



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería Electrónica Industrial y Automática

**Diseño eléctrico y electrónico de una
motocicleta eléctrica infantil 2x2**

Autor:

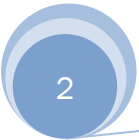
MINGUELA BENITO, IVÁN

Tutor:

**DOMÍNGUEZ VÁZQUEZ, JOSÉ
ANTONIO**

Tecnología Electrónica

Valladolid, Abril de 2016.





Resumen y palabras clave

Este proyecto, se basa en la creación de un control de tracción a las dos ruedas para una motocicleta eléctrica.

Dicho documento, surgió de una propuesta de la empresa española Enriding, que desarrolla motos eléctricas, junto con la colaboración de la Asociación LACECAL, ente comprometido en la mejora de los recursos tecnológicos de Castilla y León.

Ambas entidades, están comprometidas con el medio ambiente y volcadas en medios de transporte propulsados de forma totalmente eléctrica.

El estudio, se ha desarrollado con la intención de implementar dicho control de tracción en una motocicleta eléctrica de uso dual, convirtiéndose en un medio de transporte alternativo y ecológico, no sólo para un uso diario, sino también para ofrecer el disfrute del mismo durante el fin de semana, circulando por el campo.

Palabras claves: **Control de tracción 2x2**





Índice

Resumen y palabras clave.....	3
Índice de Ilustraciones.....	7
Índice de Imágenes.....	8
Índice de Tablas.....	10
Índice de Ecuaciones.....	10
1. Introducción y objetivos.....	11
1.1 Introducción.....	11
1.2 Objetivos.....	12
2. Estado actual de la técnica industrial.....	13
2.1 Reseña del vehículo eléctrico.....	14
2.1.1 Historia.....	14
2.1.2 Situación actual.....	15
2.1.3 Definición de vehículo eléctrico.....	16
2.1.4 Clasificación del vehículo eléctrico.....	17
2.1.5 Ventajas e inconvenientes del vehículo eléctrico frente al de combustión.....	19
2.2 Historia de la moto eléctrica.....	21
2.2.1 Fuentes de energía.....	21
2.2.2 Historia.....	22
2.2.3 Motos eléctricas vs motos de gasolina.....	24
2.2.4 Ventajas e inconvenientes de las motocicletas eléctricas.....	24
2.2.5 Análisis DAFO de las motos eléctricas.....	25
2.3 Motos eléctricas infantiles ya existentes.....	27
3. Estudio de las soluciones existentes.....	29
3.1 ¿Por qué una moto bimotor?.....	29
3.1.1 Reseña Yamaha.....	29
¿Cómo funciona la Yamaha WR450-F2 con tracción a las dos ruedas?	30
3.1.2 Reseña KTM.....	31
3.1.3 Reseña Christini technology.....	33
Desarrollo del TFG.....	35
4. Planteamiento de solución electrónica mediante motores eléctricos en los cubos de las ruedas.....	35

5.	Diseño del algoritmo y simulación mediante simulink	37
5.1	Definición básica del algoritmo a realizar	37
5.1.1	Entradas.....	37
5.1.2	Proceso	37
5.1.3	Salida	38
5.1.4	Simulación mediante Simulink.....	38
5.2	Diseño del algoritmo.....	38
6.	Estudio de los componentes necesarios y propuesta electrónica	47
7.	Construcción del prototipo	49
7.1	Diseño Hardware	49
7.1.1	Soporte o estructura.....	50
7.1.2	Colocación de los imanes y de los sensores Hall.....	50
7.1.3	Circuito impreso	52
7.2	Software.....	58
7.2.1	Selección del modo (mediante un switch)	59
7.2.2	Contaje de flancos positivos (mediante interrupciones). “La señal recibida por estos pines será filtrada”	60
7.2.3	Cálculo de la diferencia de pulsos por tiempo transcurrido o por número de flancos positivos máximo.	66
7.2.4	Toma de decisiones.....	69
7.2.5	Señal de mando al motor delantero mediante un PID.....	69
7.2.6	Regulador de potencia.....	79
7.2.7	Lectura del sensor hall lineal del acelerador.	81
8.	Pruebas realizadas.....	85
9.	Conclusiones	87
10.	Bibliografía	89
11.	Anexos	93
11.1	Tipos de motos eléctricas de Oset Bikes	93
11.2	Tipos de motos eléctricas de Xispa Motorbikes	95
11.3	Investigación de los motores empleados en los sistemas de tracción de VE, los motores brushless.....	97
11.3.1	Introducción y clasificación de motores eléctricos.....	97
11.3.2	Motores DC.....	98



- 11.4 ¿Por qué elegir Arduino Rev3 como plataforma electrónica? 113
 - 11.4.1 ¿Qué es Arduino? 113
 - 11.4.2 ¿Qué significa que Arduino sea de “software libre”? 113
 - 11.4.3 ¿Qué significa que Arduino sea de “hardware libre”? 113
 - 11.4.4 ¿Por qué elegir Arduino? 114
 - 11.4.5 Nuestra elección: Arduino Uno Rev 3 116
 - 11.4.6 Especificaciones técnicas de Arduino Uno Rev 3 116
 - 11.4.7 Alimentación, entradas y salidas de Arduino 117
 - 11.4.8 Esquema y pines 120
- 11.5 Simulación mediante Simulink 121
- 11.6 Diseño del circuito secuencial síncrono para el mantenimiento de la señal de tracción total de la rueda delantera 135
- 11.7 Diagrama de flujo 139
- 11.8 Diseño de la estructura principal 141
- 11.9 Diseño de la colocación de los imanes en las ruedas 145
- 11.10 Colocación de los sensores 147
- 11.11 Código del programa 149

Índice de Ilustraciones

- Ilustración I – Esquema de los elementos utilizados para este control de tracción 47
- Ilustración II – Filtrado del conteo de flancos de las ruedas 64
- Ilustración III – Simulación del filtrado mediante Simulink 64
- Ilustración IV – Resultados de la salida filtrada 64
- Ilustración V – Relación de la velocidad con la diferencia de pulsos 68
- Ilustración VI – Relación de la velocidad con la diferencia de pulsos 69
- Ilustración VII – Sistema de control en lazo cerrado 70
- Ilustración VIII – Esquema de nuestro sistema de control 71
- Ilustración IX – Diagrama de bloques de un PID 73
- Ilustración X – Parte integral 74
- Ilustración XI – Parte integral con saturación 75
- Ilustración XII – Parte integral y parte proporcional 75
- Ilustración XIII – Parte integral proporcional y derivativa 76
- Ilustración XIV – PID con filtro 77
- Ilustración XV - Cronograma 135
- Ilustración XVI – Diagrama de flujo 136
- Ilustración XVII – Diagrama de flujo final 137

Ilustración XVIII – Soportes realizados para los sensores Hall	147
Ilustración XIX – Enfrentamiento de los sensores Hall respecto a los imanes	147

Índice de Imágenes

Imagen 2.1.1:1 - La Jamais Contente	14
Imagen 2.1.3:1 - Estructura básica de un vehículo eléctrico.....	16
Imagen 2.1.4:1 - Estructura básica de un Vehículo Eléctrico de Baterías.....	17
Imagen 2.1.4:2 – Estructura básica de un Vehículo Eléctrico Híbrido	18
Imagen 2.1.4:3 – Estructura básica de un Vehículo Eléctrico Híbrido Recargable	18
Imagen 2.1.4:4 – Estructura básica de un Vehículo Eléctrico de Pilas de Combustible.....	19
Imagen 2.1.5:1 – Curva característica de un motor asíncrono con control de par.....	20
Imagen 2.2.1:1 – Yamaha Gen-RYU.....	22
Imagen 3.1.1:1 – Yamaha WR450-F2	29
Imagen 3.1.2:1– KTM con motor eléctrico en la rueda delantera.....	31
Imagen 3.1.2:2 – Esquema de la motocicleta de tracción a las dos ruedas de KTM.....	32
Imagen 3.1.3:1 – Christini technology	33
Imagen 7.1.1:1 - Estructura.....	50
Imagen 7.1.2:1 – Colocación de los imanes en las ruedas	51
Imagen 7.1.2:2 – Sensor Hall.....	51
Imagen 7.1.2:3 – Colocación sensores hall	52
Imagen 7.1.3:1 – Funcionamiento de una resistencia Pull-up.....	53
Imagen 7.1.3:2 – Circuito de un filtro paso-bajo.....	56
Imagen 7.1.3:3 – Circuito impreso.....	57
Imagen 7.2.2:1 – Formas en que se puede producir una interrupción	60
Imagen 7.2.2:2 PCICR	61
Imagen 7.2.2:3 – PCMSK0, 1, 2	62
Imagen 7.2.6:1 – Conexión del regulador de potencia al Arduino	80
Imagen 11.1.1 – Oset 20.0 Racing 2015.....	93
Imagen 11.1.2 – Oset 20.0 Eco 2015	93
Imagen 11.1.3 – Oset 16.0 Eco 2015	93
Imagen 11.1.4 – Oset 12.5 R 2015.....	94
Imagen 11.2:1 – Xispa Trial E 25.....	95
Imagen 11.2:2 – Xispa Enducross E 10.....	95
Imagen 11.2:3 – Xispa Supermotard E 10.....	96
Imagen 11.3.2:1 – Motor DC.....	98
Imagen 11.3.2:2 - Rotor de un motor DC.....	99
Imagen 11.3.2:3 - Conmutador de un motor DC.....	99

Imagen 11.3.2:4 - Estator de un motor DC	99
Imagen 11.3.2:5 - Escobillas de un motor DC.....	100
Imagen 11.3.2:6 - Motor BDC.....	102
Imagen 11.3.2:7 - Motor de imanes permanentes	103
Imagen 11.3.2:8 - Estator.....	103
Imagen 11.3.2:9 - Rotor	104
Imagen 11.3.2:10 - Representación de un PMSM con sensores de efecto Hall. Se pueden apreciar lo seis sectores definidos por la combinación de señales Hall, así como los vectores de excitación de cada sector para movimientos clockwise.....	105
Imagen 11.3.2:11 - Representación de los seis estados del inversor.....	106
Imagen 11.3.2:12 - Primera secuencia del inversor	107
Imagen 11.3.2:13 - Segunda secuencia del inversor	107
Imagen 11.3.2:14 - Tercera secuencia del inversor	107
Imagen 11.3.2:15 - Cuarta secuencia del inversor.....	108
Imagen 11.3.2:16 - Quinta secuencia del inversor	108
Imagen 11.3.2:17 - Sexta secuencia del inversor.....	108
Imagen 11.3.2:18 - Motor BLDC	111
Imagen 11.3.2:19 - Motor Brushless Inrunner.....	112
Imagen 11.3.2:20 - Motor Brushless Outrunner	112
Imagen 11.4.7:1 - Pines de alimentación	117
Imagen 11.4.7:2 - Pines de entrada y salida digitales	118
Imagen 11.4.7:3 - Pines analógicos.....	119
Imagen 11.4.8:1 - Relación de los pines con el microcontrolador (1).....	120
Imagen 11.4.8:2 - Relación de los pines con el microcontrolador (2).....	120
Imagen 11.4.8:1 - Alzado de la estructura.....	141
Imagen 11.4.8:2 - Perfil de la estructura	142
Imagen 11.4.8:3 - Planta de la estructura	142
Imagen 11.4.8:4 - Construcción de la estructura	143
Imagen 11.4.8:5 - Estructura pintada.....	144
Imagen 11.4.8:6 - Estructura final con el montaje de los componentes.....	144
Imagen 11.4.8:1 - Tapa Lexan para realizar la jaula de los imanes.....	145
Imagen 11.4.8:2 - Montaje final de la jaula de imanes en el eje de la rueda	146
Imagen 11.4.8:3 - Alzado de la planta de imanes en el eje de la rueda	146
Imagen 11.4.8:1 - Montaje de los sensores Hall en el eje de las ruedas.....	148
Imagen 11.4.8:2 - Visualización del montaje de las ruedas en el soporte... ..	148

Índice de Tablas

Tabla 1 – Cronología de la motocicleta eléctrica	24
Tabla 2 – Formas de actuar del control de tracción sobre el motor delantero	43
Tabla 3 – Conmutador de cuatro posiciones	53
Tabla 4 – Elección del condensador del filtro de paso bajo	56
Tabla 5 – Pines utilizados de Arduino	58
Tabla 6 – Toma de decisiones.....	67
Tabla 7 – Toma de decisiones final.....	68
Tabla 8 – Vectores de conmutación y activación de los sensores Hall	109
Tabla 9 – Comparación BLDC Motor Vs Brushed Motor	111
Tabla 10 – Comparación BLDC Motor Vs Brushed Motor.....	111
Tabla 11 – Especificaciones técnicas de Arduino Uno	116
Tabla 12 – Codificación del DIP Switch C6713 DSK.....	130
Tabla 13 – Tabla de fases	136
Tabla 14 – Tabla de fusión.....	137

Índice de Ecuaciones

Ecuación 1 – Ecuación de un controlador PID	73
---	----

1. Introducción y objetivos

1.1 Introducción

¿Porqué una moto eléctrica de uso dual y que se necesita para la conducción en campo?

La empresa Enriding, ya ha fabricado prototipos de motocicletas eléctricas duales (“trail”) y goza de cierta experiencia en este ámbito.

Uno de los principales artífices de dicha empresa, es el tutor de este proyecto, José Antonio Domínguez. El cual continúa en su afán de desarrollar tecnologías para este tipo de motos eléctricas. De ahí el interés de dicho proyecto.

Este control de tracción, podrá ser implantado en cualquier motocicleta, independientemente del tipo de tracción trasera utilizado. La experiencia previa del tutor recomienda que, dicho proyecto, se enfoque para motos eléctricas de campo infantiles.

Si lo que deseamos es llegar al público de las dos ruedas, podemos aprovecharnos de los puntos débiles de las mismas y afrontarlos con soluciones prácticas, económicas y limpias para, facilitar la conducción de la moto al piloto así como encontrar un hueco en el mercado.

Poniéndonos en la piel de cualquier motorista, hay situaciones en las que la conducción de una motocicleta, es complicada y en las que se desearía poder tener un mejor control sobre la misma para evitar caídas, mejorar su conducción, e incluso tener una serie de ventajas con respecto a la motocicleta convencional como una mejora de respuesta ante ciertas situaciones de riesgo. Todo ello, podría otorgar a la motocicleta una serie de ventajas tanto de estabilidad y de fácil manejo, como de seguridad y mejor agarre al terreno, lo que en definitiva viene siendo una mejor tracción.

Con el control de tracción, que se ha realizado se proporcionará, en este caso, a los niños que se introducen en el mundo de las motos, una conducción más fácil, así como una mayor confianza a sus padres.

La selección del modo de funcionamiento y del tipo de tracción, de la moto a diseñar, debe ser una opción seleccionable por el piloto, para así poder adaptar su pilotaje a las condiciones cambiantes tanto del terreno como de la meteorología, o simplemente para realizar la conducción de una forma más cómoda.

Por lo tanto, nuestro sistema constará de un selector, de los distintos modos de funcionamiento, accionable en cualquier momento durante la conducción.

El control de tracción realizado, sólo actuará sobre el motor de la rueda delantera, en función del modo en el que nos encontremos.

A continuación, se definen los cuatro modos de funcionamiento, seleccionables mediante un interruptor de 4 posiciones. Estos modos son:

- 1- MODO 2X1 OFF.
- 2- MODO 2X1 ON.
- 3- MODO 2X2 OFF.
- 4- MODO 2X2 ON.

El funcionamiento de los distintos modos, será explicado más adelante.

1.2 Objetivos

Para el desarrollo de este control de tracción, será necesario realizar el diseño completo, tanto del hardware como del software.

Para comprobar el funcionamiento del diseño Hardware, se realizará la construcción de un prototipo de laboratorio.

Para el diseño Software, se necesitará realizar la programación en algún tipo de lenguaje para que el control realice lo deseado.

2. Estado actual de la técnica industrial

En las últimas décadas, el hecho de la excesiva utilización del vehículo privado, ha provocado un considerable aumento de la contaminación atmosférica, en las zonas urbanas y periurbanas, donde el tráfico de vehículos es una de las principales fuentes de polución.

El aumento de la contaminación atmosférica, conlleva tanto el calentamiento global debido al efecto invernadero, como a la lluvia ácida. Además, da lugar a otros efectos externos que disminuyen notablemente la calidad de vida de los ciudadanos, como congestión, atascos, ruido, mayor riesgo de accidentes, etc.

Tanta es la utilización de los vehículos, que el consumo de energía del sector transporte supone alrededor del 40% del total de la energía final a nivel nacional. Además se encuentra en constante crecimiento y es superior al de cualquier otro sector económico.

Para evitar este problema, se está trabajando, desde hace tiempo, en la transformación del sistema de transporte actual a uno ambiental y socialmente sostenible como lo es el movido por energía eléctrica.

Otro motivo más por el cual transformar el sistema de transporte actual a uno eléctrico, es debido a la diferencia de eficiencia entre ambos vehículos. Los vehículos de motor de combustión interna, tienen una gran variedad de problemas. Uno de los principales problemas de estos vehículos es que sólo el 25% de la energía del combustible se utiliza para mover el vehículo, el resto son pérdidas en forma de calor, ruido, vibraciones y residuos que se lanzan a la atmósfera. Esto significa que, de cada diez litros de gasolina se desperdician entre 7 y 7 litros y medio.

Sin embargo en el vehículo eléctrico, el 92% de la energía liberada por las baterías sirve para mover el motor, por lo que la eficiencia es entre un 65 - 70% superior con respecto al vehículo convencional con motor de explosión.

Por ello, uno de los tipos de motores que se perfilan como los claros sustitutos de los motores térmicos, son los motores eléctricos, debido a que son altamente controlables, tienen rendimientos elevados, por encima del 90% y sin emisión de residuos a la atmósfera.

Además, cabe resaltar que los motores eléctricos son capaces de proporcionar un mayor par y potencia que los motores de combustión desde el arranque de los mismos, debido a que la curva del par es prácticamente recta. Provocando una mayor eficiencia por poder evitar la utilización de la caja de cambios y el embrague.

En este apartado se va a definir, de forma breve, qué es un vehículo eléctrico, se realizará una clasificación de los vehículos eléctricos según su abastecimiento eléctrico y se comparará el vehículo eléctrico con respecto al de combustión interna.

2.1 Reseña del vehículo eléctrico

2.1.1 Historia

Los coches eléctricos, a finales del siglo XIX y principios del XX, predominaban en el mercado automovilístico, por encima de los coches con motor de gasolina y los coches de vapor. Incluso, por aquella época, poseían récords de velocidad y de distancia recorrida.

El primer vehículo eléctrico conocido, fue un pequeño modelo construido por el Profesor Stratingh, en la ciudad alemana de Groninberg en 1835. Pero los primeros utilitarios, fueron construidos por Thomas Davenport, en los Estados Unidos y por Robert Davison, en Edimburgo en 1842. En principio, estos vehículos usaban células eléctricas no recargables, hasta que se inventó la batería hacia 1881.

En 1899, el belga Camille Jenatzy, consiguió el record de velocidad mundial con un vehículo propulsado mediante motores eléctricos lleno de baterías, llamado “La Jamais Contente”.



Imagen 2.1.1:1 - La Jamais Contente

A finales del siglo XIX, no había un tipo de propulsión (eléctrica, gasolina o vapor) predominante, por ello uno de los visionarios de la época fue H. Piper, que trabajó en la idea de combinar un motor de combustible con otro eléctrico a batería eléctrica, para así obtener una mayor potencia, cosa que sólo con el motor de combustión no se conseguía.

Con el desarrollo de los vehículos eléctricos, en Nueva York en 1897, comenzaron a funcionar los primeros taxis eléctricos y en Europa salieron a la calle los primeros vehículos totalmente eléctricos, como el “Electric Victoria”.

Años después del diseño de H. Piper, la Woods Motor Vehicle Company lanzó al mercado, en 1917, el Woods Dual Power, el primer híbrido puro con frenada regenerativa y que carecía de marchas. Otro ejemplo, de vehículo híbrido, es el Owen Magnetic Model 60, en el cual el motor de gasolina se utilizaba para alimentar de energía la propulsión eléctrica, era uno de los primeros vehículos híbridos de lujo.

Pero con los años, las esperanzas del vehículo eléctrico o su variante híbrida, en lugar de aumentar fueron disminuyendo, debido al desarrollo de los motores de combustión.

2.1.2 Situación actual

Actualmente, el parque móvil nacional tiene aproximadamente 30 millones de vehículos, de los que más de 20 millones son turismos. Por lo que, como media, a cada hogar español le corresponden dos turismos.

Debido a la gran cantidad de vehículos, a la contaminación que provocan y a sus carencias de eficiencias se lleva planteando en la sustitución de estos por otros como lo son los eléctricos.

El gran problema que tiene el motor eléctrico, para sustituir al térmico, es la capacidad de acumulación de energía eléctrica, ya que es muy baja en comparación con la capacidad de acumulación de energía en forma de combustible. Alrededor de 1 kg de baterías puede almacenar tanta energía como unos 20 gramos de combustible, lo que supone una barrera tecnológica importante.

Aunque hoy en día, por primera vez, se dan todas las condiciones que posibilitan la electrificación del sistema de transporte: en primer lugar el desarrollo de las baterías de ión-litio y otros materiales, que permiten la “autonomía necesaria” y en segundo lugar el desarrollo de las energías renovables, que suministran electricidad necesaria sin emisiones de CO₂ ni otros contaminantes perjudiciales para los seres vivos y a un coste inferior que el de la gasolina o gasóleo.

Los vehículos eléctricos de hoy en día pueden llegar a autonomías medias del orden de 160 km, que es mucho menor que lo que se recorre con un depósito de gasolina, pero es una distancia relativamente superior a la que se suele recorrer en una ciudad diariamente. A su favor tienen, su funcionamiento sencillo gracias al alto par de empuje de los motores eléctricos y disponen de

buena aceleración y potencia en ausencia de ruidos, vibraciones y contaminantes.

Por otro lado, los sistemas eléctricos modernos no requieren prácticamente mantenimiento, aunque las baterías deben ser revisadas periódicamente y el sistema de refrigeración limpiado de forma regular.

2.1.3 Definición de vehículo eléctrico

Un vehículo eléctrico, es aquel que se propulsa total o parcialmente mediante un motor eléctrico, alimentado por medio de baterías que se recargan a través de la red eléctrica.

En la imagen 2.1.3:1, se puede observar un pequeño esquema de la estructura básica de un vehículo eléctrico. En ella podemos observar los principales elementos que lo componen, como las baterías, que ocupan casi la totalidad del chasis, colocadas preferiblemente en la zona central del vehículo, debido a que por su volumen y sobre todo por su peso. Es importante su colocación para no interferir en el equilibrado de peso del vehículo.

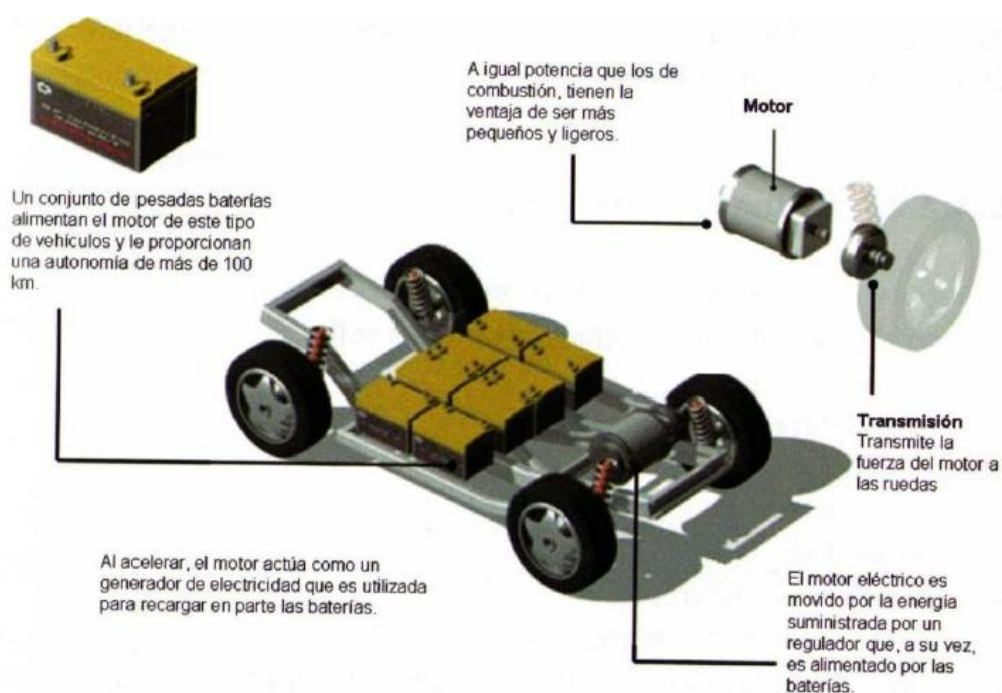


Imagen 2.1.3:1 - Estructura básica de un vehículo eléctrico

El motor eléctrico es el otro elemento más importante del coche, que puede estar ubicado en diferentes partes del vehículo. Además, existe la posibilidad de colocar motores independientes en cada rueda. De esta forma, se pueden obtener vehículos con tracción a las cuatro ruedas o elegir si queremos hacerlo de tracción trasera o delantera, con la ventaja de que la potencia de todos los motores que utilicemos es aditiva, es decir, que se suman todas

ellas, por lo que cuantos más motores tengamos mayor potencia podrá tener el vehículo eléctrico.

Pero ello conlleva la suma de los pesos de los motores, que incrementan notablemente el peso del vehículo, provocando que el consumo de éste pueda verse incrementado.

La electrónica que controla los motores, será más complicada cuantos más motores introduzcamos, pero siempre es más sencillo que realizar lo mismo con elementos mecánicos.

2.1.4 Clasificación del vehículo eléctrico

Se puede hacer una clasificación según el tipo de abastecimiento energético:

1.-Vehículos Eléctricos de Baterías (VEB) (Battery Electric Vehicles)

Son vehículos eléctricos que utilizan la energía química almacenada en unas baterías recargables.

Este tipo de vehículo eléctrico funciona únicamente con motor eléctrico y se pueden recargar las baterías externamente (se pueden conectar a la red eléctrica).

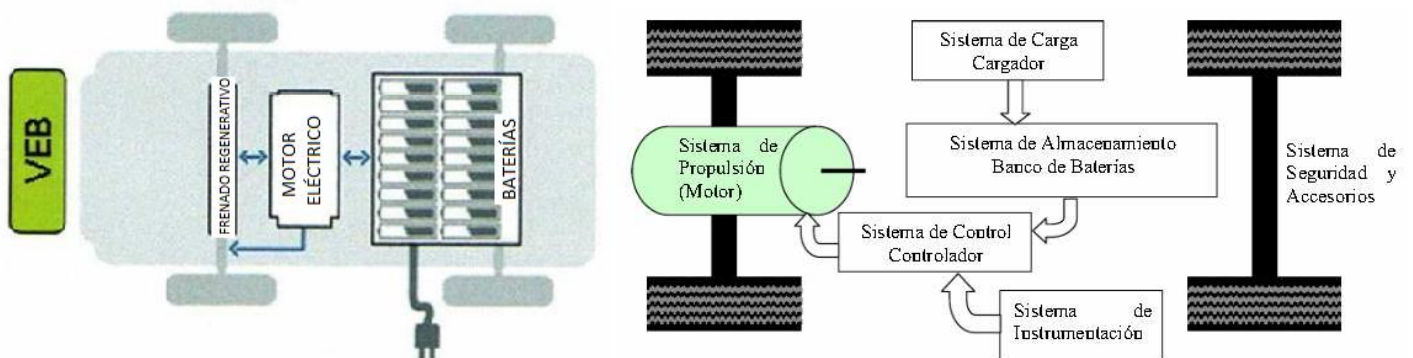


Imagen 2.1.4:1 - Estructura básica de un Vehículo Eléctrico de Baterías

2.-Vehículos Eléctricos Híbridos (VEH) (Hybrid Electric Vehicles)

Son vehículos eléctricos en los que la energía eléctrica, que los impulsa, procede de baterías y alternativamente de un motor de combustión interna que mueve un generador. Normalmente, el motor térmico también puede impulsar las ruedas del vehículo de manera directa.

En este tipo de vehículos no se pueden recargar las baterías externamente, sólo mediante el frenado regenerativo o mediante la conducción en carretera, funcionando el motor eléctrico como generador, al ser arrastrado por las ruedas movidas por medio del motor de combustión.

Algunos no consideran los vehículos híbridos como vehículos eléctricos, puesto que no se ajusta a la definición anteriormente descrita (no se recargan externamente mediante la red eléctrica).

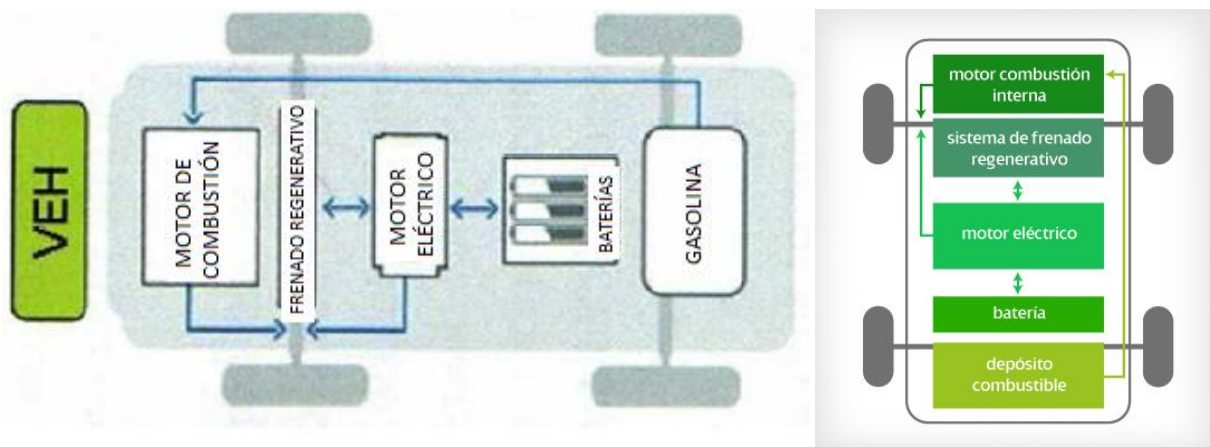


Imagen 2.1.4:2 – Estructura básica de un Vehículo Eléctrico Híbrido

3.-Vehículos Eléctricos Híbridos Recargables o Enchufables (VEHR) (Plug-in Hybrid Electric Vehicles)

Son vehículos eléctricos similares a los anteriores pero con la posibilidad de recargar las baterías por medios externos, es decir, conectándolos a la red.

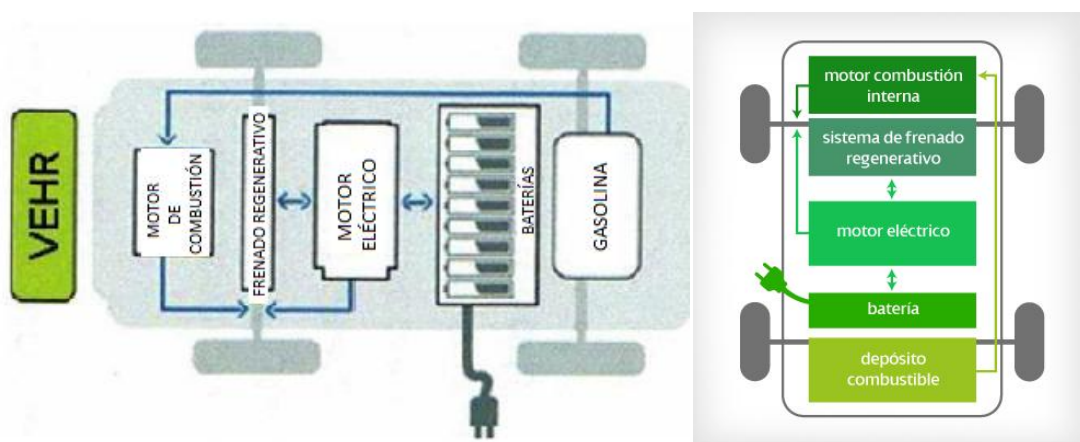


Imagen 2.1.4:3 – Estructura básica de un Vehículo Eléctrico Híbrido Recargable

4.-Vehículos Eléctricos de Pilas de Combustibles (VEPC) (Fuel Cell Electric Vehicles)

Son vehículos de accionamiento eléctrico que disponen de una pila de combustible de hidrógeno, que genera la electricidad a partir de ese gas almacenado en un depósito.

Las pilas de combustible tienen buena respuesta para modos de funcionamiento continuo, por ejemplo en carretera, pero para situaciones de aceleraciones su forma de respuesta es lenta, por ello se utilizan baterías o

ultracondensadores para en esos casos, utilizar la energía almacenada en ellas y así favorecer la respuesta inmediata en esos momentos de aceleración.

Sin embargo, tienen en su contra que el hidrógeno es muy inestable y su almacenamiento no es compacto.

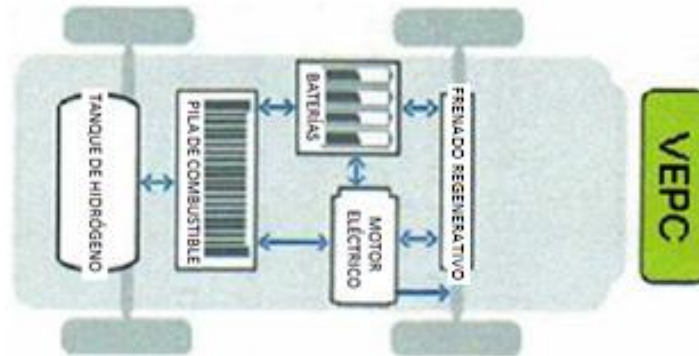


Imagen 2.1.4:4 – Estructura básica de un Vehículo Eléctrico de Pilas de Combustible

2.1.5 Ventajas e inconvenientes del vehículo eléctrico frente al de combustión

Ventajas del vehículo eléctrico

1. La electricidad es un vector energético muy flexible, pudiéndose obtener de numerosas fuentes, incluidas las renovables. Esto reduce los riesgos derivados de una concentración excesiva en una única fuente primaria de energía, como pasa con el petróleo.
2. Son más eficientes que los motores de combustión interna (casi el doble de eficientes).
3. No producen emisiones contaminantes locales, objetivo importante en entornos urbanos, además apenas emiten ruido y vibraciones.
4. Los costes de mantenimiento son menores, debido a la simplicidad del sistema.
5. Son más fiables y duraderos que los motores de combustión interna.
6. Son más fáciles de conducir.
7. Tienen menos sistemas que los de combustión interna.
8. Proporcionan más par a la rueda y más elasticidad que un motor convencional. Son capaces de entregar par y potencia desde el mismo instante del arranque (desde cero) y su curva del par es prácticamente recta (disminuye paulatinamente con el régimen de giro), sin embargo con los motores de combustión no tenemos par en el arranque, por eso en ellos es necesario la utilización de la caja de velocidades y del embrague.

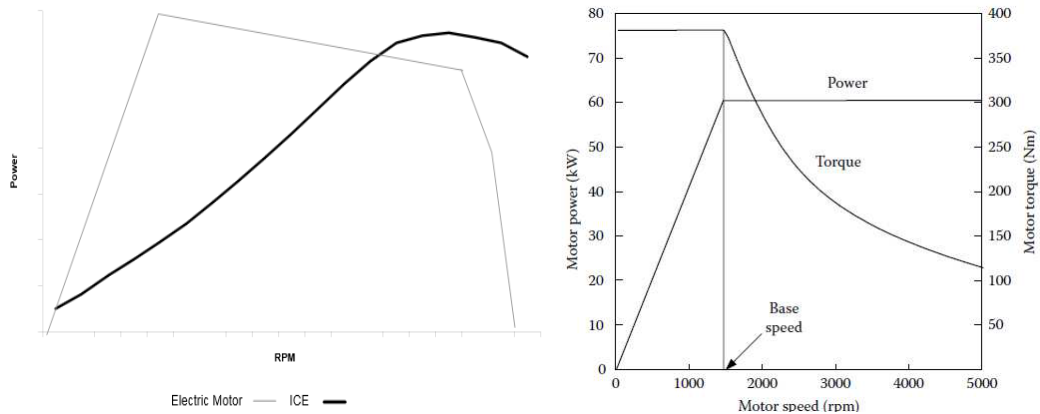


Imagen 2.1.5:1 – Curva característica de un motor asíncrono con control de par

9. Tienen una respuesta más inmediata que los motores de combustión.
10. Se puede recuperar parte de la energía durante las deceleraciones.
11. El sistema de frenado tiene además capacidad regenerativa de la potencia absorbida, lo que reduce las pérdidas de eficiencia.

Inconvenientes del vehículo eléctrico

1. El coste inicial de un vehículo eléctrico es mayor que el de un vehículo de combustión interna de idénticas características, además existe un coste extra, el de las baterías, el cual, dependiendo de la tecnología puede llegar a ser muy elevado. Pero el tipo y la capacidad de la batería determinan la autonomía del vehículo, su velocidad máxima, la duración de la vida útil de la batería y su tiempo de recarga.
2. Generalmente la autonomía de un vehículo eléctrico es reducida y se encuentra limitada por la capacidad de las baterías.
3. Son más lentos que los vehículos de combustión interna en cuanto a velocidad punta máxima (monomarcha).
4. Se trata de una tecnología poco conocida por la mayoría de las personas, aunque esto va cambiando poco a poco.
5. Se requieren instalaciones para la recarga de las baterías y actualmente apenas hay infraestructura de suministro de electricidad en las carreteras.
6. Se requiere también formación especializada de los mecánicos para su mantenimiento.
7. Escasa oferta de vehículos: en el mercado español hasta hace poco no había ningún turismo de la categoría M1 (vehículos destinados al transporte de personas que tengan como máximo ocho personas sentadas además del conductor), siendo todos los modelos del tipo cuadriciclo ligero (L6e) o cuadriciclo pesado (L7e).

2.2 Historia de la moto eléctrica

Antes de empezar con el desarrollo del estudio, se realizará un breve repaso sobre la historia de la motocicleta eléctrica, sus fuentes de energía, fabricantes, comparaciones con las motocicletas de combustión interna y algunos ejemplos de motos eléctricas infantiles actuales para informar sobre el mercado actual.

Una motocicleta eléctrica es un vehículo eléctrico con dos ruedas que utiliza un motor eléctrico como medio de propulsión. La electricidad se almacena a bordo de unas baterías recargables, que acciona uno o más motores eléctricos.

A día de hoy, existen una gran variedad de fabricantes de motocicletas eléctricas como son: Brammo, Zero Motorcycles, Lightning, Quantya, Electric Motorsport, Hollywood Electrics.

Otros fabricantes, van a introducirse o acaban de iniciarse en la fabricación de motocicletas eléctricas. Uno de esos casos es el fabricante de motocicletas Yamaha, que planea entrar en el mercado en breve con al menos dos modelos, otro caso es el fabricante Bultaco, que tiene la intención de entrar en el mercado de la motocicleta eléctrica en este 2015 y por último Harley Davidson, que presentó un nuevo prototipo de motocicleta eléctrica en junio de 2014.

KTM lleva desde hace más de 5 años intentando comercializar motos eléctricas y en este año pretende comercializar las Freeride E-XC y E-SX.

También, BMW lanzó al mercado la scooter eléctrica C Evolución en Alemania, en mayo de 2014 y Honda realizó pruebas de conducción de su scooter eléctrica en 2012, pero aún no ha anunciado su disponibilidad para su venta. Terra Motors, un fabricante de vehículos eléctricos japoneses, comenzará a vender motos eléctricas en la India entre 2015 y 2016.

Por último cabe destacar otro tipo de fabricantes de motos eléctricas, pero en este caso para el sector infantil, como lo son: motos eléctricas Xispa, Oset bikes, Kuberg, Torrotelectric, etc.

2.2.1 Fuentes de energía

El tipo de almacenamiento más utilizado, generalmente, en las motocicletas eléctricas es la batería, pero se han desarrollado algunos prototipos de pilas de Hidrógeno como el ENV de Intelligent Energy, la scooter de Honda Honda FC Stack y la Yamaha FC-AQEL,

Hay varios prototipos experimentales que utilizan pilas de combustible de tecnología ENV (Vehículo de Emisión Neutra), desarrollado por Intelligent

Energy que es un prototipo de pila de hidrógeno. Suzuki, Yamaha y Honda también han desarrollado prototipos de motocicleta de este tipo.

También están en desarrollo prototipos de motocicletas híbridas con motor de gasolina y motor eléctrico. Algunos ejemplos son el Ecycle hybrid, Piaggio MP-Hybrid, la Yamaha Gen-RYU.



Imagen 2.2.1:1 – Yamaha Gen-RYU

No obstante, los modelos de fabricación en serie son de baterías. La mayoría de las motocicletas eléctricas de hoy en día son alimentadas por baterías recargables de ion-litio, aunque algunos de los primeros modelos utilizaban baterías de níquel-metal hidruro. El fabricante de Z Electric Vehicle ha sido pionero en el uso de una batería de silicato de plomo-sodio (una variación de la clásica batería de plomo ácido inventada en 1859, que sigue siendo frecuente su uso en los automóviles), que se compara favorablemente con las baterías de litio en tamaño, peso, y capacidad de energía, por el menor coste.

Todas las motocicletas eléctricas, además proporcionan para la recarga de sus baterías tomas de corriente normales, que por lo general tardan alrededor de ocho horas en cargarse (es decir, durante la noche). Algunos fabricantes han diseñado como accesorio, el cargador de nivel 2 de alta potencia CHAdeMO, que puede cargar las baterías hasta un 95% en una hora. Algunos fabricantes como Zero Motorcycles permiten el intercambio rápido de baterías.

2.2.2 Historia

Cronología	
1895	Surgen las primeras patentes de motocicletas eléctricas.
1911	Popular mechanics publica un artículo sobre la motocicleta eléctrica.
1919	Ransomes, Sims y Jefferies crean un prototipo de motocicleta eléctrica.

1936	Se funda Socovel una empresa de motocicletas eléctricas.
1946	Se fundó una empresa de marketing, basada en una motocicleta eléctrica creada por Merle Williams.
1967	Karl Kordesch realizó una motocicleta eléctrica con pila de hidrógeno. El prototipo de ciclomotor eléctrico "Papoose" es creado por Indian Motorcycle Company, bajo la dirección de Floyd Clymer.
1970	Se vendió el primer cargador de motos eléctricas pequeñas.
1974	Mike Corbin Quick Silver estableció el récord de velocidad con motocicleta eléctrica de 266,165 kmh. Corbin-Gentry Inc. Saca a la venta motocicletas eléctricas legales.
1978	La Harley Davidson MK2 fue creada por Transitron manufactured en Honolulu, Hawaii.
1988	Se crea la motocicleta eléctrica Kawashocki, apareciendo en una revista importante.
1996	Se produce la primera producción masiva de scooters eléctricas lanzada por Peugeot, la Peugeot Scoot ´Elect.
Finales 1990	Scott Cronk y EMB crean la motocicleta eléctrica EMB Lectra VR24, que utiliza motores de reluctancia variable y se empieza a comercializar.
2000	El desarrollo de las baterías de ión-litio y los potentes motores eléctricos hizo que la incorporación de las motos eléctricas sea más factible en esta década.
2007	La motocicleta eléctrica Killacycle, alimentada por baterías de ión-litio, realiza un cuarto de milla en 7.824 segundos y con una velocidad máxima de 270 km/h en Phoenix.
2008	Orlando Tony Parker en Omaha (Nebraska) crea la Electra Green, que es la primera motocicleta eléctrica creada en este estado.
2009	California acoge las "24 horas de motocross eléctrico" seleccionada por Zero Motorcycles. Se realiza la primera Time Trial Xtreme Grand Prix, que era una carrera

	de motos de calle eléctricas.
2011	Chip Yates establece el record Guinness de velocidad con una moto eléctrica con 316,899 km/h.
2013	Primera FIM eRoad Racing World Cup. Motocross: En 2013, la FIM anunció un evento llamado E-MX, que se celebró en Bélgica, lo que podemos llamar la primera carrera de motos de motocross eléctrica. Aunque ya se realizaron otras como Minimoto SX y de las 24 hours of Electricross.

Tabla 1 – Cronología de la motocicleta eléctrica

2.2.3 Motos eléctricas vs motos de gasolina

En este apartado, se van a especificar una serie de ventajas de las motocicletas eléctricas, frente a las motocicletas de combustión interna.

- Las motocicletas eléctricas, presentan una mejor aceleración, ya que desarrollan un par completo de inmediato.
- Las motocicletas eléctricas, sufren una considerable desventaja en cuanto a su autonomía y velocidad punta, ya que las baterías no pueden almacenar tanta energía como un tanque de combustible. Algo más de 210 km con una sola carga de baterías se considera un rango excepcionalmente largo.
- Las motocicletas eléctricas prácticamente no necesitan mantenimiento.
- El coste de la electricidad, para una motocicleta eléctrica, es de entre un céntimo o céntimo y medio por km (dependiendo de las tarifas eléctricas), las motos eléctricas gozan de una enorme ventaja en costos de combustible.
- Los vehículos eléctricos son mucho más silenciosos que los de combustión interna, tan silenciosos que puede sorprender a los peatones desprevenidos. Algunos están equipados para emitir ruido artificial para evitar accidentes con peatones.

2.2.4 Ventajas e inconvenientes de las motocicletas eléctricas

Las motocicletas eléctricas aumentan su popularidad, en la medida en la que aumentan los precios de la gasolina, aunque este año el precio de la gasolina ha bajado por temas meramente coyunturales, esta situación no será definitiva. La tecnología de las baterías mejora de forma continua haciendo más práctico este medio de transporte.

Ventajas:

- No producen contaminación atmosférica.
- No producen contaminación sonora.

- Los costos de combustible, para el motor eléctrico, son aproximadamente el 25% de los costos para el motor de gasolina en 2008.
- La exención del impuesto de matriculación y del impuesto al carbono en algunos países.
- Ningún impuesto de circulación en algunos países.
- El uso de la red eléctrica, para cargar las baterías, causa menos contaminación que las motos de gasolina; la cantidad de emisiones y residuos causados, depende de las fuentes de la electricidad utilizada.
- Se puede utilizar en interiores.
- No se necesitan viajes a la gasolinera. Se pueden recargar en el garaje.
- Mantenimiento reducido y económico. Un ejemplo, no es necesario cambiar el aceite.
- Más asequible que los coches eléctricos.

Inconvenientes:

- Los gastos iniciales, son mayores que en la motocicleta o scooter de gasolina equivalente.
- Menor autonomía.
- Mayor tiempo de repostaje.
- Velocidad máxima inferior a precios similares.
- Las tomas de corriente eléctrica a menudo no se encuentran en las calles o en los garajes. A menudo es difícil o imposible recargarla batería en un aparcamiento.
- La capacidad de la batería puede bajar un 20%, en un día frío de invierno con -20 °C (batería de plomo).

2.2.5 Análisis DAFO de las motos eléctricas

El análisis DAFO es una herramienta que permite realizar un diagnóstico del momento actual del sector mediante la identificación de los aspectos positivos y negativos de la industria y las condiciones en las que se encuentra inmersa.

A continuación se desarrolla el análisis DAFO de las motocicletas eléctricas:

Debilidades:

- Ausencia de una buena sinergia entre los diferentes agentes.
- Distintos planes de desarrollo en cada país y de sistemas de recarga normalizados.
- La visibilidad de la motocicleta es reducida dentro del vehículo eléctrico.
- Desconocimiento de los posibles usuarios.

- Su autonomía y velocidades máximas.
- Los gastos iniciales.
- Pocos sistemas de recarga habilitados en las vías públicas.

Amenazas:

- Situación económica de los países.
- Incapacidad de crear y mantener la infraestructura de recarga necesaria, de manera que se pueda ganar la aceptación del posible usuario y se cree una demanda importante.
- Ausencia de imagen de marca frente a los usuarios.
- Falta de estandarización en los vehículos eléctricos y la infraestructura de recarga.
- El precio de las baterías y la incertidumbre de su funcionamiento.
- Ausencia de una política de apoyo al vehículo eléctrico a largo plazo.

Fortalezas:

- Las motocicletas eléctricas no emiten contaminantes, por lo que ayudan a disminuir la contaminación en el ámbito urbano.
- Las motocicletas eléctricas no emiten ruido, por lo que ayudan a disminuir el nivel de ruido.
- El producto es económicamente competitivo en comparación con la motocicleta convencional a largo plazo.
- Su mantenimiento es más reducido y económico.
- La tecnología de las baterías está en pleno desarrollo.

Oportunidades:

- Presión legislativa sobre los vehículos combustión interna.
- Preocupación del usuario por temas medioambientales.
- Incremento de las iniciativas de energías renovables.
- Alza e inestabilidad del precio de los combustibles fósiles.
- Prohibición o imposición de tasas a vehículos contaminantes en el acceso a las ciudades.
- El crecimiento de los núcleos urbanos aumentará las necesidades de movilidad en ese ámbito.
- El desarrollo de las baterías mejorando su capacidad y energía.

2.3 Motos eléctricas infantiles ya existentes

La implantación del control de tracción, se desea realizar sobre una motocicleta eléctrica infantil, por ello se investiga el mercado actual de este sector.

Unas de las principales empresas, especializadas en la fabricación de motocicletas eléctricas de trial o para conducción en campo son: Oset bikes y Xispa Motorbikes. En este apartado vamos a comentar brevemente los tipos de motocicletas que fabrican a modo de investigación.

Oset bikes

Las motos eléctricas de trial OSET están en el mercado a nivel global desde el año 2006, y han transformado completamente el proceso de aprendizaje de los pilotos jóvenes. Hay dos tamaños disponibles: uno con ruedas de 31.75 cm/diámetro (12.5") para niños de 2 a 5 años, y otro con ruedas de 40.64 cm/diámetro (16") para niños de 5-10 años.

Mientras que los motores eléctricos no tienen potencia suficiente para llevar a un adulto en terrenos off-road, si tienen potencia suficiente para los niños. Estas motocicletas, pilotadas por niños pequeños, pueden funcionar durante dos horas con una batería simple.

Por encima de todo esto, las motos OSET se venden a un precio bastante inferior a las de motor de gasolina de 50cc. Además son ligeras, potentes, ajustables mediante control parental.

Cada moto tiene un control ajustable de la velocidad, de esta manera el adulto responsable puede limitar la velocidad desde cero hasta full-speed. Y cuentan con un sistema de suspensión ajustado al peso de los niños.

Algunos de los diferentes tipos de motos eléctricas de la empresa Oset Bikes, se adjuntan en el Anexo "Tipos de motos eléctricas Oset Bikes".

Xispa motorbikes

Esta empresa, con sede en Girona, está especializada desde 2005 en la fabricación de motos de trial y motos infantiles, productos que recientemente ha lanzado reconvertidos en eléctricos.

Las motos fabricadas por Xispa, cuyo peso, potencia y dimensiones están pensadas para niños de unos 3-7 años.

Algunos de los diferentes tipos de motos eléctricas de la empresa Xispa Motorbikes, se adjuntan en el Anexo "Tipos de motos eléctricas Xispa Motorbikes".



3. Estudio de las soluciones existentes

El por qué de la elección de la utilización de la doble tracción tiene su clara explicación.

3.1 ¿Por qué una moto bimotor?

La diferencia es clara, con una motocicleta convencional el tren delantero va más o menos por donde se le pide, pero el trasero depende de la potencia aplicada, del tipo de terreno, de la situación del peso del piloto, etc. Si la rueda delantera encuentra un obstáculo, tiende a frenarse y a ser adelantada por la rueda trasera, es decir, la moto se cruza.

En el caso de una motocicleta de doble tracción, esto cambia notablemente: A velocidad muy alta, aun circulando en barro o en arena muy suelta, se mantiene firme como si rodara sobre raíles. Y al pegar fuerte con piedras o raíces de los árboles, la tracción ayuda a sobrepasarlas con mayor facilidad.

También en situaciones de cambios de rasante con mucha pendiente o en terrenos resbaladizos o mojados, cuando la rueda trasera empieza a patinar y se empieza a perder tracción, podríamos sufrir una caída. En estos casos, también la tracción delantera ayudaría a un mejor agarre y estabilidad para sobrepasar esa situación de peligro.

3.1.1 Reseña Yamaha

En el año 2003, en uno de los movimientos más importantes en la historia de la motocicleta, Yamaha anunció que lanzaría una moto de dos ruedas motrices a principios de 2004. Por supuesto, no es el primer intento de tracción total en la historia de la motocicleta, pero la noticia causó gran revuelo en el mundo del motor, ya que, parecía que Yamaha había dado en el clavo con la moto de tracción total y estuvo dispuesto a ser el primero en venderlas en sus concesionarios.



Imagen 3.1.1:1 – Yamaha WR450-F2

Aunque la primera motocicleta, que se ofrece con el sistema de Yamaha 2-Trac, es una moto de Todo Terreno basada en la Yamaha WR450F, anteriormente Yamaha realizó pruebas con una R1 1000cc de carretera, con las que demostró una diferencia increíble, en velocidad sobre el asfalto mojado.

La investigación de este sistema de tracción total se inició en secreto en 1985, pero posteriormente, en 1998, un prototipo 2-Trac equipado en la YZ250 de cross fue revelado a la prensa. Al año siguiente, Yamaha corrió con una 2-trac-TT600R en el UAE Desert Challenge. Posteriormente también se pudo ver la motocicleta Wr450F 2-Trac participando en algunos rallies con resultados discretos, y con el transcurso del tiempo parece que en competición no cuajó demasiado el invento, ya que parece que cuanto más inexperto sea el piloto, mayor beneficio obtendrá de estos sistemas de tracción. KTM poco después también realizó algún tímido intento de equipar alguno de sus modelos con esta tracción delantera.

¿Cómo funciona la Yamaha WR450-F2 con tracción a las dos ruedas?

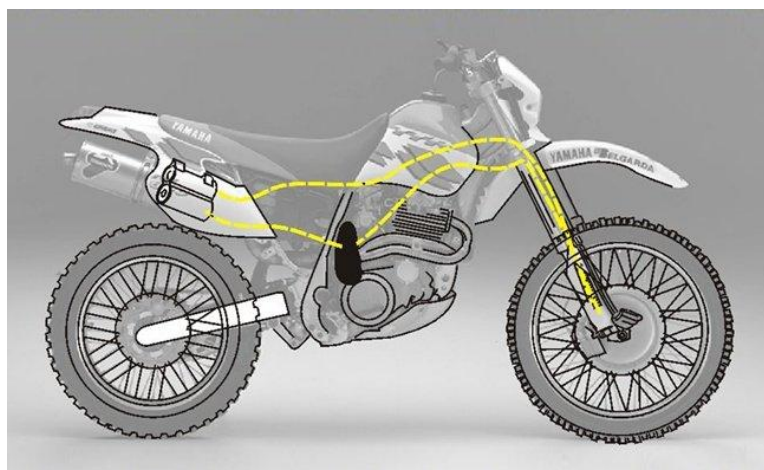


Imagen 3.1.1:2 – Yamaha WR450-F2 esquema del flujo hidráulico

El sistema, patentado con el nombre 2-Trac, utiliza una bomba hidráulica situada encima de la caja de cambios que bombea fluido hidráulico a través de unos conductos de alta presión hasta el buje delantero, el cual alberga un pequeño motor hidráulico de un solo engranaje. Con esto se consigue que, de una forma totalmente independiente, el motor proporcione más presión hidráulica a la rueda delantera, conforme la trasera va perdiendo tracción, aumentando así la potencia de tracción de la rueda delantera. La distribución de la potencia, entre la rueda delantera y trasera, es variable con el fin de optimizar la tracción.

La rueda delantera no puede girar más rápido que la rueda trasera, y la potencia transmitida a la rueda delantera, no es nunca superior a la utilizada

para la parte trasera. Este sistema auto-regulado permite que la tracción de la rueda delantera, se reduzca lentamente para que la rueda trasera vaya recuperando la tracción.

Cuando se cierra la válvula reguladora, no se transmite potencia a la rueda delantera, pero si la válvula se abre de repente y la rueda trasera empieza a perder adherencia, el repentino aumento de revoluciones de la bomba, aumenta la presión hidráulica del sistema y una mayor proporción de la potencia del motor se transmite a la rueda delantera. Si la rueda trasera continúa girando, más potencia es enviada a la rueda delantera. La proporción de potencia enviada a la rueda delantera es por lo tanto, controlada por la válvula reguladora de la tracción de la rueda trasera.

Las principales diferencias, respecto de la WR 450 F estándar se aprecian en el lado izquierdo del motor (con la bomba hidráulica y el mecanismo que alberga la cadena dentada) y en los latiguillos anchos que recorren toda la parte delantera hasta el buje (un buje es el elemento de una máquina donde se apoya y gira un eje), también de tamaño muy particular. El resto de diferencias son el equipo de suspensiones, el amortiguador de dirección Öhlins y el diseño específico de los gráficos y plásticos protectores. La doble tracción no se puede desconectar, ni está previsto.

3.1.2 Reseña KTM

KTM realizó planes para comercializar a partir de 2010 una moto eléctrica off-road, la marca austríaca ahonda en la posibilidad de dotar a sus motos de tracción total (2x2) gracias a un pequeño motor eléctrico. Sería la primera moto híbrida de KTM.



Imagen 3.1.2:1– KTM con motor eléctrico en la rueda delantera

KTM y Yamaha han desarrollado con anterioridad sistemas de tracción 2x2 basados en la hidráulica. La diferencia de esta 2x2 de KTM con la que

Yamaha tuvo en su momento en el catálogo de productos (2-Trac) –con el desarrollo de la marca sueca de suspensiones Öhlins-, radica en que no se trata de un motor hidráulico el que aporta la energía que necesita la rueda delantera, sino un pequeño motor eléctrico instalado en el buje. Ha patentado un sistema por el que un “minimotor” eléctrico, instalado en el buje de cada rueda, acaba dando aquel par adicional que la moto necesita. La gran ventaja respecto al sistema alternativo hidráulico es la reducción de peso del conjunto y que el motor no pierde potencia, ya que podría activarse el sistema a voluntad del piloto, si fuera necesario.

Por lo tanto, dicha moto alberga un motor de combustión para la rueda trasera y además dos motores eléctricos, uno en cada rueda para dar el par adicional.

El líder del proyecto en KTM, Wolfgang Felber, está actualmente trabajando en el sistema para intentar mejorar los primeros esbozos con esta nueva tecnología. A falta de información directa, o indirecta de fábrica, sólo comentar que la marca austriaca ya probó el 2x2, hidráulico, con el manager Kurt Nicoll en una prueba de enduro extremo en Austria (Erzberg), en 2004, aunque ello fue con una moto de cilindrada media. En las patentes, el nuevo sistema está instalado, en una KTM Rally.

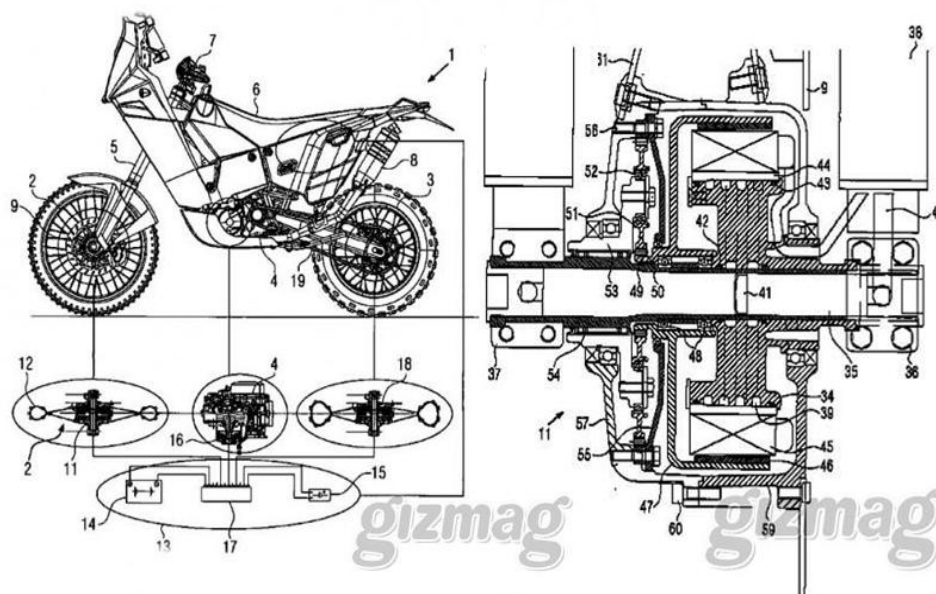


Imagen 3.1.2:2 – Esquema de la motocicleta de tracción a las dos ruedas de KTM

3.1.3 Reseña Christini technology

Otra opción al sistema 2-Trac de Yamaha es la transformación, de forma totalmente mecánica, de una motocicleta convencional a una moto con tracción a las dos ruedas.

La empresa americana Christini Technologies, es especialista en dotar a las motos de Enduro y Cross de tracción integral. Por unos 3500 euros, dependiendo del modelo y año, transforman una motocicleta convencional, incluyendo toda la preparación específica, para dotarla de tracción en la rueda delantera por vía mecánica.

El sistema se basa en un piñón doble del que sale una cadena hacia una caja de transferencia, que empalma con una junta cardan. La junta cardan transmite el movimiento a otra caja debajo de la horquilla (cristo) de la dirección. En esa caja, la transmisión de movimiento la distribuye a otros dos cárdenes telescópicos, que son los que hacen girar a la rueda delantera.



Imagen 3.1.3:1 – Christini technology



Desarrollo del TFG

4. Planteamiento de solución electrónica mediante motores eléctricos en los cubos de las ruedas

Una vez realizada la introducción sobre las motocicletas eléctricas se plantea una solución para el desarrollo del control de tracción.

En este caso se propone como solución de tracción más viable, la utilización de motores eléctricos de corriente continua por proporcionar numerosas ventajas, como ya se explicó en el apartado anterior.

Además, cabe recalcar que el sistema que se ha realizado, es totalmente válido para cualquier tipo de tracción que utilice la motocicleta. Para transformar una motocicleta convencional en una motocicleta con tracción 2x2 con nuestro sistema, se necesita de:

- Un motor eléctrico que accione la rueda delantera.
- Y un control de tracción, diseñado de forma totalmente electrónica, que gobierne el accionamiento del motor eléctrico, como lo es el que se diseña en este proyecto.

Los motores eléctricos elegidos son motores de tipo brushless DC, debido a sus numerosas ventajas respecto otros motores eléctricos como son:

- Buena capacidad de respuesta.
- Elevada vida útil.
- No requieren mantenimiento.
- Son silenciosos.
- Alta eficiencia.
- Son más baratos de fabricar y su peso suele ser inferior a otros tipos de motores.
- Gozan de una gran relación velocidad – par.

Para más información sobre cómo funcionan este tipo de motores, se incluye el anexo “Tipos de Motores DC, los Motores Brushless”.

En cuanto a su instalación en el vehículo, podremos distinguir entre aquellos que van anclados a un chasis y los que se sitúan en el cubo (bujes) de las ruedas, Hemos elegido esta última opción puesto que para pequeñas potencias, representan un ahorro en espacio, peso y simplicidad del vehículo.



5. Diseño del algoritmo y simulación mediante Simulink

Para poder conseguir nuestro propósito, primero se tiene que realizar el diseño de un algoritmo.

Posteriormente se observarán los posibles problemas o dificultades que puedan surgir, tanto en el desarrollo del algoritmo como en la coherencia del mismo. De los problemas encontrados, se seleccionará la solución más conveniente, en función del resultado final que se desee obtener y de los elementos disponibles

Por último, se expresará el algoritmo mediante un programa en un tipo de lenguaje de programación, con el que se realizarán todo tipo de pruebas para comprobar la capacitación de dicho sistema.

5.1 Definición básica del algoritmo a realizar

En orden a la mejor realización y explicación de nuestro algoritmo, primero se tendrán que definir las entradas del sistema, el proceso a realizar por dicho sistema y las salidas que se obtendrán del mismo.

5.1.1 Entradas

Las entradas de las que va a constar este proyecto son:

- Modo de funcionamiento del control de tracción. (Cuatro posibles modos de funcionamiento)
- Velocidad de cada rueda.
- Posición del acelerador.
- Otras entradas opcionales que nos permitirán una mayor flexibilidad y adaptación del control son:
 - o Limitación de la potencia de la tracción delantera (como se ha observado anteriormente en las motocicletas eléctricas infantiles ya existentes).
 - o Una entrada, para la utilización de una llave de seguridad, que el piloto lleve atada a la muñeca. Así, en caso de caída, el sistema se apagará y la motocicleta dejará de funcionar y se parará.
 - o Otra entrada para el encendido del sistema mediante un interruptor de Start/Stop.

5.1.2 Proceso

El objeto del proceso, es proporcionar una tensión determinada al motor delantero del vehículo para que ejecute el modo de tracción seleccionado. Por ello, lo primero a tener en cuenta es la elección del modo de funcionamiento del sistema de tracción.

En función del modo se realizará (excepto en el modo 2x1 OFF) la lectura de velocidades de cada rueda para su posterior comparación y en función de la diferencia entre ellas, calcular el valor de la señal del acelerador de tal forma que el motor delantero haga las acciones correspondientes.

Por último, recordar que otras señales de entrada, también deberán tenerse en cuenta a la hora de tomar la decisión, como lo es la regulación de la potencia de la tracción delantera y la señal del acelerador trasero.

5.1.3 Salida

La salida que se obtendrá del sistema, será el accionamiento de la rueda delantera para equiparar la velocidad de esta con la de la rueda trasera.

5.1.4 Simulación mediante Simulink

Una vez, que ya se ha identificado cuáles son las entradas y salidas del proceso, se va a realizar una simulación por medio de Simulink. Para ello se va a desmenuzar el problema a resolver en rutinas, para la posterior realización del algoritmo.

Con dicha simulación se podrá observar ciertas situaciones, que sin ella no podríamos apreciar y profundizar más en estas, que no se tenían en cuenta anteriormente, para estudiarlas y poner la solución más adecuada antes de realizar el diseño del algoritmo.

En el anexo “Simulación mediante Simulink”, se adjunta la simulación mediante Simulink, junto con la explicación de la realización de la misma, como de su funcionamiento

5.2 Diseño del algoritmo

¿Cómo controlar una motocicleta 2x2?

Para realizar el control de tracción de una motocicleta 2x2, se necesitará accionar la tracción delantera en función de unas señales de entrada y de unas decisiones tomadas por el mismo control de tracción.

Al subdividir el problema final en rutinas, en el algoritmo diseñado por lo tanto se podrán observar las mismas.

En esta etapa de diseño del algoritmo, se determina cómo el sistema desarrollará y resolverá la tarea solicitada. Uno de los métodos más eficaces para este proceso de diseño, se basa en el conocido por divide y vencerás. Este método consiste en, la resolución de un problema complejo dividiéndolo en subproblemas o rutinas, los cuales además se pueden dividir en otros de nivel más bajo.

Por lo tanto, el programa a diseñar tendrá un diseño modular, que constará de un programa principal que irá llamando a esas rutinas o subprogramas.

De esta manera, las rutinas o módulos pueden ser elaborados, comprobados, programados, depurados independientemente y posteriormente combinarlos entre sí, resultando mucho más fácil la resolución del problema final.

El algoritmo se va a dividir en cinco rutinas totalmente definidas:

- 1- Selección del modo.
- 2- Contabilización de flancos ascendentes originados por la señal de los sensores, siempre que sea necesaria.
- 3- Comparación del número de flancos ascendentes contabilizados de cada rueda, siempre que sea necesaria.
- 4- Toma de decisiones realizada por el control de tracción
- 5- La actuación, del sistema, sobre el motor delantero:

1. Selección del modo: En este sistema se podrá seleccionar el tipo de conducción deseado, elegido en función de las preferencias del conductor tanto por comodidad, condiciones del terreno y seguridad como por ahorro de energía, debido a que los distintos modos de funcionamiento, van a tener un mayor o menor consumo de energía en función del mayor o menor uso de la doble tracción.

Los modos de funcionamiento entre los que el piloto podrá elegir van a ser cuatro:

- Modo 2x1 off:
- Modo 2x1 on:
- Modo 2x2 off:
- Modo 2x2 on:

Estos cuatro modos se pueden agrupar en dos grandes modos:

- Modo 2x1: El vehículo funcionará con tracción convencional, a excepción de que se produzca pérdida de tracción en la rueda trasera cuando esté seleccionado el modo 2x1 on.
- Modo 2x2: El vehículo circula siempre con tracción a las dos ruedas.

A continuación, se realiza la explicación de los 4 modos de funcionamiento:

- **Modo 0, o modo 2x1 OFF:** Sólo la tracción trasera entrará en funcionamiento. Realmente este modo es equivalente al modo de funcionamiento de una motocicleta convencional. Lo único que le difiere es que, esta motocicleta tiene un motor eléctrico brushless en la rueda delantera. Los motores brushless, al estar formados por un

rotor de imanes permanentes y un estator de devanados, debido al efecto de la magnetización entre ellos, cuando se produce el giro del rotor respecto al estator, sin que el motor esté alimentado, este opone una resistencia al movimiento. Por lo tanto, para que el funcionamiento sea idéntico al de una moto convencional se le tiene que proporcionar al motor delantero una tensión mínima para que la rueda delantera gire libremente sin oponer resistencia al movimiento, por ello no se considera como si la tracción delantera estuviera accionada.

- **Modo 1, o modo 2x1 ON:** Este modo de funcionamiento será seleccionado, cuando se desee accionar la tracción delantera sólo en situaciones de pérdida de tracción en la rueda trasera.

Cuando la motocicleta se encuentre en una situación de funcionamiento normal o de rueda delantera en el aire (la rueda delantera gire más deprisa que la trasera), se le dará una tensión mínima al motor delantero (exactamente igual que en el modo 0). Pero cuando se produzca una situación de pérdida de tracción en la rueda trasera (la rueda trasera gire más deprisa que la delantera), se accionará el motor delantero hasta que se corrija el error pudiendo llegar al 100% de la limitación prefijada por el piloto, es decir, al motor delantero se le proporcionará una tensión hasta que estas giren aproximadamente a la misma velocidad.

Se ha comprobado a posteriori que esta acción de control, debe permanecer activa durante un tiempo mínimo, este tiempo en las pruebas fue de tres segundos, tanto para superar el obstáculo, como para evitar cambiar constantemente la acción sobre el motor, excepto cuando la rueda delantera gire mucho más deprisa que la rueda trasera. En este caso se cambiará inmediatamente la forma de actuación sobre el motor delantero y se le proporcionará una tensión mínima por motivos de seguridad.

Este modo de funcionamiento, será seleccionado para tener un mayor agarre en situaciones de pérdida de tracción en la rueda trasera.

- **Modo 2, o modo 2x2 OFF:** Se accionará la tracción delantera de la motocicleta un tanto por ciento proporcional al accionamiento del acelerador, a excepción de que la rueda delantera gire más rápido que la rueda trasera. Cuando la rueda delantera, se encuentre en el aire y gire a mayor velocidad que la trasera, entonces se dejará la rueda delantera libre proporcionando una mínima tensión al motor delantero (como en el modo 2x1 off y 2x1 on).

Con este modo de funcionamiento, se podrá tener un cierto agarre tanto en situaciones normales como en situaciones de pérdida de tracción en la rueda trasera.

- **Modo 3, o modo 2x2 ON:** La tracción delantera entrará en funcionamiento siempre, a excepción de que la rueda delantera gire más deprisa que la rueda trasera. Ahora bien, la forma de funcionar depende de la situación en la que nos encontremos. Si nos encontramos en funcionamiento normal, la tracción delantera se accionara un tanto por ciento proporcional al accionamiento del acelerador, por otro lado, si nos encontramos en situación de pérdida de tracción en la rueda delantera entrará en funcionamiento al 100%, hasta igualar las velocidades de las ruedas.

Se puede observar una clara diferencia entre ellos, sobre todo entre el modo 0 y el resto. En el modo 0, siempre se actúa de la misma forma y sin embargo, en el resto, siempre se requiere la comparación de las velocidades de las ruedas, para que el control de tracción pueda tomar una decisión sobre el funcionamiento en el que se encuentra la motocicleta.

Esta observación nos será de mucha ayuda a la hora de realizar el diagrama de flujo.

2. Contabilización de flancos ascendentes: La lectura de velocidades de las ruedas, o conteo de flancos positivos detectados por los sensores hall, debido al enfrentamiento con los imanes de las ruedas es lo primero que se debe realizar para la posterior comprobación de las velocidades de cada rueda. Para ello se utilizarán unos sensores hall, con los que se obtendrán unos pulsos, provocados por el enfrentamiento con unos imanes distribuidos por las ruedas. De esta forma se podrá obtener la velocidad, pero se optó por la utilización y comparación de los flancos positivos de los pulsos directamente, pues su tratamiento es mucho más sencillo, rápido y directo.

3. Comparación de velocidades: Para realizar la toma de decisiones, sobre la situación de funcionamiento en que se encuentra circulando la motocicleta en ese instante, se necesita comparar las velocidades de las ruedas y con ello dictaminar si es necesario o no, que la tracción delantera entre en funcionamiento.

Esta comparación de velocidades (flancos positivos contabilizados) se realizará:

- Cíclicamente, cada cierto periodo de tiempo.
- O al superar un número de flancos positivos máximo contabilizados.

El por qué de dos formas distintas para realizar la comparación de velocidades, es debido a que a velocidades bajas, la comparación será necesaria realizarla cada cierto tiempo, ya que si fuera por un número determinado de flancos positivos máximo contabilizados, quizás pase

excesivo tiempo y ya sea demasiado tarde para la actuación del sistema. Pero si la velocidad es alta, la comparación de velocidades entonces tendrá que realizarse por número máximo de flancos contabilizados, ya que si se realizara cada cierto tiempo, igual que cuando se circula a bajas velocidades, quizás el tiempo sea demasiado extenso para realizar la comparación y demasiado tarde para actuar ante alguna situación comprometida.

4. Toma de decisiones: En función de la diferencia de velocidades o de los flancos positivos contabilizados por cada sensor y del modo seleccionado, se tendrá que decidir la forma de actuación sobre el motor de la rueda delantera, para que este se accione o no y de qué forma hacerlo.

El sistema realizado, tomará una serie de decisiones para determinar la tensión que proporcionará al motor delantero.

La primera decisión que determinará será la situación en la que la motocicleta se encuentre circulando en cada momento.

La situación, en la que el sistema determinará en la que se encuentra la motocicleta, dependerá de la diferencia de velocidades entre las ruedas de la misma.

Los tipos de situaciones, entre las cuales nuestro sistema podrá determinar el funcionamiento actual de la motocicleta son los siguientes:

- **Situación normal:** Se determinará que nos encontramos ante este tipo de funcionamiento, cuando entre las ruedas apenas haya diferencia de velocidad. Se especificará, un margen de flancos ascendentes contabilizados por cada sensor, para el cual nuestro sistema determine que se está circulando con la motocicleta en una situación normal.
- **Situación de pérdida de tracción en la rueda trasera:** Se determinará que nos encontramos ante este tipo de funcionamiento, cuando la rueda trasera supere en una cierta cantidad de velocidad a la rueda delantera. Se comparará el contaje de flancos ascendentes contabilizado de cada rueda y se especificará un margen de error en el cual consideraremos que la situación se está produciendo.
- **Situación de rueda delantera en el aire:** Se determinará que nos encontramos ante este tipo de funcionamiento, cuando la rueda delantera supere en una cierta cantidad de velocidad a la rueda trasera. Se comparará el contaje de flancos ascendentes contabilizado de cada rueda y se especificará un margen de error en el cual consideraremos que la situación se está produciendo.

Una vez visto los tipos de situaciones, entre los cuales el sistema podrá determinar el funcionamiento de la motocicleta, el sistema tomará una **segunda decisión**, en función del modo seleccionado y de la situación que haya determinado, con la cual, junto con la acción del regulador de potencia y en algunas ocasiones también del accionamiento del acelerador, con la que determinará la señal de salida comandada al motor eléctrico delantero.

A continuación se procede a la definición gráfica mediante una tabla, de la forma de actuar del control de tracción, sobre el motor delantero, en función del modo de funcionamiento, del control de tracción, seleccionado y de la situación en la que, el control de tracción determine que se encuentra la motocicleta:

FUNCIONAMIENTO SITUACIÓN	MODO 2x1 OFF	MODO 2x1 CONTROL DE TRACCIÓN	MODO 2x2 OFF	MODO 2x2 CONTROL DE TRACCIÓN
NORMAL	Compensar esfuerzo(*)	Compensar esfuerzo(*)	Tracción % trasera(**)	Tracción % trasera(**)
PÉRDIDA DE TRACCIÓN EN LA RUEDA TRASERA	Compensar esfuerzo(*)	Corregir 100% (3seg)(***)	Tracción % trasera(**)	Corregir 100% (3seg)(***)
RUEDA DELANTERA EN EL AIRE	Compensar esfuerzo(*)	Compensar esfuerzo(*)	Compensar esfuerzo(*)	Compensar esfuerzo(*)

Tabla 2 – Formas de actuar del control de tracción sobre el motor delantero

(*) Compensar esfuerzo, equivale a proporcionar una tensión mínima al motor delantero, para que no oponga resistencia al movimiento.

(**) Tracción % trasera, equivale a proporcionar una tensión al motor delantero, un tanto por ciento proporcional al accionamiento del acelerador.

(***)Corregir 100%, equivale a proporcionar una tensión al motor delantero, proporcional a la diferencia de flancos positivos contabilizados entre las ruedas, para conseguir que la rueda delantera gire a la misma velocidad que la rueda trasera.

5. Formas de actuar sobre el motor delantero: Una vez que sabemos cómo queremos actuar sobre el motor delantero, nos falta saber cómo poder actuar sobre el motor delantero, para conseguir nuestro propósito.

Como se ha podido observar, seleccionado el **MODO 2x1 off**, la forma de actuación de nuestro sistema, sobre el motor delantero, va a ser siempre la misma: Sea cual sea la situación en la que la motocicleta este circulando en ese momento, la actuación sobre el motor va a ser dar una tensión mínima.

No se realizará ni la “comprobación de velocidades” de cada rueda, ni su posterior comparación.

Además en este modo, ni la señal de entrada del regulador de potencia ni la del acelerador, tendrá ninguna consecuencia sobre la señal de actuación sobre el motor delantero.

Si seleccionamos cualquier otro modo, distinto al modo 2x1 off, la forma de actuar sobre el motor delantero, variará en función de la situación en la cual se encuentre la motocicleta circulando, por lo que en estos modos, si será necesario el contaje de flancos ascendentes por los sensores y la posterior comparación de estos.

Seleccionado cualquier modo de estos, si el sistema determina, que la motocicleta está circulando en la situación de rueda delantera en el aire, la actuación sobre el motor delantero va a ser como en el modo 2x1 off, es decir, dar una tensión mínima a la rueda delantera. Por lo tanto ni la señal de entrada del regulador de potencia ni la del acelerador tendrá ninguna consecuencia sobre la señal de actuación sobre el motor delantero.

Seleccionado el **MODO 2x1 on**, si la situación, que determine el sistema en la que circula la motocicleta, es normal, la forma de actuar sobre el motor delantero es proporcionando una tensión mínima a este. Y por lo tanto las señales de entrada tanto del acelerador como del regulador de potencia no tendrán ningún efecto, en la señal de actuación sobre el motor delantero.

Pero si, en este modo de funcionamiento, el sistema determina que la motocicleta circula en una situación de pérdida de tracción en la rueda trasera, el control de tracción proporcionará una tensión al motor delantero, proporcional a la diferencia de flancos ascendentes entre las ruedas de la motocicleta, con la intención de igualar las velocidades de las mismas y así superar satisfactoriamente la “situación de peligro”.

En esta situación, la señal de entrada del regulador de potencia si tendrá actuación sobre la señal proporcionada al motor delantero, consiguiéndose así poder actuar sobre la potencia ofrecida a la tracción delantera.

Por último resaltar que, la señal de actuación sobre el motor delantero, en la que se le proporciona una tensión proporcional a la diferencia de flancos ascendentes entre las ruedas, se mantendrá durante 3 segundos. Esto se ha decidido, debido a las pruebas realizadas mediante la simulación con Simulink, para tanto poder salvar perfectamente la situación de riesgo, como para no realizar cambios constantes en la señal de entrada al motor delantero ante situaciones de pérdida de tracción en la rueda trasera.

Pero si dentro de ese periodo de tiempo, el sistema detecta la situación de que la rueda delantera esté en el aire, la actuación sobre el motor delantero cambiará inmediatamente a proporcionar una tensión mínima al motor delantero por razones de seguridad.

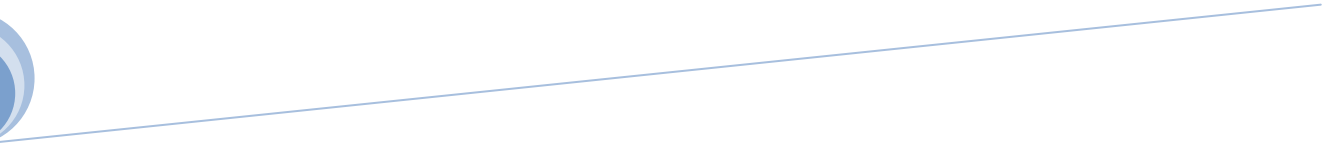
Pasados esos 3 segundos, si el sistema determina que nos encontramos ante una situación de pérdida de tracción en la rueda trasera, nuevamente, se continuará proporcionando esa señal, hasta que detecte otra situación de funcionamiento de la motocicleta diferente, en cuyo caso, la señal de actuación sobre el motor cambiará inmediatamente de la forma pertinente.

Seleccionado el **MODO 2x2 off**, la forma de actuación sobre el motor delantero va a ser la misma, tanto si el sistema determina que la motocicleta se encuentra circulando en una situación normal como en una situación de pérdida de tracción en la rueda trasera. En cuyo caso se proporcionará, al motor delantero, una tensión un tanto por ciento proporcional al accionamiento del acelerador. En este caso, tanto la señal de entrada del acelerador como la del regulador de potencia afectarán a la señal comandada al motor delantero.

Por último si se selecciona el **MODO 2x2 on**, si el sistema determina que la motocicleta circula en una situación normal, la forma de actuar sobre el motor delantero, es proporcionar una tensión un tanto por ciento proporcional al accionamiento del acelerador. En este caso, tanto la señal de entrada del acelerador como la del regulador de potencia afectarán a la señal comandada al motor delantero. Mientras que si determina que circula en situación de pérdida de tracción en la rueda trasera, la forma de actuar sobre el motor delantero, es proporcionar una tensión al motor delantero proporcional a la diferencia de flancos ascendentes entre las ruedas de la motocicleta, con la intención de igualar las velocidades de las mismas y así superar satisfactoriamente la “situación de peligro”. En este último caso sólo la señal del regulador de potencia afectará a la señal comandada al motor delantero.

Finalmente, se adjunta el diagrama de flujo en el Anexo “Diagrama de flujo”, en el que se resume el funcionamiento del control de tracción.

Señal de salida a comandar: Las señales de salida que la unidad de control y mando comandará serán por un lado, proporcionar una tensión al motor en función de la decisión tomada por esta unidad, del accionamiento del acelerador y del regulador de potencia, y por otro lado la señalización del modo de funcionamiento mediante el encendido y apagado de leds.



6. Estudio de los componentes necesarios y propuesta electrónica

Ahora necesitamos especificar los componentes necesarios para llevar a cabo nuestro sistema.

El control de tracción se va a componer de:

- Una unidad de control y mando, la cual va a ser el cerebro del sistema (esta unidad será el propio control de tracción).
- Un sensor de giro en cada rueda.
- Dos motores eléctricos brushless en los cubos de las ruedas (el proyecto está enfocado a dos motores eléctricos, pero necesariamente uno, el de la rueda delantera, tiene que ser eléctrico).
- Un acelerador, con acción directa sobre el motor eléctrico de la rueda trasera.
- Un regulador de potencia, para poder regular la tracción delantera.
- Un selector de modo, para seleccionar los distintos modos en los que puede funcionar el control de tracción.
- Un elemento, que tenga una acción directa sobre el motor eléctrico delantero (en nuestro caso, será la propia unidad de mando y control).

El esquema, del modo de actuación de nuestros elementos, es el siguiente:

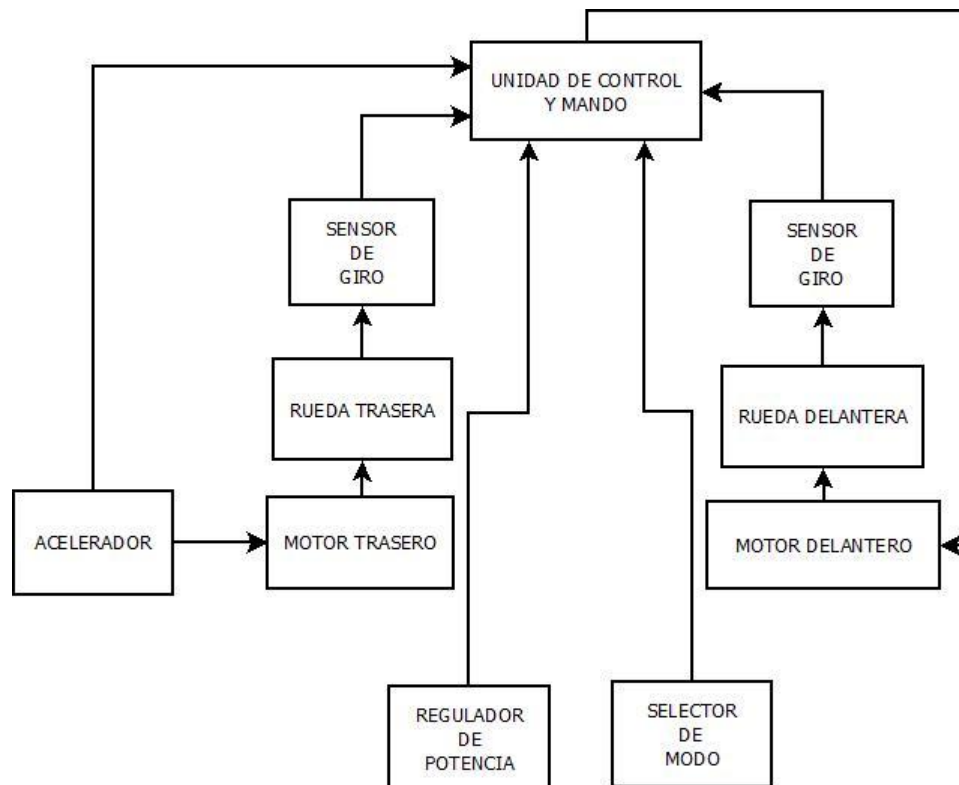


Ilustración I – Esquema de los elementos utilizados para este control de tracción

El acelerador, es un acelerador de empuñadora con el cual se gobernará el accionamiento del motor trasero. Este acelerador está compuesto por un sensor hall lineal.

La selección del modo se realizará por medio de un conmutador de 4 posiciones, con el cual seleccionar los 4 modos distintos de funcionamiento.

Como regulador de potencia de la tracción delantera, se utilizará un potenciómetro, para así adecuar la actuación de la tracción delantera a gusto de consumidor.

Y por último, como unidad de mando y control, de entre las muchas posibilidades que hay en la actualidad, se eligió utilizar un control electrónico, mediante un microcontrolador ATmega328 implementado en una plataforma electrónica como es Arduino.

El porqué más detallado de la elección de este tipo de microcontrolador, por su sencillez, coste y facilidad de implementación, junto las características del mismo se explica en el anexo “¿Por qué elegir Arduino Rev 3 como plataforma electrónica?”.

7. Construcción del prototipo

Una vez aclarada la función a realizar por el sistema y que se ha explicado cómo llevar a cabo dicha función, basándonos en la simulación con Simulink y del diagrama de flujo, solo falta llevar a la realidad un prototipo, tanto a nivel hardware como a nivel software, con el cual desarrollar las pruebas pertinentes y comprobar su funcionamiento.

7.1 Diseño Hardware

Para poder realizar este sistema, primero necesitaremos de un soporte físico.

Para la realización de la motocicleta eléctrica con tracción 2x2 se encargaron los siguientes componentes:

- Dos ruedas con sus respectivos motores brushless (sin escobillas) eléctricos situados en el cubo de las ruedas.
- Dos controladores (uno para cada motor).
- Un acelerador de empuñadura.
- Dos manetas de frenos.
- Dos sensores de efecto hall con sus correspondientes imanes.
- Cuatro leds identificativos de modo.
- Un switch selector del modo en el que nos encontremos.
- Un potenciómetro para la regulación de potencia.
- Un Arduino Uno Rev3.
- Una fuente de alimentación.
- Un interruptor start/stop.
- Una llave de seguridad para caídas.
- Diversos componentes RC

El proyecto estaba enfocado para realizar una motocicleta con motores de entre 1/4 y un Kw, pero por restricciones de presupuesto, los motores eléctricos son de 250 w. Aún así son lo suficientemente potentes para mover un vehículo de 60 Kgs brutos a más de 35 Kmh.

Para la selección de motores, se realizó una comparativa entre los distintos tipos de motores, en función de su potencia dimensiones, precio, etc.

Los controladores, se seleccionaron en función de la potencia de los motores. Finalmente se escogió uno de 250 w y otro de 500 w (por expectativas de futuros proyectos).

Los sensores elegidos son sensores de efecto hall, como ya se ha explicado anteriormente. Este tipo de sensores fueron elegidos por su facilidad de instalación y bajo coste.

Ahora que sabemos que elementos son necesarios, solo nos queda montarlos en una estructura.

7.1.1 Soporte o estructura

Para poder continuar con el desarrollo del proyecto, se decidió fabricar una estructura para colocar las ruedas en el aire y así poder realizar las pruebas pertinentes.



Imagen 7.1.1:1 - Estructura

Los detalles del diseño de dicho soporte, se adjuntan en el Anexo “Diseño de la estructura principal”.

Una vez construida la estructura, con las ruedas y los controladores montados, lo siguiente a realizar es la colocación de los sensores y de los imanes en los ejes de las ruedas.

7.1.2 Colocación de los imanes y de los sensores Hall

Lo siguiente a realizar, es la colocación del conjunto de los imanes en los ejes de las ruedas, para que giren solidariamente a estas. Para ello, la solución por la que nos decantamos, fue crear una especie de jaula para unir el imán a la tapa del motor de la rueda y así conseguir el giro solidario de ambos.

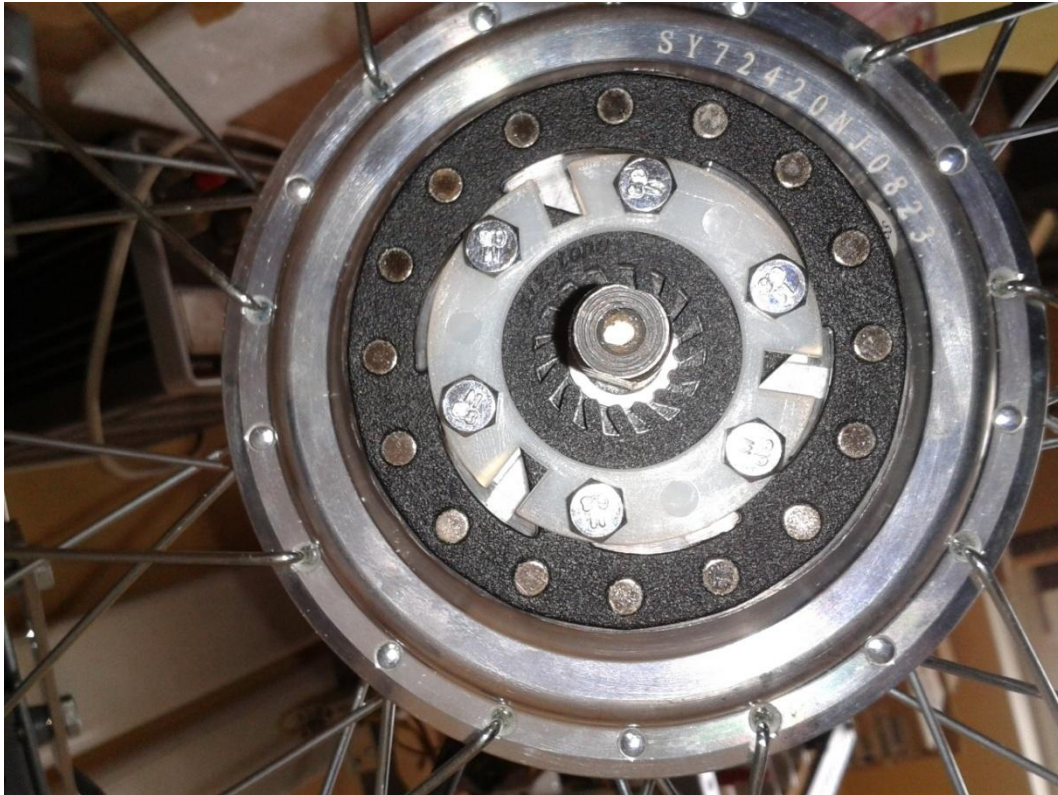


Imagen 7.1.2:1 – Colocación de los imanes en las ruedas

Los detalles, de cómo se realizó y el diseño de la “jaula”, se adjuntan en el anexo “Diseño de la colocación de los imanes en las ruedas”.

Posteriormente, una vez realizada la colocación de los imanes, cabe pensar en cómo colocar el sensor hall para la detección de los anteriores.

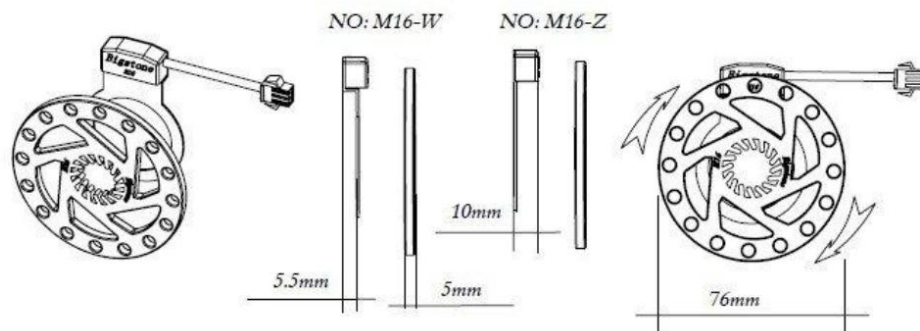


Imagen 7.1.2:2 – Sensor Hall

Como los sensores hall, deben estar enfrentados a una cierta distancia de los imanes, estos también tienen que estar situados en el eje de la rueda.

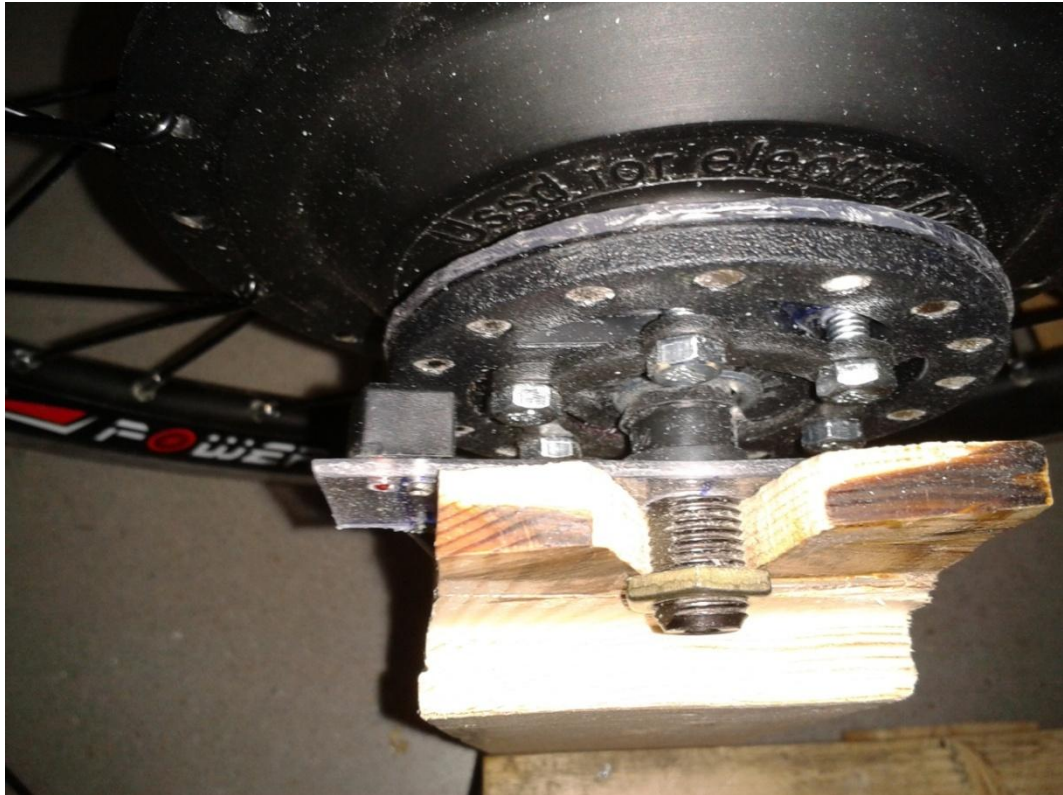


Imagen 7.1.2:3 – Colocación sensores hall

Los detalles, de cómo se realizó y el diseño de la colocación de los sensores hall en los ejes de las ruedas, se adjuntan en el anexo “Colocación de los sensores”.

7.1.3 Circuito impreso

Una vez construida la estructura, con los elementos colocados y preparados, lo siguiente a realizar es el diseño del control de tracción. Para ello, primero se realizará un prototipo mediante una placa perforada y los componentes necesarios, la cual se conectará a Arduino.

Posteriormente, se implementará ese mismo circuito electrónico en un circuito impreso, para posibles reproducciones del mismo y para una mejor presentación de cara al público. En este proyecto se realizó el diseño del mismo para la posterior materialización en futuros proyectos.

Este circuito electrónico estará compuesto de los componentes necesarios explicados anteriormente, y estará incrustado en el componente Arduino. Para ello se utilizarán los pines necesarios de este para conseguir nuestro propósito.

La selección del modo de funcionamiento se realizará mediante un conmutador de cuatro posiciones, con el que se podrá elegir, en cada

momento, uno de los cuatro modos de funcionamiento que tiene el control de tracción realizado:

ENTRADA	SALIDA
1	MODO 2X1 OFF
2	MODO 2X1 ON
3	MODO 2X2 OFF
4	MODO 2X2 ON

Tabla 3 – Conmutador de cuatro posiciones

Para ello, se utilizarán 4 entradas digitales de Arduino y además se habilitarán las resistencias pull-up, de cada una de esas entradas digitales. La habilitación de las resistencias pull-up es necesaria para que las entradas digitales, no puedan tomar cualquier valor y mantener la entrada siempre a un valor conocido.

Una entrada digital puede tomar un valor de 5v o de 0v y no se puede dejar al aire porque podría tomar un valor cualquiera e interferir en nuestro diseño electrónico.

Si una de las salidas de nuestro conmutador se conectara directamente entre tensión de 5v y el pin digital, lo que podría suceder es lo siguiente:

1. Se proporcionaría, a la entrada digital, tanta intensidad como suministre la fuente de 5V y la puede quemar
2. Cuando el interruptor está abierto, la entrada no está forzada a 0V ni a 5V, y puede tomar cualquier valor, pudiendo dar lugar a falsos estados producidos por ruido eléctrico.

Al utilizar resistencia pull-up en las entradas, la lógica se invierte, por lo que si ahora se utiliza un interruptor, con el interruptor cerrado, la señal que le llega a la entrada de Arduino es de 0 v. Y sin embargo, si el interruptor se abre, la señal que le llega a la entrada de Arduino es de 5v.

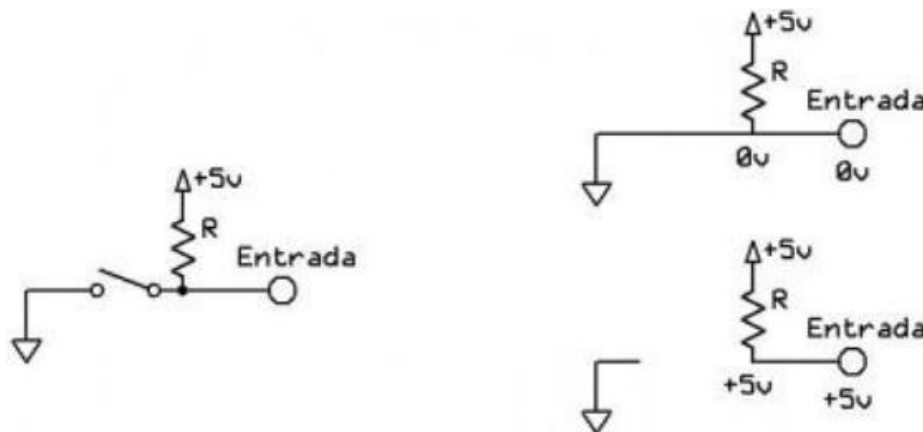


Imagen 7.1.3:1 – Funcionamiento de una resistencia Pull-up

Todo esto, se soluciona añadiendo una resistencia de pull-up de 10k, que forzará un 1 en la entrada mientras el interruptor esté abierto y que limitará la corriente cuando esté cerrado.

Las entradas digitales de Arduino, tienen una resistencia interna de pull-up de 20KΩ, que nos puede simplificar el conexionado y podemos prescindir de la resistencia externa, pero en nuestro programa habrá que tenerlo en cuenta (más adelante se especificará cómo)

Para la visualización del modo de funcionamiento en el que se encuentra el control de tracción en cada momento, se colocarán cuatro leds, que se encenderán en función del modo seleccionado mediante el switch.

Para ello, se utilizarán cuatro salidas digitales de Arduino y se colocará una resistencia, para cada salida, de 220 ohmios para limitar la corriente sobre los leds.

Además para un futuro, por si se desea incorporar una pantalla LCD, en el código se ha incluido unos mensajes, de en qué modo se encontrará el piloto mientras conduzca la motocicleta eléctrica. El modo que el piloto haya seleccionado con el conmutador se mostrará continuamente en el LCD.

Para la realización del conteo de flancos de subida, se utilizarán dos entradas de Arduino, habilitadas para realizar interrupciones externas, que en nuestro caso se producirá cuando el sensor hall detecte cada imán.

Estos sensores hall, tienen incorporado una resistencia pull-up de 3,3 Kohm entre alimentación y señal, por lo que no es necesario utilizar la resistencia pull-up interna de esos dos pines de Arduino.

La lógica de los sensores, al contener una resistencia pull-up, también se invierte. Por lo tanto, cuando el sensor hall detecta el imán (ON) la tensión que manda al Arduino es de 0v y cuando no detecta imán (OFF), se enciende un led y la tensión que manda es de 5v.

Si nuestro deseo es realizar el conteo de flancos ascendentes, entonces la interrupción se realizará cuando el sensor hall pase de detectar imán a no detectarlo.

A la hora de realizar la conexión de los sensores, con la placa perforada y el Arduino, tenemos que incluir el conexionado de condensadores de desacoplo, para dar una mayor estabilidad a la hora de alimentar estos sensores, ya que evitan que existan oscilaciones en la entrada de la alimentación de estos.

Estos condensadores de desacoplo, se colocan en paralelo lo más próximo que se pueda de la conexión de alimentación de los sensores, para mantener la inductancia lo más baja posible en las pistas que van desde el condensador de desacoplo al sensor

Las pistas y cables tienen una inductancia, que hace que cuando se requiera una intensidad importante por parte de la conmutación de las señales de los sensores, tenga un retardo en el suministro de esta intensidad y por lo tanto una caída de tensión en la alimentación. Los condensadores de desacoplo también sirven como camino de retorno a las corrientes de alta frecuencia.

El regulador de potencia, será identificado en el control de tracción por medio de un potenciómetro, con el cual se quiere controlar la tensión que se comandará al motor delantero en unas determinadas circunstancias. Con este elemento, se podrá tener por lo tanto una acción reguladora sobre la tracción delantera, que gobierna nuestro control de tracción.

Para ello, se utilizará una entrada analógica de Arduino, la cual leerá la señal que le llegue a través del potenciómetro y con ella regularemos la acción de suministro de energía al motor delantero.

Lectura del acelerador, se realizará a través de una entrada analógica, en la que se comprobará el accionamiento del acelerador. Este acelerador, está construido mediante un sensor hall lineal. El funcionamiento es muy similar al del regulador de potencia, pues por la entrada analógica de Arduino, se leerá la señal que le llegue a través del acelerador y con ella se regulará la acción de suministro de energía al motor delantero, en el caso que le corresponda.

Señal comandada al motor delantero, como ya se ha comentado anteriormente, la señal que se desea comandar es proporcionar una tensión al motor delantero en función de una serie de entradas del sistema y de una serie de decisiones tomadas por el mismo.

Una de las salidas digitales que ofrece Arduino es de tipo pwm (pseudo-analógica), es decir, modulación por ancho de pulso, y es una técnica, que consiste en transferir información o energía a un dispositivo mediante una señal cuadrada, compuesta por un valor alto de 5v y un valor bajo de 0v.

La señal de tipo PWM, se utiliza como técnica para controlar circuitos analógicos. El periodo y el ciclo de trabajo (duty cycle) del tren de pulsos, puede determinar la tensión entregada a dicho circuito. Si, por ejemplo, se tiene un voltaje de 5v y se modula con un duty cycle del 10%, se obtendrán 0.5V de señal analógica de salida.

Si dicha señal de salida, se proporcionara al motor eléctrico brushless delantero, provocaría un funcionamiento discontinuo (pulsante), debido a la característica de la señal pwm. Ello es debido a que, al comandar dicha señal al motor delantero, por mucho que la señal de salida comandada al motor delantero, tenga una tensión media idéntica a la tensión que proporcionamos con el accionamiento del acelerador, no es lo mismo, pues esta tensión es pulsatoria de 0 a 5v. Por ello, necesitamos incorporar un filtro de paso bajo, para mantener esa tensión constante a la salida y así generar o convertirla en una señal continua.

Este filtro de paso bajo, se va a realizar por medio de una resistencia y condensador, de manera que la salida que se obtenga sea aproximadamente el valor medio de dicha señal.

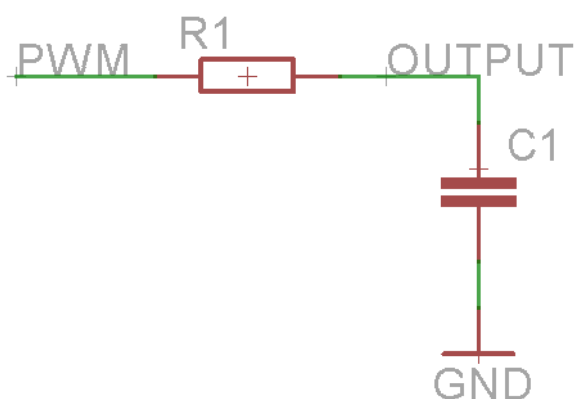


Imagen 7.1.3:2 – Circuito de un filtro paso-bajo

Dependiendo del valor del condensador y de la resistencia, se obtendrá un rizado diferente de la señal de salida.

Por ello, se realizó una serie de pruebas, por medio del osciloscopio, para comprobar tanto la tensión que se obtenía después del filtro como el rizado que se producía con él, hasta obtener así el filtrado de la señal con una tensión de salida media y con un rizado aceptable.

Condensador	Rizado en estado estacionario(mV pp)	Rizado en el momento de la aceleración (mV pp)
69 uF	500 - 550 mV	600 - 700 mV
94 uF	400 - 450 mV	400 - 450 mV
141 uF	350 - 400 mV	350 - 400 mV
210 uF	280 - 330 mV	375 - 450 mV
257 uF	220 - 280 mV	280 - 375 mV
304 uF	170 - 220 mV	250 - 300 mV
351 uF	125 - 165 mV	220 - 260 mV
398 uF	93 - 125 mV	150 - 225 mV

Tabla 4 – Elección del condensador del filtro de paso bajo

El condensador y resistencia elegidos son de 398 microfaradios y de 470 ohmios por los buenos resultados que se obtuvieron de la señal de salida.

Una vez que sabemos los componentes de los que se compondrá nuestro prototipo, llega el momento de realizarlo físicamente.

Circuito impreso del prototipo mediante fritzing.

Una vez llevado a cabo el prototipo del sistema electrónico, para realizar la placa del control de tracción y realizadas las pertinentes pruebas, para comprobar el correcto funcionamiento del mismo, se lleva a cabo el diseño del circuito impreso de la misma mediante la utilización del programa Fritzing.

Este programa, facilita el diseño de un circuito impreso, debido a que utiliza unas librerías con las cuales se pueden ir colocando los elementos virtualmente sobre la placa Arduino, que se elija, y posteriormente pasarlo directamente a un circuito impreso pcb.

Este programa, te permite realizar un autorutado de las líneas de cobre, pero en nuestro caso se realizó manualmente.

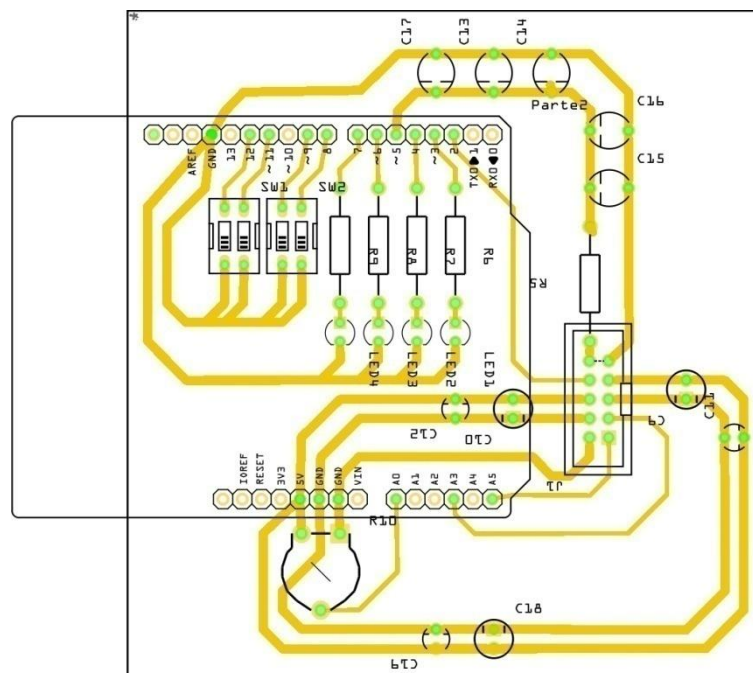


Imagen 7.1.3:3 – Circuito impreso

Arriba a la derecha, se puede observar el filtrado de la señal de salida a comandar al motor delantero.

En el centro a la izquierda, se puede observar el switch selector de modo.

Mediante el conector situado en el centro a la derecha se realizará la alimentación de los sensores hall, con la colocación de los condensadores de desacoplo.

En la parte de abajo a la izquierda, se puede observar el potenciómetro que actúa como regulador de potencia de la tracción delantera.

Y por último la señal del acelerador llegará a través del conector situado en el centro a la izquierda.

Entradas y salidas utilizadas de Arduino:

ENTRADAS DIGITALES	
PIN DIGITAL UTILIZADO	ELEMENTO CONECTADO
PIN 2	SENSOR HALL DE LA RUEDA DELANTERA
PIN 8	SELECTOR DEL MODO 3
PIN 9	SELECTOR DEL MODO 2
PIN 11	SELECTOR DEL MODO 1
PIN 12	SELECTOR DEL MODO 0

ENTRADAS ANALÓGICAS	
PIN ANALÓGICO UTILIZADO	ELEMENTO CONECTADO
PIN A0	LIMITADOR DE POTENCIA DE LA RUEDA DELANTERA
PIN A3	SENSOR HALL DE LA RUEDA TRASERA
PIN A5	ACELERADOR TRASERO

SALIDAS DIGITALES	
PIN DIGITAL UTILIZADO	ELEMENTO CONECTADO
PIN 3	LED IDENTIFICATIVO MODO 3
PIN 4	LED IDENTIFICATIVO MODO 2
PIN 5	SEÑAL DE SALIDA PWM QUE JUNTO CON EL FILTRO DE PASO BAJO SE COMANDA LA TENSIÓN AL MOTOR DELANTERO
PIN 6	LED IDENTIFICATIVO MODO 1
PIN 7	LED IDENTIFICATIVO MODO 0

Tabla 5 – Pines utilizados de Arduino

7.2 Software

Una vez realizado el diseño del hardware, tan sólo falta realizar el diseño y realización del software necesario para que dicho sistema realice lo deseado.

El código, al igual que en el desarrollo del algoritmo, estará dividido en las mismas rutinas en las que se subdividió el problema.

Partes del código:

7.2.1 Selección del modo (mediante un switch)

En la selección del modo, al ser realizada por medio de un conmutador de cuatro posiciones, será necesario la utilización de cuatro entradas digitales y en esas entradas a las que se conecte el switch, se activarán las resistencias pull-up internas. Por lo que su lógica estará invertida.

Los cuatro pines digitales, deberán declararse como entradas antes de ser utilizados en el programa:

```
const int BOTON_2X1_OFF = 12; // BOTON QUE ACCIONA EL MODO 0
const int BOTON_2X1_ON = 11; // BOTON QUE ACCIONA EL MODO 1
const int BOTON_2X2_OFF = 9; // BOTON QUE ACCIONA EL MODO 2
const int BOTON_2X2_ON = 8; // BOTON QUE ACCIONA EL MODO 3
```

```
int FLAG_0 = 0; // Bandera para solo realizar una vez ciertas acciones cuando el MODO_0
está activado o desactivado
```

Y además, se habilitará la resistencia pull-up de esos pines:

```
pinMode(BOTON_2X1_OFF, INPUT_PULLUP); // Declaramos el pin del botón como entrada y
habilitamos la resistencia pull-up interna
```

Una vez realizado esto, se procederá a la lectura de los pines de entrada y en función del accionamiento del conmutador, variará la lectura de los cuatro pines y por lo tanto la selección del modo del control de tracción:

```
int VALOR_BOTON_2X1_OFF = digitalRead(BOTON_2X1_OFF);

if (VALOR_BOTON_2X1_OFF == HIGH) // SI EL SWITCH NO ESTÁ EN LA POSICIÓN 2X1 OFF
{
  if (FLAG_0 == 1) // PARA SOLO REALIZARLO UNA VEZ
  {
    //digitalWrite(LED_0,LOW);// DE MOMENTO COMENTADO, SE APAGA EL LED QUE INDICA
QUE EL MODO_0 ESTÁ ACTIVADO
    FLAG_0 = 0;
    MODO_0 = 0; // NO ESTÁ SELECCIONADO EL MODO 0
  }
}
else // SI EL SWITCH ESTÁ EN LA POSICIÓN 2X1 OFF
{
  if (FLAG_0 == 0) // PARA SOLO REALIZARLO UNA VEZ
  {
    Serial.println("MODO 2X1 OFF\t"); // SE MUESTRA POR PANTALLA QUE ESTAMOS EN EL
MODO 2X1 OFF
    cli(); // DESHABILITAMOS LAS INTERRUPCIONES CUANDO ACCIONAMOS EL MODO_0
    //digitalWrite(LED_0,HIGH);// DE MOMENTO COMENTADO SE ENCIENDE EL LED QUE
INDICA QUE EL MODO_0 ESTÁ ACTIVADO
    MODO_0 = 1; // ESTÁ SELECCIONADO EL MODO 0
    FLAG_0 = 1;
  }
}
```

```

PRIMERA_VEZ_MODAL_0 = 1; // CADA VEZ QUE CAMBIEMOS ESTE MODO ESTA VARIABLE
SE PONDRÁ A 1
}
}

```

7.2.2 Contaje de flancos positivos (mediante interrupciones). “La señal recibida por estos pines será filtrada”.

El contaje de flancos positivos, obtenidos de las señales recibidas por los sensores, se realizará por medio de las interrupciones.

Arduino consta de dos pines digitales para poder realizar interrupciones en el programa mediante agentes externos. Estos pines son el pin 2 y el pin 3.

La forma, en la que la una interrupción se produce es seleccionable entre una de las cuatro formas disponibles:

- **LOW:** La interrupción se producirá cuando la señal que llega al pin cambie a valor bajo (low).
- **CHANGE:** La interrupción se producirá cuando la señal que llega al pin cambia de valor.
- **RISING (flanco ascendente):** La interrupción se producirá cuando la señal que llega al pin cambie de valor bajo a valor alto.
- **FALLING (flanco descendente):** La interrupción se producirá cuando la señal que llega al pin cambie de valor alto a valor bajo.

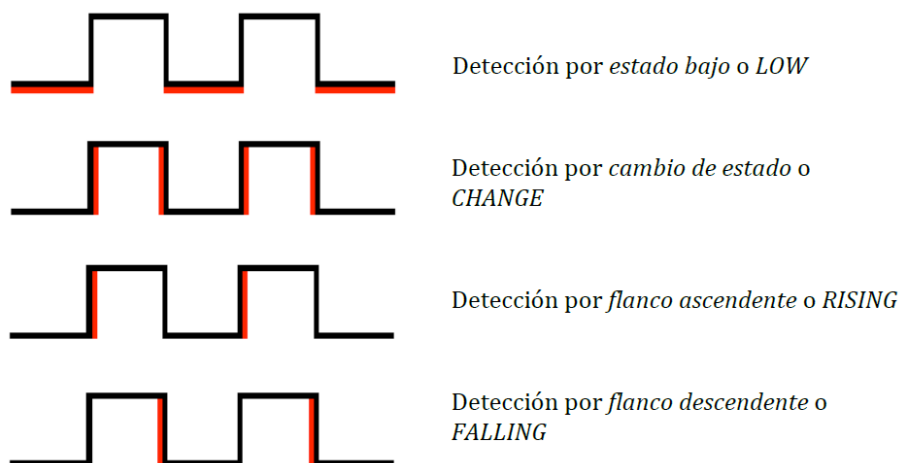


Imagen 7.2.2:1 – Formas en que se puede producir una interrupción

En este proyecto, se decidió realizar el conteo de flancos ascendentes, por lo tanto la interrupción se producirá cuando el sensor pase de detectar un imán a no detectarlo, debido al cambio de lógica provocado por la activación de las resistencias pull-up que las entradas de la placa Arduino llevan incorporadas internamente.

Al conectar la señal de los dos sensores, junto con los condensadores de desacoplo, a las entradas de Arduino indicadas, se pudo observar que al estar muy próximas se producían interferencias o ruido entre ellas. Ello podría dar lugar a un mal funcionamiento de dicho control de tracción. Para evitarlo, se decidió realizar el cambio de pin de una de las interrupciones.

El cambio de pin de la interrupción, provoca que la forma en que esta se produce no sea seleccionable, produciéndose esta cuando se produzca un cambio en la señal de entrada, es decir, tanto por flanco ascendente como por flanco descendente. El cambio de pin de una interrupción se puede realizar en uno de los tres puertos (B, C, D) que tiene Arduino, por medio del registro de control del cambio de pin de las interrupciones (PCICR).

Bit	7	6	5	4	3	2	1	0
0x68	-	-	-	-	-	PCIE2	PCIE1	PCIE0
Read/Write	R	R	R	R	R	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

Imagen 7.2.2:2 PCICR

Con la habilitación de interrupción por cambio de pin (PCIE) podemos seleccionar el puerto en el que situaremos la interrupción. Pero de momento cualquier pin de ese puerto podría causar una interrupción.

- PCIE2 para habilitar interrupciones en PCINT[23:16] (Puerto B)
- PCIE1 para habilitar interrupciones en PCINT[14:8] (Puerto C)
- PCIE0 para habilitar interrupciones en PCINT[7:0] (Puerto D)

Una vez seleccionado en que puerto se va a encontrar la interrupción, podemos elegir que pin o pines de ese puerto va a estar habilitada la interrupción mediante los registro de máscara de cambio de estado de pin (PCMSK)

- En el caso de PCIE2 se selecciona el pin mediante PCMSK2
- En el caso de PCIE1 se selecciona el pin mediante PCMSK1
- En el caso de PCIE0 se selecciona el pin mediante PCMSK0

PCMSK2 – Registro 2 de Máscara de Cambio de Estado de Pin

Bit	7	6	5	4	3	2	1	0
(0x6D)	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

PCMSK1 – Registro 1 de Máscara de Cambio de Estado de Pin

Bit	7	6	5	4	3	2	1	0
(0x6C)	-	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

PCMSK0 – Registro 0 de Máscara de Cambio de Estado de Pin

Bit	7	6	5	4	3	2	1	0	
(0x6B)	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	PCMSK0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Imagen 7.2.2:3 – PCMSK0, 1, 2

Por lo tanto, se realizó una sección de código para que el contador, de flancos ascendentes, sólo aumente su cantidad, cuando en el pin seleccionado se produzca un flanco ascendente.

Para realizar el contaje de flancos ascendentes, se aumentará una variable cada vez que se entre en la interrupción de la forma seleccionada.

Las interrupciones tendrán que habilitarse y deshabilitarse en función del modo que sea seleccionado, y serán configuradas de la siguiente manera:

```

/*CONFIGURACIÓN DE LAS INTERRUPTACIONES*/
cli(); //deshabilitamos las interrupciones globales
EIMSK |= (1<<INT0); //Declaramos el uso de la INTERRUPTACIÓN 0 EN EL PIN DIGITAL 2
EICRA |= ((1<<ISC01) |(1<<ISC00)); //Definimos que la detección de la INTERRUPTACIÓN 1
sea por flanco ascendente. También podemos usar cualquiera de los dos o "EICRA" o
"MCUCR" -> MCUCR |= ((1<<ISC01) |(1<<ISC00));
PCICR |= (1<<PCIE1); // SE HABILITA INTERRUPTACION POR CAMBIO DE PIN Y AHORA SE
UBICARÁ LA INTERRUPTACION 1 ENTRE LOS PINES PCINT14-8 CON EL CAMBIO DE PIN AHORA
LA DETECCIÓN SERÁ POR CAMBIO DE ESTADO YA SEA ASCENDENTE O DESCENDENTE Y ESTO
NO SE PUEDE CAMBIAR
PCMSK1 |= (1<<PCINT11); //SELECCIONAMOS EN PIN PCINT 11 PARA QUE SEA LA
INTERRUPTACIÓN 1, QUE QUIVALE AL PIN A3.
sei(); // habilitamos las interrupciones globales
}

```

Una vez configuradas las interrupciones, se introducen las rutinas o funciones a las que se llamará cuando se produzca la interrupción:

```

/*FUNCIONES A LAS QUE SE LLAMA CON LAS INTERRUPTACIONES*/
ISR(PCINT1_vect) // SENSOR TRASERO AL PIN A3
// Rutina de servicio de interrupción. Cada PCInt 8..14
{
    // MANEJADOR DE LA INTERRUPTACIÓN POR CAMBIO DE PIN PARA EL PIN A3, UN CAMBIO EN
EL PIN ANALÓGICO 3 GENERARÁ UNA INTERRUPTACIÓN
    if (digitalRead(HALL TRASERO) == HIGH)//SE PUEDE PONER COMO IF (PINC & (1<<PINC3)
== 1) QUE SUPUESTAMENTE REALIZARÍA LO MISMO ES SI EL PIN A3 HA CAMBIADO Y ESTÁ A
1
    {
        /* PCINT11 HA TENIDO UN FLANCO POSITIVO */
        CONTADOR TRASERO++; // AUMENTAMOS EL CONTADOR DE FLANCOS ASCENDENTES
DE LA RUEDA TRASERA
    }
}

ISR(INT0_vect) // SENSOR DELANTERO AL PIN 2

```

```
{  
  CONTADOR_DELANTERO++; // AUMENTAMOS EL CONTADOR DE FLANCOS ASCENDENTES  
  DE LA RUEDA DELANTERA  
}  
/*-----*/
```

Por último, cabe prever que para diseñar y sintonizar el PID, las lecturas de los sensores tienen que ser totalmente fiables y lo más exactas posible, excluyendo el riesgo de que pueda producirse alguna lectura falsa y así evitar errores de contaje en las entradas del microprocesador. Si esto no fuera así, podrían dar lugar a un efecto malintencionado sobre el PID y que este intente corregir constantemente diferencias de velocidades entre las ruedas que son inexistentes en la realidad.

Por ello, para evitar algún error en la lectura de los flancos por alguna causa ajena, se propone como solución el filtrado de las señales de los sensores, es decir, el filtrado de la señal de referencia (contaje de flancos ascendentes realizado por el sensor de la rueda trasera) y de la señal de retroalimentación a nuestro sistema (contaje de flancos ascendentes realizado por el sensor de la rueda delantera).

Con este filtrado, se consigue que cualquier cambio mínimo puntual en este contaje por causas ajenas, apenas tenga efecto sobre la entrada del regulador PID, debido a que la señal apenas variará y el PID no tendrá tendencia sobrecorrectiva.

Las señales que debemos filtrar para eliminar señales espúreas tienen la siguiente frecuencia máxima:

Existe un tiempo de contaje que es de 0.5 segundos, pero teniendo en cuenta que el contaje de flancos se podía reiniciar por un número de flancos máximo de 12 flancos, si se condujera a una velocidad de 25 kmh, velocidad máxima, el tiempo de muestreo podría llegar a ser de alrededor de 0.13 segundos aproximadamente, tiempo mínimo. Por lo tanto, la frecuencia y la frecuencia de corte son:

$$T = 1/0.13s = 7.69 \text{ Hz}; \omega = 2\pi \cdot 7.69 = 48.31 \text{ rad/seg}$$

Por lo tanto, situamos nuestro filtro pasa bajos alrededor de esa frecuencia, por ello escogemos el filtro $50/(50+s)$ que sitúa la frecuencia de corte en 50rad/seg.

El filtro que se utiliza es del tipo: $\frac{1}{1+Ts}$ que se convierte en $\rightarrow \frac{1}{\frac{1}{T}+s}$,

sabiendo que $\omega_c = 1/T$ entonces: $\frac{\omega_c}{\omega_c+s}$

Entonces el filtro tiene la forma: $50/(50+s)$, esta es la función de transferencia de nuestro filtro.

El retardo en nuestra señal es de 45° máximo, pero no es problemático por el tiempo de cálculo, bastante mayor que el retardo introducido en la señal de conteaje.



Ilustración II – Filtrado del conteaje de flancos de las ruedas

Esta función de transferencia se simuló mediante Simulink:

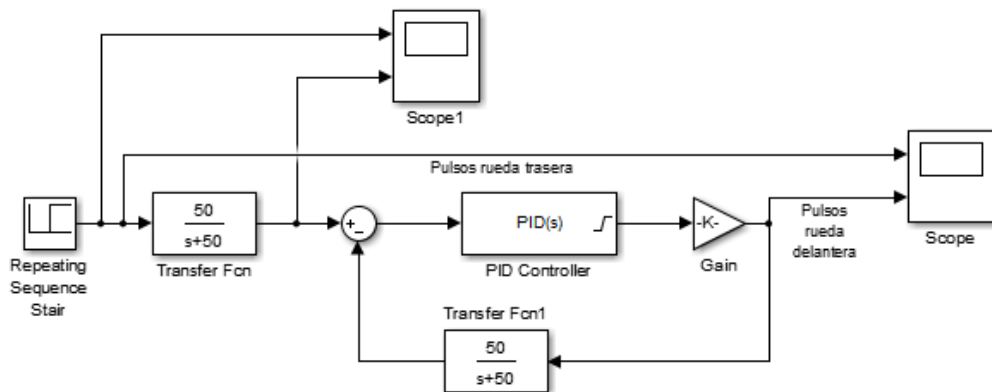


Ilustración III – Simulación del filtrado mediante Simulink

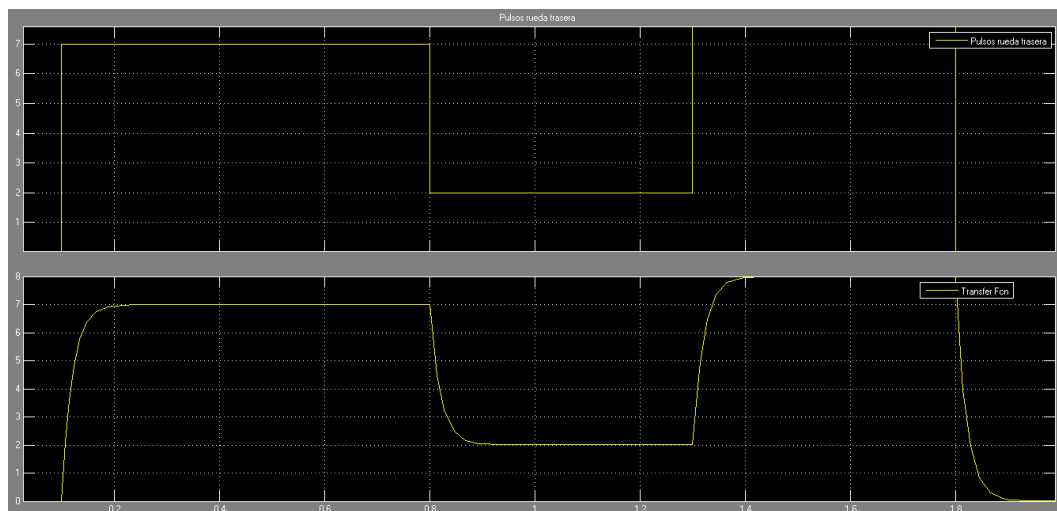


Ilustración IV – Resultados de la salida filtrada

En la anterior imagen, se muestra la forma de los pulsos de la rueda trasera, antes y después del filtro, y como se puede observar el filtro es bastante rápido.

A la hora de programar el filtro en Arduino, se tendrá que codificar la señal de cantidad de flancos ascendentes contabilizados por el sensor, tanto de la rueda trasera como de la rueda delantera, con la introducción del filtro, mediante la siguiente fórmula quedando de la siguiente manera:

$$V_T \xrightarrow{50/(50+s)} V_{T \text{ final}}$$

Por lo que:

$$V_{T \text{ final}} = V_T \cdot 50/(50+s)$$

$$V_{T \text{ final}} \cdot (50+s) = V_T \cdot 50$$

$$V_{T \text{ final}} \cdot s = V_T \cdot 50 - V_{T \text{ final}} \cdot 50$$

$$(V_{T \text{ final}} - V_{T \text{ final anterior}}) / T_m = V_T \cdot 50 - V_{T \text{ final}} \cdot 50$$

$$V_{T \text{ final}} = V_T \cdot 50 \cdot T_m - V_{T \text{ final}} \cdot 50 \cdot T_m + V_{T \text{ final anterior}}$$

$$V_{T \text{ final}} + V_{T \text{ final}} \cdot 50 \cdot T_m = V_T \cdot 50 \cdot T_m + V_{T \text{ final anterior}}$$

$$V_{T \text{ final}} (1 + 50 \cdot T_m) = V_T \cdot 50 \cdot T_m + V_{T \text{ final anterior}}$$

Velocidad de la rueda trasera:

$$V_{T \text{ final}} = (V_T \cdot 50 \cdot T_m + V_{T \text{ final anterior}}) / (1 + 50 \cdot T_m)$$

Velocidad de la rueda delantera:

$$V_{D \text{ final}} = (V_D \cdot 50 \cdot T_m + V_{D \text{ final anterior}}) / (1 + 50 \cdot T_m)$$

Estas fórmulas son la que hay que implementar en el programa de Arduino, antes de calcular la diferencia de flancos ascendentes entre las ruedas realizado por el PID, pues el Error o diferencia de velocidades entre las ruedas es $V_{\text{trasera}} - V_{\text{delantera}}$ y ahora las dos velocidades se expresan de la forma que acabamos de ver.

Ese filtro, como se ha dicho anteriormente, es bastante rápido. Si se quiere un filtro más lento, se puede tomar como filtro: $5/(5+s)$ y si se quiere un filtro aún más rápido que el que tenemos, se puede tomar como filtro: $500/(500+s)$

7.2.3 Cálculo de la diferencia de pulsos por tiempo transcurrido o por número de flancos positivos máximo.

Una vez realizado el conteo de flancos de subida de cada rueda, con su respectivo filtrado, lo siguiente a realizar es la comparación de los contadores de cada rueda y en función de ello, que el sistema tome una decisión, sobre si la motocicleta se encuentra en una situación de **pérdida de tracción en la rueda trasera**, la rueda delantera se encuentra en el aire o en situación de funcionamiento normal.

Nuestro sistema de control de tracción 2x2 va a tener dos formas de tomar las decisiones:

- 1- Superado un determinado tiempo (0.5 segundos).
- 2- Superado un determinado número de flancos de subida contabilizados de la rueda delantera.

La condición del número de flancos de subida de la rueda delantera, se toma para en casos de que circulemos a una velocidad elevada, la comparación se realice cada cierto tiempo menor que 0.5 segundos y se decide que sea de la rueda delantera porque es la que tiene menor posibilidad de sufrir pérdida de tracción en la rueda trasera.

La diferencia de flancos de subida con la que, nuestro control de tracción, determinará la situación en la que la motocicleta se encuentra funcionando debe variar en función de la velocidad. Por ello para tomar la decisión de qué diferencia de flancos de subida, entre las ruedas, tomar para cada rango de velocidades en la que nos encontremos, se realizaron los siguientes cálculos:

Cálculo del número de diferencia de flancos de subida

Para realizar el cálculo, del número de diferencia de flancos de subida en función de la velocidad a la que se circule, hay que tener en cuenta las dimensiones de nuestras ruedas. Las ruedas son de 16 pulgadas, por lo tanto:

$$1 \text{ vuelta realizada equivale } a = 2 \cdot \pi \cdot R = \pi \cdot D = \pi \cdot 0.4064 \\ = 1.27674624 \text{ metros recorridos}$$

Para una velocidad de 35 Km/h se obtendrán:

$$35 \frac{\text{Km}}{\text{h}} \cdot \frac{1000 \text{ m}}{1 \text{ Km}} \cdot \frac{1 \text{ h}}{60 \text{ min}} \cdot \frac{1 \text{ rev}}{1.27674624 \text{ metros}} = 456.8906 \text{ rpm}$$
$$456.8906 \text{ rpm} \cdot \frac{16 \text{ pulsos}}{1 \text{ rev}} \cdot \frac{1 \text{ min}}{60 \text{ seg}} = 121.8375 \text{ flancos de subida/segundo}$$

$$121.8375 \frac{\text{pulsos}}{\text{segundo}} \cdot \frac{1 \text{ segundo}}{10 \text{ décimas de segundo}} = 12.18375 \frac{\text{flancos de subida}}{\text{décimas de segundo}}$$

Cada lectura de imán en un segundo, equivale a circular a una velocidad de:

$$1 \frac{\text{pulso}}{\text{segundo}} \cdot \frac{60 \text{ segundos}}{1 \text{ min}} \cdot \frac{1 \text{ rev}}{16 \text{ pulsos}} = 3.75 \text{ rpm}$$

$$3.75 \text{ rpm} \cdot \frac{1.27674624 \text{ metros}}{1 \text{ rev}} \cdot \frac{1 \text{ Km}}{1000 \text{ m}} \cdot \frac{60 \text{ min}}{1 \text{ h}} = 0,287267904 \frac{\text{Km}}{\text{h}}$$

Si la lectura de cada imán se realizara cada 0,5 segundos, entonces equivaldría a circular a una velocidad de:

$$\frac{1 \text{ pulso}}{0.5 \text{ segundos}} \cdot \frac{60 \text{ segundos}}{1 \text{ min}} \cdot \frac{1 \text{ rev}}{16 \text{ pulsos}} = 7.5 \text{ rpm}$$

$$7.5 \text{ rpm} \cdot \frac{1.27674624 \text{ metros}}{1 \text{ rev}} \cdot \frac{1 \text{ Km}}{1000 \text{ m}} \cdot \frac{60 \text{ min}}{1 \text{ h}} = 0,57535808 \frac{\text{Km}}{\text{h}}$$

Tabla de toma de decisiones:

Velocidad	13.1 rpm	32.6 rpm	65.3 rpm	90 rpm	130.5 rpm	195.8 rpm	261.1 rpm	326.35 rpm
Velocidad	1 Km/h	2.5 Km/h	5 Km/h	7 Km/h	10 Km/h	15 Km/h	20 Km/h	25 Km/h
Flancos ↑seg	4 flancos/seg	9 flancos/seg	18 flancos/seg	24 flancos/seg	35 flancos/seg	52 flancos/seg	70 flancos/seg	87 flancos/seg
Diferencia de Flancos ↑	2 flancos	4 flancos	6 flancos	8 flancos	9 flancos	10 flancos	10 flancos	9 flancos
Flancos ↑ /0.1seg	0.4 flancos/seg	0.9 flancos/seg	1.8 flancos/seg	2.4 flancos/seg	3.5 flancos/seg	5.2 flancos/seg	7 flancos/seg	8.7 flancos/seg
Diferencia de flancos ↑ en 0.1seg	0.2 flancos	0.4 flancos	0.6 flancos	0.8 flancos	0.9 flancos	1 flancos	1 flancos	0.9 flancos
Flancos ↑ /0.5seg	2 flancos/seg	4.5 flancos/seg	9 flancos/seg	12 flancos/seg	17.5 flancos/seg	26 flancos/seg	35 flancos/seg	43.5 flancos/seg
Diferencia de flancos ↑ en 0.5seg	1 flanco	2 flancos	3 flancos	4 flancos	4.5 flancos	5 flancos	5 flancos	4.5 flancos
Diferencia en % de flancos	50%	44.44 %	33.33 %	33.33 %	25.71 %	19.23 %	14.29 %	10.34 %

Tabla 6 – Toma de decisiones

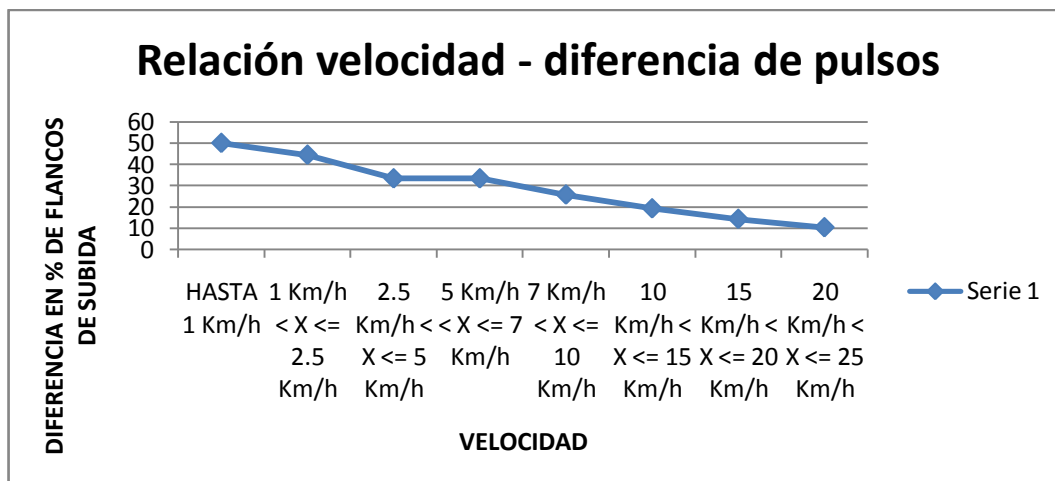


Ilustración V – Relación de la velocidad con la diferencia de pulsos

Como se puede observar, se ha tomado la decisión de que la diferencia en % de flancos de subida en relación con el número de Flancos/segundo totales, leídos por los sensores, a velocidades bajas sea mayor que a velocidades elevadas. Ello es debido a que, cuando circulamos a velocidades bajas el número de Flancos/segundo es reducido, por lo que si la diferencia de flancos de subida con la que tomamos la decisión de estar en una situación de funcionamiento o en otra, es muy pequeña, podría darse el caso de que por algún error de lectura en los sensores o por una causa ajena a la de diferencias de velocidades, haya una diferencia mínima entre las lecturas de los sensores y de lugar a una decisión de situación de funcionamiento de nuestro vehículo incorrecta, que no coincide con la situación real. Por ello a velocidades bajas el porcentaje de diferencia de flancos de subida con respecto a los Flancos/segundo leídos es mayor que a velocidades elevadas, ya que al tener un mayor número de Flancos/segundo contabilizados se desea que si se produce pérdida de tracción en la rueda se pueda actuar con la mayor rapidez, pero también manteniendo un margen de error que permita distinguir entre pérdida de tracción o falsas lecturas de los sensores.

Se decidió que el número de flancos ascendentes límite a partir de la cual se realice la comparación de velocidades sea 12 flancos. Por lo tanto la tabla de toma de decisiones queda de la siguiente manera:

Velocidad	13.1 rpm	32.6 rpm	65.3 rpm	90 rpm	130.5 rpm	195.8 rpm	261.1 rpm	326.35 rpm
Velocidad	1 Km/h	2.5 Km/h	5 Km/h	7 Km/h	10 Km/h	15 Km/h	20 Km/h	25 Km/h
Flancos ↑ /0.5seg	2 flancos/seg	4.5 flancos/seg	9 flancos/seg	12 flancos/seg	12 flancos/seg	12 flancos/seg	12 flancos/seg	12 flancos/seg
Diferencia de flancos ↑ en 0.5seg	1 flanco	2 flancos	3 flancos	4 flancos	3 flancos	3 flancos	2 flancos	2 flancos
Diferencia en % de flancos	50%	44.44 %	33.33 %	33.33 %	25%	25%	16.66 %	16.66 %

Tabla 7 – Toma de decisiones final

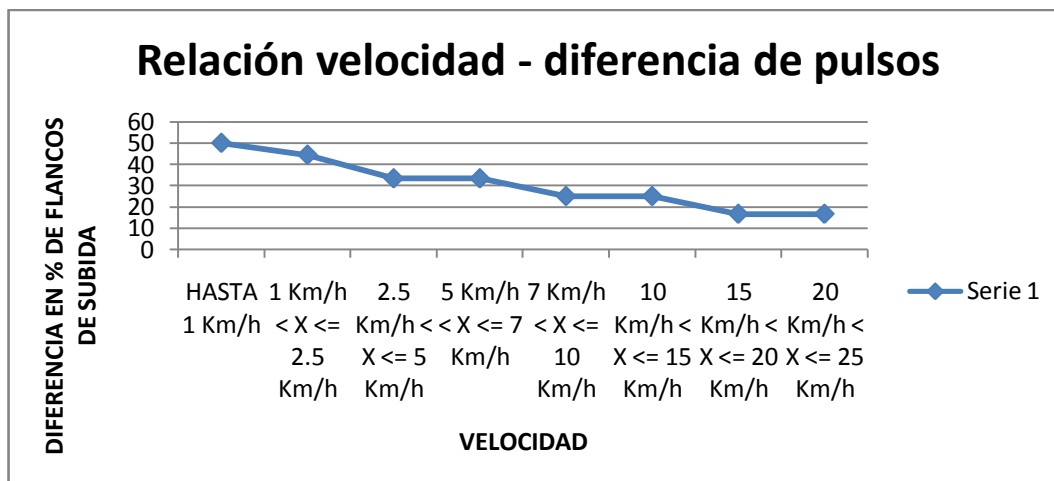


Ilustración VI – Relación de la velocidad con la diferencia de pulsos

7.2.4 Toma de decisiones.

La primera decisión del sistema, se ha tomado con la determinación de la situación en la que se encuentra la motocicleta en ese instante, mediante la comparación de flancos ascendentes contabilizados por los sensores.

La segunda decisión que tiene que tomar el sistema es, en función del modo seleccionado y de la decisión tomada anteriormente, de en qué situación se encuentra circulando la motocicleta, determinar la señal comandada al motor eléctrico delantero.

7.2.5 Señal de mando al motor delantero mediante un PID.

Una de las partes más importantes, a la hora de realizar este control de tracción, es cómo proporcionar la señal adecuada al motor delantero, cuando el sistema determine la acción más efectiva con el modo de funcionamiento seleccionado, la situación que este mismo sistema haya determinado en la que se encuentra funcionando la motocicleta y en algunas ocasiones en función del regulador de potencia de la tracción delantera y del acelerador.

Si se decide que estamos en situación de pérdida de tracción en la rueda trasera y tenemos que accionar el motor delantero hasta igualar velocidades, hay que comandar una tensión al motor delantero proporcional a la diferencia de flancos ascendentes entre las ruedas.

Para ello, primero físicamente se tuvo que modificar la señal de salida del Arduino, porque era una señal de tipo pwm y se convirtió mediante un filtro paso bajo en una señal de continua y así poder proporcionar la señal deseada al motor delantero, como ya se explicó anteriormente.

Para conseguir que la rueda delantera siga a la rueda trasera a la misma velocidad, se solucionará mediante un controlador PID.

El sistema de control, que queremos realizar es un sistema de control en lazo cerrado, por lo que nuestro control será retroalimentado. Este control tiene como objetivo reducir la diferencia entre la salida de un sistema y alguna entrada de referencia, que en presencia de ciertas perturbaciones, puede haber variado.

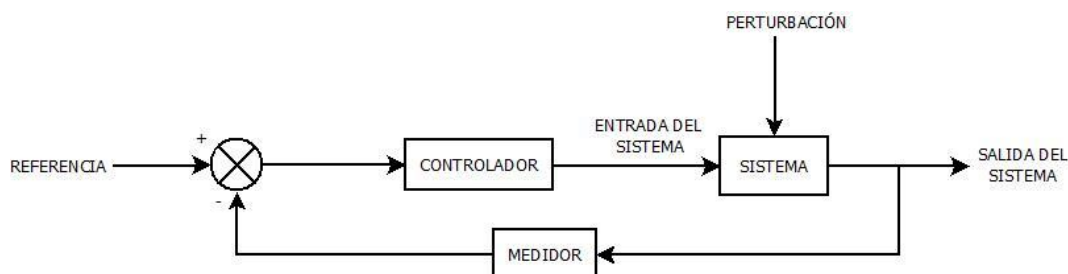


Ilustración VII – Sistema de control en lazo cerrado

En nuestro caso, la referencia es la velocidad de la rueda trasera, la perturbación o perturbaciones, son las condiciones del terreno, las condiciones ambientales y cualquier tipo de incidencia que pueda provocar pérdida de tracción en la rueda trasera. La entrada del sistema, es la tensión que se manda al controlador delantero para igualar la velocidad de la rueda delantera (que es la salida del sistema) a la de la rueda trasera, para evitar cualquier situación de peligrosidad en la moto. Y la señal de retroalimentación es la velocidad real de la rueda delantera.

Para conseguir la reducción de la diferencia entre la salida y la entrada, también llamado error, se manipulará alguna variable de entrada del sistema, siendo la magnitud de dicha variable de entrada, función de la diferencia entre la variable de referencia y salida del sistema.

Por lo tanto se tienen los siguientes elementos en este lazo de control:

- Sistema a controlar: Que las velocidades de las ruedas de nuestro vehículo sean iguales, es decir, que el número de flancos ascendentes contabilizado por cada sensor de cada rueda sea igual o este dentro de un determinado margen de error).
- Controlador: Mediante Arduino comparando las velocidades de las ruedas y tomando las decisiones oportunas, como es la variación de tensión de la rueda delantera en función de las entradas del sistema y de las decisiones tomadas por el mismo.
- Actuador: Mediante el pin digital, con señal del tipo pwm, del Arduino y el filtro de paso bajo, se enviará la tensión determinada a la rueda delantera, junto con la ayuda de la programación de un controlador PID para cuando el sistema decida comandar una tensión al motor delantero proporcional a la diferencia de velocidades entre las ruedas.

También tendrán actuación sobre la señal comandada, el acelerador, el regulador de potencia y el selector de modo.

- Medidor: Mediante el Arduino, los sensores hall e imanes de cada rueda, para calcular la velocidad de cada rueda o contabilizar los flancos ascendentes en un determinado tiempo.
- Referencia: Velocidad de la rueda trasera o el número de flancos de subida contabilizados por el sensor trasero en un tiempo.
- Salida del sistema 1 y 2: Son el movimiento de las ruedas.
- Entrada del sistema: Es la tensión que se le proporciona al controlador, de la rueda delantera, a través del pin de Arduino, con señal del tipo pwm, que al ser una señal de tensión pulsatoria la convertimos mediante un filtro paso bajo en una señal de tensión continua adaptada totalmente para que el controlador delantero la pueda leer correctamente.
- La salida 2 se retroalimenta, de la rueda delantera saldría la retroalimentación al medidor (sensor hall delantero) que es el que calcularía la velocidad de la rueda delantera o contabilizaría el número de flancos de subida en un tiempo en el nuevo intervalo de ejecución.
- La salida 1 se retroalimenta, de la rueda trasera saldría la retroalimentación al medidor (sensor hall trasero) que es el que calcularía la velocidad de la rueda trasera o contabilizaría el número de flancos de subida en un tiempo en el nuevo intervalo de ejecución que es la nueva referencia.

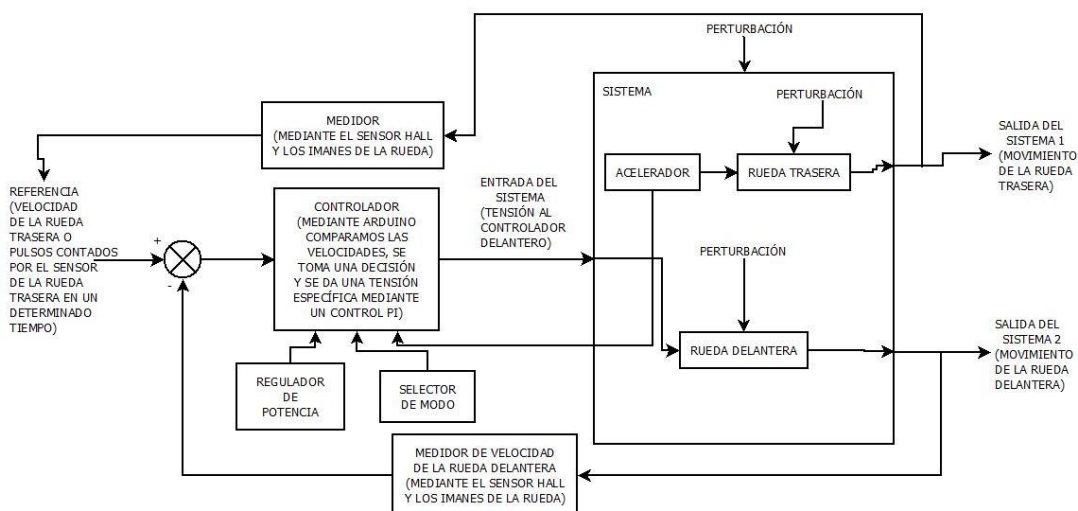


Ilustración VIII – Esquema de nuestro sistema de control

Elección del controlador

La primera decisión en el diseño de un sistema de control PID es la elección del controlador. Posteriormente se ajustarán los parámetros del mismo. Para

una buena elección del tipo de controlador a emplear (P, PI, PD o PID) ayudarán las siguientes consideraciones:

Controlador P:

En ciertos tipos de procesos es posible trabajar con una ganancia elevada sin tener ningún problema de estabilidad en el controlador. Muchos procesos que poseen una constante de tiempo dominante o son integradores puros caen en esta categoría. Una alta ganancia en un controlador P significa que el error en estado estacionario será pequeño y no se necesitará incluir acción integral.

Controlador PD:

En líneas generales, el control PD puede ser apropiado cuando el proceso a controlar incorpore ya un integrador. También son válidos procesos en que es posible trabajar con ganancias elevadas en el controlador sin que sea necesario introducir la acción integral. La acción derivativa es sensible al ruido, ya que a altas frecuencias tiene una ganancia relativamente elevada. Por lo tanto, en presencia de altos niveles de ruido se debe limitar dicha ganancia, o prescindir de la acción derivativa.

Controlador PI:

Es la estructura más usual del controlador. La introducción de la acción integral es la forma más simple de eliminar el error en régimen permanente. Otro caso en el que es común utilizar la estructura PI es cuando el desfase que introduce el proceso es moderado. La acción derivativa, más que una mejora en esta situación es un problema ya que amplifica el ruido existente. También se recomienda la acción PI cuando hay retardos en el proceso, ya que, es este tipo de procesos la acción derivativa no resulta apropiada en este tipo de sistemas. Un tercer caso en el que se debería prescindir de la acción derivativa es cuando el proceso está contaminado con niveles de ruido elevados.

Controlador PID:

La acción derivativa suele mejorar el comportamiento del controlador, ya que permite aumentar las acciones proporcional e integral. Se emplea para mejorar el comportamiento de procesos que no poseen grandes retardos pero que sí presentan grandes desfases. Este es el caso típico de procesos con múltiples constantes de tiempo.

Una vez tenemos una idea sobre los tipos de controladores, como nuestra intención es eliminar el error entre las velocidades de las ruedas utilizaremos un control proporcional, integral y derivativo. Con este control PID relacionaremos la tensión que se comandará al controlador del motor delantero con la diferencia de flancos ascendentes entre las ruedas.

La ecuación de un controlador PID es la siguiente:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt}$$

Ecuación 1 – Ecuación de un controlador PID

Que para entenderlo de forma más clara el diagrama de bloques del controlador PID queda de la siguiente manera:

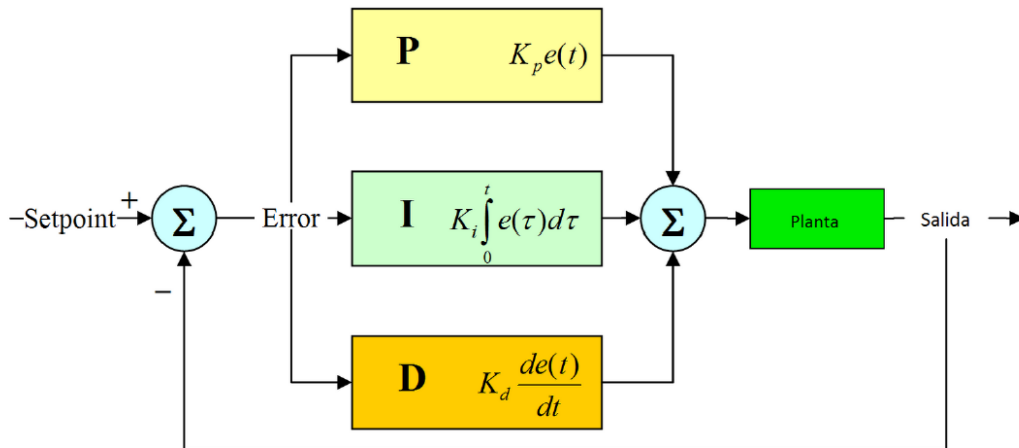


Ilustración IX – Diagrama de bloques de un PID

De la ecuación, se pueden hacer las siguientes afirmaciones:

- $e(t)$ es el error de la señal.
- $u(t)$ salida del controlador y entrada de control al proceso.
- K_p es la ganancia proporcional.
- T_i es la constante de tiempo integral.
- T_d es la constante de tiempo derivativa.

Del diagrama de flujo se puede decir lo siguiente:

- El primer bloque de control (*proporcional*) consiste en el producto entre la señal de error y la constante proporcional, quedando un error en estado estacionario casi nulo.
- El segundo bloque de control (*integral*) tiene el propósito de disminuir y eliminar el error en estado estacionario, provocado por el modo proporcional. El control integral actúa cuando hay una desviación entre la variable y el punto de consigna o referencia, integrando esta desviación en el tiempo y sumándola a la acción proporcional.
- El tercer bloque de control (*derivativo*) considera la tendencia del error y permite una repercusión rápida de la variable después de presentarse una perturbación en el proceso. El control derivativo mide la pendiente instantánea de $e(t)$, prediciendo que tan grande será el

sobreimpulso aplicando las correcciones apropiadas antes de que se presente ese sobreimpulso (tratamiento predictivo).

Parte integral:

El integrador se puede definir de la forma: $\frac{Ki}{s}$

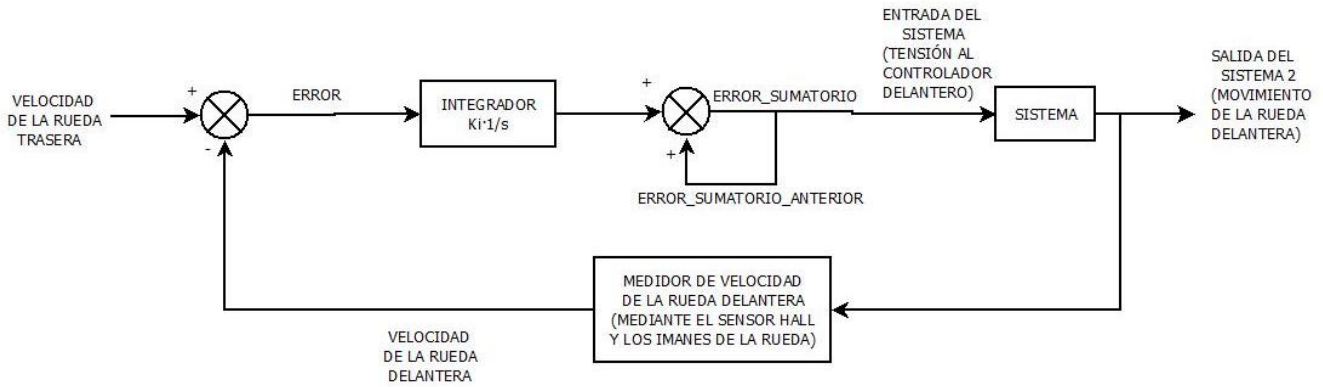


Ilustración X – Parte integral

Por lo tanto, si el error es:

$$Error (\varepsilon) =$$

$$Velocidad\ de\ la\ rueda\ trasera - Velocidad\ de\ la\ rueda\ delantera(*)$$

Entonces el **término integral** queda de la siguiente forma:

$$Error_sumatorio = \varepsilon \cdot \frac{Ki}{s}$$

$$Error_sumatorio \cdot s = \varepsilon \cdot Ki$$

$$\frac{Error_sumatorio - Error_sumatorio_anterior}{tm} = \varepsilon \cdot Ki$$

$$Error_sumatorio = Error_sumatorio_anterior + \varepsilon \cdot Ki \cdot tm$$

(*)Las velocidades de las ruedas serán flancos ascendentes contabilizados en un determinado tiempo

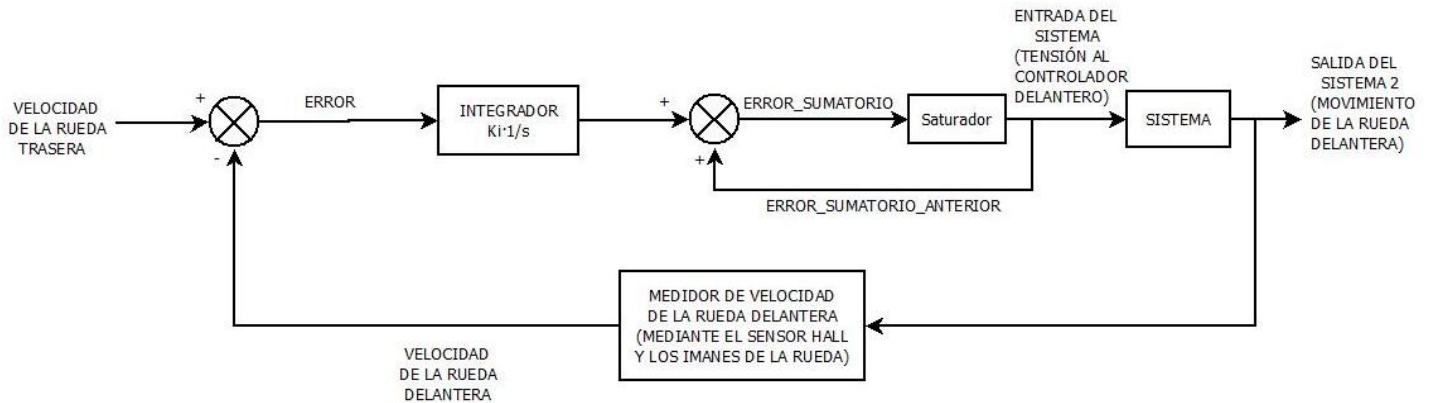


Ilustración XI – Parte integral con saturación

Después del integrador se coloca un saturador para que este no tome valores desorbitados. Como valores de saturación del integrador se tomaron los valores de salida que son 0.54 y 3.8. Como límite superior se eligió el máximo de salida para no proporcionar más de la señal que puede recibir el motor eléctrico delantero y como límite inferior, el valor mínimo de salida para evitar retardos en la señal, efecto anti wind-up.

Parte proporcional:

Al integrador le sumamos el proporcional que se puede definir como: K_p

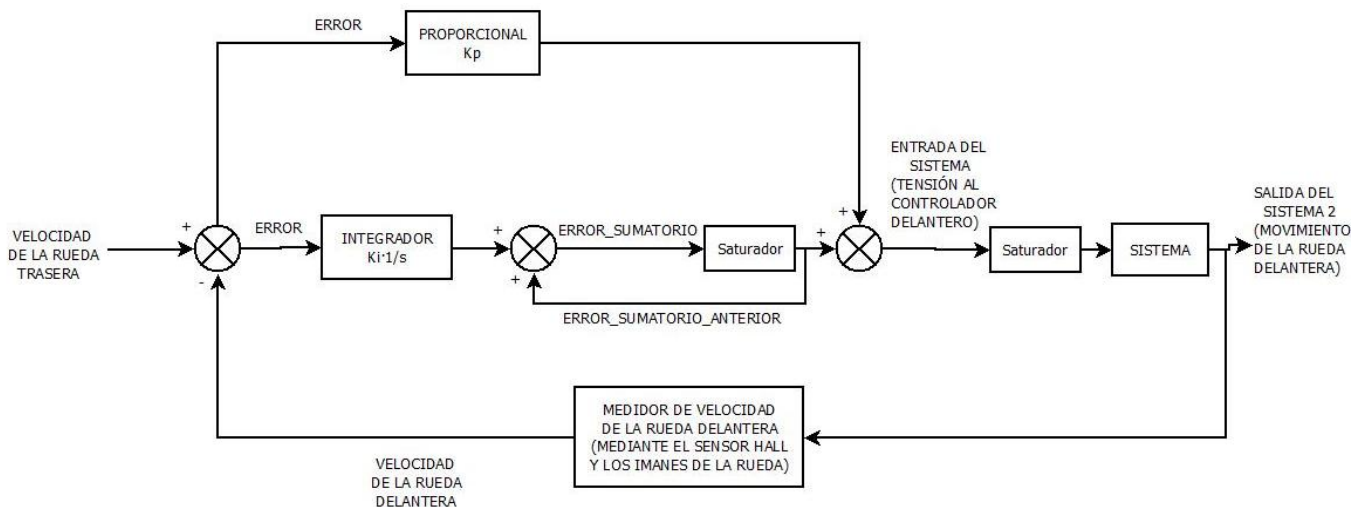


Ilustración XII – Parte integral y parte proporcional

El *término proporcional* se expresa de la siguiente forma:

$$Error_sumatorio + Error_prop = Error_sumatorio + \varepsilon \cdot K_p$$

$$Error_prop = \varepsilon \cdot K_p$$

Parte derivativa:

Finalmente se añadió el término derivativo para mejorar el control a realizar sobre la tensión que se proporciona al controlador delantero para igualar la velocidad de la rueda delantera a la de la rueda trasera.

El **término derivativo** se representa de la siguiente forma:

$$Error_Der = \varepsilon * kd * s$$

$Error (\varepsilon) =$

$Velocidad\ de\ la\ rueda\ trasera - Velocidad\ de\ la\ rueda\ delantera(*)$

(*)Las velocidades de las ruedas serán flancos ascendentes contabilizados en un determinado tiempo

$$Error_Der = ((Error(\varepsilon) - Error_anterior(\varepsilon_ant)) * kd)/tm$$

Tanto este error como el proporcional también tendrán unos límites los cuales se tomarán añadiendo un saturador. El saturador proporcional tomará unos límites de 3.8 y -3.8 y el saturador derivativo de 2 y -2.

El diagrama de bloques del PID realizado queda por lo tanto de la siguiente manera:

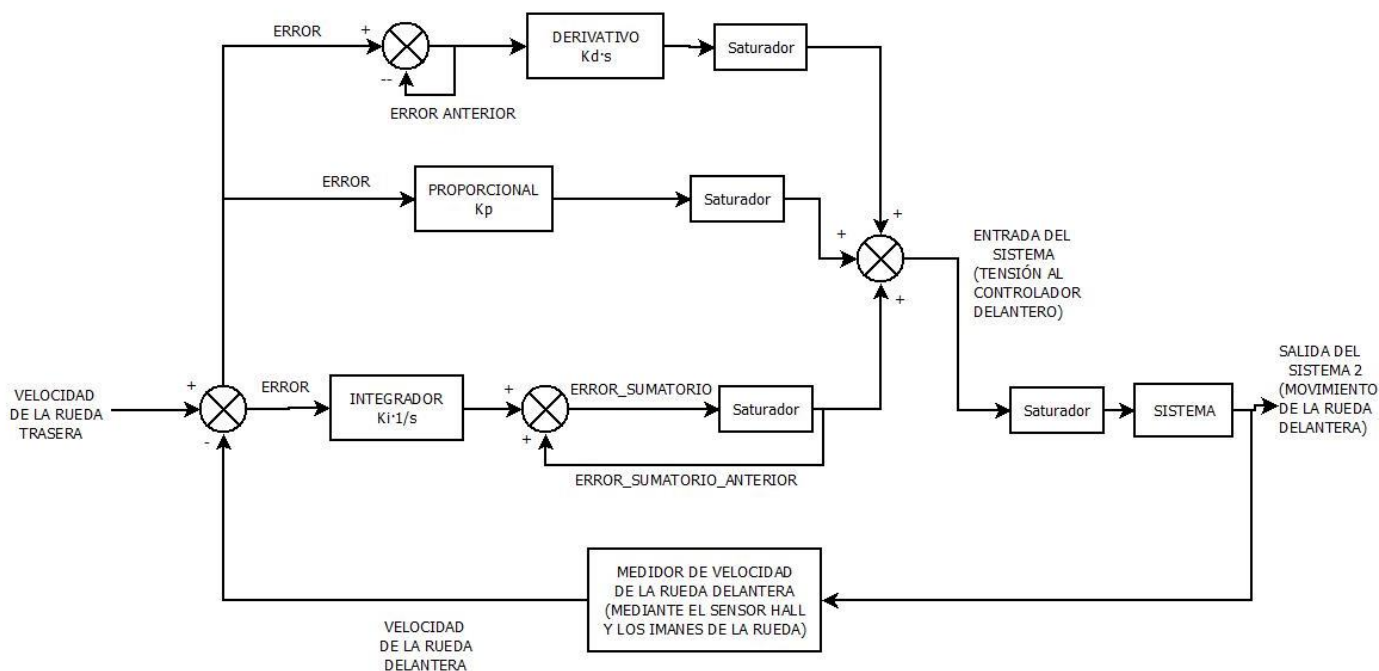


Ilustración XIII – Parte integral proporcional y derivativa

EL valor final después del PID también se encuentra limitado por un saturador para que el valor final tampoco tome unos valores desorbitados. Los valores máximos y mínimos de este saturador coincide con los valores mínimos y máximos que se proporcionará al motor delantero, 0.54 y 3.8.

Como valor mínimo que se proporciona al motor delantero es 0.54 Voltios, porque esa tensión es la que evitará que los imanes ejerzan una oposición al giro de la rueda. Y como valor máximo se proporcionará 3.8 Voltios, pues era la tensión máxima que se le podía proporcionar.

Por último se añade el filtro tanto en la referencia como en la señal retroalimentada como se explicó anteriormente:

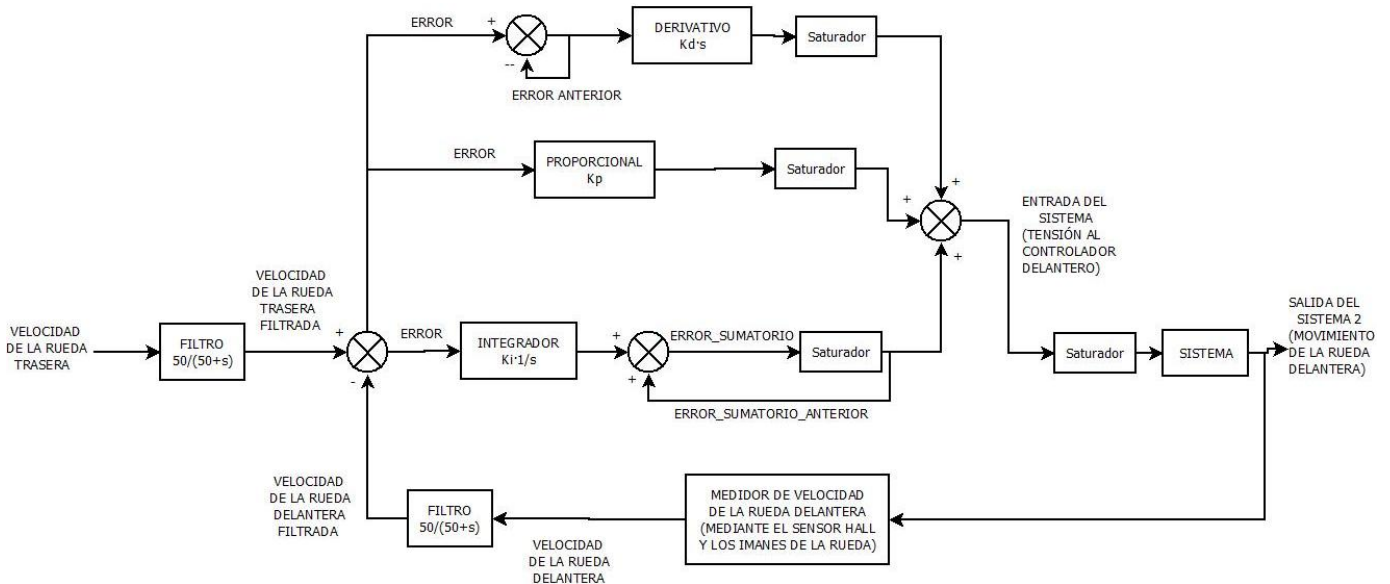


Ilustración XIV – PID con filtro

Por lo tanto la salida, que es la tensión de entrada al motor delantero, equivale a la suma del error proporcional más el error integral más el error derivativo

Por ello la ecuación del PID es la suma de los errores:

Salida = error proporcional + error sumatorio + error derivativo

En el programa de Arduino aparece de la siguiente forma:

```

/* REALIZACIÓN DEL PID*/
void PID()
{
  nuevo_Automatico = FuncionamientoAutomatico;
  Ahora_PID = micros(); // Tiempo actual en el que se produce el PID
  dT = (Ahora_PID - tiempo_anterior_PID); // ASÍ SE GUARDA EN UNA VARIABLE EL TIEMPO DESDE QUE SE PRODUJO EL PID ANTERIOR Y EL NUEVO
  Error = (velocidad_rueda_trasera_final - velocidad_rueda_delantera_final); // PARA CALCULAR EL ERROR O DIFERENCIA DE FLANCOS ASCENDENTES CONTABILIZADOS EN UN TIEMPO DETERMINADO ENTRE LAS RUEDAS
  if(nuevo_Automatico && !Automatico) // SI SE PROVIENE DE UN MODO DE DAR TENSIÓN AL CONTROLADOR DELANTERO QUE NO SEA MEDIANTE EL PID
  {

```

```

    Inicializar(); // Para inicializar el PID cada vez que pasemos de un modo de
funcionamiento en el que no se utilice el PID
    Serial.println("Inicializamos PID");
}
errorProp = kp*Error; // SE CALCULA LA PARTE PROPORCIONAL DEL PID
// se limita el error proporcional
if (errorProp > 3.8)
{
    errorProp = 3.8; // Límite superior
}else if (errorProp < -3.8)
{
    errorProp = -3.8; // Límite inferior
}
errorsum += (Error*dT*kj/1000000); // es igual que errorsum = errorsum +
Error*dT*kj/1000000000 la división entre 1000000000 es para ponerlo en función de
microsegundos // SE CALCULA LA PARTE INTEGRAL DEL PID
//hay que limitar el error integral que es el errorsum
if (errorsum > 3.8)
{
    errorsum = 3.8; // Límite superior
}else if (errorsum < 0.54)
{
    errorsum = 0.54; // Límite inferior
}
errorDer = ((Error-Error_anterior)*kd)/dT; // no se multiplica por 1000000 porque kd lo va
a absorber // SE CALCULA LA PARTE DERIVATIVA DEL PID
// se limita el error derivativo
if (errorDer > 2)
{
    errorDer = 2; // Límite superior
}else if (errorDer < -2)
{
    errorDer = -2; // Límite inferior
}
salida = ((errorsum)+ errorProp + errorDer); // LA SALIDA DEL PID TOTAL ES LA SUMA DE
TODAS LOS TÉRMINOS, EL PROPORCIONAL, EL INTEGRAL Y EL DERIVATIVO
//hay que limitar la salida total
if (salida > 3.8)
{
    salida = 3.8; // Límite superior
}else if (salida < 0.54)
{
    salida = 0.54; // Límite inferior
}else
{
    //No se hace nada pues se queda como está
}
velocidad_RD_float = ((salida * (31.4)) + 80); // SE TRADUCE ESA SEÑAL DE TENSIÓN A
VALORES ENTRE 0 Y 255 (0 Y 5 VOLTIOS) Y SE REALIZA DE ESTA FORMA PARA QUE AMBOS
MOTORES EMPIECEN A FUNCIONAR A LA VEZ
velocidad_RD_int = velocidad_RD_float;

```

```
if(velocidad_RD_float - velocidad_RD_int > 0.5)
{
  velocidad_RD_int += 1;
}
velocidad_RD_int = constrain(velocidad_RD_int,0,175); // Se limita la señal a comandar al
motor delantero
Mandar_al_controlador();
//-----
Error_anterior = Error; // Se guarda la diferencia de flancos ascendentes anterior
Automatico = nuevo_Automatico;
tiempo_anterior_PID = Ahora_PID; // Se guarda el tiempo en el que se realizó el último PID
}
/* FIN REALIZACIÓN DEL PID*/
```

7.2.6 Regulador de potencia.

La limitación de la potencia del motor de la rueda delantera se realizará, como se ha dicho, por medio de un potenciómetro. Para ello leeremos la entrada analógica a la que se conectará el potenciómetro.

El potenciómetro es un elemento mecánico que dependiendo de la posición de giro de su eje proporciona un valor variable de resistencia. Este elemento en realidad es un divisor de tensión formado por dos resistencias conectadas en serie donde los pines extremos del potenciómetro corresponden al extremo de cada resistencia y el punto común de ellas al pin central del potenciómetro.



Imagen 7.2.6:1 – Regulador de potencia

Girando el eje del potenciómetro se varía el valor de ambas resistencias y por lo tanto se variará la tensión que se proporcionará a la salida.

Arduino tiene un convertidor analógico-digital (ADC) que lee una tensión de 0 a 5 Voltios y devuelve un número en una escala de 0 a 1023.

Mediante el comando analogRead(pin analógico al que se conecta la salida del potenciómetro) Arduino devolverá el valor del ADC que es aproximadamente proporcional al valor del voltaje aplicado al pin de entrada analógica.

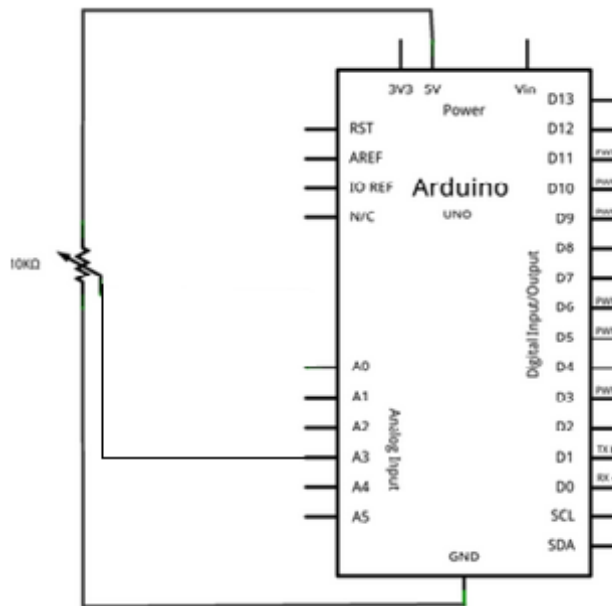


Imagen 7.2.6:1 – Conexión del regulador de potencia al Arduino

Nosotros, leeremos la entrada analógica a la que se conecta la salida del potenciómetro y guardaremos en una variable un valor en función del valor de lectura del pin. Posteriormente, la señal de salida comandada al motor delantero será multiplicada por ese valor para así poder limitar el valor de la misma.

El valor de la variable variará de 0.1 a 1 en función de que la lectura del pin analógico sea 0 o 1023.

A continuación se muestra la sección de código del programa que equivale al limitador de potencia:

```

/* FUNCIÓN REGULACIÓN DE POTENCIA */ // SIN TESTEAR
/*void FUNCION_REGULACION_DE_POTENCIA()
{
  valor_regulador = analogRead(Regulador_de_potencia); // Guardamos el valor del acelerado
  r
  if (valor_regulador <= 100)
  {
    valor_nuevo = 0.1;
  }
  else if (100 < valor_regulador && valor_regulador <= 200)
  {
    valor_nuevo = 0.2;
  }
  else if (200 < valor_regulador && valor_regulador <= 300)
  {
    valor_nuevo = 0.3;
  }
  else if (300 < valor_regulador && valor_regulador <= 400)
  {

```



```
        valor_nuevo = 0.4;
    }
    else if (400 < valor_regulador && valor_regulador <= 500)
    {
        valor_nuevo = 0.5;
    }
    else if (500 < valor_regulador && valor_regulador <= 600)
    {
        valor_nuevo = 0.6;
    }
    else if (600 < valor_regulador && valor_regulador <= 700)
    {
        valor_nuevo = 0.7;
    }
    else if (700 < valor_regulador && valor_regulador <= 800)
    {
        valor_nuevo = 0.8;
    }
    else if (800 < valor_regulador && valor_regulador <= 900)
    {
        valor_nuevo = 9;
    }
    else// (900 < valor_regulador && valor_regulador <= 1023)
    {
        valor_nuevo = 1.0;
    }
    //Serial.println(valor_regulador);
    //Serial.println(valor_nuevo);
}
*/
/* FIN FUNCIÓN REGULACIÓN DE POTENCIA */

/*----FUNCIÓN QUE MANDA LA SEÑAL AL CONTROLADOR----*/
void Mandar_al_controlador()
{
    analogWrite(controlador_delantero,(velocidad_RD_int*valor_nuevo));
}
/*----FIN FUNCIÓN QUE MANDA LA SEÑAL AL CONTROLADOR----*/
```

7.2.7 Lectura del sensor hall lineal del acelerador.

El acelerador que utilizamos tiene en su interior un sensor hall lineal, para la lectura del accionamiento del acelerador utilizaremos como en el caso anterior un pin analógico y mediante la lectura de ese pin podremos saber la posición del acelerador. Con la lectura del accionamiento del acelerador suministraremos una tensión al motor delantero de nuestro vehículo.

Como en el regulador de potencia leeremos el pin analógico al que se conectará el acelerador mediante AnalogRead (pin analógico al que se conecta la salida del acelerador) y guardaremos en una variable un valor en función del valor de lectura del pin.

Posteriormente, mapearemos esa variable para conseguir que, con el mismo accionamiento del acelerador, las ruedas de nuestro vehículo comiencen a girar al mismo tiempo. Esto se consigue con el comando: `map(valor_acelerador,0,1023,40,247)`

Y tendremos que limitar el valor de salida para no proporcionar una tensión superior a la que pueda soportar el motor delantero para ello:

```
if(valor_acelerador_traducido >= 165) // SE LIMITA EL MÁXIMO
{
    valor_acelerador_traducido = 165;
}
```

Y posteriormente, a la lectura del acelerador, tendremos que añadir la influencia del tanto por ciento proporcional al que queremos que se le dé una tensión al motor delantero respecto a la señal del acelerador y también la influencia del limitador de potencia.

A continuación se muestra la sección de código del programa que equivale a comandar una tensión al motor delantero un tanto por ciento proporcional al accionamiento del acelerador:

```
/* FUNCION PARA DAR UNA TRACCIÓN UN TANTO POR CIENTO PROPORCIONAL */
void TRACCION_PROPORCIONAL()
{
    Automatico = FuncionamientoManual;
    nuevo_Automatico = FuncionamientoManual;
    valor_acelerador = analogRead(Acelerador); // LEEMOS EL VALOR QUE LLEGA POR EL PIN
    AL QUE SE CONECTA EL ACELERADOR
    valor_acelerador_traducido=map(valor_acelerador,0,1023,40,247); // Con el mapeado
    conseguimos que ambas ruedas empiecen a funcionar a la vez pero no significa que ambas
    ruedas circulen constantemente a la misma velocidad.
    if(valor_acelerador_traducido >= 165) // SE LIMITA EL MÁXIMO
    {
        valor_acelerador_traducido = 165;
    }
    /*valor_acelerador_traducido_float =
    valor_acelerador_traducido*proporcion_acelerador*valor_nuevo;
    valor_acelerador_traducido = valor_acelerador_traducido_float;
    if(valor_acelerador_traducido_float - valor_acelerador_traducido > 0.5)
    {
        valor_acelerador_traducido += 1;
    }*/
    analogWrite(controlador_delantero,valor_acelerador_traducido);
    tiempo_anterior_PID = micros();
}
/* FIN FUNCION PARA DAR UNA TRACCIÓN UN TANTO POR CIENTO PROPORCIONAL */
```



Con esta parte del programa conseguimos dar una tensión totalmente proporcional al acelerador, pero no un tanto por ciento proporcional y se tiene en cuenta el regulador de potencia.

Si está planteada la posibilidad de que sea un tanto por ciento proporcional y que el regulador de potencia afecte a esta señal, está como comentario y sin testear.

8. Pruebas realizadas

En el desarrollo de este control de tracción de una motocicleta eléctrica y para comprobar su correcto funcionamiento, se diseñó y construyó los elementos, explicados anteriormente, tanto a nivel hardware como software, con los que se realizaron una serie de pruebas: Previamente a las pruebas físicas desarrolladas, se realizó una simulación mediante Simulink para tanto facilitar el diseño del algoritmo como para poder observar situaciones que no se tenían en cuenta anteriormente. Posteriormente se diseñó el algoritmo del sistema, con el cual se realizó el diseño software de nuestro control de tracción.

Pruebas realizadas:

1. La elección del modo de funcionamiento del control de tracción mediante un conmutador de cuatro posiciones.
2. La visualización del modo seleccionado. Para ello se utilizaron 4 leds y si en un futuro se desea incluir una pantalla LCD, se han incluido unos mensajes identificativos de modo que se transmitirán mediante los pines 0 y 1 de Arduino los cuales son por los que se transmiten y reciben la información. Además tener en cuenta que el LCD tiene que incorporar un aplicativo para utilizar sólo 4 pines de Arduino, estos son SDA, SCL, 5V y GND e incluir la librería LiquidCrystal_I2C.
3. La identificación de las velocidades de las respectivas ruedas, para lo cual se hizo necesaria la utilización de las interrupciones externas de Arduino y de unos sensores Hall. Se diseñó y realizó el filtro de estas señales para evitar que las señales espúreas tengan efecto en nuestro sistema. Y además se colocaron unos condensadores de desacoplo próximos a la alimentación de cada sensor hall para evitar que la tensión de estos caiga por cualquier motivo y ello pueda provocar falsas lecturas.
4. Se programó la lógica y las decisiones a tomar por nuestro sistema en función de la comparación de las velocidades, para identificar la situación en la que se encontraba dicho vehículo en cada momento, y en función de la situación en la que se encuentre el vehículo y del modo seleccionado por el piloto se comandará la señal de actuación al motor delantero.
5. Para la entrega de tensión de nuestro sistema al motor delantero se tuvo que adecuar la señal entregada por nuestro sistema al motor, debido a que la salida de nuestro sistema era pseudo-analógica (pwm) y el motor necesariamente requiere una señal analógica. Para ello se diseñó un filtro RC de paso bajo, mediante una serie de pruebas con la ayuda del osciloscopio, observándose la salida de este junto con el

rizado de la señal y se seleccionó los valores del RC con los que se obtuvo mejores resultados.

6. Para la transmisión de una tensión mínima, al motor delantero y evitar así que este ejerza una fuerza que se oponga al giro de la rueda debido a los campos magnéticos de los imanes de este motor, se realizó un estudio de que tensión entregar a este para que sea la suficiente como para evitar esa oposición al movimiento sin que esta gire debido a esta tensión entregada.
7. Para el caso de entregar al motor delantero una tensión específica en función de la diferencia de velocidades de las ruedas hasta corregir esa diferencia de velocidades e igualar estas se realizó un control en lazo cerrado en el cual el controlador elegido es un PID. La identificación de los términos de este controlador se realizó mediante el método de prueba y error hasta conseguir los resultados deseados.
8. Para el caso de entregar una tensión un tanto por ciento proporcional al accionamiento del acelerador se realizó la lectura del sensor hall lineal que lleva el acelerador mediante un pin analógico del Arduino y en función del accionamiento de este se comandará una tensión un tanto por ciento proporcional a la lectura realizada.
9. Se diseñó y realizó un regulador de potencia de la rueda delantera con el cual limitar la potencia entregada al motor delantero. Para ello se utilizó un potenciómetro con el cual junto con la lectura de su salida mediante un pin analógico de Arduino poder tener acceso a la potencia entregada al motor delantero y así regularla a gusto de cada conductor.
10. En el caso de entregar al motor delantero una tensión en función de la diferencia de velocidades hasta igualar estas, para evitar que nuestro sistema esté en constante cambio sobre la forma de actuar en la rueda delantera, esta señal se decidió mantener durante tres segundos a excepción de que la rueda delantera gire más rápido que la trasera. Esta especificación se diseñó mediante la realización del estudio de una máquina secuencial asíncrona.

9. Conclusiones

Como resultado final de dicho proyecto, me siento satisfecho con el trabajo realizado. Dicho Trabajo de Fin de Grado estaba enfocado para ser realizado por dos estudiantes, uno de la especialidad de Electrónica o Eléctrica y otro de la especialidad de mecánica, para abordar dicho estudio al mismo nivel, tanto en el ámbito del diseño eléctrico y electrónico del control de tracción como del diseño mecánico de la estructura del vehículo eléctrico.

Pese a ello, gracias al entusiasmo con el que afronté dicho TFG y por la motivación de mi tutor José Antonio, decidí adentrarme en el mundo del vehículo eléctrico y en la investigación y realización de dicho control de tracción a las dos ruedas.

El objetivo era el diseño eléctrico y electrónico de un control de tracción, y este se ha realizado con éxito, pese a las dificultades económicas, las cuales fueron una de las limitaciones del proyecto.

Unas de las mayores dificultades que se encontraron en la realización de dicho estudio fueron la adecuación de la señal de salida del Arduino al controlador del motor delantero junto con la creación del PID y de filtrado de los sensores hall con los que obteníamos la medición de la velocidad. Todas ellas resueltas con éxito.

El control de tracción tiene diferentes modos de funcionamiento, seleccionables mediante un conmutador de cuatro posiciones.

El funcionamiento de los cuatro modos fue verificado mediante la construcción de una estructura de pruebas y además se incluyó a mayores un limitador de potencia (por testear en futuros proyectos) y algunas modificaciones posibles que puedan ser utilizadas para futuros proyectos que sean una continuación de este ya realizado.

En general, este estudio ha merecido la pena y los resultados obtenidos superan las dificultades encontradas en la realización del proyecto.

Este TFG me ha servido para darme cuenta de que ni lo que parece fácil resulta ser tan fácil, ni que lo que parece difícil resulta ser tan complicado. Con empeño, ganas y esfuerzo se puede conseguir el propósito deseado.

Por último, agradecer tanto al departamento de tecnología electrónica como a la asociación LACECAL y la empresa Enriding por el apoyo económico para la realización de dicho trabajo y de la facilitación de las instalaciones necesarias para el desarrollo práctico del mismo. Y como no, agradecer al tutor de este

proyecto José Antonio Domínguez Vázquez por el apoyo recibido durante la realización del mismo.



10. Bibliografía

ALBA Ingenieros Consultores S.L. (2007). *Guía de la energía en el Sector del Automóvil*.

Arconada, L. F. (s.f.). Recuperado el 9 de 2015, de http://www.isa.cie.uva.es/~felipe/docencia/ra12itielec/tema1_trasp.pdf

Arduino. (s.f.). Recuperado el 8 de 2015, de <https://www.arduino.cc/en/Main/ArduinoBoardUno>

Basicx. (s.f.). Recuperado el 8 de 2015, de <http://www.basicx.com/>

Electrical-Knowhow. (s.f.). Recuperado el 7 de 2015, de <http://www.electrical-knowhow.com/2012/05/classification-of-electric-motors.html>

Electrical-KnowHow. (s.f.). Recuperado el 7 de 2015, de http://www.electrical-knowhow.com/2012/05/classification-of-electric-motors-part_24.html

Electrical-KnowHow. (s.f.). Recuperado el 7 de 2015, de <http://www.electrical-knowhow.com/2012/05/classification-of-electric-motors-part.html>

Electrical-KonwHow. (s.f.). Recuperado el 7 de 2015, de http://www.electrical-knowhow.com/2012/05/classification-of-electric-motors-part_22.html

Endesa. (s.f.). *Endesa*. Recuperado el 06 de 2015, de Vehículo eléctrico: <http://www.endesavehiculoelectrico.com/vehiculo-electrico/el-vehiculo/tipos>

Freedom defined. (s.f.). Recuperado el 8 de 2015, de <http://freedomdefined.org/OSHW>

Gizmag. (s.f.). Recuperado el 7 de 2015, de <http://www.gizmag.com/ktm-2wd-hybrid-dirt-bike/10348/>

Guillén, J. G. (06 de 2011). Proyecto final de carrera. *MErcado de la motocicleta de combustión interna y oportunidades de la industria de la motocicleta eléctrica*. Madrid: Universidad Carlos III.

HandyBoard. (s.f.). Recuperado el 8 de 2015, de <http://handyboard.com/hb/about/>

Iberdrola. (s.f.). Recuperado el 05 de 2015, de <https://www.iberdrola.es/clientes/hogar/movilidad-verde/vehiculos>

Juárez, R. E. (2013). La persistente atracción de la doble tracción. *Informoto*, 4.

Laboratorio de Sistemas Electrónicos e Instrumentación. (s.f.). Recuperado el 10 de 2015, de <http://labs.eie.ucv.cl/labsei/NotasTecnicas/Desacoplo/desacoplo.htm>

Medina, D. Á. (s.f.). *Youtube.* Recuperado el 10 de 2015, de <https://www.youtube.com/watch?v=vL9AVsyFpvl>

Motor Pasion. (s.f.). Recuperado el 8 de 2015, de <http://www.motorpasionmoto.com/tecnologia/yamaha-2-trac-traccion-en-las-dos-ruedas>

Motos eléctricas Xispa. (s.f.). Recuperado el 6 de 2015, de <http://www.mismotoselectricas.com/marcas-modelos/xispa-motorbikes>

Oset Bikes. (s.f.). Recuperado el 6 de 2015, de <http://osetbikes.es/>

Parallax. (s.f.). Recuperado el 8 de 2015, de <http://www.parallax.com/>

Phidgets. (s.f.). Recuperado el 8 de 2015, de <http://www.phidgets.com/>

Playground. (s.f.). Recuperado el 9 de 2015, de <http://playground.arduino.cc/ArduinoNotebookTraduccion/Appendix3>

Processing. (s.f.). Recuperado el 8 de 2015, de <https://www.processing.org/>

Prometec. (s.f.). *Prometec.* Recuperado el 12 de 2015, de <http://www.prometec.net/bus-i2c/>

Slideplayer. (s.f.). Recuperado el 7 de 2015, de <http://slideplayer.es/slide/1078489/>

Taller de Arduino. (s.f.). Recuperado el 10 de 2015, de <http://tallerarduino.com/2012/02/10/arduino-tutorials-cap-4-interrupciones-externas/>

Vázquez, J. A. (2014). Apuntes de la asignatura Electrónica Industrial, 4º Curso del Grado en Electrónica y Automática Industrial.

Wikipedia. (s.f.). Recuperado el 8 de 2015, de <https://es.wikipedia.org/wiki/Arduino>

Wikipedia. (s.f.). Recuperado el 6 de 2015, de https://es.wikipedia.org/wiki/Motocicleta_el%C3%A9ctrica#cite_note-4

Wikipedia. (s.f.). Recuperado el 5 de 2015, de https://en.wikipedia.org/wiki/Electric_motorcycles_and_scooters

Wiring. (s.f.). Recuperado el 8 de 2015, de <http://wiring.org.co/>



Wordpress. (s.f.). Recuperado el 5 de 2015, de
<https://evmc2.wordpress.com/2013/07/27/the-history-of-electric-motorcycles-notes/>

Wordpress. (s.f.). Recuperado el 6 de 2015, de
<https://evmc2.wordpress.com/category/history/>

Xispa motos eléctricas. (s.f.). Recuperado el 6 de 2015, de
<http://www.electromaps.com/motos-electricas/xispa/e-25>

11. Anexos

11.1 Tipos de motos eléctricas de Oset Bikes

A continuación, se van a presentar algunos tipos de motocicletas eléctricas de esta marca, que se encuentran actualmente en el mercado:

MOTO ELÉCTRICA OSET 20.0 RACING 2015

Peso 35 Kg

Neumático 20x2.0 delantero y rueda trasera completa más ancha 14x3.0

Potencia 1200 W 48 V



Imagen 11.1:1 – Oset 20.0
Racing 2015

MOTO ELÉCTRICA OSET 20.0 ECO 2015

Peso 35 Kg

Neumático 20x2.0 delantero y trasero Maxxis

Potencia 1200 W 48 V



Imagen 11.1:2 – Oset 20.0 Eco
2015

MOTO ELÉCTRICA OSET 16.0 ECO 2015

Motor Exclusivo OSET de 800W, más eficaz y mayor calidad



Imagen 11.1:3 – Oset 16.0
Eco 2015

MOTO ELÉCTRICA OSET 12.5 R 2015

Motor de 600 W de diseño oset más eficaz y potente

2 x baterías de 12V/10Ah, más capacidad=más autonomía



**Imagen 11.1:4 – Oset 12.5
R 2015**

11.2 Tipos de motos eléctricas de Xispa Motorbikes

Algunos tipos de motocicletas eléctricas de esta marca, actualmente en el mercado, son:

MOTO ELÉCTRICA XISPA TRIAL E 25

Precio: 1.374 €

Motor: { 48 voltios
Potencia: 350 W
Pico de potencia: 3000 W

Batería: LifePo4

Autonomía: { Configuración modo 1: 2 horas
Configuración modo 2: 3 horas

Peso: 21 kg

Peso máximo del conductor: 25 kg



Imagen 11.2:1 – Xispa Trial E 25

MOTO ELÉCTRICA XISPA ENDUCROSS E 10

Precio: 1.482 €

Motor: { 48 voltios
Potencia: 350 W
Pico de potencia: 1000 W

Batería: LifePo4

Autonomía: { Configuración modo 1: 2 horas
Configuración modo 2: 3 horas

Peso: 22 kg

Peso máximo del conductor: 25 kg



Imagen 11.2:2 – Xispa Enducross E 10

MOTO ELÉCTRICA XISPA SUPERMOTARD E 10

Precio: 1.565 €

Motor: {
Xispa Electric (brushless)
48 voltios
Potencia: 350 W
Pico de potencia: 1000 W

Batería: LifePo4

Autonomía: {
Configuración modo 1: 2 horas
Configuración modo 2: 3 horas

Peso: 22 kg

Peso máximo del conductor: 25 kg



Imagen 11.2:3 – Xispa Supermotard E 10

11.3 Investigación de los motores empleados en los sistemas de tracción de VE, los motores brushless

11.3.1 Introducción y clasificación de motores eléctricos

Los motores eléctricos permiten transformar la energía eléctrica en mecánica. Este hecho fundamental, hace de los motores eléctricos la solución más adecuada en entornos industriales, domésticos o como en nuestro caso en sistemas de tracción de vehículos eléctricos, ya sea a modo de apoyo, como de sustitución de los motores de combustión interna.

Al ser la energía eléctrica el tipo de energía inicial, el uso de estos motores, cuenta con todas las ventajas que se derivan de este hecho.

Los motores eléctricos, se pueden clasificar principalmente en función de la fuente de tensión que alimente al motor y de las principales características que los diferencian:

1. Motores de corriente alterna (AC)

- 1.1 Motores de AC asíncronos
 - Jaula de ardilla
 - Rotor bobinado
- 1.2 Motores de AC síncronos
 - Motores no excitados.
 - Motores de reluctancia y de reluctancia variable
 - Motores de histéresis
 - Motores de imán permanente
 - ✚ Motores de Flujo axial
 - Motores de CC con excitación.
- 1.3 Motores de AC Lineales

2. Motores de corriente continua (DC)

- 2.1 Motores DC con escobillas
 - 2.1.1 Motores de DC con estator bobinado
 - Serie
 - Paralelo
 - Compuesto
 - 2.1.2 Motores de DC de imanes permanentes
 - 2.1.3 Motores de DC especiales
 - Servomotores
 - Motores paso a paso
- **2.2 Motores de DC sin escobillas**

Dentro de cada tipo hay una amplia variedad de ellos, pero nos centraremos en los motores de corriente continua, que son los más utilizados en los vehículos de tracción eléctrica.

11.3.2 Motores DC

Un motor DC, recibe una corriente y a partir de ella produce una rotación mecánica en el eje. La misma máquina es capaz de invertir el proceso, entonces se le denomina generador.

Los sistemas de alimentación de corriente continua, no son muy comunes en la práctica de la ingeniería contemporánea, sin embargo se han utilizado en aplicaciones industriales durante años tanto en elevadoras, extrusoras, aplicaciones textiles, aviones, automóviles...etc.

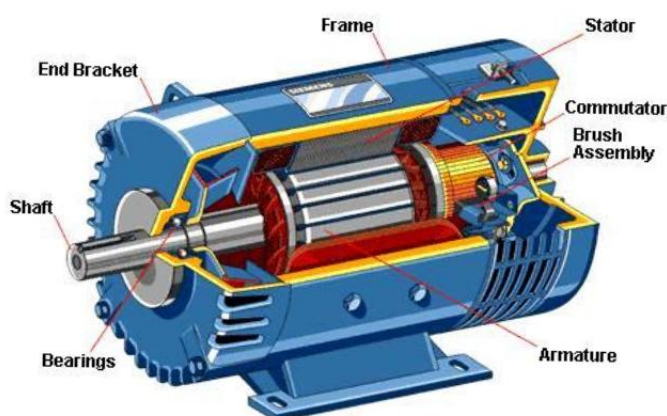


Imagen 11.3.2:1 – Motor DC

Los motores de corriente continua están compuestos fundamentalmente por: un rotor, un estator, un conmutador y unas escobillas (los motores brushless carecen de estas últimas).

El **rotor**, puede estar compuesto por uno o más devanados (rotor bobinado), o por un imán permanente (rotor de imanes permanentes). Cuando estos devanados son excitados se produce un campo magnético. Los polos magnéticos, del campo magnético creado por el rotor, se sentirán atraídos por los polos opuestos generados por el estator, provocando que el rotor gire. El rotor es la parte principal del motor, gira sobre dos cojinetes creando par. La rotación, por lo tanto, se produce por la circulación de corriente por los devanados de las bobinas.



Imagen 11.3.2:2 - Rotor de un motor DC

A medida que el motor gira, los devanados se excitan siguiendo una secuencia, provocando que el campo magnético, que crea el rotor, cambie y sea atraído por el campo magnético del estator, produciendo el giro del rotor de forma continua.

El **conmutador o colector de delgas**, consiste en un conjunto de segmentos de cobre fijados alrededor de la circunferencia del rotor. Cada segmento debe estar separado de los otros, además cada contacto actúa como un interruptor; ya que la corriente fluye sólo a través del bobinado del rotor, con el que las escobillas están en contacto. Este flujo de corriente crea una fuerza electromotriz, que a su vez es la responsable de crear el par.

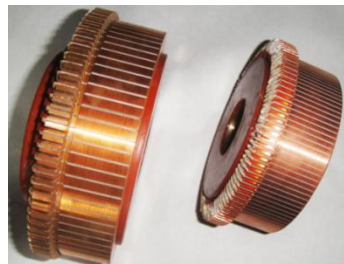


Imagen 11.3.2:3 - Conmutador de un motor DC

El **estator**, genera un campo magnético estacionario que rodea el rotor. Este campo magnético, se genera mediante un conjunto de imanes permanentes, también conocidos como polos de campo. Estos polos son curvados para adaptarse a la forma del rotor. Este elemento, es el que induce el campo magnético al rotor.



Imagen 11.3.2:4 - Estator de un motor DC

Y por último, las **escobillas**, es la parte del motor que está compuesta de un material conductor, similar al carbón colocada en la “cola del motor”.

En los motores o generadores eléctricos, se debe establecer una conexión eléctrica de la parte fija de la máquina con las bobinas del rotor que gira.

Para realizar esta conexión, se fijan dos anillos en el eje de giro, generalmente de cobre, aislados de la electricidad del eje y conectados a los terminales de la bobina rotatoria. Enfrentados a los anillos, se disponen unos bloques de carbón, que mediante unos resortes, hacen presión sobre ellos para establecer el contacto eléctrico necesario. Estos bloques de carbón, son las anteriormente denominadas escobillas y los anillos rotatorios forman el anteriormente denominado colector de delgas.



Imagen 11.3.2:5 - Escobillas de un motor DC

Funcionamiento de un motor DC

Para que un motor DC funcione como tal, se tiene que suministrar una tensión DC externa a las bobinas del rotor. Este funcionamiento se basa en el Teorema de Oersted, en el que pone de manifiesto que si por un hilo conductor circula una corriente eléctrica, este creaba un campo magnético. Ese campo magnético se puede calcular por medio de la Ley de Biot y Savart.

$$B = \frac{\mu \cdot \pi}{2 \cdot \pi \cdot r}$$

Con la creación de ese campo y relacionándolo con la segunda ley de Lorentz, podemos observar que un hilo conductor, con corriente dentro de un campo, sufre una fuerza mecánica.

$$F = L \cdot I \cdot B \cdot \text{sen}\theta$$

Dónde L es la longitud del conductor, I es la corriente que recorre el conductor, B es la intensidad del campo magnético en teslas y θ el ángulo entre el campo y la corriente.

Al provocar un campo magnético en el rotor y enfrentarse al campo magnético del estator, los polos iguales se repelerán, por lo que el rotor se ve impulsado a realizar un movimiento de rotación. Si el campo del rotor va cambiando continuamente, al modificar el suministro de tensión sobre las bobinas del rotor, se conseguirá un movimiento continuo del rotor.

Ventajas de los motores DC:

- Su velocidad, es fácil de controlar en un amplio rango. Su curva característica par-velocidad, es más fácil de adaptar que la de motores de CA. Esta es la razón por la que, la mayoría de los motores de tracción y servomotores han sido máquinas de corriente continua.
- Sus reducidas dimensiones permiten un considerable ahorro de espacio.

Desventajas de los motores DC:

- Necesitan de escobillas para conectar el devanado del rotor. Se produce desgaste de las escobillas, y que aumenta de manera espectacular en el medio ambiente de baja presión. Por lo que, no se pueden utilizar en corazones artificiales. Si se utilizaran en los aviones, las escobillas necesitarían reemplazo después de una hora de funcionamiento. Además, debido a estas escobillas, se produce fricción y con lo cual una reducción de la eficiencia.
- Las chispas de las escobillas pueden causar una explosión si el entorno contiene material explosivo.
- El ruido de RF (radio-frecuencia), de las escobillas, puede interferir en dispositivos electrónicos.
- Los motores de corriente continua son caros en relación a los motores de CA.

Algunos de los motores de corriente continua más utilizados en los sistemas de tracción de vehículos eléctricos son:

Brush DC Motors (BDC)

Estos, son motores de corriente continua con escobillas, por lo tanto la conmutación se realiza en su interior mediante estas, y necesita de una fuente de alimentación de corriente continua. Son ampliamente utilizados.

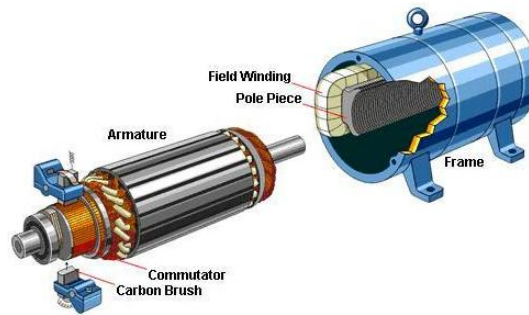


Imagen 11.3.2:6 - Motor BDC

Ventajas:

- Son de bajo costo, fáciles de manejar y hay una amplia gama de formas y tamaños.

Los diferentes tipos de motores de BDC, se distinguen por la construcción del estator, o la forma en que los devanados electromagnéticos están conectados a la fuente de alimentación.

Algunos de los tipos de motores BDC, utilizados en vehículos de tracción, son por ejemplo los de imanes permanentes.

Imán permanente

Es un tipo de motor de corriente continua con escobillas, cuyos polos son unos imanes permanentes. Estos imanes permanentes, son los que producen el campo del estator.

Ventajas:

- Al no necesitar ningún circuito de campo externo, no hay pérdidas en el cobre.
- Como no se necesitan devanados de campo, pueden ser más pequeños (menor volumen).
- Son muy utilizados en aplicaciones de baja potencia (bicicletas).
- El devanado de campo se sustituye por un imán permanente (construcción simple).
- No hay requisito de excitación externa.
- Tienen mayor densidad de potencia y mejor rendimiento que las máquinas asíncronas.

Desventajas:

- Dado que los imanes permanentes producen densidades de flujo más débiles que los campos de derivación con apoyo externo, estos motores producen menor par.
- Siempre hay un riesgo de desmagnetización, debido al calentamiento o debido a los efectos de la reacción del inducido.

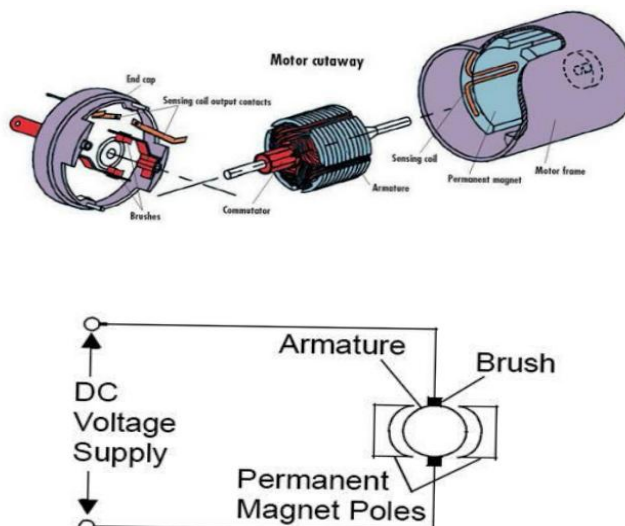


Imagen 11.3.2:7 - Motor de imanes permanentes

Motor brushless DC

Como su propio nombre indica, los motores BLDC no utilizan escobillas en la conmutación para realizar la excitación del rotor, sin embargo, son conmutados electrónicamente mediante un control electrónico externo, por ello no hay ese desgaste, ni chispas, ni reducción del rendimiento por fricción o calor que había en los BDC por el colector de delgas, ni requieren tanto mantenimiento.

Los motores de corriente continua sin escobillas (BLDC), son uno de los tipos de motores que más popularidad están ganando en los últimos años. Los Motores BLDC se utilizan en industrias tales como electrodomésticos, automóviles eléctricos, aeroespacial, equipos médicos, equipos industriales de automatización e instrumentación, etc.

Estos motores, son un tipo de motor síncrono. Esto significa que, el campo magnético generado por el estator y el campo magnético generado por el rotor, giran a la misma frecuencia.

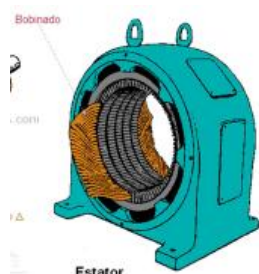


Imagen 11.3.2:8 - Estator

Los motores BLDC, pueden tener una configuración de una sola fase, de dos o de tres fases. Correspondientemente a su tipo, el estator tiene el mismo número de láminas de acero, con devanados colocados en las ranuras cortando axialmente la periferia interior. De estos, los motores de 3 fases son los más populares y ampliamente utilizados.

El rotor constituye la parte móvil del motor. El rotor es el elemento de transferencia mecánica, ya que de él depende la conversión de energía eléctrica a mecánica.

El motor brushless DC incorpora como rotor imanes permanentes, que pueden variar de dos a ocho pares de polos alternándose Norte y Sur, en vez de circuito inductor:



Imagen 11.3.2:9 - Rotor

- 1) La conmutación se realiza de forma electrónica en lugar de mecánica.
- 2) Los imanes permanentes van alojados en el rotor en lugar de en el estator.
- 3) Las bobinas del inducido van alojadas en el estator, constituyendo un devanado monofásico o polifásico.

Su funcionamiento, se basa en la alimentación secuencial de cada una de las fases del estator, de forma sincronizada con el movimiento del rotor. La alimentación, por medio de corriente continua, simplifica la electrónica del convertidor.

Existen dos tipos de motores brushless, que se diferencian por la forma de alimentar a los devanados del estator. Estos dos tipos son: los motores trapezoidales, también conocidos como brushless DC (BLDC), y los motores brushless sinusoidales, también llamados motores síncronos de imanes permanentes (PMSM).

En los motores BLDC, cada fase se alimenta con pulsos de tensión rectangulares desfasados 120° . Estos suelen ser motores de pequeña potencia, de prestaciones dinámicas no muy exigentes y más bruscos en su funcionamiento. Sin embargo, los motores PMSM se alimentan con tensión alterna trifásica, son motores de altas prestaciones que se utilizan principalmente en servosistemas y son más suaves en su funcionamiento.

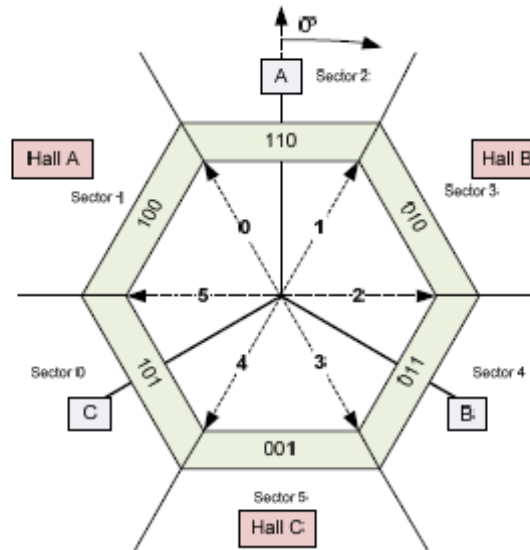


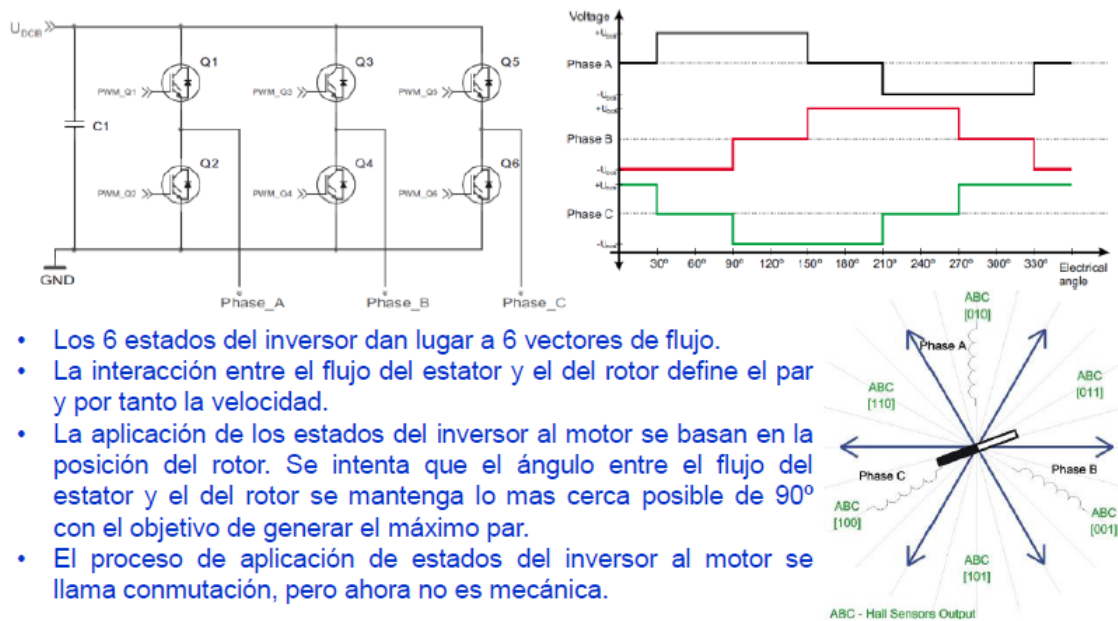
Imagen 11.3.2:10 - Representación de un PMSM con sensores de efecto Hall. Se pueden apreciar lo seis sectores definidos por la combinación de señales Hall, así como los vectores de excitación de cada sector para movimientos clockwise

Como ya se ha especificado anteriormente, a diferencia de un motor de corriente continua con escobillas, la conmutación de un motor BLDC se controla electrónicamente. Para girar el motor BLDC, los devanados del estator deben ser excitados siguiendo una secuencia. Es importante conocer la posición del rotor, con el fin de saber que devanado debe ser excitado siguiendo la secuencia de excitación.

La posición del rotor, será detectada mediante sensores de efecto Hall integrados en el estator, por ello la mayoría de los motores BLDC tienen tres sensores Hall integrados en el estator.

Cada vez que los polos magnéticos del rotor pasan cerca de los sensores Hall, estos dan una señal alta o baja, indicando que el polo norte o sur está pasando cerca de los sensores. Basándose en la combinación de estas tres señales de los sensores Hall, se puede determinar la secuencia exacta de la conmutación.

Un inversor, es el encargado de generar las tensiones positivas o negativas, que se aplican a los devanados del estator a partir de una fuente DC.



- Los 6 estados del inversor dan lugar a 6 vectores de flujo.
- La interacción entre el flujo del estator y el del rotor define el par y por tanto la velocidad.
- La aplicación de los estados del inversor al motor se basan en la posición del rotor. Se intenta que el ángulo entre el flujo del estator y el del rotor se mantenga lo mas cerca posible de 90° con el objetivo de generar el máximo par.
- El proceso de aplicación de estados del inversor al motor se llama conmutación, pero ahora no es mecánica.

Imagen 11.3.2:11 - Representación de los seis estados del inversor

Los sensores Hall pueden estar desfasados 60° ó 120° entre sí. Basándonos en esto, el fabricante del motor definirá la secuencia de conmutación, que debe seguirse cuando se controla el motor.

Nota: Los sensores Hall requieren una fuente de alimentación determinada. El voltaje puede variar de 4 voltios a 24 voltios. La Corriente requerida puede variar de 5 a 15mAmps.

Los motores brushless trabajan por medio de variadores, también llamados controladores de velocidad (electronic speed controller o ESC), que transforman la corriente continua de las baterías en una tensión alterna trifásica y la alimentan a los bobinados en cierta secuencia dependiendo de la posición del rotor en cada momento, para lo cual se utilizan dos técnicas básicamente, dependiendo de la existencia o no de sensores en el motor, lo que los divide en dos familias: con sensores (sensored) y sin sensores (sensorless)

Sensored: Disponen de sensores de efecto hall o de encoders, que indican la posición del rotor. Es habitual que tengan 3 sensores separados 120 grados, uno para cada devanado del motor.

Sensorless: No tienen sensores; la posición se determina mediante la medición del efecto de la fuerza contraelectromotriz sobre los devanados.

A continuación se muestra un ejemplo, de una secuencia a realizar por el inversor, para que el motor gire en el sentido de las agujas del reloj:

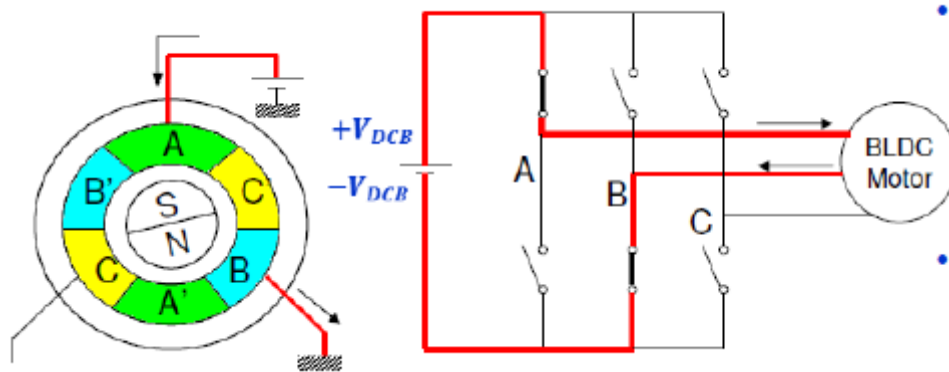


Imagen 11.3.2:12 - Primera secuencia del inversor

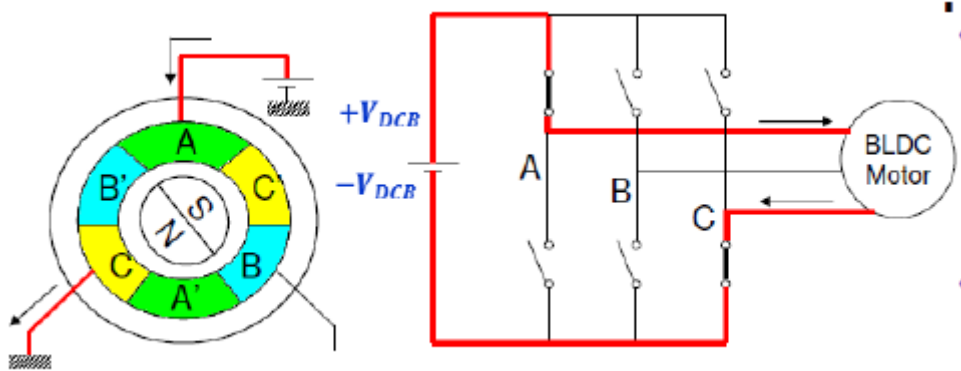


Imagen 11.3.2:13 - Segunda secuencia del inversor

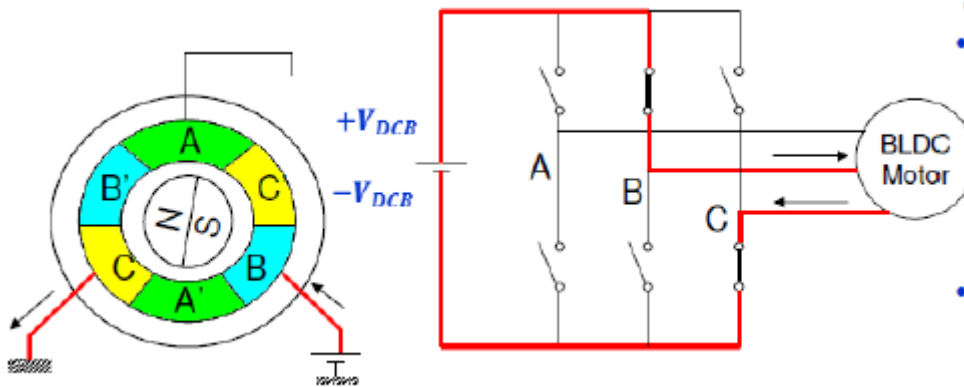


Imagen 11.3.2:14 - Tercera secuencia del inversor

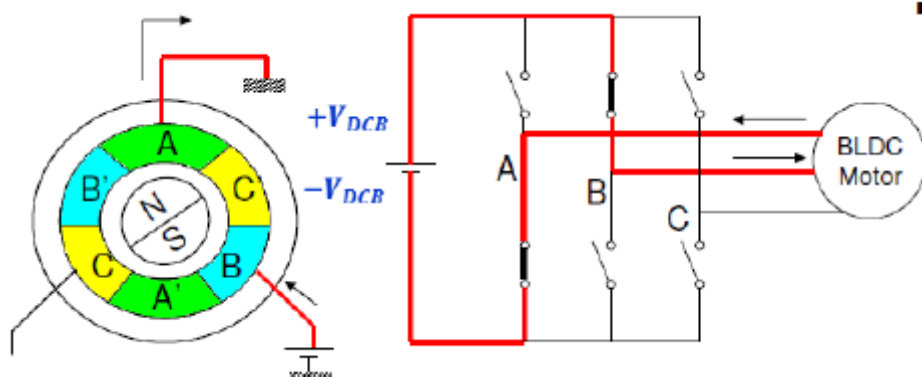


Imagen 11.3.2:15 - Cuarta secuencia del inversor

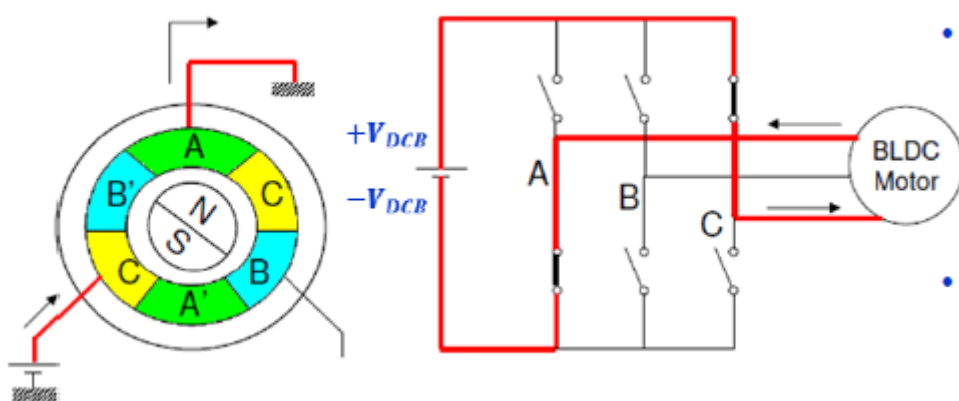


Imagen 11.3.2:16 - Quinta secuencia del inversor

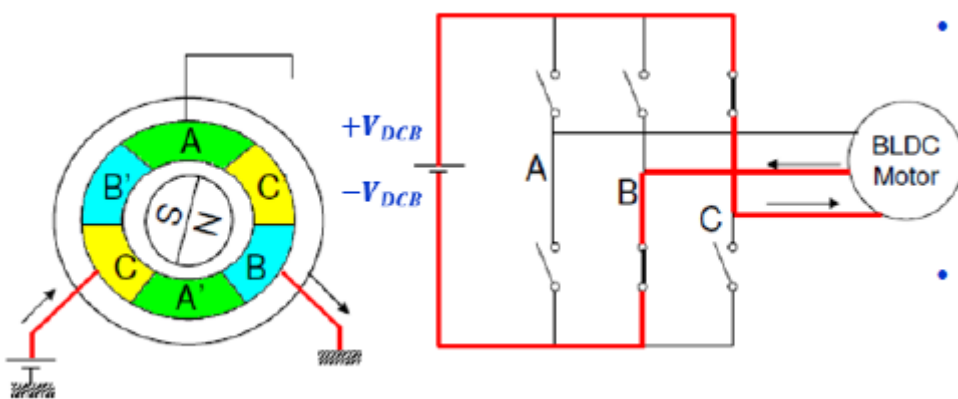


Imagen 11.3.2:17 - Sexta secuencia del inversor

CW

Table 3. Clockwise direction

Commutation vector			Vector	Hall sensor pattern definition			Hall sensor pattern result
Phase A	Phase B	Phase C		Hall Sensor C	Hall Sensor B	Hall Sensor A	
NC	+V _{DCB}	-V _{DCB}	A	1	0	1	5
-V _{DCB}	+V _{DCB}	NC	B	1	0	0	4
-V _{DCB}	NC	+V _{DCB}	C	1	1	0	6
NC	-V _{DCB}	+V _{DCB}	D	0	1	0	3
+V _{DCB}	-V _{DCB}	NC	E	0	1	1	2
+V _{DCB}	NC	-V _{DCB}	F	0	0	1	1

CCW

Table 4. Counterclockwise direction

Commutation vector			Vector	Hall sensor pattern definition			Hall sensor pattern result
Phase A	Phase B	Phase C		Hall Sensor C	Hall Sensor B	Hall Sensor A	
NC	-V _{DCB}	+V _{DCB}	D	1	0	1	5
-V _{DCB}	NC	+V _{DCB}	C	0	0	1	1
-V _{DCB}	+V _{DCB}	NC	B	0	1	1	3
NC	+V _{DCB}	-V _{DCB}	A	0	1	0	2
+V _{DCB}	NC	-V _{DCB}	F	1	1	0	6
+V _{DCB}	-V _{DCB}	NC	E	1	0	0	4

Las combinaciones [000] y [111] de los sensores Hall indican error.

Tabla 8 – Vectores de conmutación y activación de los sensores Hall

Explicación del modo de funcionamiento:

El par, se produce debido a la interacción entre el campo magnético generado por los devanados del estator y los imanes permanentes del rotor. Con el fin de mantener el motor en marcha, el campo magnético producido por los devanados debe cambiar de posición, ya que el rotor se mueve para alinearse con el campo del estator. Lo que define la secuencia de activación de los devanados, es lo que se conoce como "conmutación en seis pasos".

En la conmutación en seis pasos, sólo dos de los tres devanados, del motor DC sin escobillas, se excitan a la vez. Cada secuencia de conmutación tiene: uno de los devanados excitados positivamente (la corriente entra en el bobinado), el segundo devanado se encuentra excitado negativamente (la corriente sale del bobinado) y el tercero no está excitado.

Cada paso es equivalente a que el rotor gire 60 grados, por lo que seis pasos hacen una rotación completa de 360 grados del rotor. Suele ser útil en aplicaciones que requieren una velocidad de conmutación a altas frecuencias.

Aplicaciones típicas de motor BLDC

Podemos clasificar el tipo de control de motor BLDC en tres tipos principales:

1. Las aplicaciones con carga constante:

En este tipo de aplicaciones, una velocidad variable es más importante que mantener con exactitud una velocidad establecida. Además, las aceleraciones y deceleraciones no están cambiando dinámicamente. En estos tipos de aplicaciones, la carga se acopla directamente al eje del motor. Por ejemplo, ventiladores, bombas y sopladores vienen bajo este tipo de aplicaciones. Estas aplicaciones requieren controladores de bajo coste, en su mayoría operan enlace abierto.

2. Las aplicaciones con cargas variables:

En este tipo de aplicaciones, en las que la carga en el motor varía en un rango de velocidades, se puede exigir una precisión de control de alta velocidad y buenas respuestas dinámicas.

Por ejemplo:

- En el hogar: lavadoras, secadoras y compresores.
- En el control del automóvil de combustible de la bomba, control de la dirección electrónica, control del motor y el control del vehículo eléctrico.
- En el sector aeroespacial, hay una serie de aplicaciones, como centrífugas, bombas, controles del brazo robótico, controles giroscopio y así sucesivamente.

Estas aplicaciones, pueden utilizar dispositivos de retroalimentación de velocidad y pueden funcionar en lazo semicerrado o total del bucle cerrado.

3. Las aplicaciones de posicionamiento.

La mayoría de las automatizaciones entran en esta categoría. Las aplicaciones de esta categoría tienen algún tipo de transmisión de energía, que podría ser engranajes mecánicos o cinturonas de temporizador, o un sistema accionado por correa simple. En estas aplicaciones, la respuesta dinámica de la velocidad y el par son importantes. Estas aplicaciones pueden tener inversión del sentido de giro. Estos sistemas operan principalmente en bucle cerrado.

Ventajas de este tipo de motores respecto a los de DC con escobillas (BDC):

1. Tienen buena respuesta dinámica.
2. Tiene una vida útil elevada y no requieren de un mantenimiento periódico como los de escobillas.
3. Son silenciosos.
4. Tienen una mayor eficiencia
5. Los rangos de velocidad son mayores y mejor relación velocidad – par motor.
6. No producen chispas.
7. Son más baratos de fabricar y pesan menos.

Inconvenientes:

1. Se necesitan sensores para detectar la posición del rotor. Esta detección, de la ubicación de los polos magnéticos, normalmente se hace con sensores de efecto Hall, aunque existen modelos que utilizan sensores ópticos, que funcionan de manera similar a los encoders.

2. Requieren sistemas electrónicos para realizar la conmutación.

Por último, en las siguientes imágenes, se muestra una comparación entre los motores con escobillas de corriente continua (BDC) y los motores DC sin escobillas (BLDC).

Feature	BLDC Motor	Brushed DC Motor
Commutation	Electronic commutation based on Hall position sensors.	Brushed commutation.
Maintenance	Less required due to absence of brushes.	Periodic maintenance is required.
Life	Longer.	Shorter.
Speed/Torque Characteristics	Flat – Enables operation at all speeds with rated load.	Moderately flat – At higher speeds, brush friction increases, thus reducing useful torque.
Efficiency	High – No voltage drop across brushes.	Moderate.
Output Power/Frame Size	High – Reduced size due to superior thermal characteristics. Because BLDC has the windings on the stator, which is connected to the case, the heat dissipation is better.	Moderate/Low – The heat produced by the armature is dissipated in the air gap, thus increasing the temperature in the air gap and limiting specs on the output power/frame size.
Rotor Inertia	Low, because it has permanent magnets on the rotor. This improves the dynamic response.	Higher rotor inertia which limits the dynamic characteristics.
Speed Range	Higher – No mechanical limitation imposed by brushes/commutator.	Lower – Mechanical limitations by the brushes.
Electric Noise Generation	Low.	Arcs in the brushes will generate noise causing EMI in the equipment nearby.
Cost of Building	Higher – Since it has permanent magnets, building costs are higher.	Low.
Control	Complex and expensive.	Simple and inexpensive.
Control Requirements	A controller is always required to keep the motor running. The same controller can be used for variable speed control.	No controller is required for fixed speed; a controller is required only if variable speed is desired.

Tabla 9 – Comparación BLDC Motor Vs Brushed Motor

	Motor DC convencional	Motor DC sin escobillas
Estructura Mecánica	Elementos magnéticos en el estator	Elementos magnéticos en el rotor
Principales Características	Respuesta rápida y excelente controlabilidad	Fácil mantenimiento
Conexión de los embobinados	Conexión Δ	conexión Δ o Y
Método de Conmutación	Contacto mecánico entre las escobillas y el conmutador	Conmutación electrónica por medio de transistores
Método para detectar la posición del rotor	Detectada automáticamente por las escobillas	Sensor de <i>efecto Hall</i> , <i>encoder óptico</i> , etc.
Método de reversa	Cambiando la polaridad del voltaje	Cambiando la lógica

Tabla 10 – Comparación BLDC Motor Vs Brushed Motor



Imagen 11.3.2:18 – Motor BLDC

TIPOS DE MOTORES SIN ESCOBILLAS

Los motores BLDC se pueden fabricar de dos tipos: inrunner y outrunner.

Los motores inrunner desarrollan una mayor velocidad y suelen ser más pequeños. Estos motores, entregan su torque máximo a muy altas revoluciones por minuto, por lo que se usan siempre con engranajes reductores, que pueden ser un inconveniente ya que pueden ser una fuente de averías y de mantenimiento. En estos motores, el elemento móvil es el eje, sobre el cual se encuentran instalados los imanes permanentes, mientras que el bobinado es exterior.



Imagen 11.3.2:19 - Motor Brushless Inrunner

Por otra parte, los motores outrunner desarrollan su torque máximo a velocidades más bajas, por lo que usualmente no necesitan reducción, y se pueden acoplar directamente a un eje. En estos motores, los imanes permanentes están instalados en la carcasa externa del motor, que en este caso es la que gira rodeando al bobinado y a la que se conecta el eje. El bobinado se encuentra en la parte interior central. Cuando gira la carcasa externa lo hace también el eje, pues este se encuentra unido a la carcasa.



Imagen 11.3.2:20 - Motor Brushless Outrunner

11.4 ¿Por qué elegir Arduino Rev3 como plataforma electrónica?

11.4.1 ¿Qué es Arduino?

Arduino es una plataforma de electrónica abierta, para la creación de prototipos, que consta de, software y hardware flexible (Open Source Hardware), cuyo hardware consiste en una placa con un microcontrolador Atmel AVR, varios puertos de entrada/salida digitales y analógicos y un entorno de desarrollo de fácil uso.

El microcontrolador que usa el Arduino Uno Rev 3, es el Atmega328 que se caracteriza por su sencillez y bajo coste, que permite el desarrollo de una gran variedad de diseños.

Arduino puede tomar información del exterior, a través de sus pines de entrada mediante toda una gama de sensores y puede accionar luces, motores y otros actuadores.

El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). Los proyectos realizados con Arduino, pueden ejecutarse sin necesidad de conectarlos a un ordenador, si bien, tienen la posibilidad de hacerlo y comunicar con diferentes tipos de software (p.ej. Flash, Processing, MaxMSP).

11.4.2 ¿Qué significa que Arduino sea de “software libre”?

Un programa es de software libre si da a los usuarios la libertad de poder ejecutarlo, copiarlo y distribuirlo (a cualquiera y a cualquier lugar), estudiarlo, cambiarlo y mejorarlo, sin tener que pedir ni pagar permisos al desarrollador original ni a ninguna otra entidad específica. La distribución de las copias puede ser con o sin modificaciones propias y puede ser gratis o no, el “software libre” es un asunto de libertad no de precio.

11.4.3 ¿Qué significa que Arduino sea de “hardware libre”?

El hardware libre (también llamado “open-source” o de “fuente abierta”) comparte muchos de los principios y metodologías del software libre y de código abierto.

A diferencia del mundo del software libre, donde las licencias son variadas y abundantes, en el ámbito del hardware libre todavía no existen prácticamente licencias, ya que el concepto “hardware libre” es relativamente nuevo.

Algunos de los principios que se pretenden promover para considerar productos como hardware libre son los siguientes:

- Publicar la documentación, incluyendo los archivos de los diseños mismos, que debe permitir su modificación y distribución.
- Ofrecer el software para el visionado de los archivos de diseño y de la documentación, para que se pueda escribir el código open-source fácilmente.
- Ofrecer una licencia que permita producir derivados y modificaciones, además de su re-distribución bajo la licencia original, así como su venta y manufactura.
- Especificar qué proporción del diseño es abierta en caso de que no se liberen todos sus componentes.
- La licencia no debe restringir que se venda o comparta la documentación necesaria.
- La licencia no debe discriminar a ningún grupo o persona.
- La licencia no debe restringir a ningún campo o actividad el uso de la obra.
- El licenciamiento de la obra no puede depender de un producto en particular.
- La licencia no debe restringir otro hardware o software, es decir que no puede insistir en que otro hardware o software externos a los dispositivos sean también open-source.
- La licencia tiene que ser neutral, ninguna disposición de la misma debe basarse en una tecnología específica, parte o componente, material o interfaz para su uso.

Cumpliendo con estas premisas, los diseños y esquemas Arduino se distribuyen bajo licencia “Creative Commons Attribution-ShareAlike”.

Al seguir este tipo de licencias, donde toda la información es pública, todas las placas pueden ser construidas por uno mismo o bien comprarlas ya montadas. El software puede ser descargado de forma gratuita. Los ficheros de diseño de referencia (CAD), al estar disponibles bajo una licencia abierta, pueden ser libremente adaptados a las necesidades particulares.

11.4.4 ¿Por qué elegir Arduino?

Existen muchas otras placas de diferentes fabricantes que, aunque incorporan diferentes modelos de microcontroladores, son comparables y ofrecen una funcionalidad más o menos similar a la de las placas Arduino, como podrían ser Parallax, Basic Stamp, BX-24 de NETmedia, Phidgets o Handyboard de MIT por citar algunas.

Todas ellas también vienen acompañadas de un entorno de desarrollo agradable, cómodo y de un lenguaje de programación sencillo y completo.

No obstante la plataforma Arduino (Hardware + Software) ofrece una serie de ventajas:

- **Multiplataforma:** Se puede instalar y ejecutar en sistemas Windows, MAC OS y Linux. Esto no ocurre con el software de muchos otros microcontroladores. La mayoría de los entornos para microcontroladores están limitados a Windows.
- **Entorno de programación simple y directa:** El entorno de programación de Arduino, es fácil de usar para principiantes y lo suficientemente flexible para los usuarios avanzados. Además, existe mucha documentación con ejemplos detallados y gran cantidad de proyectos publicados en diferentes formatos.
- **Arduino tiene una gran comunidad:** Muchas personas lo utilizan, enriquecen la documentación y comparten continuamente sus ideas.
- **Hardware extensible y de código abierto:** Arduino está basado en los microcontroladores ATMEGA 168, 328 y 1280 y los planos de los módulos están publicados bajo licencia Creative commons por lo que, cualquiera que desee ampliar y mejorar tanto el diseño hardware de las placas como el entorno de desarrollo software y el propio lenguaje de programación, puede hacerlo sin problemas.
- **Software ampliable y de código abierto:** El software, al estar publicado bajo licencia libre, puede ser ampliado por programadores experimentados. El lenguaje puede ampliarse a través de librerías de C++, de igual modo que se puede añadir directamente código en C en los programas.
- **Asequible:** Las placas Arduino son más asequibles comparadas con otras plataformas de microcontroladores. La placa estándar, llamada Arduino UNO, preensamblada y lista para funcionar, cuesta alrededor de 20-25 euros y la versión más cara cuesta alrededor de 60 euros. Incluso, uno mismo se la podría construir adquiriendo los componentes por separado debido a que Arduino es de hardware libre.
- **Reutilizables y versátiles:** Reutilizables porque se puede aprovechar la misma placa para varios proyectos, ya que es muy fácil de desconectarla, reconectarla y reprogramarla, y versátiles porque contienen varios tipos de entradas y salidas de datos, las cuales permiten capturar información de sensores y enviar señales a actuadores de múltiples formas.

11.4.5 Nuestra elección: Arduino Uno Rev 3

Existe una gran variedad de placas Arduino, capaces de brindar al usuario la capacidad de programar de forma extremadamente versátil sus aplicaciones.

La finalidad del proyecto es, realizar el control de tracción de una motocicleta bimotor y se decidió realizar mediante la plataforma Arduino, por las múltiples ventajas que anteriormente se expusieron.

Teniendo en cuenta que las necesidades del proyecto no requiere de una gran potencia, ya que van a ser señales de mando, control y visualización, la principal opción es utilizar la placa de Arduino Uno Rev3, la cual teniendo una sencilla idea de los pines de I/O que se van a utilizar, con cuyas capacidades puede ser suficiente para los requeriremos de dicho proyecto. Podríamos haber seleccionado una placa algo superior, como es el Arduino mega que dispone de muchas más I/O, pero su precio era unos 10 euros más cara aproximadamente.

11.4.6 Especificaciones técnicas de Arduino Uno Rev 3

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current for I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328)
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Tabla 11 – Especificaciones técnicas de Arduino Uno

Arduino uno puede ser alimentado vía conexión USB o mediante una fuente de alimentación externa.

11.4.7 Alimentación, entradas y salidas de Arduino

Pines de alimentación (Power Pins)



Imagen 11.4.7:1 – Pines de alimentación

Ya sea alimentando Arduino mediante la conexión USB o mediante una fuente externa (recomendada de 7-12V), va a tener una serie de salidas de tensión continua debido a unos reguladores de tensión y condensadores de estabilización.

Estos pines son:

- VIN: Se trata de la fuente tensión de entrada, que contendrá la tensión a la que se esté alimentando al Arduino mediante la fuente externa.
- 5V: Fuente de tensión regulada de 5V, esta tensión puede venir ya sea del pin VIN a través de un regulador interno, o se suministra a través de USB o de otra fuente de 5V regulada.
- 3.3V: Fuente de 3.3 voltios generados por el regulador interno con un consumo máximo de corriente de 50mA.
- GND: Pines de tierra.
- IOREF: Este pin sirve para que la placa reconozca el tipo de alimentación que requieren los shields.
- Reset: Si en este pin se suministra un valor de tensión de 0 V se reinicia el microcontrolador.

Entradas y salidas digitales

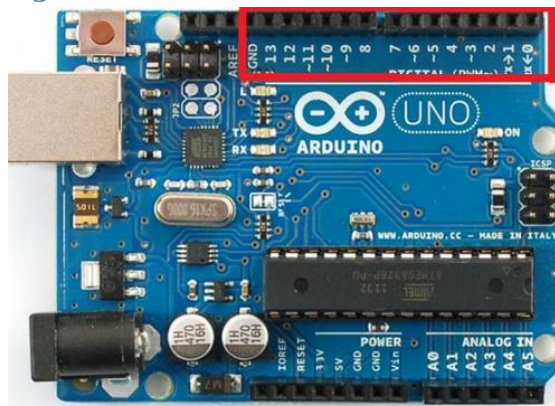


Imagen 11.4.7:2 – Pines de entrada y salida digitales

Cada uno de los 14 pines digitales, se puede utilizar como una entrada o salida. Cada pin, puede proporcionar o recibir un máximo de 40 mA y tiene una resistencia pull-up (desconectado por defecto) de 20 a 50 kOhm. Además, algunos pines tienen funciones especializadas como:

- Pin 0 (RX) y 1 (TX). Se utilizan para el recibimiento (RX) y la transmisión (TX) de datos serie TTL.
- Pin 2 y 3. Se utilizan para interrupciones externas. Se trata de pines encargados de interrumpir el programa secuencial establecido por el usuario.
- Pin 3, 5, 6, 9, 10 y 11. Se utilizan como señal PWM (modulación por ancho de pulso). Constituyen 8 bits de salida PWM con la función `analogWrite()`.
- Pin 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Estos pines son de apoyo a la comunicación SPI.
- Pin 13. LED. Hay un LED conectado al pin digital 13. Cuando la señal del pin es de valor alto, el LED se enciende, por el contrario cuando la señal del pin es de valor bajo, el LED se apaga.
- La versión Rev3 además tiene dos pines añadidos el SDA y el SCL.

Analog Inputs



Imagen 11.4.7:3 - Pines analógicos

Arduino posee 6 entradas analógicas, etiquetadas desde la A0 a A5, cada una de las cuales ofrecen 10 bits de resolución (es decir, 1024 estados). Por defecto, tenemos una tensión de 5V, pero podemos cambiar este rango utilizando el pin de AREF y utilizando la función `analogReference()`, donde se le introduce una señal externa de continua que la utilizará como referencia.

El microcontrolador ATmega 328, en las placas Arduino Uno, viene precargado con un gestor de arranque (boot loader que utiliza 0,5 KB de los 32 KB de la memoria flash), que permite cargar un nuevo código sin necesidad de un programador por hardware externo. La carga de un nuevo código, se realiza a través del entorno de desarrollo Arduino y la conexión serie USB)

También es posible saltar el gestor de arranque y programar directamente el microcontrolador a través del puerto ISCP. En tal caso, se debe utilizar un programador externo.

11.4.8 Esquema y pines

Los esquemas completos de Arduino Uno Rev 3 se pueden encontrar en su página web oficial.

En la siguiente imagen se puede observar la correspondencia de pines entre el ATmega/328 y Arduino.

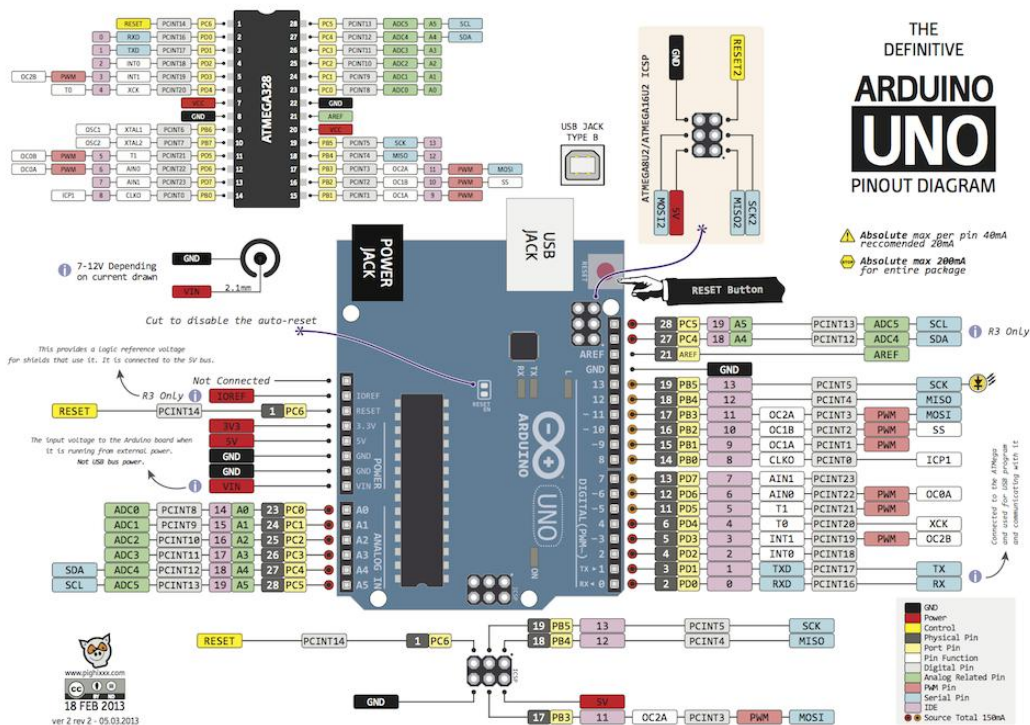


Imagen 11.4.8:1 - Relación de los pines con el microcontrolador (1)

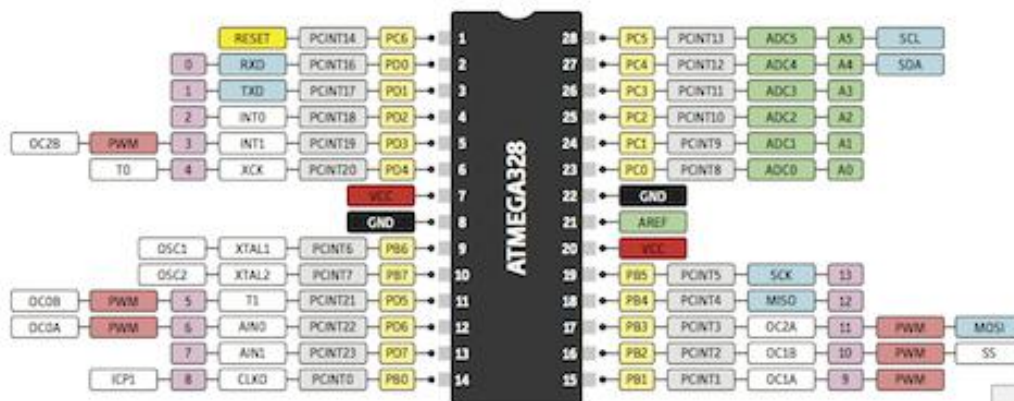
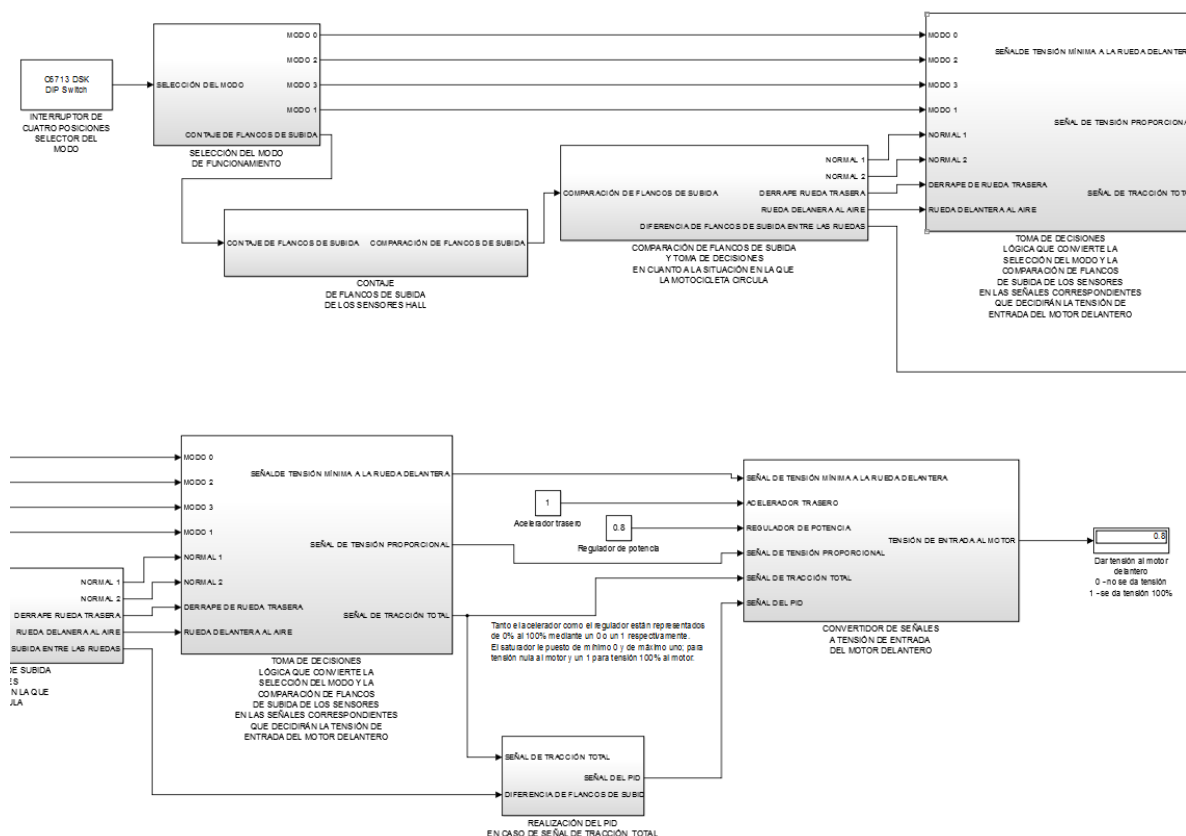


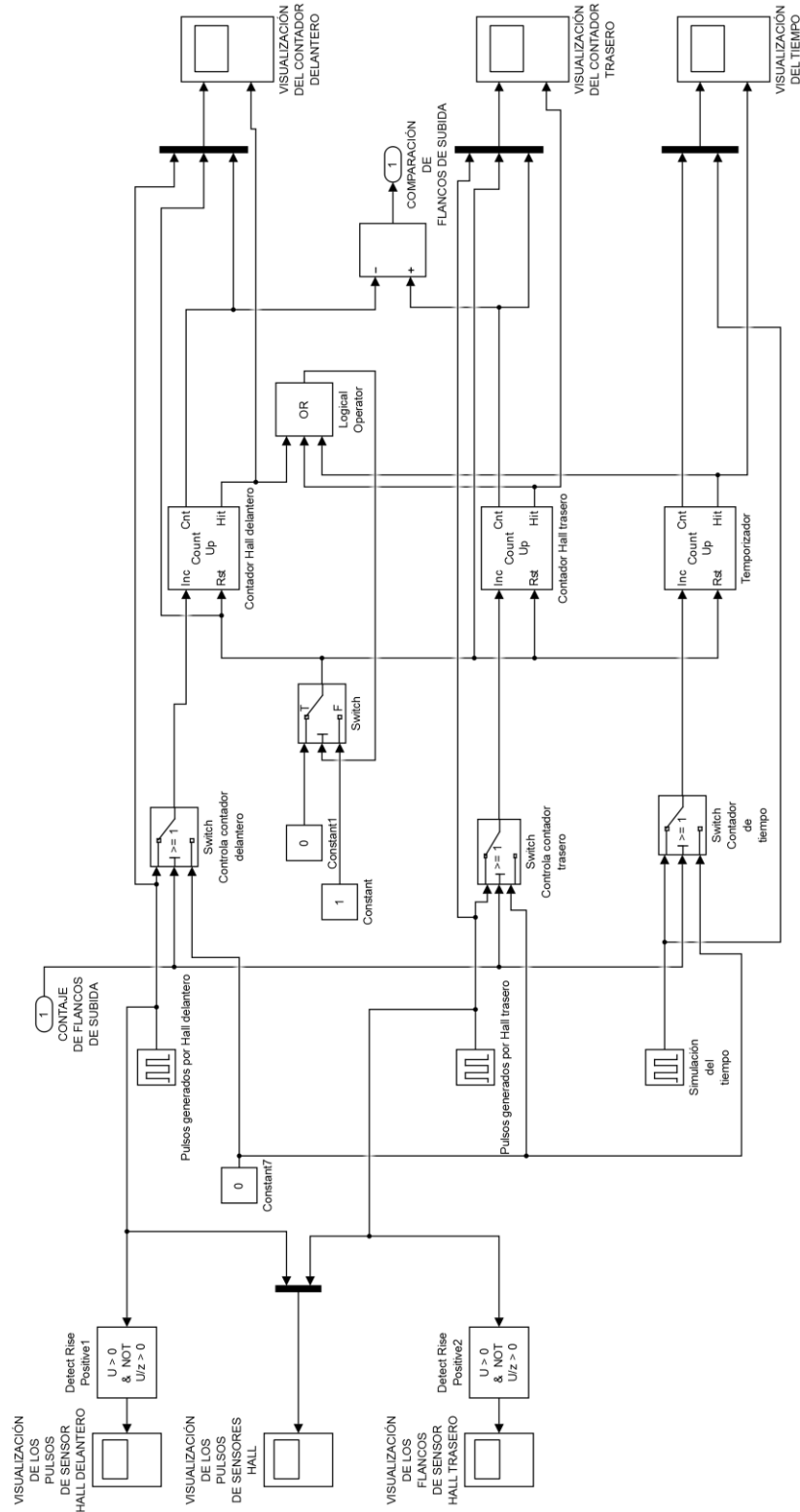
Imagen 11.4.8:2 - Relación de los pines con el microcontrolador (2)

11.5 Simulación mediante Simulink

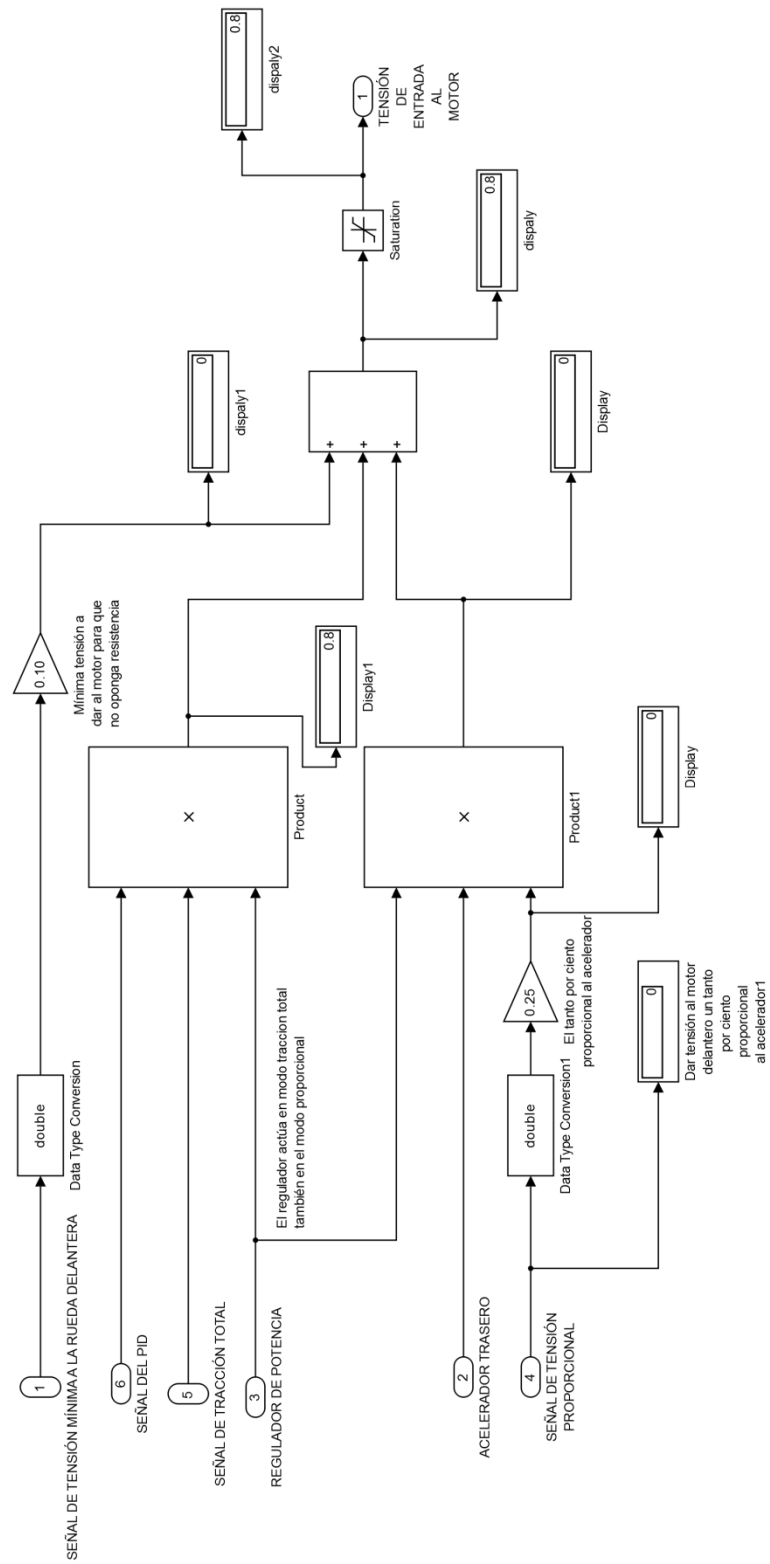
El modelo, con el que se va a realizar la simulación mediante Simulink, va a contener las mismas rutinas en las que se dividió el problema y en cada una de ellas se podrá observar su funcionamiento independientemente de las demás.



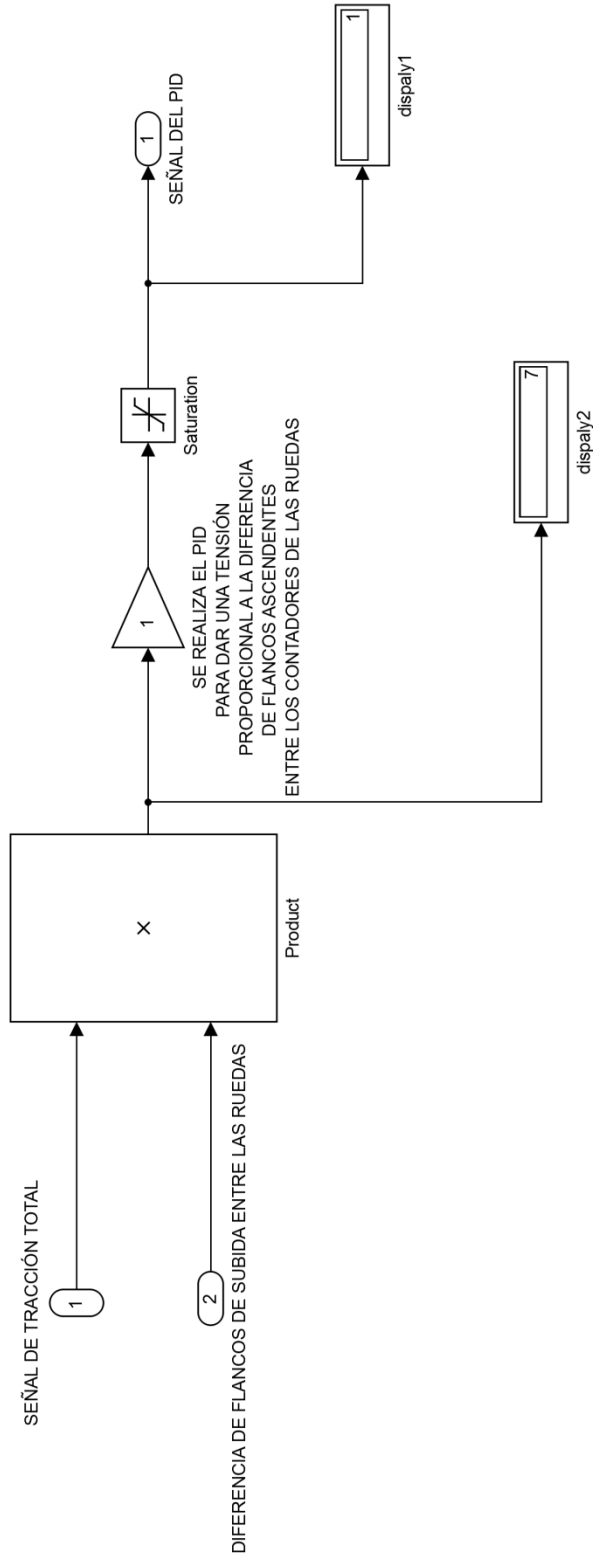
Contaje de flancos de subida por los sensores de cada rueda.



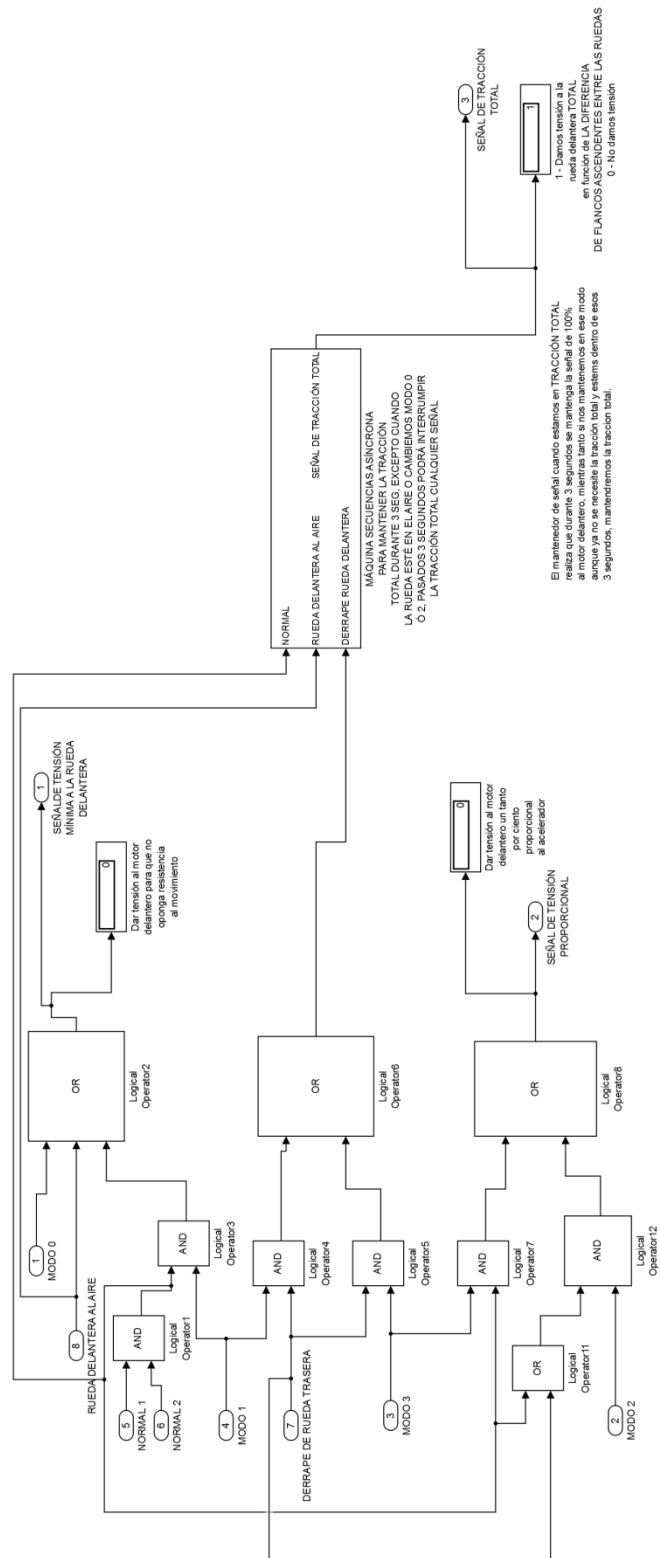
Señal de salida comandada.



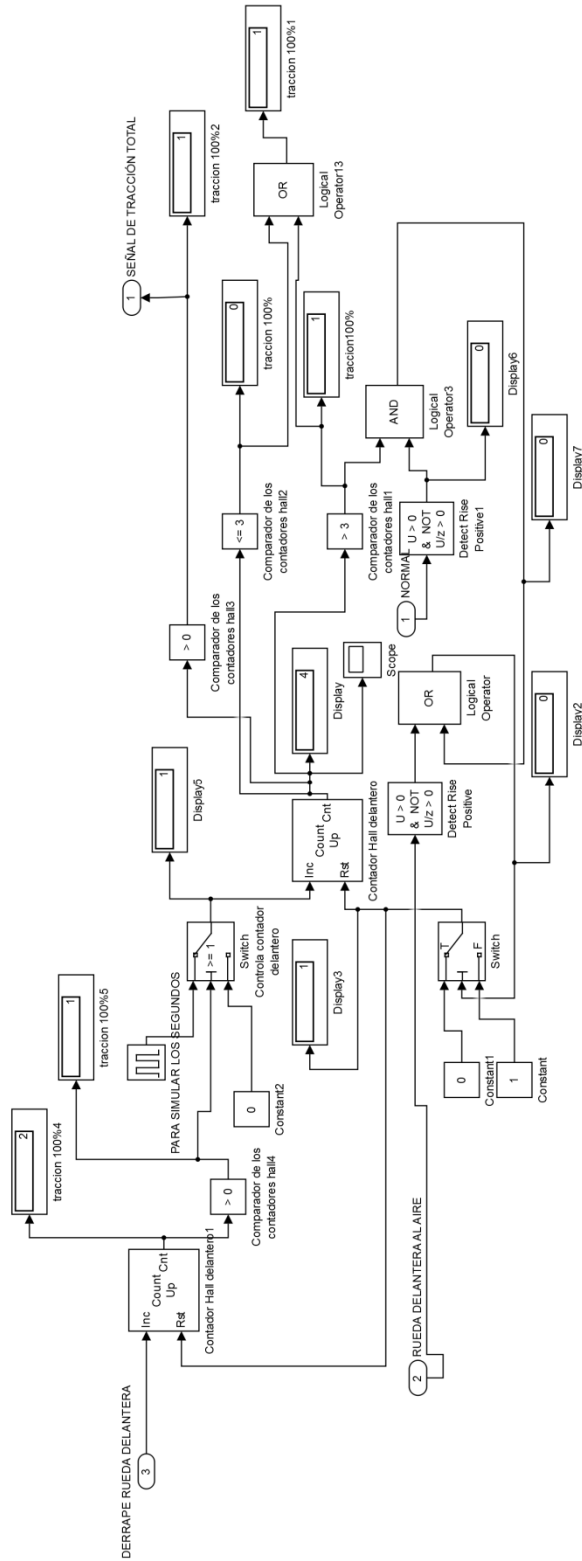
Señal de salida del PID.



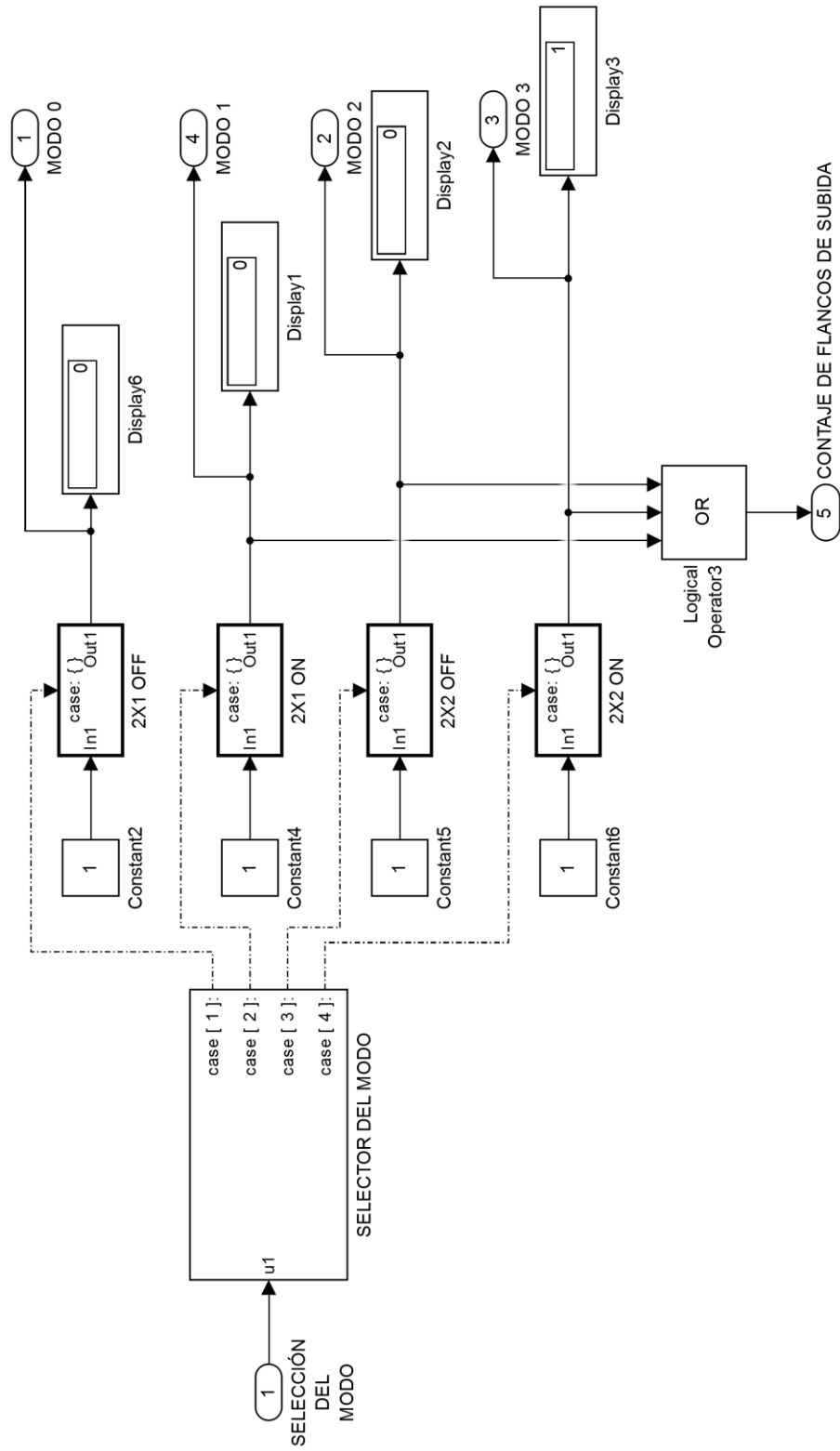
Toma de decisiones de la puesta en marcha o no del motor delantero.



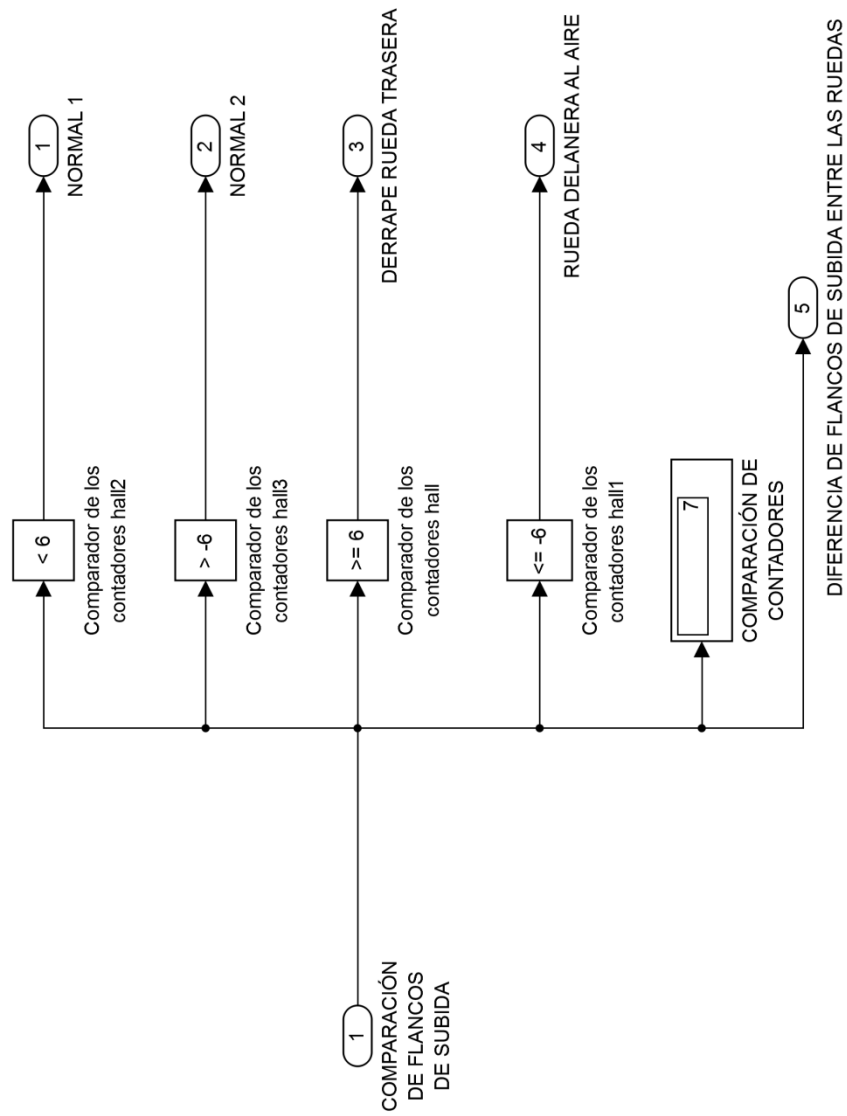
Máquina secuencial síncrona para la señal de tracción total.



Selección del modo con un switch.



Comparación de velocidades (flancos de subida contabilizados por los sensores)



El diseño del modelo se ha llevado a cabo mediante las rutinas en las que se dividió el problema, a partir de las especificaciones del diseño y de las distintas acciones a realizar, en función de las diferentes entradas que tiene el proceso.

En primer lugar nos encontramos con la rutina de selección del modo, con el cual se seleccionará el modo de funcionamiento del sistema. La selección de modo se podrá realizar mediante un interruptor de 4 posiciones, pero en Simulink se ha implementado mediante un **DIP Switch C6713 DSK**, con el cual se puede seleccionar hasta quince posiciones. De las quince posiciones se van a utilizar cuatro, una para cada modo. El resto de posiciones quedarán sin utilizar o como posiciones de reserva:

0001	MODO 0 ó MODO 2X1 OFF
0010	MODO 1 ó MODO 2X1 ON
0011	MODO 2 ó MODO 2X2 OFF
0100	MODO 3 ó MODO 2X2 ON
0101	-
0110	-
0111	-
1000	-
1001	-
1010	-
1011	-
1100	-
1101	-
1110	-
1111	-

Tabla 12 – Codificación del DIP Switch C6713 DSK

En el **bloque selector de modo** se puede observar que el modo 0, es el único que no inicia el conteo de flancos de subida de los sensores hall, para comparar la diferencia de flancos y tomar una decisión. Ello es debido a que el modo 0 siempre funciona de la misma forma, en este modo sólo entra en funcionamiento la tracción trasera. Como estamos utilizando motores eléctricos brushless para que el funcionamiento sea idéntico al de una motocicleta estándar se tendrá que proporcionar una tensión mínima al motor delantero para que los imanes que contiene el motor no ofrezca resistencia al movimiento, pero esta tensión mínima que se proporciona al motor delantero no se puede considerar como que la tracción delantera entre en funcionamiento ya que el propósito es que la rueda gire libremente sin la oposición que proporcionan los imanes internos del motor brushless cuando no está alimentado.

Sin embargo, los otros tres modos (modo 1, modo 2, modo 3) inician el conteo de flancos de subida del sensor hall de cada rueda, para comprobar

en qué situación de funcionamiento nos encontramos, y a partir de ahí tomar una decisión sobre la salida a controlar (la tensión del motor delantero).

El contaje de flancos de subida en Simulink se va a simular mediante tres generadores de onda, uno para simular los pulsos generados por el sensor de efecto hall de la rueda delantera, otro para simular los pulsos generados por el sensor de efecto hall de la rueda trasera y otro para simular el tiempo que transcurre, para determinar posteriormente si la toma de decisiones se va a realizar por tiempo transcurrido o por número de flancos de subida máximo contabilizados. En esta simulación cada segundo se producirá un pulso.

La señal de cada generador de onda se llevará a un contador de flancos de subida y cada contador de flancos, se reiniciará en función de un número máximo de flancos de subida.

En el caso de los generadores de onda de los sensores hall, su límite de flancos que reinicia el contaje es 48 y para el generador de onda que simula el tiempo, el límite será un segundo pues queremos que como máximo se encuentre realizando el contaje cada segundo.

Si uno de los contadores llega a su máximo, reseteará todos los contadores y se producirá la comparación del número de flancos de subida, contabilizado por los dos sensores de efecto hall. Por lo tanto se producirá la comparación de los contadores de dos formas distintas:

- 1) Cuando uno de los contadores de los sensores llegue al número de flancos positivos máximo.
- 2) Ó cuando se exceda un tiempo máximo determinado de conteo.

A continuación se realiza la **toma de decisiones ante la comparación de flancos de subida**. Si la diferencia de flancos entre la rueda trasera y la rueda delantera, es mayor o igual que doce, se decide que la motocicleta se encontrará en una situación de pérdida de tracción en la rueda trasera. Si la diferencia de flancos entre la rueda delantera y la rueda trasera, es mayor o igual que doce se decide que la motocicleta se encontrará en una situación de que la rueda delantera está en el aire. Si la diferencia de flancos entre la rueda trasera y la rueda delantera, es menor que doce y mayor que menos doce se decide que la motocicleta se encontrará en una situación de funcionamiento normal.

Una vez que el control de tracción ha tomado la decisión, de en qué situación se encuentra funcionando la motocicleta en ese instante, lo siguiente a determinar por este sistema, es la acción que debe llevar a cabo sobre el motor delantero en función de:

- El modo seleccionado por el piloto.
- La decisión de la situación de funcionamiento de la moto.

Esta acción, que determinará la forma de actuación sobre el motor delantero, se ha implementado en Simulink mediante el bloque: Bloque que contiene la lógica de la toma de decisiones.

Bloque que contiene la lógica de las tomas de decisiones: Con este bloque de lógica, se determinará la acción a llevar a cabo sobre el motor delantero, la cual va a depender del modo de funcionamiento seleccionado, que se elegirá antes de comenzar su funcionamiento, y de la decisión de la situación de la motocicleta en ese instante tomada por el mismo sistema.

Si el modo de funcionamiento cambia a lo largo del mismo, hasta la siguiente comparación de velocidades este cambio no se tendrá en cuenta. Si se desea que la acción sobre el switch selector de modo sea más inmediata, además tendremos que añadir a mayores condiciones en el bloque de lógica y de la máquina de secuencia.

La diversidad de las acciones sobre el motor delantero son las siguientes:

- Señal de tensión mínima a la rueda delantera, la cual podrá darse en los siguientes casos:
 - Siempre que se tenga seleccionado el MODO 0.
 - Cuando el sistema, por medio de la comparación de flancos de subida de los sensores, detecte que la motocicleta se encuentra en una situación de que la rueda delantera se encuentre en el aire (número de flancos del contador delantero sea superior al número de flancos del contador trasero en un número de pulsos de diferencia específico), sea cual sea el modo de funcionamiento seleccionado del control de tracción.
 - Siempre que se tenga seleccionado el MODO 1 y el sistema detecte que la motocicleta se encuentra en una situación de funcionamiento normal.
- Señal de tensión proporcional al acelerador, la cual podrá darse en los siguientes casos:
 - Siempre que se tenga seleccionado el MODO 2 y el sistema no detecte que la motocicleta se encuentre en una situación de que la rueda esté en el aire.
 - Cuando se tenga seleccionado el MODO 3 y el sistema detecte que la motocicleta se encuentra en una situación de funcionamiento normal.
- Señal de tensión de tracción total, la cual podrá darse en los siguientes casos:

- Siempre que se tenga seleccionado el modo 1, o el modo 3 y además el sistema detecte que la motocicleta se encuentra en una situación de pérdida de tracción en la rueda trasera.

La señal de tensión de tracción total, se deseará mantener durante 3 segundos, para evitar continuos cambios en la señal comandada al motor delantero cuando se produzca pérdida de tracción, debido a posibles variaciones de velocidad. Por ello se decide mantener 3 segundos esa señal, hasta que se estabilice la velocidad de la rueda delantera en caso de pérdida de tracción y así salvaguardar, correctamente, la situación de peligro (prevaleciendo la situación de seguridad del piloto).

La señal de tensión, de tracción total, se comandará realizando previamente el diseño de un control PID, para proporcionar una tensión, al motor delantero, proporcional a la diferencia de velocidades.

Para poder diseñar y simular la situación de pérdida de tracción en la rueda trasera, con el accionamiento del sistema de tracción 2x2 y la posibilidad de que la rueda delantera gire loca, como consecuencia de estar la moto en el aire, se ha realizado el estudio de esta situación mediante el diseño de un circuito secuencial síncrono. Este estudio se explica en el Anexo “Diseño de circuito secuencial síncrono”, para el mantenimiento de la señal de tracción de la rueda delantera.

Para simular la máquina secuencial síncrona mediante Simulink, se realizó el bloque de máquina secuencial síncrona, en el cual su función primordial es mantener la señal de tensión de tracción total durante 3 segundos, a excepción de, si en ese periodo de tiempo el sistema detecta que la motocicleta se encuentra en una situación de rueda delantera en el aire, entonces en este caso la señal de tensión de tracción no se mantendrá 3 segundos y cambiará inmediatamente por motivos meramente de seguridad y evitar accidentes.

¿Por qué en esa situación se interrumpirá la señal de tensión de tracción total?

Esto es debido a que prevalece la acción de señal de tensión mínima, a la de rueda delantera en el aire, por medio de la detección del sistema de que la rueda delantera gira mucho más deprisa que la trasera. Porque si la motocicleta se encuentra en el aire y gira loca, si girara muy deprisa y la moto se apoyara primero sobre la rueda delantera podría dar lugar a una caída.

Pasados esos 3 segundos podrá interrumpir la señal de tensión de tracción total cualquier señal.

Por último, una vez decidida la acción a tomar sobre el motor delantero, se comandará la señal de tensión que se proporcionará al motor delantero en función de la acción a tomar sobre el motor, del accionamiento del acelerador y del regulador de potencia de la tracción delantera. Esto se realizará en la simulación por medio del bloque señal de salida comandada. El máximo y el mínimo nivel de tensión, que será proporcionado al motor delantero, es representado en la simulación por medio de 1 y 0 respectivamente.

11.6 Diseño del circuito secuencial síncrono para el mantenimiento de la señal de tracción total de la rueda delantera

A la hora de realizar este diseño, del circuito secuencial síncrono, se necesitará de un cronograma tipo, con las distintas decisiones del control de tracción y la acción a comandar por este sistema, en función de esa decisión tomada.

Leyenda del cronograma:

La señal T, es la de señal de situación de pérdida de tracción en la rueda trasera que acciona la tracción delantera S durante 3 segundos, a excepción de que se produzca la señal R, que es la señal de situación de que la motocicleta tiene la rueda delantera en el aire.

Cronograma:

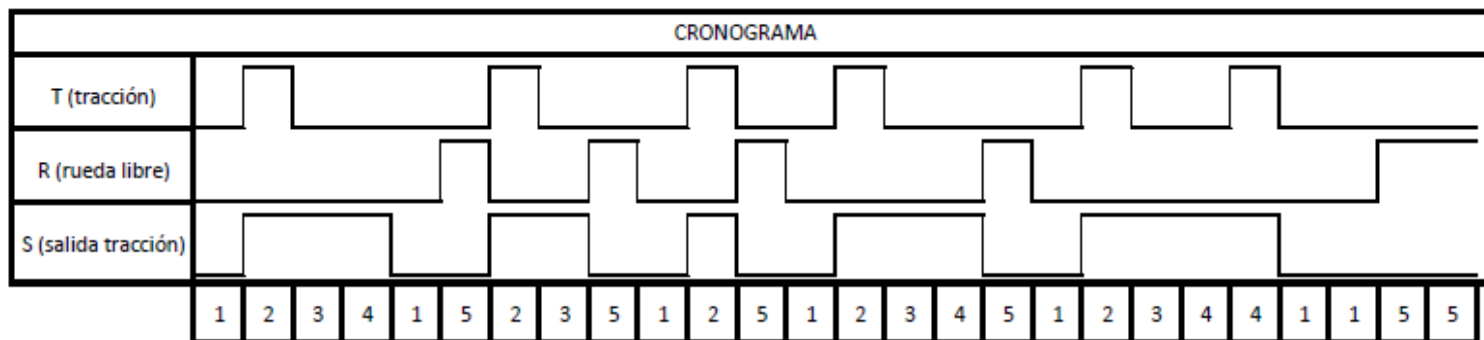


Ilustración XV - Cronograma

A partir del cronograma anterior, se van a desarrollar los diagramas de flujo correspondientes, para estudiar el caso anteriormente mencionado y poder determinar la correcta actuación del sistema ante dicha situación.

En los diagramas de flujo posteriormente representados, se va a representar las señales del cronograma anterior con la siguiente simbología: TR/S y en cuyo lugar ocuparan los números, 0 cuando esa señal no está activa y 1 cuando sí lo está.

Diagrama de flujo

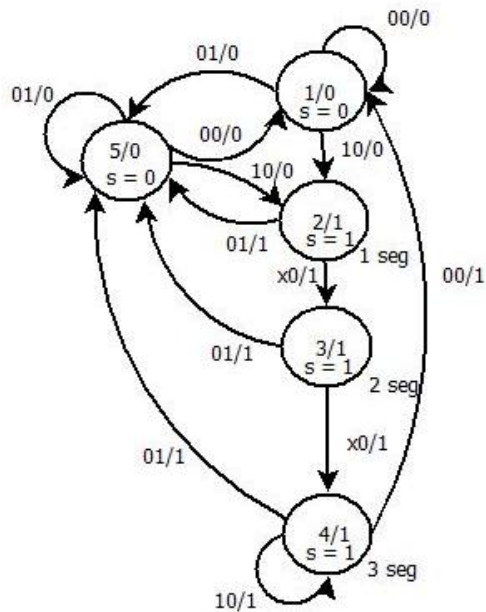


Ilustración XVI – Diagrama de flujo

Tabla de fases obtenida del diagrama de flujo

EA	00	01	11	10
1	① /0	5/0	-	2/0
2	3/1	5/1	-	3/1
3	4/1	5/1	-	4/1
4	1/1	5/1	-	⑤ /1
5	1/0	⑤ /0	-	2/0

Tabla 13 – Tabla de fases

Tabla reducida de fases

En este caso la tabla reducida de fases coincide con la tabla de fases.

Tabla de fusión

EA	00	01	11	10
1	①/0	①/0	-	2/0
2	3/1	1/1	-	3/1
3	4/1	1/1	-	4/1
4	1/0	1/0	-	④/1

Tabla 14 – Tabla de fusión

A partir de la tabla de fusión podemos obtener el diagrama de flujo final:

Diagrama de flujo final

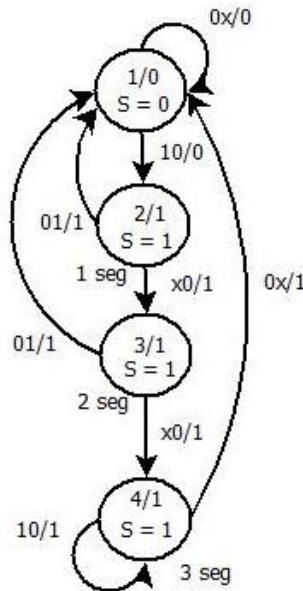
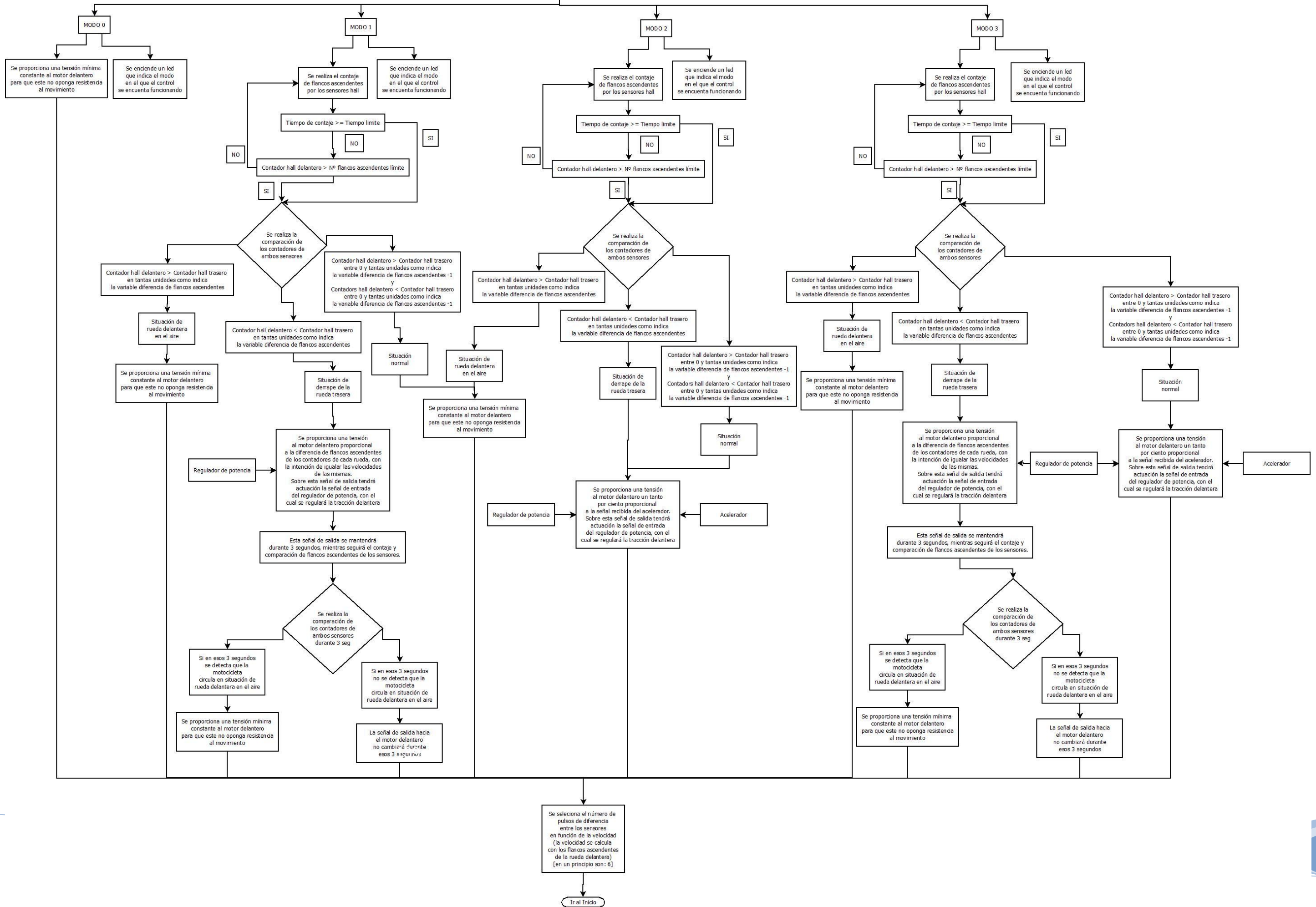


Ilustración XVII – Diagrama de flujo final

Con la realización del diseño secuencial síncrono, podemos observar que al motor de la rueda delantera, en caso de que el control de tracción decida que la motocicleta se encuentra en situación de pérdida de tracción y esté seleccionado el modo correspondiente se accionará el motor delantero hasta que se corrija el error pudiendo llegar al 100% de la limitación prefijada por el piloto, es decir, al motor delantero se le proporcionará una tensión hasta que estas giren aproximadamente a la misma velocidad. Esa acción se mantendrá durante tres segundos como mínimo, a excepción de que el sistema detecte que la rueda delantera gira más rápido que la trasera, en cuyo lugar la tensión que se le proporcionará al motor delantero es una tensión mínima, como ya se explicó anteriormente.

Si pasados esos tres segundos, nuestro sistema de control detecta que continuamos en situación de pérdida de tracción en la rueda trasera, se seguirá dando a la rueda delantera la tensión necesaria para igualar las velocidades de las ruedas, hasta que el control detecte que estamos en situación de funcionamiento normal o que la rueda delantera esté en el aire, en ese caso, cambiará inmediatamente la forma de actuación en función de la decisión que haya tomado.



11.8 Diseño de la estructura principal

Lo primero a construir es un soporte, con el cual las ruedas queden totalmente en el aire y puedan así girar libremente

A la hora de realizar el soporte o estructura, para el posterior desarrollo de las pruebas, se decidió reutilizar un palet de madera y posteriormente darle una capa de pintura.

El diseño básico de la estructura en AutoCad es el siguiente:

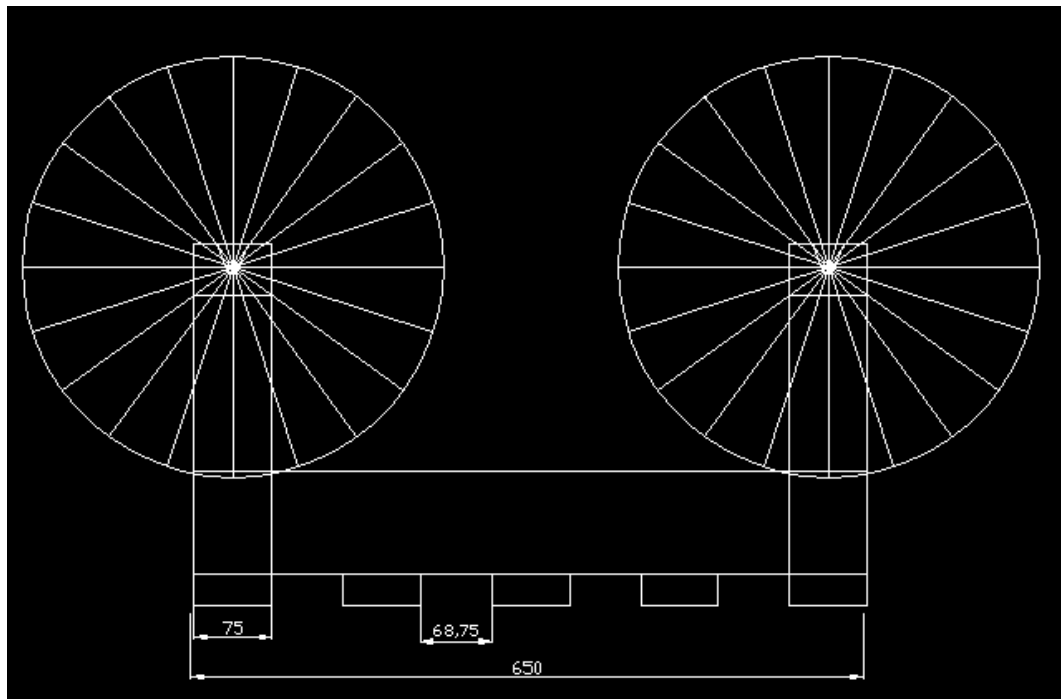


Imagen 11.4.8:1 – Alzado de la estructura

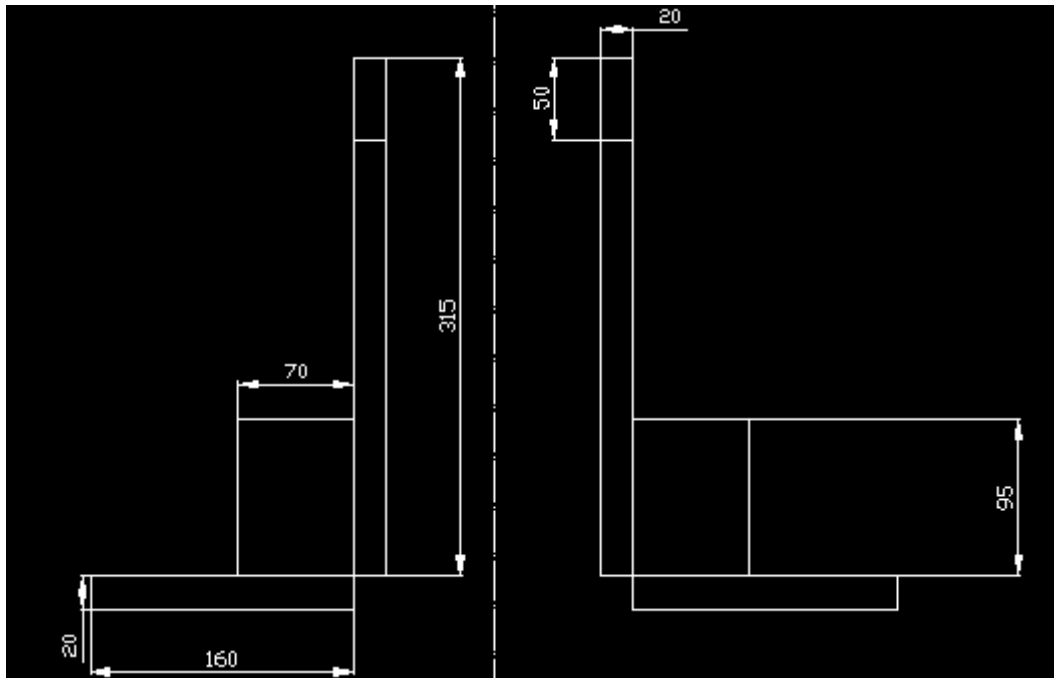


Imagen 11.4.8:2 – Perfil de la estructura

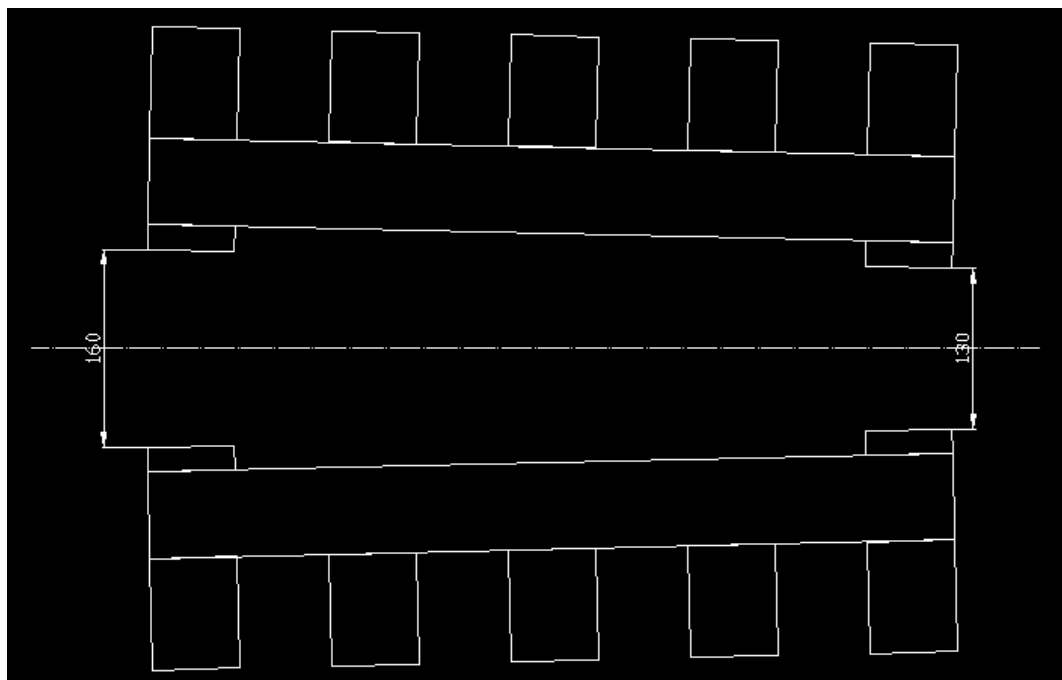


Imagen 11.4.8:3 – Planta de la estructura

Como se puede apreciar hay una diferencia, de distancia, entre los puntos de apoyo de la rueda delantera y de la rueda trasera. Ello es debido a que el eje de la rueda delantera tiene una menor longitud que el de la rueda trasera y el hub-motor es menos ancho que el motor de la rueda trasera.

Además en esa distancia, entre los apoyos del eje de la rueda, debe tenerse en cuenta que deberá introducirse la rueda con el hub motor, el imán y el

sensor hall a una cierta distancia (unos 5 mm) para detectar correctamente el campo magnético creado por los imanes.

Una vez realizado el diseño de la estructura, se lleva a cabo la construcción de la misma:



Imagen 11.4.8:4 – Construcción de la estructura

Se añadieron dos travesaños, tanto para fijar la distancia entre los apoyos como para dotar de más robustez y consistencia a la estructura.

Posteriormente se llevo a cabo el pintado de la estructura:



Imagen 11.4.8:5 – Estructura pintada

Y el resultado final ya le habéis visto anteriormente:



Imagen 11.4.8:6 - Estructura final con el montaje de los componentes

11.9 Diseño de la colocación de los imanes en las ruedas

La colocación del conjunto de 16 imanes, se va a realizar en el eje de cada rueda. Para lo cual, se decidió unir el conjunto de imanes al motor mediante la realización de una especie de “jaula”.

Cada motor tiene una tapa, con sus respectivos tornillos para la sujeción de esta al motor. Esa tapa, junto con los tornillos de sujeción, se aprovechó para, junto la construcción de una tapa circular del mismo diámetro que los imanes realizada con plástico lexan, crear una jaula que consiga la unión del imán a la rueda y así gire solidariamente a ella.

El diseño de la tapa de plástico lexan es el siguiente:

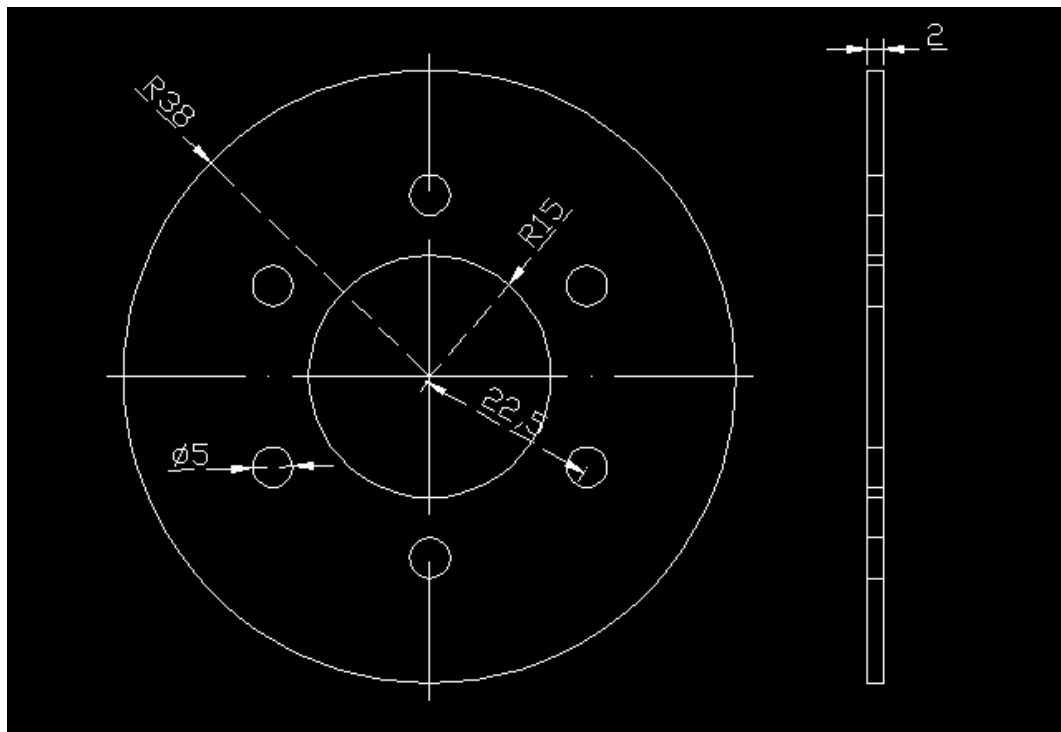


Imagen 11.4.8:1 – Tapa Lexan para realizar la jaula de los imanes

Y el resultado final se puede apreciar a continuación:

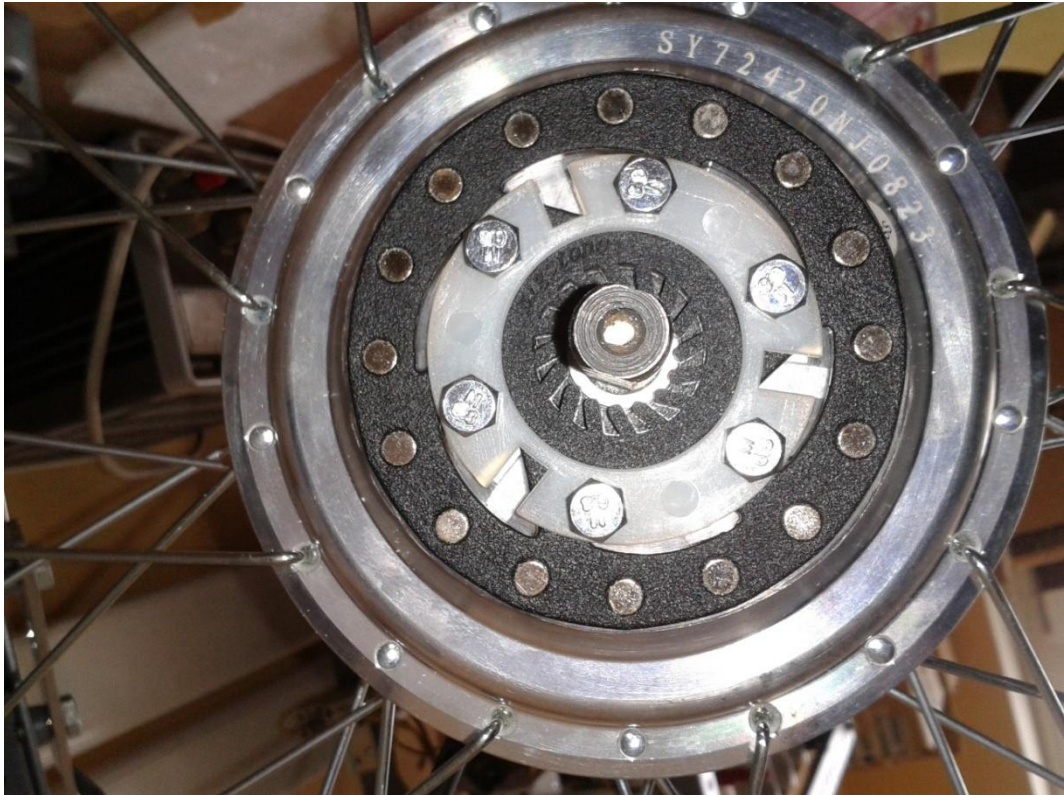


Imagen 11.4.8:2 - Montaje final de la jaula de imanes en el eje de la rueda



Imagen 11.4.8:3 - Alzado de la planta de imanes en el eje de la rueda

11.10 Colocación de los sensores

Como ya se ha mencionado anteriormente, los sensores tienen que estar enfrentados a los imanes a una cierta distancia, para poder captar su campo magnético y así poder transmitir las señales correspondientes. Por ello los sensores hall deben colocarse en el eje de las ruedas.

Los sensores hall, vienen diseñados con un soporte del diámetro exacto para la colocación del mismo en el eje de los pedales, que es donde suele ir colocado.

Por lo tanto, se pensó en realizar un nuevo soporte, mediante plástico lexan, con el cual podamos colocar dicho sensor, en el eje de la rueda, enfrentándolo al conjunto de imanes.

Para ello se realizó el diseño del soporte del sensor hall:

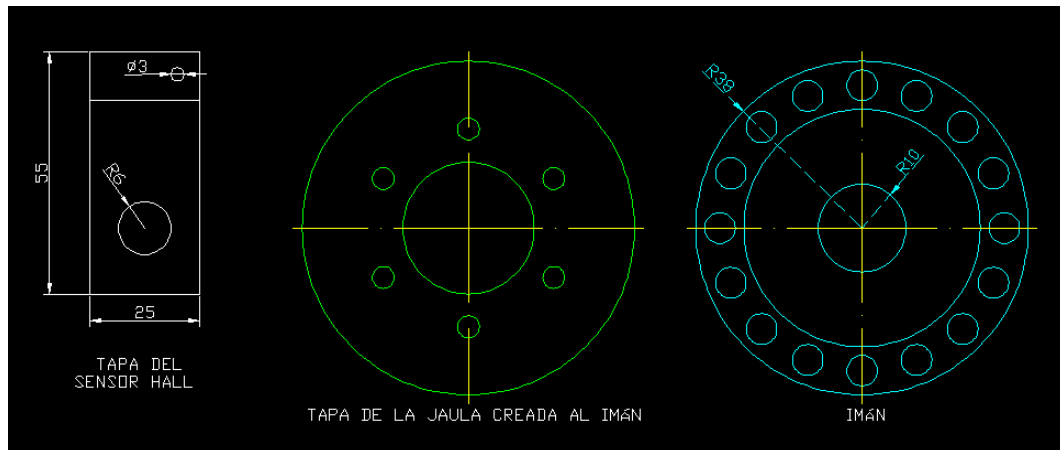


Ilustración XVIII – Soportes realizados para los sensores Hall

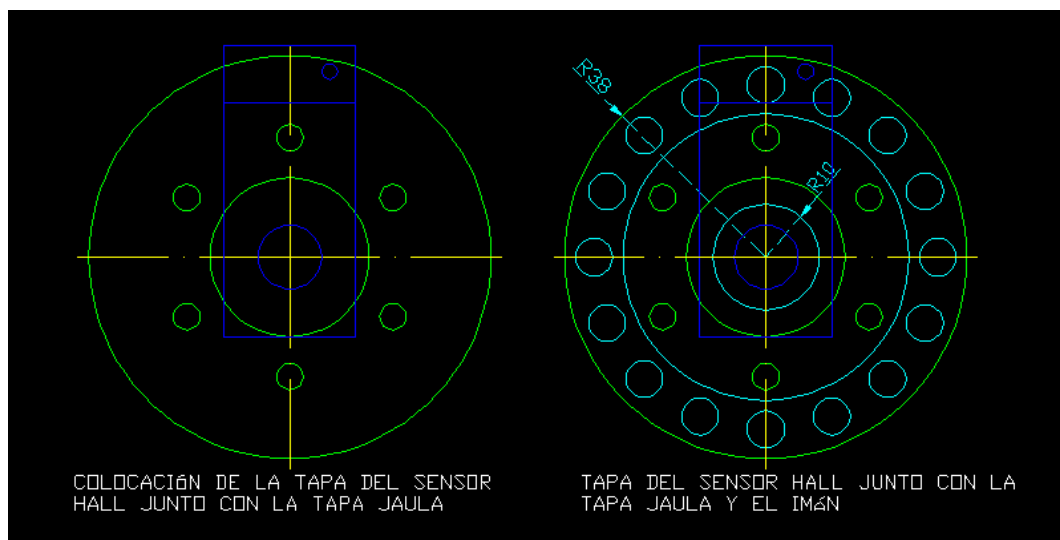


Ilustración XIX – Enfrentamiento de los sensores Hall respecto a los imanes

Y así es como queda implantando en el eje de la rueda:



Imagen 11.4.8:1 - Montaje de los sensores Hall en el eje de las ruedas



Imagen 11.4.8:2 - Visualización del montaje de las ruedas en el soporte

11.11 Código del programa

```
//-- LIBRERÍAS UTILIZADAS --//
#include <avr/interrupt.h> // Librería necesaria para el uso de las interrupciones
#include <avr/io.h> // Librería necesaria para la definición de entradas y salidas y para poder
realizar el cambio de pin de la interrupción
// ----- //

//--ASIGNACIÓN DE LOS PINES UTILIZADOS--//
const int HALL_DELANTERO = 2; //Sensor de efecto hall delantero, INTERRUPCIÓN 0
const int HALL TRASERO = 17; //Sensor de efecto hall trasero EN EL PIN A3, INTERRUPCIÓN
1
const int controlador_delantero = 5; //Pin al que se conectará el controlador delantero
const int BOTON_2X1_OFF = 12; // BOTON QUE ACCIONA EL MODO 0
const int BOTON_2X1_ON = 11; // BOTON QUE ACCIONA EL MODO 1
const int BOTON_2X2_OFF = 9; // BOTON QUE ACCIONA EL MODO 2
const int BOTON_2X2_ON = 8; // BOTON QUE ACCIONA EL MODO 3
//const int LED_0 = 7; //LED QUE INDICA QUE ESTAMOS EN MODO 0 // COMENTADO
//const int LED_1 = 6; //LED QUE INDICA QUE ESTAMOS EN MODO 1// COMENTADO
//const int LED_2 = 4; //LED QUE INDICA QUE ESTAMOS EN MODO 2// COMENTADO
//const int LED_3 = 3; //LED QUE INDICA QUE ESTAMOS EN MODO 3// COMENTADO
//const int Regulador_de_potencia = 0; //Pin del regulador es el A0 (Analog IN), COMENTADO
const int Acelerador = 5; // Pin del acelerador es el A5(Analog in)
//-----//

//--VARIABLES UTILIZADAS--//

int FLAG_0 = 0; // Bandera para solo realizar una vez ciertas acciones cuando el MODO_0
está activado o desactivado
int FLAG_1 = 0; // Bandera para solo realizar una vez ciertas acciones cuando el MODO_1
está activado o desactivado
int FLAG_2 = 0; // Bandera para solo realizar una vez ciertas acciones cuando el MODO_2
está activado o desactivado
int FLAG_3 = 0; // Bandera para solo realizar una vez ciertas acciones cuando el MODO_3
está activado o desactivado

int MODO_0 = 0; //CUANDO MODO_0 = 1, NOS ENCONTRAMOS EN EL MODO_0 --> 2X1 OFF
int MODO_1 = 0; //CUANDO MODO_1 = 1, NOS ENCONTRAMOS EN EL MODO_1 --> 2X1 ON
int MODO_2 = 0; //CUANDO MODO_2 = 1, NOS ENCONTRAMOS EN EL MODO_2 --> 2X2 OFF
int MODO_3 = 0; //CUANDO MODO_3 = 1, NOS ENCONTRAMOS EN EL MODO_3 --> 2X2 ON

int PRIMERA_VEZ_MODO_0 = 0; // Variable que indica que el MODO_0 a pasado de inactivo
a activo
int UNA_VEZ = 0; // Variable para mantener durante 3 segundos la señal de Tracción Total
int BLOQUEO = 0; // Variable con la que en Funcionamiento Tracción Total, se indica que si
en 3 segundos se produce funcionamiento normal se ignora.

int valor_regulador = 0; // Variable en la que se almacena el valor del regulador de potencia
double valor_nuevo = 1.0; // Variable con la que indicaremos el valor de nuestro regulador
ente 0 y 1 con escalones de 0.1
int valor_acelerador = 0; // Variable en la que se almacena el valor del acelerador
```

```

int valor_acelerador_traducido = 0; // Variable en la que se almacena el valor de tensión que
se deberá de mandar al controlador delantero
double valor_acelerador_traducido_float = 0.0; // Variable en la que se almacena el valor de
tensión que se deberá de mandar al controlador delantero pero del tipo coma flotante
double proporcion_acelerador = 0.8; //Este valor sería ajustable por el piloto vía software. Es
el valor por el cual se multiplica la señal en modo Tracción Proporcional

int diferencia = 0; // variable en la que guardamos la diferencia entre los contadores de
flancos ascendentes entre la rueda trasera y la rueda delantera
double diferencia_filtrada = 0.0; // Variable en la que guardamos la diferencia entre los
contadores de los flancos ascendentes de la rueda trasera y los de la rueda delantera,
depués del filtro
const long tiempo_calculo_diferencia = 500000; // Tiempo en el que se cuenta los flancos
positivos para después realizar la comparación de ellos. LA COMPARACIÓN DE FLANCOS SE
REALIZA CADA 0.5 SEGUNDOS
long tiempo_diferente = 0; // Variable en la que guardamos el tiempo en que se tarda en
entrar de nuevo en el bucle de comparación de velocidades desde la última vez
long tiempo_anterior = 0; // Tiempo en el que se realizó el último bucle
unsigned long tiempo_actual = 0; // Tiempo actual del programa

volatile int CONTADOR_TRASERO; // Contador de la rueda trasera en el que se contabiliza el
número de flancos ascendentes detectados en la señal que manda el sensor hall trasero.
volatile int CONTADOR_DELANTERO; // Contador de la rueda delantera en el que se
contabiliza el número de flancos ascendentes detectados en la señal que manda el sensor
hall delantero.

int PULSOS_LIMITE = 12; // PULSOS LÍMITE CON LOS QUE PARTIMOS INICIALMENTE HASTA
LOS CUALES CONTARAN LOS CONTADORES SI NO SE SUPERA EL MEDIO SEGUNDO, ESTE SE
MANTIENE CONSTANTE
int diferencia_pulsos_limite = 4; // DIFERENCIA DE FLANCOS ASCENDENTES CON LA CUAL SE
TOMARÁN LAS DECISIONES.
volatile int velocidad_rueda_trasera = 0; // Aquí se guarda el contaje de flancos ascendentes
de la rueda trasera
volatile int velocidad_rueda_delantera = 0; // Aquí se guarda el contaje de flancos
ascendentes de la rueda delantera
double velocidad_rueda_trasera_final = 0.0; // Contaje de flancos ascendentes de la rueda
trasera filtrado
double velocidad_rueda_delantera_final = 0.0; // Contaje de flancos ascendentes de la
rueda delantera filtrado
double velocidad_rueda_trasera_final_anterior = 0.0; // Cantidad de flancos ascendentes de
la rueda trasera filtrada contabilizados en la anterior comparación de velocidades
double velocidad_rueda_delantera_final_anterior = 0.0; // Cantidad de flancos ascendentes
de la rueda delantera filtrada contabilizados en la anterior comparación de velocidades

float pi = 3.1416; // Número PI
float DIAMETRO_RUEDA = 0.4064; // Diámetro de la rueda
float velocidad_RD_float = 0.0; // SEÑAL A COMANDAR AL MOTOR DELANTERO EN COMA
FLOTANTE
int velocidad_RD_int = 0; // SEÑAL A COMANDAR AL MOTOR DELANTERO (ENTERO)

```



```
unsigned long tiempo_anterior_PID = 0.0; // Tiempo en el que finalizó el último control
realizado por el PID
unsigned long Ahora_PID = 0.0; // tiempo en el que se está volviendo a ejecutar el nuevo
control del PID
volatile double Error = 0.0; // Diferencia de flancos ascendentes filtrados entre la rueda
trasera y delantera
volatile double salida = 0.0, errorDer = 0.0, Error_anterior = 0.0, errorsum = 0.0, errorProp
=0.0; // Definición de distintas variables para la realización el PID
double kp = 0.05,ki = 0.35, kd = 10000; // Término proporcional, integral y derivativo
double dT = 0.0; // Tiempo que tarda en volverse a ejecutar el control
double dT_distinto = 0.0; // Tiempo que tarda en volverse a ejecutar la comparación de
flancos ascendentes para los filtros de las señales de los sensores Hall
```

```
unsigned long tiempo_anterior_mantener_traccion = 0.0; // tiempo anterior para mantener la
señal al motor delantero en caso de pérdida de tracción de la rueda trasera
unsigned long tiempo_mantener_traccion = 0.0; // tiempo para mantener la señal al motor
delantero en caso de pérdida de tracción de la rueda trasera
```

```
//--
VARIABLES PARA IDENTIFICAR CUANDO SE DA TENSIÓN AL MOTOR DELANTERO UTILIZANDO
EL PID Y CUANDO NO--//
boolean Automatico = false; // INICIALIZAMOS LA VARIABLE
boolean nuevo_Automatico = false; // INICIALIZAMOS LA VARIABLE
#define FuncionamientoManual 0 // SE PROPORCIONA TENSIÓN MÍNIMA O PROPORCIONAL
AL ACELERADOR AL MOTOR DELANTERO
#define FuncionamientoAutomatico 1 // SE PROPORCIONA TENSIÓN A TRAVÉS DEL PID AL
MOTOR DELANTERO
//-- --//
//-----//
```

```
//--CONFIGURACIÓN DE LOS PINES, DE LAS INTERRUPTIONES Y DEL PUERTO SERIE--//
void setup()
{
  // Se configura el puerto serie
  Serial.begin(9600);
  // Se configuran los pines como entradas, salidas y se activan las resistencia interna pull-
up, LOS PINES ANALÓGICOS NO HACE FALTA CONFIGURARLOS COMO ENTRADAS O COMO
SALIDAS
  pinMode(HALL_TRASERO, INPUT); // Declaramos el pin donde se conecta el sensor hall
trasero como entrada. Es analógico no es necesario configurarlo como entrada o salida
  pinMode(HALL_DELANTERO, INPUT); // Declaramos el pin donde se conecta el sensor hall
delantero como entrada.
  pinMode(BOTON_2X1_OFF, INPUT_PULLUP); // Declaramos el pin del botón como entrada y
habilitamos la resistencia pull-up interna
  pinMode(BOTON_2X1_ON, INPUT_PULLUP); // Declaramos el pin del botón como entrada y
habilitamos la resistencia pull-up interna
  pinMode(BOTON_2X2_OFF, INPUT_PULLUP); // Declaramos el pin del botón como entrada y
habilitamos la resistencia pull-up interna
  pinMode(BOTON_2X2_ON, INPUT_PULLUP); // Declaramos el pin del botón como entrada y
habilitamos la resistencia pull-up interna
  //pinMode(LED_0, OUTPUT); // Comentado, declaramos el pin del led como salida
```



```

//pinMode(LED_1, OUTPUT); // Comentado, declaramos el pin del led como salida
//pinMode(LED_2, OUTPUT); // Comentado, declaramos el pin del led como salida
//pinMode(LED_3, OUTPUT); // Comentado, declaramos el pin del led como salida
pinMode(controlador_delantero, OUTPUT); // Declaramos el pin al que se conectará el motor
delantero como salida

```

```

/*CONFIGURACIÓN DE LAS INTERRUPTACIONES*/
cli(); //deshabilitamos las interrupciones globales
EIMSK |= (1<<INT0); //Declaramos el uso de la INTERRUPTACIÓN 0 EN EL PIN DIGITAL 2
EICRA |= ((1<<ISC01) |(1<<ISCO0)); //Definimos que la detección de la INTERRUPTACIÓN 1
sea por flanco ascendente. También podemos usar cualquiera de los dos o "EICRA" o
"MCUCR" -> MCUCR |= ((1<<ISC01) |(1<<ISCO0));
PCICR |= (1<<PCIE1); // SE HABILITA INTERRUPTACION POR CAMBIO DE PIN Y AHORA SE
UBICARÁ LA INTERRUPTACION 1 ENTRE LOS PINES PCINT14-8 CON EL CAMBIO DE PIN AHORA
LA DETECCIÓN SERÁ POR CAMBIO DE ESTADO YA SEA ASCENDENTE O DESCENDENTE Y ESTO
NO SE PUEDE CAMBIAR
PCMSK1 |= (1<<PCINT11); //SELECCIONAMOS EN PIN PCINT 11 PARA QUE SEA LA
INTERRUPTACIÓN 1, QUE QUIVALE AL PIN A3.
sei(); // habilitamos las interrupciones globales
}

```

```

/*FUNCIONES A LAS QUE SE LLAMA CON LAS INTERRUPTACIONES*/
ISR(PCINT1_vect) // SENSOR TRASERO AL PIN A3
// Rutina de servicio de interrupción. Cada PCInt 8..14
{
// MANEJADOR DE LA INTERRUPTACIÓN POR CAMBIO DE PIN PARA EL PIN A3, UN CAMBIO EN
EL PIN ANALÓGICO 3 GENERARÁ UNA INTERRUPTACIÓN
if (digitalRead(HALL TRASERO) == HIGH)//SE PUEDE PONER COMO IF (PINC & (1<<PINC3)
== 1) QUE SUPUESTAMENTE REALIZARÍA LO MISMO ES SI EL PIN A3 HA CAMBIADO Y ESTÁ A
1
{
/* PCINT11 HA TENIDO UN FLANCO POSITIVO */
CONTADOR TRASERO++; // AUMENTAMOS EL CONTADOR DE FLANCOS ASCENDENTES
DE LA RUEDA TRASERA
}
}

```

```

ISR(INT0_vect) // SENSOR DELANTERO AL PIN 2
{
CONTADOR DELANTERO++; // AUMENTAMOS EL CONTADOR DE FLANCOS ASCENDENTES
DE LA RUEDA DELANTERA
}
/*-----*/

```

```

/--PROGRAMA PRINCIPAL--//
void loop()
{
SELECCION DEL MODO();
CALCULO DIFERENCIA();
}

```

```
//--FIN DEL PROGRAMA PRINCIPAL--//

/* FUNCIÓN CON LA QUE SELECCIONAMOS EL MODO DE FUNCIONAMIENTO */
void SELECCION_DEL_MODO()
{
    // Leemos el valor de los botones y lo guardamos en una variable
    int VALOR_BOTON_2X1_OFF = digitalRead(BOTON_2X1_OFF);
    int VALOR_BOTON_2X1_ON = digitalRead(BOTON_2X1_ON);
    int VALOR_BOTON_2X2_OFF = digitalRead(BOTON_2X2_OFF);
    int VALOR_BOTON_2X2_ON = digitalRead(BOTON_2X2_ON);

    /*Hay que tener en cuenta que las resistencias pull-up
    invierten la lógica de los pulsadores. En el pin obtendremos
    nivel ALTO cuando el pulsador esté sin pulsar y BAJO cuando
    el interruptor esté pulsado.*/
    if (VALOR_BOTON_2X1_OFF == HIGH) // SI EL SWITCH NO ESTÁ EN LA POSICIÓN 2X1 OFF
    {
        if (FLAG_0 == 1) // PARA SOLO REALIZARLO UNA VEZ
        {
            //digitalWrite(LED_0,LOW);// DE MOMENTO COMENTADO, SE APAGA EL LED QUE INDICA
            QUE EL MODO_0 ESTÁ ACTIVADO
            FLAG_0 = 0;
            MODO_0 = 0; // NO ESTÁ SELECCIONADO EL MODO 0
        }
    }
    else // SI EL SWITCH ESTÁ EN LA POSICIÓN 2X1 OFF
    {
        if (FLAG_0 == 0) // PARA SOLO REALIZARLO UNA VEZ
        {
            Serial.println("MODO 2X1 OFF\t"); // SE MUESTRA POR PANTALLA QUE ESTAMOS EN EL
            MODO 2X1 OFF
            cli();// DESHABILITAMOS LAS INTERRUPCIONES CUANDO ACCIONAMOS EL MODO_0
            //digitalWrite(LED_0,HIGH); // DE MOMENTO COMENTADO SE ENCIENDE EL LED QUE
            INDICA QUE EL MODO_0 ESTÁ ACTIVADO
            MODO_0 = 1; // ESTÁ SELECCIONADO EL MODO 0
            FLAG_0 = 1;
            PRIMERA_VEZ_MODO_0 = 1; // CADA VEZ QUE CAMBIEMOS ESTE MODO ESTA VARIABLE
            SE PONDRÁ A 1
        }
    }
    if (VALOR_BOTON_2X1_ON == HIGH) // SI EL SWITCH NO ESTÁ EN LA POSICIÓN 2X1 ON
    {
        if (FLAG_1 == 1) // PARA SOLO REALIZARLO UNA VEZ
        {
            //digitalWrite(LED_1,LOW); // DE MOMENTO COMENTADO, SE APAGA EL LED QUE INDICA
            QUE EL MODO_1 ESTÁ ACTIVADO
            FLAG_1 = 0;
            MODO_1 = 0; // NO ESTÁ SELECCIONADO EL MODO 1
        }
    }
    else // SI EL SWITCH ESTÁ EN LA POSICIÓN 2X1 ON
```

```

{
  if (FLAG_1 == 0) // PARA SOLO REALIZARLO UNA VEZ
  {

    //digitalWrite(LED_1,HIGH); // DE MOMENTO COMENTADO, SE ENCIENDE EL LED QUE
INDICA QUE EL MODO_1 ESTÁ ACTIVADO
    FLAG_1 = 1;
    MODO_1 = 1; // ESTÁ SELECCIONADO EL MODO 1
    CONTADOR_TRASERO = 0; // SE INICIALIZA EL CONTADOR CADA VEZ QUE CAMBIEMOS DE
MODO
    CONTADOR_DELANTERO = 0; // SE INICIALIZA EL CONTADOR CADA VEZ QUE CAMBIEMOS
DE MODO
    tiempo_anterior = tiempo_actual; // SE ACTUALIZA EL VALOR DEL TIEMPO EN EL QUE SE
REALIZÓ EL BUCLE ANTERIOR DE COMPARACION DE FLANCOS ENTRE LAS RUEDAS
    tiempo_anterior_PID = tiempo_actual; // SE ACTUALIZA EL VALOR DEL TIEMPO EN EL QUE
SE EJECUTÓ EL PID ANTERIOR
    sei();// HABILITAMOS LAS INTERRUPCIONES
    Serial.println("MODO 2X1 ON\t"); // SE MUESTRA POR PANTALLA QUE ESTAMOS EN EL
MODO 2X1 ON
  }
}
if (VALOR_BOTON_2X2_OFF == HIGH) // SI EL SWITCH NO ESTÁ EN LA POSICIÓN 2X2 OFF
{
  if (FLAG_2 == 1) // PARA SOLO REALIZARLO UNA VEZ
  {
    //digitalWrite(LED_2,LOW); // DE MOMENTO COMENTADO, SE APAGA EL LED QUE INDICA
QUE EL MODO_2 ESTÁ ACTIVADO
    FLAG_2 = 0;
    MODO_2 = 0; // NO ESTÁ SELECCIONADO EL MODO 2
  }
}
else // SI EL SWITCH ESTÁ EN LA POSICIÓN 2X2 OFF
{
  if (FLAG_2 == 0) // PARA SOLO REALIZARLO UNA VEZ
  {
    //digitalWrite(LED_2,HIGH); // DE MOMENTO COMENTADO, SE ENCIENDE EL LED QUE
INDICA QUE EL MODO_2 ESTÁ ACTIVADO
    FLAG_2 = 1;
    MODO_2 = 1; // ESTÁ SELECCIONADO EL MODO 2
    CONTADOR_TRASERO = 0; // SE INICIALIZA EL CONTADOR CADA VEZ QUE CAMBIEMOS DE
MODO
    CONTADOR_DELANTERO = 0; // SE INICIALIZA EL CONTADOR CADA VEZ QUE CAMBIEMOS
DE MODO
    tiempo_anterior = tiempo_actual; // SE ACTUALIZA EL VALOR DEL TIEMPO EN EL QUE SE
REALIZÓ EL BUCLE ANTERIOR DE COMPARACION DE FLANCOS ENTRE LAS RUEDAS
    tiempo_anterior_PID = tiempo_actual; // SE ACTUALIZA EL VALOR DEL TIEMPO EN EL QUE
SE EJECUTÓ EL PID ANTERIOR
    sei();// HABILITAMOS LAS INTERRUPCIONES
    Serial.println("MODO 2X2 OFF\t"); // SE MUESTRA POR PANTALLA QUE ESTAMOS EN EL
MODO 2X2 OFF
  }
}

```



```
    }
    if (VALOR_BOTON_2X2_ON == HIGH) // SI EL SWITCH NO ESTÁ EN LA POSICIÓN 2X2 ON
    {
        if (FLAG_3 == 1) // PARA SOLO REALIZARLO UNA VEZ
        {
            //digitalWrite(LED_3,LOW);// DE MOMENTO COMENTADO, SE APAGA EL LED QUE INDICA
QUE EL MODO_3 ESTÁ ACTIVADO
            FLAG_3 = 0;
            MODO_3 = 0; // NO ESTÁ SELECCIONADO EL MODO 3
        }
    }
    else // SI EL SWITCH ESTÁ EN LA POSICIÓN 2X2 ON
    {
        if (FLAG_3 == 0) // PARA SOLO REALIZARLO UNA VEZ
        {
            //digitalWrite(LED_3,HIGH);// DE MOMENTO COMENTADO, SE ENCIENDE EL LED QUE
INDICA QUE EL MODO_3 ESTÁ ACTIVADO
            FLAG_3 = 1;
            MODO_3 = 1; // ESTÁ SELECCIONADO EL MODO 3
            CONTADOR_TRASERO = 0; // SE INICIALIZA EL CONTADOR CADA VEZ QUE CAMBIEMOS DE
MODO
            CONTADOR_DELANTERO = 0; // SE INICIALIZA EL CONTADOR CADA VEZ QUE CAMBIEMOS
DE MODO
            tiempo_anterior = tiempo_actual; // SE ACTUALIZA EL VALOR DEL TIEMPO EN EL QUE SE
REALIZÓ EL BUCLE ANTERIOR DE COMPARACION DE PULSOS ENTRE LAS RUEDAS
            tiempo_anterior_PID = tiempo_actual; // SE ACTUALIZA EL VALOR DEL TIEMPO EN EL QUE
SE EJECUTÓ EL PID ANTERIOR
            sei(); // HABILITAMOS LAS INTERRUPTIONES
            Serial.println("MODO 2X2 ON\t"); // SE MUESTRA POR PANTALLA QUE ESTAMOS EN EL
MODO 2X2 ON
        }
    }
}
/* FIN FUNCIÓN CON LA QUE SELECCIONAMOS EL MODO DE FUNCIONAMIENTO */

/* FUNCIÓN CON LA QUE CALCULAMOS LA DIFERENCIA DE FLANCOS POSITIVOS ENTRE LAS R
UEDAS*/
void CALCULO_DIFERENCIA ()
{
    tiempo_actual = micros(); // GUARDAMOS EL TIEMPO ACTUAL
    if ((MODO_0 == 1) && (PRIMERA_VEZ_MODO_0 == 1)) // SI TENEMOS SELECCIONADO EL
MODO 0 Y ES LA PRIMERA VEZ QUE SE HA ENTRADO EN EL MODO 0
    {
        COMPENSAR_ESFUERZO(); // SE LLAMA A ESTA FUNCIÓN PARA DAR UNA TENSIÓN
MÍNIMA A LA RUEDA DELANTERA
        UNA_VEZ = 0;
        BLOQUEO = 0;
        PRIMERA_VEZ_MODO_0 = 0;
    }
}
```

```

if(((tiempo_actual - tiempo_anterior >= tiempo_calculo_diferencia) ||
(CONTADOR_DELANTERO >= PULSOS_LIMITE)) && (MODO_0 == 0) && (MODO_1 == 1 ||
MODO_2 == 1 || MODO_3 == 1))
{
  // SI ESTAMOS EN UNO DE LOS MODOS 1,2,3 Y SI HA PASADO UN TIEMPO DE 0.5
  SEGUNDOS O SE HAN CONTADO 12 PULSOS O MAS EN LA RUEDA DELANTERA
  cli(); //DESHABILITAMOS INTERRUPCIONES
  tiempo_diferente = (tiempo_actual - tiempo_anterior); // Observamos el tiempo
  transcurrido entre la anterior comparación y la actual
  dT_distinto = tiempo_diferente;
  velocidad_rueda_trasera = CONTADOR_TRASERO;
  velocidad_rueda_delantera = CONTADOR_DELANTERO;
  diferencia = (velocidad_rueda_trasera - velocidad_rueda_delantera); // Diferencia de
  flancos ascendentes entre las ruedas
  velocidad_rueda_trasera_final = (((velocidad_rueda_trasera*50*dT_distinto/1000000) +
  (velocidad_rueda_trasera_final_anterior))/ (1 + ((50*dT_distinto)/1000000))); //
  Contabilización de flancos ascendentes filtrada mediante el filtro (50/(50+s))
  velocidad_rueda_delantera_final = ((velocidad_rueda_delantera*50*dT_distinto/100000
  0) + (velocidad_rueda_delantera_final_anterior))/ (1 + ((50*dT_distinto)/1000000))); //
  Contabilización de flancos ascendentes filtrada mediante el filtro (50/(50+s))
  diferencia_filtrada = (velocidad_rueda_trasera_final -
  velocidad_rueda_delantera_final); // Diferencia de flancos ascendentes filtrados entre las
  ruedas
  //CALCULO_DIFERENCIA_DE_PULSOS; // Comentado, está sin testear
  //FUNCION_REGULACION_DE_POTENCIA; // Comentado, está sin testear. Llamar a la
  función regulación de potencia de la rueda delantera para obtener el valor nuevo
  //TOMA_DE_DECISIONES();//DEBERÍA HACERSE AQUÍ PERO SINO NO SE VE LOS
  COMENTARIOS ENVIADOS POR EL PUERTO SERIE
  CONTADOR_TRASERO = 0; // RESETEAMOS EL CONTADOR DE FLANCOS
  CONTADOR_DELANTERO = 0; // RESETEAMOS EL CONTADOR DE FLANCOS
  tiempo_anterior = tiempo_actual; // Actualizamos el valor del tiempo en que se realizó la
  última comparación de flancos
  velocidad_rueda_delantera_final_anterior = velocidad_rueda_delantera_final; //
  Guardamos el valor del número de flancos contabilizado de la rueda delantera filtrado
  velocidad_rueda_trasera_final_anterior = velocidad_rueda_trasera_final; // Guardamos el
  valor del número de flancos contabilizado de la rueda trasera filtrado
  sei();// HABILITAMOS INTERRUPCIONES
  TOMA_DE_DECISIONES();
}
}
/* FIN FUNCIÓN CON LA QUE CALCULAMOS LA DIFERENCIA DE PULSOS */

/* FUNCIÓN CON LA QUE TOMAMOS LAS DECISIONES */
void TOMA_DE_DECISIONES()//La toma de decisiones sólo se realiza en los modos 0, 1 y 2
{
  if((-diferencia_pulsos_limite < diferencia_filtrada) && (diferencia_filtrada <
  diferencia_pulsos_limite) && (BLOQUEO == 0)) // FUNCIONAMIENTO NORMAL
  {
    // SI LAS VELOCIDADES DE LAS RUEDAS SON APROXIMADAMENTE IGUALES Y NO SE HA
    DETECTADO TRACCIÓN TOTAL ANTERIORMENTE DURANTE 3 SEGUNDOS
    UNA_VEZ = 0;
    if (MODO_1) // SI ESTAMOS EN EL MODO 1
    {

```

```
Serial.println(" FUNCIONAMIENTO NORMAL, COMPENSANDO ESFUERZO EN MODO 1\t");
COMPENSAR_ESFUERZO();// SE LLAMA A ESTA FUNCIÓN PARA DAR UNA TENSIÓN
MÍNIMA A LA RUEDA DELANTERA
}
else if (MODO_2 || MODO_3)// SI ESTAMOS EN EL MODO 2 Ó 3
{
Serial.println("FUNCIONAMIENTO NORMAL, TRACCIÓN PROPORCIONAL EN MODO 2 Ó
3\t");
TRACCION_PROPORCIONAL();//LA TRACCIÓN DELANTERA ES UN TANTO POR CIENTO
PROPORCIONAL AL ACELERADOR Y PROPORCIONAL AL REGULADOR DE POTENCIA
}
} // FIN FUNCIONAMIENTO NORMAL
else if(diferencia_filtrada <= -diferencia_pulsos_limite) // SI LA RUEDA DELANTERA VA MAS
DEPRISA QUE LA RUEDA TRASERA
{// FUNCIONAMIENTO RUEDA DELANTERA EN EL AIRE
Serial.println("FUNCIONAMIENTO RD AL AIRE COMPENSANDO ESFUERZO EN MODO 1 2 Ó
3\t");
COMPENSAR_ESFUERZO();// SE LLAMA A ESTA FUNCIÓN PARA DAR UNA TENSIÓN MÍNIMA
A LA RUEDA DELANTERA
BLOQUEO = 0;
UNA_VEZ = 0;
} // FIN FUNCIONAMIENTO RUEDA DELANTERA EN EL AIRE
else if((diferencia_filtrada >= diferencia_pulsos_limite) || (BLOQUEO == 1)) // SI LA RUEDA
TRASERA VA MAS DEPRISA QUE LA RUEDA DELANTERA
{// FUNCIONAMIENTO PÉRDIDA DE TRACCIÓN EN LA RUEDA TRASERA (DERRAPE)
if((MODO_2) && (diferencia_filtrada >= diferencia_pulsos_limite)) // SI ESTAMOS EN EL
MODO 2
{
Serial.println("FUNCIONAMIENTO DERRAPE, TRACCION PROPORCIONAL MODO 2\t");
TRACCION_PROPORCIONAL();//LA TRACCIÓN DELANTERA ES UN TANTO POR CIENTO
PROPORCIONAL AL ACELERADOR Y PROPORCIONAL AL REGULADOR DE POTENCIA
}
else if (MODO_1 || MODO_3) // SI ESTAMOS EN EL MODO 1 Ó 3
{
if(diferencia_filtrada > -diferencia_pulsos_limite) // PARA QUE PUEDA OCURRIR TANTO EN
DERRAPE COMO EN NORMAL DURANTE 3 SEGUNDOS
{//SI NO ESTÁ LA RUEDA DELANTERA EN EL AIRE
tiempo_mantener_traccion = micros();
if(UNA_VEZ == 0)
{
BLOQUEO = 1; // PARA MANTENER LA SEÑAL DURANTE 3 SEGUNDOS A EXCEPCIÓN
DE QUE LA RUEDA DELANTERA GIRE MÁS DEPRISA QUE LA TRASERA
tiempo_anterior_mantener_traccion = tiempo_mantener_traccion;
UNA_VEZ = 1;
}
}
if(tiempo_mantener_traccion - tiempo_anterior_mantener_traccion > 3000000)
{
BLOQUEO = 0; // PASADOS 3 SEGUNDOS LA SEÑAL NO SE MANTIENE A NO SER QUE
VUELVA A DETECTARSE PÉRDIDA DE TRACCIÓN EN LA RUEDA TRASERA
}
}
Serial.println("FUNCIONAMIENTO DERRAPE, TRACCION TOTAL MODO 1 Y 3\t");
```

```

    PID(); // SE LLAMA A LA FUNCIÓN PID
  } // FIN DE SI NO ESTA LA RUEDA EN EL AIRE
} // FIN MODO 1 Ó 3
} // FIN FUNCIONAMIENTO PÉRDIDA DE TRACCIÓN EN LA RUEDA TRASERA (DERRAPE)
}
/* FIN FUNCIÓN CON LA QUE TOMAMOS LAS DECISIONES */

/* REALIZACIÓN DEL PID*/
void PID()
{
  nuevo_Automatico = FuncionamientoAutomatico;
  Ahora_PID = micros(); // Tiempo actual en el que se produce el PID
  dT = (Ahora_PID - tiempo_anterior_PID); // ASÍ SE GUARDA EN UNA VARIABLE EL TIEMPO
  DESDE QUE SE PRODUJO EL PID ANTERIOR Y EL NUEVO
  Error = (velocidad_rueda_trasera_final - velocidad_rueda_delantera_final); // PARA
  CALCULAR EL ERROR O DIFERENCIA DE FLANCOS ASCENDENTES CONTABILIZADOS EN UN
  TIEMPO DETERMINADO ENTRE LAS RUEDAS
  if(nuevo_Automatico && !Automatico) // SI SE PROVIENE DE UN MODO DE DAR TENSIÓN
  AL CONTROLADOR DELANTERO QUE NO SEA MEDIANTE EL PID
  {
    Inicializar(); // Para inicializar el PID cada vez que pasemos de un modo de
    funcionamiento en el que no se utilice el PID
    Serial.println("Inicializamos PID");
  }
  errorProp = kp*Error; // SE CALCULA LA PARTE PROPORCIONAL DEL PID
  // se limita el error proporcional
  if (errorProp > 3.8)
  {
    errorProp = 3.8; // Límite superior
  }else if (errorProp < -3.8)
  {
    errorProp = -3.8; // Límite inferior
  }
  errorsum += (Error*dT*ki/1000000); // es igual que errorsum = errorsum +
  Error*dT*ki/1000000000 la división entre 1000000000 es para ponerlo en función de
  microsegundos // SE CALCULA LA PARTE INTEGRAL DEL PID
  //hay que limitar el error integral que es el errorsum
  if (errorsum > 3.8)
  {
    errorsum = 3.8; // Límite superior
  }else if (errorsum < 0.54)
  {
    errorsum = 0.54; // Límite inferior
  }
  errorDer = ((Error-Error_anterior)*kd)/dT; // no se multiplica por 1000000 porque kd lo va
  a absorber // SE CALCULA LA PARTE DERIVATIVA DEL PID
  // se limita el error derivativo
  if (errorDer > 2)
  {
    errorDer = 2; // Límite superior
  }
}

```



```
}else if (errorDer < -2)
{
    errorDer = -2; // Límite inferior
}
salida = ((errorsum)+ errorProp + errorDer);// LA SALIDA DEL PID TOTAL ES LA SUMA DE
TODAS LOS TÉRMINOS, EL PROPORCIONAL, EL INTEGRAL Y EL DERIVATIVO
//hay que limitar la salida total
if (salida > 3.8)
{
    salida = 3.8; // Límite superior
}else if (salida < 0.54)
{
    salida = 0.54; // Límite inferior
}else
{
    //No se hace nada pues se queda como está
}
velocidad_RD_float = ((salida * (31.4)) + 80); // SE TRADUCE ESA SEÑAL DE TENSIÓN A
VALORES ENTRE 0 Y 255 (0 Y 5 VOLTIOS) Y SE REALIZA DE ESTA FORMA PARA QUE AMBOS
MOTORES EMPIECEN A FUNCIONAR A LA VEZ
velocidad_RD_int = velocidad_RD_float;
if(velocidad_RD_float - velocidad_RD_int > 0.5)
{
    velocidad_RD_int += 1;
}
velocidad_RD_int = constrain(velocidad_RD_int,0,175); // Se limita la señal a comandar al
motor delantero
Mandar_al_controlador();
//-----
Error_anterior = Error; // Se guarda la diferencia de flancos ascendentes anterior
Automatico = nuevo_Automatico;
tiempo_anterior_PID = Ahora_PID; // Se guarda el tiempo en el que se realizó el último PID
}
/* FIN REALIZACIÓN DEL PID*/

/* CALCULO DE LA DIFERENCIA DE PULSOS */ // SIN TESTEAR
/*
void CALCULO_DIFERENCIA_DE_PULSOS()
{
    if (tiempo_diferente >= 500000)
    {
        if(velocidad_rueda_delantera_final <= 2)
        {
            diferencia_pulsos_limite = 1;
        }
        else if(2 < velocidad_rueda_delantera_final <= 4.5)
        {
            diferencia_pulsos_limite = 2;
        }
        else if(4.5 < velocidad_rueda_delantera_final <= 9)
```

```

{
  diferencia_pulsos_limite = 3;
}
else if(9 < velocidad_rueda_delantera_final <= 12)
{
  diferencia_pulsos_limite = 4;
}
}
else if(342857.14 <= tiempo_diferente < 500000)
{
  diferencia_pulsos_limite = 3;
}
else if(230769.24 <= tiempo_diferente < 342857.14)
{
  diferencia_pulsos_limite = 3;
}
else if(171428.57 <= tiempo_diferente < 230769.24)
{
  diferencia_pulsos_limite = 2;
}
else if(137931.03 <= tiempo_diferente < 171428.57)
{
  diferencia_pulsos_limite = 2;
}
}
*/
/* FIN DEL CALCULO DE LA DIFERENCIA DE PULSOS */

/* FUNCIÓN REGULACIÓN DE POTENCIA */ // SIN TESTEAR
/*void FUNCION_REGULACION_DE_POTENCIA()
{
  valor_regulador = analogRead(Regulador_de_potencia); // Guardamos el valor del acelerado
  r
  if (valor_regulador <= 100)
  {
    valor_nuevo = 0.1;
  }
  else if (100 < valor_regulador && valor_regulador <= 200)
  {
    valor_nuevo = 0.2;
  }
  else if (200 < valor_regulador && valor_regulador <= 300)
  {
    valor_nuevo = 0.3;
  }
  else if (300 < valor_regulador && valor_regulador <= 400)
  {
    valor_nuevo = 0.4;
  }
  else if (400 < valor_regulador && valor_regulador <= 500)

```



```
{
    valor_nuevo = 0.5;
}
else if (500 < valor_regulador && valor_regulador <= 600)
{
    valor_nuevo = 0.6;
}
else if (600 < valor_regulador && valor_regulador <= 700)
{
    valor_nuevo = 0.7;
}
else if (700 < valor_regulador && valor_regulador <= 800)
{
    valor_nuevo = 0.8;
}
else if (800 < valor_regulador && valor_regulador <= 900)
{
    valor_nuevo = 9;
}
else// (900 < valor_regulador && valor_regulador <= 1023)
{
    valor_nuevo = 1.0;
}
//Serial.println(valor_regulador);
//Serial.println(valor_nuevo);
}
*/
/* FIN FUNCIÓN REGULACIÓN DE POTENCIA */

/*----FUNCIÓN COMPENSAR ESFUERZO----*/
void COMPENSAR_ESFUERZO()
{
    Automatico = FuncionamientoManual;
    nuevo_Automatico = FuncionamientoManual;
    velocidad_RD_int = 97; // esto es lo que tenía calculado inicialmente QUE CORRESPONDE A
    LA TENSIÓN MÍNIMA QUE DAR A LA RUEDA DELANTERA PARA QUE LOS IMANES DEJEN DE
    EJERCER OPOSICIÓN AL GIRO DE LA RUEDA
    //velocidad_RD_float = ((0.54 * (31.4)) + 80);//EL MÍNIMO QUE SE DABA ERA CON SALIDA
    0.54
    //velocidad_RD_int = velocidad_RD_float;
    //if(velocidad_RD_float - velocidad_RD_int > 0.5) // int son 96 y float 96.96 por lo que la
    resta es mayor que 0.5 por lo que se suma 1
    //{
    // velocidad_RD_int += 1; // DE QUÍ VIENE EL 97
    //}
    analogWrite(controlador_delantero, velocidad_RD_int);
    tiempo_anterior_PID = micros();
}
/*----FIN FUNCIÓN COMPENSAR ESFUERZO----*/
```

```

/* FUNCION PARA DAR UNA TRACCIÓN UN TANTO POR CIENTO PROPORCIONAL */
void TRACCION_PROPORCIONAL()
{
  Automatico = FuncionamientoManual;
  nuevo_Automatico = FuncionamientoManual;
  valor_acelerador = analogRead(Acelerador); // LEEMOS EL VALOR QUE LLEGA POR EL PIN
AL QUE SE CONECTA EL ACELERADOR
  valor_acelerador_traducido=map(valor_acelerador,0,1023,40,247); // Con el mapeado
conseguiamos que ambas ruedas empiecen a funcionar a la vez pero no significa que ambas
ruedas circulen constantemente a la misma velocidad.
  if (valor_acelerador_traducido >= 165) // SE LIMITA EL MÁXIMO
  {
    valor_acelerador_traducido = 165;
  }
  /*valor_acelerador_traducido_float
valor_acelerador_traducido_float =
valor_acelerador_traducido*proporcion_acelerador*valor_nuevo;
valor_acelerador_traducido_float = valor_acelerador_traducido_float;
if(valor_acelerador_traducido_float - valor_acelerador_traducido > 0.5)
{
  valor_acelerador_traducido += 1;
}*/
  analogWrite(controlador_delantero,valor_acelerador_traducido);
  tiempo_anterior_PID = micros();
}
/* FIN FUNCION PARA DAR UNA TRACCIÓN UN TANTO POR CIENTO PROPORCIONAL */

/* FUNCION QUE INICIALIZA ALGUNOS PARÁMETRO DEL PID AL PASAR DEL MODO MANUAL A
L AUTOMATICO */
void Inicializar()
{
  errorsum = 0.0;
  Error_anterior = 0.0;
}
/* FIN FUNCION QUE INICIALIZA ALGUNOS PARÁMETRO DEL PID AL PASAR DEL MODO MANU
AL AL AUTOMATICO */

/*----FUNCIÓN QUE MANDA LA SEÑAL AL CONTROLADOR----*/
void Mandar_al_controlador()
{
  analogWrite(controlador_delantero,(velocidad_RD_int*valor_nuevo));
}
/*----FIN FUNCIÓN QUE MANDA LA SEÑAL AL CONTROLADOR----*/

```