

UNIVERSIDAD TÉCNICA DEL NORTE.



Facultad de Ingeniería en Ciencias Aplicadas.

Carrera de Ingeniería en Mantenimiento Eléctrico.

**IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL INTEGRADO A UN
PROTOTIPO DE VEHÍCULO ELÉCTRICO PARA MOVILIDAD PERSONAL**

Trabajo de grado presentado ante la Universidad Técnica del Norte previo a la
obtención del título de Ingeniero en Mantenimiento Eléctrico.

Autor:

Cabezas Perugachi Jesús Ricardo

Director:

Ing. Eliana Carolina Ormeño Mejía, MSc.

Asesores:

Ing. Francisco Roberto Naranjo Cobo, MSc.

Ing. Jhonny Javier Barzola Iza, MSc.

Ibarra, 2021



UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN
A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIFICACIÓN	1004306716		
APELLIDOS Y NOMBRES:	Cabezas Perugachi Jesús Ricardo		
DIRECCIÓN:	Cotacachi, Vía Imantag- Comunidad Perafán		
EMAIL:	jrcabezasp@utn.edu.ec		
TELÉFONO FIJO:	-	TELÉFONO MOVIL:	0960579717

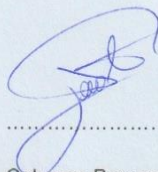
DATOS DE OBRA	
TÍTULO	IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL INTEGRADO A UN PROTOTIPO DE VEHÍCULO ELÉCTRICO PARA MOVILIDAD PERSONAL
AUTOR(ES)	Cabezas Perugachi Jesús Ricardo
FECHA:	07/10/2021
SOLO PARA TRABAJO DE GRADO	
PROGRAMA	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TITULO POR EL QUE OPTA	Ingeniero en Mantenimiento Eléctrico
ASESOR/DIRECTOR	Msc. Eliana Carolina Ormeño Mejía

2. CONSTANCIA.

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por tanto, la obra es original y que es el titular los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 07 días del mes de octubre de 2021

EL AUTOR:



.....
Cabezas Perugachi Jesús Ricardo

100430671-6



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN DEL DIRECTOR DE TRABAJO DE GRADO

Yo, Ormeño Mejía Eliana Carolina en calidad de tutor del señor, Cabezas Perugachi Jesús Ricardo certifico que ha cumplido con las normas establecidas en la elaboración del trabajo de investigación titulado **“IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL INTEGRADO A UN PROTOTIPO DE VEHÍCULO ELÉCTRICO PARA MOVILIDAD PERSONAL”**. Para la obtención del título de Ingeniero en Mantenimiento Eléctrico; aprobando la defensa, impresión y empastado.

.....
Ing. Eliana Carolina Ormeño Mejía MSc.

DIRECTOR DE TRABAJO DE GRADO

AGRADECIMIENTOS

Agradezco a Dios, por permitirme culminar esta etapa de mi vida, por cada una de las lecciones aprendidas, por los momentos de alegres y difíciles que me han demostrado que siempre se puede salir adelante.

A mis padres que siempre me han brindado su apoyo incondicional y desinteresado, por el cariño y la confianza que han depositado en mí y por todos esos valores que me han enseñado en el transcurso de mi vida.

A mis hermanos, por darme aliento y seguridad ante las adversidades y por ser ejemplo de superación y constancia.

A la MSc. Eliana Ormeño, tutora del trabajo de grado, por el asesoramiento, sugerencias, recomendaciones realizadas para la realización de este trabajo.

A los amigos y personas que aportaron de manera directa o indirecta a la realización de este trabajo.

Jesús Ricardo Cabezas Perugachi

TABLA DE CONTENIDOS

INTRODUCCIÓN	XV
A.1. Contextualización.....	XV
A.2. Planteamiento del problema.....	XVI
A.3. Justificación	XVII
A.4. Alcance del trabajo.....	XVII
A.5. Objetivo General	XVIII
A.6. Objetivos Específicos	XVIII
A.7. Viabilidad del trabajo.....	XVIII
CAPÍTULO 1	1
1.1. Introducción.....	1
1.2. Definición de vehículo eléctrico	2
1.3. Características principales de un EV	2
1.4. Sistemas de control.....	3
1.1.1. Componentes de un sistema de control	3
1.1.2. Controlador	4
1.1.3. Control de motores.....	5
1.1.4. Control de velocidad.....	9
1.1.5. Sistema de monitoreo y control	10
1.1.6. Sistema HMI.....	11
1.5. Identificación de sistemas	13
1.6. Control PID para regular la velocidad de los motores	16
1.7. Control de velocidad avanzados para accionamiento de BLDC.....	17
1.7.1. Control difuso	17
1.7.2. Control de red neuronal.....	19
1.7.3. Grey control.....	21
1.8. Regulaciones para el uso de scooter eléctricos.....	21
CAPÍTULO 2	25
2.1. Introducción.....	25

2.2.	Descripción de softwares	25
2.3.	Metodología	26
2.4.	Formulación de parámetros matemáticos	27
2.4.1.	Determinación del regulador de voltaje.....	27
2.4.2.	Caída de tensión debido a DC-DC Buck.....	27
2.4.3.	Sensor de voltaje.....	28
2.4.4.	Sensor de velocidad.....	29
2.4.5.	Distancia lineal recorrida	29
2.4.6.	Filtro paso bajo y alto exponencial EMA.....	29
2.4.7.	Modelo matemático del controlador PID.....	29
2.4.8.	Función de transferencia mediante el análisis de la curva de reacción.....	30
2.5.	Descripción de equipos y dispositivos electrónicos.....	32
2.5.1.	Elección de componentes.....	32
2.5.2.	Controlador del motor Brushless	33
2.5.3.	Sensor de voltaje.....	36
2.5.4.	Sensor de acelerador y freno	37
2.5.5.	Módulo DAC MCP4725	38
2.5.6.	Interfaz de usuario.....	39
2.5.7.	Sistema de iluminación	41
2.5.8.	Plataforma de desarrollo	42
2.5.9.	Convertidor DC-DC Buck	45
2.6.	Resultados	46
2.6.1.	Diseño del algoritmo de control	46
2.6.2.	Identificación del sistema.	49
2.6.3.	Diseño del controlador.	53
2.6.4.	Diseño de la interfaz gráfica.....	55
CAPÍTULO 3		57
3.1.	Introducción.....	57
3.2.	Metodología	57

3.3. Montaje del sistema de control con los sensores y actuadores en la estructura del scooter	58
3.3.1. Montaje de la tarjeta Arduino Mega 2560 y los dispositivos de control.	59
3.3.2. Conexión del DAC MCP4725, Módulo DS3231	59
3.3.3. Pantalla touch.	59
3.4. Validación de las mediciones	66
3.4.1. Lectura de los rpm.....	66
3.4.2. Lectura de voltaje de batería.	67
3.5. Implementación del control PI	68
3.6. Pruebas de funcionamiento del sistema de control.....	69
3.7. Manual de usuario.....	71
3.8. Descripción de los costos de dispositivos electrónicos seleccionados.....	71
3.9. Discusión.....	72
CONCLUSIONES.....	73
RECOMENDACIONES	74
Referencias	75
Anexos	80

ÍNDICE DE FIGURAS

Fig.1.1. Demanda del scooter eléctricos	2
Fig.1.2. Esquema básico de un sistema de control	3
Fig.1.3. Secuencia de alimentación de las bobinas para el accionamiento del motor BLDC.8	
Fig.1.4. Modulación de ancho de pulso para control de velocidad	9
Fig.1.5. Sensor hall con transistor de colector abierto	11
Fig.1. 6 Pantallas de Nextion de varios tamaños.....	13
Fig.1.7. Esquema de identificación de sistemas.....	14
Fig.1.8. Proceso de identificación del sistema.	15
Fig.1.9. Clasificación de los métodos de identificación de sistemas	16
Fig.1.10. diagrama de un controlador PID	16
Fig.1.11. . Esquema de control difuso	18
Fig.1. 12. Modelo de una red neuronal artificial	19
Fig.2.1. Estructura de desarrollo del sistema de control.....	26
Fig.2. 2. Circuito divisor de tensión para la medición de una batería de 67V	28
Fig.2.3. Respuesta transitoria de primer orden	31
Fig.2. 4 Esquema básico del scooter eléctrico	32
Fig.2. 5. Terminales de conexión del inversor trifásico	35
Fig.2. 6. Amplificadores operacionales cuádruple	36
Fig.2. 7. Acelerador manual.	37
Fig.2. 8. Interruptor de freno.....	38
Fig.2. 9. módulo DAC MCP4725	38
Fig.2. 10 Pantalla touch Nextion de 5 pulgadas.....	40
Fig.2. 11. Luces delanteras tipo led	41
Fig.2. 12. Foco trasero de siete leds con indicador de freno	41
Fig.2. 13. Componentes del Arduino Due	44
Fig.2. 14. . Convertidor DC-DC Buck.....	46
Fig.2. 15. Diagrama de bloque.....	46
Fig.2. 16. Lectura de los valores del acelerador y suavizado de señal	47
Fig.2. 17. Lectura del rpm del motor	47
Fig.2. 18. Determinación de la variable de control mediante el algoritmo PID	48
Fig.2. 19. Respuesta en lazo abierto	49
Fig.2. 20. Identificación de sistema con los datos obtenidos de la respuesta de la plant .	51
Fig.2. 21. Estimación de modelo de proceso.....	51
Fig.2. 22. Modelo Matemático estimado.....	52

Fig.2. 23. Modelo matemático estimado.....	52
Fig.2. 24. Función de transferencia de la planta	52
Fig.2. 25.Importación de la función de transferencia.....	53
Fig.2. 26. Modificación DE LGR	54
Fig.2. 27. Parámetros del controlador.	54
Fig.2. 28. Diagrama de flujo de la utilización de la pantalla con el microcontrolador.....	55
Fig.2. 29. Interfaz HMI	56
Fig.3. 1.Distribución de los dispositivos electrónicos.	58
Fig.3. 2.Distribución de los elementos del sistema de control.....	59
Fig.3. 3.Terminales de conexión de la pantalla touch	60
Fig.3. 4.Conexión de los dispositivos electrónicos del sistema de control del scooter	66
Fig.3. 5. (a) Lecturas del Arduino, (b) Lecturas del tacómetro.	66
Fig.3. 6. (a) Lecturas del Arduino, (b)Lecturas multímetro	67
Fig.3. 7. Diagrama de bloque del sistema de control de velocidad.	68
Fig.3. 8. Respuesta del control PI.	68
Fig.3. 9. (a)Funcionamiento del HMI, (b Luces delanteras y frenos encendidos.....	69
Fig.3. 10.(a) Indicador de carga de batería baja. (b)Indicador de batería a media carga .70	
Fig.3. 11. Velocidad de referencia y velocidad alcanzada por el scooter 20 y 10 km/h	71

ÍNDICE DE TABLAS

TABLA 1.1. CARACTERÍSTICAS DE ARDUINO Y RASPBERRY.....	5
TABLA 1.2. DISPOSITIVOS HMI.....	12
TABLA 1.3. EFECTO DE LOS PARÁMETROS DEL CONTROLADOR.....	17
TABLA 1.4. APLICACIONES DEL CONTROL DIFUSO.....	19
TABLA 1.5. CONTROL DE BLDC CON APLICACIONES DE REDES NEURONALES .	20
TABLA 2. 1.PUNTUACIÓN DE CARACTERÍSTICAS Y SIGNIFICADO	33
TABLA 2. 2.COMPARACIÓN DE TIPOS DE DRIVES PARA MOTOR BRUSHLESS	34
TABLA 2. 3. COMPARACIÓN DE INTERFACES HMI.....	39
TABLA 2. 4.ENTRADAS Y SALIDAS DEL MICROCONTROLADOR	42
TABLA 2. 5.COMPARACIÓN ENTRE PLATAFORMAS Y MICROCONTROLADORES .	43
TABLA 2. 6. CORRIENTE DE FUNCIONAMIENTO DE LOS DISPOSITIVOS.	45
TABLA 2. 7. ELEMENTOS DE LA PANTALLA	56
TABLA 3. 1. CONEXIÓN DE PINES DEL ARDUINO	60
TABLA 3. 2. MEDICIONES ARDUINO Y TACOMETRO DIGITAL.....	66
TABLA 3. 3. LECTURA DE VOLTAJE.....	67
TABLA 3. 4. VALIDACIÓN DE LAS DISTANCIA RECORRIDA.....	70
TABLA 3. 5. COSTO DE COMPONENTES ELÉCTRONICOS PARA EL SISTEMA DE CONTROL.....	72

RESUMEN

Los vehículos eléctricos (VE) son un medio de transporte que se consideran como una estrategia para reducir la contaminación ambiental, con un futuro para transporte masivo de personas, además de reducir la dependencia de combustibles fósiles y sus derivados. En la actualidad ya se construyen buses, motos, bicicletas, entre otros que tienen propulsión eléctrica que presenta al mercado mundial, variedad de opciones para la movilidad individual o colectiva.

Este trabajo trata sobre la implementación de un sistema de control y una interfaz de usuario para un prototipo de scooter eléctrico, mediante la investigación se determinó los elementos constitutivos y se seleccionó los componentes. Para luego determinar el tipo de control adecuado para el vehículo. Las regulaciones internacionales exigen que los vehículos de este tipo cuenten con luces, bocina, frenos; aspectos tomados en cuenta para la realización del trabajo. El sistema se desarrolló en una plataforma de desarrollo de código abierto, con dispositivos que se pueden encontrar a nivel nacional y de bajo costo. Como resultado del trabajo implementado del sistema de control, se puede realizar: activación del sistema de iluminación, regulación de velocidad y se muestra el nivel de carga del batería. Se utilizó un control PI para la regulación de la velocidad y eliminar el error de estado estable encontrado en la etapa de identificación de la planta que permitió que vehículo tenga una aceleración suave con un tiempo de establecimiento de 4 segundos, ofreciendo comodidad y seguridad al usuario.

Palabras claves: Scooter eléctrico, Sistema de control, Control PI, Interfaz gráfica.

ABSTRACT

Electric vehicles (EV) are a new kind of transport that are considered as a strategy to reduce environmental pollution, with a future for mass transportation of people, in addition to reducing dependence on fossil fuels. At present, buses, motorcycles, bicycles, among others that have electric propulsion, are already being built that presents to the world market, a variety of options for individual or collective mobility.

This work deals with the implementation of a control system and an user interface for an electric scooter prototype. Through the research carried out, the components were determined to then make the selection and determine the type of control suitable for the vehicle. The system was developed on an open source development platform, with devices that can be found in the country and cheap, because of the implemented work of the control system, it can be performed: activation of the lighting system, speed regulation and the battery charge level is displayed. A PI controller was used to regulate the speed of the vehicle that allows a smooth acceleration of the scooter, offering comfort and safety to the user.

Keywords: Electric Scooter, Control System, PI Control, Graphic Interface.

INTRODUCCIÓN

A.1. Contextualización

Los vehículos eléctricos (VE) son un medio de transporte que se consideran como una estrategia para reducir la contaminación ambiental, con un futuro para transporte masivo de personas, además de reducir la dependencia de combustibles fósiles y sus derivados. (Liu, Chau, Wu, & Gao, 2013). En la actualidad ya se construyen buses, motos, bicicletas, entre otros que tienen propulsión eléctrica que presenta al mercado mundial, variedad de opciones para la movilidad individual o colectiva. (Caicedo, y otros, 2014)

Los vehículos eléctricos no solamente tienen ventajas ambientales, sino que también son superiores en aspectos como: respuesta de torque, corto proceso de aceleración, control directo de la velocidad, relativa facilidad de implantación. De ahí la importancia de su desarrollo e investigación. Los principales aspectos que considerar de un VE son el motor y la tecnología de control, esto contribuirá al rendimiento del vehículo, además, deberían tener características de amplio rango de velocidad ajustable, altas velocidades y par de arranque, ligero, tamaño compacto, eficiencia del frenado regenerativo. (Xia, 2012)

Actualmente, el desarrollo de nuevas tecnologías VE se centran en sistemas individuales, como: maquinas eléctricas, sistemas de accionamiento, pilas de combustible, baterías, etc. Se presenta avances en el diseño, análisis en control de motores. (Liu, Chau, Wu, & Gao, 2013). Los requisitos de las maquinas eléctrica para VE son más estrictas y exigentes que las de la industria (Chau, 2015) . En la actualidad existen varios sistemas de control para los vehículos eléctricos como: control vectorial, control difuso en control de vehículos eléctricos, la aplicación de control de estructura variable, control difuso, red neuronal de control, sistema experto, algoritmo genético y otras tecnologías de control inteligente no lineales que tiene como finalidad el mejoramiento de la autonomía y rendimiento. (Xia, 2012)

El control vectorial apareció hace 30 años atrás y con el pasar del tiempo, han logrado grandes avances, que pueden aplicarse en varias aplicaciones. Fodorean,(2012) afirma:

Varios tipos de unidades controladas por vectores que permiten rendimiento dinámico de los variadores de CA para que coincida o a veces incluso para superar el de la unidad de CC. En la actualidad, la tendencia principal es utilizar unidades de vectores sin sensores, donde la información de velocidad y posición es obtenida por monitoreo de voltajes o corrientes de entrada. (p.1179)

Los PID tradicionales, no permiten la carga dinámica en vehículos eléctricos que utilizan motor DC sin escobillas, si se mantiene este tipo de controladores, la respuesta del sistema al estado estable será muy larga lo que provocará que el motor tenga un mal rendimiento.

Los controladores PID-fuzzy resuelven el problema, el sistema debe mantener la condición de estado estable de la velocidad del motor, para mejorar el rendimiento del motor. (Padhy, Khadanga, & Panda, 2018)

Los sistemas de control tienen un impacto directo sobre el rendimiento, confiabilidad y estabilidad del VE, debido a la energía limitada que los vehículos es prescindible tener un control adecuado, es conveniente que puedan ser desarrollados en corto tiempo con un mantenimiento relativamente fácil y puedan procesar datos y permitan el control en tiempo real. (Chenguang, Zhiqiang, Yi, Lingbo, & Ling, 2018). Las plataformas open source se presentan como una alternativa para la implementación de controles digitales, además se presentan varios proyectos en donde se muestra el uso de bibliotecas de PID, que disminuyen el costos de implementación . (Aguilar & V., 2013). La mayoría del software y hardware utilizados especializado en adquisición y envió de datos entre dispositivos es patentado, por ello se crean plataformas integradas para el control de velocidad y par de los motores eléctricos en código abierto. (Asprillal, Matinez, Mufioz, & Cortes, 2017)

A nivel nacional mediante la resolución No. 016-2019, redujo al 0% el arancel para importación de vehículos eléctricos para uso personal, transporte y cargadores para electrolineras, baterías. Buscando así, reducir el impacto ambiental, además de promover la construcción de sistemas de movilidad sustentable con modernos medios de transporte. (Ministerio de Producción Comercio Exterior Inversiones y Pesca, 2019)

A.2. Planteamiento del problema.

En la mayoría de los vehículos eléctricos la velocidad, torque, autonomía y demás parámetros son regulados por los sistemas de control, que están constituidos de equipos electrónicos y eléctricos los cuales deben estar configurados de acuerdo con los parámetros deseados para realizar la función de movilidad personal. Los sistemas patentados resultan tener costos más altos que los de libre distribución, por tanto, no es accesible para todo el público. Las plataformas open source disminuyen los precios de implementación para así, brindar un correcto funcionamiento y confiabilidad al usuario.

El problema

¿Cómo implementar un sistema de control adecuado para el scooter y los equipos necesarios para su funcionamiento, basado en aplicaciones open source?

A.3. Justificación

Los vehículos eléctricos son una alternativa innovadora y de tecnología en crecimiento para la movilidad de las personas, además del gran interés sobre ella, debido a los problemas ambientales causadas por el consumo de combustibles. Sin embargo, estos vehículos requieren de control de velocidad, torque, indicador de batería entre otros, para aumentar su autonomía para llegar a ser un transporte eficiente. Igualmente, el impacto ambiental causado es menor que la de los motores a combustión.

En uno de los ejes del plan nacional de desarrollo 2017-2019 toda una vida en su objetivo 3 dice que se debe garantizar los derechos de la naturaleza para actuales y futuras generaciones que busca reducir la dependencia de energía fósil para evitar emisiones que contribuyen al cambio climático, debido a esto en el plan se contempla en apoyar y promover las buenas prácticas que reduzcan la contaminación y aportan a la conservación del medio ambiente. Además, beneficia a las personas que puedan adquirir un vehículo eléctrico para su movilidad, reduciendo sus gastos en transporte.

A.4. Alcance del trabajo.

El proyecto busca la implementación de un sistema de control integrado basado en aplicaciones open source a un prototipo de vehículo eléctrico para movilidad personal que permita mejorar la autonomía y permita controlar la velocidad, además que indique los parámetros y estado del vehículo, que brinde comodidad y seguridad al usuario.

El diseño del control y programación se realizará en un microcontrolador y software libre con los diferentes elementos necesarios para la implementación para controlar la velocidad del vehículo eléctrico.

El sistema HMI (Interfaz Hombre Maquina) permitirá mostrar el estado del vehículo como: estado de la batería, velocidad, distancia recorrida, indicador de carga. Por tanto, el alcance del trabajo es práctico en el área de control de vehículos eléctricos.

A.5. Objetivo General

Implementar un sistema de control integrado mediante una aplicación open source a un prototipo de vehículo eléctrico para movilidad personal.

A.6. Objetivos Específicos

- Realizar un análisis de los dispositivos de control en vehículos eléctricos de movilidad personal para monitorización de parámetros del sistema.
- Diseñar un sistema de control integrado de velocidad para un prototipo de vehículo eléctrico de movilidad personal (scooter).
- Evaluar el funcionamiento del sistema de control integrado de velocidad para un vehículo eléctrico de personal basado en dispositivos open source.

A.7. Viabilidad del trabajo.

Para la realización del proyecto, los materiales a utilizarse se encuentran disponibles a nivel nacional que no representan un costo alto para su realización, la información necesaria se encuentra en revistas de investigación, portales de internet, libros, artículos científicos, por ser un tema de interés mundial. Además, las investigaciones realizadas acerca de los vehículos eléctricos permiten la implementación del proyecto.

CAPÍTULO 1

Revisión bibliográfica

En el presente capítulo se realiza un análisis bibliográfico de los sistemas de control orientado a los vehículos eléctricos, comparando los distintos métodos y equipos para su posterior selección en base a los requerimientos y alcance del proyecto.

1.1. Introducción

Los vehículos eléctricos (EV) se consideran en el sector de transporte como una tecnología prometedora para reducir el consumo de energía procedente de combustibles fósiles, evitando las emisiones de gases de efecto invernadero, además reducción la contaminación del aire de la localidad. (Daghero, 2019).

Para considerar que un EV tienen un impacto menor en el medio ambiente, se deben cumplir algunos criterios para afirmar que su contaminación es nula, como, por ejemplo: la forma de generación de energía para el uso de los vehículos debe provenir de centrales de energía renovable, también se verifica el uso de batería, si genera algún tipo de contaminación tanto como en la fabricación, uso y fin de vida útil. (Herrera P. , 2017)

Las ventas a nivel mundial de los vehículos eléctricos van en crecimiento exponencial a lo largo de los años, recalando en América latina su crecimiento es sumamente menor al resto de regiones. Sin embargo, la existen con alrededor de 28 modelos distintos de EV, con ventas cercanas a las 517 mil unidades hasta 2016 a nivel mundial. (Nava, 2017).

En lo referente a los vehículos de movilidad personal (VMP) se nota cada vez más, debido al gran crecimiento y aumento de popularidad de los scooters eléctricos que, además, representan un 60% de todos los VMP, según los datos de una tienda en línea (idealo.es). (Otero, 2019).

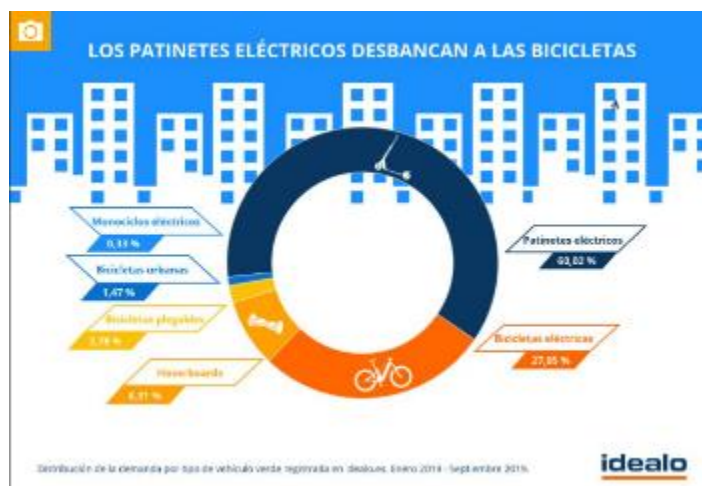


Fig.1.1. Demanda del scooter eléctricos
Fuente: (Otero, 2019).

En la Figura 1.1., se muestra el porcentaje de demanda de los scooters eléctricos en comparación de los demás VMP, siendo mayor incluso que las bicicletas, el incremento se dio en tan solo un año.

1.2. Definición de vehículo eléctrico

Un vehículo eléctrico utiliza la energía almacenada en las baterías, para activar los motores que funcionan por medio de energía eléctrica generar movimiento. Posee una mínima cantidad de componentes para su funcionamiento en comparación con los vehículos con motores a combustión. (García Tobar, 2018).

Generalmente, los VPM se conforman de dos ruedas y la transmisión, el motor comúnmente se encuentra en el interior del asiento del piloto. Debido a su costo, versatilidad, son preferidos al momento de transportarse en la zona urbana.

1.3. Características principales de un EV

Los requerimientos de los usuarios que utilizan los EV solicitan suficiente torque y velocidad en especial en lugares de topología variable y vías inestables, capaces de recorrer grandes distancias, por tal motivo, se requiere convertidores altamente eficientes y compactos, sistemas de almacenamiento de energía, su mantenimiento e implementación debe ser relativamente simple. (Asprilla, Martinez, Mufioz, & Cortes, 2017).

Las unidades de control de motores deben convertir la energía eléctrica en mecánica, que tienen como función principal el accionamiento de motores, los requisitos de las máquinas para los vehículos eléctricos son mayores que los de la industria. (Chau, 2015) afirma: los principales requerimientos:

- Alta densidad de torque y alta densidad de potencia.
- Amplio rango de velocidad, que cubre el arrastre de baja velocidad y el crucero de alta velocidad.
- Alta eficiencia en amplios rangos de par y velocidad
- Amplia capacidad de operación de potencia constante
- Gran capacidad de par para lanzamiento eléctrico y escalada
- Alta capacidad de sobrecarga intermitente para adelantar
- Alta fiabilidad y robustez para el entorno vehicular.
- Bajo ruido acústico
- Costo razonable

1.4. Sistemas de control

Los sistemas de control son utilizados para gestionar los recursos del vehículo con la ventaja de control de potencia, buena eficiencia y rápida capacidad de aceleración. (Zhang, Wen, & Zeng, 2009), además deben permitir mejorar la autonomía, un control suave del vehículo, una operación de bajo ruido. (Yusong, Yijie, & Heng, 2016).

1.1.1. Componentes de un sistema de control

Un sistema de control es el conjunto de dispositivos que interactúan entre sí con el fin de controlar una variable de interés que se asemeje al valor establecido, sin importar los cambios que influyen en su comportamiento. Para realizar dichas operaciones es necesario medir el valor de la variable y tomar una acción en función de su valor y seguidamente se encargará de transmitirlos al sistema controlado a través de sus accionamientos. (Marcillo, y otros, 2017). En la Figura 1.2. se muestra los elementos comunes de un sistema de control.

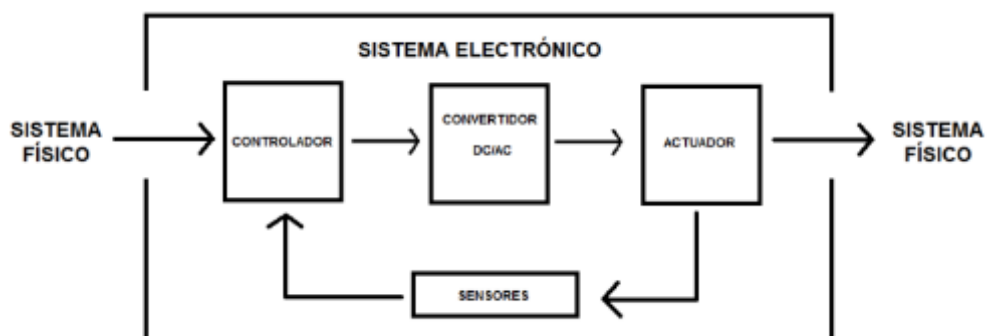


Fig.1.2. Esquema básico de un sistema de control
Fuente: (Rodríguez J. , 2015).

1.1.2. Controlador

Un controlador es el encargado de recibir la señal proveniente de las unidades de medición y la compara con otra señal de referencia o la deseada, en función de la diferencia entre ambas señales y el tipo de controlador se genera una señal que se envía al elemento final de control con el fin de contrarrestar la diferencia. (Marcillo, y otros, 2017).

Un EV posee varios sensores y circuitos que miden la energía almacenada, con el fin de tener datos precisos para mejorar la eficiencia de los vehículos. La información adquirida de los sensores es recopilada por sistemas integrados digitales, para ser procesados. La complejidad de las operaciones depende de las características del EV, cada vez es más necesario procesadores potentes para realizar operaciones complejas. (Asprilla, Martínez, Muñoz, & Cortes, 2017).

En los vehículos eléctricos el controlador se encarga del comportamiento en cualquier instante de tiempo. Una de las funciones básicas es tener el control de la velocidad, mediante una señal de baja potencia (comunmente un potenciómetro incorporado en un pedal o una empuñadura), el controlador proporciona la alimentación necesaria al motor, es decir, aísla el circuito de baja potencia con el de alta, normalmente cuentan con diferentes parámetros para el control de velocidad en diversas situaciones. (Herrera P. , 2017).

Debido a que los controladores comerciales son específicos para componentes eléctricos y mecánicos de un producto determinado, que en consecuencia en la mayoría de los casos se debe comprar todo el producto, además su garantía se pierde si no se utiliza su producto. (Cock, 2015)

El controlador del sistema puede ser desarrollado en diversas plataformas con distintos dispositivos de acuerdo a las necesidades de control y las características del microcontrolador, estos pueden desarrollarse en entornos de Arduino, Raspberry PI, LabVIEW, entre otros. Entre las plataformas de código abierto más utilizadas por su costo y su versatilidad para la realización de proyectos para pequeñas y medianas aplicaciones en diferentes campos, son: arduino y raspberry, que se describen a continuación:

- Arduino

Es una plataforma de desarrollo open source, compuesta de hardware y software para diseñar y realizar pruebas de productos, además, de prototipos electrónicos. El hardware está constituido por un microcontrolador, y con distintos componentes electrónicos, la placa puede ser programada mediante el software con varias aplicaciones. El lenguaje de programación

de Arduino (Arduino Programming Language, APL), permite que su programación sea fácil, sin la necesidad de conocimientos de algoritmos, códigos, en tanto a hardware no es necesario un profundo conocimiento de los fundamentos eléctricos y/o electrónicos para el desarrollo de hardware. (Casco, 2014).

- Raspberry Pi

Es un dispositivo de computación embebido de reducido tamaño, que puede conectarse con un monitor, TV, ratón, teclado USB convencionales. Es capaz de realizar actividades de un computador portátil o de escritorio; puede navegar en internet, reproducir videos, realizar hojas de cálculo, procesamiento de texto entre otras. (Herrera S. , 2016)

- Comparación entre Arduino y Raspberry.

Para la comparación entre ambos dispositivos se realiza la Tabla 1.1., que muestra las principales características de ambos microcontroladores.

TABLA 1.1.
CARACTERÍSTICAS DE ARDUINO Y RASPBERRY

Características generales	
Arduino	Raspberry
<ul style="list-style-type: none"> • Diseño compacto para reducir el tamaño de la circuitería. • Posee un IPS(System Programmer) que permite la transferencia de la programación dentro del microcontrolador, sin cambios en el circuito. • El modelo básico tiene un microcontrolador AVR de 8 bits con diferentes componentes para la regulación de 5V, además de conectores de salida, adaptadores de entrada. • Los puertos de entrada y salida E/S se localizan en la parte superior, con fácil acceso para su rápida conexión o a su vez para el intercambio de shield, utilizadas para extender las capacidades del microcontrolador. • Se puede interactuar fácilmente con sensores 	<ul style="list-style-type: none"> • Posee un integrado en donde se encuentre todos los componentes del sistema. • En los modelos más avanzados llevan un Broadcom BCM2835, que incluye un procesador (ARM1176JZF-S), tarjeta gráfica con aceleración 3D y video de alta calidad, 512 MB de RAM, además de una tarjeta de sonido, bus USB, tarjeta de red (conector RJ-45) con una velocidad de 10 a 100 Mbps, puerto para una memoria SD, salida de audio (Jack 3,5mm), salida HDMI. • Varios pines de entrada y salida para propósito generales (GPIO) que añaden un puerto SPI (puerto serie), un puerto I²C, salidas PWM, salidas y entradas digitales.

Fuente: (Casco, 2014) y (Monk, 2016).

Debido a las facilidades presentadas y versatilidad de la plataforma Arduino, es principalmente utilizado en prototipos de todo tipo de proyectos, debido a su compatibilidad con los diferentes sensores e información y soporte en foros, blogs, medios digitales, etc.

1.1.3. Control de motores

Los motores son los elementos principales de los vehículos eléctricos, encargados de convertir la energía eléctrica en movimiento mecánico, los requerimientos de las máquinas

eléctricas para los vehículos eléctricos son más exigentes que los requerimientos industriales. (Chau, 2015).

La clasificación de los motores eléctricos se muestra en el Anexo A, los tipos con negrita son aquellos que se aplicaron para la propulsión del VE.

El controlador del motor o driver se utiliza para transmitir la energía necesaria para generar movimiento en el motor para luego se trasmite a la rueda. Además, debe controlar parámetros como: regulación de velocidad, regulación de par. Debe tener protecciones para el motor y sus componentes, también disipador de calor. (Parión, 2019).

1.1.3.1. Control de motores DC

Para el control de velocidad del motor D.C. existen diferentes configuraciones electrónicas, que consiste en aplicar distintos niveles de tensión al motor. Antiguamente se utilizaba una resistencia variable para aplicar un nivel de tensión más baja. Debido a la energía disipada en forma de calor no es eficiente, además de no proporcionar un funcionamiento suave que resulta en molestas sacudidas, que evidentemente tienen un impacto sobre la autonomía del vehículo. (Martínez, 2013).

En general, para el control de motores D.C. se puede realizar por dos métodos: control de voltaje de armadura y por debilitamiento del flujo. Cuando se reduce la tensión de armadura se reduce a corriente del motor de D.C.; disminuye así su velocidad, en contra parte si se eleva el voltaje de armadura se incrementa su velocidad. (Chau, 2015).

Si se suministra un voltaje fijo, las máquinas de DC solo trabajaran a velocidad y torque nominales, para proporcionar un control adecuado para los VE es imprescindible el uso de convertidores DC-DC. Los convertidores DC-DC se utilizan para el control de voltaje de los motores DC, adoptan un modo de operación de corte que nos brinda una salida de voltaje continua menor a la de entrada (Carrascosa, 2015). Con aplicaciones de la electrónica de potencia los controladores modernos por medio de PWM (Pulse Width Modulation) regula el nivel de voltaje aplicado al motor con menor pérdida de energía. Morales (2018) dice:

PWM son las siglas de Pulse Width Modulation, traducido literalmente como modulación por ancho de pulso. Una señal PWM consiste en una señal con frecuencia fija, y dos valores de tensión (alto y bajo), Otro parámetro característico de una señal PWM es el denominado ciclo de trabajo (duty cycle), que hace referencia al porcentaje de tiempo que el pulso (tensión entregada) está en activo durante un ciclo. (p.35).

En la actualidad, existen controladores de motores D.C. sin la dependencia de sensores mecánicos(encoders), se basan en la estimación en tiempo real y de mediciones indirectas.

Se realiza la única medición de la corriente eléctrica del inducido, por lo cual es importante la instrumentación para realizar las mediciones correspondientes que representen el valor real. El proceso de estimación se realiza en dos etapas; la primera se utiliza el diseño de observadores de modos deslizantes para estimar el par eléctrico. En la segunda etapa el diseño de observadores por ubicación de polos. (Ramírez, y otros, 2017).

1.1.3.2. Control de motores de inducción

Las máquinas eléctricas de C.A. poseen varias ventajas en comparación que las de D.C., siendo más pequeños y más robustas con la misma potencia, además de ser más económicas. Con los avances en los semiconductores ha permitido el desarrollo de los accionamientos de velocidad variable. Sin embargo, los modelos matemáticos son más complejos que los de las maquinas D.C., por tanto, necesitan un control más complicado y convertidores más costosos para realizar el control de par y de velocidad. (García J. S., 2012)

El control de estos motores requiere de desafíos tecnológicos para poder desarrollar un suministro alterno y trifásico a partir de fuentes de DC. Se logra mediante la implementación de inversores trifásicos, además debe funcionar como variador de frecuencia para poder controlar el motor de inducción, permitiendo el control de frecuencia y amplitud de voltaje. El control de velocidad es una de las aplicaciones más utilizadas en el manejo de motores de inducción. Existen variadores de frecuencia que son más complejos orientados al control de torque mediante el control vectorial orientado al campo. Este tipo de dispositivos requieren un procesamiento de señales que permita operar y coordinar el accionamiento de los interruptores de potencia (IGBT o MOSFET) para la generación de voltaje trifásico. (Martínez, 2013).

Los inversores u onduladores son fundamentales si se utiliza motores C.A., es la encargada de suministrar energía que proviene de la batería. Además, regula la tensión de acuerdo con los requerimientos del conductor. También tiene la capacidad de cargar las baterías del vehículo debido al freno regenerativo. (Zumbado, 2015).

Los principales métodos de control de motores de inducción son: variación de voltaje y frecuencia, control orientado al campo (FOC), control de par directo (DTC).

1.1.3.3. Control de motores Brushless

Los motores que se denominan sin escobillas de corriente continua (Brushless DC o BLDC) son motores síncronos de imán permanente con campos magnéticos distribuidos uniformemente (Trapezoidal Back-Emf Synchronous Motor) en el entrehierro, también se lo

conoce conocido como motor de fuerza contraelectromotriz trapezoidal. (Guerrero, Santibañez, & Ollervides, 2016)

Normalmente están contruidos con decodificador(encoder) óptico, con medidores de corriente, sensores de efecto Hall para conmutación de corriente, son sensores de posición ubicados en el eje del rotor, que sirve para el operar el motor, también se pueden construir sin sensores por lo que se clasifican en dos familias: con sensores (sensored), sin sensores (sensorless). (Cotte & Moreno, 2010)

A diferencia del motor de corriente continua con escobillas, los BLCD se controla electrónicamente, para accionar el giro, los bobinados del estator deberán estar activados secuencialmente. Para conocer la posición se utiliza sensores de efecto Hall que estar integrado en el estator. (Morales, 2018).

Para su puesta en marcha y control de la velocidad se utilizan métodos de conmutación como: control trapezoidal, senoidal, vectorial, siendo el más fácil la conmutación trapezoidal que consiste en controlar la corriente que circula por los terminales del motor, excitando simultáneamente un par, manteniendo el tercero desconectado, se alterna el par de terminales hasta realizar las 6 combinaciones posibles.

El par del motor de genera debido a la interacción entre el campo magnético de las bobinas, el del estator y los imanes permanentes. Generalmente el par máximo se genera cuando los dos campos se encuentran a 90° y cae cuando ambos campos magnéticos se mueven juntos. Para mantener en marcha al motor, el campo magnético de las bobinas debe cambiar de posición (activación y desactivación de las bobinas). Es decir que según varía el rotor, se actualice el campo del estator, como se muestra en la Figura 1.3.

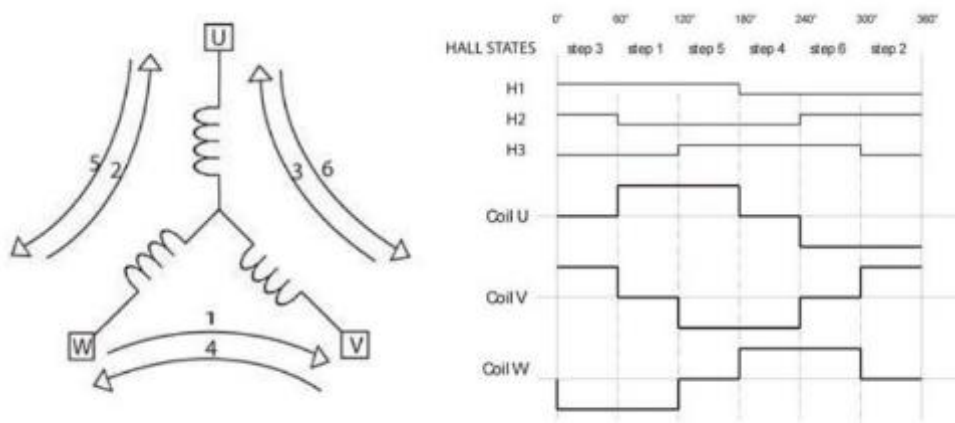


Fig.1.3. Secuencia de alimentación de las bobinas para el accionamiento del motor BLDC
Fuente: (Rodríguez M. , 2015).

1.1.4. Control de velocidad

Si ya se tiene un control de conmutación entre las bobinas del estator, se hace necesario realizar un control de velocidad del giro del motor. Convenientemente el campo magnético generado en el estator es proporcional a la tensión de alimentación y su velocidad será proporcional. Se busca regular la tensión que se suministra a los devanados para así modificar la velocidad de giro. Comúnmente la tensión es suministrada por una batería que alimenta un inversor trifásico que realiza la conmutación entre fases. (Fernández, 2016).

Debido a que la batería proporciona una tensión constante, en el control del inversor se aplica la técnica PWM. Que consiste en modificar el ciclo de trabajo de una señal periódica, es decir modifica el ciclo donde el valor es alto. En la Figura 1.4., se representa la gráfica de un PWM en una señal de 5V. (Fernández, 2016)

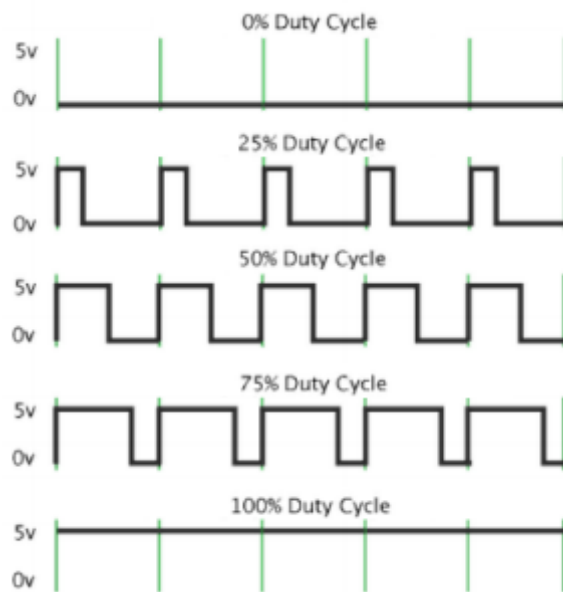


Fig.1.4. Modulación de ancho de pulso para control de velocidad
Fuente: (Fernández, 2016)

Los cargadores son dispositivos que permiten la carga total de las baterías del EV, el suministro de energía se realiza a través de la red eléctrica. El sistema de recargar permite almacenar la energía eléctrica en el banco de baterías. El proceso de recarga se debe ejecutar con base en criterios técnicos para el mantenimiento y conservación de la vida útil de las baterías, sin comprometer los materiales constructivos, proporcionando la energía necesaria para provocar la reacción electroquímica inversa en la batería. (Martínez, 2013).

Un controlador de carga debe ser capaz de evitar la sobre carga y sobre descarga de la batería, evitar que el voltaje de la batería retorne a la entrada del sistema de suministro, debe ser el medio de conexión entre la fuente de energía y la batería. (Saavedra, 2019).

El proceso de carga y descarga de la batería, se conoce como ciclado o ciclo, en el período de carga la batería acumula la intensidad eléctrica (régimen de carga), en el periodo de descarga, en cambio suministra la energía al sistema (régimen de descarga).

1.1.5. Sistema de monitoreo y control

Los sensores hall forman parte de la retroalimentación del sistema, nos proporcionan la posición del rotor necesarios para activar las bobinas de manera secuencial las bobinas del motor para su puesta en marcha.

- Sensor Hall.

El principio de funcionamiento se basa en que, si a un conductor conduce una corriente eléctrica, se genera un campo magnético a su vez genera una fuerza transversal en los portadores de carga que mueve a un a un lado el conductor siendo más evidente en conductores planos. (Rodríguez J. , 2015)

Una acumulación de la carga en los lados del conductor equilibrará la influencia magnética producida y producirá variaciones de tensión entre los lados. La diferencia de potencial transversal se llama efecto Hall, descubierta por Edwin Herbert Hall en 1879. (Morales, 2018).

En la mayoría de los BLDC los sensores hall se encuentran en el estator, en el extremo expuesto al rotor. Cada vez que los polos magnéticos del rotor pasan cerca de los sensores Hall, presentan una señal de “alto o bajo”, indicando el polo N o S está cerca. De acuerdo en estas señales del sensor se puede determinar la secuencia de conmutación. (Daghero, 2019)

En algunos motores los dispositivos están instalados en el rotor, los sensores Hall se encuentran normalmente en una placa de circuitos impreso y se fija a un extremo aislado, permitiendo ajustar el conjunto de sensores que pueden alinearse al rotor. Pueden estas a 60° o 120° de variación de fases. El fabricante define la secuencia de conmutación para el control del motor. (Morales, 2018).

Los sensores Hall necesitan de una fuente de alimentación externa, puede ser de 4V a 24V, con corrientes de 5 hasta 15 miliamperios. Para el diseño del controlador es necesario las especificaciones del motor, para poder trabajar con el sensor Hall adquirido. La salida de sensor normalmente es de tipo de colector abierto, por lo que se requiere una resistencia pull-up en la parte del controlador. (Rodríguez J. , 2015), el esquemático del sensor se muestra la Figura 1.5.

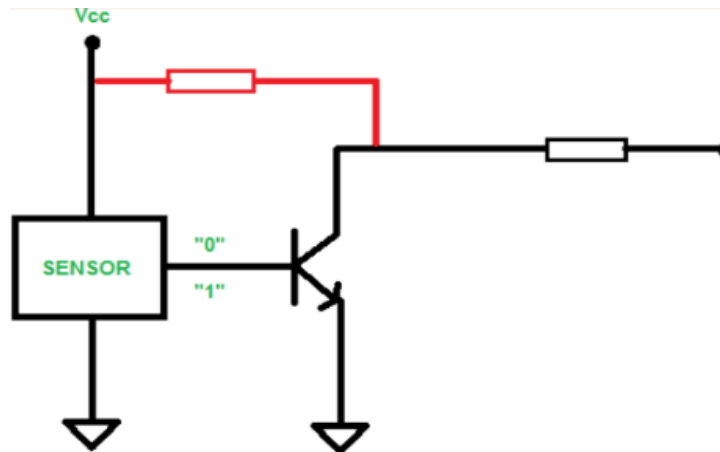


Fig.1.5. Sensor hall con transistor de colector abierto
Fuente: (Rodríguez J. , 2015)

1.1.6. Sistema HMI

Una interfaz gráfica hombre-máquina (HMI), presenta información al usuario sobre el estado de un proceso, acepta comandos y ejecuta ordenes en el sistema controlado. Se presenta como un formato realista del sistema. (Patil, Bidari, Sunag, Gulahosur, & Shettar, 2016).



Generalmente, los HMI reciben instrucciones a través de una pantalla táctil y contiene paneles táctiles con interfaces gráficos de color (GUI), la comunicación con el microcontrolador se puede realizar mediante la comunicación serial o mediante el uso de BUS CAN. Los sistemas HMI normalmente pueden ser utilizadas en plantas de producción, para control de vehículos no tripulados, también para procesos de automoción y para la interacción entre el usuario y el vehículo. (Department of Electronics and Instrumentation Engineering, 2017).

Los principales componentes de los HMI son: la unidad de detección táctil, puerto SPI, TFT(Thin Film Transistor) touch, reloj, un convertidor analógico digital (ADC), entre otros componentes. Los TFT LCD reaccionan al contacto con los dedos del usuario y pueden manejar gestos multitáctiles, para mejorar su utilidad. (Department of Electronics and Instrumentation Engineering, 2017)

Para un conductor es importante la visualización de los parámetros del vehículo, la finalidad del sistema es alertar oportunamente niveles de batería, velocidad, además muestra de forma gráfica la activación de los elementos como por ejemplo el encendido de luces. En los EV debido al número de variables que maneja se hace indispensable la implementación de los sistemas de visualización que serán útiles para la detección de posibles fallas y prevención de problemas de conducción. (Haro, Naranjo, Sarzosa, & Sotomayor, 2017).

Existen muchos fabricantes que ofertan pantallas HMI con diferentes características y diferentes versiones que tienen varias aplicaciones en la industria, también se pueden conectar con los diferentes dispositivos, a continuación, se describen algunas HMI que se encuentran en el mercado.

TABLA 1.2.
DISPOSITIVOS HMI

Fabricante	Descripción	Gráfico
Pro-face Standard HMI	Tienen una alta conectividad, amplia gama de controladores industriales, variedad de tamaños revestimiento y marco abierto. Adquisición rápida de información sobre disponibilidad de conexión, cableada. (Pro-face, 2021)	
SIMATIC HMI	Poseen paneles con pantallas de 4" hasta 12" en su serie básica y operación combinada con sus teclas táctiles, de alta resolución, tiene un puerto USB para conectar mouse o teclado. (SIEMENS, 2021)	

Fuente: Tomado de (SIEMENS, 2021) y (Pro-face, 2021)

Los dispositivos de la TABLA 1.2., son utilizados a nivel industrial y utilizan software propietario para su programación y resultan costosas. Para proyectos pequeños y medianos es muy popular el fabricante Nextion.

- HMI Nextion

Una de las opciones de HMI, son los dispositivos de Nextion, que proporciona una interfaz de visualización y control, entre el usuario y el proceso, se aplica en campos de internet de las cosas y en la electrónica de consumo. (Untuña, 2017)

Las pantallas Nextion TFT se presentan en diferentes tamaños que van desde 2.42" hasta los 10.1" de diferentes series, desde series básicas hasta modelos inteligentes que se puede observar en la Figura 6, la plataforma de desarrollo es el Editor Nextion.

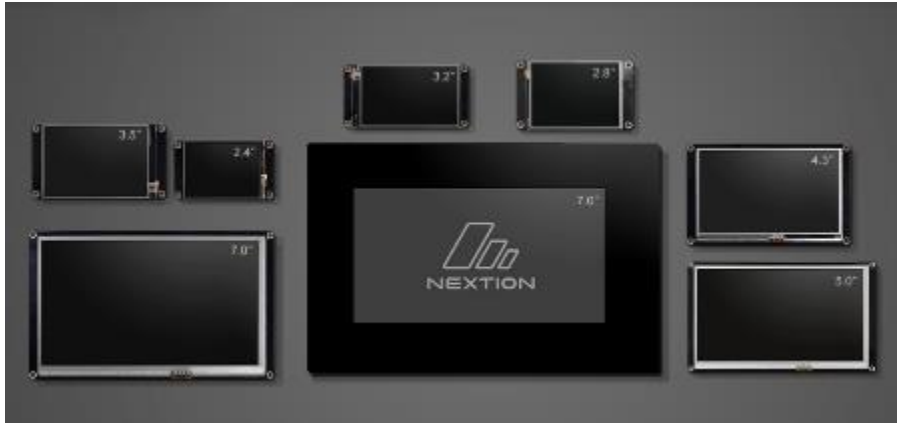


Fig.1. 6 Pantallas de Nextion de varios tamaños
Fuente: Tomado de (Nextion, 2019)

En la Figura 1.6., se puede observar los distintos tamaños de pantallas ofertadas por el fabricante, se elige en función del proceso que se va a indicar y los requerimientos del proceso. El tamaño ideal de la pantalla para mostrar los parámetros del scooter es aquella que donde se pueda visualizar correctamente todos los elementos de control, en primera instancia se analiza la opción de 5 pulgadas, debido a que es un tamaño adecuado para su implementación en el scooter.

1.5. Identificación de sistemas

Para el diseño de controladores es importante tener un modelo que describa el comportamiento de la planta o sistemas que se desea controlar. La identificación de sistemas trata de construir modelos matemáticos de sistemas dinámicos en base de las entradas y salidas observadas.

Teóricamente, se puede obtener un modelo matemático mediante método analítico o método experimental, el objetivo es conseguir una función de transferencia o unas ecuaciones de estados que describan el sistema identificado. (Arafet, Chang, Torres, & Dominguez, 2008)

Método analítico: determina las ecuaciones y parámetros que intervienen basándose en los principios físicos, químicos, o de cualquiera otra ciencia. Sin embargo, la complejidad de este método en la mayoría de los casos, son inadecuadas para la descripción de la planta. (Jarrín, 1996)

Método experimental: se considera al sistema como una “caja negra”, con determinadas entradas y salidas, como muestra la Figura 1.7.

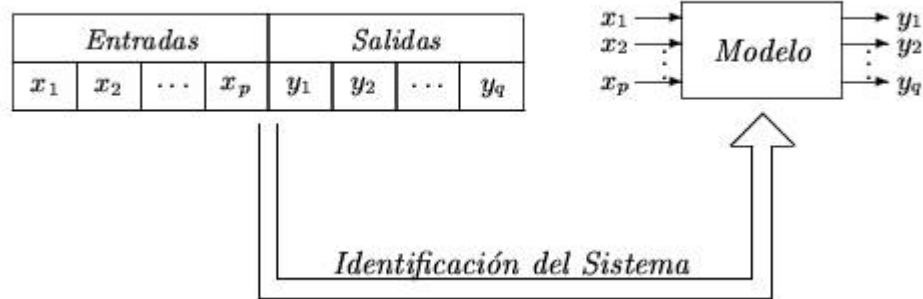


Fig.1.7. Esquema de identificación de sistemas
Fuente: (Lopez, 2014)

Al cual, se realiza un conjunto de experimentos para obtener medidas de las entradas y salidas mientras el sistema pase a estado estacionario, a partir de estos datos se determinará el modelo del sistema.

Para realizar la identificación es necesario 3 componentes básicos:

1. Los datos de entrada – salida, conseguidos por mediciones experimentales en la planta o mediante simulación.
2. Un grupo de modelos candidatos (estructura del sistema). Se debe plantear opciones para escoger el que se asemeje más al comportamiento de la planta.
3. Un criterio para seleccionar un modelo en particular de entre el grupo, basándose en la información de los datos obtenidos.

Para la identificación de sistemas se sigue los siguientes pasos:

- Realizar un experimento para obtener datos de entrada. Salida del proceso a ser identificado.
- Tratamiento de los datos obtenidos, aplicar filtrado para evitar las perturbaciones y mejorar la precisión.
- Seleccionar y definir un modelo continuo o discreto, si es necesario también se define el periodo de muestreo.
- Obtención de los parámetros del sistema de acuerdo con los datos medidos y un criterio dado de ajuste.

- Determinar si el modelo obtenido es suficientemente bueno, sometiéndole a una excitación común con el sistema real para luego comparar ambos resultados, si no son similares, posiblemente se intente con otro modelo o ampliar el rango de valores de entrada usados en la experimentación

En la Figura 1.8., se muestra el proceso de identificación del sistema.

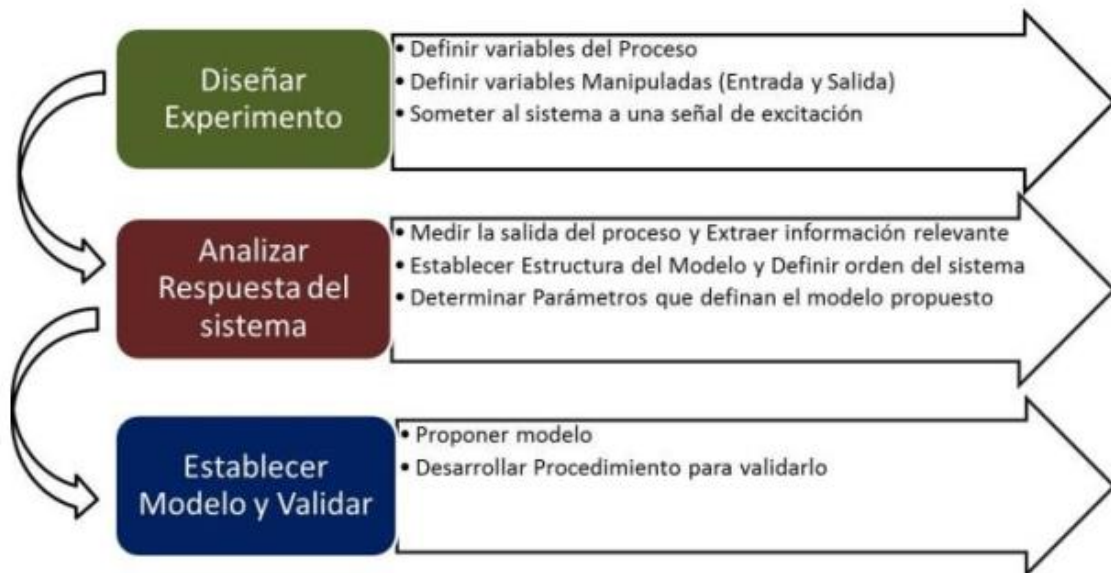


Fig.1.8. Proceso de identificación del sistema.
Fuente: (Vergara, Salazar, & Zapata, 2017)

El problema que surge con la identificación es que se complica cuando se trata de sistemas no lineales, en su mayoría se encuentra con esto, Sin embargo, se puede linealizar en torno a un punto de operación, pero si el sistema pasa a funcionar en un punto de operación lejano, no funciona adecuadamente. (Lopez, 2014)

Según las características del modelo se pueden clasificar en:

Modelos paramétricos: se desea obtener los valores de los coeficientes de las funciones o matrices de transferencia, o los elementos de las matrices de representación en espacios de estados.

Modelos no paramétricos: Es el caso de las gráficas de modulo y fase en las respuestas frecuenciales. En estos métodos se encuentran, por ejemplo: Bode, Nyquist y Nichols.

En la Figura 9 se muestra gráficamente la clasificación de los modelos.

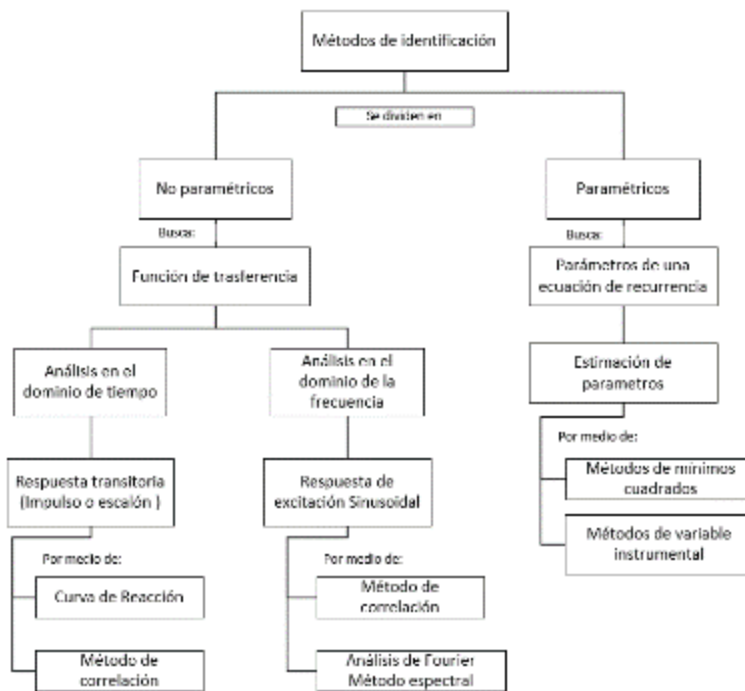


Fig.1.9. Clasificación de los métodos de identificación de sistemas
Fuente: (Vergara, Salazar, & Zapata, 2017).

1.6. Control PID para regular la velocidad de los motores

Para el control de la velocidad de los motores BLDC, por lo general se usa dos tipos: los de bucle abierto y los de bucle cerrado siendo el más común los de control de velocidad de bucle cerrado, es necesario considerar la influencia de los parámetros del motor que varían en función del tiempo (resistencia, inductancia y momento de inercia). (Cotte & Moreno, 2010).

- Definición de control PID

Los controladores PID son muy utilizados en la industria como una estrategia en control lineal debido a su simplicidad y su robustez, fiabilidad y fácil ajuste de parámetros, su estructura es la siguiente:

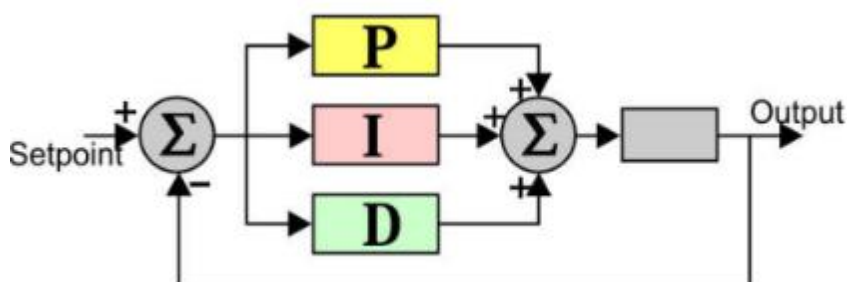


Fig.1.10. diagrama de un controlador PID
Fuente: (Herrera P., 2017)

Los términos proporcionales, integrales y derivativos; conocidos también como P, I y D. University, Istanbul Technical, (2016) afirma:

- P: representan los valores actuales del error. Por ejemplo, si el error es grande y positivo la salida de control será igual.
- I: toma en cuenta los valores anteriores del error. Por ejemplo, si la salida de corriente no es lo suficientemente fuerte, el error se irá acumulando y con el tiempo la acción de control será más fuerte.
- D: tiene en cuenta los posibles valores futuros de error, basándose en su tasa de cambio actual. (p.2).

Los coeficientes k_p , k_i , k_d , son constantes, que tiene efecto directo sobre la acción del controlador. En la Tabla 1.3. se indica el efecto que produce cada termino sobre el controlador.

TABLA 1. 3.
EFECTO DE LOS PARÁMETROS DEL CONTROLADOR

Parámetro	Tiempo de subida	Sobre pico	Tiempo de establecimiento	Error de estado estable
Kp	Reduce	Aumenta	Pequeño cambio	Reduce
Ki	Reduce	Aumenta	Aumenta	Elimina
Kd	Pequeño cambio	Reduce	Reduce	Pequeño cambio

Fuente: (Delgado & Bolaños, 2013).

1.7. Control de velocidad avanzados para accionamiento de BLDC

El controlador PID se utiliza generalmente como controlador de velocidad para motores, caracterizado por su robustez, simplicidad y confiabilidad, que se han mejorado e introducido nuevos métodos matemáticos (el método de descenso de gradiente), y métodos inteligentes (redes difusas y neuronales) con el fin de obtener una versión eficiente de un control PID adaptativo. En resumen, el control PID adaptativo ajusta la línea de los parámetros del control PID mejorando el rendimiento del sistema, además se puede aplicar en más áreas. (Ezzaldean, 2018)

1.7.1. Control difuso

El control difuso proporciona según (Herrera S., 2016): “una metodología formal para representar, manipular e implementar el conocimiento heurístico del ser humano acerca de cómo controlar un sistema” (p.29.).

Un sistema de control difuso típico se conforma de un controlador difuso y la planta, que consta de 4 componentes: fusificación, base de datos (base de reglas), interferencia difusa y defusificación, como se muestra en la Figura 1.11. Se compara con el razonamiento humano, este método de control inteligente que es independiente del modelo matemático del objeto de control ya sea lineal o no lineal, el controlador puede ser implementado de manera efectiva y posee robustez y adaptabilidad. (Chang-liang, 2012)

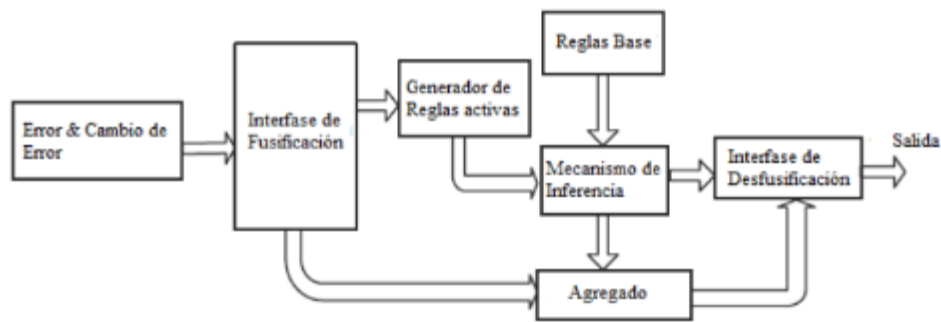


Fig.1.11. . Esquema de control difuso
Fuente: (Herrera S. , 2016)

En la Figura 1.11., se muestra que en el control difuso intervienen: las reglas básicas; que constan de cuantificaciones lógicas difusas para conseguir un control óptimo, el generador de reglas activas; en donde cada variable genera una dirección de memoria, también consta de un mecanismo de inferencias, una interfaz de fusificación, una etapa de agregado y una interfaz de defusificación.

Aplicaciones en los vehículos eléctricos.

El control difuso se puede aplicar en los distintos campos de los EV. Que se describen en la Tabla 1.4.

TABLA 1. 4.
 APLICACIONES DEL CONTROL DIFUSO.

Aplicaciones del control difuso.		
Aplicaciones	Descripción	Referencia
Control de frenado regenerativo	El control difuso maximizará el uso del frenado del motor, y mejorará la eficiencia de recuperación de energía del frenado. El par de frenado total y el par de frenado de las ruedas se parece al de un vehículo convencional sin sistema de frenado, a su vez mantiene el par de frenado eléctrico máximo.	(Wu, Cui, Li, Zhang, & Li, 2018)
Control de tracción	Mediante la aplicación de control difuso se busca mejorar el reconocimiento de las intenciones abstractas de aceleración, verificado el estado actual las condiciones del vehículo y el cambio del estado de pedal que representa las intenciones de aceleración, el par de referencia se compensa de acuerdo con las diferentes intenciones de aceleración. Mejorando el rendimiento dinámico y el rendimiento operativo, haciendo que el vehículo sea más práctico en condiciones de escalada de pendientes, se mejora entre el 5 % a 10% la eficiencia en comparación con las estrategias de control tradicionales.	(Cheng, Xiong, Wang, & Zhang, 2019)
Control de carga.	Se han desarrollado método de carga inteligente mediante la utilización de la teoría difusa para el control de estación de carga, de acuerdo con si se necesita la carga rápida o lenta dependiendo del estado de carga de la batería. Para la carga de los vehículos el control difuso calcula el factor de prioridad, cargando primero los EV que representan mayor peso. El control difuso utiliza variables de control como el voltaje medio común a través de los convertidores de potencia (MVDC),y el estado de carga del batería(SOC).	(Te, Bwo, Yu, & Chen, 2017) y (Triviño, Fernández, Torreglosa, & Jurado, 2016)

1.7.2. Control de red neuronal.

Una red neuronal se conforma por una capa de entrada, un número determinado de capas ocultas y una capa de salida, entre cada una de las capas existen matrices de pesos sinápticos, que proporciona a la red su potente cálculo y capacidad de aprendizaje, además varían a medida que el sistema aprende. (Torrubia, 2010). El modelo se muestra en la Figura 1.12.

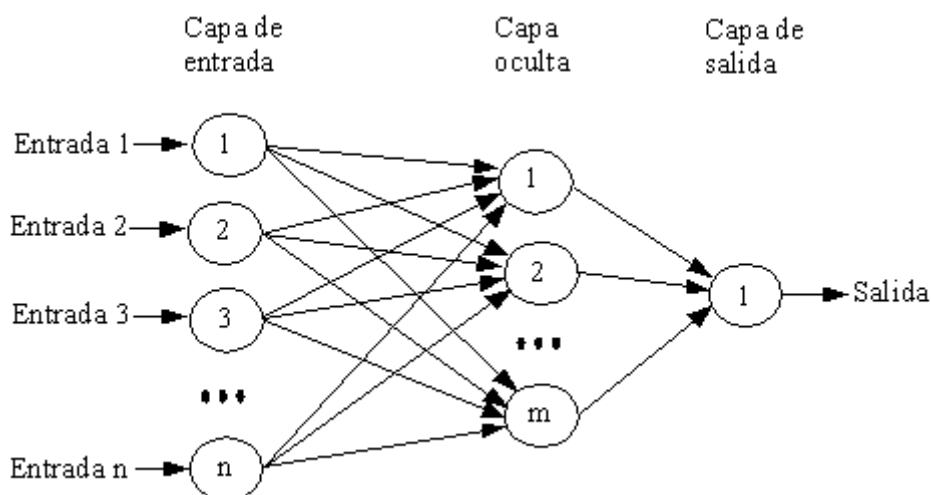


Fig.1. 12. Modelo de una red neuronal artificial
 Fuente: (Torrubia, 2010)

Las redes neuronales artificiales se aplican al control de motores síncronos de imán permanentes, motores de reluctancia conmutada, motores ultrasónicos y motores BLDC y otro tipo de motores, normalmente se controla la posición del roto, la velocidad, además la identificación y estimación de los parámetros del motor. (Chang-liang, 2012). Es una opción para el desarrollo de sistemas de control en los cuales se dificulte la modelación matemática o computacional. (Castillo, 2010) En las aplicaciones de control de velocidad de motores BLDC se encuentran a redes neuronales de retro programación (BP) , redes de función de base radial(RBF), redes wavelt, redes de neuronales simples entre otras. (Chang-liang, 2012).

Las redes neuronales tienen tres niveles de análisis, en el primero se desarrollan sistemas de aprendizaje supervisado, en el segundo se construyen sistemas más complejos en donde se realiza la reproducción y optimización dinámica del sistema. El tercer nivel se muestra con otros sistemas con el fin de la creación de sistemas complejos para aplicaciones específicas. (Castillo, 2010).

Aplicaciones de redes neuronales al control de motores BLDC.

Mediante la aplicación de redes neuronales aplicados al control de motores BLDC se describen en la Tabla 1.5.

TABLA 1. 5.
CONTROL DE BLDC CON APLICACIONES DE REDES NEURONALES
Aplicaciones de redes neuronales al control de motores BLDC.

Aplicaciones	Descripción	Referencia
Modelo inverso de la velocidad del motor BLDC	El control neuronal que se basa en la red neuronal artificial mide el comportamiento dinámico del sistema para obtener datos, esta capacidad es adecuada para aplicarse en sistema donde el controlador debe adaptarse a los cambios de comportamientos sistema. Es usado para obtener buenas respuestas dinámicas. Se realiza la capacitación del sistema con el fin de mejorar el valor ponderado hasta que la salida del ANN sea igual a la salida objetivo.	(Budi, Rizal, & Onnilita, 2017)
Sistema de control de orientación de campo del motor BLDC	Se realiza para mejoramiento del rendimiento del sistema de control del motor BLDC, este sistema combina la red neuronal con el FOC. Dando como resultado la disminución de la ondulación de par y de la velocidad. Además, la respuesta del sistema es más rápida con una operación más estable.	(Jin-feng, Lei, Min, & Fei, 2012)
Método de Conmutación sin sensores.	La red neuronal RBF se aplica para compensar el error de conmutación. Cuando los devanados se energizan en el tiempo de conmutación exacto, los voltajes de EMF son iguales, con un error de conmutación los voltajes no son iguales. Mediante la aplicación del método se puede compensar efectivamente los errores de conmutación tanto en altas velocidades como a velocidad constante.	(Chen, Li, Sun, & Liu, 2018)

Fuente: Autor

1.7.3. Grey control.

La teoría del sistema de Gray-Predictor se usa como una herramienta que se utiliza para la predicción. Utiliza solo pequeñas cantidades de datos para obtener predicciones precisas sin realizar excesivos cálculos. Se basa en la Operación de Generación Acumulada (AGO) en la secuencia de información para producir una nueva serie para luego ser usada para formular ecuaciones diferenciales y los coeficientes pueden ser obtenidos por utilizando el algoritmo de mínimos cuadrados. Se puede utilizar para predecir la variación de la velocidad del motor. (Ezzaldeen, 2018).

La teoría de control grey no necesita reglas de distribución de funciones, siendo superior en la resolución de problemas con un mínimo de información que pueden ser inexacta e incompleta. El control de motores típico puede considerarse un sistema gris, debido a que las perturbaciones de los parámetros internos y la carga del motor pueden considerarse incertidumbres. Se pueden obtener rendimientos de control requeridos tanto para los motores de inducción, BLDC, síncronos. (Chang-liang, 2012).

1.8. Regulaciones para el uso de scooter eléctricos

Los scooters eléctricos se incluyen en el sistema de transporte urbano por ser una nueva solución de movilidad y se está volviendo cada vez más popular en el caso de los operadores de movilidad compartida y también en la nueva tendencia social. La aparición de unas grandes cantidades de scooters en condiciones de tráfico urbano puede ocasionar problemas relacionados con la seguridad y cuestiones legales. Los problemas recurrentes tienen que ver con asuntos relacionados con el movimiento y el estacionamiento en las calles, intersecciones y pavimentos. (Turoñ & Czech, 2019).

Los problemas ocasionados por los scooters eléctricos han llevado a la creación de políticas y regulaciones orientadas al usuario de estos dispositivos. A continuación, se describen algunos ejemplos en los diferentes países del mundo.

En Reino Unido se prohíbe el uso de scooters eléctricos en aceras y calles, su utilización se limita a la propiedad privada con previa autorización del propietario. La penalización por incumplir prohibición es de £ 300 y reducción de 6 puntos a la licencia de conducir. Además, se consideran vehículos personales eléctricos ligeros (PLEV), con motor por tal razón están sometidos a las normas y políticas normales de un vehículo tradicional, por falta de autorización no se permite el uso. (BBC NEWS, 2019).

En Francia no se permite la circulación por aceras con la excepción si se está empujado con el motor apagado, la velocidad máxima permitida es de 25 km/h en ciclo vías y en

carreteras a 50km/h. Su estacionamiento se restringe a los lugares designados para este fin, y solo pueden transportar una persona. Las penalizaciones por estacionamiento en lugares prohibidos son de € 35 y por tráfico en acera de € 135. (RFI, 2019).

En el caso de Australia, permite velocidades de hasta 25 km/h, la circulación en ciclo vías es permitida, deben someterse a los mismos límites de velocidad y leyes de tránsito, deben tener implementadas frenos, timbre e iluminación. (Gesley, 2019)

En Alemania, todos los scooters deben tener autorización de tráfico. La velocidad máxima permitida es de 20 km/h, no existe la obligación de conducir con casco, pero si debe tener todos los equipamientos como: iluminación y freno de mano. (The Focal, 2019).

En Estados Unidos existen regulaciones de acuerdo con cada estado, en California el usuario debe tener una licencia de conducir, es obligatorio el uso de casco, con velocidad máxima de 15 mph, no es permitido el uso de aceras. En cambio, en Texas es permitido la circulación por aceras con velocidad menor a 35 mph, se aplica las mismas leyes que a las bicicletas eléctricas. En New York se permite libre circulación en aceras y calles.

En ciudades latinoamericanas se está comenzando a regularizar el uso de los scooters eléctricos debido a los accidentes e inconvenientes ocasionados. Por ejemplo, en Lima se prohíbe el uso en aceras, en Ciudad de México únicamente pueden circular por el carril de bicicletas o calles, se recomiendan el uso de casco, además impuso un límite a la cantidad de MVP por empresa. En Bogotá se exige el uso de casco, mientras que en Brasil se puede circular a una velocidad de 6 km/h en carriles de bicicletas. (The Star , 2019).

A nivel nacional no se cuenta con regulaciones que controlen el tránsito de los VMP. Sin embargo, el secretario de Movilidad Guillermo Abad señala que estos vehículos deben estar dentro de una norma jurídica y reglamentaria y deben ser legislados por la ley de tránsito. Además, se está trabajando establecer los reglamentos idóneos basados en normas internacionales. (El Comercio, 2019).

CAPÍTULO 2

Diseño del sistema de control integrado de velocidad para un prototipo de vehículo eléctrico (scooter).

2.1. Introducción

En el siguiente capítulo, se describe el procedimiento para el desarrollo de la programación del microcontrolador seleccionado, a partir de los elementos que se disponen en función de sus características técnicas. Se identifica las variables y elementos que se controlan del scooter. Se muestran los dispositivos utilizados para la implementación del sistema. Se presenta el diagrama de conexión de los diferentes elementos para su correcto funcionamiento.

2.2. Descripción de softwares

Para la realización del proyecto se utilizó los siguientes programas:

The open-source Arduino Software (IDE): permite la creación de algoritmos en donde se definen las entradas y salidas de la placa, así como condiciones de control de las variables y componentes del scooter, mediante este software se carga todas las instrucciones al microcontrolador.

Paralaje Data Acquisition herramienta (PLX-DAQ): utilizado para la adquisición de datos en de la respuesta del motor, guarda los valores obtenidos en una tabla de Excel.

Nextion Editor: Se utilizó para la creación de la interfaz gráfica, donde se puede agregar botones, textos, slider, un panel de instrumentos. Los cuales pueden ser usados para la visualización de los distintos parámetros del sistema de control. Se pueden implementar un panel HMI a proyectos existentes, se debe tomar en cuenta que usar el protocolo de comunicación UART (Universal Asynchronous Receiver-Transmitter). (Calapiña, 2018).

MATLAB 2019: Mediante el uso de sus herramientas, permitió la identificación de sistema, en este caso del motor DC Brushless, permitiendo la estimación de los parámetros del controlador.

Inkscape: utilizado para el diseño de la interfaz gráfica de las ventanas del HMI agregando imágenes, texto, color, etc.

2.3. Metodología

La secuencia de actividades que se siguieron para el desarrollo del proyecto, fueron:

- Recopilación de información, que permitió tener una base de conocimiento en la utilización del microcontrolador, al igual que sus actuadores, sensores y periféricos, así como la pantalla touch para la implementación de interfaz gráfica.
- Desarrollo de estrategias de pruebas utilizando los dispositivos existentes y las demás herramientas seleccionadas. En esta sección se implementó y se probó la técnica de control propuesta, desarrollando el código de control de velocidad, así como de los demás elementos del sistema como: luces; direccionales, velocímetro de acuerdo con los parámetros deseados.

Se muestra un diagrama de flujo en Figura 2.1, donde se muestra la metodología realizada para la realización del presente capítulo

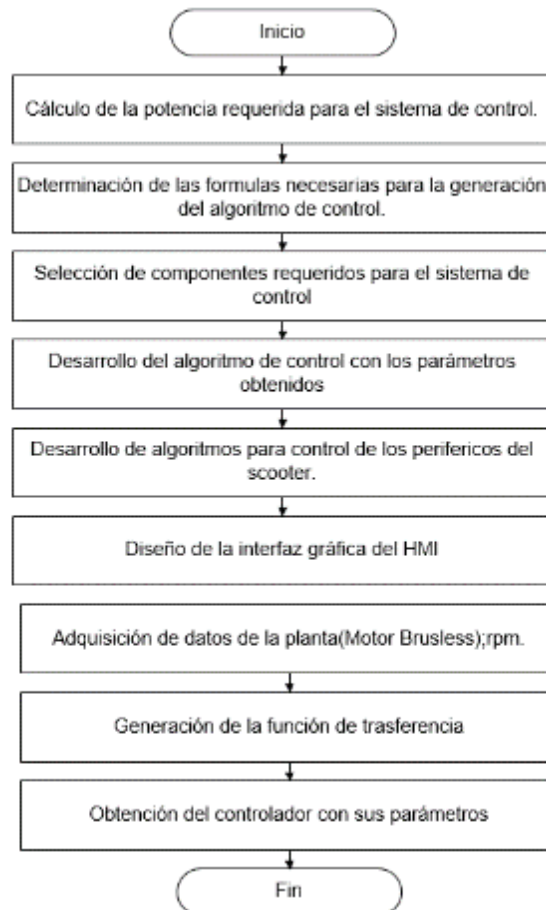


Fig.2.1. Estructura de desarrollo del sistema de control.

Fuente: Autor.

2.4. Formulación de parámetros matemáticos

2.4.1. Determinación del regulador de voltaje

El sistema de control requiere de una potencia que deberá ser suministrada por la batería, debido a que el voltaje es muy alto se utiliza un DC-DC Buck. Para conocer la potencia consumida por los dispositivos conectados al regulador, se utiliza la ecuación (1). (Navalón, 2015)

$$W_c = V_{out} * I_{max} \quad (1)$$

W_c = Potencia del sistema de control

V_{out} = Voltaje de salida del convertidor DC-DC Buck

I_{max} = Corriente máxima del sistema

2.4.2. Caída de tensión debido a DC-DC Buck

Se debe considerar la caída de tensión y la corriente de entrada proporcionada por la batería, el convertidor deberá disipar esa potencia, provocando daños al sistema en caso de superar el límite. Se comprueba numéricamente que cumpla los requisitos en el caso más desfavorable con la ecuación (2) y (3) (Navalón, 2015).

$$I_{in} = \frac{W_c}{V_{in}} \quad (2)$$

I_{in} = Corriente de entrada al DC-DC Buck

V_{in} = Voltaje de entrada

$$W_{buck} = V_{buck} * I_n \quad (3)$$

W_{buck} = Potencia disipada

V_{buck} = Voltaje de salida

2.4.3. Sensor de voltaje

Es necesario para medir el voltaje de la batería para dar conocer el nivel de carga, y debido a que el voltaje de la batería no debe de descender de un valor en concreto, en tal caso podría sufrir daños la batería.

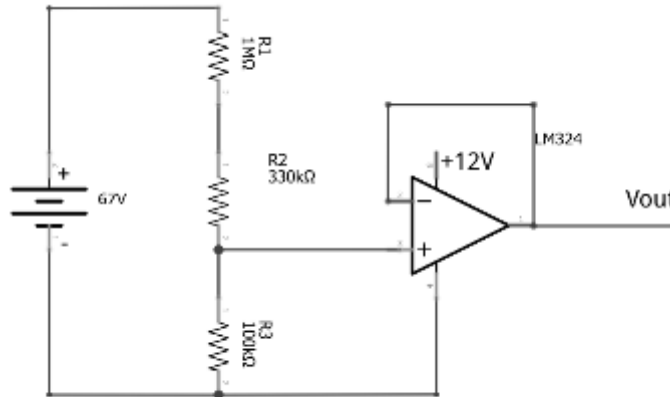


Fig.2. 2. Circuito divisor de tensión para la medición de una batería de 67V

Fuente: Autor

Además, el microcontrolador puede leer señales de máximo 5V, por lo tanto, para lectura de tensiones mayores se utiliza un divisor de tensión, con el fin de tener tensiones en el rango de 0 a 5V, como se muestra en la Figura 2.2. Se utiliza resistencia en el orden de MOhmios. Para determinar el valor de la resistencia se utiliza ecuación 4 (Navalón, 2015)

$$V_{out} = V_{in} * \frac{R3}{R1 + R2 + R3} \quad (4)$$

V_{out} = Voltaje de salida al microcontrolador

V_{in} = Voltaje de entrada de la batería

R1, R2, R3 = Resistencia del divisor de voltaje.

2.4.4. Sensor de velocidad

Las RPM del motor se miden mediante la terminal de velocidad del controlador que maneja un rango de 0 a 32 V en función de las RPM del motor, al igual que el apartado anterior se realiza un divisor de voltaje. Para luego escalar el valor entre valores de 0 a 455 rpm, que puede alcanzar el motor.

2.4.5. Distancia lineal recorrida

Para determinar la distancia lineal recorrida por el scooter se usa la ecuación 6 que se incluye en la programación del microcontrolador. (alphons, 2021)

$$V_{Km/h} = \frac{3}{25} * \pi * r * N(rpm) \quad (5)$$

$V_{km/h}$ = velocidad en km/h

r = radio de la rueda

w = velocidad angular

2.4.6. Filtro paso bajo y alto exponencial EMA

Debido a que puede presentarse ruido en las señales de los sensores empleados que puede ocasionar desestabilización en la respuesta del sistema fue necesario la implementación de un filtro que permita mejorar o suavizar la señal recibida por el microcontrolador. El filtro EMA(Exponential Moving Average), parte de la fórmula matemática (Llamas, 2017).

$$A_n = \alpha M + (1-\alpha)A_{n-1} \quad (6)$$

A_n = El valor filtrado

A_{n-1} = el valor filtrado anterior

M = El valor de muestreo de la señal a filtrar

α = factor con valores entre 0 a 1

2.4.7. Modelo matemático del controlador PID

El controlador PID estándar calcula la desviación $e(t)$ entre el valor de referencia y el valor actual. Entonces, la planta es controlada por la variable $u(t)$ con una combinación lineal de términos proporcionales-integrales-derivados. La ley de control PID correspondiente en continuo la forma se puede expresar como la ecuación 7 (Cotte & Moreno, 2010)

$$u(t) = K_p(e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt}) \quad (7)$$

Donde K_p , es la ganancia proporcional, K_i es la ganancia integral y K_d es la ganancia derivativa. Cada uno de los parámetros de la función de transferencia cumple una función específica dentro del comportamiento del sistema.

Frecuentemente el controlador PID que se utiliza en motores modernos, con el fin de mejorar la confiabilidad del sistema, en esta situación la ecuación del control continuo PID debe ser discretizada para que sea comprensible para el microcontrolador. Se puede cambiar con una buena aproximación como la siguiente ecuación utilizando la transformada Z:

$$U(z) = \left[K_p + \frac{K_i}{1 - z^{-1}} + K_d(1 - z^{-1}) \right] E(z) \quad 8$$

Reorganizando:

$$U(z) = \left[\frac{(K_p + K_i + K_d) + (-K_p - 2K_d)z^{-1} + K_d z^{-2}}{1 - z^{-1}} \right] E(z)$$

$$K_1 = K_p + K_i + K_d \quad 9)$$

$$K_2 = K_p - 2K_d$$

$$K_3 = K_d$$

Que se puede escribir:

$$U(z) - z^{-1}U(z) = [K_1 + K_2 z^{-1} + K_3 z^{-2}] E(z) \quad 10$$

Y se puede convertir en:

$$u[k] = u[k - 1] + K_1 * u[k] + K_2 * u[k - 1] + K_3 * u[k - 2] \quad (11)$$

Así, $u[k]$ quiere decir la salida de control actual, $u[k - 1]$, es la salida anterior del PID, Y K_1 , K_2 , K_3 , son parámetros del error actual, del error anterior y el error antes anterior.

2.4.8. Obtención de la función de transferencia mediante el análisis de la curva de reacción

El método de identificación de sistemas mediante la curva de reacción se utiliza para determinar los parámetros que conforman la función de transferencia. Los sistemas de primer orden con retardo tienen la estructura de la ecuación 12. (Vergara, Salazar, & Zapata, 2017)

$$\frac{y(s)}{u(s)} = \frac{K^{-Ls}}{\tau s + 1} \quad (122)$$

$$K = \frac{y_f - y_i}{u_f - u_i} \quad (13)$$

Donde:

$y(s)$ = Señal de salida del sistema

$u(s)$ = Señal de entrada del sistema

K = Ganancia estática.

τ = Constante de tiempo, corresponde al tiempo en el cual la salida alcanza el 63,2% del valor final

L = retardo.

La respuesta del sistema de primer orden con retardo se muestra en la Figura 2.3.

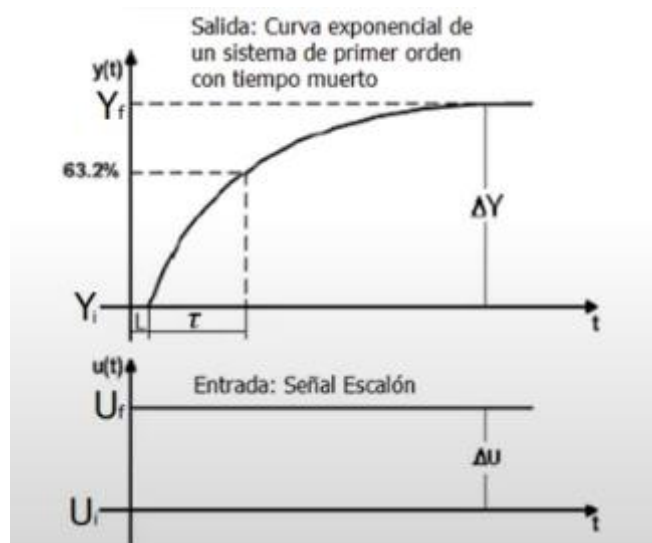


Fig.2.3. Respuesta transitoria de primer orden

Fuente: (Vergara, Salazar, & Zapata, 2017)

2.5. Descripción de equipos y dispositivos electrónicos

A continuación, se presenta un diagrama de bloques general con los distintos elementos del prototipo.

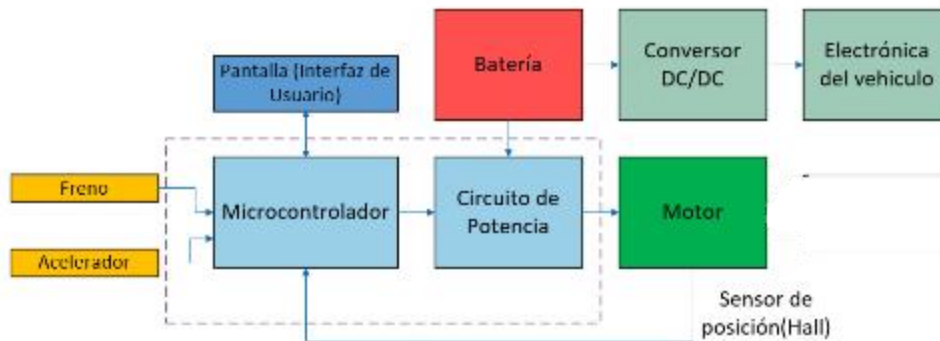


Fig.2. 4 Esquema básico del scooter eléctrico

Fuente: Autor

En el esquema se trata de representar los principales componentes del sistema y su relación, que muestra una visión global del funcionamiento del prototipo, sin considerar la alimentación de cada uno de ellos. El driver de potencia es el encargado del movimiento del motor, que acelere o frene según la necesidad del usuario, las ordenes son enviadas por el microcontrolador que interpreta las señales recibidas del exterior y del motor para clasificarlos y ordenarlos con una lógica de control, además la interfaz no solo muestra el estado del vehículo, si no que permite participar en el control. La electrónica del vehículo contempla la alimentación de los sensores, sistema de iluminación y el panel táctil.

2.5.1. Elección de componentes

Para el desarrollo del sistema de control se requieren los componentes mencionados en la figura anterior, de los cuales se deben especificar las características y motivo de la selección, los equipos necesarios son: microcontrolador, driver inversor, pantalla, conversor dc-dc reductor, sensor para determinar la velocidad.

La selección de cada componente se realizó de acuerdo con una ponderación de un puntaje a las características más relevantes de equipos similares. Se asigna valores entre 0 a 1 resultando con mayor puntaje equipo o dispositivo que se ajusta a los requerimientos del sistema de control. La ponderación de los valores se muestra en la Tabla 2.1.

TABLA 2. 1.
PUNTUACIÓN DE CARACTERÍSTICAS Y SIGNIFICADO

Valor Numérico	Significado
0	No adecuado
0.5	Medianamente adecuado
1	Apropiado

Fuente: Autor

2.5.2. Controlador del motor Brushless

El inversor energiza secuencialmente las bobinas del estator que generan un campo eléctrico giratorio que “arrastra” al rotor. Al tener una secuencia apropiada para suministrar a las fases del estator se mantiene un campo giratorio que conserva al rotor persiguiendo a los polos del electroimán. El control preciso de la velocidad y torsión garantiza que el motor funcione con el pico de eficiencia.

Al usar los inversores es necesario el control de voltaje de salida del inversor en función de las variaciones de la entrada.


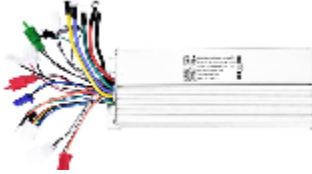

Las características generales de un inversor son:

- Nivel de tensión de la batería y del motor
- Nivel de corriente de la batería y del motor
- Frecuencia de conmutación

En la etapa de potencia, en lugar de diseñar desde cero un inversor trifásico (seleccionando los MOSFETs y adaptándolos en una placa de conexiones (PCB)) se optó por adquirir un inversor trifásico comercial específico para el control del motor Brushless seleccionado.

Existen varias marcas y especificaciones de inversores trifásicos que se pueden encontrar con el nombre de “Brushless Motor Controller Driver”, se pueden encontrar varios modelos distintos según la potencia requerida o aplicación del motor Brushless. Para la selección del driver se consideran los parámetros de la batería y del motor. A continuación, se describen los controladores disponibles en el mercado para la aplicación de control de motores Brushless.

TABLA 2. 2.
COMPARACIÓN DE TIPOS DE DRIVES PARA MOTOR BRUSHLESS

	Tarjeta de control de velocidad de motores BLDC	Controlador de motor Brushless de aleación de aluminio	Controlador de alta velocidad sin escobillas BLDC
Imagen referencial			
Protección y disipador de calor	0 (No dispone)	1 (Carcasa de aluminio)	1 (Carcasa de aluminio)
Facilidad de adquisición en el mercado nacional	0.5	1	0
Potencia Máxima	0 350W-700W	1 350W-4000W	1 500W
Soporte de regulación de velocidad	1 Si	1 Si	1 Si
Precio	1	1	0.5
Tipos de puertos para implementación.	0 (Se debe soldar los terminales)	1 Terminales de conexión	1 Puertos de conexión.
Total	2.5	6	4.5

Fuente: (amazon, 2021), (aliexpress, 2021)

Además, se consideró las especificaciones como: número de fases, potencia, voltaje de alimentación y conexión de los sensores de efecto Hall. Debe tener una entrada para la señal de velocidad de referencia que es mediante la aplicación de un voltaje análogo de 1V A 4V.

Se seleccionó el inversor que se muestra la Figura 2.5., de acuerdo con la Tabla 2.2., que tiene las características adecuadas para el scooter. Tiene los siguientes terminales de conexión:

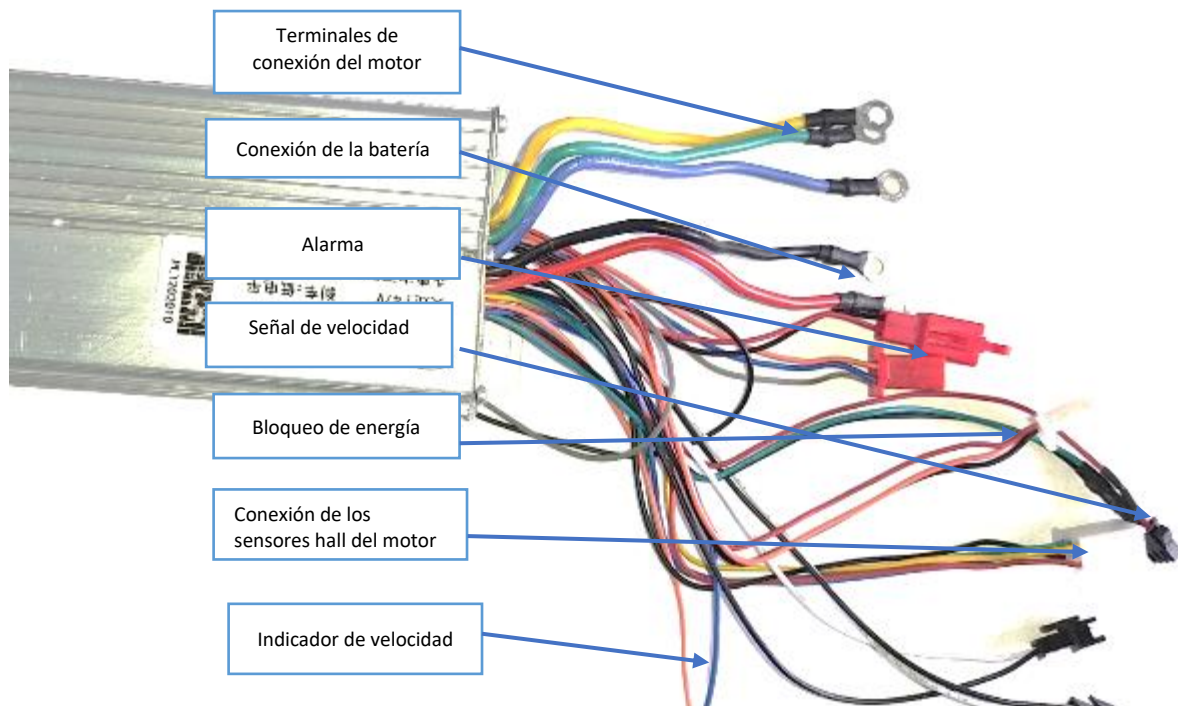


Fig.2. 5. Terminales de conexión del inversor trifásico

Fuente: Autor

- **Terminales de conexión del motor:** Fase A, Fase B, Fase C, respectivamente, mostrada en la Fig.2.5
- **Conexión con la batería:** VCC (Alimentación positiva) y GND (conexión a masa o alimentación negativa).
- **Señal de velocidad:** entrada de la señal de un voltaje análogo para el control de la velocidad en un rango de 1V a 4V.
- **Bloqueo de energía:** Vout, señal, GND
- **Conexión de los sensores hall:** +5V, hall A, hall B, hall C, GND.
- **Terminal indicador de velocidad:** +32V(Azul)

El inversor seleccionado tiene método de control sinusoidal que requiere de sensores de efecto hall. Además, su voltaje de entrada se ajusta a la batería y motor de 1500 W, que dispone el prototipo de scooter.

Características técnicas:

- Voltaje de operación y alimentación: 67V.
- Potencia: 1500W.
- Operación en bajo voltaje: 47V

2.5.3. Sensor de voltaje

Las resistencias utilizadas en el divisor de tensión deben estar en el orden de kohmios, se debe incluir un diodo para que no se produzca una circulación de corriente indeseada desde la batería al microcontrolador. Para determinar el valor de R3 de divisor de voltaje, donde existe una caída de tensión de 5V, se aplica la ecuación 4.

$$5V = 67V * \frac{R3}{R1 + R2 + R3}$$

$$\frac{R3}{R1 + R2 + R3} = 0,075$$

$$R3 = 1078337.838 \Omega \approx 100k\Omega$$

Para el cálculo se utilizó valores de R1=1MΩ, R2=330kΩ.

Debido a la impedancia de las entradas del Arduino, que cambia valor de la resistencia con ello el valor de salida del sensor de voltaje, fue necesaria la implementación de un seguidor de voltaje que permite tener a la salida del circuito la misma tensión que la de entrada sin cambiar el valor de las resistencias. Para esto se utilizó un amplificador operacional que se describe a continuación.



Fig.2. 6. Amplificadores operacionales cuádruple

Fuente: (Electronica plug and play, 2021)

En la Figura 2.6. se muestra un amplificador operacional cuádruple LM324, consta de 4 amplificadores operacionales utilizado para realizar un seguidor de voltaje

Características:

- Voltaje de alimentación: 3 V to 32 V
- Drenaje de corriente de suministro muy bajo (700 μA)
- Rango de voltaje de entrada diferencial igual al voltaje de la fuente de alimentación
- Cuatro amplificadores operacionales compensados internamente en un solo paquete

2.5.4. Sensor de acelerador y freno

El sensor de aceleración es el encargado de enviar al microcontrolador la velocidad de referencia, también proporcionan comodidad al agarre del volante del scooter. El acelerador seleccionado consta de un mango de agarre que va colocado en el volante del scooter, posee 3 terminales que se muestran en la Fig.2.7.



Fig.2. 7. Acelerador manual.

Fuente: Autor

- **Terminal de conexión:** +5V(rojo), GND (negro), Vout (Verde).

Los terminales de conexión alimentan el sensor hall situado en la manija del acelerador, el cual varía el voltaje al acercarse a un imán, y mediante el terminal Vout se da lectura de la posición del acelerador, su funcionamiento se asemeja a la de un potenciómetro, sin embargo, no posee partes mecánicas que tienden a desgastarse motivo por el cual fue seleccionado.

Los sensores de freno indican cuando el usuario desea detenerse, la activación de estos permite al microcontrolador dejar de enviar alimentación al motor y desactiva al acelerador, conjuntamente se pueden implementar los frenos mecánicos para que el frenado sea confiable.

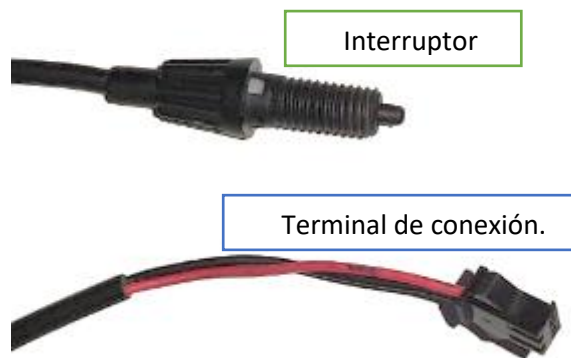


Fig.2. 8. Interruptor de freno
Fuente: Autor

- **Interruptor:** situado en la maneta del freno mecánico.
- **Terminal de conexión:** +5V(rojo), Vout(negro).

El sensor seleccionado es comúnmente utilizado en bicicletas y scooter eléctricos que se pueden encontrar en el mercado local y son de fácil instalación, consta de un interruptor que va situado en la manija de los frenos mecánicos.

Su funcionamiento se asemeja a un pulsador; envía un valor alto al microcontrolador al momento de frenado indicando el frenado del vehículo y un valor bajo si no se aprieta la manija del freno.

2.5.5. Módulo DAC MCP4725

Debido a que el controlador no soporta regulación de velocidad mediante PWM, se implementó un convertor digital analógico que es el encargado de enviar la referencia de velocidad. Se comunica mediante el bus I^2C (inter integrated circuits). Se muestra en la Figura 2.9.



Fig.2. 9. módulo DAC MCP4725
Fuente: (naylampmechatronics, 2021)

El chip integrado es capaz de operar a 3.3V o 5V, el rango de salida dependerá de la alimentación del módulo.




Características:

- Voltaje de alimentación: 2.7V a 5.5V
- Resolución de 12-bit (1.22mV si VCC=5V)
- Interfaz I2C (Standard, Fast, y High-Speed soportados)
- Dirección I2C: 0x62 o 0x63.
- Pequeño empaque
- EEPROM interna para almacenar configuraciones

2.5.6. Interfaz de usuario.

La selección de pantalla se realizó mediante la comparación entre los distintos sistemas HMI disponibles en el mercado de acuerdo con la funcionalidad requerida por el scooter y microcontrolador.

TABLA 2. 3.
COMPARACIÓN DE INTERFACES HMI

Características	Nextion	Longrunner	EATONMXV300
Imagen referencial			
Software de programación	1 (Nextion Editor)	0 (Python)	1 (Galileo 10)
Grado de protección	0 (No posee)	1 (Carcasa de plástico)	1 (Carcasa de plástico)
Número de puertos de conexión	1 (4, comunicación serial)	0 (HDMI)	0.5 (Puerto Ethernet)
Alimentación	1 5V	1 5V	0 24V
Variedad de tamaño	1 (2 inch-10inch)	1 (5inch-10inch)	1 (5inch-15inch)
Compatibilidad con microcontroladores	1 (compatible con varios microcontroladores)	0 (Funciona con miniordenador, computador)	0 (Funciona con PLC)
Total	5	3	3.5

Fuente: Tomado de (eaton, 2021), (Mercadolibre, 2021), (amazon, 2021)

De acuerdo con la Tabla 2.8., el fabricante Nextion tiene características adecuadas para la implementación en el scooter debido a su software de programación y su compatibilidad con los microcontroladores.

Se utilizó una pantalla Nextion de 5" que proporciona al conductor una interfaz de control y visualización de los parámetros del scooter. Debido a su tamaño se pueden visualizar correctamente la velocidad, estado de los sensores y se puede interactuar adecuadamente en la activación de luces y ajuste de velocidad del vehículo, se muestra en la Figura 2.10.



Fig.2. 10 Pantalla touch Nextion de 5 pulgadas

Fuente: Autor

- **Terminales de conexión:** +5V(rojo), GND(negro),Rx(amarillo),Tx(azul).

Características:

- Alimentación: 5 Vcc 410mA
- Resolución de 800 * 480 pixeles
- Compatible con Raspberry Pi, Arduino, etc.
- Pantalla LCD TFT con panel táctil resistivo integrado
- Fácil de conectar con interfaz serial de 4 hilos TTL
- Actualización de firmware mediante tarjeta microSD
- Brillo de la pantalla ajustable

2.5.7. Sistema de iluminación

Es necesario que el vehículo tenga luces para su conducción al llegar la noche, así el conductor podrá observar los distintos obstáculos en las calles, además permite que el vehículo sea visible ante los vehículos convencionales, evitando así en arrollamiento del usuario del scooter. Se implementan luces led por su bajo consumo, se seleccionan las luces que se muestran en la Figura 2. 11..



Fig.2. 11. Luces delanteras tipo led

Fuente: Autor

Características:

- Carcasa de Aluminio
- Voltaje: 12 y 24 V.
- Potencia:12W

Para la parte trasera del scooter es necesario utilizar luces led de color rojo, además deben contar con luces de frenado y direccionales por seguridad, mostrada en la Figura 2.12

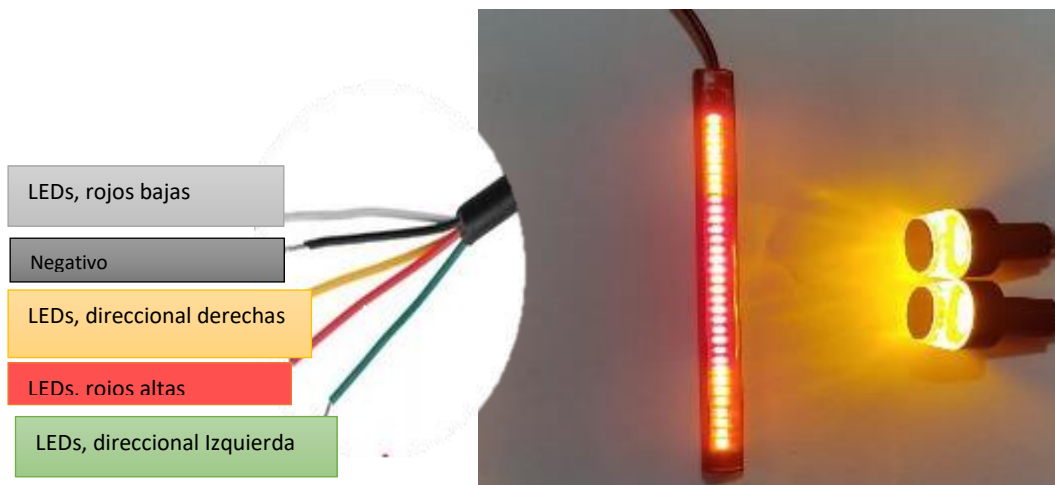


Fig.2. 12. Foco trasero de siete leds con indicador de freno

Fuente: Autor

Características:

- Material: tira de goma suave
- Color de la luz LED: 32 rojos para freno y parada, 16 ámbar para señal de giro.
- Número total de LEDs: 48 SMD(Tecnología de montaje superficial)
- Voltaje: 12V.
- Resistente al agua, con cubierta de goma de PVC duradera que se puede utilizar en días lluviosos.
- Potencia: 10W

2.5.8. Plataforma de desarrollo

Para la selección del microcontrolador es necesario conocer el número de pines a utilizar y los tipos de entradas y salidas necesario para conexión con dispositivos antes mencionados, se muestra en la Tabla 2.4., el tipo de entrada necesarios. Además, se debe considerar la velocidad de procesamiento, que determinara la velocidad de reacción del sistema.

TABLA 2. 4.
ENTRADAS Y SALIDAS DEL MICROCONTROLADOR

Descripción	Tipo
Acelerador.	Entrada Analógica
Sensor de voltaje.	Entrada Analógica
Señal frenos	Entrada Digital
Señal de referencia de aceleración	Salida Analógica
Luz Delantera	Salida Digital
Luz Trasera	Salida Digital
Direccional Derecha	Salida Digital
Direccional Izquierda	Salida Digital
Sensor de velocidad	Entrada analógica

Fuente: Autor.

De acuerdo con la anterior tabla se debe considerar microcontroladores que posean más de 2 entradas analógicas, 1 salida analógica, 4 salidas digitales y 3 entradas digitales. Además, debe contar con puertos de comunicación serial e I2C necesarios para lectura de sensores y la comunicación con la pantalla.

Los microcontroladores que cuentan con los tipos de E/S necesarios y tienen gran velocidad de procesamiento, los más comunes son: Arduino MEGA, STM32 NUCLEO-F446RE, Raspberry Pico que se compara en la Tabla 2.5.

TABLA 2. 5.
COMPARACIÓN ENTRE PLATAFORMAS Y MICROCONTROLADORES

Características	Arduino MEGA	STM32F411RE	Raspberry PI Pico
Complejidad programación (disposición de librerías y soporte en foros)	1 (IDE)	1 (STM32 ODE)	0.5 (SDK (Software Development Kit))
Velocidad de procesamiento	0.5 16 MHz	1 180 MHz	1 133 MHz
Precio	1 (18\$-22\$)	0.5 (20\$-25\$)	1 (15\$-17\$)
Disponibilidad en el mercado local	1	0	1
Puertos	0.5 54 GPIO 2 I2C 12 ADC 12-bits	1 50 GPIO 2 I2C 3 ADC 12-bits 1 DAC 12-bits	0.5 26 GPIO 2 I2C, 3 ADC 12- bits
Memoria flash	1 512 KB	1 512 KB	1 2 MB
Voltaje de alimentación	1 (de~7-12V)	1 (3.3 V, 5 V, 7 - 12 V)	0 (de ~1.8- 5.5 V)
Voltaje de trabajo	1 5V	1 5V	0.5 3.3V
Total	6	6	5.5

GPIO (General Purpose Input Output), I2C (inter integrated circuits) ADC (convertidor analógico a digital), DAC. (convertidor digital a analógico). Fuente: Tomado de (Mercadolibre, 2021), (Arduino, 2021), (bricogeek, 2021)

La plataforma en la que se desarrolló el controlador fue Arduino MEGA por la ventajas y facilidades en comparación con las otras plataformas, sus ventajas como:

- Voltaje de alimentación que permitió la implementación con el reductor de voltaje seleccionado
- La memoria flash tiene la capacidad suficiente para almacenar la programación realizada.
- Tiene los pines adecuados para la interacción con dispositivos seleccionados.
- En caso de avería o cambio; se encuentra disponible en el mercado nacional.

Los pines que posee la tarjeta se describen en el siguiente gráfico.

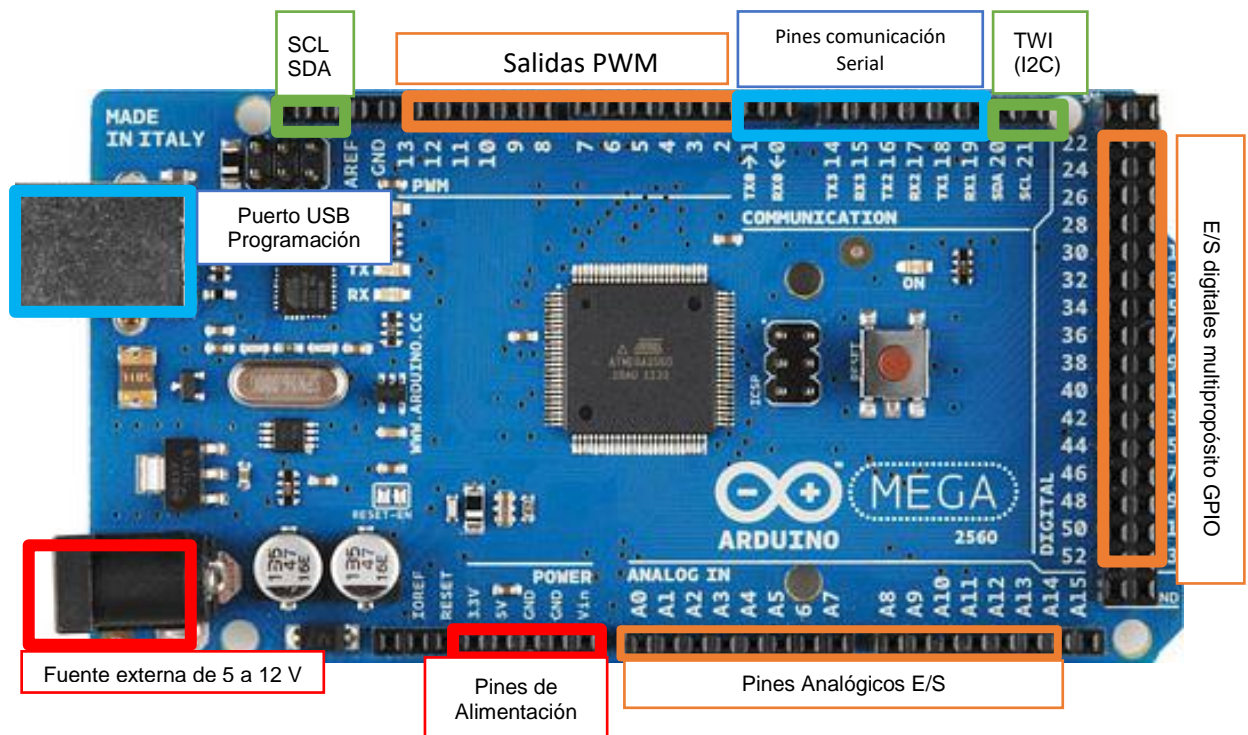


Fig.2. 13. Componentes del Arduino Due

Fuente: Autor

En la Figura 2.13., se puede observar los pines donde van conectados los dispositivos mencionados en este apartado.

- **Pines de Alimentación:** Pines que proporcionan salidas de tensiones a distintos niveles.
- **Pines digitales:** Denominados GPIO (General Purpose Input/Output), pueden ser utilizados como entradas y salidas digitales, con una tensión de 5V y una corriente máxima de 40mA.
- **Pines Analógicos:** son capaces de leer señales analógicas cuyos valores se encuentran entre 0V y 5V.
- **Fuente externa:** permite alimentar el Arduino mediante una fuente externa con voltajes de 5V a 12V
- **Puerto USB:** Permite carga el programa de control hacia el microcontrolador.
- **TWI:** System Data (SDA) y System Clock (SCL). TWI soporte de comunicación.
- **Comunicación Serie:** Protocolo utilizado con el HMI (RTX0 y TX0).

2.5.9. Convertidor DC-DC Buck

El convertidor de energía permite tener en sus pines de salida un voltaje de 12V y una corriente máxima de 10 A, permite alimentar los dispositivos electrónicos como: Microcontrolador, Pantalla touch, los sensores de frenos y acelerador, sistema de iluminación, entre otros.

Con la utilización de la ecuación 1 se determina que la potencia requerida por el microcontrolador, pantalla y el sistema de iluminación, no debe supere la potencia del convertidor DC-DC BUCK. La corriente de cada elemento se especifica en la Tabla 2.6.

TABLA 2. 6.
CORRIENTE DE FUNCIONAMIENTO DE LOS DISPOSITIVOS.

Dispositivo	I[mA]
Arduino MEGA 2560	500
Pantalla Nextion 5"	410
Luces delanteras	1000
Luces traseras	800
Claxon	1000
Módulo L298D	72
Total	3,7A

Fuente: Autor

$$W_c = 12V * 3,7A$$

$$W_c = 44,4W$$

La potencia requerida por el controlador es de 44,4W, a continuación, se determina la potencia disipada por el convertidor con la ecuación 2 y 3.

$$I_{in} = \frac{44,4W}{67V} = 0,66A$$

$$W_{buck} = (67 - 12V) * 0,66 = 33,6W$$

El convertidor utilizado suministrar 100W, se muestra en la Figura.2.14, son comercializados para el uso en scooter eléctricos. La potencia del controlador es menor a la suministrada por el convertidor, es ideal para el uso en el scooter.



Fig.2. 14. . Convertidor DC-DC Buck
Fuente: Autor

Características:

- Tipo: Módulo reductor no aislado DC-DC
- Voltaje de entrada: (36-72V)
- Voltaje de salida: 12V
- Corriente de salida: 10 A
- Potencia de salida: 100W.

2.6. Resultados

2.6.1. Diseño del algoritmo de control

Para realizar el control propuesto (PID), es necesario tener las señales Sp: punto de ajuste deseado, Pv: variable de medida del proceso(rpm), y Cv: variable controlada como se muestra en la Figura 2.15.

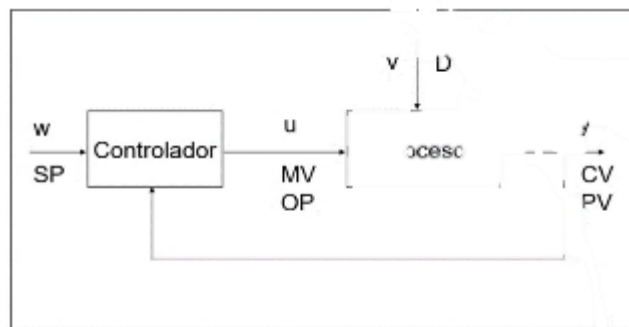


Fig.2. 15. Diagrama de bloque
Fuente (De la Fuente, 2017)

El código desarrollado para la lectura del Sp entregada por el acelerador se explica en la Figura 2.16.

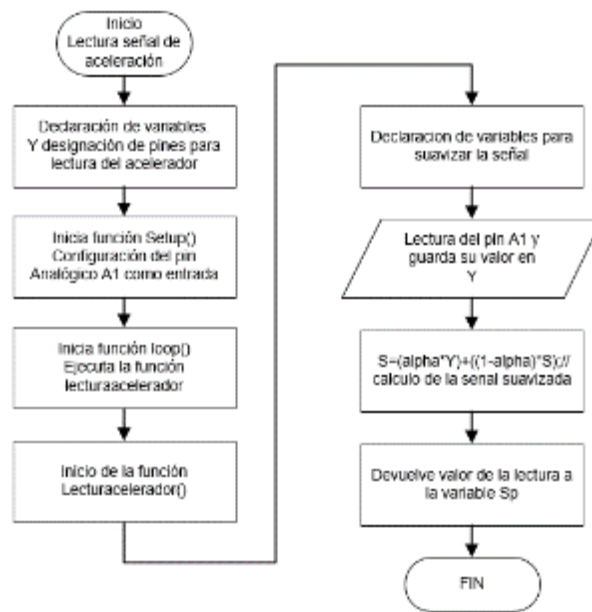


Fig.2. 16. Lectura de los valores del acelerador y suavizado de señal

Fuente: Autor

La señal de referencia o setpoint queda guardada en la variable Sp del controlador, la lectura del acelerador se realiza a través del puerto analógico A1 y con el fin de atenuar la señal se utiliza la media móvil exponencial con la ecuación 5

De igual forma se desarrolló el código para la determinación de la variable Pv, determina los rpm del proceso, la explicación del funcionamiento se explica en la Figura 2.17.

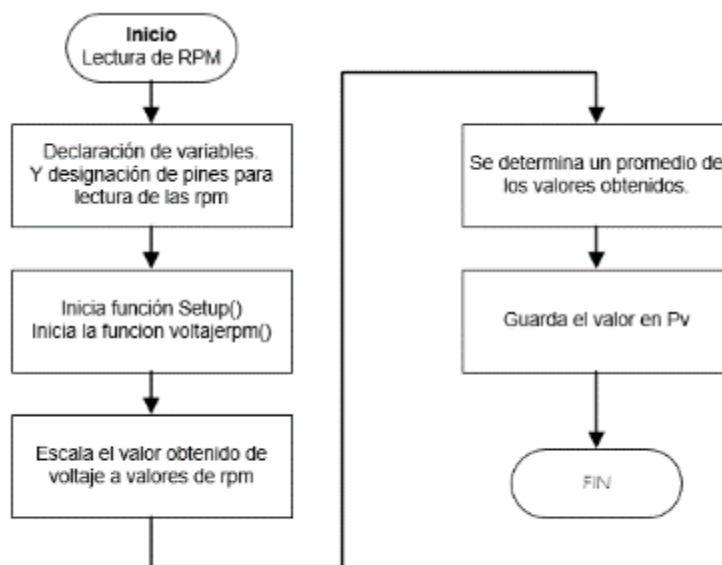


Fig.2. 17. Lectura del rpm del motor

Fuente: Autor

Aprovechando que el controlador tiene un terminal indicador de velocidad, aplicando un divisor de voltaje es posible dar lectura al voltaje que varía en función de la velocidad de esta terminal, para seguidamente escalar los valores a rpm que proporciona el motor.

Para determinar la señal Cv, primero se calculó mediante el algoritmo PID que es la suma de los termino P, I, D. La señal de salida es un voltaje análogo mediante el DAC siendo la variable que controla la velocidad del motor. El algoritmo PID se explica en la Figura 2.18., mediante un diagrama de flujo.

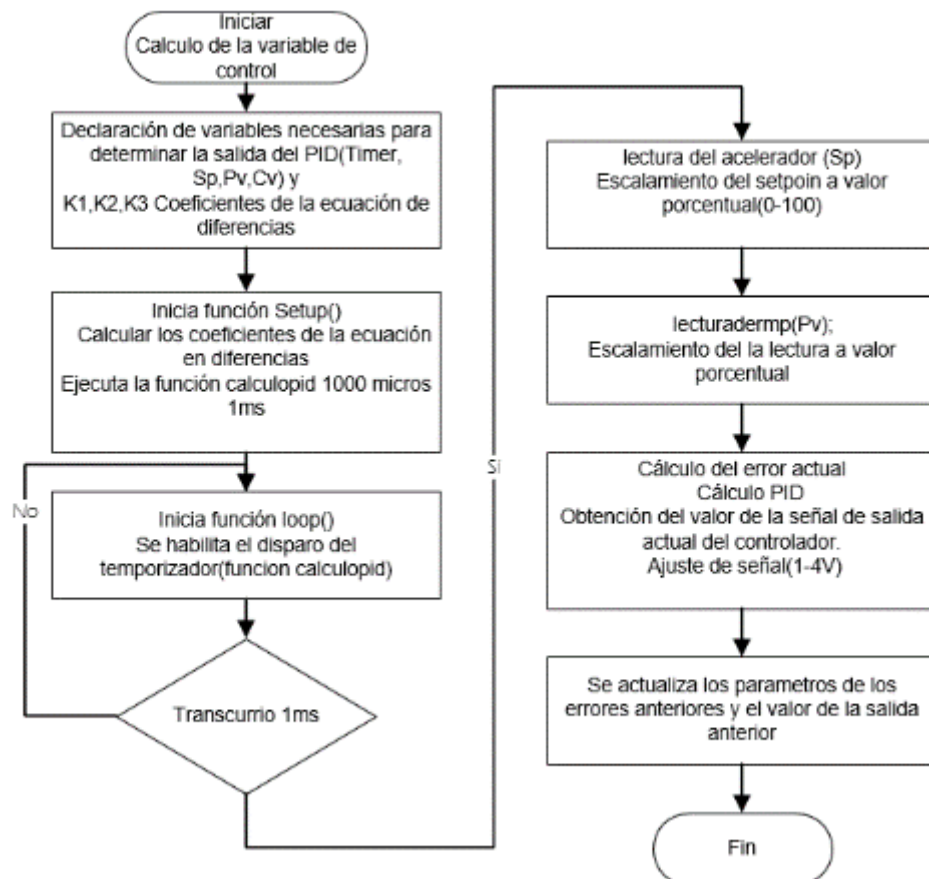


Fig.2. 18. Determinación de la variable de control mediante el algoritmo PID

Fuente: Autor

Como muestra la Figura 2.18 el algoritmo del control PID se ejecuta cada 1milisegundo, en donde se ejecuta la lectura de la señal de referencia Sp y de la señal de retroalimentación Pv para luego determinar la salida del controlador mediante la variable Cv. Se calcula el error actual y anterior para el siguiente proceso.

Se determina la salida del controlador con la aplicación de la ecuación 11. El resultado obtenido se limita entre los valores 800,3150 que corresponde a 1-4 [V], para su posterior salida mediante el DAC hacia la señal de referencia de velocidad del controlador.

Para finalizar, se actualizan los valores de error actual, error anterior, error dos veces anterior y los valores de salida actual y anterior.

Para determinar los valores de k_p , k_i , k_d , es necesario un modelo de la planta para obtener los valores de las ganancias, para luego implementar en el microcontrolador. Se lo realiza de la siguiente manera.

2.6.2. Identificación del sistema.

Para implementar un controlador PID digital es necesario el entender funcionamiento del proceso a controlar, por tal razón se necesita la respuesta de la planta al aplicarle un entrada escalón, este procedimiento se conoce como identificación de sistema por medio de la curva de reacción y su salida se representa con una función de transferencia.

Para obtener la función de transferencia en lazo abierto es necesario aplicar en la entrada una señal de tipo escalón con valores de 30% y 40% del valor máximo, en este caso es voltaje análogo aplicado al controlador por el DAC, con valores de 800 como el 0% y 3150 como el 100%. Con el 35% de la variable de control se espera que se establezca la velocidad del scooter(P_v).

La adquisición de datos se realizó con el software PLX-DAQ, que guarda los valores obtenidos en una tabla en Excel, para su posterior análisis, el tiempo de muestreo se definió en 0.01 segundo durante un periodo de 5 segundos, la respuesta resultante se muestra en la Figura 2.19.

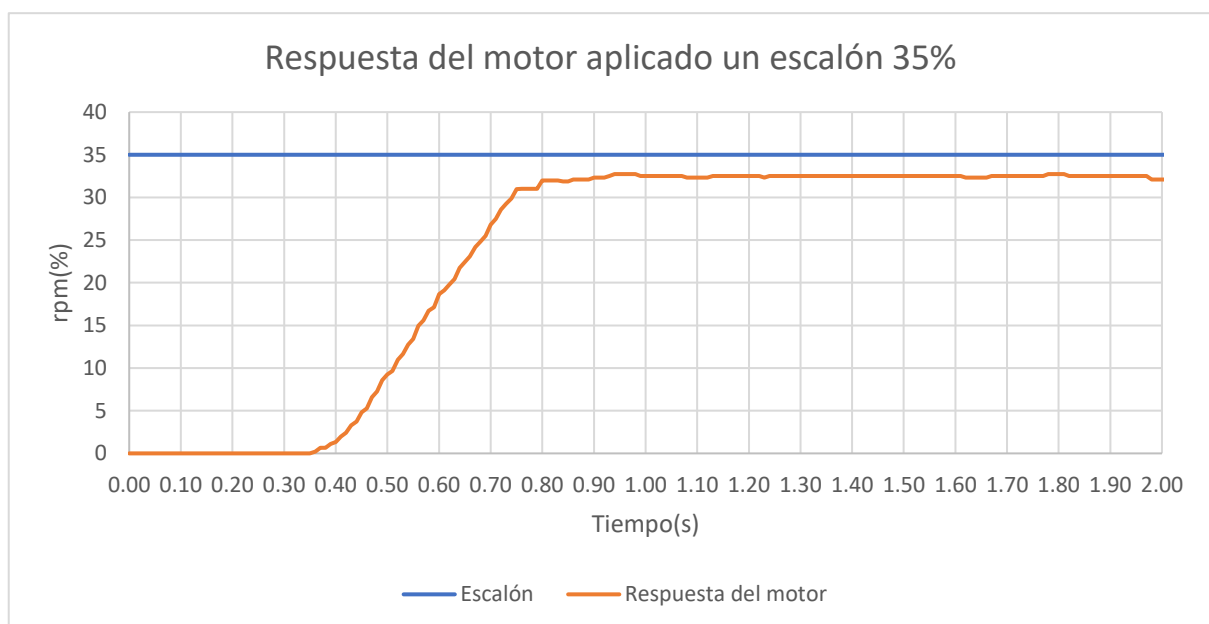


Fig.2. 19. Respuesta en lazo abierto

Fuente: Autor

El escalón aplicado se representa de color azul, y la respuesta del motor de color naranja, y se puede observar un error de estado estable el cual ser corregido por el controlador. Se deduce que es un sistema de primer con retardo, debido a que se tiene un comportamiento exponencial como se muestra en la Figura 2.19.

Para determinar la función de transferencia se obtuvo mediante dos métodos: análisis de la curva de reacción y con la utilización de la función “sytemIdentificacion” del software Matlab

- Obtención de la función de transferencia mediante el análisis de la curva de reacción

Con la respuesta obtenida de la planta se obtiene los parámetros de la función de transferencia, deduciendo que es un sistema de primer orden con retardo.

El parámetro K se obtiene utilizando la ecuación 13, que es el valor máximo de la salida del sistema entre el valor máximo de entrada del sistema.

$$K = \frac{33}{35} = 0.9428$$

El parámetro de retardo L , se determina analíticamente de la gráfica, es el tiempo que se demora en reaccionar el sistema.

$$L = 0.35$$

La constante de tiempo τ , se obtiene analizando la curva de reacción de reacción del sistema y se determina el tiempo en que la variable de salida no cambia y se resta el retraso

$$\tau = 0.8 - 0.35 = 0.2844$$

Los parámetros obtenidos se reemplazan en la ecuación 12, obteniendo la función de transferencia de la planta

$$G(s) = \frac{0.9428e^{-0.35s}}{0.2844s + 1}$$

Con el fin de comprobar el resultado obtenido se realiza la estimación de modelo matemático mediante la utilización de software Matlab que se explica a continuación.

- Obtención de la función de transferencia mediante la función “sytemIdentificacion”

Los datos obtenidos de entrada y salida del sistema se ingresan en el Workspace para determinar el modelo matemático que describe el proceso. Se ejecuta la función “sytemIdentificacion” en donde se importó los datos en el dominio del tiempo como se muestra en la Figura 2.20.

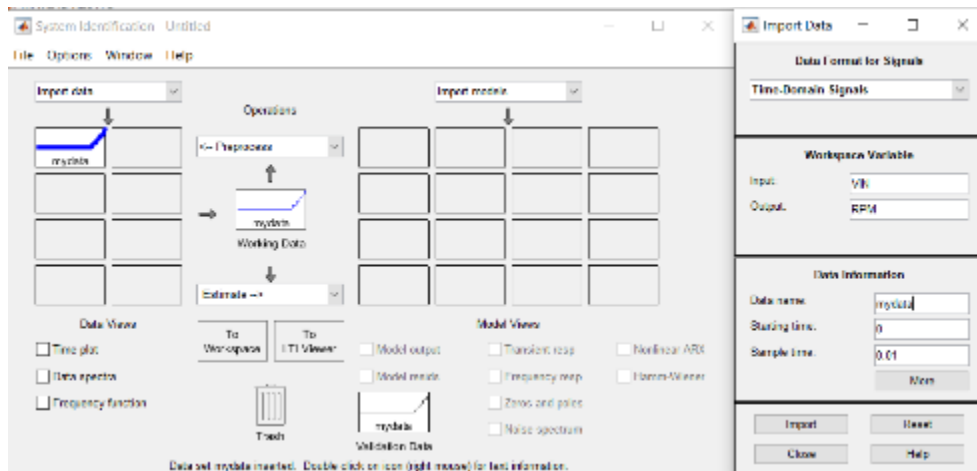


Fig.2. 20. Ventana de identificación de sistema con los datos obtenidos de la respuesta de la planta

Fuente: Autor

Posteriormente, se selecciona Process Models sabiendo que se desea aproximar a un sistema de primer orden más tiempo muerto, y presionar el botón “Estimate” como se indica en la Figura 2.21.

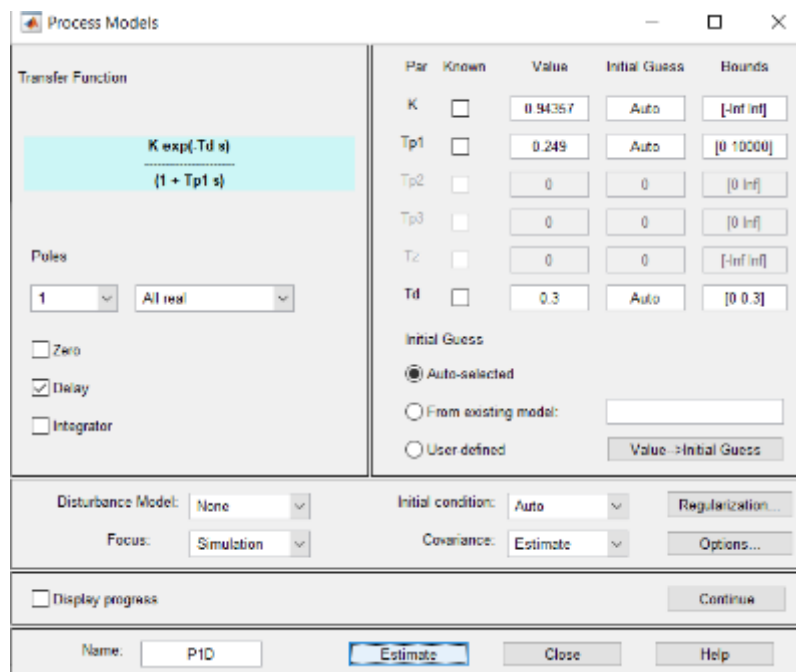


Fig.2. 21. Estimación de modelo de proceso

Fuente: Autor

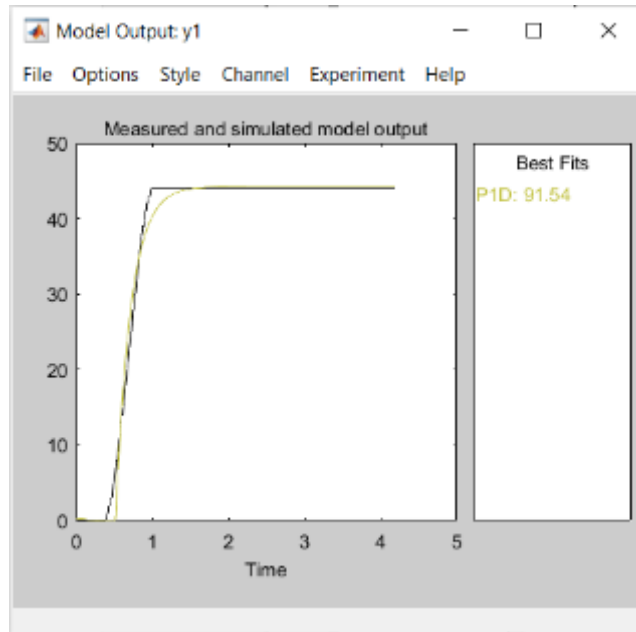


Fig.2. 22. Modelo Matemático estimado

Fuente: Autor

Finalmente, se habilita la opción " Model output" para visualizar la respuesta estima del modelo matemático en la siguiente figura se muestra un porcentaje de aceptación de 91% El modelo dinámico estimado se muestra en la Figura 2.23.

$$G(s) = \frac{Kp}{1+Tp1*s} * \exp(-Td*s)$$

$$Kp = 0.94357$$

$$Tp1 = 0.249$$

$$Td = 0.3$$

Fig.2. 23. Modelo matemático estimado

Fuente: Autor

Para transformar el modelo dinámico a modo de función de trasferencia se utiliza la instrucción tf, obteniendo:

G =

```

From input "u1" to output "y1":
      0.9436
exp(-0.3*s) * -----
              0.249 s + 1

```

Fig.2. 24. Función de trasferencia de la planta

Fuente: Autor

Las funciones de transferencias obtenidas son similares, que no presenta variaciones significativas en sus parámetros, por lo que se puede utilizar para el diseño del controlador que se explica en el siguiente apartado

2.6.3. Diseño del controlador.

Como muestra la Figura 2.18., la respuesta presenta un error de estado estable que se corrige con un control integral. Además, se define los parámetros de diseño con un tiempo de establecimiento de 4 segundos.

En el presente trabajo se aplicó la técnica de modificación de LGR(lugar geométrico de las raíces) para asignar la posición de los polos del sistema que cumplan con las especificaciones antes mencionadas, utilizando la herramienta “rltool” de Matlab que se explica a continuación.

Se inicia “rltool” y se importa la función de transferencia obtenida en la arquitectura de control que se muestra en la Figura 2.25.

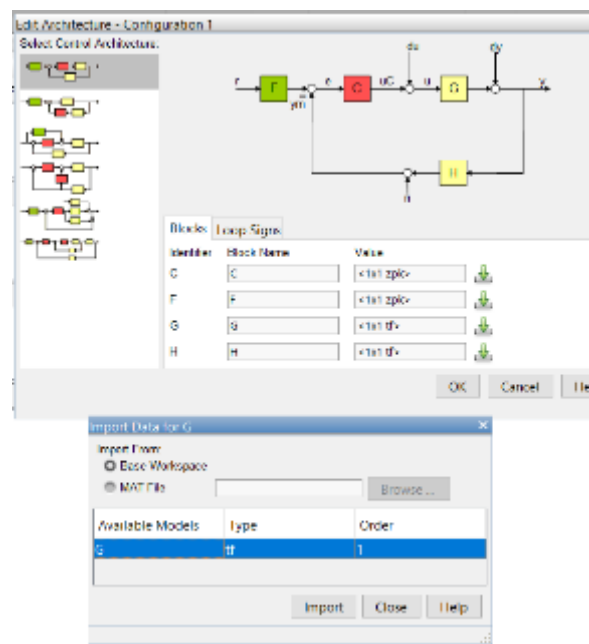


Fig.2. 25.Importación de la función de transferencia.

Fuente: Autor

Se introducen las especificaciones requeridas de la respuesta temporal como el tiempo de establecimiento y la sobre oscilación y se agrega un integrador y un zero real para las acciones de control, por lo tanto, el controlador agregado es un PI, como se muestra en la Figura 2.26.

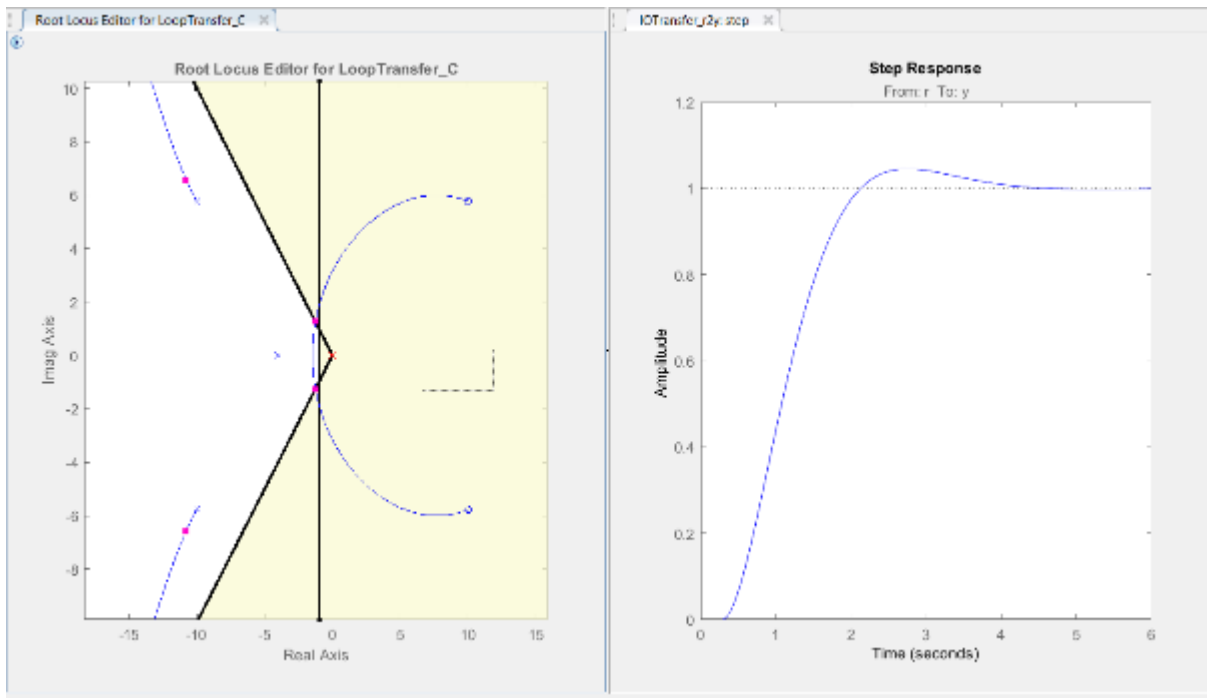


Fig.2. 26. Modificación DE LGR

Fuente: Autor

Finalmente se exporta el modelo del controlador al Workspace y se extraen los valores de k_p , k_i , k_d , con el comando “[k_p, k_i, k_d] = piddata(C). Los valores obtenidos se muestran en la Figura 2.27.

$$\begin{aligned}
 K_p &= \\
 &0.0124 \\
 \\
 K_i &= \\
 &0.9964 \\
 \\
 K_d &= \\
 &0
 \end{aligned}$$

Fig.2. 27. Parámetros del controlador.

Fuente: Autor.

Los valores obtenidos se introducen en el algoritmo de control del Arduino que se explica en la siguiente sección.

2.6.4. Diseño de la interfaz gráfica.

El funcionamiento general del sistema y de la pantalla se explica en el siguiente diagrama de flujo.

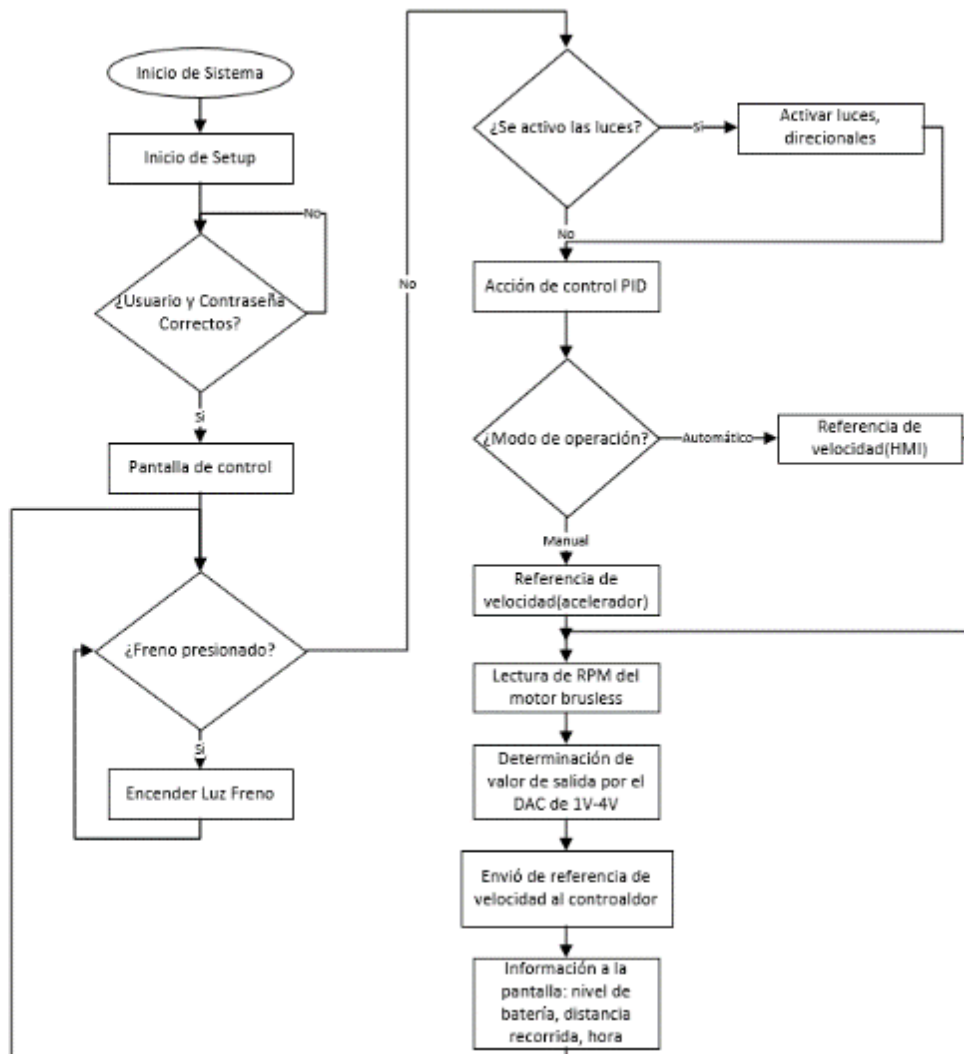


Fig.2. 28. Diagrama de flujo de la utilización de la pantalla con el microcontrolador

Fuente: Autor

Para la utilización de la pantalla touch se utilizó la librería “Nextion.h”, la comunicación con el microcontrolador se da mediante los pines RX1 Y TX1, en la siguiente Tabla 2.7. se indica los elementos de la pantalla y tipo.

TABLA 2. 7.
ELEMENTOS DE LA PANTALLA

Objetivo	Nombre del objeto	Tipo	PAGEID	ID
Indicador de nivel de batería	n2	Number	1	21
Activación de direccional izquierda	btiz	Dual-state button	1	7
Activación de direccional derecha	btderecha	Dual-state button	1	8
Activación de luces	luces	Dual-state button	1	9
Indicador de frenado	bt1	Button	1	19
Cambio de referencia de acelerador a pantalla	btset	Dual-state button	1	10
Referencia de velocidad	h0	Slider	1	18
Bajar referencia	bajarve	Button	1	5
Indica el valor de la referencia de velocidad	n1	Number	1	12
Indicador de máxima	n3	Number	1	17
Total de distancia recorrida	n4	Number	1	18
Velocímetro	n0	Number	1	13
Indicador de hora	hora	Number	1	16
Indicador de minutos	minuto	Number	1	15

Fuente: Autor

Para que el microcontrolador identifique el elemento es necesario asignar el ID y la pagina donde se encuentra. En la Figura 2.31., se muestra la interfaz HMI, donde se encuentra los elementos descritos en la Tabla 2.7.

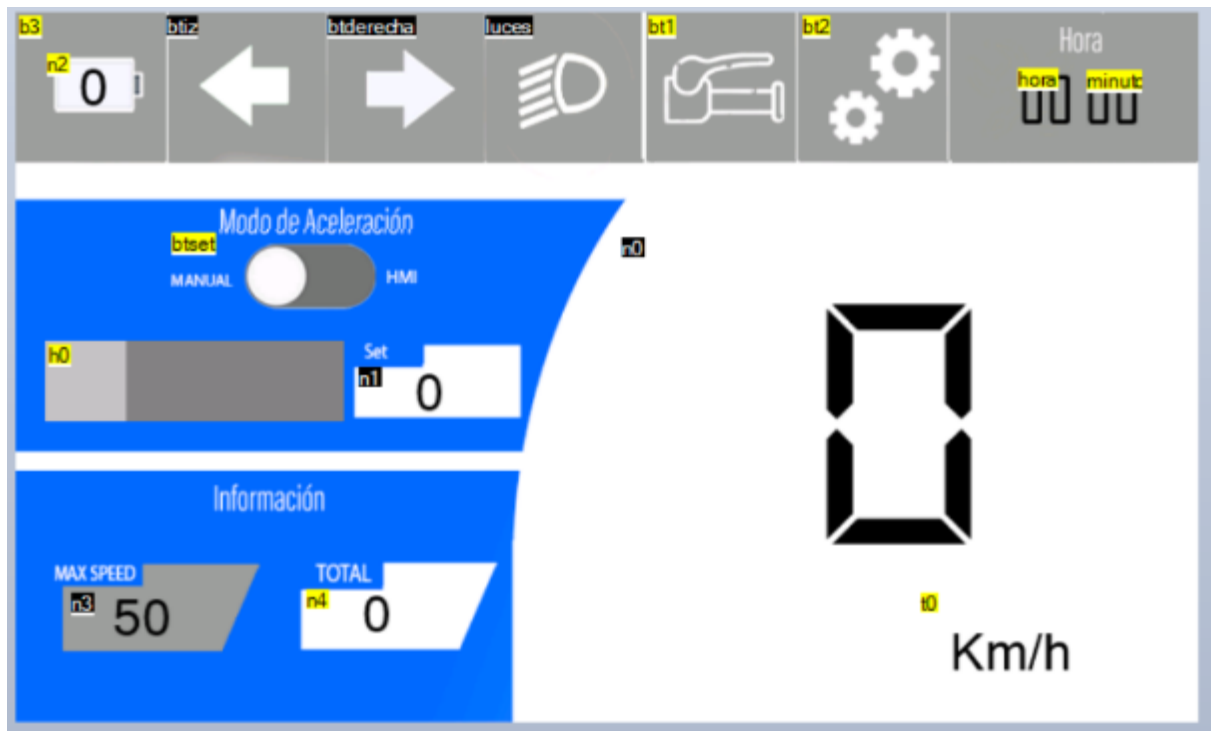


Fig.2. 29. Interfaz HMI

Fuente: Autor

La velocidad de comunicación entre la pantalla y la tarjeta Arduino es de bauds=115200, es la máxima velocidad permitida por la pantalla del fabricante Nextion mediante la comunicación Serial.

La programación realizada se encuentra en el anexo E, donde se encuentran todas las librerías e instrucciones utilizadas para la elaboración del código de control.

CAPÍTULO 3

Implementación y pruebas de funcionamiento.

3.1. Introducción

En esta sección del documento se detalla la implementación del sistema de control del scooter con los componentes seleccionados en el capítulo anterior, también se menciona la conexión entre los dispositivos electrónicos. Además, se muestran las pruebas de funcionamiento del sistema de control y del HMI. Finalmente se muestra las pruebas de distancia recorrida por el scooter.

3.2. Metodología

Se siguieron los siguientes pasos para realizar la implantación del sistema.

- Se distribuyó los dispositivos electrónicos; módulo de reloj, DAC, luces, pantalla, etc., en la estructura del scooter.
- Conexión entre los dispositivos y asignación de los puertos de la tarjeta Arduino.
- Pruebas de comunicación entre el Arduino y la pantalla y la activación del sistema de iluminación y control de velocidad.
- Experimentación con la planta (motor DC brushless), verificando si su aceleración es paulatina y su arranque es suave. Y la comprobación de la interfaz gráfica en donde se puede interactuar directamente con el setpoint de la velocidad deseada, como la activación o desactivación de los dispositivos del scooter.

3.3. Montaje del sistema de control con los sensores y actuadores en la estructura del scooter

Los dispositivos electrónicos se distribuyen por la estructura del scooter como: luces, sensor de freno, la tarjeta Arduino, convertidores de energía se encuentran en el gabinete y la pantalla se encuentran en el manubrio del vehículo, como se muestra en la Figura 3.1.

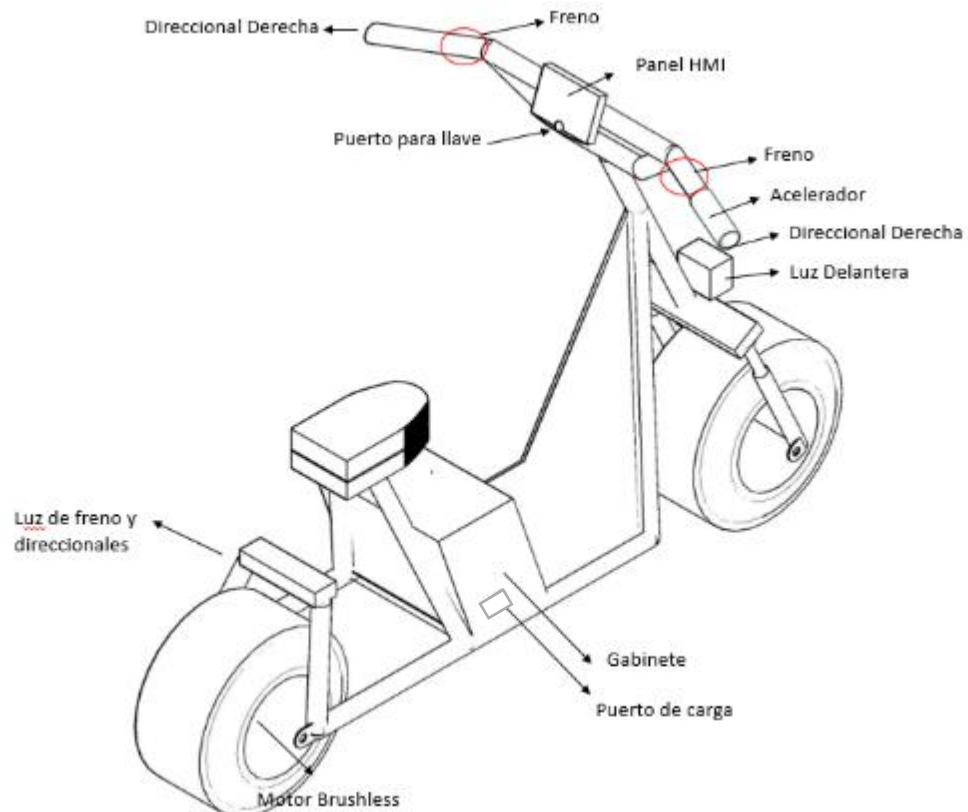


Fig.3. 1 Distribución de los dispositivos electrónicos.

Fuente: Autor

En el gabinete se encuentran la batería y el controlador del motor, así como el sensor de voltaje, también se muestran la colocación de las luces tanto delanteras, traseras y direccionales, en el manubrio se encuentra el interruptor del pito y el acelerador.

Los elementos que conforman el sistema de control se encuentran situados en una base que evita el contacto con el controlador. Los dispositivos electrónicos se encuentran cerca de la tarjeta para evitar largos cableados, se pueden desmontar cada elemento en caso de requerir reemplazarlos.

3.3.1. Montaje de la tarjeta Arduino Mega 2560 y los dispositivos de control.

Está montado en una base junto a módulo de reloj ds3231 y DAC MCP4725, barra general de 5V y GND, realizado para la conexión de los diferentes dispositivos que funcionan a ese nivel de voltaje. El montaje se muestra en la Figura 3.2.

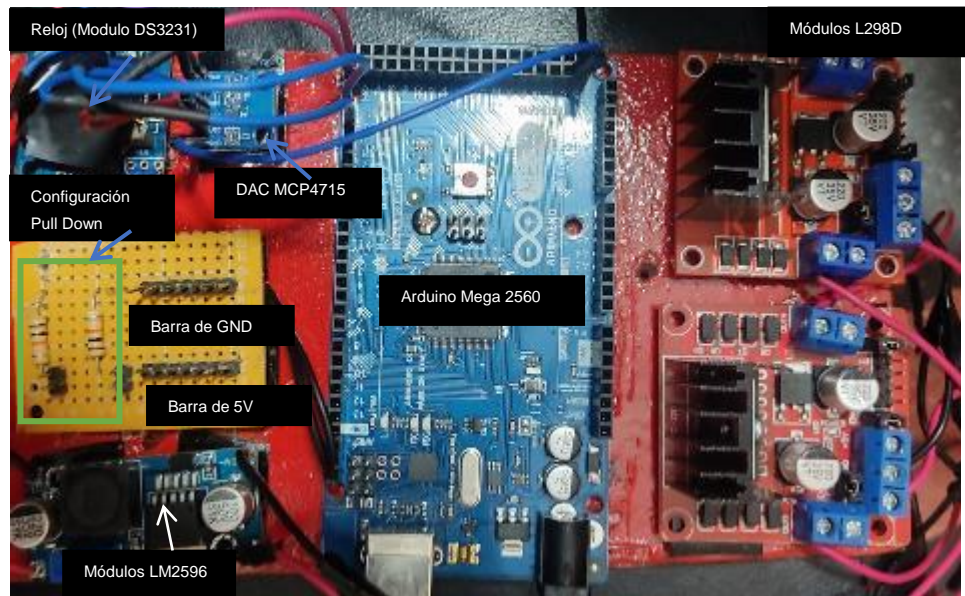


Fig.3. 2 Distribución de los elementos del sistema de control

Fuente: Autor

Los elementos mostrados en la figura 3.2. son dispositivos utilizados para realizar el control de velocidad y activación de las luces, necesarios para el montaje del HMI como; módulo Lm2596, DAC MCP4715, Reloj del módulo DS3231, Módulo L298D, sus conexiones se explican en los siguientes apartados.

3.3.2. Conexión del DAC MCP4725, Módulo DS3231

Colocados cerca de la tarjeta Arduino, utilizan el protocolo de comunicación I2C para su comunicación con el microcontrolador a través de los pines SCL, SDA y su alimentación es de 5V. EL DAC se conecta a los terminales de regulación de velocidad del controlador ya que este regula la tensión de referencia de velocidad entre 1V-4V, su conexión con el controlador se muestra en la Figura 3.2.

3.3.3. Pantalla touch.

Colocado en el volante del scooter dentro de la carcasa plástica para su protección junto a la bocina. El diseño de la carcasa se muestra en el Anexo C

Se alimenta la pantalla mediante un reductor de voltaje LM2596 mostrado en la Figura 3.2, que puede suministrar la corriente requerida por la pantalla.

La pantalla tiene 4 pines de conexión, que se indican en la Figura 3.3.



Fig.3. 3 Terminales de conexión de la pantalla touch

Fuente: Autor

La conexión de los dispositivos antes mencionados se detalla en la Tabla 3.1.

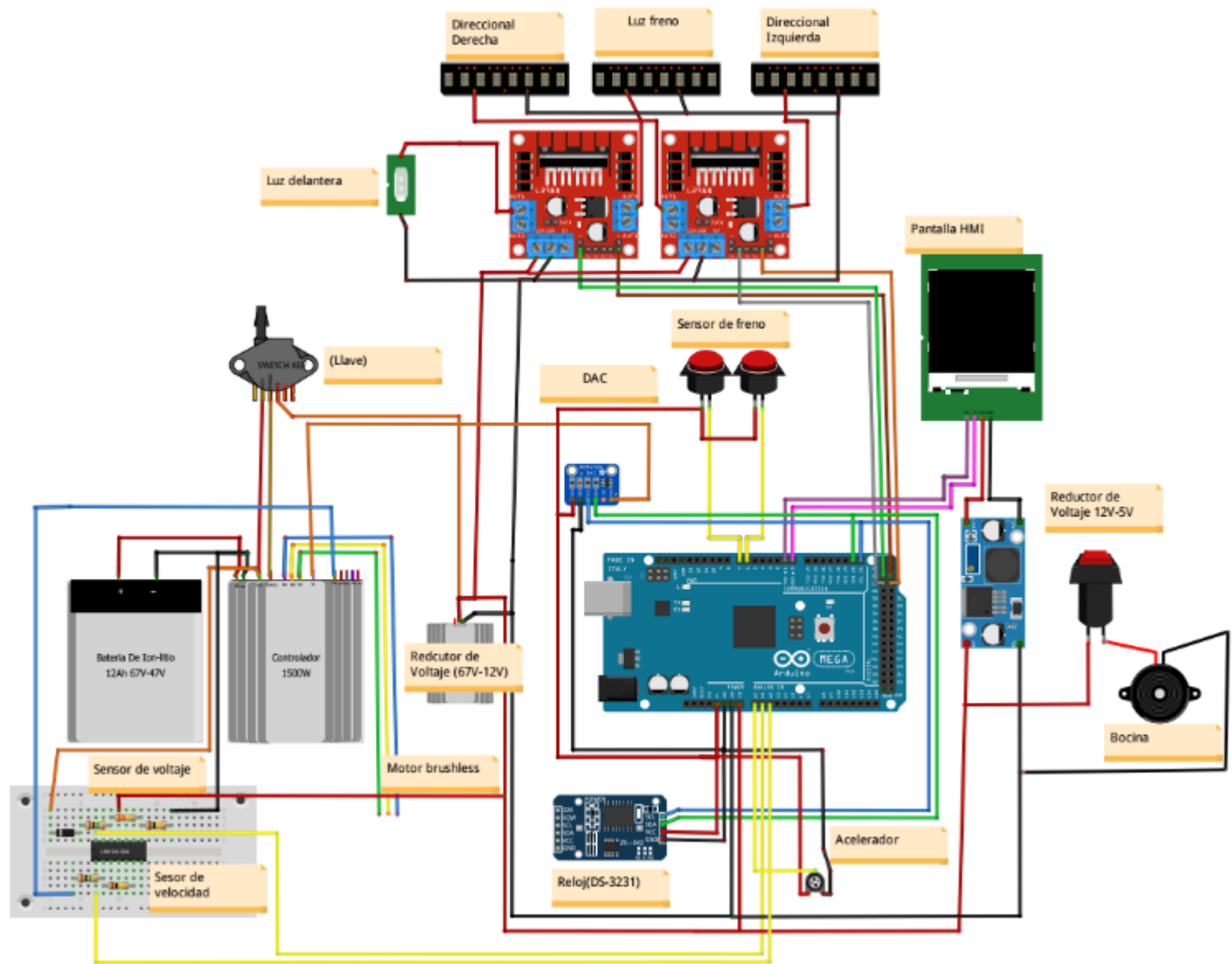
TABLA 3. 1.
CONEXIÓN DE PINES DEL ARDUINO

Arduino MEGA 2560	Conexiones
+5V	+5V (sensor de pulsos, freno acelerador, módulos ds3231, DAC MCP4715,.)
GND	GND (sensor de pulsos, freno acelerador, módulos ds3231, DAC MCP4715,.)
PIN A0	Vout(Velocidad del motor)
PIN A1	Vout (Acelerador)
PIN A2	Vout (sensor de voltaje)
PIN DIGITAL 6,7	Vout (sensor de freno)
PIN DIGITAL 24	IN1 (Módulo L298D (1))
PIN DIGITAL 25	IN4 (Módulo L298D (1))
PIN DIGITAL 26	IN1 (Módulo L298D (2))
PIN DIGITAL 27	IN4 (Módulo L298D (2))
RX(0)	TX (panel touch)
TX(0)	RX (panel touch)

Fuente: Autor

En la anterior tabla se detalla los pines que están configuradas tanto como entradas y salidas del microcontrolador.

El diagrama eléctrico general se muestra en la Figura 3.4., donde se muestra todos los elementos conectados del sistema de control.



fritzing

Fig.3. 4 Esquema de conexión de los dispositivos electrónicos del sistema de control del scooter eléctrico

Fuente: Autor

3.4. Validación de las mediciones

Para realizar el control del sistema, además, se muestre una correcta información en la pantalla es necesario validar las mediciones realizadas por los sensores implementados, a continuación, se muestra la verificación de los valores medidos por instrumentos de medición y los valores obtenidos por los sensores realizados.

3.4.1. Lectura de los rpm

Los rpm del motor se obtienen mediante el pin de velocidad proporcionada por el controlador, que varía su voltaje en función del rpm del motor, el microcontrolador escala el voltaje obtenido a rpm, para su posteriormente procesarlo y obtener la distancia recorrida y la velocidad del scooter. Para la validación de la medición se utilizó un tacómetro digital "DT-2224". Los valores obtenidos se muestran en la Tabla 3.2.

TABLA 3. 2.
MEDICIONES ARDUINO Y TACOMETRO DIGITAL

Velocidad	Arduino.	Tacómetro digital	Unidad
Mínima	93	92.4	rpm
Media	206	208.7	rpm

Fuente: Autor

Se realizaron 2 mediciones, a media, mínima velocidad del motor, que se muestran en Figura 3.5.

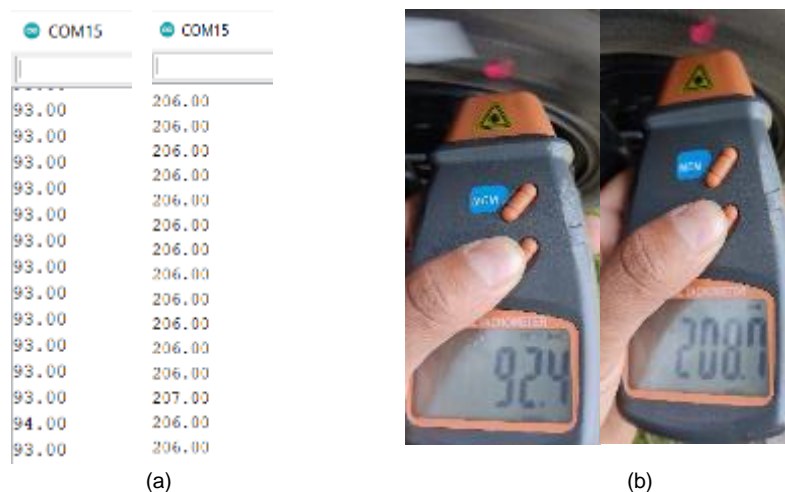


Fig.3. 5. (a) Lecturas del Arduino, (b) Lecturas del tacómetro.

Fuente: Autor

La medición realizada por el Arduino no presenta gran diferencia entre los realizados con el tacómetro, indicando que los valores obtenidos son adecuados para el uso en el sistema de control.

La velocidad máxima se determina con la ecuación 5, con los rpm que alcanza el motor, obteniendo:

$$V_{Km/h} = \frac{3}{25} * \pi * 0.29 * 455 = 50 \text{ km/h}$$

La velocidad máxima alcanzada por el vehículo es de 50 km/h cuando esté completamente cargada la batería.

3.4.2. Lectura de voltaje de batería.

Las lecturas se obtienen del sensor de voltaje descrito en el capítulo anterior, corresponde al voltaje de la batería que varía de 47V a 67V, que corresponde al nivel de batería. Para su medición se utilizó un multímetro “TRUPER MUT-39”.

TABLA 3. 3.
LECTURA DE VOLTAJE

Arduino.	Multímetro	Interfaz gráfica (%)
57,31V	57,3V	50%
66,9V	67V	100%

Fuente: Autor

Se realizaron 2 mediciones, a su carga completa, a media carga, las mediciones realizadas se muestran en la Figura 3.6.

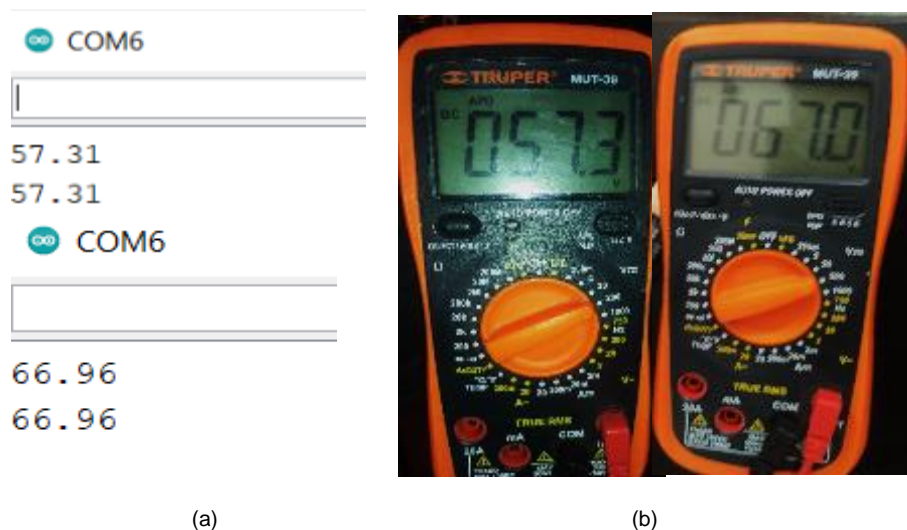


Fig.3. 6. (a) Lecturas del Arduino, (b) Lecturas multímetro

Fuente: Autor

Se compararon ambas lecturas determinando que presenta una diferencia significativa entre lectura, por lo que se usó la lectura del Arduino para representar el nivel de batería en la pantalla HMI.

3.5. Implementación del control PI

El diagrama de bloque del sistema se muestra en la Figura 3.7, donde un valor de error es calculado cuando se realiza la diferencia entre el punto de referencia y el valor medido de la velocidad, para que el microcontrolador calcule la variable de control que va hacia el motor y se controle la velocidad del vehículo.

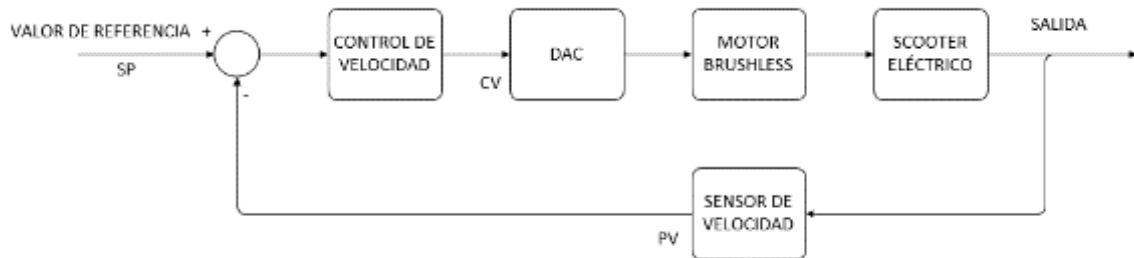


Fig.3. 7. Diagrama de bloque del sistema de control de velocidad.

Fuente: Autor

Se ingresan los valores obtenidos como k_p , k_i , al algoritmo de control PI realizado anteriormente, para luego observar la respuesta del sistema. Obteniendo la siguiente respuesta del control que se muestra en la Figura 3.8.

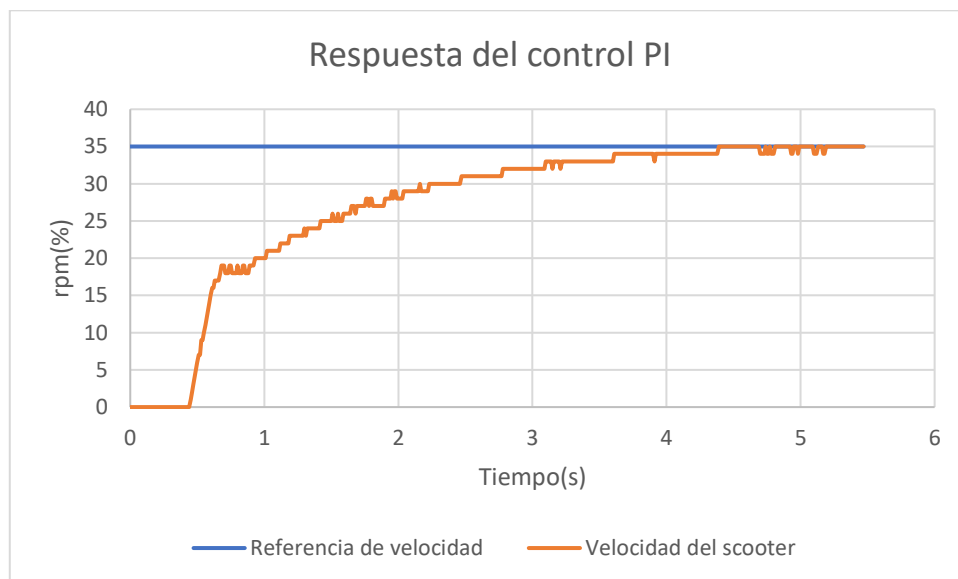


Fig.3. 8. Respuesta del control PI.

Fuente: Autor

Se observa que se cumple las especificaciones requeridas en el diseño del controlador, eliminado el error de estado estable y el tiempo de establecimiento de 4 segundos.

3.6. Pruebas de funcionamiento del sistema de control.

Para comprobar el funcionamiento del sistema de control se realizan pruebas al sistema de iluminación, frenos, indicador de velocidad, la regulación de velocidad, y visualización de los parámetros como nivel de batería, y distancia recorrida.

3.6.1. Funcionamiento del sistema de iluminación.

A continuación, se comprueba la activación y funcionamiento del HMI, se realizó la prueba de activación de luces, direccionales, luz de freno que se muestra Figura 3.7.



Fig.3. 9. (a)Funcionamiento del HMI, (b) Luces delanteras y frenos encendidos

Fuente: Autor

Como se puede observar en la Figura las luces delanteras y las luces del freno funcionan correctamente y pueden ser activadas en cualquier momento en el que se esté utilizando el vehículo.

3.6.2. Visualización de parámetros.

El nivel de batería permite conocer el estado de carga del scooter evitando así descargarlo completamente que afecta la vida útil de la misma, el HMI nos permite observar en porcentaje la carga de batería, y al momento que se encuentra por debajo del 20%, se enciende el indicador rojo que significa que el scooter debe ser cargado.

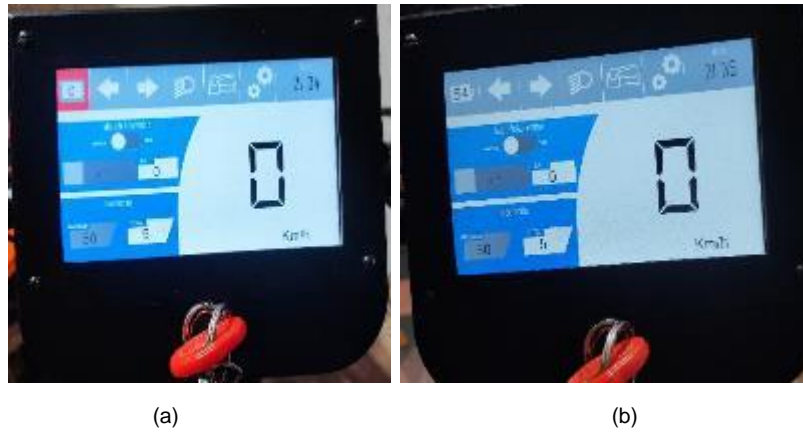


Fig.3. 10.(a) Indicador de carga de batería baja. (b) Indicador de batería a media carga.

Fuente: Autor

Como se muestra en la figura 3.8., el indicador de carga se torna de color rojo al momento de estar por debajo del 20%, y sobre pasa ese nivel tiene un color gris.

Para comprobar si la distancia recorrida mostrada en la pantalla es correcta. Se trazó varios trayectos la aplicación de Google Maps para determinar la distancia existen desde el punto de partida y llegada, comparándolas con el valor mostrado por el HMI. Los resultados obtenidos se muestran en la Tabla 3.4.

TABLA 3. 4.
VALIDACIÓN DE LAS DISTANCIA RECORRIDA

Trayecto	Interfaz gráfica	Google Maps	Unidad
1	3	3,3	km
2	5	5	km
3	4	4	km

Fuente: Autor

Se recorrieron diferentes distancias, partiendo con el contador de recorrido en 0 y verificando al final del trayecto que la pantalla mostraba los km recorridos. Sin embargo, la información de la pantalla se muestra solo en números enteros, las trayectorias se muestran en el Anexo B.

3.6.3. Regulación de velocidad

La señal de velocidad puede ser enviada desde el HMI, en donde se puede regular la velocidad deseada por el conductor o también desde el acelerador ubicado en el manubrio.

Se pudo constatar que la aceleración del scooter es de forma lenta y paulatina hasta alcanzar la velocidad seleccionada en la pantalla, de igual forma si se utiliza el acelerador. Para la realización de las pruebas, se utilizó el modo de aceleración desde el HMI debido a que se puede observar la velocidad seteada. Se obtuvieron los siguientes resultados, mostrados en la Figura 3.9.



Fig.3. 11. Velocidad de referencia y velocidad alcanzada por el scooter 20 y 10 km/h, respectivamente

Fuente: Autor

Las pruebas se realizaron en calzadas planas que no presentaban inclinación, en donde se constató que el vehículo alcanza la velocidad de referencia con una oscilación de un 1km/h. En inclinaciones no tan pronunciadas es capaz de llegar al valor de referencia, en calles muy empinadas el scooter se mueve a la velocidad mínima si su carga es la máxima, caso contrario se detiene.

3.7. Manual de usuario.

El manual de usuario se encuentra en el Anexo D, donde se indica el procedimiento para utilizarlo, así como, las funciones añadidas en la interfaz gráfica.

3.8. Descripción de los costos de dispositivos electrónicos seleccionados para el montaje del sistema de control.

Se presenta los valores comerciales de los diferentes materiales utilizados para la implementación, que se muestra en la siguiente tabla.

TABLA 3. 5.
COSTO DE COMPONENTES ELÉCTRONICOS PARA EL SISTEMA DE CONTROL

CANTIDAD	MATERIALES	PRECIO UNITARIO (\$)	COSTO TOTAL (\$)
1	Arduino Mega 2560	19.00	19.00
1	Pantalla touch Nextion 5"	112.00	112.00
1	Convertor DAC MCP4725	5.00	5.00
1	Reductor de voltaje (LM2596 12V-5V)	3.75	3.75
1	DC-DC reductor de voltaje(67V-12V)	28.00	28.00
1	Módulo RTC DS3132	4.00	4.00
2	Módulo L298D	3.50	7.00
1	Driver Inversor para motor brusless	65.00	65.00
1	Bocina	3.80	3.80
3	Resistencias(1M Ω ,330k Ω ,100k Ω)	0.10	0.30
1	Luces (Delantera, trasera, direccionales)	20.00	20.00
1	Acelerador	35.00	35.00
1	Key switch	4.00	4.00
2	Sensor de freno	2.00	4.00
4	Cable calibre 22 AWG	0.70	2.80
3	Bornera doble Kf301-2p	0.35	1.05
TOTAL			314.07

Fuente: Autor

Como se muestra en la tabla anterior, el costo total de los componentes electrónicos fue de 322.65 dólares; dispositivos que se pueden comprar a nivel nacional.

3.9. Discusión.

El sistema de control implementado en el prototipo de scooter eléctrico es capaz de regular la velocidad del motor Brushless mediante un control PI diseñado para corregir el error de estado estable determinado en la identificación de la planta y así alcanzar la velocidad deseada por el usuario. La aceleración se puede realizar mediante la interfaz HMI o desde el acelerador. Además de poder controlar el encendido de luces y direccionales, también permite conocer parámetros del sistema que permite al usuario conocer la velocidad del vehículo, nivel de carga y la distancia que ha recorrido.

Se realizó pruebas del sistema de control en busca de errores que pueden afectar al momento de conducción como errores en el frenado, la aceleración brusca, nivel de batería errónea, que podían afectar la experiencia de uso del vehículo, así como accidentes.

Es importante conocer que el vehículo no está diseñado para trayecto con prolongada inclinación ya que podrían afectar el rendimiento y cuásar daños a los componentes del scooter.

El vehículo puede transitar sin problema por cualquier tipo de calzada sin que las vibraciones afecten al sistema de control, como se muestra en el Anexo B, que cuenta con trayectos con tramos de adoquinado, empedrado y asfalto. Sin embargo, la vibración puede aflojar los tornillos o sujeciones del vehículo. En el manual de usuario se detalla el mantenimiento a seguir para que el scooter sea confiable.

CONCLUSIONES.

El presente trabajo está enfocado en la realización de un sistema de control mediante la plataforma de desarrollo Arduino de código abierto. La implementación del controlador es factible, lo cual permitió llegar a las siguientes conclusiones:

- Con el análisis bibliográfico realizado se logró conocer los componentes necesarios que conforman un sistema de control. Además, de los diferentes tipos de control aplicables a los vehículos eléctricos. Con la información recopilada se definió los componentes que se implementaron en el scooter como: frenos, timbre, iluminación que exigen las regulaciones internacionales. Además, el control de velocidad es necesaria en lugares con límites de velocidad que deben ser respetadas por el conductor.
- El algoritmo de control se realizó en una plataforma de código abierto, que puede implementarse con varios dispositivos electrónicos que se encuentran en el mercado nacional, sin la necesidad de la utilización de un software o hardware propietarios, permitiendo la sustitución de cualquier componente en caso de avería o daño. Además, el HMI permite expandir la funcionalidad del controlador mediante la visualización de parámetros del sistema.
- En la obtención de la función de transferencia de la respuesta de la planta se constató la existencia de un error de estado estable que no permitía al vehículo alcanzar la referencia seleccionada, se corrigió con la implementación de un controlador PI, con los parámetros; $k_p=0.0124$, $k_i=0.9964$, obtenidos en el diseño del controlador.
- El controlador de velocidad permite ajustar la velocidad deseada por el usuario entre valores de 0 a 50km/h, con un tiempo de establecimiento de 4 segundos hasta llegar a la referencia de velocidad ingresada, evitando así el arranque brusco del vehículo.
- Las pruebas de funcionamiento dieron como resultado la aceleración suave y control de velocidad utilizando los dispositivos seleccionados, demostrando así que la aplicación de un PI es adecuada para la regulación de velocidad del motor brushless hasta alcanzar la velocidad de referencia. También, la adición del HMI muestra de forma oportuna el estado de los parámetros del vehículo en tiempo real, para que el conductor pueda interpretar y tomar decisiones respectivas como: activación de luces, direccionales, reducción de velocidad del vehículo y el momento en que la batería debe ser cargada.

RECOMENDACIONES

- Para futuras mejoras del vehículo utilizar otra placa de desarrollo con mayor velocidad de procesamiento que permita realizar un control más complejo en la regulación de velocidad. Además, que permitan la transferencia de datos mediante redes inalámbricas, integrando así al scooter al internet de las cosas(IoT), que ayudaría a mejorar la funcionalidad del HMI, ya que podría visualizar sus parámetros directamente en un Smartphone.
- Para mejorar la autonomía es recomendable aumentar la capacidad de la batería, así como, de incluir un sistema de freno regenerativo que permite cargar el vehículo mientras se usa en pendientes. Además, se puede realizar la mejora de crear un sistema que permita extraer la batería y pueda ser reemplazar la batería descargada.
- Para mejorar los tiempos de carga se recomienda investigar sistemas de control de carga de batería que no degraden la vida útil de la batería y el tiempo de carga sea mucho más corto.
- Investigar los diferentes tipos de control aplicables a los vehículos eléctricos y el impacto que tienen sobre ellos.

Referencias

- alphons. (01 de 08 de 2021). *alphonsio*. Obtenido de alphonsio: <https://alphons.io/2299/what-is-the-formula-to-convert-revolutions-per-minute-rpm-to-kilometers-per-hour-km-slash-h/>
- amazon. (01 de 08 de 2021). *amazon* . Obtenido de amazon: https://www.amazon.com/-/es/Controlador-Bewinner-rendimiento-controlador-escobillas/dp/B07TRTVLCP/ref=pd_sbs_5/135-4487763-7758113?pd_rd_w=31ye6&pf_rd_p=43345e03-9e2a-47c0-9b70-a50aa5ecbd5c&pf_rd_r=PPG4GBAR72BH2GNSVH9A&pd_rd_r=5e833320-1503-4dec-b56f-a795
- Arafet, P., Chang, F., Torres, M., & Dominguez, H. (2008). *Métodos de Identificación dinámica*. Universidad de Oriente.
- Asprilla, A., Martinez, W., Mufioz, L., & Cortes, C. (2017). Design of an Embedded Hardware for Motor Control of a High Performance Electric Vehicle. *IEEE*, 5.
- BBC NEWS. (15 de 07 de 2019). *How safe are electric scooters?* Obtenido de <https://www.bbc.com/news/uk-48106617>
- Budi, D., Rizal, A., & Onnilita, A. F. (2017). Model Reference Neural Adaptive Control Based BLDC Motor Speed Control. *International Conference on Electrical, Electronics and Information Engineering*. IEEE.
- Calapiña, J. L. (2018). *SISTEMA ELECTRONICO PORTABLE PARA LA MEDICION DEL INDICE DE CONTAMINACION DE SUELOS*. Ambato: UNIVERSIDAD TÉCNICA DE AMBATO.
- Carrascosa, M. (2015). *Diseño del Regulador de velocidad de un*. Madrid.
- Casco, S. M. (2014). *Raspberry Pi, Arduino y Beaglebone Black*. Asunción .
- Castillo, M. (2010). *Control de motores C.D. por aprendizaje por imitación basado en redes neuronales*. Mexico D.F.
- Chang-liang, X. (2012). *PERMANENT MAGNET*. Singapore : John Wiley & Sons Singapore Pte. Ltd.
- Chau, K. T. (2015). *ELECTRIC VEHICLE MACHINES AND DRIVES DESIGN, ANALYSIS AND APPLICATION*. Singapore: John Wiley & Sons Singapore Pte. Ltd.
- Chen, X., Li, H., Sun, M., & Liu, G. (2018). Compensación de error de conmutación sin sensor del motor de CC sin escobillas de alta velocidad basado en el método de red neuronal RBF. *IEEE*.

- Cheng, A., Xiong, Q., Wang, L., & Zhang, Y. (2019). *Design of Drive Control Strategy for Mini Pure Electric Vehicles on the Condition of Slope Climbing*. IEEE.
- Chenguang, W., Zhiqiang, H., Yi, Y., Lingbo, G., & Ling, W. (2018). Control System Design for Micro AUV Based on Open Source Hardware. 2018 IEEE International Conference on Information and Automation (ICIA).
- Cock, L. E. (2015). *DISEÑO DE UNA ARQUITECTURA DE CONTROL PARA VEHICULOS ELECTRICOS OMNIDIRECCIONALES*. Medellín.
- Cotte, J., & Moreno, A. (2010). *DISEÑO DE CONTROL ROBUSTO DE VELOCIDAD DE MOTORES BRUSHLESS*.
- Daghero, M. D. (2019). *Scooter Eléctrico*. UNIVERSIDAD TECNOLÓGICA NACIONAL FACULTAD REGIONAL VILLA MARÍA.
- De la Fuente, M. d. (2017). *Controladores Operación manual de un proceso*. Obtenido de <http://www.isa.cie.uva.es/~maria/controladores.pdf>
- Delgado, J., & Bolaños, C. (2013). *CONTROL DE VELOCIDAD PARA MOTOR DC BRUSHLESS SIN SENSORES*.
- Department of Electronics and Instrumentation Engineering. (2017). DESIGNING HMI USING RX SERIES RX63N. *IEEE International Conference On Recent Trends in Electronics Information & Communication Technology*. India : IEEE.
- eaton. (1 de 08 de 2021). *eaton*. Obtenido de eaton: https://www.eaton.com/Eaton/ProductsServices/Electrical/ProductsandServices/AutomationandControl/Automation/OperatorInterface/XV-300/PCT_1629508
- El Comercio. (29 de 09 de 2019). *El comercio* . Obtenido de <https://www.elcomercio.com/actualidad/quito-regulacion-escuter-es-transporte-transito.html>
- Electro Industria. (Noviembre de 2015). *Electro industria*. Obtenido de <http://www.emb.cl/electroindustria/articulo.mvc?xid=2661>
- Electronica plug and play. (012 de 08 de 2021). *electronicaplugandplay*. Obtenido de electronicaplugandplay: <https://www.electronicaplugandplay.com/circuitos-integrados/product/480-amplificador-operacional-lm324>

- electronicsclub. (06 de Diciembre de 2017). *electronicsclub*. Obtenido de electronicsclub: <https://www.electronicshub.org/raspberry-pi-vs-arduino/>
- Ezzaldeen, M. M. (2018). Design of Speed-Controller for Brushless Design of Speed-Controller for Brushless Controller. *Engineering and Technology Journal*, 900-905.
- Fernández, D. (2016). *Programación de Arduino mediante MATLAB/Simulink. Aplicación al control de velocidad de motores BLDC*. Valladolid: UNIVERSIDAD DE VALLADOLID.
- García Tobar, M. (2018). *Determinación de la autonomía del vehículo eléctrico mediante ciclos controlados*. Cuenca.
- García, J. S. (2012). *CONTROL DEL SISTEMA DE ELÉCTRICO*. MÉXICO, D.F: INSTITUTO POLITÉCNICO NACIONAL.
- García, M. (2017). *Técnicas de Control de un Motor de Inducción en un Vehículo Eléctrico*. Madrid.
- Gesley, J. (31 de 05 de 2019). Obtenido de Library of congress: <https://www.loc.gov/law/foreign-news/article/austria-ban-on-riding-e-scooters-on-sidewalks-enacted/>
- Gudino, M. (05 de Marzo de 2018). *arrow*. Obtenido de arrow: <https://www.arrow.com/en/research-and-events/articles/comparing-arduino-uno-and-raspberry-pi-3#:~:text=The%20Arduino%20Uno%20has%20an,monitor%2Fkeyboard%2Fmouse>).
- Guerrero, C., Santibañez, V., & Ollervides, J. (2016). Control por Retroalimentación de Salida. *Memorias del Congreso Nacional de Control Automático*, 7.
- Haro, D., Naranjo, O., Sarzosa, M., & Sotomayor, N. (2017). Diseño y construcción de un sistema de Diseño y construcción de un sistema de del vehículo eléctrico (EVEO). *XXVII Jornadas en Ingeniería Eléctrica y Electrónica*.
- Herrera, P. (2017). *DISEÑO E IMPLEMENTACIÓN DE SISTEMA DE CONTROL DE VELOCIDAD PARA VEHÍCULO ELÉCTRICO EXPERIMENTAL*.
- Herrera, S. (Febrero de 2016). IMPLEMENTACIÓN DE ALGORITMOS DE CONTROL DE VELOCIDAD Y TORQUE DE IMPLEMENTACIÓN DE ALGORITMOS DE CONTROL DE VELOCIDAD Y TORQUE DE MULTIPROPÓSITO DE POTENCIA MEDIA UTILIZANDO RASPBERRY PI. PEREIRA.
- Jarrín, M. (1996). *Identificación de sistemas usando estimación de modelos paramétricos*. Quito: Escuela Politécnica Nacional.

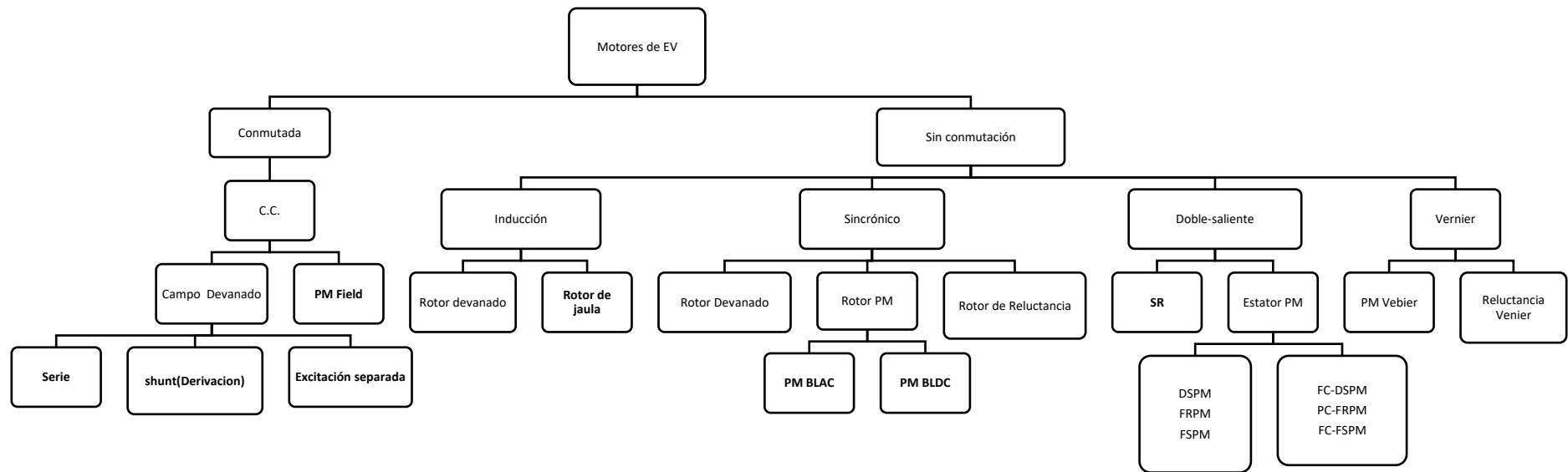
- Jianyao, H., WU, J., Hemeng, P., Qi, P., & Qingli, H. (2017). *Application of Fuzzy Logic Algorithm for Optimization of Control Strategy in Electric Vehicles*. IEEE.
- Jin-feng, X., Lei, Z., Min, O., & Fei, Z. (2012). BLDC MOTOR FIELD ORIENTATION CONTROL SYSTEM BASED ON LPIDBP NEURAL NETWORK. *IEEE*.
- Llamas, L. (2017 de Marzo de 2017). *LUIS LLAMAS*. Obtenido de LUIS LLAMAS: <https://www.luisllamas.es/arduino-paso-bajo-exponencial/>
- Lopez, D. (2014). *SISTEMA DE CONTROL LONGITUDINAL PARA VEHÍCULO ELÉCTRICO URBANO*. Cantabria.
- Marcillo, M., Mero, A., Ortiz, M., Borja, F., Álvarez, R., & Ecuador Franklin, J. (2017). *ELEMENTOS BÁSICOS DEL CONTROL DE PROCESOS*. Alicante: 3ciencias.
- Martínez, L. A. (2013). *VEHÍCULO ELÉCTRICO: ANALISIS Y PROSPECTIVA DE FACTORES TECNOLÓGICOS Y ECONÓMICOS*.
- Mercadolibre. (01 de 08 de 2021). *Mercadolibre*. Obtenido de Mercadolibre: https://articulo.mercadolibre.com.ec/MEC-431285125-pantalla-nextion-hmi-43-pulgadas-arduino-raspberry-_JM#position=1&search_layout=stack&type=item&tracking_id=8c7456b9-94a3-4d96-aa3a-d590707d6324
- Ministerio de Producción Comercio Exterior Inversiones y Pesca. (2019). *Produccion.gob.ec*. Obtenido de <http://www.produccion.gob.ec/reduccion-arancelaria-a-vehiculos-electricos-baterias-y-cargadores-para-vehiculos-electricos-promoviendo-el-acceso-a-movilidad-limpia-y-moderna/>
- Monk, S. (2016). *Raspberry Pi Cookbook*. Sebastopol: O'Reilly Media, Inc.
- Morales, J. (2018). *Control de velocidad de motores brushless mediante modulación PWM*. 2018.
- moviltronic. (s.f.). *moviltronic*. Obtenido de moviltronic: <https://moviltronics.com/tienda/sensor-41f/>
- Nava, M. (2017). *El camino de los vehículos eléctricos*. Observatorio Económico EEUU.
- Navalón, I. (2015). *DISEÑO Y PROGRAMACIÓN DE UN SISTEMA DE CONTROL BASADO EN MICROCONTROLADOR PARA EL MOTOR DE UNA BICICLETA ELÉCTRICA*. Valencia: UNIVERSIDAD POLITÉCNICA DE VALENCIA.

- naylampmechatronics. (01 de 08 de 2021). *naylampmechatronics*. Obtenido de naylampmechatronics: <https://naylampmechatronics.com/drivers/175-modulo-dac-mcp4725.html>
- Nextion. (2019). *nextion.tech*. Obtenido de nextion.tech: https://cdn.nextion.tech/wp-content/uploads/2019/06/NEXTION%E4%BA%8C%E7%BA%A7%E9%A1%B5%E9%9D%A2-%E9%BB%91%E8%89%B2-%E6%8A%98%E5%8F%A0_02.jpg
- Otero, C. (27 de 09 de 2019). *BETECH*. Obtenido de BETECH: https://as.com/meristation/2019/09/26/betech/1569534664_328235.html
- Parión, D. (2019). *IMPLEMENTACIÓN DE UN SISTEMA DE CARGA CON ENERGÍA NO CONVENCIONAL*. Ibarra.
- Patil, S., Bidari, I., Sunag, B., Gulahosur, S. V., & Shettar, P. (2016). Application of HMI Technology in Automotive Sector. *6 International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques* .
- Pro-face. (s.f.). *Proface by Schneider Electric*. Obtenido de <https://www.proface.com/en/product/hmi/st6000/top>
- Ramírez, R., Ramírez, R., Valenzuela, F., Morfin, O., Martínez, F., & Martínez, F. (2017). *Control de velocidad de un motor de CD basado en mediciones de la corriente de armadura*. Ingeniería Investigación y tecnología.
- RFI. (04 de 05 de 2019). Obtenido de New French law to ban electric scooters on pavements: <http://www.rfi.fr/en/france/20190504-new-french-law-ban-electric-scooters-pavements>
- Rodríguez, J. (2015). *Control de velocidad de motores BLDC, mediante sistemas basados en Arduino para aplicaciones en vehículos eléctricos*. Valladolid.
- Rodríguez, M. (2015). *Diseño del Regulador de velocidad de un*. Madrid.
- Saavedra, L. (2019). *Diseño e implementación de un sistema para mejorar la autonomía en un scooter eléctrico en base a energía electromagnética*. Lima.
- Sean, M., & Cook, M. (2017). *Raspberry Pi For Dummies*. Hoboken: John Wiley & Sons, Inc.
- SIEMENS. (s.f.). *SIEMENS Ingenuity for life*. Obtenido de <https://new.siemens.com/global/en/products/automation/simatic-hmi/panels/basic-panels.html>

- Te, T., Bwo, R., Yu, L., & Chen, K. (2017). *Application of Fuzzy Control for Smart Charging of Electric Vehicles*. IEEE.
- The Focal. (17 de 05 de 2019). Obtenido de THE LOCAL: <https://www.thelocal.de/20190517/e-scooters-get-the-green-light-on-germanys-roads>
- The Star . (08 de 07 de 2019). Obtenido de The Star : <https://www.thestar.com.my/tech/tech-news/2019/07/08/electric-scooters-zip-into-traffic-chaos-in-latin-america>
- Torrubia, S. (2010). *REDES NEURONALES MULTIMODELO APLICADAS AL CONTROL DE SISTEMAS*. Barcelona.
- Triviño, P., Fernández, L., Torreglosa, J., & Jurado, F. (2016). Fuzzy Logic Control for an Electric Vehicles Fast Charging Station. *International Symposium on Power Electronics,Electrical Drives, Automation and Motion*. IEEE.
- Turoń, K., & Czech, P. (2019). The Concept of Rules and Recommendations for Riding Shared and Private E-Scooters in the Road Network in the Light of Global Problems. *16th Scientific and Technical Conference "Transport Systems*. (págs. 275-284). Springer.
- University, Istanbul Technical. (2016). Implementation of PID Controller by Microcontroller of PIC (18 Series). *Advances in Robotics*, 4.
- Untuña, L. (2017). *DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA*. Sagolquí: ESPE.
- Vergara, A., Salazar, E., & Zapata, O. (2017). *Obtención de la función de transferencia de un motor de DC mediante el análisis de la curva de reacción*. Zacapoaxtla.
- Wang, Y., & Su, Y. (2015). A Research for Brake Strategy based on Fuzzy Control in Pure Electric Vehicles. *4th International Conference on Computer Science and Network Technology* . IEEE.
- Wu, X., Cui, G., Li, S., Zhang, B., & Li, Z. (2018). Braking Force Distribution of Electric Vehicles Based on Fuzzy Control. *5th International Conference on Information Science and Control Engineering*. IEEE.
- Yusong, H., Yijie, Z., & Heng, N. (2016). *Open Winding PMSM System for Electric Vehicles*. IEEE.
- Zhang, J., Wen, X., & Zeng, L. (2009). Research of parameter self-learning fuzzy control strategy in motor control system for electric vehicles.
- Zumbado, M. (2015). *AUTOS ELÉCTRICOS*. Instituto Tecnológico de Costa Rica.

Anexos

Anexo A: Clasificación de motores para los vehículos eléctricos



Fuente: (Chau, 2015).

Anexo B: Trayectos recorridos.

Trayecto 1

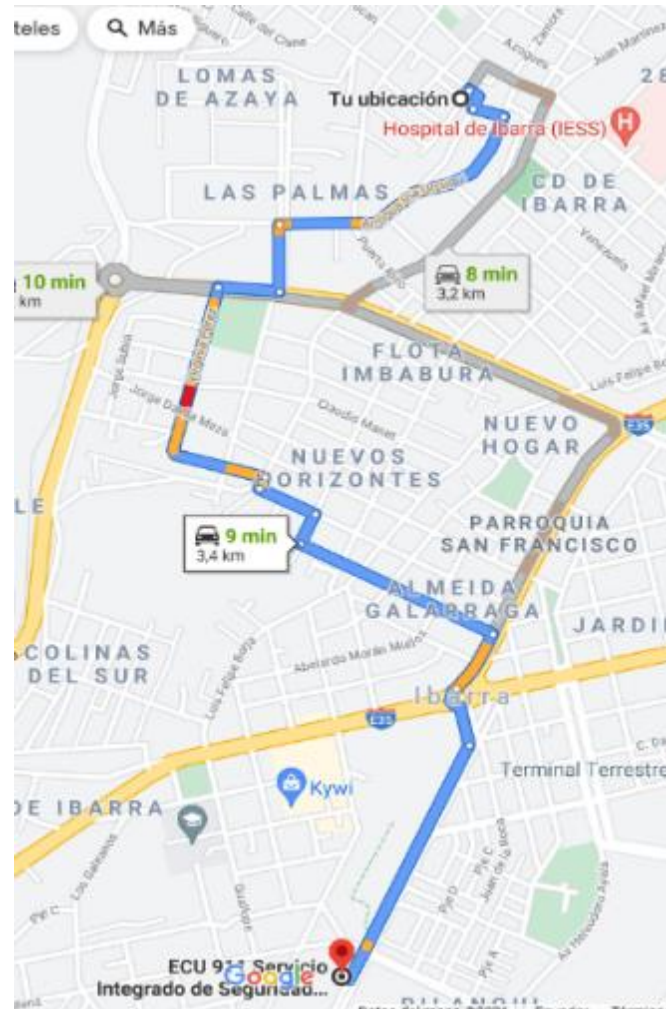


Imagen 1. Trayecto 1 recorrido para validación de distancia recorrida
Tomado de Google Maps

Trayecto 2

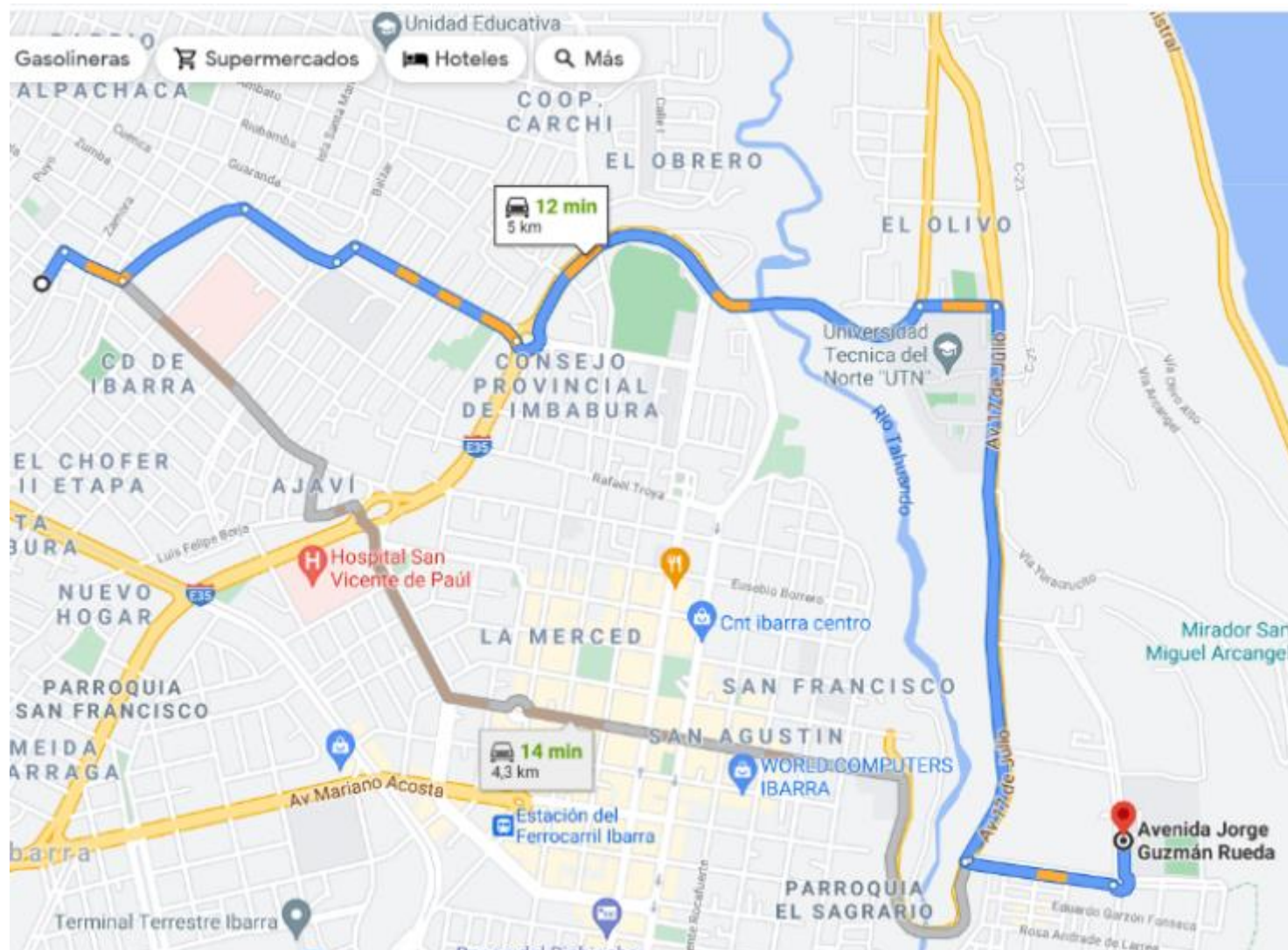


Imagen 2. Trayecto 2 recorrido para verificar distancia recorrida y pruebas de conducción
Tomado de Google Maps

Trayecto 3



Imagen 3 Trayecto 3 recorrido para verificar distancia recorrida y pruebas de conducción
Tomado de Google Maps

Anexo C. Diseño de la carcasa de la pantalla

Tapa superior

Tapa Inferior

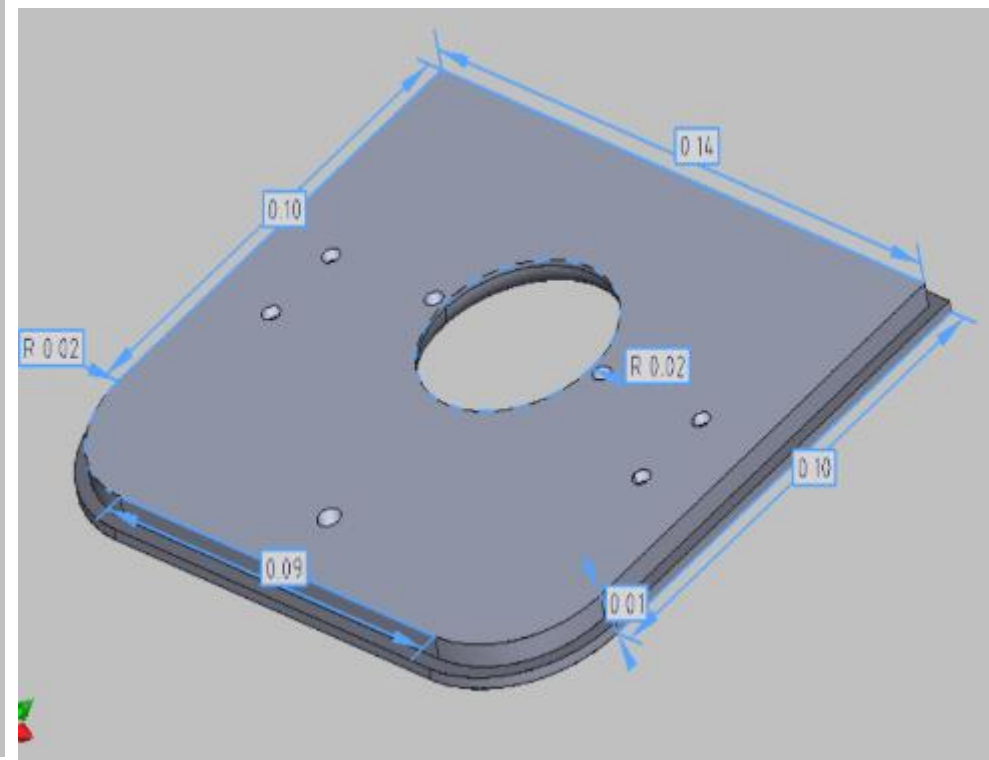
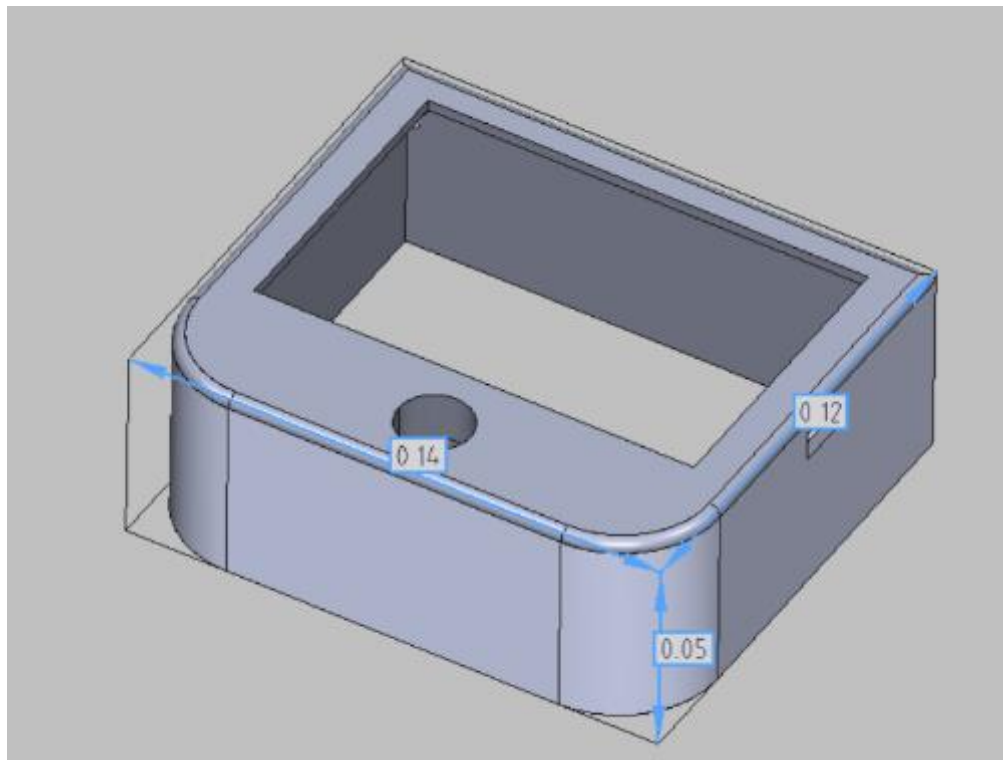


Imagen 4. diseño de la carcasa para la pantalla
Tomado de ABViewer14

Anexo D. Manual de usuario

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencia Aplicadas

Carrera de Ingeniería en Mantenimiento Eléctrico

MANUAL DE USUARIO SOBRE LA IMPLMETACIÓN DE UN SCOOTER ELÉCTRICO PARA MOVILIDAD PERSONAL Y IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL INTEGRADO A UN PROTOTIPO DE VEHÍCULO ELÉCTRICO PARA MOVILIDAD PERSONAL

Autor:

Cabeza Perugachi Jesús Ricardo

Castaño González Sebastián

Director:

MSc. Eliana Carolina Ormeño Mejía

Ibarra - Ecuador

2021

1. INTRODUCCIÓN

Por favor, lea cuidadosamente este manual de usuario antes de usar el vehículo. Un uso inadecuado de éste podría provocar daños personales, lesiones o accidentes de tráfico. Para sacar el máximo provecho de su scooter por favor lea este manual antes de usarlo.

Advertencia	Un uso erróneo puede provocar serias lesiones u accidentes de tránsito.
Atención	Un mal uso puede causar daños a su scooter
Sugerencia	Leer y seguir las instrucciones del presente manual de usuario para tener el vehículo en óptimas condiciones.

2. ADVERTENCIAS

El usuario necesita familiarizarse con el uso y funcionamiento de este vehículo antes de conducirlo. Por lo tanto, por favor siga las recomendaciones de estas advertencias.

- El usuario de este vehículo está sujeto a las mismas normas de circulación que los peatones.
- Por su seguridad, por favor siga y adhiérase las mismas leyes que los peatones.
- Circule sólo zonas peatonales y ciclo vías. Nunca circule por autopistas ni autovías.
- Tenga cuidado con el tráfico al cruzar la calle o al circular por carreteras.
- Sea extremadamente cuidadoso cuando circule con su scooter por áreas muy concurridas.
- No utilizar el scooter para subir pendientes mayores a 20 grados.
- Evitar frenados fruscos.
- Por favor, no utilice su scooter después de consumir alcohol o se encuentre bajo sustancias sujetas a fiscalización.
- Por favor, cuidado al conducir
- Antes de usar su scooter en áreas concurridas o potencialmente peligrosas familiarícese con el funcionamiento de su scooter. Practique en un área ancha

y despejada, como un parque. Para evitar accidentes con su scooter mientras conduce.

- Por favor, utilice la velocidad mínima para sus prácticas.
- Asegúrese que alguien le acompaña para su seguridad cuando conduzca por primera vez.
- Use la máxima velocidad cuando este seguro de manejar y controlar su scooter.
- El scooter fue diseñado para transportar una persona al mismo tiempo. No lleve pasajeros en su scooter (niños incluidos)
- El peso máximo que puede transportar es de 199 Kg. (incluyendo el ocupante y cualquier carga).
- Este scooter no está diseñado para uso todo terreno, solo puede ser utilizado en zonas urbanas.

EL USO INADECUADO DEL SCOOTER Y/O LA FALTA DE LECTURA DE ESTE MANUAL DEL USUARIO PUEDE LLEVARTE A LESIONES CORPORALES SERIAS. EL DISEÑADOR NO SE HACE RESPONSABLE DEBIDO A LAS LESIONES CAUSADAS POR EL USO INDEBIDO POR NO SEGUIR LO EXPLICADO EN ESTE MANUAL.

NO EXCEDA LOS LÍMITES DE VELOCIDAD Y FRENE CAUTELOSAMENTE. UTILICE SIEMPRE LA PATA CUANDO APARQUE SU SCOOTER Y RETIRE LAS LLAVES.

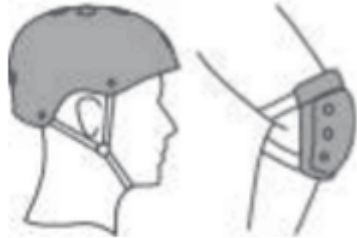
EL SCOOTER NO ESTÁ DISEÑADO PARA SOPORTAR LLUVIAS O ESTAR EXPUESTA AL AGUA POR TIEMPOS PROLONGADOS.

¡Importante!

Este scooter eléctrico debe cargarse y guardarse de forma segura. Por motivos de seguridad y para proteger la vida útil de la batería, nunca cargue el scooter eléctrico a temperaturas inferiores a 5 °C o superiores a 45 °C. Igualmente, desconecte siempre el scooter eléctrico del cable de alimentación una vez que esté completamente cargado. Use siempre el adaptador de alimentación que se suministra para cargar el scooter eléctrico.

3. MEDIDAS DE SEGURIDAD

Por favor, ten en cuenta que puedes perder el equilibrio, chocar o caerte cuando estés montando el scooter. Para evitar lesionarte, deberías estar familiarizado con este manual y mantener el scooter en buenas condiciones.



- Las medidas de seguridad deberían ser tomadas antes de montar el scooter, es decir, usar zapatos deportivos, casco, rodilleras, protectores para los codos, entre otros.
- NO USAR mientras llueve.

4. DURANTE LA CONDUCCIÓN

No use su vehículo bajo las siguientes circunstancias:

- En superficies fangosas, con grava, llenas de baches, estrechas, nevadas, heladas, o por caminos muy cercanos a canales que no están protegidos por vallas o por setos. Manténgase alejado de lugares donde las ruedas puedan quedar bloqueadas.
- No conducir cuando está lloviendo, nevando, cuando haya niebla o fuertes vientos.
- No conduzca su vehículo haciendo maniobras peligrosas ni realice giros bruscos.
- No suba con su scooter a escaleras mecánicas.
- No use su teléfono móvil o cualquier dispositivo de comunicación inalámbrico mientras conduce.
- Apague siempre su scooter y saque la llave de contacto antes de usar un teléfono móvil.
- No cargue el teléfono móvil u otro dispositivo eléctrico desde la batería de su scooter.

- No active el modo velocidad automática cuando circule en una rampa, pendiente o se encuentre en vías con tráfico.

5. PARÁMETROS DE FUNCIONAMIENTO

A continuación, se puede apreciar una tabla con los parámetros básicos de funcionamiento del scooter:

Parámetro	Resultado
Autonomía	23 km
Tiempo de carga	5 h 41 min
Velocidades Máximas	
Velocidad 1	35 Km/h
Velocidad 2	50 Km/h

Utilice el limitador de velocidad de 35 km/h, en zonas de mucho tráfico y afluencias de peatones.

6. PARÁMETROS SOBRE LA BATERÍA

A continuación, se puede apreciar una tabla con los parámetros básicos de la batería del scooter eléctrico:

PARÁMETRO	VALOR
Tiempo de carga	4-6 Horas
Voltaje	60 V
Temperatura de trabajo	5 °C - 45 °C
Voltaje de operación	48 V – 67.2 V
Corriente nominal	12 Ah
Corriente de carga mínima	4 A
Corriente de descarga máxima	25 A

No continuar usando cuando sucedan las siguientes condiciones:

- La batería del scooter eléctrico debería ser usada solamente de acuerdo con las leyes locales. El mantenimiento y desensamblaje de la batería debe ser llevado a cabo por profesionales.
- Olor o calentamiento excesivo; fuga de cualquier sustancia; no toques ningún componente o líquido que salga de la batería.
- No dejes que los niños o animales toquen la batería. Debes desconectar el enchufe del cargador antes de instalar la batería o montar.
- Si quieres que tu scooter eléctrico funcione eficientemente, la batería debería ser cargada y usada dentro de los rangos de temperatura indicados en la tabla de especificaciones de la batería.
- La batería contiene sustancias peligrosas. Por favor no abras la batería. No insertes nada dentro de la batería.
- Solamente usa el cargador original para cargar el scooter eléctrico.

Por favor, tenga en cuenta: Incluso si están completamente cargadas, las baterías se quedarán sin potencia si no se usa el scooter eléctrico durante un periodo prolongado de tiempo. Asegúrese de desconectar la batería antes de que estén al 100% para evitar daños a las celdas de la batería.

7. REVISIÓN Y MANTENIMIENTOS

• Mantenimiento de los cojinetes

El scooter eléctrico dispone de cojinetes sellados y a prueba de polvo de forma que no es necesaria su lubricación. Compruebe el desgaste de los cojinetes a intervalos regulares para evitar una conducción insegura. No suba y baje y no aplique una fuerza excesiva para evitar dañar los cojinetes. Este no es un scooter deportivo.

• Mantenimiento de las ruedas

Compruebe el desgaste de las ruedas a intervalos regulares para evitar una conducción insegura y se aconseja revisar la presión de los neumáticos cada dos semanas aproximadamente, debe estar entre 2.5 y 3 bares. Con ello se disminuirá el riesgo de accidentes y se tendrá una frenada mucho más precisa.

• Mantenimiento del freno

La zapata del freno está fabricada de metal. Compruebe el desgaste de la zapata del freno a intervalos regulares para evitar una conducción insegura. No aplique una fuerza excesiva para evitar dañar el freno.

- **Mantenimiento de tornillos y otras sujeciones**

Asegúrese de que todos los tornillos están apretados debidamente para evitar situaciones peligrosas.

ADVERTENCIA

Asegúrese de comprobar que el sistema de dirección está ajustado correctamente y que todos los componentes de la conexión están firmemente asegurados y no están rotos antes de conducir; remítase el montaje del manillar y a las instrucciones de seguridad.

8. COMO FUNCIONA EL SCOOTER ELÉCTRICO.

Para conocer el funcionamiento del scooter es necesario conocer donde se encuentra cada uno de los elementos del vehículo, en la siguiente gráfica se indica los componentes del scooter.

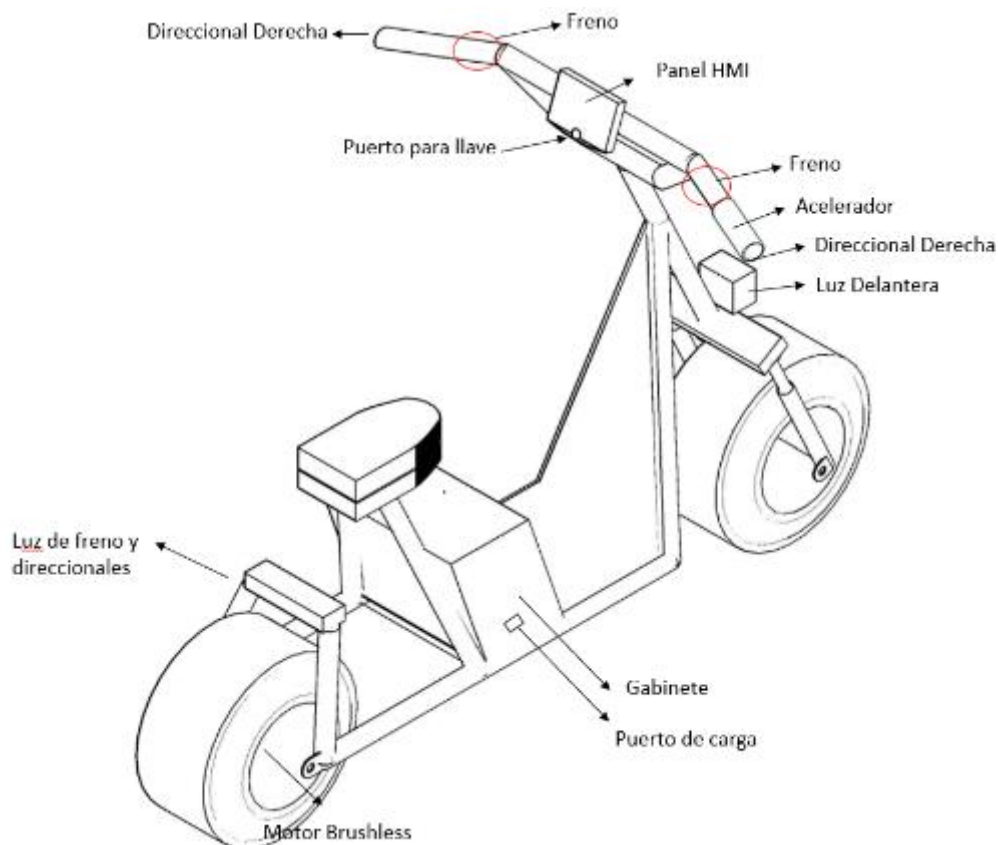


Imagen 5. Partes del scooter eléctrico
Fuente: Autor

- **Puerto para llave**

La llave de contacto actúa como interruptor de encendido del scooter. Para conectar, gire la llave en sentido de las agujas del reloj dentro su ranura y esperar aproximadamente 10 s hasta que se comuniquen los diferentes dispositivos de control.

Para desconectar, gire la llave en sentido contrario a las agujas de reloj. La pantalla de encendido debe apagarse y la llave de contacto puede ser extraída.

- **Panel HMI**

En la pantalla podemos observar la información que nos proporciona el sistema como porcentaje de batería, velocidad del scooter, estado de frenado, Hora, distancia recorrida desde el encendido, y nos permite encender las luces y elegir un modo de conducción.

- **Motor Brushless**

Se encuentra situado en la llanta trasera del vehículo, que proporciona el movimiento del vehículo.

- **Lampara delantera**

Encargada de iluminar el camino en caso de la conducción durante la noche, con la activación de la lampara delantera se activan las luces laterales que ayudan la visibilidad del vehículo ante los peatones y otros vehículos.

- **Direccionales**

Permite mostrar a los demás vehículos la intención de girar

- **Luz de freno**

Indica que el vehículo comenzó a detenerse.

- **Acelerador**

Permite la aceleración manual del scooter de manera.

- **Puerto de carga**

Conector que permite enchufar el scooter a la red eléctrica

- **Selector de velocidad**

En la parte izquierda del manubrio, existe un botón rojo con 3 posiciones, posición 1 es primera velocidad y velocidad 2 y 3 es la segunda velocidad. Seleccione la velocidad para determinar la máxima velocidad del scooter.

- **Frenos**

En el manubrio, existen frenos en la parte derecha e izquierda que permiten el frenado rápido y seguro del scooter y además deja de alimentar el motor.

- **Claxon**

Pulse el claxon para emitir el sonido. Deje de oprimir el claxon para detener el sonido.

9. Funcionamiento del panel HMI (Interfaz Hombre Máquina).

Al momento de encender el scooter, el panel HMI se iniciará con la pantalla de control de acceso, en donde deberá ingresar el usuario y el PIN de seguridad para tener acceso a la pantalla de control.

a. Pantalla de control de acceso



Imagen 6. Pantalla de bloqueo HMI
Fuente: Autor

Para introducir el usuario o el pin de seguridad, presione el espacio en blanco asignado para el usuario de igual forma con el PIN de seguridad.

Pantalla de ingreso de usuario:

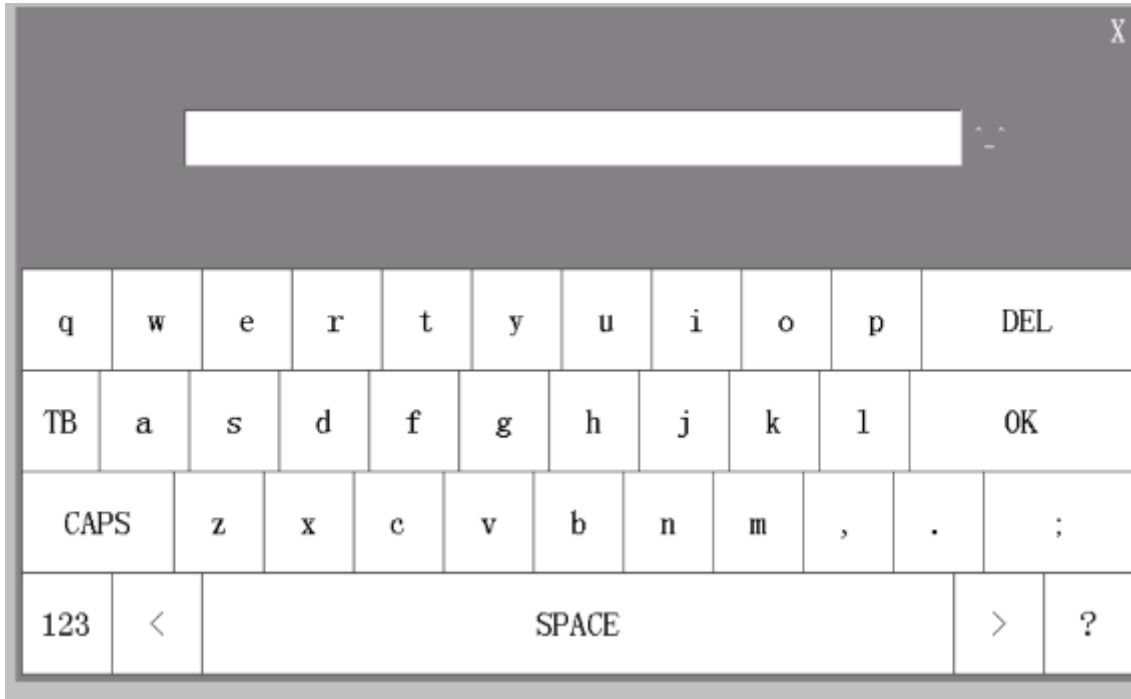


Imagen 7. pantalla de ingreso de usuario.
Fuente: Autor

Pantalla de ingreso de pin:

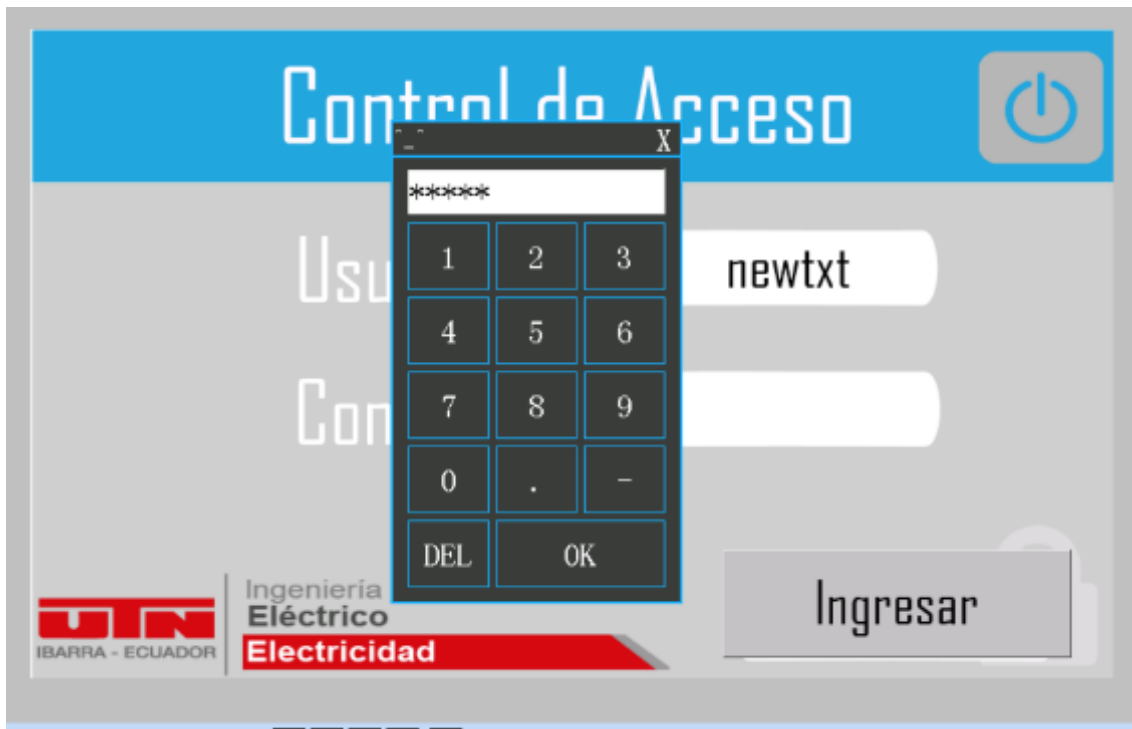


Imagen 8. Pantalla de ingreso de pin de seguridad.
Fuente: Autor

Si introduce erróneamente el usuario o pin de seguridad el sistema envía un mensaje de “Error en el usuario o contraseña”

Si se ha ingresado correctamente el usuario y el PIN al presionar el botón se ingresará a la pantalla de control

Cuando el sistema esté listo para iniciar aparece el mensaje de “OK”

b. Pantalla de control.

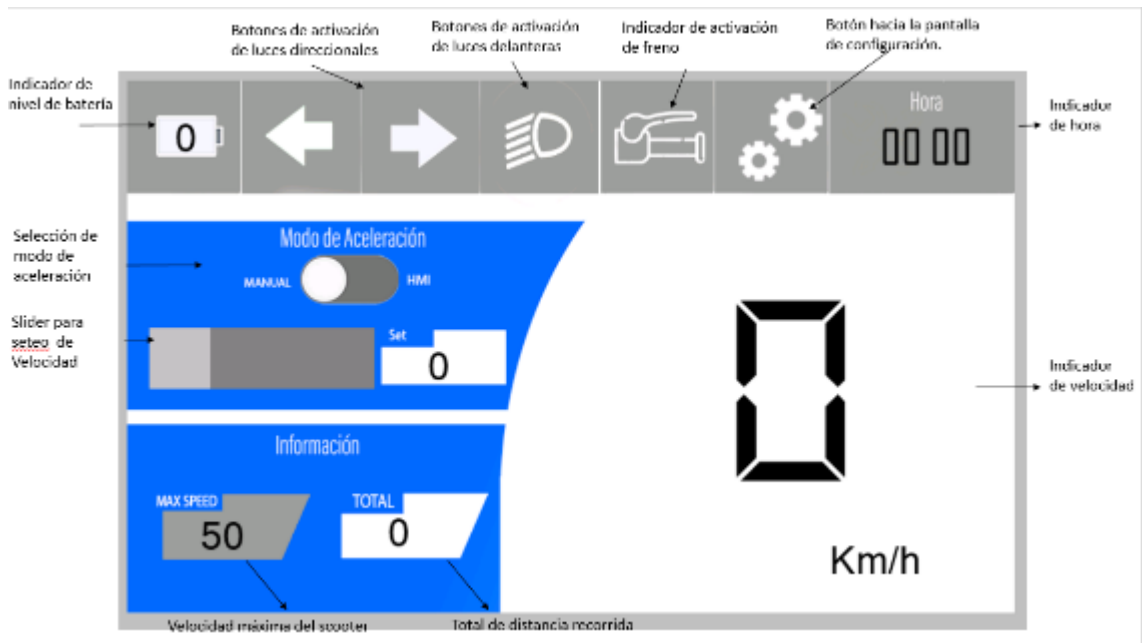


Imagen 9. Pantalla de control del sistema.

Fuente: Autor

A continuación, se muestra los botones e indicadores de la pantalla de control.

Encontramos los siguientes elementos.

Indicador de batería: proporciona información del nivel de batería del scooter, se indica de manera porcentual, en un rango de 0 a 100%. Si el nivel de batería es menor a 20% el indicador se vuelve rojo, indicando que tiene de 10 a 15 minutos para que se agote la batería.

Botones de activación de direccionales: para activar las luces direccionales basta con presionar la direccional derecha o izquierda y se encenderán las direccionales correspondientes, y se apagarán de forma automática luego de 10 segundos.

Botón de activación de luces: Al presionarse, enciende la luz delantera y las luces laterales y se apaga al volver a presionarlo.

Indicador de frenado: Se activa al momento de presionar las manijas de freno del scooter, en caso de que los frenos queden presionados este botón se mantiene encendido y no permitirá el movimiento del scooter

Botón de configuración: nos dirige a la pantalla de configuración del scooter.

Indicador de Hora: proporciona la hora actual en formato de 24 horas.

Modo de aceleración: permite seleccionar desde donde se enviará la señal de velocidad.

- Modo manual: el vehículo se acelera girando la manija del acelerador.
- Modo HMI: la velocidad deseada se ingresa mediante la slider situada en la parte inferior del switch de modo de aceleración

Para cambiar de modo solamente presione el switch (manual/HMI), el sistema inicia en modo manual.

Para el seteo de velocidad en modo de aceleración HMI debe mover el slider de izquierda a derecha para ingresar la velocidad deseada.

Tome en cuenta el modo de aceleración HMI el vehículo se detendrá completamente cuando se setea la una velocidad de cero o se cambie a modo manual

Max Speed: indica la velocidad máxima alcanzada por el scooter

TOTAL: Indica la distancia recorrida por el scooter desde el momento de encendido.

Indicador de velocidad: muestra la velocidad actual del vehículo

c. Pantalla de configuración.

La pantalla configuración nos muestra.



Imagen 10. Pantalla de configuración del HMI.
Fuente: Autor

Botón volver: dirige a la pantalla de control

Administrador de usuario: dirige a la pantalla de administrador de usuario.

Ajuste de brillo: permite aumentar o disminuir el brillo de la pantalla

Botón volver bloqueo: nos permite bloquear el scooter nuevamente.

Botón reset: reinicia la distancia recorrida guardada.

d. Pantalla administradora de usuarios.

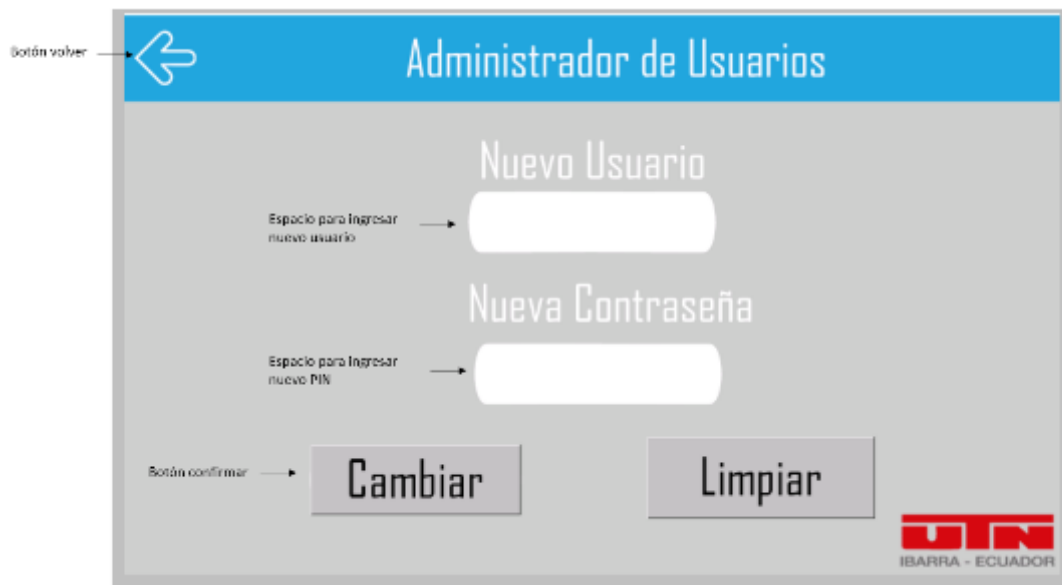


Imagen 11. pantalla de administrador de usuario

Fuente: Autor

Botón volver: devuelve a la pantalla de control

Nuevo usuario: permite el ingreso de nuevo usuario.

Nuevo PIN: permite el ingreso de un nuevo PIN.

Botón cambiar: confirma los cambios realizados.

Botón limpiar: Borra los espacios de Nuevo usuario y contraseña.

ANEXO E. Código de programación del sistema de control

```
/*
 * UNIVERDIDAD TÉCNICA DEL NORTE
 * SISTEMA DE CPNTROL DE VELOCIDAD
 * AUTOR: JESÚS CABEZAS
 */
// Declaración de librerías.
#include <EEPROM.h>// librería para lectura y escritura de la EEPROM
#include "Nextion.h"//libreria para recepción y transmición de datos con la pantalla
#include <Wire.h>//libreria para la comunicación I2C
#include <Adafruit_MCP4725.h>//librería para comunicación con del DAC
#include <arduino-timer.h>// librería para ejecutar el timer del control PID
#include <RtcDS1307.h>// Librería para comunicación con el rtc reloj
RtcDS1307<TwoWire> Rtc(Wire);// Librería para comunicación con el rtc reloj
//
Adafruit_MCP4725 dac;
Timer<1, micros> timer;// declaracion del timer

//Variables para modo de aceleracion y activación de luces
uint32_t estadode;
uint32_t estadoiz;
uint32_t estadokset;
uint32_t valorset;
int estadoiz1,estadode1,set,sets=0;
// Variables para el panel de acceso, usuario y contraseña.
String user="";
String PSWRD="";
int pass;
//Variables para parpadeo de dirrecionales.
int pin=HIGH;
int parpadeo1=0;
int parpadeo2=0;
unsigned long tiempo1 = 0;
```



```

unsigned long tiempo2 = 0;
// Declaracion de los pines de entrada y salida
const int freno1=A7;//sensores de freno
//const int freno2=7;//sensores de freno
const int luzizquier=26;//activacion de direccional izquierda
const int luzdere=27;//activacion de direccional derecha
const int luzdelante=24;//activacion de luces delanteras
const int luzfreno=25;//activacion de luces de freno
//const int sensorbocina=28;//activacion de luces de freno
//int sensorbat=A2;//entrada lectura nivel de bateria
//int sensorace=A1;//entrada lectura acelerador
// declaracion de variables para el contador de recorrido
unsigned long contadorkm=0;
float totalkm=0;
int n=1;
int pulsos=0;
int npage=0;
//
unsigned long pasado=0;
unsigned long actual;
//
unsigned long tiempo pasado=0;
unsigned long tiempo actual;
// Variables para suavizar la senal del acelerador y bateria
float alpha=0.05,Y=0;
float S=Y;
float alpha1=0.01,Y1=0;
float S1=Y1;

//Variables para el calculo del PID
float Sp;//valor de referencia
float Pv;//valor medido rpm
int Cv=0;
float e[3]={0,0,0};// para el error e[0]error actual, e[1] error anterior, e[2],error dos veces
anterior.
float u[2]={0,0};// u[0] salida actual,u[1] salida anterior.

```

```

float kp=7.7;
float ki=23;
float kd=0.6;
float T=0.001;
int T1=100;//Tiempo de de demora para enviar informacion a la pantalla
float K1,K2,K3;//K1 acompaña el error actual K2 el error anterior, k3 el error dos veces
anterior
//Variables para lectura y suavizado de la señal de voltaje y velocidad
#define NUMREADINGS 20 //10 los valores para leer
int readings[NUMREADINGS]; //Las lecturas de la entrada analógica
int index = 0; //Indice de la lectura actual
int total = 0; // lectura Total
int average = 0; //Media
#define NUMREADINGS1 20 //10 los valores para leer

int readings1[NUMREADINGS1]; //Las lecturas de la entrada analógica
int index1 = 0; //Indice de la lectura actual
int total1 = 0; // lectura Total
int average1 = 0; //Media

float vcc = 5; // valor real de la alimentacion de Arduino, Vcc
float r1 = 1000000; // 1M
float r2 = 100000; // 100K
//Lectura de bateria.
//float vcc = 5; // valor real de la alimentacion de Arduino, Vcc
float r12 = 1330000; // 1M
float r22 = 100000; // 100K
int nivelbateria;
//VARIABLE PARA EL CONTROL DE PANTALLA
NexPage page0 = NexPage(0, 0, "page0");
NexPage page1 = NexPage(1, 0, "page1");
NexPage page2 = NexPage(2, 0, "page2");
NexPage page3 = NexPage(3, 0, "page3");
//Variables de pagina 1
//NexPicture p1 = NexPicture(1,7, "p1");
NexDSButton btizquierda = NexDSButton(1, 3, "btiz");
NexDSButton btderecha= NexDSButton(1, 4, "btderecha");

```

```

NexDSButton btluces = NexDSButton(1, 5, "luces");
NexDSButton btset = NexDSButton(1, 6, "btset");
NexButton botonfreno = NexButton(0, 15, "bt1");
NexButton botonguardar = NexButton(0, 20, "b0");
NexNumber n0 = NexNumber(1, 9, "n0");
NexNumber n1 = NexNumber(1, 8, "n1");
NexSlider h0 = NexSlider(1, 17, "h0");
//variables pagina 2
NexButton botonreset = NexButton(2, 10, "b1");
//Variables de pagina 3
NexText t0 = NexText(3, 6, "t0");
NexText t1 = NexText(3, 7, "t1");
NexButton b4 = NexButton(3, 3, "b4");
;
char buffer[100] = {0};
NexTouch *nex_listen_list[] =
{
    &page0,
    &page1,
    &page2,
    &page3,
    &botonreset,
    &btset,
    &btizquierda,
    &btderecha,
    &btluces,
    &botonfreno,
    &botonguardar,
    &h0,
    &t0,
    &t1,
    &b4,
    NULL
};
void page0PopCallback(void *ptr)
{
    npage=0;

```

```

Serial.print("t2.txt=");
Serial.print("\n");
Serial.print(user);
Serial.print("\n");
Serial.write(0xff);
Serial.write(0xff);
Serial.write(0xff);
Serial.print("t3.txt=");
Serial.print("\n");
Serial.print(PSWRD);
Serial.print("\n");
Serial.write(0xff);
Serial.write(0xff);
Serial.write(0xff);

}

void page1PopCallback(void *ptr)
{
npage=1;
}

void page2PopCallback(void *ptr)
{
npage=2;
}

void page3PopCallback(void *ptr)
{
npage=3;
}

void buttonbizquierda(void *)
{
    btizquierda.getValue(&estadoiz);
    estadoiz1=estadoiz;
}

```

```

}
void buttonderecha(void *)
{
    btderecha.getValue(&estadode);
    estadode1=estadode;

}
void buttonluces(void *){

    uint32_t estadoluz;
    btluces.getValue(&estadoluz);
    if(estadoluz==1){
        digitalWrite(luzdelante,1);
    }else{
        digitalWrite(luzdelante,0);
    }

}
void buttonset(void *){

    btset.getValue(&estadoset);
}

void h0PopCallback(void *ptr)
{
    uint32_t number = 0;
    h0.getValue(&number);
    valorset=number;
    float km1=(number*2*3.6*3.14*0.3)/60;
    int rpm2=int(km1);
    Serial.print("n1.val=");
    Serial.print(rpm2);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
}

```

```

}
void resetdistancia(void *ptr)
{
  EEPROM.writeInt(80,0);
  EEPROM.writeInt(90,0);
}

bool calculopid(void *) {

  // Lectura del punto de ajuste,Sp
  if(estadoset==1){// lectura de switch de modo de aceleracion
    Sp=abs(map(valorset,0,455,0,100));// se escala los valores de la pantalla a porcentaje
  }else{//
    Sp=promedioacelerador();
    Sp=map(Sp,177,874,0,100);//Escalamiento del setpoint(acelerador)
    if(Sp<=874){
      e[2]=0;
      e[1]=0;
      u[1]=0;
      e[0]=0;
    }
  }

  float rpm1=voltajerpm();
  Pv=rpm1;
  Pv=abs(map (Pv,0,455,0,100));
  //Pv=abs(map (Pv,0,455,0,100));// Escalamiento del la lectura de RPM
  e[0]=Sp-Pv;//Escalamiento del setpoint(potenciometro)
  //calcular PID
  u[0]=K1*e[0]+K2*e[1]+K3*e[2]+u[1];//salida actual del controlador.
  Cv=u[0];//ajuste de senal
  //Cv=map(Sp,0,100,800,3150);
  if(Cv>3150){
    Cv=3150;
  }
  Cv=constrain(Cv,800,3150);
}

```

```

    dac.setVoltage(Cv,false);
    //corrimientos por software
    e[2]=e[1];
    e[1]=e[0];
    u[1]=u[0];
    return true; // repeat? true
}
// Funcion para cambiar usuario y contraseña y guardarla en la memoria EEPROM
void cambiar()
{
    memset(buffer, 0, sizeof(buffer)); // Clear the buffer, so we can start using it
    t0.getText(buffer, sizeof(buffer));
    user=buffer;// Read the text on the object t7 and store it on the buffer
    memset(buffer, 0, sizeof(buffer)); // Clear the buffer, so we can start using it
    t1.getText(buffer, sizeof(buffer));
    PSWRD=buffer;
    update_StringEEPROM(0, user);
    pass=PSWRD.toInt();
    EEPROM.writeInt(40,pass);

}
bool write_StringEEPROM(int Addr, String input)
{
    char charbuf[15];
    input.toCharArray(charbuf,15);
    return EEPROM.writeBlock<char>(Addr, charbuf, 15);
}

bool update_StringEEPROM(int Addr, String input)
{
    char charbuf[15];
    input.toCharArray(charbuf,15);
    return EEPROM.updateBlock<char>(Addr, charbuf, 15);
}

String read_StringEEPROM(int Addr)
{

```

```

String outputEEPROM;
char output[] = " ";

EEPROM.readBlock<char>(Addr, output, 15);
//convert to string
outputEEPROM = String (output);
return outputEEPROM;
}

void setup() {
  // Lectura de usuario y contraseña de la memoria EEPROM
  pass=EEPROM.readInt(40);
  PSWRD=pass;
  user = read_StringEEPROM(0); //String = read_StringEEPROM(Address)
  contadorkm=EEPROM.readInt(80);
  n=EEPROM.readInt(90);
  {
  for (int i = 0; i < NUMREADINGS; i++)
  readings[i] = 0; //Pone toda la matriz a 0
  }
  // calcular los coeficientes de la ecuacion en diferencias
  K1=kp+(ki*T)+(kd/T);
  K2=(-2.0*(kd/ T))-kp;
  K3=kd/T;
  // llama la funcion calcular pid cada 1ms
  timer.every(1000,calculopid);
  dac.begin(0x60);
  nexlInit();
  delay(500); //
  Serial.begin(115200);
  pinMode(luzizquier,OUTPUT);
  pinMode(luzdere,OUTPUT);
  pinMode(luzdelante,OUTPUT);
  pinMode(luzfreno,OUTPUT);
  Rtc.Begin();
  dac.setVoltage(0,false);
}

```



```

btizquierda.attachPush(buttonbizquierda);
btderecha.attachPush(buttonderecha);
btluces.attachPush(buttonluces);
btset.attachPush(buttonset);
page0.attachPush(page0PopCallback);
page1.attachPush(page1PopCallback);
page2.attachPush(page2PopCallback);
page3.attachPush(page3PopCallback);
botonreset.attachPush(page3PopCallback);
b4.attachPush(resetdistancia);
h0.attachPop(h0PopCallback);
tiempo1 = millis();
}

```

```

void loop() {

```

```

    nexLoop(nex_listen_list);//verifica si se presiono algun boton de la pantalla
    if(npage==0){// envia datos de usuario y contraseña a la pantalla y mensaje de
confirmacion de comunicacion entre pantalla y arduino

```

```

    Serial.print("t2.txt=");
    Serial.print("\n");
    Serial.print(user);
    Serial.print("\n");
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.print("t3.txt=");
    Serial.print("\n");
    Serial.print(PSWRD);
    Serial.print("\n");
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.print("t5.txt=");
    Serial.print("\n");
    Serial.print("OK");
    Serial.print("\n");

```

```

Serial.write(0xff);
Serial.write(0xff);
Serial.write(0xff);
e[0]=0;
Cv=0;

}
if(npage==1){
  //t_timer.tick();
  if(analogRead(freno1)<=900){
    digitalWrite(luzfreno,0);
    timer.tick(); // tick the timer
    actual=millis ();// medidos el tiempo de inicio del programa
    unsigned long dt=actual-pasado;
    if (dt>T1){//tiempo de retardo para enviar informacion a la pantalla
      pasado=actual;//valor de pasado se igual al actual para una nueva etapa de muestreo
      int rpm1=voltajerpm();// lectura de rpm del motor
      float km1=(rpm1*2*3.6*3.14*0.3)/60;//trasformación de rpma a km
      Serial.print("n0.val=");
      Serial.print(int(km1));
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.print("bt1.val=");
      Serial.print(0);
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.write(0xff);
      tiempoactual=millis ();
      //Acumulador de revoluciones del motor.
      unsigned long unsegundo=tiempoactual-tiempopasado;
      if(unsegundo>1000){
        tiempopasado=tiempoactual;
        int dist=voltajerpm();
        pulsos=dist/60;
        contadorkm=contadorkm+pulsos;

```

```

}
// calculo de distancia recorrida.
totalkm=(1.8*contadorkm)/1000;
if(totalkm>n)
{
    // envio de de datos a la pantalla de distancia recorrida.
    n++;
    int totalkm1=int(totalkm);
    //guarda datos de distancia recorrida y comparador en la memoria EEPROM de arduino
    EEPROM.writeInt(80,contadorkm);
    EEPROM.writeInt(90,n);
    Serial.print("n4.val=");
    Serial.print(totalkm1);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
}
RtcDateTime now = Rtc.GetDateTime();
printDateTime(now);
}
}

if (analogRead(freno1)>900){
    e[2]=0;
    e[1]=0;
    u[1]=0;
    e[0]=0;
    int rpm1=voltajerpm();
    float km1=(rpm1*2*3.6*3.14*0.3)/60;
    Serial.print("n0.val=");
    Serial.print(int(km1));
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    digitalWrite(luzfreno,1);
    dac.setVoltage(800,false);
    Serial.print("bt1.val=");

```

```

Serial.print(1);
Serial.write(0xff);
Serial.write(0xff);
Serial.write(0xff);
int nivelbat=lecturabateria();
nivelbateria=map(nivelbat,4700,6600,0,100);
nivelbateria=constrain(nivelbateria,0,100);
Serial.print("n2.val=");
Serial.print(nivelbateria);
  Serial.write(0xff);
Serial.write(0xff);
Serial.write(0xff);
if(nivelbateria<20){
Serial.print("b3.val=");
Serial.print(1);
  Serial.write(0xff);
Serial.write(0xff);
Serial.write(0xff);
}else{
Serial.print("b3.val=");
Serial.print(0);
  Serial.write(0xff);
Serial.write(0xff);
Serial.write(0xff);

}
Cv=0;
int totalkm1=int(totalkm);
Serial.print("n4.val=");
Serial.print(totalkm1);
Serial.write(0xff);
Serial.write(0xff);
Serial.write(0xff);

}
}

```

```
/*PARPADEO DE DIRECCIONALES*/
```

```
//luz izquierda
```

```
if (estadoiz1 == 1){  
    tiempo2 = millis();  
    if(tiempo2 > (tiempo1+500)){ //Si ha pasado 0.5 segundo ejecuta if  
        tiempo1 = millis(); //Actualiza el tiempo actual  
        digitalWrite(luzizquier,pin);  
        pin=!pin;  
        parpadeo1=parpadeo1+500;  
  
    }  
    if(parpadeo1 > 5000){  
        digitalWrite(luzizquier,0);  
        estadoiz1=0;  
  
        parpadeo1=0;  
  
    }  
}
```

```
//luz derecha
```

```
if (estadode1 == 1){  
    tiempo2 = millis();  
    if(tiempo2 > (tiempo1+500)){ //Si ha pasado 0.5 segundo ejecuta if  
        tiempo1 = millis(); //Actualiza el tiempo actual  
        digitalWrite(luzdere,pin);  
        pin=!pin;  
        parpadeo2=parpadeo2+500;  
    }  
    if(parpadeo2 > 5000){  
        digitalWrite(luzdere,0);  
        estadode1=0;  
        parpadeo2=0;  
    }  
}  
  
if(npag==3){  
    RtcDateTime now = Rtc.GetDateTime();
```

```

        printDateTime(now);
    }

}

int promedioacelerador()// creacion de la funcion promedio
{
    Y=(float)analogRead(A0);// lectura del potenciometro y guardado en Y
    S=(alpha*Y)+((1-alpha)*S);// calculo de la senal suavizada
    delayMicroseconds(100);//retardo de 100 microsegundos
    return S;
}

int lecturabateria()// creacion de la funcion promedio
{
    total1 -= readings1[index1]; //Resta la última lectura
    readings1[index1] = analogRead(A1); //Lee los valores de A0 y guarda en la matriz
    total1 += readings1[index1]; //Añade la lectura a total
    index1 ++; //Incrementa el índice
    if (index1 >= NUMREADINGS1){
        index1 = 0; //Al final de la matriz se restablecerá el índice a 0
    }
    average1 = total1 / NUMREADINGS1; //Calcula la media
    float v12 = (average1 * vcc) / 1024.0;
    float v22 = ((v12 / (r22 / (r12 + r22)))+0.7)*100;
    return v22;
}

int voltajerpm()// creacion de la funcion promedio
{
    total -= readings[index]; //Resta la última lectura
    readings[index] = analogRead(A2); //Lee los valores de A0 y guarda en la matriz
    total += readings[index]; //Añade la lectura a total
    index ++; //Incrementa el índice
    if (index >= NUMREADINGS){
        index = 0; //Al final de la matriz se restablecerá el índice a 0
    }
    average = total / NUMREADINGS; //Calcula la media
    float v1 = (average * vcc) / 1024.0;

```

```
float v2 = (v1 / (r2 / (r1 + r2))*100);
v2=map(v2,0,3270,0,455);
return v2;
}
void printDateTime(const RtcDateTime& dt)
{
  Serial.print("hora.val=");
  Serial.print(dt.Hour(), DEC);
  Serial.write(0xff);
  Serial.write(0xff);
  Serial.write(0xff);
  Serial.print("minuto.val=");
  Serial.print(dt.Minute(), DEC);
  Serial.write(0xff);
  Serial.write(0xff);
  Serial.write(0xff);
}
```