

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL**

CARRERA DE INGENIERÍA ELECTRÓNICA

**TRABAJO DE TITULACIÓN PREVIO A LA
OBTENCIÓN DEL TÍTULO DE INGENIERO ELECTRÓNICO**

PROYECTO TÉCNICO:

**“DISEÑO E IMPLEMENTACIÓN DE DOS ROBOTS
SOCCER AUTÓNOMOS (MIROSOT) PARA LA
CARRERA DE ELECTRÓNICA Y AUTOMATIZACIÓN”**

AUTORES:

**Juan Carlos Guevara Davalos
Ricardo Andrés Tenorio Vélez**

TUTOR:

Ing. Byron Xavier Lima Cedillo, MSc.

GUAYAQUIL – ECUADOR

2019

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA

Nosotros, Juan Carlos Guevara Davalos y Ricardo Andrés Tenorio Vélez autorizamos a la **Universidad Politécnica Salesiana** la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además, se declara que los conceptos y análisis desarrollados y conclusiones del presente trabajo son de exclusiva responsabilidad del autor.

Juan Carlos Guevara Davalos
Cédula:0931562342

Ricardo Andrés Tenorio Vélez
Cédula: 0831546808

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR

Nosotros, **Juan Carlos Guevara Davalos**, con documento de identificación N°0931562342 y **Ricardo Andrés Tenorio Vélez**, con documento de identificación N°0831546808, manifestamos nuestra voluntad y ceder a la **UNIVERSIDAD POLITÉCNICA SALESIANA** la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de grado titulado: **“DISEÑO E IMPLEMENTACIÓN DE DOS ROBOTS SOCCER AUTÓNOMOS (MIROSOT) PARA LA CARRERA DE ELECTRÓNICA Y AUTOMATIZACIÓN”** mismo que 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 antes cedidos.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscrito este documento en el momento que se realiza la entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.

Juan Carlos Guevara Davalos
Cédula:0931562342

Ricardo Andrés Tenorio Vélez
Cédula: 0831546808

CERTIFICADO DE DIRECCIÓN DE TRABAJO DE TITULACIÓN

Yo declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación “**DISEÑO E IMPLEMENTACIÓN DE DOS ROBOTS SOCCER AUTÓNOMOS (MIROSOT) PARA LA CARRERA DE ELECTRÓNICA Y AUTOMATIZACIÓN**” con resolución de aprobación de Consejo de Carrera N° **RESOLUCION** realizado por los estudiantes **Juan Carlos Guevara Davalos**, con documento de identificación N°0931562342 y **Ricardo Andrés Tenorio Vélez**, con documento de identificación N°0831546808, obteniendo un producto que cumple con los objetivos del diseño de aprobación, informe final y demás requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Guayaquil, Fecha

Ing. Byron Xavier Lima Cedillo, MSc

Docente

C.C. 0921971768

DEDICATORIA

El presente trabajo de grado va dedicado a Dios, quien como estuvo en el caminar de mi vida, bendiciéndome y dándome fuerza para continuar con mis metas trazadas sin desfallecer.

A mis padres que con apoyo incondicional, amor y confianza permitieron que logre culminar mi carrera profesional.

A mis hermanos que están presente en mi vida los cuales aprecio y me ayudan a seguir adelante.

Juan Carlos Guevara Davalos

DEDICATORIA

Este logro lo dedico a Dios por acompañarme en todo momento, fortaleciendo mi carácter para continuar y llegar a esta etapa importante de mi profesión.

A mis padres América Vélez y Jorge Tenorio porque a lo largo de mi vida han sabido guiarme, con valores y principios que han permitido no desfallecer en momentos difíciles, gracias por acompañarme en este proceso y por ser mi apoyo incondicional para alcanzar esta meta de ser un profesional.

A mi novia María por su constante apoyo, por compartir momentos significativos en mi vida, por su predisposición en ayudarme en todo momento y quien con su amor y dedicación me acompañó en todas las instancias de este proceso de lograr uno más de mis sueños.

Ricardo Andrés Tenorio Vélez

AGRADECIMIENTO

Principalmente a Dios por guiarme y bendecirme con este logro de ser un profesional y no decaer en los momentos más difícil que tuve a lo largo de mi carrera.

A mi familia por el cariño y la confianza que me tuvo en todo el trascurso de mi preparación de mi carrea profesional.

A mi tutor de proyecto técnico el Ing. Byron Lima, MSc. Que con su apoyo supo guiarme en todo momento hasta la presente finalización de mi proyecto de grado

A los docentes, amistades y compañeros que también fueron partes del proceso de culminación de proyecto de grado

A mi amigo y compañero de tesis Ricardo Tenorio por su apoyo y amistad brindada a lo largo de toda mi preparación como profesional.

Juan Carlos Guevara Davalos.

AGRADECIMIENTO

Dejo constancia y sinceros agradecimiento a las instituciones y personas que colaboraron significativamente en la realización de este trabajo.

Universidad Politécnica Salesiana de Guayaquil, carrera de Ingeniería Electrónica, por haberme acogido en sus aulas.

A mi tutor de proyecto técnico el Ing. Byron Lima, MSc. Quien con sus conocimientos supo guiarme hasta la culminación de este trabajo de tesis, que gracias a sus acertadas sugerencias y su tiempo dedicado permitió llegar a feliz término esta meta de ser un profesional.

A todas aquellas personas, amigos y docentes que de una u otra manera fueron partícipes de este logro por su colaboración prestada.

A mi amigo y compañero Juan Carlos Guevara por brindarme su amistad y por su gran aporte para lograr esta meta.

A todos ustedes estoy muy agradecido.

Ricardo Andrés Tenorio Vélez

RESUMEN

AÑO	ALUMNOS	DIRECTOR DE PROYECTO TÉCNICO	TEMA DE PROYECTO TÉCNICO
2019	JUAN CARLOS GUEVARA DAVALOS RICARDO ANDRÉS TENORIO VÉLEZ	ING. BYRON XAVIER LIMA CEDILLO, MSc.	DISEÑO E IMPLEMENTACIÓN DE DOS ROBOTS SOCCER AUTÓNOMOS (MIROSOT) PARA LA CARRERA DE ELECTRÓNICA Y AUTOMATIZACIÓN

El presente proyecto técnico tiene como objetivo la implementación de dos robots futbol autónomos, mediante un sistema embebido lattepanda utilizando el procesamiento de imagen mediante una cámara y un software de instrumentación virtual labview acompañado del complemento visión y movimiento.

El objetivo principal es elaborar un algoritmo relacionado en la detección de color para identificar los objetos móviles dentro del perímetro de juego y finalmente se procede a verificar el desempeño del sistema y se obtendrá algunos resultados de sensibilidad del sistema con cambios de luminosidad, por lo cual se procederá a probar dos ambientes con diferentes condiciones de luz, y también analizar la exactitud de la posición del objeto (la pelota de juego) de interés suministrado por el sistema, evaluando el desempeño con y sin distorsión radial.

Adicionalmente se da la facilidad de un entorno de programación de los embebidos Lattepanda usando un núcleo basado en el ATMEL 32UA para el sistema de control a implementar.

Palabras claves: LATTEPANDA, sistemas embebidos, ATMEL, sistemas de control, controles controladores.

ABSTRACT

YEAR	STUDENTS	DIRECTOR OF TECHNICAL PROJECT	TECHNICAL PROJECT THEME
2019	JUAN CARLOS GUEVARA DAVALOS RICARDO ANDRÉS TENORIO VÉLEZ	ING. BYRON XAVIER LIMA CEDILLO, MSc.	DISEÑO E IMPLEMENTACIÓN DE DOS ROBOTS SOCCER AUTÓNOMOS (MIROSOT) PARA LA CARRERA DE ELECTRÓNICA Y AUTOMATIZACIÓN

This technical project is aimed at the implementation of two robot football autonomous, using a lattepanda using image processing and virtual instrumentation accompanied by complement labview software vision and movement.

The main objective is to develop an algorithm related color detection to identify moving objects within the perimeter of the game and finally proceed to verification the performance of the system and you will get some results of sensitivity of the system with changes in luminosity.

Proceed to test two environments with different light conditions and to analyze the accuracy of the position of the object of interest provided by the system, evaluating the performance with and without radial distortion.

Additionally, given the ease of a programming environment of the Lattepanda using a kernel based on the ATMEL for the control system to implement.

Key words: Lattepanda, embedded systems, drivers.

ÍNDICE GENERAL

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA	I
CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR	II
CERTIFICADO DE DIRECCIÓN DE TRABAJO DE TITULACIÓN.....	III
DEDICATORIA	IV
DEDICATORIA	V
AGRADECIMIENTO	VI
AGRADECIMIENTO	VII
RESUMEN.....	VIII
ABSTRACT	IX
ÍNDICE GENERAL.....	X
ÍNDICE DE FIGURAS	XII
INTRODUCCIÓN	1
1. EL PROBLEMA	2
1.1. Antecedentes.....	2
1.2. Importancia y Alcances.....	2
1.3. Delimitación	2
1.3.1. Temporal	2
1.3.2. Espacial.....	2
1.3.3. Académica	3
1.4. Innovación.....	3
1.5. Objetivos.....	3
1.5.1. Objetivo general.....	3
1.5.2. Objetivos específicos	3
2. FUNDAMENTOS TEÓRICOS.....	4
2.1. Robot Autónomo mirosot.....	4
2.2. Robocup	4
2.3. Robotsoft Fira2005	4
2.4. Robochallenge Romania -Futbol.....	5
2.5. Fútbol Robótico – Componente Inteligente	5
2.6. Lattepanda	6
2.7. Atmel 32UA.....	7
2.8. Software de instrumentación virtual Labview	7
2.9. Controlador de Motor (Sabertooth 2x12).....	7
2.9.1. Modo Entrada Serial.....	8

2.10. Tracción Omnidireccional	8
2.11. Procesamiento de imagen	9
3. MARCO METODOLÓGICO	10
3.1. Diseño del robot mirosot.....	10
3.2. Diagrama de conexiones	12
3.2.1. Etapa de Alimentación	13
3.2.2. Etapa de Adquisición de imagen.....	13
3.2.3. Etapa de Procesamiento	14
3.3. Funcionamiento del robot en secuencias de inicio.....	14
3.3.1. Etapa de potencia.....	15
3.4. Algoritmos de control Omnidireccional.....	16
3.5. Diseño Electrónico	18
3.6. Diagrama de conexiones Entradas, salidas Digitales	18
3.7. Software de Reconocimiento	18
4. RESULTADOS	19
4.1. Autonomía del robot	19
4.2. Funcionamiento del robot en pista	19
4.3. Funcionamiento del robot en secuencia frente al arco	22
4.4. Respuesta del procesamiento de imagen.....	23
4.5. Funcionamiento del robot recolectando balones	24
4.6. Funcionamiento del robot Competencia.....	26
5. CONCLUSIONES	27
6. RECOMENDACIONES	28
7. REFERENCIAS BIBLIOGRAFICAS.....	29
ANEXOS.....	31
Anexo 1 Librería Sabertooth [19].....	31
Anexo 2 Control Omnidireccional	33
Anexo 3 Diagrama de control Labview	36
Anexo 4 Especificaciones Motor Fingertech table rpm vs aplicación de voltaje	43
Anexo 5 Especificaciones Motor Fingertech table torque vs aplicación de voltaje...	44
Anexo 6 diseño del prototipo.....	45
Anexo 7 Movimientos del robot.....	49

ÍNDICE DE FIGURAS

Figura 1: MIROSOT Robo Cup [2]	4
Figura 2: Campeonato de Futbol internacional de robótica-FIRA [3]	5
Figura 3: Robot challenge-Romania [4].....	5
Figura 4: Robot Turtle [5].....	5
Figura 5: Lattepanda 4GB RAM -64GB SDD [7].....	6
Figura 6: Atmel 32 ua [8].....	7
Figura 7: Controladora de Motor Sabertooth [12]	7
Figura 8: Modo Entrada Serial Velocidades [12].....	8
Figura 9: Modo Maestro Esclavo [12]	8
Figura 10: Tracción omnidireccional 4 ruedas [13].....	9
Figura 11: Vision and Motion de Labview [14].....	9
Figura 12: Robot Futbolista	10
Figura 13: Robot Futbolista Nivel Inferior	11
Figura 14: Robot Futbolista Nivel Medio	11
Figura 15: Robot Futbolista Nivel Superior	12
Figura 16: Diagrama De Control.....	12
Figura 17: Módulo De Protección	13
Figura 18: Etapa De Adquisición	13
Figura 19: Software De Instrumentación Virtual Labview [17].....	14
Figura 20: Interfaz De Control Mirost.....	14
Figura 21: Etapa De Potencia	16
Figura 22: Diagramas De Flujo Para El Movimiento Del Robot.....	17
Figura 24: Conexión a los periféricos.....	18
Figura 25: Detección de área de juego.....	19
Figura 26: Área oscura referencia a ir al área blanca	19
Figura 27: Área Oscura Secuencia 1	20
Figura 28: Área Oscura Secuencia 2	20
Figura 29: Área Blanca Secuencia 3.....	21
Figura 31: Área Negra Centro	22
Figura 32: Área Blanca Centro.....	22
Figura 33: Área Blanca Secuencia Arco.....	23
Figura 34: Área Oscura Secuencia Arco	23
Figura 35: Ejecución del programa en competencia área oscura	24
Figura 36: ejecución del programa en competencia área blanca	24
Figura 37: Área Oscura Recolección De Balones.....	25
Figura 38: Área Blanca Recolección De Balones	25
Figura 39: Participación Robochallenge Bucarest Rumania	26

INTRODUCCIÓN

Actualmente los concursos de robótica a nivel mundial estimulan el desarrollo y la investigación de nuevas formas de control usando herramientas de software libre en el campo de la visión artificial, además del procesamiento necesario usando sistemas embebidos.

El presente proyecto está enfocado al club de robótica de la carrera de Ingeniería Electrónica con mención en sistemas industriales de la universidad Politécnica Salesiana sede Guayaquil para que de esta manera obtengan una rutina para el entendimiento de nuevas tecnológicas y procesos de control aplicados a la par con la visión artificial en los cuales se utilice un sistema de tracción omnidireccional para el movimiento del robot.

La plataforma para la programación será el software de instrumentación virtual labview para la adquisición de imagen a la par con el ide de Arduino teniendo como meta que el robot anote la mayor cantidad de goles en un tiempo determinado.

1. EL PROBLEMA

1.1. Antecedentes

Actualmente los concursos de robótica van adquiriendo un aumento en el nivel de las tecnologías implementadas en los prototipos en específico la categoría de robot futbolista. Los estudiantes que conforman el club de robótica se ven en la necesidad de ir adquiriendo nuevos conocimientos para realizar una participación optima representando al país y la institución en concursos internacionales.

1.2. Importancia y Alcances

El presente trabajo contribuye al desarrollo de conocimientos referidos a Vision artificial utilizados en la categoría futbolistas (mirosot).

Con el presente proyecto el estudiante puede desarrollar rutinas para el aprendizaje de sistemas control aplicados en el software de implementación virtual y sistemas embebidos.

1.3. Delimitación

1.3.1. Temporal

La implementación de este proyecto se realizo en un intervalo de un año a partir de la aprobación del mismo.

1.3.2. Espacial

Este proyecto será llevado a cabo en la Universidad Politécnica Salesiana Sede Guayaquil en el bloque D (Club de Robótica).

1.3.3. Académica

La implementación del presente trabajo está orientado al desarrollo de robot autónomos de manera que inciten al estudio de sistemas de control e instrumentación virtual.

1.4. Innovación

Para la implementación del proyecto se empleó nuevas tecnologías de código libre y visión artificial la cuales poseen un nivel de procesamiento altos teniendo como núcleo un sistema embebido superando a sistemas convencionales basados en controladores de 8 bits

1.5. Objetivos

1.5.1. Objetivo general

- Diseñar e Implementar dos robots soccer autónomos (Mirobot) para el club de robótica.

1.5.2. Objetivos específicos

- Construir robots con respecto a las medidas reglamentarias en competencias.
- Implementar el proceso del software de autonomía que se aplicara a cada uno de los Robots-Soccer.
- Realizar un algoritmo capaz de detectar el color del prototipo para su diferenciación.
- Realizar pruebas mediante el análisis de sensibilidad de luminosidad referente al medio que se encuentre.
- Ejecutar un análisis del funcionamiento del robot para comprobar el mejoramiento de la calidad de su operatividad.

2. FUNDAMENTOS TEÓRICOS

2.1. Robot Autónomo mirosot

El objetivo principal es elaborar un algoritmo relacionado en la detección de color para identificar los objetos móviles dentro del perímetro de juego y finalmente se procede a ejecutar una rutina para atrapar los balones y anotar la mayor cantidad de goles en la portería contraria.

2.2. Robocup

El nombre Robo Cup es una contracción del nombre completo de la competencia, "Robot Soccer Word Cup " fue precedida por el primer Torneo Internacional de Micro Copa Mundial de Robots (MIROSOT) realizado por KAIST en Taejeon , Corea , en noviembre de 1996 [1].

El sistema de fútbol de robot consta de jugadores robot, transmisores de RF, cámaras de visión artificial y computadora de juegos véase Figura 1.

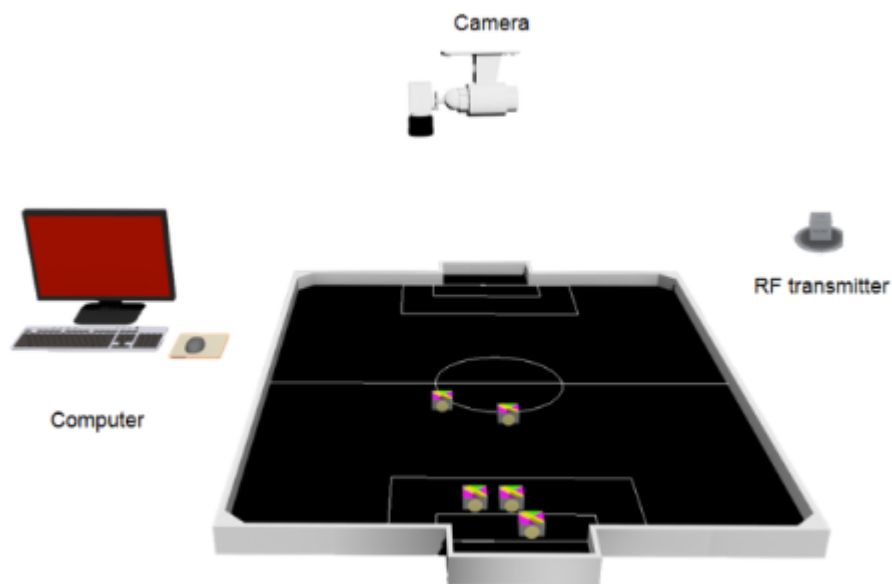


Figura 1: MIROSOT Robo Cup [2]

2.3. Robotsoft Fira2005

Robotsoft es una competencia donde dos equipos se rigen a las normativas de la Fira(Federación internacional de robot soccer asociación) en la cual se cumple con las dimensiones 7,5 centímetros cuadrados, el cual es teledirigido por un computador central encargado del procesamiento y toma de decisiones como se muestra en la Figura 2.

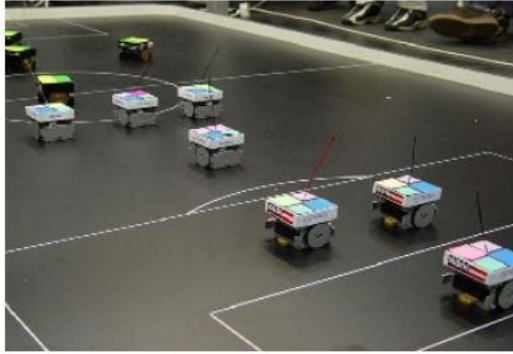


Figura 2: Campeonato de Futbol internacional de robótica-FIRA [3]

2.4. Robochallenge Romania -Futbol

Robochallenge es una competencia la cual posee una categoría llamada Futbol donde dos equipos se rigen a las normativas las cuales cumple con las dimensiones 25 centímetros cuadrados, el cual deberá tener un sistema netamente autónomo para encontrar los balones en el área de juego y anotar en el campo contrario El objetivo de esto es el de que más puntos realice gana Figura 3.



Figura 3: Robot challenge-Romania [4]

2.5. Futbol Robótico – Componente Inteligente

Este prototipo consta de algoritmos y diagramas de estados para la obtención del balón, evitar obstáculos, para no quedar atrapados en los bordes y un algoritmo para definir los movimientos del arquero. Este prototipo de la Figura 4 represento al país en Corea 2002 en un concurso organizado por la Fera [5].

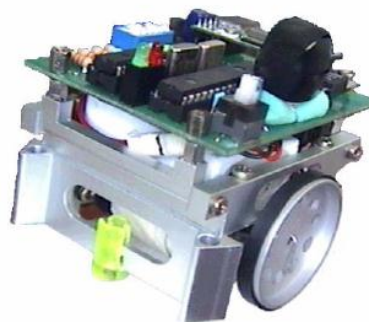


Figura 4: Robot Turtle [5]

2.6. Lattepanda

Lattepanda es un embebido que consta con un procesador Cherry Trail Atom y un microprocesador atmega32ua con puertos de salidas GPIO.

Lattepanda posee edición completa de Windows 10, que incluye potentes herramientas como Visual Studio, NodeJS, Java, producción, y mucho más. Utilización de las API existentes, puede desarrollar sus propios proyectos de software y hardware en Lattepanda como lo haría en un PC normal – C #, JavaScript, Ruby y así sucesivamente [6].

En la figura 5 se aprecia al embebido lattepanda y sus características:

- Intel Cherry Trail Z8350 Quad Core Processor
- Frecuencia del procesador: 1.44GHz
- Operating System: Windows 10 Enterprise LTSC x64 with activate key
- RAM: 4GB DDR3L
- Storage Capacity: 64GB
- GPU: Intel HD Graphics, 12 EUs @200-500Mhz, single-channel memory
- USB 3.0 x 1, USB 2.0 x 2,Wi-Fi 802.11n 2.4G,Bluetooth 4.0
- Integrated Arduino Co-processor: ATmega32u4 (Arduino Leonardo)
- Video output: HDMI and MIPI-DSI
- Supports 100Mbps Ethernet
- Intel Processor GPIO x 6, ATmega Processor GPIO x 20
- Gravity Interface Connectors x 6
- Voltage: 5V@2A
- RoHS, FCC and CE Compliant

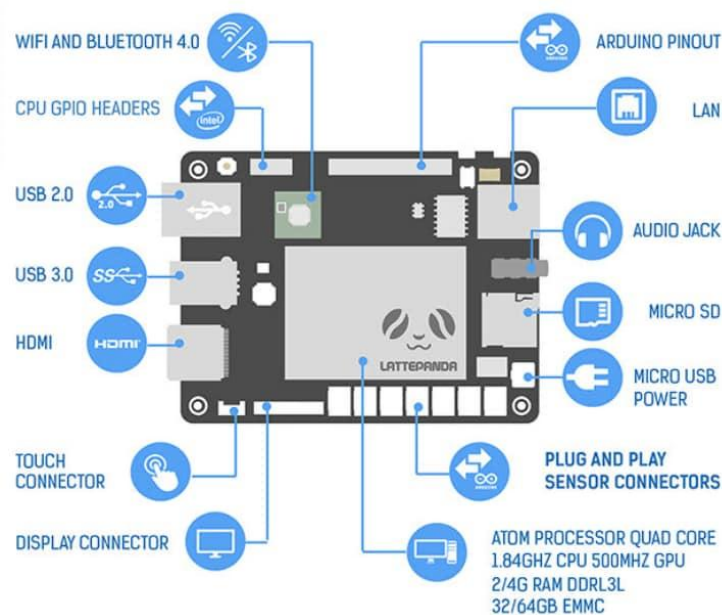


Figura 5: Lattepanda 4GB RAM -64GB SDD [7]

2.7. Atmel 32UA

El Atmel 32ua es un microcontrolador el cual tiene una arquitectura Harvard, posee 8 bits, 32Kb de memoria Flash y un empaquetado VFQFN-44 [8].

Posee 20 pines de entrada y salida digitales de los cuales 7 pueden ser usados como salidas PWM y 12 como entradas análogas. [9].

En la figura 6 se muestra el Atmel 32ua como elemento de montaje superficial.

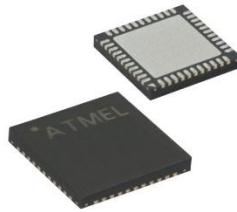


Figura 6: Atmel 32 ua [8]

2.8. Software de instrumentación virtual Labview

Labview es un software de instrumentación virtual el cual trabaja con un lenguaje gráfico, ya que nos permite diseñar interfaces de usuario mediante una consola interactiva basada en software [10].

2.9. Controlador de Motor (Sabertooth 2x12)

La controladora Sabertooth 2x12 (Figura 7) es fabricada por la empresa Dimensión Engineering y es uno de los controladores de motores duales más versátiles, eficientes y fáciles de usar, esta tarjeta ofrece control de dos motores Dc brushed (con escobillas), con un máximo de 12 Amperios por cada motor soportando picos de 24 Amperios por cada motor [11].

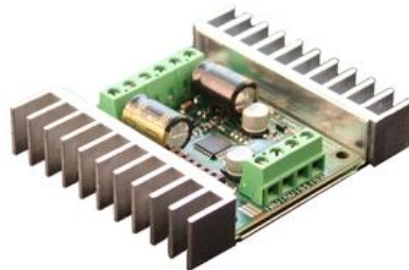


Figura 7: Controladora de Motor Sabertooth [12]

2.9.1. Modo Entrada Serial

El modo en serie utiliza datos en serie de nivel TTL RS-232 para establecer la velocidad y la dirección del motor. Esto se usa para conectar el Sabertooth a una PC o microcontrolador. Si usa una PC, un nivel convertidor como un chip MAX232 debe ser utilizado. Un convertidor en serie USB-a-TTL también es una opción para la PC [12]. La velocidad en baudios se establece mediante interruptores DIP Figura 8.

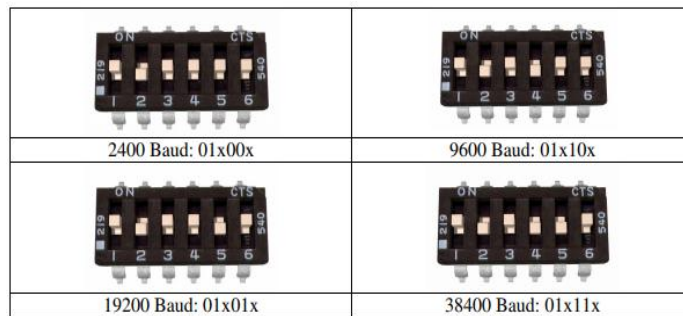


Figura 8: Modo Entrada Serial Velocidades [12]

Los comandos son de un solo byte. Ahí está también un modo de selección de esclavo que permite el uso de varios Sabertooth 2x12 véase en Figura 9 desde un único puerto serie del microcontrolador [12].

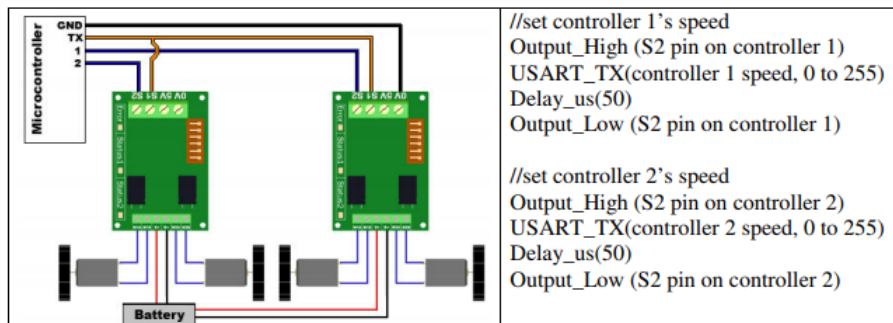


Figura 9: Modo Maestro Esclavo [12]

2.10. Tracción Omnidireccional

La tracción omnidireccional consta de una máxima maniobrabilidad ya que permite moverse en cualquier dirección sin necesidad de rotar o reorientarse, a diferencia de otros sistemas tradicionales de dos ruedas [13].

Este tipo de tracción se puede desarrollar usando tres, cuatro o más ruedas omnidireccionales como se muestra en la Figura 10.

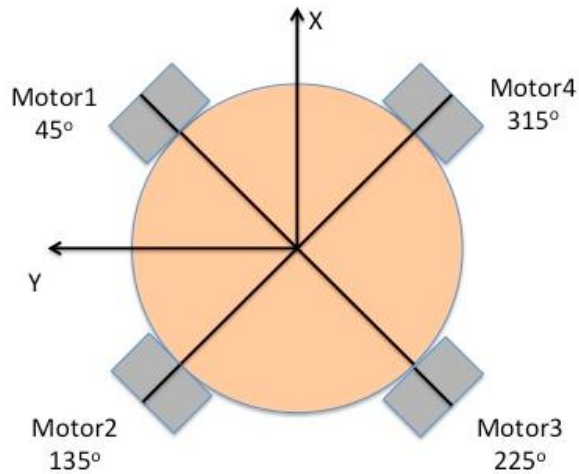


Figura 10: Tracción omnidireccional 4 ruedas [13]

2.11. Procesamiento de imagen

La implementación de sistemas Vision de Labview permite el procesamiento para el análisis de imágenes mediante periféricos al mundo exterior mediante el uso de la cámara de video.

En la figura 11 se muestra las herramientas que posee este complemento de labview.

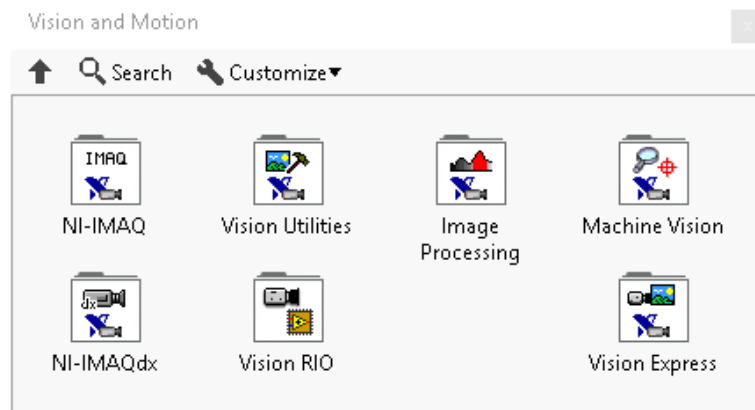


Figura 11: Vision and Motion de Labview [14]

3. MARCO METODOLÓGICO

3.1. Diseño del robot mirosot

El robot futbol está diseñado mediante un núcleo lattepanda usando un sistema de locomoción omnidireccional de 4 ruedas con una parte de potencia que constan con 2 driver de motor Sabertooth.

El software de implementación para el procesamiento de imagen y procesamiento de los comandos es Labview a la par con un atmega 32ua usado como tarjeta adquisidora de datos.

La cámara para la adquisición de video posee una conexión física mediante un USB a la lattepanda.

El diseño del robot mirosot o robot futbolista se realizó en el software de diseño 3d Solidword véase Figura 12 permitiendo una etapa de desarrollo optima por sus funciones para configurar el peso de los diferentes elementos(revisar Anexo 6).

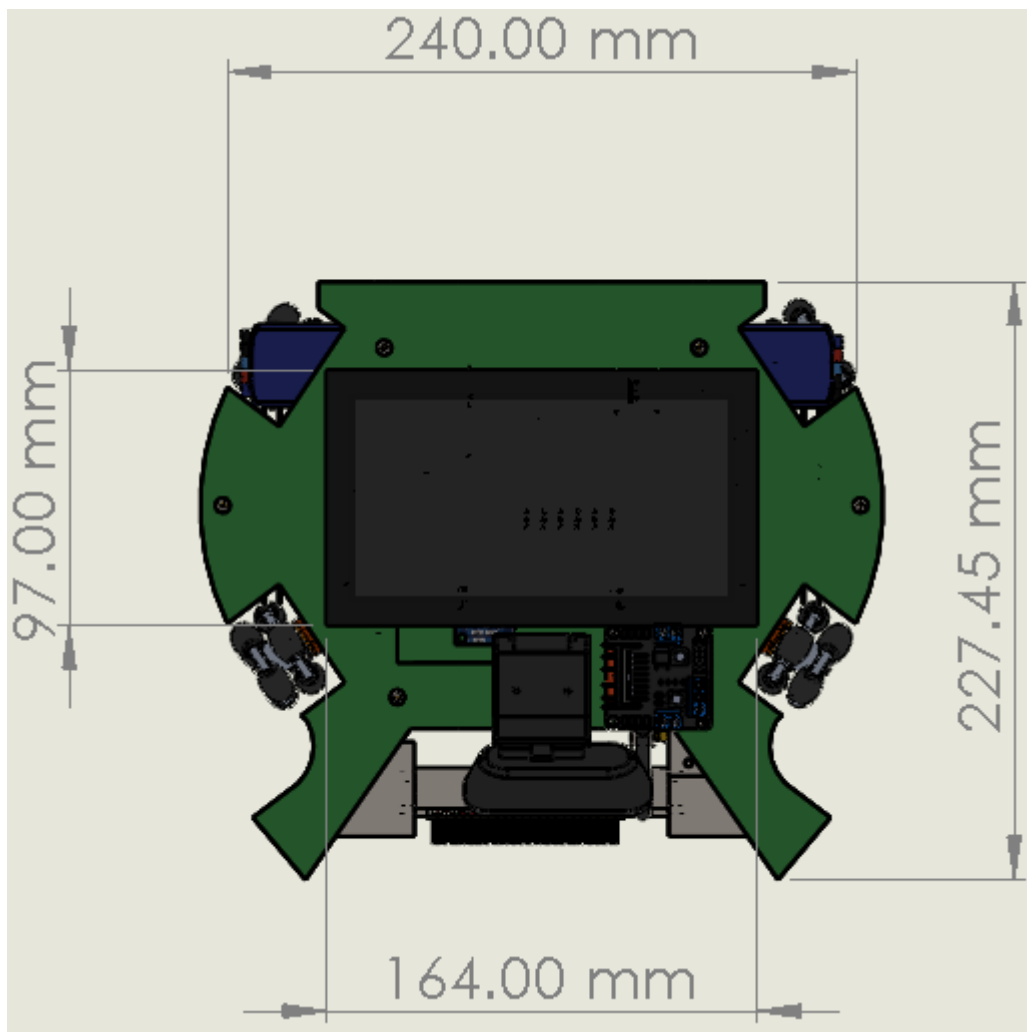


Figura 12: Robot Futbolista

Los prototipos poseen 3 niveles donde el nivel inferior que muestra en la Figura 13 será la de los actuadores acoplados a las llantas omnidireccionales.

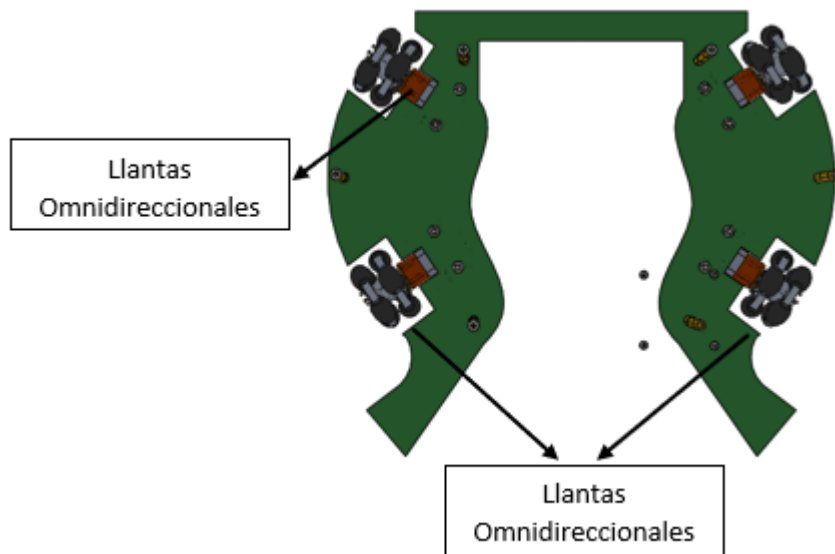


Figura 13: Robot Futbolista Nivel Inferior

En el nivel medio constara con el Core o lattepanda, la etapa de potencia, baterías y la etapa de regularización de voltajes también con el actuador que sirve para la recolección de los balones como se detalla en la Figura 14.

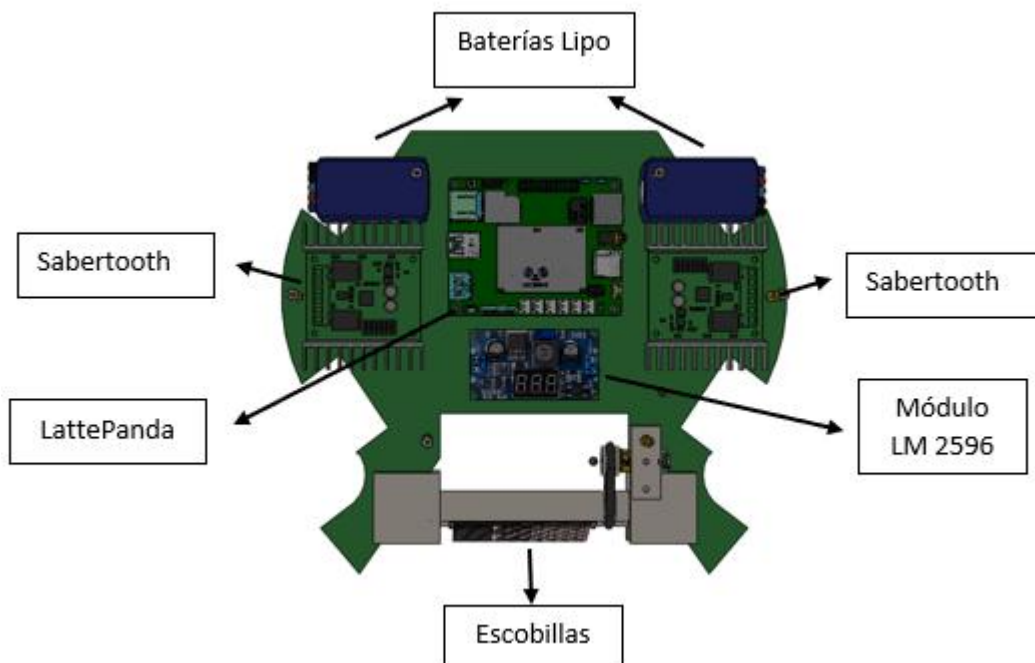


Figura 14: Robot Futbolista Nivel Medio

En el nivel superior posee la interfaz de usuario o pantalla táctil donde se comandará la configuración y el inicio del robot, también la cámara que sirve para la etapa de adquisición de imagen y el controlador de potencia del actuador como se visualiza en la Figura 15.

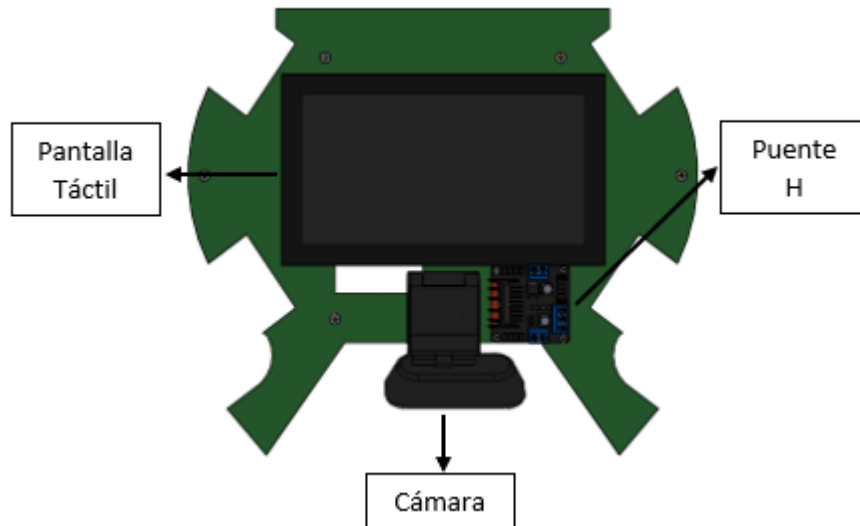


Figura 15: Robot Futbolista Nivel Superior

3.2. Diagrama de conexiones

El diagrama de conexiones del robot consta de 4 etapas para el óptimo funcionamiento del robot tales como etapa de alimentación, protección, adquisición, procesamiento y potencia como se muestra en la Figura 16.

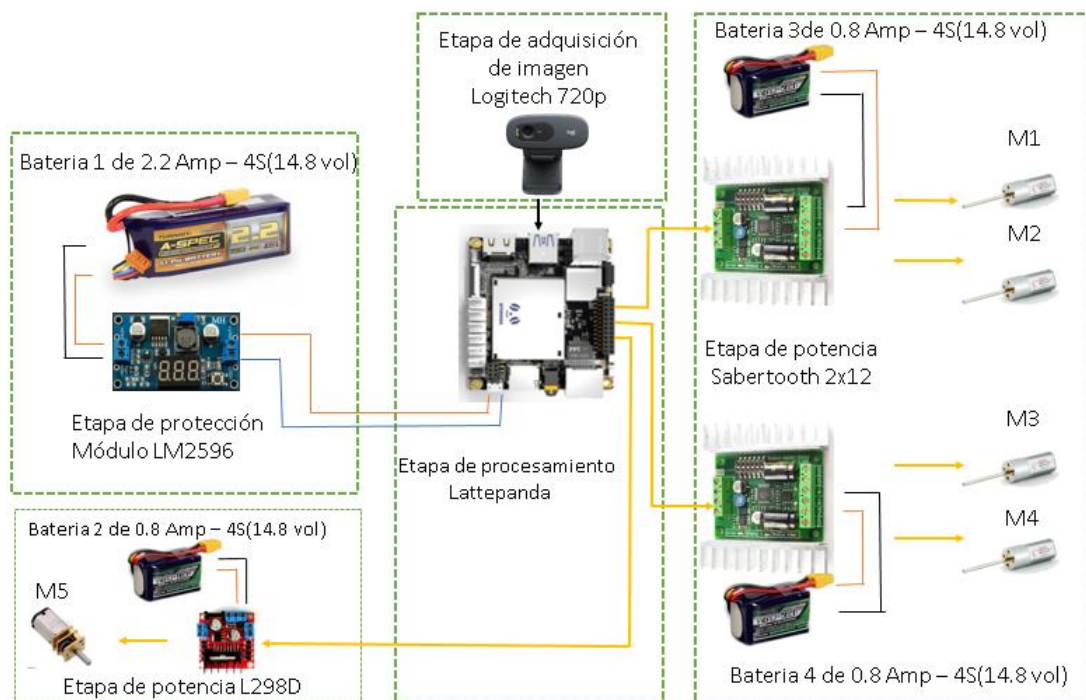


Figura 16: Diagrama De Control

3.2.1. Etapa de Alimentación

La etapa de alimentación está compuesta por 3 baterías de litio y polímero denominadas Lipo, 2 de estas poseen un amperaje de 850 miliamperios para la etapa de potencia del robot y una de 2200 miliamperios para la etapa de procesamiento la cual tendrá una protección regulando el voltaje a 5VDCM mediante un módulo LM2596(regulador de voltaje) que admite una entrada de 0 ~ 40V DC con una precisión de $\pm 0.05V$ [15] como se muestra en la figura 17

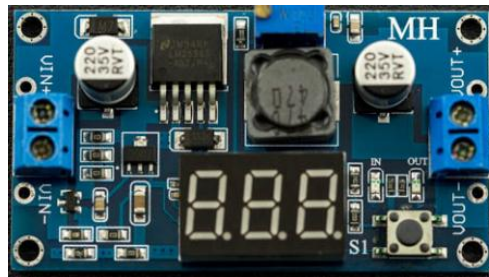


Figura 17: Módulo De Protección

3.2.2. Etapa de Adquisición de imagen

La etapa de imagen consta con una cámara C270 HD la cual consta de una corrección autónoma de luz para colores brillantes con 30 cuadros por segundo y una resolución de 720p con un campo visual de 60 grados como se muestra en la Figura 18. [16]



Figura 188: Etapa De Adquisición

3.2.3. Etapa de Procesamiento

La etapa de procesamiento está compuesta por la lattepanda en la que se realiza el computo usando el software de instrumentación virtual LABVIEW con la ayuda de la herramienta visión y movimiento [17] como se muestra en la Figura 19.



Figura 199: Software De Instrumentación Virtual Labview [17]

3.3. Funcionamiento del robot en secuencias de inicio

Consta con 4 secuencias de inicios las cuales son configurables mediante el panel frontal que consta con 3 pantallas, selectores, botoneras e indicadores de luces en el entorno Labview previamente al funcionamiento de este como se muestra en la Figura 20.

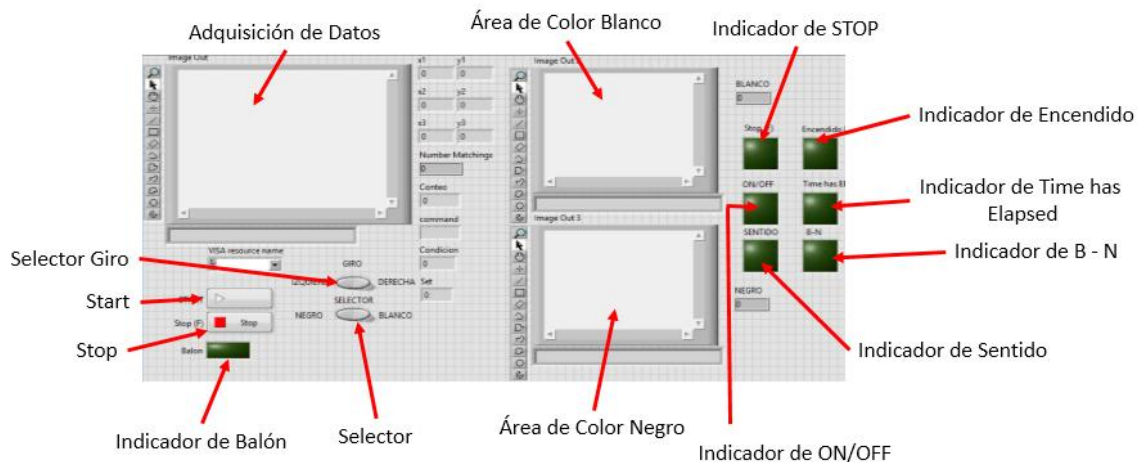


Figura 20: Interfaz De Control Mirosoft

El Cuadro Adquisicion, es donde se muestra la adquisicion de datos del procesamiento del robot.

El Cuadro Blanco, es donde se va a visualizar la adquisicion del arco, en este caso el arco de color blanco.

El Cuadro Negro, es donde se va a visualizar la adquisicion del arco, en este caso el arco de color negro.

El Selector, es para ver a que área va a ir el robot si es al área negra o al área blanca de la cancha.

El Selector Giro, es para ver el lugar donde iniciará el robot si del lado derecho o del lado izquierdo de la cancha.

El botón Start, para inicializar el procesamiento del robot.

El botón Stop, para darle finalizar al procesamiento del robot.

El indicador Balón, cuando esta encendido indica que esta detectando el balón en la cancha.

El indicar Sentido, nos señala en que sentido esta realizando el proceso el robot cuando esta encendido indica que va para la derecha y cuando esta apagado es que va para la izquierda.

El indicador B-N, nos muestra en que lado esta el robot si esta en área blanca o en el área negra de la cancha si esta se encuentra encendido indica que esta en el área blanca y si se encuentra apagado esta en el área negra.

El indicador Stop, como su nombre lo indica es la finalización del procesamiento del robot.

El indicador ON/OFF, cuando esta encendido es cuando el programa de labview esta en funcionamiento pero el procesamiento del robot aun no esta en funcionamiento hasta cuando se le de Start.

El indicador Encendido, señala cuando esta encendido es por que esta en marcha el procesamiento del robot.

El indicador Time has Elapsed, este indicador nos señala un cierto tiempo que el procesamiento esta encendido y se activa el indicador.

3.3.1. Etapa de potencia

La etapa de potencia posee dos tarjetas controladoras de motor Sabertooth [12], en modo de configuración serial referente a la sección 2.9.1, usando como control de los motores de tracción véase en Anexo 4 y 5, con una alimentación de poder de una batería Lipo 0.8Amperios cada controlador como se muestra en la Figura 21.

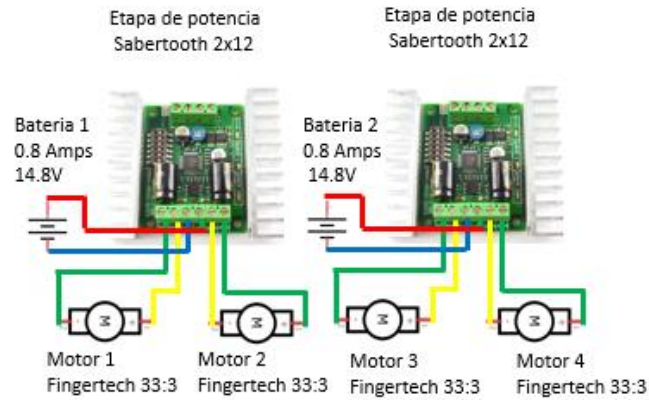


Figura 21: Etapa De Potencia

3.4. Algoritmos de control Omnidireccional

El control consta como lenguaje de programación C por su alto nivel en lo que es la creación de secuencias lógicas a la par con el ide de Arduino.

Los comandos están la sección de Anexo 2 para que el robot realice movimientos es mediante la combinación de rutinas de programación lógica siguiendo el diagrama que se muestra en la imagen 22 realizando el procesamiento de imagen mediante labview y esperando la respuesta mediante dos puertos seriales.

Mediante el procesamiento se obtiene el comando para que los actuare cumpa con las funciones detalladas Anexo 7 las cuales son:

- Adelante
- Atrás
- Desplazamiento Derecha
- Desplazamiento Izquierda
- Giro derecho
- Giro izquierdo
- Recolectar Balón
- Expulsar Balón
- Detener

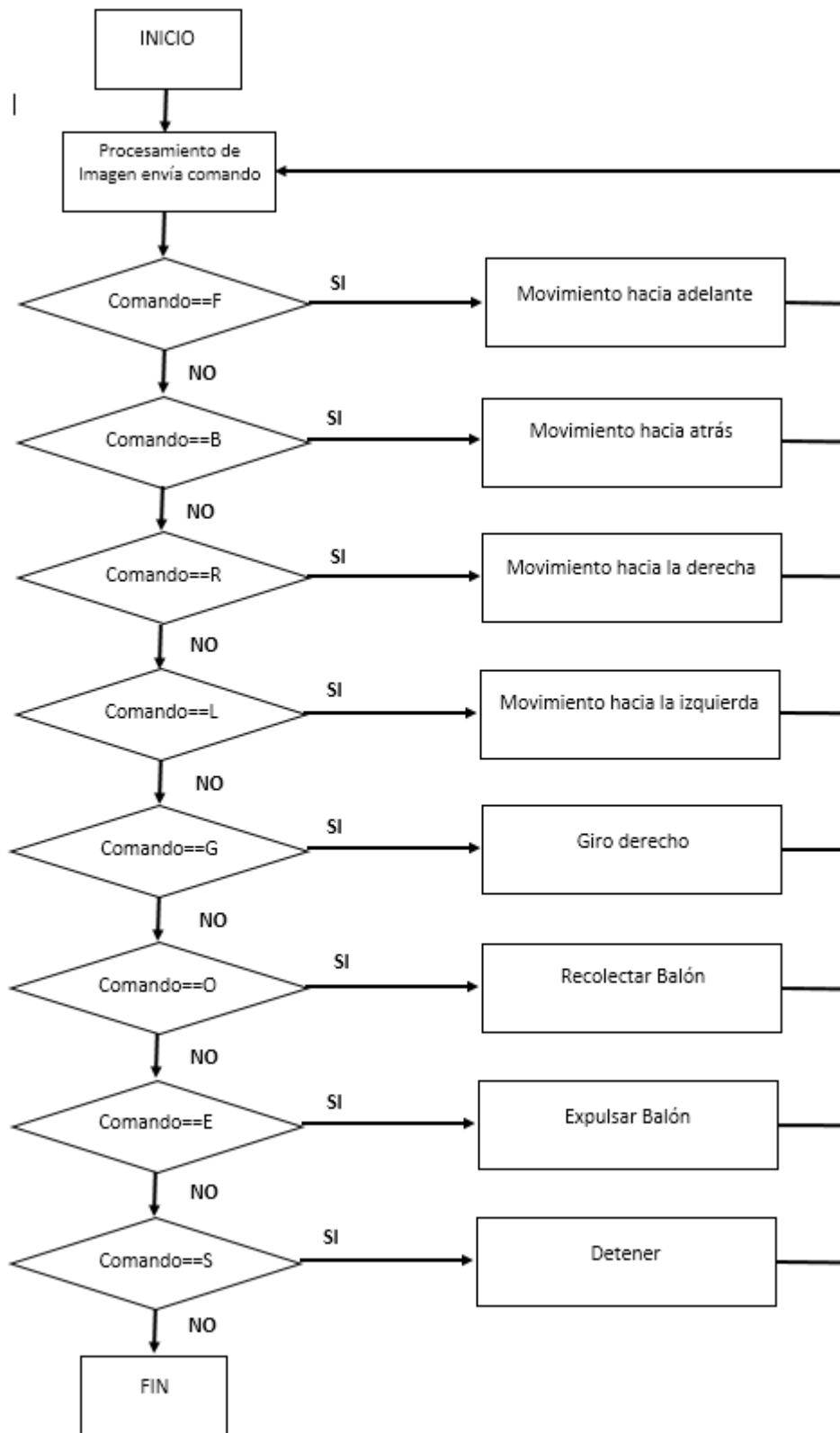


Figura 22: Diagramas De Flujo Para El Movimiento Del Robot

3.5. Diseño Electrónico

La parte del diseño electrónica consta por la lattepanda, los 2 driver de potencia Sabertooth y un driver de potencia para el actuador que recolectara los balones esto se refleja en la Figura 23.

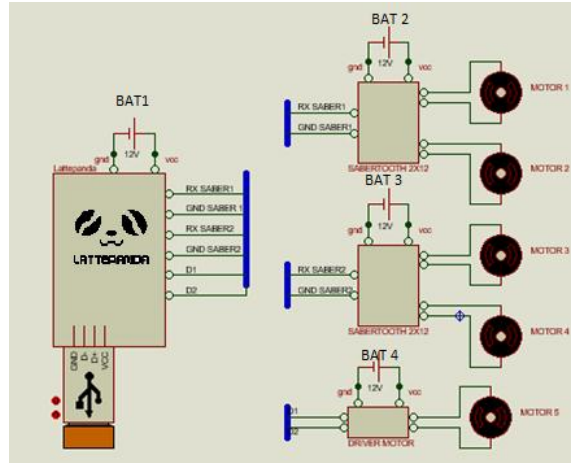


Figura 23: Diseño Electrónico

3.6. Diagrama de conexiones Entradas, salidas Digitales

Las entradas digitales del embebido trabajan a 5V por lo cual se necesita tener driver para controlar y un driver para el acoplamiento de la señal Pwm a un motor [18].

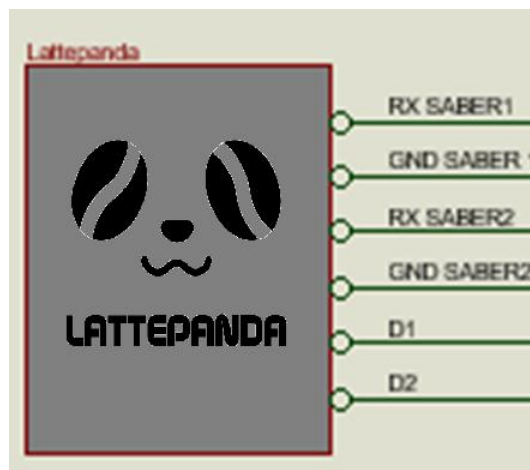


Figura 23: Conexión a los periféricos

3.7. Software de Reconocimiento

Para el reconocimiento se usó labview con la ayuda de la herramienta visión y movimiento.

4. RESULTADOS

4.1. Autonomía del robot

La autonomía del robot es de 5 minutos ya que este tiempo está reglamentado a la categoría es de 3 minutos de esta posee un tiempo prudencial para prevenir el daño a las baterías de Lipo y a los motores de tracción.

4.2. Funcionamiento del robot en pista

El robot consta con un sistema de control para identificar en qué lugar del área de juego esta si esta en área blanca o negra, respetando la referencia buscar ir al área contraria y detectar balones como se muestra en la Figura 25.

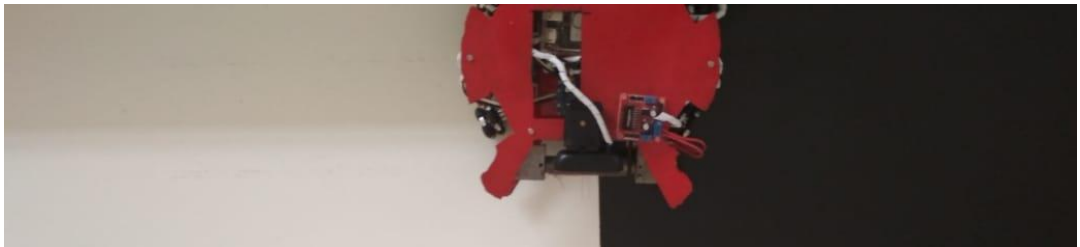


Figura 245: Detección de área de juego

Posteriormente si el robot tiene como referencia la pista oscura busca ir al área blanca como muestra la Figura 26.

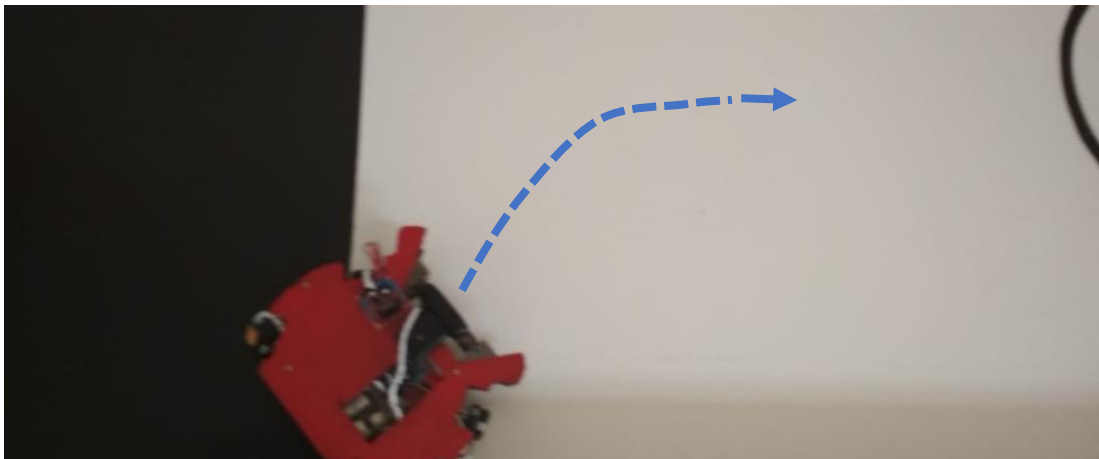


Figura 25: Área oscura referencia a ir al área blanca

La primera secuencia es en la parte negra en el lado izquierdo de esta manera al dar inicio gira a la derecha buscando la mitad del área de juego como se muestra en la Figura 27

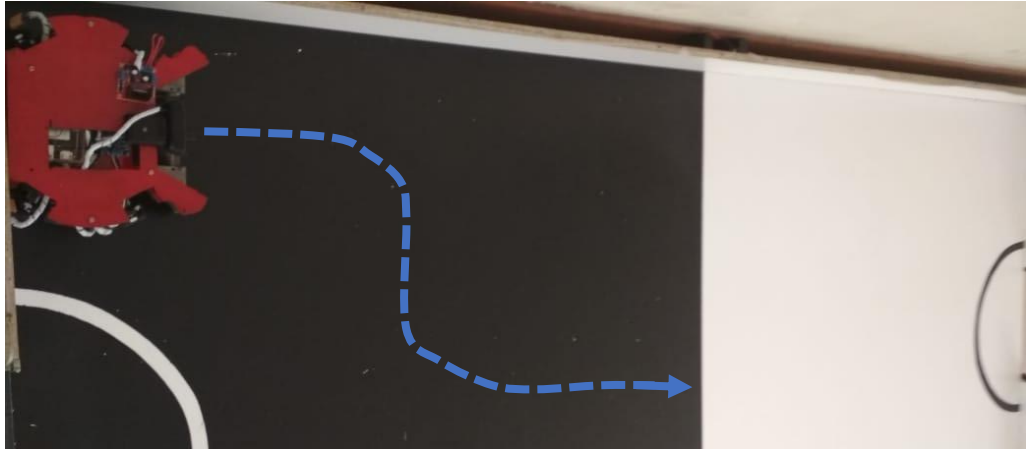


Figura 26: Área Oscura Secuencia 1

En la segunda secuencia iniciara en la parte derecha del área oscura buscando el centro del área de juego como se muestra en la Figura 28.

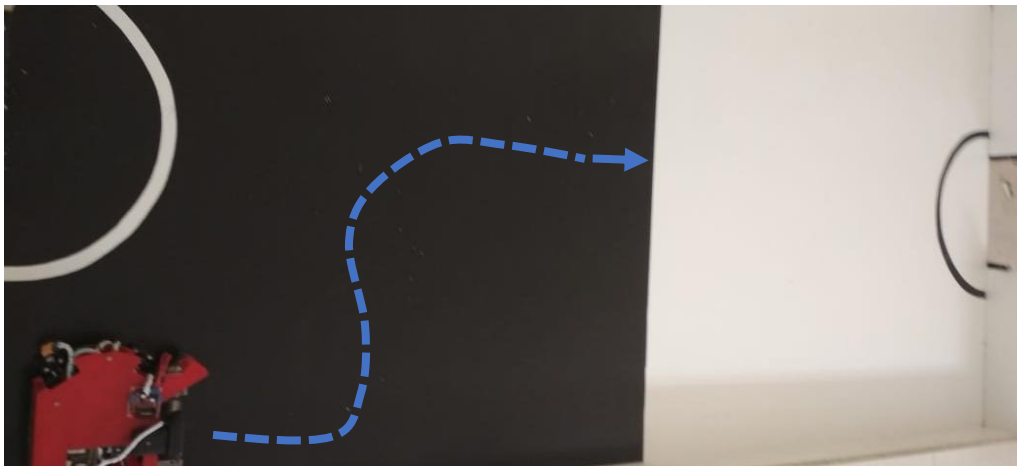


Figura 27: Área Oscura Secuencia 2

La tercera rutina se ejecuta en el área blanca estando de esta manera en la parte derecha de la pista y realizando una secuencia para ir al área central del área de juego como se muestra en la Figura 29.

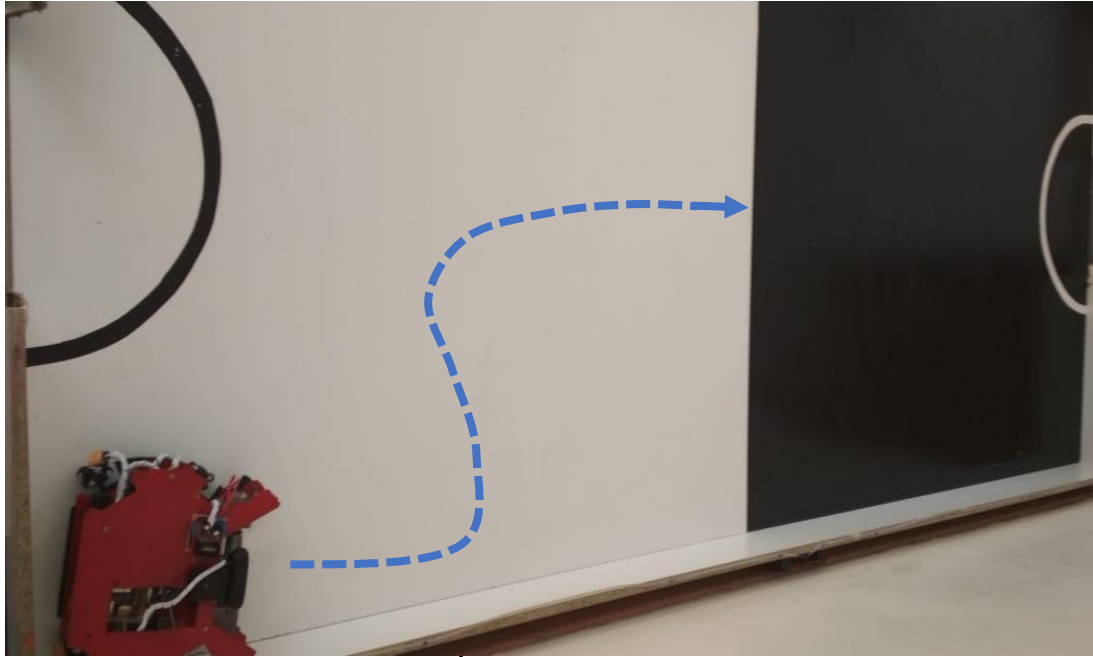


Figura 28: Área Blanca Secuencia 3

La cuarta rutina se realizará en la parte izquierda del área blanca buscando el área central como se muestra en la Figura 30.

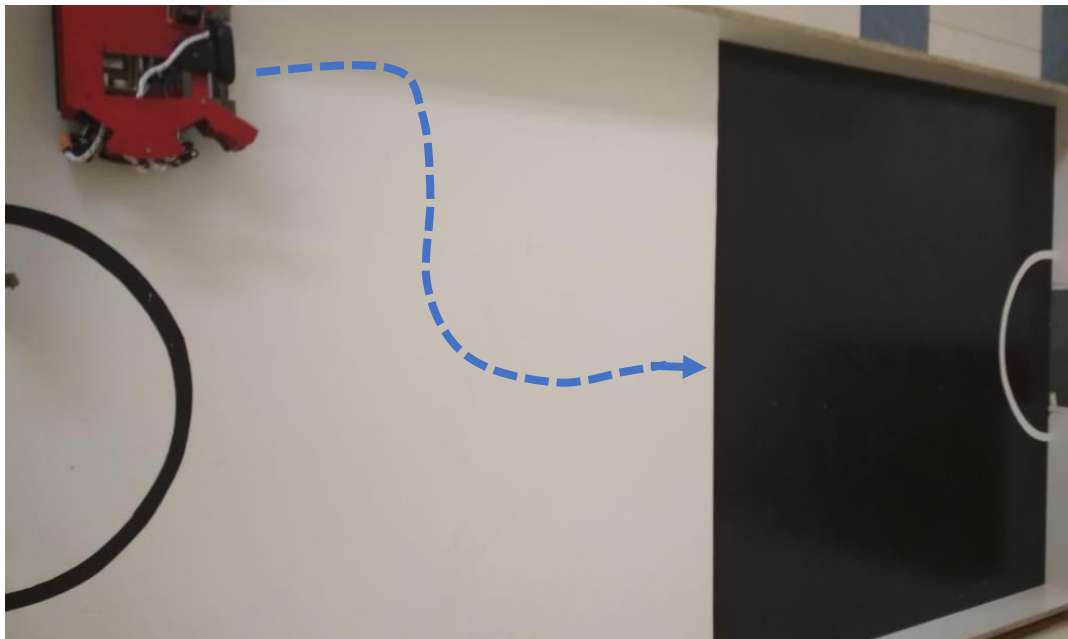


Figura 30: Area blanca secuencia 4

En la Figura 31 se muestra el robot en el área central de la pista luego de usar la secuencia 1 o 2 en el área oscura.

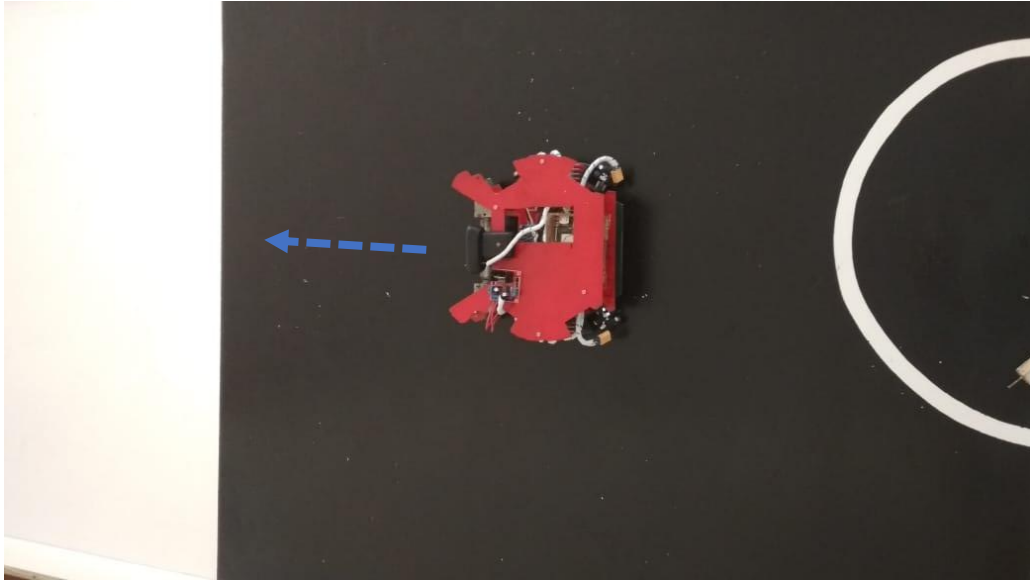


Figura 29: Área Negra Centro

En la Figura 32 se muestra el robot en el área central de la pista luego de usar la secuencia 3 o 4 en el área blanca.

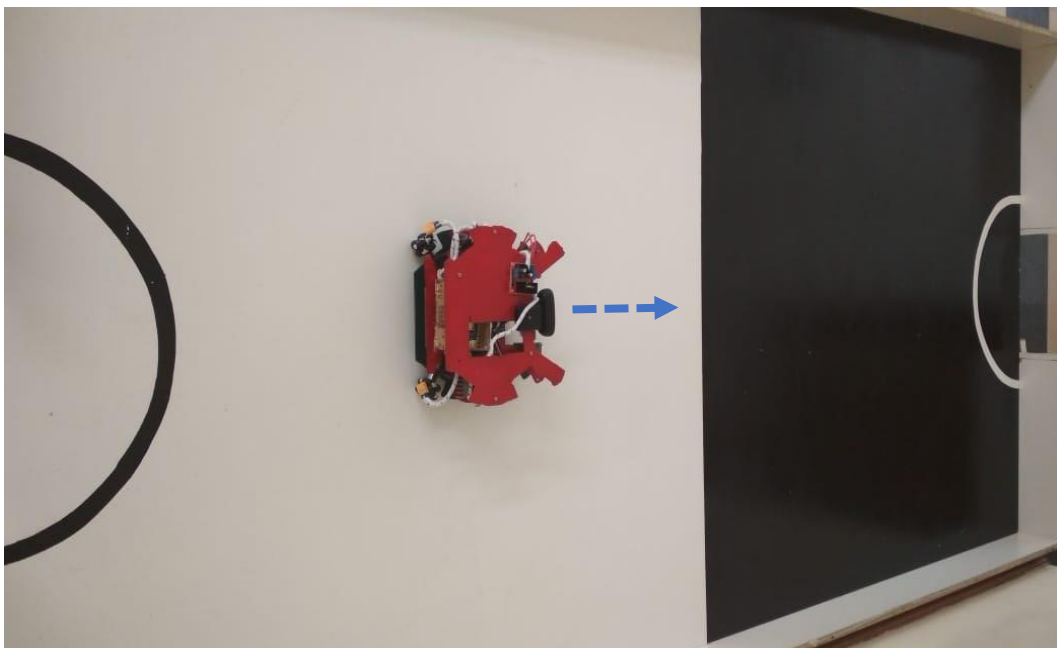


Figura 30: Área Blanca Centro

4.3. Funcionamiento del robot en secuencia frente al arco

En la Figura 33 se muestra el robot en el área blanca ejecutando la rutina donde se encuentre frente al arco contrario de la referencia.

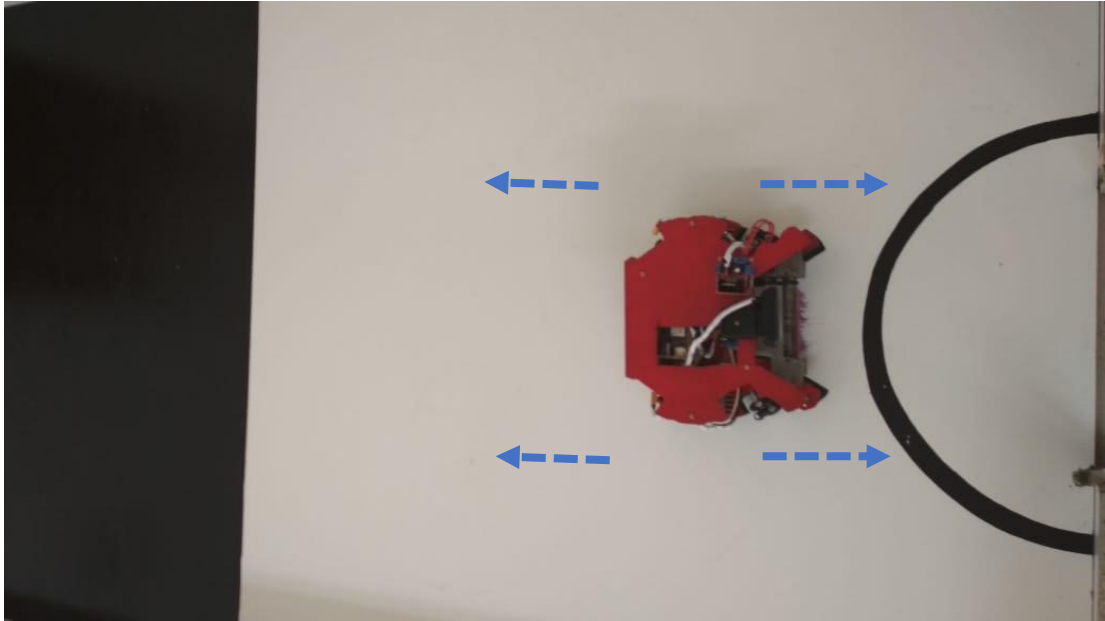


Figura 31: Área Blanca Secuencia Arco

La rutina que se muestra en la Figura 34 se ejecuta cuando el robot se posiciona en el área del arco en la pista negra.

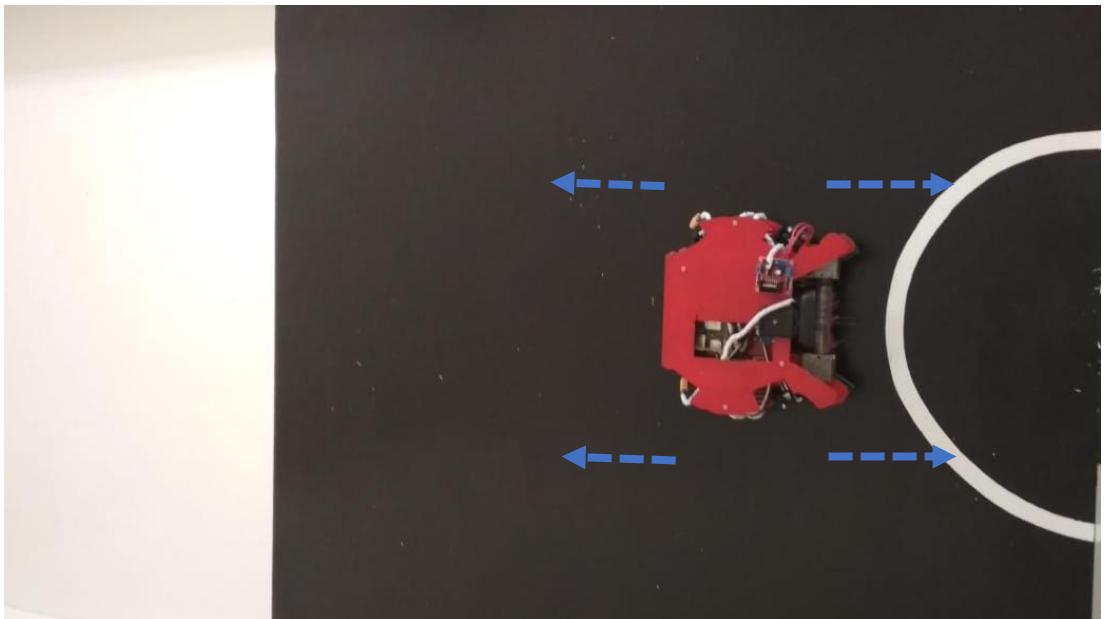


Figura 32: Área Oscura Secuencia Arco

4.4. Respuesta del procesamiento de imagen

La respuesta del procesamiento de imagen se muestra en la Figura 35 y Figura 35 donde se aprecia la ejecución en tiempo real del robot participando en el concurso de robótica Robochallenge realizado en Bucarest, Rumania.



Figura 33: Ejecución del programa en competencia área oscura

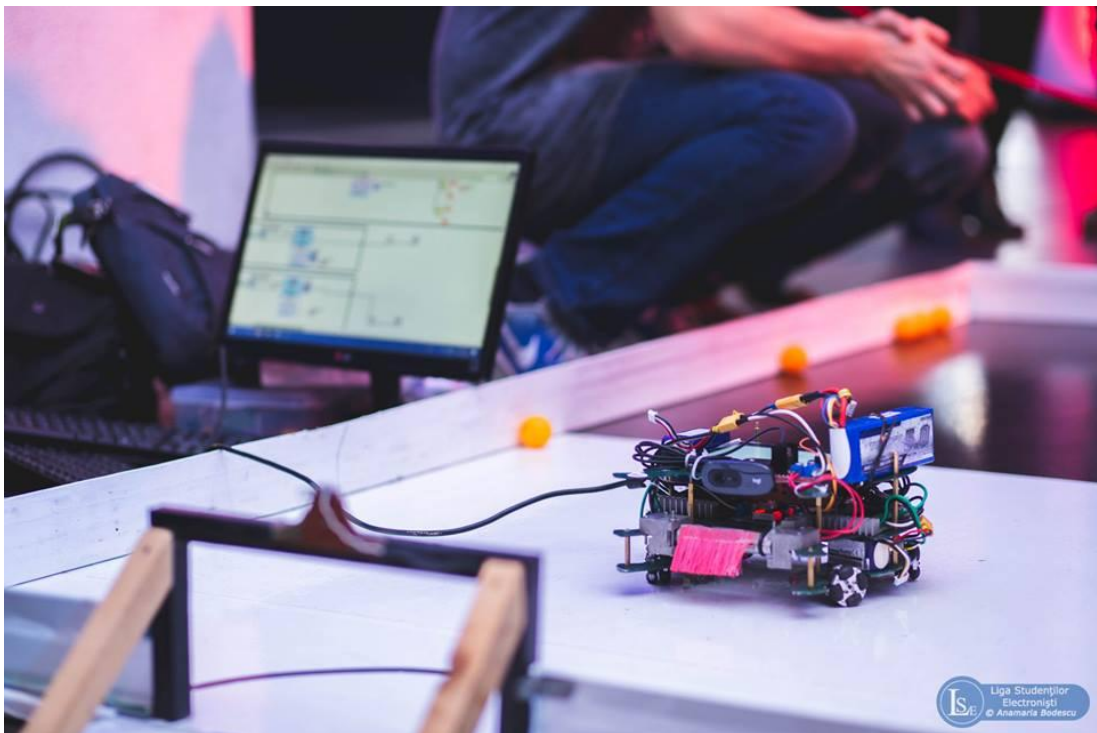


Figura 34: ejecución del programa en competencia área blanca

4.5. Funcionamiento del robot recolectando balones

El recolectado de balones en el área negra se muestra en la Figura 37 donde el robot realiza el procesamiento y las rutinas para su correcto funcionamiento.



Figura 35: Área Oscura Recolección De Balones

El recolectado de balones en el área blanca se muestra en la Figura 38 donde el robot realiza el procesamiento y las rutinas para su correcto funcionamiento.



Figura 36: Área Blanca Recolección De Balones

4.6. Funcionamiento del robot Competencia

En las Figuras 39 se muestran algunas capturas de la participación en la competencia de robótica internacional Robochallenge que se realizó en Rumania.

Los prototipos mirosot llamados “Valeria” y “Ronaldo”, tuvieron una destaca participación dando por alto el nombre de ecuador y de la universidad politécnica salesiana obteniendo un cuarto lugar siendo su primera participación internacionalmente en esta categoría.

La iluminación afecto negativamente al robot ya que eran luces RGB ya que este tipo de luces cambian continuamente su gama de colores en un tiempo aleatorio.

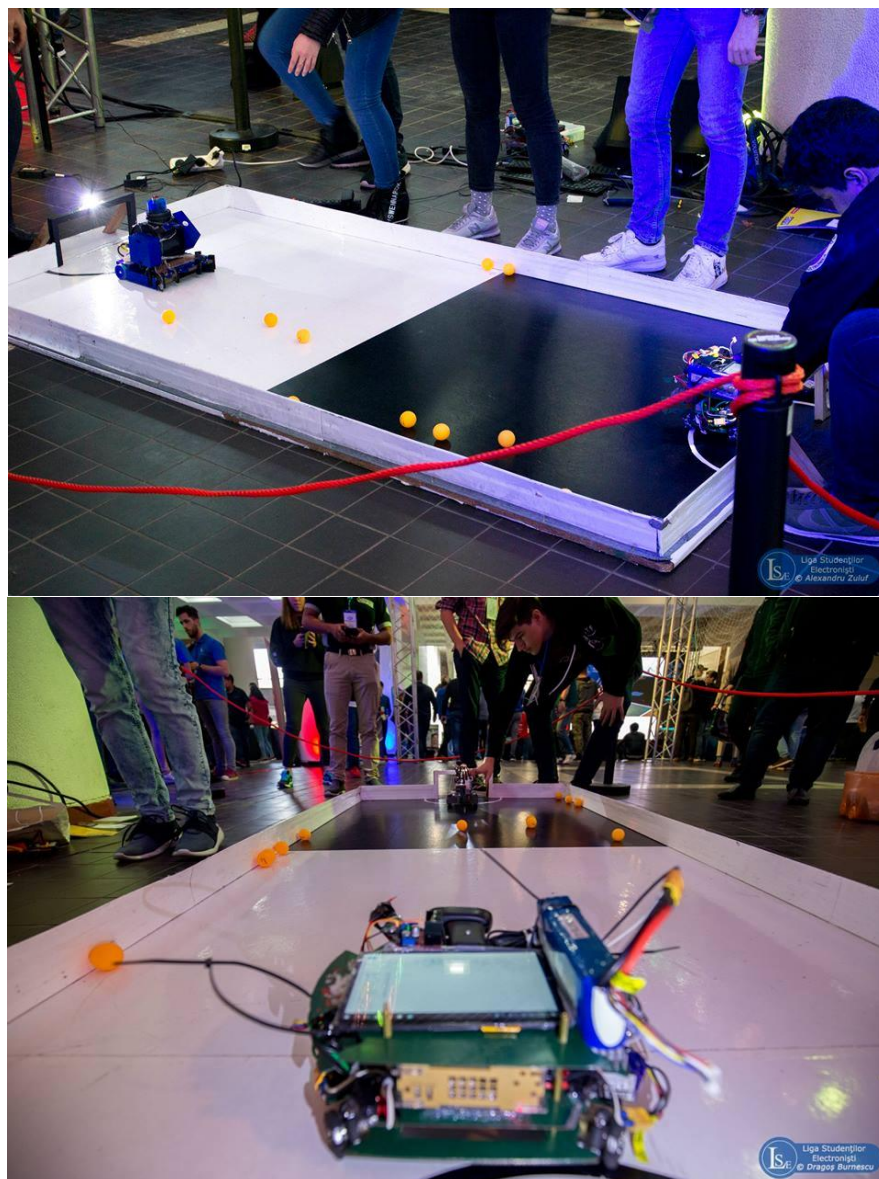


Figura 37: Participación Robochallenge Bucarest Rumania

5. CONCLUSIONES

1. Para el mejor desempeño es necesario seguir las medidas reglamentarias de tal manera que el centro de masa este lo más bajo posible y evitar volcamientos.
2. Para el procesamiento de imágenes es necesaria una cámara que cumpla con los requerimientos para la obtención de imágenes ya que la resolución o pixelaje influye al proceso.
3. Para la obtención de datos es necesario hacer las calibraciones con el robot en estado de reposo y no en movimiento.
4. Una de las grandes ventajas que presenta del embebido que posee una interfaz de programación usando un Atmel 32ua para la adquisición y el envío de datos seriales.
5. La tarjeta de control del robot futbolista posee una entrada USB la cual sirve para la cámara y salidas digitales con protocolo serial para la comunicación con las Sabertooth.

6. RECOMENDACIONES

1. Tener conocimiento básico en lo que refiere a programación tipo bloques y un correcto manejo del software de instrumentación virtual labview.
2. La lattepanda debe tener protecciones para su alimentación y tener sus respectivos disipadores para evitar el sobrevoltaje del procesador.
3. Precautelar que el disparador o recolector no tenga ninguna obstrucción que no permita su correcto movimiento.
4. Tomar en cuenta la posición angular de la cámara y evitar que se mueva o pierda su ángulo establecido.
5. Tomar en cuenta el tiempo de trabajo con respecto al amperaje de las baterías y evitar su avería.
6. Se recomienda utilizar la versión de LABVIEW 2016 para evitar problemas de compatibilidad y acceso a código.

7. REFERENCIAS BIBLIOGRAFICAS

- [1] Robocup, 2018. [En línea]. Available: <https://robocupap2018.org/>.
- [2] MIROSOT, «MIROSOT,» 2016. [En línea]. Available: <http://www.inovamicro.com/articles/how-is-robot-soccer-game-played-fira-mirosot/>.
- [3] F. R. W. Congress, «FIRA Robot World Congress,» 2003. [En línea]. Available: <http://www.ihrt.tuwien.ac.at/FIRAWC03/Content.htm>.
- [4] Robochallenge-Rom, «Robochallenge,» 2006. [En línea]. Available: <http://bfrz.ro/en>.
- [5] D. P. Chiluiza y V. T. Salamea , « DISEÑO E IMPLEMENTACIÓN DE UN EQUIPO DE ROBOTS AUTÓNOMOS CON DECISIONES EN TIEMPO REAL: FÚTBOL ROBÓTICO COMPONENTE ELETROMECAÁNICO,» *Escuela Superior Politécnica del Litoral* , p. 20, 2003.
- [6] Electronilab, «Electronilab,» 2018. [En línea]. Available: <https://electronilab.co/tienda/lattepanda-4gb-ram-64gb-mini-pc-windows-10/>.
- [7] DfRobot, «DFROBOT,» 2017. [En línea]. Available: <https://www.dfrobot.com/product-1585.html>.
- [8] Microchip, «Didacticas electronicas,» 2016. [En línea]. Available: <https://www.didacticaselectronicas.com/index.php/microcontroladores/microcontrolador-atmel-atmega32u4-mur-detail>.
- [9] Arduino, «Arduino,» 2008. [En línea]. Available: <https://arduino.cl/arduino-leonardo/>.
- [10] G. B. L. VERGARA, «REALIZACIÓN DE UN SISTEMA SCADA PARA EL CONTROL DE IMAGEN UTILIZANDO IMAQ DE LABVIEW Y CAMARA DE OBJETOS PARA EL LABORATORIO DE MECATRONICA,» *G. VERGARA*, p. 111, 2012.
- [11] J. A. L. Celi, «Diseño de un prototipo de sillas de ruedas electrica, con sistema de ascenso y elevacion,» Universidad Politecnica Salesiana, Cuenca, 2013.
- [12] Dimension Engineering, «Dimension Engineering,» 2004. [En línea]. Available: <https://www.dimensionengineering.com/products/sabertooth2x12>.
- [13] B. Lozoya, « Desarrollo de una plataforma hardware para la Robocup Small Soccer League (SSL).,» *B.J.*, p. 120, 2010.
- [14] Labview - Vision Development Module, «National Instrument,» 2017. [En línea]. Available: <http://www.ni.com/download/vision-development-module-2017/6640/en/>.
- [15] Df Robot, «DfRobot,» 2008. [En línea]. Available: <https://www.dfrobot.com/product-1552.html>.
- [16] Logitech, «C270 HD,» 2019. [En línea]. Available: <https://www.logitech.com/es-mx/product/hd-webcam-c270>.
- [17] LABVIEW, «<http://www.ni.com>,» 2016. [En línea]. Available: <http://www.ni.com/es-cr/shop/labview.html>.
- [18] D. C. Holguin, « Diseño e implementación de un control Pid con ganancias programadas utilizando un sistema embebido launchpad mps430 para planta balón y viga,» *Holguin, David Carpio*, p. 104, Agosto 2018.

- [19] Dimension Engineering LLC., «<https://www.dimensionengineering.com>,» 2003. [En línea]. Available: <https://www.dimensionengineering.com/info/arduino>.
- [20] MSP430, «MSP430 LaunchPad Value Line Development kit,» Septiembre 2014. [En línea]. Available: <http://www.ti.com/tool/MSP-EXP430G2>.
- [21] H. U. V. GONZALEZ, 309696 - MODULO DE MICROPROCESADORES & MICROCONTROLADORES, CHIQUINQUIRA: UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA, 2009.
- [22] S. Heath, Embedded System Design, vol. 2, Oxford: Newnes, 2003.
- [23] FutureElectronics, «FutureElectronics,» 3 Marzo 2017. [En línea]. Available: <http://www.futureelectronics.com/en/drivers/motor-driver.aspx>.
- [24] A. S. Poznyak, «Modelado Matematico de los sistemas mecanicos, electrico y electromecanicos,» 2 Mayo 2005. [En línea]. Available: https://www.ctrl.cinvestav.mx/~coordinacion/documents/cursos/modelado_matematico.pdf.
- [25] Gottlieb, Electric Motors & Control Techniques, 1994.
- [26] FingerTechRobotics, «FingerTech "Silver Spark" 16mm Gearmotor,» 2016. [En línea]. Available: <https://www.fingertechrobotics.com/proddetail.php?prod=ft-Sspark16>.
- [27] Logitech, «Logitech,» 2015. [En línea]. Available: <https://www.logitech.com/es-mx/product/hd-webcam-c270>.

ANEXOS

Anexo 1 Librería Sabertooth [19]

```
#ifndef SabertoothSimplified_h
#define SabertoothSimplified_h

#if defined(ARDUINO) && ARDUINO >= 100
#include <Arduino>
#else
#include <WProgram.h>
#endif

#if defined(USBCON)
#define SabertoothTXPinSerial Serial1 // Arduino Leonardo has TX->1 on
Serial1, not Serial.
#else
#define SabertoothTXPinSerial Serial
#endif
#define SyRenTXPinSerial SabertoothTXPinSerial

/*!
class SabertoothSimplified
{
public:
    SabertoothSimplified();

    /*!
    Initializes a new instance of the SabertoothSimplified class.
    The specified serial port is used.
    \param port The port to use.
    */
    SabertoothSimplified(Print& port);

public:
    /*!
    Sets the power of motor 1.
    \param power The power, between -127 and 127.
    */
    void motor(int power);

    /*!
    Sets the power of the specified motor.
    \param motor The motor number, 1 or 2.
    \param power The power, between -127 and 127.
    */
    void motor(byte motor, int power);

    /*!
```

```

Sets the driving power.
\param power The power, between -127 and 127.
*/
void drive(int power);

/*!
Sets the turning power.
\param power The power, between -127 and 127.
*/
void turn(int power);

/*!
Stops.
*/
void stop();

private:
void mixedMode(boolean enable);
void mixedUpdate();
void raw(byte motor, int power);

private:
boolean _mixed;
int _mixedDrive, _mixedTurn;
boolean _mixedDriveSet, _mixedTurnSet;
Print& _port;
};

#endif

```

Anexo 2 Control Omnidireccional

```
#include <SoftwareSerial.h>
```

```
#include <SabertoothSimplified.h>
```

```
SoftwareSerial SWSerial1(NOT_A_PIN, 11); // pin RXn no es usado solo el pin TX y asignado al pin 11 (to S1).
```

```
SabertoothSimplified ST1(SWSerial1); // asignacion de objeto como SWSerial como puerto serial.
```

```
SoftwareSerial SWSerial2(NOT_A_PIN, 10); // pin RXn no es usado solo el pin TX y asignado al pin 10 (to S2).
```

```
SabertoothSimplified ST2(SWSerial2); // asignacion de objeto como SWSerial2 como puerto serial.
```

```
int maxup=50;//velocidad máxima de robot
```

```
int velocidad=40;//velocidad de giro =maxup+velocidad
```

```
char inChar;//dato de entrada serial
```

```
void setup()
```

```
{  
  Serial.begin(9600);//configuración de puerto  
  SWSerial1.begin(9600); //configuración de puerto  
  SWSerial2.begin(9600); //configuración de puerto  
}
```

```
void loop()
```

```
{  
  opcion();  
}
```

```
void opcion(){
```

```
  inChar = (char)Serial.read(); //espera dato desde la lattepanda mediante puerto serial
```

```
  if (inChar=='s'){  
    detener();//ir al método detener  
  }
```

```
  else if (inChar=='f'){  
    adelante();//ir al método adelante  
  }
```

```
  else if (inChar=='b'){  
    atras();//ir al método atras  
  }
```

```
  else if (inChar=='l'){  
    lado_izquierda();//ir al método izquierda  
  }
```

```
  else if (inChar=='r'){  
    lado_derecha();//ir al método derecha  
  }
```

```
  else if (inChar=='g'){  
    giro_derecha();  
  }
```

```
  else if (inChar=='t'){  
    giro_izquierda();  
  }
```

```

else if (inChar=='c'){//colectar
  digitalWrite(d1,LOW);
  analogWrite(d2,255);
}
else if (inChar=='e'){//expulsar
  analogWrite(d1,255);
  digitalWrite(d2,LOW);
}
else if (inChar=='d'){//detener
  digitalWrite(d1,LOW);
  digitalWrite(d2,LOW);
}
else{
  detener();
}
}
void detener(){
  ST1.motor(1,0);//set motor saber1
  ST1.motor(2,0); //set motor saber1
  ST2.motor(1,0); //set motor saber2
  ST2.motor(2,0); //set motor saber2
}

void adelante(){
  ST1.motor(1, maxup); //set motor 1 saber1
  ST1.motor(2, maxup); //set motor 2 saber1
  ST2.motor(1, maxup); //set motor 3 saber2
  ST2.motor(2, maxup); //set motor 4 saber2
  delay(5);
}

void atras(){
  ST1.motor(1, -maxup); //set motor 1 saber1
  ST1.motor(2, -maxup); //set motor 2 saber1
  ST2.motor(1, -maxup); //set motor 3 saber2
  ST2.motor(2, -maxup); //set motor 4 saber2
  delay(5);
}

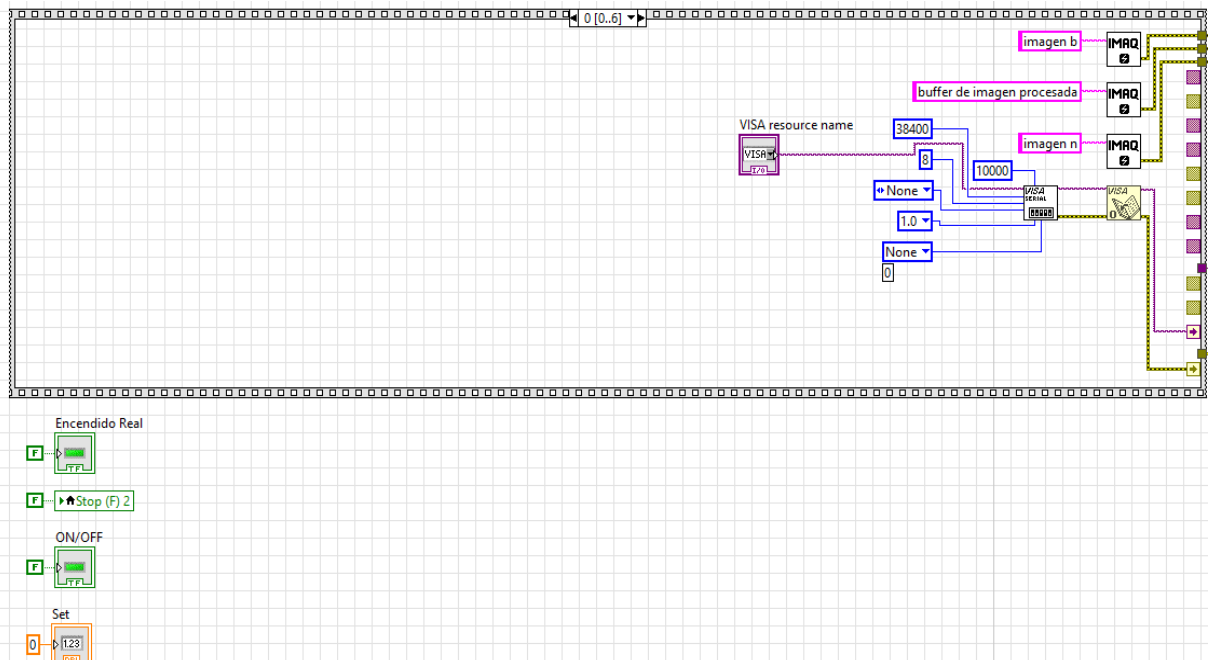
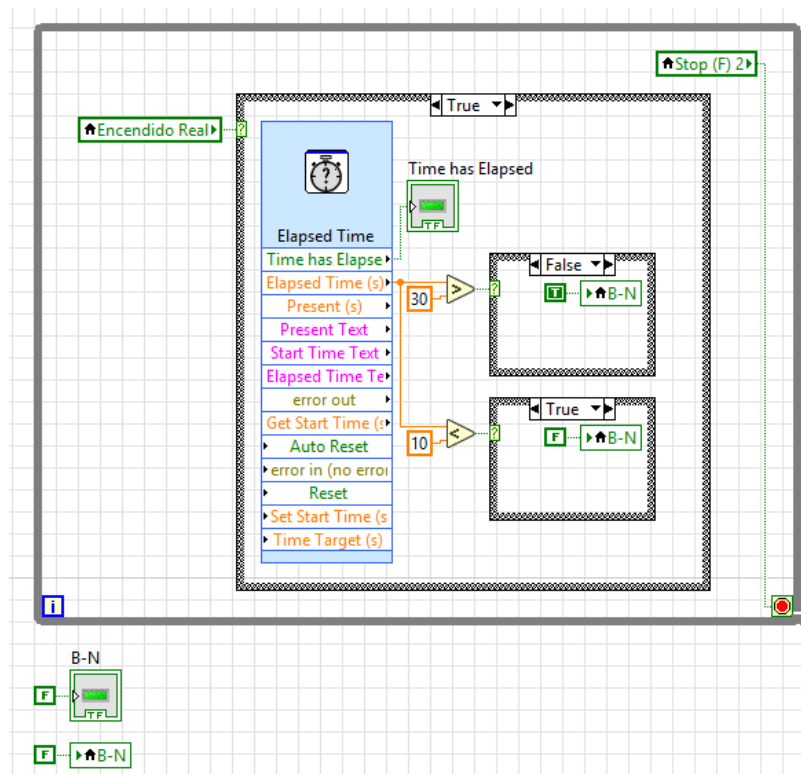
void giro_derecha(){
  ST1.motor(1, maxup+velocidad); //set motor 1 saber1
  ST1.motor(2, maxup+velocidad); //set motor 2 saber1
  ST2.motor(1, -maxup-velocidad); //set motor 3 saber2
  ST2.motor(2, -maxup-velocidad); //set motor 4 saber2
  delay(5);
}

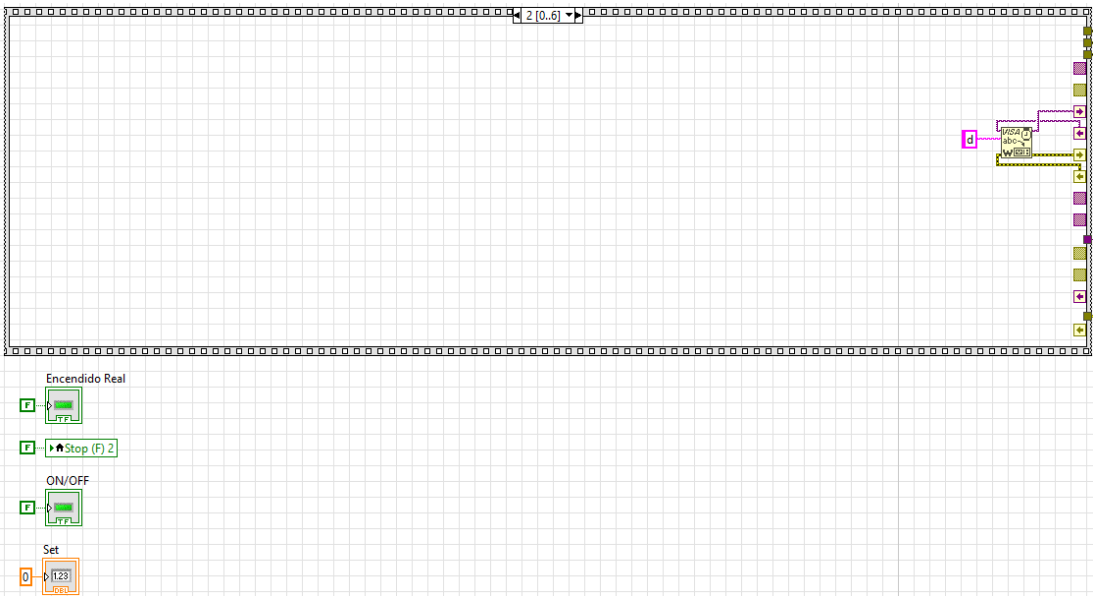
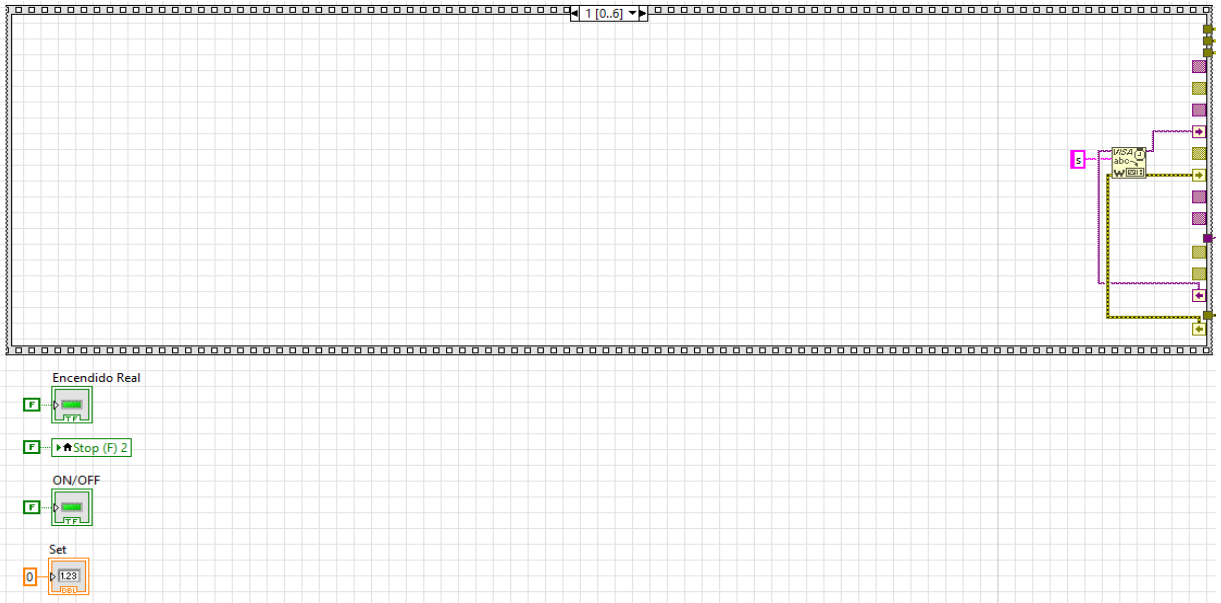
void giro_izquierda(){
  ST1.motor(1, -maxup-velocidad); //set motor 1 saber1
  ST1.motor(2, -maxup-velocidad); //set motor 2 saber1
  ST2.motor(1, maxup+velocidad); //set motor 3 saber2
  ST2.motor(2,maxup+velocidad); //set motor 4 saber2
  delay(5);
}

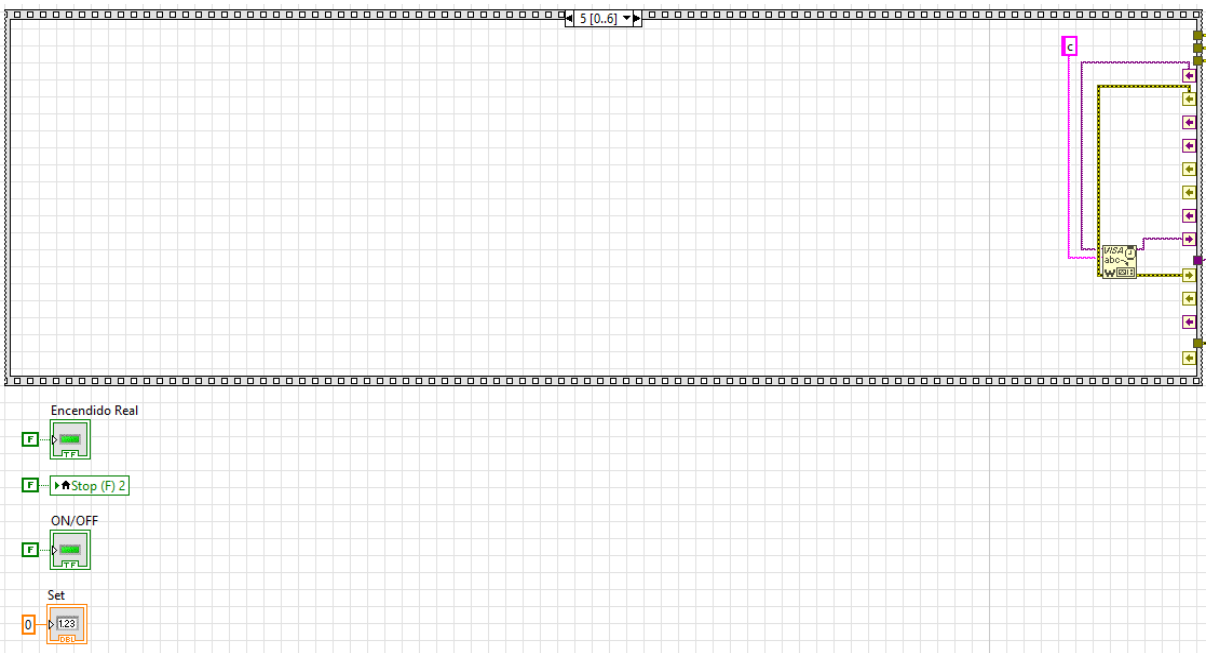
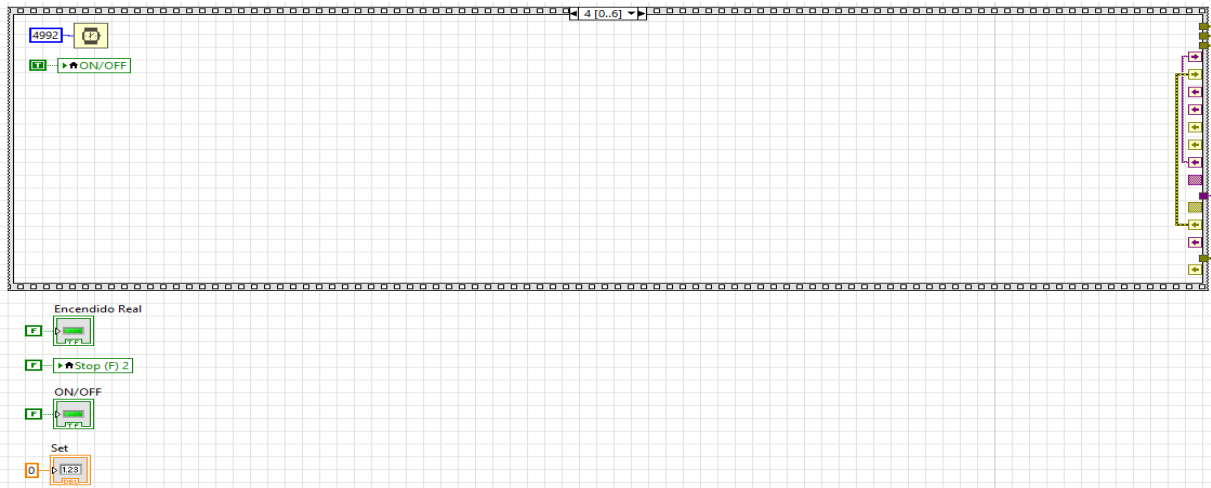
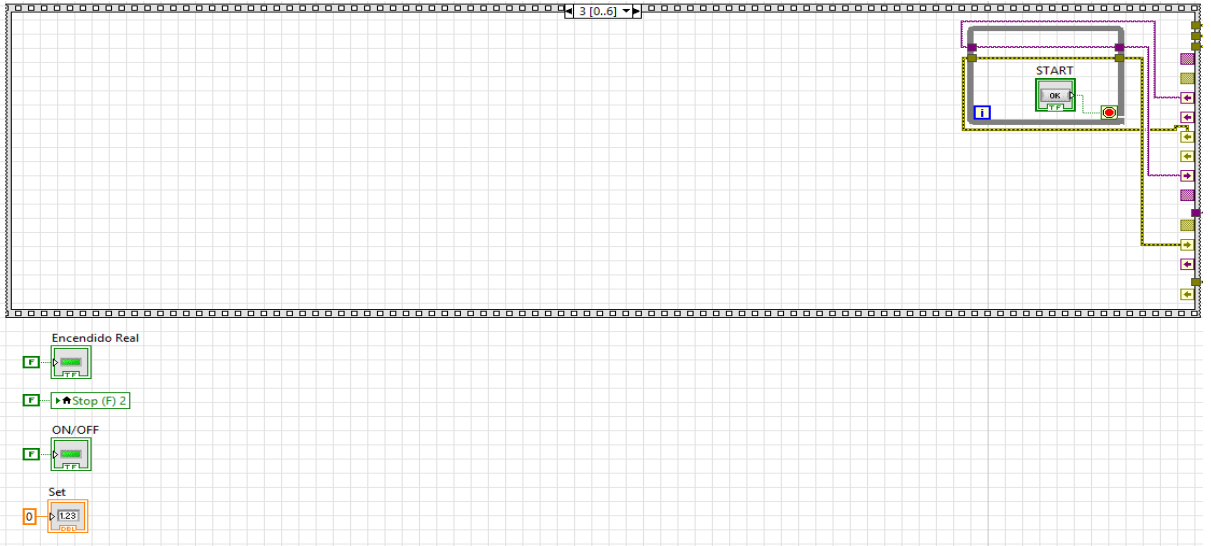
```

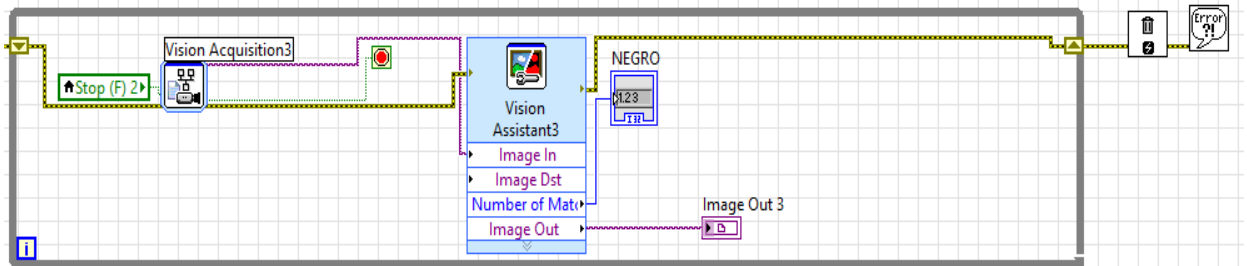
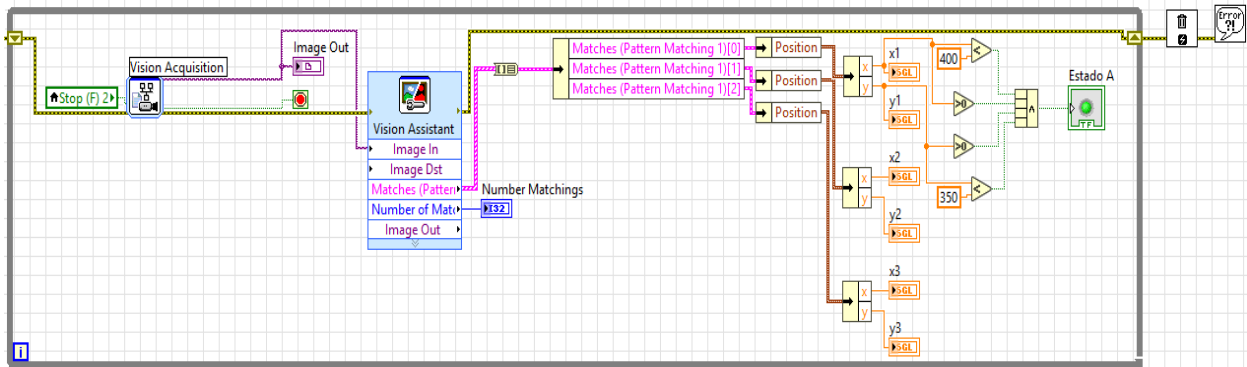
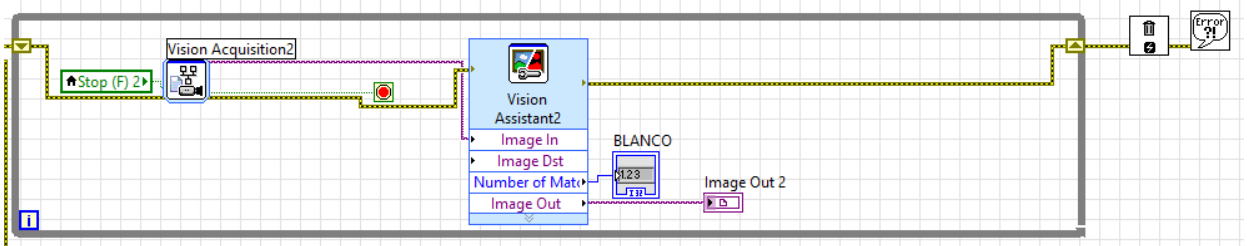
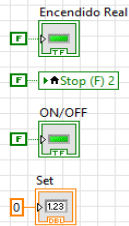
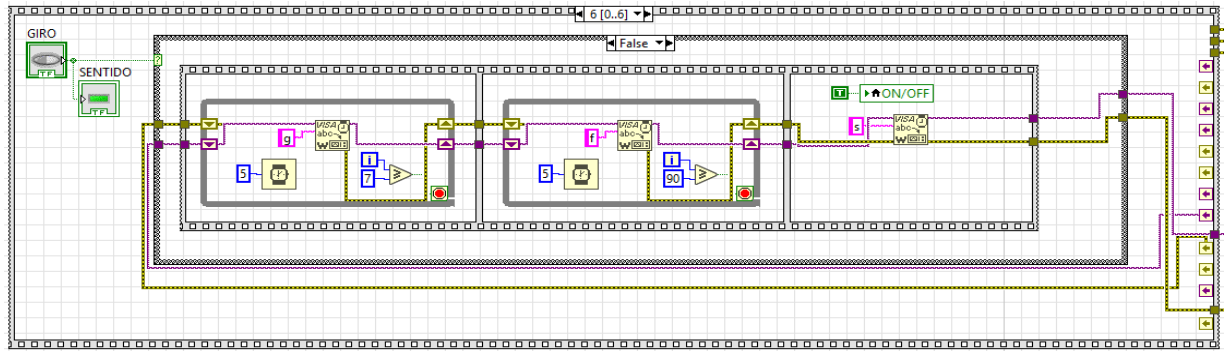
```
}  
void lado_derecha(){  
    ST1.motor(1, maxup); //set motor 1 saber1  
    ST1.motor(2, -maxup); //set motor 2 saber1  
    ST2.motor(1, -maxup); //set motor 3 saber2  
    ST2.motor(2,maxup); //set motor 4 saber2  
    delay(5);  
}  
void lado_izquierda(){  
    ST1.motor(1, -maxup); //set motor 1 saber1  
    ST1.motor(2, maxup); //set motor 2 saber1  
    ST2.motor(1, maxup); //set motor 3 saber2  
    ST2.motor(2,-maxup); //set motor 4 saber2  
    delay(5);  
}
```

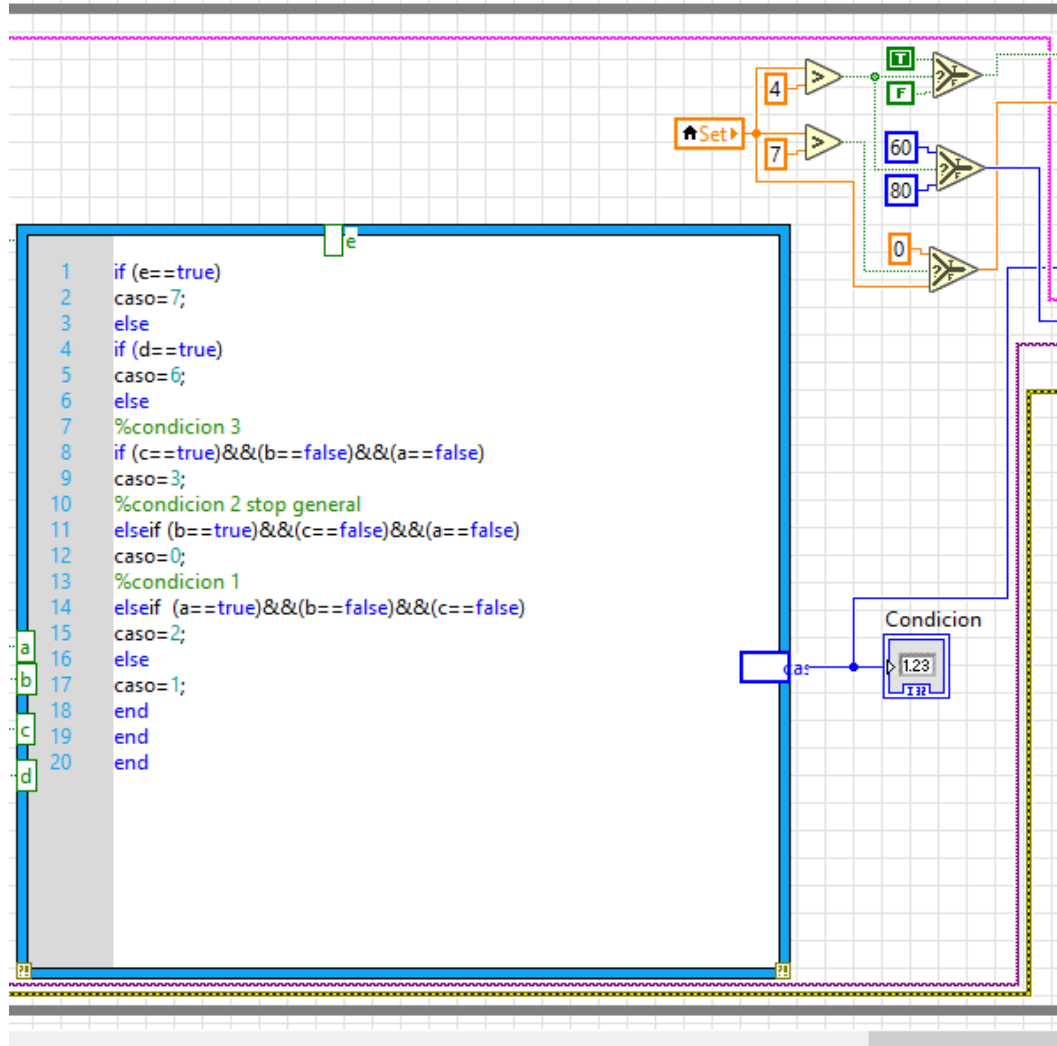
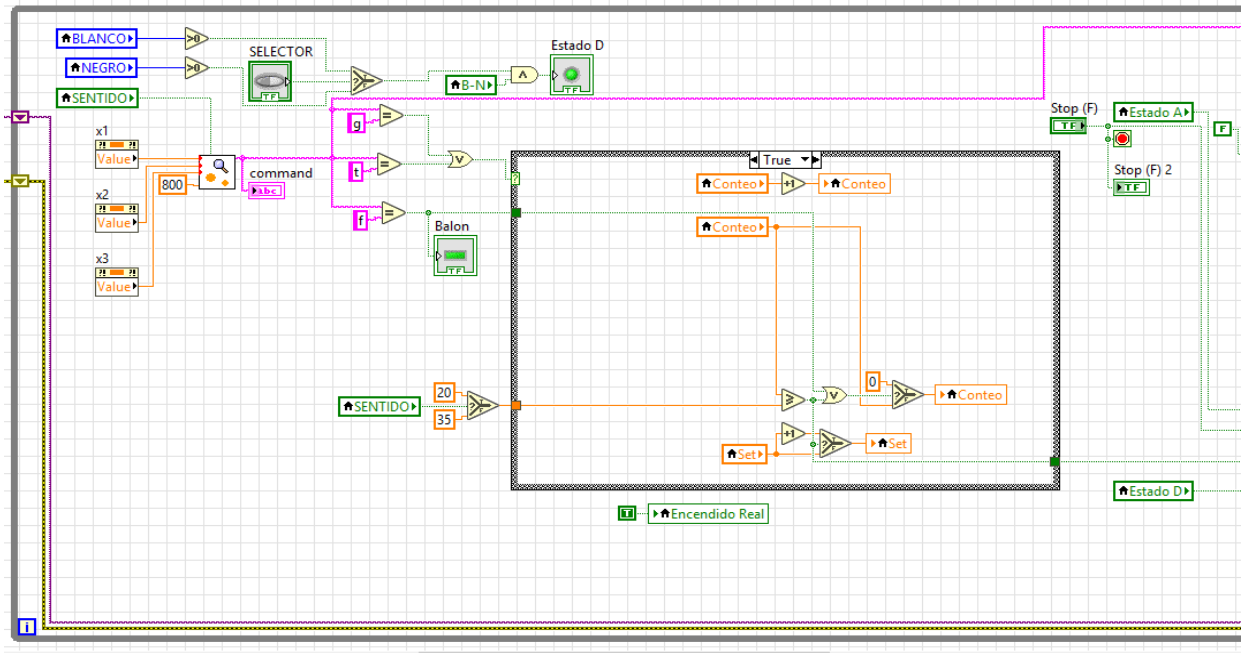

Anexo 3 Diagrama de control Labview

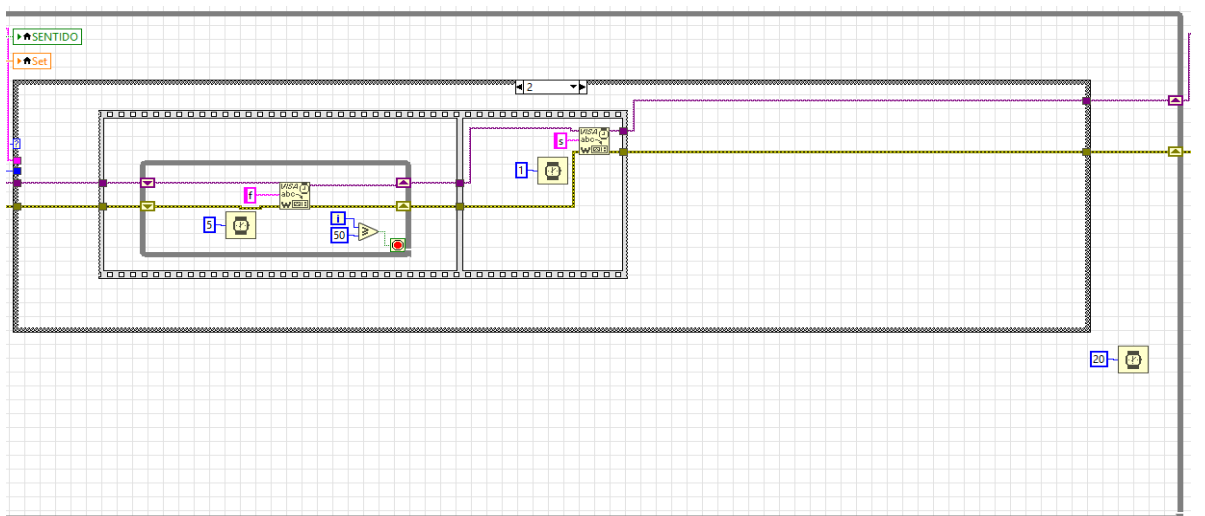
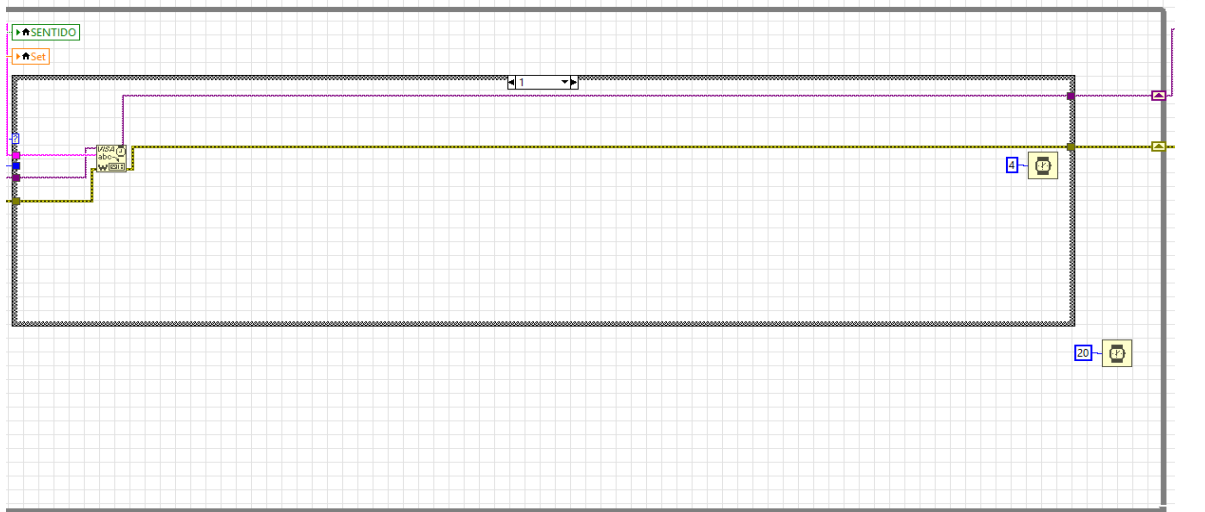
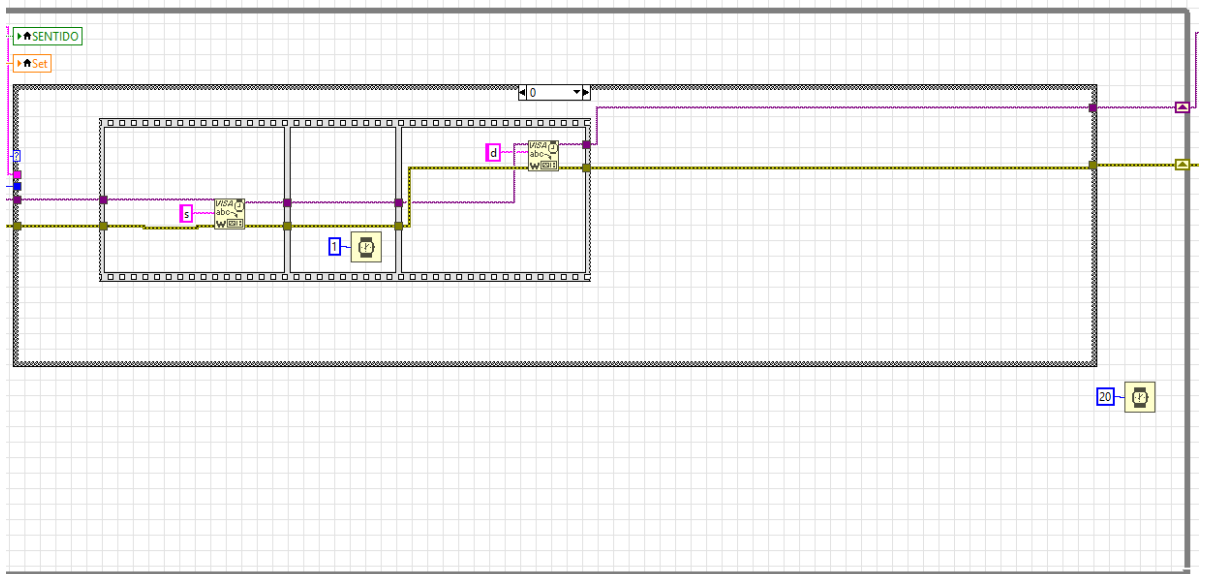


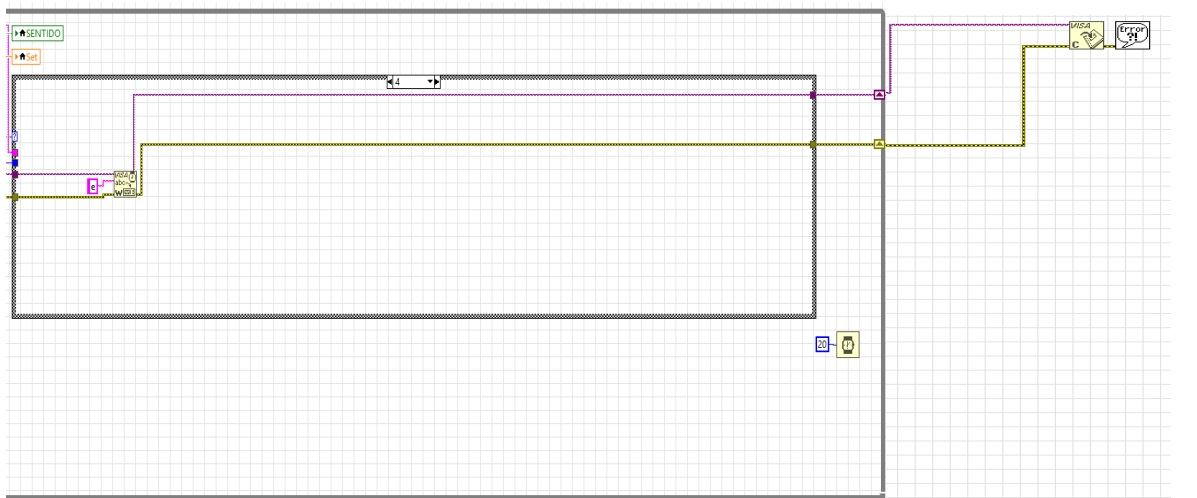
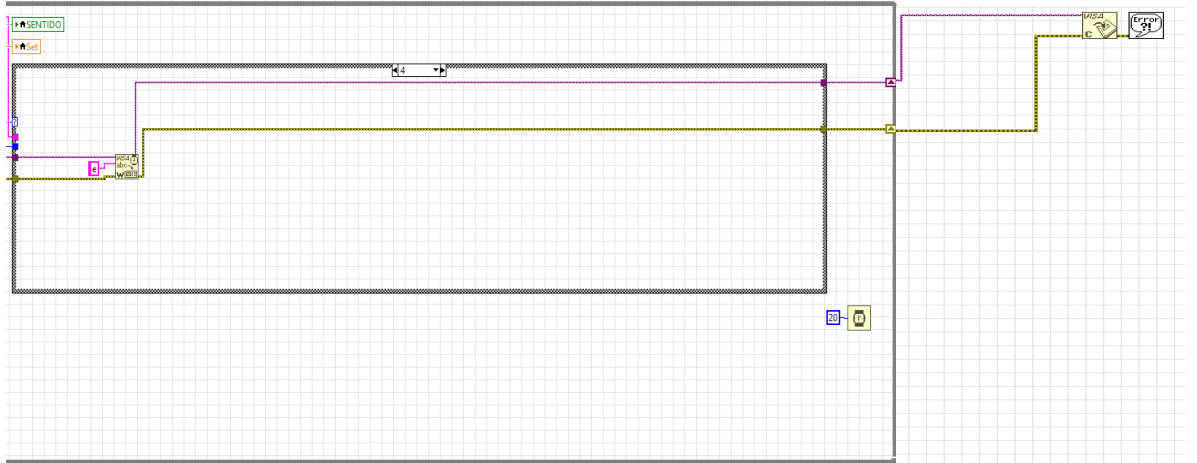
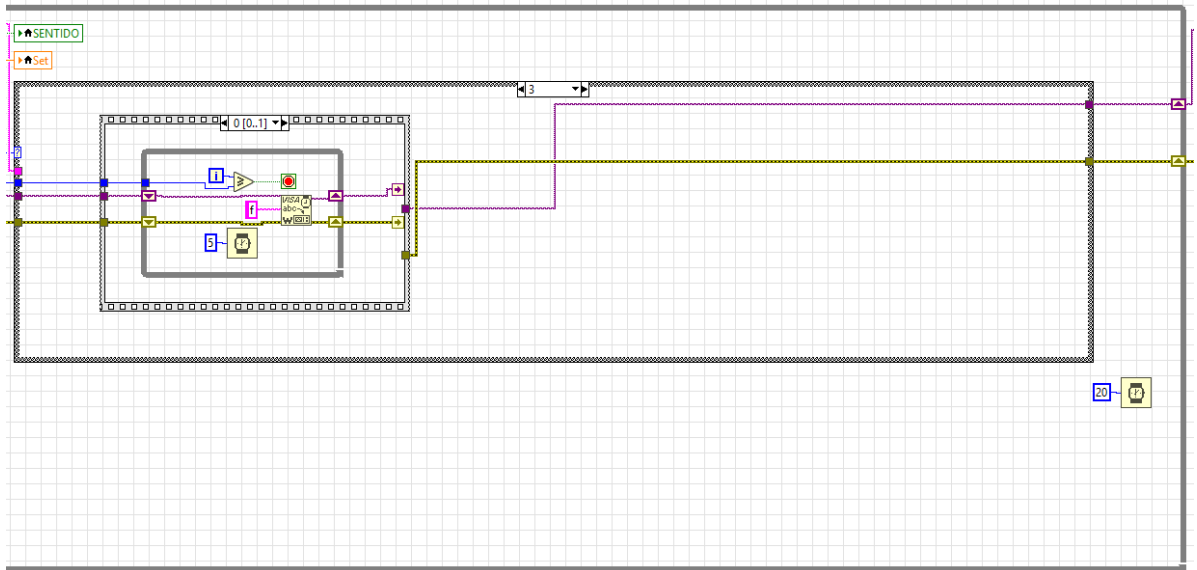


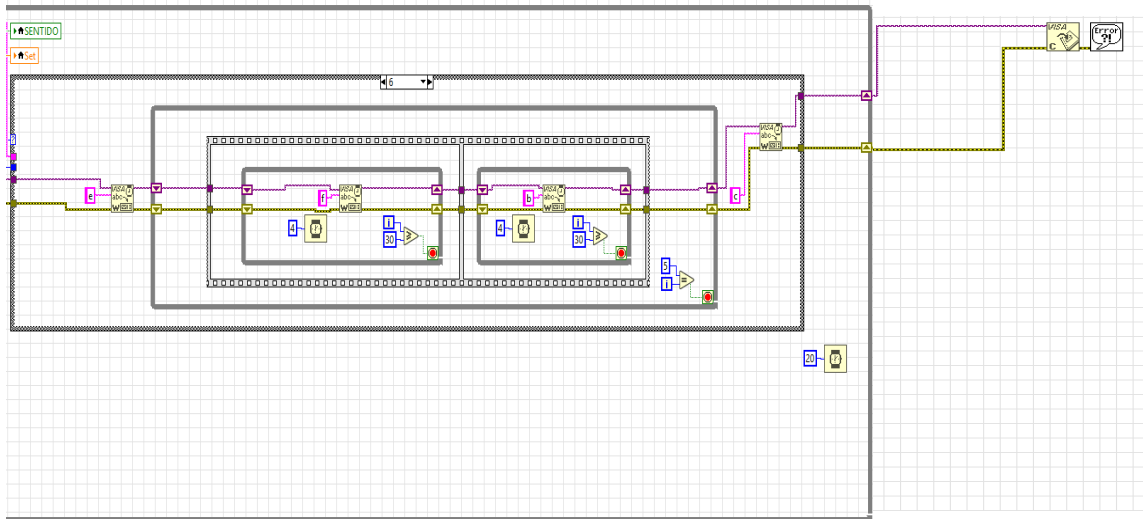
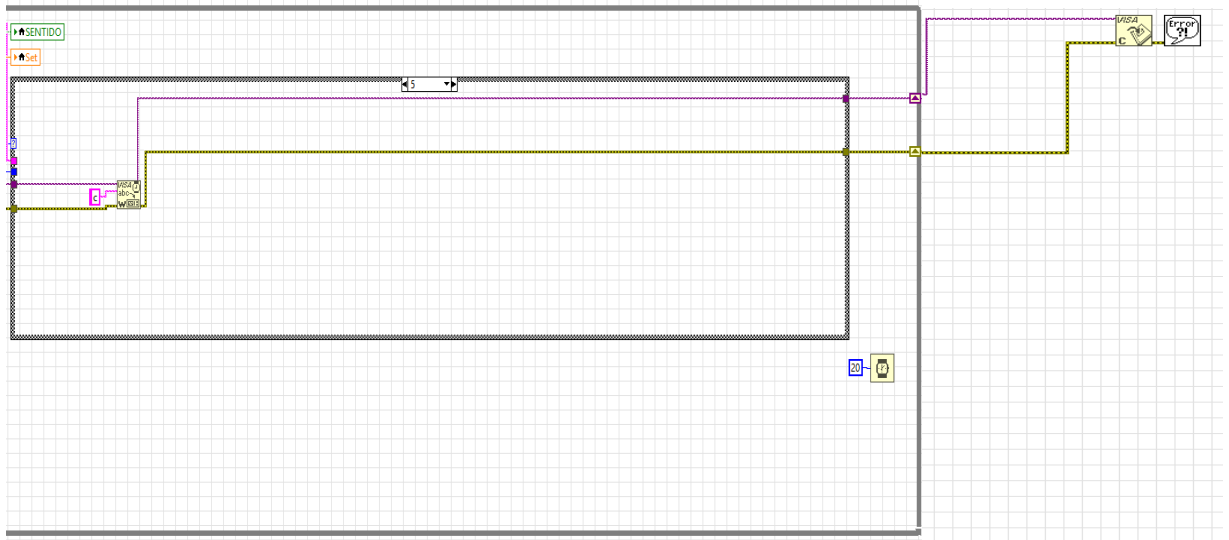












Anexo 4 Especificaciones Motor Fingertech table rpm vs aplicación de voltaje

Speed (rpm)	Voltage						
Gear ratio	5	6	7.4	11.1	14.8	18.5	22.2
11.1 :1	709	850	1049	1573	2097	2622	3146
22.2 :1	354	425	524	786	1049	1311	1573
33.3 :1	236	283	350	524	699	874	1049
50 :1	157	189	233	349	466	582	698
83.3 :1	94	113	140	210	279	349	419
100 :1	79	94	116	175	233	291	349
200 :1	39	47	58	87	116	146	175
300 :1	26	31	39	58	78	97	116
600 :1	13	16	19	29	39	49	58

Anexo 5 Especificaciones Motor Fingertech table torque vs aplicación de voltaje

Torque (kg-cm)	Voltage						
Gear ratio	5	6	7.4	11.1	14.8	18.5	22.2
11.1 :1	0.35	0.42	0.52	0.77	1.03	1.29	1.55
22.2 :1	0.70	0.84	1.03	1.55	2.07	2.58	3.10
33.3 :1	1.05	1.26	1.55	2.32	3.10	3.87	4.65
50 :1	1.57	1.89	2.33	3.49	4.65	5.82	6.98
83.3 :1	2.62	3.14	3.88	5.81	7.75	9.69	11.63
100 :1	3.14	3.77	4.65	6.98	9.30	11.63	13.96
200 :1	6.29	7.54	9.30	13.96	18.61	23.26	27.91
300 :1	9.43	11.32	13.96	20.93	27.91	34.89	41.87
600 :1	18.86	22.63	27.91	41.87	55.82	69.78	83.74

Motor (Mabuchi FK-050SH-13125) Specifications

Operating Voltage: 3V - 18.5V (>10V decreases motor life)

Nominal Voltage: 6Vdc

No Load RPM:

11530rpm

No Load Current:

0.09A

Stall Current: 1.3A

Stall Torque: 0.0461kg-cm (0.64oz-in)

Kt: 0.0352kg-cm/A (0.49oz-in/A)

Kv: 1573rpm/V

Peak Efficiency: 52%

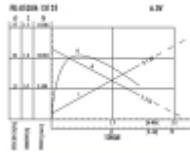
RPM @ Peak Eff:

9100

Current @ Peak Eff:

0.34A

Click [here](#) for the motor torque, current, rpm and efficiency curve.



Physical Specifications

Gearbox Lengths:

ratios 11: 11.5mm (0.45in)

ratios 22, 33, 50: 12.7mm (0.50in)

ratios 83, 100: 14.3mm (0.56in)

ratios 200, 300: 16.4mm (0.65in)

ratios 600: 17.9mm (0.70in)

Motor Length: 28.7mm (1.13in)

Motor weight (no gearbox): 17g (0.60oz)

Gearbox Diameter: 16mm (0.63in)

Shaft Diameter: 3mm (0.12in) with flat along length

Shaft Length: 38mm (1.5in)

Mounting Holes (2): #2-56 spaced 11mm (0.433in) apart

-do not use screws that protrude more than 2mm (0.078in) into gearbox

Weight:

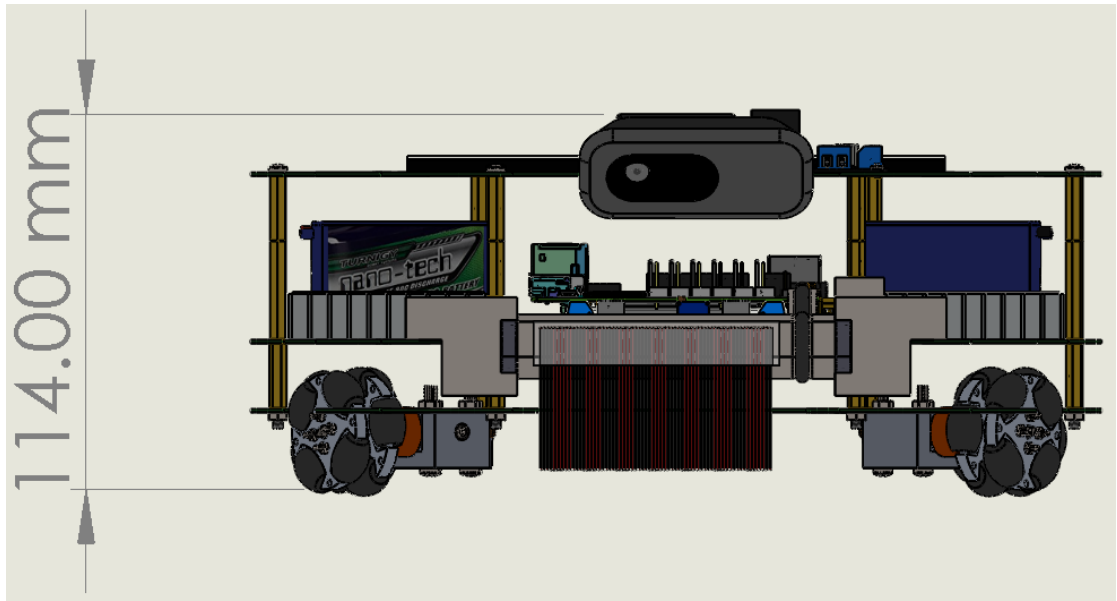
ratios 11: 28g (0.99oz)

ratios 22, 33, 50, 83, 100: 29g (1.02oz)

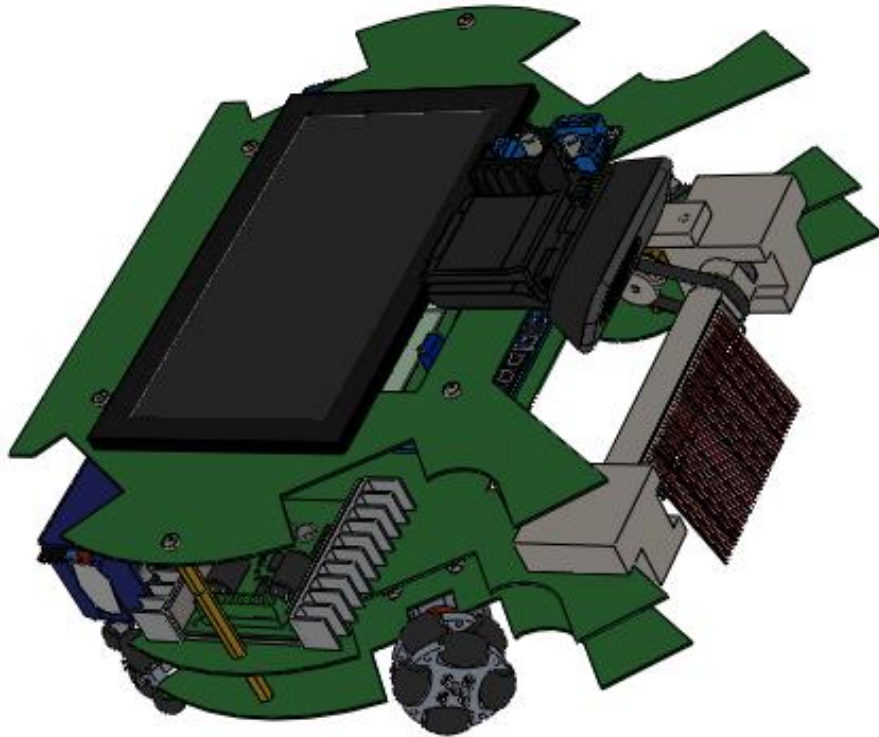
ratios 200, 300: 31g (1.09oz)

ratios 600: 32g (1.13oz)

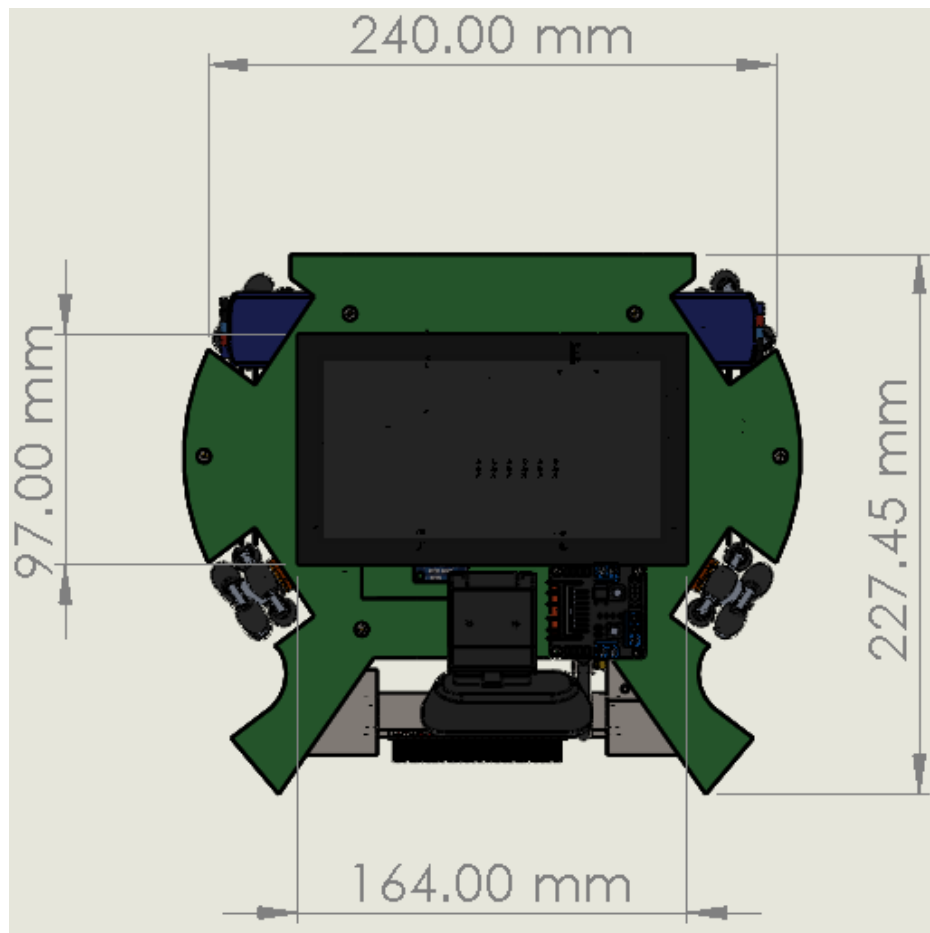
Anexo 6 diseño del prototipo



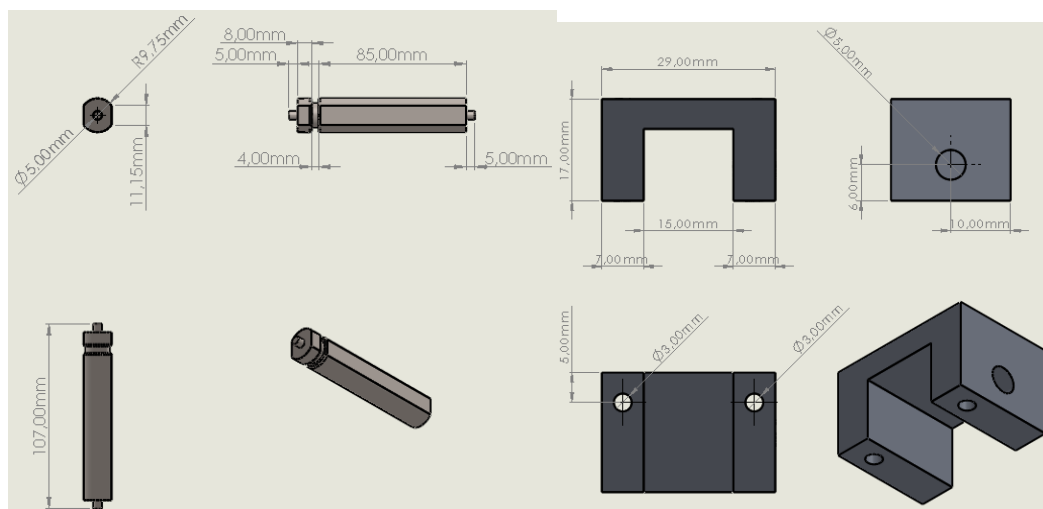
Vista Frontal



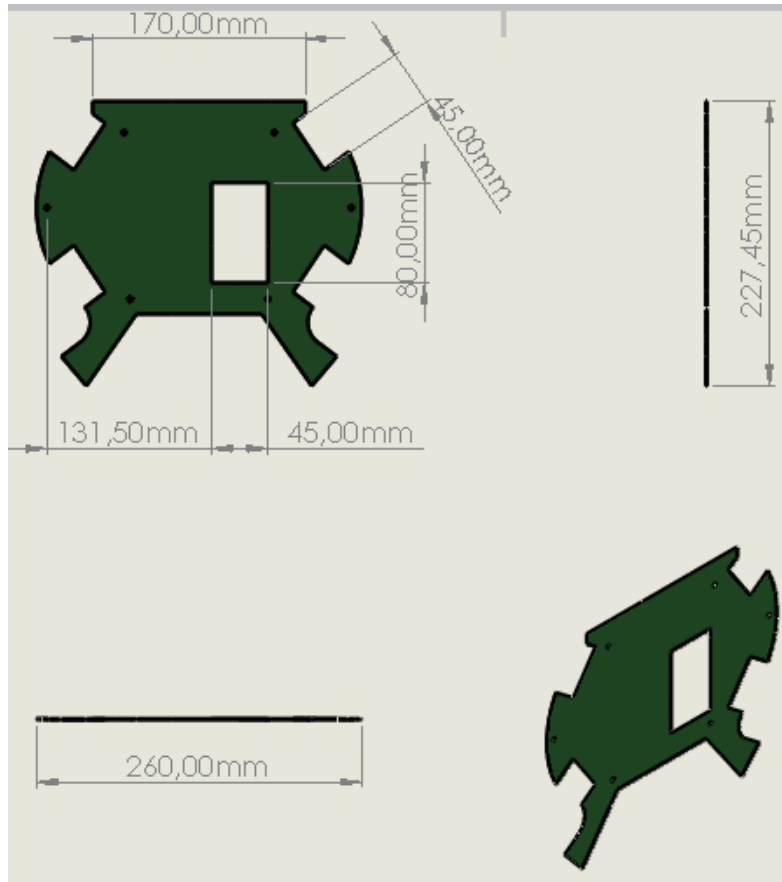
Vista Isometrica



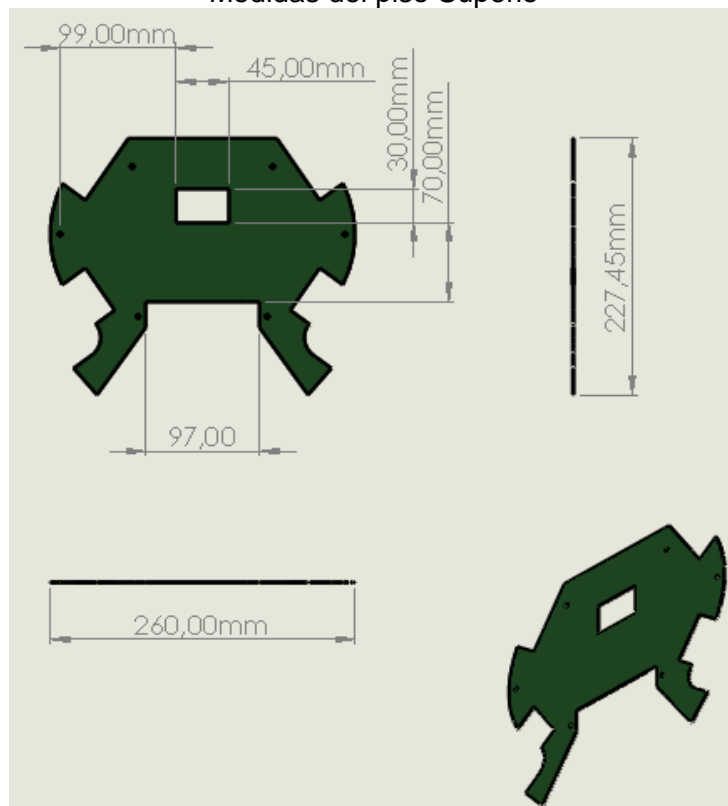
Vista Superior



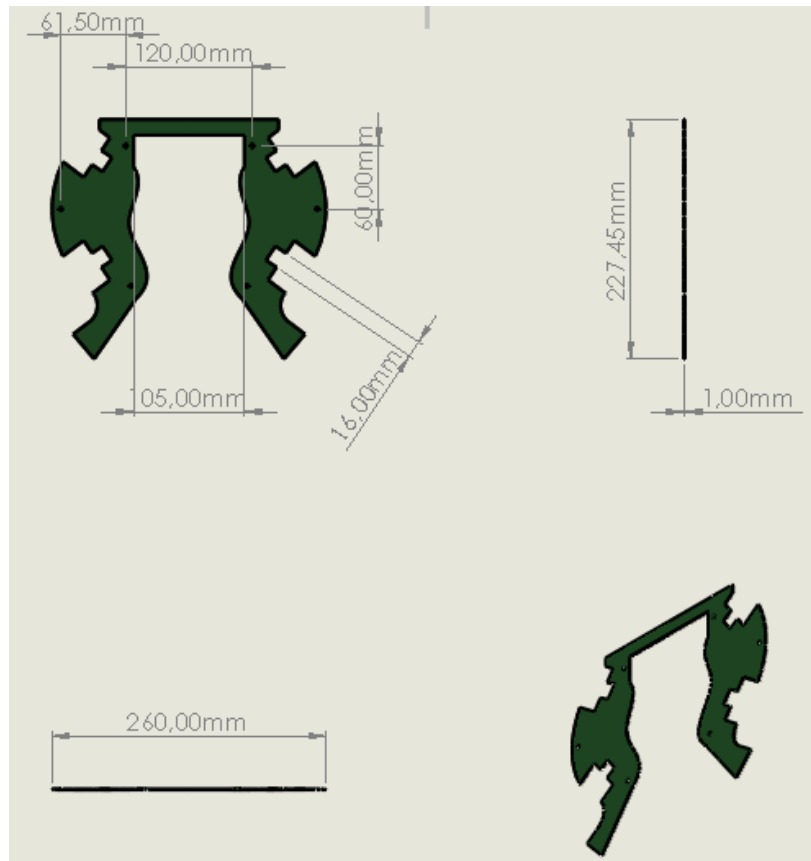
Elementos para suspensión de motor y pilares



Medidas del piso Superior

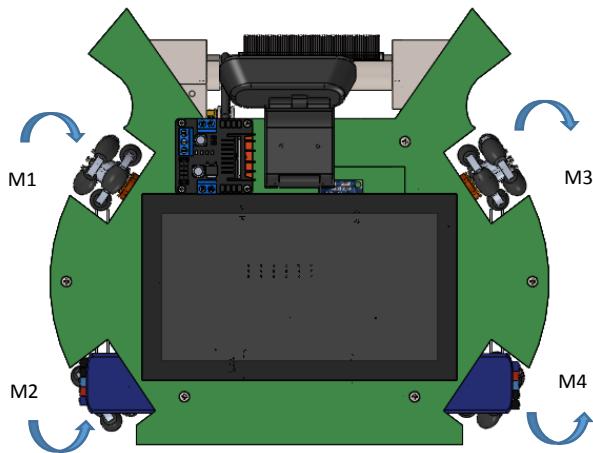


Medidas del piso Medio

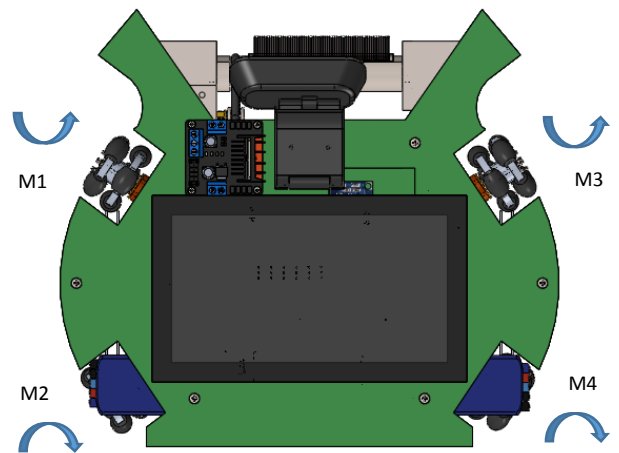


Medidas del piso inferior

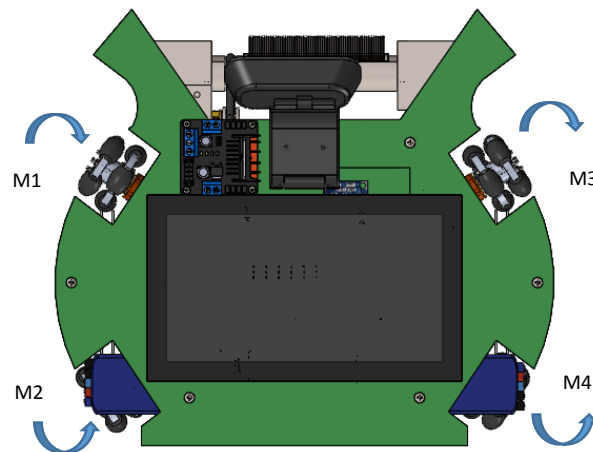
Anexo 7 Movimientos del robot



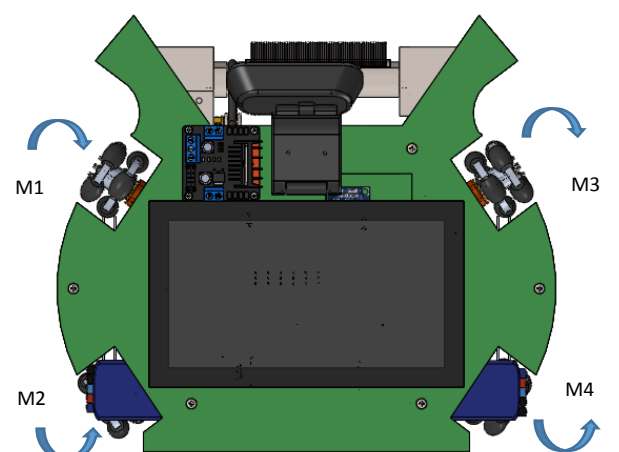
Movimiento Hacia Delante



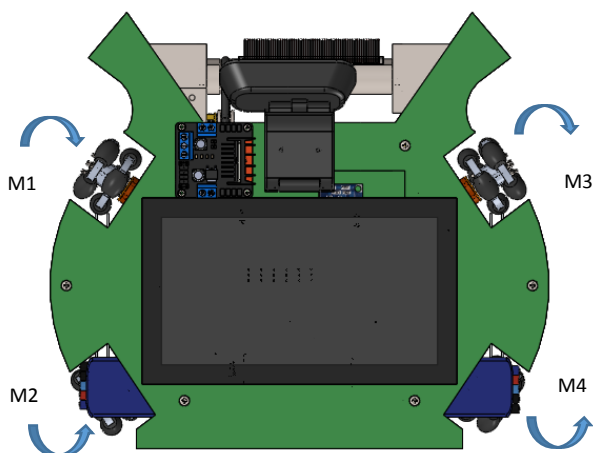
Movimiento Hacia Atrás



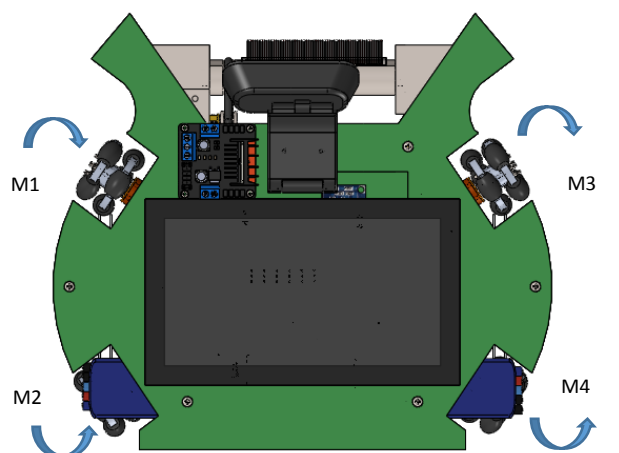
Movimiento Hacia la Izquierda



Movimiento Hacia la Derecha



Giro Izquierdo



Giro Derecho