



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería Electrónica Industrial y Automática

**Control sincronizado de dos motores
síncronos de diferentes características
mediante LabVIEW.**

Autor:

Vallejo Luis, Raúl

Tutor:

**Parra Gonzalo, Eduardo Pedro
Departamento de Ingeniería
Eléctrica**

Valladolid, Julio de 2021

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Resumen

En este trabajo se pretende realizar el control sincronizado de dos motores eléctricos síncronos de diferentes características, mediante una sencilla aplicación que se puede controlar desde el ordenador o, desde un panel de pulsadores. Como controlador emplearemos una tarjeta Raspberry PI.

Además del programa de control, que cuenta con dos bucles while loop, destinados cada uno de ellos al control independiente de cada motor, también se ha creado la parte física del controlador, es decir, dos placas de circuito impreso en las que se encuentra el circuito integrado que controlan el motor y, también unos optoacopladores, que permiten separar eléctricamente las señales de la tarjeta de control, del circuito integrado, para evitar que un fallo en la parte de potencia pueda destruir la Raspberry PI.

Durante el proyecto, se probarán varios motores, desde un motor paso a paso de un disco duro, motores síncronos y el alternador de un vehículo.

Abstract

In this assignment, the synchronized control of two synchronous electric motors with different characteristics will be done, with a simple application that can be controlled from the computer or from a button panel. As the controller we will use a Raspberry PI card.

The control program has two while loops, for independent control of each motor. The physical part of the controller has also been created, that is, two printed circuit boards that contain the integrated circuit that control the motor and the optocouplers, which electrically separate the signals from the control card, from the integrated circuit, to avoid a failure in the power part that can destroy the Raspberry PI.

During the project, we will test several motors, for example, a hard drive stepper motor, a synchronous motor and a car's alternator.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Palabras clave

1. Motor.
2. Raspberry.
3. LabVIEW.
4. Síncrono.
5. Frecuencia.

Keywords

1. Engine.
2. Raspberry.
3. LabVIEW.
4. Synchronous.
5. Frequency.

Agradecimientos

Me gustaría agradecer en primer lugar a todas las personas que me han apoyado durante todos estos años, mis padres, mis abuelos y mi hermano.

También a mi tutor de este trabajo, Eduardo Parra, que me ha ayudado en cuantas cosas le he pedido.

Y, por último, agradecer a todos los miembros del Departamento de Ingeniería Eléctrica de la Escuela de Ingenierías Industriales de Valladolid que en tantas cosas me han ayudado a lo largo de estos meses. Alberto, Moisés, Carlos, David y Diego, muchas gracias por todo.

¡Gracias a todos!

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

Índice

Resumen	3
Abstract.....	3
Palabras clave	4
Keywords	4
Agradecimientos	5
1. Introducción.....	9
1.1. Objetivos:.....	9
1.2. Estructura:	10
2. Motores eléctricos.	11
2.1. Motor de Corriente Continua.....	11
2.2. Motor Síncrono.....	12
2.3. Motor asíncrono.	13
2.4. Motor de Reluctancia Conmutada.....	15
2.5. Motor Paso a Paso.....	16
2.6. Motor de Flujo Axial.....	17
3. Control del Motor Síncrono como un Motor Paso a Paso.	19
3.1. Estudio teórico.	19
3.2. Software y hardware empleado para el control.	28
3.2.1. Raspberry PI 4.....	28
3.2.2. Controlador con IRAMY20UP60B.	29
3.2.3. LabVIEW.....	32
3.3. Programa de control.	33
3.4. Análisis de los resultados obtenidos.	47
4. Control sincronizado para dos motores de diferentes características.....	53
4.1. Estudio teórico.	53
4.2. Programa de control.	55
4.3. Implementación física.	61
4.4. Resultados obtenidos.....	77
4.5. Estudio económico.....	86
5. Conclusiones obtenidas y trabajos relacionados.	87
5.1. Conclusiones.....	87
5.2. Trabajos relacionados.	88

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

Índice de figuras.....	89
Índice de tablas.....	93
Bibliografía	95

1. Introducción

Actualmente uno de los principales desarrollos tecnológicos que se está llevando a cabo, tiene que ver con el vehículo eléctrico. En los últimos años se ha ido convirtiendo cada vez más en una realidad y, por ello, se ha decidido realizar este trabajo fin de grado, en el que se pretende realizar de manera eléctrica el mismo trabajo que de manera mecánica realiza el diferencial de las ruedas de un vehículo.

Además, también se va a realizar el control de velocidad para un motor síncrono, como los empleados en los nuevos vehículos eléctricos.

1.1. Objetivos:

En este trabajo fin de grado, como objetivo principal se ha marcado realizar un control sincronizado para dos motores síncronos de diferentes características.

A continuación, vamos a describir uno por uno los objetivos que se pretenden alcanzar durante todas las pruebas que se van a realizar.

- Construir y programar por medio de LabVIEW un programa que permita controlar la velocidad de un motor síncrono, alimentándolo como un motor paso a paso, es decir, alimentando de manera secuencial las bobinas del estator.
- Implementar el control del programa por medio de una tarjeta de bajo coste como puede ser la Raspberry PI 4.
- Realizar una variación de la velocidad de un motor mientras que la del otro se mantiene constante, con el objetivo de conseguir de manera eléctrica lo que de forma mecánica realiza el diferencial de un vehículo.
- Implementar en una placa de circuito impreso todo el montaje del control para reducir problemas de malas conexiones o fallos en la alimentación.
- Incorporar todo el montaje a un bastidor desde el cual se alimentará a los motores, conectando las entradas de potencia y las salidas del motor mientras se evita tener que tocar toda la parte de las señales de control.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

1.2. Estructura:

Este Trabajo Fin de Grado se ha estructurado de la siguiente manera:

- Capítulo 1: Se relata una breve introducción acerca de cómo se va a desarrollar el trabajo, describiendo los objetivos de partida y finalmente se plantea el desarrollo estructural del trabajo.
- Capítulo 2: En este apartado se realiza una descripción acerca de cómo funcionan los principales tipos de motores eléctricos que más se emplean en la actualidad.
- Capítulo 3: Se realiza un estudio teórico acerca de cómo crear un programa para controlar un motor síncrono como un motor paso a paso, se describen también el circuito integrado de control que se va a emplear, la tarjeta que controlará todo el proceso y contendrá el programa, el software empleado y finalmente se analizarán los resultados obtenidos.
- Capítulo 4: En esta parte del trabajo se va a crear finalmente el programa que permite el control de dos motores de diferentes características y se detalla el proceso de construcción de la placa de circuito empleada para los componentes electrónicos destinados al control. También en este capítulo se va a ensamblar todo el circuito creado dentro de un bastidor para hacer más robusto el controlador creado y finalmente, se tomarán varias medidas en distintas situaciones y con distintos motores.
- Capítulo 5: En este último capítulo se van a exponer las conclusiones finales a las que se ha llegado una vez finalizado el trabajo y, también, otros posibles trabajos o ampliaciones de este que se pudieran llevar a cabo en un futuro.
- A continuación de estos capítulos se añade un índice de todas las figuras y tablas empleadas a lo largo del trabajo y la bibliografía empleada.

2. Motores eléctricos¹.

Actualmente en el mercado existen gran cantidad de máquinas eléctricas, pero dependiendo la utilidad para la que se necesite, debemos escoger según sus características. Por ello, a continuación, se describen los tipos de motores eléctricos más se emplean en la actualidad.

2.1. Motor de Corriente Continua.

El motor de corriente continua, como bien indica su nombre, es una máquina eléctrica que se alimenta en corriente continua. El funcionamiento de la máquina de continua convencional se basa en crear un imán, es decir, un campo magnético fijo, por medio de la circulación de corriente continua en el bobinado del estator, y crear otro campo magnético en el rotor de la misma forma. Es necesario excitar ambas partes de la máquina en corriente continua para su funcionamiento.

A continuación, se representa en la Figura 1 el esquema eléctrico equivalente de una máquina de corriente continua dependiendo de su excitación.

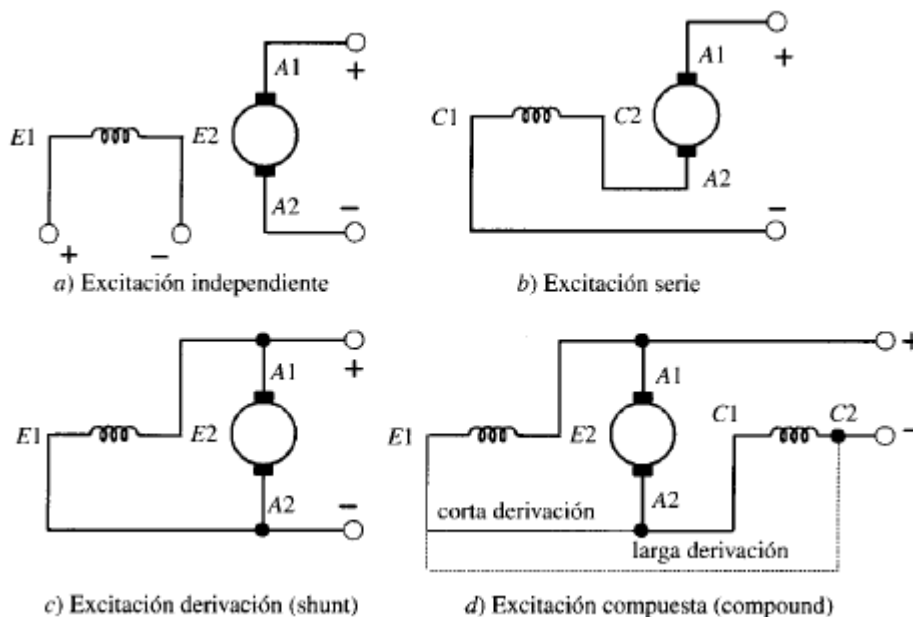


Figura 1: Tipos de excitación de las máquinas de Corriente Continua. [1]

Esta máquina necesita un sistema de escobillas, que contacten con diferentes electrodos según el giro, para que el campo creado por el bobinado del rotor cambie de polaridad. Así, cuando el polo del imán del rotor está a punto de atrapar al polo fijo opuesto del estator, el imán del

¹ Extraído parcialmente de TFG “Movilidad eléctrica en el transporte de viajeros” [11] y de la información extraída acerca de los distintos tipos de motores del libro El vehículo eléctrico, Desafíos tecnológicos, infraestructuras y oportunidades de negocio. [4]

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

rotor debe volver a girar un cierto ángulo atraído por el imán del estator. Esta sucesión de cambios de polaridad en la corriente inyectada al rotor se realiza de manera mecánica gracias a las escobillas y las delgas y así se consigue que el imán del rotor no llegue nunca a atrapar al imán del estator, consiguiendo de este modo un giro continuo del eje con cierto par.

Las ventajas que nos ofrece son la alta densidad de potencia que tiene, el elevado par de arranque, una amplia gama de velocidades fácilmente controlables, la posibilidad de operación de la máquina en los cuatro cuadrantes, un control sencillo, par constante y una buena robustez mecánica.

Como inconvenientes está la baja eficiencia de la máquina (cercana al 80%), los ruidos y vibraciones que produce, el tamaño y el peso de la misma, la difícil fabricación, el elevado coste y, sobre todo, la necesidad de mantenimiento tanto de las escobillas como del colector de delgas.

Este tipo de motor ya no se suele emplear actualmente para la automoción, únicamente continúa utilizándose el motor brushless, es decir, sin escobillas, para algunas motocicletas de baja potencia.

2.2. Motor Síncrono.

El motor síncrono, es una máquina eléctrica que recibe su nombre debido a su funcionamiento interno, ya que la velocidad de los imanes del rotor y del estator es la misma, se dice que giran síncronamente. En la Figura 2, se muestra el circuito equivalente de la máquina síncrona.

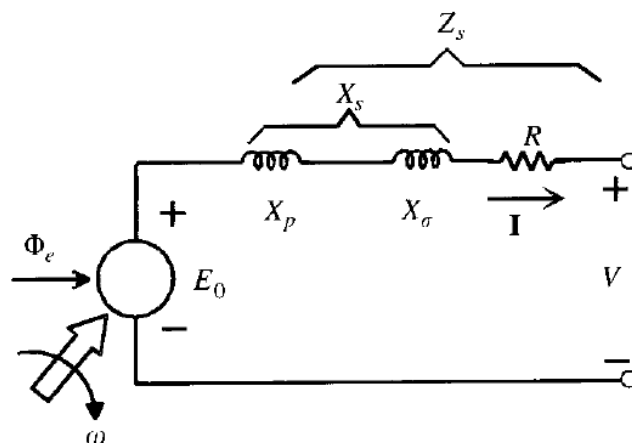


Figura 2: Circuito equivalente de una Máquina Síncrona. [1]

Esta máquina emplea una alimentación en corriente alterna trifásica, aprovechando las propiedades de las ondas desfasadas para crear campos magnéticos giratorios. Si se coloca estratégicamente el bobinado del estator, se puede conseguir que la superposición de los campos

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

magnéticos inducidos en el bobinado de cada fase forme el campo magnético giratorio del estator. La rotación del imán creado en el estator intentará ser copiada por el imán del rotor. Si la máquina funciona como motor el imán del rotor será el que siga al del estator, mientras que, si ésta trabaja como generador, el imán del estator será el que persiga al del rotor.

Podemos distinguir tres tipos de motores síncronos, según la forma constructiva del rotor:

- Rotor cilíndrico o liso: se trata de un rotor en el que el bobinado es enrollado en ranuras y se emplea sobre todo para máquinas de elevada velocidad.
- Rotor de polos salientes: en este caso, el bobinado se enrolla en una pieza de acero con los polos marcados. Se emplea para máquinas de menor velocidad.
- Imán permanente o también llamado brushless: en este caso el imán del rotor en lugar de ser creado por una corriente eléctrica es un imán de materiales provenientes de las tierras raras, como el neodimio. Se emplea en máquinas de potencias que no superan los 200 – 250 KW y es el motor más común utilizado actualmente para la tracción eléctrica e híbrida.

Las ventajas que ofrece este tipo de motor eléctrico son la alta densidad de potencia, el elevado par de arranque, la alta eficiencia, la posibilidad de operación en los cuatro cuadrantes, el control sencillo de la máquina, bajos niveles de ruidos y vibraciones, presenta un par constante con poca vibración, es una máquina bastante robusta térmica y mecánicamente, además la fabricación y el mantenimiento son bastante fáciles.

Como inconvenientes tiene la baja gama de velocidades y el coste de fabricación elevado.

2.3. Motor asíncrono.

El motor asíncrono debe su nombre a que las velocidades a las que giran los campos magnéticos del rotor y del estator no son las mismas. También se le conoce como motor de inducción debido a que las corrientes que circulan por el rotor no son inyectadas, sino que son inducidas.

A continuación, se muestra en la Figura 3 el modelo eléctrico de este motor.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

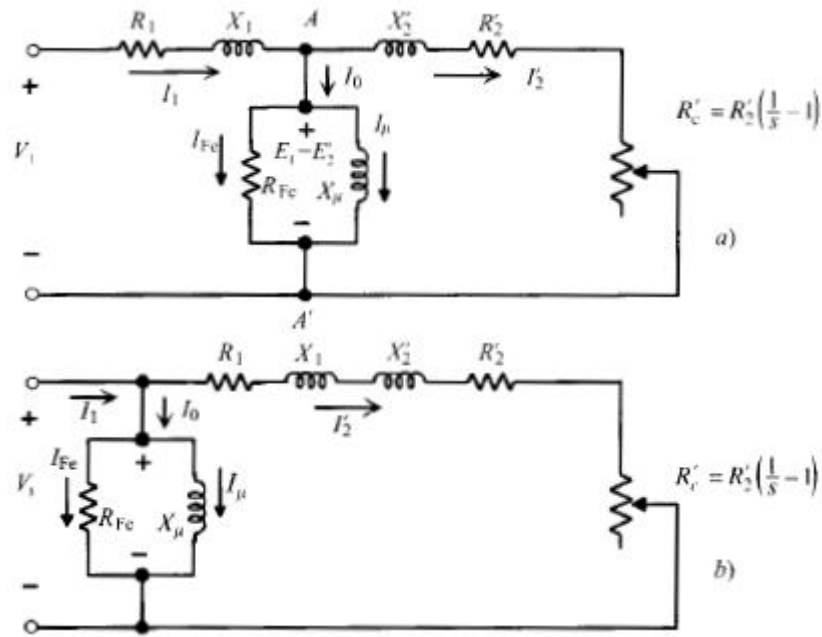


Figura 3: Circuito equivalente del motor asíncrono: a) exacto; b) aproximado. [1]

El estator del motor asíncrono es exactamente igual que el de la máquina síncrona, con los devanados desfasados 120° , mientras que el rotor en este caso está compuesto por multitud de chapas apiladas en forma de cilindro, entre las cuales hay unas ranuras que dan lugar a dos arquitecturas diferentes:

- Rotor bobinado: en las ranuras de la chapa se enrollan los conductores que forman el devanado. Éste es igual que el del estator, pero con más espiras debido a la utilización de conductores de menor sección, que aumentan la resistividad del conjunto. El bobinado del rotor acostumbra a ser también trifásico, con una conexión en estrella en un extremo de los conductores y salida al exterior del rotor a través de unos anillos rozantes y escobillas en el otro extremo. Las tres fases salientes a través de las escobillas se cortocircuitan en el exterior una vez se ha completado el procedimiento de arranque, mediante el cual se obtiene un par mayor acoplado una resistencia al bobinado.
- Rotor de jaula de ardilla: las ranuras se llenan de aluminio inyectado de manera que se formen barras de este material, que quedarán unidas por anillos transversales en los extremos del rotor, como en una jaula de ardilla. Cada una de estas barras, junto con la contigua y los anillos que la unen, forman una espira conductora. La variación del flujo de campo magnético que atraviese esta espira inducirá una corriente que recorrerá el conductor en sentido contrario.

Como ventajas de este motor eléctrico tenemos la amplia gama de velocidades, el buen par de arranque, la alta eficiencia, la posibilidad de

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

operación en los cuatro cuadrantes, el control sencillo de la máquina, bajos niveles de ruidos y vibraciones, presenta un par constante con poca vibración, es una máquina bastante robusta térmica y mecánicamente, la fabricación y el mantenimiento son bastante fáciles y el coste es bajo.

El principal inconveniente es la baja densidad de potencia.

El uso de estos motores en automoción es para los vehículos de peso mediano, aunque también se emplea como motor auxiliar para numerosas aplicaciones.

2.4. Motor de Reluctancia Conmutada.

El motor de reluctancia conmutada está formado por un rotor dentado de material ferromagnético que se orienta según el campo magnético creado en el estator. El polo del estator en el que se crea instantáneamente el campo magnético atrae al diente del rotor más próximo, de forma que entre el polo y el material ferromagnético del rotor haya la mínima distancia. Es decir, las líneas de campo magnético intentan ir por el camino más corto posible del estator al rotor para cerrar el circuito magnético. En la Figura 4, se representa cómo funciona internamente un motor de estas características.

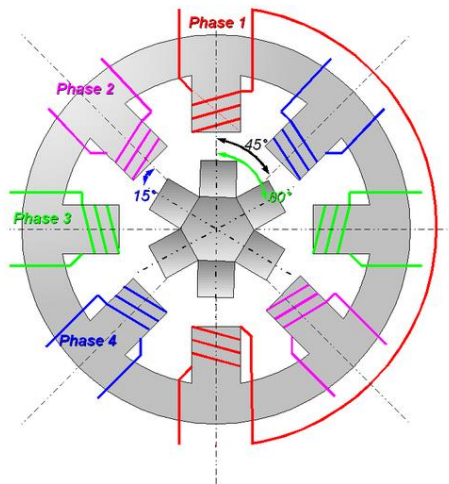


Figura 4: Motor de Reluctancia Conmutada. [2]

La conmutación y la variación del campo creado en el estator se realiza mediante inversores y convertidores, gracias a los cuales es posible modificar la velocidad y el par de giro del motor.

Las ventajas que aporta son la alta densidad de potencia, la alta eficiencia, la amplia gama de velocidades, la posibilidad de operación en los cuatro cuadrantes, el control sencillo del motor, el tamaño y el peso bajo, la buena robustez térmica y mecánica que posee, la facilidad de fabricación y mantenimiento y el bajo coste.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Pero como inconvenientes está la baja capacidad de sobrecarga, el muy bajo par de arranque, los altos niveles de ruidos y vibraciones y, sobre todo, que este tipo de motores entregan un par fluctuante, lo cual no es nada bueno de cara a emplearle como motor de tracción, ya que da la sensación de avanzar a trompicones.

Actualmente se ha empleado para ensayos en algunos prototipos o incluso en alguno de los autobuses híbridos que se describen en el capítulo 2 que ya están en el mercado.

2.5. Motor Paso a Paso.

El motor Paso a Paso es un dispositivo electromecánico que convierte una serie de impulsos eléctricos en variaciones angulares, por lo que es capaz de alcanzar una posición determinada (en grados) dependiendo del número de impulsos recibidos. Es capaz de realizar desplazamientos con mucha precisión.

En la Figura 5, se muestra el funcionamiento de este motor.

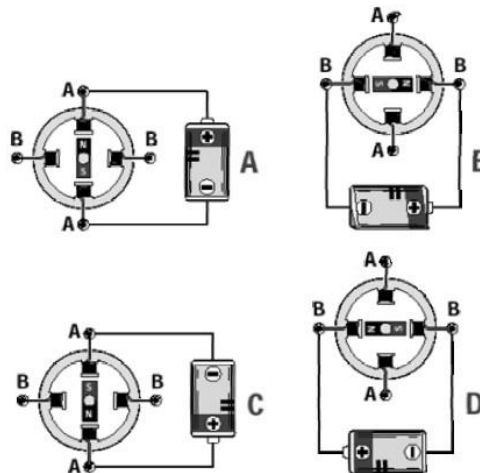


Figura 5: Motor Paso a Paso. [3]

El principio de funcionamiento en el que se basan este tipo de motores consiste en un estator que contiene varios bobinados de un material ferromagnético y un rotor que es capaz de girar libremente. En cuanto al rotor, se puede componer de distintas maneras, una de ellas, es con imanes permanentes, es decir, el funcionamiento será similar al de una máquina de corriente continua, cuando una bobina del estator esté conmutada. Otra forma de rotor consiste en hacer un motor de reluctancia conmutada (explicado en el apartado anterior) y, por último, un rotor híbrido, es decir, que emplee una combinación de las dos tecnologías.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Para que el rotor pueda girar, es necesario que se inyecten una serie de pulsos eléctricos a las bobinas del estator, como se ve en la Figura 5 y, además, éstos han de ir en una secuencia correcta, que permita el avance del motor. Una peculiaridad de este motor es que, si se deja una o más bobinas con tensión, el rotor se queda fijo en esa posición.

Por otra parte, podemos distinguir dos tipos de motores paso a paso, según la conexión eléctrica que realicemos:

- **Bipolares:** Por lo general tienen 4 cables de conexión, es decir, uno para cada extremo de las dos bobinas que contiene el motor. Para su control, es necesario un puente en H para cada una de las bobinas.
- **Unipolares:** Su control es mucho más sencillo que en el caso de un motor paso a paso bipolar. Pueden tener 5 cables (uno para cada extremo de las bobinas y un común conectado al centro de ambas bobinas), 6 cables (uno para cada extremo de las bobinas y dos comunes, conectados cada uno al centro de cada una de las bobinas) u 8 cables (cada bobina está dividida en dos partes, es decir, cada polo de un par de polos se alimenta de forma independiente).

El principal uso de este tipo de motores es para aplicaciones en las que se requiera un control del movimiento sin necesidad de un encoder, por ejemplo, en impresoras 3D.

2.6. Motor de Flujo Axial.

El motor de flujo axial es totalmente diferente a los motores anteriores ya que, en este caso, el flujo magnético que permite mover el eje va en la dirección axial.

El uso de este motor no es nuevo, debido a que en los orígenes del automóvil eléctrico ya se empleó en algún prototipo. El objetivo del uso de este tipo de motores es colocar uno en cada rueda, de manera que se ahorrarían muchas piezas del vehículo y se conseguiría una tracción total e independiente.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

A continuación, en la Figura 6 se puede visualizar cómo está formado este motor.

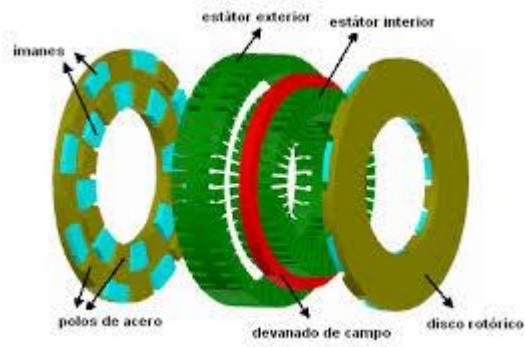


Figura 6: Máquina de Flujo Axial. [4]

3. Control del Motor Síncrono como un Motor Paso a Paso.

Uno de los objetivos marcados para la realización de este Trabajo Fin de Grado, es lograr controlar un motor síncrono como un motor paso a paso. Para ello, una vez comprendida la forma de funcionamiento del motor paso a paso, es decir, que recibe pulsos de corriente continua, vamos a estudiar cómo es posible realizar esta actuación para este otro motor de una tecnología diferente, pero con características constructivas similares.

3.1. Estudio teórico.

La idea principal que vamos a detallar en este apartado consiste en determinar de qué forma es necesario alimentar al motor síncrono (por simplicidad vamos a imaginar que es un motor síncrono de imanes permanentes y, así, olvidamos la regulación en la excitación).

Para llevar a cabo esta idea, vamos a analizar detalladamente cómo trabaja un motor paso a paso y su regulador. El funcionamiento de este tipo de motores consiste en una secuencia de pulsos eléctricos que originan finalmente el movimiento del rotor. Cada uno de estos pulsos es similar a que se fuesen alimentando cada una de las bobinas, con una tensión continua y de forma individual, en un orden correcto, como se ve en la Figura 7.

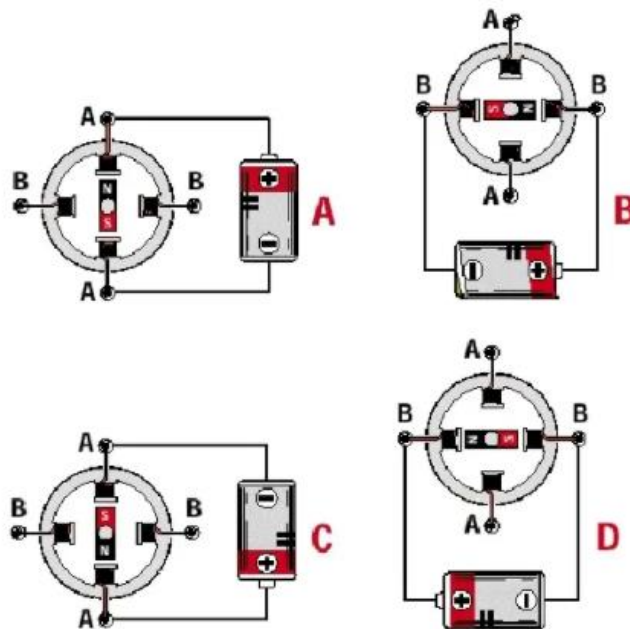


Figura 7: Secuencia de alimentación del Motor Paso a Paso. [3]

Si analizamos como está formado el motor síncrono de manera estructural, vemos que tiene tres bobinas separadas 120° y un rotor que es un imán

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

(bien un imán permanente o un electroimán), que puede tener uno o varios pares de polos. En la Figura 8, se muestra esta descripción.

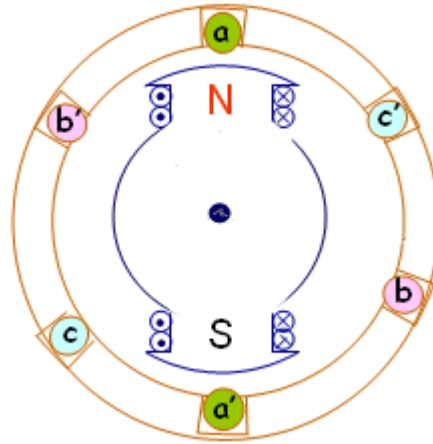


Figura 8: Motor Síncrono. [2]

El siguiente paso, es detallar el conexionado que vamos a realizar. Para ello, tenemos en cuenta que la alimentación del motor se va a realizar por medio de un puente de IGBTs. Como vamos a tener 3 salidas del puente, entonces realizaremos la unión de a' , b' y c' en estrella y será necesario que siempre se alimente con el positivo de la tensión continua una de las entradas, otra con el negativo y la tercera quede libre.

De esta forma, siempre se alimentan dos bobinas del motor, que a su vez generarán el campo magnético necesario para mover el rotor.

En la Figura 9, se muestra cómo se ha de llevar a cabo el conexionado del motor descrito anteriormente.

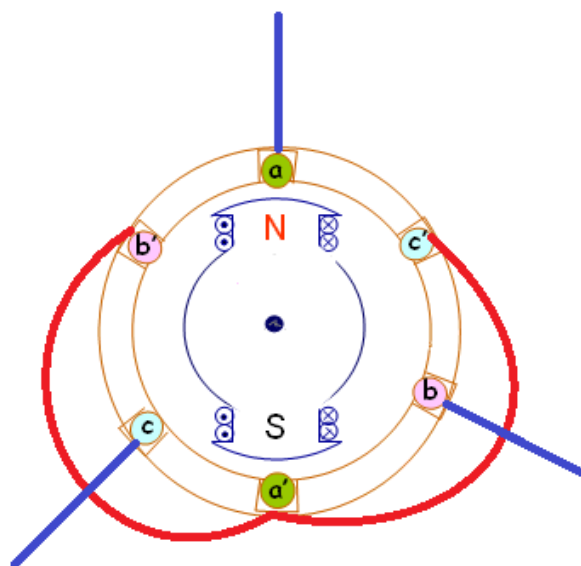


Figura 9: Conexionado del Motor. [2]

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Antes de generar la secuencia de pulsos que controle el motor, necesitamos analizar cómo está constituido el puente de IGBTs necesario para crear las señales de entrada. En la Figura 10, se muestra una imagen de cómo se compone dicho puente, con las señales de entrada para activar dichos IGBTs y las señales de salida que alimentan el motor.

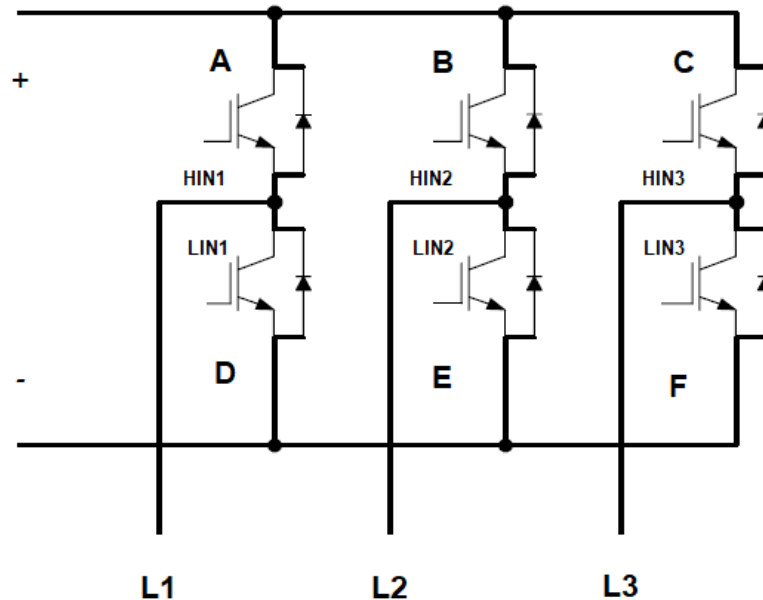


Figura 10: Puente de IGBTs para control. (Elaboración propia)

Analizando el puente, vemos que hay tres IGBTs en la parte alta y tres en la parte baja, por lo que hay que tener cuidado a la hora de hacerlos conmutar, para evitar que se produzca un cortocircuito al conmutar dos de la misma rama.

En cuanto a las señales de control necesarias para controlar el movimiento del motor, en la Figura 11, se muestra cómo se han de conmutar estos IGBTs de forma correcta para que el motor realice un giro completo. En esta representación, se ha empleado un rotor con un único par de polos, por lo que el motor dará una vuelta en 6 pasos. Si el motor tiene varios pares de polos en el rotor, girará una vuelta completa el eje del motor cuando se reciban $(6 * n^{\circ} \text{ pares de polos})$ pasos.

También podemos ver que el desplazamiento del eje del motor en cada paso, para un par de polos en el rotor, es de $60^{\circ}/\text{paso}$ y, para un número de pares de polos distinto, este desplazamiento angular en será de $60^{\circ}/(n^{\circ} \text{ pares de polos})$ por paso.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

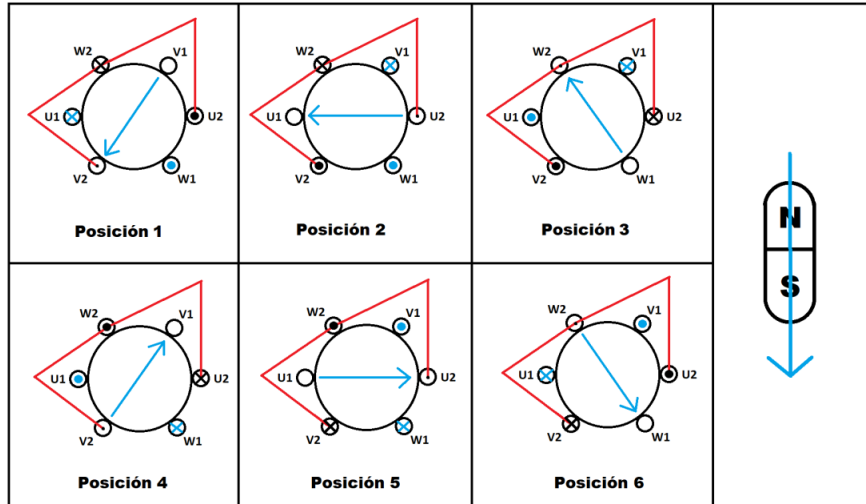


Figura 11: Secuencia de pasos del motor. [5]

Una vez obtenidas las posiciones de desplazamiento del rotor dependiendo de la secuencia de pasos, creamos una tabla (Tabla 1) que relacione la alimentación de las bobinas con el paso en que se encuentra.

	1	2	3	4	5	6
U1	V+		V-	V-		V+
V1		V+	V+		V-	V-
W1	V-	V-		V+	V+	

Tabla 1: Secuencia de alimentación del motor. (Elaboración propia)

A continuación, creamos la tabla de conmutación de cada uno de los IGBTs del puente (Tabla 2), que será la secuencia que habrá que programar en la unidad de control.

Posición\Transistor	HIN1	LIN1	HIN2	LIN2	HIN3	LIN3
1	1	0	0	0	0	1
2	0	0	1	0	0	1
3	0	1	1	0	0	0
4	0	1	0	0	1	0
5	0	0	0	1	1	0
6	1	0	0	1	0	0

Tabla 2: Secuencia de conmutación de los IGBTs. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Los nombres empleados en la Tabla 2 para los IGBTs son los que aparecen en la Figura 10.

Para comprobar que el funcionamiento del puente es el correcto, que los pulsos creados y que el motor funciona correctamente con este tipo de alimentación, se ha procedido a simular mediante Matlab el circuito empleado, como se ve en la Figura 12.

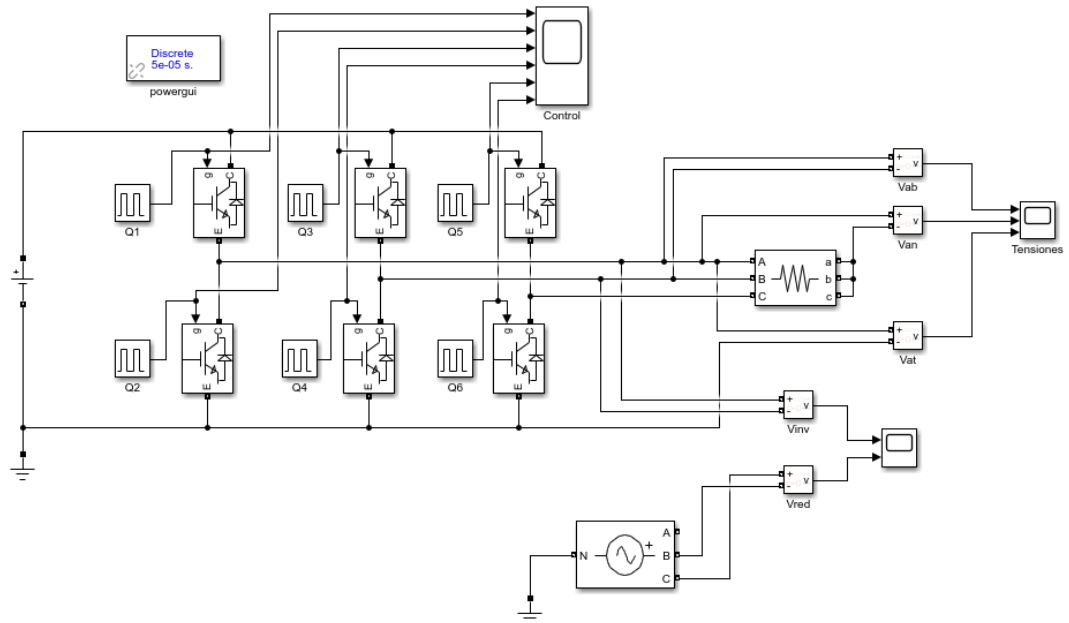


Figura 12: Simulación del puente de IGBTs. (Elaboración propia)

En esta primera simulación, se han introducido tres resistencias en estrella como carga de la red. También se ha incorporado un bloque generador de red trifásica, cuya onda ha de ser similar a la generada a la entrada de las resistencias.

En la Figura 13, se muestran las gráficas de la secuencia de conmutación de los IGBTs. De esta forma podemos comprobar que se cumple con lo establecido para crear las secuencias de conmutación que se realizaron en las Tablas 1 y 2.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

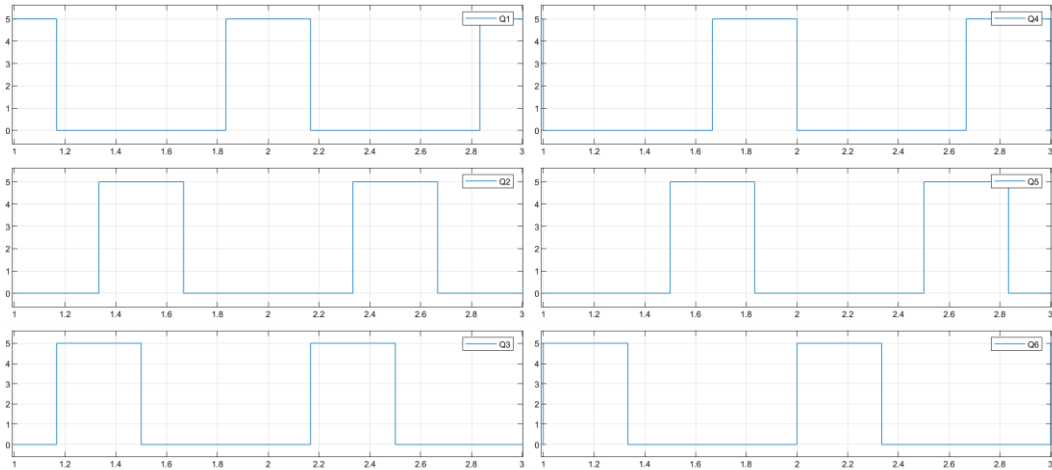


Figura 13: Simulación de las señales de control de los IGBTs. (Elaboración propia)

A continuación, comparamos las señales de tensión que hay entre dos fases de las resistencias y dos fases de la red generada. En la Figura 14, vemos que la frecuencia de la onda creada es la misma que la de la red y la tensión de pico también es la misma. Por tanto, podemos concluir que la señal de tensión a aplicar en las bobinas es correcta.

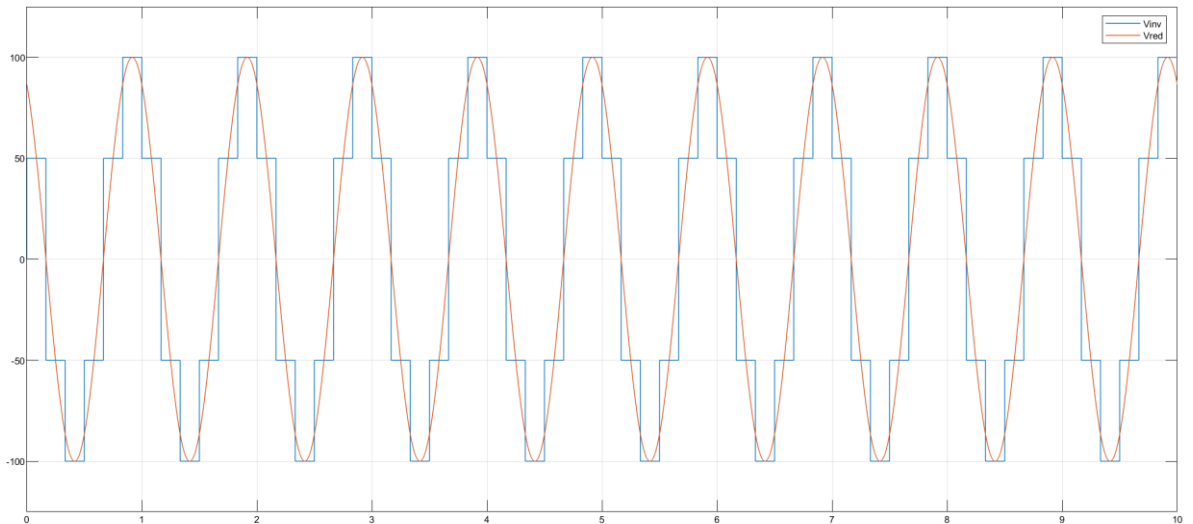


Figura 14: Señal de la red comparada con la señal de salida del puente. (Elaboración propia)

La siguiente simulación, representada en la Figura 15, se corresponde con las gráficas obtenidas al medir la tensión compuesta que aparece entre las fases a y b, en la segunda gráfica se representa la tensión simple que aparece entre la fase a y el neutro que aparece en la estrella de las resistencias y, finalmente, en la tercera gráfica, la tensión que hay entre la fase a y el terminal negativo de la alimentación en tensión continua del puente.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

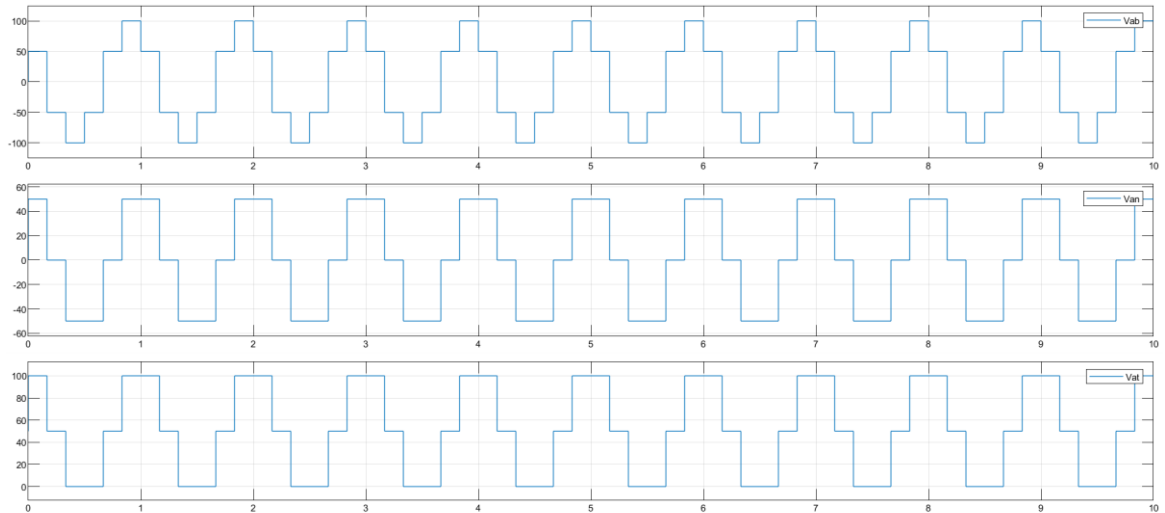


Figura 15: Simulación de tensiones compuestas y simples entre fases. (Elaboración propia)

Analizando la Figura 15, en la segunda gráfica, donde se muestra la tensión simple de la fase a, podemos ver que la onda generada siempre realiza la misma secuencia, que consiste en que durante dos pulsos de las señales de control está a tensión positiva, durante un pulso a tensión 0 V, durante otros dos pulsos a tensión negativa y, durante el sexto pulso vuelve a estar en 0 V. Esta secuencia se repite continuamente durante los seis pulsos de control de los IGBTs. La tensión máxima que aparece en la onda es de 50 V, aunque la alimentación se realiza a 100 V. Esto se debe a que cuando alimentamos el motor, lo hacemos entre dos fases, es decir, siempre se alimentan dos bobinas a la vez y, por ello, al medir entre la fase y el neutro (que es la estrella de las tres resistencias) solo aparece la mitad de la tensión de alimentación, al ser el neutro el punto central entre las dos resistencias alimentadas. Por otra parte, la tensión negativa en la onda producida aparece cuando se cambia a otro paso de la secuencia de control y, la tensión que aparece en la resistencia es negativa al estar invertidos los terminales de alimentación (vistos desde la propia resistencia).

Si nos fijamos en la tercera gráfica, que muestra la tensión entre una fase y el terminal negativo de la alimentación, las tensiones producidas en el puente van entre 0 V y 100 V.

Por último, en la primera gráfica, se representa la tensión compuesta entre dos fases, la a y la b. Esta gráfica, también se puede obtener al restar la tensión simple entre dos fases.

En la siguiente simulación realizada, cuyo esquema en Simulink se ve en la Figura 16, se simula el comportamiento del motor síncrono de imanes permanentes (de la librería Simscape) conectado a la señal de pulsos de corriente continua creados por el puente de IGBTs, es decir, en lugar de

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

emplear resistencias para la simulación, ahora empleamos la propia máquina para ver que funciona adecuadamente y poder medir corrientes y tensiones antes de trabajar con ella realmente.

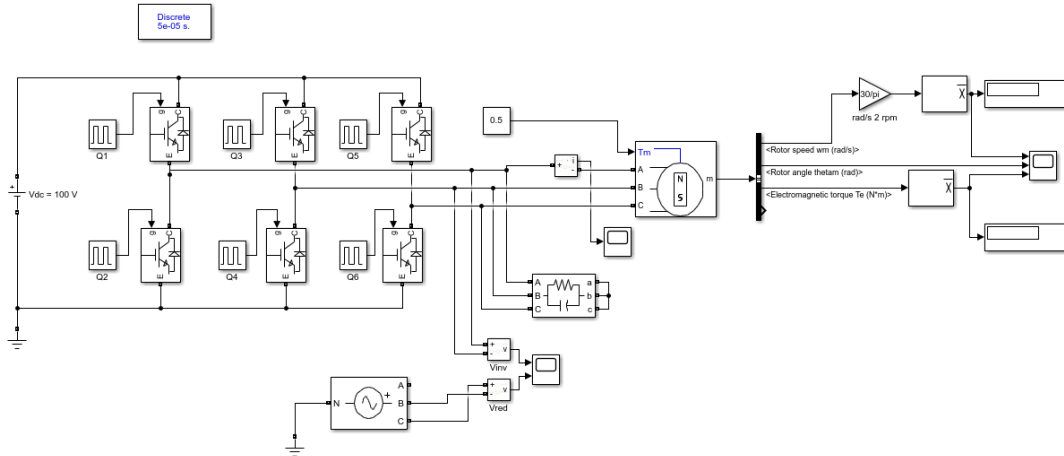


Figura 16: Simulación del funcionamiento del motor. (Elaboración propia)

Para esta simulación, se ha seleccionado un motor síncrono de imanes permanentes, con dos pares de polos en el rotor. Dicho motor está sometido a un par de 0.5 N*m, para asegurarnos de que tiene alguna carga conectada.

En la Figura 17, vemos la comparación entre la señal del generador de red y la señal de entrada en cada una de las fases.

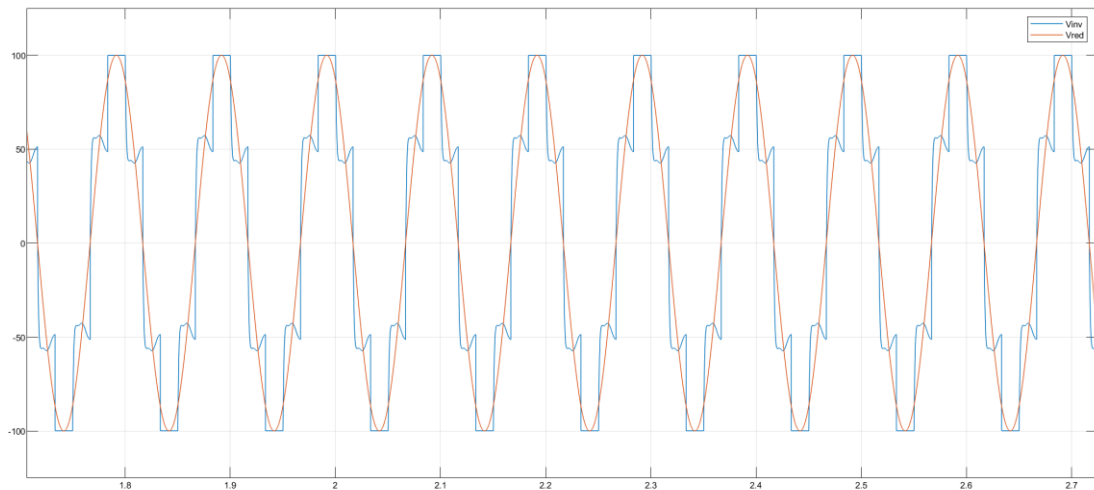


Figura 17: Señal de la red comparada con la señal de salida del puente. (Elaboración propia)

En esta gráfica, la onda de tensión que genera el puente de IGBTs, tiene una ligera distorsión respecto a la de la Figura 14, debido a que alimenta un motor ahora. Para evitar que la deformación sea mayor, se ha introducido un filtro que consta de una resistencia y un condensador conectados en estrella a la red, antes del motor.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

En la onda de tensión obtenida del puente, podemos ver que es de 10 Hz de frecuencia y, tiene una tensión de pico de 100 V. Por tanto, podemos comprobar que la velocidad a la que ha de girar el motor en este caso es:

$$n = 60 \cdot f / n^\circ \text{ pares de polos}$$

Para nuestro caso, la velocidad de giro del motor ha de ser $n = 60 \cdot 10 / 2 = 300$ r/min y, en la primera gráfica que aparece en la Figura 18, comprobamos que es cierto.

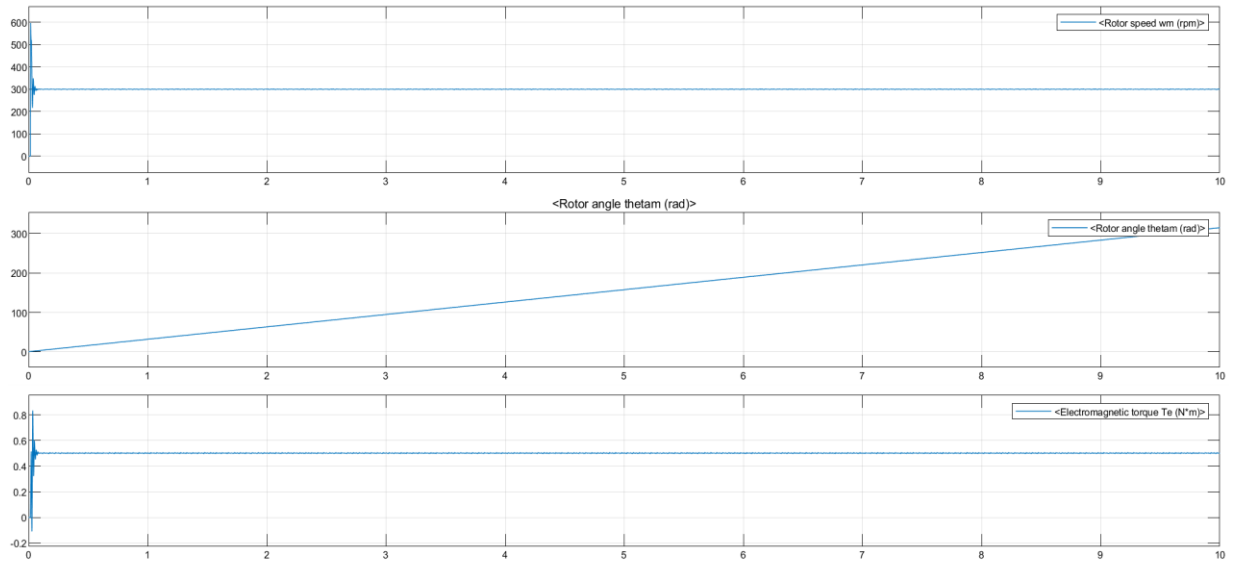


Figura 18: Resultados simulación del motor. (Elaboración propia)

En la segunda gráfica de la simulación, vemos el ángulo girado por el rotor, que, en este caso, parte de 0 rad en la posición inicial, hasta un ángulo final cuando se finaliza la simulación. Si nos fijamos en la pendiente de la gráfica, vemos que es constante y, esto se debe a que la frecuencia de la señal de alimentación es constante y por tanto la velocidad es constante durante toda la simulación.

Por último, en la tercera gráfica, se muestra el par al que está sometido el motor, que es constante en 0.5 N*m, que se corresponde con el valor que se introdujo en la entrada T_m al motor.

Finalmente, se visualiza en la Figura 19 la corriente que circula por la fase A del motor. De esta forma comprobamos que la forma de onda es similar a la de la tensión, también con ciertas distorsiones.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

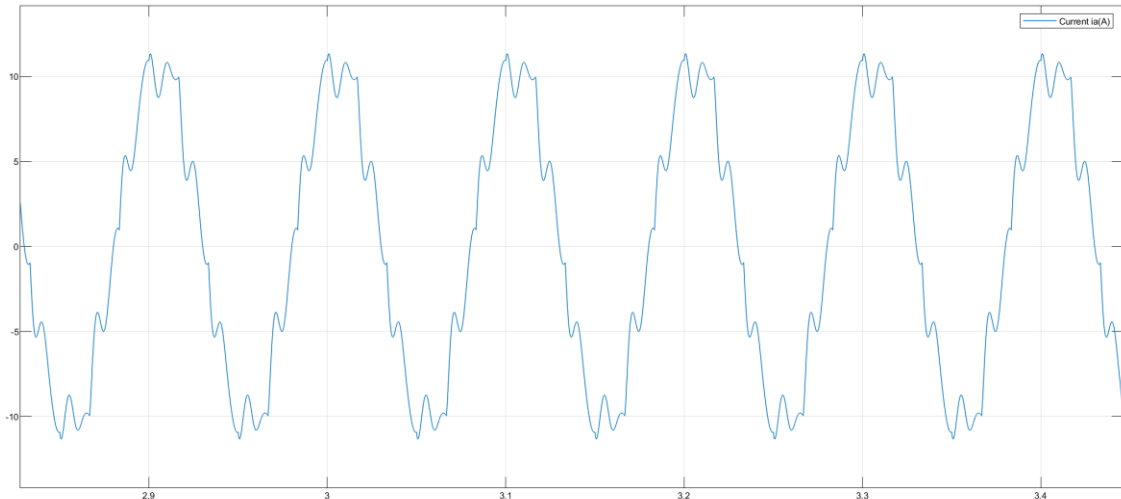


Figura 19: Corriente ia (A). (Elaboración propia)

3.2. Software y hardware empleado para el control.

En este apartado, vamos a decidir y describir de qué forma se realiza el control del motor y cómo se han de programar todos los equipos necesarios para llevar a cabo la finalidad de este trabajo.

Como unidad de control se va a emplear una Raspberry PI 4, en la cual se generarán las señales de control que, a su vez, se van a conectar al puente de IGBTs empleado, que será el circuito integrado IRAMY20UP60B.

También, se describirá el lenguaje de programación empleado, que va a ser LabVIEW 2020 de National Instruments, con la librería especial para conectarse a la Raspberry PI.

3.2.1. Raspberry PI 4.

La Raspberry PI es una placa de ordenador, con multiprocesador, de tamaño reducido, que incorpora su propio sistema operativo basado en Linux, llamado Raspbian. El modelo empleado en este trabajo cuenta con una memoria RAM de 2 GB.

Esta placa se puede conectar a internet a través de Wifi o por cable de red, de forma que pueda ser controlada desde un ordenador.

Cuenta con dos salidas HDMI para conectar a una pantalla, con 4 puertos USB (2 de ellos USB 3.0), un puerto de ethernet y una salida de sonido.

En la Figura 20 se muestran los 40 pines de conexión de salida/entrada que posee esta tarjeta y una breve descripción de qué es cada pin.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

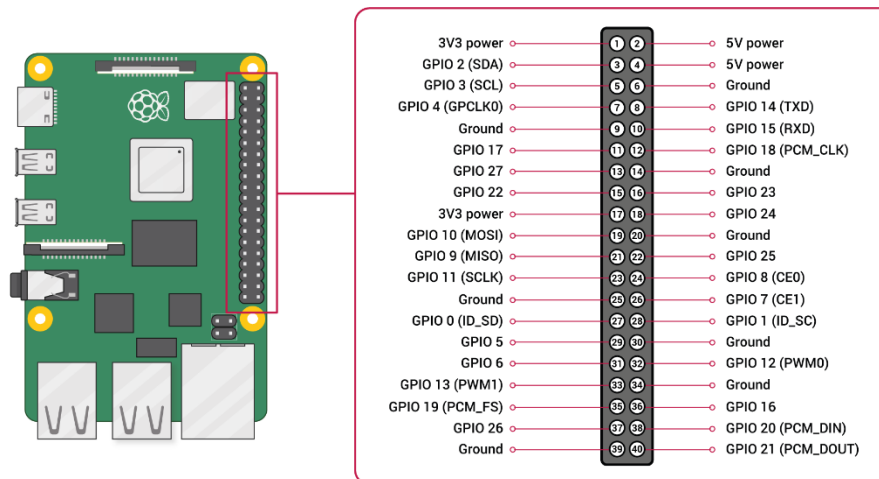


Figura 20: GPIO Raspberry Pi 4. [6]

En la aplicación empleada, necesitamos seis señales de control, las cuales parten de seis pines de la Raspberry que estén habilitados para ser señales de salida que, en este caso, serán pulsos de 3.3 V.

En el Anexo, se incluye una descripción más detallada de para qué sirve o que funciones realizan cada uno de estos pines, además de los valores de corriente y tensión máximos para las entradas/salidas.

3.2.2. Controlador con IRAMY20UP60B.

Como controlador del motor síncrono como un paso a paso, vamos a emplear el módulo IRAMY20UP60B, que consiste en un integrado en el que hay un puente de 6 IGBTs y un controlador de la señal de disparo de estos, cuyo esquema interno se ve en la Figura 21.

Aunque este tipo de circuitos se emplean en los inversores para pasar de corriente continua a alterna, nosotros solamente vamos a emplearlo para dar pulsos de corriente continua al motor siguiendo la secuencia adecuada para el desplazamiento del motor.

Las señales que controlan el disparo de cada uno de los IGBTs, según el catálogo del fabricante (que también se incorpora en el Anexo) pueden funcionar con una tensión de entrada de 3.3 V, como es la proporcionada por la Raspberry. Aun así, por una mayor seguridad de la tarjeta, se establecerá una separación eléctrica entre la Raspberry Pi y las entradas de control. Esta separación se realizará por medio de optoacopladores, cuya salida será de 5 V, que es un rango de tensión de entrada que también se encuentra entre los límites dados por el fabricante.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

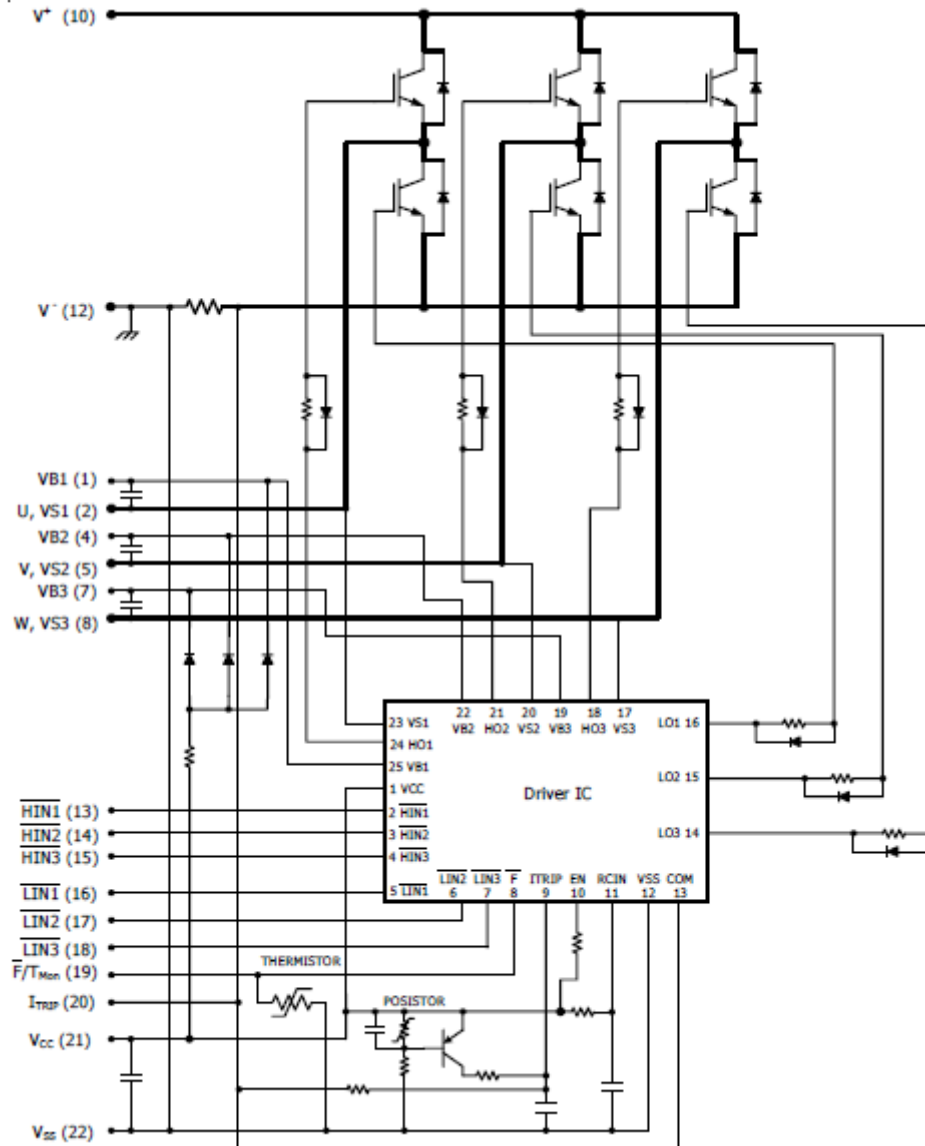


Figura 21: Circuito Integrado en el IRAMY20UP60B. [7]

La parte de control, teniendo en cuenta las especificaciones del catálogo del fabricante es capaz de soportar una corriente máxima de 20 A y una tensión de 600V, lo que le confiere a este controlador un carácter extremadamente compacto, además de un alto rendimiento.

Por otra parte, hay que destacar que este circuito incluye protecciones frente a sobrecorrientes y sobrecalentamientos, incorporando un monitor de temperatura. Para la disipación de calor, se instalará un radiador tanto en la parte delantera como trasera, contando este, además, con un ventilador.

A continuación, en la Figura 22, podemos ver los valores máximos de tensiones y corrientes que admite el circuito.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Parameter	Description	Value	Units
V_{CES} / V_{RRM}	IGBT/Voltaje de Bloqueo de Diodos	600	V
V^+	Voltaje de Entrada de Bus Positivo	450	
$I_O @ T_C=25^\circ C$	RMS Corriente de Fase (Note 1)	20	A
$I_O @ T_C=100^\circ C$	RMS Corriente de Fase (Note 1)	10	
I_O	Pulsed RMS Phase Current (Note 2)	40	
F_{PWM}	PWM Frecuencia de la Portadora	20	kHz
P_D	Potencia Disipada por IGBT @ $T_C = 25^\circ C$	68	W
V_{ISO}	Voltaje de Aislamiento(1min)	2000	V_{RMS}
T_J (IGBT & Diodos)	Rango de Temperatura de la Unión en Funcionamiento	-40 to +150	°C
T_J (Driver IC)	Rango de Temperatura de la Unión en Funcionamiento	-40 to +150	
T	Rango de Par de Montaje (Tomillo M4)	0.7 to 1.17	Nm

Nota 1: Modulación Sinusoidal a $V^+=400V$, $T_J=150^\circ C$, $F_{PWM}=20kHz$, Profundidad de Modulación=0.8, PF=0.6, Ver Figura 3.
Nota 2: $t_p < 100ms$; $T_C=25^\circ C$; $F_{PWM}=20kHz$. Limitado por $I_{BUS-TRIP}$, ver Tabla "Inverter Section Electrical Characteristics"

Figura 22: Valores Máximos del IRAMY20UP60B. [7]

Y, en la Figura 23, podemos ver los valores recomendados de tensión durante el funcionamiento.

Symbol	Definition	Min	Max	Units
$V_{B1,2,3}$	High side floating supply voltage	V_S+12	V_S+20	V
$V_{S1,2,3}$	High side floating supply offset voltage	Note 4	450	
V_{CC}	Low side and logic fixed supply voltage	12	20	V
V_{ITRIP}	I_{TRIP} input voltage	V_{SS}	$V_{SS}+5$	V
V_{IN}	Logic input voltage LIN, HIN	V_{SS}	$V_{SS}+5$	

Figura 23: Valores recomendados de tensión. [7]

En las Figuras 22 y 23 se muestran los valores más significativos, pero el resto de información contenida por el datasheet del componente se encuentra en el Anexo.

Otro aspecto importante de este controlador es que trabaja con lógica negativa, como se ve en la Figura 24. Este hecho hace que sea necesario modificar la Tabla 2, con la secuencia de conmutación de los IGBTs, porque ahora, las señales que salgan de la Raspberry PI han de ser adaptadas a la lógica negativa del controlador.

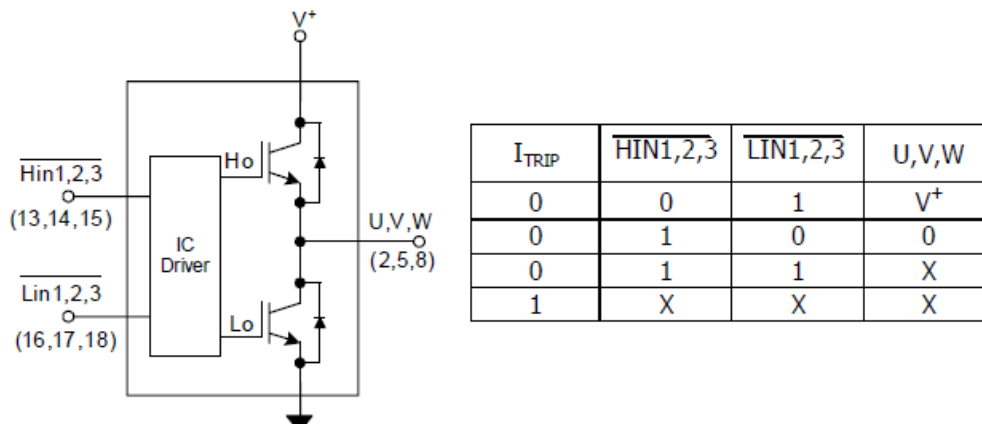


Figura 24: Tabla de verdad del controlador IRAMR20UP60B. [7]

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

A continuación, se muestra la nueva tabla (Tabla 3) de con la secuencia de conmutación de los IGBTs adaptada a la lógica negativa del controlador.

Posición\Transistor	HIN1	LIN1	HIN2	LIN2	HIN3	LIN3
1	0	1	1	1	1	0
2	1	1	0	1	1	0
3	1	0	0	1	1	1
4	1	0	1	1	0	1
5	1	1	1	0	0	1
6	0	1	1	0	1	1

Tabla 3: Secuencia de conmutación de los IGBTs (lógica negativa). (Elaboración propia)

3.2.3. LabVIEW.

Para realizar la programación y el control del motor, por medio de un ordenador, se va a emplear como herramienta de programación LabVIEW de National Instruments.

LabVIEW consiste en un lenguaje de programación visual gráfico, en el que por medio de bloques se construye el programa completo que, en este caso, se introducirá en la Raspberry PI para su ejecución.

Para poder utilizar LabVIEW en una Raspberry PI es necesario tener instalada la librería que ofrece National Instruments para controlar distintos tipos de tarjetas (Arduino, Raspberry PI, Beagle Bone, ...). Una vez instalada esta librería, se conecta la Raspberry PI a la misma red Wifi que el ordenador y se descarga a través de LabVIEW un archivo para que la Raspberry PI pueda ejecutar código de LabVIEW.

Una vez configurada la tarjeta correctamente, se procede a crear el código del programa correspondiente, es decir, crear un archivo .vi. Estos archivos constan de dos pantallas, una que es el panel frontal que se trata de la interfaz gráfica, en la cual, durante la ejecución del programa se pueden controlar los interruptores o controles disponibles en este y en la segunda pantalla, que es el diagrama de bloques, contiene el código programado, aunque en este caso, se trata de bloques por ser un programador en lenguaje gráfico.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

3.3. Programa de control.

En este apartado, vamos a describir en qué consiste el programa creado para controlar un motor síncrono de imanes permanentes como motor paso a paso. Para la realización de este programa se ha tomado como base el que ya se creó en el TFG “Control de una Máquina Síncrona como Motor Paso a Paso y Generador”². Tomando como base ese programa, se han realizado las modificaciones pertinentes para que sea compatible con la Raspberry PI, aunque, en este programa a desarrollar, su misión es comprobar que, con la nueva tarjeta empleada, y con los pines tomados, se puede controlar un motor y variar su velocidad. En el siguiente apartado, será en el que se realice el control a la vez de dos motores con características diferentes.

Para el funcionamiento del programa lo primero que hay que realizar es la conexión con la Raspberry PI. Este proceso se realiza abriendo la aplicación de “NI LabVIEW 2020 SP1 (32-bit)”, que es la versión empleada para la realización del programa.

A continuación, vamos a Tools > Hobbyist > Target Configuration, como se muestra en la Figura 25.

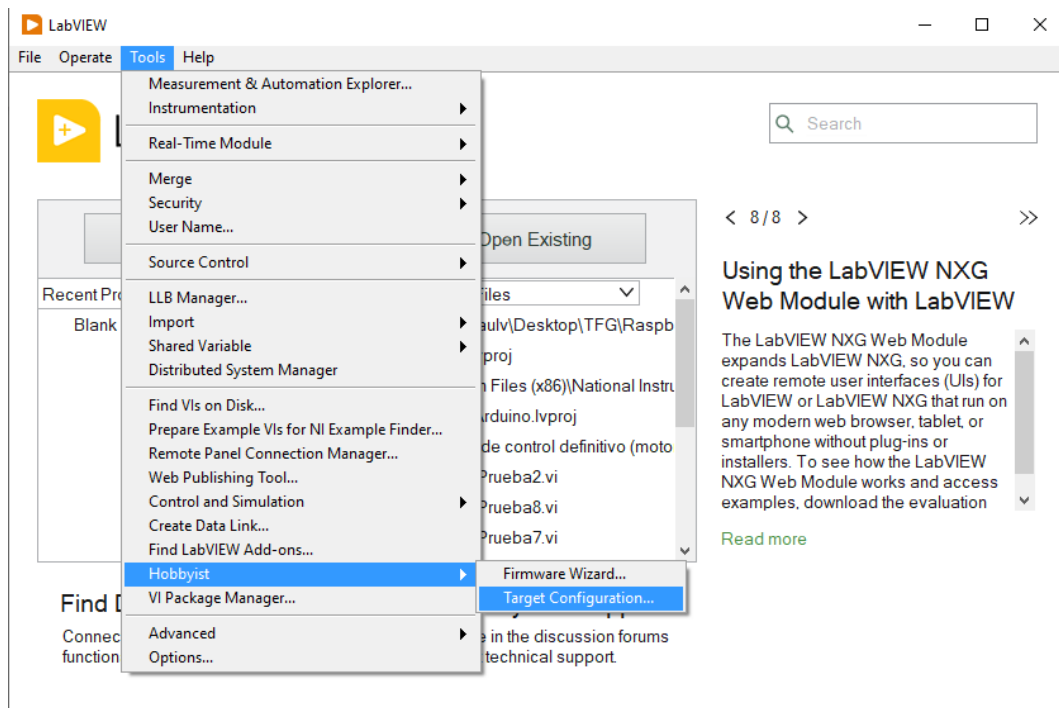


Figura 25: Configuración de la Raspberry PI. (Elaboración propia)

El siguiente paso, es establecer la conexión con la Raspberry PI, cuyo nombre a buscar en la red será *raspberrypi*, y para acceder a ella, se

² Citado a partir del trabajo fin de grado “Control de una Máquina Síncrona como Motor Paso a Paso y Generador” [5]

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

introducirá un usuario: *pi* y una contraseña: *raspberry*. En la Figura 26, se ilustra el proceso de conexión del ordenador PC con la Raspberry PI mediante comunicación online.

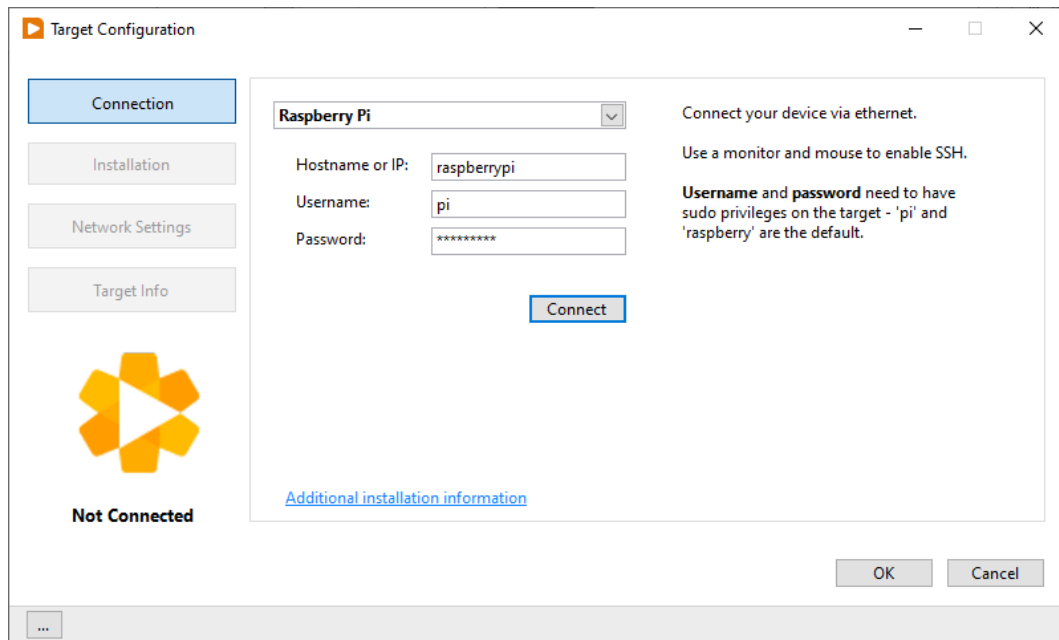


Figura 26: Conexión del ordenador a la Raspberry PI online. (Elaboración propia)

Una vez conectada y funcionando en línea la Raspberry PI, se comienza a construir el programa, para ello, se crea un archivo *.vi* en el que se abren dos pantallas (como se comentó en el punto anterior) que son el panel frontal y el diagrama de bloques.

En el diagrama de bloques lo primero que hacemos es crear una estructura llamada “Flat Sequence”, con tres ventanas. Este bloque permite realizar la ejecución del programa en un orden adecuado y así, de esta forma, cuando el programa se ejecuta desde el ordenador, lo primero que realiza es la comunicación con la Raspberry PI, después de esto, se pasa a la segunda ventana de la estructura, donde se encuentra un bucle “while loop” que se ejecuta continuamente hasta que ocurre algún problema en la comunicación con la Raspberry PI o si el usuario pulsa el botón de STOP que hay en el menú del panel frontal. Y después, una vez finalizado este bucle, se pasa a la tercera ventana de la estructura Flat Sequence, que consiste en cerrar la comunicación del ordenador con la Raspberry PI y a la vez, poner el botón de STOP en la posición de false, es decir, no pulsado, para evitar que al iniciar la siguiente vez el programa finalice inmediatamente por estar aún pulsado este. Una vez finalizadas todas estas acciones, finaliza la ejecución del programa.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

En la Figura 27, se muestra un pequeño recorte de cómo va a distribuirse la estructura Flat Sequence anteriormente descrita y la comunicación con la Raspberry PI.

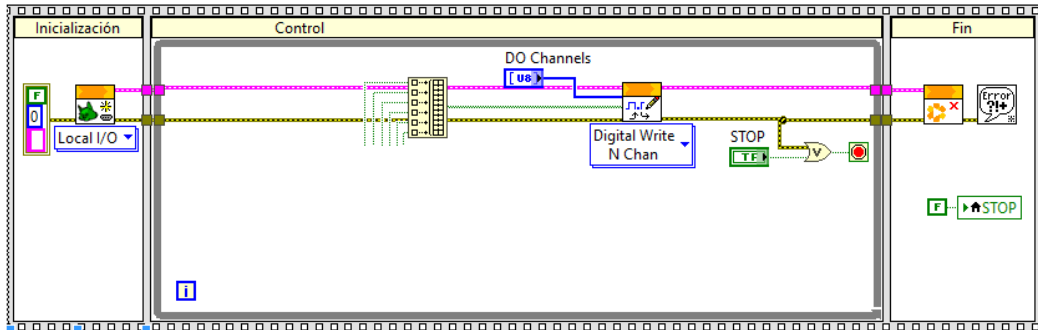


Figura 27: Recorte del diagrama de bloques. (Elaboración propia)

A continuación, nos centraremos en el programa que se encuentra en la ventana de control. Aquí encontraremos el bucle while loop que, como antes se comentó, se repite de manera indefinida hasta que se provoca la interrupción, bien porque lo decide el usuario o bien porque se produce algún fallo. Dentro de este bucle va a haber dos bloques principales, uno encargado de crear la secuencia de pulsos que van al controlador del motor y otro, que será el encargado de controlar la frecuencia de giro del motor, es decir, un variador de frecuencia que permite modificar la velocidad del motor sin que este se bloquee por realizar un cambio brusco en la frecuencia de la señal de alimentación.

- Secuencia de pulsos:

Este bloque llamado secuencia de pulsos, cuyo código se puede ver en la Figura 28, es el encargado de variar la señal de los pines de salida, cada vez que se ejecuta el bucle while loop. Para conocer en qué estado se encontraba la señal de salida, se ha creado un bucle de realimentación, de forma que, si antes la señal de salida estaba en la posición dos, cuando se vuelve a ejecutar el bucle while, si se cumplen todas las condiciones de las entradas para que el motor siga avanzando, cambiará la salida a posición 3 y, esto provocará que se activen las salidas de la tarjeta correspondientes a esta posición.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

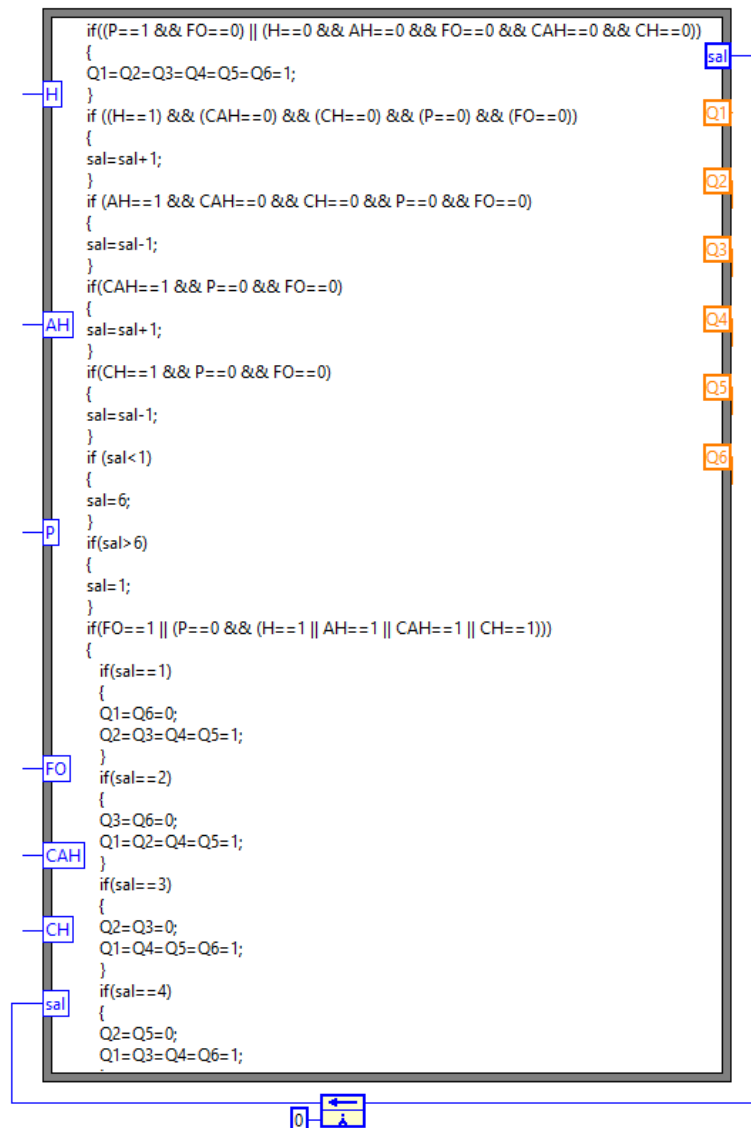


Figura 28: Bloque de secuencia de pulsos. (Elaboración propia)

Como podemos ver, este bloque cuenta con seis entradas, además de la entrada correspondiente a la realimentación de la posición de salida. Estas entradas se corresponden con el giro horario, el giro antihorario, el freno, dejar el motor enclavado cuando está parado, el cambio de giro horario a giro antihorario y el cambio de giro antihorario a giro horario.

Por la otra parte, como salidas, tenemos la posición en la que se encuentra la salida, que va a la realimentación y las seis salidas del control del motor. En la Figura 29, se ve cómo se convierten estas salidas a un valor booleano, es decir, mediante un bloque comparador, se comprueba si la salida es igual a 1, si esto se cumple, la señal de salida del bloque es un 1 lógico, si no se cumple, será un 0 lógico. Estas señales booleanas son las que se envían a la Raspberry PI para controlar los pines asignados a cada una de las salidas.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

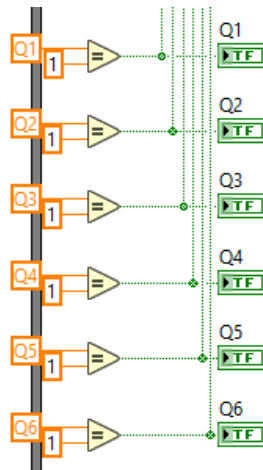


Figura 29: Señales de salida del bloque de pulsos. (Elaboración propia)

En las entradas, una parte muy importante consiste en que, si el motor se activa en giro horario, no se permita un cambio brusco a giro antihorario, puesto que esto puede provocar un sobrecalentamiento del motor, llegando incluso a quemarse o estropearse. En la Figura 30, se muestra el esquema establecido para evitar que una situación así pueda ocurrir.

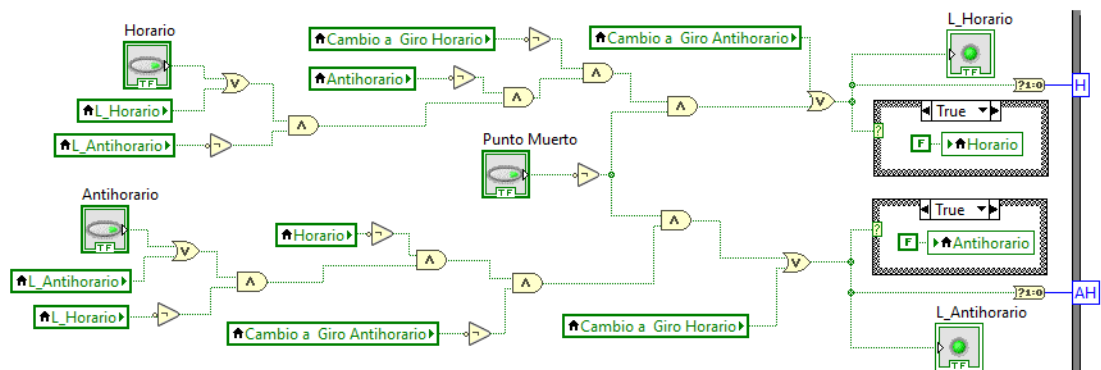


Figura 30: Entradas de Giro Horario y Giro Antihorario. (Elaboración propia)

Se ha elaborado un esquema de control marcha paro, con accionamiento en giro horario o antihorario. Para que se pueda dar el arranque en giro horario, es necesario que o bien se active el pulsador de giro horario, o que este ya haya sido activado y la entrada del bloque este activa (y, por tanto, esté luciendo el indicador L_Horario). Además de cumplirse cualquiera de estas dos condiciones, ha de cumplirse que no esté activada la señal de giro antihorario del bloque (no ha de lucir el indicador L_Antihorario). Además de esta condición, ha de cumplirse que no estén tampoco pulsadas ni la tecla de arranque en giro antihorario, ni la tecla de punto muerto, ni que se esté produciendo un cambio a giro horario. Cuando está activada la señal de Cambio a Giro Antihorario, también se activa esta señal mientras el motor sigue girando en sentido horario desde su velocidad hasta 0.

En el caso de la señal de entrada de giro antihorario, el funcionamiento es el mismo que en el caso anterior. Es necesario que o bien se active el

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

pulsador de giro antihorario, o que este ya haya sido activado y la entrada del bloque este activa (y, por tanto, esté luciendo el indicador L_Antihorario). Además de cumplirse cualquiera de estas dos condiciones, ha de cumplirse que no esté activada la señal de giro horario del bloque (no ha de lucir el indicador L_Horario). Además de esta condición, ha de cumplirse que no estén tampoco pulsadas ni la tecla de arranque en giro horario, ni la tecla de punto muerto, ni que se esté produciendo un cambio a giro antihorario. Cuando está activada la señal de Cambio a Giro Horario, también se activa esta señal mientras el motor sigue girando en sentido antihorario desde su velocidad hasta 0.

También podemos ver, que cuando se activa alguna de las dos señales descritas, se desactiva la tecla pulsada. Esto sucede porque están configuradas como interruptores y para evitar que se queden pulsadas, se desactivan ellas automáticamente al activarse la señal.

Las secuencias empleadas para girar el motor en un sentido o en otro, son siempre la misma, lo que pasa, es que en giro horario se hace ir 1-2-3-4-5-6 y en sentido antihorario, va 6-5-4-3-2-1. De esta forma, se varía el giro del motor, al variar los pulsos y no es necesario ningún cambio en el conexionado eléctrico del motor.

Las siguientes entradas del bloque, como se ve en la Figura 31, corresponden a la señal de paro, la de frecuencia igual a cero y las de cambio a giro horario y cambio a giro antihorario.

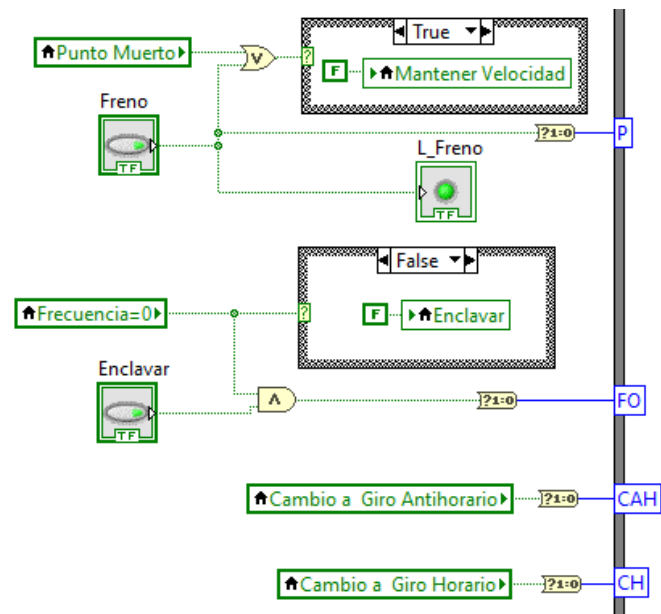


Figura 31: Señales de entrada al bloque de pulsos. (Elaboración propia)

La señal de paro P, se activa cuando la tecla de freno está activada, de forma que detiene el avance del motor, dejando a nivel 1 lógico todas las

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

salidas de la Raspberry PI. De esta forma, se desactivan todos los IGBTs, ya que emplean lógica negativa.

La entrada FO, se activa cuando la frecuencia del motor es 0, es decir, está parada y el variador le indica velocidad 0 y, además, también se activa la tecla de enclavamiento. Esta acción, hace que se quede activada la última posición de pulsos de salida, quedando el motor fijo en esa posición.

También observamos que, si las teclas del Punto Muerto o el Freno están activadas, se imposibilita la opción de pulsar la tecla mantener velocidad. Y, asimismo, si la señal de Frecuencia = 0 tiene un valor de cero lógico, no se puede dejar el motor enclavado y se imposibilita tocar la tecla de enclavamiento.

Por último, están las funciones de cambio a giro horario y cambio a giro antihorario, que, cuando se activan, provocan que la secuencia de pulsos vaya cambiando hasta conseguir el cambio de giro.

En la Figura 32, vemos que si se cumple que la señal de freno o de Frecuencia = 0 o de Punto Muerto están a nivel 1 lógico, tanto la tecla de giro Horario como de giro Antihorario, se desactivan, si es que estuvieran pulsadas.

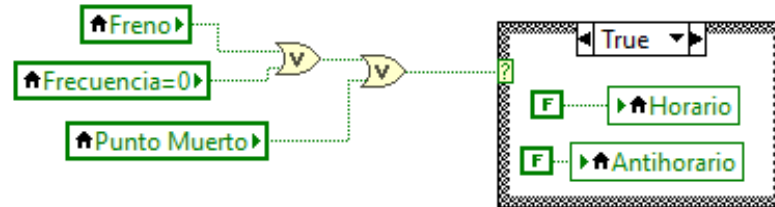


Figura 32: Señales de control del motor. (Elaboración propia)

En resumen, en la Tabla 4, se muestra la descripción breve de cada una de las entradas y salidas del bloque de creación de secuencia de pulsos.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

Nombre de la variable	Tipo de variable	Función
H	Entrada	Crea la secuencia de pulsos para el giro horario.
AH	Entrada	Crea la secuencia de pulsos para el giro antihorario.
P	Entrada	Detiene el avance del motor, dejando todas las salidas activadas.
FO	Entrada	Cuando se activa esta entrada, deja el motor enclavado en la última posición.
CAH	Entrada	Activación del cambio de giro a sentido antihorario.
CH	Entrada	Activación del cambio de giro a sentido horario.
sal	Entrada/ Salida	Posición en la que se encuentra la secuencia de pulsos.
Q1	Salida	Valor de la señal de salida a HIN1
Q2	Salida	Valor de la señal de salida a HIN2
Q3	Salida	Valor de la señal de salida a HIN3
Q4	Salida	Valor de la señal de salida a HIN4
Q5	Salida	Valor de la señal de salida a HIN5
Q6	Salida	Valor de la señal de salida a HIN6

Tabla 4: Variables de entrada y salida del bloque de secuencia. (Elaboración propia)

- Variador de frecuencia:

Para realizar un control de la velocidad del motor, es necesario controlar con qué periodo de tiempo se repite el bucle while loop. Para ello, se emplea otro bloque llamado “Wait Until Next ms Multiple” (Figura 33) que tiene una entrada con el número de milisegundos que está detenida la ejecución del programa hasta que vuelva a ejecutarse el bucle while loop. De esta forma se puede controlar la velocidad a la que gira el motor y, también variarla. El principal problema ocurre durante esta variación, debido a que, si se realiza un cambio de frecuencia muy brusco en el motor, éste se puede llegar a quedar bloqueado al perder la secuencia de pasos que le corresponde. Para evitarlo, se ha creado este bloque llamado variador de frecuencia, que se muestra en la Figura 34 e incorpora un código que hace una variación de frecuencia más lenta de forma que no se bloquee el motor.

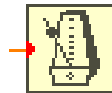


Figura 33: Bloque Wait Until Next ms Multiple. (Elaboración propia)

En la Figura 34, podemos ver un fragmento del código creado para el bloque variador de frecuencia y, también las entradas y salidas que serán necesario conectar, e incluso, realimentar.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

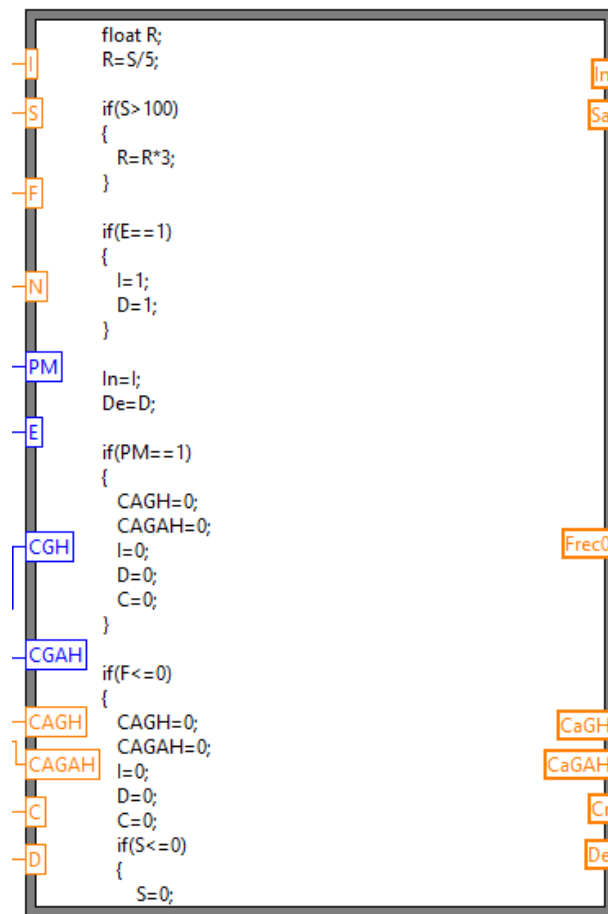


Figura 34: Bloque Variador de Frecuencia. (Elaboración propia)

En la Figura 35, podemos ver cómo se realiza el conjunto de conexiones de las entradas y salidas a los diversos bloques empleados. Entre ello, destaca que tenemos seis lazos de realimentación, que se inician con un valor de 0. A continuación, vamos a describir en qué consiste cada uno de estos lazos y cuál es su misión.

- Lazo de realimentación entre In – I: El valor numérico que puede tomar este lazo es 0 o 1. Toma el valor 1, cuando queremos hacer un cambio de sentido de giro del motor. Como el valor está expresado en una representación tipo double, se coloca un bloque de comparación con una constante igual a 1, de forma que, si es cierto, la salida del bloque será un 1 lógico, encendiéndose una señal en el panel frontal para advertir que el motor está cambiando el sentido de giro.
- Lazo de realimentación entre Sa – S: Este lazo de realimentación, es el encargado de que la frecuencia de la señal de alimentación del motor vaya variando poco a poco hasta alcanzar el valor establecido como consigna y evitar de esta forma que el motor se bloquee. El valor de salida de este bloque es el que se emplea para

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

alimentar el bloque de periodo de ejecución del bucle. Se multiplica por 1000 ya que el bloque requiere de un valor en milisegundos.

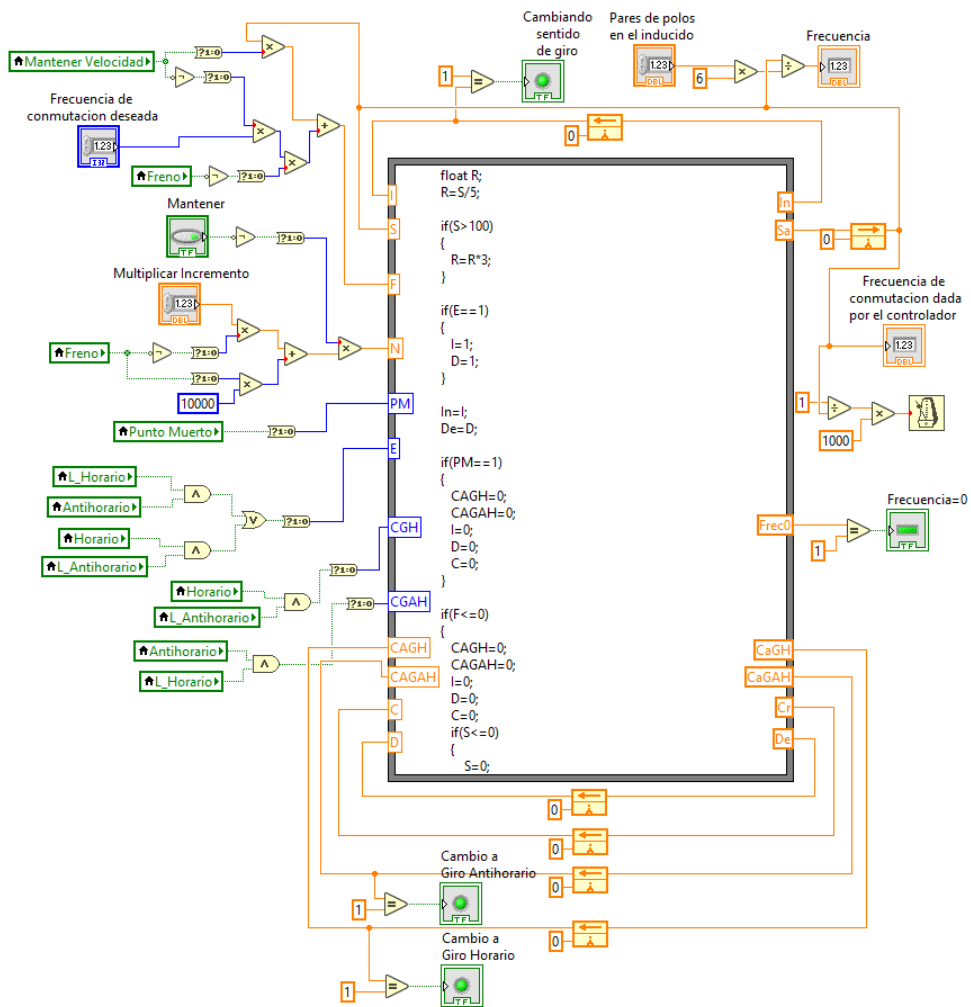


Figura 35: Conexión de entradas y salidas al bloque variador de frecuencia. (Elaboración propia)

- Lazo de realimentación entre CaGH – CAGH: Este lazo es el encargado de realimentar la señal de cambio a giro horario. Si no se ha solicitado por parte del usuario ningún cambio de giro, tiene valor 0, pero cuando se hace un cambio de giro a sentido horario, se pone a valor 1 (al activarse la entrada CGH). Solamente tiene el valor 1 mientras el motor sigue girando en sentido antihorario hasta llegar a velocidad 0, desactivándose en ese momento. También controla la variable “Cambio a Giro Horario”, empleada en otras partes del programa.
- Lazo de realimentación entre CaGAH – CAGAH: Este lazo es el encargado de realimentar la señal de cambio a giro antihorario. Si no se ha solicitado por parte del usuario ningún cambio de giro, tiene valor 0, pero cuando se hace un cambio de giro a sentido antihorario, se pone a valor 1 (al activarse la entrada CGAH). Solamente tiene el

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

valor 1 mientras el motor sigue girando en sentido horario hasta llegar a velocidad 0, desactivándose en ese momento. También controla la variable “Cambio a Giro Antihorario”, empleada en otras partes del programa.

- Lazo de realimentación entre Cr – C: El valor de este lazo puede ser de 0 o 1, y se activa cuando el lazo In – I está a valor 1, es decir, el motor está realizando un cambio de giro. Se mantiene activada desde el momento en que el motor llega a velocidad 0, hasta que el motor alcanza la frecuencia de giro en el nuevo sentido establecido.
- Lazo de realimentación entre De – D: Este lazo de realimentación, al igual que los anteriores varía entre un valor 0 y 1. Toma el valor de 1, durante todo el proceso de cambio de giro del motor, es decir, desde que comienza a disminuir la frecuencia de giro en un sentido, hasta que poco a poco se va alcanzando la misma frecuencia de giro, pero en el otro sentido.

Además de los lazos de realimentación, el bloque también cuenta con otras seis entradas y una salida más, que vamos a describir a continuación.

- Entrada F: Por esta entrada, se introduce la frecuencia de conmutación de los polos deseada, salvo que se hayan pulsado la tecla de mantener velocidad, que mantiene la frecuencia que lleva en ese momento el variador o, el freno, que pone a 0 la frecuencia de entrada.
- Entrada N: Esta entrada es por la cual se introduce la señal de Multiplicar Incremento, que consiste en una variable numérica que es la encargada de regular con que “velocidad” realiza el cambio de frecuencia el variador. Sería como ajustar la rampa que llevan los variadores de frecuencia. En caso de que se active el freno, este valor pasa a valer 10000, para que se produzca una detención rápida del giro del motor.
- Entrada PM: Esta entrada avisa al bloque variador de frecuencia de que se ha pulsado el botón punto muerto y, por tanto, detiene en ese instante el giro del motor.
- Entrada E: En esta entrada, se realiza una comprobación de si se ha pulsado uno de los botones de giro del motor para hacer un cambio del sentido de rotación. En caso de que se vaya a realizar el cambio de giro, la entrada es 1, en caso contrario, será 0.
- Entrada CGH: Esta entrada sirve para indicar al bloque variador de frecuencia que se quiere realizar un cambio de giro a sentido horario.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

- Entrada CGAH: Esta entrada sirve para indicar al bloque variador de frecuencia que se quiere realizar un cambio de giro a sentido antihorario.
- Salida Frec0: Esta salida, activa una variable booleana, cuando la frecuencia introducida en el bloque variador y la frecuencia de salida son 0.
- Salida Sa: Esta salida, va conectada al lazo de realimentación de Sa - S, como ya se comentó anteriormente, pero, además, de esta variable se extrae el valor de la frecuencia de conmutación de los polos, que será empleado para controlar cada cuanto se repite el bucle while loop.

A continuación, para concluir con la explicación del bloque variador de frecuencia, se puede visualizar en la Figura 36 el código que lleva este bloque.

```

float R;
R=S/5;

if(S>100)
{
R=R*3;
}

if(E==1)
{
l=1;
D=1;
}

ln=l;
De=D;

if(PM==1)
{
CAGH=0;
CAGAH=0;
l=0;
D=0;
C=0;
}

if(F<=0)
{
CAGH=0;
CAGAH=0;
l=0;
D=0;
C=0;
if(S<=0)
{
S=0;
}
if(S>0 && S<=10)
{
S=S-(0.1*N);
}
if(S>10 && S<=30)
{
S=S-(0.05*N);
}
if(S>30)
{
S=S-((0.025*N)/R);
}
}

if(F>0 && l==1)
{
if(D==1 && C==1)
{
CAGH=0;
CAGAH=0;
if(S>=F)
{
S=F;
l=0;
D=0;
C=0;
}
if(S>30 && S<F)
{
S=S+((0.025*N)/R);
}
if(S>10 && S<=30 && S<F)
{
S=S+(0.05*N);
}
if(S<=10 && S<F)
{
S=S+(0.1*N);
}
}
}
if(D==1 && C==0)
{
if(CGH==1)
{
CAGH=1;
}
if(CGAH==1)
{
CAGAH=1;
}
if(S<=0)
{
S=0;
C=1;
}
if(S>0 && S<=10)
{
S=S-(0.1*N);
}
if(S>10 && S<=30)
{
S=S-(0.05*N);
}
if(S>30)
{
S=S-((0.025*N)/R);
}
}
ln=l;
De=D;
}

if(F>0 && l==0)
{
if(F>S)
{
if(S>30 && S<F)
{
S=S+((0.025*N)/R);
}
if(S>10 && S<=30 && S<F)
{
S=S+(0.05*N);
}
if(S<=10 && S<F)
{
S=S+(0.1*N);
}
}
if(F<=S)
{
S=F;
}
if(F<S)
{
if(S>0 && S<=10 && S>F)
{
S=S-(0.1*N);
}
if(S>10 && S<=30 && S>F)
{
S=S-(0.05*N);
}
if(S>30 && S>F)
{
S=S-((0.025*N)/R);
}
}
if(F>=S)
{
S=F;
}
if(S<=0)
{
S=0;
}
}
}

if(S==0 && F==0)
{
Frec0=1;
}
if(Sl=0)
{
Frec0=0;
}
CaGH=CAGH;
CaGAH=CAGAH;
Cr=C;
Sa=S;
}

```

Figura 36: Código del bloque variador de frecuencia. [5]

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Por último, se ha incorporado un marcador gráfico y otro numérico de la velocidad del motor (en r/min), para que se pueda visualizar desde el panel frontal, cuyo esquema de bloques se muestra en la Figura 37.

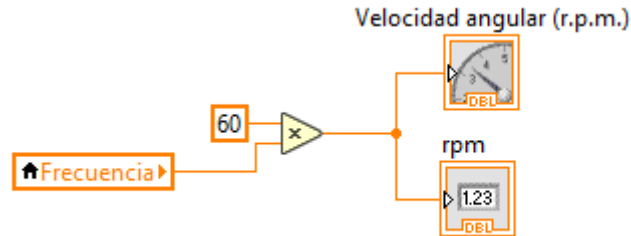


Figura 37: Visualización de la velocidad en r/min. (Elaboración propia)

Una vez elaborado el diagrama de bloques, pasamos a ordenar todos los controles que hemos creado en el panel frontal de la aplicación, que serán con los que controlemos y veamos los valores de la velocidad del motor.

En la Figura 38, se muestra una captura de pantalla del resultado final del panel frontal que emplearemos en esta primera prueba.

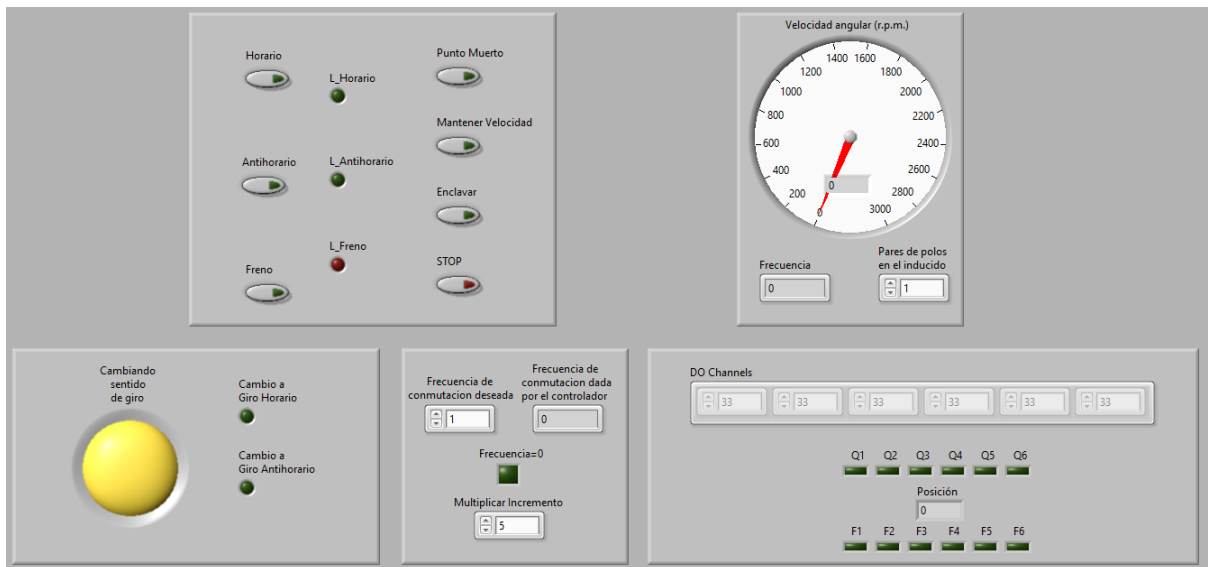


Figura 38: Panel Frontal. (Elaboración propia)

Y, en la Figura 39, se encuentra el diagrama de bloques completo empleado en este primer programa utilizado para comprobar el funcionamiento del motor mientras es controlado por la Raspberry PI.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

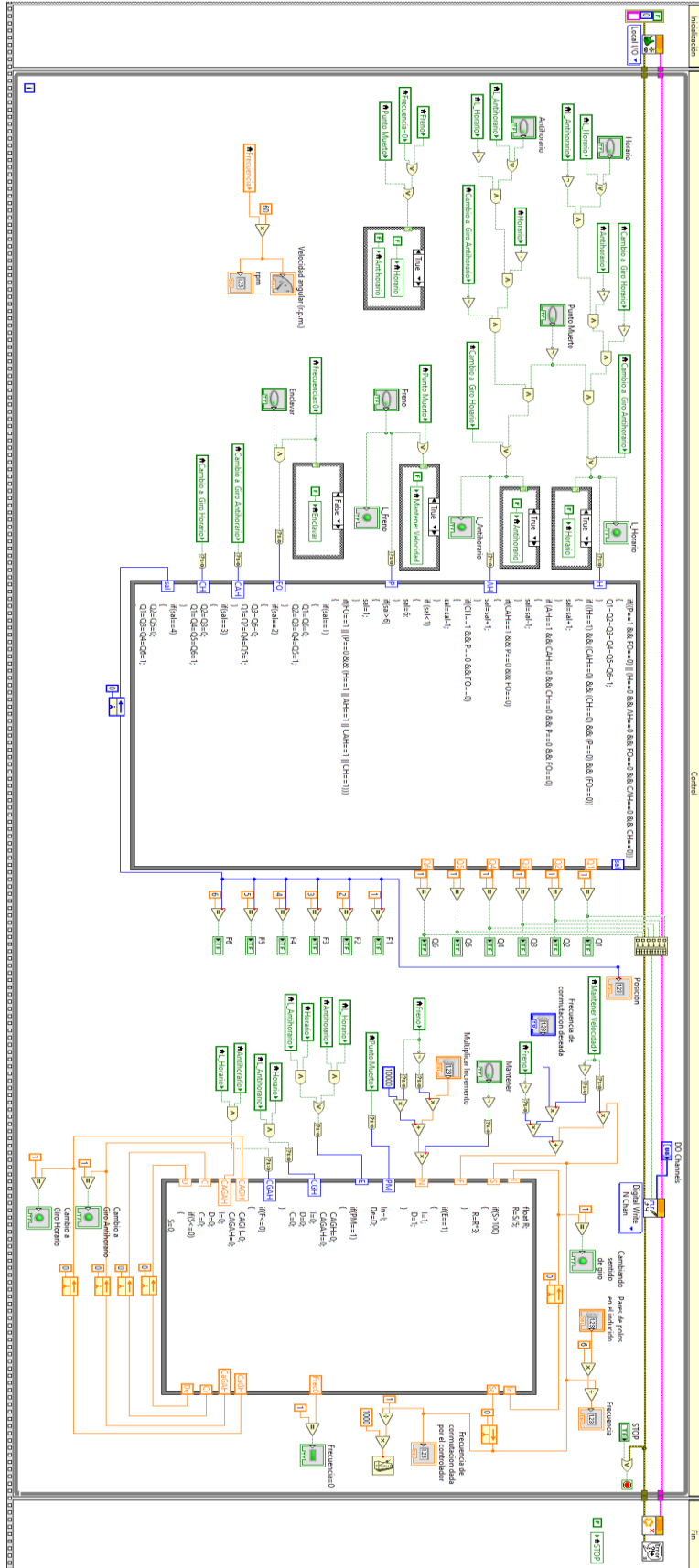


Figura 39: Diagrama de Bloques. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

3.4. Análisis de los resultados obtenidos.

Una vez creado el programa de control que se cargará en la Raspberry PI para realizar el control de la velocidad del motor, vamos a comprobar que funciona correctamente. Para ello, lo primero que realizamos es una comprobación de que las señales de disparo de cada uno de los IGBTs. Para ello, empleamos un osciloscopio, y el resultado obtenido, que se puede ver en la Figura 40, permite ver que el funcionamiento es el adecuado.

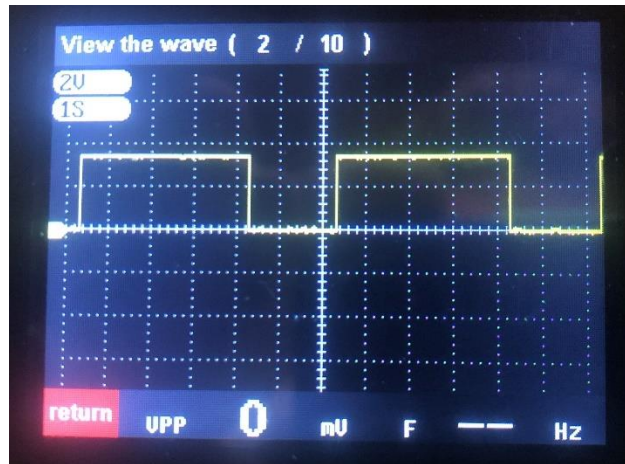


Figura 40: Señal de disparo. (Elaboración propia)

La señal visualizada en la Figura 40, se corresponde con una de las señales de disparo definidas en la Tabla 3. Como se puede apreciar en la imagen, es una señal de lógica negativa, en la que está a nivel alto (IGBTs desactivados) durante 4/6 del período de la señal y después, durante 2/6 del período, se mantiene a nivel bajo, disparando el IGBT. De esta manera, hemos comprobado que las señales de disparo funcionan correctamente.

En el siguiente paso, vamos a realizar una recreación del IRAMY20UP60B, pero con 6 transistores de tipo NPN cableados de forma adecuada, como se muestra en el esquema del propio circuito integrado. En el caso del motor, para simular éste componente, se van a emplear tres resistencias de 1 k Ω colocadas en estrella. El estator de un motor síncrono se compone de tres bobinas, a las que, si medimos su impedancia, tienen una pequeña parte resistiva y otra parte inductiva, por ello, en nuestra prueba reducimos el estator a unas resistencias en estrella y tomamos medidas.

En la Figura 41, se muestra cómo quedó esta aproximación del circuito final. Esta prueba se ha montado sobre una placa de prácticas y, para realizar las medidas, se han empleado las entradas analógicas de una placa Arduino.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

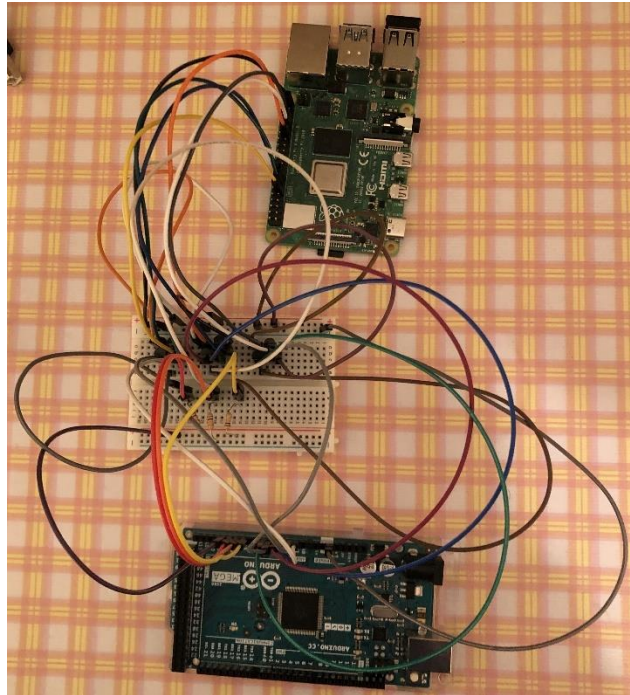


Figura 41: Circuito de aproximación al IRAMY20UP60B. (Elaboración propia)

En las siguientes figuras, vamos a analizar las respuestas de las gráficas obtenidas:

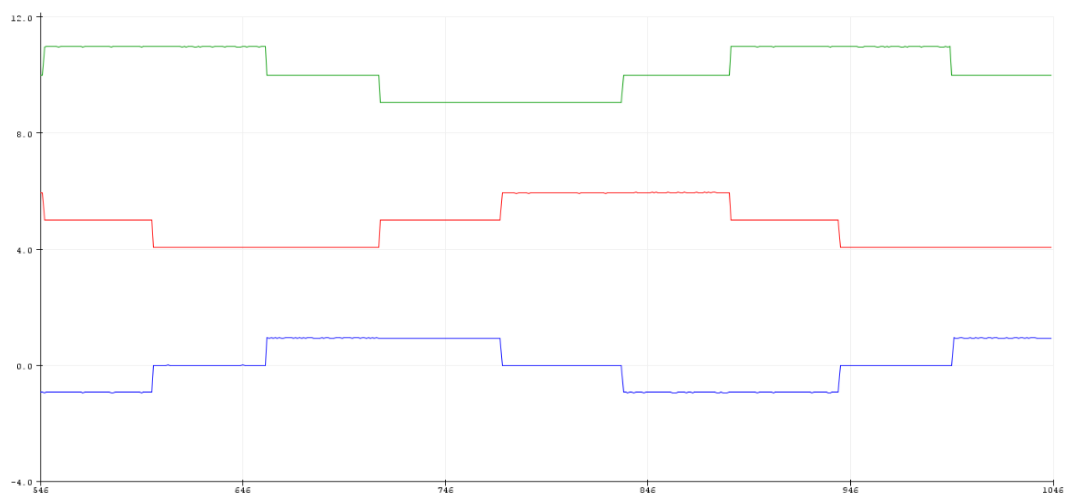


Figura 42: Tensión fase - neutro en giro antihorario. (Elaboración propia)

En la Figura 42, podemos ver la señal de tensión entre una de las fases y la estrella de las resistencias (que sería el neutro del sistema) cuando se activa el giro en sentido antihorario.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

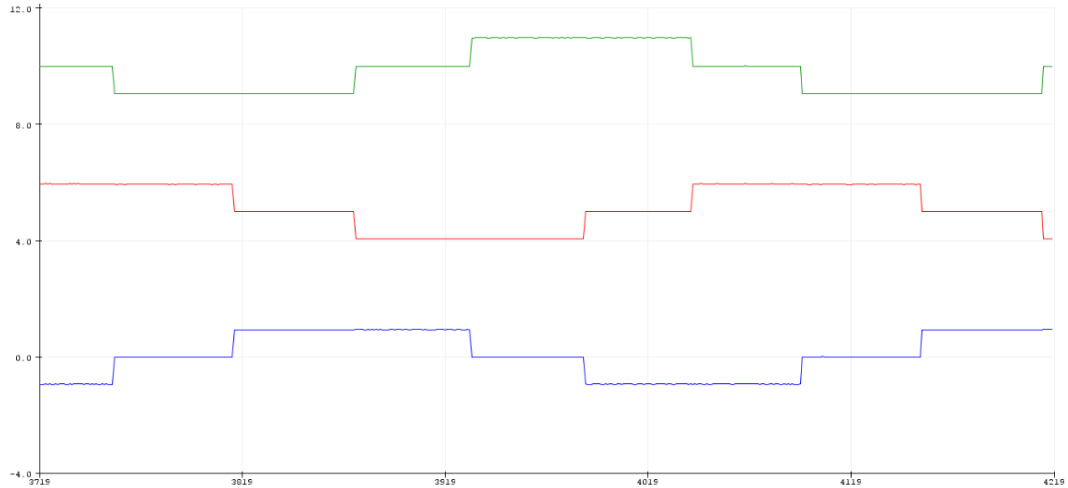


Figura 43: Tensión fase - neutro en giro horario. (Elaboración propia)

En la Figura 43, se visualiza también la tensión fase neutro en un instante, cuando se activa el modo de giro horario. En esta figura, vemos que las dos primeras señales están cambiadas, de forma que ha cambiado la secuencia. Este hecho hace que el motor gire al revés que en el caso anterior. Por tanto, también hemos comprobado que funciona correctamente esta opción del programa.

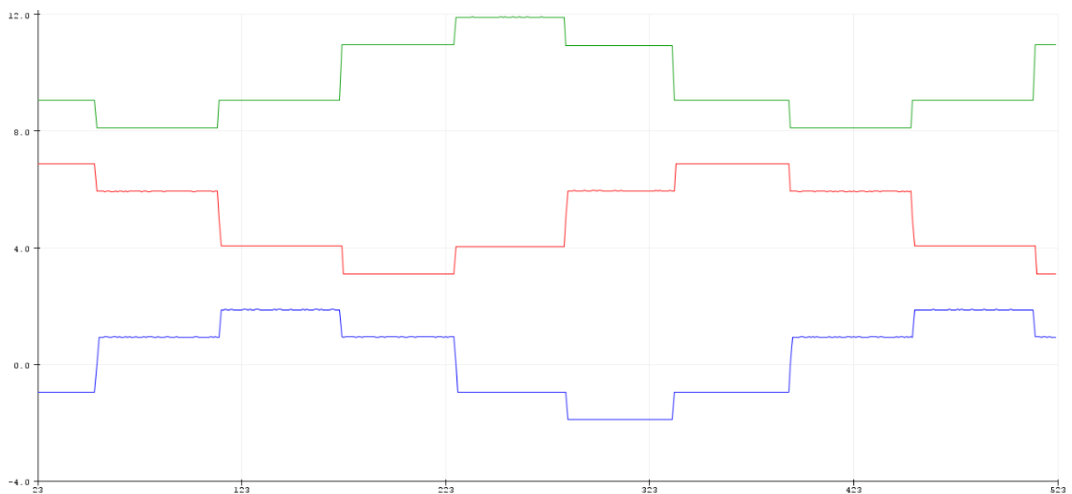


Figura 44: Tensión fase - fase. (Elaboración propia)

En la Figura 44, comprobamos cual es el aspecto de las señales de tensión si medimos entre dos fases.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

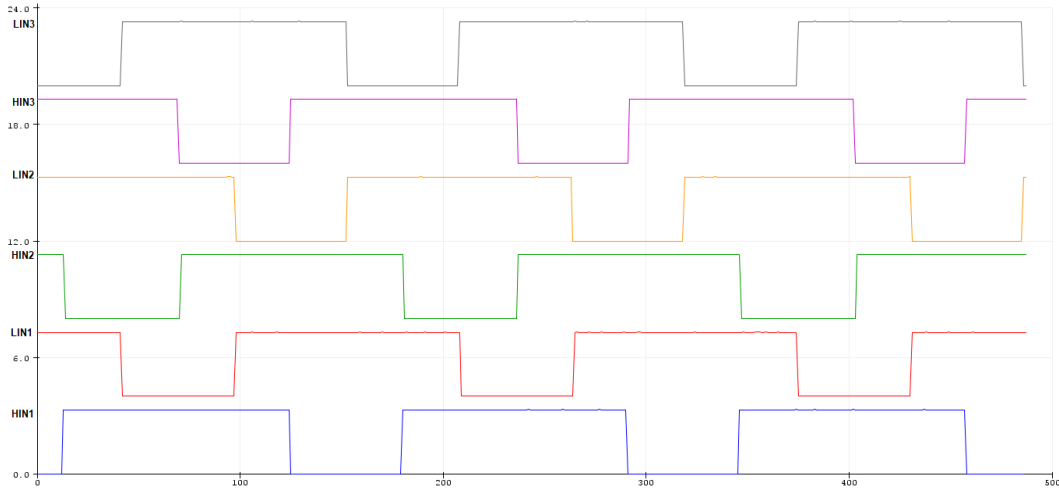


Figura 45: Señales de disparo de los transistores. (Elaboración propia)

En la Figura 45 hemos medido todas las señales de disparo de los transistores. Con ello, comprobamos que funcionan adecuadamente las secuencias de pulsos creadas para llevar a cabo el movimiento del motor.

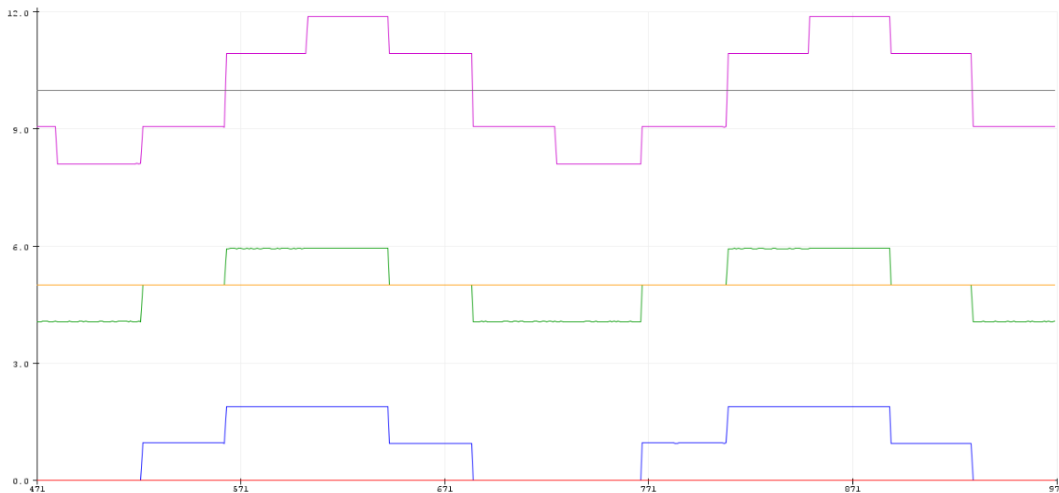


Figura 46: Tensión compuesta, simple y entre fase y tierra. (Elaboración propia)

En la Figura 46, se han representado las gráficas de tensión compuesta, simple y entre fase y tierra. Con esta figura, comprobamos que el sistema funciona perfectamente, pues en la simulación de la Figura 15, se representaron las gráficas de las tensiones tomadas en los mismos puntos que estas y el resultado es el mismo.

A continuación, el siguiente paso a dar, es probar el mismo circuito, pero con el IRAMY20UP60B en la placa de pruebas. Este circuito se ve en la Figura 47.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

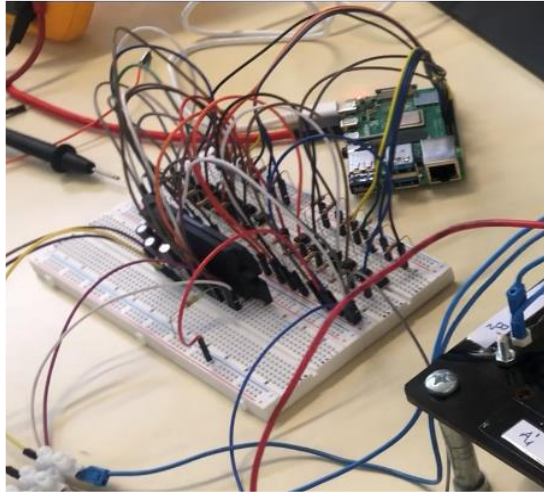


Figura 47: Prueba del IRAMY20UP60B. (Elaboración propia)

Una vez comprobado también su funcionamiento, se procede a alimentar un motor paso a paso, conectando sus seis bobinas individuales de una forma similar a un motor síncrono, es decir, tres bobinas dobles conectadas en estrella. En la Figura 48, se puede ver dicho motor alimentado.

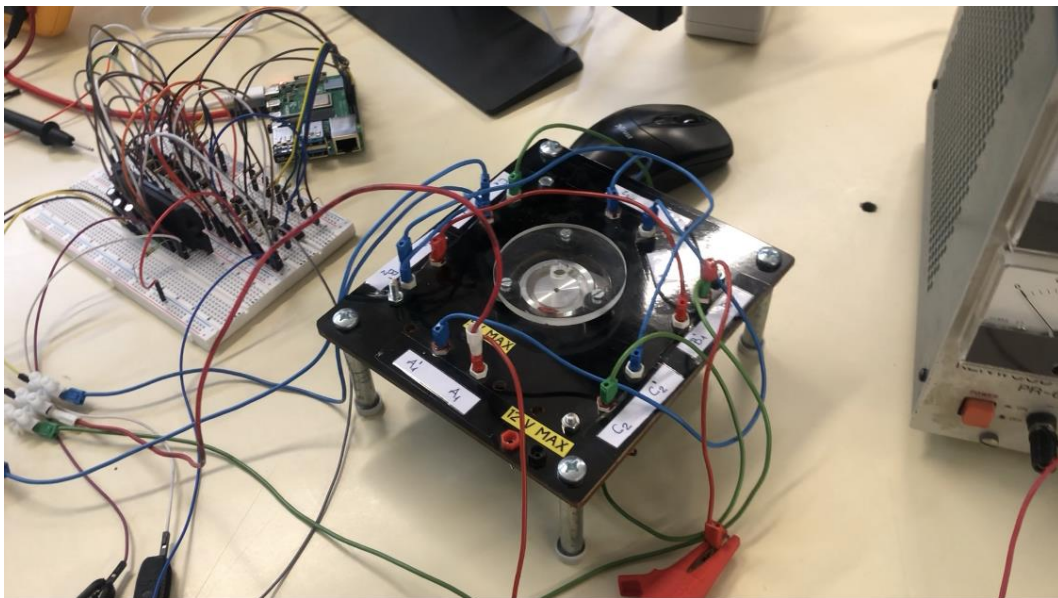


Figura 48: Motor paso a paso de 5V alimentado. (Elaboración propia)

Este motor empleado, tiene una tensión de alimentación de 5V. Una vez cargado el programa en la Raspberry PI, se puede probar cómo funciona aumentando y disminuyendo la velocidad. También este motor nos permite comprobar que, si la variación de velocidad se hace de forma brusca, el motor se bloquea. Este hecho, es el que hace necesario que se programe la rampa del variador de frecuencia.

Y, finalmente, en la Figura 49, se ha medido con el osciloscopio la tensión entre dos fases.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

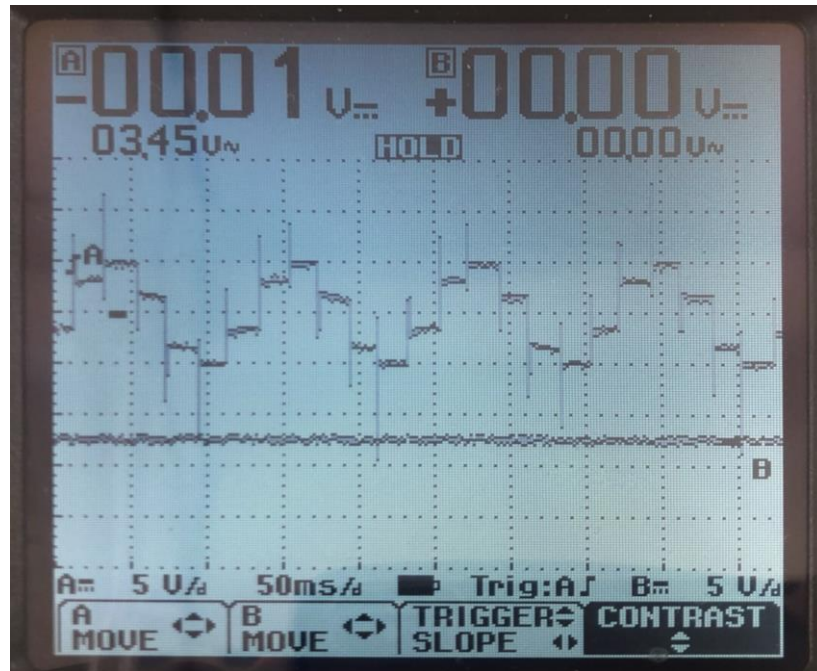


Figura 49: Alimentación del motor paso a paso de 5V. (Elaboración propia)

Con la gráfica obtenida, comprobamos que el motor está alimentado como hemos planteado anteriormente, la onda de tensión es igual que la simulación de la Figura 15 y que la prueba de la Figura 46 y al comprobar la frecuencia de la señal, se verificó que la velocidad indicada para el giro del motor en el programa creado es igual que la que se obtiene con dicha frecuencia.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

4. Control sincronizado para dos motores de diferentes características.

En este apartado se va a tratar el objetivo que se pretende alcanzar en este trabajo, que consiste en hacer girar dos motores síncronos, como si fueran motores paso a paso, a la misma velocidad y, cuando se haya alcanzado esta, variar la velocidad de uno de ellos, mientras que la del otro se mantiene constante. De esta forma, se pretende recrear de forma eléctrica el comportamiento que tiene el diferencial de las ruedas de un vehículo, es decir, cuando se toma una curva, una de las ruedas del mismo eje gira más lenta que la otra para que el coche no derrape.

4.1. Estudio teórico.

Para este apartado, el estudio teórico de cómo controlar el motor es exactamente igual que el descrito en el Apartado 3.1 de este trabajo. Vamos a emplear dos motores síncronos como motores paso a paso, pues como hemos visto, sus características constructivas son similares. Existen varias formas de alimentar a estos motores, con las cuales se crea un campo magnético que gira en un determinado sentido, una de ellas, consiste en alimentar cada bobina de forma individual (si tenemos accesibles los extremos de las bobinas) y en una secuencia de activación adecuada. En la Figura 50, se muestra el conexionado necesario y la tabla de activación de cada una de las bobinas.

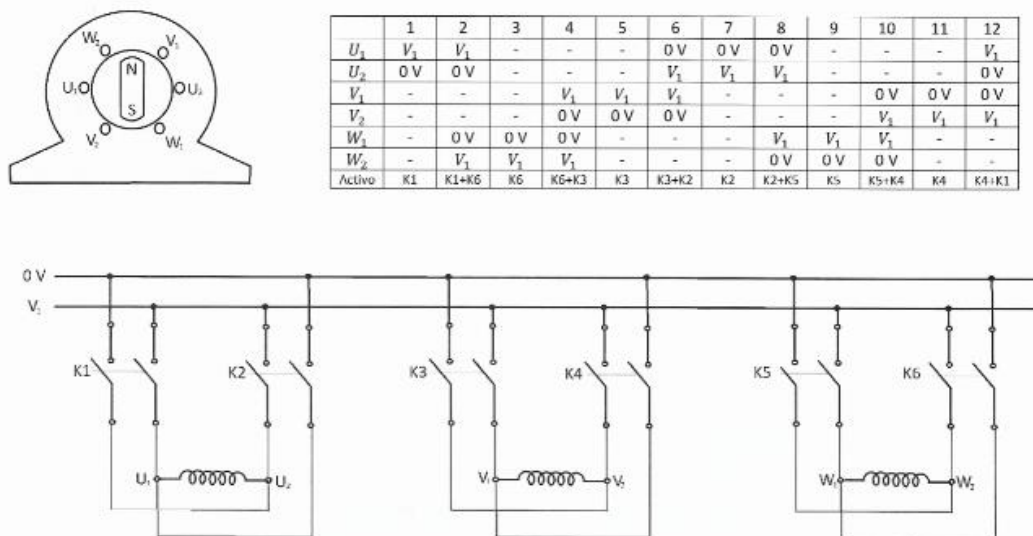


Figura 50: Motor síncrono como paso a paso con control de todas las bobinas. [8] [9]

Esta forma descrita sería propiamente la de alimentar el motor como un paso a paso, pero como no todos los motores tienen accesibles los dos extremos de las bobinas, es necesario que diseñemos otro tipo de alimentación. Además, de esta manera se requieren doce transistores para

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

el control de los pasos, y el circuito integrado de IGBTs que vamos a emplear para el control y alimentación del motor solo tiene seis. De esta forma sería necesario emplear dos IRAMY20UP60B por cada motor.

Para solucionar estos problemas de si no están accesibles las bobinas o el de tener que emplear dos circuitos integrados por motor, se plantea otra alternativa, que consiste en unir en estrella un extremo de las bobinas y alimentar directamente entre dos fases de entrada, es decir, siempre estarían activas dos bobinas en cada momento.

En la Figura 51, se muestran dos alternativas para alimentación del motor con las bobinas conectadas en estrella.

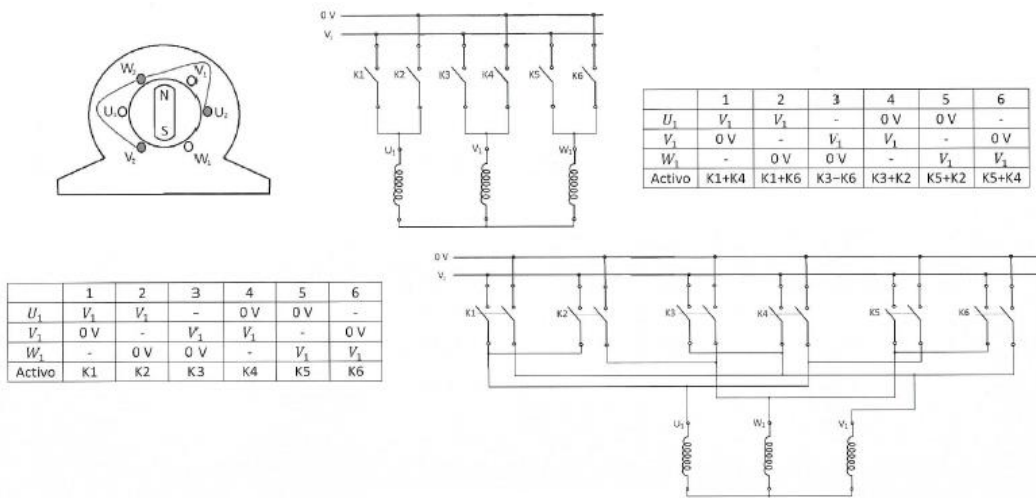


Figura 51: Motor síncrono como paso a paso con bobinas en estrella. [8] [9]

Entre estas dos opciones nuevas, vemos que la que se encuentra en la parte superior de la figura es la más adecuada y óptima. Para esta opción, se emplean solamente seis IGBTs que controlan si la bobina se conecta al positivo de la tensión continua o al negativo.

La segunda opción planteada en la figura emplea doce IGBTs, lo que nos conduciría a la misma situación que en casos anteriores, es decir, el empleo de dos circuitos integrados.

Por otra parte, como nuestro programa se va a emplear para alimentar dos motores de diferentes características, lo más lógico es que el método de alimentación sea lo más genérico posible. Entre los motores con los que vamos a probar, tres sí que cuentan con los extremos de las bobinas accesibles, pero, sin embargo, en el cuarto motor (que se trata del alternador de un coche) tiene ya conectadas las bobinas en estrella de forma interna.

4.2. Programa de control.

El programa de control del capítulo anterior funcionaba de forma correcta cuando se prueba de manera experimental con las resistencias que simulan el motor o incluso con el pequeño motor de 5V empleado para las pruebas, pero tenía algunos aspectos que no terminaban de funcionar de manera correcta, como el cambio de sentido de giro, ya que, en las pruebas realizadas, hemos visto que con motores grandes el funcionamiento a baja velocidad no funciona adecuadamente. A menos de 60 r/min suelen dar problemas. Esto se debe en gran medida a que el IRAMY20UP60B es un controlador empleado para hacer señales PWM de alta frecuencia y, al emplearlo en bajas frecuencias hay que instalar unos condensadores en sus salidas que permiten mejorar un poco la respuesta del mismo a bajas frecuencias.

El principal cambio en el programa es que ahora se han implementado dos bucles while loop iguales. Cada uno de ellos se encarga de controlar la velocidad de un motor y, con un bloque llamado *“Wait Until Next ms Multiple”* podemos controlar la frecuencia con la que trabaja cada uno de dichos bucles. Se puede trabajar a diferentes frecuencias, porque la Raspberry PI cuenta con un multiprocesador, de forma que puede hacer varias cosas a la vez. Para comprobar que podía trabajar a diferentes frecuencias, se programó un sencillo ejecutable que controlaba dos diodos led, uno parpadeando a 3 veces la frecuencia del otro y, al comprobar que no se producía ningún retraso en las frecuencias, se confirmó la idea de emplear la Raspberry PI.

También se han hecho algunos cambios en el programa para optimizar su carga computacional. Para ello, se ha modificado el código que actualiza las salidas, empleando un bucle de tipo switch y no una serie continua de bucles if. Por otra parte, se ha anulado el sistema de cambio de sentido de giro, ya que no aportaba una solución útil al funcionamiento, dado que no funciona el motor a bajas velocidades.

En la Figura 52, se muestra el nuevo bloque de activación de salidas. También se ha definido que el tipo de salida es int32, por lo que ahora aparece en azul y no en naranja como anteriormente.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

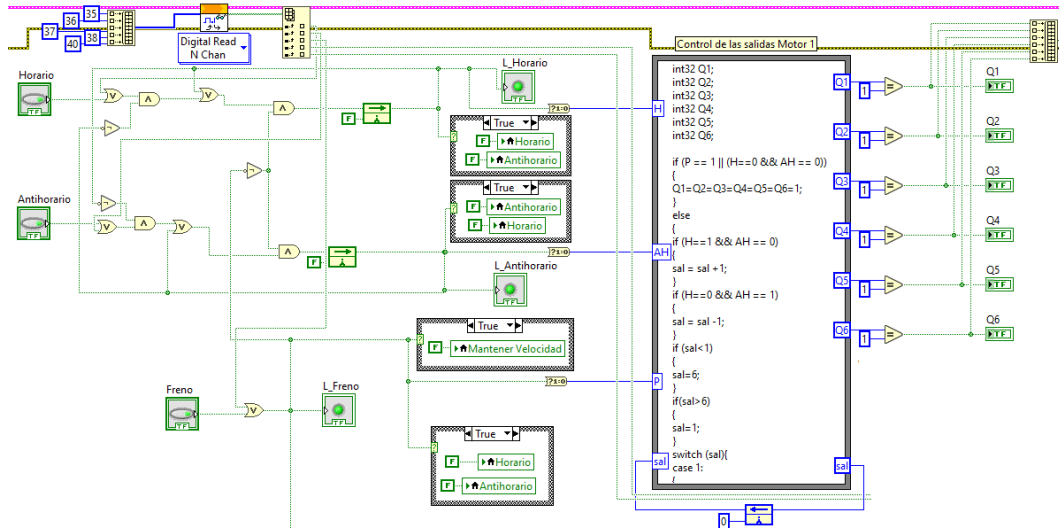


Figura 52: Bloque de activación de salidas nuevo. (Elaboración propia)

También, se han eliminado del programa la utilización de variables locales. De esta forma, cuando LabVIEW utiliza una variable, la lee directamente de su entrada sin necesidad de perder tiempo en leer una variable local de memoria.

Por otra parte, el sistema de marcha y paro es nuevo, es decir, se ha reformado de manera que, si hemos pulsado el botón de marcha en sentido horario, al pulsar el de antihorario no va a pasar nada hasta que el motor no se haya detenido por pulsar el freno.

Por último, como novedad de esta parte, se han añadido cinco entradas digitales, de las cuales, tres están destinadas al control de la marcha en un sentido o en otro y el paro del motor. De esta forma, cargando el programa en la Raspberry PI, no es necesario utilizar el ordenador, ya que se puede iniciar y parar la marcha de forma física por medio de los pulsadores colocados en la caja de mando.

Por la parte del bloque variador de frecuencia, los cambios introducidos, se basan en eliminar el cambio de giro, ya que como antes hemos comentado, con motores grandes no funciona por el tipo de controlador empleado y, además, se ha añadido un dial, con el que se permite variar la velocidad de un motor respecto del otro. Como cada bucle lleva su propio variador de frecuencia, es como que cada motor está controlado de forma independiente.

En la Figura 53, se pueden ver todos estos cambios introducidos en este segundo bloque del programa. Otro cambio significativo es que ahora al programa le podemos indicar que el motor ha de girar a una determinada velocidad y no a una frecuencia dada como sucedía antes. Como también indico el número de pares de polos de cada motor, el propio programa es

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

capaz de saber con qué frecuencia ha de alimentar a cada uno de los motores para que giren a la misma velocidad.

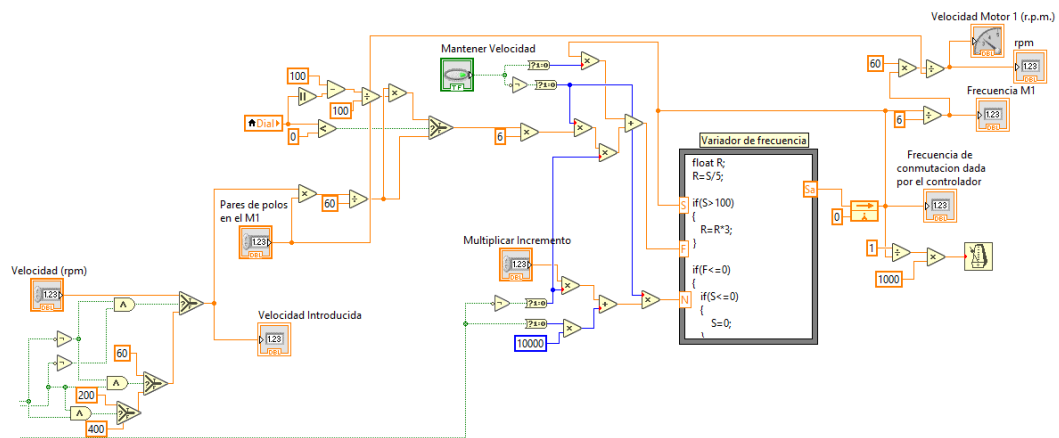


Figura 53: Bloque variador de frecuencia. (Elaboración propia)

Anteriormente, hemos comentado que se han introducido cinco señales de entrada digitales, con las dos últimas, controlamos la velocidad del motor de forma manual. Se trata de dos interruptores, con los cuales si están en la posición de 0 0, se introduce la velocidad definida por el ordenador, con 0 1, se hace girar el motor a 60 r/min, con 1 1 a 200 r/min y con 1 0 a 400 r/min. De esta forma, se puede hacer un ensayo rápido del programa sin necesidad de emplear un ordenador.

En la Figura 54, se muestra en detalle todo el apartado de la variación de giro de un motor respecto al otro y, como funcionan las entradas digitales.

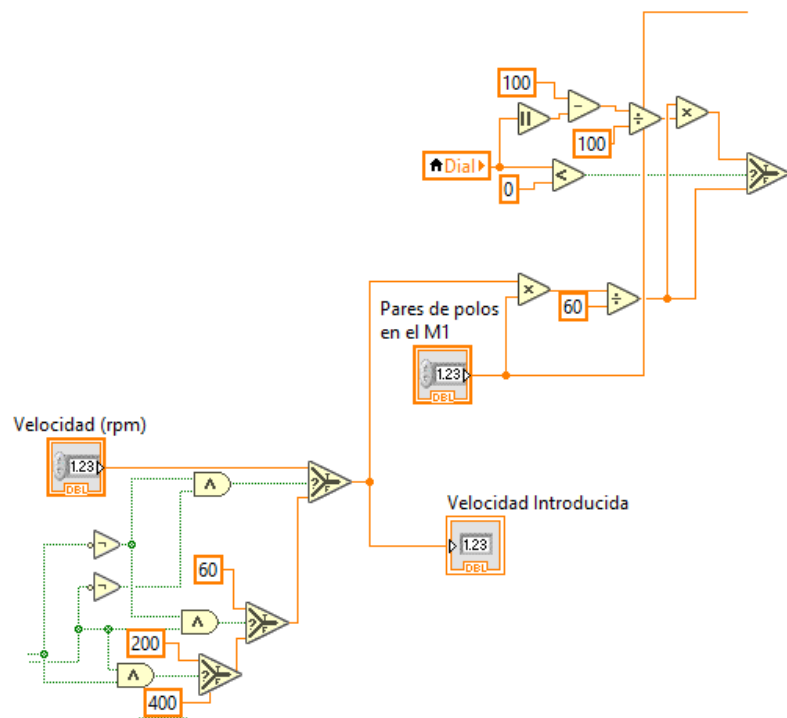


Figura 54: Control de la velocidad. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

En la parte inferior de la Figura, vemos como se introduce una consigna de velocidad, llamada “Velocidad (rpm)”, desde el panel frontal de la aplicación y dependiendo de cómo estén activadas las entradas digitales, este valor se puede cambiar por el de una de las constantes correspondientes a las velocidades del modo manual.

En la parte superior, aparece la variable “Dial”, que consiste en una ruleta en la que el valor numérico de salida varía de -20 a 20. Su función es la de reducir la velocidad de uno de los motores mientras que el otro mantiene la misma velocidad. Cuando el valor de entrada se encuentra entre - 20 y 0, el motor 1 es el que reducirá su velocidad el tanto por ciento introducido en el “Dial”, respecto de la velocidad fijada, que será la que tenga el motor 2. Si el valor numérico del “Dial” se encuentra entre 0 y 20, la velocidad que varía es la del motor 2, manteniendo constante la del motor 1. Como máximo, se ha fijado que la diferencia entre las velocidades sea de un 20%, para que pueda ser visualizado posteriormente al hacer pruebas reales.

Por otra parte, como en cada bucle existe un bloque variador de frecuencia, al hacer este cambio de velocidad, no se produce un cambio brusco en el momento introducido, sino que se va variando poco a poco para evitar que el motor se bloquee.

En las Figura 55 se va a mostrar el aspecto final del panel frontal que aparece en el ordenador al ejecutar el programa en la Raspberry PI y desde el cual se pueden manejar todos los controles del motor. En la siguiente figura (56) se puede ver el diagrama de bloques del programa completo, con los dos bucles de tipo while loop y todas las variables empleadas, la lectura de las entradas digitales y la escritura de las salidas digitales, responsables de los disparos de los IGBTs.

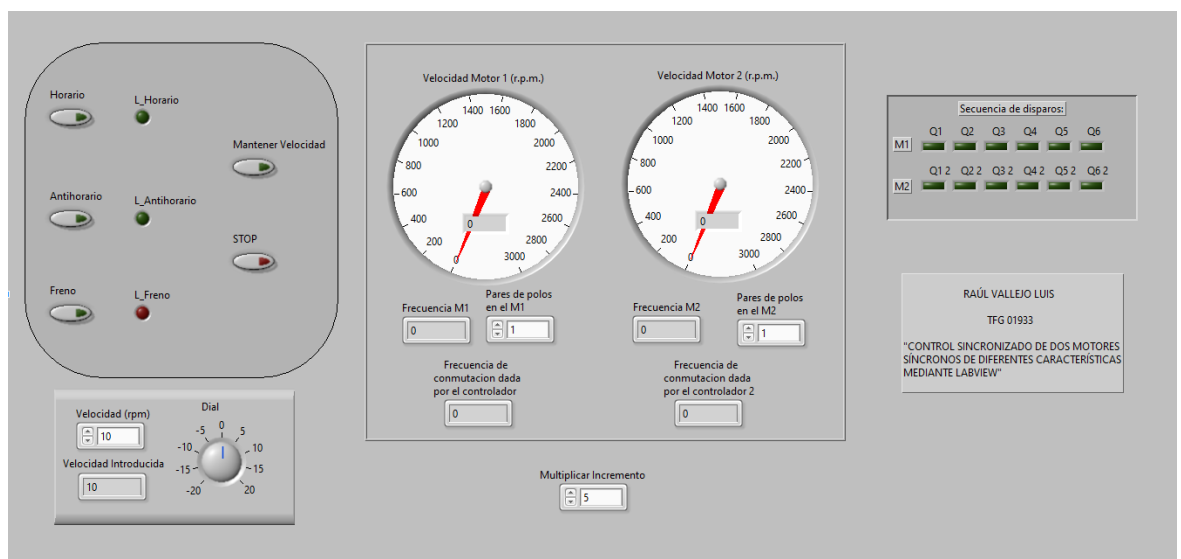


Figura 55: Panel frontal de la aplicación. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

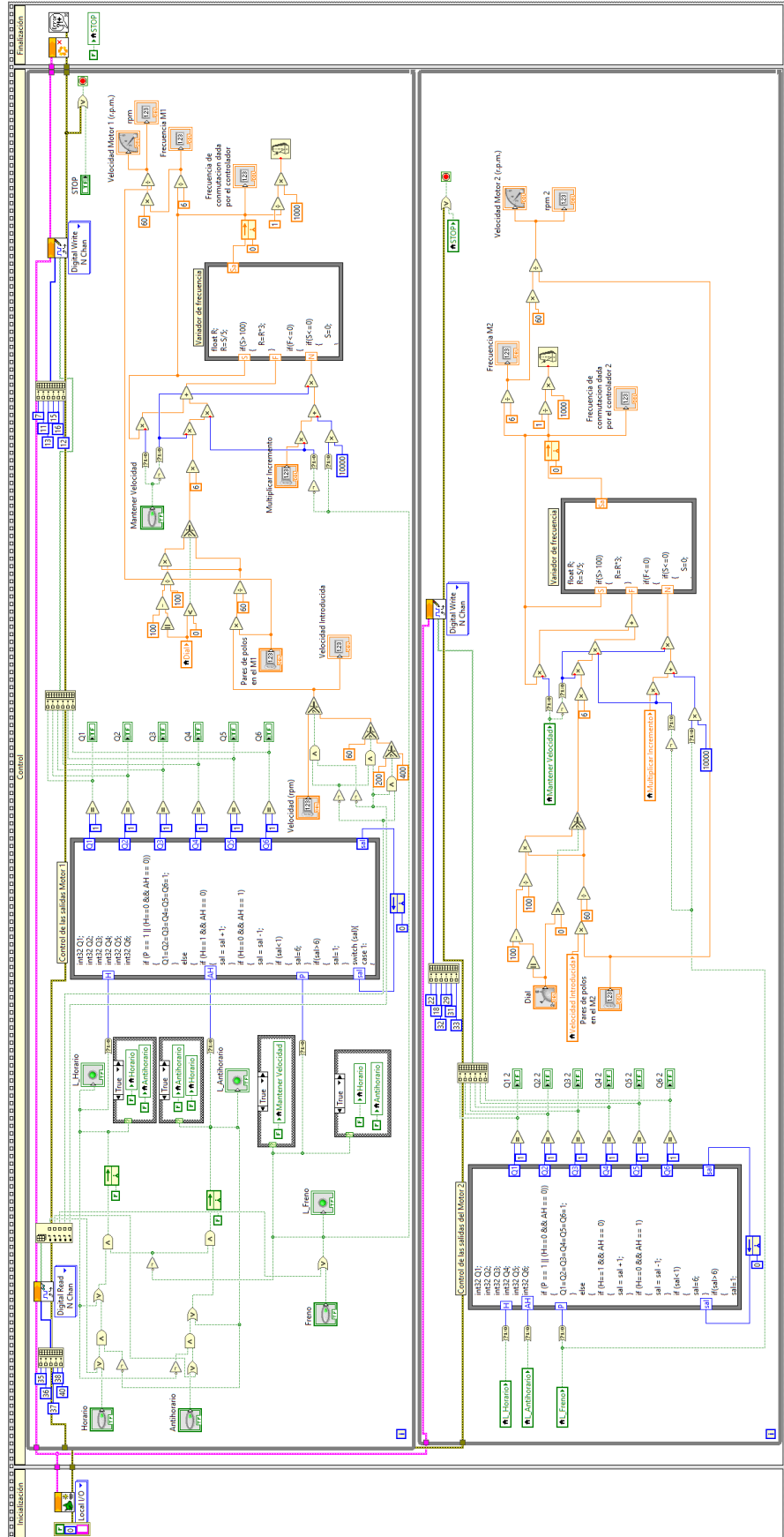


Figura 56: Diagrama de Bloques. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

En la Figura 57, se muestra la caja que contiene el control manual del programa con los interruptores de las entradas digitales, que estará instalada junto a la Raspberry PI en el bastidor que contiene todo el montaje.

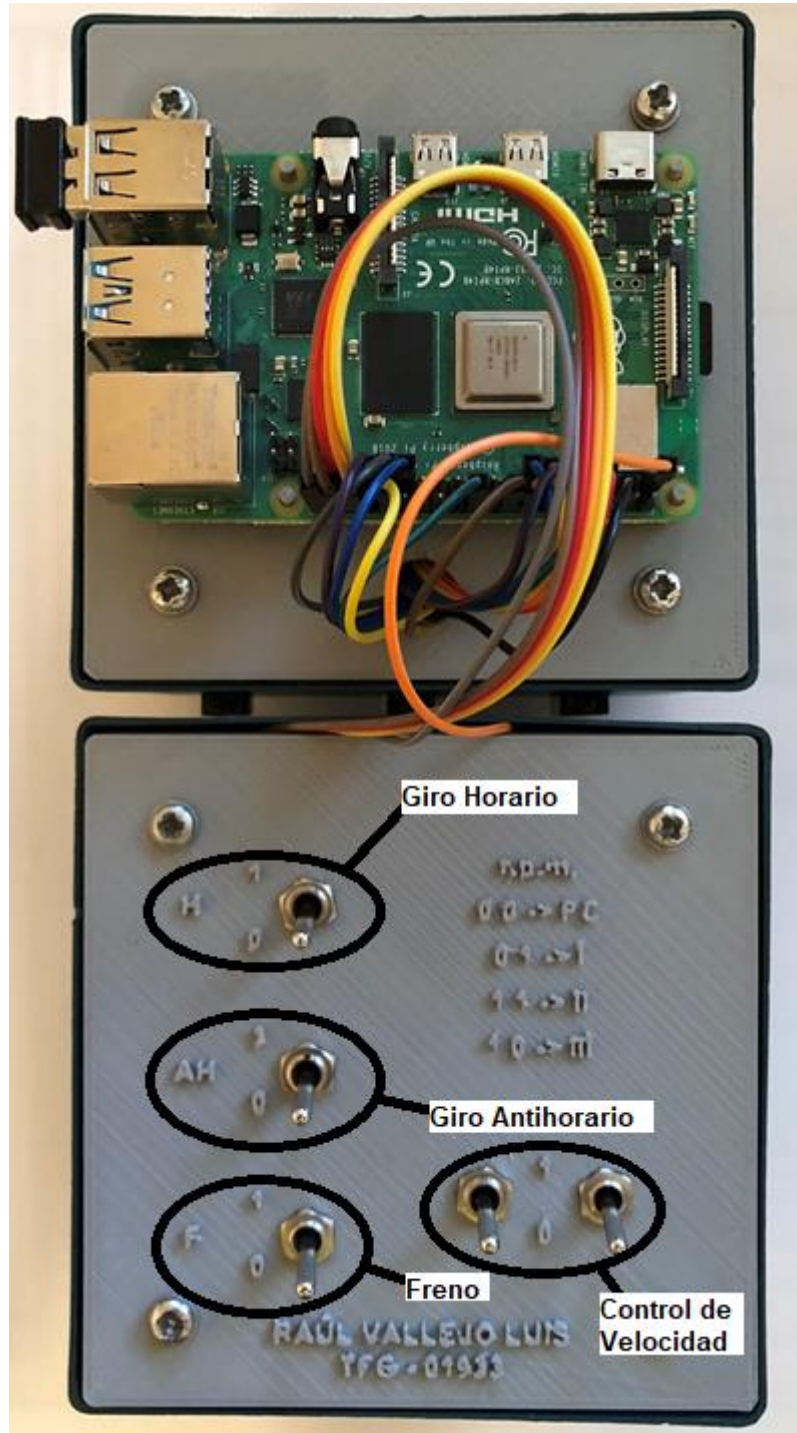


Figura 57: Control manual del programa. (Elaboración propia)

4.3. Implementación física.

Para conseguir un funcionamiento óptimo del programa creado, se va a realizar una implementación física de todo el circuito eléctrico y así de esta forma, se reduce al mínimo los problemas de cableado o malas conexiones.

El objetivo de este apartado es crear una placa de circuito impreso en la que se encuentren los componentes electrónicos que vamos a emplear en el circuito de control de cada motor.

Para realizar la placa de circuito, analizamos los componentes a utilizar:

- IRAMY20UP60B: se trata del puente de IGBTs que controlan los pulsos de tensión continua con los que alimentamos el motor.
- Condensadores: es necesario colocar tres condensadores, uno en cada salida de las fases del módulo IRAMY20UP60B, según se indica en el catálogo del fabricante.
- Borneros en las salidas, para realizar la conexión con el motor y la Raspberry PI.
- Optoacopladores, para realizar una separación eléctrica de las señales digitales que salen de la Raspberry PI y las que entran en el circuito integrado de control. De esta forma, se protege a la Raspberry PI en caso de un fallo eléctrico en la zona de potencia.
- Resistencias, para los optoacopladores y para la señal de temperatura que ofrece el circuito integrado.

El siguiente paso, consiste en crear el esquema eléctrico de dicho circuito con el programa MSIM8, que luego nos va a permitir crear la placa de circuito impreso.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

En la Figura 58, se puede ver el esquema eléctrico, con todas las conexiones de salida.

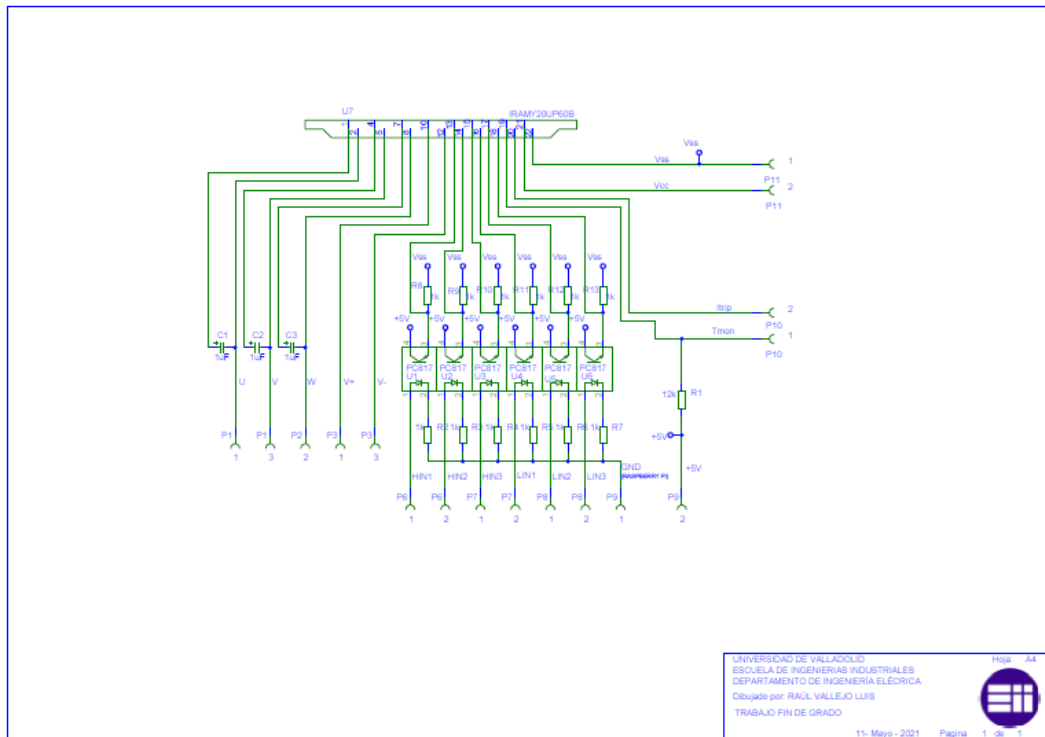


Figura 58: Esquema eléctrico. (Elaboración propia)

En el caso de los optoacopladores (Figura 59), el montaje se ha realizado de la siguiente manera. En la patilla 1, se introduce la salida digital de la Raspberry PI, en la patilla 2, se coloca una resistencia al GND de la Raspberry PI, de forma que en esta zona del optoacoplador habrá pulsos entre 0 y 3.3 V. Después, en la patilla 3 se une a la entrada del integrado correspondiente y también, se coloca una resistencia unida al pin Vss del integrado (es decir, al GND de la parte de potencia) y finalmente, la patilla 4 se une a la entrada de alimentación de 5 V. De esta forma, los pulsos que recibe el integrado van entre 0 y 5 V. Aunque para las señales de entrada también valía alimentar con 3.3 V y que el integrado detectara un 1 lógico, según el catálogo del fabricante, se ha optado por alimentar a 5 V ya que esta tensión se va a emplear después en la señal de temperatura que ofrece el controlador y de esta manera aprovechamos la fuente de alimentación.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

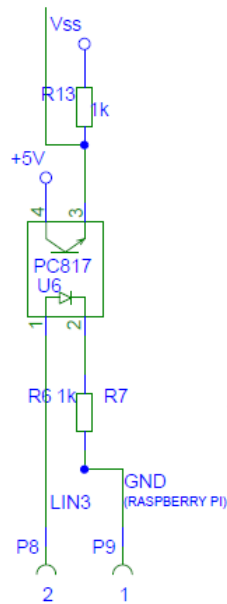


Figura 59: Conexión Optoacopladores. (Elaboración propia)

El siguiente paso, es realizar el diseño físico que se imprimirá en una placa de circuito impreso. Con el mismo programa y tras haber elegido los footprints adecuados de cada componente, se procede a ordenarles de manera adecuada en el espacio de la placa y realizar el rutado.

Para realizar el circuito, hemos de tener en cuenta algunos datos técnicos presentes para dimensionar el tamaño de las pistas y la separación entre las mismas.

PISTAS DE POTENCIA:

$$W = \frac{A}{e}$$

- e (espesor de la pista) = 35 μm , que en milésimas de pulgada son 1.378 mils.
- ΔT (variación de temperatura en las pistas) = 70 $^{\circ}\text{C}$
- I máx = 20 A.

La fórmula a emplear para un tipo de pista externa es $A = \left(\frac{I}{0.048 * \Delta T^{0.44}} \right)^{\frac{1}{0.725}}$, de donde se obtendrá el área de la sección de la pista en milésimas de pulgada al cuadrado. Después, se obtiene el ancho de la pista como $W = \frac{A}{e}$ en milésimas de pulgada, que es la unidad empleada por el programa de diseño de placas de circuito impreso.

- I = 20 A $\rightarrow A = \left(\frac{20}{0.048 * 70^{0.44}} \right)^{\frac{1}{0.725}} = 287.424 \text{ mils}^2$

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

$$W = \frac{287.424}{1.378} = 208.58 \text{ mils}$$

Para las pistas de la zona de potencia, la anchura obtenida es de 208 milésimas de pulgada, un valor bastante ancho, por lo que construiremos nuestras pistas con una anchura entre 80 y 150 milésimas de pulgada intentando así respetar la distancia entre pistas en los pads del circuito integrado. Para compensar esta disminución en el tamaño de las pistas, una vez finalizada la construcción de la placa de circuito, estañaremos todas las pistas con una buena capa de estaño, de forma que habremos aumentado su espesor, que antes, cuando hablábamos de la pista de cobre era de 35 μm .

PISTAS DE CONTROL:

- e (espesor de la pista) = 35 μm , que en milésimas de pulgada son 1.378 mils.
- ΔT (variación de temperatura en las pistas) = 70 °C
- I máx = 5 mA.

- $I = 5 \text{ mA} \rightarrow A = \left(\frac{0.005}{0.048 * 70^{0.44}} \right)^{\frac{1}{0.725}} = 0.0031 \text{ mils}^2$

$$W = \frac{0.0031}{1.378} = 0.0022 \text{ mils}$$

En las pistas de la zona de control, la anchura obtenida es de 0.0022 milésimas de pulgada, un valor bastante pequeño, que vamos a aumentar hasta las 25 milésimas de pulgada. Este aumento se debe a que vamos a fabricar la placa con ácido de forma manual y no es conveniente dejar pistas muy estrechas para evitar que se rompan.

SEPARACIÓN ENTRE PISTAS:

Para la separación entre las pistas, vamos a utilizar la tabla que aparece en la Figura 60, que define la separación eléctrica entre conductores dependiendo de los tipos de pistas empleados. En nuestro circuito, las pistas son de tipo B2, es decir, conductores externos, sin recubrimiento y su empleo va a estar entre la altura del nivel del mar y 3050 metros.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Table 6-1 Electrical Conductor Spacing

Voltage Between Conductors (DC or AC Peaks)	Minimum Spacing						
	Bare Board				Assembly		
	B1	B2	B3	B4	A5	A6	A7
0-15	0.05 mm	0.1 mm	0.1 mm	0.05 mm	0.13 mm	0.13 mm	0.13 mm
16-30	0.05 mm	0.1 mm	0.1 mm	0.05 mm	0.13 mm	0.25 mm	0.13 mm
31-50	0.1 mm	0.6 mm	0.6 mm	0.13 mm	0.13 mm	0.4 mm	0.13 mm
51-100	0.1 mm	0.6 mm	1.5 mm	0.13 mm	0.13 mm	0.5 mm	0.13 mm
101-150	0.2 mm	0.6 mm	3.2 mm	0.4 mm	0.4 mm	0.8 mm	0.4 mm
151-170	0.2 mm	1.25 mm	3.2 mm	0.4 mm	0.4 mm	0.8 mm	0.4 mm
171-250	0.2 mm	1.25 mm	6.4 mm	0.4 mm	0.4 mm	0.8 mm	0.4 mm
251-300	0.2 mm	1.25 mm	12.5 mm	0.4 mm	0.4 mm	0.8 mm	0.8 mm
301-500	0.25 mm	2.5 mm	12.5 mm	0.8 mm	0.8 mm	1.5 mm	0.8 mm
> 500 See para. 6.3 for calc.	0.0025 mm /volt	0.005 mm /volt	0.025 mm /volt	0.00305 mm /volt	0.00305 mm /volt	0.00305 mm /volt	0.00305 mm /volt

B1 - Internal Conductors
 B2 - External Conductors, uncoated, sea level to 3050 m
 B3 - External Conductors, uncoated, over 3050 m
 B4 - External Conductors, with permanent polymer coating (any elevation)
 A5 - External Conductors, with conformal coating over assembly (any elevation)
 A6 - External Component lead/termination, uncoated
 A7 - External Component lead termination, with conformal coating (any elevation)

Figura 60: Separación eléctrica entre conductores. [10]

Para el cálculo de la separación necesitamos saber la diferencia de potencial máxima que va a existir entre conductores próximos.

- Pistas de Potencia: 300 V -> Según la tabla, escogemos el apartado que va entre 301 y 500 V, para evitar cualquier problema. La separación es de: 0.25 mm (98.425 mils).
- Pistas de Control: 17 V -> Según la tabla, escogemos el apartado que va entre 16 y 30 V. La separación es de: 0.1 mm (3.973 mils).

Una vez realizado este cálculo, el siguiente paso es elegir los borneros, de forma que estos cumplan con los requisitos de separación y corriente máxima. En la Figura 61, se muestra la descripción de los borneros a emplear.

5.08mm High Power Rising Clamp Terminal

- 2-12 pole interlocking
- Captive screw
- Moulded in green UL94-V0 flame retardant PA
- Professional rising clamp terminals
- 0.05-3.3mm² (30-12 AWG)
- approved

CTB0709



SPECIFICATION

Rating	20A 300V
Insulation resistance	6000MΩ/1000V
Withstand voltage	AC2000V/1min
Contact Resistance	20mΩ
Working Temperature	-33°C to +120°C
Mouldings	UL94-V0 flame retardant PA66
Pin/Terminal	Tin plated brass
Wire clamp	Nickel plated brass
Screw/Torque	M3 Steel, Nickel Plated 0.4Nm
Recommended PCB hole size	ø1.6mm

TYPE NO.	POLES	LENGTH
CTB0709/2	2	10.16mm
CTB0709/3	3	15.24mm
CTB0709/4	4	20.32mm
CTB0709/6	6	28.4mm
CTB0709/6	6	30.48mm
CTB0709/7	7	36.66mm
CTB0709/8	8	40.64mm
CTB0709/9	9	45.72mm
CTB0709/10	10	50.8mm
CTB0709/11	11	55.88mm
CTB0709/12	12	60.96mm

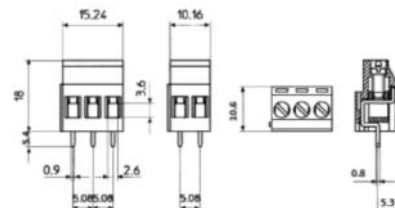


Figura 61: Datasheet del bornero. (RS Componentes)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Como estos borneros no cumplen la separación entre pistas, la solución tomada para este problema consiste en montar toda la regleta de bornes dejando un pin libre entre cada salida. De esta forma la separación entre dos pines contiguos será de 10 mm.

En cuanto a los bornes de la zona de control, se han empleado otros, cuyas características eléctricas son inferiores, dado que apenas circulará corriente por ellos y que la distancia entre conductores no va a ser un problema. En la Figura 62 aparecen sus características eléctricas.

Descripción

Bloque de terminales de tornillo
Número de pines: 2
Parcela: 5mm
Tensión nominal: 300V, corriente nominal: 15A.
Para el tamaño del cable: 14-22AWG
Color Verde
El paquete incluye:
Bloque de terminales de tornillo de 100pcs 2 pines
Caja de almacenamiento de 1 pcs.
Destornillador cableado 1 piezas

Figura 62: Descripción eléctrica de los bornes de la parte de control. (Amazon)

Por último, en la Figura 63 aparecen las características eléctricas de los optoacopladores empleados para este proyecto.

Descripción

Parámetros eléctricos:

PC817: Optoacoplador compacto de paquete doble de serie de un solo canal Velocidad de transmisión actual: (CTR: mínimo 50%, IF = 5mA, V CE = 5V)
El valor efectivo del alto voltaje de aislamiento entre entrada y salida es 5000V.
Corriente continua (ICEO): 50 mA
Corriente pico DC (ICE max): 1A
Tensión inversa: 6V
Consumo de energía: 70mW
Voltaje del emisor colector: 35V
Voltaje del colector emisor: 6V
Corriente de colector: 50mA
Consumo de energía del colector: 150mW
Consumo total de energía: 200mW
Temperatura de trabajo: -30 °C ~ + 100 °C
Voltaje de saturación del emisor del colector: 0.1V (valor típico)

Figura 63: Características eléctricas de los optoacopladores. (Amazon)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Una vez definidos todos los esquemas eléctricos, componentes, la anchura y separación de las pistas se procede al rutado manual de la tarjeta. En las siguientes figuras (64 - 67) se muestran los fotolitos obtenidos, correspondientes a cada una de las capas.

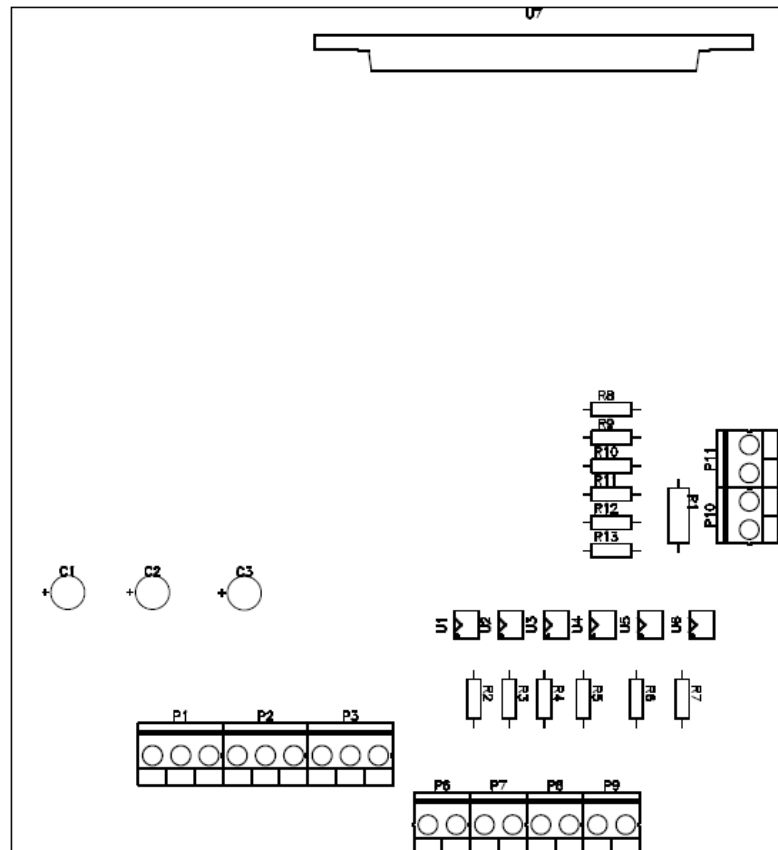


Figura 64: Fotolito de la Cara de Componentes. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

MASCARA DE SOLDADURA SUPERIOR

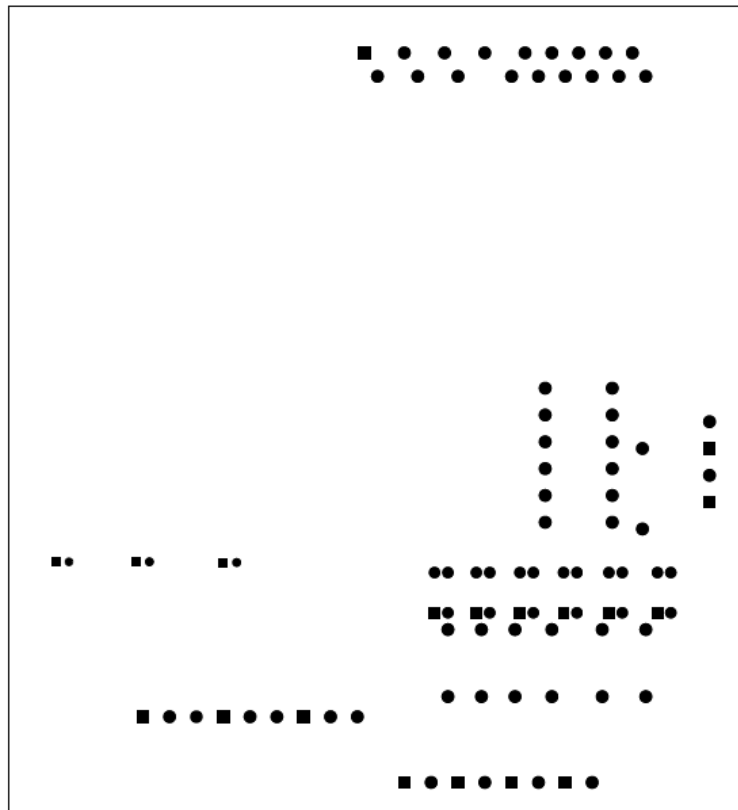


Figura 65: Fitolito Máscara de Soldadura Superior. (Elaboración propia)

FOTOLITO CARA DE SOLDADURA

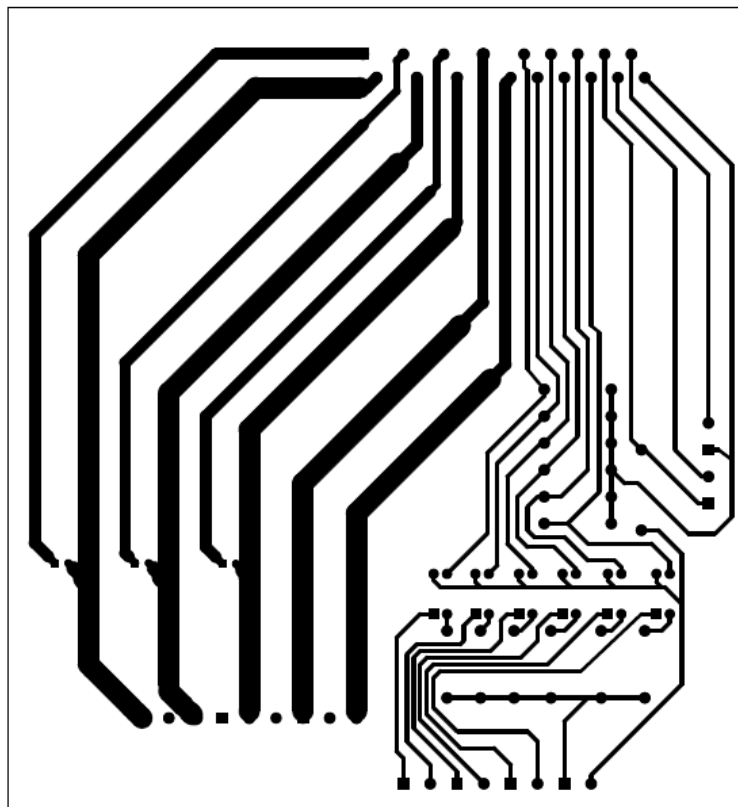


Figura 66: Fitolito Cara de Soldadura. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

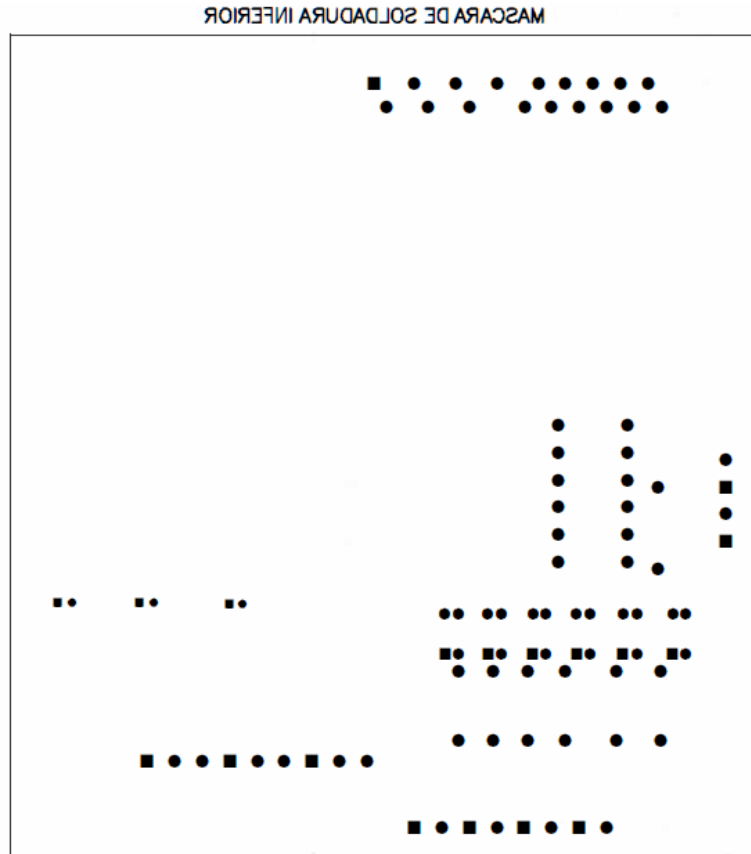


Figura 67: Fotolito Máscara de Soldadura Inferior. (Elaboración propia)

Finalmente, para realizar la impresión de la placa utilizaremos el fotolito de la cara de soldadura. Se introducirá en una insoladora junto a la placa de cobre y posteriormente se revela la imagen insolada. Después, por medio de tratamientos químicos se deshace el cobre sobrante y se obtiene la placa de circuito impreso, como se muestra en la Figura 68.

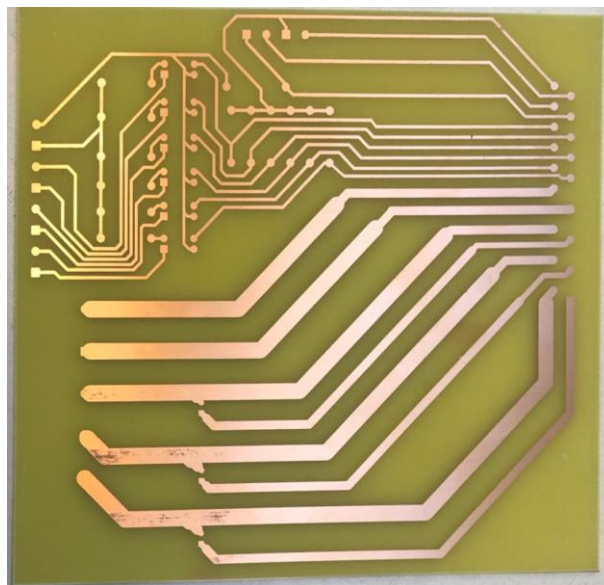


Figura 68: Resultado final de la PCB. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Después de obtener la tarjeta, se estañan las pistas, como se comentó anteriormente, para aumentar su espesor y de esta forma, la intensidad máxima que pueda circular, se realiza el taladrado de los pads para poder introducir los componentes y se sueldan estos también. En la Figura 69, se muestra el resultado final de todo el proceso.



Figura 69: Resultado final de la PCB. (Elaboración propia)

Como elementos de protección, instalaremos un fusible rápido de 2 A, que irá colocado en el panel del bastidor, donde se ubicará el montaje final. Este fusible de protección va a colocado en el cable de alimentación de la patilla V+ de la zona de potencia del integrado para evitar que una sobrecorriente pueda dañar alguno de los componentes de la placa, como ocurrió durante una de las primeras pruebas (Figura 70), donde se fundieron las patillas de entrada de tensión continua durante el arranque de un motor.

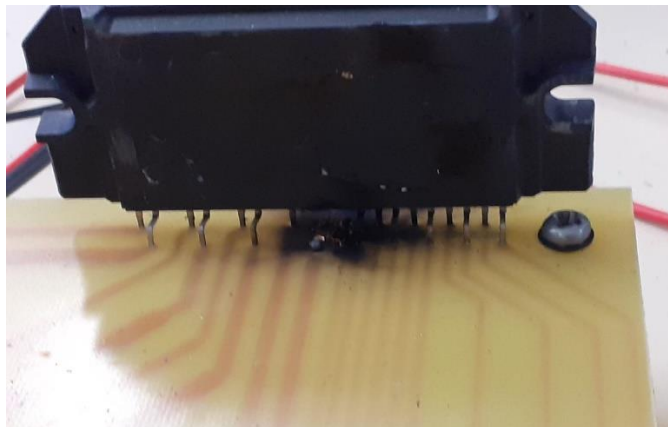


Figura 70: Circuito integrado dañado por una sobrecorriente. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

A continuación, realizamos el cálculo del elemento disipador de calor que será necesario colocar para evitar que un aumento excesivo del calor pueda destruir también el circuito integrado. Lo primero, es obtener el valor de la resistencia térmica entre la unión y el encapsulado ($R_{\theta(J-C)}$) que se encuentra en el datasheet del componente (Figura 71).

Thermal and Mechanical Characteristics

Symbol	Parameter	Min	Typ	Max	Units	Conditions
$R_{\theta(J-C)}$	Thermal resistance, per IGBT	---	1.6	1.8	°C/W	Flat, greased surface. Heatsink compound thermal conductivity 1W/mK
$R_{\theta(J-C)}$	Thermal resistance, per Diode	---	2.2	3		
$R_{\theta(C-S)}$	Thermal resistance, C-S	---	0.1	---		
C_D	Creepage Distance	3.5	---	---	mm	See outline Drawings

Figura 71: Características térmicas del IRAMY20UP60B. [7]

Con este dato obtenido, ya podemos crear el circuito térmico que existirá en el componente, mostrado en la Figura 72.

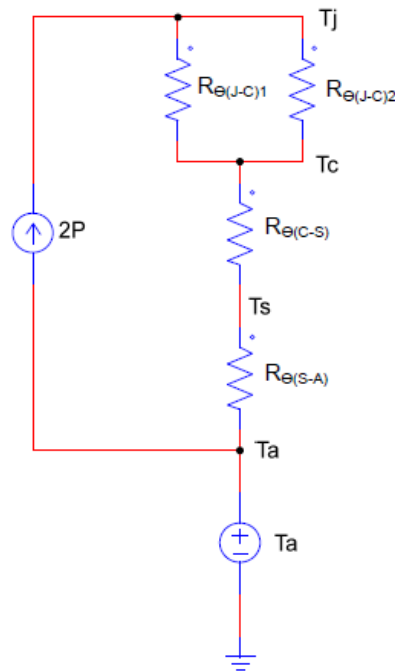


Figura 72: Circuito térmico. (Elaboración propia)

Al analizar el anterior circuito, vemos que existen dos resistencias $R_{\theta(J-C)}$ en paralelo y, esto se debe a que en el catálogo del fabricante nos indica que la resistencia térmica es por cada IGBT. Por tanto, como durante el funcionamiento del propio circuito siempre van a estar activados dos IGBTs, la resistencia se colocará en paralelo.

La fuente de corriente representa la potencia disipada en cada componente, por ello, la potencia de dicha fuente será 2P, ya que en cualquier instante están funcionando dos IGBTs. Esta potencia disipada, se

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

calcula como la caída de tensión entre el colector y emisor del IGBT, que según el catálogo tiene un valor típico de 1,75 V, multiplicado por la corriente máxima que circule por dicho IGBT que, en este caso, será de 2 A, ya que el fusible de protección limita la corriente máxima en el circuito de potencia a este valor.

Por otra parte, la fuente de tensión representa la temperatura en el ambiente.

Para calcular la resistencia térmica del disipador de calor que necesitamos, aplicamos la siguiente fórmula:

$$P = \frac{T_j - T_a}{R_{\theta(J-C)} + R_{\theta(C-S)} + R_{\theta(S-A)}}$$

La temperatura de funcionamiento del circuito integrado está entre -40°C y 150°C, por lo que vamos a definir que la temperatura de la unión (Tj) sea como máximo de 50°C para que no se encuentre en ningún momento comprometida la durabilidad del integrado por el calor.

Después, calculamos la potencia disipada en un IGBT, que será:

$$P = V_{CE(Sat)} * I_{\text{máx}} = 1,75 * 2 = 3,5 \text{ W}$$

, aunque este valor después ha de multiplicarse por 2 ya que trabajan en todo momento dos IGBTs.

Y, por último, el valor del paralelo de las resistencias $R_{\theta(J-C)1}$ y $R_{\theta(J-C)2}$ que como son del mismo valor, es decir, 1,6 °C/W, será la mitad, es decir, 0,8 °C/W.

En cuanto a la resistencia $R_{\theta(C-S)}$ se trata de la resistencia térmica entre la carcasa del componente y el disipador, es decir, en nuestro caso resistencia de la pasta colocada para realizar un buen contacto. Suele tener un valor comprendido entre 0.5°C/W y 3°C/W y en nuestro caso, elegimos 2°C/W.

$$2 * 3,5 = \frac{50 - 25}{0,8 + 2 + R_{\theta(S-A)}}$$

Despejando el valor de $R_{\theta(S-A)}$ obtenemos:

$$\underline{R_{\theta(S-A)} = 0,77^\circ\text{C/W}}$$

El disipador de calor a emplear deberá de tener esa resistencia térmica calculada, pero, en nuestro caso, vamos a reutilizar varios disipadores de calor de antiguas fuentes de alimentación de ordenadores. Para nuestro proyecto, hemos colocado en las dos placas de circuito fabricadas un

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

radiador de aluminio por ambas caras del componente y, además, por una de ellas también se ha colocado un ventilador. De esta forma, se crea también una refrigeración forzada, aumentando el poder de disipación de los radiadores colocados.

El aspecto final de las dos placas de circuito ya instaladas en el bastidor es el siguiente:

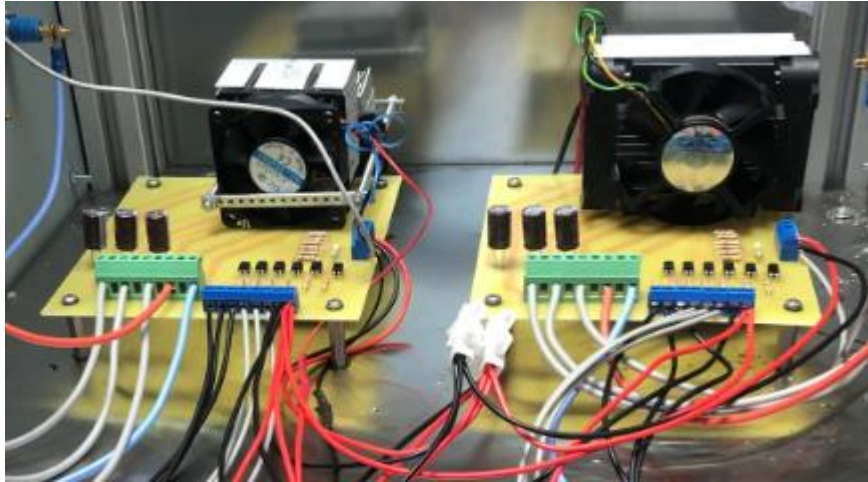


Figura 73: Resultado final de las placas de circuito. (Elaboración propia)

Después de finalizar la construcción de ambas placas y probar de manera manual que funcionaban correctamente, se colocaron fijas en un bastidor, empleando para las conexiones bornes de seguridad en las entradas de potencia y bornes normales (macho - hembra) en las entradas de control y, también en la caja de la Raspberry PI para no tener que estar conectando directamente a los pines de esta. De esta manera, aumenta la fiabilidad del equipo creado para el control de motores síncronos, pues la conexión al motor queda muy simplificada.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

En la Figura 74, se muestra la carátula creada para cada uno de los laterales del bastidor.

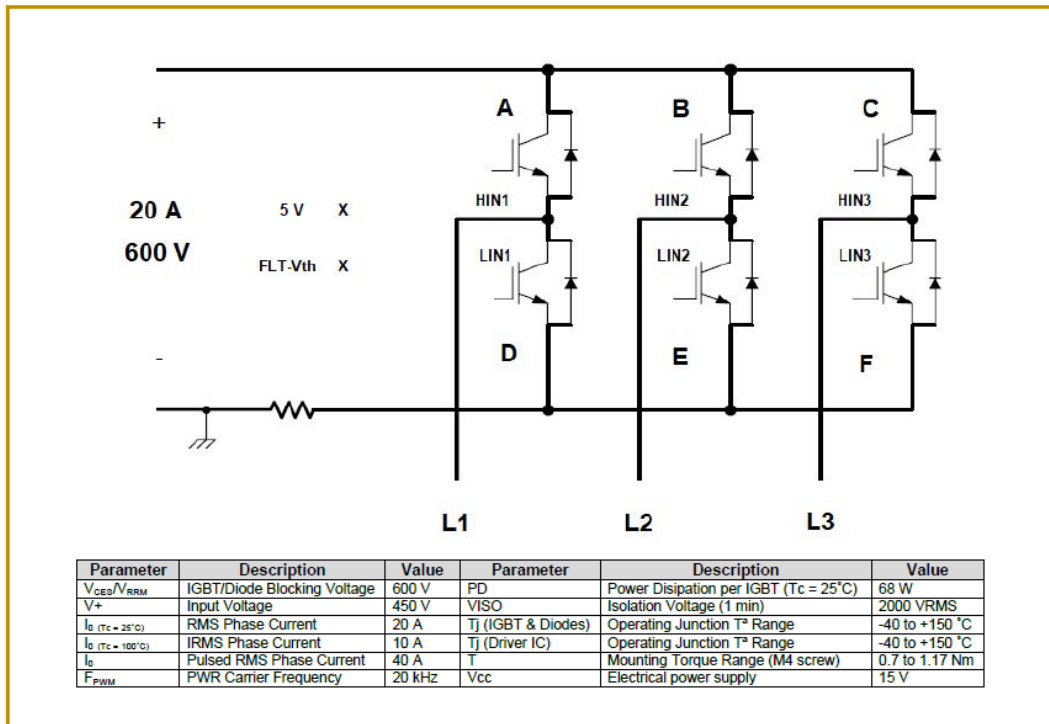


Figura 74: Carátula bastidor. (Elaboración propia)

En las Figuras 75 a 78 se visualiza el aspecto final del bastidor creado para la instalación de las placas de circuito con el controlador.

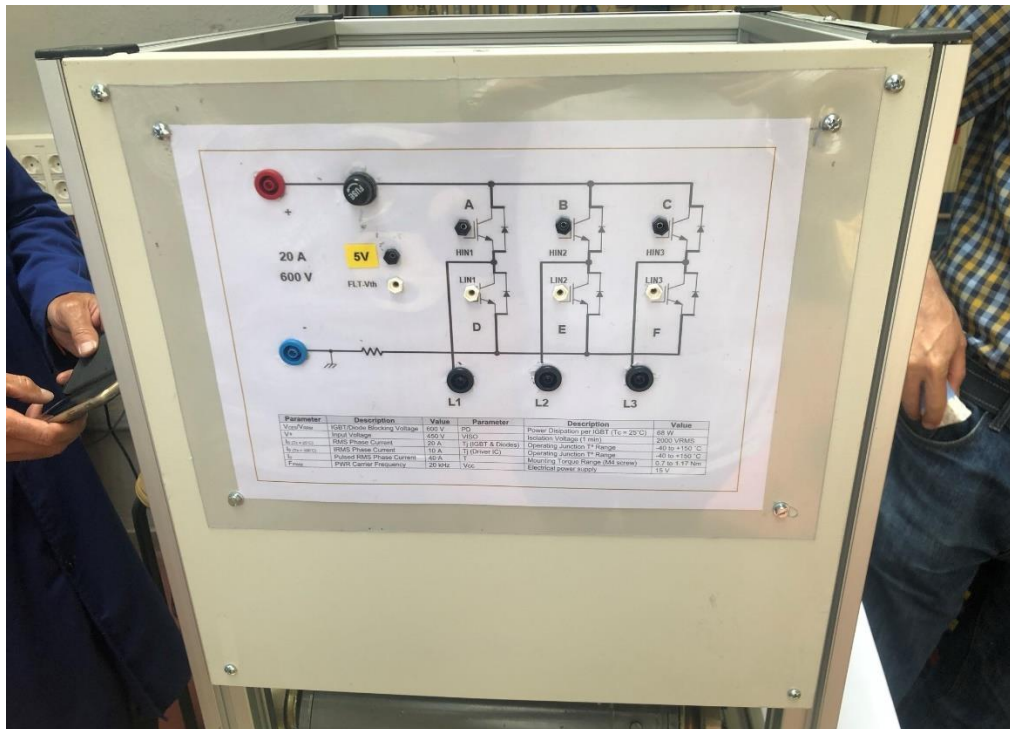


Figura 75: Carátula exterior montada en el bastidor. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

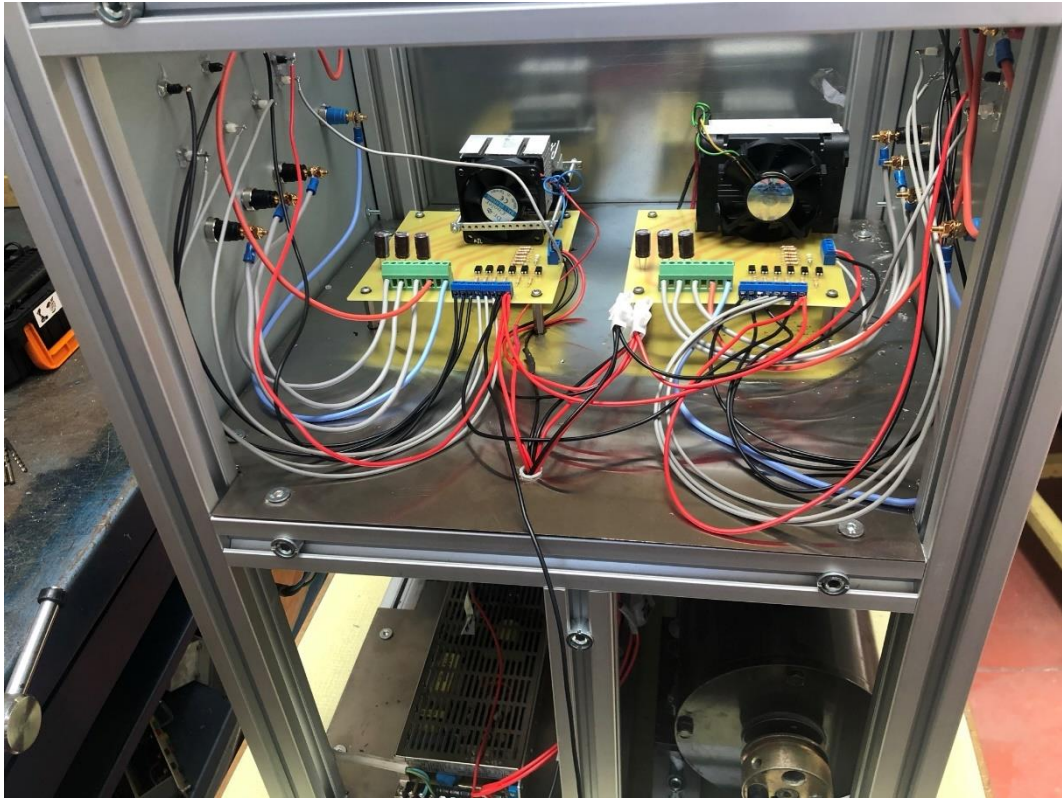


Figura 76: Situación de las dos placas en el interior del bastidor. (Elaboración propia)



Figura 77: Detalle de conexionado de una de las placas con los bornes del bastidor. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

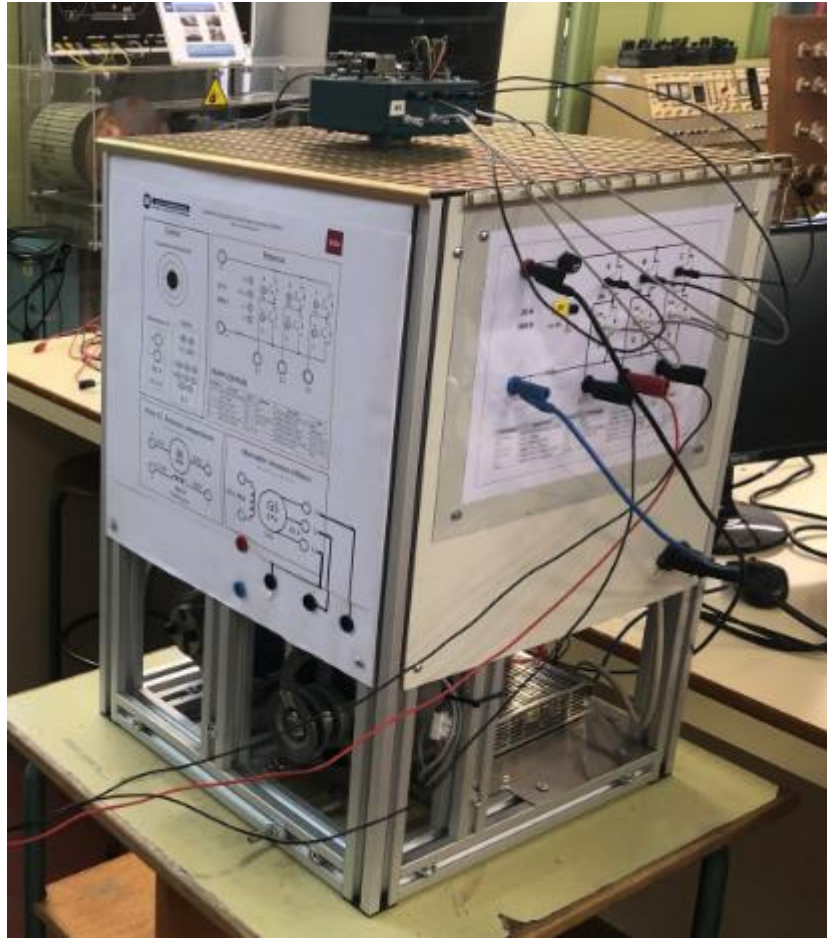


Figura 78: Conexión del bastidor a la Raspberry Pi y a un motor. (Elaboración propia)

El último detalle que queda por concretar para el funcionamiento de todo el controlador implementado es la fuente de alimentación. La fuente empleada se alimenta a 230 V y, tiene tres salidas, de las que vamos a emplear la de 15 V y la de 5 V.

Como las salidas de tensión no tienen el borne negativo común, es decir, son como dos fuentes separadas, podemos unirlos en serie de tal manera que obtenemos una salida de 5 V para alimentar a los optoacopladores, otra de 20 V para alimentar al circuito integrado (hay que tener cuidado en esta alimentación, porque con menos de 17 V no funciona correctamente) y, otra de 15 V que alimenta los ventiladores de los disipadores de calor.

En la figura 79 se muestra en detalle la fuente de alimentación empleada para este proyecto.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW



Figura 79: Fuente de alimentación. (Elaboración propia)

4.4. Resultados obtenidos.

Una vez creado el controlador para dos motores síncronos de diferentes características e instalado correctamente en el bastidor con todas las medidas de protección colocadas, se procede a realizar una primera comprobación de las dos placas de control con el motor paso a paso de 5V. De esta forma, con una fuente de corriente limitada, si existe algún problema no existirán sobrecorrientes que puedan dañar el circuito.

Tras realizar la prueba, se verifica que ambas placas funcionan correctamente, se puede variar la velocidad y, también dejar la velocidad de un motor fija mientras se varía la del otro.

Una vez comprobadas las placas con un motor, se comprueban con una carga resistiva pura, como unas bombillas incandescentes (Figura 80) para comprobar cómo afecta la carga al controlador.



Figura 80: Carga resistiva en estrella. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Al hacer esta comprobación, vemos que a bajas frecuencias la señal de tensión creada no es la que se espera y, por tanto, al probar con un motor a baja velocidad, este no gira correctamente.

En el catálogo del fabricante del circuito integrado IRAMY20UP60B, se indica en una tabla la capacidad de los condensadores a emplear en la salida. En la Figura 81, se muestra esta tabla.

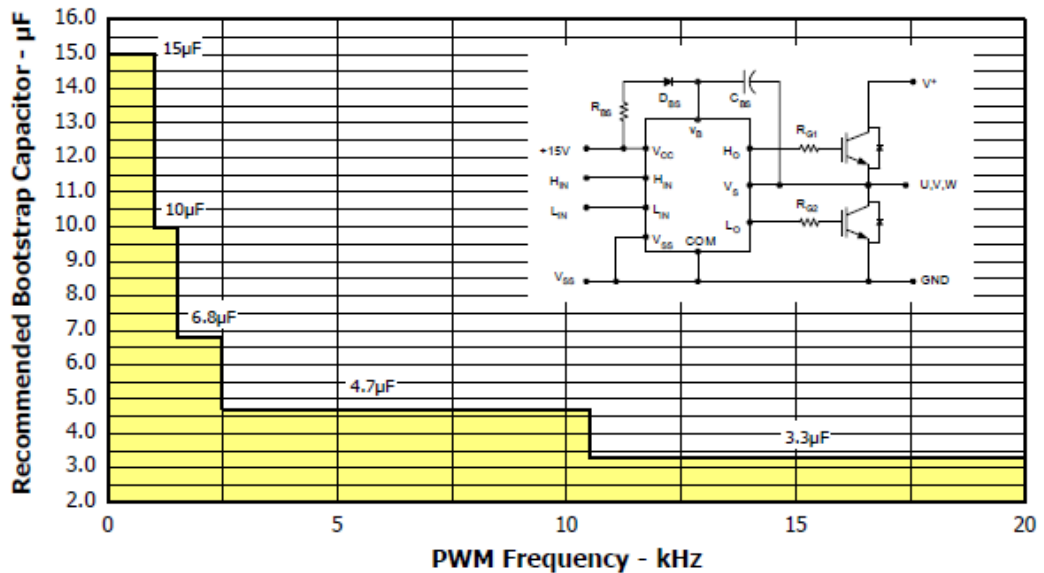


Figura 81: Capacidad recomendada vs frecuencia de funcionamiento. [7]

En la tabla vemos que para frecuencias inferiores a 1 kHz el condensador ha de tener una capacidad de 15 μF y, los empleados tienen una capacidad de 1 μF , por tanto, procedemos a modificar estos condensadores para ver si de esta manera mejora la respuesta.

Finalmente, se han instalado unos condensadores con capacidad de 10 μF , pues se ha aprovechado unos existentes en el laboratorio.

Después de realizar este cambio, se vuelve a probar con la carga resistiva y, el resultado mejora, aunque sigue sin poder funcionar a bajas frecuencias. En la Figura 82, se muestra la gráfica de tensión del osciloscopio al conectar las bombillas con una tensión de alimentación de 75 V y una frecuencia de parpadeo de 10 Hz.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

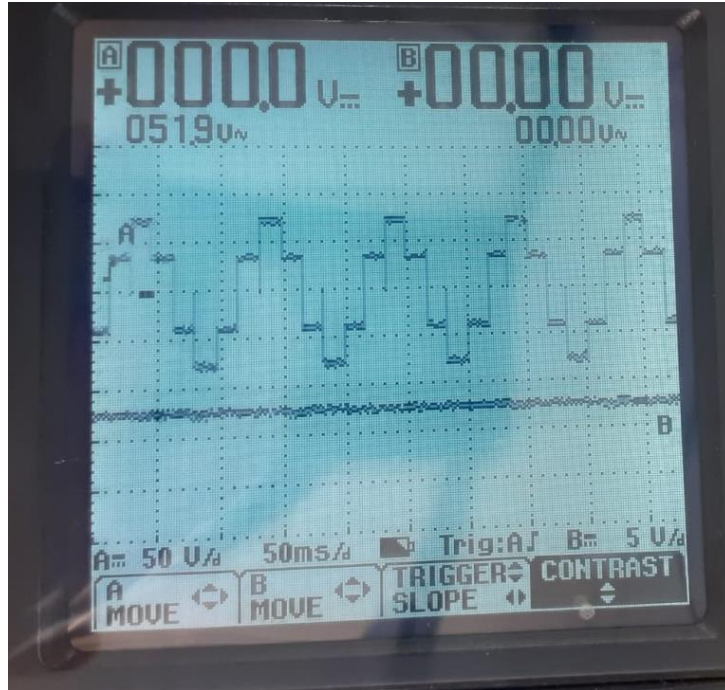


Figura 82: Gráfica de tensión entre fases en las bombillas. (Elaboración propia)

Finalmente, se procede a conectar dos motores síncronos diferentes, los que se ven en las Figuras 83 y 84. Para la prueba se han empleado dos motores con 2 pares de polos, lo que se indicará en el panel frontal para que la velocidad sea la correcta, ya que este parámetro es el que afecta a la frecuencia con la que hay que alimentar el motor.



Figura 83: Motor síncrono 1. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW



Figura 84: Motor síncrono 2. (Elaboración propia)

Al realizar la prueba, se han tomado diferentes medidas, como son la tensión fase fase en ambos motores, la tensión fase neutro en uno de los dos motores y, la corriente de una de las fases de los dos motores también.

En las Figuras 85 – 89, que se encuentran a continuación, podemos ver que estos resultados obtenidos son los esperados y, verificamos que el control de los motores se está realizando de manera adecuada.

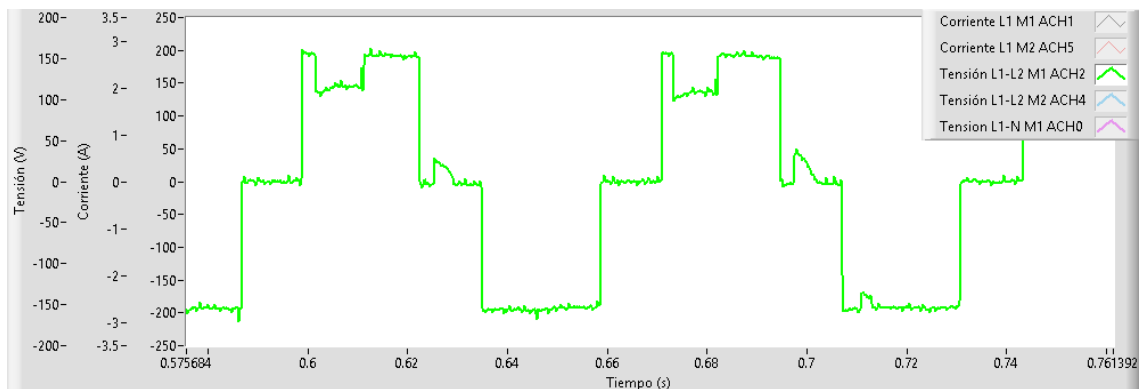


Figura 85: Tensión entre fases del motor 1. (Elaboración propia)

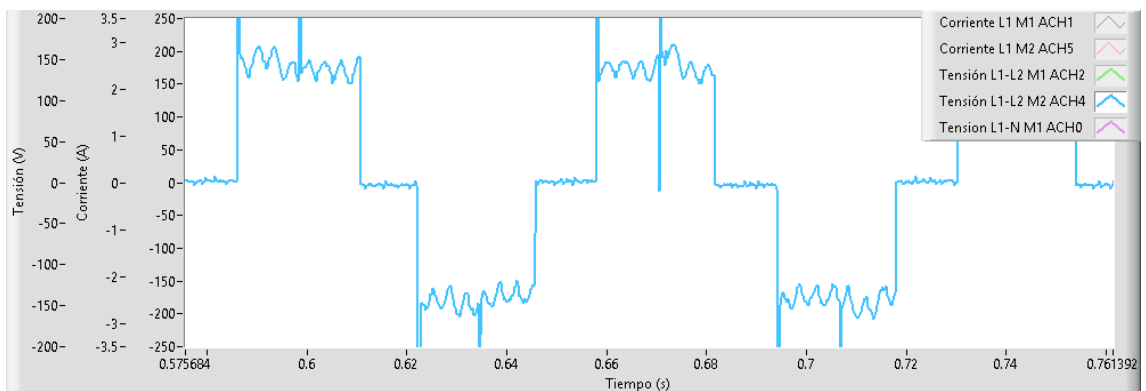


Figura 86: Tensión entre fases del motor 2. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

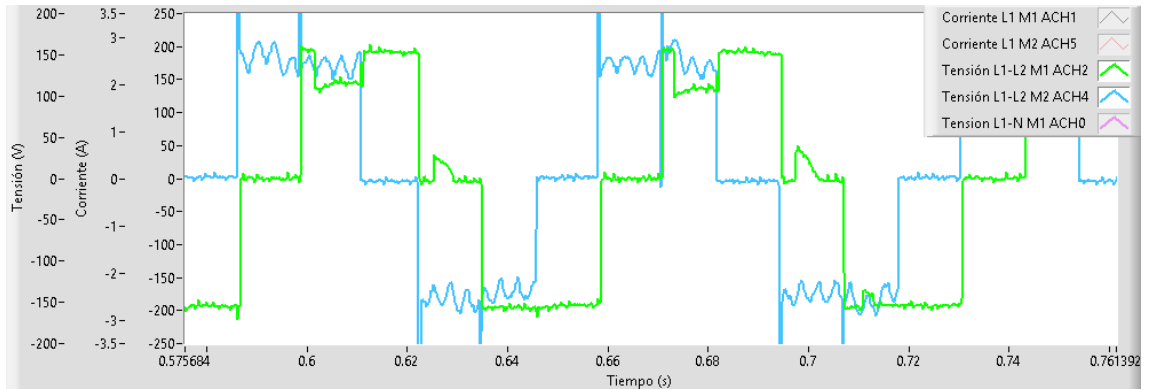


Figura 87: Tensión entre fases del motor 1 y del motor 2. (Elaboración propia)

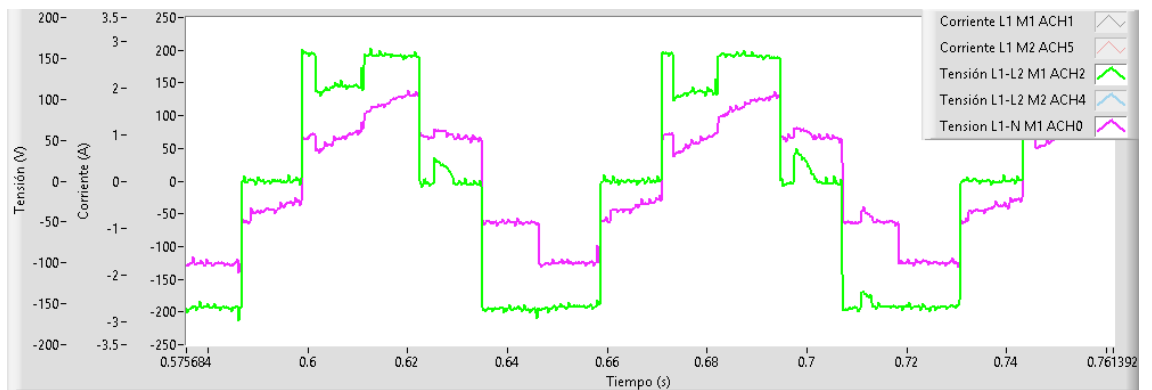


Figura 88: Tensión simple y compuesta del motor 1. (Elaboración propia)

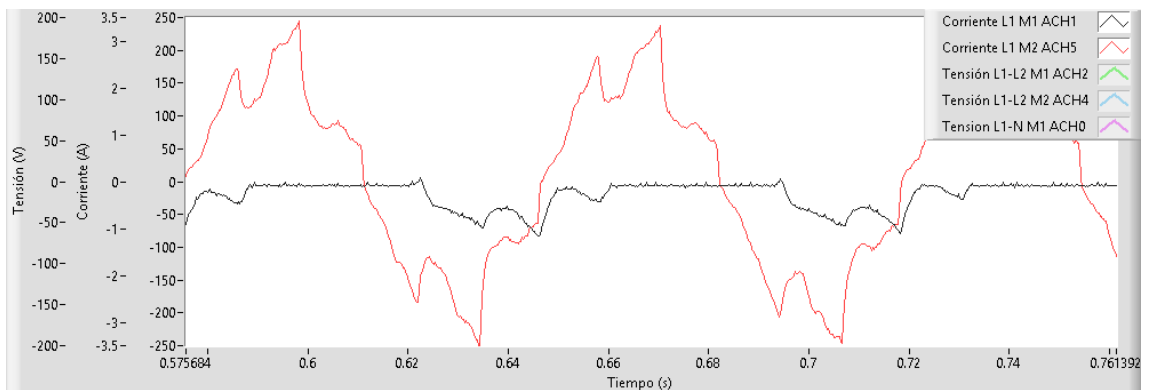


Figura 89: Corrientes del motor 1 y motor 2. (Elaboración propia)

Si nos fijamos en la gráfica de la Figura 89, vemos que la corriente del motor 2 sí que hace una onda senoidal, mientras que por el motor 1 apenas circula corriente. Esto se debe a que el motor 1 está en vacío y el motor 2, está arrastrando otras dos máquinas y un freno, lo que le supone algo de carga al eje del motor.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

En este apartado, hemos comprobado también las velocidades de los dos motores por medio del reloj de velocidad, como se ve en la Figura 90.



Figura 90: Velocidad de los motores 1 y 2. (Elaboración propia)

Por último, si probamos la función de variar la velocidad en un motor mientras que el otro se mantiene constante, vemos que también funciona. En el ejemplo mostrado a continuación, se fija una velocidad de 400 r/min, que al medir con un tacómetro marcaba finalmente sobre 420 r/min. Se introduce con el dial del panel frontal que uno de los motores ha de girar un a 20% menos de velocidad que el otro y, el resultado, visualizado en la Figura XX, vemos que quedará en torno a 320 r/min según la aguja del panel frontal y, 318 r/min según el tacómetro del motor.



Figura 91: Velocidades de giro de los dos motores. (Elaboración propia)

Con esta prueba, que visualizamos el resultado en la Figura 91, comprobamos que el funcionamiento del programa creado y del controlador implementado es correcto.

También, se ha realizado esta medición con el reloj medidor de velocidad (Figura 92), y los resultados obtenidos, son similares a los del marcador digital del tacómetro de los motores.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW



Figura 92: Velocidad de los motores 1 y 2 durante un giro simulado. (Elaboración propia)

Como dato, se ha comprobado que el arranque de los motores empleados ha de hacerse como mínimo a 60 r/min y, si disminuimos de esta velocidad, el motor comienza a fallar debido al controlador utilizado.

Como último experimento, hemos llevado las dos máquinas a una velocidad de 700 r/min. En este caso, las tensiones de fase han sido más elevadas, pues a medida que aumentamos velocidad, hay que ir elevando la tensión de alimentación. En la siguiente imagen (Figura 93), se muestra la tensión entre fases del motor 1 y, en la siguiente se comprueba que la velocidad es la indicada en el panel frontal, midiendo la frecuencia de la señal.

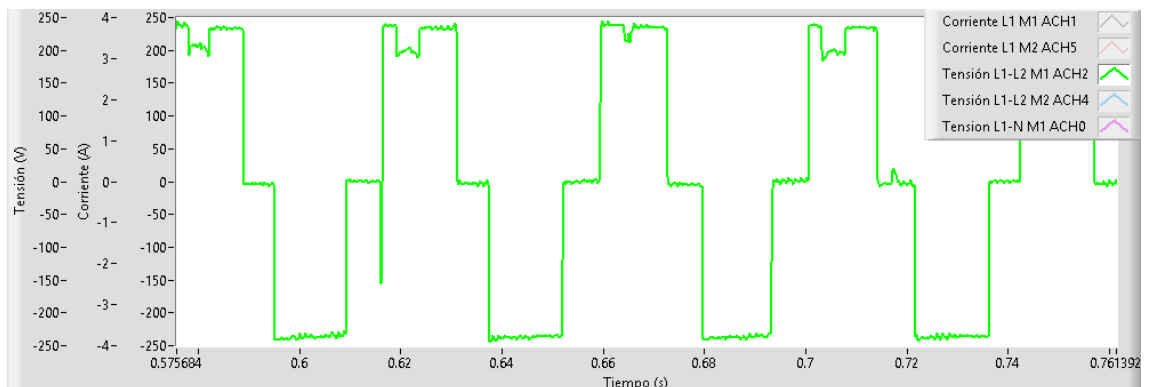


Figura 93: Tensión entre fases del motor 1 a 700 r/min. (Elaboración propia)

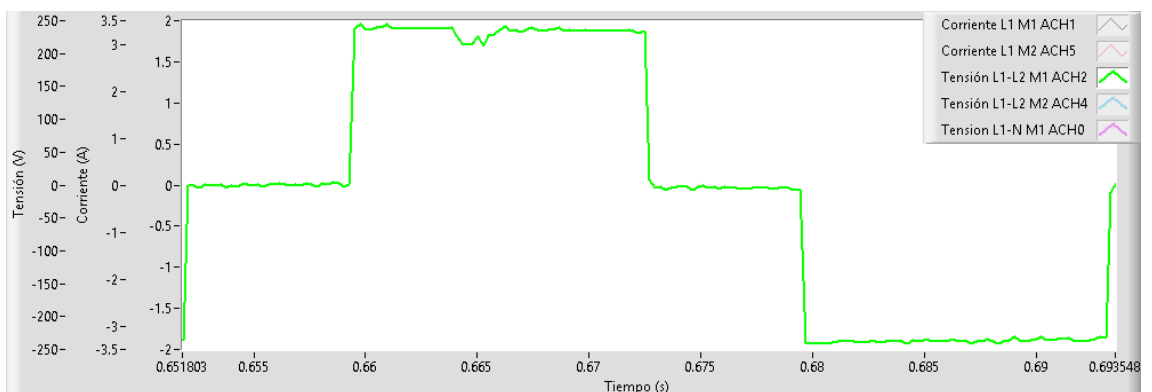


Figura 94: Período de la tensión del motor 1 a 700 r/min. (Elaboración propia)

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

Comprobamos la velocidad del motor por medio de la gráfica de la Figura 94:

- Período: $T = 0.693548 - 0.651803 = 0.041745$ s.
- Frecuencia: $F = 1/T = 1/0.041745 = 23.955$ Hz
- Velocidad: $\omega = \frac{60 * F}{n^{\circ} \text{ pares de polos}} = \frac{60 * 23.955}{2} = 718.65$ r/min.

Si medimos la velocidad del eje con el reloj de velocidad (Figura 95), obtenemos una lectura de 700 r/min, algo inferior a la velocidad obtenida en la frecuencia debido a que era necesario aumentar algo más la tensión del motor para que alcanzase la velocidad marcada, pero no se elevó más esta porque ya estaba en 250 V, que es ligeramente superior a su valor máximo.



Figura 95: Velocidad del motor 2. (Elaboración propia)

También se ha probado el sistema con un alternador de un coche y uno de los motores síncronos del banco de motores. En la Figura 96, se muestra el alternador de coche empleado, cuya alimentación y excitación se realizan a 12 V.

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

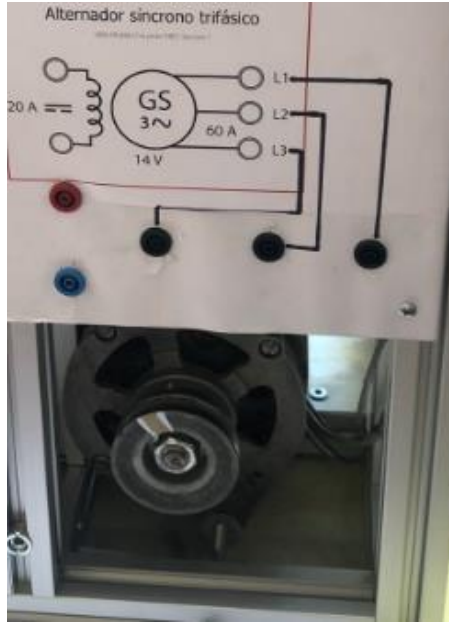


Figura 96: Alternador de coche empleado en el proyecto. (Elaboración propia)

Una última función que también está disponible en el panel frontal del bastidor es la de medir la temperatura interna del circuito integrado, que se realiza midiendo la tensión entre el borne de 5 V y el de Vth, que contiene la señal de temperatura ofrecida por el integrado. Con el valor leído en un polímetro, se entra en la tabla de la siguiente Figura y, se obtiene la temperatura a la que se encuentra funcionando el integrado.

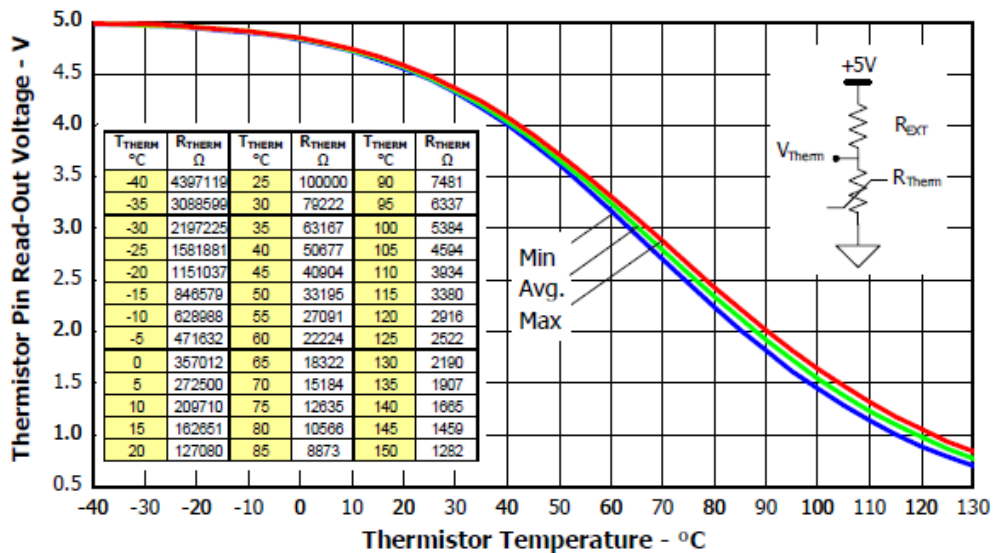


Figura 97: Gráfica de temperatura del IRAMY20UP60B. [7]

Como en nuestro montaje medimos entre 5V y Vth y, la tabla nos indica que hay que medir entre Vth y GND, para poder entrar en la tabla con el valor leído hay que realizar la siguiente operación (5 - V_{leído}).

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

4.5. Estudio económico.

Una vez finalizado el trabajo de diseño y construcción de las placas de circuito impreso y el montaje de estas en el bastidor se ha calculado el coste de los materiales empleados, que se encuentra desglosado en la Tabla 6, a continuación.

COMPONENTE	Nº de Unidades	Precio Unidad (€)	Precio Final (€)
IRAMY20UP60B	2	20.19 €	40.38 €
Condensador 10 µF	6	0.55 €	3.30 €
Resistencia 1kΩ	20	0.083 €	1.66 €
Resistencia 12kΩ	2	0.11 €	0.22 €
Borneo Potencia de 3 entradas	6	1.192 €	7.15 €
Borneo Control de 2 entradas	12	0.20 €	2.40 €
Optoacopladores	12	0.18 €	2.16 €
Disipador de calor 80x50x40 mm	4	7.20 €	28.80 €
Ventilador para radiador	2	26.36 €	52.72 €
Conector banana hembra para entradas de potencia	10	3.70 €	37.00 €
Conector banana macho para cables de control	24	1.32 €	31.68 €
Conector banana hembra para entradas de control	24	1.35 €	32.40 €
Fuente de alimentación 15V, 5V	1	33.00 €	33.00 €
Raspberry PI 4 2GB Model B	1	43.00 €	43.00 €
Placa de cobre para impresión	2	4.20 €	8.40 €
Cable 2.5mm ² , 10m	10	0.55 €	5.50 €
TOTAL COSTE MATERIALES			329.77 €

Tabla 5: Coste de materiales del proyecto. (Elaboración propia)

Además del coste de los materiales empleados, también surgen otros costes aparte, como sería el de la mano de obra empleada y el diseño del programa creado. En la Tabla 7, se describen estos otros costes asociados al proyecto.

CONCEPTO	Nº de horas	Precio Hora (€)	Precio Final (€)
Diseño del programa ingeniero junior	6	36.30 €	217.30 €
Mano de obra	12	36.30 €	435.60 €
TOTAL COSTE HORAS			653.40 €

Tabla 6: Otros costes del proyecto. (Elaboración propia)

Finalmente, podemos ver que el coste total del proyecto montado asciende a 329.77€ + 653.40€ = 983.17€ (IVA incluido en todos los precios).

5. Conclusiones obtenidas y trabajos relacionados.

5.1. Conclusiones.

Una vez realizado este trabajo, las conclusiones a las que podemos llegar son las siguientes:

Las metas propuestas en el primer capítulo quedan desarrolladas de forma adecuada. Se ha logrado crear un programa que permite la variación de la velocidad de los motores, permitiendo incluso que uno de ellos la mantenga constante mientras el otro la disminuye. Además, se ha construido un bastidor que permite alojar de forma robusta y segura el sistema de control.

A su vez, se ha alcanzado el objetivo de controlar dos motores de distintas características. Como se ha descrito con anterioridad a lo largo de los capítulos tercero y cuarto, se han empleado al mismo tiempo un motor síncrono de 230 V y 2 pares de polos y un alternador de un coche de 12 V con 6 pares de polos; también se ha utilizado un motor paso a paso de 5V y 5 pares de polos.

Debido a la gran carga práctica de este trabajo se ha conseguido entender de forma precisa el funcionamiento del controlador, pero también ha permitido ampliar conocimientos sobre la realización de la construcción y refrigeración de placas de circuito impreso.

Por último, se ha instalado un control manual del programa desde el propio bastidor, por medio de cinco entradas digitales a la Raspberry PI. Este control facilita un manejo rápido y sencillo del programa, pues mientras se encuentre cargado en la Raspberry PI no es necesario servirse del ordenador para pequeñas pruebas. Sin embargo, para un ajuste mayor de los diversos parámetros del programa sí que es imprescindible su uso, por ejemplo, para configurar el número de pares de polos de los motores.

5.2. Trabajos relacionados.

A partir de este trabajo, surgen diversos temas sobre los que se pueden aún seguir ampliando el trabajo realizado:

Control de la temperatura de los circuitos integrados. Para ello, sería conveniente emplear un medidor de corriente continua como el empleado para los tacómetros, ya que permite introducir una curva de medida con diferentes pendientes. Otra opción es monitorizar en el ordenador estas temperaturas por medio del programa de lectura de datos.

Realizar un control tensión frecuencia de forma automática, pues en las prácticas realizadas tenemos que ir aumentando la tensión de alimentación del motor a medida que aumenta la frecuencia.

Probar con otros circuitos integrados de IGBTs para ver si la respuesta del motor a bajas velocidades mejora.

Utilizar un encoder en el motor para medir la velocidad y hacer la realimentación del lazo de frecuencia en el bloque variador de frecuencia. De este modo, nos aseguramos de que la frecuencia aumenta a la vez que lo hace en el motor. Es decir, garantizamos que la realimentación sea real y no solamente desde el valor del programa.

Controlar la caja de pulsadores por medio de una tarjeta de tipo Arduino o Esp32 para recibir las señales de entrada. De esta forma, podremos utilizar un potenciómetro para controlar la velocidad de los motores y otro para la variación de esta solamente en uno de ellos ya que estos dispositivos cuentan con entradas analógicas. La comunicación de datos entre la tarjeta empleada y la Raspberry PI se puede llevar a cabo mediante la conexión por puerto serie.

Índice de figuras.

Figura 1: Tipos de excitación de las máquinas de Corriente Continua. [1]	11
Figura 2: Circuito equivalente de una Máquina Síncrona. [1]	12
Figura 3: Circuito equivalente del motor asíncrono: a) exacto; b) aproximado. [1]	14
Figura 4: Motor de Reluctancia Conmutada. [2]	15
Figura 5: Motor Paso a Paso. [3].....	16
Figura 6: Máquina de Flujo Axial. [4]	18
Figura 7: Secuencia de alimentación del Motor Paso a Paso. [3].....	19
Figura 8: Motor Síncrono. [2]	20
Figura 9: Conexión del Motor. [2]	20
Figura 10: Puente de IGBTs para control. (Elaboración propia)	21
Figura 11: Secuencia de pasos del motor. [5].....	22
Figura 12: Simulación del puente de IGBTs. (Elaboración propia).....	23
Figura 13: Simulación de las señales de control de los IGBTs. (Elaboración propia).....	24
Figura 14: Señal de la red comparada con la señal de salida del puente. (Elaboración propia).....	24
Figura 15: Simulación de tensiones compuestas y simples entre fases. (Elaboración propia).....	25
Figura 16: Simulación del funcionamiento del motor. (Elaboración propia) ...	26
Figura 17: Señal de la red comparada con la señal de salida del puente. (Elaboración propia).....	26
Figura 18: Resultados simulación del motor. (Elaboración propia)	27
Figura 19: Corriente i_a (A). (Elaboración propia).....	28
Figura 20: GPIO Raspberry PI 4. [6].....	29
Figura 21: Circuito Integrado en el IRAMY20UP60B. [7]	30
Figura 22: Valores Máximos del IRAMY20UP60B. [7].....	31
Figura 23: Valores recomendados de tensión. [7]	31
Figura 24: Tabla de verdad del controlador IRAMR20UP60B. [7].....	31
Figura 25: Configuración de la Raspberry PI. (Elaboración propia).....	33
Figura 26: Conexión del ordenador a la Raspberry PI online. (Elaboración propia).....	34
Figura 27: Recorte del diagrama de bloques. (Elaboración propia).....	35
Figura 28: Bloque de secuencia de pulsos. (Elaboración propia)	36
Figura 29: Señales de salida del bloque de pulsos. (Elaboración propia).....	37
Figura 30: Entradas de Giro Horario y Giro Antihorario. (Elaboración propia) .	37
Figura 31: Señales de entrada al bloque de pulsos. (Elaboración propia).....	38
Figura 32: Señales de control del motor. (Elaboración propia)	39
Figura 33: Bloque Wait Until Next ms Multiple. (Elaboración propia)	40
Figura 34: Bloque Variador de Frecuencia. (Elaboración propia).....	41

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

Figura 35: Conexión de entradas y salidas al bloque variador de frecuencia. (Elaboración propia).....	42
Figura 36: Código del bloque variador de frecuencia. [5].....	44
Figura 37: Visualización de la velocidad en r/min. (Elaboración propia).....	45
Figura 38: Panel Frontal. (Elaboración propia)	45
Figura 39: Diagrama de Bloques. (Elaboración propia)	46
Figura 40: Señal de disparo. (Elaboración propia)	47
Figura 41: Circuito de aproximación al IRAMY20UP60B. (Elaboración propia)	48
Figura 42: Tensión fase – neutro en giro antihorario. (Elaboración propia)....	48
Figura 43: Tensión fase – neutro en giro horario. (Elaboración propia)	49
Figura 44: Tensión fase – fase. (Elaboración propia)	49
Figura 45: Señales de disparo de los transistores. (Elaboración propia)	50
Figura 46: Tensión compuesta, simple y entre fase y tierra. (Elaboración propia)	50
Figura 47: Prueba del IRAMY20UP60B. (Elaboración propia).....	51
Figura 48: Motor paso a paso de 5V alimentado. (Elaboración propia)	51
Figura 49: Alimentación del motor paso a paso de 5V. (Elaboración propia) .	52
Figura 50: Motor síncrono como paso a paso con control de todas las bobinas. [8] [9]	53
Figura 51: Motor síncrono como paso a paso con bobinas en estrella. [8] [9]	54
Figura 52: Bloque de activación de salidas nuevo. (Elaboración propia).....	56
Figura 53: Bloque variador de frecuencia. (Elaboración propia).....	57
Figura 54: Control de la velocidad. (Elaboración propia)	57
Figura 55: Panel frontal de la aplicación. (Elaboración propia).....	58
Figura 56: Diagrama de Bloques. (Elaboración propia)	59
Figura 57: Control manual del programa. (Elaboración propia)	60
Figura 58: Esquema eléctrico. (Elaboración propia)	62
Figura 59: Conexión Optoacopladores. (Elaboración propia)	63
Figura 60: Separación eléctrica entre conductores. [10]	65
Figura 61: Datasheet del bornero. (RS Componentes)	65
Figura 62: Descripción eléctrica de los bornes de la parte de control. (Amazon)	66
Figura 63: Características eléctricas de los optoacopladores. (Amazon)	66
Figura 64: Fotolito de la Cara de Componentes. (Elaboración propia)	67
Figura 65: Fotolito Máscara de Soldadura Superior. (Elaboración propia).....	68
Figura 66: Fotolito Cara de Soldadura. (Elaboración propia).....	68
Figura 67: Fotolito Máscara de Soldadura Inferior. (Elaboración propia).....	69
Figura 68: Resultado final de la PCB. (Elaboración propia)	69
Figura 69: Resultado final de la PCB. (Elaboración propia).....	70
Figura 70: Circuito integrado dañado por una sobrecorriente. (Elaboración propia).....	70

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

Figura 71: Características térmicas del IRAMY20UP60B. [7].....	71
Figura 72: Circuito térmico. (Elaboración propia).....	71
Figura 73: Resultado final de las placas de circuito. (Elaboración propia)	73
Figura 74: Carátula bastidor. (Elaboración propia)	74
Figura 75: Carátula exterior montada en el bastidor. (Elaboración propia) ...	74
Figura 76: Situación de las dos placas en el interior del bastidor. (Elaboración propia).....	75
Figura 77: Detalle de conexionado de una de las placas con los bornes del bastidor. (Elaboración propia).....	75
Figura 78: Conexión del bastidor a la Raspberry PI y a un motor. (Elaboración propia).....	76
Figura 79: Fuente de alimentación. (Elaboración propia).....	77
Figura 80: Carga resistiva en estrella. (Elaboración propia).....	77
Figura 81: Capacidad recomendada vs frecuencia de funcionamiento. [7] ...	78
Figura 82: Gráfica de tensión entre fases en las bombillas. (Elaboración propia)	79
Figura 83: Motor síncrono 1. (Elaboración propia).....	79
Figura 84: Motor síncrono 2. (Elaboración propia).....	80
Figura 85: Tensión entre fases del motor 1. (Elaboración propia).....	80
Figura 86: Tensión entre fases del motor 2. (Elaboración propia).....	80
Figura 87: Tensión entre fases del motor 1 y del motor 2. (Elaboración propia)	81
Figura 88: Tensión simple y compuesta del motor 1. (Elaboración propia) ...	81
Figura 89: Corrientes del motor 1 y motor 2. (Elaboración propia).....	81
Figura 90: Velocidad de los motores 1 y 2. (Elaboración propia).....	82
Figura 91: Velocidades de giro de los dos motores. (Elaboración propia).....	82
Figura 92: Velocidad de los motores 1 y 2 durante un giro simulado. (Elaboración propia).....	83
Figura 93: Tensión entre fases del motor 1 a 700 r/min. (Elaboración propia)	83
Figura 94: Período de la tensión del motor 1 a 700 r/min. (Elaboración propia)	83
Figura 95: Velocidad del motor 2. (Elaboración propia).....	84
Figura 96: Alternador de coche empleado en el proyecto. (Elaboración propia)	85
Figura 97: Gráfica de temperatura del IRAMY20UP60B. [7].....	85

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

Índice de tablas.

Tabla 1: Secuencia de alimentación del motor. (Elaboración propia)	22
Tabla 2: Secuencia de conmutación de los IGBTs. (Elaboración propia).....	22
Tabla 3: Secuencia de conmutación de los IGBTs (lógica negativa). (Elaboración propia).....	32
Tabla 4: Variables de entrada y salida del bloque de secuencia. (Elaboración propia).....	40
Tabla 5: Coste de materiales del proyecto. (Elaboración propia).....	86
Tabla 6: Otros costes del proyecto. (Elaboración propia).....	86

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES
CARACTERÍSTICAS MEDIANTE LABVIEW

Bibliografía

- [1] J. Fraile Mora, Máquinas Eléctricas, Aravaca (Madrid): Mc Graw Hill, 2003.
- [2] tuveras.com - El Motor Síncrono, «El Motor Síncrono,» [En línea]. Available: <http://endrino.pntic.mec.es/jhem0027/motorsincrono/motorsincrono.htm>. [Último acceso: 20 Abril 2021].
- [3] M. Rincon Delgadillo, «Scribd.com,» [En línea]. Available: <https://es.scribd.com/doc/53144094/Motores-paso-a-paso>. [Último acceso: 15 Abril 2021].
- [4] Sociedad Técnica de Automoción, El vehículo eléctrico. Desafíos tecnológicos, infraestructuras y oportunidades de negocio, Barcelona: Libbooks Barcelons, S.L.L., 2011.
- [5] D. Díez Calvo, «UVaDOC,» 2019. [En línea]. Available: <https://uvadoc.uva.es/handle/10324/36794>. [Último acceso: 6 Marzo 2021].
- [6] RASPBERRY PI FOUNDATION, «Raspberry Pi 4 Computer Model B,» Enero 2021. [En línea]. Available: <https://datasheets.raspberrypi.org/rpi4/raspberry-pi-4-product-brief.pdf>. [Último acceso: 20 Abril 2021].
- [7] Internacional Rectifier, «irf.com,» [En línea]. Available: <https://www.infineon.com/dgdl/iramy20up60b.pdf?fileId=5546d462533600a4015355da3e391879>. [Último acceso: 2 Mayo 2021].
- [8] F. J. Alonso Ripoll, «Apuntes Sistemas de Transporte de Tracción Eléctrica,» 2018. [En línea]. Available: <https://campusvirtual.uva.es/>. [Último acceso: 20 Febrero 2019].
- [9] UVA, «Campus Virtual - Electrónica Industrial,» Electrónica Industrial, Marzo 2021. [En línea]. Available: <https://campusvirtual.uva.es/>. [Último acceso: 21 Abril 2021].
- [10] «Alciro.org,» [En línea]. Available: http://www.alciro.org/alciro/Plotter-Router-Fresadora-CNC_1/Alimentacion-motores-paso-a-paso_56.htm. [Último acceso: 15 Abril 2021].

CONTROL SINCRONIZADO DE DOS MOTORES SÍNCRONOS DE DIFERENTES CARACTERÍSTICAS MEDIANTE LABVIEW

- [11] R. Vallejo Luis, «UVaDOC,» Julio 2019. [En línea]. Available: <https://uvadoc.uva.es/handle/10324/37776>. [Último acceso: Mayo 2021].
- [12] A. Cadenas, «youtube,» Septiembre 2017. [En línea]. Available: <https://www.youtube.com/channel/UC4Y2ssTA2KamKP1h0IWebtg>. [Último acceso: 26 Abril 2021].
- [13] M. San Martín Ojeda, «Aulamoisan,» 13 Junio 2020. [En línea]. Available: <https://www.aulamoisan.com/raspberry-beaglebone-labview>. [Último acceso: Abril 2021].
- [14] UVa, «Campus Virtual - Máquinas Eléctricas I y II,» Máquinas Eléctricas, Marzo 2018. [En línea]. Available: <https://campusvirtual.uva.es/>. [Último acceso: 10 Mayo 2021].
- [15] UVa, «Campus Virtual - Motores Especiales,» Motores Especiales, Marzo 2019. [En línea]. Available: Máquinas Eléctricas I y II. [Último acceso: 10 Mayo 2021].
- [16] UVa, «Campus Virtual - Métodos y Herramientas de Diseño Electrónico,» Métodos y Herramientas de Diseño Electrónico, Marzo 2020. [En línea]. Available: Motores Especiales. [Último acceso: 10 Mayo 2021].
- [17] RASPBERRY PI FOUNDATION, «Raspberry Pi 4 Model B,» Junio 2019. [En línea]. Available: <https://www.raspberrypi.org>. [Último acceso: 20 Abril 2021].
- [18] MATLAB, «Mathworks,» 21 Abril 2021. [En línea]. Available: <https://es.mathworks.com/>.