



**INSTITUTO TECNOLÓGICO SUPERIOR**

**VIDA NUEVA**



**CARRERA:**

**TECNOLOGÍA EN ELECTROMECAÁNICA**

**TEMA DEL PROYECTO DE TITULACIÓN:**

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO, PARA EL CONTROL INALÁMBRICO DE UNA ONE WHEEL.**

**AUTOR:**

**REINOSO TOASA MARCO ANTONIO**

**TUTOR:**

**ING. RUIZ GUANGAJE CARLOS RODRIGO**

**FECHA:**

**MARZO 2018**

**QUITO – ECUADOR**

## **CESIÓN DE DERECHOS DE AUTOR**

Yo, **REINOSO TOASA MARCO ANTONIO** portador/a de la cedula de ciudadanía **172358644 - 0**, facultado/a de la carrera **TECNOLOGÍA EN ELECTROMECAÁNICA**, autor/a de esta obra certifico y proveo al Instituto Tecnológico Superior Vida Nueva, usar plenamente el contenido plasmado en este escrito con el tema “**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO, PARA EL CONTROL INALÁMBRICO DE UNA ONE WHEEL.**”, con el objeto de aportar y promover la lectura e investigación, autorizando la publicación de mi trabajo de titulación en la colección digital del repositorio institucional bajo la licencia de Creative Commons: Atribución-NoComercial-SinDerivadas.

En la ciudad de Quito, al mes de Marzo de 2018.

-----  
**REINOSO TOASA MARCO ANTONIO**

**C.I.: 172358644 - 0**

## **CERTIFICACIÓN**

En mi calidad de Tutor del Proyecto: **DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO, PARA EL CONTROL INALÁMBRICO DE UNA ONE WHEEL**, Presentado por el estudiante **REINOSO TOASA MARCO ANTONIO**, para optar por el título de Tecnólogo en Electromecánica, certifico que dicho proyecto ha sido revisado en todas sus partes y considero que reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal que se designe.

-----

Tutor: Ing. Ruiz Guangaje Carlos Rodrigo

## **APROBACIÓN DEL TRIBUNAL EXAMINADOR**

Los miembros del tribunal examinador aprueban el informe del Proyecto de Aplicación Práctica, con el tema:

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO, PARA EL CONTROL INALÁMBRICO DE UNA ONE WHEEL.**

Del Sr. estudiante: REINOSO TOASA MARCO ANTONIO

De la Carrera, Tecnología en Electromecánica.

Para constancia firman:

.....

.....

.....

## **DECLARACIÓN DE AUTENTICIDAD**

Yo, **REINOSO TOASA MARCO ANTONIO**, estudiante del Instituto Tecnológico Superior Vida Nueva, declaro que he realizado este trabajo de titulación tomando en consideración citas bibliográficas que se nombran en este texto.

A través de la presente declaración cedo mis derechos de propiedad correspondiente a este trabajo de titulación al Instituto Tecnológico Superior Vida Nueva.

-----  
REINOSO TOASA MARCO ANTONIO

C.I 1723586440

## **AGRADECIMIENTO**

En primer lugar a Dios por haberme bendecido dándome fortaleza para continuar, a mis padres, quienes a lo largo de mi vida han velado por mi bienestar y educación siendo mi apoyo incondicional en todo momento y seré un gran orgullo para ellos y para todos los que confiaron en mí, a mis hermanos Jesús y Gabriel por siempre haberme dado su fuerza y apoyo incondicional que me han ayudado y llevado hasta donde estoy ahora. Es por ellos que soy, lo que soy ahora.

Mis más sinceros agradecimientos al Instituto Tecnológico Superior Vida Nueva, a mi tutor el Ingeniero Carlos Ruiz por su valiosa guía, en la realización de mi proyecto de titulación.

Reinoso Toasa Marco Antonio

## **DEDICATORIA**

Dedico este proyecto a Dios y a mis padres y mis hermanos, a Dios porque ha estado conmigo día a día con cada paso que doy, cuidándome y guiándome para continuar, a mis padres, por el amor que me brindan siendo mi apoyo en todo momento, a mis hermanos quienes están en todo momento conmigo aconsejándome y ayudándome en mi formación profesional; depositando su entera confianza en cada reto que se me presentaba sin dudar ni un solo momento de mi capacidad.

Reinoso Toasa Marco Antonio



## ÍNDICE GENERAL

<b>OBJETIVOS</b> .....	<b>3</b>
<b>1.1. Objetivo General</b> .....	<b>3</b>
<b>1.2. Objetivos Específicos</b> .....	<b>3</b>
<b>2. DESARROLLO O CUERPO PRINCIPAL</b> .....	<b>4</b>
<b>4.1.1. Marco Teórico – Conceptual</b> .....	<b>4</b>
<i>4.1.2. Automatización</i> .....	<i>4</i>
<i>4.1.3. Arduino Uno</i> ... ..	<i>4</i>
<i>4.1.4. Pines de entradas, salidas y alimentación de Arduino Uno</i> ... ..	<i>5</i>
<i>4.1.5. Tarjeta IShield</i> .....	<i>5</i>
<i>4.1.6. Pantalla 5” TFT Touch</i> .....	<i>6</i>
<i>4.1.7. Conexión entre la pantalla TFT y Arduino Uno.</i> .....	<i>7</i>
<i>4.1.8. Sensor de orientación</i> ... ..	<i>7</i>
<i>4.1.10. Alimentación</i> ... ..	<i>8</i>
<i>4.1.11. Bateria</i> .....	<i>9</i>
<i>4.1.12. Motores eléctricos</i> ... ..	<i>9</i>
<i>4.1.13. Motores DC</i> .....	<i>10</i>
<i>4.1.14. Motor de imanes permanentes</i> .....	<i>10</i>
<i>4.1.15. Puente H</i> .....	<i>10</i>
<i>4.1.15.1. Aplicación de un puente H</i> .....	<i>10</i>
<i>4.1.14. Neumático</i> ... ..	<i>11</i>
<b>4.2. PROCEDIMIENTO – METODOLOGÍA</b> .....	<b>11</b>
<i>4.2.1 Diseño</i> ... ..	<i>11</i>
<i>4.2.2. Sistema de control</i> ... ..	<i>11</i>
<i>4.2.3. Dispositivo de control programable</i> ... ..	<i>12</i>
<i>4.2.4. Red inalámbrica</i> ... ..	<i>13</i>

4.2.5. Enlace de Arduino Uno y 1Sheeld .....	13
4.2.6. Descripción general e implementación .....	13
4.2.7. Proceso de control.....	14
<b>4.3. CONSTRUCCIÓN.....</b>	<b>14</b>
4.3.1. Armazón .....	14
4.3.2. Sistemas de control.....	15
4.3.3. Programación en el software de Arduino... ..	16
4.3.3.1. Programación de LCD .....	17
4.3.3.2. Programación de Toggle Button.....	17
4.3.3.3. Programación de Orientación.....	18
<b>4.4. SISTEMA DE POTENCIA.....</b>	<b>21</b>
4.4.1. Motores DC ... ..	21
4.4.2. Construcción de partes... ..	22
4.4.3. Acoplamiento de los motores reductores y baterías .....	23
4.4.4. Regulador de voltaje.....	24
4.4.5. Inversión de giro .....	26
4.4.6. Distribución del cableado... ..	27
4.4.7. Alimentación.....	27
<b>3. CONCLUSIONES.....</b>	<b>29</b>
<b>4. RECOMENCACIONES.....</b>	<b>30</b>
<b>5. FUENTES .....</b>	<b>31</b>
<b>5.1. BIBLIOGRAFÍA. ....</b>	<b>31</b>
<b>6. ANEXOS .....</b>	<b>32</b>

## ÍNDICE DE FIGURAS

FIGURA N° 1 Tarjeta Arduino Mega con 1Sheeld...	6
FIGURA N° 2 Pantalla 5" TFT Touch (HMI)...	6
FIGURA N° 3 Orientación en tres grados de libertad...	7
FIGURA N° 4 Módulo Sheld de relés de 8 canales...	8
FIGURA N° 5 Composición eléctrica...	9
FIGURA N° 6 Motor AC y Motor DC...	9
FIGURA N° 7 Llanta de caucho 5/8"...	11
FIGURA N° 8 Diseño del armazón...	15
FIGURA N° 9 Unión de 1Sheeld y Arduino Uno	15
FIGURA N° 10 Ubicación de 1Sheeld y Arduino Uno en acrílico	16
FIGURA N° 11 Visualización del escudo LCD	17
FIGURA N° 12 Visualización del escudo Toggle Button...	18
FIGURA N° 13 Visualización del escudo de Orientación...	22
FIGURA N° 14 Motores DC utilizados...	22
FIGURA N° 15 Motores reductores y baterías...	23
FIGURA N° 16 Chumacera	24
FIGURA N° 17 Eje- piñón- chumacera	24
FIGURA N° 18 Regulador de voltaje...	25
FIGURA N° 19 Diagrama del circuito regulador de voltaje...	25
FIGURA N° 20 Diseño del circuito en baquelita	26
FIGURA N° 21 Circuito completo...	26
FIGURA N° 22 Puente H...	26
FIGURA N° 23 Puente H con relés	27
FIGURA N° 24 Distribución del cableado por secciones	27
FIGURA N° 25 Baterías de 12v para cada motor DC...	28
FIGURA N° 26 Baterías de 6v conectados en seria para cada motor DC...	28
FIGURA N° 27 Comprobación del sistema	36
FIGURA N° 28 Construcción del eje para la llanta...	36
FIGURA N° 29 Construcción del chaveteo en el eje...	36
FIGURA N° 30 Piñón en el eje del motor...	37
FIGURA N° 31 Acople del piñón en el eje	37

FIGURA N° 32 Vista preliminar .....	37
FIGURA N° 33 .Construcción de base con tubo de 1”1/4 .....	38
FIGURA N° 34 Construcción de la base para las chumaceras .....	38
FIGURA N° 35 Construcción de chavetero en eje .....	38
FIGURA N° 36 Montaje en la base .....	39

## ÍNDICE DE TABLAS

TABLA 1. Características de Arduino Uno. ....	4
TABLA 2. Conexión de la pantalla TFT a Arduino Uno.....	7
TABLA 3. Pines para el módulo de relés de 8 canales... ..	8
TABLA 4. Datos técnicos de los motores de DC .....	23
TABLA 5. Corrientes y voltajes en vacío .....	23
TABLA 6. Corrientes y voltajes con carga .....	23

## ÍNDICE DE ANEXOS

8.1.1. Código de programación para mover adelante, atrás y giro .....	31
8.1.2. Programación para encender una tira de Led con relé... ..	33
8.1.3. Programación con datos específicos... ..	33
8.1.5. Construcción de eje en el torno... ..	35
8.1.6. Elaboración de chavetero en el eje de la llanta... ..	35
8.1.7. Acople del piñón en el eje del motor.....	35
8.1.7. Acople de piñón en el eje de la llanta.....	36
8.1.8. Ensamblaje de piezas vista preliminar... ..	36
8.1.9. Base de tubo de 1" ¼.....	36
8.1.10. Base para alojar las chumaceras... ..	37
8.1.11.Fijación de llantas en la base .....	37
8.1.12. Ensamblaje de los motores y llantas... ..	37

## INTRODUCCIÓN

El proyecto tiene como finalidad controlar un sistema mecánico de manera inalámbrica, con tecnología que se utiliza hoy en día; la comunicación inalámbrica nos brinda la posibilidad de transferir datos de manera rápida, manteniendo la conectividad a una red con las mismas características como si estuviéramos conectados a una red cableada, esta tecnología reduce los precios respecto a una instalación cableada.

El código abierto y software extensible, Arduino posee herramientas de códigos abierto, basado en microcontroladores AT-MEGA 168 Atmel, utiliza un lenguaje mediante librerías C++, compatible para shields modulares que se insertan encima de otras para dar funcionalidad extra a un Arduino la comunicación son por pines digitales y analógicos o por puerto serie.

La utilización de la tarjeta 1Sheeld permite extraer los sensores virtuales del dispositivo Smartphone con el propósito de controlar inalámbricamente el sistema mecánico, facilitando de esta manera la utilización de varios sensores del dispositivo siendo un sistema completamente seguro, se utilizara los sensores virtuales giroscopio y acelerómetro y de orientación para el control del sistema mecánico

Se implementará un sistema de monitoreo inalámbrico de bajo costo, gracias a los avances tecnológicos de visualización digital, se requiere que los datos adquiridos por los distintos sensores virtuales del Smartphone se refleje en la pantalla (HMI), proporcionada por una interfaz gráfica de control y visualización entre el hombre y la máquina, Cabe destacar que la única forma de interacción entre la aplicación y el usuario, es mediante su interfaz gráfica, la cual está diseñado Nextion Edition.

Además el proyecto se basa en la construcción de una patineta ONE WHEEL; el control que se realizará será completamente inalámbrico con la implantación de Arduino Uno, el cual posee toda la programación para los movimientos; por otro lado la tarjeta 1Sheeld se encargará de extraer los sensores virtuales del Smartphone siendo así un proyecto innovador, creativo y seguro para el usuario.

## ANTECEDENTES

La automatización es un proceso aplicado en diferentes tecnologías, para el control y monitoreo de una serie de procedimientos, recibiendo señales de entrada las cuales son procesadas y posteriormente ejecutan una acción; esto permite controlar la secuencia de las operaciones sin la intervención del hombre y de esta manera disminuir el esfuerzo físico; aumentando así el rendimiento, la calidad y la productividad.

Según, Gutierrez (2000) Arduino fue diseñado como una plataforma educativa basada en WIRING, para programar el microcontrolador ATMEGA328P-PU; accesible al código abierto en hardware y software, fue pensado para estudiantes, con la finalidad de reducir precios; se debe tomar en cuenta que los proyectos de Arduino pueden ser autónomos o pueden comunicarse con software en ejecución en un ordenador (p.8).

La tarjeta 1Sheeld convierte un Smartphone en un Arduino Shield configurable; está interconectado a una aplicación móvil totalmente gratuita, que permite el uso de todos los sensores del Smartphone en el sistema operativo de Android, tales como: Giroscopio, Acelerómetro, Magnetómetro, GSM, Wi-Fi, GPS, entre otros.

Nextion Human Machine Interface por sus siglas (HMI), es una solución que incluye hardware en una serie de placas TFT y de software que es el editor del display, para la comunicación utiliza un puerto serial, y proporciona una interfaz de control y visualización entre el hombre y un proceso, optimizando de esta manera el tiempo para el registro de control.

Los sistemas inalámbricos facilitan la comunicación entre el emisor y el receptor; con el objetivo de transferir información a través de ondas electromagnéticas entre dispositivos digitales; el propósito es controlar y monitorear el proceso en una distancia determinada, sin la necesidad de poseer cables en exceso.



## **RESUMEN**

El control inalámbrico hoy en día es aplicado en todo tipo de sistemas o aplicaciones ya sean eléctricas como mecánicas, el objetivo principal del control inalámbrico de sistemas es dotar al usuario de facilidades del control a distancia con el objetivo de controlar los actuadores de forma remota, entre las aplicaciones más conocida se encuentran los sistemas domóticos, los mismo que buscan controlar todas las cargas conectadas a sistema eléctrico residencial por medio de un celular o computadora, otro tipo de aplicaciones se la puede evidenciar en las industrias de producción o manufactura en donde los procesos industriales son controlados incluso desde otras ciudades con el objetivo de arrancar o detener los procesos, determinar los daños provocados en tiempo real e incluso simplemente realizar el conteo del material que se produce. Es por eso que el presente proyecto tiene como finalidad controlar un sistema mecánico por medio de los sensores que tiene un celular, específicamente por medio del acelerómetro, este sensor permite detectar el movimiento del mismo e incluso identificar cuando se lo ha girado o inclinado, los datos obtenidos por este sensor son transferido mediante bluetooth a un Arduino Uno el cual procesa la cantidad de movimiento o los grados que se encuentra girado el celular para enviar señales de control hacia un bloque de relés para establecer cuáles son aquellos que se deben encender, es importante considerar que estos relés controlan un motor reductor de corriente continua que se ubica en la base de la ONE WHEEL y permite el desplazamiento hacia adelante, atrás o girarla hacia la derecha o la izquierda.

### **PALABRAS CLAVE:**

Arduino Uno, One Shield, Motor reductor, Acelerómetro, Control inalámbrico.

## ABSTRACT

### **3. OBJETIVOS**

#### **3.1. Objetivo General**

Diseñar e implementar un sistema electrónico, mediante el uso de 1Sheeld y Arduino Uno, para el control inalámbrico de una ONE WHEEL, en el Instituto Tecnológico Superior Vida Nueva.

#### **3.2. Objetivos específicos**

- Diseñar la plataforma de comunicación de 1Sheeld aplicando software código de programación libre.
- Establecer la interfaz gráfica en el software de Nextion Edition del display TFT de 5 pulgadas (HMI), para la visualización del panel de diseño y manipulación de la aplicación.
- Implementar el sistema electrónico de potencia para el control de los motores de corriente continua.
- Evaluar el funcionamiento del sistema electrónico y de potencia en la parte mecánica.

## 4. DESARROLLO O CUERPO PRINCIPAL

### 4.1.1. Marco Teórico – Conceptual

#### 4.1.2. Automatización

Los primeros sistemas de control surgen en la Revolución Industrial a finales del siglo XIX y principios del siglo XX, están basados en componentes mecánicos y electromagnéticos con un proceso continuo, estos sistemas son capaces de simular la movilidad de máquinas donde el operario disminuye esfuerzos físicos.

#### 4.1.3. Arduino Uno

Arduino es una plataforma de código abierto basada en hardware y software fácil de programar, posee un microcontrolador ATMEGA 328P-PU Arduino fue pensado para realizar diseños de prototipos, mediante la recepción de señales de entradas, utilizando una variedad de sensores mediante el control de actuadores como luces, motores que se pueden utilizar para desarrollar elementos autónomos.

**Tabla 1**

*Características de Arduino Uno*

<b>Característica</b>	<b>Descripción</b>
Microcontrolador	ATmega 328P-PU
Voltaje de operación	5v
Tensión de entrada (recomendada)	7-12v
Tensión de entrada (límite)	6-20v
Pines digitales de E/S	14 ( de los cuales 6 proveen salidas PWM )
Pines de entradas analógicas	6
Corriente DC por pin E/S	40 miliamperio
Corriente DC para pin 3.3v	50 miliamperio
Memoria flash	16KB ( de los cuales 2KB usados para bootloader )
SRAM	1KB
EEPROM	512 bytes
Frecuencia de reloj	16MHz

Fuente: Elaboración propia

#### **4.1.4. Pines de entradas, salidas y alimentación de Arduino Uno**

La tarjeta Arduino posee 14 pines que pueden ser utilizados como entradas y salidas, usando funciones `pinMode()`, `digitalWrite()` y `digitalRead()`. Cada uno de los pines funciona a 5 volt, cada pin proporciona y recibe un máximo de corriente 40mA.

**VIN.** La entrada de tensión a la placa Arduino cuando está usando una fuente de alimentación externa (al contrario de los 5 voltios de la conexión USB u otra fuente de alimentación regulada). Puede suministrar tensión a través de este pin, o, si suministra tensión a través del conector de alimentación, acceder a él a través de este pin.

**5V.** El suministro regulado de energía usado para alimentar al microcontrolador y otros componentes de la placa. Desde VIN a través de un regulador en la placa, o ser suministrado por USB u otro suministro regulado de 5v. Un suministro de 3.3 V generado por el chip FTDI de la placa. La corriente máxima es de 50 mA.

**GND.** Pines de Tierra.

#### **4.1.5. Tarjeta 1Sheeld**

La tecnología que presenta la tarjeta 1Sheeld convierte a un dispositivo móvil Smartphone/Tablet, siendo una tarjeta electrónica compatible para la plataforma de Arduino (UNO, DUE, MEGA, LEONARDO.) y proporciona conectividad inalámbrica vía Bluetooth entre la tarjeta y el Smartphone basada en una plataforma de software se debe instalar una app que proporciona diferentes Shields y periféricos virtuales sobre el dispositivo móvil, en el software de Arduino se debe instalar los drives cuyas funciones permiten realizar programaciones que interactúan con los periféricos virtuales del móvil.

La tarjeta 1Sheeld se conectará únicamente con tarjetas compatibles de Arduino (UNO, DUE, MEGA, LEONARDO.), el móvil dotado de sistema operativo de Android en su versión según el fabricante 2.3 o superior.

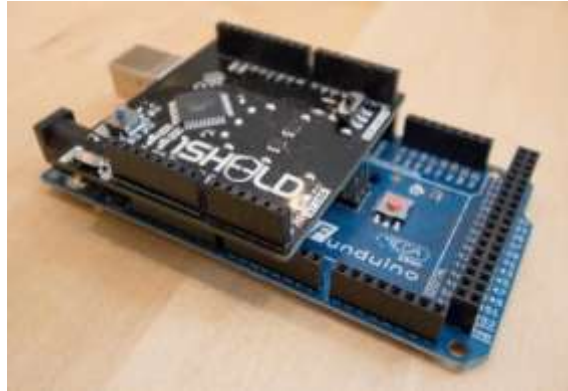


FIGURA 1. Tarjeta Arduino Mega con IShield.

#### 4.1.6. Pantalla 5" TFT Touch

Las pantallas TFT touch compatible para versión de Arduino están equipados con interfaces de control LCD, la creación de un puerto serie HMI separa por completo el control de usuario de la pantalla; por medio de la utilización del software Visual TFT instalado en los equipos de control, por lo tanto realiza la reducción de las dificultades en el desarrollo y la mejora de la eficiencia de trabajo y un rendimiento de visualización, está integrado con una memoria flash no volátil de 1GB, lo que podría permitir a los usuarios almacenar texto, imágenes fijas, imágenes GIF y archivos de sonido, etc. Para ello, los usuarios deben utilizar el software de PC para descargar imágenes requeridas en la memoria flash del panel de operador por medio de USB, y luego controlar funciones como la visualización de imágenes mediante el envío de instrucciones en serie en el puerto correspondiente, todo el proceso es fácil de entender y operar.



FIGURA 2. Pantalla 5" TFT Touch (HMI)

#### 4.1.7. Conexión entre la pantalla TFT y Arduino

**Tabla 2**

*Conexión de la pantalla TFT a Arduino Uno*

Módulo TFT	Arduino Uno, NANO
SDO(MISO)	Pin 12
LED	Pin 3.3v o fuente externa 3.3v
SCK	Pin12
SDI(MOSI)	Pin 11
D/C	Pin 9
RESET	Pin 3.3v o fuente externa 3.3v
CS	Pin 10
GND	Pin GND
VCC	Pin 3.3v o fuente externa

FUENTE: Elaboración propia.

#### 4.1.8. Sensor de orientación

El sensor de orientación en el dispositivo Smartphone da información del valor del acimut en grados sexagesimales y las rotaciones pitch y roll.

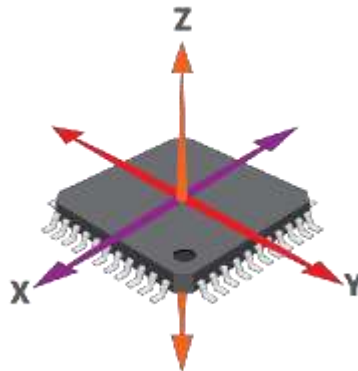


Figura 3. Orientación en tres grados de libertad

#### 4.1.9. Módulo de relés de 8 canales

En la tarjeta Arduino los pines de salida son útiles para controlar cargas que no consuman excesiva corriente, pero son insuficientes para cargas mayores; los módulos de relés para Arduino, funcionan a 5 Voltios, capaces de manejar cargas de hasta 10 Amperes en 250 Voltios, aislados mediante optoacopladores de las entradas, las que cuentan con leds

individuales que sirven como indicadores de estado; cada uno de los relés que presenta el bloque actúan como interruptores para controlar cargas AC Y DC.



FIGURA 4. Módulo Shield de relés de 8 canales

#### 4.1.10. Alimentación

El módulo de relés presenta pines de conexiones; el jumper que posee tres pines se puede seleccionar de manera que todo el módulo funcione en conjunto, es decir seleccionar con el jumper jd-vcc y vcc, o bien alimentar de forma independiente tanto optoacopladores como relés quitando el jumper.

La nomenclatura de los pines son GND - IN1 - IN2 – IN3 – IN4 – IN5 – IN6 – IN7 – IN8 - VCC.

**Tabla 3**

*Pines para el módulo de relés de 8 canales*

Pines del módulo de relés	Pines de la tarjeta de Arduino
<b>GND</b>	Pin a tierra
<b>IN1</b>	Pines designados por el programador, como salidas para la activación y desactivación de los relés.
<b>IN2</b>	
<b>IN3</b>	
<b>IN4</b>	
<b>IN5</b>	
<b>IN6</b>	
<b>IN7</b>	
<b>IN8</b>	
<b>VCC</b>	5 volt de la tarjeta de Arduino Uno

Fuente: Elaboración propia.



#### 4.1.11. Batería

Las baterías poseen una infinidad de formas físicas internamente poseen químicos que transmiten electrones, es un proceso electro—químico que genera electricidad a través de reacciones químicas; las baterías poseen dos polos, un polo positivo (+) y un polo negativo (-). Los electrones de carga negativa se dispersan al polo negativo hacia el polo positivo, o sea, son recogidos por el polo positivo.

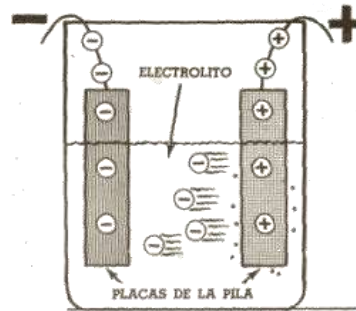


Figura 5. Composición Eléctrica.

#### 4.1.12. Motores Eléctricos

Los motores eléctricos son máquinas eléctricas que transforman en energía mecánica la energía eléctrica, por medio de integraciones electromagnéticas, los motores de AC y DC se basan en el mismo funcionamiento, el cual establece que si un conductor “esmaltado” por el circula una corriente eléctrica se encuentra dentro de la acción de un campo magnético, este tiende a desplazarse perpendicularmente a las líneas de acción del campo magnético; el conductor tiende a funcionar como un electroimán debido a la corriente que es sometido obteniendo propiedades magnéticas en el polo norte y el polo sur, el resultado de esta acción tiende a girar produciéndose así energía mecánica.

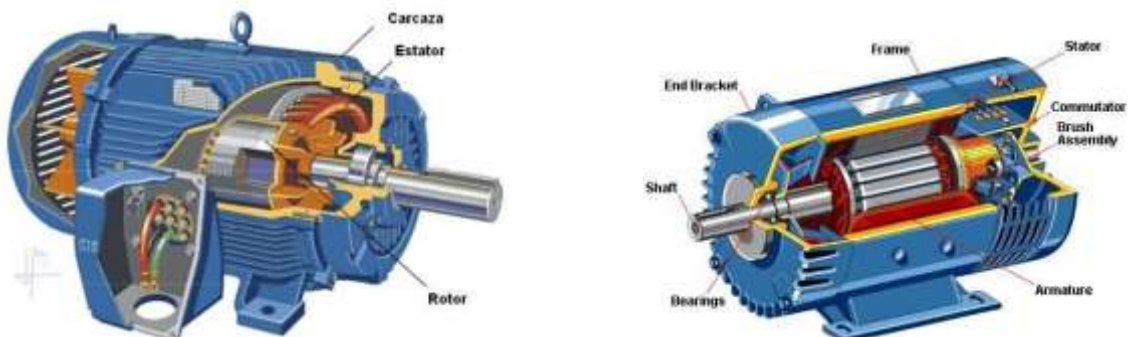


Figura 6. Motor AC y Motor DC

#### **4.1.13. Motor de DC**

Los motores de DC, convierten la energía eléctrica a energía mecánica; la principal característica de los motores de DC, es la posibilidad de controlar y regular la velocidad desde vacío hasta con carga este tipo de motor puede actuar como generador o motor.

El estator posee polos salientes, su bobinado es de cobre y la corriente se transmitirá a través de las escobillas, haciendo que el rotor gire producido por el campo magnético; solo en casos especiales el estator posee imanes permanentes.

#### **4.1.14. Motor de imanes permanentes**

El campo magnético de un motor DC, es generado por bobinas o imanes permanentes, permitiendo así el movimiento del rotor, se pueden clasificar de acuerdo con el diseño de la armadura.

Los motores DC tienen escobillas mecánicas y conmutadas, dependiendo de la armadura se clasifican en tres tipos:

- De núcleo de hierro
- De devanado superficial
- De bobina móvil

#### **4.1.15. Puente H**

En la actualidad este tipo de circuitos, permite a un motor eléctrico DC girar en ambos sentidos anti horario y horario se puede construir con elementos eléctricos y/o electrónicos.

Se compone de 4 interruptores mecánicos relés o mediante transistores, el puente H no solo se usa para invertir el giro de un motor también es utilizado para realizar un frenado de manera brusca al hacer un corto en los terminales del motor.

##### ***4.1.15.1. Aplicaciones de un Puente H***

- Automatizaciones de procesos industriales
- En ascensores de edificios
- En teclas eléctricas
- Circuitos electrónicos
- Transportar objetos
- Puentes grúas

#### **4.1.16. Neumático**

Generalmente son de forma toroide de caucho blando interiormente tiene una cámara de tuvo lleno de aire que se coloca en las ruedas de vehículos o máquinas.

Su principal función es permitir un contacto adecuado en la adherencia y fricción con el pavimento, posibilitando el arranque, el frenado y la guía.



*Figura 7. Llanta de caucho 5/8"*

## **4.2. Procedimiento – Metodología.**

### **4.2.1. Diseño**

Los parámetros que influyen directamente en el diseño mecánico del proyecto.

- Capacidad máxima de carga = 90kg.
- Dimensiones sugeridas para la estructura = 120 mtrs x 50 cm 25 cm.
- Velocidad máxima de desplazamiento = 4.5 m/s.
- Inclinación máxima de ascenso 10°

Tomando en cuenta que los parámetros establecidas son datos propios con sistemas que justifican factores de seguridad.

### **4.2.2. Sistema de control.**

El sistema de control es el siguiente proceso:

- Programación mediante PC.
- Implementación de la tarjeta 1SHEELD Y ARDUINO UNO.
- Simulación con relés.

La simulación es necesaria en cualquier tipo de proceso principalmente para corregir errores antes de la implementación del sistema definitivo.

Se utilizó el software de Arduino para la programación de la tarjeta Arduino Uno; la aplicación de 1Sheeld se la descarga de Play Store tomando en consideración la versión del Android que posee el dispositivo Smartphone.

- Reconocimiento del sensor virtual por el Smartphone
- Lectura del sensor virtual de Orientación
- Lectura del sensor virtual de LCD
- Lectura del sensor virtual Toggle Button
- Salida de señales digitales control de los relés
- Movimiento del motor 1 y motor 2.

El objetivo del control por medio del sensor virtual de orientación es obtener dos grados de libertad en el eje de las (X) y el eje de las (Y) para su movimiento en cuatro direcciones adelante, atrás, izquierda y derecha.

El circuito de control está separado del sistema de potencia de los motores para evitar corto circuito en el mecanismo.

Para el mecanismo; la protección del sistema de control se dimensiono un fusible por medio de la lectura de la pinza amperimétrica el consumo de las corrientes del motor uno y motor dos y se llegó a la conclusión de colocar dos fusibles de 5ampérios por motor.

#### **4.2.3. Dispositivo de control programable**

Actualmente los sistemas electrónicos y sistemas de telecomunicaciones utilizan un nivel de requerimientos en el diseño y su funcionamiento se vuelve complejo; en la actualidad el desarrollo de nuevas plataformas hardware han hecho que se desarrollen circuitos integrados digitales programables reduciendo la complejidad y abaratando costos en circuitos

Un dispositivo de control permite configurar a través de la programación un diseño digital lo cual puede ser realizado por el usuario con las especificaciones necesarias para realizar los circuitos siendo un control rápido y eficiente en el diseño.

#### **4.2.4. Red Inalámbrica.**

La implementación de un nuevo sistema de control inalámbrico fue pensado para solucionar el inconveniente del cableado, para que el usuario tenga la seguridad y comodidad de controlar el sistema, evitando los problemas que se han presentado en la utilización de distintos sensores.

El cableado excesivo en los sistemas de control puede ocasionar fallos eléctricos y daños físicos, ocasionados por corto circuito al no llevar una distribución apropiada; el exceso de cableado provoca interferencias en la transmisión de señales a largas distancias; una mejor manera de controlar actuadores y visualizar los sensores es vía Bluetooth, permitiendo manipular el sistema mecánico, eliminando así el cableado que poseía anteriormente para recibir las señales de comunicación, siendo así un sistema seguro para el usuario y disminuyendo el costo de los sensores.

#### **4.2.5. Enlace de Arduino Uno y 1Sheeld.**

La unión de tarjeta 1Sheeld y Arduino Uno se convertirá en una sola tarjeta electrónica diseñada para extraer todos los datos y sensores virtuales del dispositivo Smartphone; disminuyendo el cableado excesivo que se utilizaría para cada uno de los sensores implementados, el control inalámbrico vía bluetooth, es una de las mejores formas para la transmisión y recepción de datos; la implantación de la pantalla TFT de 5" (HMI) en el sistema, el usuario podrá manipular y observar sus funciones y el estado del sistema mecánico.

#### **4.2.6. Descripción general e implementación.**

El diseño será de programación de lazo abierto, ya que no tendrá ninguna restricción por el programador, el usuario tendrá todo el beneficio del sistema, podrá mejorar a un futuro el funcionamiento del mecanismo.

La tarjeta 1Sheeld y Arduino Uno, será la etapa de control el cual ejecutará la programación principal, para el control de potencia de los motores DC, la activación y desactivación los relés, el movimiento y el sentido de giro de los motores DC; la aplicación del control inalámbrico se la obtiene de Play Store con versión 4.1.2 en adelante; la comunicación será vía bluetooth, evitando así el cableado que sería para la emisión y recepción de datos; la

pantalla TFT 5" (HMI) será una interfaz gráfica que permitirá observar y ser manipulado por el usuario.

La etapa de potencia será controlada por el módulo de relés de 8 canales que controlará el giro de dos motores DC con un máximo de corriente de 10 amperios cada uno, constará con el circuito de alimentación se utilizará dos baterías de 12 voltios independiente para cada motor DC con una corriente de 1.5 amperios.

#### **4.2.7. Proceso de control**

El sistema mecánico se dispondrá de un interruptor ubicado en el control principal el cual encenderá y apagará todo el sistema de control y posee dos fusibles de protección en el sistema de potencia.

La aplicación de 1Sheeld será el control inalámbrico; primero se emparejará el dispositivo Smartphone con 1Sheeld vía Bluetooth, se seleccionara los tres escudos principales: el escudo de orientación, el escudo de push button y el escudo del LCD.

La interfaz gráfica de la pantalla TFT, indicará aspectos de los datos proporcionados por el programador hacia el usuario.

El sistema de potencia consta de dos motores DC de 12 voltios el cual esta acoplado por medio de dos piñones, el primero piñón esta acoplado al eje del motor DC y por medio de la cadena esta acoplado al eje de la llanta, el cual permitirá los movimientos adelante atrás izquierda y derecha.

### **4.3. Construcción**

#### **4.3.1. Armazón**

El diseño y construcción del armazón es el sistema de soporte para implementar los diversos elementos que conformaran el sistema mecánico que permitirán el control de los movimientos de los motores DC.

La estructura es diseñada con el fin de soportar diversos esfuerzos que será sometido por el usuario y que sea lo más ligero posible; por el movimiento de los motores DC.



*Figura 8. Diseño del armazón*

#### **4.3.2. Sistemas de control**

El sistema electrónico se ubicó en la parte superior del armazón optimizando el espacio interno para que el sistema de control este aislado de la parte de potencia; la implantación de 1Sheeld y Arduino Uno se ubicó en una base de acrílico, se colocaron de tal forma de que el usuario pueda acceder a la manipulación de estos elementos.



*Figura 9. Unión de 1Sheeld y Arduino UNO*

La alimentación de la tarjeta Arduino Uno y 1Sheeld contará con un regulador de voltaje de 12v a 0v, conectado en serie un fusible de 1 amperio, para la protección de las tarjetas electrónicas; el sistema tendrá un interruptor general que encenderá y apagará todo el mecanismo.

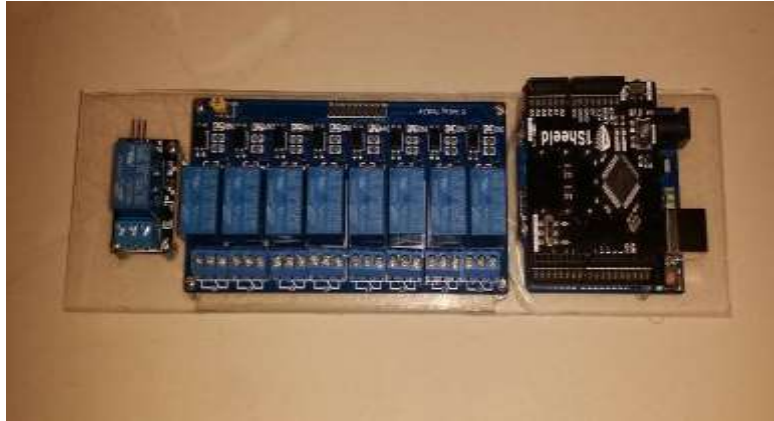


Figura 10. Ubicación de IShield y Arduino uno en el acrílico

### 4.3.3. Programación en el software de Arduino

#### 4.3.3.1. Programación de LCD

```
#define CUSTOM_SETTINGS
#define INCLUDE_LCD_SHIELD
#include <OneSheeld.h>
int An;
void setup() {
  OneSheeld.begin();
  LCD.begin();
  LCD.clear();
}
void loop()
{
  LCD.home();
  An=analogRead(A0);
  LCD.print(" PROYECTO TESIS MARCO REINOSO ");
}
```

La librería INCLUDE\_LCD\_SHIELD, se incluirá las líneas de código y se define la constante int que se establece como variable para la lectura de un canal analógico, define los pines que se va a utilizar como salidas para enviar los datos de programación.



La Función `setup ()` se establece como configuración inicial en el programa; `OneSheeld.begin` inicia la comunicación con el dispositivo remoto, `LCD.begin` da inicio al LCD en la primera posición y la primera columna el `LCD.clear` borra la pantalla LCD.

La función `loop ()` ejecuta un número infinito de veces, en este caso es la conversión de `An0`, y en `LCD.print` se visualiza en el display del dispositivo.

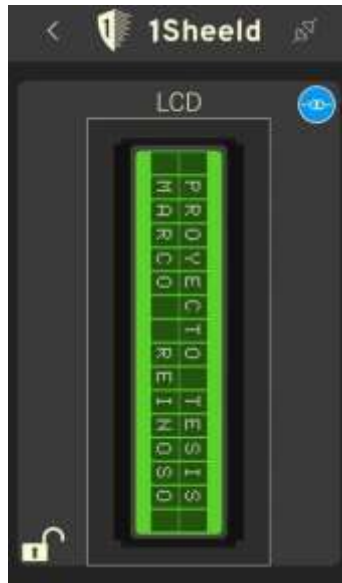


Figura 11. Visualización del Escudo LCD

#### 4.3.3.2. Programación de Toggle Button

```
#define CUSTOM_SETTINGS
#define INCLUDE_TOGGLE_BUTTON_SHIELD
#include <OneSheeld.h>
byte rele = 4;
void setup(){
  pinMode(rele,OUTPUT);
  OneSheeld.begin();
}
void loop(){
  digitalWrite(rele,ToggleButton.getStatus());
}
```

La librería `INCLUDE_TOGGLE_BUTTON_SHIELD`, establece el código y define las constante para Toggle Button; la constante `int` se establece como variable para la lectura de un canal digital, se declara el pin 4 como salida para enviar los datos de programación.

La función `setup ()` se declara `pinMode` y se define como salida para el réle.

La función `loop ()` representa el estado del interruptor del dispositivo remoto, en este proceso se repetirá varias veces con el nombre de réle.

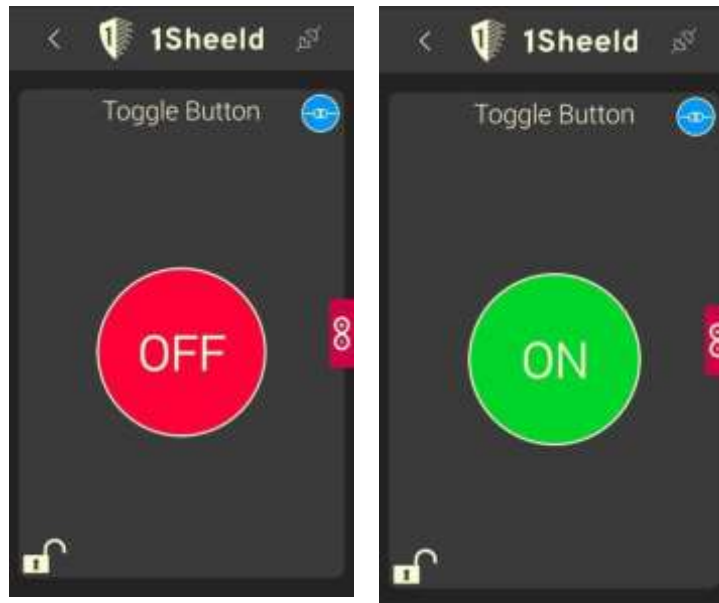


Figura 12. Visualización del Escudo Toggle Button

#### **4.3.3.4. Programación de Orientación**

```
#define CUSTOM_SETTINGS
#define INCLUDE_ORIENTATION_SENSOR_SHIELD
#include <OneSheeld.h>
int R1 = 12;
int R2 = 11;
int R3 = 10;
int R4 = 9;
int R5 = 7;
int R6 = 6;
int R7 = 5;
int R8 = 4;
```

```

void setup() {
OneSheeld.begin();
pinMode(R1,OUTPUT);
pinMode(R2,OUTPUT);
pinMode(R3,OUTPUT);
pinMode(R4,OUTPUT);
pinMode(R5,OUTPUT);
pinMode(R6,OUTPUT);
pinMode(R7,OUTPUT);
pinMode(R8,OUTPUT);
digitalWrite(R1, HIGH);
digitalWrite(R2, HIGH);
digitalWrite(R3, HIGH);
digitalWrite(R4, HIGH);
digitalWrite(R5, HIGH);
digitalWrite(R6, HIGH);
digitalWrite(R7, HIGH);
digitalWrite(R8,HIGH);
void loop(){
if(OrientationSensor.getY() > 30 && OrientationSensor.getZ() > -10 &&
OrientationSensor.getZ() < 10){
    digitalWrite(R1,LOW);
    digitalWrite(R2,LOW);
    digitalWrite(R3,HIGH);
    digitalWrite(R4,HIGH);
    digitalWrite(R5,LOW);
    digitalWrite(R6,LOW);
    digitalWrite(R7,HIGH);
    digitalWrite(R8,HIGH);
}
if(OrientationSensor.getY() < -30 && OrientationSensor.getZ() > -10 &&
OrientationSensor.getZ() < 10){
    digitalWrite(R4,LOW);
    digitalWrite(R3,LOW);

```

```

digitalWrite(R2,HIGH);
digitalWrite(R1,HIGH);
digitalWrite(R8,LOW);
digitalWrite(R7,LOW);
digitalWrite(R6,HIGH);
digitalWrite(R5,HIGH);
}
if(OrientationSensor.getZ() > 30 && OrientationSensor.getY() > -10 &&
OrientationSensor.getY() < 10){
digitalWrite(R1,LOW);
digitalWrite(R2,LOW);
digitalWrite(R3,HIGH);
digitalWrite(R4,HIGH);
digitalWrite(R5,HIGH);
digitalWrite(R6,HIGH);
digitalWrite(R7,LOW);
digitalWrite(R8,LOW);
}
if(OrientationSensor.getZ() < -30 && OrientationSensor.getY() > -10 &&
OrientationSensor.getY() < 10){
digitalWrite(R4,LOW);
digitalWrite(R3,LOW);
digitalWrite(R2,HIGH);
digitalWrite(R1,HIGH);
digitalWrite(R8,HIGH);
digitalWrite(R7,HIGH);
digitalWrite(R6,LOW);
digitalWrite(R5,LOW);
}
if(OrientationSensor.getY() > -28 && OrientationSensor.getY() < 28 &&
OrientationSensor.getZ() > -28 && OrientationSensor.getZ() < 28){
digitalWrite(R4,HIGH);
digitalWrite(R3,HIGH);
digitalWrite(R2,HIGH);

```

```
digitalWrite(R1,HIGH);  
digitalWrite(R8,HIGH);  
digitalWrite(R7,HIGH);  
digitalWrite(R6,HIGH);  
digitalWrite(R5,HIGH);  
} }
```

La librería `INCLUDE_ORIENTACION_SENSOR_SHIELD`, las siguientes líneas de código define las constantes para el LCD la contante `int` estableciendo como la lectura de un canal digital, se debe incluir los pines que se va a utilizar para enviar los datos.

La función `setup ()` se define 8 salidas digitales con `pinMode`, y 8 `digitalWrite` y se establece como valores altos (`HIGH`), inicia la comunicación con el dispositivo remoto.

La función `loop ()` lectura de la orientación de los tres ejes:

Eje x el ángulo del eje x esta entre los 30 y 10°, activa la salida como señal digital caso contrario lo desactiva.

Eje y si el ángulo del eje de las y es +/- 30 y 10° activa la salida como señal digital, caso contrario lo desactiva.

Eje z si el ángulo del eje z es mayor +/- 15° se activa la salida como señal digital caso contrario lo desactiva.



Figura 13. Visualización del Escudo de Orientación

#### 4.4. Sistema de potencia

##### 4.4.1. Motores DC

Consta de dos motores DC están ubicados a cada extremo de la estructura, con el fin de que el peso este equilibrado en ambas partes, cada motor DC opera a 12 voltios.



Figura 14. Motores DC utilizados

**Tabla 4**

*Datos Técnicos de los motores de DC*

<b>MOTOR 1</b>		<b>MOTOR 2</b>	
<b>Marca:</b>	CR-ZD1897A	<b>Marca:</b>	CR-ZD1897A
<b>Serie:</b>	Wef4t4cgsf9873	<b>Serie:</b>	Itk5693kt9321b56
<b>Voltaje:</b>	12 de DC	<b>Voltaje:</b>	12v de DC
<b>Corriente:</b>	12v de DC	<b>Corriente:</b>	12v de DC
<b>Potencia:</b>	280W	<b>Potencia:</b>	280W

Fuente: Elaboración propia

**Tabla 5**

*Corrientes y voltajes en vacío*

<b>Voltaje1</b>	(-)	(+)	<b>Corriente1</b>	<b>Voltaje2</b>	(-)	(+)	<b>Corriente2</b>
<b>12 DC</b>	12 DC		0.33A	<b>12 DC</b>	12 DC		0.23A

Fuente: Elaboración propia

**Tabla 6**

*Corrientes y voltajes con carga*

<b>Voltaje1</b>	(-)	(+)	<b>Corriente1</b>	<b>Voltaje2</b>	(-)	(+)	<b>Corriente2</b>
<b>12 DC</b>	11 DC		41A	<b>12 DC</b>	12 DC		40A

Fuente: Elaboración propia

#### **4.4.2. Construcción de partes.**

#### **4.4.3. Acoplamiento de los motores reductores y baterías.**

El propósito de colocar los motores reductores y las baterías a cada extremo de la estructura tiene como finalidad de igualar el peso a cada lado del sistema mecánico.



*Figura 15. Motores reductores y baterías.*

La chumacera es utilizada para el soporte de ejes giratorios, generalmente se utilizan dos chumaceras para cada extremo de un eje.

En el centro de la chumacera se aloja un rodamiento de bolas, su armadura es de acero el cual posee agujeros en los extremos para la sujeción a cualquier base.



*Figura 16.* Chumaceras P204

El eje mantiene la posición del neumático, en los extremos del eje se usarán dos chumaceras sujetas a la base del sistema para realizar el giro; del lado izquierdo del eje va acoplado el piñón de diámetro 100mm, se utilizará una cadena para conectar al piñón con el eje del motor de diámetro 50mm.



*Figura 17.* Eje- piñón y chumaceras

#### **4.4.4. Regulador de voltaje**

Se diseñó un circuito electrónico para regular el voltaje de alimentación de Arduino Uno y 1Shield, que funcionará con 8v, cuyo voltaje se ajustará dependiendo el valor deseado por el usuario; el regulador variable LM817T controlará una corriente máxima de 1.5 amperios, el exceso de potencia eléctrica se disipará en forma de calor por efecto Joule.



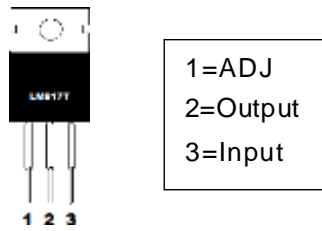


Figura 18. Regulador de voltaje

Materiales para la realización del circuito:

- LM317T
- Potenciómetro 10K
- Resistencia 470 ohms
- Fuente 12Vdc
- LED rojo 5mm
- Resistencia 1K
- Condensador 0,1 uf 16V
- Condensador 1 uf 16V

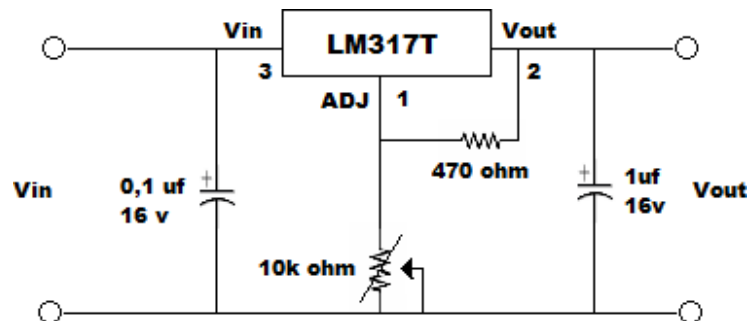


Figura 19. Diagrama del circuito regulador de voltaje

El diseño de este circuito, se realizó en una placa de baquelita (figura N° 20) y será utilizado para la alimentación de las tarjetas electrónicas, mediante este circuito se regula el voltaje de 12V hasta 8V, el transistor LM317T es un regulador de voltaje variable, el condensador de 0,1 uf a 16v va conectada a la entrada del voltaje, las dos resistencias es la relación de la variación del potenciómetro con respecto a la resistencia y a la salida un condensador de 1 uf a 16v.

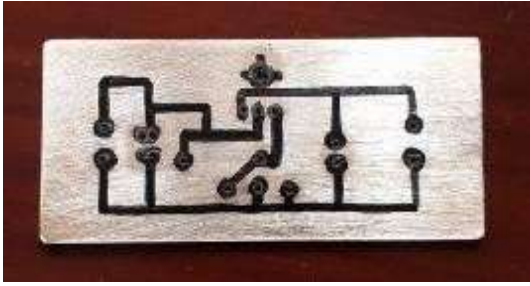


Figura 20. Diseño del circuito en baquelita



Figura 21. Circuito completo

#### 4.4.5. Inversión de giro

La inversión de giro del motor DC se logra invirtiendo la polaridad de la conexión eléctrica del campo magnético.

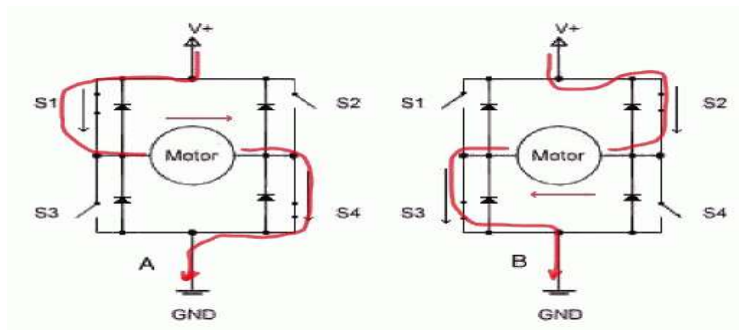


Figura 22. Puente H

Los relés funcionan como botones mecánicos accionados eléctricamente, es decir en un sistema eléctrico que a su vez acciona otro sistema eléctrico; este mecanismo permite controlar voltajes altos, a partir de voltajes muchos menores.

La alimentación eléctrica de los relés está controlada por solo dos pulsos emitidos por Arduino Uno y 1Shield estos permiten accionar los relés por parejas, el pulso 1 alimenta los relés R1 y R4 y el pulso 2, R2 y R3, además se puede observar que el circuito que controla los relés pueden ser completamente distinto del que alimenta al motor.

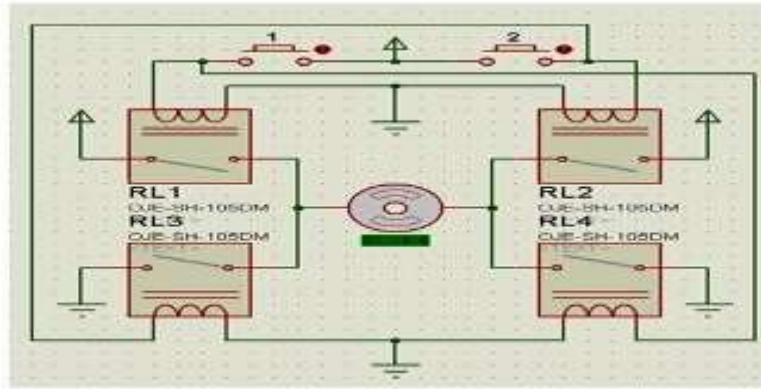


Figura 23. Puente H con relés

#### 4.4.6. Distribución del cableado

En la distribución del cableado se utilizara cable AWG # 14 para la conexión del sistema de potencia y para el circuito eléctrico de inversión de giro de los motores DC.

En la parte del control se utilizará cable AWG # 16 para conectar la alimentación de las baterías con el sistema eléctrico, se colocará canaleta de 1cm de grosor para, el cableado del sistema eléctrico de control y de potencia.

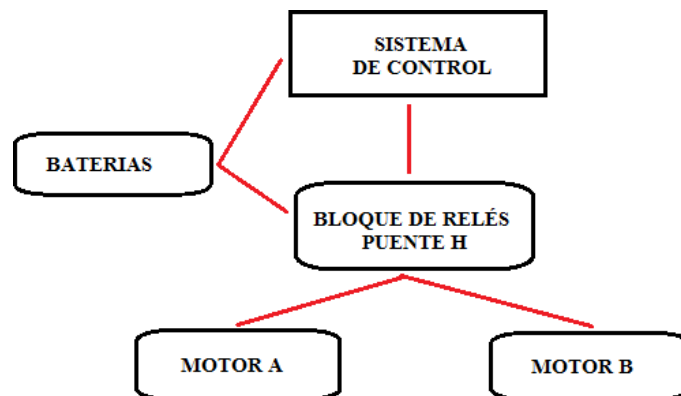


Figura 24. Distribución del cableado por secciones

#### 4.4.7. Alimentación

El sistema contiene una fuente de energía la cual suministra 12v a través de dos baterías sellada de plomo recargable, cada una de 12voltios para el motor A el mismo método para el motor B.

El mecanismo tendrá la disponibilidad y opción de utilizar las baterías de la siguiente forma en la (figura N° 25) se muestra la forma de conexión de la batería; una batería independiente para el motor A y el motor B.

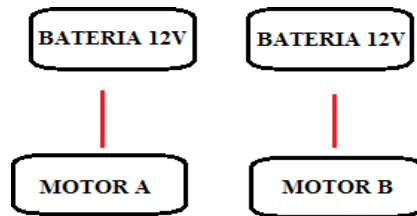


Figura 25. Baterías de 12v para cada motor

En la (figura N°26) se presenta otra manera de conexión de las baterías, dos baterías de 6v conectadas en serie obteniendo 12v voltios para cada motor.

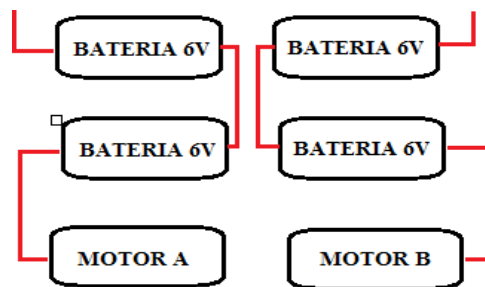


Figura 26. Baterías de 6v conectados en serie para cada motor

#### 4. CONCLUSIONES

- Se diseñó un sistema eléctrico y electrónico obteniendo como resultado la plataforma móvil y así adaptándose a su entorno, por medio de una comunicación inalámbrica, vía bluetooth con dispositivo Smartphone y 1SHEELD.
- La implementación de la pantalla TFT (HMI) en el sistema con la finalidad de crear interfaces gráficas en el display y controlar salidas digitales mediante Arduino Uno, para la interacción con el usuario en la cual no es necesario que sea un panel touch.
- La utilización del módulo de relés de Arduino Uno, facilita el funcionamiento al activar y desactivar los contactos mediante una señal de pulsos eléctricos con una intensidad mucho menor que la intensidad que va a consumir, siendo un control para voltajes elevados.
- Para aumentar la fuerza de tracción del mecanismo se utilizó dos motores reductores de 12 voltios conectados por medio de una cadena, al eje de la caja reductora y al eje del neumático para su movilidad, reduciendo su velocidad al 25% y aumentando la fuerza a un 75% de su totalidad.
- Siendo un sistema inalámbrico brinda grandes ventajas; ya que se elimina el exceso de cables, aumentando la velocidad de transmisión y recepción de datos; es más eficiente en comparación con otros sistemas de comunicación.
- Se culminó con éxito el objetivo principal del funcionamiento del sistema mecánico; con la implementación de la tarjeta 1SHEELD y Arduino Uno, se pudo reducir los costos por cada sensor utilizado.
- La innovación de nuevos sistemas electrónicos de control, será el inicio para realizar mejoras en el mecanismo.

## 5. RECOMENDACIONES

- Para controlar el sistema mecánico se recomienda que el dispositivo Smartphone posea una versión de Android de 4.1.2, en adelante.
- Ejecutar en el dispositivo Smartphone la aplicación “1SHEELD” emparejar vía bluetooth el dispositivo con la tarjeta 1Sheeld y seleccionar las SHEELD según la programación del usuario.
- Se debe desconectar el sistema eléctrico, cuando el usuario deje de utilizar el mecanismo; así no se descargarán las baterías.
- Verificar que las baterías estén cargadas antes de que el usuario controle el mecanismo.
- Se recomienda cargar las baterías por un periodo de 4 a 8 horas.
- Evitar el ingreso de agua al sistema eléctrico y electrónico, ya que puede causar cortocircuito dañando así todo el sistema mecánico.
- Se recomienda no exceder el peso por el usuario, porque el sistema realizaría excesivo esfuerzo y puede quemar los motores DC.

## 6. FUENTES

### 7.1.1. BIBLIOGRAFÍA

Gutierrez, J. M. (26 de Mayo de 2000). *Manual de usuario de arduino*. Obtenido de <https://arduinoobot.pbworks.com/f/Manual%2BProgramacion%2BArduino.pdf>

Herrador, R. E. (28 de Abril de 2009). *Guía De Usuario De Arduino*. Obtenido de [http://www.uco.es/aulasoftwarelibre/wpcontent/uploads/2010/05/Arduino\\_user\\_manual\\_es.pdf](http://www.uco.es/aulasoftwarelibre/wpcontent/uploads/2010/05/Arduino_user_manual_es.pdf)

Monasterio-Huelin, F. (06 de Octubre de 2016). *Motor DC, etapa de potencia y PWM*. Obtenido de Motor DC, etapa de potencia y PWM: <http://www.robolabo.etsit.upm.es/asignaturas/seco/apuntes/aplicacion-MotorDC.pdf>

PEIROTEN, R. A. (28 de Julio de 2013). *Sistema domótico de una casa inteligente*. Obtenido de <https://www.iit.comillas.edu/pfc/resumenes/51b998731bb57.pdf>

Ebel. (2008). *Libro técnico Festo*. Mexico: Fonseca.

Santana, J. A., Sierra, J. M., & Bravo, J. A. (2013). *Automatismos Industriales*. Mexico: Alfaomega.

Andrès, A.A (2016). *Diseño e implementación de un sistema*. sangolqui: San José  
Primera edición.

Bolton, W. (2013). *MECATRÓNICA*. Obtenido de <https://docs.google.com/file/d/0B750YZM-uweRZm1sWEI5LVREVDQ/edit>  
Mexico: Alfaomega.

Coughlin F.Driscoll (1993). *Amplificadores operacionales y circuitos integrados lineales*. Prentice-Hall Hispanoamericana, S.A.

Manuel T.(2000). *Circuitos eléctricos* MEXICO: NEMA

## 8. ANEXOS

### 8.1. Código de programación para mover adelante, atrás y giro.

```
#define CUSTOM_SETTINGS
#define INCLUDE_ORIENTATION_SENSOR_SHIELD
#include <OneSheeld.h>
int R1 = 12;
int R2 = 11;
int R3 = 10;
int R4 = 9;
int R5 = 7;
int R6 = 6;
int R7 = 5;
int R8 = 4;
void setup()
{
  OneSheeld.begin();
  pinMode(R1,OUTPUT);
  pinMode(R2,OUTPUT);
  pinMode(R3,OUTPUT);
  pinMode(R4,OUTPUT);
  pinMode(R5,OUTPUT);
  pinMode(R6,OUTPUT);
  pinMode(R7,OUTPUT);
  pinMode(R8,OUTPUT);
  digitalWrite(R1, HIGH);
  digitalWrite(R2, HIGH);
  digitalWrite(R3, HIGH);
  digitalWrite(R4, HIGH);
  digitalWrite(R5, HIGH);
  digitalWrite(R6, HIGH);
  digitalWrite(R7, HIGH);
  digitalWrite(R8,HIGH);
```



```

void loop(){
if(OrientationSensor.getY() > 30 && OrientationSensor.getZ() > -10 &&
OrientationSensor.getZ() < 10){
    digitalWrite(R1,LOW);
    digitalWrite(R2,LOW);
    digitalWrite(R3,HIGH);
    digitalWrite(R4,HIGH);
    digitalWrite(R5,LOW);
    digitalWrite(R6,LOW);
    digitalWrite(R7,HIGH);
    digitalWrite(R8,HIGH);
}
if(OrientationSensor.getY() < -30 && OrientationSensor.getZ() > -10 &&
OrientationSensor.getZ() < 10){
    digitalWrite(R4,LOW);
    digitalWrite(R3,LOW);
    digitalWrite(R2,HIGH);
    digitalWrite(R1,HIGH);
    digitalWrite(R8,LOW);
    digitalWrite(R7,LOW);
    digitalWrite(R6,HIGH);
    digitalWrite(R5,HIGH);
}
if(OrientationSensor.getZ() > 30 && OrientationSensor.getY() > -10 &&
OrientationSensor.getY() < 10){
    digitalWrite(R1,LOW);
    digitalWrite(R2,LOW);
    digitalWrite(R3,HIGH);
    digitalWrite(R4,HIGH);
    digitalWrite(R5,HIGH);
    digitalWrite(R6,HIGH);
    digitalWrite(R7,LOW);
}

```

```

    digitalWrite(R8,LOW);
}
if(OrientationSensor.getZ() < -30 && OrientationSensor.getY() > -10 &&
OrientationSensor.getY() < 10){
    digitalWrite(R4,LOW);
    digitalWrite(R3,LOW);
    digitalWrite(R2,HIGH);
    digitalWrite(R1,HIGH);
    digitalWrite(R8,HIGH);
    digitalWrite(R7,HIGH);
    digitalWrite(R6,LOW);
    digitalWrite(R5,LOW);
}
if(OrientationSensor.getY() > -28 && OrientationSensor.getY() < 28 &&
OrientationSensor.getZ() > -28 && OrientationSensor.getZ() < 28){
    digitalWrite(R4,HIGH);
    digitalWrite(R3,HIGH);
    digitalWrite(R2,HIGH);
    digitalWrite(R1,HIGH);
    digitalWrite(R8,HIGH);
    digitalWrite(R7,HIGH);
    digitalWrite(R6,HIGH);
    digitalWrite(R5,HIGH);
}
}
}

```

### 8.1.2. Programación para encender una tira de Led con relé

```

#define CUSTOM_SETTINGS
#define INCLUDE_TOGGLE_BUTTON_SHIELD
#include <OneSheeld.h>
byte ledBlanco = 4;
voidsetup()

```

```

{
  pinMode(ledBlanco,OUTPUT);
  OneSheeld.begin();
}

void loop()
{
  digitalWrite(ledBlanco,ToggleButton.getStatus());
}

```

### **8.1.3. Programación con datos específicos**

```

#define CUSTOM_SETTINGS
#define INCLUDE_LCD_SHIELD
#include <OneSheeld.h>
int An;
void setup()
{
  OneSheeld.begin();
  LCD.begin();
  LCD.clear();
}
void loop()
{
  LCD.home();
  An=analogRead(A0);
  LCD.print(" PROYECTO TESIS MARCO REINOSO ");
}

```

#### 8.1.4. Comprobación de la programación



*Figura 27. Comprobación del prototipo*

#### 8.1.5. Construcción de eje en el torno



*Figura 28. Construcción del eje para la llanta*

#### 8.1.6. Elaboración de chavetero en el eje de la llanta



*Figura 29. Construcción de chavetero en eje*

### 8.1.7. Acople del piñón en el eje del motor



*Figura 30 .Piñón en el eje del motor*

### 8.1.7. Acople de piñón en el eje de la llanta



*Figura 31. Acople del piñón en el eje*

### 8.1.8. Ensamblaje de piezas vista preliminar



*Figura 32. Vista preliminar*

### 8.1.9. Base de tubo de 1"1/4



*Figura 33.* Construcción de base con tubo de 1"1/4

### 8.1.10. Base para alojar las chumaceras



*Figura 34.* Construcción de la base para las chumaceras

### 8.1.11. Fijación de las llantas en la base



*Figura 35.* Fijación de los ejes en chavetero

### 8.1.12. Ensamblaje de los motores y llantas



*Figura 36.* Montaje en la base



*Figura 37.* Proyecto finalizado