



**UNIVERSIDAD POLITÉCNICA SALESIANA  
SEDE QUITO**

**CARRERA DE INGENIERÍA ELECTRÓNICA**

**ROBOT MÓVIL PARA LAS COMPETENCIAS DE INSECTO CON OBSTÁCULOS,  
SIN OBSTÁCULOS Y TREPADOR CON CORRECCIÓN DE TRAYECTORIA**

Trabajo de titulación previo a la obtención del  
Título de Ingeniero Electrónico

AUTOR: Miguel Ángel Uvillus Chancusig

TUTORA: Carmen Johanna Celi Sánchez

Quito-Ecuador

2023

## CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, Miguel Ángel Uvillus Chancusig documento de identificación N° 1724930589 manifiesto que:

Soy el autor y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 10 de marzo del año 2023

Atentamente,



Miguel Ángel Uvillus Chancusig

1724930589

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE  
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Yo, Miguel Ángel Uvillus Chancusig con documento de identificación N° 1724930589, expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del proyecto técnico: “Robot móvil para las competencias de insecto con obstáculos, sin obstáculos y trepador con corrección de trayectoria”, el cual ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Quito, 10 de marzo del año 2023

Atentamente,



Miguel Ángel Uvillus Chancusig

1724930589

## **CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN**

Yo, Carmen Johanna Celi Sánchez con documento de identificación N° 1717437808, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: ROBOT MÓVIL PARA LAS COMPETENCIAS DE INSECTO CON OBSTÁCULOS, SIN OBSTÁCULOS Y TREPADOR CON CORRECCIÓN DE TRAYECTORIA, realizado por Miguel Ángel Uvillus Chancusig con documento de identificación N° 1724930589, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 10 de marzo del año 2023



Ing. Carmen Johanna Celi Sánchez, MSc

1717437808

## **DEDICATORIA**

Por demostrarme que con mucho esfuerzo y voluntad se puede alcanzar cada una de las metas que me preponga, este trabajo de titulación se lo dedico a mis padres que siempre me brindan su amor, cariño y comprensión, gracias a ustedes puedo hoy en día decir lo logre y seguiré avanzando para que se sientan orgullosos.

Los adoro con todo mi corazón.

A mi hija por brindarme todo su amor incondicional durante el poco tiempo que estoy junto a ella, demostrándole que la dedicación una y mil veces es recompensada, que la perseverancia y la humildad son valores que nos ayudan a crecer como personas, mi niña mi princesa seguiré adelante para darte lo mejor que te mereces, así como mis padres lo hicieron.

Te amo Andrea Noemi

**Miguel**

## **AGRADECIMIENTO**

Primero agradezco a Dios y al gran apoyo que día a día realizaron mis padres para que mi sueño de ser un profesional se convierta en realidad, mi hija que sin hablar y con su mirada me llenaba de alegría, mi hermano que con cada palabra me motivo a seguir adelante sin importar las adversidades que se presentaban, a cada uno de mis familiares que con sus palabras de aliento me daban fuerzas para continuar en el camino, mis amigos que con cada palmadita me decían si puedes y ahora que a lo largo de más de 5 años de carrera con mucha satisfacción y felicidad les digo esta meta está cumplida.

**Miguel**

## INDICE DE CONTENIDO

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN .....	ii
CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA .....	iii
CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN .....	iv
DEDICATORIA .....	v
AGRADECIMIENTO .....	vi
INDICE DE TABLAS .....	vi
RESUMEN .....	vii
ABSTRACT .....	viii
INTRODUCCIÓN .....	ix
CAPÍTULO I .....	1
1 ANTECEDENTES .....	1
1.1 Planteamiento del problema .....	1
1.2 Tema .....	1
1.3 Justificación .....	1
1.4 Objetivos .....	2
1.4.1 Objetivo general .....	2
1.4.2 Objetivos específicos .....	2
1.5 Metodología .....	2
CAPÍTULO II .....	4
2 MARCO TEÓRICO .....	4
2.1 Sistema Electrónico .....	4
2.2 Control PID .....	4
2.3 Matlab .....	4
2.4 Arduino .....	5
2.5 Sensor Ultrasónico .....	5
2.6 Sensor Inductivo .....	6
2.7 Drive Puente H L298N .....	6
2.8 Electroimanes .....	6
2.9 Sistema Mecánico .....	7
2.10 Sensor Encoder .....	7

2.11	Motorreductor .....	7
2.12	MPU 6050.....	8
2.13	Imanes de Neodimio .....	8
2.14	MDF.....	8
2.15	Duralón .....	9
2.16	Reglamento para la categoría “Robot de carrera de insectos con y sin obstáculos” Concurso Ecuatoriano de Robótica (CER) 2019 .....	9
2.16.1	Capítulo 1: Disposiciones generales.....	9
2.16.2	Capítulo 2: De las consideraciones de la pista .....	9
2.16.3	Capítulo 4: De las características de los robots .....	10
2.17	Reglamento para la categoría “Robot Trepador” Concurso Ecuatoriano de Robótica (CER) 2019 .....	11
2.17.1	Capítulo 2: De los requerimientos técnicos del prototipo .....	11
2.17.2	Capítulo 3: De los requerimientos técnicos de las pistas .....	11
2.18	Impreso 3D .....	12
CAPÍTULO III .....		13
3	DISEÑO DEL SISTEMA.....	13
3.1	Diseño Sistema Electrónico (Placa PCB) .....	13
3.2	Diseño del Sistema Mecánico (AutoCAD).....	16
3.2.1	Estructura Central.....	16
3.2.2	Estructura Brazos Categoría Robot Trepador.....	18
3.2.3	Brazos Categoría Robot Insecto con Obstáculos y sin Obstáculos .....	21
3.3	Función para el sistema Electro Mecánico .....	25
3.4	Funcionamiento de Motores .....	28
3.4.1	Micromotor 1 .....	28
3.4.2	Motorreductor Caja Amarilla .....	28
3.4.3	Micromotor 2.....	29
3.4.4	Motorreductor Metálico.....	30
3.4.5	Motorreductor Novamax 6V 400RPM.....	30
3.5	Creación Categoría Insecto con Obstáculos y sin Obstáculos .....	32
3.6	Diagrama de Flujo Robot Insecto con y sin Obstáculos.....	39
3.7	Creación Robot Categoría Trepador .....	40
3.8	Diagrama de Flujo Robot Trepador .....	41
3.9	Características del prototipo .....	42



3.10	Resumen de costos .....	42
	CAPITULO IV .....	44
4	PRUEBAS Y RESULTADOS DE FUNCIONAMIENTO .....	44
4.1	Robot Categoría Insecto con y sin Obstáculos .....	47
	CONCLUSIONES .....	51
	RECOMENDACIONES .....	53
	REFERENCIAS .....	54
5	ANEXOS .....	56

## INDICE DE FIGURAS

Figura 2.1 Dimensiones pista categoría insecto con y sin obstáculos .....	10
Figura 2.2 Dimensiones de la pista categoría trepador.....	12
Figura 3.1 Simulación en Proteus.....	13
Figura 3.2 Visualización 3D.....	15
Figura 3.3 Conexiones de la placa.....	15
Figura 3.4 Techo.....	16
Figura 3.5 Base.....	16
Figura 3.6 Pie de Amigo.....	17
Figura 3.7 Pared estructura principal.....	17
Figura 3.8 Soporte motores .....	18
Figura 3.9 Soporte brazos categoría trepador.....	18
Figura 3.10 Brazos de acrílico.....	19
Figura 3.11 Brazos de acrílico.....	19
Figura 3.12 Separador 1 .....	20
Figura 3.13 Soporte de brazo para electroimán y sensor inductivo.....	20
Figura 3.14 Soporte para electroimán.....	21
Figura 3.15 Soporte brazos categoría insecto .....	21
Figura 3.16 Separador 2 .....	22
Figura 3.17 Pared protección de piñones .....	22
Figura 3.18 Piñones .....	23
Figura 3.19 Enganche de articulaciones y motor .....	23
Figura 3.20 Articulaciones robot insecto.....	24
Figura 3.21 Soporte para sensor ultrasónico .....	24
Figura 3.22 Confirmación de variables .....	25
Figura 3.23 Herramienta ident.....	26
Figura 3.24 Estimación porcentaje y testeo.....	27
Figura 3.25 Función de transferencia .....	27
Figura 3.26 Micromotor 1 con Encoder .....	28
Figura 3.27 Motorreductor caja amarilla.....	29
Figura 3.28 Micromotor 2 sin Encoder .....	30
Figura 3.29 Motorreductor metálico NovaMax.....	31
Figura 3.30 Prototipo Adherido Prueba 1.....	32

Figura 3.31 Parámetros para piñones .....	33
Figura 3.32 Cálculos de piñones.....	34
Figura 3.33 Control PID .....	35
Figura 3.34 Extremidades prueba categoría insecto.....	35
Figura 3.35 Prototipo Robot Categoría Insecto Prueba 2.....	36
Figura 3.36 Consumo de corriente robot insecto.....	37
Figura 3.37 Acople de baterías .....	37
Figura 3.38 Prototipo Robot Categoría Insecto Prueba 3.....	38
Figura 3.39 Prototipo Robot Categoría Insecto con sensores ultrasónicos .....	38
Figura 3.40 Extremidades de diferentes tamaños con motorreductores de 200RPM.....	40
Figura 3.41 Extremidades finales con motorreductores de 100RPM.....	40
Figura 4.1 Robot trepador con sus componentes.....	46
Figura 4.2 Robot trepador.....	47
Figura 4.3 Robot trepador Final .....	47
Figura 4.4 Prototipo Robot Categoría Insecto .....	48
Figura 4.5 Prototipo Robot Categoría Insecto Prueba 4 Mecanismo Theo Jansen .....	49
Figura 4.6 Robot Categoría Insecto .....	49
Figura 5.1 Extremidad fallida categoría insecto.....	57
Figura 5.2 Batería Lipo.....	57
Figura 5.3 Extremidad Robot Trepador en Duralón.....	57
Figura 5.4 Obstáculos Robot Insecto.....	58
Figura 5.5 Peso Categoría Trepador .....	58
Figura 5.6 Peso Categoría Insecto .....	58

## INDICE DE TABLAS

Tabla 3.1 Elementos y Características.....	14
Tabla 3.2 Características Motorreductores .....	31
Tabla 3.3 Presupuesto final del proyecto.....	42
Tabla 4.1 Extremidades implementadas .....	44
Tabla 4.2 Tiempos de competencia robot trepador 200RPM a 15V .....	45
Tabla 4.3 Tiempos de competencia robot trepador 100RPM.....	46
Tabla 4.4 Tiempos de competencia robot insecto motores de 200RPM a 15V.....	48
Tabla 4.5 Tiempos de competencia robot insecto motores de 400RPM .....	50

## RESUMEN

En este trabajo de titulación se presentará el modelado mediante software CAD (AutoCAD 2021) e implementación del Robot Insecto y Trepador para competencia, implementando un control de trayectoria para su adecuada participación, basándose en el reglamento (CER) 2019: Para la categoría “Robot de carrera de insectos con y sin obstáculos”, se utilizará un MPU 6050 el cual será intervenido por un control PID ayudando a una estabilización óptima, sensores ultrasónicos con un control ON/OFF para evitar desviaciones pronunciadas a lo largo de la pista, motorreductores fuertes y rápidos. Para la categoría “Robot Trepador” se manejará con electroimanes que soporten la carga del prototipo, sensores inductivos digitales que ayudan a la activación de los electroimanes, imanes de neodimio lo cual ayudará a un ascenso idóneo, motorreductores rápidos con buen torque.

Dicho prototipo será controlado por elementos ya antes mencionados, cumpliendo con las expectativas para ser adversarios habilidosos, se tomará en consideración el material a ser utilizado a lo largo de las pruebas en este caso MDF, Acrílico, Impreso 3D, Duralón, entre otros. Estas nuevas implementaciones incentivarán a los competidores a realizar ejemplares increíbles e innovadores, al finalizar el prototipo en cada categoría se realizará la respectiva comparación y se analizará el comportamiento con control y sin control del mismo.

## **ABSTRACT**

In this degree work, modeling will be presented using CAD software (AutoCAD 2021) and implementation of the Insect and Climbing Robot for competition, implementing a trajectory control for its adequate participation, based on the regulation (CER) 2019: For the category "Robot Insect race with and without obstacles", an MPU 6050 will be used, which will be intervened by a PID control helping optimal stabilization, ultrasonic sensors with ON/OFF control to avoid pronounced deviations along the track, strong geared motors and fast. For the "Climbing Robot" category, it will be managed with electromagnets that support the load of the prototype, digital inductive sensors that help activate the electromagnets, neodymium magnets which will help an ideal ascent, fast geared motors with good torque.

Said prototype will be controlled by elements already mentioned above, meeting the expectations to be skilled adversaries, the material to be used throughout the tests will be taken into consideration in this case MDF, Acrylic, 3D Printed, Duralón, among others. These new implementations will encourage competitors to make incredible and innovative specimens. At the end of the prototype in each category, the respective comparison will be made and its controlled and uncontrolled behavior will be analyzed.

## INTRODUCCIÓN

Debido a la dificultad y calamidad en más de dos años, tanto la economía y el estudio fue decreciendo, evitando que los estudiantes sobrelleven un amplio conocimiento referente a ser visionarios en el campo de la robótica inclusive solucionar problemáticas a nivel industrial.

El campo de la robótica al ser complejo conlleva métodos electrónicos y mecánicos, al no tener conocimientos de los procesos provoca elaborar prototipos básicos que muchas veces no satisfacen el nivel de educación, por lo que los competidores se cohíben en encontrar una visión más amplia de realizar robots.

El ámbito de la tecnología que impulsa el desarrollo de robots, ha estado presente desde hace décadas, Sin embargo, últimamente, nuevos ámbitos de investigación como la Inteligencia Artificial y la tecnología de sensores se han unido a la robótica para crear robots autónomos avanzados, con un abanico mucho más amplio de posibles aplicaciones. (Keisner, Raffo, & Sacha Wunsch, 2016)

El incentivar a competidores a la creación de prototipos innovadores para las competencias, mejorando su nivel investigativo y creatividad, siendo promotor en un grupo que sea capaz de sobrellevar la ideología, buscando nuevos ideales y una satisfacción tanto propia como colectiva.

# **CAPÍTULO I**

## **1 ANTECEDENTES**

### **1.1 Planteamiento del problema**

Debido a la pandemia en diferentes campos de estudio dedicados al arte de la robótica principalmente en competencia, se ha tenido innumerables inconvenientes para transmitir el conocimiento a futuros estudiantes y competidores de parte del grupo ASU de la Universidad Politécnica Salesiana, por lo que el conllevar un instructivo respectivo para la implementación de diversos prototipos se complica por no tener el conocimiento respectivo.

Dicha problemática igualmente consiste en que los robots tanto insectos como trepadores al ser autónomos no tienen un control adecuado de su trayectoria y tienden a desviarse de su objetivo principal, obteniendo penalizaciones al momento de concursar y hasta la propia eliminación de su participación.

### **1.2 Tema**

ROBOT MÓVIL PARA LAS COMPETENCIAS DE INSECTO CON OBSTÁCULOS, SIN OBSTÁCULOS Y TREPADOR CON CORRECCIÓN DE TRAYECTORIA.

### **1.3 Justificación**

Los competidores de robótica deben construir dos o hasta tres ejemplares distintos para competir en las categorías de robot trepador, insecto con obstáculos e insecto sin obstáculos, por lo que se pierde tiempo en la construcción de cada prototipo y dinero.

Dicho proyecto ayudará a encaminar prototipos innovadores para las competencias, teniendo una aspiración amplia para la investigación y de la misma manera estimulando a jóvenes estudiantes a adquirir importantes habilidades que les ayudará toda su trayectoria e incluso durante su vida diaria, teniendo un enfoque científico para resolver problemas y fomentar su creatividad, todo esto permitirá la recopilación de información necesaria, para futuros proyectos técnicos, contribuyendo en analizar las ventajas y las desventajas de la implementación del prototipo múltiple para las competencias de robots. (Morales Rodríguez, 2018)



La tentativa es diseñar el prototipo que pueda participar en los concursos de robótica ya sea en la categoría de insecto con obstáculos, sin obstáculos y trepador en el cual se eliminará una parte de imanes y se implementará electroimanes, para evitar la sobre explotación al uso de los motores, lo cual provoca daños estimulados por los picos de corriente al momento de su accionar, por lo que también se implementará un circuito o módulo de protección para dicha problemática evitando daños a corto plazo, de la misma manera enfocarse en eliminar la desviación de la trayectoria mediante sensores de posición angular, control PID y variando la velocidad de los motores aumentando su competitividad.

## **1.4 Objetivos**

### **1.4.1 Objetivo general**

Desarrollar un robot móvil de competencia en las categorías de insecto y trepador con un control de trayectoria.

### **1.4.2 Objetivos específicos**

- Investigar el estado del arte sobre robots móviles, para corregir los problemas que tienen en las competencias tanto el autómatas trepador e insecto con obstáculos, sin obstáculos utilizando los reglamentos del concurso ecuatoriano de robótica (CER) 2019.
- Diseñar una estructura para robot móvil que satisfaga la participación en tres categorías de competencia mediante software CAD.
- Desarrollar un sistema realimentado para el control de velocidad y trayectoria con hardware y software especializado.
- Verificar el funcionamiento del robot móvil para su validación mediante pruebas experimentales en tres entornos de competencia.

## **1.5 Metodología**

Tomando en cuenta la actual situación, en este proyecto se utilizará la metodología inductiva ya que se realizará un estudio, recolectando información relevante de artículos investigativos, tesis, papers, entre otros.

Por medio de la metodología descriptiva e inductiva se bosquejará un diseño antes de la implementación, para evidenciar como se desarrollará el prototipo antes de ser construido, tomando en cuenta en tamaño de cada uno de los elementos.

Con la ayuda de la metodología descriptiva y deductiva se implementará tanto el hardware como el software de desarrollo que se verán plasmados en el documento o monografía del trabajo propuesto.

A través de la metodología experimental se analizarán los resultados obtenidos de las pruebas realizadas y se procederá a comparar los prototipos individuales junto al diseñado.

## CAPÍTULO II

### 2 MARCO TEÓRICO

#### 2.1 Sistema Electrónico

La definición de la arquitectura de un mini robot se discute desde los aspectos funcionales, físicos y comportamentales; este último incluye la arquitectura de software, igualmente aspectos claves de la implementación como el consumo de potencia, el sistema de comunicación y la fuente de energía del robot. Se completan también las consideraciones microelectromecánicas. (Vega Castillo, Vílchez Monge, Villegas Lemus, & Alvarado Moya, 2011)

En base al diseño para robots miniaturizados el reducir tanto elementos como diseño interno de placa ayuda a incrementar la visión de construcción sometiendo a diversos bosquejos de conexión en un software ideal, lo que conllevar un manejo adecuado del sistema interno.

#### 2.2 Control PID

Es un control utilizado en la industria debido a su confiabilidad, rapidez ante el cambio de la señal de referencia, bajo costo de implementación y sencillez en el manejo de la calibración, este tipo de controlador es eficaz alrededor de una variable de operación ya que mantiene su punto de operación en valores establecidos según la referencia deseada por el usuario. (Salazar Zambrano, 2020)

Este tipo de control es una de las metodologías más utilizada en regulación automática, porque admite alcanzar resultados efectivos en las diferentes aplicaciones en las que es utilizado debido a su flexibilidad.

#### 2.3 Matlab

En el estudio que exhibe Chávez (2020), refuta que “Es un software que aprueba un alto rendimiento en cálculo numérico y además de poseer varias herramientas de desarrollo que admiten la visualización y simulación de sistemas dinámicos como Simulink” (págs. 23-24). Este programa facilita varias herramientas de desarrollo para diversos proyectos como pruebas de funcionamiento, integración de controles, como también para análisis y rendimientos,

realizando códigos de manera rápida, ejecutando grandes algoritmos y al tener toolbox especializado ayuda a optimizar el tiempo de trabajo obteniendo cada paso documentado para diversas aplicaciones.

## **2.4 Arduino**

En cuanto a Arduino es una herramienta fácil en su utilización además en su utilización tanto para expertos o sin experiencia en programación y en electrónica.

Lo más importante a la hora de entender Arduino es que no es únicamente una placa con un microcontrolador, como se suele creer, Arduino es el resultado de la suma de los siguientes elementos:

- Una placa de hardware libre.
- Un software libre, gratis y multiplataforma.
- Un lenguaje de programación libre.

La placa Arduino Nano no tiene una toma de alimentación DC como otras placas Arduino, sino que tiene un puerto mini-USB. Este puerto se utiliza tanto para la programación como para la monitorización en serie. La característica fascinante de Nano es que elegirá la fuente de energía más fuerte con su diferencia de potencial, y la fuente de energía que selecciona el puente no es válida. (Juan, 2020)

Cabe recalcar que por el tamaño y peso ayuda libremente a su manipulación, siendo un elemento importante al momento de crear prototipos en cualquier campo especialmente en la electrónica creando robots.

## **2.5 Sensor Ultrasónico**

El sensor ultrasónico HC-SR04 que sirve para medir distancias por medio de sus dos transductores: un micrófono y altavoz. Crea pulsos de alta frecuencia (no apreciable por el ser humano) que rebota en los objetos cercanos y es reflejado hacia el sensor, que es captado por un micrófono. (MADE Mecatrónica, Automatización y Diseño Electrónico, s.f.)

Este módulo aparte de ser sensible a la manipulación tiene un error de medición alrededor de 2 a 3 mm, lo que muchas veces es mínima para procesos grandes, para procesos pequeños dicha medición puede llegar afectar, tiene un consumo mínimo de corriente, puede ser energizado

con menos voltaje, este sensor puede medir la distancia como lo hacen los murciélagos y los delfines.

## **2.6 Sensor Inductivo**

Los sensores de proximidad inductivos sirven para detectar la presencia o ausencia de objetos metálicos. Cuando el sensor revela un objeto metálico su salida eléctrica cambia de estado igual a un interruptor. De fácil combinación con sistemas digitales como Arduino, Pics o PLCs. Solo es necesario conectar la salida del sensor a una entrada digital del microcontrolador. La salida de este sensor es de tipo NPN normalmente abierto, por lo que al detectar un objeto la salida será GND y en estado de reposo VCC. (NAYLAMP MECHATRONICS SAC, 2021)

Este elemento a pesar de ser robusto, de fácil adaptabilidad, permite trabajar a menor voltaje de lo establecido en su datasheet, sin disminuir el rango de medición siendo capaz de tomar datos, lo cual junto a un electroimán dan un funcionamiento adecuado para prototipos de robots en la categoría de trepadores.

## **2.7 Drive Puente H L298N**

Es el módulo más utilizado para manejar motores DC de hasta 2A. El chip L298N internamente posee dos puentes H completos que permiten controlar 2 motores DC o un motor paso a paso bipolar/unipolar. (NAYLAMP MECHATRONICS SAC, 2021)

Este driver soporta voltajes altos, teniendo muy en cuenta que se puede reemplazar el capacitor de fábrica para poder aumentar un mayor voltaje y tanto entradas como salidas permanezcan intactas, teniendo una detección de bajo voltaje y protección térmica.

## **2.8 Electroimanes**

Un electroimán es un tipo de imán en el que el campo magnético es causado por el flujo de corriente eléctrica. Si el flujo de corriente eléctrica desaparece igualmente desaparece el campo magnético y el efecto proveniente del mismo. Existen diferentes tipos de electroimanes según la dirección de la corriente y la potencia deseada. (Planas, 2021)

Tomando en cuenta al existir diferentes tipos de electroimanes, se opta por disminuir peso, tamaño, forma, etc. Los electroimanes P20/15 son los más aptos para la elaboración de robots debido a su fuerza de retención y al mínimo consumo de corriente.

## **2.9 Sistema Mecánico**

Existe una metodología de diseño mecatrónico de robots industriales basada en la ingeniería Concurrente CE, los sistemas de Diseño Asistido por Computador CAD y la ingeniería Asistida por Computador CAE. Los cuales describen tres tareas fundamentales: modelamiento matemático de la cinemática y de la dinámica del robot, modelamiento gráfico computadorizado de la configuración, simulación grafica del funcionamiento y de la acción de control e integración de las tecnologías que constituyen el nuevo producto mecatrónico. (Carvajal Rojas, 2005)

El sistema mecánico está constituido por una estructura con segmentos rigurosos que van unidos entre si mediante partes o articulaciones. La estructura mecánica es dirigida gracias a diferentes actuadores ya sean neumáticos, hidráulicos o electrónicos.

## **2.10 Sensor Encoder**

Silva (2016) señala que: “Son dispositivos de medición acoplados a un objeto giratorio, este consta de la etapa del transductor que es la conversión del movimiento del disco en señales de pulso que se pueden codificar de forma digital” (págs. 432-433).

Este disco es un codificador que va incrementando una pista principal en zonas potencialmente extendidas y semejantes, las cuales utilizan para iniciar el conteo de pulsos para la medición de posición angular y detectar las revoluciones completas.

## **2.11 Motorreductor**

Los motorreductores son mecanismos que normalizan la velocidad de motores eléctricos, haciéndoles funcionar a un determinado ritmo. Están formados por una serie de engranajes que conforman la cadena cinemática, que son los que funcionan sobre las piezas giratorias. (CLR COMPAÑÍA LEVANTINA DE REDUCTORES, 2022)

Los motorreductores ayudan a variar la velocidad de rápida a lenta dependiendo el trabajo que deba realizar, así mismo reduce las sacudidas y ruido, siendo muy fiable, versátil y adaptable en varios campos de la robótica, existen ejemplares de diferentes voltajes, torques, modelos, etc.

### **2.12 MPU 6050**

El módulo MPU6050 consta de un sensor de temperatura y dos sensores micromecánicos: un giroscopio para medir las rotaciones y un acelerómetro para medir las aceleraciones en el espacio. Estos sensores suelen constar de una parte mecánica y una parte piezoeléctrica. Cuando el objeto está en movimiento, la parte mecánica se mueve contra la parte piezoeléctrica que envía una señal eléctrica. (AranaCorp, 2020)

Este dispositivo es de mucha utilidad debido a que su uso ayuda a controlar, redirigir, estabilizar diversos elementos como por ejemplo teléfonos inteligentes y en este caso robots que necesiten una trayectoria sin desviaciones, siendo un elemento mecánico y eléctrico.

### **2.13 Imanes de Neodimio**

Los imanes de neodimio son imanes superfuertes hechos con tierras raras que producen un rápido campo magnético. El imán de neodimio (también conocido como NdFeB, NIB o Neomagnet) es el tipo de imán de tierras raras más utilizado. Es un imán permanente hecho de una aleación de neodimio, hierro y boro para formar el Nd<sub>2</sub>Fe<sub>14</sub>B estructura cristalina tetragonal. (BUNTING, 2023)

Cabe recalcar que se oxidan muy fácilmente, son sensibles a altas temperaturas y se deben manipularse cuidadosamente, existen de diferentes tamaños, modelos, etc.

### **2.14 MDF**

MDF significa tablero de fibras de densidad media, está fabricado de fibras de maderas y resinas sintéticas comprimidas, la madera prensada no se le obtiene de la naturaleza, sino de un producto derivado de ella. Además, puede no ser necesaria la utilización de clavos o tornillos, ya que los pegamentos para madera presentan un muy buen rendimiento. (GABYCAR Soluciones para Carpintería y Obra, 2021)

Una de las características fundamentales es la densidad, precio, como también ayuda a evitar la aparición de hongos o moho, teniendo una poca resistencia al agua.

## **2.15 Duralón**

Es un polímero termoplástico cuyas propiedades físicas y mecánicas conciben que sea uno de los materiales poliméricos con mayor uso centralmente en la industria de la manufactura. (Valásquez Salazar, 2018)

Al ser considerado uno de los plásticos más utilizados por su versatilidad, ligereza se lo utiliza en diferentes máquinas y mecanismos teniendo un volumen mucho más considerable que el acero, siendo un material difícil y caro de conseguir.

## **2.16 Reglamento para la categoría “Robot de carrera de insectos con y sin obstáculos” Concurso Ecuatoriano de Robótica (CER) 2019**

### **2.16.1 Capítulo 1: Disposiciones generales**

Artículo 1.6: Este reglamento aplica para las categorías de carrera de insecto con y sin obstáculos considerando los siguientes puntos:

- Carrera de insectos con obstáculos: La pista tendrá obstáculos pequeños, siendo estos materiales electrónicos inferiores a 3cm.
- Carrera de insectos sin obstáculos: La pista no tendrá ningún obstáculo a lo largo o ancho.

### **2.16.2 Capítulo 2: De las consideraciones de la pista**

Artículo 2.1: La pista tendrá divisiones de carriles de madera de 10cm de alto, para evitar que un robot invada otro carril, la pista posee paredes laterales.

Artículo 2.2: La salida estará marcada en la pista.

Artículo 2.3: El ancho de cada carril de la pista es de 25cm. El largo de la pista es 220cm. La Figura 2.1 muestra el diseño de la pista.

Artículo 2.4: Los concursantes o alguna otra persona que asista al evento, se abstendrán de pisar, modificar o maltratar la pista.



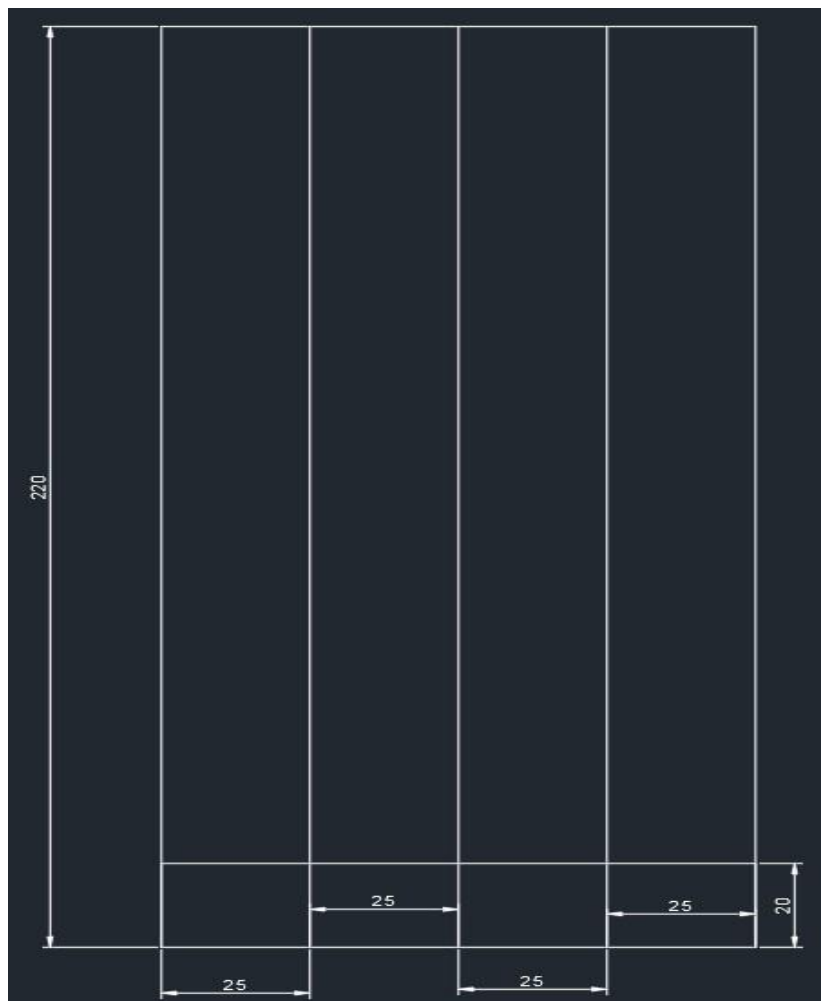
### 2.16.3 Capítulo 4: De las características de los robots

Artículo 4.1: Presentar un robot con forma de insecto autónomo, que sea capaz de avanzar.

- No podrán ser robots insectos que vuelen.
- Deberán presentar patas como articulaciones (no se permitirán ruedas, orugas u otro tipo de avance por desplazamiento o saltos).
- Los robots participantes deberán ser autónomos en cuanto a su sistema de control y de alimentación (no cables conectados a fuentes de alimentación a distancia).

Artículo 4.2: Las dimensiones máximas del robot serán de 20cm de largo, por 20cm de ancho, sin haber restricciones en cuanto a altura. Los robots que no cumplan con estas especificaciones serán descalificados.

Figura 2.1 Dimensiones pista categoría insecto con y sin obstáculos



Diseño pista robot insecto AutoCAD, Miguel Uvillus

## **2.17 Reglamento para la categoría “Robot Trepador” Concurso Ecuatoriano de Robótica (CER) 2019**

### **2.17.1 Capítulo 2: De los requerimientos técnicos del prototipo**

Artículo 2.1: Los prototipos deberán cumplir los requerimientos técnicos comunes que se detallan a continuación:

- El prototipo debe ser autónomo en su totalidad, es decir que por ningún motivo puede ser manejado a control remoto, por computadora o cualquier dispositivo que se encuentre en el exterior.
- Longitud máxima extendida 30cm, la altura y peso del prototipo no existe limitaciones.
- La fuente de energía para alimentar los circuitos electrónicos, así como los motores del prototipo, será cualquier tipo de baterías de corriente continua.
- El accionamiento del prototipo será en forma manual una vez que lo indique el juez, los robots no pueden tener partes en movimiento, antes de la señal de salida.
- Se permite el uso de imanes o electroimanes para la adherencia del robot a la pista.
- Se prohíbe el uso de elementos adhesivos para sujeción o adherencia parcial o total del robot en la pista.
- Se permite únicamente el uso de dispositivos mecánicos para el desplazamiento lineal del robot sobre la superficie metálica de la pista.
- Se prohíbe el uso de ruedas o dispositivos similares que tengan un desplazamiento angular que representen ventaja sobre los mecanismos de desplazamiento lineal.
- La fuente de energía para alimentar los circuitos electrónicos, así como los motores del prototipo, será cualquier tipo de baterías de corriente continua.

Artículo 2.2: Los prototipos deberán constar de hardware y software diseñado por los participantes y/o tarjetas de desarrollo. El tipo de controlador del sistema es libre.

Artículo 2.3: Se prohíbe el uso de prototipos conformados en su totalidad por kits de desarrollo para robots, entiéndase por estos hardware y software.

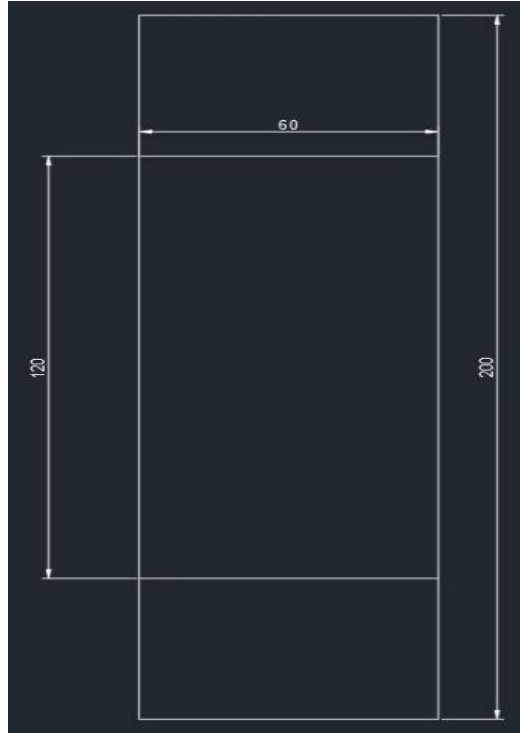
### **2.17.2 Capítulo 3: De los requerimientos técnicos de las pistas**

Artículo 3.1: El escenario para la categoría Robot Trepador tendrá un muro metálico.

Artículo 3.2: Las especificaciones de la pista son las siguientes:

- Tendrá una altura de 1.20m y un ancho de 0.60m.
- La salida y llegada estarán claramente identificadas (Figura 2.2)

Figura 2.2 Dimensiones de la pista categoría trepador



Diseño pista robot trepador AutoCAD, Miguel Uvillus

## 2.18 Impreso 3D

Es un grupo de tecnologías de fabricación por adición capaz de crear un objeto tridimensional mediante la superposición de capas sucesivas de un determinado material. Un proceso por el que se crean objetos físicos a través de la colocación de un material en capas a partir de un modelo digital. (Adeva, 2022)

Se trata de un proceso en el que se crea un objeto físico a través de modelos digitales mediante una impresora 3D, asumiendo el uso de diferentes materiales que pueden ser utilizados, mejorando diferentes procesos y evidenciando una calidad óptima al momento de ponerlo a prueba y comparándolos con diversos materiales.

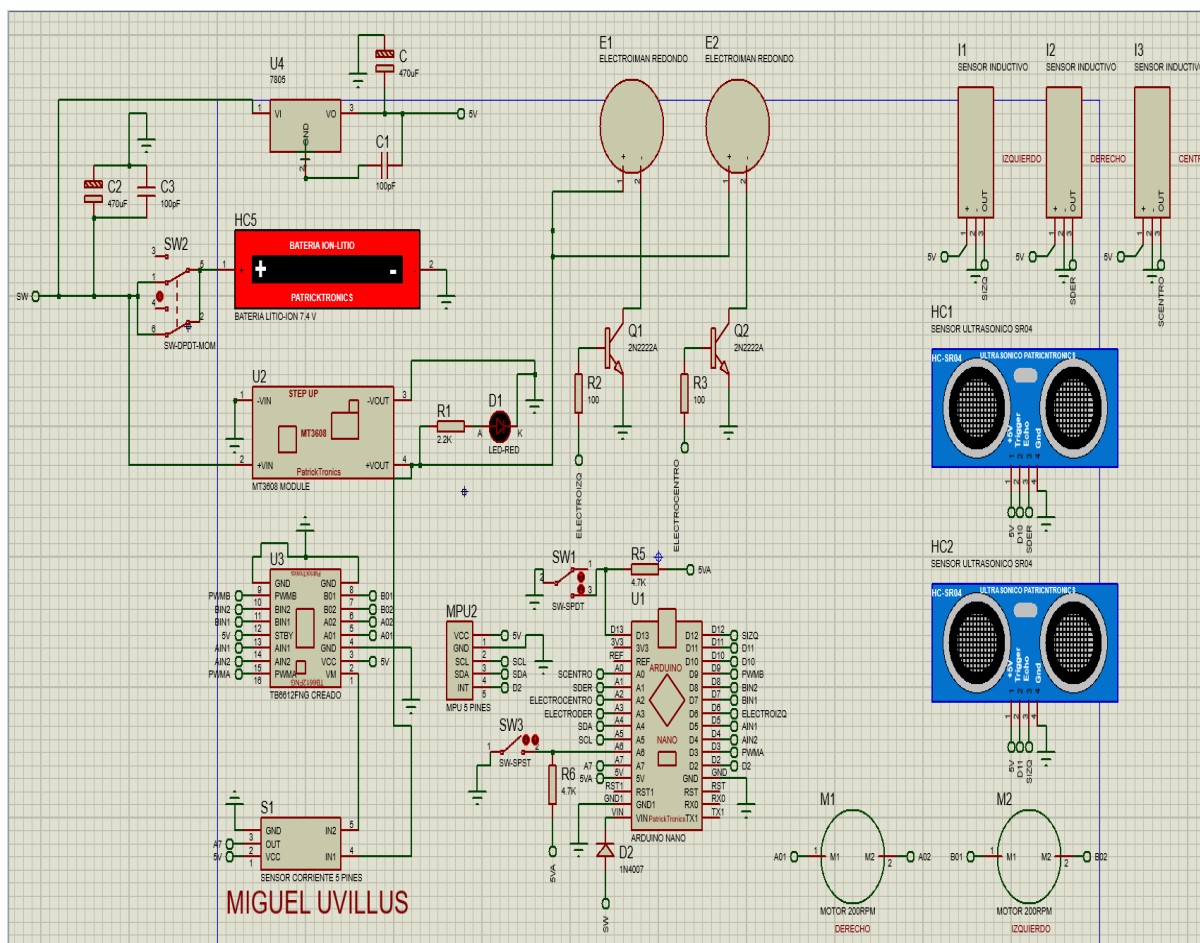
# CAPÍTULO III

## 3 DISEÑO DEL SISTEMA

### 3.1 Diseño Sistema Electrónico (Placa PCB)

En el sistema que se presenta en la Figura 3.1, se aprecia cada uno de los elementos utilizados, lo que conforma el sistema electrónico, tomando en cuenta el uso de elementos que no incluye la librería de Proteus, por lo que se cumple con la creación de elementos propios. Las conexiones simuladas son necesarias para el diseño de la placa PCB, con todo esto se debe tomar en cuenta los componentes a utilizarse, en este caso se detallará en la Tabla 3.1 y componentes complementarios como resistencias, switch, pulsador transistores, capacitores y leds lo cual nos ayudará a comprobar el funcionamiento al ser acoplados en el circuito.

Figura 3.1 Simulación en Proteus



Conexiones de los diferentes dispositivos que conforman la placa para el funcionamiento del robot móvil, Miguel Uvillus

Tabla 3.1 Elementos y Características

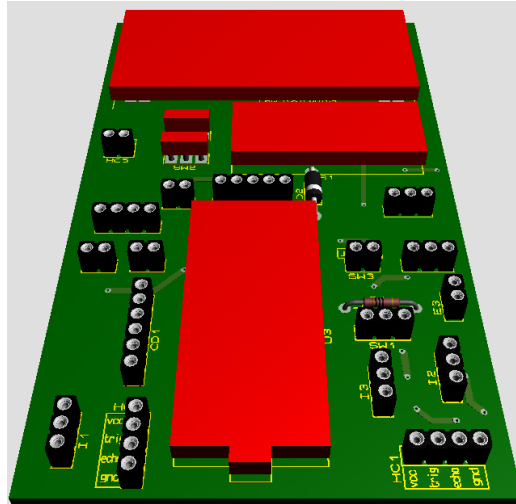
<b>ELEMENTO</b>	<b>CARACTERISTICAS</b>
<b>Batería Lipo recargable (TCB WORTH)</b>	<ul style="list-style-type: none"> <li>• Corriente de salida 900 [mA]</li> <li>• Voltaje 11.1 [V] – 12.6 [V]</li> </ul>
<b>Modulo elevador (Step up)</b>	<ul style="list-style-type: none"> <li>• Integrado Lm2577</li> <li>• Voltaje de entrada 3[V] - 35[V]</li> <li>• Corriente de salida 5 [A]</li> <li>• Eficacia de conversión 92%</li> </ul>
<b>Modulo elevador (Step down)</b>	<ul style="list-style-type: none"> <li>• Convertidor DC-DC Buck: XL4005</li> <li>• Voltaje de entrada: 5.0V a 32V DC</li> <li>• Voltaje de salida: 0.8V a 30V DC</li> <li>• Corriente de salida: 5A máx., 4.5A recomendado (usar disipador para corrientes mayores a 2.5A)</li> <li>• Protección de sobre-temperatura: SI (apaga la salida)</li> <li>• Protección de corto circuito: SI (hasta 8A)</li> <li>• Protección limitadora de corriente: SI</li> </ul>
<b>Sensor de Corriente</b>	<ul style="list-style-type: none"> <li>• Integrado Acs712</li> <li>• Voltaje de operación 4.5[V] – 5.5 [V]</li> <li>• Corriente de salida 30 [A] en AC y DC</li> </ul>
<b>Driver para motores DC Puente H L298N</b>	<ul style="list-style-type: none"> <li>• Chip: L298N</li> <li>• Canales: 2 (soporta 2 motores DC o 1 motor PAP)</li> <li>• Voltaje lógico: 5V</li> <li>• Voltaje de potencia (V motor): 5V - 35V DC</li> <li>• Consumo de corriente (lógico): 0 a 36mA</li> <li>• Capacidad de corriente: 2A (picos de hasta 3A)</li> <li>• Potencia máxima: 25W</li> </ul>
<b>Arduino Nano</b>	<ul style="list-style-type: none"> <li>• Microcontrolador: ATmega328P</li> <li>• Velocidad de reloj: 16 MHz</li> <li>• Voltaje de trabajo: 5V</li> <li>• Pinout: 14 pines digitales (6 PWM) y 8 pines analógicos</li> </ul>
<b>Electroimán</b>	<ul style="list-style-type: none"> <li>• Voltaje de entrada 12 [V]</li> <li>• Corriente 0.25 [A]</li> <li>• Consumo de energía 3 [W]</li> <li>• Fuerza de retención 2.5 [kg]</li> </ul>
<b>Sensor de Proximidad Inductivo</b>	<ul style="list-style-type: none"> <li>• Modelo SN04-N, distancia de detección 4 [mm]</li> <li>• Salida NPN, IP67</li> <li>• Voltaje de alimentación 6 [V] – 36 [V]</li> <li>• Corriente de salida 8 [mA]</li> </ul>
<b>Módulo MPU 6050</b>	<ul style="list-style-type: none"> <li>• Giroscopio Acelerómetro GY-521</li> <li>• Salida digital 6 ejes</li> <li>• Voltaje de alimentación 2.37 [V] – 3.46 [V]</li> <li>• Conexión al Arduino SDA (A4) y SCL (A5)</li> </ul>

Caracterización de los elementos utilizados en el sistema electrónico, Miguel Uvillus

Teniendo en cuenta cada uno de los elementos a utilizarse, se debe cumplir el ajuste y colocación, donde se priorizará la ubicación, lo cual permitirá que cada elemento este

organizado dentro del espacio definido para la placa lo cual nos ayuda a reducir cableado, como se puede observar en la Figura 3.2.

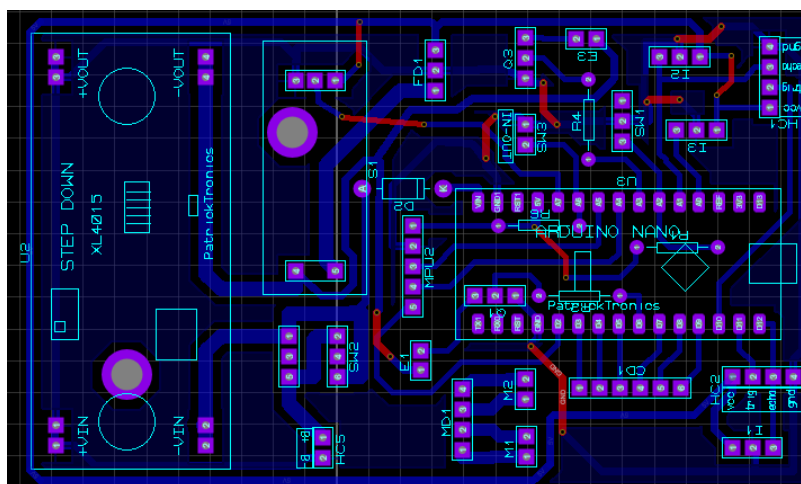
Figura 3.2 Visualización 3D



Placa organizada con los diferentes elementos a utilizarse, Miguel Uvillus

Conjuntamente se debe limitar el ancho de las pistas debido a la separación, por lo que se busca caminos funcionales como se puede observar en la Figura 3.3, con todo esto se busca una proporción de conexión de señal, energía y tierra en la placa, previniendo los cortos circuitos, ahorro de tiempo, producción factible, siendo una de las opciones más confiables para una amplia gama de usos.

Figura 3.3 Conexiones de la placa



Diseño de las pistas de la placa junto con los elementos electrónicos, Miguel Uvillus

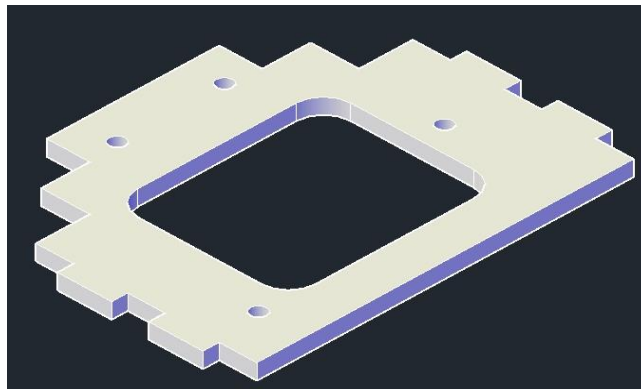
## 3.2 Diseño del Sistema Mecánico (AutoCAD)

Para iniciar con el proceso se debe tomar en cuenta lo liviano del robot, por lo que se opta en manejar los materiales MDF, Acrílico, Impreso 3D, pernos, tuercas y previamente tomar las medidas correspondientes a cada elemento a utilizarse, teniendo una estructura central idónea junto a la placa PCB, un balance ideal, agilidad en la categoría de robot trepador y las limitaciones de medición del robot insecto con obstáculos y sin obstáculos.

### 3.2.1 Estructura Central

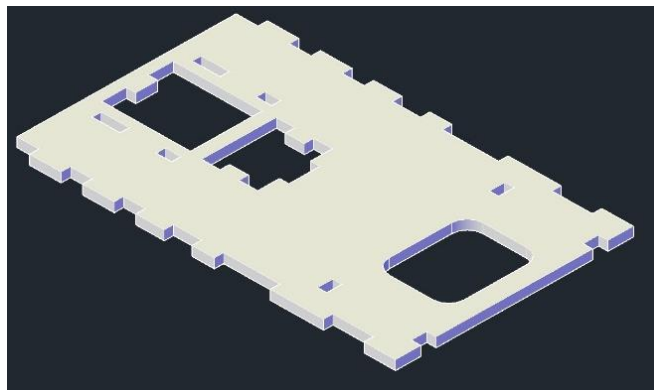
El cuerpo principal debe ser una estructura rígida en este caso la base, techo o asentamiento de la placa echa con MDF, la cual se puede visualizar en la Figura 3.4, Figura 3.5 y llevará elementos como estructura mecánica, actuadores, sensores, elementos terminales y controlador.

Figura 3.4 Techo



Diseño estructura principal para el robot, Miguel Uvillus

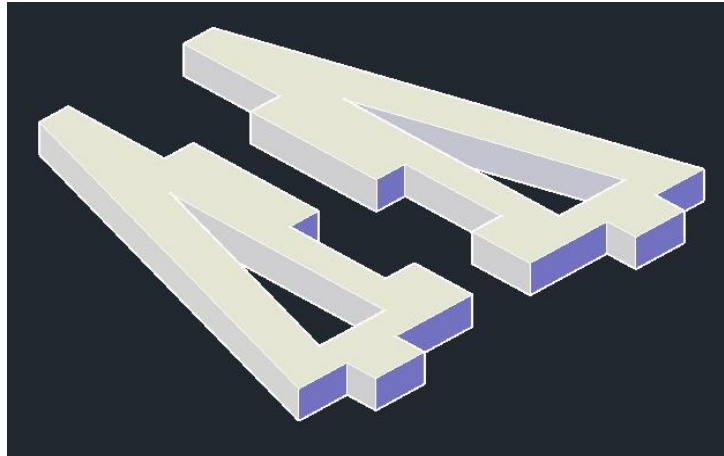
Figura 3.5 Base



Diseño estructura principal para el robot, Miguel Uvillus

Para tener una estructura mucho más concisa y apta, se opta por colocar piezas denominadas pie de amigo como se observa en la Figura 3.6, los cuales se fijan sin problema, evitando rupturas y soporte la actividad del robot de manera brusca en cada competición.

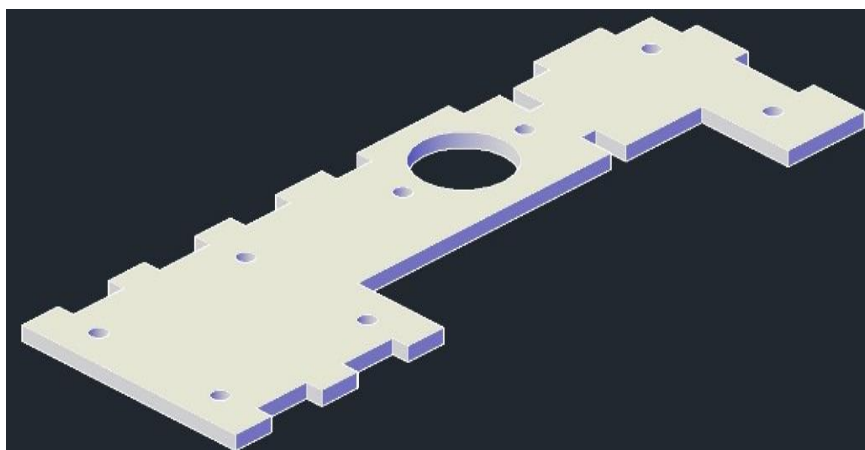
Figura 3.6 Pie de Amigo



Diseño soportes para la dureza del robot con MDF 3mm, Miguel Uvillus

En la Figura 3.7 se puede observar las paredes de la estructura vital, van a ser el anclaje tanto de los brazos del robot para la categoría de trepador como para las articulaciones de insecto de manera individual, junto a los soportes de los motores como se muestra en la Figura 3.8 y tuercas teniendo una estructura concisa y evitando desgastes, movimientos involuntarios entre otros.

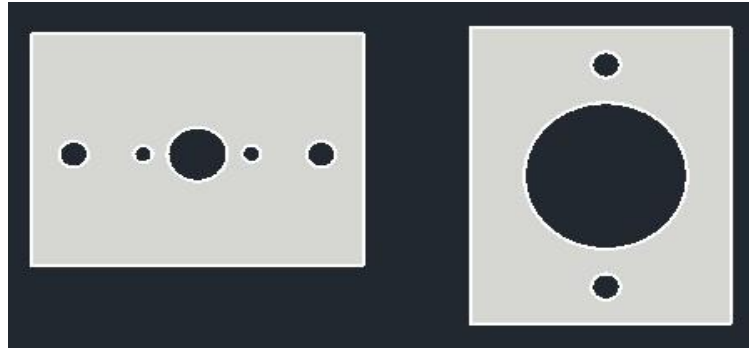
Figura 3.7 Pared estructura principal



Diseño para anclaje de brazos con MDF 3mm, Miguel Uvillus



Figura 3.8 Soporte motores

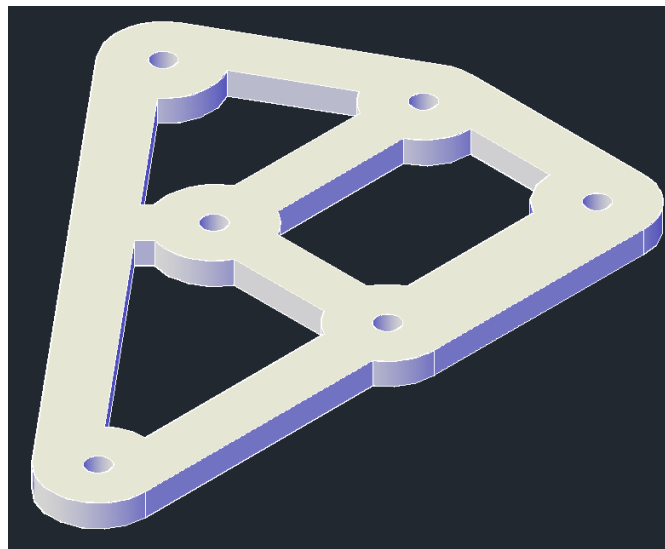


Diseño para impedir el movimiento de pernos al ser atornillados y motores con MDF 3mm y 4mm, Miguel Uvillus

### 3.2.2 Estructura Brazos Categoría Robot Trepador

El montaje respectivo inicia con el soporte de las articulaciones lo cual permitirá anclar los brazos del robot para la categoría de trepador, con orificios para pernos de 3 mm, un cuerpo hueco par alivianar peso como se observa en la Figura 3.9, siendo un importante acople por su forma y resistencia evitando rupturas a pesar del movimiento de los brazos.

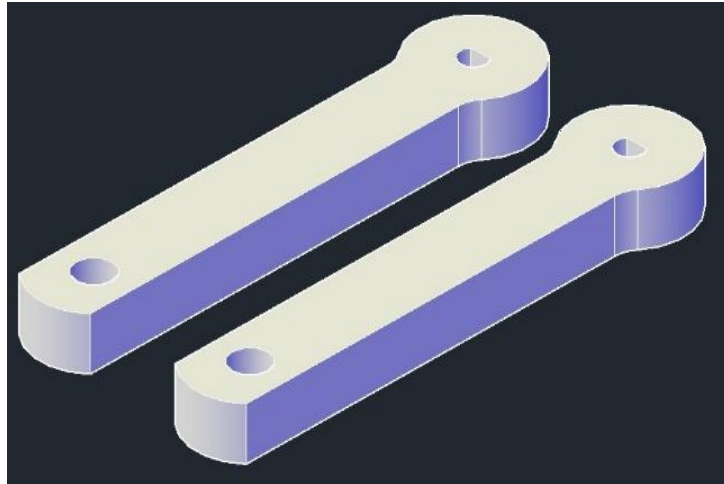
Figura 3.9 Soporte brazos categoría trepador



Diseño soportes para la categoría robot trepador con MDF 3mm, Miguel Uvillus

Los brazos principales enganchados a los motores son elaborados con el material acrílico, con una muesca idéntica al rodamiento del motor para evitar el aislamiento al momento del movimiento como se puede apreciar en la Figura 3.10.

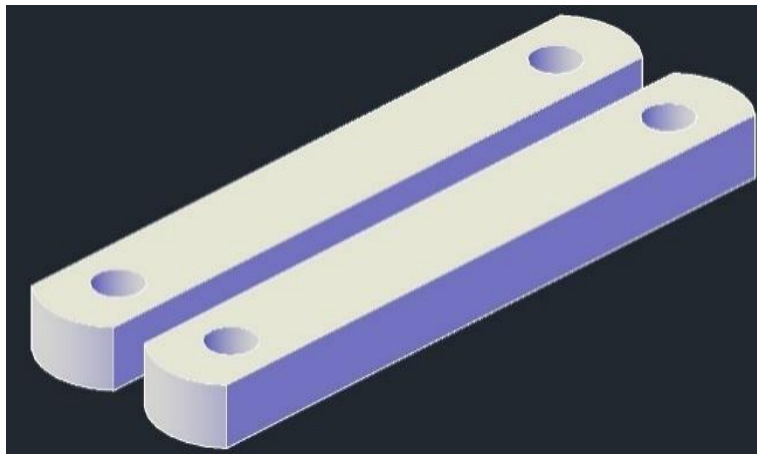
Figura 3.10 Brazos de acrílico



Diseño de brazos para la categoría robot trepador con acrílico 4mm, Miguel Uvillus

Los pares de brazos como se observa en la Figura 3.11 ayudarán a completar la estructura del mismo y siendo de acrílico, ayudan a evitar la fricción al momento de su movimiento, se debe tomar muy en cuenta la distancia de centro a centro para no ocasionar una mala fuerza al motor y provoque trabas.

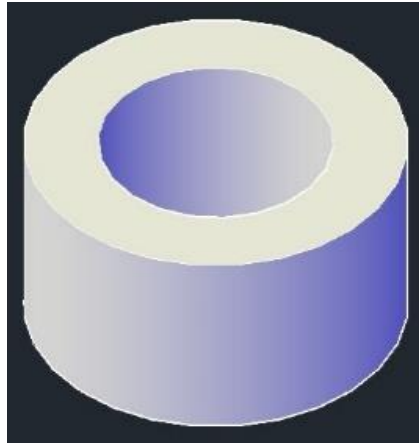
Figura 3.11 Brazos de acrílico



Diseño de brazos para la categoría robot trepador con acrílico 4mm, Miguel Uvillus

La manera de separar las articulaciones y no realizar alguna fricción se lo puede observar en la Figura 3.12, dichos separadores se juntan con arandelas para producir un movimiento libre, asumiendo tener un juego en cada articulación sin afectar el enganche junto a los pernos.

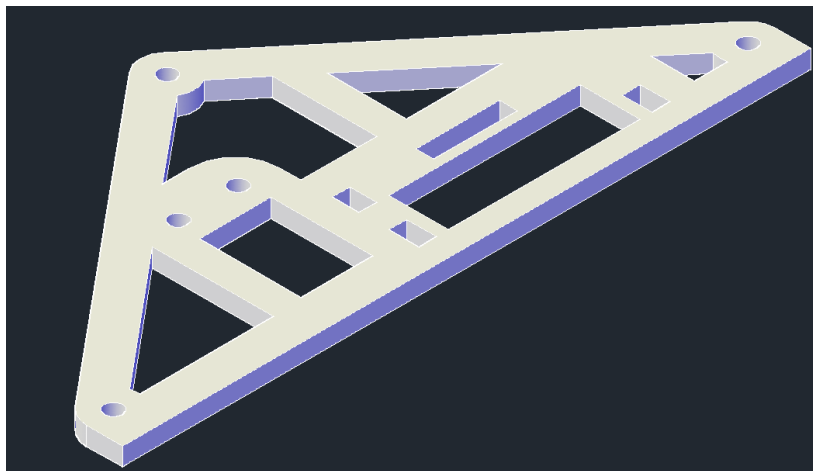
Figura 3.12 Separador 1



Diseño de separadores para evitar fricción con acrílico 4mm, Miguel Uvillus

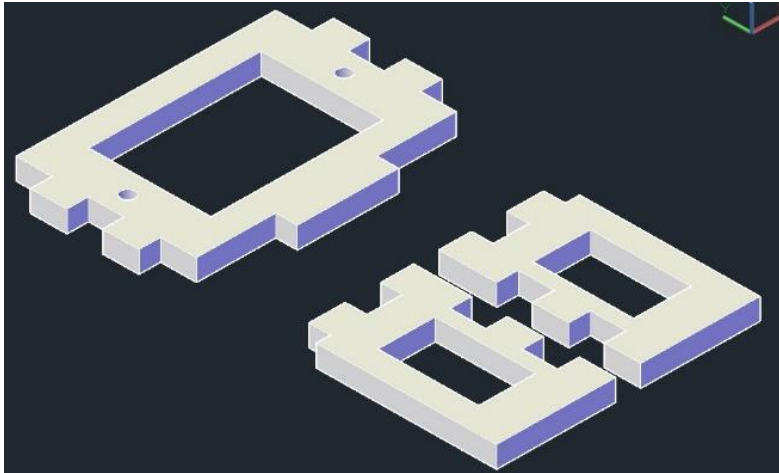
En la Figura 3.13 se observa la estructura donde se ubicará un sensor inductivo con pernos de 3mm y un electroimán acoplado con su soporte Figura 3.14 a cada lado del prototipo, siendo pieza fundamental para el ascenso, teniendo una rigidez apta para el golpe hacia la pared metálica.

Figura 3.13 Soporte de brazo para electroimán y sensor inductivo



Diseño soportes para un sensor inductivo y electroimán con MDF 3mm, Miguel Uvillus

Figura 3.14 Soporte para electroimán



Diseño soporte acoplado para electroimanes con MDF 3mm, Miguel Uvillus

### 3.2.3 Brazos Categoría Robot Insecto con Obstáculos y sin Obstáculos

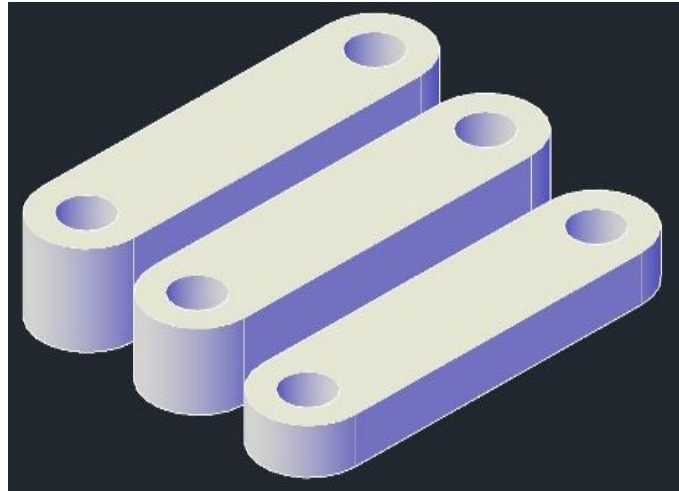
El soporte de las articulaciones como muestra la Figura 3.15 realizado en impreso 3D, accederá al anclaje del motor y las articulaciones del robot para la categoría de insecto con obstáculos y sin obstáculos, con orificios para pernos de 3mm, junto con la Figura 3.16, acoplan las articulaciones evitando el choque, fricción de piezas contra el cuerpo principal.

Figura 3.15 Soporte brazos categoría insecto



Diseño soporte para anclar la patas del robot insecto, impreso 3D 3mm, Miguel Uvillus

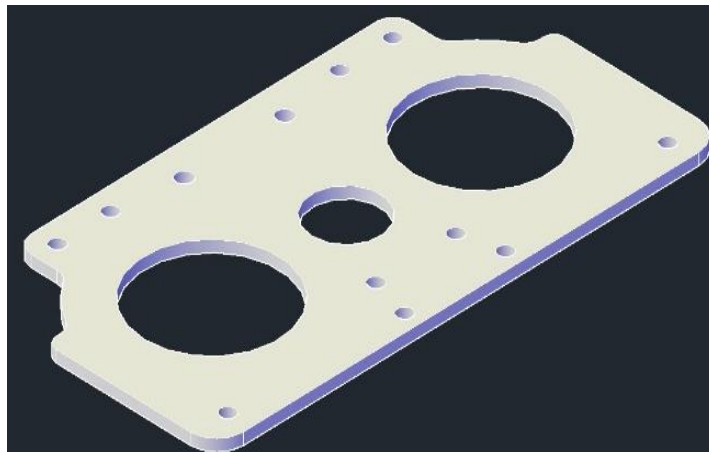
Figura 3.16 Separador 2



Diseño para separar estructura principal y paredes de piñones del robot insecto con y sin obstáculos, realizados en impreso 3D, Miguel Uvillus

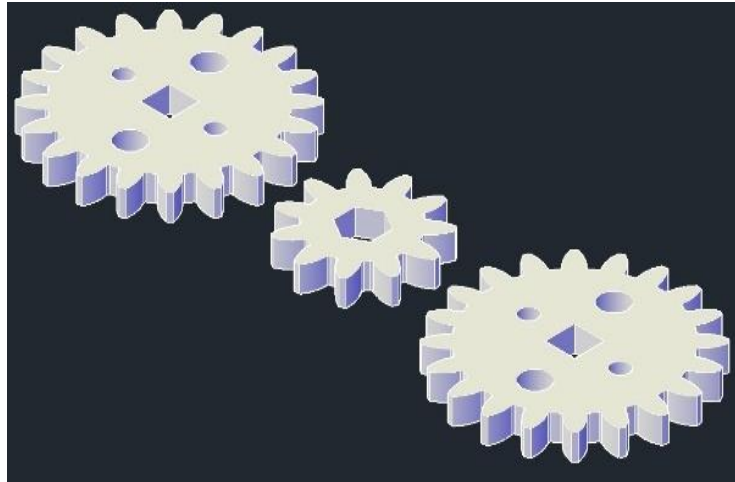
La estructura que muestra la Figura 3.17 ayudará a la protección de los piñones Figura 3.18, lo cual evitará el atasco al momento de su accionar formando un caparazón, donde se podrá envolver al rodamiento con crema para piñones u otros beneficios que eviten un movimiento seco, lento e inapropiado.

Figura 3.17 Pared protección de piñones



Diseño para proteger a los piñones realizado con Acrílico de 2mm, Miguel Uvillus

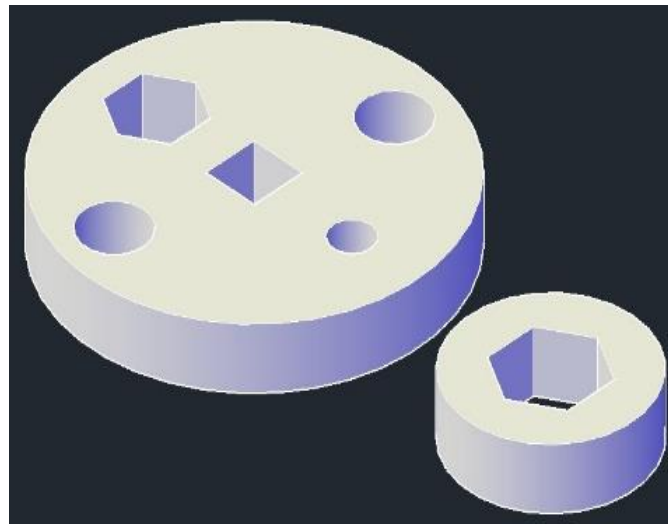
Figura 3.18 Piñones



Diseño de piñones realizados con Acrílico 4mm, Miguel Uvillus

Al ser los piñones unos rodamientos fundamentales se los implementa con piezas enganchadas a las articulaciones principales sosteniendo el accionamiento central de las articulaciones (patas) Figura 3.19, tanto delanteras como traseras, ayudando a obtener un movimiento simultaneo de lado a lado, además el eje central enganchado al motor beneficia un aumento de fuerza.

Figura 3.19 Enganche de articulaciones y motor



Diseño de enganches para evitar fricciones realizados con Acrílico 4mm, Miguel Uvillus

Las articulaciones son una parte fundamental para el movimiento, por lo que se debe realizar 4 piezas de diferentes tamaños Figura 3.20 para cumplir con la forma y caminado del insecto,

cada articulación es unida con pernos de 3mm e impresas en 3D, ayudando a tener una estabilidad y ligereza al momento de ser implementado.

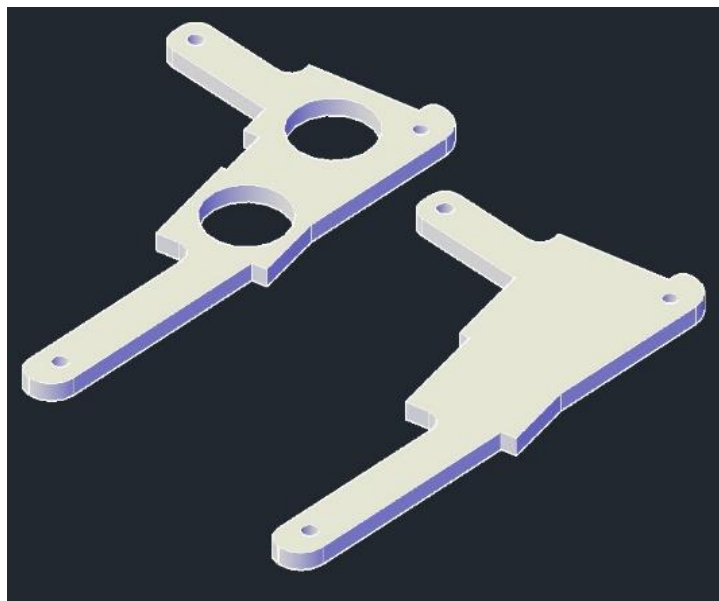
Figura 3.20 Articulaciones robot insecto



Diseño de articulaciones o patas impreso 3D 3mm, Miguel Uvillus

Al utilizar sensores ultrasónicos se necesita protección por su sensibilidad a la manipulación y golpes al momento de su competición como se visualiza en la Figura 3.21, ayudando a evitar la toma de datos falsos.

Figura 3.21 Soporte para sensor ultrasónico



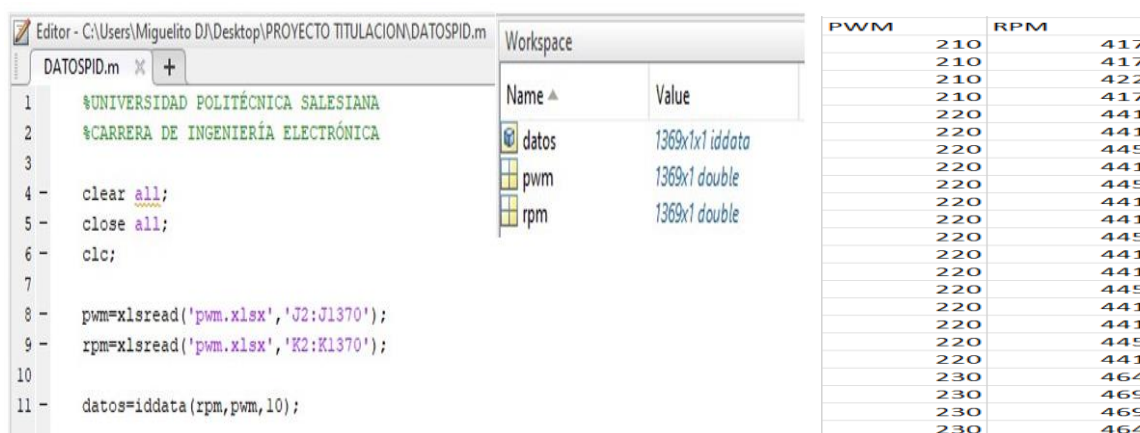
Diseño soporte acoplado a articulaciones impreso 3D 3mm, Miguel Uvillus

### 3.3 Función para el sistema Electro Mecánico

Para la obtención de la función de transferencia se debe tomar en cuenta los siguientes pasos:

- Definir una muestra de datos tomando en cuenta las variables (PWM vs RPM) facilitadas por el motor DC, dichos datos se registran en una hoja de Excel para consecutivamente ser cargados al Workspace de Matlab como se observa en la Figura 3.22.

Figura 3.22 Confirmación de variables

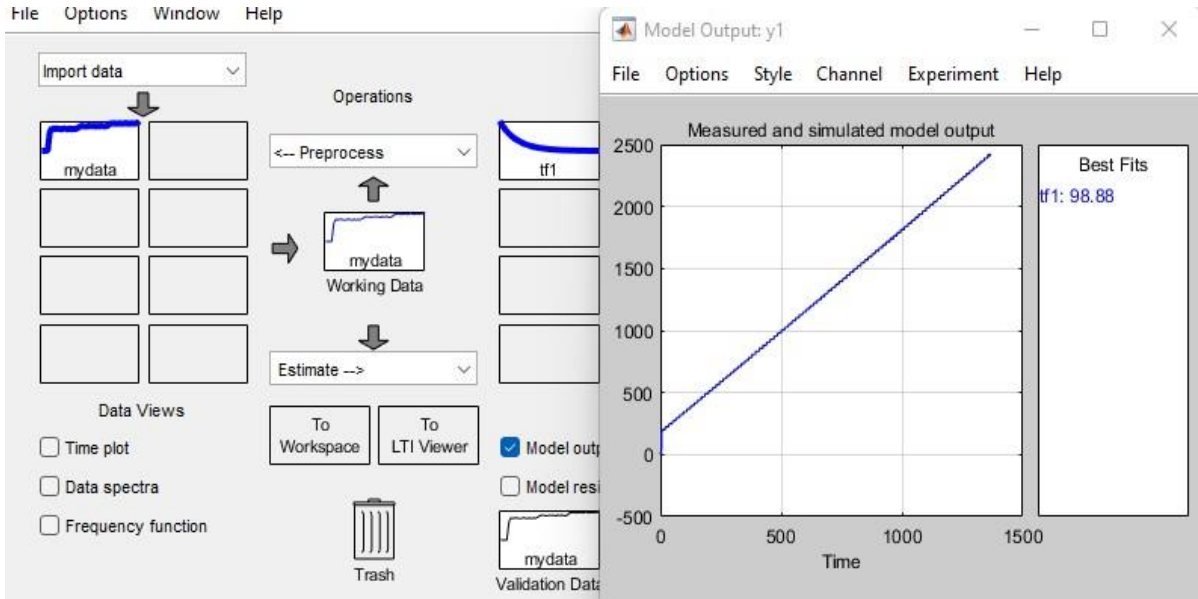


Se define cada una de las variables a utilizar para la obtención de la función de transferencia, Miguel Uvillus

- Utilizar la herramienta ident de Matlab Figura 3.23, subir el muestreo de la variable de entrada y salida, donde el teorema de muestreo de Nyquist nos plantea que una señal analógica que ha sido muestreada se puede rehacer perfectamente a partir de las muestras, tomando en cuenta la frecuencia de muestreo, para lo cual se precisa un tiempo de 1seg y mediante la opción de Transfer Function Models precisamos dos polos y un cero.



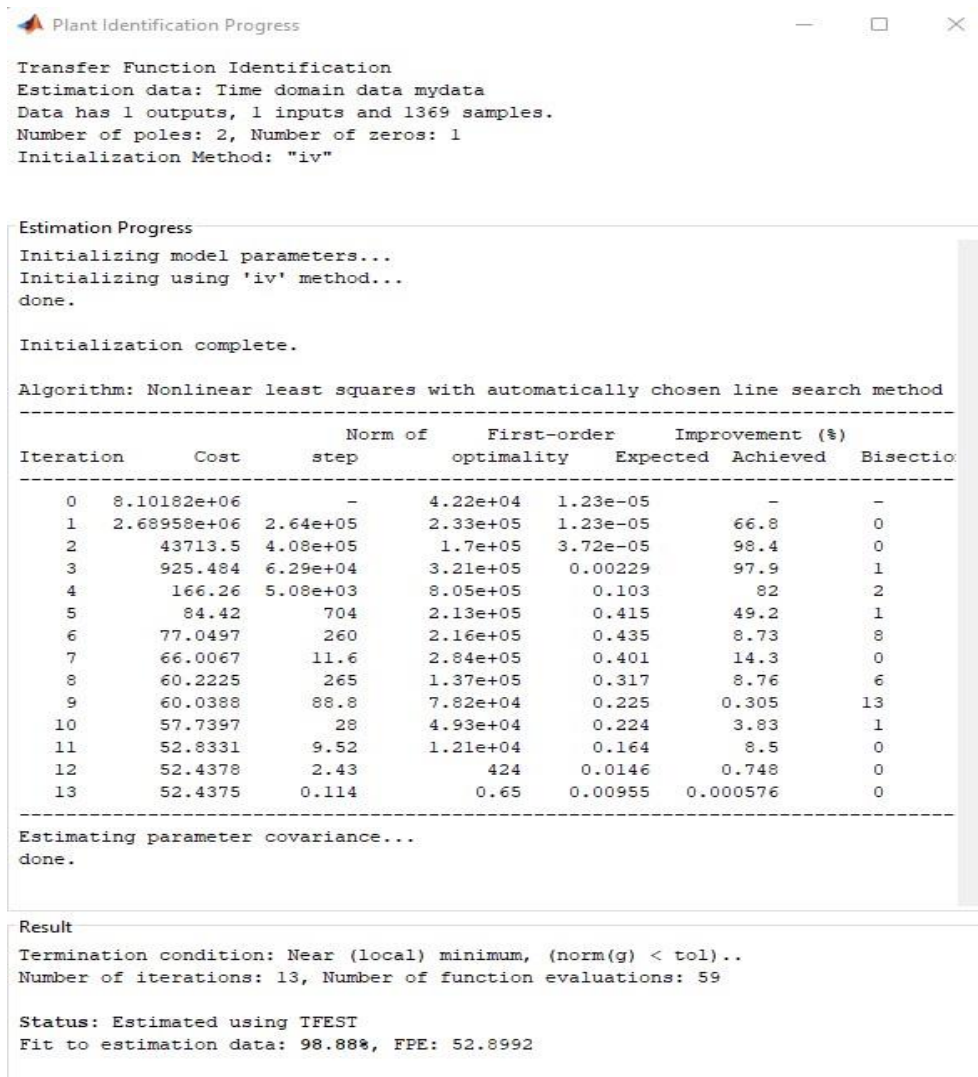
Figura 3.23 Herramienta ident



Herramienta sirve para encontrar la función de transferencia del sistema dinámico, Miguel Uvillus

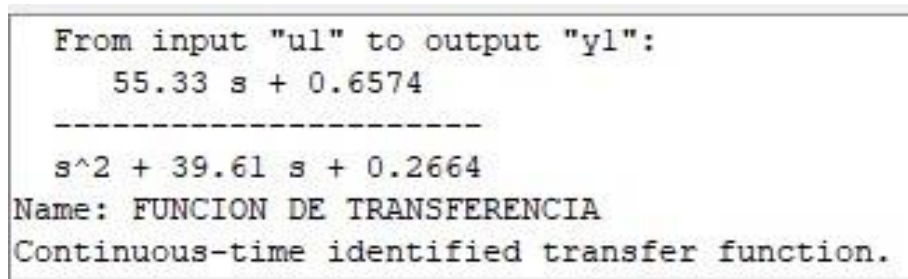
- Al finalizar el testeo, si la estimación es menor al 75% la función de transferencia no es adecuada para un buen control, debido a que se debe aumentar más polos o más ceros a la función o en si los datos están mal interpretados, al contrario, si se tiene una regularidad mayor al 75% dicho porcentaje indica el beneficio de la función de transferencia para poder implementarla en el sistema como se muestra en la Figura 3.24 y Figura 25.

Figura 3.24 Estimación porcentaje y testeo



Finalización para la valoración del muestreo tomado para la función de transferencia, Miguel Uvillus

Figura 3.25 Función de transferencia



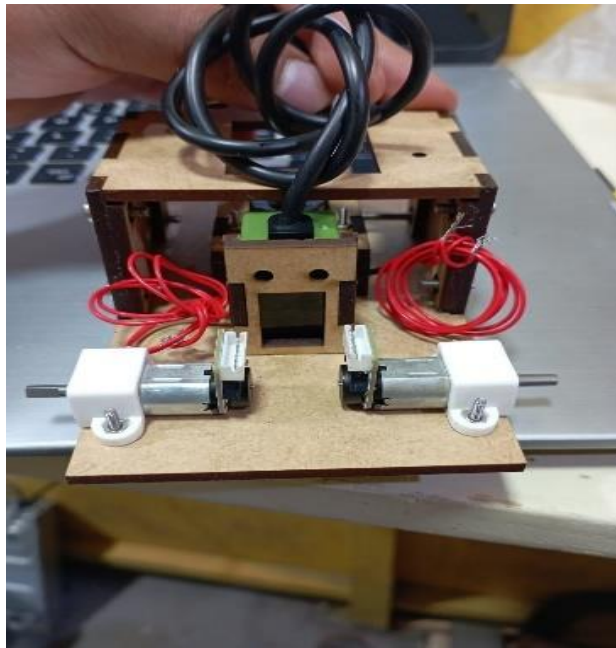
Función de transferencia encontrada con base a los datos obtenidos en Excel, Miguel Uvillus

### 3.4 Funcionamiento de Motores

#### 3.4.1 Micromotor 1

Las pruebas iniciaron evaluando el peso inicial de todos los elementos a utilizarse junto al cuerpo principal, iniciando con un micro motor N20 a 6V con encoder Arduino de 500RPM Figura 3.26, especulando en eliminar peso, por ende, las pruebas realizadas en el suelo mostraban un funcionamiento idóneo, tomando en cuenta que los mismos motores funcionaban para la categoría de insecto y trepador, al momento de ponerlos a prueba en la rampa para la categoría trepador, no obtuvieron suficiente torque para elevar los brazos y avanzar, produciendo trabas y daños internos a los motores, dichos motorreductores ayudaron a exportar datos como PWM y RPM para realizar el control respectivo al momento de implementar un PID adecuado.

Figura 3.26 Micromotor 1 con Encoder



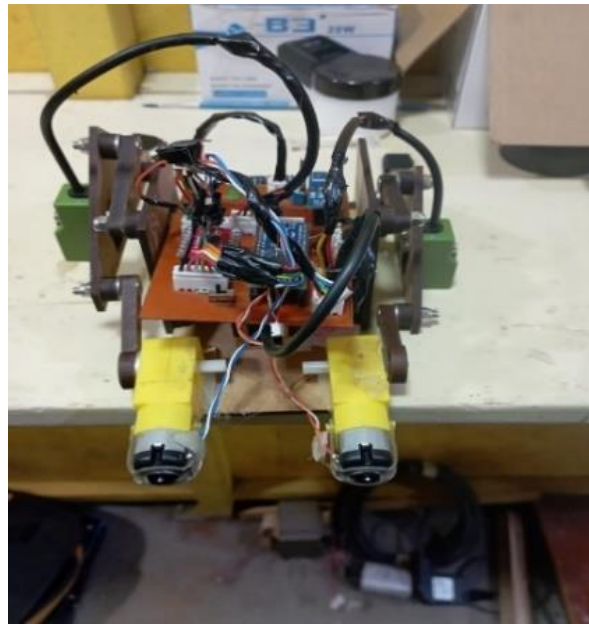
Fotografía micromotores 500RPM estructurados al sistema mecánico principal prueba 1, Miguel Uvillus

#### 3.4.2 Motorreductor Caja Amarilla

Al ser utilizados en diversos campos de competencia, distintas categorías y teniendo un precio reducido, atraían una amplia visión de soportar el peso del prototipo principalmente en la categoría de trepador Figura 3.27, por lo que al utilizar dos motores se sobrellevo el trabajo de

forma adecuada, pero el inconveniente más recalcado fue al ir elevando el voltaje en el módulo step up y PWM en la programación, los piñones al ser de plástico trastrabillaron y los engranajes se fueron destrozando, tomando en consideración que dicha falla podría ocasionar problemas a futuro y no se podría realizar un buen control de velocidad.

Figura 3.27 Motorreductor caja amarilla



Fotografía motorreductores caja amarilla estructurados al sistema mecánico principal prueba 2, Miguel Uvillus

### 3.4.3 Micromotor 2

El micromotor N20 a 6V sin encoder Arduino de 100RPM Figura 3.28, cumplió con las expectativas de sobrellevar el peso, en el suelo como insecto y en la rampa como trepador, asumiendo el único inconveniente de reducir de forma brusca la velocidad al momento de su funcionamiento en las diferentes categorías por motivos que el voltaje máximo que soportaba era de 12V, lo cual al momento de superar dicho valor funcionaba con normalidad, pero al accionar se dañaban los engranajes.

Figura 3.28 Micromotor 2 sin Encoder



Fotografía micro motores 100RPM estructurados al sistema mecánico principal prueba 3, Miguel Uvillus

#### **3.4.4 Motorreductor Metálico**

Al ser un motorreductor de 100 y 200RPM a 12V, piñonera metálica réplica del jsumo titan DC motor como se observa en la Figura 4.4, compensaba el uso en las categorías de competencias establecidas, por el aumento de voltaje en el módulo step up soportando 18V DC sin obtener picos de corriente y trabajando con un PWM máximo de 255 en la programación, teniendo un torque resistente, lo complicado de ocupar dicho motorreductor es que al no tener un datasheet se dificultaba la compra al momento de un daño.

#### **3.4.5 Motorreductor Novamax 6V 400RPM**

Específicamente para competencias de robots sumo y mini sumo, no comúnmente utilizados por su precio, teniendo una velocidad y torque adecuados, funcionaron adecuadamente en la categoría de insecto con y sin obstáculos, ampliando un campo de competencia apropiado, creando un pico de corriente hasta 1A, trabajando a menos de 12V y ampliando el rendimiento en su velocidad como se observa en la Figura 3.29.

Figura 3.29 Motorreductor metálico NovaMax



Fotografía motorreductor 400RPM estructurados al sistema mecánico principal, Miguel Uvillus

El prototipo para las competencias de Insecto con obstáculos y sin obstáculos necesitan mayor velocidad y menor torque, en cambio en la categoría trepador se necesita mayor torque y menor velocidad para una mejor estabilidad, tomando en cuenta la reducción del peso, tamaño y en la Tabla 3.2 se observa las características de los motorreductores utilizados.

Tabla 3.2 Características Motorreductores

TIPO DE MOTOR	CARACTERISTICAS
<b>Micromotor Motorreductor 6V 500RPM 60:1 Pololu</b>	<ul style="list-style-type: none"> <li>• Torque: 0.2kg*cm</li> <li>• Longitud: 68 mm</li> <li>• Diámetro: 25 mm</li> <li>• Longitud del eje: 9.5 mm</li> <li>• Diámetro del eje: 4 mm</li> <li>• Voltaje: DC 6V</li> </ul>
<b>Motor Reductor Dc Doble Eje Amarillo, Carro Arduino</b>	<ul style="list-style-type: none"> <li>• Voltaje de Operación: 3V – 6V</li> <li>• Velocidad Angular nominal: 200 RPM</li> <li>• Reducción: 48:1</li> <li>• Consumo máximo de corriente: 150mA</li> <li>• Diámetro Exterior de Llanta: 65 mm</li> <li>• Diámetro eje de llanta: 6mm</li> <li>• Peso: 50g</li> </ul>
<b>MICROMOTOR DC N20 6V/100RPM</b>	<ul style="list-style-type: none"> <li>• Voltaje de Operación nominal: 6V DC</li> <li>• Consumo de corriente sin carga: 40mA</li> <li>• Consumo de corriente nominal: 100mA</li> <li>• Consumo de corriente eje detenido: 800mA (Pico)</li> <li>• Velocidad Angular nominal: 100 RPM</li> <li>• Reducción: 300:1</li> <li>• Eje de salida: 3mm diámetro y 10mm largo</li> <li>• Dimensiones: 24 x 10 x 12 mm</li> <li>• Peso: 15g</li> </ul>
<b>Motorreductor 100RPM y 200RPM a 12V</b>	<ul style="list-style-type: none"> <li>• Replica JSUMO TITAN DC MOTOR</li> <li>• Voltaje: DC 12V</li> </ul>
	<ul style="list-style-type: none"> <li>• Tensión de funcionamiento: 6 a 16 voltios</li> </ul>

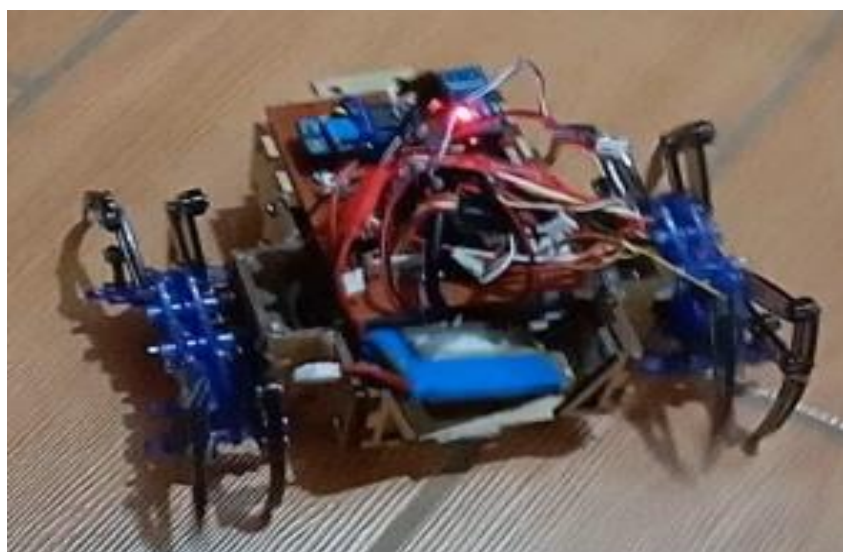
<b>Motorreductor Novamax 6V 400RPM</b>	<ul style="list-style-type: none"> <li>• Tensión normal de funcionamiento: 6V</li> <li>• Máximo. Voltaje de funcionamiento: 16V.</li> <li>• Velocidad: 400 rpm (rpm)</li> <li>• Corriente de funcionamiento: 190mA</li> <li>• Detener actual: 3.65A</li> <li>• Par de carga nominal: 4,4 kg / cm</li> <li>• Parada de par: 6,5 kg / cm</li> <li>• Diámetro del motor: 16mm</li> <li>• Eje: 3mm D Eje Outlet Center</li> <li>• Longitud del eje: 8mm</li> <li>• Longitud: 48 mm (Adecuado para Mini Sumo Reglas de Competición de robots)</li> <li>• Peso: 24 gr.</li> </ul>
--	---

Caracterización de los motores utilizados en el sistema mecánico, Miguel Uvillus

### 3.5 Creación Categoría Insecto con Obstáculos y sin Obstáculos

Se opta por adjuntar un prototipo de insecto de juguete al cuerpo principal Figura 3.30, para evidenciar las maniobras de cada articulación teniendo piñones para su respectivo movimiento, tomando en consideración el peso de la estructura principal y el tamaño.

Figura 3.30 Prototipo Adherido Prueba 1



Fotografía con patas de insecto de competencia, Miguel Uvillus

Al observar que dicho prototipo avanzaba correctamente se opta realizar piñones propios basándose en ecuaciones matemáticas referentes a mecánica, estipulando los parámetros esenciales como se observa en la Figura 3.31 y mediante un código en Matlab Figura 3.32

resolver rápidamente las ecuaciones con valores adecuados, los cuales mediante AutoCAD permitirán realizar un corte ideal obteniendo una piñonera apta para el trabajo propuesto.

Figura 3.31 Parámetros para piñones



Diseño para realizar el cálculo de los piñones, Miguel Uvillus

Diámetro primitivo (DP)

- $DP = Z * M$

Paso circular (PC)

- $PC = \pi * M$

Cabeza del diente (A)

- $A = M$

Espesor del diente (S)

- $S = PC * \left(\frac{19}{40}\right)$



Pie del diente (PD)

- $PD = 1.25 * M$

Radio de entalle (R)

- $R = \frac{PC}{12}$

Angulo del espesor (AS)

- $AS = \frac{360 * S}{2 * \pi * (\frac{DP}{2})}$

Figura 3.32 Cálculos de piñones

The image shows a MATLAB script editor window on the left and a Command Window on the right. The script calculates various gear parameters based on user input for the number of teeth (Z=20) and primitive diameter (DP=35). The results are displayed in the Command Window.

```
Editor - C:\Users\Miguelito D\Desktop\PROYECTO TITULACION\calculos.m
calculos.m x +
1 - clear all;close all;clc;
2 - Z=input('Ingrese el numero de dientes: ');
3 - %Z=20; %numero de dientes
4 - DP=input('Ingrese el diametro primitivo: ');
5 - %DP=35; %diametro primitivo
6
7 - %MODULO
8 - M=DP/Z;
9 - fprintf(1,'%a\n','El modulo es: [mm]')
10 - disp(M);
11 - %PASO CIRCULAR
12 - PC=pi*M
13 - fprintf(1,'%a\n','El paso circular es: [mm]')
14 - disp(PC);
15 - %CABEZA DEL DIENTE
16 - A=M
17 - fprintf(1,'%a\n','La cabeza del diente es: [mm]')
18 - disp(A);
19 - %ESPESOR DEL DIENTE
20 - S=PC*(19/40)
21 - fprintf(1,'%a\n','El espesor del diente es: [mm]')
22 - disp(S);
23 - %PIE DE DIENTE
24 - PD=1.25*M
25 - fprintf(1,'%a\n','El pie de diente es: [mm]')
26 - disp(PD);
27 - %RADIO DE ENTALLE
28 - R=PC/12
29 - fprintf(1,'%a\n','El radio de estalle es: [mm]')
30 - disp(R);
31 - %ANGULO DE ESPESOR
32 - AS=((360*S)/2*pi*(DP/2))
33 - fprintf(1,'%a\n','El angulo de espesor es: [grados]')
```

Command Window

```
Ingrese el numero de dientes: 20
Ingrese el diametro primitivo: 35
El modulo es: [mm]
1.7500
El paso circular es: [mm]
5.4978
La cabeza del diente es: [mm]
1.7500
El espesor del diente es: [mm]
2.6114
El pie de diente es: [mm]
2.1875
El radio de estalle es: [mm]
0.4581
El angulo de espesor es: [grados]
2.5843e+04
```

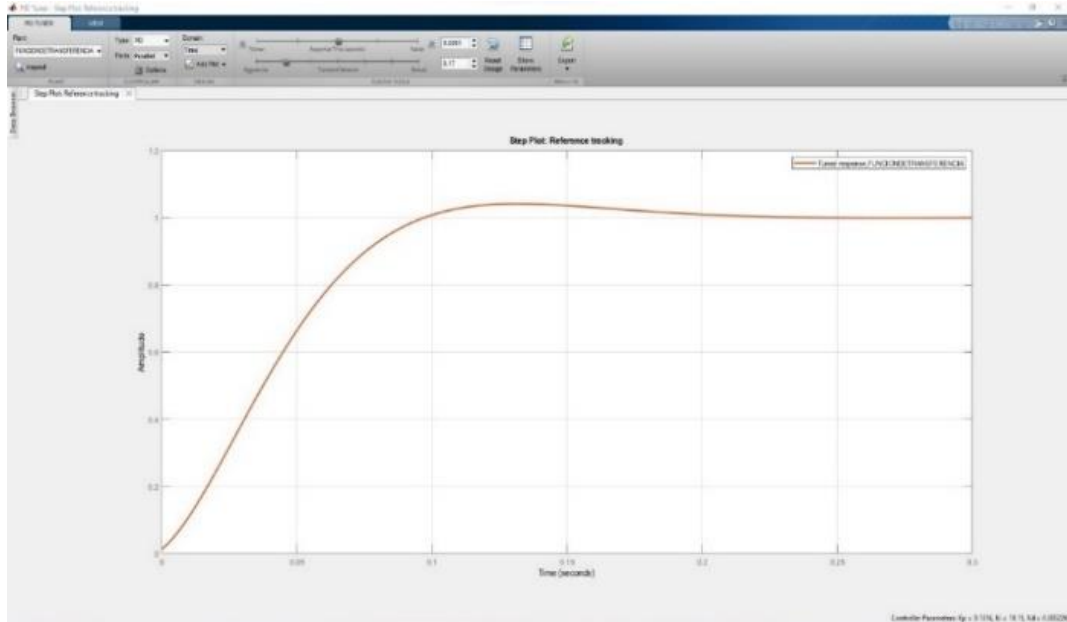
Diseño programa para cálculo de los piñones, Miguel Uvillus

Tomando en consideración que en el ámbito de competencia del robot insecto en la categoría sin obstáculos depende de una mayor velocidad, se realiza un control PID para una respuesta más optima del módulo MPU 6050 al momento de corregir su trayectoria, evitando la pérdida de su recorrido en la pista.

Manejando la plataforma PIDTuner en Matlab se procede a importar la función de transferencia encontrada anteriormente en la plataforma Ident, donde especificando el tipo de controlador

(PID) nos entrega valores de  $K_p=0.1316$ ,  $K_i=19.15$  y  $K_d=0.00023$  correspondientes como se observa en la Figura 3.33.

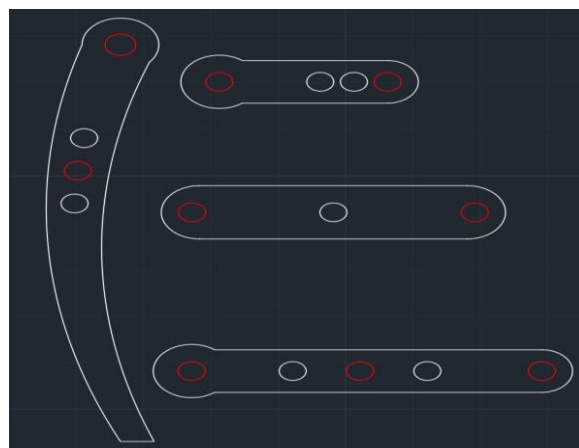
Figura 3.33 Control PID



Respuesta de estabilidad del control PID plataforma PIDTuner, Miguel Uvillus

Observando el prototipo de competencia, se elige implementar las extremidades (patas) del robot para las dos categorías de insecto Figura 3.34, realizando diversos agujeros en las piezas diseñadas para encontrar una estabilidad al momento de su movilidad.

Figura 3.34 Extremidades prueba categoría insecto

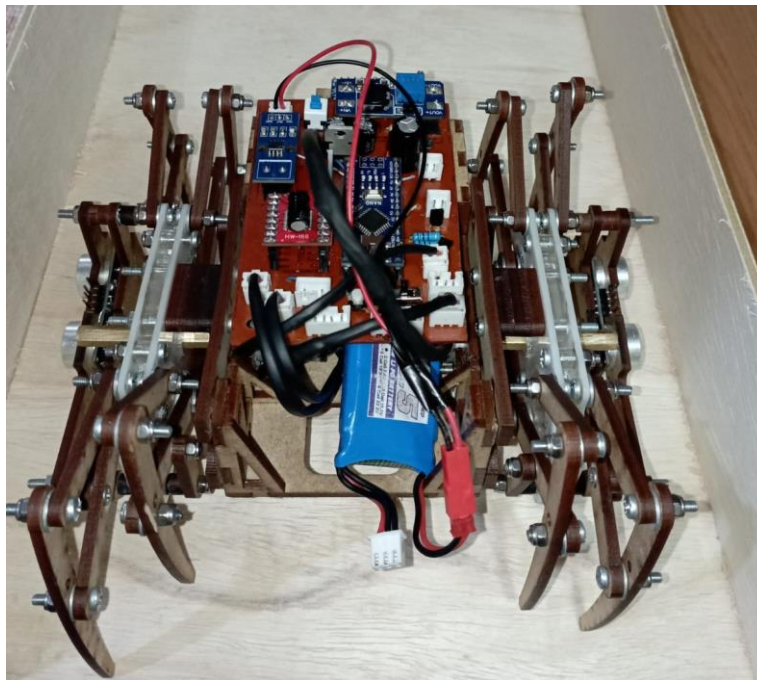


Diseño de brazos para la categoría robot insecto de diferentes medidas, Miguel Uvillus

Se observó que entre más cerca este el perno que une las extremidades a los ejes, la extremidad principal tiende a contraerse, ocasionando trabas y rupturas al momento de su movimiento de manera brusca por motivo de aumento de PWM y voltaje, mientras que, al alejarse, los pernos de los ejes proceden a extenderse demasiado provocando un movimiento incomodo forzando el rodamiento de los motores.

Inmediatamente al optimizar el nivel de las extremidades se procede a ensamblar todo el prototipo con pernos de 4mm, articulaciones de MDF, batería de 7.4V y motorreductores de 200RPM como se observa en la Figura 3.35, provocando un aumento considerable de peso y al realizar las pruebas de funcionamiento induce una lentitud al momento de caminar.

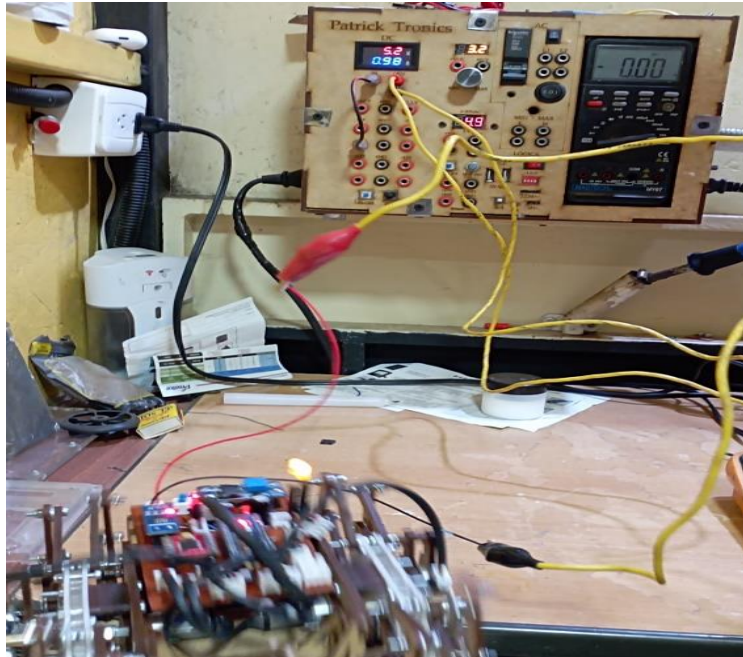
Figura 3.35 Prototipo Robot Categoría Insecto Prueba 2



Fotografía segundo prototipo de Robot para la categoría de insecto con pernos de 4mm, Miguel Uvillus

Al trabajar con una batería lipo de 2 celdas de carga, 500mA a 7.4V, se produce un aumento de corriente considerable de 0.98A como se visualiza en la Figura 3.36, al tener un abastecimiento mínimo los picos de corriente aumentan tanto por el peso y tamaño provocando inconvenientes como el colapso del módulo MPU 6050, sensores ultrasónicos y driver de los motores evitando un control adecuado de la trayectoria.

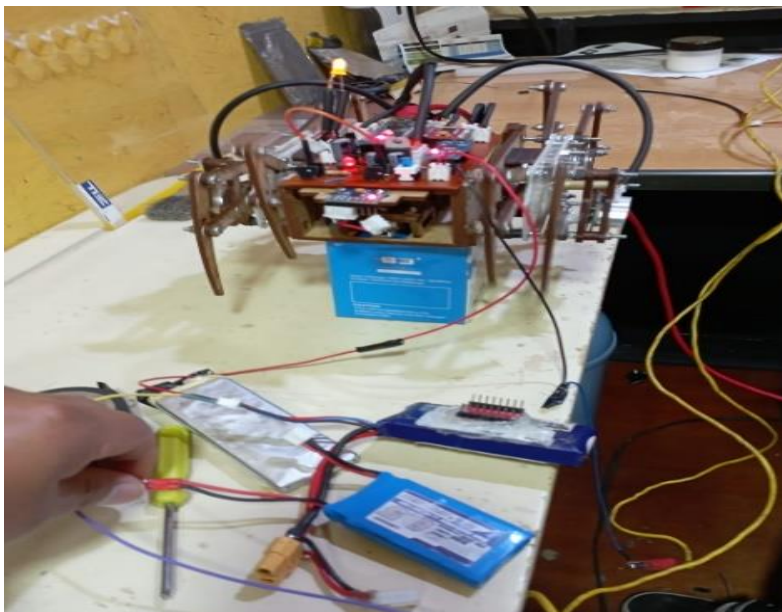
Figura 3.36 Consumo de corriente robot insecto



Fotografía segundo prototipo de Robot para la categoría de insecto, Miguel Uvillus

Al realizar una conexión momentánea acoplando dos baterías diferentes Figura 3.37, se obtuvo tres celdas de carga, 850mA a 12V obteniendo un resultado favorable, evitando la pérdida de datos y reduciendo el consumo de corriente.

Figura 3.37 Acople de baterías



Fotografía segundo prototipo de Robot para la categoría de insecto aumentando una celda de carga y corriente, Miguel Uvillus

Observando los contratiempos producidos y para mejorar tanto la competitividad se realiza el cambio a pernos de 3mm, batería lipo de tres celdas 950mA a 12V reduciendo el peso aumentando la velocidad y estabilizando la estructura principal Figura 3.38, al realizar dichas mejoras los motores aumentaron su capacidad de fuerza sometiéndolos a 15V sin consumir demasiada corriente.

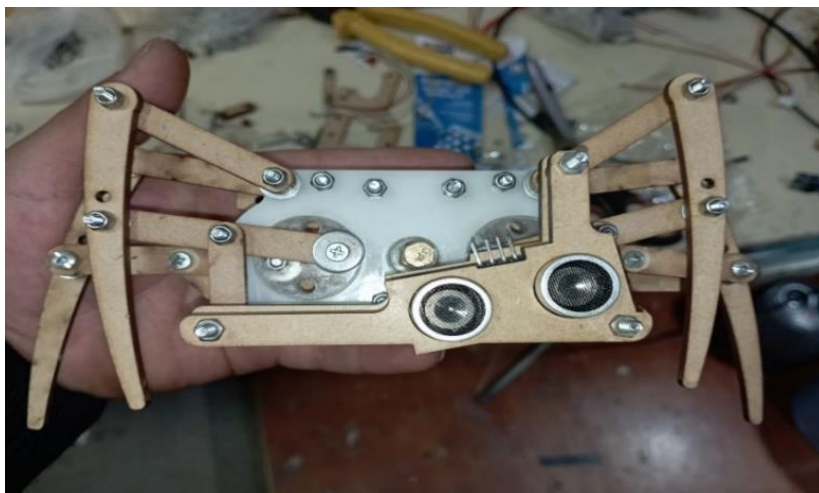
Figura 3.38 Prototipo Robot Categoría Insecto Prueba 3



Fotografía tercer prototipo de Robot para la categoría de insecto con pernos de 3mm, Miguel Uvillus

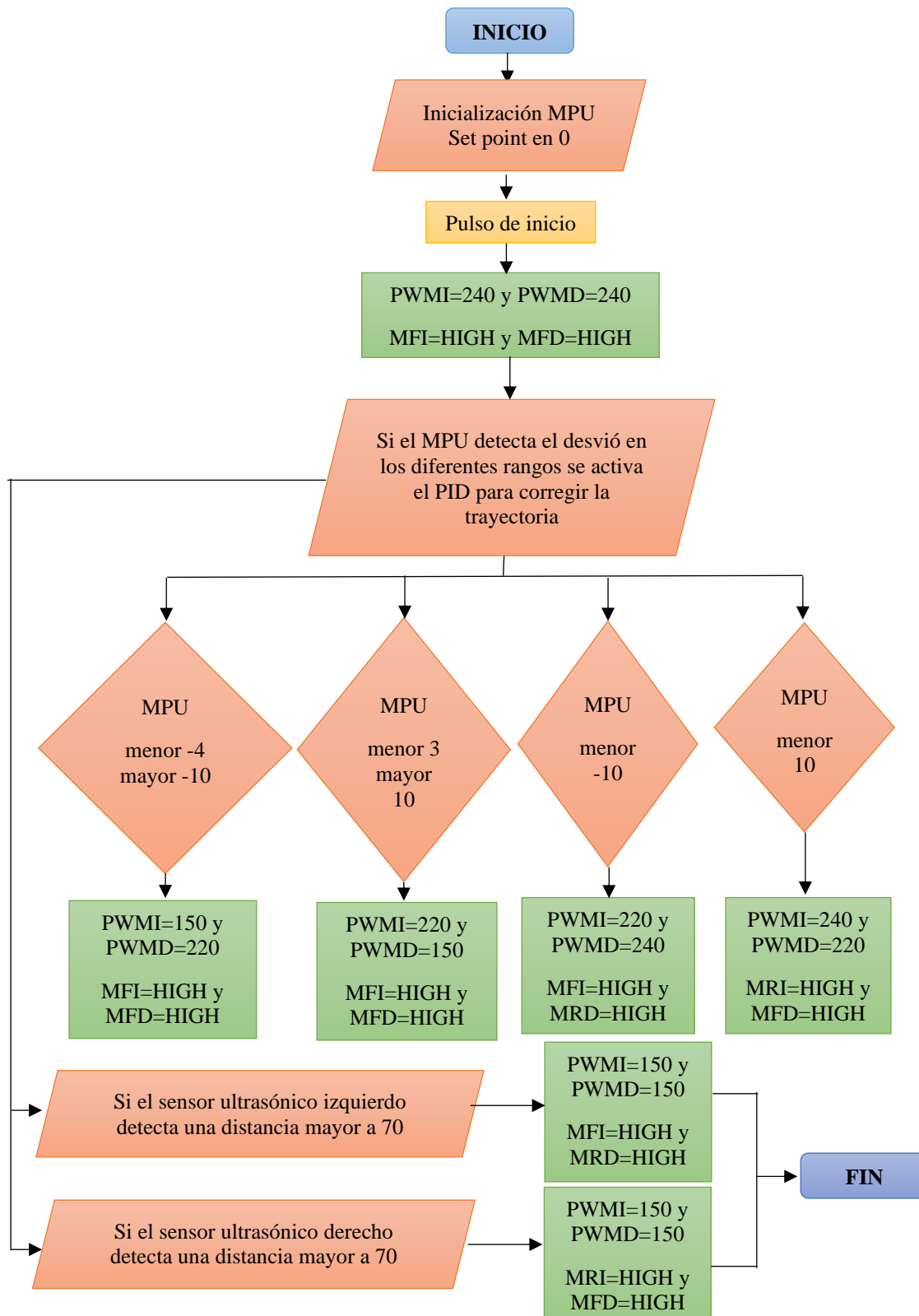
Teniendo un control de trayectoria eficaz en la pista sin obstáculos se procede a implementar sensores ultrasónicos a los lados de las extremidades de manera individual Figura 3.39, ayudando a intervenir en el trayecto de la pista por motivo de una desviación pronunciada al momento de pasar por encima de los obstáculos.

Figura 3.39 Prototipo Robot Categoría Insecto con sensores ultrasónicos



Fotografía tercer prototipo de Robot para la categoría de insecto con sensores ultrasónicos, Miguel Uvillus

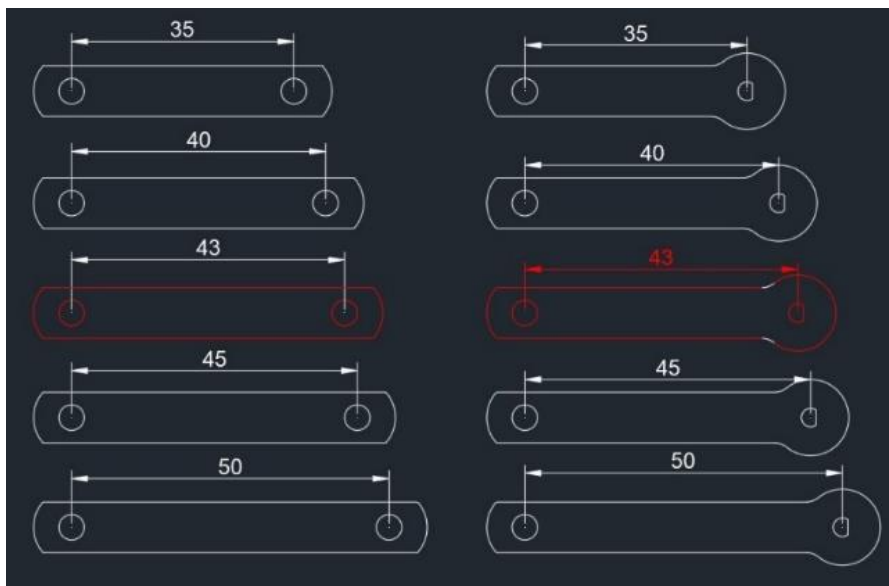
### 3.6 Diagrama de Flujo Robot Insecto con y sin Obstáculos



### 3.7 Creación Robot Categoría Trepador

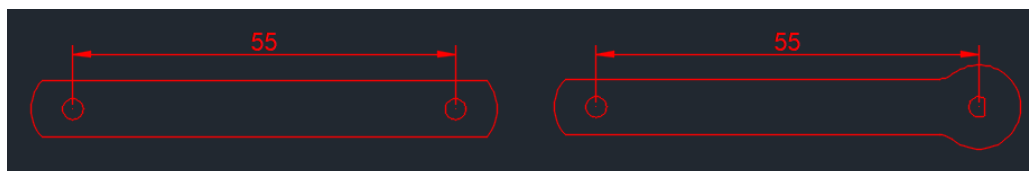
Se empezó diseñando los brazos de diferentes medidas con centros exactos para evitar variaciones al momento de ir ensamblando las piezas junto con los pernos, en sí, para los brazos adheridos a los motores el materia de acrílico fue el más utilizado evitando carcomerse internamente e impidiendo realizar juego al ingresar al acople, las demás extremidades al ser ejes igualmente se utiliza acrílico, reduciendo el peso general y la fricción al momento de su accionar, en este caso se tomó modelados diferentes de centro a centro como se puede observar en la Figura 3.40, Figura 3.41 utilizando los motorreductores de 200RPM y motorreductores de 100RPM, los cuales se utilizaron finalmente en el prototipo y en la Tabla 4 los resultados.

Figura 3.40 Extremidades de diferentes tamaños con motorreductores de 200RPM



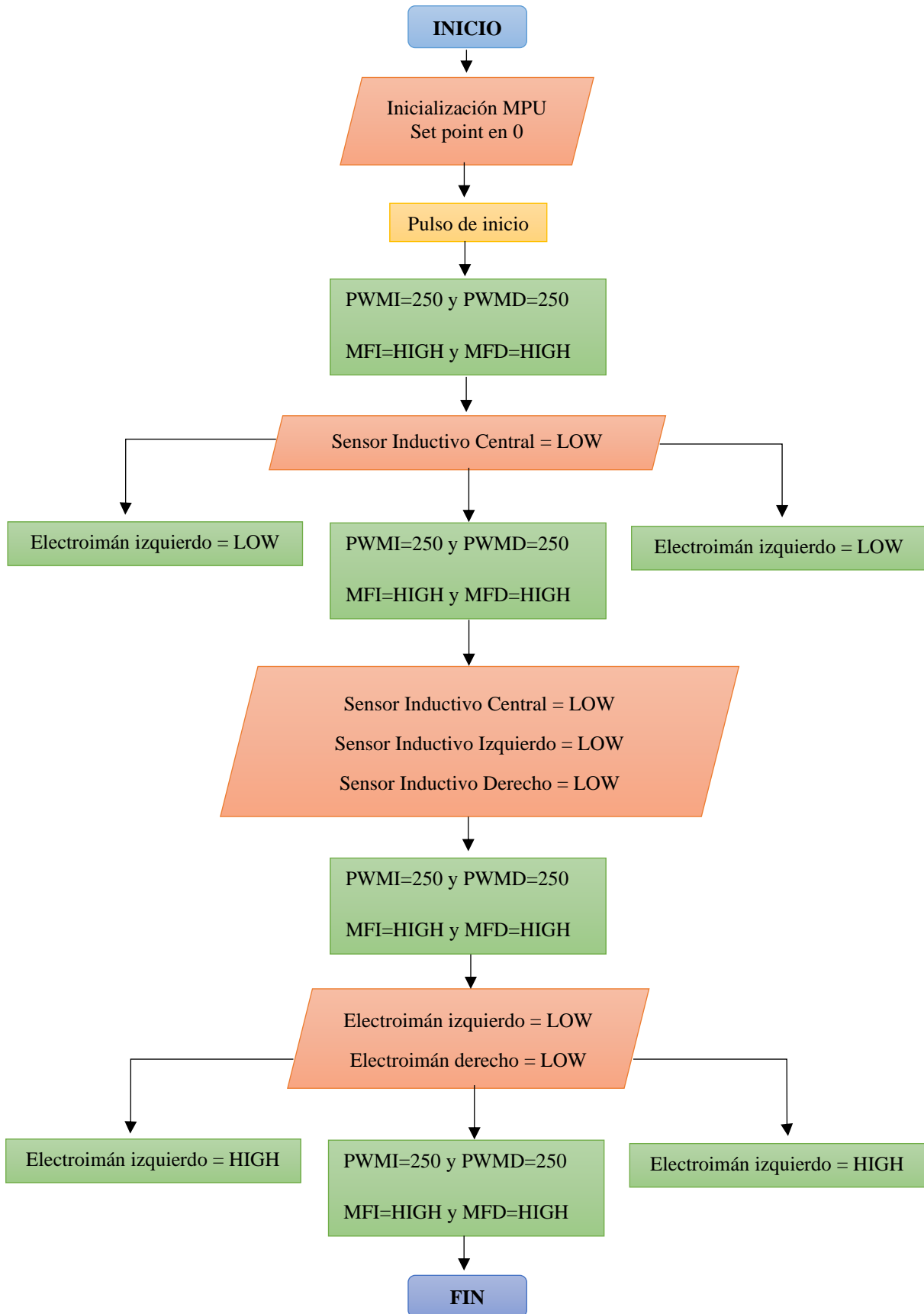
Diseño de brazos para la categoría robot trepador de diferentes medidas en acrílico, Miguel Uvillus

Figura 3.41 Extremidades finales con motorreductores de 100RPM



Diseño de brazos finales para la categoría robot trepador en acrílico, Miguel Uvillus

### 3.8 Diagrama de Flujo Robot Trepador





### 3.9 Características del prototipo

- Tamaño categoría trepador: 16.9 x 19.3 (cm)
- Tamaño categoría insecto: 18.9 x 19.5 (cm)
- Peso categoría trepador: 0.50 kg
- Peso categoría insecto: 0.65 kg

### 3.10 Resumen de costos

Tabla 3.3 Presupuesto final del proyecto

Tipo	Descripción	cantidad	precio Unitario	precio final
<b>Materiales</b>	Arduino Nano + Cable	1	\$8	\$8
	Cargador B3 pro (2s-3s) + conector hembra y macho	1	\$16.50	\$16.50
	Batería Lipo TCBWorth 3s 11.1V 900mA	1	\$28	\$28
	Electroimán 12vdc 2.5kg cerradura alarma Arduino	2	\$7	\$14
	Sensor de proximidad inductivo NPN Sn04-n	3	\$11.75	\$35.25
	Driver Puente H L298N	1	\$5.50	\$5.50
	Micro Ph Conector Jst Plug Conector Macho Y Hembra	24	\$0.95	\$28.50
	Pernos, arandelas y tuercas de seguridad 3mm y 4mm	70	\$0.75	\$52.50
	Micromotor 6v 500rpm 60:1 Pololu con encoder	2	\$12.50	\$25
	Motor Reductor Dc Novamax 400 rpm 6V	2	\$25	\$50
	Motor Reductor Dc sin datasheet 100 rpm 12V	2	\$16.50	\$33
	Elementos varios		\$45	\$45
<b>Servicios externos</b>	Torno y maquinas por hora	10	\$2.50	\$25
	Cortes en MDF y Acrílico por minuto	150 min	\$0.50	\$75
	Impresión 3D por hora	19 horas	\$1.90	\$36.10

<b>Gastos varios</b>	horas de ingeniería + transporte	80 horas	\$3	\$240
<b>Valor Total de hardware y software</b>				\$441.25
<b>Valor total trabajo de ingeniería + transporte</b>				\$240
<b>Valor total</b>				<b>\$ 681.25</b>

Tabla de costo de proyecto, Miguel Uvillus

## CAPITULO IV

### 4 PRUEBAS Y RESULTADOS DE FUNCIONAMIENTO

Para realizar la validación del prototipo se realizaron tentativas desde el inicio hasta el final, verificando el hardware, software, basándose en pruebas y errores, en este caso por los cortes realizados al material MDF, Acrílico e impresión en 3D, lo cual se evidencio que un milímetro afecta al movimiento en general creando trabas, y daños en la estructura, también se determinó posibles fallas en el software corrigiéndolas y finalmente se verifico su efectividad en los diferentes campos de competencia.

En la Tabla 4.1 se especificará el estudio y la diferencia de los cortes realizados en el material MDF para los brazos que están acoplados a los motores en la categoría robot trepador:

Tabla 4.1 Extremidades implementadas

<b>Tamaño de Brazos Con motores de 100 RPM y 200 RPM</b>	<b>Resultado Obtenido en (1.20m)</b>
<b>35 mm (200RPM)</b>	<ul style="list-style-type: none"><li>• Brazadas (16 a 18)</li><li>• Tiempo de competencia (mayor 11 seg)</li><li>• Muy inestable</li></ul>
<b>40 mm (200RPM)</b>	<ul style="list-style-type: none"><li>• Brazadas (15 a 16)</li><li>• Tiempo de competencia (mayor 9 seg)</li><li>• Inestable</li></ul>
<b>43 mm (200RPM)</b>	<ul style="list-style-type: none"><li>• Brazadas (14 a 15)</li><li>• Tiempo de competencia (7 a 9 seg)</li><li>• Poca Estabilidad</li></ul>
<b>44 mm – 50 mm (200RPM)</b>	<ul style="list-style-type: none"><li>• Brazos muy largos</li><li>• Tiempo de competencia (ninguno)</li></ul>
<b>55mm (100RPM)</b>	<ul style="list-style-type: none"><li>• Brazadas (12)</li><li>• Tiempo de competencia (7 a 9 seg)</li><li>• Estable</li></ul>
<b>56 mm – 60 mm (100RPM)</b>	<ul style="list-style-type: none"><li>• Brazos muy largos</li><li>• Tiempo de competencia (ninguno)</li></ul>

Caracterización de los brazos utilizados en el sistema mecánico categoría trepador, Miguel Uvillus

En la Tabla 4.2 se presenta los tiempos realizados por el robot en la categoría trepador a una altura de 1.20m, con motores de 200RPM, variando el PWM y 15V, considerando un tiempo de estudio de 8.04seg, manteniendo un lapso de competencia adecuado.

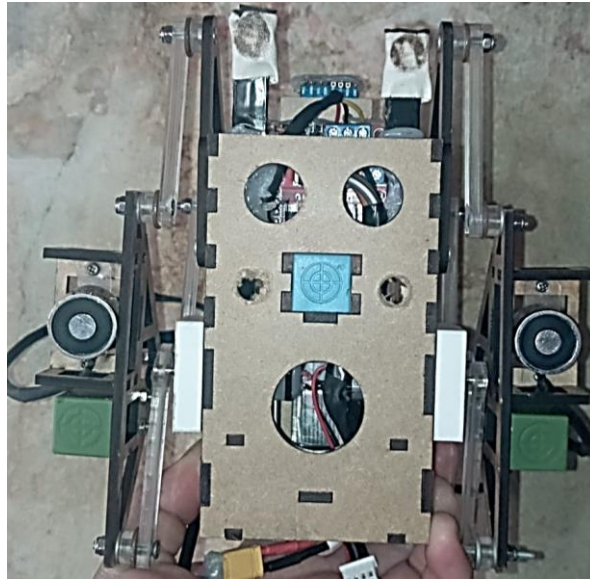
Tabla 4.2 Tiempos de competencia robot trepador 200RPM a 15V

<b>N°</b>	<b>PWM MOTORES</b>	<b>TIEMPOS DE PRUEBA (seg)</b>
<b>1</b>	230	Ninguno
<b>2</b>	230	Ninguno
<b>3</b>	235	11.10
<b>4</b>	235	11.05
<b>5</b>	240	10.42
<b>6</b>	245	9.85
<b>7</b>	250	9.16
<b>8</b>	255	8.04
<b>9</b>	255	8.17
<b>10</b>	255	8.15

Tiempos de prueba variando PWM, Miguel Uvillus

Obteniendo un resultado favorable al momento de ya implementar el sistema para la categoría de robot trepador con dos electroimanes, tres sensores inductivos, 2 imanes de neodimio y el ajuste completo con pernos de 3mm a la estructura Figura 4.1, realizando un ascenso eficaz, se toma en cuenta que al aumentar la velocidad en los motores provoca un desequilibrio notorio y una desviación, la cual provoca una pérdida de tiempo en la competencia, al aumentar más de 15V provoca un daño a los piñones por realizar demasiado esfuerzo, en este caso la manera para optimizar el prototipo es utilizar motorreductores con un buen torque y menor velocidad, articulaciones más larga que permitan menos brazadas aumentando el avance que articulaciones cortas que aumente las brazadas y desequilibre al prototipo, provocando resbalones. También se debe considerar la condición de la pista debido a que los electroimanes de cada brazo no tienden adherirse simultáneamente y se pierde el balance al momento del ascenso.

Figura 4.1 Robot trepador con sus componentes



Implementación para la categoría robot trepador, Miguel Uvillus

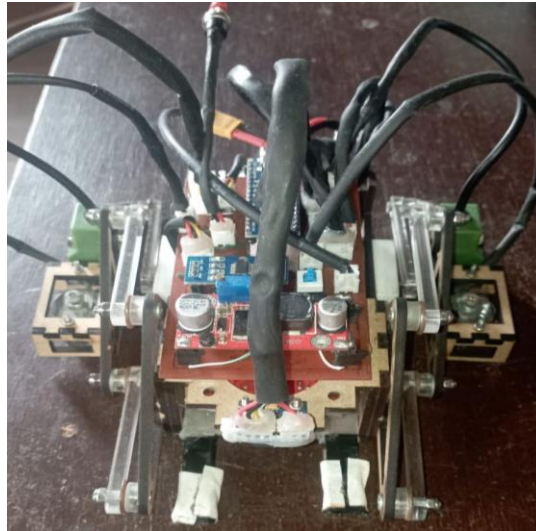
En la Tabla 4.3 se presenta los tiempos realizados por el prototipo en la categoría trepador a una altura de 1.20m, con motorreductores de 100RPM a 18V, variando el PWM con extremidades más largas como se observa en la Figura 4.2.

Tabla 4.3 Tiempos de competencia robot trepador 100RPM

N°	PWM MOTORES	TIEMPOS DE PRUEBA (seg)
1	230 a 15V	Ninguno
2	235 a 16V	11.30
3	235 a 17V	11.25
4	240 a 18V	10.50
5	240 a 18V	10.54
6	245 a 17V	9.18
7	250 a 18V	9.07
8	250 a 17V	8.91
9	250 a 18V	8.03
10	252 a 18V	8.15
11	252 a 19V	8.13
12	253 a 18V	8.24
13	253 a 19V	8.15
14	254 a 18V	8.14
15	254 a 19V	8.11
16	255 a 18V	7.04 Estable
17	255 a 19V	7.35
18	255 a 19V	7.30
19	255 a 20V	7.59 Inestable
20	255 a 21V	Ninguno

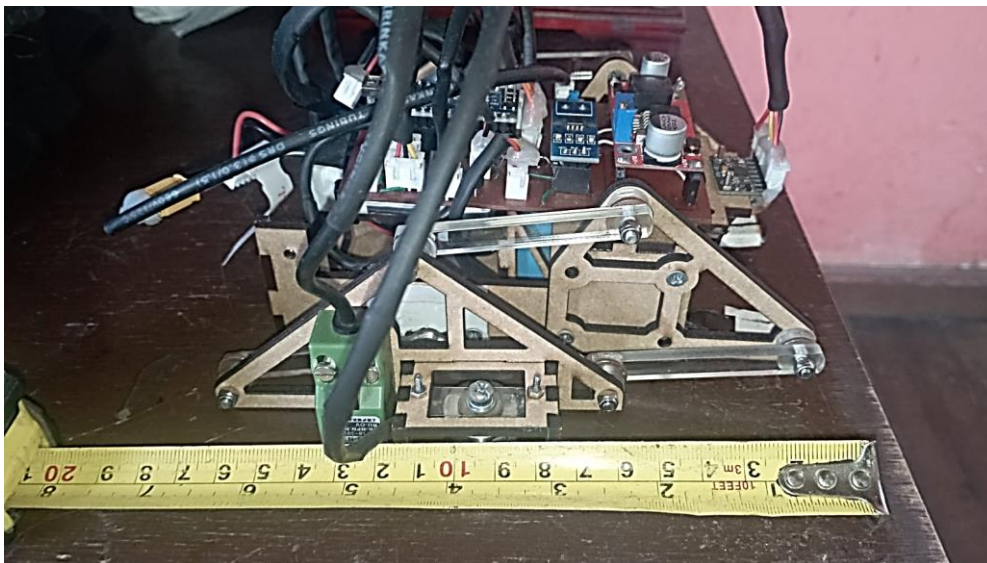
Tiempos de prueba variando PWM, Miguel Uvillus

Figura 4.2 Robot trepador



Implementación para la categoría robot trepador, Miguel Uvillus

Figura 4.3 Robot trepador Final



Implementación para la categoría robot trepador, Miguel Uvillus

#### 4.1 Robot Categoría Insecto con y sin Obstáculos

Al ponerlo a prueba en la pista Figura 4.3, el funcionamiento fue adecuado, al intervenir el MPU 6050 y los sensores ultrasónicos producen una reacción inmediata ordenando a los motorreductores a retroceder o seguir adelante en el trayecto de la pista recuperando su set point el cual es seteado antes de iniciar la competencia.

Figura 4.4 Prototipo Robot Categoría Insecto



Implementación para la categoría robot insecto con obstáculos y sin obstáculos, Miguel Uvillus

En la Tabla 4.4 se presenta los tiempos realizados por el prototipo en la categoría insecto con obstáculos (piedra-piedra-madera) y sin obstáculos en una distancia de 2m.

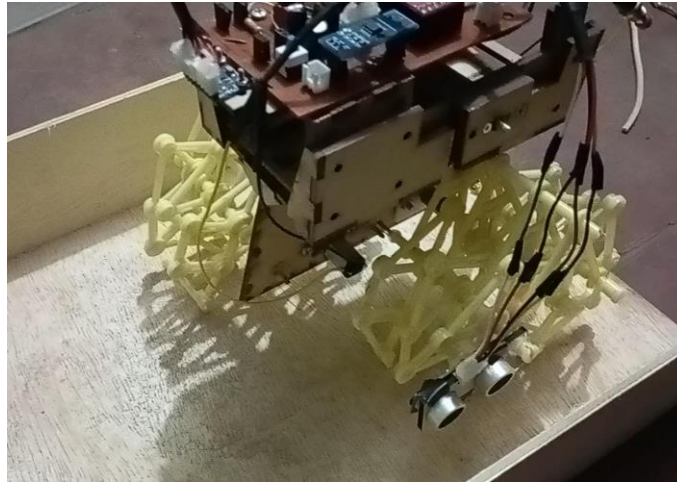
Tabla 4.4 Tiempos de competencia robot insecto motores de 200RPM a 15V

N°	PWM MOTORES	Tiempos de Prueba Sin Obstáculos (seg)	Tiempos de Prueba Con Obstáculos (seg)
1	230	16.89	25.85
2	240	16.25	23.85
3	240	16.29	23.20
4	240	16.08	23.35
5	240	16.20	23.18
6	250	15.10	21.30
7	250	15.21	21.10
8	250	15.90	21.13
9	250	15.05	20.95
10	250	15.26	21.05

Tiempos de prueba variando PWM, Miguel Uvillus

Para ir mejorando los tiempos de competencia se opta por acoplar el mecanismo Theo Jansen Figura 4.4, con motorreductores N20 de 300RPM, obteniendo un resultado nada favorable por el cumplimiento del reglamento y el prototipo no tenía un balance adecuado, se puede tomar muy en cuenta el uso de este mecanismo en competencias abiertas por su movimiento.

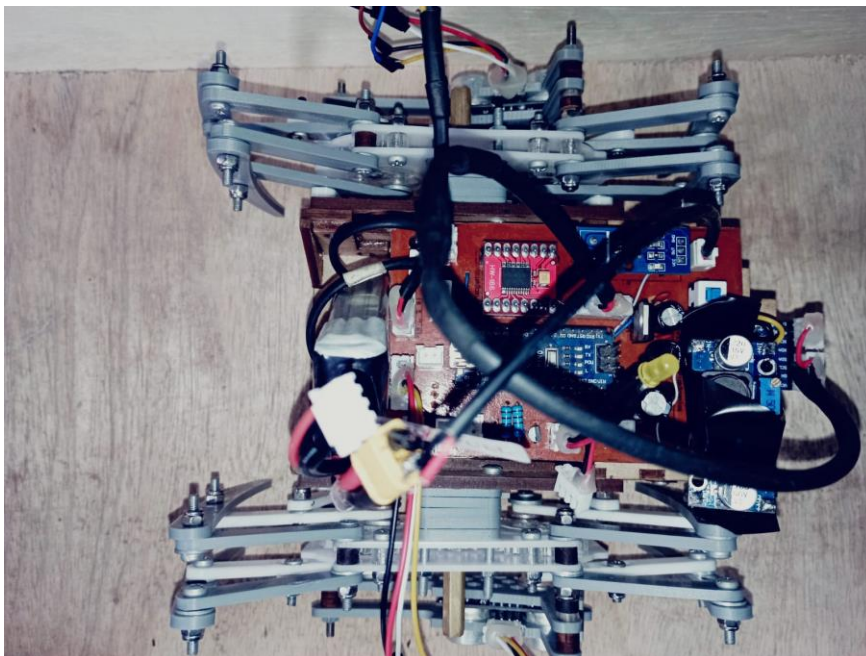
Figura 4.5 Prototipo Robot Categoría Insecto Prueba 4 Mecanismo Theo Jansen



Fotografía cuarto prototipo de Robot para la categoría de insecto, Miguel Uvillus

Optando en mejorar el prototipo, se realizan cambios como: las patas del robot insecto en impreso 3D, con motorreductores de 6V a 400RPM utilizados para competencias de sumo y mini sumo Figura 4.5, minorizando el peso y ayudando a obtener una corrección de trayectoria adecuada, una velocidad idónea para las competencias en las dos categorías.

Figura 4.6 Robot Categoría Insecto



Fotografía cuarto prototipo de Robot para la categoría de insecto, Miguel Uvillus



En la Tabla 4.5 se muestra los tiempos realizados por el prototipo en la categoría insecto con obstáculos (piedra-piedra-madera) y sin obstáculos en una distancia de 2m.

Tabla 4.5 Tiempos de competencia robot insecto motores de 400RPM

<b>N°</b>	<b>PWM MOTORES</b>	<b>Tiempos de Prueba Sin Obstáculos (seg)</b>	<b>Tiempos de Prueba Con Obstáculos (seg)</b>
<b>1</b>	200 a 7V	12.15	16.85
<b>2</b>	200 a 8V	11.34	15.65
<b>3</b>	220 a 8V	10.48	16.45
<b>4</b>	230 a 9V	10.43	15.02
<b>5</b>	230 a 9V	9.84	14.25
<b>6</b>	235 a 8V	8.49	13.10
<b>7</b>	235 a 9V	8.36	12.45
<b>8</b>	240 a 10V	7.21	10.95
<b>9</b>	245 a 8V	7.40	10.15
<b>10</b>	245 a 8V	7.32	11.42
<b>11</b>	248 a 8V	12.15	10.85
<b>12</b>	249 a 9V	11.34	11.65
<b>13</b>	250 a 8V	10.48	10.45
<b>14</b>	250 a 9V	7.98	10.02
<b>15</b>	252 a 9V	9.84	12.25
<b>16</b>	252 a 8V	8.49	11.10
<b>17</b>	253 a 8V	8.36	12.45
<b>18</b>	254 a 9V	7.24	11.95
<b>19</b>	255 a 9V	7.40	11.51
<b>20</b>	255 a 10V	Ninguno Inestable	Ninguno

Tiempos de prueba variando PWM, Miguel Uvillus

## CONCLUSIONES

En este trabajo se evidenció que en el prototipo trepador utiliza electroimanes y al ser programado ayuda a eliminar trabas al momento de su adhesión, mejorando su estabilidad eliminando la desviación al trepar, junto con motorreductores los cuales pueden ser controlados mediante su PWM y voltaje, los cual al trabajar en conjunto producen un acompañamiento eficaz junto al sistema mecánico, se debe tomar en consideración la contextura de la pista o muchas veces el pintado u oxidado lo cual el uso de imanes de neodimio ayuda a soportar completamente el peso en el transcurso de la competencia.

Dentro del análisis para la categoría de insecto con obstáculos el uso del módulo MPU 6050 es fundamental evitando desviaciones violentas, ayudando a mantener un control de velocidad máxima en la competición, mientras tanto en la categoría de insecto con obstáculos la implementación de sensores ultrasónicos mediante un control ON/OFF combinado refuerzan al prototipo, además al tener desviaciones pronunciadas por los obstáculos consiguen sacar de rumbo al módulo MPU 6050 y realizan un llamado inmediato a la corrección de la trayectoria variando el movimiento y velocidad de los motores alcanzando nuevamente su set point lo más eficaz posible.

Se concluye que al realizar un prototipo desmontable múltiple es fundamental el uso de materiales ligeros, al realizar un diseño propio se puede establecer los parámetros de medida adecuados para cada elemento a ser colocado en la estructura, realizando un trabajo óptimo y tomando en consideración el manejo del software AutoCAD que permite optimizar procesos al momento de realizar los cortes laser en MDF, ayudando a ahorrar tiempo y dinero., en otro caso el utilizar un modelado 3D ayuda a disminuir en parte el peso con un único problema de ir realizando las piezas a prueba y error lo cual provoca pérdida de tiempo al máximo por pieza, obteniendo una diferencia notoria al montar el prototipo con impreso 3D y MDF.

Las pruebas de funcionamiento con el prototipo en la categoría trepador alcanzaron un tiempo entre 7 a 9seg variando el PWM y el voltaje hacia los motores, teniendo un rango máximo de 18V de salida, evitando daños al producir una fuerza extrema al trepar, al ir sobrepasando dicho valor tanto por el peso y el manejo brusco del autómat, se desgastan los piñones del motorreductor provocando daños y la invalidación del mismo.

El prototipo en la categoría insecto sin obstáculos estableció un tiempo entre 7 y 10seg con un PWM entre 220 y 255, entre 9 y 11V , teniendo un control contiguo y realizando una competencia eficaz, en la categoría de insecto con obstáculos se estableció un tiempo entre 10 y 16seg teniendo en cuenta los obstáculos colocados en la pista en este caso (piedras-piedras-madera), con un PWM entre 220 y 240, entre 9 y 11V, el tiempo puede ir variando tanto por las desviaciones provocadas y la autocorrección de la trayectoria, se debe tomar en cuenta la fricción ya que puede dañar la comunicación de los sensores o el módulo MPU 6050 interrumpiendo el control.

Se concluye que el proyecto presentado puede generar nuevas expectativas y mejoras a futuro, con cada avance de la tecnología, innovación de elementos electrónicos, personas interesadas en robótica, entre otras, se toma en consideración que muchas competencias no son estandarizadas solamente de ámbito abierto, al generalizar las reglas pueden forjar ambiciones sanas al momento de crear diferentes prototipos.

## RECOMENDACIONES

Evitar el uso de imanes de neodimio con una fuerza de atracción elevada, debido a que desestabilizan la estructura principal provocando más torque en los motores y disminuyendo la velocidad al escalar, la utilización de electroimanes y sensores de proximidad inductivos ayudan a un mejor control, lo cual implementando diferente hardware y software se puede obtener un resultado óptimo.

Evitar el exceso de peso en el prototipo lo cual disminuye el consumo de corriente, eligiendo de manera adecuada los elementos electrónicos a ser conectados en la placa, tomando las medidas exactas, organizando de manera idónea el circuito para evitar colocar cables sobre montados que conecten a los puertos.

Desde un inicio se debe elegir el material va a ser estructurado el prototipo para evitar gastos innecesarios, tomando en cuenta la reducción del peso lo cual es muy importante al momento de realizar pruebas de funcionamiento evitando dañar elementos importantes como motores.

Al momento de probar la funcionalidad del prototipo tomar en cuenta las características de cada elemento al conectarse, evitando daños internos así mismo conocer los campos de competencia debido a que no siempre el prototipo va a funcionar de igual manera por lo que se debe ir programando y adecuando a la competición.

Evitar la creación de piñones en MDF, debido a que provoca fricción y desgaste al momento de su movimiento, evitando tener la velocidad propia de los motores y no se puede colocar ningún producto que ayude a suavizar la piñonera.

Realizar los ajustes pertinentes de pernos evitando desgaste de las piezas y un ajuste forzado, verificando la movilidad por motivos de trabas.

## REFERENCIAS

- Adeva, R. (24 de noviembre de 2022). *ZONA INFOCOMPUTER*. Obtenido de Todo lo que debes saber sobre la impresión 3D y sus utilidades:  
<https://www.adslzone.net/reportajes/tecnologia/impresion-3d/>
- AranaCorp*. (2 de noviembre de 2020). Obtenido de Usando un módulo MPU6050 con Arduino: <https://www.aranacorp.com/es/usando-un-modulo-mpu6050-con-arduino/>
- BUNTING*. (2023). Obtenido de ¿QUÉ ES UN IMÁN DE NEODIMIO?:  
<https://buntingmagnetics.com/es/blog-de-la-industria/que-es-un-iman-de-neodimio/>
- Carvajal Rojas, J. H. (2005). Metodología de diseño mecatrónico de robots. 91-101.
- CLR COMPAÑÍA LEVANTINA DE REDUCTORES*. (3 de junio de 2022). Obtenido de Todo lo que debes saber sobre los motorreductores y sus componentes:  
<https://clr.es/blog/es/todo-lo-que-debes-saber-sobre-los-motorreductores-y-sus-componentes/>
- GABYCAR Soluciones para Carpintería y Obra*. (6 de diciembre de 2021). Obtenido de TABLEROS MDF: ¿CUÁLES SON SUS CARACTERÍSTICAS?:  
<https://gabycar.com/materiales/cuales-son-las-caracteristicas-de-los-tableros-mdf/>
- García, j. (2018). *SCIELO*. Obtenido de <https://n9.cl/ibpn0>
- J. D. (4 de marzo de 2020). *Arduino Nano Pinout y características*. Obtenido de <https://www.electrogeekshop.com/arduino-nano-pinout-y-caracteristicas/>
- Jansen, T. (2007). *The Great Pretender*. Rotterdam: Publishers.
- Keisner, A., Raffo, J., & Sacha Wunsch, V. (diciembre de 2016). *Tecnologías revolucionarias: robótica y P.I.* Obtenido de OMPI:  
[https://www.wipo.int/wipo\\_magazine/es/2016/06/article\\_0002.html](https://www.wipo.int/wipo_magazine/es/2016/06/article_0002.html)
- MADE Mecatrónica, Automatización y Diseño Electrónico*. (s.f.). Obtenido de Sensor Ultrasónico HC-SR04: <https://electronicamade.com/sensor-ultrasonico/>
- Mora, W. B. (18 de septiembre de 2020). Desarrollo de aplicaciones prácticas de sistemas dinámicos en MatLab/Simulink y Octave para la asignatura de control y automatismo. Guayaquil, Guayas, Ecuador. Obtenido de <http://repositorio.ucsg.edu.ec/handle/3317/15529>
- Morales Rodríguez, N. (16 de septiembre de 2018). Uso de nuevas tecnologías para la competitividad del país. *Investiga.TEC*, 33.
- NAYLAMP MECHATRONICS SAC*. (2021). Obtenido de SENSOR DE PROXIMIDAD INDUCTIVO SN04-N (NPN-NO): <https://naylampmechatronics.com/sensores-proximidad/404-sensor-de-proximidad-inductivo-sn04-n-npn-no.html>
- NAYLAMP MECHATRONICS SAC*. (2021). Obtenido de DRIVER PUENTE H L298N 2A: <https://naylampmechatronics.com/drivers/11-driver-puente-h-l298n.html>

- Planas, O. (14 de septiembre de 2021). *ENERGÍA NUCLEAR*. Obtenido de <https://energia-nuclear.net/energia/energia-electrica/ejemplos/electroimanes>
- Rodríguez, & Morales, N. (16 de septiembre de 2018). *TEC Tecnológico de Costa Rica*. Obtenido de [https://revistas.tec.ac.cr/index.php/investiga\\_tec/article/view/3875](https://revistas.tec.ac.cr/index.php/investiga_tec/article/view/3875)
- Salazar Zambrano, E. I. (2 de junio de 2020). *Análisis comparativo de técnicas de optimización para la sintonización de controladores PID adaptativos*. Obtenido de <https://bibdigital.epn.edu.ec/handle/15000/20917>
- Silva, C. W. (2016). *Sensors and Actuators - Engineering System Instrumentation* (2 ed.). Boca Ratón: CRC Press. Obtenido de <https://books.google.com.ec/books?id=ukZOCgAAQBAJ&pg=PA433&dq=sensor+encoder&hl=es-419&sa=X&ved=2ahUKEwj0uYnFvYbxAhUyF1kFHU1ZAZoQ6AEwBnoECAUQA>
- Valásquez Salazar, F. J. (2018). *Repositorio Dspace*. Obtenido de Caracterización de las propiedades mecánicas del Duralón: <http://repositorio.espe.edu.ec/xmlui/handle/21000/15085>
- Vega Castillo, P., Vílchez Monge, M., Villegas Lemus, M., & Alvarado Moya, P. (8 de diciembre de 2011). Consideraciones de diseño para robots miniaturizados. *Revista trimestral Tecnología en marcha*, 60.
- Zambrano, E. I. (Febrero de 2020). Analisis comparativo de técnicas de optimización para la sintonización de controladores PID adaptativos. Quito, Pichincha, Ecuador. Obtenido de <http://bibdigital.epn.edu.ec/handle/15000/20917>

## 5 ANEXOS

### **Instrucciones de funcionamiento**

- ✓ Primeramente, se evidencia el switch negro de dos posiciones, las cuales nos indicarán el modo de operación del prototipo: Si el switch se encuentra ubicado al lado izquierdo funcionará como robot insecto y se encuentra al lado derecho como robot trepador.
- ✓ El encendido del prototipo se lo ejecuta presionando un switch celeste ubicado junto al sensor de corriente, si esta presionado significa que el prototipo se encuentra apagado y si este se encuentra alzado se accionará.
- ✓ En el Arduino nano se observará el led interno el cual al estar apagado indicará el modo trepador y al estar encendido el modo insecto este sistema ayuda a verificar el modo de funcionamiento.
- ✓ Antes de cada competencia se alista el prototipo incorporando los acoples ya sea para la categoría de trepado o insecto mediante pernos para sujetar cada acople y conectando los sockets respectivos.
- ✓ En la categoría de trepador se coloca el robot en la línea de partida, se lo enciende y mediante un led indicador el cual mostrará que ya se puede presionar el pulsador rojo para iniciar la escalada.
- ✓ En la categoría de insecto con obstáculos y sin obstáculos se coloca el robot en la línea de partida, se enciende ubicándolo en la dirección que desee competir para obtener un punto de inicio (set point) correspondiente lo cual es muy necesario para que el MPU 6050 trabaje correctamente, mediante un led indicador mostrará que ya se puede presionar el pulsador rojo para iniciar.
- ✓ Para apagar se debe presionar el mismo switch celeste de encendido.

## Diseños Extras

Figura 5.1 Extremidad fallida categoría insecto



Fotografía diseño e implementación fallida de extremidad con pernos de 4mm y piñonearía en triángulo, Miguel Uvillus

Figura 5.2 Batería Lipo



Fotografía batería lipo 500mA a 6 [V] 2s Miguel Uvillus

Figura 5.3 Extremidad Robot Trepador en Duralón



Fotografía diseño e implementación fallida de extremidad con dularón, Miguel Uvillus



Figura 5.4 Obstáculos Robot Insecto



Fotografía trabas para la categoría insecto con obstáculos (piedras-piedras-madera), Miguel Uvillus

Figura 5.5 Peso Categoría Trepador



Fotografía peso en kilogramos del diseño para la categoría de trepador, Miguel Uvillus

Figura 5.6 Peso Categoría Insecto



Fotografía peso en kilogramos del diseño para la categoría de insecto con pernos de 3mm, Miguel Uvillus

## Programación en Arduino

```
#include "Simple_MPU6050.h" // incluye libreria Simple_MPU6050
#define MPU6050_ADDRESS_AD0_LOW 0x68 // direccion I2C con AD0 en LOW o sin conexion
#define MPU6050_ADDRESS_AD0_HIGH 0x69 // direccion I2C con AD0 en HIGH
#define MPU6050_DEFAULT_ADDRESS MPU6050_ADDRESS_AD0_LOW // por defecto AD0 en LOW
Simple_MPU6050 mpu; // crea objeto con nombre mpu
// ENABLE_MPU_OVERFLOW_PROTECTION(); // activa proteccion, ya no se requiere
// #define OFFSETS -5114, 484, 1030, 46, -14, 6 // Colocar valores personalizados
#define spamtimer(t) for (static uint32_t SpamTimer; (uint32_t)(millis() - SpamTimer) >= (t); SpamTimer = millis())
// spamtimer funcion para generar demora al escribir en monitor serie sin usar delay()
#define printfloatx(Name,Variable,Spaces,Precision,EndTxt) print(Name); {char S[(Spaces + Precision + 3)];Serial.print(F(" ")); Serial.print(dtostrf((float)Variable,Spaces,Precision ,S));}Serial.print(EndTxt);
// printfloatx funcion para mostrar en monitor serie datos para evitar el uso se multiples print()
// mostrar_valores funcion que es llamada cada vez que hay datos disponibles desde el sensor
// pines driver motores
#define MRI 4 // Motor izquierdo reversa izquierdo
#define MFI 5 // Motor izquierdo frente izquierdo
#define PWMI 3 // velocidad izquierdo
#define MRD 8 // Motor derecho reversa derecho
#define MFD 7 // Motor derecho frente derecho
#define PWMD 9 // velocidad derecho
// pines electroimanes
#define elecizq 6 // electroiman IZQUIERDO
#define elecderled A3 // electroiman DERECHO e indicador LED en modo insecto
// pines sensores inductivo y ultrasonicos
#define scentro A0 // sensor inductivo centro
#define tsuizqled 11 // tigger sensor izquierdo e indicadro led en modo trepador
#define tsuder 10 // tigger sensor derecho
// COMPARTIDOS
#define esusiizq 12 // echo sensor ultrasonico e inductivo izquierdo
#define esusider A1 // echo sensor ultrasonico e inductivo derecho
//pines pulsadores, selectores y sensor de corriente
#define start A6 // pulsador de inicio
#define modo A2 // seleccionador de robot
#define sc A7 // sensor de corriente
#define inmodos 13 // INDICADOR DE MODOS
//DECLARAMOS VELOCIDAD DEL ROBOT OJO LLEGA HASTA 255
int inicio = 0, vmpu = 0, auxvoid = 0, trepador = 0, insecto = 0, auxbrazos=0;
float disder = 0, disizq = 0, distancia = 0, centro = 0;
//VARIABLES DE ALMACENAMIENTO
int P=0;
int I=0;
int D=0;
int LAST=0, err=0;
int velmizq=0, velmder=0;
float kp=0.02; // 0.02
float kd=1.2; //1.20
float ki=0.0052; //0.01
float vel;
//DECLARAMOS VELOCIDAD DEL ROBOT OJO LLEGA HASTA 255
int velmax =250;
unsigned int position=0;
void inicializacionMPU() {
  uint8_t val;
  #if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE // activacion de bus I2C a 400 Khz
    Wire.begin();
    Wire.setClock(400000);
  #elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
    Fastwire::setup(400, true);
  #endif
  Serial.begin(115200); // inicializacion de monitor serie a 115200 bps
  while (!Serial); // espera a enumeracion en caso de modelos con USB nativo
  Serial.println(F("Inicio:")); // muestra texto estatico
```

```

#ifdef OFFSETS          // si existen OFFSETS
  Serial.println(F("Usando Offsets predefinidos")); // texto estatico
  mpu.SetAddress(MPU6050_ADDRESS_AD0_LOW).load_DMP_Image(OFFSETS); // inicializacion de sensor
#else
  // sin no existen OFFSETS
  Serial.println(F(" No se establecieron Offsets, haremos unos nuevos.\n" // muestra texto estatico
    " Colocar el sensor en un superficie plana y esperar unos segundos\n"
    " Colocar los nuevos Offsets en #define OFFSETS\n"
    " para saltar la calibracion inicial \n"
    "\t\tPresionar cualquier tecla y ENTER"));
  // while (Serial.available() && Serial.read()); // lectura de monitor serie
  // while (!Serial.available()); // si no hay espera
  // while (Serial.available() && Serial.read()); // lecyura de monitor serie
mpu.SetAddress(MPU6050_ADDRESS_AD0_LOW).CalibrateMPU().load_DMP_Image(); // inicializacion de
sensor
#endif
mpu.on_FIFO(mostrar_valores); // llamado a funcion mostrar_valores si memoria FIFO tiene valores
if (insecto == 1) {
  digitalWrite(elecderled, HIGH);
}
if (trepador == 1) {
  digitalWrite(tsuiqled, LOW);
}
}

void mostrar_valores (int16_t *gyro, int16_t *accel, int32_t *quat, uint32_t *timestamp) {
  uint8_t SpamDelay = 100; // demora para escribir en monitor serie de 100 mseg
  Quaternion q; // variable necesaria para calculos posteriores
  VectorFloat gravity; // variable necesaria para calculos posteriores
  float ypr[3] = { 0, 0, 0 }; // array para almacenar valores de yaw, pitch, roll
  float xyz[3] = { 0, 0, 0 }; // array para almacenar valores convertidos a grados de yaw, pitch, roll
  spantimer(SpamDelay) { // si han transcurrido al menos 100 mseg entonces proceder
    mpu.GetQuaternion(&q, quat); // funcion para obtener valor para calculo posterior
    mpu.GetGravity(&gravity, &q); // funcion para obtener valor para calculo posterior
    mpu.GetYawPitchRoll(ypr, &q, &gravity); // funcion obtiene valores de yaw, ptch, roll
    mpu.ConvertToDegrees(ypr, xyz); // funcion convierte a grados sexagesimales
    // Serial.printfloatx(F("Yaw") , xyz[0], 9, 4, F(", ")); // muestra en monitor serie rotacion de eje Z, yaw
    // Serial.printfloatx(F("Pitch"), xyz[1], 9, 4, F(", ")); // muestra en monitor serie rotacion de eje Y, pitch
    // Serial.printfloatx(F("Roll") , xyz[2], 9, 4, F(", ")); // muestra en monitor serie rotacion de eje X, roll
    //Serial.println(xyz[0]); // salto de linea
    vmpu = xyz[0];
  }
}

void setup() {

  // declaraciones pines motor
  pinMode(MFI, OUTPUT);
  pinMode(MRI, OUTPUT);
  pinMode(PWMI, OUTPUT);
  pinMode(MFD, OUTPUT);
  pinMode(MRD, OUTPUT);
  pinMode(PWMD, OUTPUT);
  // declaracion pines electroimanes
  pinMode(elecizq, OUTPUT); // electroiman IZQUIERDO
  pinMode(elecderled, OUTPUT); // electroiman DERECHO e indicador LED
  // pines sensores inductivo y ultrasonicos
  pinMode(scentro, INPUT); // sensor inductivo centro
  pinMode(tsuiqled, OUTPUT); // tigger sensor izquierdo
  pinMode(tsuder, OUTPUT); // tigger sensor derecho
  // COMPARTIDOS
  pinMode(esusiizq, INPUT); // echo sensor ultrasonico e inductivo izquierdo
  pinMode(esusider, INPUT); // echo sensor ultrasonico e inductivo derecho
  //pines pulsadores, selectores y sensor de corriente
  pinMode(start, INPUT); // pulsador de inicio
  pinMode(modos, INPUT); // seleccionador de robot
}

```

```

pinMode(sc, INPUT); // sensor de corriente
pinMode(inmodos, OUTPUT); // led indicador arduino
// inicializaciones pines motor
digitalWrite(MFI, LOW);
digitalWrite(MRI, LOW);
digitalWrite(MFD, LOW);
digitalWrite(MRD, LOW);
// declaracion pines electroimanes
digitalWrite(elecizq, LOW); // electroiman IZQUIERDO
digitalWrite(elecderled, LOW); // electroiman DERECHO
//digitalWrite(tsuiqled, LOW); // tigger sensor izquierdo
//digitalWrite(tsuder, LOW); // tigger sensor derecho
//pines pulsadores, selectores y sensor de corriente
//digitalWrite(start, HIGH); // pulsador de inicio
//digitalWrite(modos, HIGH); // indicador de robot
delay(500);
if (analogRead(modos) < 500) {
  insecto = 1;
  trepador = 0;
  digitalWrite(inmodos, HIGH);
  digitalWrite(elecderled, LOW);
} else {
  insecto = 0;
  trepador = 1;
  digitalWrite(inmodos, LOW);
  digitalWrite(tsuiqled, HIGH);
}
inicializacionMPU();
}
void loop() {
  //ROBOT INSECTO
  if (insecto == 1) {
    //inicio = 1;
    if (analogRead(start) < 500 && inicio==0) {
      delay(1);
      inicio = 1;
      delay(1);
      analogWrite(PWMD, 240); //250
      analogWrite(PWMI, 240); //245
      digitalWrite(MFI, HIGH);
      digitalWrite(MRI, LOW);
      digitalWrite(MFD, HIGH);
      digitalWrite(MRD, LOW);
    }
    if (inicio == 1) {
      mpu.dmp_read_fifo(); // funcion que evalua si existen datos nuevos en el sensor y llama
      Serial.println(vmpu);
      if (vmpu < -4 && vmpu >= -10){ //robot se inclina a la izquierda leve
        Serial.println("girar der");
        analogWrite(PWMD, 150); //190
        analogWrite(PWMI, 220); //250
        digitalWrite(MFI, HIGH);
        digitalWrite(MRI, LOW);
        digitalWrite(MFD, HIGH);
        digitalWrite(MRD, LOW);
        I=0;
        D=0;
        err=0;
        delay(500);
      } else if (vmpu > 3 && vmpu <= 10) { //robot se inclina a la derecha leve
        Serial.println("girar izq");
        analogWrite(PWMD, 220); //250
        analogWrite(PWMI, 150); //190

```

```

digitalWrite(MFI, HIGH);
digitalWrite(MRI, LOW);
digitalWrite(MFD, HIGH);
digitalWrite(MRD, LOW);
I=0;
D=0;
err=0;
delay(500);
} else if (vmpu < -20){ //robot se inclina a la izquierda bastante
  Serial.println("retro der");
  analogWrite(PWMD, 220); //230
  analogWrite(PWMI, 220); //255
  digitalWrite(MFI, HIGH);
  digitalWrite(MRI, LOW);
  digitalWrite(MFD, LOW);
  digitalWrite(MRD, HIGH);
  delay(500); //tiempo de regreso
  I=0;
  D=0;
  err=0;
} else if (vmpu > 10) {
  Serial.println("retro izq");
  analogWrite(PWMD, 220); //255
  analogWrite(PWMI, 220); //230
  digitalWrite(MFI, LOW);
  digitalWrite(MRI, HIGH);
  digitalWrite(MFD, HIGH);
  digitalWrite(MRD, LOW);
  delay(500);
  I=0;
  D=0;
  err=0;
} else {
  analogWrite(PWMD, 240); //190
  analogWrite(PWMI, 240); //250
  digitalWrite(MFI, HIGH);
  digitalWrite(MRI, LOW);
  digitalWrite(MFD, HIGH);
  digitalWrite(MRD, LOW);
  //CALCULO DE PID
  P = vmpu; /// ERROR
  I=I+P;
  D=P-LAST;
  LAST=P;
  err=kp*P+ki*I+kd*D; //calculo del error
  //Serial.println(err);
  velmizq=velmax-err;
  velmder=velmax+err;
  if(velmizq<1){
    velmizq=0;
  }
  if(velmder<1){
    velmder=0;
  }
}
}
disizq = medidordistancia(esusiizq, tsuizqled); // funcion medi
disder = medidordistancia(esusider, tsuder); // funcion medidor de distancia medidordistancia(pin echo, pin
tigger);
if (disizq > 70){
  Serial.println("DER");
  analogWrite(PWMD, 150); //130
  analogWrite(PWMI, 150); //255
  digitalWrite(MFI, HIGH);
  digitalWrite(MRI, LOW);

```

```

    digitalWrite(MFD, HIGH);
    digitalWrite(MRD, LOW);
    delay(800);
    I=0;
    D=0;
    err=0;
}
if (disder > 70) {
    Serial.println("DIZQ");
    analogWrite(PWMD, 150); //255
    analogWrite(PWMI, 150); //130
    digitalWrite(MFI, HIGH);
    digitalWrite(MRI, LOW);
    digitalWrite(MFD, HIGH);
    digitalWrite(MRD, LOW);
    delay(800);
    I=0;
    D=0;
    err=0;
}

Serial.print(disizq);
Serial.print(" ");
Serial.println(disder);
}
}
// ROBOT TREPADOR
if (trepador == 1) {
// SOLO AQUI TRABAJAR
if (analogRead(start) < 500 && inicio==0) {
    delay(1);
    inicio = 1;
    delay(1);
    auxbrazos=0;
    analogWrite(PWMI, 250); // CAMBIA VALORES DE VELOCIDAD EMPEZAR DESD 160 A 255
    analogWrite(PWMD, 250);
    digitalWrite(MFI, HIGH);
    digitalWrite(MRI, LOW);
    digitalWrite(MFD, HIGH);
    digitalWrite(MRD, LOW);
}
if (inicio == 1) {
if (digitalRead(scentro) == LOW) {
    analogWrite(PWMI, 250); // CAMBIAR VALORES DE VELOCIDAD EMPEZAR DESD 160 A 255
    analogWrite(PWMD, 250);
    //delayMicroseconds(1);
    digitalWrite(elecderled, LOW);
    digitalWrite(elecizq, LOW);
}
while (digitalRead(esusider) == LOW && digitalRead(esusiizq) == LOW && digitalRead(scentro) == LOW)
{
    analogWrite(PWMI, 250); // CAMBIAR VALORES DE VELOCIDAD EMPEZAR DESD 160 A 255
    analogWrite(PWMD, 250);
}
if (digitalRead(esusider) == LOW && digitalRead(esusiizq) == LOW ){
    analogWrite(PWMI, 250); // CAMBIAR VALORES DE VELOCIDAD EMPEZAR DESD 160 A 255
    analogWrite(PWMD, 250);
    digitalWrite(elecderled, HIGH);
    digitalWrite(elecizq, HIGH);
}
}
}
}
}
float medidordistancia( int echo, int tigger) {

```

```

float duracion;
digitalWrite(trigger, HIGH); // generacion del pulso a enviar
delayMicroseconds(1000); // al pin conectado al trigger
digitalWrite(trigger, LOW); // del sensor
duracion = pulseIn(echo, HIGH); // con funcion pulseIn se espera un pulso
// alto en Echo
duracion = duracion / 58.2; // distancia medida en centimetros
return duracion;}

```

## Cálculo de piñones (Matlab)

```

clear all;close all;clc;
Z=input('Ingrese el numero de dientes: ');
%Z=20; %numero de dientes
DP=input('Ingrese el diametro primitivo: ');
%DP=35; %diametro primitivo
%MODULO
M=DP/Z;
fprintf(1,'%s\n','El modulo es: [mm]')
disp(M);
%PASO CIRCULAR
PC=pi*M;
fprintf(1,'%s\n','El paso circular es: [mm]')
disp(PC);
%CABEZA DEL DIENTE
A=M;
fprintf(1,'%s\n','La cabeza del diente es: [mm]')
disp(A);
%ESPESOR DEL DIENTE
S=PC*(19/40);
fprintf(1,'%s\n','El espesor del diente es: [mm]')
disp(S);
%PIE DE DIENTE
PD=1.25*M;
fprintf(1,'%s\n','El pie de diente es: [mm]')
disp(PD);
%RADIO DE ENTALLE
R=PC/12;
fprintf(1,'%s\n','El radio de estalle es: [mm]')
disp(R);
%ANGULO DE ESPESOR
AS=((360*S)/2*pi*(DP/2));
fprintf(1,'%s\n','El angulo de espesor es: [grados]')
disp(AS);

```