



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

# **Rediseño de un robot móvil tipo ballbot**

**TESIS**

Que para obtener el título de  
**Ingeniero Mecatrónico**

**P R E S E N T A N**

Humberto Alejandro Cruz Moreno

Gabriel Gutiérrez Santillán

**DIRECTOR DE TESIS**

M.I. Yukihiro Minami Koyama



**Ciudad Universitaria, Cd. Mx., 2019**

---

## **AGRADECIMIENTOS**

---

Investigación realizada gracias al proyecto UNAM-DGAPA-PAPIIT IT102615 "Robots no convencionales para exploración y búsqueda".

Primeramente quiero agradecerle a Dios por permitirme terminar una etapa más de mi vida, sin él nada de esto sería posible, le agradezco infinitamente por todas las enseñanzas que me ha brindado, también por esas buenas personas que ha puesto en mi camino para ayudarme, y finalmente por darme el mejor regalo que es mi familia la cual siempre ha estado conmigo y me ha apoyado en todo momento.

A mi amada universidad, la UNAM, le agradezco por todos los conocimientos que me brindó para que pudiera formarme como profesionista, también le agradezco mucho por dejarme vivir una de las mejores experiencias de mi vida, y es que desde que forme parte de esta comunidad universitaria, conocí a grandes amigos con los que hice maravillosos recuerdos, profesores que me dejaron grandes lecciones y esas bellas instalaciones donde nunca deje de conocer algo nuevo.

A mi mamá Laura Moreno Reyes, quiero agradecerle por todo el apoyo que me ha brindado, por creer en mí y por el ejemplo que me ha transmitido de nunca rendirse ante las adversidades. A mi padre Eugenio Cruz Miguel, por su apoyo en todos estos años y por los sabios consejos que me ha transmitido. A mis hermanas Laura y Anayely, por su apoyo y cariño, a mis sobrinos Uriel y Ángel, los cuales siempre me impulsan a ser una mejor persona, muchas gracias.

A mi director de tesis, Yukihiro Minami Koyama, le agradezco de todo corazón las enseñanzas que me ha transmitido, no solo como profesionista sino también como ser humano. A todos los miembros del Taller de Robótica Abierta, les agradezco el tiempo que estuve trabajando con ellos, en especial al coordinador Gerardo Ramos, por su paciencia y amabilidad al ayudarme a mis actividades dentro del taller. Al equipo FinDER UNAM por su apoyo durante la RoboCup, por último, a mis dos grandes amigos, Sergio Hernández y Mauro Rivero, por todo el apoyo que me brindaron en este proyecto y por los buenos momentos que hemos pasado juntos.

Hay que mencionar que este trabajo no fue realizado por una sola persona, por lo que quiero agradecerle a mi compañero y amigo Gabriel Gutiérrez Santillán, muchas gracias por todo el apoyo que me brindaste, también, quiero agradecerle a su familia, a la cual en estos últimos años le he tomado mucho cariño, gracias por todo el apoyo que me han brindado, por el buen trato que me han dado y por siempre hacerme sentir parte de su familia, de verdad muchas gracias.

A mi amigo Juan Carlos Pérez Ruiz, muchas gracias por creer en este proyecto y apoyarme para su culminación, a mi amigo Juan Antonio Mendoza por su compañía durante el escrito de la tesis, de verdad este periodo fue mucho más ameno gracias a tu compañía, también quiero agradecerle a mi amigo Eli por sus ánimos durante el desarrollo de la tesis, finalmente quiero agradecerles a todos mis amigos de la facultad por la gran amistad que hemos tenido hasta el momento, me siento muy feliz de culminar mi carrera en compañía de ustedes, gracias.

Por último, quisiera agradecerle a mi amigo Julio y decirle lo siguiente: Julio, después de tanto tiempo y mucho esfuerzo, por fin voy culminar una etapa más de vida, la verdad estoy un poco nervioso por las nuevas cosas que vendrán, sin embargo, al mismo tiempo estoy muy feliz porque he cumplido la promesa de terminar la universidad por los dos, finalmente, no me queda más que agradecerle por la inspiración que generaste en mí y espero que donde te encuentres, hayas podido disfrutar junto conmigo la vida universitaria que tanto quisiste vivir.

Humberto Alejandro Cruz Moreno Aka. Parky.

“El que da, no debe volver acordarse; pero el que recibe nunca debe olvidar”

Proverbio judío.

Agradezco a la UNAM, mi alma mater, por formarme profesionalmente dentro de la Facultad de Ingeniería a lo largo de todos estos años.

A toda mi familia por su amor, ayuda y apoyo incondicional, en especial a mis padres que sé que con un gran esfuerzo siempre me han brindado los recursos necesarios para que yo estudiara a lo largo de toda mi vida; asimismo quiero agradecer a mis hermanos por motivarme, hacer siempre todo más ameno y divertido. Quiero que sepan que los amo a todos y que les dedicó este trabajo porque siempre han estado ahí.

A mi amigo y compañero de tesis, Humberto Alejandro, por darme la oportunidad de trabajar con él en este proyecto, porque pese a las complicaciones que siempre estaban presentes, encontramos la manera de seguir adelante y al fin concluir este proceso. Quiero que sepa que he aprendido mucho de él porque es una gran persona y lo admiro mucho. Gracias también por aguantarme tanto tiempo, porque sé que puede ser difícil acoplarte para sacar lo mejor de un equipo.

A todos mis amigos del Taller de Robótica Abierta, del CIA, de la carrera y de la vida, por estar siempre al pendiente, por acompañarme desde el inicio de este proceso hasta el final, por sus consejos, los cuales siempre me han ayudado a tomar mejores decisiones. Muchas gracias por compartir tantos momentos, como por ejemplo una tarde videojuegos, una pequeña charla en la comida o una salida al cine. Los quiero mucho a todos.

A mis profesores dentro de la Facultad de Ingeniería, por compartir parte de su vida y conocimiento para ayudarme a desarrollarme como ingeniero. A nuestro asesor y director de tesis, Yukihiro Minami Koyama, por guiarnos a lo largo de todo este proyecto, por motivarnos a seguir adelante y por su paciencia. Por último quiero agradecer nuestros sinodales, por dedicarnos parte de su tiempo y por darnos sus puntos de vista para mejorar nuestro trabajo.

Muchas gracias a todos, porque sin ustedes nada de esto hubiera sido posible.

Gabriel Gutiérrez Santillán.

---

# Contenido

<b>AGRADECIMIENTOS</b> .....	1
<b>RESUMEN</b> .....	1
<b>Capítulo 1 INTRODUCCIÓN</b> .....	2
Estudio del estado de la técnica.....	2
Justificación .....	7
Objetivo .....	7
Hipótesis.....	7
Resumen por capítulos.....	8
<b>Capítulo 2 METODOLOGÍA</b> .....	10
Proceso de rediseño .....	11
<b>Capítulo 3 PLANTEAMIENTO Y ANÁLISIS DEL PROBLEMA</b> .....	13
Planteamiento del problema .....	13
Características del robot anterior .....	14
Análisis del problema .....	16
<b>Capítulo 4 SUBSISTEMA MECÁNICO</b> .....	18
Búsqueda de soluciones para el rediseño mecánico .....	18
Selección de los motores .....	19
Selección de las ruedas .....	24
Rediseño de la estructura .....	25
Acomodo de los actuadores.....	26
Generación de conceptos .....	27
Evaluación y selección del concepto .....	29
Diseño a detalle.....	29
Proceso de manufactura .....	32
<b>Capítulo 5 SUBSISTEMA ELECTRÓNICO</b> .....	33
Búsqueda de soluciones para el rediseño electrónico.....	33
Etapa de potencia.....	34
Selección de la tarjeta controladora. ....	34
Convertidor de niveles lógicos .....	35
Pruebas a la etapa de potencia.....	35
Sensor de orientación .....	36
Selección de la IMU.....	37
Prueba de las IMU .....	39
Sensores de corriente .....	41
Unidad de procesamiento.....	41
Alimentación .....	42

Diseño de las conexiones electrónicas.....	43
<b>Capítulo 6 PROGRAMACIÓN Y CONTROL.....</b>	<b>44</b>
Búsqueda de soluciones para el desarrollo de la parte lógica.....	44
Modelo del motor.....	46
Modelo principal del ballbot.....	47
Energías del sistema.....	49
Entrada del sistema.....	50
Ecuaciones de Lagrange.....	51
Modelo completo del sistema.....	53
Relación entre el par del modelo y el par de las tres ruedas.....	53
Ángulo de inclinación y ángulo del plano de inclinación.....	53
Matriz de transformación de par.....	59
Ecuaciones de relación de velocidad.....	62
Diseño y simulación de los controladores.....	65
Control de par del motor.....	65
Control con relación de velocidad.....	70
<b>Capítulo 7 PRUEBAS DE ESTABILIDAD.....</b>	<b>71</b>
Primeras pruebas.....	71
Etapa de análisis y cambios en el diseño.....	74
<b>Capítulo 8 PRUEBAS DE MOVILIDAD.....</b>	<b>80</b>
Primeras pruebas.....	80
Etapa de análisis y cambios en el diseño.....	82
Pruebas finales.....	83
<b>RESULTADOS Y CONCLUSIONES.....</b>	<b>85</b>
<b>TRABAJO A FUTURO.....</b>	<b>87</b>
<b>APÉNDICE.....</b>	<b>89</b>
A1. Parámetros para el diseño de correderas y soportes de motores.....	89
A2. Deducción de las energías para el modelo del sistema.....	91
A3. Cálculo de los parámetros del modelo.....	95
A4. Simulación de controlador por realimentación de estados.....	102
A5. Evaluación de placas de desarrollo.....	105
A6. Componentes del robot.....	107
A7. Código de programación.....	110
A8. Planos.....	120
<b>REFERENCIAS.....</b>	<b>131</b>

---

# RESUMEN

---

El presente trabajo aborda el rediseño de un robot móvil tipo ballbot desarrollado en el Taller de Robótica Abierta de la Facultad de Ingeniería de la UNAM. Este proyecto partió del interés de explorar un tipo de movilidad diferente al que usualmente se implementa en los robots (ruedas o patas). Dentro de los objetivos se planteó el diseño de una nueva estructura, el desarrollo de un algoritmo de control para su estabilidad y su navegación incipiente.

Para el desarrollo del robot, se estudió el estado de la técnica y los antecedentes del proyecto, para su análisis el trabajo fue dividido en tres partes principales, una parte mecánica, una electrónica y una de control y programación. El rediseño de cada una de estas partes se abordó utilizando las cinco etapas del proceso de diseño en la ingeniería.

Una vez terminada la etapa de diseño del robot, se realizaron las pruebas de estabilización y navegación, posteriormente, se evaluaron los resultados con los objetivos planteados al inicio del proyecto y finalmente se dieron las conclusiones y el trabajo a futuro.

**PALABRAS CLAVE:** robot móvil, estabilidad, ballbot, ángulo de inclinación, teoría de control.

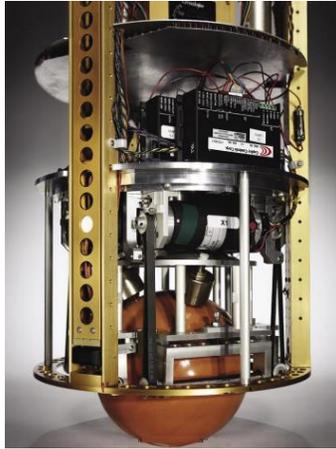
Un ballbot es un robot móvil capaz de permanecer en equilibrio dinámico sobre una esfera, que le permite al robot desplazarse en superficies con poca inclinación utilizando solamente un punto de apoyo, lo que resulta ser muy ventajoso con respecto a otros tipos de robots móviles, ya que puede maniobrar sobre entornos reducidos o con obstáculos, sin embargo, no es capaz de navegar sobre superficies irregulares ya que esto dificulta la estabilidad del robot.

## **Estudio del estado de la técnica**

A continuación, se presenta una breve recopilación de robots de este tipo que se han desarrollado a lo largo del tiempo, haciendo mención de las principales características, mejoras o avances que estos presentan.

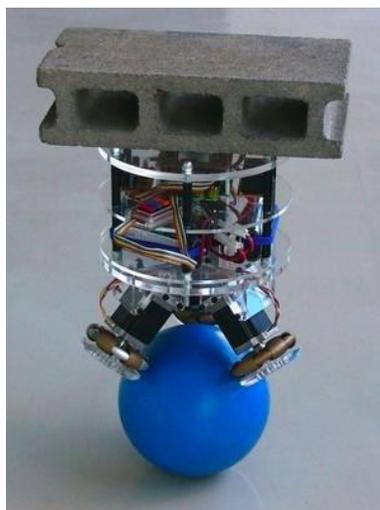
El primer robot tipo ballbot fue construido en el 2006 en la Universidad de Carnegie Mellon [1], que contaba con un cuerpo cilíndrico de 40 cm de diámetro, tenía una altura de 1.5 m y un peso de 45 kg. Para su funcionamiento se utilizó un mecanismo similar al de un ratón de computadora, pues contaba con un conjunto de rodillos, los cuales eran accionados para generar un movimiento en la esfera, ver

Figura 1-1. Para el control de dicho robot se implementó un controlador regulador cuadrático lineal (LQR), con el cual se logró la estabilización, sin embargo, su rendimiento fue pobre.



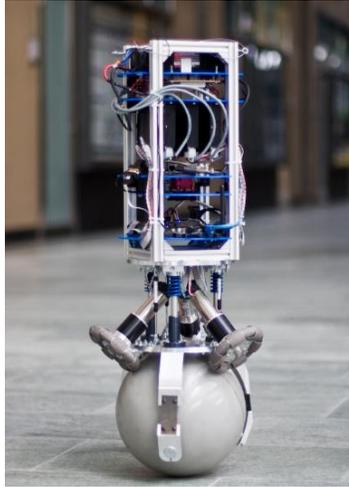
*Figura 1-1 ballbot de la Universidad de Carneige Mellon.*

Poco después, en la Universidad de Tohoku Gakuin, ubicada en Sendai, Japón [2] se desarrollaron dos ballbots, el primero tenía una altura de 1.3 m, un peso de 12 kg y contaba con tres ruedas, las cuales estaban a  $120^\circ$  entre sí, cuyos ejes tenían un ángulo de inclinación con respecto a la vertical de  $45^\circ$ . El segundo diseño que se desarrolló, nombrado "BallP", fue pensado para poder desplazarse con una carga adicional de hasta 11 kg, tenía una altura de 50 cm, un peso de 11 kg y el ángulo de las ruedas con respecto a la vertical era de  $40^\circ$ . En ambos casos fue utilizada una bola de boliche recubierta con un polímero, la cual tenía 20 cm de diámetro y pesaba 3.6 kg. El acomodo de las ruedas en ambos robots no tuvo más que el objetivo de que estos pudieran girar sobre su propio eje, ver Figura 1-2. Para la estabilidad de ambos robots se implementaron controladores proporcionales derivativos (PD), los cuales controlaban el ángulo de cabeceo (normalmente conocido como *pitch*), el ángulo de alabeo (normalmente conocido como *roll*) y la posición del ballbot simultáneamente. El controlador fue usado tanto para mantener la posición de equilibrio, como para dar movimiento al robot.



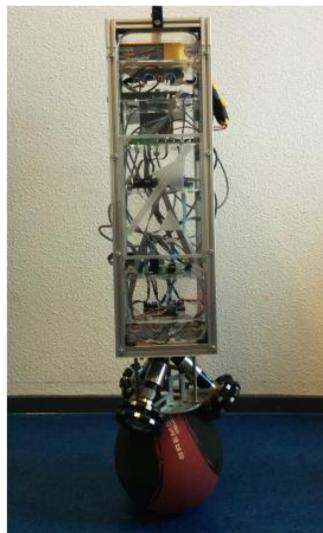
*Figura 1-2 Ballbot de la Universidad de Tohoku Gakuin.*

En el 2010 en la Universidad Politécnica de Zúrich, ubicada en Suiza [3], se desarrolló un robot tipo ballbot llamado Rezero, que se muestra en la Figura 1-3. En comparación con los mencionados anteriormente, este tuvo un mejor desempeño, ya que el robot se podía inclinar hasta  $20^\circ$  con respecto a la horizontal sin perder el equilibrio. Para su control se implementó un controlador no lineal basado en la teoría de los controladores LQR.



*Figura 1-3 Rezero de la Universidad Politécnica de Zúrich.*

En el 2014 en la Universidad de Twente, Países Bajos [4], se desarrolló un ballbot capaz de estabilizarse y desplazarse en cualquier dirección, ver Figura 1-4, además se realizaron simulaciones, cambiando los parámetros del robot como la altura de su centro de masa, el ángulo de inclinación de las ruedas con respecto a la vertical y el peso, con el fin de saber cómo dichos cambios afectaban su estabilidad. Al igual que en el robot de la universidad de Zúrich, se implementó un control no lineal.



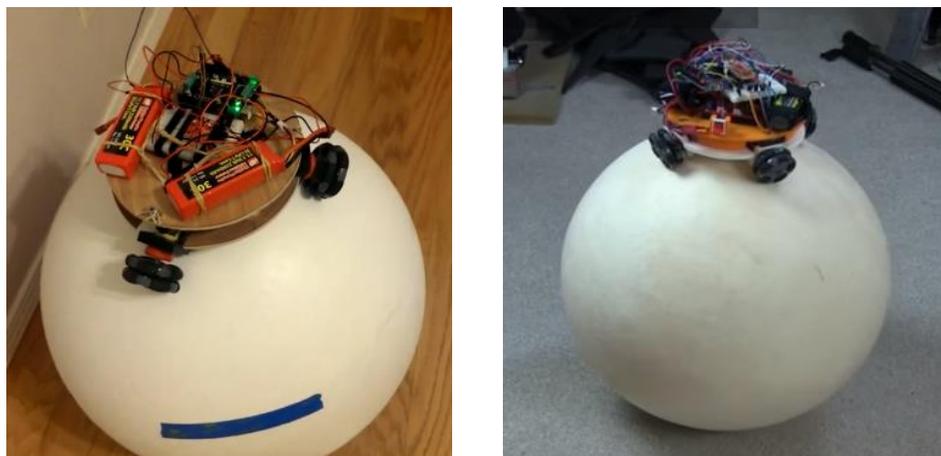
*Figura 1-4 Ballbot de la Universidad de Twente.*

Además de los ballbots desarrollados para fines de investigación, se encontraron ballbots desarrollados por personas interesadas en este tipo de robots, tal es el caso de un estudiante de la Universidad de Queensland, quien en el 2015 diseño uno, cuya estructura era de acrílico y se equilibraba sobre una pelota de basquetbol [5], ver figura 1-5. Para su estabilización, se utilizó un controlador proporcional integral derivativo (PID) de posición, además de utilizar las lecturas de unos decodificadores (conocidos como *encoders*) para calcular la posición del robot.



*Figura 1-5 Ballbot de acrílico.*

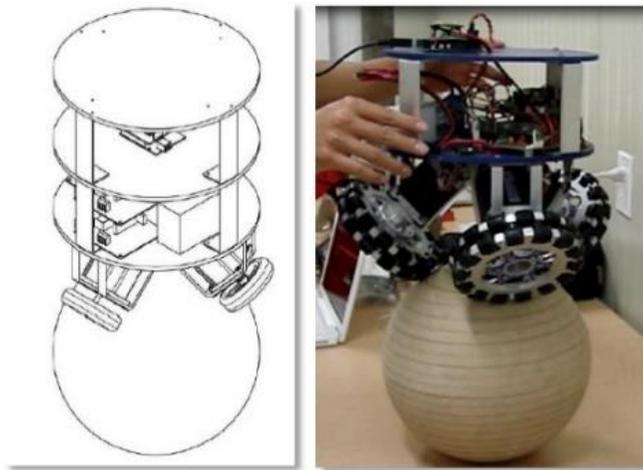
También en el 2015 “Xrobots”, una página web dedicada al desarrollo de proyectos de electrónica [6], desarrolló un robot tipo ballbot de cuatro ruedas, mostrado en la figura 1-6, en donde, para la estabilización del robot se implementó un controlador de posición tomando al robot como un péndulo invertido y su movimiento en dos planos acoplados entre sí, además se implementó la navegación del robot por comunicación inalámbrica. Posteriormente en el 2016, un estudiante de ingeniería en computación implementó la estabilización de un ballbot utilizando la misma arquitectura de control que el ballbot desarrollado por la página Xrobots [7], pero solamente utilizando tres ruedas, ver figura 1-6.



*Figura 1-6 ballbot de tres y cuatro ruedas.*

Asimismo, dentro de la Facultad de Ingeniería se han desarrollado proyectos similares, el primero de ellos fue una tesis de licenciatura de la carrera de Ingeniería Eléctrica Electrónica [8]. La tesis se basó en utilizar *Legó Mindstorms (línea de juguetes de robótica fabricada por la empresa LEGO)* para desarrollar un robot tipo ballbot, en donde para su control se utilizó el software Matlab y una interfaz directa conectada al dispositivo. Este proyecto logró conseguir la estabilización del robot.

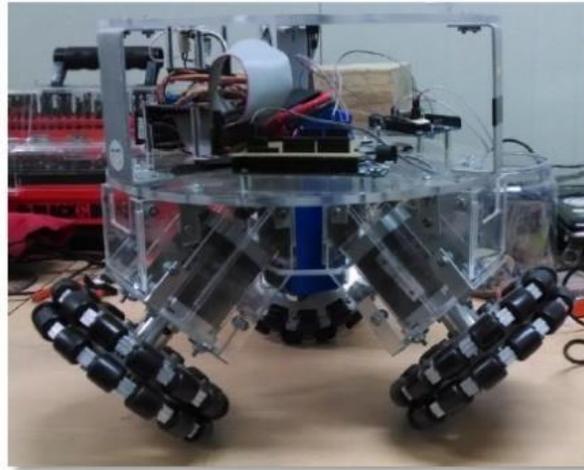
El siguiente proyecto de este tipo que se desarrolló dentro de la Facultad, más específicamente dentro del Taller de Robótica Abierta (TRA), fue una tesis de licenciatura de la carrera de Ingeniería Mecatrónica. En este trabajo se buscaba el desarrollo de un ballbot incluyendo su diseño mecánico. Dentro de las metas obtenidas se logró construir el robot que se muestra en la **¡Error! No se encuentra el origen de la referencia.**, el cual era capaz de realizar los movimientos necesarios para desplazarse horizontalmente; sin embargo, el robot no consiguió estabilizarse sobre una esfera. Se concluyó que las razones principales fueron los retrasos que se tenían en las lecturas de la posición angular del robot y el desempeño de los componentes electrónicos, ya que en ellos se presentaba ruido. Este proyecto es la primera iteración que antecede al presente trabajo [9].



*Figura 1-7 Diseño conceptual y prototipo final de la primera iteración.*

Se desarrolló una segunda iteración de este proyecto, la cual también fue una tesis de licenciatura de la carrera de Ingeniería Mecatrónica [10]. En esta segunda iteración se rediseñó toda la estructura mecánica y en cuanto a la parte electrónica, ver **¡Error! No se encuentra el origen de la referencia.**, se adquirieron y fabricaron nuevos componentes, además de que se reacomodaron, todo esto con el fin de mejorar el desempeño del robot. Este robot logró la estabilización, pero por un tiempo muy corto. En este caso se concluyó que se tuvieron algunos problemas, principalmente debido a que el par en los motores no era suficiente; asimismo se menciona que se generaban movimientos no deseados en el robot debido a la discontinuidad entre los rodillos de las ruedas y además existía ruido en el sensor que determinaba el ángulo de inclinación del robot. En esta tesis, como un trabajo a futuro, se propone que

se realicen algunos cambios en el robot, los cuales son mencionados posteriormente, para mejorar la estabilidad del mismo.



*Figura 1-8 Segundo Prototipo del Ballbot.*

## **Justificación**

El presente trabajo pretende generar conocimiento dentro de la Facultad de Ingeniería, enfocado al desarrollo de robots tipo ballbot o robots móviles cuyas características de movimiento sean similares, mediante la fabricación de un nuevo robot en donde se toma como referencia la última iteración del ballbot desarrollado dentro del TRA.

Se espera que este trabajo sirva como guía dentro de la misma Facultad para el desarrollo de futuros proyectos, ya sea que estén orientados a la investigación o incluso al desarrollo de un producto en el área de transporte, servicio doméstico o en el sector industrial.

## **Objetivo**

Rediseñar el robot tipo ballbot desarrollado en el Taller de Robótica Abierta de la Facultad de Ingeniería de la UNAM, con el fin de que éste pueda estabilizarse y sea capaz de seguir una trayectoria simple.

## **Hipótesis**

La resolución de cada uno de los problemas detectados en el robot anterior, los cuales son descritos en el trabajo que antecede a este proyecto, harán que el robot tenga un mejor desempeño en su funcionamiento.

## Resumen por capítulos

El primer capítulo es una breve introducción, en el que se detalla el Estudio del estado de la técnica de los ballbots desarrollados en otras universidades, además de los trabajos que anteceden a este proyecto dentro de la Facultad de Ingeniería. Asimismo, se establece la justificación de este proyecto, su objetivo e hipótesis.

En el segundo capítulo se habla de la metodología que se utilizó para el desarrollo de este proyecto, es decir, se explica cómo fue dividido el trabajo y se da una breve explicación de los pasos del proceso de diseño que se siguieron.

El tercer capítulo detalla el planteamiento y el análisis del problema del proyecto.

El cuarto capítulo abarca el rediseño mecánico del ballbot, en el que, gracias al proceso de diseño de ingeniería se logró llegar a un diseño previo y posteriormente un diseño final, el cual fue manufacturado y ensamblado.

El quinto capítulo aborda el rediseño electrónico del robot, en el que se evaluaron y seleccionaron los nuevos dispositivos electrónicos que conformaron al robot con base en los requerimientos que se plantearon, y finalmente se muestra cómo se realizaron las conexiones electrónicas y se montaron los dispositivos en el ballbot para ser probados.

En el sexto capítulo se habla de la programación y control del robot, en el que se plantearon y evaluaron los controladores que se emplearían para lograr la estabilización y posteriormente la navegación del ballbot. Asimismo, se plantean las ecuaciones cinemáticas y dinámicas que determinan el comportamiento del sistema y se explica cómo se logra medir el ángulo de inclinación del robot.

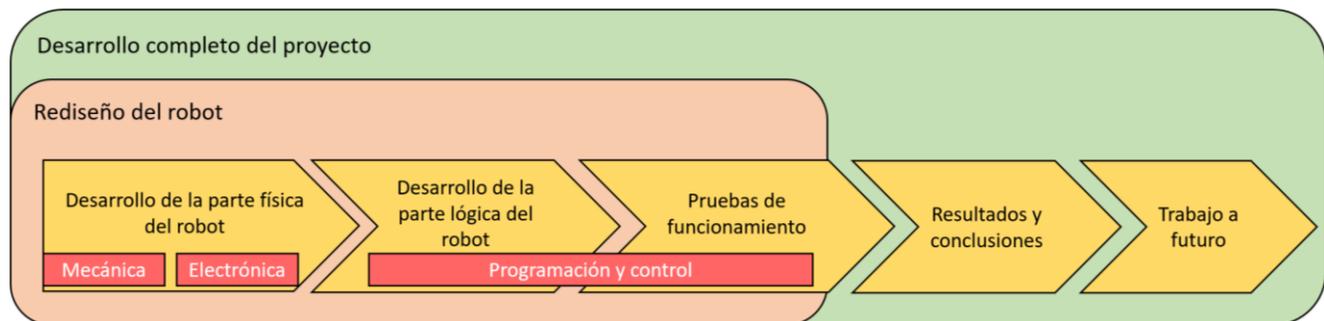
En el séptimo capítulo se detallan las pruebas de estabilidad que se realizaron al robot utilizando los controladores propuestos en el capítulo anterior; posteriormente se realizó una evaluación para saber cuál es el controlador más adecuado para lograr la estabilidad y se propusieron cambios para mejorar el desempeño del robot. Finalmente, una vez hechas las modificaciones se realizaron nuevamente las pruebas de estabilización para comprobar si mejoraba el comportamiento del robot.

En el octavo capítulo se abordan las pruebas de navegación incipiente, en el que se detallan las pruebas de movilidad a las que fue sometido el robot; luego se realizó un análisis de su desempeño y se propusieron algunas modificaciones para mejorar la navegación. Finalmente, una vez hechas dichas modificaciones se realizaron nuevamente las pruebas de movilidad para verificar si el desempeño del robot había mejorado.

Posteriormente se muestra el análisis de los resultados que se obtuvieron, en donde se evaluó si estos cumplieron con los objetivos planteados al inicio del presente trabajo y se obtuvieron algunas conclusiones. Finalmente se detallaron algunos aspectos que no fueron cubiertos durante el proyecto, los cuales podrían servir como referencia para la continuación del mismo.

En la parte de los apéndices se presentan algunas especificaciones de los elementos mecánicos y electrónicos utilizados en este proyecto, los planos de fabricación de las estructuras utilizadas, parte del desarrollo del modelado matemático del robot y los códigos de programación que se utilizaron para lograr la estabilización y navegación del ballbot.

Puesto que el desarrollo completo de todo el proyecto implica varios procesos y la comprensión de muchos aspectos, se dividió en varias etapas, las cuales son representadas con flechas de color amarillo en la figura 2-1. Dichas etapas se describen a continuación.



*Figura 2-1 Etapas del desarrollo del proyecto*

Tanto la mecánica como la electrónica conforman lo que es la etapa de desarrollo de la parte física del robot; esta fue la primera a desarrollar pues se consideró importante tener el robot armado para hacer pruebas posteriores. Asimismo, ésta etapa abarcó la mayor cantidad de tiempo, debido a la manufactura, el ensamble y el tiempo de adquisición de componentes.

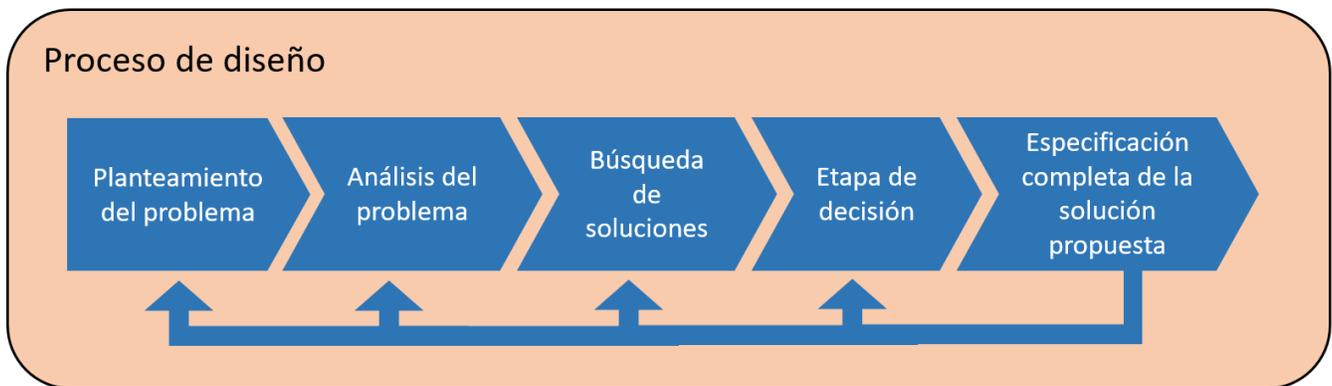
Posterior al desarrollo de la parte física, siguió el desarrollo de la parte lógica, que no es más que el programa que le da movimiento al robot. Puesto que el objetivo principal de este proyecto plantea lograr la estabilidad del ballbot y posteriormente implementar un algoritmo sencillo de navegación, se tomó como fundamento toda la teoría de modelado y control para desarrollar dicho programa.

Teniendo ya el diseño del robot, se realizaron pruebas de funcionamiento para evaluar el desempeño del robot y, con base en ello, realizar cambios tanto en la parte física del robot como en su

programación para mejorar su funcionamiento. Tanto esta etapa como las antes mencionadas conforman todo lo que es el proceso de diseño en la ingeniería, el cual se explica con mayor detalle en el siguiente apartado del capítulo. Finalmente, con los resultados obtenidos se obtuvieron las conclusiones y se propuso un trabajo a futuro para iteraciones posteriores.

## Proceso de rediseño

Se debe de mencionar que cuando se está diseñando, o en nuestro caso “rediseñando”, para facilitar dicho proceso se siguen una serie de pasos. Para el desarrollo de este proyecto en particular se utilizaron las cinco etapas del proceso de diseño en la ingeniería de Krick [11], que se muestran en la figura 2-2.



*Figura 2-2 Etapas del proceso de diseño.*

### Planteamiento del problema

En esta parte, el problema se define de forma amplia y sin detalles. Es decir, se explica lo que se intenta resolver sin saber a ciencia cierta cómo hacerlo.

### Análisis del problema

En esta parte del proceso, el problema se define de una forma más detallada con base en las investigaciones y consultas realizadas. De tal modo que se definen las variables de entrada, de salida y variables de solución para dicho problema. También se definen las restricciones y los criterios de selección.

### Búsqueda de soluciones

Esta parte está conformada por la generación de conceptos que pueden ayudar a resolver el problema, en donde se toma como base la investigación realizada, conocimientos adquiridos a lo largo del tiempo y la creatividad.

### **Etapas de decisión**

En esta parte se evalúan las características más relevantes de cada concepto, las cuales se tienen que detallar de mejor manera para analizar sus ventajas y desventajas. Finalmente, mediante un proceso de selección se elige el o los conceptos que solucionen mejor el problema planteado con base en los criterios de selección.

### **Especificación completa de la solución propuesta al problema identificado**

En esta parte se hace una descripción más detallada de la solución propuesta, considerando atributos físicos y características de funcionamiento, con el fin de que pueda ser realizada dicha solución. Se debe de mencionar que el proceso de diseño es iterativo por lo que se puede regresar a etapas anteriores para dar una mejor solución al problema.

### Planteamiento del problema

En el último trabajo que se desarrolló dentro de la Facultad de Ingeniería de la UNAM acerca del desarrollo de un robot tipo ballbot se menciona que este fue capaz de estabilizarse por un tiempo de 5 segundos aproximadamente, sin embargo, este desempeño no fue suficiente para implementar algún algoritmo sencillo de navegación. Se concluyó que una de las principales razones fue que los motores no tenían el par suficiente para mantener al ballbot en equilibrio. Asimismo, se menciona que se observaron movimientos no deseados en el robot debido a las discontinuidades que existen entre los rodillos de las ruedas. Por otro lado, se encontraron problemas de ruido en la adquisición de datos de la IMU (Unidad de Medición Inercial) la cual determina el ángulo de inclinación del robot con respecto a la vertical.

Para dar solución al problema antes mencionado, primero se definió el estado inicial y final del sistema tal y como se muestra en la figura 3-1. En ella, el robot puede ser visto como una caja negra, sin embargo, para tener un mejor entendimiento del mismo, se revisaron todas las características del anterior diseño, las cuales se mencionan a continuación.



*Figura 3-1 Estado inicial y final del sistema.*

## Características del robot anterior

El robot anterior, que se mostró en la figura 1-8, estaba compuesto principalmente de una parte física (*hardware*) y de una parte lógica (*software*). La parte física del robot puede dividirse a su vez en una parte mecánica y una electrónica.

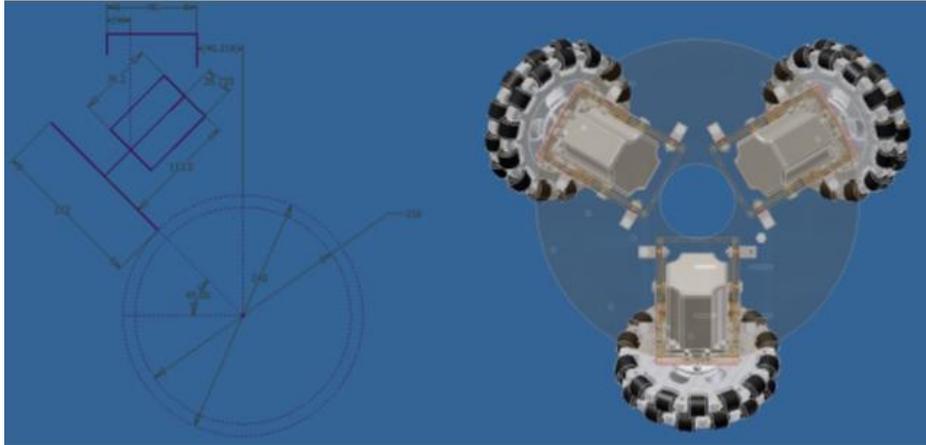
Una de las partes que conformaban la mecánica del robot es la estructura; está fue fabricada con placa de acrílico y para su ensamble se utilizaron tornillos con tuercas de seguridad. La mayor parte de la estructura se enfocó en la sujeción de los actuadores, pero adicionalmente se agregaron platos, uno para el acomodo de los componentes electrónicos y el otro para la manipulación del mismo robot; se utilizaron separadores entre dichos platos los cuales fueron hechos de lámina de aluminio. En las pruebas de estabilización, esta estructura tuvo un buen desempeño, refiriéndose a que en ningún momento presentó deformaciones debidas a las fuerzas que actuaban en la misma estructura. La esfera que se utilizó fue una pelota de basquetbol de 25 cm de diámetro.

Otra parte importante de la mecánica fueron los tres actuadores que se utilizaron, los cuales están conformados por un motor paso a paso, una rueda omnidireccional y un acoplamiento que los une. Los motores paso a paso que se utilizaron y que se muestran en la figura 3-2, cuentan con un par máximo de 1.85 N·m y fueron utilizados durante las pruebas de estabilización a una velocidad de hasta 60 RPM. En cuanto a las ruedas omnidireccionales, se utilizaron unas de 6" de diámetro.



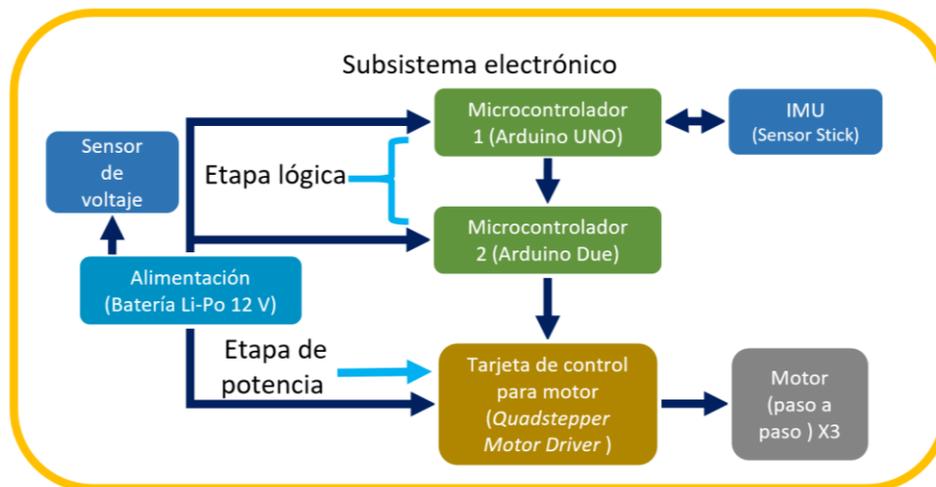
*Figura 3-2 Motor paso a paso modelo 23Y202S-LW8.*

Para su acomodo en la estructura, los motores fueron acoplados de tal modo que estuvieran colocados a  $120^\circ$  uno del otro y además que sus ejes tuvieran un ángulo de inclinación de  $45^\circ$  con respecto a la horizontal. Asimismo, se buscó que los planos tangentes a la esfera que pasan en el punto de contacto de cada rueda con la esfera fueran paralelos a cada eje del motor, con el fin de reducir problemas debido a la fricción, además de que en esta posición es posible transmitir el par de los motores a los ejes que pasan por el centro de la esfera, ver figura 3-3, a lo largo de este trabajo a esta característica se le denominará tangencia entre las ruedas y la esfera. La configuración antes descrita fue elegida con base en trabajos que se realizaron en otras universidades ya que, al tener únicamente tres ruedas, el robot no sólo sería capaz de desplazarse sobre un plano sino también sería capaz de girar sobre su propio eje.



*Figura 3-3 Configuración de los motores del diseño anterior.*

Para describir el sistema electrónico del robot anterior se elaboró un diagrama, que se muestra en la figura 3-4, en donde se revisó la arquitectura de dicho sistema y las características de los componentes que lo conforman.



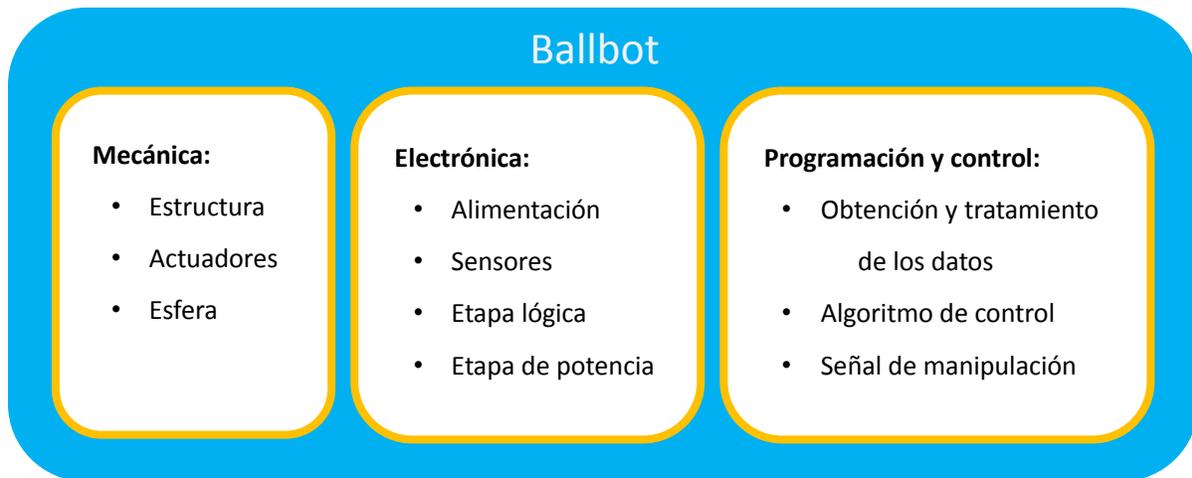
*Figura 3-4 Diagrama del subsistema electrónico del diseño anterior.*

Como se muestra en la figura 3-4, el robot contaba con una etapa de potencia y una etapa lógica. La etapa de potencia, la cual se encarga de alimentar y dar movimiento a los motores dependiendo de una señal de control, estaba conformada por una tarjeta controladora para los 3 motores paso a paso. Por otro lado, la parte lógica que se encarga de adquirir datos del entorno, los cuales se obtienen mediante sensores para posteriormente procesarlos y convertirlos en señales que se encargan del control de los actuadores, estaba conformada por dos microcontroladores, un Arduino Uno, el cual se encargaba de adquirir los datos de la IMU y realizar el algoritmo de control, y un Arduino Due, el cual se encargaba de transformar la señal de control en una señal que pudiera leer la tarjeta controladora de los motores paso a paso.

En cuanto al algoritmo de control que se utilizó para estabilizar al robot, fue un control PD, en donde la variable controlada fue el ángulo de inclinación del robot con respecto a la vertical, y la variable manipulada fue la aceleración de cada una de las ruedas. Por otro lado, se utilizó un filtro pasa bajas para eliminar el ruido de la señal de los datos obtenidos por la IMU.

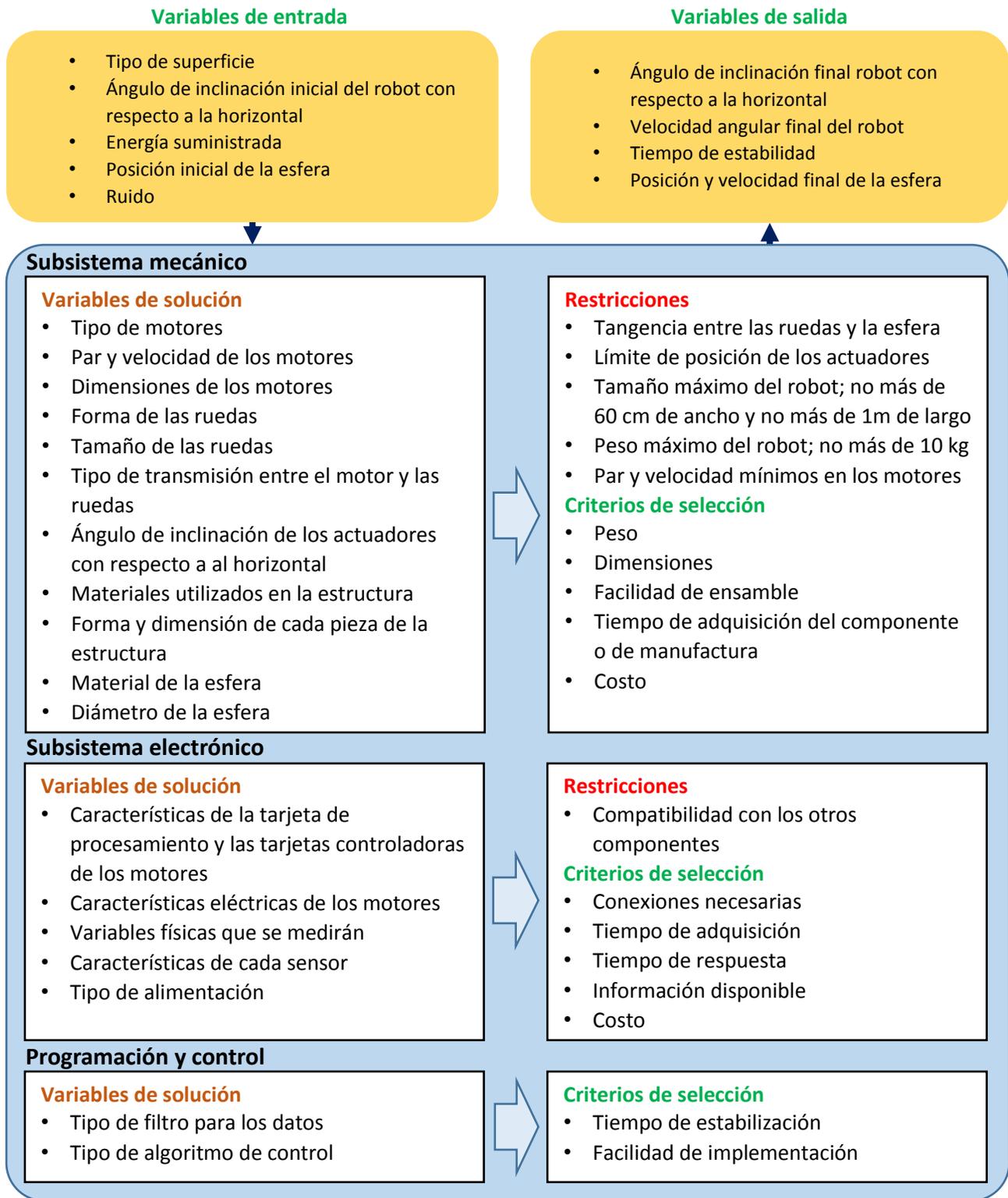
## Análisis del problema

Ya que se habla de un rediseño, se realizó un diagrama mostrado en la figura 3-5, en donde se identificaron las partes que debería de tener el nuevo robot partiendo de las características del diseño anterior. Para el nuevo diseño el robot igualmente fue dividido en tres subsistemas, uno mecánico, uno electrónico y finalmente uno de programación y control. Cada parte se clasificó en uno de estos subsistemas sin dar alguna especificación.



*Figura 3-5 Componentes del robot para el nuevo diseño.*

Para completar el análisis del problema se definieron las variables de entrada y salida, las cuales no son más que condiciones iniciales y finales de nuestro sistema. Asimismo, se definieron las variables solución, que son los parámetros que se pueden cambiar en el sistema con el fin de dar solución al problema, estos se muestran en la figura 3-6. Por otra parte, se definieron las restricciones y los criterios de selección para las ideas propuestas.



*Figura 3-6 Análisis del problema.*

En cuanto a las siguientes partes del proceso de diseño, como son la búsqueda de soluciones, la etapa de decisión y la especificación completa de la solución, fueron revisadas para cada uno de los subsistemas que se definieron anteriormente.

### **Búsqueda de soluciones para el rediseño mecánico**

La primera parte que se trabajó en el rediseño fue la mecánica. Para esto, además de revisar el análisis del problema que se aborda en el capítulo 3, el rediseño mecánico también se hizo con algunas de las propuestas que se plantearon en la tesis anterior para mejorar las características del robot. Dichas propuestas son:

- Utilizar otros motores, con un par suficiente para regresar el robot a su estado estable
- Cambiar los motores paso a paso por unos de corriente directa
- Diseñar una estructura que permita montar los motores a diferentes ángulos de inclinación con respecto a la horizontal, ya que en la estructura anterior sólo se podían colocar a un ángulo de 45°, esto con el fin de poder analizar si al cambiar la inclinación de los motores existía alguna posición óptima para su funcionamiento.
- Utilizar ruedas omnidireccionales con mayor continuidad entre cada rodillo y con menor diámetro para reducir los movimientos no deseados

Puesto que los problemas mecánicos detectados tenían que ver más con los actuadores que con la estructura o con la esfera, se decidió como primera tarea seleccionar tanto unos nuevos motores como unas nuevas ruedas. Se determinó que la transmisión entre los motores y las ruedas fuera directa con el fin de no complicar la parte de control.

Se encontró que la estructura anterior fue diseñada mayoritariamente en función de la dimensión de los actuadores, y ya que estos cambiarían, se decidió rediseñar toda la estructura completa en vez de remplazar solamente algunas partes; sin embargo, muchas de las características del robot anterior se retomaron e incluso se consideraron restricciones, pues al cambiar dichas características podría verse afectado el funcionamiento del robot.

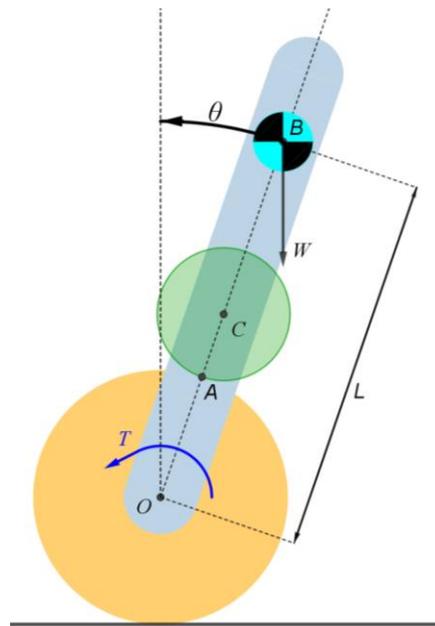
## Selección de los motores

Para seleccionar los nuevos motores, se debía obtener la aproximación del par máximo que necesitaría cada motor; para esto primero se realizó un análisis estático, cuyo objetivo principal fue calcular el par necesario para mantener al robot en equilibrio a una cierta inclinación. Dicho análisis se hizo mediante una simplificación del robot que se muestra en la figura 4-1. Como se puede observar, el robot está compuesto por un cuerpo, una esfera y una rueda virtual, la cual representa a las tres ruedas que le dan movimiento al robot. Esta representación se hace sobre un plano que no es más que el plano en donde el robot se inclina. Esto se revisa con mayor detalle a lo largo del capítulo 6 de programación y control, pues dicha representación se utilizó para el modelo del comportamiento dinámico del sistema.

Las variables que se utilizaron en el análisis se muestran en la tabla 4-1.

*Tabla 4-1 Variables del diagrama del ballbot.*

$T$	Par de los motores ejercido en la esfera
$\theta$	Ángulo de desplazamiento del robot
$L$	Distancia entre el centro de la esfera y el centro de masa del robot
$W$	Peso del cuerpo del robot



*Figura 4-1 Simplificación del ballbot.*

Con base en el diagrama anterior, se realizó la suma de momentos con respecto al centro de la esfera, obteniendo la siguiente ecuación:

$$\sum M_o = T - WL\text{sen}\theta = 0 \quad (4.1)$$

De la ecuación anterior se puede despejar al par  $T$ , por lo que la ecuación queda de la siguiente manera:

$$T = WL \operatorname{sen} \theta \quad (4.2)$$

Considerando que en el centro de la rueda virtual hay un motor al cual le llamaremos motor virtual, el par que necesitaría entregar dicho motor para producir el par  $T$  se calcula mediante la siguiente ecuación:

$$T_{Mv} = T \frac{r_{Rv}}{r_e} \quad (4.3)$$

En donde  $T_{Mv}$  es el par del motor virtual,  $r_{Rv}$  es el radio de la rueda virtual y  $r_e$  es el radio de la esfera.

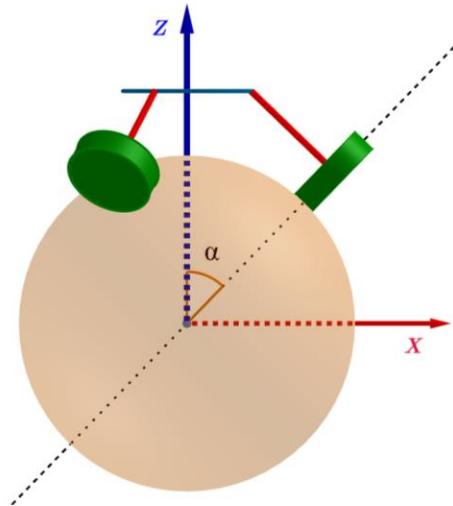
Si se sustituye el par  $T$  de la ecuación anterior por el par de la ecuación 4.3, la ecuación 4.2 queda de la siguiente manera:

$$T_{Mv} = \frac{WLr_{Rv}}{r_e} \operatorname{sen}(\theta) \quad (4.4)$$

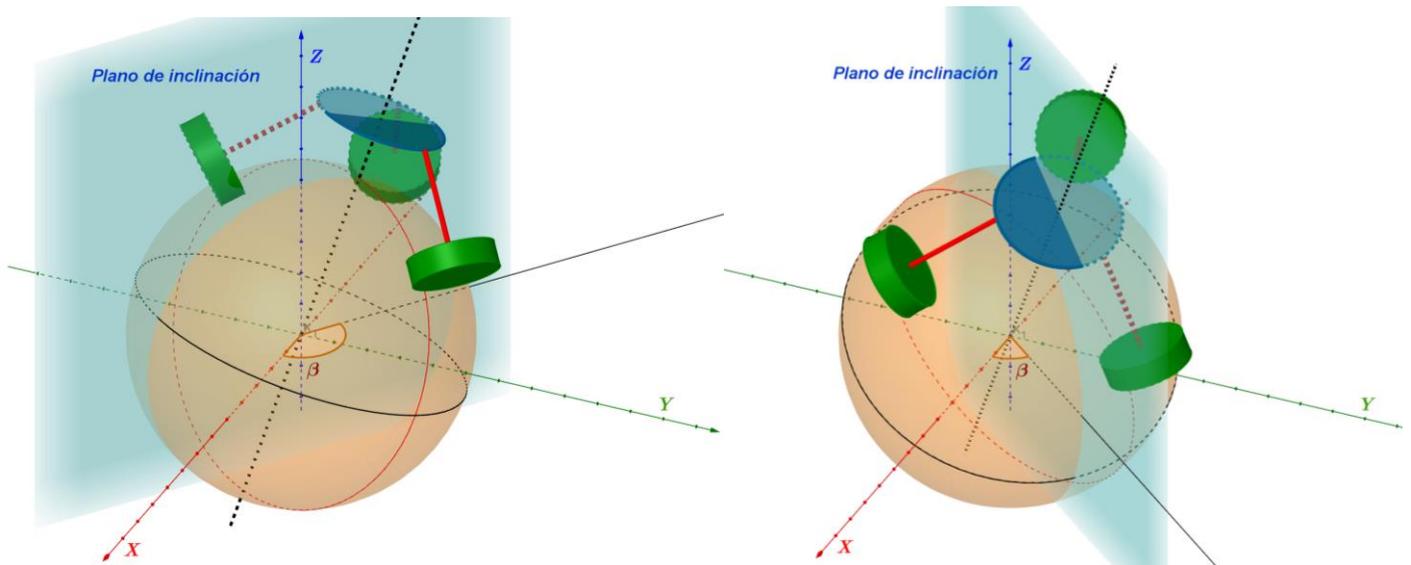
Esta ecuación permite calcular el par que requiere el motor virtual para mantener al robot estático en una cierta inclinación; sin embargo, este no es el par que se necesitaría en cada uno de los tres motores que mueven al robot. Para esto se obtuvo una relación entre el par del motor virtual y el par de cada uno de los tres motores reales. Dicha relación se muestra a continuación mediante una ecuación matricial:

$$\frac{\tau_{Mv}}{3 \cos(\alpha)} \begin{bmatrix} 2 \operatorname{sen}(\beta) \\ \sqrt{3} \cos(\beta) - \operatorname{sen}(\beta) \\ -\sqrt{3} \cos(\beta) - \operatorname{sen}(\beta) \end{bmatrix} = \begin{bmatrix} \tau_{m1} \\ \tau_{m2} \\ \tau_{m3} \end{bmatrix} \quad (4.5)$$

Donde  $\tau_{mi}$  es el par de cada motor  $i$ , donde  $i = 1,2,3$ ,  $\alpha$  es el ángulo que determina la inclinación que tienen las ruedas con respecto al eje vertical del robot y  $\beta$  es un ángulo que determina hacia donde se inclina el robot, ambos ángulos se muestran en la figuras 4-2 y 4-3. Al igual que la simplificación del modelo del robot, la ecuación 4.5 se revisa con mayor detalle en el capítulo 6 pues se relaciona con la parte del modelado y la programación del robot, por lo que en dicho capítulo se explica cómo se obtuvo (ir a la página 53 y revisar el tema de *Relación entre el par del modelo y el par de las tres ruedas*). Por el momento lo importante de esta ecuación es ver que el par en cada motor no únicamente varía dependiendo del ángulo de inclinación del robot, sino también dependiendo de hacia dónde se incline, por tal motivo, utilizando la ecuación matricial se hizo un análisis del comportamiento del par en cada motor variando el ángulo  $\beta$ , pues es el parámetro que determina hacia donde se está inclinando el robot, en el que se obtuvo una gráfica que se muestra en la figura 4-4.

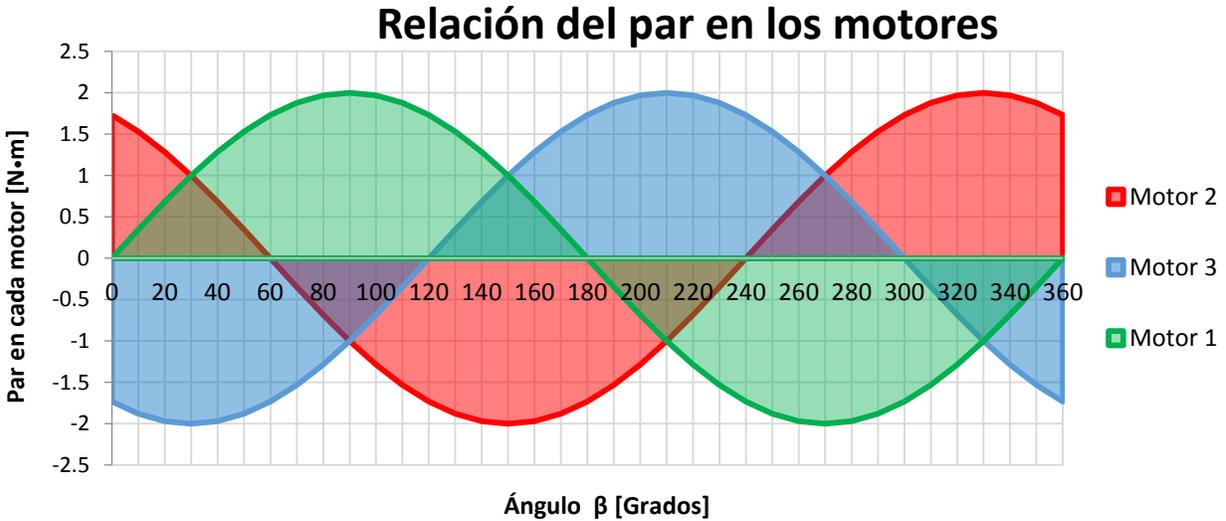


**Figura 4-2** *Angulo de inclinación de las ruedas con respecto a la vertical.*



**Figura 4-3** *Representación del ángulo  $\beta$  determinado por el plano de inclinación del robot.*

Como se puede observar en la figura 4-4, dependiendo del valor del ángulo  $\beta$ , el par que se le demanda a cada motor variará incluso si el robot tiene la misma inclinación con respecto a la vertical. Además, se observa que el comportamiento del par de cada motor es el mismo solo que con un desfase de  $120^\circ$  entre cada uno, por lo que, se eligió solamente utilizar los valores obtenidos del motor 1 para los cálculos posteriores.



*Figura 4-2 Comportamiento del par de los motores con respecto al ángulo  $\beta$ .*

Para la gráfica del motor 1, la demanda del mayor par se encuentra cuando el ángulo  $\beta$  está a  $90^\circ$  o a  $270^\circ$ , dado que el valor del par es el mismo en cualquiera de estos ángulos; el análisis solo se hizo para el ángulo de  $90^\circ$ , sustituyendo este valor en la ecuación 4.5, se obtiene:

$$\frac{\tau_{Mv}}{3 \cos(\alpha)} 2 \text{sen}(90^\circ) = \tau_{m1_{max}} \quad (4.6)$$

Al sustituir  $T_{Mv}$  de la ecuación anterior por el par  $T_{Mv}$  de la ecuación 4.4, se obtiene la siguiente expresión:

$$\tau_{m1_{max}} = \frac{2 W L r_{Rv}}{3 r_e \cos(\alpha)} \text{sen}(\theta) \text{sen}(90^\circ) \quad (4.7)$$

Para realizar la sustitución de las variables en la ecuación anterior, se hicieron las siguientes consideraciones:

- El robot no pesará más de 10 kg
- El radio de la rueda virtual es igual al radio de las ruedas reales
- Se ocupará la misma esfera para el diseño nuevo
- La distancia entre el centro de masa de la esfera y el centro de masa del cuerpo del robot no será mayor a un metro
- El ángulo máximo de inclinación del robot para ser estabilizado será de  $10^\circ$
- Ya que se considera que el ángulo de inclinación de los motores puede variar, el mínimo ángulo de inclinación será de  $20^\circ$  con respecto a la horizontal

Con base en las consideraciones anteriores, se le asignó el valor máximo a cada una de las variables, por lo que, haciendo la sustitución de éstas, la ecuación queda de la siguiente manera:

$$\tau_{m1_{max}} = \frac{2(10 * 9.81)(0.0762)}{3(0.25)\cos(20^\circ)} \text{sen}(10^\circ)\text{sen}(90^\circ)$$

$$\tau_{m1_{max}} = 3.68 \text{ N}\cdot\text{m}$$

Como se puede observar, el par estático máximo que se requiere en cada motor es de 3.68 N·m. Haciendo la consideración de que se requiere por lo menos el doble del par estático para obtener el par dinámico, se necesitarían unos motores con un par de operación de 7.36 N·m para estabilizar al robot con las características antes descritas.

Una vez obtenido el valor del par que se necesitaba en los motores, se realizó una búsqueda de motores de corriente directa con las características requeridas. En la página de internet AndyMark se encontró una gran variedad de motores de corriente directa que podían ser utilizados para este proyecto, la propuesta que más llamó la atención fue el motor con reducción planetaria, ya que dependiendo de la reducción que se utilizara, se podían obtener diferentes salidas tanto de par como de velocidad con un mismo modelo de motor y sin que las dimensiones de la reducción cambiaran más que de su longitud. Dicha característica resultó interesante pues se pensó en que en caso de ser necesario se podrían probar distintas reducciones con el mismo modelo de motor, sin la necesidad de hacer cambios en la estructura.

El motor con características más cercanas a las deseadas fue el PG27 mostrado en la figura 4-5, el cual cuenta con una reducción planetaria, que permitía obtener velocidades de hasta 198 RPM y un par de hasta 8.5 N·m. Además de esto, dicho motor ya tenía un decodificador incluido, el cual podría ser utilizado para el control de estabilidad y navegación.



*Figura 4-3 Motor PG27.*

## Selección de las ruedas

Para seleccionar las nuevas ruedas omnidireccionales, se realizó una búsqueda en el mercado, donde se encontraron dos tipos de rueda omnidireccional, simples y dobles, mostradas en las figuras 4-6 y 4-7. La desventaja principal que se encontró con las ruedas simples fue la discontinuidad que existe entre cada uno de sus rodillos, aunque también se encontraron opciones que no tenían este problema, sin embargo, su precio era muy elevado. Por otro lado, la desventaja que se encontró en las ruedas dobles fue que el punto de contacto con la superficie cambia de una rueda a otra, sin embargo, permiten movimientos más suaves debido a que las dos ruedas evitan los problemas generados por la discontinuidad de los rodillos.



*Figura 4-4 Rueda omnidireccional sencilla.*



*Figura 4-5 Rueda omnidireccional doble.*

Analizando las ventajas y desventajas de cada una de las ruedas, se optó por elegir ruedas dobles para garantizar un movimiento más estable. Asimismo, se decidió que el diámetro de las ruedas fuera de 4", con base en los trabajos realizados tanto en la universidad de Zúrich como en la universidad de Tohoku Gakuin, en donde utilizaron ruedas con diámetros similares. Una vez establecido lo anterior, se adquirieron las ruedas que se muestran en la figura 4-8.



*Figura 4-6 Ruedas omnidireccionales de 4" seleccionadas.*

## Rediseño de la estructura

Una vez seleccionados tanto los motores como las ruedas, la siguiente tarea fue el desarrollo de la nueva estructura; para esto, primero se establecieron los requerimientos que la estructura debería de cumplir, tomando como referencia las características del robot anterior y las nuevas necesidades que surgieron a partir de dar solución al problema de la estabilización. Dichos requerimientos se muestran a continuación:

- Se debe asegurar que el contacto entre las ruedas y la esfera sea tangencial
- El ángulo de inclinación de las ruedas con respecto a la vertical del robot debe considerarse variable
- En la estructura deben caber todos los componentes electrónicos
- En la medida de lo posible, la estructura debe de permitir futuros cambios
- Tamaño máximo del robot no debe sobrepasar los 60 cm de ancho y el metro de longitud
- El robot no debe pesar más de 10 kg
- La estructura debe de soportar tanto el peso de todos los componentes, como las fuerzas generadas por los motores

Posteriormente, con base en las etapas de diseño en la ingeniería, se establecieron las siguientes actividades para el rediseño de la estructura:

- a Analizar cómo sería el acomodo de los actuadores
- b Generar propuestas para el nuevo diseño de la estructura
- c Evaluar y seleccionar la propuesta más conveniente
- d Desarrollar la propuesta seleccionada a detalle
- e Realizar el proceso de manufactura

## Acomodo de los actuadores

Para el rediseño de la estructura, se tuvieron que revisar las limitaciones que se tendrían para la generación de conceptos, estas limitaciones tuvieron que ver con la configuración ya establecida de los actuadores, refiriéndose a que las ruedas debían de ser tangentes a la esfera y estar a  $120^\circ$  entre sí. Puesto que se emplearon los actuadores que ya se tenían, se empezó a revisar la forma en la que estos podían ser acoplados sin cambiar dicha configuración.

Se encontraron solo dos variables que podían ser modificadas para el acomodo de los actuadores, la primera fue el ángulo de inclinación de las ruedas con respecto a la vertical y la segunda fue la orientación de los motores.

En cuanto a la orientación de los motores, solo se encontraron dos formas: una en la que los ejes de los motores estuvieran en dirección al suelo y las ruedas estuvieran por debajo de ellos, tal como se muestra figura 4-9, y la otra en la que los ejes de los motores estuvieran en dirección contraria y las ruedas estuvieran por encima de ellos, como se puede observar en la figura 4-10.

Ya que en el trabajo anterior se sugirió variar el ángulo de inclinación de las ruedas, con el fin de encontrar el óptimo para mejorar la estabilización, se realizaron simulaciones para revisar hasta qué ángulo se podía llegar, probando con las dos orientaciones de los motores. Para estas simulaciones se tomaron en cuenta las dimensiones de los motores, las ruedas, las bridas (*hubs*) y la misma esfera que se utilizó en el robot anterior.

En la primera simulación, se pusieron los motores en su orientación original, es decir, por encima de las ruedas. Al analizar hasta dónde se podía variar el ángulo de inclinación de las ruedas, se observó que este podía ser de hasta  $90^\circ$ , sin embargo, debido a las dimensiones de los motores, el ángulo de inclinación mínimo al que se podía llegar era de  $52^\circ$ , ya que para ángulos menores los motores chocarían entre sí.

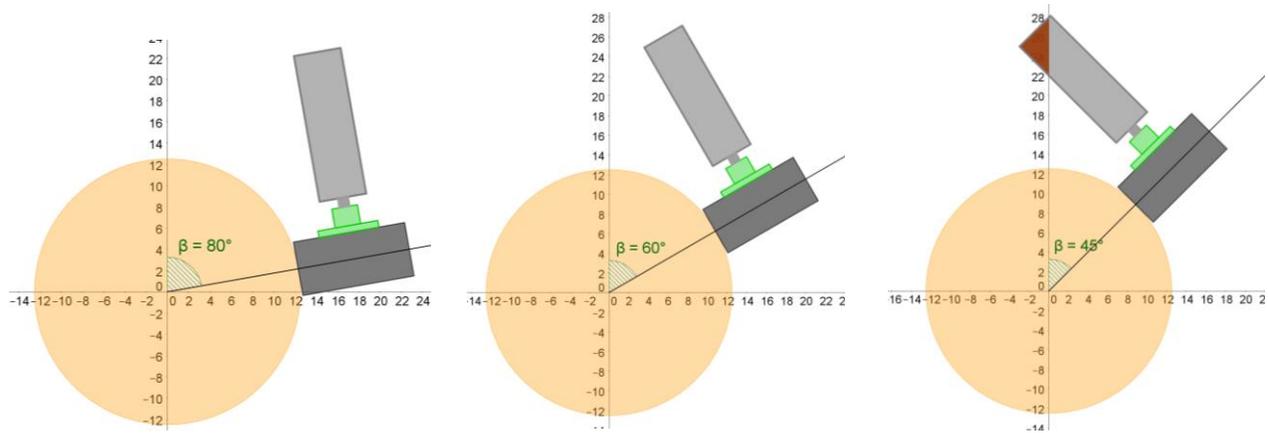
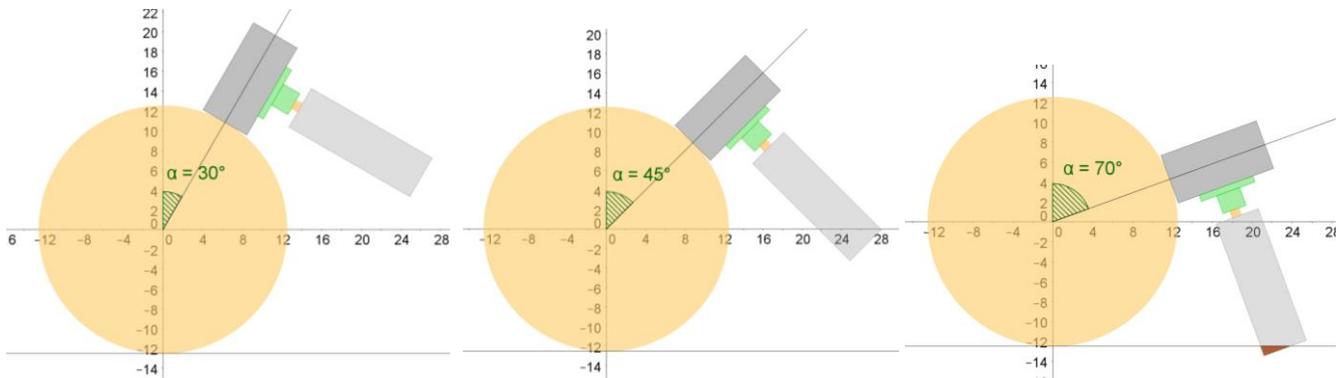


Figura 4-7 Primera configuración de los motores.

La segunda simulación, en la que los motores estaban orientados de forma contraria, mostró que el ángulo de inclinación de las ruedas podía ser menor a los  $30^\circ$ , sin embargo, el ángulo máximo de inclinación no podía ser mayor a los  $70^\circ$  debido a que los motores harían contacto con el piso.



**Figura 4-8 Segunda configuración de los motores.**

Después de analizar las dos simulaciones, se decidió que sería mejor utilizar la orientación en la que los motores estaban con el eje hacia arriba, esto debido a que el rango para variar el ángulo de inclinación en las ruedas con respecto a la vertical era más versátil, es decir, el ángulo podía tomar valores cercanos a los  $45^\circ$ , los cuales eran de mayor interés para el proyecto.

### **Generación de conceptos**

Una vez definidos los requerimientos, el rango en el que el ángulo de las ruedas puede variar y la posición de los motores, se procedió a la tarea de generar conceptos para el diseño de la nueva estructura. Se debe de mencionar que la función crítica de la estructura fue que esta tuviera la capacidad de variar el ángulo de inclinación de las ruedas. Se tuvo la idea de hacer partes intercambiables para variar el ángulo, sin embargo, se llegó a la conclusión de que esto sería ineficiente, por lo que se optó por el diseño de algún mecanismo.

Se generaron cuatro diseños en forma de bocetos los cuales posteriormente se realizaron en un software de CAD, esto con el fin de detallarlos más y tener un mejor entendimiento de los mismos, tanto dimensionalmente como funcionalmente, lo que resultó útil puesto que era complicado imaginarse el acomodo de los actuadores sólo en dos dimensiones.

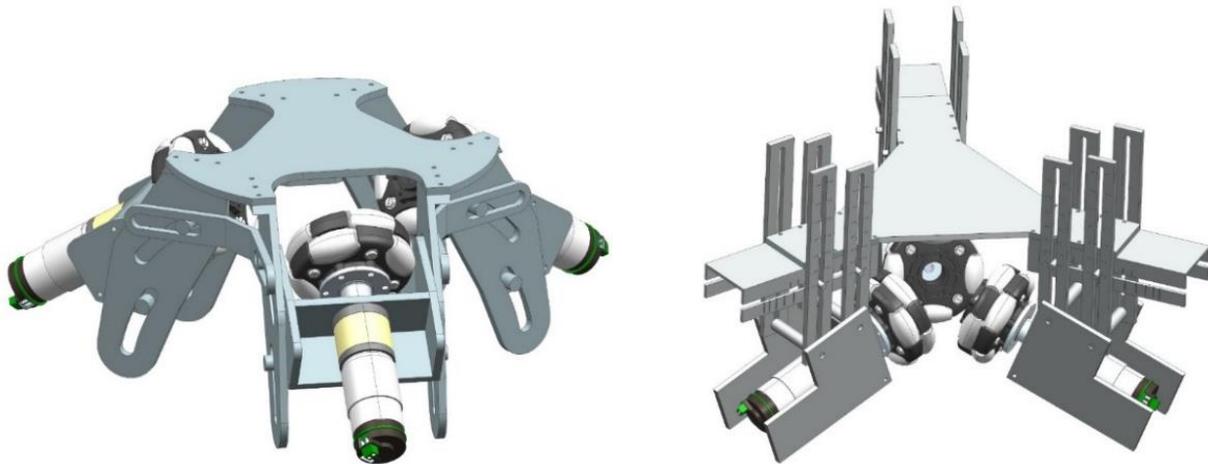
Dentro de los diseños propuestos, se encontraron dos formas en las que se podía variar el ángulo de las ruedas, de manera continua o de manera discreta, es decir, que se tuvieran ángulos ya definidos para posicionar a los motores. Cabe destacar que en cada uno de los diseños se garantizó la tangencia de las ruedas con respecto a la esfera y para lograrlo se realizaron algunos cálculos que se muestran en el apéndice 1.

Se propusieron dos diseños en los que el ángulo de inclinación variaba de manera discreta y en ambos casos se pensó en utilizar orificios para acomodar los motores, tal como se muestra en la figura 4-11. Para mantener la posición de los motores se propuso utilizar dos ejes cilíndricos o un eje de perfil cuadrado.



*Figura 4-9 Diseños con ángulo fijo.*

Asimismo, para variar el ángulo de inclinación de las ruedas de manera continua, se propusieron otros dos diseños, como los mostrados en la figura 4-12. A pesar de que en ambos casos se utilizaron correderas, los conceptos eran distintos; la principal diferencia entre ambos diseños fue la forma en la que se acomodarían los motores; mientras que en una se tenía una trayectoria circular para el acomodo de cada motor, en el otro los motores podían moverse a través de la correderas, en dirección horizontal o vertical. Para los dos casos se propuso graduar las correderas con el fin de saber cuál era el ángulo de inclinación del eje de los motores.

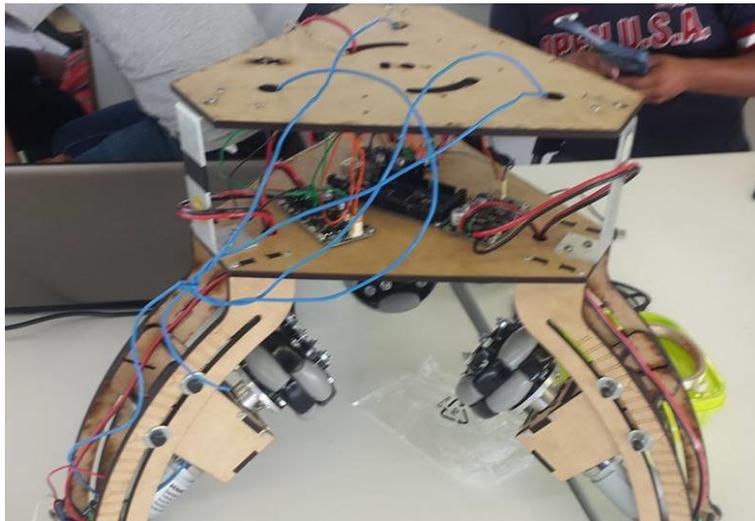


*Figura 4-10 Diseños con ángulo variable.*

### *Evaluación y selección del concepto*

Las propuestas para el rediseño de la estructura fueron evaluadas con base en los criterios de selección que se plantearon en la parte del análisis del problema. Asimismo, las propuestas fueron presentadas en una reunión con algunos profesores de la Facultad de Ingeniería, quienes, con su experiencia en diseño mecánico, dieron sus puntos de vista acerca de las ventajas y desventajas que encontraban en cada propuesta.

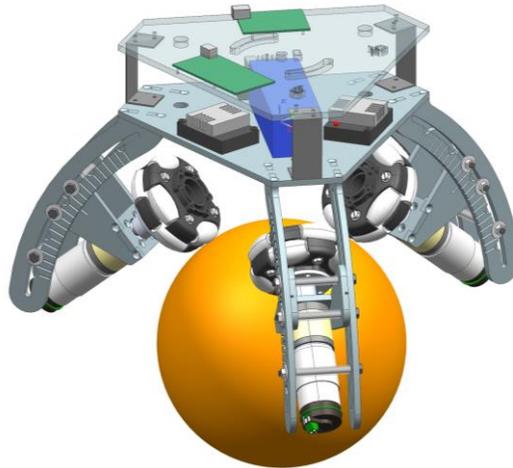
Finalmente, se integró un nuevo diseño, en donde se tomaron las mejores características de cada una de las cuatro propuestas anteriores. Una vez teniendo el CAD del diseño final, se realizó un prototipo en MDF, mostrado en la figura 4-13, con el fin de tener una idea más clara de cómo luciría el diseño. Asimismo, el prototipo sirvió para encontrar posibles mejoras y verificar el correcto funcionamiento del mecanismo para mover el ángulo de inclinación de las ruedas. También se empezó a revisar cómo sería el posible acomodo de los componentes electrónicos.



*Figura 4-11 Prototipo en MDF de la propuesta final.*

### *Diseño a detalle*

Con base en el prototipo de MDF, se realizaron algunas modificaciones en el nuevo diseño, con el fin de reducir material, altura y peso, por lo que finalmente se llegó al diseño que se muestra en la figura 4-14. Posteriormente se empezaron a revisar las especificaciones de dicho diseño.



*Figura 4-12 CAD del diseño final.*

Se decidió que la estructura fuera conformada principalmente con placa de aluminio 6061, ya que este tipo de aleación presenta buena resistencia a la corrosión y es sencillo de maquinarse. Además, este material es usado comúnmente para la fabricación de diversas estructuras, por lo que dicha aleación es de las más versátiles y disponibles en el mercado.

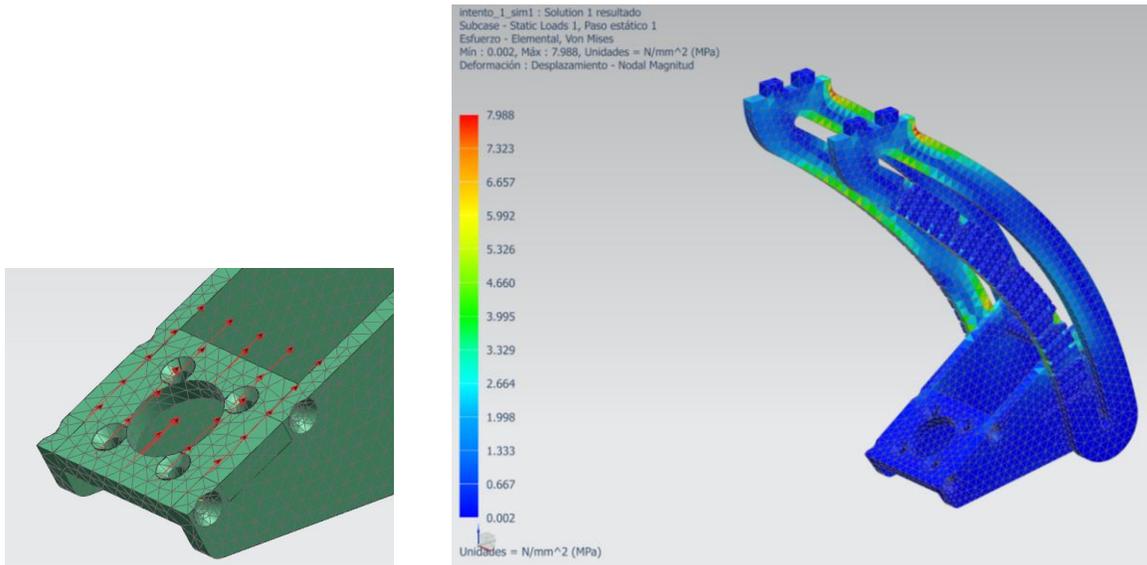
Para ensamblar las piezas de aluminio y sujetar los motores a la estructura, se pensó utilizar tornillos M4 de cabeza plana, tornillos M6, rondanas y tuercas de seguridad para los ejes de la corredera, así como tornillos M3 y tuercas para la unión entre los separadores y los platos. Dichos separadores fueron diseñados para que fueran hechos con lámina de aluminio calibre 14.

Se pensó que el plato superior fuera de acrílico de 6 mm, ya que dicho plato no estaría expuesto a grandes cargas, además de esto se podría reducir el tiempo de manufactura si fuese necesario reemplazarlo o agregar platos extra. Para la parte del cableado, se realizaron algunos orificios en los platos. Para la graduación de la corredera, se pensó que esta fuera grabada con láser.

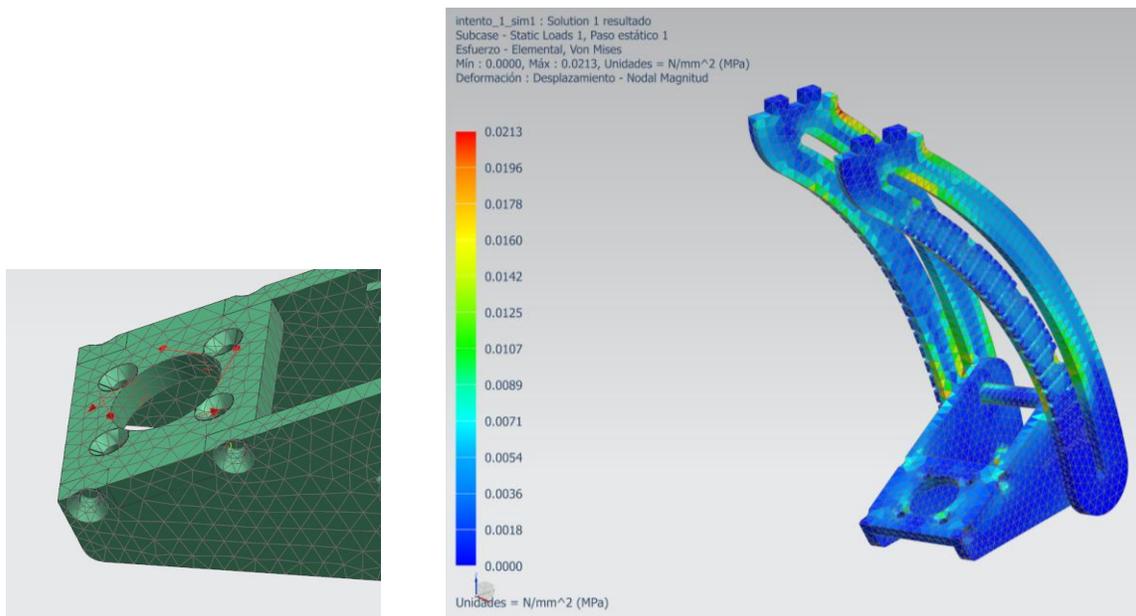
Una vez obtenidas las especificaciones, se realizó un análisis por elemento finito, conocido por sus siglas en inglés como FEM, para verificar que la estructura no sufriría algún daño debido a las fuerzas que actúan en la misma. El análisis FEM se realizó en las correderas, ya que se consideró que en éstas se concentraría la mayor cantidad de esfuerzos y deformaciones. Las cargas que se tomaron en cuenta para el análisis fueron tanto el par de torsión que podría generar el motor acoplado a la corredera, como el peso del mismo robot.

Debido a que la estructura fue diseñada para poder variar el ángulo de inclinación de las ruedas, es importante mencionar que sería complejo, tardado e innecesario realizar un análisis para cada ángulo, por lo que se tomó el ángulo en el que se podría generar mayor cantidad de esfuerzos, que es cuando los motores están acoplados en la zona más baja de la corredera.

Las imágenes de la figura 4-15 y 4-16 muestran el comportamiento de los esfuerzos generados en la corredera. Con el análisis FEM, se observó que el mayor esfuerzo se produce en la parte más alta de la corredera y el mayor desplazamiento se da en la parte más baja de la corredera; asimismo, se observó que dicho desplazamiento es del orden de décimas de milímetro, por lo que se consideró que la estructura no sufriría daños.



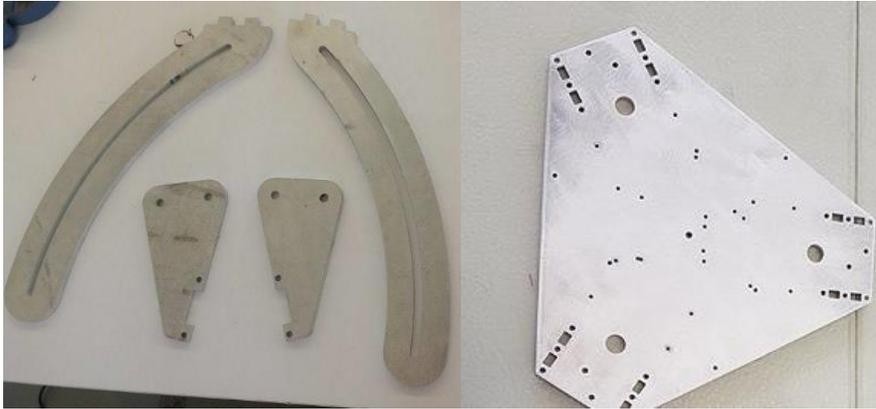
**Figura 4-13** Análisis FEM de los esfuerzos en la corredera generados por el peso del robot.



**Figura 4-14** Análisis FEM de los esfuerzos en la corredera generados por el par en los motores.

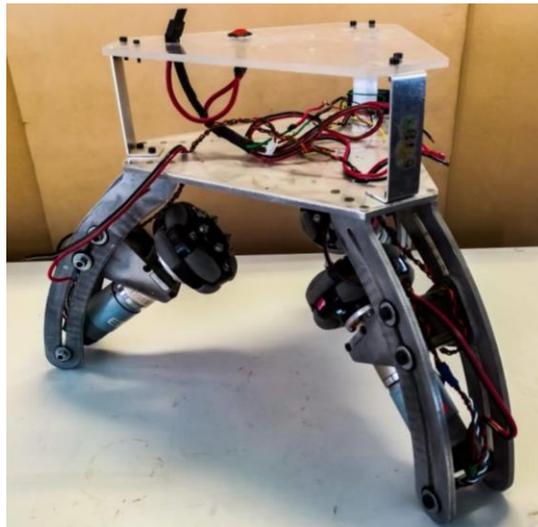
### *Proceso de manufactura*

Para el proceso de manufactura, se decidió que lo más adecuado sería el corte con chorro de agua, sin embargo, para el acabado en las orillas de las piezas y para los barrenos se decidió que el mejor proceso sería el maquinado. Para ambos casos se realizaron los planos de cada una de las piezas. Ya que los procesos anteriormente mencionados ocasionaron rayones en las superficies de las piezas, éstas tuvieron que ser lijadas para mejorar su acabado, su estética y facilitar el grabado de la graduación en las correderas. El resultado del acabado de las piezas se muestra en la figura 4-17.



*Figura 4-15 Piezas del robot después del lijado y maquinado.*

Finalmente se realizó el grabado y el ensamble de las piezas tal y como se muestra en la figura 4-18



*Figura 4-16 Ensamble final.*

### **Búsqueda de soluciones para el rediseño electrónico**

Uno de los problemas que tuvo la versión anterior fue el ruido de la IMU, si bien esto se puede resolver mediante la aplicación de filtros, también se tuvo la idea de mejorar la parte electrónica con el fin de reducir dicho problema. Además, también se tenían que realizar ajustes en dicha parte debido al cambio de motores y a que las señales para controlar el sistema serían diferentes. Por tales motivos, se hicieron las siguientes propuestas para resolver la parte electrónica:

- Adquirir una nueva tarjeta controladora para los nuevos motores de corriente directa.
- Seleccionar los sensores que son necesarios para implementar los algoritmos de control.
- Revisar que las tarjetas de procesamiento que se utilizaron en el robot anterior (Arduino Due y Arduino Uno), sean adecuadas para el nuevo diseño; en caso de no ser así, seleccionar otra para el correcto funcionamiento de la parte lógica.
- Revisar que el suministro de energía es conveniente para alimentar todo el sistema.
- Una vez adquiridos los componentes electrónicos, realizar pruebas de funcionamiento de cada componente por separado y en conjunto. Además, revisar si existen problemas de ruido.

Tal y como se muestra en el análisis del problema del capítulo 3, los criterios para la etapa de selección de los nuevos componentes electrónicos fueron el costo, el tiempo de adquisición, la facilidad de uso y la documentación que se encontraba en la internet acerca de su funcionamiento. Asimismo, es importante mencionar que la principal limitación para la selección fue que debía de existir compatibilidad entre ellos.

## Etapa de potencia

### *Selección de la tarjeta controladora.*

Para la selección de la nueva tarjeta controladora, se tomó como referencia el voltaje de operación de los motores y la corriente máxima. Para el caso de la corriente se consideraron dos casos. El primero fue en el que el robot estaría operando la mayor parte del tiempo a un ángulo cercano al de equilibrio, por lo que el consumo de corriente sería mínimo. El segundo caso fue en el que el robot tendría un ángulo de inclinación muy grande, para el cual el consumo de corriente sería mayor, pero por lapsos de tiempo muy cortos.

Una vez analizados los casos de operación del ballbot, se propuso que la corriente continua mínima que la tarjeta controladora debía suministrar sería de al menos el 50% de la corriente máxima del motor, y que en caso de que el ballbot estuviera en el caso crítico, bastaría con que la tarjeta controladora soportara picos de corriente iguales o mayores a la corriente máxima del motor.

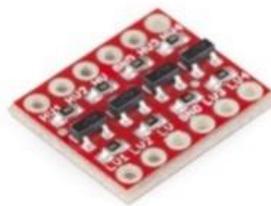
Con base en el análisis anterior, se adquirieron tres tarjetas controladoras de motor Cytron, capaces de suministrar una corriente continua de hasta 13 A, y una corriente pico de 30 A durante un periodo de 10 segundos.



*Figura 5-1 Cytron.*

### *Convertidor de niveles lógicos*

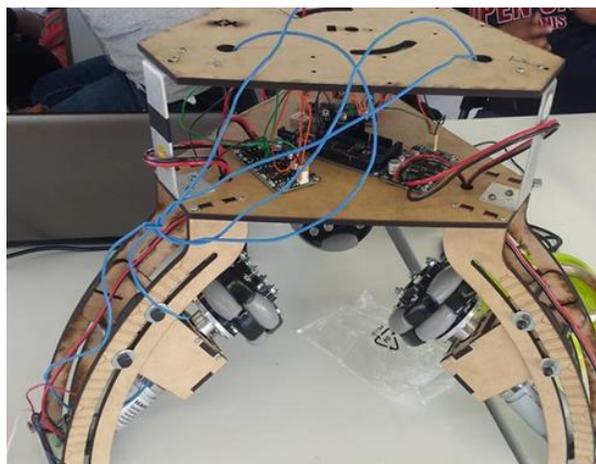
Debido a que el nivel lógico de operación de la placa de desarrollo que se ocupó para el control de los motores (Arduino Due) es de 3.3 V y el de las tarjetas Cytron es de 5 V, fue necesario agregar convertidores de niveles lógicos entre las conexiones de dichos componentes.



*Figura 5-2 Convertidores de niveles lógicos.*

### *Pruebas a la etapa de potencia*

Para comprobar el correcto funcionamiento de la etapa de potencia, se realizaron pruebas de movimiento de los motores de CD en conjunto con las tarjetas controladoras Cytron y los convertidores de niveles lógicos. Para estas pruebas también se utilizó la tarjeta de desarrollo Arduino Due y el prototipo de MDF en el que se montaron todos los componentes. La primera prueba consistió en enviar una señal PWM y una señal de dirección a cada uno de los Cytron. Una vez comprobado que los motores no presentaban problemas para ser controlados mediante una señal PWM, se agregaron las lecturas proporcionadas por los decodificadores, para generar un programa que controlara la velocidad de cada uno de los motores, en la figura 5-3 se muestra la estructura montada para las pruebas.



*Figura 5-3 Estructura de pruebas.*

## Sensor de orientación

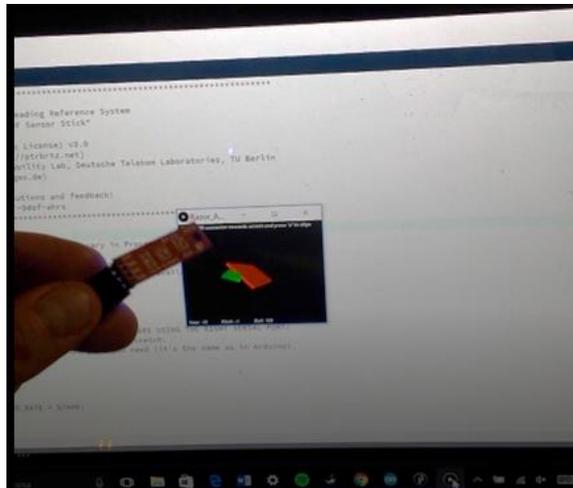
Una IMU o unidad de medición inercial, es un dispositivo que permite determinar la orientación de un objeto mediante el uso de un sistema local. Dicho dispositivo generalmente está conformado de un acelerómetro, un giroscopio y un magnetómetro. El uso de una IMU es sumamente importante en este proyecto pues con esta se puede medir la inclinación del robot con ayuda de los ángulos de Euler (alabeo, cabeceo y guiño) o mejor conocidos como *roll*, *pitch* y *yaw*.

En el proyecto anterior se utilizó una IMU *Sensor Stick*, mostrada en la figura 5-4. Dicho sensor se manejó a una frecuencia de muestreo de 200 Hz y para su comunicación se utilizó el protocolo I2C. Para obtener los datos con el microcontrolador se utilizó una biblioteca la cual se encontraba en la página del fabricante, en dicha biblioteca estaba incluido un filtro complementario; sin embargo, además se utilizó un filtro pasa bajas debido a la cantidad de ruido. A pesar de los filtros, la señal siguió presentando ruido.



*Figura 5-4 IMU sensor stick.*

Además de revisar las características de dicho sensor, se realizaron pruebas con el mismo, en las que se tuvo que conectar a un microcontrolador para leer los datos de la posición angular; posteriormente se enviaron estos datos por comunicación serial a una computadora para desplegarlos de manera gráfica a través de una interfaz, tal y como se muestra en la figura 5-5. En estas pruebas se encontró que la IMU presentó problemas de comunicación con el microcontrolador, ya que después de un cierto periodo de tiempo dejaba de enviar las lecturas de los ángulos. Se cambiaron las conexiones y se realizaron las mismas pruebas con distintos microcontroladores; sin embargo, se presentó el mismo problema. Finalmente se concluyó que era más conveniente cambiar de IMU antes de revisar qué tipo de filtro implementar.



*Figura 5-5 Prueba de funcionamiento de la Sensor Stick.*

### ***Selección de la IMU***

Una vez que se decidió que se debía de cambiar la IMU, se establecieron las características que ésta debía tener para el proyecto; para esto se hizo un análisis del problema de la adquisición de los datos y se concluyó que el sensor debía tener una frecuencia de muestreo alta, además de que la señal de dicho sensor debería de ser lo más limpia posible. Para establecer la frecuencia mínima que debía tener la IMU, se analizó el tiempo de respuesta del motor, el cual era de aproximadamente 60 ms. Con base en la teoría de control, se determinó que era conveniente tener una frecuencia de muestreo de 5 a 10 veces mayor que la respuesta del sistema, por lo que la frecuencia mínima de muestreo que debía tener la IMU era de aproximadamente 100 Hz.

Después de realizar una búsqueda, se encontraron algunas IMU que podrían sustituir a la actual, las características de dichas IMU se describen a continuación.

### **MPU-6050**

La MPU-6050, mostrada en la figura 5-6, es una unidad de medición inercial de seis grados de libertad (6DOF), la cual tiene integrado un acelerómetro y giroscopio, la comunicación para la obtención de sus datos puede ser mediante protocolo SPI o I2C, la tensión de alimentación es entre 2.3 y 3.4 V. Este dispositivo trabaja en conjunto con un microcontrolador para la obtención de los datos y la frecuencia de muestreo de esta IMU puede ser ajustada desde 20 Hz hasta 200 Hz.



*Figura 5-6 Sensor MPU 6050.*

## **UM7-LT**

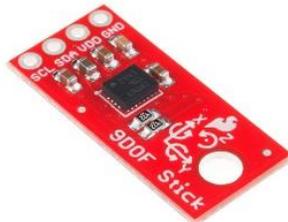
La UM7-LT, mostrada en la figura 5-7, cuenta con un acelerómetro, un giroscopio y un magnetómetro para la obtención de los ángulos *pitch*, *roll* y *yaw*, los cuales son procesados mediante un filtro Kalman extendido; esta IMU hace uso de un microcontrolador integrado a una frecuencia de 500 Hz. La comunicación con un microcontrolador para la obtención de los datos puede realizarse por protocolo TTL, UART o SPI y la frecuencia de muestreo de los datos puede ser ajustada desde 20 Hz hasta 255 Hz.



*Figura 5-7 Sensor UM7-LT.*

## **Sensor Stick 9DOF**

Se encontró una IMU *Sensor Stick* diferente a la que se utilizó en el robot anterior y que se muestra en la figura 5-8. Dicha IMU cuenta con nueve grados de libertad, es decir, tiene integrado un acelerómetro, un giroscopio y un magnetómetro para la obtención de los datos. La comunicación de esta IMU con un microcontrolador puede realizarse por I2C, la tensión de alimentación de 3.3 v y la frecuencia de muestreo es a 150 Hz.



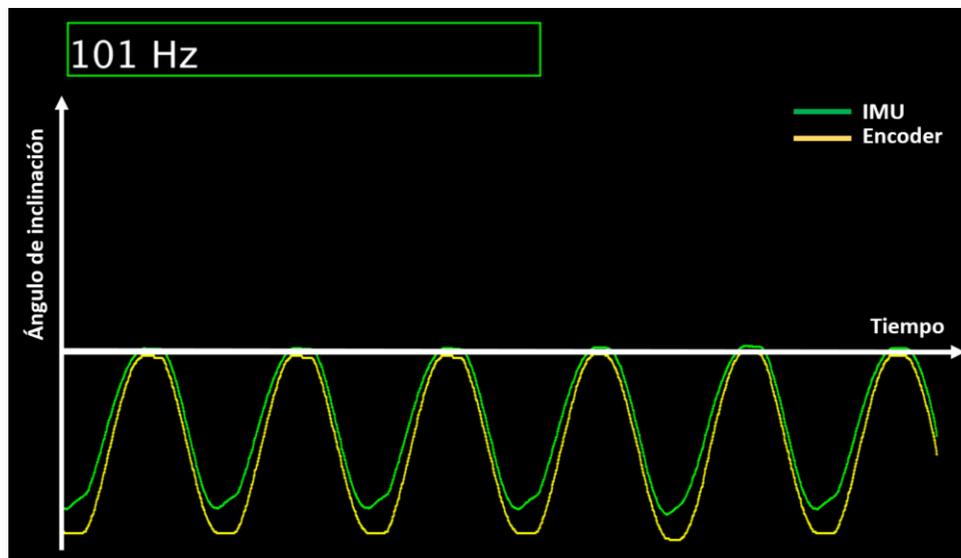
*Figura 5-8 Sensor Stick 9DOF.*

Las IMU que se seleccionaron se consiguieron posteriormente para comparar sus señales mediante una prueba, la cual se describe a continuación. Dicha prueba fue tomada del artículo *Aplicación de una unidad de medición inercial en tiempo real para el control de un ballbot [12]*.

### *Prueba de las IMU*

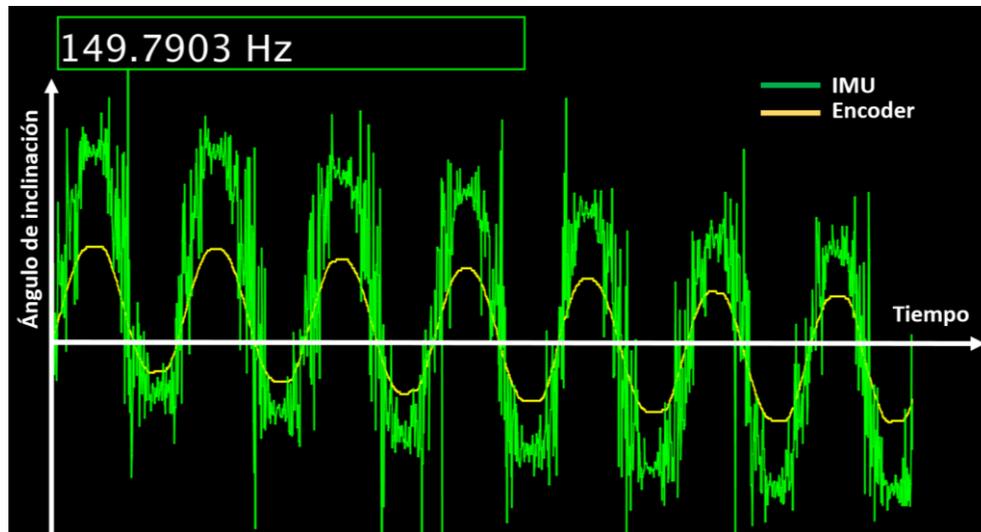
La prueba consiste en simular un ambiente parecido al del ballbot, donde la IMU se acopla al eje de un motor de CD con un decodificador; posteriormente, con la ayuda de un microcontrolador y un puente H, se hace coincidir el cero de la IMU y el cero del decodificador haciendo girar el eje del motor. Finalmente se hace oscilar el motor con una señal sinusoidal y se dibuja la gráfica de la señal del decodificador y la IMU, posteriormente se calcula la frecuencia de muestreo midiendo el tiempo entre un dato y otro.

Una vez realizadas las pruebas se procedió a analizar cada una de las gráficas, la primera de ellas fue la de la MPU-6050, cuya salida se muestra en la figura 5-9. Como se puede observar en la imagen, esta IMU no presentaba tantos problemas de ruido como la IMU anterior; además, haciendo la comparación con las lecturas de los ángulos leídos por los decodificadores, la MPU hacía un buen seguimiento del ángulo de inclinación. Sin embargo, aunque su frecuencia máxima era de 200 Hz, se obtuvieron mejores resultados a 100 Hz, debido a que la señal se perdía si el sensor era utilizado a su máxima frecuencia.



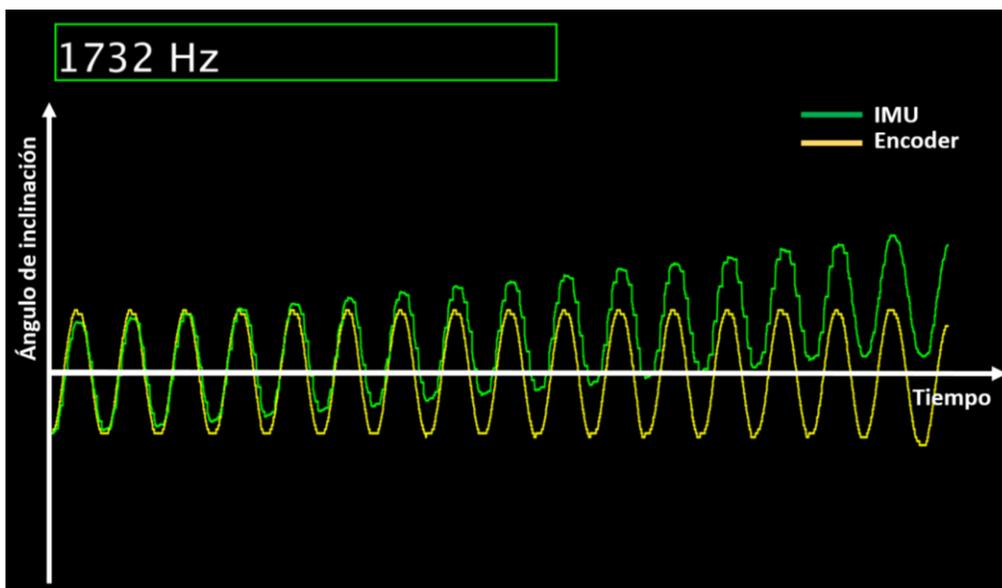
*Figura 5-9 Prueba con la MPU 6050 a 100 Hz.*

La figura 5-10 muestra los datos obtenidos de la *Sensor Stick 9DOF*; esta IMU se operó con una frecuencia de muestreo de 150 Hz. La IMU daba un buen seguimiento del ángulo de inclinación, que era leído por el decodificador, sin embargo, esta IMU presentaba demasiado ruido para la lectura del ángulo, por lo que, en caso de ser seleccionado, tendría que utilizarse un filtro para limpiar su señal.



*Figura 5-10 Prueba con la IMU Sensor Stick 9DOF a 150 Hz.*

Por último, se realizaron pruebas con la UM7-LT; esta IMU se pudo utilizar a una frecuencia de 255 Hz, por lo que fue la que tuvo la mayor frecuencia de muestreo, además de tener una señal sin ruido; sin embargo, esta IMU presentaba una especie de deriva pasado un cierto tiempo, tal y como se muestra en la figura 5-11. Debido a esto, se realizaron pruebas para calibrar el sensor, además de cambiar el microcontrolador para descartar posibles fallas con la placa de control; sin embargo, el problema de deriva siguió presente.



*Figura 5-11 Prueba con la IMU UM7-LT a 255 Hz.*

Con base en los resultados obtenidos en las pruebas, se concluyó que la MPU-6050 era la opción más conveniente para el desarrollo del proyecto.

## Sensores de corriente

En el estudio del estado de la técnica se encontraron trabajos en los que se realizaba un control por par para estabilizar al robot y como se decidió probar un control parecido, se realizó una investigación en donde se encontró que generalmente se utilizan sensores de corriente para hacer un control de par en un motor de corriente directa. Ya que no se tenía la completa certeza de cómo sería el algoritmo de control, se decidió adquirir dichos sensores.

Para elegir los sensores de corriente adecuados, se tomó como referencia el análisis que se realizó para elegir las placas controladoras para los motores, en los que se estimaba la corriente máxima que demandaría el robot; por tal motivo se estableció que los sensores de corriente adecuados tendrían que soportar una corriente pico de hasta 30 A. El sensor de corriente que se eligió para el robot se muestra en la figura 5-12.



*Figura 5-12 Sensor de corriente ACS712.*

## Unidad de procesamiento

En el proyecto anterior se utilizó un Arduino Uno y un Arduino Due para la parte lógica del robot; sin embargo, debido al cambio de componentes electrónicos, se encontró la posibilidad de simplificar esta parte. Por ello, se plantearon las necesidades y los requerimientos que debería de cumplir la placa de desarrollo que se utilizaría.

### Necesidades

- La placa de desarrollo debe ejecutar el algoritmo de control en un corto periodo de tiempo para controlar al robot.
- La placa de desarrollo debe tener preferentemente la capacidad de establecer una comunicación inalámbrica para futuras pruebas de navegación.
- La placa de desarrollo debe ser capaz de recibir todos los datos provenientes de los sensores y al mismo tiempo ser capaz de mandar las señales de control a los actuadores.

## Requerimientos

- 1 Pines requeridos
  - Líneas de alimentación: 3 V, 5 V y GND
  - Pines de entrada analógica: tres para los sensores de corriente
  - Pines de salida digital: tres para los controladores Cytron
  - Pines de salida PWM: tres para los controladores Cytron
  - Pines de comunicación serial: 1 Tx y 1 Rx
  - Pines de interrupción: 6 para los decodificadores
- 2 La placa de desarrollo debe ser capaz preferentemente de comunicarse inalámbricamente; de no ser así, debe de tener la posibilidad de poder integrar un módulo para dicha comunicación
- 3 La velocidad de respuesta del motor para llegar a un estado estable con una respuesta a escalón es de aproximadamente 70 ms. Dicho lo anterior, si se considera que la velocidad de muestreo debe de ser al menos de 5 a 10 veces mayor a la velocidad de respuesta del sistema, la velocidad de muestreo requerida tiene que ser aproximadamente de 100 Hz. Con la IMU ya seleccionada, se hicieron pruebas con distintas placas de desarrollo que se tenían en el taller de robótica para saber cuál debería de ser la velocidad de procesamiento mínima en las placas, para permitir obtener los datos de la IMU a dicha velocidad de muestreo. De manera práctica se concluyó que la placa de desarrollo a utilizar debía tener una frecuencia de operación de 80 MHz

Una vez establecidas las necesidades y los requerimientos, se determinó que con la placa de desarrollo Arduino Due era más que suficiente para conformar la parte lógica del robot.

## Alimentación

Para la alimentación de los componentes electrónicos, se propuso mantener la idea de seguir utilizando una batería tipo Li-Po; sin embargo, debido al cambio de algunos componentes electrónicos, los niveles de voltaje de operación cambiarían, por lo que se tuvo que buscar otra batería. Finalmente se adquirió una batería Li-Po de 3 celdas con una carga de 6400 mAh mostrada en figura 5-13.



*Figura 5-13 Batería Li-Po Turnigy nano-tech 6400mah 3S 40~80C Lipo Pack.*

## Diseño de las conexiones electrónicas

Una vez realizada la etapa de selección de los nuevos componentes electrónicos y habiendo ya sido probados, la siguiente tarea fue realizar las conexiones electrónicas que llevaría el robot, para esto se elaboró un diagrama en el que se describen las conexiones que hay entre cada dispositivo, el cual se muestra en la figura 5-14.

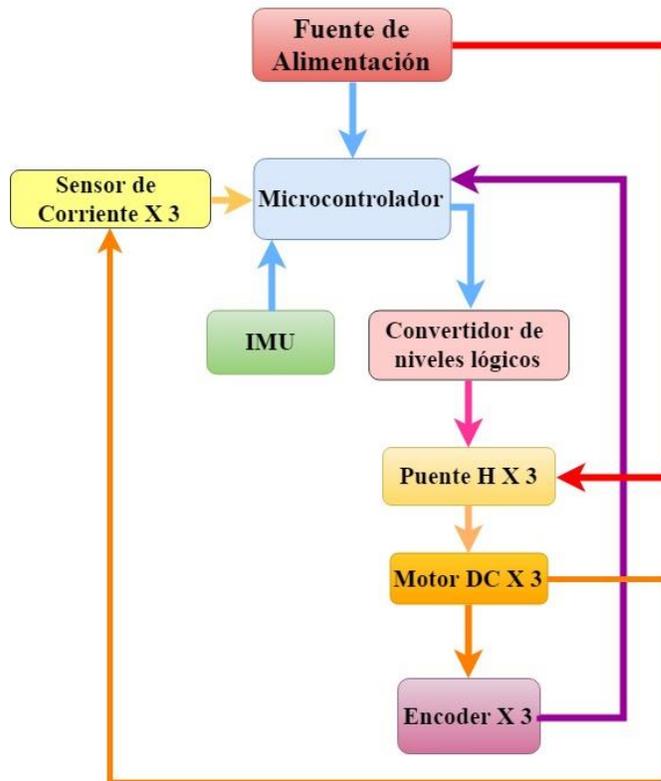


Figura 5-14 Conexiones electrónicas.

Se buscó que las conexiones fueran fáciles de conectar y desconectar, por lo que se decidió realizar una placa central en donde se pudiera montar tanto la placa de desarrollo como los mismos sensores mediante conectores, ver figura 5-15.

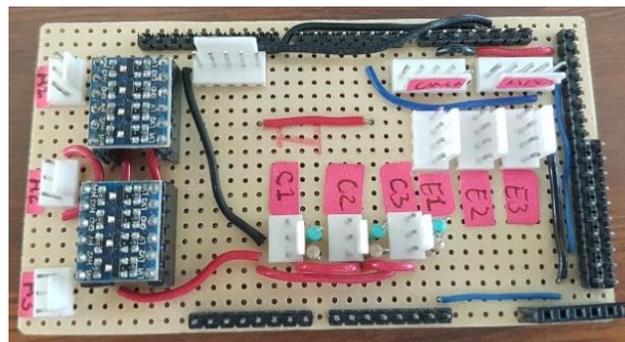


Figura 5-15 Placa electrónica del ballbot.

### Búsqueda de soluciones para el desarrollo de la parte lógica

Una vez realizados los cambios necesarios en los subsistemas mecánico y electrónico, se procedió a desarrollar la parte lógica del robot, para lo que se revisó el estado de la técnica de los algoritmos de control que se han utilizado para este tipo de robots. Se encontró que, en varias universidades donde se habían utilizado motores de corriente directa para mover al robot, se había implementado un algoritmo de control por par para lograr la estabilidad. Para poder implementar un control por par en el robot se propusieron dos formas:

- Realizar un control principal en el que la variable controlada sea el ángulo de inclinación del robot y la variable manipulada sea el par en cada uno de los motores, además de realizar un control secundario en el que la variable controlada sea el par en cada uno de los motores y la variable manipulada sea una señal PWM. Para esto se utilizarían sensores de corriente
- Realizar solo un controlador principal, en el que la variable controlada sea el ángulo de inclinación del robot y la variable manipulada sea una señal PWM en cada uno de los motores

Para ambos casos se debía estudiar el modelo completo del sistema, es decir, tanto el modelo de los motores como el modelo principal del ballbot. Dicho lo anterior, se establecieron las siguientes tareas:

- Revisar el modelo completo del robot
- Diseñar los algoritmos de control con base en el modelo
- Realizar simulaciones e implementar el o los algoritmos de control en el robot

En el proyecto anterior, para el análisis del ballbot se consideró que el robot tenía un comportamiento similar al de un péndulo invertido simple que oscilaba en un plano; esto permitió hacer una simplificación del sistema, en el que el comportamiento general del robot se podía considerar como el de dos péndulos simples localizados en dos planos verticales ortogonales. Dicha simplificación ayudó a poder analizar el comportamiento del robot de manera más sencilla; sin embargo, el comportamiento del sistema en cada plano era visto de manera desacoplada lo cual generaba ciertos errores y además este modelo no consideraba el movimiento de las ruedas para su análisis. Adicionalmente, se estableció en las ecuaciones que relacionan el movimiento de las ruedas con el movimiento deseado en el robot que las ruedas se moverían sobre un plano y no sobre una esfera.

Con base en el análisis anterior, se hizo una búsqueda para proponer un nuevo modelo que considerara las omisiones hechas en el modelo anterior, además de que facilitara el desarrollo del controlador por par.

Se encontró que tanto la universidad de Zúrich como la de Twente realizaron un control por par para lograr la estabilidad de sus robots, además de que utilizaron el modelo de un péndulo invertido para su análisis al igual que en el trabajo anterior; sin embargo, agregaron una rueda virtual para tomar en cuenta el comportamiento de las ruedas en el sistema y, por otra parte, obtuvieron el modelo de sus motores para integrar su comportamiento al modelo principal del robot. A pesar de que en ambas universidades se revisó el modelo en dos dimensiones, debido al error del desacoplamiento antes mencionado, ambas universidades terminaron por analizar el sistema en tres dimensiones para acercarse aún más al modelo al sistema real.

A pesar de que, establecer un modelo en tres dimensiones sería más cercano al sistema real tal y como se menciona en el párrafo anterior, se decidió establecer un modelo en dos dimensiones por cuestión de facilidad; sin embargo, para solucionar el problema del desacoplamiento no se analizó en un sistema cartesiano, sino en un sistema cilíndrico, en donde el plano en el que se analiza el robot está determinado por la dirección en la que se inclina, lo cual se explica con mayor detalle más adelante.

## Modelo del motor

Para establecer el modelo de cada uno de los motores utilizados en el robot, se utilizó el diagrama de un motor de corriente directa simplificado que se muestra en la figura 6-1; asimismo, los significados de las variables se muestran en la tabla 6-1.

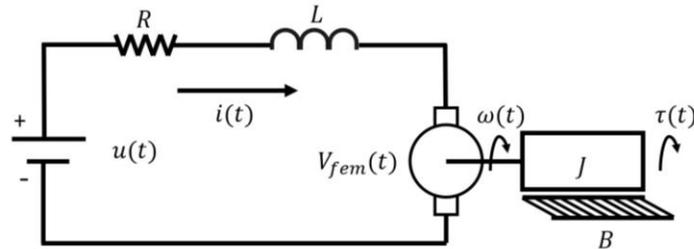


Figura 6-1 Modelo de un motor de CD simplificado.

Tabla 6-1 Significado de las variables para el modelado del motor.

$R$	Resistencia del embobinado del motor
$L$	Inductancia del embobinado del motor
$J$	Momento de Inercia del motor
$B$	Coefficiente de fricción viscosa
$u(t)$	Voltaje de entrada
$\omega(t)$	Velocidad angular del motor
$\tau(t)$	Par del motor
$i(t)$	Corriente del sistema
$V_{fem}(t)$	Voltaje inducido

El análisis de un motor de corriente directa puede ser visto como el de un generador, en el que se utiliza la relación entre la corriente y el par del motor y la relación de la velocidad con la fuerza electromotriz, con el fin de acoplar tanto la parte mecánica como la parte electrónica. A continuación, se muestran las ecuaciones del motor en el dominio del tiempo.

De la parte eléctrica del motor se puede obtener la siguiente ecuación:

$$u(t) = Ri(t) + L \frac{di(t)}{dt} + V_{fem}(t) \quad (6.1)$$

Y de la parte mecánica se puede obtener la siguiente ecuación:

$$\tau(t) = B\omega(t) + J \frac{d\omega(t)}{dt} \quad (6.2)$$

Asimismo, se encuentran las siguientes relaciones:

$$\tau(t) = K_T i(t) \quad (6.3)$$

$$V_{fem}(t) = K_E \omega(t) \quad (6.4)$$

Donde  $K_T$  y  $K_E$  son las constantes mecánica y eléctrica respectivamente.

Las ecuaciones en el dominio de Laplace quedan de la siguiente manera:

$$u(s) = Ri(s) + Lsi(s) + V_{fem}(s) \quad (6.5)$$

$$\tau(s) = B\omega(s) + Js\omega(s) \quad (6.6)$$

$$\tau(s) = K_T i(s) \quad (6.7)$$

$$V_{fem}(s) = K_E \omega(s) \quad (6.8)$$

Con base en las ecuaciones anteriores, se obtuvo la función de transferencia entre el par del motor y el voltaje de entrada.

$$H(s) = \frac{\tau(s)}{u(s)}; H(s) = K_T \left( \frac{Js+B}{JLs^2+(JR+LB)s+(BR+K_T K_E)} \right) \quad (6.9)$$

Como se puede observar, la función de transferencia queda de segundo orden. Se puede simplificar el análisis anterior si se considera que la dinámica de la parte eléctrica se desarrolla de manera mucho más rápida que la dinámica de la parte mecánica; de esta forma se puede considerar que la corriente se vuelve constante en un tiempo muy pequeño, por lo que, tomando esta consideración, la función de transferencia queda de la siguiente forma:

$$H(s) = \frac{K_T}{R} \left( \frac{s+\frac{B}{J}}{s+\left(\frac{BR+K_T K_E}{JR}\right)} \right) = K_g \left( \frac{s+\alpha}{s+\beta} \right) \quad (6.10)$$

donde  $K_g = \frac{K_T}{R}$ ;  $\alpha = \frac{B}{J}$ ;  $\beta = \frac{BR+K_T K_E}{JR}$

## Modelo principal del ballbot

Es importante mencionar que el modelo que se muestra a continuación es una simplificación del modelo en 2D que se desarrolló en la universidad de Zúrich [3]. Es una simplificación pues en este modelo no se considera el desplazamiento angular en el plano vertical.

Para entender de manera adecuada el modelo principal del ballbot, se presenta a continuación el diagrama del sistema mostrado en la figura 6-2 y la tabla 6-2 con el significado de cada variable. El análisis se realizó solo para un plano vertical, el cual por facilidad de nomenclatura fue el plano YZ; sin embargo, el análisis es el mismo para cualquier plano vertical en el que se incline el robot.

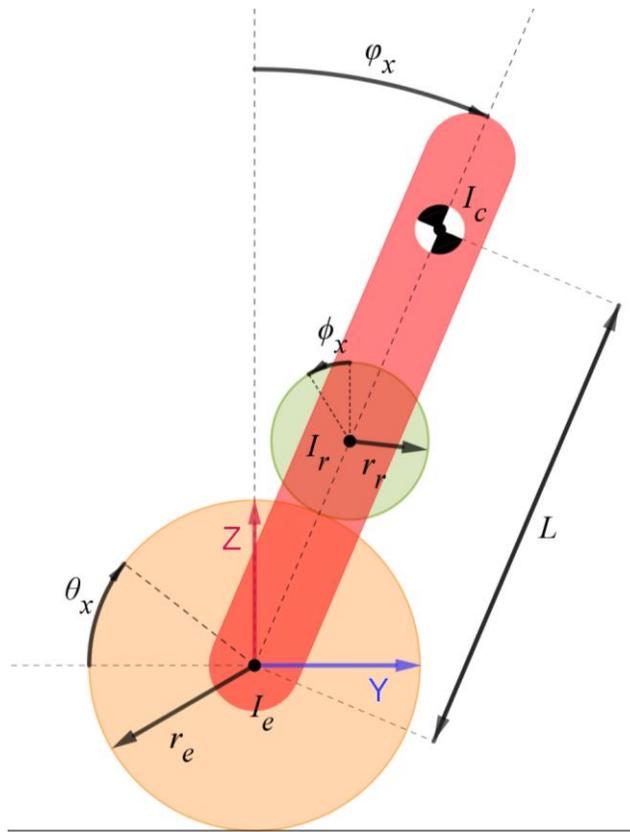


Figura 6-2 Simplificación del modelo del sistema.

Tabla 6-2 Significado de las variables para el modelado del sistema.

$\theta_x$	Ángulo de desplazamiento de la esfera
$\varphi_x$	Ángulo de desplazamiento del robot
$\phi_x$	Ángulo de desplazamiento de la rueda virtual
$I_c$	Momento de inercia del cuerpo del robot
$I_r$	Momento de inercia de la rueda virtual
$I_e$	Momento de inercia de la pelota
$r_r$	Radio de la rueda virtual
$r_e$	Radio de la esfera
$L$	Distancia entre el centro de masa de la esfera y el centro de masa del robot

En la figura 6-2 se presentan tres partes que conforman al robot: la esfera, el cuerpo del robot y finalmente una rueda virtual, la cual representa a los tres actuadores del robot.

Para obtener las ecuaciones que definen el comportamiento del sistema, se establecen las coordenadas generalizadas, las cuales en este caso son tanto el desplazamiento angular del cuerpo del robot como el desplazamiento angular de la esfera; dichas coordenadas se muestran a continuación:

$$\bar{q} = \begin{bmatrix} \theta_x \\ \varphi_x \end{bmatrix} \quad (6.11)$$

donde

Para obtener las ecuaciones de movimiento, se utiliza el método de Lagrange, el cual consiste de los siguientes pasos:

- Expresar tanto la energía cinética  $T$ , como la energía potencial  $V$  de todos los cuerpos rígidos, en función de las coordenadas generalizadas
- Expresar todos los momentos externos (o no potenciales)  $\overline{\tau_{ext}}$  en función de las coordenadas generalizadas
- Definir el lagrangiano  $\mathcal{L}$  como  $\mathcal{L}(\bar{q}, \dot{\bar{q}}) = T - V$  (donde  $q$  es el vector de coordenadas generalizadas)
- Calcular las ecuaciones de Euler-Lagrange  $\frac{d}{dt} \left( \frac{\delta \mathcal{L}}{\delta \dot{q}_i} \right) - \frac{\delta \mathcal{L}}{\delta q_i} + \frac{\delta D}{\delta \dot{q}_i} = \overline{\tau_{ext}}$  donde:  
 $i = 1, \dots, n$  y  $n$  es la longitud del vector  $\bar{q}$ . El resultado muestra las ecuaciones de movimiento

### ***Energías del sistema***

Se reitera que el análisis que se hace es únicamente para un plano vertical del sistema, en este caso el YZ, por lo que las ecuaciones de las energías de los cuerpos rígidos del sistema que se muestran a continuación son sólo para dicho plano. El desarrollo completo de dichas ecuaciones se encuentra en el apéndice 2.

- Para la esfera:

$$T_e = \frac{1}{2} m_e r_e^2 \dot{\theta}_x^2 + \frac{1}{2} I_e \dot{\theta}_x^2 \quad (6.12)$$

$$V_e = 0 \quad (6.13)$$

donde  $T_e$  es la energía cinética de la esfera,  $m_e$  es la masa de la esfera y  $V_e$  es la energía potencial de la esfera.

- Para el cuerpo del robot:

$$T_c = \frac{1}{2}m_c \left( r_e^2 \dot{\theta}_x^2 + 2r_e L \dot{\theta}_x \dot{\phi}_x \cos \varphi_x \right) + \frac{1}{2}(I_{co})\dot{\phi}_x^2 \quad (6.14)$$

$$V_c = m_c g L \cos \varphi_x \quad (6.15)$$

donde  $T_c$  es la energía cinética del cuerpo del robot,  $m_c$  es la masa del cuerpo del robot,  $I_{co}$  es el momento de inercia del cuerpo del robot en el centro de la esfera,  $g$  la constante de gravedad y  $V_c$  es la energía potencial del cuerpo del robot.

- Para la rueda virtual

$$T_r = \frac{1}{2}m_r \left( r_e^2 \dot{\theta}_x^2 + 2r_e(r_e + r_r)\dot{\theta}_x \dot{\phi}_x \cos \varphi_x + (r_e + r_r)^2 \dot{\phi}_x^2 \right) + \frac{1}{2}I_r \left( \frac{r_e}{r_r}(\dot{\theta}_x - \dot{\phi}_x) \right)^2 \quad (6.16)$$

$$V_r = m_r g (r_r + r_e) \cos \varphi_x \quad (6.17)$$

donde  $T_r$  es la energía cinética de la rueda,  $m_r$  es la masa de la rueda y  $V_r$  es la energía potencial de la rueda.

### ***Entrada del sistema***

En el modelo, la única entrada que recibe el sistema para el plano analizado YZ es el componente en X del par del actuador virtual, el cual transmite el movimiento a una rueda virtual; dicho par es llamado  $\tau$ .

Usando la ecuación A.27 que se muestra en el apéndice 2, la cual establece la relación que existe entre el ángulo  $\phi_x$  y las coordenadas generalizadas  $\theta_x$  y  $\varphi_x$ ; se puede determinar un jacobiano geométrico, el cual puede ser utilizado para obtener una relación entre el par de entrada  $\tau$  y los pares que controlan directamente a las coordenadas generalizadas, es decir  $\tau_{ext}$ . Dicho jacobiano se muestra a continuación:

$$\dot{\phi}_x = \begin{bmatrix} r_e & -r_e \\ r_r & r_r \end{bmatrix} \begin{bmatrix} \dot{\theta}_x \\ \dot{\phi}_x \end{bmatrix} \quad (6.18)$$

$$J = \begin{bmatrix} r_e & -r_e \\ r_r & r_r \end{bmatrix} \quad (6.19)$$

Entonces el par de entrada, puede ser visto de la siguiente manera:

$$\overline{\tau_{ext}} = J^T \tau \quad (6.20)$$

$$\overline{\tau_{ext}} = \begin{bmatrix} \frac{r_e}{r_r} \\ \frac{r_e}{r_r} \\ -\frac{r_e}{r_r} \end{bmatrix} \tau \quad (6.21)$$

### *Ecuaciones de Lagrange*

Teniendo ya las energías tanto cinética como potencial del sistema y la entrada al sistema, se puede obtener el lagrangiano:

$$\mathcal{L} = T - V \quad (6.22)$$

donde:

$$T = T_e + T_c + T_r \quad (6.23)$$

$$V = V_e + V_c + V_r \quad (6.24)$$

Es importante mencionar que las ecuaciones de Euler-Lagrange pueden ser descritas como una ecuación matricial, la cual se muestra a continuación:

$$M(\bar{q})\ddot{\bar{q}} + C(\bar{q}, \dot{\bar{q}})\dot{\bar{q}} + K(\bar{q}) = \overline{\tau_{ext}} \quad (6.25)$$

donde las matrices  $M$ ,  $C$  y  $K$  que se obtienen del sistema son:

$$M(\bar{q}) = \begin{bmatrix} I_e + r_e^2 m_{total} + I_r \frac{r_e^2}{r_r^2} & r_e \lambda \cos \varphi_x - I_r \frac{r_e^2}{r_r^2} \\ r_e \lambda \cos \varphi_x - I_r \frac{r_e^2}{r_r^2} & I_{co} + I_r \frac{r_e^2}{r_r^2} + m_r r_{total}^2 \end{bmatrix} \quad (6.28)$$

$$C(\bar{q}, \dot{\bar{q}}) = \begin{bmatrix} 0 & -r_e \lambda \dot{\varphi} \sin \varphi_x \\ 0 & 0 \end{bmatrix} \quad (6.29)$$

$$K(\bar{q}) = \begin{bmatrix} 0 \\ -g \lambda \sin \varphi_x \end{bmatrix} \quad (6.30)$$

donde:

$$m_{total} = m_e + m_c + m_r \quad (6.31)$$

$$r_{total} = r_e + r_r \quad (6.32)$$

$$\lambda = m_r r_{total} + m_c L \quad (6.33)$$

La solución de la ecuación se obtiene despejando a  $\ddot{\bar{q}}$  (matriz que representa las aceleraciones  $\ddot{\theta}_x$  y  $\ddot{\varphi}_x$ ):

$$\ddot{\bar{q}} = M(\bar{q})^{-1} \tau_{externo} - M(\bar{q})^{-1} C(\bar{q}, \dot{\bar{q}}) \dot{\bar{q}} - K(\bar{q}) \quad (6.34)$$

El despeje se desarrolló con el software Mathematica. Posteriormente, el sistema se linealizó mediante series de Taylor de donde se obtiene un nuevo sistema, el cual tiene un comportamiento parecido al sistema original siempre y cuando sea utilizado cerca del punto en el que se evaluaron las series de Taylor, que para este caso fue el punto de equilibrio.

El sistema lineal entonces puede ser representado de la siguiente manera:

$$\dot{\bar{x}} = A\bar{x} + Bu \quad (6.35)$$

$$\bar{y} = C\bar{x} \quad (6.36)$$

donde se consideró que:

$$\bar{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}; \bar{y} = \begin{bmatrix} \theta \\ \varphi \\ \dot{\varphi} \end{bmatrix}; u = \tau \quad (6.37)$$

donde las matrices A, B y C se pueden obtener de la siguiente manera:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \left. \frac{\partial \ddot{q}_1}{\partial \theta} \right]_{Pe} & \left. \frac{\partial \ddot{q}_1}{\partial \dot{\theta}} \right]_{Pe} & \left. \frac{\partial \ddot{q}_1}{\partial \varphi} \right]_{Pe} & \left. \frac{\partial \ddot{q}_1}{\partial \dot{\varphi}} \right]_{Pe} \\ 0 & 0 & 0 & 1 \\ \left. \frac{\partial \ddot{q}_2}{\partial \theta} \right]_{Pe} & \left. \frac{\partial \ddot{q}_2}{\partial \dot{\theta}} \right]_{Pe} & \left. \frac{\partial \ddot{q}_2}{\partial \varphi} \right]_{Pe} & \left. \frac{\partial \ddot{q}_2}{\partial \dot{\varphi}} \right]_{Pe} \end{bmatrix} \quad (6.38)$$

$$B = \begin{bmatrix} 0 \\ \left. \frac{\partial \ddot{q}_1}{\partial u} \right]_{Pe} \\ 0 \\ \left. \frac{\partial \ddot{q}_2}{\partial u} \right]_{Pe} \end{bmatrix} \quad (6.39)$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.40)$$

El desarrollo de todo el modelado se realizó con Mathematica, en el que todo se dejó en función de los parámetros del sistema.

## Modelo completo del sistema

Una vez analizando tanto el modelo del motor como el modelo principal del ballbot, se evaluó cuál de las dos formas, que se mencionaron al principio del capítulo, se utilizaría para realizar un control por par, si haciendo un control principal para el ángulo de inclinación del robot y un control secundario para el par de los motores, o solo con un control general para la inclinación del robot.

Se determinó que, si se quería realizar un control general, se tendría que sustituir el par de entrada del sistema principal del ballbot por la siguiente ecuación:

$$\tau(t) = k_T(u(t) - k_E\omega(t)) \quad (6.41)$$

En donde  $\tau(t)$  sería el par entregado por el motor virtual,  $u(t)$  su voltaje de entrada y  $\omega(t)$  su velocidad angular. No se encontró de qué forma se podría determinar la relación del voltaje de entrada del motor virtual del modelo con el voltaje de entrada de los motores reales del sistema. Sin embargo, en el trabajo realizado en la universidad de Zúrich [3] sí se hablaba de la relación que existía entre el par del motor virtual y el par de los tres motores reales, por lo que se decidió hacer el control secundario de par para cada motor y posteriormente implementar un control principal para el ángulo de inclinación.

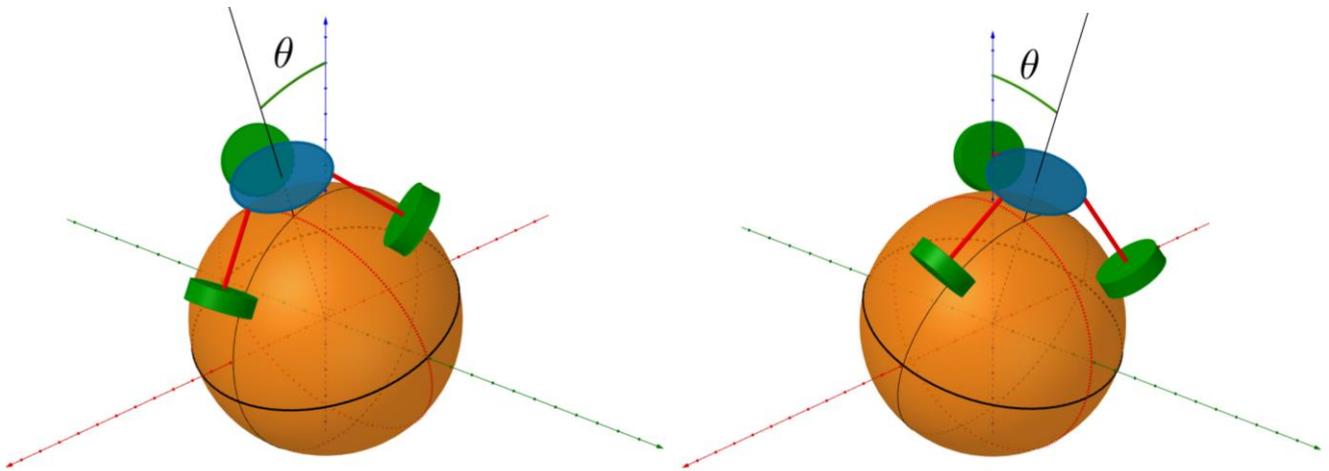
Por otro lado, se pensó en otra forma de controlar al robot utilizando una relación de velocidad, la cual determinaría la velocidad que debería de tener cada uno de los tres motores para hacer que el robot regresara a su posición inicial. Dicha relación sería utilizada para determinar el voltaje que debería de tener cada uno de los tres motores al momento de realizar el control.

Tanto la relación que existe entre el par del motor virtual del modelo con el par de los tres motores reales, como la relación que existe entre la velocidad de las tres ruedas con la dirección del movimiento generado en el robot, se describen en el siguiente apartado del capítulo. Para esto se tuvo que revisar primero de qué forma se determinaría el plano del modelo en el sistema real y el ángulo de inclinación del robot con ayuda de la IMU.

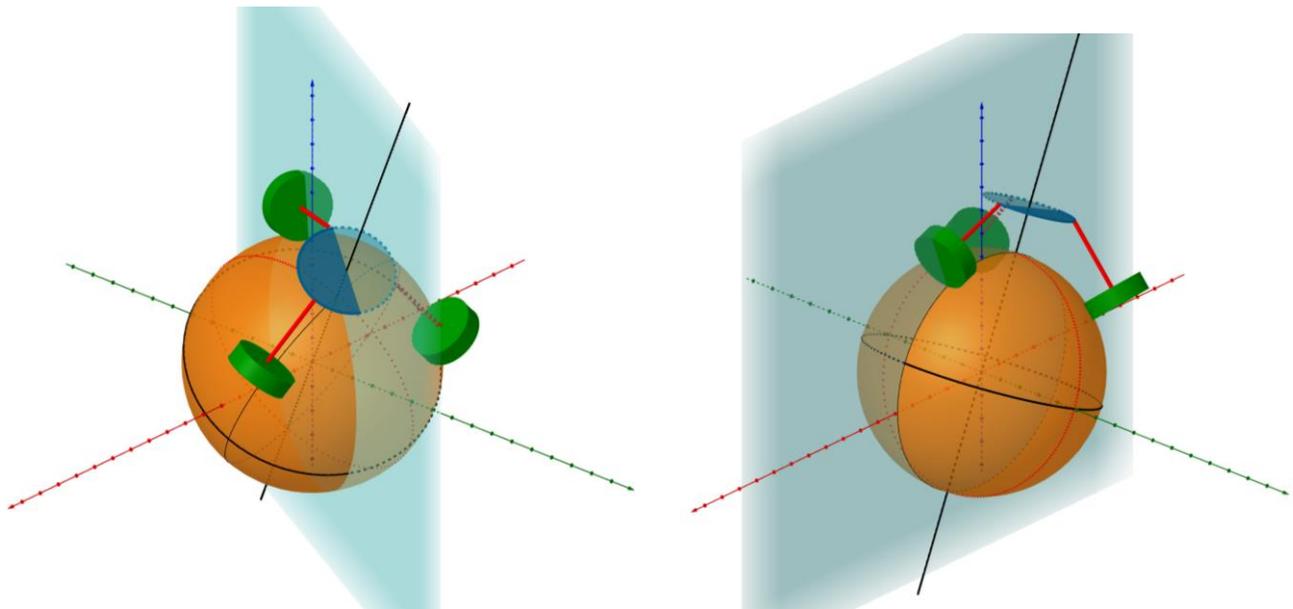
## Relación entre el par del modelo y el par de las tres ruedas

### *Ángulo de inclinación y ángulo del plano de inclinación*

Para obtener la relación que existe entre el par de las tres ruedas del robot con el par que se tiene en el modelo, primero se deben de determinar dos ángulos, los cuales ayudan a describir el comportamiento del movimiento del robot, el primero para definir qué tanto se inclina el robot, ver figura 6-3, y el segundo para definir en qué plano vertical se está inclinando el robot, ver figura 6-4.



*Figura 6-3 Representación del ángulo de inclinación del robot.*



*Figura 6-4 Representación del plano vertical en donde se inclina el robot (mostrado en azul).*

Para determinar dichos ángulos se deben de definir tres sistemas de referencia. El primero es el sistema global, ver figura 6-5, el segundo es el sistema local determinado por la IMU, ver figura 6-6 y finalmente el sistema local del plano de inclinación del robot, ver figura 6-7.

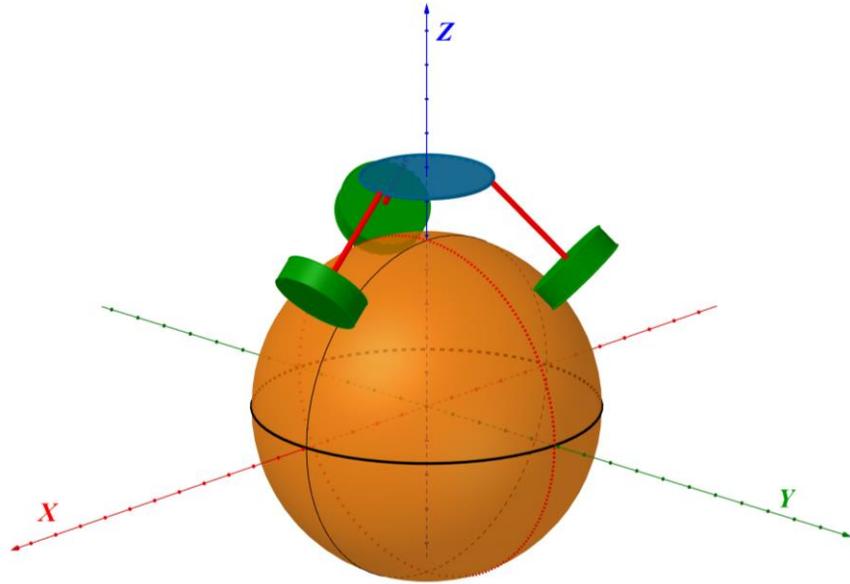


Figura 6-5 Representación simplificada del robot en 3D, los ejes que se muestran representan al sistema global.

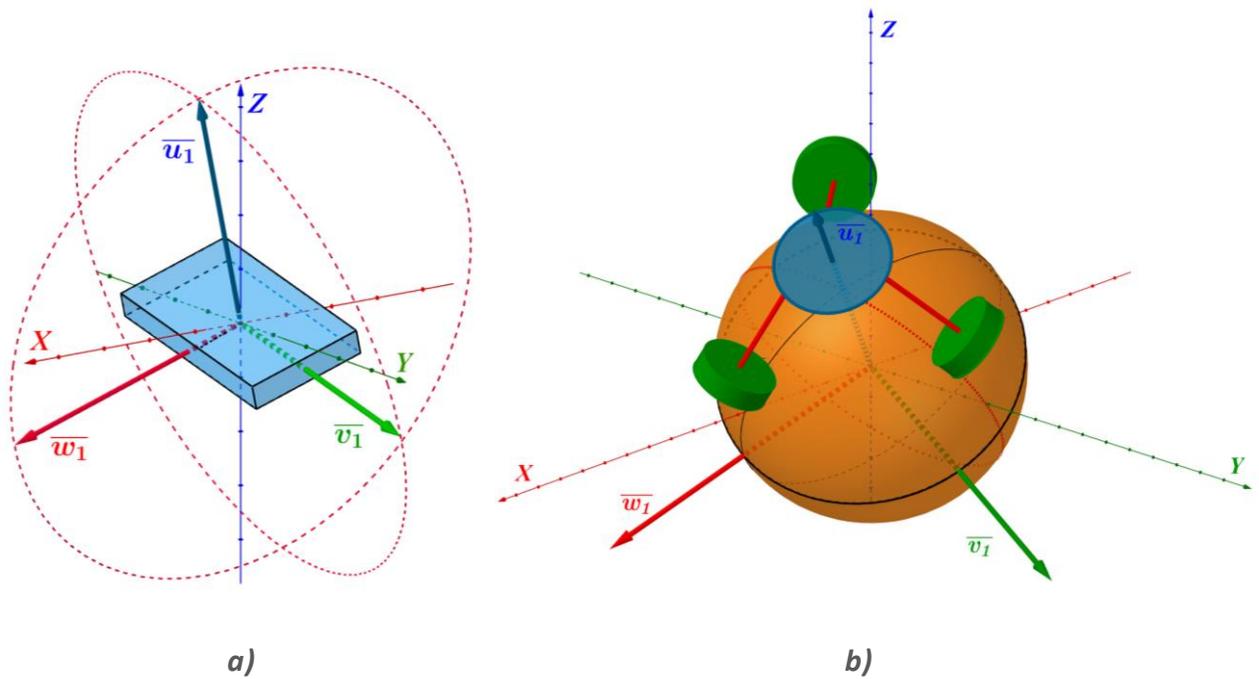
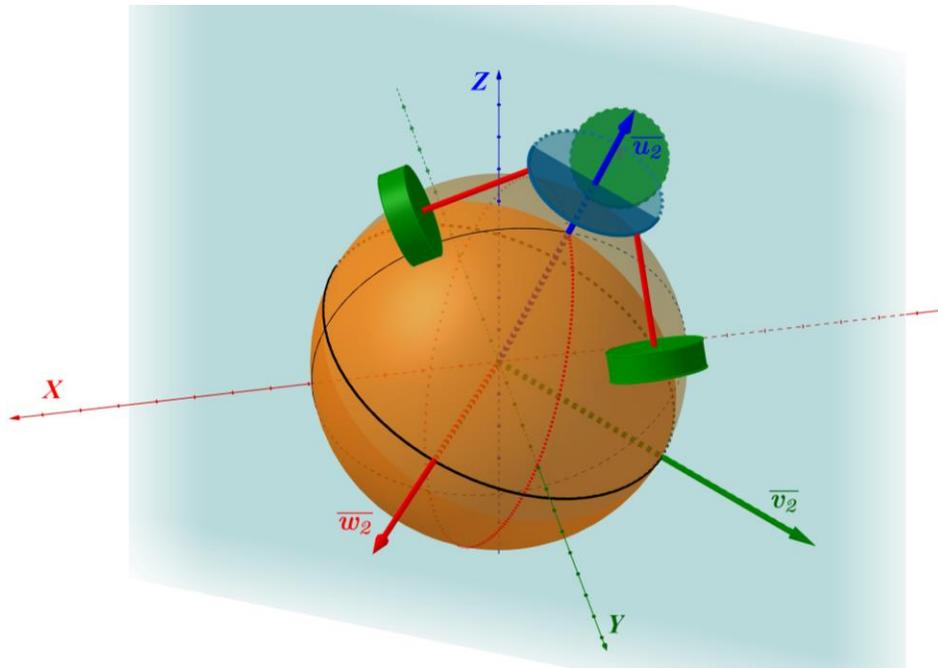


Figura 6-6 Sistema de referencia determinado por la posición de la IMU, representado por los vectores  $\overline{w}_1$ ,  $\overline{v}_1$  y  $\overline{u}_1$ . Es importante mencionar que la IMU está alineada con el eje central del robot y con una de sus ruedas. Considerando solo los ángulos, el sistema de a) es el mismo sistema de b).



*Figura 6-7 Sistema de referencia determinado por el plano en el que se inclina el robot, representado por los vectores  $\overline{w}_2$ ,  $\overline{v}_2$  y  $\overline{u}_2$ . En dicho sistema el vector  $\overline{w}_2$  es un vector normal a dicho plano y el vector  $\overline{u}_2$  se encuentra alineado con el eje central del cuerpo del robot.*

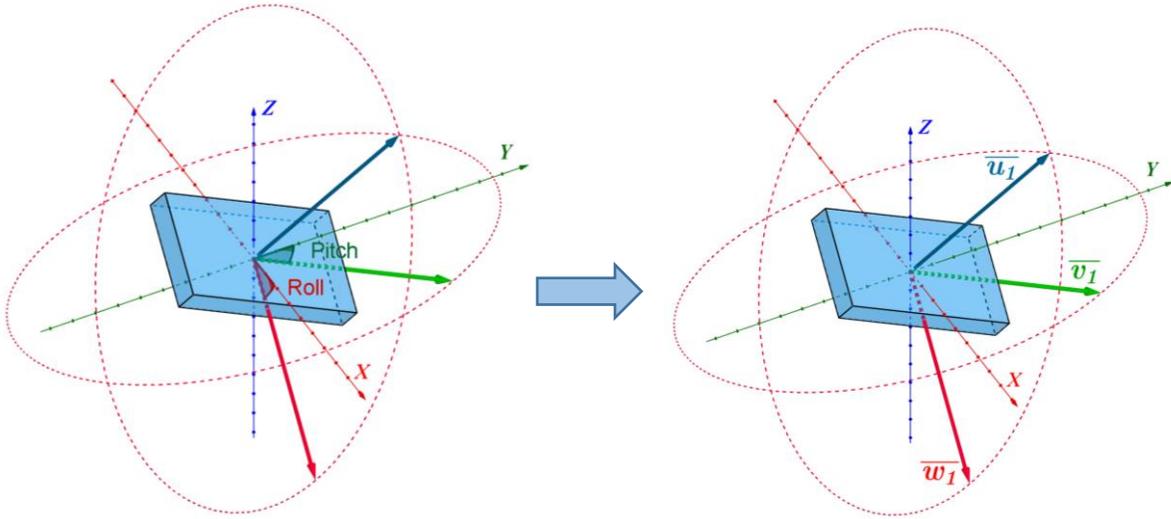
Para obtener los ángulos que se mencionaron previamente, se utiliza tanto el ángulo *pitch* como el ángulo *roll* de la IMU, de los cuales se pueden sacar tres vectores ortogonales que definen al sistema local de la IMU, mostrado en la figura 6-8. Estos vectores se expresan de la siguiente manera:

$$\overline{w}_1 = (\cos \gamma, 0, \sin \gamma) \quad (6.42)$$

$$\overline{v}_1 = (0, \cos \delta, \sin \delta) \quad (6.43)$$

$$\overline{u}_1 = \overline{w}_1 \times \overline{v}_1 \quad (6.44)$$

donde  $\gamma$  es el ángulo *roll*, y  $\delta$  el ángulo *pitch*.



**Figura 6-8 Vectores unitarios que representan al sistema local de la IMU.**

Teniendo el vector  $\overline{u}_1$  se puede determinar el ángulo de inclinación del robot, que se muestra en la figura 6-9, el cual se expresa de la siguiente manera:

$$\theta = \arccos \frac{\overline{u}_1 \cdot \overline{u}_z}{|\overline{u}_1| |\overline{u}_z|} \quad (6.45)$$

donde  $\overline{u}_z$  es el vector unitario que va en dirección al eje Z del sistema global.

Asimismo, se puede obtener el ángulo que existe entre el vector  $\overline{w}_1$  y el plano en el que se inclina el robot, los cuales se pueden apreciar en la figura 6-10; para esto primero se obtiene un vector normal a dicho plano:

$$\overline{n}_i = \overline{u}_1 \times \overline{u}_z \quad (6.46)$$

por lo que dicho ángulo se define de la siguiente manera:

$$\beta = \arcsen \frac{\overline{n}_i \cdot \overline{w}_1}{|\overline{n}_i| |\overline{w}_1|} \quad (6.47)$$

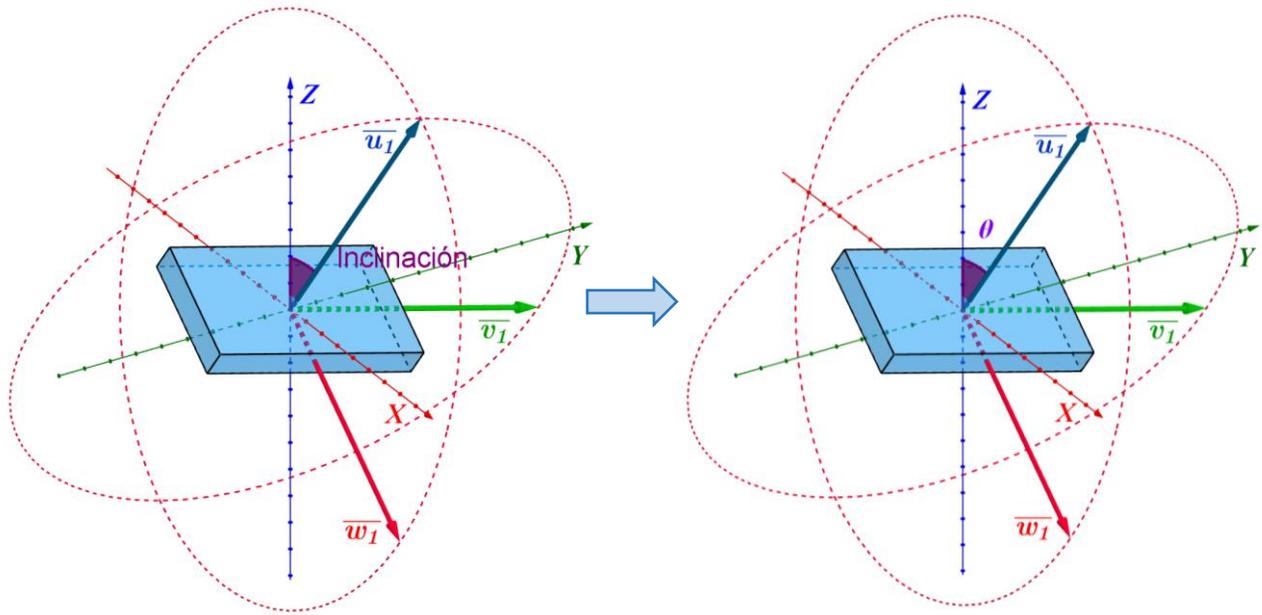


Figura 6-9 Ángulo de inclinación del robot.

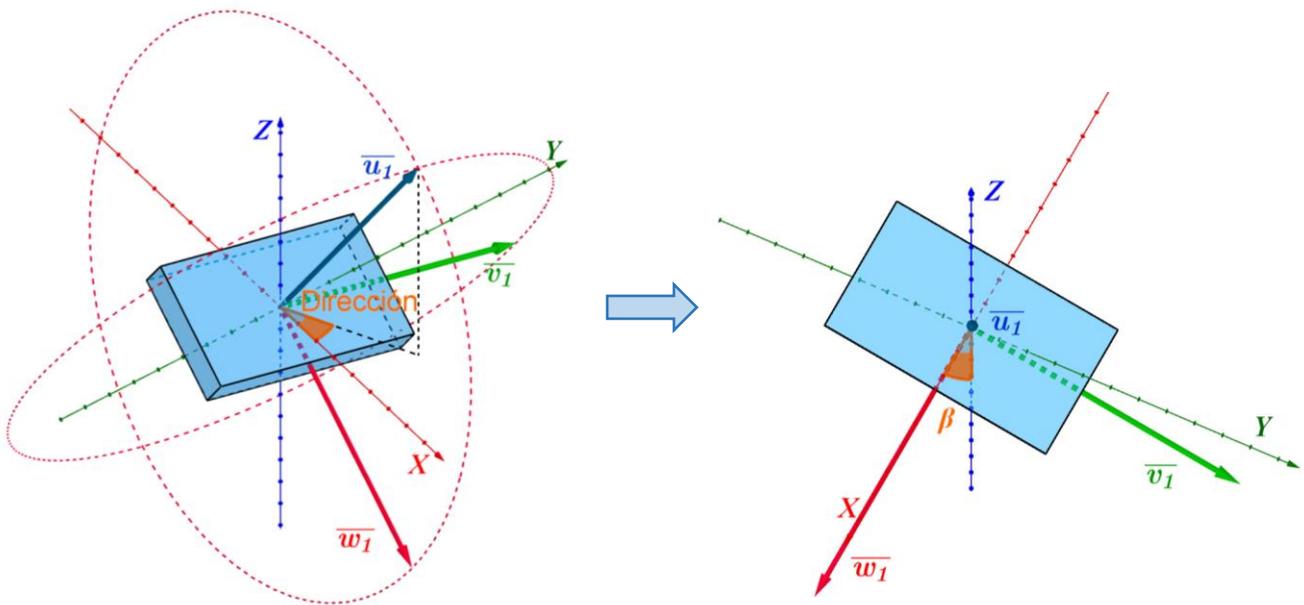


Figura 6-10 Ángulo entre el plano de inclinación y el vector unitario  $\overline{w_1}$ , al cual se le nombra ángulo de dirección.

### Matriz de transformación de par

Con base en los ángulos que se describieron anteriormente, se puede obtener una matriz de transformación para relacionar el par necesario que se necesita en el plano de inclinación del robot, con el par de cada una de las tres ruedas. Para esto se utilizaron las figuras 6-11, 6-12 y 6-13 que se muestran a continuación.

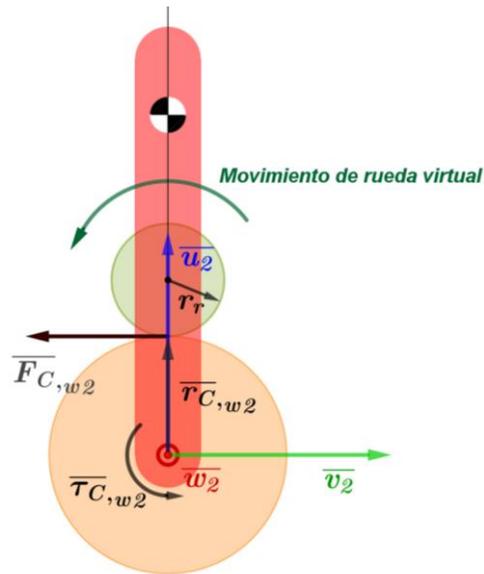


Figura 6-11 Representación de la fuerza  $\overline{F}_{C,w2}$  generada por una rueda virtual en el plano de inclinación del robot, la cual se encuentra a una distancia  $\overline{r}_{C,w2}$  del centro de la esfera y genera un par  $\overline{\tau}_{C,w2}$ .

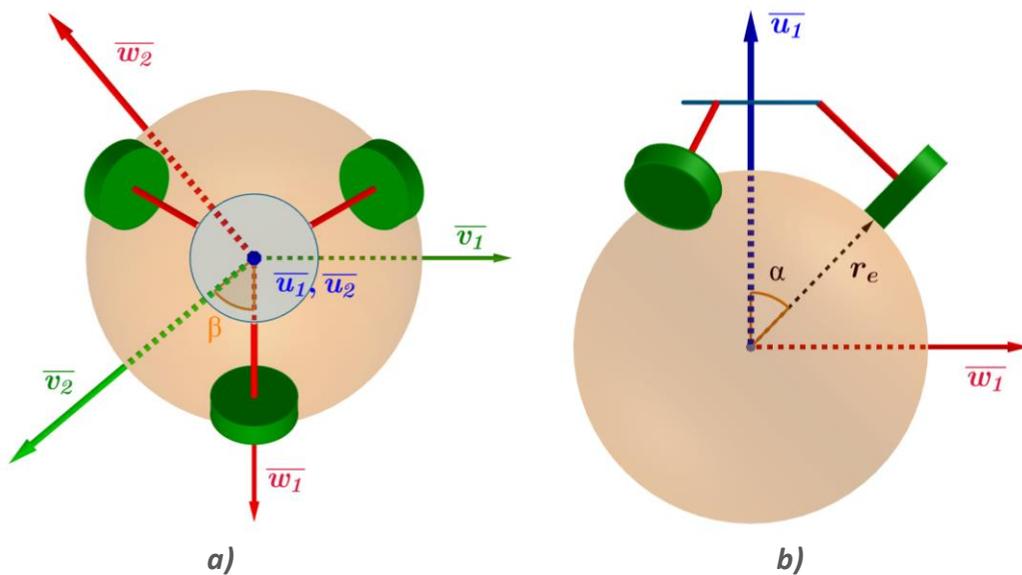
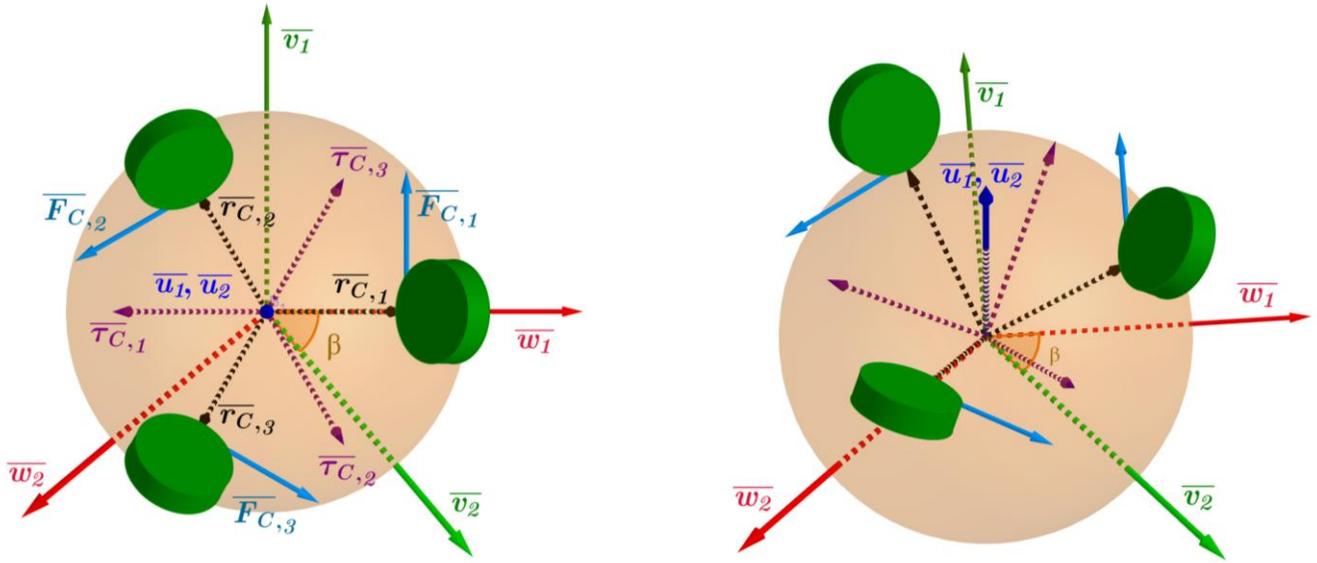


Figura 6-12 La imagen a) muestra la relación que existe entre el sistema del plano de inclinación y el sistema de la IMU; la imagen b) representa el ángulo de inclinación de las ruedas con respecto al eje central del robot.



*Figura 6-13 Representación de las fuerzas  $\overline{F}_{C,i}$  ejercidas por las tres ruedas, cuyas distancias entre sus puntos de contacto al centro de la esfera están representadas mediante los vectores  $\overline{r}_{C,i}$  y generan tres pares  $\overline{\tau}_{C,i}$  en el cuerpo del robot, en los que  $i=1,2,3$ .*

Como al pasar de un sistema a otro el par se conserva, se puede decir que el par que se desea en el plano en el que se inclina el robot debe de ser el mismo que el par resultante de las tres ruedas, esto se puede escribir de la siguiente manera:

$$\overline{\tau}_{c,w2} = \overline{\tau}_{c,1} + \overline{\tau}_{c,2} + \overline{\tau}_{c,3} \quad (6.48)$$

Para calcular cada uno de los pares que actúan en el robot, se tienen las siguientes ecuaciones:

$$\overline{\tau}_{c,w2} = \overline{r}_{c,w2} \times \overline{F}_{c,w2} \quad (6.49)$$

$$\overline{\tau}_{c,1} = \overline{r}_{c,1} \times \overline{F}_{c,1} \quad (6.50)$$

$$\overline{\tau}_{c,2} = \overline{r}_{c,2} \times \overline{F}_{c,2} \quad (6.51)$$

$$\overline{\tau}_{c,3} = \overline{r}_{c,3} \times \overline{F}_{c,3} \quad (6.52)$$

Con el sistema de referencia de la IMU, los vectores  $\overline{F}_{c,w2}$  y  $\overline{r}_{c,w2}$  se definen de la siguiente manera:

$$\overline{F}_{c,w2} = \frac{\tau}{r_r} \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \quad (6.53)$$

$$\bar{r}_{c,w2} = r_e \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (6.54)$$

en donde  $\tau$  es la magnitud del par de entrada entregado por el motor virtual,  $r_e$  es el escalar que representa el radio de la esfera y  $r_r$  es el escalar que representa el radio de la rueda virtual, el cual es igual que el de las ruedas reales.

Asimismo, con el sistema de referencia de la IMU, se pueden definir los vectores  $\bar{F}_{c,i}$  y  $\bar{r}_{c,i}$  en los que  $i = 1,2,3$ , de la siguiente manera:

$$\bar{F}_{c,1} = \frac{\tau_1}{r_r} \begin{bmatrix} -\cos(\beta) \\ -\text{sen}(\beta) \\ 0 \end{bmatrix}; \bar{F}_{c,2} = \frac{\tau_2}{r_r} \begin{bmatrix} -\cos(\beta + \frac{2\pi}{3}) \\ -\text{sen}(\beta + \frac{2\pi}{3}) \\ 0 \end{bmatrix}; \bar{F}_{c,3} = \frac{\tau_3}{r_r} \begin{bmatrix} -\cos(\beta - \frac{2\pi}{3}) \\ -\text{sen}(\beta - \frac{2\pi}{3}) \\ 0 \end{bmatrix} \quad (6.55)$$

$$\bar{r}_{c,1} = r_e \begin{bmatrix} -\text{sen}(\alpha) \cdot \text{sen}(\beta) \\ \text{sen}(\alpha) \cdot \cos(\beta) \\ \cos(\alpha) \end{bmatrix}; \bar{r}_{c,2} = r_e \begin{bmatrix} -\text{sen}(\alpha) \cdot \text{sen}(\beta + \frac{2\pi}{3}) \\ \text{sen}(\alpha) \cdot \cos(\beta + \frac{2\pi}{3}) \\ \cos(\alpha) \end{bmatrix}; \bar{r}_{c,3} = r_e \begin{bmatrix} -\text{sen}(\alpha) \cdot \text{sen}(\beta - \frac{2\pi}{3}) \\ \text{sen}(\alpha) \cdot \cos(\beta - \frac{2\pi}{3}) \\ \cos(\alpha) \end{bmatrix} \quad (6.56)$$

en donde  $\tau_1, \tau_2$  y  $\tau_3$  son las magnitudes de los pares entregados por cada uno de los tres motores reales.

Con las ecuaciones anteriores,  $\bar{\tau}_{c,w2}$ ,  $\bar{\tau}_{c,1}$ ,  $\bar{\tau}_{c,2}$  y  $\bar{\tau}_{c,3}$  se pueden representar de la siguiente manera:

$$\bar{\tau}_{c,w2} = \begin{bmatrix} \frac{r_e \cdot \tau}{r_r} \\ 0 \\ 0 \end{bmatrix} \quad (6.57)$$

$$\tau_{c,1} = \begin{bmatrix} \frac{r_e \cdot \cos(\alpha) \cdot \tau_1 \cdot \text{sen}(\beta)}{r_r} \\ -\frac{r_e \cdot \cos(\alpha) \cdot \tau_1 \cdot \cos(\beta)}{r_r} \\ \frac{\text{sen}(\alpha) \cdot r_e \cdot \tau_1}{r_r} \end{bmatrix}; \tau_{c,2} = \begin{bmatrix} \frac{r_e \cdot \cos(\alpha) \cdot \tau_2 \cdot \text{sen}(\beta + \frac{2\pi}{3})}{r_r} \\ -\frac{r_e \cdot \cos(\alpha) \cdot \tau_2 \cdot \cos(\beta + \frac{2\pi}{3})}{r_r} \\ \frac{\text{sen}(\alpha) \cdot r_e \cdot \tau_2}{r_r} \end{bmatrix}; \tau_{c,3} = \begin{bmatrix} \frac{r_e \cdot \cos(\alpha) \cdot \tau_3 \cdot \text{sen}(\beta - \frac{2\pi}{3})}{r_r} \\ -\frac{r_e \cdot \cos(\alpha) \cdot \tau_3 \cdot \cos(\beta - \frac{2\pi}{3})}{r_r} \\ \frac{\text{sen}(\alpha) \cdot r_e \cdot \tau_3}{r_r} \end{bmatrix} \quad (6.58)$$

Sustituyendo las expresiones en la ecuación 5.48, entonces se obtiene:

$$\frac{r_e \tau}{r_r} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \frac{r_e \cdot \tau_1}{r_r} \begin{bmatrix} \cos(\alpha) \cdot \text{sen} \beta \\ -\cos(\alpha) \cdot \cos \beta \\ \text{sen}(\alpha) \end{bmatrix} + \frac{r_e \cdot \tau_2}{r_r} \begin{bmatrix} \cos(\alpha) \cdot \text{sen}(\beta + \frac{2\pi}{3}) \\ -\cos(\alpha) \cdot \cos(\beta + \frac{2\pi}{3}) \\ \text{sen}(\alpha) \end{bmatrix} + \frac{r_e \cdot \tau_3}{r_r} \begin{bmatrix} \cos(\alpha) \cdot \text{sen}(\beta - \frac{2\pi}{3}) \\ -\cos(\alpha) \cdot \cos(\beta - \frac{2\pi}{3}) \\ \text{sen}(\alpha) \end{bmatrix} \quad (6.59)$$

$$\tau \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \tau_1 \begin{bmatrix} \cos(\alpha) \cdot \text{sen } \beta \\ -\cos(\alpha) \cdot \cos \beta \\ \text{sen}(\alpha) \end{bmatrix} + \tau_2 \begin{bmatrix} \cos(\alpha) \cdot \text{sen} \left( \beta + \frac{2\pi}{3} \right) \\ -\cos(\alpha) \cdot \cos \left( \beta + \frac{2\pi}{3} \right) \\ \text{sen}(\alpha) \end{bmatrix} + \tau_3 \begin{bmatrix} \cos(\alpha) \cdot \text{sen} \left( \beta - \frac{2\pi}{3} \right) \\ -\cos(\alpha) \cdot \cos \left( \beta - \frac{2\pi}{3} \right) \\ \text{sen}(\alpha) \end{bmatrix} \quad (6.60)$$

$$\tau \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \cos(\alpha) \cdot \text{sen}(\beta) & \cos(\alpha) \cdot \text{sen} \left( \beta + \frac{2\pi}{3} \right) & \cos(\alpha) \cdot \text{sen} \left( \beta - \frac{2\pi}{3} \right) \\ -\cos(\alpha) \cdot \cos(\beta) & -\cos(\alpha) \cdot \cos \left( \beta + \frac{2\pi}{3} \right) & -\cos(\alpha) \cdot \cos \left( \beta - \frac{2\pi}{3} \right) \\ \text{sen}(\alpha) & \text{sen}(\alpha) & \text{sen}(\alpha) \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \quad (6.61)$$

$$\tau \begin{bmatrix} \cos(\alpha) \cdot \text{sen}(\beta) & \cos(\alpha) \cdot \text{sen} \left( \beta + \frac{2\pi}{3} \right) & \cos(\alpha) \cdot \text{sen} \left( \beta - \frac{2\pi}{3} \right) \\ -\cos(\alpha) \cdot \cos(\beta) & -\cos(\alpha) \cdot \cos \left( \beta + \frac{2\pi}{3} \right) & -\cos(\alpha) \cdot \cos \left( \beta - \frac{2\pi}{3} \right) \\ \text{sen}(\alpha) & \text{sen}(\alpha) & \text{sen}(\alpha) \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \quad (6.62)$$

$$\tau \begin{bmatrix} \frac{2 \text{sen}(\beta)}{3 \cos(\alpha)} & -\frac{2 \cos(\beta)}{3 \cos(\alpha)} & \frac{1}{3 \text{sen}(\alpha)} \\ \frac{\sqrt{3} \cos(\beta) - \text{sen}(\beta)}{3 \cos(\alpha)} & \frac{\sqrt{3} \text{sen}(\beta) + \cos(\beta)}{3 \cos(\alpha)} & \frac{1}{3 \text{sen}(\alpha)} \\ -\frac{\sqrt{3} \cos(\beta) + \text{sen}(\beta)}{3 \cos(\alpha)} & -\frac{\sqrt{3} \text{sen}(\beta) - \cos(\beta)}{3 \cos(\alpha)} & \frac{1}{3 \text{sen}(\alpha)} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \quad (6.63)$$

$$\frac{\tau}{3 \cos(\alpha)} \begin{bmatrix} 2 \text{sen}(\beta) \\ \sqrt{3} \cos(\beta) - \text{sen}(\beta) \\ -\sqrt{3} \cos(\beta) - \text{sen}(\beta) \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \quad (6.64)$$

De la última expresión se obtiene el par que se necesitaría en cada uno de los motores con respecto al par de entrada que se desea.

## Ecuaciones de relación de velocidad

Además de plantear una relación entre el par que debían de suministrar los tres actuadores del sistema real con el par del motor “virtual” del modelo, también se decidió calcular una relación de velocidad que debía de existir entre las tres ruedas para que el robot fuera a una dirección determinada, todo esto con el fin de intentar controlar dicha dirección para regresar al robot a su posición de equilibrio. Se propuso que dicha relación podría probarse para implementar un algoritmo de control de forma empírica, pues dentro del estudio del estado de la técnica no se encontró de qué forma podría implementarse.

Para obtener la relación de velocidad, nuevamente se utilizó la simplificación del robot en 2D, en la que el plano que se está analizando es el plano donde se inclina el robot, ver figura 6-14. Se puede apreciar en dicha simplificación que, para regresar al robot a su posición de equilibrio, la rueda virtual debe de

moveirse a una determinada velocidad angular y el cuerpo del robot también tendrá una determinada velocidad angular, a esta velocidad se le nombró  $\overline{\omega}_c$ .

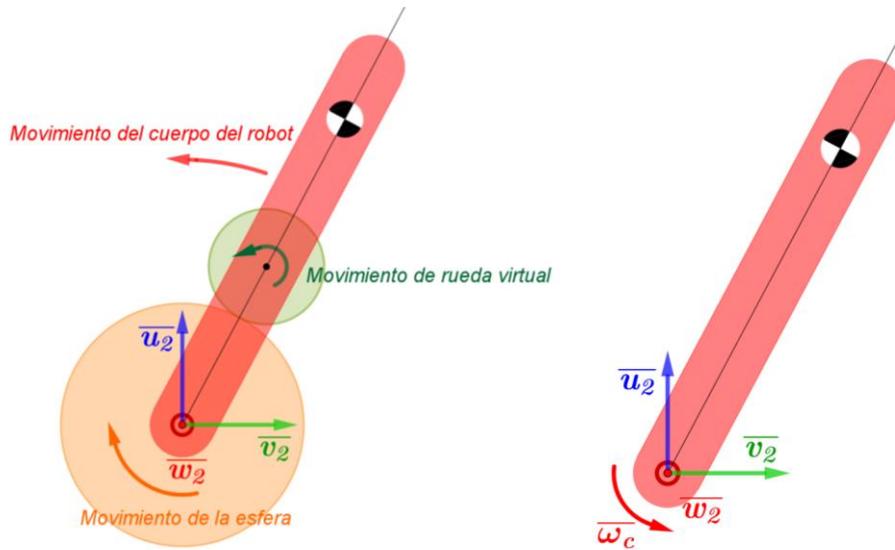
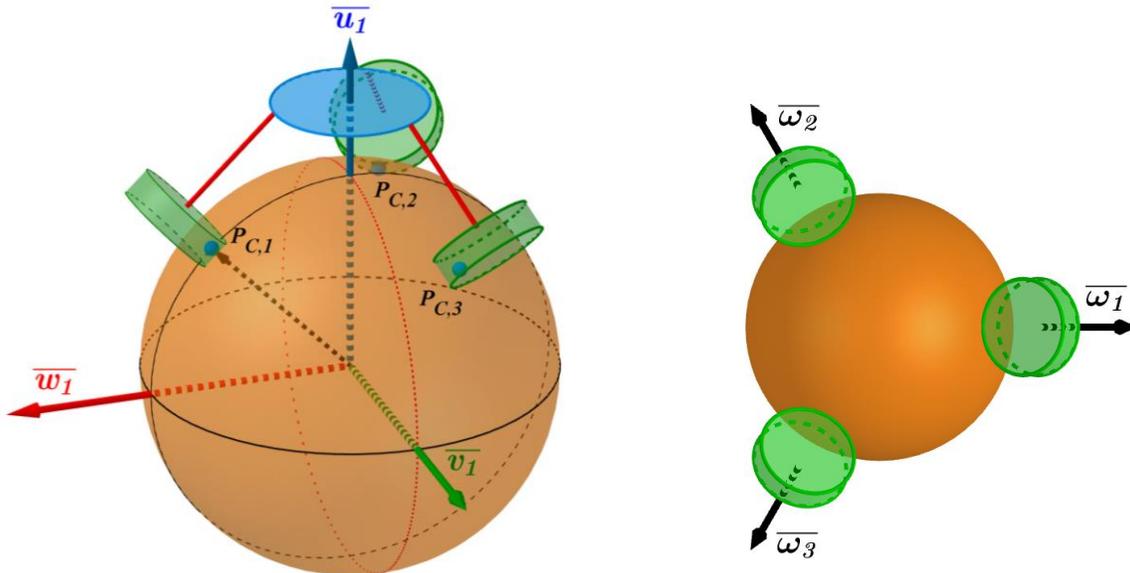


Figura 6-14 Modelo simplificado en el que se observa la dirección del movimiento de los componentes del robot y la velocidad angular del cuerpo del robot.

En el sistema real cada rueda tiene un punto de contacto con la esfera, tal y como se aprecia en la figura 6-15, a cada punto de contacto se le nombró  $P_{C,i}$ , en el que  $i = 1,2,3$  dependiendo de cada

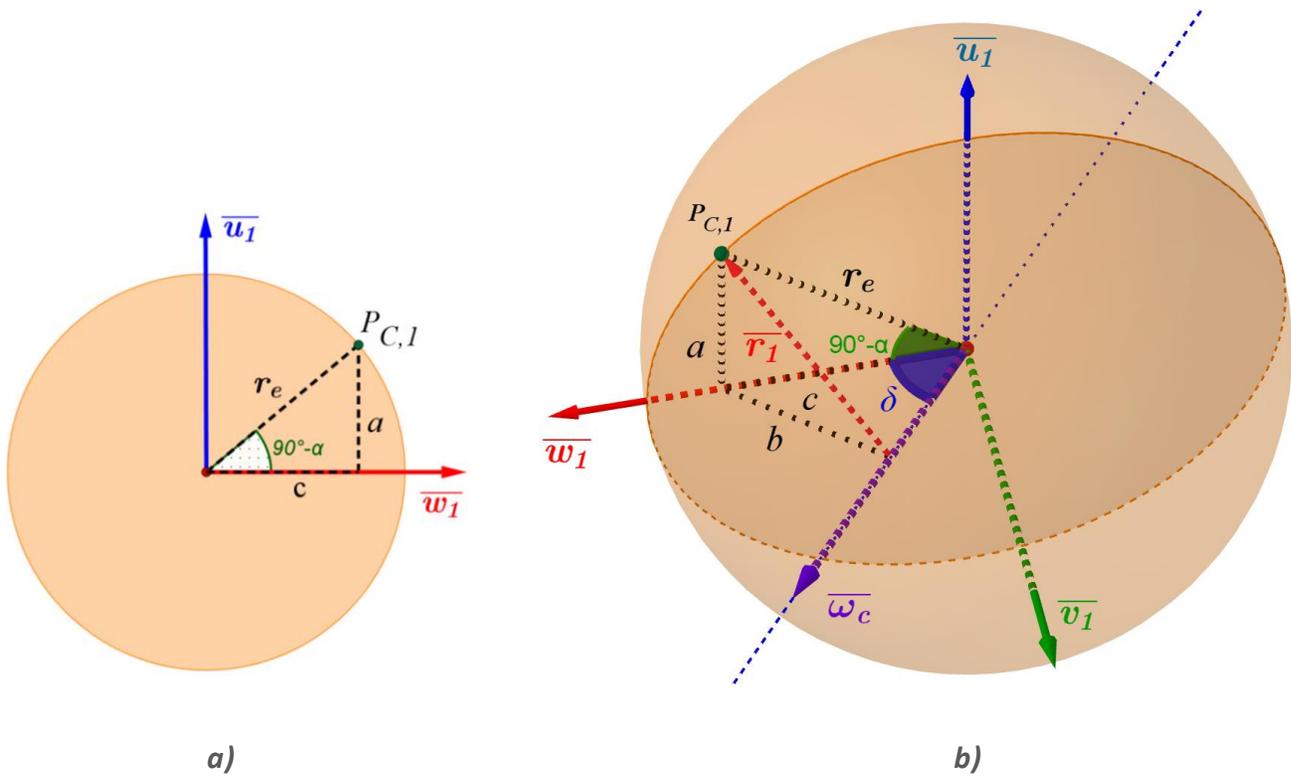


rueda.

Figura 6-15 Representación de cada punto de contacto  $P_{C,i}$  y representación de los vectores de velocidad angular  $\overline{\omega}_i$ , en los que  $i=1,2,3$ .

Teniendo la velocidad angular del cuerpo del robot  $\overline{\omega}_c$  y teniendo la posición de cada punto de contacto de las ruedas con la esfera, se pudo determinar la velocidad tangencial que se debería de tener en cada rueda justo en su punto de contacto, a esta velocidad se le llamó  $\overline{V}_{P_{C_i}}$ . Sin embargo, la línea de acción de las ruedas no necesariamente coincide con la dirección de dicha velocidad, por lo que para sacar la velocidad angular  $\overline{\omega}_i$  que debería de tener cada una de las ruedas, se tenía que obtener la componente de cada velocidad  $\overline{V}_{P_{C_i}}$  que iba en dirección a la línea de acción de cada rueda  $i$ . Dado que tanto la dirección y el sentido de cada vector velocidad  $\overline{\omega}_i$  se podía determinar fácilmente con ayuda del ángulo de inclinación de la IMU, únicamente se hizo un análisis para obtener la magnitud de cada velocidad angular, es decir  $|\overline{\omega}_i|$ .

Para el cálculo de la magnitud de la velocidad angular de la rueda que estaba alineada con uno de los ejes del sistema local determinado por la IMU, a la cual se le denominó  $|\overline{\omega}_1|$ , las variables que se utilizaron se muestran de manera gráfica en las figuras 6-16 a) y 6-16 b). Asimismo, se muestra el significado de cada variable en la tabla 6-3.



**Figura 6-16 Representación gráfica de los parámetros utilizados para calcular la velocidad angular de cada una de las ruedas.**

**Tabla 6-3 Variables que se utilizaron para obtener las ecuaciones de velocidad angular de las ruedas.**

$\alpha$	Ángulo de inclinación de las ruedas con respecto a la vertical.
$\delta$	Ángulo entre el vector velocidad angular $\overline{\omega_c}$ y el eje $\overline{w_1}$ el cual está alineado con la rueda 1.
$r_e$	Radio de la esfera.
$\overline{\omega_c}$	Vector velocidad angular del cuerpo del robot.
$\overline{r_1}$	Vector perpendicular a $\overline{\omega_c}$ que se encuentra entre el eje de rotación del robot y el punto de contacto $P_{c,1}$ .
$\overline{\omega_1}$	Velocidad angular de la rueda 1.
$r_r$	Radio de las ruedas.

La ecuación que se obtuvo para calcular el módulo de la velocidad angular de la rueda 1 queda de la siguiente manera:

$$|\overline{\omega_1}| = |\overline{\omega_c}| \cdot r_r \cdot \cos(\delta) \cdot \sqrt{r_e^2(\cos^2(\alpha) + \sin^2(\alpha) \cdot \sin^2(\delta))} \quad (6.65)$$

El ángulo que determina el plano de inclinación del robot  $\beta$ , el cual se obtiene con la IMU, es el ángulo complementario de  $\delta$ , por lo que la velocidad angular de la rueda 1 también podía escribirse de la siguiente manera:

$$|\overline{\omega_1}| = |\overline{\omega_c}| \cdot r_r \cdot \sin(\beta) \cdot \sqrt{r_e^2(\cos^2(\alpha) + \sin^2(\alpha) \cdot \cos^2(\beta))} \quad (6.66)$$

Para no hacer el mismo análisis con las otras dos ruedas, pues estas no coinciden con algún eje y por tal motivo el cálculo sería más complicado, sólo bastó con considerar que la misma ecuación se cumple igual para las otras dos ruedas, con la diferencia de que el ángulo  $\beta$  sería sustituido por  $\beta + 120^\circ$  o  $\beta + 240^\circ$  dependiendo de para qué rueda se calcularía la velocidad angular.

## Diseño y simulación de los controladores

Una vez planteadas las ecuaciones que describen el comportamiento del robot, se empezó a revisar la forma en la que se implementarían los algoritmos de control que se plantearon al principio del capítulo. Dichos algoritmos serían evaluados haciendo pruebas en el robot con el fin de evaluar su desempeño, y establecer cuál algoritmo funcionó de manera más adecuada para futuras pruebas, ya sea de estabilidad o navegación.

### Control de par del motor

El primer controlador a desarrollar para lograr la estabilidad del robot fue el controlador por par, que se muestra en la figura 6-17. Además de que este controlador se propuso en el trabajo previo con el fin

de mejorar el desempeño del robot, se encontró que se podía revisar de manera aislada el comportamiento dinámico de los motores, lo cual suponía una cierta ventaja pues permitiría desarrollar el controlador por partes y revisar su correcto funcionamiento. Por otro lado, en caso de cambiar alguna característica del robot, se podrían conservar algunas propiedades del control anterior.

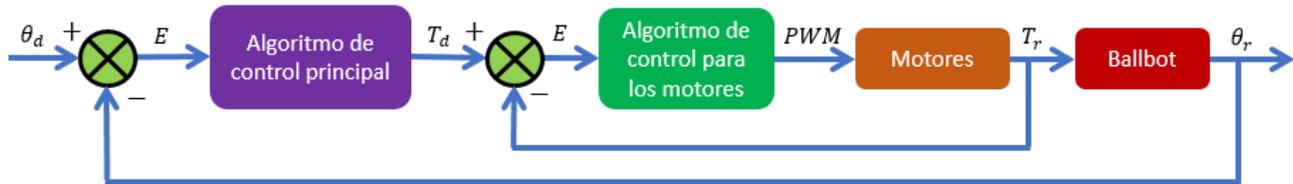


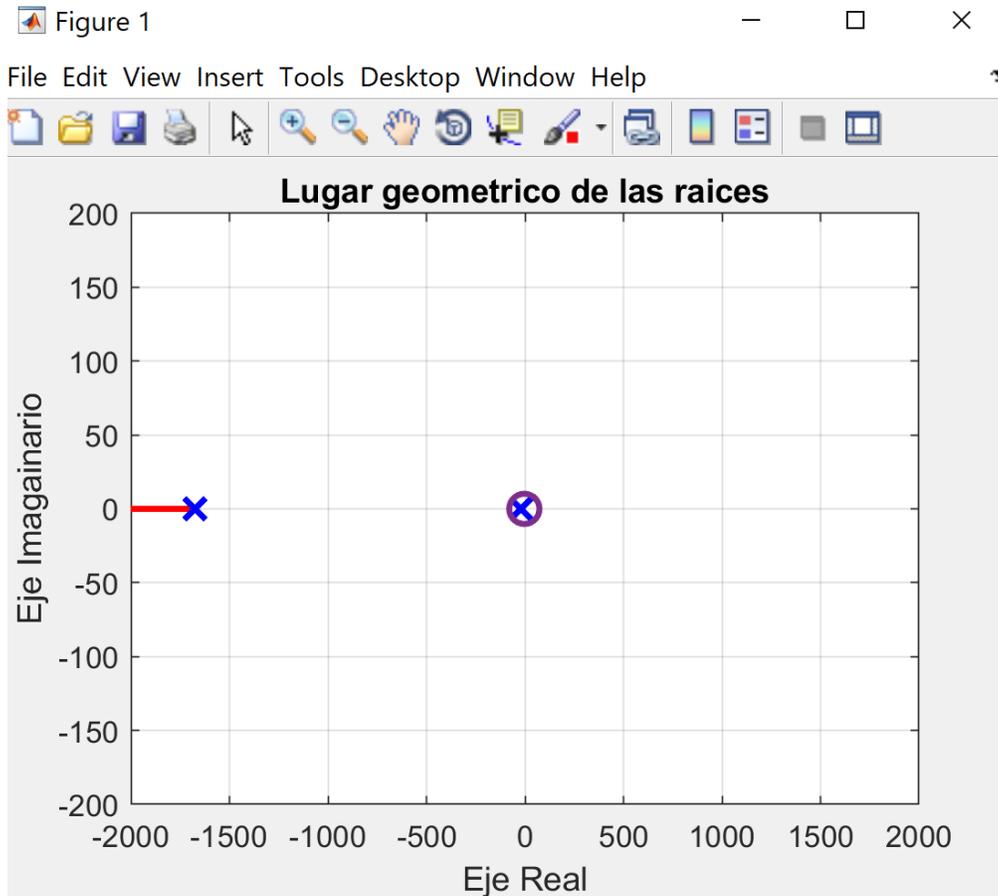
Figura 6-17 Esquema del algoritmo de control de par del robot.

Para saber qué algoritmo de control sería implementado para la parte de los motores, primero se revisó tanto el modelo de primer orden como el de segundo orden para un motor de corriente directa. Tal y como se mostró anteriormente, para ambos casos se obtuvo una función de transferencia en la que se relacionaba el voltaje de entrada con el par de salida. Cuando se obtuvieron las ecuaciones, se hizo la suposición de que la dinámica eléctrica del motor podría ser muy rápida y por tal motivo podría verse el sistema como uno de primer orden. Para comprobar lo que se mencionó anteriormente, se obtuvieron los polos y los ceros del sistema ya con los parámetros reales del motor. Los parámetros de un motor pueden ser proporcionados por el fabricante o pueden ser obtenidos de manera experimental; para este proyecto fue necesario obtenerlos de manera experimental pues en la página del fabricante no se encontraban dichos parámetros, por lo que se tuvo que revisar un trabajo titulado *Determinación de los parámetros de un motor de CD por medición física directa* [13]. Una vez realizadas las pruebas que se mencionan en dicho trabajo, los parámetros que se obtuvieron se muestran en la siguiente tabla.

Tabla 6-4 Parámetros del motor PG27.

Parámetro	$K_T [N \cdot m/A]$	$K_E [V \cdot s/rad]$	$R [\Omega]$	$b [N \cdot m \cdot s/rad]$	$J [kg \cdot m^2]$
Valor	0.3493	0.485	1.6934	0.0199	0.006

Para obtener los polos y ceros del sistema de segundo orden no se obtuvo la inductancia de los motores, pero se utilizó como variable para ver cómo sería el comportamiento de los polos y los ceros al cambiar dicha variable. Al momento de dibujar la gráfica de los polos y los ceros en Matlab, tal y como se muestra en la figura 6-18, se encontró que la inductancia afectaba en mayor medida a uno de los polos, el cual, a su vez estaba muy alejado del eje real. Es importante mencionar que se utilizaron valores de inductancia no muy alejados de la realidad. Con base en las gráficas de polos y ceros, se concluyó que el modelo de primer orden podría ser utilizado sin ningún problema.

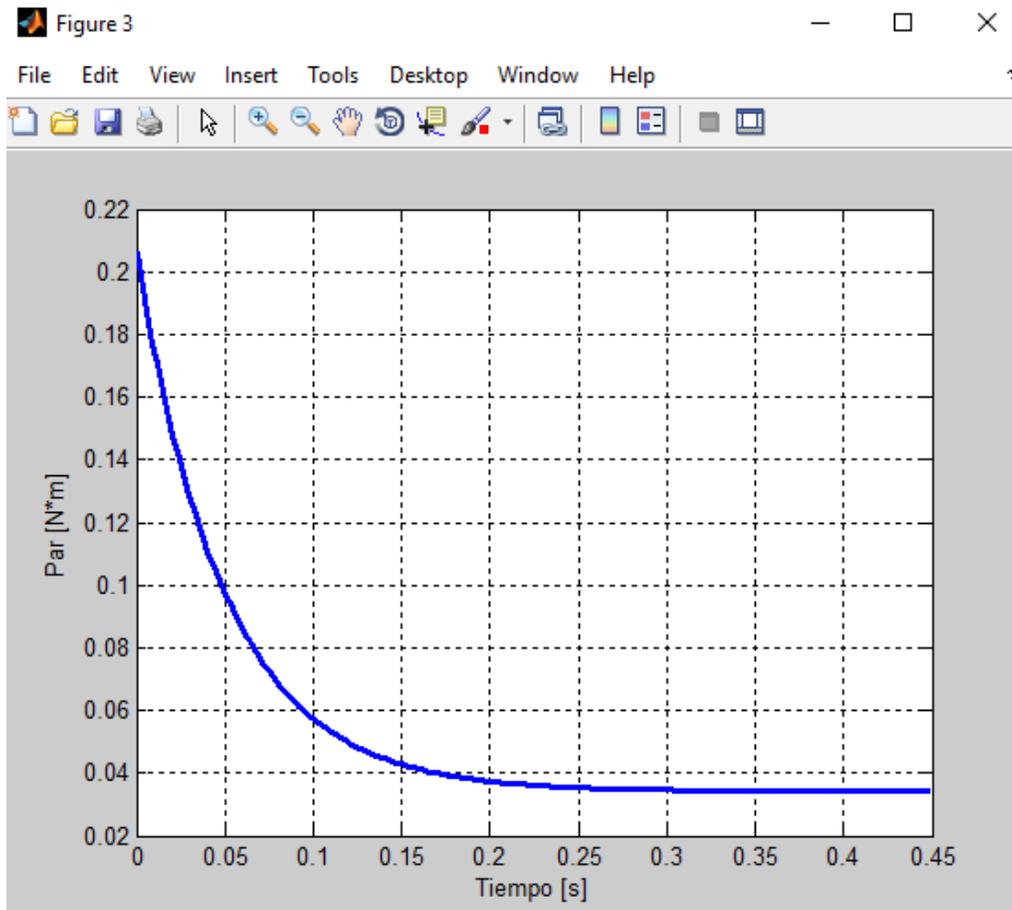


*Figura 6-18 Gráfica de polos y ceros del modelo del motor.*

Con los parámetros del motor obtenidos, la función de transferencia de primer orden queda de la siguiente manera.

$$H(s) = 0.2062 \left( \frac{s + 3.3166}{s + 19.99} \right)$$

Como se puede observar, el polo del sistema es  $p_m = -19.99$ . Como el polo se encuentra en el semiplano complejo izquierdo se puede asegurar que el sistema es estable. Para analizar la respuesta del motor, se hizo una simulación en Matlab a una entrada escalón unitario, ver figura 6-19.

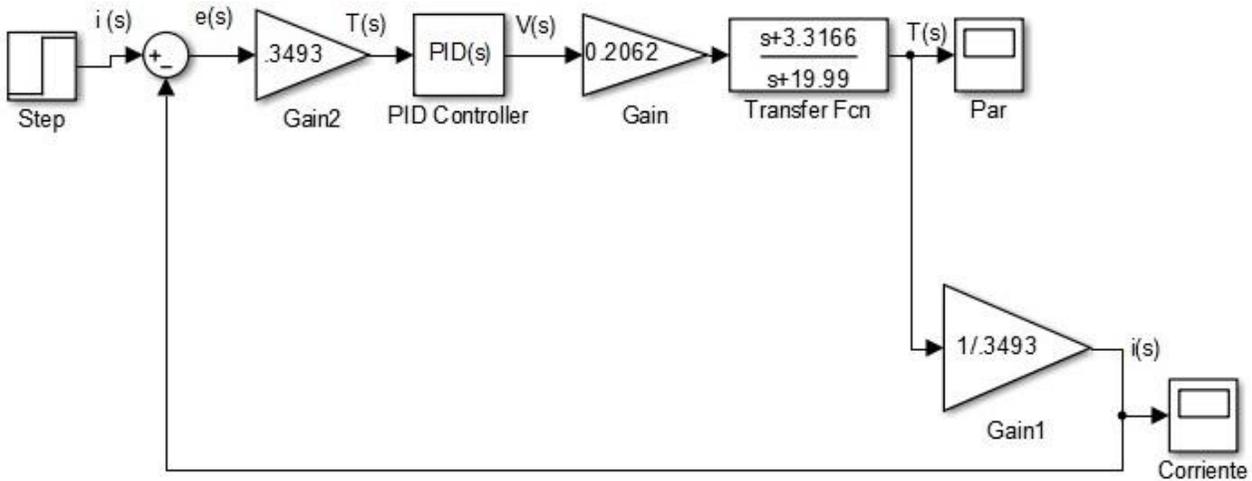


*Figura 6-19 Simulación de la respuesta del motor con una entrada escalón unitario.*

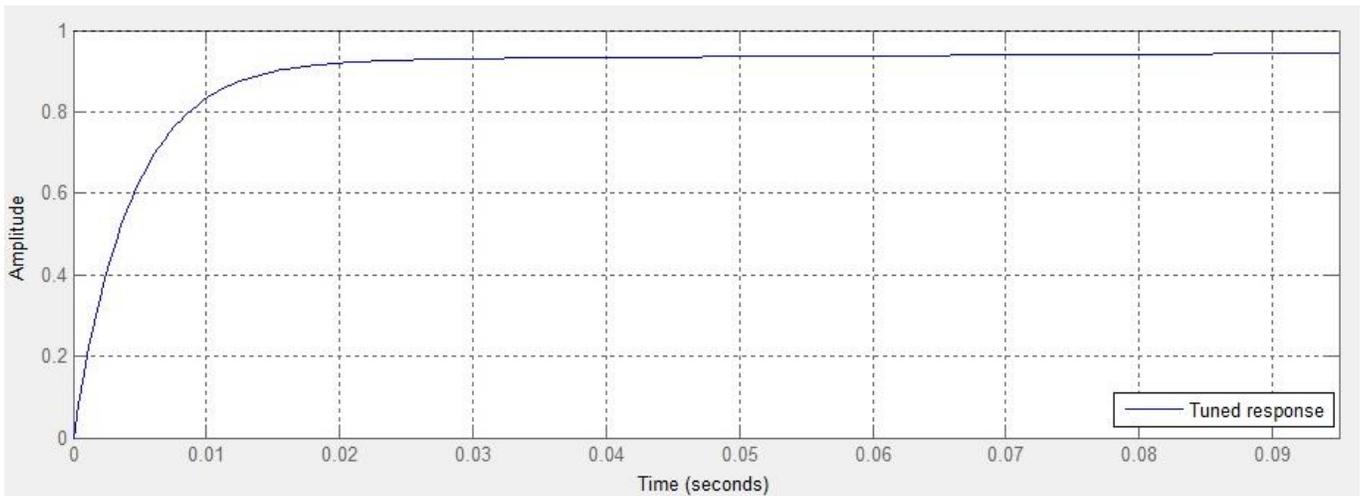
De acuerdo a la gráfica se puede observar que el par del motor decrece conforme pasa el tiempo; este comportamiento se asocia a que, en el modelo planteado, el motor está operando sin carga, y en dichas condiciones no se demanda mucho par en estado permanente.

Para elegir el controlador, se buscó dentro de la literatura controladores sencillos que pudieran ser implementados para controlar el par de un motor de corriente directa. En el libro de *Ingeniería de control moderna* [14] se encontró que se puede utilizar un control PI para controlar el par de un motor de CD con una función de transferencia de primer orden, así que, con base en lo anterior, se decidió usar un controlador PI para los motores. Las ganancias fueron calculadas de manera iterativa con la herramienta PID Tool de Matlab; para esto fue necesario revisar el tiempo de respuesta que se deseaba en el sistema. Con base en el trabajo que antecede a este proyecto, se consideró que todo el sistema debería de alcanzar la estabilidad ante cualquier perturbación cuando menos en 500 ms, por lo que el tiempo de respuesta del motor debería ser menor, así que se propuso que el tiempo de levantamiento fuera como máximo de 100 ms. Una vez establecido este parámetro, se calcularon las constantes de partida para el control PI de los motores.

Con ayuda de la herramienta de Matlab, se logró obtener una simulación de la salida del sistema para el control de par del motor mediante un diagrama de bloques, mostrado en la figura 6-20, en la que la señal de realimentación del sistema fue la corriente, ya que en el modelo físico no era posible medir directamente el par de los motores; sin embargo, es posible establecer la relación del par con la corriente mediante la constante  $K_t$ , y mediante sensores es posible medir la corriente consumida por dichos motores. En la figura 6-21 se puede observar la gráfica de la simulación del control del motor ante una entrada escalón unitario de corriente.



**Figura 6-20 Diagrama de bloques del control de par.**



**Figura 6-21 Respuesta a entrada escalón unitario de la simulación del control del motor.**

En cuanto al control principal del sistema, se decidió controlar el ángulo de inclinación con un controlador PID, en el que, para obtener las constantes proporcional, integral y derivativa, se midieron algunos parámetros del sistema de forma aproximada (como por ejemplo los momentos de inercia de

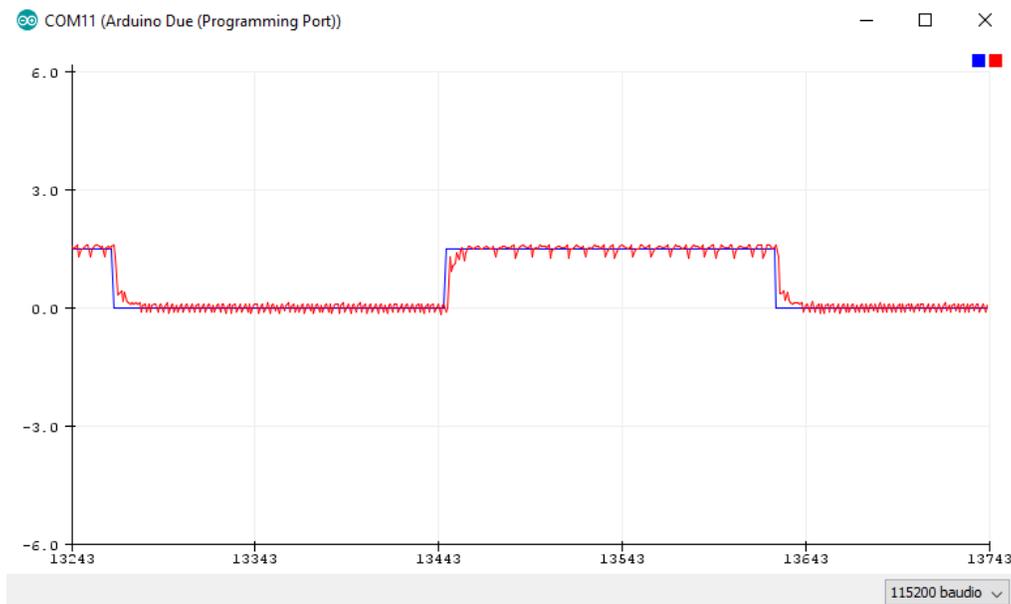
cada componente) para sustituirlos en el modelo; la forma en la que se obtuvieron dichos parámetros se muestra en el apéndice 3. Para el cálculo de las constantes, igualmente se utilizó, la herramienta PID Tool de Matlab. En cuanto al par que se demandaría en cada motor, este dependería tanto de las constantes como de la relación de par obtenida anteriormente y el ángulo que determina el plano de inclinación del robot.

### ***Control con relación de velocidad***

Como se mencionó anteriormente, también se intentaría probar un control utilizando la relación de velocidad que se obtuvo; esta relación se utilizaría en vez de la relación de par. Para esto se decidió utilizar también un control PID, en el que las constantes proporcional, integral y derivativa se sintonizarían de manera empírica pues no se tenía un modelo en el que se tomara a la velocidad como una señal de entrada para controlar la inclinación del robot. La forma en la que se controlaría el ángulo de inclinación del robot sería mandando una señal PWM a cada rueda, con base en dicha relación de velocidad.

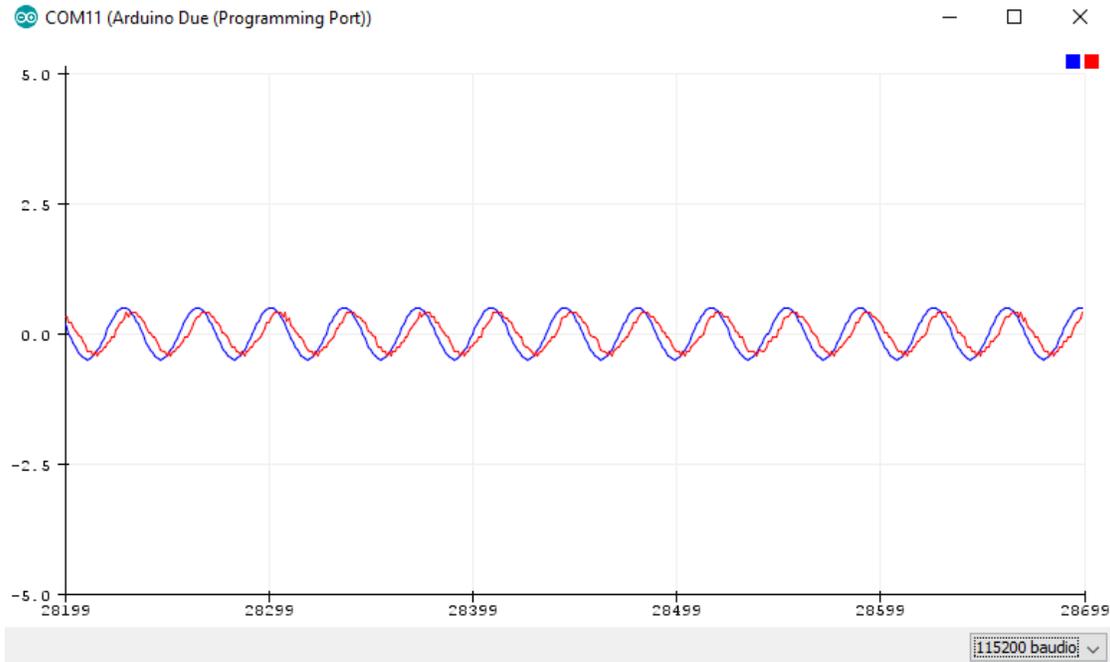
## Primeras pruebas

El primer controlador que se probó para la estabilidad del robot fue el de par. Antes de implementar todo el algoritmo de control en el robot, primero se probó por separado el control PI en cada uno de los motores. Para esto se desarrolló un programa que envía una señal cuadrada correspondiente al par al que cada uno de los motores tiene que llegar (señal de referencia). Como se puede observar en la figura 7-1, la respuesta que tuvo el motor (gráfica roja) fue muy similar a la que se debía obtener (gráfica azul); esta prueba se realizó con los tres motores por separado utilizando las constantes que se obtuvieron con la herramienta PIDTool de Matlab. En cada motor se obtuvieron resultados similares.



*Figura 7-1 Prueba con señal cuadrada.*

La siguiente prueba consistió en enviar una señal sinusoidal, pues se pensó que el sistema se comportaría de una manera muy parecida. Se hicieron pruebas con la señal sinusoidal a diferentes amplitudes. En las pruebas, se observó que se pudo recrear la señal esperada, sin embargo, con amplitudes menores, la señal obtenida del par de los motores empezaba a tener un pequeño desfase con respecto a la señal de referencia, esto se puede apreciar en la figura 7-2.

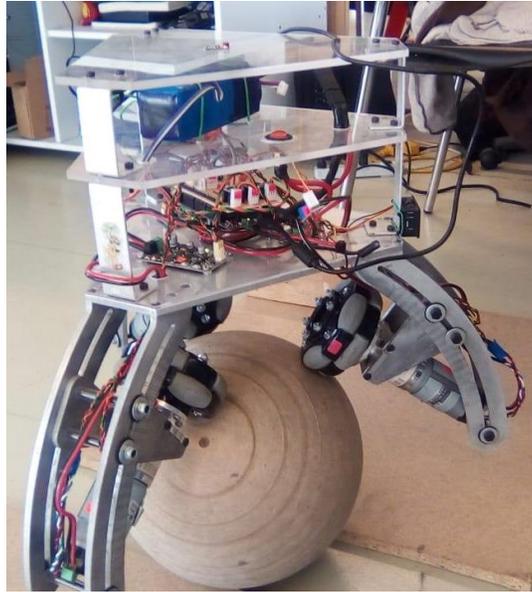


**Figura 7-2 Pruebas con respuesta sinusoidal.**

Luego de las pruebas de control de par en cada uno de los motores, se desarrolló el programa completo para el control general del robot, con un controlador PID, en el que para calcular las constantes proporcional, integral y derivativa, de igual manera se utilizó la herramienta PIDTool de Matlab.

Las pruebas para el control general consistieron en colocar al robot sobre la esfera para que este intentara estabilizarse, sin embargo, el robot no lo consiguió. Para mejorar el desempeño del robot, se intentó modificar las constantes del control PID de forma empírica, y también se probó en lugar de la esfera de basquetbol, que hasta el momento se había utilizado, una esfera de madera que anteriormente ya se había manufacturado para las versiones anteriores. Con los cambios antes mencionados, el robot mejoró su desempeño, pero no logró estabilizarse más de cinco segundos. Se observó que, durante las pruebas, el robot oscilaba demasiado, provocando que las ruedas dejaran de tener contacto con la esfera, lo que producía que el controlador no funcionara adecuadamente.

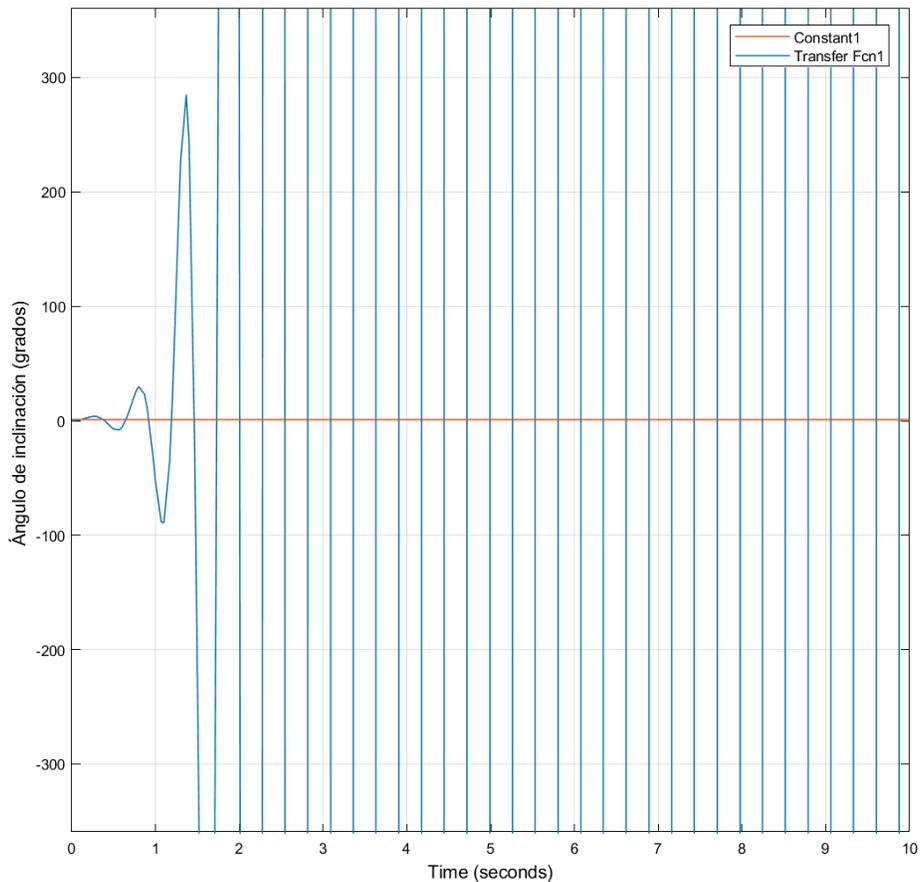
Asimismo, también se probó el algoritmo de control en el cual se utilizó la relación de velocidad. A pesar de que el robot logró estabilizarse hasta un tiempo de ocho segundos, se concluyó que este tiempo era insuficiente para intentar realizar pruebas de navegación, en la figura 7-3 se muestra al robot en operación durante la prueba de estabilidad.



*Figura 7-3 Ballbot durante la prueba de estabilidad.*

## Etapa de análisis y cambios en el diseño

Ya que el robot no logró la estabilización esperada en las pruebas, se realizó un diagrama de bloques en Matlab para simular el control PID general del ballbot, esto con el fin de tener un mejor entendimiento de su comportamiento, en él se utilizaron las constantes proporcional, integral y derivativa que se calcularon con la herramienta PIDTool. La gráfica del sistema se muestra en la figura 7-4 en la que se puede observar que el robot no fue capaz de permanecer estable aun con las ganancias calculadas.



*Figura 7-4 Simulación de respuesta del sistema ante una entrada escalón unitario.*

Por otro lado, se observó que el cuerpo del robot era demasiado grande y pesado, por lo que la inercia provocaba demasiadas oscilaciones sin que el robot lograra estabilizarse. Asimismo, se observó que, debido a la relación de tamaño entre el robot y la esfera, el robot salía de su zona estable rápidamente.

Con base en lo anterior, se concluyó que, para lograr una mejor estabilidad, se debían de realizar algunos cambios en el robot, ya sea en la parte física o en la parte lógica.

Una de las propuestas que se plantearon para mejorar la estabilidad fue implementar un controlador más robusto, y por tal motivo se empezó a desarrollar un controlador por realimentación de estados, cuya simulación se muestra en el apéndice 4; sin embargo, se encontró complejo aplicar dicho controlador al sistema real porque no se podía medir de manera directa todas las variables de estado del sistema, como por ejemplo el desplazamiento de la esfera, por lo que, para aplicar dicho control se tendrían que implementar observadores, lo cual implicaba que el algoritmo de control tendría que ser ejecutado por una computadora o un microcontrolador con mejores características.

Por otro lado, también se propuso realizar cambios mecánicos en la estructura y la esfera de acuerdo a las siguientes consideraciones:

- La nueva estructura debía ser más ligera que la anterior
- El tamaño de la nueva estructura debe ser más pequeño que el tamaño de la esfera

Con base en estas consideraciones y siguiendo la misma metodología de diseño que se utilizó en la estructura ya armada, se volvió a revisar el estado de la técnica con el fin de encontrar diseños ya probados que pudieran servir de referencia para el rediseño de la estructura. En esta búsqueda se encontró que algunos robots utilizaban esferas grandes de unicel con recubrimiento, y para su estructura utilizan materiales como acrílico o algún polímero, ya sea PLA o ABS para hacer estructuras más ligeras. Dentro de los diseños encontrados en el estudio de la técnica, se tomó como referencia el ballbot desarrollado en Xrobots, ya que se tiene evidencia de que dicho ballbot es capaz de estabilizarse con una estructura relativamente sencilla, su arquitectura electrónica es muy similar a la que se utiliza en el proyecto, además de que las características de la estructura cumplen con los requerimientos planteados anteriormente. Finalmente, con base en la información obtenida, se concluyó que mejorar el desempeño del robot desde su parte mecánica sería más sencillo de implementar que desde su parte lógica, por lo que, se comenzaron a plantear las tareas para el nuevo diseño.

Como primera tarea, se buscaron las características de los componentes del ballbot que se tomarían como referencia. En la página de internet <http://www.xrobots.co.uk/parts-concept> se encontró información del diseño del ballbot de referencia, así como las características de los componentes que los conforman. Dicha página se ocupó para obtener información tanto de los motores que se emplearon, como de las ruedas omnidireccionales y la esfera.

El motor que se seleccionó para el nuevo diseño fue el motor Pololu 25D con reducción 75:1 que se muestra en la figura 7-5. Este motor es capaz de dar un par máximo de hasta 1.05 N·m, una velocidad máxima sin carga de 130 RPM y opera a un voltaje de operación de 6V.



*Figura 7-5 Motor Pololu 25D.*

Para la selección de las ruedas omnidireccionales, se buscaron unas que fueran similares a las de la página antes mencionada. Se adquirieron unas ruedas omnidireccionales de 60 mm de diámetro como las que se muestran en la figura 7-6.



*Figura 7-6 Rueda omnidireccional de 60 mm de diámetro.*

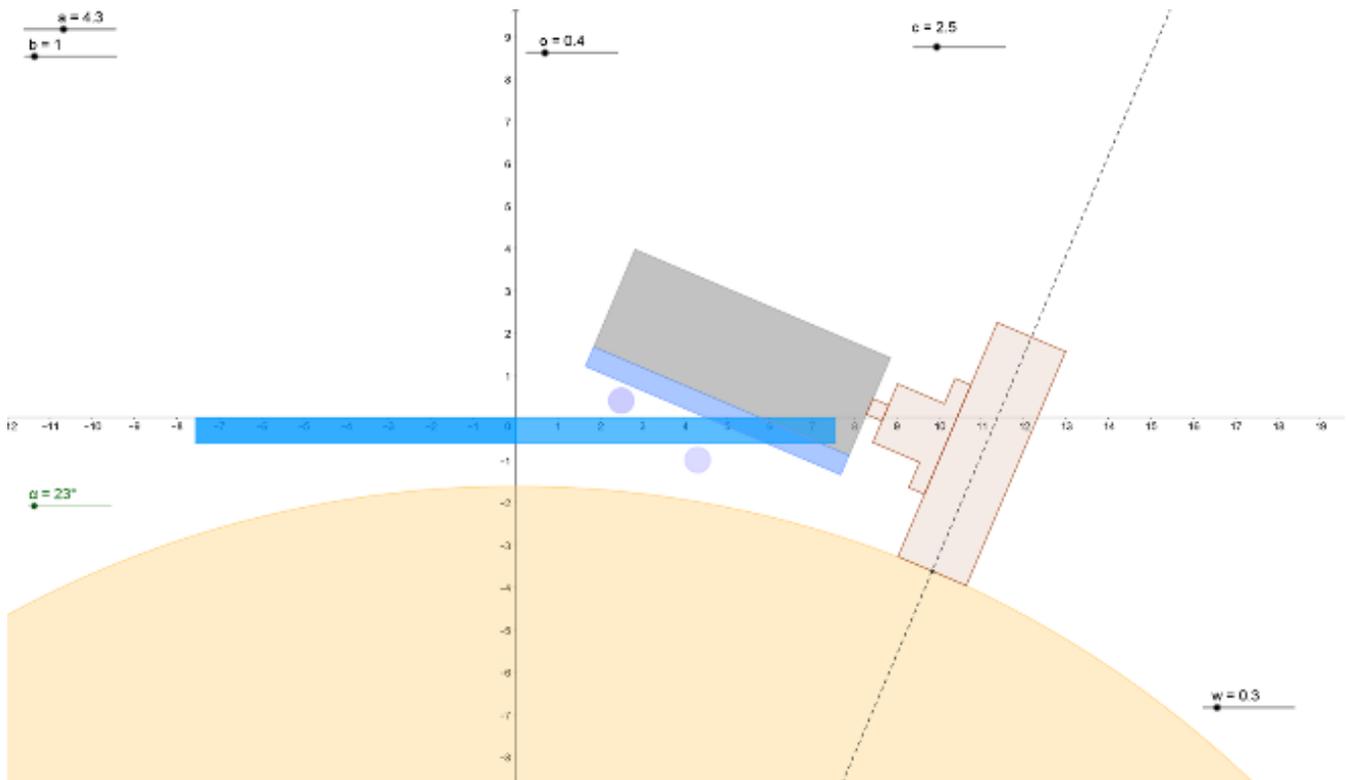
La esfera que el ballbot de referencia usa para estabilizarse es una de unicel hueca de 50 cm de diámetro, a la cual se le agregó un recubrimiento para evitar que las ruedas dañaran la esfera. Para este proyecto se compró una esfera de unicel hueca de 55 cm de diámetro, a la cual se le aplicó un recubrimiento de periódico, pintura acrílica y rellenador, con el fin de mejorar las características de su superficie, el proceso realizado se pudo apreciar en la figura 7-7.



*Figura 7-7 Proceso de recubrimiento de la primera esfera.*

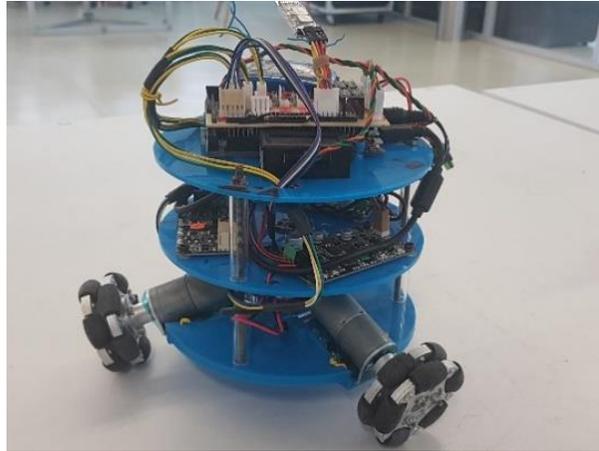
Una vez adquiridos los componentes que conformarían al robot, se procedió al diseño de la nueva estructura. Con el fin de acortar el tiempo de diseño, se pensó en utilizar conceptos similares a los que ya se habían ocupado para la estructura que ya se había armado, solo que empleando materiales más ligeros y con dimensiones más pequeñas.

Igualmente, se realizaron los cálculos dimensionales de los componentes que conformarían el diseño mediante una simulación previa; cuya imagen se muestra en la figura 7-8, se tomaron como referencia las dimensiones de componentes tales como la esfera, la rueda, el hub y el motor. Con base en dicha simulación se calcularon de manera más sencilla las dimensiones de los platos y los soportes para los motores. Dado que en pruebas anteriores se comprobó que el ángulo de las ruedas no era un factor decisivo en la estabilidad, sólo se buscó que las ruedas fueran tangentes a la esfera y que el diámetro del plato fuera lo suficientemente grande para colocar los componentes electrónicos. Los ejes de los motores fueron colocados a un ángulo de  $23^\circ$  con respecto a la horizontal, quedando el diámetro del plato de 16 cm.



*Figura 7-8 Simulación para el diseño de la estructura del ballbot.*

Por último, para el diseño de la nueva estructura se utilizó un software de CAD, posteriormente se realizó la manufactura de cada una de las piezas en corte laser, finalmente se realizó el ensamble de los componentes mecánicos y electrónicos, en la figura 7-9 se puede apreciar el diseño final del robot.



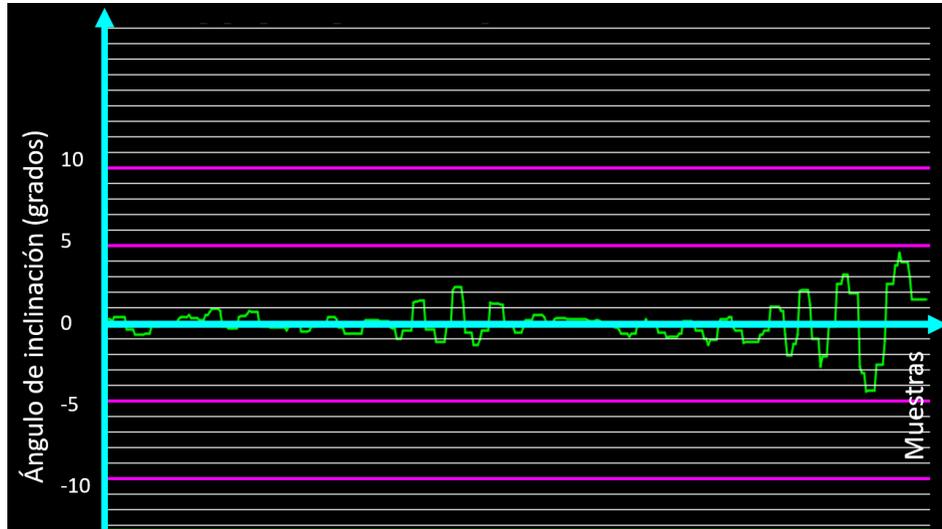
*Figura 7-9 Diseño final del nuevo ballbot.*

Una vez realizados los cambios a la estructura y a la esfera, se realizó nuevamente la prueba de estabilidad. Para esta parte se siguieron manejando los mismos algoritmos de control. Luego de ajustar las variables de manera empírica, el robot logró estabilizarse durante un tiempo de siete minutos, en la figura 7-10 se pudo observar al robot durante la prueba.



*Figura 7-10 Ballbot durante la prueba de estabilidad.*

Posteriormente, para tener un mejor entendimiento del comportamiento del robot durante la prueba, se graficó uno de los ángulos de inclinación del robot, dicha grafica se pude observa en la figura 7-11. Con base en el análisis de la gráfica, se observó que el robot mantenía su estabilidad mientras no se inclinara más de 2°, una vez excedido dicho valor, el robot comenzaba a oscilar hasta caerse de la esfera, por último, cabe mencionar que al implementar la gráfica del ángulo de inclinación, el desempeño del ballbot cambio drásticamente, ya que su periodo de estabilidad se redujo a 30 segundos, esto debido a que el tiempo de procesamiento del programa aumentaba.

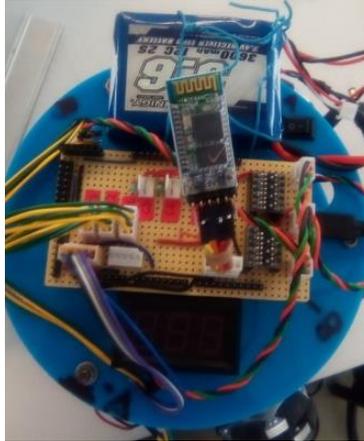


*Figura 7-11 Gráfica del ángulo roll con respecto al tiempo.*

### Primeras pruebas

Una vez realizadas las pruebas de estabilidad, el siguiente objetivo fue realizar las pruebas de navegación del ballbot; para esto, primero se plantearon los alcances de las pruebas en los cuales se concluyó que solo se probaría que el robot fuera capaz de seguir una trayectoria recta y que pudiera moverse de un punto A a un punto B. También se decidió qué para facilitar las pruebas, el robot se movería de manera tele operada.

Se decidió manejar al robot mediante comunicación inalámbrica debido a que así se tenía la ventaja de manejar el robot a una cierta distancia sin la necesidad de un cable; Dentro de los protocolos que existen actualmente se optó por utilizar comunicación bluetooth debido a su sencilla implementación en la plataforma de Arduino. Una vez decidido qué protocolo se utilizaría para las pruebas, se modificó la placa electrónica para poder incluir un módulo bluetooth en el ballbot, tal como se muestra en la figura 8-1.



*Figura 8-1 Modulo bluetooth en el robot.*

El algoritmo de navegación que se implementó consistió en modificar las variables de los ángulos pitch y roll mediante un pequeño desfase para cambiar el ángulo de inclinación del robot, haciendo que de esta manera se moviera en una dirección intentando equilibrarse y también se moviera la esfera. Modificando dichas variables el robot tendría la capacidad de moverse en cuatro direcciones específicas: adelante, atrás, izquierda y derecha.

Para la interacción con el operador, se desarrolló en Processing una interfaz gráfica, que se muestra en la figura 8-2, la cual permitía establecer la comunicación con el robot para posteriormente manipularlo. En cuanto a la manipulación, la interfaz contaba con cuatro botones acordes a los movimientos que podía realizar el robot; las direcciones son descritas en los botones como Aumenta A, Disminuye A, Aumenta B y Disminuye B. Asimismo, la interfaz permitía que el operador manipularse las direcciones mediante los botones del teclado de la computadora. Además, se mostraba en la interfaz el desfase producido en los ángulos roll y pitch y el ángulo de inclinación del robot. El código del programa se encuentra en el apéndice 7.



*Figura 8-2 Interfaz gráfica para la navegación.*

La primera prueba de movilidad consistió en hacer que el robot se moviera en línea recta; para esto se colocaron dos objetos, los cuales corresponderían al punto de partida y al punto de llegada. Dichos objetos se alinearon en línea recta tal como se muestra en la figura 8-3; posteriormente se colocó al robot en el punto de partida y se comenzó a mover ligeramente en dirección al otro punto.

En esta primera prueba se observó que el robot solo era capaz de moverse unos cuantos centímetros, posterior a eso ya no era capaz de mantenerse sobre la esfera. Se realizaron algunos cambios en el programa con el fin de inclinar al robot a un ángulo más pequeño al que se estaba manejando, sin embargo, se tuvieron los mismos resultados.



*Figura 8-3 Acomodo de los objetos para la prueba de movilidad.*

## **Etapa de análisis y cambios en el diseño**

Debido a que no se consiguió la navegación deseada, se hizo una búsqueda de posibles soluciones; se encontró que la navegación del ballbot podría ser llevada a cabo utilizando balines dentro de la esfera tal como se muestra en la figura 8-4.



*Figura 8-4 Solución con balines para la navegación del ballbot.*

Tomando esta idea como referencia, se fabricó otra esfera la cual fue llenada en su interior con canicas de 16 mm de diámetro. El número de canicas que se utilizó fue asignado de manera empírica, cuidando que no fueran tantas como para impedir que el robot pudiera mover la esfera ni tan pocas como para hacer que el efecto de las canicas no ayudara a la estabilización del robot. Además, a dicha esfera se le agregó un recubrimiento de hojas de periódico para evitar que sus dos mitades se separaran, y también para evitar que las ruedas del robot dañasen el unisel, ver la figura 8-5.



*Figura 8-5 canicas utilizadas y esfera para las pruebas de movilidad.*

## Pruebas finales

Una vez hechas las modificaciones, se procedió a realizar nuevamente la primera prueba de movilidad, ver figura 8-6, esta vez el robot fue capaz de desplazarse en línea recta de un punto a otro, sin embargo, debido a que el ángulo al que se inclinaba el robot era lo suficientemente grande como para provocar que este se volcará, se tuvieron que realizar pruebas de calibración para establecer el

correcto desfase de los ángulos pitch y roll, para que el robot fuera capaz de moverse sin caerse de la esfera.



*Figura 8-6 Preparación del robot para la primera movilidad con la nueva esfera.*

La segunda prueba consistió en desplazar al robot en una trayectoria cualquiera tal como se muestra en la figura 8-7, para lo cual se colocaron dos objetos en posiciones aleatorias con el fin de que el robot pudiera llegar a cada uno de estos dos puntos. Posteriormente se colocó el robot a cierta distancia de ellos y una vez que comenzó a funcionar y alcanzó la estabilidad sobre la esfera, se comenzó a operar a través de la computadora para poderlo llevar primero a uno de los objetos y una vez que llegó al punto de referencia, se llevó al siguiente punto. Una vez comprobado que el robot era capaz de moverse en dichas trayectorias, se realizaron más pruebas de navegación cambiando tanto el lugar de los objetos de referencia como la distancia entre ellos y el robot. En estas pruebas se observó que el robot no tuvo dificultad para alcanzar los puntos deseados.



*Figura 8-7 Acomodo de los objetos para la segunda prueba de movilidad.*

---

## RESULTADOS Y CONCLUSIONES

---

El objetivo planteado al inicio del presente trabajo, el cual consistía en lograr la estabilidad y la navegación incipiente del ballbot, se logró mediante el rediseño de todo el robot, empezando por la parte mecánica, en la cual se tuvieron que cambiar las ruedas omnidireccionales, el tipo de motor y la estructura. Posteriormente se modificó la parte electrónica, en la que principalmente se revisaron los ajustes que debían de realizarse debido al cambio de los motores y también se mejoró la señal obtenida por la IMU. Finalmente, en la parte lógica del robot, se implementó un algoritmo de control PID en el cual se controlaron los motores mediante señales PWM y con una relación de velocidad para regresar al robot a su estado estable. Se consiguió la estabilidad del robot por un tiempo aproximado de siete minutos en una esfera de uniceal a la cual se le aplicó un acabado superficial con el fin de aumentar su peso y para que no se dañara con las ruedas del robot. Asimismo, el robot fue probado en otra esfera de uniceal, a la cual se le agregaron canicas en el interior, con lo que el robot mejoró su

desempeño, logrando la estabilidad por más tiempo. Con la última esfera se logró implementar un algoritmo de movilidad mediante comunicación inalámbrica con la computadora y para esto se desarrolló una interfaz para que el usuario manipulara el robot. A continuación, se mencionan algunas observaciones que podrían servir como punto de partida para mejorar el desempeño del robot:

- A pesar de que se consiguió la navegación del robot, la cual fue implementada con una esfera rellena de canicas; podría mejorarse la robustez del algoritmo de control con el fin de lograr una navegación con una esfera sin canicas, lo cual se intentó, pero el robot no era capaz de mantenerse sobre la esfera.
- Aunque el controlador de par fue uno de los objetivos del proyecto, este no pudo ser implementado de manera satisfactoria; esto se pudo deber a diversas razones, por ejemplo que las ruedas del robot en ángulos de inclinación muy grandes perdían el contacto con la esfera, lo cual ocasionaba que el control no funcionara de manera adecuada. Otra razón importante es que no se podía validar de manera total el modelo con respecto al sistema real, por lo que las variables tuvieron que ser sintonizadas de manera empírica. Aunado a esto, no se revisó de qué forma discretizar el modelo. Sin embargo, es importante mencionar que en las pruebas realizadas se logró validar el modelo de un motor de corriente directa, asimismo se logró implementar un control por par en cada uno de los motores.
- Se observó que tanto el peso como el tamaño de la estructura en relación con el peso y el tamaño de la esfera influye mucho en la estabilización, pues en las pruebas con la estructura de aluminio, ésta se estabilizó menos tiempo en comparación con la estructura de acrílico.
- Con base en las pruebas realizadas, se concluyó que el ángulo de inclinación de los motores no jugaba un papel tan importante para lograr la estabilización, es decir, no existía un ángulo óptimo; sin embargo, este ángulo no debe de estar cerca de los  $90^\circ$  ni de los  $0^\circ$ . A pesar de eso, el ángulo de inclinación de los motores debe de ser considerado para establecer el algoritmo de navegación, pues así se obtiene una relación de movimiento entre las tres ruedas y el movimiento que se desea en el robot.
- Se observó que la superficie de las dos esferas utilizadas en el último diseño del ballbot no eran totalmente uniformes, lo cual también afectaba en la estabilidad del robot.
- El par que se calculó para los motores que se utilizaron en la primera estructura que se armó, estaba muy por encima del que realmente se demandaba en el sistema real.
- Aunque se aumentó el tiempo de estabilidad del ballbot utilizando la esfera sin canicas, el controlador implementado no es del todo robusto, esto debido a que no en todas las pruebas era capaz de mantenerse en equilibrio por un largo periodo de tiempo, además, ligeras modificaciones en las constantes, o una modificación al programa, generaba un cambio drástico en el desempeño del robot.

---

## TRABAJO A FUTURO

---

Si bien el sistema que se presenta en este trabajo tuvo un buen desempeño en las pruebas de estabilidad y movilidad, se plantean a continuación algunas tareas que podrían ayudar a mejorar su funcionamiento.

- Probar algoritmos de estabilización y navegación distintos a los que se implementaron en este proyecto con el fin de mejorar el desempeño del robot. Se sugiere utilizar un controlador por realimentación de estados, el cual únicamente fue simulado para la estructura de aluminio, o si se requiere un controlador más robusto como uno no lineal. Es importante mencionar que, para la implementación del controlador, se debe validar el modelo que se está ocupando.
- Si se desea continuar con la idea de la implementación de un controlador por par en los motores, se considera importante revisar de qué manera se puede asegurar el contacto entre las ruedas y la esfera.
- Como se menciona en este trabajo, para implementar el algoritmo de navegación fue necesario utilizar una esfera con canicas en el interior. Si se mejoran las características del robot, tanto en su parte física como en su parte lógica, mejorando así su estabilidad, se podría prescindir de las canicas. Por otra parte, se podrían mejorar las características de la esfera con el fin de que esta sea más uniforme, refiriéndose a que el centro de masa coincida con su centro geométrico, que la esfera sea completamente redonda o que la superficie no tenga grietas o imperfecciones.
- Se podría cambiar la tarjeta controladora por otra que tenga mejores características; este cambio se sugiere pues si se desea utilizar algún controlador más robusto, puede que el Arduino Due no sea suficiente para implementarlo. Además, cabe la posibilidad de elegir una placa con un

protocolo de comunicación integrado como Wi-Fi, el cual podría mejorar el rango de alcance y la tasa de transferencia de datos. Se puede consultar en el apéndice 5 una tabla comparativa de algunas placas disponibles mercado, algunas de estas opciones cuentan con comunicación integrada.

- Se puede rediseñar la estructura con el fin de hacer un buen acomodo de componentes, tanto de los que ya se tienen como de los que se vayan agregando posteriormente. Por otro lado, el tipo de material puede ser el mismo, con el fin de que la estructura siga siendo ligera, pero se sugiere hacer un análisis FEM para saber si la estructura soportaría los esfuerzos a los que estaría sometida, ya que se observó que la última estructura que se utilizó era más frágil que la de aluminio y en algunas partes esta podría llegar a romperse si es sometida a esfuerzos mucho mayores.
- Los motores que se utilizaron al final fueron elegidos con base en un trabajo ya realizado, en el cual se tenía un robot con características electrónicas similares, pero con una estructura más ligera. En caso de cambiar la estructura con la que se cuenta actualmente, sería necesario hacer un análisis de las características que se requieren en los motores para que el robot funcione adecuadamente.
- Se puede mejorar la interfaz gráfica con el fin de que ésta sea más intuitiva para el operador del robot; asimismo, se podría intentar monitorear las señales de los sensores con el fin de tener un mejor control del robot.

## A1. Parámetros para el diseño de correderas y soportes de motores

Para asegurar la tangencia de las ruedas con respecto a la pelota, las correderas y los soportes para sujetar el motor fueron diseñados con base en las dimensiones tanto de los motores como de las ruedas y las bridas. En la figura A-1 se muestran las distancias que se utilizaron, es importante mencionar que los puntos A y B representan la posición de los ejes que sujetan al motor.

- Distancia entre el centro de la rueda y el centro de la esfera (Línea morada)
- Distancia entre el centro de la rueda y la cara frontal del motor (Línea roja)
- Distancia del centro del motor al punto A (Línea negra)
- Distancia del centro del motor al punto B (Línea rosa)
- Distancia que indica la separación entre los puntos A y B (Línea amarilla)
- Línea de graduación (Línea azul)
- Superficie de la esfera (Círculo naranja)
- Trayectoria de la corredera (Círculo verde)

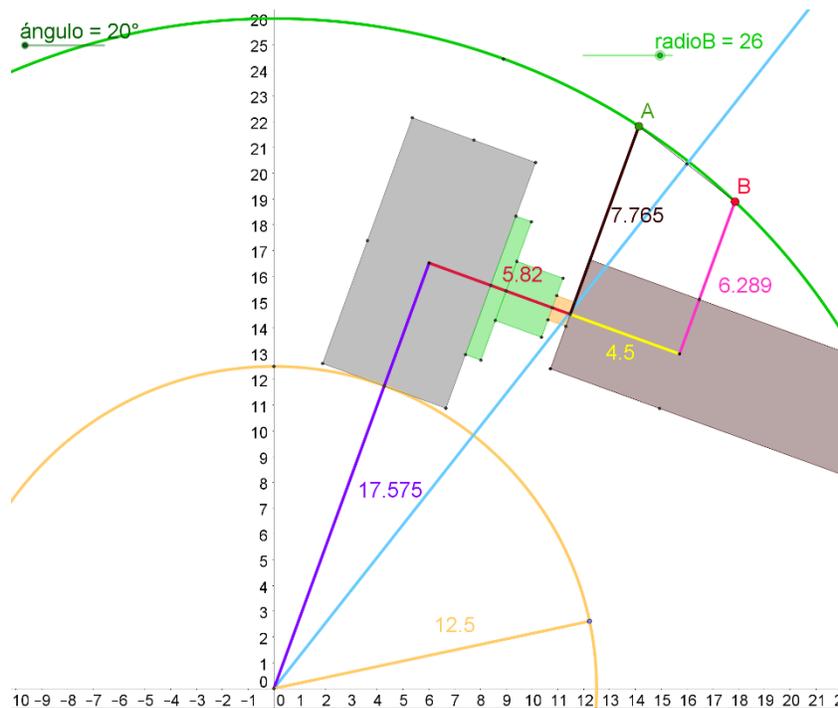
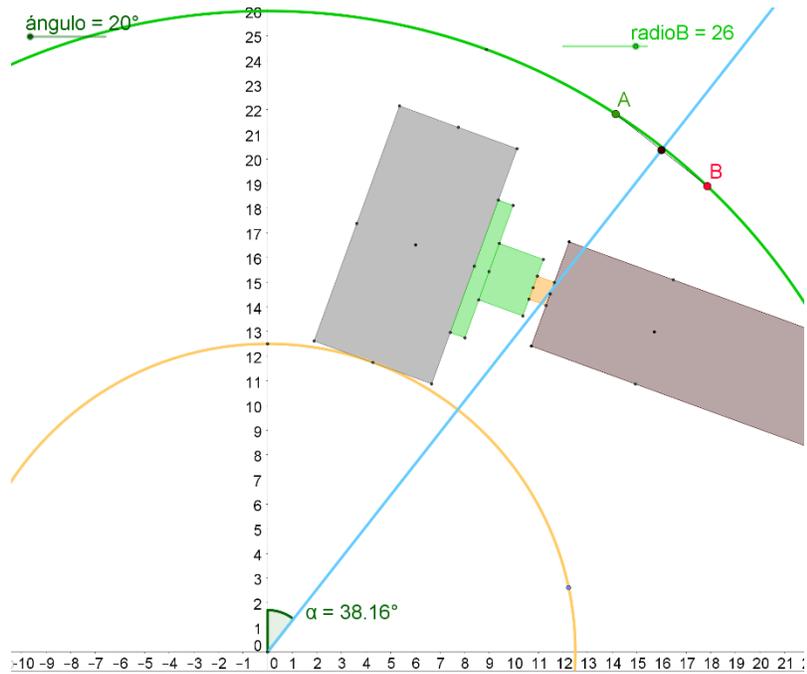
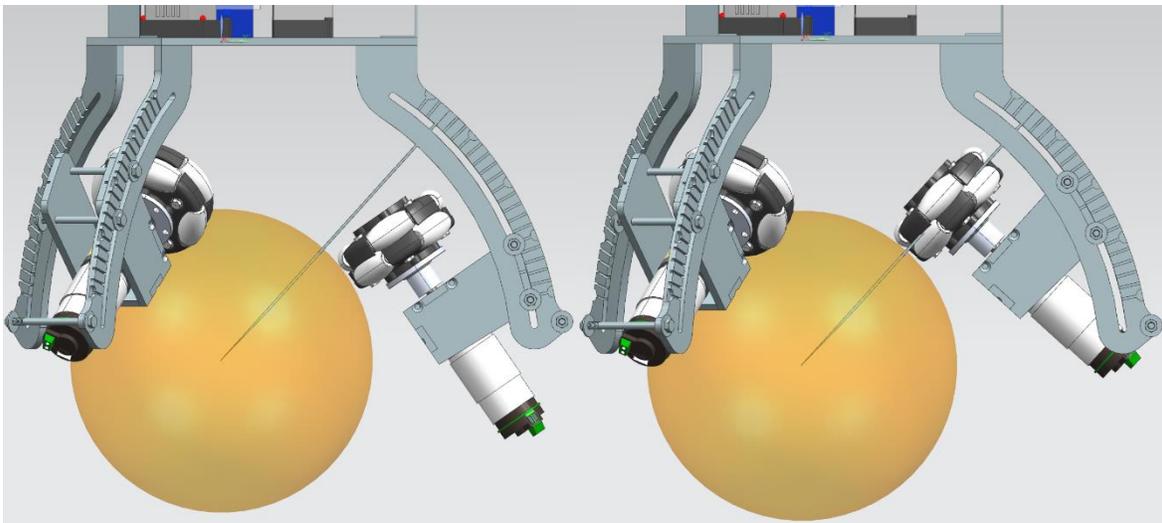


Figura A-1 Distancias utilizadas para el diseño de la estructura en cm.

Para poder colocar las ruedas a una inclinación deseada, se realizó una graduación en cada una de las correderas, con una resolución de 2 grados. Se debe de mencionar que el ángulo de las líneas de graduación no corresponde al ángulo que tiene la rueda, pues fue diseñado de esa forma con el fin de simplificar el diseño. Para entender mejor este desfase, se puede revisar la figura A-2 en la que se puede observar que la marca que indica los 20° que tiene la rueda con respecto a la vertical, se encuentra a 38.16°.



**Figura A-2** Desfase entre el ángulo de inclinación de la rueda y el ángulo de la marca de la graduación.



**Figura A-3** Verificación de la tangencia de las ruedas y la graduación de las correderas.

## A2. Deducción de las energías para el modelo del sistema

Para obtener tanto la energía cinética como la potencial de los componentes que conforman al robot, primero se obtienen las posiciones en el plano  $YZ$  de cada uno de los elementos del robot en función de las coordenadas generalizadas, estas se muestran a continuación de forma matricial:

$$\begin{bmatrix} y_e \\ z_e \end{bmatrix} = \begin{bmatrix} r_e \theta_x \\ 0 \end{bmatrix} \quad (\text{A.1})$$

$$\begin{bmatrix} y_c \\ z_c \end{bmatrix} = \begin{bmatrix} r_e \theta_x + L \text{sen}(\varphi_x) \\ L \text{cos}(\varphi_x) \end{bmatrix} \quad (\text{A.2})$$

$$\begin{bmatrix} y_r \\ z_r \end{bmatrix} = \begin{bmatrix} r_e \theta_x + (r_e + r_r) \text{sen}(\varphi_x) \\ (r_e + r_r) \text{cos}(\varphi_x) \end{bmatrix} \quad (\text{A.3})$$

En la figura A-4 se muestran los diagramas que se utilizaron para determinar dichas posiciones y en la tabla A-1 se explican las variables involucradas.

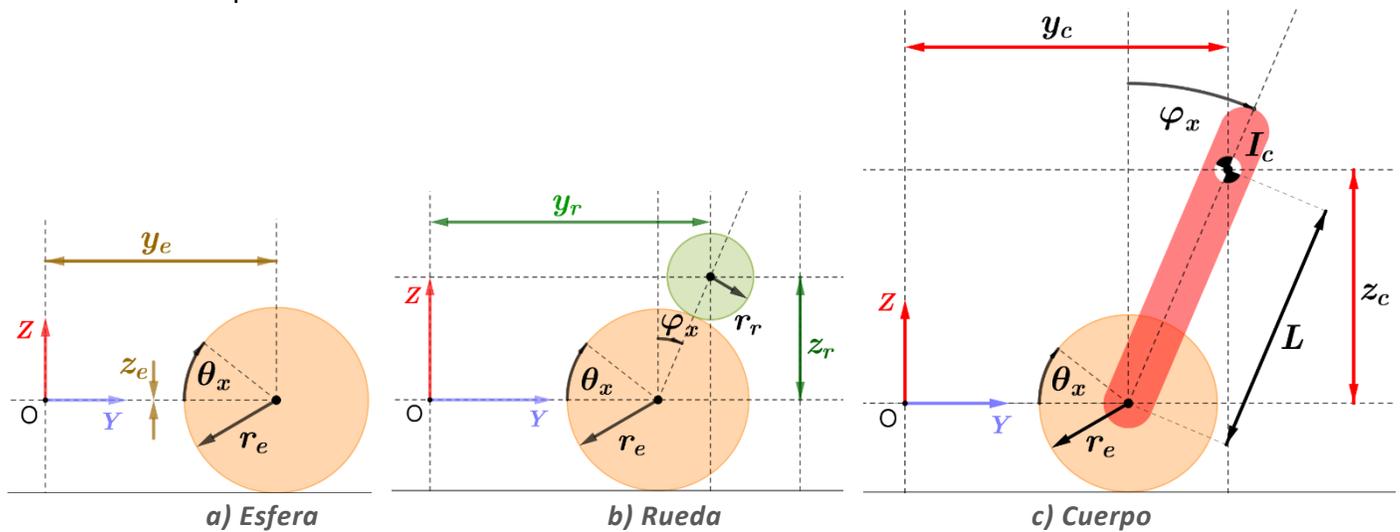


Figura A-4 Posición de cada elemento del robot.

Tabla A-1 Significado de variables para análisis de desplazamiento

$y_e, z_e$	Desplazamiento tanto en $y$ como en $z$ de la esfera
$y_c, z_c$	Desplazamiento tanto en $y$ como en $z$ del cuerpo del robot
$y_r, z_r$	Desplazamiento tanto en $y$ como en $z$ de la rueda
$r_e$	Radio de la esfera
$r_r$	Radio de la rueda
$L$	Distancia entre el centro de masa del cuerpo del robot y centro de masa de la esfera
$\theta_x$	Desplazamiento angular de la esfera

$\varphi_x$	Desplazamiento angular del robot
-------------	----------------------------------

Las ecuaciones mostradas fueron utilizadas para el desarrollo que se muestra a continuación de las ecuaciones de energía que se emplearon en el modelado. La ecuación final de cada una de las energías se mostrará en negritas.

### **Para la esfera**

La energía cinética de la esfera está conformada por la energía debida tanto a su rotación, como a su traslación, por lo que se expresa de la siguiente manera:

$$T_e = \frac{1}{2}m_e v_e^2 + \frac{1}{2}I_e \dot{\theta}_x^2 \quad (\text{A.4})$$

donde  $m_e$  es la masa de la esfera,  $v_e$  es la rapidez tangencial de la esfera,  $I_e$  es el momento de inercia que tiene la esfera con respecto a un eje perpendicular al plano que pasa por su centro de masa y  $\dot{\theta}_x$  es la rapidez angular de la esfera.

La rapidez tangencial de la esfera también puede ser vista como:

$$v_e = r_e \dot{\theta}_x \quad (\text{A.5})$$

Por lo que al sustituir la ecuación A.5 en A.4, la energía cinética de la esfera queda de la siguiente manera:

$$T_e = \frac{1}{2}m_e r_e^2 \dot{\theta}_x^2 + \frac{1}{2}I_e \dot{\theta}_x^2 \quad (\text{A.6})$$

Se asume que la energía potencial de la esfera es 0, pues esta solo se mueve sobre la superficie horizontal.

$$V_e = 0 \quad (\text{A.7})$$

### **Para el cuerpo del robot**

La energía cinética del cuerpo del robot se obtiene de manera análoga a la energía cinética de la esfera.

$$T_c = \frac{1}{2}m_c v_c^2 + \frac{1}{2}I_c \dot{\phi}_x^2 \quad (\text{A.8})$$

donde  $m_c$  es la masa del cuerpo,  $v_c$  es la rapidez del cuerpo,  $I_c$  es el momento de inercia que tiene el cuerpo del robot con respecto a un eje perpendicular al plano que pasa por su centro de masa y  $\dot{\varphi}_x$  es la rapidez angular del cuerpo.

La rapidez del cuerpo elevada al cuadrado también puede ser vista como:

$$v_c^2 = \dot{y}_c^2 + \dot{z}_c^2 \quad (\text{A.9})$$

Sustituyendo  $y_c$  y  $z_c$  de la ecuación de posición A.2 en A.9, se obtiene la siguiente expresión:

$$v_c^2 = \left[ \frac{d}{dt} (r_e \theta_x + L \sin \varphi_x) \right]^2 + \left[ \frac{d}{dt} (L \cos \varphi_x) \right]^2 \quad (\text{A.10})$$

$$v_c^2 = [r_e \dot{\theta}_x + L \dot{\varphi}_x \cos \varphi_x]^2 + [-L \dot{\varphi}_x \sin \varphi_x]^2 \quad (\text{A.11})$$

$$v_c^2 = r_e^2 \dot{\theta}_x^2 + 2r_e L \dot{\theta}_x \dot{\varphi}_x \cos \varphi_x + L^2 \dot{\varphi}_x^2 \quad (\text{A.12})$$

Por lo que al sustituir  $v_c^2$  en la ecuación A.8 de la energía cinética del cuerpo, esta queda de la siguiente manera:

$$T_c = \frac{1}{2} m_c (r_e^2 \dot{\theta}_x^2 + 2r_e L \dot{\theta}_x \dot{\varphi}_x \cos \varphi_x + L^2 \dot{\varphi}_x^2) + \frac{1}{2} I_c \dot{\varphi}_x^2 \quad (\text{A.13})$$

De acuerdo al teorema de ejes paralelos se puede obtener el momento de inercia del cuerpo con respecto a otro eje de giro que en este caso puede ser un eje paralelo al plano que pase por el centro de la esfera; a este momento se le llamará  $I_{co}$ .

$$I_{co} = I_c + m_c L^2 \quad (\text{A.14})$$

Dado que  $T_c$  se puede escribir de la siguiente manera:

$$T_c = \frac{1}{2} m_c (r_e^2 \dot{\theta}_x^2 + 2r_e L \dot{\theta}_x \dot{\varphi}_x \cos \varphi_x) + \frac{1}{2} (I_c + m_c L^2) \dot{\varphi}_x^2 \quad (\text{A.15})$$

$$T_c = \frac{1}{2} m_c (r_e^2 \dot{\theta}_x^2 + 2r_e L \dot{\theta}_x \dot{\varphi}_x \cos \varphi_x) + \frac{1}{2} (I_{co}) \dot{\varphi}_x^2 \quad (\text{A.16})$$

Asimismo, la energía potencial del cuerpo del robot se define como:

$$V_c = m_c g L \cos \varphi_x \quad (\text{A.17})$$

donde  $m_c$  es la masa del cuerpo,  $g$  es la constante de gravedad y  $L$  es la distancia que hay entre el centro de masa del cuerpo del robot y el centro de masa de la esfera y  $\dot{\phi}_x$  es la rapidez del ángulo de inclinación del cuerpo del robot.

### **Para la rueda virtual**

La energía cinética de la rueda virtual se obtiene de la misma forma que se obtuvo tanto la energía cinética del cuerpo del robot como la energía cinética de la esfera:

$$T_r = \frac{1}{2}m_r v_r^2 + \frac{1}{2}I_r \dot{\phi}_x^2 \quad (\text{A.18})$$

donde  $m_r$  es la masa de la rueda virtual,  $v_r$  es la rapidez de la rueda virtual,  $I_r$  es el momento de inercia que tiene la rueda virtual con respecto a un eje que pasa por su centro de masa y  $\dot{\phi}_x$  es la rapidez angular de la rueda virtual.

La rapidez de la rueda virtual elevada al cuadrado también puede ser escrita como:

$$v_r^2 = \dot{y}_r^2 + \dot{z}_r^2 \quad (\text{A.19})$$

Sustituyendo  $y_r$  y  $z_r$  de la ecuación de posición A.3 en A.19,  $v_r^2$  queda de la siguiente forma:

$$v_r^2 = \left[ \frac{d}{dt}(r_e \theta_x + (r_e + r_r) \text{sen} \varphi_x) \right]^2 + \left[ \frac{d}{dt}((r_e + r_r) \text{cos} \varphi_x) \right]^2 \quad (\text{A.20})$$

$$v_r^2 = [r_e \dot{\theta}_x + (r_e + r_r) \dot{\phi}_x \text{cos} \varphi_x]^2 + [-(r_e + r_r) \dot{\phi}_x \text{sen} \varphi_x]^2 \quad (\text{A.21})$$

$$v_r^2 = r_e^2 \dot{\theta}_x^2 + 2r_e(r_e + r_r) \dot{\theta}_x \dot{\phi}_x \text{cos} \varphi_x + (r_e + r_r)^2 \dot{\phi}_x^2 \quad (\text{A.22})$$

Al sustituir la ecuación A.22 en la ecuación de la energía cinética de la rueda virtual A.18, esta queda de la siguiente manera:

$$T_r = \frac{1}{2}m_r \left( r_e^2 \dot{\theta}_x^2 + 2r_e(r_e + r_r) \dot{\theta}_x \dot{\phi}_x \text{cos} \varphi_x + (r_e + r_r)^2 \dot{\phi}_x^2 \right) + \frac{1}{2}I_r \dot{\phi}_x^2 \quad (\text{A.23})$$

Sin embargo, el ángulo  $\phi_x$  puede obtenerse mediante los ángulos  $\theta_x$  y  $\varphi_x$  gracias a que existe una relación en el movimiento de las tres partes del sistema. Con base en la figura 6-2 que se muestra en la parte del modelado del sistema, se verifica que si el ángulo  $\varphi_x$  se deja fijo y  $\theta_x$  aumenta,  $\phi_x$  aumentará de manera proporcional a la relación que existe entre el radio de la esfera y el radio de la rueda virtual.

Asimismo, si el ángulo  $\theta_x$  se deja fijo y  $\varphi_x$  aumenta,  $\phi_x$  disminuirá de manera proporcional a la relación que existe entre el radio de la esfera y el radio de la rueda virtual. Lo anterior puede expresarse mediante las siguientes expresiones:

$$\phi_x = \frac{r_e}{r_r} \theta_x \quad (\text{A.24})$$

$$\phi_x = -\frac{r_e}{r_r} \varphi_x \quad (\text{A.25})$$

En dichas ecuaciones, no se describe la relación completa que existe entre el ángulo  $\phi_x$  y los ángulos  $\theta_x$ ,  $\varphi_x$ , pues se deja fijo alguno de los dos ángulos; sin embargo, se puede deducir que la relación completa queda de la siguiente manera:

$$\phi_x = \frac{r_e}{r_r} (\theta_x - \varphi_x) \quad (\text{A.26})$$

Y por lo tanto  $\dot{\phi}_x$  se puede expresar de la siguiente manera:

$$\dot{\phi}_x = \frac{r_e}{r_r} (\dot{\theta}_x - \dot{\varphi}_x) \quad (\text{A.27})$$

Por lo que finalmente la ecuación de la energía cinética de la rueda virtual  $T_r$  se puede expresar como:

$$T_r = \frac{1}{2} m_r (r_e^2 \dot{\theta}_x^2 + 2r_e(r_e + r_r) \dot{\theta}_x \dot{\varphi}_x \cos \varphi_x + (r_e + r_r)^2 \dot{\varphi}_x^2) + \frac{1}{2} I_r \left( \frac{r_e}{r_r} (\dot{\theta}_x - \dot{\varphi}_x) \right)^2 \quad (\text{A.28})$$

La energía potencial de la rueda virtual se obtiene del mismo modo que la energía potencial del cuerpo del robot:

$$V_r = m_r g (r_r + r_e) \cos \varphi_x \quad (\text{A.29})$$

### A3. Cálculo de los parámetros del modelo

Para calcular los parámetros que se necesitaron en el modelado de la primera estructura que se diseñó en este proyecto, se consideró que el sistema está compuesto de una esfera, tres actuadores y la estructura que conforma el cuerpo del robot.

#### Masas y radios

Entre los parámetros que se pudieron obtener de manera sencilla, se encuentran las masas de cada uno de los componentes y el radio de la esfera. Asimismo, se consideró que el radio de la rueda virtual (parámetro que está presente en el modelado) y el radio de las ruedas omnidireccionales es el mismo.

## Centros de masa

Se consideró que el centro de masa de la esfera y el centro geométrico de la esfera es el mismo, por lo que no se requirió de un análisis para poder ubicarlo. Por otro lado, se procedió a calcular la posición del centro de masa del cuerpo del robot, con objeto de obtener la distancia entre este y el centro de masa de la esfera (parámetro al cual se le denominó  $L$ ); para esto se utilizó el principio de suma de momentos que se muestra en la figura A-5. Se debe de mencionar que, para poder calcular el centro de masa de manera espacial, dicho principio se aplicó en tres planos ortogonales.

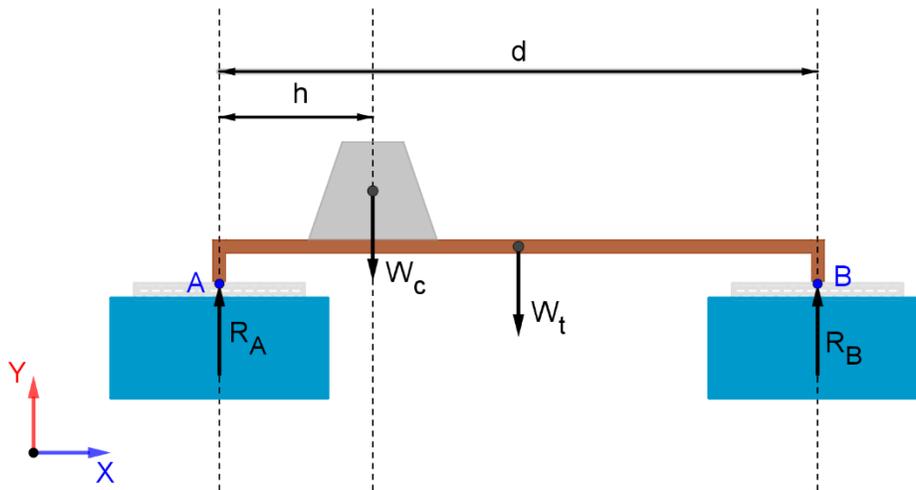


Figura A-5 Diagrama para el cálculo del centro de masa en un plano XY.

En la figura anterior se muestra que, para obtener la posición del centro de masa de un cuerpo, se obtiene la suma de momentos tanto en el punto A, como en el punto B. Si el cuerpo está sostenido por una base de peso considerable, se debe de tomar en cuenta para hacer el análisis. Se le denominó al peso de la base  $W_t$ , al peso del cuerpo  $W_c$ ,  $R_A$  y  $R_B$  a las reacciones en el punto A y punto B respectivamente,  $d$  a la distancia entre el punto A y punto B y finalmente  $h$  a la distancia en  $X$  que existe entre el punto A y el centro de masa del cuerpo. Dicho lo anterior, se pueden obtener las siguientes ecuaciones:

$$\sum M_A: -W_c h - W_t \frac{d}{2} + R_B d = 0 \quad (\text{A.30})$$

$$\sum M_B: W_t \frac{d}{2} + W_c (d - h) - R_A d = 0 \quad (\text{A.31})$$

De la ecuación A.30 se obtiene que:

$$h = \frac{d}{W_c} \left( -\frac{W_t}{2} + R_B \right) \quad (\text{A.32})$$

Y de la ecuación A.31 se obtiene que:

$$h = \frac{d}{W_c} \left( \frac{W_t}{2} + W_c - R_A \right) \quad (\text{A.33})$$

Sumando tanto el lado derecho como el lado izquierdo de las dos ecuaciones anteriores, se obtiene la siguiente expresión:

$$2h = \frac{d}{W_c} (R_B + W_c - R_A) \quad (\text{A.34})$$

que es lo mismo que:

$$h = \frac{d}{2} \left( \frac{R_B - R_A}{W_c} + 1 \right) \quad (\text{A.35})$$

donde los valores de  $R_A$  y  $R_B$  pueden obtenerse mediante básculas o dinamómetros colocados en los puntos A y B respectivamente y  $W_c$  es el peso del cuerpo del robot.

Para poder realizar las medidas de manera adecuada se tomaron los platos del robot como referencia para poder orientar al robot. Una vez obtenido el centro de masa del robot, se obtuvo la distancia  $L$  que existe entre el centro de masa y el centro de la esfera del robot, que se corroboró con ayuda del CAD.

### **Momento de inercia**

Para calcular el momento de inercia de la esfera de madera que se utilizó para la primera estructura que se diseñó, se tomó la ecuación del momento de inercia de una esfera sólida completamente uniforme, la cual se muestra a continuación:

$$I_e = \frac{2}{5} m_e r_e^2 \quad (\text{A.36})$$

Por otra parte, si se utilizara una esfera hueca como la que se utilizó en la segunda estructura que se diseñó, la inercia de la esfera se calcularía de la siguiente manera:

$$I_e = \frac{2}{3} m_e r_e^2 \quad (\text{A.37})$$

El cuerpo del robot fue puesto en una base con el fin de hacerlo oscilar como si fuera un péndulo; de esta forma, mediante un análisis dinámico, se pudo obtener su momento de inercia con respecto al eje de oscilación tanto en el eje X como en el Y, así como se muestra en la figura A-6.

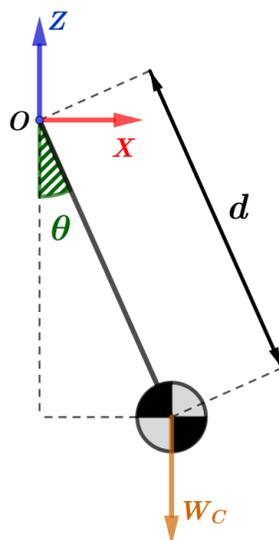


Figura A-6 Diagrama del péndulo.

Analizando los momentos generados en el punto O de la figura A-6, se puede obtener la siguiente ecuación:

$$\curvearrowright \Sigma M_o: -W_C d \text{sen} \theta = I_o \alpha \quad (\text{A.38})$$

Donde  $I_o$  es el momento de inercia del cuerpo del robot con respecto a un eje de giro perpendicular al plano que pasa por el punto O.

Para ángulos pequeños se puede considerar que  $\text{sen} \theta \approx \theta$ , por lo que la ecuación (A.38) puede expresarse como:

$$I_o \alpha + m_c g d \theta = 0 \quad (\text{A.39})$$

O bien:

$$\ddot{\theta} + \frac{m_c g d}{I_o} \theta = 0 \quad (\text{A.40})$$

Ya que la ecuación anterior es una ecuación diferencial de segundo orden que corresponde a un movimiento armónico simple, la frecuencia angular al cuadrado del sistema es:

$$\omega_2^2 = \frac{m_c g d}{I_o} \quad (\text{A.41})$$

Por lo que:

$$I_o = \frac{m_c g d}{\omega_2^2} \quad (\text{A.42})$$

Para la ecuación anterior, la frecuencia angular del sistema se pudo obtener de manera experimental haciendo oscilar el cuerpo del robot, de tal modo que se pudo medir su periodo de oscilación, con ayuda de un sensor, para así obtener su frecuencia, la cual se multiplicó por  $2\pi$ .

Finalmente, para obtener el momento de inercia del cuerpo del robot con respecto a un eje centroidal paralelo al eje de oscilación, al cual se le nombró  $I_c$ , se utilizó el teorema de ejes paralelos:

$$I_o = I_c + d^2 m_c \quad (\text{A.43})$$

Por lo que:

$$I_c = I_o - d^2 m_c \quad (\text{A.44})$$

Para estimar el momento de inercia de la rueda virtual  $I_r$ , se consideró que la energía cinética de la rueda virtual debida a la rotación es igual a la energía cinética de las tres ruedas omnidireccionales, por lo que  $T_r$  se puede expresar de las siguientes dos formas :

$$T_r = \frac{1}{2} I_r \dot{\phi}_x^2 \quad (\text{A.45})$$

$$T_r = \frac{1}{2} (I_a \omega_{x,r1}^2 + I_a \omega_{x,r2}^2 + I_a \omega_{x,r3}^2) \quad (\text{A.46})$$

donde  $\dot{\phi}_x$  es la velocidad angular de la rueda virtual,  $\omega_{x,ri}$  es la componente en X de la velocidad angular de cada una de las ruedas omnidireccionales e  $I_a$  es el momento de inercia de cada actuador del robot, el cual se considera que está compuesto por la rueda y el motor.

Igualando las ecuaciones A.45 y A.46 se obtiene que:

$$I_r \dot{\phi}_x^2 = I_a (\omega_{x,r1}^2 + \omega_{x,r2}^2 + \omega_{x,r3}^2) \quad (\text{A.47})$$

El momento de inercia  $I_a$  puede calcular de la siguiente manera:

$$I_a = (k^2 I_{rotor} + I_{ro}) \quad (\text{A.48})$$

donde  $I_{ro}$  es el momento de inercia de la rueda omnidireccional con respecto a su eje de giro,  $k$  es la reducción del motor e  $I_{rotor}$  es el momento de inercia propio del rotor del motor. Se hizo la suposición de que cada rueda omnidireccional era como un cilindro sólido por lo que  $I_{ro}$  se calculó como:

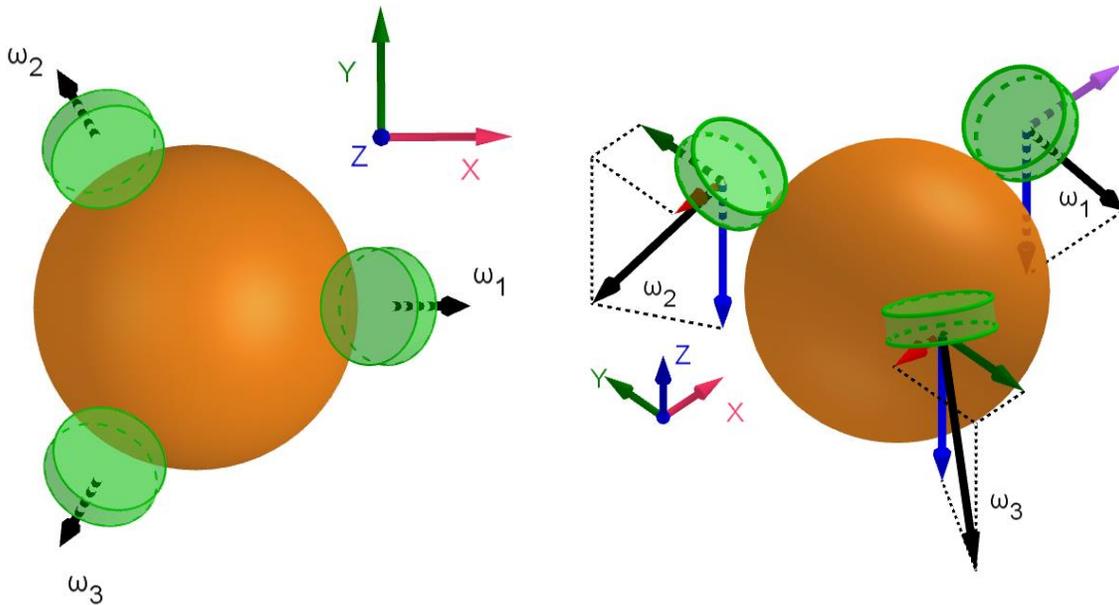
$$I_{ro} = \frac{m_{ro}r_{ro}^2}{2} \quad (A.49)$$

Por lo que la ecuación (A.47) queda de la siguiente forma:

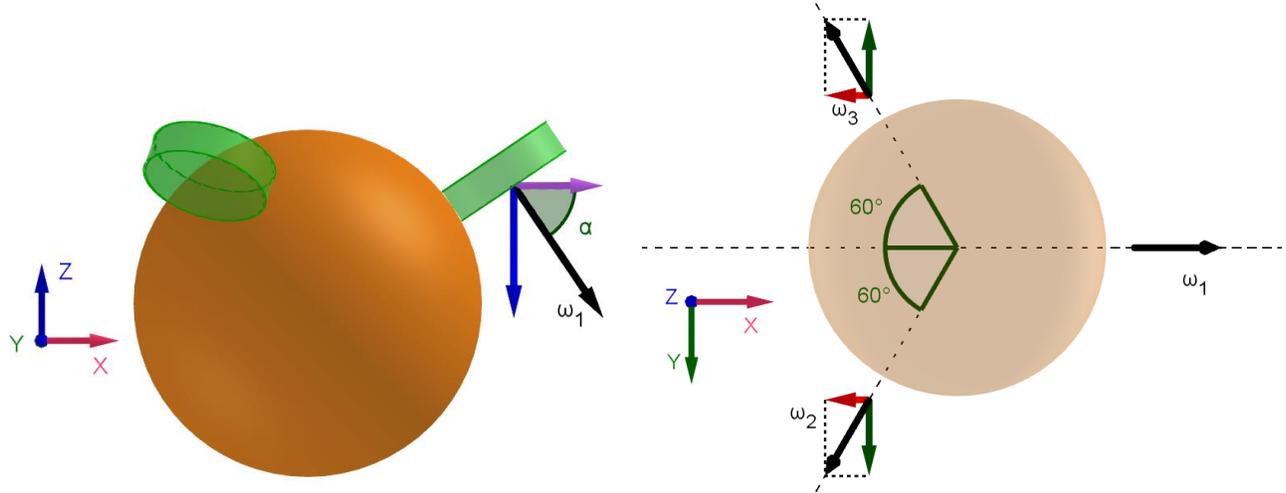
$$I_r \dot{\phi}_x^2 = (k^2 I_{Motor} + \frac{m_{ro}r_{ro}^2}{2})(\omega_{x,r1}^2 + \omega_{x,r2}^2 + \omega_{x,r3}^2) \quad (A.50)$$

Es importante mencionar que solo se utilizaron las componentes en  $X$  de las velocidades de cada una de las ruedas omnidireccionales porque el análisis de la energía cinética solo se hizo para el plano que se utilizaría en el modelo, el cual correspondería al plano sobre el cual se inclinaría el robot.

Para calcular los componentes de las velocidades angulares de cada rueda omnidireccional, con el objetivo de sustituir los valores en la ecuación A.50 se utilizaron las figuras A-7 y A-8 que se muestran a continuación.



*Figura A-7 Velocidades angulares en las ruedas y sus componentes en X, Y y Z.*



**Figura A-8** Ángulos para determinar el valor de las componentes en X, Y y Z de las velocidades angulares.

En la figura A-7 y A-8, la velocidad angular de cada una de las ruedas omnidireccionales están representadas por los vectores de color negro  $\omega_i$  y sus componentes están representados por los vectores de colores acordes al sistema de referencia que se muestra en cada imagen.

Los componentes en X de las velocidades angulares de las ruedas son:

$$\omega_{x,r1} = \dot{\phi}_x \cos(\alpha) \quad (\text{A.51})$$

$$\omega_{x,r2} = -\dot{\phi}_x \cos(60^\circ) \cos(\alpha) ; \omega_{x,r2} = -\frac{1}{2} \dot{\phi}_x \cos(\alpha) \quad (\text{A.52})$$

$$\omega_{x,r3} = -\dot{\phi}_x \cos(60^\circ) \cos(\alpha) ; \omega_{x,r3} = -\frac{1}{2} \dot{\phi}_x \cos(\alpha) \quad (\text{A.53})$$

Teniendo los componentes de las velocidades angulares en X, se obtiene que:

$$\omega_{x,r1}^2 + \omega_{x,r2}^2 + \omega_{x,r3}^2 = \frac{3}{2} \dot{\phi}_x^2 \cos^2 \alpha \quad (\text{A.54})$$

Sustituyendo la ecuación (a.54), en la ecuación (a.50), se tiene que:

$$I_r \dot{\phi}_x^2 = \frac{3}{2} \left( k^2 I_{Motor} + \frac{m_{ro} r_{ro}^2}{2} \right) \dot{\phi}_x^2 \cos^2 \alpha \quad (\text{A.56})$$

Por lo que la Inercia de la rueda virtual puede ser vista como:

$$I_r = \frac{3}{2} \left( k^2 I_{Motor} + \frac{m_{ro} r_{ro}^2}{2} \right) \cos^2 \alpha \quad (\text{A.57})$$

Se supuso el mismo valor del momento de inercia de la rueda virtual  $I_r$  del modelo para cualquier plano en el que se inclinara el robot, ya que si se sacaban las componentes no solo en  $X$ , sino también en  $Y$  de las velocidades de cada una de las ruedas omnidireccionales,  $\omega_{y,ri}$ , con ayuda de las figuras A-7 y A-8 el valor ahora para  $\omega_{y,r1}^2 + \omega_{y,r2}^2 + \omega_{y,r2}^2$  sería igualmente  $\frac{3}{2}\dot{\phi}_x^2 \cos^2\alpha$ , por lo que se supuso que ese valor no cambiaba. Es importante mencionar que no se obtuvieron las componentes en el eje  $Z$  de las velocidades angulares de cada rueda pues se consideró que para estabilizar al robot, no sería necesario que el robot rotase en esa dirección.

#### **A4. Simulación de controlador por realimentación de estados**

Como se menciona en el capítulo 6, se realizó una simulación de un algoritmo de control por realimentación de estados, tomando como referencia los parámetros obtenidos del robot de manera experimental; para esto se siguió utilizando el modelo en 2D del robot, obtenido mediante el método de Lagrange en el cual la entrada es el par de los motores y la salida es el ángulo de inclinación del robot. Se pensó en la aplicación de un control por realimentación de estados porque era un control más robusto que el PID.

Es importante mencionar que para poder aplicar el control por realimentación de estados es necesario poder medir todas las variables de estado del sistema. En el caso del sistema del robot, las variables que se deberían medir son el ángulo de inclinación del cuerpo del robot, su velocidad angular, así como el desplazamiento de la esfera y su velocidad angular. Si bien se pueden medir de manera sencilla el ángulo de inclinación del cuerpo robot como su velocidad angular, el movimiento de la esfera no se puede determinar de manera sencilla y solo se podría estimar con ayuda de los decodificadores o mediante el diseño de observadores.

Se realizaron dos diagramas de bloques en Matlab para simular el algoritmo de control, mostrados en las figuras A-9 y A-10; en el segundo diagrama de bloques se agregó un observador para resolver el problema que se mencionó anteriormente. Para ambos casos el robot logró estabilizarse sin problema, tal y como se muestra en las figuras A-11 y A-12, en donde se utilizó una entrada escalón unitario y una entrada sinusoidal de 2° de amplitud.

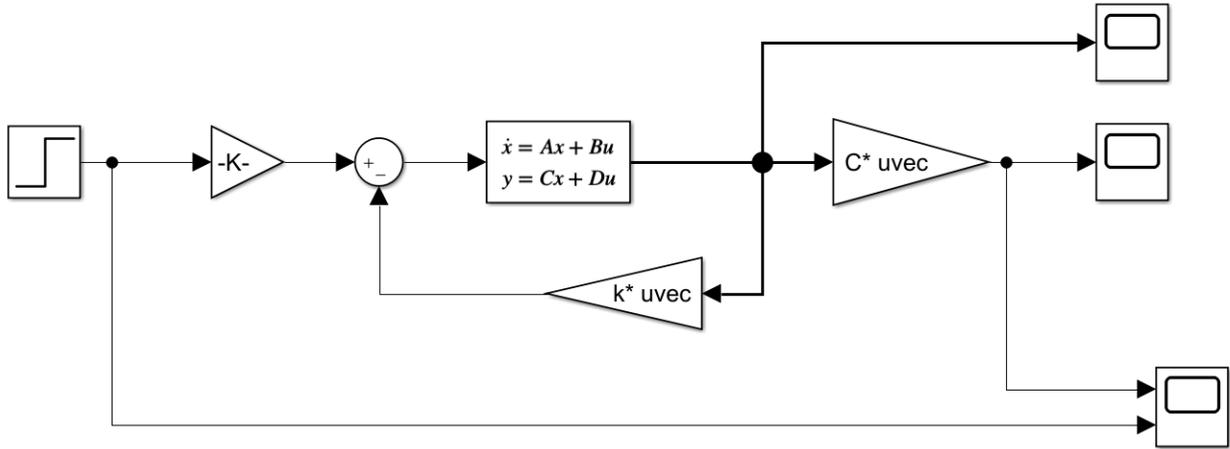


Figura A-9 Diagrama de bloques del controlador por realimentación de estados.

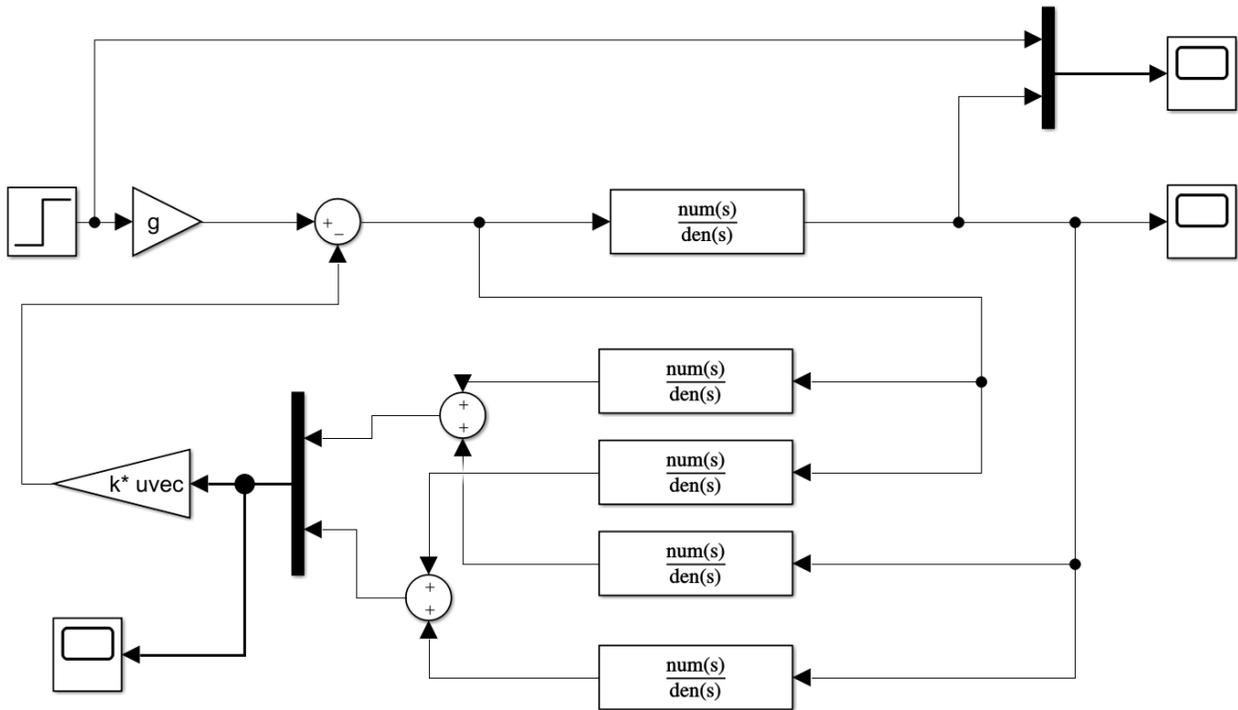
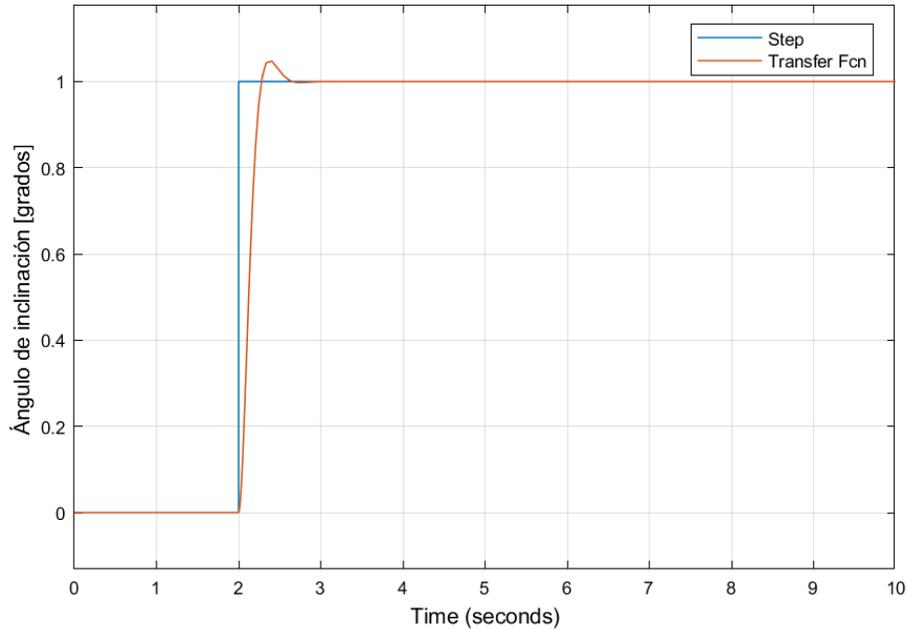
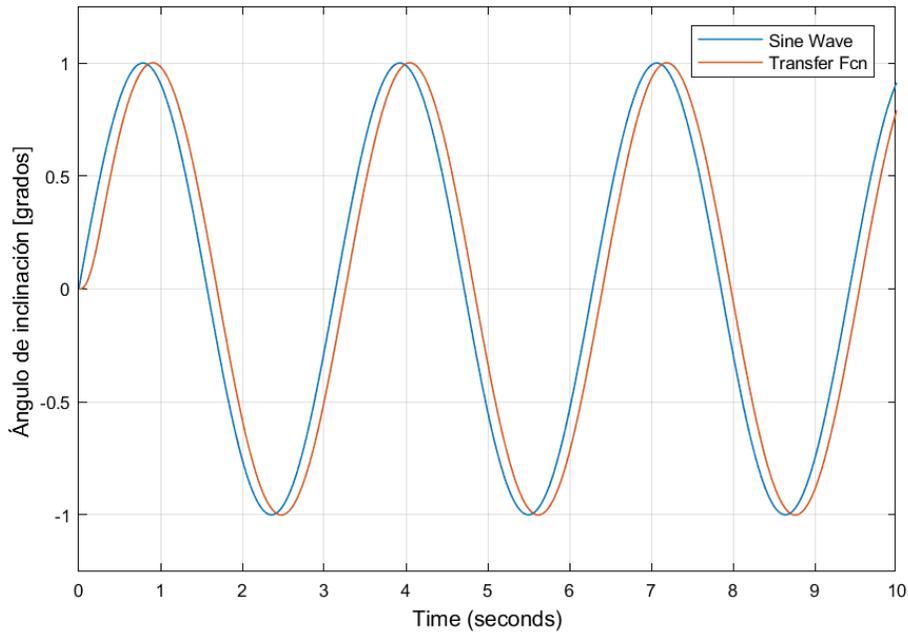


Figura A-10 Diagrama de bloques del controlador por realimentación de estados con observador.



**Figura A-11** Respuesta del sistema simulado con control por realimentación de estados, ante una entrada escalón unitario.



**Figura A-12** Respuesta del sistema simulado con control por realimentación de estados, ante una entrada sinusoidal.

Se consideró que la mejor manera de aplicar el control por realimentación de estados era mediante la implementación del observador; sin embargo, se determinó que para lograrlo sería mejor que el algoritmo de control lo hiciera una computadora en vez del microcontrolador, se propone utilizar el mismo programa de Matlab para este fin como un trabajo a futuro.

## A5. Evaluación de placas de desarrollo

Como se menciona en el trabajo a futuro, se considera conveniente cambiar el Arduino Due, que se encarga de la parte lógica, por otra placa de desarrollo para mejorar el rendimiento del robot. Se revisaron alternativas que podrían sustituir al Arduino Due, considerando que se necesitaría mejorar el algoritmo de navegación y adicionalmente se le podrían agregar sensores en el futuro. También, dado que en el mercado existen módulos que permiten establecer una comunicación Wi-Fi con un microcontrolador, o en su caso, microcontroladores con un módulo Wi-Fi integrado, resultó interesante explorar estas alternativas.

Se buscaron microcontroladores que pudieran usar un módulo Wi-Fi o que tuvieran uno integrado, que tuvieran los pines digitales y analógicos suficientes para los actuadores y sensores que se ocuparían, y además, se revisó que las placas tuvieran una velocidad cercana o mayor a los 80MHz. Asimismo, se buscaron placas que pudieran ser programadas con la IDE de Arduino, esto debido a la familiaridad que se tiene con el software y posteriormente se elaboró la tabla A-2 para comparar las características de las placas seleccionadas.

*Tabla A-2 Características de las placas de desarrollo.*

Características	Teensy 3.6	Tiva TM4C123G	Tiva TM4C1294	Arduino Due	ESP 32
Procesador	ARM Cortex M4	ARM Cortex M4	ARM Cortex M4	ARM Cortex M3	Dual Core LX6
Velocidad de reloj en MHz	180	80	120	84	240
Numero de pines Digitales	58	35	70	54	28
Memoria flash	1024 kB	256 kB	1024 kB	512 kB	4 Mb
Salidas analógicas (DAC)	2	-	-	2	2
Entradas analógicas	25	12	20	12	18
Salidas PWM	22	23	24	12	16
Puertos UART	6	8	8	4	3
Puertos I2C	4	6	10	2	2
Puertos SPI	3	4	4	1	3
Voltaje de operación en V	3.3	3.3	3.3	3.3	3.3
Voltaje de alimentación en V	5	5	5	7-12	3.3
Comunicación inalámbrica	No	No	No	No	Si
Costo (USD) con módulo incluido	36.2	19.94	26.94	44.35	19.95
Tamaño (cm x cm)	6.1 x 1.8	6.6 x 5.1	12.5 x 5.6	10.2 x 5.4	5.5 x 2.8
Lenguaje de programación	C	C	C	C	C

Asimismo, se elaboró una matriz de selección, mostrada en la tabla A-3, en la que se le asignó un nivel de importancia a cada característica; 4 es el valor más importante y 1 el menos importante; además se numeró del 1 al 6 cada placa de desarrollo en cada criterio, siendo 6 la que mejor cumplía con dicho

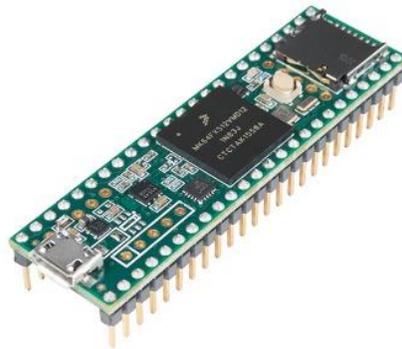
criterio. Para la selección de la nueva placa, se multiplicó el número de la importancia de cada criterio con el número que tenía cada placa en dicho criterio y se sumaron los puntos obtenidos. Una vez realizada la evaluación se concluyó que la mejor opción era la placa de desarrollo ESP-32.

*Tabla A-3 Matriz de selección de la nueva placa.*

<b>Criterios de selección</b>	Teensy 3.6	Tiva TM4C123G	Tiva TM4C1294	Arduino Due	ESP 32	Importancia
Velocidad de muestreo	4	1	3	2	5	4
Costo	2	6	4	1	5	1
Tamaño	6	4	1	2	5	2
Módulo Wi-Fi integrado	1	1	1	1	2	3
Suma de puntos	33	21	21	16	41	

Se pudieron realizar pruebas de funcionamiento con la placa ESP-32 con los actuadores y sensores del robot, en las que se observó que cada uno funcionaba de manera correcta de manera independiente, pero de manera conjunta no fue posible debido a errores de compilación en el programa. Se hizo una búsqueda de la posible razón del problema, sin embargo, no se encontró una solución.

También se pudieron realizar las mismas pruebas de funcionamiento con la placa Teensy 3.6, mostrada en la figura A-13, que salió como segunda opción en la tabla de selección. Con dicha placa no se presentaron problemas al utilizar los dispositivos del robot de manera independiente o conjunta, por lo que se concluyó que podría utilizarse en un rediseño posterior.



*Figura A-13 Placa Teensy 3.6.*

## A6. Componentes del robot

### Motores

Para el movimiento del primer diseño se utilizaron tres motores PG27 de AndyMark, como el que se muestra en la figura A-14, y para el segundo diseño se utilizaron los moto reductores Pololu 25D, ver figura A-15. Las características de ambos motores se muestran en las tablas A-4 y A-5.



*Figura A-14 Motor PG27 de AndyMark.*

*Tabla A-4 Características del motor PG27 de AndyMark.*

Masa	0.8 kg
Voltaje nominal	12 V
Velocidad sin carga	198 rpm
Corriente sin carga	0.6 A
Corriente a rotor bloqueado	22 A
Par a roto bloqueado	8.5 N·m
Reducción	26.9:1



*Figura A-15 Moto reductor Pololu 25D.*

*Tabla A-5 Características del moto reductor Pololu 25D.*

Masa	103 g
Voltaje nominal	6 V
Velocidad sin carga	130 rpm
Corriente sin carga	550 mA
Corriente a rotor bloqueado	6500 mA
Par a rotor bloqueado	1.06 N·m
Reducción	74.86:1

### **Ruedas omnidireccionales**

Las ruedas adquiridas para el primer diseño fueron unas ruedas omnidireccionales de 4" de diámetro adquiridas en la página de AndyMark.



*Figura A-16 Rueda omnidireccional de 4".*

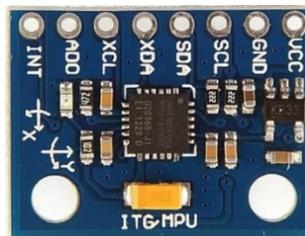
Para el segundo diseño se utilizaron unas ruedas Omnidireccionales de aluminio de 60 mm de diámetro como la mostrada en la figura A-17, adquiridas de la página de SanDoRobotics.



*Figura A-17 Rueda Omnidireccional de aluminio de 60mm.*

## IMU

Para la lectura de la posición angular del robot se utilizó la IMU MPU6050, mostrada en la figura A-18, a una frecuencia de 100 Hz cuyas características se muestran en la tabla A-6.



*Figura A-18 mpu6050.*

*Tabla A-6 Características mpu6050.*

Voltaje de alimentación	3.3-5V
Rangos de medición	$\pm 2g, \pm 4g, \pm 8g, \pm 16g$
Protocolo de comunicación	$I^2C$

## A7. Código de programación

### Código de estabilidad y navegación

A continuación, se muestra una parte del código de estabilidad y navegación desarrollada en Arduino; este fragmento corresponde al código principal del ballbot, en el cual se tienen algunas funciones que realizan operaciones específicas, ya sea calcular el error del algoritmo de control, mover los motores, realizar la lectura de la IMU, entre otras.

```
////////// BIBLIOTECAS //////////
```

```
#include "I2Cdev.h"  
#include "MPU6050_6Axis_MotionApps20.h"  
#include "Wire.h"  
#include <PID_v1.h> // biblioteca PID
```

```
////////// VARIABLES DE LA IMU //////////
```

```
MPU6050 mpu;  
#define INTERRUPT_PIN 2 // Usar pin 2 en Arduino para señal de interrupción  
#define LED_PIN 13
```

```
bool blinkState = false;  
bool dmpReady = false;  
uint8_t mpuintStatus;  
uint8_t devStatus;  
uint16_t packetSize;  
uint16_t fifoCount;  
uint8_t fifoBuffer[64];
```

```
////////// VARIABLES DE MOVIMIENTO Y ORIENTACIÓN //////////
```

```
Quaternion q;           // [w, x, y, z]    Quaternion  
VectorInt16 aa;         // [x, y, z]    Sensor de aceleración  
VectorInt16 aaReal;     // [x, y, z]    Mediciones del sensor de aceleración sin gravedad  
VectorInt16 aaWorld;    // [x, y, z]    Medidas del sensor de aceleración con referencia mundial  
VectorFloat gravity;    // [x, y, z]    Vector de gravedad  
float euler[3];         // [psi, theta, phi] Almacenamiento de los ángulos de Euler  
float ypr[3];           // [yaw, pitch, roll] Almacenamiento de yaw/pitch/roll y vector de gravedad
```

```
////////// Estructura del paquete para InvenSense teapot demo//////////
uint8_t teapotPacket[14] = { '$', 0x02, 0,0, 0,0, 0,0, 0,0, 0x00, 0x00, '\r', '\n' };
```

```
////////// DETECCIÓN DE INTERRUPCIONES //////////
```

```
volatile bool mpulInterrupt = false;
void dmpDataReady()
{
    mpulInterrupt = true;
}
```

```
////////// Variables de la IMU//////////
```

```
float yaw=0, pitch=0, roll=0;
```

```
//////////////////// VARIABLES THETA Y BETA //////////////////////
```

```
float w1[3];
float v1[3];
float u1[3];
float n1[3];
```

```
float ux[3]={1,0,0};
float uy[3]={0,1,0};
float uz[3]={0,0,1};
float theta,beta;
float beta_ant;
```

```
//////////////////// VARIABLES DEL CYTRON //////////////////////
```

```
int dir1=8, dir2=6, dir3=4;
int pwm1=7, pwm2=5, pwm3=3;
boolean direccion1=HIGH, direccion2=HIGH, direccion3=HIGH;
```

```
//////////////////// VARIABLES DE RELACIÓN DE PAR //////////////////////
```

```
float cosi=0, seni=0, seno=0, coseno=0;
float alpha=67;
float T_1, T_2, T_3;
float T1d, T2d, T3d;
float T1m, T2m, T3m;
float Ep1=0, Ep2=0, Ep3=0;
```

```
////////// VARIABLES DEL CODIFICADOR //////////
```

```
int canal_a1= 46;  
int canal_b1= 44;  
int canal_a2= 42;  
int canal_b2= 40;  
int canal_a3= 38;  
int canal_b3= 36;
```

```
////////// TIEMPO ENTRE CADA PULSO//////////
```

```
static volatile float tant1=0, tact1=0, count1=0;  
static volatile float tant2=0, tact2=0, count2=0;  
static volatile float tant3=0, tact3=0, count3=0;  
static volatile boolean A_1, B_1, A_2, B_2, A_3, B_3;  
static volatile byte state1, statep1, indexx1;  
static volatile byte state2, statep2, indexx2;  
static volatile byte state3, statep3, indexx3;  
static volatile int QEM[16]= {0,-1,0,1,1,0,-1,0,0,1,0,-1,-1,0,1,0};  
float vel1=0;  
float pos1=0;  
float vel2=0;  
float pos2=0;  
float vel3=0;  
float pos3=0;  
float w_e1=0, w_e2=0, w_e3=0;
```

```
////////// VARIABLES DEL ALGORITMO DE CONTROL //////////
```

```
double kp=82, kd=0.82, ki=8.2, kt=0; //Variables de la esfera sin canicas con voltaje de 7.4 V a kp de 80  
//Con un voltaje más bajo de 7.2 V, se le pone un kp de 8.2  
double errorj=0,Uj=0;  
double kpj=0.12,kdj=0.0,kij=0.0;///D=1500 kp era 0.12  
float Pti=0,Ptj=0,Pt=0,angulo_pos=0;  
double angulo=0,error=0,errorpasado=0,error_d=0;  
double U;  
  
float tact=0,tant=0,tmuestreo=7;  
float Entrada_1=0, Entrada_2=0, Entrada_3=0;  
float Error_suma=0;  
boolean inicio=true;
```

```
////////// VARIABLES DE LOS SENSORES DE CORRRIENTE //////////
```

```
float sens_1=0.0374,sens_2=0.0378,sens_3=0.034; //sensibilidad en V/Amp para sensor77 de 30A  
float off_1=1.354,off_2=1.392,off_3=1.37;  
float I_1,I_2,I_3;
```

```
////////////////// VARIABLES DE LA DIRECCIÓN //////////////////
```

```
char numero;  
float count_dir=0.005;  
float a=0,b=0,c=0;
```

```
////////////////// FUNCIÓN DE CONFIGURACIÓN //////////////////
```

```
void setup()
```

```
{  
  Wire.begin();//Para la IMU  
  //Wire.setClock(400000); // Reloj a 400kHz I2C. Si se tienen dificultades de compilación  
  Serial.begin(115200);  
  Serial1.begin(115200);  
  //Se ponen los parámetros iniciales para iniciar la IMU  
  //Aquí se encuentran los valores para calibrarla  
  //Revisar pestaña IMU.....  
  Inicia_IMU();  
  
  /*****/  
  //Se declaran los pines y las direcciones para iniciar los motores  
  //Ver pestaña Motores...  
  Inicia_Motores();  
  
  /*****/  
  
  //Se declaran pines e interrupciones para los encoders  
  //Revisar pestaña Encoders...  
  Inicia_Encoders();  
}
```

////////////////////////////////// FUNCIÓN PRINCIPAL //////////////////////////////////////

```
void loop()
{
  //Aquí se miden los ángulos pitch, roll y yaw para el control
  Mide_Angulo();
  //Leer_pot();
  /****** Para evitar que al principio los motores se muevan *****/
  if (millis(>7000)
  inicio=false;
  /******/
  if(!inicio)
  {
    Calcula_Velocidad();
    /******/
    //Esto es para el cálculo de la dirección que se le dará al robot con ayuda de la PC
    //Revisar parte del código en la pestaña dirección...
    Calcula_dir();
    /******/
    //Aquí se calculan los ángulos rheta y beta para realizar el control
    //Para ver el código ir a la pestaña Theta_Beta...
    //Los ángulos están en radianes
    Calcula_Thetabeta();
    /******/
    //Para realizar el control en 3D
    //Las ruedas se deben de mover de cierta forma para lograr mover al robot
    //Para ver el código, abrir pestaña Relación_Par...
    Relacion_Velocidad();
    //En esta parte se calculan los errores PID para mantener al robot en equilibrio
    //Para ver el código, abrir pestaña Control...
    Realiza_Control();
    /******/
    //Para la etapa de potencia, se manda una señal PWM a cada controlador de motor CYTRON
    //Para ver el código, revisar la pestaña Motores...
    Control_Motores();
    /******/
    Serial1.print(a);
    Serial1.print(",");
    Serial1.print(b);
    Serial1.print(",");
    Serial1.println(-U); } }
```

## Código de la interfaz gráfica

A continuación, se muestra el código de la interfaz para las pruebas de navegación, la cual fue desarrollada con Processing.

```
import processing.serial.*;
Serial myPort;
int cx=100;
int cy=200;
int inicio;
PFont f;
PFont f2;
int puerto;
int num_port;
String resp="";

void setup()
{
  f=createFont("Arial",16,true);
  f2=createFont("Arial Black",16,true);
  num_port=Serial.list().length;
  background(0,0,0);
  size(800,700);
  int y=80;
  fill(255,255,255);
  textFont(f2,30);
  text("Selecciona un puerto",50,50);
  textFont(f,15);
  for(int i=0;i<num_port;i++)
  {

    text(i,50,y);
    text("-",70,y);
    text(Serial.list()[i],90,y);
    y=y+20;
  }
}
```

```

void keyPressed()
{
  if(inicio!=1)
  {
    if(key=='0')
    {
      puerto=0;
    }
    if(key=='1')
    {
      puerto=1;
    }
    if(key=='2')
    {
      puerto=2;
    }
    if(key=='3')
    {
      puerto=3;
    }
    if(key=='4')
    {
      puerto=4;
    }
    setup2();
  }
  if(inicio==1)
  {
    direccion();
  }
}

void setup2()
{
  background(0,0,0);
  f=createFont("Arial",16,true);
  f2=createFont("Arial Black",16,true);
  myPort=new Serial(this,Serial.list()[puerto],115200);
  myPort.bufferUntil('\n');
  fill(255,0,0);
  rect(cx,cy,300,100);
}

```

```

fill(0,0,255);
rect(310+cx,cy,300,100);
fill(255,0,0);
rect(cx,110+cy,300,100);
fill(0,0,255);
rect(310+cx,110+cy,300,100);
fill(255,255,255);
textFont(f2,30);
text("DIRECCIONES BALLBOT",6,60);
fill(183,183,183);
textFont(f2,45);
text("Primer Intento",220,785);
fill(255,255,255);
textFont(f,50);
text("Aumenta A",cx+20,cy+83);
text("Aumenta B",cx+20,cy+183);
text("Disminuye A",cx+320,cy+83);
text("Disminuye B",cx+320,cy+183);
inicio=1;
}

```

```

void draw()
{
  if(inicio==1)
  {
    background(0,0,0);
    fill(255,0,0);
    rect(cx,cy,300,100);
    fill(0,0,255);
    rect(310+cx,cy,300,100);
    fill(255,0,0);
    rect(cx,110+cy,300,100);
    fill(0,0,255);
    rect(310+cx,110+cy,300,100);
    fill(255,255,255);
    textFont(f2,30);
    text("DIRECCIONES BALLBOT",6,60);
    fill(183,183,183);
    textFont(f2,45);
    text("Primer Intento",220,785);
    fill(255,255,255);
  }
}

```

```

    textFont(f,50);
    text("Aumenta A",cx+20,cy+83);
    text("Aumenta B",cx+20,cy+183);
    text("Disminuye A",cx+320,cy+83);
    text("Disminuye B",cx+320,cy+183);
    textFont(f2,45);
    text(resp,220,620);
}
}
void direccion()
{
    if (key==CODED)
    {
        if(keyCode==UP)
        {
            myPort.write(48);
        }
        if(keyCode==DOWN)
        {
            myPort.write(49);
        }
        if(keyCode==LEFT)
        {
            myPort.write(50);
        }

        if(keyCode==RIGHT)
        {
            myPort.write(51);
        }
    }
    if(key=='A' || key=='a')
    {
        puerto=1;
    }
    if(key=='B' || key=='b')
    {
        puerto=1;
    }
}
}

```

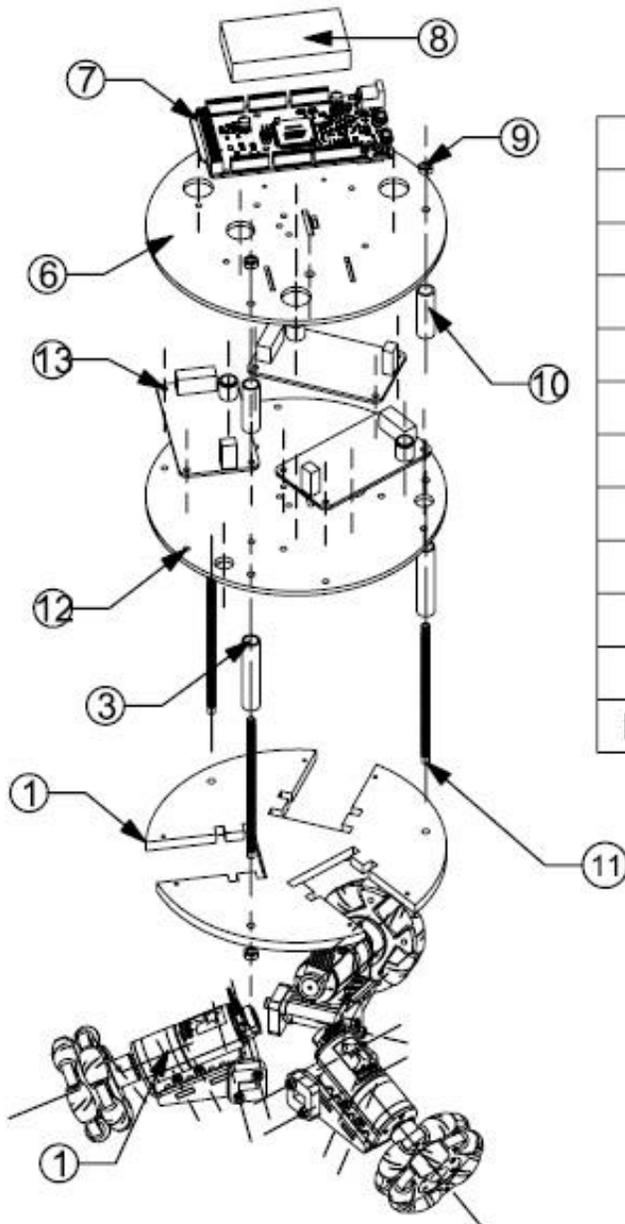
```
void keyReleased()
```

```
{  
  if(inicio==1)  
  {  
    myPort.write(54);  
  }  
}
```

```
void serialEvent(Serial myPort)
```

```
{  
  resp=myPort.readStringUntil('\n');  
}
```

## A8. Planos



11	Espárrago M4	3	PZA-M002
10	Tubo_platos_2	3	MEC-005
9	Tuercas M4	6	PZA-M001
8	Batería LiPo	1	PZA-E003
7	Arduino Due	1	PZA-E002
6	Plato_sup3	1	MEC-004
5	Cytron 13A	3	PZA-E001
4	Plato_sup2	1	MEC-003
3	Tubo_platos	3	MEC-002
2	Plato_bajo	1	MEC-001
1	Módulo de motores	3	ENS-001
No.	Nombre de parte	Cant.	Plano

	Dimensiones	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO		
	Escala	FACULTAD DE INGENIERÍA		
	Revisión	<b>Ballbot v2</b> Proyecto Pieza		Material
Fecha	Hoja	Dibujó	Revisó	Plano
2018-12-06	1 de 1	Humberto Cruz M.	Gerardo Ramos V.	ENS-000

1

2

3

4

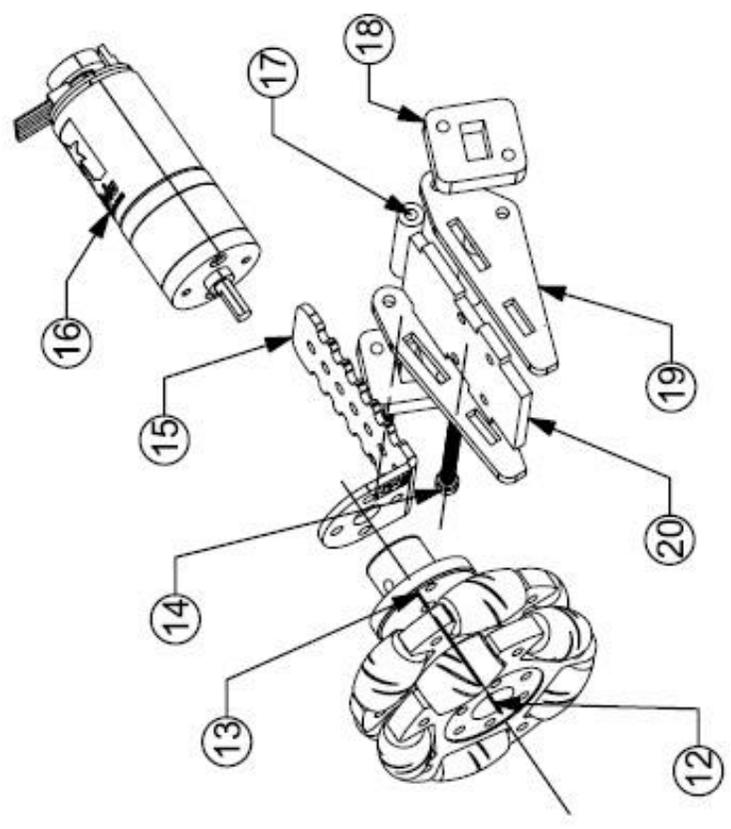
1 2 3 4 5 6

A A

B B

C C

D D



20	Base	3	MEC-009
19	Laterales	6	MEC-008
18	Sujetador	6	MEC-007
17	Tubo	6	MEC-006
16	Pololu 25D 75:1	3	PZA-E004
15	Bracket Pololu	3	PZA-M006
14	Tornillo M3x120 c/cil.	6	PZA-M005
13	Hub Pololu	3	PZA-M004
12	Rueda Omnidireccional 60 mm	3	PZA-M003
No.	Nombre de parte	Cant	Plano

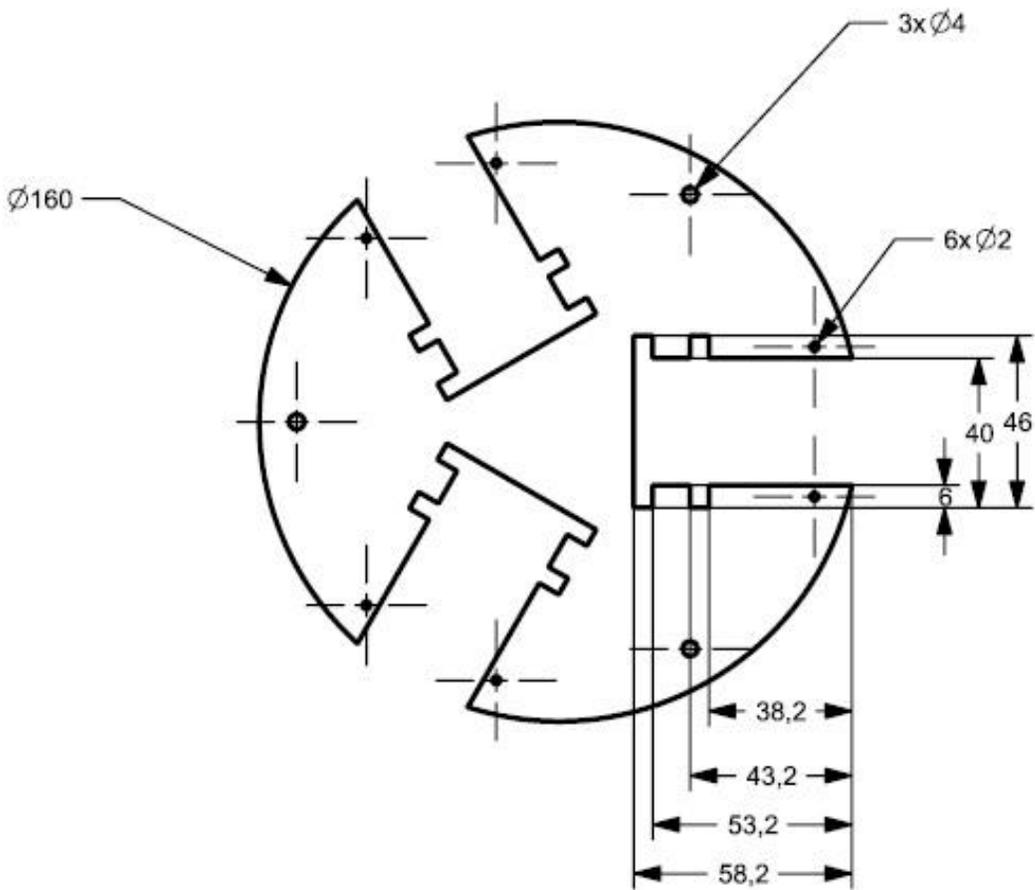


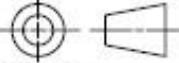
Dimensiones  
Escala  
Revisión  
A  
Hoja  
2018-12-06 1 de 1

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE INGENIERÍA  
**Ballbot v2**  
Modulo de motores  
Humberto Cruz M. Gerardo Ramos V.  
Revisó

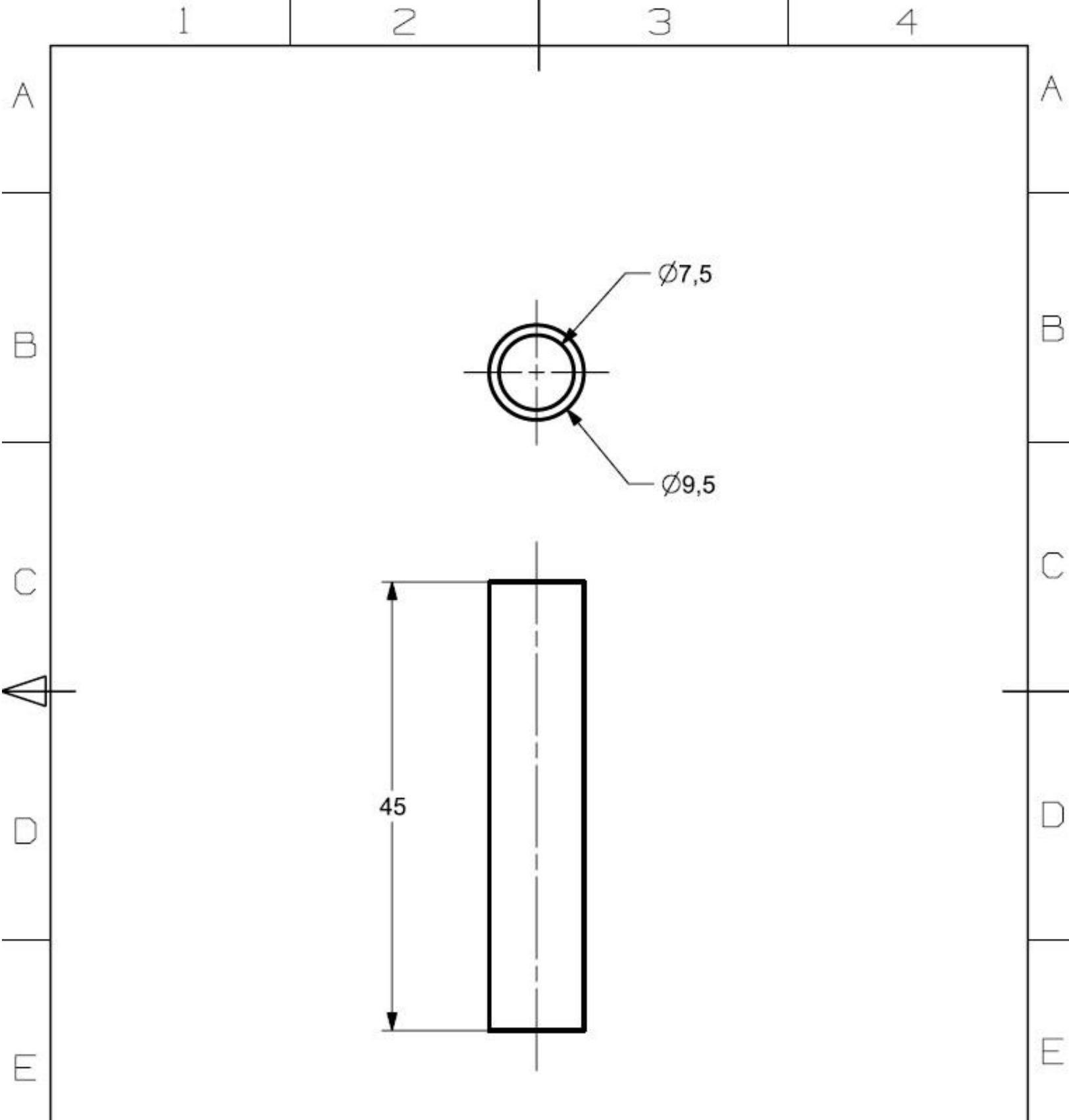
Proceso  
Material  
Plano  
ENS-001

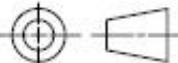
1 2 3 4 5 6



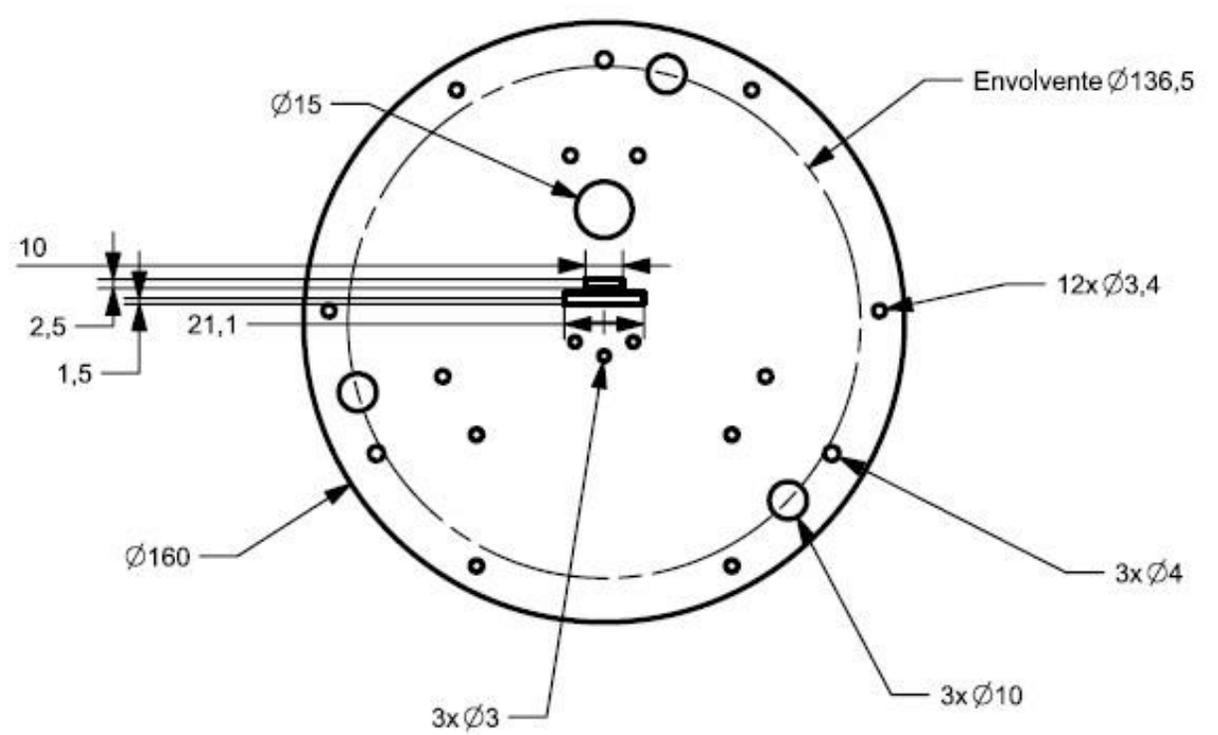
	Dimensiones	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO FACULTAD DE INGENIERÍA <b>Ballbot v2</b>		
	mm			
Fecha	Hoja	Dibujó	Revisó	Material
2018-12-06	1 de 1	Humberto Cruz M.	Gerardo Ramos V.	Acrílico
				Plano
				MEC-001

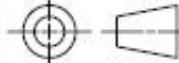


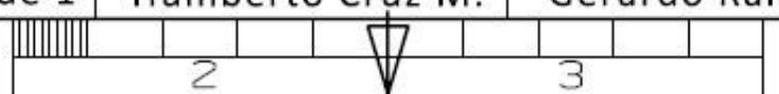


	Dimensiones	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO FACULTAD DE INGENIERÍA		
	mm			
	Escala	Proyecto	Material	
2:1	Pieza	Tubo Plato	Acrílico	
Revisión	A	Dibujó	Revisó	Plano
2018-12-06	Hoja	Humberto Cruz M.	Gerardo Ramos V.	MEC-002
	1 de 1			





	Dimensiones	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO		
	mm	FACULTAD DE INGENIERÍA		
	Escala	<b>Ballbot v2</b>		Proceso
Revisión	1:2	Proyecto		Material
A		Pieza	<b>Plato_sup2</b>	<b>Acrílico</b>
Fecha	Hoja	Dibujó	Revisó	Plano
2018-12-06	1 de 1	Humberto Cruz M.	Gerardo Ramos V.	<b>MEC-003</b>





1

2

3

4

A

A

B

B

C

C

D

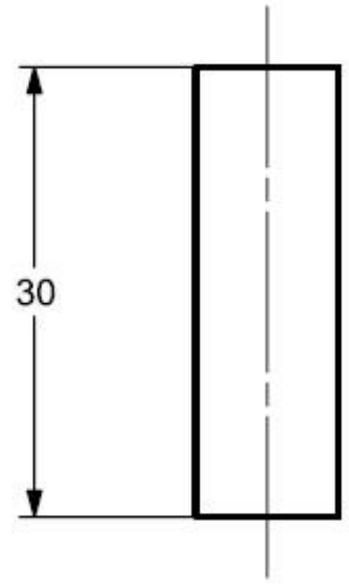
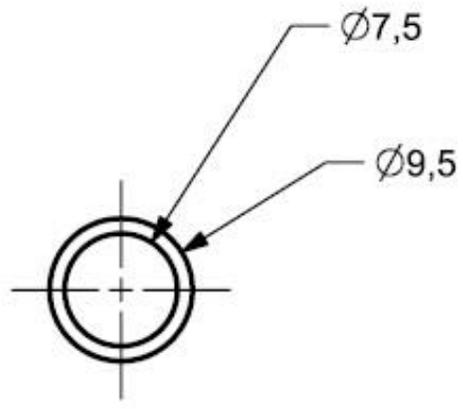
D

E

E

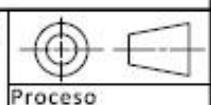
F

F



Dimensiones  
**mm**  
Escala  
**2:1**  
Revisión  
**A**

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE INGENIERÍA



Proyecto  
Pieza  
**Tubo\_platos\_2**

Material  
**Acrílico**

Fecha  
**2018-12-06**

Hoja  
**1 de 1**

Dibujó  
**Humberto Cruz M.**

Revisó  
**Gerardo Ramos V.**

Plano  
**MEC-005**

1

2

3

4

1

2

3

4

A

A

B

B

C

C

D

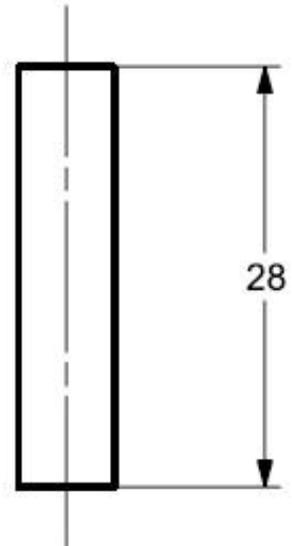
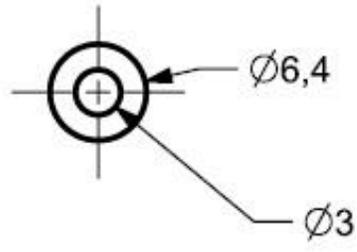
D

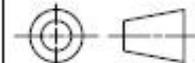
E

E

F

F



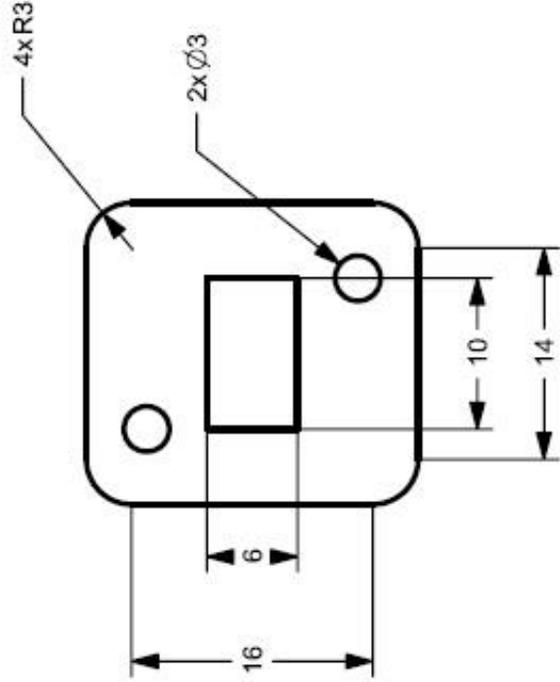
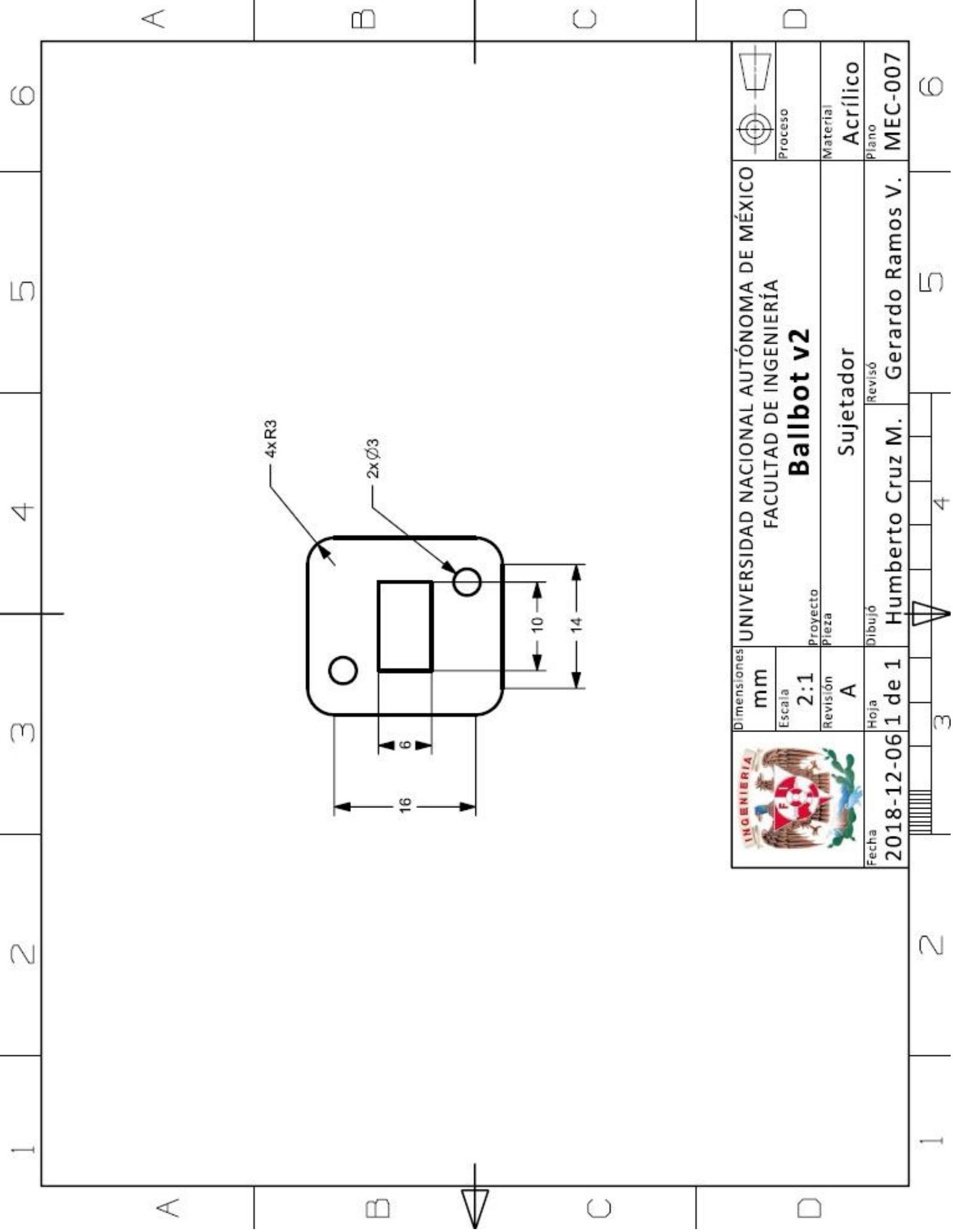
	Dimensiones	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO FACULTAD DE INGENIERÍA		
	mm			
	Escala	2:1	Proyecto	Proceso
Revisión	A	Pieza	Material	
Fecha	2018-12-06	Tubo		Acrílico
Hoja	1 de 1	Dibujó	Revisó	Plano
		Humberto Cruz M.	Gerardo Ramos V.	MEC-006

1

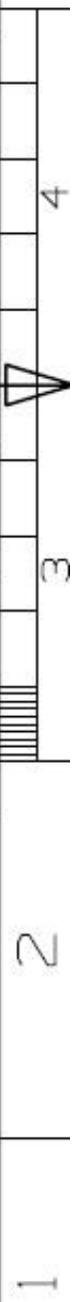
2

3

4



 <b>INGENIERIA</b> Fecha <b>2018-12-06</b>	Dimensiones <b>mm</b>	<b>UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO</b> <b>FACULTAD DE INGENIERÍA</b> <b>Ballbot v2</b>		 Proceso
	Escala <b>2:1</b>	Proyecto Pieza	Material <b>Acrílico</b>	
Revisión <b>A</b>	Dibujo Sujetador		Plano <b>MEC-007</b>	
Hoja <b>1 de 1</b>	Revisó <b>Humberto Cruz M.</b>	Revisó <b>Gerardo Ramos V.</b>		



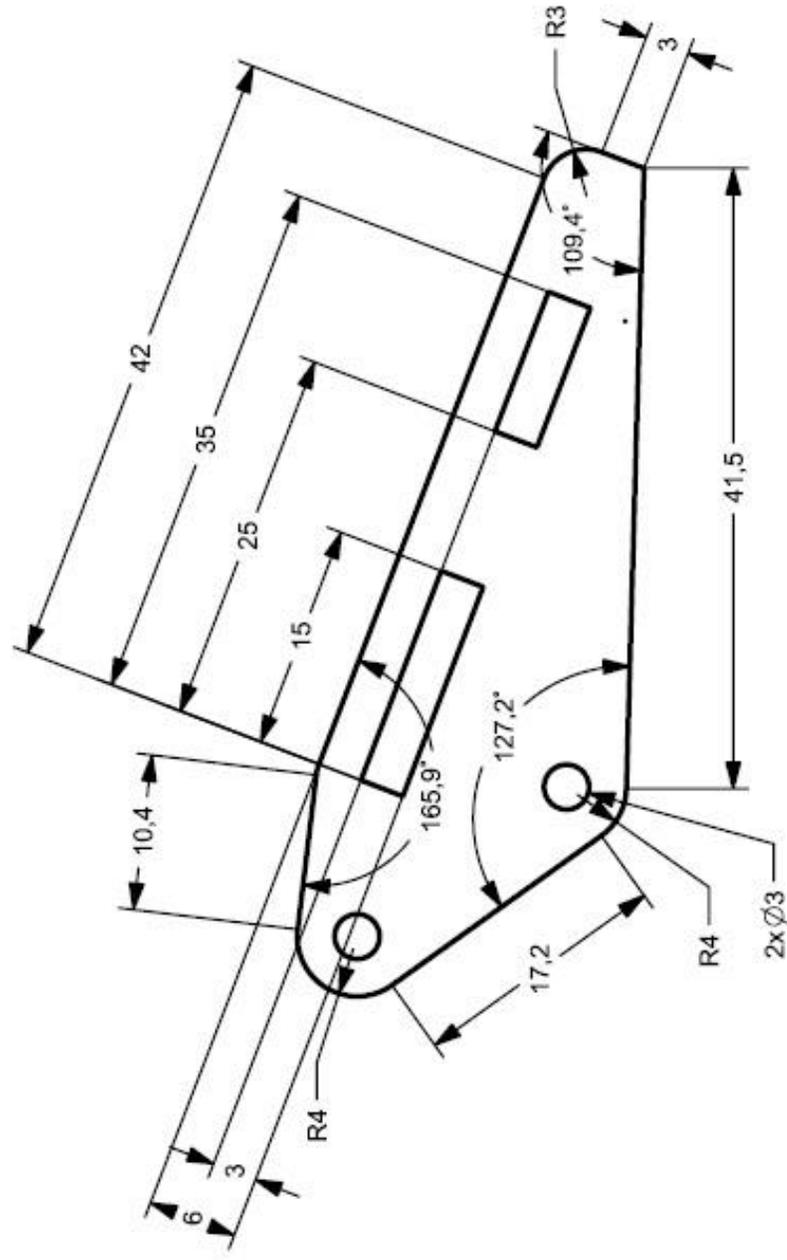
1 2 3 4 5 6

A

B

C

D



Fecha  
2018-12-06

Hoja  
1 de 1

Dimensiones  
mm

Escala  
2:1

Revisión  
A

Proyecto  
Pieza

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE INGENIERÍA

**Ballbot v2**

Laterales

Dibujó  
Humberto Cruz M.

Revisó  
Gerardo Ramos V.

Proceso

Material  
Acrílico

Plano  
MEC-008

1 2 3 4 5 6

A

B

C

D

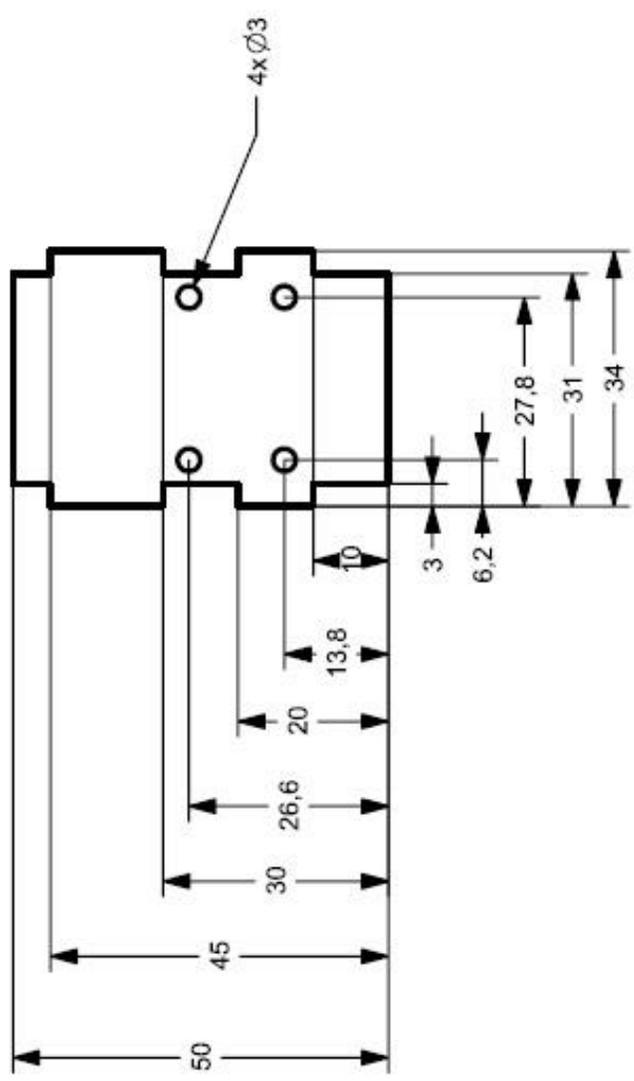
1 2 3 4 5 6

A

B

C

D



Dimensiones: mm  
 Escala: 1:1  
 Revisión: A  
 Fecha: 2018-12-06  
 Hoja: 1 de 1

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
 FACULTAD DE INGENIERÍA  
**Ballbot v2**  
 Proyecto: Base  
 Pieza: Base  
 Dibujo: Humberto Cruz M.  
 Revisó: Gerardo Ramos V.

Proceso:   
 Material: Acrílico  
 Plano: MEC-009

1 2 3 4 5 6

A

B

C

D

---

## REFERENCIAS

- [1] Lawers, T. B., Kantor, G. A., & Hollis, R. L. (2006b, 26 junio). "A dynamically stable single-wheeled mobile robot with inverse mouse-ball drive". Recuperado 18 febrero, 2019, de <https://ieeexplore.ieee.org/document/1642139>.
- [2] Kumagai, M., & Ochiai, T. (2010, 20 junio). "Development of a Robot Balanced on a Ball - First Report, Implementation of the Robot and Basic Control". Recuperado 18 febrero, 2019, de <https://www.fujipress.jp/jrm/rb/robot002200030348/>.
- [3] Fankhauser, P., & Gwerder, C. (2010). "Modelling and control of a Ballbot ". Recuperado 18 febrero, 2019, de <https://www.research-collection.ethz.ch/handle/20.500.11850/154271>.
- [4] Van der Blonk, K. (2014, julio). "Modelling and Control of a Ball-Balancing Robot". Recuperado 18 febrero, 2019, de [https://essay.utwente.nl/65559/1/vanderBlonk\\_MSc\\_EEMCS.pdf](https://essay.utwente.nl/65559/1/vanderBlonk_MSc_EEMCS.pdf).
- [5] Chen, B. (2015, 25 octubre). Ballbot Part 1. Recuperado 20 octubre, 2018, de <https://www.brianc.me/blog/2015/10/25/ballbot-part-1/>.
- [6] Bruton, J. (2015, 10 mayo). Star Wars EP7 BB-8 Droid. Recuperado 20 octubre, 2018, de <http://www.xrobots.co.uk/parts-concept/>.
- [7] Johnson, K. (2016, 2 diciembre). Skoth/Ballbot. Recuperado 20 octubre, 2018, de <https://github.com/Skoth/Ballbot>.
- [8] Hernández, P. T. (2010). "Control de robots móviles: el caso del Ballbot". Recuperado 18 febrero, 2019, de <http://132.248.9.195/ptb2010/noviembre/0664627/Index.html>.

- [9] Garcia, B., & Minami, Y. (2012). "Diseño, construcción y control de estabilidad de un robot que se balancea sobre una esfera". Recuperado 18 febrero, 2019, de <http://132.248.9.195/ptd2012/julio/0681308/Index.html>.
- [10] Gutiérrez, J. H., & Espinosa, J. L. (2012, febrero). "Equilibrio dinámico en tiempo real de un robot tipo ballbot". Recuperado 18 febrero, 2019, de <http://132.248.9.195/ptd2014/febrero/0709459/Index.html>.
- [11] Krick, E. (1979). *Introducción a la ingeniería y al diseño en la ingeniería*. México: Limusa.
- [12] García, B., & Minami, Y. (2012). "*Aplicación de una unidad de medición inercial en tiempo real para el control de un ballbot*". SOMI XXVII Congreso de Instrumentación, congreso llevado a cabo en Culiacán, Sinaloa, México.
- [13] Sánchez, S. A., Cabrera, M., & García, R. (2014, agosto). Determinación de los parámetros de un motor de CD por medición física directa. Recuperado 2 agosto, 2018, de [https://www.academia.edu/9614705/Obtenci%C3%B3n\\_de\\_Par%C3%A1metros\\_de\\_un\\_Motor\\_de\\_CD](https://www.academia.edu/9614705/Obtenci%C3%B3n_de_Par%C3%A1metros_de_un_Motor_de_CD)
- [14] Katsuhiko, O., (2010), "*Ingeniería de control moderna*", Madrid, España: Pearson Educación.