



UNIVERSIDAD UTE

**FACULTAD DE CIENCIAS DE LA INGENIERÍA E
INDUSTRIAS**

CARRERA DE INGENIERÍA MECATRÓNICA

**Diseño y construcción de un sistema móvil autónomo para el
control de malezas**

**TRABAJO PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO MECATRÓNICO**

NELSON PATRICIO SUÁREZ MONTES

DIRECTOR: RAÚL VICENTE PAREDES LOOR

Quito, marzo 2023

© Universidad UTE. 2023

Reservados todos los derechos de reproducción

FORMULARIO DE REGISTRO BIBLIOGRÁFICO

TRABAJO DE TITULACIÓN

DATOS DE CONTACTO	
CÉDULA DE IDENTIDAD:	1724245145
APELLIDO Y NOMBRES:	Suárez Montes Nelson Patricio
DIRECCIÓN:	Urbanización Vencedores de Pichincha, calle Oe9B, casa S16-100 y calle S16
EMAIL:	nelson.suarez@ute.edu.ec
TELÉFONO FIJO:	02-2845784
TELÉFONO MOVIL:	0979017247

DATOS DE LA OBRA	
TÍTULO:	Diseño y construcción de un sistema móvil autónomo para el control de malezas.
AUTOR O AUTORES:	Nelson Patricio Suárez Montes
FECHA DE ENTREGA DEL PROYECTO DE TITULACIÓN:	21/03/2023
DIRECTOR DEL PROYECTO DE TITULACIÓN:	Raúl Vicente Paredes Loor
PROGRAMA	PREGRADO <input checked="" type="checkbox"/> POSGRADO <input type="checkbox"/>
TÍTULO POR EL QUE OPTA:	Ingeniero Mecatrónico
RESUMEN: Mínimo 250 palabras	El presente proyecto de investigación muestra el diseño y construcción de un sistema móvil autónomo para el control de malezas el cual se lo propone como una alternativa a las técnicas usadas tradicionalmente en la agricultura que afectan tanto al cultivo como al agricultor, para ello se implementó una estructura con un sistema de tracción tipo oruga el cual

le permite una mayor superficie de contacto logrando así un desplazamiento adecuado en suelo irregular, la eliminación de la maleza se la realizó mediante el empleo de una herramienta de un grado de libertad cuya función es cortar la maleza que crece alrededor de los cultivos, esto se logra con un algoritmo desarrollado en el lenguaje de programación Python que permite el reconocimiento de las plantas de cultivo mediante la adquisición de imágenes e implementación de visión artificial, para la detección de objetos se usa dos estrategias diferentes, la primera segmenta objetos por medio de la identificación de su color y la segunda es una red neuronal llamada Haar-Cascade que fue entrenada para el reconocimiento de las plantas de interés en este caso lechugas, la primera estrategia obtuvo un porcentaje de exactitud y precisión del 93% y 96% respectivamente y la segunda del 84% y 92.2%, cuando el algoritmo detecta el objeto también determina el centroide del mismo considerando a este punto como una posición final o deseada a la cual el robot tiene que desplazarse, esto se lo consigue por intermedio de la estrategia de control que utiliza el modelo cinemático del robot móvil diferencial, matriz Jacobiana, y teoría de Lyapunov, esta estrategia se encarga de determinar el conjunto de velocidades que son necesarias para trasladarse hacia el punto deseado controlando las velocidades de los motores del robot por medio del protocolo de comunicación Bluetooth con el microcontrolador ESP32 embebido.

PALABRAS CLAVES:

Python, visión artificial, Haar-Cascade, robot diferencial, Jacobiano, Lyapunov.

ABSTRACT:

This research project shows the design and construction of an autonomous mobile system for weed control which is proposed as an alternative to the techniques traditionally used in agriculture that affect both the crop and the farmer, for this a structure was implemented with a traction system type caterpillar which allows a greater contact surface thus achieving an adequate displacement on uneven ground, the elimination of weeds was done by using a tool with a degree of freedom whose function is to cut the weeds growing around crops, This is achieved with an algorithm developed in the Python programming language that allows the recognition of crop plants through the acquisition of images and implementation of artificial vision, for the detection of objects two different strategies are used, the first segments objects by identifying their color and the second is a neural network called Haar-Cascade that was trained to recognize the plants of interest in this case lettuce, the first strategy obtained a percentage of accuracy and precision of 93% and 96% respectively and the second of 84% and 92. 2%, when the algorithm detects the object it also determines the centroid of the object considering this point as a final or desired position to which the robot has to move, this is achieved through the control strategy that uses the kinematic model of the differential mobile robot, Jacobian matrix, and Lyapunov theory, this strategy is responsible for determining the set of speeds that are necessary to move to the desired

	point by controlling the speeds of the robot motors through the Bluetooth communication protocol with the embedded ESP32 microcontroller.
KEYWORDS	Python, computer vision, Haar-Cascade, differential robot, Jacobian, Lyapunov.

Se autoriza la publicación de este Proyecto de Titulación en el Repositorio Digital de la Institución.



f: _____

SUÁREZ MONTES NELSON PATRICIO

1724245145

DECLARACIÓN Y AUTORIZACIÓN

Yo, **NELSON PATRICIO SUÁREZ MONTES**, CI 1724245145 autor del trabajo de titulación: **Diseño y construcción de un sistema móvil autónomo para el control de malezas** previo a la obtención del título de **INGENIERO MECATRÓNICO** en la Universidad UTE.

1. Declaro tener pleno conocimiento de la obligación que tienen las Instituciones de Educación Superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación de grado para que sea integrado al Sistema Nacional de información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.
2. Autorizo a la BIBLIOTECA de la Universidad UTE a tener una copia del referido trabajo de titulación de grado con el propósito de generar un Repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Quito, 21 de marzo de 2023

f:



SUÁREZ MONTES NELSON PATRICIO
1724245145

CERTIFICACIÓN DEL TUTOR

En mi calidad de tutor de tesis de grado, certifico que el presente trabajo que lleva por título **Diseño y construcción de un sistema móvil autónomo para el control de malezas** para aspirar al título de **Ingeniero Mecatrónico** fue desarrollado por **Suárez Montes Nelson Patricio**, bajo mi dirección y supervisión, en la Facultad de Ciencias de la Ingeniería e Industrias; y que dicho trabajo cumple con las condiciones requeridas para ser sometido a la presentación pública y evaluación por parte del Jurado examinador que se designe.

Raúl Vicente Paredes Loor
DIRECTOR DEL TRABAJO
C.I. 1721986790

DECLARACION JURAMENTADA DEL AUTOR

Yo, Nelson Patricio Suárez Montes, portador de la cédula de identidad N° 1724245145, declaro que el trabajo aquí descrito es de mi autoría, que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en ese documento.

La Universidad UTE puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.



f:

SUÁREZ MONTES NELSON PATRICIO

1724245145

CARTA DE CONFORMIDAD DE LA INSTITUCIÓN/EMPRESA AUSPICIANTE



QUITO-ECUADOR ABANISERCONEX CIA LTDA

Quito, 3 de agosto de 2021.

Sr. PhD
Vladimir Bonilla
Director de la Carrera de Mecatrónica
Universidad UTE
Presente

De mi consideración:

Por medio del presente, informo el interés y compromiso que tiene la compañía ABANISERCONEX en auspiciar el desarrollo del trabajo de titulación "Diseño y construcción de un sistema móvil autónomo para control de malezas", que será desarrollado por el señor Suárez Montes Nelson Patricio, Cédula: 1724245145.

Aprovecho la oportunidad para reiterar mis sentimientos de consideración y estima.

Atentamente,

Ing. Rodrigo Venegas
Secretario de la Compañía



DEDICATORIA

A Dios por brindarme la sabiduría y entendimiento para lograr con éxito este proyecto.

Le dedico este resultado a mis padres Consuelo Montes y José Suárez por darme su apoyo incondicional y enseñarme que ante la adversidad no hay que rendirse sino levantarse y aprender de los errores.

A mi hermana Paola Suárez que siempre ha sido mi apoyo y consejo cuando más lo he necesitado.

A mi esposa Victoria Sánchez compañera de vida quien ha sido el pilar fundamental en este proyecto que con su amor, consejo y reflexión me ayudó a alcanzar la finalización del mismo.

A mis abuelitos María Santacruz, Delia Peñafiel y Rubén Montes (†) que por sus enseñanzas y cariño los llevo conmigo siempre.

AGRADECIMIENTOS

Agradezco a la Universidad UTE, a la carrera de Ingeniería Mecatrónica y a mis docentes quienes compartieron su sabiduría y experiencia para mi formación profesional.

A mi docente tutor Raúl Paredes por su confianza y asesoramiento en la realización del presente proyecto.

A Rubén Montes, José Sánchez, Bolívar Suárez y Edison S. por sus valiosos aportes en el desarrollo de este proyecto.

A mis amigos que me brindaron su ánimo, alegría y apoyo durante la carrera universitaria.

ÍNDICE DE CONTENIDOS

PÁGINA

RESUMEN	1
ABSTRACT	2
1. INTRODUCCIÓN	3
2. METODOLOGÍA	10
2.1 REQUERIMIENTOS	11
2.2 DISEÑO CONCEPTUAL	13
2.3 DISEÑO ESPECÍFICO	14
2.3.1 DISEÑO MECÁNICO	14
2.3.1.1 Diseño del chasis del robot	14
2.3.1.2 Mecanismo de tracción	15
2.3.1.3 Mecanismo de suspensión	17
2.3.1.4 Mecanismo de eliminación de maleza	18
2.3.1.5 Ensamble de la estructura	20
2.3.2 DISEÑO ELECTRÓNICO	21
2.3.2.1 Selección de motores	21
2.3.2.2 Selección de driver para los motores	27
2.3.2.3 Selección del microcontrolador	28
2.3.2.4 Fuente de alimentación	28
2.3.2.5 Convertidor de voltaje step down	29
2.3.2.6 Pantalla oled	29
2.3.2.7 Integración de componentes electrónicos	30
2.3.3 SISTEMAS DE CONTROL	31
2.3.3.1 Diseño del controlador de los motores	31
2.3.3.2 Identificación del modelo del motor	34
2.3.3.3 Control PID de los motores	43
2.3.4 CINEMÁTICA DIFERENCIAL	51
2.3.4.1 Modelo cinemático con punto desplazado	55
2.3.5 DISEÑO DE ALGORITMOS DE CONTROL	57
2.3.5.1 Teoría de estabilidad de Lyapunov	58
2.3.6 CONTROLADOR POR VISIÓN ARTIFICIAL	60

2.3.6.1 Adquisición del conjunto de imágenes.....	60
2.3.6.2 Controlador por detector de color	61
2.3.6.3 Detección de color	64
2.3.6.4 Entrenamiento en Haar-Cascade	66
2.3.7 PROGRAMACIÓN.....	72
3. RESULTADOS Y DISCUSIÓN.....	74
3.1 RESULTADOS DEL DISEÑO DEL ROBOT	74
3.2 RESULTADOS DEL CONTROLADOR DE MOTOR PID LAMBDA	75
3.3 RESULTADOS CONTROLADOR DEL ROBOT	76
3.4 RESULTADOS DEL SISTEMA DE CONTROL POR VISIÓN ARTIFICIAL	78
4. CONCLUSIONES Y RECOMENDACIONES	82
BIBLIOGRAFÍA	84
ANEXOS.....	88

ÍNDICE DE TABLAS

PÁGINA

Tabla 1. Lista de componentes empleados en el ensamblaje de la estructura.	20
Tabla 2. Tabla de verdad de las señales A y B del encoder de cuadratura.	32
Tabla 3. Parámetros del controlador PI Ziegler Nichols - Motor Derecho....	44
Tabla 4. Parámetros del controlador PID Ziegler Nichols - Motor Derecho .	44
Tabla 5. Parámetros del controlador PI Ziegler Nichols - Motor Izquierdo ..	45
Tabla 6. Parámetros del controlador PID Ziegler Nichols - Motor Izquierdo	46
Tabla 7. Parámetros del controlador PID Lambda Motor derecho.....	47
Tabla 8. Parámetros del controlador PID Lambda Motor izquierdo	49
Tabla 9. Referencia de rango de colores en HSV	65
Tabla 10. Lista de componentes que conforman el sistema móvil autónomo eliminador de malezas.	74
Tabla 11. Matriz de confusión del detector de color.	79
Tabla 12. Matriz de confusión de Haar Cascade.....	79
Tabla 13. Comportamiento de variables de la matriz de confusión.	80
Tabla 14. Valores porcentuales de las métricas a partir de la matriz de confusión.	81

ÍNDICE DE FIGURAS

	PÁGINA
Figura 1. Robot con sistema tipo oruga modelo TRI-TRACK.....	6
Figura 2. Esquema general de la organización de un robot.....	6
Figura 3. Robot de campo equipado con manipulador de 5 grados de libertad y efector micro-dosificación de aplicación directa.....	8
Figura 4. Plataforma de robot agrícola.....	8
Figura 5. Robot agrícola Deméter.....	9
Figura 6. Metodología de modelo en V aplicada al diseño y construcción de un sistema móvil autónomo para el control de malezas.	10
Figura 7. Diagrama de requerimientos generales de sistema móvil autónomo para el control de malezas.	11
Figura 8. Diagrama de requerimientos de visión artificial y herramienta de un sistema móvil autónomo para el control de malezas.	12
Figura 9. Diseño conceptual del sistema autónomo para control de malezas.	13
Figura 10. Diseño del chasis del sistema autónomo para control de malezas.	14
Figura 11. Diseño de la rueda motriz para el sistema de tracción.....	15
Figura 12. Diseño de rueda conducida.	16
Figura 13. Diseño de la cadena de tracción.....	16
Figura 14. Disposición trapezoidal de ruedas oruga.	17
Figura 15. Balancín de suspensión.	17
Figura 16. Resorte de tensión.	18
Figura 17. Herramienta eliminadora de malezas.....	18
Figura 18. Elementos superiores del mecanismo de eliminación de maleza.	19
Figura 19. Elementos inferiores del mecanismo de eliminación de maleza.	19
Figura 20. Ensamblaje total de la estructura.	20
Figura 21. Caja para componentes electrónicos y soporte de cámara para herramienta.....	21
Figura 22. Diagrama de cuerpo libre del móvil diferencial	21
Figura 23 Integración del modelo cinemático del motor con el modelo dinámico de los motores para obtener torque requerido por el robot móvil diferencial.	24
Figura 24. Simulación de trayectoria realizada por el robot móvil para llegar al punto deseado.	25
Figura 25. Torque requerido por la rueda derecha para llegar al punto deseado.....	25
Figura 26. Torque requerido por la rueda izquierda para llegar al punto deseado.....	26
Figura 27. Características de los motores comerciales.....	26

Figura 28. Motor DC 12V con encoder de efecto Hall	27
Figura 29. Driver L298N puente H.	27
Figura 30. Microcontrolador ESP32.	28
Figura 31. Batería seca recargable	29
Figura 32. Convertidor de voltaje LM2596 Step Down	29
Figura 33. Pantalla Oled	29
Figura 34. Diagrama esquemático de los componentes electrónicos del robot	30
Figura 35. Representación de un encoder magnético.....	31
Figura 36. Señales de los canales A y B de un encoder de cuadratura	31
Figura 37. Lectura de velocidad del motor por comunicación serial en Simulink.	35
Figura 38. Respuesta al escalón en lazo abierto - Motor derecho.	36
Figura 39. Importación de datos a la herramienta System Identification Matlab	36
Figura 40. Entrada y salida de la función de transferencia	37
Figura 41. Estimación de los valores de la función de transferencia mediante Process Models	37
Figura 42. Gráfica de la función de transferencia estimada.	38
Figura 43. Función de transferencia con retardo del motor derecho.....	38
Figura 44. Comparación de la respuesta del motor con su función de transferencia	39
Figura 45. Gráficas obtenidas de la comparación de la respuesta del motor con su función de transferencia	39
Figura 46. Estimación de los valores de la función de transferencia en motor izquierdo mediante Process Models	40
Figura 47. Respuesta al escalón en lazo abierto - Motor izquierdo.....	40
Figura 48. Gráfica de la función de transferencia estimada	41
Figura 49. Función de transferencia con retardo del motor izquierdo.....	41
Figura 50. Comparación de la respuesta del motor izquierdo con su función de transferencia	42
Figura 51. Gráficas obtenidas de la comparación de la respuesta del motor izquierdo con su función de transferencia.....	42
Figura 52. Control en lazo cerrado.....	43
Figura 53. Respuesta control PI y PID Ziegler Nichols FT Motor derecho ..	45
Figura 54. Respuesta control PI y PID Ziegler Nichols FT Motor izquierdo ..	46
Figura 55. Respuesta al escalón PID Lambda Motor derecho	48
Figura 56. Respuesta a la trayectoria PID Lambda Motor derecho.....	48
Figura 57. Respuesta control PID Lambda Sintonía Fina - Motor derecho .	49
Figura 58. Respuesta al escalón PID Lambda Motor izquierdo	50
Figura 59. Respuesta a la trayectoria PID Lambda Motor izquierdo	50
Figura 60. Respuesta control PID Lambda Sintonía Fina - Motor izquierdo .	51
Figura 61. Modelo cinemático de un robot móvil diferencial	52
Figura 62. Desplazamiento y orientación robot móvil diferencial	53

Figura 63. Modelo cinemático con punto de control desplazado	55
Figura 64. Ejemplo del proceso de adquisición de una imagen digital	60
Figura 65. Cámara ESP32Cam.....	61
Figura 66. Thresholding	61
Figura 67. Transformaciones Morfológicas	62
Figura 68. Contornos	63
Figura 69. Momentos	64
Figura 70. Espacio de color HSV	64
Figura 71. Detección del color verde en espacio HSV	65
Figura 72. Imágenes positivas para entrenamiento en Haar-Cascade	66
Figura 73. Imágenes negativas para entrenamiento en Haar-Cascade	66
Figura 74. Procesamiento de imágenes positivas y negativas.....	67
Figura 75. Almacenamiento de imágenes positivas y negativas en Haar-Cascade.....	67
Figura 76. Configuración de ajustes en la pestaña Train de Cascade Trainer GUI	68
Figura 77. Configuración de ajustes en la pestaña Common de Cascade Trainer GUI	69
Figura 78. Configuración de ajustes en la pestaña Cascade de Cascade Trainer GUI	69
Figura 79. Configuración de ajustes en la pestaña Boost de Cascade Trainer GUI	70
Figura 80. Entrenamiento del programa Haar Cascade	70
Figura 81. Diagrama de flujo del microcontrolador Esp32 programado en Arduino IDE.....	72
Figura 82. Diagrama de flujo del control por visión artificial mediante los métodos de detección de color y Haar-Cascade	73
Figura 83. Prototipo final del sistema móvil autónomo eliminador de malezas.	74
Figura 84. Comparativa entre los controladores PI, PID Ziegler Nichols y PID Lambda.....	76
Figura 85. Simulación del desplazamiento del robot diferencial.....	76
Figura 86. Gráficas de la velocidad lineal y angular.....	77
Figura 87. Gráfica de error de posición.....	77
Figura 88. Detector de objeto por color.....	78
Figura 89. Gráfica de las métricas obtenidas a partir de la matriz de confusión.	81

ÍNDICE DE ANEXOS

	PÁGINA
ANEXO 1. PROCESO DEL CORTE LASER	90
ANEXO 2. PROCESO DE SOLDADURA DEL CHASIS	91
ANEXO 3. DETALLE DE LA PANTALLA OLED	92
ANEXO 4. PRUEBAS DE CAMPO	93
ANEXO 5. FUNCIONAMIENTO MODO BUSQUEDA	94
ANEXO 6. FUNCIONAMIENTO MODO DETECTOR.....	94
ANEXO 7. FUNCIONAMIENTO MODO CORTE	95
ANEXO 8. FUNCIONAMIENTO MODO GIRO IZQUIERDA	95
ANEXO 9. FUNCIONAMIENTO MODO GIRO DERECHA	96
ANEXO 10. CÓDIGO QR GIRO IZQUIERDA.....	96
ANEXO 11. CÓDIGO QR GIRO DERECHA.....	97
ANEXO 12. PLANOS DEL ROBOT MOVÍL	97

RESUMEN

El presente proyecto de investigación muestra el diseño y construcción de un sistema móvil autónomo para el control de malezas el cual se lo propone como una alternativa a las técnicas usadas tradicionalmente en la agricultura que afectan tanto al cultivo como al agricultor, para ello se implementó una estructura con un sistema de tracción tipo oruga el cual le permite una mayor superficie de contacto logrando así un desplazamiento adecuado en suelo irregular, la eliminación de la maleza se la realizó mediante el empleo de una herramienta de un grado de libertad cuya función es cortar la maleza que crece alrededor de los cultivos, esto se logra con un algoritmo desarrollado en el lenguaje de programación Python que permite el reconocimiento de las plantas de cultivo mediante la adquisición de imágenes e implementación de visión artificial, para la detección de objetos se usa dos estrategias diferentes, la primera segmenta objetos por medio de la identificación de su color y la segunda es una red neuronal llamada Haar-Cascade que fue entrenada para el reconocimiento de las plantas de interés en este caso lechugas, la primera estrategia obtuvo un porcentaje de exactitud y precisión del 93% y 96% respectivamente y la segunda del 84% y 92.2%, cuando el algoritmo detecta el objeto también determina el centroide del mismo considerando a este punto como una posición final o deseada a la cual el robot tiene que desplazarse, esto se lo consigue por intermedio de la estrategia de control que utiliza el modelo cinemático del robot móvil diferencial, matriz Jacobiana, y teoría de Lyapunov, esta estrategia se encarga de determinar el conjunto de velocidades que son necesarias para trasladarse hacia el punto deseado controlando las velocidades de los motores del robot por medio del protocolo de comunicación Bluetooth con el microcontrolador ESP32 embebido.

Palabras Clave: Python, visión artificial, Haar-Cascade, robot diferencial, Jacobiano, Lyapunov.

ABSTRACT

This research project shows the design and construction of an autonomous mobile system for weed control which is proposed as an alternative to the techniques traditionally used in agriculture that affect both the crop and the farmer, for this a structure was implemented with a traction system type caterpillar which allows a greater contact surface thus achieving an adequate displacement on uneven ground, the elimination of weeds was done by using a tool with a degree of freedom whose function is to cut the weeds growing around crops, This is achieved with an algorithm developed in the Python programming language that allows the recognition of crop plants through the acquisition of images and implementation of artificial vision, for the detection of objects two different strategies are used, the first segments objects by identifying their color and the second is a neural network called Haar-Cascade that was trained to recognize the plants of interest in this case lettuce, the first strategy obtained a percentage of accuracy and precision of 93% and 96% respectively and the second of 84% and 92. 2%, when the algorithm detects the object it also determines the centroid of the object considering this point as a final or desired position to which the robot has to move, this is achieved through the control strategy that uses the kinematic model of the differential mobile robot, Jacobian matrix, and Lyapunov theory, this strategy is responsible for determining the set of speeds that are necessary to move to the desired point by controlling the speeds of the robot motors through the Bluetooth communication protocol with the embedded ESP32 microcontroller.

Keywords:

Python, computer vision, Haar-Cascade, differential robot, Jacobian, Lyapunov.

1. INTRODUCCIÓN

Uno de los problemas que afecta de manera significativa a los cultivos es la maleza con la cual compiten por nutrientes, luz y humedad necesaria para su crecimiento (Perreira, Alves, & Peñaherrera, 2011), dando lugar a plantas cuyas estructuras se presentan enfermas, débiles y con poca capacidad de crecimiento de estructuras vegetativas, es por ello que su control es uno de los procesos de mayor importancia en la producción agrícola.

El problema de la presencia de malezas en cultivos como la lechuga es que además de causar problemas que afectan a su crecimiento y desarrollo, también sirven como huéspedes de plagas de áfidos, gusano de fruto, nematodos y minadores causando enfermedades como la mancha bacteriana (Theodoropoulos, Lardizabal, & Arias, 2009).

Desde la antigüedad el proceso de remoción de malezas ha sido una actividad realizada de forma manual por los agricultores en sus cultivos. Con el pasar del tiempo, esta labor se fue mecanizando, sin embargo, en 1950 con la aparición de los herbicidas para su control y en la década de los 60 con el desarrollo de herbicidas penetrantes a través del suelo se priorizó el uso de estas sustancias (Encinas, 2020).

Como consecuencia se ha desencadenado el deterioro ambiental acelerado y de salud por su excesiva aplicación ya que al ser compuestos de síntesis química se acumulan de manera permanente en los organismos vivos (Fernández, 1982). En el aspecto ambiental el intensivo uso de un solo herbicida como lo menciona Vidal, et al. (2010) causa el incremento de la densidad de especies que desarrollan tolerancia a este producto y a su vez su aplicación favorece de cierta manera la evolución de especies resistentes.

En Ecuador, un país de amplia biodiversidad donde una de sus principales actividades económicas es la agricultura, el método más utilizado para la remoción de malezas en cultivos son los herbicidas ya que de acuerdo a la información emitida por INEC, (2016) en nuestro territorio de la superficie sembrada con cultivos permanentes y con cultivos transitorios; en el 50,03% y en el 78,24% respectivamente se han aplicado insumos de origen químico como son los pesticidas y herbicidas.

Para disminuir el uso de agroquímicos como principal método de control de malezas es necesario que se desarrollen herramientas tecnológicas que además de no causar impactos negativos en el ambiente también sean eficaces al momento de su empleo en cultivos mejorando su calidad e incrementando sus niveles de productividad.

Con la invención de nuevas tecnologías como los robots, es cada vez más común observar su integración dentro del área agrícola como lo es la implementación de sistemas de precisión mecánica que tienen la finalidad de efectuar un control efectivo de maleza detectándola de manera puntual sin

afectar a las plantas ni a su sustrato de crecimiento, el suelo (Sogamoso, Aponte, & Solaque, 2015).

Moreano, Cajamarca, & Tenicota Garcia, (2020) consideran a la agricultura de precisión como el conjunto de tecnologías que permiten la aplicación de insumos agrícolas de forma variable en superficies de terreno. Su desarrollo como lo indica Godwin, et al. (2001) inicia desde su aparición en los años 80 con modelos de estructura limitada y logra evolucionar hasta el año 2002 en Estados Unidos donde se construyen modelos más complejos con la idea de ahorrar recursos en la aplicación de insumos químicos, mientras que en Latinoamérica la aplicación de tecnologías que ayudan al incremento de producción agrícola aparece a partir del año 2006 en países como Argentina, Chile, Uruguay y Bolivia.

En sus inicios, este tipo de tecnología se basaba únicamente en la adquisición de información, análisis y aplicación de insumos; sin embargo, en la actualidad su utilidad se ha extendido a diversas áreas como el análisis de suelos, labrado selectivo, siembra automatizada y cosecha asistida (Moreano & Ribeiro, 2017).

De acuerdo a Guato (2019), este tipo de agricultura se puede desarrollar en varias etapas como son la adquisición de datos tanto del cultivo como de su entorno, posterior registro de los mismos, la toma de decisiones y la actuación en campo para la aplicación de los recursos que sean considerados necesarios sin una afectación ambiental de por medio.

La implementación de sistemas mecánicos automatizados ha tomado fuerza ante la toma de conciencia ambiental global ya que integran agricultura de precisión y al mismo tiempo disminuyen el uso de agroquímicos debido a que se encuentran conformados por sistemas con la capacidad de medir las propiedades físicas del cultivo además de tener capacidad de decisión y procesamiento de información recolectada así como actuadores que permitan realizar la tarea de remoción de maleza (Sogamoso, Aponte, & Solaque, 2015).

Los robots móviles destinados a tareas agrícolas deben cumplir determinados requerimientos de movilidad tomando siempre en cuenta las necesidades que se tengan en el campo agrícola para lo cual se debe tener presente que el desplazamiento en estos ambientes requiere de una planificación de rutas, de su movimiento y de la orientación automática que tenga el vehículo (Vougioukas, 2019). Es por ello que para poder realizar operaciones de navegación del vehículo es necesario que este cuente con una serie de sensores que sirvan como fuentes de información para la toma de decisiones y en las operaciones (Bechar & Vigneault, 2016).

En las operaciones de control de maleza dirigidas por robots autónomos Kanagasingham, Ekpanyapong, & Chaihan consideran la necesidad de

mantener una guía de manera precisa a través de las filas de cultivo así como la capacidad de identificar el final de una fila y el inicio de otra con la realización de un giro y cambio de fila, para lo cual emplearon en su investigación un sistema guiado por GPS-RTK y enrutamiento en base a una brújula con el objetivo de tener una guía para los giros de entrada en una nueva fila siempre que las condiciones ambientales como la nubosidad y porcentaje de humedad para el sistema sean óptimas.

Para lograr la eliminación de maleza de una forma precisa lo es importante que se realice su detección lo que permitirá el movimiento de los actuadores (Osorio, et al. 2020). Es por ello que la visión artificial por computadora en robots móviles es la más utilizada en esta tarea ya que se encarga de detectar y clasificar objetos de manera más satisfactoria en comparación de aquellos robots que utilizan sistemas GPS de alta precisión en los cuales se emplean mapas de navegación lo cual representa una desventaja al presentarse como una tecnología mucho más costosa (Vásquez, 2020).

Un robot como lo define Vásquez (2020) es una máquina que comprende componentes eléctricos, electrónicos, mecánicos y de comunicaciones controlados por un sistema informático. Por otra parte, los robots móviles han sido diseñados para desarrollar funciones más complejas con el incremento de nuevos mecanismos, sensores y sistemas de control.

Los robots móviles autónomos destacan por la relación que existe entre sus operaciones de percepción y acción que poseen. Durante la percepción, es indispensable que el robot tenga una relación con su entorno mediante una síntesis de información que es proporcionada por los sensores, además que también debe tener la capacidad de tomar decisiones sobre acciones en base al entorno percibido para cumplir los objetivos previstos, a esta cualidad se la denomina razonamiento (Vásquez, 2020).

La gran ventaja que presenta la utilización de robots móviles en el campo agrícola como lo menciona García (2015) es la disminución a la limitación de un espacio fijo de trabajo, realizando tareas como la manipulación de cargas, remoción de escombros, recolección de frutos y remoción de malezas en cultivos. Es por ello, que se han desarrollado sistemas de tracción como el de ruedas cuyo movimiento no se ve limitado ante diferentes tareas.

Para la operación de desplazamiento en terrenos, uno de los sistemas de tracción con ruedas más utilizado por su eficacia es el sistema de tipo oruga equipado con pistas de deslizamiento para el movimiento. Posee orugas orientables en cada uno de sus lados las cuales le permiten al robot un ajuste a la forma del terreno, usualmente se encuentran unidas al chasis del robot por un eje ubicado en su centro para guiar a la oruga ante las irregularidades del terreno simulando las ruedas de maquinaria pesada como una retroexcavadora. Su estructura se aprecia en la Figura 1 (García, 2015).



Figura 1. Robot con sistema tipo oruga modelo TRI-TRACK.
(Echeverría, Guerra, & Quiñones, 2020)

El diseño de un robot móvil como se detalla en la Figura 2, está configurado por una serie de elementos como los actuadores, sensores y el sistema de control. Los actuadores son dispositivos con la capacidad de generar fuerza partiendo de la transformación de la energía (Dorador-Gonzalez, 2009). Se los puede clasificar en: neumáticos cuya fuente de energía es el aire a presión, hidráulicos cuya fuente de energía es un fluido y eléctricos cuya fuente de energía es la electricidad (Corona, Abarca, & Mares, 2014).

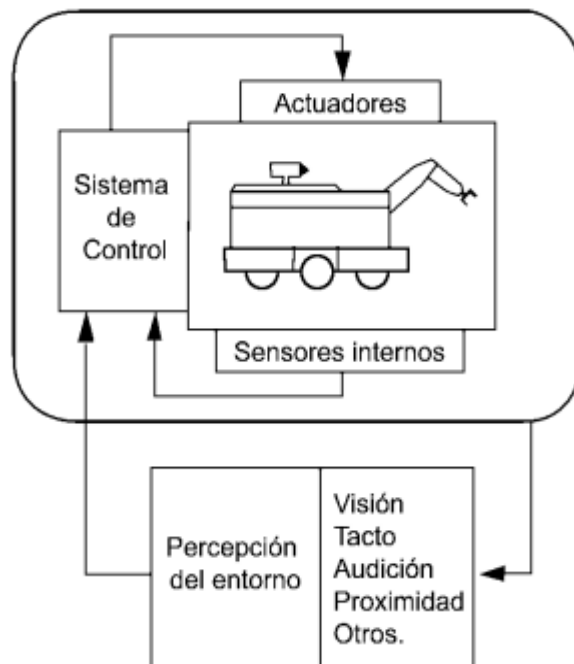


Figura 2. Esquema general de la organización de un robot.
(Vásquez, 2020)

Para la tarea de detección y eliminación de malezas de una manera precisa es necesario el empleo de sistemas de visión artificial en donde se utilizan

sensores que trabajan dentro de un espectro visible o un rango multispectral brindando la información necesaria (Espinoza; et al, 2021).

Los sistemas de visión artificial pueden ser directos, es decir que realizan su función en un tiempo real; como indirectos cuyo análisis de datos se realiza de manera posterior a la recolección de la información obtenida. Es precisamente la versatilidad que tienen estos sistemas lo que los hace ideales para una navegación automatizada y controlada en vehículos de tipo terrestre. Ambas clases tienen amplia aplicabilidad ya que al emplear visión por computador se obtiene una buena calidad de imagen o video, lo que permite su correcta interpretación (Russo; et al, 2017).

Para el análisis y comprensión de imágenes en visión artificial existe un proceso de obtención de imágenes, aprendizaje de la información, procesamiento e interpretación de resultados obtenidos (Moreira & Tobar, 2022).

Los problemas más comunes que se presentan durante el procesamiento de información en la detección y clasificación de objetos son la similitud en su color y textura entre el cultivo y la maleza que lo rodea, problemas de oclusión, así como la iluminación. Ante lo cual en la actualidad se emplean técnicas de alta eficacia como la morfología matemática que emplea el cálculo de los índices exceso de verde (ExG), exceso de rojo (ExR), la diferencia entre estos dos índices (ExGR) y el índice de vegetación de diferencia normalizada (NDVI) (Liu et al., 2020). De las técnicas antes mencionadas, la que ha tenido resultados más eficaces ha sido el empleo de redes neuronales artificiales puesto que utilizan algoritmos con una gran capacidad de aplicación (Hasan et al., 2021).

Como lo menciona (Viola & Jones 2001); una de las técnicas que destacan en el reconocimiento de objetos en una imagen o video separando formas determinadas en la construcción de árboles de clasificación es la conocida como el método de clasificación Haar Cascades el cual concentra varios clasificadores de los cuales cada uno analiza una porción específica de una imagen o frame en el caso de que se trate de un video. De esta manera en el momento de la clasificación se organiza la imagen de acuerdo al valor de píxeles que se encuentren dentro del espacio de detección cerrada.

En lo referente a prototipos diseñados para la remoción de malezas existen diversos modelos de acuerdo a las necesidades previstas, como por ejemplo, el robot con efector final de precisión que cuenta con una serie de estructuras que se pueden apreciar en la Figura 3 iniciando por un ordenador encargado de procesar imágenes a través de una cámara, un brazo con sierra como efector final para el corte de malezas además de bombas de dosificación para la aplicación directa de una sustancia química cuya finalidad es quemar los residuos de maleza. Está diseñado con una plataforma de ruedas direccionales para su desplazamiento, en su parte posterior contiene un

manipulador equipado con servomotores cuya finalidad es permitir la rotación de las articulaciones (Sogamoso, Aponte, & Solaque, 2015).



Figura 3. Robot de campo equipado con manipulador de 5 grados de libertad y efector micro-dosificación de aplicación directa. (Sogamoso, Aponte, & Solaque, 2015)

El sistema adecuado para plantaciones de maíz o algodón de Benavides, (2016) está conformado por un carro direccional, una cámara de navegación automática para identificación de maleza, así como un sistema de control. Debajo de la plataforma del vehículo se ubica un dispositivo de extracción mecánico con cuchillas que es accionado eléctricamente para su acción entre plantas, mientras que para surcos se dispone al costado como se puede observar en la Figura 4. El accionar de estos sistemas se da a través de cilindros y vástagos horizontales y verticales que se ubican en la parte posterior de la estructura lo que lo hace altamente eficiente pues tiene gran capacidad para eliminar todo tipo de maleza entre surcos y también entre cultivos.

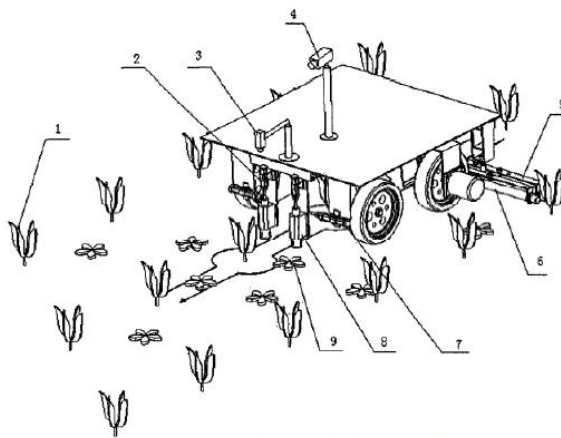


Figura 4. Plataforma de robot agrícola. (Benavides, 2016)

En modelos de robots con autonomía en su navegación local, la remoción de maleza se realiza en base al entrenamiento de un módulo PixyCam de Arduino donde un sensor realiza el reconocimiento de color que le permite ir en dirección recta siguiendo una línea central. Su estructura como se muestra en la Figura 5 está construida en acero, cuenta con motores de escobillas y transmisión conjunta de un reductor de velocidad de tornillo y cadena (Ardila, Uribe, & Meneses, 2022).

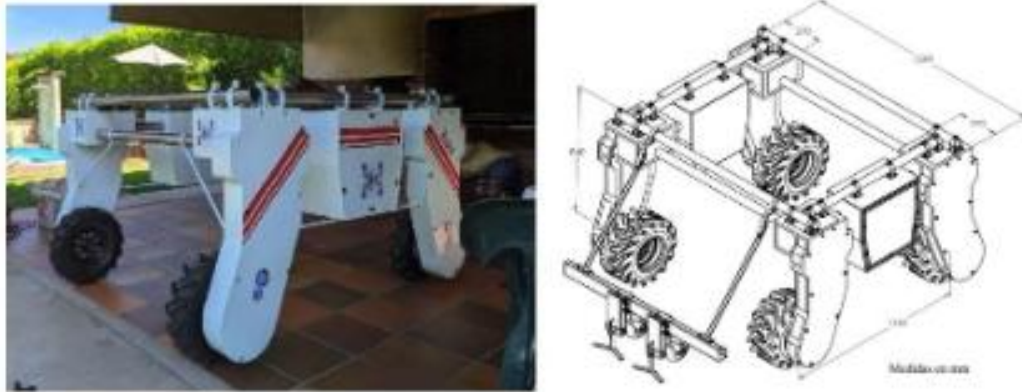


Figura 5. Robot agrícola Deméter.
(Ardila, Uribe, & Meneses, 2022)

Debido a que la remoción de malezas en los sembríos ecuatorianos se ha realizado durante varias décadas de manera manual y posteriormente con el uso de agroquímicos, el presente trabajo de investigación se presenta como una alternativa viable frente al deshierbe manual que usualmente le toma al agricultor gran parte de su jornada de producción así como la reducción de aplicación de sustancias de composición química ya que pretende diseñar y construir un sistema móvil utilizando sistemas de visión artificial para la navegación autónoma y control de malezas con un actuador de un grado de libertad, mediante el diseño e implementación de un modelo de la estructura mecánica del robot para lograr una navegación autónoma en condición de suelo irregular además de crear un sistema de control de dirección aplicando algoritmos de redes neuronales con visión artificial, diseñando un sistema para la detección de cultivos y la eliminación de malezas para posteriormente validar el funcionamiento del dispositivo mediante pruebas de campo.

2. METODOLOGÍA

El diseño y construcción de un sistema móvil autónomo para el control de malezas se realizó mediante la aplicación de la “Metodología de modelo en V” el cual es utilizado en el desarrollo de sistemas mecatrónicos como se detalla en la Figura 6.

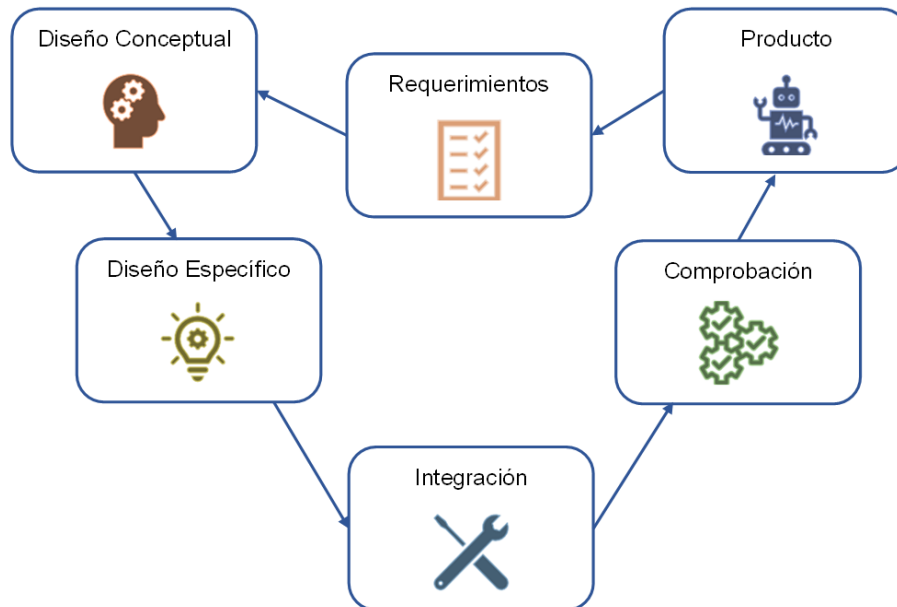


Figura 6. Metodología de modelo en V aplicada al diseño y construcción de un sistema móvil autónomo para el control de malezas.

Esta metodología contribuyó con el establecimiento de los requisitos del sistema entregados por el usuario que determinan el alcance y limitación que posee el mismo, después se continuó con el diseño conceptual.

Para poder realizar el diseño específico se inició con el diseño mecánico, es decir se definió la de forma del chasis y dimensiones óptimas para el sistema móvil, así como también las dimensiones de la herramienta para la eliminación de maleza, también se definió el tipo de tracción que debe poseer para desplazarse en un terreno irregular, luego se procedió con el diseño del controlador y elección de los componentes electrónicos.

El sistema mecánico, eléctrico, electrónico y de control se integraron posteriormente tras su desarrollo de manera individual donde se pudo apreciar la interacción entre ellos para realizar posteriores correcciones en errores encontrados tomando en cuenta los requerimientos y etapas del modelo en V.

Durante la fase de comprobación de hardware, software, modelo cinemático diferencial, control PID, modelo simulado y sistema físico se utilizaron pruebas de posicionamiento, seguimiento de trayectorias y visión artificial. De esta manera se obtuvo como producto final un robot móvil capaz de detectar un tipo de planta de cultivo y eliminar maleza a su alrededor.

2.1 REQUERIMIENTOS

Los requerimientos que se establecieron en el diseño y construcción de un sistema móvil autónomo para el control de malezas se detallan en un diagrama de requerimientos del sistema en la Figura 7.

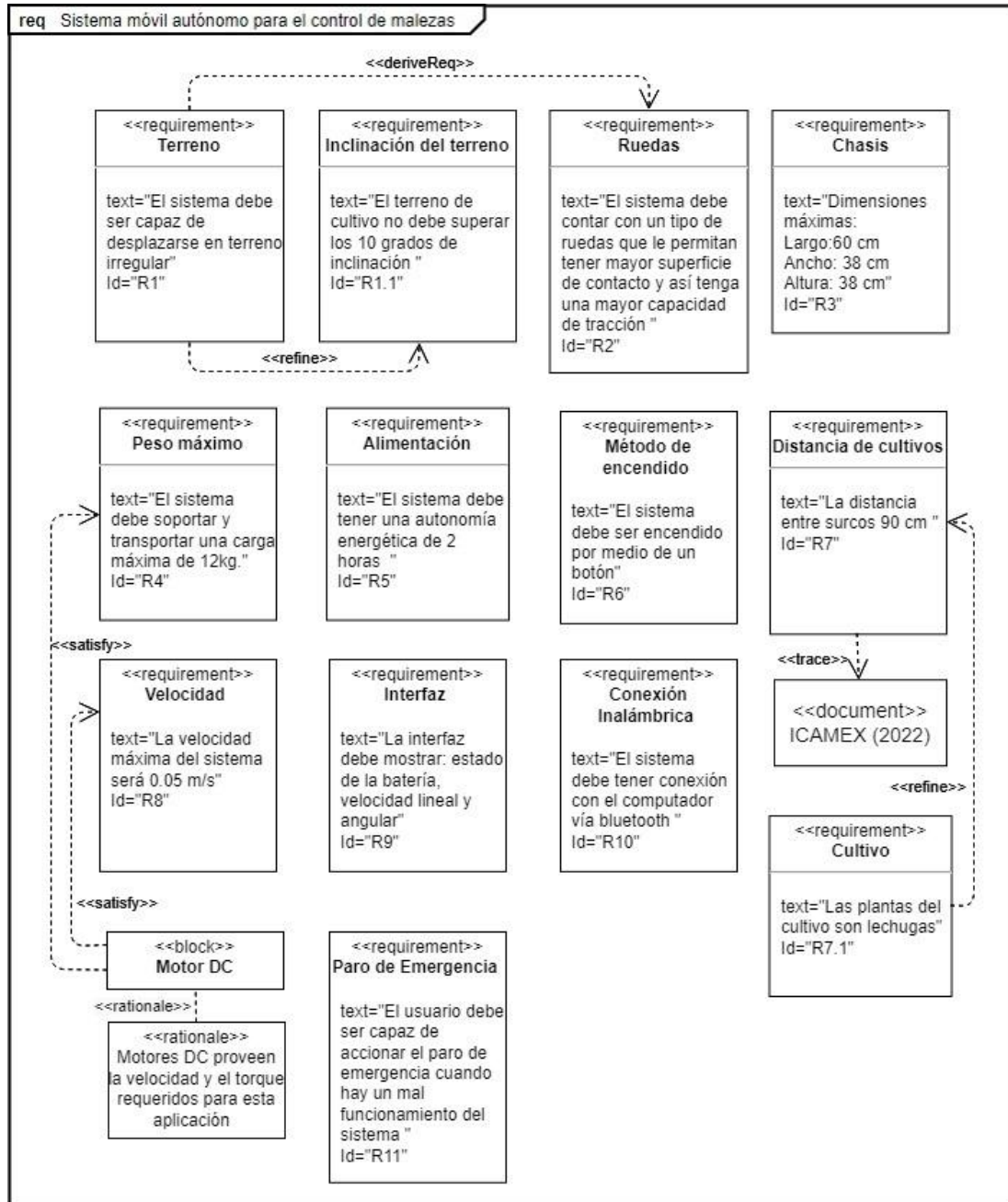


Figura 7. Diagrama de requerimientos generales de sistema móvil autónomo para el control de malezas.

En la Figura 8 se representan los aspectos relevantes a cumplirse sobre su funcionamiento.

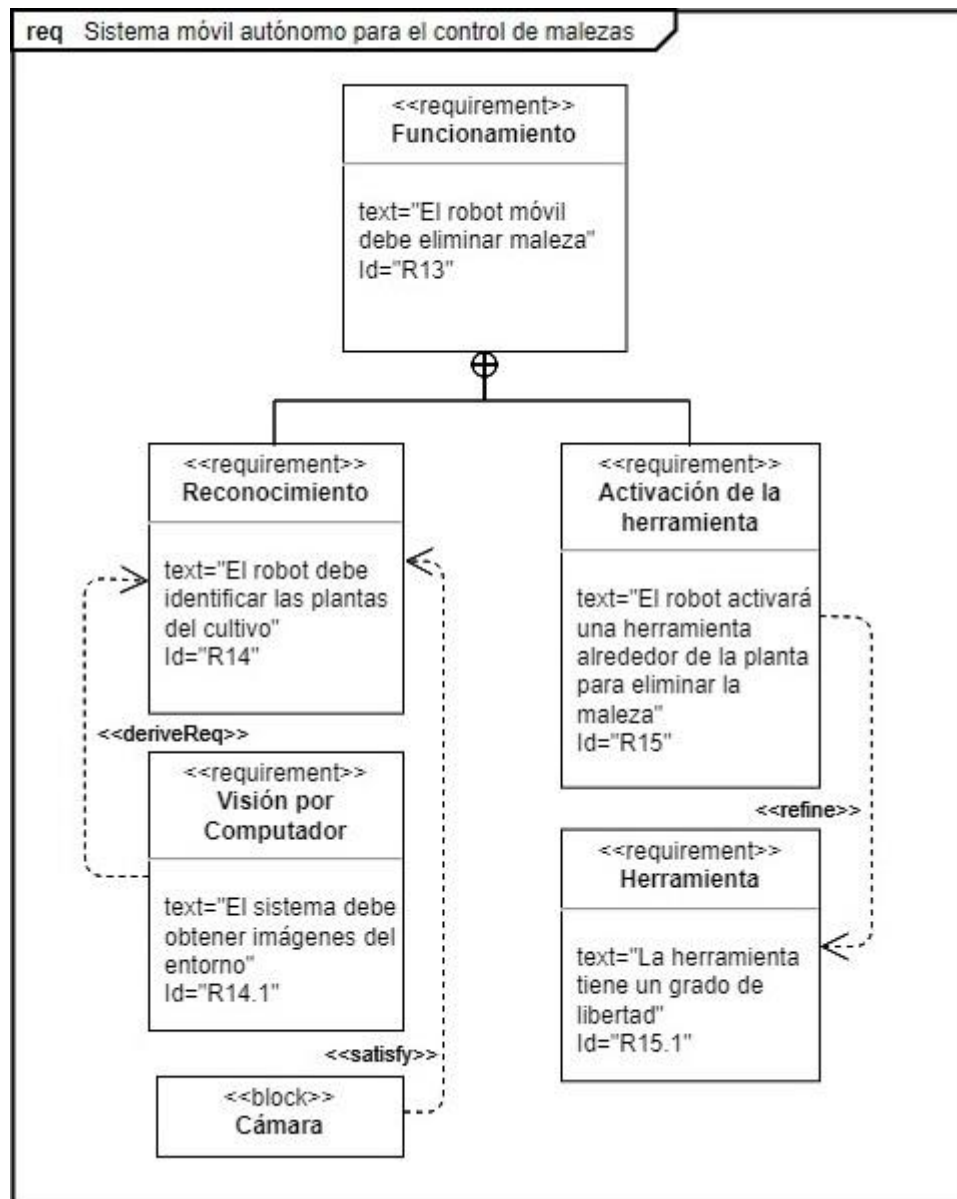


Figura 8. Diagrama de requerimientos de visión artificial y herramienta de un sistema móvil autónomo para el control de malezas.

2.2 DISEÑO CONCEPTUAL

En la Figura 9 se detalla el diseño conceptual del sistema autónomo para control de malezas el cual contiene la estructura general del robot. Está constituido por una estructura mecánica que consta de un mecanismo de tracción tipo oruga. También se detalla el sistema de control que está encargado de los drivers de los motores, activación de la herramienta para la eliminación de maleza y visión artificial.

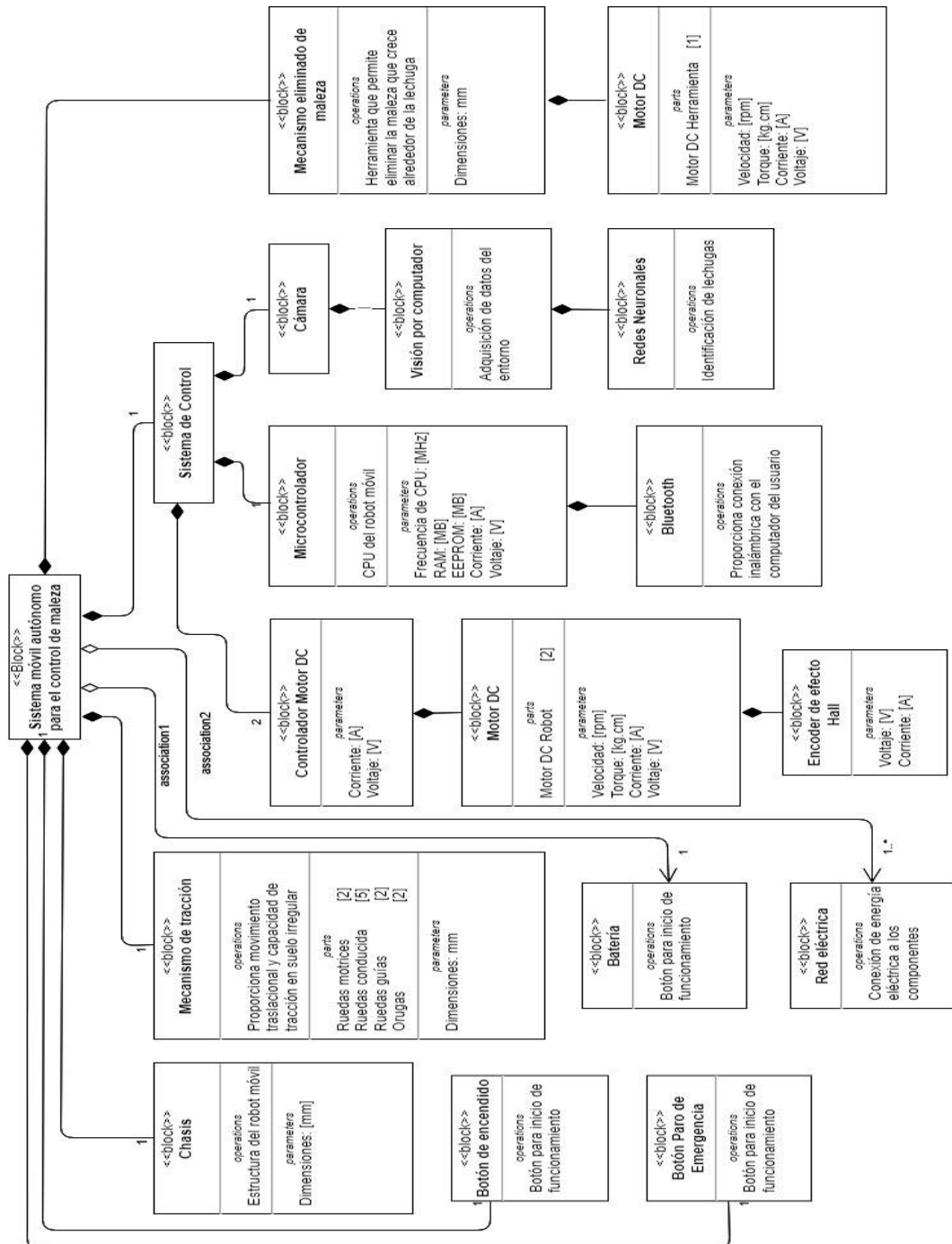


Figura 9. Diseño conceptual del sistema autónomo para control de malezas.

2.3 DISEÑO ESPECÍFICO

2.3.1 DISEÑO MECÁNICO

2.3.1.1 Diseño del chasis del robot

El robot eliminador de maleza se lo realizó tomando en cuenta que está orientado al servicio de la agricultura para lo cual se empleó el software CAD para la elaboración de su diseño.

En cuanto a las dimensiones para el diseño del chasis se tomaron en cuenta plantas de 20 centímetros de alto además de los parámetros de siembra de lechuga sugeridos por (ICAMEX, 2022) como lo son una distancia de siembra entre plantas de 30 centímetros y una distancia entre surcos de 90 centímetros.

Adicionalmente como se muestra en la Figura 10, se dimensionaron perforaciones donde van posicionados los elementos soporte del chasis, sistema de tracción tipo oruga como ruedas motrices, ruedas conducidas, ruedas guías, resortes de suspensión, motores y herramienta de eliminación de maleza.

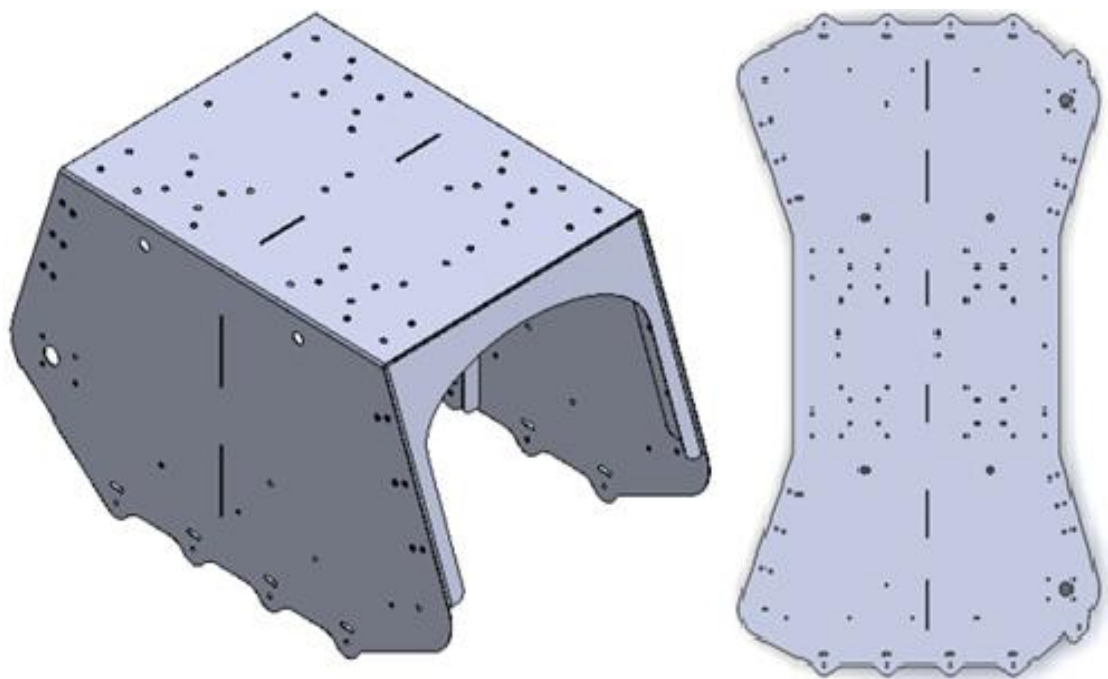


Figura 10. Diseño del chasis del sistema autónomo para control de malezas.

2.3.1.2 Mecanismo de tracción

Teniendo en cuenta que la superficie por la cual se desplaza el sistema móvil es irregular, el sistema de tracción más adecuado que se implementó fue el de orugas o bandas de rodamiento continuas por su mayor superficie de contacto con el suelo.

Este mecanismo de tracción tiene tres elementos que lo componen. El primero de ellos se lo denomina como rueda motriz o rueda conductora tipo engranaje la cual se muestra en la Figura 11, esta recibe a través del eje conductor la potencia del motor DC y por medio de sus engranajes transmite el movimiento angular mediante enganche hacia las ruedas oruga transformándose en desplazamiento lineal.



Figura 11. Diseño de la rueda motriz para el sistema de tracción.

Como segundo elemento se presentan las ruedas conducidas o guías cuya función es la de soportar el peso del robot, así como alinear y dar una guía en el desplazamiento de las orugas cuando entran en contacto con el suelo. Su superficie como se muestra en la Figura 12 es lisa.

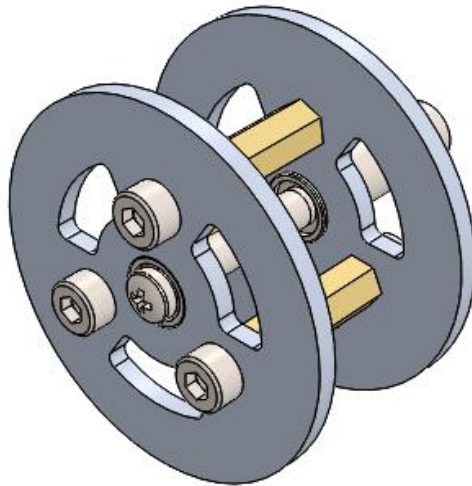


Figura 12. Diseño de rueda conducida.

El tercer elemento es la oruga o cadena de tracción que se indica en la Figura 13 la cual trabaja de manera conjunta con la rueda conducida soportando el peso del robot. Su función es la de proporcionar tracción al contacto con el suelo y así generar movimiento.

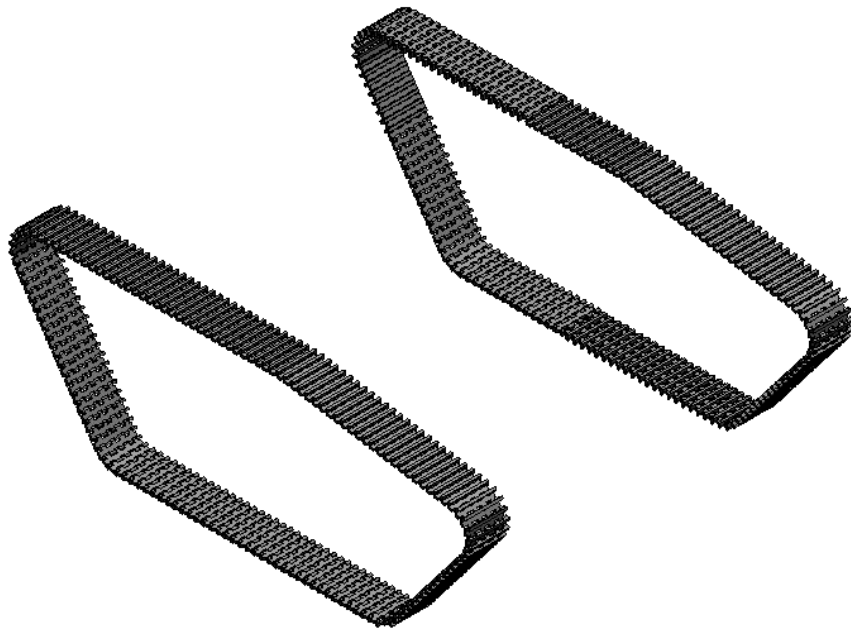


Figura 13. Diseño de la cadena de tracción.

La ubicación de las ruedas oruga se organizó en una configuración tipo trapecoidal escaleno que se caracteriza por tener una rueda conducida

delantera situada encima de las ruedas de apoyo como se puede observar en la Figura 14, transfiriendo parte del peso del robot a esa rueda.

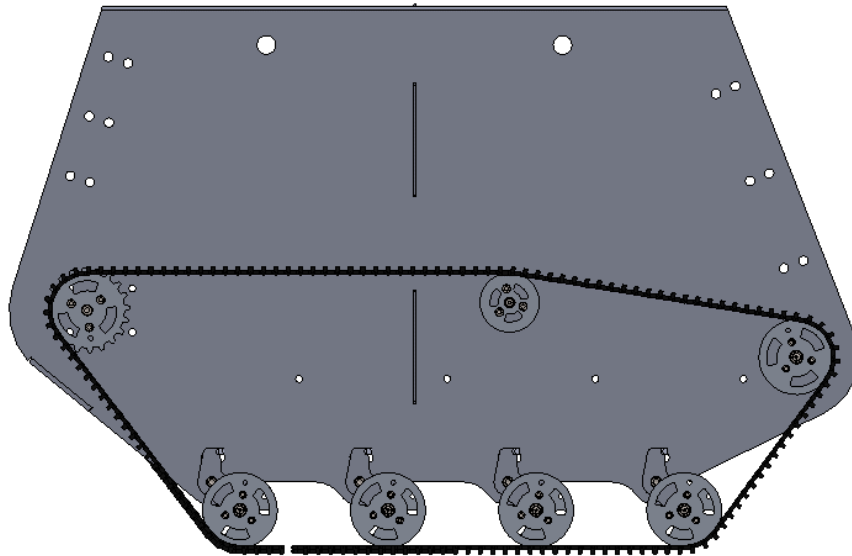


Figura 14. Disposición trapezoidal de ruedas oruga.

2.3.1.3 Mecanismo de suspensión

El mecanismo de suspensión que se utilizó estuvo conformado por dos elementos. El primero, un balancín de suspensión que se muestra en la Figura 15 el cual transmite el movimiento de la rueda conducida cuando esta atraviesa una irregularidad hacia el resorte de tensión situado en el otro extremo pivotando en su eje intermedio que se encuentra unido al chasis.

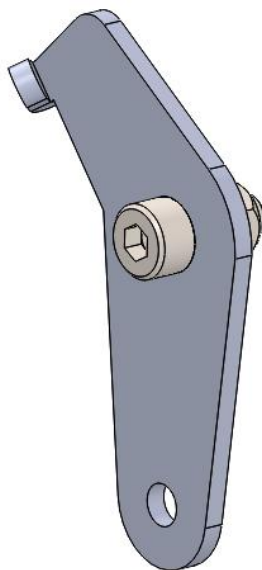


Figura 15. Balancín de suspensión.

El segundo y último elemento que conforma este mecanismo es el resorte de tensión que como se observa en la Figura 16, es una espiral metálica con ganchos o terminales laterales que ayudan a establecer una conexión con el balancín de suspensión.

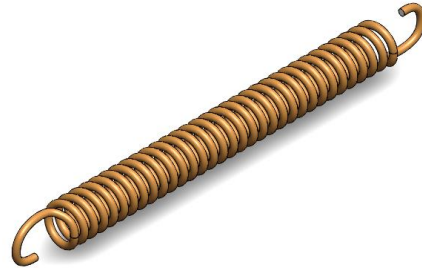


Figura 16. Resorte de tensión.

2.3.1.4 Mecanismo de eliminación de maleza

Se diseñó una herramienta para la eliminación de malezas cuya estructura se puede apreciar en la Figura 17 la cual se compone de cinco elementos.

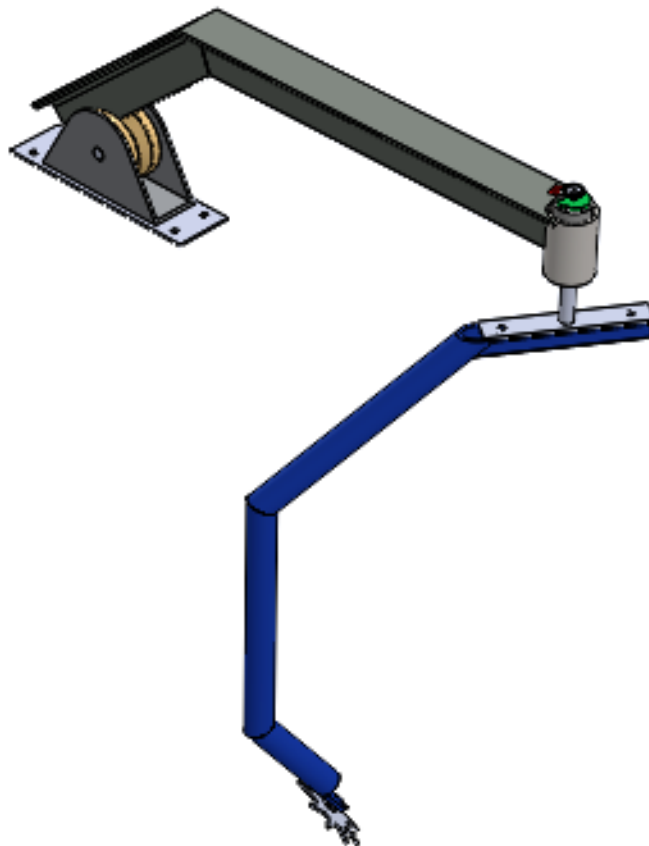


Figura 17. Herramienta eliminadora de malezas.

El primer elemento representado en la Figura 18 como (a) es un soporte que conecta la base de la herramienta con el segundo y tercer elemento de manera directa, el motor (b) encargado de dar el movimiento requerido para el funcionamiento integral que se encuentra fijado mediante pernos al elemento soporte de motor (c).

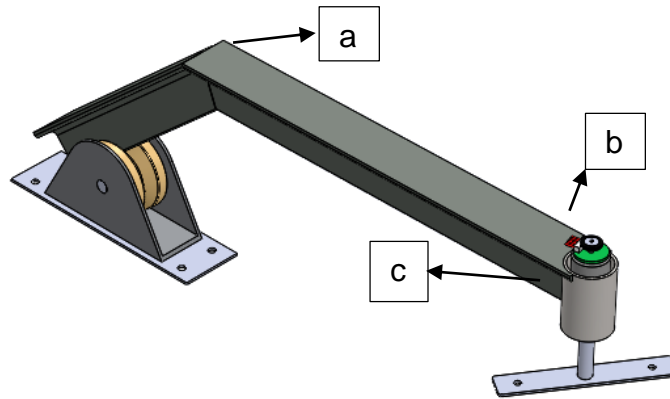


Figura 18. Elementos superiores del mecanismo de eliminación de maleza.

En la Figura 19 se encuentran el cuarto y quinto elemento correspondientes al acople del motor (d) que permite la unión del soporte de motor con el efector final (e) en una sola estructura.

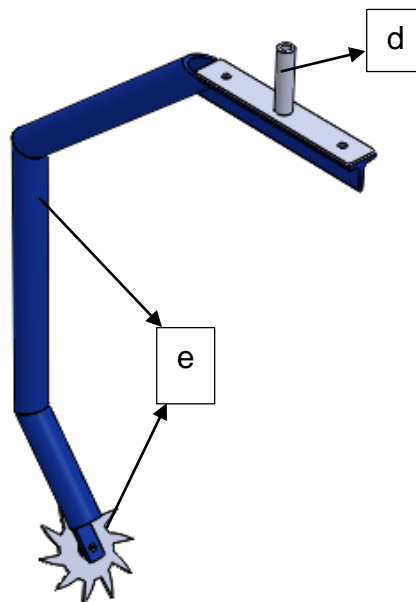


Figura 19. Elementos inferiores del mecanismo de eliminación de maleza.

2.3.1.5 Ensamble de la estructura

Los mecanismos y elementos mencionados anteriormente se los puede observar en la Figura 20 en el ensamble total de la estructura.

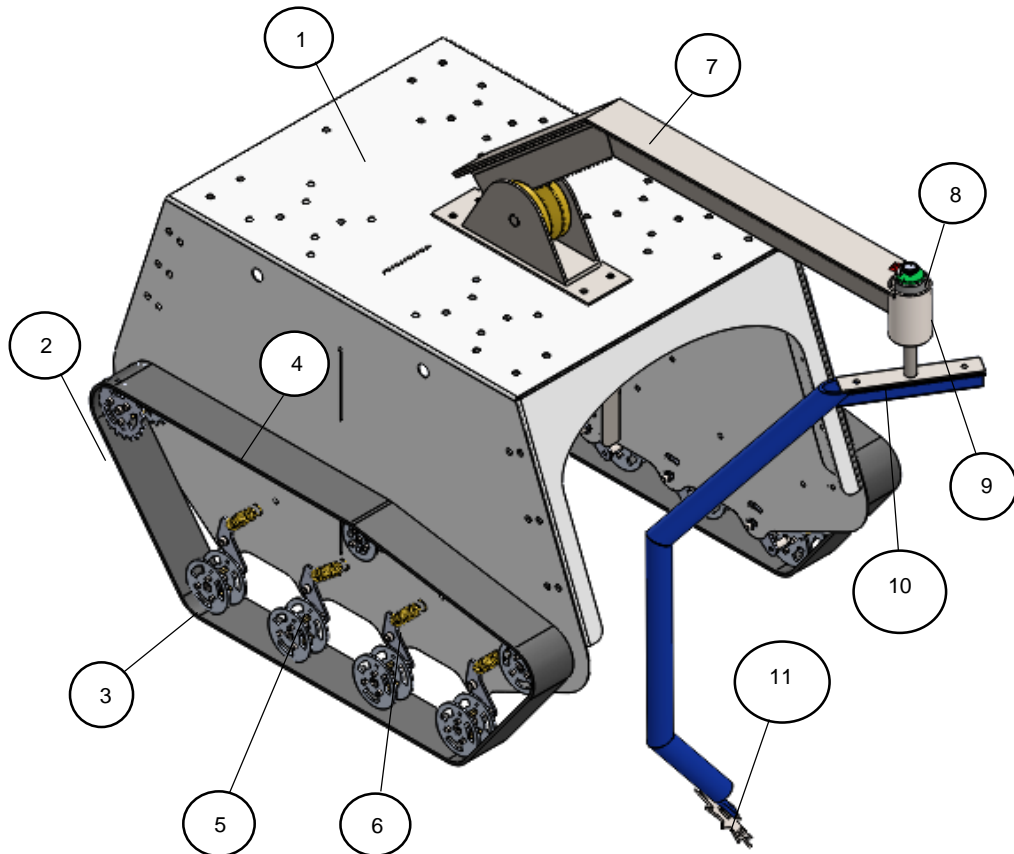


Figura 20. Ensamblaje total de la estructura.

En la Tabla 1 se detalla de forma textual y numerada los componentes empleados en el ensamblaje de la estructura.

Tabla 1. Lista de componentes empleados en el ensamblaje de la estructura.

N.º	Cantidad	Componente
1	1	Chasis
2	8	Rueda motriz
3	8	Rueda conducida
4	2	Oruga o cadena de tracción
5	8	Balancín de suspensión
6	8	Resorte de tensión
7	1	Soporte de herramienta
8	1	Motor
9	1	Soporte de motor
10	1	Acople de motor
11	1	Efactor final

Adicionalmente como se muestra en la Figura 21, se realizó el diseño de una caja para el almacenamiento de los componentes electrónicos, botones, pantalla de interfaz de usuario, así como también un soporte para fijar la cámara en la parte delantera de la herramienta.



Figura 21. Caja para componentes electrónicos y soporte de cámara para herramienta.

2.3.2 DISEÑO ELECTRÓNICO

2.3.2.1 Selección de motores

En cuanto a la elección de motores para el sistema se tomó en cuenta que debe mover la carga especificada en los requerimientos la cual fue de 12kg y se calculó la fuerza.

Datos del robot:

- Masa del robot $m = 12 [kg]$
- Radio de la rueda motriz $r = 25 [mm]$

A continuación, se muestra en la Figura 22 el diagrama de cuerpo libre y las ecuaciones de velocidad lineal y velocidad angular del móvil diferencial

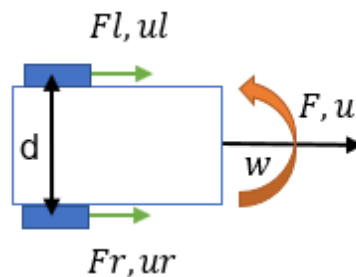


Figura 22. Diagrama de cuerpo libre del móvil diferencial

$$u = \frac{ur + ul}{2} = \frac{wr \cdot r + wl \cdot r}{2} \quad [1]$$

$$w = \frac{ur - ul}{d} \quad [2]$$

$$w = \frac{wr \cdot r - wl \cdot r}{d} \quad [3]$$

Donde:

u : Velocidad lineal global del móvil

ur : Velocidad lineal rueda derecha

ul : Velocidad lineal rueda izquierda

w : Velocidad angular global del móvil

wr : Velocidad angular rueda derecha

wl : Velocidad angular rueda izquierda

d : Distancia entre las ruedas

r : Radio de las ruedas del móvil

Las Ecuación 1 de velocidad lineal y Ecuación 2 de velocidad angular se trasladan a su equivalente matricial.

$$\begin{pmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{d} & -\frac{r}{d} \end{pmatrix} \cdot \begin{pmatrix} wr \\ wl \end{pmatrix} = \begin{pmatrix} u \\ w \end{pmatrix} \quad [4]$$

Se despeja las velocidades angulares de cada rueda

$$\begin{pmatrix} wr \\ wl \end{pmatrix} = \begin{pmatrix} u \\ w \end{pmatrix} \begin{pmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{d} & -\frac{r}{d} \end{pmatrix}^{-1} \quad [5]$$

Para obtener el torque de cada rueda se usa la siguiente ecuación

$$T = F \cdot r \quad [6]$$

Donde:

T : Es el torque de una rueda

F : Es la fuerza longitudinal ejercida en la rueda

r : Radio de las ruedas del móvil

La sumatoria de fuerzas es igual a la masa por la aceleración

$$F = Fl + Fr = m\dot{u} \quad [7]$$

$$\frac{Tl}{r} + \frac{Tr}{r} = m\dot{u} \quad [8]$$

La sumatoria de torques es igual a la inercia del móvil por la aceleración angular, el valor de la inercia se obtuvo de las propiedades físicas del ensamblaje que se realizó en el programa SolidWorks.

$$T = J\dot{\omega} \quad [9]$$

$$Fr \cdot \frac{d}{2} - Fl \cdot \frac{d}{2} = J\dot{\omega}$$

$$\frac{Tr}{r} \cdot \frac{d}{2} - \frac{Tl}{r} \cdot \frac{d}{2} = J\dot{\omega} \quad [10]$$

Donde:

\dot{u} : Aceleración lineal del móvil

Fl : Fuerza de la rueda izquierda

Fr : Fuerza de la rueda derecha

d : Distancia entre las ruedas

Tl : Torque de la rueda izquierda

Tr : Torque de la rueda derecha

$\dot{\omega}$: Aceleración angular

A las Ecuaciones 8 y 10 se las trasladaron a su forma matricial

$$\begin{pmatrix} \frac{r}{m} & \frac{r}{m} \\ \frac{d}{2rJ} & -\frac{d}{2rJ} \end{pmatrix} \cdot \begin{pmatrix} Tr \\ Tl \end{pmatrix} = \begin{pmatrix} \dot{u} \\ \dot{\omega} \end{pmatrix} \quad [11]$$

Se despeja los torques

$$\begin{pmatrix} Tr \\ Tl \end{pmatrix} = \begin{pmatrix} \dot{u} \\ \dot{w} \end{pmatrix} \begin{pmatrix} \frac{r}{m} & \frac{r}{m} \\ \frac{rd}{2J} & -\frac{rd}{2J} \end{pmatrix}^{-1} \quad [12]$$

La determinación de torques requeridos por cada rueda se realizó por medio de un programa en Simulink, de esta manera, el robot móvil se encuentra en condiciones iniciales las cuales son las componentes $x=0$, $y=0$; las posiciones deseadas fueron en $x=-4$, $y=-4$ como se puede observar en la Figura 23.

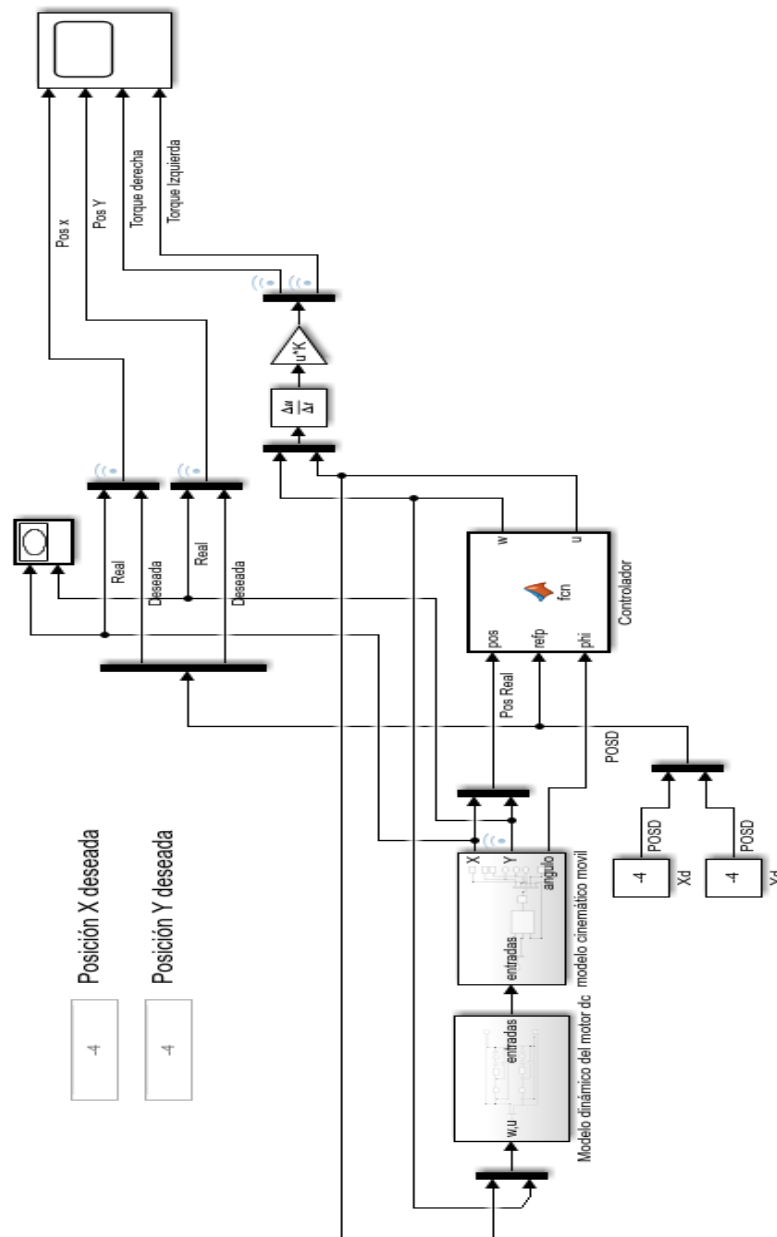


Figura 23. Integración del modelo cinemático del motor con el modelo dinámico de los motores para obtener torque requerido por el robot móvil diferencial.

En la Figura 24 se muestra la simulación de la trayectoria del robot móvil tipo diferencial realizado en *Simulink*.

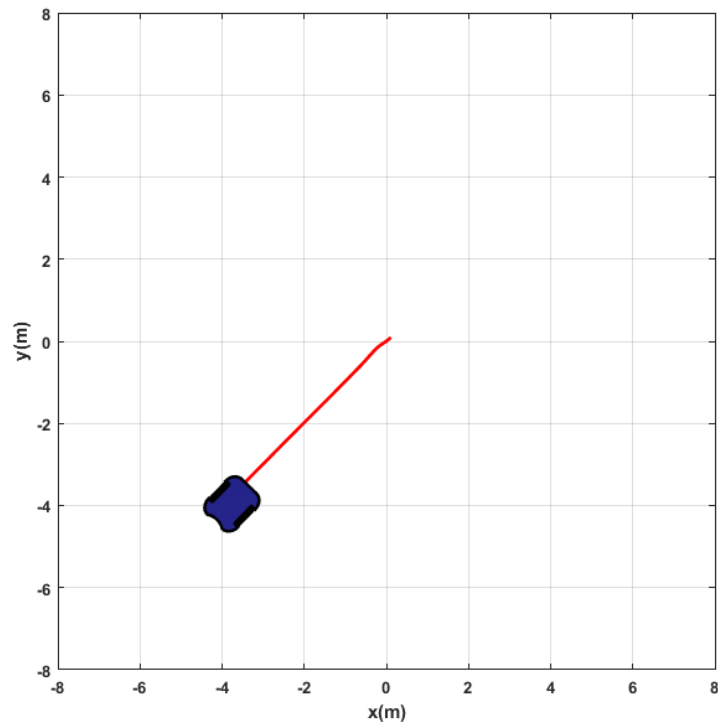


Figura 24. Simulación de trayectoria realizada por el robot móvil para llegar al punto deseado.

En la Figura 25 se muestra la gráfica de torque requerido por el motor derecho para llegar al punto deseado.

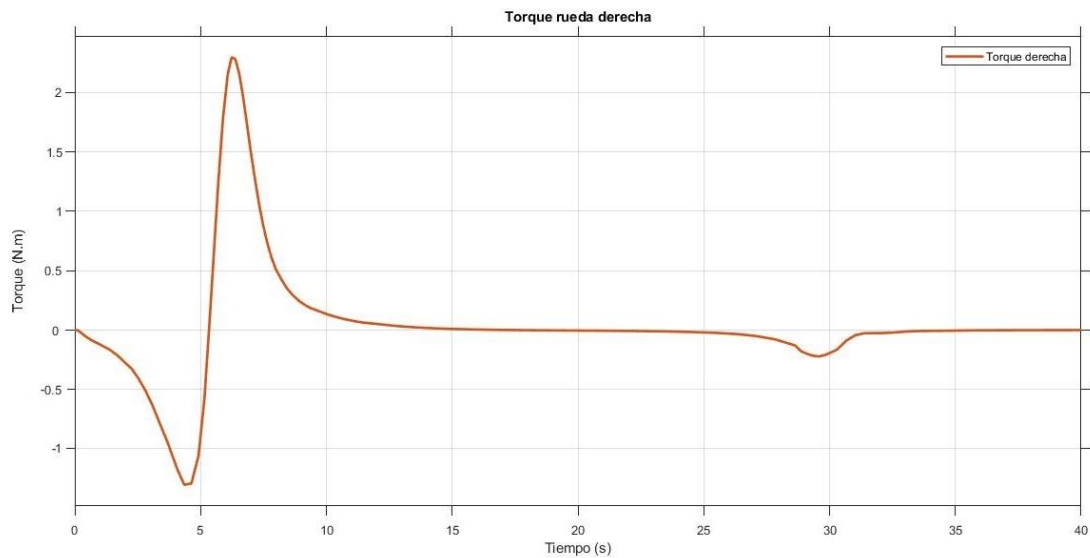


Figura 25. Torque requerido por la rueda derecha para llegar al punto deseado.

En la Figura 26 se muestra la gráfica de torque requerido por el motor izquierdo para llegar al punto deseado.

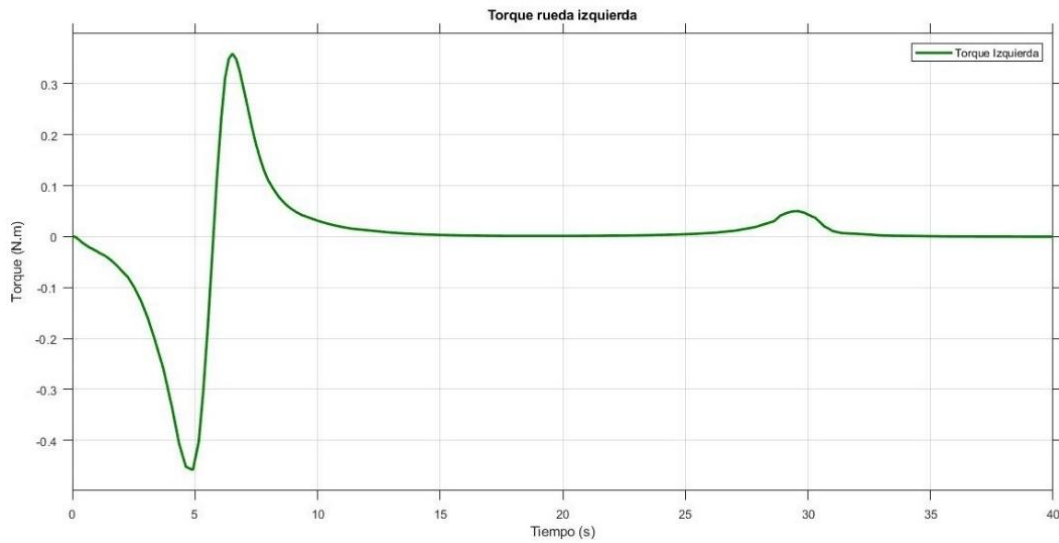


Figura 26. Torque requerido por la rueda izquierda para llegar al punto deseado.

De esta manera se demostró que para llegar a esta posición se requiere un torque máximo de la rueda derecha de 3.4 Nm ya que al móvil primero se lo hace girar en sentido antihorario hasta alcanzar la orientación deseada y luego avanzar en línea recta hasta el punto deseado $x=-4, y=-4$.

Conversión de unidades de Nm a kgf.cm

$$T = 3.4 \text{ Nm} * \left[\frac{1 \text{ Kgf}}{9.81 \text{ N}} \right] * \left[\frac{100 \text{ cm}}{1 \text{ m}} \right]$$

$$T = 34.65 \text{ [kgf.cm]}$$

El torque que se calculó fue de $T=34.65 \text{ [kgf.cm]}$, este valor contribuyó en la selección del motor DC el cual debe tener un mayor torque que el calculado por ello como se muestra en la Figura 27, se eligió aquel que entrega un torque de $T=42 \text{ [kgf.cm]}$.

DC 12V, no load current-0.5A, rated current-1.2A							
Reduction ratio	17	31	50	100	200	290	500
No load speed (rpm)	470	260	160	80	40	27	12
Rated torque (Kg.cm)	2.5	4.5	7.3	14.5	29	42	60
Rated speed (rpm)	380	200	130	60	30	20	12

Figura 27. Características de los motores comerciales. (Bringsmart, 2022)

El motor DC seleccionado como se muestra en la Figura 28, funciona con un voltaje de 12V y una corriente de 1.2A, entrega una velocidad de 27 RPM, posee una caja reductora de radio reducción de 1:290 y también posee un encoder de cuadratura de efecto Hall (Bringsmart, 2022).



Figura 28. Motor DC 12V con encoder de efecto Hall (Bringsmart, 2022)

2.3.2.2 Selección de driver para los motores

Con la finalidad de controlar la velocidad y la dirección de giro de los motores, se seleccionó el controlador L298N ya que puede entregar una capacidad de corriente de hasta 2A. El control de la dirección de giro y velocidad lo hace mediante señales de tipo TTL obtenidas de un microcontrolador como puede ser Raspberry, Arduino o PIC por medio de la utilización de ancho de pulso (PWM) (Quintero , et al. 2018). Posee una tarjeta con conexiones hacia el motor por medio de bloques terminales y *headers* macho estándar a manera de entradas de control como se puede observar en la Figura 29.

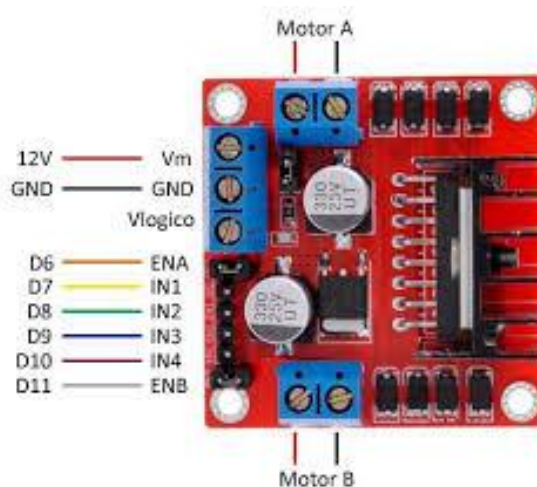


Figura 29. Driver L298N puente H. (Del Pozzo , 2021)

2.3.2.3 Selección del microcontrolador

El microcontrolador escogido fue el módulo ESP32 como se muestra en la Figura 30, ya que contiene dos elementos integrados: una antena Bluetooth y otra Wifi las cuales permiten una comunicación inalámbrica. Además, cuenta con un microprocesador de 32 bits de núcleo simple o doble con una operación de 160 a 240 MHz cuya función principal es la comunicación con el sensor de infrarrojos. Adicionalmente incluye un detector de radiofrecuencia, amplificador de potencia, amplificador receptor de bajo ruido, filtros y módulos de administración de energía acoplados y la mayoría de sus pines posee salida PWM (Peris, 2020).

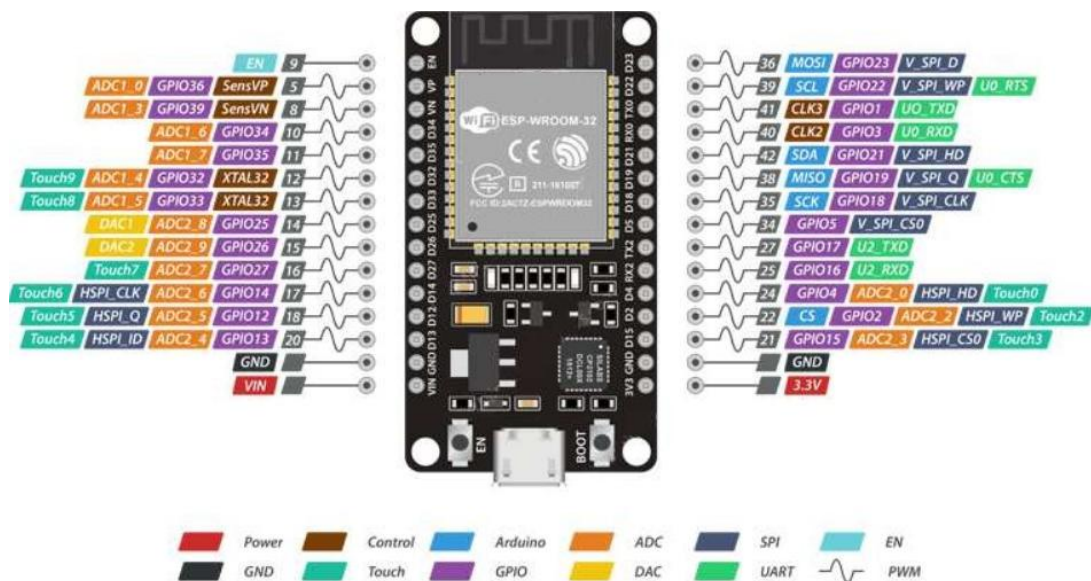


Figura 30. Microcontrolador ESP32.
(Guamá, 2020)

2.3.2.4 Fuente de alimentación

Como fuente de energía se seleccionó una batería tipo seca recargable como se observa en la Figura 31, la cual está conformada por placas que contienen plomo ácido y están selladas con un electrolito con separadores de fibra de vidrio lo que la convierte en una batería de libre mantenimiento (Sánchez & Figueroa, 2022). Entre sus características principales se encuentra el voltaje de 12 voltios, 12 amperios, sus dimensiones son: largo 151mm, ancho 99 mm y alto 100mm; su principal ventaja es la capacidad de desempeño de 12V 12A/20 horas de trabajo continuo (Cepeda-Camacho, Garzón-Muñoz, Guasumba-Maila, & Oramas-Proaño, 2022)



Figura 31. Batería seca recargable
(Sánchez & Figueroa, 2022)

2.3.2.5 Convertidor de voltaje *step down*

Se eligió el convertidor de voltaje LM2596 que se puede observar en la Figura 32, debido a que la batería entrega un voltaje de 12V y el microcontrolador necesita 5V estables para su funcionamiento, además que soporta una salida de corriente de hasta 3A.

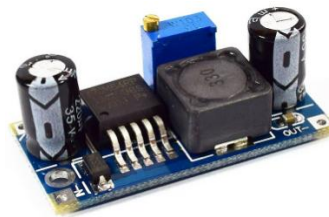


Figura 32. Convertidor de voltaje LM2596 Step Down
(Guamá, 2020)

2.3.2.6 Pantalla oled

Se seleccionó la pantalla Oled SSD1306 de 128x64 pixeles que se puede observar en la Figura 33 por su eficiencia energética ya que no requiere una luz de fondo para poder visualizar el contenido, también porque se usan cuatro pines para su funcionamiento los de energía Vcc y GND; los de envío y recepción de datos por medio del protocolo de comunicación serial I2C que tiene los cables de Serial Data SDA y Serial Clock SCL. Se usó para la interfaz gráfica para que el usuario pueda monitorear el estado de funcionamiento del robot, datos de velocidad lineal, angular y también el voltaje de la batería.



Figura 33. Pantalla Oled
(Ravi & Kopulwar, 2016)

2.3.2.7 Integración de componentes electrónicos

En la Figura 34, se muestra un diagrama de la conexión de los componentes electrónicos mencionados anteriormente realizado en el programa Proteus.

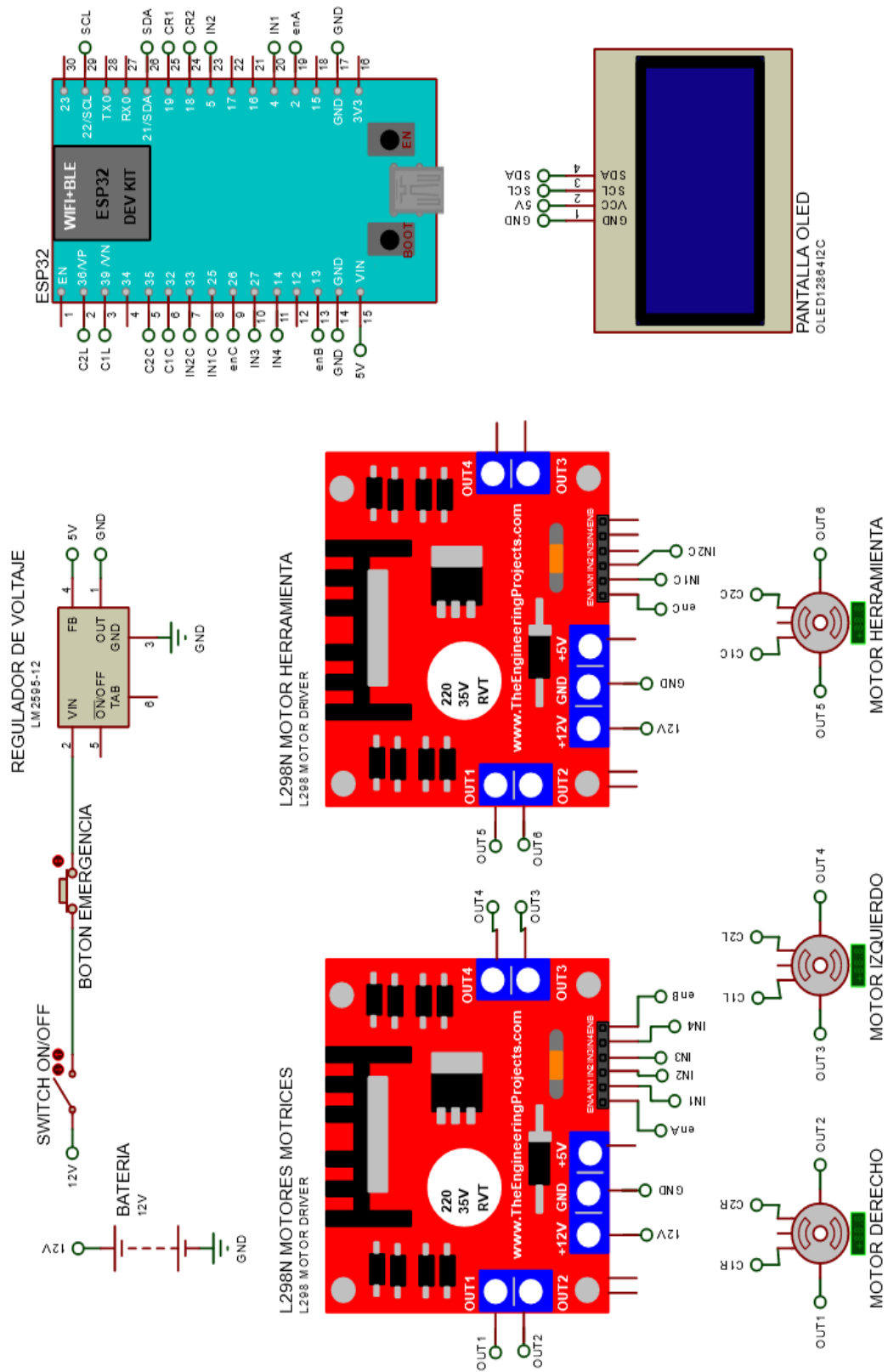


Figura 34. Diagrama esquemático de los componentes electrónicos del robot.

2.3.3 SISTEMAS DE CONTROL

2.3.3.1 Diseño del controlador de los motores

El motor que se seleccionó posee un encoder rotativo incremental de cuadratura de efecto Hall el cual cuenta con un disco acoplado a un eje de rotación con la presencia de polos a lo largo de su circunferencia como se muestra en la Figura 35. Su funcionamiento consiste en que mientras el disco gira se produce una tensión en estos sensores por el cambio de un polo a otro y de esta manera el circuito de acondicionamiento la transforma en un impulso digital (Cantero, 2019).

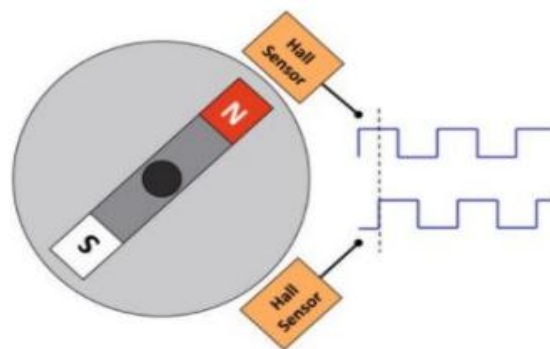


Figura 35. Representación de un encoder magnético.
(IEEE Student Branch, 2021)

El encoder magnético de efecto Hall, indica con gran exactitud la posición, la velocidad y la dirección horaria o antihoraria del eje del motor, todos estos datos se los pudo adquirir por medio de dos sensores los cuales generan dos señales de pulsos como se pueden observar en la Figura 36 que a su vez son desfazadas en π rad o 90° .

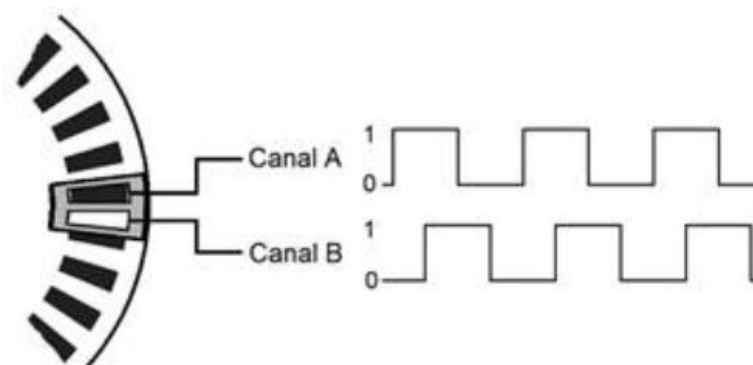


Figura 36. Señales de los canales A y B de un encoder de cuadratura.
(Suárez & Pérez, 2020)

El desfase entre las señales A y B del encoder se tomaron como referencia para la determinación del sentido de giro. Al trabajar con dos canales, se llegan a tener cuatro estados que son la combinación de A y B. Asumiendo la salida A como el bit más significativo y la salida B como el menos significativo en la tabla de verdad representada en la Tabla 2 se obtuvo la dirección de giro comparando el estado actual con el anterior. De esta manera si gira en sentido antihorario tendría un signo positivo (+), de lo contrario en sentido horario tendría signo negativo (-). Se debe tomar en cuenta que existen cuatro combinaciones que no pueden aparecer en la señal y están marcados con una E.

Tabla 2. Tabla de verdad de las señales A y B del encoder de cuadratura.

Estado	Anterior	0	1	2	3
Actual	AB	00	01	10	11
0	00	0	-1	+1	E
1	01	+1	0	E	-1
2	10	-1	E	0	+1
3	11	E	+1	-1	0

En lo correspondiente a su resolución, este emite 16 cuentas por revolución del eje del motor y se utilizó una precisión de tipo cuádruple que permitió detectar el sentido de giro del eje del motor. De esta manera, la resolución se calculó con la ecuación de pulsos por vuelta (Mamani, 2020) de la siguiente manera:

$$R = mH * s * r \quad [13]$$

Donde:

R: Es la resolución del encoder para una precisión cuádruple.

mH: Es el número de cuentas por revolución del eje del motor.

s: Es el número de estados generados por los canales AB.

r: Es la relación de reducción de la caja reductora (ratio).

Por lo tanto,

$$R = 16 * 4 * 290 = 18560 \quad [14]$$

Para medir la posición relativa del eje del motor se utilizó la fórmula descrita por Hernández, P; Navarrete, J (2022):

$$P = \frac{n * 360}{R} \quad [15]$$

Donde:

P: Es la posición relativa en grados.

n: Es el número de cuentas generadas.

R: Es la resolución del encoder para una precisión cuádruple

Para medir las revoluciones por minuto se utilizó la fórmula descrita a continuación por Hernández, P; Navarrete, J. (2022):

$$N = \frac{n * 60}{t * R} \quad [16]$$

Donde:

N: Es la velocidad de rotación en Revoluciones Por Minuto (RPM).

n: Es el número de conteo generado en determinado tiempo (t).

t: Es el tiempo de generación de los pulsos en segundos (s).

R: Es la resolución del encoder para una precisión cuádruple.

Hay que tomar en cuenta que el tiempo t es en segundos, sin embargo, en microcontroladores se trabaja en el orden de microsegundos o milisegundos para lo cual se debe tener en presente la conversión 1s= 1000ms (Hernández, P; Navarrete, J., 2022) y aplicarlo a la fórmula:

$$N = \frac{n * 60 * 1000}{t * R} \quad [17]$$

Para el cálculo de la velocidad angular se utilizaron las revoluciones por minuto (RPM) tomando en cuenta que una revolución equivale a 2π radianes y 1 minuto equivale a 60 segundos (Hernández, P; Navarrete, J., 2022). Por lo tanto, se la puede calcular de la siguiente manera:

$$\omega = \frac{2\pi * n}{t * R} \quad [18]$$

Donde:

ω : Es la velocidad de rotación en radianes por segundo (rad/s).

n: Es el número de conteo generado en determinado tiempo t.

- t : Es el tiempo de generación de los pulsos en segundos (s).
 R : Es la resolución del encoder para una precisión cuádruple.

Para calcular la velocidad lineal en función del número de cuentas y del diámetro de la llanta (Hernández, P; Navarrete, J., 2022) se utilizó la siguiente fórmula:

$$u = \frac{d_{wheel} * \pi * n}{t * R} \quad [19]$$

Donde:

- u : Es la velocidad lineal en metros por segundo m/s.
 d_{wheel} : Es el diámetro de la rueda en metros (m).
 n : Es el número de conteo generado en determinado tiempo t .
 t : Es el tiempo de generación de los pulsos en segundos (s).
 R : Es la resolución del encoder para una precisión cuádruple.

2.3.3.2 Identificación del modelo del motor

Para la sintonización del controlador del motor en lazo cerrado, se necesitó tener la información del comportamiento dinámico del proceso que usualmente se representa en forma de un modelo de orden reducido (primero o segundo orden) en este caso se utilizó un modelo de primer orden más un tiempo muerto. La ganancia, tiempo muerto y constantes de tiempo son parámetros que se pueden identificar mediante la respuesta del proceso a una entrada tipo escalón conocida como la curva de reacción del proceso. (Alfaro, 2006).

Sistema de primer orden más un tiempo muerto:

Representación en el dominio del tiempo:

$$p_v(t) = \begin{cases} 0 & 0 \leq t < L \\ k \left(1 - e^{-\frac{t-L}{\tau}} \right) \Delta c_v & t \geq L \end{cases} \quad [20]$$

Representación en la transformada de Laplace:

$$G(s) = \frac{k \cdot e^{-Ls}}{\tau s + 1} \quad [21]$$

Donde:

k : Es la ganancia proporcional

L : Es el tiempo de retardo

c_v : Es la variable de control

τ : Es el tiempo que el proceso tarda en alcanzar el 63,2% de su valor final

Como se muestra en la Figura 37, para la estimación de los valores del comportamiento de los motores en orden reducido se realizó una escala de 0 a 100 de los valores de PWM de entrada que son de 0 a 255, en cuanto a la velocidad de salida cuyos valores se encuentran entre 0 a 2 [rad/s] de igual manera se utilizó una escala de 0 a 100.

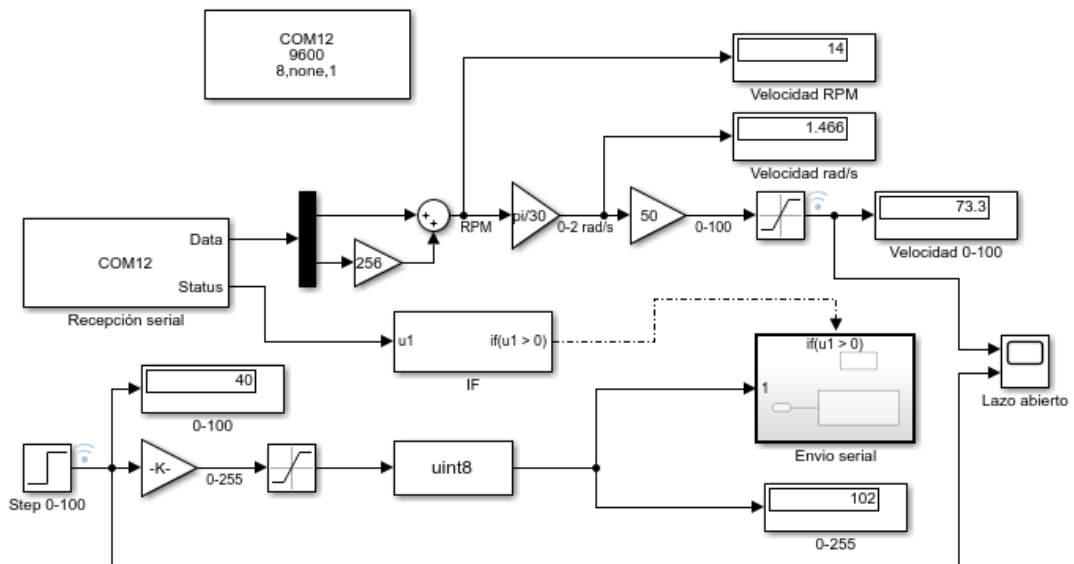


Figura 37. Lectura de velocidad del motor por comunicación serial en *Simulink*.

Motor derecho

La respuesta al escalón en control de lazo abierto se muestra en la Figura 38. Los valores obtenidos tanto de velocidad (variable de proceso) y la señal PWM (variable de control) fueron almacenados en una lista generada en Matlab para la estimación del proceso.

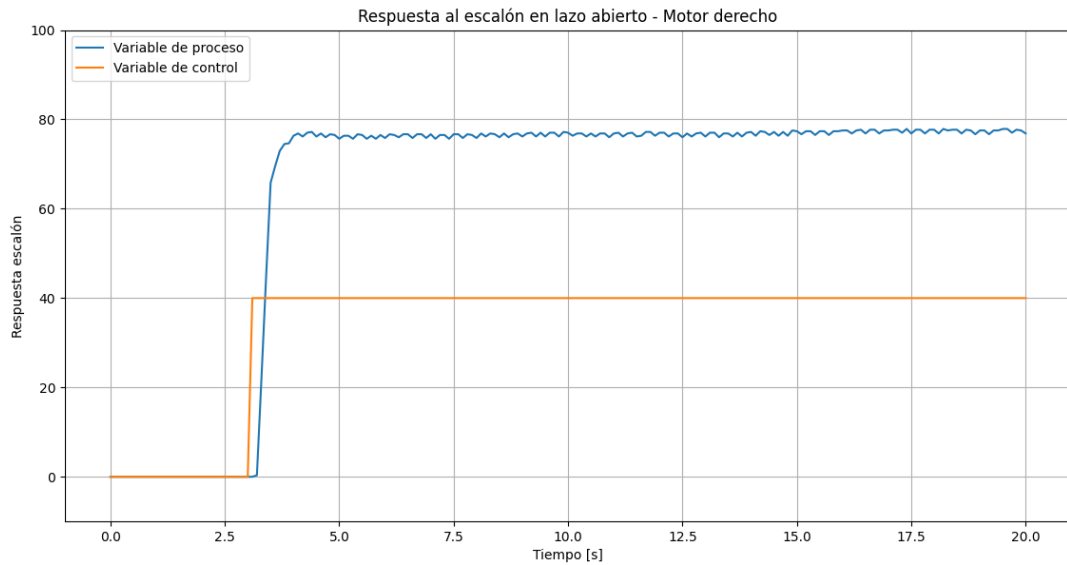


Figura 38. Respuesta al escalón en lazo abierto - Motor derecho.

Los datos obtenidos anteriormente fueron exportados en vectores de los valores de entrada que corresponden a la señal escalón y los valores de salida que corresponden la velocidad real medida fueron de utilidad para realizar la identificación de las funciones de transferencia de los motores. Para esto se utilizó la herramienta *System Identification* de Matlab como se muestra en la Figura 39 donde se importaron los datos obtenidos en el dominio del tiempo, de esta manera como entrada (*Input*) se colocó el vector del escalón y como salida (*Output*) el vector de velocidad con un tiempo de muestreo (*Sample time*) de 0.1s.

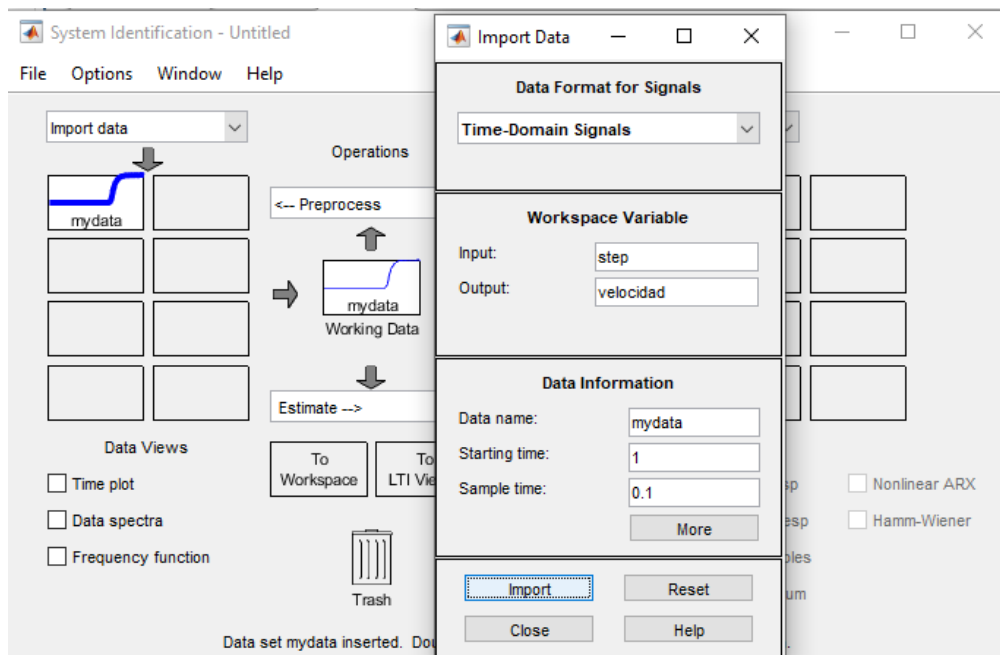


Figura 39. Importación de datos a la herramienta *System Identification* Matlab.

Posteriormente se comprobaron que los datos que se cargaron al *Import data* dieron como resultado la misma grafica que se indicó en la Figura 39. Para poder visualizar esta grafica se seleccionó a *Time plot* como se muestra en la Figura 40.

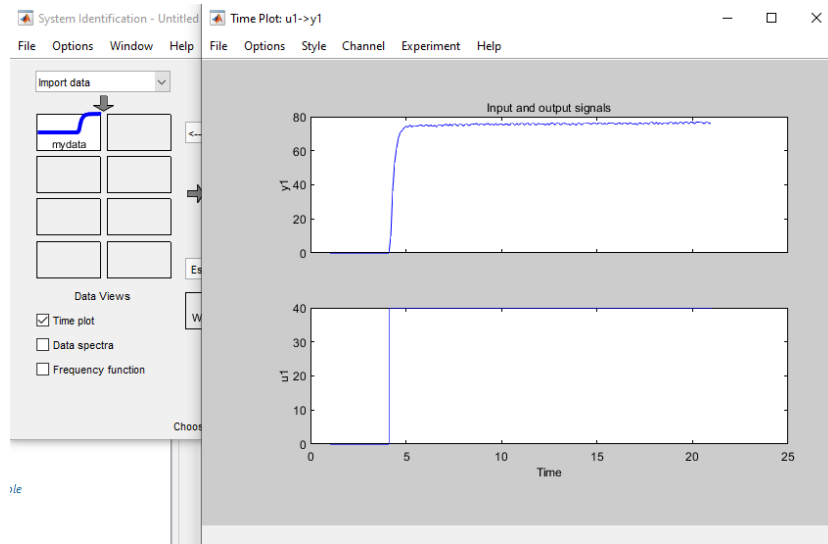


Figura 40. Entrada y salida de la función de transferencia.

Para la estimación en *System Identification* se seleccionó la opción de *Process Models* para poder obtener la función de transferencia de orden reducido con retardo y el programa se encargó de generar los valores como se muestra en la Figura 41.

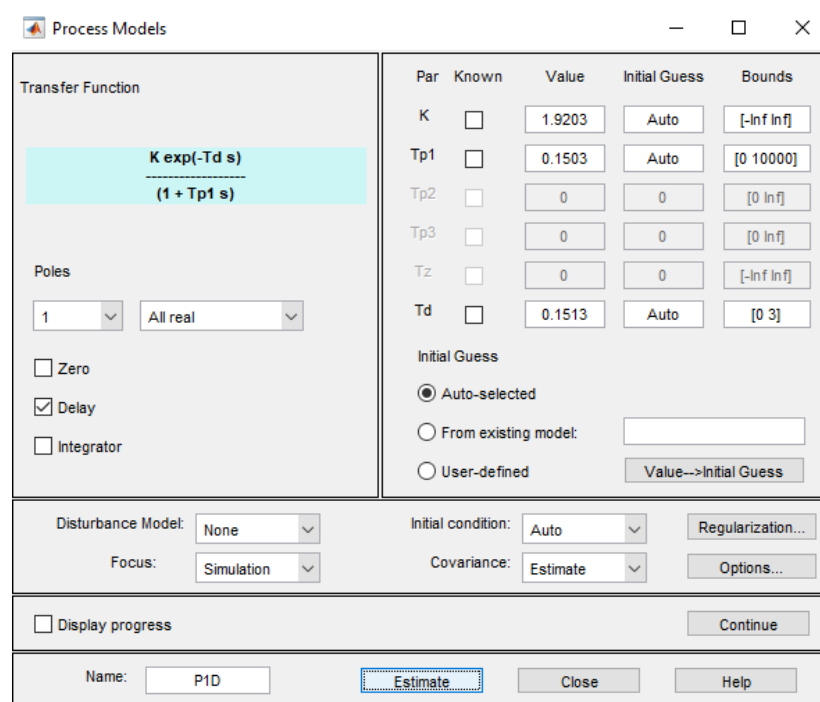


Figura 41. Estimación de los valores de la función de transferencia mediante *Process Models*.

Para poder visualizar la gráfica de la función de transferencia estimada llamada P1D se seleccionó la opción *Model output* y como se puede observar en la Figura 42, se presentó una coincidencia del 97.76% con el comportamiento del motor.

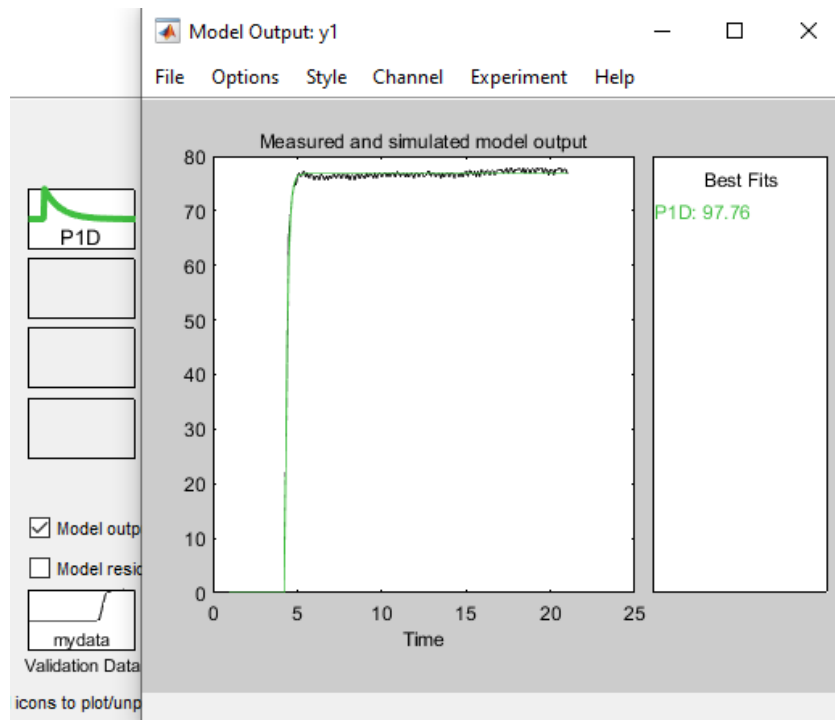


Figura 42. Gráfica de la función de transferencia estimada.

Posteriormente, se exportó la función de transferencia encontrada a *workspace* de Matlab, dando así la expresión de la Figura 43.

```

Command Window
>> G=tf(P1D)

G =

From input "u1" to output "y1":
          1.92
exp(-0.151*s) * -----
                0.1503 s + 1

Name: P1D
Continuous-time transfer function.
    
```

Figura 43. Función de transferencia con retardo del motor derecho.

Se realizó la comparación de la señal emitida por el motor con la función de transferencia obtenida mediante el programa que se visualiza en la Figura 44, desarrollado en *Simulink*.

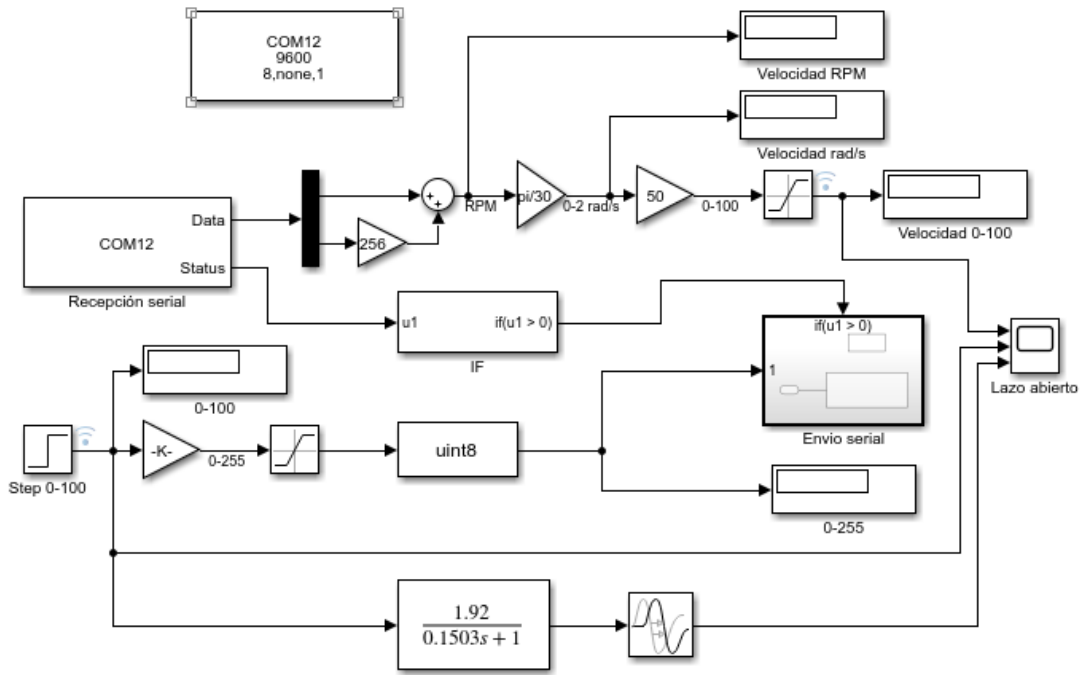


Figura 44. Comparación de la respuesta del motor con su función de transferencia.

Finalmente, las gráficas obtenidas de las señales de respuesta del motor y de la función de transferencia estimada se las puede visualizar en la Figura 45.

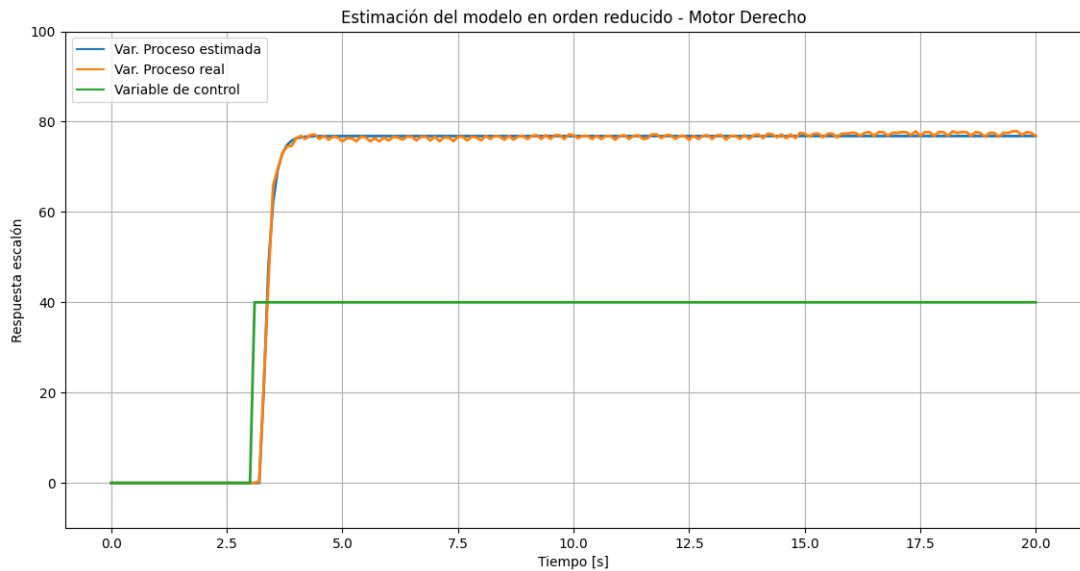


Figura 45. Gráficas obtenidas de la comparación de la respuesta del motor con su función de transferencia.

Motor izquierdo

La respuesta al escalón en control de lazo abierto se muestra en la Figura 46. Los valores obtenidos tanto de velocidad de la variable de proceso y la

variable de control fueron almacenados en una lista generada en Matlab para la estimación del proceso.

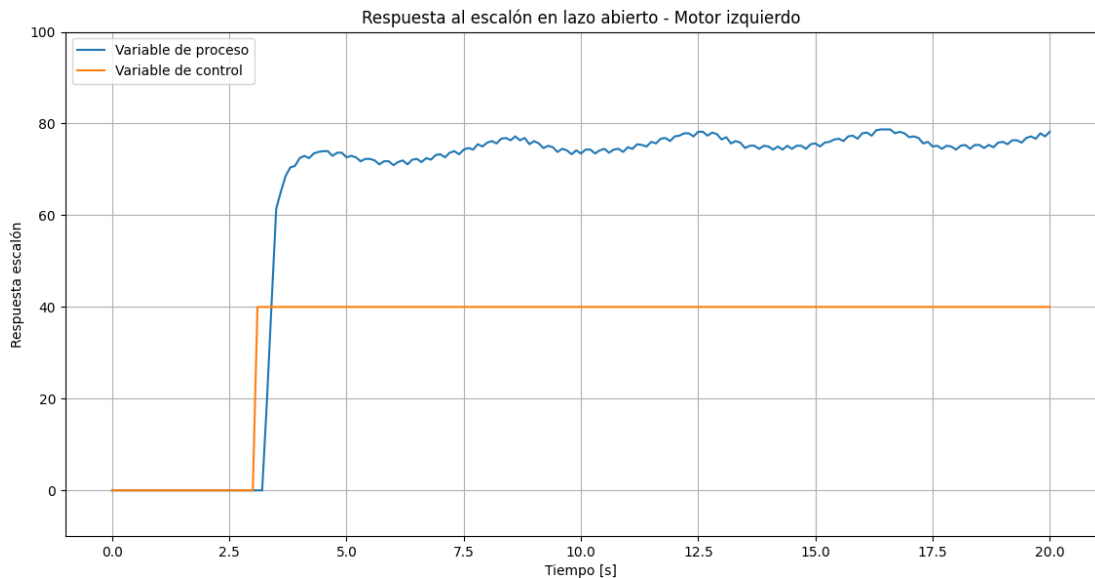


Figura 46. Estimación de los valores de la función de transferencia en motor izquierdo mediante *Process Models*.

Para la estimación en *System Identification* al igual que en el motor derecho, se seleccionó la opción de *Process Models* para obtener la función de transferencia de orden reducido con retardo obteniéndose los valores mostrados en la Figura 47 mediante el programa.

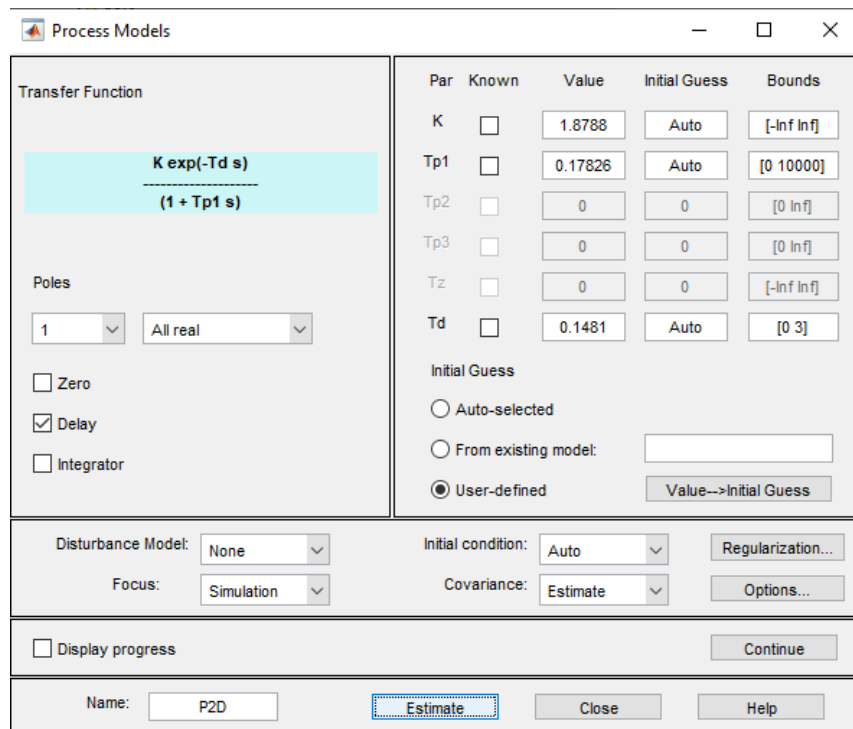


Figura 47. Respuesta al escalón en lazo abierto - Motor izquierdo.

Para lograr la visualización de la gráfica de la función de transferencia estimada P2D al igual que en el motor derecho, se seleccionó Model output presentándose una coincidencia del 94% con el comportamiento del motor, resultado que se puede visualizar en la Figura 48.

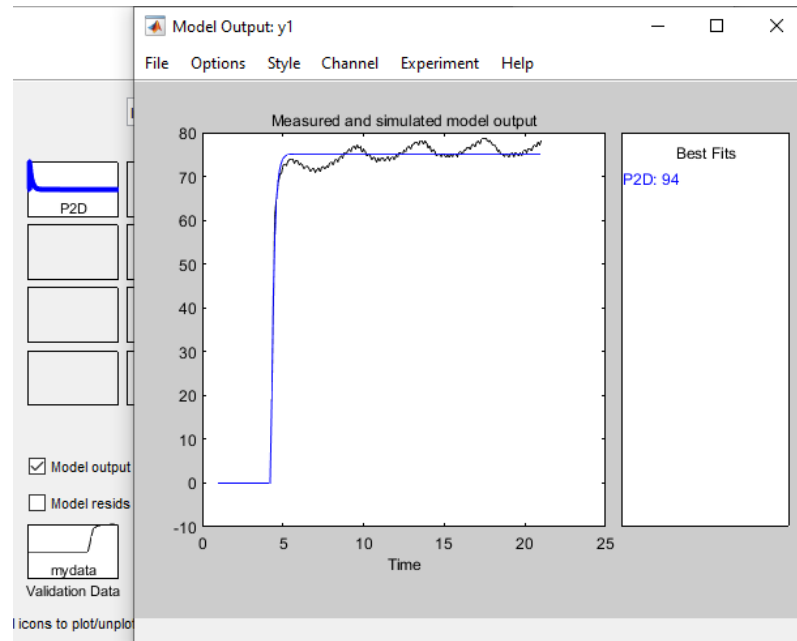


Figura 48. Gráfica de la función de transferencia estimada.

A continuación, se exportó la función de transferencia encontrada a *workspace* de Matlab, presentándose así la expresión de la Figura 49.

```

Command Window
>> GI=tf(P2D)

GI =

    From input "u1" to output "y1":
           1.879
    exp(-0.148*s) * -----
                   0.1783 s + 1

    Name: P2D
    Continuous-time transfer function.
  
```

Figura 49. Función de transferencia con retardo del motor izquierdo.

Posteriormente se realizó la comparación de la señal emitida por el motor con la función de transferencia obtenida mediante el programa desarrollado en *Simulink* como se aprecia en la Figura 50.

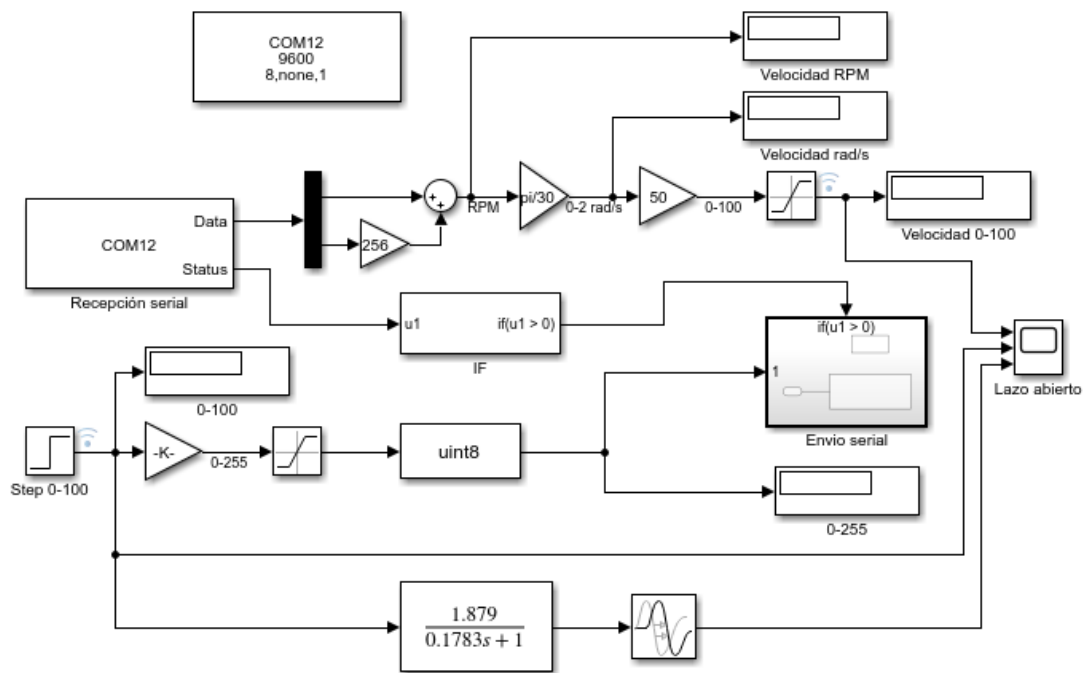


Figura 50. Comparación de la respuesta del motor izquierdo con su función de transferencia.

Finalmente, las gráficas obtenidas como resultado de las señales de respuesta del motor y de la función de transferencia estimada se las puede visualizar en la Figura 51.

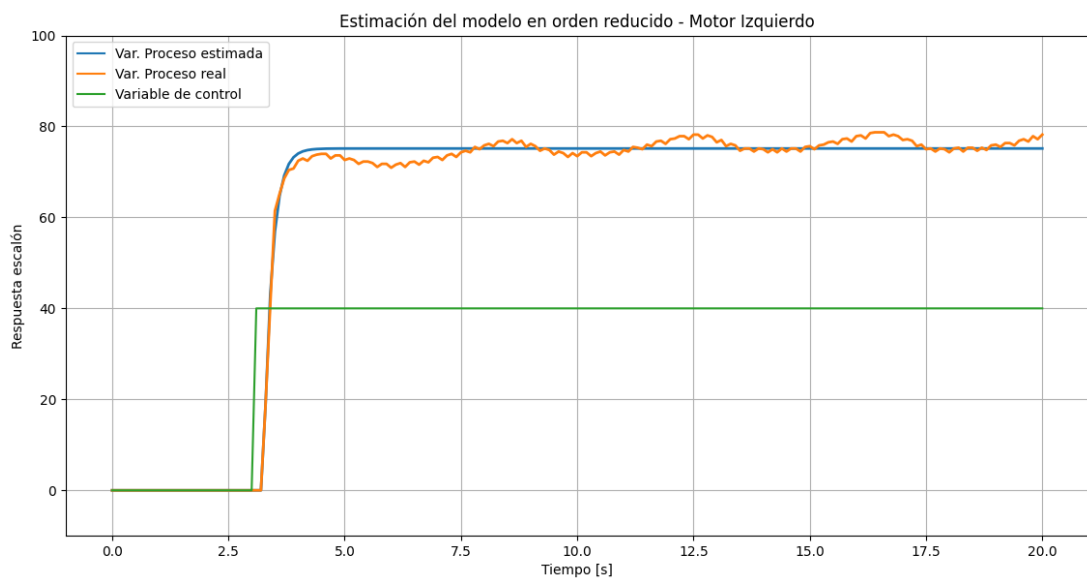


Figura 51. Gráficas obtenidas de la comparación de la respuesta del motor izquierdo con su función de transferencia.

2.3.3.3 Control PID de los motores

Para poder controlar los motores de una manera adecuada se implementaron dos métodos de sintonía estos fueron los métodos propuestos por Ziegler Nichols que es el más utilizado y el método Lambda el cual ha sido ampliamente usado y este tiene la ventaja de ser un controlador robusto y que no presenta sobre impulso ni oscilaciones.

Para obtener los valores de los controladores PI y PID se realizaron cálculos aplicando las fórmulas tanto para del método Ziegler Nichols como para el método Lambda. En la Figura 52 se muestra el control en lazo cerrado de las funciones de transferencia de los motores.

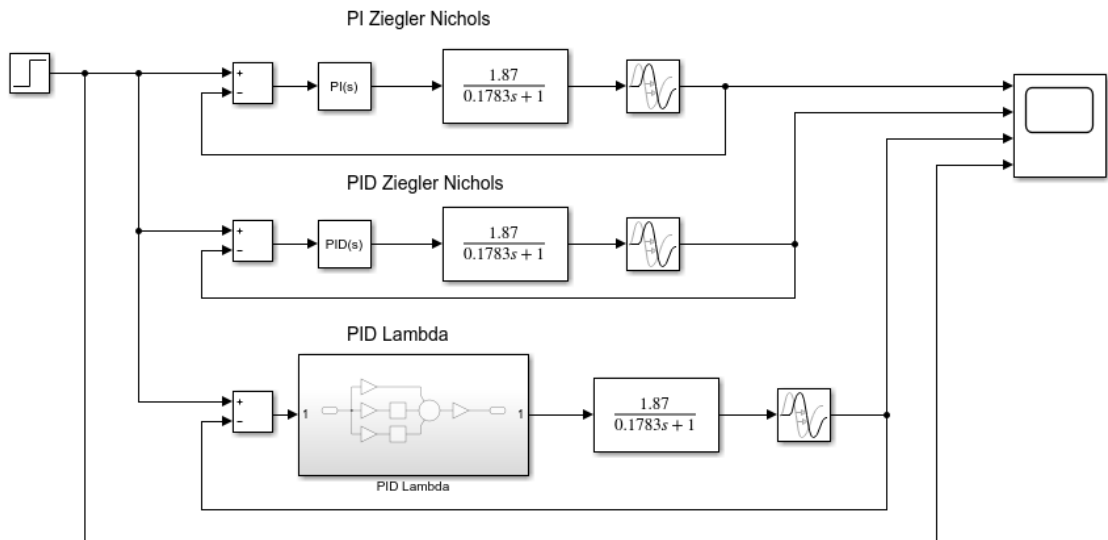


Figura 52. Control en lazo cerrado.

Fórmulas para el cálculo de los parámetros PI y PID Ziegler Nichols

Controlador PI

$$k_p = \frac{0.9 \cdot \tau}{k \cdot L} \quad [22]$$

$$T_i = 3L \quad [23]$$

$$k_i = \frac{k_p}{T_i} \quad [24]$$

Controlador PID

$$k_p = \frac{1.2 \cdot \tau}{k \cdot L} \quad [25]$$

$$T_i = 2L \quad [26]$$

$$Td = 0.5L \quad [27]$$

$$ki = \frac{kp}{Ti} \quad [28]$$

$$kd = kp \cdot Td \quad [29]$$

Controlador PI y PID Ziegler Nichols – Motor derecho

La función de transferencia que se obtuvo del motor derecho es la siguiente:

$$G(s) = \frac{1.92 \cdot e^{-0.1513s}}{0.1503s + 1} \quad [30]$$

Los cálculos de los parámetros del controlador PI Ziegler – Nichols del motor derecho se encuentran en la Tabla 3.

Tabla 3. Parámetros del controlador PI Ziegler Nichols - Motor Derecho

PI	Kp	$\frac{0.9 \cdot \tau}{k \cdot L} = \frac{0.9 \cdot 0.1783}{1.92 \cdot 0.1513} = 0.466$
	Ti	$3L = 0.454$
	Ki	$\frac{kp}{Ti} = \frac{0.466}{0.454} = 1.026$

Los cálculos de los parámetros del controlador PID Ziegler – Nichols del motor derecho se encuentran en la Tabla 4.

Tabla 4. Parámetros del controlador PID Ziegler Nichols - Motor Derecho

PID	Kp	$\frac{1.2 \cdot \tau}{k \cdot L} = \frac{1.2 \cdot 0.1503}{1.92 \cdot 0.1513} = 0.621$
	Ti	$2L = 0.3026$
	Td	$0.5L = 0.07$
	Ki	$\frac{kp}{Ti} = \frac{0.621}{0.3026} = 2.0522$
	Kd	$kp \cdot Td = 0.621 \cdot 0.07 = 0.043$

En la Figura 53 se muestra la respuesta de la función del motor derecho ante una entrada escalón en lazo cerrado aplicando los controladores PI y PID por el método Ziegler Nichols.

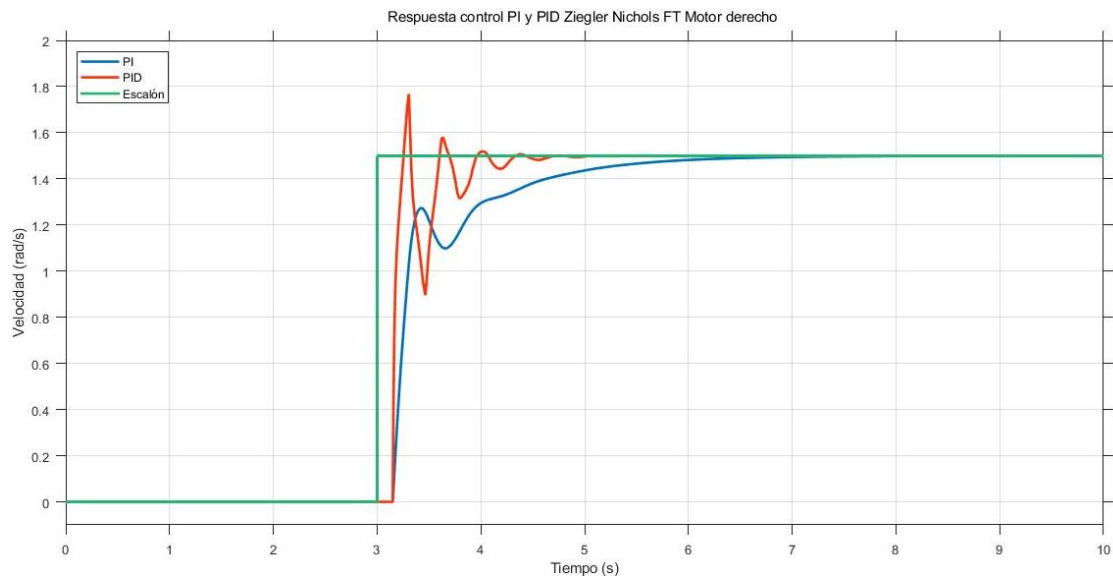


Figura 53. Respuesta control PI y PID Ziegler Nichols FT Motor derecho

Controlador PI y PID Ziegler Nichols – Motor izquierdo

La función de transferencia que se obtuvo del motor izquierdo es la siguiente:

$$G(s) = \frac{1.879 \cdot e^{-0.1481s}}{0.1783s + 1} \quad [31]$$

Los cálculos de los parámetros del controlador PI Ziegler – Nichols del motor izquierdo se encuentran en la Tabla 5.

Tabla 5. Parámetros del controlador PI Ziegler Nichols - Motor Izquierdo

PI	K_p	$\frac{0.9 \cdot \tau}{k_p \cdot L} = \frac{0.9 \cdot 0.1783}{1.879 \cdot 0.1481} = 0.5766$
	T_i	$3L = 0.443$
	K_i	$\frac{k_p}{T_i} = \frac{0.5766}{0.443} = 1.3015$

Los cálculos de los parámetros del controlador PID Ziegler – Nichols del motor izquierdo se encuentran en la Tabla 6.

Tabla 6. Parámetros del controlador PID Ziegler Nichols - Motor Izquierdo

PID	Kp	$\frac{1.2 \cdot \tau}{k_p \cdot L} = \frac{1.2 \cdot 0.1783}{1.879 \cdot 0.1481} = 0.7688$
	Ti	$2L = 0.2962$
	Td	$0.5L = 0.074$
	Ki	$\frac{k_p}{T_i} = \frac{0.7688}{0.2962} = 2.5955$
	Kd	$k_p \cdot T_d = 0.7688 \cdot 0.074 = 0.033$

En la Figura 54 se muestra la respuesta de la función del motor izquierdo ante una entrada escalón en lazo cerrado aplicando los controladores PI y PID por el método Ziegler Nichols.

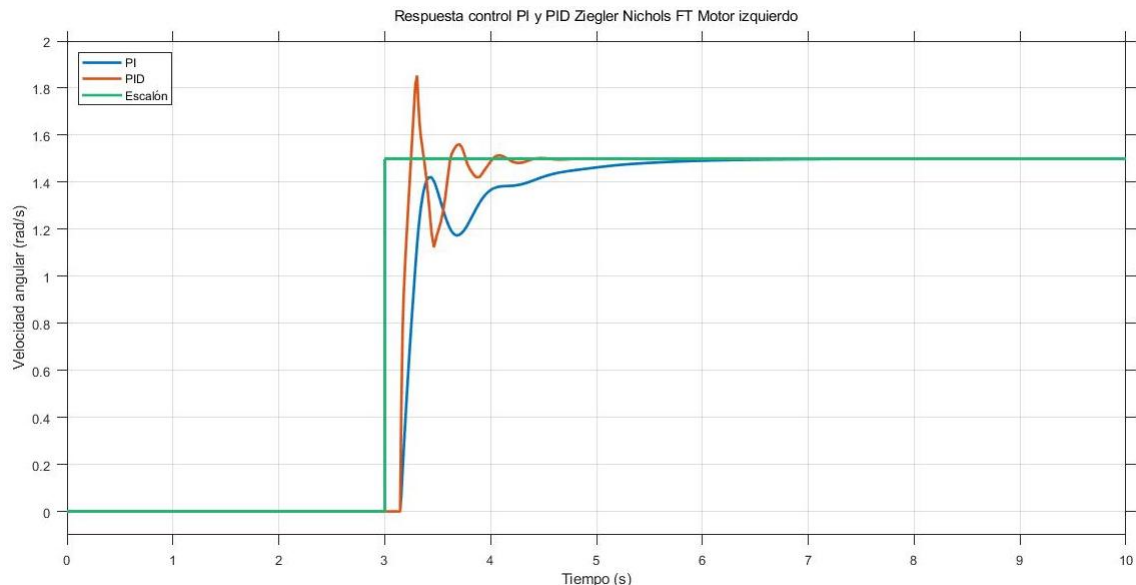


Figura 54. Respuesta control PI y PID Ziegler Nichols FT Motor izquierdo

Controlador PID Lambda

Fórmulas para el cálculo de los parámetros PID Lambda (Pruna, Sasig, & Mullo, 2017)

$$k_p' = \frac{\tau}{k \cdot \left(\frac{L}{2} + 3 \cdot \lambda\right)} \quad [32]$$

$$T'_i = \tau \quad [33]$$

$$T'_d = \frac{L}{2} \quad [34]$$

Donde $\lambda = 3\tau$ para un controlador robusto y $\lambda = \tau$ para una sintonía inmediata.

$$kp = kp' \cdot \frac{T'_i + T'_d}{T'_i} \quad [35]$$

$$ki = T'_i + T'_d \quad [36]$$

$$kd = \frac{T'_i \cdot T'_d}{T'_i + T'_d} \quad [37]$$

Controlador PID Lambda – Motor derecho

Los cálculos de los parámetros del controlador PID Lambda del motor derecho se encuentran en la Tabla 7.

Tabla 7. Parámetros del controlador PID Lambda Motor derecho

PID	Kp	$kp' = \frac{\tau}{k \cdot \left(\frac{L}{2} + 3 \cdot \lambda\right)} = \frac{0.1503}{1.92 \cdot \left(\frac{0.1513}{2} + 3 \cdot 0.1503\right)} = 0.1486$
	Ti	$T'_i = \tau = 0.1503$
	Td	$T'_d = \frac{L}{2} = \frac{0.1513}{2} = 0.075$
	Kp	$kp = kp' \cdot \frac{T'_i + T'_d}{T'_i} = 0.1486 \cdot \frac{0.1503 + 0.075}{0.1503} = 0.2228$
	Ki	$T'_i + T'_d = 0.1503 + 0.075 = 0.2253$
	Kd	$\frac{T'_i \cdot T'_d}{T'_i + T'_d} = \frac{0.1503 \cdot 0.075}{0.1503 + 0.075} = 0.05$

En la Figura 55 se muestra la respuesta del motor derecho ante una entrada escalón en lazo cerrado aplicando el controlador PID por el método Lambda.

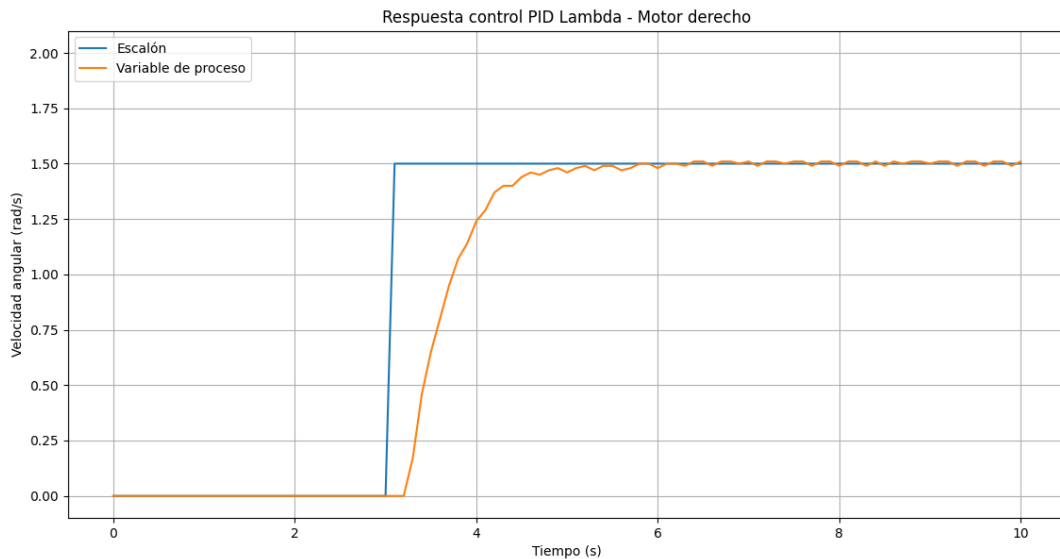


Figura 55. Respuesta al escalón PID Lambda Motor derecho

En la Figura 56 se muestra la respuesta de la función del motor derecho ante una trayectoria en lazo cerrado aplicando el controlador PID por el método Lambda.

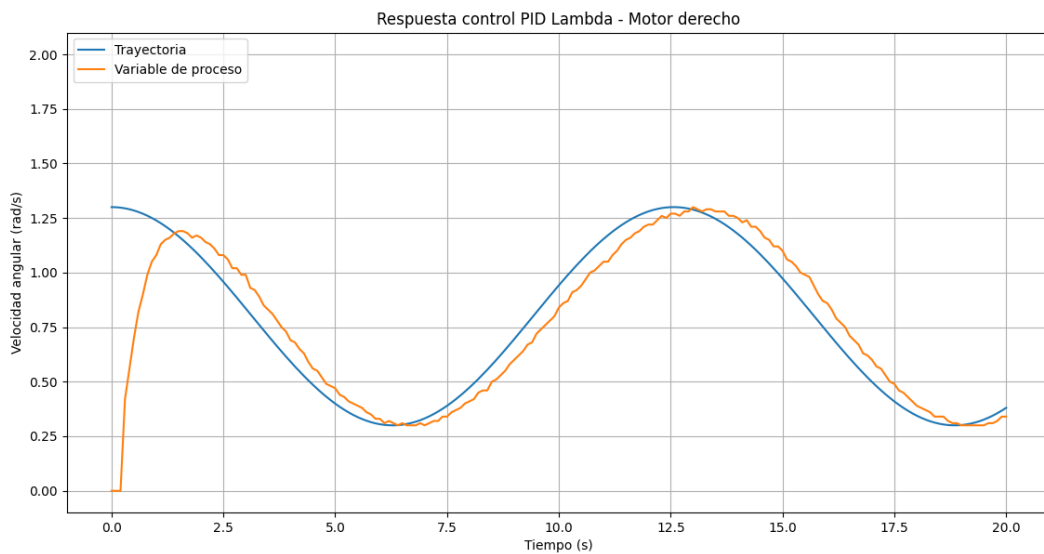


Figura 56. Respuesta a la trayectoria PID Lambda Motor derecho

En referencia a la figura se evidenció que la respuesta a la trayectoria tiene un retardo en su respuesta por ello fue necesario realizar una sintonía fina modificando los valores de k_p , k_i y k_d , tomando en cuenta que si se aumenta la ganancia proporcional disminuye la estabilidad, si se disminuye el tiempo de integración el error decae rápidamente a su vez que disminuye la estabilidad y si se aumenta el tiempo derivativo incrementa la estabilidad. De esta manera se obtuvieron los nuevos valores del controlador PID Lambda

($k_p = 0.27$; $k_i = 0.12$; $k_d = 0.03$) y como se puede evidenciar en la Figura 57 mejora su respuesta a la trayectoria.

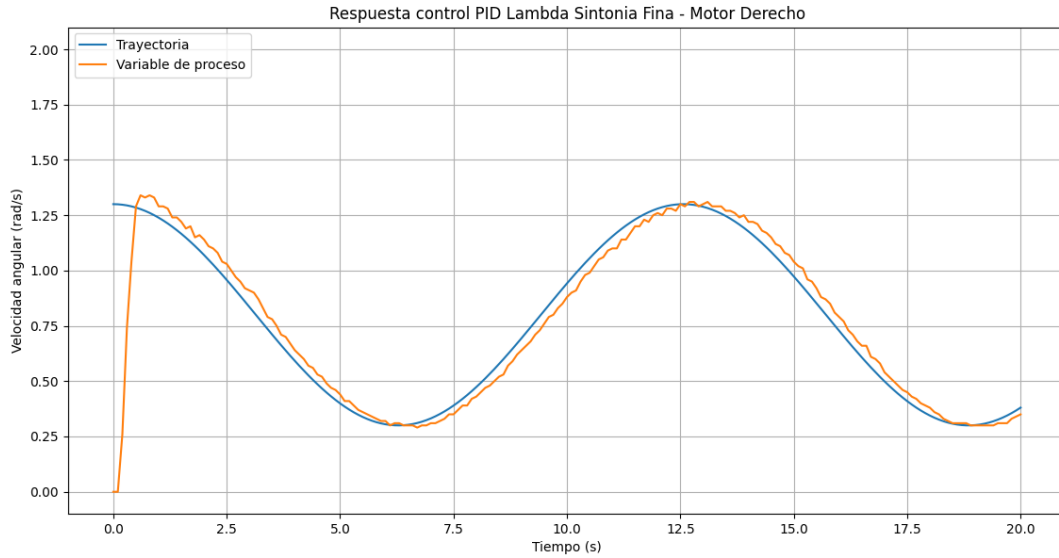


Figura 57. Respuesta control PID Lambda Sintonía Fina - Motor derecho

Controlador PID Lambda – Motor izquierdo

Los cálculos de los parámetros del controlador PID Lambda del motor izquierdo se encuentran en la Tabla 8.

Tabla 8. Parámetros del controlador PID Lambda Motor izquierdo

PID	K_p	$k_p' = \frac{\tau}{k \cdot \left(\frac{L}{2} + 3 \cdot \lambda\right)} = \frac{0.1783}{1.879 \cdot \left(\frac{0.1485}{2} + 3 \cdot 0.1783\right)} = 0.155$
	T_i	$T'_i = \tau = 0.1783$
	T_d	$T'_d = \frac{L}{2} = \frac{0.1485}{2} = 0.074$
	K_p	$k_p = k_p' \cdot \frac{T'_i + T'_d}{T'_i} = 0.155 \cdot \frac{0.1783 + 0.074}{0.1783} = 0.2193$
	K_i	$T'_i + T'_d = 0.1783 + 0.074 = 0.2523$
	K_d	$\frac{T'_i \cdot T'_d}{T'_i + T'_d} = \frac{0.1783 \cdot 0.074}{0.1783 + 0.074} = 0.0522$

En la Figura 58 se muestra la respuesta del del motor izquierdo ante una entrada escalón en lazo cerrado aplicando el controlador PID por el método Lambda.

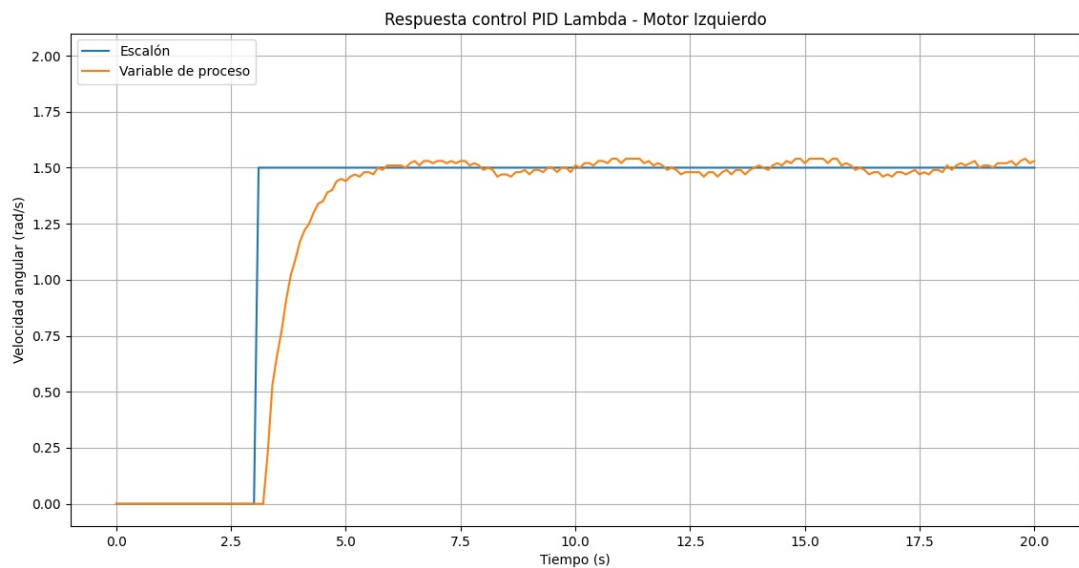


Figura 58. Respuesta al escalón PID Lambda Motor izquierdo

En la Figura 59. se muestra la respuesta de la función del motor derecho ante una trayectoria en lazo cerrado aplicando el controlador PID por el método Lambda.

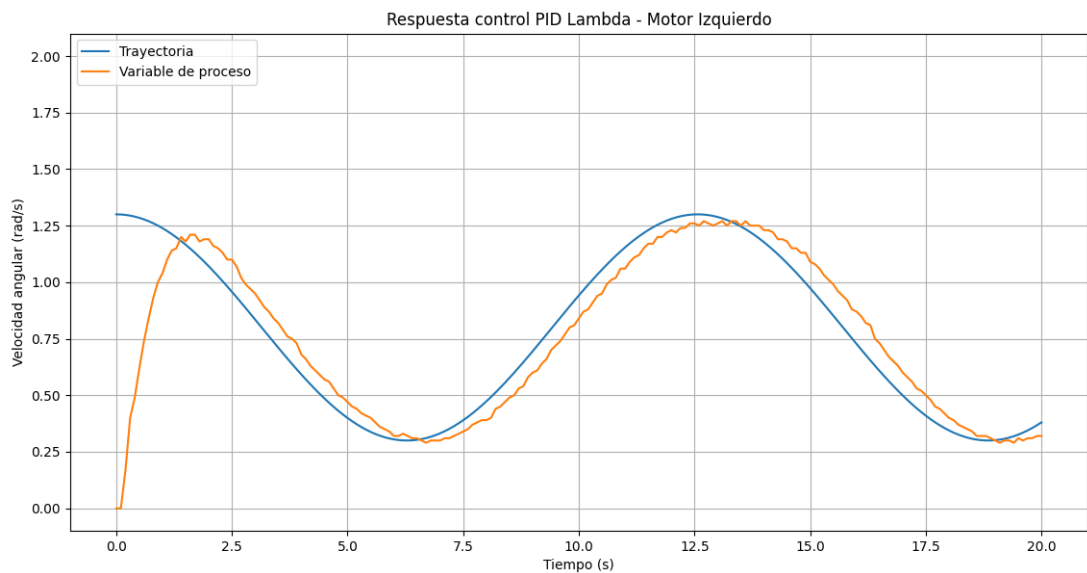


Figura 59. Respuesta a la trayectoria PID Lambda Motor izquierdo

Al igual que en el motor derecho se realizó una sintonía fina y se obtuvieron los nuevos valores del controlador del control PID Lambda

($k_p = 0.28$; $k_i = 0.10$; $k_d = 0.03$), y como se puede evidenciar en la Figura 60 mejora su respuesta a la trayectoria.

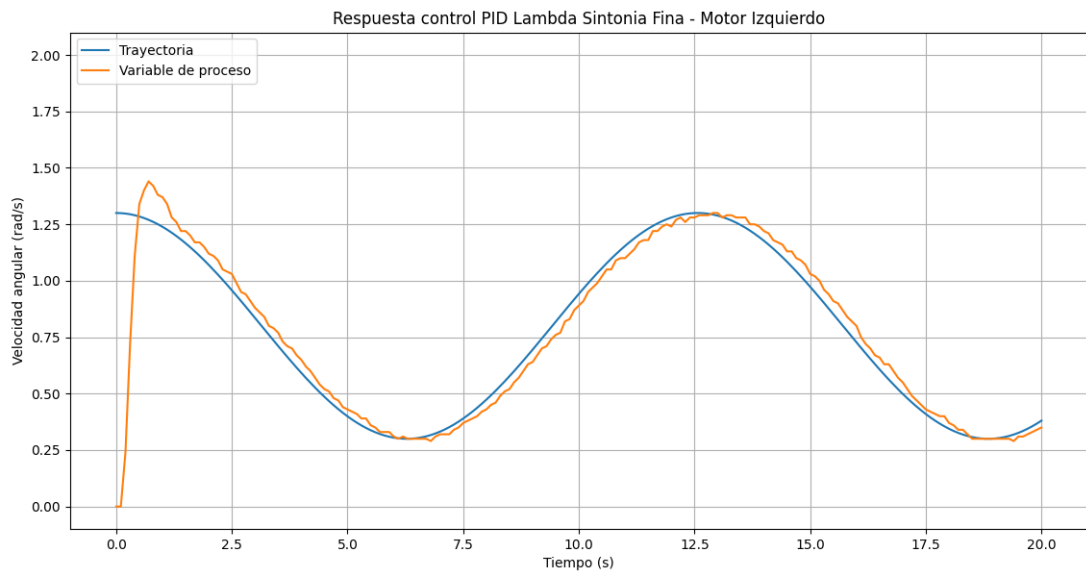


Figura 60. Resposta control PID Lambda Sintonía Fina - Motor izquierdo

2.3.4 CINEMÁTICA DIFERENCIAL

La cinemática diferencial se la utilizó con el objetivo de alcanzar una autonomía en el movimiento del robot siguiendo los surcos de cultivo. Para lograrlo, se inició con el desarrollo del modelo cinemático ya que al ser un robot móvil cumple con especificaciones de desplazamiento de un punto a otro en un plano.

De manera inicial se ubicó el robot en una posición inicial en el sistema de referencias o coordenadas globales (x, y) y se puede apreciar la posición de origen del robot con una orientación de cero grados con respecto al eje x , el origen de coordenadas está centrado en el eje central de las ruedas del robot y luego se lo desplazó a una posición de punto de control $h(x,y)$ para realizar desplazamientos en las coordenadas x_1, y_1 y también un cambio de orientación θ , esto se muestra en la Figura 61.

El objetivo de utilizar cinemática diferencial fue establecer la relación existente entre las velocidades del punto de interés o control y las velocidades de los actuadores o motores para determinar donde se encuentra el robot en el plano de referencias. Este desplazamiento se logró gracias a los dos grados de libertad que posee el robot: velocidad lineal permite que el robot se desplace hacia adelante y hacia atrás y velocidad angular que permite girar al robot en sentido horario u antihorario.

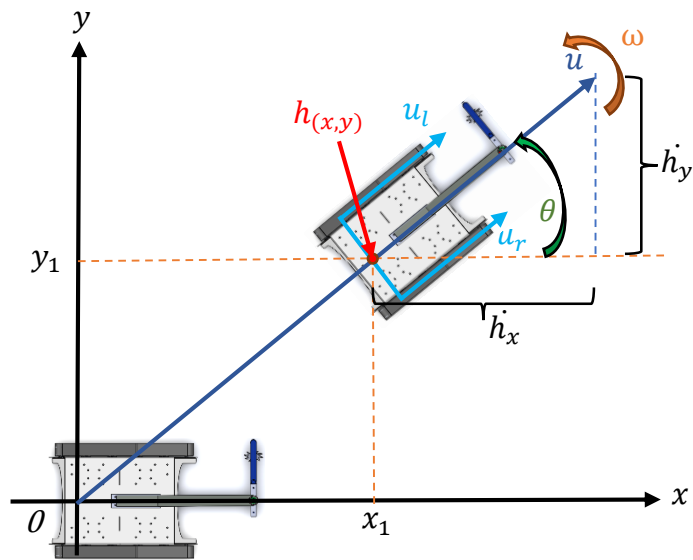


Figura 61. Modelo cinemático de un robot móvil diferencial.

Para poder usar el modelo cinemático se debe relacionar el punto de interés $h(x, y)$ con las velocidades globales del robot (u, ω) por medio de geometría para poder plantear un sistema de ecuaciones. La componente en h_x es igual al desplazamiento x_1 y la componente en h_y es igual al desplazamiento y_1 . Con los parámetros anteriormente establecidos se encontró el modelo cinemático del robot donde h_x, h_y y θ determinaron las posiciones actuales del punto de control y se establecieron las siguientes ecuaciones:

$$h_x = x_1 \quad [38]$$

$$h_y = y_1 \quad [39]$$

$$\theta \quad [40]$$

Donde:

h_x : coordenada en x del punto de interés [m]

h_y : coordenada en y del punto de interés [m]

x_1 : desplazamiento en x [m]

y_1 : desplazamiento en y [m]

θ : ángulo de orientación del robot [rad]

Se relacionaron las velocidades del punto de interés con las velocidades globales del robot; por lo tanto, para obtener estas velocidades se derivaron las posiciones mediante las ecuaciones:

$$\dot{h}_x = u * \cos(\theta) \quad [41]$$

$$\dot{h}_y = u * \sin(\theta) \quad [42]$$

$$\dot{\theta} = \omega \quad [43]$$

Donde:

\dot{h}_x : primera derivada del punto de interés en x

\dot{h}_y : primera derivada del punto de interés en y

u : velocidad lineal global del robot [m/s]

$\dot{\theta}$: primera derivada de la orientación del robot

ω : velocidad angular global del robot [rad/s]

El modelo cinemático base del robot diferencial se representa en la forma matricial, donde se relacionan las velocidades del punto de interés con las velocidades globales del robot multiplicadas por la matriz Jacobiana.

$$[\dot{h}] = [J][\dot{q}] \quad [44]$$

$$\begin{bmatrix} \dot{h}_x \\ \dot{h}_y \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} \quad [45]$$

Se debió relacionar las velocidades lineales y angulares globales del robot con las velocidades de cada motor, como se observa en la Figura 62 mediante las siguientes ecuaciones:

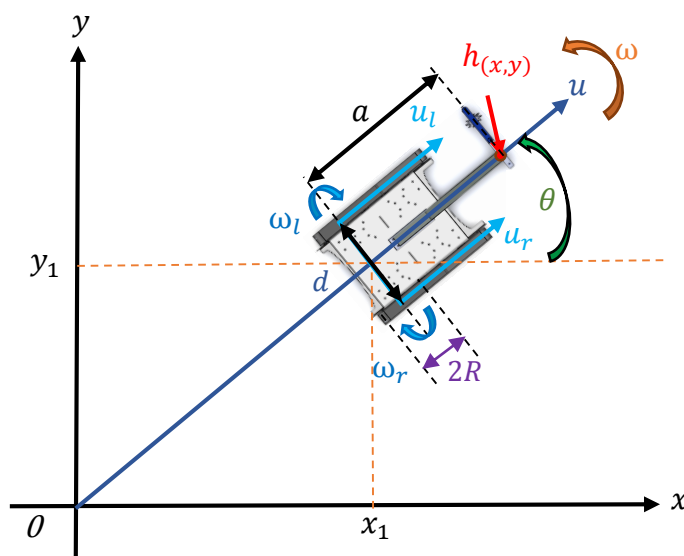


Figura 62. Desplazamiento y orientación robot móvil diferencial.

$$u = \frac{u_r + u_l}{2} \quad [46]$$

$$\omega = \frac{u_r - u_l}{d} \quad [47]$$

Donde:

d: Distancia de separación entre las 2 ruedas motrices

2R: Diámetro de las ruedas

R: Radio de las ruedas

Estas ecuaciones se las puede expresar de forma matricial:

$$\begin{bmatrix} u \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{d} & -\frac{1}{d} \end{bmatrix} \begin{bmatrix} u_r \\ u_l \end{bmatrix} \quad [48]$$

Se obtuvieron también las velocidades lineales de cada rueda aplicando las siguientes fórmulas:

$$u_r = u + \frac{d\omega}{2} \quad [49]$$

$$u_l = u - \frac{d\omega}{2} \quad [50]$$

También se las puede expresar de la forma matricial:

$$\begin{bmatrix} u_r \\ u_l \end{bmatrix} = \begin{bmatrix} 1 & \frac{d}{2} \\ 1 & -\frac{d}{2} \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} \quad [51]$$

La relación entre la velocidad lineal y angular se obtuvo por las siguientes ecuaciones:

$$u_r = R\omega_r \quad [52]$$

$$u_l = R\omega_l \quad [53]$$

Donde:

ω_r : Velocidad angular de la rueda motriz derecha

ω_l : Velocidad angular de la rueda motriz izquierda

2.3.4.1 Modelo cinemático con punto desplazado

Con el objetivo de desplazar el punto de control del centro del eje de las ruedas hacia una distancia “a” como se observa en la Figura 63. Se realizó un desplazamiento a 50 cm a partir del eje central de las ruedas motrices, de esta manera se posicionó el nuevo punto de control a esa distancia debido a que se encuentra en el centro de la herramienta y de esta manera se controla al robot con el punto de control en la herramienta de corte maleza, es decir en su efector final.

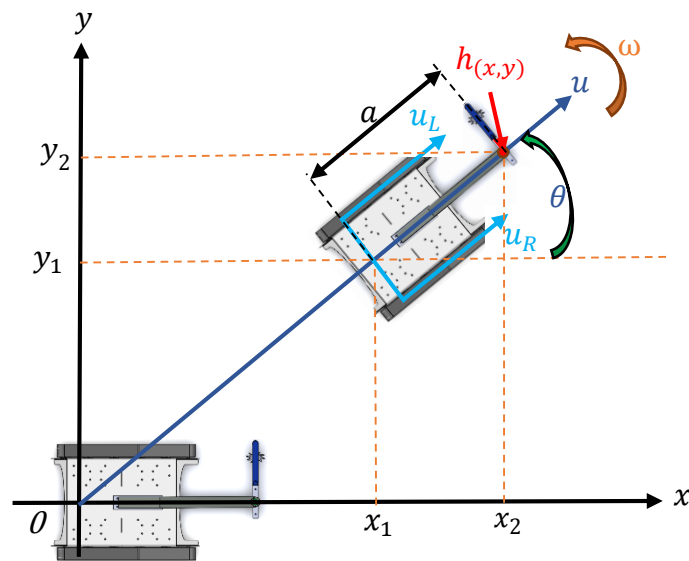


Figura 63. Modelo cinemático con punto de control desplazado.

Cuando se desplazó el punto de control cambiaron las posiciones h_x y h_y sin embargo su orientación θ continuó siendo la misma. De esta manera las posiciones del nuevo punto de control se describen a continuación:

$$h_x = x_1 + x_2 \quad [54]$$

$$h_y = y_1 + y_2 \quad [55]$$

En donde:

$$x_2 = a \cdot \cos(\theta) \quad y_2 = a \cdot \sin(\theta) \quad [56]$$

Derivando las Ecuaciones [54] y [55] además del análisis realizado a las Ecuaciones [41], [42] y [43] del modelo base se obtuvieron:

$$h_x = u \cdot \cos(\theta) - a\omega \cdot \sin(\theta) \quad [57]$$

$$h_y = u \cdot \sin(\theta) + a\omega \cdot \cos(\theta) \quad [58]$$

$$\theta = \omega \quad [59]$$

De esta manera, el modelo cinemático del robot móvil con punto desplazado se lo representa mediante:

$$\begin{bmatrix} \dot{h}_x \\ \dot{h}_y \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -a \cdot \sin(\theta) \\ \sin(\theta) & a \cdot \cos(\theta) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} \quad [60]$$

Una vez que se obtuvo el modelo cinemático del sistema se lo llevó a la forma de una ecuación cinemática diferencial de la siguiente forma:

$$\dot{h} = J\dot{q} \quad [61]$$

Esta ecuación permitió generalizar la relación de las velocidades del punto de control con las velocidades globales del sistema en donde:

\dot{h} : Matriz de velocidades del punto de control con respecto al espacio de trabajo XY.

J : Matriz Jacobiana que representa la relación del desplazamiento del punto de control con el desplazamiento de la posición general del sistema.

\dot{q} : Matriz de velocidades globales del sistema (u, ω)

Por lo tanto, la ecuación cinemática diferencial del sistema esta expresada por:

$$\begin{bmatrix} \dot{h}_x \\ \dot{h}_y \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -a \cdot \sin(\theta) \\ \sin(\theta) & a \cdot \cos(\theta) \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} \quad [62]$$

El controlador se lo diseñó para determinar el conjunto de velocidades \dot{q} que son necesarias para que el robot llegue al punto deseado. Para ejecutar este tipo de control primero se estableció la Ecuación [61] en tiempo continuo de la siguiente manera:

$$\dot{h}(t) = J(q(t))\dot{q}(t) \quad [63]$$

Para obtener las velocidades globales del sistema se las determinaron con la ecuación cinemática diferencial inversa, en la cual se despejó \dot{q} de la Ecuación

[63] e invirtiendo la matriz Jacobiana como se muestra en la siguiente ecuación:

$$\dot{q}(t) = J(q(t))^{-1}\dot{h}(t) \quad [64]$$

La Ecuación [64] permitió obtener las velocidades del robot, pero estas deben pasar a sistemas digitales por lo tanto las velocidades se las transformó a tiempo discreto $\dot{q}(k)$ existiendo siempre una diferencia con las velocidades en tiempo continuo $\dot{q}(t)$. Para solucionar este problema se incorporó un sistema de solución en el que se tuvo en cuenta el error entre la posición y la orientación del punto de control deseado y real.

$$h_e = h_d - h \quad [65]$$

Donde:

h_d : Punto deseado de control

h : Punto real de control

Al derivar el error obtiene:

$$\dot{h}_e = \dot{h}_d - \dot{h} \quad [66]$$

Al despejar \dot{h} de la Ecuación [66] se obtiene:

$$\dot{h} = \dot{h}_d - \dot{h}_e \quad [67]$$

En la Ecuación [67] se reemplazó la Ecuación [64] y se obtuvo como resultado

$$\dot{q}(t) = J(q(t))^{-1}(\dot{h}_d - \dot{h}_e) \quad [68]$$

2.3.5 DISEÑO DE ALGORITMOS DE CONTROL

Se implementó una ley de control con el objetivo de que la respuesta del sistema tenga un comportamiento deseado. En este caso al robot se le asignó una posición final o deseada y como respuesta se obtuvo un desplazamiento hacia el mismo.

2.3.5.1 Teoría de estabilidad de Lyapunov

En el análisis de estabilidad del sistema y diseño de controladores se utilizó la teoría de Lyapunov, una herramienta matemática aplicada a sistemas no lineales y también para aquellos sistemas que varían con el tiempo con el objetivo de encontrar la función candidata que cumpla con las condiciones necesarias que aseguren la estabilidad dentro de sus estados de equilibrio conocidos también como puntos estacionarios, constantes o de reposo ya que si se ubica de manera inicial en el punto de equilibrio ($x = x_e$) luego permanece en el mismo punto ($x(t) = x_e$) para todo tiempo mayor que cero ($t > 0$) (Tzafestas, 2014).

En la función candidata de Lyapunov $V(x)$ se tiene en cuenta que los estados son los errores y su punto de equilibrio es igual a cero, cumpliendo con las siguientes condiciones:

1. La función candidata $V(x)$ es continua y tiene derivadas continuas.

2. Para el valor de $x=0$, la función candidata $V(x)$ sea igual a cero.

$$V(x) = 0 \text{ para } x = 0$$

3. Para valores diferentes de cero $x \neq 0$ la función candidata sea mayor que cero, es decir una función definida positiva.

$$V(x) > 0 \text{ para } x \neq 0$$

4. Para que el sistema sea asintóticamente estable alrededor de su punto de equilibrio debe cumplir las condiciones:

$$\frac{dV(x)}{dt} = 0 \text{ para } x = 0$$

$$\frac{dV(x)}{dt} < 0 \text{ para } x \neq 0$$

Para el diseño de algoritmos de control de posición el cual consistió en que el robot se desplace hacia un punto deseado de referencia, es decir dada la posición inicial del robot el algoritmo de control proporcionó las velocidades al robot con el objetivo que llegue al punto final o punto deseado sin considerar la orientación del robot.

En la ecuación [68]

$$\dot{q}(t) = J(q(t))^{-1}(\dot{h}_d - \dot{h}_e)$$

Se consideró que h_d es un punto fijo por lo tanto constante y su derivada $\dot{h}_d = 0$. Por tanto, se obtuvo la siguiente ecuación:

$$\dot{q}(t) = J(q(t))^{-1}(-\dot{h}_e) \quad [69]$$

La función propuesta para candidata de Lyapunov, fue el error cuadrático dividido entre dos y se la consideró de la siguiente manera de forma matricial:

$$V(h_e) = \frac{h_e^T h_e}{2} \quad [70]$$

A esta función candidata se le obtiene la primera derivada con respecto al tiempo, obteniendo la siguiente expresión:

$$\dot{V} = h_e^T \dot{h}_e \quad [71]$$

Para este caso es simple analizar los errores, ya que estos pueden ser positivos, negativos o iguales a cero, pero en la parte de derivadas de errores se realizó un cambio para que solo quede en función de los errores sin sus derivadas, de la siguiente manera:

$$\dot{h}_e = -Kh_e \quad [72]$$

Al reemplazar la Ecuación [72] en la Ecuación [71] se obtuvo la siguiente expresión:

$$\dot{V} = -h_e^T Kh_e < 0 \quad [73]$$

Se considera a la matriz de ganancia K (matriz diagonal) como una matriz definida positiva, el sistema cumplió para que sea asintóticamente estable. Por último, se reemplazó en la Ecuación [69] la Ecuación [72] para obtener la ley de control, sabiendo que es asintóticamente estable.

$$\dot{q}(t) = J(q(t))^{-1}Kh_e \quad [74]$$

La Ecuación [74] permitió proveer la ley de control final donde el sistema calcula las velocidades lineales y angulares para que llegue al punto de control deseado.

2.3.6 CONTROLADOR POR VISIÓN ARTIFICIAL

Para la detección de cultivos se aplicó inteligencia artificial (IA) como lo es el algoritmo de *Machine Learning* (ML). Adicionalmente se utilizó dos métodos el primero permite la detención de color y segundo es *Haar-Cascade* el cual es un algoritmo de aprendizaje automático que se entrenó para que detecte un objeto de interés en este caso lechugas.

2.3.6.1 Adquisición del conjunto de imágenes

Para la captura de una imagen se utilizó el proceso de adquisición digital de imágenes que se puede apreciar en la Figura 64 en donde una fuente de luz emite su radiación al objeto de interés y este absorbe ciertas ondas de luz y refleja otras, las cuales son captadas por la cámara que las transforma en señales eléctricas para la obtención de la imagen digital.

La imagen capturada es una función de dos variables, las cuales son las filas y las columnas, obteniendo una matriz de $M \times N$ valores. Para cada combinación de la matriz tiene un distinto valor que será la intensidad de color, a cada una de estas muestras se las conoce como pixel, a su vez el número de pixeles de la imagen determina su resolución.

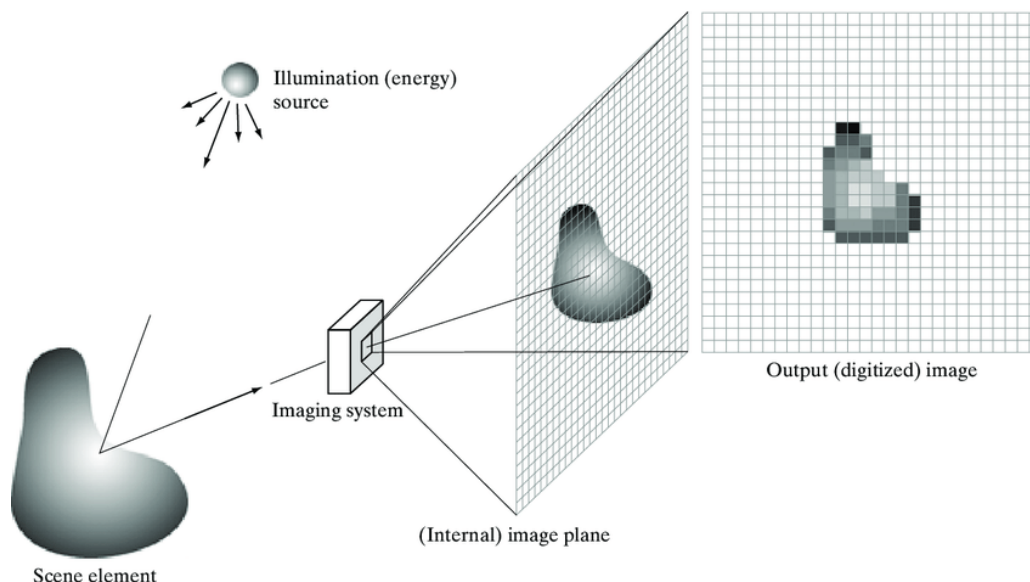


Figura 64. Ejemplo del proceso de adquisición de una imagen digital.
(Poljak, 2013)

Dentro del proceso para adquirir imágenes se utilizó una cámara ESP32Cam que se puede apreciar en la Figura 65 por la ventaja que presenta al transmitir

imágenes desde un servidor web o mediante el puerto de serie de Arduino del que es programable a través de su ID, procesando de manera embebida las imágenes en la cámara (Martín, 2022).



Figura 65. Cámara ESP32Cam.
(Martín, 2022)

2.3.6.2 Controlador por detector de color

2.3.6.3 Umbralización

Los objetos se detectaron mediante herramientas que detectan características como color y forma. Para ello se usó de la herramienta de Open CV en Python llamada umbralización o *thresholding* la cual sirve para dar umbral a una imagen, el umbral es un tipo de segmentación de la imagen y sirvió para convertir una imagen de color o escala de grises a una imagen binaria, es decir en blanco y negro.

Esta herramienta como se muestra en la Figura 66 permitió seleccionar en la imagen un área de interés así ignorando las partes de la imagen que no interesan como el fondo de la imagen.

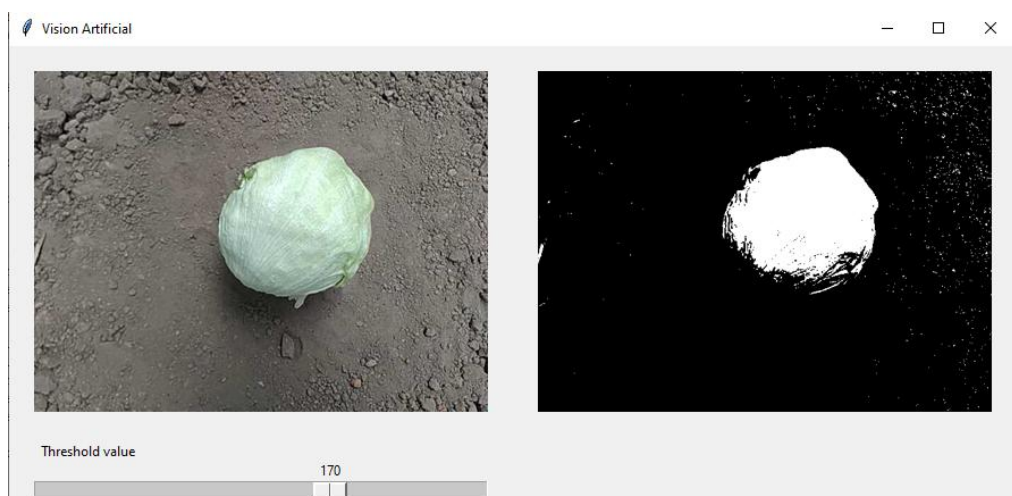


Figura 66. Umbralización.

2.3.6.4 Transformaciones morfológicas

Las transformaciones morfológicas, herramientas de *OpenCV* en *Python* fueron utilizadas en la imagen binaria para realizar operaciones simples basadas en la forma de la imagen, para ello fue necesario tener la imagen original y una máscara o *kernel* que realiza la operación. Estas transformaciones tienen dos operadores morfológicos básicos los cuales son erosión (los píxeles en el contorno de la imagen original son descartados por tanto disminuye su grosor) y dilatación (es la operación contraria a la erosión y aumenta píxeles en el contorno de la imagen original), a partir de ellos se derivan sus variantes que son las combinaciones de las operaciones básicas las cuales son apertura y cierre.

Para la detección de la planta en el suelo fue necesario usar la operación de cierre como se observa en la Figura 67(d), que es la combinación de la operación dilatación seguida de la operación erosión, esto fue de utilidad para cerrar pequeños agujeros o puntos negros pequeños que se encontraban situados dentro del objeto de interés en este caso la planta.

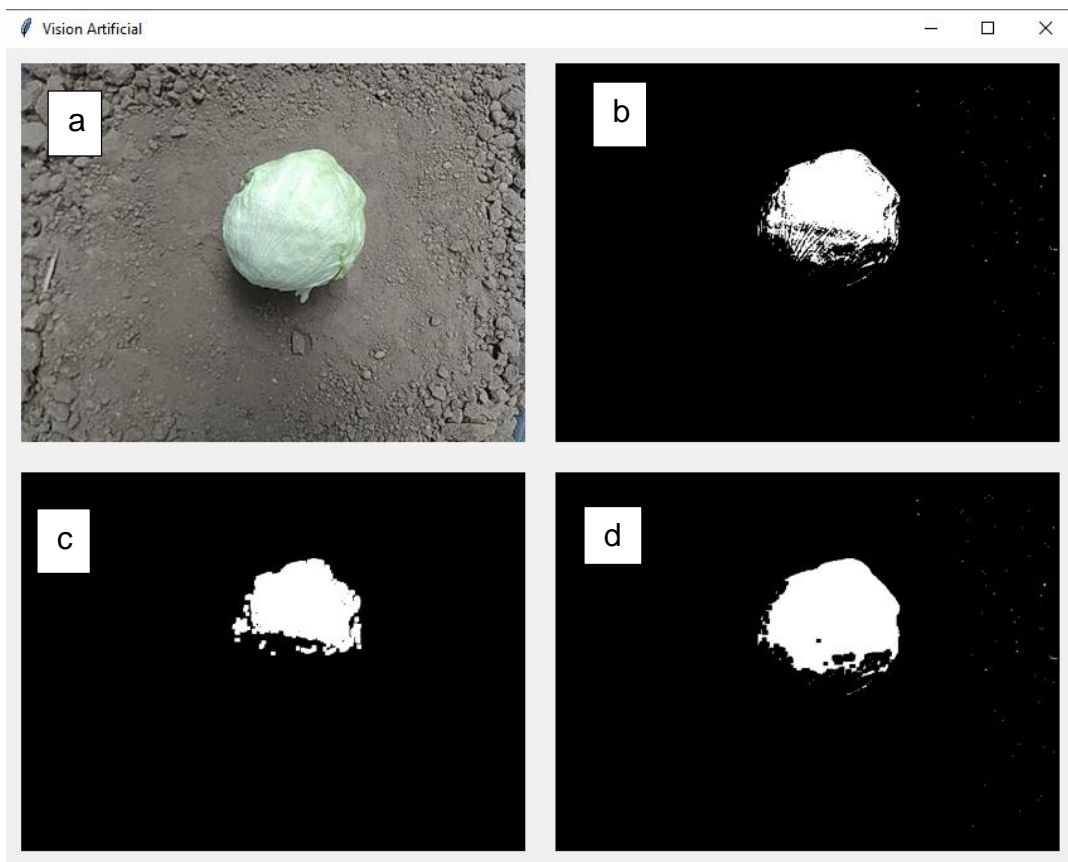


Figura 67. Transformaciones Morfológicas (a). Imagen RGB. (b) Umbralización. (c) Apertura. (d) Cierre.

2.3.6.5 Contornos

La herramienta de *contours* o contornos se usó para poder detectar los bordes de los objetos blancos en un fondo negro. esta sirvió para la segmentación del objeto en la imagen, detección y reconocimiento. Para el uso de esta herramienta fue necesario tener previamente la operación de cierre ya que la herramienta *contours* trabaja sobre ella y da como resultado la gráfica de una línea en el contorno del objeto en la imagen original como se muestra en la Figura 68 (a).

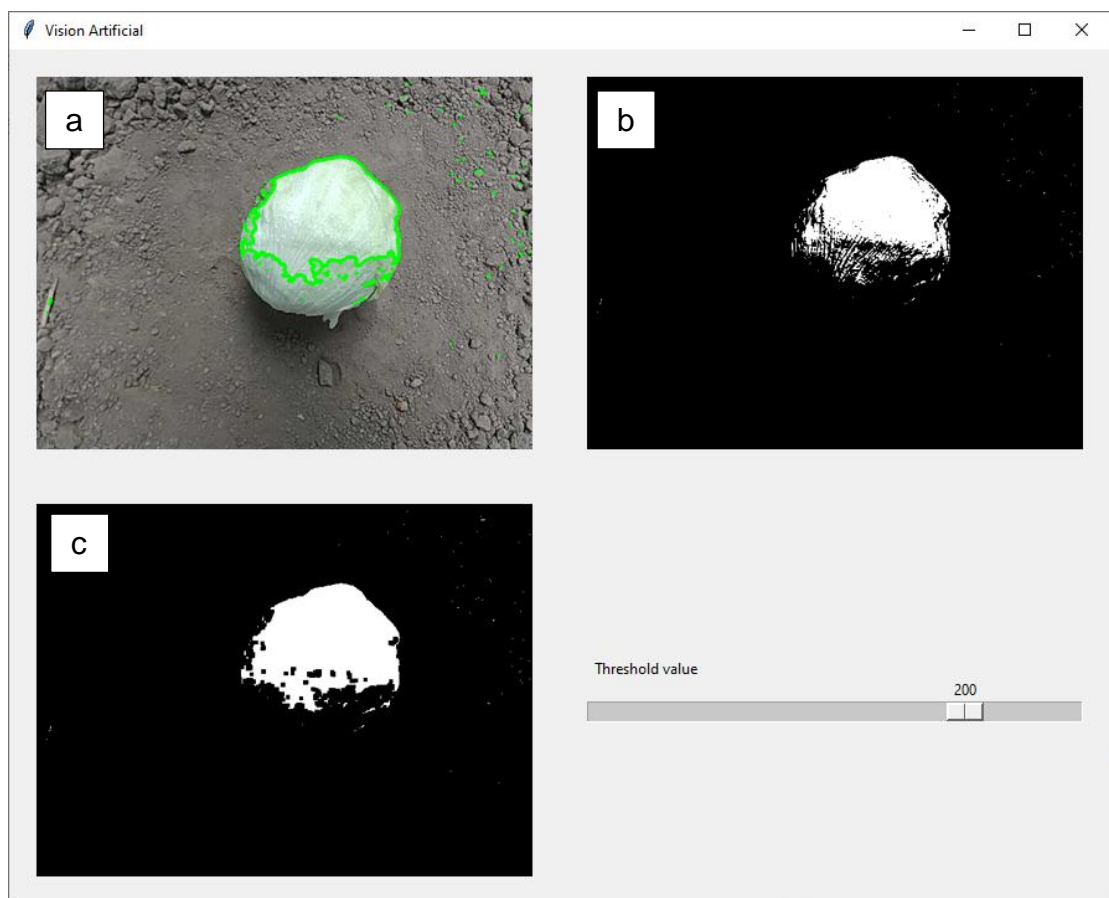


Figura 68. Contornos (a) Imagen RGB con contorno. (b) Umbralización. (c) Contornos

2.3.6.6 Momentos

La herramienta momentos se empleó para calcular el promedio de las intensidades de los píxeles de una imagen permitiendo determinar ciertas características del objeto de interés como el centro de masa o centroide (el cual se marca con un punto rojo) y el área del mismo como se muestra en la

Figura 69 para así poder eliminar ruido ya que si no cumple con un área mínima se lo descarta como objeto de interés.

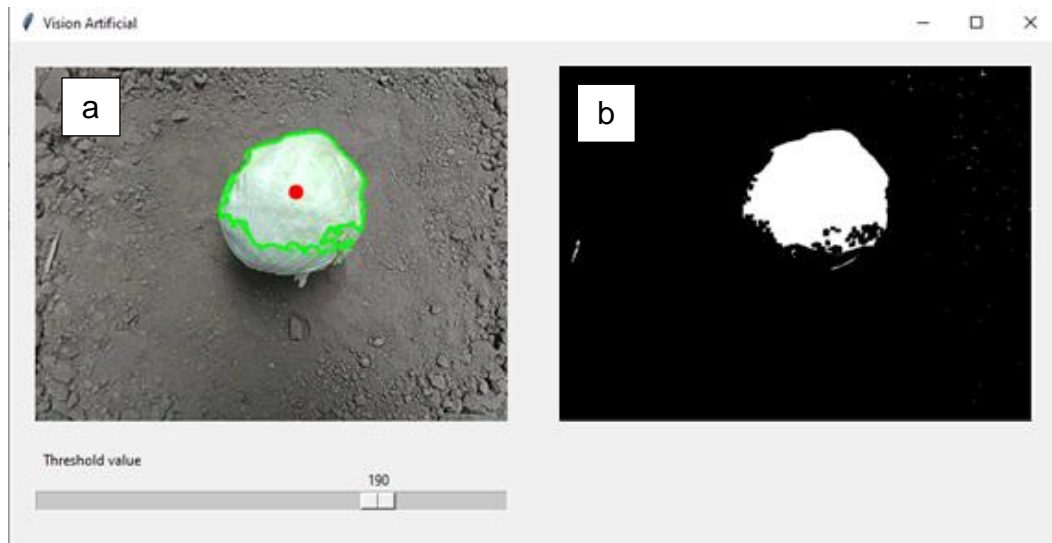


Figura 69. Momentos (a) Imagen RGB con centroide (b) Cierre.

2.3.6.3 Detección de color

Para la detección de colores fue necesario el uso de la herramienta HSV (*Hue* Matiz – *Saturation* Saturación – *Value* Brillo) la cual es una transformación de la imagen en espacio de color RGB (Red – Green – Blue) que es el formato de salida más frecuentemente usado en la adquisición de imágenes por los dispositivos como webcams, cámaras de mano. (Horak, Cip , & Davidek, 2016). Para identificar un color en HSV se utilizó la componente *Hue* que comprende los colores rojo, naranja, amarillo, verde, azul, violeta, y se los sintoniza por valores que comprenden de 0 a 179, las componentes *Saturation* y *Value* tienen un rango de valores de 0 a 255, dichos rangos se pueden observar en la Figura 70.

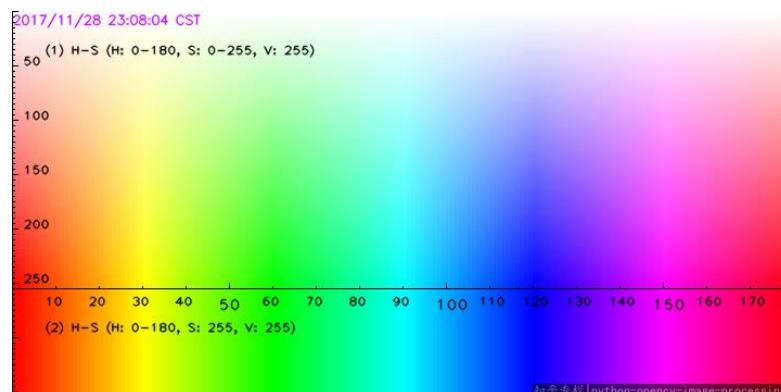


Figura 70. Espacio de color HSV.
(Voug, 2020)

Como referencia para detectar los colores también se utilizaron los valores de la Tabla 9. (Voug, 2020)

Tabla 9. Referencia de rango de colores en HSV

Color	Hue		Saturation		Value	
	Bajo	Alto	Bajo	Alto	Bajo	Alto
Rojo	0	8	65	255	95	255
Naranja	8	18	120	255	50	255
Amarrillo	18	27	50	255	20	255
Verde	27	100	50	255	10	255
Azul	100	130	50	255	50	255
Morado	130	150	20	255	20	255
Rosado	150	180	50	255	50	255

En la identificación de la planta en la componente *Hue* se usó un rango de 35 a 90, en *Saturation* un rango de 20 a 255 y en *Value* de 10 a 255, obteniendo el siguiente resultado como se aprecia en la Figura 71.

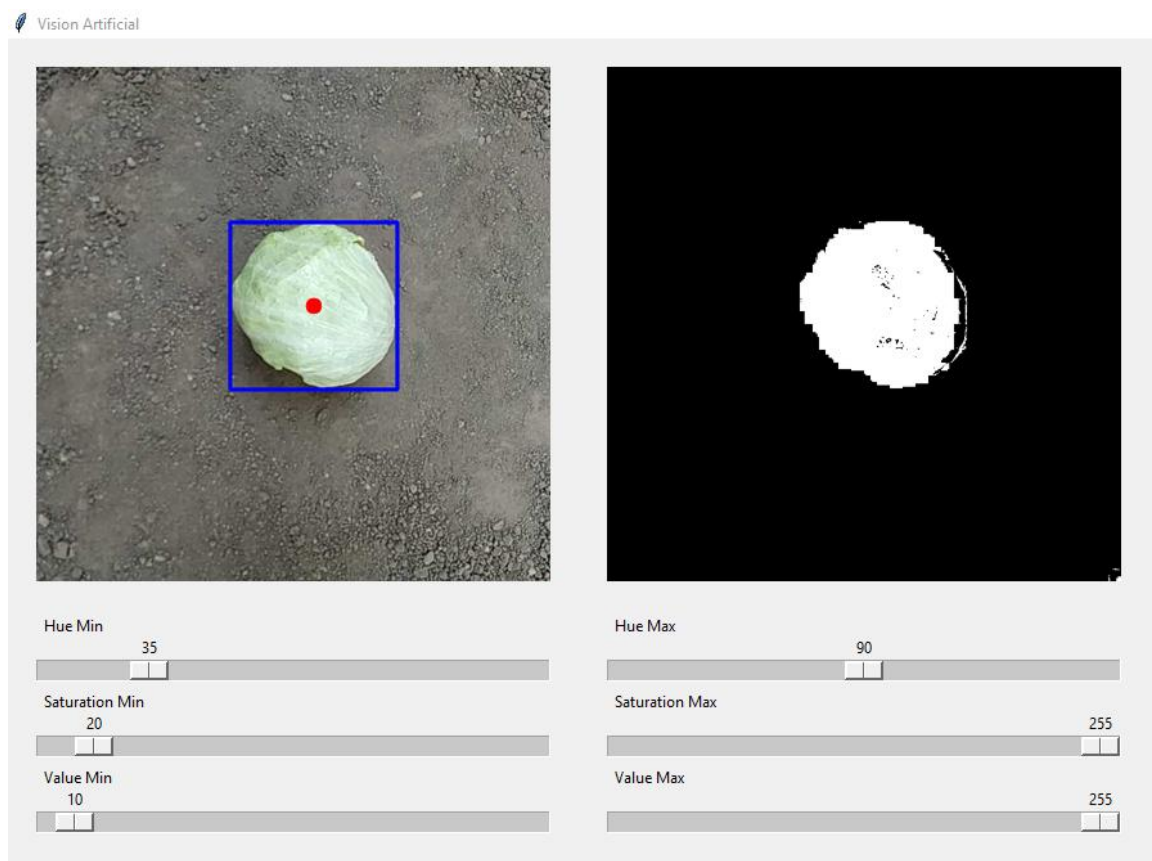


Figura 71. Detección del color verde en espacio HSV.

2.3.6.4 Entrenamiento en Haar-Cascade

El conjunto de datos obtenidos de las capturas realizadas por la cámara ESP32Cam se agruparon en imágenes positivas y negativas. Entre las imágenes positivas se agruparon 150 fotografías de plantas de lechuga que fueron utilizadas para realizar el entrenamiento en *Haar-Cascade*. En la Figura 72 se puede apreciar dicho conjunto de imágenes.

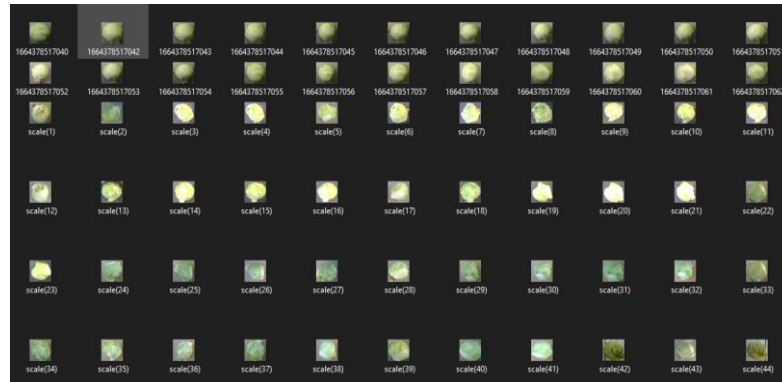


Figura 72. Imágenes positivas para entrenamiento en Haar-Cascade.

Se obtuvieron 50 imágenes negativas del entorno del cultivo como del espacio de terreno y sus alrededores para determinar que objetos no deben ser detectados y así se evitó que el modelo entrenado confunda otros elementos propios del entorno por lechugas. En la Figura 73 se observa la agrupación de imágenes para el entrenamiento en *Haar-Cascade* realizado.

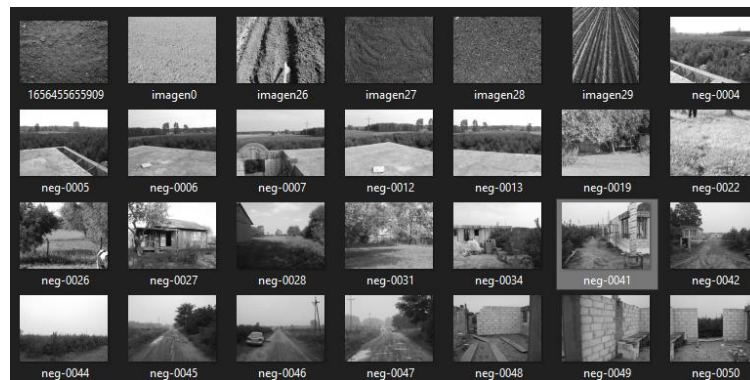


Figura 73. Imágenes negativas para entrenamiento en Haar-Cascade.

Para poder entrenar este algoritmo es necesario realizar un procesamiento previo las imágenes tanto positivas y negativas. A las imágenes positivas se realizó un recorte para solo obtener la imagen de interés y posteriormente se transforma su tamaño de 1200x1200 pixeles a 32x32 pixeles, de la misma manera se realizó un escalamiento de pixeles con las imágenes negativas

siendo el único procesamiento realizado a estas quedando también de un tamaño de 32x32 píxeles. En la Figura 74 se observa cómo se realizó el procesamiento de las imágenes previo a realizar el entrenamiento del algoritmo en *Haar-Cascade*.

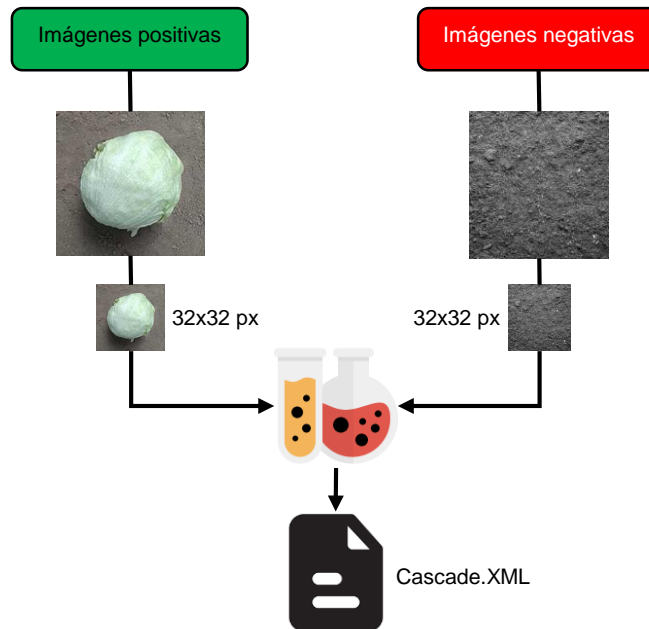


Figura 74. Procesamiento de imágenes positivas y negativas.

Luego de haber obtenido los datos de entrenamiento se creó una carpeta denominada "Entrenamiento Visión" que a su vez almacena dos subcarpetas, la primera "n" en cuyo contenido se encuentran las imágenes negativas y una segunda "p" que contiene las imágenes positivas como se muestra en la Figura 75.

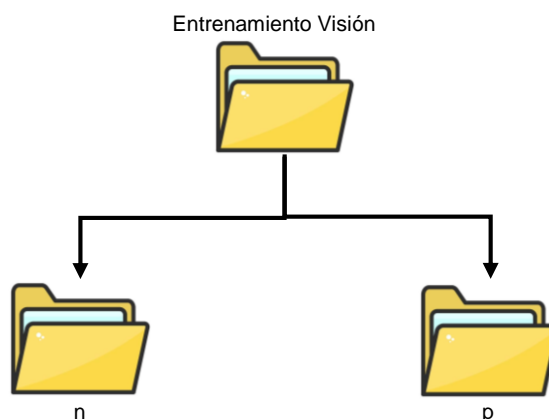


Figura 75. Almacenamiento de imágenes positivas y negativas en *Haar-Cascade*.

Para entrenar la red neuronal de reconocimiento de lechugas se ejecutó el programa *Cascade Trainer GUI*, el cual es una interfaz de programación de aplicaciones (API), que permite de una forma sencilla la configuración de los parámetros necesarios en el entrenamiento del algoritmo *Haar-Cascade*.

En el programa se usó la pestaña *Train* la cual posee a su vez cuatro pestañas de opciones adicionales. En la primera pestaña se le proporcionó la dirección de la carpeta principal donde se encuentran las imágenes tanto positivas (p) como negativas (n). También se requirió ingresar el porcentaje de las imágenes positivas a utilizar en este caso se usó el 100%, y el número total de imágenes negativas (50), como se muestra en la Figura 76.

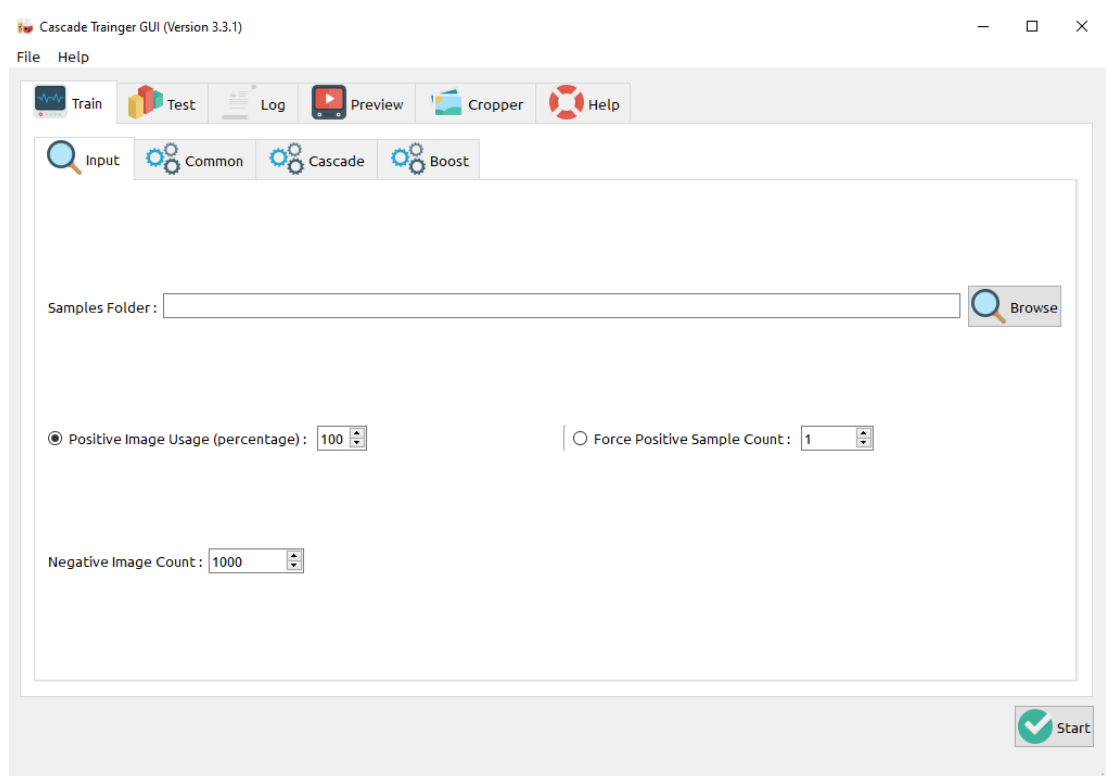


Figura 76. Configuración de ajustes en la pestaña *Train* de *Cascade Trainer GUI*.

En la segunda pestaña denominada *Common* fueron configuradas ciertas opciones según la capacidad de memoria del computador, por defecto se muestra el uso de 1Gb de RAM para el entrenamiento y se aumentó este valor a 4096 que equivale al uso de 4Gb de RAM para realizar el proceso, el resto de opciones se las dejó por defecto como se muestra en la Figura 77.

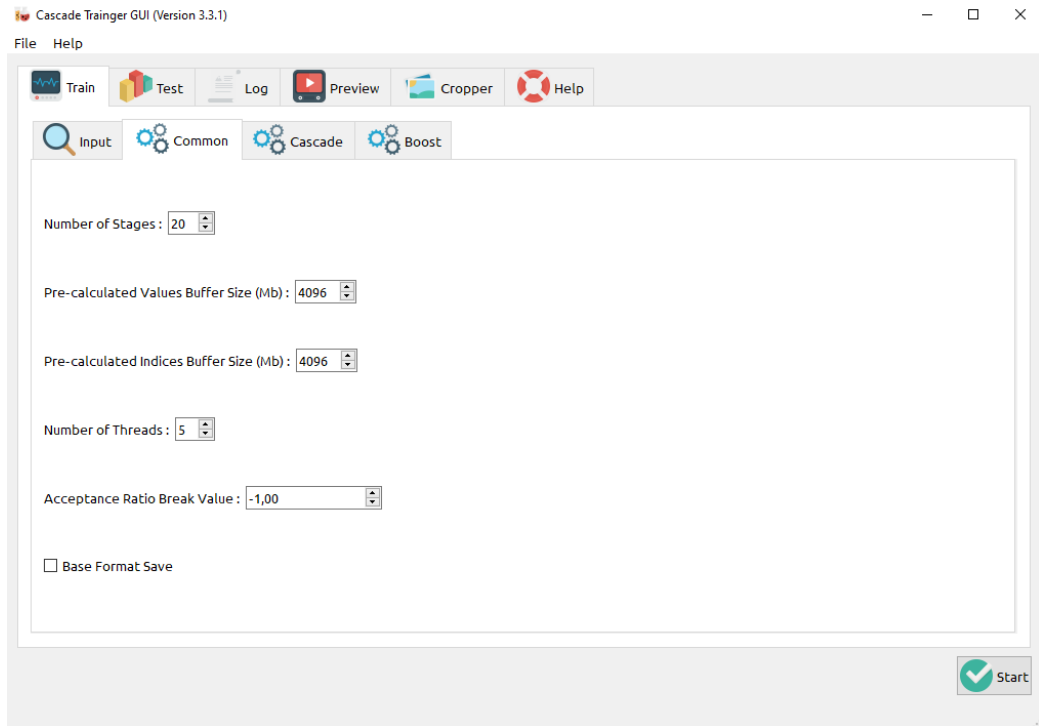


Figura 77. Configuración de ajustes en la pestaña *Common* de *Cascade Trainer GUI*.

En la tercera pestaña, para el establecimiento del tamaño de imágenes se modificó su valor tanto de ancho y alto a 32x32 píxeles, en el tipo de entrenamiento se dejó por defecto *Haar* y en tipo de característica *HAAR* se eligió *CORE*, como se muestra en la Figura 78.

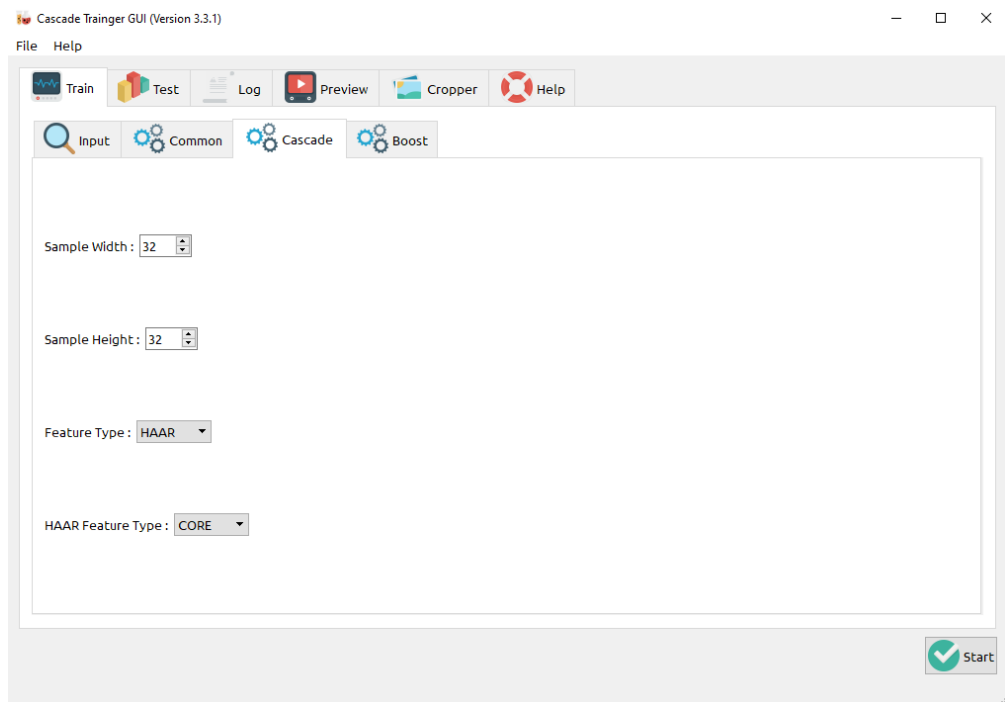


Figura 78. Configuración de ajustes en la pestaña *Cascade* de *Cascade Trainer GUI*.

En la cuarta pestaña todos los valores se mantuvieron por defecto sin modificación alguna, como se muestra en la Figura 79.

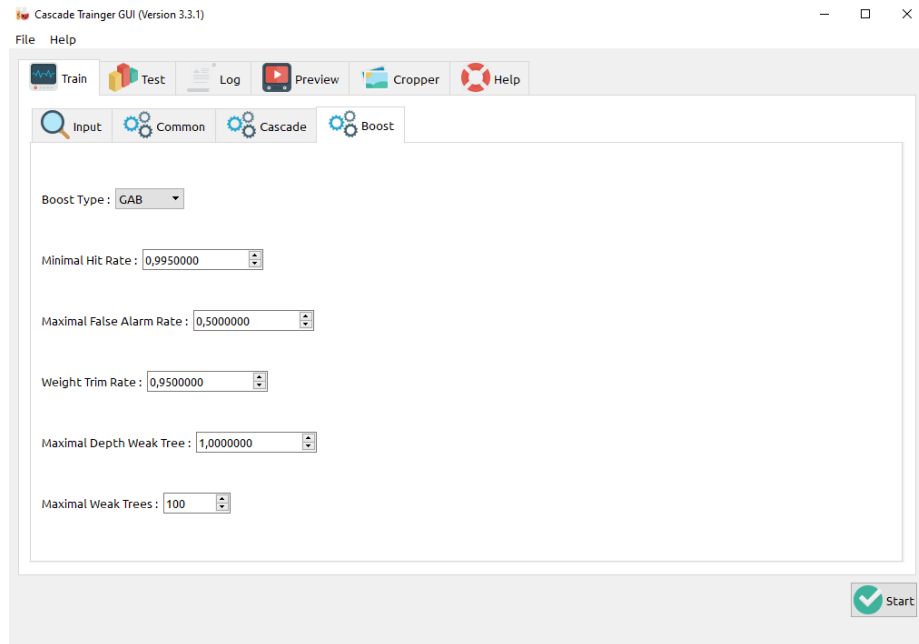


Figura 79. Configuración de ajustes en la pestaña *Boost* de *Cascade Trainer GUI*.

Finalmente, luego de haber establecido todos los parámetros se ejecutó el entrenamiento del programa *Haar-Cascade* como se muestra en la Figura 80 con una duración aproximada de 20 minutos.

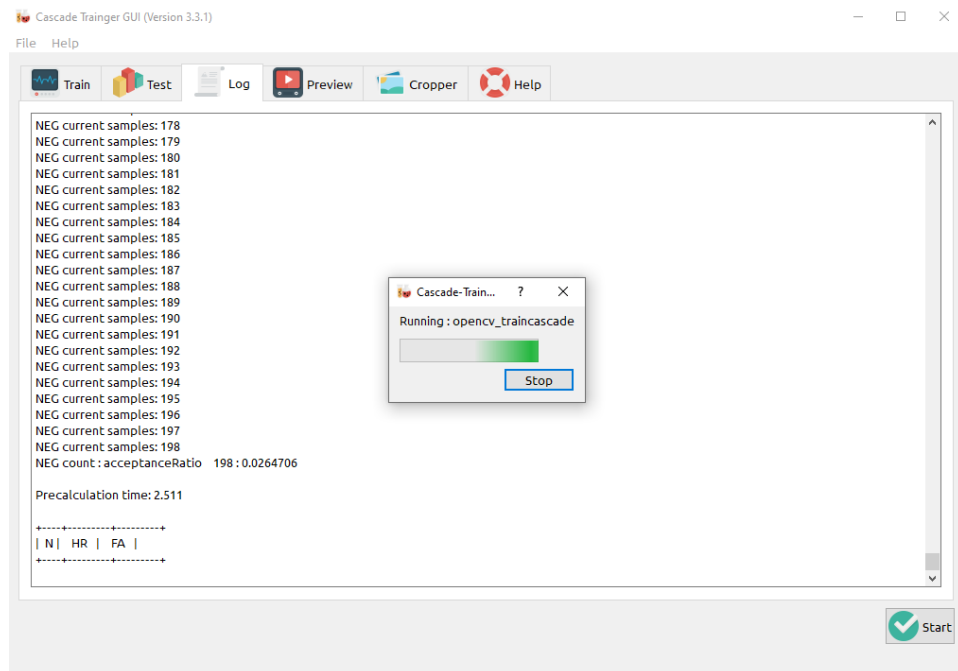


Figura 80. Entrenamiento del programa *Haar-Cascade*.

Al finalizar el entrenamiento se obtuvo dentro de la carpeta principal "Entrenamiento Visión" una nueva llamada "*Classifier*" que contiene los archivos que usó en el entrenamiento y también el archivo "*cascade.xml*" que posteriormente será usado dentro del programa de Python que posee la cinemática diferencial y el controlador por Lyapunov.

2.3.7 PROGRAMACIÓN

En la Figura 81 se muestra el diagrama de flujo de la programación del código que se encuentra en el microcontrolador Esp32 realizado en la herramienta Arduino IDE.

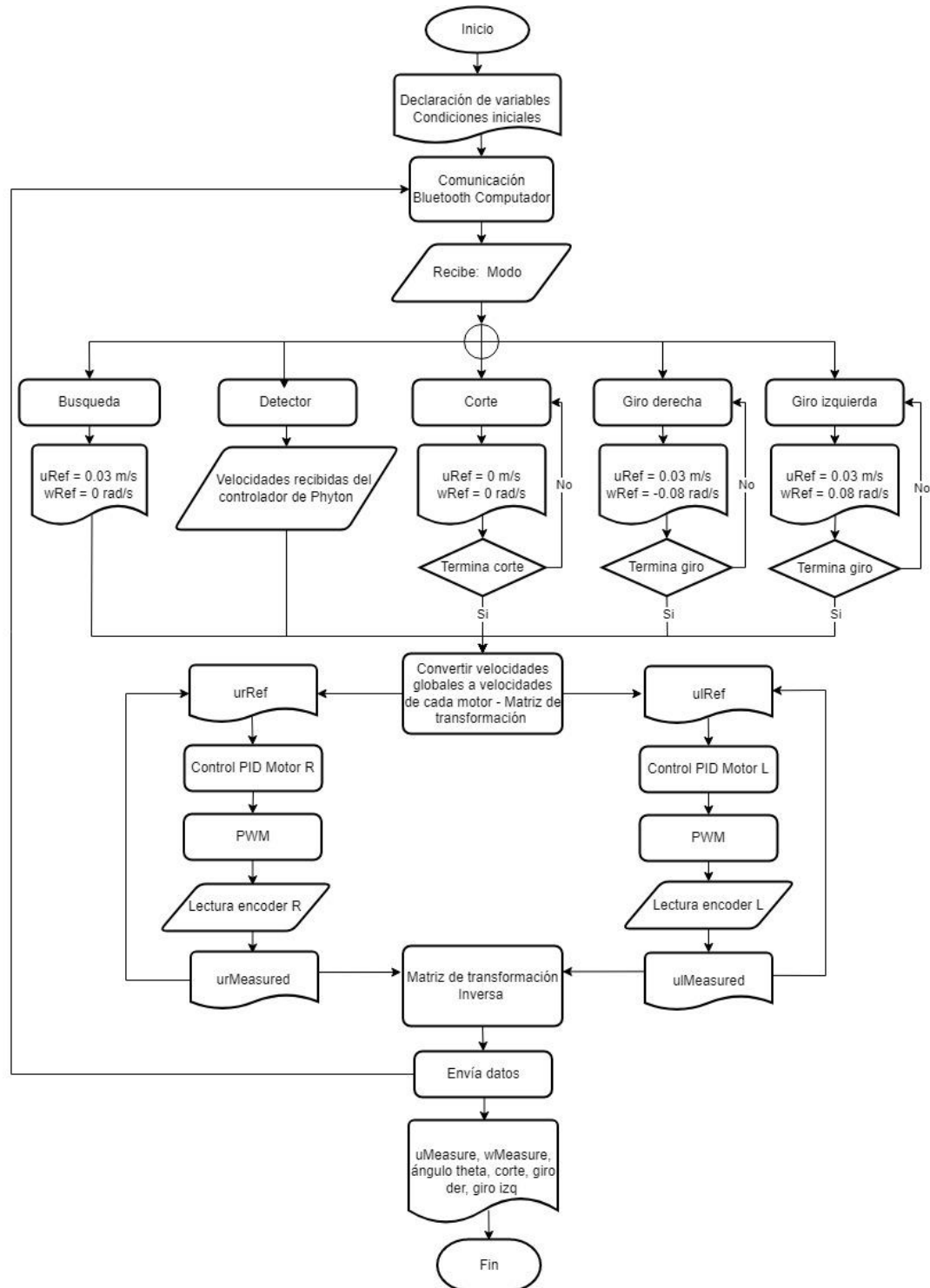


Figura 81. Diagrama de flujo del microcontrolador Esp32 programado en Arduino IDE.

En la Figura 82 se muestra el diagrama de flujo de la programación del código de control por visión artificial mediante los métodos de detección de color y Haar-Cascade, realizados en el programa Python.

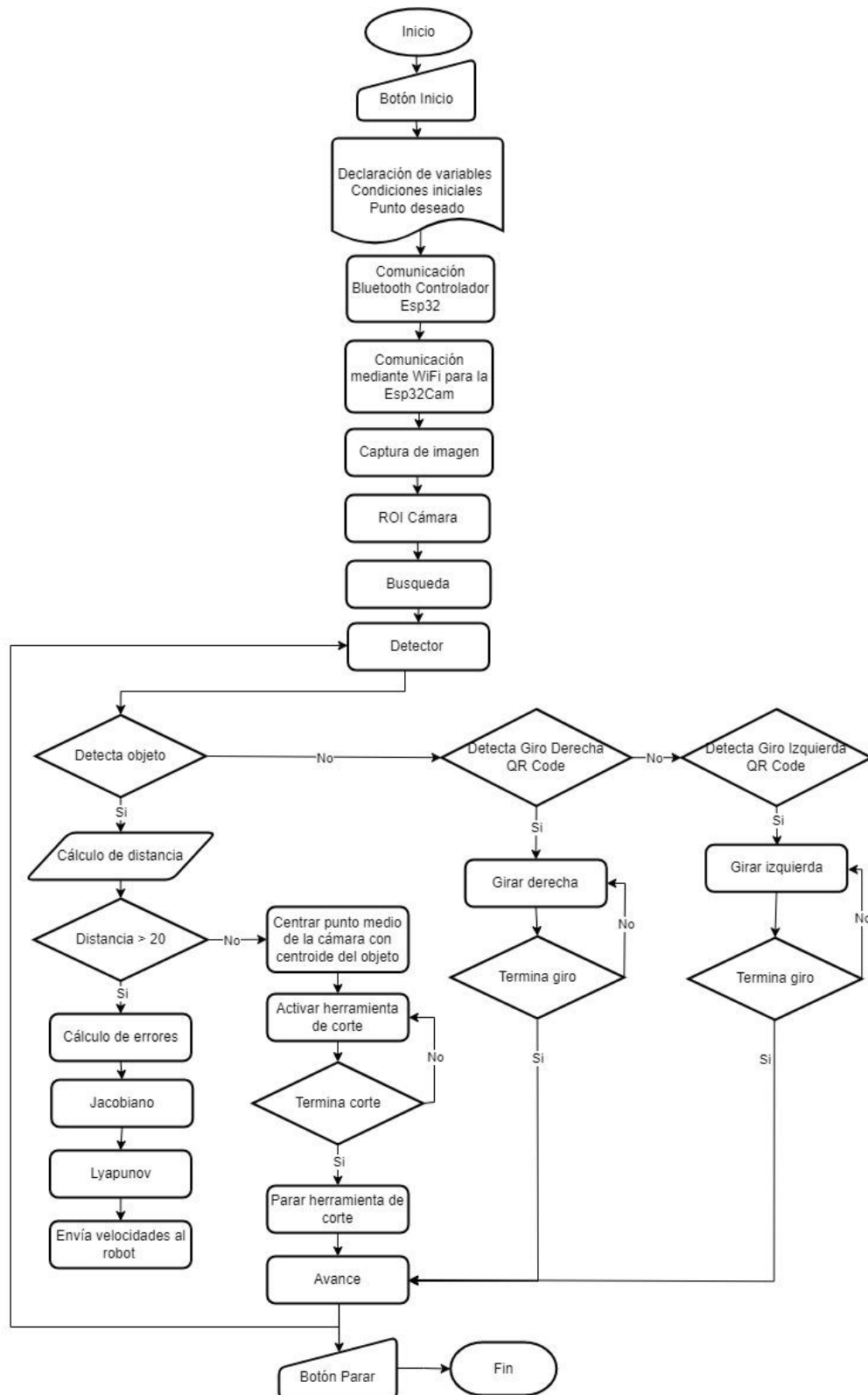


Figura 82. Diagrama de flujo del control por visión artificial mediante los métodos de detección de color y Haar-Cascade, realizados en el programa Python.

3. RESULTADOS Y DISCUSIÓN

3.1 RESULTADOS DEL DISEÑO DEL ROBOT

Como resultado del presente proyecto se obtuvo un sistema autónomo móvil equipado con una herramienta para remover maleza, orugas y sistema de suspensión que permitieron un óptimo desplazamiento además de brindar estabilidad a la estructura durante su recorrido en suelo irregular. En la Figura 83 se puede observar el prototipo final construido.

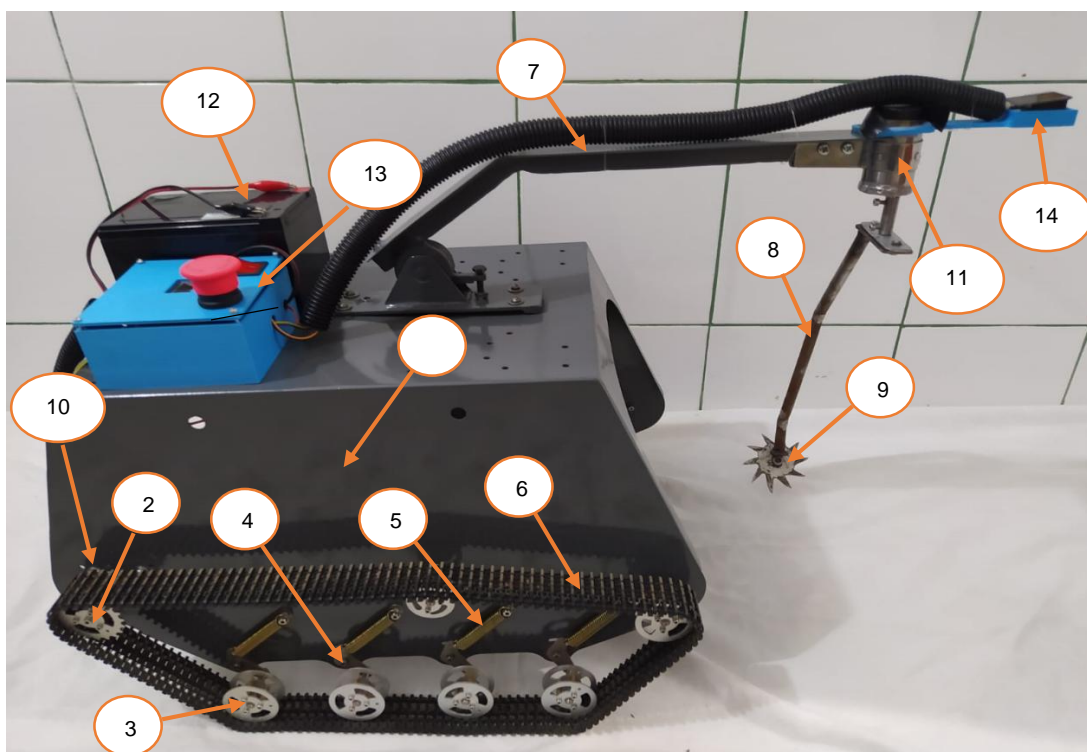


Figura 83. Prototipo final del sistema móvil autónomo eliminador de malezas.

En la Tabla 10 se detalla la lista de los componentes del sistema móvil autónomo.

Tabla 10. Lista de componentes que conforman el sistema móvil autónomo eliminador de malezas.

Lista de componentes				
Ítem	Componente	Material	Cantidad	Medidas
1	Chasis	Acero A36	1	L = 550 mm A = 330 mm H = 320 mm e = 1.4 mm
2	Rueda conductora	Aluminio 7075	2	d = 50 mm e = 2 mm
3	Rueda conducida	Aluminio 7075	10	d = 48 mm e = 2 mm
4	Balancín de suspensión	Aluminio 7075	8	L = 40 mm A = 20 mm e = 2 mm

continua...

continuación

5	Resortes	Latón	8	L = 50 mm d = 0.8 mm
6	Orugas	PLA	2	L = 1.20 m A = 45 mm
7	Brazo metálico	Acero A36	1	L = 450 mm A = 50 mm H = 140 mm
8	Brazo giratorio de efector final	Acero A36	1	L = 400 mm d = 10 mm
9	Herramienta de corte	Acero inoxidable	1	d = 80 mm e = 1.4 mm
10	Motores de tracción con caja reductora	N/A	2	L = 45 mm A = 42 mm H = 130 mm
11	Motor de herramienta con caja reductora	N/A	1	L = 80 mm d = 40 mm
12	Batería	N/A	1	L = 155 mm A = 100 mm H = 100 mm
13	Caja de componentes electrónicos	PLA	1	L = 120 mm A = 120 mm H = 60 mm
14	Soporte de la cámara	PLA	1	L = 150 mm A = 50 mm H = 20 mm

3.2 RESULTADOS DEL CONTROLADOR DE MOTOR PID LAMBDA

El controlador PID por método Lambda fue usado en los motores del robot dando los mejores resultados al seguir una referencia ya que no posee sobre impulso ni oscilaciones en comparación al control PI y PID por el método Ziegler Nichols, tal y como lo menciona (Pruna, Sasig, & Mullo, 2017).

Para obtener una respuesta sin sobre impulso ni oscilaciones por el controlador Ziegler Nichols hay que modificar los valores calculados por fórmulas de una manera considerable y es por ello que este método no se consideró el apropiado para esta aplicación.

En la Figura 84 se muestra la respuesta de la comparativa entre los controladores PI, PID Ziegler Nichols y PID Lambda.

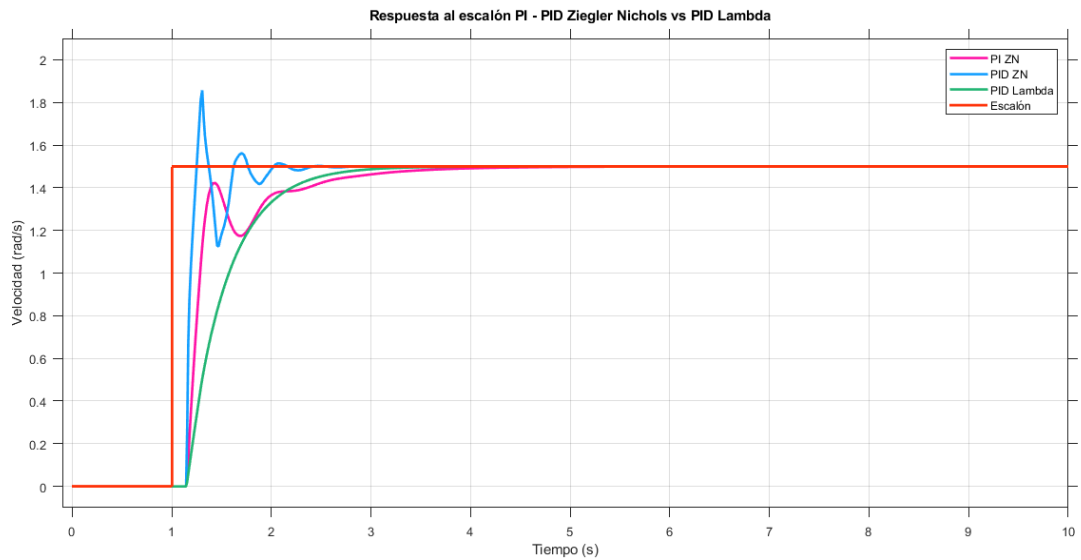


Figura 84. Comparativa entre los controladores PI, PID Ziegler Nichols y PID Lambda.

3.3 RESULTADOS CONTROLADOR DEL ROBOT

Para alcanzar una autonomía en el movimiento del robot se utilizó la ley de control usando el modelo cinemático del robot diferencial, matriz Jacobiana, cálculo de errores y teoría de Lyapunov con el objetivo de determinar el conjunto de velocidades \dot{q} que son necesarias para que el robot llegue al punto deseado controlando las velocidades de los actuadores o motores. De esta manera se obtuvo la ley de control final representada por la ecuación [74]: $\dot{q}(t) = J(q(t))^{-1}Kh_e$ con la cual se comprobó de forma simulada el desplazamiento del robot tipo diferencial hacia cualquier punto deseado en el plano x-y como se puede observar en la Figura 85.

Punto deseado de la simulación: (x=2, y=3)

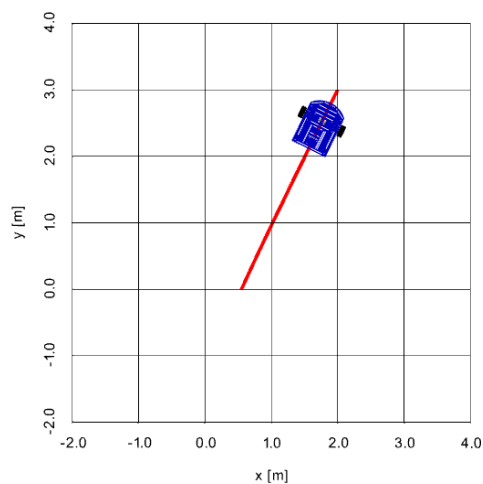


Figura 85. Simulación del desplazamiento del robot diferencial.

En la Figura 86 se muestran las gráficas de la velocidad lineal y angular aplicadas por el controlador para llegar al punto deseado.

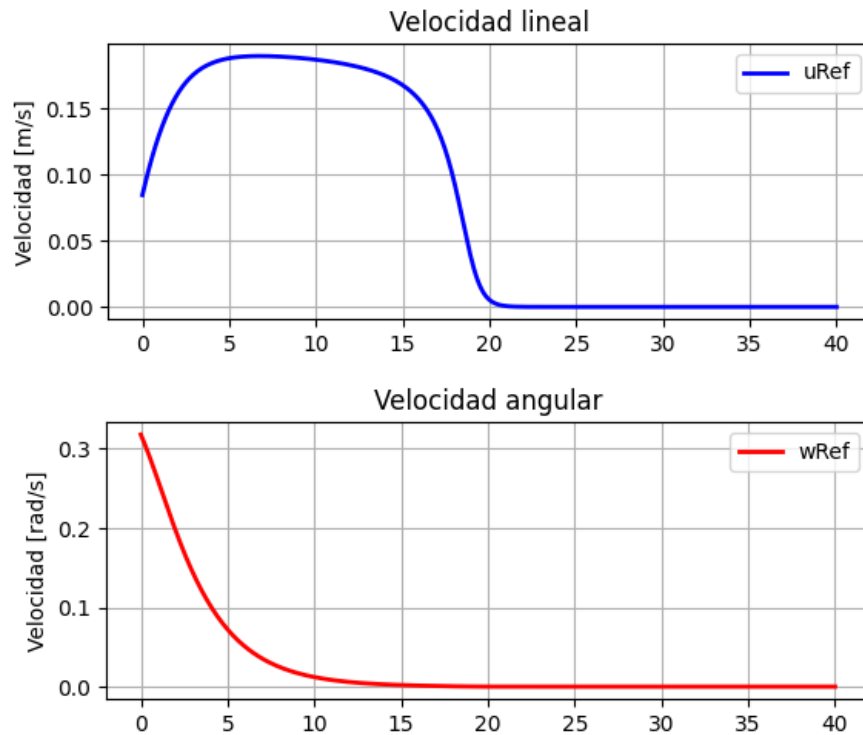


Figura 86. Gráficas de la velocidad lineal y angular.

En la Figura 87 se muestra el error de posición que tiene el robot en el instante inicial y decae a cero cuando llega a su posición deseada.

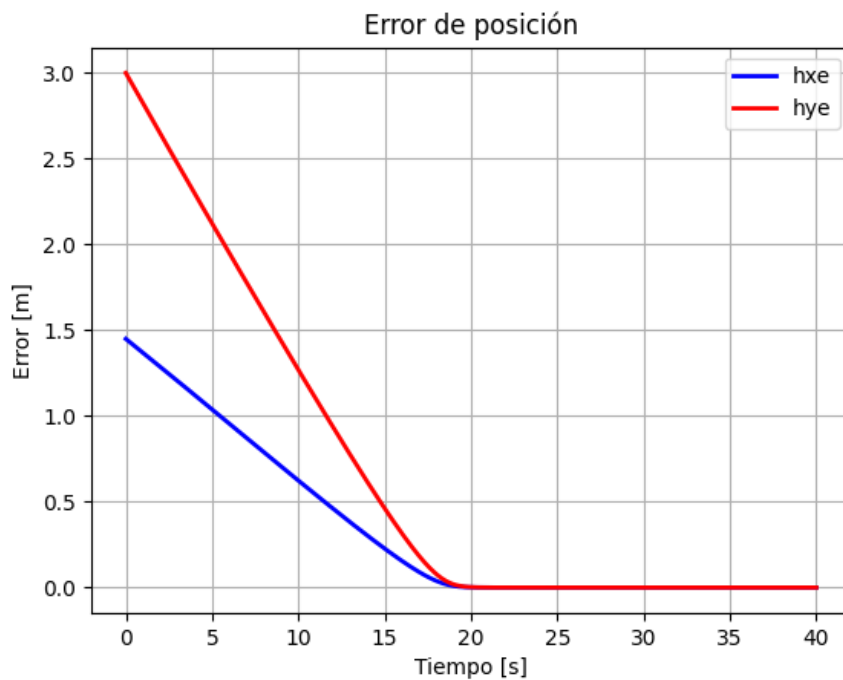


Figura 87. Gráfica de error de posición.

3.4 RESULTADOS DEL SISTEMA DE CONTROL POR VISIÓN ARTIFICIAL

Para la validación del funcionamiento de los modelos entrenados en la detección de lechugas se utilizó el conjunto de imágenes capturadas que representan el 30% de las imágenes totales del conjunto de imágenes; por lo tanto, se obtuvieron 80 imágenes de 10 lechugas en diferente posición e iluminación y 20 imágenes diferentes del suelo de cultivo. A estas imágenes se les denominó como negativas ya que se obtuvieron solo del fondo sin presencia de lechugas, además de la existencia de imágenes donde se indican objetos que pueden aparecer.

En la detección de objetos (lechugas) se utilizaron dos herramientas. La primera destinada a la detección de color como se puede observar en la Figura 88 y la segunda (Haar-Cascade), sin embargo, los mejores resultados obtenidos fueron con la primera herramienta reconociendo la mayoría de los cultivos empleados en pruebas de campo a diferencia de la segunda herramienta que reconoció en menor número los objetos a detectar.

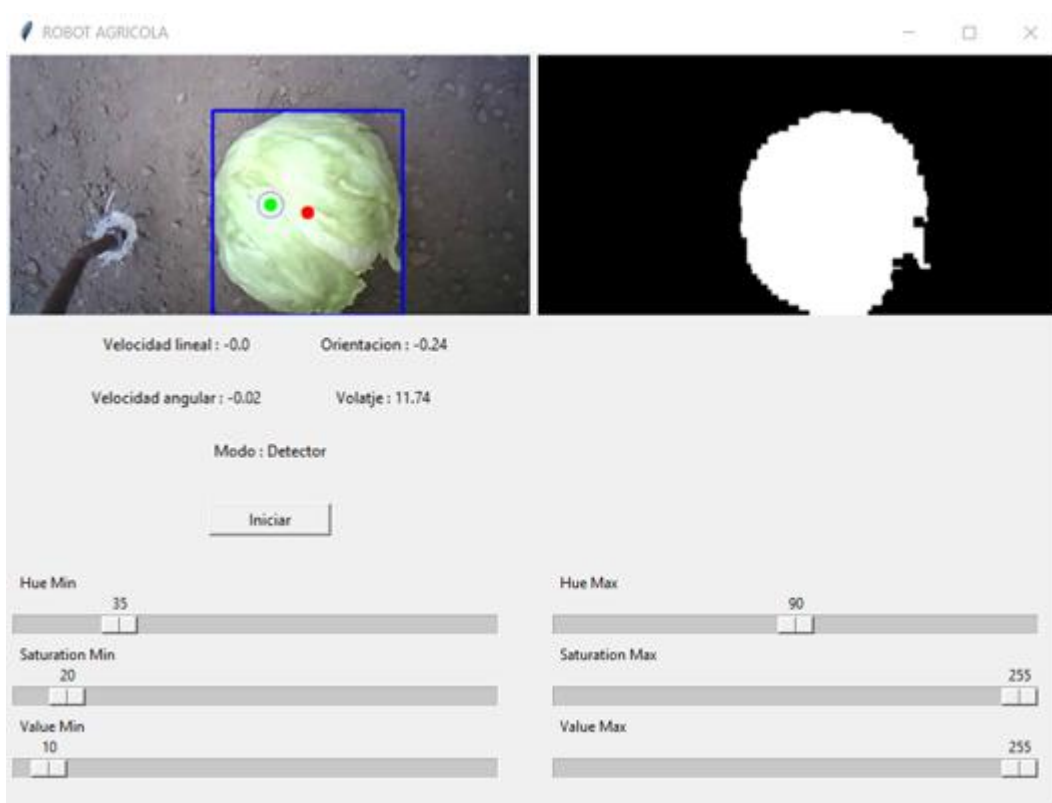


Figura 88. Detector de objeto por color.

3.5 MATRIZ DE CONFUSION

Para evaluar el desempeño obtenido por el algoritmo de Haar Cascade y el algoritmo de detección de color para la detección de las lechugas se realizó una cuantificación de los datos obtenidos en una matriz de confusión para

cada algoritmo de detección, el modelo de predicción da una comparación de las muestras reales obtenidas en campo.

En las Tabla 11 se observa la matriz de confusión para el algoritmo de detección de color.

Tabla 11. Matriz de confusión del detector de color.

Matriz de confusión del detector de color		Predicción		
		Positivo	Negativo	Total
Visualización	Positivo	VP = 76	FN = 4	np= 80
	Negativo	FP = 3	VN = 17	nn= 20
	Total	npp =79	nnp= 21	n= 100

En las Tabla 12 se observa la matriz de confusión para el algoritmo de identificación de objetos *Haar-Cascade*.

Tabla 12. Matriz de confusión de *Haar-Cascade*.

Matriz de confusión de Haar Cascade		Predicción		
		Positivo	Negativo	Total
Visualización	Positivo	VP = 71	FN = 9	np= 80
	Negativo	FP = 7	VN = 13	nn= 20
	Total	npp = 77	nnp= 25	n= 100

Donde:

VP: Verdadero positivo

FN: Falso negativo

FP: Falso positivo

VN: Verdadero negativo

npp: Número total de predicciones positivas

nnp: Número total de predicciones negativas.

np: Número total de positivos

nn: Número total de negativos

n: Número total de datos

En la Tabla 13 se detalla cómo se comportan las variables de la matriz de confusión.

Tabla 13. Comportamiento de variables de la matriz de confusión.

Matriz de confusión	Clase = Lechuga	Predicción
Verdadero Positivo (VP)	Si es (+)	Si la clasificó (+)
Falso Negativo (FN)	Si es (+)	No la clasificó (-)
Falso Positivo (FP)	No es (-)	Si la clasificó (+)
Verdadero Negativo (VN)	No es (-)	No la clasificó (-)

A partir de los datos de la matriz de confusión se calculó la exactitud, tasa de error, sensibilidad, especificidad, precisión y tasa predictiva negativa, mediante las siguientes ecuaciones:

La exactitud calcula cuanto porcentaje de los datos clasificados correctamente:

$$Exactitud = \frac{VP + VN}{n}$$

La tasa de error calcula el porcentaje de los datos clasificados incorrectamente:

$$Tasa\ de\ error = \frac{FP + FN}{n}$$

La sensibilidad: Cuando los datos son positivos cuanto porcentaje logra clasificar:

$$Sensibilidad = \frac{VP}{np}$$

La Especificidad: Cuando los datos son negativos cuanto porcentaje logra clasificar:

$$Especificidad = \frac{VN}{nn}$$

La precisión: Cuando predice positivos cuanto porcentaje clasifica correctamente:

$$Precisión = \frac{VP}{npp}$$

La tasa de predicción negativa: Cuando predice negativos cuanto porcentaje clasifica correctamente:

$$Tasa\ predicción\ negativa = \frac{VN}{npn}$$

En la Tabla 14 se pueden observar los valores calculados con las fórmulas anteriormente expuestas y los datos fueron obtenidos de la matriz de confusión.

Tabla 14. Valores porcentuales de las métricas a partir de la matriz de confusión.

	Detector de color	Haar Cascade
Exactitud	93%	84%
Tasa de error	7%	16%
Sensibilidad	95%	88%
Especificidad	85%	65%
Precisión	96%	92,2%
Tasa de predicción negativa	80,9%	52%

Como se puede observar en la Figura 89, se muestra de manera gráfica los valores calculados de las métricas a partir de la matriz de confusión.

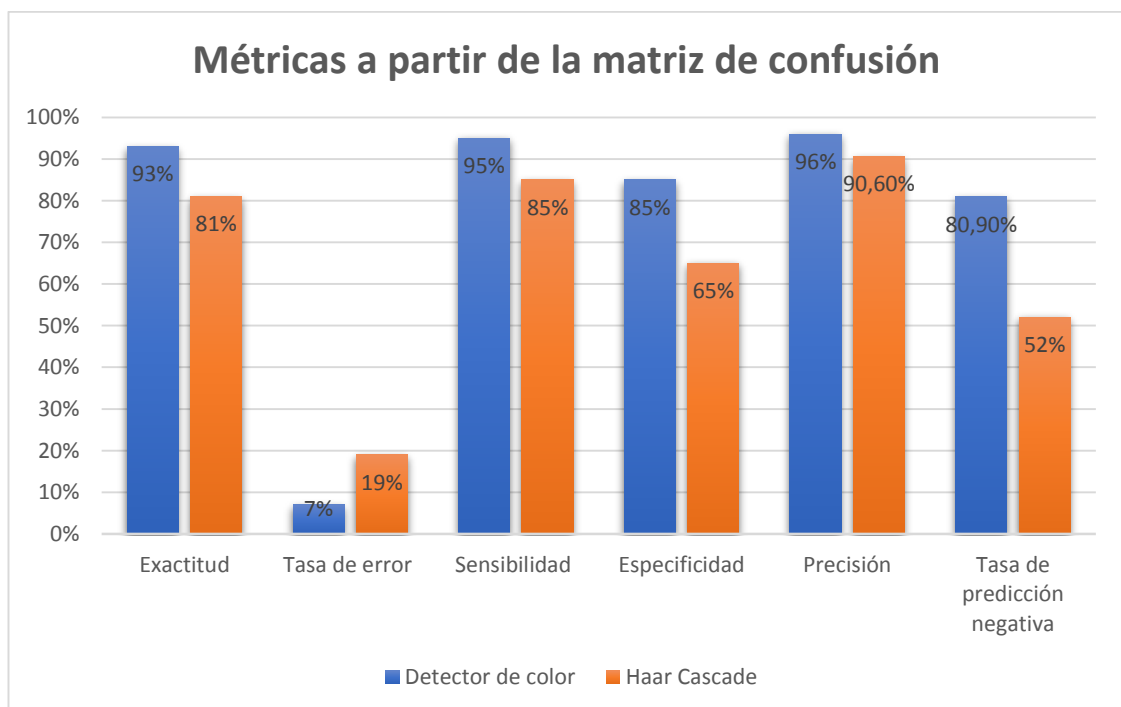


Figura 89. Gráfica de las métricas obtenidas a partir de la matriz de confusión.

4. CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

- Como resultado del diseño de la estructura del robot se obtuvo un chasis al cual se le implementó un sistema de tracción tipo oruga conformado por ruedas motrices tipo engranaje, ruedas conducidas que conforman una distribución trapezoidal y por cadenas de tracción que le permiten una mayor superficie de contacto para una adecuada tracción en el suelo irregular.
- En la presente investigación se utilizó el control PID Lambda para asegurar la funcionalidad correcta de los actuadores o motores ya que presentó una respuesta sin sobre impulso ni oscilaciones lo que lo convierte en el controlador ideal para el uso en robótica por su rendimiento robusto y satisfactorio al seguir una referencia escalón y de trayectoria.
- El diseño del sistema de control se lo realizó mediante el modelo cinemático del robot diferencial, matriz Jacobiana, cálculo de errores y teoría de Lyapunov los cuales se emplearon para determinar un conjunto de velocidades requeridas logrando así el desplazamiento del robot hacia el cultivo identificado, controlando las velocidades de los actuadores o motores.
- En cuanto a la detección de objetos se usaron dos estrategias diferentes las cuales son el algoritmo de identificación de color y el algoritmo *Haar-Cascade* de los cuales, el que demostró el mejor resultado fue el primer método ya que permitió detectar objetos en la gama de color verde con un porcentaje de exactitud del 93% y precisión del 96%.
- Para comprobar el funcionamiento del robot se realizaron pruebas de validación en campo, en las cuales se obtuvieron buenos resultados tanto en desplazamiento por suelo irregular, identificación de cultivo y tareas de remoción de maleza alrededor de la planta.
- Gracias al rendimiento obtenido en pruebas de campo, se pudo observar que la utilización de robots móviles autónomos para la remoción de maleza es una gran alternativa frente a las técnicas tradicionales usadas en la agricultura.

RECOMENDACIONES

- Seleccionar un material distinto al Acero ASTM A36 para reducir el peso de la estructura del chasis del robot.
- Considerar el uso de un computador embebido como la Raspberry Pi 4 que tenga conexión directa con el microcontrolador ya que la comunicación por protocolo Bluetooth es de corto alcance y puede existir una pérdida de datos.
- Reemplazar la cámara ESP32 de conexión por comunicación Wifi ya que puede tener retardos en la transmisión de video, a una de conexión por bus de datos tipo AWM como la cámara Raspberry Pi Noir ideal para visión artificial.
- Añadir más grados de libertad a la herramienta de corte para de esta manera poder regular la altura a la cual se desea realizar la remoción de maleza en terrenos con una inclinación superior a los 10 grados.
- Implementar otro tipo de clasificador en la visión artificial para la identificación de objetos como las redes neuronales convolucionales.
- Aplicar un controlador de seguimiento de trayectoria o camino mediante la asignación de puntos de trayectoria deseada, teniendo en consideración las dimensiones del terreno y separación de las líneas de cultivo.

BIBLIOGRAFÍA

- Alfaro, V. (2006). Identificación de modelos de orden reducido a partir de la curva de reacción del proceso. *Ciencia y Tecnología*, 24(2), 197-216. Recuperado el 01 de 09 de 2022, de <https://revistas.ucr.ac.cr/index.php/cienciaytecnologia/article/download/2647/2598/>
- Ardila, C., Uribe, J., & Meneses, J. (2022). Deméter: Prototipo funcional de robot eliminador de maleza en cultivos rectos de lechuga. *Ingeniería: Ciencia, Tecnología e Innovación*, 9(1), 16-25. doi:<https://doi.org/10.26495/icti.v9i1.2162>
- Bechar, A., & Vigneault, C. (2016). Agricultural robots for field operations: Concepts and components. *Biosystems Engineering*, 94-111.
- Benavides, J. (2016). Diseño de un sistema automático de control mecánico de malezas en cultivos de algodón. Lima. Recuperado el 22 de 08 de 2022, de <http://hdl.handle.net/20.500.12404/6902>
- Bringsmart. (2022). Recuperado el 02 de 08 de 2022, de <http://www.bringsmart.com/col.jsp?id=101>
- Cantero, J. (07 de 2019). Desarrollo y puesta a punto de una plataforma de ensayos de manos protésicas antropomorfas y robóticas. Castellón, España: UNIVERSITAT JAUME I. Recuperado el 17 de 08 de 2022, de http://repositori.uji.es/xmlui/bitstream/handle/10234/184603/TFM_Jesus_Cantero_Ramis.pdf?sequence=1&isAllowed=y
- Cepeda-Camacho, S., Garzón-Muñoz, C., Guasumba-Maila, J., & Oramas-Proaño, D. (04 de 2022). Descripción de las características de los diferentes tipos de baterías utilizadas en los vehículos eléctricos. *Polo del Conocimiento*, 7(4), 376-391. doi:10.23857/pc.v7i4.3830
- Corona, L., Abarca, G., & Mares, J. (2014). Sensores y Actuadores. Patria. Recuperado el 22 de 06 de 2022, de <https://books.google.es/books?hl=es&lr=&id=wMm3BgAAQBAJ&oi=fnd&pg=PP1&dq=actuadores+definici%C3%B3n&ots=6O6okAe0WB&sig=bhuFH3Jn35VbCuQXNLsZ1oD78Z0#v=onepage&q=actuadores%20definici%C3%B3n&f=false>
- Del Pozzo , A. (23 de 09 de 2021). Estudio, diseño y construcción de un levitador acústico para desplazamiento de partículas. Valencia, España: Universitat Politècnica de València. Obtenido de <https://riunet.upv.es/bitstream/handle/10251/175590/Pozzo%20-%20Estudio%20diseno%20y%20construccion%20de%20un%20Levitador%20Acustico%20para%20desplazamiento%20de%20particulas.pdf?sequence=2>

- Dorador-Gonzalez, J. (2009). MECANISMOS DE TRANSMISIÓN Y ACTUADORES UTILIZADOS EN PRÓTESIS DE MANO. *ACADEMIA*, 335-345. Recuperado el 20 de 06 de 2022, de https://d1wqtxts1xzle7.cloudfront.net/7570485/a1_216-with-cover-page-v2.pdf?Expires=1655766429&Signature=F1ub6T~558V8Gzm7dNYxnQfe7SOEGQfktujsxP6TisPRSLgRUNx8hozSURO6~nFXUu5Xe3aG9R1QvBTM-sLlaEqgOnXGUv8Lc7ZqxHWlvEaY3PdMIZMan3AkqDdi9IX-8eAz0eH7nw5oBn99UI9dFg
- Echeverría, H., Guerra, L., & Quiñones, R. (2020). Diseño de un robot móvil teleoperado a distancia para rastreo de personas en un derrumbe de una edificación. Huancayo. Recuperado el 30 de 04 de 2022, de <https://hdl.handle.net/20.500.12394/9069>
- Encinas, J. (junio de 2020). *Control de poblaciones de amapola sensible y resistente a herbicidas sulfonilureas mediante el uso del compuesto orgánico quinato*. Obtenido de https://academica-e.unavarra.es/xmlui/bitstream/handle/2454/37953/B.TFG_Completo_JosuEncinas.pdf?sequence=1&isAllowed=y
- Espinoza, J., Juárez, C., Mota, C., & Romantchik, E. (2021). Control de maleza mediante la robótica. *Revista de Ingeniería Agrícola*, 11(4), 54-66. Obtenido de <https://www.redalyc.org/journal/5862/586268743008/586268743008.pdf>
- Fernández, O. (1982). Manejo integrado de malezas. *Scielo*, 2, 69-79. Recuperado el 31 de 06 de 2022, de <https://www.scielo.br/j/pd/a/Cxn84R98Nt8sx767cXSCZgF/?lang=es&format=pdf>
- García, J. (octubre de 2015). Navegación de un robot móvil sobre terreno irregular con contacto de su brazo con el suelo. Recuperado el 13 de 05 de 2022, de <http://hdl.handle.net/10630/11630>
- Godwin, R., Earl, R., Taylor, J., Wood, G., Bradley, R., Welsch, J. P., . . . Blackmore, S. (2001). Precision farming of cereals: A five year experiment to develop management guidelines. Recuperado el 15 de 05 de 2022, de https://www.researchgate.net/profile/Fabio-Leiva/publication/228425520_La_agricultura_de_precision_una_produccion_mas_sostenible_y_competitiva_con_vision_futurista/links/555ce20508ae8c0cab2a6901/La-agricultura-de-precision-una-produccion-mas-sostenible-y-

- Guamá, S. (07 de 2020). Construcción de un módulo de medición de radiación solar. Quito, Ecuador. Recuperado el 22 de 08 de 2022, de <https://bibdigital.epn.edu.ec/bitstream/15000/21110/1/CD%2010627.pdf>
- Guato, K. (2019). Análisis de las redes de sensores inalámbricos en la agricultura de precisión en el Ecuador. Ecuador. Recuperado el 25 de 05 de 2022, de <http://181.198.35.98/Archivos/GUATO%20FERNANDEZ%20KAREN%20ESTAFANI.pdf>
- Hasan, A., Sohel, F., Diepeveen, D., Laga, H., & Jones, M. (2021). A survey of deep learning techniques for weed detection. *Computers and Electronics in Agriculture*, 184. doi:: <https://doi.org/10.1016/j.compag.2021.106067>
- Hernández, P; Navarrete, J. (2022). *Implementación de un Robot de desinfección ultravioleta para microorganismos con registro visual para asistir en la limpieza de áreas en cuidados intensivos*. Guayaquil, Ecuador. Recuperado el 27 de 06 de 2022, de <http://dspace.ups.edu.ec/handle/123456789/22831>
- Horak, K., Cip , P., & Davidek, D. (2016). Automatic Traffic Sign Detection and Recognition Using Colour Segmentation and Shape Identification. Brno, República Checa: ICIEA. doi:10.1051/mateconf/20166817002
- ICAMEX. (2022). Recuperado el 10 de 08 de 2022, de <https://icamex.edomex.gob.mx/lechuga>
- IEEE Student Branch. (2021). *Micromouse Lab 2*. Recuperado el 12 de 08 de 2022, de Encoders: <https://iee.berkeley.edu/micromouse-lab-2/>
- Instituto Nacional de Estadística y Censos (INEC). (2016). *Información Ambiental en la Agricultura*. Recuperado el 08 de 08 de 2022, de https://www.ecuadorencifras.gob.ec/documentos/web-inec/Encuestas_Ambientales/Informacion_ambiental_en_la_agricultura/2016/PRESENTACION_AGRO_AMBIENTE_2016.pdf
- Kanagasingham, S., Ekpanyapong, M., & Chaihan, R. (2020). Integrating machine vision-based row guidance with GPS and compass-based routing to achieve autonomous navigation for a rice field weeding robot. *Springer Link*, 21, 831-855. doi:<https://doi.org/10.1007/s11119-019-09697-z>.
- Liu, H., Sun, H., Li, M., & Iida, M. (2020). Application of color featuring and deep learning in maize plant detection. *Remote Sensing*, 12(14). doi:<https://doi.org/10.3390/rs12142229>

- Mamani, J. (2020). Diseño y construcción de un sistema automático de seguimiento solar basado en visión artificial aplicado a sistemas fotovoltaicos. Arequipa, Perú. Recuperado el 22 de 07 de 2022, de <http://repositorio.unsa.edu.pe/handle/20.500.12773/12287>
- Martín, J. (2022). Diseño y programación de un sistema de bajo coste basado en la ESP 32-CAM para el control de un brazo robótico. Valencia, España: UNIVERSITAT POLITÈCNICA DE VALÈNCIA. Recuperado el 03 de 09 de 2022, de <http://hdl.handle.net/10251/184354>
- Moreano, G., Cajamarca, J., & Tenicota Garcia, A. (01 de 2020). Agricultura de Precisión: procesamiento y segmentación de imágenes para obtención de una ruta de navegación autónoma terrestre. *Revista Politécnica*, 44(2), 43-50. doi:<https://doi.org/10.33333/rp.vol44n2.05>.
- Moreira, J., & Tobar, D. (2022). Desarrollo de una herramienta inteligente centrada en visión artificial y redes neuronales para el reconocimiento de maleza en plantaciones de arroz, usando lenguaje de programación Phyton. Guayaquil, Ecuador. Recuperado el 11 de 07 de 2022, de <http://repositorio.ug.edu.ec/handle/redug/59849>
- Osorio, K., Puerto, A., Pedraza, C., Jamaica, D., & Rodríguez, L. (2020). A Deep Learning Approach for Weed Detection in Lettuce Crops Using Multispectral Images. *AgriEngineering*, 2(3), 471-488. doi: <https://doi.org/10.3390/agriengineering2030032>
- Parreira, M., Peñaherrera-Colina, L., Alves, P., & Pereira, F. (2013). Interferencia de malezas en el cultivo de frijol en dos sistemas de labranzas. *Scielo*, 31(2), 319-327. doi:<https://doi.org/10.1590/S0100-83582013000200009>
- Poljak, J. (06 de 2013). Defining of printing plate quality level through introduction of different image analysis. Madrid, España. Recuperado el 29 de 09 de 2022, de https://eprints.ucm.es/id/eprint/22653/1/final_MsThesismodificado.pdf
- Pruna, E., Sasig, E., & Mullo, S. (17 de 09 de 2017). PI and PID Controller Tuning Tool Based on the Lambda Method. Quito, Ecuador: IEEE. doi:<https://doi.org/10.1109/CHILECON.2017.8229616>
- Quintero , A., Sandoval , A., Rincon, A., & Tarazona, B. (2018). Vinculación de la teleoperación, las aplicaciones móviles y la adquisición de datos en el desarrollo de un prototipo de vehículo móvil para la captura y almacenamiento de variables atmosféricas. Santander, Colombia. Recuperado el 23 de 08 de 2022, de <http://repositorio.uts.edu.co:8080/xmlui/bitstream/handle/123456789/1248/1.%20Teleoperacion%20Y%20Aplicaciones%20Moviles.pdf?sequence=1>

- Ravi, K., & Kopulwar, S. (2016). Low cost ambient monitoring using ESP8266. 779-782. doi:10.1109/IC3I.2016.7918788
- Russo, C., Ramón, H., Seferino, S., Cicerchia, M., Sabore, A., Balmer, E., . . . y Faroppa, M. (2017). Visión artificial aplicada en Agricultura de Precisión. Buenos aires.
- Sánchez, J., & Figueroa, M. (2022). Diseño e implementación de un prototipo para el control y monitoreo del consumo energético de una red WiFi alimentada por paneles fotovoltaicos en la Escuela de Educación Básica Particular Gabriel Olmedo Arroba Espinoza en la ciudad de Guayaquil. Guayaquil, Ecuador. Recuperado el 27 de 07 de 2022, de <http://repositorio.ug.edu.ec/bitstream/redug/59832/1/B-CINT-PTG-N.809%20S%c3%a1nchez%20PI%c3%baas%20Jos%c3%a9%20Roy%20%20.%20%20%20Figueroa%20%20S%c3%a1nchez%20Miguel%20%c3%81ngel.pdf>
- Sogamoso González, D. S., Aponte Rodríguez, J. A., & Solaque Guzmán, L. E. (22 de Octubre de 2015). *Universidad EIA*. Obtenido de Integración de la mecatrónica al desarrollo de la agricultura de precisión aplicada al control mecánico de maleza: <https://revistapostgrado.eia.edu.co/index.php/mem/article/view/842/760>
- Suárez, A., & Pérez, W. (2020). MODULO IP BASADO EN FPGA PARA LA DECODIFICACIÓN DE ENCODERS DE CUADRATURA. *Revista Politécnica*, 16(32), 68-76. doi:<https://doi.org/10.33571/rpolitec.v16n32a6>
- Theodoropoulos, M., Lardizabal, R., & Arias, S. (2009). Manual de producción: Producción de lechuga. Lima. Recuperado el 23 de 07 de 2022, de http://bvirtual.infoagro.hn/xmlui/bitstream/handle/123456789/72/EDA_manual_Produccion_Lechuga_02_09.pdf?sequence=1.
- Tzafestas, S. (2014). Introduction to Mobile Robot Control. Atenas, Grecia: Elsevier Inc. doi:<https://doi.org/10.1016/C2013-0-01365-5>
- Vásquez, J. (2020). Diseño conceptual de un robot móvil enfocado a la agricultura de precisión en campo abierto para la eliminación de malezas en el cultivo de palma de aceite. Recuperado el 16 de 06 de 2022, de <https://repositorio.utp.edu.co/server/api/core/bitstreams/d323a9c3-7c61-404c-8e7e-ae9372af8be1/content>
- Vidal, R., Rainero, H., Kalsing, A., & Trezzi, M. (2010). Prospección de las combinaciones de herbicidas para prevenir malezas tolerantes y

resistentes al glifosato. *Scielo*, 28(1).
doi:<https://doi.org/10.1590/S0100-83582010000100019>

- Viola, P., & Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple. Features. *IEEE International Conference on Computer Vision and Pattern Recognition.*, 1, 511–518.
- Voug, S. (Mayo de 2020). Using Color Thresholding and Contouring to Understand Coral Reef Biodiversity. San José, California, EEUU: San José State University. doi:<https://doi.org/10.31979/etd.tdbq-k4u6>
- Vougioukas, S. (2019). Agricultural robotics. *Annual Review of Control. Redalyc*, 2(1), 365-392. doi:<https://doi.org/10.1146/annurev-control-053018-0236170>.

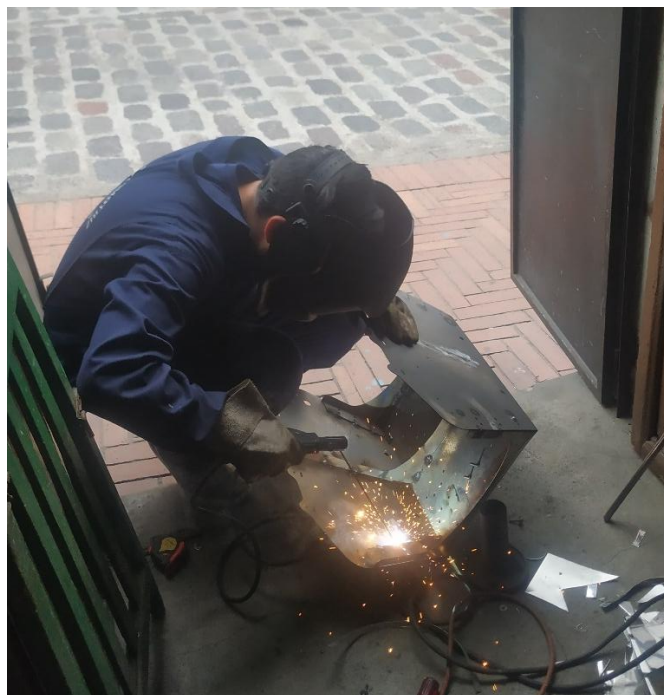
ANEXOS

ANEXO 1.

PROCESO DEL CORTE LASER DEL CHASIS EN LÁMINA DE ACERO A36 1.4 mm

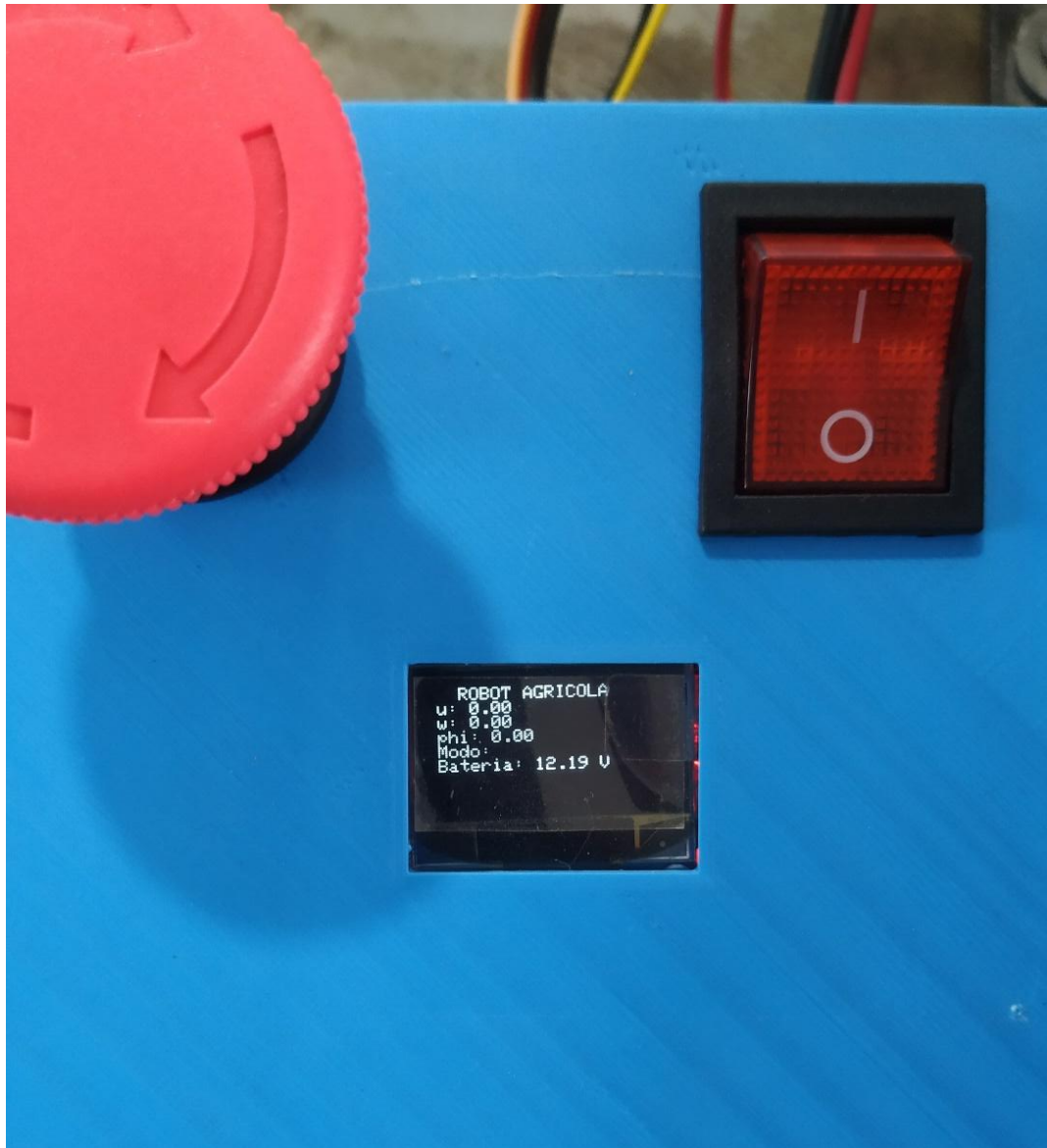


ANEXO 2.
PROCESO DE SOLDADURA DEL CHASIS



ANEXO 3.

DETALLE DE LA PANTALLA OLED



ANEXO 4. PRUEBAS DE CAMPO



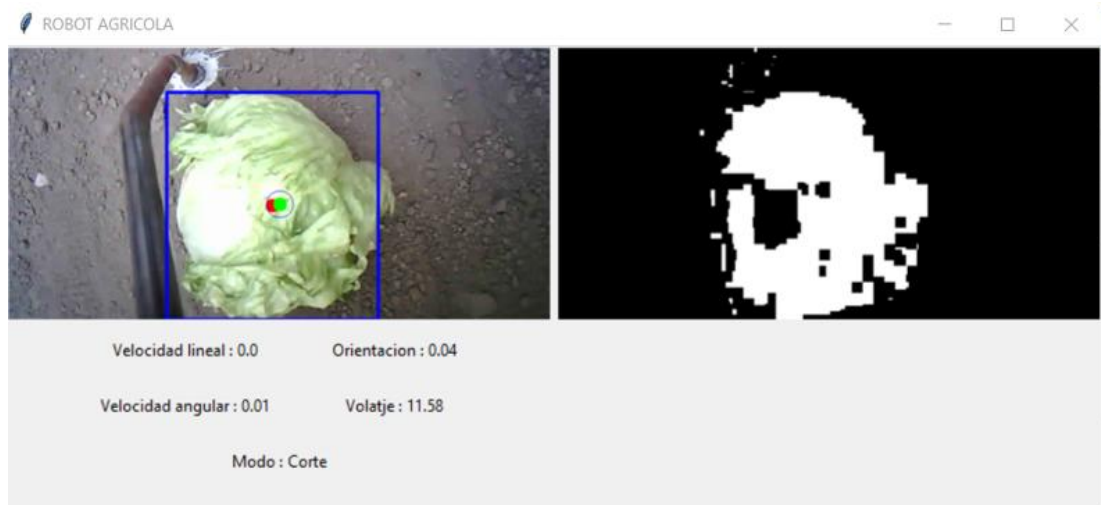
ANEXO 5. FUNCIONAMIENTO MODO BUSQUEDA



ANEXO 6. FUNCIONAMIENTO MODO DETECTOR



ANEXO 7. FUNCIONAMIENTO MODO CORTE



ANEXO 8. FUNCIONAMIENTO MODO GIRO IZQUIERDA



ANEXO 9.

FUNCIONAMIENTO MODO GIRO DERECHA



ANEXO 10.

CÓDIGO QR GIRO IZQUIERDA



ANEXO 11.
CÓDIGO QR GIRO DERECHA



ANEXO 12.
PLANOS DEL ROBOT MOVÍL

1

2

3

4

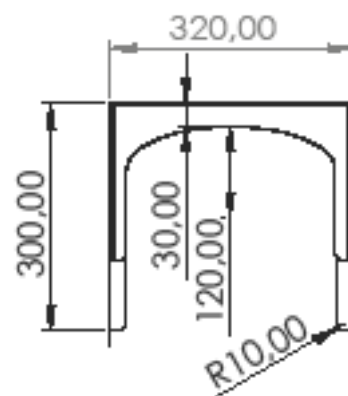
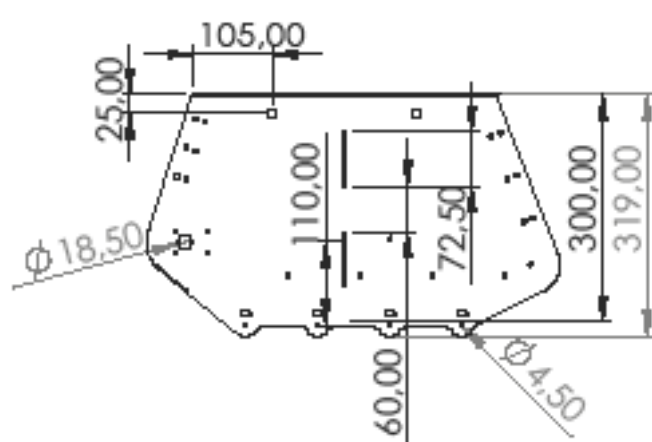
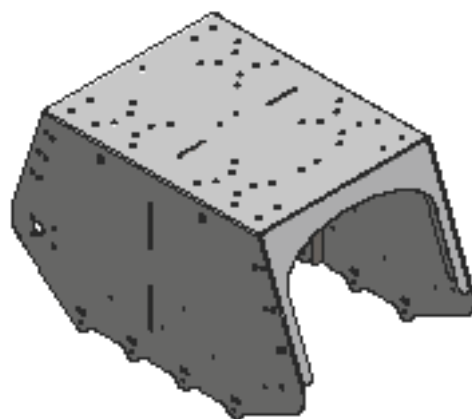
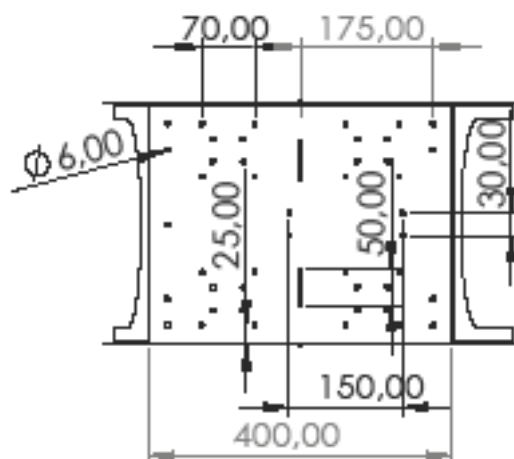
A


B

C

D

E



				TOLERANCIA:	PESO (kg)	MATERIAL:	
				±0.1		Acero ASTM A36	
				Fecha	Nombre	Chasis	
				Dibujado 30/08/2022	Nelson Suárez		
				Revisado			
				Aprobado			
				FIRMA / EMPRESA:		ESCALA:	
						1:10	
						HOJA:	
Estado	Modificación	Fecha	Nombre			A4	

1

2

3

4

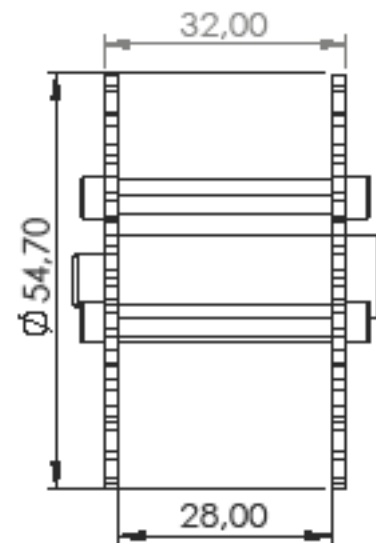
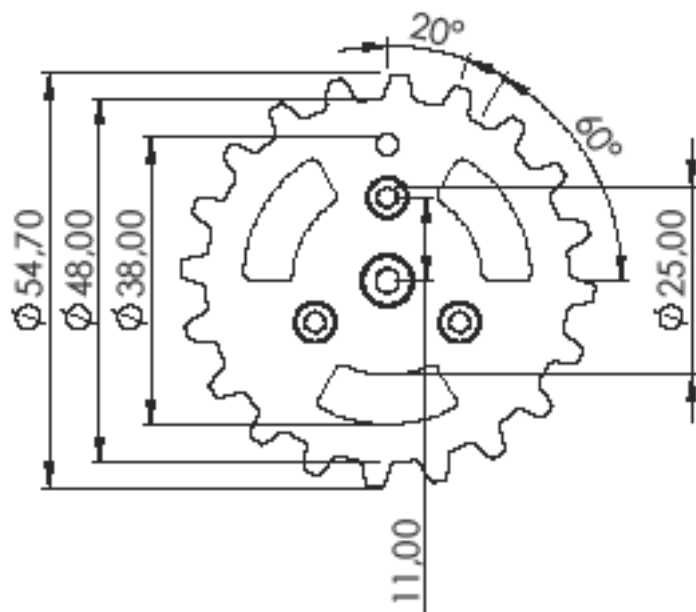
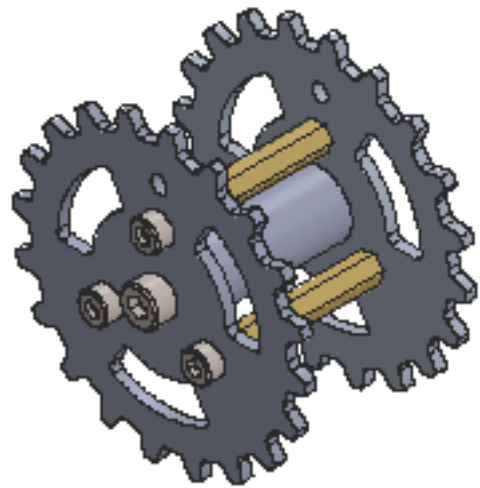
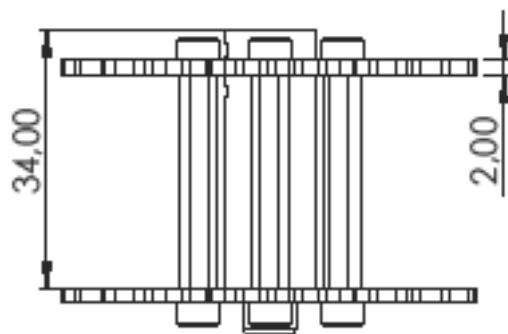
A

B

C

D

E



TOLERANCIA:

PESO (kg)

MATERIAL:

 ± 0.1 *Aluminio 7075*

Fecha

Nombre

Dibujado 30/08/2022

Nelson Suárez

Revisado

Aprobado

Rueda conductora

ESCALA:

1:1

FIRMA / EMPRESA:

DBLUO N°:

HOJA:

A4

Edición Modificación Fecha Nombre



1

2

3

4

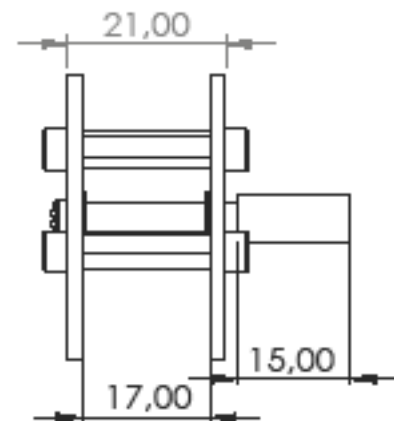
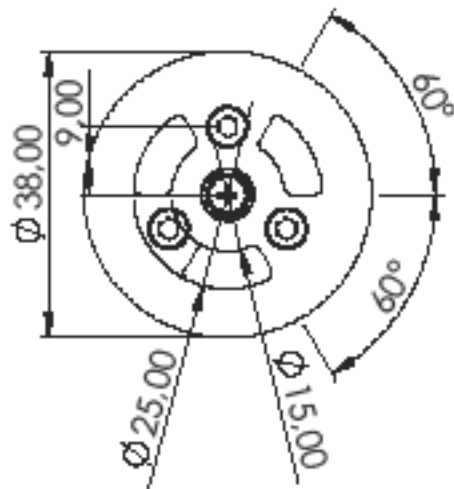
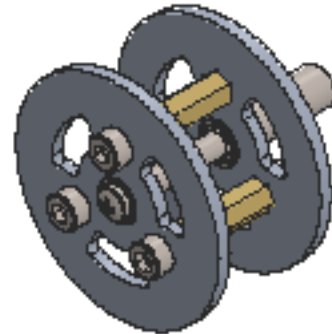
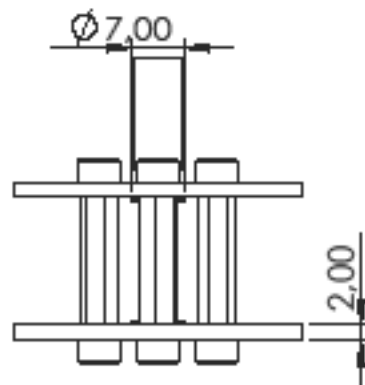
A

B

C

D

E



TOLERANCIA:

PESO (kg):

MATERIAL:

 ± 0.1 *Aluminio 7075*

Fecha

Nombre

Dibujado 30/08/2022

Nelson Suárez

Revisado

Aprobado

Rueda guía

ESCALA:

1:1

FIRMA / EMPRESA:

DIBUJO Nº:

HOJA:

A4

Edición Modificación Fecha Nombre



1

2

3

4

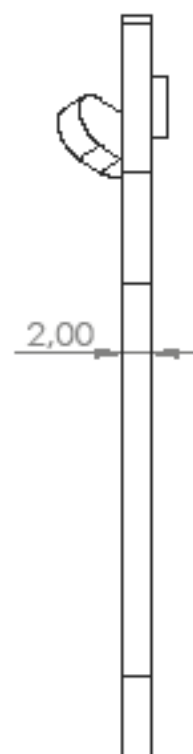
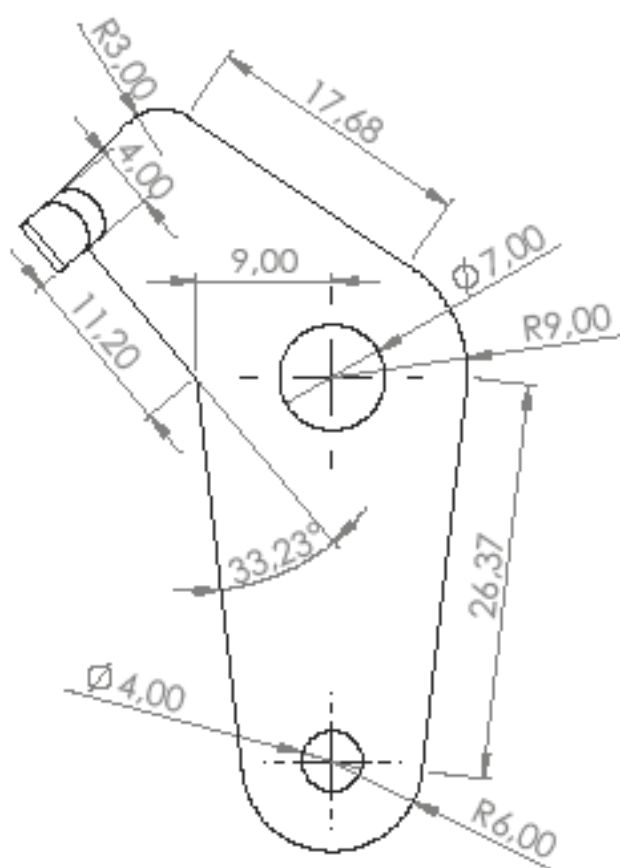
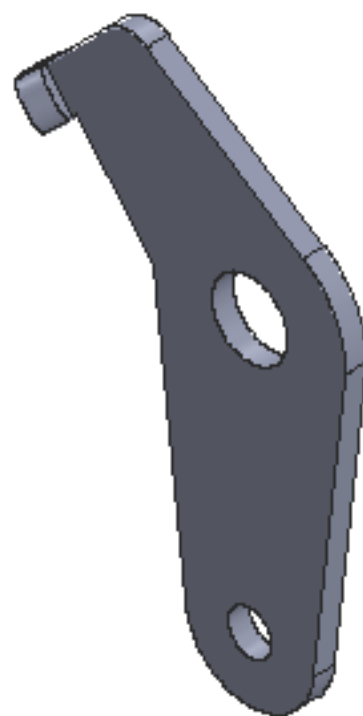
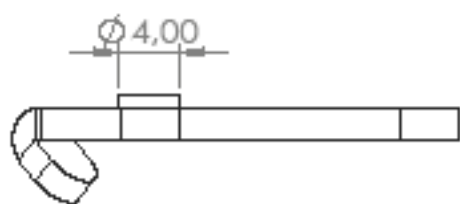
A

B

C

D

E



TOLERANCIA:

PESO (kg)

MATERIAL:

 ± 0.1

Aluminio 7075

Fecha

Nombre

Dibujado 30/08/2022

Nelson Suárez

Revisado

Aprobado

Balancín de suspensión

ESCALA:

2:1

FIRMA / EMPRESA:

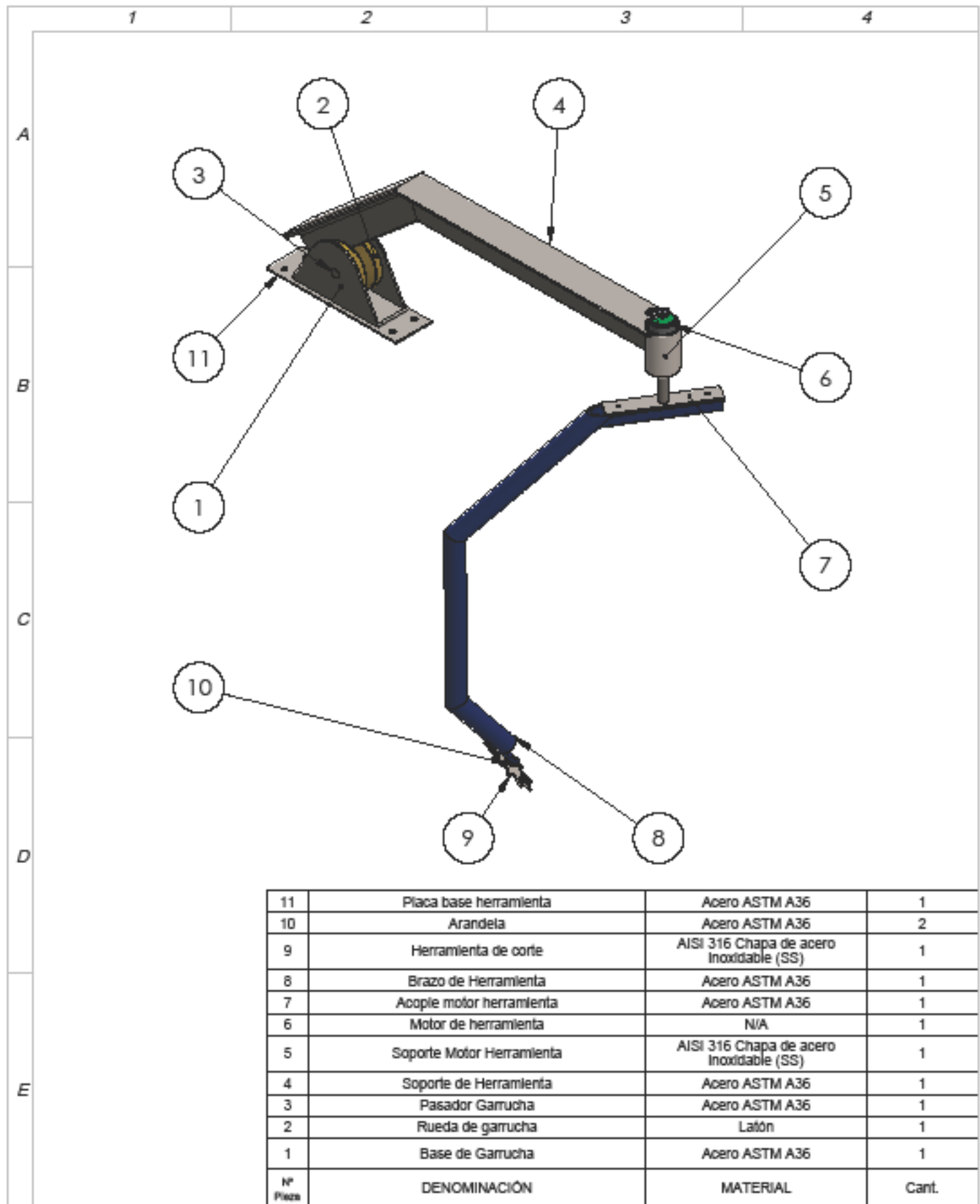
DIBUJO Nº:

HOJA:

A4

Edición Modificación Fecha Nombre





11	Placa base herramienta	Acero ASTM A36	1
10	Arandela	Acero ASTM A36	2
9	Herramienta de corte	AISI 316 Chapa de acero Inoxidable (SS)	1
8	Brazo de Herramienta	Acero ASTM A36	1
7	Acople motor herramienta	Acero ASTM A36	1
6	Motor de herramienta	N/A	1
5	Soporte Motor Herramienta	AISI 316 Chapa de acero Inoxidable (SS)	1
4	Soporte de Herramienta	Acero ASTM A36	1
3	Pasador Garrucha	Acero ASTM A36	1
2	Rueda de garrucha	Latón	1
1	Base de Garrucha	Acero ASTM A36	1
Nº Piezas	DENOMINACIÓN	MATERIAL	Cant.

TOLERANCIA:		PESO (kg):		MATERIAL:	
±0.1					
		Fecha		Nombre	
		Dibujado 30/08/2022		Nelson Suárez	
		Revisado			
		Aprobado			
FIRMA / EMPRESA:		DIBUJO Nº:		ESCALA:	
				1:10	
				HOJA:	
				A4	
Edición	Modificación	Fecha	Nombre		



1

2

3

4

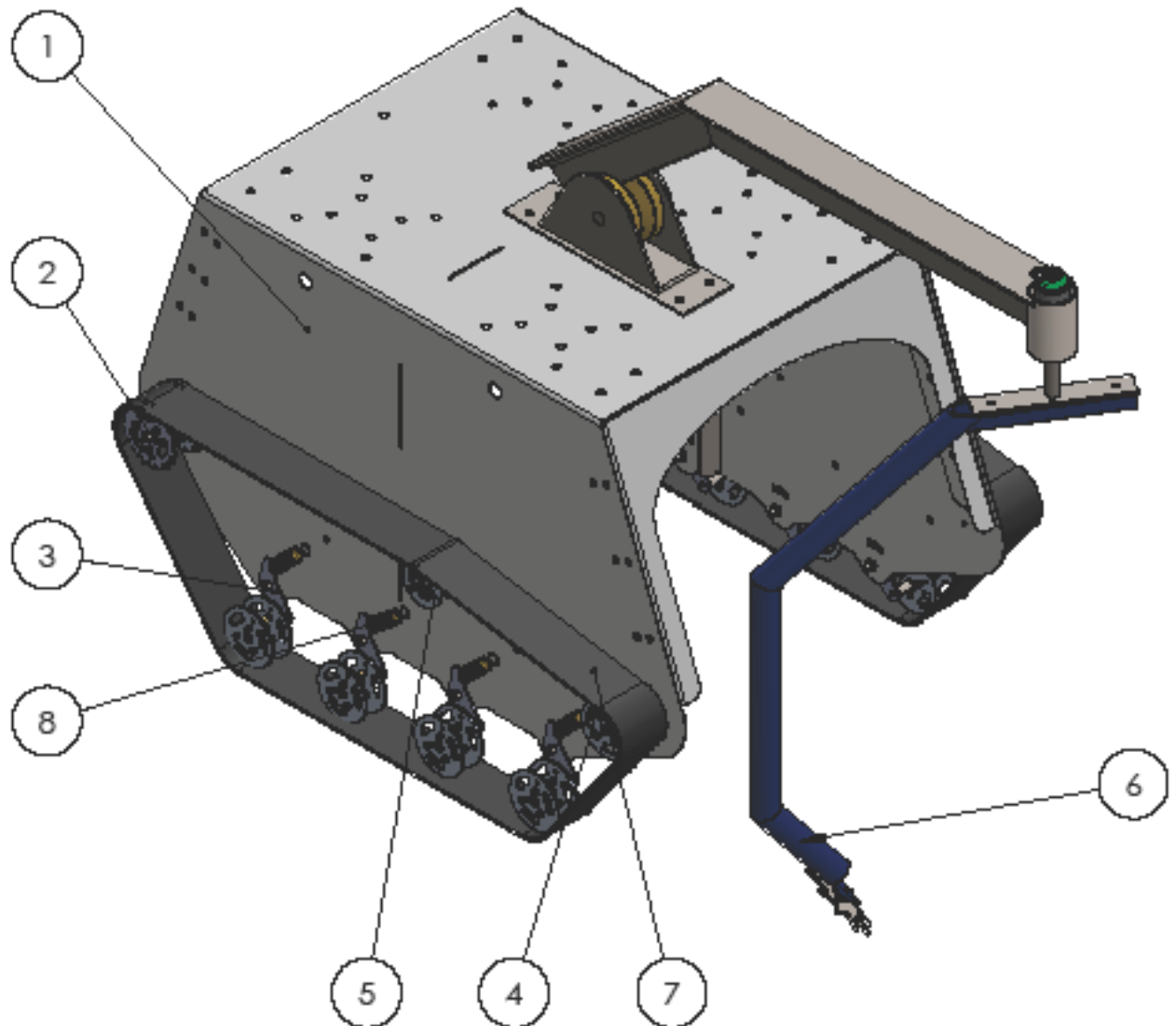
A

B

C

D

E



8	Resorte	Latón	8
7	Oruga o cadena de tracción	PLA	2
6	Herramienta	Acero ASTM A36	1
5	Rueda guía	Aluminio 7075	2
4	Balancín de suspensión	Aluminio 7075	8
3	Rueda conducida	Aluminio 7075	10
2	Rueda conductora	Aluminio 7075	2
1	Chasis	Acero ASTM A36	1
N° Pieza	DENOMINACIÓN	MATERIAL	Cant.

TOLERANCIA:

PESO (Kg)

MATERIAL:

 ± 0.1

Fecha

Nombre

Dibujado 30/08/2022 Nelson Suárez

Revisado

Aprobado

FIRMA / EMPRESA:

DBLUO N°:

ESCALA:

Ensamblaje Robot

1:20

HOJA:

A4

Edición Modificación Fecha Nombre

