

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería



Diseño e implementación de un sistema electrónico y un mando inalámbrico a partir de comunicación Bluetooth para el control de una silla de ruedas

Trabajo de graduación en modalidad de Tesis presentado por:

Juan Carlos Jiménez Hoffens

para optar al grado académico de Licenciado en Ingeniería Electrónica

Guatemala

2018



Diseño e implementación de un sistema electrónico y un mando inalámbrico a partir de comunicación Bluetooth para el control de una silla de ruedas

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería



Diseño e implementación de un sistema electrónico y un mando inalámbrico a partir de comunicación Bluetooth para el control de una silla de ruedas

Trabajo de graduación en modalidad de Tesis presentado por:

Juan Carlos Jiménez Hoffens

para optar al grado académico de Licenciado en Ingeniería Electrónica

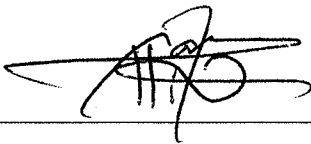
Guatemala

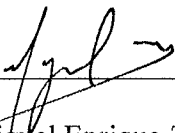
2018

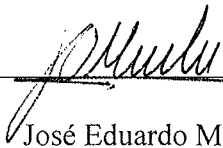
Vo. Bo.:

(f)   
\_\_\_\_\_  
(Pablo Roberto Oliva Fonseca)

Tribunal Examinador:

(f)   
\_\_\_\_\_  
Carlos Alberto Esquit Hernández

(f)   
\_\_\_\_\_  
Miguel Enrique Zea Arenales

(f)   
\_\_\_\_\_  
José Eduardo Morales Espinoza

Fecha de aprobación: Guatemala 20 de junio de 2018!

# ÍNDICE

<b>LISTA DE CUADROS .....</b>	<b>VI</b>
<b>LISTA DE FIGURAS .....</b>	<b>VII</b>
<b>RESUMEN.....</b>	<b>IX</b>
<b>I. INTRODUCCIÓN.....</b>	<b>1</b>
A. OBJETIVOS .....	2
B. JUSTIFICACIÓN .....	3
C. MARCO TEÓRICO .....	4
D. ANTECEDENTES.....	9
E. METODOLOGÍA .....	10
F. RESULTADOS.....	19
<b>III. CONCLUSIONES.....</b>	<b>33</b>
<b>IV. RECOMENDACIONES .....</b>	<b>34</b>
<b>V. BIBLIOGRAFÍA.....</b>	<b>35</b>
<b>VI. ANEXOS .....</b>	<b>36</b>

## LISTA DE CUADROS

Cuadro 1. Comparativa de nivel de voltaje, digitalización y duty cycle (PWM).....	12
Cuadro 2. Modos de operación y su dirección de selección. ....	14
Cuadro 3. Configuraciones especiales modo “Analog Input” .....	15
Cuadro 4. Listado de comandos AT y su descripción.....	17
Cuadro 5. Estados de las entradas para la tarjeta de control. ....	19
Cuadro 6. Componentes utilizados para el circuito generador de señal de control motor DC.....	20
Cuadro 7. Componentes utilizados para circuito de retardo a la conexión. ....	24
Cuadro 8. Valores numéricos iniciales y finales de simulación load switch. ....	28

## LISTA DE FIGURAS

Figura 1. Comportamiento de carga y corriente en un circuito RC. ....	5
Figura 2. Circuito RC en estado inicial y final. ....	5
Figura 3. Configuración de descarga de un capacitor. ....	6
Figura 4. Comparativa carga y descarga de un circuito RC. ....	6
Figura 5. Load Switch Circuit, partes principales. ....	7
Figura 6. MOSFET, tipo de canales y sus terminales. ....	7
Figura 7. Load Switch, transistor de paso tipo N. ....	8
Figura 8. Load Switch, transistor de paso tipo P. ....	9
Figura 9. Software online Onshape. ....	11
Figura 10. Joystick. ....	11
Figura 11. Adafruit Feather M0. ....	12
Figura 12. Circuito filtro RC para PWM. ....	13
Figura 13. Circuito amplificador no inversor. ....	13
Figura 14. Controlador Sabertooth 2x12. ....	14
Figura 15. Configuración final modo de operación “Analog Input” ....	15
Figura 16. Circuito de retardo de alimentación. ....	16
Figura 17. Módulo Bluetooth HC-05. ....	16
Figura 18. Configuración maestro-esclavo comunicación Bluetooth. ....	17
Figura 19. Controlador motor DC Brushless. ....	18
Figura 20. Conector J2, entradas y salidas. ....	18
Figura 21. Circuito generador de señal de control motor DC. ....	19
Figura 22. Implementación función de transferencia filtro pasa bajas. ....	20
Figura 23. Barrido DC señal PWM filtrada. ....	21
Figura 24. Datos numéricos medición de cursores. ....	21

Figura 25. Circuito de retardo a la conexión para controladores de motores DC. ....	22
Figura 26. Circuito de conexión al retardo. ....	23
Figura 27. Circuito interruptor de carga (load switch) para la etapa de potencia. ....	24
Figura 28. Simulación de circuito Load Switch en Matlab para análisis. ....	25
Figura 29. Indexado de datos para transistor mosfet tipo-P. ....	26
Figura 30. Indexado de datos para transistor mosfet tipo-N. ....	26
Figura 31. Gráfica de comportamiento mosfet tipo-P con valor mínimo de $V_{th}$ . ....	27
Figura 32. Gráfica de comportamiento mosfet tipo-P con valor máximo de $V_{th}$ . ....	27
Figura 33. Diseño Botton Layer PCB principal. ....	29
Figura 34. Diseño Top Layer PCB principal. ....	30
Figura 35. Diseño PCB Mando inalámbrico. ....	31
Figura 36. Mando inalámbrico montaje interno. ....	31
Figura 37. Mando inalámbrico apariencia externa. ....	32

## RESUMEN

En el año 2012, se realizó el megaproyecto “Profectus” el cual consistía en una silla de ruedas autónoma, con la capacidad de movilizar a un usuario y además esta tendría la capacidad de escalar gradas. El megaproyecto se encontraba dado de baja. El objetivo del proyecto que se presentara a continuación es el diseño e implementación de un nuevo sistema electrónico que controle y reemplace el anterior. El control se realizara utilizando los mecanismos ya disponibles de la silla.

Para ello se implementó un sistema a partir de un controlador “Adafruit Feather M0”, el cual recibe instrucciones de un mando inalámbrico que se comunica mediante comunicación Bluetooth implementado a partir dos módulos “HC-05” configurados como maestro/esclavo. El controlador toma las instrucciones y genera señales PWM, que posteriormente son acondicionadas para generar una señal analógica que varía en un rango de voltaje entre 0 y 5. Esta señal es utilizada por los controladores que accionan los motores que producen movimiento. Se implementó un sistema de retardo a la conexión para los controladores, con la finalidad de preservar la integridad del usuario al encender la silla, previniendo movimientos involuntarios.

Se llegó a la determinación de que el mando inalámbrico supone una gran versatilidad y comodidad para el usuario al momento de su uso, además de ser un mando muy intuitivo para quien lo utiliza. Adicionalmente se recomienda para futuras optimizaciones, la realización de un sistema de control para el enlace mecánico de los triángulos y para la aplicación del escalado de gradas.

# I. INTRODUCCIÓN

El presente trabajo de graduación consiste en diseñar e implementar un sistema electrónico capaz de controlar una silla de ruedas por medio de un mando inalámbrico mediante comunicación Bluetooth, el cual permite el movimiento de traslación y escalar gradas a la misma. Dicho sistema fue implementado en la silla de ruedas construida para el megaproyecto Profectus 2012. Para este proyecto se utilizó la estructura mecánica construida en el megaproyecto antes mencionado, y se diseñó un sistema electrónico con capacidad para controlar el sistema mecánico de la silla de ruedas.

Para el sistema electrónico se diseñó e implementó un PCB, este contiene los componentes electrónicos necesarios para el funcionamiento de la silla de ruedas. Su funcionamiento se reduce a recibir comandos de un mando inalámbrico, interpretarlos y convertirlos en señales eléctricas requeridas por los controladores de los actuadores, para que estos ejecuten el movimiento solicitado por el usuario.

El mando inalámbrico supone ser una interfaz intuitiva entre el usuario y la silla de ruedas. Este fue diseñado por medio de software de diseño CAD y fabricado en la UVG mediante corte laser e impresión 3D. La vía de comunicación de dicho mando con la tarjeta principal de la silla es mediante Bluetooth, implementada mediante dos módulos HC-05 configurados como maestro-esclavo. Uno de ellos que cumple la función de esclavo se encuentra en el interior del mando y el otro que cumple la función de maestro se encuentra en la tarjeta principal en la silla de ruedas.

El sistema de comunicación inalámbrica mediante Bluetooth, le brinda versatilidad y comodidad al proyecto. Además el mando construido a partir de un joystick, lo hace muy intuitivo y amigable para el usuario, ya que este no supone nada de complejidad al momento de controlar la silla de ruedas.

## II. CUERPO

### A. OBJETIVOS

#### 1. General.

Diseñar e implementar un sistema electrónico que permita el movimiento de traslación sobre el suelo y escalar gradas de la silla de ruedas (Profectus), comandado a partir de un mando inalámbrico.

#### 2. Específicos.

- a. Diseñar y construir un mando inalámbrico a partir de un joystick, implementando comunicación Bluetooth.
- b. Diseñar e implementar un único PCB, que integre los componentes requeridos para el funcionamiento del sistema.

## B. JUSTIFICACIÓN

La principal motivación del proyecto supone dar a conocer las habilidades, capacidades y conocimientos obtenidos a lo largo de la carrera, pero dándole un sentido práctico mediante la realización de un proyecto que sea beneficioso para el cuerpo docente y estudiantil de la Universidad del Valle de las carreras de Electrónica y Mecatrónica. Ya que con él se puede dar a conocer a los jóvenes las virtudes de las carreras, además de cómo estas aportan proyectos valiosos y beneficiosos para la sociedad.

Dicho proyecto propone la reestructuración del sistema electrónico de control para la silla de ruedas implementada en el megaproyecto “Profectus”. Se diseñará el sistema con una única placa de control que integra todo el sistema electrónico para el funcionamiento de la silla de ruedas, esto supone una ventaja al sistema de alimentación del sistema, además de la facilidad de arranque y encendido del mismo.

Así mismo la silla de ruedas será controlada mediante un mando inalámbrico, comunicado al sistema a través de comunicación Bluetooth, lo que supone versatilidad y comodidad para el usuario. Se realizará esta interfaz del usuario con la silla de ruedas, de forma sencilla e intuitiva para que sea amigable al usuario y al mismo tiempo cumpla el objetivo de controlarla.

De la misma forma que se pretende atraer jóvenes estudiantes a las carreras, también se busca dar a conocer a todas las personas las posibilidades que existen dentro de la tecnología y que a través de ella se pueden crear proyectos innovadores y útiles a la sociedad.

## C. MARCO TEÓRICO

1. Circuito RC (primer orden). Dicho circuito está compuesto por resistencias y capacitores, pero en su caso más simple, el circuito RC de primer orden, está compuesto únicamente por una resistencia y un capacitor. Dicho circuito es llamado de primer orden, ya que este puede ser descrito únicamente por una ecuación diferencial.

$$i(t) = \frac{\xi}{R} \left( e^{-t/\tau} \right); \tau = RC \quad (1)$$

Donde  $i(t)$  es la corriente del circuito que depende del tiempo,  $\xi$  es el voltaje de alimentación de la fuente del voltaje,  $R$  es el valor de la resistencia,  $C$  es el valor del capacitor,  $t$  es el valor del tiempo y  $\tau$  es la constante de tiempo de carga del circuito que equivale a  $RC$ .

La ecuación (1) describe el comportamiento de la corriente en el circuito RC. Como se muestra en la Figura 2 (a), en  $t = 0$ , la corriente corresponde a  $i(t) = \frac{V_s}{R}$ , lo que indica que el Capacitor con carga cero se comporta como un cable sin resistencia. Contrario como se muestra en la Figura 2 (b), con  $t = \infty$ , la corriente  $i(t) \rightarrow 0$ , lo que nos indica que el capacitor con carga se comporta como un circuito abierto.

$$i(t) = \frac{dq}{dt} \quad (2)$$

$$\frac{dq}{dt} = \frac{\xi}{R} \left( e^{-t/RC} \right) \rightarrow \int_0^q dq = \int_0^t \frac{\xi}{R} \left( e^{-t/RC} \right) dt \quad (3)$$

$$q(t) = \xi C \left( 1 - e^{-t/\tau} \right) \quad (4)$$

$$V_c(t) = \frac{q(t)}{C} \quad (5)$$

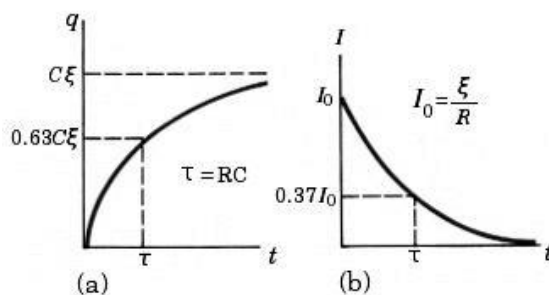
$$V_c(t) = \xi \left( 1 - e^{-t/\tau} \right) \quad (6)$$

A partir de la ecuación (2) y su sustitución en (1) se puede obtener la ecuación diferencial (3) y por medio de la operatoria de la integral obtenemos la ecuación (4). Dicha ecuación describe el comportamiento de la carga del capacitor en el circuito RC. Como se muestra en la Figura 2 (a), en  $t = 0$ , la carga del capacitor  $q(t) = 0$ . Lo que nos indica que en efecto el capacitor no tiene carga alguna. Contrario como se muestra en la Figura 2 (b), en  $t = \infty$ , la carga del capacitor  $q(t) = \xi C$ . También se puede obtener el valor del voltaje del capacitor en un tiempo específico, a partir de la sustitución de la ecuación (5) en (4) se obtiene la ecuación para el voltaje del capacitor (6).

Cabe hacer notar que en realidad  $t = \infty$  es tan solo una notación que se utiliza para denotar el tiempo final en el que el capacitor obtuvo su carga máxima. El momento en el capacitor obtiene su carga máxima es un valor de tiempo finito que se puede aproximar a  $t \approx 5\tau$  o bien  $t \approx 5RC$ .

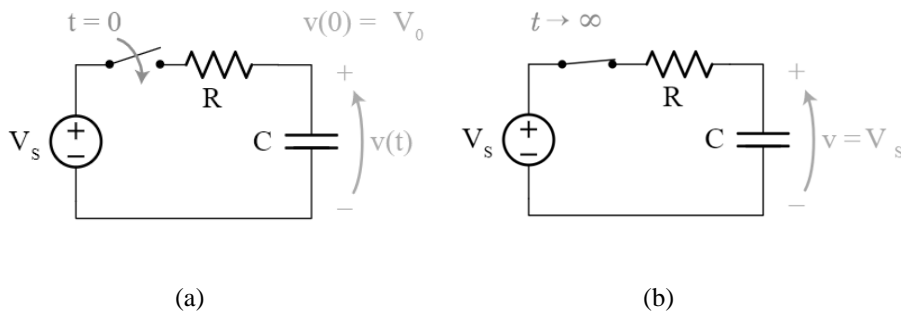
El comportamiento de la corriente y de la de la carga del capacitor se describe en la Figura 1, en ella se puede comprobar lo antes mencionado al describir la ecuación de la corriente y de la carga, ya que se observa como en  $t = 0$ , la corriente está en su valor máximo y la carga del capacitor se encuentra en cero. Contrario en el segundo caso cuando  $t = \infty$ , la corriente baja a cero y la carga del capacitor sube a su valor máximo. También se puede intuir que la ecuación tiene sentido, no solo en sus extremos, sino también en el comportamiento exponencial de su transición.

Figura 1. Comportamiento de carga y corriente en un circuito RC.



(Khan academy, 2018)

Figura 2. Circuito RC en estado inicial y final.



(Khan academy, 2018)

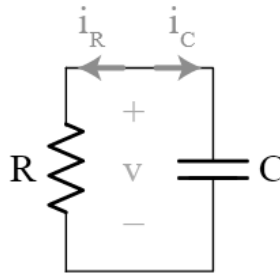
Ahora se verá la diferencia entre carga y descarga de un circuito RC, la situación se muestra en la Figura 3, en la que ya no existe una fuente que alimente el circuito, sino que ahora se utiliza la resistencia para la descarga del capacitor. La corriente dentro del circuito se comporta de igual manera. A continuación se muestra cómo difiere la carga.

$$Ri(t) - \frac{q}{c} = 0 \rightarrow \frac{dq}{dt} - \frac{q}{RC} = 0 \rightarrow \int_{\xi C}^q \frac{dq}{q} = \int_0^t \frac{dt}{RC} \quad (7)$$

$$q(t) = \xi C \left( e^{-t/\tau} \right) \quad (8)$$

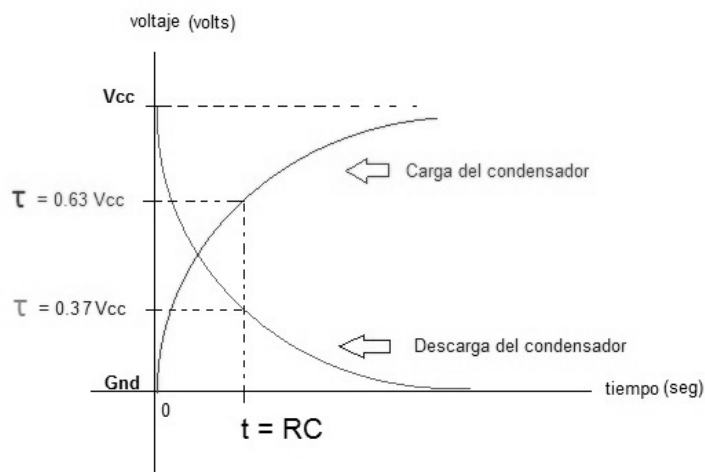
Si se observa la deducción de la expresión (7) por la ley de Kirchoff se obtiene que la diferencia de potenciales en el nodo debe ser nulo, y si se trabaja un poco la expresión, con la sustitución de la expresión (2) dentro de ecuación se obtiene una ecuación diferencial, que posteriormente se integrara para obtener la expresión de carga del capacitor (8). Si se analiza dicha expresión se nota que es contraria a la expresión (4), ya que esta es una función exponencial que decae y tiende a cero. En la Figura 4, se puede observar cómo ambas expresiones se comportan para darle sentido a lo antes mencionado.

Figura 3. Configuración de descarga de un capacitor.



(Khan academy, 2018)

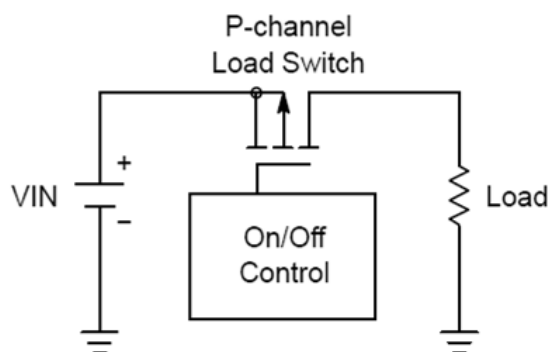
Figura 4. Comparativa carga y descarga de un circuito RC.



(Khan academy, 2018)

2. **Load Switch Circuit.** Los circuitos de interrupción de carga poseen la capacidad de conectar y desconectar una carga específica a un riel de voltaje, en otros términos se podría decir que opera como un controlador on/off, con la diferencia que opera en dos partes. Dichas partes son: un bloque de control on/off y un transistor de paso como se muestra en la Figura 5. El interruptor de carga es un medio simple de alimentación para una carga y le permite al sistema la maximización de su rendimiento.

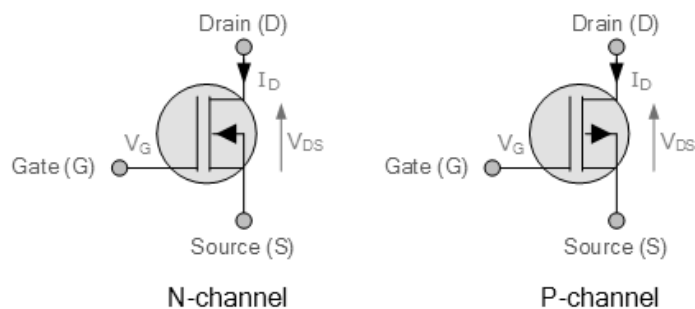
Figura 5. Load Switch Circuit, partes principales.



(ON Semiconductor, 2017)

El circuito interruptor de carga usualmente se construye utilizando transistores tipo MOSFET (Figura 6). Para la selección del tipo de canal del MOSFET a utilizar como transistor de paso, se debe tomar en cuenta la aplicación en la cual utilizaremos el circuito. El canal tipo N tiene una ventaja sobre el canal tipo P, ya que este posee mayor movilidad de electrones, provoca que la resistencia entre DRAIN y SOURCE ( $R_{DS}$ ) sea menor.

Figura 6. MOSFET, tipo de canales y sus terminales.



(Q. Deng, 2007)

Dicha ventaja radica directamente en la eficiencia del circuito, siendo esta fundamental para la administración general de energía del sistema. En el circuito interruptor de carga, la corriente exigida por la carga fluye a través del transistor de paso, donde se produce la mayor pérdida de potencia. Dicha potencia de pérdida está descrita por la expresión (9), a partir de ella se puede inferir que el valor de  $R_{DS}$  es proporcional al valor de la potencia de pérdida.

$$P_{loss} = I_{load}^2 \cdot R_{DS} \quad (9)$$

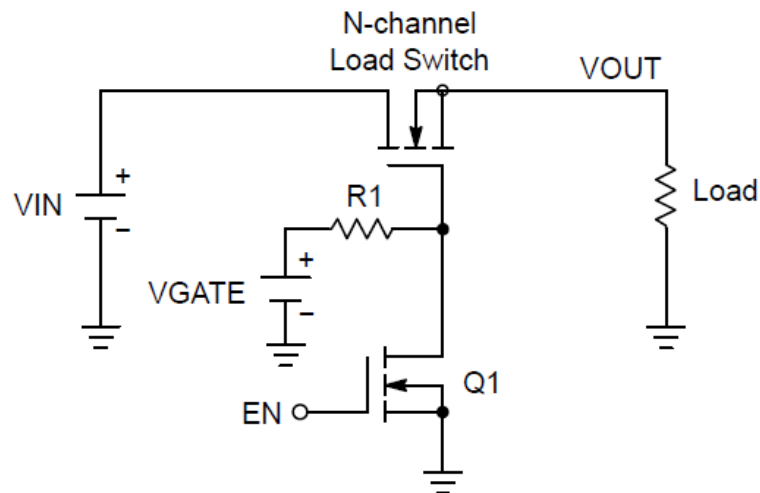
El valor de  $R_{DS}$  del transistor de paso también puede influir a la caída de voltaje entre el voltaje de entrada ( $V_{in}$ ) y el voltaje de salida ( $V_{out}$ ), causando que el voltaje salida en el transistor no sea igual al de la entrada. Dicha caída de voltaje se describe a partir de la expresión (10), de igual forma se puede inferir que la caída de voltaje es proporcional al de  $R_{DS}$  lo que provoca una disminución en el voltaje de salida.

$$V_{out} = V_{in} - (I_{load} \cdot R_{DS}) \quad (10)$$

A partir de las expresiones descritas anteriormente y dado que la resistencia ( $R_{DS}$ ) de un canal tipo N suele ser dos veces menor a la de un canal tipo P, se puede inferir que el canal tipo N brinda una ventaja para aplicaciones de potencia, donde se manejan corrientes muy altas. Y un canal tipo P, no tendría gran impacto en la potencia de pérdida y la caída de voltaje en una aplicación de baja corriente.

La construcción de un circuito interruptor de voltaje a partir de un transistor de paso utilizando un canal tipo N se describe en la Figura 7. Su operación depende de la entrada de control (EN), mientras EN permanezca en referencia de señal baja, es decir 0V (tierra) el transistor de paso estará encendido, es decir permitirá el paso de voltaje de  $V_{in}$  a  $V_{out}$ . Y mientras EN permanezca en referencia de señal alta (V+), el transistor de paso estará apagado, es decir no permitirá el paso del voltaje.

Figura 7. Load Switch, transistor de paso tipo N.





Con respecto a la forma de controlar la silla de ruedas, esta empleaba comandos de voz y señales de electromiografía para el control y el movimiento del sistema. Pero este sistema tenía la desventaja que requería de una serie de calibraciones de voz y de la señales de electromiografía con el usuario. Esta era una desventaja, ya que cada vez que el sistema reiniciara se requería de su calibración o cada vez que se cambiara de usuario.

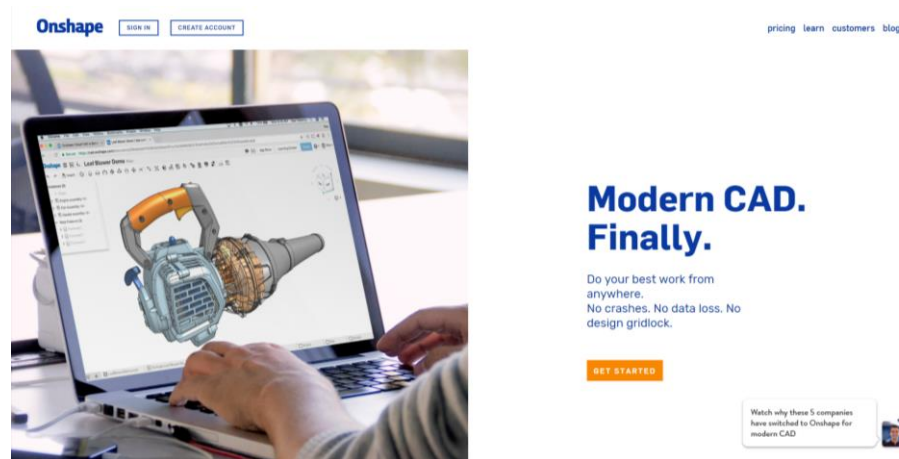
## E. METODOLOGÍA

Para el control de la silla de ruedas se diseñó un sistema sencillo en el que a partir de un joystick se envían señales a un controlador el cual interpreta, procesa y convierte en cuatro señales digitales PWM por medio de sus puertos de salida digital. Luego dichas salidas digitales se filtran a partir de un filtro de primer orden RC, esta es filtrada con la finalidad de obtener la componente DC pura del tren de pulsos generado en el PWM. Una vez filtrada las señales pasan a ser señales analógicas que varía entre 0 - 3.3V, estas varían dependiendo de la posición del joystick. Estas señales analógicas son acondicionadas por medio de un amplificador no inversor, para que sus parámetros de variación sean entre 0 - 5V. La etapa de acondicionamiento es necesaria ya que las señales son utilizadas por los controladores de los motores DC que controlan el movimiento lineal de la silla, requieren de una señal de entrada analógica que varíe dentro de dichos parámetros.

Para la comunicación inalámbrica se utilizó un controlador que recibe las señales analógicas del joystick del mando, este las interpreta, codifica y envía mediante comunicación serial a un módulo Bluetooth HC-05. Dicho módulo está configurado como esclavo, con la finalidad de que todo el tiempo envíe el estatus del mando inalámbrico a otro módulo, configurado como maestro. Dicho módulo maestro está localizado en la tarjeta principal de la silla y envía, mediante comunicación serial al controlador de la tarjeta principal de la silla el estatus del mando inalámbrico, para que este realice lo antes mencionado.

Para el diseño 3D del mando inalámbrico se utilizó software especializado en diseño CAD. Para las piezas del interior del mando se utilizó el software online llamado Onshape (Figura 7) y para la carcasa del mando se utilizó AutoCAD. En el caso de Onshape se modeló piezas 3D que posteriormente se manufacturaron en la impresora 3D con PLA y para el caso de AutoCAD se realizó el modelaje de un plano, que posteriormente se manufacturó en la cortadora laser con MDF.

Figura 9. Software online Onshape.



(Onshape, 2017)

El mando inalámbrico se diseñó de forma sencilla para que fuera intuitivo para el usuario, es decir no tenga ningún problema en entender cómo funciona y que pueda controlarlo. Para ello se escogió un joystick (Figura 8), y que este controle la silla en un plano de dos dimensiones, con coordenadas  $(x,y)$ . Siento “x” el movimiento horizontal, es decir el que nos permite virar, ya sea a la izquierda o a la derecha. Y siendo “y” el movimiento vertical, es decir el que nos permite, avanzar o retroceder.

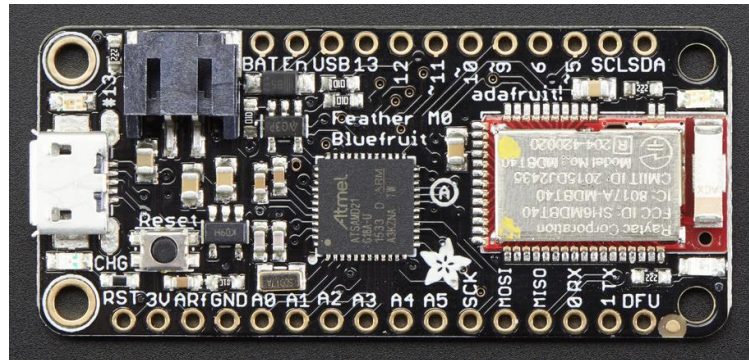
Figura 10. Joystick.



(Adafruit, 2016)

Para el procesamiento de las señales y el control del sistema se realizó utilizando un controlador Adafruit Feather M0 (Figura 9). Se utilizó dicho controlador con la finalidad de economizar tiempo de procesamiento, ya que este es un procesador de alta velocidad que maneja una velocidad de reloj de 48 MHz. También se buscaba un controlador que brindara una alta disponibilidad de puertos para salidas digitales y que estos tuvieran la capacidad de utilizar PWM. En total el Feather M0 posee 10 puertos de salidas digitales y 8 de ellos tienen la capacidad de generar PWM. Pero una de sus principales ventajas radica en que es un controlador de bajo consumo energético.

Figura 11. Adafruit Feather M0.



(Adafruit, 2016)

El controlador recibe la instrucción del mando inalámbrico, pero la instrucción recibida es la digitalización de un nivel de voltaje entre 0 - 3.3V. Esta información digitalizada en un byte de información es procesada por el controlador y este genera una señal de PWM. Dicha señal varía su ciclo de trabajo según el nivel de voltaje recibido en el controlador. En el Cuadro 1, se muestra como a partir de un nivel de voltaje el ciclo de trabajo del PWM varía.

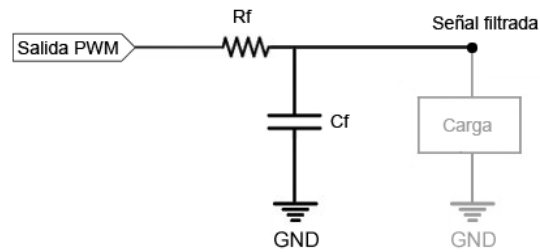
Cuadro 1. Comparativa de nivel de voltaje, digitalización y Duty Cycle (PWM).

Nivel de Voltaje	Valor de Señal digitalizada (Byte)	Duty Cycle (PWM)
0 V	0	0 %
1.65 V	128	50%
3.3 V	255	100%

La señal PWM es filtrada a partir de un filtro de primer orden RC como se muestra en la Figura 10, con la finalidad de obtener la componente DC del PWM, es decir obtener nuevamente el nivel de voltaje que varía entre 0 - 3.3V. Además de filtrar la señal, está pasa por una etapa de amplificación para ello se utilizó un amplificador no inversor como se muestra en la Figura 11. Se diseñó con una ganancia de amplificación de 1.5, a partir de la expresión (13) para que el nivel de voltaje ahora varié entre 0 - 5V.

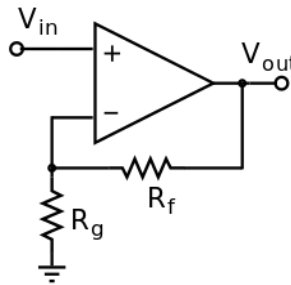
$$V_{out} = G \cdot V_{in} \leftrightarrow G = 1 + \frac{R_f}{R_g} \quad (13)$$

Figura 12. Circuito filtro RC para PWM.



(Khan academy, 2018)

Figura 13. Circuito amplificador no inversor.



(Khan academy, 2018)

Los encargados del movimiento lineal de la silla, es decir su traslación, son cuatro motores DC de 12V de operación. Estos motores son controlados por dos tarjetas de control Sabertooth 2x12 (Figura 12), como su nombre lo indica cada tarjeta tiene la capacidad de operar dos motores a 12A como corriente máxima en operación continua. Además pueden operar en voltajes de entre 6 - 24V y soporta picos de corriente de hasta 25A. También poseen cuatro modos de operación como se muestra en el Cuadro 2, y el modo de operación utilizado es el de “Analog Input”.

Figura 14. Controlador Sabertooth 2x12.



(Dimension Engineering, 2012)

Para seleccionar el modo de operación se utiliza el Dip-Switch localizado en la tarjeta como se muestra en la Figura 12. Para elegir el modo de operación se utilizan los primeros de selectores S1 y S2, como se muestra en el Cuadro 2. Se utiliza el modo de operación de entrada analógica, ya que el proceso de las señales anteriormente descrito, fue diseñado para poder aprovechar dicho modo de operación. Modo de operación que funciona de modo muy sencillo, pero muy preciso para la aplicación que realizamos.

Cuadro 2. Modos de operación y su dirección de selección.

Modos de operación	Dip-Switch S1	Dip-Switch S2
Analog Input	ON	ON
R/C Input	OFF	ON
Simplified Serial	ON	OFF
Packetized Serial	OFF	OFF

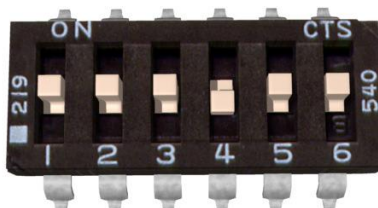
Además los siguientes 4 selectores sirven para configurar dentro del modo de operación algunas configuraciones y funcionamientos especiales. En el Cuadro 3 se describen las configuraciones especiales para el modo de operación de “Analog Input”. El S3 permite seleccionar si se utilizará una batería de litio o de cualquier otro tipo. El S4 indica si se utilizará las entradas analógicas de forma mixta, es decir una de ellas

para el control de la dirección y otra para la velocidad de ambos motores o si se utilizará de forma convencional, es decir cada entrada funciona para cada motor. El S5 nos indica si la respuesta de la velocidad a la señal analógica es lineal o exponencial. El S6 permite elegir la sensibilidad de operación, es decir si se quiere usar un rango de voltaje de 0 - 5V, siendo 2.5V el punto cero de operación o bien utilizar el rango de voltaje de 1.875 – 3.125V con el mismo punto cero de operación. Finalmente la configuración utilizada se describe en la Figura 13.

Cuadro 3. Configuraciones especiales modo “Analog Input”.

No. de Switch	Switch “ON”	Switch “OFF”
Switch 3	Batería de litio	Batería convencional
Switch 4	Modo mixto	Modo independiente
Switch 5	Respuesta lineal	Respuesta exponencial
Switch 6	Sensibilidad normal	Sensibilidad x4

Figura 15. Configuración final modo de operación “Analog Input”

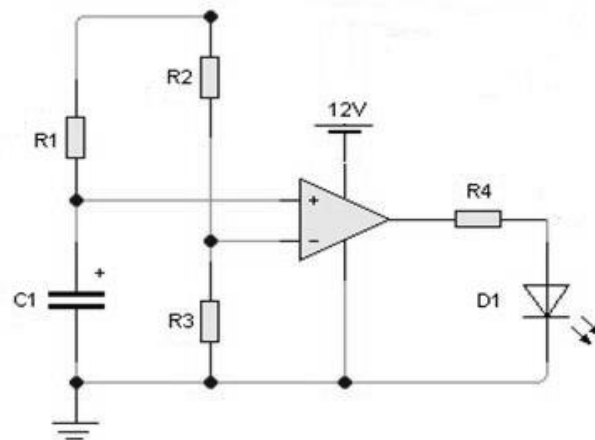


(Dimension Engineering, 2012)

Como medida de seguridad del sistema para preservar la integridad del usuario, se diseñó un sistema de protección al encendido de la silla, debido a que al encender el sistema tarda algunos segundos que el controlador inicialice sus configuraciones y rutinas. En ese lapso de tiempo los motores no reciben señal alguna del controlador y estos se accionan, provocando movimiento involuntario. Para ello se diseñó un sistema de retardo, cuya finalidad es aplazar la alimentación de las tarjetas controladoras hasta que el sistema este inicializado por completo.

El circuito que se utilizó esta descrito en la Figura 13, es básicamente un circuito RC y un circuito comparador de voltaje. Se utilizó el circuito RC como temporizador y se configura el comparador de voltaje para que este se activara en una constante de tiempo  $\tau$ , que equivale aproximadamente al 63% de la carga en el capacitor. Entonces justo cuando el capacitor llegue a dicho porcentaje de carga el comparador se activará en la salida y este activará un opto acoplador, que a su vez activará un interruptor de carga construido a partir de Mosfets de potencia que alimentará las tarjetas controladoras de los motores.

Figura 16. Circuito de retardo de alimentación.

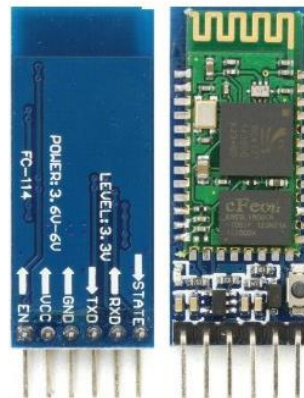


(N. James; R. Susan, 2005)

Para la implementación de la comunicación Bluetooth se utilizó dos módulos HC-05 (Figura 15). Con ellos se implementó una red inalámbrica de dos dispositivos, uno que opera como maestro y el otro como esclavo, la interpretación de esto es sencillo. El dispositivo que opera como maestro está esperando que el esclavo le envíe información para que este actúe con base en esta información. En este caso el esclavo que se encuentra en el mando inalámbrico, envía la información de estatus del mando, es decir lo que el usuario requiere del sistema.

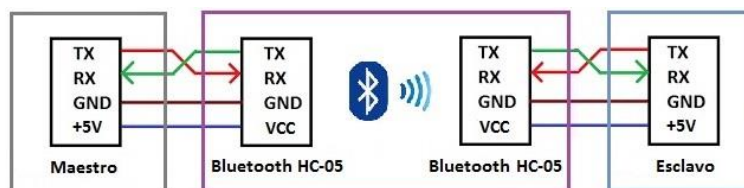
El maestro al recibir dicha información la traslada vía comunicación serial al controlador. Dicho proceso de intercambio de información sucede continuamente, por tiempo indefinido mientras que el sistema y el mando estén activos. El intercambio continuo permite al sistema conocer todo el tiempo, cual es el requerimiento del usuario, es decir que acción tomar. En la Figura 16, se detalla la conexión de la configuración antes descrita.

Figura 17. Módulo Bluetooth HC-05.



Nilsson, James W.; Riedel, Susan A. 2005

Figura 18. Configuración maestro-esclavo comunicación Bluetooth.



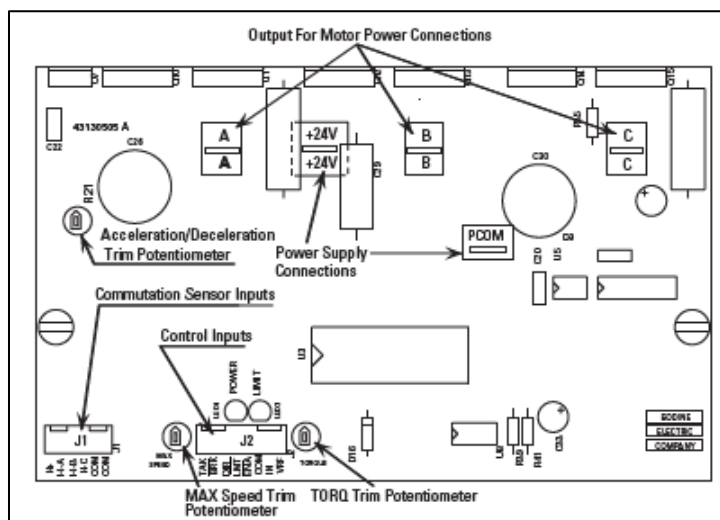
Para la configuración de dichos módulos se utilizan comandos los comandos AT, estos son instrucciones codificadas que conforman un lenguaje de comunicación entre el usuario y el módulo. En palabras más simples son comandos que permiten al usuario comunicarse con los módulos, para que estos puedan ser configurados dentro de su memoria interna. En el Cuadro 4, se observa los comandos AT utilizados y la descripción de cada uno de ellos. Para entrar en este modo de configuración se implementó un código de comunicación serial entre el módulo y el controlador. Utilizando un monitor serial se implementaron los comandos AT para su configuración.

Cuadro 4. Listado de comandos AT y su descripción.

Comando AT	Descripción
AT	Comando para confirmar comunicación serial.
AT NAME = <name>	Comando para cambiar nombre del módulo.
AT PSWD = <pin>	Comando para cambiar contraseña de acceso a modulo.
AT UART = <baud>,<stopbit>,<parity>	Comando para elegir baudrate, si se requiere bit de stop y de paridad.
AT ROLE = <role>	Comando para elegir el rol del módulo; 0 esclavo y 1 maestro.
AT CMODE = <mode>	Comando para elegir modo de comunicación; 0 para conectarse a una dirección específica y 1 para conectarse a dirección disponible.
AT BIND = <address>	Comando para especificar dirección específica a la cual conectarse.
AT ADDR?	Comando para preguntarle al módulo cuál es su dirección física (MAC).

Para el control del sistema de triángulos con los que la silla escala gradas, se utilizaron las tarjetas controladoras que se muestran en la Figura 17. En el mando inalámbrico se adaptó un botón que alterna las funciones de movimiento lineal y el de escalado en gradas, es decir el botón activa o desactiva dichas funciones según el requerimiento del usuario.

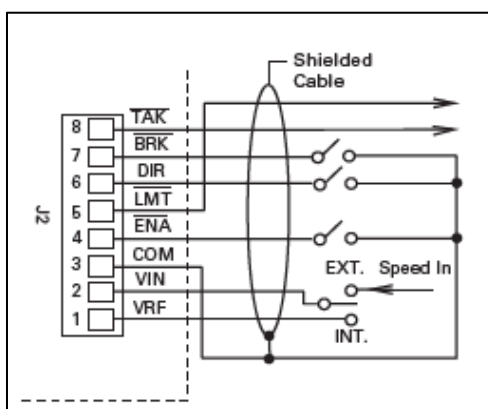
Figura 19. Controlador motor DC Brushless.



(Bodine Electric, 2012)

En la Figura 18, se muestra el conector J2 contenido en la tarjeta controladora. El conector posee los puertos de entradas y salidas. En el Cuadro 5, observamos las entradas de interés para el control de los motores, además de la configuración a cada una de ellas para su funcionamiento. Para el funcionamiento de la misma se asignaron 4 pines de salidas digitales del controlador. Uno para habilitar los 4 motores, uno para el freno de los 4 motores y dos para el sentido de giro de los motores. Se optimizó el uso de pines de este modo ya que no es necesario tener un pin para cada BRK y ENA de cada motor, y se utilizaron dos DIR por que el montaje de los motores en la estructuras cambia su sentido de giro a cada lado de la silla, es decir los motores del lado izquierdo giran en la dirección contraria a los del lado derecho por la posición de su montaje.

Figura 20. Conector J2, entradas y salidas.



(Bodine Electric, 2012)

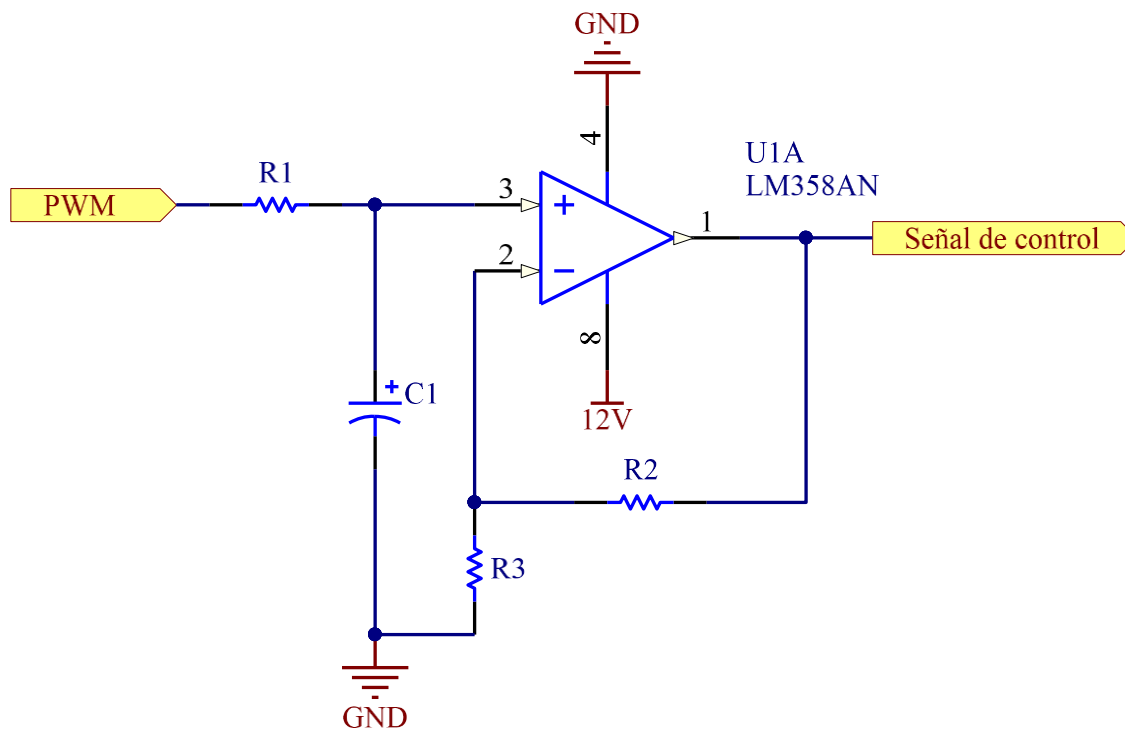
Cuadro 5. Estados de las entradas para la tarjeta de control.

Pin	Voltaje en bajo	Voltaje en alto
ENA	Habilitado	Deshabilitado
DIR	Giro antihorario	Giro horario
BRK	Frena el motor	Permite giro del motor

## F. RESULTADOS

La señal PWM generada por el controlador entra al circuito diseñado que se muestra en la Figura 19, circuito en el cual la señal es filtrada y amplificada para generar la señal de control para los motores DC. En el Cuadro 6, se muestran los valores de las resistencias y capacitores utilizados. Dichos valores aseguran el correcto funcionamiento del proyecto. Cabe hacer notar que el proyecto utiliza cuatro circuitos (Figura 19), uno para cada motor de la silla de ruedas.

Figura 21. Circuito generador de señal de control motor DC.



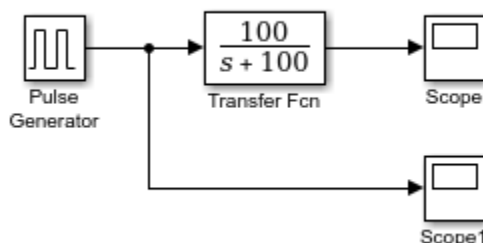
Cuadro 6. Componentes utilizados para el circuito generador de señal de control motor DC.

Componente	Valor
R <sub>1</sub>	10kΩ
R <sub>2</sub>	10kΩ
R <sub>3</sub>	20kΩ
C <sub>1</sub>	1μF

Se realizó un análisis temporal del filtro pasa bajas pasivo implementado en la Figura 19, con la finalidad de conocer el posible retraso que inyecte a la señal de control. Conocer dicho valor es fundamental para la implementación del mismo, ya que de ser muy extenso el retraso supone una desventaja al diseño. Dicho análisis fue implementado a partir del complemento Simulink del software Matlab, como se muestra en la Figura 20. Se realizó a partir del modelaje de la función de transferencia del filtro descrita en la expresión (14), utilizando los valores que se muestran en el Cuadro 6.

$$H(S) = \frac{1}{RC} \frac{1}{S + \frac{1}{RC}} \quad (14)$$

Figura 22. Implementación función de transferencia filtro pasa bajas.



La señal PWM del circuito fue recreada a partir de un generador de pulsos con valores de frecuencia 185 kHz y Duty Cycle 50%, a su vez el filtro mediante el bloque de función de transferencia como se observa en la Figura 20. Mediante la utilización del software fue posible realizar un barrido DC del comportamiento de la señal PWM filtrada, como se observa en la Figura 21. A partir de ello podemos observar que el retraso es poco significativo puesto que el valor es del orden de los milisegundos (79.4 ms) como se observa en la Figura 22.

Figura 23. Barrido DC señal PWM filtrada.

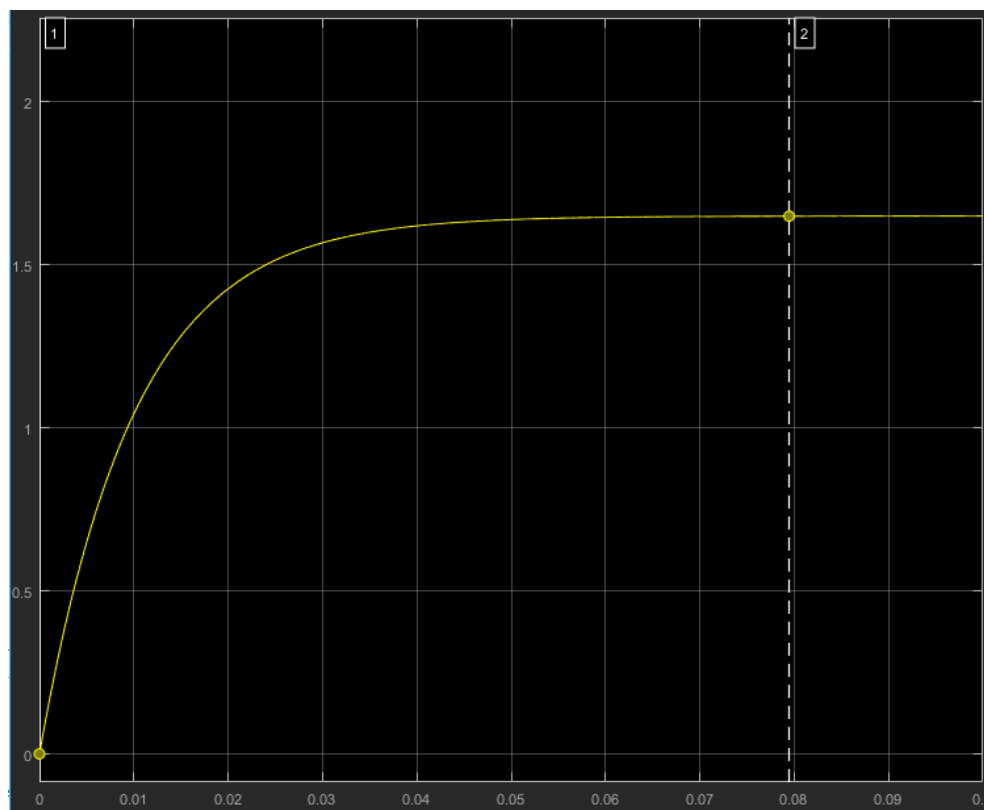
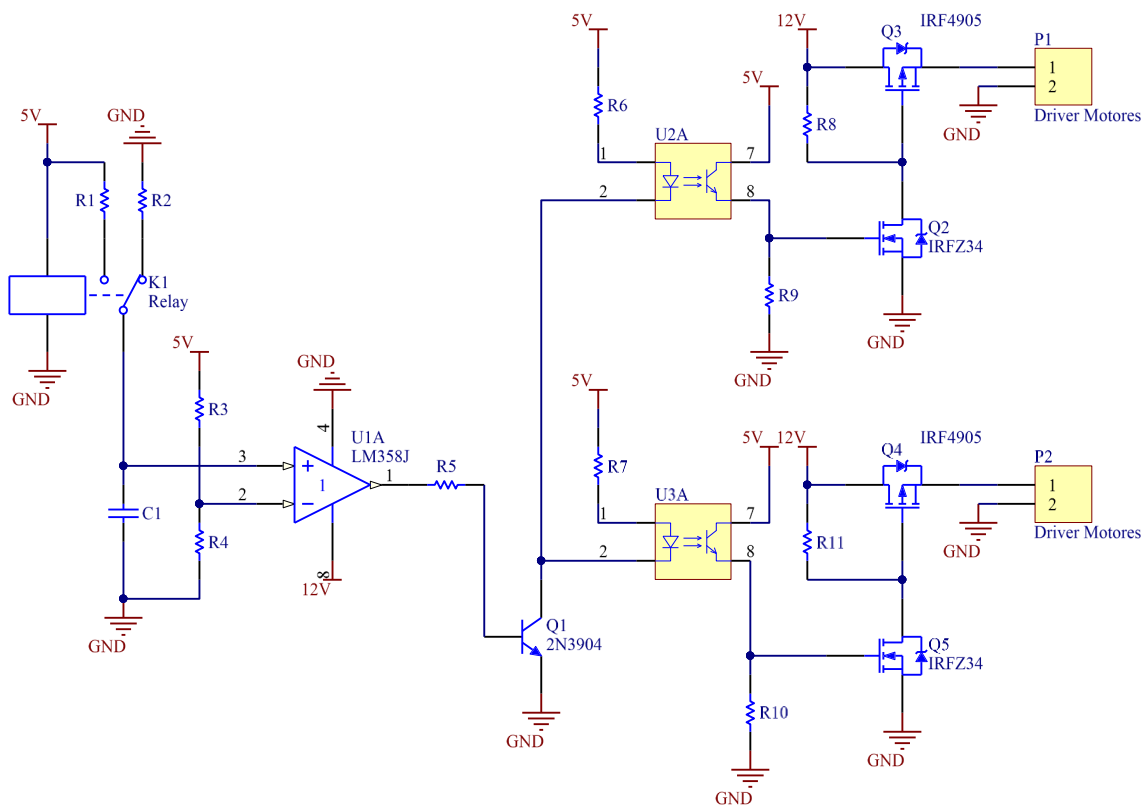


Figura 24. Datos numéricos medición de cursores.

Cursor Measurements			
Settings			
Measurements			
	Time	Value	
1	0.000e+00	0.000e+00	
2	0.079	1.650e+00	
$\Delta T$	79.457 ms	$\Delta Y$	1.650e+00
$1 / \Delta T$		12.585 Hz	
$\Delta Y / \Delta T$		20.760 (/s)	

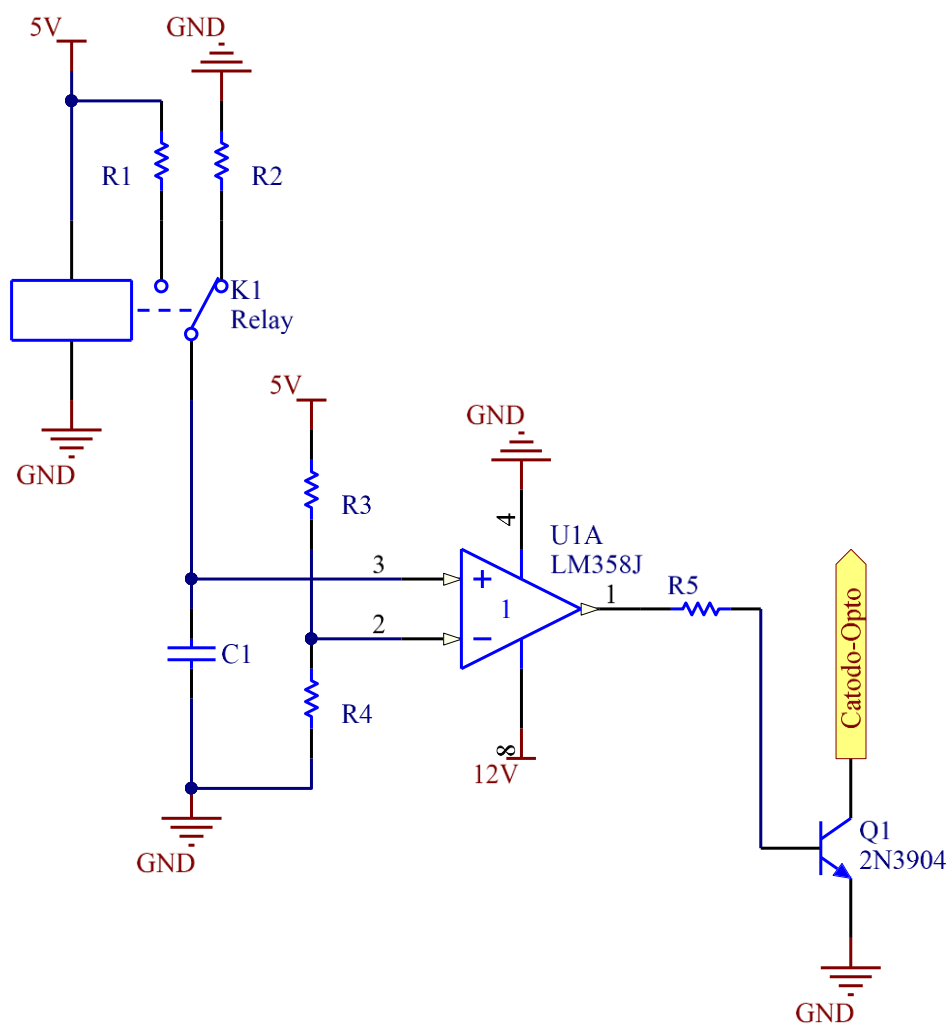
Para la protección de arranque del sistema por la preservación de la integridad del usuario, se diseñó un circuito de retardo a la conexión de los controladores para los motores DC. El circuito está descrito en la Figura 20, el cual puede subdividirse en dos etapas; circuito de activación al retardo y circuito interruptor de carga (Load Switch) para etapa de potencia. En el Cuadro 7, se muestran los valores de resistencias y capacitores utilizados en el circuito.

Figura 25. Circuito de retardo a la conexión para controladores de motores DC.



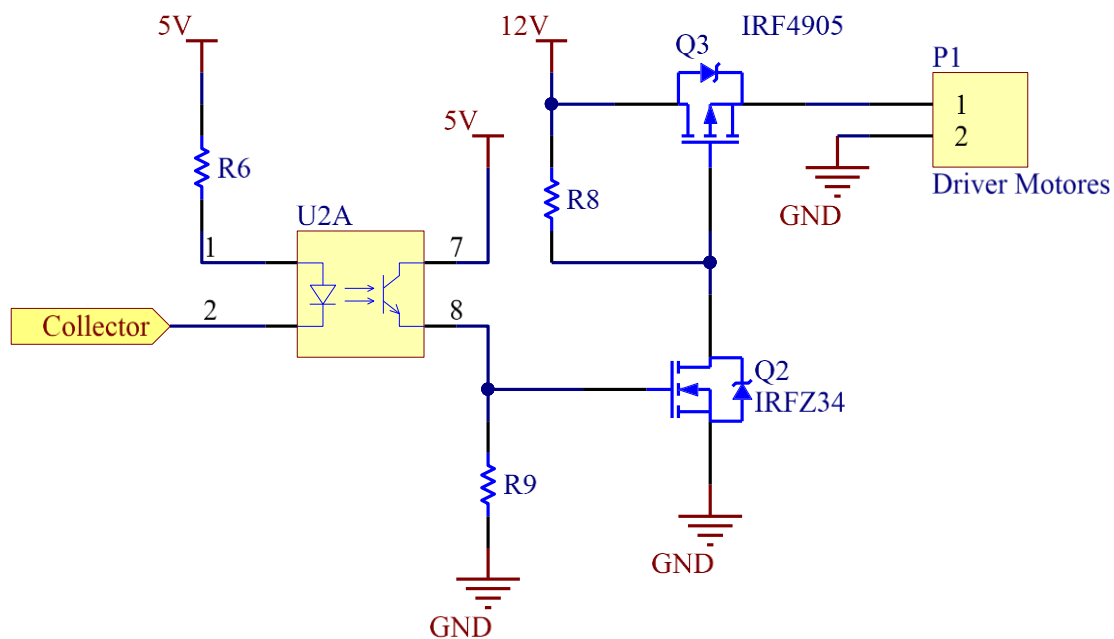
La etapa del circuito de activación al retardo (Figura 21), funciona como un temporizador a partir de la carga del capacitor en el circuito RC. El comparador de voltaje se activa justo a los 8 segundos, según la configuración del divisor de voltaje. El comparador activa el transistor de BJT, que opera como interruptor para el opto acoplador. La utilización del relay como se observa en la Figura 21, es útil para la carga y descarga del capacitor en el circuito. Ya que al estar encendido el sistema, el relay conecta a la resistencia de circuito RC a 5V, lo que provoca la carga del capacitor. Contrario si el sistema está apagado el relay conecta a tierra la resistencia, lo que provoca una conexión en paralelo de la resistencia y el capacitor, que a su vez descarga el capacitor.

Figura 26. Circuito de conexión al retardo.



En la Figura 22, se muestra el circuito interruptor de carga (load switch) para la etapa de potencia. Dicho circuito está diseñado para operar como un driver on/off para la alimentación de las tarjetas de control para los motores DC. Dicho driver está construido a partir de mosfets de potencia, capaces de soportar la corriente demandada por los controladores de los motores. La utilidad del opto acoplador dentro del circuito que se muestra en la Figura 22, radica en su velocidad de conexión y desconexión, además de aislar el sistema electrónico de la etapa de potencia. La importancia de aislar dichas partes, es para preservar y proteger la integridad de todos los componentes vitales dentro del sistema al momento de un fallo o corto circuito en la etapa de potencia.

Figura 27. Circuito interruptor de carga (load switch) para la etapa de potencia.

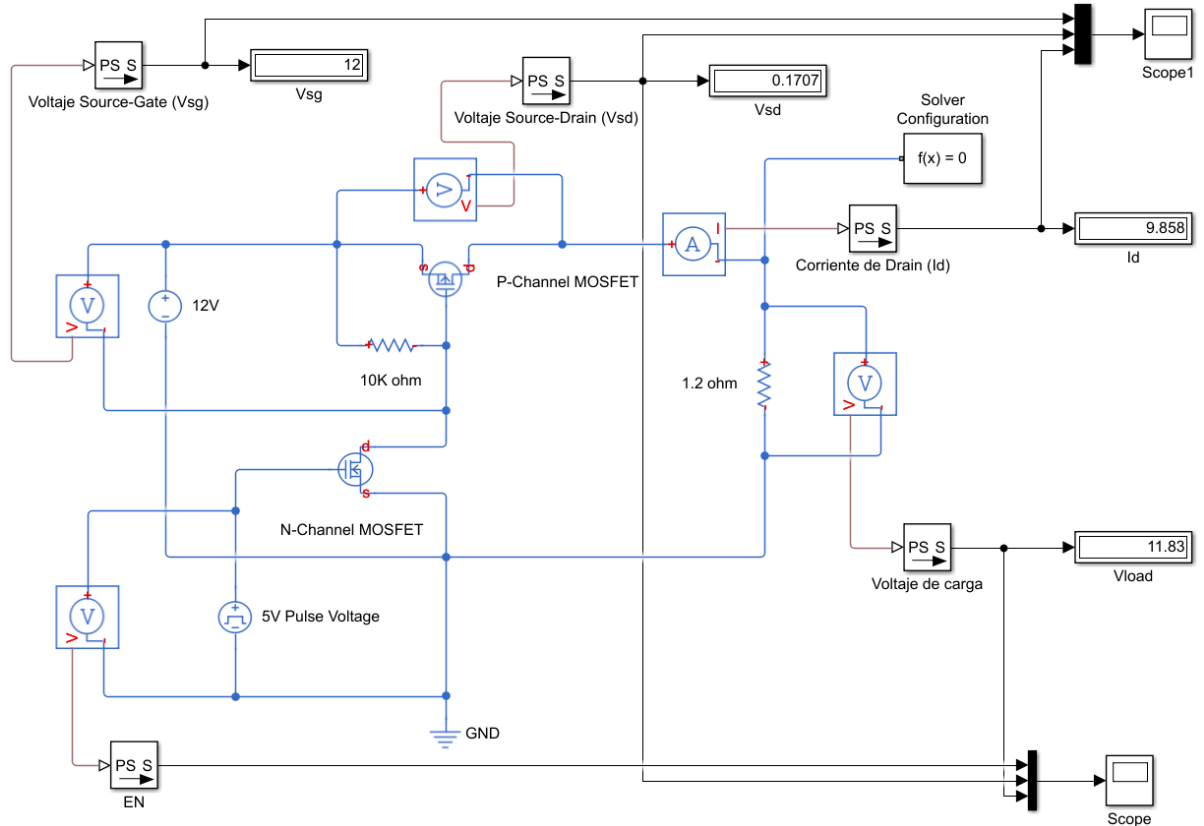


Cuadro 7. Componentes utilizados para circuito de retardo a la conexión.

Componente	Valor
R <sub>1</sub>	68kΩ
R <sub>2</sub>	1kΩ
R <sub>3</sub>	6.8kΩ
R <sub>4</sub>	10kΩ
R <sub>5</sub>	1kΩ
R <sub>6</sub>	1kΩ
R <sub>7</sub>	1kΩ
R <sub>8</sub>	10kΩ
R <sub>9</sub>	10kΩ
R <sub>10</sub>	10kΩ
R <sub>11</sub>	10kΩ
C <sub>1</sub>	100μF

Se realizó un análisis del circuito load switch mostrado en la Fig. 25, con la finalidad de conocer el comportamiento del transistor mosfet tipo-P que opera como canal de carga. La importancia de este análisis recae en conocer si dicho transistor opera correctamente para lograr su máximo desempeño óptimo. El análisis se realizó mediante la extensión Simulink del software Matlab, como se muestra en la Fig. 26. En dicho esquema se puede observar cómo se recreó el circuito load switch y algunas mediciones tales como el voltaje de drain-source ( $V_{DS}$ ), voltaje de gate-source ( $V_{GS}$ ), corriente de drain ( $I_D$ ) y el voltaje de carga. Para que el análisis sea veraz, se realizó un modelaje de los transistores mosfet tipo-N y tipo-P a partir de los datos que provee el fabricante en su hoja de datos, como se muestra en las Fig. 27 y Fig. 28.

Figura 28. Simulación de circuito Load Switch en Matlab para análisis.



Se crearon dos escenarios para la simulación, esto condicionado por el valor del voltaje de umbral ( $V_{th}$ ) del transistor mosfet tipo-P. Hablando propiamente del transistor IRF4905 que su valor de  $V_{th}$  ronda entre los -2V y -4V. A partir de dichos escenarios se obtuvieron las gráficas mostradas en las Fig. 29 y Fig. 30 respectivamente, donde se muestran el comportamiento de los valores de voltaje  $V_{DS}$  y  $V_{GS}$ , además de la corriente  $I_D$ .

Figura 29. Indexado de datos para transistor mosfet tipo-P.

Block Parameters: P-Channel MOSFET

P-Channel MOSFET

This block represents a P-channel MOSFET (or IGFET). Choose either a threshold-voltage or surface-potential MOSFET model by right-clicking the block in the model and selecting Simscape > Block choices.

Settings

Main Ohmic Resistance Junction Capacitance Body Diode Temperature Dependence

Parameterization: Specify from a datasheet

Drain-source on resistance,  $R_{DS(on)}$ : 0.02 Ohm

Drain current,  $I_{ds}$ , for  $R_{DS(on)}$ : -38 A

Gate-source voltage,  $V_{gs}$ , for  $R_{DS(on)}$ : -10 V

Gate-source threshold voltage,  $V_{th}$ : -4.0 V

Channel modulation, L: 0 1/V

Measurement temperature: 25 degC

OK Cancel Help Apply

Figura 30. Indexado de datos para transistor mosfet tipo-N.

Block Parameters: N-Channel MOSFET

N-Channel MOSFET

This block represents an N-channel MOSFET (or IGFET). Choose either a threshold-voltage or surface-potential MOSFET model by right-clicking the block in the model and selecting Simscape > Block choices.

Settings

Main Ohmic Resistance Junction Capacitance Body Diode Temperature Dependence

Parameterization: Specify from a datasheet

Drain-source on resistance,  $R_{DS(on)}$ : 0.04 Ohm

Drain current,  $I_{ds}$ , for  $R_{DS(on)}$ : 16 A

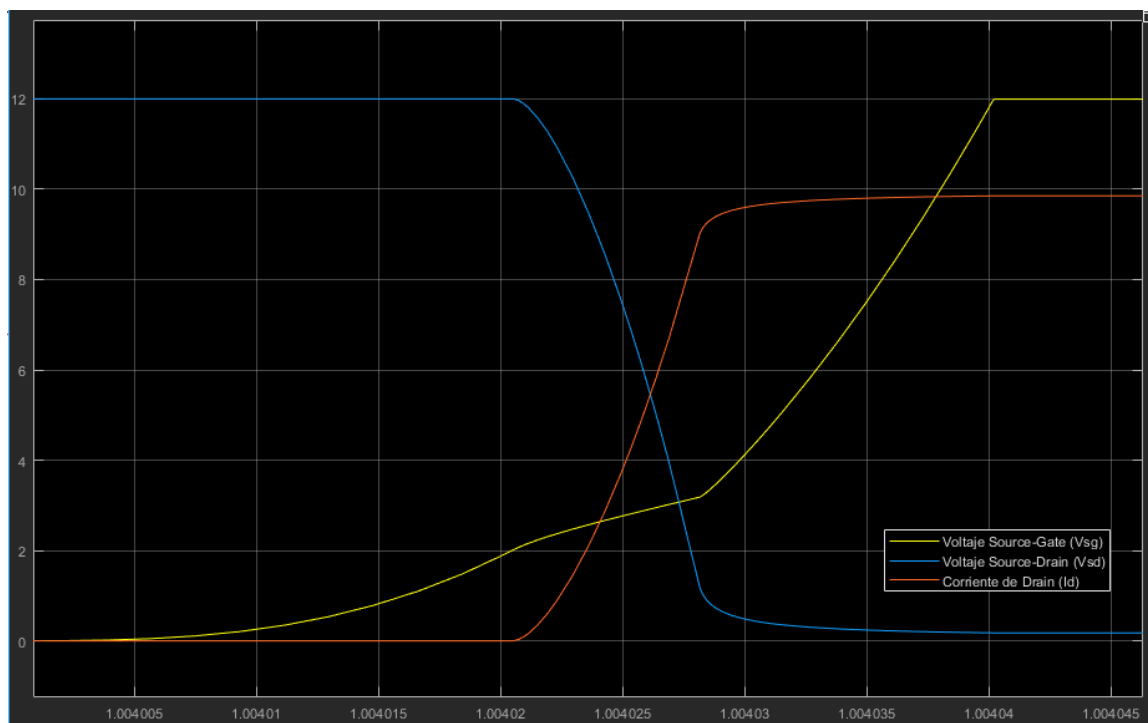
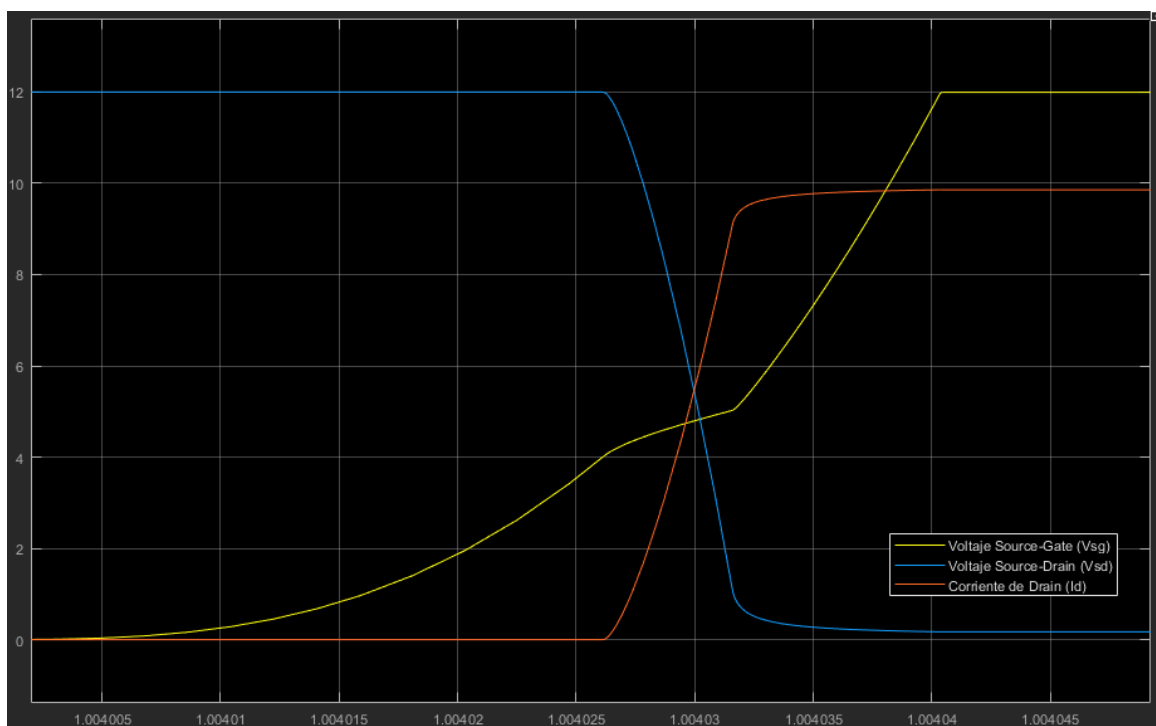
Gate-source voltage,  $V_{gs}$ , for  $R_{DS(on)}$ : 10 V

Gate-source threshold voltage,  $V_{th}$ : 4 V

Channel modulation, L: 0 1/V

Measurement temperature: 25 degC

OK Cancel Help Apply

Figura 31. Gráfica de comportamiento mosfet tipo-P con valor mínimo de  $V_{th}$ .Figura 32. Gráfica de comportamiento mosfet tipo-P con valor máximo de  $V_{th}$ .

Cuadro 8. Valores numéricos iniciales y finales de simulación load Switch.

Descripción	Valor inicial	Valor final
$V_{SG}$	0V	-12V
$V_{SD}$	12V	-0.17V
$V_{LOAD}$	0V	11.83V
$I_D$	0A	9.86A

A partir de los resultados obtenidos en la simulación se puede inferir respecto al funcionamiento del circuito, sobre todo el funcionamiento del mosfet tipo-P. Dicho componente resulta ser el mas importante dentro del circuito, puesto que es quien suministra la alimentación de los motores DC para el control del proyecto y es esencial que funcione optimamente. Basándonos en las gráficas obtenidas y los valores descritos en el Cuadro 8, se puede observar como el transistor va superando cada una de sus fases de operación.

En un principio se ve en su etapa de corte, donde el voltaje de gate a source ( $V_{GS}$ ) es cero y este es menor que el valor de voltaje de umbral ( $V_{th}$ ) apoyado por la expresión (15) y aun más claro lo vemos en los valores de voltaje ( $V_{LOAD}$ ) y corriente ( $I_D$ ) de la carga puesto que estos son cero y nos indica que no está siendo alimentado.

$$-V_{GS} < -V_{th} \quad (15)$$

Luego se observa en su etapa lineal, la que sucede a partir de la activación del transistor tipo-N y conecta a tierra el Gate del transistor tipo-P lo que permite que el valor de voltaje ( $V_{GS}$ ) ascienda y al superar el valor de umbral ( $V_{th}$ ) entra en su etapa lineal. Siempre con la condición descrita en la expresión (16) que indica que el valor de voltaje ( $V_{DS}$ ) sea mayor a la suma entre ( $V_{GS}$ ) y ( $V_{th}$ ).

$$0 < -V_{DS} < -V_{GS} + V_{th} \quad (16)$$

Finalmente se observa su etapa de saturación, donde la suma de voltajes entre ( $V_{GS}$ ) y ( $V_{th}$ ) supera el valor de voltaje de ( $V_{DS}$ ) como se muestra en la expresión (17). Dicha etapa es la que nos interesa puesto que podemos obtener el funcionamiento optimo del circuito. En ella se obtiene el valor de resistencia interna más baja lo que nos permite disipar menos potencia y se obtiene la menor pérdida de voltaje. Podemos observar lo mencionado anteriormente más claramente en los valores de voltaje ( $V_{LOAD}$ ) y corriente ( $I_D$ ) de la carga.

$$0 < -V_{GS} + V_{th} < -V_{DS} \quad (17)$$

Respecto a las diferencias entre los dos casos simulados lo vemos claramente en las gráficas donde el caso del mínimo y el máximo ( $V_{th}$ ) suceden en tiempos distintos y esto tiene sentido si observamos las expresiones antes descritas puesto que el valor de ( $V_{th}$ ) influye directamente en cómo se comporta el funcionamiento.

El diseño del PCB se realizó utilizando el software Altium Designer, a partir de los esquemáticos antes mostrados, además del controlador, headers y terminales de tornillo como complemento del circuito. Los headers y terminales de tornillo son utilizados para entradas y salidas del PCB. En la Figura 23 se muestra el diseño de la Bottom Layer, en esta capa se manejan las conexiones a tierra y 12V de la etapa de potencia del circuito. Por ellos se utilizan polígonos irregulares para la conexión de los mismos, esto con la finalidad de disipar la potencia de las altas corrientes que se manejan en el sistema. En la Figura 24 se muestra el diseño de la Top Layer, en el se encuentran las conexiones de las señales de baja corriente del sistema y algunas señales que manejan corrientes altas, por ello se observa la utilización de polígonos irregulares para su conexión.

Figura 33. Diseño Botton Layer PCB principal.

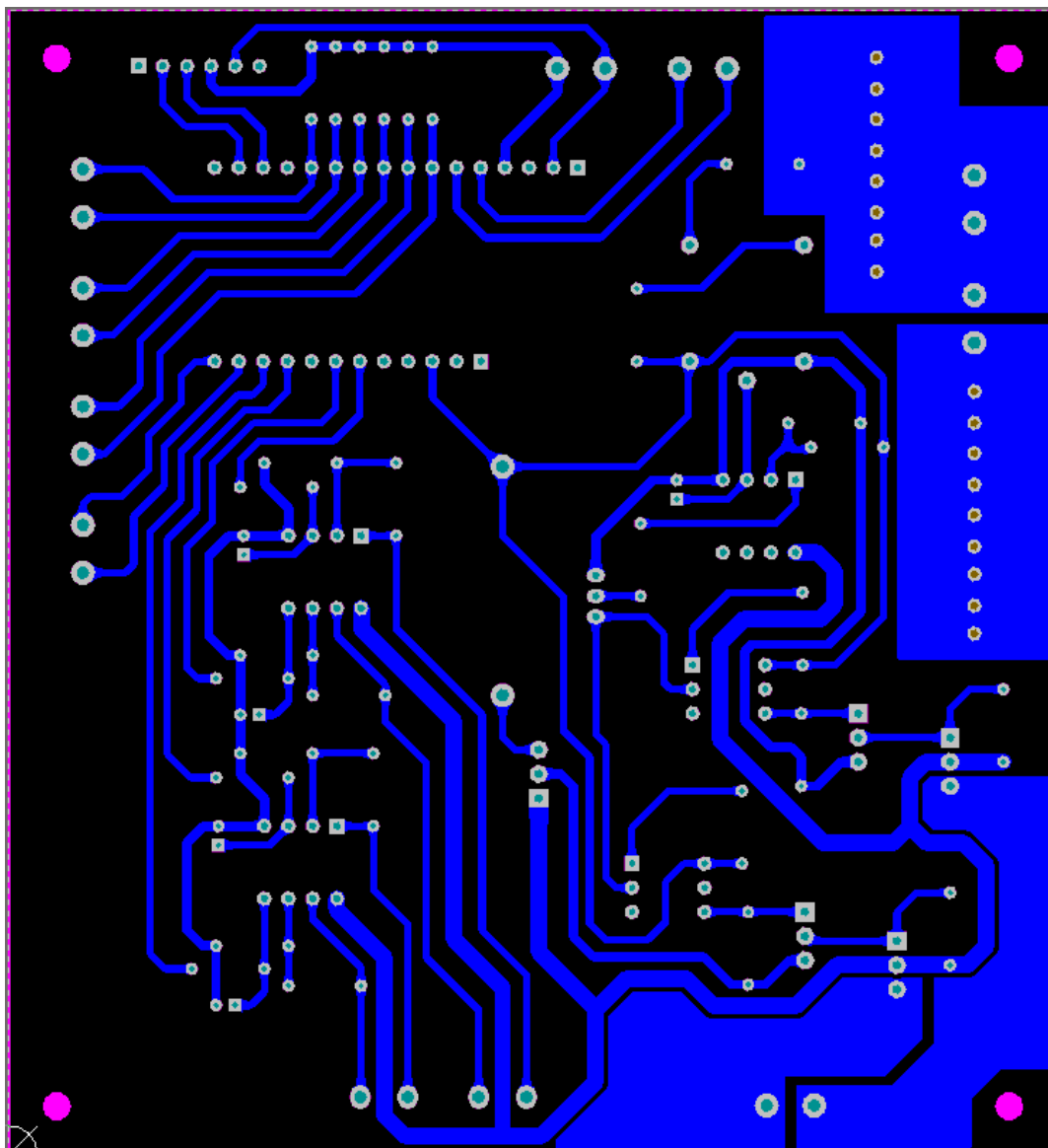
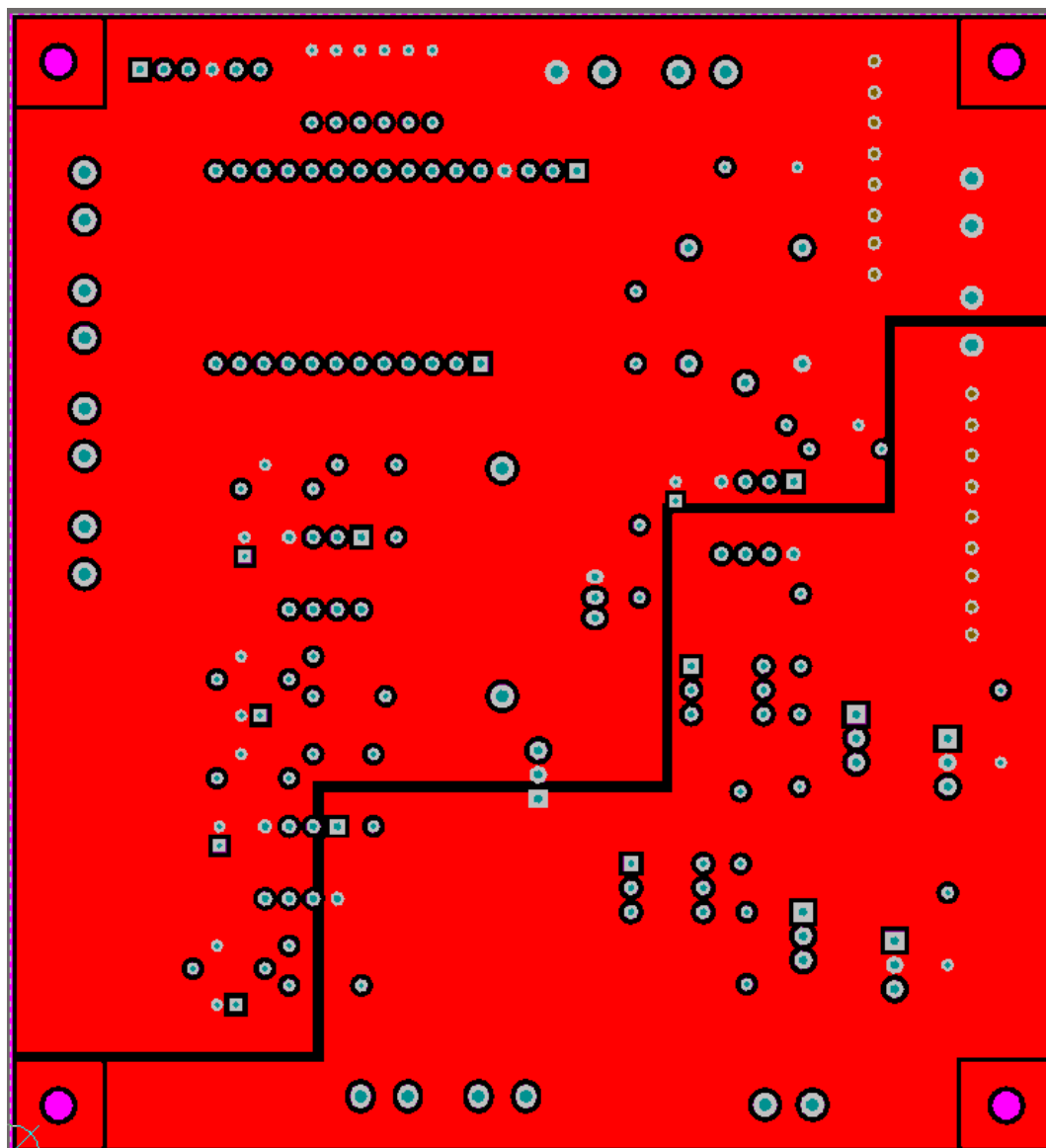
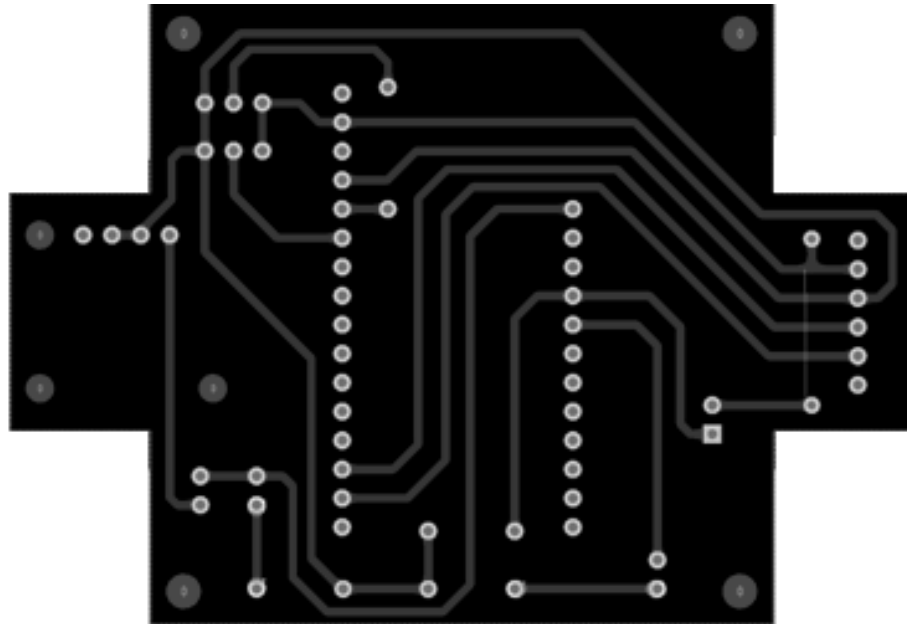


Figura 34. Diseño Top Layer PCB principal.



En la Figura 25, se observa el diseño del PCB para el mando inalámbrico de la silla de ruedas, en él se encuentran, el controlador, resistencias, headers para entradas y salidas, y el montaje de un cargador de batería. El cargador de batería es un módulo de carga de baterías de litio, debido a que es un mando inalámbrico este utiliza una batería integrada y con dicho módulo de carga es como se abastece de carga la batería.

Figura 35. Diseño PCB Mando inalámbrico.



En la Figura 26, se puede observar las piezas diseñadas ensambladas y el montaje de la batería con el cargador, además del PCB. En la Figura 27, se observa el ensamblaje y montaje final del mando inalámbrico. Se puede observar que posee dos leds indicadores, uno para indicar si el estado del mando es encendido y otro para indicar si la carga de la batería es baja. Además de un interruptor de encendido y apagado.

Figura 36. Mando inalámbrico montaje interno.

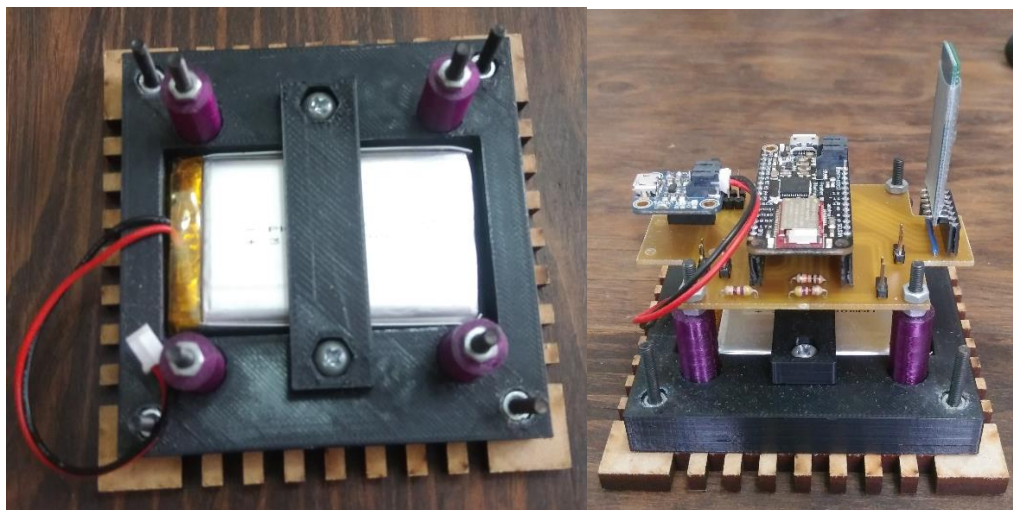
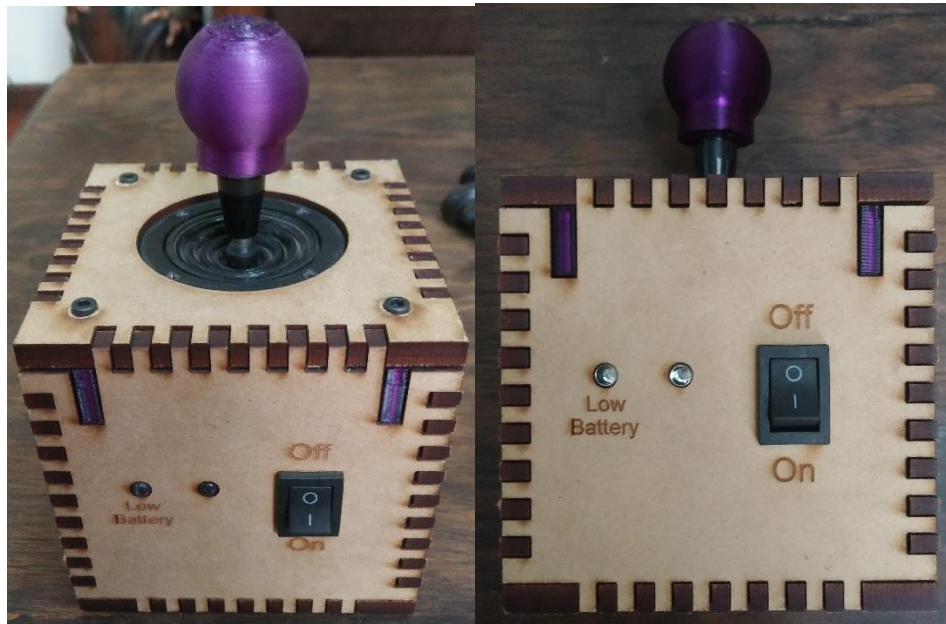


Figura 37. Mando inalámbrico apariencia externa.



### III. CONCLUSIONES

En principio los objetivos planteados estaban trazados por las limitaciones que el sistema mecánico posee, ya que el sistema de triángulos en las ruedas deja a la silla apoyada sobre ocho ruedas, y esto no permite la rotación de la misma sobre su propio eje. Esta limitación llevó a plantear el poder realizar un sistema que operara con más autonomía, se pretendía realizar un sistema de control que permitiera a la silla colocarse en cuatro puntos de apoyo únicamente, con la finalidad de lograr el giro sobre su propio eje, además de permitirle escalar gradas. Pero esto implicaba una dificultad, ya que para mantener los motores en una sola posición, tendría implicaciones de potencia y en aumentarían el consumo de energía en la batería.

Con base en a las limitaciones, se pensó en emplear un enlace mecánico, cuyo sistema de control consistiría en llevar los triángulos a un punto de equilibrio y luego un actuador lineal enclavaría el triángulo en esa posición, dejando al sistema en cuatro puntos de apoyo. Finalmente este sistema no fue implementado debido al costo de los actuadores lineales, los encoders y que era necesario modificar la estructura mecánica para situar los sensores y actuadores, necesarios para sistema de control.

Pero la parte del enlace mecánico se solucionó, realizando un enlace manual con pines de acero que soportan el esfuerzo de corte que realiza el peso de la estructura sobre ellos. Esta solución fue un ahorro monetario y de tiempo para continuar con el diseño e implementación del sistema electrónico.

El sistema de comunicación inalámbrica mediante Bluetooth, le brindó versatilidad y comodidad al proyecto. Además el mando construido a partir de un joystick, lo hace muy intuitivo y amigable para el usuario, ya que este no supone nada de complejidad al momento de controlar la silla de ruedas. Sin descartar el hecho de que cumple con el objetivo de controlar el movimiento lineal y escalar gradas de la silla de ruedas.

Dentro del desarrollo del sistema electrónico surgió un inconveniente con respecto al controlador seleccionado puesto que eran inciertos algunos valores de tiempos de inicialización. El problema radica específicamente en que el tiempo de inicialización del módulo Feather M0 con el cual se controla el proyecto, además del lenguaje de programación utilizado (Arduino). Este tiempo resultó suficientemente significativo para el control del proyecto puesto que en dicho tiempo las tarjetas controladoras requieren de tener una referencia específica para que estas no permitan el movimiento de los motores y la silla no suponga un peligro para el usuario al moverse sin una orden del mismo. Dicho problema fue solventado mediante un circuito de retardo a la conexión para la alimentación de las tarjetas controladoras. La solución solvento el problema, sin embargo se considera que no fue óptima. El sistema electrónico final utilizado es funcional y cumple con los objetivos planteados al inicio del proyecto, sin embargo se llegó a la conclusión que no es la más óptima. Debido al problema antes mencionado y además fue necesario la utilización de hardware adicional para acondicionamiento de una señal PWM lo que pudo ser evitado al utilizar un controlador que posea salidas analógicas DAC.

## IV. RECOMENDACIONES

Para mejorar el proyecto se recomienda la implementación de un sistema de control robusto para el movimiento del sistema de triángulos de la silla de ruedas, mediante la utilización de encoders y otros sensores. Dicho sistema supone varias ventajas para el proyecto, entre ellas; el enclave automático de los triángulos con actuadores lineales para el desplazamiento lineal de la silla. Este sistema es necesario ya que actualmente el enclave de los triángulos se realiza de forma manual para apoyar la silla en cuatro puntos de apoyo, esto supone ineficiencia al tener que poner o remover los pines cuando lo es requerido.

Además este sistema serviría para la implementación de un sistema de lazo cerrado para el control de los triángulos al momento de subir o bajar gradas. Actualmente el sistema funciona con un sistema de lazo abierto, es decir el controlador espera al usuario para actuar. Esto supone una desventaja, ya que no es preciso cuánto deben girar los triángulos y el riesgo de cometer errores depende exclusivamente del usuario. Estas desventajas se corregirían con un sistema de control que le indique a los actuadores cuanto girar y el mismo puede corregir errores, disminuyendo el riesgo.

También se recomienda realizar un sistema de control para el movimiento lineal de la silla, esto supone que el sistema sabe a qué velocidad y cuánto girar los motores DC. Estos motores no son construidos perfectamente iguales, lo que supone que no giran igual. La ventaja del sistema de control radica en la desventaja de los motores, ya que actualmente las trayectoria largas de la silla no son perfectamente en línea recta, tienden a torcer la trayectoria.

Se recomienda respecto al problema del controlador Feather M0, la utilización de un nuevo controlador específicamente que no sea un módulo prefabricado, como por ejemplo un PIC el cual puede ser programado mediante el lenguaje de C++. Esto supondría una ventaja respecto al módulo ya utilizado puesto que el control que el programador tiene sobre él es total. El poder programar en un lenguaje de bajo nivel permite además poder controlar de mejor forma los tiempos y optimizar el problema anterior o bien que este no suceda. Además se recomienda que este controlador posea salidas analógicas para evitar el uso de señales PWM y evitar la utilización de hardware adicional para el acondicionamiento de señales.

## V. BIBLIOGRAFÍA

- Adafruit Feather M0 Bluefruit LE*. <https://learn.adafruit.com/adafruit-feather-m0-bluefruit-le/overview>  
[Mayo de 2016]
- Instructions for Installation and Operation Type ABL-3905 and ABL-3907 24-Volt Brushless DC Motor Controls*. 2012. Estados Unidos: Bodine Electric Company. 24 págs.
- La Respuesta natural de un circuito RC*. <https://es.khanacademy.org/science/electrical-engineering/ee-circuit-analysis-topic/ee-natural-and-forced-response/a/ee-rc-natural-response> [Febrero de 2018]
- La Respuesta de un circuito RC a un voltaje de escalón*. <https://es.khanacademy.org/science/electrical-engineering/ee-circuit-analysis-topic/ee-natural-and-forced-response/a/ee-rc-natural-response>  
[Febrero de 2018]
- Nilsson, James W.; Riedel, Susan A. 2005. *Circuitos Eléctricos*. 7ª ed. Madrid: Pearson Educación S.A.  
1048 págs.
- On Semiconductor. *Using MOSFETs in Load Switch Applications*.  
<http://www.onsemi.com/pub/Collateral/AND9093-D.PDF> [Marzo de 2018]
- Q. Deng. 2007. *A Primer on High-Side FET LoadSwitches*. EE Times. 7 págs.
- Sabertooth 2x12 user's guide*. 2012. Estados Unidos: Dimension Engineering. 21 págs.
- Rashid, Mahummad H. 2004. *Electrónica de potencia*. 3ª ed. México: Pearson Educación. 904 págs.

## VI. ANEXOS

### 1. Programa principal controlador silla.

```

/*****
// Se definen las variables y constantes a utilizar para el proceso de control
*****/

/*-----*/
//variables comunicacion inalambrica
/*-----*/
    unsigned long Tiempo_TX;
    unsigned char START_BYTE = 0x53; // ASCII "S"
    unsigned char Value1 = 0;
    unsigned char Value2 = 0;
    unsigned char Value3 = 0;
    unsigned char Value4 = 0;
    unsigned char staticValue = 0;
    unsigned char checksum1 = 0;
    unsigned char checksum2 = 0;
    unsigned char checksum3 = 0;
    unsigned char checksum4 = 0;

/*-----*/
//variables proceso de control
/*-----*/
    int pos_1 = 127;
    int pos_2 = 127;
    int pos_3 = 127;
    int pos_4 = 127;

    boolean syncByteFound = false;
    boolean bandera = false;

    int xPos = 0;
    int yPos = 0;

    int pos_x = 0;
    int pos_y = 0;
/*-----*/

/*****/

```

```

/*****
// Programa principal controlador silla
*****/
/*-----*/
//Iniciacion de programa
/*-----*/
void setup(){
  Serial1.begin(9600); // Comienzo de la comunicación serie.
  Serial.begin(9600); // comienzo comunicacion monitor serial para monitoreo.
  //pinMode(Pulsador, INPUT); // Pin digital 4 como entrada.
  Tiempo_TX = millis();
  analogWrite(11,pos_1);
  analogWrite(10,pos_2);
  analogWrite(6,pos_3);
  analogWrite(5,pos_4);
  //attachInterrupt(0, serialEvent, CHANGE);
}
/*-----*/

/*-----*/
//Ciclo infinito programa
/*-----*/
void loop(){
  serialEvent1(); //llamado a interrupcion serial
}
/*-----*/

/*-----*/
//Programa interrupcion serial
/*-----*/
void serialEvent1() {
  /*-----*/
  //inicializacion variables temporales
  /*-----*/
  unsigned char rxByte = 0;
  unsigned char calculatedChecksum1 = 0;
  unsigned char calculatedChecksum2 = 0;
  unsigned char calculatedChecksum3 = 0;
  unsigned char calculatedChecksum4 = 0;
  /*-----*/
}

```

```

if (Serial1.available() > 0) {

    // pregunta por un el bit de sincronizacion, si es verdadero sigue con el proceso.
    if (syncByteFound == false) {
        rxByte = Serial1.read();
        // si encuentra el start byte cambia el bit de sincronizacion a verdadero.
        if (rxByte == 0x53){
            syncByteFound = true;
            Serial.println("bitsync");
        }
        //sino no encuentra strat byte mantiene el bit de sincronizacion en falso.
        else{
            syncByteFound = false;
            Serial.println("not bitsync");
        }
    }
}

// si encontro el bit de sincronizacion verdadero, pregunta por cuantos bytes hay en el puerto serial
if ((Serial1.available() > 8)) {

    //lee los bytes en el puerto serial y los almacena en el orden indicado
    Value1 = Serial1.read();
    Value2 = Serial1.read();
    Value3 = Serial1.read();
    Value4 = Serial1.read();
    staticValue = Serial1.read();
    checksum1 = Serial1.read();
    checksum2 = Serial1.read();
    checksum3 = Serial1.read();
    checksum4 = Serial1.read();

    //Calcula el checksum de cada valor recibido
    calculatedChecksum1 = Value1 + staticValue;
    calculatedChecksum2 = Value2 + staticValue;
    calculatedChecksum3 = Value3 + staticValue;
    calculatedChecksum4 = Value4 + staticValue;
}

```

```

//si el valor de cheksum1 recibido con el calculado concuerdan escribe el pwm al puerto.
if (calculatedChecksum1 == checksum1){
  Serial.println("[Checksum Passed 1]");
  pos_1 = Value1;
  analogWrite(11,pos_1);
}

else{
  Serial.println("[Checksum FAILED]");
}

//si el valor de cheksum2 recibido con el calculado concuerdan escribe el pwm al puerto.
if (calculatedChecksum2 == checksum2){
  Serial.println("[Checksum Passed 2]");
  pos_2 = Value2;
  analogWrite(10,pos_2);
}

else{
  Serial.println("[Checksum FAILED]");
}

//si el valor de cheksum3 recibido con el calculado concuerdan escribe el pwm al puerto.
if (calculatedChecksum3 == checksum3){
  Serial.println("[Checksum Passed 3]");
  pos_3 = Value3;
  analogWrite(6,pos_3);
}

else{
  Serial.println("[Checksum FAILED]");
}

//si el valor de cheksum4 recibido con el calculado concuerdan escribe el pwm al puerto.
if (calculatedChecksum4 == checksum4){
  Serial.println("[Checksum Passed 4]");
  pos_4 = Value4;
  analogWrite(5,pos_4);
}

else{
  Serial.println("[Checksum FAILED]");
}

//Al finalizar vuelve falso el bit de sincronizacion y vacia el puerto serial.
syncByteFound = false;
Serial1.flush();
}
}
/*-----*/

```

## 2. Programa controlador mando inalámbrico.

```

/*****
// Se definen las variables y constantes a utilizar para el proceso de control
*****/
/*-----*/
//definicion de nombres para puertos
/*-----*/
#define xPin A0
#define yPin A1
#define VBATPIN A7

const int button = 13; // Pin digital para el LED
char estado;
int ind_bat = 12;

/*-----*/
//variables comunicacion inalambrica
/*-----*/
unsigned long Tiempo_TX;
unsigned char START_BYTE = 0x53; // ASCII "S"
unsigned char staticValue = 1;
unsigned char checksum1 = 0;
unsigned char checksum2 = 0;
unsigned char checksum3 = 0;
unsigned char checksum4 = 0;
unsigned int valor1 = 0;
unsigned int valor2 = 0;
unsigned int valor3 = 0;
unsigned int valor4 = 0;
/*-----*/

*****/

```

```

/*****/
//Definicion clase para la generacion de PWM a partir de un vaor analogico
/*****/
class PWM
{
    int pwm;
    int Position;
    int analog_pin;

public:

    PWM(int pin) {
        analog_pin = pin;
    }

    int Generate(int lim_Min, int lim_Max, int lim_min, int lim_max) {
        Position = analogRead(analog_pin);
        pwm = (((Position - lim_Min) * (lim_max - lim_min)) / (lim_Max - lim_Min)) + (lim_min);
        //analogWrite(pwmpin,pwm);
        return pwm;
    }
};
/*****/

//creacion de objetos para PWMs del programa principal
PWM pwm1(xPin);
PWM pwm2(yPin);
PWM pwm3(xPin);
PWM pwm4(yPin);

/*****/
// Programa mando inalambrico silla
/*****/
/*-----*/
//Iniciazilacion de programa
/*-----*/
void setup(){
    Serial1.begin(9600); // Comienzo de la comunicaci3n serie
    Serial.begin(9600); // Comienzo de la comunicaci3n monitor serie para monitoreo
    pinMode(button, INPUT);
    pinMode(ind_bat, OUTPUT);
    Tiempo_TX = millis();
}
/*-----*/

```

```

void loop()
{
  /*-----*/
  //rutina para medicion nivel de bateria
  /*-----*/
  float measuredvbat = analogRead(VBATPIN);
  measuredvbat *= 2;
  measuredvbat *= 3.3;
  measuredvbat /= 1024;

  if (measuredvbat <= 3.4) {
    digitalWrite(ind_bat, HIGH);
  }

  if (measuredvbat > 3.4) {
    digitalWrite(ind_bat, LOW);
  }
  /*-----*/

  /*-----*/
  //rutina para medicion nivel de bateria
  /*-----*/
  //Generacion de PWMs.
  valor1 = pwm1.Generate(0, 1023, 0, 254);
  valor2 = pwm2.Generate(0, 1023, 0, 254);
  valor3 = pwm3.Generate(0, 1023, 0, 254);
  valor4 = pwm4.Generate(0, 1023, 0, 254);

  //Generacion de checksums.
  checksum1 = valor1 + staticValue;
  checksum2 = valor2 + staticValue;
  checksum3 = valor3 + staticValue;
  checksum4 = valor4 + staticValue;

  //Envio serial de valores
  if (millis() - Tiempo_TX > 100) {
    Serial1.write(START_BYTE);
    Serial1.write(valor1);
    Serial1.write(valor2);
    Serial1.write(valor3);
    Serial1.write(valor4);
    Serial1.write(staticValue);
    Serial1.write(checksum1);
    Serial1.write(checksum2);
    Serial1.write(checksum3);
    Serial1.write(checksum4);
    Tiempo_TX = millis();
  }
  /*-----*/
}

```