facilidad de manufactura, etc.

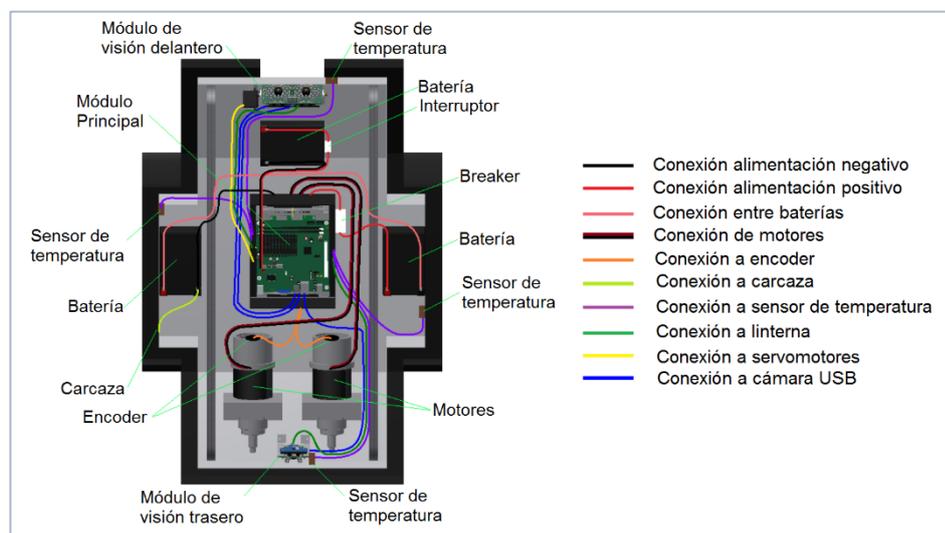
Los proyectos implementados con mayor frecuencia son robots hexápodos y rodantes de los cuáles se realizará un análisis general.

En la tesis Argudo Cobos & Arpi Saldaña (2012), desarrollada en la Universidad Politécnica Salesiana de Cuenca, se implementó un robot móvil teleoperado para aplicaciones de asistencia en operaciones de alto riesgo para el Cuerpo de Bomberos de Cuenca. El mecanismo se basa un robot móvil con locomoción por ruedas, cuenta con varios sensores de: temperatura, de aceleración, de razón de giro, de humedad y encoders. El sistema de control de los motores se desarrolló mediante sintonización de un PID implementado durante las pruebas de los mismos. También cuenta con un módulo de comunicación inalámbrico y una cámara, que son necesarios para el control remoto del robot.

Un esquema completo de la estructura del robot es mostrado en la Figura 7 y el diseño final del robot en la Figura 6.



**Figura 6. Robot móvil terminado**  
**Fuente: (Argudo Cobos & Arpi Saldaña, 2012)**



**Figura 7. Esquema de conexiones del robot móvil**  
**Fuente: (Argudo Cobos & Arpi Saldaña, 2012)**

Los son: estabilidad sobre terreno irregular y la capacidad de soportar temperaturas requerimientos de trabajo del robot, basados en encuestas realizadas al Cuerpo de Bomberos altas de trabajo. Éste superó las pruebas, sin embargo el aislamiento térmico que tenía mediante la carcasa externa no funcionó de la manera esperada y se filtró el calor al medio interno del mismo, por lo que expuesto continuamente a grandes temperaturas el sistema electrónico del robot puede llegar a fallar (Argudo Cobos & Arpi Saldaña, 2012). Otra desventaja que presentó el proyecto es que debido al tipo de locomoción escogida, puede manejarse solo sobre terrenos medianamente irregulares como se muestra en la Figura 8, por lo que su desplazamiento sobre el tipo de terrenos solicitado se vio limitado.



**Figura 8. Pruebas sobre terreno irregular de robot móvil**

**Fuente: (Argudo Cobos & Arpi Saldaña, 2012)**

Este proyecto permite razonar sobre las aplicaciones sociales que se puede dar a los robots móviles; pero también reconocer las posibilidades de desarrollar robots caminantes en lugar de robots móviles con ruedas. Siendo una de las principales, el poder desplazarse sobre terrenos irregulares sin ningún inconveniente. También, se ve la necesidad de realizar una innovación sobre el prototipo, ya que el país tiene un mayor índice de incendios forestales últimamente, por lo que el robot a desarrollarse debería ser apto de moverse sobre terrenos con irregularidades grandes, como son las áreas de bosques, parques forestales, parques ecológicos, etc. Además es necesario realizar estudio de materiales que permitan que el robot resista a altas temperaturas sin posibilidad a fallos.

En la Universidad de las Fuerzas armadas ESPE se llevó a cabo otro proyecto de tesis sobre robots teledirigidos denominado: “Diseño e implementación de un Robot Móvil tipo Hexápodo Teledirigido” realizado por Fuertes & Llumiquinga (2005), el robot posee dos grados de libertad por extremidad y es controlado mediante el microcontrolador PIC, que forma parte del cerebro del robot.

Presenta un sensor ultrasónico en la parte superior, como se observa en la Figura 9 de tal forma que le permita detectar obstáculos grandes que impidan su correcto desempeño. El algoritmo utilizado por Fuertes & Llumiquinga (2005) es el siguiente:

- El robot recibe la orden de movimiento.
- Si no existe obstáculos entonces avanza un paso.
- Verifica si todavía permanece la orden de avanzar.
- Si no existe obstáculos y avanza otro paso.
- En caso de existir obstáculos, el robot hace caso omiso de la orden avanzar y permanece en su posición inicial.

Debido al uso del sensor ultrasónico, pequeñas irregularidades sobre terreno en el que se desplace va a dar como resultado que permanezca en su posición inicial sin efectuar movimiento.



**Figura 9. Caminata sobre superficie lisa - Robot hexápodo**

**Fuente: (Fuertes & Llumiquinga, 2005)**

Los comandos se envían al microcontrolador mediante un potenciómetro, el cual posee límites predefinidos para: avanzar hacia adelante, hacia atrás, giro hacia la izquierda y giro hacia la derecha (Fuertes & Llumiquinga, 2005).

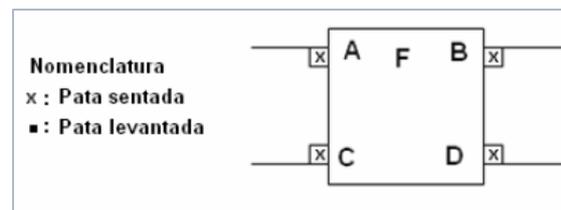
Una de las características importantes de este proyecto es que posee gran estabilidad en superficies lisas, debido a que el método de caminata escogida es la de paso por ondulación, la cual permite al robot tener cuatro patas en contacto con el piso en la fase de transferencia al momento de efectuar un paso (Fuertes & Llumiquinga, 2005). Por lo tanto, se puede comprobar mediante este prototipo, que los hexápodos brindan una mayor estabilidad, tanto estática como dinámica, que los cuadrúpedos.

De este proyecto se concluye que el tener robots móviles con extremidades o caminantes, brinda al usuario una amplia gama de

aplicaciones y ventajas sobre los rodantes. Por otra parte Fuertes & Llumiquinga (2005) concluyen que robot es incapaz de trabajar sobre superficies medianamente irregulares, como terrenos con césped, ya que la onda emitida por el sensor ultrasónico sería absorbida y no se reflejaría de manera correcta.

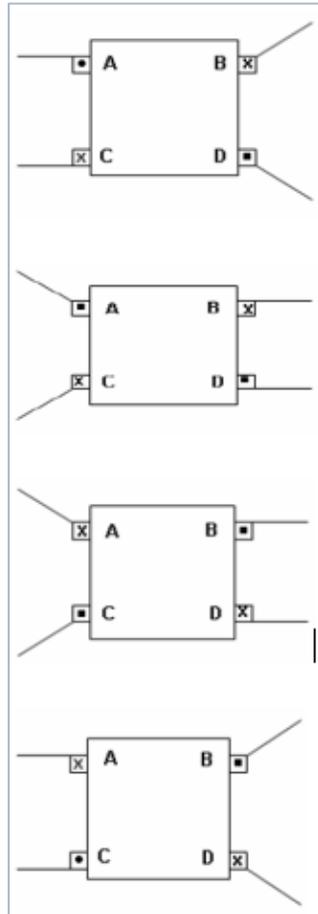
En la Universidad Politécnica Nacional se desarrolló un proyecto titulado “Diseño y construcción de un robot de vigilancia” de Pinta Arrobo (2007), con un diseño parecido al de Fuertes & Llumiquinga (2005), presenta un robot cuadrúpedo con dos grados de libertad por extremidad. Cuenta al igual que el proyecto anterior con sensores ultrasónicos y además tiene un sensor térmico para medir la temperatura de los objetos que lo rodean. El microcontrolador usado para este caso también es un PIC16F86A.

Para la locomoción este robot se escogió el tipo de caminata continua, con dos extremidades desplazándose a la vez. Tomando en cuenta la nomenclatura y configuración inicial indicada en la Figura 10, la secuencia de locomoción del robot es mostrada en la Figura 11.



**Figura 10. Configuración inicial del cuadrúpedo**

**Fuente: (Pinta Arrobo, 2007)**



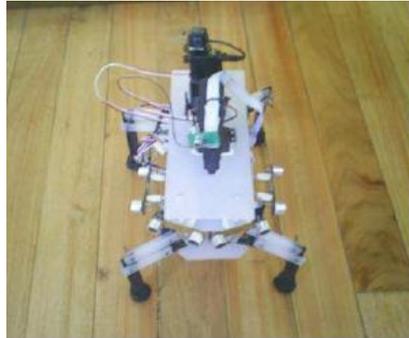
**Figura 11. Secuencia de locomoción del cuadrúpedo**

**Fuente: (Pinta Arrobo, 2007)**

El robot tiene dos modos de control, uno manual y uno automático. El control automático lo hace mediante la lectura de los sensores ultrasónicos; mientras que en el modo manual se rige mediante los comandos enviados desde un ordenador, permitiéndole desplazarse en los sentidos: adelante, atrás, izquierda y derecha (Pinta Arrobo, 2007).

El proyecto también tiene dos modalidades de comportamiento, una normal y una de combate. En la normal se desplaza evitando obstáculos y sin efectuar ninguna acción en caso de encontrar un ser humano. En el modo de combate, si encuentra una persona en su trayectoria, lanza pintura sobre la misma mediante una pistola accionada por un servomotor (Pinta Arrobo, 2007).

Como se puede observar en la Figura 12, el robot es capaz de desplazarse sin ningún inconveniente sobre superficies lisas. También posee una cámara que funciona inalámbricamente para el envío de datos.



**Figura 12. Robot cuadrúpedo en funcionamiento**

**Fuente: (Pinta Arrobo, 2007)**

Este proyecto realizó varias innovaciones sobre el diseño realizado en la Universidad de las Fuerzas Armadas ESPE por Fuertes & Llumiquinga (2005) entre ellas:

- El uso de más sensores de ultrasonido para la detección de obstáculos.
- Darle una aplicación al robot con las dos modalidades de comportamiento.
- Es un modelo menos costoso debido a la disminución de extremidades y por ende disminución de actuadores.

No se realizó un modelo cinemático ni dinámico del robot debido a la complejidad de su elaboración, las posiciones en el espacio de trabajo se encontraron empíricamente, por lo que tampoco se toma en cuenta análisis de estabilidad para su funcionamiento.

Como una observación final, Fuertes & Llumiquinga (2005) recomiendan la implementación de por lo menos tres grados de libertad por extremidad para futuros proyectos, con la finalidad de tener un mejor movimiento de las mismas.

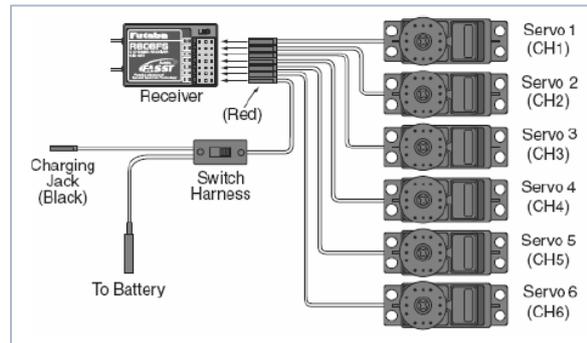
En la Universidad de las Fuerzas Armadas ESPE, en la sede de Latacunga, se realizó un proyecto de un robot móvil con apéndices elaborado por Mullo Mullo & Aymacaña Masapanta (2009). La estructura del robot hexápodo fue elaborada mediante el corte de planchas de aluminio de dos milímetros de espesor. El diseño de la pata consta de tres grados de libertad (ver Figura 13) y sus actuadores son servomotores de 3 kg.cm de torque.



**Figura 13. Diseño de la pata del robot hexápodo**  
**Fuente: (Mullo Mullo & Aymacaña Masapanta, 2009)**

Para el circuito de control se utilizó el microcontrolador dsPIC30F4011, cumpliendo con todas las entradas y salidas necesarias para el desarrollo del proyecto. Para la elaboración de la placa se implementó, además de los circuitos necesarios para su control, capacitores para reducir el ruido de los servomotores e inhibir que el microcontrolador se resetee (Mullo Mullo & Aymacaña Masapanta, 2009).

Este modelo cuenta con un sistema de transmisión inalámbrica de datos más desarrollado que en los proyectos anteriores. Se utilizaron transmisores y receptores FUTABA para la comunicación, los cuales codifican la señal enviada y la convierte en pulsos para el funcionamiento de los servomotores, por lo que se los conecta directamente al receptor (Mullo Mullo & Aymacaña Masapanta, 2009). Ver Figura 14.



**Figura 14. Conexiones al receptor R6006FS Futaba**  
**Fuente: (Mullo Mullo & Aymacaña Masapanta, 2009)**

El hexápodo consta de un sistema de control remoto, mediante el transmisor Futaba ya mencionado, además tiene una cámara inalámbrica que permite tener una visión del ambiente en el que se desempeña, acoplada a un servomotor para girar en la dirección deseada de acuerdo al movimiento (Mullo Mullo & Aymacaña Masapanta, 2009). El ensamblaje final se indica en la Figura 15.



**Figura 15. Robot hexápodo - Universidad de las Fuerzas Armadas**  
**Fuente: (Mullo Mullo & Aymacaña Masapanta, 2009)**

Este proyecto representa un gran avance en el desarrollo de robots móviles en el país, debido a su sistema eficaz de transmisión de datos y su estructura liviana que le permite movilizarse sin dificultad alguna con servomotores de torque pequeño, lo cual posibilita la optimización de recursos.

A pesar de que el robot presenta varias mejoras sobre modelos anteriores, también posee las siguientes desventajas:

- No cuenta con análisis cinemático ni dinámico para su movimiento.
- No se desarrolló análisis de estabilidad.
- La batería del robot es una batería de Niquel Cadmio por lo cual su duración es pequeña.
- Puede desplazarse solo sobre terreno plano y rugoso.
- Debido a que no se desarrolló un análisis de estabilidad, el robot solo puede manejarse en pequeñas velocidades para no perder el equilibrio.

La investigación desarrollada en la Universidad Estatal Península de Santa Elena por Naranjo Suárez (2013), es un robot móvil con ruedas planeado para el monitoreo y seguridad de un laboratorio de la misma universidad. Tiene dos modos de control: autónomo y teleoperado. Los actuadores de las ruedas son cuatro motores de corriente continua controlados mediante la placa Arduino.

El ensamblaje final es el mostrado en la Figura 16, controlado inalámbricamente mediante los módulos de comunicación XBee, de fácil configuración y comunicación con el ordenador (Naranjo Suárez, 2013).



**Figura 16. Robot móvil Universidad Estatal Península de Santa Elena**

**Fuente: (Naranjo Suárez, 2013)**

El mecanismo es incapaz de movilizarse sobre terreno irregular debido a su diseño como robot móvil con ruedas, además las baterías deben ser recargadas después de cada sesión de navegación. Por otra parte, mostró un buen desempeño al inhibir obstáculos mediante el sensor PIR ubicado en su parte superior (Naranjo Suárez, 2013).

Se puede observar que los diseños móviles con ruedas tienen menor dificultad que los diseños con extremidades en el desarrollo del modelo cinemático de los mismos, sin embargo los robots móviles con apéndices muestran una gran variedad de ventajas sobre los rodantes.

A pesar que el siguiente trabajo no fue realizado en el Ecuador, consta en la base de datos del SENECYT, debido a que se efectuó mediante una beca financiada por dicha entidad. El proyecto se desarrolló en Australia en la Universidad de Queensland por Erazo Sosa (2014), denominado "*Terrain Adaptive Gaits for Hexapod*".

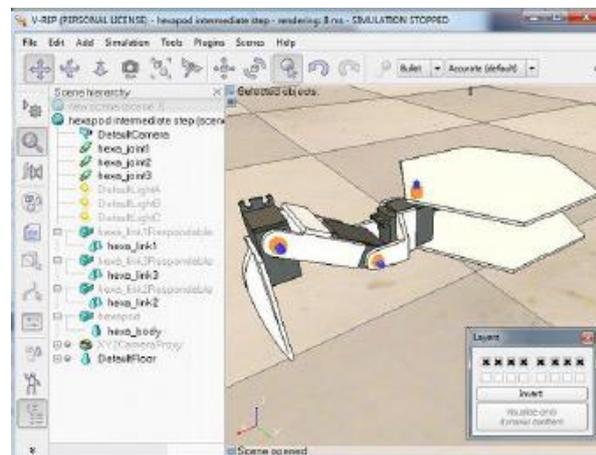
Este prototipo presenta mayor similitud a la investigación desarrollada en el presente trabajo, de acuerdo a las características de software, interfaz gráfica y desarrollo cinemático.

Se realizó un análisis cinemático para el desarrollo del movimiento, examinando la cinemática directa e inversa de cada extremidad de acuerdo a las características físicas de su diseño. Por otra parte, también se realizó un análisis de estabilidad para su movimiento, eligiendo el tipo de caminata que le brinde mayor firmeza al mecanismo. El hexápodo tiene tres grados de libertad por extremidad, es capaz de caminar en terrenos con obstáculos y cambiar su comportamiento para moverse sobre superficies inclinadas (Erazo Sosa, 2014).

La innovación de este estudio es que adquiere información sobre el medio en el que se desarrolla el robot, mediante la obtención de datos de los servomotores. La interpolación y combinación de pequeñas variaciones en los mismos, hacen que se pueda interpretar su lectura.

Se desarrolló sobre el sistema operativo Linux y la implementación del programa se lo hizo en *Robot Operating System* (ROS), por su gran versatilidad para la creación de este tipo de proyectos; la interfaz visual se la desarrolló en *Virtual Experimentation Platform* (V-rep) que es un simulador del ambiente en el cual se desenvuelve, teniendo compatibilidad de comunicación con otras plataformas como lo es ROS y por consiguiente teniendo la capacidad de recibir información de fuentes externas para la ejecución de los algoritmos programados para su movilidad (Erazo Sosa, 2014).

A pesar de que los algoritmos desarrollados para el funcionamiento del robot, cumplieron eficientemente con sus objetivos, la desventaja de este proyecto, es que no fue implementado, sin embargo se realizaron simulaciones en el entorno virtual de V-rep ya mencionado. Por otra parte, no se elaboró un diseño mecánico del robot ya que el prototipo fue obtenido de los modelos precargados en V-rep y modificado por el autor del proyecto. En la Figura 17 se puede observar la configuración de una extremidad del robot en la plataforma virtual.

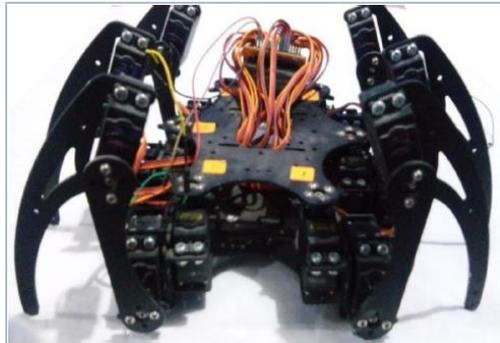


**Figura 17. Ensamblaje del robot hexápodo de la Universidad de Queensland**

**Fuente: (Erazo Sosa, 2014)**

En la Universidad Estatal Península de Santa Elena, se desarrolló otro robot hexápodo que posee aplicaciones de limpieza, realizado por Hidalgo Panchana (2015). El robot semiautónomo, cuenta con tres grados de libertad por pata, utilizó servomotores como actuadores para su movilidad, además de servos adicionales para la pinza encargada de la recolección de basura ubicada en la parte inferior de su cabeza.

La estructura fue comprada y se realizó el montaje de sus piezas. El ensamblaje final del robot comercial, se muestra en la Figura 18.



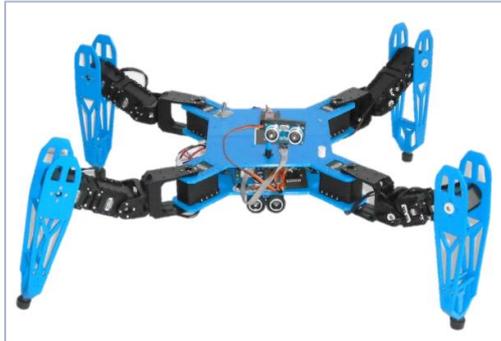
**Figura 18. Robot hexápodo de la universidad de Santa Elena**

**Fuente: (Hidalgo Panchana, 2015)**

A pesar de mencionarse el análisis cinemático del robot en el marco teórico, no se lo desarrolla posteriormente. Además, ya que el robot fue diseñado para movilización en el interior de un laboratorio, está planteado para caminar sobre superficies lisas.

La interfaz gráfica está sobre el software de Arduino, mediante el despliegue de los datos en el monitor serial. Por otra parte, el robot mantiene una modalidad de control teleoperado comunicado mediante el cable de transmisión serial de datos de la placa de control (Hidalgo Panchana, 2015).

Machuma Mendoza & López Ayala, (2015) en “Locomoción de robot cuadrúpedo basada en redes neuronales artificiales”, desarrollaron un cuadrúpedo con tres grados de libertad por extremidad y locomoción mediante una Red Neuronal Artificial Perceptrón Multicapa. Su estructura final se puede observar en la Figura 19.



**Figura 19. Estructura del cuadrúpedo Universidad de Chimborazo**

**Fuente: (Machuma Mendoza & López Ayala, 2015)**

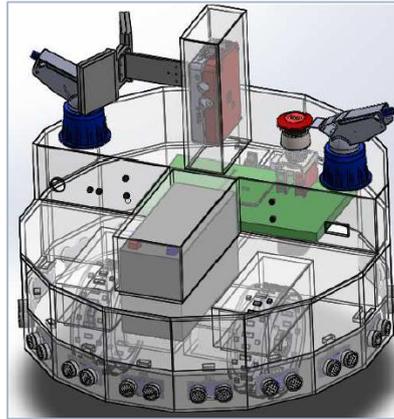
El robot posee módulos XBee para su comunicación, también consta de un acelerómetro para determinar la estabilidad del robot y un sensor ultrasónico para detectar obstáculos. La interfaz de monitoreo fue realizada en el software de Matlab; el robot se desplaza de forma autónoma y está constantemente en funcionamiento (Hidalgo Panchana, 2015).

Sus principales ventajas son: una mayor estabilidad de acuerdo a los resultados obtenidos en las pruebas y la capacidad de desplazarse sobre terreno irregular y superficies inclinadas, demostrando de esta manera la eficacia de la red neuronal implementada.

Otra ventaja es el uso de servomotores con capacidades de envío de datos de: posición, torque y velocidad; lo que permite tener un sistema retroalimentado.

En la Universidad de las Fuerzas Armadas ESPE, Alomoto Ortiz & Jiménez Guamán (2015) desarrollaron el trabajo: “Diseño y construcción de un prototipo de robot móvil controlado con tecnología Raspberry Pi para la inspección y fumigación focalizada de cultivos de haba”. El control del robot se lo realizó mediante un Arduino Mega 2560, mientras que el procesamiento de las imágenes se lo ejecutó mediante un Raspberry Pi. Posee sensores ultrasónicos y una cámara para la aplicación de visión artificial para la abstracción del entorno en el que se desenvuelve el mecanismo.

El esquema de la estructura del robot se lo puede observar en la Figura 20, el cual cuenta con dos ruedas en la parte delantera accionadas por motores DC y una rueda loca para brindar estabilidad al mecanismo.



**Figura 20. Esquema del robot móvil para la inspección de cultivos de haba**

**Fuente: (Alomoto Ortiz & Jiménez Guamán, 2015)**

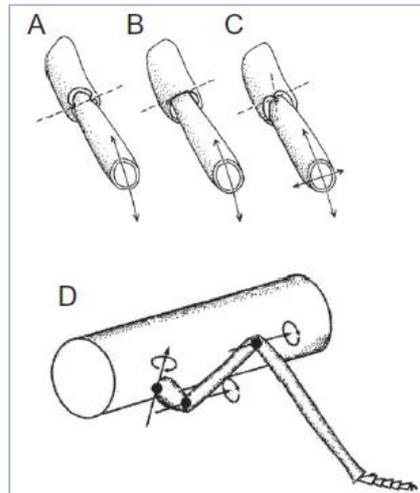
Entre sus características importantes están: el uso de software libre para su desarrollo, además del enfoque social y bajo costo que tiene el proyecto.

## **2.4 LOCOMOCIÓN TERRESTRE DE INSECTOS**

Un punto importante en el desarrollo de los robots caminantes, es el tipo de estructura para locomoción que presentan. Esta característica cambia por completo el algoritmo de control del mecanismo, al variar de un tipo a otro. La morfología de las extremidades y la caminata escogida son bases primordiales para el diseño del robot.

Forero, Mora, Loaiza, & Hernández (2015) en su estudio “Análisis para la simulación de modelos de animales tipo hexápodo”, muestran primero un análisis de la morfología y locomoción del robot antes de realizar un modelo de la estructura; estas particularidades inspiradas en el estudio biológico de los insectos, son analizadas por los autores que presenta una investigación de acuerdo a los aspectos fisiológicos de los mismos, basados en Klowden (2007).

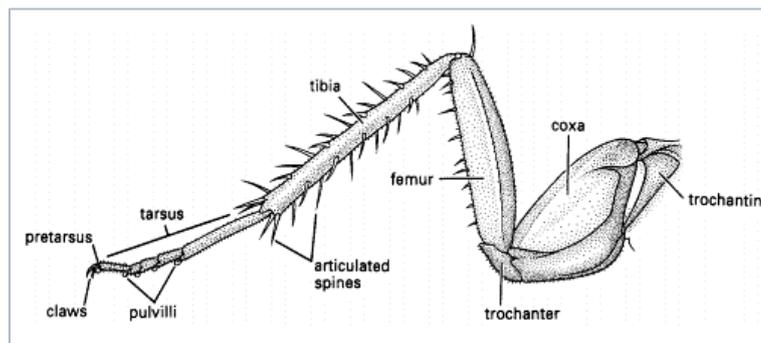
Klowden (2007) en “Sistemas Fisiológicos de los Insectos” muestra la disposición y estructura de la pata. Ver Figura 21. (ROS.org, 2017)



**Figura 21. Tipos de uniones en artrópodos. A. Junta de pivote. B. Junta articulada vertical. C. Junta monoarticulada. D. Grados de libertad de la pata.**

**Fuente: (Klowden, 2007)**

Se puede observar que biológicamente, el insecto consta de tres uniones en sus extremidades para el desarrollo de su movimiento. Siendo sus eslabones: coxa, fémur, tibia y tarso. Ver Figura 22.



**Figura 22. Partes de una extremidad de un insecto**

**Fuente: (Gullan & Cranston, 2009)**

En el diseño de la estructura por lo general el tarso es omitido debido a su funcionalidad de soporte de la extremidad, por lo tanto se obtiene un diseño con tres eslabones y tres grados de libertad.

#### 2.4.1 TIPOS DE PATAS SEGÚN EL MEDIO EN QUE SE DESARROLLAN

Ross, Charles, & Juneprrie (1982) en su libro “*A textbook of entomology*” definen los tipos de patas de acuerdo a su adaptación de caminata en la Figura 23.

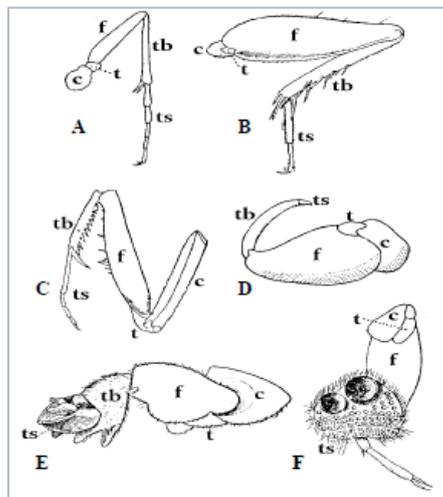


Figura 23. Tipo de patas de insectos. A. Pata caminante. B. Pata trasera (saltamontes). C. Pata delantera (mantis religiosa). D. Pata abrazadora delantera de un insecto. E. Pata delantera excavadora (grillo topo). F. Pata de sostenimiento (insecto acuático). c.

Coxa. t. Trocánter. f. Fémur. tb. Tibia. ts. Tarsos.

Fuente: (Ross, Charles, & Juneprrie, 1982)

Mientras que En *Tripod Gait Robotics* (2016), se divide a los tipos de extremidades según su especialización, en cinco grupos:

- Ambulatorias: patas especializadas en caminata.
- Cursoriales: patas especializadas en correr.
- Fosoriales: pata especializada en excavar.
- Natatorias: pata especializada en nadar.
- Rapaces: patas modificadas para capturar presas.

#### **2.4.2 COEFICIENTES DE CORRELACIÓN DE LONGITUD ENTRE LOS SEGMENTOS DE LA PATA**

En la investigación realizada por Fichter & Fichter (1988), se hace un estudio de relaciones de longitud entre los segmentos de la pata del insecto, definidos anteriormente en la Figura 22.

En este análisis se señala que los segmentos de las patas del insecto, específicamente arácnidos, se relacionan entre sí y crecen o disminuyen proporcionalmente de acuerdo al uso particular de la pata y su desarrollo en el medio ambiente que rodee al insecto.

Las regresiones de longitud entre el segmento más grande de la extremidad y los otros segmentos de la misma, varían en un rango de 0.81 a 0.96 (Fichter & Fichter, 1988).

De acuerdo a lo anterior, la longitud de la tibia tiene un alto coeficiente de correlación con la longitud del fémur, igual a 0.97. Mientras que, la longitud de la tibia no muestra ninguna correlación con el tarso. Además la longitud de la coxa tiene un coeficiente de correlación de 0.82 con la longitud del tarso (Fichter & Fichter, 1988).

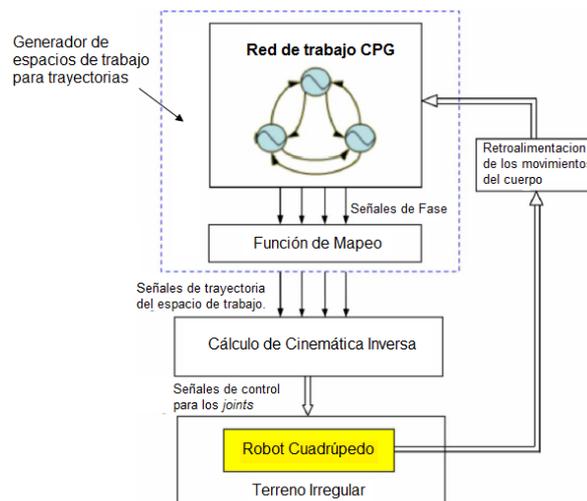
#### **2.5 ESTRUCTURAS DE CONTROL DEL ROBOT ZOOMÓRFICO**

El estudio de estructuras de control para los robots zoomórficos se basa esencialmente en los tipos de locomoción y métodos que utilizan los animales que se desenvuelven en terrenos irregulares y de difícil acceso. De las investigaciones que más resaltan se encontró el análisis realizado por Trimmer, Lin, Baryshyan, Leisk, & Kaplan (2012), en el que se estudia un movimiento llamado “suave” debido a la cantidad de patas y secciones que poseen insectos como las orugas. Dichas características implementadas en un robot permitirían una mejor adaptabilidad a terrenos desconocidos y mayor estabilidad. Estudios más específicos en el control de cuadrúpedos se presentarán a continuación como medios modernos y sobre todo estructuras de control de alto nivel para obtener robots mucho más autónomos y adaptables a terrenos desconocidos.

### 2.5.1 GENERADOR CENTRAL DE PATRONES

En el artículo realizado por Liu, Chen, & Wang (2011) se propone la realización de una arquitectura de control llamada “Generador Central de Patrones” (CPG). El tipo de estructura mencionado tiene la capacidad de generar espacios de trabajo y trayectorias dependiendo del terreno donde se movilice el robot. El funcionamiento del CPG se basa en la retroalimentación que generan los sensores implementados en la plataforma robótica para que así, compare los datos recibidos y determine los parámetros establecidos para el reconocimiento de un determinado terreno y su trayectoria más adecuada.

Para poder implementar un robot lo suficientemente adaptable al ambiente que lo rodea, el espacio de trabajo establecido y su trayectoria, desempeñan un papel clave al momento de incursionar en nuevos terrenos, ya que de ellos depende su estabilidad. La estructura analizada puede generar una adecuada forma de control para el tetrápodo a futuro, debido a que en la naturaleza, existen ambientes irregulares que cambian sus condiciones constantemente por lo que el robot tendrá que adaptarse continuamente a estos cambios siendo más versátil y autónomo. El esquema del control analizado se muestra en la Figura 24.



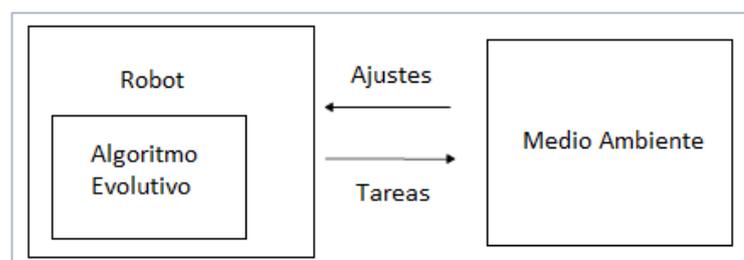
**Figura 24. Arquitectura del sistema de control de la locomoción**

**Fuente: (Liu, Chen, & Wang, 2011)**

## 2.5.2 ALGORITMO EVOLUTIVO

En el momento que un robot de varias extremidades ingresa en un terreno irregular, tiene la probabilidad de que su mecanismo se vuelva inestable o dar como resultado que alguna de sus extremidades pierda su contacto con el piso, debido a cambios bruscos de pendiente o debido a obstáculos del medio en el que se desenvuelve. Por lo tanto se hace necesario alguna arquitectura de control que pueda sobrellevar los problemas mencionados. El algoritmo evolutivo genera un control para guiar al robot a funcionar sin una extremidad y así también a estabilizarse por sí mismo en terrenos desconocidos al aprender sus características.

Según el artículo realizado por Qiu & Wu (2011), la clave de este tipo de control es la forma en la que se implementan los sensores en el robot para que puedan exponer fallas en alguna de las extremidades y así mismo, constantemente enviar datos de su funcionamiento excelente o deficiente dependiendo la situación. Una estructura de control como la mencionada puede aportar de forma significativa en robots caminantes que explorarán terrenos desconocidos, ya que pueden evolucionar en su funcionamiento a medida que van examinando nuevas condiciones del medio ambiente e incluso pueden sobrellevar el mal desempeño de una extremidad sin la intervención humana. El esquema de la robótica evolutiva se muestra en la Figura 25.



**Figura 25. Esquema de la Robótica Evolutiva**

**Fuente: (Qiu & Wu, 2011)**

## 2.6 CINEMÁTICA DE LAS EXTREMIDADES DEL ROBOT ZOOMÓRFICO

### 2.6.1 MÉTODOS DE ANÁLISIS CINEMÁTICO

Para el desarrollo de la cinemática directa e inversa con el objetivo de la localización de la extremidad en el espacio existen varios métodos, los más utilizados en este caso son el algoritmo de Denavit Hartenberg y el método geométrico, correspondientemente. Sin embargo los métodos propuestos en cada proyecto son amplios y varían de acuerdo al tipo de investigación y aplicación del robot; entre ellos:

- Método de la cinemática inversa
- Método de aproximación Fuzzy
- Método por matrices de transformación homogénea
- Método Jacobiano

El método de cinemática inversa y uso de la matriz de senos permite la localización en el espacio de la extremidad de tres grados de libertad mediante el cálculo de la cinemática inversa del robot por el método geométrico, mientras que para su posicionamiento o cinemática directa se efectúa el análisis de las funciones obtenidas en la cinemática inversa expresadas en funciones sinusoidales. La conclusión a la que se llegó es que este método es efectivo para el desarrollo de la cinemática en robots cuadrúpedos, siendo uno de los factores que permitieron que el robot mantenga su estabilidad frente a terrenos regular e irregular (Hidayat, Jati, & Saputra, 2016).

Método de aproximación Fuzzy: Li, Xiao, & Ge (2014) en su artículo "*Fuzzy approximation adaptive control of quadruped robots with kinematics and dynamics uncertainties*", desarrollan un control híbrido de movimiento/fuerza para compensar las perturbaciones externas. A pesar de que el estudio se basa en el control de la plataforma robótica, éste parte del análisis dinámico del cuadrúpedo y mediante una matriz de regresión se formula la parte cinemática del robot.

Método por matrices de transformación homogénea: Rong, Li, Ruan, & Song (2010) plantea un análisis cinemático tanto directo como inverso, mediante las matrices de transformación homogénea y desarrolla su comprobación mediante el software *Solid Works Motion*, concluyendo que la simulación cinemática del robot brinda grandes ventajas como el análisis en tiempo real de las variables en la estructura del robot.

Método Jacobiano: Chang et al. (2009) en su artículo “*Kinematics-based gait planning of a quadruped gecko-like model*”, desarrollan la cinemática directa del robot mediante el método de la matriz de transformación homogénea, mientras que para el análisis de la cinemática inversa se desarrolla el jacobiano del cuerpo entero y para la resolución del sistema redundante se aplica como solución el método de optimización de espacio nulo.

Los métodos mencionados anteriormente son una base pequeña de la cantidad de métodos y soluciones investigadas para el desarrollo del análisis cinemático del robot, teniendo entre otros: método jacobiano basado en elementos finitos (Zhang, Dequidt, Kruszewski, Largilliere, & Duriez, 2016), método de transposición para el análisis cinemático inverso (Hock, Drgona, Jaros, & Havrila, 2016), método de algoritmo genético para el desarrollo de modelos cinemáticos adaptables a distintos tipos de morfología (Büttner, Roennau, Heppner, Pfotzer, & Dillmann, 2016), etc. Sin embargo en el estudio de la cinemática inversa según Paramani (2011) de manera general existen simplemente dos métodos para su análisis: uno con respecto a las condiciones y tipo de robot, y el otro mediante soluciones numéricas.

### **2.6.2 CONSIDERACIONES INICIALES PARA EL ANÁLISIS**

Según Büttner et al. (2016) en su artículo “*Bio-inspired optimization of kinematic models for multi-legged walking robots*”, para la generación de la caminata del robot y el inicio de un análisis cinemático es necesario primero definir el tipo de morfología del cuadrúpedo. Los tipos de morfología más comunes según el autor son:

- Mamíferos: Sus patas están por debajo de su cuerpo y la unión que conforma la rodilla están dispuestas en dirección a favor o en contra de su movimiento (Morgan, 1925).
- Insectos: Sus patas están dispuestas a los lados del cuerpo con sus rodillas apuntando hacia arriba (Cruse, Dean, Muller, & Schmitz, 1991).
- Reptiles: Sus patas están dispuestas en un ángulo de 45 grados, sus rodillas y codos apuntan hacia adentro (Tedeschi & Carbone, 2014).

Según los tipos de morfologías se obtiene un diseño preliminar de la disposición geométrica de eslabones y uniones para su posterior estudio cinemático.

Otro punto importante al momento de desarrollar la cinemática inversa es la posibilidad de dejar la base del robot como flotante, sin embargo Mrva (2014) señala en su tesis de maestría "*Design of motion primitives for a hexapod walking robot operating in a rough environment*", que el desarrollo matemático se hace más complejo, por lo que debería considerarse en el estudio una base fija, simplificando de esta manera el análisis cinemático.

## 2.7 ANÁLISIS GLOBAL

El análisis del desarrollo y evolución de robots móviles con extremidades permite tener una visión clara de las características mínimas de diseño para el correcto desempeño del robot, entre ellas:

- El análisis real del movimiento de los animales con extremidades que son base del modelo del mecanismo a desarrollarse, es de suma importancia para el mejor entendimiento de las características de locomoción, que repercute en un buen desempeño del algoritmo del robot.
- La elección del número de grados de libertad por extremidad, influye significativamente en el espacio de trabajo del robot y también en su estabilidad frente a terrenos difíciles. Tres grados de libertad, le brindan al robot eficacia en las características mencionadas anteriormente.
- La implementación de un control mínimo para la movilidad del robot es de gran importancia. En el robot hexápodo desarrollado por Irawan et al. (2014), citado en la Tabla 1, se puede observar la falta de versatilidad en la capacidad del operario para maniobrar el robot. Esto se debe a que demandaba un conocimiento exhaustivo del funcionamiento del robot para poder controlarlo, además de una gran presión de trabajo al tener que coordinar varios grados de libertad para que el robot permanezca en pie y no pierda estabilidad.

Por otra parte, el desarrollo de robots caminantes en el Ecuador ha evolucionado, sobre todo en los últimos años. El estudio de los distintos enfoques que se dieron a estos proyectos permite rescatar características de diseño que posibilitaron que los robots se desempeñen de manera más eficaz y estable, entre éstas:

- El aluminio es uno de los materiales más usado para este tipo de proyectos, debido a su resistencia y bajo peso.

- Los servomotores mencionados en el proyecto de tesis de Machuma Mendoza & López Ayala (2015), son ideales para el manejo de aplicaciones en robots caminantes que no alcancen gran tamaño. Sin embargo su costo limita mucho el uso de los mismos.
- Una de las características que más se destaca en la estabilidad del robot, es el análisis estático y dinámico del mecanismo, lo cual brinda un mejor desempeño al momento de movilizarse y transitar por terrenos inhóspitos o de regularidad media.
- El análisis cinemático del movimiento del robot es de vital importancia para su desarrollo. Algunos proyectos que desarrollaron de forma empírica el posicionamiento de sus extremidades, tuvieron cierta funcionalidad, sin embargo afectaron la estabilidad del mecanismo.
- Siempre se debe elegir un tipo de movimiento adecuado para la locomoción del robot, de acuerdo al polígono de estabilidad que forman sus extremidades.
- Las plataformas robóticas de software libre son más completas y permiten desarrollar de mejor manera proyectos de este tipo. *Robot Operating System* demostró ser una herramienta muy eficaz, en el proyecto de titulación de la Universidad de Queensland, desarrollado por Erazo Sosa (2014). Siendo también flexible para trabajar con distintas interfaces gráficas para el manejo del usuario.
- La implementación de una interfaz gráfica amigable con el usuario y que posea todas las características necesarias para el correcto manejo del robot es esencial en el desarrollo del proyecto.

## CAPÍTULO 3. DISEÑO MECATRÓNICO

### 3.1 MODELAMIENTO ROBÓTICO

El estudio cinemático y dinámico del robot permiten determinar los parámetros mínimos necesarios para el diseño de su estructura. Por dicha razón se va a presentar un modelamiento del robot para el posterior desarrollo del diseño de los componentes mecánicos del cuadrúpedo.

#### 3.1.1 ANÁLISIS CINEMÁTICO

##### 3.1.1.2 CINEMÁTICA DIRECTA DE LA PATA

Se elaboró un diseño preliminar de la estructura del robot, tomando en cuenta la estructura de locomoción de insectos definida por Klowden (2007).

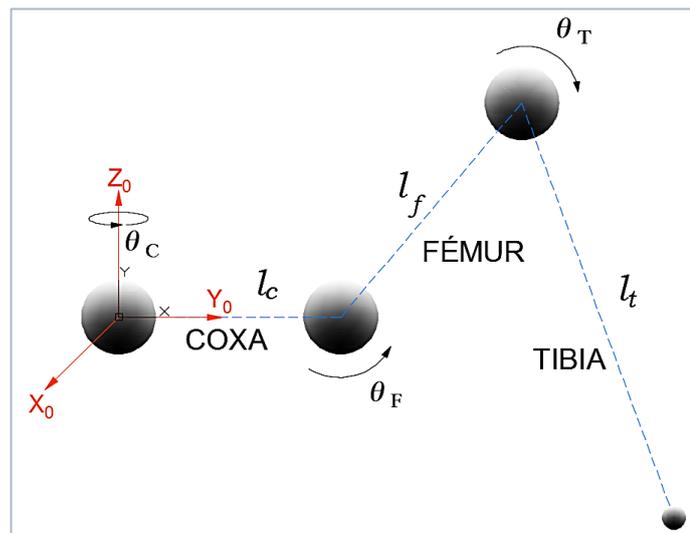
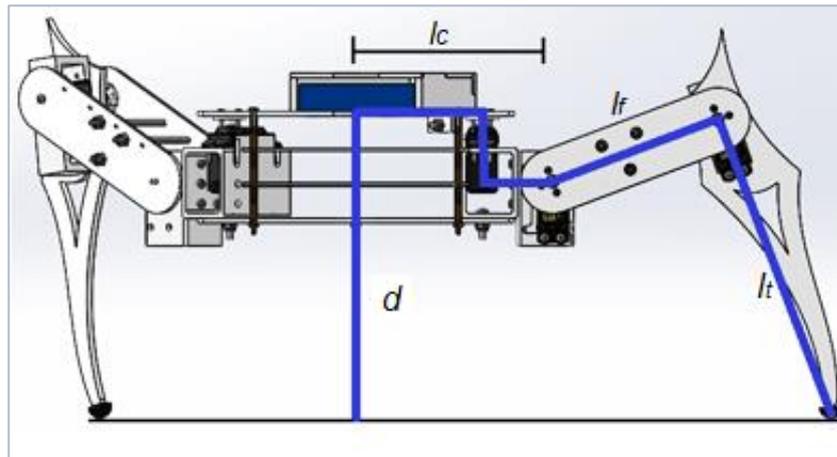


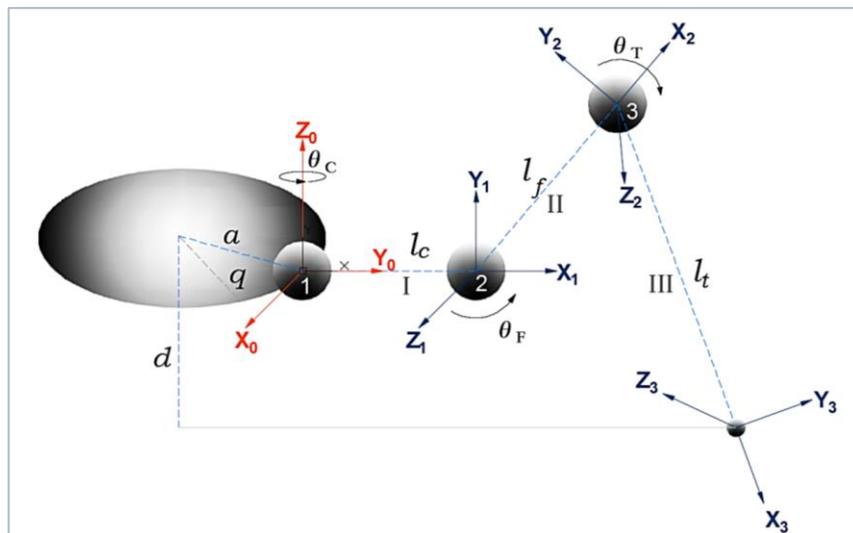
Figura 26. Distribución de uniones y eslabones de la pata.

En la Figura 26 se observa la disposición de uniones y eslabones de la extremidad del robot, según las partes necesarias para el desarrollo de la locomoción, determinadas por Gullan & Cranston (2009). A partir del esquema de la Figura 26 se desarrollará el análisis de la cinemática directa mediante el algoritmo de Denavit Hartenberg.



**Figura 27. Parámetros de Denavit Hartenberg – Modelo real**

La Figura 27 representa las distancias tomadas de los eslabones con respecto al sistema de referencia global de la extremidad que está en el suelo proyectado verticalmente a una distancia  $d$  del centro de gravedad del cuerpo, con el objetivo de realizar el análisis mencionado anteriormente. El esquema final con los parámetros del algoritmo se observan en la Figura 28.



**Figura 28. Parámetros de Denavit Hartenberg de la extremidad del robot.**

Los parámetros obtenidos de la Figura 28 se desarrollan en la Tabla 2. Al eslabón que conforma la coxa se le va a dar una pequeña diferencia de alturas para poder desarrollar el diseño sin ningún inconveniente. En caso de adaptar el diseño real a una distancia vertical nula, se eliminaría dicha constante.

Tabla 2

Parámetros de Denavit Hartenberg de la extremidad del robot cuadrúpedo

Articulación	$\theta$	$d$	$\alpha$	$a$
I	$q$	$d$	0	$a$
<i>Coxa</i> – 1	$q_1$	$d_c$	$-\pi/2$	$a_c$
<i>Fémur</i> – 2	$q_2$	0	0	$l_f$
<i>Tibia</i> – 3	$q_3$	0	0	$l_t$

A partir de la Tabla 2, se realiza la composición de matrices de transformación homogéneas del problema cinemático de acuerdo a la Ec. 1 planteada por Barrientos, Peñín, Balaguer, & Aracil (1996).

$${}^{i-1}A_i = \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. 1}$$

Según la Ec. 1 se desarrolla las matrices para cada articulación.

$${}^0A_I = \begin{bmatrix} \cos(q) & -\sin(q) & 0 & a \cos(q) \\ \sin(q) & \cos(q) & 0 & a \sin(q) \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. 2}$$

$${}^I A_1 = \begin{bmatrix} \cos(q_1) & 0 & -\sin(q_1) & a_c \cos(q_1) \\ \sin(q_1) & 0 & \cos(q_1) & a_c \sin(q_1) \\ 0 & -1 & 0 & d_c \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. 3}$$

$${}^1 A_2 = \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & l_f \cos(q_2) \\ \sin(q_2) & \cos(q_2) & 0 & l_f \sin(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. 4}$$

$${}^2 A_3 = \begin{bmatrix} \cos(q_3) & -\sin(q_3) & 0 & l_t \cos(q_3) \\ \sin(q_3) & \cos(q_3) & 0 & l_t \sin(q_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. 5}$$

La matriz de localización  $T$  del extremo de la pata del robot con respecto al sistema de referencia inicial viene dada por:

$$T = {}^0A_3 = {}^0A_I {}^1A_1 {}^2A_2 {}^3A_3 \quad \text{Ec. 6}$$

Para lo cual se multiplicará las ecuaciones: Ec. 2, Ec. 3, Ec. 4 y Ec. 5 respectivamente.

$$T = {}^0A_3 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. 7}$$

$$T = {}^0A_3 = \begin{bmatrix} \cos(q + q_1) \cos(q_2 + q_3) & -\cos(q + q_1) \sin(q_2 + q_3) & -\sin(q + q_1) & p_x \\ \cos(q_2 + q_3) \sin(q + q_1) & -\sin(q + q_1) \sin(q_2 + q_3) & \cos(q + q_1) & p_y \\ -\sin(q_2 + q_3) & -\cos(q_2 + q_3) & 0 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La resolución del problema cinemático está dado por la matriz de la Ec. 7 de Barrientos et al. (1996). En donde  $n$ ,  $o$  y  $a$  representan la orientación y el vector  $p$  representa la posición.

Finalmente, las ecuaciones que representan la posición del extremo de la pata con respecto a la base del robot, correspondientes al vector posición de la cuarta columna de la matriz de localización son:

$$p_x = \cos(q + q_1) (a_c + l_f \cos(q_2) + l_t \cos(q_2 + q_3)) + a \cos(q) \quad \text{Ec. 8}$$

$$p_y = \sin(q + q_1) (a_c + l_f \cos(q_2) + l_t \cos(q_2 + q_3)) + a \sin(q) \quad \text{Ec. 9}$$

$$p_z = d + d_c - l_t \sin(q_2 + q_3) - l_f \sin(q_2) \quad \text{Ec. 10}$$

### 3.1.1.3 CINEMÁTICA INVERSA DE LA PATA

Desarrollada la Ec. 7 y obtenidas las ecuaciones de posición Ec. 8, Ec. 9 y Ec. 10 se encontrará la cinemática inversa del robot a partir de la matriz de transformación homogénea.

Relacionando las ecuaciones: Ec. 8 y Ec. 9; se obtiene la primera ecuación que define la cinemática inversa del robot, Ec. 12:

$$q_1 = \tan^{-1} \left( \frac{p_x - a \cos(q)}{p_y - a \sin(q)} \right) - q \quad \text{Ec. 11}$$

A partir de la Ec. 6 se tendrá que:

$$\begin{aligned} ({}^0A_I)^{-1}T &= {}^1A_1 {}^1A_2 {}^2A_3 \\ ({}^1A_1)^{-1}({}^0A_I)^{-1}T &= {}^1A_2 {}^2A_3 \end{aligned} \quad \text{Ec. 12}$$

Desarrollando la parte izquierda de la Ec. 12, se tiene la Ec. 13:

$$\begin{aligned} &({}^1A_1)^{-1}({}^0A_I)^{-1}T \\ &= \begin{bmatrix} n_x c_{01} + n_y s_{01} & o_x c_{01} + o_y s_{01} & a_x c_{01} + a_y s_{01} & p_x' \\ -n_z & -o_z & -a_z & p_y' \\ n_y c_{01} - n_x s_{01} & o_y c_{01} - o_x s_{01} & a_y c_{01} - a_x s_{01} & p_z' \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad \text{Ec. 13}$$

En donde,  $c_{01} = \cos(q + q_1)$  y  $s_{01} = \sin(q + q_1)$ ; además la cuarta columna de la matriz, está dada por la Ec. 14.

$$\begin{aligned} p_x' &= p_x \cos(q + q_1) - a_c + p_y \sin(q + q_1) - a \cos(q_1) \\ p_y' &= d + d_c - p_z \\ p_z' &= p_y \cos(q + q_1) - p_x \sin(q + q_1) + a \sin(q_1) \end{aligned} \quad \text{Ec. 14}$$

Desarrollando la parte derecha de la Ec. 12, se tiene la Ec. 15:

$${}^1A_2 {}^2A_3 = \begin{bmatrix} \cos(q_2 + q_3) & -\sin(q_2 + q_3) & 0 & p_x'' \\ \sin(q_2 + q_3) & \cos(q_2 + q_3) & 0 & p_y'' \\ 0 & 0 & 1 & p_z'' \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{Ec. 15}$$

En donde, la cuarta columna de la matriz está dada por la Ec. 16:

$$\begin{aligned} p_x'' &= l_t \cos(q_2 + q_3) + l_f \cos(q_2) \\ p_y'' &= l_t \sin(q_2 + q_3) + l_f \sin(q_2) \\ p_z'' &= 0 \end{aligned} \quad \text{Ec. 16}$$

Igualando la Ec. 13 con la Ec. 16, se obtienen las ecuaciones que definen los otros grados de libertad del robot, desarrolladas en las ecuaciones Ec. 17 y Ec. 18.

$$q_2 = 2 \tan^{-1} \left\{ \frac{\left[ 2l_f o - (-l_f^4 + 2l_f^2(l_t^2 + o^2 + p^2) - l_t^4 + 2l_t^2(o^2 + p^2) - o^4 - 2o^2 p^2 - p^4)^{\frac{1}{2}} \right]}{(l_f^2 + 2l_f p - l_t^2 + o^2 + p^2)} \right\} \quad \text{Ec. 17}$$

$$q_3 = 2 \tan^{-1} \left\{ \frac{\left[ (-l_f^2 + 2l_f l_t - l_t^2 + o^2 + p^2)(l_f^2 + 2l_f l_t + l_t^2 - o^2 - p^2) \right]^{\frac{1}{2}}}{(-l_f^2 + 2l_f l_t - l_t^2 + o^2 + p^2)} \right\} \quad \text{Ec. 18}$$

En donde,  $p$  y  $o$  corresponden a los valores de  $p_x'$  y  $p_y'$  respectivamente, de la Ec. 16.

Finalmente, las ecuaciones: Ec. 11, Ec. 17 y Ec. 18 definen por completo la cinemática inversa del robot.

### 3.1.2 PARÁMETROS DEL ROBOT

Debido a que el robot del proyecto va a desarrollarse como un prototipo para siguientes estudios en el Laboratorio de Instrumentación de Universidad de las Fuerzas Armadas ESPE, los parámetros para su desarrollo son los indicados en la Tabla 3.

**Tabla 3**

**Parámetros físicos generales del robot**

Parámetros	Especificación
Material	De elección libre
Tamaño	De elección libre
Número de extremidades	Seis o cuatro
Locomoción	Insectos

En base al tipo de locomoción deseada, se conformó al robot de las siguientes partes:

- Cuerpo.

- Cuatro extremidades conformadas por: coxa, fémur y tibia. Concernientes al desarrollo cinemático expresado anteriormente en la Figura 28.

El tamaño del robot es de elección libre como se puede apreciar en la Tabla 3. Por lo tanto, las dimensiones del robot se diseñaron conforme al tamaño de su cuerpo, de manera que las mismas puedan delimitar con su estructura una placa de máximo diez milímetros y una batería LiPo de tamaño estándar, que comprenderá el control del mecanismo detallado posteriormente.

El tipo de pata del robot se diseñará como la ambulatoria o caminante. Ver Figura 23.

Por otra parte, se observa un factor de correlación entre tibia y fémur de 0.97 en arácnidos. Sin embargo, parte del tarso se ubica antes del contacto con el suelo en varias especies de insectos caminantes, por lo que forma una especie de articulación fémur-tarso entre la juntura de la tibia y el piso. Conforme a lo mencionado se va a dar un factor de correlación entre la longitud de la tibia y la longitud del fémur mayor a 1.

Las constantes que cumplen con los parámetros anteriores y definen la estructura del robot se observan en la Tabla 4.

**Tabla 4**  
**Definición de las longitudes del robot**

Constante	Longitud
$d$	181.483 mm
$a$	110.871 mm
$d_c$	-41 mm
$a_c$	43.8 mm
$l_f$	110 mm
$l_t$	190.85 mm

### 3.1.3 ESPACIO DE TRABAJO

Para poder realizar el análisis del espacio de trabajo de la extremidad es necesario obtener previamente las restricciones físicas del robot.

Subsiguientemente se obtiene el espacio de trabajo para cada extremidad mediante un método escogido para su desarrollo.

El método en el cual se basó la obtención del espacio de trabajo fue el estudio de Cao, Lu, Li, & Zang (2011) denominado Monte Carlo, aplicado posteriormente.

### 3.1.3.1 LÍMITES MÍNIMO Y MÁXIMO DE OPERACIÓN DE LAS UNIONES EN BASE A LAS RESTRICCIONES FÍSICAS DEL ROBOT

Los límites angulares del movimiento de los motores se definieron de acuerdo a las restricciones físicas del robot, señaladas en la Figura 29 y la Figura 30.

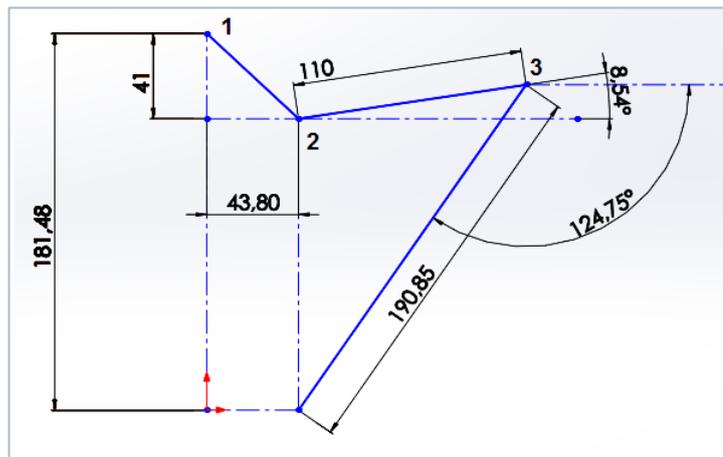
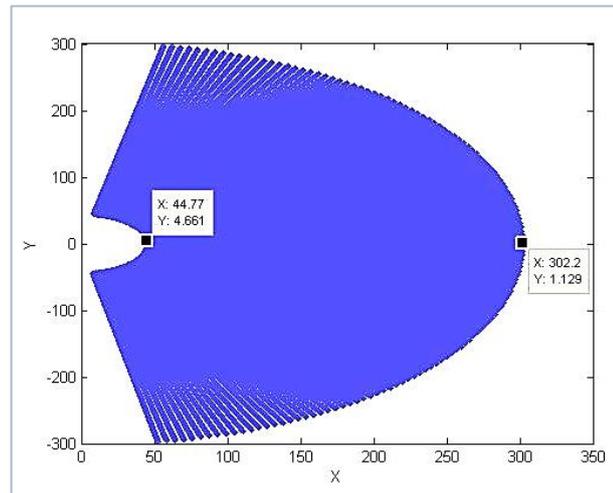


Figura 29. Límites angulares mínimos de las uniones 2 y 3.



Debido a que en el proyecto realizado solo se necesita calcular el espacio de trabajo en dos dimensiones sin tomar en cuenta el eje z, el método mencionado será suficiente para alcanzar este objetivo.



**Figura 31. Espacio de trabajo Método Monte Carlo.**

Para encontrar la gráfica de la Figura 31 se muestrean los valores que adquieren los ángulos de las uniones entre los límites físicos definidos anteriormente.

De la Figura 31 se obtienen los límites del espacio de trabajo de la extremidad en el plano  $XY$ , en dónde se adquiere aproximadamente los puntos máximos y mínimos:

$$\begin{aligned} \lim_{MCx}(44,300)mm \\ \lim_{MCy}(-300,300)mm \end{aligned}$$

### 3.1.4 GENERACIÓN DE LOCOMOCIÓN DEL ROBOT

La generación de pasos o locomoción está definida por el tipo de movimiento que tienen las extremidades del robot zoomórfico en coordinación con el movimiento del cuerpo al desplazarse.

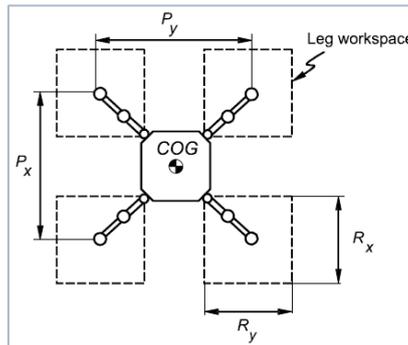
Para definir el tipo de locomoción existen varias características que permiten a la vez el análisis y desarrollo de la generación de pasos. Santos, García, & Estremera (2007) en “*Quadrupedal Locomotion*”, realiza un estudio sobre la generación de movimiento con los parámetros presentados a continuación.

1. El factor de trabajo de la pata  $i$ ,  $\beta_i$ , es la fracción de ciclo en la cual la pata permanece en el suelo (de Santos, García, & Estremera, 2007).
2. La fase de la pata  $i$ ,  $\phi_i$ , es el tiempo en el cual se da el posicionamiento de la pata  $i$  en el suelo justo después del posicionamiento de la pata 1, las fases para cada extremidad se definen en la Ec. 19 (de Santos, García, & Estremera, 2007).

$$\begin{aligned}\phi_1 &= 0 \\ \phi_2 &= \frac{1}{2} \\ \phi_3 &= \beta \\ \phi_4 &= F\left(\beta - \frac{1}{2}\right)\end{aligned}\tag{Ec. 19}$$

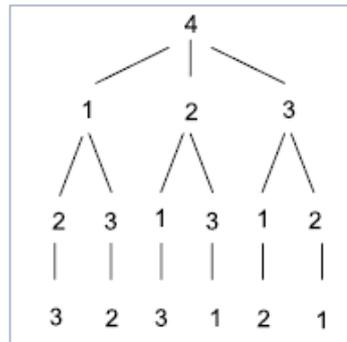
$F$  es una función por partes, en donde:  $Y = F(X)$  e  $Y$  está definida por:  $X$  si  $X \geq 0$  y  $1 - |X|$  si  $X < 0$ .

3. El recorrido de la pata,  $R$ , definido por  $R_x$  y  $R_y$  (de Santos, García, & Estremera, 2007). Ver Figura 32.
4. El paso del recorrido,  $P$ , definido por  $P_x$  y  $P_y$  (de Santos, García, & Estremera, 2007). Ver Figura 32.
5. La longitud de paso,  $\lambda$ , es la distancia recorrida desde el centro de gravedad del cuerpo a lo largo de un ciclo de locomoción. Si el movimiento es periódico se tiene que  $\lambda = R/\beta$  (de Santos, García, & Estremera, 2007).



**Figura 32. Definiciones geométricas**  
**Fuente: (de Santos, García, & Estremera, 2007)**

Para mantener la estabilidad estática de un cuadrúpedo es necesario que tres patas permanezcan en fase de soporte mientras una está en fase de transferencia o movimiento. Por lo tanto, el número máximo de combinaciones de secuencias de movimientos es seis. Ver Figura 33.



**Figura 33. Secuencias posibles de las extremidades**  
**Fuente: (de Santos, García, & Estremera, 2007)**

De acuerdo a lo formulado anteriormente, las características que definen la generación de movimiento del robot en el presente proyecto son las determinadas en la Tabla 5.

Tabla 5

Valores tomados en cuenta en la generación de movimiento del robot

No.	Definición	Valor
1	Factor de trabajo de las extremidades	$\beta$
2	Fases de las extremidad	$\phi_1 = 0$ $\phi_2 = \frac{1}{2}$ $\phi_3 = \beta$ $\phi_4 = F\left(\beta - \frac{1}{2}\right)$
3	Recorrido de la extremidad	$R_x = 170 \text{ mm}$ $R_y = 170 \text{ mm}$
4	Paso del recorrido	$P_x = 336.8 \text{ mm}$ $P_y = 336.8 \text{ mm}$
5	Longitud de paso	$\lambda = \frac{254.56 \text{ mm}}{\beta}$

Las características de la Tabla 5 que pertenecen a magnitudes de tiempo serán definidas por completo al momento de determinar los actuadores a utilizarse y su tiempo de respuesta. Los demás valores fueron obtenidos de acuerdo a las características geométricas del robot y el espacio de trabajo.

### 3.1.4.1 TIPO DE LOCOMOCIÓN

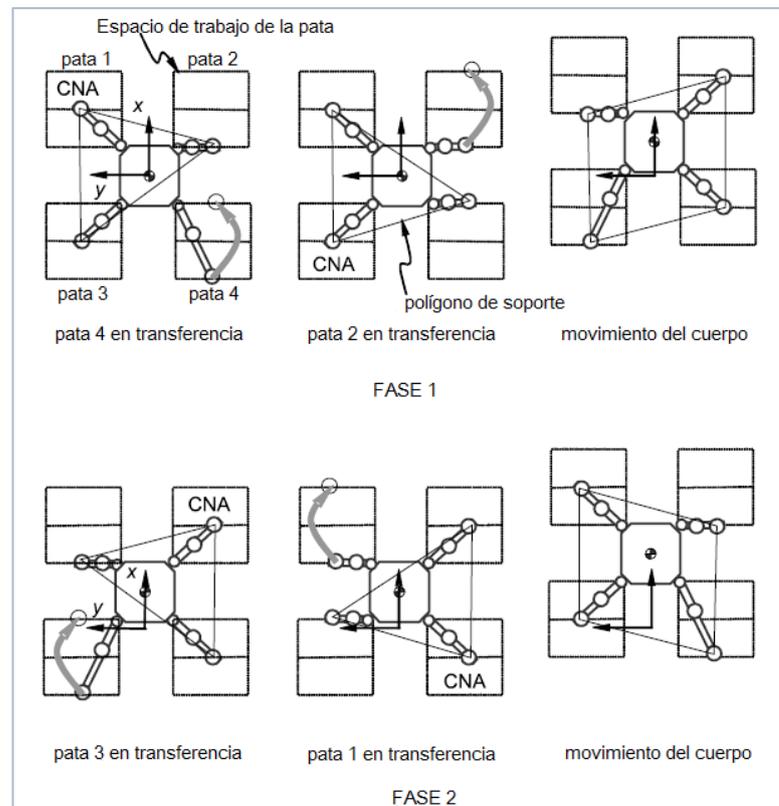
El tipo de locomoción seleccionada para el desarrollo de este proyecto es la caminata discontinua periódica de dos fases. En la cual, la fase de transferencia de una pata se efectúa con las otras tres soportadas sobre el piso y el cuerpo es impulsado hacia adelante por el movimiento simultáneo de las cuatro extremidades, mientras éstas permanecen en contacto con el suelo (de Santos, García, & Estremera, 2007).

Para que el movimiento se dé con este tipo de locomoción, es necesario:

- a. Si una extremidad se encuentra en su límite cinemático posterior en fase de soporte, cambiará en su fase de transferencia a su límite cinemático frontal (de Santos, García, & Estremera, 2007).

- b. Al menos una extremidad debe permanecer en su límite posterior cinemático después del movimiento del cuerpo, para poder realizar una siguiente fase de transferencia (de Santos, García, & Estremera, 2007). Permitiendo de esta manera que el movimiento sea periódico.
- c. La pata que es contralateral y no adyacente (CNA) con respecto a la que se encuentra en fase de transferencia, debe ser colocada de cierta forma que después de la ubicación de la pata en fase de transferencia, el centro de gravedad permanezca en el interior del polígono de estabilidad formado por las extremidades en fase de soporte (de Santos, García, & Estremera, 2007).

La Figura 34 explica gráficamente el tipo de locomoción a desarrollarse, como también el movimiento sucesivo de las extremidades y el cuerpo para la ejecución del movimiento.



**Figura 34. Caminata discontinua de dos fases**  
**Fuente: (de Santos, García, & Estremera, 2007)**

### **3.1.4.1.1 GENERACIÓN DE CAMINATA DISCONTINUA DE DOS FASES**

El tipo específico de locomoción escogido es el de caminata discontinua de dos fases. Ver Figura 34.

Como su nombre lo expresa, este tipo de marcha posee dos fases, las cuales constan de dos movimientos de transferencia de una extremidad consecutivos y de un movimiento del cuerpo que permitirá que el robot sea propulsado hacia adelante. Por lo tanto, la generación completa de locomoción constará de cuatro movimientos de transferencia de las extremidades y de dos movimientos del cuerpo del robot, desarrollándose secuencial y periódicamente para generar una trayectoria.

La secuencia que entrega la mayor estabilidad para este tipo de caminata, es la indicada en la Figura 34, que consta de las siguientes partes:

1. Movimiento de la pata 4 (Fase 1).
2. Movimiento de la pata 2 (Fase 1).
3. Movimiento del cuerpo (Fase 1).
4. Movimiento de la pata 3 (Fase 2).
5. Movimiento de la pata 1 (Fase 2).
6. Movimiento del cuerpo (Fase 2).

Además, según Siciliano & Khatib (2008) la secuencia correcta para la caminata de arrastre recta es 1-4-2-3, que pertenece a la descrita anteriormente pero empezando con una extremidad diferente.

Con el análisis de la cinemática directa e inversa de la extremidad, la generación de movimiento se da mediante la colocación de las patas secuencialmente en el interior del espacio de trabajo determinado.

### 3.1.4.1.2 GENERACIÓN DE CAMINATA DISCONTINUA GIRATORIA

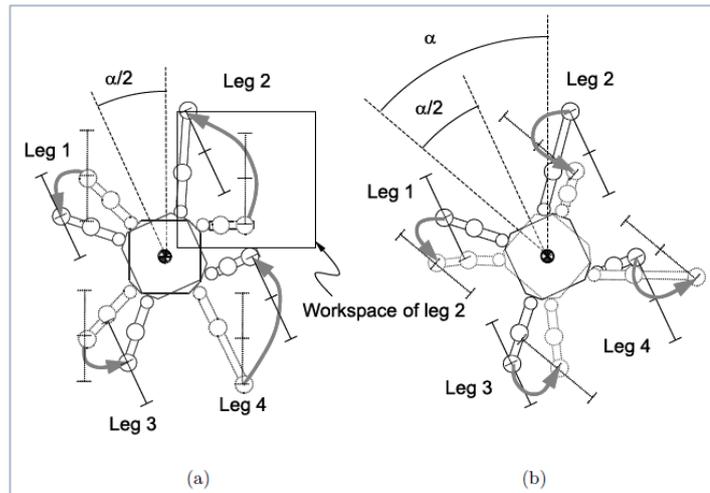
El tipo de caminata detallada anteriormente permite el seguimiento de una trayectoria en línea recta. Sin embargo, para la generación de pasos que permita el giro del robot cuadrúpedo, se cumplen las mismas definiciones y características que la anterior forma de caminata.

- Se mantiene una pata en el límite cinemático posterior en la fase de soporte, la misma que cambiará en la fase de transferencia al límite cinemático anterior del espacio de trabajo, para mantener un movimiento secuencial del cuadrúpedo.
- Los puntos de soporte en el espacio de trabajo de cada pata al inicio de la primera fase deben ser iguales a los puntos de soporte al final de la segunda fase para conservar la continuidad del movimiento.
- Las extremidades deben ser ubicadas dentro del espacio de trabajo de tal forma que el robot mantenga estabilidad.

La caminata discontinua giratoria al igual que la anterior consta de dos fases que conforman el movimiento completo del cuadrúpedo. Sin embargo, cada fase consta de cuatro movimientos de extremidades en lugar de dos como se determinó en el anterior movimiento detallado en la Figura 34.

Para conformar la primera fase se mueven primero dos patas secuencialmente y luego el cuerpo es rotado, manteniendo las otras extremidades en sus puntos de soporte. Para ubicar las patas en sus posiciones respectivas son necesarios dos movimientos de transferencia más (de Santos, García, & Estremera, 2007).

Para este tipo de movimiento, según de Santos, García, & Estremera (2007) la secuencia que le brinda mayor estabilidad al cuadrúpedo para la primera fase es 3-4-2-1 y para la segunda fase es 1-2-4-3.



**Figura 35. Caminata discontinua giratoria. a. Primera fase. b. Segunda fase**  
**Fuente: (de Santos, García, & Estremera, 2007)**

En la Figura 35(a) se puede identificar las posiciones iniciales del cuerpo y de las extremidades con línea punteada y las posiciones finales con línea continua. En la Figura 35(b) se identifica las posiciones iniciales y finales con línea continua y con línea punteada respectivamente (de Santos, García, & Estremera, 2007).

Se puede apreciar en la Figura 35 que el posicionamiento de las patas en fase de transferencia debe estar localizado dentro del espacio de trabajo de la pata, como se puede ver en el recuadro de la Figura 35(a) que representa el espacio de trabajo real de la pata 2 necesario para poder realizar su movimiento de transferencia. Por lo tanto, el recorrido de la extremidad escogido para este movimiento debe ser menor que el espacio de trabajo real de la extremidad del robot (de Santos, García, & Estremera, 2007).

Como se puede observar en la Figura 35, es necesario completar los cuatro movimientos de transferencia para que el robot cumpla una fase y gire la mitad del ángulo de rotación, la otra mitad del ángulo será alcanzado en la siguiente fase.

El sentido de giro en este tipo de caminata dependerá únicamente del sentido del ángulo. Si se desea un sentido de giro antihorario como el desarrollado en la Figura 35, el ángulo debe ser positivo; si se desea en cambio, un sentido de giro horario el ángulo debe ser negativo.

### 3.1.5 GENERACIÓN DE LA TRAYECTORIA DE LA EXTREMIDAD

La trayectoria se conforma de varios puntos por los cuales la extremidad del robot tiene que pasar a una designada velocidad.

La trayectoria de la extremidad elegida para el tipo de caminata discontinua de dos fases es una trayectoria elíptica, ideal incluso para desarrollo de marcha tipo galope en robots cuadrúpedos, debido a que contribuye en la estabilidad del robot.

Debido a que la posición en el eje  $Y$  se mantiene constante, se aplicará la Ec. 20 y la Ec. 21 correspondientes a la elipse parametrizada para los ejes:  $X$  e  $Z$ .

$$x(t) = \rho_1 \cos(\omega t) + x_0 \quad \text{Ec. 20}$$

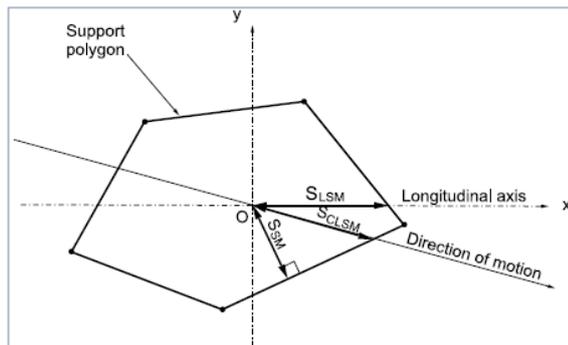
$$z(t) = \rho_2 \sin(\omega t) + z_0 \quad \text{Ec. 21}$$

En donde:  $\rho_1$  y  $\rho_2$  son respectivamente la mitad de la longitud al eje mayor y menor;  $x_0$  e  $y_0$  corresponden a la posición del centro de la elipse.

Este método fue aplicado y desarrollado por Kim & Park (2008) para robots cuadrúpedos con el tipo de locomoción de mamíferos, para desarrollar marcha normal y galope. La Ec. 20 y la Ec. 21 dependen de la velocidad de la extremidad, por lo que se muestreó puntos de acuerdo a la misma.

### 3.1.6 ANÁLISIS DE ESTABILIDAD ESTÁTICA DEL CUADRÚPEDO

Se han investigado distintos criterios de estabilidad de acuerdo a: el tipo de terreno que debe superar el robot, efectos de inercia y velocidad. La Figura 36 muestra los criterios de estabilidad principales desarrollados para robots caminantes.



**Figura 36. Márgenes de estabilidad y polígono de estabilidad.**

**Fuente: (de Santos, García, & Estremera, 2007)**

En donde  $S_{SM}$ ,  $S_{LSM}$  y  $S_{CLM}$  de la Figura 36, están definidas por:

El margen de estabilidad estática (*Static Stability Margin*)  $S_{SM}$ , es definido por McGhee & Frank (1968), como la menor distancia de la proyección del centro de gravedad a las líneas que definen el polígono de soporte.

El  $S_{SM}$  es óptimo para el análisis de robots sobre terrenos conocidos, sin embargo su cálculo es muy extenso y complicado, por lo que Zhang & Song (1989) definen el margen de estabilidad longitudinal (*Longitudinal Stability Margin*)  $S_{LSM}$ , el cual es una aproximación eficiente del  $S_{SM}$  y es más fácil de calcular.

El margen de estabilidad longitudinal de cangrejo (*Crab Longitudinal Stability Margin*)  $S_{CLSM}$ , se define por Zhang & Song (1990), para robots caminantes no ideales con efectos de inercia y aceleración, representa la menor distancia de la proyección del centro de gravedad a las líneas frontales o traseras del polígono de soporte a lo largo del de la dirección del movimiento, es más utilizado en robots de seis extremidades en adelante.

De estos tres criterios de estabilidad Santos, García, & Estremera (2007) selecciona el margen de estabilidad longitudinal  $S_{LSM}$  por:

- Su desempeño aproximado al margen de estabilidad estática.
- Por la facilidad de cálculo.

- Porque es ideal para robots caminantes sobre terrenos conocidos, como es el caso del proyecto a desarrollarse.

### 3.1.6.1 MARGEN DE ESTABILIDAD LONGITUDINAL

El margen de estabilidad longitudinal  $S_{LSM}$ , está determinado como la distancia más corta desde la proyección vertical del centro de gravedad del robot a los límites frontal y posterior del polígono de soporte a lo largo del eje longitudinal (Zhang & Song, 1989).

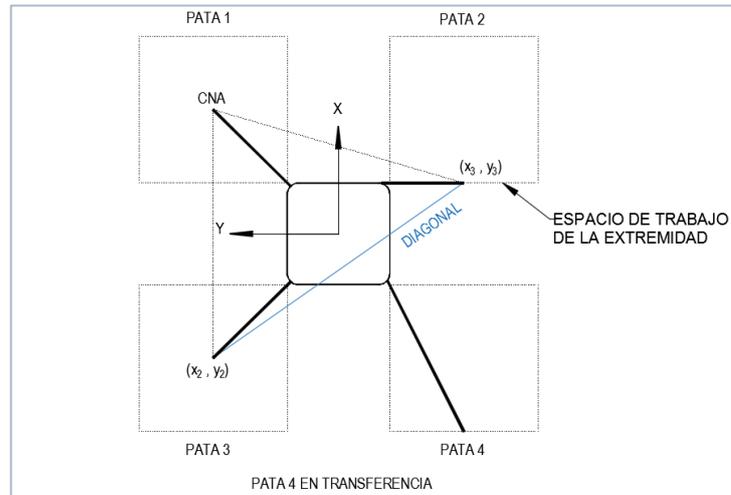
#### 3.1.6.1.1 ANÁLISIS DE CAMINATA DISCONTINUA DE DOS FASES RECTA

El margen de estabilidad longitudinal según la definición detallada anteriormente en la Figura 36 está definida por dos puntos de soporte de extremidades CNA, determinados siempre por una extremidad ubicada en medio espacio de trabajo y otra ubicada en su límite cinemático cerca de la proyección vertical del centro de gravedad del robot (de Santos, García, & Estremera, 2007). Se puede obtener una visión más clara de esta definición en la Figura 37.

La Ec. 22 desarrollada en de Santos, García, & Estremera (2007), permite el cálculo del margen de estabilidad longitudinal de caminata discontinua con trayectoria rectilínea. En donde,  $(x_2, y_2)$  y  $(x_3, y_3)$  son los puntos inicial y final, respectivamente; correspondientes a los puntos de soporte de las patas 2 y 3.

$$S_{LSM_D} = \left| -y_2 \left( \frac{x_3 - y_2}{y_3 - y_2} \right) + x_2 \right|$$

Ec. 22



**Figura 37. Puntos de soporte que determinan la diagonal del Margen de Estabilidad Longitudinal**

La Figura 37 muestra los puntos de soporte de las extremidades contralaterales no adyacentes al momento en que la pata 4 se encuentra en movimiento de fase de transferencia, sin embargo para los movimientos de las otras extremidades el análisis es el mismo.

Entonces, según la Ec. 22 se tiene el margen de estabilidad para cada pata.

**Tabla 6**

**Márgenes de Estabilidad Longitudinal para cada movimiento de extremidad en transferencia – Caminata discontinua**

	$x_2$	$y_2$	$x_3$	$y_3$	$S_{LSM}$
<b>Pata 1</b>	$P_x/2$	$-P_y/2$	$-P_x/2 + R_x/2$	$P_y/2$	$R_x/4$
<b>Pata 2</b>	$P_x/2$	$P_y/2$	$-P_x/2 + R_x/2$	$-P_y/2$	$R_x/4$
<b>Pata 3</b>	$-P_x/2$	$-P_y/2$	$P_x/2 - R_x/2$	$P_y/2$	$R_x/4$
<b>Pata 4</b>	$-P_x/2$	$P_y/2$	$P_x/2 - R_x/2$	$-P_y/2$	$R_x/4$

Como se puede ver en la Tabla 6, al reemplazar los valores de los puntos extremos de la diagonal utilizados para el cálculo del  $S_{LSM}$ , se obtiene el mismo valor en los cuatro movimientos de transferencia. Lo cual permite concluir que el robot se mantendrá estable siempre durante el movimiento.

Para el caso del proyecto desarrollado, el margen de estabilidad longitudinal para el movimiento discontinuo tendrá el valor constante señalado en la Ec. 23.

$$\begin{aligned} S_{LSM} &= \frac{R_x}{4} \\ S_{LSM} &= \frac{170mm}{4} \\ S_{LSM} &= 42.5mm \end{aligned} \quad \text{Ec. 23}$$

### 3.1.6.1.2 ANÁLISIS DE CAMINATA DISCONTINUA DE DOS FASES GIRATORIA

El Margen de Estabilidad Longitudinal para caminata discontinua giratoria  $S_{LSM_R}$ , de acuerdo con el desarrollo de la Ec. 22, viene dado por las ecuaciones Ec. 24 (de Santos, García, & Estremera, 2007) y Ec. 25 (de Santos, García, & Estremera, 2007) para ángulos positivo y negativo, respectivamente.

$$S_{LSM_{R+}} = \frac{P_y R_x \cos(\alpha) - (P_x^2 + P_y^2 - P_x R_x) \sin(\alpha)}{4 \cos\left(\frac{\alpha}{2}\right) \left(P_y \cos\left(\frac{\alpha}{2}\right) + P_x \sin\left(\frac{\alpha}{2}\right)\right)} \quad \text{Ec. 24}$$

$$S_{LSM_{R-}} = \frac{P_y R_x \cos(\alpha) - (P_x^2 + P_y^2 - P_x R_x) \sin(\alpha)}{4 \cos\left(\frac{\alpha}{2}\right) \left(P_y \cos\left(\frac{\alpha}{2}\right) + P_x \sin\left(\frac{\alpha}{2}\right)\right)} \quad \text{Ec. 25}$$

### 3.1.7 ANÁLISIS DINÁMICO DEL ROBOT

El análisis dinámico de un robot determina las relaciones entre su movimiento y las fuerzas aplicadas en el mismo. Como resultado de aplicar este análisis se obtienen ecuaciones matemáticas entre las coordenadas articulares con sus derivadas (velocidades y aceleraciones), las fuerzas y pares aplicados; como también los parámetros propios de construcción del robot como son masas, inercias, su geometría, entre otras.

Lagrange-Euler y Newton-Euler son los métodos más utilizados para encontrar el modelo dinámico, pero debido a que el primer método nos brinda ecuaciones diferenciales cerradas, se eligió el mismo para encontrar los torques necesarios para el movimiento de cada articulación sobre todo, en los puntos que requieran mayor energía (Maldonado Del Toro, Silva Ortigoza, Ramos Silvestre, Hernández Guzman, & Rivera Díaz, 2011).

Según la investigación realizada por Maldonado Del Toro et al. (2011), el método de Lagrange-Euler realiza un análisis de energías tanto cinética como potencial para encontrar los momentos aplicados en cada eslabón. Por lo tanto el Lagrangiano se define en la Ec. 26.

$$L(\theta, \dot{\theta}) = K(\theta, \dot{\theta}) - U(\theta) \quad \text{Ec. 26}$$

En donde  $K(\theta, \dot{\theta})$  es la energía cinética y  $U(\theta)$  es la energía potencial de cada eslabón, mientras que  $\theta$  y  $\dot{\theta} = \frac{d}{dt}\theta$  definen la variable articular y su velocidad angular. Teniendo este elemento, se aplican las ecuaciones de Euler-Lagrange que modelan el movimiento de un robot rígido de  $n$  grados de libertad representadas por la Ec. 27.

$$\frac{d}{dt} \left( \frac{\partial L(\theta, \dot{\theta})}{\partial \dot{\theta}} \right) - \frac{\partial L(\theta, \dot{\theta})}{\partial \theta} = \tau \quad \text{Ec. 27}$$

Tomando en cuenta que todas las variables tendrán una connotación vectorial ya que el robot analizado tendrá en cada pata 3 grados de libertad por lo tanto:  $\theta(t) = [\theta_1(t), \theta_2(t), \theta_3(t)]^T$  y  $\dot{\theta}(t) = [\dot{\theta}_1(t), \dot{\theta}_2(t), \dot{\theta}_3(t)]^T$ , el resultado que viene a representar el vector de los torques aplicados a cada pata es  $\tau = [\tau_1, \tau_2, \tau_3]^T$  (Maldonado Del Toro et al., 2011).

En la Figura 38 se ubican los parámetros físicos para el análisis de las energías, tanto cinética como potencial dependiendo la ubicación de cada unión con los eslabones.

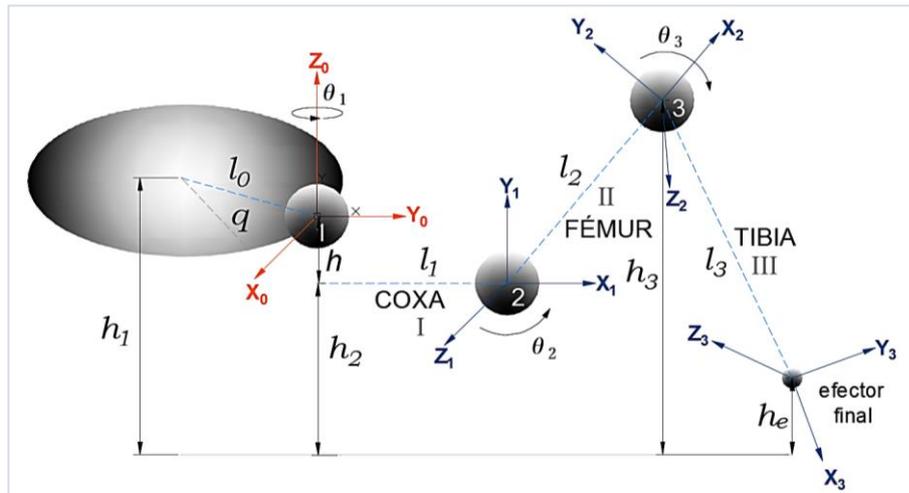


Figura 38. Esquema de una pata para análisis de energías.

Para el análisis se tomará en cuenta que:  $q = 45^\circ$  y  $\theta_1$  adquiere un valor diferente de 90.

Asumiendo que  $S = \text{sen}$  y  $C = \text{cos}$ , se procede a estudiar las energías que posee cada eslabón en un instante de tiempo.

### 3.1.7.1 ANÁLISIS DE ENERGÍA DEL ESLABÓN 1

#### 3.1.7.1.1 ENERGÍA CINÉTICA

La energía cinética del primer eslabón se detalla en la Ec. 28.

$$K_1 = \frac{1}{2} I_1 \dot{\theta}_1^2 \quad \text{Ec. 28}$$

#### 3.1.7.1.2 ENERGÍA POTENCIAL

La energía potencial para el primer eslabón se detalla en la Ec. 29.

$$U_1 = m_1 g h_{CG1} \quad \text{Ec. 29}$$

### 3.1.7.2 ANÁLISIS DE ENERGÍA DEL ESLABÓN 2

#### 3.1.7.2.1 ENERGÍA CINÉTICA

La energía cinética del segundo eslabón se detalla en la Ec. 30.

$$K_2 = \frac{1}{2} m_2 v_2^2 + \frac{1}{2} I_2 (\dot{\theta}_1^2 + \dot{\theta}_2^2) \quad \text{Ec. 30}$$

En donde  $v_2$  se define en función de las velocidades lineales con respecto al centro de gravedad en cada eje. Ver Ec. 31.

$$v_2^2 = \dot{x}_{CG2}^2 + \dot{y}_{CG2}^2 + \dot{z}_{CG2}^2 \quad \text{Ec. 31}$$

Las coordenadas del centro de gravedad del eslabón 2 con respecto al eje de referencia están definidas por las Ec. 32, Ec. 33 y Ec. 34.

$$x_{CG2} = l_0 C 45^\circ + l_1 C(\theta_1 + 45^\circ) + \frac{l_2}{2} C \theta_2 C(\theta_1 + 45^\circ) \quad \text{Ec. 32}$$

$$y_{CG2} = l_0 S 45^\circ + l_1 S(\theta_1 + 45^\circ) + \frac{l_2}{2} C \theta_2 S(\theta_1 + 45^\circ) \quad \text{Ec. 33}$$

$$z_{CG2} = h_2 + \frac{l_2}{2} S \theta_2 \quad \text{Ec. 34}$$

Derivando con respecto al tiempo las ecuaciones: Ec. 32, Ec. 33 y Ec. 34 se tiene las componentes de la velocidad lineal en cada eje. Ver Ec. 35, Ec. 36 y Ec. 37.

$$\dot{x}_{CG2} = -l_1 S(\theta_1 + 45^\circ) \dot{\theta}_1 - \frac{l_2}{2} S \theta_2 C(\theta_1 + 45^\circ) \dot{\theta}_2 - \frac{l_2}{2} C \theta_2 S(\theta_1 + 45^\circ) \dot{\theta}_1 \quad \text{Ec. 35}$$

$$\dot{y}_{CG2} = l_1 C(\theta_1 + 45^\circ) \dot{\theta}_1 - \frac{l_2}{2} S \theta_2 S(\theta_1 + 45^\circ) \dot{\theta}_2 + \frac{l_2}{2} C \theta_2 C(\theta_1 + 45^\circ) \dot{\theta}_1 \quad \text{Ec. 36}$$

$$\dot{z}_{CG2} = \frac{l_2}{2} (C \theta_2) \dot{\theta}_2 \quad \text{Ec. 37}$$

Reemplazando las ecuaciones Ec. 35, Ec. 36 y Ec. 37 en la Ec. 31 se obtiene la velocidad lineal para el eslabón, definida en la Ec. 38.

$$v_2^2 = \left( l_1^2 + \frac{l_2^2}{4} C^2 \theta_2 + 2l_1 l_2 C \theta_2 \right) \dot{\theta}_1^2 + \frac{l_2^2}{4} \dot{\theta}_2^2 \quad \text{Ec. 38}$$

Sustituyendo la Ec. 38 en la Ec. 30 se obtiene la energía cinética resultante mostrada en la Ec. 39.

$$K_2 = \frac{1}{2} m_2 \left[ \left( l_1^2 + \frac{l_2^2}{4} C^2 \theta_2 + 2l_1 l_2 C \theta_2 \right) \dot{\theta}_1^2 + \frac{l_2^2}{4} \dot{\theta}_2^2 \right] + \frac{1}{2} I_2 (\dot{\theta}_1^2 + \dot{\theta}_2^2) \quad \text{Ec. 39}$$

### 3.1.7.2.2 ENERGÍA POTENCIAL

La energía potencial del segundo eslabón se detalla en la Ec. 40.

$$U_2 = m_2 g \left( h_2 + \frac{l_2}{2} S \theta_2 \right) \quad \text{Ec. 40}$$

### 3.1.7.3 ANÁLISIS DE ENERGÍA DEL ESLABÓN 3

#### 3.1.7.3.1 ENERGÍA CINÉTICA

La energía cinética del tercer eslabón se detalla en la Ec. 41.

$$K_3 = \frac{1}{2} m_3 v_3^2 + \frac{1}{2} I_3 (\dot{\theta}_1^2 + (\dot{\theta}_2 - \dot{\theta}_3)^2) \quad \text{Ec. 41}$$

El valor de  $v_3^2$  se obtiene de acuerdo a la Ec. 42.

$$v_3^2 = \dot{x}_{CG3}^2 + \dot{y}_{CG3}^2 + \dot{z}_{CG3}^2 \quad \text{Ec. 42}$$

Las coordenadas del centro de gravedad del eslabón 3 con respecto al eje de referencia están definidas por las ecuaciones Ec. 43, Ec. 44, Ec. 45.

$$x_{CG3} = l_0 C 45^\circ + l_1 C(\theta_1 + 45^\circ) + l_2 C \theta_2 C(\theta_1 + 45^\circ) + \frac{l_3}{2} C(\theta_3 - \theta_2) C(\theta_1 + 45^\circ) \quad \text{Ec. 43}$$

$$y_{CG3} = l_0 S 45^\circ + l_1 S(\theta_1 + 45^\circ) + l_2 C \theta_2 S(\theta_1 + 45^\circ) + \frac{l_3}{2} C(\theta_3 - \theta_2) S(\theta_1 + 45^\circ) \quad \text{Ec. 44}$$

$$z_{CG3} = h_2 + l_2 S \theta_2 - \frac{l_3}{2} S(\theta_3 - \theta_2) \quad \text{Ec. 45}$$

Derivando las ecuaciones Ec. 43, Ec. 44 y Ec. 45, se tiene las componentes de la velocidad lineal en cada eje. Ver Ec. 46, Ec. 47 y Ec. 48.

$$\begin{aligned} \dot{x}_{CG3} = & -l_1 S(\theta_1 + 45^\circ) \dot{\theta}_1 - l_2 S \theta_2 C(\theta_1 + 45^\circ) \dot{\theta}_2 - l_2 C \theta_2 S(\theta_1 + 45^\circ) \dot{\theta}_1 \\ & - \frac{l_3}{2} S(\theta_3 - \theta_2) C(\theta_1 + 45^\circ) (\dot{\theta}_3 - \dot{\theta}_2) - \frac{l_3}{2} C(\theta_3 \\ & - \theta_2) S(\theta_1 + 45^\circ) \dot{\theta}_1 \end{aligned} \quad \text{Ec. 46}$$

$$\begin{aligned} \dot{y}_{CG3} = & l_1 C(\theta_1 + 45^\circ) \dot{\theta}_1 - l_2 S \theta_2 S(\theta_1 + 45^\circ) \dot{\theta}_2 + l_2 C \theta_2 C(\theta_1 + 45^\circ) \dot{\theta}_1 \\ & - \frac{l_3}{2} S(\theta_3 - \theta_2) S(\theta_1 + 45^\circ) (\dot{\theta}_3 - \dot{\theta}_2) + \frac{l_3}{2} C(\theta_3 \\ & - \theta_2) C(\theta_1 + 45^\circ) \dot{\theta}_1 \end{aligned} \quad \text{Ec. 47}$$

$$\dot{z}_{CG3} = l_2 C \theta_2 \dot{\theta}_2 - \frac{l_3}{2} C(\theta_3 - \theta_2) (\dot{\theta}_3 - \dot{\theta}_2) \quad \text{Ec. 48}$$

Reemplazando las ecuaciones Ec. 46, Ec. 47 y Ec. 48 en Ec. 42 se define la velocidad lineal del eslabón 3 en la Ec. 49.

$$\begin{aligned} v_3^2 = & l_1^2 \dot{\theta}_1^2 + 2l_1 l_2 (C \theta_2) \dot{\theta}_1^2 + l_1 l_3 C(\theta_3 - \theta_2) \dot{\theta}_1^2 + l_2^2 \dot{\theta}_2^2 - l_2 l_3 C \theta_3 \dot{\theta}_2 (\dot{\theta}_3 - \dot{\theta}_2) \\ & + l_2^2 (C^2 \theta_2) \dot{\theta}_1^2 + \frac{l_3^2}{2} S^2(\theta_3 - \theta_2) (\dot{\theta}_3 - \dot{\theta}_2)^2 \\ & + l_2 l_3 (C \theta_2) C(\theta_3 - \theta_2) \dot{\theta}_1^2 + \frac{l_3^2}{4} C^2(\theta_3 - \theta_2) \dot{\theta}_1^2 \end{aligned} \quad \text{Ec. 49}$$

La energía cinética resultante se obtiene reemplazando la Ec. 49 en la Ec. 41. Ver Ec. 50.

$$\begin{aligned} K_3 = & \frac{1}{2} m_3 \left[ l_1^2 \dot{\theta}_1^2 + 2l_1 l_2 (C \theta_2) \dot{\theta}_1^2 + l_1 l_3 C(\theta_3 - \theta_2) \dot{\theta}_1^2 + l_2^2 \dot{\theta}_2^2 \right. \\ & + l_2 l_3 C \theta_3 \dot{\theta}_2 (\dot{\theta}_3 - \dot{\theta}_2) + l_2^2 (C^2 \theta_2) \dot{\theta}_1^2 \\ & + \frac{l_3^2}{2} S^2(\theta_3 - \theta_2) (\dot{\theta}_3 - \dot{\theta}_2)^2 + l_2 l_3 (C \theta_2) C(\theta_3 - \theta_2) \dot{\theta}_1^2 \\ & \left. + \frac{l_3^2}{4} C^2(\theta_3 - \theta_2) \dot{\theta}_1^2 \right] + \frac{1}{2} I_3 (\dot{\theta}_1^2 + (\dot{\theta}_2 - \dot{\theta}_3)^2) \end{aligned} \quad \text{Ec. 50}$$

### 3.1.7.3.2 ENERGÍA POTENCIAL

La energía potencial del tercer eslabón se detalla en la Ec. 51.

$$U_3 = m_3 g \left( h_2 + l_2 S \theta_2 - \frac{l_3}{2} S(\theta_3 - \theta_2) \right) \quad \text{Ec. 51}$$

### 3.1.7.4 ANÁLISIS DE ENERGÍA DEL EFECTOR FINAL

#### 3.1.7.4.1 ENERGÍA CINÉTICA

La energía cinética del efector final se detalla en la Ec. 52.

$$K_4 = \frac{1}{2} m_4 v_4^2 \quad \text{Ec. 52}$$

En dónde  $v_4^2$  está definido por la Ec. 53.

$$v_4^2 = \dot{x}_{CG4}^2 + \dot{y}_{CG4}^2 + \dot{z}_{CG4}^2 \quad \text{Ec. 53}$$

Las coordenadas del centro de gravedad del eslabón 3 con respecto a cada eje de referencia están definidas por las Ec. 54, Ec. 55 y Ec. 56.

$$x_{CG4} = l_0 C 45^\circ + l_1 C(\theta_1 + 45^\circ) + l_2 C \theta_2 C(\theta_1 + 45^\circ) + l_3 C(\theta_3 - \theta_2) C(\theta_1 + 45^\circ) \quad \text{Ec. 54}$$

$$y_{CG4} = l_0 S 45^\circ + l_1 S(\theta_1 + 45^\circ) + l_2 C \theta_2 S(\theta_1 + 45^\circ) + l_3 C(\theta_3 - \theta_2) S(\theta_1 + 45^\circ) \quad \text{Ec. 55}$$

$$z_{CG4} = h_2 + l_2 S \theta_2 - l_3 C(\theta_3 - \theta_2) \quad \text{Ec. 56}$$

Derivando con respecto al tiempo las ecuaciones Ec. 54, Ec. 55 y Ec. 56 se tiene las ecuaciones Ec. 57, Ec. 58 y Ec. 59.

$$\dot{x}_{CG4} = -l_1 S(\theta_1 + 45^\circ) \dot{\theta}_1 - l_2 S \theta_2 C(\theta_1 + 45^\circ) \dot{\theta}_2 - l_2 C \theta_2 S(\theta_1 + 45^\circ) \dot{\theta}_1 - l_3 S(\theta_3 - \theta_2) C(\theta_1 + 45^\circ) (\dot{\theta}_3 - \dot{\theta}_2) - l_3 C(\theta_3 - \theta_2) S(\theta_1 + 45^\circ) \dot{\theta}_1 \quad \text{Ec. 57}$$

$$\begin{aligned} \dot{y}_{CG4} = & l_1 C(\theta_1 + 45^\circ) \dot{\theta}_1 - l_2 S \theta_2 S(\theta_1 + 45^\circ) \dot{\theta}_2 + l_2 C \theta_2 C(\theta_1 + 45^\circ) \dot{\theta}_1 \\ & - l_3 S(\theta_3 - \theta_2) S(\theta_1 + 45^\circ) (\dot{\theta}_3 - \dot{\theta}_2) + l_3 C(\theta_3 \\ & - \theta_2) C(\theta_1 + 45^\circ) \dot{\theta}_1 \end{aligned} \quad \text{Ec. 58}$$

$$\dot{z}_{CG4} = l_2 C \theta_2 \dot{\theta}_2 + l_3 S(\theta_3 - \theta_2) (\dot{\theta}_3 - \dot{\theta}_2) \quad \text{Ec. 59}$$

Reemplazando las ecuaciones Ec. 57, Ec. 58 y Ec. 59 en la Ec. 53 se obtiene la velocidad lineal del efector final mostrada en la Ec. 60.

$$\begin{aligned} v_4^2 = & l_1 \dot{\theta}_1^2 + 2l_1 l_2 (C \theta_2) \dot{\theta}_1^2 + 2l_1 l_3 C(\theta_3 - \theta_2) \dot{\theta}_1^2 + l_2^2 \dot{\theta}_2^2 + 2l_2 l_3 C \theta_3 \dot{\theta}_2 (\dot{\theta}_3 - \dot{\theta}_2) \\ & + l_2^2 (C^2 \theta_2) \dot{\theta}_1^2 + 2l_3^2 S^2(\theta_3 - \theta_2) (\dot{\theta}_3 - \dot{\theta}_2)^2 \\ & + 2l_2 l_3 (C \theta_2) C(\theta_3 - \theta_2) \dot{\theta}_1^2 + l_3^2 C^2(\theta_3 - \theta_2) \dot{\theta}_1^2 \end{aligned} \quad \text{Ec. 60}$$

La energía cinética resultante se define reemplazando la Ec. 60 en la Ec. 52.

$$\begin{aligned} K_4 = & \frac{1}{2} m_4 \left[ l_1 \dot{\theta}_1^2 + 2l_1 l_2 (C \theta_2) \dot{\theta}_1^2 + 2l_1 l_3 C(\theta_3 - \theta_2) \dot{\theta}_1^2 + l_2^2 \dot{\theta}_2^2 \right. \\ & + 2l_2 l_3 C \theta_3 \dot{\theta}_2 (\dot{\theta}_3 - \dot{\theta}_2) + l_2^2 (C^2 \theta_2) \dot{\theta}_1^2 \\ & + 2l_3^2 S^2(\theta_3 - \theta_2) (\dot{\theta}_3 - \dot{\theta}_2)^2 + 2l_2 l_3 (C \theta_2) C(\theta_3 - \theta_2) \dot{\theta}_1^2 \\ & \left. + l_3^2 C^2(\theta_3 - \theta_2) \dot{\theta}_1^2 \right] \end{aligned} \quad \text{Ec. 61}$$

### 3.1.7.4.2 ENERGÍA POTENCIAL

La energía potencial del efector final se detalla en la Ec. 62.

$$U_4 = m_4 g (h_2 + l_2 S \theta_2 - l_3 S(\theta_3 - \theta_2)) \quad \text{Ec. 62}$$

### 3.1.7.5 ANÁLISIS DE ENERGÍA TOTAL

Para encontrar el Lagrangiano se debe obtener las energías cinética y potencial totales del sistema, para ello se suman las energías parciales de cada eslabón obtenidas anteriormente (Maldonado et al., 2011).

#### 3.1.7.5.1 ENERGÍA CINÉTICA

Se obtiene la energía cinética total en la Ec. 63 sumando las ecuaciones: Ec. 28, Ec. 39, Ec. 50 y Ec. 61.

$$\begin{aligned}
K_T = & \frac{1}{2}I_1\dot{\theta}_1^2 + \frac{1}{2}m_2 \left[ \left( l_1 + \frac{l_2^2}{4}C^2\theta_2 + l_1l_2C\theta_2 \right) \dot{\theta}_1^2 + \frac{l_2^2}{4}\dot{\theta}_2^2 \right] \\
& + \frac{1}{2}I_2 \left( \dot{\theta}_1^2 + \dot{\theta}_2^2 \right) \\
& + \frac{1}{2}m_3 \left[ l_1^2\dot{\theta}_1^2 + 2l_1l_2(C\theta_2)\dot{\theta}_1^2 + l_1l_3C(\theta_3 - \theta_2)\dot{\theta}_1^2 \right. \\
& + l_2^2\dot{\theta}_2^2 + l_2l_3C\theta_3\dot{\theta}_2(\dot{\theta}_3 - \dot{\theta}_2) + l_2^2(C^2\theta_2)\dot{\theta}_1^2 \\
& + \left. \frac{l_3^2}{2}S^2(\theta_3 - \theta_2)(\dot{\theta}_3 - \dot{\theta}_2)^2 + l_2l_3(C\theta_2)C(\theta_3 - \theta_2)\dot{\theta}_1^2 \right. \\
& + \left. \frac{l_3^2}{4}C^2(\theta_3 - \theta_2)\dot{\theta}_1^2 \right] + \frac{1}{2}I_3 \left( \dot{\theta}_1^2 + (\dot{\theta}_2 + \dot{\theta}_3)^2 \right) \\
& + \frac{1}{2}m_4 \left[ l_1\dot{\theta}_1^2 + 2l_1l_2(C\theta_2)\dot{\theta}_1^2 + 2l_1l_3C(\theta_3 - \theta_2)\dot{\theta}_1^2 \right. \\
& + l_2^2\dot{\theta}_2^2 + 2l_2l_3C\theta_3\dot{\theta}_2(\dot{\theta}_3 - \dot{\theta}_2) + l_2^2(C^2\theta_2)\dot{\theta}_1^2 \\
& + 2l_3^2S^2(\theta_3 - \theta_2)(\dot{\theta}_3 - \dot{\theta}_2)^2 + 2l_2l_3(C\theta_2)C(\theta_3 - \theta_2)\dot{\theta}_1^2 \\
& + \left. l_3^2C^2(\theta_3 - \theta_2)\dot{\theta}_1^2 \right]
\end{aligned} \tag{Ec. 63}$$

### 3.1.7.5.2 ENERGÍA POTENCIAL

Se obtiene la energía potencial total en la Ec. 64 sumando las ecuaciones: Ec. 29, Ec. 40, Ec. 51 y Ec. 62.

$$\begin{aligned}
U_T = & m_1gh_{CG1} + m_2g \left( h_2 + \frac{l_2}{2}S\theta_2 \right) \\
& + m_3g \left( h_2 + l_2S\theta_2 - \frac{l_3}{2}S(\theta_3 - \theta_2) \right) \\
& + m_4g \left( h_2 + l_2S\theta_2 - l_3S(\theta_3 - \theta_2) \right)
\end{aligned} \tag{Ec. 64}$$

### 3.1.7.6 LAGRANGIANO

Se define el Lagrangiano en la Ec. 65, reemplazando las ecuaciones Ec. 63 y Ec. 64 en Ec. 26.

$$\begin{aligned}
L = & \frac{1}{2}I_1\dot{\theta}_1^2 + \frac{1}{2}m_2 \left[ \left( \left( l_1 + \frac{l_2}{2}C\theta_2 \right)^2 + l_1l_2C\theta_2 \right) \dot{\theta}_1^2 + \frac{l_2^2}{4}\dot{\theta}_2^2 \right] + \frac{1}{2}I_2 (\dot{\theta}_1^2 + \dot{\theta}_2^2) \\
& + \frac{1}{2}m_3 \left[ l_1^2\dot{\theta}_1^2 + 2l_1l_2(C\theta_2)\dot{\theta}_1^2 + l_1l_3C(\theta_3 - \theta_2)\dot{\theta}_1^2 + l_2^2\dot{\theta}_2^2 \right. \\
& + l_2l_3C\theta_3\dot{\theta}_2(\dot{\theta}_3 - \dot{\theta}_2) + l_2^2(C^2\theta_2)\dot{\theta}_1^2 \\
& + \frac{l_3^2}{2}S^2(\theta_3 - \theta_2)(\dot{\theta}_3 - \dot{\theta}_2)^2 + l_2l_3(C\theta_2)C(\theta_3 - \theta_2)\dot{\theta}_1^2 \\
& \left. + \frac{l_3^2}{4}C^2(\theta_3 - \theta_2)\dot{\theta}_1^2 \right] + \frac{1}{2}I_3 (\dot{\theta}_1^2 + (\dot{\theta}_2 + \dot{\theta}_3)^2) \\
& + \frac{1}{2}m_4 \left[ l_1^2\dot{\theta}_1^2 + 2l_1l_2(C\theta_2)\dot{\theta}_1^2 + 2l_1l_3C(\theta_3 - \theta_2)\dot{\theta}_1^2 + l_2^2\dot{\theta}_2^2 \right. \\
& + 2l_2l_3C\theta_3\dot{\theta}_2(\dot{\theta}_3 - \dot{\theta}_2) + l_2^2(C^2\theta_2)\dot{\theta}_1^2 \\
& + 2l_3^2S^2(\theta_3 - \theta_2)(\dot{\theta}_3 - \dot{\theta}_2)^2 + 2l_2l_3(C\theta_2)C(\theta_3 - \theta_2)\dot{\theta}_1^2 \\
& \left. + l_3^2C^2(\theta_3 - \theta_2)\dot{\theta}_1^2 \right] - m_1gh_{CG1} - m_2g \left( h_2 + \frac{l_2}{2}S\theta_2 \right) \\
& - m_3g \left( h_2 + l_2S\theta_2 - \frac{l_3}{2}S(\theta_3 - \theta_2) \right) \\
& - m_4g(h_2 + l_2S\theta_2 - l_3S(\theta_3 - \theta_2))
\end{aligned}$$

Ec. 65

### 3.1.7.7 MODELO DE TORQUES PARA CADA ESLABÓN

#### 3.1.7.7.1 MODELO DEL TORQUE 1

Se encuentra el torque aplicando la Ec. 27 en la Ec. 65, para  $\theta_1$ . Ver Ec. 66.

$$\begin{aligned}
\tau_1 = & \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_1} \right) - \frac{\partial L}{\partial \theta_1} \\
= & \left[ m_2 \left( 1 + \frac{l_2}{2}C\theta_2 \right)^2 + 2l_1l_2C\theta_2 + I_1 + I_2 + (m_3 + m_4)l_1^2 \right. \\
& + (2m_4 + m_3)l_1l_2C\theta_2 + (2m_4 + m_3)C(\theta_3 - \theta_2) \\
& + (m_3 + m_4)l_2^2C\theta_2 + (2m_4 + m_3)l_2l_3C\theta_2C(\theta_3 - \theta_2) \\
& \left. + (m_4 + m_3)l_3^2C^2(\theta_3 - \theta_2) + I_1 \right] \ddot{\theta}_1 \\
& + \left[ -(m_3 + 2m_4)l_1l_3S(\theta_3 - \theta_2) \right. \\
& - (m_3 + 2m_4)l_1l_3C\theta_2S(\theta_3 - \theta_2) \\
& \left. - \left( \frac{m_3}{2} + 2m_4 \right) l_3^2S(2\theta_3 - 2\theta_2) \right] \dot{\theta}_3\dot{\theta}_1 \\
& + \left[ -2m_2l_1l_2S\theta_2 - \frac{m_2l_2^2}{2}C\theta_2 - (2m_3 + 2m_4)l_1l_2S\theta_2 \right. \\
& + (m_3 + 2m_4)l_1l_3S(\theta_3 - \theta_2) - (m_3 + 2m_4)l_2^2S(2\theta_2) \\
& - (m_3 + 2m_4)l_2l_3S(2\theta_2 - \theta_3) \\
& \left. + \left( \frac{m_3}{2} + m_4 \right) l_3^2S(2\theta_3 - 2\theta_2) \right] \dot{\theta}_2\dot{\theta}_1
\end{aligned}$$

Ec. 66

### 3.1.7.7.2 MODELO DEL TORQUE 2

Se encuentra el torque 2 aplicando la Ec. 27 en la Ec. 65. Ver Ec. 67.

$$\begin{aligned}
 \tau_2 = \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_2} \right) - \frac{\partial L}{\partial \theta_2} &= \left[ (m_3 + m_4) l_2 l_3 C \theta_3 - \left( \frac{m_3}{2} + m_4 \right) l_3^2 S^2(\theta_3 - \theta_2) + I_3 \right] \ddot{\theta}_3 \\
 &+ \left[ -\frac{m_2 l_2}{4} + (m_3 + m_4) l_2^2 - (m_3 + 2m_4) l_2 l_3 C \theta_3 \right. \\
 &+ \left. \left( \frac{m_3}{2} + 2m_4 \right) l_3^2 S^2(\theta_3 - \theta_2) + I_2 + I_3 \right] \ddot{\theta}_2 \\
 &+ \left[ -\left( \frac{m_3 + 7m_4}{4} \right) l_3^2 S(2\theta_3 - 2\theta_2) - \left( \frac{m_3}{2} + m_4 \right) l_2 l_3 S \theta_3 \right] \dot{\theta}_3^2 \\
 &+ \left[ (m_3 + 2m_4) l_2 l_3 S \theta_3 + \left( \frac{m_3 + 7m_4}{2} \right) l_3^2 S(2\theta_3 - 2\theta_2) \right] \dot{\theta}_3 \dot{\theta}_2 \\
 &+ \left[ -\left( \frac{m_3 + 7m_4}{4} \right) l_3^2 S(2\theta_3 - 2\theta_2) \right] \dot{\theta}_2^2 \\
 &+ \left[ -(m_2 l_1 l_2 + m_2 l_2^2) S \theta_2 + (m_2 + m_4) l_1 l_2 S \theta_2 \right. \\
 &- \left. \left( \frac{m_3}{2} + m_4 \right) l_1 l_3 S(\theta_3 - \theta_2) + \left( \frac{m_3}{2} + m_4 \right) l_2^2 S(2\theta_2) \right. \\
 &+ \left. \left( \frac{m_3 + m_4}{2} \right) l_2 l_3 S(2\theta_2 - \theta_3) \right. \\
 &- \left. \left( \frac{m_3 + 4m_4}{8} \right) l_3^2 S(2\theta_3 - 2\theta_2) \right] \dot{\theta}_1^2 \\
 &+ \left[ \frac{m_2 g l_2}{2} C \theta_2 + m_3 g \left( l_2 C \theta_2 - \frac{l_3}{2} C(\theta_3 - \theta_2) \right) \right. \\
 &\left. + m_4 g (l_2 C \theta_2 - l_3 C(\theta_3 - \theta_2)) \right]
 \end{aligned}$$

Ec. 67

### 3.1.7.7.3 MODELO DEL TORQUE 3

Se encuentra el torque 3 aplicando la Ec. 27 en la Ec. 65. Ver Ec. 68.

$$\begin{aligned}
 \tau_3 = \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_3} \right) - \frac{\partial L}{\partial \theta_3} &= \left[ \left( \frac{m_3}{2} + 2m_4 \right) l_3^2 S^2(\theta_3 - \theta_2) + I_3 \right] \ddot{\theta}_3 \\
 &+ \left[ -\left( \frac{m_3}{2} + 2m_4 \right) l_3^2 S^2(\theta_3 - \theta_2) + \left( \frac{m_3}{2} + m_4 \right) l_2 l_3 C \theta_3 + I_3 \right] \ddot{\theta}_2 \\
 &+ \left[ \left( \frac{m_3}{2} + 2m_4 \right) l_3^2 S(\theta_3 - \theta_2) C(\theta_3 - \theta_2) \right] \dot{\theta}_3^2 \\
 &- \left[ (m_3 + 4m_4) l_3^2 S(\theta_3 - \theta_2) C(\theta_3 - \theta_2) \right] \dot{\theta}_3 \dot{\theta}_2 \\
 &+ \left[ \left( \frac{m_3}{2} + m_4 \right) l_2 l_3 S \theta_3 \right. \\
 &- \left. \left( \frac{m_3}{2} + 2m_4 \right) l_3^2 S(\theta_3 - \theta_2) C(\theta_3 - \theta_2) \right] \dot{\theta}_2^2 \\
 &+ \left[ \left( \frac{m_3}{2} + m_4 \right) l_1 l_3 S(\theta_3 - \theta_2) + \left( \frac{m_3}{2} + m_4 \right) l_2 l_3 C \theta_2 S(\theta_3 - \theta_2) \right. \\
 &+ \left. \left( \frac{m_3}{2} + m_4 \right) l_3^2 C(\theta_3 - \theta_2) S(\theta_3 - \theta_2) \right] \dot{\theta}_1^2 \\
 &+ \left[ \frac{m_3 g l_3}{2} C(\theta_3 - \theta_2) + m_4 g l_3 C(\theta_3 - \theta_2) \right]
 \end{aligned}$$

Ec. 68

### 3.1.7.8 DEFINICIÓN NUMÉRICA DE LOS TORQUES DE CADA ESLABÓN

Para poder encontrar los valores numéricos de cada torque es necesario analizar el modelo obtenido en un instante de tiempo en donde se posea toda la información necesaria. El instante analizado será cuando la pata este en el punto más alto sobre el nivel del suelo, ya que se tendrá tanto el máximo de energía potencial como cinética.

El punto analizado será:  $x = 135,64 \text{ mm}$ ,  $y = 168,4 \text{ mm}$ ,  $z = 60 \text{ mm}$ ; en donde los datos obtenidos se los puede observar en la Tabla 7.

**Tabla 7**

**Características de los eslabones del robot**

Elemento	Valor	Unidad
$m_1$	0.16	<i>kg</i>
$I_1$	0.0482	<i>kg/cm<sup>2</sup></i>
$l_1$	4.38	<i>cm</i>
$\theta_1$	0.22	<i>rad</i>
$\dot{\theta}_1$	6.16	<i>rad/s</i>
$\ddot{\theta}_1$	0	<i>rad/s<sup>2</sup></i>
$m_2$	0.05	<i>kg</i>
$I_2$	0.0004	<i>kg/cm<sup>2</sup></i>
$l_2$	11	<i>cm</i>
$\theta_2$	-1.33	<i>rad</i>
$\dot{\theta}_2$	6.16	<i>rad/s</i>
$\ddot{\theta}_2$	0	<i>rad/s<sup>2</sup></i>
$m_3$	0.09	<i>kg</i>
$I_3$	0.1595	<i>kg/cm<sup>2</sup></i>
$l_3$	20.2	<i>cm</i>
$\theta_3$	2.71	<i>rad</i>
$\dot{\theta}_3$	6.16	<i>rad/s</i>
$\ddot{\theta}_3$	0	<i>rad/s<sup>2</sup></i>
$m_4$	0.005	<i>kg</i>
$g$	981	<i>cm/s<sup>2</sup></i>

Reemplazando los datos de la Tabla 7 en las ecuaciones: Ec. 66, Ec. 67 y Ec. 68; se define los valores de los torques máximos de los actuadores de la extremidad del robot al momento de cumplir con un movimiento de fase de la pata. Ver Ec. 69.

$$\tau_1 = 0,46 \text{ kg.cm}$$

$$\tau_2 = 13.40 \text{ kg.cm}$$

Ec. 69

$$\tau_3 = 2,79 \text{ kg.cm}$$

## 3.2 COMPONENTES MECÁNICOS

### 3.2.1 ACTUADORES

El robot zoomórfico poseerá doce actuadores, tres por cada extremidad que permitirán el movimiento de los eslabones.

Debido a que los torques necesarios para este prototipo son pequeños, se optó por el uso de servomotores. Los mismos que por su tamaño, versatilidad y fácil control hacen de este motor el ideal para este tipo de aplicaciones.

#### 3.2.1.1 SERVO MOTORES

### DIMENSIONAMIENTO

De acuerdo a los torques de la Ec. 69, obtenidos mediante el análisis dinámico de las extremidades de robot, se procede a realizar el dimensionamiento de los servo motores.

Aplicando un factor de seguridad de 1.5 a los torques de la Ec. 69, debido a características no tomadas en cuenta en el modelamiento como: factores medio ambientales, la fuerza del viento, fuerzas de rozamiento del lugar en el que se opere el robot, etc.; se definen los torques necesarios en la Ec. 70.

$$\tau_1 = 0,46(1,5) = 0,7 \text{ kg.cm}$$

$$\tau_2 = 13.40(1,5) = 20.1 \text{ kg.cm}$$

Ec. 70

$$\tau_3 = 2,79(1,5) = 4,19 \text{ kg.cm}$$

En base al torque más alto de la Ec. 70 se eligió el servo motor TowerPro MG958, con las características señaladas en la Tabla 8.

Tabla 8

Características del servo motor elegido para el segundo eslabón

Fuente: (Torq Pro & Tower Pro, 2015)

Marca	TowerPro
<b>Serie</b>	MG958
<b>Peso</b>	55g
<b>Dimensiones</b>	40.7 × 19.7 × 42.9 mm
<b>Velocidad rotacional</b>	(4.8v) 0.18 s / 60° (6.0v) 0.15 s / 60°
<b>Voltaje de operación</b>	4.8 - 6.6 V
<b>Corriente de operación</b>	500 – 900 mA
<b>Torque</b>	(4.8v) 18 kg.cm (6.6v) 20 kg.cm
<b>Rango de temperatura</b>	0 - 55°C
<b>Banda muerta</b>	1 us
<b>Longitud del cable</b>	32 cm
<b>Engranajes y ejes</b>	Aluminio 6061-T6

El servo motor se seleccionó debido a los siguientes puntos:

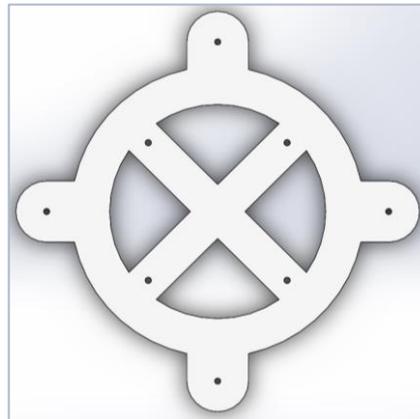
- Posee el torque necesario para cada extremidad.
- El peso pequeño del motor permite disminuir el peso total del robot.
- Las dimensiones pequeñas se ajustan a las medidas escogidas en el modelamiento robótico.
- Posee engranes de metal en su caja reductora, que a la vez que le permite al robot cumplir con el torque deseado, le brinda al servomotor más confianza y tiempo de duración.

### 3.2.2 ESTRUCTURA

La estructura del robot se conforma del cuerpo y las extremidades de acuerdo a las longitudes determinadas en la Tabla 4. En esta sección se examinará las partes que comprende el cuadrúpedo, las funciones que desempeñan en su estructura y las características físicas principales.

### 3.2.2.1 ESTRUCTURA DEL CUERPO

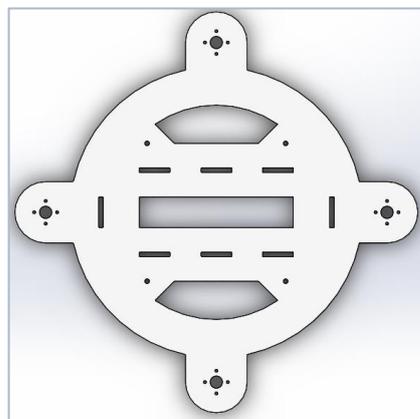
El cuerpo se diseñó de tal forma que soporte en su interior las placas de control y de potencia con todos sus componentes. Éste consta de dos piezas planas sujetadas mediante las uniones de los cuatro servos que van a conformar el primer grado de libertad de cada extremidad. La pieza inferior del cuerpo del robot se puede observar en la Figura 39.



**Figura 39. Placa inferior del cuerpo del robot**

Las cuatro perforaciones ubicadas longitudinal y transversalmente en las dos piezas, sirven para el soporte de las placas de control y potencia mediante postes, colocados también en el diseño de las PCB.

En la parte superior de la primera placa se va a colocar la batería, por lo que va a tener las perforaciones para este propósito. Ver Figura 40.



**Figura 40. Placa superior del cuerpo del robot**

Las perforaciones ubicadas en la parte externa del cuerpo observadas en Figura 39 y Figura 40 se establecieron de acuerdo a los parámetros físicos de los servomotores y su función es la sujeción de los mismos.

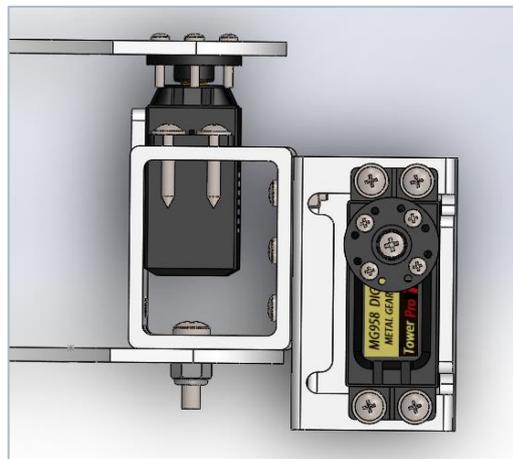
### 3.2.2.2 ESTRUCTURA DE LAS EXTREMIDADES

Al elegirse la estructura de insecto para el diseño de las extremidades del robot, tal como se representa en la Figura 21; las partes de las que va a constar, como se mencionó en el modelamiento robótico son:

- Coxa
- Fémur
- Tibia

#### 3.2.2.2.1 COXA

La coxa está formada de dos piezas que ayudan a la sujeción de los servos referentes al primer y al segundo grado de libertad. Ver Figura 41.



**Figura 41. Estructura de la coxa del cuadrúpedo**

Los *brackets* utilizados fueron diseñados de acuerdo a las dimensiones de los servomotores elegidos. Las longitudes del primer *bracket* son importantes debido a que su altura va a dimensionar también la altura del cuerpo en la que se colocará los componentes electrónicos del proyecto.

### 3.2.2.2.2 FÉMUR

El eslabón correspondiente al fémur del cuadrúpedo se conforma de dos piezas que sirven como sujeción para los servos que pertenecen a la primera y segunda unión. Ver Figura 42 y Figura 43.

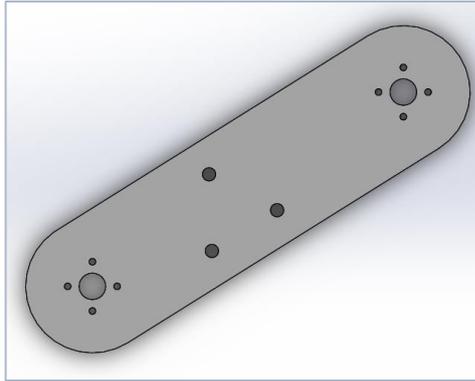


Figura 42. Bracket servo fémur-1

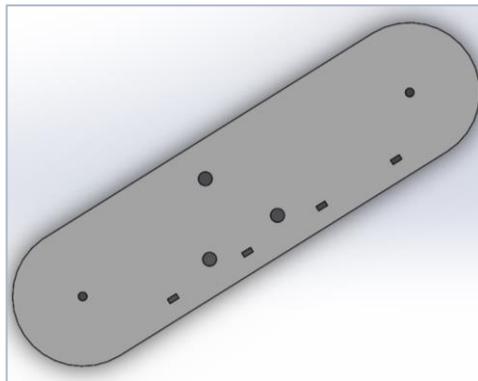
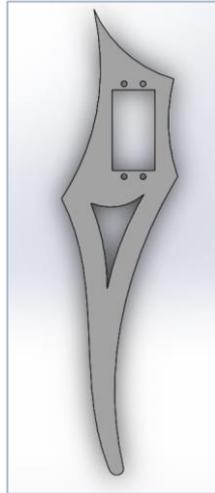


Figura 43. Bracket servo fémur-2

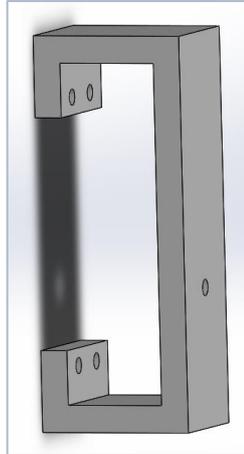
### 3.2.2.2.3 TIBIA

La tibia se conforma por la pieza indicada en la Figura 44, la misma servirá como bracket del servo correspondiente al tercer grado de libertad.



**Figura 44. Bracket servo tibia**

La tibia también se conformará de otra pieza para complementar su estructura. Ver Figura 45.



**Figura 45. Pieza de soporte servo tibia**

La estructura final de la tibia se puede observar en la Figura 46.

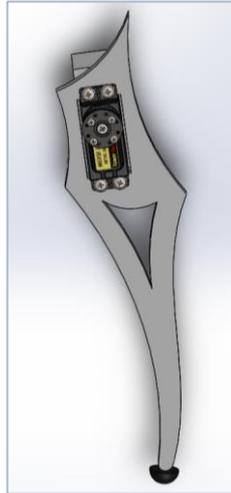


Figura 46. Estructura de la tibia del robot

### 3.2.3 ANÁLISIS MECÁNICO DEL ROBOT

Para elegir el material se establecieron ponderaciones de acuerdo sus características, así fueron calificados de 1 a 5, siendo 1 nada favorable y 5 muy favorable observado en la Tabla 9.

Tabla 9. Elección del material

Materiales	Pon.	Al	%	ABS	%	Acrílico	%
Resistencia	25%	5	1.25	2	0.5	3	0.75
Elasticidad	25%	5	1.25	2	0.5	3	0.75
Costo	20%	2	0.4	3	0.6	4	0.8
F. Manufactura	15%	2	0.3	3	0.45	4	0.6
Disponibilidad	15%	1	0.15	3	0.45	5	0.75
TOTAL			3.35		2.5		3.65

Finalmente el material que tiene mayor calificación es el acrílico debido a su bajo costo, facilidad de maquinabilidad, disponibilidad y una resistencia última adecuada. El mismo se implementó para la mayoría de partes del robot y el ABS fue necesario para elaborar piezas complejas mediante una impresora 3D.

### 3.2.3.1 ANÁLISIS DE FUERZAS ESTÁTICO DEL ROBOT

Para el análisis estático del robot es necesario calcular las reacciones normales en cada extremidad generadas por el peso del cuadrúpedo. Ver Figura 47.

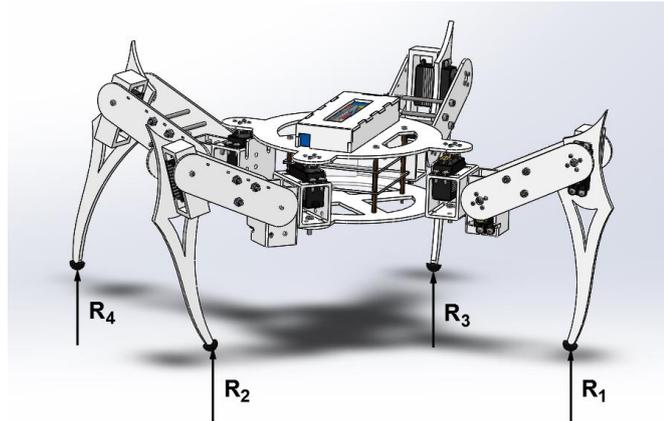


Figura 47. Reacciones normales de cada pata

Para el estudio se debe elegir una postura en la que el robot esté en reposo, para ello se seleccionó la postura inicial; la cual se efectuará dos veces en el movimiento, una por cada fase, como se puede observar en la Figura 34, la que detalla la secuencia del movimiento. La postura es la indicada en la Figura 48.

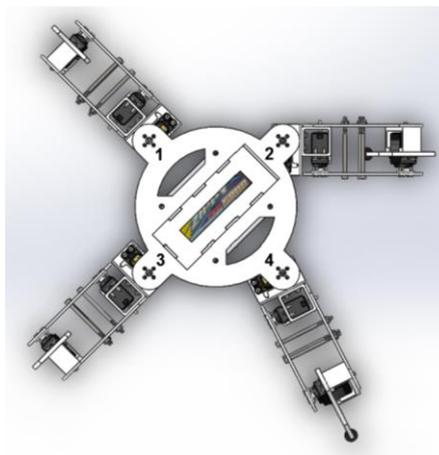
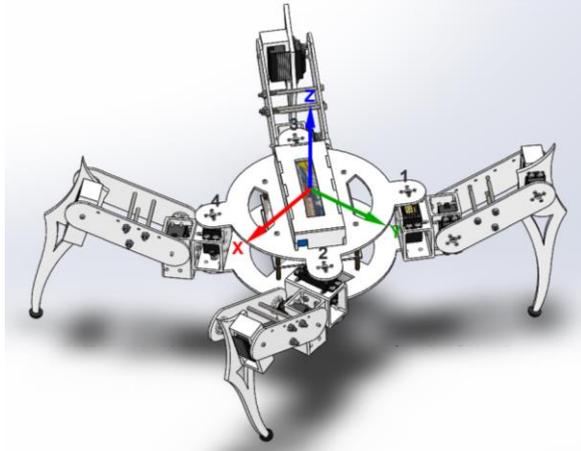


Figura 48. Posición inicial del robot

El sistema de referencia elegido para el análisis se puede observar en la Figura 49.



**Figura 49. Sistema de referencia del robot**

Debido a que las patas 1 y 3 tienen la misma disposición estructural en esta posición, se considera mediante el método de área cooperante que las reacciones normales son iguales para ambas. Teniendo la Ec. 71

$$R_1 = R_3 = R \quad \text{Ec. 71}$$

Realizando el equilibrio de fuerzas en el eje z se tiene la Ec. 72.

$$\sum F_z = 0$$

$$R_4 + R_2 + 2R = W_b + 4(W_1 + W_2 + W_3)$$

$$R_2 + R_4 + 2R = 28.72 \quad \text{Ec. 72}$$

Efectuando el equilibrio de momentos alrededor del eje X y el eje Y en el centro de gravedad del robot, se desarrolló el diagrama de cuerpo libre para los planos YZ y XZ (ver Ec. 48 y Ec. 49). Los pesos y centros de masa de cada eslabón observados se detallan en la Tabla 10 y Tabla 11.

Tabla 10

Parámetros de peso y espacio de trabajo del robot

Parámetro	
$W_b$	25.73 N
$W_1$	1.55 N
$W_2$	0.45 N
$W_3$	0.99 N
$Px$	0.326 m
$Rx$	0.17 m

Tabla 11

Centro de masa de cada eslabón en cada extremidad

Pata	Eslabón	X (m)	Y (m)
1	1	0.0923	0.0954
	2	0.1377	0.1509
	3	0.1658	0.1852
2	1	0.1003	0.0754
	2	0.1756	0.0649
	3	0.2238	0.0582
3	1	0.0959	0.0917
	2	0.1529	0.1352
	3	0.1889	0.1627
4	1	0.0869	0.0986
	2	0.1154	0.1161
	3	0.1399	0.2222

Realizando el equilibrio de momentos alrededor del eje  $Y$  en el centro de gravedad del robot, partiendo de la vista del plano  $XZ$  (ver Figura 50) que muestra el diagrama de cuerpo libre de los pesos y reacciones que actúan, se obtiene la Ec. 73.

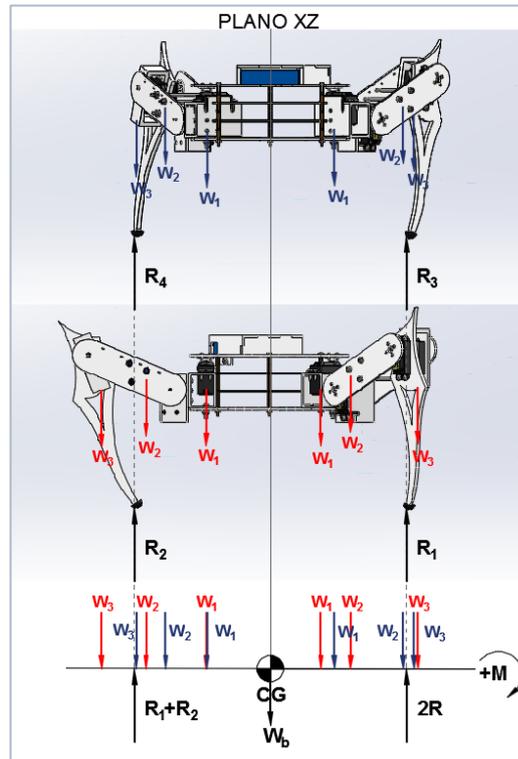


Figura 50. Diagrama de cuerpo libre del robot (Plano XZ)

$$\sum M_y = 0$$

$$W_1(x_{11} + x_{31}) + W_2(x_{12} + x_{32}) + W_3(x_{13} + x_{33}) + (R_4 + R_2)(P_x/2) \\ = 2R(P_x/2) + W_1(x_{21} + x_{41}) + W_2(x_{22} + x_{42}) + W_3(x_{23} + x_{43})$$

$$0.77 + (R_4 + R_2) * 0.163 = 0.78 + 2R * 0.163$$

$$0.163R_2 + 0.163R_4 - 0.326R = 0.01$$

Ec. 73

Realizando el equilibrio de momentos alrededor del eje X, en el centro de gravedad del robot, partiendo de la vista del plano YZ (ver Figura 50) que muestra el diagrama de cuerpo libre de los pesos y reacciones que actúan, se obtiene la Ec. 74.

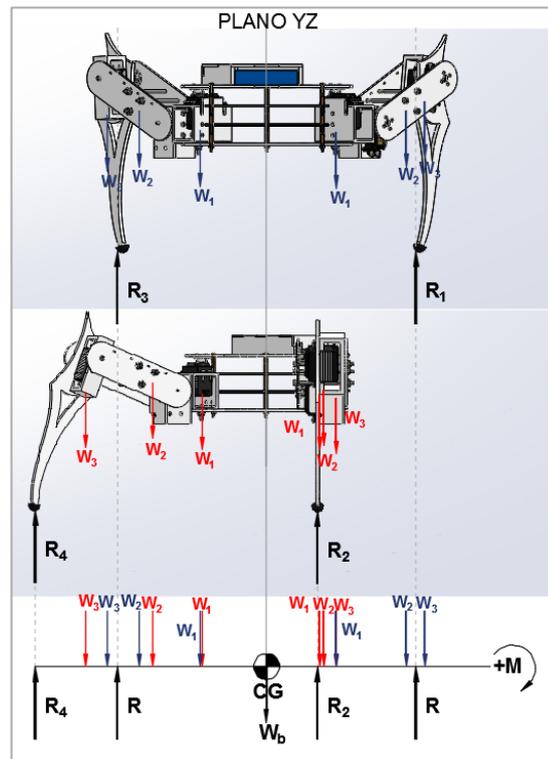


Figura 51. Diagrama de cuerpo libre del robot (Plano YZ)

$$\sum M_x = 0$$

$$\begin{aligned} W_1(y_{21} + y_{11}) + W_2(y_{22} + y_{12}) + W_3(y_{23} + y_{13}) + R(P_x/2) + R_4(P_x/2 + R_x/2) \\ = W_1(y_{41} + y_{31}) + W_2(y_{42} + y_{32}) + W_3(y_{43} + y_{33}) + R_2(P_x/2 - R_x/2) \\ + R(P_x/2) \end{aligned}$$

$$0.6 + R * 0.163 + R_4(0.163 + 0.085) = 0.79 + R_2 * (0.163 - 0.085) + R * 0.163$$

$$-0.078R_2 + 0.248R_4 = 0.19 \quad \text{Ec. 74}$$

Resolviendo el sistema de ecuaciones conformado por la Ec. 72, Ec. 73 y Ec. 74, se obtiene los valores de las reacciones en las cuatro extremidades.

$$R_2 = 10.36N$$

$$R_4 = 4.05N$$

$$R_1 = 7.16N$$

$$R_3 = 7.16N$$

### 3.2.3.2 ANÁLISIS DE ESFUERZOS POR ESLABÓN

Este estudio se llevará a cabo de acuerdo al punto crítico de equilibrio, en el cuál tres de sus patas permanecen en contacto con el suelo, mientras la otra se encuentra en su fase de transferencia. Partiendo de la definición del método de área cooperante para el análisis de esfuerzos, cuando la pata 4 se encuentra en fase de transferencia, la reacción que soportará la pata 2 corresponde a la suma de las reacciones 2 y 4, o igual a la mitad del peso del cuerpo más suma de los pesos de los eslabones. Entonces se tiene la Ec. 83.

$$R_2 = \frac{1}{2}W_b + W_2 + W_3$$

$$R_2 = 14.41 \text{ N}$$

Ec. 75

Una vez obtenido el valor de la reacción cuando solamente tres patas permanecen en contacto con el suelo, se procede a analizar las fuerzas que actúan en cada eslabón para su posterior análisis mecánico.

#### 3.2.3.2.1 ESLABÓN 3

Partiendo de la descomposición de la reacción de la pata 2 y el peso del correspondiente al tercer eslabón, en los ejes longitudinal y normal. Se tiene el diagrama de cuerpo libre del eslabón detallado en la Figura 52.

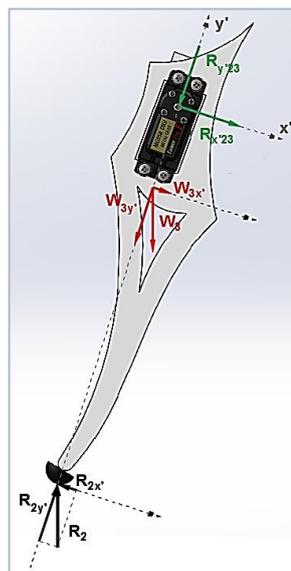


Figura 52. Diagrama de cuerpo libre del eslabón 3

Realizando el equilibrio de fuerzas en los ejes longitudinal y transversal se obtiene las ecuaciones la Ec. 76 y Ec. 77.

$$\begin{aligned}\sum F_{x'} &= 0 \\ -R_{2x'} + R_{x'23} + W_{3x'} &= 0 \\ R_{x'23} &= R_2 \sin 16 - W_3 \sin 16 \\ R_{x'23} &= 3.70 \text{ N}\end{aligned}\quad \text{Ec. 76}$$

$$\begin{aligned}\sum F_{y'} &= 0 \\ R_{2y'} - R_{y'23} - W_{3y'} &= 0 \\ R_{y'23} &= R_2 \cos 16 - W_3 \cos 16 \\ R_{y'23} &= 12.90 \text{ N}\end{aligned}\quad \text{Ec. 77}$$

### 3.2.3.2 ESLABÓN 2

Partiendo de las ecuaciones las reacciones obtenidas en la Ec. 76 y Ec. 77, se procede a analizar el eslabón 2. Este estudio se basa en el análisis de la pieza bracket servo fémur-1, detallado en el apartado de estructura de este capítulo.

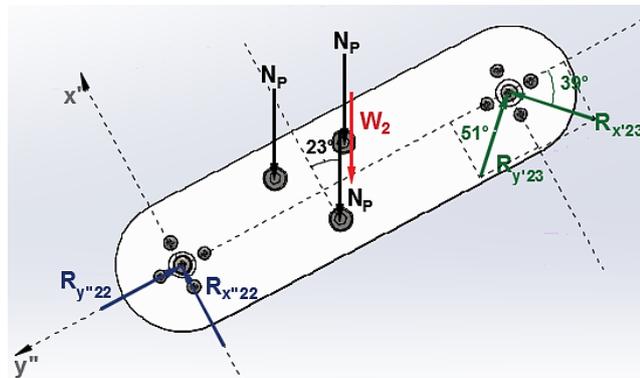


Figura 53. Diagrama de cuerpo libre del eslabón 2

Realizando sumatoria de momentos en el punto de aplicación de la fuerza  $R_{2y''}$ , se tiene la Ec. 78.

$$\begin{aligned}-110R_{y'23} \sin 51 - 110R_{x'23} \sin 39 - 55W_2 \cos 23 \\ + (36.31 \cos 23 - 10.2 \sin 23)N_p \\ + (59.38 \cos 23 - 9.84 \sin 23)N_p \\ + (47.53 \cos 23 + 9.97 \sin 23)N_p &= 0 \\ N_p &= 5.133 \text{ N}\end{aligned}\quad \text{Ec. 78}$$

Realizando el equilibrio de fuerzas en los ejes longitudinal y transversal se obtiene la Ec. 79 y la Ec. 80.

$$\begin{aligned} \sum F_{x''} &= 0 \\ R_{x''22} - (W_2 + 3N_p) \cos 23 + R_{x'23} \sin 39 + R_{y'23} \sin 51 &= 0 \\ R_{x''22} &= 8.21N \end{aligned} \quad \text{Ec. 79}$$

$$\begin{aligned} \sum F_{y''} &= 0 \\ -R_{y''22} + (W_2 + 3N_p) \sin 23 + R_{x'23} \cos 39 - R_{y'23} \cos 51 &= 0 \\ R_{x''22} &= 3.48N \end{aligned} \quad \text{Ec. 80}$$

### 3.2.3.2.3 ESLABÓN 1

Finalmente para el eslabón 1, serán necesarias únicamente las resultantes de las reacciones ejercidas por el eslabón 2 en los ejes longitudinal y transversal del mismo. Ver Figura 54.

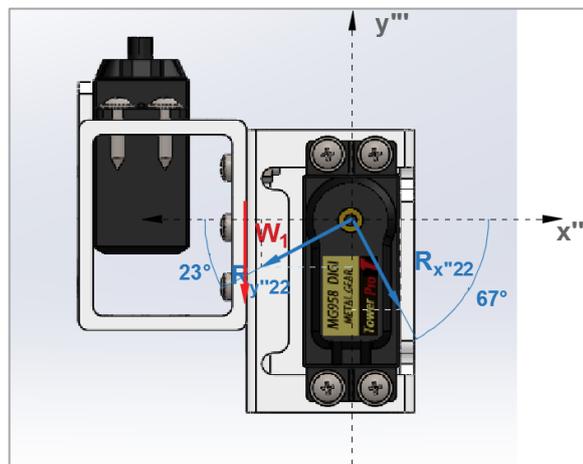


Figura 54. Diagrama del cuerpo libre del eslabón 1

Para obtener las resultantes se realiza la sumatoria de fuerzas en el punto de acción de las reacciones y se obtiene la Ec. 81 y Ec. 82.

$$\begin{aligned} R_{y'''} &= R_{y''22} \sin 23 + R_{x''22} \sin 67 \\ R_{y'''} &= 8.92N \end{aligned} \quad \text{Ec. 81}$$

$$\begin{aligned} R_{x'''} &= R_{y''22} \cos 23 - R_{x''22} \cos 67 \\ R_{x'''} &= 0N \end{aligned} \quad \text{Ec. 82}$$

### 3.2.4 ANÁLISIS DE LAS PIEZAS

Utilizando las reacciones calculadas para cada eslabón se procede a simular los esfuerzos producidos por dichas fuerzas y el factor de seguridad crítico mediante una herramienta CAD.

#### 3.2.4.1 ESLABÓN 3

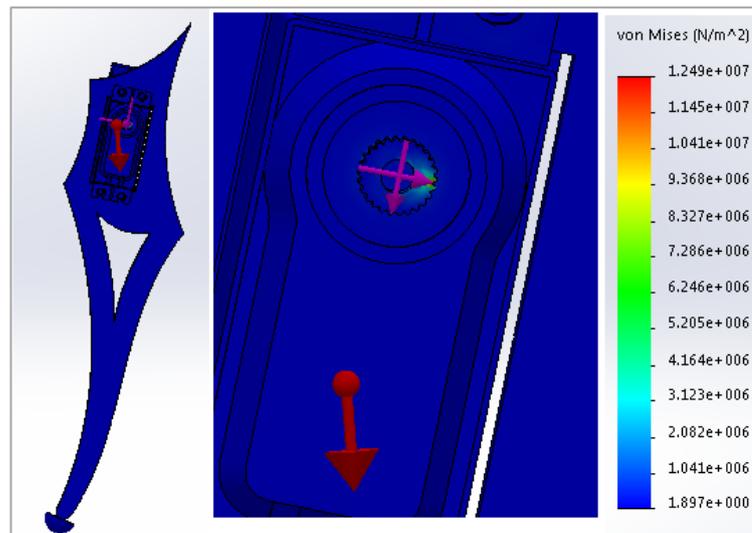


Figura 55. Diagrama de esfuerzos del eslabón 3

En la Figura 55 se describe la distribución de esfuerzos en el eslabón 3, por tanto el esfuerzo crítico fue encontrado en la parte derecha el engrane del servomotor con un valor de 12.49 MPa.

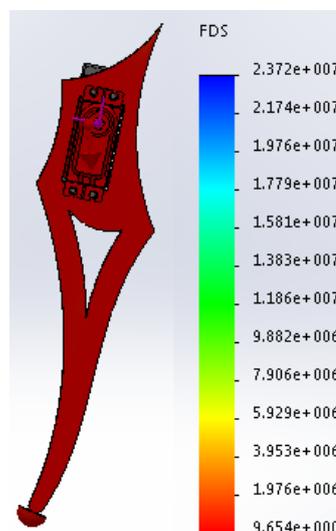


Figura 56. Factor de seguridad en el eslabón 3

El factor de seguridad crítico en el eslabón 3 tiene un valor de 9 (ver Figura 56), concluyendo que el material aplicado en este eslabón es el adecuado.

### 3.2.4.2 ESLABÓN 2

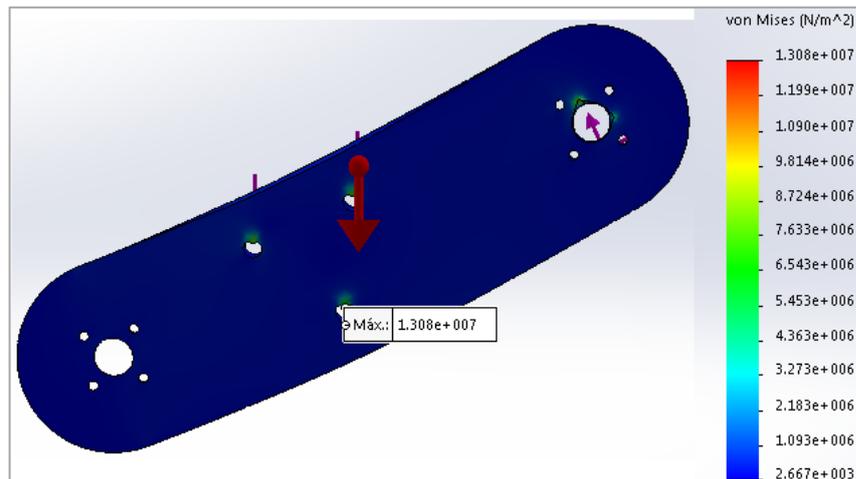


Figura 57. Diagrama de esfuerzos del eslabón 2

El esfuerzo máximo realizado en el eslabón 2 está ubicado en el agujero debajo de la línea de acción del eje longitudinal, con un valor de 13.08 MPa. Ver Figura 57.

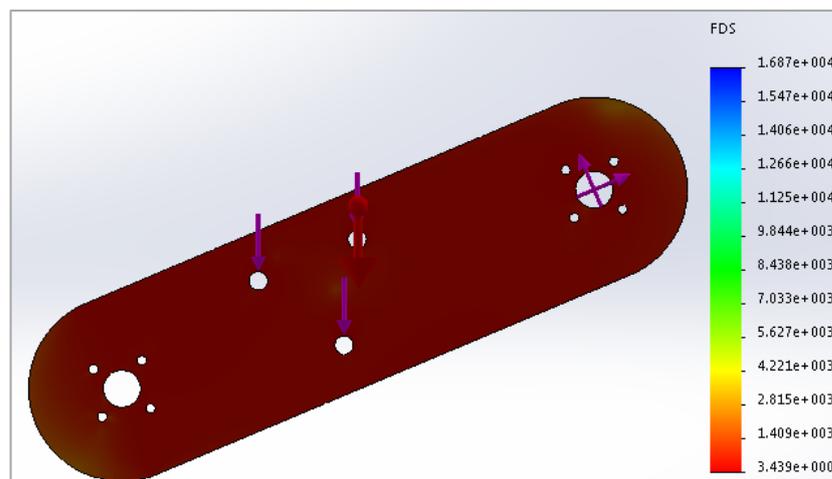
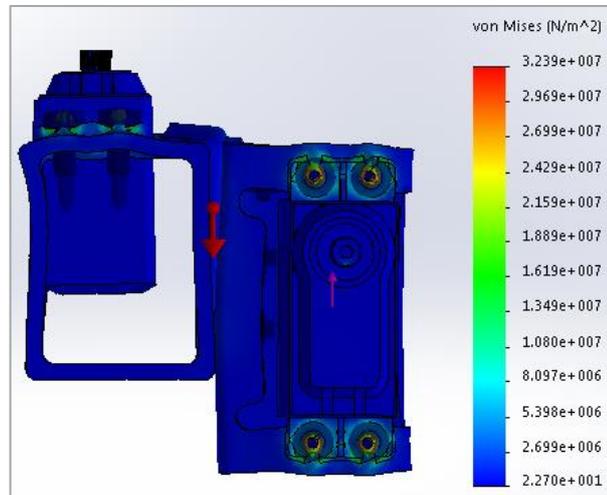


Figura 58. Factor de seguridad del eslabón 2

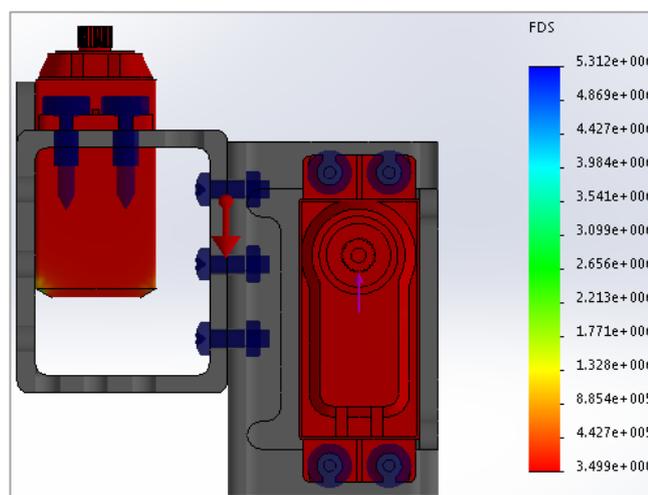
El factor de seguridad crítico tiene un valor de 3.44, concluyendo que el material establecido en dicho eslabón es apropiado para las cargas que soporta. Ver Figura 58.

### 3.2.4.3 ESLABÓN 1



**Figura 59. Diagrama de esfuerzos del eslabón 1**

El esfuerzo crítico está localizado en los alrededores de cada tornillo que sujeta al servo motor de la derecha del eslabón 1 con un valor de 32.39 MPa. Ver Figura 59.



**Figura 60. Factor de seguridad del eslabón 1**

El valor del factor de seguridad crítico en el eslabón 1 es de 3.5, por lo tanto los materiales empleados en dicho eslabón están correctamente seleccionados. Ver Figura 60.

### 3.3 DISEÑO DEL SISTEMA DE POTENCIA

El sistema de potencia necesario para el desarrollo del robot se basa en la alimentación necesaria para todos los elementos eléctricos y electrónicos utilizados.

#### 3.3.1 FUENTE DE ALIMENTACIÓN

De acuerdo a los parámetros de comparación mostrados en la Tabla 12 entre baterías de Níquel-Cadmio (NiCd), Níquel-Hidruro Metálico (NiMH) y Litio-Polímero (LiPo), se realizó la Tabla 14 para escoger la batería que más se ajuste a las características necesarias para este proyecto.

Tabla 12

Características de las baterías principales disponibles en el mercado

Característica	NiCd	NiMH	LiPo
<b>A</b> Densidad de almacenamiento	Baja	Baja (Aprox. 50% más que las de NiCd)	Alta
<b>B</b> Efecto memoria	Si	Bajo	-
<b>C</b> Costo	Barato	Barato	Alto
<b>D</b> Autodescarga	1% diario	1.5% diario	0.2% diario
<b>E</b> Vida útil (carga/descarga)	1000 ciclos	1000 ciclos	500 ciclos
<b>F</b> Contaminante	Sí	No	No

La primera característica de la Tabla 12 se refiere a la masa y el volumen que tiene una batería con relación a la capacidad de almacenamiento de la misma. En la Tabla 13 se puede observar las comparaciones numéricas entre los parámetros que conforman esta propiedad.

Tabla 13

Parámetros de densidad de almacenamiento de los distintos tipos de baterías

Parámetro	NiCd	NiMH	LiPo
<b>Dimensiones</b>	164 x 64 x 33mm	50.8x172.7x27.9 mm	138x47x25mm
<b>Peso</b>	680.4g	422g	301g
<b>Capacidad</b>	5000mAh	5000mAh	5000mAh
<b>Voltaje</b>	6V	7.2V	7.4V

Tabla 14

## Matriz de selección de baterías

	A		B		C		D		E		F		Total
Pond.	1	50%	1	6%	5	10%	1	9%	5	20%	1	5%	100%
NiCd	1	0.5	1	0.06	5	0.5	1	0.09	5	1	1	0.05	2.20
NiMH	2	1	4	0.24	4	0.4	1	0.09	5	1	4	0.2	2.93
LiPo	5	2.5	5	0.3	3	0.3	5	0.6	3	0.6	4	0.2	4.35

En la Tabla 14 se realiza la matriz de selección de las baterías, la cual muestra los puntajes en el rango de 1 a 5 y las ponderaciones de las distintas características de la Tabla 12 de acuerdo al nivel de importancia, obteniendo una mejor puntuación en las baterías de Litio-Polímero. La característica de más importancia para el desarrollo de las ponderaciones es la densidad de almacenamiento, debido a que es relevante tener una batería que no ocupe mucho espacio y su peso sea el menor posible para que no afecte la estabilidad del robot, sin embargo su capacidad de almacenamiento debe ser alta para proporcionar el suministro necesario a los servomotores y demás elementos de la placa.

La fuente de alimentación seleccionada es una batería recargable LiPo de 7.4 V y 5000mAh, cumpliendo con la demanda de potencia necesaria para los elementos indicados en la Tabla 15.

Tabla 15

## Elementos electrónicos principales y consumos de energía.

Elemento	Número de elementos	Voltaje necesario	Amperaje necesario	Amperaje parcial necesario
		V	mA	mA
Servo motores	12	6.6	700	8400
Arduino Mega	1	5	93	93
Cámara IP	1	5	1000	1000
Módulo de Frecuencia	1	5	20	20
Amperaje total necesario				9513

Bajo la condición de que los servomotores trabajen simultáneamente en su torque máximo y de acuerdo a los valores obtenidos en la Tabla 15, se tiene una duración mínima de la batería mostrada en la Ec. 83.

$$t_{DBAT} = \frac{5000mAh}{9513mA}$$

$$t_{DBAT} = 0.53h$$

$$t_{DBAT} = 32 \text{ min}$$

Ec. 83

### 3.3.2 DISEÑO DE LA PLACA PCB DE POTENCIA

#### 3.3.2.2 NORMAS IPC PARA EL DESARROLLO DE CIRCUITOS IMPRESOS

En las normas IPC para el desarrollo de circuitos impresos se encuentra la familia de estándares de diseño que comprende la serie IPC-2220, dentro de la cual está la norma IPC-2221B que se refiere a los lineamientos generales para el diseño de placas electrónicas.

Los requerimientos mínimos de ancho y espesor del material conductor para el desarrollo de placas de circuitos impresos o PCB (*Printed Board Circuit*) descritos en la norma IPC-221B, dependen de la temperatura máxima del circuito, de la corriente que va a pasar por el conductor y del tipo de placa. Este último, referente a la película de cobre, se ha dispuesto que tenga una magnitud de  $10z/\pi e^2$ .

Ingresando con los datos de corriente y temperatura se accede a la gráfica de la Figura 61 y se obtiene la sección transversal en mils, unidad determinada por:

$$1 \text{ mils} = 0.001 \text{ in} = 0.0254 \text{ mm}$$

Ec. 84

Con el valor de mils determinado, se accede a la gráfica de la Figura 62 para encontrar el valor mínimo necesario de ancho de pista para el conductor de la placa.

En la Figura 61 y Figura 62 se muestran los valores de las placas de control y potencia, determinados posteriormente.

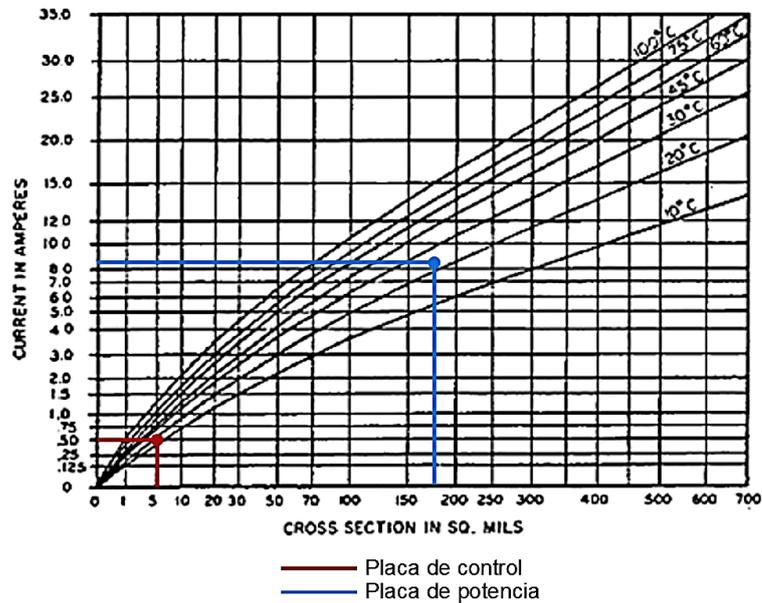


Figura 61. Gráfica de Corriente vs Sección transversal para diseño de PCB  
Fuente: (IPC, 2014)

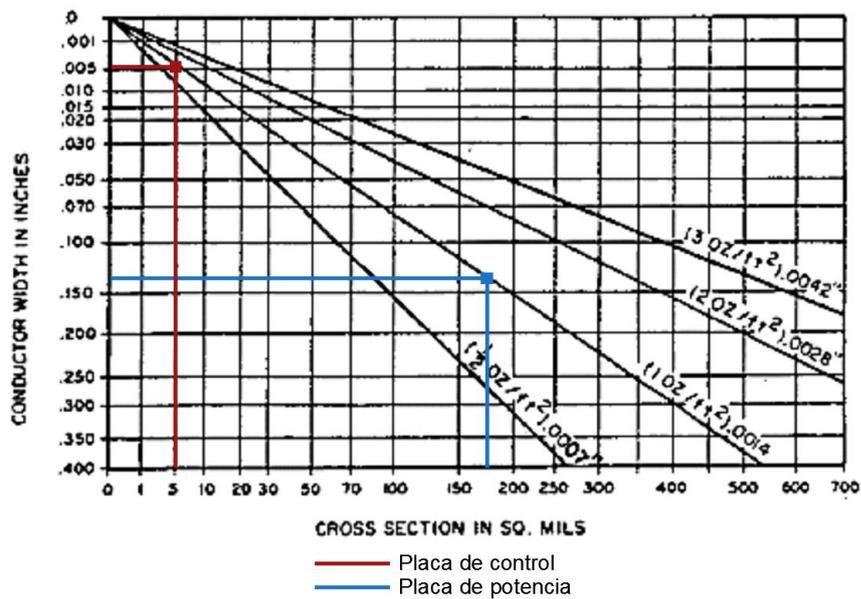


Figura 62. Gráfica de Ancho del conductor vs Sección transversal para diseño de PCB  
Fuente: (IPC, 2014)

### 3.3.2.3 PARÁMETROS DE LA PLACA DE POTENCIA

Estimando un consumo de corriente medio continuo de los servo motores igual a 0.7 A, teniendo un total de 12 motores, se llega un consumo de 8.4 A. Entonces, para valores de temperatura ambiente igual a 25°C y observando la Figura 61, se obtiene la sección transversal:

$$A = 175 \text{ mils}^2$$

De esta forma determinamos el ancho de pista necesario, observando la Figura 62:

$$\text{Ancho} = 0.125 \text{ in} = 3.175 \text{ mm}$$

Este es el ancho de pista mínimo para que la placa de potencia pueda soportar la corriente necesaria para los servomotores y que funcione sin inconvenientes.

Consecuentemente, se tiene la placa de potencia en la Figura 63.

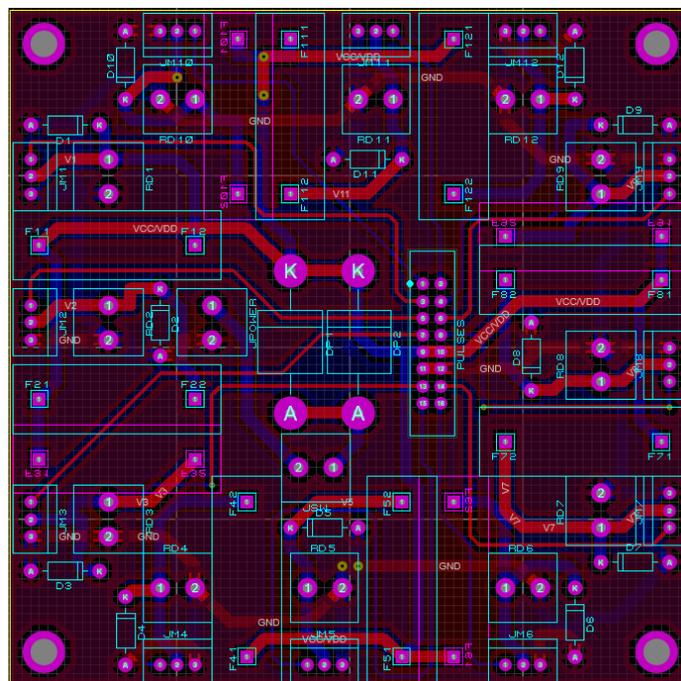
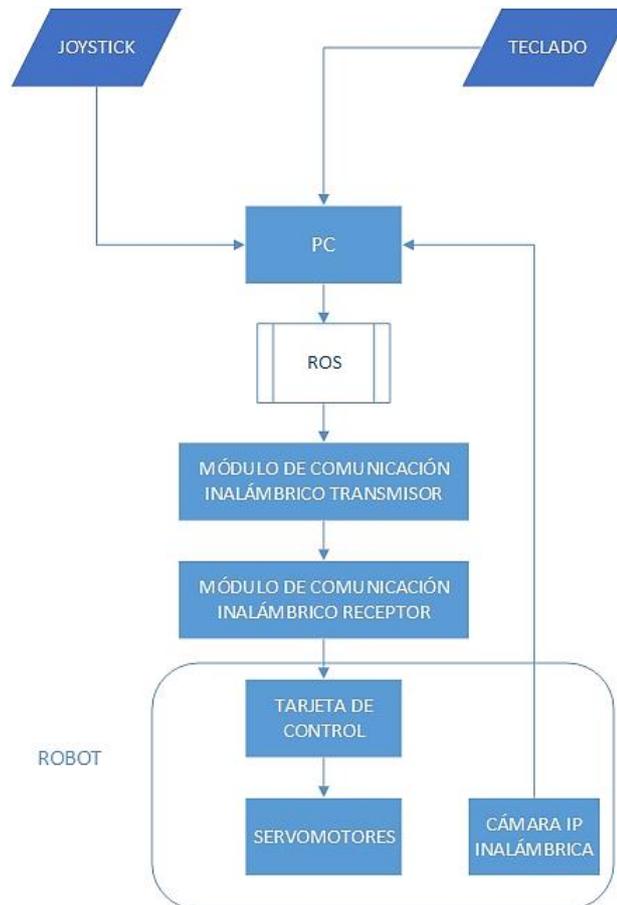


Figura 63. Placa de potencia

Esta PCB además de alimentar a los motores, comunica las señales de los PWM con la placa de control.

### 3.4 DISEÑO DEL SISTEMA DE CONTROL

#### 3.4.1 ARQUITECTURA DEL SISTEMA



**Figura 64. Arquitectura del sistema de control.**

En la Figura 64, se tiene los siguientes elementos del sistema de control:

- *Joystick*
- Computador con *Robot Operating System* (ROS) integrado.
- Módulo de comunicación inalámbrica transmisor
- Módulo de comunicación inalámbrica receptor
- Tarjeta de control
- Servo motores

El mando análogo *joystick* y el teclado serán las entradas del sistema, las cuales se manejarán mediante una interfaz gráfica compatible con ROS detallada posteriormente. ROS procesará estas entradas y se comunicará con

el puerto serial del computador para enviar los datos al transmisor RF, que a su vez se comunicará con el módulo receptor conectado en el puerto serial de la tarjeta de control. En la tarjeta se recibirán los comandos de control de la plataforma zoomórfica que mediante programación enviará los PWM necesarios a los servomotores para la ejecución del movimiento.

Una vez que el robot se desplace, empezará la transmisión de video de la cámara IP inalámbrica, colocada en el cuerpo del robot.

De esta manera, los elementos en que se focaliza el sistema de control del mecanismo son: ROS y la tarjeta de control elegida para el desarrollo del proyecto. Las características de *Robot Operating System* se describirán en la sección de Desarrollo de Software.

### 3.4.2 TARJETA DE CONTROL

Para el presente proyecto se usará la tarjeta de control Arduino MEGA 2560; debido a su particularidad de hardware y software libre y a las características señaladas en la Tabla 16.

**Tabla 16**

#### **Características Técnicas del Arduino MEGA 2560**

**Fuente: (ARDUINO, 2015)**

	<b>Características</b>	<b>Especificación</b>
1	Microcontrolador	ATmega2560
2	Voltaje de operación	5V
3	Voltaje de entrada	7-12V
4	Límite de voltaje	6-20V
5	Terminales digitales I/O	54(15 PWM)
6	Terminales análogos de entrada	16
7	Corriente DC por terminal I/O	20mA
8	Corriente DC por terminal 3.3V	50mA
9	Memoria Flash	256 KB ( 8KB de <i>bootloader</i> )
10	SRAM	8 KB
11	EEPROM	4 KB
12	Reloj	16 MHz
13	Longitud	101.52mm
14	Ancho	53.3 mm
15	Peso	37 g

El robot diseñado va a manejar un total de 12 PWM, correspondientes a los 12 grados de libertad, por lo que de acuerdo a la característica número 5 de la Tabla 16 que describe el número de pines digitales de entrada/salida, permite que esta tarjeta de control sea factible para el desarrollo del proyecto.

### 3.4.3 CAMARA IP WIRELESS

La cámara utilizada para monitorización fue seleccionada en base a los parámetros físicos y facilidad de conexión.

Tabla 17

#### Especificaciones técnicas de la cámara

Fuente: (WIFI/IP Camera)

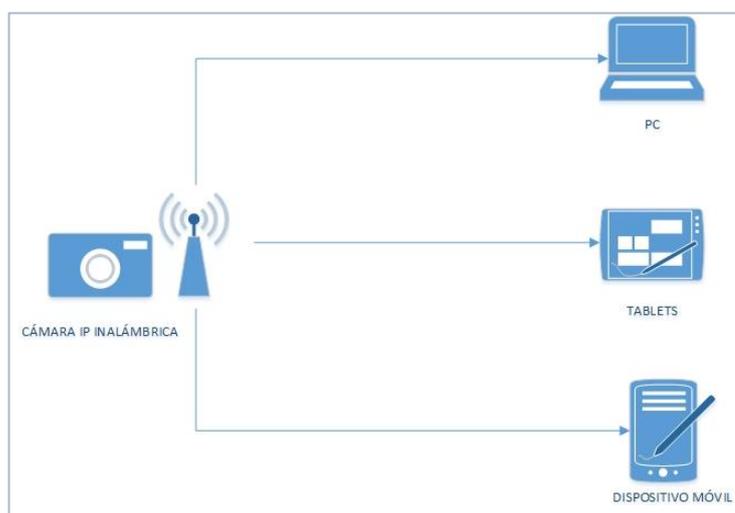
Parámetro	Especificación
Nombre	Cámara WIFI/IP
Formato de video	AVI
Formato de imagen	M-JPEG
Resolución de video	640x480 VGA
Resolución de video IP	320x240 QVGA
Resolución	10fps±1fps
Antena	2.4G 802.11n WIFI
Memoria	TF-32G
Proporción de video	4:3
Sistema	Windows ISO Android
Voltaje de Carga	5V DC
USB	Mini 5Pin USB
Tarjeta	Tarjeta Micro SD ( Tarjeta TF )

En la Tabla 17 se muestran las especificaciones técnicas de la cámara y las características que se adaptan al propósito de este proyecto son:

- Posee una entrada de tarjeta micro SD mediante la cual se puede realizar grabaciones continuas de manera local
- El formato de video es compatible con todos los dispositivos móviles y computadoras.

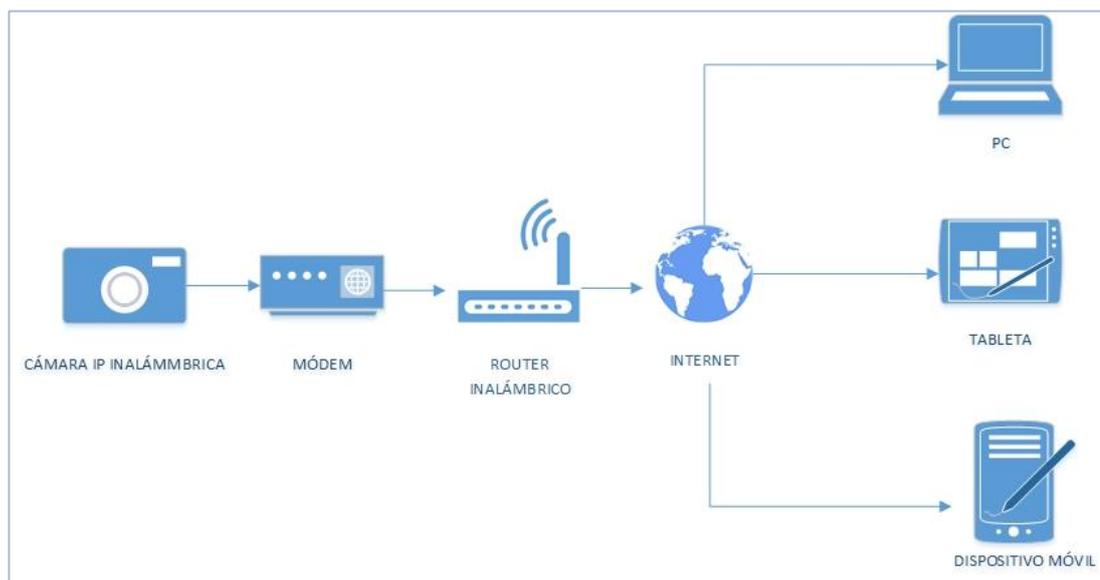
Para la supervisión remota la cámara tiene dos modalidades de conexión: Red de Área Local (LAN) y Red Inalámbrica LAN (WLAN).

En la conexión LAN, la cámara emite una señal wireless, que puede ser localizada en las redes del computador o dispositivo móvil. De esta manera la cámara se conecta punto a punto con el dispositivo. Ver Figura 65.



**Figura 65. Conexión LAN de la cámara**

En la conexión WLAN, la cámara se conecta a una red inalámbrica y se accede a ésta a través la misma. Ver Figura 65.



**Figura 66. Conexión WLAN de la cámara**

Las conexiones LAN y WLAN se dan a través de un software proporcionado por el fabricante denominado SYSM Monitor el cuál se puede

descargar de la página web del fabricante [www.scc21.net](http://www.scc21.net), en la misma que se encuentra los software para distintos sistemas y el manual de usuario para la operación del programa.

La cámara tiene un rango de distancia máximo de 80 metros desde el computador o dispositivo móvil, a partir de esta longitud la señal se vuelve débil y se interrumpe la transmisión.

Las características físicas por las cuáles se escogió son su tamaño y peso compactos. Las dimensiones son  $5.2 \times 2 \times 1.2$  cm y su peso alcanza los 16g.

#### 3.4.4 DISEÑO DE LA PLACA PCB DE CONTROL

De acuerdo a lo expuesto en la arquitectura del sistema de control, los elementos que se conectarán a la placa de control Arduino MEGA 2560 son: el módulo de comunicación inalámbrica receptor y las entradas de control de los 12 servomotores. En la Tabla 18 se muestra la distribución de terminales en la placa de control.

**Tabla 18**  
**Terminales utilizados en la placa de control**

Terminal	Tipo	Especificación
D2	Salida digital	PWM servomotor 1
D3	Salida digital	PWM servomotor 2
D4	Salida digital	PWM servomotor 3
D5	Salida digital	PWM servomotor 4
D6	Salida digital	PWM servomotor 5
D7	Salida digital	PWM servomotor 6
D8	Salida digital	PWM servomotor 7
D9	Salida digital	PWM servomotor 8
D10	Salida digital	PWM servomotor 9
D11	Salida digital	PWM servomotor 10
D12	Salida digital	PWM servomotor 11
D13	Salida digital	PWM servomotor 12
D18	TX1	Módulo de comunicación
D19	RX1	Módulo de comunicación

Según la Tabla 18 se ocupan 14 terminales digitales, los mismos que consumen 20mA cada uno, teniendo por lo tanto un consumo total de 300mA.

Para evitar sobrecarga en la placa se dimensionó un consumo corriente máxima de 500mA.

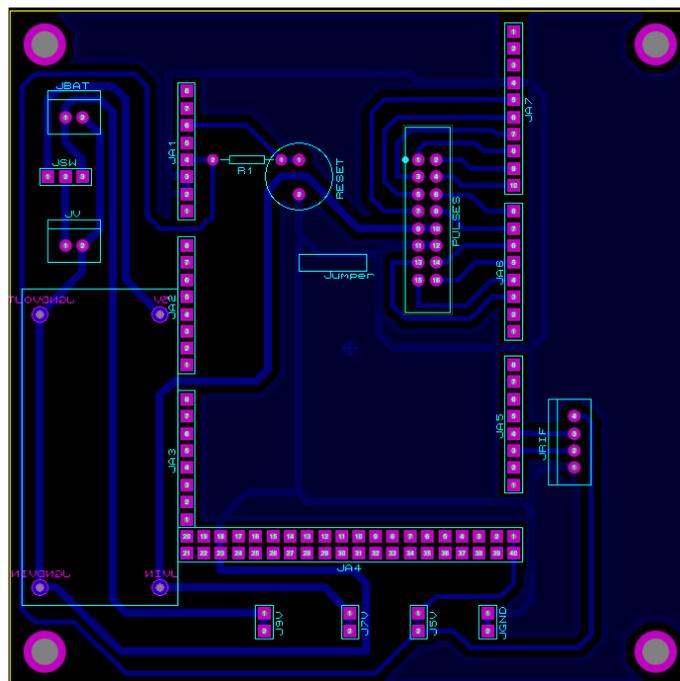
Basándose en la Figura 61 y con los parámetros de operación de  $t = 25^{\circ}C$  e  $I = 500mA$ , se obtiene una sección transversal de:

$$A = 5 \text{ mils}^2$$

Observando la Figura 62 y encontrando el ancho de pista de acuerdo con el valor de la sección transversal obtenida:

$$\text{Ancho} = 0.005 \text{ in} = 0.127 \text{ mm}$$

Con estos valores se diseña la placa de control detallada en la Figura 67.



**Figura 67. Placa de control**

La placa de control consta del elemento *pulses* mediante el cual se comunica los terminales de control de los servos alimentados desde la placa de potencia a la control.

Además esta placa consta de un convertor DC/DC para la alimentación del Arduino MEGA 2560, el mismo que poseerá dos alternativas de alimentación: mediante la batería LiPo de 7V o con una batería alcalina normal de 9V. Mediante la segunda opción de alimentación se controla los picos de corriente en el transitorio de encendido y reseteo del Arduino, por lo que se recomienda ésta para estabilidad del robot.

### **3.5 DESARROLLO DE SOFTWARE**

#### **3.5.1 ROBOT OPERATING SYSTEM (ROS)**

En esta sección se analizarán los conceptos y herramientas del framework utilizado, necesarios para el desarrollo del software del presente proyecto.

ROS es un meta sistema operativo de fuente libre diseñado para aplicaciones robóticas en el 2007 por la Universidad de Stanford en Estados Unidos.

En ROS.org (2017) detalla que los servicios que presta ROS son:

- Abstracción de hardware.
- Control de dispositivos de bajo nivel.
- Implementación de funcionalidad de uso común.
- Gestión de paquetes y envío de mensajes entre procesos.

ROS implementa una red de procesos comunicados de diferentes formas, incluyendo la transmisión síncrona de servicios con RPC (*remote procedure call*), la transmisión síncrona de datos a través de *topics* y el almacenamiento de datos por medio del servidor de parámetros (ROS.org, 2017).

La ventaja de este *framework* es que maneja librerías mediante repositorios de libre acceso, por lo que los trabajos efectuados pueden ser compartidos a la comunidad.

ROS posee un lenguaje de programación flexible, debido a que su programación se puede realizar mediante distintos lenguajes, entre los cuáles están: Python, C++ y Lisp (ROS.org, 2017).

Las distribuciones se definen como las versiones que existen del sistema para ser instaladas, similares a las distribuciones de Linux; dichas distribuciones hacen más fácil la instalación y mantenimiento del software.

Han sido concebidas varias distribuciones para ROS entre ellas ROS.org (2017) enumera:

- ROS Box Turtle (2010)
- ROS C Turtle (2010)
- ROS Diamondback (2011)
- ROS Electric Emys (2011)
- ROS Fuerte Turtle (2012)
- ROS Groovy Galapagos (2012)
- ROS Hydro Medusa (2013)
- ROS Indigo Igloo (2014)
- ROS Jade Turtle (2015)
- ROS Kinetic Kame (2016)

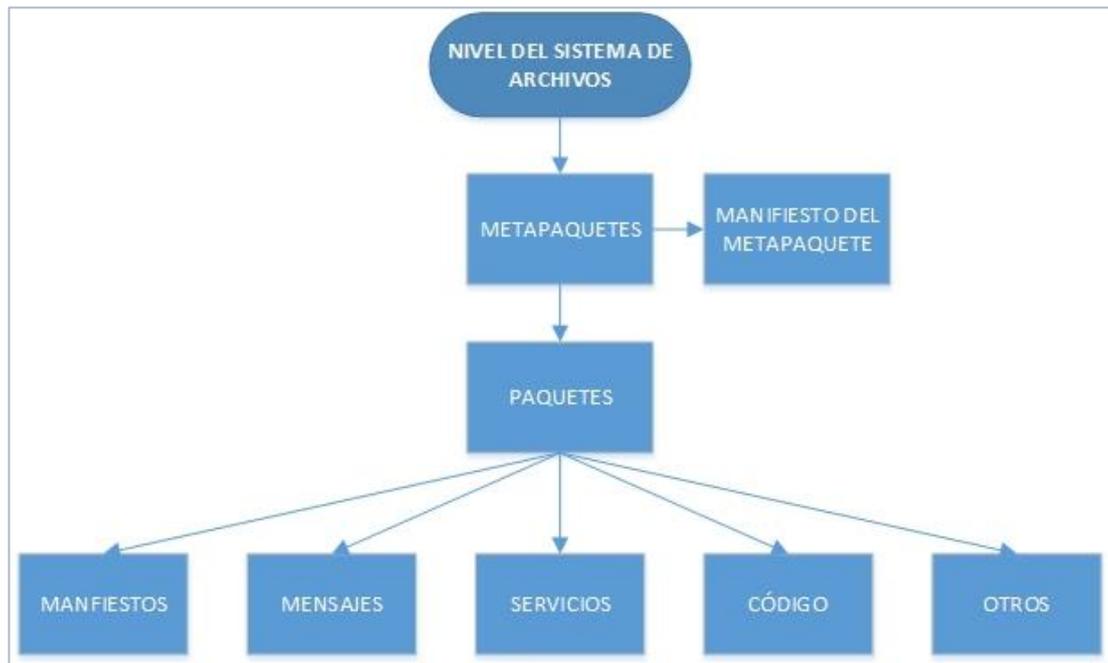
El desarrollo de este proyecto, se efectuó sobre la distribución de ROS Indigo Igloo, recomendado por su estabilidad, debido a que se estaban ejecutando pruebas sobre las versiones más recientes.

ROS trabaja sobre plataformas Unix, por lo que para el desarrollo del proyecto se escogió el sistema operativo Linux.

### **3.5.1.1 ARQUITECTURA DE ROS**

#### **3.5.1.1.1 NIVEL DE ARCHIVOS DEL SISTEMA**

Este sistema se basa en los recursos que tiene ROS para su correcto funcionamiento, la estructura de este nivel se visualizar en la Figura 68.



**Figura 68. Nivel del Sistema de Archivos de ROS**

**Fuente: (Martínez & Fernández, 2013)**

Metapaquetes: representan a un grupo de paquetes especializados y relacionados entre sí, utilizados para hacer compatibles los nuevos paquetes con versiones anteriores de ROS (Martínez & Fernández, 2013).

Paquetes: son la unidad principal para organizar el software en ROS. Un paquete contiene programas llamados “nodos” siendo una de las estructuras más pequeñas de organización (Martínez & Fernández, 2013).

Manifiesto (*package.xml*): provee información acerca del nombre, licencia, dependencias, banderas de compilación, entre otras características de un paquete (Martínez & Fernández, 2013).

Mensaje (*msg*): define la estructura y descripción de un dato que se envía entre procesos, está almacenado en el directorio `my_package/msg/MyMessageType.msg` (Martínez & Fernández, 2013).

Servicio (*srv*): define la estructura de petición y respuesta de un servicio así como su descripción, está almacenado en el directorio `my_package/srv/MyServiceType.srv` (Martínez & Fernández, 2013).

### 3.5.1.1.2 NIVEL GRÁFICO COMPUTACIONAL

ROS crea una red de trabajo denotado en la Figura 69, en donde todos los procesos están conectados, ya que cualquier nodo del sistema puede acceder a la red interactuando con otros nodos, y así mismo, recibir o enviar información a través de la red.



**Figura 69. Nivel Gráfico Computacional**

**Fuente: (Martínez & Fernández, 2013)**

**Nodo:** es un proceso creado para desempeñar una función específica como cálculo de valores o procesamiento de señales. Un nodo puede ser escrito con las librerías llamadas cliente que tiene ROS como `roscpp` o `rospy`, con lenguajes de programación C++ y *python*, respectivamente (Martínez & Fernández, 2013).

**Maestro:** el nodo maestro crea un registro de nombres de todos los nodos. Es muy importante que este nodo sea ejecutado para que pueda haber la comunicación entre los demás nodos por medio de servicios o mensajes (ROS.org, 2017).

**Servidor de parámetros:** entrega la posibilidad de almacenar datos usando claves en una localización central que se encuentran en el nodo maestro. Con este parámetro podemos configurar ciertas características de los nodos mientras están ejecutándose (ROS.org, 2017).

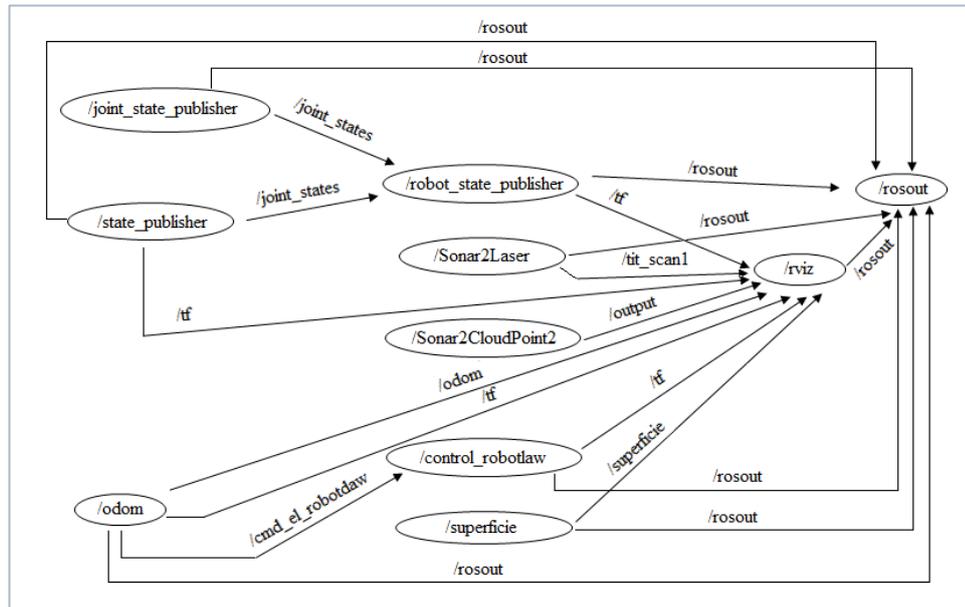
**Mensajes:** los nodos se comunican a través de un mensaje que contiene información relevante para otro nodo. Un mensaje comprende una estructura de datos del mismo campo. ROS tiene varios tipos de mensajes estándar y también tiene la posibilidad de personalizar mensajes propios (Martínez & Fernández, 2013).

**Tópicos:** es un sistema de comunicación para enviar mensajes por medio de una publicación y suscripción. Para enviar el mensaje, un nodo debe ser publicador y para recibir el mensaje otro nodo debe ser suscriptor. Al realizar el proceso de enlace, estos nodos deben tener el mismo *topic* el cual tendrá un nombre único que identifique el mensaje que se está publicando o recibiendo. Pueden existir varios editores y suscriptores para un mismo tópico (ROS.org, 2017).

**Servicios:** es un sistema de comunicación que está constituido por un nodo que ofrece un servicio (servidor) y otro que recibe o utiliza ese servicio (cliente); la diferencia respecto al sistema con *topics* radica en que al momento de realizar la comunicación el nodo servidor va a recibir una respuesta del cliente cuando el servicio haya sido proporcionado (ROS.org, 2017).

La Figura 70 representa el funcionamiento de un robot bajo condiciones reales, en donde los nodos están representados por circunferencias y los tópicos están representados por los nombres sobre las flechas que comunican cada nodo.

El nodo ROS master realizará un llamado a los nodos comunicados con las funciones llamadas *callbacks*; las que actúan cuando la información de registro cambia para permitir crear conexiones dinámicas con nuevos nodos que estén ejecutándose. Esta arquitectura permite un funcionamiento desacoplado mediante los cuales se puede construir estructuras más grandes, por lo que cualquier nodo puede ser inicializado y finalizado en cualquier momento sin conducir a errores (Martínez & Fernández, 2013).



**Figura 70. Representación del nivel gráfico del funcionamiento de un robot**

**Fuente: (Martínez & Fernández, 2013)**

### 3.5.1.1.3 COMUNIDAD DE DESARROLLO LIBRE

Está representado por fuentes de información que son liberadas para toda la comunidad de ROS permitiendo un intercambio de software y conocimiento.

- Repositorios: es una red en donde se almacena software de diferentes instituciones que desarrollan y liberan paquetes y programas para la comunidad de ROS (ROS.org, 2017).
- ROS Wiki: es el foro principal para la documentación e información acerca de ROS (ROS.org, 2017).

### 3.5.1.1.4 ROSCORE

Corresponde al nodo de ROS *Master*, encargado de ejecutar en el sistema un conjunto de nodos y permitir su interconectividad y comunicación. Los nodos que ejecuta el comando se encuentran especificados en el archivo `roslaunch/roscore.xml`, y son: ROS *Master*, ROS *Parameter Server* y *rosout logging node*, los cuales son primordiales para el funcionamiento de ROS (ROS.org, 2017).

### 3.5.1.1.5 ROSLAUNCH

Es una herramienta de gran importancia debido a que tiene la capacidad de ejecutar varios nodos de ROS al mismo tiempo, incluido el nodo ROS *Master* que se ejecutará automáticamente al utilizar este comando. Tiene la particularidad de poder establecer parámetros de los nodos incluidos como también volver a ejecutar automáticamente nodos o procesos concluidos. El lenguaje de programación para este tipo de archivos es XML pero consta de una extensión *launch* para su lanzamiento (ROS.org, 2017).

### 3.5.1.2 NODO DE CONTROL JOYSTICK

Para poder interpretar los datos de entrada que el *joystick* ingresa a la computadora en el entorno de ROS, es necesario instalar un conjunto de paquetes que ya fueron desarrollados para su control y correcto desempeño; tomando en cuenta la versión de ROS que se va a trabajar, en este caso será ROS Indigo. El conjunto de paquetes se puede encontrar como: *ros-indigo-joystick-drivers* (Martínez & Fernández, 2013). Una vez instalado se debe verificar que funcione adecuadamente por medio de los comandos de la consola observados en la Tabla 19.

Tabla 19

Comandos de verificación y funcionamiento del *joystick*  
(Martínez & Fernández, 2013)

Comando	Tipo	Función
\$ ls /dev/input/	Verificación	Reconocimiento de la conexión a la computadora.
\$ sudo jstest /dev/input/js0	Verificación	Verificación del funcionamiento del puerto creado.
\$ rosrn joy joy_node	Funcionamiento	Activación del nodo que controla el <i>joystick</i> .
\$ rostopic echo /joy	Funcionamiento	Desempeño de los ejes y botones.

Para poder utilizar el *joystick*, siempre se debe ejecutar el nodo de control del mismo, con el cuál se puede enviar mensajes al pulsar cualquier botón, agrupados en dos arreglos llamados *buttons* y *axes*, representados en la Figura 71.

```

---
header:
seq: 157
stamp:
  secs: 1357571648
  nsecs: 430257462
  frame_id: ''
axes: [-0.0, -0.0, 0.0]
buttons: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

```

Figura 71. Mensaje de envío de datos del nodo *joystick* a ROS  
(Martínez & Fernández, 2013)

### 3.5.1.3 HERRAMIENTAS PARA LA CONSTRUCCIÓN DE UN PROYECTO EN ROS

#### 3.5.1.3.1 ESPACIO DE TRABAJO

Un espacio de trabajo es la estructura principal para el desarrollo de una aplicación en ROS, ya que dentro de él se desarrollan y construyen los nodos y archivos *launch* (ROS.org, 2017).

Al crear un espacio de trabajo a su vez se crean automáticamente dos carpetas: *build* en donde se almacenan los archivos *cmakelist.txt* de cada paquete, los cuales construyen y verifican errores en la programación de los nodos y la carpeta *devel* en donde se guardan los ejecutables de cada nodo creado.

#### 3.5.1.3.2 PAQUETE

Un paquete es un conjunto de nodos organizados por carpetas que debe ubicarse dentro del espacio de trabajo en una carpeta llamada *src* que debe ser creada. Es importante conocer las dependencias o librerías para cada nodo que se encuentra dentro de este paquete, ya que si no se toma en cuenta todas las necesarias, no se podrá construirlo correctamente. . Para los nodos programados en lenguaje C++ debe existir una carpeta igualmente llamada

*src* y para los nodos programados en *python* debe existir una llamada *scripts* (ROS.org, 2017).

Un paquete debe tener dos archivos: el archivo *CMakeList.txt*, que va a compilar los nodos creados dentro de los paquetes y el *package.xml*. Las dependencias de cada nodo se incluirán en los archivos *CMakeList.txt* y *package.xml* (ROS.org, 2017).

#### **3.5.1.3.3 NODOS**

Los nodos siempre se crean dentro de un paquete para que puedan ser compilados y construidos.

#### **3.5.1.3.4 SRC**

Los nodos escritos en C++ deben estar definidos en el archivo *CMakeList.txt*, debido a que este se encarga de compilar el nodo en busca de errores y crea un ejecutable para poder utilizarlo (ROS.org, 2017).

#### **3.5.1.3.5 SCRIPTS**

Los nodos escritos en el lenguaje de programación *python* no necesitan ser definidos en el archivo *CMakeList.txt*, pero si es necesario crear su ejecutable por medio del comando: `$ chmod +x [nombre_del_nodo.py]` (ROS.org, 2017).

#### **3.5.1.3.6 URDF**

Para crear el modelo de un robot virtual en el entorno de ROS se debe realizar un diseño escrito mediante código en un archivo URDF. Con sus siglas en inglés *Unified Robot Description Format* es un archivo con formato XML que describe las partes, uniones, dimensiones, velocidad y demás características de un robot (Martínez & Fernández, 2013).

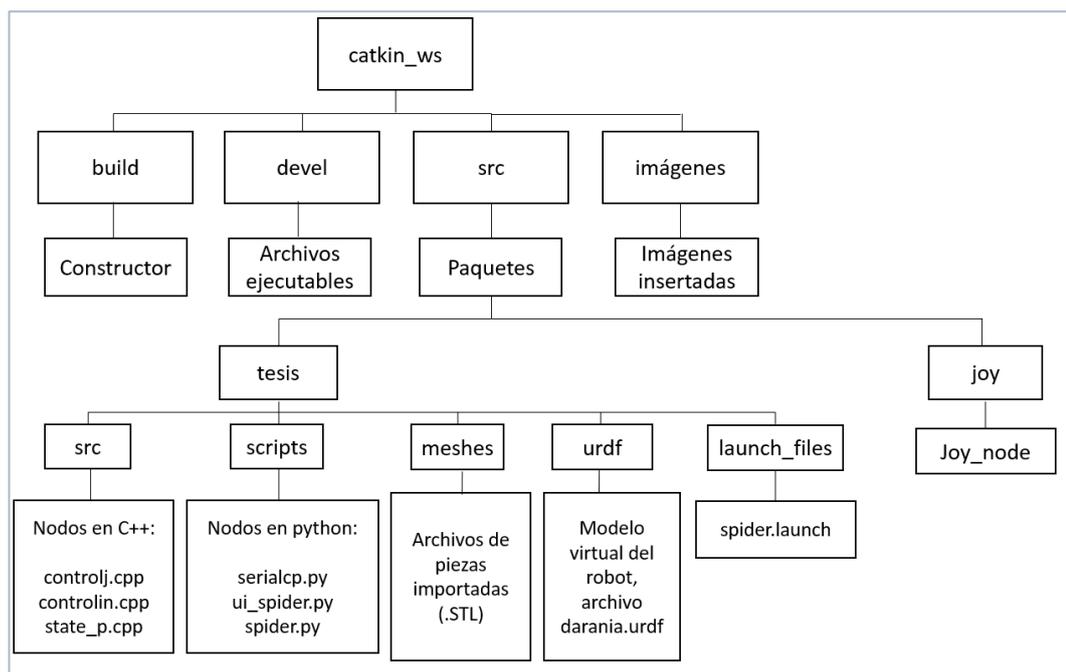
El simulador incorporado en el entorno de ROS llamado RVIZ es básico cuando se refiere a diseñar formas complejas, por lo que se necesita utilizar otros programas para diseñar las piezas de robot y posteriormente exportarlas con un programa llamado `sw_urdf_exporter` para su uso en el simulador.

### 3.5.1.3.7 MESHES

Un archivo con extensión `.STL` llamado *mesh*, que en español significa malla, se utiliza cuando se necesita ensamblar piezas más complejas en el modelo virtual de un robot (ROS.org, 2017).

## 3.5.2 ARQUITECTURA DEL SOFTWARE PROYECTO

La Figura 72 representa esquemáticamente la estructura que tiene el proyecto desarrollado a nivel de paquetes, nodos y organización jerárquica.



**Figura 72. Estructura del proyecto**

La Figura 72 muestra inicialmente el espacio de trabajo llamado “`catkin_ws`”, además se puede observar la carpeta “`imágenes`” que contiene las ilustraciones de la interfaz gráfica y debe estar ubicada en el mismo espacio de trabajo para su correcto funcionamiento.

### 3.5.2.1 PARTES DEL PROYECTO

El paquete creado para el desarrollo del proyecto se estableció con el nombre “tesis”, con las dependencias: roscpp, rospy, urdf, tf, std\_msg. El otro paquete mencionado en la Figura 72 denominado “joy”, corresponde al nodo del *joystick* y se instaló de acuerdo a las características especificadas en la Tabla 19.

#### 3.5.2.1.1 CARPETA SRC

Dentro de la carpeta “src” se tiene los nodos programados en C++ con las siguientes funcionalidades:

- El nodo llamado “controlin.cpp” se encarga de procesar las coordenadas ingresadas en la interfaz gráfica para enviar los ángulos correspondientes al simulador y la señal inalámbrica al robot para que se realicen los ciclos repetitivos necesarios de traslación o rotación para alcanzar dichas coordenadas.
- El nodo llamado “controlj.cpp” realiza el procesamiento de las señales enviadas por el *joystick* a través del “*joy\_node*” para posteriormente transmitir los ángulos al simulador y enviar la señal inalámbrica al robot dependiendo el botón activado.
- El nodo llamado “state\_p.cpp” recibe los ángulos tanto de la interfaz gráfica como del *joystick* para posteriormente publicarlos en el modelo URDF cargado en el programa RVIZ y así observar la simulación.

#### 3.5.2.1.2 CARPETA SCRIPTS

Dentro de la carpeta “scripts” se tiene los nodos programados en *python* con las siguientes funcionalidades:

- El nodo llamado “serialcp.py” se encarga de recibir las señales de entrada, tanto la interfaz gráfica como del *joystick*, comunicar el

puerto USB con el dispositivo inalámbrico y finalmente enviar estos datos inalámbricamente de forma serial.

- El nodo llamado “ui\_spider.py” establece mediante código la configuración gráfica de la interfaz inicialmente creada en un programa denominado *Qt Creator*.
- El nodo llamado “spider.py” ejecuta las acciones de cada botón, etiqueta y cuadro de texto, que haya sido establecido en la configuración gráfica de la interfaz, así como también, envía las coordenadas y señales de los botones ejecutados al nodo correspondiente para su procesamiento.

#### 3.5.2.1.3 ARCHIVO URDF

El ensamble de las piezas del robot configuradas en archivos con extensión .STL, las uniones entre eslabones conocidas como *joints* y su movimiento relativo con el entorno del simulador fueron establecidos mediante el archivo “darania.urdf”.

#### 3.5.2.1.4 ARCHIVO LAUNCH

El archivo “*spider.launch*” tiene como misión ejecutar todos los nodos que se necesitan para el correcto funcionamiento del proyecto tanto en el plano virtual como en el plano real. Este archivo será el encargado de ejecutar los nodos descritos en la Tabla 20.

Tabla 20

Nodos de lanzamiento en el archivo *launch*

Función	Control	Procesamiento
<b>Joystick</b>	<i>joy_node</i>	controlj.cpp
<b>HMI</b>	<i>spider.py</i>	controlin.cpp
<b>Comunicación Serial</b>	serialcp.cpp	-
<b>Simulador RVIZ</b>	state_p.cpp	darania.urdf

En el caso del simulador RVIZ, el archivo “darania.urdf” necesariamente debe ser cargado en el simulador por medio de un archivo *launch*.

El nodo “*spider.py*” puede ser ejecutado por el archivo “*spider.launch*”, pero en un archivo *launch* no se puede cargar imágenes, por lo que tendrá que ser ejecutado por separado para que se pueda observar correctamente el diseño completo de la HMI.

### 3.6 DISEÑO DEL SISTEMA DE COMUNICACIÓN

La comunicación entre el ordenador y la tarjeta de control ubicada en el robot se realizó de manera serial mediante módulos de radiofrecuencia.

#### 3.6.1 MÓDULOS DE RADIOFRECUENCIA DE COMUNICACIÓN INALÁMBRICA



**Figura 73. Módulos de radiofrecuencia 3DR *Radio Telemetry* (Maestro y Esclavo)**

**Fuente: (Geeetech Wiki, 2014)**

El equipo elegido para realizar esta función se denomina *3DR Radio Telemetry*, el cual consta de dos módulos: uno de transmisión y otro de recepción de datos. Es muy versátil, tiene un amplio rango de transmisión y un costo relativamente moderado con relación a las facilidades que ofrece. Ver Figura 73.

Las especificaciones técnicas de estos dispositivos se despliegan en la Tabla 21.

Tabla 21

**Características técnicas de 3DR *Radio Telemetry***

Fuente: (Geeetech Wiki, 2014)

Característica	Especificación
Rango de frecuencias de operación	915MHz – 433MHz
Sensibilidad de recepción	-121 dBm
Potencia de transmisión	20dBm(100mW)
Velocidad de transmisión por aire	250 KBPS
Factor de trabajo	Configurable
Voltaje de alimentación	5V
Dimensiones	18x51mm(esclavo)-18x71mm (maestro)
Peso	50g

Además de las características mencionadas en la Tabla 21, los dispositivos de comunicación poseen las siguientes ventajas:

- El maestro se instala con una antena larga y el esclavo se acomoda con una antena corta para reducir la carga del dispositivo a controlar (Geeetech Wiki, 2014).
- El módulo maestro tiene una conexión USB para facilitar su conexión al ordenador, mientras que el módulo esclavo posee un socket de 5 pines para facilitar su conexión a la placa de control (Geeetech Wiki, 2014).
- El rango de transmisión puede alcanzar grandes distancias en el orden de kilómetros (Geeetech Wiki, 2014).

**3.6.1.1 SOFTWARE DE CONFIGURACIÓN**

Aunque su configuración se puede realizar mediante cualquier programa que maneje el puerto serial del computador; el creador del producto sugiere el uso del programa de distribución libre *Mission Planer*, debido a que se puede realizar la configuración de los dos módulos a la vez y sus características se pueden definir fácilmente mediante una interfaz amigable.

Debido a que el software mencionado no se limita a la configuración de los dispositivos de radiofrecuencia, al ser compatible con ciertas tarjetas de control tiene la capacidad de desempeñar las siguientes funciones según Geeetech Wiki (2014):

- Controlar dinámicamente algún vehículo autónomo.
- Configurar y establecer valores para un óptimo funcionamiento del vehículo autónomo o plataforma robótica.
- Su interfaz posee un simulador de vuelo PC, para aplicaciones de drones.
- Puede monitorear la plataforma robótica, guardar información de sensores y controlar remotamente la misma mediante comunicación con los servomotores.

Para la configuración de los dispositivos es necesario conectar el módulo maestro a la computadora y el esclavo debe estar alimentado (Geeetech Wiki, 2014). La velocidad de transmisión que se estableció para el proyecto 115200 baudios en el puerto adecuado. Las frecuencias y factor de trabajo que se elegirán dependen de las regulaciones del país en el que se utilice y de la tarjeta electrónica usada para su conexión. Se puede observar todas las opciones de configuración en la Figura 74.



Figura 74. Interfaz para configurar el 3DR *Radio Telemetry*

Fuente: (ArduPilot Dev team, 2016)

Los parámetros más importantes se presentan a continuación:

- **ID:** se utiliza para darle una identificación al vehículo autónomo debido a que pueden existir en la misma área más vehículos siendo controlados por *Mission Planner*, el número común es el 25 (Geeetech Wiki, 2014).
- **Baud:** es la velocidad de transmisión establecida por defecto en 57600 baudios (Geeetech Wiki, 2014).
- **Air Speed:** es la velocidad por la cual los dos radios se comunicarán entre sí, por defecto se establece en 64 kbps; pero si se requiere incrementar el rango se debe reducir este valor.
- **ECC:** habilita el control de errores (Geeetech Wiki, 2014).
- **Tx Power:** se refiere a la potencia de transmisión, que debe ser escogida según el país, el valor por defecto es 20 (Geeetech Wiki, 2014).
- **Duty Cycle:** es el máximo porcentaje de tiempo que el radio transmitirá los paquetes, dependiendo del país la normativa permite mayor potencia de transmisión o mayor ancho de banda (Geeetech Wiki, 2014).

### 3.6.1.2 CONEXIÓN

El dispositivo esclavo debe obligatoriamente ser alimentado con 5v y con la polaridad correcta, debido a que no tiene protección de polaridad y puede dañar todo el circuito si se conecta con polaridad invertida (Geeetech Wiki, 2014). El uso de los pines se especifica en la Figura 75.



**Figura 75. Conexiones del módulo esclavo a la tarjeta de control y fuente**

**Fuente: (Geetech Wiki, 2014)**

Existen leds indicadores en cada dispositivo, tanto en el maestro como en el esclavo, que tienen un significado específico descrito en la Tabla 22.

**Tabla 22**

**Comportamiento de las LEDS indicadores**

**Fuente: (Geetech Wiki, 2014)**

Indicador	Acción
<b>LED verde prendido y parpadeando</b>	Buscando receptor de radio
<b>LED verde prendido sin parpadeo</b>	Enlazando receptor de radio
<b>LED rojo prendido y parpadeando</b>	Transmitiendo datos
<b>LED rojo prendido y sin parpadeo</b>	Actualización de firmware

### 3.7 DISEÑO DE LA INTERFAZ HOMBRE MÁQUINA (HMI)

Por sus siglas en inglés una *Human Machine Interface* (HMI), es una interfaz humano-máquina utilizada para la comunicación entre el ser humano y cualquier tipo de máquina o proceso industrial. En la ISO 9241-110 una interfaz de usuario se define como todas las partes de un sistema interactivo. (Ponsa & Granollers). Una HMI tiene como funciones además de la comunicación, reducir el desgaste mental de un operador y facilitar su control en cualquier tipo de proceso.

### 3.7.1 METODOLOGÍA EN EL DESARROLLO DEL PROYECTO

La metodología de desarrollo del proyecto se basó en la Guía Ergonómica de Diseño para Interfaces de Supervisión (GEDIS), utilizada para el diseño del diseño de interfaces de usuario, el desarrollo de la ésta comprende diez pasos organizados en los niveles indicados en la Figura 76.

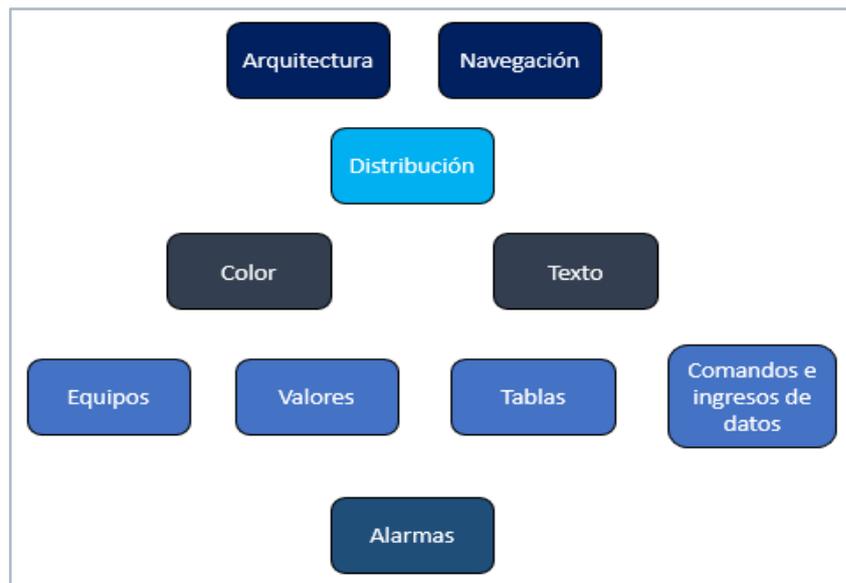


Figura 76. Metodología GEDIS

Fuente: (Pere & Granollers)

#### 3.7.1.1 ARQUITECTURA

La pantalla que será utilizada en la HMI se reducirá a una pantalla de comandos, debido a que solo se necesita controlar los movimientos del robot, estos movimientos pueden ser preestablecidos o definidos por las coordenadas que serán ingresadas. Existirán dos cuadros de diálogo que se desplegarán para informar los autores del proyecto y las limitaciones de los valores de las coordenadas.

### 3.7.1.2 NAVEGACIÓN

Los botones utilizados son totalmente intuitivos comenzando desde la posición inicial del robot que se representa como “Inicial” y “Arriba”, el botón para su activación o desactivación “Activo”, facilitando el aprendizaje del operador con un modelo intuitivo. Además son lo suficientemente grandes para que el operador pueda observar claramente sus títulos y están agrupados dependiendo su funcionalidad similar. En la Figura 77 se puede observar los botones organizados.

### 3.7.1.3 DISTRIBUCIÓN

Los botones más importantes que podrán activar o desactivar la interfaz se ubicarán en la parte superior, en lugar de un mímico del proceso será implementado un simulador en una ventana independiente. En la parte inferior se localizarán los comandos de movimiento, tanto para ingresar coordenadas como para seleccionar movimientos preestablecidos y además, se colocará un cuadro de texto que indique mediante mensajes paso a paso cada movimiento del usuario para que exista un conocimiento de las acciones realizadas desde que se ejecutó la interfaz. La Figura 77 muestra todos los grupos mencionados.

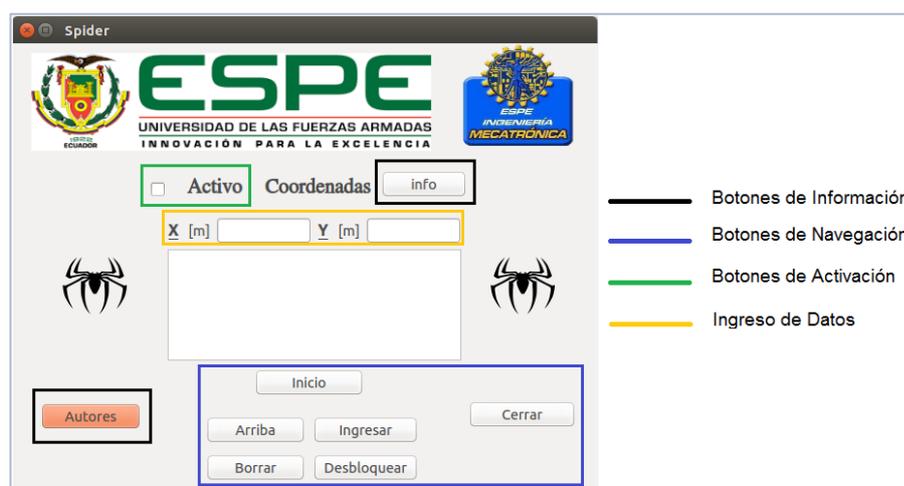


Figura 77. Distribución de botones del HMI

### 3.7.1.4 USO DEL COLOR

El uso del color se enfocó en los tonos que normalmente se utilizan para tableros de control industrial mostrados en la Tabla 23.

**Tabla 23**

**Colores utilizados en la interfaz**

Tarea	Color
Habilitar/Deshabilitar	
Movimientos	
Error de Ingreso/Borrar	

### 3.7.1.5 INFORMACIÓN TEXTUAL

En los botones, etiquetas y en el cuadro de diálogo se utilizó el tipo de letra Calibri para una fácil lectura de la interfaz y el tamaño de letra se estableció en 12 para el texto normal, y 16 para las etiquetas más representativas.

### 3.7.1.6 COMANDO DE DATOS

La interfaz diseñada se centra en una plantilla de comandos, por lo que casi todos los elementos que pertenecen a la misma, son órdenes de movimiento o ingreso de datos para la traslación o rotación del robot. Los elementos insertados son fácilmente entendibles gracias a los nombres establecidos para cada botón, etiquetas en los cuadros de texto e imágenes. Las tareas de cada botón y cuadro de texto se muestran en la Tabla 24.

**Tabla 24**

**Comandos de la HMI**

Función	Elemento
<b>Habilitar</b>	Activo, Ingresar, Desbloquear
<b>Ingresar datos</b>	Cuadros de texto (X,Y)
<b>Borrar datos</b>	Borrar
<b>Movimientos preestablecidos</b>	Inicio, Arriba
<b>Información</b>	Autores, info.

### **3.7.2 SOFTWARE QT CREATOR EN EL DESARROLLO DE LA INTERFAZ**

En el diseño de la interfaz gráfica se utilizó un programa con licencia libre llamado *Qt Creator*, el cual tiene varias facilidades para realizar entornos gráficos, permite diseñar desde los espacios en blanco hasta el acceso de botones y cuadros de texto mediante teclado con tabulación o teclas combinadas, sin utilización necesaria del *mouse*.

El entorno de *Qt creator*, se almacena en un archivo *.ui* que incluye detalles de la forma de los elementos de la interfaz conocidos como *widgets* y organizadores. Se puede asociar las etiquetas de texto con determinados botones para que exista un acceso directo a ellos con una simple tabulación, lo que se conoce como *buddies*, existe también otra herramienta conocida como *signals and slots* que sirve para delegar a botones determinados funciones pre-programadas.

#### **3.7.2.1 HERRAMIENTAS**

##### **3.7.2.1.1 ETIQUETAS**

Se utilizan para nombrar ciertos botones, insertar imágenes a una interfaz por medio de un archivo fuente que debe estar guardado en la misma carpeta donde se encuentra todo el proyecto. La interfaz está diseñada con algunas etiquetas usadas especialmente para insertar imágenes.

##### **3.7.2.1.2 CUADROS DE TEXTO**

Un cuadro de texto se encuentra en la sección *Input Widgets*, en donde se escogieron dos elementos: *Line Edit* que fue utilizada para ingresar las coordenadas que va a desarrollar el robot y *Text Edit* utilizado como medio para que haya una comunicación interactiva con el usuario dependiendo el texto que se ingresó.

### 3.7.2.1.3 BOTONES

Los botones se encuentran en el grupo llamado *Buttons*, el cual posee dos de los elementos utilizados en la interfaz *Push Button* y *Check Box*. Se puede incluir imágenes dentro de un botón así como también manejar sus dimensiones.

Los botones son una parte muy importante de este diseño y desempeñan las funciones específicas mostradas en la Tabla 25.

**Tabla 25**

**Funciones de los botones de la interfaz**

<b>Tipo</b>	<b>Nombre</b>	<b>Tarea</b>
<b>Check Button</b>	Activo	Habilitar o no la publicación de todos los botones e información ingresada.
	Info	Desplegar información relevante del funcionamiento del robot.
<b>Push Button</b>	Autores	Desplegar información acerca de los autores.
	Inicio	Establecer posición inicial antes de comenzar cualquier movimiento.
	Desbloquear	Habilita el accionamiento de las posiciones iniciales.
	Arriba	Establecer posición en la que el robot se eleva del suelo y se sostiene en sus patas.
	Borrar	Eliminar los valores ya ingresados o los que se van a ingresar.
	Ingresar	Establecer valores ingresados en el nodo de control de la HMI.
	Cerrar	Cerrar la ventana de la interfaz

### 3.7.2.1.4 ORGANIZADORES

Del grupo clasificado como Layouts, se seleccionaron 2 tipos: “Horizontal Layout” que fue utilizado para organizar los botones y el “Grid Layout” que fue utilizado para organizar los cuadros de texto de una manera más ergonómica y ordenada.

### **3.7.3 MODELACIÓN Y SIMULACIÓN**

#### **3.7.3.1 MODELADO URDF**

Por medio de archivos URDF es posible simular todas las características de un robot, incluso establecer parámetros físicos reales como fricción, torque, gravedad entre otros.

En un archivo URDF se puede establecer dos modelos para la simulación, el visual que se utiliza para controlar y verificar movimientos, y el de colisión que se utiliza para simulaciones que involucren procesos más específicos como evasión de objetos, configurados en el mismo archivo. En el proyecto realizado solo se utilizó el modelo visual ya que el fin de la simulación es emular el correcto posicionamiento de las patas para el desplazamiento del robot dependiendo la rotación o traslación deseada.

##### **3.7.3.1.1 ESTRUCTURA DE UN URDF**

Un archivo URDF define dos partes esenciales de un robot, el *link* o eslabón y el *joint* o unión, cada uno tiene su propia configuración interna. Al inicio de cualquier modelo se debe establecer el link de referencia de toda la estructura, en el caso del proyecto será llamado “c\_ar”. El lenguaje XML que sirve para desarrollar los archivos URDF, se guía por medio de la sangría de cada etiqueta para compilar. Las unidades con las que se trabaja son del S.I.

##### **3.7.3.1.2 ESTRUCTURA DE UN LINK**

La estructura y características de la configuración de un *link* se pueden observar en la Figura 78.

```

<link name="base_link">
  <visual>
    <geometry>
      <box size="0.2 .3 .1"/>
    </geometry>
    <origin rpy="0 0 0" xyz="0 0 0.05"/>
    <material name="white">
      <color rgba="1 1 1 1"/>
    </material>
  </visual>
</link>

```

**Figura 78. Configuración de un link**

**Fuente: (Martínez & Fernández, 2013)**

Cada etiqueta tiene las funciones descritas en ROS.org (2017), detalladas a continuación:

- “*link name*”, especifica el nombre del *link*, y si es el inicial, también especifica el eslabón de referencia.
- “*geometry*”, establece la figura geométrica que tiene el link, pero debido a que solo se puede insertar formas básicas, las más complejas se deben definir dentro de un archivo *mesh*.
- “*mesh*”, en éste se escribe la dirección en donde está guardado el código fuente que define una pieza con forma compleja.
- “*origin*”, establece el punto en el que se ubica dicho *link* con respecto al origen de coordenadas ingresando los valores en *XYZ* y también se puede definir las rotaciones respecto a los ejes coordenados *X*, *Y* e *Z* respectivamente con *RPY*.
- “*material*”, define la textura y el color de un link, se puede configurar cambiando los 4 valores RGBA de 0 a 1.

### 3.7.3.1.3 ESTRUCTURA DE UNA UNIÓN

La configuración y características de una unión se observan en la Figura 79.

```

<joint name="c_ar_to_tuerca11" type="revolute">
  <axis rpy="0 0 0" xyz="0 0 1"/>
  <limit effort="1000.0" lower="-1.57" upper="1.57" velocity="1"/>
  <parent link="c_ar"/>
  <child link="tuerca11"/>
  <origin xyz="0.078397 0.078397 0.1278"/>
</joint>

```

**Figura 79. Configuración de una unión**

Cada etiqueta tiene las funciones descritas en ROS.org (2017), detalladas a continuación:

- “*joint name*”, establece el nombre y el tipo de unión entre dos *links*, los mismos que dependen del movimiento que se requiera.
- El parámetro “*type*”, puede tener los tipos de uniones presentadas en la Tabla 26.
- “*axis*”, especifica el eje referente al cual se moverá el link dependiendo el tipo de unión configurada.
- “*limit*”, los límites de giro superior e inferior y el establecimiento de la velocidad son necesarios sobre todo cuando se configura la unión de revolución, ya que si no se especifican estos valores, el movimiento no se llevará a cabo en la simulación. El límite de esfuerzo dependerá de las condiciones que se desee simular.
- “*parent link*”, establece el nombre del eslabón de referencia.
- “*child link*”, establece el nombre del eslabón móvil. Los nombres que sean establecidos en las etiquetas “*parent link*” y “*child link*” deben ser iguales a los escritos en la pestaña “*link name*” dependiendo cuales eslabones se deseen unir.
- “*origin*”, se establece las coordenadas con *XYZ* en donde se ubicará el eslabón móvil con respecto a la ubicación del eslabón de referencia. Se puede establecer rotaciones del eslabón móvil con respecto al eslabón de referencia con *RPY*.

Existen otras características que no fueron usadas para el proyecto, ya que las mencionadas son suficientes para poder simular los movimientos de rotación y traslación.

Tabla 26

**Tipos de uniones**

Tipo	Descripción
<b>fixed</b>	No permite movimiento de rotación ni traslación en los 6 grados de libertad.
<b>revolute</b>	Giro respecto a un eje especificado con límites que se deben establecer.
<b>continuous</b>	Giro respecto a un eje especificado que no tiene límites.
<b>floating</b>	Permite el movimiento de los 6 grados de libertad.
<b>planar</b>	Traslación respecto a un plano perpendicular a un eje especificado.

### 3.7.3.2 SIMULACIÓN EN EL SOFTWARE RVIZ

El simulador conocido como RVIZ (ver Figura 80), se instala automáticamente al mismo tiempo que la versión de ROS elegida. Este simulador tiene varias opciones para establecer características de un robot en la vida real como fricción, inercia, gravedad, entre otras (ROS.org, 2017).

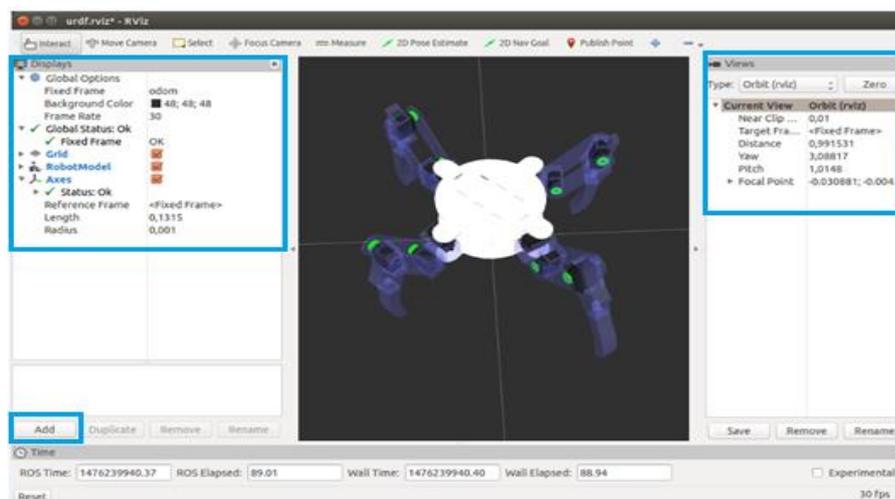


Figura 80. Simulador RVIZ

Entre las características que más resaltan, en la parte izquierda se encuentra el menú llamado *Displays* donde podemos agregar opciones y configuraciones del simulador pulsando el botón *Add*. En el presente proyecto

se utilizó las opciones *Grid*, *RobotModel* y *Axes*. En la parte derecha, el menú llamado *Viewers* muestra las características de la vista 3D que podemos manipular con el *mouse* para girar la cámara respecto a los 3 ejes y acercar o alejar la vista.

### 3.7.3.2.1 HERRAMIENTAS DEL SOFTWARE RVIZ UTILIZADAS

#### 3.7.3.2.1.1 AXES

Es una opción para insertar los 3 ejes coordenados  $X$ ,  $Y$ ,  $Z$  de color verde, rojo y azul, respectivamente; se utiliza para poder observar el origen de los ejes coordenados y tomar como referencia los movimientos de cualquier robot ya sea móvil o estático. Se puede modificar la longitud y el radio de los cilindros que forman los 3 ejes (ROS.org, 2017).

#### 3.7.3.2.1.2 ROBOT MODEL

Esta opción mostrada en la Figura 81, es indispensable para cargar cualquier tipo de modelo virtual de un robot en el simulador, tiene varias opciones muy importantes desde verificar si el modelo cargado está correctamente configurado, hasta poder observar de qué forma está estructurado cada *link* y *joint* (ROS.org, 2017). El modelo virtual debe estar almacenado como un archivo con extensión *.urdf*. Brinda opciones muy útiles al momento del movimiento del robot ya que puede plasmar gráficamente la trayectoria que tome cualquiera de los elementos que aparecen en la pantalla a medida que se vaya realizando el movimiento. Ver Figura 81.



Figura 81. Modelo del Robot RVIZ

Las funciones más relevantes utilizadas de esta herramienta se describen en la Tabla 27.

**Tabla 27**  
**Funciones de *Robot Model***

Nombre	Descripción
<b><i>Visual Enabled</i></b>	Habilita el modelo visual de robot
<b><i>Collision Enabled</i></b>	Habilita el modelo de colisión del robot
<b><i>Update Interval</i></b>	Tiempo de actualización de posición de cada <i>link</i> en segundos.
<b><i>Alpha</i></b>	La cantidad de transparencia aplicada a cada <i>link</i>
<b><i>Robot Description</i></b>	La configuración del robot establecida en el archivo urdf
<b><i>TF Prefix</i></b>	Establece los prefijos de cada cadena cinemática
<b><i>Links</i></b>	La estructura del robot en el archivo urdf

### 3.7.3.3 COMPROBACIÓN DEL MODELO URDF

#### 3.7.3.3.1 MODELO EN EL SIMULADOR

Para visualizar el modelo en el simulador y controlarlo se incluyó ciertas líneas de código en el archivo *spider.launch*, el cual ejecuta todos los nodos necesarios. Ver Figura 82.

```
<arg name="model" />
<arg name="gui" default="True" />
<param name="robot_description" textfile="$(find tesis)/urdf/araniam.urdf" />
<param name="use_gui" value="$(arg gui)"/>
```

**Figura 82. Comando para cargar el modelo URDF al simulador**

Los parámetros que se establecieron en las pestañas de *<arg name>* son: el nombre de la aplicación y la presentación del modelo urdf al momento de ejecutar el archivo *launch*.

En las pestañas *<param name>* se establece la descripción del robot siendo un parámetro definido en el simulador, el directorio donde se encuentra el código fuente del modelo urdf y su ejecución conjuntamente con el simulador descrito en la Figura 83.

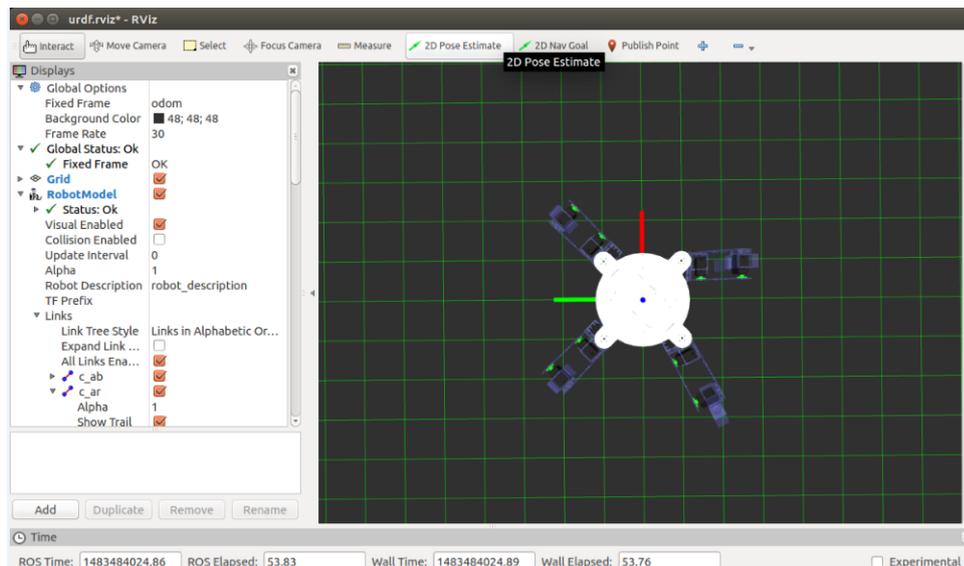
```
<node name="robot_state_publisher" pkg="robot_state_publisher" type="state_publisher" />
<node name="rviz" pkg="rviz" type="rviz" args="-d $(find tesis)/urdf.rviz" />
```

**Figura 83. Ejecución de nodos para el simulador**

Los parámetros finales en la pestaña *<node>* son los nodos que se van a ejecutar, el uno es propiamente el simulador RVIZ, y el otro es el nodo que va a publicar y controlar constantemente las posiciones de cada unión móvil del robot virtual, establecidas anteriormente ya sea con un ángulo o con una longitud.

### 3.7.3.3.2 SIMULACIÓN

Cada vez que el robot realiza una acción, el simulador grafica su trayectoria en la cuadrícula que emula el nivel de referencia. Los movimientos principalmente se componen de la posición inicial (ver Figura 84), el movimiento de traslación hacia adelante (ver Figura 85), el movimiento de rotación hacia la derecha (ver Figura 86), y el movimiento de rotación hacia la izquierda (ver Figura 87).



**Figura 84. Posición inicial.**

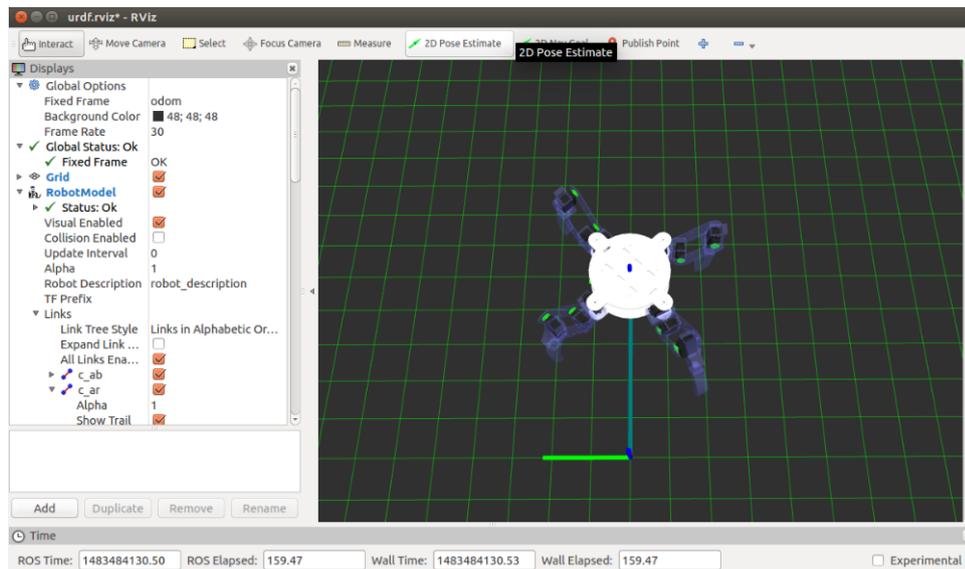


Figura 85. Traslación hacia adelante.

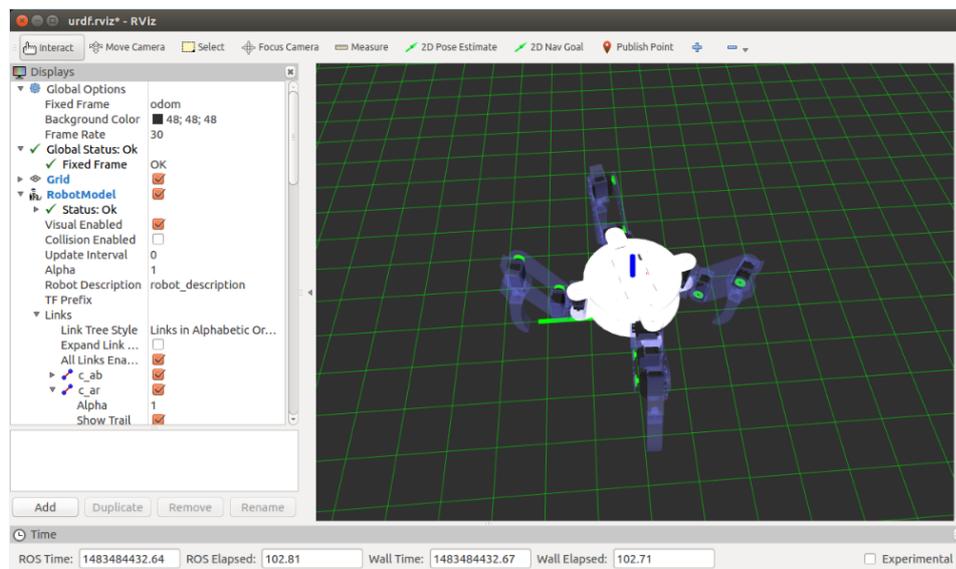
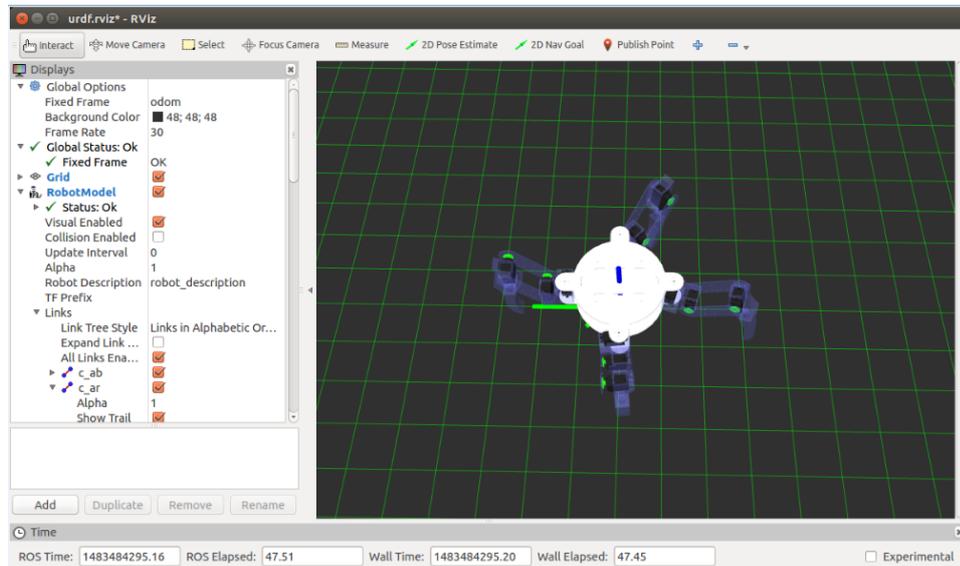
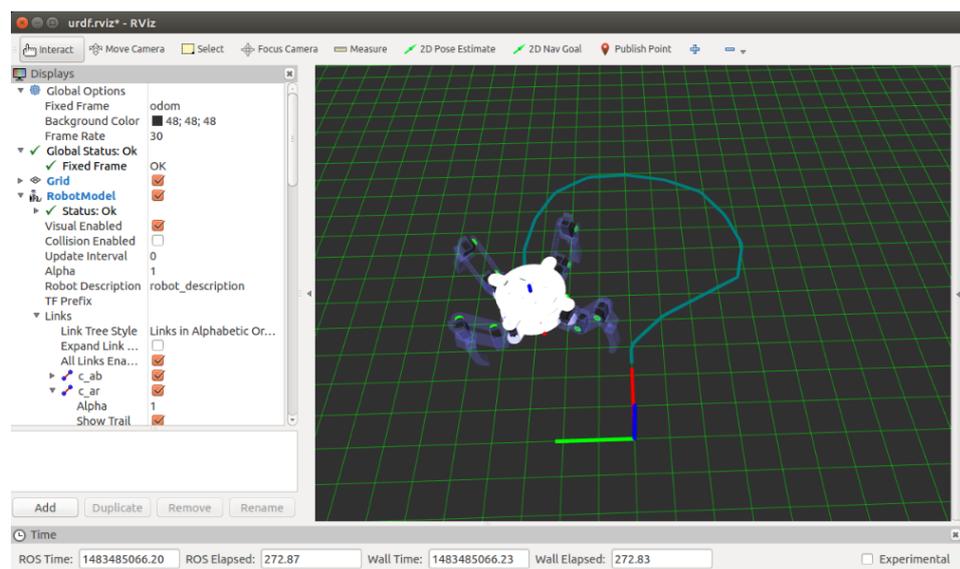


Figura 86. Rotación hacia la derecha.



**Figura 87. Rotación hacia la izquierda.**

La finalidad de los movimientos independientes es formar parte de una trayectoria determinada ya sea con coordenadas que toman como referencia el origen de partida, ingresadas por medio de la interfaz gráfica; o movimientos individuales ingresados por el *joystick*. Una trayectoria aleatoria se muestra en la Figura 88, en donde se observa que se puede generar cualquier forma dependiendo los movimientos.



**Figura 88. Trayectoria aleatoria.**

## CAPÍTULO 4. CONSTRUCCIÓN E IMPLEMENTACIÓN

### 4.1 IMPLEMENTACIÓN DE LA ESTRUCTURA Y COMPONENTES MECÁNICOS

Las piezas que conforman la estructura del robot fueron elaboradas mediante los procesos y materiales, detallados en la Tabla 28.

Tabla 28

Material y operación de las piezas que conforman la estructura del robot

No.	Pieza	Material	Operación
1	cuerpo pieza superior	Acrílico	Corte láser
2	cuerpo pieza inferior	Acrílico	Corte láser
3	<i>bracket</i> servo coxa 1	ABS	Impresión 3D
4	<i>bracket</i> servo coxa 2	ABS	Impresión 3D
5	<i>bracket</i> servo fémur 1	Acrílico	Corte láser
6	<i>bracket</i> servo fémur 2	Acrílico	Corte láser
7	<i>bracket</i> servo tibia	Acrílico	Corte láser
8	Soporte servo tibia	ABS	Impresión 3D
9	case batería 1	Acrílico	Corte láser
10	case batería 2	Acrílico	Corte láser
11	case batería 3	Acrílico	Corte láser
12	case batería 4	Acrílico	Corte láser

Como se puede observar en la Tabla 28, las únicas piezas elaboradas con ABS fueron los *brackets* de los servos que constituyen el primer y segundo grado de libertad, debido a la complejidad de su forma; las demás piezas incluidas el case de la batería, fueron elaboradas mediante acrílico. De esta manera los tiempos de elaboración de cada pieza de acuerdo a los programas utilizados para la cortadora láser y la impresora 3D Prusa, se exponen a continuación.

#### 4.1.1 TIEMPOS DE CORTE E IMPRESIÓN

En la Figura 89, Figura 90, Figura 91, Figura 92, Figura 93, Figura 94, Figura 95 y Figura 96 se detallan los tiempos de corte e impresión de cada pieza, de acuerdo al orden mostrado en la Tabla 28.

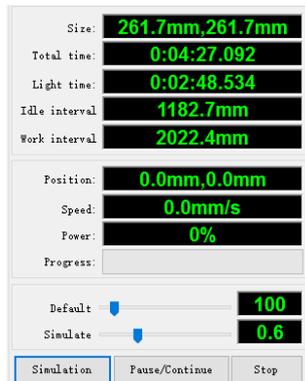


Figura 89. Tiempo de corte de la pieza: cuerpo pieza superior

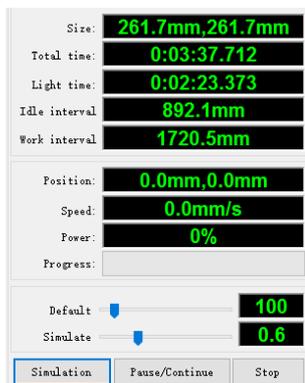


Figura 90. Tiempo de corte de la pieza: cuerpo pieza inferior

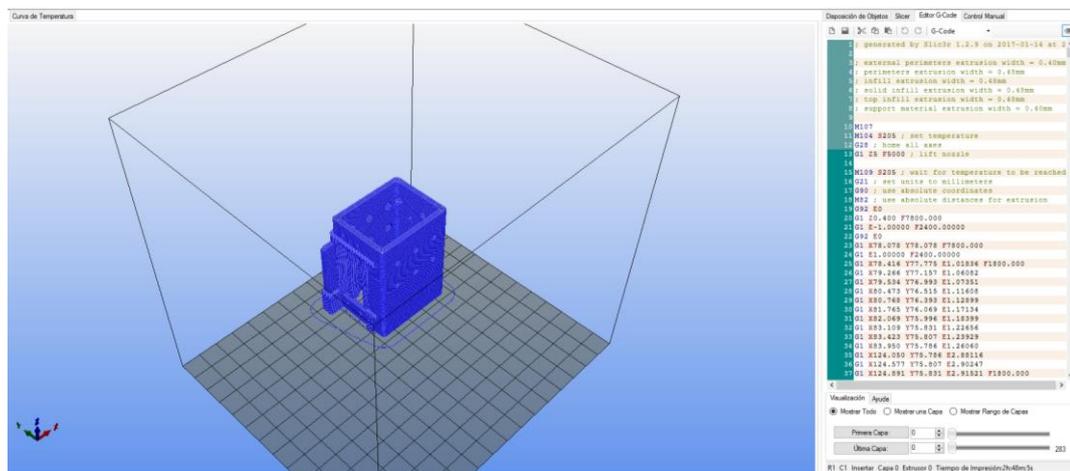


Figura 91. Tiempo de impresión de la pieza: *bracket* servo coxa 1

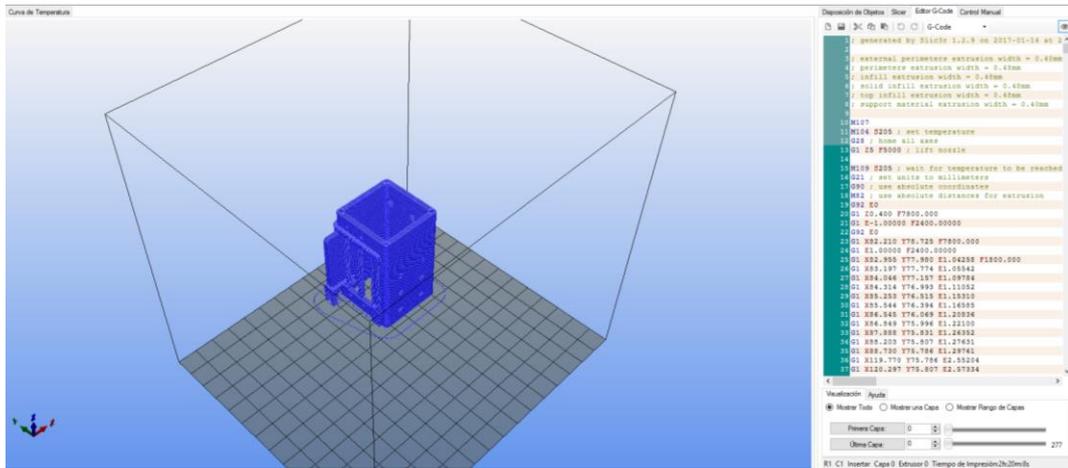


Figura 92. Tiempo de impresión de la pieza: *bracket servo coxa 2*

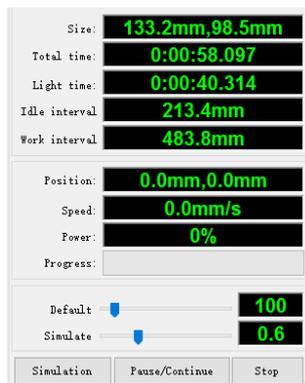


Figura 93. Tiempo de corte de la pieza: *bracket servo fémur 1*

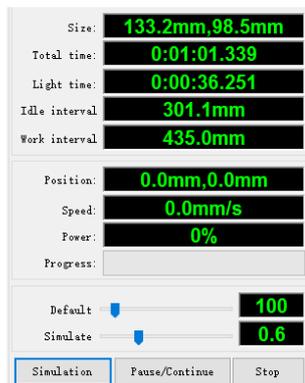


Figura 94. Tiempo de corte de la pieza: *bracket servo fémur 1*

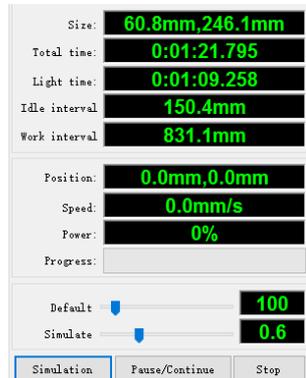


Figura 95. Tiempo de corte de la pieza: *bracket* servo tibia

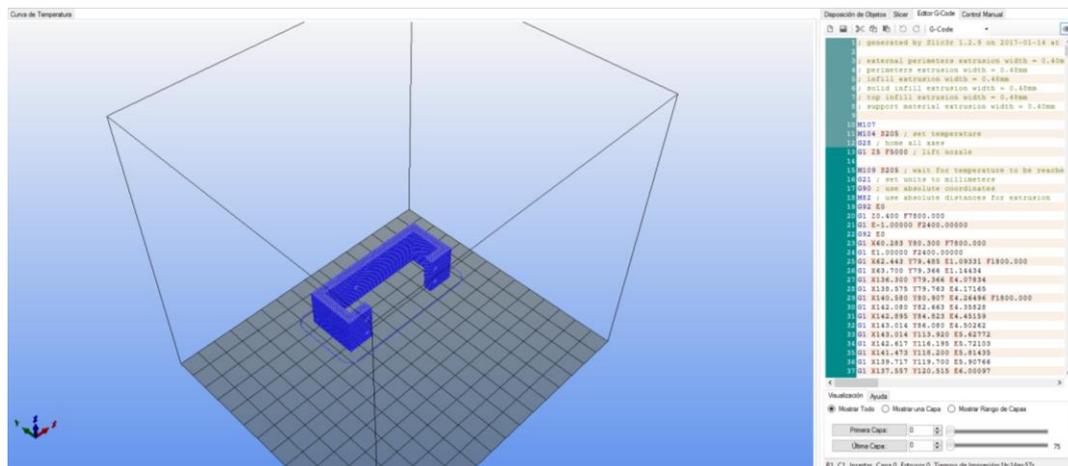
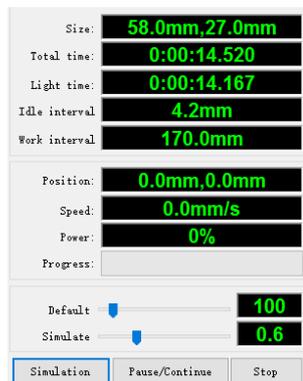


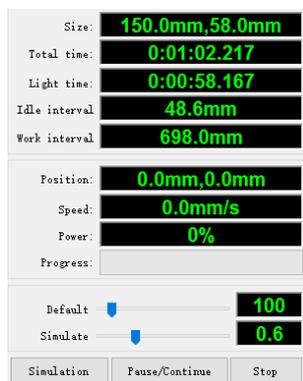
Figura 96. Tiempo de impresión de la pieza: soporte servo tibia



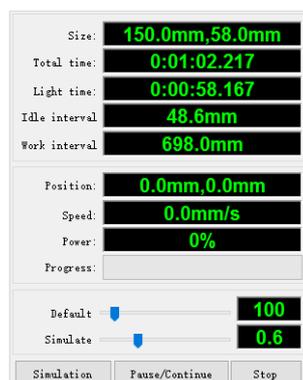
Figura 97. Tiempo de corte de la pieza: case batería 1



**Figura 98. Tiempo de corte de la pieza: case batería 2**



**Figura 99. Tiempo de corte de la pieza: case batería 3**



**Figura 100. Tiempo de corte de la pieza: case batería 4**

De acuerdo a la Figura 89, Figura 90, Figura 91, Figura 92, Figura 93, Figura 94, Figura 95, Figura 96, Figura 97, Figura 98, Figura 99 y Figura 100; se conforma la Tabla 29 correspondiente a los tiempos determinados por los programas para cada pieza.

Tabla 29

Tiempos de operaciones de elaboración de las piezas de la estructura del robot

No.	Pieza	Tiempo de elaboración
1	cuerpo pieza superior	4min27.092s
2	cuerpo pieza inferior	3min37.712s
3	<i>bracket</i> servo coxa 1	2h48m5s
4	<i>bracket</i> servo coxa 2	2h20min8s
5	<i>bracket</i> servo fémur 1	58.097s
6	<i>bracket</i> servo fémur 2	1min1.339s
7	<i>bracket</i> servo tibia	1min21.795s
8	Soporte servo tibia	1h14min57s
9	case batería 1	32.354s
10	case batería 2	14.520s
11	case batería 3	1min2.217s
12	case batería 4	22.028s

#### 4.1.2 TIEMPOS DE ELABORACIÓN

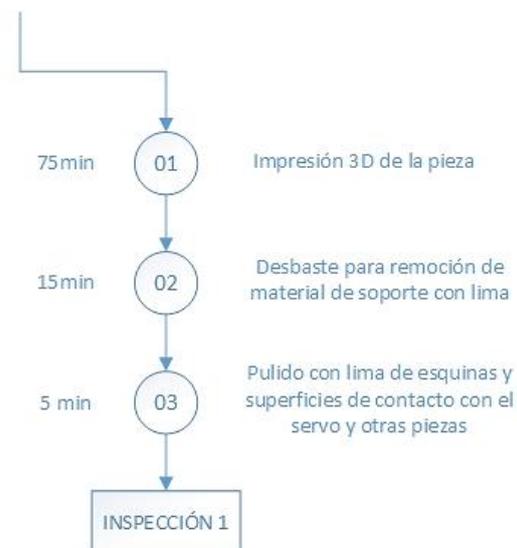
Para las piezas elaboradas en acrílico no es necesario realizar otro proceso. Sin embargo para las piezas impresas en ABS es preciso realizar más procesos para completar su elaboración, teniendo los diagramas detallados en la Figura 101, Figura 102 y Figura 103.



Figura 101. Diagrama de proceso de bracket servo coxa 1



**Figura 102. Diagrama de proceso de bracket servo coxa 2**



**Figura 103. Diagrama de proceso de soporte servo tibia**

Los tiempos de fabricación para cada pieza se detallan en la Tabla 30 con el proceso correspondiente para cada una. Para los procesos de desbaste y pulido se utilizó limas predeterminadas para cada función.

**Tabla 30****Tiempos de fabricación de cada pieza del robot**

Pieza	#	Proceso				Total (min)
		Corte Láser	Impresión 3D	Desbaste	Pulido	
cuerpo pieza superior	1	4.5	-	-	-	4.5
cuerpo pieza inferior	1	3.5	-	-	-	3.5
<i>bracket</i> servo coxa 1	4	-	168	20	5	772
<i>bracket</i> servo coxa 2	4	-	140	20	5	660
<i>bracket</i> servo fémur 1	4	1.0	-	-	-	4
<i>bracket</i> servo fémur 2	4	1.0	-	-	-	4
<i>bracket</i> servo tibia	4	1.5	-	-	-	6
Soporte servo tibia	4	-	75	15	5	380
case batería 1	1	0.5	-	-	-	0.5
case batería 2	1	0.2	-	-	-	0.2
case batería 3	1	1.0	-	-	-	1.0
case batería 4	1	0.4	-	-	-	0.4
<b>Total (min)</b>		<b>24.1</b>	<b>1532</b>	<b>220</b>	<b>60</b>	<b>1836.1</b>

Por lo tanto según la Tabla 30, se tiene un tiempo total de fabricación de 1836.1 minutos, correspondiente a 30 horas 36 minutos y 6 segundos.

El accesorio de los servomotores que tiene la funcionalidad de sujetar el actuador al bracket del robot, viene sin hoyos para la colocación de los tornillos, por lo que también se realizó este proceso con un tiempo total detallado en la Tabla 31.

**Tabla 31****Tiempo de taladrado de la pieza bracket servo accesorio**

Pieza	#	Agujeros	Proceso	Total
			Taladrado	(min)
<b>Bracket servo accesorio</b>	12	4	0.5	24

De las tablas Tabla 30 y Tabla 31 se obtiene un tiempo total de elaboración de las piezas de 31 horas y 6 segundos.

### 4.1.3 TIEMPOS DE MONTAJE

En la tabla se explica el tiempo de montaje para cada parte biológica del robot: coxa, fémur y tibia; y también sus actuadores.

Tabla 32

Tiempos de montaje de las partes del robot

Parte del robot	#	Tiempo(min)	Total(min)
<b>Coxa</b>	4	20	80
<b>Fémur</b>	4	20	80
<b>Tibia</b>	4	5	20
<b>Cuerpo</b>	1	30	30
<b>Servo motores</b>	12	20	240
<b>Total</b>			450

De esta manera se tiene que el tiempo de montaje total de las piezas que conforman el robot es de 7 horas y 30 minutos.

## 4.2 IMPLEMENTACIÓN DEL SISTEMA DE POTENCIA

La tarjeta electrónica requiere elementos que enlacen dispositivos delicados especialmente servomotores, por lo que se hace indispensable el uso de *molex* adecuados para asegurar una conexión con la polarización correcta de la batería y servos hacia la placa. Así mismo, sus protecciones requieren un acople adecuado, para ello se utilizaron porta fusibles que brinden la opción de intercambiar este elemento.

Se agregaron igualmente diodos que permitan conducir la cantidad necesaria de amperaje para evitar cambios en el sentido de la corriente y que consuman un pequeño voltaje para asegurar que la energía entregada a cada servomotor sea la idónea. La totalidad de los elementos utilizados se muestran en la Tabla 33 con sus valores técnicos.

**Tabla 33**

**Lista de elementos de la placa de potencia.**

<b>Elemento</b>	<b>Medida</b>	<b>Cantidad</b>
Motor Tower PRO MG958	4 a 6.6v	12
Porta fusibles	-	12
Fusibles	1.5 A / 250v	12
Resistencia	47 $\Omega$ / ½ W	12
Capacitor cerámico	0.47uF / 250v	12
Switch batería de LIPO	6A / 125v	1
Batería de LIPO	5000 mAh / 7.4v	1
Diodo 1N4007	1A / 700V	12
Diodo 6A10	6A /100v	12
Molex 3 pines	-	12
Molex 2 pines	-	1

#### 4.2.1 TIEMPO DE ELABORACIÓN DE PUNTOS DE SOLDADURA

Teniendo en cuenta un tiempo estimado de 45 segundos por cada punto de soldadura, se elabora la Tabla 34 con el número de puntos y el tiempo promedio de realización.

**Tabla 34**

**Tiempos de elaboración de puntos de soldadura de la placa de potencia**

<b>Capa de la placa</b>	<b>Número de puntos</b>	<b>Total(min)</b>
Capa superior	88	66
Capa inferior	92	69
<b>Total</b>	180	135

De acuerdo a la Tabla 34, se tiene un total de elaboración de la placa de potencia de 2 horas y 16 minutos.

#### 4.3 IMPLEMENTACIÓN DEL SISTEMA DE CONTROL

La placa de control está enfocada en el acoplamiento de la tarjeta Arduino Mega 2650. Para su alimentación es necesario incluir un conversor DC-DC que regula el voltaje entregado por una batería alcalina, además se necesita implementar un pequeño sistema de reseteo con un pulsador y resistencia, y finalmente se requiere unir las salidas de PWM mediante un conector de cable plano. Los elementos utilizados se encuentran en la Tabla 35.

Tabla 35

Lista de elementos de la placa de control.

Elemento	Especificación	Cantidad
Convertor DC-DC	20 a 5v	1
Pulsador	50 mA / 12v	1
Resistencia	220 $\Omega$ / $\frac{1}{4}$ W	1
Arduino MEGA 2560	7 a 12v	1
Switch batería alcalina	250 mA / 9v	1
Conector batería alcalina	250 mA / 9 a 12v	1
Batería alcalina	170 mAh / 9v	1
Molex 2 pines	-	2
Cable plano	15 líneas	1
Conector cable plano	15 íneas	2

#### 4.3.1 TIEMPO DE ELABORACIÓN DE PUNTOS DE SOLDADURA

En la placa de control se implementaron 133 puntos de soldadura con un tiempo promedio de 45 segundos por cada punto, por lo que en total se utilizó 1 hora y 40 minutos.

#### 4.4 IMPLEMENTACIÓN DEL SISTEMA DE COMUNICACIÓN

Para la implementación del sistema de comunicación se siguió los siguientes pasos:

- Configuración de nodo serial en ROS.
- Configuración de los dispositivos de comunicación mediante el software *Mission Planner*.
- Configuración de los parámetros de la comunicación en la placa de control.

La configuración de la comunicación en el sistema operativo ROS se realizó mediante el enlace del nodo serial en Python (`serial_node.py`) con el nodo de la interfaz gráfica y el nodo *joystick*,

##### 4.4.1 DIAGRAMA DE COMUNICACIÓN SERIAL NODO PYTHON

El nodo serial de ROS está configurado como se muestra en la Figura 104.

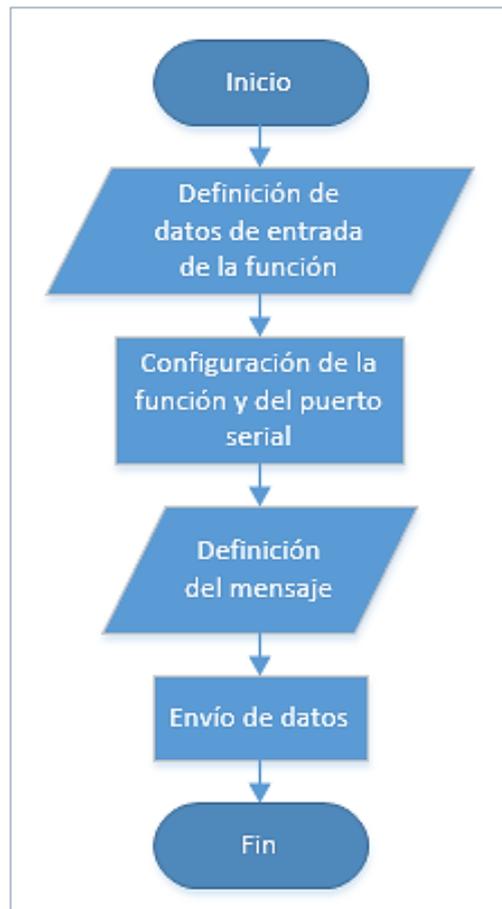


Figura 104. Programación del nodo serial\_node.py

#### 4.4.2 CONFIGURACIÓN DE LOS PARÁMETROS DE COMUNICACIÓN DE LA PLACA DE CONTROL

En la placa de control se realizó la configuración de los parámetros de velocidad de transmisión, bit de inicio y bit de salida, para el reconocimiento de las órdenes enviadas desde ROS mediante el nodo serial de Python.

La velocidad de transmisión sugerida para los dispositivos de comunicación es de hasta 115200 baudios, por lo que se eligió la misma para mayor rapidez.

La configuración del bit de entrada y bit de salida se lo realizó manualmente mediante el reconocimiento de caracteres en la cadena enviada.

## 4.5 PROGRAMACIÓN DEL SISTEMA DE CONTROL Y HMI

### 4.5.1 PROGRAMACIÓN DEL SISTEMA DE CONTROL

El control se lo realizó en lazo abierto debido al tipo de actuadores utilizados, y su diagrama se presenta en la Figura 105.

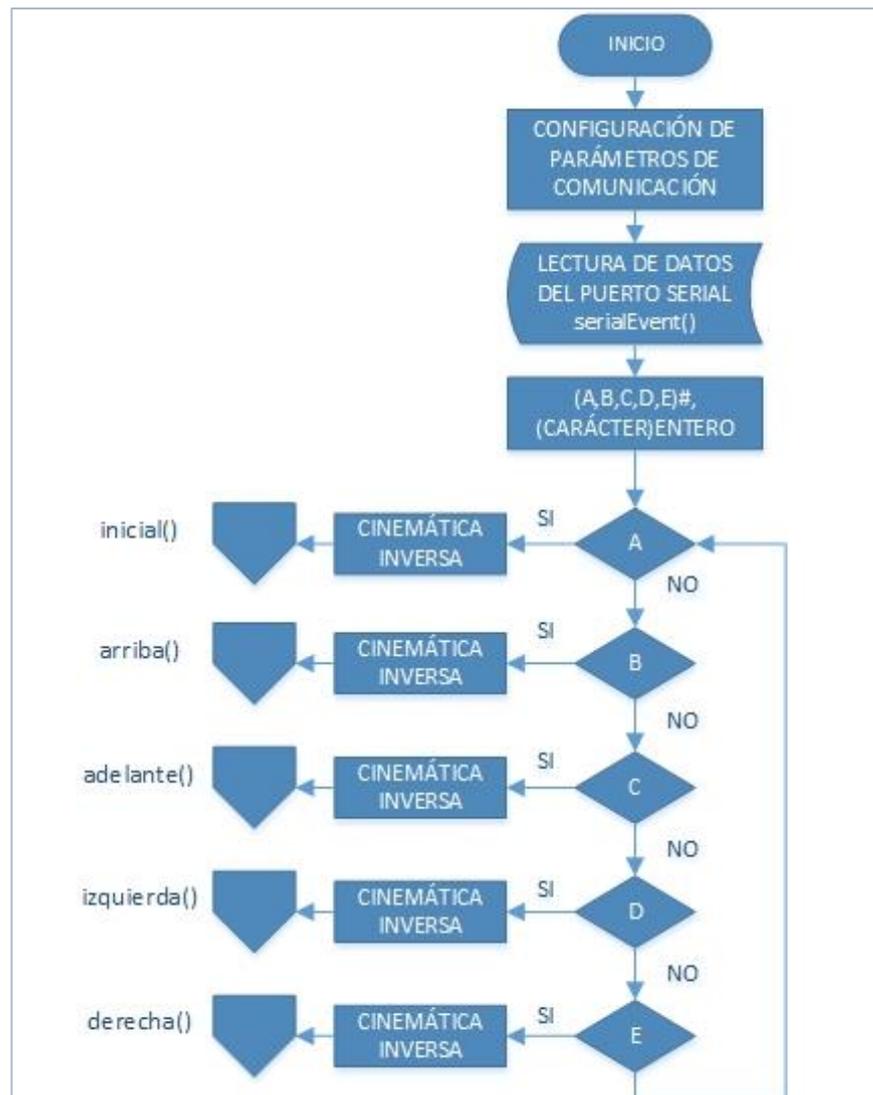
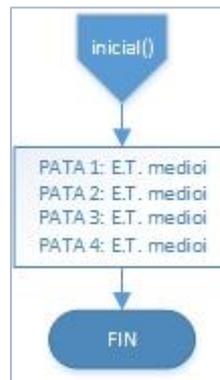
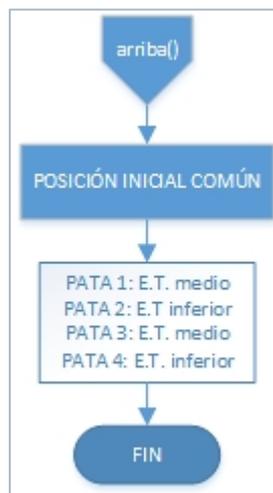


Figura 105. Diagrama de flujo de la programación del sistema de control

Los movimientos inicial y arriba se detallan en la Figura 106 y Figura 107. En dónde E.T. se refiere al espacio de trabajo del robot y “E.T. medio” representa la posición del robot en el espacio de trabajo medio ubicado al nivel del piso. Las secuencias de movimientos restantes fueron detalladas en el modelamiento robótico del proyecto.



**Figura 106. Posición inicial del cuadrúpedo**



**Figura 107. Posición arriba del cuadrúpedo**

#### 4.5.2 PROGRAMACIÓN DEL HMI

En la Figura 108 se puede observar el proceso que tiene que seguir el usuario al utilizar la interfaz gráfica. Es muy importante comenzar seleccionando el botón “Activo”, para que todas las acciones realizadas sean publicadas en el simulador y para la transmisión de datos. Posteriormente se debe seleccionar “Inicio” y “Arriba” para que la araña pueda ubicarse correctamente en la posición inicial. Después de las acciones descritas, se debe escribir las coordenadas deseadas y presionar el botón “Ingresar”.



Una vez realizadas las acciones, el nodo de la interfaz procesará el cuadrante al que debe ir el robot, si las coordenadas ingresadas tienen un ángulo se procede a la rotación y posteriormente los ciclos de traslación necesarios para completar la tarea indicada. Si las coordenadas tienen algún error de teclado, se puede proceder a pulsar el botón “Borrar” para volver a escribir las coordenadas de forma correcta. Todas las acciones realizadas a la par de la simulación enviarán una bandera por medio de comunicación serial para que la placa de control ejecute los movimientos.

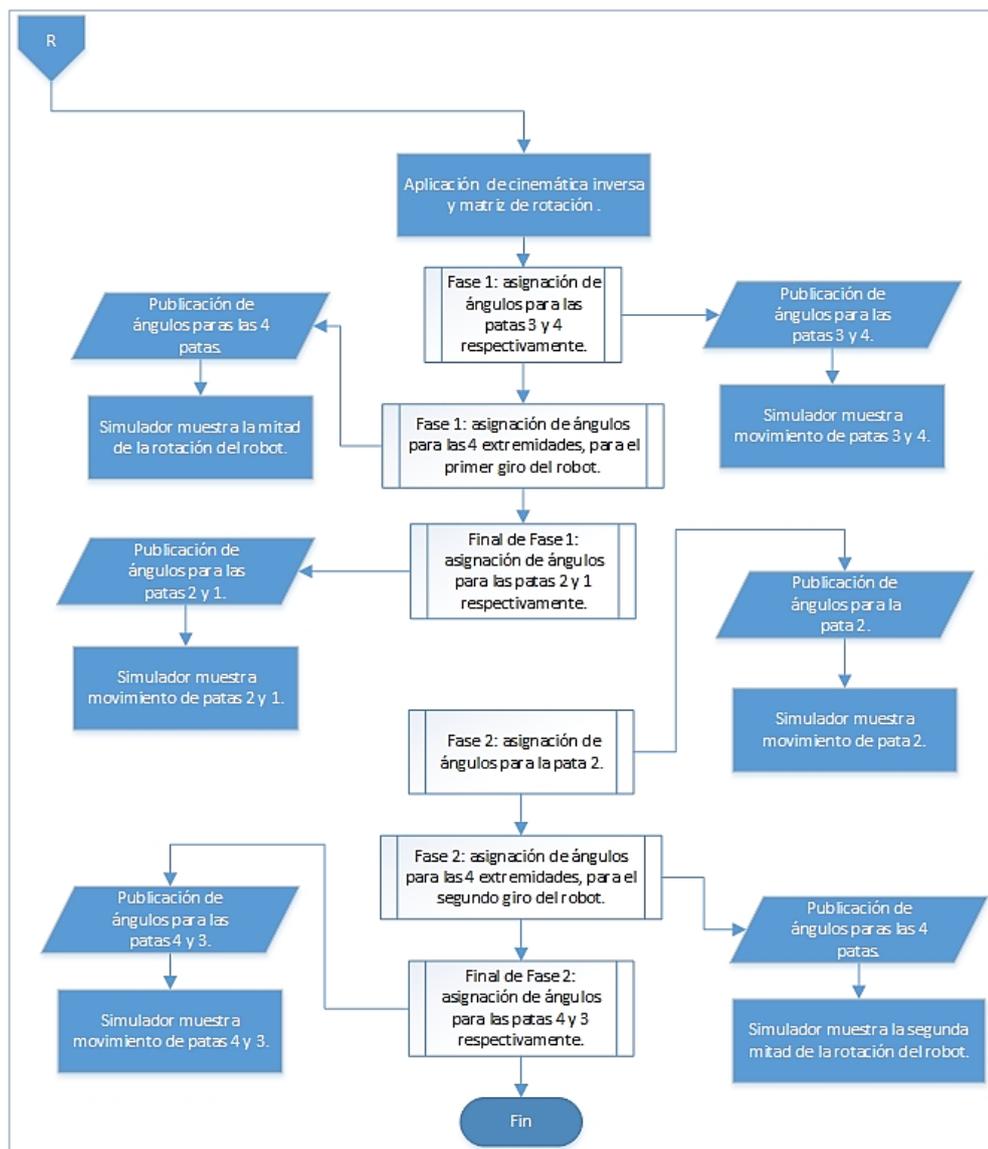
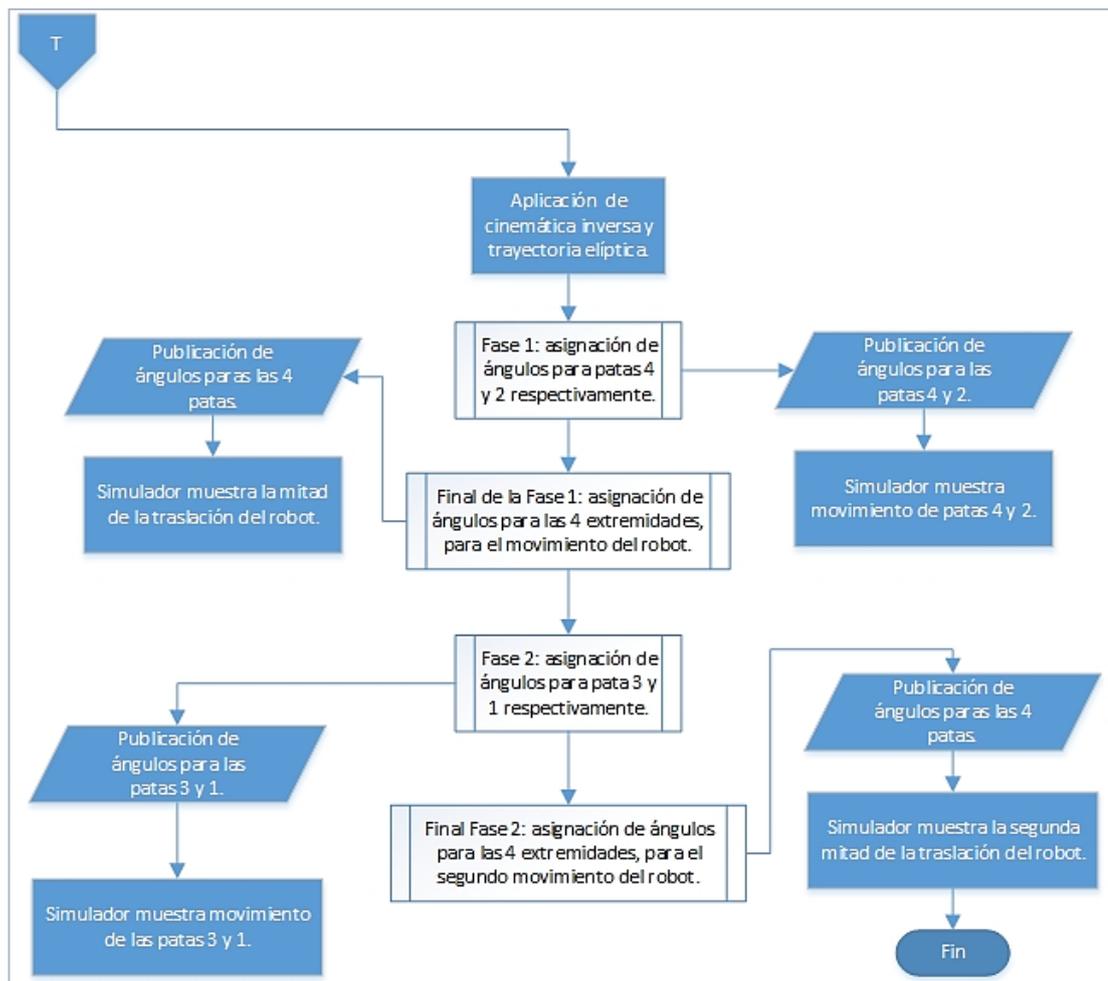


Figura 109. Diagrama de flujo del proceso de rotación

La Figura 109 representa el proceso de rotación dividido en dos fases. Los movimientos son publicados inmediatamente en el simulador y a la par son ejecutados por el robot con un retraso de un segundo. Las acciones descritas se repetirán hasta que el número de iteraciones que fue calculado por el nodo de la interfaz sea cumplido.



**Figura 110. Diagrama de flujo del proceso de traslación**

Finalmente la Figura 110 describe el proceso de traslación. Progresivamente cada acción es publicada en el simulador y enviada serialmente para que se ejecute el número de ciclos necesarios.

## **CAPÍTULO 5. PRUEBAS Y RESULTADOS**

### **5.1 PRUEBAS MECÁNICAS**

#### **5.1.1 PRUEBAS DE LOS SERVOMOTORES**

El objetivo es realizar pruebas del funcionamiento de los servomotores trabajando como unidad y en conjunto en el mecanismo del robot.

##### **5.1.1.1 PROCEDIMIENTO**

Para las pruebas en vacío se realizó el siguiente procedimiento:

- Energizar los servomotores.
- Envío de señal de control para ángulos máximo y mínimo.
- Verificar el funcionamiento de los servomotores.
- Verificar el alcance de los límites: mínimo y máximo.
- Verificar temperatura de los servomotores.

En las pruebas del mecanismo se realizó el siguiente procedimiento:

- Conexión de los 12 servomotores a la placa.
- Energizado de la placa.
- Ejecutar movimientos del robot sobre el suelo.
- Verificar funcionamiento de los servomotores.
- Verificar temperatura de los servomotores.

##### **5.1.1.2 RESULTADOS**

En la Tabla 36 se puede apreciar los parámetros de las pruebas de los motores sin carga. Los alcances del límite máximo y mínimo de giro con la configuración de acuerdo a los PWM especificados por el fabricante no son 0° y 180°, este problema se solucionó al encontrar experimentalmente la configuración de los servomotores para alcanzar dichos ángulos.

**Tabla 36****Resultados de las pruebas de los límites angulares de los servomotores sin carga**

<b>Parámetro</b>	<b>Resultados</b>
Encendido de servomotores	Si
Alcance de límites angulares	No
Sobrecalentamiento presente	No

En la Tabla 37 se detallan los resultados obtenidos luego de la corrección de los límites angulares de los servomotores, alcanzando resultados satisfactorios.

**Tabla 37****Resultados de las pruebas de los límites angulares de los servos corregidos sin carga**

<b>Parámetro</b>	<b>Resultados</b>
Encendido de servomotores	Si
Alcance de límites angulares	Si
Sobrecalentamiento presente	No

Los resultados de las pruebas de los servomotores con carga se muestran en la Tabla 38. El sobrecalentamiento de un motor estuvo presente, para corregir este problema se lo intercambié por otro en una unión que requería un torque menor, que según el análisis dinámico fue la correspondiente al primer grado de libertad. Por otra parte para el pico de corriente inicial se implementaron redes Snubber de uso general.

**Tabla 38****Resultados de la pruebas de los servomotores con el robot en movimiento**

<b>Parámetro</b>	<b>Resultados</b>
Encendido de servomotores	Si (pico de corriente provoca movimiento aleatorio de los servos durante el encendido, espasmo inicial de las extremidades considerable)
Ejecución del movimiento requerido	Si
Sobrecalentamiento presente	Si (motor del eslabón 2 pata 3)

Después de realizar los cambios se llegó a los resultados de la Tabla 39, se puede observar que el sobrecalentamiento de los motores es ausente, sin embargo sigue existiendo un pico de corriente inicial; para corregir el problema se implementó una fuente externa para el Arduino.

**Tabla 39**

**Resultado de las pruebas de los servos con el robot en movimiento corrección 1**

Parámetro	Resultados
Encendido de servomotores	Si ( espasmo inicial menor)
Ejecución del movimiento requerido	Si
Sobrecalentamiento presente	No

Los resultados de la implementación de una fuente externa para evitar el ruido eléctrico debido a la conexión a una fuente común con los servomotores, se observan en la Tabla 40, probando la efectividad del ensayo.

**Tabla 40**

**Resultado de las pruebas de los servos con el robot en movimiento, corrección final**

Parámetro	Resultados
Encendido de servomotores	No
Ejecución del movimiento requerido	Si
Sobrecalentamiento presente	No

## 5.2 PRUEBAS DE LA PLACAS DE POTENCIA Y DE CONTROL

El objetivo de esta prueba es verificar la alimentación y el envío de la señal de control a los servomotores sea la correcta.

### 5.2.1 PROCEDIMIENTO

- Conexión de la placa de potencia a la batería.
- Verificación del funcionamiento de la placa mediante los puntos de prueba.
- Conexión de la placa de control a la batería.
- Verificación del funcionamiento de la placa mediante los puntos de prueba.
- Conexión de un servomotor y envío de la señal de control.

- Verificación de llegada de la señal de control
- Conexión de los 12 servomotores y envío de señales de control.
- Verificación de llegada de las señales de control.
- Verificación de sobrecalentamiento de las placas.

### 5.2.2 RESULTADOS

En la Tabla 41 se observa los resultados obtenidos de acuerdo al procedimiento anterior, como se puede ver no se encontró ningún problema concluyendo con el éxito de la prueba.

**Tabla 41**

#### Pruebas de potencia y control

Parámetro	Resultado
Comprobación de voltajes en los puntos de prueba de la placa de potencia	Positivo
Comprobación de voltajes en los puntos de prueba de la placa de control	Positivo
Funcionamiento de un servomotor conectado a las placas	Positivo
Funcionamiento de los doce servomotores conectados a las placas.	Positivo
Sobrecalentamiento de las placas	Ausente

### 5.3 PRUEBAS DE COMUNICACIÓN ROS - MÓDULO RF - PLACA DE CONTROL

El objetivo es verificar que la comunicación desde el entorno de ROS hasta la placa de control sea efectiva y no se pierdan datos.

#### 5.3.1 PROCEDIMIENTO

- Conexión de los módulos de comunicación.
- Energizar placa de control.
- Ejecución de los nodos de ROS del programa.
- Conexión del puerto serial del Arduino con la computadora.
- Impresión de los datos enviados en el terminal.
- Impresión de los datos recibidos mediante el monitor serial del Arduino.
- Verificación de la cadena enviada.

### 5.3.2 RESULTADO

La Tabla 42 se observa los resultados de las pruebas de comunicación. En estas pruebas el programa completo con el algoritmo de locomoción estaba en ROS, y mediante el módulo de comunicación de transmisión se enviaba la cadena completa de caracteres más el bit de inicio y bit de salida correspondientes para cada uno. Como se puede observar el envío de datos fue efectivo, sin embargo en la recepción los datos se perdían.

**Tabla 42**

**Resultados de las pruebas de comunicación iniciales**

Parámetro	Resultado
Conexión ROS-Arduino	Positivo
Envío de la cadena de datos	Positivo
Recepción de la cadena	Negativo (datos perdidos)

Se concluyó que los problemas indicados en la Tabla 42 fueron debido a que el Arduino tiene una velocidad baja de transmisión y recepción. Por lo que para corregir este problema se procedió a cargar el algoritmo de movimiento en el Arduino y mantener en el sistema operativo de ROS solo comando de control. De esta manera, el número de caracteres enviados disminuye considerablemente y la placa de control es capaz de procesar estés órdenes.

**Tabla 43**

**Resultados de las pruebas de comunicación corregida.**

Parámetro	Resultado
Conexión ROS-Arduino	Positivo
Envío de la cadena de datos	Positivo
Recepción de la cadena	Positivo

En la Tabla 43 se observa las pruebas de las correcciones realizadas, teniendo resultados positivos y comprobando la efectividad de la prueba.

## 5.4 PRUEBAS DE LA RESPUESTA CINEMÁTICA INVERSA DEL ROBOT

Se realizaron dichas pruebas con el fin de verificar los límites: medio inferior y superior, a los que se debe posicionar la extremidad para realizar los diferentes tipos de movimiento.

### 5.4.1 PROCEDIMIENTOS

- Elaboración de una cuadrícula para poder identificar los ejes principales del robot y su perpendicularidad.
- Ubicación del robot en un punto medio de la cuadrícula tomando como referencia sus ejes principales.
- Efectuar los movimientos de cada pata hacia cada límite del espacio de trabajo, partiendo de diferentes posiciones.
- Medición de los 5 puntos tomando como referencia la proyección perpendicular del punto de contacto de la pata.
- Registro de las lecturas reales en una tabla y comparación con los valores teóricos para la obtención del error.

La Tabla 44 muestra los datos teóricos encontrados mediante el espacio de trabajo, cinemática inversa y las medidas de los eslabones.

**Tabla 44**

**Datos teóricos del espacio de trabajo-límites: inferior, medio y superior.**

	Límite Superior		Límite Medio		Límite Inferior	
	x[mm]	y[mm]	x[mm]	y[mm]	x[mm]	y[mm]
<b>P1</b>	163.4	248.4	163.4	163.4	163.4	78.4
<b>P2</b>	163.4	248.4	163.4	163.4	163.4	78.4
<b>P3</b>	163.4	78.4	163.4	163.4	163.4	248.4
<b>P4</b>	163.4	78.4	163.4	163.4	163.4	248.4

### 5.4.2 RESULTADOS EXTREMIDAD 1

La Tabla 45 muestra las medidas del punto de contacto de la pata 1 proyectado a la cuadrícula.

**Tabla 45****Mediciones de la pata 1**

Límite Superior		Límite Medio		Límite Inferior	
x [mm]	y [mm]	x [mm]	y [mm]	x [mm]	y [mm]
158	254	165	168	170	79
160	255	165	165	170	77
157	254	162	168	167	78
159	252	160	169	165	78
158	254	160	170	165	78

La Tabla 46 presenta los errores para cada coordenada en cada límite establecido para la pata 1.

**Tabla 46****Errores de cada límite en la pata 1**

Límite Superior		Límite Medio		Límite Inferior	
x [%]	y [%]	x [%]	y [%]	x [%]	y [%]
3.30	2.25	0.98	3.30	2.25	0.98
2.08	2.66	0.98	2.08	2.66	0.98
3.92	2.25	0.86	3.92	2.25	0.86
2.69	1.45	2.08	2.69	1.45	2.08
3.30	2.25	2.08	3.30	2.25	2.08

**5.4.3 RESULTADOS EXTREMIDAD 2**

La Tabla 47 muestra las mediciones del punto de contacto de la pata 2 proyectado a la cuadrícula.

**Tabla 47****Mediciones de la pata 2**

Límite Superior		Límite Medio		Límite Inferior	
x [mm]	y [mm]	x [mm]	y [mm]	x [mm]	y [mm]
177	255	171	165	172	77
174	255	173	165	170	76
176	258	173	162	172	74
176	257	173	162	171	75
174	258	173	161	171	76

La Tabla 48 presenta los errores para cada coordenada en cada límite establecido para la pata 2.

**Tabla 48****Errores de cada límite en la pata 2**

Límite Superior		Límite Medio		Límite Inferior	
x [%]	y [%]	x [%]	y [%]	x [%]	y [%]
8.32	2.66	4.65	0.98	5.26	1.79
6.49	2.66	5.88	0.98	4.04	3.06
7.71	3.86	5.88	0.86	5.26	5.61
7.71	3.46	5.88	0.86	4.65	4.34
6.49	3.86	5.88	1.47	4.65	3.06

#### 5.4.4 RESULTADOS EXTREMIDAD 3

La Tabla 49 muestra las mediciones del punto de contacto de la pata 3 proyectado a la cuadrícula.

**Tabla 49****Mediciones de la pata 3**

Límite Superior		Límite Medio		Límite Inferior	
x [mm]	y [mm]	x [mm]	y [mm]	x [mm]	y [mm]
175	73	180	168	175	258
177	75	179	166	178	258
176	75	177	168	178	259
178	73	178	166	177	260
177	75	179	166	175	259

La Tabla 50 presenta los errores para cada coordenada en el límite establecido para la pata 3.

**Tabla 50****Errores de cada límite en la pata 3**

Límite Superior		Límite Medio		Límite Inferior	
x [%]	y [%]	x [%]	y [%]	x [%]	y [%]
7.10	6.89	10.16	2.82	7.10	3.86
8.32	4.34	9.55	1.59	8.94	3.86
7.71	4.34	8.32	2.82	8.94	4.27
8.94	6.89	8.94	1.59	8.32	4.67
8.32	4.34	9.55	1.59	7.10	4.27

#### 5.4.5 RESULTADOS EXTREMIDAD 4

La Tabla 51 muestra las mediciones del punto de contacto de la pata 4 proyectado a la cuadrícula.

**Tabla 51**

**Mediciones de la pata 4**

Límite Superior		Límite Medio		Límite Inferior	
x [mm]	y [mm]	x [mm]	y [mm]	x [mm]	y [mm]
170	81	167	170	165	269
169	82	167	172	166	268
167	83	168	173	167	269
169	82	168	173	165	267
169	82	169	172	168	269

La Tabla 52 presenta los errores para cada coordenada en cada límite establecido para la pata 4.

**Tabla 52**

**Errores de cada límite en la para 4**

Límite Superior		Límite Medio		Límite Inferior	
x [%]	y [%]	x [%]	y [%]	x [%]	y [%]
4.04	3.32	2.20	4.04	0.98	8.29
3.43	4.59	2.20	5.26	1.59	7.89
2.20	5.87	2.82	5.88	2.20	8.29
3.43	4.59	2.82	5.88	0.98	7.49
3.43	4.59	3.43	5.26	2.82	8.29

De acuerdo a la Tabla 46, Tabla 48, Tabla 50 y Tabla 52 se pueden observar valores aceptables en las cuatro patas ya que no comprometen el movimiento del robot. El error máximo se localizó en la pata 3 con un valor de 10% mientras que el valor mínimo se encontró en la pata 1 con un valor de 3.30%.

## 5.5 PRUEBAS DE CAMINATA DISCONTINUA DEL ROBOT

El objetivo de estas pruebas es verificar el funcionamiento del robot en los distintos movimientos. Para ello se realizarán pruebas de caminata discontinua de traslación y rotación (giro izquierdo y derecho).

**Tabla 53**

**Resultados de las pruebas de caminata discontinua traslación**

No.	x[mm]	y[mm]	Error (%)
1	12.5	161.5	4.72
2	12	164	3.27
3	8	166	2.24
4	12.5	164.5	2.96
5	8	161.5	4.88
6	2.5	159	4.69
7	13.5	162.5	4.08
8	5	160.5	4.95
9	3.5	161.5	4.98
10	-9.5	157.5	4.66

En la Tabla 53, se detalla los alcances obtenidos en la caminata discontinua de traslación para diez pruebas, dando como resultado errores menores de 5%. Para esta prueba se efectuó los movimientos requeridos para llegar a 170 mm equivalente a diez espacios de trabajo.

En la Tabla 54 se muestran los resultados para el movimiento de giro antihorario, al igual que en el anterior se procedió a realizar diez pruebas y el registro del ángulo alcanzado, de esta manera se obtuvieron errores menores al 4% concluyendo en la efectividad del giro.

Tabla 54

**Resultados de las pruebas de caminata discontinua de rotación (giro antihorario)**

No.	Ángulo (°)	Error (%)
1	90	0.00
2	88	2.22
3	88	2.22
4	87	3.33
5	88	2.22
6	89	1.11
7	84	6.67
8	91	1.11
9	91	1.11
10	91	1.11

En la Tabla 55 se muestran los resultados para el movimiento de giro horario, mediante los diez ensayos realizados se obtuvo un error máximo del 6.67% comprobando la efectividad de la prueba.

Tabla 55

**Resultados de caminata discontinua de rotación (giro horario)**

No.	Ángulo (°)	Error (%)
1	89	1.11
2	88	2.22
3	89	1.11
4	90	0.00
5	92	2.22
6	89	1.11
7	89	1.11
8	91	1.11
9	90	0.00
10	90	0.00

## CONCLUSIONES

El diseño implementado cumplió con una sinergia eficiente de las fases de los sistemas: mecánico, de control, de potencia y comunicación; permitiendo alcanzar una estructura robótica ligera y estable, con una respuesta efectiva entre la estación remota y la plataforma.

El diseño del algoritmo de locomoción en base al análisis cinemático y dinámico, aplicado al método de caminata discontinua de dos fases elegida, entregó un error máximo del 10% para el movimiento de cada extremidad y un 5% para el movimiento conjunto de todo el mecanismo ya sea en traslación o rotación, logrando una exactitud del 90% que aseguró una locomoción estable y coordinada del cuadrúpedo.

De acuerdo con los valores de torque obtenidos en el análisis dinámico, se eligieron motores dimensionados con un factor de seguridad de 1.5, con lo cual se logró un correcto desempeño en los desplazamientos del robot y se evitó un esfuerzo excesivo de cada servomotor.

En el análisis de esfuerzos de cada eslabón para el punto crítico en que una extremidad permanece en su fase de transferencia, se obtuvieron factores de seguridad entre 3.4 y 9.6, afirmando que los materiales están correctamente seleccionados y cumplieron con la meta de tener una estructura apropiada que soporte todos los componentes mecánicos y electrónicos.

La batería LiPo elegida demostró ser capaz de suministrar al circuito la potencia necesaria para su funcionamiento, mientras que la batería alcalina como fuente de alimentación independiente para la placa de control, fue una solución viable para evitar el reseteo del Arduino debido al sobre pico de corriente en el transitorio de encendido, consiguiendo de esta manera un sistema de alimentación confiable y estable.

Al dividir las funciones de control entre ROS y la placa, la comunicación serial se pudo efectuar mediante la transmisión de una cadena corta de caracteres, cumpliendo con un tiempo de transmisión de datos de 0.5 segundos, suficiente para la ubicación de cada extremidad. Consiguiendo que el robot pueda movilizarse de forma estable al alcanzar una transferencia de datos óptima y confiable.

La arquitectura de ROS basada en nodos de manera modular, fue de vital importancia para un procesamiento de datos óptimo e independiente, que pueda comunicarse fácilmente con todos los dispositivos implementados en el presente proyecto, consiguiendo un software robusto y confiable ante eventuales fallos.

El programa *Qt Creator* es una herramienta versátil en donde se consiguió implementar todos los comandos necesarios para el control del robot dentro de una interfaz. Así mismo se alcanzó una interacción completa con el usuario a través de un cuadro de diálogo que siempre esté informando que acciones se realizaron desde el momento de la activación de la HMI para facilitar su monitoreo constante.

## RECOMENDACIONES

Para futuros proyectos se puede optimizar costos de elaboración mediante el uso de materiales livianos y de alta resistencia como el aluminio para su estructura, de manera que los actuadores se dimensionen con menores torques de trabajo.

Es conveniente implementar una morfología que se componga de figuras simples en la estructura del robot para facilitar su modelamiento, en la medida que no afecte su funcionalidad.

Los servomotores que poseen mejores cualidades para este tipo de aplicaciones son los desarrollados por la marca Dynamixel, debido a sus características de engranajes de metal, mecanismo anti-retorno y medición de: posición, torque y velocidad al momento de ejecutarse el movimiento. Por otra parte tienen compatibilidad con ROS y brindan un sistema de control retroalimentado.

En próximas investigaciones se debe acoplar una tarjeta de adquisición de datos, para el desarrollo de nuevos algoritmos de control en base a la implementación de sensores y otros dispositivos; ya que el procesamiento y control del robot tienen un alto coste computacional para la tarjeta de control.

Se puede realizar un análisis más amplio de las características de la superficie de contacto del último eslabón en cuanto a su dimensión y tipo de material, para observar su repercusión en la estabilidad y adaptabilidad con el terreno.

## BIBLIOGRAFÍA

- Alomoto Ortiz, C. A., & Jiménez Guamán, S. V. (2015). Diseño y construcción de un prototipo de robot móvil controlado con tecnología Raspberry Pi para la inspección y fumigación focalizada de cultivos de haba. (*Tesis de Pregado de Ingeniería Mecatrónica*). Universidad de las Fuerzas Armadas ESPE , Sangolquí .
- Andino Alberca, C. A., & Rodríguez Sánchez, D. X. (2016). *Diseño e implementación de algoritmos de control inteligente para un robot Phoenix tipo hexápodo mediante la tarjeta STM32F4 discovery y simulink de matlab*. Universidad de las Fuerzas Armadas ESPE, Carrera de Ingeniería en Electrónica, Automatización y Control.
- ARDUINO. (27 de 07 de 2015). *ARDUINO*. Obtenido de Detalle de productos: [www.arduino.cc](http://www.arduino.cc)
- ArduPilot Dev team. (2016). *Mission Planner*. Obtenido de <http://ardupilot.org/planner/>
- Argudo Cobos, G. S., & Arpi Saldaña, A. D. (2012). Diseño y construcción de un robot móvil tele operado para la asistencia en operaciones de alto riesgo del Cuerpo de Bomberos. (*Tesis de Pregrado de Ingeniería Electrónica*). Universidad Politécnica Salesiana, Cuenca.
- Barrientos, A., Peñín, L. F., Balaguer, C., & Aracil, R. (1996). *Fundamentos de Robótica*. Madrid: McGraw Hill.
- Büttner, T., Roennau, A., Heppner, G., Pfozter, L., & Dillmann, R. (2016). Bio-inspired optimization of kinematic models for multi-legged walking robots. *In Biomedical Robotics and Biomechatronics (BioRob)*. 2016 6th IEEE International Conference on, págs. 430-435. IEEE.

- Cao, Y., Lu, K., Li, X., & Zang, Y. (2011). Accurate Numerical Methods for Computing 2D and 3D Robot Workspace. *International Journal of Advanced Robotic Systems, INTECH*, 6 , págs. 1-13.
- Cao, Y., Lu, K., Li, X., & Zang, Y. (2011). Accurate Numerical Methods for Computing 2D and 3D Robot Workspace. *INTECH*, 2.
- Caterpillar. (20 de Septiembre de 2016). *Caterpillar*. Obtenido de Caterpillar News: <http://www.caterpillar.com/en/news/caterpillarNews/history/the-story-of-big-muskie.html>
- Ceccarelli, M. (2004). Development of Walking Machines; Historical Perspective. *International Symposium on History of Machines and Mechanisms: proceedings of HMM 2004. (Vol. 4)*, págs. 357-370. Springer Science & Business Media.
- Chang, D., Son, D., Seo, T., Nam, W., Jeon, D., & Kim, J. (2009). Kinematics-based gait planning of a quadruped gecko-like model. *In Robotics and Biomimetics (ROBIO), 2009 IEEE. International Conference on*, págs. 233-238. IEEE.
- Chimarro Amaguaña, J. D., & Enriquez Herrera, A. D. (2015). "Diseño, construcción e implementación de un robot esférico de 4 grados de libertad para la manipulación de objetos utilizando la plataforma Robotic Operating System (ROS). (*Tesis dde Pregrado en Ingeniería Macatrónica*). Universidad de las Fuerzas Armadas ESPE, Sangolquí.
- Cruse, H., Dean, J., Muller, U., & Schmitz, J. (1991). The stick insect as a walking robot. *In Advanced Robotics, 1991.'Robots in Unstructured Environments', 91 ICAR. Fifth International Conference on*, págs. 936-940. IEEE.
- de Santos, P. G., García, E., & Estremera, J. (2007). *Quadrupedal locomotion: an introduction to the control of four-legged robots*. Springer Science & Business Media.

- Erazo Sosa, A. S. (2014). Terrain adaptative gaits for hexapod robots. (*Tesis de Maestría en Ingeniería Electrónica en Automatización y Control*). The University of Queensland, Australia.
- Ferninand , B., Johnston, R., John, D., & Mazuerk, D. (2010). *Mecánica de Materiales*. México: Mc Graw Hill.
- Fichter, E. F., & Fichter, B. L. (1988). A survey of legs of insects and spiders from a kinematic perspective. *Robotics and Automation, 1988. Proceedings. 1988 IEEE International Conference on*, págs. 984-986. IEEE.
- Forero, D. Y., Mora, P. A., Loaiza, J. L., & Hernández, R. D. (2015). Análisis para la simulación de modelos animales. *Vector*, 54-62.
- Fuertes, C., & Llumiquinga, R. (Mayo de 2005). Diseño e Implementación de un Robot Móvil tipo Hexápodo Teledirigido. (*Tesis de Pregrado en Ingeniería Electrónica*). Universidad de las Fuerzas Armadas ESPE, Ecuador.
- Gallardo Andrade, J. A., & Gaona Román, A. D. (2015). Diseño y construcción de una plataforma robótica móvil para interiores capaz de realizar SLAM (Simultaneous Localization and Mapping). (*Tesis de Pregrado en Ingeniería Mecatrónica*). Universidad de las Fuerzas Armadas ESPE, Sangolquí.
- Geeetech Wiki. (24 de Septiembre de 2014). *3DR Radio Telemetry*. Obtenido de [http://www.geeetech.com/wiki/index.php?title=3DR\\_Radio\\_Telemetry&oldid=4976](http://www.geeetech.com/wiki/index.php?title=3DR_Radio_Telemetry&oldid=4976)
- Gullan, P. J., & Cranston, P. S. (2009). *The insects: an outline of entomology*. John Wiley & Sons.
- Hidalgo Panchana, A. C. (2015). Implementación de un prototipo de un robot móvil tipo hexápodo teledirigido para la contribución de la limpieza

aplicando tecnología open source. (*Tesis de Pregado en Ingeniería Electrónica y Telecomunicaciones*). Universidad Estatal Península de Santa Elena, Santa Elena.

Hidayat, A., Jati, A. N., & Saputra, R. E. (2016). Autonomous quadruped robot locomotion control using inverse kinematics and sine pattern methods. *In Control, Electronics, Renewable Energy and Communications (ICCEREC), 2016. International Conference on*, págs. 96-100. IEEE.

Hirose, S., Fukuda, Y., Yoneda, K., Nagakubo, A., Tsukagoshi, H., Arikawa, K., . . . Hodoshima, R. (2009). Quadruped walking robots at tokyo institute of technology: design, analysis, and gait control methods. *IEEE Robotics and Automation Magazine*, 16(2), 104-114.

Hock, O., Drgona, P., Jaros, V., & Havrila, R. (2016). Transposition method for inverse kinematics. *In ELEKTRO, 2016* (págs. 152-155). IEEE.

IPC. (2014). *Generic Standar on printed board design*. Illinois: IPC.

IPC. (2014). *IPC-2221B Generic Standard on Printed Board Design*. Illinois: IPC.

Irawan, A., Mohd Razali, D., & Nonami, K. B. (2014). *Hydraulically Actuated Hexapod Robots: Design, Implementation and Control*. (Vol. 66): Springer.

Kim, K. Y., & Park, J. H. (2008). Ellipse based leg trajectory generation for galloping quadruped robots. *Journal of mechanical science and technology*, 22(11), págs. 2099-2106.

Klowden, M. J. (2007). *Physiological Systems in Insects* (2nd ed.). Academic Press.

Li, Z., Xiao, S., & Ge, S. S. (2014). Fuzzy approximation adaptive control of quadruped robots with kinematics and dynamics uncertainties. *In Fuzzy Systems (FUZZ-IEEE), 2014 IEEE. International Conference on*, págs. 701-706. IEEE.

- Liu, C., Chen, Q., & Wang, D. (2011). CPG-inspired workspace trajectory generation and adaptive locomotion control for quadruped robots. *Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(3) (págs. 867-880). IEEE.
- Machuma Mendoza, C. R., & López Ayala, L. F. (2015). Locomoción de un robot cuadrúpedo basada en redes neuronales. (*Tesis de Pregrado en Ingeniería Electrónica en Control y Redes Industriales*). Escuela Superior Politécnica de Chimborazo, Riobamba.
- Maldonado Del Toro, H. M., Silva Ortigoza, R., Ramos Silvestre, E. R., Hernández Guzman, V. M., & Rivera Díaz, J. C. (2011). Modelado y simulación de un robot rígido de dos grados de libertad. *Lat. Am. J. Phys. Educ.*, 5(1), 322-324.
- Marguedas. (23 de Mayo de 2016). *ROS.org*. Obtenido de <http://wiki.ros.org/ROS/Tutorials/InstallingandConfiguringROSEnvironment>
- Martínez, A., & Fernández, E. (Septiembre de 2013). *Learning ROS for Robotics Programming*. Birmingham, Reino Unido: PACKT PUBLISHING.
- McGhee, R. B. (1976). Robot locomotion. *Neural control of locomotion*, (págs. 237-264).
- McGhee, R. B. (1977). Control of legged locomotion systems. *Proc. 18th Automatic Control*, (págs. 205-215). San Francisco.
- McGhee, R. B., & Frank, A. A. (1968). On the stability properties of quadruped creeping gaits. *Mathematical Biosciences* 3, (págs. 331-351).
- Morgan, T. H. (1925). *Evolution and genetics*. Princeton University Press.
- Morrison, R. A. (1968). Iron mule train. In *Cornell Aeronautical Lab. ISTVS Off-Road Mobility Research Symposium*. Washington DC.

- Mrva, J. (2014). Design of motion primitives for a hexapod walking robot operating in a rough environment. (*Master's Thesis of Electrical Engineering*). Czech Technical University in Prague.
- Mullo Mullo, E. G., & Aymacaña Masapanta, W. R. (2009). Diseño y construcción de un robot hexápodo con sistema de articulaciones de apéndice flexible diseñado para lugares de difícil acceso, con sistema de control remoto y monitoreo en tiempo real. (*Tesis de Pregado en Ingeniería Electrónica e Instrumentación*). Universidad de las Fuerzas Armadas ESPE, Latacunga .
- Naranjo Suárez, K. Z. (2013). Implementación de un prototipo de robot móvil con visión nocturna para vigilancia y seguridad. (*Tesis de Pregado en Ingeniería Electrónica y Telecomunicaciones*). Universidad Estatal Península de Santa Elena , Santa Elena .
- Paramani, A. P. (2011). Optimum inverse kinematic method for a 12 DOF manipulator. *In Mechatronics and Automation (ICMA), 2011. International Conference on*, págs. 2020-2026. IEEE.
- Pere, P., & Granollers, A. (s.f.). Diseño de Pantalla Máste en Interacción Persona Ordenador . Universidad Politécnica de Catalunya.
- Pinta Arrobo, F. H. (2007). Diseño y construcción de un robot de vigilancia. (*Tesis de Pregrado de Ingeniería Electrónica y Control*). Escuela Politécnica Nacional, Ecuador.
- Ponsa, P., & Granollers, A. (s.f.). *Diseño de Pantalla MIPO*. Master en Interacción Persona Ordenador, Universidad Politécnica de Catalunya, Catalunya.
- Qiu, G. Y., & Wu, S. H. (2011). The evolutionary locomotion of tripedal and quadrupedal biomorphic robots. *In Technologies and Applications of Artificial Intelligence (TAAI). 2011 Intrnacional Conference on*, págs. 45-50. IEEE.

- Raibert, M. H. (1986). Legged robots. *Communications of the ACM*, 29(6), págs. 499-514.
- Raibert, M., & Chepponis, M. . (1986). Running on four legs as though they were one. *IEEE Journal on Robotics and Automation*, 2(2), 70-82.
- Rong, X. W., Li, Y. B., Ruan, J. H., & Song, H. J. (2010). Kinematics analysis and simulation of a quadruped robot. *In Applied Mechanics and Materials. Vol. 26*, págs. 517-522. Trans Tech Publications.
- ROS.org. (Enero de 2017). ROS. Obtenido de Wiki: Documentation: <http://www.ros.org/>
- Rosheim, M. E. (1997). In the footsteps of leonardo [articulated anthropomorphic robot]. *IEEE Robotics & Automation Magazine*, 4(2), 12-14.
- Ross, H. H., Charles, A. R., & Juneprrie, A. R. (1982). *A textbook of entomology*.
- Sandler, B. Z. (1999). *Robotics: designing the mechanisms for automated machinery*. Academic Press.
- Senplades. (2003-2017). PNBV. *Plan Nacional del Buen Vivir*.
- Siciliano, B., & Khatib, O. (Edits.). (2008). *Springer handbook of robotics*. Springer Science & Business Media.
- Tedeschi, F., & Carbone, G. (2014). Design issues for hexapod walking robots. *Robotics*, 3(2), págs. 181-206.
- Terra. (3 de Mayo de 2013). Terra. Obtenido de Discurso en español de Barack Obama en Museo de Antropología: <http://noticias.terra.com.mx/mexico/discurso-en-espanol-de-barack-obama-en-museo-de-antropologia,c06f377277c6e310VgnVCM3000009acceb0aRCRD.html>

- Torq Pro & Tower Pro. (2015). *TowerPro*. Obtenido de TowerPro: <http://www.towerpro.com.tw/>
- Trimmer, B. A., Lin, H. T., Baryshyan, A., Leisk, G. G., & Kaplan, D. L. (2012). Towards a biomorphic soft robot: design constraints and solutions. *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on* (págs. 599-605). IEEE.
- Tripod Gait Robotics. (17 de 11 de 2016). *Insect inspired robots*. Obtenido de The anatomy of insect leg: <http://tripodgaitrobotics.weebly.com/the-anatomy-of-insect-legs.html>
- WiFi/IP Camera. (s.f.). Manual User.
- Zhang, C.-D., & Song, S.-M. (1989). Gaits and geometry of a walking chair for the disabled. *Journal of Terramechanics*, 26(3-4), págs. 211-233.
- Zhang, C.-D., & Song, S.-M. (1990). Stability analysis of wave-crab gaits of a quadruped. *Journal of Robotic Systems*, 7(2), 243-276.
- Zhang, Z., Dequidt, J., Kruszewski, A., Largilliere, F., & Duriez, C. (2016). Kinematic modeling and observer based control of soft robot using real-time Finite Element Method. *In Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ. International Conference on* , págs. 5509-5514. IEEE.