

GRADO EN INGENIERIA ELECTRONICA  
INDUSTRIAL Y AUTOMATICA  
**TRABAJO FIN DE GRADO**

***DISEÑO, DESARROLLO E  
IMPLEMENTACIÓN DEL CONTROL DE  
UNA LAVADORA***

**Alumna:** Pérez, Fortes, Nerea

**Director:** Sainz de Murieta, Mangado, Joseba Andoni

**Curso:** 2017 - 2018

**Fecha:** Bilbao, 19, junio, 2018

## RESUMEN

Debida a la obsolescencia programada que existe en las lavadoras de hoy en día, en este proyecto se diseñará su control con un software libre como es el de Arduino. Puesto que se requieren controlar varios accionamientos (motor, válvulas, bomba de agua...) se empezará con el diseño de control del motor ya que es el más complejo.

El primer paso es decidir qué tipo de control se va a llevar a cabo, control en corriente alterna o en continua. Una vez analizadas las dos alternativas se ha decidido realizar un control en continua ya que es más eficiente. El siguiente paso será diseñar el circuito de potencia para el microcontrolador elegido. Basado en estudios previos, dada la resolución de la señal PWM requerida para este motor, se utilizará el Arduino Mega.

Gaur egun garbitzaileetan egoten den zaharkitzapen programatua dela eta, proiektu honetan diseinatuko da garbitzaile baten kontrola. Kontrol honek, software libre batekin egingo da (Arduino). Garbitzaile baten parterik garrantzitsuena motorra da, beraz, proiektu honek motorean zentratuko da.

Lehenik eta behin ze kontrol mota aukeratuko den erabaki behar da, hau da, korrante zuzenean edo korrante alternoan. Bi aukerak aztertu eta gero, erabaki da korrante zuzenean egitea kontrola efizienteagoa delako. Hurrengo pausoa potentzia zirkuitua diseinatzea da, kontuan izanda ze mikrokontrolatzaile erabiliko den (Kasu honetan Arduino Mega erabiliko da).

Nowadays in a lot of devices, like washing machines, exist planned obsolescence. Because of this, in this Project is going to be design the control of a washing machine with a free software like Arduino is. Considering that this device is composed with a engine, valves, a water pump...the Project is going to focus on the control of the engine as is the most complicated part of the machine.

At first, it is going to decide what kind of control will be choose (control in direct current or alternating current). Secondly, it is going to design the power circuit and study what is the best Arduino for this project.

## ÍNDICE DE CONTENIDO

1. INTRODUCCION .....	8
1.1 OBJETIVOS .....	8
1.1.1 Objetivos sociales y de medio ambientales:.....	8
1.1.2 Objetivos técnicos:.....	8
1.1.3 Objetivos personales: .....	9
1.2 JUSTIFICACIÓN .....	9
2. ANTECEDENTES.....	10
2.1 ESTADO DEL ARTE .....	10
2.1.1 Electrónica de potencia .....	10
2.1.2 Sistemas digitales de control - sistemas embebidos.....	10
2.1.3 Motores universales de corriente continua.....	11
2.1.4 Instrumentación electrónica .....	13
2.1.5 Software open-Source .....	14
3. MARCO TEÓRICO .....	15
3.1 Descripción general .....	15
3.2 Obsolescencia programada .....	15
3.2.1 Origen.....	16
3.2.2 Tipos de obsolescencia.....	16
3.2.3 Consecuencias .....	17
3.2.4 Alternativas .....	20
3.3 Motor universal.....	22
3.3.1 Principio de funcionamiento en corriente continua.....	22
3.3.2 Principio de funcionamiento en corriente alterna .....	22
3.3.3 Bobinado de compensación.....	23
3.4 Circuito de potencia .....	24
3.4.1 Transistor IGBT .....	24
3.4.2 Driver MC33153P4.....	26
3.4.3 Transformador.....	27
3.4.4 Rectificador .....	29
3.4.5 Regulador de tensión MC78L12APG .....	32
3.4.6 Descripción del circuito .....	33
3.5 Microcontrolador: Arduino. Control PID .....	34
3.5.1 Arduino Uno.....	35
3.5.2 Arduino Mega .....	37
3.5.3 Control PID .....	41
4. DISEÑO .....	44

4.1 Motor .....	44
4.1.1 Bornes del motor .....	44
4.1.2 Tacómetro.....	45
4.2 Circuito de potencia .....	47
4.3 Arduino. Diseño del control del motor .....	53
4.3.1 Arduino Uno.....	53
4.3.2 Arduino Mega .....	56
5. RESULTADOS .....	63
5.1 Conclusiones .....	63
5.2 Líneas de desarrollo futuras .....	64
6. PLANOS .....	66
6.1 Circuito de potencia .....	66
6.1.1 Primera versión .....	66
6.1.2 Versión definitiva.....	66
6.2 Conexiones del motor .....	67
7. PROGRAMAS .....	68
7.1 Variación de la velocidad del motor en lazo abierto con arduino uno.....	68
7.2 Variación de la velocidad del motor en lazo abierto con arduino mega.....	68
7.3 Variación de la velocidad del motor en lazo cerrado con arduino mega.....	70
8. PRESUPUESTO.....	72
9. ANEXOS .....	73
9.1 Arduino Mega and arduino Uno- Clock distribution.....	73
9.2 Reset and interrupt Vectors.....	74
9.3 16bit Timer Blck Diagram .....	75
9.4 CTC Mode- Timing diagram .....	76
9.5 Phase correct PWM – Timing diagram.....	76
9.6 Phase and frequency correct PWMmode – Timing diagram .....	76
10. BIBLIOGRAFIA .....	77

## ÍNDICE DE FIGURAS

Figura 1 - Electronica de pontecia. Diagrama de bloques.....	10
Figura 2 - Esquema de bloques de un uP- CISC .....	11
Figura 3 - Esquema de bloques de un uC-CISC.....	11
Figura 4 - Estructura interna de un motor universal.....	12
Figura 5 - Diagrama de bloques del acondicionamiento de una señal .....	13
Figura 6 - Diagrama de bloques global .....	15
Figura 7 - Simbología de un IGBT como un BJT .....	24
Figura 8 - Simbología de un IGBT como un MOSFET .....	24
Figura 9 - Curvas características del IGBT .....	25
Figura 10 - Circuito equivalente simplificado de un IGBT.....	25
Figura 11- Pines del Driver MC33153 .....	26
Figura 12- Conexionado para una aplicación básica suministrada por Motorola .....	27
Figura 13- Transformador .....	28
Figura 14- Rectificador de media onda .....	29
Figura 15- Rectificador de media onda- Tensión en la carga.....	30
Figura 16- Rectificador de doble onda .....	30
Figura 17- Tensión de entrada y tensión en la carga en un rectificador de doble onda. Simulación realizada en el programa Multisim.....	31
Figura 18- Rectificador en puente de Graetz.....	32
Figura 19- Tensión en la carga. Rectificador en puente de Graetz.....	32
Figura 20- Regulador MC78L12AP .....	33
Figura 21- MC78L12AP. Conexionado recomendado por el fabricante.....	33
Figura 22- Circuito de control y potencial.....	33
Figura 23- Entorno Arduino.....	35
Figura 24- Pinout Arduino Uno.....	36
Figura 25- Pinout Arduino Mega .....	39
Figura 26- Comportamiento del sistema variando la parte proporcional de un controlador.....	42
Figura 27- Comportamiento del sistema variando la parte integral de un controlador .	43
Figura 28- Comportamiento del sistema variando la parte derivativa de un controlador .....	43
Figura 29- conexiones del bornero del motor.....	45
Figura 30- Salida del tacómetro a 2790rpm .....	45
Figura 31- Salida del tacómetro a 6644rpm .....	45
Figura 32- Circuito de control y potencia-Esquemático.....	47
Figura 33- Circuito de control y potencia- PCB.....	47
Figura 34- Señal PWM generada por arduino (Duty cycle 50%) .....	49
Figura 35- Señal PWM generada por arduino (Duty cycle 50%) y señal de salida del driver (cannel 2, azul).....	49
Figura 36- Segundo circuito de potencia diseñado.....	50
Figura 37- Señal PWM Duty Cycle: 0% .....	51
Figura 38- Señal PWM Duty Cycle: 15% .....	51
Figura 39- Señal PWM Duty Cycle: 50% .....	51
Figura 40- Señal PWM Duty Cycle: 100% .....	51
Figura 41- Circuito para la medición de frecuencia .....	52
Figura 42- Timer/counter1 Control Register A .....	53
Figura 43- Timer/counter1 Control Register B .....	54
Figura 44- Timer/counter5 Control Register A .....	56
Figura 45- Timer/counter5 Control Register B .....	57

Figura 46- registro TCCR4A.....	60
Figura 47- registro TCCR4A.....	61
Figura 48- Registro TIMSK4 .....	61

## ÍNDICE DE TABLAS

Tabla 1- Descripción de los pines de arduino uno.....	37
Tabla 2- Descripción de los pines de arduino mega.....	40
Tabla 3- Características del motor.....	44
Tabla 4- bornero del motor.....	44
Tabla 5- Prueba de funcionamiento de la PCB.....	48
Tabla 6- Funcionalidad de los bits COM1x1 cuando los bits WGM13:0 estan configurados para no generar una señal PWM.....	53
Tabla 7- Señales que pueden ser generadas y sus configuraciones.....	54
Tabla 8- Descripción de los bits de selección del reloj utilizado en el temporizador/contador.....	55
Tabla 9- Configuración final registro TCCR1A.....	55
Tabla 10- Configuración final del registro TCCR1B.....	55
Tabla 11- Funcionalidad de los bits COM5x1 y COM5x0 cuando los bits WGM3:0 estan configurados para generar una señal PWM en modo Phase correct PWM.....	57
Tabla 12- Señales que pueden ser generadas y sus configuraciones.....	58
Tabla 13- Descripción de los bits de selección del reloj utilizado en el temporizador/contador.....	58
Tabla 14- configuración final del registro TCCR5A.....	59
Tabla 15- configuración final del registro TCCR5B.....	59
Tabla 16- Funcionalidad de los bits COM5x1 y COM5x0 cuando los bits WGM3:0 están configurados en modo CTC.....	60
Tabla 17- Configuración del registro TCCR4A.....	62
Tabla 18- Configuración del registro TCCR4B.....	62
Tabla 19- Configuración del registro TIMSK4.....	62
Tabla 20- Ajuste PID. Método de la respuesta en frecuencia.....	65

## ÍNDICE DE GRÁFICAS

Gráfica 1 - Cantidad de chatarra electrónica que se produce en el mundo .....	18
Gráfica 2- volumen de facturación por ventas de bienes de consumo tecnológico en España entre 2014 y 2017.....	19
Gráfica 3 - Extracción de estaño en Bangka (Mongabay Latam, 2013) .....	20
Gráfica 4 - Relación entre el tacómetro (Hz) y la velocidad (rpm) .....	46
Gráfica 5- Medición de la velocidad del motor variando el Duty cycle (Lazo abierto). Arduino Uno.....	56
Gráfica 6- Arduino Mega. Relación duty cycle - velocidad.....	59



# 1. INTRODUCCION

El presente proyecto surge ante los problemas con los que se encuentra la división de electrodomésticos de la cooperativa KOOPERA a la hora de reparar los electrodomésticos que les llegan a su sede para su reparación y reutilización. Un parte importante de estos problemas viene dada por la dificultad de reparar la tarjeta de control de los electrodomésticos (en particular la de las lavadoras) ante las trabas que los fabricantes ponen a la reparación de las mismas (imprimación de resina, borrado de referencias de componentes, etc.).

De esta forma y en colaboración con la ONGD y asociación universitaria Ingeniería sin Fronteras País Vasco - Euskal Herriko Mugarik Gabeko Ingeniaritza (ISF-MGI) se plantea la posibilidad de desarrollar un sistema de control universal de lavadora de bajo costo bajo filosofía Open Source.

## 1.1 OBJETIVOS

El objetivo principal de este proyecto es el diseño, desarrollo e implementación de un sistema de control capaz de gobernar cualquier lavadora equipada con un motor universal permitiendo así su reparación a pesar de la obsolescencia programada en las mismas. Con ello se apoya la sostenibilidad medioambiental (permitiendo la reutilización de lavadoras recicladas) y se disminuye el consumismo ya que incita al cliente a comprar un equipo reparado y no uno nuevo.

A continuación, se enumeran los objetivos específicos del presente trabajo, dividiéndolos en tres grupos.

### 1.1.1 Objetivos sociales y de medio ambientales:

- Colaborar y fortalecer la actividad de KOOPERA como un agente social de transformación.
- Fortalecer la visión que tiene Ingeniería sin Fronteras (ISF) sobre el consumo sostenible, el reciclado y la electrónica ética.
- Visibilizar la necesidad de una electrónica más ética y sostenible.

### 1.1.2 Objetivos técnicos:

- Diseñar e implementar un control eficiente y eficaz para una lavadora universal.
- Seguir la filosofía Open Source.

### 1.1.3 Objetivos personales:

- Hacer uso de los conocimientos adquiridos durante el grado.
- Hacer frente a los hábitos consumistas impulsado el reciclado y eliminando la obsolescencia programada en las lavadoras.

## 1.2 JUSTIFICACIÓN

El proyecto surge como solución a los problemas generados en la empresa de innovación social y ambiental KOOPERA a la hora de reparar lavadoras. Esta empresa se pone en contacto con un profesor que forma parte de ISF-MGI para solucionar dicho problema.

Los aparatos electrónicos ayudan cada vez más en tareas cotidianas de la vida, en los puestos de trabajo y en el tiempo de ocio. Sin embargo, la producción y el consumo de dichos equipos conllevan efectos negativos sobre los países en vías de desarrollo, tanto en el medio ambiente como en la salud de las personas y en sus derechos laborales. La complejidad de los equipos hace que la variedad de metales que se emplean en la electrónica haya aumentado considerablemente en los últimos años, estos minerales provienen de países en vías de desarrollo. Además, dichos metales son altamente contaminantes y una vez terminada su vida útil muchos de ellos acaban siendo enviados a los países del tercer mundo. Allí los restos son almacenados en vertederos no controlados lo cual conlleva dañar la salud de los habitantes y producir contaminación del medio ambiente.

Ekorrepara es una cooperativa de reinserción social, sin ánimo de lucro, dedicada a la recuperación de RAEE (Residuos de Aparatos Eléctricos y Electrónicos). Miembro de la red social Koopera, cuenta con protocolos de reparación que dan garantía y seguridad a su trabajo.

La empresa KOOPERA se ha encontrado con el problema de que los fabricantes diseñan los electrodomésticos con una vida útil intencionalmente reducida y dificultan sus reparaciones.

En el caso de las lavadoras, la mayoría de las veces no pueden reutilizarlas ya que las placas de control vienen impregnadas con una resina que impide su reparación. Los precios de una lavadora nueva y una tarjeta de control nueva son similares por lo que los usuarios optan por comprar una nueva lavadora.

## 2. ANTECEDENTES

### 2.1 ESTADO DEL ARTE

Para la elaboración de este proyecto se ha hecho uso de los siguientes conocimientos:

- Electrónica de potencia
- Sistemas electrónicos digitales
- Motores universales de corriente continua
- Instrumentación electrónica
- Software Open-Source

A continuación, se hará una introducción de los temas indicados:

#### 2.1.1 Electrónica de potencia

La electrónica de potencia se encarga de transformar la electricidad con la finalidad de alimentar otros equipos, transportar la energía eléctrica u controlar el funcionamiento de máquinas eléctrica.

A continuación, se muestra el diagrama de bloques de un circuito de potencia:

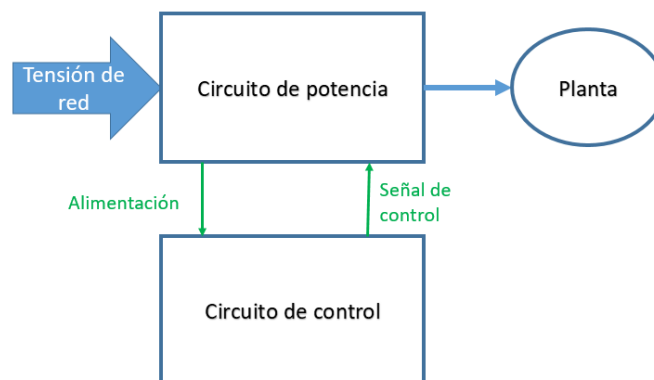


Figura 1 - Electrónica de potencia. Diagrama de bloques

Como se puede observar, un circuito de potencia siempre va ligado a un circuito de control. Este circuito se encargará de generar las señales de control que requiere el circuito de potencia para accionar la planta, en este caso, un motor.

La electrónica de potencia tiene multitud de aplicaciones, entre otros, en sistemas de control, en sistemas de compensación del factor de potencia y en la eliminación de armónicos.

#### 2.1.2 Sistemas digitales de control - sistemas embebidos

Los sistemas embebidos son aquellos cuyo objetivo no es computacional pero sí incluyen un computador que los controla. A continuación, se muestran los sistemas embebidos más usuales en la industria:

### Microprocesadores y microcontroladores:

Los microprocesadores son circuitos integrados digitales que pueden programarse por medio de instrucciones para realizar operaciones específicas con datos.

Los microcontroladores en cambio, son computadores miniaturizados que cuentan con recursos limitados y se integran en un chip para que resulten más baratos, portables y de menor consumo.

La principal diferencia entre ambos es que un microcontrolador es un microprocesador con periferia integrada. Como se puede observar en las siguientes figuras, un microcontrolador incorpora una memoria de programa, una de datos y puertos de entrada/salida, además de la unidad de control y la ALU (arithmetic logic unit). (SED, 2017)

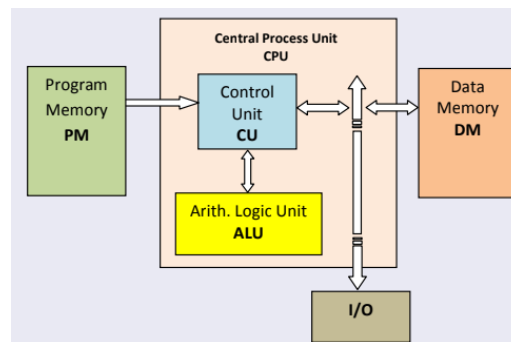


Figura 2 - Esquema de bloques de un uP- CISC

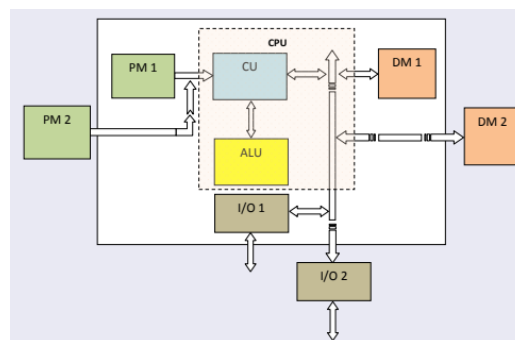


Figura 3 - Esquema de bloques de un uC-CISC

### 2.1.3 Motores universales de corriente continua

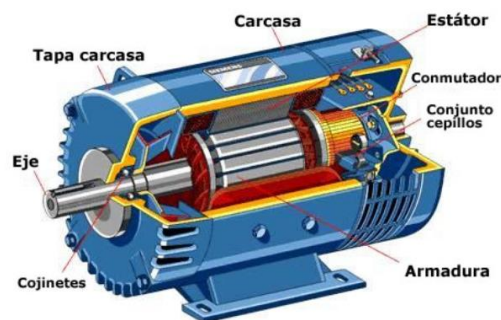
El motor universal es un motor capaz de funcionar tanto en corriente alterna como en continua. Cuando el motor se conecta en CC con una carga constante, la velocidad y la potencia aumentan de forma proporcional a la tensión aplicada. Por otro lado, si el motor universal se conecta en corriente alterna (CA) con una carga constante, la velocidad y la potencia aumentan proporcionalmente con la tensión aplicada a partir de 3000rpm. La velocidad dada para el mismo voltaje aplicado en CC y en CA es diferente, en CA la velocidad será menor.

Estos motores tienen un alto par de arranque, pueden funcionar a alta velocidad y son ligeros y compactos. Los inconvenientes de los motores universales son los siguientes:

- El colector consta de unas escobillas las cuales se desgastan con el uso del motor. Por ello, son mucho menos utilizados para equipos que se encuentren en uso continuo.
- Suelen ser muy ruidosos, tanto acústica como electromagnéticamente.

### Construcción de un motor universal

En cuanto a su construcción, es muy similar a la de una máquina de corriente continua en serie. Las partes principales son la carcasa, el estator y el rotor.



*Figura 4 - Estructura interna de un motor universal*

- Carcasa: Generalmente es de acero o de aluminio. Su función es mantener firme las piezas del estator y disipar el calor generado cuando el motor se encuentra en funcionamiento.
- Estator: Consiste en un paquete de láminas circulares prensadas, fijadas con remaches, que en su interior tienen unos polos salientes con forma necesaria para recibir, generalmente, un solo par de bobinas inductoras.
- Rotor o inducido: Consiste en un paquete de láminas de acero que forman el núcleo, con unas ranuras en las que se alojan varias bobinas, cuyos extremos van soldados a un colector.
- Colector: Es una pieza circular montada en el eje hecha con numerosas láminas de cobre, llamadas delgas. Éstas, están aisladas unas de otras con una mica intermedias. Sobre el colector, cuya superficie es completamente lisa, van unas portaescobillas con sus respectivas escobillas de carbón, que permiten la conexión eléctrica en serie con el rotor.

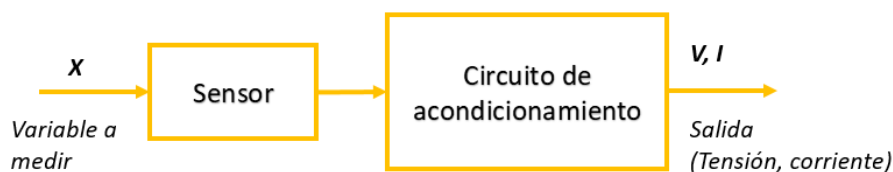
### Principio de funcionamiento

El motor universal basa su funcionamiento en la ley de Laplace: todo conductor por el que circula una corriente eléctrica, inmerso en un campo magnético, experimenta una fuerza que lo tiende a poner en movimiento. El bobinado inductor y el bobinado inducido están conectados en serie. Al ser recorridos por una corriente, el bobinado inductor forma el campo magnético y el inducido (por la ley de Laplace) se desplaza, dando origen al giro del rotor. Si aumenta el campo aumenta la fuerza y en consecuencia aumenta la velocidad.

#### **2.1.4 Instrumentación electrónica**

Los sistemas de instrumentación o sistemas de medida se encargan de extraer información del entorno mediante la identificación y cuantificación de sus variables características. Hoy en día los sistemas electrónicos instrumentales son imprescindibles y están presentes de forma masiva en las vidas cotidianas de los seres humanos. Por ejemplo, un teléfono móvil está repleto de instrumentación: el micrófono, el cual se encarga de recoger la voz, tratarla y trasmitirla a cualquier punto del mundo, la cámara, los sensores táctiles de la pantalla...

La instrumentación electrónica entrega la información mediante señales eléctricas a otros sistemas electrónicos, como puede ser un microcontrolador o un microprocesador (García, 2014).



*Figura 5 - Diagrama de bloques del acondicionamiento de una señal*

Las señales producidas a la salida pueden ser tratadas principalmente de dos formas:

- Sistemas que solo actúan como medidores. En este caso, la información de salida se representará en un visualizador, como puede ser un display.
- La señal de salida se lleva a circuitos de control que tomarán decisiones en función de los valores recibidos, como puede ser
- un microcontrolador.

### 2.1.5 Software open-Source

Open Source, también llamado código Abieto, es un tipo de software que permite al usuario final utilizar, modificar y realizar mejoras en el código fuente del programa. Seguidamente, se enumeran sus principales ventajas:

- Sin costos o muy bajos: su licencia de uso suele ser gratuita o significativamente inferior al que tienen productos equivalentes con licencias propietarias.
- Alta confiabilidad, escalabilidad y rendimiento: los productos mejoran y resuelven sus problemas continuamente gracias a toda la comunidad de usuarios.
- Utilización de estándares: Facilita su uso por parte de millones de programadores alrededor del mundo.
- Control: los procesos y la información de un producto de código abierto es de dominio público y por lo tanto puede ser auditado para asegurar que no existan problemas en los desarrollos.

## 3. MARCO TEÓRICO

### 3.1 Descripción general

Como se ha mencionado anteriormente, en este proyecto se va a diseñar el control del motor de una lavadora ya que es el elemento más complejo de dicho equipo.

En la siguiente imagen podemos visualizar las diferentes partes de las que consta el proyecto:

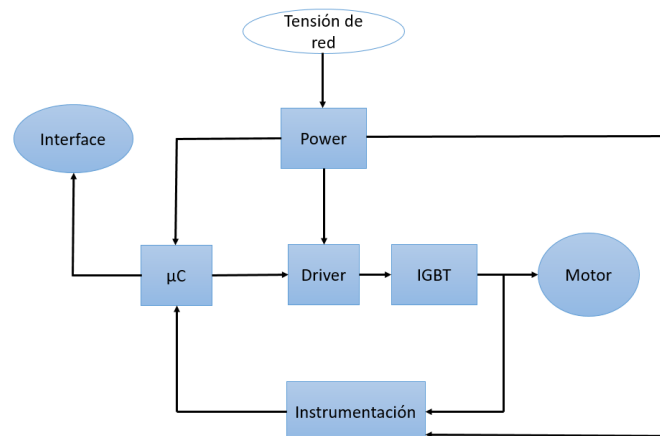


Figura 6 - Diagrama de bloques global

En primer lugar, se realizará un estudio teórico sobre la obsolescencia programa, puesto que es el motivo principal por el que se realiza este TFG. En este apartado se detallará el origen de la obsolescencia, tipos de obsolescencia, consecuencias y posibles soluciones. El segundo paso será definir el motor que se quiere controlar, es decir, se definirán sus características eléctricas y mecánicas.

Seguidamente, se hará un estudio sobre las distintas posibilidades que existen en cuanto al circuito de potencia, es decir, se definirá que tipo de transistores se van a utilizar y cuál será el driver que accionará dichos transistores.

Por último, se estudiarán los diferentes microcontroladores que tiene Atmel y se elegirá el más óptimo para éste caso.

### 3.2 Obsolescencia programada

La obsolescencia programa es la determinación del fin de la vida útil de un producto. Es decir, tras un periodo de tiempo calculado por el fabricante en la fase de diseño del producto, éste queda inservible. Por ejemplo, en el caso de una lavadora se dan casos donde tan solo se daña uno de los componentes pero no existen repuestos, por lo que el fabricante solo te da la opción de comprar una nueva. Las empresas utilizan la



obsolescencia programada con el objetivo de generar más ingresos debido a compras más frecuentes, con ello consiguen generar relaciones de adicción. Al contrario de las empresas excelentes, estas organizaciones no suministran productos de calidad ya que no tienen en cuenta las necesidades de los clientes, ni las repercusiones medioambientales. Esta práctica ha creado un creciente malestar entre los consumidores, por lo que hoy en día diferentes sectores están intentando revertir dicha práctica. (Eco medio ambiente, 2016)

### 3.2.1 Origen

El primer producto afectado por la obsolescencia programada fue la bombilla incandescente. Al principio, el objetivo era crear bombillas más durables. Sin embargo, todo cambió cuando se creó el Cártel Phoebus para establecer una duración máxima de 1000 horas de uso y penalizar a los fabricantes que violaran la nueva norma.

Se cree que el origen de la obsolescencia programada fue en 1932, cuando Bernard London propuso terminar con la Gran Depresión (crisis del 29) a través de la obsolescencia programada.

Un acontecimiento importante fue el *iPod* fabricado por Apple Computer ya que su batería fue diseñada para durar tan solo 18 meses ya que en ese intervalo de tiempo se iba a lanzar al mercado el próximo modelo.

El 8 de enero de 2018, la Fiscalía de Francia inició una investigación contra Apple por presunta obsolescencia programada de ciertos *iPhones* antiguos, ya que requiere actualizar periódicamente el sistema operativo. Esta actualización ralentiza el teléfono móvil notablemente.

En el lado opuesto, se encuentra la bombilla LED, la cual tiene una vida útil de 25 años aproximadamente. Sin embargo, sus ventas son bajas ya que su precio inicial es mayor que el de las bombillas fluorescentes.

### 3.2.2 Tipos de obsolescencia

- ***Obsolescencia programada:*** Como se ha comentado anteriormente, trata de reducir la vida útil de un producto. Es decir, cuando se haya utilizado un número de veces concreto el aparato quede inservible.
- ***Obsolescencia indirecta:*** Este tipo de obsolescencia se da en los casos que no es posible reparar un equipo, por falta de repuestos o por ser imposible su reparación.

- Obsolescencia funcional por defecto: Al fallar un componente todo el dispositivo deja de funcionar.
- Obsolescencia por incompatibilidad: Se da cuando un programa informático deja de funcionar al actualizarse el sistema operativo. En la mayoría de estos casos, el cliente opta por la comprar un equipo nuevo por el plazo y el precio de la reparación.
- Obsolescencia psicológica: Derivada de las campañas de mercadotecnia de las empresas encaminadas a hacer que los consumidores perciban como obsoletos los productos existentes.
- Obsolescencia estética: Cuando un producto es reemplazado por cuestiones de moda o diseño. Un claro ejemplo es la ropa, debido a la rapidez con la que cambian las modas. También puede extenderse a los electrónicos, como los smartphones.
- Obsolescencia por notificación: Un claro ejemplo son las impresoras, ya que convierten en obsoletos los cartuchos de tinta antes de que se agoten.
- Obsolescencia por caducidad: En muchos casos la industria alimentaria acorta la fecha de caducidad del producto, aunque todavía sea consumible sin riesgo para la salud.
- Obsolescencia ecológica: Bajo el argumento *verde* se justifica el abandono de los dispositivos antiguos, aún en perfecto estado, para promover la compra de nuevos equipos. El argumento en este caso es que dichos productos son menos dañinos para el medio ambiente. Este argumento carece de sentido ya que llevando a cabo esta práctica se aumenta considerablemente la generación de residuos siendo alguno de ellos difíciles de eliminar.
- Obsolescencia percibida: Es aquella que lleva al consumidor a sentir la necesidad de sustituir un bien por su modelo actual, aun no teniendo la necesidad de reemplazarlo.

### 3.2.3 Consecuencias

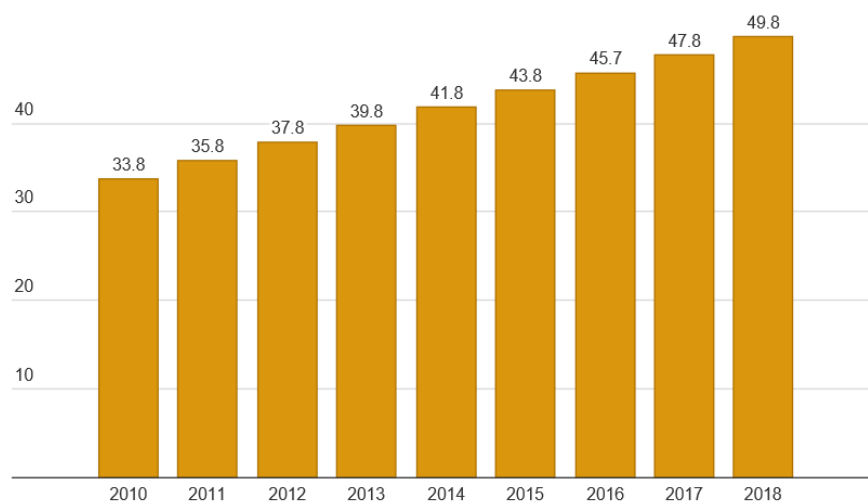
Como se ha mencionado anteriormente, la obsolescencia programada conlleva utilizar una gran cantidad de recursos naturales ya que se está constantemente produciendo. Una de las consecuencias más importantes, es el efecto negativo que tiene la producción sobre los países en vías de desarrollo tanto en el medio ambiente como en la salud de las personas y en sus derechos laborales. La mayoría de la basura generada en la producción acaba siendo vertida en estos países. Esto conlleva la contaminación del agua, combustibles fósiles, minerales etc. Además, el consumo de

energía contribuye a la aceleración del cambio climático, lo cual afecta a todo el planeta, pero con peores consecuencias a las poblaciones pobres de países del tercer mundo.

Por ejemplo, el consumo promedio de recursos para la fabricación de un ordenador es de 1500L de agua, 5300kWh de energía, 240kg de combustibles fósiles y 22kg de otros químicos.

La basura electrónica es el tipo de basura que crece más rápidamente en el mundo. En 2010 se generaron en el planeta 50 millones de toneladas y ese año se generó en Europa 8,7 millones de toneladas, de las cuales solo se recolectó y trató adecuadamente el 25%. (Ingeniería para el desarrollo humano).

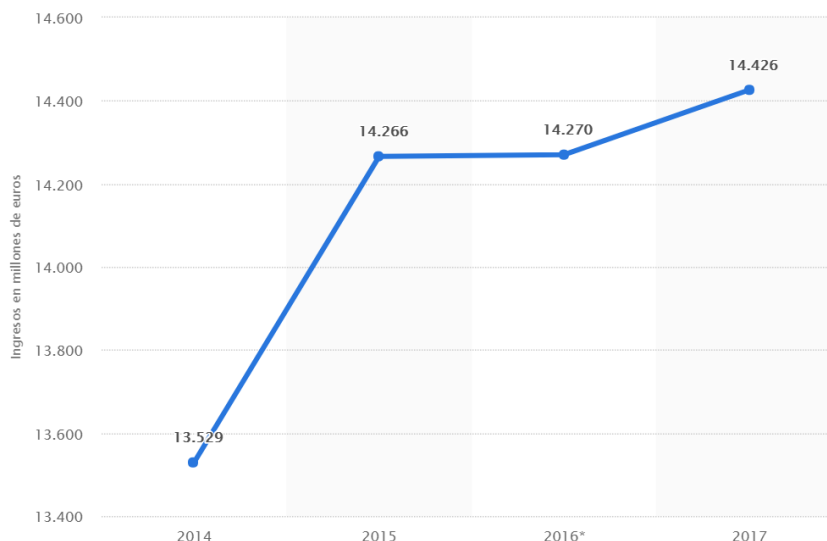
La siguiente gráfica muestra la cantidad de chatarra electrónica que se produce en el mundo en millones de toneladas. (Ottaviani, 2018)



Gráfica 1 - Cantidad de chatarra electrónica que se produce en el mundo

Otra consecuencia importante a tener en cuenta es el aumento del consumo en los últimos años. Como se ha mencionado anteriormente, los aparatos electrónicos se diseñan para tener una vida útil más reducida lo cual crea una tendencia a tener que comprar los aparatos antes. Además, el ingreso en este sector también ha aumentado, por lo tanto, es una estrategia de mercado interesante para las grandes empresas.

En la siguiente gráfica se puede observar el aumento del volumen de facturación por venta de electrodomésticos en España (statista, 2017)



Gráfica 2- volumen de facturación por ventas de bienes de consumo tecnológico en España entre 2014 y 2017

Para que la producción aumente con el consumo, los productos deben tener un precio competitivo, lo cual implica reducir el precio de los materiales y la mano de obra. Como consecuencia de esto, surgen conflictos geopolíticos, siendo uno de ellos el coltán. De este mineral se extrae el tantalio, componente indispensable para fabricar los condensadores electrolíticos de tantalio. Estos condensadores se utilizan en múltiples equipos, por ejemplo, los smartphones, las pantallas de plasma, consolas, reproductores, portátiles y tablets.

El problema que supone el tantalio es que no es un recurso renovable y el 80% de las reservas se encuentran en la República Democrática del Congo. Además del coltán, El Congo también tiene niobio, estaño, cobre, tungsteno, oro y cobalto, los denominados minerales de conflicto o minerales de sangre. Sin embargo, este no es el único país afectado como consecuencia de tener recursos naturales vitales para la tecnología. Metales como el cobre, indispensables en los circuitos electrónicos hoy en día, son extraídos principalmente en Chile y Perú donde no se tiene ningún control en cuanto a la contaminación en el proceso de extracción de los minerales y las condiciones laborales de los trabajadores no son óptimas. El estaño, usado para soldar componentes se puede encontrar en China, Malasia, Perú, Bolivia, Brasil y sobre todo, en Indonesia, donde existe un alto nivel de mortalidad y mano de obra infantil.



Gráfica 3 - Extracción de estaño en Bangka (Mongabay Latam, 2013)

### 3.2.4 Alternativas

Con el objetivo de poner fin a la obsolescencia programada, el Parlamento Europeo ha aprobado un texto que apoya el alargar la vida tanto de los smartphones como de otros dispositivos informáticos. Las medidas propuestas son las siguientes:

- Establecer un criterio de resistencia para cada tipo de producto que garantice su reparabilidad y su calidad.
- Incentivos fiscales para los productos duraderos, de calidad y fácilmente reparables.
- Alargar la garantía de los productos si una reparación conlleva más de un mes.
- Tener la opción de optar por un reparador independiente, es decir, que no dependa exclusivamente del fabricante.
- Componentes esenciales como las pantallas y baterías deben ser intercambiables.
- Los fabricantes deben ofrecer repuestos a un precio adecuado.
- Introducir pruebas contra la obsolescencia programada.

Por último, el Parlamento Europeo ha solicitado a Bruselas que estudie la posibilidad de crear una etiqueta europea voluntaria en la que se incluirían datos como la durabilidad del producto, el diseño ecológico, la posibilidad de reparación y la capacidad de modulación de conformidad. (La Ser, 2017)

Desde el punto de vista del consumidor también se puede apoyar la lucha contra la obsolescencia programada, por ejemplo, apoyando a entidades como Triodos Bank. En 2016 publicó un artículo donde indicaba cómo luchar contra la obsolescencia programada. *“Existen múltiples iniciativas para enfrentarse a una realidad que maximiza algunos*

*beneficios empresariales mientras merma recursos naturales y afecta a la calidad de vida de comunidades enteras. Escapar de los dictados de la obsolescencia es posible y se nos proponen varias vías diferentes para hacerlo.”* (Somos Triodos, 2016)

El primer paso es saber que es la obsolescencia programa *“una impresora deja de funcionar y, al intentar arreglarla, el usuario recibe siempre el mismo consejo, es mejor e incluso más barato el comprar una nueva que reparar el aparato averiado.”* (Dannoritzer, 2011)

El segundo paso es actuar. La ONG amigos de la Tierra ha elaborado un [directorio](#) en el que se recogen ciertos establecimientos que reparan y recuperan objetos, compran y venden de segunda mano (como el caso de Ekorrepara), alquilan o hacen truces. Desde el punto de vista del usuario, existen dos formas de apoyar la causa. La primera y la más obvia es priorizar el consumo en estas organizaciones y la segunda opción, menos visible, es apoyar bancos como Triodos Bank los cuales financian este tipo de proyectos. Por otro lado, la fundación Feniss (Fundación Energía para la Innovación Sostenible Sin Obsolescencia Programada) ha creado el sello ISSOP para empresas y organizaciones. Se trata de una certificación gratuita que distingue a aquellas entidades que priorizan la compra de productos y la contratación de servicios respetuosos con el medio ambiente, fabricados sin obsolescencia programada, y si es fabricante de algún producto, lo hacen sin obsolescencia programada, utilizando preferiblemente producto local y el comercio justo.

Otras organizaciones en cambio han propuesto soluciones innovadoras. Por ejemplo, La [Fundació Deixalles](#), implantada en Baleares es una de ellas. En su actividad se refleja un doble sentido social y medioambiental. “Recogemos aparatos electrónicos que funcionen o que necesiten una reparación, en cuyo caso la realizan personas en proceso de inserción laboral. A continuación ponemos en el mercado objetos de necesidad básica a un precio modesto”, explica Xesca Martí, la directora general de la entidad. En cuanto al cuidado del medio ambiente, se incluye tanto la reutilización como la correcta gestión de residuos. Los programas que dependen de Deixalles son además un claro ejemplo de economía solidaria. No solo revierten de forma positiva en la sociedad y en el planeta sino que además presentan un alto nivel de autofinanciación. “El 75% de nuestro presupuesto lo cubrimos gracias a lo que genera nuestra propia actividad. Solo el 25% restante proviene de subvenciones de las diferentes administraciones”, aclara Martí.

En el caso de Koopera Electro, se puede ver como cumple muchas de las soluciones propuestas por la ONG Amigos de la Tierra. Además de impulsar la reparación de productos viejos con garantía y calidad, también promueve la incorporación al mercado laboral de personas vulnerables.

### 3.3 Motor universal

Como ya se ha indicado en el apartado *Estado del Arte*, el motor universal es un motor capaz de funcionar en corriente continua y en corriente alterna. Es similar a un motor en serie de CC, aunque con algunas diferencias:

- Los núcleos polares y el circuito magnético están contruidos con chapas de hierro al silicio aisladas y apiladas con el fin de reducir las pérdidas de energía por las corrientes parásitas. Estas corrientes se producen con las variaciones de flujo magnético al conectar el motor a la red de corriente alterna.
- Estos motores tienen un menor número de espiras en el inductor con el fin de no saturar magnéticamente su núcleo y aumentar la intensidad. Con ello se consigue disminuir las pérdidas por corrientes de Foucault y por histéresis, aumentar el par y mejorar el factor de potencia (mejorar el rendimiento).
- Otra gran diferencia es el número de espiras en el inducido. En este caso, los motores universales tienen un mayor número de espiras en el inducido para así compensar la disminución del flujo debido a lo comentado en el anteriormente.

#### 3.3.1 Principio de funcionamiento en corriente continua

En corriente continua se comporta como un motor serie. Al invertir la corriente continua del motor en serie, el sentido de rotación permanece constante. Al aplicar corriente alterna a un motor en serie, el flujo de corriente en la armadura y en el campo se invierte simultáneamente, por lo que el motor seguirá girando en el mismo sentido.

#### 3.3.2 Principio de funcionamiento en corriente alterna

Cuando el motor universal se conecta en alterna, su flujo varía cada medio ciclo.

En la primera mitad de la onda de corriente alterna, la corriente en los devanados de la armadura fluye en sentido horario, es decir, de izquierda a derecha. Mientras que el flujo tiene un sentido antihorario.

En la segunda mitad de la onda de corriente alterna, denominada negativa, el voltaje aplicado invierte su polaridad, así mismo la corriente cambia su dirección y

ahora está de derecha a izquierda, también el flujo invierte su sentido. El par de arranque no cambia su dirección, puesto que en la mitad negativa se invierten tanto la dirección de la corriente, como la del flujo.

El motor universal consta de una menor potencia en corriente alterna que en continua, debido a que en alterna el par es pulsatorio. Además, la corriente está limitada por la impedancia, formada por el inductor y la resistencia del bobinado. Por lo tanto habrá una caída de tensión debido a reactancia cuando funcione con corriente alterna, lo que conlleva una disminución del par.

Por otro lado, también hay un mayor chispeo en las escobillas cuando funciona en corriente alterna, debido a que las bobinas del inducido están atravesadas por un flujo alterno cuando se ponen en cortocircuito por las escobillas, lo que obliga a poner un devanado compensador en los motores medianos para contrarrestar la fuerza electromotriz inducida por ese motivo.

### **3.3.3 Bobinado de compensación**

Los motores universales tienen la misma característica de velocidad y par cuando funcionan en C.A. o en C.C. Los motores universales pequeños, en general, no requieren devanados compensadores ya que el número de espiras de su armadura es reducido y por lo tanto, también lo será su reactancia. Por ello, los motores inferiores a 0,5CV se construyen sin compensación. El costo de los motores universales no compensados es relativamente bajo por lo que su aplicación es muy común en aparatos domésticos ligeros (aspiradoras, licuadoras...).

Por otro lado, los motores universales grandes suelen requerir algún tipo de compensación. Esta compensación se realiza especialmente para contrarrestar los problemas de la reacción de armadura.

La mayoría de las veces, la compensación implica incluir devanados compensadores distribuidos en los polos del motor. Si el motor va a ser utilizado tanto en corriente alterna como en corriente continua, el devanado compensador se conecta siempre en serie con la armadura y se dice que el motor está compensado conductivamente. Si el devanado compensador está conectado en corto circuito sobre sí mismo, se dice que el motor está compensado inductivamente.



### 3.4 Circuito de potencia

En este apartado se definirán los componentes que van a formar parte del circuito de potencia.

#### 3.4.1 Transistor IGBT

El transistor IGBT (Insulated Gate Bipolar Transistor o transistor bipolar de puerta aislada) es un dispositivo que se suele utilizar como interruptor controlado. Este transistor es un híbrido que combina las ventajas de los transistores BJT Y MOSFET. Por un lado, los transistores IGBT poseen las características de las señales de puerta de los MOSFET, es decir, se controlan por tensión y tiene una alta velocidad de conmutación, igual que los transistores de efecto de campo. Por otro lado, estos transistores tienen un bajo voltaje de saturación, como los transistores bipolares, es decir, las pérdidas en un IGBT son similares a las de un BJT.

El transistor bipolar de puerta aislada generalmente se utiliza en circuitos de potencia ya que puede llegar a conmutar grandes voltajes aplicando en su puerta unos 15V. Esto ofrece la ventaja de controlar sistemas de potencia aplicando una señal eléctrica pequeña en la puerta.

A continuación, se muestra las dos formas de representar el transistor IGBT:

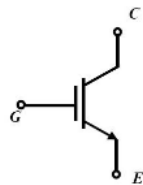


Figura 7 - Simbología de un IGBT como un BJT

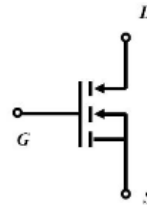


Figura 8 - Simbología de un IGBT como un MOSFET

Seguidamente, se muestran las curvas características de un IGBT y su circuito equivalente:

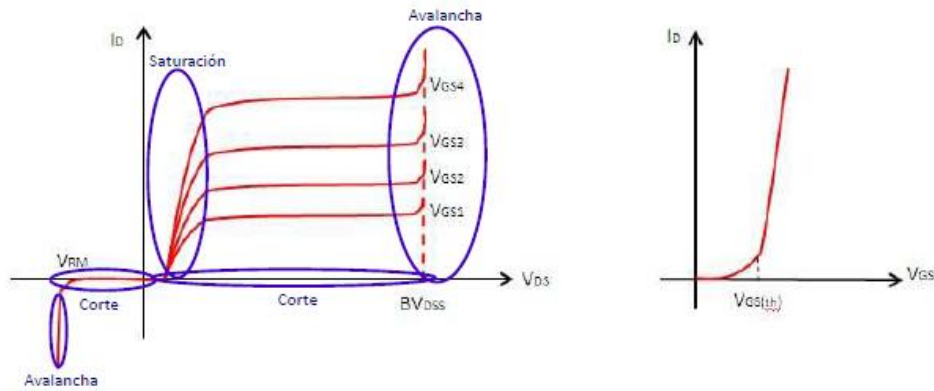


Figura 9 - Curvas características del IGBT

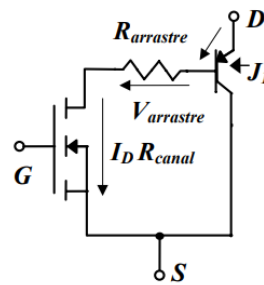


Figura 10 - Circuito equivalente simplificado de un IGBT

Si un voltaje  $V_{GS}$  es aplicado en la puerta (gate), el IGBT se enciende inmediatamente, la corriente  $I_D$  es conductiva y el voltaje  $V_{DS}$  se va desde el valor de bloqueo hasta 0. La intensidad  $I_D$  permanece mientras que se aplique la tensión  $V_{GS}$ . Se debe tener en cuenta que para encender el IGBT, el drenador debe ser polarizado positivamente con respecto al surtidor o fuente. Para apagar el IGBT, simplemente se debe aplicar una tensión menor a  $V_{GS}$  es la puerta. La transición del estado de conducción al estado de bloqueo puede tomar apenas 2 micro segundos, por lo que la frecuencia de conmutación puede estar en el rango de los 50 kHz (Industrial, 2015).

Como se ha mencionado anteriormente, la desventaja de este transistor es la disipación de potencia en el gate, puesto que se controla por tensión, la disipación de potencia es baja.

EL IGBT requiere un valor límite  $V_{GS}(Th)$  para el cambio de encendido a apagado y viceversa (usualmente de 4V). Por encima de este valor, el voltaje  $V_{DS}$  cae a unos 2V aproximadamente.

Si se está valorando utilizar este transistor, es importante tener las siguientes características en cuenta:

- $I_{Dmax}$  Limitada por efecto Latch-up.
- La  $V_{GSmax}$  está limitada por el espesor del óxido de silicio.

- Se diseña para que cuando  $V_{GS} = V_{GSmax}$  la corriente de cortocircuito sea entre 4 y 10 veces la nominal (zona activa con  $V_{DS}=V_{max}$ ) y pueda soportarla durante unos 5 a 10  $\mu s$ . De este modo, la protección electrónica puede actuar cortando desde la puerta.
- $V_{DSmax}$  es la tensión de ruptura del transistor PNP. Existen en el mercado IGBTs con valores de 600, 1.200, 1.700, 2.100V y 3.300V.
- La temperatura máxima de la unión suele ser de 150°C.
- Existen en el mercado IGBTs encapsulados que soportan hasta 400 o 600 A.
- La tensión  $V_{DS}$  apenas varía con la temperatura.
- Actualmente es usado para potencias que van desde varios kW y un par de MW, trabajando a frecuencias comprendidas entre los 5 kHz y 40kHz.

### 3.4.2 Driver MC33153P4

El driver utilizado para el IGBT es el MC33153. Este driver está especialmente diseñado para el control de transistores IGBT que se encuentran trabajando en alta tensión, por lo tanto, incluye aplicaciones como control de motores inductivos de corriente alterna y motor brushless de corriente continua. (On semiconductor)

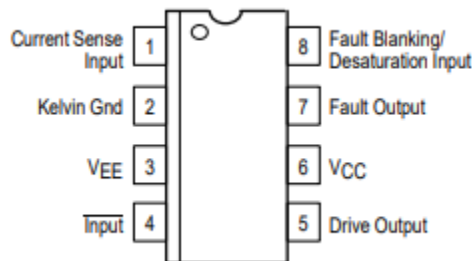


Figura 11- Pines del Driver MC33153

Siguiendo la nota de aplicación del fabricante, para el caso de un circuito básico, el conexionado es el siguiente:

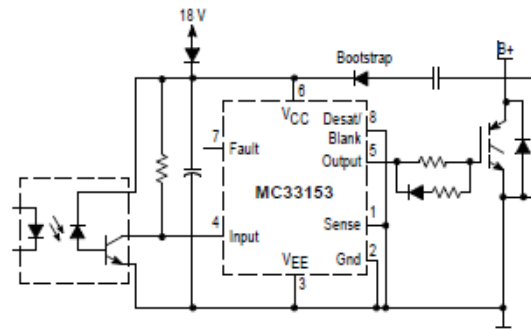


Figura 12- Conexión para una aplicación básica suministrada por Motorola

Datos importantes a tener en cuenta:

- En el caso de utilizar un optoacoplador, se requiere una resistencia pull up.
- Si no se requieren características de protección, los pines 1 (current sense input) y 8 (Fault blanking / desaturation input) deben conectarse al pin 2 (Kelvin ground).
- Si se va a utilizar una única fuente, el pin 2 (kelvin ground) y 3 (VEE) se deben conectar juntos.

### 3.4.3 Transformador

El transformador es un dispositivo eléctrico que permite aumentar o disminuir la tensión de corriente alterna manteniendo la potencia. En un transformador ideal, no habría pérdidas, es decir, la potencia del primario sería igual a la del secundario. No obstante, las máquinas reales presentan un porcentaje de pérdida. Esta pérdida depende, entre otros, de su diseño y tamaño.

Como bien se ha mencionado anteriormente, el transformador convierte el nivel de tensión. Esta transformación se lleva a cabo mediante el fenómeno de inducción electromagnética. Físicamente, los transformadores están constituidos por dos bobinas de material conductor, devanadas sobre un núcleo cerrado de material ferromagnético, pero aisladas entre sí eléctricamente. La única conexión entre las bobinas la constituye el flujo magnético común que se establece en el núcleo. El núcleo, generalmente, es fabricado bien sea de hierro o de láminas apiladas de acero eléctrico, aleación apropiada para optimizar el flujo magnético. Las bobinas o devanados se denominan primario (entrada del sistema) y secundario (salida del sistema). También existen transformadores con más devanados, por ejemplo, puede existir un devanado terciario de menor tensión que el secundario (Wikipedia, 2018).

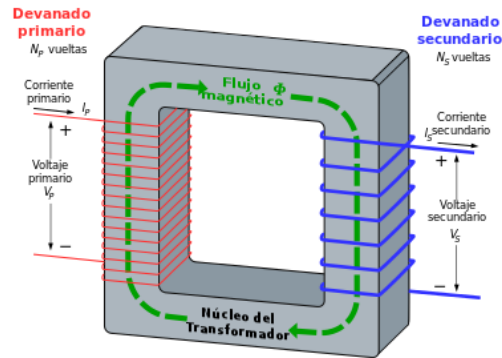


Figura 13- Transformador

### Transformador ideal

Se considera un transformador ideal aquel en el que no hay pérdidas de ningún tipo.

En un transformador ideal, debido a la inducción electromagnética, las tensiones en los devanados son proporcionales a la variación del flujo magnético que las atraviesa y al número de espiras del devanado:

$$\frac{V_p}{V_s} = \frac{N_p}{N_s}$$

Donde:

$V_p$ = Tensión en el primario

$V_s$ = Tensión en el secundario

$N_p$ = nº espiras en el primario

$N_s$ = nº espiras en el secundario

Se denomina relación de transformación a la relación de tensiones entre el primario y el secundario:

$$N = \frac{V_p}{V_s} = \frac{N_p}{N_s}$$

### Transformador real

En la práctica, no existe ningún transformador ideal. Los transformadores reales presentan los siguientes inconvenientes:

- Los devanados tienen resistencia eléctrica y capacidades parásitas.
- En el interior del núcleo hay corrientes de Foucault.

- El ciclo de magnetización y desmagnetización del núcleo, consume energía debido a la histéresis magnética.
- La permeabilidad magnética del núcleo depende de la frecuencia.
- La saturación magnética del núcleo provoca que la inductividad de los devanados no sea constante.

### 3.4.4 Rectificador

Los rectificadores son elementos o circuitos que convierten la corriente alterna en corriente continua. Esto se realiza mediante diodos rectificadores.

Si el rectificador está conectado a una fase de la red eléctrica, se le denomina rectificador monofásico, mientras que si se alimenta a tres fases, se denomina rectificador trifásico.

Estos elementos son ampliamente utilizados en la alimentación de máquinas de corriente continua de alta potencia y en electrodomésticos para la alimentación de los diferentes circuitos que lo componen.

Puesto que en este proyecto se van a utilizar dos rectificadores monofásicos, se va a centrar este apartado en ellos.

Existen tres tipos de rectificadores monofásicos: Rectificadores monofásicos de media onda, de doble onda y rectificadores monofásico en puente de Graetz.

#### Rectificadores monofásico de media onda

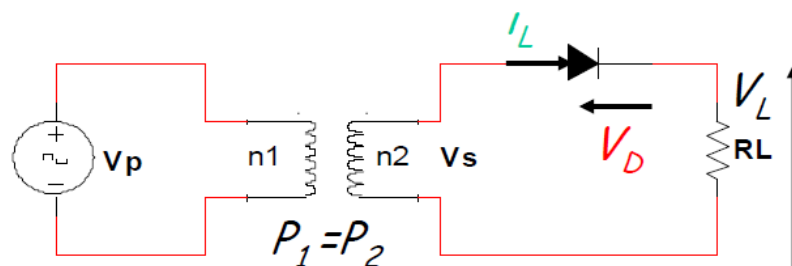


Figura 14- Rectificador de media onda

Como se puede observar, los rectificadores de media onda está compuestos por un solo diodo, el cual está situado, normalmente, entre el secundario de un transformador y la carga.

Considerando el diodo ideal, la tensión en la carga sería la siguiente:

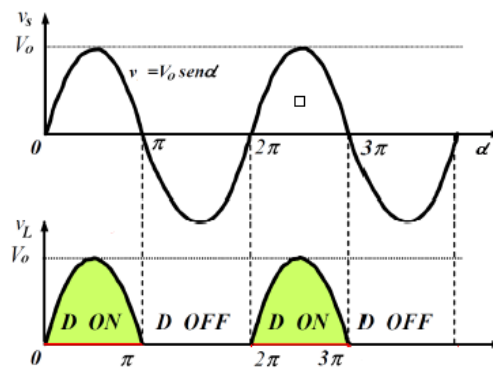


Figura 15- Rectificador de media onda- Tensión en la carga

Cuando la tensión de entrada es positiva, el diodo se encuentra polarizado directamente y se comporta como un interruptor cerrado. Si la tensión de entrada es negativa, el diodo se polariza inversamente y se comporta como un interruptor abierto. Por lo tanto, si el diodo está polarizado directamente (PD), la tensión en la carga se puede hallar por medio de la relación de un divisor de tensión. Cuando la alimentación se encuentra en el semiciclo negativo, es decir, el diodo está inversamente polarizado (PI), la corriente es cero, por consiguiente, la tensión de salida también es cero. Este rectificador no es muy eficiente debido a que durante la mitad de cada ciclo la entrada se bloquea completamente desde la salida, perdiendo así la mitad de la tensión de alimentación. El voltaje de salida en este tipo de rectificador es aproximadamente 0.45 veces el voltaje eficaz de la señal de entrada.

El rendimiento de este tipo de rectificador es muy bajo, debido fundamentalmente a que solo resulta útil el semiciclo positivo de la onda alterna del secundario. Este rendimiento aumentaría mucho si se pudieran utilizar los dos semiciclos, de forma que circule la corriente por la carga durante ambos (analógica, 2016).

### Rectificadores monofásico de doble onda

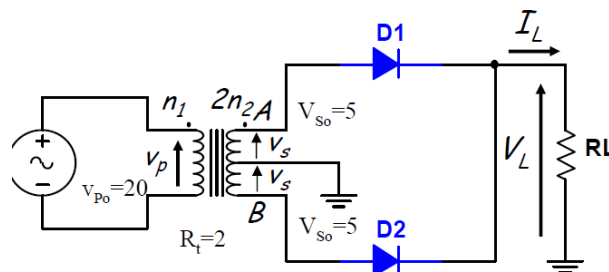


Figura 16- Rectificador de doble onda

Para aumentar el rendimiento de un rectificador monofásico, se emplean dos rectificadores de media onda desfasados  $180^\circ$ , alimentado a la misma carga. De esta forma

siempre circula corriente por la carga. En este caso, se obtendrán valores de corriente y tensiones medias dobles y valores eficaces  $\sqrt{2}$  mayores que en el caso de utilizar un rectificador de media onda.

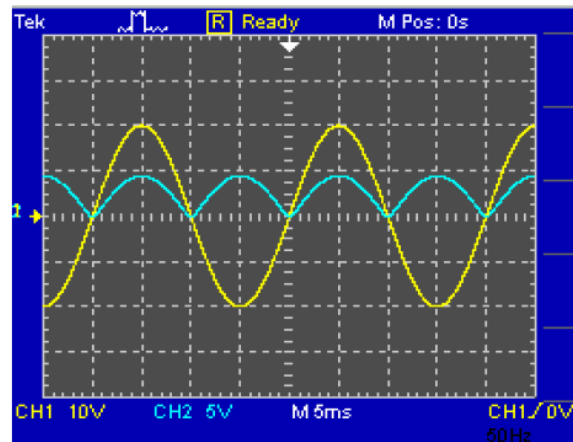


Figura 17- Tensión de entrada y tensión en la carga en un rectificador de doble onda. Simulación realizada en el programa Multisim

Seguidamente, se muestran las diferencias fundamentales con el rectificador de media onda:

- Doble rendimiento que el de media onda.
- Cuatro veces de potencia media en carga y el doble de potencia en la carga.
- Mejora el factor de forma y el rizado de la corriente en la carga.
- El primer armónico en la carga es de doble frecuencia respecto a la red por lo que será más fácil eliminar o disminuir los armónicos.
- En los rectificadores de doble onda, los diodos soportan una tensión inversa de pico (PIV) doble que el diodo empleado en el rectificador de media onda.

### Rectificadores en puente de Graetz



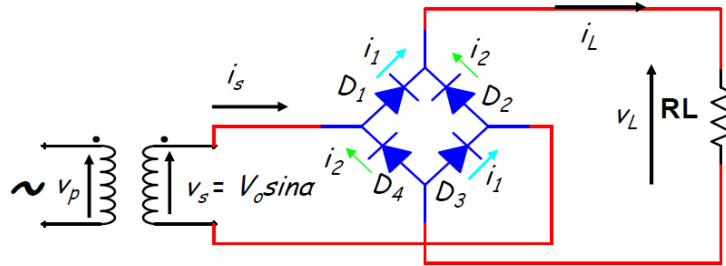


Figura 18- Rectificador en puente de Graetz

Si la tensión en el secundario del transformador es  $V_s = V_o \sin \alpha$ , durante el semiciclo positivo conducirán  $D_1$  y  $D_3$ , permaneciendo bloqueados  $D_2$  y  $D_4$ . En el semiciclo negativo en cambio, conducen los diodos  $D_2$  y  $D_4$ , quedando  $D_1$  y  $D_3$  bloqueados.

La gran diferencia entre este tipo de rectificador y el mencionado anteriormente, es la tensión inversa de pico que soportan los diodos. En este caso  $PIV=V_o$ .

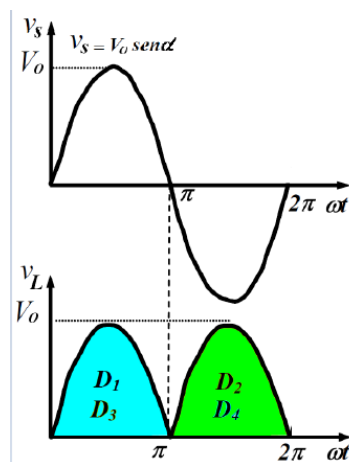


Figura 19- Tensión en la carga. Rectificador en puente de Graetz

### 3.4.5 Regulador de tensión MC78L12APG

Puesto que el driver que controla el IGBT y el microcontrolador se van a alimentar a 12V, se va a hacer uso de dicho regulador. Es un elemento muy simple, el cual solo consta de tres pines:

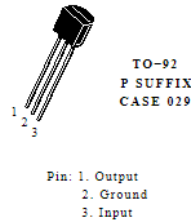


Figura 20- Regulador MC78L12AP

Siguiendo las indicaciones del fabricante, el conexionado que se requiere para una aplicación estándar es el siguiente:

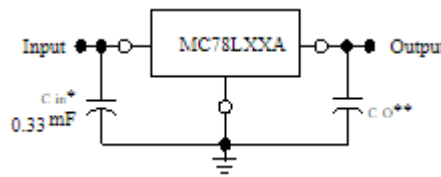


Figura 21- MC78L12AP. Conexionado recomendado por el fabricante

El fabricante indica que se requiere una tierra común entre el pin de entrada y el de salida. Por otro lado, indica que la tensión de entrada se tiene que mantener 2V por encima de la tensión de salida. En cuanto a los condensadores,  $C_{in}$  es necesario si el regulador se encuentra a una distancia considerable de la fuente de alimentación y  $C_o$  sirve para mejorar la respuesta en el transitorio.

### 3.4. 6 Descripción del circuito

A continuación, se muestra el circuito de potencia junto al controlador y al motor:

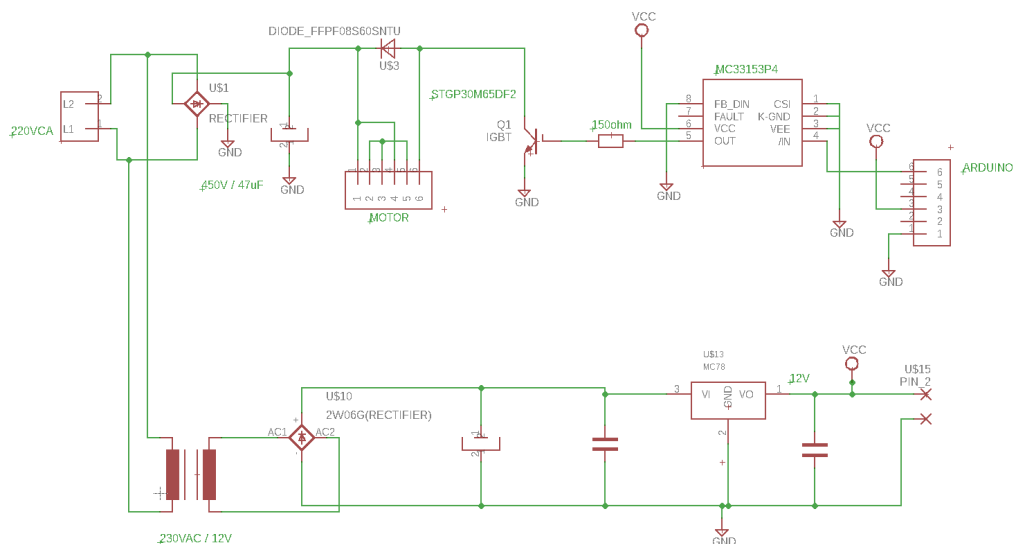


Figura 22- Circuito de control y potencial

Como se puede observar, todos los elementos han sido previamente explicados excepto dos: El diodo de recuperación rápida (FFPF08S60SNTU) y los filtros a la salida de los rectificadores.

En primer lugar, se puede apreciar que se ha tomado la decisión de utilizar dos rectificadores monofásicos en puente de Graetz, esto se debe a las ventajas señaladas en el apartado 3.4.4, por ejemplo, al ser el PIV que soportan los diodos mucho menor en comparación a los diodos de un rectificador de doble onda, los diodos sufrirán menos, lo cual conlleva una vida útil más larga.

En cuanto al filtro por condensador, es necesario para esta aplicación ya que a la salida del rectificador se obtiene una tensión unidireccional pero no continua, dicho de otra forma, se obtiene una tensión alterna (armónicos) superpuesta a otra continua. Haciendo uso de los condensadores se intentan eliminar los armónicos lo máximo posible pero nunca se va a conseguir eliminarlos por completo. Para el diseño del circuito, los condensadores han sido seleccionados para tener el menor rizado posible.

Por otro lado, se debe tener en cuenta que el motor es una carga inductiva. Como toda carga inductiva, al desconectarlo de la red intentará mantener la intensidad constante, lo que puede afectar al IGBT. Si la puerta del IGBT no se encuentra alimentada, pero si recibe corriente por el surtidor (colector), se puede llegar a destruir dicho componente. Por ello, se introduce un diodo entre los terminales 1 y 6 del motor, de forma que al desconectar el motor la corriente no se derivará nunca al IGBT.

### **3.5 Microcontrolador: Arduino. Control PID**

Arduino es una plataforma de open source y open hardware, basada en una placa con un microcontrolador y un entorno de desarrollo. La principal característica del software de programación y del lenguaje de programación de Arduino, es su sencillez y facilidad de uso.

Arduino se utiliza para desarrollar elementos autónomos, conectándose a dispositivos e interactuar tanto con el hardware como con el software. Sirve tanto para controlar un elemento, como por ejemplo, un motor que sube o baja una persiana basándose en la luz existente en una habitación, para ello se utilizaría un sensor de luz conectado al Arduino, o bien para leer la información de una fuente, como puede ser un teclado, y convertir la información en una acción como puede ser encender una luz.

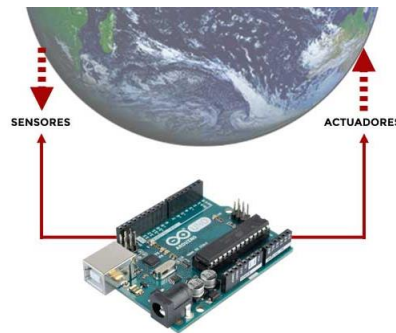


Figura 23- Entorno Arduino

Otro factor importante es la comunicación. Los usuarios pueden compartir conocimientos, elaborar librerías y publicar proyectos para que puedan ser utilizados, mejorados o ser base para otros trabajos.

En cuanto al Hardware (HW) de Arduino, es básicamente un microcontrolador. Como se ha mencionado en el apartado 2.1.2 un microcontrolador es un microprocesador con periferia integrada.

Las principales características de un microcontrolador son las siguientes:

- Velocidad del reloj u oscilador
- Tamaño de palabra
- Memoria: SRAM, Flash, EEPROM, ROM...
- I/O Digitales
- Entradas Analógicas
- Salidas analógicas (PWM)
- DAC (Convertidos digital analógico)
- ADC (Convertidos analógico digital)
- Buses
- UART
- Otras comunicaciones.

### 3.5.1 Arduino Uno

Arduino Uno es una placa electrónica basada en el microcontrolador ATmega328. Cuenta con 14 entradas/salidas digitales, de las cuales seis pueden ser utilizadas como salidas PWM (modulación por ancho de pulsos) y otras seis son entradas analógicas. Además, incluye un reloj de 16 MHz, un conector USB, un conector de alimentación, un conector ICSP (In circuit Serial Programming) y un botón de reseteo. El conector ICSP sirve para programar el BootLoader del microprocesador y así poder cargar los programas

directamente en el micro sin tener necesidad de programas externos. La placa incluye todo lo necesario para que el microcontrolador haga su trabajo, únicamente se debe conectar la placa un ordenador con un cable USB o a la corriente eléctrica a través de un transformador (Aprende Arduino, 2016).

### Características Técnicas

- ✓ Micro controlador: ATmega328
- ✓ Voltaje: 5V
- ✓ Voltaje de entrada recomendado: 7-12V
- ✓ Voltaje límite de entrada: 6-20V
- ✓ Digital I/O Pins: 14, de los cuales 6 son salida PWM
- ✓ Entradas Analógicas: 6
- ✓ DC Current per I/O Pin: 40 mA
- ✓ DC Current parar 3.3V Pin: 50 mA
- ✓ Memoria Flash: 32 KB de los cuales 0.5 KB son utilizados para el arranque.
- ✓ SRAM: 2 KB
- ✓ EEPROM: 1 KB
- ✓ Velocidad de reloj: 16 MHz

### Pinout

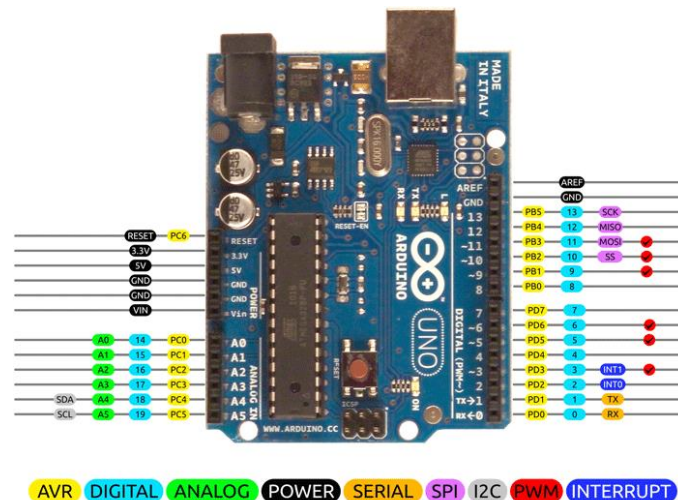


Figura 24- Pinout Arduino Uno

Tabla 1- Descripción de los pines de arduino uno

Nombre del pin	Descripción
$V_{in}$	tensión de alimentación de arduino cuando se va a utilizar una fuente de alimentación externa
3.3V	Fuente de alimentación de 3.3V. Corriente máxima de 50mA.
5V	Alimentación regulada utilizada para alimentar el microcontrolador y otros componentes de la placa. Esta puede venir de $V_{IN}$ a través de un regulador en placa o ser proporcionada por USB u otra fuente regulada de 5V.
Reset	Sirve para resetear el microcontrolador
A0 - A5	6 entradas analógicas que varían entre 0V y 5V
Digital pins 0 - 13	Pines digitales que pueden configurarse como entradas o salidas
0(Rx), 1(Tx)	Estos pines no se pueden utilizar como e/s digitales si se están utilizando comunicación serie (por ejemplo Serial.begin).
2, 3	Interrupciones externas
3, 5 6, 9, 11	Salidas PWM de 8 bits
13	único pin que tiene un led conectado en serie a la entrada, si se quiere testear algo mediante el brillo de un led, se puede utilizar este pin
10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	Comunicación SPI
A4 (SDA), A5 (SCA)	Comunicación TWI
AREF	Mantiene una tensión de referencia. Es útil para realizar conversiones analógicas a digitales.

### 3.5.2 Arduino Mega

Arduino Mega es una tarjeta de desarrollo open-source construida con un microcontrolador modelo Atmega2560 que posee pines de entradas y salidas (E/S), analógicas y digitales. Esta tarjeta es programada en un entorno de desarrollo que implementa el lenguaje Processing/Wiring.

El Arduino Mega tiene 54 pines de entradas/salidas digitales (14 de las cuales pueden ser utilizadas como salidas PWM), 16 entradas análogas, 4 UARTs (puertos serial por hardware), cristal oscilador de 16MHz, conexión USB, jack de alimentación, conector ICSP y botón de reset. Arduino. Arduino Mega es compatible con la mayoría de los shields diseñados para Arduino Due o UNO.

Esta nueva versión de Arduino Mega 2560 utiliza un microcontrolador ATmega8U2 en vez del circuito integrado FTDI. Esto permite mayores velocidades de transmisión por su puerto USB y no requiere drivers para Linux o MAC. Además ahora cuenta con la

capacidad de ser reconocido por el PC como un teclado, mouse, joystick, etc (Protagoniatec, s.f.).

### Características técnicas

- ✓ Microcontrolador: ATmega2560
- ✓ Voltaje Operativo: 5V
- ✓ Voltaje de Entrada: 7-12V
- ✓ Voltaje de Entrada (límites): 6-20V
- ✓ Pines digitales de Entrada/Salida: 54 (de los cuales 15 proveen salida PWM)
- ✓ Pines análogos de entrada: 16
- ✓ Corriente DC por cada Pin Entrada/Salida: 40 mA
- ✓ Corriente DC entregada en el Pin 3.3V: 50 mA
- ✓ Memoria Flash: 256 KB (8KB usados por el bootloader)
- ✓ SRAM: 8KB
- ✓ EEPROM: 4KB
- ✓ Velocidad de reloj: 16 MHz

### Pinout

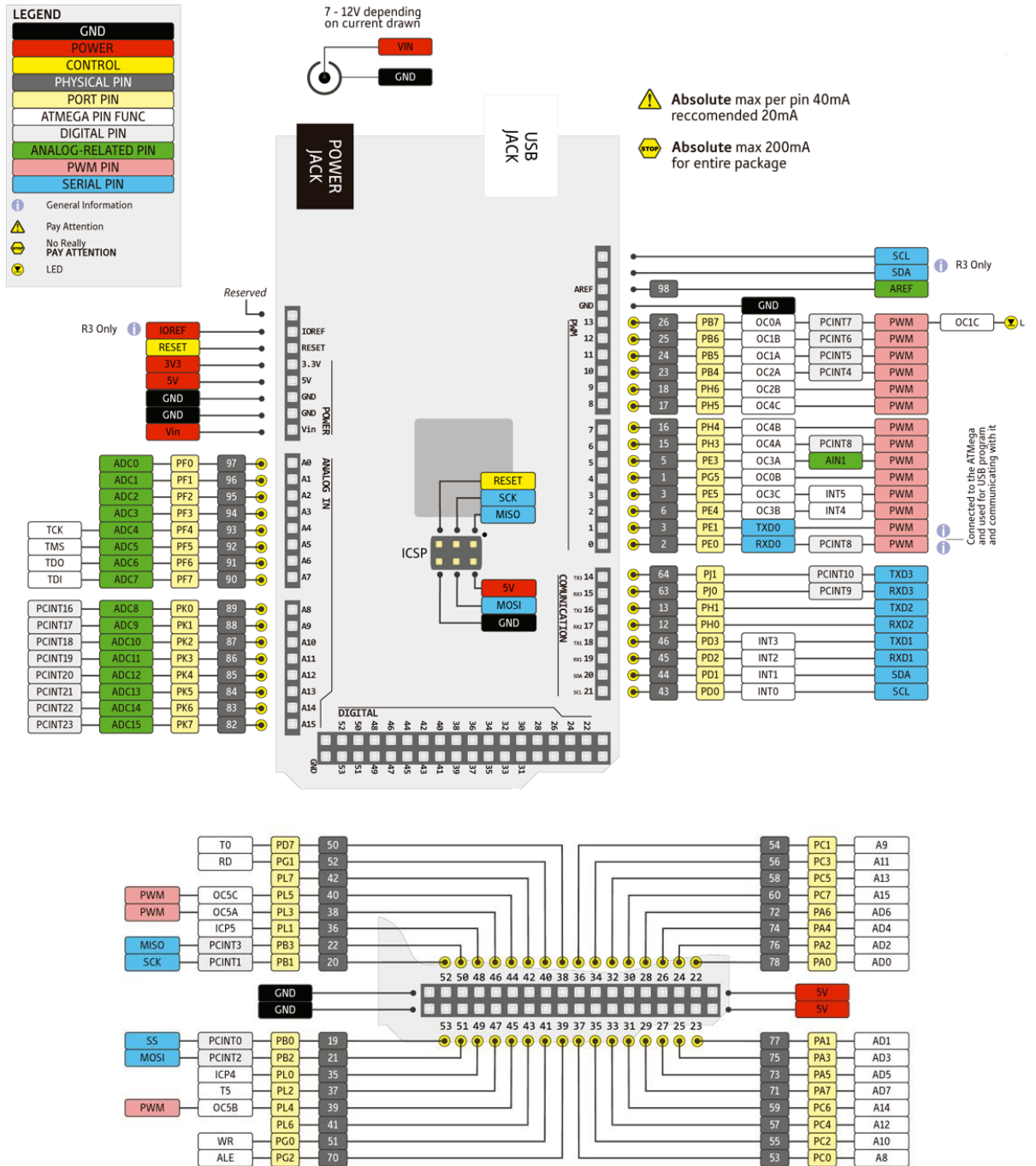


Figura 25- Pinout Arduino Mega



Tabla 2- Descripción de los pines de arduino mega

Nombre del pin	Descripción
$V_{in}$	tensión de alimentación de arduino cuando se va a utilizar una fuente de alimentación externa
3.3V	Fuente de alimentación de 3.3V. Corriente máxima de 50mA.
5V	Fuente de alimentación de 5V
Reset	Sirve para resetear el microcontrolador
0(Rx), 1(Tx)	Serie 0. Usados para recibir (RX) y transmitir (TX) datos a través de puerto serie TTL. Estos pines están conectados a los pines correspondientes del chip FTDI USB-to-TTL.
19 (Rx), 18(Tx)	Serie 1. Usados para recibir (RX) transmitir (TX) datos a través de puerto serie TTL
17 (Rx) , 16 (Tx)	Serie 2. Usados para recibir (RX) transmitir (TX) datos a través de puerto serie TTL
15 (RX) y 14 (TX)	Serie 3. Usados para recibir (RX) transmitir (TX) datos a través de puerto serie TTL
2	Interrupción externa (interrupción 0). Este pin se puede configurar para lanzar una interrupción en un valor LOW(0V), en flancos de subida o bajada (cambio de LOW a HIGH (5V) o viceversa), o en cambios de valor. Ver la función attachInterrupt() para más detalles.
3	Interrupción externa (interrupción 1). Este pin se puede configurar para lanzar una interrupción en un valor LOW (0V), en flancos de subida o bajada (cambio de LOW a HIGH (5V) o viceversa), o en cambios de valor. Ver la función attachInterrupt() para más detalles.
18	Interrupción externa (interrupción 5). Este pin se puede configurar para lanzar una interrupción en un valor LOW (0V), en flancos de subida o bajada (cambio de LOW a HIGH (5V) o viceversa), o en cambios de valor. Ver la función attachInterrupt() para más detalles.
19	Interrupción externa (interrupción 4). Este pin se puede configurar para lanzar una interrupción en un valor LOW (0V), en flancos de subida o bajada (cambio de LOW a HIGH (5V) o viceversa), o en cambios de valor. Ver la función attachInterrupt() para más detalles.
20	Interrupción externa (interrupción 3). Este pin se puede configurar para lanzar una interrupción en un valor LOW (0V), en flancos de subida o bajada (cambio de LOW a HIGH (5V) o viceversa), o en cambios de valor. Ver la función attachInterrupt() para más detalles.
21	Interrupción externa (interrupción 2). Este pin se puede configurar para lanzar una interrupción en un valor LOW (0V), en flancos de subida o bajada (cambio de LOW a HIGH (5V) o viceversa), o en cambios de valor. Ver la función attachInterrupt() para más detalles.

0 al 13	Salidas PWM de 8 bits de resolución ( valores de 0 a 255)
13	único pin que tiene un led conectado en serie a la entrada, si se quiere testear algo mediante el brillo de un led, se puede utilizar este pin
SPI: 50 (SS), 51 (MOSI), 52 (MISO), 53 (SCK)	Proporcionan comunicación SPI, usando librerías SPI
A0 - A15	16 entradas analógicas con una resolución de 10 bits (1024 valores). Por defecto se mide desde 0V a 5V, aunque es posible cambiar la cota superior de este rango usando el pin AREF y la función analogReference().
20 (SDA) y 21 (SCL)	Soporte para el protocolo de comunicaciones I2C (TWI) usando la librería Wire.
AREF	Voltaje de referencia para las entradas analógicas

### 3.5.3 Control PID

Un PID es un mecanismo de control por realimentación que calcula la desviación o error entre un valor medido y el valor que se quiere obtener, para aplicar una acción correctora que ajuste el proceso. El algoritmo de cálculo del control PID se da en tres parámetros distintos: el proporcional, en integral y el derivativo. El valor proporcional determina la reacción del error actual. El integral genera una corrección proporcional a la integral del error, esto asegura que aplicando un esfuerzo de control suficiente, el error de seguimiento se reduce a cero. El derivativo determina la reacción del tiempo en el que el error se produce (Asignatura de Regulación y control, 2016).

#### Acción del control proporcional (controlador P)

Se busca reducir el error en el estado estacionario. Puesto que este control no considera el tiempo, normalmente produce sobreoscilación, es decir, no se le puede dar cualquier valor a la constante proporcional, solo un rango determinado.

$$P(s) = K_p$$

#### Acción de control proporcional- integral (controlador PI)

La parte integral busca disminuir el error en el estado estacionario, actuando cuando hay una desviación entre la variable y la consigna. Se integra esta desviación en el tiempo y se suma a la acción proporcional.

$$PI(s) = K_p \left( 1 + \frac{1}{T_i \cdot s} \right)$$

#### Acción de control proporcional- derivativa (PD)

La derivada actúa cuando hay un cambio en el valor absoluto del error (si el error es constante, solo actuaría la parte integral). Busca mantener el error al mínimo corriéndolo proporcionalmente con la misma velocidad con que se produce.

$$PD(s) = K_p(1 + T_d \cdot s)$$

#### Acción de control proporcional-integral- derivativa (PID)

El objetivo de ajustar los tres parámetros es lograr que el bucle de control corrija eficazmente y en el mínimo tiempo posible los efectos de las perturbaciones. El principal problema del PID es que es lineal, por lo tanto, en sistemas no lineales su efecto es variable. Además, en la parte derivativa, el ruido puede afectar al sistema, provocando que pequeñas variaciones se conviertan en cambio muy grandes a la salida.

$$PID(s) = K_p \left( 1 + \frac{1}{T_i \cdot s} + T_d \cdot s \right)$$

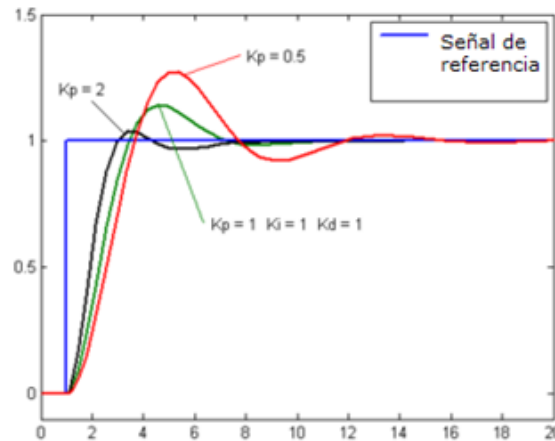


Figura 26- Comportamiento del sistema variando la parte proporcional de un controlador

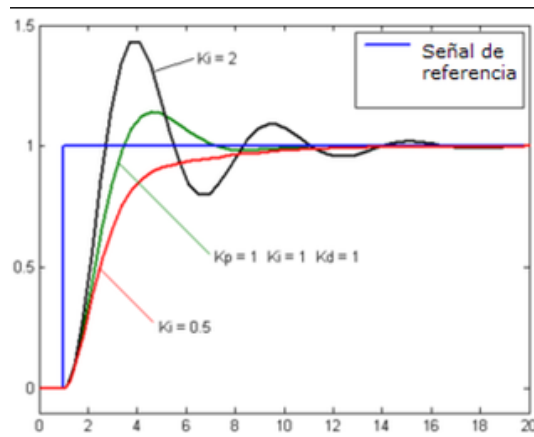


Figura 27- Comportamiento del sistema variando la parte integral de un controlador

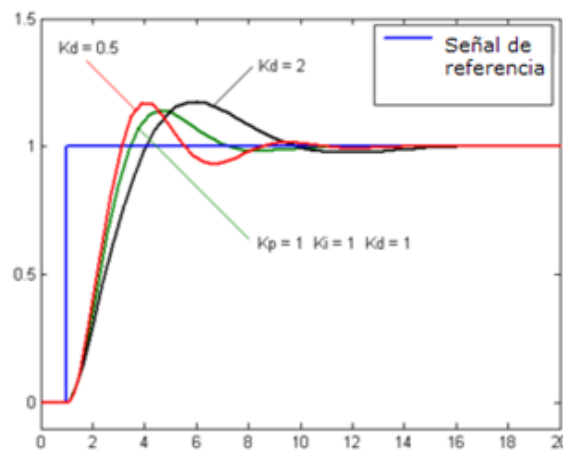


Figura 28- Comportamiento del sistema variando la parte derivativa de un controlador

## 4. DISEÑO

### 4.1 Motor

Como se ha mencionado en el apartado de antecedentes, este proyecto se ha basado en dos proyectos del curso 2016/17. Puesto que el motor no tiene placa de características y en KOOPERA no realizan ningún análisis sobre el motor, Álvaro Núñez recopiló información y realizó las pruebas necesarias para obtener las características básicas del equipo.

Los datos del cual parte este proyecto son los siguientes:

*Tabla 3- Características del motor*

Características	
<b>Modelo</b>	1BA6755-0LC
<b>Máxima velocidad</b>	12600 rpm
<b>Corriente nominal</b>	2,3 A
<b>Potencia</b>	460 W
<b>Eficiencia</b>	64%
<b>Tensión de funcionamiento</b>	220V a 50 Hz
<b>Peso</b>	6,3 Kg

#### 4.1.1 Bornes del motor

Para identificar como está constituido el motor, se han medido las resistencias entre los siete bornes de los que consta este motor. Los resultados son los siguientes:

*Tabla 4- bornero del motor*

Terminales	Resistencia medida( $\Omega$ )	conclusión
1 y 2	0,3	Estator principal
2 y 3	1,2	y compensado
1 y 3	1,1	-
4 y 5	5,9	Rotor
1 y 5	6,6	-
2 y 5	6,8	-
6 y 7	24	Tacómetro

Finalmente, se llega a la conclusión de que las conexiones son las siguientes:

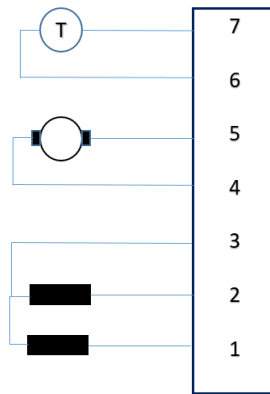


Figura 29- conexiones del bornero del motor

#### 4.1.2 Tacómetro

Como se ha mencionado en el apartado anterior, el motor lleva incorporado un tacómetro para poder leer la velocidad de giro del motor. Puesto que no se tienen datos del equipo, se plantea la duda de si varía la amplitud o la frecuencia cuando varía la velocidad del motor. Para ello, en proyectos anteriores, se ha visualizado la señal proporcionada por el tacómetro para dos velocidades diferentes:

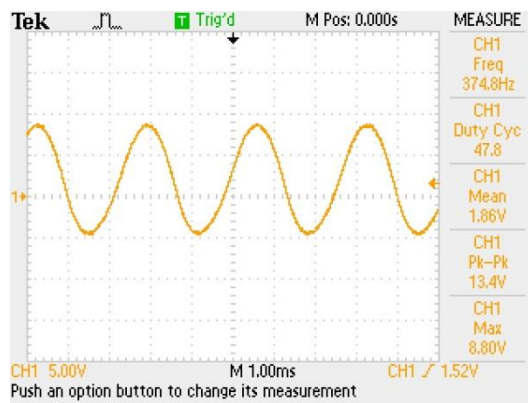


Figura 30- Salida del tacómetro a 2790rpm

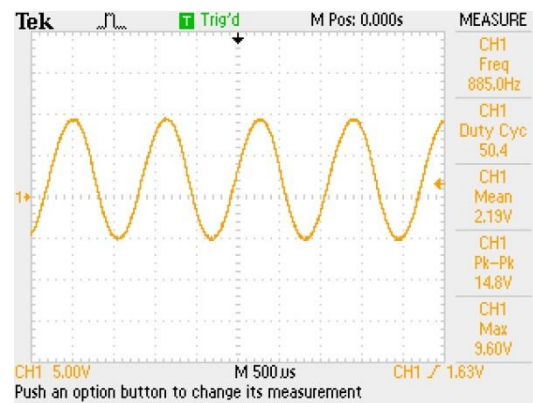


Figura 31- Salida del tacómetro a 6644rpm

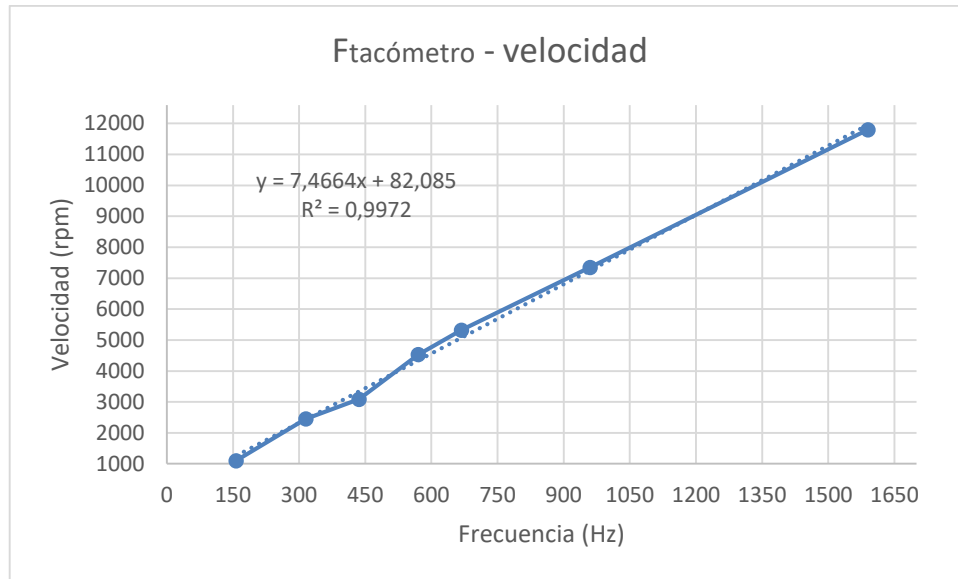
Como se puede observar, al variar la velocidad del motor varía la frecuencia y no la amplitud.

El siguiente paso ha sido hallar la relación frecuencia – velocidad del tacómetro. Hay dos métodos, el primero es desmontar el motor y contar en número de dientes y el segundo es hallar dicha relación midiendo la frecuencia (Hz) a velocidades diferentes (rpm).

En este caso, se ha optado por la segunda opción.

A continuación, se variará el duty cycle de la señal PWM y se medirá la frecuencia y la tensión eficaz de la señal proporcionada por el tacómetro con un osciloscopio.

En la siguiente gráfica se muestran los datos obtenidos:



Gráfica 4 - Relación entre el tacómetro (Hz) y la velocidad (rpm)

Como se puede observar, hay una relación directa entre la frecuencia de la señal del tacómetro y la velocidad de salida del motor:

$$Velocidad = 7,4664 \cdot F_{tacómetro} + 82,085$$

Esta ecuación se utilizará en el programa para el lazo cerrado.

## 4.2 Circuito de potencia

Una vez diseñada la PCB, se procede a verificar su correcto funcionamiento. Para ello, se medirán con el voltímetro los siguientes puntos de la placa:

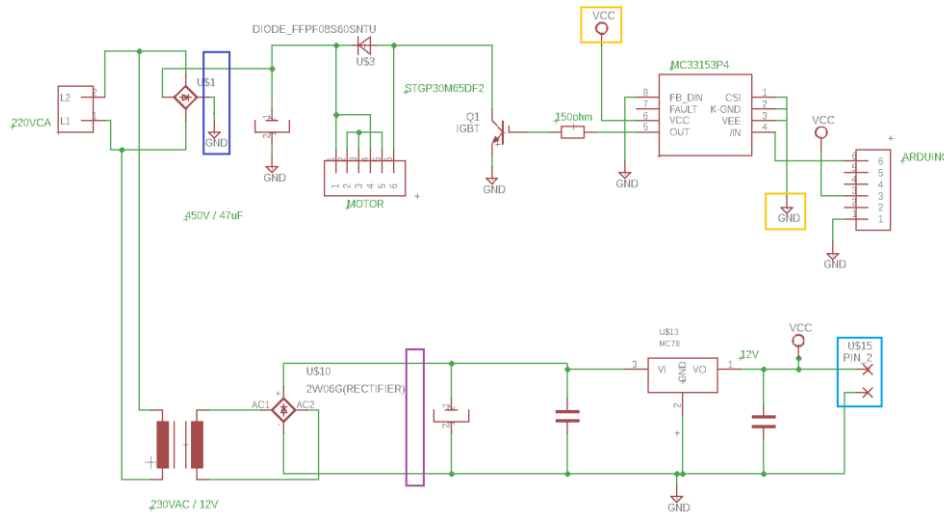


Figura 32- Circuito de control y potencia-Esquemático



Figura 33- Circuito de control y potencia- PCB



Tabla 5- Prueba de funcionamiento de la PCB

	<i>Teoría</i>	<i>Práctica</i>
<i>Fuente para alimentar arduino y el driver</i>	12V	11,79V
<i>Tensión de alimentación del driver</i>	12V	11,79V
<i>Continuidad entre masas</i>	Sí	
<i>Tensión de salida del rectificador U\$1</i>	$220 \cdot \sqrt{2} = 311,12V$	$233V_{\text{rms}}$
<i>Tensión de salida del rectificador U\$10</i>	$12 \cdot \sqrt{2} = 16,97V$	$22V_{\text{rms}}$

Antes de probar la placa con el motor, se va a verificar su funcionamiento con una carga resistiva (una bombilla). Para ello, se ha realizado un pequeño programa donde se generará una señal PWM de frecuencia variable con arduino uno. De esta forma, se variará la luminosidad de la bombilla.

Se debe tener en cuenta que con la instrucción `analog.write()` solo se puede variar el duty cycle de la señal PWM pero no su frecuencia. Por consiguiente, se va a utilizar uno de los timers de los que consta este microprocesador para generar dicha señal de control.

El microprocesador de arduino uno consta de tres timers, dos de ellos de 8 bits y el restante de 16 bits.

En primer lugar, se ha cargado el programa 7.1 en arduino y se visualizado la PWM generada con un osciloscopio. Una vez que se ha comprobado que el programa funciona correctamente, se ha conectado arduino a la PCB.

En este punto del proyecto se ha identificado un fallo, el circuito no estaba bien diseñado. Puesto que se había comprobado la alimentación y se había visualizado la PWM generada por arduino, el siguiente paso ha sido medir la señal de salida del driver. Midiendo en el pin 5 del driver (output) se ha verificado que éste no estaba funcionando, es decir, se había fundido.

Para descartar un mal uso del circuito, se ha procedido a cambiar el driver por uno nuevo y se han seguido los siguientes pasos para ir acotando los posibles fallos:

- 1- Medir la señal de salida (pin 9) de arduino

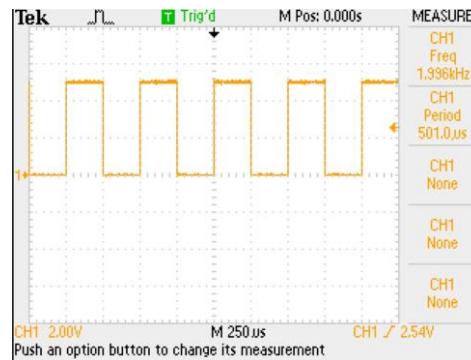


Figura 34- Señal PWM generada por arduino (Duty cycle 50%)

- 2- Alimentar arduino y el driver con una fuente de 12V y medir la señal de salida del driver (CH2).

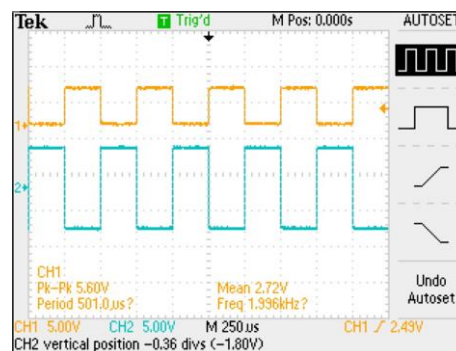


Figura 35- Señal PWM generada por arduino (Duty cycle 50%) y señal de salida del driver (cannel 2, azul)

- 3- Conectar la PCB a la red, quitando así la fuente de alimentación de 12V y poner la carga (bombilla). En este punto, se ha comprobado que el driver se ha vuelto a fundir.

Observando la figura 22, se puede ver como el problema está en las masas flotantes. El rectificador u\$1, en uno de sus semiciclos, une la referencia que tenga la red y la masa, por lo tanto la fuente de 12V diseñada no se sabe a qué voltaje está referenciada.

Viendo los problemas causados con este diseño, el siguiente paso ha sido diseñar otro circuito de control más sencillo.

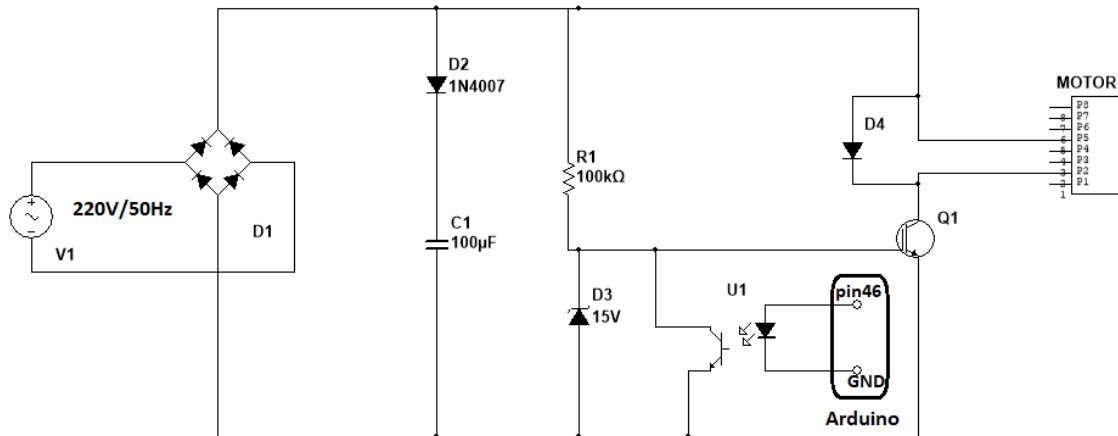


Figura 36- Segundo circuito de potencia diseñado

Este diseño es más sencillo ya que para saturar el IGBT simplemente se ha utilizado un zener de 15V y se ha eliminado el driver.

Al igual que en el diseño anterior, lo primero es rectificar la señal de red (rectificador D1). A continuación, se satura el transistor IGBT con el zener D3 y para hacer que el IGBT trabaje en corte, se utiliza el optoacoplador. Cuando arduino de la señal, el optoacoplador se acciona y el transistor entra en corte. Una ventaja importante de este diseño es el aislamiento entre la potencia y el controlador. Con este diseño, se puede tener conectado arduino al ordenador mientras la PCB está conectada a la red ya que el optoacoplador protege al microcontrolador.

Posteriormente, se ha comprobado su funcionamiento. En primer lugar, se ha vuelto a cargar el programa de arduino anteriormente mencionado y se han realizado las siguientes mediciones. En el programa de arduino se genera una señal PWM de 2kHz, por lo tanto la resolución de la PWM es de:

$$Resolucion_{PWM} = \frac{LOG(TOP + 1)}{LOG(2)} = 11,96 \cong 12 \text{ bits}$$

$$F_{PWM} = \frac{F_{clk}}{2 \cdot N \cdot TOP} = \frac{16MHz}{2 \cdot 1 \cdot 4000} = 2000Hz = 2KHz$$

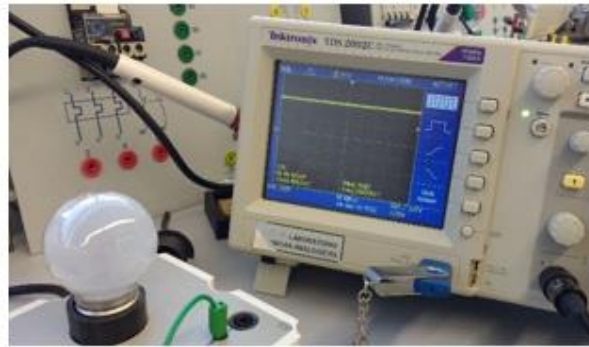
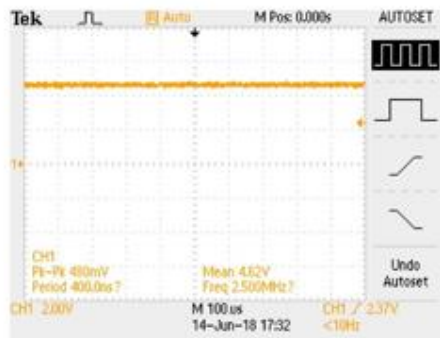


Figura 37- Señal PWM Duty Cycle: 0%

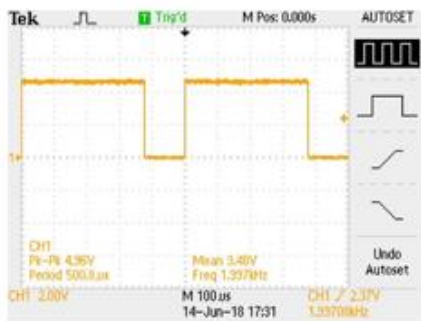


Figura 38- Señal PWM Duty Cycle: 15%

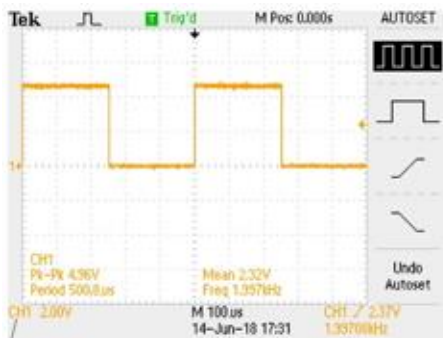


Figura 39- Señal PWM Duty Cycle: 50%



Figura 40- Señal PWM Duty Cycle: 100%

Como se puede observar, el circuito diseñado funciona correctamente.

Por otro lado, puesto que se quiere utilizar un control PID, será necesario diseñar un circuito para leer la señal del tacómetro. El circuito diseñado es el siguiente:

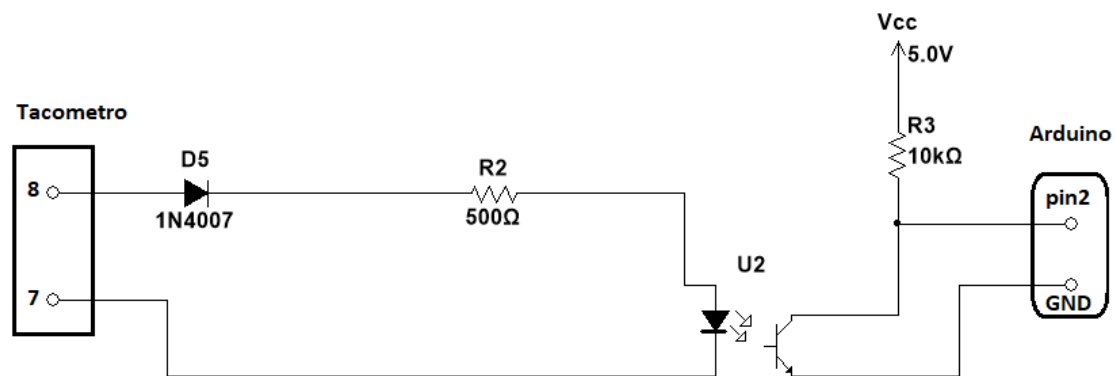


Figura 41- Circuito para la medición de frecuencia

El funcionamiento del circuito representado en la figura 41 es simple. El diodo D5 tiene la misma función que un rectificador de media onda, es decir, hace que solo se tengan tensiones negativas. La resistencia R2 se ha diseñado para limitar la corriente a 50mA ya que dicha corriente es la máxima que puede circular por el optoacoplador:

$$\frac{V_{rms\ max} - 0,7 - 2}{R_{min}} < I_{D\ max} \rightarrow \frac{30 - 0,7 - 2}{R_2} < 50mA \rightarrow R_2 = 500\Omega$$

NOTA: 0,7V es la caída del diodo D5 y 2V es la caída de tensión en el optoacoplador cuando está en estado de conducción.

Este circuito va a hacer que cuando el tacómetro supere la tensión umbral del diodo y del optoacoplador, se desactivará el pin 2 de arduino. Mientras que la tensión sea menor a la tensión umbral de dichos componentes, el pin 2 estará activado. Con estos flancos de bajada se activará una interrupción que capture el tiempo y así poder calcular el periodo de la señal del tacómetro.

## 4.3 Arduino. Diseño del control del motor

### 4.3.1 Arduino Uno

#### 4.3.1.1 Lazo abierto

Como se ha mencionado en el apartado anterior, para verificar el funcionamiento de la PCB, se ha elaborado un programa en lazo abierto con arduino uno. Para ello, es necesario conocer y configurar los siguientes registros:

#### Registro TCCR1A

Bit	7	6	5	4	3	2	1	0	
(0x80)	COM1A1	COM1A0	COM1B1	COM1B0	–	–	WGM11	WGM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Figura 42- Timer/counter1 Control Register A

**Bit 7:6 – compare output mode for channel A:** Estos bits controlan el comportamiento del pin de salida del comparado (OC1A). Si uno de los dos bits está a uno, OC1A ignora la funcionalidad normal del pin al que se encuentre conectado.

**Bit 5:4 – compare output mode for channel B:** Mismo comportamiento pero para el canal B.

Para utilizar las salidas OC1A y OC1B es necesario habilitarlas desde el registro Data Direction Register (DDR).

Seguidamente, se muestran las diferentes configuraciones posibles con estos cuatro bits mencionados:

Tabla 6- Funcionalidad de los bits COM1x1 cuando los bits WGM13:0 están configurados para no generar una señal PWM

COM1A1/COM1B1	COM1A0/COM1B0	Description
0	0	Normal port operation, OC1A/OC1B disconnected.
0	1	Toggle OC1A/OC1B on Compare Match.
1	0	Clear OC1A/OC1B on Compare Match (Set output to low level).
1	1	Set OC1A/OC1B on Compare Match (Set output to high level).

**Bit 1:0 – Waveform generation mode:** Con estos dos bits y con dos bits del registro que se detallará a continuación, se configura el tipo de señal que se quiere generar.

#### Registro TCCR1B

Bit	7	6	5	4	3	2	1	0	
(0x81)	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Figura 43- Timer/counter1 Control Register B

**Bit 7 –ICNC1 Input capture noise canceler:** No es necesario para esta aplicación.

**Bit 6 –ICES1 Input capture esge select:** Con este bit se define cuando se quiere hacer una captura de la señal, por ejemplo, con flanco de subida o flanco de bajada. Cuando IC1R se usa como el valor máxima (TOP), este pin se encuentra deshabilitado, es decir, la función de captura de pulsos está deshabilitada.

**Bit 4:3 Wave Generation mode:**

Tabla 7- Señales que pueden ser generadas y sus configuraciones

Mode	WGM13	WGM12 (CTC1)	WGM11 (PWM11)	WGM10 (PWM10)	Timer/Counter Mode of Operation	TOP	Update of OCR1x at	TOV1 Flag Set on
0	0	0	0	0	Normal	0xFFFF	Immediate	MAX
1	0	0	0	1	PWM, Phase Correct, 8-bit	0x00FF	TOP	BOTTOM
2	0	0	1	0	PWM, Phase Correct, 9-bit	0x01FF	TOP	BOTTOM
3	0	0	1	1	PWM, Phase Correct, 10-bit	0x03FF	TOP	BOTTOM
4	0	1	0	0	CTC	OCR1A	Immediate	MAX
5	0	1	0	1	Fast PWM, 8-bit	0x00FF	BOTTOM	TOP
6	0	1	1	0	Fast PWM, 9-bit	0x01FF	BOTTOM	TOP
7	0	1	1	1	Fast PWM, 10-bit	0x03FF	BOTTOM	TOP
8	1	0	0	0	PWM, Phase and Frequency Correct	ICR1	BOTTOM	BOTTOM
9	1	0	0	1	PWM, Phase and Frequency Correct	OCR1A	BOTTOM	BOTTOM
10	1	0	1	0	PWM, Phase Correct	ICR1	TOP	BOTTOM
11	1	0	1	1	PWM, Phase Correct	OCR1A	TOP	BOTTOM
12	1	1	0	0	CTC	ICR1	Immediate	MAX
13	1	1	0	1	(Reserved)	–	–	–
14	1	1	1	0	Fast PWM	ICR1	BOTTOM	TOP
15	1	1	1	1	Fast PWM	OCR1A	BOTTOM	TOP

En este caso se ha configurado el modo 10 (phase correct PWM mode) ya que el fabricante recomienda este modo para el control de motores. La ventaja de este modo es que es capaz de proporcionar a la señal una mayor resolución que las demás configuraciones. Por otro lado, tiene la desventaja de tener una menor frecuencia máxima que el modo Fast.



**Bit 2:0 Clock Select:** Con estos tres bits se selecciona el reloj que va a utilizar el temporizador/contador.

Tabla 8- Descripción de los bits de selección del reloj utilizado en el temporizador/contador

CS12	CS11	CS10	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clk <sub>IO</sub> /1 (No prescaling)
0	1	0	clk <sub>IO</sub> /8 (From prescaler)
0	1	1	clk <sub>IO</sub> /64 (From prescaler)
1	0	0	clk <sub>IO</sub> /256 (From prescaler)
1	0	1	clk <sub>IO</sub> /1024 (From prescaler)
1	1	0	External clock source on T1 pin. Clock on falling edge.
1	1	1	External clock source on T1 pin. Clock on rising edge.

En este caso, no se ha preescalado la señal del clock (configuración 001).

Finalmente, la configuración de los registros es la siguiente:

Tabla 9- Configuración final registro TCCR1A

COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10
1	0	0	0	0	0	1	0

Tabla 10- Configuración final del registro TCCR1B

ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10
0	0	0	1	0	0	0	1

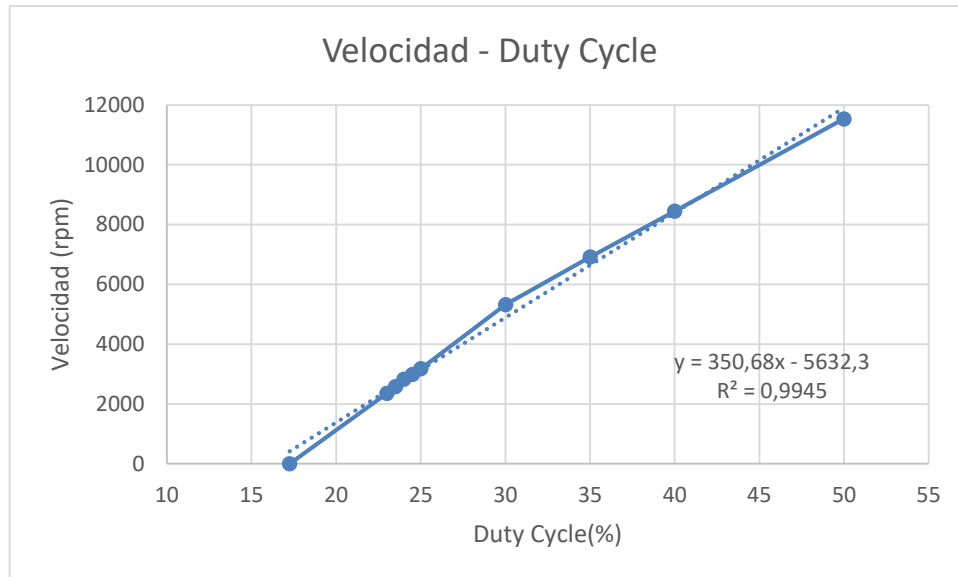
Una vez decidido cómo se van a configurar la señal PWM, se procede a cargar el programa 7.1 en el microcontrolador y conectar el motor al circuito de potencia. Para este caso, se ha generado una señal de 4KHz ganando así en frecuencia pero perdiendo resolución.

$$Resolucion_{PWM} = \frac{LOG(TOP + 1)}{LOG(2)} = \frac{LOG(2000 + 1)}{LOG(2)} = 10,96 \cong 11 \text{ bits}$$

$$F_{PWM} = \frac{F_{clk}}{2 \cdot N \cdot TOP} = \frac{16 \cdot 10^6 \text{ Hz}}{2 \cdot 1 \cdot 2000} = 4000 \text{ Hz} = 4 \text{ KHz}$$

Los resultados obtenidos son los siguientes:





Gráfica 5- Medición de la velocidad del motor variando el Duty cycle (Lazo abierto). Arduino Uno.

$$Velocidad = 350,68 \cdot DC - 5632,3$$

Haciendo uso de la ecuación característica, sabemos que para un duty cycle de 51,99% (valor de consigna 1039) la velocidad de salida es de 12600rpm (velocidad máxima que puede soportar el motor). Por lo tanto, todo el rango de velocidad se encuentra en un 34,74% del duty cycle, es decir, de la resolución máxima del timer 1 se aprovecha:

$$Resolucion_{PWM} = \frac{LOG(TOP + 1)}{LOG(2)} = \frac{LOG(1039 + 1)}{LOG(2)} = 10,02 \cong 10 \text{ bits}$$

### 4.3.2 Arduino Mega

#### 4.3.2.1 Lazo abierto

Como se ha mencionado anteriormente, el microcontrolador que se va a utilizar será el arduino mega ya que tienen unas características superiores las cuales se pueden consultar en el apartado 3.5.2. Por ello, se van a detallar la configuración de los registros que se necesitan para generar la señal PWM. A diferencia del arduino uno, ahora se va a utilizar el timer 5 y sus registros son los siguientes:

#### Registro TCCR5A

Bit	7	6	5	4	3	2	1	0	
(0x120)	COM5A1	COM5A0	COM5B1	COM5B0	COM5C1	COM5C0	WGM51	WGM50	TCCR5A
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Figura 44- Timer/counter5 Control Register A

**Bit 7:6 – compare output mode for channel A:** Estos bits controlan el comportamiento del pin de salida del comparado (OC5A). Si uno de los dos bits está a uno, OC5A ignora la funcionalidad normal del pin al que se encuentre conectado.

*Bit 5:4 – compare output mode for channel B:* Mismo comportamiento pero para el canal B.

*Bit 3:2 – compare output mode for channel C:* Mismo comportamiento pero para el canal C.

Para utilizar las salidas OC5A, OC5B y OC5C es necesario habilitarlas desde el registro Data Direction Register (DDR).

Seguidamente, se muestran las diferentes configuraciones posibles para la configuración de la PWM seleccionada (Phase correct PWM):

Tabla 11- Funcionalidad de los bits COM5x1 y COM5x0 cuando los bits WGM3:0 están configurados para generar una señal PWM en modo Phase correct PWM

COMnA1 COMnB1 COMnC1	COMnA0 COMnB0 COMnC0	Description
0	0	Normal port operation, OCnA/OCnB/OCnC disconnected
0	1	WGM13:0 =9 or 11: Toggle OC1A on Compare Match, OC1B and OC1C disconnected (normal port operation). For all other WGM1 settings, normal port operation, OC1A/OC1B/OC1C disconnected
1	0	Clear OCnA/OCnB/OCnC on compare match when up-counting Set OCnA/OCnB/OCnC on compare match when downcounting
1	1	Set OCnA/OCnB/OCnC on compare match when up-counting Clear OCnA/OCnB/OCnC on compare match when downcounting

*Bit 1:0 – Waveform generation mode:* Con estos dos bits y con dos bits del registro que se detallará a continuación, se configura el tipo de señal que se quiere generar.

### Registro TCCR5B

Bit	7	6	5	4	3	2	1	0	
(0x121)	ICNC5	ICES5	–	WGM53	WGM52	CS52	CS51	CS50	TCCR5B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Figura 45- Timer/counter5 Control Register B

*Bit 7 –ICNCn Input capture noise canceler:* No es necesario para esta aplicación.

*Bit 6 –ICESn Input capture esge select:* Con este bit se define cuando se quiere hacer una captura de la señal, por ejemplo, con flanco de subida o con flanco de bajada. Cuando ICRn se usa como el valor máxima (TOP), este pin se encuentra deshabilitado, es decir, la función de captura de pulsos está deshabilitada.

**Bit 4:3 Wave Generation mode:**

Tabla 12- Señales que pueden ser generadas y sus configuraciones

Mode	WGMn3	WGMn2 (CTCn)	WGMn1 (PWMn1)	WGMn0 (PWMn0)	Timer/Counter Mode of Operation	TOP	Update of OCRnX at	TOVn Flag Set on
0	0	0	0	0	Normal	0xFFFF	Immediate	MAX
1	0	0	0	1	PWM, Phase Correct, 8-bit	0x00FF	TOP	BOTTOM
2	0	0	1	0	PWM, Phase Correct, 9-bit	0x01FF	TOP	BOTTOM
3	0	0	1	1	PWM, Phase Correct, 10-bit	0x03FF	TOP	BOTTOM
4	0	1	0	0	CTC	OCRnA	Immediate	MAX
5	0	1	0	1	Fast PWM, 8-bit	0x00FF	BOTTOM	TOP
6	0	1	1	0	Fast PWM, 9-bit	0x01FF	BOTTOM	TOP
7	0	1	1	1	Fast PWM, 10-bit	0x03FF	BOTTOM	TOP
8	1	0	0	0	PWM, Phase and Frequency Correct	ICRn	BOTTOM	BOTTOM
9	1	0	0	1	PWM, Phase and Frequency Correct	OCRnA	BOTTOM	BOTTOM
10	1	0	1	0	PWM, Phase Correct	ICRn	TOP	BOTTOM
11	1	0	1	1	PWM, Phase Correct	OCRnA	TOP	BOTTOM
12	1	1	0	0	CTC	ICRn	Immediate	MAX
13	1	1	0	1	(Reserved)	-	-	-
14	1	1	1	0	Fast PWM	ICRn	BOTTOM	TOP
15	1	1	1	1	Fast PWM	OCRnA	BOTTOM	TOP

En este caso se ha configurado el modo 10 (phase correct PWM mode) ya que el fabricante recomienda este modo para el control de motores. La ventaja de este modo es que es capaz de proporcionar a la señal una mayor resolución que las demás configuraciones. Por otro lado, tiene la desventaja de tener una menor frecuencia máxima que el modo Fast.

**Bit 2:0 Clock Select:** Con estos tres bits se selecciona el reloj que va a utilizar el temporizador/contador.

Tabla 13- Descripción de los bits de selección del reloj utilizado en el temporizador/contador

CSn2	CSn1	CSn0	Description
0	0	0	No clock source. (Timer/Counter stopped)
0	0	1	$clk_{IC}/1$ (No prescaling)
0	1	0	$clk_{IC}/8$ (From prescaler)
0	1	1	$clk_{IC}/64$ (From prescaler)
1	0	0	$clk_{IC}/256$ (From prescaler)
1	0	1	$clk_{IC}/1024$ (From prescaler)
1	1	0	External clock source on Tn pin. Clock on falling edge
1	1	1	External clock source on Tn pin. Clock on rising edge

En este caso, se ha utilizado una preescala de 8 (configuración 010).

Finalmente, la configuración de los registros será la siguiente:

Tabla 14- configuración final del registro TCCR5A

COM5A1	COM5A0	COM1B1	COM1B0	COM1C1	COM1C0	WGM51	WGM50
1	0	0	0	0	0	1	0

Tabla 15- configuración final del registro TCCR5B

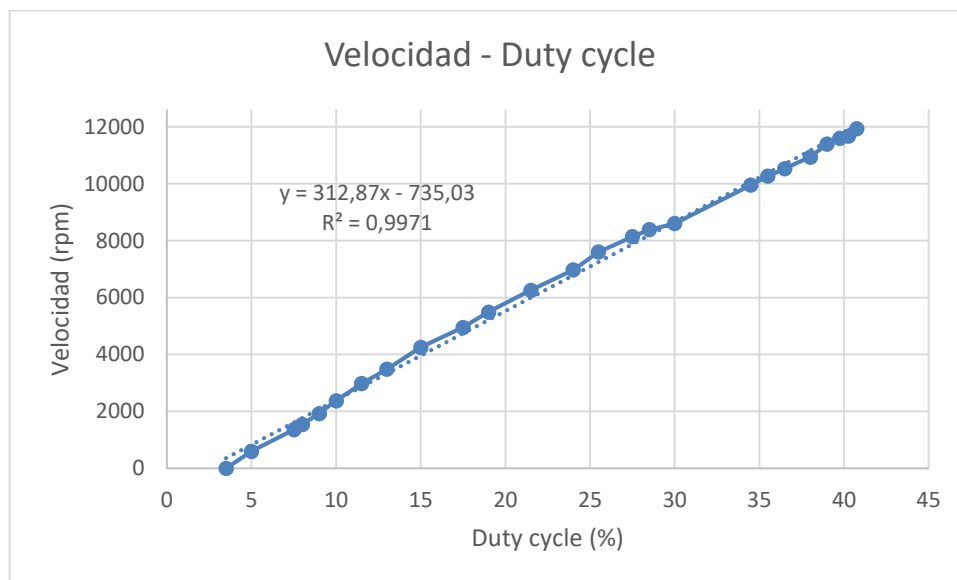
ICNC5	ICES5	-	WGM53	WGM52	CS52	CS51	CS50
0	0	0	1	0	0	1	0

A continuación, se calcula la resolución y la frecuencia de la salida PWM configurada:

$$Resolucion_{PWM} = \frac{LOG(TOP + 1)}{LOG(2)} = \frac{LOG(2000 + 1)}{LOG(2)} = 10,96 \cong 11 \text{ bits}$$

$$F_{PWM} = \frac{F_{clk}}{2 \cdot N \cdot TOP} = \frac{16 \cdot 10^6 \text{ Hz}}{2 \cdot 8 \cdot 2000} = 500 \text{ Hz}$$

Con la finalidad de ver si realmente merece la pena utilizar un microcontrolador más potente que arduino uno, como es el arduino mega, se va a realizar la misma práctica en el laboratorio. Cargando el *programa 7.2* en el microcontrolador, se ha medida la velocidad de salida del motor para diferentes valores de la consigna:



Gráfica 6- Arduino Mega. Relación duty cycle - velocidad

En este caso se han tomado un mayor número de muestras (25) para que la medida sea más fiable.

$$Velocidad = 312,87 \cdot DC - 735,03$$

Haciendo uso de la ecuación característica, sabemos que para un duty cycle de 42,62% (valor de consigna 852,4) la velocidad de salida es de 12600rpm (velocidad máxima que puede soportar el motor). Por lo tanto, todo el rango de velocidad se encuentra en un 39,12% del duty cycle, es decir, de la resolución máxima del timer 5 se aprovecha:

$$R = \frac{\log(TOP + 1)}{\log(2)} = \frac{\log(852,4 + 1)}{\log(2)} = 9,7bits \approx 10bits$$

#### 4.3.2.2 Lazo cerrado

Una vez verificado el correcto funcionamiento en lazo abierto, se va a cerrar el lazo de velocidad. Para ello, se ha utilizado el timer 4.

La función del timer 4 es generar una interrupción cada 50ms y recalculer el error de posición, error integral y el error diferencial. A continuación se muestran los registros de dicho timer que se deben manejar:

#### Registro TCCR4A

Bit (0xA0)	7	6	5	4	3	2	1	0
	COM4A1	COM4A0	COM4B1	COM4B0	COM4C1	COM4C0	WGM41	WGM40
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

Figura 46- registro TCCR4A

**Bit 7:6 – compare output mode for channel A:** Al igual que el timer 5, estos bits controlan el comportamiento del pin de salida del comparado (OC4A). Si uno de los dos bits está a uno, OC4A ignora la funcionalidad normal del pin al que se encuentre conectado.

**Bit 5:4 – compare output mode for channel B:** Mismo comportamiento pero para el canal B.

**Bit 3:2 – compare output mode for channel C:** Mismo comportamiento pero para el canal C.

Para utilizar las salidas OC4A, OC4B y OC4C es necesario habilitarlas desde el registro Data Direction Register (DDR).

Seguidamente, se muestran las diferentes configuraciones posibles para la configuración del timer seleccionada:

Tabla 16- Funcionalidad de los bits COM5x1 y COM5x0 cuando los bits WGM3:0 están configurados en modo CTC

COMnA1 COMnB1 COMnC1	COMnA0 COMnB0 COMnC0	Description
0	0	Normal port operation, OCnA/OCnB/OCnC disconnected
0	1	Toggle OCnA/OCnB/OCnC on compare match
1	0	Clear OCnA/OCnB/OCnC on compare match (set output to low level)
1	1	Set OCnA/OCnB/OCnC on compare match (set output to high level)

*Bit 1:0 – Waveform generation mode:* Con estos dos bits se configura el tipo de señal que se quiere general.

En este caso, el timer 4 va a trabajar como un temporizador, es decir, no se va a configurar para sacar una señal PWM. Por ello, el modo de operación del timer será el CTC (modo 4 de la tabla 12).

### Registro TCCR4B

Bit	7	6	5	4	3	2	1	0
(0xA1)	ICNC4	ICES4	–	WGM43	WGM42	CS42	CS41	CS40
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

Figura 47- registro TCCR4A

*Bit 7 – ICNCn Input capture noise canceler:* No es necesario para esta aplicación.

*Bit 6 – ICESn Input capture esge select:* Con este bit se define cuando se quiere hacer una captura de la señal, por ejemplo, con flanco de subida o con flanco de bajada. Cuando ICRn se usa como el valor máxima (TOP), este pin se encuentra deshabilitado, es decir, la función de captura de pulsos está deshabilitada.

*Bit 4:3 Wave Generation mode:* Junto con los bits 1:0 del registro TCCR4A, se selecciona el modo de trabajo de timer 4 previamente citado.

*Bit 2:0 Clock Select:* Con estos tres bits se selecciona el reloj y la preescala que va a utilizar el temporizador/contador. En este caso, se utilizará el reloj primario sin preescala (tabla 13).

### OCR4A

Con este registro se definirá cada cuanto tiempo saltará la interrupción, es decir, cada cuanto realizará cálculos el PID. En este caso, se ha fijado de la siguiente forma:

Tiempo requerido: 50ms

Por lo tanto, Frecuencia=20Hz

$$F_{OCn} = \frac{F_{clk}}{2 \cdot N \cdot (1 + OCR4A)} \rightarrow OCR4A = 399999$$

### Registro TIMSK4

Bit	7	6	5	4	3	2	1	0
(0x72)	–	–	ICIE4	–	OCIE4C	OCIE4B	OCIE4A	TOIE4
Read/Write	R	R	R/W	R	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

Figura 48- Registro TIMSK4

*Bit 5 – ICIE4: timer/counter4, Input Capture Interrupt Enable:* Con este pin se habilita la posibilidad de generar una interrupción con el input capture.

**BIT 3- OCIE4C, timer/counter4, Output Capture C Match Interrupt enable** Con este pin se habilita la posibilidad de generar una interrupción con el output capture del canal C.

**BIT 2- OCIE4B, timer/counter4, Output Capture C Match Interrupt enable:** Misma función que el bit 3 pero para el canal B.

**BIT 1- OCIE4A, timer/counter4, Output Capture C Match Interrupt enable:** Misma función que el bit 3 pero para el canal A.

**BIT 0- TOIE4, timer/counter4, Overflow Interrupt Enable:** Con este pin se habilita la posibilidad de generar una interrupción por rebose del timer 4.

Como se ha comentado en el apartado 4.2, se va a medir la frecuencia del tacómetro mediante interrupciones. Para ello, se ha utilizado la función de arduino `AttachInterrupt()`. La función `attachInterrupt()`, realiza una interrupción cuando en el pin elegido hay un cambio a high, a low, con cualquier cambio, o si está permanentemente en high o en low. Se crean dos subprogramas que se van llamando una a la otra de manera secuencial cuando detectan un cambio de High a Low. Se calcula el tiempo entre las dos interrupciones con la función `micros()`, ya que funciones como `millis()` o `delay()` no se pueden utilizar en este tipo de interrupciones. Las variables que se utilizan en estos programas deben ser declarados como volátiles (Llamas, 2016).

```
void rising()
{
  attachInterrupt(4, falling, FALLING);
  prev_time = micros(); //Obtenemos tiempo
}
void falling()
{
  attachInterrupt(4, rising, FALLING);
  periodo_value = micros()-prev_time;
}
```

Finalmente, la configuración de los registros será la siguiente:

Tabla 17- Configuración del registro TCCR4A

COM4A1	COM4A0	COM4B1	COM4B0	COM4C1	COM4C0	WGM41	WGM40
0	0	0	0	0	0	0	0

Tabla 18- Configuración del registro TCCR4B

ICNC4	ICES4	-	WGM43	WGM42	CS42	CS41	CS40
0	0	0	0	1	0	0	1

Tabla 19- Configuración del registro TIMSK4

-	-	ICIE4	-	OCIE4C	OCIE4B	OCIE4A	TOIE4
0	0	0	0	0	0	1	0

## 5. RESULTADOS

### 5.1 Conclusiones

La obsolescencia programada ha sido y siempre será un enemigo para el desarrollo sostenible. Con este proyecto se ha argumentado la necesidad del uso de la tecnología ética y sostenible lo cual conlleva, mayor practicidad, mayor durabilidad del producto y un menor impacto ambiental. Para ello las tecnologías Open-Source son la herramienta necesaria, ya que habilitan la lucha contra las malas prácticas de algunas empresas privadas haciendo que el conocimiento sea compartido, de esta forma, diferentes usuarios pueden crear nuevas posibilidades que permitan el desarrollo la tecnología.

Si bien a primera vista el tener la posibilidad de comprar electrodomésticos baratos parece buena, sus consecuencias ambientales y sociales no lo compensan. Por ello, es necesario que sean los propios consumidores los que encuentren vías para detener esta situación. No se trata solo de las vías legales ya comentadas, sino de las iniciativas de reciclaje y reutilización en las que hay priorizar el beneficio social sobre el económico, pero haciendo que su desarrollo sea viable.

A nivel técnico se ha descartado crear un sistema universal de lavadoras. Los motores que se utilizan son muy diferentes y la única solución a este problema sería crear un sistema modular que permita intercambiar los controladores de los motores. Por lo tanto, se ha empezado por el control del motor más usual: el motor universal. Se ha descartado utilizarlo en modo alterna ya que en este modo la mecánica del motor sufre más y hay que tener en cuenta que se va a trabajar con equipos de segunda mano, los cuales ya están desgastados. Por consiguiente, se ha elegido la opción de alimentarlo en corriente continua y ser controlado mediante un transistor IGBT y una señal PWM. Se ha descartado utilizar controladores analógicos. Son controladores obsoletos y muy rígidos. En este caso además, el objetivo es cubrir una gran variedad de motores con diferentes características mecánicas y eléctricas y con diferentes tipos de tacómetros que necesitan de un sistema de control digital, que pueda adaptarse de forma sencilla a cada uno.

Los microcontroladores de Arduino utilizados tienen unas capacidades diferentes. Por un lado el Uno no permite realizar un control lo suficientemente estable pero su precio resulta muy competitivo. Por ellos, se ha utilizado su hermano mayor Mega. Este tiene recursos suficientes para poder controlar correctamente el tiempo de muestreo y a la vez tomar los datos de entrada. Aun así la respuesta del motor no cambia mucho ya que la resolución de su PWM es igual que la del Arduino Uno.



## 5.2 Líneas de desarrollo futuras

Como se ha mencionado anteriormente, este proyecto es a largo plazo ya que se requiere un control del motor que funcione perfectamente y además añadir el resto de actuadores como la bomba de agua y las válvulas. Por ello, el siguiente paso sería realizar el ajuste del PID y ver si realmente merece la pena implantar un PID o es deficiente con un control PI.

En la asignatura sistemas digitales de control se han adquirido conocimientos acerca de cómo ajustar un controlador PI para un motor de corriente continua, como por ejemplo, haciendo uso de la guía de usuario de Texas instrument “InstaSPIN-FOC and InstaSPIN-MOTION”, la nota de aplicación AN4642 de Freescale o la tesis “Sensorless Control of Permanent-Magnet Synchronous Motor Drivers” de Chandana Perera. En este caso, no se puede ajustar el PID con estos métodos ya que todos ellos requieren un conocimiento total de las características del motor y en el caso de los motores utilizados en las lavadoras, nunca se van a tener datos como la inercia o la inductancia del motor. Por lo tanto hay dos opciones:

La primera es ir ajustando por separado de forma empírica la parte proporcional, la parte integral y por último la parte derivativa para una consigna dada. Cada vez que se modifique alguna de las constantes del controlador, se debe graficar la salida y ver como se ajusta a la consigna dada. Una vez que la salida sea igual a la entrada y no se tenga un tiempo de establecimiento alto ni un rebose muy elevado, el ajuste del controlador se dará por satisfactorio.

La segunda opción es utilizar los métodos de sintonización de Ziegrel- Nichols. Son los métodos clásicos y más conocidos para determinar los parámetros de controlador PID. Usualmente este método se emplea para obtener valores iniciales y después se realiza un reajuste fino por métodos heurísticos (sintonía manual y prueba-error). Ziegrel- Nichols permite tratar el sistema desde un punto de vista externo entrada- salida sin necesitar un modelo interno detallado. El primer método se llama “método de la respuesta escalón”, que consiste en aproximar la respuesta del sistema en lazo abierto ante un escalón, considerando que el sistema es de primer orden. Sobre esta curva se trazando tangentes y se calculan parámetros que luego se ajustan mediante una tabla. Es un método gráfico que puede resultar más sencillo al trabajar con sistemas reales, pero también puede dar lugar a errores al interpretar la curva. El segundo método se llama “método de la respuesta en frecuencia” y se aplica al sistema en lazo cerrado. Partiendo de un control únicamente

proporcional, consiste en encontrar para qué ganancia el sistema presenta una oscilación de amplitud constante. Esa ganancia se llama  $K_u$  y su periodo  $T_u = 2\pi/\omega$ . En función de estos valores se obtienen los parámetros del controlador  $G_c = 1 + \frac{1}{T_s \cdot s} + T_d \cdot s$

Tabla 20- Ajuste PID. Método de la respuesta en frecuencia

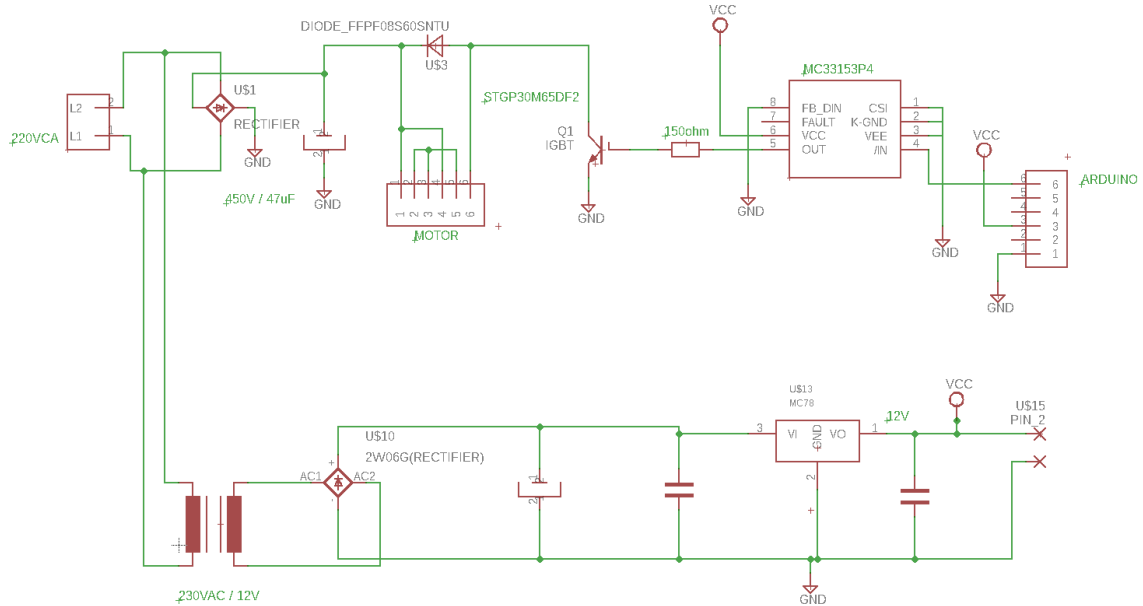
Controlador	$K_p$	$T_i$	$T_d$
<b>P</b>	$K_u/2$	-	-
<b>PI</b>	$K_u/2,2$	$T_u/1,2$	-
<b>PID</b>	$K_u/1,7$	$T_u/2$	$T_u/8$

Una vez creado el control del motor, el siguiente paso es diseñar el control del resto de elementos de la lavadora y ver como estos dos controles pueden compatibilizarse. Si se quiere desarrollar controles para otro tipo de motores será necesario pensar en un sistema modular con microcontroladores independientes donde haya un microcontrolador maestro. Si se decide seguir únicamente con los motores universales entonces es posible que microcontroladores como Arduino Mega sean capaces de controlar ellos solos todos los componentes de una lavadora. En caso de que este controlador no sea capaz, se podría valorar el uso del Arduino Due, ya que es mucho más potente. Habría que analizar este balance económico-practico y pensar en dar el paso a diseñar un modelo comercial. Puesto que el número de motores que se manejan en la instalación de Kooperera es enorme, sería interesante valorar el uso de una Raspberry y así el usuario se podría comunicar de forma remota (vía internet) al control de la lavadora.

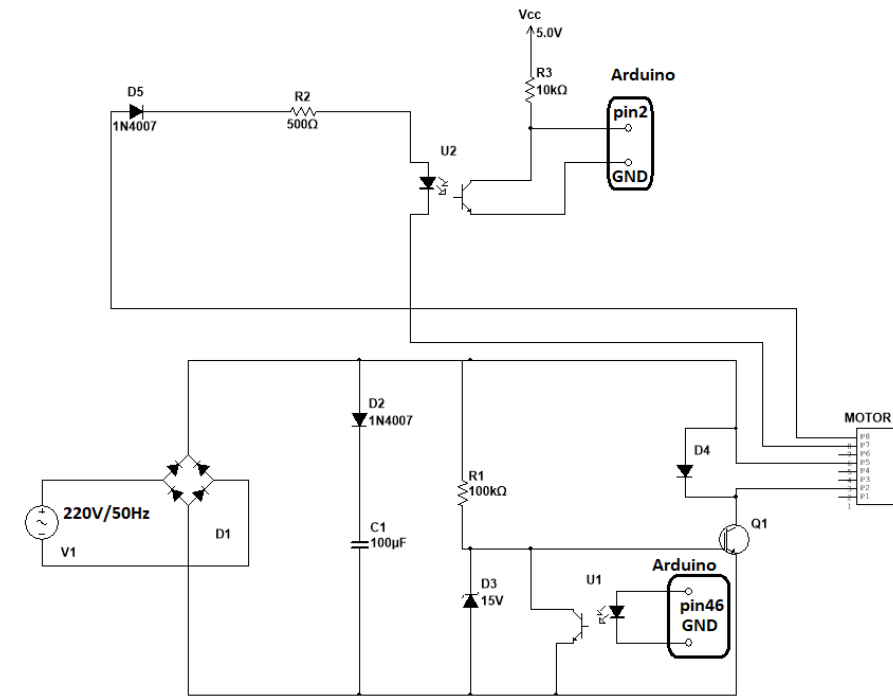
## 6. PIANOS

### 6.1 Circuito de potencia

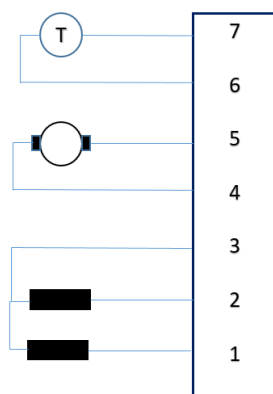
#### 6.1.1 Primera versión



#### 6.1.2 Versión definitiva



## 6.2 Conexiones del motor



## 7. PROGRAMAS

### 7.1 Variación de la velocidad del motor en lazo abierto con arduino uno

A continuación se muestra el programa utilizado para realizar la prueba de funcionamiento de la PCB diseñada, es decir, el programa utilizado para variar la luminosidad de la bombilla y posteriormente, la velocidad del motor.

```

SE EMPLEA EL TIMER 1 (16 BITS) EN MODO PWM PHASE CORRECT PARA GENERAR UNA
SEÑAL PWM DE 2KHZ

F=16MHz/(2*N*TOP) ---> Con N=1 --> TOP=4000

EL MICROCONTROLADOR TIENE EL PIN OC1A ASOCIADO AL TIMER 1 CABLEADO A LA SALIDA DIGITAL 9

*/

const int PWM_Pin = 9; //Pin de salida del TIMER 1
char cadena[24]; //string para almacenar los char recibidos del puerto serie
byte contador = 0; //contador para llevar la cuenta de characters recibidos

int Freq = 4000; //4000->2KHz.
int DutyCycle = Freq;

void setup() {

  pinMode(PWM_Pin, OUTPUT); //Configuración del pin como salida
  Serial.begin (19200);

  //CONFIGURACION DE LOS REGISTROS DEL TIMER 1

  TCCR1A = B10000010; //PWM PHASE CORRECT
  TCCR1B = B00010001; //N=1 When IC1 is used as TOP value-> input capture fuction in disabled (we configure this with the bits WGM13:0)
  IC1 = Freq; // TOP
  OCR1A = DutyCycle; //consigna
}

void loop(){
  if(Serial.available()){ //Dice si hay datos dentro del buffer
    memset(cadena, 0, sizeof(cadena)); //Limpia el array despues de cada lectura
    while (Serial.available()>0){ // Comprueba si en el buffer hay datos
      delay(5); //Pequeño delay para mejorar la recepcion de datos
      cadena[contador]=Serial.read(); // Lee cada caracter y lo almacena en cadena
      contador++;
    }
    DutyCycle=atoi(cadena); //Convierte la cadena de caracteres en enteros

    OCR1A=DutyCycle;
    contador=0;
  }
}

```

Para poder variar la velocidad mientras el programa está corriendo, se introduce la consigna mediante el puerto serie. La consigna puede variar entre 0 – 4000, siendo 0 la frecuencia máxima y 4000 la mínima. Para evitar que el motor sufra por arranques bruscos se decidió darle un valor inicial a duty cycle de 0, es decir, inicialmente el motor está parado.

Para graficar la gráfica 5 se modificó la frecuencia máxima de 2KHz a 4KHz. Para realizar este cambio simplemente se debe variar el valor de la variable *Freq= 2000*.

### 7.2 Variación de la velocidad del motor en lazo abierto con arduino mega

Como se puede observar en el siguiente código, en este caso se utiliza una pantalla LCD como interface con el usuario. En ella puede visualizar en todo momento la frecuencia de la señal PWM.

```

// include the library code:
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

char cadena[24];
byte contador=0;
int DutyCycle = 0; // engine stopped
int Freq = 2000;

void setup() {
  lcd.begin(16, 2); // set up the LCD's number of columns and rows:
  lcd.print("LECTURA SERIE"); // Print a message to the LCD.
  Serial.begin(19200);
  pinMode(46, OUTPUT);

  TCCR5A = B10101010; //phase correct pwm
  TCCR5B = B00010010; //Preescaler=8
  ICR5 = Freq;
  OCR5A = DutyCycle;
}

void loop(){
  if(Serial.available()){
    memset(cadena, 0, sizeof(cadena));
    while (Serial.available()>0){
      delay(5);
      cadena[contador]=Serial.read();
      contador++;
    }
    DutyCycle = atoi(cadena);
    contador=0;
  }

  OCR5A = Freq - DutyCycle;

  lcd.setCursor(0, 1);
  lcd.print(" ");
  lcd.setCursor(0, 1);
  lcd.print(DutyCycle);
  delay (200);
}

```

Al igual que en el código de arduino uno, para poder variar la velocidad del motor mientras el programa se está ejecutando, se utiliza el puerto serie para introducir la consigna. Dicha consigna puede variar entre 0 y 2000, siendo 0 la velocidad mínima y 2000 la velocidad máxima. Estos valores son teóricos ya que en el apartado 4.3.2.1 se ha demostrado que se alcanza la velocidad máxima con una consigna de 852,4 y el motor se para con un valor de consigna de 70.

Si se introduce una consigna menor a 70 no hay ningún problema, en cambio, si se introduce un valor mayor a 853.4, el motor podría llegar a sobrecalentarse y destruirse ya que intentará alcanzar una velocidad mayor que su velocidad máxima permitida.

## 7.3 Variación de la velocidad del motor en lazo cerrado con arduino mega

```

#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

char cadena[24];
int filtro[6];
byte contador=0;
int DutyCycle = 0; // Inicialmente motor parado (consigna)
int Freq = 2000; //TOP
float tacoFreq=0, mediaTacoF=0, tacoRpm=0;
int consignaRpm, i; //es la variable DutyCycle pero en rpm
float error_pos=0, error_dif=0, error_int=0, error_pos_ant=0;
int n=6;
float salida_PID, Kp, Ki, Kd;
volatile int periodo_value=0, prev_time=0;

void setup() {
  lcd.begin(16, 2); // set up the LCD's number of columns and rows:
  lcd.print("LECTURA SERIE"); // Print a message to the LCD.
  Serial.begin(19200);
  pinMode(46, OUTPUT);
  TCCR5A = B10101010; //phase correct PWM
  TCCR5B = B00010010; //Preescaler=8
  ICR5 = Freq;
  OCR5A = DutyCycle;
  //***** CONFIGURACION DEL TIMER 4 COMO FUENTE DE INTERRUPCION *****
  TCCR4A = B00000000; //WGM41=0 y WGM40=0
  TCCR4B = B00001001; //001-1 010-8 011-64 100-256 101-1024 . Preescaler=1
  OCR4A = 399999; //T=(1/Fosc)*2*PS*(1+OCR4A)
  TIMSK4 = B00000010;
}

void loop(){
  if(Serial.available()){
    memset(cadena, 0, sizeof(cadena));
    while (Serial.available()>0){
      delay(5);
      cadena[contador]=Serial.read();
      contador++;
    }
    DutyCycle = atoi(cadena);
    contador=0;
  }

  OCR5A = Freq - DutyCycle;
  lcd.setCursor(0, 1);
  lcd.print(" ");
  lcd.setCursor(0, 1);
  lcd.print(DutyCycle);
  delay (200);
}
//***** CADA 50ms el programa principal se detiene y se ejecuta la siguiente interrupción *****
ISR(TIMER4_COMPA_vect) {

  for (int i=0; i<n-1; i++) //Desplazo el array a la izquierda
  {
    filtro[i]=filtro[i+1];
  }
  tacoFreq=1000000/periodo_value; // Conversion de microsegundos a segundos y por ultimo a Hz
  filtro[n-1] = tacoFreq; //La Frecuencia actual se guarda al final del array

  tacoFreq=0;
  for (i=0; i<n; i++) //Medimos 6 veces el valor del tacometro para mayor precisión
  {
    tacoFreq= tacoFreq+filtro[i];
  }
  mediaTacoF= tacoFreq/n;
  tacoRpm=7.4664*mediaTacoF+82.085; // Pasamos de frecuencia a rpm
  consignaRpm=15.643*DutyCycle-735.03;
  error_pos_ant= error_pos;
}

```

```
error_pos=consignaRpm-tacoRpm;
error_dif= error_pos - error_pos_ant;
error_int = error_int + error_pos;
salida_PID = Kp * error_pos + Kd * error_dif + Ki * error_int;
DutyCycle= DutyCycle + salida_PID;
DutyCycle= constrain(DutyCycle, 0, 850);

}

//***** Medida de la frecuencia de la señal del tacometro*****
void rising()
{
  attachInterrupt(4, falling, FALLING); // Si en el pin 2 (INT4) hay un flanco de bajada, llama a la función falling
  cli(); // mientras que se calcula el periodo de la señal, nos aseguramos de que ninguna interrupción se active
  prev_time = micros(); //Obtenemos el tiempo el el cual se ha generado la interrupción
  sei(); // rehabilitamos las interrupciones
}

void falling()
{
  attachInterrupt(4, rising, FALLING); // Si en el pin 2 (INT4) hay un flanco de bajada, llama a la función rising
  cli(); // mientras que se calcula el periodo de la señal, nos aseguramos de que ninguna interrupción se active
  periodo_value = micros()-prev_time; //Medimos el periodo de la señal del tacómetro
  sei(); // rehabilitamos las interrupciones
}
```



## 8. PRESUPUESTO

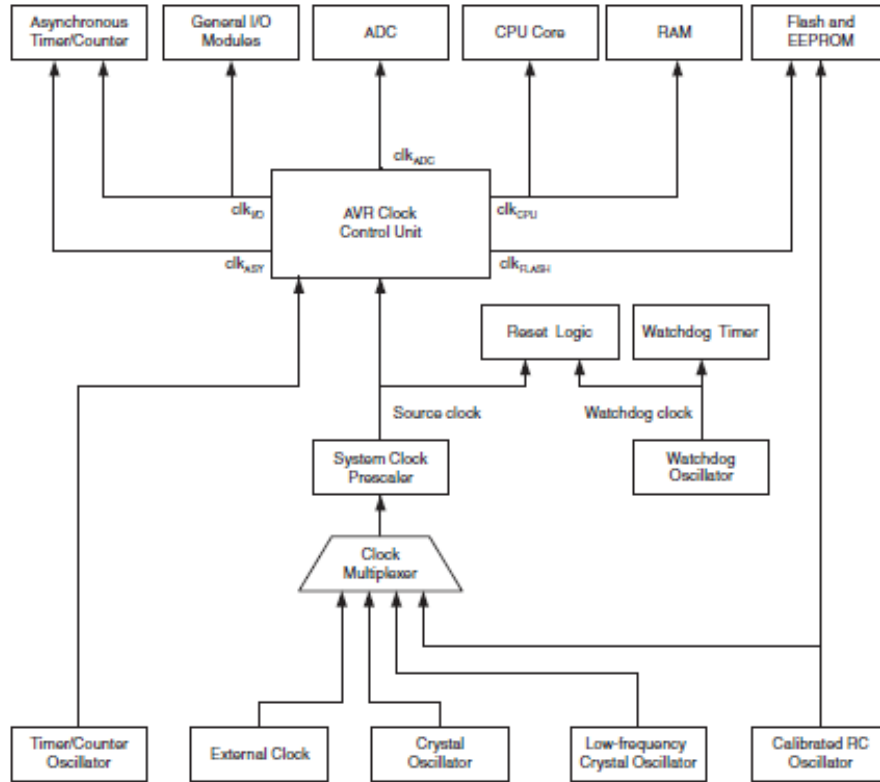
Componente	Denominación	Referencia	Necesidad	Precio unitario (€)	Precio unitario - inc. IVA (€)	Lote mínimo	Link de compra
Rectificador	D1	GBU4J	1	0,54	0,68	5	<a href="#">Comprar</a>
Diodo	D2	1N4007	2	0,116	0,146	50	<a href="#">Comprar</a>
Condensador	C1	-	1	0,7	0,85	2	<a href="#">Comprar</a>
Resistencia	R1	100kΩ	1	0,091	0,111	10	<a href="#">Comprar</a>
Resistencia	R2	500Ω	1	0,174	0,214	10	<a href="#">Comprar</a>
Resistencia	R3	10kΩ	1	0,045	0,055	10	<a href="#">Comprar</a>
Zener 15V	D3	1N5352BG	1	0,353	0,423	10	<a href="#">Comprar</a>
Optoacoplador	U1, U2	H11D1	2	0,562	0,682	10	<a href="#">Comprar</a>
Diodo	D4	FFPF08S60SN	1	0,692	0,832	5	<a href="#">Comprar</a>
Transistor IGBT	Q1	STGP30H65F	1	2,31	2,8	2	<a href="#">Comprar</a>
Arduino Mega	-	-	1	43,51	52,65	1	<a href="#">Comprar</a>

Coste IVA no incluido: 73,74€

Coste IVA incluido: 89,66€

## 9. ANEXOS

### 9.1 Arduino Mega and arduino Uno- Clock distribution

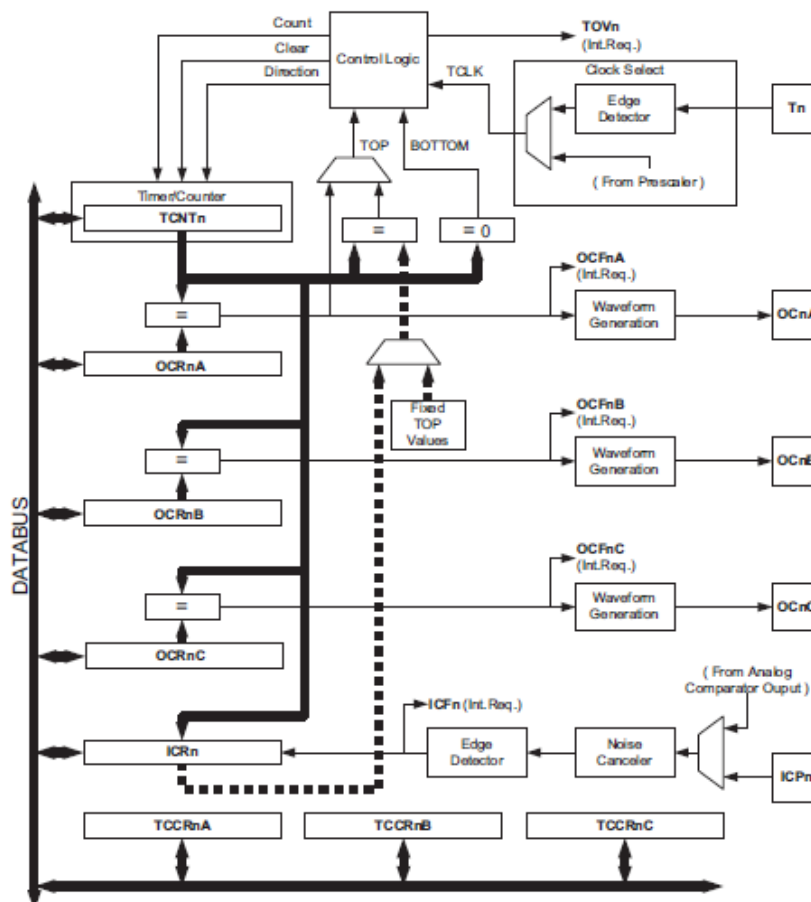


## 9.2 Reset and interrupt Vectors

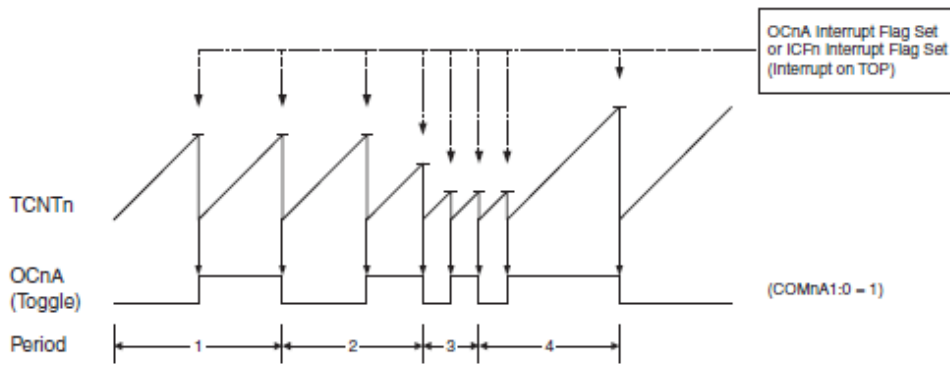
Vector No.	Program Address <sup>(2)</sup>	Source	Interrupt Definition
1	\$0000 <sup>(1)</sup>	RESET	External Pin, Power-on Reset, Brown-out Reset, Watchdog Reset, and JTAG AVR Reset
2	\$0002	INT0	External Interrupt Request 0
3	\$0004	INT1	External Interrupt Request 1
4	\$0006	INT2	External Interrupt Request 2
5	\$0008	INT3	External Interrupt Request 3
6	\$000A	INT4	External Interrupt Request 4
7	\$000C	INT5	External Interrupt Request 5
8	\$000E	INT6	External Interrupt Request 6
9	\$0010	INT7	External Interrupt Request 7
10	\$0012	PCINT0	Pin Change Interrupt Request 0
11	\$0014	PCINT1	Pin Change Interrupt Request 1
12	\$0016 <sup>(3)</sup>	PCINT2	Pin Change Interrupt Request 2
13	\$0018	WDT	Watchdog Time-out Interrupt
14	\$001A	TIMER2 COMPA	Timer/Counter2 Compare Match A
15	\$001C	TIMER2 COMPB	Timer/Counter2 Compare Match B
16	\$001E	TIMER2 OVF	Timer/Counter2 Overflow
17	\$0020	TIMER1 CAPT	Timer/Counter1 Capture Event
18	\$0022	TIMER1 COMPA	Timer/Counter1 Compare Match A
19	\$0024	TIMER1 COMPB	Timer/Counter1 Compare Match B
20	\$0026	TIMER1 COMPC	Timer/Counter1 Compare Match C
21	\$0028	TIMER1 OVF	Timer/Counter1 Overflow
22	\$002A	TIMER0 COMPA	Timer/Counter0 Compare Match A
23	\$002C	TIMER0 COMPB	Timer/Counter0 Compare match B
24	\$002E	TIMER0 OVF	Timer/Counter0 Overflow
25	\$0030	SPI, STC	SPI Serial Transfer Complete
26	\$0032	USART0 RX	USART0 Rx Complete
27	\$0034	USART0 UDRE	USART0 Data Register Empty
28	\$0036	USART0 TX	USART0 Tx Complete
29	\$0038	ANALOG COMP	Analog Comparator
30	\$003A	ADC	ADC Conversion Complete

Vector No.	Program Address <sup>(2)</sup>	Source	Interrupt Definition
31	\$003C	EE READY	EEPROM Ready
32	\$003E	TIMER3 CAPT	Timer/Counter3 Capture Event
33	\$0040	TIMER3 COMPA	Timer/Counter3 Compare Match A
34	\$0042	TIMER3 COMPB	Timer/Counter3 Compare Match B
35	\$0044	TIMER3 COMPC	Timer/Counter3 Compare Match C
36	\$0046	TIMER3 OVF	Timer/Counter3 Overflow
37	\$0048	USART1 RX	USART1 Rx Complete
38	\$004A	USART1 UDRE	USART1 Data Register Empty
39	\$004C	USART1 TX	USART1 Tx Complete
40	\$004E	TWI	2-wire Serial Interface
41	\$0050	SPM READY	Store Program Memory Ready
42	\$0052 <sup>(3)</sup>	TIMER4 CAPT	Timer/Counter4 Capture Event
43	\$0054	TIMER4 COMPA	Timer/Counter4 Compare Match A
44	\$0056	TIMER4 COMPB	Timer/Counter4 Compare Match B
45	\$0058	TIMER4 COMPC	Timer/Counter4 Compare Match C
46	\$005A	TIMER4 OVF	Timer/Counter4 Overflow
47	\$005C <sup>(3)</sup>	TIMER5 CAPT	Timer/Counter5 Capture Event
48	\$005E	TIMER5 COMPA	Timer/Counter5 Compare Match A
49	\$0060	TIMER5 COMPB	Timer/Counter5 Compare Match B
50	\$0062	TIMER5 COMPC	Timer/Counter5 Compare Match C
51	\$0064	TIMER5 OVF	Timer/Counter5 Overflow
52	\$0066 <sup>(3)</sup>	USART2 RX	USART2 Rx Complete
53	\$0068 <sup>(3)</sup>	USART2 UDRE	USART2 Data Register Empty
54	\$006A <sup>(3)</sup>	USART2 TX	USART2 Tx Complete
55	\$006C <sup>(3)</sup>	USART3 RX	USART3 Rx Complete
56	\$006E <sup>(3)</sup>	USART3 UDRE	USART3 Data Register Empty
57	\$0070 <sup>(3)</sup>	USART3 TX	USART3 Tx Complete

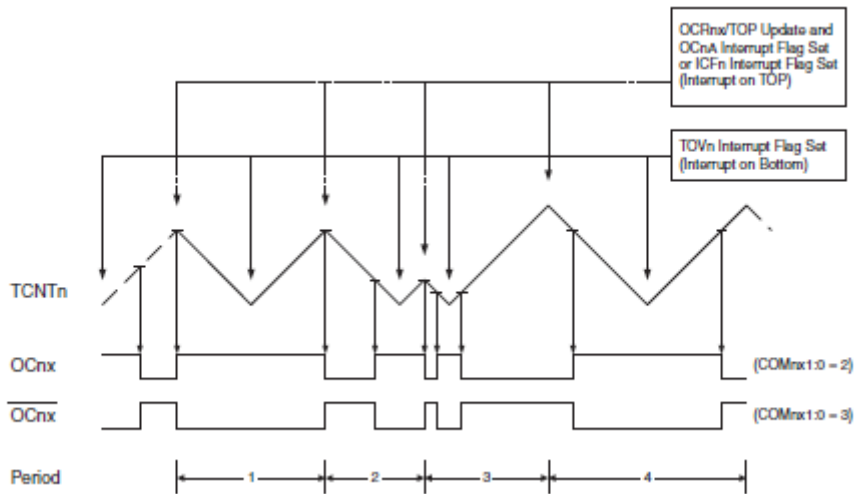
### 9.3 16bit Timer Blck Diagram



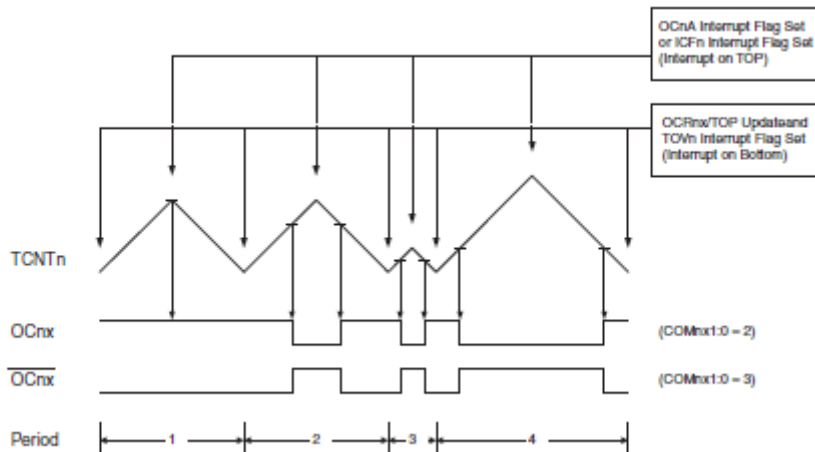
### 9.4 CTC Mode- Timing diagram



### 9.5 Phase correct PWM – Timing diagram



### 9.6 Phase and frequency correct PWMmode – Timing diagram



## 10. BIBLIOGRAFIA

- (2013). Obtenido de Mongabay Latam: <https://kidsnews.mongabay.com/2013/07/apple-investigating-if-tin-for-smart-phones-involved-child-labor/>
- (2016). Obtenido de Somos Triodos: <https://www.somostriodos.com/como-luchar-contra-la-obsolencia-programada-en-3-pasos/>
- (2017). Obtenido de statista: <https://es.statista.com/estadisticas/504552/tecnologias-para-el-hogar-ingresos-anuales-espana/>
- (2017). Obtenido de La Ser: [http://cadenaser.com/ser/2017/07/05/ciencia/1499239540\\_041988.html](http://cadenaser.com/ser/2017/07/05/ciencia/1499239540_041988.html)
- analogica, E. (2016). *Apuntes de la asignatura de electronica analogica*.
- Aprende Arduino*. (2016). Obtenido de <https://aprendiendoarduino.wordpress.com/2016/09/25/que-es-arduino/>
- Asignatura de Regulación y control*. (2016).
- Dannoritzer, C. (Dirección). (2011). *Comprar, tirar, comprar* [Película].
- Eco medio ambiente*. (11 de febrero de 2016). Obtenido de <http://ecomedioambiente.com/consumo-responsable/obsolencia-programada-consecuencias/>
- Garcia, M. A. (2014). *Instrumentación electronica*. Paraninfo.
- Industrial, e. (2015). *Apuntes de la asignatura Electronica industrial*.
- Ingeniería para el desarrollo humano. (s.f.). Guía de consumo responsable de productos electrónicos. 2013.
- Llamas, L. (26 de Abril de 2016). *Luis Llamas*. Obtenido de <https://www.luisllamas.es/que-son-y-como-usar-interrupciones-en-arduino/>
- On semiconductor. (s.f.). *AllDataSheet*. Obtenido de [//www.alldatasheet.es/datasheet-pdf/pdf/12020/ONSEMI/MC33153P.html](http://www.alldatasheet.es/datasheet-pdf/pdf/12020/ONSEMI/MC33153P.html)
- Ottaviani, J. (2018). *El Pais*. Obtenido de <https://elpais.com/especiales/2015/basura-electronica/>
- Protagoniatec*. (s.f.). Obtenido de <https://saber.patagoniatec.com/2014/06/arduino-mega-2560-atmega-mega-arduino-clon-compatible-argentina-tutorial-basico-informacion-arduino-argentina-ptec/>
- SED. (2017). *Apuntes de la asignatura Sistemas Electronicos Digitales*.
- Wikipedia*. (2018). Obtenido de [https://es.wikipedia.org/wiki/Rectificador#Rectificaci%C3%B3n\\_monof%C3%A1sica\\_no\\_controlada](https://es.wikipedia.org/wiki/Rectificador#Rectificaci%C3%B3n_monof%C3%A1sica_no_controlada)