



Simulación e Implementación de una Configuración de Robot Hexápodo Utilizando Sistemas de Robótica Modular para Evaluar su Locomoción

Mateo Cotera Berdugo

Tutor:

Ing. Ricardo Andrés Castillo Estepa, PhD

Universidad Militar Nueva Granada

Facultad de Ingeniería

Programa de Ingeniería en Mecatrónica

Bogotá

2017



Simulación e Implementación de una Configuración de Robot Hexápodo Utilizando Sistemas de Robótica Modular para Evaluar su Locomoción

Mateo Cotera Berdugo

Tutor:

Ing. Ricardo Andrés Castillo Estepa, PhD

Trabajo de Grado para Optar por el Título de Ingeniero en Mecatrónica

Universidad Militar Nueva Granada

Facultad de Ingeniería

Programa de Ingeniería en Mecatrónica

Bogotá

2017

Bogotá (2017)

Nota de Aceptación

Firma del Presidente del Jurado

Firma de Jurado

Firma de Jurado

Agradecimientos

A mis padres Donaldo Enrique Cotera y Luz Elena Berdugo, por el apoyo incondicional que siempre me han brindado. Por su cariño y educación, sin ellos no sería la persona que soy ahora y no estaría cumpliendo esta meta.

A mis hermanos, amigos y familiares que me han acompañado a lo largo de este logro. Ellos hicieron que este camino fuese más fácil de llevar.

Al profesor Ricardo Castillo PhD. y al Ing. David Gómez que me guiaron en este proyecto. Gracias a que me brindaron parte de su tiempo y conocimientos es que este logro ha sido posible.

A la Universidad Militar Nueva Granada, al programa de ingeniería Mecatrónica por abrirme sus puertas. A los docentes y personal de laboratorio por todo el trabajo realizado para darme una formación integral y hacerme una mejor persona.

Contenido

1	Introducción.....	14
1.1	Planteamiento del Problema.....	15
1.2	Objetivo General	16
1.3	Objetivos Específicos.....	16
1.4	Justificación.....	17
1.5	Metodología	18
2	Referentes Teóricos	19
2.1	Marco Teórico.....	19
2.1.1	Robótica e Historia	19
2.1.2	Locomoción de los robots móviles.....	20
2.1.3	Robots caminantes.....	20
2.1.4	Robots hexápodos.....	22
2.1.5	Robótica modular	27
2.2	Antecedentes en Robótica Modular	35
2.2.1	Módulos Polybot G1.....	35
2.2.2	Modulo M-TRAN.....	36
2.2.3	Sistema robótico modular con aplicaciones a la exploración espacial.....	37
2.2.4	Diseñado evolucionado y auto-reproductores robótica modular.....	38
2.2.5	Cubo M.....	39
2.2.6	UBot (2011).....	40
2.2.7	Módulo robótico exploración autónoma y con varios modos de locomoción.....	41
2.2.8	Módulo MECABOT	42
2.2.9	MECABOT 2.0.....	43

2.2.10	MECABOT 3.0.....	44
2.2.11	Módulo MECABOT 4.0.....	45
3	Modelo Matemático Configuración Hexápodo	46
3.1	Cinemática Directa de una pata del hexápodo MECABOT	51
3.1.1	Cinemática directa de la conexión cara-pivote cambiando orientación	53
3.2	Cinemática inversa de una pata del hexápodo MECABOT	57
3.2.1	Cinemática inversa del módulo con acople cara-pivote cambiando el sentido	58
4.	Simulación Sistema Robótico Modular MECABOT 4.0 Configuración Hexápodo.....	61
4.1	Simulador Webots	62
4.2	Importación a Webots del módulo MECABOT 4.0.....	65
4.3	Diseño de robot modular Hexápodo en Webots.....	68
4.4	Programación marchas del hexápodo MECABOT 4.0	71
4.4.1	Avance hacia adelante	73
4.4.2	Movimiento hacia atrás	77
4.4.3	Movimiento lateral izquierdo	78
4.4.4	Movimiento lateral derecho.....	82
4.4.5	Movimiento rotacional hacia la derecha.....	83
4.4.6	Movimiento rotacional hacia la izquierda	86
5.	Implementación del robot hexápodo con los módulos MECABOT 4.0	87
5.1	MECABOT 4.0	88
5.2	Caracterización de los servomotores.....	94
6.	Resultados de implementación del Hexápodo MECABOT 4.0.....	96
6.1	Diseño e implementación de Software de comandos e interfaz de Usuario	97
6.1.1	Programa de la tarjeta de control Teensy 3.2	99
6.2	Implementación de movimientos	101

6.2.1	Avance hacia adelante:	101
6.2.2	Movimiento Lateral	108
6.2.3	Movimiento Rotacional	111
7.	Conclusiones y recomendaciones a trabajos futuros	112
	Referencias	114
8.	Anexos.....	117
	Apéndice 8-1. Tabla de estados de movimientos:	117
	Apéndice 8-2 Tabla recolección pruebas de simulación Movimiento hacia adelante.....	123
	Apéndice 8-3 Tabla recolección pruebas de implementación movimiento adelante	124
	Apéndice 8-4 Tabla recolección pruebas de simulación movimiento lateral.....	125
	Apéndice 8-5 Tabla recolección pruebas de implementación movimiento lateral.....	126
	Apéndice 8-6 Diagrama de flujo algoritmo de programación para comando de módulo	127
	Apéndice 8-7. Tabla recolección de datos para la gráfica 4-14.	128
	Apéndice 8-8. Tabla de datos figura 4-19 desplazamiento lateral variando velocidad ..	129
	Apéndice 8-9. Tabla de datos grafica figura-23, rotación hexápodo en simulación.	130
	Apéndice 8-10. Tabla de la figura 6-6, grafica avance del hexápodo	131
	Apéndice 8-11. Tabla de datos de la Figura 6-8, velocidad avance distintos terrenos.....	131
	Apéndice 8-12. Tabla de la figura 6-9, desplazamiento hexápodo diferentes terrenos ...	132
	Apéndice 8-13. Tabla figura 6-10, grafica velocidad hexápoda en simulación y físico. .	133
	Apéndice 8-14. Tabla de datos la figura 6-13, desplazamiento del robot hexápodo	133
	Apéndice 8-15. Tabla de datos figura 6-13, desplazamiento hexápodo	134
	Apéndice 8-16. Tabla de datos figura 6-15, rotación del hexápodo físico.....	135

Lista de Tablas

2-1	Patras en contacto con el suelo y suspendidas.....	24
3-1	Parámetros D-H para un robot de n articulaciones.....	50
3-2	Tabla Denavid-Hartenberg de la conexión de modulo pivote-pivote.....	52
5-2	Componentes módulo MECABOT 4.0.....	85
6-1	Cadena de caracteres para comando de hexápodo	92
6-2	Paso de caracteres a bits para ejecutar movimientos.....	95

Índice de Figuras

1-1	Metodología del trabajo	18
2-1	Desplazamiento de un robot móvil terrestre.....	19
2-2	Desplazamiento de un Ventajas de los robots caminantes	22
2-3	Triangulo de apoyo insecto tenebrio.....	23
2.4	Ventajas de la locomoción por patas topología hexápodo.....	24
2-5	Trayectoria que sigue una pata para que el robot se desplace.....	25
2-6	Hexápodos de a) 1 grado, b) 2 grados y c) más grados de libertad.	26
2-7	Robot modular reticular: Módulos DFA en arquitectura de red	28
2-8	Robot modular tipo cadena. a) Polybot G1 de rueda a oruga, b) Oruga con módulos CoSMO	29
2-9	Robots tipo cadena topología 1D	30
2-10	Robots tipo cadena topología 2D	30
2-11	Robots cuadrúpedos utilizando topología 2D tipo cadena	31
2-12	Robot tipo cadena topología 3D	31
2-13	Robot tipo cadena topología 3D. Concepción artística de robots modulares en el espacio.	32
2-14	Robos modulares del tipo colonial.....	32
2-15	Robot Polybot G1 en configuración rueda y serpiente.....	35
2-16	Vista esquemática modulo M-TRAN.....	36
2-17	Robot cuadrúpedo avanzando.....	38
2-18	Diseño secuencia de auto reproducción asistida.....	39
2-19	a) Modulo M, b) Configuración cuadrúpedo y c) configuración serpiente con módulos.....	40
2-20	Sistema UBot en configuración cuadrúpedo.....	40
2-21	Marcha del cuadrúpedo con módulos Scout robot sorteando un obstáculo.....	44
2-22	Módulo MECABOT.....	43
2-23	Módulo MECABOT 2.0.....	43
2-24	Configuración oruga implementada con los modulos MECABOT 3.0.....	44
2-25	Módulo MECABOT 4.0.....	45
3-1	Polígono de estabilidad configuración Hexápodo MECABOT 4.0.....	47

3-2	Módulo MECABOT 4.0 conexión de semimódulos cara-cara.....	48
3-3	Módulo MECABOT 4.0 en conexión Pivote-Pivote.....	49
3-4	Módulo MECABOT 4.0 en conexión de semimódulos Cara-Pivote	49
3-5	MECABOT 4.0 acople Cara-pivote cambio de sentido en los semimódulos.....	50
3-6	Matriz de transformación homogénea.....	51
3-7	Módulo MECABOT 4.0 en conexión cara-pivote cambiando el sentido.....	55
3-8	Modelo simplificado MECABOT 4.0 conexión cara-pivote cambiando sentido ...	54
3-9	Matriz de transformacion de la pata en acople cara-pivote.....	55
3-10	Grafica de matlab. Pata hexápodo en conexión cara-pivote cambiando el sentido..	56
3-11	Parámetros cinemática inversa MECABOT 4.0.....	58
3-12	Cinemática inversa de la pata del hexápodo plano x, y.....	59
3-13	Cinemática inversa de la pata sobre el plano y, z.....	60
4-1	Ventana de iniciación de Webots el sentido.....	62
4-2	Jerarquía principal del nodo World.....	64
4-3	Semimódulo MECABOT 4.0 vista explosionada.....	65
4-4	Modulo simplificado importado a Webots.....	66
4-5	Jerarquía de nodos para los módulos MECABOT en Webots.....	67
4-6	Robots modulares hexápodos con patas frontales y laterales.....	68
4-7	Robots hexápodos modulares con patas laterales	69
4-8	Robot hexápodo modular utilizando los módulos MECABOT 4.0.....	70
4-9	Asignación de variables del robot en el controlador de Webots.....	71
4-10	Robot hexápodo indicando la nomenclatura de cada semimódulo	72
4-11	Algoritmo de programación para marchar hacia adelante.....	73
4-12	Secuencia para dar marcha hacia adelante en el simulador Webots	74
4-13	Graficas de velocidad hexápodo con los módulos MECABOT 4.0.....	75
4-14	Graficas desplazamiento del robot hexápodo variando la velocidad de los moores..	76
4-15	Algoritmo de programación para marcha hacia atrás.....	77
4-16	Algoritmo de programación para movimientos laterales hacia la izquierda.....	78
4-17	Secuencia para dar movimientos hacia la izquierda en el simulador Webots.....	79
4-18	Grafica de velocidad del robot hexápodo en desplazamiento lateral.....	80
4-19	Grafica desplazamiento lateral variando la velocidad de los motores.....	81
4-20	Algoritmo de programación para movimientos laterales hacia la derecha	82

4-21	Algoritmo de programación para movimiento rotacional hacia la derecha.....	83
4-22	Secuencia para movimientos rotacionales hacia la derecha simulada en Webots	84
4-23	Grafica rotación hexápodo en simulación.....	85
4-24	Algoritmo de programación para movimiento rotacional hacia la izquierda.....	86
5-1	Semimódulos MECABOT 4.0 y 3.0 vista lateral.....	88
5-2	Módulo MECABOT 4.0.....	89
5-3	a) Impresión de la pieza, b) pieza impresa.....	90
5-4	Ensamble de piezas electrónicas	91
5-5	Unión de caras.....	91
5-6	Un Módulo MECABOT 4.0 implementado	92
5-7	Motor HD-1810MG y sus características.....	93
5-8	Step-Down 5v/2.5 A D24V22F5 y características.....	93
5-9	Caracterización servomotor derecho.....	94
5-10	Caracterización servomotor izquierdo.....	95
6-1	Robot modular hexápodo a partir de módulos MECABOT 4.0.....	96
6-2	Interfaz de usuario para control de hexápodo	99
6-3	Teensy 3.2.....	100
6-4	Marcha hacia adelante del Robot Hexápodo modular MECABOT 4.0.....	101
6-5	Grafica velocidad avance del hexápodo.....	102
6-6	Grafica velocidad avance del hexápodo cambiando el ángulo del pivote.....	103
6-7	Hexápodo MECABOT 4.0 avanzando sobre terreno irregular pasto y rocoso.....	104
6-8	Gráfica velocidad de avance del hexápodo en distintos terrenos.....	105
6-9	Desplazamiento del hexápodo en diferentes terrenos variando velocidad motores.....	106
6-10	Gráfica velocidad hexápodo en simulación y físico.....	107
6-11	Marcha lateral izquierda del Robot Hexápodo modular MECABOT 4.0.....	108
6-12	Grafica velocidad avance lateral	109
6-13	Desplazamiento del robot Hexápodo cambiando la aceleración de los motores.....	110
6-14	Hexápodo modular girando a 45 y 90 grados.....	111
6-15	Grafica rotación hexápodo físico.....	111

Resumen

Dentro de la Universidad Militar Nueva Granada, en el grupo de investigación Davinci, se desarrolló el sistema robótico modular conocido como MECABOT, el cual se encuentra actualmente en la versión 4.0, con el que se busca crear robots móviles a partir de distintas configuraciones. Este trabajo se centra en la configuración de tipo hexápodo, modelo bio-inspirado en los insectos muy utilizado en la robótica debido a su gran estabilidad y movimientos con independencia del terreno, realizando su simulación e implementación con los módulos MECABOT 4.0. En este se busca evaluar las capacidades motoras de esta configuración a partir de módulos, para esto se utilizó el software de simulación Webots que permite simular las características físicas del modelo y su comportamiento ante el programa implementado en este. Se llevó a cabo la construcción de los módulos necesarios para montar la configuración utilizando los robots modulares MECABOT 3.0, realizados por el grupo de investigación, pero debido a que estos no contaban con las características necesarias para mover la configuración, se cambió a un modelo de módulo más compacto, ligero y con motores de mayor fuerza, el nuevo robot modular se denominó MECABOT 4.0, los cuales fueron construidos para ser usados en la configuración Hexápodo y Rueda. También se diseñó e implementó un software con interfaz gráfica en el software MATLAB® para ejecutar los movimientos de la morfología hexápodo, junto con un programa en el software Arduino cargado en la tarjeta teensy 3.2, que comanda los módulos que le dan movimiento al robot modular. Para evaluar las capacidades de locomoción de esta arquitectura, se realizaron pruebas en las que se variaban parámetros como la velocidad y aceleración en los motores de los módulos, para conocer la velocidad de avance de la configuración y los parámetros que permiten un movimiento adecuado. Esta configuración está enfocada en lograr desplazamientos sobre terrenos irregulares, con el fin de ser utilizada en futuros proyectos que impliquen tareas de exploración, búsqueda y rescate. Como una primera aproximación se programaron e implementaron los movimientos: adelante, atrás, lateral izquierdo y lateral derecho, además de rotaciones en ambos sentidos para dar maniobrabilidad al robot modular hexápodo.

Abstract

Inside the Universidad Militar Nueva Granada, in the research group Davinci developed the robotic modular system known as MECABOT, which meets nowadays in the version 4.0 the one that seeks to create mobile robots from different configurations. This work centers on the configuration of type hexapod, bio model inspired by the insects very used in the robotics due to his great stability and movements with independence of the area, realizing his simulation and implementation with the modules MECABOT 4.0. In this one, the motor capacities of this configuration are evaluated by means of the Webots simulation software, which allows to simulate the physical characteristics of the model and its behavior when using the program implemented in it. There was carried out the construction of the necessary modules to mount the configuration after several redesigns in the module MECABOT 3.0 realized ones for the group of investigation, turning out to be a more compact, light design and with engines of major force acquaintances MECABOT 4.0, which were constructed to be used in the configuration hexapod. It was also designed and implemented software with a graphic interface in the MATLAB® software to execute the movements of the hexapod morphology, together with a program in the Arduino software loaded in the teensy 3.2 card, which commands the modules that give movement to the modular robot. To evaluate the capacities of locomotion of this architecture, there were realized tests in which parameters were changed as the speed and acceleration in the engines of the modules, to know the speed of advance of the configuration and the parameters that allow a suitable movement. This configuration is focused in achieving displacements on irregular areas, in order to be used in future projects that imply tasks of exploration, search and rescue, as the first approximation the movements were programmed and implemented: forward, behind, left and lateral right wings, besides rotations in both senses to give maneuverability to the modular robot hexapod.

1 Introducción

Actualmente la mayoría de investigaciones en robótica, han estado centradas en el desarrollo de robots de construcción robusta y con articulaciones definidas para realizar trabajos específicos, en donde las condiciones de trabajo se mantienen constantes, como es el caso de robots manipuladores y cartesianos lo que limita el número de tareas que estos pueden realizar. Debido a esto, y con la intención de crear sistemas capaces de realizar múltiples tareas y con capacidades adaptativas nace la robótica modular, la cual consiste en utilizar pequeños módulos independientes entre sí que pueden ser fácilmente acoplados a otros módulos iguales para crear sistemas como si de bloques de construcción se trataran. El hecho de agregar y quitar módulos para crear cualquier tipo de morfología, hace de la robótica modular un sistema flexible y económico que permite que el mantenimiento y rediseño se hagan de forma inmediata, remplazando módulos afectados y agregando más de estos si se requiere una morfología con mayor capacidad.

Los robots modulares se clasifican según su diseño, presentando los que poseen una interfaz de conexión para una morfología específica indicando que son del tipo configurable, o si emplean algoritmos geométricos acoplando los bloques cercanos para cambiar la estructura siguiendo reglas de construcción para la necesidad que se presente, estos últimos son de tipo auto-configurable. Las recientes investigaciones han creado programas capaces de diseñar el sistema de comando y comunicación utilizando datos de entrada como el número de módulos y la forma de estos, dando a conocer todos los diseños posibles para un número determinado de eslabones y el algoritmo a seguir para construir un sistema en específico (Rai, Correll, & Anne van Rossum, 2011, págs. 4783-4785). Dado a que la mayoría de investigaciones se han concentrado en diseños estáticos y no en evaluar los movimientos que se logran utilizando morfologías basadas en módulos, este trabajo abordará este tema, implementado un robot modular capaz de desplazarse con independencia del terreno tomando como inspiración la locomoción de los insectos que tiene una morfología hexápodo.

El módulo MECABOT 4.0 desarrollado en el grupo de investigación Davinci de la Universidad Militar Nueva Granada es el eje de investigación de este trabajo, en el que se vienen trabajando distintas configuraciones robóticas evaluando sus capacidades locomotoras al estar compuestas por módulos. Actualmente se trabajan las configuraciones aro, serpiente, oruga y

hexápodo, siendo este último el centro de investigación de este trabajo de grado. Para el diseño de la configuración hexápodo a partir del MECABOT 4.0, se tuvo en cuenta las capacidades individuales de un módulo y los movimientos logrados al estar acoplados a un semejante, teniendo en cuenta que la forma final de la configuración se adapte al concepto hexápodo utilizando el menor número de módulos garantizando su movilidad. Para lograr un mejor diseño, se realizaron diferentes simulaciones en el software Webots, programa que permite observar la respuesta del robot ante el software de movimientos, esto en un entorno ideal teniendo en cuenta los parámetros reales que influyen en su locomoción como el peso y la forma del robot. El mejor diseño se escogió teniendo en cuenta que fuera un robot hexápodo, que utilizara el menor número de módulos posibles y permitiera movimientos omnidireccionales al coordinar los módulos de las patas. Para avalar los estados de movimientos implementados se realizó el modelo matemático de la morfología aplicando cinemática inversa y directa sobre las posibilidades de acople dadas por los módulos, escogiendo la conexión entre módulos que mejor se adapte a los movimientos de un robot hexápodo. Se implementó la configuración física, construyendo los módulos MECABOT 4.0 que al final fueron acoplados y evaluados sobre terreno plano e irregular para el avance del hexápodo, evaluando distintos parámetros para encontrar los que brinden un mayor rendimiento al desplazarse. Estos resultados ayudarán en el desarrollo de futuros proyectos en el área de la robótica modular y la implementación de diseños más complejos y de mayor capacidad.

1.1 Planteamiento del Problema

La robótica modular, plantea la elaboración de robots compuestos por módulos individuales y construcción simple, donde el número de diseños depende de la cantidad de módulos que se usan por lo que de un número definido de módulos se pueden construir diferentes modelos con aplicaciones distintas, a pesar de esto cabe aclarar que cada módulo funciona de manera independiente, por lo que para lograr un desplazamiento se necesitan dos módulos que se muevan en conjunto de forma sinusoidal lo que plantea limitantes como cambiar de dirección, el tiempo y energía utilizada para hacer un recorrido. Debido a los problemas de locomoción y coordinación se logra dar solución a este problema con diseños inspirados en la naturaleza, más en específico en los insectos cuyo diseño simple se puede representar en módulos que dividen su cuerpo y seis extremidades que les permite desplazarse con facilidad en terrenos irregulares. Para dar solución a

los problemas que presentan los robots convencionales que utilizan llantas u orugas como método de locomoción, se propone la implementación de un diseño hexápodo capaz de reestructurarse fácilmente intercambiando y acoplando módulos. En el uso de módulos es pertinente evaluar si dicho modelo de construcción, puede utilizarse para la creación de robots móviles ya que plantea un sistema barato y versátil al que se tiene que sacar el mayor provecho. Actualmente se está llevando en la UNIVERSIDAD MILITAR NUEVA GRANADA un estudio de locomoción sobre estos módulos evaluando diferentes morfologías dentro del grupo de investigación DAVINCI. Debido a su simplicidad, estos robots presentan algunos problemas de coordinación y locomoción ya que en su mayoría son del tipo cadena, y sus movimientos se basan en arrastrarse por el piso como la haría una serpiente o una oruga, por lo que tampoco son muy rápidos. Entonces ¿puede una morfología inspirada en la marcha de criaturas hexápodas presentar una mejor propuesta de movilidad para creación de robots modulares y ampliar los terrenos a los que pueden llegar estos? ¿Es posible construir un robot modular funcional, que sirva como acople a otros robots que requieran desplazarse por terrenos difíciles en futuras investigaciones en la Universidad Militar Nueva Granada?

1.2 Objetivo General

Desarrollar y evaluar un robot móvil de morfología hexápodo capaz de desplazarse con independencia del terreno, con el objetivo de evaluar si dicho modelo es funcional y puede ser implementado como acople en robots que trabajan en superficies irregulares.

1.3 Objetivos Específicos

1. Modelar matemáticamente la morfología hexápodo para un sistema robótico modular.
2. Simular el movimiento de la configuración hexápodo para un sistema robótico modular
3. Evaluar los movimientos posibles con la morfología hexápodo estudiado.
4. Implementar físicamente la morfología hexápodo y pruebas de locomoción con este.
5. Evaluar la funcionalidad y rendimiento de la morfología hexápodo sobre el terreno de prueba.

1.4 Justificación

Hoy en día la mayoría de implementaciones en robótica se realizan a partir de diseños robustos, implementados para trabajos específicos como manipuladores y robot cartesianos con movimientos en plano limitados, en contraste a esto, actualmente la robótica se está dirigiendo a modelos inspirados en la naturaleza donde la diversidad, resultado de la evolución presenta modelos de locomoción para cualquier tipo de terreno y condiciones ambientales. Con la intención de imitar esta versatilidad, se diseñan robots basados en módulos homogéneos con total independencia que permiten infinidad de diseños y debido a su simplicidad, se busca crear modelos auto reparativos y con capacidad para adaptarse cambiando de forma. En adición a lo anterior, son sistemas de fácil mantenimiento y construcción, lo que los hace más baratos que los robots contruidos con articulaciones grandes y fijas que al presentar un fallo, en una configuración modular solo es necesario cambiar el modulo afectado que se acopla fácilmente, en lugar de desmontar toda una pieza del robot. Es debido a las ventajas que presenta la robótica modular que se quiere implementar un sistema móvil utilizando este principio, que permitiría trabajar en lugares en donde las condiciones son hostiles y el acceso a repuestos no es posible, como una zona de desastre por terremoto o para realizar trabajos en el espacio.

Es importante realizar estudios que permitan a estos sistemas de módulos realizar movimientos y desplazarse, ya que no tienen limitaciones de diseño y las posibilidades de construcción son infinitas. Por esto el presente trabajo, busca aportar un sistema de movilidad utilizando la robótica modular que no va a estar limitada solo a la morfología trabajada, en lugar de esto, se quiere que el sistema sea utilizado como acople de nuevas estructuras que brinden un mayor alcance a la robótica actual.

1.5 Metodología

En la Figura 1-1, se presenta el diagrama de flujo de la metodología implementada en el trabajo. Se inició con la recopilación de información que aborda el tema del trabajo, así como antecedentes sobre trabajos similares y su aporte a la robótica modular móvil. Para desarrollar el robot modular hexápodo, se simularon y ensamblaron en un entorno virtual los módulos MECABOT, así como la programación de los movimientos coordinados en la configuración que posteriormente se implementaron en el robot físico para ser puestos a prueba.

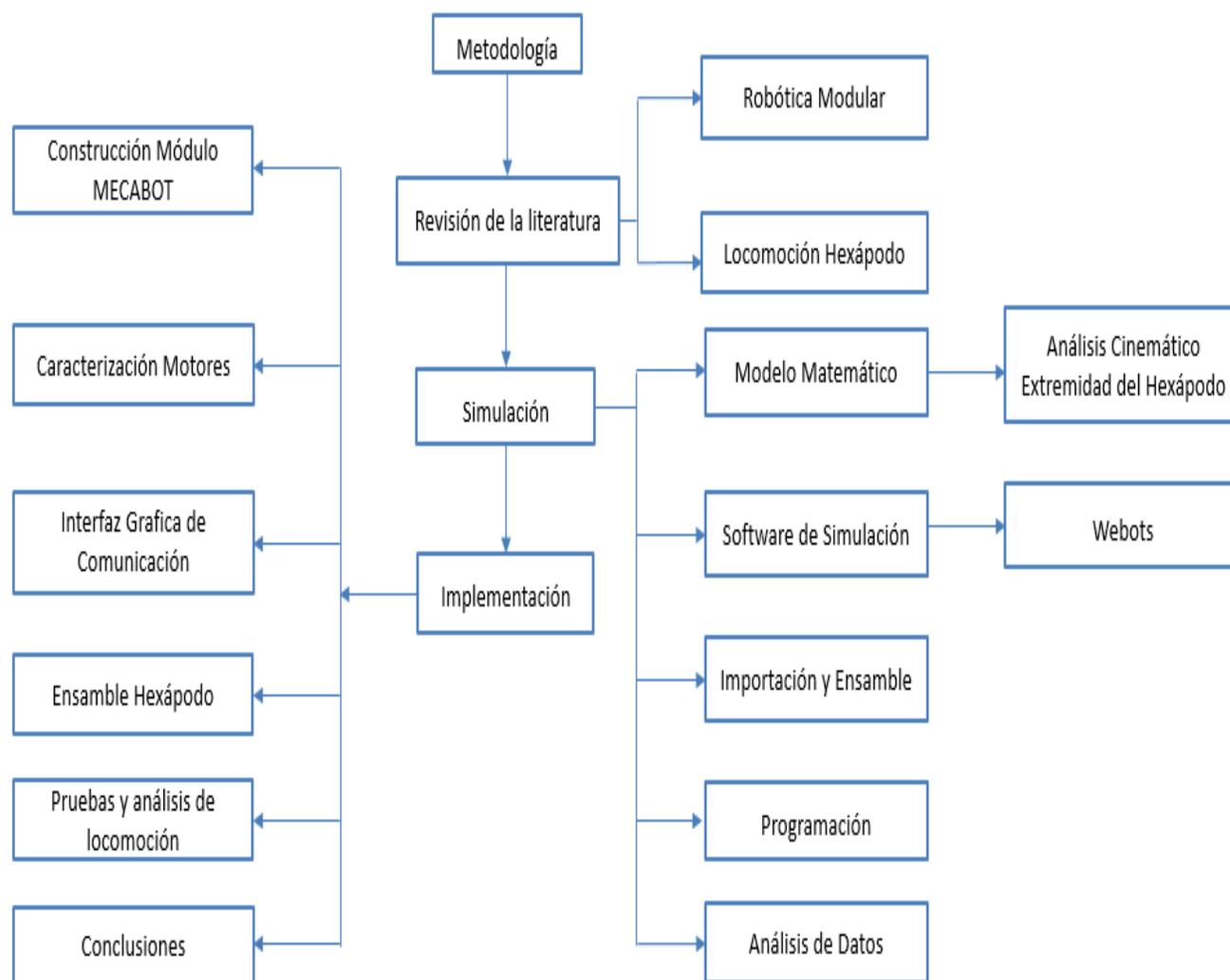


Figura 1-1. Metodología del trabajo

2 Referentes Teóricos

2.1 Marco Teórico

En este capítulo se describirán algunos conceptos dentro de los que se engloba el trabajo de la implementación del hexápodo utilizando un sistema robótico modular y que permiten comprender las características de los robots modulares y la locomoción de la topología hexápodo.

2.1.1 Robótica e Historia

Desde sus inicios el ser humano se ha caracterizado por el uso y desarrollo de herramientas que le permitan realizar trabajos con mayor facilidad y dar un mejor uso a los recursos disponibles, de ahí el interés y fascinación por los mecanismos capaces de realizar dichas tareas de forma autónoma como si de seres vivos se trataran.

El término “robot” fue inventado por el escritor checo Karel Kapek en su obra teatral R.U.R en 1920, la expresión proviene de la palabra checa “robota” que hace referencia al trabajo, más específicamente al de un siervo o esclavo (Weng, Creación del Modelo del Robot Humanoide Teo para el Simulador WEBOTS, 2012). Aún con el aporte de Kapek se viene manejando el concepto de máquina autónoma desde mucho tiempo atrás, siendo los griegos quienes usaron la palabra “automatos” de la que deriva autómatas para referirse a aquellos mecanismos compuestos de partes hidráulicas, poleas y palancas que imitaban figuras y movimientos de un ser animado en los que se destacan los mecanismos de Herón de Alejandría (85 d.C) (Cárdenas Verdugo, 2011) . Actualmente la palabra “robot” es usada para referirse a las máquinas capaces de realizar tareas o determinadas funciones con cierto grado de autonomía.

La robótica moderna toma sus bases de la revolución industrial y en sus inicios con las máquinas textiles, donde Joseph Jacquard inventó una máquina textil automatizada utilizando tarjetas programables (Weng, Creación del Modelo del Robot Humanoide Teo para el Simulador WEBOTS, 2012). Esto permitió repartir la fabricación de elementos en pequeñas tareas dando paso a las estaciones de montaje que hoy en día utilizan sistemas de automatización y control avanzados en casi todos los momentos de un proceso industrial.

2.1.2 Locomoción de los robots móviles

La robótica móvil como su nombre lo indica, abarca todo sistema robótico destinado a realizar desplazamientos en su entorno, comprendiendo aquellos capaces de llegar a su destino por tierra, agua, aire o cualquier terreno para el cual el robot fue diseñado. Existen diversas formas para lograr movimientos, algunos usando modelos inspirados en la naturaleza como los robots humanoides que simulan la marcha humana y otros a partir de mecanismos comúnmente usados en vehículos como la rueda y orugas. Estos robots desempeñan trabajos de exploración, vigilancia, labores de rescate entre otros, por lo que la gran mayoría están equipados con sensores que le permiten conocer su ubicación y el tipo los objetos que lo rodean.

Este trabajo se centra en un robot terrestre, este tipo de robots son diseñados para desplazarse por uno o varios tipos de superficie determinadas. Esta categoría comprende a los robots capaces de caminar, correr, escalar, deslizarse o rodar por lo que existen mecanismos para todo tipo de terreno entre los más comunes las patas, orugas y ruedas. En la figura 2-1 se aprecian los tipos más comunes de robot terrestres.

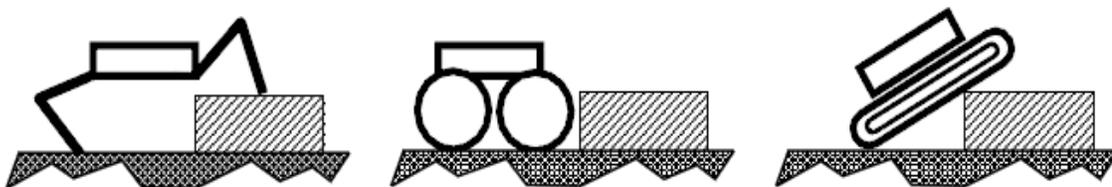


Figura 2-1 Desplazamiento de un robot móvil terrestre

Tomado de: (Cárdenas Verdugo, 2011)

2.1.3 Robots caminantes

Dentro del diseño de robots móviles, uno de los más utilizados es el que se da a base de marchas que implican la coordinación de patas para lograr desplazarse. El uso de patas en la robótica data de los diversos modelos bio inspirados presentes en la naturaleza y el éxito que han tenido las distintas especies para adaptarse a las condiciones más difíciles, logrando desplazarse de forma eficiente sobre superficies irregulares de todo tipo.

El uso de extremidades implica una coordinación motriz rítmica en cada paso que puede apreciarse, como una señal cíclica que cuenta con periodo y fase. Esta señal depende también de la estructura del animal, dando como resultado una serie de movimientos conocidos como patrones

de locomoción o patrones de marcha, en el que se logra especificar sus capacidades motoras, entre las que se encuentran el trote, correr, reptar, saltar entre otras (De León Melgarejo, 2010). Los robots caminantes aprovechan los diseños dados por la adaptación para determinar sus capacidades y limitantes. Un ejemplo de estos son los robots bípedos inspirados en la marcha humana, en donde uno de los mayores problemas es el equilibrio, problema que no se presenta en los cuadrúpedos por su estabilidad, pero se sigue trabajando en ellos debido a la facilidad que tienen los humanos para saltar, subir escalera y sortear obstáculos en comparación con especies de cuatro patas.

Los robots caminantes presentan varias ventajas en comparación de otros sistemas como los que utilizan ruedas, estas están dadas gracias a las capacidades como la flexibilidad y apoyo que otorga cada pata en las que se destacan:

- Mayor movilidad al ser capaces de cambiar de dirección sin importar la orientación del cuerpo.
- Capacidad para superar obstáculos al pasar sobre ellos subiendo las patas.
- Suspensión activa al mantenerse apoyado sobre el terreno en todo momento, esto es gracias a la capacidad de estirar y contraer cada pata.
- Marcha con independencia del terreno al no necesitar un espacio continuo para moverse, estos son capaces de moverse sobre arena, piedras, pasto en otras superficies difíciles para las ruedas.
- Se reduce el deslizamiento al levantar y apoyarse cada pata en la superficie en lugar de girar y verse afectado por la resistencia del terreno.
- Menor desgaste por el terreno debido a un menor tiempo de contacto.
- Velocidad promedio en terrenos irregulares, a pesar de ser más lentos que los que usan ruedas en superficies planas, estos son más constantes al marchar sobre cualquier terreno.

En la figura 2-2 se representan estas ventajas para dar un mejor entendimiento de los robots caminantes.

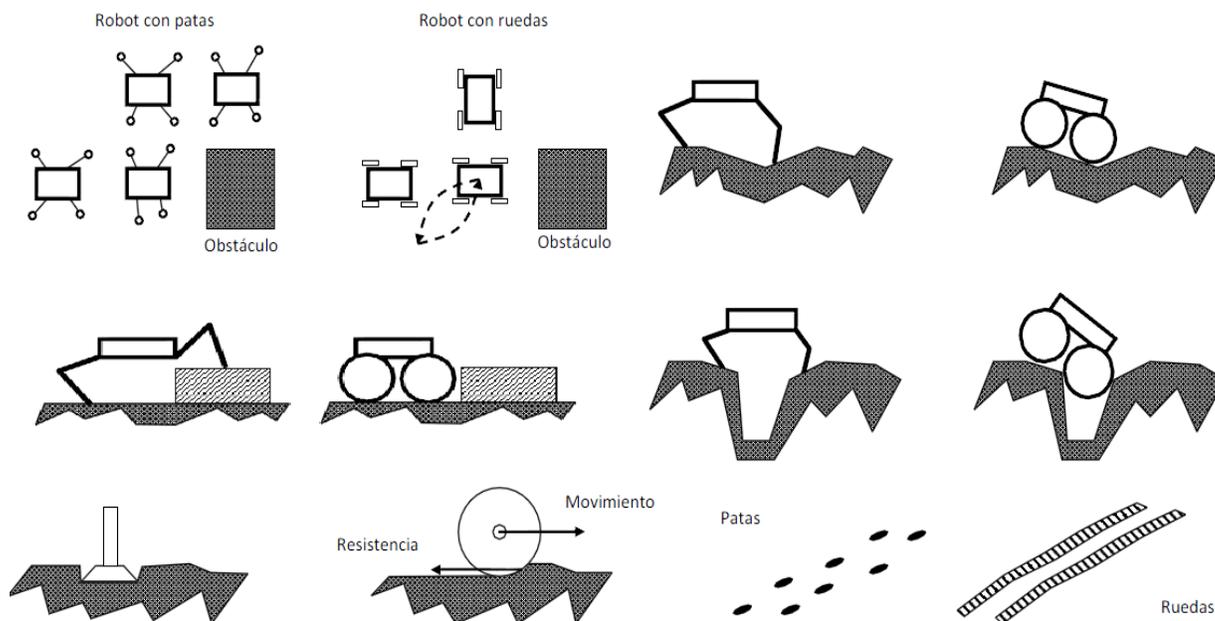


Figura 2-2 Ventajas de los robots caminantes.

Tomado de: (Cárdenas Verdugo, 2011)

2.1.4 Robots hexápodos

Dentro de la gama de robots móviles, los insectos han sido uno de los modelos de inspiración más utilizados, debido al éxito que han tenido estos para adaptarse a todo tipo de terreno, destacando su velocidad de desplazamiento y la estabilidad en sus movimientos que mantienen incluso estando de cabeza. Su característica principal es que poseen 6 patas a lo que se denomina como morfología hexápodo, el número de patas implica mayor puntos de contacto con la superficie durante la locomoción, esta cualidad les permite mantenerse estables siempre, resolviendo uno de los problemas más comunes en el diseño de robots simplificando el control (Cárdenas Verdugo, 2011). Los robots hexápodos tienen en cuenta sus articulaciones y grados de libertad imitando los 3 grados por pata de los insectos, su marcha consiste en mantener apoyadas 3 de sus patas mientras usa las restantes para avanzar, tanto en los robots como en los insectos este movimiento genera un triángulo de apoyo que mantiene el centro de gravedad dentro del área de apoyo (Sedeño Bustos, 2011). En la figura 2-3 se observa el triángulo de estabilidad que se forma sobre las tres patas apoyadas del insecto tenebrio que posee morfología hexápodo.



Figura 2-3. Triangulo de apoyo insecto tenebrio.

Tomado de: (Rodríguez Calderón, Ramos Parra, & Peña Giraldo, 2015)

Los insectos y animales con más de cuatro patas son capaces de caminar inmediatamente nacen, teniendo resuelto el problema de balanceo a diferencia de los cuadrúpedos, que demoran un tiempo más en ponerse de pie y algo más de tiempo para empezar a marchar, estando por último los bípedos como los humanos que tardan meses para ponerse de pie y aprender a caminar al no contar con posiciones estáticamente estables (Cárdenas Verdugo, 2011). Esto es tomado en cuenta por los investigadores, ya que a pesar de que los cuadrúpedos también forman ese polígono de estabilidad, los robots con mayor número de patas son más robustos y se les facilita lograr los movimientos, obteniendo todas las ventajas de los robots caminantes con mayor estabilidad. Estas ventajas y capacidades de movimiento se logran apreciar en la Figura 2-4 que representa las ventajas de la topología hexápodo.

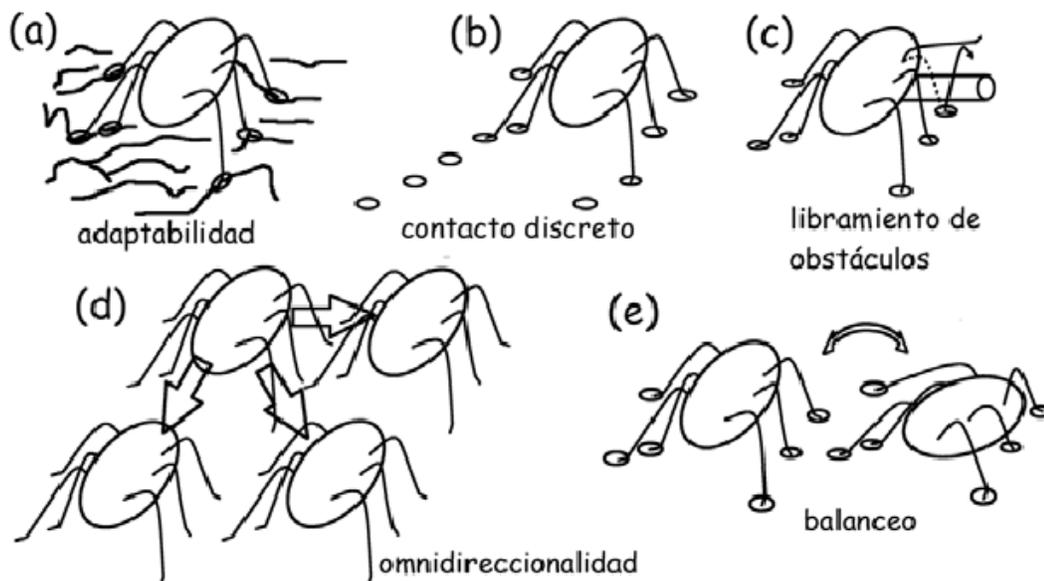


Figura 2-4 Ventajas de la locomoción por patas topología hexápodo

Tomado de: (De León Melgarejo, 2010)

2.1.4.1 Tipos de locomoción

Como ya se ha visto, la morfología hexápodo puede generar movimientos omnidireccionales como traslación y rotación, en donde el movimiento es observable con las posiciones del cuerpo del robot ya que las patas permanecen sujetas a este, y solo ejecutan un movimiento coordinado siguiendo una trayectoria en cortos periodos de caídas controladas en el que se levanta y se deja caer cada pata, cuidando que el sistema no se torne inestable (Sedeño Bustos, 2011). El movimiento de cada pata forma arcos con respecto a la superficie, en donde se manejan parámetros como la línea en contacto directo con el suelo que define el desplazamiento y dirección que va a tomar el hexápodo, junto a los puntos suspendidos y de contacto que siguen a la trayectoria.

En la figura 2-5 se describe el movimiento de la pata para dar un paso, en donde la extremidad pasa de estar apoyada en el suelo, a ser levantada y llevada en la dirección deseada.

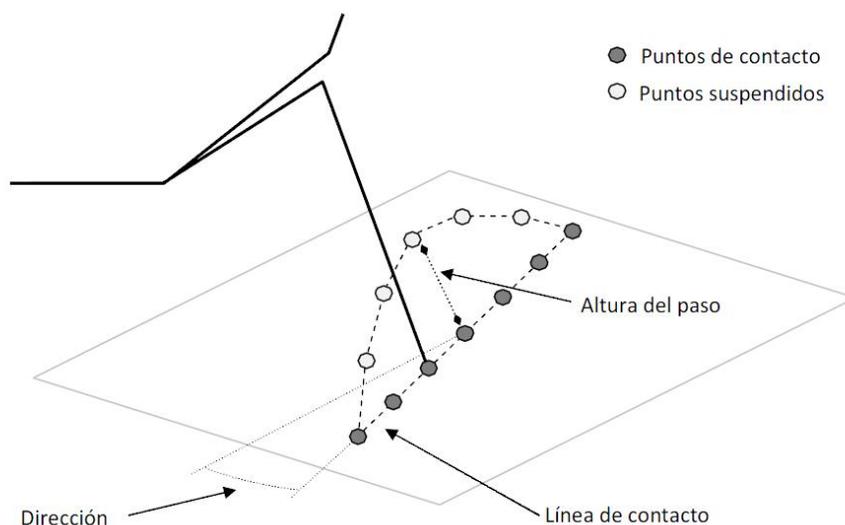


Figura 2-5. Trayectoria que sigue una pata para que el robot se desplace

Tomado de: (Cárdenas Verdugo, 2011).

En un hexápodo, el número mínimo de patas sobre las superficies necesarias para lograr moverse es de tres, en donde la técnica de movimiento cambia dependiendo cuántas patas mueve dejando las demás estáticas. Las marchas agrupan las patas en cada paso, separando las que se mueven y las que no, si se mantienen cinco, cuatro o tres sobre el suelo moviendo 1, 2 y 3 respectivamente (Cárdenas Verdugo, 2011). La relación de patas en movimiento y las que permanecen quietas se describe en la Tabla 2-1.

Técnica	Número mínimo de patas en contacto con el suelo	Número máximo de patas separadas del suelo
3 patas por movimiento	3	3
2 patas por movimiento	4	2
1 pata por movimiento	5	1

Tabla 2-1. Patas en contacto con el suelo y suspendidas en cada forma de desplazamiento.

Tomado de: (Cárdenas Verdugo, 2011).

2.1.4.2 Clasificación de los hexápodos según su grado de libertad

Los robots hexápodos son cualquier clase de robots que posea 6 patas, diferenciándose unos de otros por la complejidad del diseño, en donde las capacidades de cada hexápodo se definen por los grados de libertad que permiten sus articulaciones. El aumento de grados de libertad implica el uso de más actuadores, lo que complica los sistemas en aspectos como la alimentación y mecanismos, junto con el programa de comando que coordina los movimientos. Dentro de la gama de robots hexápodos se encuentran:

- **Hexápodos de un grado de libertad:** Son hexápodos con articulaciones rígidas que solo logran un tipo de movimiento, estos son de construcción simple y no usan muchos actuadores, no requieren una programación compleja.
- **Hexápodos de dos grados de libertad:** Estos hexápodos requieren el uso de más actuadores lo que vuelve más complejo todo el sistema de control para coordinar los movimientos.
- **Hexápodos de más grados de libertad:** Estos implican un sistema de control avanzado ya que tratan de imitar el diseño mecánico de los insectos con todas sus capacidades motoras.

En la Figura 2-6 se muestran algunos robots pertenecientes a cada clasificación por grados de libertad.

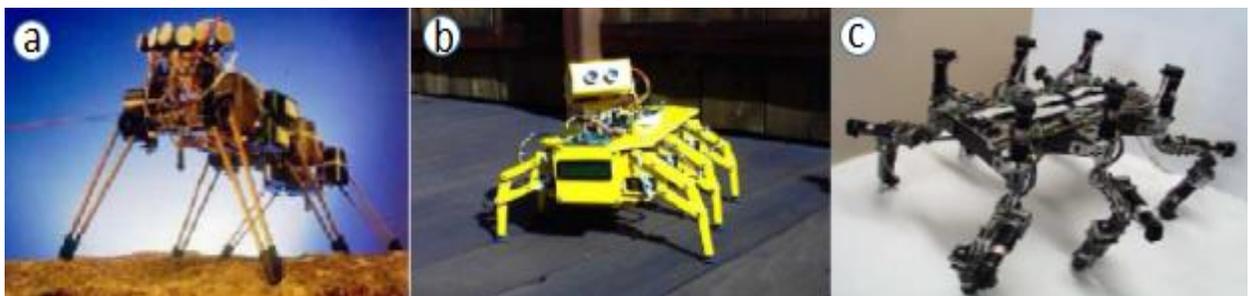


Figura 2-6. Hexápodos de a) 1 grado, b) 2 grados y c) más grados de libertad.

Tomado de: (Escobar Olarte & Silva Ortega, 2011)

2.1.5 Robótica modular

La robótica modular es la línea de la robótica dedicada a diseñar robots contruidos a partir de pequeños bloques, cada robot es independiente, con una construcción simple y geométrica que unido a otros robots iguales, pueden crear un robot más complejo y con mayores capacidades donde el número de módulos plantean posibilidades de diseño ilimitadas.

Los robots modulares se hicieron con el fin de ser reconfigurables, surgiendo por primera vez en la década de los ochenta en el contexto de brazos manipuladores. La idea fue popularizada por el trabajo de Mark Yim en Stanford a principios de los noventa, cuyo trabajo se extendió la idea de robots totalmente autónomos incluyendo su locomoción. Desde entonces, la comunidad que trabaja la robótica modular se ha centrado principalmente en robots homogéneos (los contruidos de muchas copias de un solo tipo de módulo).

En 1995 Mark Yim propuso en su tesis doctoral un enfoque distinto al problema de locomoción presente en el diseño de robots, que en ese momento consistía en analizar el terreno y diseñar un robot específico para dicho terreno. Pero en la práctica la topología de trabajo puede presentar variaciones, como podría ser pasar de un terreno plano a uno con imperfecciones para luego subir una escalera, por lo que se necesitarían tres robots distintos para llegar al punto deseado. Como solución a esto propuso el diseño de robots conformados por pequeños módulos, capaces de ensamblarse uno con otros para formar distintas configuraciones. De esta forma al determinar que configuración es la más adecuada para cada terreno, estos solo necesitarían re ensamblarse para seguir su camino, como fue el caso del robot usado para llegar del edificio de robótica de Stanford hasta el edificio cruzando la calle, en el que se usaron las configuraciones rueda, oruga y cuadrúpedo para llegar a su destino (González Gómez, 2008, págs. 30-31).

2.1.5.1 Clasificación de los robots modulares

Los robots modulares, están diseñados para realizar diversas configuraciones a partir de la conexión entre módulos, por lo que su clasificación está más enfocada en el tipo de estructura y la conexión que se establece para generar movimientos. Mark Yim propuso una división basada en la movilidad de los módulos, dividiéndolos en tres grupos: reticulares, coloniales y de tipo cadena que a su vez se clasifican según las dimensiones en las que trabaja cada topología (González Gómez, 2008).

2.1.5.1.1 Robots modulares reticulares (Lattice)

Los robots reticulares o de malla, se caracterizan porque sus módulos solo pueden alcanzar un conjunto limitado de posiciones espaciales. Este tipo de robots se mueven dentro de un retículo de 3 dimensiones acomodados como una red cubica o hexagonal, esta configuración se asemeja a los átomos de un cristal. El costo computacional para estos módulos es bastante simple y puede ser ampliada a sistemas más complejos utilizando un control en paralelo (Choque Aspiazú, 2009). Estos robots son del tipo auto-configurables y pueden clasificarse según el tipo de estructura en 2D y 3D (González Gómez, 2008). Un ejemplo de robot lattice es la arquitectura de red formada por los módulos DFA de la Figura 2-7.



Figura 2-7 Robot modular reticular: Módulos DFA en arquitectura de red

Tomado de: (Feczko, y otros, 2015)

2.1.5.1.2 Robots modulares tipo cadena

En esta arquitectura de cadena o árbol las unidades se conectan entre si formando una cadena, que a su vez pueden estar ensambladas entre sí a través de nodos para formar árboles. Estos árboles permiten crear configuraciones de mayor complejidad y con mayores grados de libertad, siendo de un ejemplo de esto la configuración cuadrúpedo, que es la unión de tres cadenas de módulos, permitiéndole moverse sobre el plano en varias direcciones, movimientos que no podrían realizarse con una sola cadena, la cual solo lograría moverse en un eje. La arquitectura en cadena puede alcanzar cualquier punto en el espacio, sin embargo tienen un mayor costo computacional. Un ejemplo de este tipo de robots son el polybot y CosMO de la Figura 2-8. Los robots modulares de tipo cadena a su vez son clasificados según si su tipología es 1D, 2D y 3D.

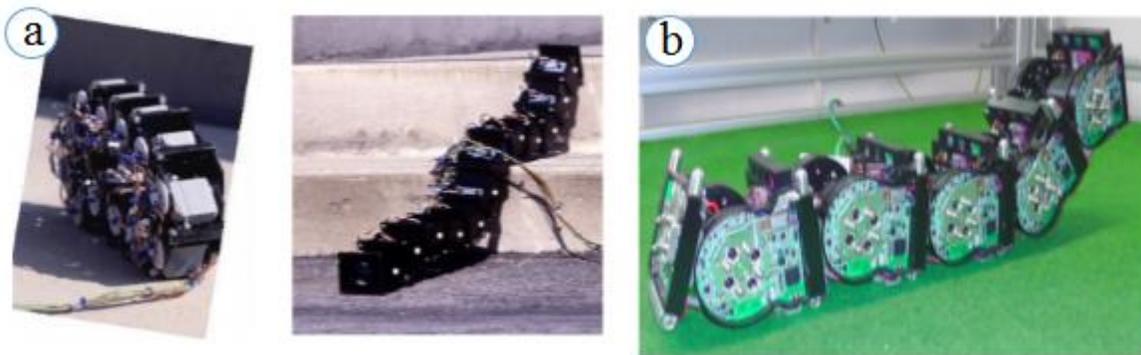


Figura 2-8 Robot modular tipo cadena. a) Polybot G1 de rueda a oruga, b) Oruga con módulos CoSMO
Tomado de: a) (González Gómez, 2009), b) (Vonásek, Neumann, Oertel, & Worn, 2015)

- **Topología 1D:** Cadena de módulos en una sola dirección como lo pueden ser morfologías de serpientes, gusanos, brazos o patas. Entran en la categoría de Robots ápodos, los cuales realizan movimientos coordinados y oscilatorios para generar movimientos (**Lancheros Guzmán & Sanabria Galvis, 2016**). Este tipo de robots se aprecian en la Figura 2-9.

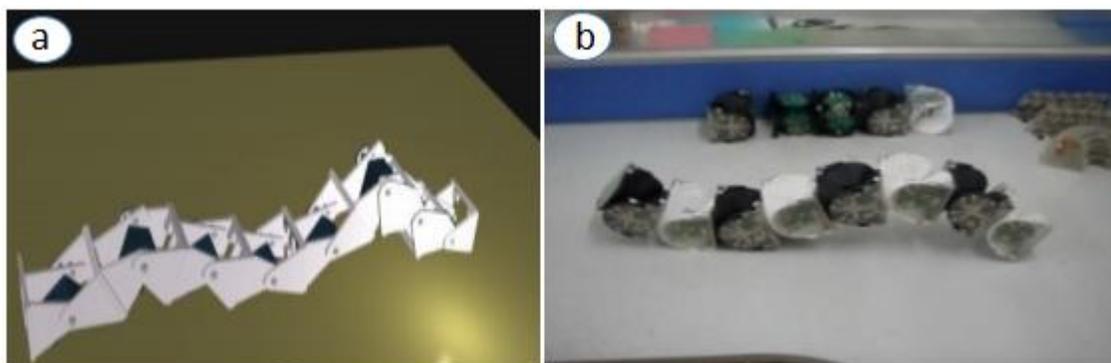


Figura 2-9 Robots tipo cadena topología 1D:

Tomado de: a) (González Gómez, 2009), b) (Jie, Xindan, Yanhe, & Shufeng, 2011)

- **Topología 2D:** Más de una cadena de módulos conectados en diferentes ejes, permite movimientos en varias direcciones sobre el plano. Esta topología permite una gran variedad de diseños robots como robots con patas o manipuladores al usar módulos como las articulaciones de una pinza. En Muchas investigaciones se estudia los robots modulares móviles utilizando esta topología para lograr desplazamientos, siendo la más utilizada la configuración cuadrúpeda que implementa una cadena central y dos perpendiculares para las cuatro patas. Este trabajo implementará la topología hexápodo, siendo del tipo 2D al usar una cadena central para el cuerpo y los módulos que son cadenas de 2 semi-módulos como patas como los ejemplos en la Figura 2-10 y Figura 2-11.

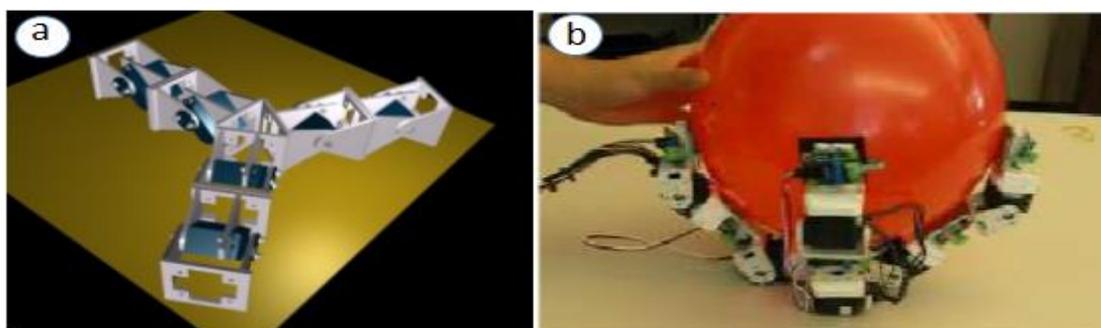


Figura 2-10 Robots tipo cadena topología 2D.

Tomado de: a) (González Gómez, 2009), b) (Chih-Han & Radhika, 2009)

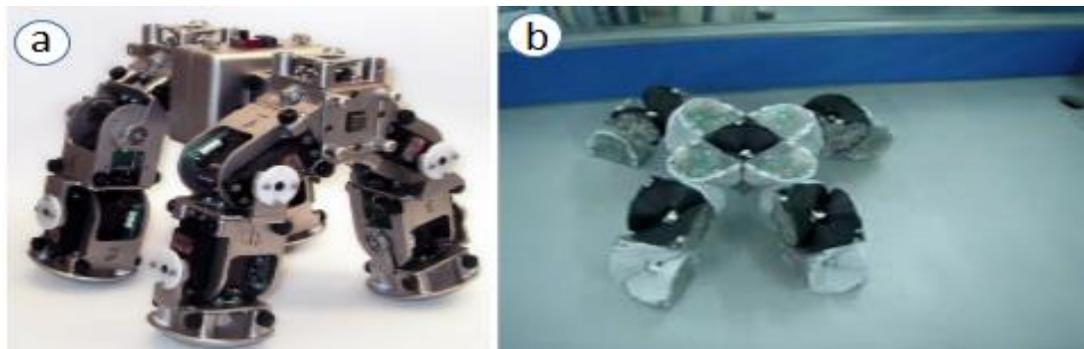


Figura 2-11 Robots cuadrúpedos utilizando topología 2D tipo cadena.

Tomado de: a) (D. Hancher & S. Hornby, A Modular Robotic System with Applications to Space Exploration, 2006), b) (Jie, Xindan, Yanhe, & Shufeng, 2011).

- **Topología 3D:** Demás configuraciones en el espacio. Permite configuraciones con movimientos tridimensionales, lo que les permite además de desplazamientos sobre el plano, poder subir y bajar el robot como el modelo de la Figura 2-12. Esta topología se está estudiando para exploración espacial, debido a que en condiciones de gravedad cero es necesario poder desplegar las articulaciones de una configuración en cualquier dirección como el robot de la Figura 2-13.

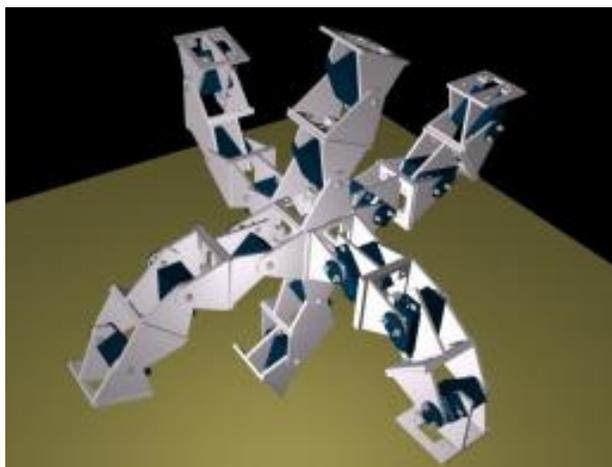


Figura 2-12 Robot tipo cadena topología 3D.

Tomada de: (González Gómez, 2008)

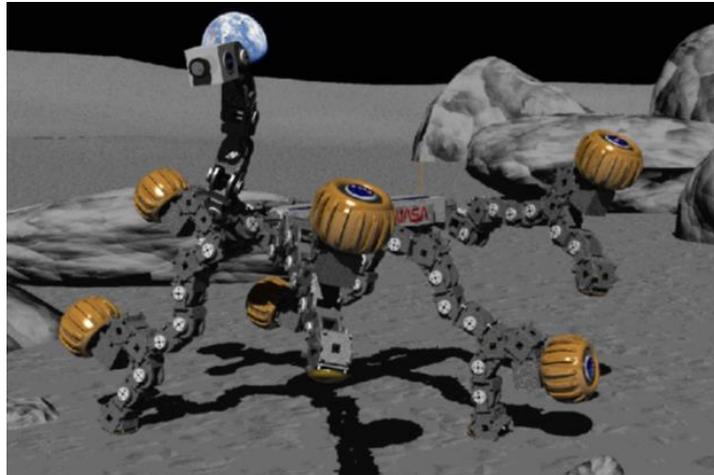


Figura 2-13 Robot tipo cadena topología 3D. Concepción artística de robots modulares en el espacio.

Tomado de: (D. Hancher & S. Hornby, A Modular Robotic System with Applications to Space Exploration, 2006)

2.1.5.1.3 Robots Coloniales

Los robots coloniales o de enjambre, son un grupo de robots móviles idénticos que operan de forma cooperativa al igual que un enjambre de abejas o una colonia de hormigas. Estos robots se mueven de forma independiente, pero responden a un controlador central que coordina los movimientos de toda la colonia de robots para realizar algún tipo de tarea que de manera individual no podrían hacer. Estos robots resultan muy convenientes en labores de exploración ya que se comunican entre sí y pueden cubrir más terreno de forma rápida y eficiente a mayor número de robots como los grupos de robots en la Figura 2-14.

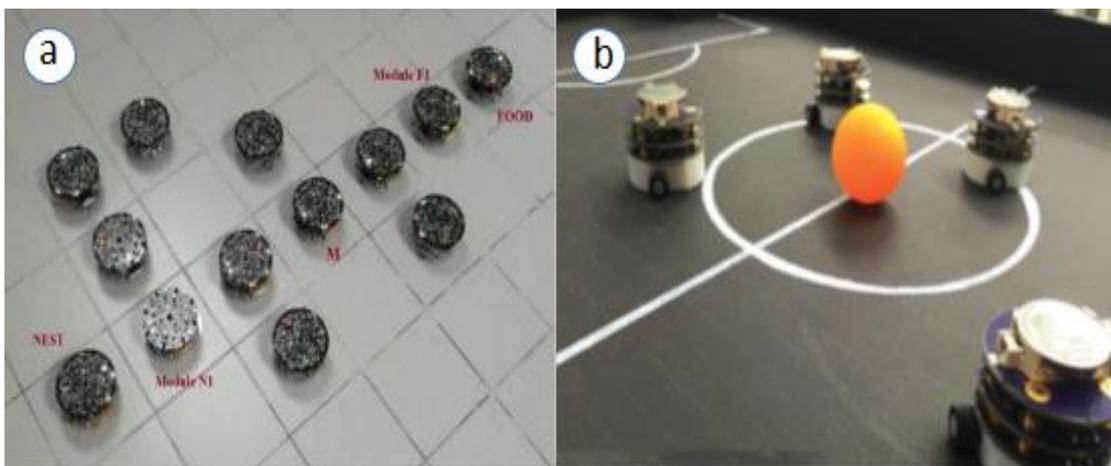


Figura 2-14 Robos modulares del tipo colonial.

Tomado de: a) (Van Tung & Trung Dung, 2014), b) (Ekawahyu, y otros, 2016).

2.1.5.2 Ventajas de los robots modulares

En los últimos años el interés por los robots modulares ha ido aumentando a medida que las investigaciones avanzan y los beneficios de estos sistemas se hacen más evidentes, en donde aplicaciones para la educación, diseños inspirados en módulos y exploración espacial son algunas de las múltiples aplicaciones previstas usando este tipo de robots (Zhang, Gonzales Gomez, Chen, & Zhang, 2008). Dada su versatilidad y flexibilidad, los sistemas robóticos modulares presentan varias ventajas con respecto a los robots convencionales, entre las que destacan:

- **Versatilidad:** Dado que las configuraciones se construyen a partir de pequeños bloques, estos son capaces de cambiar de forma al re ensamblarse, adaptándose a los cambios que se presenten teniendo en cuenta la morfología más eficiente, dentro de su repertorio para terminar el trabajo. Esta capacidad resulta muy útil en trabajos de búsqueda y rescate donde el terreno es irregular y se pueden presentar derrumbes que alteren drásticamente el lugar de operación.
- **Reconfiguración:** La comunicación es algo muy importante en la robótica modular, en un grupo de módulos cada uno debe ser capaz de reconocer a su semejante y conectarse de forma eficiente para construir una morfología específica. Esto se logra al ejecutar una secuencia de pasos en donde cada módulo sabe qué papel ocupa en las etapas de construcción, teniendo en cuenta su posición y la geometría del módulo. Esto le permite cambiar de configuración con una sola instrucción, como el caso de pasar de una rueda que aporta velocidad a una serpiente para pasar por lugares estrechos.
- **Eficiencia energética:** Cada módulo es un robot independiente por lo que en la mayoría de diseños estos cuentan con su propia fuente de energía para mover sus propios motores y demás componentes. Para trabajos que demanden mayor consumo energético estos solo se asocian con otros módulos para soportar la carga.

- **Mayor robustez:** Los módulos son robots compactos siguiendo formas geométricas, por lo que no tienen partes demasiado grandes como brazos u otro tipo de articulaciones susceptibles a daños, en caso de golpes todo el daño se distribuye en el módulo y dado su diseño es muy difícil que pierda partes.
- **Económicos:** Los módulos tienen un diseño simple de fácil construcción en donde todas las piezas son iguales para una configuración robótica modular, esto facilita los trabajos de mantenimiento al solo tener que cambiar el módulo dañado en una configuración, que a diferencia de otros robots esto no cuentan con varios tipos de pieza como brazos o ruedas, toda aplicación es construida con los mismos módulos y sus piezas. Esto resulta muy práctico en zonas donde los repuestos son difíciles de conseguir como lo puede ser el espacio.
- **Ligeros:** Los módulos son robots muy livianos a comparación de los convencionales debido a que solo soportan el peso de sus componentes como motores y controladores sujetos a una carcasa de material liviano como el usado en impresiones 3D.

2.2 Antecedentes en Robótica Modular

En esta sección se presentan las ideas más relevantes de cada uno de los estudios desarrollados de manera cronológica que aportan de manera significativa a la realización del trabajo presente y futuras investigaciones. En este, se aprecia la evolución de los sistemas robóticos modulares y los aportes más significativos que sirvieron como inspiración para el desarrollo de los módulos MECABOTS diseñados y construidos en la UNIVERSIDAD MILITAR NUEVA GRANADA.

La robótica modular hace referencia a los sistemas robóticos que implementan pequeñas unidades semejantes en toda la estructura conocidos como módulos, estos están compuestos por partes electrónicas y mecánicas puestas de manera compacta para crear un sistema móvil, que a su vez se acoplan a otros módulos para crear configuraciones más complejas y con mayor capacidad de movimiento (Arredondo, 2006). Estas investigaciones han sido mayormente dirigidas al diseño y software de movimiento de los distintos módulos, resaltando las ventajas técnicas y económicas al usar módulos robóticos en sus sistemas. Algunas de las presentes están en etapa de desarrollo, pero sus avances en algoritmos para robot auto configurables, permiten simular las distintas configuraciones resultantes de la conexión entre módulos teniendo en cuenta las geometrías de cada robot y los contactos posibles de cada uno.

2.2.1 Módulos Polybot G1

Marc Yim fue el pionero en la robótica modular, con el concepto de grupos de robots del mismo tipo que se acoplan para formar nuevas configuraciones dependiendo de las necesidades que se presenten con el modelo Polypod, demostrando las ventajas de la robótica modular. Con el concepto establecido en 1997 junto con Duff y Roufas, implementó los robots modulares Polybot G1 (Yim, Duff, & Roufas, 2000, págs. 514-520). Estos módulos utilizaban tablas de control para realizar los movimientos de las configuraciones oruga, rueda y una araña cuadrúpeda. Estos robots tienen forma de cubo para ser conectados por las caras que en esta etapa no eran auto configurables y poseían un servomotor para moverlas. Cada configuración tenía que ser acoplada con tornillos, además de la alimentación y la electrónica era externa. Para probar que los módulos eran factibles se propuso la tarea de recorrer todo el trayecto desde el edificio de la universidad de Stanford hasta el edificio cruzando la calle, para el recorrido se fue cambiando de configuración, comenzando con

la rueda para el terreno plano, cambiando a oruga para pasar una escalera y luego la araña para llegar al punto deseado.

En la figura 2-15 se presenta la configuración rueda y serpiente, las cuales fueron implementadas de tal manera que la configuración rueda se desplace sobre terreno plano al tener mayor velocidad y la serpiente por su movimiento ondulatorio puede bajar la escalera.

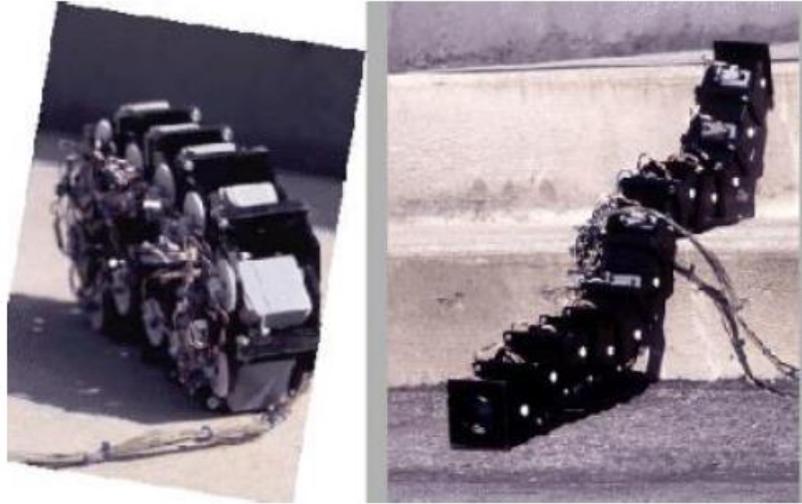


Figura 2-15. Robot Polybot G1 en configuración rueda y serpiente.

Tomado de: (Yim, Duff, & Roufas, 2000)

2.2.2 Modulo M-TRAN

Trabajo realizado por Akiya Kamimura, Haruhisa Kurokawa, Eiichi Yoshida, Satoshi Murata, Kohji Tomita y Shigeru Kokaji miembros de la IEEE. En este se propone un método de diseño de controladores estandarizados que de manera automática diseña el control de locomoción para cualquier morfología de un robot modular. Se demuestra la eficiencia del método mediante el software de simulación GLPA que da salida de locomoción para cualquier patrón en línea recta en una dirección determinada, donde se aplica el modelo de oscilador neuronal CPG y el algoritmo genético que determina la locomoción más eficiente para cualquier estructura. Los experimentos fueron realizados sobre el modelo M_TRAN II, Figura 2-16, robot modular auto-configurable diseñado por el equipo investigador (Kamimura, y otros, 2005).

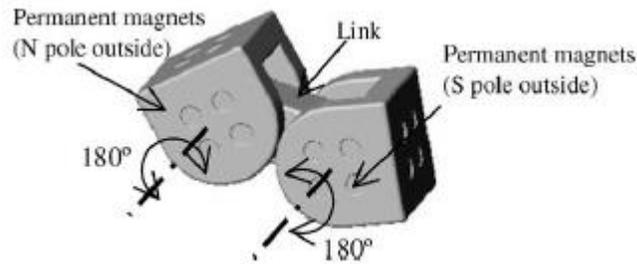


Figura 2-16 Vista esquemática modulo M-TRAN

Tomado de: (Kamimura, y otros, 2005)

2.2.3 Sistema robótico modular con aplicaciones a la exploración espacial

Su trabajo presentó un modelo para el diseño y automatización de robot modulares, solucionando los problemas de marcha para aplicaciones de exploración espacial y trabajo en condiciones hostiles, tomando como referencia, los requerimientos técnicos para futuras misiones de exploración espacial que se ven limitadas por las condiciones de trabajo donde incluso robots humanoides y manipuladores tienen dificultades para moverse en el espacio exterior. Se desarrolló el prototipo de hardware del robot modular tomando cada módulo en función de un solo núcleo reduciendo la complejidad del diseño y costos de elaboración. Se realizó la simulación del sistema de locomoción de un robot modular cuadrúpedo evaluando cada controlador sobre una marcha periódica con el fin de relacionar la distancia recorrida y la energía utilizada para poder determinar la marcha más eficiente implementándose más adelante en su propio robot modular cuadrúpedo en el que analizaron la evolución de la marcha. La investigación busca resaltar la utilidad y conveniencia de los robots modulares en trabajos de exploración, montaje, mantenimiento entre otras exploraciones espaciales que pueden ser fácilmente realizadas debido a la capacidad casi infinita de morfologías presentes en la robótica modular (D. Hancher & S. Hornby, *Evolving Quadruped Gaits with a Heterogeneous Modular Robotic System*, 2007). La secuencia de movimiento para avanzar esta descrita en la Figura 2-17.

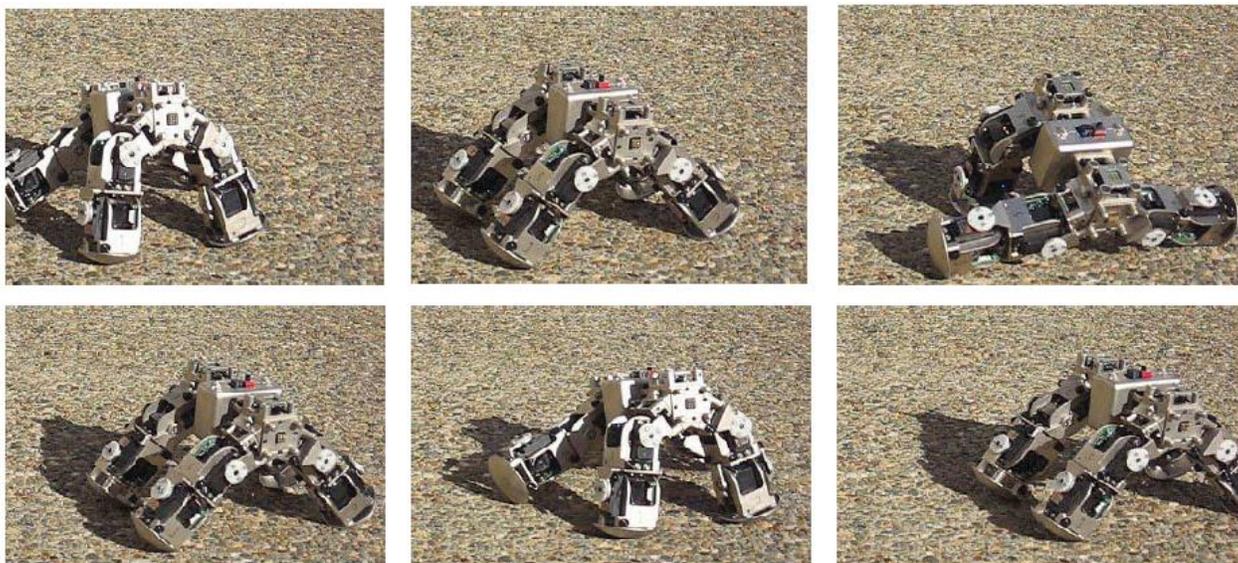


Figura 2-17 Robot cuadrúpedo avanzando.

Tomado de: (D. Hancher & S. Hornby, *Evolving Quadruped Gaits with a Heterogeneous Modular Robotic System*, 2007)

2.2.4 Diseño evolucionado y auto-reproductores robótica modular

Este trabajo presenta una gran variedad de máquinas de topología 2D y 3D con la capacidad de auto reparación que se logra gracias a la implementación de módulos equipados con electroimanes controlados de forma selectiva para crear la configuración deseada. Se realizaron varios diseños implementados en software de simulación y de fabricación hecha a mano, demostrando la versatilidad de los módulos con el objetivo de crear máquinas capaces de auto-replicarse, es decir que tras haber sufrido daño, estas puedan repararse de forma autónoma cambiando los módulos afectados. Este trabajo no solo aplica para robots convencionales sino que el mismo sistema se aplica para nano tecnología y micro tecnología aunque implicando mecanismos más sofisticados pero que implican que se usen modelos simples y que sean iguales entre sí (Zykov, Mytilinaios, Desnoyer, & Lipson, 2007). Los resultados del algoritmo simulado se aprecian en la Figura 2-18 en que se le aplican varias transformaciones.

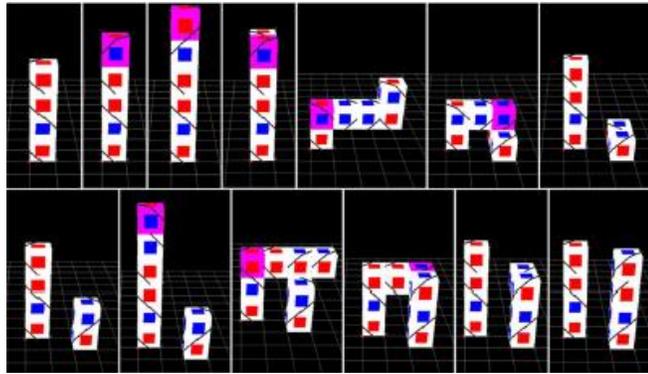


Figura 2-18 Diseño secuencia de auto reproducción asistida.

Tomado de: (Zykov, Mytilinaios, Desnoyer, & Lipson, 2007)

2.2.5 Cubo M

Este trabajo desarrolla en un sistema inteligente integrado a módulos con los últimos avances en el proyecto modular Cubo M un modelo de cuerpo único de un grado de libertad, que se compone de seis partes en las que se destaca un servo RC y un controlador eléctrico siendo un modelo fácil de montar. Este modelo es la versión mejorada del modelo de robot modular Y1, y el trabajo realizado se centra en la realización de un hardware de control distribuido ya que cada módulo cuenta con un controlador independiente al que se le envían comandos ya sea a uno solo o a todos los módulos al tiempo, demostrando que este tipo de configuración resulta ser versátil, robusto y fácil de implementar, dejándolo ver con la construcción de un robot serpiente hecho con los Cubos M y otras configuraciones armadas como el cuadrúpedo y trípode (Zhang, Gonzales Gomez, Chen, & Zhang, 2008).

En la Figura 2-19 se muestra el módulo Cubo M implementado y su configuración cuadrúpeda y serpiente.

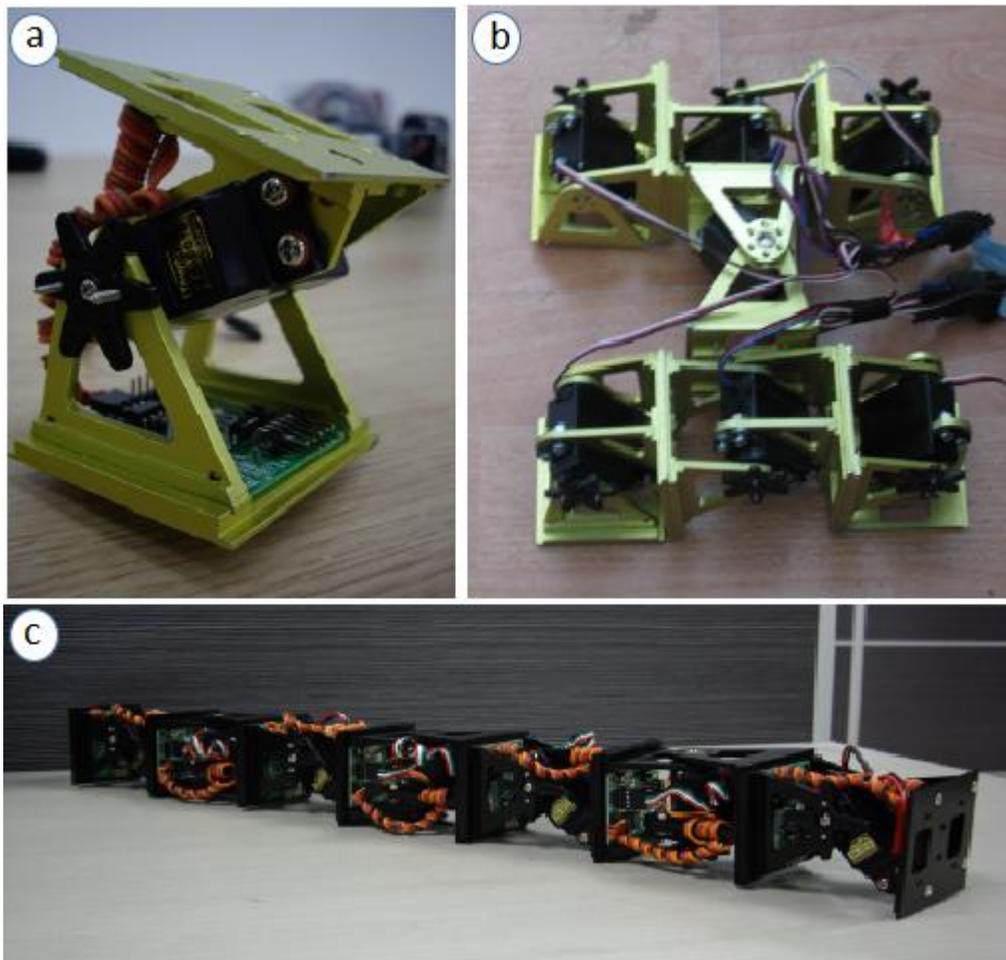


Figura 2-19. a) Módulo M, b) Configuración cuadrúpeda y c) configuración serpiente con módulos M.

Tomado de: (Zhang, Gonzales Gomez, Chen, & Zhang, 2008)

2.2.6 UBot (2011)

Los módulos UBot (Figura 2-20), son un grupo de robots modulares auto-reconfigurables de estructura simple, capaces de generar movimientos a través de sus conexiones creando configuraciones. Cada módulo posee una estructura cúbica con dos juntas de rotación y cuatro caras de conexión, para conectar los módulos activos y pasivos utilizando unos ganchos de conexión en los módulos activos sobre los pasivos que se acomodan antes de la unión por medio de imanes. El sistema del Ubot cuenta con un ordenador central que le permite simular y poner en marcha varios sistemas de locomoción, dentro de los cuales

fueron estudiados y analizadas las configuraciones rueda, serpiente y cuadrúpedo teniendo en cuenta la secuencias de conexión para la auto configuración y las capacidades de cada una (Jie, Xindan, Yanhe, & Shufeng, 2011).

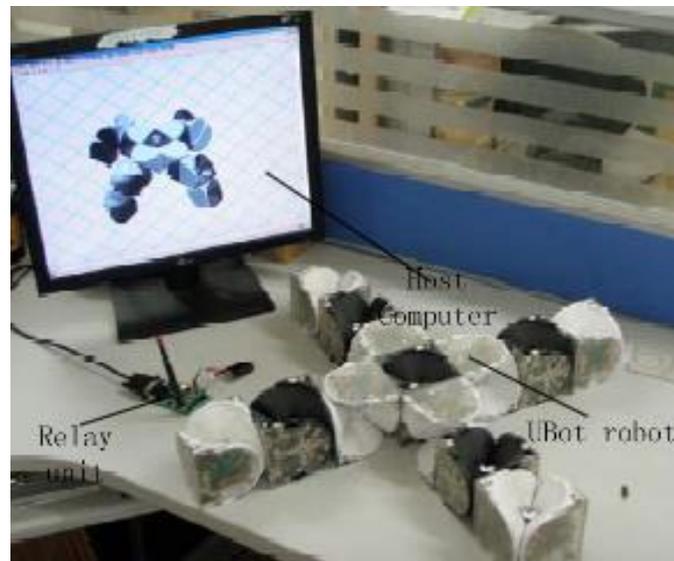


Figura 2-20. Sistema UBot en configuración cuadrúpedo.

Tomado de: (Jie, Xindan, Yanhe, & Shufeng, 2011).

2.2.7 Módulo robótico para la exploración autónoma y con varios modos de locomoción

Su trabajo consistió en el diseño y evaluación de robots modulares, evaluando sus capacidades móviles y rendimiento, para trabajos de exploración y versatilidad en entornos irregulares. Para esto se diseñaron cinco prototipos guiados por imagen y dotados con sensores que les permitían detectar la posición y ángulo de objetos presentes en el terreno, estos también fueron puestos a trabajar en conjunto para evaluar las capacidades cooperativas de los robots modulares. Dentro de las pruebas realizadas con el *Scout* robot, módulo robótico implementado se pusieron a prueba sus capacidades individuales subiendo una rampa y evaluando su desplazamiento en terreno plano. También se probaron las configuraciones oruga y cuadrúpedo con estos módulos midiendo su avance y el tipo de movimiento en su locomoción (Russo, y otros, 2013).

En la figura 2-21 se aprecia el robot Scout en configuración cuadrúpedo avanzando y acomodándose para pasar sobre un obstáculo.

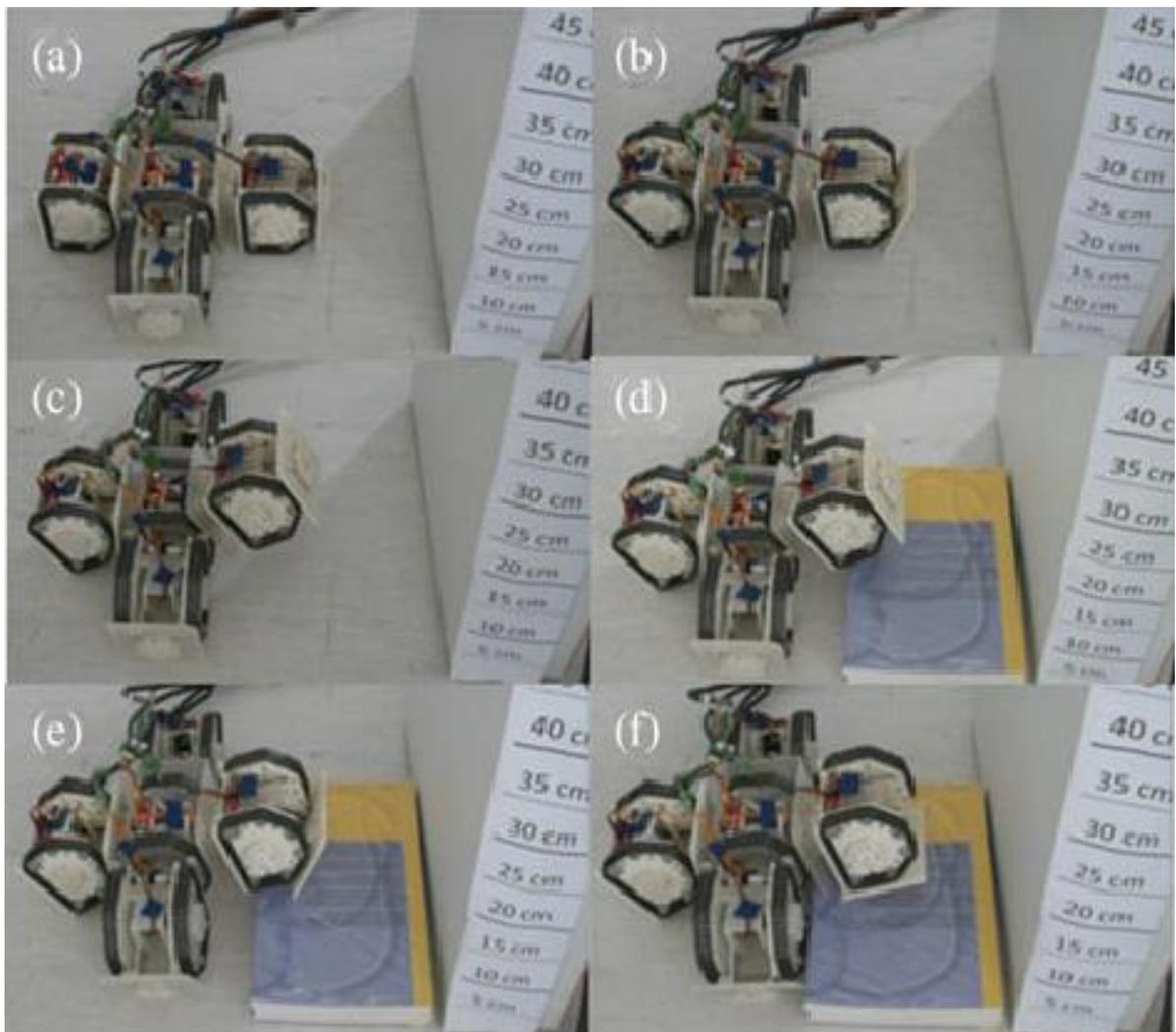


Figura 2-21. Marcha del cuadrúpedo con módulos Scout robot sorteando un obstáculo.

Tomado de: (Russo, y otros, 2013)

2.2.8 Módulo MECABOT

Es la primera versión de los módulos implementados en este trabajo, desarrollados en la Universidad Militar Nueva Granada en el año 2014 para ser acoplado con otros módulos MECABOT para formar configuraciones. Este módulo posee 5 servomotores, dos para mover los pivotes y 3 para las caras rotacionales, pero debido a su gran tamaño los motores no generaban el torque necesario para moverse con otros módulos acoplados, lo que descartaba la implementación de configuraciones de movimiento.

En la Figura 2-22 se muestra el módulo MECABOT que dio inicio al trabajo de robótica modular que actualmente implementa la cuarta generación de estos.



Figura 2-22. Módulo MECABOT

Tomado de: (Hurtado Herasco & Rubiano Montaña, 2013)

2.2.9 MECABOT 2.0

La segunda versión del módulo MECABOT, que al igual que su versión anterior posee 2 servomotores para mover el pivote y 3 para las caras rotacionales. El MECABOT 2.0 es igual a su predecesor en diseño y componentes, siendo su única diferencia la reducción del módulo con el fin de disminuir el peso de este y así los motores soporten el peso de otros módulos acoplados. A pesar de ser un modelo más liviano, éste no cuenta con la fuerza para mantener una configuración y moverla, por lo que se dio paso a un rediseño del módulo para su siguiente versión.

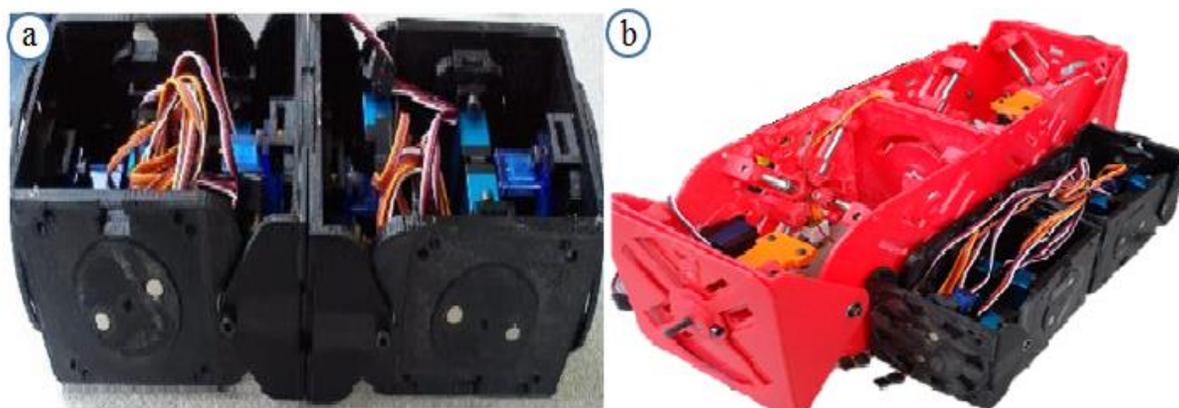


Figura 2-23. a) Módulo MECABOT 2.0, b) Comparación módulo MECABOT 2.0 frente a su predecesor módulo (rojo)

2.2.10 MECABOT 3.0

Después de los primeros modelos implementados del módulo MECABOT se realizó un rediseño considerando todas las falencias de estos, como su gran tamaño y la falta de fuerza en los motores. El módulo MECABOT 3.0 presenta un diseño compacto y más estilizado que consta de 6 servomotores para cada semimódulo, 2 para pivote y 4 para las caras rotacionales, además de usar una tarjeta Teensy 3.2 para su comando y sistema de comunicación mediante Xbee Pro Serie 1 que envía las instrucciones a los módulos. El rediseño de los módulos permitió implementar las configuraciones oruga y serpiente en las que se acoplaba una serie de módulos mediante dos imanes en caras como se aprecia en la Figura 2-24. La fuerza de esta conexión permitió generar movimientos ondulatorios típicos de las topologías 1D, pero para las configuraciones rueda y oruga es necesario que los módulos levantes más peso y su acople sea más fuerte.



Figura 2-24. Configuración oruga implementada con los módulos MECABOT 3.0

Tomado de: (Lancheros Guzmán & Sanabria Galvis, 2016)

2.2.11 Módulo MECABOT 4.0

Versión actual del módulo MECABOT implementado durante el desarrollo de este trabajo con el fin de ser usado en las configuraciones rueda y hexápodo. Este módulo es más pequeño que su versión anterior pero de mayor capacidad, al utilizar motores metálicos de 4 kg de torque en sus pivotes. Este presenta un diseño más compacto al eliminar los motores rotacionales de los lados, dejando ese espacio para los motores de los pivotes, este ajuste no afecta a los demás componentes electrónicos en el que se incluyó un regulador Step-Down para proteger al módulos de sobre picos de corriente, las características del MECABOT 4.0 se abordan en detalle en el capítulo 5. A diferencia de las versiones anteriores, los semimódulos que componen al MECABOT no son iguales, el semimódulo trasero se adaptó para llevar dos baterías de 7 voltios, 3 amperios para la alimentación de los componentes del módulo, brindándole una mayor autonomía al no necesitar una fuente de energía conectada por cables y sacándole provecho a la comunicación inalámbrica de los MECABOTS para control a distancia.



Figura 2-25. Módulo MECABOT 4.0

3 Modelo Matemático Configuración Hexápodo

Cuando se está trabajando en robótica móvil, una de las partes más importantes para determinar las capacidades móviles y los tipos de locomoción es el estudio de movimiento. En el caso del hexápodo esto permite conocer las trayectorias de las patas para generar determinado movimiento respecto a un eje de referencia, en el que se identifican las relaciones de posición y orientación dada por los actuadores que son los servomotores y el efector final. Para el estudio de movimientos se manejan dos métodos fundamentales de solución que son por cinemática directa e inversa.

- **Solución por cinemática directa:** Este método consiste en determinar la posición y orientación final del robot por medio de transformaciones homogéneas. Para esto se establece un sistema de coordenadas de referencia en el que se conocen las dimensiones y ángulo de cada articulación, en donde se busca la posición del punto final del robot (**Escobar Olarte & Silva Ortega, 2011**).
- **Solución por cinemática inversa:** Al contrario de la cinemática directa, este método busca los ángulos que debe tomar cada articulación para llevar el extremo del sistema a una posición y rotación específica. Existen varias formas de calcular la cinemática inversa, siendo el más simple el método geométrico debido a que la cinemática inversa solo es aplicable a sistemas de 3 grados de libertad como máximo (**Cárdenas Verdugo, 2011**).

Para realizar un modelo del hexápodo es necesario conocer sus puntos críticos, básicamente es un cuerpo sostenido por seis patas en el que se describe un polígono de estabilidad cuyo centro de masa se encuentra justamente en el centro del modelo y se mantiene bajo, brindándole de entrada una gran estabilidad. El cuerpo permanece estático, por lo que en realidad lo único que se mueven son las patas en las que se centra el estudio cinemático. Estas logran desplazarse mediante movimientos coordinados y periódicos utilizando el mismo movimiento en cada pata al ser exactamente iguales, lo único que cambia es la secuencia en la que se moverá cada pata para determinado movimiento y ya que utilizan el mismo programa, se simplifica el modelo matemático debido a que solo se necesita analizar una pata que se repite otra cinco veces en la configuración.

En la Figura 3-1 se presenta la configuración modular hexápodo en la que se ubica el centro de masa y se describe el triángulo de estabilidad que forma con tres de sus patas.

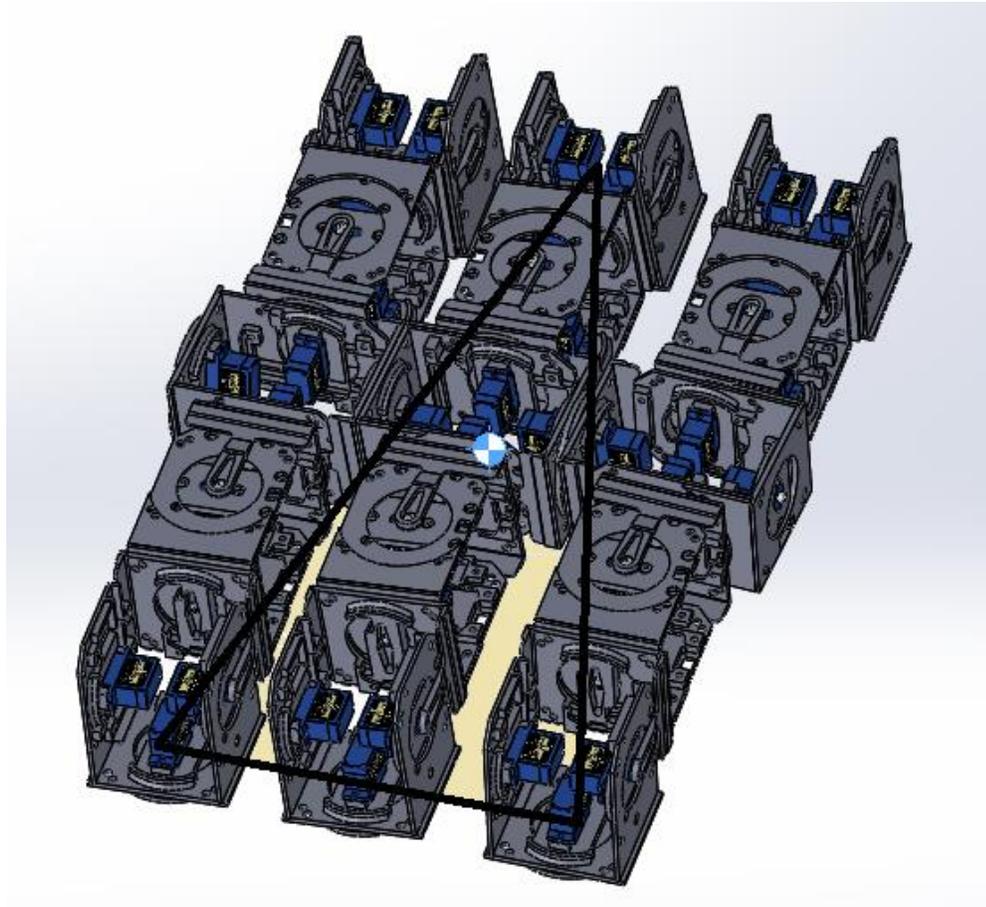


Figura 3-1. Polígono de estabilidad configuración Hexápodo MECABOT 4,0..

Pero antes de llegar al modelo final visto en la figura 3-1, se hizo un análisis de los tipos de patas posibles con el MECABOT4.0, debido a que son módulos con varias formas de conexión entre módulos y semimódulos. Principalmente están dadas si el módulo o semimódulo están conectados por la cara, que es la parte posterior del semimódulo la cual es estática y solo permite movimientos rotacionales dados por el motor sujeto a una rueda imantada y el pivote que se mueve de arriba abajo por medio de dos motores. Estas conexiones determinan los movimientos que puede realizar cada módulo en donde también se tiene en cuenta la orientación de los semimódulos, por lo tanto es necesario evaluar las capacidades que provee cada conexión y cuál es la mejor para ser

usada como pata del robot hexápodo y facilite el ejecutar movimientos omnidireccionales. Las siguientes son las conexiones pensadas para la pata del hexápodo:

- **Conexión Cara-Cara:** Esta forma conecta los dos semimódulos por sus caras rotacionales dejando sus pivotes como extremos. Para formar una pata esta conexión queda descartada debido a que al conectarlas solo permite girar los semimódulos y no logra flexionar las articulaciones de la pata, capacidad básica de los seres con extremidades por lo que un hexápodo con patas de este estilo se movería de forma muy rígida dejando solo a los pivotes para movimientos de locomoción que al estar en los extremos no le darían la altura ni el alcance necesarios para realizar una marcha. Este modelo se logra apreciar en la figura 3-2.

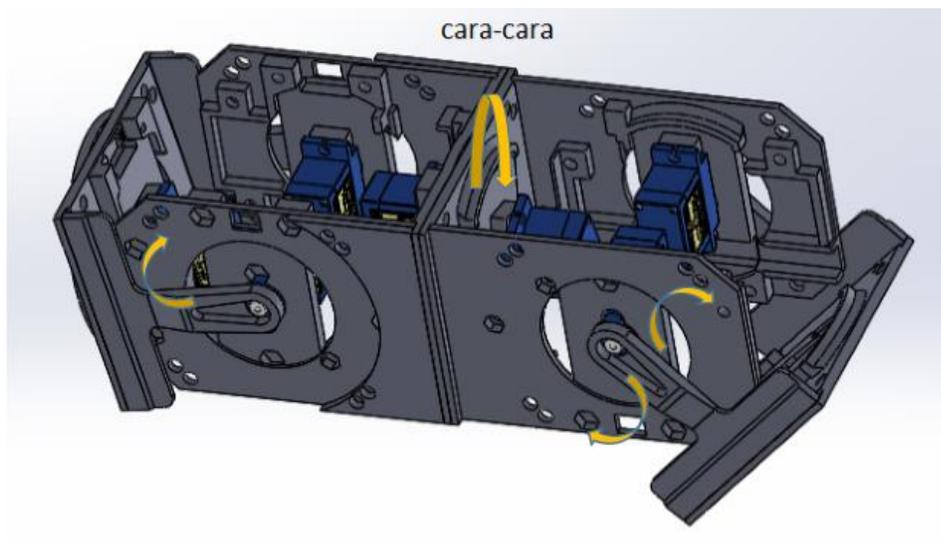


Figura 3-2. Módulo MECABOT 4.0 conexión de semimódulos cara-cara.

- **Conexión Pivote-Pivote:** Al acoplar los dos pivotes de los semimódulos como en la Figura 3-3, la pata resultante logra un mayor ángulo en la articulación que le podría dar mayor altura al hexápodo y mejoraría los movimientos laterales aunque resulta redundante pegar los dos pivotes debido a que se necesitan otros movimientos como mover la pata hacia adelante.

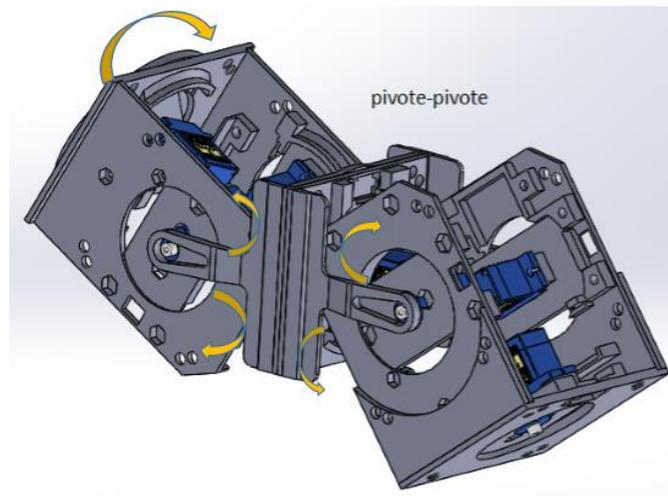


Figura 3-3. Módulo MECABOT 4.0 en conexión Pivote-Pivote

- **Conexión Cara-Pivote en el mismo sentido:** Esta forma de acople busca darle mayor movimiento dándole movimientos de rotación y flexión cerca al cuerpo y al final de la articulación, asemejándose más a los movimientos de la extremidad de un insecto. Aun así dependería de movimientos rotacionales para dirigir la pata hacia adelante, complicando la marcha hacia adelante o hacia atrás. El acople cara-pivote puede ser apreciado en la figura 3-4.

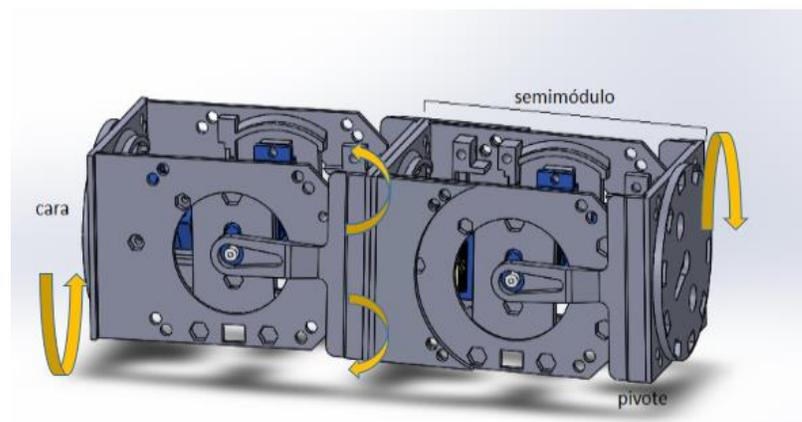


Figura 3-4. Módulo MECABOT 4.0 en conexión de semimódulos Cara-Pivote.

- **Conexión Cara-Pivote cambiando sentido:** Los semimódulos cuentan con caras rotacionales en la parte delantera y posterior lo que permite cambiar el sentido del semimódulo cuando están acoplados en módulos, esta característica de diseño aumenta las configuraciones posibles que se pueden crear con los módulos MECABOT y le da más opciones de movimiento al realizar tareas específicas. Al ser el hexápodo un modelo bioinspirado se tiene cuenta que las patas de los insectos se mueven hacia delante y atrás con la articulación pegada al cuerpo, y la articulación final sube y baja para compensar las irregularidades del terreno. Teniendo en cuenta estas capacidades se pensó en una variación de la conexión cara-pivote, dejando en sentidos opuestos cada semimódulo en donde el principal cambio es la orientación de los pivotes. El cambio de orientación imita el movimiento de marcha de los hexápodos en general en su forma más básica y utilizando el pivote acoplado al cuerpo del robot de forma horizontal para darle movimientos de avance y retroceso, junto con el pivote del semimódulo del extremo para subir y bajar la pata, se logra imitar de manera simplificada la marcha de un hexápodo, lo que brinda una idea de la secuencia de programación. En la Figura 3-5 se presenta este tipo de acople y se agregan flechas que indican los movimientos de las articulaciones.

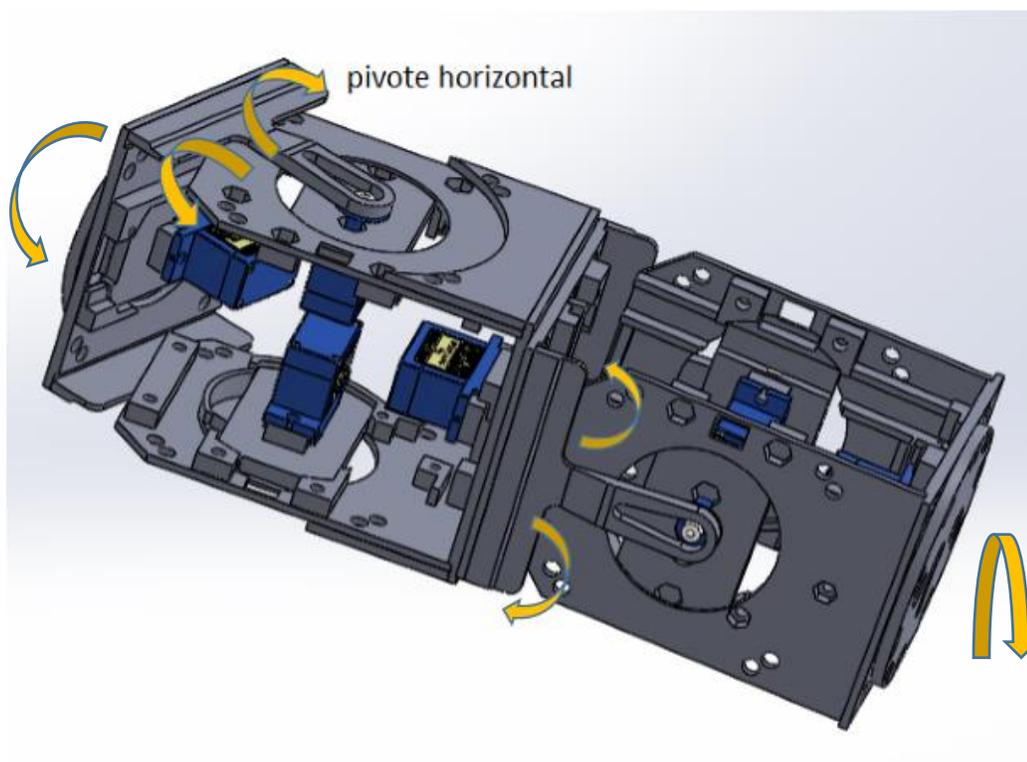


Figura 3-5. MECABOT 4.0 acople Cara-pivote cambio de sentido en los semimódulos

3.1 Cinemática Directa de una pata del hexápodo MECABOT

Una vez identificados los posibles acoples para la pata y determinar que la conexión carapivote en la que los semimódulos se acoplan en sentidos opuestos (Figura 3-5), era la que mejor se adaptaba a las formas de locomoción pensadas para la configuración hexápodo, es necesario realizar un modelo matemático que permita identificar los ángulos y posiciones que logra el sistema. La cinemática directa permite determinar la forma de la pata en cada posición lo que ayuda a generar la secuencia de pasos y con ayuda del software Matlab, facilitar estos cálculos y visualizar gráficamente esos movimientos.

Para realizar la cinemática directa, es necesario conocer la matriz de transformación homogénea que representa la posición y orientación de un sistema de coordenadas a otro visto en los enlaces. Los vectores dentro de la matriz de translación incluyen las submatrices de rotación, translación, escalado y perspectiva.

Donde la matriz T de la Figura 3-6 de translación queda de la siguiente forma, en donde la submatriz de perspectiva suele ser nula al no presentar ninguna transformación y la de escalado corresponde a una unidad al trabajar con las dimensiones del robot.

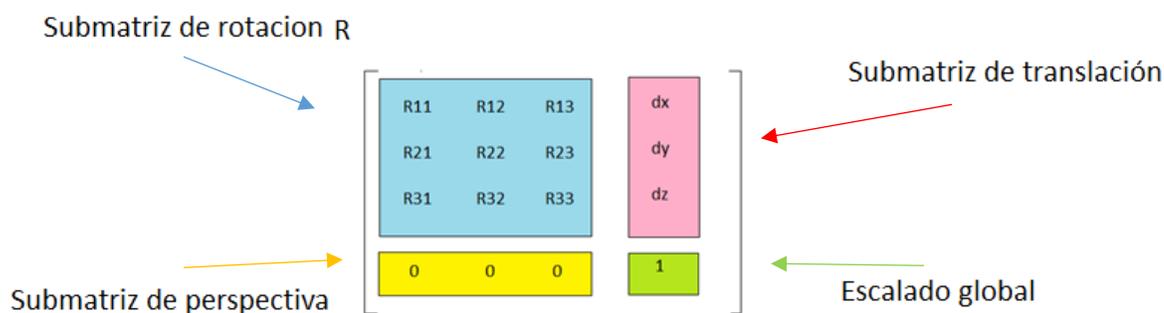


Figura 3-6. Matriz de transformación homogénea

La matriz T esta en parámetro de las articulaciones del robot y sus grados de libertad, por lo que se halla una matriz que describe las relaciones de posición de un enlace con respecto al eslabón que le sigue. De esta forma al realizar el producto de las matrices resultantes, se logra representar la cadena cinemática del robot estudiado hasta su posición y orientación final.

Donde A0 vendría siendo la matriz de transformación del primer escalón y cuando se tienen todos los eslabones se forma la matriz T, que abarca todas las transformaciones de los eslabones con respecto al anterior para que el extremo del robot llegue a determinada posición.

$$T = A0 * A1 * A2 * ... * An$$

Para llegar a la matriz T se debe escoger adecuadamente el sistema de coordenadas para cada eslabón por lo que varias soluciones, entre los que se destacan el método geométrico y Denavid-Hartenberg, siendo este último el que se utilizará para realizar la cinemática directa. Este método fue propuesto en 1995 por Denavid-Hartenberg y establece una solución matricial en la que se usa un sistema de coordenadas para cada articulación de un grado de libertad, hasta completar todos los eslabones del robot (Escobar Olarte & Silva Ortega, 2011). En cada transformación se tienen en cuenta las rotaciones y translaciones que relacionan el sistema de referencia del eslabón-n con la siguiente n+1, en donde se tienen los siguientes parámetros:

- Rotación del eje Z_{n-1} en un ángulo θ_n .
- Translación eje Z_{n-1} en una distancia d_n en un vector d_n (0, 0, d_n).
- Translación a lo largo de X_n a una distancia a_n en un vector a_n (0, 0, a_n).
- Rotación del eje X_{n-1} en un ángulo α_n

Estos datos se ubican en la Tabla 3-1 D-H

Articulación	θ	d	a	α
1	θ_1	d_1	a_1	α_1
2	θ_2	d_2	a_2	α_2
..	$\theta_{..}$	$d_{..}$	$a_{..}$	$\alpha_{..}$
n	θ_n	d_n	a_n	α_n

Tabla 3-1. Parámetros D-H para un robot de n articulaciones.

Cada articulación desarrolla la expresión:

$$A_{i-1}^i = T(z, \theta_i) * T(0,0, d_i) * T(a_i, 0,0) * T(x, \alpha_i)$$

Ec. 3-1

$$A_{i-1}^i = \begin{bmatrix} \cos\theta_i & -\text{sen}\theta_i & 0 & 0 \\ \text{sen}\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & \alpha_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\text{sen}\alpha_i & 0 \\ 0 & \text{sen}\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ec. 3-2

Donde T es el resultado de multiplicar estas matrices $T=A_0*A_1*A_2*...A_n$ y la cinemática directa es el producto de la matriz T con el punto del extremo del robot. $P=T*P_f$.

3.1.1 Cinemática directa de la conexión cara-pivote cambiando orientación

La Figura 3-7 presenta la conexión cara-pivote cambiando el sentido de orientación de los semimódulos, quedando perpendiculares entre sí.

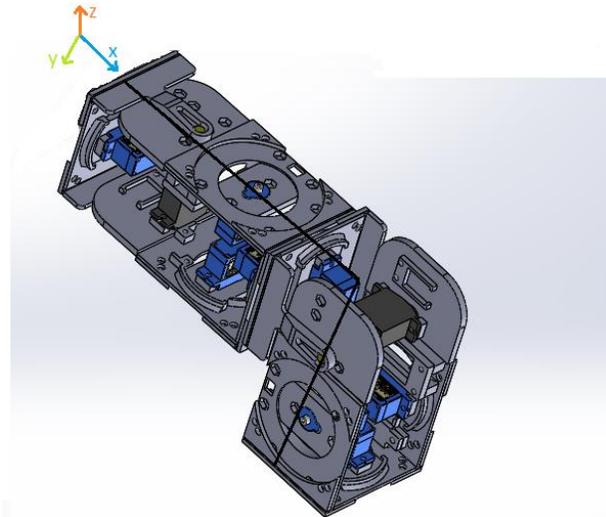


Figura 3-7. Módulo MECABOT 4.0 en conexión cara-pivote cambiando el sentido

En la Figura 3-8 se presenta el modelo simplificado del módulo acoplado de forma cara-pivote cambiando la orientación de los semimódulos

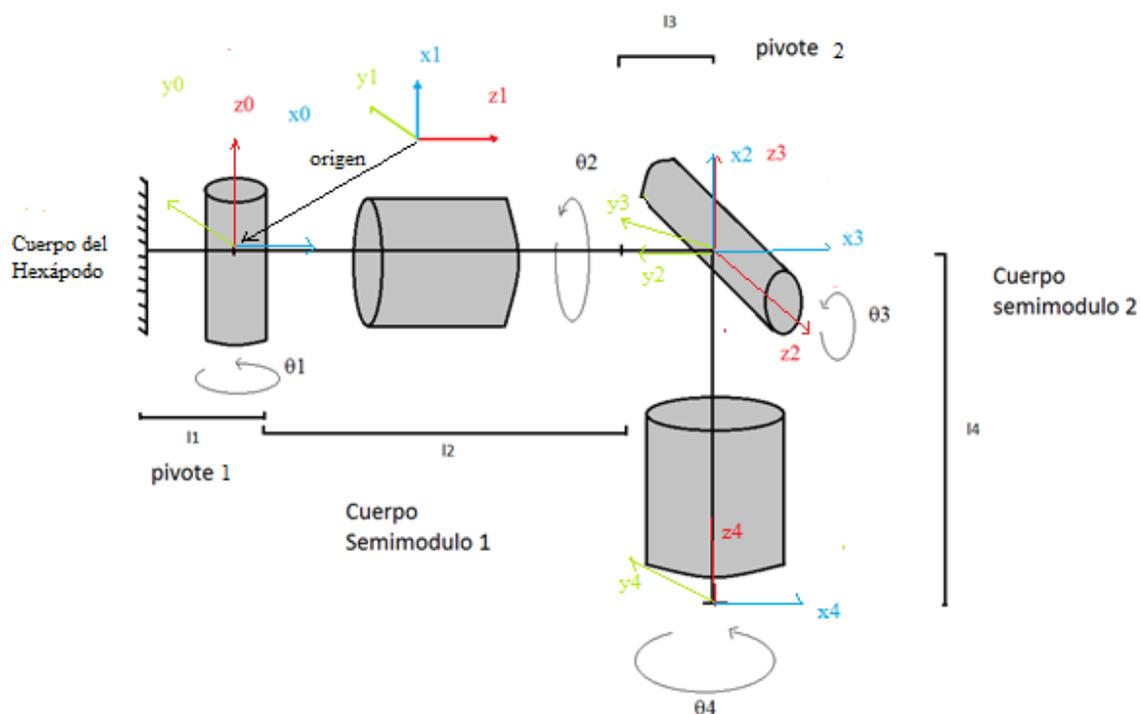


Figura 3-8. Modelo simplificado MECABOT 4.0 conexión cara-pivote cambiando sentido

La Tabla 3-2 se presenta la matriz D-H a partir del modelo simplificado del acople cara-pivote cambiando el sentido.

Pivote horizontal - cara	θ	d	a	α
1	θ_1	l_1	0	$\pi/2$
2	θ_2	l_2	0	$-\pi/2$
3	θ_3	l_3	0	$\pi/2$
4	θ_4	l_4	0	0

Tabla 3-2. Tabla Denavid-Hartenberg de la conexión de modulo cara-pivote

Utilizando matlab para facilitar el cálculo de las matrices de transformación de las cuatro articulaciones que forman el módulo y la pata del hexápodo. En la figura 3-10 se muestra la transformada de la primera articulación en donde q_1, q_2, q_3, q_4 son los ángulos $\theta_1, \theta_2, \theta_3, \theta_4$, respectivamente y α son los valores α vistos en la Tabla 3-2.

$$A_{01} = \begin{bmatrix} \cos(q_1) & -\sin(q_1) * \cos(\alpha_1) & \sin(q_1) * \sin(\alpha_1) & a_1 * \cos(q_1) \\ \sin(q_1) & \cos(q_1) * \cos(\alpha_1) & -\cos(q_1) * \sin(\alpha_1) & a_1 * \sin(q_1) \\ 0 & \sin(\alpha) & \cos(\alpha) & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ec. 3-3

Donde T es igual a:

$$T = A_{01} * A_{12} * A_{23} * A_{34}$$

Ec. 3-4

En la cinemática directa se conocen los ángulos para hallar la posición de cualquier punto, por lo que se utiliza el programa de matlab para observar el punto en el espacio al que llegaría el extremo de la pata con determinados movimientos, representado las articulaciones de la pata por líneas en la Figura 3-10, en la que se toma como ejemplo los ángulos de articulación $\theta_1=10, \theta_2=0, \theta_3=-90$ y $\theta_4=0$. La matriz de transformación resultante (Figura 3-9) es igual a:

$$T = \begin{bmatrix} -0.1104 & -0.0000 & 0.9939 & 33.2902 \\ 0.9939 & -0.0000 & 0.1104 & 70.2599 \\ 0 & 1.0000 & 0.0000 & 26.0000 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix}$$

Figura 3-9. Matriz de transformación de la pata en acople cara-pivote

En la Figura 3-10 se aplican las transformaciones en las que se puede determinar el vector de posición (x, y, z) de cada sistema de coordenadas, estos puntos se usan en la función plot de matlab para representar los movimientos de la pata en base al módulo MECABOT 4.0.

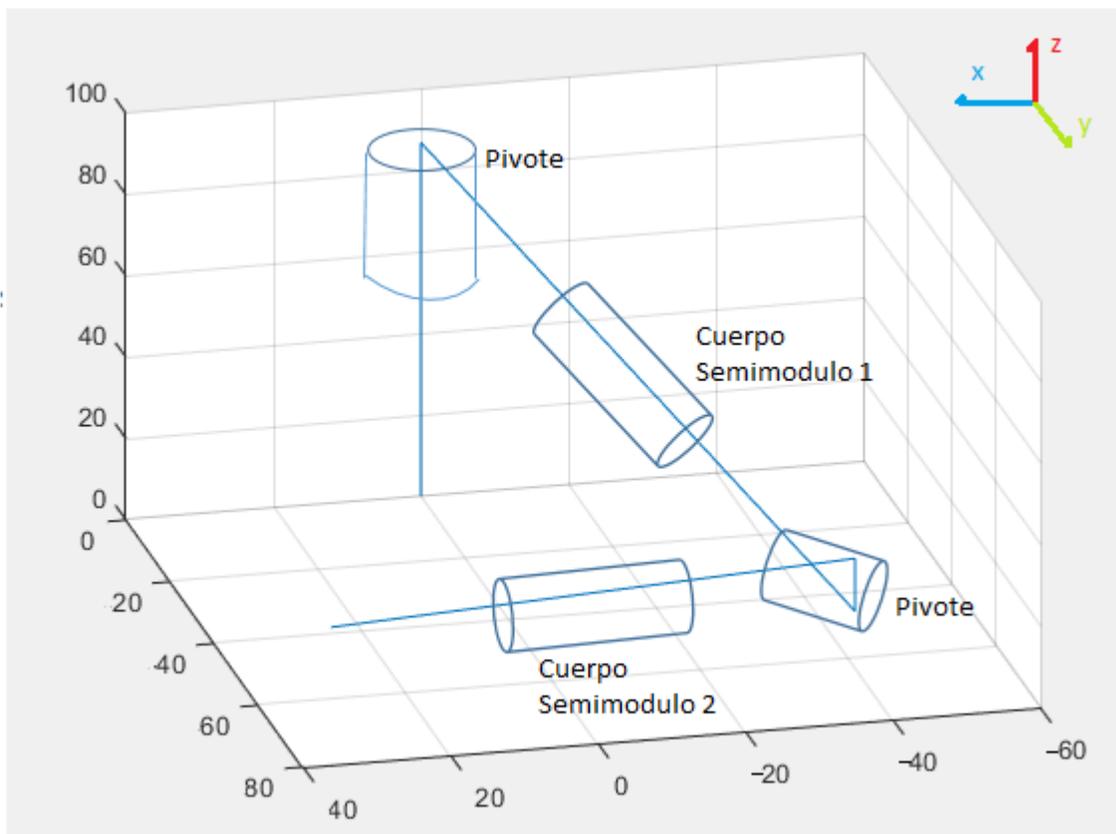


Figura 3-10. Grafica de matlab. Pata hexápodo en conexión cara-pivote cambiando el sentido

La configuración cara-pivote cambiando el sentido de los semimódulos a diferencia de las conexiones anteriores presenta variación en los tres ejes de posición, en este caso se eligió mover el pivote sujeto al cuerpo hacia adelante y el pivote del extremo a un ángulo de -90° , lo que deja la pata en una posición de estabilidad para así poder dar un paso hacia adelante. Este avance se dio por la inclinación del semimódulo sujeto al cuerpo que se obtiene de mover el pivote, que para el ejemplo fue de 10° , el cual es un ángulo cómodo para que el módulo se mueva sin chocarse con otra pata. Esta forma de acople permite ir hacia adelante o hacia atrás sin la necesidad de girar el módulo, de hecho no se utilizan las caras rotacionales ya que todo el movimiento lo hacen los pivotes, uno para movimientos horizontales y el otro verticales permitiendo levantar la pata

mientras avanza asemejándose a la marcha de los insectos. En comparación con las otras formas de módulo, esta plantea una solución más simple y como se aprecia en los puntos, puede cambiar de dirección en todo sentido lo que permitiría realizar movimientos omnidireccionales. Esto simplifica la programación al solo utilizar los pivotes de cada semimódulo para mover el hexápodo, disminuyendo el número de pasos por movimiento y al usar menos actuadores, menor será el consumo y la batería del módulo durará más tiempo, haciendo más practica la locomoción de robot hexápodo construido a base de módulos MECABOT 4.0.

3.2 Cinemática inversa de una pata del hexápodo MECABOT

El objetivo de la cinemática inversa contrario a la directa es hallar los valores de las coordenadas articulares, es decir los ángulos θ_n conociendo la posición y orientación del extremo del robot. Existen varios métodos para dar solución por cinemática inversa dado que se soluciona un problema de máximo 3 grados de libertad y para este caso se utilizará uno de los métodos más simples que es el método geométrico. Este método parte de conocer la posición final del robot (p_x , p_y , p_z) y mediante relaciones trigonométricas y geométricas se solucionan los triángulos formados por las articulaciones del robot (Escobar Olarte & Silva Ortega, 2011). En conclusión la solución general de la cinemática inversa es encontrar una función que halle los ángulos a partir del punto final.

$$\theta_n = fn(px, py, pz, \alpha, \beta, \gamma)$$

Ec. 3-5

En donde θ_n son los ángulos de la articulación n que indica los grados de libertad, $p(x, y, z)$ es la coordenada del punto final y α, β, γ son los ángulos de orientación del sistema.

De análisis cinemático anterior, se determinó que la conexión entre semimódulos carapivote cambiando el sentido de orientación, es la solución más práctica para ser usada como patas del robot hexápodo modular MECABOT, por lo que solo se analizará la cinemática inversa de esta configuración.

3.2.1 Cinemática inversa del módulo con acople cara-pivote cambiando el sentido

Por lo visto en el análisis de la cinemática directa, esta conexión (Figura 3-11), solo necesita de los pivotes para moverse por lo que no se incluyen los rotacionales simplificando el modelo.

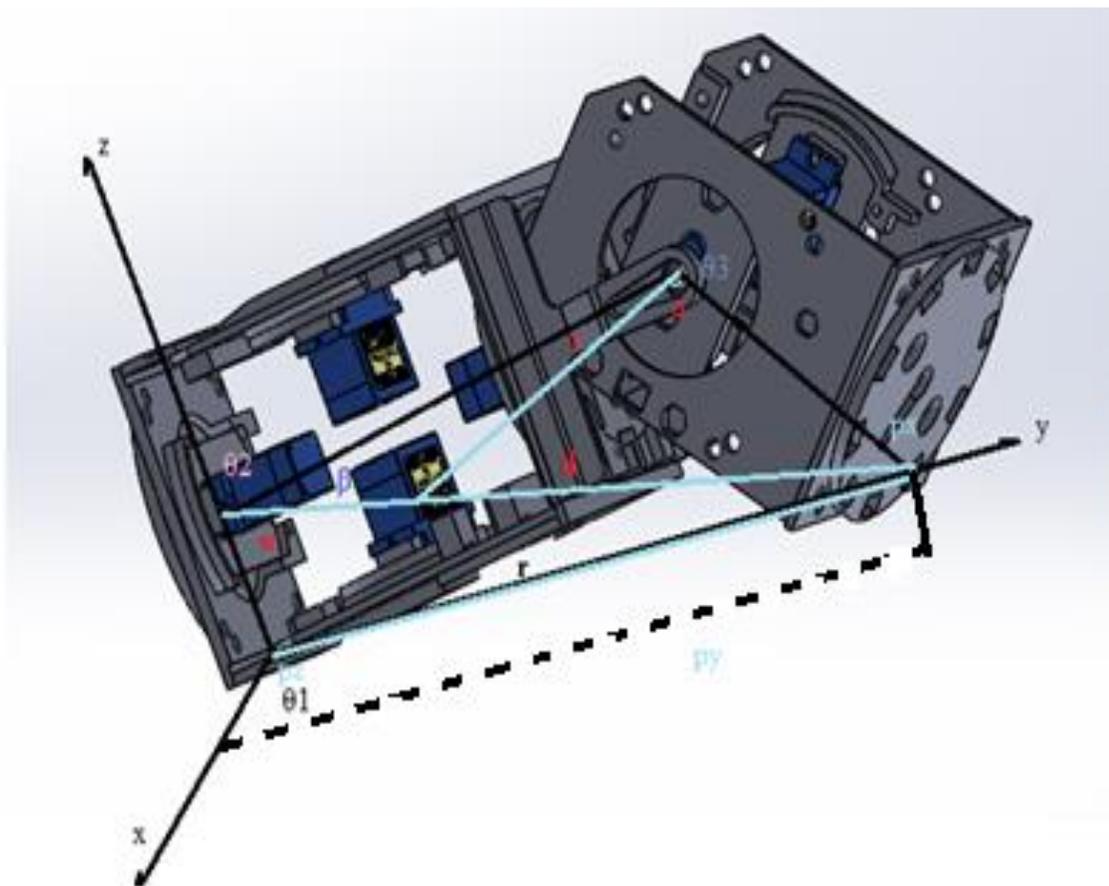


Figura 3-11. Parámetros cinemática inversa MECABOT 4.0

Realizando dos planos de las vistas del modelo simplificado, se logra reconocer mejor los parámetros del sistema y realizar las relaciones trigonométricas. Para la primera parte se toma la vista en el plano x,y para calcular θ_2 y θ_3 , y para hallar θ_1 se utiliza la vista del plano z, y.

Figura 3-12 Vista lateral del modelo simplificado de la pata del robot hexápodo.

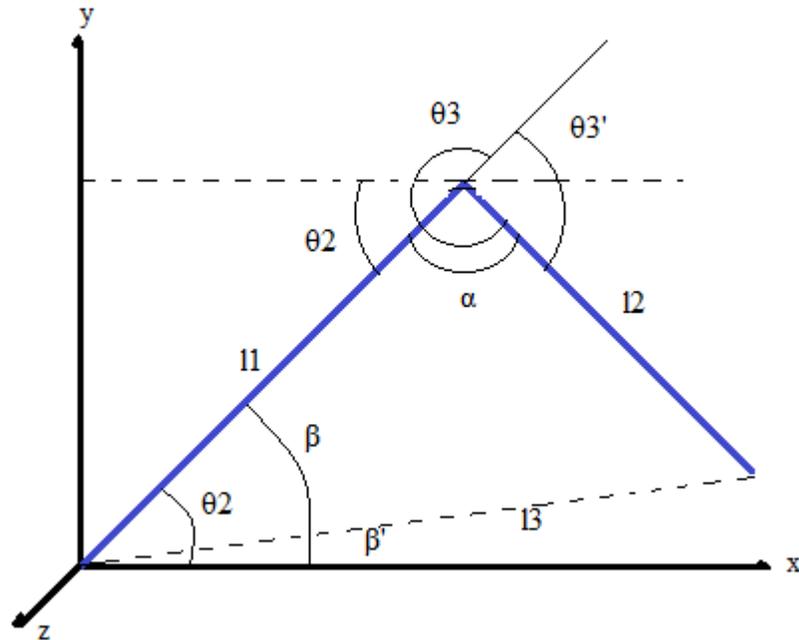


Figura 3-12. Cinemática inversa de la pata del hexápodo plano x, y
Utilizando las relaciones trigonométricas y la ley de cosenos se encuentra:

$$l3 = \sqrt{(x^2 + y^2 + z^2)}$$

Ec. 3-6

$$\beta' = \tan^{-1}\left(\frac{y^2}{\sqrt{(x^2 + z^2)}}\right)$$

Ec. 3-7

$$\alpha = \cos^{-1}\left(\frac{l1^2 + l2^2 - l3^2}{2 * l1 * l2}\right)$$

Ec. 3-8

$$\beta = \cos^{-1}\left(\frac{l1^2 + l3^2 - l2^2}{2 * l1 * l3}\right)$$

Ec. 3-9

En la Figura 3-13 se muestra una vista de la pata modular en la que se aprecia el ángulo θ_1 que permite mover hacia adelante o hacia atrás la pata del hexápodo.

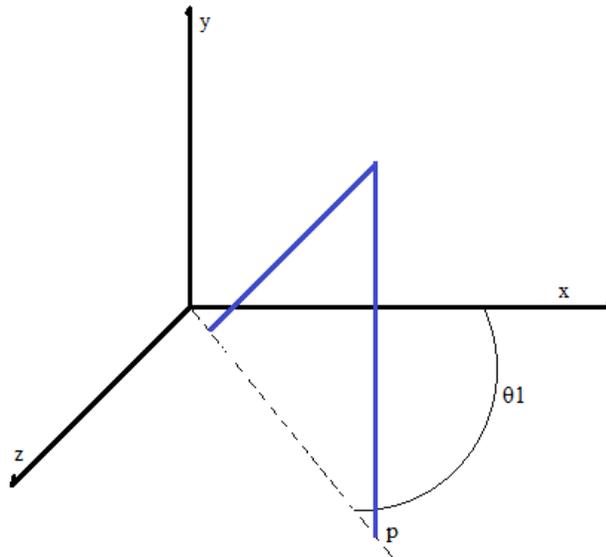


Figura 3-13. Cinemática inversa de la pata sobre el plano y, z.

Donde $P(x-dx, y+dy, z-dz)$ donde dx, dy, dz son la diferencia entre el punto de origen y la posición final.

$$\theta_1 = \tan^{-1}\left(\frac{z - dz}{x - dx}\right)$$

Ec. 3-10

$$\theta_2 = \beta + \beta'$$

Ec. 3-11

$$\theta_3 = 180 + \alpha$$

Ec. 3-12

$$\theta_3' = 180 - \alpha$$

Ec. 3-13

De esta forma se hallan los ángulos de cada articulación y por ende la forma que toma la pata del robot hexápodo para llegar a un punto específico.

4. Simulación del Sistema Robótico Modular MECABOT 4.0 en Configuración Hexápodo

En este capítulo se presenta la simulación de los módulos MECABOT 4.0, sucesor de los módulos MECABOT 3.0 desarrollados en la UNIVERSIDAD MILITAR NUEVA GRANADA en configuración hexápodo implementado las formas de locomoción a partir de módulos. Para poder simular las propiedades físicas y el comportamiento del hexápodo ante los programas de movimiento, se utilizó el software de simulación Webots desarrollado por Cyberbotics© de cual el grupo de investigación Davinci de la U.M.NG cuenta con las licencias de la versión Webots PRO 7.4.3. Este software trabaja importando las piezas del robot, ubicándolas en el nodo World (mundo) que será el entorno virtual donde se probará el robot, estas piezas han sido diseñadas a semejanza de los módulos físicos en el software de simulación CAD (diseño asistido por computador) SolidWorks, que permite elaborar cada pieza a partir de sus medidas reales y ensamblarlas para formar el modulo y a su vez la configuración hexápodo creada con estos módulos.

Una vez importado el robot se asignan las propiedades de los actuadores que posee, se procede a realizar el programa de comandos utilizando el compilador del propio Webots que permite programar en 6 lenguajes de programación, de los cuales se usara el lenguaje de programación en C para crear el programa. Este código ejecutará una secuencia de las marchas programadas para el hexápodo teniendo en cuenta las características de los motores en cada articulación y el algoritmo diseñado para esta. Considerando los movimientos posibles del análisis cinemático se implementaron 8 tipos de movimientos para el hexápodo, los cuales son: marcha hacia adelante, atrás, izquierda y derecha, movimientos rotatorios para cambiar de dirección derecha e izquierda, y llevar al robot a su posición de estabilidad sobre la que inicia las demás marchas y una de reposo en donde extiende sus patas sobre el piso. El avance de las marchas se evaluó utilizando el tiempo de simulación y cambiando el piso por una cuadrícula de cuadros de 5x5 cm, ubicando un cubo de 10cm como guía para visualizar mejor el avance. Con esto podrá implementar el sistema físico teniendo en cuenta los movimientos y velocidades adecuadas para un buen funcionamiento.

4.1 Simulador Webots

Webots es un software de simulación profesional que permite programar y simular en un entorno 3D robot móviles, teniendo en cuenta propiedades físicas que se pueden establecer como la gravedad, la masa del objeto, textura, fricción entre otras, que buscan darle realismo a los movimientos y condiciones del sistema. Dentro de sus funciones Webots importa figuras, piezas o robots completos en formato VRML a los cuales se les puede aginar todo tipo de sensores y actuadores como lo son cámaras, sensores infrarrojos, servomotores entre otros. Webots permite programar en los lenguajes C, C++, Java, Python y Matlab, cuyos controladores se pueden trasladar a robots reales, teniendo de antemano el comportamiento deseado visto en el entorno de simulación. Para el caso del MECABOT 4.0 configuración hexápodo se programó en lenguaje C, que es el lenguaje más usado entre los ejemplos que provee el mismo Webots y que pueden ser vistos la primera vez que se corre el programa, en la Figura 4-1 se muestra la ventana de inicio de Webots.



Figura 4-1. Ventana de iniciación de Webots

Tomado de: (2014 Cyberbotics Ltd., 2014)

Webots trabaja con una jerarquía de nodos para el mundo virtual y la programación típicos de los mundos virtuales VRML, además de otros propios para ir dándole propiedades y capacidades para crear sistemas más complejos entre los que están:

- **Nodo Worlinfo:** En este se establece las propiedades del ambiente 3D junto con la información de este, como ¿quién lo creó? y la fecha en que se hizo, además de definir propiedades físicas como la gravedad en los ejes de coordenadas, el paso del tiempo y la velocidad de simulación.
- **Nodo Background:** Para establecer el color de fondo en el subnodo skyColor.
- **Nodo Viewpoint:** Permite orientar la cámara que interactúa con el mundo permitiendo rotar, hacer zoom y trasladar la cámara en sus respectivos subnodos.
- **Nodo Pointlight:** Dentro de sus subnodos permite cambiar la luminosidad, el color, intensidad, la posición y número de fuentes de luz, así como manipular las sombras generadas.
- **Nodo RectangleArena:** Este nodo permite cambiar las propiedades de la superficie donde actuará al robot, el tipo de piso, las texturas y materiales, así como la orientación de este. Por defecto el mundo de Webots trae un piso cuadrículado al que se le puede modificar el tamaño de cuadro y el espacio que ocupa en el mundo.
- **Nodo World:** Son nodos a los que se le asigna una figura o robot y un subnodo de transformación en el que se le asigna la apariencia y orientación del robot. Este nodo se utiliza como base para creación del robot ya que en este se define el tipo pieza, asignándole propiedades mediante nodos más específicos como los son: el **nodo Robot**, generalmente asignado a todo un ensamble importado, al que se le pueden asignar todas las propiedades de un robot, como lo son el controlador que usa, si posee batería y cómo sería el consumo, además de los subnodos de las articulaciones y especificar las colisiones entre estas. También se utilizan nodos más específicos para sensores y otros componentes, por ejemplo para usar servomotores se llama el **Nodo Servo** en el que se especifica la velocidad, torque, posición y de más características de los motores. El **Nodo Solid** hace referencia a un objeto y sus propiedades físicas como dimensiones, masa y material. Y también está el **Nodo Shape** para la apariencia y geometría para crear objetos.

En la siguiente Figura 4-2 de Jinhua Weng, se parecía mejor la jerarquía del nodo World.

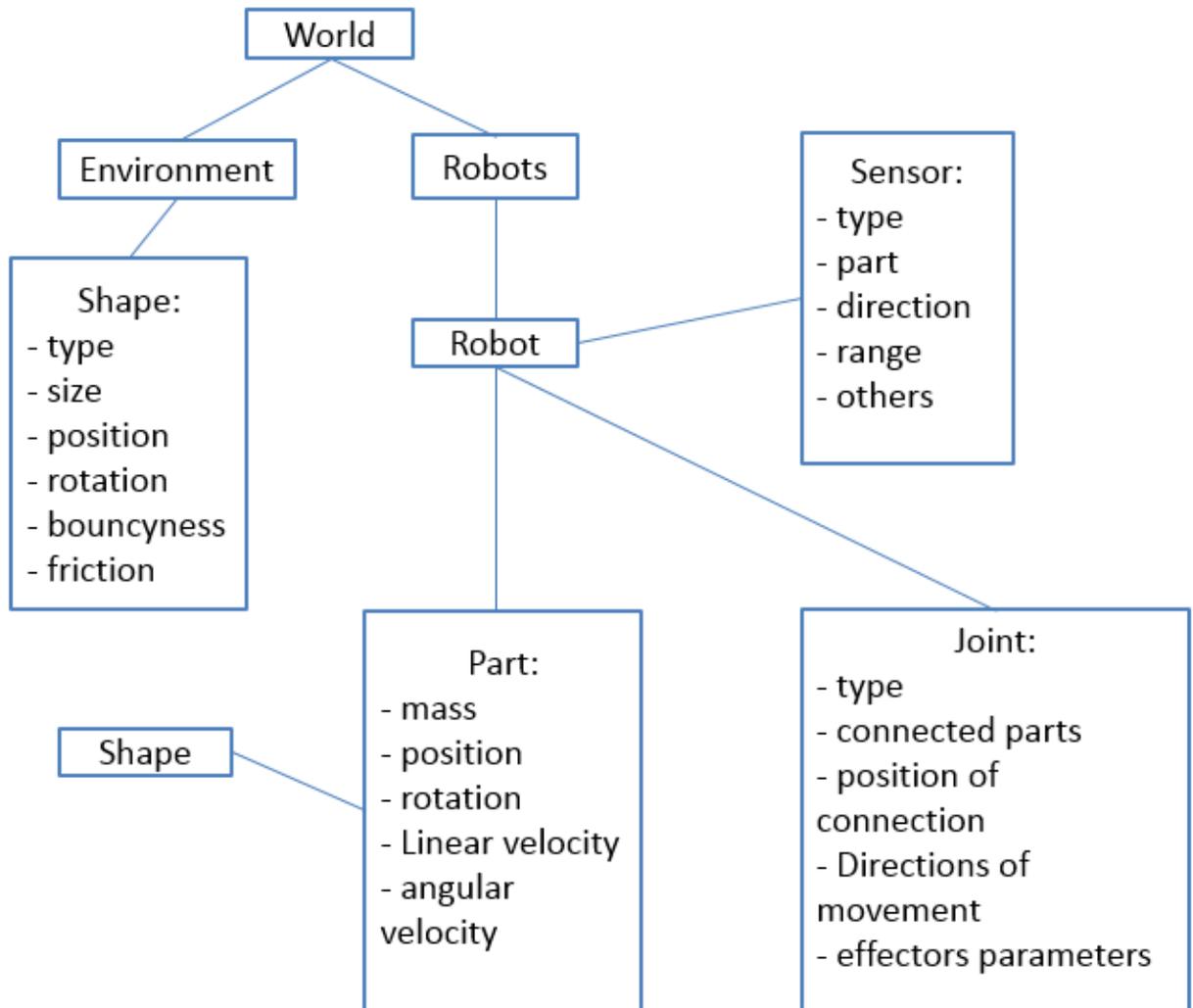


Figura 4-2. Jerarquía principal del nodo World.

Tomado de: (Weng, Creación del Modelo del Robot Humanoide Teo para el Simulador WEBOTS, 2012)

4.2 Importación a Webots del módulo MECABOT 4.0

Los módulos MECABOT, conforman el sistema robótico modular que se ha venido trabajando dentro del grupo de investigación Davinci en la Universidad Militar Nueva Granada y actualmente está en su versión 4 después de varios rediseños, realizados por miembros del grupo de investigación manteniendo el concepto original de modulo auto configurable. Inicialmente se realizó el modelo CAD de los módulos MECABOT 3.0 para ser usados en las configuraciones rueda y hexápodo por lo que se hicieron algunos prototipos del robot hexápodo con estos módulos, pero tomando en cuenta el trabajo anterior y que las nuevas configuraciones requerían otras necesidades, se decidió implementar la configuración con los nuevos módulos, más pequeño y robusto llamados MECABOT 4.0, donde las diferencias entre los prototipos así como los componentes de los módulos serán detalladas en el capítulo 5.

Para la simulación del sistema robótico modular se modelaron las piezas que componen el semimódulo Figura 4-3, así como su ensamble en el software de diseño SolidWorks que permite recrear las dimensiones de cada objeto a detalle para luego mediante relaciones de posición construir el modelo del módulo en 3D.

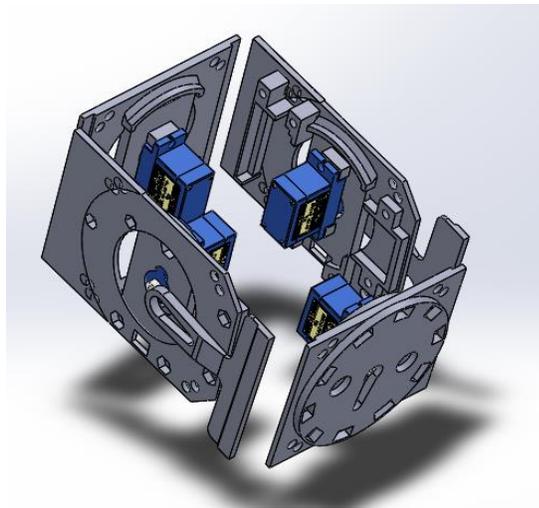


Figura 4-3. Semimódulo MECABOT 4.0 vista explosionada

Entre más complejo el objeto robot, el tiempo de simulación se incrementa, por lo que para su importación a Webots se usa un modelo simplificado del semimódulo, con la características más importantes en este caso solo se necesita la carcasa y el pivote para representar los semimódulos. Este modelo simplificado es exportado de SolidWorks como formato VRML (.wrl) cuyas siglas hacen referencia a lenguaje para modelado de realidad virtual a Webots de la siguiente manera, Figura 4-4:

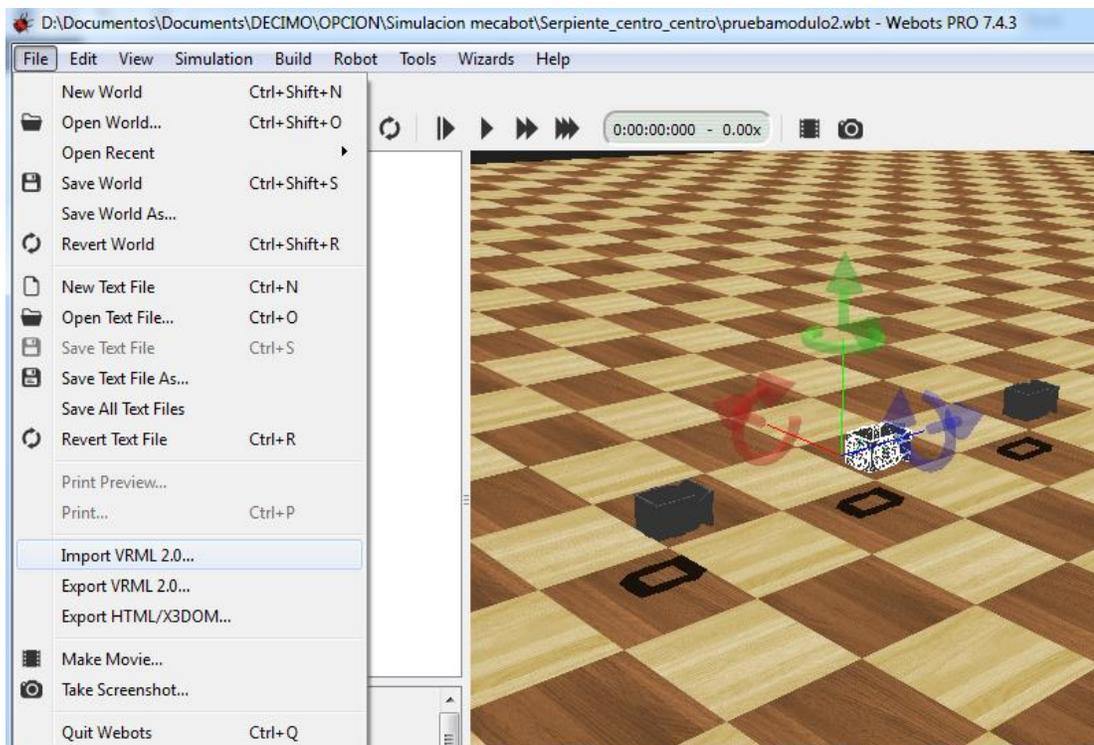


Figura 4-4. Módulo simplificado importado a Webots

Una vez importado se genera un nodo de transformación con la figura importada, al cual se le asigna un nodo *World* para darle las propiedades a la pieza, en este caso al haber importado todas las partes y ensamblar mediante relaciones de posición, se toma una referencia y se le asigna un nodo *Robot* para el semimódulo, que a su vez posee nodos de transformación *children* para las piezas que lo componen siguiendo la jerarquía en árbol hasta darle todas las propiedades que se necesitan, como es el caso de los pivotes a los que se le asigna un nodo motor para poder moverlos.

En la Figura 4-5 se muestra la ventana de nodos de Webots, detallando el árbol de nodos del módulo 3.

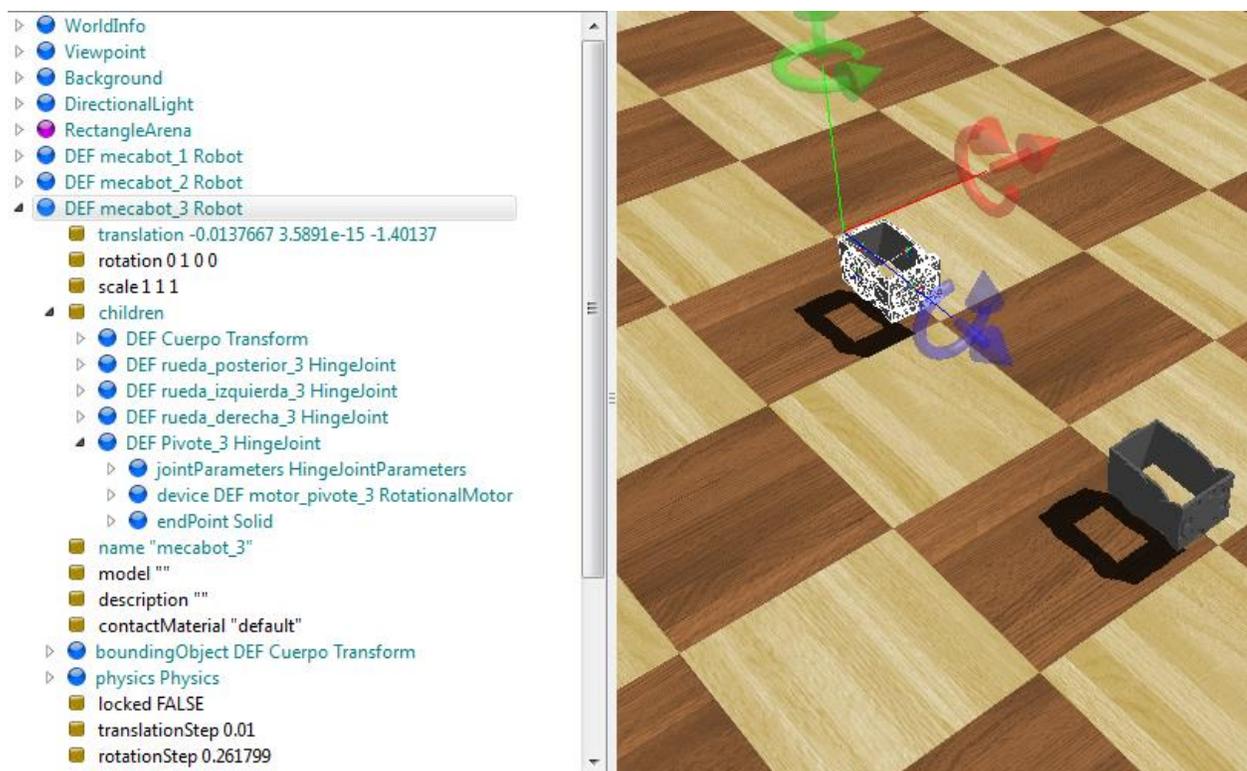


Figura 4-5. Jerarquía de nodos para los módulos MECABOT en Webots

Una vez establecido el semimódulo y todas sus propiedades, solo se necesita copiar el nodo robot para generar nuevos semimódulos idénticos, sin necesidad de repetir todos los pasos cambiándoles el nombre para así llamarlos dentro de la programación y mediante relaciones de posición construir la configuración hexápodo modular MECABOT 4.0. Cabe aclarar que cada nodo robot puede tener su propio controlador por lo que se pueden tener varios robots haciendo tareas distintas dentro del mismo mundo.

4.3 Diseño de robot modular Hexápodo en Webots

Los semimódulos de forma independiente no logran mayores desplazamientos, pero al estar unido a otro semejante, forman un módulo que permite ampliar esos movimientos y al unirlos con otros se forman las configuraciones. Para este caso se diseñará un robot hexápodo, es decir que posea seis patas que le permitan desplazarse, por lo que los módulos se pueden dividir en los usados en las patas y los módulos del cuerpo que permanecen estáticos. Teniendo en cuenta lo anterior, se realizaron varios diseños considerando el número de módulos utilizados y que el diseño previsto permita desplazamientos. Los siguientes fueron los prototipos de hexápodo desarrollados durante el trabajo:

- **Hexápodos con patas frontales y laterales:**

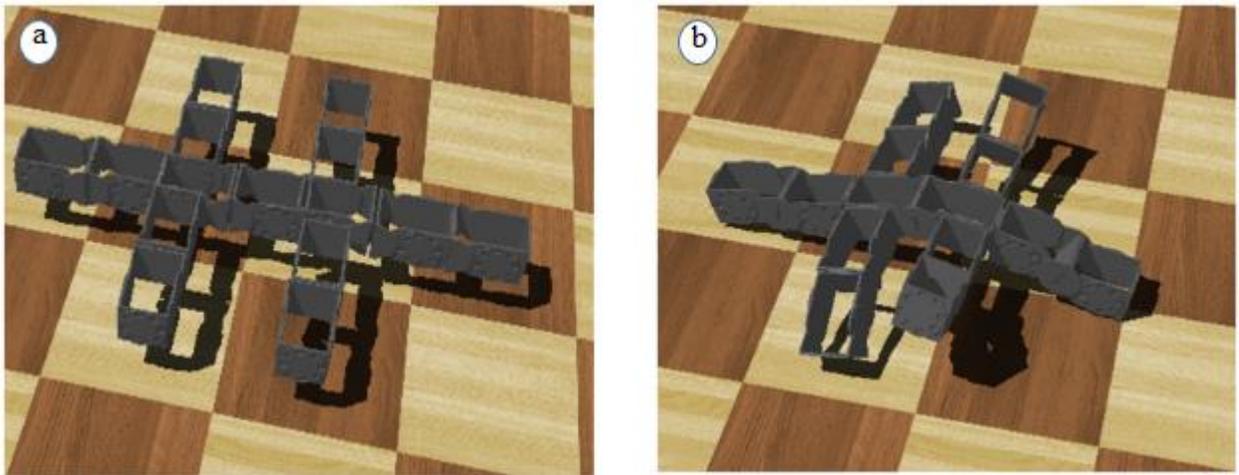


Figura 4-6. Robots modulares hexápodos con patas frontales y laterales.

Este primer concepto visto en la Figura 4-6 se desarrolló con los módulos MECABOT 3.0 en donde se planteó hacer los movimientos hacia adelante y laterales colocando una pata en su respectiva dirección como se aprecia en la figura 4-6. Estando el robot modular (a) figura 4-6, compuesto por 15 semimódulos de los cuales 3 componen el cuerpo en conexión pivote-pivote y 6 módulos para las patas en concesión cara-pivote. Este robot posee un semimódulo en la mitad para separar las patas laterales y darle distancia entre ellas, parte que resulta innecesaria debido a que las patas laterales no pueden ir hacia el frente y solo van hacia los lados. Teniendo en cuenta la observación anterior, se eliminó ese semimódulo para el prototipo (b) figura 4-6 en conexión cara pivote con la diferencia de que se coloca el pivote en la parte exterior del módulo para poder variar

la forma en que se apoyan las patas. Estos hexápodos con patas frontales y laterales al moverse describían una onda similar a la configuración oruga y serpiente del trabajo de (Lancheros Guzmán & Sanabria Galvis, 2016) al momento de desplazarse, movimientos que ya han sido establecidos y no se asemejan a la marcha mediante pasos que se quiere implementar.

- **Hexápodos con patas laterales:**

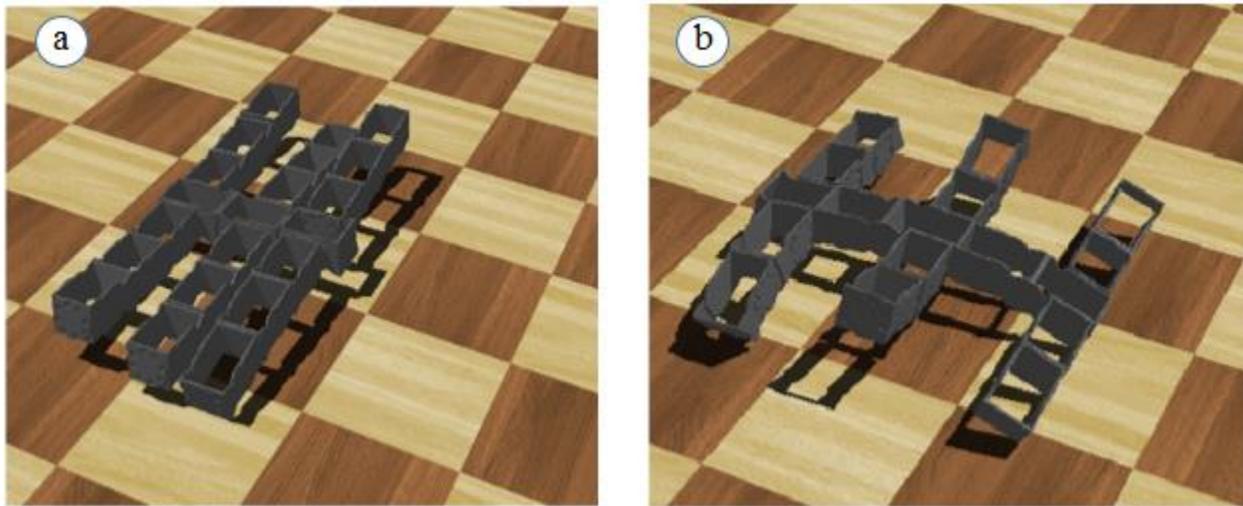


Figura 4-7. Robots hexápodos modulares con patas laterales

Estos modelos están inspirados en la morfología de los insectos utilizando módulos a los lados para asemejarse a las articulaciones de las patas e imitar los movimientos que estos logran. El modelo (a) de la Figura 4-7 utiliza 3 semimódulos en cada pata, esto con la intención de darle más altura al hexápodo y le sea más fácil sortear obstáculos, pero considerando la implementación sería más costoso construir y programar esos 21 semimódulos para una sola configuración, además que al ser tan larga las patas no se tiene el espacio suficiente para que estas se desplacen hacia adelante.

El prototipo (b) figura 4-7 se compone de 17 semimódulos, 5 para el cuerpo y 12 para las patas, en esta configuración se deja un semimódulo entre cada pata para que esta puedan avanzar. Este hexápodo requiere de movimientos rotacionales y de pivotes para dar un solo paso por lo que se busca simplificar aún más el hexápodo para su implementación.

- **Hexápodo final utilizando los módulos MECABOT 4.0**

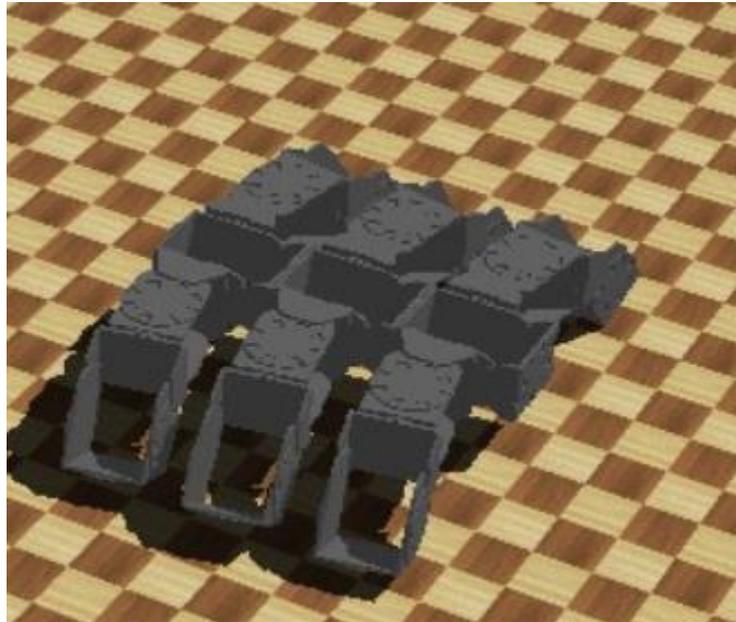


Figura 4-8. Robot hexápodo modular utilizando los módulos MECABOT 4.0

Este modelo de hexápodo de la Figura 4-8, implementa los módulos MECABOT 4.0 que son más pequeños que los MECABOT 3.0 solucionando el problema de espacio y permitiendo reducir el número de semimódulos para armar el hexápodo con un total de 15 semimódulos, 7 módulos y medio que se acoplan de tal forma que imitan las articulaciones de un insecto. Del análisis cinemático visto en el capítulo 3 se determinó que cambiar la orientación de los semimódulos, permite despegar la pata del suelo y darle dirección utilizando solo los pivotes, simplificando la secuencia para cada movimiento implementado.

4.4 Programación marchas del hexápodo MECABOT 4.0

La morfología del hexápodo está diseñada para moverse en todas direcciones con independencia del terreno coordinando las patas que se mueven y dejando las demás apoyadas sobre la superficie. Este movimiento puede ser descrito en una serie de pasos desde el momento en que las patas están apoyadas, se levantan 3 de ellas, se mueven en la dirección deseada y vuelven a apoyarse sobre el suelo repitiendo los pasos para las patas que se quedaron estáticas. Estos movimientos se recrean en el programa del robot modular hexápodo, llamando una serie de estados de movimiento para cada marcha.

Esto se logra dentro del controlador de Webots primero identificando todas las partes del robot y asignándole un tipo de dispositivo (`WbDeviceTag`) Figura 4-9, para que sea reconocido dentro del código. En caso de los semimódulos todas las partes móviles son rotacionales por lo que se definen como motores.

```

11 static WbDeviceTag motor_rueda_derecha_1;
12 static WbDeviceTag motor_rueda_izquierda_1;
13 static WbDeviceTag motor_rueda_posterior_1;
14 static WbDeviceTag motor_rueda_pivote_1;
15 static WbDeviceTag motor_pivote_1;
16 static WbDeviceTag motor_rueda_derecha_2;
17 static WbDeviceTag motor_rueda_izquierda_2;

```

Figura 4-9. Asignación de variables del robot en el software (controlador) de Webots.

Una vez las variables son identificadas, se pueden llamar dentro de las funciones de cada marcha, en donde se pueden cambiar las propiedades dependiendo del tipo. Para este caso, la línea *motor* permite mover los pivotes y cambiar su velocidad con las siguientes líneas: en donde la posición se da en radianes y se establece la velocidad con la que se moverá en la simulación. Para su implementación el controlador de la teensy 2.0 se hizo la conversión de radianes a grados de cada estado de movimiento, revisar Apéndice 8-1.

```

93 wb_motor_set_position(motor_pivote_1,-1.22173);
94 wb_motor_set_velocity(motor_pivote_1,1);

```

Todos los módulos utilizan el mismo controlador y se programan de la misma manera, por lo que se establece un número para cada semimódulo como se ve en la Figura 4-10, donde cada globo indica el número con el que se llamó cada semimódulo dentro del hexápodo, nomenclatura que también asumen todas las partes de este para diferenciarlos cuando son llamados en el programa. De esta manera para mover el pivote del semimódulo 1, se llama a al dispositivo `motor_pivote_1`, y para mover la rueda derecha del módulo 5 se llama `motor_rueda_derecha_5` dentro del programa.

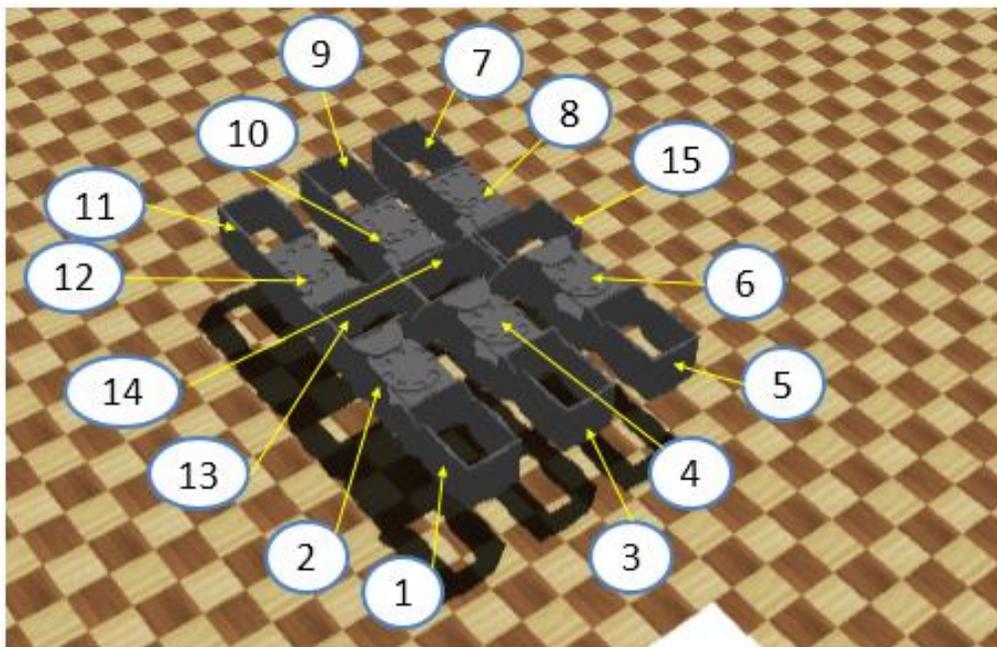


Figura 4-10. Robot hexápodo indicando la nomenclatura de cada semimódulo

De esta forma se logran realizar los siguientes movimientos, los cuales han sido elegidos por ser movimientos que los animales hexápodos son capaces de ejecutar:

4.4.1 Avance hacia adelante

El siguiente diagrama Figura 4-11, explica el algoritmo seguido para generar el movimiento hacia adelante. Este movimiento comienza con el hexápodo en posición de estabilidad con las seis patas apoyadas sobre el piso, flexionadas -70 grados.

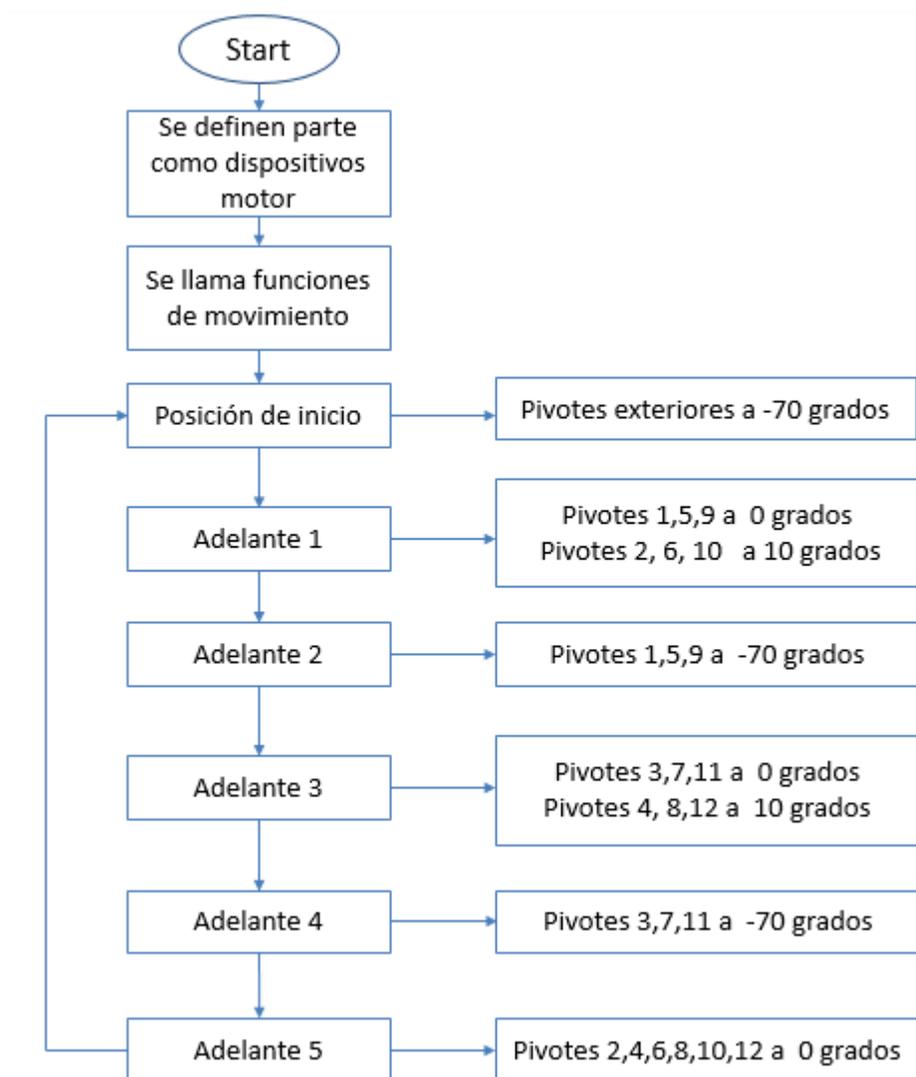


Figura 4-11. Algoritmo de programación para marchar hacia adelante

En la figura 4-12 se muestra la respuesta del robot modular hexápodo en el entorno de simulación, en la que se aprecia la secuencia dada por el programa para que este dé un paso hacia adelante.

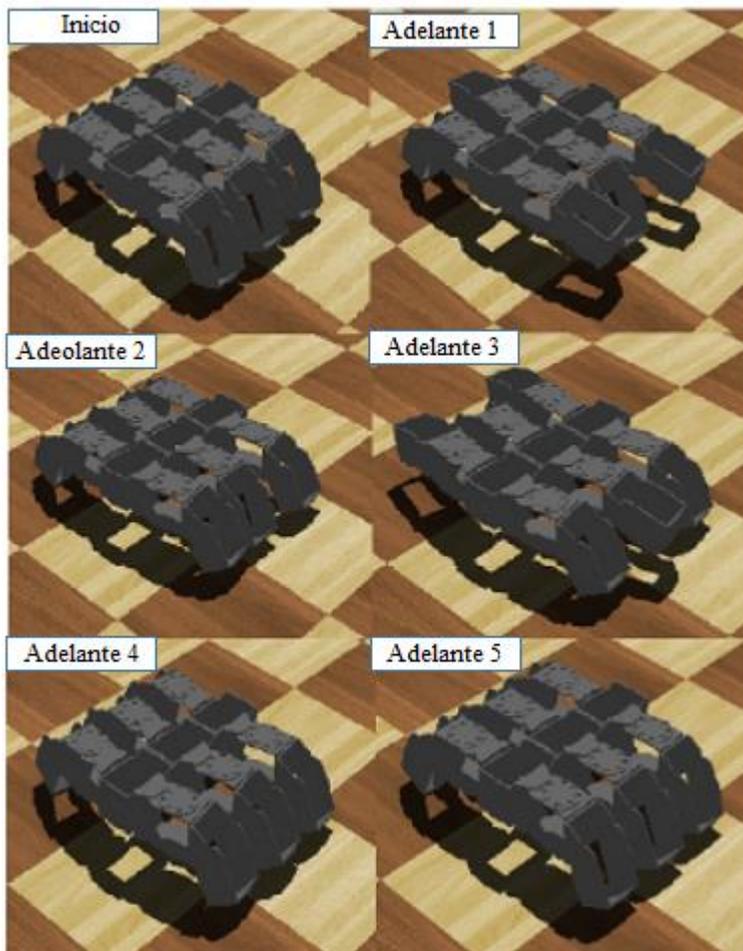


Figura 4-12. Secuencia para dar marcha hacia adelante en el simulador Webots

4.4.1.1 Análisis de simulación del avance hacia adelante

Para análisis el movimiento hacia adelante se creó una secuencia que repetía el movimiento adelante consecutivamente generando una marcha. Una vez corría la simulación con ayuda del tiempo de simulación y el piso cuadrículado se determinó la velocidad del hexápodo considerando el número de cuadros de 5x5 cm que avanzaba el robot por segundo. Se halló esta velocidad cambiando los parámetros de aceleración de los motores y su velocidad dada en rad/s para conocer bajo que parámetros se daba la mejor locomoción. La Figura 4-13 presenta la gráfica resultante de enfrentar la velocidad base de los motores y la velocidad de desplazamiento del robot modular hexápodo, para cada aceleración permitida por Webots cuyos datos se tabulan en el Apéndice 8-2.

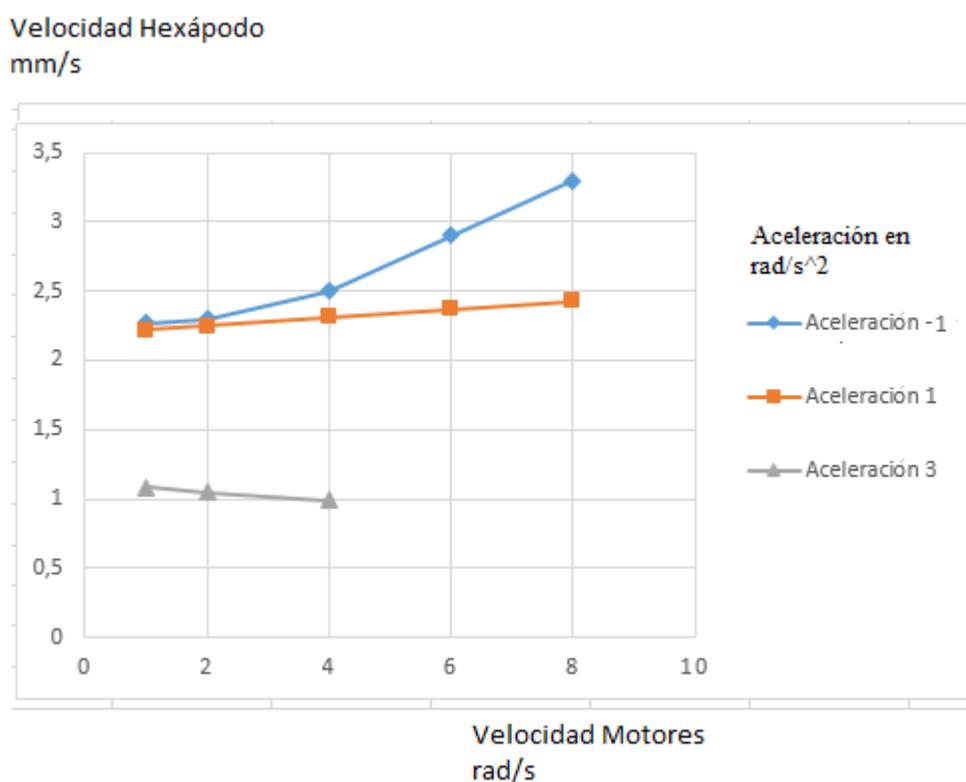


Figura 4-13. Gráficas de velocidad de la configuración hexápodo con los módulos MECABOT 4.0

De las pruebas mencionadas se encontró que la aceleración -1 es la que logra una mayor velocidad de desplazamiento, siendo un movimiento de motores desacelerado y que es la aceleración que por defecto trae Webots. Al incrementar la aceleración de los motores, contrario a lo que se creía se aprecia que velocidad del hexápodo va decreciendo, pues a mayor aceleración y velocidad de los motores, las patas modulares se van resbalando sobre la superficie desperdiciando

movimientos. Este deslizamiento se hizo más evidente para una aceleración de 3, que solo alcanza una velocidad máxima de 1,08 mm/s que fue decreciendo hasta llegar a las pruebas con velocidades de motor superiores a 4 en donde el hexápodo no avanzaba al resbalar sus patas. A partir de esto, se determinó que para los movimientos que simulan pasos es mejor iniciar los motores a una velocidad máxima e ir desacelerando a medida que se avanza.

En la Figura 4-14, se aprecia la distancia recorrida por el hexápodo en un tiempo determinado, utilizando una aceleración de -1 rad/s^2 , en donde el movimiento es estable y no se presenta deslizamiento. Trabajar con una aceleración más baja, permite variar la velocidad de los motores y apreciar un aumento en la velocidad que no se lograba con aceleraciones mayores en donde el robot se desliza. Para una velocidad de motores de 1 rad/s el hexápodo modular recorre 0,5 m en 3,48 m (208,96 s), tiempo mayor al empleado utilizando los motores a 8 rad/s, tomándole 2,46 minutos en recorrer 0,5 metros, un poco más de 1 minuto de diferencia, reflejo de como la velocidad del robot aumenta con la velocidad de los motores, mientras se mantiene una aceleración en la que el hexápodo no se deslice. En el Apéndice 8-7, se encuentra la tabla con los valores de las pruebas de simulación.

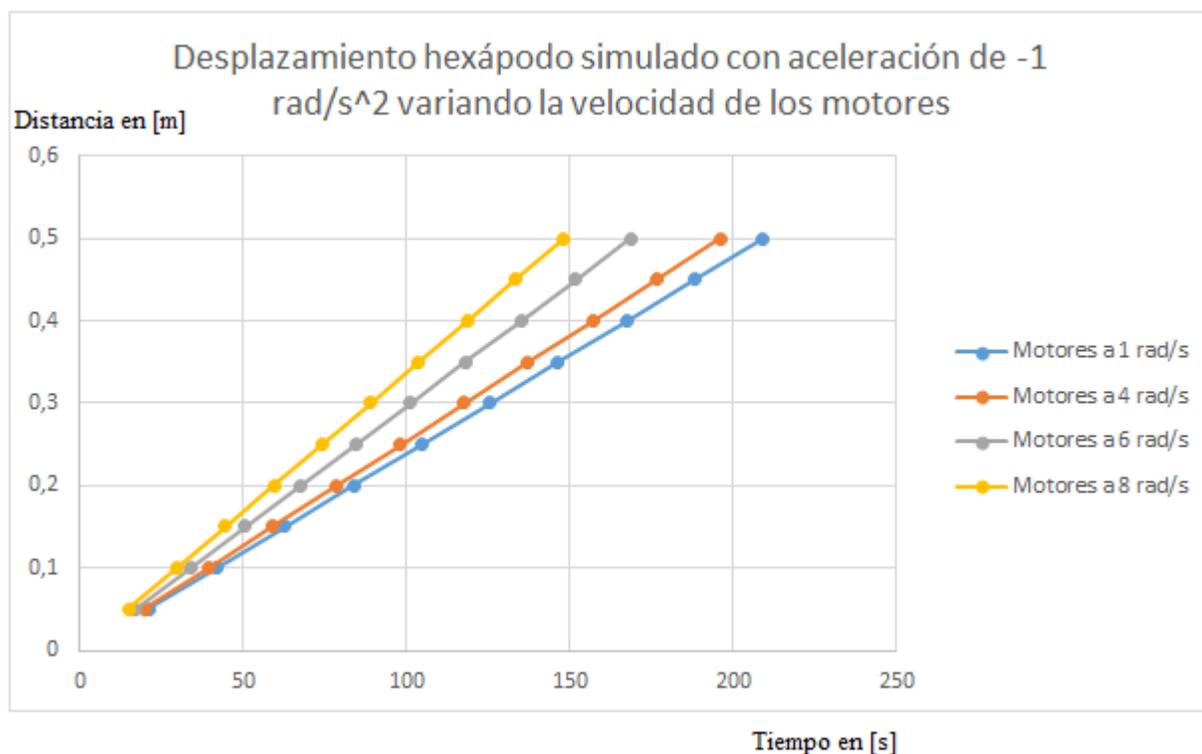


Figura 4-14. Grafica desplazamiento del robot hexápodo variando la velocidad de los motores.

4.4.2 Movimiento hacia atrás

Es movimiento hacia atrás utiliza el mismo algoritmo del movimiento hacia adelante invirtiendo los ángulos de dirección para ir hacia atrás, por lo que presenta un comportamiento igual al movimiento hacia adelante al cambiar los parámetros. En la Figura 4-15 se muestra un diagrama de flujo con el algoritmo para generar un paso hacia atrás con el robot modular hexápodo.

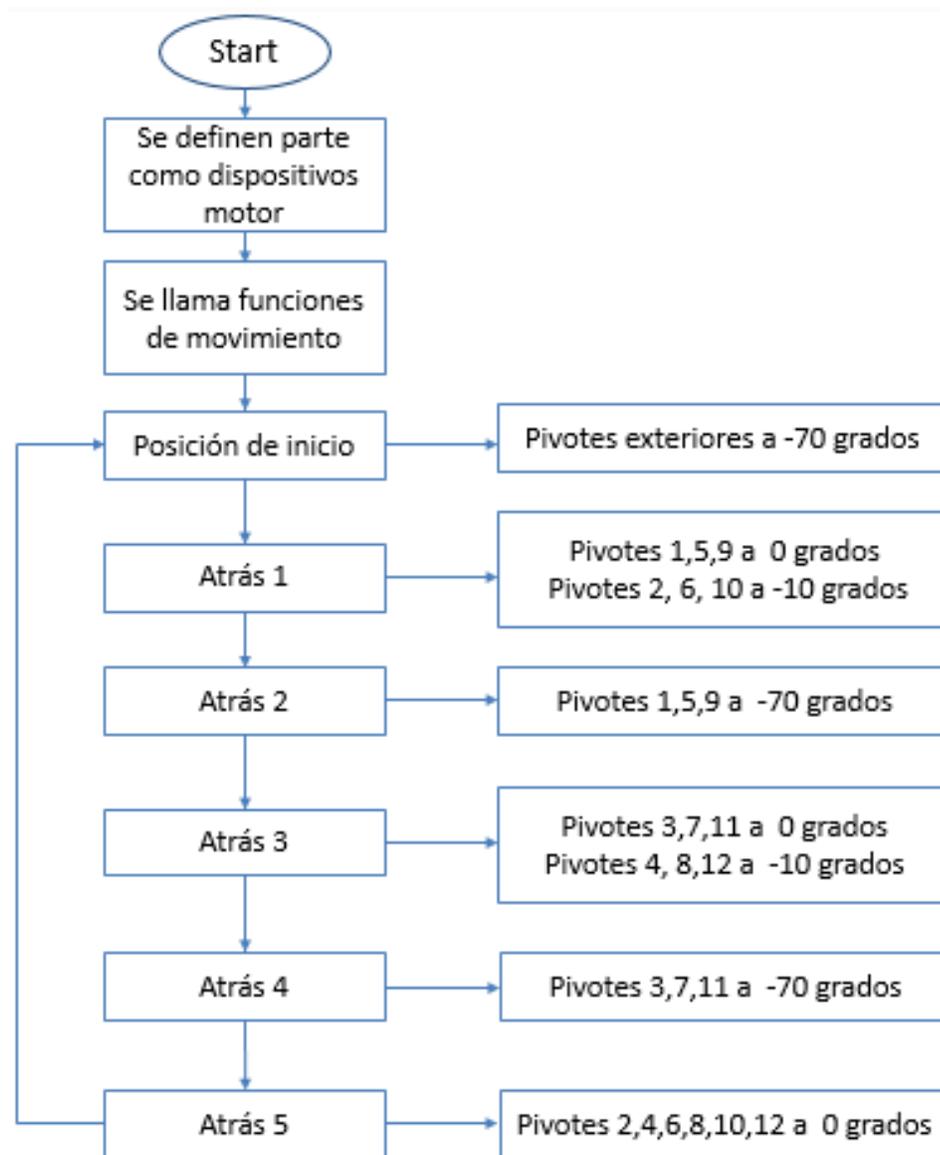


Figura 4-15. Algoritmo de programación para marcha hacia atrás.

4.4.3 Movimiento lateral izquierdo

Este movimiento se asemeja a los desplazamientos laterales de un cangrejo extendiendo y contrayendo las patas hacia los lados. Y la secuencia para lograr este movimiento se describe en el diagrama de flujo de la Figura 4-16.

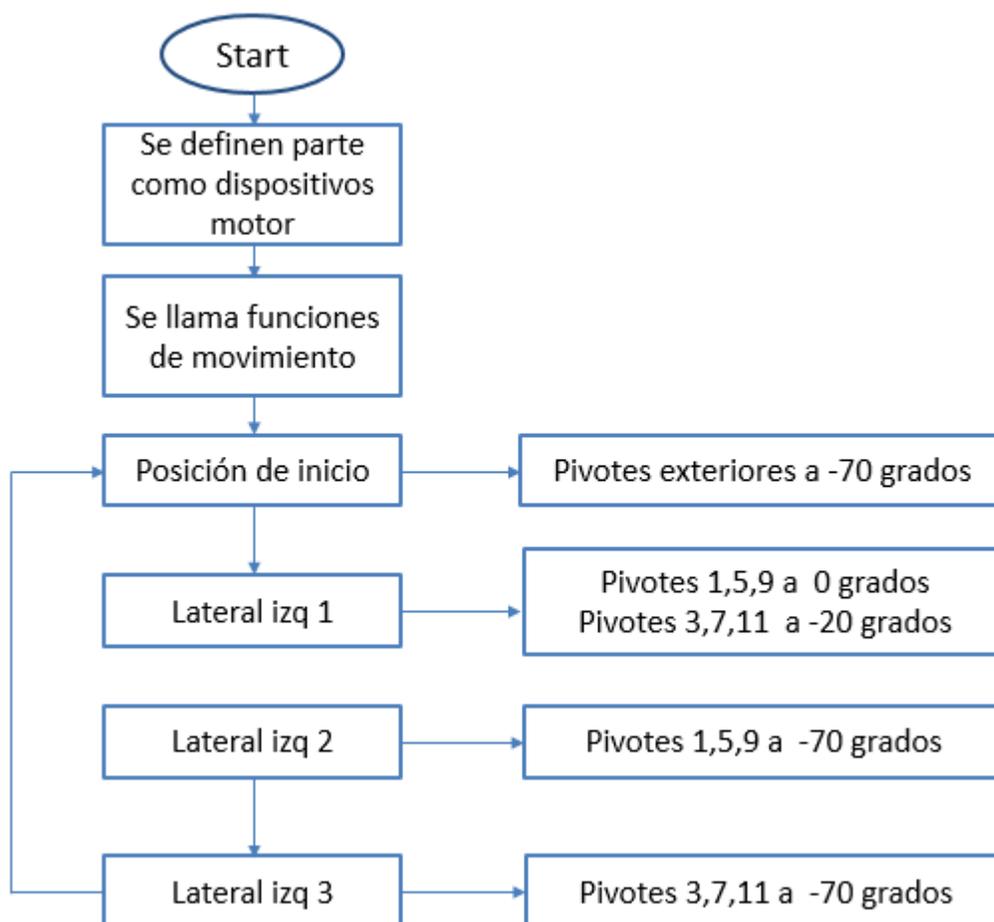
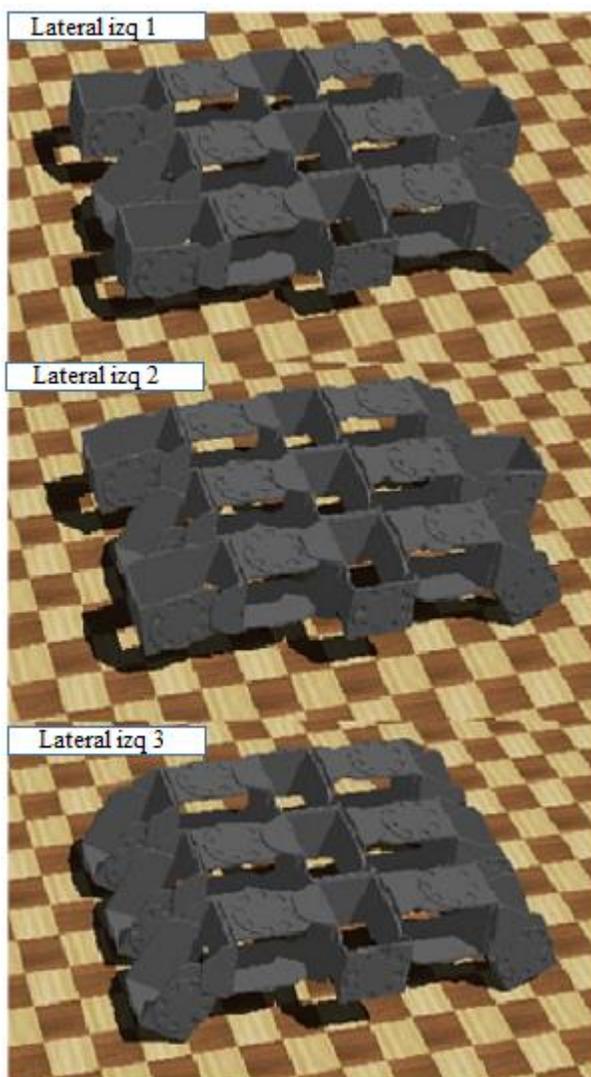


Figura 4-16. Algoritmo de programación para movimientos laterales hacia la izquierda.

En la figura 4-17 se muestra la respuesta del robot modular hexápodo en el entorno de simulación en la ejecutando la secuencia para movimientos laterales hacia la izquierda.



4-17. Secuencia para dar movimientos hacia la izquierda en el simulador Webots.

4.4.3.1 Análisis de simulación movimiento lateral izquierdo

Al igual que con el movimiento hacia adelante para conocer la velocidad alcanzada por el hexápodo en movimiento lateral, se utilizó una secuencia que llamaba la función de desplazamiento lateral varias veces. Se halló la velocidad del hexápodo para cada aceleración, cambiando la velocidad de los motores virtuales que mueven los módulos de las patas y los resultados de estas pruebas se aprecian en la Figura 4-18 y los datos se muestran en el Apéndice 8-4.

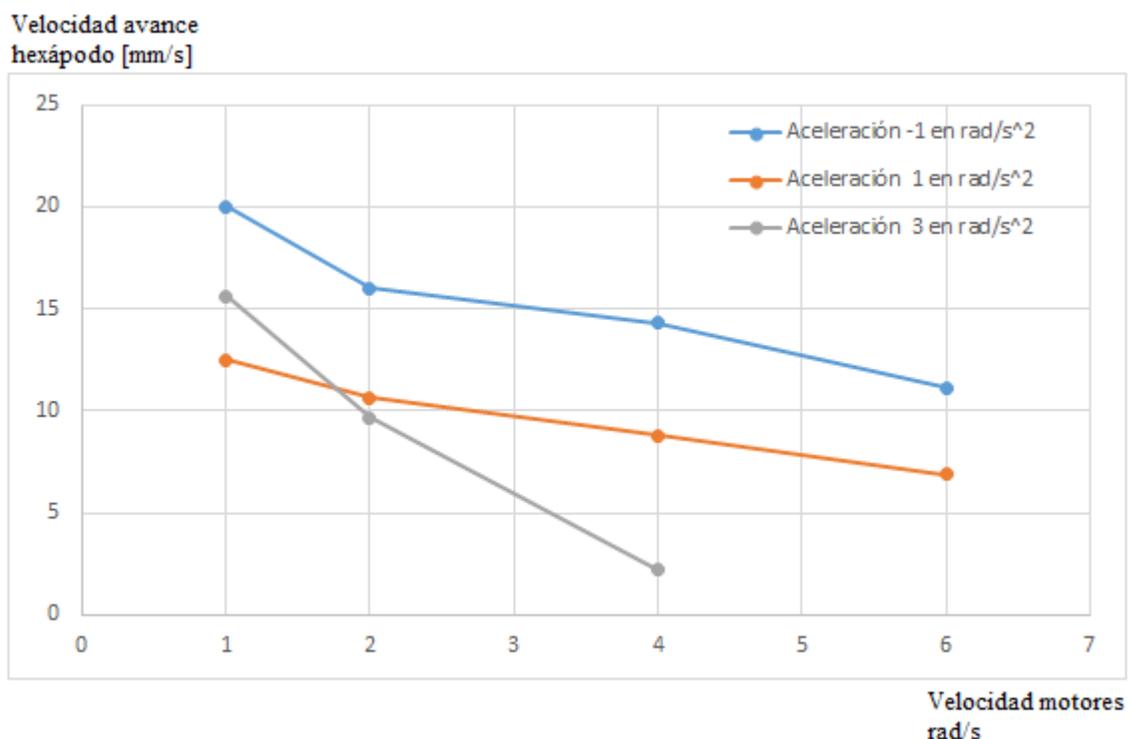


Figura 4-18. Grafica de velocidad del robot hexápodo en desplazamiento lateral.

Similar al movimiento hacia adelante, la velocidad y avance lateral de la configuración se va reduciendo a medida que se incrementa la velocidad de los motores, aunque para la aceleración 3 la velocidad uno parece soportarla sin sufrir deslizamiento. Se encontró que el movimiento lateral es el que mayor velocidad de desplazamiento alcanza superando significativamente al movimiento hacia adelante. Un ejemplo de esto es que para una aceleración de -1 y velocidad de motores de 1 rad/s, el movimiento lateral se mueve a 20 mm/s que comprado a los 2,22 mm/s del movimiento hacia adelante, son 9 veces más rápidos los desplazamientos laterales, similares a la locomoción de los cangrejos.

La Figura 4-19, presenta la gráfica de desplazamiento lateral del hexápodo variando la velocidad de los motores, con una aceleración de -1 rad/s^2 . A diferencia del movimiento hacia adelante y atrás en el que el robot levantaba las patas para avanzar, el desplazamiento lateral presenta un arrastre al extender las patas en la dirección dada, lo que genera que el robot se deslice cuando el movimiento de las patas se hace muy rápido reduciendo el avance. Como se aprecia en la siguiente gráfica, al aumentar la velocidad de los motores el tiempo que tarda en recorrer cierta distancia disminuye, como ejemplo se tiene que a una velocidad de motores de 6 rad/s , el robot tarda 45 segundos en recorrer 0,5 metros, 25 segundos más de lo que le toma al mismo robot llegar con motores a una velocidad de 1 rad/s , recorriendo medio metro en 20 segundos. La tabla de los valores de movimiento lateral simulado se encuentra en el Apéndice 8-8.

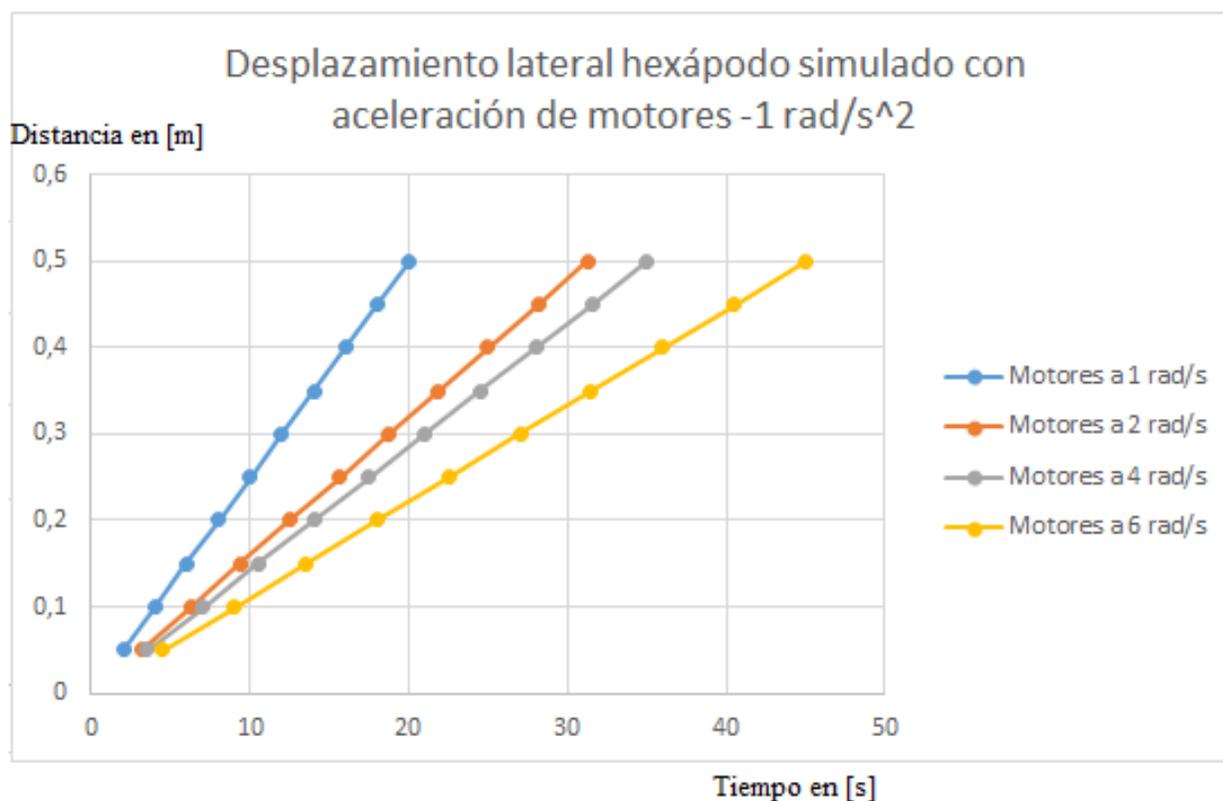


Figura 4-19. Gráfica desplazamiento lateral variando la velocidad de los motores

4.4.4 Movimiento lateral derecho

Este movimiento implementa el mismo algoritmo que el desplazamiento lateral izquierdo, invirtiendo los pivotes que se mueven para que el hexápodo valla hacia la derecha. Al ser igual que el izquierdo no hace falta realizar un análisis a detalle y su algoritmo se presenta en la Figura 4-20.

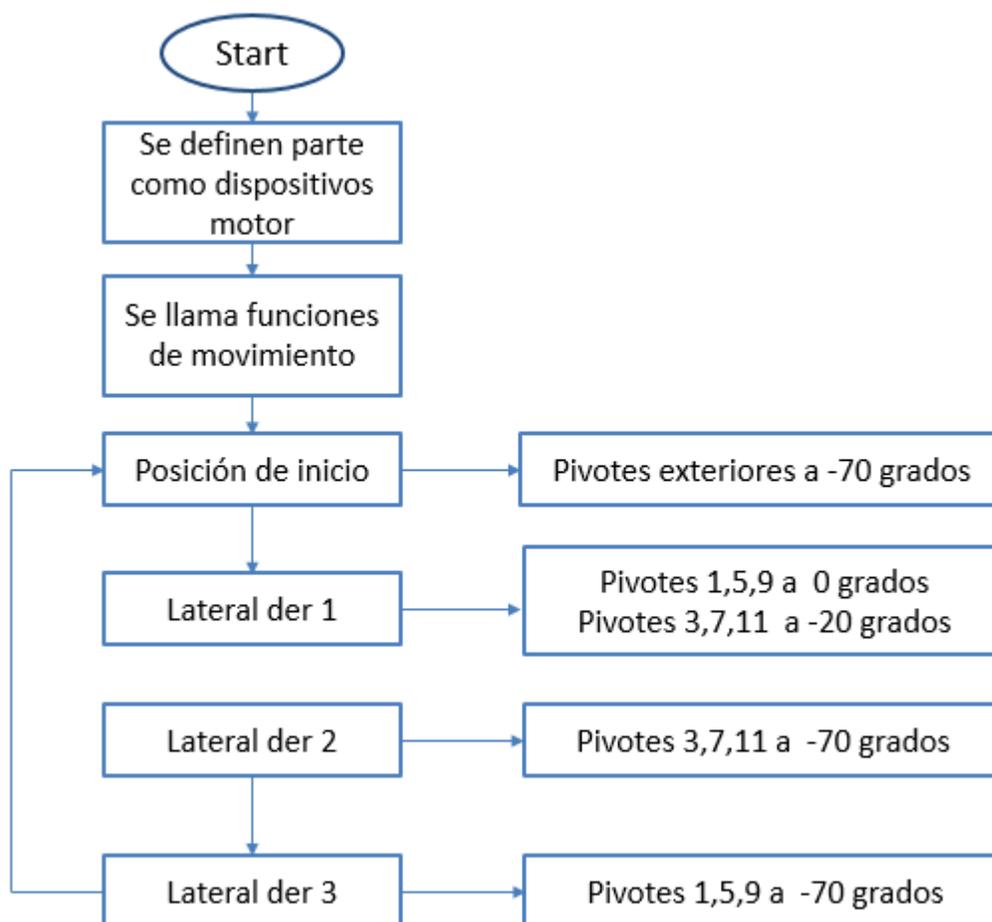


Figura 4-20. Algoritmo de programación para movimientos laterales hacia la derecha

4.4.5 Movimiento rotacional hacia la derecha

Este movimiento permite girar todo el robot hexápodo manteniendo su posición, lo que permite cambiar la dirección a la que avanza el robot, esto puede ser utilizado para evadir obstáculos. La secuencia de estados de movimientos para generar una rotación se describe en la Figura 4-21.

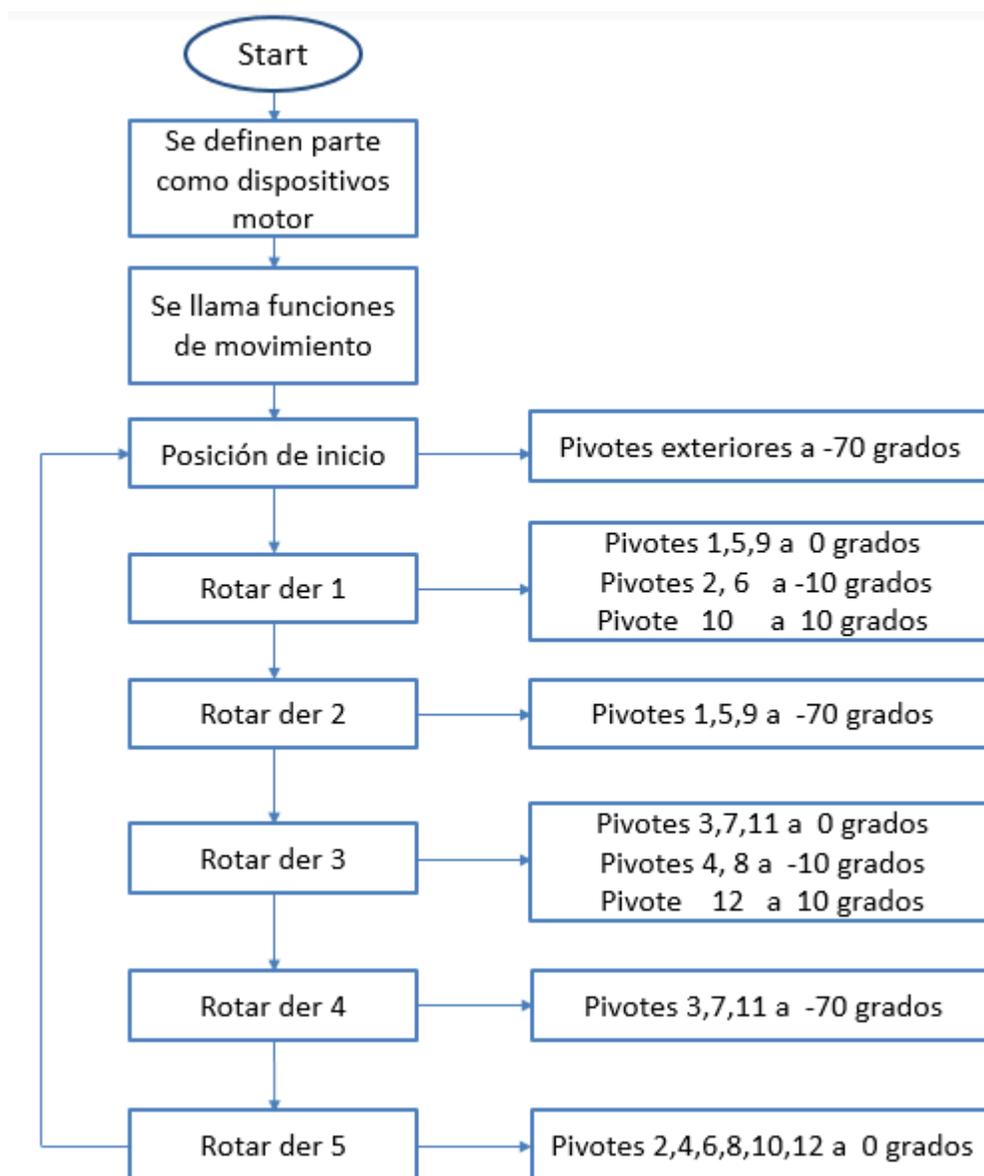


Figura 4-21. Algoritmo de programación para movimiento rotacional hacia la derecha.

En la figura 4-22 se muestra la respuesta del robot modular hexápodo en el entorno de simulación en la ejecutando la secuencia para cambiar de orientación girando hacia la derecha.



Figura 4-22. Secuencia para movimientos rotacionales hacia la derecha simulada en Webots.

4.4.5.1 Análisis de simulación movimiento rotacional hacia la derecha.

Este movimiento depende del ángulo que se le den a los pivotes horizontales pegados al cuerpo, para el caso de la imagen 4-23 fueron movimiento de 10 grados, moviendo dos patas del mismo lado 10 grados y una pata opuesta 10 grados en el sentido contrario. Este movimiento permite cambiar la orientación del hexápodo sin que haya desplazamiento. En la Figura, se muestra la gráfica de la rotación del robot y el tiempo que le toma llegar a cada ángulo, variando la velocidad de los motores. En el caso del movimiento rotacional, aumentar la velocidad de los motores no presenta un cambio significativo en el tiempo que tarda en girar a determinado ángulo, para una velocidad de motores de 1 rad/s, al hexápodo le toman 2,4 minutos en girar 90 grados y para 8 rad/s le toma 2,29 minutos (137,9 s). Girar es el movimiento más lento que posee la configuración hexápodo, pero resulta de gran utilidad al eludir obstáculos brindándole una mayor maniobrabilidad. En el Apéndice 8-9, se presenta la tabla de resultados obtenidos de la simulación del movimiento rotacional.

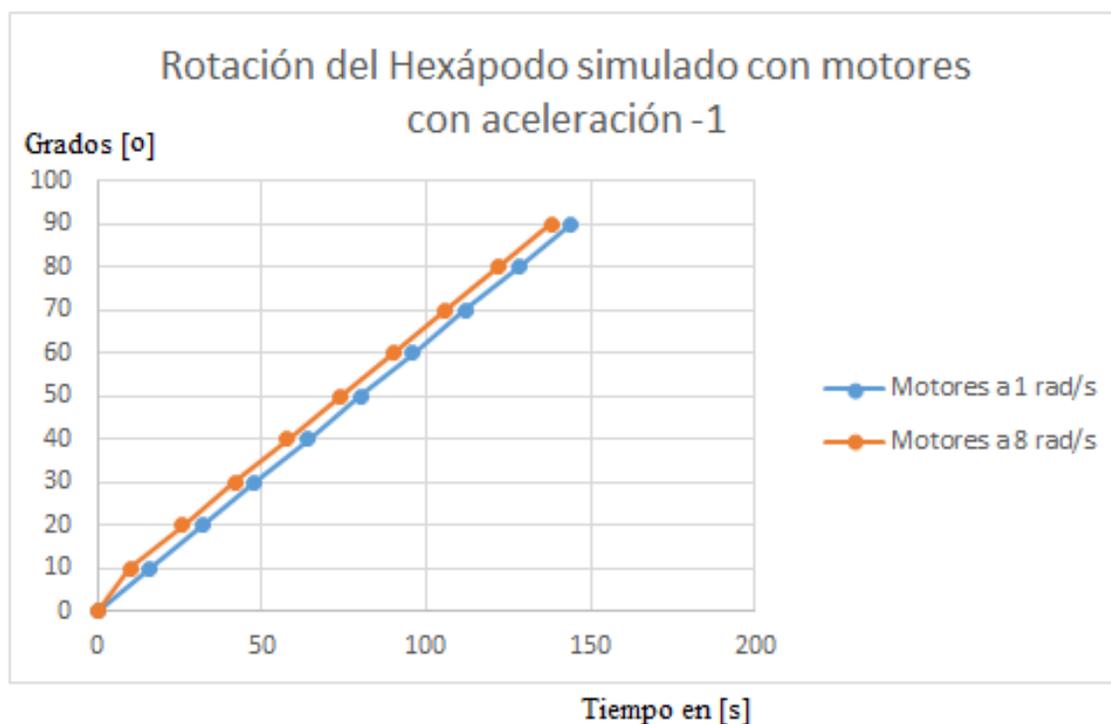


Figura 4-23. Grafica rotación hexápodo en simulación

4.4.6 Movimiento rotacional hacia la izquierda

Este movimiento implementa el algoritmo del movimiento rotacional hacia la derecha, invirtiendo los ángulos de dirección para que el hexápodo se mueva en esa dirección, por lo que no es necesario hacer un análisis a detalle y el algoritmo implementado está descrito en la Figura 4-24.

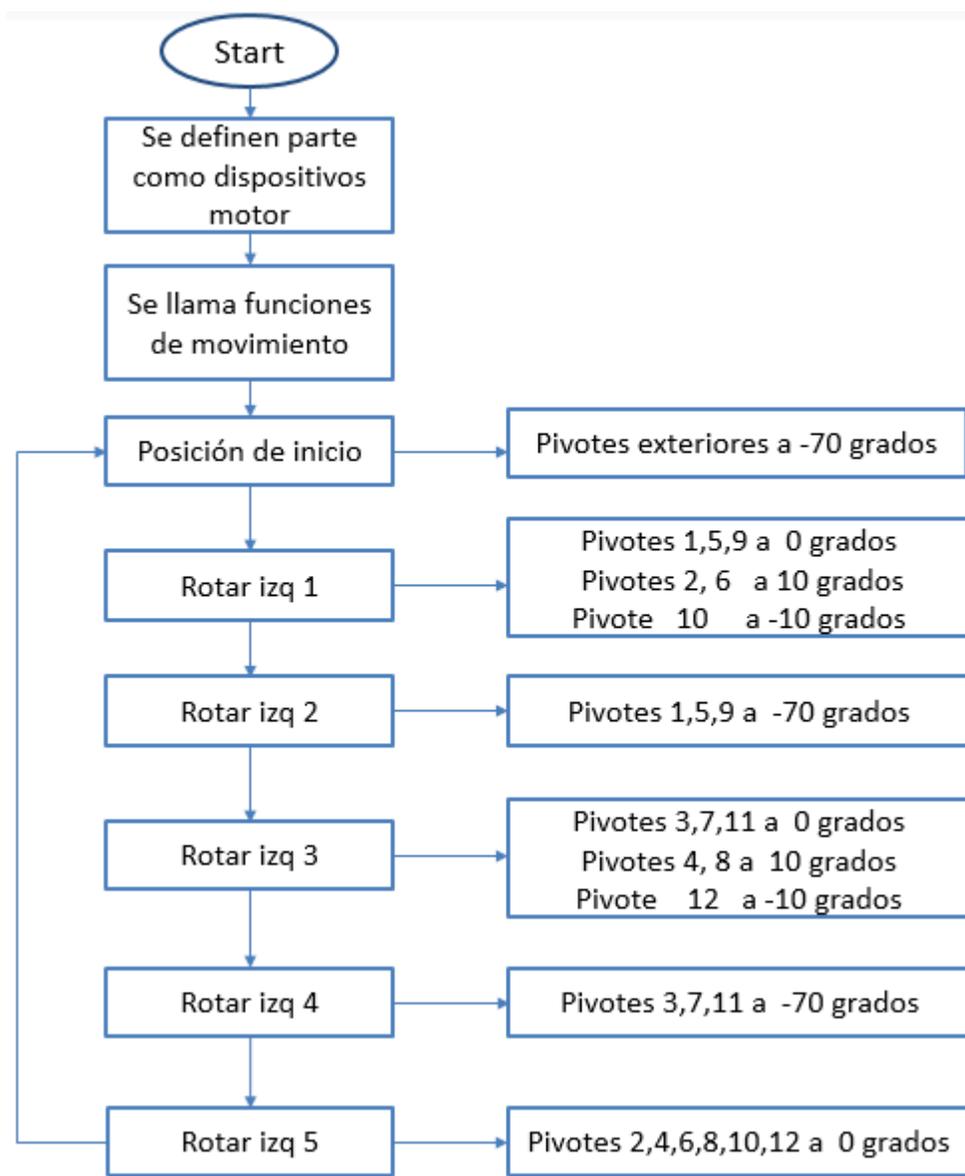


Figura 4-24. Algoritmo de programación para movimiento rotacional hacia la izquierda.

5. Implementación del robot hexápodo con los módulos MECABOT 4.0

El sistema robótico modular MECABOT 4.0, es el resultado de varios rediseños del mismo concepto de módulo auto configurable realizado por miembros del grupo de investigación Davinci de la Universidad Militar Nueva Granada. Los módulos MECABOT se conforman de dos semimódulos con varios grados de libertad, que al juntarlos se asemejan a una articulación. Los semimódulos MECABOT 4.0 fueron desarrollados rediseñando los semimódulos del MECABOT 3.0 desarrollados en el trabajo “Simulación e implementación de movimientos para sistema robótico modular considerando diferentes configuraciones”, en este se trabajaron las configuraciones oruga y serpiente, que son morfologías apodos (sin patas) en donde se generaban movimientos a partir de oscilaciones con los pivotes que le daban amplitud a las ondas descritas. El siguiente paso era realizar más configuraciones, pensando en implementar la rueda y una configuración caminante muy usadas en el estudio de robótica modular, pero con los módulos 3.0 se apreció que no tendrían la fuerza para levantar otros módulos por lo que se decidió cambiar los servo motores por unos de piñonera metálica, aun así al simular las configuraciones su tamaño dificultaba los movimientos y le daba peso, por lo que la solución fue reducir esos módulos haciendo los más compactos y livianos con motores más fuertes. Estos cambios dieron origen al módulo MECABOT 4.0 el cual fue pensado para moverse de forma independiente equipado con su propia batería, conservando parte del sistema electrónico de su predecesor.

En este capítulo se describen las características mecánicas del MECABOT 4.0 que fueron los módulos utilizados para implementar la configuración hexápodo de este trabajo. El diseño de los módulos y la elección de componentes del MECABOT, hacen parte del proyecto de investigación del grupo de investigación Davinci. En este trabajo se realizó la construcción de los módulos MECABOT 4.0 como aporte al proyecto de investigación, de los cuales se utilizaron siete módulos y medio para crear la configuración hexápodo.

5.1 MECABOT 4.0

En la figura 5-1 se aprecia la reducción que tuvo el modelo 4.0 (semimódulo de abajo) con respecto a su versión anterior (semimódulo de arriba), en donde su mayor diferencia se dio al colocar los pivotes sobre las caras laterales del semimódulo para reducir espacio.

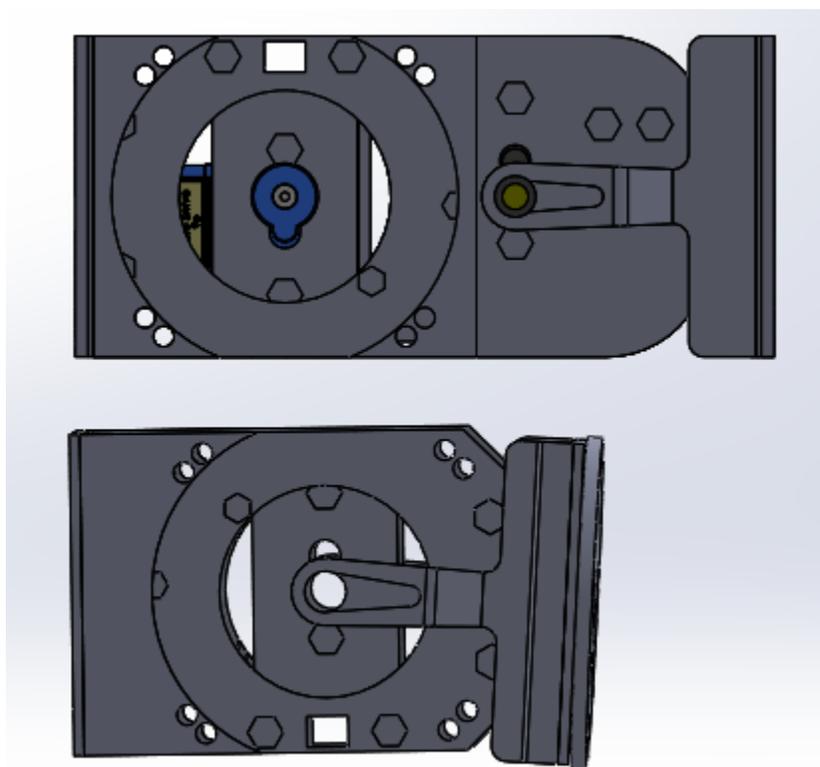


Figura 5-1. Semimódulos MECABOT 4.0 y 3.0 vista lateral.

Este ajuste de tamaño implica que solo se usarán 4 motores para el semimódulo MECABOT 4.0 con respecto a los 6. En figura 5-2 se definen los componentes de los módulos en donde la principal diferencia de los semimódulos que lo conforman, es una ranura en el semimódulo trasero para poner la batería.

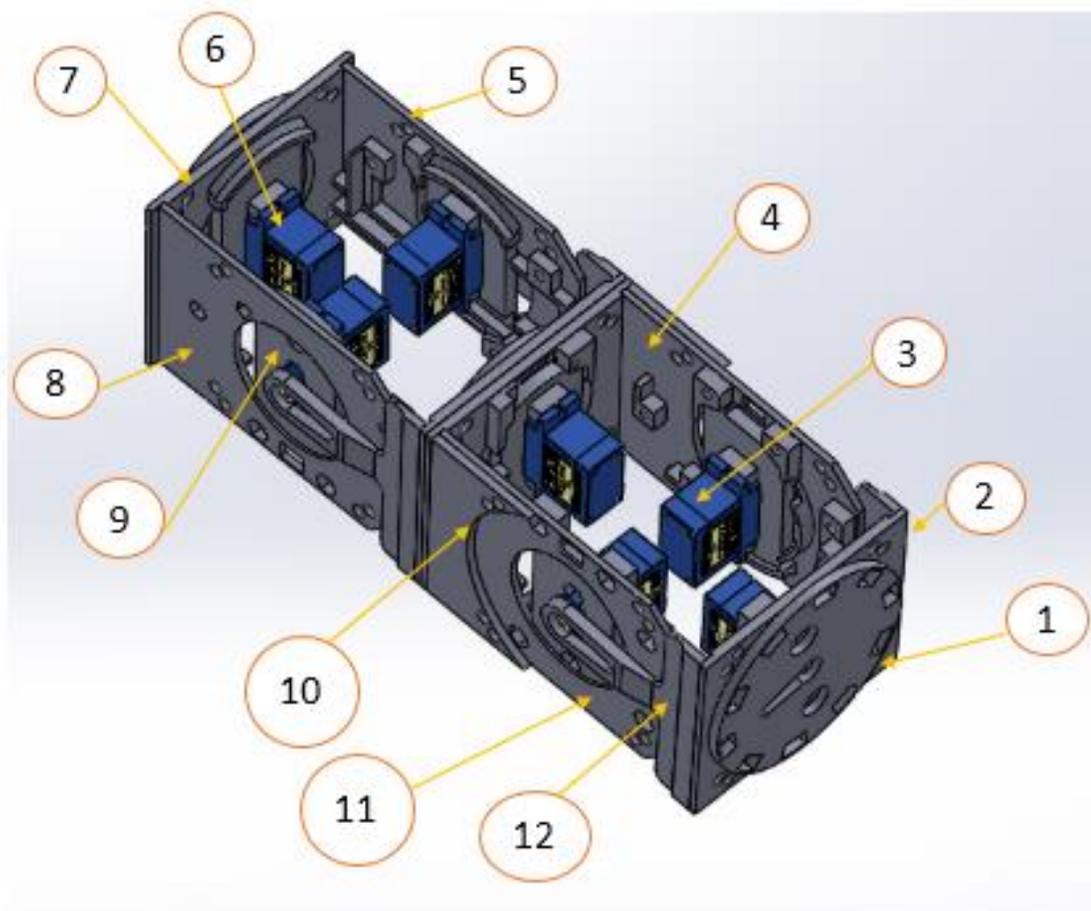


Figura 5-2. Módulo MECABOT 4.0

En la tabla 5-1 se definen las piezas según el número asignado y la cantidad usada en el módulo MECABOT 4.0.

Número de elemento	Pieza	Cantidad
1	Rueda para imanes	2
2	Cara frontal	2
3	Motor pivote	4
4	Cara izquierda	1
5	Cara izquierda pila	1
6	Motor caras	3
7	Cara trasera	2
8	Cara derecha pila	1
9	Soporte motor	7
10	Soporte rueda	4
11	Cara derecha	1
12	Pivote	2

Tabla 5-1. Componentes módulo MECABOT 4.0

Las piezas del módulo fueron construidas con la técnica de impresión 3D utilizando material ABS (Acrilonitrilo Butadieno Estireno) el cual es un plástico resistente al impacto usado en la industria automotriz y doméstica. Una vez impresa se procede al proceso de ensamble descrito a continuación:

1. **Impresión de las piezas y lijado:** La técnica de impresión 3D Figura 5-3, consiste en ir colocando capa por capa de material caliente mientras se va dibujado. Este proceso no genera un acabado superficial limpio por lo que una vez impresas, las piezas tienen que ser lijadas.

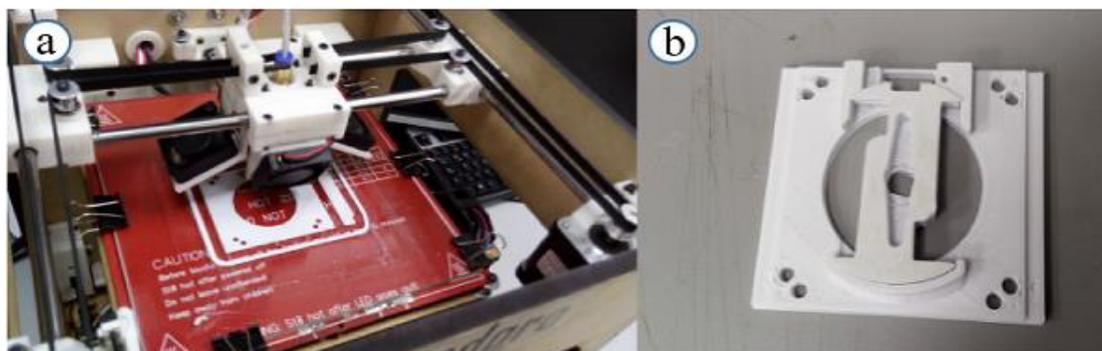


Figura 5-3. a) Impresión de la pieza, b) pieza impresa.

2. **Ensamble de elementos electrónicos:** para sujetar la mayoría de elemento se usaron tornillos Bristol M5, y para la teensy 3.2 y el regulador se utilizaron tornillos más delgados que tuvieron que ser cortados. Los cables de los componentes electrónicos que resultaron demasiado largo fueron cortados y soldados para estar más compactos en el módulo, como se ve en la Figura 5-4.

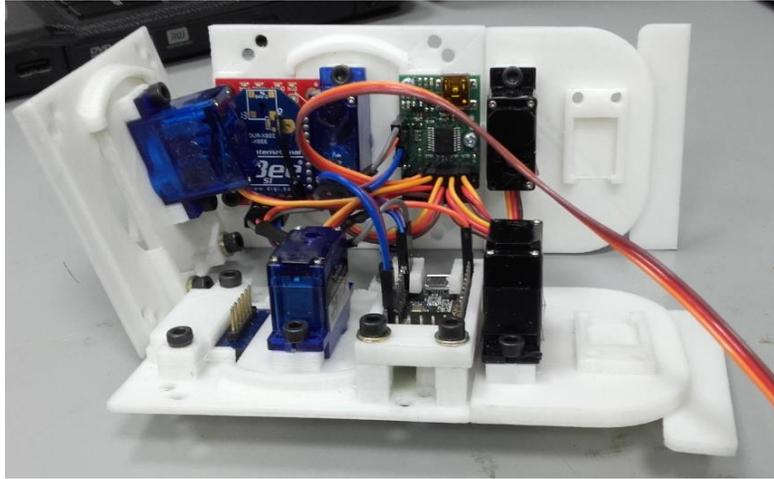


Figura 5-4. Ensamble de piezas electrónicas.

3. **Unión de las caras de módulo:** Una vez ensamblados los componentes electrónicos se procede a unir las caras con cloruro de metileno, cuidando que las caras y pivotes no queden torcidos .Figura 5-5

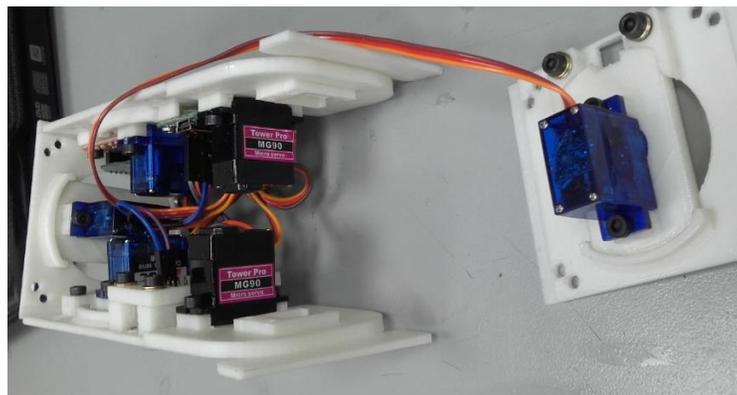


Figura 5-5. Unión de caras

4. **Batería y puesta en marcha:** Una vez se arma el modelo se coloca la batería y se conectan los componentes a la fuente de energía, comprobando que todos los sistemas se enciendan y funcionen correctamente, como se ve en la Figura 5-6 que presenta el módulo MECABOT 4.0 totalmente ensamblado.



Figura 5-6. Módulo MECABOT 4.0 implementado.

En la figura 5-7 se compara el tamaño de los módulos MECABOT 3.0 (módulo inferior) y se predecesor 4.0 (modulo superior). Esto también reduce el peso del módulo de 500 gramos a 380 gramos para el MECABOT 4.0

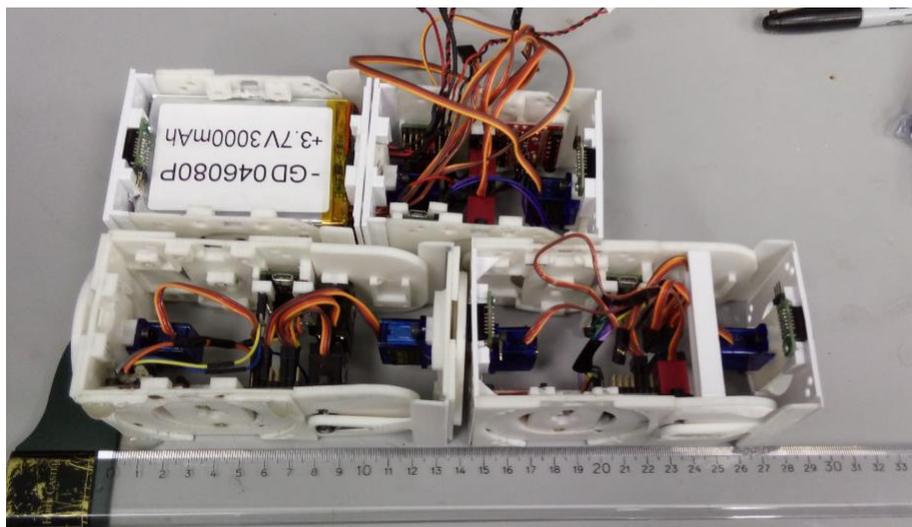


Figura 5-7. Módulos MECABOT 3.0 y 4.0

El módulo MECABOT 4.0 utiliza los mismos componentes del MECABOT 3.0, debido a que probaron ser funcionales en las configuraciones oruga y serpiente de trabajos anteriores del grupo de investigación, reduciendo el espacio y cambiando los motores de piñonería plástica de los pivotes, que no brindaban el torque suficiente para levantar otro módulo acoplado, capacidad necesaria en las nuevas configuraciones rueda y hexápodo, por motores de piñonería metálica HD-1810Mg de giro limitado (Figura 5-7) más fuerte que los anteriores, además de un regulador Step-Down 5v/2.5 A D24V22F5 (Figura 5-8), que provee un flujo de corriente constante que ofrece protección por polaridad invertida y sobre picos de corriente.



Características:

- Torque(4.8V): 3.1 kg-cm (43.05 oz/in)
- Torque(6.0V): 3.9 kg-cm (54.16 oz/in)
- Velocidad: 0.16 sec (4.8V) | 0.13 sec (6.0V)
- Rango de operación: 4.8 - 6.0 DC Volts
- Peso : 15.8 g (0.56 oz)
- Tipo de rodamiento: rodamiento de bola x 2
- Tipo de Motor : Motor DC
- Tipo de engranaje : Plastico
- Temperatura de operación: -20°C - 60°C
- Frecuencia de trabajo: 1520µs / 250hz

Dimensiones: 22.8 x 12.0 x 29.4 mm (0.90 x 0.47 x 1.16 in)

Figura 5-8. Motor HD-1810MG y sus características.

Tomado de: (Limitado, s.f.)



Dimensiones:

- Compacto: 17.8 mm x 17.8 mm x 8 mm (ancho x largo x alto)
- Dos agujeros de montaje con tornillo #2-56 o M2.

Características:

- Salida fija: 5 V
- Salida de corriente continua máxima: 2.5 A
- Eficiencia: 85% a 95% dependiendo del voltaje de entrada y carga.
- Frecuencia de switching: 400 kHz
- Protección de voltaje inverso, sobrecorriente y sobrecalentamiento.
- Consumo de 1 mA sin carga reducible por medio del pin de bajo consumo
- Pin de "buena regulación" para saber el estado del regulador

Figura 5-9. Step-Down 5v/2.5 A D24V22F5 y características.

Tomado de: (Electronics, s.f.)

5.2 Caracterización de los servomotores

Al ubicar los pivotes se debe cuidar de que estos estén alineados ya que trabajan en conjunto, debido a que si no lo están las cargas no se distribuyen equitativamente y se puede presentar desgaste en uno de los dos motores. Estos motores se caracterizan con el programa Pololu Maestro Control Center, que permite cambiar la configuración de los motores incluyendo la aceleración, los límites de giro, y el Baut Rate que viene siendo la señales que se transmiten por segundo, y para estos motores se dejaron en 115200. Se mueven los motores con el controlador Micro-servomotor ubicando los pivotes en la misma orientación del cuerpo del módulo, tomando esa posición como cero, de ahí se usa un transportador para determinar los valores de cada ángulo y así tabularlos, obteniendo la ecuación que se utiliza en el programa para enviar el valor que reconoce el controlador para mover los motores con el ángulo deseado. La tabla y la función de movimiento se realizaron en Microsoft Excel (Figura 5-9), se puede apreciar que las gráficas van en sentido contrario debido a la orientación de cada motor ubicado a la derecha e izquierda del módulo y varían sus valores por las propiedades físicas de cada motor.

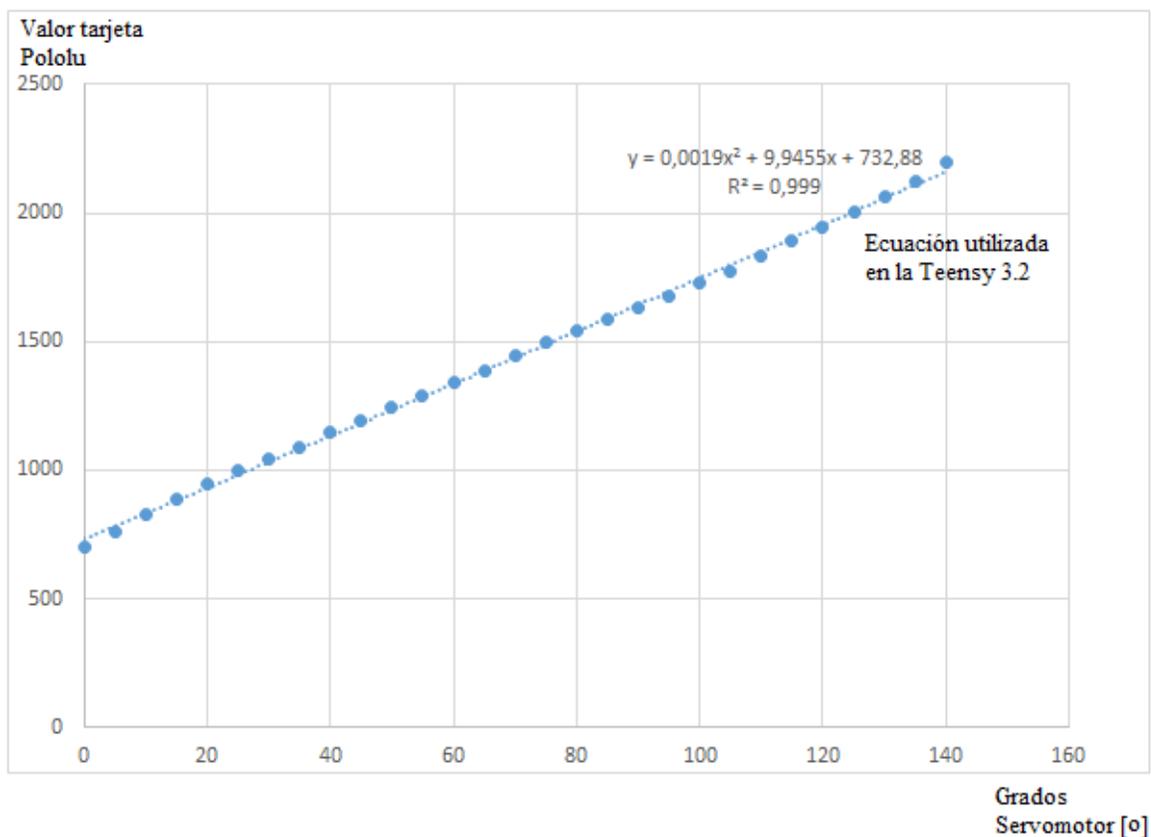


Figura 5-10. Caracterización servomotor derecho.

Figura 5-10. Caracterizaciones del motor izquierdo del pivote.

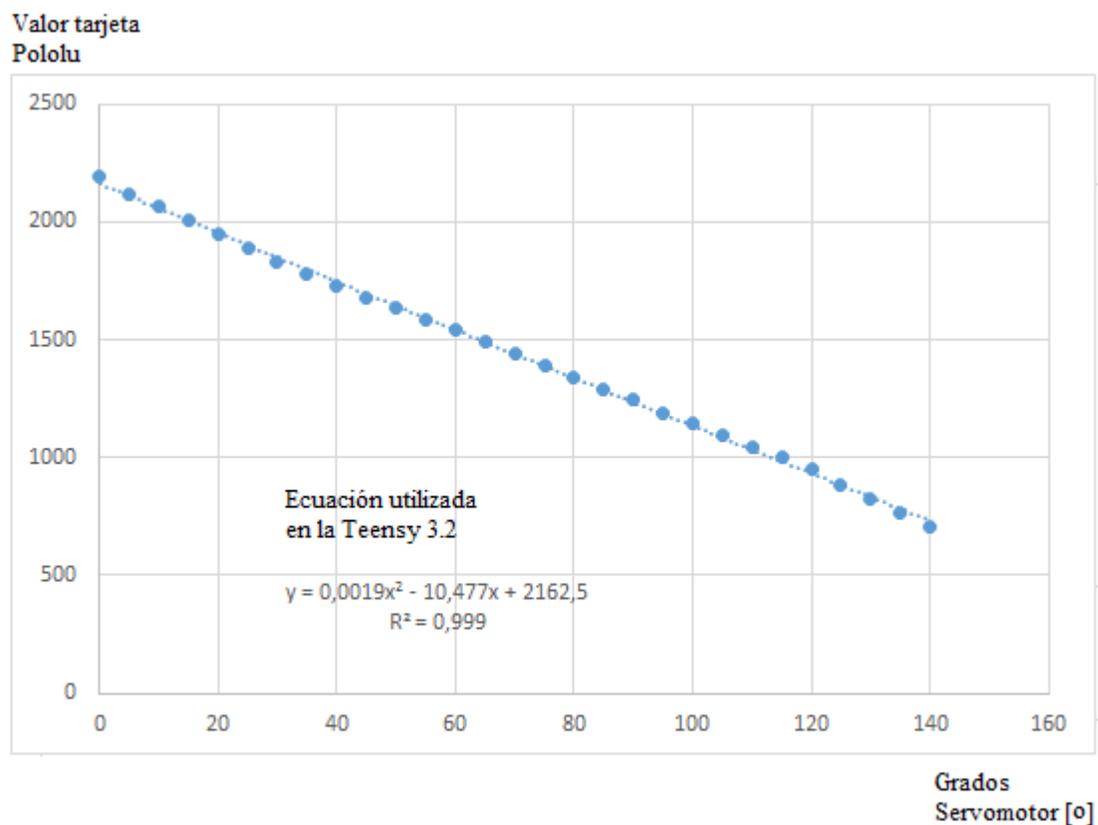


Figura 5-10. Caracterización servomotor izquierdo.

En el programa de la Teensy 3.2 utilizada como tarjeta de programación se copian las funciones de cada motor, de tal forma que si se quiere mover el pivote 90 grados se llaman a las dos funciones que calculan el valor que el controlador recibe para mover ese ángulo, siendo de 1632,5 para el motor derecho y de 1244,5 para el izquierdo, el valor que debe recibir el controlador para mover el pivote a 90 grados. Estos motores comparten el mismo valor de posición a 70 grados con 1442,25. Este valor se usó para la posición de estabilidad del hexápodo.

6. Resultados de implementación del Hexápodo MECABOT 4.0

En este capítulo se muestran los resultados de la implementación del robot hexápodo utilizando los módulos MECABOT 4.0. En este se realizó la interfaz de comandos del robot, que ejecuta el código y le indica el tipo de movimiento que se quiere de forma inalámbrica, gracias al uso de las Xbee s2 en el que se establece un xbee coordinador que transmite la señal a los xbee receptores que posee cada módulo. Todos reciben la misma señal pero para cada módulos individualmente tienen trabajos específicos por lo que se les da un id de 1 a 15 que el programa reconoce para distinguir cada módulo y ejecutar sus estados de movimiento. Las pruebas de locomoción fueron realizadas dentro de las instalaciones de la Universidad Militar Nueva Granada en el laboratorio de robótica y zonas verdes, en el que se evaluaron sus movimientos cambiando la aceleración y los ángulos para el movimiento detallando las secuencias en cada paso, así como la velocidad a la que se mueve el hexápodo. La configuración se probó en tres terrenos diferentes suelo liso, pasto y terreno rocoso para comprobar las capacidades de los hexápodos con independencia del terreno en la configuración con módulos MECABOT 4.0. Se elaboraron 15 semimódulos y se ensamblaron para formar la configuración hexápodo vista en la figura 6-1.

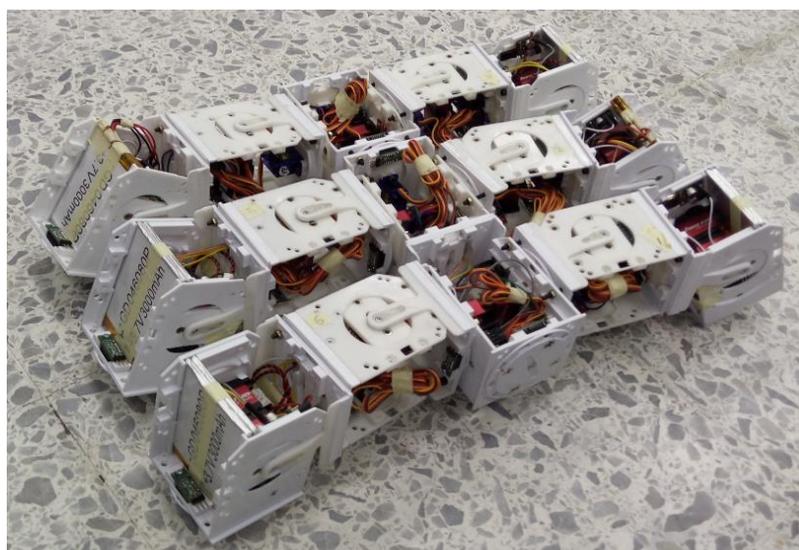


Figura 6-1. Robot modular hexápodo a partir de módulos MECABOT 4.0

6.1 Diseño e implementación de Software de comandos e interfaz de Usuario

Se desarrolló un programa en MATLAB ® para el comando de los movimientos, habilitando el puerto serial para el Xbee y así enviar el comando que activa determinado movimiento en el programa, para cada botón de la interfaz que corre una función, para él envió de esta al programa. En este también se incluye una serie de intervalos de tiempo al que se le denominó tiempo de ejecución, que se varía para generar interrupciones entre pasos haciendo movimientos más rápidos o lentos y se puede observar en la Figura 6-2. Cada unidad que se varía en el tiempo de ejecución, corresponde a medio segundo que espera el hexápodo en cada paso de la secuencia que se quiera implementar, así si se corre la secuencia para ir hacia adelante cada opción espera ese tiempo para pasar a la siguiente como puede ser pasar de Adelante 1 a Adelante 2. En cada función se envía una cadena de 6 caracteres, en donde 1 carácter corresponde al valor dado en el tiempo de ejecución y el 4 corresponde a número asignado para ejecutar cada movimiento, los demás caracteres se dejan para agregar funciones en futuros proyectos con el hexápodo, como se observa en la tabla 6-1 con la “005002” que es el comando para el movimiento lateral con 2 interrupciones que le da un tiempo de 1 segundo entre la ejecución de cada estado de movimiento.

Caracter	1	2	3	4	5	6
	0	0	5	0	0	2

Número de movimiento → (apunta al carácter 3)
 Tiempo de ejecución entre estados de movimiento → (apunta al carácter 5)

Tabla 6-1. Cadena de caracteres para comando de hexápodo.

A continuación se presentan las funciones que realiza cada objeto en la interfaz para su correcto uso y comandar el robot hexápodo. El carácter 6 nombrado v, es el tiempo de ejecución que varía dependiendo del usuario.

- **Botón Inicio:** Lleva el hexápodo a su posición de estabilidad en el que se apoya sobre su seis patas flexionadas a 70 grados. Se envía la cadena “00100v”
- **Botón Cero:** extiende las patas del hexápodo para dejarlo en posición de reposo sobre el piso. Se envía la cadena “00800v”

- **Botón Adelante:** Ejecuta la secuencia para que el robot de un paso hacia adelante. Se envía la cadena “00200v”
- **Botón Atrás:** Ejecuta la secuencia para que el robot de un paso hacia atrás. Se envía la cadena “00300v”
- **Botón Derecha:** Ejecuta la secuencia para que el robot de un paso lateral hacia la derecha. Se envía la cadena “00400v”
- **Botón Izquierda:** Ejecuta la secuencia para que el robot de un paso lateral hacia la izquierda. Se envía la cadena “00500v”
- **Botón Rotar Derecha:** Gira el hexápodo hacia la derecha sobre la misma posición. Se envía la cadena “00600v”
- **Botón Rotar Izquierda:** Gira el hexápodo hacia la izquierda sobre la misma posición. Se envía la cadena “00700v”
- **Botón Marcha:** Ejecuta la secuencia de paso hacia adelante repetidas veces para que el robot marche. Se envía la cadena “00900v”
- **Boton CDirecta:** Ejecuta la cinemática directa de la pata del hexápodo ingresando los ángulos de las articulaciones en los espacios theta 1, theta 2, theta 3, theta 4 que comprender las partes móviles del módulo MECABOT 4.0 graficando la posición final de este.
- **Tiempo de ejecución:** En este espacio se selecciona el tipo de interrupción entre cada estado de movimiento haciéndolos más rápido o lentos dependiendo del valor ingresado. Cada unidad deja un tiempo de espera 0.5 segundos.

La Figura 6-2 presenta la interfaz gráfica que corre el programa de control del robot modular hexápodo creada en el guide de matlab, que genera un archivo GUI (fig.) en el que se hace la interfaz agregándole el aspecto y los objetos como gráficas y botones que emplea el programa, también genera un archivo (.m) que contiene todo el código del programa.

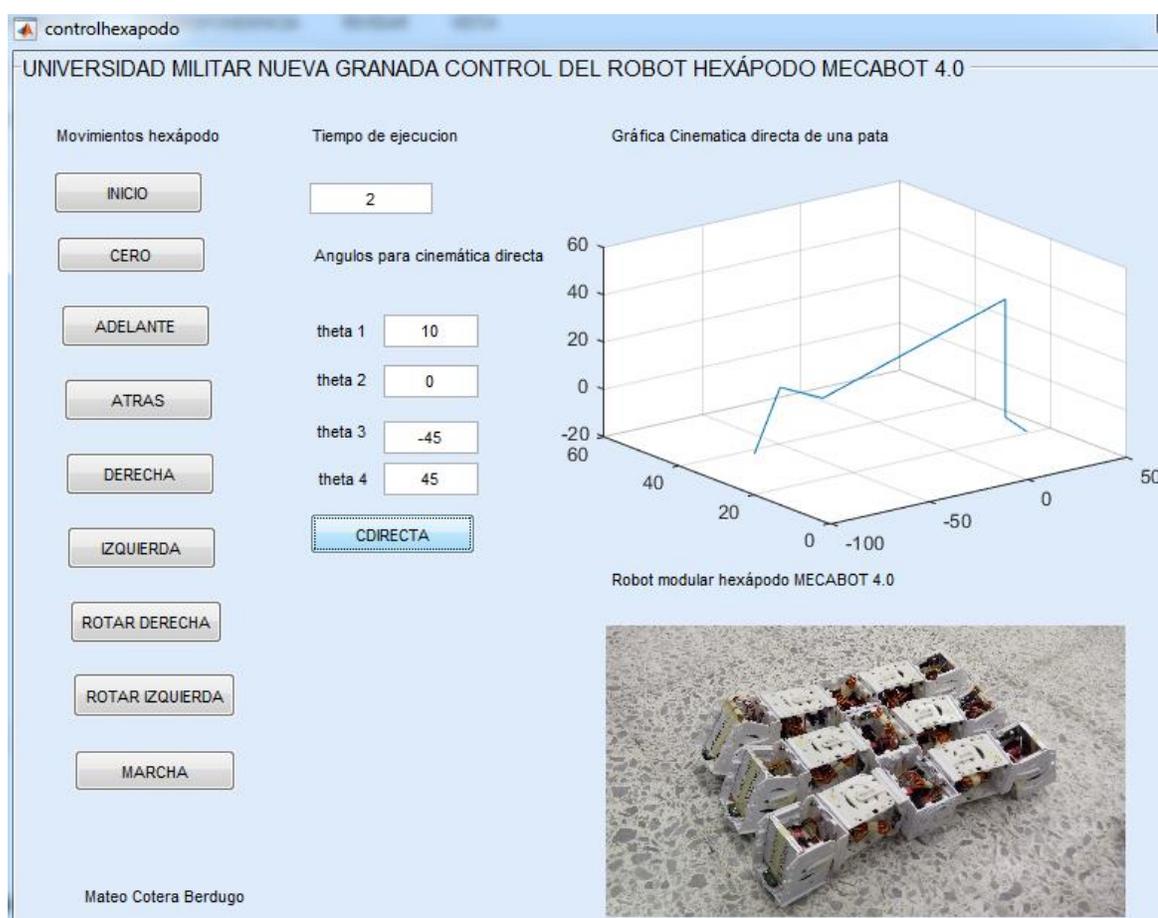


Figura 6-2. Interfaz de usuario para control de hexápodo

6.1.1 Programa de la tarjeta de control Teensy 3.2

Para manejar cada módulo se utilizó la tarjeta de procesamiento Teensy 3.2 (Figura 6-3), cuyo programa de comando fue programado y cargado con el software Arduino utilizando la librería TeensyDuino para trabajar con esta tarjeta en específico. Además se utilizó la librería PololuMaestro.h que permite utilizar el Driver de los motores Micro Pololu Maestro, convirtiendo los valores de ángulo de cada estado de movimiento, en valores equivalentes que el driver utiliza

para mover los motores a su respectivo ángulo. Este programa es igual para todos los módulos con la diferencia de que se asigna un ID para identificar el modulo y coordinar sus movimientos en conjunto con los demás para mover la configuración hexápodo. Por medio de transmisión serial recibe la cadena de caracteres desde matlab y la convierte en un trama de bits con el movimiento y tiempo entre estados de movimiento a ejecutar dependiendo del ID de cada motor. En la tabla 6-2 se aprecia cómo se toman los caracteres enviados de matlab y son convertidos para ser usados en el programa de la tarjeta.

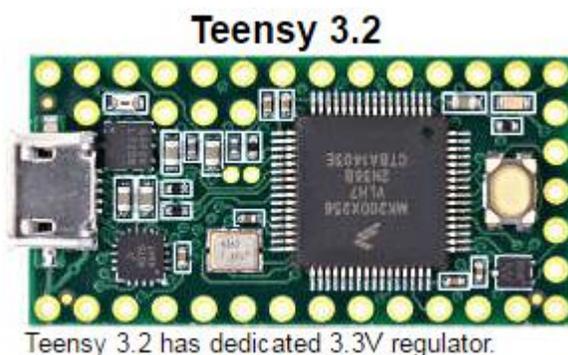


Figura 6-3. Teensy 3.2

Tomado de: (Worldwide, s.f.)

Caracter	5	4	3	2	1	0
	0	0	5	0	0	2
Trama de bit	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	0	0	5	0	0	2

Número de movimiento \rightarrow Tiempo de ejecución entre estados de movimiento \rightarrow

Tabla 6-2. Paso de caracteres a bits para ejecutar movimientos.

En el Apéndice 8-6 se presenta diagrama de flujo que presenta la programación realizada para el manejo de los módulos.

6.2 Implementación de movimientos

Una vez enviada la instrucción, este ejecuta los movimientos según el algoritmo realizado en Webots, en donde se analizarán aquellos que generan desplazamiento, en este caso los movimientos de avance y los desplazamientos laterales:

6.2.1 Avance hacia adelante:

Este movimiento recrea el comportamiento visto en la simulación, en donde el robot avanza levantando 3 patas y moviéndolas hacia adelante para repetir el mismo con las patas restantes. Comportamiento que se aprecia en la Figura 6-4.



Figura 6-4. Marcha hacia adelante del Robot Hexápodo modular MECABOT 4.0

6.2.1.1 Análisis del avance hacia adelante

Efectivamente el robot real se comportó de forma idéntica al hexápodo de la simulación, moviéndose a lo largo del laboratorio, en donde se realizaron varias pruebas cambiando la aceleración de los motores, el ángulo de avance y el tiempo en el que ejecutaba cada movimiento que cada unidad asignada equivale a 0,5 segundos, así para un intervalo de tiempo 4, en cada paso se esperan 2 segundos para el siguiente movimiento. La variación de estos parámetros determinar a qué velocidad avanzaba y para que parámetros le daban el mejor movimiento. A diferencia de la simulación, no se cuenta con un piso cuadriculado de 5x5 para conocer el avance, por lo que se elaboró un tapete con separaciones de 5cm marcadas para conocer cuánto se desplazó en determinado tiempo (datos de las pruebas en el Apéndice 8-3). De estas pruebas se obtuvo el siguiente gráfico (Figura 6-5):

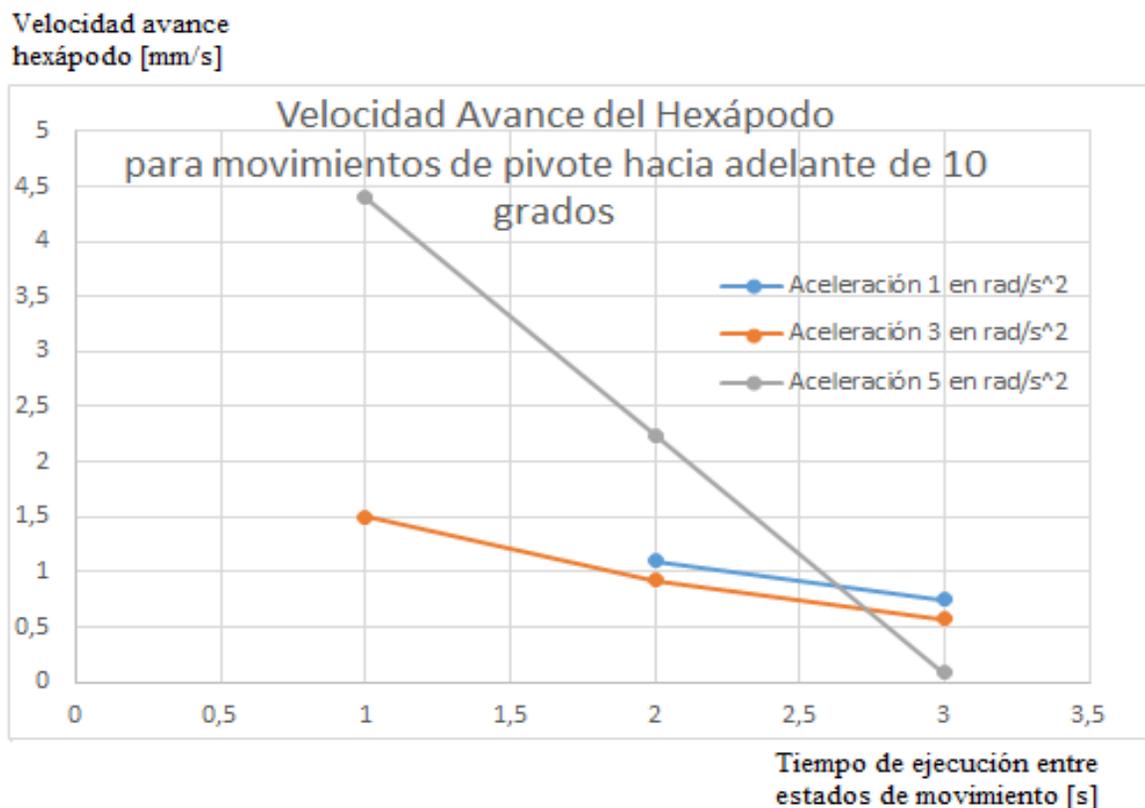


Figura 6-5. Gráfica velocidad avance del avance del hexápodo.

De las anteriores pruebas se concluyó que el robot modular hexápodo no puede realizar movimientos con aceleración 1 por debajo de 4 intervalos, que equivale a 2 segundos entre cada paso ya que los motores no alcanzan a llegar a la posición de la secuencia en ese intervalo de tiempo haciendo que se pierda y empiece a generar movimientos aleatorios. Inicialmente se trabajó con un ángulo de avance de 10 grados sin avanzar mucho cambiando a 30 grados para lograr mayores desplazamientos sin que las patas choquen entre sí. Este cambio de ángulo también sirvió para que el hexápodo avanzara de forma más lineal debido a que se resbala sobre el piso liso dándole movimientos de avance en diagonal, cosa que no ocurría tanto con el ángulo de 30 grados en el que se daban zancadas más largas. Estas pruebas concluyeron con una mejor locomoción para una aceleración de 5 con ángulos de 30 grados en intervalos de tiempo 4 para un movimiento más suave y constante sin perder velocidad, el cambio de velocidad se aprecia en la Figura 6-6. Los datos se encuentran tabulados en el Apéndice 8-10.

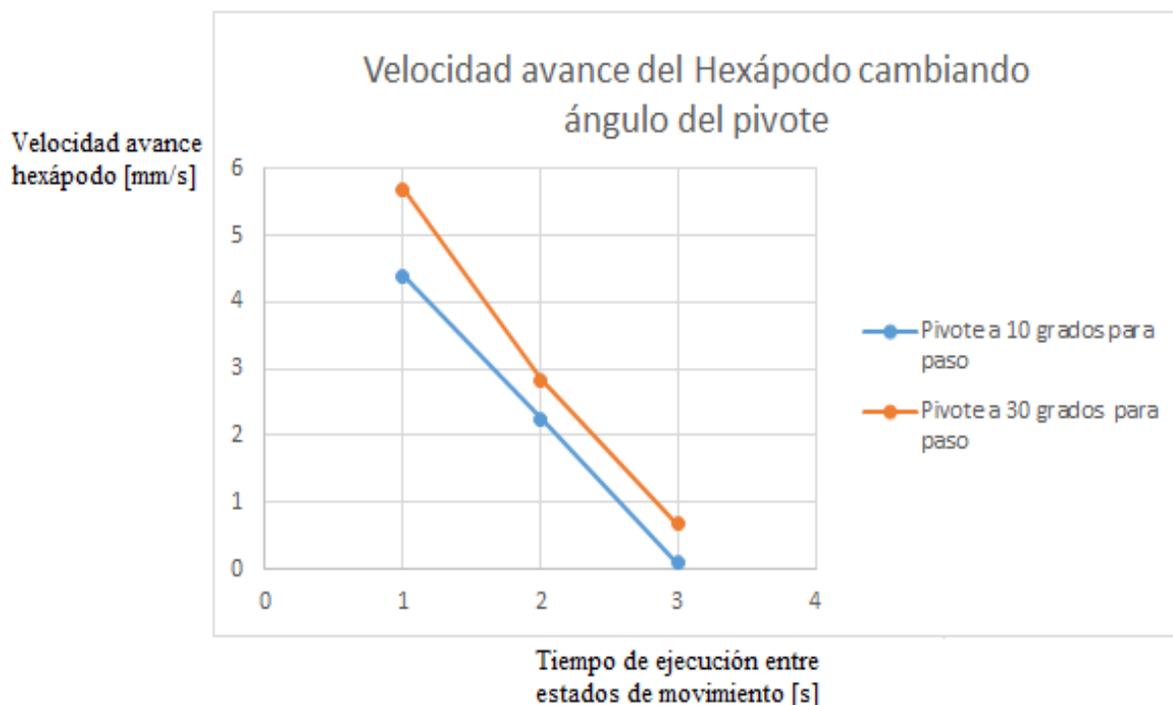


Figura 6-6. Grafica de avance del hexápodo cambiando el ángulo del pivote.

6.2.1.2 Avance del hexápodo en terrenos irregulares

También se realizaron pruebas sobre terreno rocoso y pasto (Figura 6-7), que a diferencia de las pruebas hechas en el laboratorio donde la superficie de trabajo era regular, siendo plano el piso en el que se desplazaba el robot durante toda su marcha, estos terrenos presentan hundimientos, desniveles, obstáculos y mayor fricción entre otros cambios, que permiten observar el comportamiento del hexápodo en terrenos irregulares. El resultado de estas pruebas reveló que el robot no pierde estabilidad y puede avanzar sin problemas, sin embargo debido a la poca altura del hexápodo este queda muy sobre el piso generando fricción con el pasto alto y en caso de las rocas se presenta un desgaste de las piezas por los continuo golpes sobre las rocas al no tener nada que lo amortigüe y al ser de plástico las caras se van dañando. Una de las observaciones importantes se da cuando los motores realizan mucho esfuerzo se dispara el pico de corriente y regulador que se implantó como medida de protección apaga el módulo momentáneamente haciendo que se reinicie el módulo a la espera de un comando. Esto resulta problemático cuando es un solo módulo el que falla debido a que entorpece el movimiento que continúan realizando los demás módulos del hexápodo.



Figura 6-7. Hexápodo Modular MECABOT 4.0 avanzando sobre terreno irregular pasto, rocoso y regular piso del laboratorio.

Para hacer una comparación del comportamiento del hexápodo en distintos terrenos, se tomó el piso del laboratorio como terreno regular al ser liso, y se buscó un pastizal y un terreno pedregoso como superficies irregulares, cuidando que estos no tuvieran mayor inclinación para que las pruebas fueran similares a las realizadas sobre la superficie plana del laboratorio. De las pruebas realizadas se estableció que la velocidad del hexápodo se ve afectada por el tipo de superficie, haciéndose más lento en los que ofrecen mayor fricción al contacto con las patas, así como los terrenos en los que el robot se hunde por su peso, como en la zona rocosa en la que cada paso removía varias piedras que entorpecían su marcha y forzaban más los motores de los módulos. Entre los terrenos de prueba, el pastizal es el que le da más trabajo al robot para moverse debido a la altura del pasto, y el piso liso del laboratorio es donde mejor se mueve al tener puntos de apoyo más estables, aunque presenta deslizamientos por el material de los módulos. Como se aprecia en la Figura 6-8, el tiempo de ejecución de 1 segundo permite que el hexápodo se desplace más rápido en todos los terrenos, que para una aceleración de 5 rad/s^2 en los motores y una tiempo de ejecución entre movimientos de 1 segundo, el robot se mueve a una velocidad de $5,69 \text{ mm/s}$ en la superficie plana del laboratorio siendo la velocidad más rápida de desplazamiento, seguido del terreno rocoso con $5,12 \text{ mm/s}$ y por último la velocidad lograda por el hexápodo en el pasto de $4,56 \text{ mm/s}$. La tabla de datos de la gráfica se encuentra en el Apéndice 8-11.

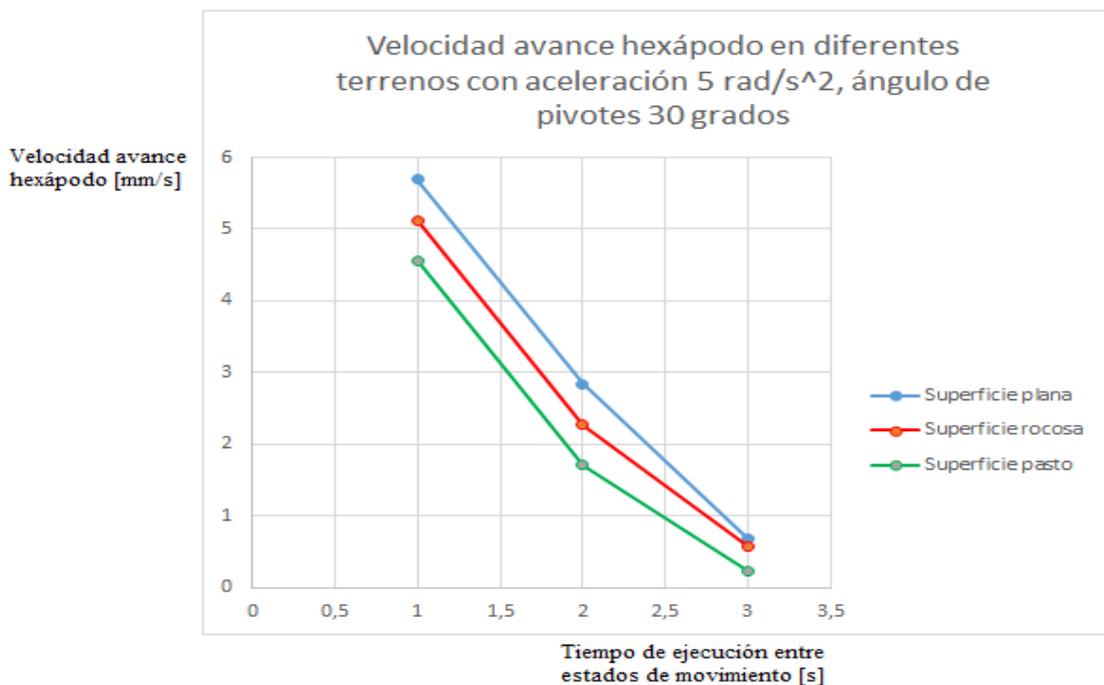


Figura 6-8. Gráfica velocidad de avance del hexápodo en distintos terrenos.

En la Figura 6-9, se muestra la gráfica del desplazamiento hacia adelante de la configuración modular hexápodo en distintos terrenos, utilizando una aceleración de 5 rad/s^2 , que es la mayor aceleración con la que el hexápodo se mueve correctamente, evidenciando el tiempo que le toma al robot desplazarse en terrenos plano, rocoso y con pasto. Como se aprecia en la gráfica, a la configuración le es más fácil moverse en un terreno regular, sin obstáculos o desniveles como es la superficie plana del laboratorio, tomándole cerca de 1,39 minutos desplazarse medio metro, siendo el terreno en el que se desplaza con mayor velocidad de los estudiados. Para terrenos irregulares como el rocoso y el pasto, el hexápodo encuentra mayor resistencia en sus movimientos, removiendo piedras al moverse deslizando el punto de apoyo de sus patas o al empujar el pasto alto restándole velocidad en el avance. La superficie más difícil es el pastizal, en donde el pasto ofrece una mayor resistencia a las patas del robot, para una velocidad de motores de 1 rad/s utilizadas en las prueba de la gráfica, el hexápodo tarda 1,82 minutos (109,6 s) en recorrer 0,5 metros, un poco más del tiempo que utiliza el robot en terreno rocoso, con 1,62 minutos para la misma distancia. La tabla de datos de las pruebas en distintos terrenos se aprecia en el Apéndice 8-12.

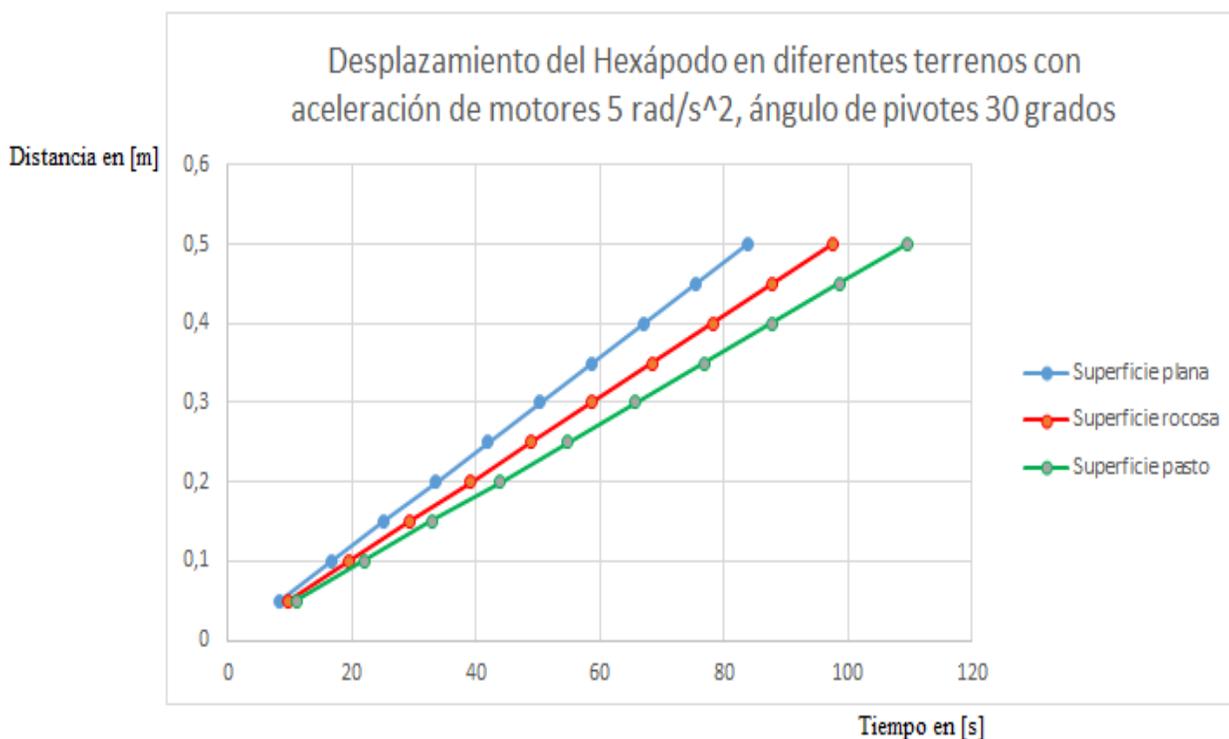


Figura 6-9. Desplazamiento del hexápodo en diferentes terrenos variando la velocidad de los motores.

6.2.1.2 Hexápodo movimiento hacia adelante físico frente a simulación.

El movimiento hacia adelante del hexápodo modular, es ejecutado en 5 movimientos que le permiten avanzar un paso, en donde la velocidad que alcanza el robot depende de los parámetros de velocidad y aceleración de los motores, así como el tiempo que se establece entre cada movimiento. El robot implementado se mueve de la misma forma que en la simulación para el programa desarrollado, con la diferencia que en la simulación se implementó un modelo simplificado que reduce considerablemente el peso del hexápodo, y aspectos como el material de los módulos, actuadores y que no se puede garantizar una simetría perfecta como en la simulación, afectan el desplazamiento del robot modular hexápodo. Como se aprecia en la gráfica de la Figura 6-10, en la que se usó una velocidad en los motores de 1 rad/s y se iba variando la aceleración de estos, las velocidades de desplazamiento del robot en simulación y físico terminan siendo opuestas. Debido al poco peso del modelo simulado, este se desliza sobre la superficie plana perdiendo apoyo en cada paso, deslizamiento que se incrementa a medida que aumenta la aceleración de los motores. Para una aceleración de 1rad/s², la configuración tenía una marcha constante, manteniendo una velocidad de desplazamiento de 2,22 mm/s, pero aumentado la aceleración este perdía velocidad hasta el punto que para una aceleración de 5 rad/s², este dejaba de avanzar y quedaba patinando en el mismo sitio. El caso contrario para el hexápodo físico, que por su peso tenía una velocidad menor que la simulación para una aceleración de motores de 1 rad/s², pero a medida que esta aumentaba también lo hacía la velocidad de avance del robot, llegando a 4,4 mm/s para una aceleración de 5 rad/s². La tabla comparativa de la gráfica se aprecia en el Apéndice 8-13.

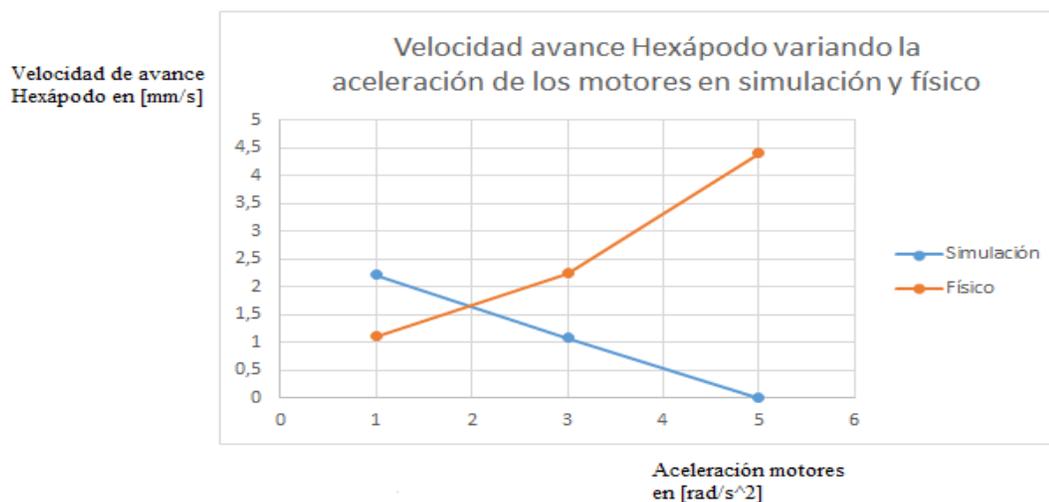


Figura 6-10. Gráfica velocidad hexápodo en simulación y físico.

6.2.2 Movimiento Lateral

El hexápodo logra desplazarse hacia los lados en realizando tres movimientos como se aprecia en la Figura 6-11.

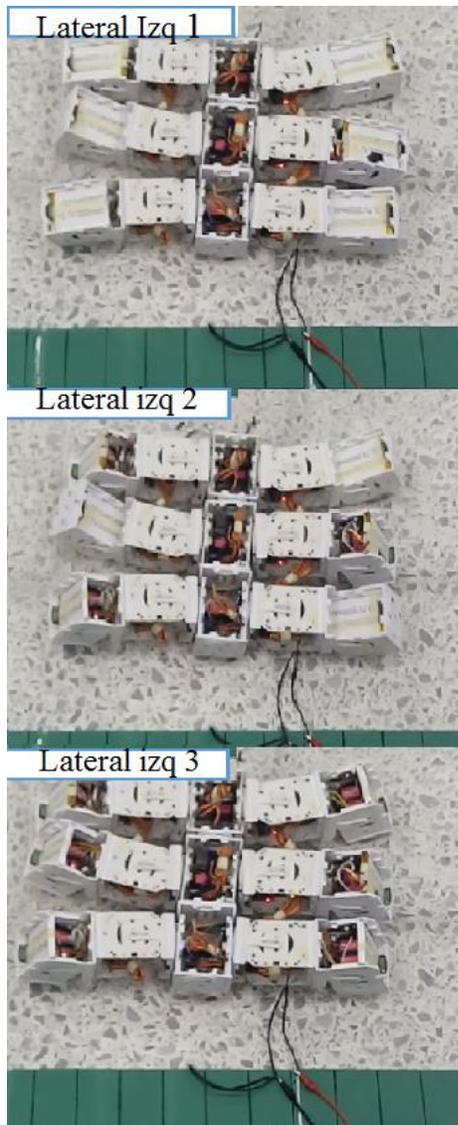


Figura 6-11. Marcha lateral izquierda del Robot Hexápodo modular MECABOT 4.0

6.2.2.1 Análisis del movimiento lateral del hexápodo.

Al igual que en la simulación se evidenció que el movimiento lateral de los módulos genera un mayor avance que el movimiento hacia adelante o atrás, por lo que para llegar más rápido de un punto a otro lo mejor sería ubicarse de manera que el hexápodo quede lateral al punto de interés, utilizando los movimientos rotacionales si el robot no se encuentra orientado. Al igual que en la implementación este es el movimiento con la mayor velocidad de desplazamiento. Aun se presentan deslizamientos cuando se sube la aceleración de los motores, pero debido al peso del sistema los cambios en la velocidad no son tan grandes, en la gráfica de la Figura 6-12 se aprecia la velocidad de avance lateral del hexápodo cambiando la aceleración de los motores. Los datos tomados se aprecian en la tabla del Apéndice 8-14.

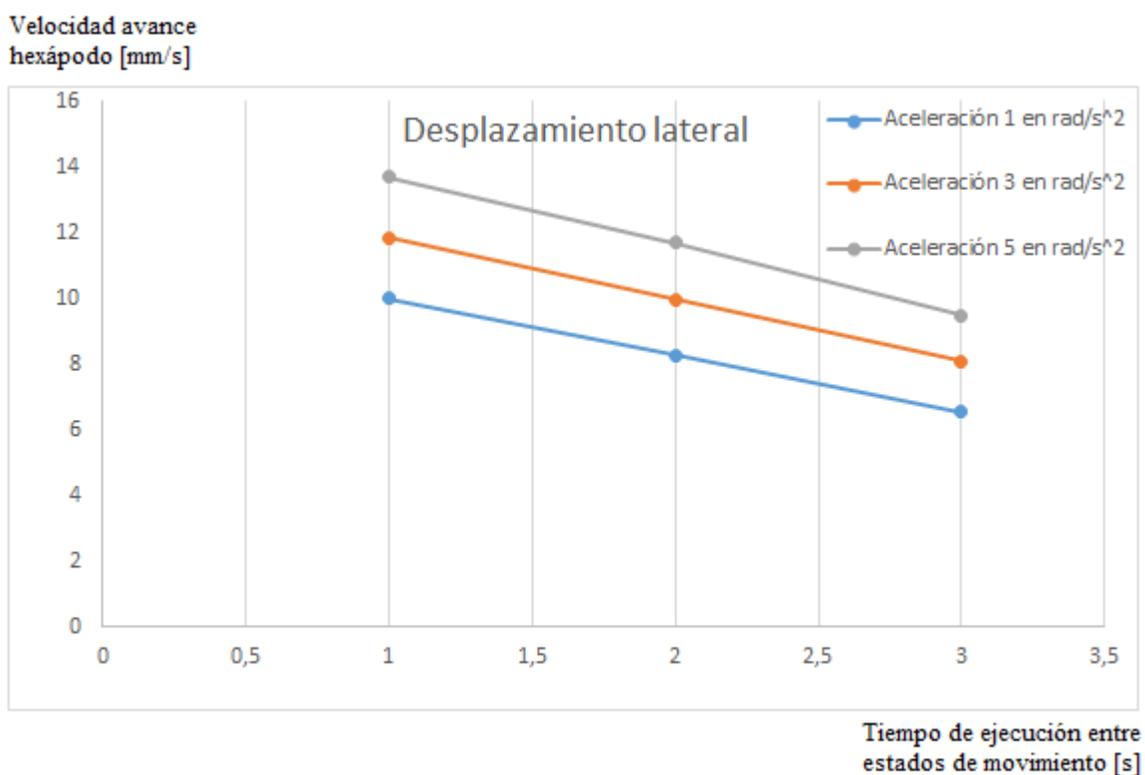


Figura 6-12. Gráfica de desplazamiento lateral.

En la Figura, se aprecia el tiempo que le toma a la configuración hexápodo en desplazamiento lateral recorrer determinadas distancias, variando la aceleración de los motores. Contrario a la simulación, utilizar movimientos más rápidos no ocasiona que el robot resbale con tanta facilidad, debido a que el peso del robot físico es mucho mayor que el simulado, compensado la poca fricción del material de los módulos que disminuía la distancia alcanzada en cada paso de la simulación. En la gráfica se observa que la configuración logra un mayor desplazamiento a una aceleración de motores de 5 rad/s^2 , tomándole cerca de 36,4 segundos recorrer una distancia de medio metro, seguido del tiempo empleado de 42,19 s que le tomo al hexápodo desplazarse la misma distancia utilizando motores con aceleración de 3 rad/s^2 , y para una aceleración de motores de 1 rad/s^2 , le toma alrededor de 50 segundos moverse 0,5 metros, siendo el valor de aceleración con el que el robot modular va más lento. Los valores de la Figura 6-13, se encuentran en la tabla del Apéndice 8-15.

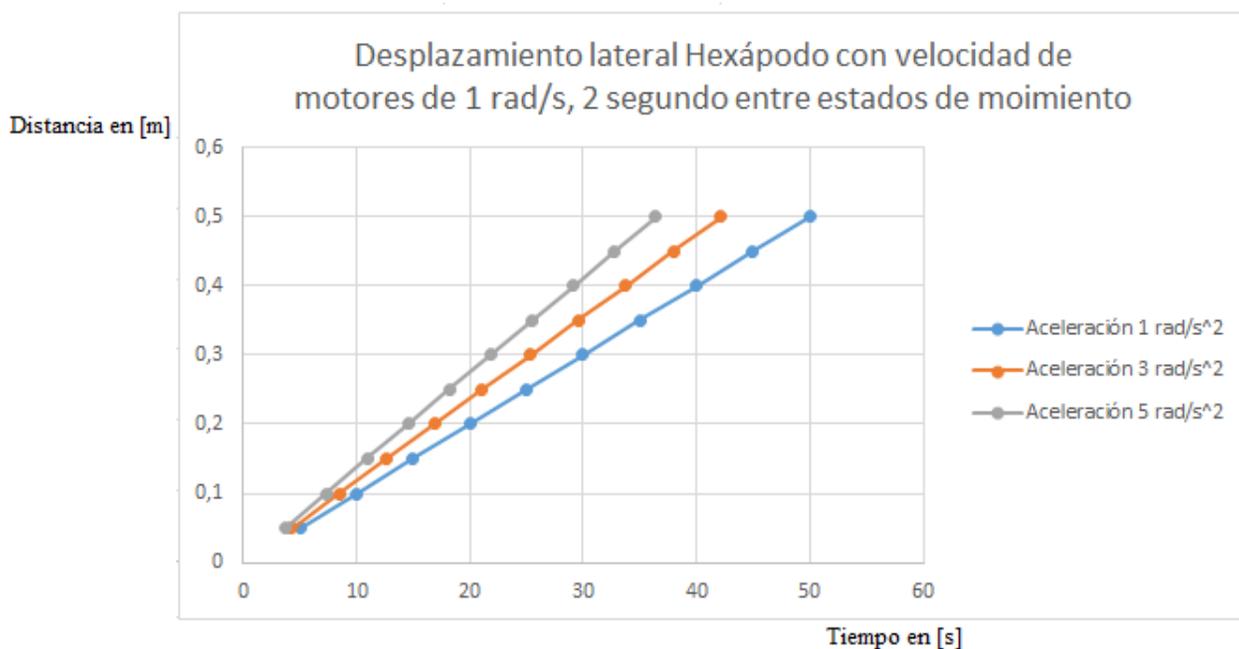


Figura 6-13. Desplazamiento del robot Hexápodo cambiando la aceleración de los motores.

6.2.3 Movimiento Rotacional

El movimiento rotacional del hexápodo en ambos sentidos, se ejecuta de la misma forma que en la simulación, permitiendo al robot cambiar su sentido de orientación (Figura 6-14). Por su peso la configuración física es mucha más lenta que la vista en la simulación, tomado un tiempo de 6,4 minutos (384,57 s) en girar 90 grados para aceleración de motores de 1 rad/s^2 , mucho más tiempo que en la simulación la cual realizó el movimiento en 4,4 minutos para la misma aceleración. El movimiento rotacional, no presenta un gran cambio al aumentar la velocidad de los motores, como se aprecia en la gráfica de la Figura 6-15, girar 90 grados le toma al robot 6,3 minutos con los motores a una velocidad de 8 rad/s , un poco menos de lo que le toma a 1 rad/s .



Figura 6-14. Hexápodo modular girando a 45 y 90 grados.

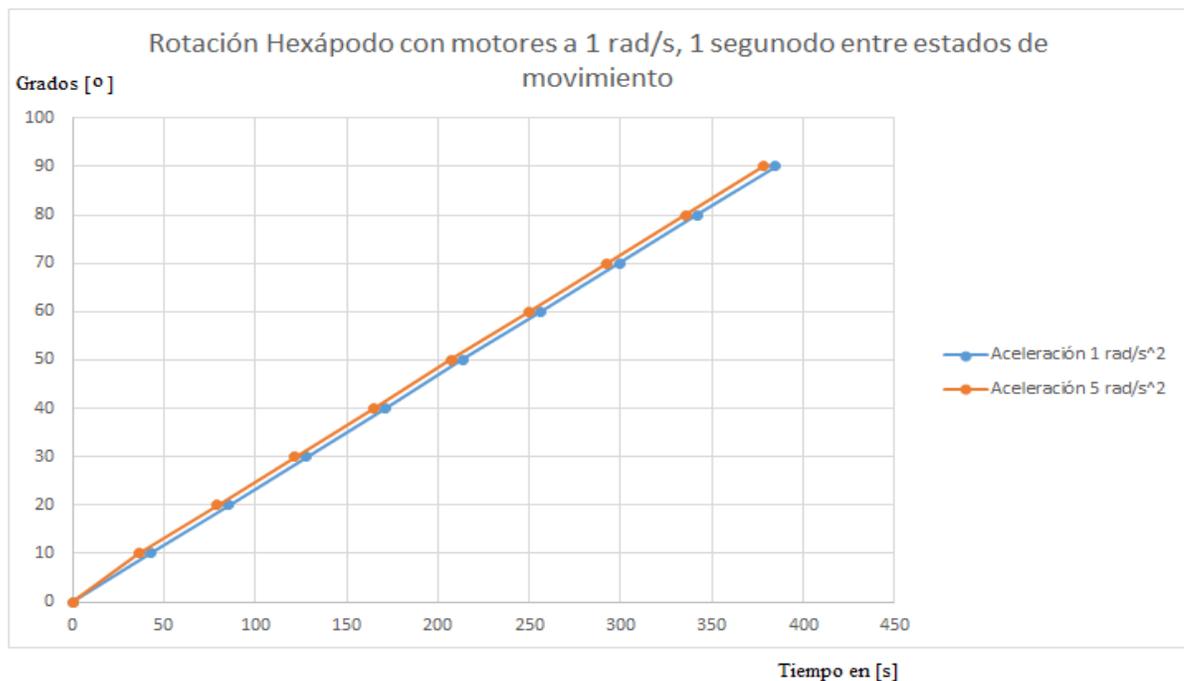


Figura 6-15. Gráfica rotación hexápodo físico. (Tabla de datos Apéndice 8-16)

7. Conclusiones y recomendaciones a trabajos futuros

Durante el trabajo se desarrolló un sistema robótico modular funcional, en donde se lograron generar movimientos para una configuración tipo hexápodo, implementada con base a una integración de módulos, cuya locomoción se programa utilizando los movimientos previamente configurados y simulados en Webots. Este modelo junto con los propios módulos MECABOT 4.0, fueron puestos a prueba analizando sus características y capacidades motoras, determinando las ventajas del hexápodo y aspectos por mejorar de los propios módulos.

Se realizó el modelo matemático de una pata del hexápodo teniendo en cuenta que eran iguales para todo el hexápodo. Se determinó con ayuda del análisis cinemático que la forma más sencilla de generar avances era cambiando el sentido de orientación de los semimódulos acoplados, delegándoles a cada uno locomoción específica que en conjunto lograban imitar los movimientos de los insectos y las ventajas que tiene su forma de caminar.

Se logró simular la configuración hexápodo junto con sus movimiento, verificando las capacidades del simulador Webots al utilizar las funciones programadas en este, se lograron los movimientos tanto físico y simulados de todas las marchas del hexápodo sin mayores cambios en el comportamiento físico con el virtual.

Se logró diseñar un nuevo módulo más compacto sin perder las capacidades del módulo MECABOT 3.0, siendo un modelo auto configurable y con su propia fuente de energía. Se construyeron varios de estos para ser utilizados en nuevas configuraciones que le den más alcance al trabajo de investigación en robótica de la Universidad Militar Nueva Granada.

Se implementó una configuración robótica modular utilizando los módulos MECABOT, con movimientos omnidireccionales capaces de generar desplazamientos, los cuales fueron evaluados cambiando parámetros en los motores y probando la locomoción del hexápodo sobre superficie plana y terrenos irregulares pasto y rocoso.

Algunas a consideraciones para futuros trabajos observadas durante el desarrollo del trabajo, se dan principalmente en los mismos módulos. Las configuraciones con los módulos MECABOT tienden a deslizarse sobre superficies lisas al no estar hechas de un material que le dé más fricción al suelo, se recomienda utilizar un polímero con mayor adhesión o adicionarle topes

de goma a las esquinas de las cara para darle mayor agarre sobre la superficie. Se recomienda cambiar el regulador de corriente por uno de mayor rango que soporte los picos de corriente generados cuando los motores hacen mucho esfuerzo.

En la configuración hexápodo no se garantiza geometría perfecta debido a los actuadores y peso, por lo que para futuros rediseños buscar materiales más livianos y mejorar las técnicas de ensamble para que las articulaciones se ajusten a los modelos simulados.

Como trabajos futuros se piensa en la comunicación entre módulos, equipar a estos con sensores que les permitan reconocer a otros módulos y a su entorno para realizar trabajos cooperativos y formen las configuraciones de forma autónoma.

Referencias

- 2014 Cyberbotics Ltd. (2014). *Webots User Guide release 7.4.3*. www.cyberbotics.com: Cyberbotics©.
- Arredondo, R. (2006). *Design and simulation of locomotion of self-organising modular robots for adaptive furniture. Master's thesis*. Swiss Federal Institute of Technology Lausanne: Biologically Inspired Robotics Group.
- Cárdenas Verdugo, M. A. (2011). *DISEÑO, CONSTRUCCIÓN Y CONTROL DE UN ROBOT HEXÁPODO*. Ciudad Universitaria, Mexico D. F: UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO.
- Chih-Han, Y., & Radhika, N. (2009). *Engineering Self-Adaptive Modular Robotics: A Bio-Inspired Approach*. Cambridge, USA: The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems.
- Choque Aspiazu, G. (2009). *Robótica modular reconfigurable*. LA PAZ-BOLIVIA: www.eldiario.net.
- D. Hancher, M., & S. Hornby, G. (2006). *A Modular Robotic System with Applications to Space Exploration*. California: 2nd IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT'06).
- D. Hancher, M., & S. Hornby, G. (2007). *Evolving Quadruped Gaits with a Heterogeneous Modular Robotic System*. NASA Ames Research Center, University of California Santa Cruz: 2007 IEEE Congress on Evolutionary Computation (CEC 2007).
- De León Melgarejo, L. A. (2010). *Generación de patrones de marcha de un hexápodo por CPG's mediante autómatas de Mealy*. Mexico D.F: INSTITUTO POLITÉCNICO NACIONAL.
- Ekawahyu, S., Jianing, L., Alvarado Rayo, Y., Peck, A. M., Montenegro, J., Gonyea, M., & Valdastrì, P. (2016). *An Affordable Modular Robotic Kit for Integrated Science, Technology Engineering, and Math Education*. IEEE ROBOTICS & AUTOMATION MAGAZINE.
- Electronics, P. R. (n.d.). *Pololu Robotic & Electronics*. Retrieved from <https://www.pololu.com/product/2858>
- Escobar Olarte, J. M., & Silva Ortega, J. D. (2011). *DISEÑO DE SISTEMA ELECTRÓNICO JERÁRQUICO MEDIANTE COMANDOS PARA ROBOT HEXÁPODO*. Bogotá D.C,

Colombia: PONTIFICIA UNIVERSIDAD JAVERIANA .FACULTAD DE INGENIERIA.

Feczko, J., Manka, M., Krol, P., Giergiel, M., Uhl, T., & Pietrzyk, A. (2015). *Review of the modular self reconfigurable robotic systems*. Poznan,Polonia: Proceedings of the 10th International Workshop on Robot Motion and Control, Poznan University of Technology.

González Gómez, J. (2008). *ROBÓTICA MODULAR Y LOCOMOCIÓN: APLICACIÓN Y ROBOTS ÁPODOS*. Madrid, España: UNIVERSIDAD AUTÓNOMA DE MADRID ESCUELA POLITÉCNICA SUPERIOR.

González Gómez, J. (2009). *Robótica Modula y locomoción* . Albacete-España: Robótica Móvil: Una apuesta de futuro. Curso de Verano.

Hurtado Herasco, C. A., & Rubiano Montaña, O. G. (2013). *Diseño y simulación de un robot modular reconfigurable*. Bogotá, Colombia: Universidad Militar Nueva Granada.

Jie, Z., Xindan, C., Yanhe, Z., & Shufeng, T. (2011). *A New Self-Reconfigurable Modular Robotic System UBot: Multi-mode locomotion and Self-reconfiguration*. Shanghai,China: 2011 IEEE International Conference on Robotics and Automation Shanghai International Conference Center.

Kamimura, A., Kurokawa, H., Yoshida, E., Murata, S., Kokaji, S., & Tomita, K. (2005). *DISEÑO AUTOMATICO DE LOCOMOCIÓN Y EXPERIMENTOS PARA UN SISTEMA ROBÓTICO MODULAR*. IEEE/ASME TRANSACTIONS ON MECHATRONICS, VOL. 10.

Lancheros Guzmán, P. N., & Sanabria Galvis, L. B. (2016). *Simulación e Implementación de Movimientos para Sistema Robótico Modular Considerando Diferentes Configuraciones*. Bogotá-Colombia: UNIVERSIDAD MILITAR NUEVA GRANADA.

Limitado, P. P. (n.d.). *Tdrobotica.co*. Retrieved from <http://tdrobotica.co/servo-mini-digital-giro-limitado-hd-1810mg-39-kg-cm/523.html>

Rai, V., Correll, N., & Anne van Rossum. (2011). *Self-Assembly of Modular Robots from Finite Number of Modules Using Graph Grammars*. San Francisco, CA, USA: IEEE/RSJ International Conference on Intelligent Robots and Systems.

Rodríguez Calderón, J. P., Ramos Parra, M. F., & Peña Giraldo, M. V. (2015). *Simulación de un Robot Hexápodo Bioinspirado en el Tenebrio*. Bogotá, Colombia: Universidad Libre.

- Russo, S., Harada, K., Ranzani, T., Manfredi, L., Stefanini, C., Menciassi, A., & Dario, P. (2013). *Design of a Robotic Module for Autonomous Exploration and Multimode Locomotion*. IEEE/ASME TRANSACTIONS ON MECHATRONICS, VOL. 18, NO. 6, DECEMBER 2013.
- Sedeño Bustos, E. (2011). *Locomoción de un Robot Cuadrúpedo: Un Enfoque a Celdas Neuronales Analógicas*. Cuernavaca, Morelos, México: Centro Nacional de Investigación y Desarrollo Tecnológico.
- Van Tung, L., & Trung Dung, N. (2014). *Virtual Pheromone based Information Foraging in Modular Robotics*. Kuala Lumpur, Malaysia: The 11th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI 2014).
- Vonásek, V., Neumann, S., Oertel, D., & Worn, H. (2015). *Online Motion Planning for Failure Recovery of Modular Robotic Systems*. Washington, U.S.A: 2015 IEEE International Conference on Robotics and Automation (ICRA) Washington State Convention Center Seattle .
- Weng, J. (2012). *Creación del Modelo del Robot Humanoide Teo para el Simulador WEBOTS*. Madrid, España: UNIVERSIDAD CARLOS III DE MADRID ESCUELA POLITECNICA SUPERIOR.
- Weng, J. (2012). *CREACIÓN DEL MODELO DEL ROBOT HUMANOIDE TEO PARA EL SIMULADOR WEBOTS*. Madrid, España: UNIVERSIDAD CARLOS III DE MADRID ESCUELA POLITÉCNICA SUPERIOR.
- Worldwide, P. E. (n.d.). Retrieved from <https://www.pjrc.com/teensy/teensy31.html#specs>
- Yim, M., Duff, D., & Roufas, G. (2000). *Polybot: a Modular Reconfigurable Robot*. San Francisco, U.S.A: International Conference on Robotics & Automation, San Francisco: IEEE.
- Zhang, H., Gonzales Gomez, j., Chen, S. Y., & Zhang, W. (2008). *Embedded Intelligent Capability of a Modular Robotic System*. Bangkok, Thailand: IEEE, International Conference on Robotics and Biomimetics.
- Zykov, V., Mytilinaios, E., Desnoyer, M., & Lipson, H. (2007). *IEEE TRANSACTIONS ON ROBOTICS, VOL. 23, NO. 2, . IEEE*.

8. Anexos

Apéndice 8-1. Tabla de estados de movimientos:

Estados de movimientos del robot modular hexápodo	Motores a rotar de cada semimódulo en cada estado de movimiento	Rotación del motor pivote de cada semimódulo en Radianes	Rotación del motor pivote de cada semimódulo en Grados
INICIO	Motor Pivote 1	-1.22173	-70
	Motor Pivote 3	-1.22173	-70
	Motor Pivote 5	-1.22173	-70
	Motor Pivote 7	-1.22173	-70
	Motor Pivote 9	-1.22173	-70
	Motor Pivote 11	-1.22173	-70
Adelante 1	Motor Pivote 1	0	0
	Motor Pivote 5	0	0
	Motor Pivote 9	0	0
	Motor Pivote 2	0.1745	10
	Motor Pivote 6	0.1745	10
	Motor Pivote 10	0.1745	10
Adelante 2	Motor Pivote 1	-1.22173	-70
	Motor Pivote 5	-1.22173	-70
	Motor Pivote 9	-1.22173	-70
	Motor Pivote 3	0	0
	Motor Pivote 7	0	0
	Motor Pivote 11	0	0

Adelante 3	Motor Pivote 4	0.1745	10
	Motor Pivote 8	-0.1745	10
	Motor Pivote 12	0.1745	10
Adelante 4	Motor Pivote 3	-1.22173	-70
	Motor Pivote 7	-1.22173	-70
	Motor Pivote 11	-1.22173	-70
Adelante 5	Motor pivote 2	0	0
	Motor pivote 6	0	0
	Motor pivote 10	0	0
	Motor pivote 4	0	0
	Motor pivote 8	0	0
	Motor pivote 12	0	0
Lateralder 1	Motor pivote 1	0	0
	Motor pivote 5	0	0
	Motor pivote 9	0	0
	Motor pivote 3	-0.349	-20
	Motor pivote 7	-0.349	-20
	Motor pivote 11	-0.349	-20
Lateralder 2	Motor pivote 3	-1.22173	-70
	Motor pivote 7	-1.22173	-70
	Motor pivote 11	-1.22173	-70
Lateralder3	Motor pivote 1	-1.22173	-70
	Motor pivote 5	-1.22173	-70
	Motor pivote 9	-1.22173	-70
Cero	Motor pivote 1	0	0

	Motor pivote 3	0	0
	Motor pivote 5	0	0
	Motor pivote 7	0	0
	Motor pivote 9	0	0
	Motor pivote 11	0	0
Atras1	Motor Pivote 1	0	0
	Motor Pivote 5	0	0
	Motor Pivote 9	0	0
	Motor Pivote 2	-0.1745	-10
	Motor Pivote 6	-0.1745	-10
	Motor Pivote 10	-0.1745	-10
Atras2	Motor Pivote 1	-1.22173	-70
	Motor Pivote 5	-1.22173	-70
	Motor Pivote 9	-1.22173	-70
Atras3	Motor Pivote 3	0	0
	Motor Pivote 7	0	
	Motor Pivote 11	0	0
	Motor Pivote 4	-0.1745	-10
	Motor Pivote 8	-0.1745	-10
	Motor Pivote 12	-0.1745	-10
Atras4	Motor Pivote 3	-1.57	-70
	Motor Pivote 7	-1.57	-70
	Motor Pivote 11	-1.57	-70
Atras5	Motor Pivote 2	0	0
	Motor Pivote 6	0	0
	Motor Pivote 10	0	0

	Motor Pivote 4	0	0
	Motor Pivote 8	0	0
	Motor Pivote 12	0	0
Lateralizq1	Motor Pivote 3	0	0
	Motor Pivote 7	0	0
	Motor Pivote 11	0	0
	Motor Pivote 1	-0.349	-20
	Motor Pivote 5	-0.349	-20
	Motor Pivote 9	-0.349	-20
Lateralizq2			
	Motor Pivote 1	-1.22173	-70
	Motor Pivote 5	-1.22173	-70
	Motor Pivote 9	-1.22173	-70
Lateralizq3	Motor Pivote 3	-1.22173	-70
	Motor Pivote 7	-1.22173	-70
	Motor Pivote 11	-1.22173	-70
Rotarder1	Motor pivote 1	0	0
	Motor pivote 5	0	0
	Motor pivote 9	0	0
	Motor pivote 2	-0.1745	-10
	Motor pivote 6	-0.1745	-10
	Motor pivote 10	-0.1745	10
Rotarder2	Motor pivote 1	-1.22173	-70
	Motor pivote 5	-1.22173	-70
	Motor pivote 9	-1.22173	-70

Rotarder3	Motor pivote 3	0	0
	Motor pivote 7	0	0
	Motor pivote 11	0	0
	Motor pivote 4	-0.1745	-10
	Motor pivote 8	-0.1745	-10
	Motor pivote 12	0.1745	10
Rotarder4	Motor pivote 3	-1.22173	-70
	Motor pivote 7	-1.22173	-70
	Motor pivote 11	-1.22173	-70
Rotarder5			
	Motor pivote 2	0	0
	Motor pivote 6	0	0
	Motor pivote 10	0	0
	Motor pivote 4	0	0
	Motor pivote 8	0	0
	Motor pivote 12	0	0
Rotarizq1	Motor pivote 1	0	0
	Motor pivote 5	0	0
	Motor pivote 9	0	0
	Motor pivote 2	0.1745	10
	Motor pivote 6	0.1745	10
	Motor pivote 10	-0.1745	-10
Rotarizq2	Motor pivote 1	-1.22173	-70
	Motor pivote 5	-1.22173	-70

	Motor pivote 9	-1.22173	-70
Rotarizq3	Motor pivote 3	0	0
	Motor pivote 7	0	0
	Motor pivote 11	0	0
	Motor pivote 4	0.1745	10
	Motor pivote 8	0.1745	10
	Motor pivote 12	-0.1745	-10
Rotarizq4	Motor pivote 3	-1.22173	-70
	Motor pivote 7	-1.22173	-70
	Motor pivote 11	-1.22173	-70
Rotarizq5			
	Motor pivote 2	0	0
	Motor pivote 6	0	0
	Motor pivote 10	0	0
	Motor pivote 4	0	0
	Motor pivote 8	0	0
	Motor pivote 12	0	0

Apéndice 8-2 Tabla recolección pruebas de simulación Movimiento hacia adelante

Aceleración de los motores rad/s ² de los módulos del hexápodo	Velocidad Motores de los módulos en rad/s	Velocidad de avance del hexápodo mm/s
-1	1	2,27
-1	2	2,3
-1	4	2,5
-1	6	2,9
-1	8	3,3
1	1	2,22
1	2	2,25
1	4	2,31
1	6	2,37
1	8	2,43
3	1	1,08
3	2	1,05
3	4	0,99

Apéndice 8-3 Tabla recolección pruebas de implementación movimiento hacia adelante

Aceleración de los motores rad/s ² de los módulos del hexápodo	Interrupciones entre cada estado de movimiento en s	Grados en los que se mueve el pivote 1	Velocidad de avance del hexápodo mm/s
1	2	10	1,1
3	1	10	1,5
3	2	10	0,92
3	3	10	0,57
5	1	10	4,4
5	2	10	2,243
5	3	10	0,645
5	1	30	5,696
5	2	30	2,845
5	3	30	0,683

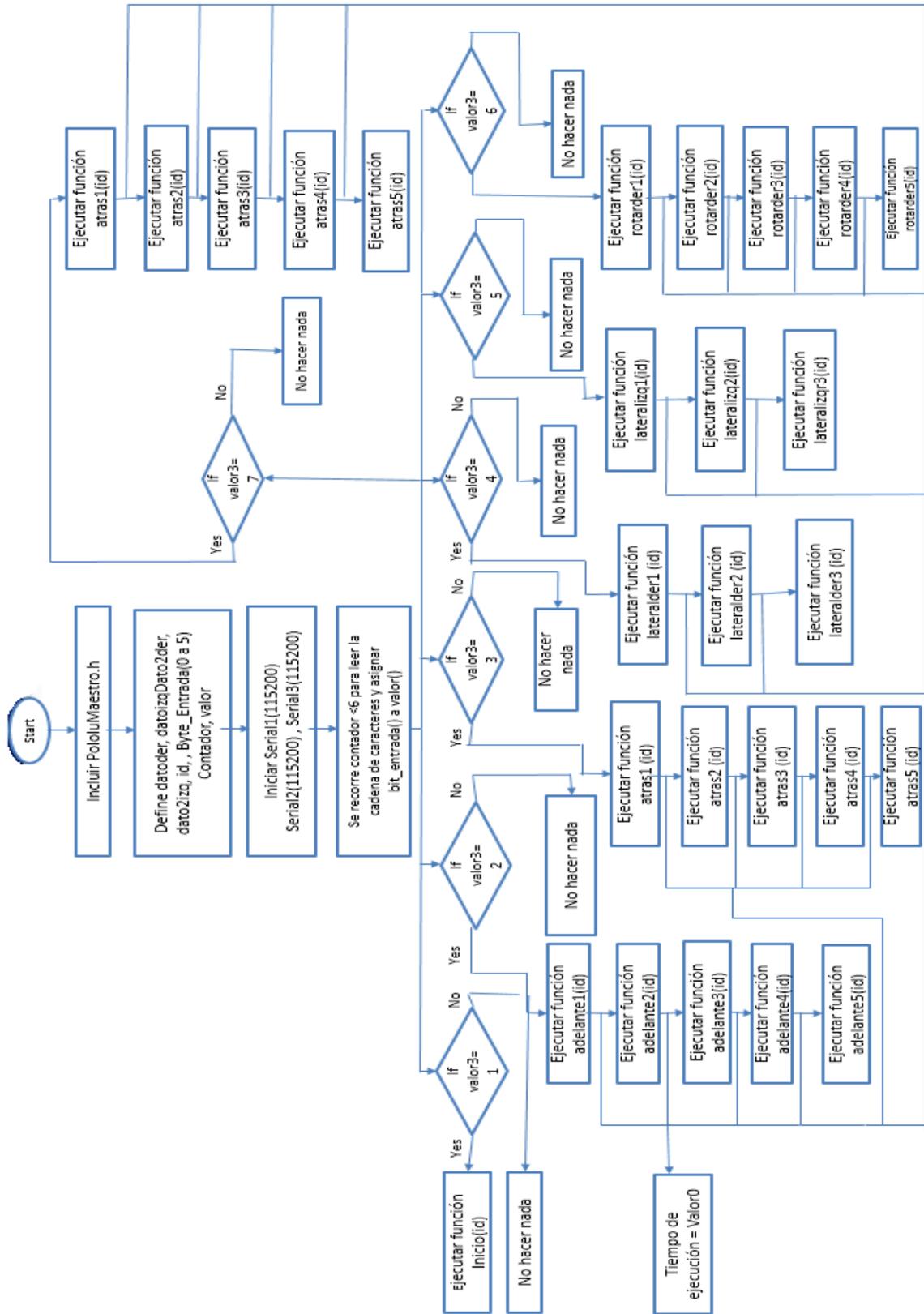
Apéndice 8-4 Tabla recolección pruebas de simulación movimiento lateral

Aceleración de los motores rad/s ² de los módulos del hexápodo	Velocidad Motores de los módulos en rad/s	Velocidad de desplazamiento lateral del hexápodo mm/s
-1	1	20
-1	2	16
-1	4	14,28
-1	6	11,11
1	1	12,5
1	2	10.63
1	4	8,76
1	6	6,89
3	1	15,62
3	2	9,68
3	4	2,2

Apéndice 8-5 Tabla recolección pruebas de implementación movimiento lateral

Aceleración de los motores rad/s ² de los módulos del hexápodo	Interrupciones entre cada estado del movimiento lateral en s	Velocidad de avance del hexápodo mm/s
1	1	10
1	2	8,28
1	3	6,56
3	1	11,85
3	2	9,98
3	3	8,1
5	1	13,7
5	2	11,7
5	3	9,5

Apéndice 8-6 Diagrama de flujo algoritmo de programación para comando de módulo



Apéndice 8-7. Tabla recolección de datos para la gráfica 4-14, desplazamiento robot hexápodo.

Velocidad motores en rad/s	Tiempo en s	Distancia en m
1	20,986	0,05
1	41,792	0,1
1	62,688	0,15
1	83,584	0,2
1	104,48	0,25
1	125,376	0,3
1	146,272	0,35
1	167,168	0,4
1	188,064	0,45
1	208,96	0,5
4	19,6	0,05
4	39,2	0,1
4	58,8	0,15
4	78,4	0,2
4	98	0,25
4	117,6	0,3
4	137,2	0,35
4	156,8	0,4
4	176,4	0,45
4	196	0,5
6	16,86	0,05
6	33,72	0,1
6	50,58	0,15
6	67,44	0,2
6	84,3	0,25
6	101,16	0,3
6	118,02	0,35
6	134,88	0,4
6	151,74	0,45
6	168,6	0,5
8	14,8	0,05
8	29,6	0,1
8	44,4	0,15

8	59,2	0,2
8	74	0,25
8	88,8	0,3
8	103,6	0,35
8	118,4	0,4
8	133,2	0,45
8	148	0,5

Apéndice 8-8. Tabla de datos figura 4-19 desplazamiento lateral variando la velocidad de los motores.

Velocidad motores en rad/s	Tiempo en s	Distancia en m
1	2	0,05
1	4	0,1
1	6	0,15
1	8	0,2
1	10	0,25
1	12	0,3
1	14	0,35
1	16	0,4
1	18	0,45
1	20	0,5
2	3,125	0,05
2	6,25	0,1
2	9,375	0,15
2	12,5	0,2
2	15,625	0,25
2	18,75	0,3
2	21,875	0,35
2	25	0,4
2	28,125	0,45
2	31,25	0,5
4	3,5014	0,05
4	7,0028	0,1
4	10,5042	0,15
4	14,0056	0,2
4	17,507	0,25
4	21,0084	0,3

4	24,5098	0,35
4	28,0112	0,4
4	31,5126	0,45
4	35,014	0,5
6	4,5	0,05
6	9	0,1
6	13,5	0,15
6	18	0,2
6	22,5	0,25
6	27	0,3
6	31,5	0,35
6	36	0,4
6	40,5	0,45
6	45	0,5

Apéndice 8-9. Tabla de datos grafica figura-23, rotación hexápodo en simulación.

Velocidad motores en rad/s	Tiempo en s	Grados °
1	0	0
1	10	10
1	32	20
1	48	30
1	64	40
1	80	50
1	96	60
1	112	70
1	128	80
1	144	90
4	0	0
4	14,7	10
4	30,7	20
4	46,7	30
4	62,7	40
4	78,7	50
4	94,7	60
4	110,7	70
4	126,7	80
4	142,7	90

8	0	0
8	9,9	10
8	25,9	20
8	41,9	30
8	57,9	40
8	73,9	50
8	89,9	60
8	105,9	70
8	121,9	80
8	137,9	90

Apéndice 8-10. Tabla de la figura 6-6, grafica avance del hexápodo cambiando el ángulo del pivote.

Grados con los que avanza el pivote	Tiempo de interrupción en s	Velocidad de avance en mm/s
10	1	4,4
10	2	2,243
10	3	0,086
30	1	5,69
30	2	2,84
30	3	0,683

Apéndice 8-11. Tabla de datos de la Figura 6-8, velocidad de avance hexápodo en distintos terrenos.

Terreno de prueba	Tiempo de interrupción en s	Velocidad de avance en mm/s
Plano	1	5,69
plano	2	2,84
Plano	3	0,683
Rocoso	1	4,56
Rocoso	2	1,71
Rocoso	3	0,23
Pasto	1	5,18
Pasto	2	2,27
Pasto	3	0,58

Apéndice 8-12. Tabla de la figura 6-9, desplazamiento hexápodo en diferentes terrenos variando la velocidad de los motores.

Superficie de prueba	Tiempo en s	Distancia en m
plano	8,38926	0,05
plano	16,77852	0,1
plano	25,16778	0,15
plano	33,55704	0,2
plano	41,9463	0,25
plano	50,33556	0,3
plano	58,72482	0,35
plano	67,11408	0,4
plano	75,50334	0,45
plano	83,8926	0,5
rocoso	9,7656	0,05
rocoso	19,5312	0,1
rocoso	29,2968	0,15
rocoso	39,0624	0,2
rocoso	48,828	0,25
rocoso	58,5936	0,3
rocoso	68,3592	0,35
rocoso	78,1248	0,4
rocoso	87,8904	0,45
rocos	97,656	0,5
pasto	10,9649	0,05
pasto	21,9298	0,1
pasto	32,8947	0,15
pasto	43,8596	0,2
pasto	54,8245	0,25
pasto	65,7894	0,3
pasto	76,7543	0,35
pasto	87,7192	0,4
pasto	98,6841	0,45
pasto	109,649	0,5

Apéndice 8-13. Tabla figura 6-10, grafica velocidad hexápoda en simulación y físico.

Hexápodo modular en	Aceleración de motores rad/s ²	Velocidad hexápodo en mm/s
simulación	1	2,22
simulación	3	1,08
simulación	5	0
físico	1	1,1
físico	3	2,24
físico	5	4,4

Apéndice 8-14. Tabla de datos la figura 6-13, desplazamiento del robot hexápodo cambiando la aceleración de los motores.

Aceleración motores en rad/s ²	Tiempo entre estados de movimiento en s	Velocidad hexápodo en mm/s
1	1	10
1	2	8,28
1	3	6,56
3	1	11,86
3	2	9,98
3	3	8,1
5	1	13,7
5	2	11,7
5	3	9,5

Apéndice 8-15. Tabla de datos figura 6-13, desplazamiento hexápodo cambiando la aceleración de los motores.

Aceleración de motores en rad/s ²	Tiempo en s	Distancia en m
1	5	0,05
1	10	0,1
1	15	0,15
1	20	0,2
1	25	0,25
1	30	0,3
1	35	0,35
1	40	0,4
1	45	0,45
1	50	0,5
3	4,2194	0,05
3	8,4388	0,1
3	12,6582	0,15
3	16,7876	0,2
3	21,097	0,25
3	25,3164	0,3
3	29,5358	0,35
3	33,7552	0,4
3	37,9746	0,45
3	42,194	0,5
5	3,64	0,05
5	7,28	0,1
5	10,92	0,15
5	14,56	0,2
5	18,2	0,25
5	21,84	0,3
5	25,48	0,35
5	29,12	0,4
5	32,76	0,45
5	36,4	0,5

Apéndice 8-16. Tabla de datos figura 6-15, rotación del hexápodo físico.

Aceleración motores en rad/s ²	Tiempo en s	Grados °
1	0	0
1	42,73	10
1	85,46	20
1	128,49	30
1	170,92	40
1	213,65	50
1	256,38	60
1	299,11	70
1	341,84	80
1	384,57	90
3	0	0
3	40,88	10
3	83,61	20
3	126,34	30
3	169	40
3	211,8	50
3	254,53	60
3	297,26	70
3	339,9	80
3	382,72	90
5	0	0
5	36,33	10
5	79,06	20
5	121,79	30
5	164,52	40
5	207,25	50
5	249,98	60
5	292,71	70
5	335,44	80
5	378,17	90