

Implementación de práctica de laboratorio con acceso remoto utilizando el microcontrolador
Arduino Uno para el control de un motor DC

Jhonatan Sneyder Ortiz Santos

Trabajo de grado para optar por el título de Ingeniero Electricista.

Director

María Alejandra Mantilla Villalobos

Docente Universidad Industrial De Santander Doctora En Ingeniería

Universidad Industrial de Santander

Facultad de Ingenierías Físico-Mecánicas

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Bucaramanga

2023

Dedicatoria

A mis padres por el apoyo emocional e incondicional de principio a fin.

A mi tía por el apoyo durante toda mi estancia en la universidad, sin ella nada de esto hubiese sido posible.

A María Fernanda, por siempre estar ahí en los momentos más difíciles, ayudándome a salir adelante a pesar de la complejidad de los problemas.

A Carmen Sofía, que me daba palabras de aliento en momentos difíciles, también por su cariño y enseñanzas.

Jhonatan Sneyder Ortiz Santos

Tabla de contenido

Introducción	11
1. Objetivos	13
1.1 Objetivo general	13
1.2 Objetivos específicos	13
2. Justificación y motivación	14
3. Elementos y componentes	16
3.1 Esquema propuesto	16
3.2 Microcontrolador Arduino Uno	17
3.3 Software Arduino Uno IDE	19
3.4 Motor DC	20
3.4.1 <i>Principios teóricos del motor de corriente continua</i>	20
3.4.1.1 Motor de excitación serie.	21
3.4.1.2 Motor de excitación en paralelo o motor	21
3.4.1.3 Motor de excitación compuesta o motor	21
3.4.2 <i>Elección del motor</i>	23
3.4.3 <i>Parámetros del motor</i>	24
3.5 Fuentes de alimentación y convertidor DC/DC reductor	26
3.5.2 <i>Convertidor DC-DC reductor</i>	27
3.5.3 <i>Circuito de adecuación y aislamiento de señales (driver).</i>	29
3.5.4 <i>Transformador de potencia</i>	30
3.5.5 <i>Fuentes de alimentación</i>	31
3.5.6 <i>Filtro de red</i>	34

3.6 Módulo codificador rotatorio (<i>encoder</i>)	36
3.7 Extensión Google desktop	37
3.8 Computadores	37
3.9 Módulo de potencia y control implementado	37
4. Guías básicas para el control de un motor DC con acceso remoto	39
4.1 Interfaz y comandos de Arduino Uno	39
4.1.1 <i>Interfaz de programación de Arduino Uno</i>	39
4.1.1.1 Accesos directos.	40
4.1.1.2 Barra de menús.	40
4.1.1.3 Interfaz de programación de código.	41
4.1.1.4 Interfaz de salida del código.	41
4.2 Comandos	41
4.3 Control propuesto	43
4.4 Cálculo de la velocidad a partir de la señal medida por el <i>encoder</i>	44
4.5 Configuración	45
4.6 Código en Arduino IDE	46
4.7 Pasos para controlar el motor remotamente	46
5. Práctica de laboratorio	48
6. Resultados	53
7. Conclusiones	58
Referencias	60
Apéndices	63

Lista de Figuras

Figura 1. <i>Esquema de conexión para práctica de laboratorio con acceso remoto</i>	16
Figura 2. <i>Arduino UNO (Pomares, Jorge, 2009)</i>	18
Figura 3. <i>Interfaz de programación</i>	20
Figura 4. <i>Excitación independiente y excitación serie, respectivamente</i>	22
Figura 5. <i>Excitación en derivación (shunt) y excitación compuesta (compound), respectivamente</i>	22
Figura 6. <i>Motor DC de imanes permanentes</i>	23
Figura 7. <i>Cálculo de la resistencia (R) del motor</i>	25
Figura 8. <i>Regresión lineal para el cálculo de la resistencia (R)</i>	26
Figura 9. <i>Diseño del esquema propuesto para el control de un motor DC</i>	27
Figura 10. <i>Convertidor DC/DC reductor</i>	28
Figura 11. <i>Esquemático del convertidor DC/DC reductor</i>	29
Figura 12. <i>Diseño circuito fuentes de alimentación.</i>	31
Figura 13. <i>Fuente de alimentación principal implementada.</i>	32
Figura 14. <i>Fuente regulada auxiliar</i>	34
Figura 15. <i>Diseño del filtro de red en Proteus</i>	35
Figura 16. <i>Filtro de red</i>	35
Figura 17. <i>Composición del enconder</i>	36
Figura 18. <i>Módulo para el control del motor DC</i>	38
Figura 19. <i>Interfaz principal Arduino Uno</i>	40
Figura 20. <i>Control propuesto de lazo cerrado</i>	44
Figura 21. <i>Esquema de conexión.</i>	46

Figura 22. <i>Interfaz de Salida de Arduino IDE</i>	47
Figura 23. <i>Circuito equivalente de un motor DC de imanes permanentes en Proteus.</i>	49
Figura 24. <i>Diagrama de bloques de un lazo de realimentación simple.</i>	51
Figura 25. <i>Módulo de laboratorio para la realización de prácticas experimentales con acceso remoto en la temática de control de velocidad de un motor DC.</i>	53
Figura 26. <i>Gráfica velocidad vs tiempo para una referencia de 100 r.p.s.</i>	54
Figura 27. <i>Gráfica velocidad vs tiempo para una referencia de 40 r.p.s.</i>	55
Figura 28. <i>Código para el cambio de velocidad del motor transcurrido X tiempo</i>	56
Figura 29. <i>Gráfica cambio de 20 r.p.s. a 80 r.p.s. transcurridos 10 segundos.</i>	56

Lista de Tablas

Tabla 1. <i>Datos específicos del motor DC.</i>	24
Tabla 2. <i>Resultados de corriente y tensión.</i>	25
Tabla 3. <i>Componentes del convertidor DC/DC reductor.</i>	28
Tabla 4. <i>Implementos fuente de alimentación principal.</i>	32
Tabla 5. <i>Componentes de la fuente regulada auxiliar.</i>	33
Tabla 6. <i>Componentes filtro de red.</i>	35
Tabla 7. <i>Comandos de Arduino Uno.</i>	42

Lista de Apéndices

Apéndice A. Guía para conectar dos ordenadores de forma remota	64
Apéndice B. Cálculos IRFZ48N	67
Apéndice C. Double Speed Measuring – Module with Photoelectric Encoders	70
Apéndice D. Código 1 velocidad de referencia ingresada manualmente	71
Apéndice E. Código 2 cambio de velocidad al pasar X tiempo	73
Apéndice F. Datasheets De ImportanciaPn2222	75

Resumen

Título: Implementación de práctica de laboratorio con acceso remoto utilizando el microcontrolador Arduino Uno para el control de un motor DC.*

Autor: Jhonatan Sneyder Ortiz Santos.**

Palabras clave: Arduino Uno, conexión remota, convertidor DC/DC, microcontrolador, motor DC.

Descripción

En el presente trabajo de grado se realiza la implementación de un módulo de laboratorio para la realización de prácticas experimentales con acceso remoto en la temática de control de velocidad de un motor en corriente continua (DC, por sus siglas en inglés). En la primera parte del documento, se describen e ilustran cada uno de los circuitos y componentes que conforman el módulo de laboratorio empleado para el control de un motor DC de imanes permanentes que varía su velocidad de referencia mediante un convertidor reductor DC/DC. En la segunda parte se presentan tres guías: la primera con una descripción básica de la interfaz de programación Arduino IDE, la segunda con una corta explicación del código de programación, y la última con la enumeración del paso a paso para realizar el control de manera remota. En la tercera parte, se propone una práctica de laboratorio dirigida hacia el control de un motor DC de imanes permanentes, que sirva como apoyo en las materias de control de sistemas eléctricos, electrónica de potencia y máquinas eléctricas I, para estudiantes de la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones (E3T). Finalmente se presentan resultados experimentales del módulo de laboratorio empleado para el control de velocidad del motor DC por medio de dos pruebas: la primera con una velocidad de referencia constante y la segunda ante un cambio tipo escalón en la velocidad de referencia.

Este trabajo de grado está vinculado al proyecto de investigación financiado por la Vicerrectoría de Investigación y Extensión de la UIS titulado "Estrategia pedagógica para el fortalecimiento de competencias en STEM mediante sistemas modulares de laboratorio", código 2825.

* Trabajo de Grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones Director. María Alejandra Mantilla Villalobos. Docente Universidad Industrial De Santander Doctora En Ingeniería

Abstract

Title: Implementation of remote access laboratory practice using the Arduino Uno microcontroller to control a DC motor.*

Author: Jhonatan Sneyder Ortiz Santos.**

Keywords: Arduino Uno, remote connection, DC/DC converter, microcontroller, DC motor.

Description

In the present degree work, the implementation of a laboratory module is carried out to carry out experimental practices with remote access on the subject of speed control of a direct current (DC) motor. In the first part of the document, each of the circuits and components that make up the laboratory module used to control a permanent magnet DC motor that varies its reference speed by means of a DC/DC step-down converter are described and illustrated. In the second part, three guides are presented: the first with a basic description of the Arduino IDE programming interface, the second with a short explanation of the programming code, and the last one with the step-by-step enumeration to carry out the control in a simple way. remote. In the third part, a laboratory practice directed towards the control of a permanent magnet DC motor is proposed, which serves as support in the subjects of control of electrical systems, power electronics and electrical machines I, for students of the School of Electrical, Electronic and Telecommunications Engineering (E3T). Finally, experimental results of the laboratory module used for the speed control of the DC motor are presented through two tests: the first with a constant reference speed and the second with a step change in the reference speed.

This degree work is linked to the research project funded by the UIS Vice-Rector for Research and Extension entitled "Pedagogical strategy for strengthening STEM skills through modular laboratory systems", code 2825.

*

**

Introducción

Los motores de corriente continua (DC, por sus siglas en inglés) son ampliamente utilizados en aplicaciones industriales, comerciales y domésticas, por ejemplo, en las áreas de automatización y control de procesos, en la medicina, y en trenes y tranvías. De esta manera, con este trabajo de grado se busca implementar un módulo de laboratorio que permita la realización de prácticas experimentales en la temática de control de velocidad de un motor de DC, con el fin de proporcionar una herramienta para estudiantes de la carrera de ingeniería eléctrica, que favorezca la apropiación de conocimientos en esta temática y, además, ofrezca la posibilidad de ejecutar pruebas experimentales no solo de manera presencial, sino también de manera remota.

Este trabajo de grado está encaminado para ser un complemento en los laboratorios de las asignaturas de la carrera de ingeniería eléctrica, tales como control de sistemas eléctricos, electrónica de potencia y máquinas eléctricas I, con el fin de orientar la realización de pruebas experimentales con acceso remoto en la temática de control de velocidad de un motor DC de imanes permanente mediante un convertidor reductor DC/DC.

Para la implementación del sistema de control del módulo de laboratorio, se escogió el microcontrolador Arduino Uno debido a que la placa y el software de programación son de bajo costo y fácil acceso para cualquier estudiante de la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones (E3T) interesado en ampliar sus conocimientos adquiridos.

Este documento está organizado de la siguiente manera: en el presente capítulo se muestran los objetivos y la justificación del problema; en el capítulo 2 se describen e ilustran cada uno de los elementos y componentes necesarios en la realización del módulo de laboratorio; en el capítulo 3 se realizaron tres guías básicas con el fin de familiarizar a los estudiantes sobre el

manejo del software de programación y comunicación utilizado en el proyecto; en el capítulo 4 se diseña una práctica orientada hacia el control de velocidad de un motor DC de manera remota; en el capítulo

5 se muestran los resultados experimentales que se obtuvieron del módulo implementado; finalmente se presentan las conclusiones más relevantes del trabajo de grado.

1. Objetivos

1.1 Objetivo general

Desarrollar práctica de laboratorio básica, para que los estudiantes de la E3T puedan controlar remotamente un motor DC por medio del microcontrolador Arduino Uno.

1.2 Objetivos específicos

- Identificar los principales componentes y herramientas que van a ser utilizados en la realización del proyecto.

- Elaborar guías básicas que enseñen a controlar remotamente el motor DC por medio del microcontrolador ARDUINO UNO.

- Diseñar práctica de laboratorio básica orientada al control de un motor DC por medio de un microcontrolador ARDUINO UNO.

2. Justificación y motivación

En los últimos años, el mundo experimentó una situación inesperada que afectó la vida cotidiana de millones de personas debido a la enfermedad provocada por el virus SARS-CoV-2. Grandes empresas se vieron obligadas a declararse en bancarrota, hospitales colapsados por falta de recursos económicos y escasez de personal, familias obligadas a permanecer en sus hogares aumentando sus deudas y disminuyendo sus ingresos y calidad de vida, entre muchas otras consecuencias. Con el pasar del tiempo, las personas tuvieron que adaptarse a una nueva forma de vida, trabajadores cumplir con sus labores desde sus hogares y estudiantes asistir a clases de manera remota, entre otras.

En la actualidad, el uso de herramientas tecnológicas en el aula aporta un conjunto de ventajas para el docente como interactividad, inmediatez, adaptabilidad, y la combinación de los múltiples formatos de enseñanza-aprendizaje, por tal razón se debe adquirir la experiencia de trabajar con tecnología enfrentando así, las dificultades planteadas por la pandemia y de esta manera recuperar, proteger y fortalecer el ritmo de aprendizaje de los estudiantes (Martha Zapata, 2012).

En la actualidad, dentro de los programas de ingeniería eléctrica y electrónica de la UIS, en el área de sistemas de control, no se dispone de prácticas de laboratorio con acceso remoto para el estudio de control de procesos y sistemas dinámicos. La asignatura antes mencionada posee una enorme acogida en los alumnos ya que es de vital importancia en el ámbito profesional. De esta manera, este proyecto integra las tecnologías de la información y las comunicaciones con la formación de pregrado, mediante la implementación de un módulo de laboratorio para la realización de prácticas experimentales con acceso remoto.

Por tal motivo, este proyecto busca implementar una práctica de laboratorio básica, para uso de los estudiantes de la E3T que facilite la visualización de parámetros y oriente la realización de prácticas experimentales en la temática de control de velocidad de un motor DC de manera remota, utilizando un microcontrolador Arduino Uno, que ayudará a aplicar conocimientos adquiridos en el nivel VII de programa de ingeniería de la E3T, más exactamente en la materia de control de sistemas eléctricos (UIS, 2021).

Los alcances de este trabajo se describen a continuación.

- Identificación de los componentes y materiales que van a ser utilizados en el desarrollo del trabajo por medio de una guía con ilustraciones y sus respectivas características.

- Realización de material didáctico (guías básicas) que permitan el aprendizaje de las principales herramientas y comandos disponibles en el software de simulación ARDUINO UNO para su utilización en las experiencias del laboratorio.

- Realización de guía (laboratorio básico) para complementar el componente práctico con ARDUINO UNO que ayude a estudiantes de la E3T.

- Identificación de los parámetros del motor DC por medio de búsqueda en internet a través de bases de datos ya establecidas.

3. Elementos y componentes

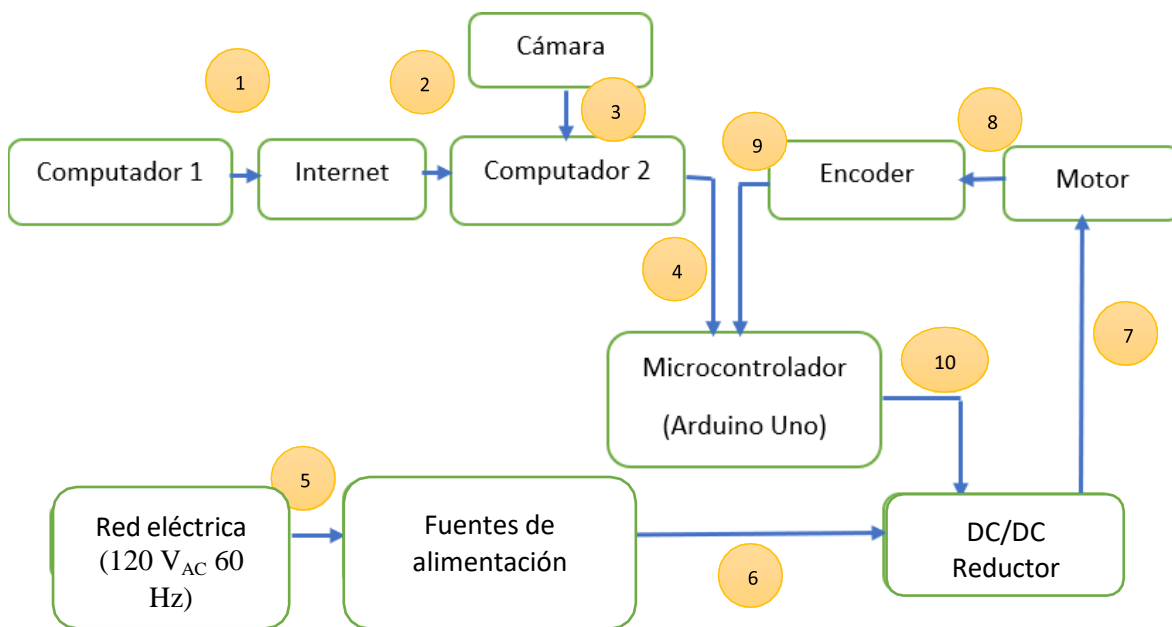
En este capítulo se presentan los elementos y componentes requeridos para la implementación del módulo de laboratorio. Asimismo, se describe el funcionamiento de cada uno de los componentes y se presentan los esquemáticos de los circuitos diseñados realizados en el software Proteus y los circuitos implementados.

3.1 Esquema propuesto

El esquema propuesto para la implementación de la práctica de laboratorio se presenta a continuación.

Figura 1.

Esquema de conexión para práctica de laboratorio con acceso remoto



La ilustración 1 muestra el esquema propuesto para el adecuado desarrollo de la práctica del laboratorio.

Según la secuencia numérica, el esquema propuesto funciona de la siguiente manera:

1. El computador 1 establece una conexión a internet.
2. La conexión a internet sirve para establecer una conexión remota con el computador 2.
3. Una cámara está conectada al computador 2 para visualizar resultados.
4. El computador 2 está conectado a una placa de microcontrolador (Arduino Uno).
5. La red eléctrica (120 VAC 60 Hz) energiza las fuentes de alimentación.
6. Las fuentes de alimentación proporcionan las señales de tensión en DC requeridas para alimentar el convertidor DC/DC reductor.
7. El convertidor DC/DC reductor regula la tensión de salida para controlar la velocidad de un motor DC de imanes permanentes.
8. El motor DC está conectado a un codificador rotatorio (*encoder*) que mide su velocidad.
9. El *encoder* envía una señal de lectura al microcontrolador Arduino Uno.
10. El Arduino Uno calcula el error entre la señal leída respecto a la velocidad de referencia, ejecuta el sistema de control y envía una señal ajustada al convertidor DC/DC reductor, de tal manera que se varía el ancho de pulso de disparo del convertidor, variando de esta manera la velocidad del motor DC de forma controlada.

3.2 Microcontrolador Arduino Uno

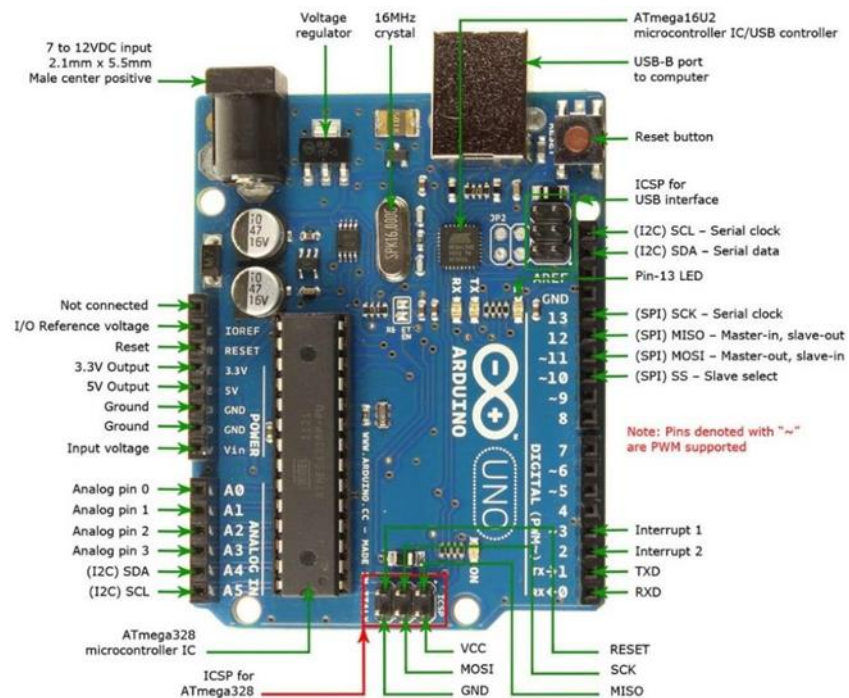
La placa de microcontrolador Arduino Uno es empleada para realizar las siguientes

funciones: recibir los datos de medición de velocidad del motor enviados por el *encoder*, ejecutar el sistema de control y enviar la señal ajustada de control al convertidor DC/DC para regular la velocidad del motor DC. Para lo anterior, a continuación, se describe cómo está compuesta la placa de ArduinoUno con el fin de identificar componentes

La placa tiene 14 pines digitales y 6 pines analógicos. Esta plataforma puede ser programada con el software Arduino IDE (Entorno de desarrollo integrado) a través de un cable USB tipo b. Asimismo, esta puede ser alimentada por el cable USB, por una batería o un adaptador externo de 9 V, aunque acepta tensiones entre 7 y 20 V.

Figura 2.

Arduino UNO (Pomares, Jorge, 2009)



En la ilustración 2 se muestra la placa del microcontrolador Arduino Uno con las

especificaciones que se describen a continuación.

- Microcontrolador: Microchip ATmega328P
- Tensión de funcionamiento: 5 voltios
- Tensión de entrada: 7 a 20 voltios
- Pines de E/S digitales: 14 (de los cuales 6 proporcionan salida PWM)
- Pines de entrada analógica: 6
- Corriente DC por Pin de E/S: 20 mA
- Corriente DC para Pin de 3.3V: 50 mA
- Memoria Flash: 32 KB de los cuales 0.5 KB utilizados por el gestor de arranque
- SRAM: 2 KB
- EEPROM: 1 KB
- Velocidad del reloj: 16 MHz
- Longitud: 68.6mm
- Ancho: 53,4mm
- Peso: 25g

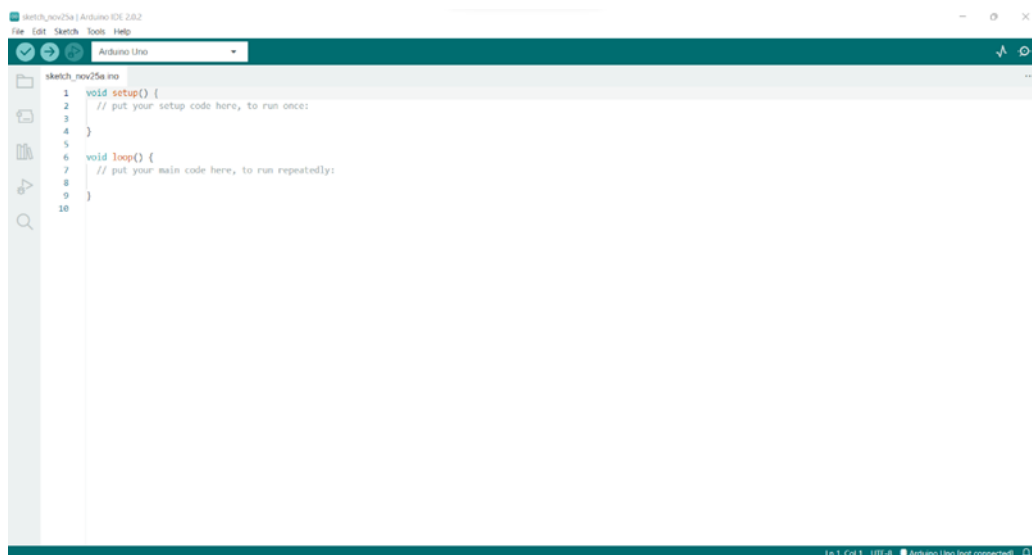
3.3 Software Arduino Uno IDE

El entorno de programación. Arduino Uno IDE (Entorno de Desarrollo Integrado) va a ser la plataforma donde se programará el microcontrolador, se basa en código abierto, es decir cualquiera puede usarlo o programar sobre él. Algunos usos de ARDUINO UNO son: Desarrollo de algoritmos, desarrollo de aplicaciones con interfaces gráficas, gráficos científicos y de ingeniería, gráficos científicos y de ingeniería, cálculo y matemática, análisis, exploración y

visualización de datos, modelamiento, simulación y prototipamiento.

Figura 3.

Interfaz de programación (arduino.cl, s.f.)



La ilustración 3 muestra la interfaz de simulación y programación de Arduino IDE.

3.4 Motor DC

3.4.1 Principios teóricos del motor de corriente continua

El motor de corriente continua es una máquina que convierte energía eléctrica en energía mecánica y gracias a la acción de un campo magnético provoca un movimiento rotatorio. Está constituido por una parte fija llamada estator y una parte móvil llamada rotor (MORA, 2003).

Estas máquinas han sido usadas para el control de velocidad y del par, ya que es que son mucho más flexibles a la hora de implementarse.

Los motores DC pueden construirse con estator bobinado o de imanes permanentes. Asimismo, se encuentran motores sin escobillas, motores paso a paso, y servomotores.

Algunas aplicaciones en el ámbito del consumo de motores de corriente continua pueden estar enfocados en lectores de cd, giradiscos, discos duros, motores para modelismo y aeromodelismo entre otros. Para la parte industrial se pueden encontrar en trenes de laminación reversibles, trenes de laminación en caliente, máquinas trefiladoras, máquinas elevadoras (couplings, s.f.) entre otras.

Para el caso de motores con estator bobinado, existen diferentes configuraciones para conectar los devanados del motor:

3.4.1.1 Motor de excitación serie. En este caso, los devanados del estator y rotor se conectan en serie. Es usado cuando se requiere un arranque alto, se necesita que varíe la velocidad y un par de torsión, mayormente usados en compresores y sistemas de tracción.

3.4.1.2 Motor de excitación en paralelo o motor *shunt*. Para esta configuración, los devanados del estator y rotor se conectan en paralelo. Se utilizan donde se requiera velocidad constante y con condiciones de inicio no severas. Las aplicaciones comunes donde se encuentra el motor *shunt* son en bombas centrifugadas, transportadores, elevadores, ventiladores, sopladores, entre otras.

3.4.1.3 Motor de excitación compuesta o motor *compound*. Se utiliza una combinación

de las configuraciones serie y paralelo. Este motor es utilizado donde se requiere un par de arranque y una velocidad muy constante. Este motor se puede encontrar en laminadores, elevadores, prensas, entre otras.

Figura 4.

Excitación independiente y excitación serie, respectivamente

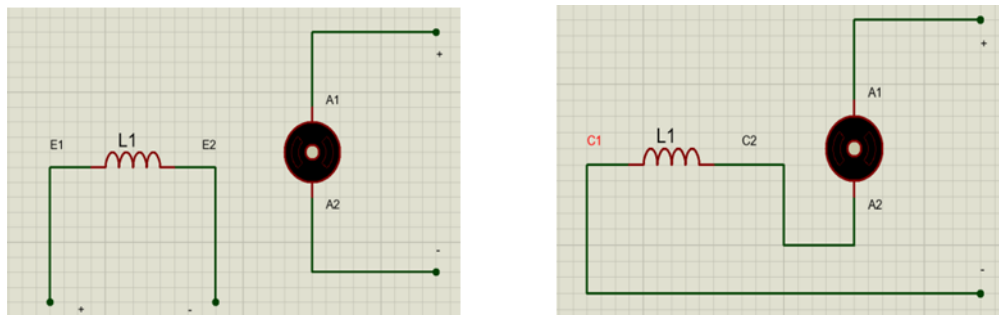
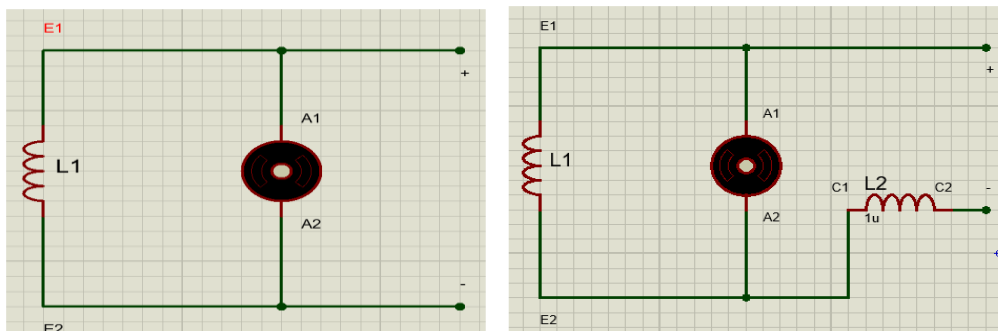


Figura 5. *Excitación en derivación (shunt) y excitación compuesta (compound), respectivamente*



La ilustración 4 muestra a la izquierda la configuración de excitación independiente, y el de la derecha la configuración de excitación en serie para arranque alto.

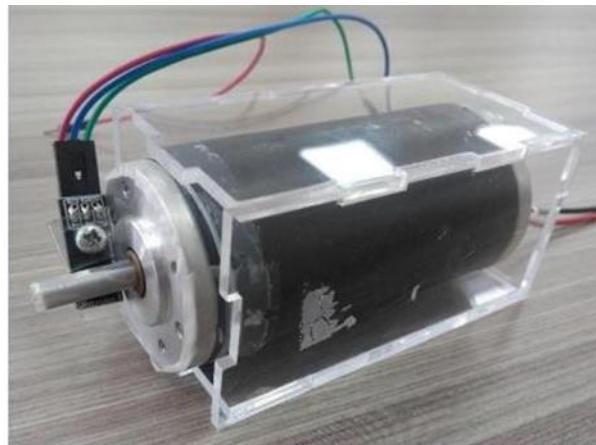
La ilustración 5 muestra a la izquierda la configuración de excitación en derivación o *shunt*, y a la derecha el motor en excitación compuesta.

3.4.2 Elección del motor

El motor DC a ser empleado para la construcción del módulo de laboratorio, fue el utilizado por el trabajo de grado titulado “Diseño e implementación de un módulo de laboratorio para el control de velocidad de un motor de corriente continua de imanes permanentes” (Sergio Andres Vargas Navarro, 2019) de la Universidad Industrial de Santander.

Figura 6.

Motor DC de imanes permanentes



La ilustración 6 muestra una imagen del motor DC de imanes permanentes de tensión nominal 24 VDC que se utilizará en el desarrollo del proyecto.

Este motor de excitación en derivación o de imanes permanentes es adecuado para el control de velocidad, mantiene la velocidad constante prácticamente para cualquier carga con la intención de evitar sobresaltos en el control.

Tabla 1.*Datos específicos del motor DC.*

Modelo	42ZYT04B	Unidades
Tensión Nominal	24	V DC
Velocidad nominalcontinua	4700	r.p.m.
Par nominalcontinuo	100	mN.m
Corriente continua	2.86	A
Par de arranque	479	mN.m
Corriente de arranque	12.79	A
Velocidad sin carga	5900	r.p.m
Corriente sin carga	0.35	A
Corriente de desmagnetización	24	A
Inercia del rotor	154	gcm ²
Peso	550	g
Longitud	90	mm

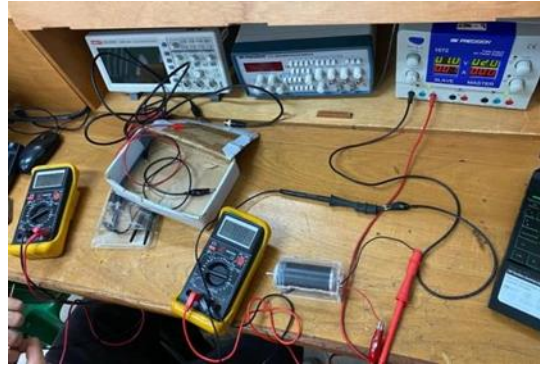
En la tabla 1 se encuentra las especificaciones del motor DC de corriente continua de imanes permanentes.

3.4.3 Parámetros del motor

Con el fin de obtener los parámetros del motor requeridos para el desarrollo del proyecto, resistencia (R) e inductancia (L), se hacen pruebas de laboratorio utilizando una fuente de tensión DC controlada y se obtuvieron los parámetros del motor DC de imanes permanentes para el control (Urueña, 2009).

Figura 7.

Cálculo de la resistencia (R) del motor



En la ilustración 7 se aprecia la imagen de las pruebas experimentales realizadas para el cálculo de la resistencia interna del motor.

Para hallar la resistencia (R) del motor se usó una fuente de tensión y dos multímetros, uno midió la cantidad de tensión y el otro de corriente, se varió la tensión desde 0 V hasta 1 V (de tal forma que el motor no girara), se hizo una gráfica en Excel realizando una regresión lineal y de ahí se tomó el valor de la resistencia.

Tabla 2.

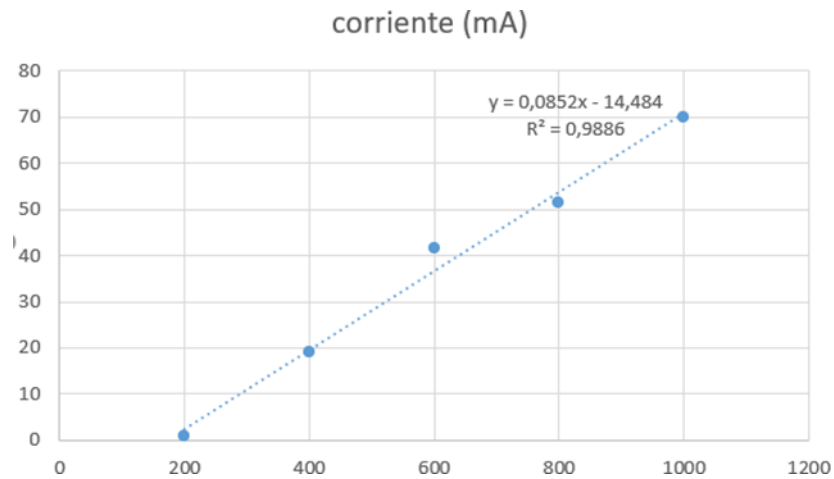
Resultados de corriente y tensión.

corriente (mA)	Voltaje (V)
70	1
51,5	0.8
41,7	0.6
19,1	0.4
0,97	0.2

En la tabla 2 se aprecian los resultados de la toma de los datos obtenidos en el laboratorio, con el fin de realizar el cálculo de la resistencia del motor DC.

Figura 8.

Regresión lineal para el cálculo de la resistencia (R)



En la ilustración 8 se muestra la regresión lineal realizada en Excel con el fin de encontrar el punto de corte (valor de resistencia), dando como resultado 14.48 .

Para hallar el valor de la inductancia (L), se usó un multímetro con función de medición de inductancia, y se tomaron medidas en los bornes del motor. Se estableció como inductancia interna el valor de 1.4 H como el más estable.

3.5 Fuentes de alimentación y convertidor DC/DC reductor

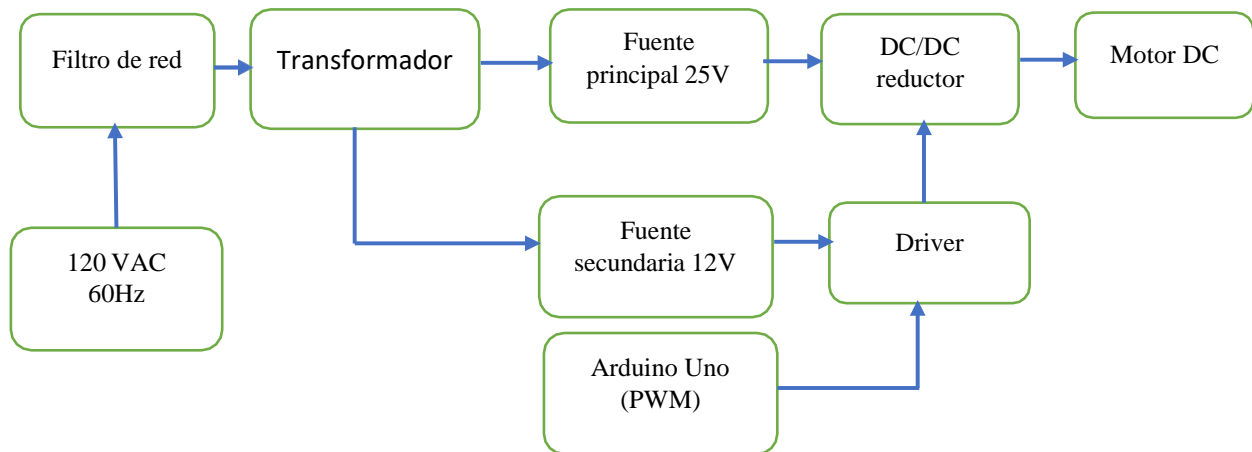
Se realizó el diseño y ensamble de un módulo de potencia que consta de una fuente de alimentación lineal y un convertidor reductor para alimentar el motor de imanes permanentes y regular su velocidad. El sistema de control y la generación de la señal modulada por ancho de pulso (PWM por sus siglas en inglés) para el control del convertidor DC/DC, se realiza mediante la placa Arduino Uno.

En la ilustración 9 se observa el diseño de un esquema compuesto de las siguientes

etapas: filtro de red a la entrada para evitar sobresaltos en corriente, transformador de potencia para adecuación y aislamiento de señales, fuente de alimentación principal de 25 V, fuente de alimentación secundaria de 12 V, sistema de adecuación y aislamiento para las señales de control (*driver*), y convertidor DC/DC reductor. A continuación, se describen cada una de estas etapas.

Figura 9.

Diseño del esquema propuesto para el control de un motor DC



3.5.2 Convertidor DC-DC reductor

Este convertidor es utilizado para regular la tensión de alimentación del motor y así poder ejercer el control en su velocidad. Mediante la variación del ciclo útil del convertidor DC/DC, se obtiene una salida de tensión controlada según se requiera.

El diseño del convertidor DC/DC reductor, se realizó con base en las ecuaciones que modelan su funcionamiento presentadas en el libro de Electrónica de Potencia de Daniel W. Hart (Hart, 2001) y en la guía básica *Buck converter Design* (Ejury, 2013).

Los elementos que se usaron para la implementación del convertidor reductor se pueden

visualizaren la Tabla 3, su circuito esquemático y componentes se encuentran en la ilustración 10.

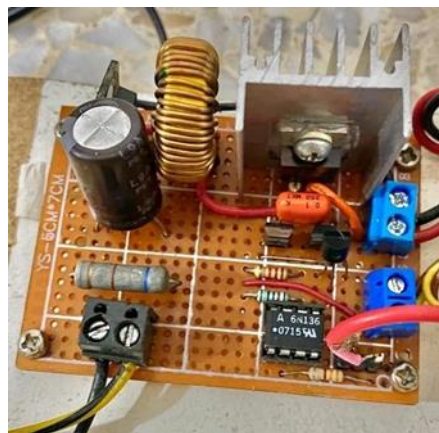
Tabla 3.

Componentes del convertidor DC/DC reductor.

COMPONENTES
MOSFET IRFZ48N
OCTOACOPLADOR 6N136
TRANSISTOR SB2040CT
RESISTENCIAS 10 k Ω , 1.2 k Ω , 680 Ω
CAPACITOR 1000 μ F
TRANSISTOR PN2222A
INDUCTOR 150 μ H
TRANSISTOR KTC1027
TRANSISTOR KTA1023

Figura 10.

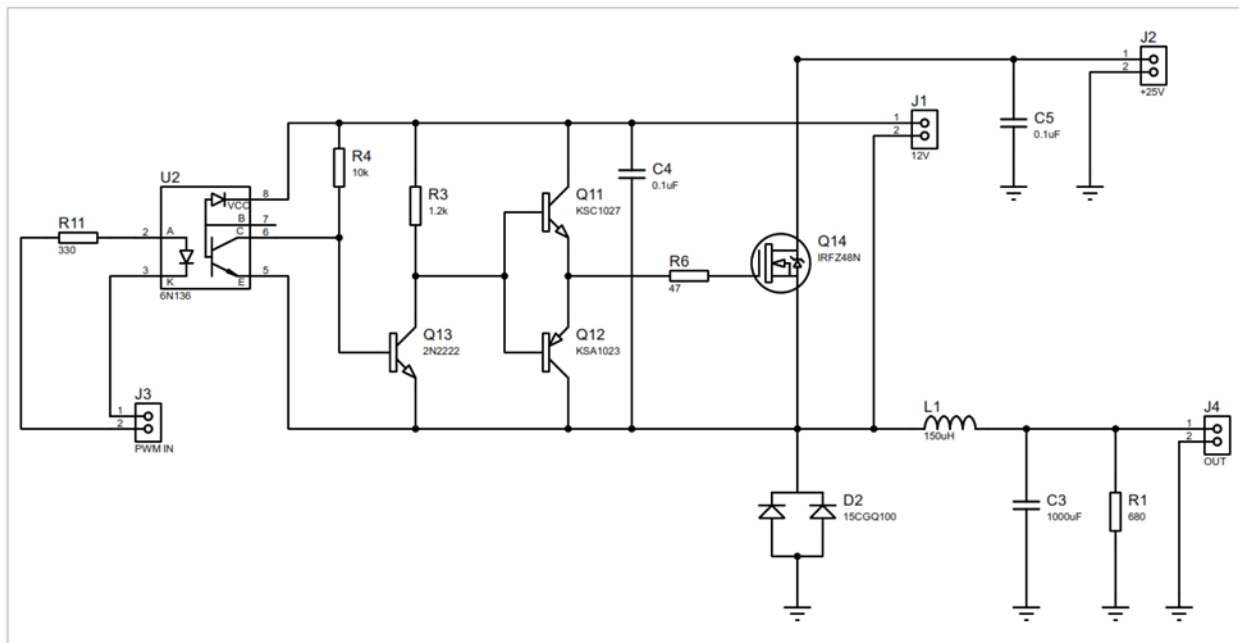
Convertidor DC/DC reductor



En la ilustración 10 se observa el convertidor DC/DC reductor diseñado e implementado.

Figura 11.

Esquemático del convertidor DC/DC reductor



En la ilustración 11 se muestra el diseño del convertidor DC/DC reductor que se implementó juntocon el circuito de adecuación y aislamiento de señales (*driver*).

3.5.3 Circuito de adecuación y aislamiento de señales (*driver*).

El MOSFET de referencia IRFZ48N es el interruptor electrónico empleado en el convertidor DC/DC reductor. Este dispositivo es conmutado continuamente, se puede considerar en esencia como una fuente de corriente controlada por tensión, debido a la alta impedancia de entrada de la compuerta (*gate*), una vez opera en las regiones de corte o saturación, no se requiere

corriente (prácticamente corrientes del orden nA y pA) para mantener ese estado. El problema ocurre en el cambio de estado de corte-saturación, apagado-encendido, debido a que existen ciertas capacitancias entre sus terminales, especialmente entre la compuerta y la fuente (*source*), y como la señal que se aplica cambia muy rápido entre un estado bajo y alto, un capacitor se impone a los cambios de tensión abruptos. Teniendo en cuenta lo anterior, se propone un circuito adicional que permita además de proveer la suficiente corriente instantánea de carga y descarga para la compuerta, también alcance su tensión de activación plena o saturación. El circuito propuesto consta de dos transistores (KTC1027 y KTA1023) con capacidad relativamente alta de corriente (tipo bipolar y colocados en *push -pull*) que permiten aplicar y evacuar las corrientes de la puerta rápidamente, estos transistores son activados por otro transistor (PN2222) configurado como inversor, para evitar circuitos complejos cambiadores de nivel. Al estar la fuente de 12V, de forma flotante se emplea un optoacoplador de alta velocidad (6N136) teniendo dos ventajas claras: evita los complejos circuitos cambiadores de nivel y provee aislamiento galvánico en la entrada de excitación que la conforma el Arduino Uno. El cálculo de corrientes de activación del MOSFET y corriente en los transistores se puede observar en apéndice B.

3.5.4 Transformador de potencia

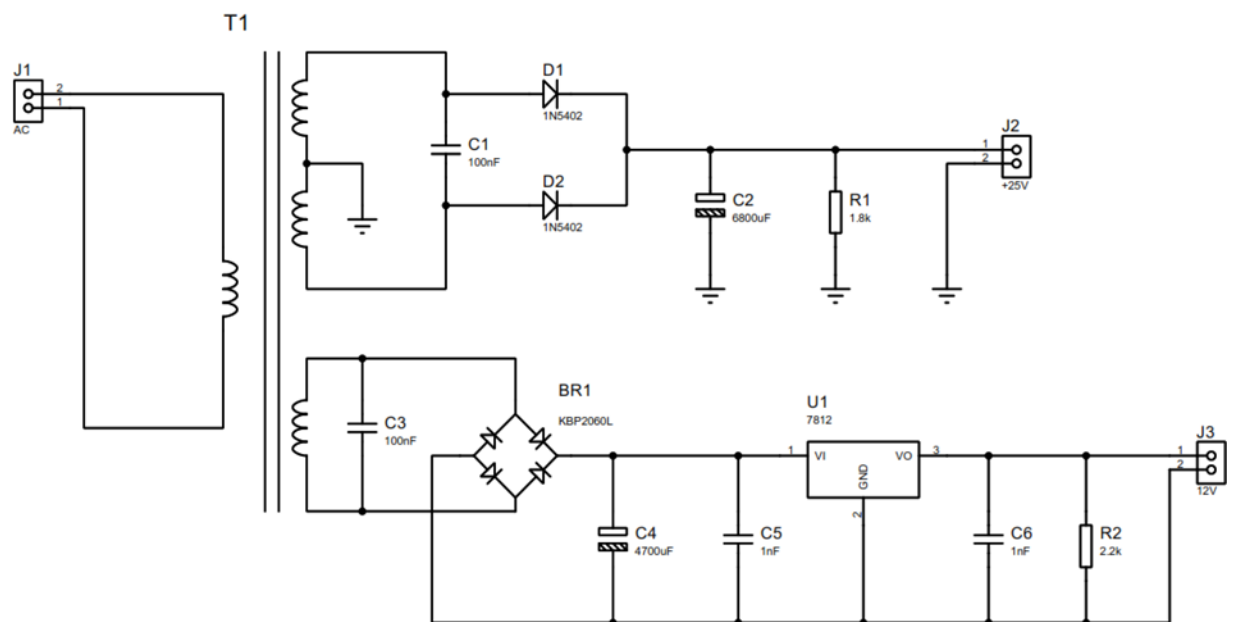
En este caso se usa un transformador de especificaciones 60 Hz, 70 VA, 120 Vrms/ (18-0-18,12 Vrms), ver ilustración 13. El secundario posee dos salidas, una salida con derivación central de 18-0-18 Vrms, la cual proporciona la tensión para la fuente principal que va a alimentar al motor. El otro devanado del secundario es de 12 V y se destina para el circuito de activación (*driver*) del MOSFET.

3.5.5 Fuentes de alimentación

En la ilustración 12 se aprecia el diseño de los circuitos para la implementación de las fuentes de alimentación. En general se requieren dos fuentes de alimentación: una principal de 25 V para alimentar el convertidor DC/DC reductor, y una fuente de alimentación auxiliar de 12 V para alimentar el sistema de adecuación y aislamiento para las señales de control (*driver*) del MOSFET.

Figura 12.

Diseño circuito fuentes de alimentación.



La fuente de alimentación principal se compone de un rectificador de onda completa con derivación central y filtro capacitivo de salida. Asimismo, se conecta una resistencia en paralelo al condensador para que este se descargue cuando el circuito se desconecte.

Tabla 4.

Implementos fuente de alimentación principal.

COMPONENTES
DIODO 1N5404
CAPACITOR 0,1 μF
CAPACITOR 6800 μF
RESISTENCIA 1,8 kΩ

En la Tabla 4 se presentan los componentes que se usaron para implementar la fuente de alimentación principal

Figura 13.

Fuente de alimentación principal implementada.



En la ilustración 13 se observa el circuito implementado correspondiente a la fuente de

alimentación principal.

Con respecto a la fuente regulada auxiliar, esta es empleada para alimentar el circuito de excitación de la puerta del MOSFET. Mediante esta fuente se obtiene una tensión regulada a 12 VDC, siendo esta la tensión requerida para que el MOSFET llegue a saturación.

Para la implementación de esta fuente, se usa un puente rectificador ya encapsulado y se dispone de un capacitor electrolítico a la salida del rectificador para el filtrado de 4700 μ F. Se emplea el regulador lineal 7812, el cual entrega una tensión de 12V a 1A.

Tabla 5.

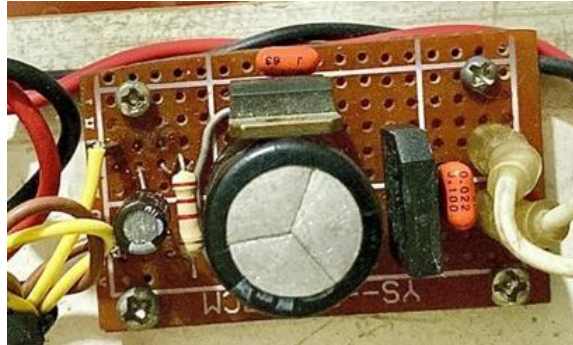
Componentes de la fuente regulada auxiliar.

COMPONENTES
INTEGRADO L7812CT
CAPACITOR 4700 μF
CAPACITOR 33 μF
CAPACITOR 0,1 μF
RESISTENCIA 2,2 kΩ
PUENTE DE DIODOS KBP2010

La Tabla 5 muestra los componentes que se usaron en la implementación de la fuente regulada auxiliar.

Figura 14.

Fuente regulada auxiliar

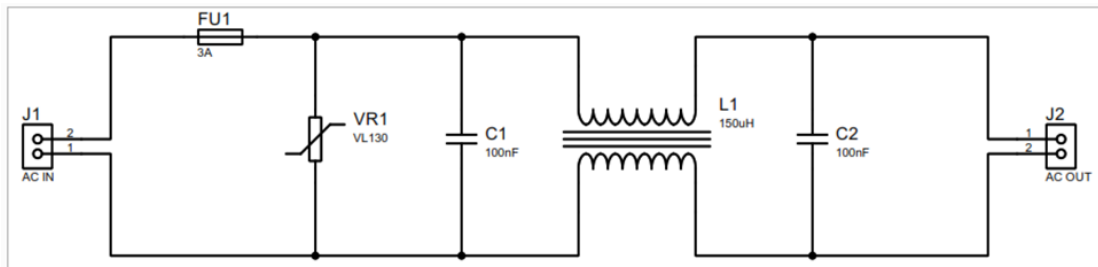


En la ilustración 14 se puede observar la fuente regulada auxiliar implementada.

Para el diseño de estas fuentes se tuvo en cuenta el libro *power supply cookbook* (Brown, 2001).

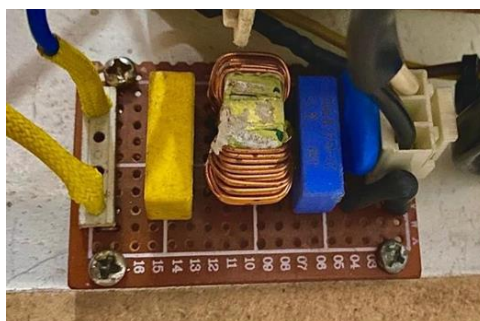
3.5.6 Filtro de red

Conformado por dos capacitores, una bobina y un varistor, tal como se observa en la ilustración 15. El primer capacitor actúa como un filtro pasa bajas de primer orden ante componentes de alta frecuencia provenientes de la red. La bobina presenta el devanado en un núcleo toroidal de ferrita, al devanar fase-neutro en el mismo núcleo, pero de la forma independiente, cumple la función de atenuar un ruido en modo común de la red. El segundo capacitor evita que las componentes de alta frecuencia producidas por los rectificadores aguas abajo del transformador, lleguen con amplitud considerable a la red. El varistor es el encargado de suprimir los picos de tensión de la red protegiendo en este caso el devanado primario del transformador.

Figura 15.*Diseño del filtro de red en Proteus***Tabla 6.***Componentes filtro de red.*

COMPONENTES
CAPACITORES 0.1 μF
INDUCTANCIA 100 μH
VARISTOR 471K014

En la Tabla 6 se plasman los componentes usados en el filtro de red con sus especificaciones.

Figura 16.*Filtro de red*

La ilustración 16 muestra el filtro de red implementado en la entrada principal de tensión del módulo.

Para el diseño de este circuito, se utilizó el libro Principio de Electrónica (Albert Paul Malvino, 2007).

3.6 Módulo codificador rotatorio (*encoder*)

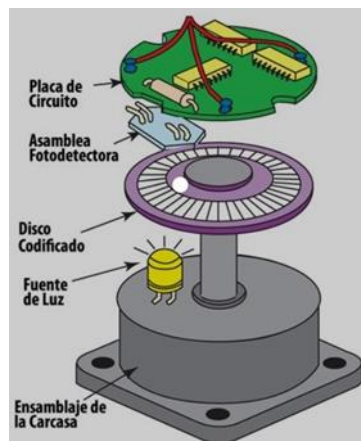
Este dispositivo actúa como sensor electromecánico a través de una placa codificada y una serie de señales eléctricas, y permite determinar la velocidad del motor.

El dispositivo que se usó es un módulo *encoder* de referencia HC-020K con salida digital de 0 y 5V (ver apéndice C), el cual permite convertir la posición angular del eje en un código digital.

Cuando la celda foto detectora del *encoder* deja pasar la luz del led (pasa por una ranura), la salida de este corresponde a 5 V, de lo contrario el valor es 0 V.

Figura 17.

Composición del encoder (Company, s.f.)



En la ilustración 17 se observa la estructura de un *encoder*.

3.7 Extensión Google desktop

La extensión Google desktop es la encargada de interconectar remotamente los dos computadores, con el fin de permitir el acceso remoto y realizar pruebas experimentales de manera remota. Una descripción sobre el proceso de descarga y uso de esta extensión se encuentra en el apéndice A.

3.8 Computadores

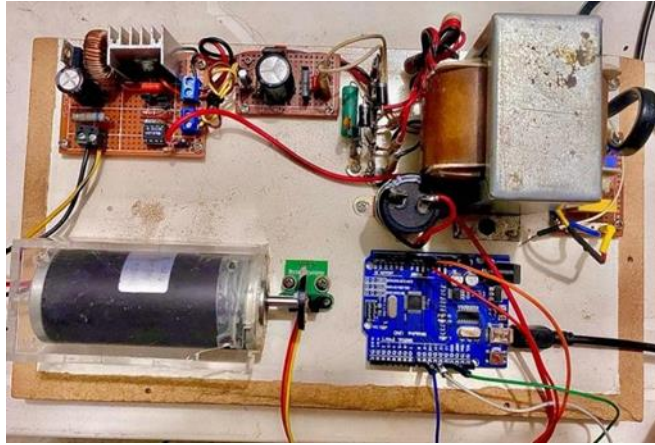
Los computadores van a ser los equipos requeridos para programar la placa Arduino UNO y a realizar la conexión remota para el laboratorio virtual. Cabe recalcar que los computadores a usar deben estar conectados a una red de internet, preferiblemente WIFI por la velocidad de transferencia de datos.

3.9 Módulo de potencia y control implementado

La ilustración 18 muestra el módulo terminado para el control del motor DC de imanes permanente.

Figura 18.

Módulo para el control del motor DC



Este módulo está conformado por varios circuitos, cuyo funcionamiento por aparte es fundamental para el desarrollo del sistema de control. Algunas hojas de datos de importancia se pueden visualizar en el apéndice F.

4. Guías básicas para el control de un motor DC con acceso remoto

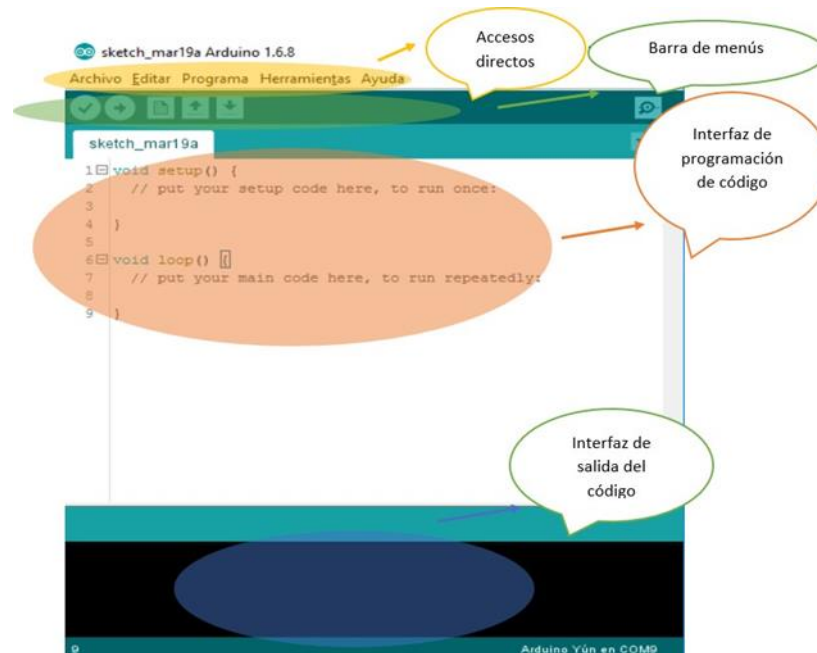
En este capítulo se encuentra una breve descripción de la interfaz de programación del microcontrolador. También se describen algunos comandos usados en el código para implementar el sistema de control, al igual que se expone el esquema de control propuesto y por último se presenta una guía básica para controlar el motor de forma remota.

4.1 Interfaz y comandos de Arduino Uno

Para realizar la implementación del control propuesto, es necesario saber cómo funciona la interfaz de programación de Arduino Uno y al mismo tiempo conocer los principales comandos que van a ser utilizados en el desarrollo del proyecto.

4.1.1 Interfaz de programación de Arduino Uno

La interfaz de programación de ARDUINO UNO es bastante sencilla, a continuación, se presenta una corta explicación sobre esta.

Figura 19.*Interfaz principal Arduino Uno*

En la ilustración 19 se observa una imagen de la interfaz principal de programación de ArduinoUno.

4.1.1.1 Accesos directos. La función de la barra de accesos directos es la de permitir abrir, guardar y editar códigos, entre otras funciones, con el fin de facilitar las tareas de los programadores. También se pueden encontrar ejemplos de código, los cuales explican explícitamente paso a paso programaciones ya establecidas.

4.1.1.2 Barra de menús. La barra de menú es la que permite subir y compilar el código programado en la placa de Arduino Uno con el fin de ejecutar el programa diseñado en el microcontrolador.

4.1.1.3 Interfaz de programación de código. En este espacio es donde se programa el código a ser ejecutado en la placa de Arduino Uno, tiene dos fases: el *void setup* y el *void loop*.

Void setup: Esta es la parte de la interfaz de programación donde se definen las variables a utilizar en el desarrollo del código, las variables pueden ser de composición entera o decimal según se requiera.

Void loop: En esta parte es donde se hace la mecánica del código por medio de comandos ya establecidos, el fin es desarrollar una serie de pasos lógicos para llegar a un fin de programación específica. En el numeral 3.1.2 se explican algunos de los comandos usados en el desarrollo de este proyecto.

4.1.1.4 Interfaz de salida del código. En esta parte de la interfaz es donde se obtiene el resultado de la compilación del código, indica si hay errores o si el Arduino no está conectado correctamente, entre otra información. Cada uno de estos datos se pueden usar a la hora de la mejora del código, inclusive para cambio de puerto COMO hacer un análisis del programa.

4.2 Comandos

Arduino IDE es un entorno con una construcción de código sencillo, basta con conocer algunos comandos dependiendo de lo que se requiera realizar para poder usarlo. En este numeral se van a describir los comandos usados para el desarrollo del proyecto.

Tabla 7.*Comandos de Arduino Uno.*

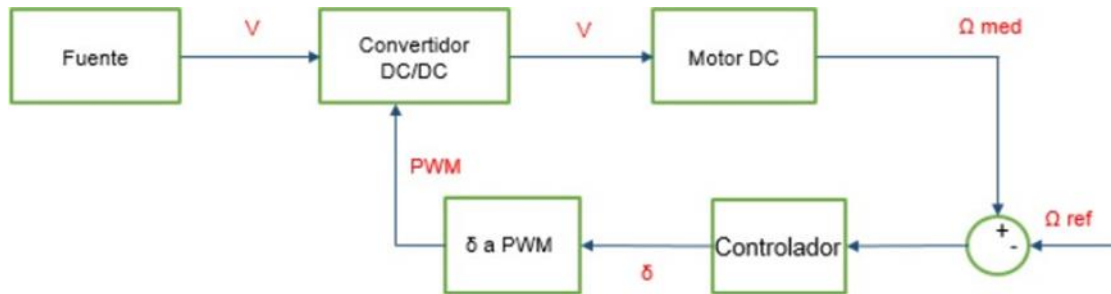
COMANDO	FUNCIONAMIENTO	SALIDA	COMANDO	FUNCIONAMIENTO	SALIDA
int	Definir una variable de tipo entero de hasta 16bits	Rango 32,767 a -32,768	pinMode (pin, INPUT)	Establece el pin indicado como entrada	Lee el pin con una resolución de 10 bits
unsigned long	Definir una variable de tipo entero sin signos de 32 bits	Se obtienen enteros de hasta 10 dígitos	void loop ()	Ejecuta el programa en el espacio específico donde se programa el código	Ejecuta el programa en el Arduino
float	Definir una variable de tipo flotante	Números decimales o con punto flotante	While	Repite las sentencias anidadas mientras la expresión <i>while</i> se evalúe como VERDADERA	Ejecuta el ciclo mientras se cumpla el condicional
double	Definir una variable de tipo flotante con doble precisión	Números decimales o con punto flotante con más precisión	If	Encargado de evaluar si la sentencia es verdadera	Ejecuta un condicional mientras se cumpla
void setup ()	Lugar específico donde se nombran las variables	Configuración de pines de entrada y salida	digitalRead()	Permite leer valores lógicos digitales de un pin de entrada de la tarjeta Arduino	Lee los valores digitales de una señal externa
Serial.begin()	Abre un Puerto serie y especifica la velocidad de transmisión.	Permite la comunicación entre el Arduino, el	Serial.print	Es usualmente útil para observar los datos producidos en el programa	Imprime datos en otros dispositivos conectados al puerto serial.

COMANDO	FUNCIONAMIENTO	SALIDA	COMANDO	FUNCIONAMIENTO	SALIDA
		computador u otros dispositivos.			
pinMode (pin, OUTPUT)	Establece el pin indicado como salida.	Configura un pin como salida.	analogWrite	Esta función se usa para generar una señal analógica o de PWM en uno de los siguientes pines permitidos (3, 5, 6, 9, 10, 11, 13)	Genera señal PWM de 490 Hz y 980 Hz.
Millis ()	Permite obtener la cantidad de milisegundos que han pasado desde que comenzó la ejecución del código	Tiempo actual del microcontrolador or			

En la Tabla 7 se da explicación de algunos de los comandos que se usaron para escribir el código de control.

4.3 Control propuesto

Para el control de la velocidad del motor de DC se implementa un sistema retroalimentado o delazo cerrado utilizando un controlador proporcional integral (PI).

Figura 20.*Control propuesto de lazo cerrado*

En la ilustración 20 se presenta un diagrama de bloques del sistema de control en lazo cerrado a desarrollar.

Para determinar los valores de las constantes proporcional (K_p) e integral (K_i) del controlador, se usó la Herramienta PIDTOOL que se encuentra en el programa Matlab, obteniendo los siguientes resultados para una respuesta subamortiguada:

$$K_p=0.058 \quad K_i=0.0001$$

4.4 Cálculo de la velocidad a partir de la señal medida por el *encoder*

El cálculo de la velocidad del motor a partir de las señales enviadas por el *encoder* es sencillo. Se conoce que la rueda del *encoder* tiene 20 ranuras que equivalen a una vuelta. El *encoder* mide cuantas veces la celda fotorreceptora capta la señal del led (pasos por las ranuras) en 10 milisegundos. Si el *encoder* marca X veces la señal alta en este tiempo, se sabe que:

$$\# \text{ de vueltas en } 10\text{ms} = \frac{\# \text{ de vueltas en } 10\text{ms}}{20(\text{ranuras del encoder})}$$

$$\# \text{ de vueltas en } 10\text{ms} = \frac{2}{20} \text{ Vueltas}$$

Lo anterior se divide por el tiempo transcurrido y se convierte a r.p.s. (revoluciones por segundo), obteniendo la velocidad a la que va el motor.

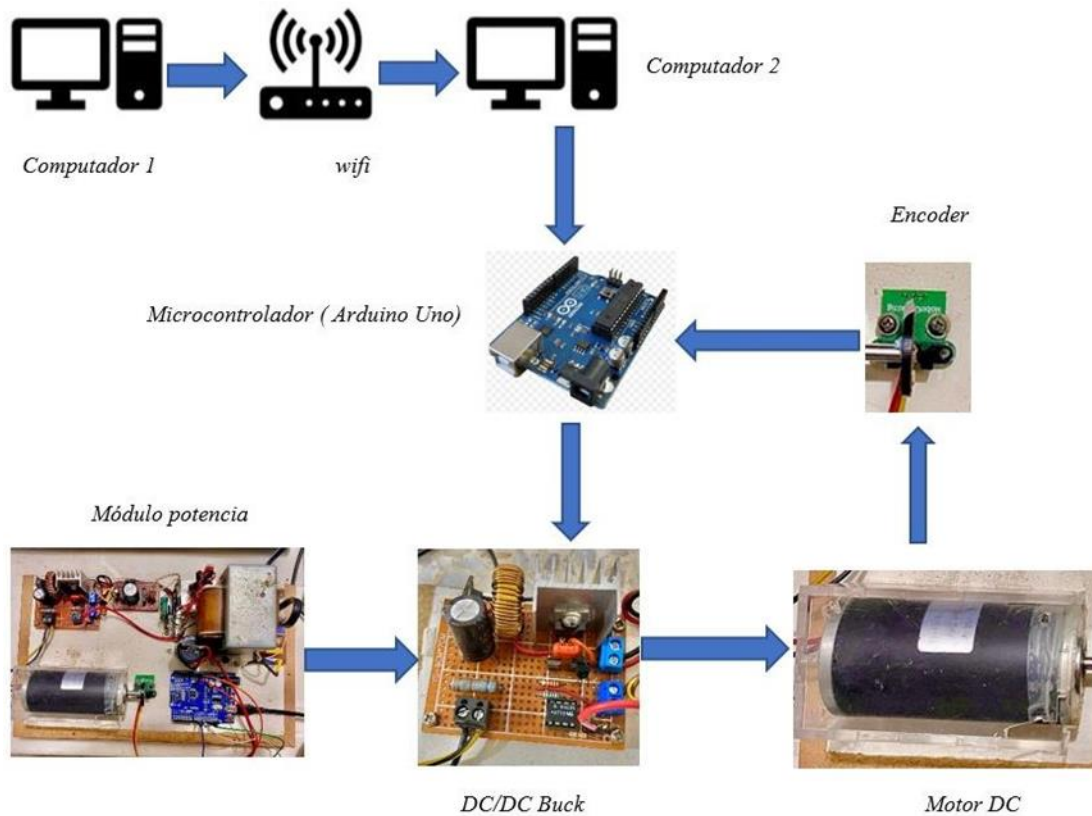
$$\text{velocidad en rps} = \frac{\frac{x}{20} \text{ vueltas}}{0.01 \text{ (segundos que son 10 mts)}}$$

$$\text{Velocidad} = 5 * X \text{ r. p. s}$$

En el código implementado, en 10 milisegundos se conoce la velocidad del motor, después actúa el control en lazo cerrado y se realimenta con el error calculado, siguiendo la velocidad de referencia ingresada al código de Arduino Uno.

4.5 Configuración

En la ilustración 21 se observa el esquema de conexión requerido para la realización de las prácticas de laboratorio. En general, se emplearon: dos computadores, un microcontrolador Arduino Uno, el módulo de potencia implementado, el *encoder*, y el motor DC.

Figura 21.*Esquema de conexión.*

4.6 Código en Arduino IDE

El código implementado en Arduino IDE se presenta en el apéndice D.

4.7 Pasos para controlar el motor remotamente

En esta parte, se presentan los pasos para realizar el proceso de control del motor DC remotamente, por medio del microcontrolador Arduino Uno.

1. Conectar el computador A (cualquiera con acceso a internet) al computador B (el que

se encuentra en el laboratorio) remotamente, esta explicación la puede encontrar en el apéndice

A.

2. Ingresar a la cámara del computador B para visualizar resultados.
3. Abrir el programa de Arduino Uno en el computador B.
4. Abrir el código con nombre “CONTROLMOTOR”.
5. Variar la velocidad remotamente desde el computador A, cambiando el código en el computador B (cada línea del código está bien explicada continuamente).
6. Visualizar si la velocidad del motor se está ajustando de forma controlada en la interfaz desalida Arduino IDE explicada en el capítulo 3.1.1.4.

La ilustración 22 muestra la interfaz de salida de Arduino IDE.

Figura 22.

Interfaz de Salida de Arduino IDE

The screenshot displays the Arduino IDE environment. On the left, a portion of the code is visible, including lines 103, 104, 105, and 106. Line 103 is highlighted with a yellow circle. The code is as follows:

```

103  if (Salida < 0) Salida = 0;           // Si se h
104
105  //Salida = 253 - Salida;             // Negam
106  analogWrite(SalidaSignal, Salida);  // Sacamos

```

Below the code editor, the 'Output' and 'Serial Monitor' tabs are visible. The 'Serial Monitor' tab is active and shows a list of data points. The first two lines of the output are highlighted with a yellow circle. The data points are as follows:

Timestamp	Value 1	Value 2	Value 3
18:47:08.038	80.00	70.00	6.58
18:47:08.038	80.00	80.00	6.00
18:47:08.038	80.00	85.00	5.71
18:47:08.071	80.00	80.00	6.00
18:47:08.071	80.00	75.00	6.29
18:47:08.071	80.00	80.00	6.00
18:47:08.151	80.00	80.00	6.00
18:47:08.151	80.00	80.00	6.00
18:47:08.151	80.00	75.00	6.30
18:47:08.156	80.00	80.00	6.01
18:47:08.156	80.00	85.00	5.71
18:47:08.156	80.00	80.00	6.00
18:47:08.170	80.00	75.00	6.30
18:47:08.170	80.00	80.00	6.01
18:47:08.170	80.00	85.00	5.71

At the bottom of the Serial Monitor window, the following status messages are displayed:

```

El Sketch usa 4960 bytes (15%) del espacio de almacenami
Las variables Globales usan 250 bytes (12%) de la memori

```

7. Seguidamente si se ve que el motor está variando la velocidad según los valores que se ingresan al código, significa que el control se está desarrollando en su totalidad.

5. Práctica de laboratorio

En este apartado se realiza la guía de laboratorio correspondiente al control de velocidad de un motor de corriente continua, especificando las actividades y pasos a realizar en la práctica, tomando como apoyo las asignaturas de máquinas eléctricas I y control de sistemas eléctricos.

Objetivos

-Realizar el control de velocidad de un motor de corriente continua de imanes permanentes de forma remota.

-Realizar un análisis de lo comprendido en la práctica de laboratorio

Actividades previas

-Realizar la lectura del documento para tener claros conceptos previos a la realización de la práctica.

-Leer sobre la interfaz de programación Arduino IDE y librerías a utilizar.

-Revisar la información sobre el módulo de potencia requerido para la implementación de la práctica y el sistema de control a ejecutar.

Equipo necesario

- Equipos de cómputo con software Arduino Uno y conexión a internet.

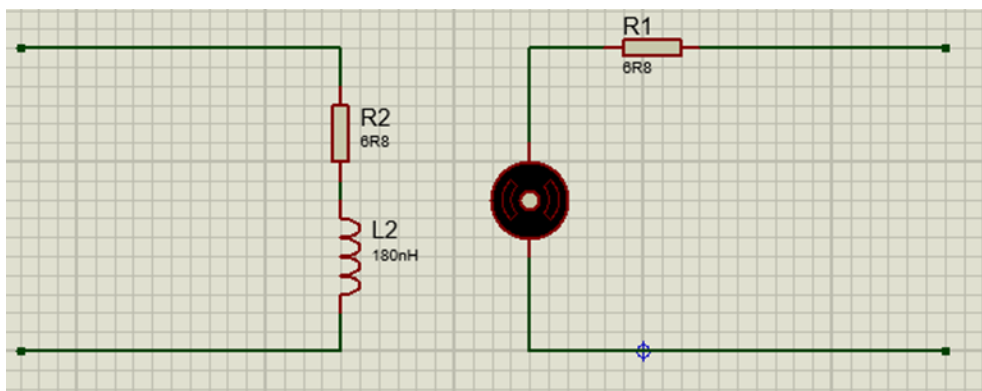
- Motor DC de imanes permanentes de 24V
- Microcontrolador Arduino uno
- Sensor de velocidad (*encoder*)
- Circuito de potencia y de control
- Cámara

Marco teórico

Los motores de corriente continua han cobrado una gran importancia a nivel educativo e industrial dadas las diferentes aplicaciones que los utilizan, como en los trenes, camiones, aviones, sistemas de automatización y control de procesos, entre otros. Una aplicación importante de los motores de corriente continua es cuando se requieren variaciones amplias de velocidad.

Figura 23.

Circuito equivalente de un motor DC de imanes permanentes en Proteus.



La ilustración 23 hace referencia al circuito equivalente de un motor DC de imanes permanentes.

Los motores DC son accionados desde una fuente de corriente continua, de esta manera se puede simplificar el análisis y la comparación entre diferentes tipos de motores.

Entre estos tenemos cuatro tipos de configuraciones de motores DC:

1. Motor de DC de excitación separada
2. Motor de DC en derivación (configuración seleccionada para la implementación de la práctica de laboratorio).
3. Motor de DC en serie.
4. Motor de DC compuesto.

Control de velocidad en los motores DC en derivación

Para lograr controlar la velocidad de un motor DC generalmente se utilizan tres métodos, dos de ellos comunes y uno menos usual. Estos métodos son:

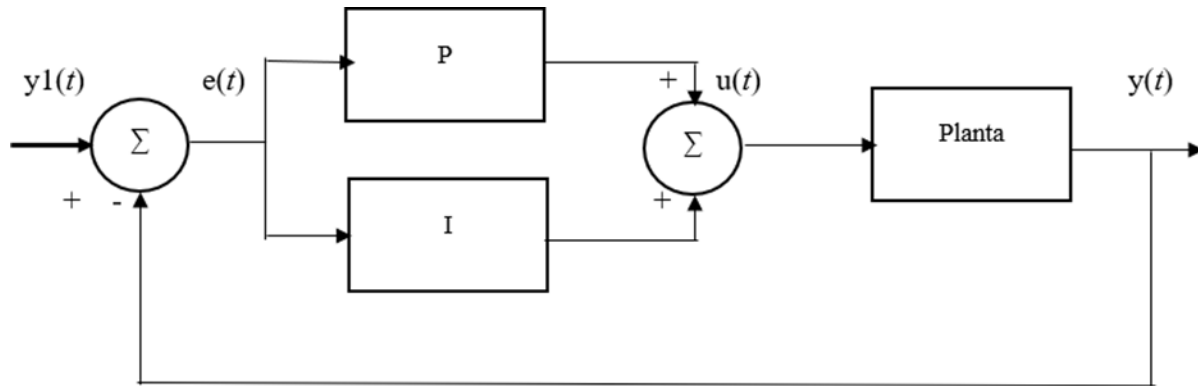
1. Ajustar la resistencia de campo R_F , lo que modifica el flujo del campo.
2. Ajustar la tensión en los terminales que se aplica al inducido.
3. Insertar una resistencia en serie con el circuito del inducido, esto va a ajustar la tensión en los terminales aplicado al inducido.

Control Proporcional-Integral

Un controlador PI (proporcional-integral) es un elemento de transferencia de un sistema de control de bucle cerrado que comprende componentes de elemento tanto proporcional (P) como integral (I).

Figura 24.

Diagrama de bloques de un lazo de realimentación simple.



La ilustración 24 muestra un diagrama de bloques de un lazo cerrado con realimentación simple. Los parámetros del control PI son los siguientes:

- La acción de control proporcional: como su nombre lo indica es proporcional a la señal de error. Esta multiplica la señal de error por una constante K_p , esta constante permite disminuir el error del sistema en régimen permanente y aumenta la velocidad en la que responde el sistema.

- La acción de control integral: calcula la integral de error, expresando como una suma la señal de error y acumulándola. Para esta acción se emplea la constante integral K_i , la cual sube levemente la velocidad del sistema y disminuye el error en régimen permanente.

Actividades del laboratorio

1. Conectar los computadores remotamente (ver apéndice A).
2. Sincronizar los computadores con el sistema de potencia y control por medio del interfaz de Arduino Uno.
3. Obtenga los datos resultantes y realice graficas de velocidad vs tiempo para los dos

casos.

4. Realice de nuevo el paso 3 cambiando la velocidad del motor.
5. Obtenga los datos en los dos casos y realice gráfica comparativa.
6. Realice un cambio en el código para que la velocidad del motor aumente en 40 r.p.s.transcurridos 10 segundos. Presente los datos y gráficas obtenidas.
7. Explique los cambios observados en los diferentes casos realizados en la práctica del laboratorio.
8. Presente conclusiones de la práctica realizada.

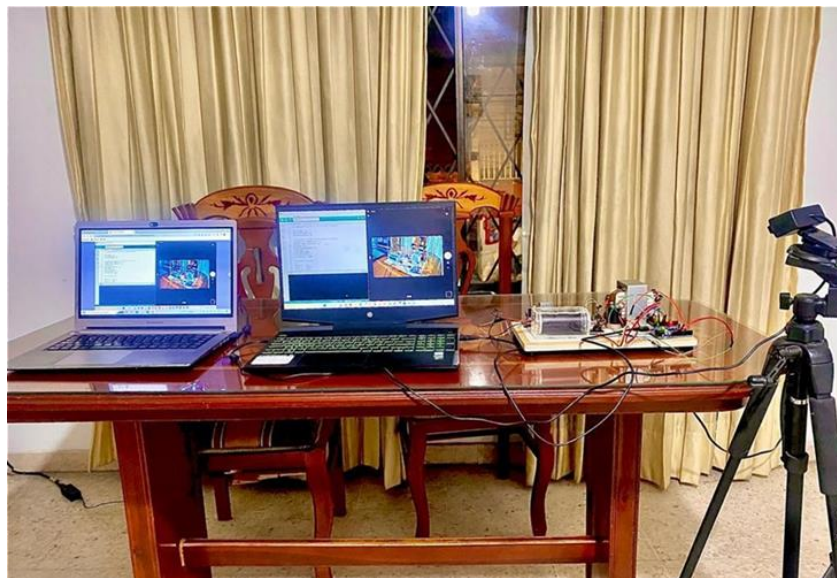
6. Resultados

El resultado de este trabajo de grado es la realización de un módulo de laboratorio con su respectiva práctica en la temática de control de velocidad de motores de DC. Este módulo es funcional y sirve para estudiantes de las carreras de ingeniería eléctrica o electrónica que quieran ampliar los conocimientos en las áreas de control, electrónica de potencia y máquinas eléctricas.

A continuación, se muestra el módulo finalizado junto con los computadores para el acceso remoto y la cámara que permite la visualización de la parte física.

Figura 25.

Módulo de laboratorio para la realización de prácticas experimentales con acceso remoto en la temática de control de velocidad de un motor DC.



En la ilustración 25 se observa el módulo de laboratorio junto con la parte remota para la realización de la práctica requerida.

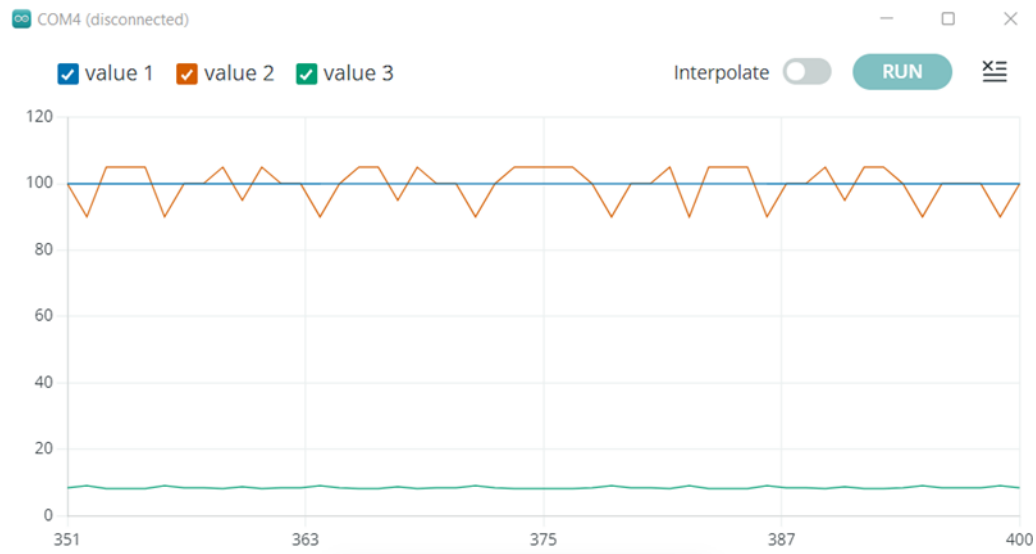
A continuación, se presentan resultados de algunas pruebas realizadas en el módulo de laboratorio implementado.

Prueba 1:

En la primera prueba se busca seguir velocidades de referencia constantes, para esto se realizan dos casos. Inicialmente se establece una velocidad de referencia de 100 r.p.s. El segundo caso corresponde a una velocidad de referencia de 40 r.p.s.

Figura 26.

Gráfica velocidad vs tiempo para una referencia de 100 r.p.s.



En la ilustración 26 se puede observar la gráfica de la velocidad resultante del motor en función del tiempo considerando una velocidad de referencia de 100 r.p.s. La línea Azul representa la velocidad de referencia que se ingresa al código, la línea roja es la velocidad del

motor la cual intenta ajustarse al valor de referencia ingresado. La línea verde representa la señal de error.

Figura 27.

Gráfica velocidad vs tiempo para una referencia de 40 r.p.s.



En la ilustración 27 se observan los resultados obtenidos considerando una velocidad de referenciade 40 r.p.s. En este caso se observa que el error es menos del 5%.

Prueba 2:

En esta prueba se realiza un cambio abrupto en la velocidad de referencia del motor de 20 a 80 r.p.s. Para esto se agregan unas líneas al código original.

Figura 28.

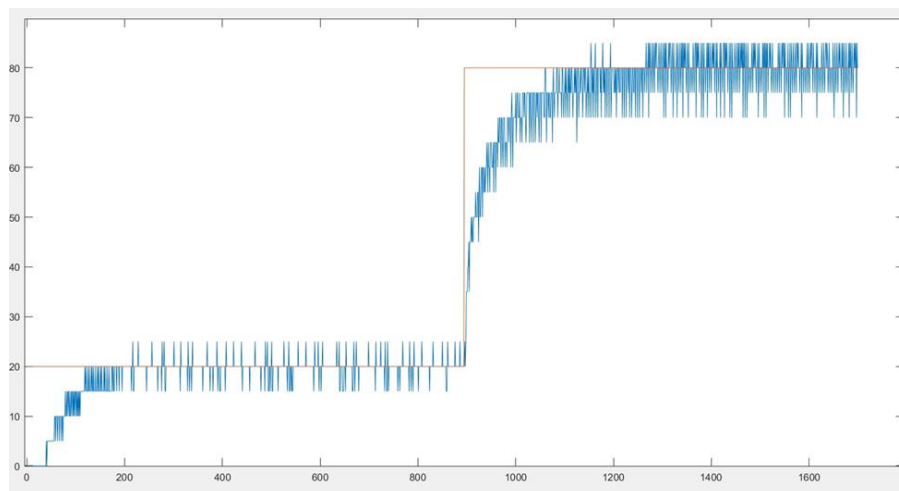
Código para el cambio de velocidad del motor transcurrido X tiempo

```
33 double PuntoReferencia;  
34 double PuntoReferencia1;  
35 double PuntoReferencia2;  
36 double ErrorAcumulado, VariacionError  
37 double Medido;  
38 int CambioDeReferencia = 10000;  
39  
40 //PuntoReferencia = analogRead(A0)/0.5;  
41  
42 PuntoReferencia1 = 20;  
43 PuntoReferencia2 = 80;  
44  
45 if (millis() < CambioDeReferencia){  
46   PuntoReferencia = PuntoReferencia1;  
47 }  
48  
49 if (millis() > CambioDeReferencia){  
50   PuntoReferencia = PuntoReferencia2;  
51 }  
52  
53 }  
54  
55 }  
56  
57 }  
58  
59 }  
60
```

En la ilustración 28 se aprecia el código utilizado para modificar la velocidad del motor transcurridos 10 segundos. El código se puede ver completo en el apéndice E.

Figura 29.

Gráfica cambio de 20 r.p.s. a 80 r.p.s. transcurridos 10 segundos.



En la ilustración 29 se observa la respuesta del sistema de control ante el cambio abrupto en la velocidad de referencia. Tal como se ve en la gráfica, la línea roja es la velocidad de referencia y la línea azul la velocidad del motor.

7. Conclusiones

En este trabajo de grado se desarrolló un módulo de laboratorio que permite realizar el control de velocidad de un motor DC de imanes permanentes con acceso remoto. Para la implementación del módulo, se diseñó e implementó un módulo de potencia compuesto de un convertidor DC/DC reductor junto al *driver* requerido para adecuación y aislamiento de señales y un sistema de alimentación. Se utilizó un *encoder* para la medición de la velocidad del motor, una placa Arduino Uno para la implementación del sistema de control, una cámara y dos computadores para la realización de prácticas experimentales de manera remota.

Este trabajo sirve para los estudiantes de la Universidad Industrial de Santander, que quieran profundizar en las áreas de control de sistemas eléctricos, electrónica de potencia y máquinas eléctricas I, por medio de una práctica de laboratorio que puede ser realizada ya sea presencial o de manera remota, y además abre las puertas a estudiantes de otras Universidades a nivel mundial, que quieran ampliar sus conocimientos en estas áreas.

La práctica de laboratorio está pensada también para los estudiantes que quieran por medio de prueba y error, obtener los valores de las constantes del PI del sistema control y observar los cambios al hacer diferentes ajustes.

Los resultados obtenidos muestran un error considerable, pero aun así se ajusta a los valores de velocidad de referencia ingresados al código. Las posibles causas del error pueden deberse a errores en la medida de la velocidad mediante al *encoder* y al tiempo requerido para realizar esta medición.

La aplicación Chrome Remote Desktop empleada para la realización de las prácticas de forma remota es de fácil acceso para los estudiantes y sin costo alguno, además es eficiente al

enviar y recibir señales por medio de internet.

Con el control implementado, el motor DC puede trabajar en un rango de velocidad entre 5 rps y 120 rps.

Con la práctica de laboratorio remoto diseñada, los estudiantes de la E3T pueden visualizar cambios que ocurren respecto a la velocidad del motor DC al variar la referencia y las constantes proporcional e integral del controlador.

Referencias

- Agencia de Información Laboral. (2021). *Régimen de Trabajo Remoto: una nueva forma de ejecución del contrato de trabajo*. Obtenido de <https://ail.ens.org.co/opinion/regimen-de-trabajo-remoto-una-nueva-forma-de-ejecucion-del-contrato-de-trabajo/#:~:text=Ahora%2C%20el%20trabajo%20remoto%20est%C3%A1%20concebido%20como%20una,para%20proceso%20de%20descargos%20laborales%20y%20Fo%20tr%C3%A1mite>
- Albert Paul Malvino, D. J. (2007). *Principios de Electronica*. Mcgraw-Hill.
- Alvarado, M. S. (2011). *Modelo matemático de un motor de corriente continua separadamente excitado: control de velocidad por corriente de armadura*. Guataquil-Ecuador.
- B.T.P. (2022). <https://informatica-robotica.site123.me/inf%C3%B3rmate-m%C3%A1s/kit-arduino-uno>.
- Brown, M. (2001). *Power Supply cookbook*.
- Carakenio73. (1 de Mayo de 2018). *Dinámica de un motor DC- diagrama de bloques, función de transferencia*. Obtenido de <https://dademuch.com/2018/05/01/dinamica-de-un-motor-dc/>
- Chapman, S. J. (2012). *Circuito equivalente de un motor dc*. Quinta edición. Ejury, J. (2013). *Buck Converter Design. fisica principio con aplicaciones*. (s.f.).
- Gordillo, J. (2020). *Desarrollo de Estrategias de Control Para Incrementar la Eficiencia de una PCH*. Obtenido de <http://repository.unipiloto.edu.co/bitstream/handle/20.500.12277/6858/PG-19-1-13%20DOCUMENTO%20FINAL%20V3%20OK.pdf?sequence=1&isAllowed=y>
- Hagglund, K. J. (s.f.). *Diagrama de bloques de un lazo de realimentación simple*.

Hart, D. W. (2001). *Electronica de Potencia Pearson educacion*. Madrid.

Herrera Asociados. (15 de Noviembre de 2021). *Teletrabajo, Trabajo en Casa y Trabajo Remoto*.

Obtenido de <https://herreraasociados.co/2021/09/20/teletrabajo-trabajo-en-casa-y-trabajo-remoto/#:~:text=Debido%20a%20lo%20sucedido%20en%20el%202020%20a,2008%20en%20la%20Ley%201221%20del%20mismo%20a%C3%B1o>.

Instituto Nacional de Educación Tecnológica. (s.f.). *google*. Obtenido de

http://www.inet.edu.ar/wp-content/uploads/2020/07/ELECTRONICA_Gu--a01-Resistencias.pdf

Martha Zapata. (2012). *Recursos educativos digitales: conceptos básicos*. Obtenido de

<https://aprendeonline.udea.edu.co/boa/contenidos.php/d211b52ee1441a30b59ae008e2d31386/845/estilo/aHR0cDovL2FwcmVuZGVlbnxpbmVhLnVkZWEuZWR1LmNvL2VzdGlsb3MvYXp1bF9jb3Jwb3JhdGl2by5jc3M=/1/contenido/>

MORA, J. F. (2003). *Máquinas Electricas*. Mc Fraw Hill.

Navarro, S. A. (2019). *diseño e implementacion de un modulo de labotatorio*. Bucaramanga.

Nyquist. (s.f.).

Ochoa, A., & Rodriguez, S. (2017). *Control en lazo cerrado de un convertidor CC/CC tipo Buck*

utilizando lógica difusa mediante la implementación de Arduino. Obtenido de <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/8403/621313016.pdf?sequence=1&isAllowed=y>

Ortega, M. G. (2009). *Obtención experimental de los parámetros del motor que se utilizara en el sistema de locomoción de una esfera rodante*. BUCARAMANGA.

Pomares, Jorge. (2009). <https://rua.ua.es/dspace/bitstream/10045/11833/1/arduino.pdf>.

Sergio Andrés Vargas Navarro, J. A. (2019). *Diseño e implementación de un módulo de*

laboratorio para el control de velocidad de un motor de corriente continua de imanes permanentes. Bucaramanga.

Simulación de Procesos Udenar. (s.f.). *Control proporcional integral derivative (PID) estabilidad sistemas dinámicos.* Obtenido de <https://simulaciondeprocesosudenar.wordpress.com/control-proporcional-integral-derivativo-pid-y-estabilidad-de-sistemas-dinamicos/>

UIS. (2021). *Objetivos del Programa Académico de Ingeniería Eléctrica.* Obtenido de <https://www.uis.edu.co/webUIS/es/academia/facultades/fisicoMecanicas/escuelas/e3t/programasAcademicos/ingenieriaElectrica/objetivos.jsp>

UIS. (2021). *Programa Académico: INGENIERIA ELECTRICA.* Obtenido de <https://www.uis.edu.co/webUIS/es/academia/facultades/fisicoMecanicas/escuelas/e3t/programasAcademicos/ingenieriaElectrica/planEstudios.html>

Uno, A. (s.f.). *arduino.cl.*

Urueña, E. B. (2009). *Obtencion experimental de los parametros del motor que se utilizara en el sistema de locomocion de una esfera rodante.* Bucaramaga.

Apéndices

Apéndice A. Guía para conectar dos ordenadores de forma remota

Para conectar de manera remota dos ordenadores se necesita realizar los siguientes pasos:

Configuración del ordenador al que se quiere conectar (el que está en el laboratorio)

- Tener disponible dos ordenadores, uno que se encontrará en el laboratorio (computador B) y al cual se va a conectar de forma remota y otro ordenador que va a cumplir con la función de manipular y realizar las acciones desde cualquier parte del mundo (computador A).
- El siguiente paso es tener conexión a internet en ambos ordenadores y tener acceso a una cuenta de Google (proporcionada al final de esta explicación).
- Para realizar la sincronización se requiere la extensión de Google Remote Desktop.
- Se descarga la extensión agregar a Chrome.

[Página principal](#) > [Extensiones](#) > [Chrome Remote Desktop](#)



Chrome Remote Desktop

remotedesktop.google.com

★★★★★ 1.898 ⓘ | [Productividad](#) | 10,000,000+ usuarios

Agregar a Chrome

- Después de descargarla se podrá ver la extensión en la parte superior derecha del ordenador.

0ebdde2d352944...



- Se abre la extensión y seguidamente va a pedir el correo y una clave para que el que quiera acceder al computador (B) se pueda identificar.
- Se ingresa el correo y la clave en los campos requeridos.
- Luego aparece la opción de aceptar, continuamente se va a instalar un archivo de configuración *.msi*.
- Se da el permiso de instalación y se ejecuta.
- Se elige un nombre con el cual aparecerá el ordenador a controlar, luego se crea un pin de seis números.
- Con lo anterior, debe aparecer el nombre del ordenador que se ha configurado y su estado online.

Configuración del ordenador desde el cual se va a manipular de forma remota

- En este computador A se debe abrir la cuenta de Google que se encuentra ingresada en el computador B.
- Aquí se debe descargar nuevamente la extensión Google Remote Desktop.
- Luego de realizar esta acción, se busca la extensión en la barra de tareas superior del navegador de Google y se da clic sobre ella.
- Seguidamente se le da acceder a mi ordenador.

Un método fácil para conectarte de forma remota al ordenador de casa o del trabajo, o de compartir tu pantalla con otras personas.



Accede de forma segura a tu ordenador cuando estés fuera, ya sea mediante tu teléfono, tu tablet u otro ordenador.

Comparte tu pantalla para colaborar en tiempo real, o conéctate a otro ordenador para ofrecer o recibir asistencia remota.

Acceder a mi ordenador

Compartir mi pantalla

- Después de realizar lo anterior, aparecerá el ordenador que se va a controlar de forma remota, se da clic sobre él, se introduce el código anteriormente creado y listo, se obtiene el control del ordenador.

Credenciales para ingresar al laboratorio virtual

Correo: Remotecontrolmotor@gmail.com clave: G-700control; código: 212121

Apéndice B. Cálculos IRFZ48N

- Se elige por su baja $R_{ds(on)}$, aproximadamente $14\text{m}\Omega$
- Capacitancia de entrada C_{in} : 1970Pf
- Carga de entrada 81Nc
- Tensión Drenador-Fuente Baja lo suficiente para este caso (55V)

Cálculo de la resistencia de puerta

Q_g : carga total de puerta

I_g : corriente de puerta

R_g : resistencia de puerta

$$Q_g = 81\text{nC}$$

$$V_{drive} = 12\text{V}$$

$$f = 30\text{kHz}$$

$$rampa = 1\% \frac{1}{f}$$

$$rampa = 1\% \frac{1}{30 * 10^3} * 0.01 = 334\text{ns}$$

$$I_g = \frac{81nc}{334ns} = 0.243 A$$

$$R_g = \frac{12}{0.243} = 48.5 \Omega$$

Corriente en los transistores

I_b: Corriente de base

I_c: Corriente del colector R_c: Resistencia del colector

β: Ganancia de corriente del transistor

KTC1027 – KIA 1023

$$I_c = 800mA, \quad PD = 1, \quad V_{CEO} = 120V$$

$$F_T = 120MHz, \quad h_{Fe} = 80 - 120$$

$$I_c = 250 mA \quad h_{Fe} = 80$$

$$I_b = \frac{250}{80} = 3.125 mA$$

Para asegurar saturación $I_b > 2 \cdot \beta \cdot I_b$

$$I_b = 10mA$$

$$R_c \frac{120V}{10mA} = 1.2 k\Omega$$

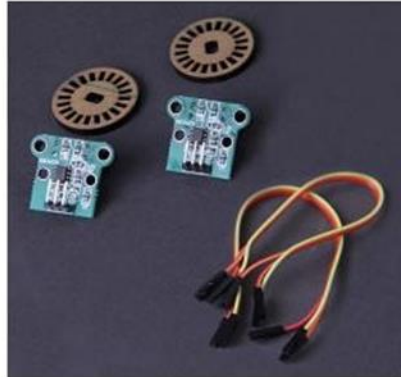
Para el 2N2222

Apéndice C. Double Speed Measuring – Module with Photoelectric Encoders

Model HC – 020K

Double Speed Measuring Module with Photoelectric Encoders

Model: HC-020K



Features:

HC-020K speed measuring sensor is a wide voltage, high resolution, short response time, and the switch output speed measurement module. It can test motor's rotational speed with black encoder (measured spec. is related to the encoder, the inner diameter of D type encoder that provided is 4mm, can be used for motor shaft w/ 4mm diameter, which is TT motor we matched, yellow shell and white axis).

Specifications:

1. Module Working Voltage: 4.5-5.5V
2. Launch Tube Pressure Drop: $V_F=1.6V$
3. Launch Tube Current: $I_f < 20mA$
4. Signal output: A, B two lines; TT power level;
5. Resolution: 0.01mm
6. Measurement frequency: 100 KHz
7. Color: Green + black
8. Disc diameter: 24mm
9. Inner Disc Diameter: 4mm
10. Encoder resolution: 20 lines
11. Speed measuring sensor configuration: measure line 1 motor speed
12. Application: For experiment

Made in China

Apéndice D. Código 1 velocidad de referencia ingresada manualmente

```

1 // Ingresamos los pines con los que vamos a trabajar
2
3 int Encoder = 11;
4 int SalidaSignal = 9;
5 int Potenciometro = 0;
6
7 // Variables para medir la velocidad con el encoder
8 unsigned long TiempoInicial = 0;
9 unsigned long TiempoFinal = 0;
10 int Pasos = 0;
11 float PasosAnteriores = 0;
12 float tempo = 0;
13 float Velocidad=0;
14 float Referencia = 0;
15 int Intervalo = 1000;
16
17 // Constantes del PID
18 /*
19 double kp = .001;
20 double ki = .000011;
21 double kd = 0.000;
22 */
23 double kp = 0.058;
24 double ki = 0.0001;
25 double kd = 0.000;
26
27 // Variables para la función Calculo PI
28 unsigned long TiempoActual, TiempoAnterior;
29 double TiempoTranscurrido;
30 double error;
31 double ErrorAnterior;
32 double Entrada, Salida;
33 double PuntoReferencia;
34 double ErrorAcumulado, VariacionError;
35 double Medido;
36
37 void setup()
38 {
39   TCCR2B = TCCR2B & B11111000 | B00000001;
40   Serial.begin(57600);
41   pinMode(Encoder, INPUT_PULLUP);
42   pinMode(Potenciometro, INPUT_PULLUP);
43 }
44
45 void loop()
46 {
47   TiempoInicial = millis(); // E:
48   TiempoFinal = TiempoInicial + 10; // Def:
49   while( millis() < TiempoFinal ) // S:
50   {
51     if(digitalRead(Encoder)) // S:
52     {
53       Pasos = Pasos + 1; // Hi:
54       while(digitalRead(Encoder)); // Ne:
55     }
56   }
57
58   tempo = Pasos - PasosAnteriores; // E:
59   PasosAnteriores = Pasos; // Si:
60   Velocidad = tempo/.2; // L:
61   if (Velocidad >120){ // S:
62     Velocidad = 120;
63   }
64   if (Velocidad < 0){ // S:
65     Velocidad = 0;
66   }
67 }

```

```

64     if (Velocidad < 0){
65     |   Velocidad = 0;
66     }
67
68     //PuntoReferencia = analogRead(A0)/8
69     PuntoReferencia = 100;
70     Medido = Velocidad;
71
72     TiempoActual = millis();
73     Salida = CalculoPID(Medido);
74
75     // Imprimimos datos en el Serial.pri
76     Serial.print(PuntoReferencia , 2);
77     Serial.print(" ");
78     Serial.print(Medido, 2);
79     Serial.print(" ");
80     Serial.print(Salida, 2);
81     Serial.print('\n');
82
83     analogWrite(8, Salida);
84
85     if (Salida > 20) {
86     |   Salida = 20;
87     |   ErrorAcumulado = 200000;
88     |   } // Si esta salida se pasa
89     if (Salida < 0) Salida = 0;
90
91     //Salida = 253 - Salida;
92     analogWrite(SalidaSignal, Salida);
93 }
94
95 // Función para el cálculo de la salida del PI
96 double CalculoPID(double Ent){
97     TiempoActual = millis();
98     TiempoTranscurrido = (double)(TiempoActual - TiempoAnterior);
99
100    error = PuntoReferencia - Ent;
101    ErrorAcumulado += error * TiempoTranscurrido;
102    VariacionError = (error - ErrorAnterior)/TiempoTranscurrido;
103
104    double Sal = kp*error + ki*ErrorAcumulado + kd*VariacionError;
105
106    ErrorAnterior = error;
107    TiempoAnterior = TiempoActual;
108
109    return Sal;
110 }
111

```

Apéndice E. Código 2 cambio de velocidad al pasar X tiempo

```

1 // Ingresamos los pines con los que vamos a trabajar
2
3 int Encoder = 11;
4 int SalidaSignal = 9;
5 int Potenciometro = 0;
6
7 // Variables para medir la velocidad con el encoder
8 unsigned long TiempoInicial = 0;
9 unsigned long TiempoFinal = 0;
10 int Pasos = 0;
11 float PasosAnteriores = 0;
12 float tempo = 0;
13 float Velocidad=0;
14 float Referencia = 0;
15 int Intervalo = 1000;
16
17 // Constantes del PID
18 /*
19 double kp = .001;
20 double ki = .000011;
21
22 */
23 double kp = 0.058;
24 double ki = 0.0001;
25 double kd = 0.000;
26
27 // Variables para la función Calculo PI
28 unsigned long TiempoActual, TiempoAnterior;
29 double TiempoTranscurrido;
30 double error;
31 double ErrorAnterior;
32 double Entrada, Salida;
33 double PuntoReferencia;
34 double PuntoReferencia1;
35 double PuntoReferencia2;
36 double ErrorAcumulado, VariacionError;
37 double Medido;
38 int CambioDeReferencia = 10000;
39
40 void setup()
41 {
42   TCCR2B = TCCR2B & B11111000 | B00000001; // for PWM frequency of 31372.55 Hz funciona entre el 6 y el 9
43   Serial.begin(57600); // Inicializamos el Serial.print para ver los datos en tiempo real
44   pinMode(Encoder,INPUT_PULLUP); // Especificamos que el Encoder es una entrada con Pullup (con resistencia para medir un voltaje referenciado)
45   pinMode(Potenciometro,INPUT_PULLUP); // El potenciómetro también va a ser una entrada con Pullup
46 }
47
48 void loop()
49 {
50   TiempoInicial = millis(); // Es el tiempo actual o la cantidad de milisegundos que han pasado desde la ejecución del código
51   TiempoFinal = TiempoInicial + 10; // Definimos un tiempo final sobre el que vamos a trabajar, que es inicio + intervalo
52   while( millis() < TiempoFinal ) // Siempre que el tiempo actual sea menor al máximo que definimos (que estemos dentro del intervalo)...
53   {
54     if(digitalRead(Encoder)) // Si se está leyendo el encoder
55     {
56       Pasos = Pasos + 1; // Haga un conteo de pasos
57       while(digitalRead(Encoder)); // No se continúa hasta que se haga una nueva lectura del encoder
58     }
59   }
60
61   tempo = Pasos - PasosAnteriores; // El tiempo equivale al número de pasos que dimos menos los anteriores. Se comienzan los anteriores en cero
62   PasosAnteriores = Pasos; // Se actualizan los pasos anteriores

```

```

63 Velocidad = tiempo/.2; // La velocidad es el tiempo dividido 20 (número de ranuras de la rueda del encoder)
64 if (Velocidad >120){ // Si la velocidad se pasa de 120, llévela a 120
65     Velocidad = 120;
66 }
67 if (Velocidad < 0){ // Si la velocidad se va a cero, déjela en cero
68     Velocidad = 0;
69 }
70
71 //PuntoReferencia = analogRead(A0)/8.5; // Este valor se convierte en la referencia (se ajusta para mantenerlo entre 0 - 120)
72 PuntoReferencia1 = 20;
73 PuntoReferencia2 = 80;
74 if (millis() < CambioDeReferencia){
75     PuntoReferencia = PuntoReferencia1;
76 }
77 if (millis() > CambioDeReferencia){
78     PuntoReferencia = PuntoReferencia2;
79 }
80
81 Medido = Velocidad; // Es el valor real medido de velocidad
82
83 TiempoActual = millis(); // Tiempo actual de simulación
84 Salida = CalculoPID(Medido); // Pasamos por el lazo del PID y obtenemos Salida como un valor entre 0 y 253 (que se va a convertir en el ciclo de trabajo)
85
86 // Imprimimos datos en el Serial.print para verlos en tiempo real
87 Serial.print(PuntoReferencia , 2); // Qué referencia le estamos ingresando con el potenciómetro (o si es constante y ya está definida)
88 Serial.print(" "); // Espacio
89 Serial.print(Medido, 2); // La velocidad que estamos midiendo
90 Serial.print(" ");
91 Serial.print(Salida, 2); // La salida del PID, que puede ser un valor mucho mayor a 253 o menor a 0 en el caso en que no se esté controlando, pero debería mantenerse
92 Serial.print('\n'); // Nueva línea
93 Serial.print('\n'); // Nueva línea
94
95 analogWrite(8, Salida); // Escribimos en el pin 8 la salida (ciclo de trabajo) (para verla en el osciloscopio)
96
97 if (Salida > 20) {
98     Salida = 20;
99     ErrorAcumulado = 200000;
100 } // Si esta salida se pasa del valor máximo, la dejamos en 253. Si no se hace este paso, el ciclo de trabajo vuelve a cero (desbordamiento)
101 if (Salida < 0) Salida = 0; // Si se hace menor a 1, se deja en cero
102
103 //Salida = 253 - Salida; // Negamos la salida (para ingresarla al convertidor Buck)
104 analogWrite(SalidaSignal, Salida); // Sacamos el dato por el pin 11, SalidaSignal
105 }
106
107 // Función para el cálculo de la salida del PI
108 double CalculoPID(double Ent){ // Ent es la variable de velocidad medida
109     TiempoActual = millis(); // Guardamos el tiempo actual
110     TiempoTranscurrido = (double)(TiempoActual - TiempoAnterior); // Se calcula el tiempo transcurrido
111
112     error = PuntoReferencia - Ent; // Calculamos el error entre la referencia y la velocidad medida (entrada)
113     ErrorAcumulado += error * TiempoTranscurrido; // Se calcula la integral discreta
114     VariacionError = (error - ErrorAnterior)/TiempoTranscurrido; // Se calcula la derivada discreta
115
116     double Sal = kp*error + ki*ErrorAcumulado ; // Sal es la salida del lazo que pasa por el PID
117
118     ErrorAnterior = error; // Guardamos el último error para usarlo en la siguiente simulación
119     TiempoAnterior = TiempoActual; // Guardar el dato del tiempo actual para la siguiente iteración
120
121     return Sal; // Sal es el dato que devuelve la función cada vez que corre
122 }

```

Apéndice F. Datasheets De Importancia Pn2222

PN2222, PN2222A

PN2222A is a Preferred Device

General Purpose Transistors

NPN Silicon

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Collector-Emitter Voltage PN2222 PN2222A	V_{CE0}	30 40	Vdc
Collector-Base Voltage PN2222 PN2222A	V_{CBO}	60 75	Vdc
Emitter-Base Voltage PN2222 PN2222A	V_{EBO}	5.0 6.0	Vdc
Collector Current – Continuous	I_C	600	mAdc
Total Device Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	625 5.0	mW mW/ $^\circ\text{C}$
Total Device Dissipation @ $T_C = 25^\circ\text{C}$ Derate above 25°C	P_D	1.5 12	Watts mW/ $^\circ\text{C}$
Operating and Storage Junction Temperature Range	T_J, T_{stg}	-55 to +150	$^\circ\text{C}$

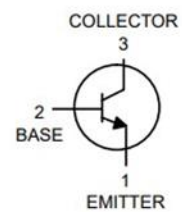
THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance Junction-to-Ambient	$R_{\theta JA}$	200	$^\circ\text{C}/\text{W}$
Thermal Resistance Junction-to-Case	$R_{\theta JC}$	83.3	$^\circ\text{C}/\text{W}$



ON Semiconductor™

<http://onsemi.com>



MARKING DIAGRAM



TO-92
CASE 29
STYLE 1

PN222x = Device Code
x = 2 or A
Y = Year
WW = Work Week

Transistor 6N136

SINGLE-CHANNEL 6N135, 6N136 HCPL-2503 HCPL-4502	DUAL-CHANNEL HCPL-2530 HCPL-2531
--	---

DESCRIPTION

The HCPL-4502/HCPL-2503, 6N135/6 and HCPL-2530/HCPL-2531 optocouplers consist of an AlGaAs LED optically coupled to a high speed photodetector transistor.

A separate connection for the bias of the photodiode improves the speed by several orders of magnitude over conventional phototransistor optocouplers by reducing the base-collector capacitance of the input transistor.

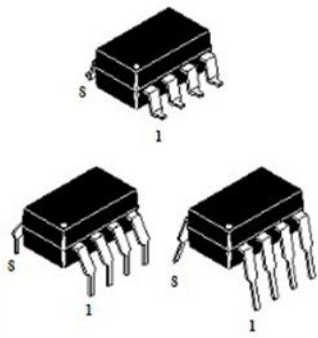
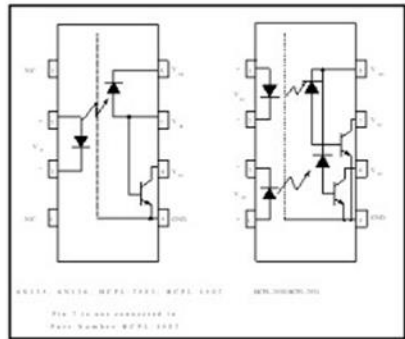
An internal noise shield provides superior common mode rejection of 10kV/ μ s. An improved package allows superior insulation permitting a 480 V working voltage compared to industry standard of 220 V.

FEATURES

- High speed-1 MBit/s
- Superior CMR-10 kV/ μ s
- Dual-Channel
HCPL-2530/HCPL-2531
- Double working voltage-480V RMS
- CTR guaranteed 0-70°C
- U.L. recognized (File # E90700)

APPLICATIONS

- Line receivers
- Pulse transformer replacement
- Output interface to CMOS-LSTTL-TTL
- Wide bandwidth analog coupling



ABSOLUTE MAXIMUM RATINGS (TA= 25°C unless otherwise specified)			
Parameter	Symbol	Value	Units
Storage Temperature	T _{STG}	-55 to +125	°C
Operating Temperature	T _{OPR}	-55 to +100	°C
Lead Solder Temperature	T _{SOL}	260 for 10 sec	°C
EMITTER			
DC/Average Forward Input Current	I _F (avg)	25	mA
Peak Forward Input Current (50% duty cycle, 1 ms P.W.)	I _F (pk)	50	mA
Peak Transient Input Current - (10 ⁻¹ μ s P.W., 300 pps)	I _F (trans)	1.0	A
Reverse Input Voltage	V _R	5V	
Input Power Dissipation	P _D	100	mW
DETECTOR			
Average Output Current	I _O (avg)	8	mA
Peak Output Current	I _O (pk)	16	mA
Emitter-Base Reverse Voltage (6N135, 6N136 and HCPL-2503 only)	V _{EBR}	5V	
Supply Voltage	V _{CC}	-0.5 to 30	V
Output Voltage	V _O	-0.5 to 20	V
Base Current (6N135, 6N136 and HCPL-2503 only)	I _B	5 mA	
Output power (6N135, 6N136, HCPL-2503, HCPL-4502) (Note 4)		100	mW
dissipation (HCPL-2530, HCPL-2531) Each Channel	P _D	35	mW

200004A

Datasheet Transistor IRFN48N

Philips Semiconductors

Product specification

**N-channel enhancement mode
TrenchMOS™ transistor**

IRFZ48N

GENERAL DESCRIPTION

N-channel enhancement mode standard level field-effect power transistor in a plastic envelope using 'trench' technology. The device features very low on-state resistance and has integral zener diodes giving ESD protection up to 2kV. It is intended for use in switched mode power supplies and general purpose switching applications.

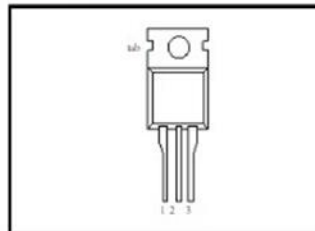
QUICK REFERENCE DATA

SYMBOL	PARAMETER	MAX.	UNIT
V_{DS}	Drain-source voltage	55	V
I_D	Drain current (DC)	64	A
P_{tot}	Total power dissipation	140	W
T_j	Junction temperature	175	°C
$R_{DS(on)}$	Drain-source on-state resistance $V_{GS} = 10\text{ V}$	16	m Ω

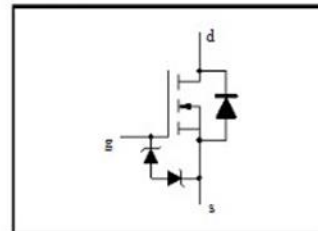
PINNING - TO220AB

PIN	DESCRIPTION
1	gate
2	drain
3	source
tab	drain

PIN CONFIGURATION



SYMBOL



LIMITING VALUES

Limiting values in accordance with the Absolute Maximum System (IEC 134)

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{DS}	Drain-source voltage	-	-	55	V
V_{DGS} $= V_{GS}$	Drain-gate voltage	$R_{GS} = 20\text{ k}\Omega$	- 5 5	-	V
	Gate-source voltage	-	-	20	V
I_D	Drain current (DC)	$T_{mb} = 25\text{ }^\circ\text{C}$	-	64	A
I_D	Drain current (DC)	$T_{mb} = 100\text{ }^\circ\text{C}$	-	45	A
I_{DM}	Drain current (pulse peak value)	$T_{mb} = 25\text{ }^\circ\text{C}$	-	210	A
P_{tot}	Total power dissipation	$T_{mb} = 25\text{ }^\circ\text{C}$	-	140	W
T_{stg}, T_c	Storage & operating temperature	-	- 55	175	°C

ESD LIMITING VALUE

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_C	Electrostatic discharge capacitor voltage, all pins	Human body model (100 pF, 1.5 k Ω)	-	2	kV

THERMAL RESISTANCES

SYMBOL	PARAMETER	CONDITIONS	TYP.	MAX.	UNIT
$R_{th(j-c)}$	Thermal resistance junction to mounting base	-	-	1.1	K/W
$R_{th(j-a)}$	Thermal resistance junction to ambient	in free air	60	-	K/W

KTC1027 TRANSISTOR (NPN)**FEATURES**

Power dissipation

$$P_{CM} = 1 \text{ W (Tamb=25°C)}$$

Collector current

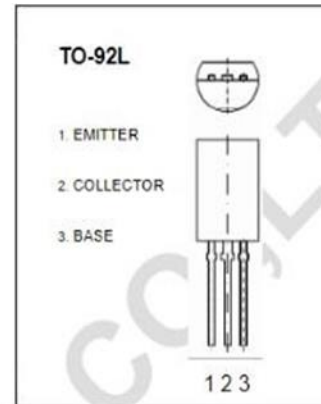
$$I_{CM} = 0.8 \text{ A}$$

Collector-base voltage

$$V_{(BR)CBO} = 120 \text{ V}$$

Operating and storage junction temperature range

$$T_J, T_{stg}: -55^\circ\text{C to } +150^\circ\text{C}$$

**ELECTRICAL CHARACTERISTICS (Tamb=25°C unless otherwise specified)**

Parameter	Symbol	Test conditions	MIN	TYP	MAX	UNIT
Collector-base breakdown voltage	$V_{(BR)CBO}$	$I_C=1\text{mA}, I_E=0$	120			V
Collector-emitter breakdown voltage	$V_{(BR)CEO}$	$I_C=10\text{mA}, I_B=0$	120			V
Emitter-base breakdown voltage	$V_{(BR)EBO}$	$I_E=1\text{mA}, I_C=0$	5			V
Collector cut-off current	I_{CBO}	$V_{CB}=120\text{V}, I_E=0$			0.1	μA
Emitter cut-off current	I_{EBO}	$V_{EB}=5\text{V}, I_C=0$			0.1	μA
DC current gain	$h_{FE(1)}$	$V_{CE}=5\text{V}, I_C=100\text{mA}$	80		240	
Collector-emitter saturation voltage	$V_{CE(sat)}$	$I_C=500\text{mA}, I_B=50\text{mA}$			1	V
Base-emitter voltage	V_{BE}	$V_{CE}=5\text{V}, I_C=500\text{mA}$			1	V
Transition frequency	f_T	$V_{CE}=5\text{V}, I_C=100\text{mA}$		120		MHz
Collector output capacitance	C_{ob}	$V_{CB}=10\text{V}, I_E=0, f=1\text{MHz}$			30	pF

Metal Oxide Varistor (MOV) Data Sheet

Features

- Wide operating voltage (V_{1mA}) range from 18V to 1800V
- Fast responding to transient over-voltage
- Large absorbing transient energy capability
- Low clamping ratio and no follow-on current
- Meets MSL level 1, per J-STD-020
- Operating Temperature: $-40^{\circ}\text{C} \sim +85^{\circ}\text{C}$
- Storage Temperature: $-40^{\circ}\text{C} \sim +125^{\circ}\text{C}$
- Safety certification: UL: E327997
CSA: 246579
VDE: 40027827



Applications

- Transistor, diode, IC, thyristor or triac semiconductor protection
- Surge protection in consumer electronics
- Surge protection in industrial electronics
- Surge protection in electronic home appliances, gas and petroleum appliances
- Relay and electromagnetic valve surge absorption

MOTOROLA
SEMICONDUCTOR TECHNICAL DATA

Order this document
by 1N5400/D

**Axial-Lead
Standard Recovery Rectifiers**

Lead mounted standard recovery rectifiers are designed for use in power supplies and other applications having need of a device with the following features:

- High Current to Small Size
- High Surge Current Capability
- Low Forward Voltage Drop
- Void-Free Economical Plastic Package
- Available in Volume Quantities

Mechanical Characteristics

- Case: Epoxy, Molded
- Weight: 1.1 gram (approximately)
- Finish: All External Surfaces Corrosion Resistant and Terminal Leads are Readily Solderable
- Lead and Mounting Surface Temperature for Soldering Purposes: 220°C Max. for 10 Seconds, 1/16" from case
- Shipped in plastic bags, 5,000 per bag.
- Available Tape and Reeled, 1500 per reel, by adding a "RL" suffix to the part number
- Polarity: Cathode Indicated by Polarity Band
- Marking: 1N5400, 1N5401, 1N5402, 1N5404, 1N5406, 1N5407, 1N5408

**1N5400
thru
1N5408**

1N5404 and 1N5408 are
Motorola Preferred Devices

**STANDARD
RECOVERY RECTIFIERS
50-1000 VOLTS
3.0 AMPERE**



MAXIMUM RATINGS

Rating	Symbol	1N5400	1N5401	1N5402	1N5404	1N5406	1N5407	1N5408	Unit
Peak Repetitive Reverse Voltage Working Peak Reverse Voltage DC Blocking Voltage	V _{RRM} V _{RWM} V _R	50	100	200	400	800	800	1000	Volts
Non-repetitive Peak Reverse Voltage	V _{RSM}	100	200	300	525	800	1000	1200	Volts
Average Rectified Forward Current (Single Phase Resistive Load, 1/2" Leads, T _L = 105°C)	I _O	3.0							Amp
Non-repetitive Peak Surge Current (Surge Applied at Rated Load Conditions)	I _{FSM}	200 (one cycle)							Amp
Operating and Storage Junction Temperature Range	T _J T _{stg}	- 65 to +170 - 65 to +175							°C

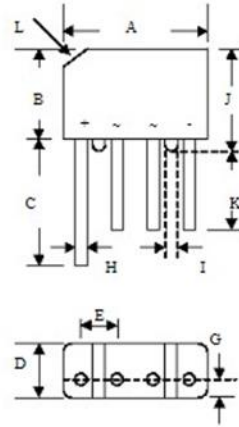


KBP200 – KBP2010

2.0A BRIDGE RECTIFIER

Features

- ! Diffused Junction
- ! Low Forward Voltage Drop
- ! High Current Capability
- ! High Reliability
- ! High Surge Current Capability
- ! Ideal for Printed Circuit Boards
- ! UL Recognized File = E157705



KBP		
Dim	Min	Max
A	14.22	15.24
B	10.67	11.68
C	15.2	—
D	4.57	5.08
E	3.60	4.10
G	2.16	2.67
H	0.76	0.86
I	1.52	—
J	11.68	12.7
K	12.7	—
L	3.2 x 45° Typical	
All Dimensions in mm		

Mechanical Data

- ! Case: Molded Plastic
- ! Terminals: Plated Leads Solderable per MIL-STD-202, Method 208
- ! Polarity: As Marked on Body
- ! Weight: 1.7 grams (approx.)
- ! Mounting Position: Any
- ! Marking: Type Number

Maximum Ratings and Electrical Characteristics @T_a = 25°C unless otherwise specified

Single Phase, half wave, 60Hz, resistive or inductive load.
For capacitive load, derate current by 20%.

Characteristic	Symbol	KBP 200	KBP 201	KBP 202	KBP 204	KBP 206	KBP 208	KBP 2010	Unit
Peak Repetitive Reverse Voltage	V _{RRM}	50	100	200	400	600	800	1000	V
Working Peak Reverse Voltage	V _{WRM}								
DC Blocking Voltage	V _R								
RMS Reverse Voltage	V _{RRMS}	35	70	140	280	420	560	700	V
Average Rectified Output Current (Note 1)	I _O	2.0							A
Non-Repetitive Peak Forward Surge Current (8.3ms Single half sine-wave superimposed on rated load (JEDEC Method))	I _{FSM}	60							A
Forward Voltage (per element) @I _F = 2.0A	V _{FM}	1.1							V
Peak Reverse Current @T _a = 25°C	I _{RM}	10							μA
At Rated DC Blocking Voltage @T _a = 100°C		500							
Rating for Fusing (t < 8.3ms)	I _{FT}	15							A
Typical Junction Capacitance per element (Note 2)	C _J	25							pF
Typical Thermal Resistance (Note 3)	R _{θJA}	30							K/W
Operating and Storage Temperature Range	T _J , T _{STG}	-55 to +165							°C



**L7800
SERIES**

POSITIVE VOLTAGE REGULATORS

- OUTPUT CURRENT TO 1.5A
- OUTPUT VOLTAGES OF 5; 5.2; 6; 8; 8.5; 9; 10; 12; 15; 18; 24V
- THERMAL OVERLOAD PROTECTION
- SHORT CIRCUIT PROTECTION
- OUTPUT TRANSITION SOA PROTECTION

DESCRIPTION

The L7800 series of three-terminal positive regulators is available in TO-220, TO-220FP, TO-220FM, TO-3 and D²PAK packages and several fixed output voltages, making it useful in a wide range of applications. These regulators can provide local on-card regulation, eliminating the distribution problems associated with single point regulation. Each type employs internal current limiting, thermal shut-down and safe area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltage and currents.

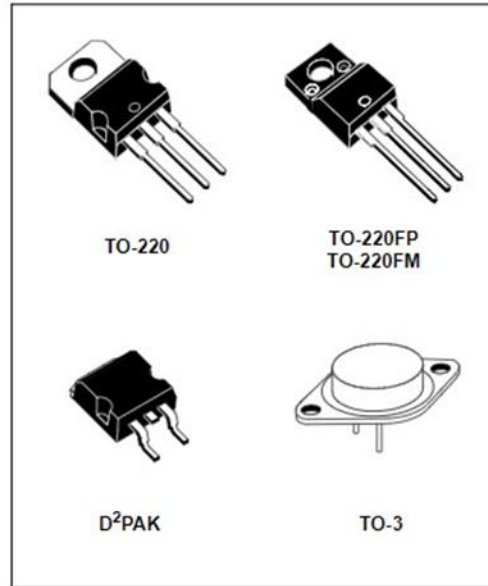


Table 1: Absolute Maximum Ratings

Symbol	Parameter		Value	Unit
V _I	DC Input Voltage	for V _O = 5 to 18V	35	V
		for V _O = 20, 24V	40	
I _O	Output Current		Internally Limited	
P _{tot}	Power Dissipation		Internally Limited	
T _{stg}	Storage Temperature Range		-65 to 150	°C
T _{op}	Operating Junction Temperature Range	for L7800	-55 to 150	°C
		for L7800C	0 to 150	

Absolute Maximum Ratings are those values beyond which damage to the device may occur. Functional operation under these condition is not implied.

Table 2: Thermal Data

Symbol	Parameter	D ² PAK	TO-220	TO-220FP	TO-220FM	TO-3	Unit
R _{thj-case}	Thermal Resistance Junction-case Max	3	5	5	5	4	°C/W
R _{thj-amb}	Thermal Resistance Junction-ambient Max	62.5	50	60	60	35	°C/W



SB2020CT – SB20100CT

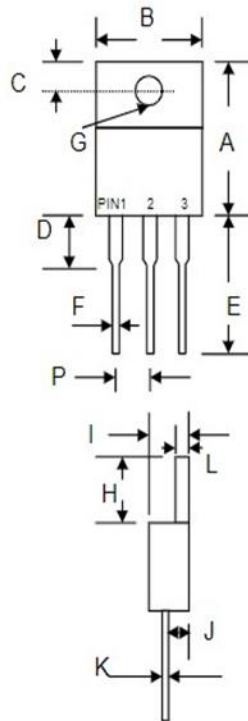
20A SCHOTTKY BARRIER RECTIFIER

Features

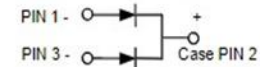
- Schottky Barrier Chip
- Guard Ring for Transient Protection
- High Current Capability, Low Forward
- Low Reverse Leakage Current
- High Surge Current Capability
- Plastic Material has UL Flammability Classification 94V-0

Mechanical Data

- Case: TO-220 Molded Plastic
- Terminals: Plated Leads Solderable per MIL-STD-750, Method 2026
- Polarity: As Marked on Body
- Weight: 2.24 grams (approx.)
- Mounting Position: Any
- Marking: Type Number



TO-220		
Dim	Min	Max
A	14.9	15.1
B	—	10.5
C	2.62	2.87
D	3.56	4.06
E	13.46	14.22
F	0.68	0.94
G	3.74 Ø	3.91 Ø
H	5.84	6.86
I	4.44	4.70
J	2.54	2.79
K	0.35	0.64
L	1.14	1.40
P	2.41	2.67
All Dimensions in mm		



Maximum Ratings and Electrical Characteristics @ $T_A=25^\circ\text{C}$ unless otherwise specified