

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)

GRADO EN INGENIERÍA ELECTROMECÁNICA

Diseño y ensayo de un regulador de tensión automático para máquinas síncronas

Autor: Miguel Berbegal Almohalla

Directores: Pablo Frías Marín, Lukas Sigrist

Madrid

Junio 2016



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)

GRADO EN INGENIERÍA ELECTROMECÁNICA

Diseño y ensayo de un regulador de tensión automático para máquinas síncronas

Autor: Miguel Berbegal Almohalla

Directores: Pablo Frías Marín, Lukas Sigrist

Madrid

Junio 2016

AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESINAS O MEMORIAS DE BACHILLERATO

1º. Declaración de la autoría y acreditación de la misma.

El autor D.Miguel Berbegal Almohalla DECLARA ser el titular de los derechos de propiedad intelectual de la obra: Diseño y ensayo de un regulador de tensión automático para máquinas sínronas, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

2º. Objeto y fines de la cesión.

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

3º. Condiciones de la cesión y acceso

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar "marcas de agua" o cualquier otro sistema de seguridad o de protección.
b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
e) Asignar por defecto a estos trabajos una licencia Creative Commons.
f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente).*

4º. Derechos del autor.

El autor, en tanto que titular de una obra tiene derecho a:

a) Que la Universidad identifique claramente su nombre como autor de la misma
b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
c) Solicitar la retirada de la obra del repositorio por causa justificada.
d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

5º. Deberes del autor.

El autor se compromete a:

a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la

intimidad y a la imagen de terceros.

c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.
d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción de derechos derivada de las obras objeto de la cesión.

6º. Fines y funcionamiento del Repositorio Institucional.

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusive del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a 12 de Junio de 2016

ACEPTA

Fdo.....

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título Diseño y ensayo de un regulador de tensión automático para máquinas síncronas en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el

curso académico 2015-2016 es de mi autoría, original e inédito y

no ha sido presentado con anterioridad a otros efectos. El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada

de otros documentos está debidamente referenciada.

Fdo.:  Fecha: 12 / 06 / 2016

Autorizada la entrega del proyecto

LOS DIRECTORES DEL PROYECTO

Fdo.:  Fecha: 13 / 6 / 2016

Fdo.:  Fecha: 14 / 6 / 2016

Vº Bº del Coordinador de Proyectos

Fdo.:  Fecha: 16 / 06 / 2016

DISEÑO Y ENSAYO DE UN REGULADOR DE TENSIÓN AUTOMÁTICO PARA MÁQUINAS SÍNCRONAS

Autor: Berbegal Almohalla, Miguel.

Directores: Frías Marín, Pablo; Sigrist, Lukas.

Entidad Colaboradora: ICAI – Universidad Pontificia de Comillas.

RESUMEN DEL PROYECTO

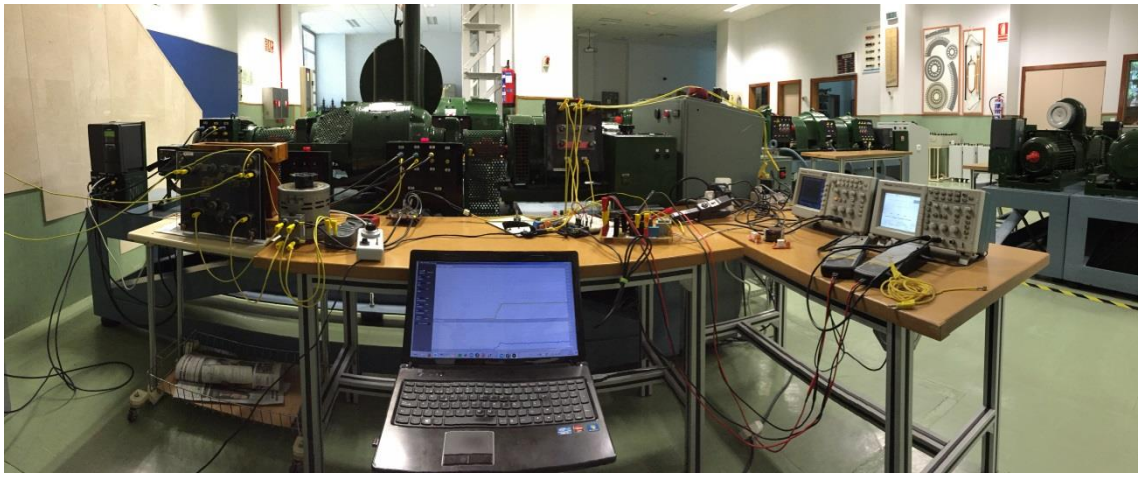
La regulación de tensión juega un papel fundamental en la generación y transporte de energía eléctrica. La tensión en bornes de un generador síncrono (el más utilizado para la producción de energía eléctrica) depende directamente de la corriente de excitación que circule por el rotor, pero hay muchos otros factores externos que influyen, siendo el más claro ejemplo la variación de carga a la que todo generador se encuentra constantemente sometido. Otros factores son la velocidad de rotación o la temperatura de los devanados. La tensión en bornes del generador debe poder mantenerse constante a pesar de todos estos factores tanto para el correcto funcionamiento de la máquina como para asegurar la calidad del suministro y es de esto precisamente de lo que se encargan los reguladores automáticos de tensión.

Un regulador automático de tensión es un dispositivo electrónico que ajusta la corriente de excitación de una máquina síncrona para mantener la tensión de estator igual a la tensión de referencia. El funcionamiento básico de un regulador consiste en medir la tensión de estator, compararla con la de consigna y subir o bajar la corriente de excitación en consecuencia.

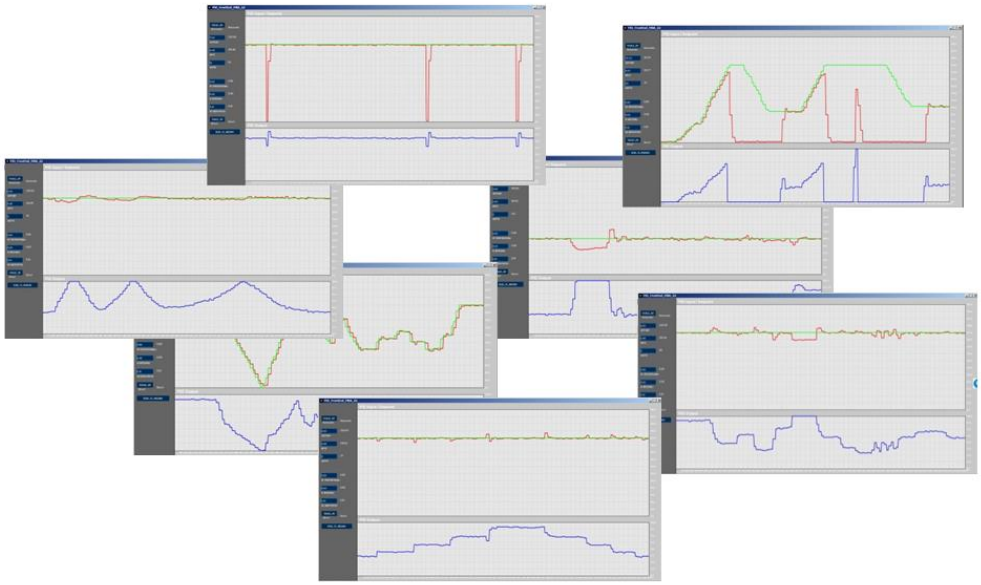
Este proyecto consiste en el diseño de un regulador de tensión para pequeñas máquinas síncronas (en el entorno de los 10kVA) con excitación estática independiente. Los objetivos del proyecto incluían el diseño del hardware de medida, control y potencia y el diseño del software incluyendo un control PID, un limitador de corriente de excitación y una interfaz que permita monitorizar y ajustar el control en tiempo real por medio de un ordenador. Además, una vez diseñado, el siguiente objetivo consistía en la construcción y ensayo de un prototipo.

Para el control se ha utilizado la placa **Arduino UNO**, que es una plataforma de hardware libre con procesador de la casa ATmel con su propio entorno de desarrollo basado en C++. Para la interfaz de monitorización y ajuste se ha utilizado **Processing**, que es un lenguaje de programación y un entorno de desarrollo, de código abierto también, basado en Java. En el apartado del hardware, la etapa de potencia se ha basado en un **transistor IGBT** actuando como “chopper”.

Este proyecto ha dado como resultado un regulador automático de tensión de bajo coste (150€), basado en hardware y software libre, que ha superado satisfactoriamente todos los ensayos demostrando ser capaz de regular la tensión eficazmente, tanto en vacío como en carga, rechazando todo tipo de perturbaciones como variaciones en la carga, la velocidad o la tensión de excitación. Además, trae consigo una interfaz que permite visualizar las principales variables del control y ajustar los parámetros de forma instantánea sin necesidad de detener el regulador. El limitador de corriente también funciona según lo esperado, cortando la excitación y señalando el estado de alarma mediante un LED cuando la corriente del rotor sobrepasa un valor determinado, esperando el rearme mediante un pulsador.



FOTOGRAFÍA DEL PROTOTIPO EN EL BANCO DE ENSAYOS



CAPTURAS DE DIVERSOS ENSAYOS UTILIZANDO LA INTERFAZ DE MONITORIZACIÓN Y AJUSTE

DESIGN AND TESTING OF AN INTEGRATED STATIC VOLTAGE REGULATOR FOR SYNCHRONOUS MACHINES

ABSTRACT

Voltage regulation plays a key role in the generation and transmission of electricity. The terminal voltage of a synchronous generator (the most commonly used generator for electricity production) directly depends on the excitation current that flows through the rotor, but there are many other external factors that may influence this voltage, being the clearest example of one of them load variation, which every generator is constantly subjected to. Other external factors are rotational speed or windings' temperature. The voltage across the power generator must remain constant despite all these factors both for the proper functioning of the machine as to ensure the quality of supply. This is precisely what automatic voltage regulators are responsible for.

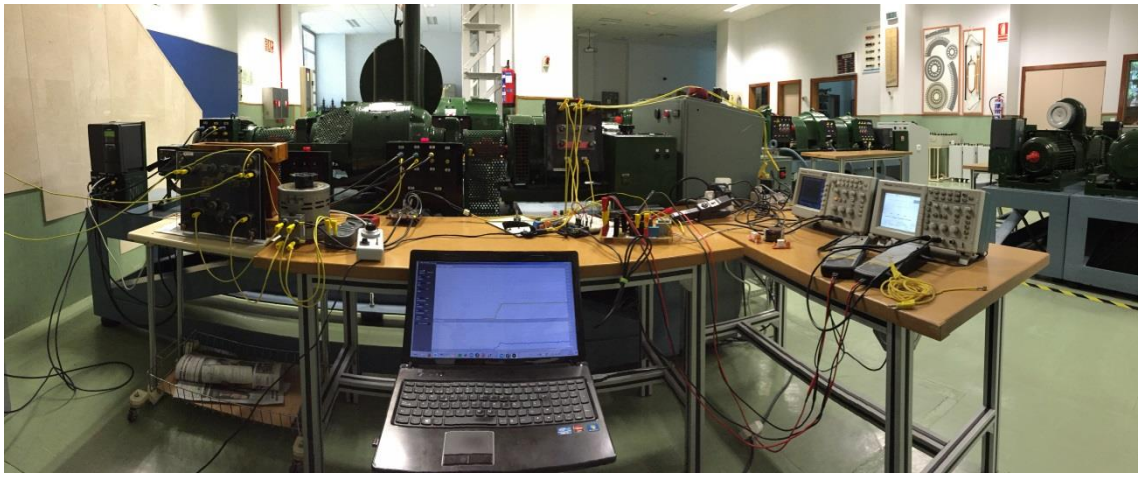
An automatic voltage regulator is an electronic device that adjusts the drive current to maintain a synchronous machine's stator voltage equal to the reference voltage. The basic operation of a regulator is to measure the stator voltage, compare it to the setpoint voltage and to raise or lower the excitation current accordingly.

This project tackles the design of a voltage regulator for small synchronous machines (around 10kVA) with separate static excitation. The objectives of the project included the design of measurement, control and power hardware as well as software design including a PID control, an excitation's current limiter and an interface that allows real-time monitoring and tuning of the control by means of a computer. In addition, once designed, the next objective was the construction and testing of a prototype.

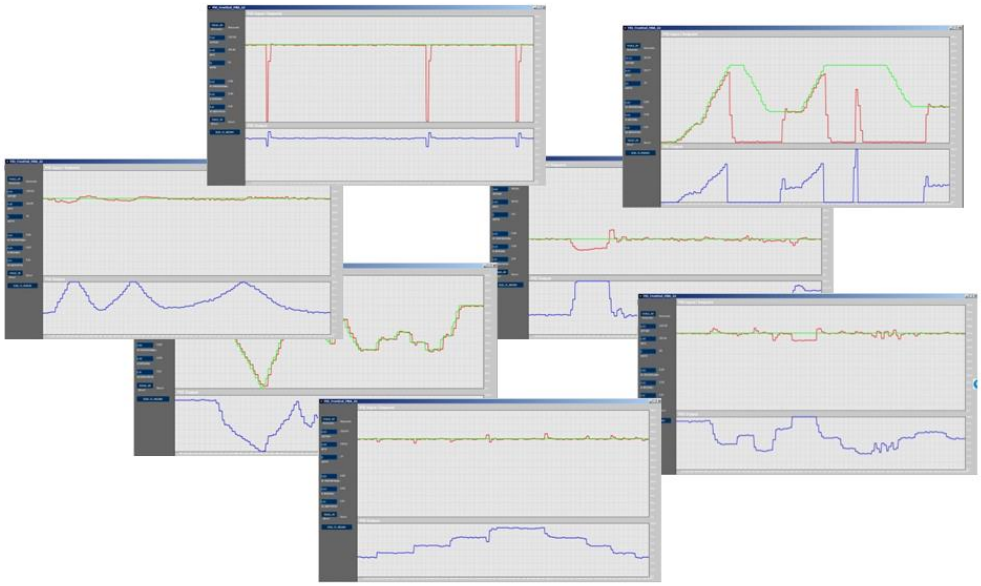
For the control Arduino UNO has been used, which is a free hardware platform with Atmel processor that has its own development environment based on C ++. For the monitoring and tuning interface Processing was used, an open source programming language and integrated development environment (IDE) built for the electronic arts, new media art, and visual design communities based on Java. Regarding the hardware, the power stage is based on an IGBT transistor acting as a "chopper".

This project led to a "low-cost" automatic voltage regulator (150 €), based on free hardware and software, which has satisfactorily passed all tests proving to be able to regulate voltage effectively, both with and without load, rejecting all kinds of disturbances such as variations in load, speed or drive voltage. In addition, it comes with an interface that displays the main variables of the control and is able to change control parameters instantly without stopping the regulator. The current limiter also

works as expected, by driving excitation current to zero and turning on the alarm state by an LED when the rotor current exceeds a certain value, waiting for reset.



PROTOTYPE CONNECTED TO THE TEST BENCH



VARIOUS TESTS' SCREENSHOTS USING THE PROCESSING'S MONITORING AND TUNING INTERFACE



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)

GRADO EN INGENIERÍA ELECTROMECÁNICA

Diseño y ensayo de un regulador de tensión automático para máquinas síncronas

Autor: Miguel Berbegal Almohalla

Directores: Pablo Frías Marín, Lukas Sigrist

Madrid

Mayo 2016

Agradecimientos

A mis padres por su constante apoyo y consejo.

A mis directores Pablo y Lukas por su imprescindible guía y supervisión.

A mi profesor Aurelio por su desinteresado soporte en temas electrónicos.

A Julián, que me ha ayudado en todo lo que he necesitado en el laboratorio.

A mis profesores de ICAI, que me han hecho amar la ingeniería.

A mis amigos de ICAI, compañeros de fatigas durante estos cuatro años.

Índice

Índice 15
Índice de figuras 17
Índice de imágenes 17
Índice de tablas 18
Índice de gráficas 19
CAPÍTULO I: INTRODUCCIÓN 21
1.1 Introducción a los AVR 23
1.2 Estado de la cuestión 27
1.3 Motivación 28
1.4 Objetivos del proyecto 29
1.4 Metodología del trabajo 30
1.5 Estructura de la Memoria 31
CAPÍTULO II: DISEÑO 33
2.1 Introducción 35
2.2 Especificación Técnica 35
2.3 Diseño de Hardware 37
2.3.1 Módulo de introducción de referencia 37
2.3.2 Módulo de medida de corriente 37
2.3.3 Módulo de medida de tensión 37
2.3.4 Módulo de potencia 38
2.3.5 Resumen de módulos y componentes 42
2.4 Diseño de Software 44
2.4.1 Medida de intensidad 45
2.4.2 Medida de tensión 45
2.4.3 Control PID 45
2.4.4 Limitador de corriente 46
2.4.5 Monitorización y edición de parámetros en tiempo real 49
2.5 Conclusiones 50
CAPÍTULO III: ENSAYOS 51
3.1 Introducción 53
3.2 Montaje y aparatos utilizados 53

3.3 Calibración medida de tensión 57
3.4 Calibración medida de intensidad 59
3.5 Ajuste del control PID 60
3.5 Ensayo escalón de tensión 63
3.6 Ensayo seguimiento de referencia 64
3.7 Ensayo escalones de carga 65
3.8 Ensayo variación de la tensión de excitación 66
3.9 Ensayo variación de la velocidad 67
3.10 Comprobación del funcionamiento del limitador de corriente 68
3.11 Comprobación efecto frecuencia de conmutación 69
3.12 Conclusiones 71
CAPÍTULO IV: ESTUDIO ECONÓMICO 73
4.1 Introducción 75
4.2 Estimación de las horas de trabajo 75
4.3 Relación de tareas y profesionales 76
4.4 Coste de las horas de trabajo 77
4.5 Coste de los componentes y materiales 77
4.7 Resumen de costes 78
4.8 Conclusiones 78
CAPÍTULO V: CONCLUSIONES 79
5.1 Introducción 81
5.2 Conclusiones del proyecto 81
5.3 Futuros desarrollos 84
5.4 Conclusiones personales 87
CAPÍTULO VI: REFERENCIAS 89
6.1 Libros, artículos, revistas y otros proyectos 91
6.2 Páginas web y Blogs 91
ANEXOS 93
Anexo I: Esquemas eléctricos 95
Anexo II: Código fuente 99
Arduino 99
Processing 105
Código adicional Arduino 114

Anexo III: Fotografías 115
Anexo IV: Datasheets 119

Índice de figuras

FIGURA 1|ESQUEMA DEL SISTEMA ELÉCTRICO 23
FIGURA 2|MÁQUINA SÍNCRONA (IZDA.) VS. MÁQUINA DE INDUCCIÓN. ASPECTOS CONSTRUCTIVOS 24
FIGURA 3|EFECTO BRÚJULA 24
FIGURA 4|INTRODUCCIÓN DE LA REFERENCIA DE TENSIÓN 37
FIGURA 5|CIRCUITO DE MEDIDA DE INTENSIDAD 37
FIGURA 6|CIRCUITO DE MEDIDA DE TENSIÓN 38
FIGURA 7|ANTIGUO CIRCUITO DE POTENCIA 38
FIGURA 8|CIRCUITO DE POTENCIA REDISEÑADO 39
FIGURA 9| ESQUEMA DE CONEXIÓN COMPLETO DEL REGULADOR 43
FIGURA 10|DIAGRAMA DE BLOQUES DEL REGULADOR 44
FIGURA 11|CIRCUITO SIMPLIFICADO DE LA TARJETA DE POTENCIA 47
FIGURA 12|ESQUEMA ENSAYOS 54
FIGURA 13|SISTEMA DE ALIMENTACIÓN POR ESCALONES AUTOMÁTICO 85
FIGURA 14|SISTEMA DE ALIMENTACIÓN POR ESCALONES AUTOMÁTICO (POSICIONES) 86

Índice de imágenes

IMAGEN 1| REGULADORES AUTOMÁTICOS DE TENSIÓN UNICONTROL 1000 DEL FABRICANTE ABB. AVR
DIGITALES CON ETAPA DE POTENCIA BASADA EN CHOPPERS IGBT PARA PEQUEÑAS MÁQUINAS SÍNCRONAS 25
IMAGEN 2|MUESTRA DE CIRCUITLAB 40
IMAGEN 3|CORRIENTE POR EL TRANSISTOR (AZUL) Y POR LA BOBINA (AMARILLO). FACTOR DE SERVICIO 0.3 48
IMAGEN 4|CORRIENTE POR EL TRANSISTOR (AZUL) Y POR LA BOBINA (AMARILLO). FACTOR DE SERVICIO 0.5 48
IMAGEN 5|CORRIENTE POR EL TRANSISTOR (AZUL) Y POR LA BOBINA (AMARILLO). FACTOR DE SERVICIO 0.8 49
IMAGEN 6|INTERFAZ PARA MONITORIZACIÓN Y AJUSTE DEL REGULADOR 50
IMAGEN 7|FOTOGRAFÍA ENSAYOS 55
IMAGEN 8|AJUSTE PID (1,0,0) 60
IMAGEN 9|AJUSTE PID (0.8,0,0) 61
IMAGEN 10|AJUSTE PID (0.8,0.1,0) 61

IMAGEN 11|AJUSTE PID (0.8,0,0.1) 62

IMAGEN 12|RESPUESTA A UN ESCALÓN DE 220V 63

IMAGEN 13|ENSAYO SEGUIMIENTO DE REFERENCIA 64

IMAGEN 14|ENSAYO ESCALONES DE CARGA DE 1kW DE 0 A 4kW Y DE 4 A 0kW, 220V 65

IMAGEN 15|ENSAYO VARIACIÓN DE LA TENSIÓN DE EXCITACIÓN 66

IMAGEN 16|ENSAYO VARIACIÓN DE LA VELOCIDAD 67

IMAGEN 17|COMPROBACIÓN DEL FUNCIONAMIENTO DEL LIMITADOR DE CORRIENTE 68

IMAGEN 18|CORRIENTE DEL ROTOR CON CONMUTACIÓN A 490Hz 69

IMAGEN 19|CORRIENTE DEL ROTOR CON CONMUTACIÓN A 490Hz (PROMEDIADO) 69

IMAGEN 20|CORRIENTE DEL ROTOR CON CONMUTACIÓN A 3921Hz 70

IMAGEN 21|CORRIENTE DEL ROTOR CON CONMUTACIÓN A 3921Hz (PROMEDIADO) 70

IMAGEN 22|FOTOGRAFÍA CONJUNTO 81

IMAGEN 23|INTERFAZ PROCESSING 82

IMAGEN 24|ENSAYOS 82

Índice de tablas

TABLA 1|CARACTERÍSTICAS NOMINALES DE LA MÁQUINA SÍNCRONA DE DISEÑO 35

TABLA 2| CARACTERÍSTICAS DE LA PLACA ARDUINO UNO 36

TABLA 3|RESUMEN DE MÓDULOS Y COMPONENTES DEL REGULADOR 42

TABLA 4|CARACTERÍSTICAS DE LA MÁQUINA SÍNCRONA (ENSAYOS) 53

TABLA 5|CARACTERÍSTICAS DE LA MÁQUINA DE INDUCCIÓN (ENSAYOS) 54

TABLA 6|CALIBRACIÓN DE LA MEDIDA DE TENSIÓN 58

TABLA 7|MEDIDA DE INTENSIDAD 59

TABLA 8|HORAS DESARROLLO Y DISEÑO 75

TABLA 9|HORAS FABRICACIÓN E INSTALACIÓN 76

TABLA 10|RELACIÓN DE TAREAS Y PROFESIONALES 76

TABLA 11|COSTE DE LAS HORAS DE TRABAJO 77

TABLA 12|COSTE DE LOS COMPONENTES DEL REGULADOR 77

TABLA 13|RESUMEN DE COSTES 78

TABLA 14|CARACTERÍSTICAS DEL REGULADOR AUTOMÁTICO DE TENSIÓN 83

TABLA 15|CARACTERÍSTICAS NOMINALES DE LA MÁQUINA DE DISEÑO (BIS) 83

TABLA 16|EJEMPLO DE MÁQUINA SÍNCRONA VÁLIDA 72

Índice de gráficas

GRÁFICA 1|ERROR EN LA MEDIDA DE TENSIÓN 57

GRÁFICA 2|VERIFICACIÓN TRAS CALIBRACIÓN DE MEDIDA DE TENSIÓN 58

CAPÍTULO I: INTRODUCCIÓN

Este capítulo servirá para dar al lector un contexto y una motivación de este proyecto. Presenta al lector los AVR, revisa las características de los AVR de los principales fabricantes, justifica la realización de este proyecto, marca los objetivos principales y expone la metodología de trabajo seguida.

1.1 Introducción a los AVR

Este proyecto consiste en el perfeccionamiento de un control automático de tensión para máquina síncrona mediante un rediseño del hardware y implementación de un control PID en el apartado del software.

Este primer apartado pretende servir de **introducción al proyecto**, situando el contexto en el que este se realiza.

Las máquinas síncronas son el pilar tecnológico sobre el cual se sustenta la sociedad actual. No se contempla la posibilidad de haber llegado al actual nivel de desarrollo sin la existencia de la generación, transporte, distribución y transformación de la energía eléctrica. Las máquinas síncronas se localizan precisamente en el primer eslabón de esta cadena energética, transformando distintas fuentes de energía en energía eléctrica, una energía fácil de transportar y con la capacidad de transformarse de nuevo en todo tipo de energía: cinética, calorífica, lumínica entre otras muchas.

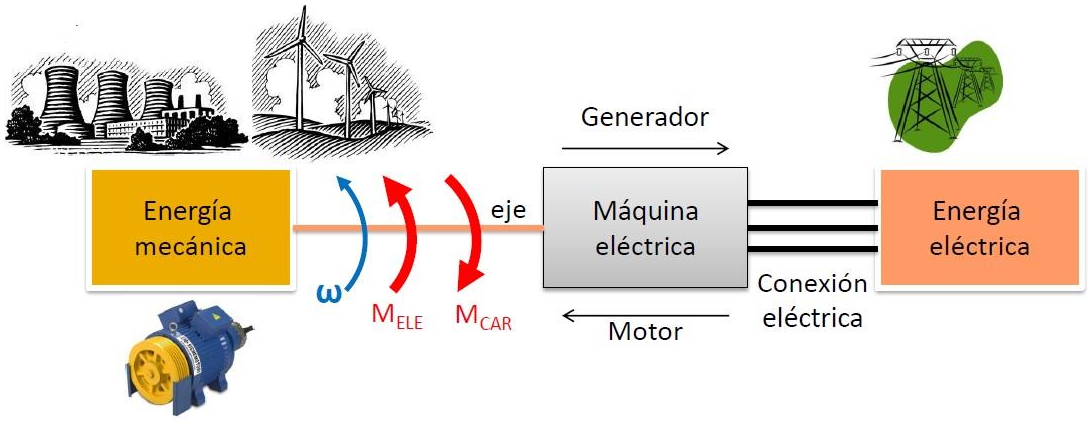


FIGURA 1|ESQUEMA DEL SISTEMA ELÉCTRICO [1]

Una máquina síncrona es una máquina rotativa que convierte la energía mecánica que se le aporta a su eje en energía eléctrica por medio de la interacción entre dos campos magnéticos.

La máquina síncrona no es el único tipo de máquina rotativa. En el ámbito de la generación eléctrica encontramos en realidad dos tipos de máquinas principalmente: las máquinas síncronas y las máquinas de inducción.

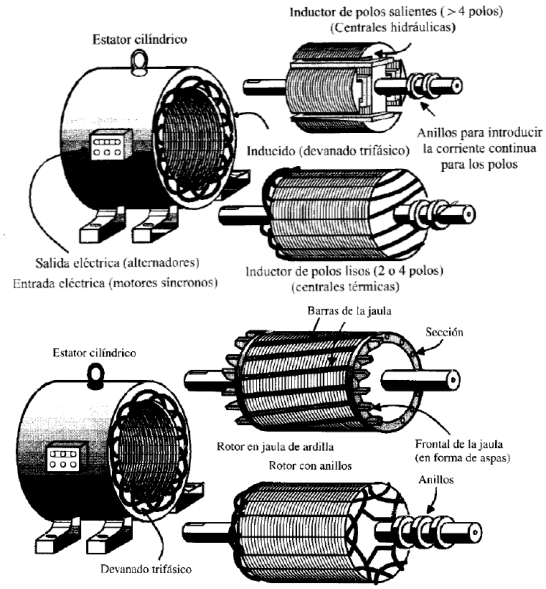


FIGURA 2|MÁQUINA SÍNCRONA (IZDA.) VS. MÁQUINA DE INDUCCIÓN. ASPECTOS CONSTRUCTIVOS [2]

La principal diferencia entre las máquinas síncronas y las máquinas de inducción (o asíncronas) radica en el rotor. Las máquinas de inducción carecen de alimentación al rotor (suelen ser rotores devanados en cortocircuito o "jaulas de ardilla"), y el campo magnético de este se crea por inducción desde el estator. Esto hace que las máquinas de inducción necesiten estar acopladas a la red para funcionar como generador. Las máquinas síncronas por el contrario si tienen alimentación en el rotor por medio de un circuito llamado circuito de excitación. Esta excitación es la que produce el campo magnético del rotor que al interactuar con el del estator (por medio del conocido como "efecto brújula", que explica la tendencia de dos campos magnéticos a alinearse produciendo par si existe un ángulo entre ellos, efecto que aprovechan todas las máquinas rotativas) convierte la energía mecánica en energía eléctrica. La corriente de excitación puede provenir desde baterías hasta de la propia corriente que genera la máquina (máquina síncrona autoexcitada). En algunos casos el campo magnético del rotor puede ser producido incluso por imanes permanentes, aunque esto no es típico para potencias altas. Esta disposición hace que las maquinas síncronas sean capaces de generar de forma independiente o "en isla", lo que significa que no necesitan estar acopladas a una red como las máquinas de inducción.

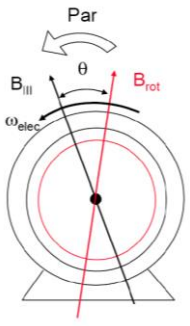


FIGURA 3|EFECTO BRÚJULA

Otra diferencia fundamental entre los dos tipos de máquinas es la velocidad de rotación. La razón por la que las máquinas de inducción son también conocidas como máquinas asíncronas es que, para dar par como motor o para producir potencia activa como generador, el rotor gira a una velocidad distinta a la velocidad de sincronismo (frecuencia de la red). Esto se debe a que su capacidad de dar o transformar par mecánico depende de la diferencia de velocidad entre la velocidad del rotor y la velocidad de sincronismo. Si no existiera esta diferencia de velocidad, denominada deslizamiento, no se inducirían corrientes en el rotor ya que sus espiras no verían una variación de flujo magnético y por tanto desaparecería el campo magnético del rotor. En el caso de las máquinas síncronas esto no sucede, ya que el campo magnético del rotor es generado por la intensidad inyectada al rotor a través del sistema de excitación. En este caso el rotor gira síncronamente y el par electromagnético se debe al ángulo que hay entre los campos del rotor y el estator, denominado ángulo de carga. Este ángulo, junto con la magnitud del campo del rotor (proporcional a la intensidad de excitación), determinan las condiciones de funcionamiento de la máquina.

La mayor parte de centrales de generación de energía eléctrica utilizan las máquinas síncronas y de su correcto control y regulación depende en gran medida la estabilidad y calidad del suministro eléctrico.

El regulador automático de tensión, conocido también como **AVR** por sus siglas en inglés, es un elemento de electrónica de potencia que se encarga de mantener la tensión en bornes de la máquina constante e igual a la tensión de consigna, independientemente de las variaciones de demanda de potencia activa y reactiva y demás factores externos que pudieran alterarla. El regulador de tensión actúa sobre el circuito de excitación para llevar a cabo esta tarea, regulando la cantidad de corriente de excitación y, en consecuencia, la magnitud del campo magnético del rotor.



IMAGEN 1| REGULADORES AUTOMÁTICOS DE TENSIÓN UNITROL 1000 DEL FABRICANTE ABB. AVR DIGITALES CON ETAPA DE POTENCIA BASADA EN CHOPPERS IGBT PARA PEQUEÑAS MÁQUINAS SÍNCRONAS [3]

Este proyecto ha dado continuidad al proyecto realizado anteriormente por Francisco de Asís Salas Igea[4], quien creó un primer prototipo de un AVR de bajo presupuesto. El prototipo del señor de Asís comprendía en el apartado de control una placa **Arduino Uno** programada con un control proporcional y en el apartado de potencia un puente rectificador de diodos y un chopper para controlar la corriente de excitación basado en un transistor IGBT. Este prototipo superó los ensayos de regulación de tensión en vacío y en carga. El prototipo también incluía una limitación de corriente. Cuando la corriente superaba un valor determinado el control bajaba la tensión de excitación, ignorando lo que el control PID pidiera, hasta que la intensidad de excitación estuviera por debajo del nivel aceptable. Lo que se ha hecho en este proyecto es revisar el diseño del hardware para implementar mejoras en aspectos de robustez, seguridad, flexibilidad y fiabilidad. Además, en el apartado del software, se ha diseñado e implementado satisfactoriamente un control PID que permite aumentar la rapidez y precisión del AVR. También se ha programado una interfaz que permite un rápido ajuste de los parámetros del control y una monitorización en tiempo real de las variables del control a través de un ordenador, lo cual facilita enormemente la utilización y comprensión del AVR.

1.2 Estado de la cuestión

La importancia de las máquinas síncronas y su correcto control ya se ha discutido previamente en la introducción anterior. En este apartado se dará una breve pincelada del estado del arte de esta tecnología.

La regulación de tensión podría teóricamente hacerse de forma manual. Una persona estaría encargada de medir constantemente la tensión en bornes de la máquina síncrona y regular la intensidad de excitación por medio de, por ejemplo, resistencias variables para mantener esa tensión siempre dentro de un umbral. Esto en la vida real es impráctico y lo que se utiliza son los reguladores automáticos de tensión.

Inicialmente los AVR llegaron a tener elementos mecánicos, lo que hacía que sus tiempos de respuesta fueran muy grandes y su mantenimiento muy costoso. Con la llegada de la electrónica los AVR pasaron a utilizar electrónica analógica y actualmente lo que prima son los AVR digitales que permiten muchas funciones adicionales además de la regulación de la tensión.

Haciendo un repaso a lo que fabricantes como **Basler, ABB, General Electric y Alstom** ofrecen vemos que los AVR digitales actuales presentan comúnmente las siguientes funciones adicionales:

- Arranque suave.
- Limitador de mínima y máxima corriente de excitación.
- Limitador de corriente máxima de estator
- Protección V/f (flujo).
- Control del factor de potencia y potencia reactiva.
- Protección de sobretensión y subtensión.
- Estabilización del sistema de potencia (PSS).
- Sincronización.
- Control PID con parámetros ajustables.

Existen modelos específicos para los distintos tipos de sistemas de excitación: excitación independiente, generadores autoexcitados, excitación por máquina DC, excitación por máquina AC (brushless)…

1.3 Motivación

Este apartado pretende explicar cuáles son los motivos que impulsaron a la realización de este proyecto.

La motivaciones principales de este proyecto fueron, desde el punto de vista **académico**, el realizar un diseño técnico para más tarde implementarlo, atravesando las distintas fases que conlleva un proyecto técnico y sorteando las dificultades que pueden surgir al pasar "del papel", que lo aguanta todo, a la implementación en prototipos reales.

Desde el punto de vista **práctico**, realizar una aportación al Laboratorio de Máquinas de ICAI, ya que estos reguladores podrán ser utilizados en diversas prácticas para realizar una gran variedad de ensayos que actualmente solo pueden ser realizados mediante simulación por ordenador. Estos ensayos en máquinas reales ayudarán a entender los principios de la regulación de tensión en los generadores síncronos en el sistema eléctrico.

No solo se trata de una herramienta de laboratorio. El regulador es capaz de manejar máquinas síncronas de hasta típicamente 15 kVA (puede variar en función de las características de la excitación de la máquina en cuestión), lo cual correspondería a un pequeño grupo electrógeno que podría ser utilizado para obras, puestos ambulantes, uso doméstico... Además, este diseño podría ser fácilmente modificado para controlar máquinas de mayor potencia o para controlar potencia reactiva en máquinas conectadas a la red.

Por último también cabe destacar que se está trabajando en una plataforma "open source" como es Arduino. Los componentes que conforman el regulador son relativamente baratos y fáciles de adquirir y por tanto todo el conjunto es accesible, fácil de replicar y abierto a futuras mejoras.

1.4 Objetivos del proyecto

Ya se han ido mencionando las metas que se propusieron para este proyecto, pero en este apartado se recopilan y reformulan de forma clara. Los objetivos del proyecto han sido:

- Rediseñar el hardware del AVR para implementar mejoras en aspectos de robustez, seguridad, flexibilidad y fiabilidad.

- Diseñar e implementar un control PID para el prototipo de AVR para aumentar la precisión y rapidez con respecto al control iterativo.

- Programar una interfaz que permita un rápido ajuste de los parámetros del control y una monitorización en tiempo real de las variables del control utilizando un ordenador.

- Realizar un análisis económico del proyecto.

- Identificación de posibles mejoras y ampliaciones.

1.4 Metodología del trabajo

El esquema de trabajo general para cualquier proyecto de diseño es el siguiente:

- **Especificación técnica.** Consiste en analizar las características de la máquina eléctrica a controlar y de la placa Arduino que se encargará del control. Es necesario conocer las características nominales y los límites de la máquina eléctrica para dimensionar tanto el circuito de medida como el de potencia, datos como intensidades y tensiones nominales o límites de corriente de estator y rotor. Como además estos dos circuitos tienen que interactuar con la placa Arduino, y ésta también tiene sus límites en cuanto a tensiones y corrientes que es capaz de leer o proporcionar en sus pines, hace falta conocer y tener en cuenta estos valores para crear una interfaz entre los circuitos de potencia y medida y Arduino.

- **Diseño de Hardware.** Diseñar los distintos circuitos que conforman el sistema. Una vez conocidos los límites y necesidades de los dos principales elementos (máquina eléctrica y placa Arduino) se deben diseñar los circuitos de medida y potencia atendiendo también a otros factores como disponibilidad, costes, precisión necesaria...

- **Diseño del software.** Este apartado comprende la elección y diseño del control. Se deben diseñar las decisiones que el Arduino tomará en función de las medidas, que deberán ser previamente interpretadas. Por medio de software se programará no solo el control si no también protecciones como un limitador de corriente y herramientas que faciliten la interacción con el usuario como un monitor en tiempo real a través del puerto serie de Arduino utilizando un ordenador que ayudarán al usuario a ajustar y comprobar el funcionamiento del control.

- **"Set up".** Este es el apartado final. Consiste en realizar todos los montajes necesarios, realizar los diferentes ensayos de comprobación y calibración y los ensayos finales. Las primeras comprobaciones se realizan utilizando tensiones e intensidades reducidas. Para comprobar que la adquisición de medidas es correcta se utilizan elementos como Variacs y resistencias para medir tensiones e intensidades controladas y de valores menores a los límites para prevenir incidentes en caso de conexiones o diseño incorrectos. Una vez comprobado el correcto funcionamiento del prototipo se realizan los ensayos necesarios para ajustar los parámetros de medida y control. Por último se realizan los ensayos finales que sirven de comprobación y demostración una vez finalizadas las calibraciones y de los cuales se sacan las conclusiones finales.

1.5 Estructura de la Memoria

Este documento consta de la siguiente estructura:

La memoria está dividida en siete capítulos.

- En el primer capítulo, dividido en seis apartados, se pretende introducir al lector en el proyecto para facilitar la comprensión del mismo. Presenta al lector los AVR, revisa las características de los AVR de los principales fabricantes, justifica la realización de este proyecto, marca los objetivos principales, expone la metodología de trabajo seguida y enumera los recursos empleados.

- El siguiente capítulo es el de diseño propiamente dicho. Se divide en especificación técnica, diseño de hardware y diseño de software.

- El tercer capítulo recoge los distintos ensayos realizados mostrando el funcionamiento del prototipo.

- A continuación se presenta el capítulo del estudio económico, donde se realiza un estudio sobre el coste de producción del regulador que se ha diseñado.

- El quinto capítulo resume lo extraído de este proyecto y los resultados obtenidos además de repasar las posibles mejoras y ampliaciones para el futuro a modo de conclusión.

- El sexto capítulo contiene las referencias del documento.

- Por último, se presentan los anexos, donde se pueden encontrar los catálogos, esquemas detallados, fotos del prototipo, programación y detalles técnicos que permitan replicar el proyecto en el futuro.

CAPÍTULO II: DISEÑO

2.1 Introducción

En este capítulo se detallará el proceso de diseño del regulador. Empezará por la especificación técnica para identificar las necesidades, requerimientos y límites del regulador. A continuación se presentará el diseño del hardware: la placa de medida de tensión, la placa de medida de intensidad y la placa de potencia. Por último se explicará el diseño del software, incluyendo tanto las funciones específicas de medida como el propio control PID y el limitador de corriente. Este apartado también incluye el diseño de la interfaz de monitorización y calibración del control en tiempo real.

Para la realización de todos los esquemas eléctricos de éste y el resto de capítulos se ha utilizado la aplicación web **CircuitLab**[9], que es una herramienta online de diseño y simulación de circuitos eléctricos y electrónicos.

2.2 Especificación Técnica

Las características de la máquina síncrona que se ha utilizado como base para desarrollar el regulador se resumen en la siguiente tabla:

CARACTERÍSTICAS NOMINALES DE LA MÁQUINA SÍNCRONA DE DISEÑO	
FABRICANTE	LETAG
TENSIÓN	230 V
INTENSIDAD	12.5 A
POTENCIA	5 kVA
FRECUENCIA	50 Hz
VELOCIDAD	1500 r.p.m.
FACTOR DE POTENCIA	0.8-1
EXCITACIÓN	
INTENSIDAD EN VACÍO	1.1 A
INTENSIDAD NOMINAL	4.3 A
TENSIÓN NOMINAL	50 V

TABLA 1|CARACTERÍSTICAS NOMINALES DE LA MÁQUINA SÍNCRONA DE DISEÑO

Por tanto, el regulador tendrá que ser capaz de medir hasta 230 V y 4.3 A como mínimo. La etapa de potencia tendrá que ser capaz de soportar 4.3 A y choppear hasta 50 V.

El regulador debe ser capaz de mantener la tensión de estator constante igual a la de consigna, independientemente de las variaciones de carga y otros factores externos, pero asegurar que nunca se supere la corriente máxima de excitación.

Además, el diseño del control se basa en la placa Arduino Uno, cuyas características relevantes se resumen a continuación en una tabla extraída de la página oficial de Arduino:

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

TABLA 2| CARACTERÍSTICAS DE LA PLACA ARDUINO UNO

Por tanto vemos que no podemos extraer más de 20mA de los pines del microprocesador, que el sistema tendrá que operar a 5 V y que la velocidad con la que el procesador es capaz de tomar decisiones es de 16MHz (62.5ns). La memoria no es problema si se utiliza un código sencillo. Se consideraron otras placas Arduino más potentes como la Due o la Mega, pero son más caras y no son necesarias sus capacidades adicionales. A pesar de esto, la programación es común a todas las placas y, por tanto, se podrían utilizar cualquiera de estas placas con tan solo cargar el programa en ellas.

2.3 Diseño de Hardware

El diseño del hardware consiste en diseñar los diferentes circuitos electrónicos y eléctricos que va a necesitar el regulador. Necesitamos una forma de introducir la consigna de tensión, una forma de medir la tensión para poder compararla con la consigna, una forma de medir la intensidad para poder limitar la corriente y una forma de regular la corriente de excitación.

2.3.1 Módulo de introducción de referencia

Se decide mantener el sistema que se utilizó en el proyecto de referencia [4]. Consiste en un potenciómetro conectado al microprocesador, el cual en función del valor de tensión que mida asociará un valor de tensión entre 0 y 230 V.

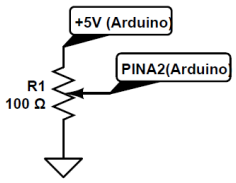


FIGURA 4|INTRODUCCIÓN DE LA REFERENCIA DE TENSIÓN

2.3.2 Módulo de medida de corriente

Consiste en un transductor de corriente que proporcionará una señal de tensión aislada y proporcional a la corriente que atraviese su primario. Es necesario que pueda medir corrientes superiores a 4.3 A. Además no debe superar los 5 V cuando se mida la corriente más alta esperada. Se decide utilizar un transductor distinto ya que el que se había utilizado anteriormente estaba estropeado. El nuevo transductor de corriente tiene un corriente nominal de 30A y una sensibilidad de 66mV/A.

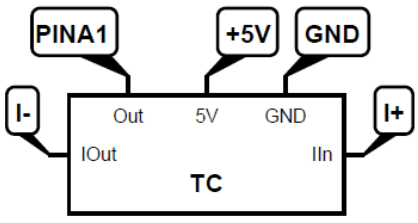


FIGURA 5|CIRCUITO DE MEDIDA DE INTENSIDAD

2.3.3 Módulo de medida de tensión

La placa que se utilizó en el proyecto de referencia [4] ha demostrado ser muy válida y por tanto no se ve la necesidad de diseñar una nueva. A continuación se detalla el funcionamiento y características:

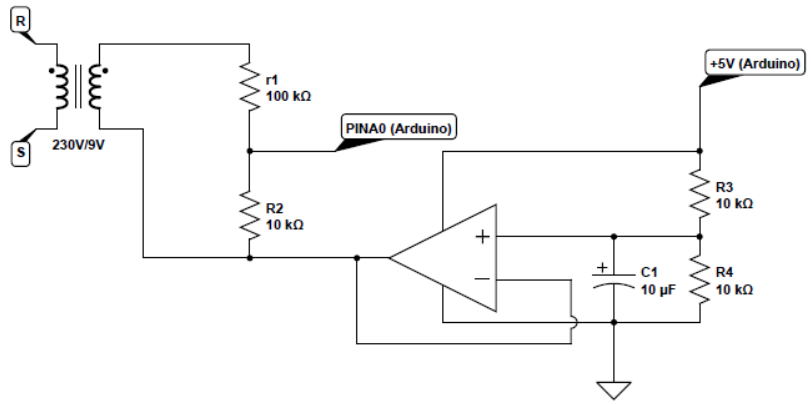


FIGURA 6|CIRCUITO DE MEDIDA DE TENSIÓN

Como se puede observar consta en primer lugar de un transformador que reduce la tensión de 230 V a 9 V. Se añade un divisor resistivo para reducir la tensión aún más (1/11), de forma que nunca se supere el límite de 5 V de Arduino. Además, como Arduino no puede leer señales negativas en sus pines, se añade un buffer de tensión de 2,5 V para que la senoidal quede centrada entre 0 y 5 V. El buffer se construye con un divisor resistivo alimentado por el propio Arduino y un amplificador operacional también alimentado por Arduino.

2.3.4 Módulo de potencia

Este módulo es lo que representa, a nivel de hardware, el mayor cambio con respecto al prototipo anterior[4]. Se decide rediseñar por completo la tarjeta de potencia con el fin de conseguir una mayor robustez y una mayor frecuencia de conmutación. Además el transistor de la tarjeta anterior se encontraba estropeado y al sustituirlo por el repuesto volvió a estropearse al poco tiempo. Por este motivo no ha sido posible realizar una comparación directa entre el funcionamiento de las dos tarjetas.

El diseño anterior[4] consistía sencillamente en un puente rectificador de diodos de doble onda, un transistor controlado directamente por Arduino y un diodo de libre circulación. A continuación se puede ver el esquema del circuito:

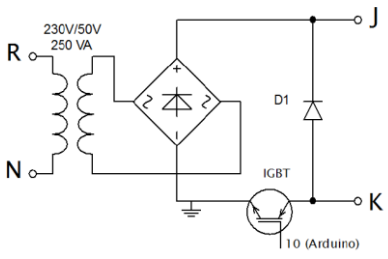


FIGURA 7|ANTIGUO CIRCUITO DE POTENCIA

Este circuito es en principio viable y demostró ser suficiente, sin embargo el transistor tenía una frecuencia de conmutación máxima de 1 kHz y estaba siendo usado a 490Hz, lo cual parece algo justo para esta aplicación. Además era disparado directamente por Arduino, el cual solo tiene 5 V y 10-20mA como tensión e intensidad máxima en sus pines, lo que no es suficiente tensión para saturar un transistor IGBT y reducir las pérdidas al mínimo ni suficiente corriente para cargar rápidamente la capacidad de la puerta.

Por otro lado el diodo de libre circulación era de propósito general por lo que su tiempo de recuperación era de 2.5 microsegundos lo cual resulta lento para evitar las sobretensiones debidas a la apertura de un circuito inductivo como es el rotor de una máquina síncrona.

Otro detalle es que en este diseño la onda que el transistor estaba choppeando era una onda doble rectificada, no una onda de tensión continua. En principio esto no plantea un problema ya que la alta inductancia del rotor hace que la corriente sea suficientemente continua, pero es muy sencillo añadir un condensador a la salida del puente para conseguir una tensión más continua.

Por último, el transistor en este diseño no tiene ningún tipo de protección ante tensiones negativas por lo que es más propenso a averiarse.

Teniendo todo esto en cuenta, se diseña una nueva tarjeta de potencia:

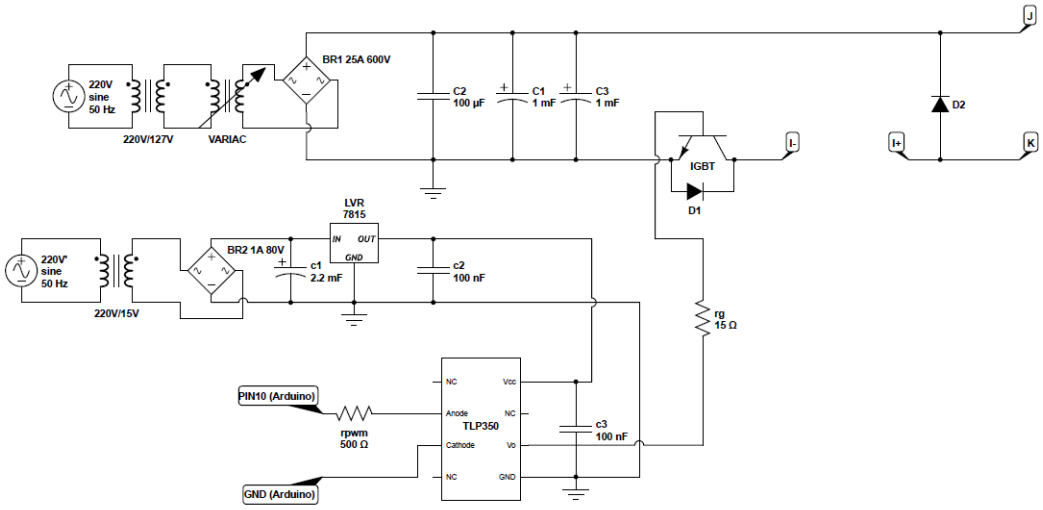


FIGURA 8|CIRCUITO DE POTENCIA REDISEÑADO

El primer elemento, al igual que en el diseño anterior, es un transformador que nos dé una tensión adecuada para la excitación de nuestra máquina. En nuestro caso, nuestra máquina tiene una tensión nominal de excitación de 50 V, por tanto una buena relación de transformación podría ser 220V/35V ($35 \cdot \sqrt{2} \approx 50$). Para mayor comodidad a la hora de realizar los ensayos se ha utilizado un transformador 220V/127V para aislar galvánicamente y un VARIAC para poder ajustar la tensión de excitación. De esta forma se puede ensayar en varias máquinas o comprobar la reacción del control ante un cambio en la tensión de excitación. La tensión alterna se rectifica en el puente de diodos BR1, el cual tiene una capacidad equivalente de 2mF a su salida (dos condensadores de 1mF, 100V en paralelo). Se ha elegido una capacidad

de este tamaño para poder alimentar con hasta 4A el rotor sin que la tensión deje de tener un rizado pequeño. *

El transistor elegido en este caso ha sido un **Infineon IKW08T120**. Es capaz de conducir 8A, soportar una tensión colector-emisor de 1200V y conmutar hasta a 20kHz. Además incluye un diodo rápido en antiparalelo que lo protege contra tensiones colector-emisor negativas. El diodo de libre circulación para el rotor se ha cambiado por un **IXYS DSEI 12-10A** que es mucho más rápido que el anterior (de 2.5µs de tiempo de recuperación hemos pasado a 50ns) con lo que limita mucho mejor las sobretensiones. Soporta 12A y 1000V.

Como se puede apreciar en el esquema anterior, la tarjeta no acaba ahí. Hay todo un circuito adicional que cumple la función de disparar el transistor correctamente. Este circuito recibe el nombre de "driver".

El transistor utilizado tiene una tensión de saturación de mínimo 5 y máximo 6.5 V. Además el valor típico que se utiliza en los ejemplos de su ficha técnica como tensión puerta-emisor es de 15 V. La placa Arduino puede proporcionar un máximo de 5 V. Por otro lado el transistor IGBT tiene una capacidad que hay que cargar al dispararlo. En este caso tiene una capacidad puerta-emisor de 564pF y una carga de la puerta de 53nC. Como la frecuencia de conmutación es de aproximadamente 4kHz para nuestro diseño, la intensidad media que consume la puerta es de aproximadamente 0,2mA. Sin embargo es recomendable que la corriente de pico esté en el orden de 1A. Los pines de Arduino solo pueden proporcionar 20mA en el mejor de los casos.

---

*El dimensionamiento de los condensadores a la salida de ambos puentes de diodos se ha llevado a cabo utilizando la ya mencionada aplicación web CircuitLab[9] mediante simulación, ajustando la capacidad hasta conseguir el rizado de tensión deseado.

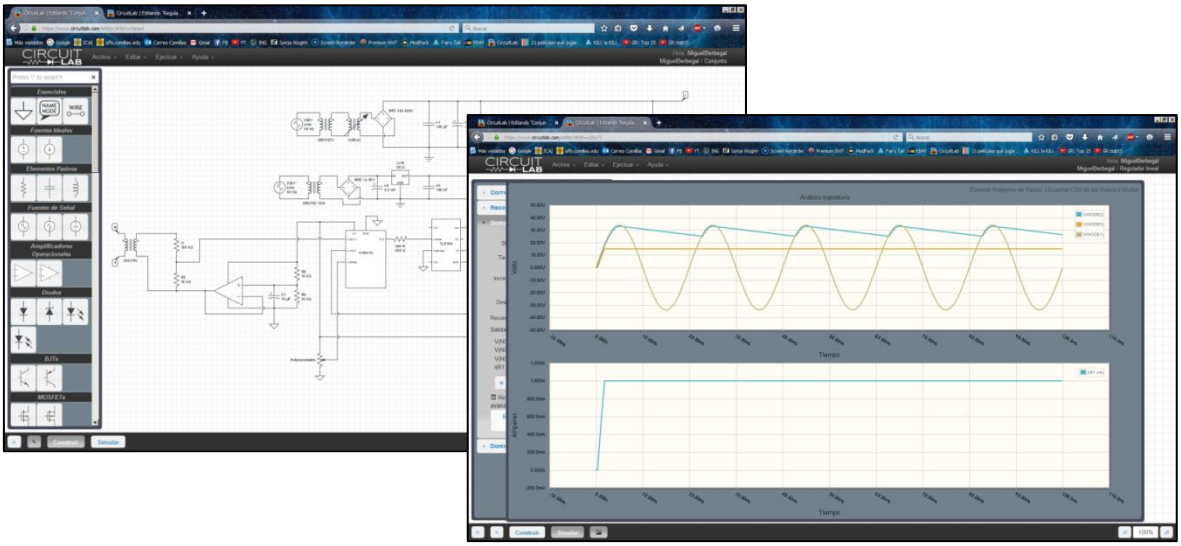


IMAGEN 2|MUESTRA DE CIRCUITLAB

Por estos dos motivos (tensión de disparo del transistor y corriente de pico) se decide utilizar un driver para disparar el transistor en vez de Arduino directamente. Un driver es un circuito que permite controlar la puerta de un transistor con la tensión y corriente deseadas, en este caso con 15 V y con 1A de pico. Revisando distintos drivers comerciales nos encontramos con que incluso los más sencillos y económicos ofrecen muchas más funciones de las que nosotros necesitamos y son por tanto más complejos que uno que nosotros pudiéramos diseñar, encareciendo el prototipo y complicándolo de forma innecesaria. Por ello, basándonos en un optoacoplador de **TOSHIBA** (**TLP350**) que viene con un secundario especialmente adaptado para disparar transistores IGBT, se ha diseñado un driver adaptado a nuestros requerimientos. Este optoacoplador puede proporcionar hasta 2,5 A de corriente de pico para disparar el transistor, funcionar con tensiones de 15 V a 30 V, puede operar hasta a 50kHz y tan solo necesita 10mA en el primario para funcionar, con lo que es perfecto para funcionar en conjunto con nuestro microprocesador y nuestro transistor.

Se diseña una fuente de alimentación para conseguir 15 V de continua. El diseño de la fuente es estándar: un transformador, un puente rectificador de diodos, un condensador y un regulador lineal de tensión. Esta fuente de alimentación utiliza los 220V de la red con un enchufe domestico para mayor comodidad. Una vez conseguidos los 15V ya solo falta añadir una resistencia de 15Ω a la puerta para limitar la corriente de pico a 1A como deseábamos y una resistencia de 500Ω en el primario del optoacoplador para conseguir los 10mA necesarios de la salida PWM de Arduino. Por motivos de estabilidad y recomendación del fabricante también se han añadido condensadores de 100nF a la salida del regulador de tensión.

El funcionamiento de nuestro driver es sencillo: la señal PWM generada por nuestra placa Arduino se convierte en pulsos de luz en el primario del optoacoplador. Estos pulsos de luz abren y cierra el fototransistor del secundario del optoacoplador, dejando pasar los 15V de continua conseguidos con nuestra fuente hacia la puerta del transistor IGBT. Gracias a la resistencia puesta en la salida del optoacoplador se garantiza que la corriente de pico no sea superior a 1A.

Para ejemplificar la diferencia de precio y complicación de nuestro driver en comparación con uno comercial, se toma como referencia el driver **POWEREX M57962L,** basado también en optoacopladores, con un coste de aproximadamente 30€ frente a los menos de 5€ que ha costado el nuestro. La hoja de características del driver comercial se puede encontrar aquí.

2.3.5 Resumen de módulos y componentes

La tabla a continuación muestra una lista de los componentes agrupados según el módulo al que pertenecen:

Módulo	Componentes
Medida	Resistencia 10kΩ(x3)
	Resistencia 100kΩ
	Condensador 10µF
	Amplificador Operacional UA 741
	Transformador 230 V/9 V
	Transductor 30A 0.066V/A
Referencia	Potenciómetro
Control	Placa Arduino UNO
Potencia	Resistencia 500Ω (>10mA)
	Resistencia 15Ω (>0.2mA)
	Condensador cerámico 100nF(x2), 50Vdc
	Condensador electrolítico 1000µF, 100Vdc (x2)
	Condensador cerámico 100µF, 1000Vdc
	Condensador electrolítico 2200µF, 63Vdc
	Optoacoplador TOSHIBA TLP350 2.5A pico, 30V
	Transistor IGBT Infineon IKW08T120 1200V, 8A, 20kHz
	Regulador lineal de tensión 7815CV
	Puente rectificador de diodos 1, 25A, 600V
	Puente rectificador de diodos 2 1A, 80V
	Diodo rápido 1000V, 12A, 50ns
	Transformador 220V/15V
	Transformador 220V/127V+VARIAC

TABLA 3|RESUMEN DE MÓDULOS Y COMPONENTES DEL REGULADOR

El esquema de todos los módulos conectados entre sí:

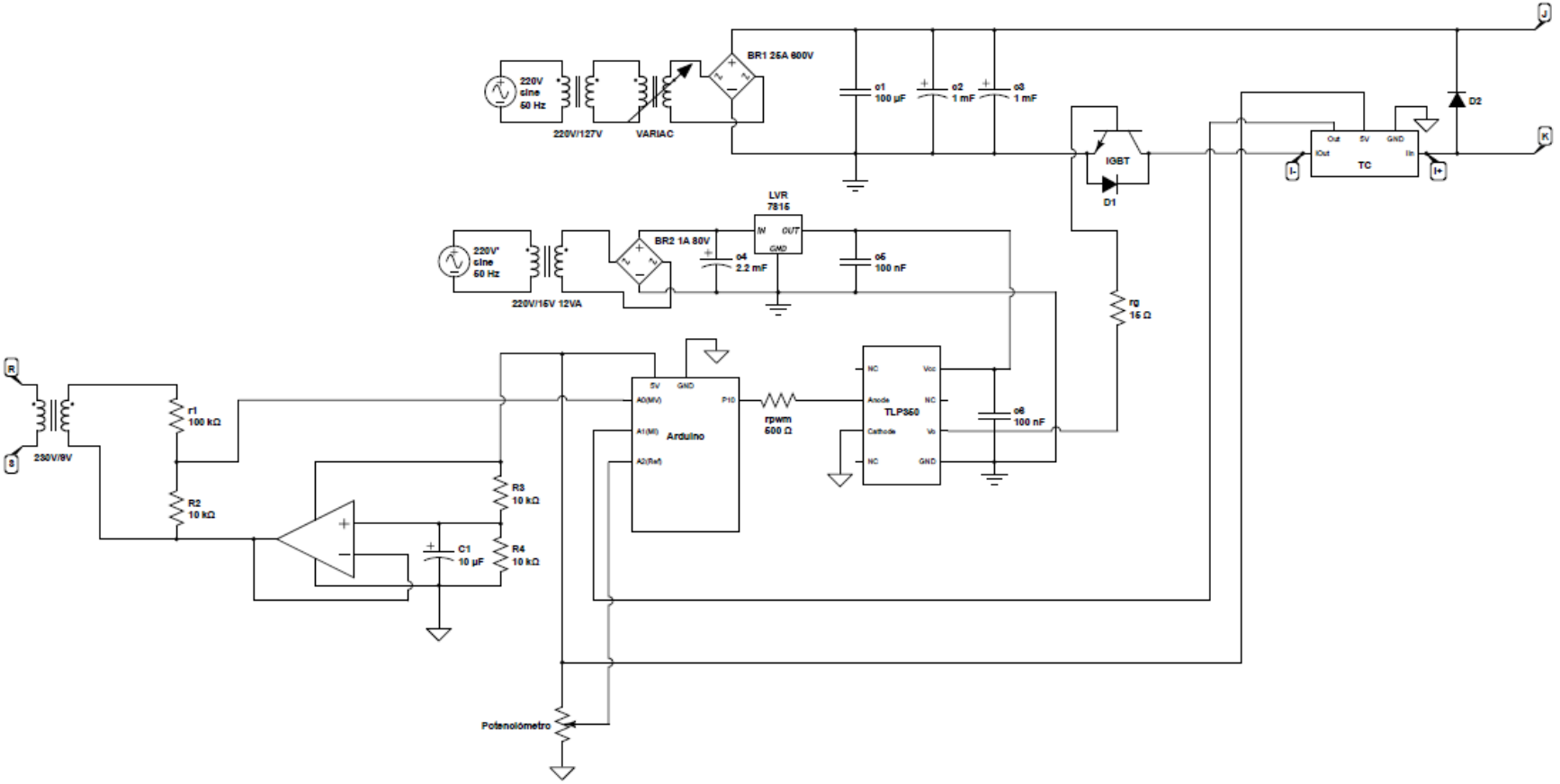


FIGURA 9| ESQUEMA DE CONEXIÓN COMPLETO DEL REGULADOR

2.4 Diseño de Software

En el diseño del software nos centramos en la parte de programación, es decir, las instrucciones para el microprocesador. Se divide en la medida de intensidad, la medida de tensión, el control PID, el limitador de corriente y la monitorización y edición de parámetros en tiempo real.

Un diagrama de bloques simplificado del regulador es el siguiente:

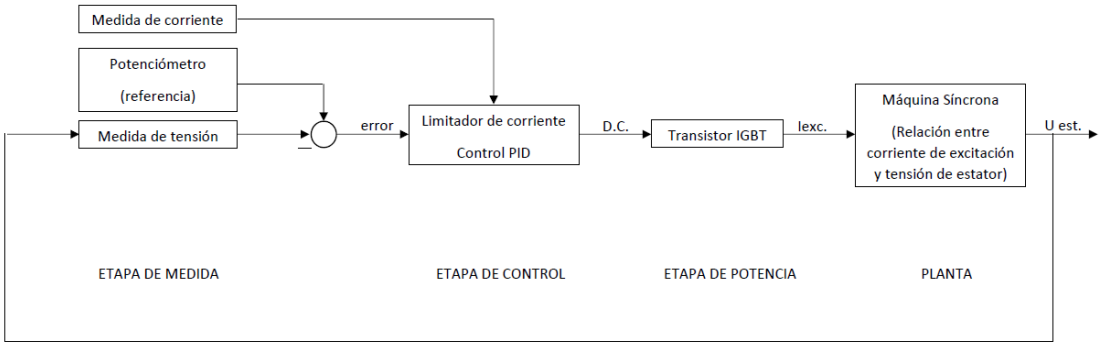


FIGURA 10|DIAGRAMA DE BLOQUES DEL REGULADOR

A nivel de software, tenemos que ser capaces de interpretar las señales que lleguen a los pines de medida y referencia traduciendo la señal de tensión a su magnitud real (Voltios de estator y Amperios de rotor). Si la corriente del rotor no supera el límite establecido procedemos a aplicar el control PID que, en función del error entre referencia y tensión medida, calcule el factor de servicio que determinará la cantidad de corriente de excitación que introducimos al rotor. Además, como ya se ha mencionado, se desea conseguir una interfaz por ordenador que permita monitorizar las principales señales del control (tensión de referencia, tensión medida y factor de servicio) y que permita ajustar los parámetros del control en tiempo real.

Hay que tener en cuenta que la relación entre intensidad de excitación y tensión de estator no es constante sino que depende de otros factores como velocidad, temperatura de los devanados, nivel de carga… También puede haber perturbaciones de otro tipo como por ejemplo variaciones en la tensión de excitación. El regulador debe ser capaz de mantener la tensión constante independientemente de todos estos factores.

2.4.1 Medida de intensidad

Para la medida de intensidad se ha utilizado la misma función que del proyecto anterior[4]. El código se puede encontrar al final de este documento. La función necesita como inputs la medida del pin encargado de medir intensidad, el tiempo de muestreo, el número de muestras deseado y devuelve el valor de la intensidad media.

Se recogen tantas muestras como se haya indicado por medio de un bucle. En cada iteración del bucle se suma la nueva muestra. Al acabar de recolectar las muestras se divide por el número de muestras totales. De esta forma tenemos un promedio de la intensidad durante el tiempo de muestreo deseado. Como la señal que recibimos del pin es una señal proporcional a la tensión proporcionada por el transductor hace falta pasar la señal primero de bits a voltios (determinado por Arduino) y luego de voltios a intensidad (determinado por el transductor). Se ha de elegir el tiempo de muestreo de forma que se consiga una medida con sentido. Como la intensidad medida es una onda PWM con una frecuencia igual a la frecuencia de conmutación habrá que asegurarse de muestrear como mínimo durante un tiempo igual al periodo de conmutación (2ms ó 255µs en función de si se ha seleccionada la frecuencia 490Hz o 3921Hz).

2.4.2 Medida de tensión

De nuevo, la función del proyecto de referencia [4] es reutilizable. Los inputs son la lectura del pin correspondiente, el tiempo de muestreo y el número de muestras.

En este caso hay que ir almacenando las muestras en un vector para luego encontrar el máximo y finalmente hallar el valor medio. Para traducir la lectura en el pin a voltios de estator hay que ir teniendo en cuenta el offset que se ha introducido con el buffer de tensión, los bits del pin de Arduino, el divisor resistivo y la relación de transformación del trafo de medida. Para asegurarnos de encontrar el máximo deberemos muestrear por un tiempo igual al periodo de la onda senoidal generada por la máquina síncrona. Si la máquina funciona a velocidad nominal la tensión de estator será de 50Hz (20ms).

2.4.3 Control PID

Para implementar el control PID se ha utilizado la librería Arduino PID library [5]. Consiste en una librería que incluye ciertas funciones para facilitar la aplicación de un control PID a unas variables. Se definen las variables que van a ser referencia (valor al cual quieres que sea igual la variable objetivo), entrada (la que se mide para compararla con la referencia y determinar el error, es decir, la variable que deseas controlar) y salida (el error procesado por el PID). En nuestro caso la referencia es el valor que se mide del potenciómetro y que es traducido a una tensión entre 0 y 220V. La variable entrada será la tensión de estator medida y la variable salida va a ser el incremento del factor de servicio del PWM. La razón por la que se utiliza el incremento del factor de servicio en vez del propio factor de servicio como salida del control es que cuando la referencia y la entrada coinciden, y por tanto el error es nulo, la salida del PID también es nula. Lo que se quiere es que el factor de servicio se quede en el valor que haga el error nulo y no que se anule cuando el error sea cero.

El ajuste de los parámetros se ha realizado gracias a la interfaz que se explica más adelante.

Si bien es cierto que el procedimiento normal sería hacer un primer ajuste del control por medio de simulación (utilizando el software **MatLab** y la herramienta **Simulink** por ejemplo), esto requeriría de la obtención de los parámetros dinámicos de la máquina síncrona por medio de diversos ensayos para poder modelar la planta. Esto es algo que hubiera llevado mucho tiempo y dadas las numerosas medidas de protección del laboratorio, la robustez de la máquina de diseño y la posibilidad de utilizar la interfaz de monitorización para cambiar de forma instantánea los parámetros del control y observar la respuesta, se decidió hacer un ajuste "empírico". Se comenzó con un control proporcional con constante de proporcionalidad igual a la unidad, el cual se esperaba que no diera ningún tipo de problemas en cuanto a estabilidad o velocidad. Desde ahí, se fueron probando distintos ajustes incluyendo controles P(proporcionales), PI(proporcionales-integrales), PD(proporcionales-derivativos) y PID(proporcionales-derivativos-integrales). Finalmente se optó por un PD con constante de proporcionalidad igual a 0.8 y constante derivativa igual a 0.1, que permitía una respuesta rápida y sin sobrepaso u oscilaciones apreciables. En el apartado de ensayos se podrá ver una respuesta a un escalón de 220V. Más detalles del ajuste se pueden encontrar en el CAPÍTULO III: ENSAYOS apartado 3.5 Ajuste del control PID.

2.4.4 Limitador de corriente

Utilizando la información de la medida de intensidad el procesador elige entre seguir realizando el control PID de forma normal o entrar en modo "alarma". El modo alarma retira la tensión de excitación y enciende un led para indicar que se ha llegado a un nivel de intensidad de excitación superior al programado. Se queda a la espera de presionar el botón de reset (incluido en la placa Arduino) para volver a poner en marcha el control. Esto garantiza que nunca se supere la intensidad de excitación máxima y que el usuario sea capaz de advertir que se ha llegado a ese límite y pueda realizar las acciones necesarias antes de volver a poner en marcha el control.

Hay que tener en cuenta en qué punto del circuito estamos midiendo al corriente. La corriente medida es la que pasa por el transistor, la cual se espera que sea una corriente cuadrada. El valor medio va a ser inferior que el valor medio de la corriente que circule por el devanado del rotor como se demostrará más adelante. El límite de corriente lo impone el elemento más restrictivo del conjunto regulador-rotor de la máquina. El elemento que menos corriente soporta de todo el conjunto es el transistor IGBT con 8A máximo, por lo que el hecho de estar midiendo la corriente por el transistor no es un problema. Sin embargo, si lo que quisiéramos limitar fuera la intensidad por el devanado del rotor tendríamos dos opciones. La primera opción consistiría en cambiar el transductor de corriente de posición y colocarlo entre el diodo de libre circulación y el rotor de forma que midiéramos directamente la corriente del devanado. La segunda opción es corregir la intensidad medida en el transistor con un factor el cual habría que determinar. A continuación se demuestra que la relación entre el valor medio de la corriente por el transistor y el valor medio de la corriente por el devanado del rotor es en realidad el factor de servicio:

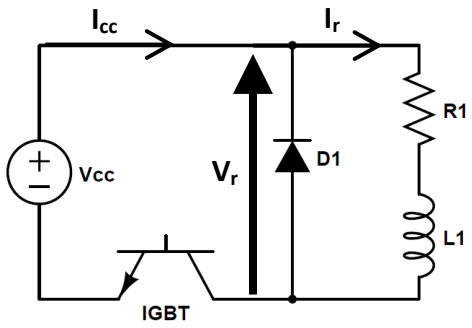


FIGURA 11|CIRCUITO SIMPLIFICADO DE LA TARJETA DE POTENCIA

La corriente que medimos con el transductor es I_{cc}.

Por equilibrio de potencias:

$$V_{cc} \cdot I_{cc} = V_r \cdot I_r \quad (1)$$

Además sabemos que:

$$V_r = D \cdot V_{cc} \quad (2)$$

Por tanto:

$$I_{cc} = D \cdot I_r \quad (3)$$

$$\frac{I_{cc}}{I_r} = D \quad (4)$$

Siendo D el factor de servicio del PWM y el resto de valores los valores medios de tensión e intensidad en los lugares señalados.

Por tanto, si estamos en lo correcto, al medir el valor medio de I_{cc} y de I_r en un determinado momento y dividirlos entre sí, deberíamos obtener el factor de servicio que se esté utilizando en ese mismo momento.

Diversas medidas realizadas con el osciloscopio confirman esta relación:

Utilizando un factor de servicio D=0.3:

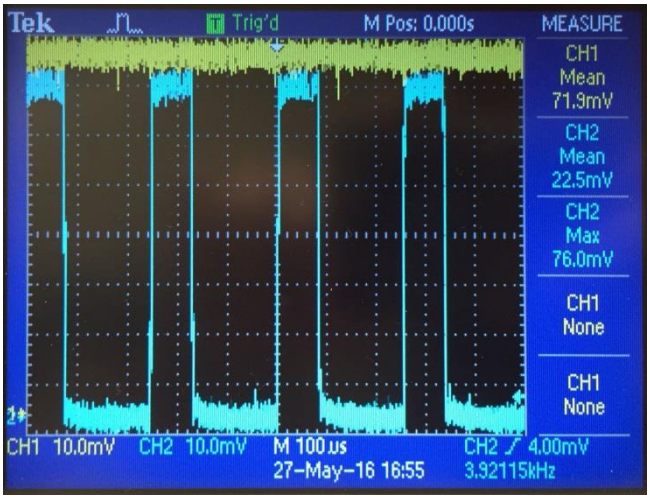


IMAGEN 3|CORRIENTE POR EL TRANSISTOR (AZUL) Y POR LA BOBINA (AMARILLO). FACTOR DE SERVICIO 0.3

$$\mathrm{D} = \frac{22.5}{71.9} = 0.31$$

Utilizando un factor de servicio D=0.5:

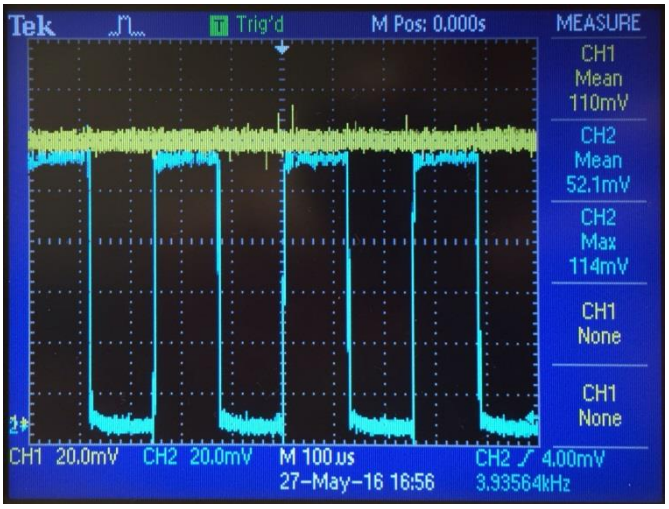


IMAGEN 4|CORRIENTE POR EL TRANSISTOR (AZUL) Y POR LA BOBINA (AMARILLO). FACTOR DE SERVICIO 0.5

$$\mathrm{D} = \frac{52.1}{110} = 0.47$$

Utilizando un factor de servicio D=0.8:

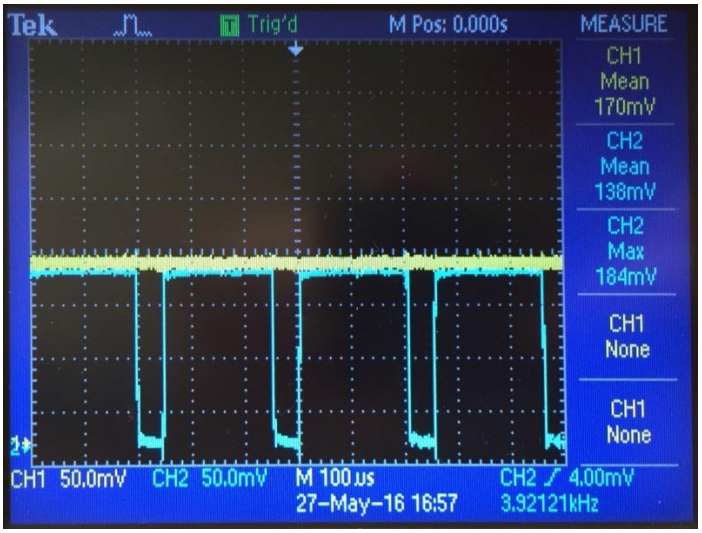


IMAGEN 5|CORRIENTE POR EL TRANSISTOR (AZUL) Y POR LA BOBINA (AMARILLO). FACTOR DE SERVICIO 0.8

$$\mathrm{D} = \frac{138}{170} = 0.81$$

2.4.5 Monitorización y edición de parámetros en tiempo real

El primer objetivo consiste en diseñar una herramienta que permita visualizar las variables más importantes del control en el tiempo mientras el control está ejecutándose por medio de un ordenador. De esta manera se podría visualizar toda la información relevante para el control en pantalla, pudiéndose observar el comportamiento del control ante una variación en la referencia, en la carga, en la tensión de excitación o en la velocidad.

Un segundo objetivo era poder modificar los parámetros del control durante la propia ejecución del programa, de manera que se puedan realizar cambios en el control de forma prácticamente instantánea y pudiendo ver inmediatamente después los efectos de ese cambio.

Se consideró la opción de utilizar Matlab, pero resultó que la placa Arduino Uno es la única de las placas Arduino que no es compatible con la monitorización en tiempo real con Matlab. Finalmente, recurriendo a foros en internet y con la ayuda de la comunidad Arduino surgió la solución. Se trata de Processing [6], un "lenguaje de programación y entorno de desarrollo integrado de código abierto basado en Java que se utiliza como medio de enseñanza y producción de proyectos multimedia e interactivos de diseño digital". Más concretamente la solución vino de la mano de la librería PID Front End [7] creada por Brett Beauregard [8]. Esta librería está pensada para crear una interacción con Arduino, recibiendo y graficando variables del mismo a través de una comunicación por el puerto USB. Adaptando su programa al proyecto se consiguió el siguiente resultado:



IMAGEN 6|INTERFAZ PARA MONITORIZACIÓN Y AJUSTE DEL REGULADOR

La señal verde muestra el nivel de la referencia en voltios, la señal roja la tensión de estator en voltios también y la señal azul el valor del factor de servicio expresado como un número de 0 a 255 en este caso (en otros casos se ha utilizado una escala de 0 a 100%). En el eje horizontal cada división representa 1s. A la izquierda se pueden leer los valores instantáneos de las tres señales y los valores de los parámetros del control (proporcional, integral y diferencial). Además se pueden escribir nuevos parámetros y enviarlos a Arduino de forma que se pueden probar múltiples configuraciones del control de una manera rápida y sencilla. En la sección de ensayos se hará uso de esta interfaz y se explicará a partir de ella el comportamiento del control.

2.5 Conclusiones

Se ha diseñado el regulador para operar con máquinas de 230V de tensión nominal y potencia nominal en el entorno de los 10kVA. El verdadero factor limitante es la corriente máxima del rotor, que es de 8A y la tensión máxima de excitación, de 100V pico. Por debajo de esos valores el regulador es capaz de manejar cualquier máquina síncrona de excitación estática independiente, siendo la tensión nominal del estator modificable por medio de programación.

A nivel de hardware, destaca el módulo de potencia, basado en un transistor IGBT que actúa como chopper para el cual se ha diseñado un driver adaptado a las necesidades concretas de la aplicación.

A nivel de software, se ha programado un control PID junto con la medida y el limitador de corriente y se ha adaptado una librería del entorno de desarrollo Processing para permitir la monitorización de las principales variables del control y el ajuste “en marcha” de los parámetros del mismo.

CAPÍTULO III: ENSAYOS

3.1 Introducción

En este capítulo se recogen los diferentes ensayos que se han realizado para comprobar y validar el funcionamiento del regulador.

En primer lugar se realizaron los ensayos necesarios para calibrar las diferentes medidas. A continuación se ajustaron los parámetros del control PID. Una vez realizados estos primeros ensayos de ajuste se procedió a realizar ensayos que demostraran y definieran el comportamiento del control. Estos ensayos son: respuesta a un escalón de 220V, seguimiento de la referencia, escalones de carga de 1kW a tensión nominal, variación de la tensión de excitación, variación de la velocidad, comprobación del funcionamiento del limitador de corriente y comprobación del efecto de la frecuencia de conmutación en la corriente. Con estos ensayos se pondrá a prueba la capacidad del control de rechazar perturbaciones externas, el funcionamiento del limitador de corriente y se observará qué diferencias hay entre alimentar la máquina con 490Hz o con 3921Hz.

3.2 Montaje y aparatos utilizados

El grupo en el que se realizaron todos los ensayos es el grupo 5 “Manuel Navajas” que consta de una máquina de inducción, una máquina de continua y una máquina síncrona. La máquina de continua no se utiliza en ningún momento. Las características de la máquina de inducción y síncrona se detallan a continuación:

CARACTERÍSTICAS DE LA MÁQUINA SÍNCRONA	
FABRICANTE	LETAG
TENSIÓN	230 V
INTENSIDAD	12.5 A
POTENCIA	5 kVA
FRECUENCIA	50 Hz
VELOCIDAD	1500 r.p.m.
FACTOR DE POTENCIA	0.8-1
EXCITACIÓN	
INTENSIDAD EN VACÍO	1.1 A
INTENSIDAD NOMINAL	4.3 A
TENSIÓN NOMINAL	50 V

TABLA 4|CARACTERÍSTICAS DE LA MÁQUINA SÍNCRONA (ENSAYOS)

CARACTERÍSTICAS DE LA MÁQUINA DE INDUCCIÓN	
FABRICANTE	AEG
TENSIÓN	220/380 V
INTENSIDAD	15/8.7 A
POTENCIA	5.5 CV
FRECUENCIA	50 Hz
VELOCIDAD	1435 r.p.m.

TABLA 5|CARACTERÍSTICAS DE LA MÁQUINA DE INDUCCIÓN (ENSAYOS)

La máquina de inducción es la encargada de mover la máquina síncrona. Para controlar la máquina de inducción se utilizó un variador **SIEMENS Micromaster 440** con un mando remoto. Con él se ajustó al velocidad a 1500r.p.m. Adicionalmente se conectó una resistencia de frenado para permitir un frenado del conjunto de menos de dos segundos.

Adicionalmente el grupo dispone de protección diferencial, de sobrecorriente y de sobrevelocidad.

Para la realización de todos estos ensayos se utilizó el mismo esquema de conexión:

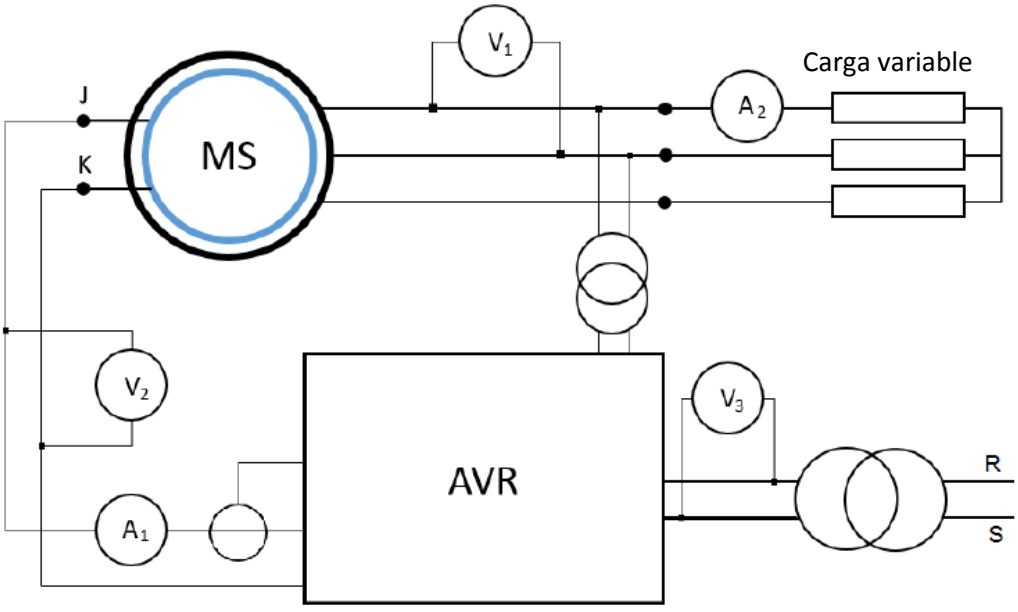


FIGURA 12|ESQUEMA ENSAYOS

A continuación se muestra una foto del banco de trabajo señalando los principales componentes:



IMAGEN 7|FOTOGRAFÍA ENSAYOS

La carga variable es una carga trifásica resistiva de 0kW a 7kW en escalones de 1kW a 220V.

Los instrumentos de medida utilizados son:

V_1: Polímetro **Promax CT-098**, Alcance 600V, ±(1.2%lect.+4dig.)

V_2: Osciloscopio **Tektronix TBS1042** (2500 muestras):

Alcances tiempo: (5–10–25–50–100–250–500)ns/div–(1–2.5–5–10–25–50–100–250–500)µs/div–(1–2.5–5–10–25–50–100–250–500)ms/div. Precisión (si amplitud mayor de 5div): 1muestra + 100ppmmedida + 0.4ns.

Alcances tensión: (5–10–20–50–100–200–500)mV/div–(1–2–5)V/div. Precisión (si alcance mayor de5mV/div): 3% medida + 0.1div + 1mV.

Eje vertical: 8 div.

Eje horizontal: 10 div.

Impedancia de entrada: 1MΩ(±2%) en paralelo con 20pF(±3pF)

Con: Sonda de tensión diferencial **YOKOGAWA 700924,** Rango 1/100, ±3%

V_3: Polímetro **Promax CT-098**, Alcance 600V, ±(1.2%lect.+4dig.)

A_1: Osciloscopio **Tektronix TBS1042** (2500 muestras):

Alcances tiempo: (5–10–25–50–100–250–500)ns/div–(1–2.5–5–10–25–50–100–250–500)µs/div–(1–2.5–5–10–25–50–100–250–500)ms/div. Precisión (si amplitud mayor de 5div): 1muestra + 100ppmmedida + 0.4ns.

Alcances tensión: (5–10–20–50–100–200–500)mV/div–(1–2–5)V/div. Precisión (si alcance mayor de5mV/div): 3% medida + 0.1div + 1mV.

Eje vertical: 8 div.

Eje horizontal: 10 div.

Impedancia de entrada: 1MΩ(±2%) en paralelo con 20pF(±3pF)

Con: Pinza amperimétrica **CHAUVIN ARNOUX E3N**, Rango 100mV/div, ±(3%lect.+50mA)

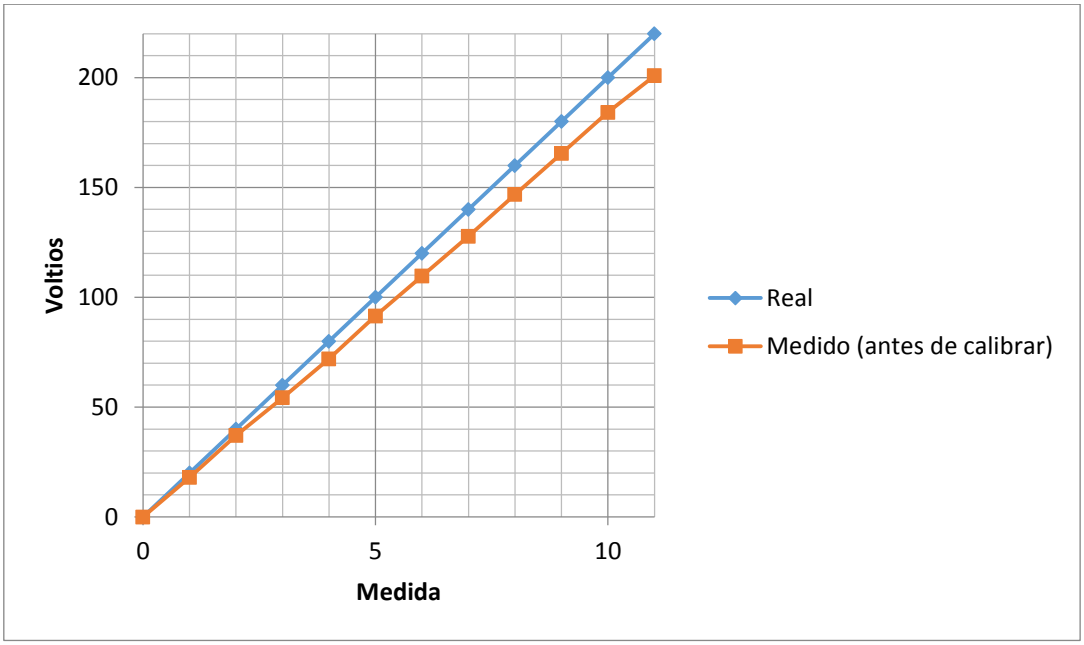
A_2: Polímetro **Promax CT-098**, Alcance 20A ±(1.5%lect.+4dig.)

3.3 Calibración medida de tensión

En este apartado se pretende mostrar la precisión de la medida de tensión del regulador. Se hará mediante la comparación con otro aparato de medida que se considerará ideal (aparato V_1).

Pruebas anteriores a la instalación del regulador en la máquina síncrona mostraron que la medida de tensión tenía un error proporcional (probablemente debido a un error en la relación de transformación del transformador), motivo por el cual se añadió un factor de corrección de 1.25. Se realizó una primera tanda de medidas para comprobar si ese factor de corrección era suficiente.

La gráfica a continuación muestra medidas tomadas desde 0 hasta 220V en intervalos de 20V con un factor de corrección en la medida de 1.25 como ya se ha mencionado anteriormente:



GRÁFICA 1|ERROR EN LA MEDIDA DE TENSIÓN

La tabla a continuación resume las medidas tomadas:

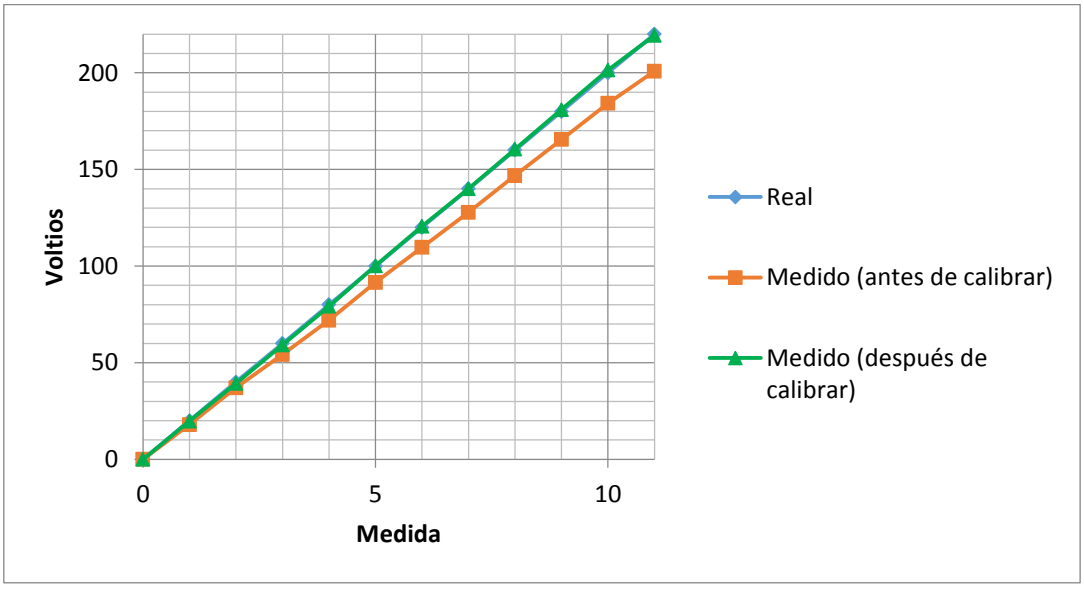
Número	Real (V)	Medido (V)	Real/Medido	Real-Medido
0	0	0	-	0
1	20	17.98	1.11	2.02
2	40	37.05	1.08	2.95
3	60	54.27	1.11	5.73
4	80	71.94	1.11	8.06
5	100	91.48	1.09	8.52
6	120	109.62	1.09	10.38
7	140	127.77	1.10	12.23

8	160	146.84	1.09	13.16
9	180	165.45	1.09	14.55
10	200	184.2	1.09	15.8
11	220	200.86	1.10	19.14

TABLA 6|CALIBRACIÓN DE LA MEDIDA DE TENSIÓN

Se puede comprobar que la medida del regulador esta siempre por debajo de la medida tomada como referencia. Además también se puede comprobar que la diferencia entre la referencia y la medida aumenta conforme aumenta la medida, sin embargo, al dividir la referencia entre la medida se obtiene un valor prácticamente constante. Esto nos demuestra que, como ya se había obtenido en pruebas anteriores, el error de medida es proporcional a la medida. Ya se había utilizado un factor de corrección de 1.25 pero sigue siendo necesario utilizar un factor adicional de 1.1. El factor de corrección resultante es por tanto 1.375.

Se realizó otra tanda de medidas con el nuevo factor de corrección. La gráfica siguiente muestra la mejora (la gráfica "real" y la "medida(después de calibrar)"coinciden hasta tal punto que cuesta distinguirlas):



GRÁFICA 2|VERIFICACIÓN TRAS CALIBRACIÓN DE MEDIDA DE TENSIÓN

3.4 Calibración medida de intensidad

La medida de intensidad está destinada a permitir una limitación en la corriente máxima de excitación para proteger la máquina y la electrónica de potencia ante sobrecorrientes.

Una medida para una tarea de este tipo no necesita ser extremadamente precisa y por ello no se ha visto la necesidad de calibrarla de forma fina como ocurría con la medida de tensión.

La corriente de excitación necesaria para alimentar una carga de 4kW a 220V (la potencia nominal de la máquina de inducción que mueve la máquina síncrona) resulta ser de 1.7A de valor medio aproximadamente. La corriente máxima que la tarjeta de potencia admite viene determinada por el transistor IGBT y es de 8A. La corriente de excitación máxima de la máquina síncrona es de 4.3A. Por ello se ha fijado a 3A la corriente del limitador.

Se ha comprobado la medida de intensidad con la máquina funcionando en las condiciones mencionadas arriba (4kW, 220V) comparándola con la medida del osciloscopio:

Osciloscopio	**Regulador**
1.69A	1.63A-1.71A

TABLA 7|MEDIDA DE INTENSIDAD

Adicionalmente, previamente a la instalación del regulador en la máquina síncrona, se hicieron pruebas con el transductor que demostró tener una precisión suficiente. Se considera que la medida es suficientemente precisa para su propósito.

3.5 Ajuste del control PID

Como ya se ha mencionado en apartados anteriores, el ajuste del control se ha realizado de forma empírica haciendo uso de la interfaz de Processing. Se recuerda que la señal verde representa la referencia en Voltios del estator, la señal roja la tensión de estator medida en las mismas unidades y la señal azul el valor del factor de servicio en porcentaje. Cada división horizontal representa 1s.

Se ha ajustado observando la respuesta a un escalón de 100V. Se utilizó un escalón de 100V en vez de 220V para que en caso de que se produjera sobrepaso no ocurrieran valores de tensión excesivamente superiores a 220V.

En primer lugar se comenzó con un control proporcional con constante unitaria:

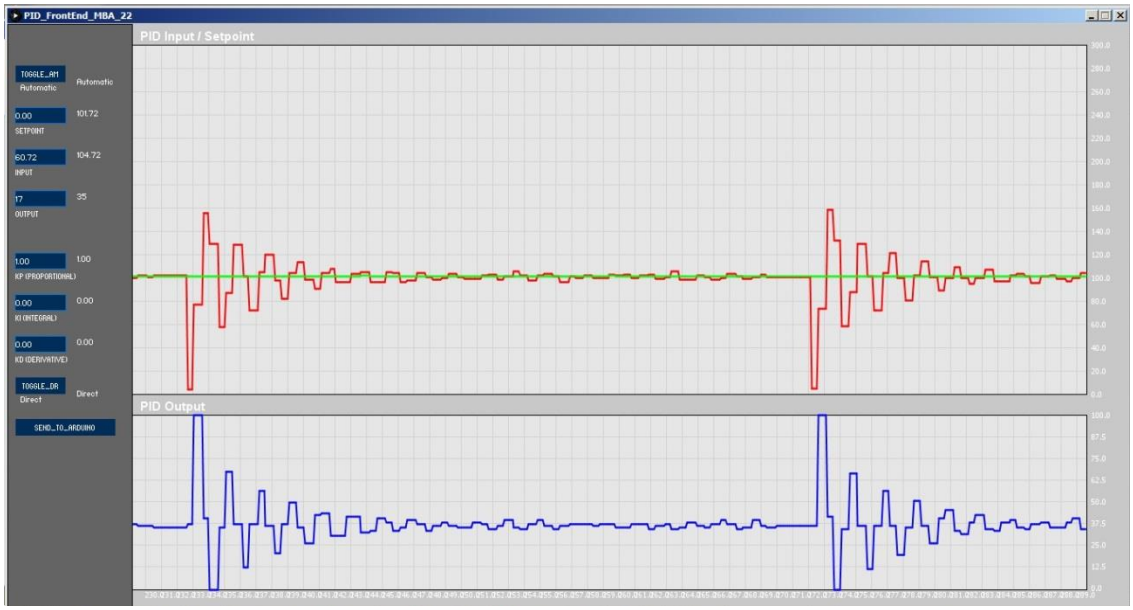


IMAGEN 8|AJUSTE PID (1,0,0)

La respuesta presenta un gran sobrepaso (60V/100V) y un tiempo de establecimiento de más de 7s (más oscilaciones de las deseadas). Por ello se decidió bajar la constante a 0.8 con el siguiente resultado:

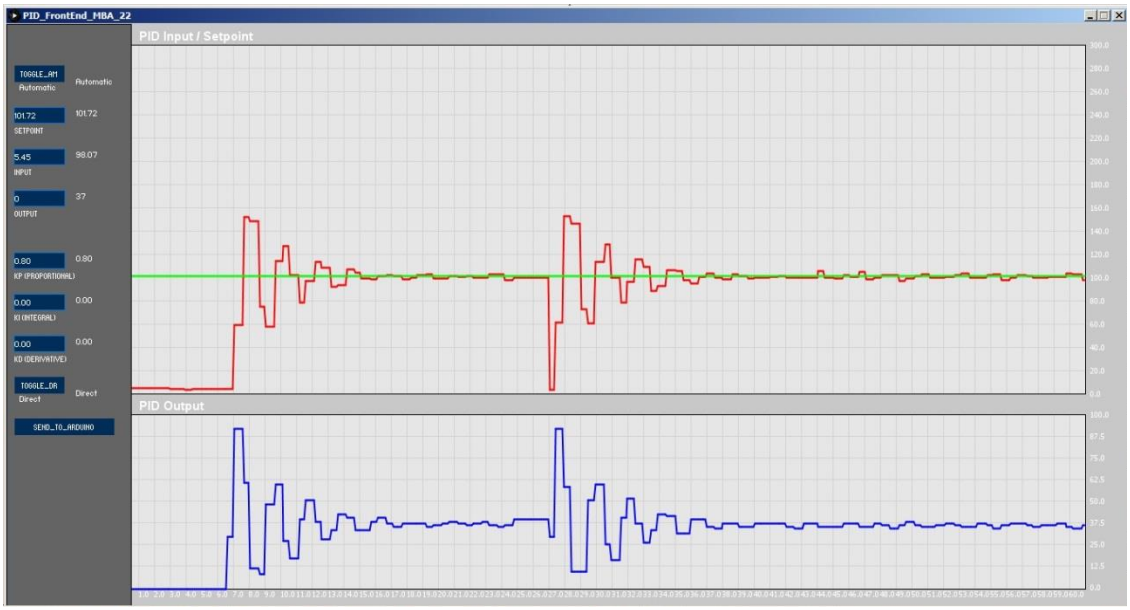


IMAGEN 9|AJUSTE PID (0.8,0,0)

El sobrepaso no se consigue reducir pero las oscilaciones son algo menores. Constantes de proporcionalidad más bajas hacían el control excesivamente lento y constantes más altas daban como resultado sobrepasos y tiempos de establecimiento demasiado grandes. A continuación se probaron varias configuraciones de controles PI dejando la constante proporcional fija a 0.8. La introducción de una constante integral provocaba un aumento de las oscilaciones y, a pesar de mejorar el tiempo de alcance, este beneficio se veía contrarrestrado con tiempos de establecimiento mayores de 20s. Un ejemplo con constante integral 0.1 se muestra a continuación:

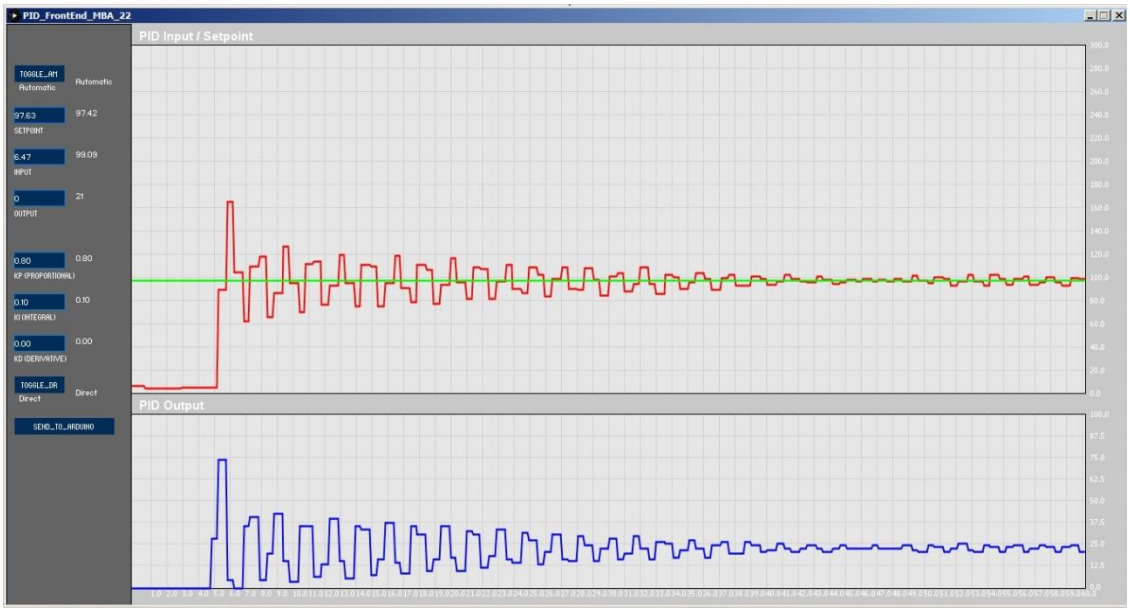


IMAGEN 10|AJUSTE PID (0.8,0.1,0)

También se probaron diferentes controles PD, dando estos un resultado mucho más satisfactorio. Tras varias pruebas se llegó a la conclusión de que un buen ajuste era P=0.8 y D=0.1:



IMAGEN 11|AJUSTE PID (0.8,0,0.1)

El sobrepaso se reduce a un tercio de lo obtenido con los anteriores controles (20V/100V) y el tiempo de establecimiento pasa a ser de apenas 2s. Mayores constantes derivativas hacían de nuevo el control demasiado lento y menores causaban un aumento considerable del sobrepaso.

También se hicieron pruebas con controles PID no encontrando una mejora sustancial con respecto al PD a no ser que la constante integral alcanzara valores donde empezara a predominar, en cuyo caso el comportamiento empeoraba asemejándolo de nuevo a un control PI.

Finalmente se opta por el control PD con P=0.8 y D=0.1.

En el apartado siguiente se muestra la respuesta de este control a un escalón de 220V.

3.5 Ensayo escalón de tensión

En este ensayo se pretende mostrar la respuesta del control ante un escalón de 0 a 220V. El escalón se consgiue fijando la referencia al valor deseado y pulsando el botón reset de Arduino. El control se trata de un PD con una constante proporcional de 0.8 y una derivativa de 0.1, el ajuste al que se llego en el apartado anterior. Este control es el que se usará en el resto de ensayos.



IMAGEN 12|RESPUESTA A UN ESCALÓN DE 220V

Se puede comprobar como se alcanza el valor 220V en poco más de un segundo y sin sobrepaso ni oscilaciones apreciables, con lo que se puede afirmar que el comportamiento mejora con respecto al escalon de 100V.

3.6 Ensayo seguimiento de referencia

En este ensayo se pretende mostrar la capacidad del regulador de seguir cambios en la referencia. De nuevo, cada división horizontal representa un segundo.

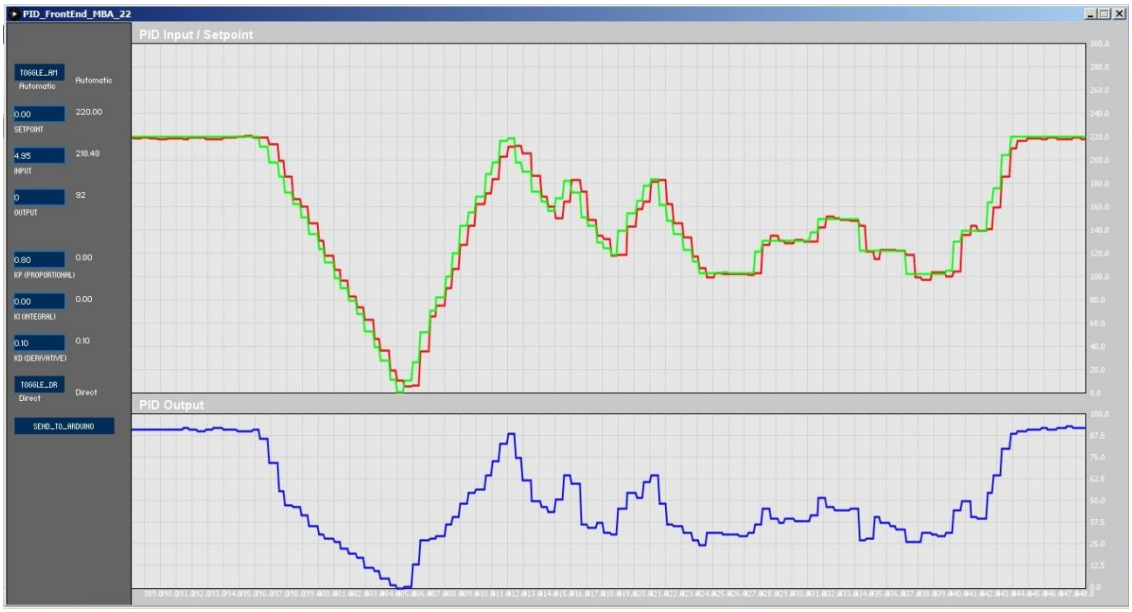


IMAGEN 13|ENSAYO SEGUIMIENTO DE REFERENCIA

Se puede comprobar como el seguimiento de la referencia es rápido y cercano, solo habiendo un retraso de menos de un segundo entre la referencia y la tensión de estator.

3.7 Ensayo escalones de carga

En este ensayo se muestra el comportamiento del regulador ante escalones de potencia generada. Se ha realizado a tensión nominal alimentando una carga resistiva. Cada escalón es de 1kW y se ensaya desde vacío hasta 4kW y de nuevo hasta vacío.



IMAGEN 14|ENSAYO ESCALONES DE CARGA DE 1kW DE 0 A 4kW Y DE 4 A 0kW, 220V

Se puede observar como en los escalones de subida hay una pequeña tendencia a que la tensión baje que es rápidamente corregida por el regulador. En los escalones de bajada ocurre al contrario, tendiendo la tensión a subir y siendo rápidamente disminuida por el regulador.

3.8 Ensayo variación de la tensión de excitación

Este ensayo pretende mostrar la robustez del regulador. Si bien es cierto que no deberia haber en una situación real un motivo por el cual la tensión de excitación variara de forma brusca, con este ensayo se busca forzar los límites del regulador y comprobar su capacidad para rechazar perturbaciones externas. Aprovechando el VARIAC que se encuentra alimentando la tarjeta de potencia se ha variado de forma brusca la tensión siendo la respuesta del control la siguiente:



IMAGEN 15|ENSAYO VARIACIÓN DE LA TENSIÓN DE EXCITACIÓN

Se puede apreciar como el regulador consigue rechazar de forma efectiva las variaciones de tensión, salvo en el caso en que la tensión de excitación disminuya en exceso y aun utilizando un factor de servicio del 100% no sea una tensión suficiente como para proporcionar la intensidad de excitación necesaria para conseguir los 220V en el estator.

3.9 Ensayo variación de la velocidad

En cualquier grupo electrógeno no solo habrá un regulador de tensión sino también un regulador de velocidad que se encargue de mantener la frecuencia constante. Aun así, también se ha querido comprobar la respuesta del control ante cambios de velocidad. Los primeros son más bruscos y el último más lento. La velocidad es la nominal (1500r.p.m.) en los momentos donde el factor de servicio es más bajo y 500r.p.m. en los momentos donde el factor de servicio satura a 100%.

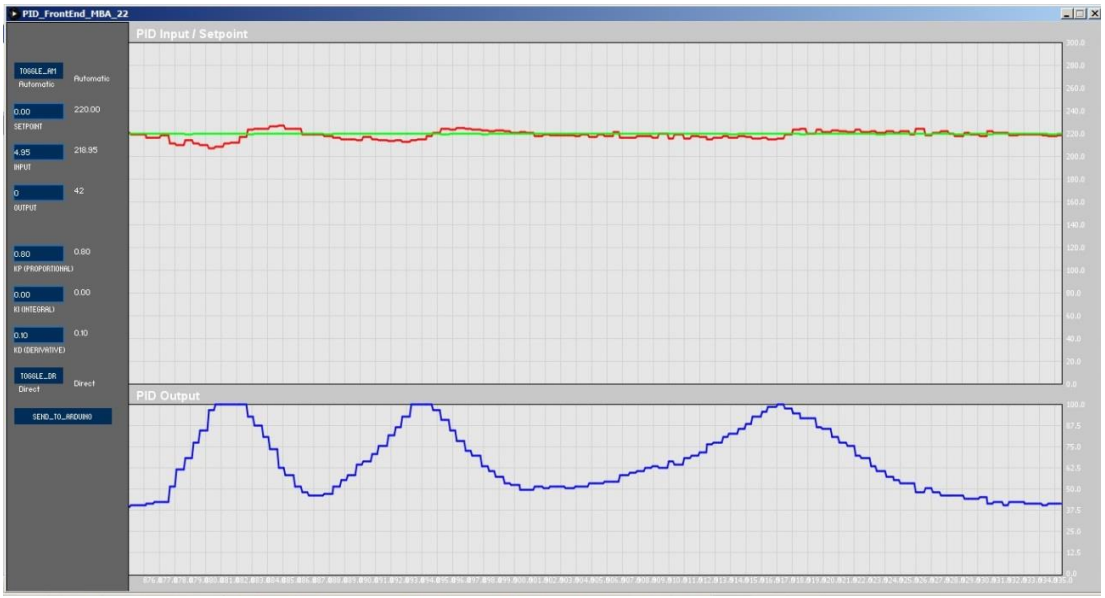


IMAGEN 16|ENSAYO VARIACIÓN DE LA VELOCIDAD

Una vez más, el regulador demuestra ser efectivo a la hora de mantener la tensión constante siempre y cuando la tensión de excitación máxima proporcionada sea suficiente. Se ha llegado a comprobar el funcionamiento con velocidades 3 veces inferiores a la nominal sin que esto haya supuesto ningún problema.

3.10 Comprobación del funcionamiento del limitador de corriente

La siguiente gráfica muestra el comportamiento del limitador de corriente. Para hacerlo saltar se ha reducido el valor a 1A ya que, como ya se meciono en el apartado 3.4 Calibración de la medida de intensidad, la corriente de excitación que la máquina consume generando 4kW a 220V es de 1.7A.

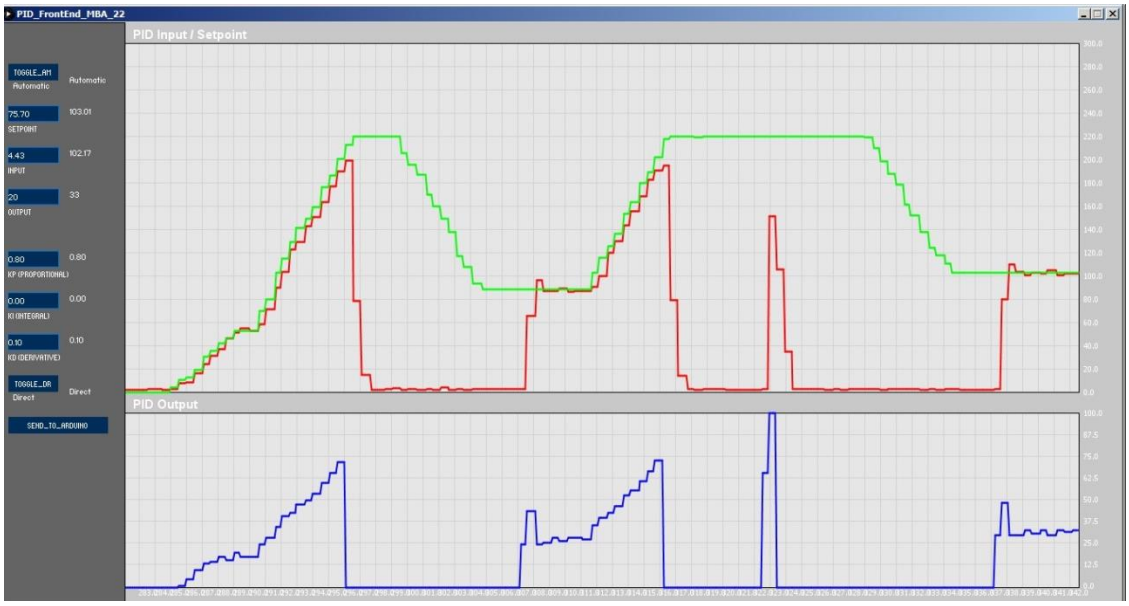


IMAGEN 17|COMPROBACIÓN DEL FUNCIONAMIENTO DEL LIMITADOR DE CORRIENTE

Se puede apreciar como antes de llegar a los 220V el limitador actúa reduciendo el factor de servicio a cero. Se baja la referencia para evitar la situación en la cual se producía la sobrecorriente y se pulsa el botón reset. Vemos que el regulador vuelve a funcionar normalmente. Se sube de nuevo la referencia para volver a provocar el disparo y de nuevo el limitador actúa. Al pulsar el botón reset el regulador intenta de nuevo arrancar, pero como esta vez no se ha bajado la referencia se vuelve a producir el exceso de corriente y el limitador corta de nuevo la excitación en menos de un segundo. Se puede concluir que el limitador funciona de la forma esperada y es eficaz evitando excesos de corriente en el rotor.

3.11 Comprobación efecto frecuencia de conmutación

Como ya se ha mencionado en apartados anteriores, la frecuencia de conmutación del prototipo anterior [4] era de 490Hz y este año se ha decidido probar a aumentarla a 3921Hz. Se ha realizado una comparación de la forma de la corriente de excitación con ambas frecuencias para ver si realmente la diferencia es apreciable.

A continuación se presentan las distintas medidas realizadas:

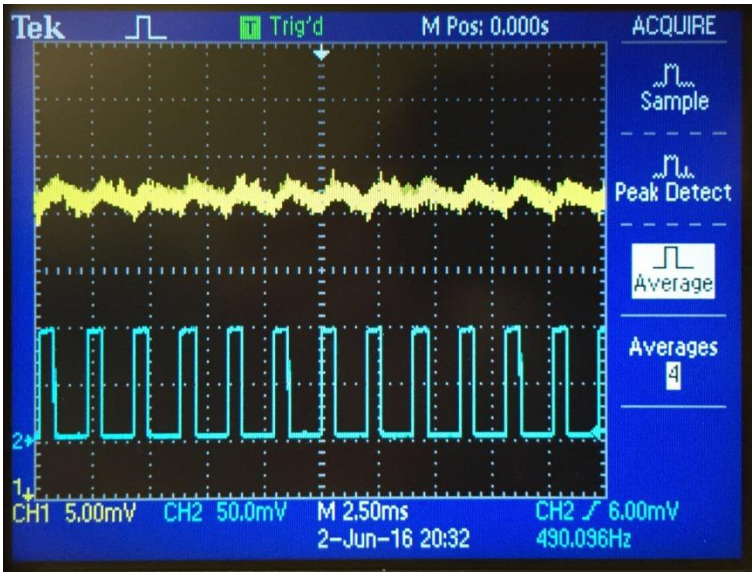


IMAGEN 18|CORRIENTE DEL ROTOR CON CONMUTACIÓN A 490Hz

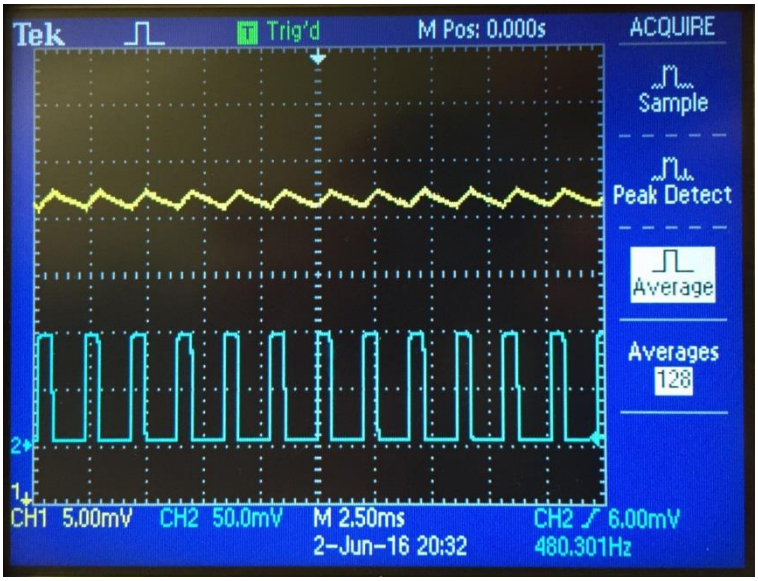


IMAGEN 19|CORRIENTE DEL ROTOR CON CONMUTACIÓN A 490Hz (PROMEDIADO)

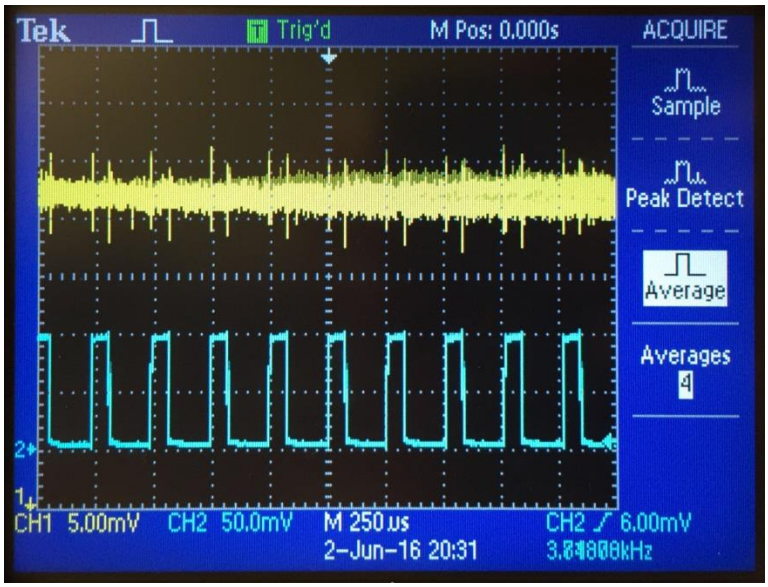


IMAGEN 20|CORRIENTE DEL ROTOR CON CONMUTACIÓN A 3921Hz

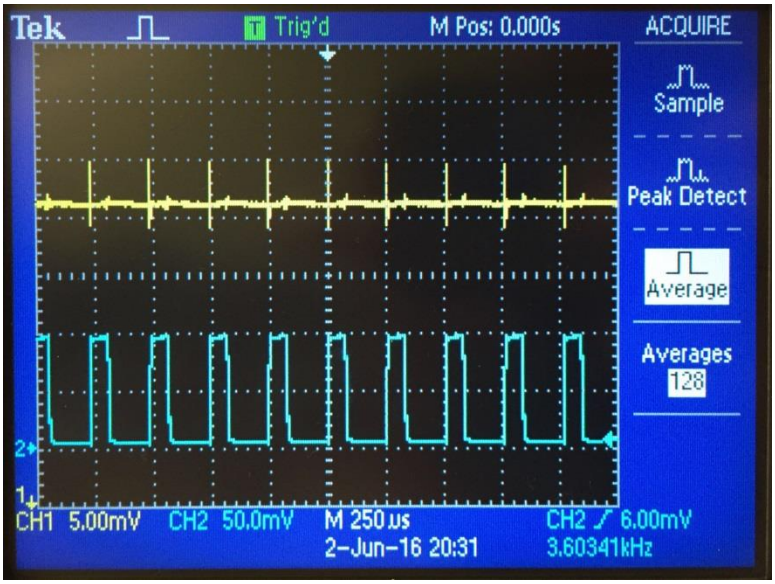


IMAGEN 21|CORRIENTE DEL ROTOR CON CONMUTACIÓN A 3921Hz (PROMEDIADO)

Se puede observar que el rizado de la corriente cuando se conmuta a 490Hz es muy pequeño (no llega a 50mA) y que, aun que mejora al subir la frecuencia, no es un rizado preocupante. Por tanto, se puede conmutar indistintamente a 490Hz o a 3921Hz. Esto es seleccionable por software y por tanto el usuario tiene la opción de elegir la frecuencia que más le convenga.

3.12 Conclusiones

Se ha calibrado la medida de tensión la cual resulta de vital importancia para conseguir la tensión de estator deseada. La medida de intensidad tiene la función de servir al limitador de corriente, por lo que su precisión no es tan crucial como la de la medida de tensión y se considera que no es necesario calibrarla. Si se comprobó una medida de 1.7A (la máxima que corriente de rotor que se puede conseguir en condiciones normales de funcionamiento) y previamente a la instalación del regulador en la máquina se hicieron pruebas con el transductor de corriente que mostraban una precisión adecuada.

También se ha realizado un ajuste del control PID haciendo uso de la interfaz de Processing, llegando a un control PD de constantes P=0.8 y D=0.1.

Una vez realizadas las calibraciones y ajustes necesarios se han realizado los ensayos para comprobar su funcionamiento. Los ensayos han comprendido la respuesta a un escalon de 220V, seguimiento de referencia, variación de carga, velocidad y tensión de excitación y comprobación del funcionamiento del limitador de corriente. Todos ellos han arrojado resultados satisfactorios demostrando un comportamiento robusto y eficaz.

Por último se ha demostrado que la inductancia del rotor es tan grande que no hay necesidad de subir la frecuencia de 490Hz a 3921Hz, al menos por motivo del rizado de la corriente del rotor.

CAPÍTULO IV: ESTUDIO ECONÓMICO

4.1 Introducción

Este capítulo tiene como objetivo dar una idea del coste de construcción de este regulador, atendiendo no sólo al coste de los componentes sino también al coste de las horas invertidas en su desarrollo. En primer lugar se realizará una estimación de las horas de trabajo que le llevaría a un equipo de ingenieros profesional realizar las distintas tareas, para más adelante valorar esas horas en términos económicos. En otro apartado se calculará el coste total de los materiales utilizados y finalmente se realizará el presupuesto del conjunto.

4.2 Estimación de las horas de trabajo

Las tareas se pueden dividir en dos grandes grupos: aquellas cuyo fin era el desarrollo y diseño conceptual del regulador y aquellas cuyo fin era, una vez terminada la fase de diseño, llevar a cabo la construcción e instalación del equipo.

En primer lugar habría que revisar el pedido para comprobar si se tiene la capacidad técnica y los medios materiales para llevarlo a cabo.

Una vez aceptado el pedido se comenzaría con la fase de **desarrollo y diseño**. Las tareas que forman parte del **desarrollo y diseño** son: la documentación y especificación técnica, el diseño del hardware y el diseño del software. Dentro del diseño de hardware se puede distinguir entre el diseño de la etapa de medida, la etapa de control y la etapa de potencia. Dentro del diseño del software se puede distinguir entre programación del control y programación de la interfaz de monitoreo y ajuste.

La estimación de las horas que se emplearían para cada tarea se muestra a continuación:

Tarea	Subtarea	Número de horas dedicadas
Documentación y especificación técnica	Documentación	2
	Especificación técnica	1
Diseño de hardware	Medida	3
	Control	1
	Potencia	16
Diseño de software	Programación del control	10
	Programación de la interfaz	4
Total horas desarrollo y diseño		37

TABLA 8|HORAS DESARROLLO Y DISEÑO

Una vez terminado el desarrollo y diseño conceptual del equipo se pasa a la fase **fabricación e instalación.** Las tareas que forman parte de la fabricación e instalación son: diseño, montaje y soldadura, ensamblado, ensayos y elaboración del manual de instrucciones y montaje. Diseño en este caso implica decidir la distribución física de los componentes en la placa. El montaje y la soldadura consiste en la fijación de los componentes a la placa, el ensamblado la correcta conexión entre los distintos módulos y los ensayos son la manera de

comprobar el correcto funcionamiento del regulador y de hacer las calibraciones y ajustes necesarios. Por último, la elaboración del manual de instrucciones consiste en redacción de un documento que permita al usuario el manejo del equipo de una forma segura.

De nuevo, las horas estimadas que se requerirían son:

Tarea	Número de horas dedicadas
Diseño de las tarjetas electrónicas	0.5
Montaje y soldadura de las tarjetas	3
Ensamblado y cableado del conjunto	1.5
Ensayos de validación y puesta a punto	6
Elaboración del manual de instrucciones y montaje	1
Total horas fabricación e instalación	12

TABLA 9|HORAS FABRICACIÓN E INSTALACIÓN

4.3 Relación de tareas y profesionales

El reparto de las tareas entre los distintos profesionales del equipo es el siguiente:

Profesional	Tareas
Ingeniero de diseño	Documentación, especificación técnica, diseño de hardware, diseño de las tarjetas electrónicas y elaboración del manual de instrucciones y montaje
Programador	Documentación y diseño de software
Técnico de montaje	Montaje y soldadura de las tarjetas, ensamblado y cableado del conjunto
Ingeniero responsable de ensayos	Ensayos y puesta a punto

TABLA 10|RELACIÓN DE TAREAS Y PROFESIONALES

4.4 Coste de las horas de trabajo

Agrupando las horas totales invertidas por cada profesional y teniendo en cuenta el sueldo por hora:

Tipo trabajo	Coste unitario	Horas estimadas	Coste total
Ingeniero de diseño	40€/h	23.5	940€
Programador	20€/h	15	300€
Técnico de montaje	15€/h	4.5	67.5€
Ingeniero responsable de ensayos	40€/h	6	240€
Total			**1547.5€**

TABLA 11|COSTE DE LAS HORAS DE TRABAJO

4.5 Coste de los componentes y materiales

El coste de los componentes (IVA incluido) utilizados para construir el regulador:

Componente	Cantidad	Precio Unitario	Total
Amplificador Operacional UA 741	1	0.26€	0.26€
Transformador 230V/9V	1	5.07€	5.07€
Transductor de corriente 30A 66mV/A	1	6€	6€
Potenciómetro	1	1.51€	1.51€
Placa Arduino UNO	1	20€	20€
Optoaclopador TLP350	1	1.71€	1.71€
Transistor IGBT 1200V, 8A, 20kHz	1	2.78€	2.78€
Regulador lineal de tensión 7815CV	1	0.48€	0.48€
Puente rectificador de diodos 25A, 600V	1	5.72€	5.72€
Puente rectificador de diodos 1A, 80V	1	0.61€	0.61€
Diodo rápido 1000V, 12A, 50ns	1	1.33€	1.33€
Transformador 220V/15V	1	14.45€	14.45€
Transformador 220V/127V+VARIAC*	1	69.58€	69.58€
Resistencias y Condensadores**	13	0.40€	5.20€
Cableado	-	-	2€
Placa para montajes	1	5.50€	5.50€
Disipadores de calor para TO-22	3	1.46€	4.38€
Otros***	-	-	5€
TOTAL COMPONENTES	**151.58€**		

TABLA 12|COSTE DE LOS COMPONENTES DEL REGULADOR

*No es la configuración definitiva, pero se incluye con un valor de 69.58€ que es el coste estimado de una alimentación definitiva en el apartado 5.2 Futuros desarrollos.

**Se ha utilizado un coste promedio de los componentes. El precio de estos componentes es muy bajo y muy variable en función del número de unidades que se compren.

***Se incluye el estaño utilizado para soldar, el LED utilizado para la alarma, los bornes de conexión…

Adicionalmente, habría que añadir el coste de una caja donde colocar el equipo para su fácil transporte y manejo. Esta acción se llevará a cabo en el presente mes de Junio.

4.7 Resumen de costes

En la siguiente tabla se resumen los costes totales de desarrollo, fabricación y puesta a punto del producto:

Capítulo	**Coste**	
Horas de trabajo del ingeniero de diseño	940€	55.32%
Horas de trabajo del programador	300€	17.66%
Horas de trabajo del técnico de montaje	67.5€	3.97%
Horas de trabajo del ingeniero de ensayos	240€	14.13%
Materiales y componentes	151.58€	8.92%
Total	**1699.08€**	

TABLA 13|RESUMEN DE COSTES

4.8 Conclusiones

Para este estudio se ha tenido en cuenta el IVA en el coste de materiales y componentes, el cual se descontaría caso de comercializarse el producto. No se han tenido en cuenta otros costes imputables como seguridad social, alquiler de las instalaciones, energía, agua, salarios de administrativos y demás gastos generales que pudiera tener la empresa que lo construye y comercializa.

Hay que tener presente además que el coste de desarrollo, que supone más del 90% del total, solo se produce una vez. En caso de que la empresa en cuestión se dedicara a fabricar este producto en masa, el coste de desarrollo se repartiría entre todas las unidades producidas. Además, el coste de los materiales se vería reducido debido a comprarse en grandes cantidades. Todo esto haría que el coste unitario del producto fuera cercano a los 150€ en vez de a los 1700€ calculados.

Por tanto, como conclusión, se puede afirmar que se ha construido un regulador automático de tensión que cuenta con un control PID y su propio software de monitorización y ajuste con un coste de apenas 150€.

CAPÍTULO V: CONCLUSIONES

5.1 Introducción

En este capítulo se compilan las conclusiones del proyecto y se incluye un apartado con futuros desarrollos hablando de las posibles mejoras que se pueden implementar al regulador.

5.2 Conclusiones del proyecto

Se ha diseñado un regulador automático de tensión para máquinas síncronas de excitación estática independiente totalmente funcional y con un coste de aproximadamente 150€. Se trata de un AVR que basa su etapa de potencia en un chopper IGBT controlado por la placa **Arduino UNO**, que es una plataforma de desarrollo hardware y software "open-source" que utiliza un microprocesador de la casa **ATmel**.

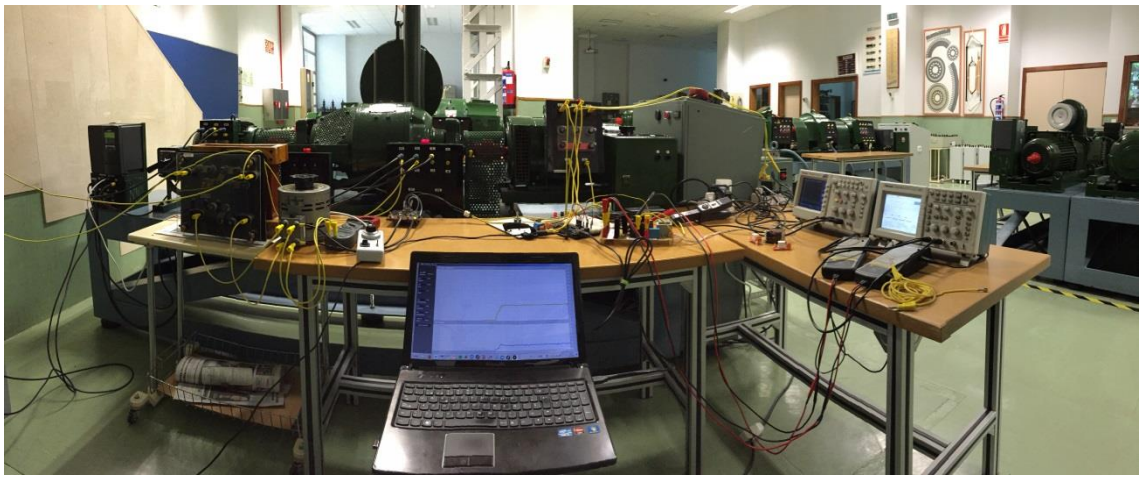


IMAGEN 22|FOTOGRAFÍA CONJUNTO

En el apartado de control utiliza un control PID completamente ajustable gracias a una interfaz que es capaz de monitorizar y configurar el regulador en tiempo real.



IMAGEN 23|INTERFAZ PROCESSING

El regulador ofrece gran flexibilidad al ser capaz de adaptarse a un gran número de máquinas síncronas de baja potencia y una gran robustez y fiabilidad al haber sido diseñado con numerosas medidas de protección tanto a nivel de software (limitador de corriente electrónico con señalizador luminoso) como a nivel de hardware (diodos de protección).

El regulador ha superado numerosos ensayos de forma sobresaliente, demostrando ser capaz de rechazar de forma satisfactoria todo tipo de perturbaciones externas, desde variaciones en la carga hasta variaciones en la velocidad de rotación o en la tensión de excitación.

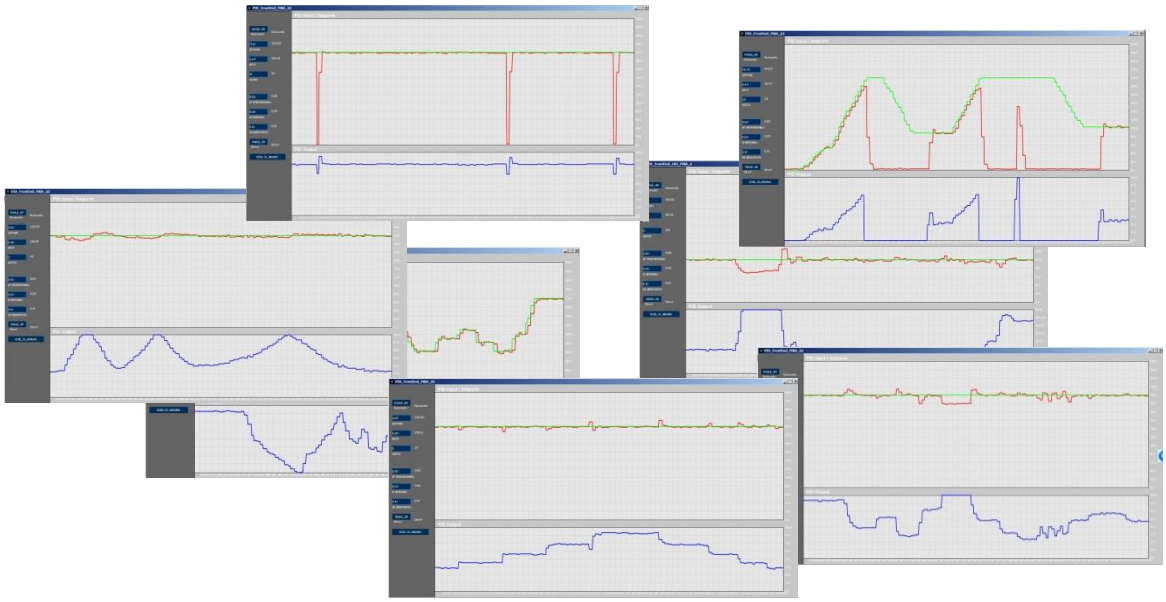


IMAGEN 24|ENSAYOS

A continuación se muestra una tabla que resume las características del regulador de tensión:

CARACTERÍSTICAS DEL REGULADOR AUTOMÁTICO DE TENSIÓN		
TIPO	VALOR	ELEMENTO LIMITANTE
TENSIÓN MÁXIMA DE EXCITACIÓN	100 Voltios (Pico) 57 Voltios (RMS)	Condensadores 1000µF
CORRIENTE MÁXIMA DE EXCITACIÓN	8 Amperios	Transistor IGBT
FRECUENCIA DE CONMUTACIÓN	Seleccionable 490Hz/3921Hz	Microprocesador/Transistor IGBT
TENSIÓN DE ALIMENTACIÓN DE LA TARJETA DE POTENCIA	230 Voltios 50Hz (Red doméstica)	-
TENSIÓN DE ALIMENTACIÓN DEL MICROPROCESADOR	5 Voltios D.C.	-
LIMITADOR DE CORRIENTE	Programable. Desexcitación de la máquina con pulsador de rearme.	-
TENSIÓN DE REFERENCIA	Ajustable de 0 a 220 Voltios por medio de un potenciómetro.	-
PRECISIÓN DEL REGULADOR	Función de la tensión de referencia y de la tensión de excitación. Típicamente ±2% a 220V.	-
TIPO DE CONTROL	PID con parámetros ajustables.	-
TIPO DE MÁQUINA SÍNCRONA	Máquina síncrona de excitación independiente.	-

TABLA 14|CARACTERÍSTICAS DEL REGULADOR AUTOMÁTICO DE TENSIÓN

Se ha añadido una columna con el título “elemento limitante” ya que reemplazando ese elemento por otro se conseguirían ampliar los límites del funcionamiento del regulador de forma muy sencilla. Por ejemplo, cambiando los condensadores a la salida del puente de diodos por otros con límite de tensión 220V la tensión de excitación máxima que podría aportar el regulador serían esos 220V. También cambiando el transistor podría aumentarse la intensidad de excitación máxima o incluso la frecuencia de conmutación si además se utiliza una placa más potente como por ejemplo la placa Arduino Mega.

A continuación se presenta otra tabla que recuerda las características de la máquina con la que se ha diseñado el regulador:

CARACTERÍSTICAS NOMINALES DE LA MÁQUINA SÍNCRONA DE DISEÑO	
FABRICANTE	LETAG
TENSIÓN	230 V
INTENSIDAD	12.5 A
POTENCIA	5 kVA
FRECUENCIA	50 Hz
VELOCIDAD	1500 r.p.m.
FACTOR DE POTENCIA	0.8-1
EXCITACIÓN	
INTENSIDAD EN VACÍO	1.1 A
INTENSIDAD NOMINAL	4.3 A
TENSIÓN NOMINAL	50 V

TABLA 15|CARACTERÍSTICAS NOMINALES DE LA MÁQUINA DE DISEÑO (BIS)

Esta tabla nos sirve para poder compararla con otra máquina para la cual el regulador sería también válido sin necesidad de ninguna modificación (salvo quizás reajustar el control):

EJEMPLO MÁQUINA SÍNCRONA VÁLIDA	
FABRICANTE	Indar
TENSIÓN	220 V
INTENSIDAD	31A
POTENCIA	12 kVA
FRECUENCIA	50 Hz
VELOCIDAD	1500 r.p.m.
FACTOR DE POTENCIA	0.8
EXCITACIÓN	
INTENSIDAD EN VACÍO	1.9 A
INTENSIDAD NOMINAL	3.3 A
TENSIÓN NOMINAL	30 V

TABLA 16|EJEMPLO DE MÁQUINA SÍNCRONA VÁLIDA

Se puede apreciar que esta máquina tiene más del doble de potencia que la anterior y sin embargo entra perfectamente dentro de los límites del regulador.

5.3 Futuros desarrollos

Por último, este AVR es susceptible a muchas mejoras y ampliación de sus capacidades. A continuación se tratan varias de las distintas mejoras posibles:

Implementación de un sistema de alimentación inteligente

Consiste en que el propio procesador, en función del factor de servicio que el control le pida utilizar, sea capaz de cambiar la tensión que alimenta el puente rectificador de diodos, disminuyéndola siempre tanto como sea posible para trabajar con factores de servicio altos, donde el control tiene una mayor precisión y estabilidad.

Sería fácilmente realizable utilizando dos transformadores de 230V/15V 200VA con dos salidas cada uno y tres contactores que contaran con dos contactos normalmente cerrados y uno normalmente abierto cada uno. Para mover los contactores con Arduino haría falta utilizar relés y/o transistores ya que la placa Arduino no tiene capacidad para excitar la bobina de un contactor directamente.

El esquema de conexión de los transformadores y contactores quedaría de la siguiente forma:

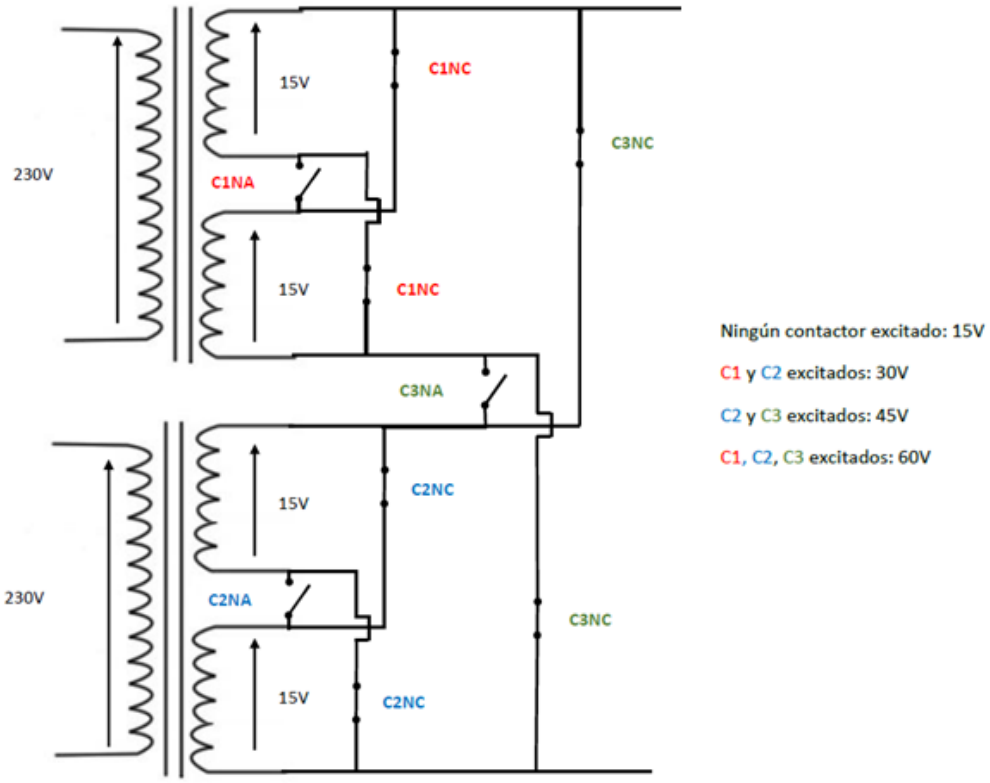
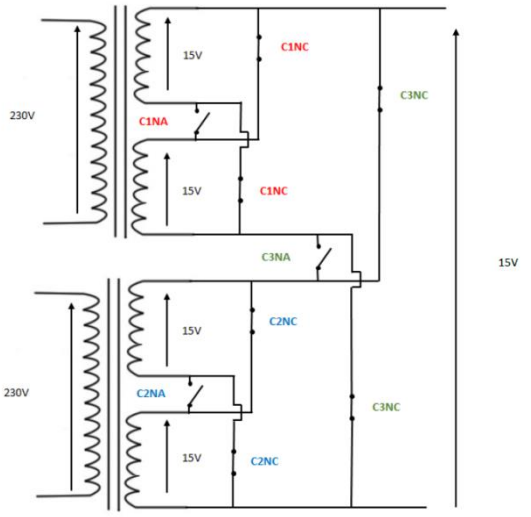
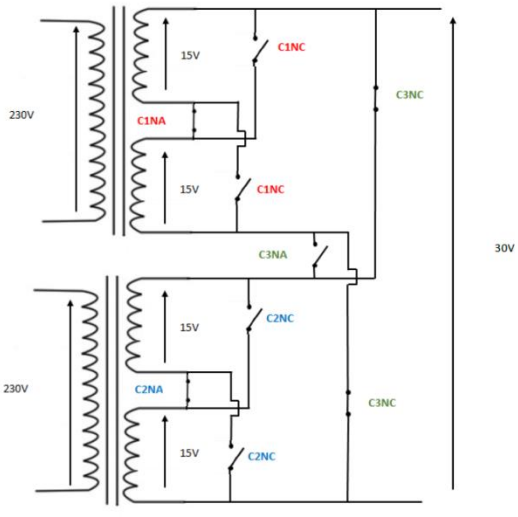


FIGURA 13|SISTEMA DE ALIMENTACIÓN POR ESCALONES AUTOMÁTICO



15V



30V

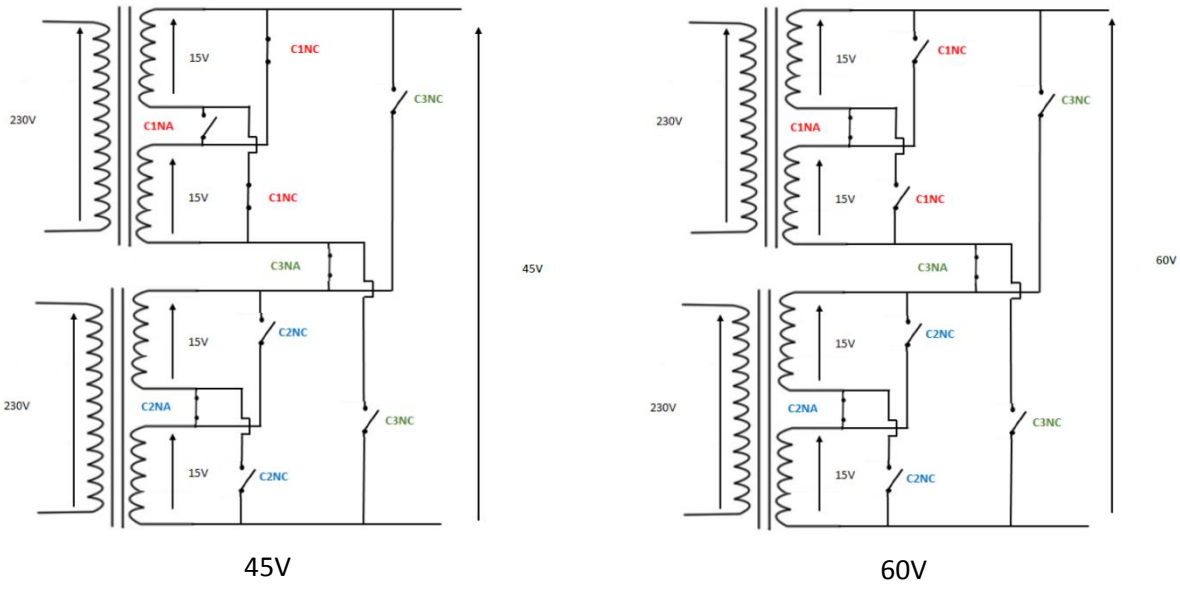


FIGURA 14|SISTEMA DE ALIMENTACIÓN POR ESCALONES AUTOMÁTICO (POSICIONES)

Habría que añadir en la programación de Arduino las siguientes líneas de código: Código adicional Arduino.

Un ejemplo de los elementos necesarios para implementar esta mejora son:

-x2 Transformador 230V/15Vx2 200VA. Precio unidad: 48.03€ (Código RS:503-934)

-x3 Relé 5Vdc, 2A. Precio unidad: 2.46€ (Código RS:815-4916)

-x3 Contactor 230Vac, 16A. Precio unidad: 19.09€ (Código RS:376-874)

Coste total de los componentes: 160.71€

Otra opción podría ser utilizar un solo transformador y obtener únicamente dos escalones de tensión, abaratando a cambio el coste de la alimentación hasta los 69.58€ (solo haría falta un transformador, un relé y un contactor) y reduciendo la posibilidad de fallo.

Construcción de un prototipo compacto

Reducir el tamaño de los componentes que actualmente son voluminosos y provisionales para poder meter todo el equipo en una caja móvil que solo necesite ser conectada a la red para poder funcionar.

Adaptación del equipo a máquinas autoexcitadas

Analizar el comportamiento en una máquina autoexcitada para ver si el equipo es directamente aplicable a este tipo de máquinas y en caso contrario realizar las modificaciones necesarias en el control o el hardware.

Adaptación del equipo para controlar potencia reactiva en máquinas conectadas a la red

Analizar la posibilidad de adaptar el equipo para poder controlar potencia reactiva con la máquina conectada a una red infinita. Podría instalarse un selector que indique al control si se está trabajando en isla o en red infinita de forma que pueda actuar en consecuencia. Requeriría cambios a nivel de software y de hardware. A nivel de hardware sería necesario poder medir la intensidad de estator y el desfase entre tensión e intensidad de estator.

Implementación de un ajustador de parámetros automático

Existen infinidad de herramientas desarrolladas para Arduino que la comunidad comparte gustosamente. Entre estas herramientas existen librerías capaces de ajustar un control PID de forma automática como por ejemplo la librería "PIDAutotuneLibrary". Podría estudiarse la utilización de esta librería para que el equipo tenga la capacidad de ajustarse solo a las distintas máquinas.

Limitador de cambios bruscos de tensión

A nivel de programación evitar que se puedan producir escalones en la referencia, de tal forma que haya una derivada máxima para la tensión y que no se produzcan picos de tensión peligrosos.

Uso de una placa más potente

Se puede considerar la opción de utilizar una palca más potente que la Arduino UNO, como por ejemplo la Arduino DUE, que permite mayores frecuencias de conmutación y tiene una mayor memoria, lo cual puede ser necesario para implementar otras mejoras ya mencionadas en este apartado.

5.4 Conclusiones personales

En este apartado pretendo hacer un resumen de todo lo que he aprendido en la realización de este proyecto.

En primer lugar, al tratarse de un proyecto técnico que incluía tanto diseño como construcción y ensayo de un prototipo, he ido atravesando las distintas fases de un proyecto de estas características y me he encontrado con obstáculos y dificultades que he tenido que ir superando.

Comencé por documentarme, profundizando en lo que es y cómo funciona un AVR, encontrando que esta información no es tan inmediatamente accesible y fácil de encontrar, especialmente a la hora de buscar detalles técnicos y constructivos de AVR comerciales. Continué analizando el trabajo previo realizado en el proyecto que tomé como referencia, el

cual me sirvió de punto de partida. Este análisis no solo consistió en leer la memoria del proyecto, sino también en el ensayo del hardware ya existente. A la hora de realizar estos ensayos iniciales es cuando me encontré con las primeras dificultades, al no funcionar el hardware como se esperaba y tener que localizar los fallos para poder solventarlos. Fue necesario hace uso del osciloscopio y reducir los circuitos a sus partes más fundamentales, comprobando el funcionamiento de cada componente.

Una vez realizado este trabajo previo, comenzó la fase de la especificación técnica, donde se identificaron las características mínimas con las que el AVR debía contar en función de las características máquina síncrona en la que se fuera a instalar y las limitaciones del microprocesador encargado del control. Teniendo la especificación clara se diseñó el hardware y software.

En el diseño del hardware me familiaricé con **CircuitLab** que, como ya se ha explicado en apartados anteriores, es una potente aplicación online para diseño y simulación de circuitos eléctricos y electrónicos.

También me dio la experiencia de buscar los componentes que se iban a comprar, teniendo en cuenta tanto el precio como las especificaciones individuales de cada uno y la compatibilidad entre ellos, para lo cual tuve que aprender a leer e interpretar de forma eficiente las hojas de características de los diferentes fabricantes. Dado que mis nociones de **electrónica** son básicas, el diseño del hardware también supuso una profundización en mi conocimiento sobre las características que definían ciertos componentes, como por ejemplo un transistor IGBT, o incluso me encontré con elementos nuevos para mí, como optoacopladores o reguladores lineales de tensión. Todo lo anterior finalmente desembocó en una "lista de compra" de los componentes requeridos.

El apartado del diseño de software también trajo consigo un gran aprendizaje. Profundicé en la **programación** en C++ gracias a **Arduino** y aprendí programación en Java gracias a **Processing**. Conocí la vasta comunidad que hay detrás de los software libres, sumergiéndome en sus foros donde encontré multitud de personas dispuestas a compartir su conocimiento y ayudar.

Los ensayos me han servido para poder practicar todo lo aprendido en asignaturas como **Circuitos**, **Electrotécnia**, **Máquinas Eléctricas** o **Instrumentación Eléctrica**, realizando montajes y medidas con osciloscopios, polímetros digitales y analógicos y comprobando de primera mano el comportamiento dinámico de la máquina síncrona y los efectos que variaciones de carga, de excitación, de velocidad etc. tienen en ella. Además he podido observar de primera mano el funcionamiento de un control PID y las distintas respuestas que ofrece en función de los parámetros y el tipo de control, cuyo ajuste de nuevo ha supuesto una aplicación directa de los conocimientos adquiridos en **Sistemas Dinámicos** y **Regulación Automática**. También ha sido muy interesante observar el efecto de condensadores e inductancias en la tensión e intensidad respectivamente, siendo un fenómeno muy estudiado en teoría pero que he podido constatar gracias a las medidas realizadas con el osciloscopio.

En suma, este proyecto me ha hecho aplicar una gran parte de los conocimientos adquiridos durante la carrera y me ha hecho aprender cosas nuevas que de seguro me serán útiles en mi futuro profesional.

CAPÍTULO VI: REFERENCIAS

6.1 Libros, artículos, revistas y otros proyectos

[1] FRÍAS MARÍN, P. (2014). *Máquinas Eléctricas Tema 2: Profundización en los fundamentos de las máquinas eléctricas rotativas.* Madrid: Universidad Pontificia Comillas.
[2] FRAILE MORA, J. (2003). *Máquinas Eléctricas*. Madrid: McGraw-Hill
[4] DE ASÍS SALAS IGEA, F. (2015). *Diseño y construcción de una regulador automático de tensión para máquina síncrona.* Director: FRÍAS MARÍN, P. Proyecto fin de grado. Madrid: Universidad Pontificia Comillas.

6.2 Páginas web y Blogs

[3] ABB*. AVR para excitaciones rotativas*.
<http://www.abb.com/product/es/9AAC100577.aspx?country=ES>

[5] Librería Arduino PID Library.
<http://playground.arduino.cc/Code/PIDLibrary >

[6] Processing.
<https://processing.org/>

[7] Brett Beauregard.
<http://brettbeauregard.com/blog/>

[8] Librería PID Front End.
<http://playground.arduino.cc/Code/PIDLibrary>

[9]CircuitLab
< https://www.circuitlab.com/>

ANEXOS

En esta sección el lector podrá encontrar el conjunto de los esquemas eléctricos, el código fuente utilizado para programar el microprocesador y la interfaz de monitoreo y ajuste, fotos del prototipo y las hojas técnicas de los componentes más importantes del regulador.

Anexo I: Esquemas eléctricos

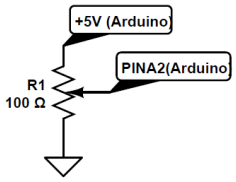


FIGURA 13|INTRODUCCIÓN DE LA REFERENCIA DE TENSIÓN (ANEXO)

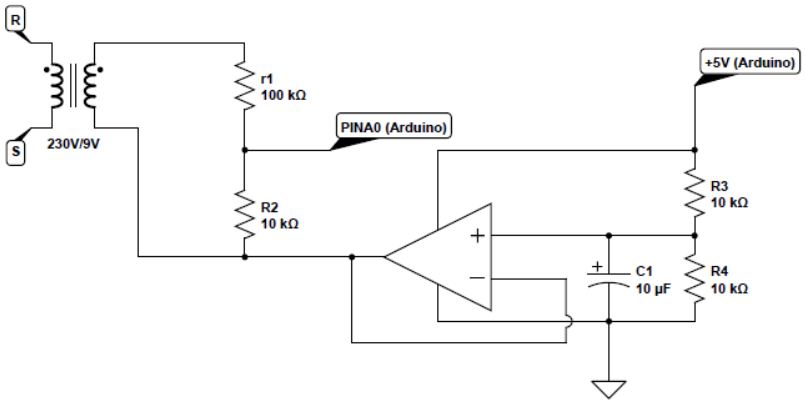


FIGURA 14|CIRCUITO DE MEDIDA DE TENSIÓN (ANEXO)

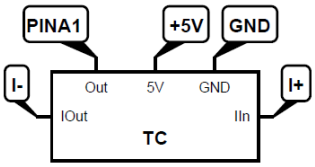


FIGURA 15|CIRCUITO DE MEDIDA DE INTENSIDAD (ANEXO)

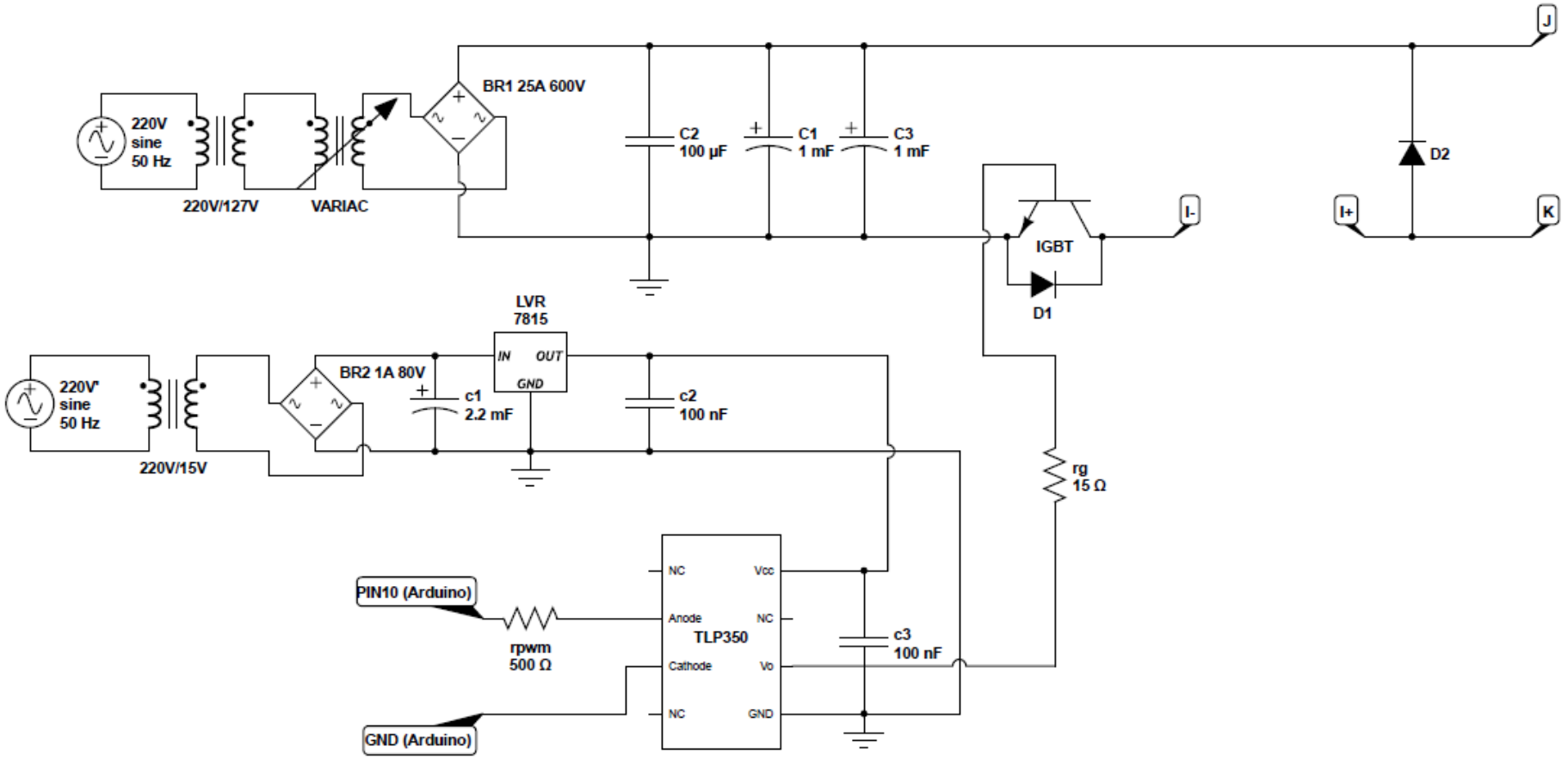


FIGURA 16|TARJETA DE POTENCIA (ANEXO)

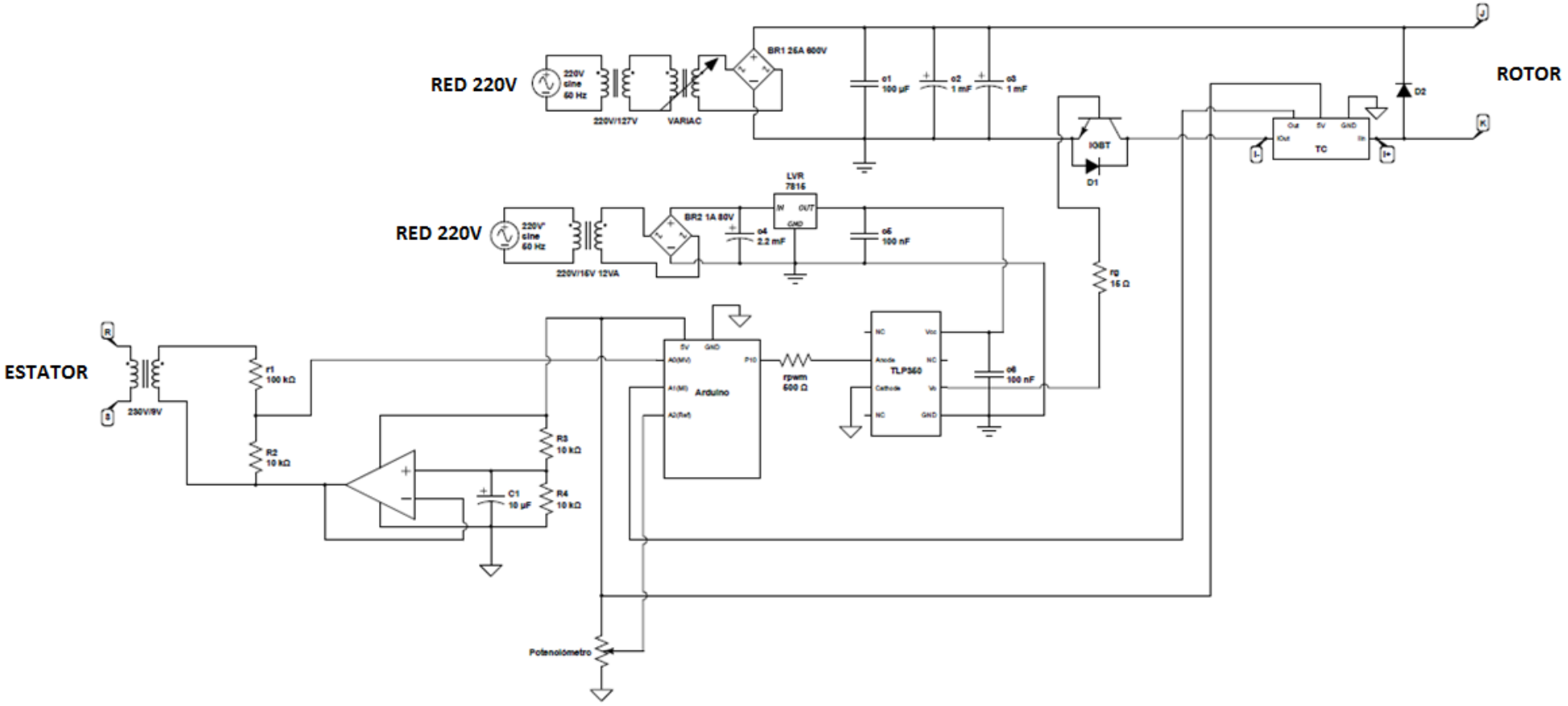


FIGURA 17|ESQUEMA DE CONEXIÓN DEL CONJUNTO (ANEXO)

Anexo II: Código fuente

Arduino

```
/* Creado por Miguel Berbegal Almohalla
Febrero 9, 2016
Ingenieria Electromecánica especialidad Eléctrica
Proyecto de Fin de Grado. Diseño y ensayo de un regulador de tensión
automático y control de velocidad para máquinas síncronas.
Software para Arduino UNO


Este código funciona en conjunto con Processing para monitorizar y
editar el control PID en tiempo real.
*/

#include <PID_v1.h> //libreria control PID

/*Declaracion de variables*/
double Urms; // Tension rms medida
int pinU = A0; // Pin en el que se va a medir la señal de tension
int pinI = A1; // Pin en el que se va a medir la señal de intensidad
int pinS = 10; // Pin en el que se dar la señal PWM de salida
int pinP = A2; // Pin en el que se conecta el potenciometro
int TmuestreoU = 22; // Periodo de muestreo medida de tension en
milisegundos
int nMuestrasU = 800; // Numero de muestras en la medida de tension
double Icont = 0; // Corriente continua medida
double Imax = 3; // Corriente maxima por el rotor en amperios
int TmuestreoI = 25.5; // Tiempo durante el que quiere hacerse el
muestreo en milisegindos
int nMuestrasI = 20; // Numero de muestras en la medida de intensidad
double ref; // Valor de la referencia de tension
int valorS = 0; // Valor de la salida PWM 0 min 255 max
int DC=0;// Valor del duty cycle 0% min 100% max
int pot; // Indicador del estado del potenciometro
double salidaPID; //Salida del PID
int alarm = 0; //0 si no hay sobrecorriente en el rotor, 1 si ha
habido sobrecorriente


PID myPID(&Urms,&salidaPID,&ref,0.8,0,0.1,DIRECT);//Define las
variables de referencia entrada y salida del PID

unsigned long serialTime; //Ayudará a la hora de relacionarse con
Processing

/*Cuerpo del programa*/

void setup()
{
Serial.begin(9600);
TCCR1B = TCCR1B & 0b11111000 | 0x02;
//Pone los pins 9 y 10 de Arduino Uno a una frecuencia determinada
```

```
//0x01=31372.55 Hz, 0x02=3921 Hz, 0x03=490 Hz
//Definición de entradas y salidas
pinMode(pinS, OUTPUT); //PWM
pinMode(pinU, INPUT); //Medida tensión
pinMode(pinI, INPUT); //Medida intensidad
pinMode(11, OUTPUT);  //LED alarma
digitalWrite(11,LOW);

myPID.SetOutputLimits(-255, 255); //Establece los limites del PID.El
PID puede dar una salida negativa ya que la salida
                                  //del PID no va a ser directamente
el valor del duty cycle sino u incremento o
                                  //decremento del mismo
myPID.SetMode(AUTOMATIC);         //Enciende el PID


}

void loop()
{
/*Medida*/
Urms = MedidaTension(pinU, TmuestreoU, nMuestrasU); //Llama a la
función de medida de tensión
Icont = MedidaIntensidad(pinI, TmuestreoI, nMuestrasI, Icont); //Llama
a la función de medida de intensidad
pot = analogRead(pinP); //Lee el pin del potenciómetro
/*Calculo de la referencia*/
ref = (float)pot*220/1023; //Pasar la lectura del potenciómetro a
voltios de estator


myPID.Compute();  //El PID da una salida que representa el aumento del
duty cycle. Puede ser negativo o positivo
valorS=valorS+salidaPID; //Se le añade al valor del duty cycle el
incremento calculado por el PID

if(Icont > Imax){ //Si la corriente suepra la maxima salta la alarma
  alarm=1;
}

if(valorS > 255){ //limita el duty cycle a 1
  valorS=255;
}
if(valorS < 0){ //limita el duty cycle a 0
  valorS=0;
}
if (alarm==0){ //Si no hay sobrecorriente
  analogWrite(pinS,valorS); //Escribe el duty cycle en el pin del PWM
  digitalWrite(11,LOW); //Apaga el LED de alarma
}
if (alarm==1){ //Si hay sobrecorriente
  valorS=0; //Corta la tensión de excitación
  analogWrite(pinS,0);
  digitalWrite(11,HIGH); //Enciende el LED de alarma
}
```

```
DC=valorS*100/255; //para graficar el duty cycle en porcentaje en vez
de 0 a 255 en Processing


//Enviar-recibir con Processing si es el momento
if(millis()>serialTime)
{
  SerialReceive();
  SerialSend();
  serialTime+=500;
}

}


float MedidaTension(int pinU, int TmuestreoU, int nMuestrasU){
/* Funcion que devuelve la medida de una tension senoidal en rms leida
en el pinU,
introduciendo el numero de muestras que quiere utilizarse y el periodo
de muestreo
*/
/*Declaracion de Variables*/
int i = 0; // Indicador para los bucles
int Muestras[nMuestrasU]; // Vector en el que se almacenaran las
muestras
float Uef; // Valor de la tension eficaz
int D = TmuestreoU*1000/nMuestrasU; // Tiempo que ha de transcurrir
entre muestra y
//muestra en microsegundos
float mmax; // Valor de la tension de pico (dato sin tratar)
int K = 1375; // Coeficiente de correccion
/*Muestreo*/
do{
Muestras[i] = analogRead(pinU);
delayMicroseconds(D);
i++;
}
while(i < (nMuestrasU));
/*Deteccion del valor de pico*/
i = 0;
mmax = Muestras[0];
for(i = 0; i < nMuestrasU; i++){
if(Muestras[i]> mmax) {
mmax = Muestras[i];
}
}
/*Calculos*/
Uef = (mmax -518.35)/1.9/sqrt(2)*K/1000; /*El coeficiente de
correccion se aplica
como producto ya que el error es distinto
para cada nivel de tension y asi se
consigue la mejor compensacion posible*/
return (Uef);
}
```

```
float MedidaIntensidad(int pinI, int TmuestreoI, int nMuestrasI,
double Icont){
  /* Funcion que devuelve la medida de intensidad continua leida en el
pinI, como
la media de 10 muestras, introduciendo el tiempo en el que quiere
realizarse
el muestreo
*/
/*Declaracion de Variables*/
float Im; // Valor final de la intensidad
float Vout; // Valor de la tension medida en pinI
int i = 0; // Indicador para los bucles
int D = TmuestreoI*1000/nMuestrasI; // Valor del paso en microsegundos
float K = 0; // Coeficiente de correccion
//float Ianterior = Icont; //valor anterior medido de la corriente
/*Muestreo*/
do{
Vout = analogRead(pinI)+Vout;
delayMicroseconds(D);
i++;
}
while(i < nMuestrasI);
Vout = Vout/nMuestrasI;
/*Calculos*/
Im =(1/0.066*(Vout*5/1024-2.5)+K); /*El coeficiente de correccion se
aplica como suma
ya que el error es pracaticamente constante para
cada nivel de corriente y asi se consigue la mejor
compensacion posible*/
return Im;
}


/********************************************
 * Serial Communication functions / helpers
 ********************************************/
//A partir de aqui es todo para poder comunicarse con Processing,
extraido de la libreria PID_Front_End

union {                 // This Data structure lets
  byte asBytes[24];     // us take the byte array
  float asFloat[6];     // sent from processing and
}                       // easily convert it to a
foo;                    // float array



// getting float values from processing into the arduino
// was no small task.  the way this program does it is
// as follows:
//  * a float takes up 4 bytes.  in processing, convert
//    the array of floats we want to send, into an array
//    of bytes.
//  * send the bytes to the arduino
//  * use a data structure known as a union to convert
//    the array of bytes back into an array of floats
```

```
//  the bytes coming from the arduino follow the following
//  format:
//  0: 0=Manual, 1=Auto, else = ? error ?
//  1: 0=Direct, 1=Reverse, else = ? error ?
//  2-5: float setpoint
//  6-9: float input
//  10-13: float output
//  14-17: float P_Param
//  18-21: float I_Param
//  22-245: float D_Param


void SerialReceive()
{

  // read the bytes sent from Processing
  int index=0;
  byte Auto_Man = -1;
  byte Direct_Reverse = -1;
  while(Serial.available()&&index<26)
  {
    if(index==0) Auto_Man = Serial.read();
    else if(index==1) Direct_Reverse = Serial.read();
    else foo.asBytes[index-2] = Serial.read();
    index++;
  }

  // if the information we got was in the correct format,
  // read it into the system
  if(index==26  && (Auto_Man==0 || Auto_Man==1)&& (Direct_Reverse==0
|| Direct_Reverse==1))
  {
    ref=double(foo.asFloat[0]);
    //Input=double(foo.asFloat[1]);       // * the user has the
ability to send the
                                          //   value of "Input"  in
most cases (as
                                          //   in this one) this is
not needed.
    if(Auto_Man==0)                       // * only change the output
if we are in
    {                                     //   manual mode.  otherwise
we'll get an
     salidaPID=double(foo.asFloat[2]);      //   output blip, then the
controller will
    }                                     //   overwrite.

    double p, i, d;                       // * read in and set the
controller tunings
    p = double(foo.asFloat[3]);           //
    i = double(foo.asFloat[4]);           //
    d = double(foo.asFloat[5]);           //
    myPID.SetTunings(p, i, d);            //

    if(Auto_Man==0) myPID.SetMode(MANUAL);// * set the controller mode
    else myPID.SetMode(AUTOMATIC);            //
```

```
    if(Direct_Reverse==0) myPID.SetControllerDirection(DIRECT);// *
set the controller Direction
    else myPID.SetControllerDirection(REVERSE);          //
  }
  Serial.flush();                          // * clear any random data
from the serial buffer
}

// unlike our tiny microprocessor, the processing ap
// has no problem converting strings into floats, so
// we can just send strings.  much easier than getting
// floats from processing to here no?


void SerialSend()
{
  Serial.print("PID ");
  Serial.print(ref);
  Serial.print(" ");
  Serial.print(Urms);
  Serial.print(" ");
  Serial.print(DC);
  Serial.print(" ");
  Serial.print(myPID.GetKp());
  Serial.print(" ");
  Serial.print(myPID.GetKi());
  Serial.print(" ");
  Serial.print(myPID.GetKd());
  Serial.print(" ");
  if(myPID.GetMode()==AUTOMATIC) Serial.print("Automatic");
  else Serial.print("Manual");
  Serial.print(" ");
  if(myPID.GetDirection()==DIRECT) Serial.println("Direct");
  else Serial.println("Reverse");
}
```

Processing

```
/********************************************************
 * Arduino PID Tuning Front-End,  Version 0.3
 * by Brett Beauregard
 * License: Creative-Commons Attribution Share-Alike
 * April 2011
 *
 * This application is designed to interface with an
 * arduino running the PID Library.  From this Control
 * Panel you can observe & adjust PID performance in
 * real time
 *
 * The ControlP5 library is required to run this sketch.
 * files and install instructions can be found at
 * http://www.sojamo.de/libraries/controlP5/
 *
 ********************************************************/

import java.nio.ByteBuffer;
import processing.serial.*;
import controlP5.*;

/*********************************************
 * User spcification section
 *********************************************/
int windowWidth = 1350;       // set the size of the
int windowHeight = 700;      // form

float InScaleMin = 0;        // set the Y-Axis Min
float InScaleMax = 300;     // and Max for both
float OutScaleMin =0;       // the top and
float OutScaleMax =100;     // bottom trends


int windowSpan = 60000;     // number of mS into the past you want to
display
int refreshRate = 100;       // how often you want the graph to be
reDrawn;

//float displayFactor = 1; //display Time as Milliseconds
float displayFactor = 1000; //display Time as Seconds
//float displayFactor = 60000; //display Time as Minutes

String outputFileName = ""; // if you'd like to output data to
// a file, specify the path here

/*********************************************
 * end user spec
 *********************************************/

int nextRefresh;
int arrayLength = windowSpan / refreshRate+1;
int[] InputData = new int[arrayLength];     //we might not need them
this big, but
int[] SetpointData = new int[arrayLength];  // this is worst case
int[] OutputData = new int[arrayLength];
```

```
float inputTop = 25;
float inputHeight = (windowHeight-70)*2/3;
float outputTop = inputHeight+50;
float outputHeight = (windowHeight-70)*1/3;

float ioLeft = 150, ioWidth = windowWidth-ioLeft-50;
float ioRight = ioLeft+ioWidth;
float pointWidth= (ioWidth)/float(arrayLength-1);

int vertCount = 60;

int nPoints = 0;

float Input, Setpoint, Output;

boolean madeContact =false;
boolean justSent = true;

Serial myPort;

ControlP5 controlP5;
controlP5.Button AMButton, DRButton;
controlP5.Textlabel AMLabel, AMCurrent, InLabel,
OutLabel, SPLabel, PLabel,
ILabel, DLabel,DRLabel, DRCurrent;
controlP5.Textfield SPField, InField, OutField,
PField, IField, DField;

PrintWriter output;
PFont AxisFont, TitleFont;

void setup()
{
  frameRate(30);
  size(1350 , 700);

  println(Serial.list());                                          //
* Initialize Serial
  myPort = new Serial(this, Serial.list()[0], 9600);               //
Communication with
  myPort.bufferUntil(10);                                          //
the Arduino

  controlP5 = new ControlP5(this);                                 //
* Initialize the various
  SPField= controlP5.addTextfield("Setpoint",10,100,60,20);        //
Buttons, Labels, and
  InField = controlP5.addTextfield("Input",10,150,60,20);          //
Text Fields we'll be
  OutField = controlP5.addTextfield("Output",10,200,60,20);        //
using
  PField = controlP5.addTextfield("Kp (Proportional)",10,275,60,20);
//
  IField = controlP5.addTextfield("Ki (Integral)",10,325,60,20);
//
```

```
  DField = controlP5.addTextfield("Kd (Derivative)",10,375,60,20);
//
  AMButton = controlP5.addButton("Toggle_AM",0.0,10,50,60,20);       //
  AMLabel = controlP5.addTextlabel("AM","Manual",12,72);             //
  AMCurrent = controlP5.addTextlabel("AMCurrent","Manual",80,65);    //
  controlP5.addButton("Send_To_Arduino",0.0,10,475,120,20);          //
  SPLabel=controlP5.addTextlabel("SP","3",80,103);                   //
  InLabel=controlP5.addTextlabel("In","1",80,153);                   //
  OutLabel=controlP5.addTextlabel("Out","2",80,203);                 //
  PLabel=controlP5.addTextlabel("P","4",80,278);                     //
  ILabel=controlP5.addTextlabel("I","5",80,328);                     //
  DLabel=controlP5.addTextlabel("D","6",80,378);                     //
  DRButton = controlP5.addButton("Toggle_DR",0.0,10,425,60,20);
//
  DRLabel = controlP5.addTextlabel("DR","Direct",12,447);
//
  DRCurrent = controlP5.addTextlabel("DRCurrent","Direct",80,440);
//

  AxisFont = loadFont("axis.vlw");
  TitleFont = loadFont("Titles.vlw");

  nextRefresh=millis();
  if (outputFileName!="") output = createWriter(outputFileName);
}

void draw()
{
  background(200);
  drawGraph();
  drawButtonArea();
}

void drawGraph()
{
  //draw Base, gridlines
  stroke(0);
  fill(230);
  rect(ioLeft, inputTop,ioWidth-1 , inputHeight);
  rect(ioLeft, outputTop, ioWidth-1, outputHeight);
  stroke(210);

  //Section Titles
  textFont(TitleFont);
  fill(255);
  text("PID Input / Setpoint",(int)ioLeft+10,(int)inputTop-5);
  text("PID Output",(int)ioLeft+10,(int)outputTop-5);


  //GridLines and Titles
  textFont(AxisFont);

  //horizontal grid lines
  int interval = (int)inputHeight/15;
  for(int i=0;i<16;i++)
  {
```

```
    if(i>0&&i<15) line(ioLeft+1,inputTop+i*interval,ioRight-
2,inputTop+i*interval);
    text(str((InScaleMax-InScaleMin)/15*(float)(15-
i)+InScaleMin),ioRight+5,inputTop+i*interval+4);

  }
  interval = (int)outputHeight/8;
  for(int i=0;i<9;i++)
  {
    if(i>0&&i<8) line(ioLeft+1,outputTop+i*interval,ioRight-
2,outputTop+i*interval);
    text(str((OutScaleMax-OutScaleMin)/8*(float)(8-
i)+OutScaleMin),ioRight+5,outputTop+i*interval+4);
  }


  //vertical grid lines and TimeStamps
  int elapsedTime = millis();
  interval = (int)ioWidth/vertCount;
  int shift = elapsedTime*(int)ioWidth / windowSpan;
  shift %=interval;

  int iTimeInterval = windowSpan/vertCount;
  float firstDisplay =
(float)(iTimeInterval*(elapsedTime/iTimeInterval))/displayFactor;
  float timeInterval = (float)(iTimeInterval)/displayFactor;
  for(int i=0;i<vertCount;i++)
  {
    int x = (int)ioRight-shift-2-i*interval;

    line(x,inputTop+1,x,inputTop+inputHeight-1);
    line(x,outputTop+1,x,outputTop+outputHeight-1);

    float t = firstDisplay-(float)i*timeInterval;
    if(t>=0)  text(str(t),x,outputTop+outputHeight+10);
  }


  // add the latest data to the data Arrays.  the values need
  // to be massaged to get them to graph correctly.  they
  // need to be scaled to fit where they're going, and
  // because 0, 0 is the top left, we need to flip the values.
  // this is easier than having the user stand on their head
  // to read the graph.
  if(millis() > nextRefresh && madeContact)
  {
    nextRefresh += refreshRate;

    for(int i=nPoints-1;i>0;i--)
    {
      InputData[i]=InputData[i-1];
      SetpointData[i]=SetpointData[i-1];
      OutputData[i]=OutputData[i-1];
    }
    if (nPoints < arrayLength) nPoints++;
```

```
    InputData[0] = int(inputHeight)-int(inputHeight*(Input-
InScaleMin)/(InScaleMax-InScaleMin));
    SetpointData[0] =int( inputHeight)-int(inputHeight*(Setpoint-
InScaleMin)/(InScaleMax-InScaleMin));
    OutputData[0] = int(outputHeight)-int(outputHeight*(Output-
OutScaleMin)/(OutScaleMax-OutScaleMin));
  }
  //draw lines for the input, setpoint, and output
  strokeWeight(2);
  for(int i=0; i<nPoints-2; i++)
  {
    int X1 = int(ioRight-2-float(i)*pointWidth);
    int X2 = int(ioRight-2-float(i+1)*pointWidth);
    boolean y1Above, y1Below, y2Above, y2Below;


    //DRAW THE INPUT
    boolean drawLine=true;
    stroke(255,0,0);
    int Y1 = InputData[i];
    int Y2 = InputData[i+1];

    y1Above = (Y1>inputHeight);                    // if both points
are outside
    y1Below = (Y1<0);                              // the min or max,
don't draw the
    y2Above = (Y2>inputHeight);                    // line.  if only
one point is
    y2Below = (Y2<0);                              // outside
constrain it to the limit,
    if(y1Above)                                    // and leave the
other one untouched.
    {                                              //
      if(y2Above) drawLine=false;                  //
      else if(y2Below) {                           //
        Y1 = (int)inputHeight;                     //
        Y2 = 0;                                    //
      }                                            //
      else Y1 = (int)inputHeight;                  //
    }                                              //
    else if(y1Below)                               //
    {                                              //
      if(y2Below) drawLine=false;                  //
      else if(y2Above) {                           //
        Y1 = 0;                                    //
        Y2 = (int)inputHeight;                     //
      }                                            //
      else Y1 = 0;                                 //
    }                                              //
    else                                           //
    {                                              //
      if(y2Below) Y2 = 0;                          //
      else if(y2Above) Y2 = (int)inputHeight;      //
    }                                              //

    if(drawLine)
    {
```

```
      line(X1,Y1+inputTop, X2, Y2+inputTop);
    }

    //DRAW THE SETPOINT
    drawLine=true;
    stroke(0,255,0);
    Y1 = SetpointData[i];
    Y2 = SetpointData[i+1];

    y1Above = (Y1>(int)inputHeight);                // if both points
are outside
    y1Below = (Y1<0);                               // the min or max,
don't draw the
    y2Above = (Y2>(int)inputHeight);                // line.  if only
one point is
    y2Below = (Y2<0);                               // outside
constrain it to the limit,
    if(y1Above)                                     // and leave the
other one untouched.
    {                                               //
      if(y2Above) drawLine=false;                   //
      else if(y2Below) {                            //
        Y1 = (int)(inputHeight);                    //
        Y2 = 0;                                     //
      }                                             //
      else Y1 = (int)(inputHeight);                 //
    }                                               //
    else if(y1Below)                                //
    {                                               //
      if(y2Below) drawLine=false;                   //
      else if(y2Above) {                            //
        Y1 = 0;                                     //
        Y2 = (int)(inputHeight);                    //
      }                                             //
      else Y1 = 0;                                  //
    }                                               //
    else                                            //
    {                                               //
      if(y2Below) Y2 = 0;                           //
      else if(y2Above) Y2 = (int)(inputHeight);     //
    }                                               //

    if(drawLine)
    {
      line(X1, Y1+inputTop, X2, Y2+inputTop);
    }

    //DRAW THE OUTPUT
    drawLine=true;
    stroke(0,0,255);
    Y1 = OutputData[i];
    Y2 = OutputData[i+1];

    y1Above = (Y1>outputHeight);                   // if both points
are outside
    y1Below = (Y1<0);                              // the min or max,
don't draw the
```

```
    y2Above = (Y2>outputHeight);                    // line.  if only
one point is
    y2Below = (Y2<0);                               // outside
constrain it to the limit,
    if(y1Above)                                     // and leave the
other one untouched.
    {                                               //
      if(y2Above) drawLine=false;                   //
      else if(y2Below) {                            //
        Y1 = (int)outputHeight;                     //
        Y2 = 0;                                     //
      }                                             //
      else Y1 = (int)outputHeight;                  //
    }                                               //
    else if(y1Below)                                //
    {                                               //
      if(y2Below) drawLine=false;                   //
      else if(y2Above) {                            //
        Y1 = 0;                                     //
        Y2 = (int)outputHeight;                     //
      }                                             //
      else Y1 = 0;                                  //
    }                                               //
    else                                            //
    {                                               //
      if(y2Below) Y2 = 0;                           //
      else if(y2Above) Y2 = (int)outputHeight;      //
    }                                               //

    if(drawLine)
    {
      line(X1, outputTop + Y1, X2, outputTop + Y2);
    }
  }
  strokeWeight(1);
}

void drawButtonArea()
{
  stroke(0);
  fill(100);
  rect(0, 0, ioLeft, windowHeight);
}

void Toggle_AM() {
  if(AMLabel.get().getText()=="Manual")
  {
    AMLabel.setValue("Automatic");
  }
  else
  {
    AMLabel.setValue("Manual");
  }
}


void Toggle_DR() {
```

```
  if(DRLabel.get().getText()=="Direct")
  {
    DRLabel.setValue("Reverse");
  }
  else
  {
    DRLabel.setValue("Direct");
  }
}

// Sending Floating point values to the arduino
// is a huge pain.  if anyone knows an easier
// way please let know.  the way I'm doing it:
// - Take the 6 floats we need to send and
//   put them in a 6 member float array.
// - using the java ByteBuffer class, convert
//   that array to a 24 member byte array
// - send those bytes to the arduino
void Send_To_Arduino()
{
  float[] toSend = new float[6];

  toSend[0] = float(SPField.getText());
  toSend[1] = float(InField.getText());
  toSend[2] = float(OutField.getText());
  toSend[3] = float(PField.getText());
  toSend[4] = float(IField.getText());
  toSend[5] = float(DField.getText());
  Byte a = (AMLabel.get().getText()=="Manual")?(byte)0:(byte)1;
  Byte d = (DRLabel.get().getText()=="Direct")?(byte)0:(byte)1;
  myPort.write(a);
  myPort.write(d);
  myPort.write(floatArrayToByteArray(toSend));
  justSent=true;
}


byte[] floatArrayToByteArray(float[] input)
{
  int len = 4*input.length;
  int index=0;
  byte[] b = new byte[4];
  byte[] out = new byte[len];
  ByteBuffer buf = ByteBuffer.wrap(b);
  for(int i=0;i<input.length;i++)
  {
    buf.position(0);
    buf.putFloat(input[i]);
    for(int j=0;j<4;j++) out[j+i*4]=b[3-j];
  }
  return out;
}


//take the string the arduino sends us and parse it
void serialEvent(Serial myPort)
{
```

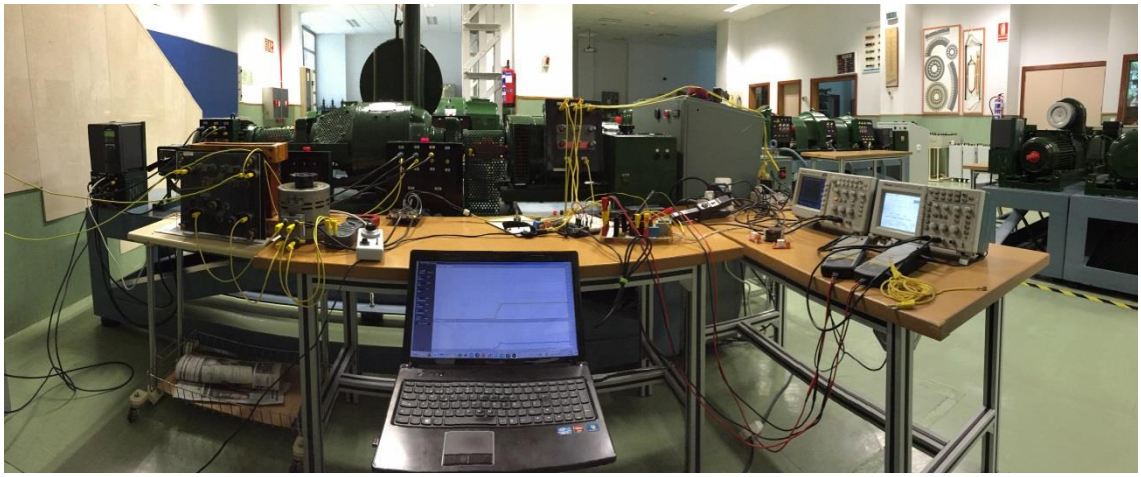
```
  String read = myPort.readStringUntil(10);
  if(outputFileName!="") output.print(str(millis())+ " "+read);
  String[] s = split(read, " ");

  if (s.length ==9)
  {
    Setpoint = float(s[1]);           // * pull the information
    Input = float(s[2]);              //   we need out of the
    Output = float(s[3]);             //   string and put it
    SPLabel.setValue(s[1]);           //   where it's needed
    InLabel.setValue(s[2]);           //
    OutLabel.setValue(trim(s[3]));    //
    PLabel.setValue(trim(s[4]));      //
    ILabel.setValue(trim(s[5]));      //
    DLabel.setValue(trim(s[6]));      //
    AMCurrent.setValue(trim(s[7]));   //
    DRCurrent.setValue(trim(s[8]));
    if(justSent)                      // * if this is the first read
    {                                 //   since we sent values to
      SPField.setText(trim(s[1]));    //   the arduino,  take the
      InField.setText(trim(s[2]));    //   current values and put
      OutField.setText(trim(s[3]));   //   them into the input fields
      PField.setText(trim(s[4]));     //
      IField.setText(trim(s[5]));     //
      DField.setText(trim(s[6]));     //
     // mode = trim(s[7]);              //
      AMLabel.setValue(trim(s[7]));         //
      //dr = trim(s[8]);                //
      DRCurrent.setValue(trim(s[8]));         //
      justSent=false;                 //
    }                                 //

    if(!madeContact) madeContact=true;
  }
}
```

Código adicional Arduino

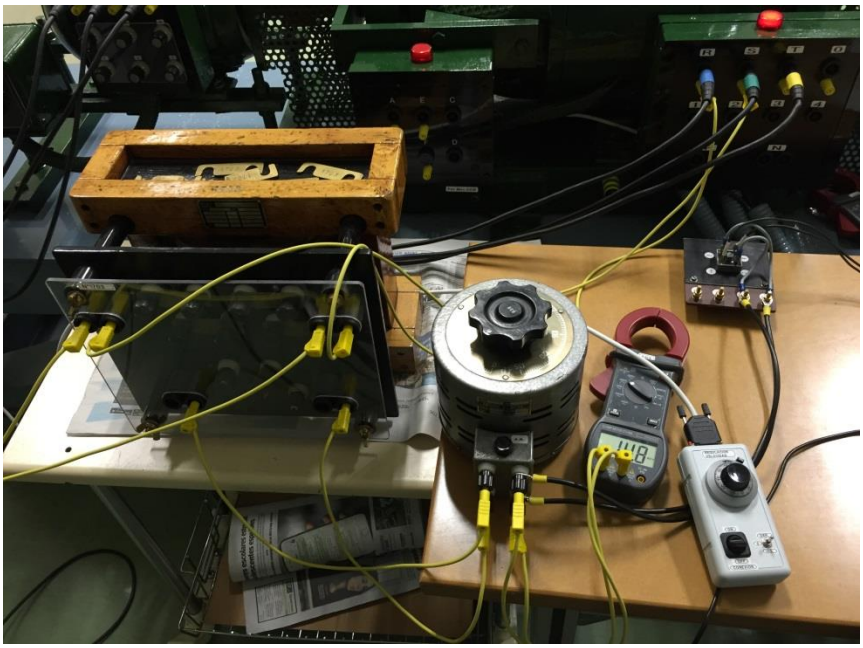
```
int pinC1= 2; //Orden excitación contactor 1
int pinC2= 4; //Orden excitación contactor 2
int pinC3= 7; //Orden excitación contactor 3

 pinMode(pinC1, OUTPUT); //Contactor1
 pinMode(pinC2, OUTPUT); //Contactor2
 pinMode(pinC3, OUTPUT); //Contactor3


//Eleccion del nivel de tension

if (valorS>200){ //si el duty cycle necesita ser mayor que el 80%
  nivel=nivel+1; //cambia a un nivel de tension superior
}

if (valorS<50){ //si el duty cycle necesita sermenor que el 20%
  nivel=nivel-1; //cambia a un nivel de tension inferior
}

if (nivel>3){
  nivel=3; //satura el nivel a 3 (60V)
}

if (nivel<0){
  nivel=0; //satura el nivel a 0 (15V)
}


//En función del nivel de tensión que toca excitar los contactores necesarios
switch (nivel){
  case 0:
  digitalWrite(pinC1, LOW);
  digitalWrite(pinC2, LOW);
  digitalWrite(pinC3, LOW);
  break;

  case 1:
  digitalWrite(pinC1, HIGH);
  digitalWrite(pinC2, HIGH);
  digitalWrite(pinC3, LOW);
  break;

  case 2:
  digitalWrite(pinC1, LOW);
  digitalWrite(pinC2, HIGH);
  digitalWrite(pinC3, HIGH);
  break;

  case 3:
  digitalWrite(pinC1, HIGH);
  digitalWrite(pinC2, HIGH);
  digitalWrite(pinC3, HIGH);
  break;
}
```

Anexo III: Fotografías

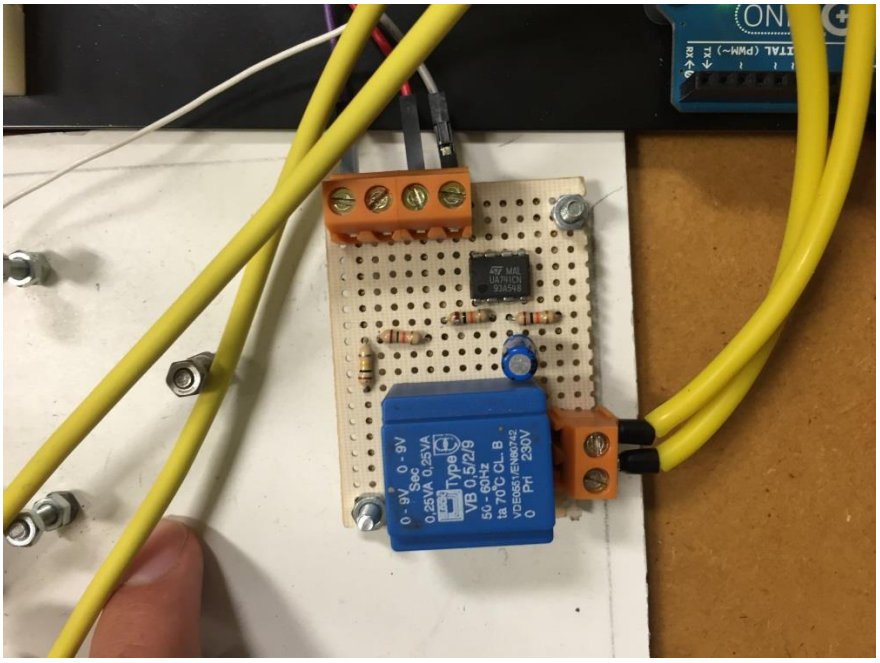
Conjunto



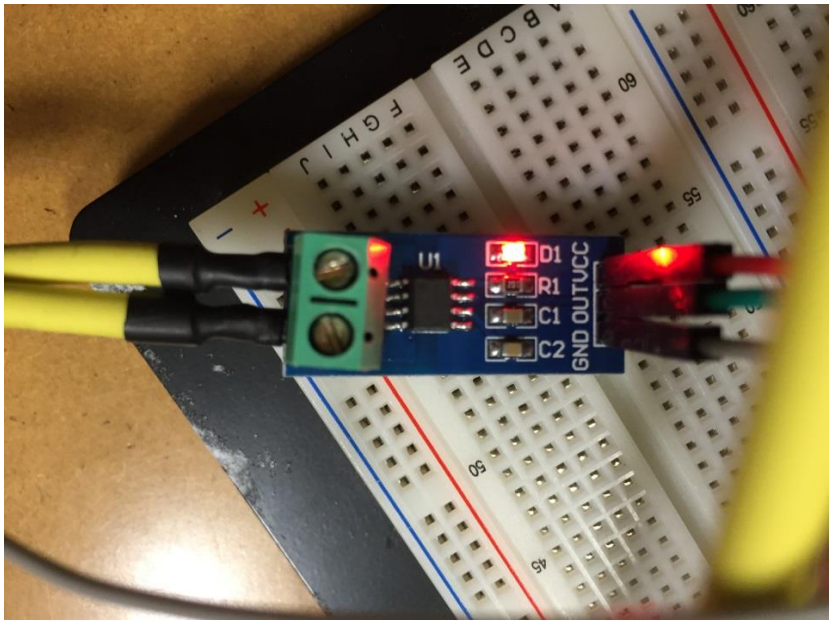
Alimentación de la excitación



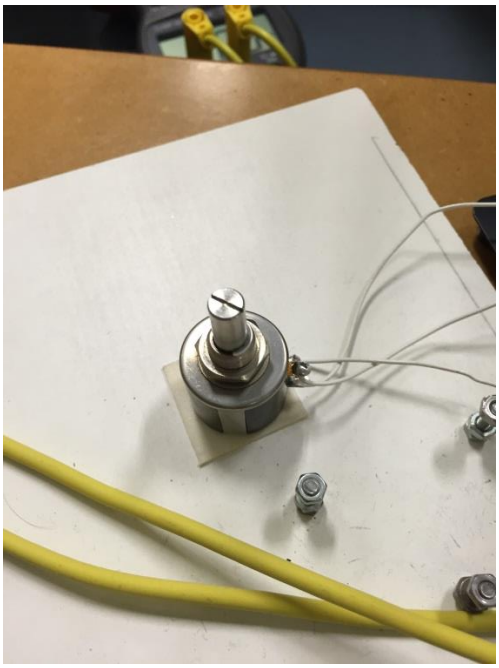
Tarjeta de medida de tensión



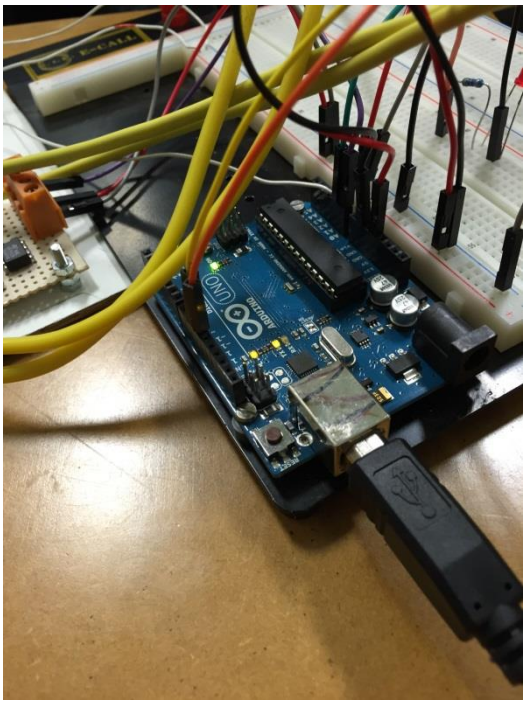
Transductor de corriente



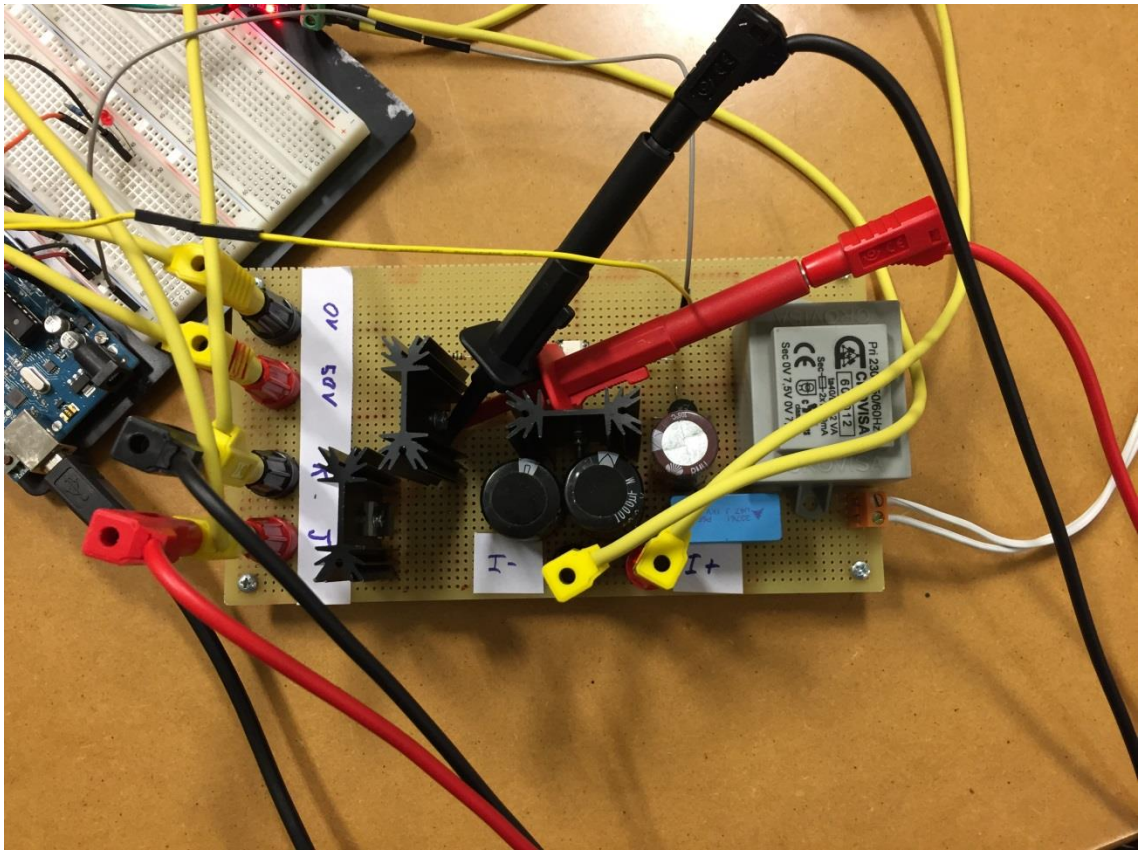
Potenciómetro



Placa Arduino UNO



Tarjeta de potencia

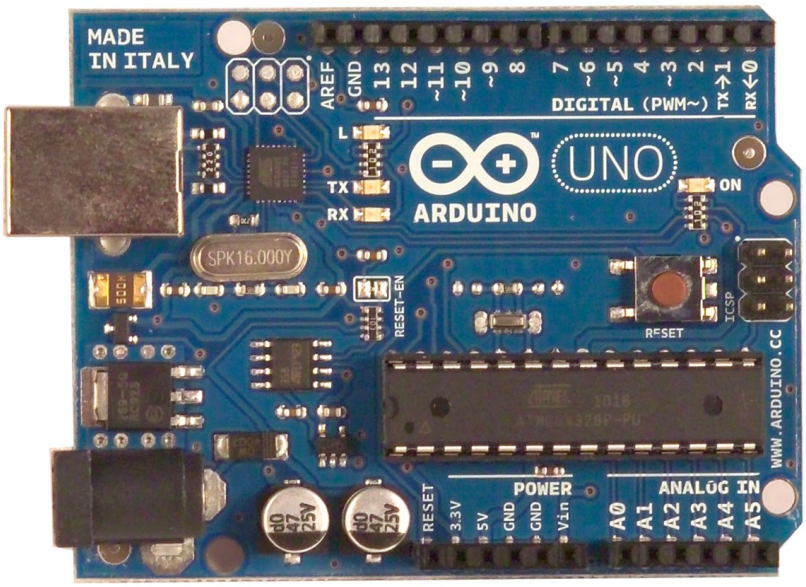


Anexo IV: Datasheets

Se han incluido los datasheets de los componentes que se consideran más relevantes o menos comunes. Podra encontrarse el datasheet de la placa Arduino, el transistor IGBT, el optoacoplador, el diodo rápido y el transductor de corriente. Datasheets de elementos más comunes y estándar como el regulador lineal 7815, el amplificador operacional UA741 o los puentes de diodos y transformadores no se han incluido.

(Comienzan en la página siguiente)

Arduino UNO



Product Overview

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduno, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

Index

Technical Specifications — Page 2

How to use Arduino
Programming Enviroment, Basic Tutorials — Page 6

Terms & Conditions — Page 7

Enviromental Policies
half sqm of green via Impatto Zero® — Page 7

RS radiospares RADIONICS ALLIED ELECTRONICS

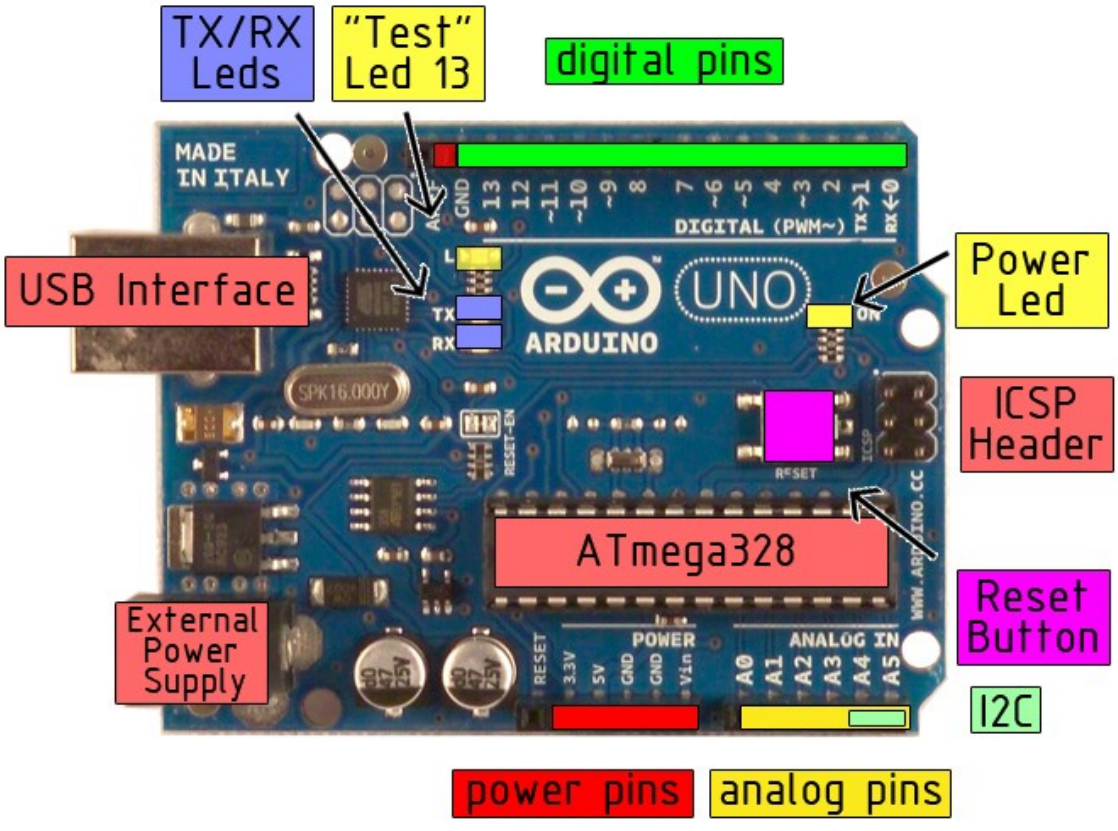
Technical Specification

EAGLE files: arduino-duemilanove-uno-design.zip Schematic: arduino-uno-schematic.pdf

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

the board



Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The Atmega328 has 32 KB of flash memory for storing code (of which 0,5 KB is used for the bootloader); It has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. TThese pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the analogWrite() function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.

- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

RS radiospares RADIONICS ALLIED ELECTRONICS AN ELECTROCOMPONENTS COMPANY

The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

- **I²C: 4 (SDA) and 5 (SCL).** Support I²C (TWI) communication using the Wire library.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with analogReference().
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the mapping between Arduino pins and Atmega328 ports.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an *.inf file is required..

The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Uno's digital pins.

The ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. To use the SPI communication, please see the ATmega328 datasheet.

Programming

The Arduino Uno can be programmed with the Arduino software (download). Select "Arduino Uno w/ ATmega328" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the reference and tutorials.

The ATmega328 on the Arduino Uno comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.

The ATmega8U2 firmware source code is available . The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader).

RS radiospares RADIONICS ALLIED ELECTRONICS

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

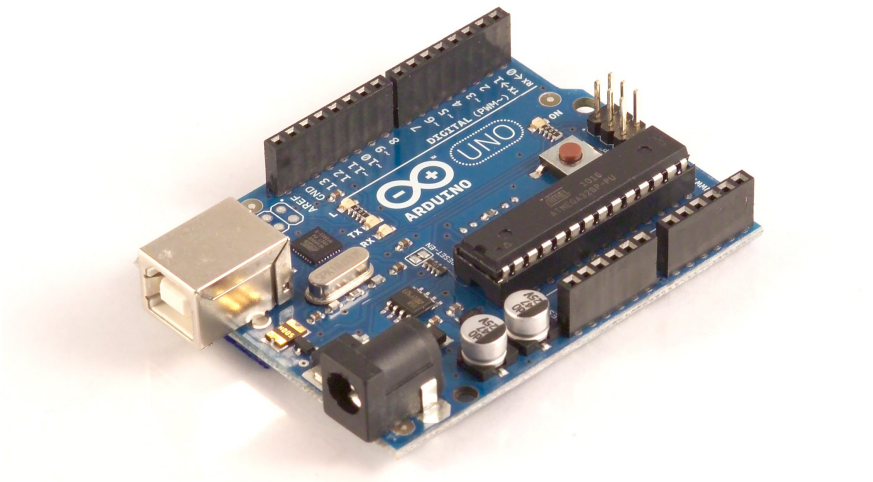
The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



RS radiospares RADIONICS ALLIED ELECTRONICS

How to use Arduino

Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platoform program. You'll have to follow different instructions for your personal OS. Check on the Arduino site for the latest instructions. *http://arduino.cc/en/Guide/HomePage*

Linux Install Windows Install Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

Blink led

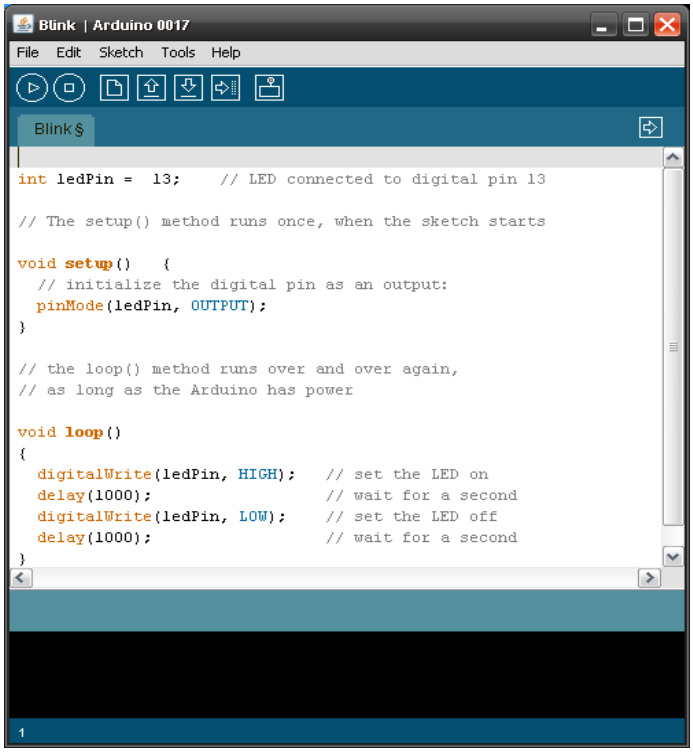
Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

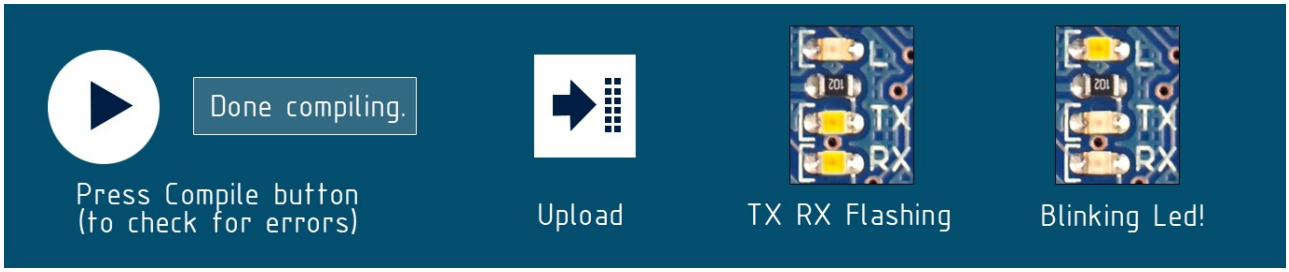
File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink

Once you have your skecth you'll see something very close to the screenshot on the right.

In **Tools>Board** select

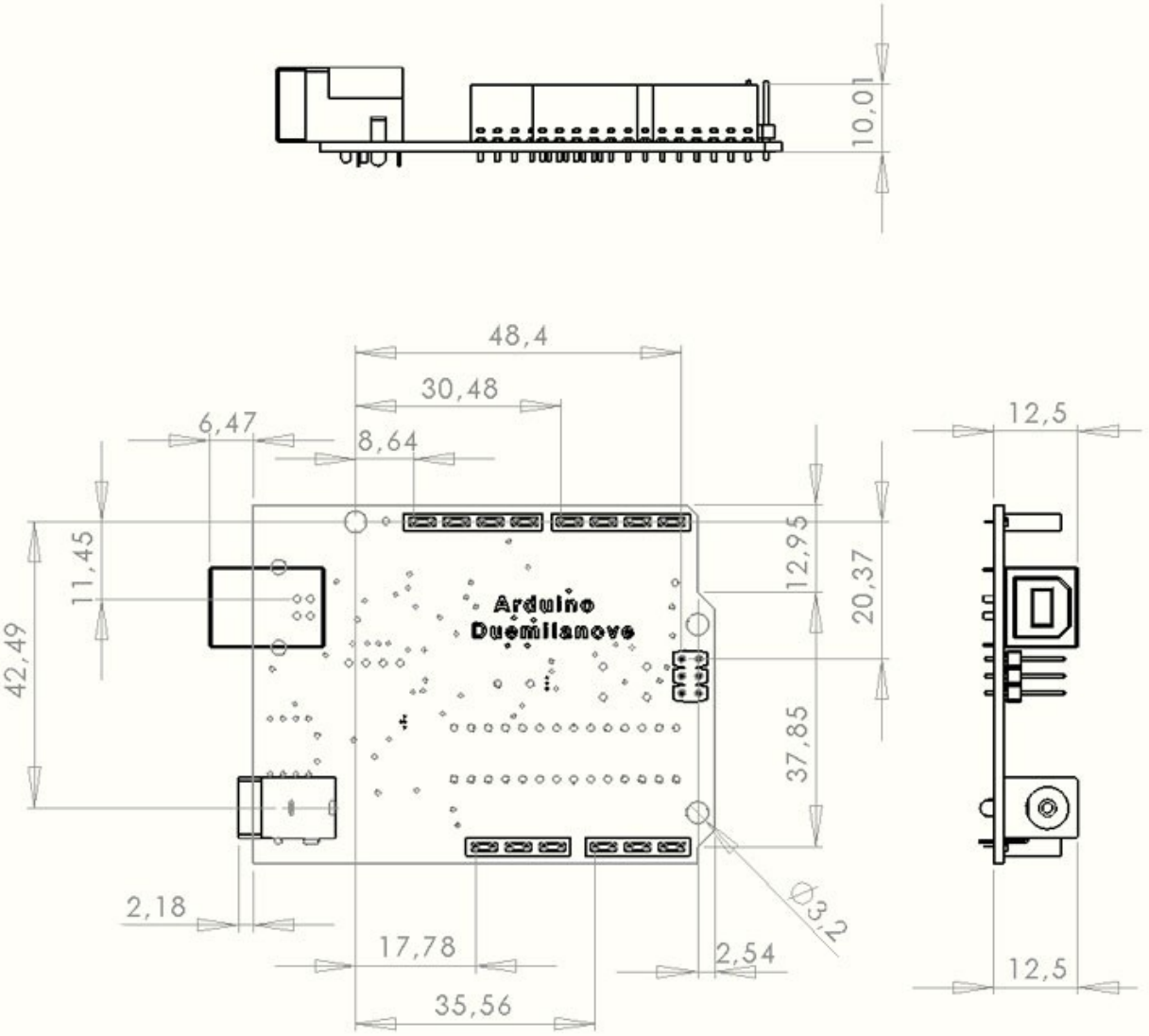
Now you have to go to
Tools>SerialPort
and select the right serial port, the one arduino is attached to.

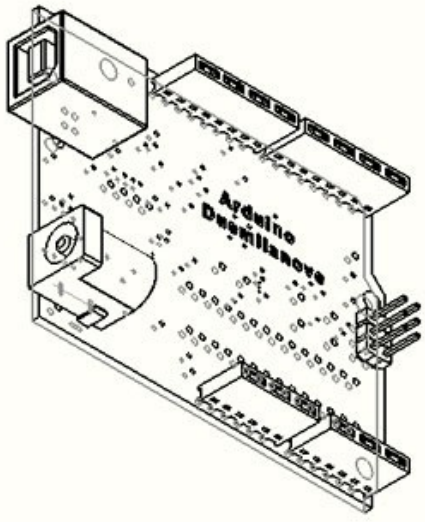




radiospares RADIONICS ALLIED ELECTRONICS

Dimensioned Drawing





RS radiospares RADIONICS ALLIED ELECTRONICS

Terms & Conditions

1. Warranties

1.1 The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mistreatment by the Customer, including improper installation or testing, or for any products that have been altered or modified in any way by a Customer. Moreover, The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the producer elects to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

1.4 Customer agrees that prior to using any systems that include the producer products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, applications or design advice, quality characterization, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.

1.5 The Arduino™ products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino™ products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Arduino™ products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino™ products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.

2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs to the extent directly caused by: (i) an actual breach by a Customer of the representation and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

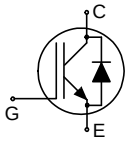
4. Changes to specifications

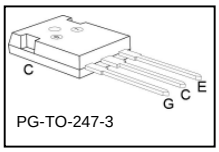
The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.

Enviromental Policies

The producer of Arduino™ has joined the Impatto Zero® policy of LifeGate.it. For each Arduino board produced is created / looked after half squared Km of Costa Rica's forest's.

Low Loss DuoPack : IGBT in **TrenchStop®** and Fieldstop technology with soft, fast recovery anti-parallel Emitter Controlled HE diode





- Approx. 1.0V reduced $V_{CE(sat)}$ and 0.5V reduced V_F compared to BUP305D
- Short circuit withstand time – 10µs
- Designed for :
  - Frequency Converters
  - Uninterrupted Power Supply
- **TrenchStop®** and Fieldstop technology for 1200 V applications offers :
  - very tight parameter distribution
  - high ruggedness, temperature stable behavior
- NPT technology offers easy parallel switching capability due to positive temperature coefficient in $V_{CE(sat)}$
- Low EMI
- Low Gate Charge
- Very soft, fast recovery anti-parallel Emitter Controlled HE diode
- Qualified according to JEDEC[1] for target applications
- Pb-free lead plating; RoHS compliant
- Complete product spectrum and PSpice Models : http://www.infineon.com/igbt/

Type	V_{CE}	I_C	$V_{CE(sat),Tj=25°C}$	$T_{j,max}$	Marking Code	Package
IKW08T120	1200V	8A	1.7V	150°C	K08T120	PG-TO-247-3

Maximum Ratings

Parameter	Symbol	Value	Unit
Collector-emitter voltage	V_{CE}	1200	V
DC collector current T_C = 25°C T_C = 100°C	I_C	 16 8	A
Pulsed collector current, t_p limited by T_{jmax}	I_{Cpuls}	24	
Turn off safe operating area $V_{CE} \le 1200V$, $T_j \le 150°C$	-	24	
Diode forward current T_C = 25°C T_C = 100°C	I_F	 16 8	
Diode pulsed current, t_p limited by T_{jmax}	I_{Fpuls}	24	
Gate-emitter voltage	V_{GE}	±20	V
Short circuit withstand time[2)] V_{GE} = 15V, $V_{CC} \le 1200V$, $T_j \le 150°C$	t_{SC}	10	µs
Power dissipation T_C = 25°C	P_{tot}	70	W
Operating junction temperature	T_j	-40...+150	°C
Storage temperature	T_{stg}	-55...+150	

[1] J-STD-020 and JESD-022
[2)] Allowed number of short circuits: <1000; time between short circuits: >1s.

Soldering temperature, 1.6mm (0.063 in.) from case for 10s	-	260	

Thermal Resistance

Parameter	Symbol	Conditions	Max. Value	Unit
Characteristic				
IGBT thermal resistance, junction – case	R_{thJC}		1.7	K/W
Diode thermal resistance, junction – case	R_{thJCD}		2.3	
Thermal resistance, junction – ambient	R_{thJA}		40	

Electrical Characteristic, at T_j = 25 °C, unless otherwise specified

Parameter	Symbol	Conditions	Value			Unit
			min.	typ.	max.	
Static Characteristic						
Collector-emitter breakdown voltage	$V_{(BR)CES}$	V_{GE}=0V, I_C=0.5mA	1200	-	-	V
Collector-emitter saturation voltage	$V_{CE(sat)}$	V_{GE} = 15V, I_C=8A				
		T_j=25°C	-	1.7	2.2	
		T_j=125°C	-	2.0	-	
		T_j=150°C	-	2.2	-	
Diode forward voltage	V_F	V_{GE}=0V, I_F=8A				
		T_j=25°C	-	1.7	2.2	
		T_j=125°C	-	1.7	-	
		T_j=150°C	-	1.7	-	
Gate-emitter threshold voltage	$V_{GE(th)}$	I_C=0.3mA, V_{CE}=V_{GE}	5.0	5.8	6.5	
Zero gate voltage collector current	I_{CES}	V_{CE}=1200V, V_{GE}=0V				mA
		T_j=25°C	-	-	0.2	
		T_j=150°C	-	-	2.0	
Gate-emitter leakage current	I_{GES}	V_{CE}=0V, V_{GE}=20V	-	-	100	nA
Transconductance	g_{fs}	V_{CE}=20V, I_C=8A	-	5	-	S
Integrated gate resistor	R_{Gint}		none			Ω

Dynamic Characteristic

Input capacitance	C_{iss}	V_{CE}=25V,	-	600	-	pF
Output capacitance	C_{oss}	V_{GE}=0V,	-	36	-	
Reverse transfer capacitance	C_{rss}	f=1MHz	-	28	-	
Gate charge	Q_{Gate}	V_{CC}=960V, I_C=8A V_{GE}=15V	-	53	-	nC
Internal emitter inductance measured 5mm (0.197 in.) from case	L_E		-	13	-	nH
Short circuit collector current[1)]	$I_{C(SC)}$	V_{GE}=15V,t_{SC}≤10µs V_{CC} = 600V, T_j = 25°C	-	48	-	A

Switching Characteristic, Inductive Load, at T_j=25 °C

Parameter	Symbol	Conditions	Value			Unit
			min.	typ.	max.	
IGBT Characteristic						
Turn-on delay time	$t_{d(on)}$	T_j=25°C, V_{CC}=600V,I_C=8A, V_{GE}=0/15V, R_G=81Ω, L_σ[2)]=180nH, C_σ[2)]=39pF Energy losses include “tail” and diode reverse recovery.	-	40	-	ns
Rise time	t_r		-	23	-	
Turn-off delay time	$t_{d(off)}$		-	450	-	
Fall time	t_f		-	70	-	
Turn-on energy	E_{on}		-	0.67	-	mJ
Turn-off energy	E_{off}		-	0.7	-	
Total switching energy	E_{ts}		-	1.37	-	
Anti-Parallel Diode Characteristic						
Diode reverse recovery time	t_{rr}	T_j=25°C, V_R=600V, I_F=8A, di_F/dt=600A/µs	-	80	-	ns
Diode reverse recovery charge	Q_{rr}		-	1.0	-	µC
Diode peak reverse recovery current	I_{rrm}		-	13	-	A
Diode peak rate of fall of reverse recovery current during t_b	di_{rr}/dt		-	420	-	A/µs

[1)] Allowed number of short circuits: <1000; time between short circuits: >1s.
[2)] Leakage inductance L_σ and Stray capacity C_σ due to dynamic test circuit in Figure E.

Switching Characteristic, Inductive Load, at T_j=150 °C

Parameter	Symbol	Conditions	Value			Unit
			min.	typ.	max.	
IGBT Characteristic						
Turn-on delay time	$t_{d(on)}$	T_j=150°C, V_{CC}=600V, I_C=8A, V_{GE}=0/15V, R_G= 81Ω, L_σ [1)]=180nH, C_σ [1)]=39pF Energy losses include “tail” and diode reverse recovery.	-	40	-	ns
Rise time	t_r		-	26	-	
Turn-off delay time	$t_{d(off)}$		-	570	-	
Fall time	t_f		-	140	-	
Turn-on energy	E_{on}		-	1.08	-	mJ
Turn-off energy	E_{off}		-	1.2	-	
Total switching energy	E_{ts}		-	2.28	-	
Anti-Parallel Diode Characteristic						
Diode reverse recovery time	t_{rr}	T_j=150°C V_R=600V, I_F=8A, di_F/dt=600A/µs	-	200	-	ns
Diode reverse recovery charge	Q_{rr}		-	2.3	-	µC
Diode peak reverse recovery current	I_{rrm}		-	20	-	A
Diode peak rate of fall of reverse recovery current during t_b	di_{rr}/dt		-	320	-	A/µs

[1)] Leakage inductance L_σ and Stray capacity C_σ due to dynamic test circuit in Figure E.

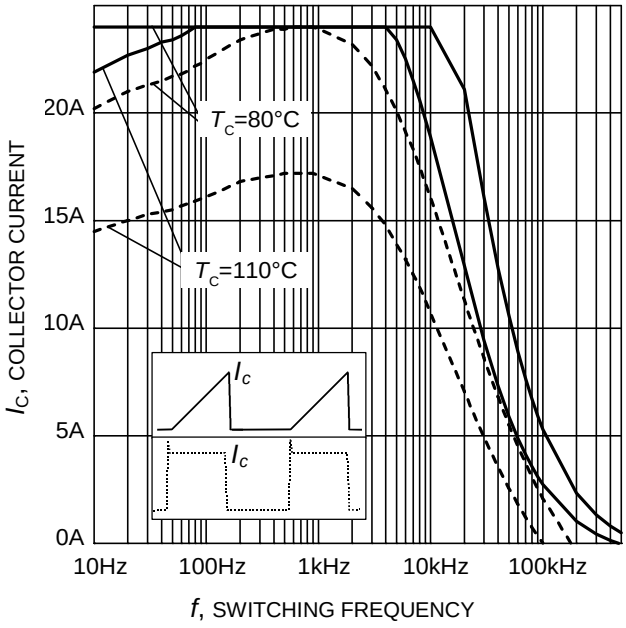


Figure 1. Collector current as a function of switching frequency
($T_j \leq 150°C$, $D = 0.5$, $V_{CE} = 600V$, $V_{GE} = 0/+15V$, $R_G = 81\Omega$)

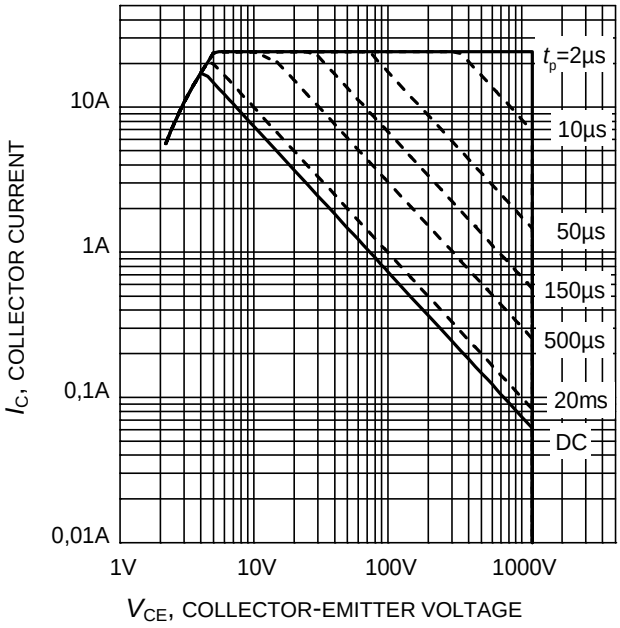


Figure 2. Safe operating area
($D = 0$, $T_C = 25°C$, $T_j \leq 150°C$; V_{GE}=15V)

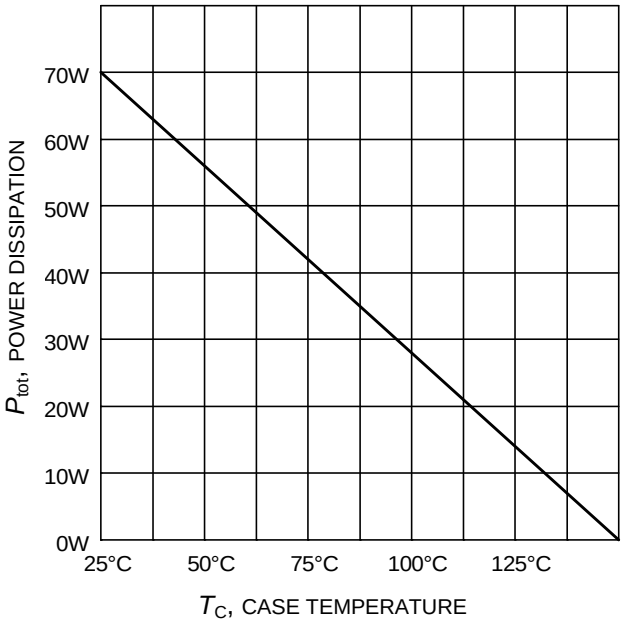


Figure 3. Power dissipation as a function of case temperature
($T_j \leq 150°C$)

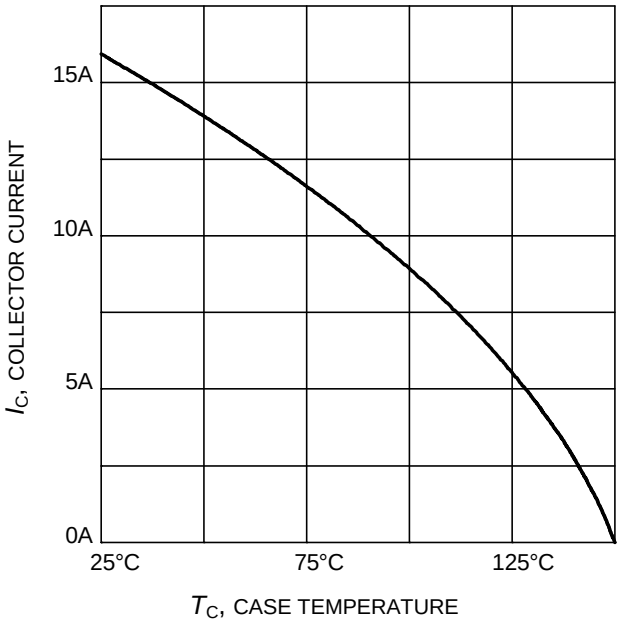


Figure 4. Collector current as a function of case temperature
($V_{GE} \geq 15V$, $T_j \leq 150°C$)

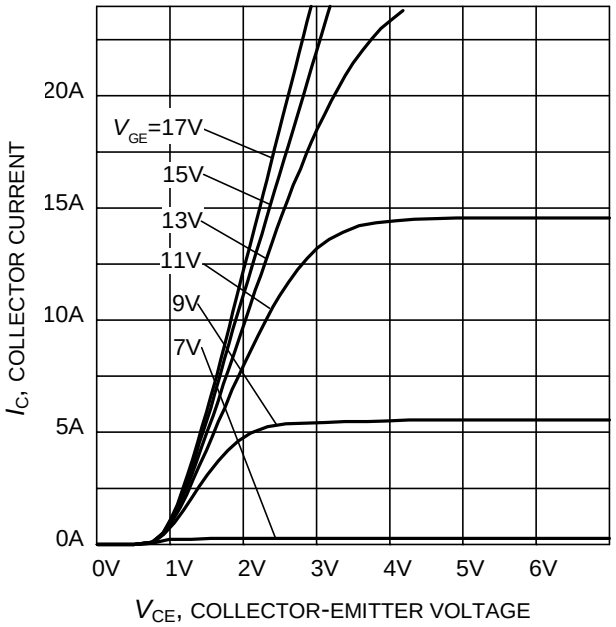


Figure 5. Typical output characteristic
(T_j = 25°C)

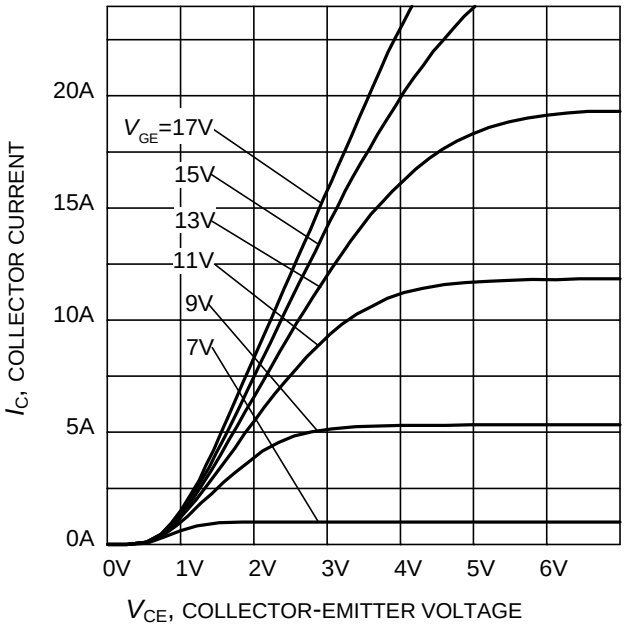


Figure 6. Typical output characteristic
(T_j = 150°C)

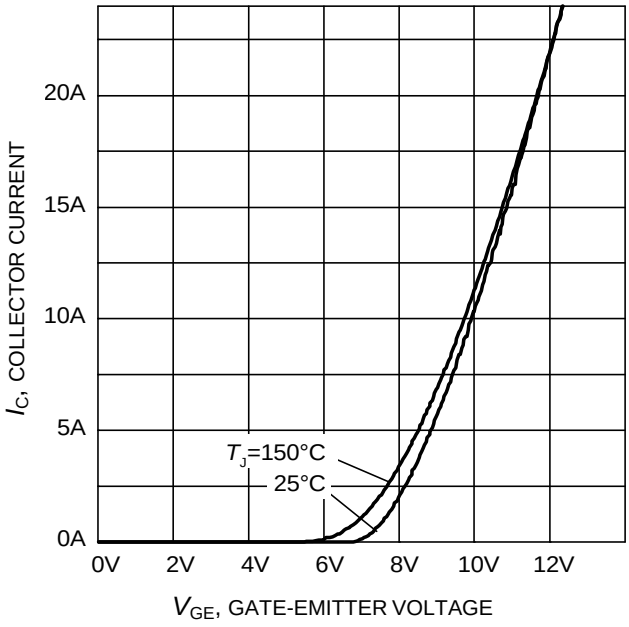


Figure 7. Typical transfer characteristic
(V_{CE}=20V)

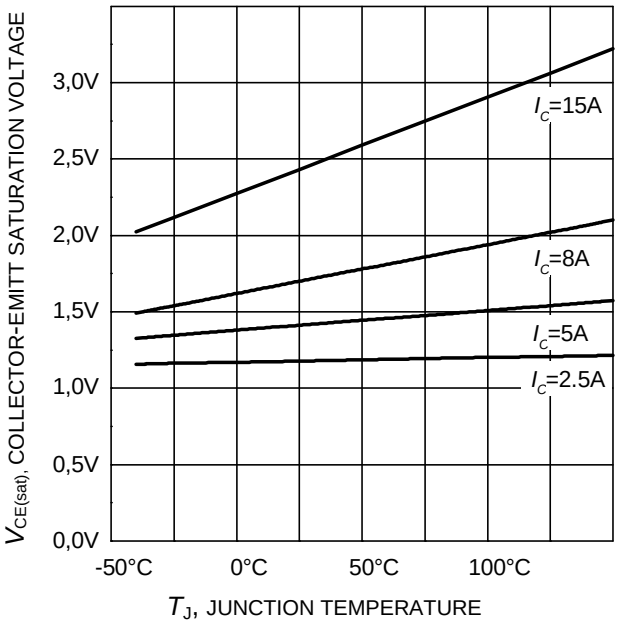


Figure 8. Typical collector-emitter saturation voltage as a function of junction temperature
(V_{GE} = 15V)

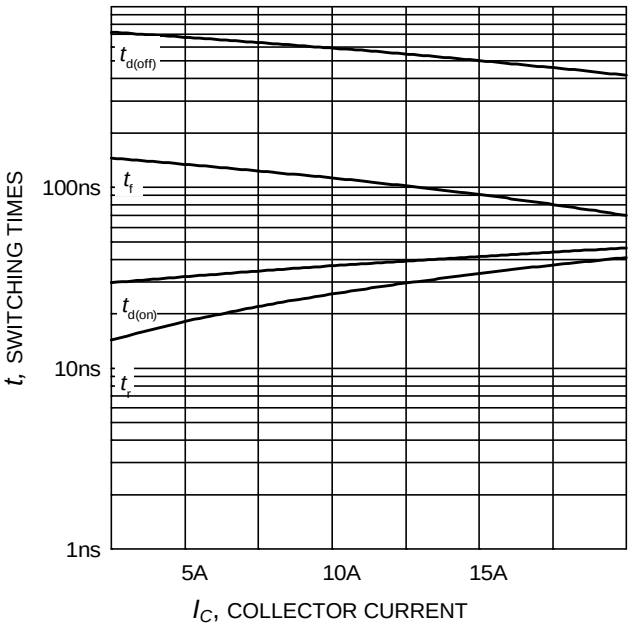


Figure 9. Typical switching times as a function of collector current
(inductive load, T_J=150°C, V_{CE}=600V, V_{GE}=0/15V, R_G=81Ω, Dynamic test circuit in Figure E)

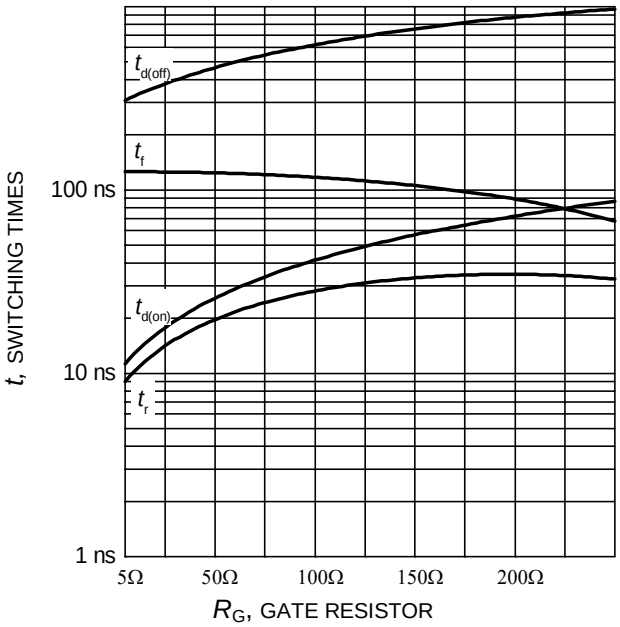


Figure 10. Typical switching times as a function of gate resistor
(inductive load, T_J=150°C, V_{CE}=600V, V_{GE}=0/15V, I_C=8A, Dynamic test circuit in Figure E)

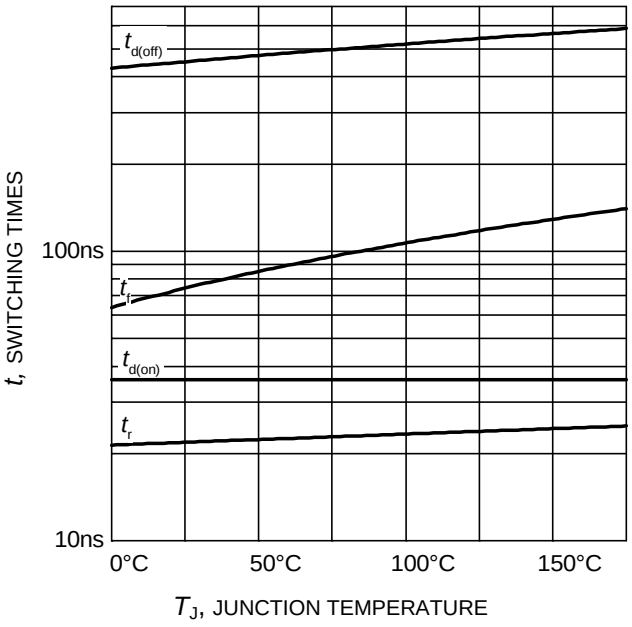


Figure 11. Typical switching times as a function of junction temperature
(inductive load, V_{CE}=600V, V_{GE}=0/15V, I_C=8A, R_G=81Ω, Dynamic test circuit in Figure E)

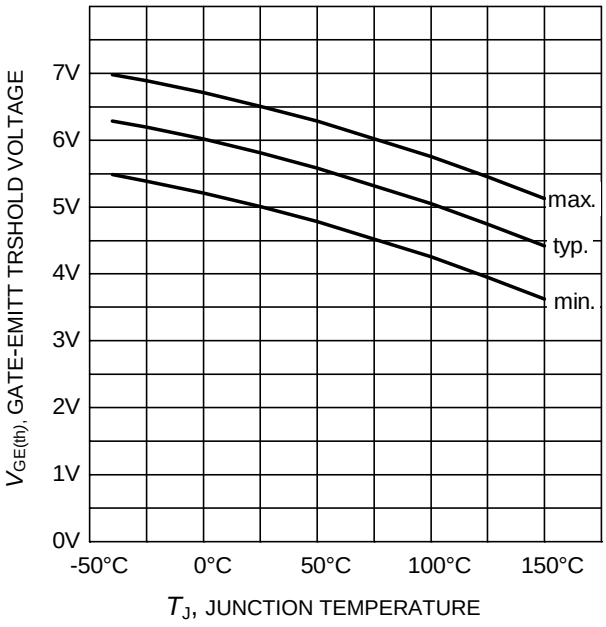


Figure 12. Gate-emitter threshold voltage as a function of junction temperature
(I_C = 0.3mA)

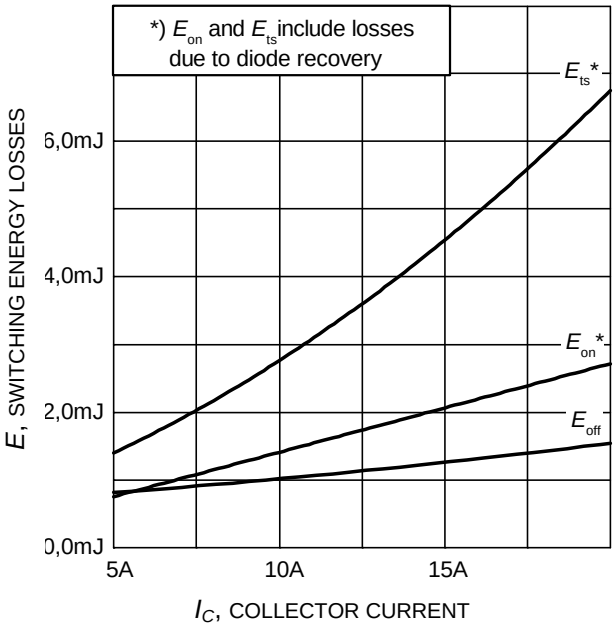


Figure 13. Typical switching energy losses as a function of collector current
(inductive load, T_J=150°C, V_{CE}=600V, V_{GE}=0/15V, R_G=81Ω, Dynamic test circuit in Figure E)

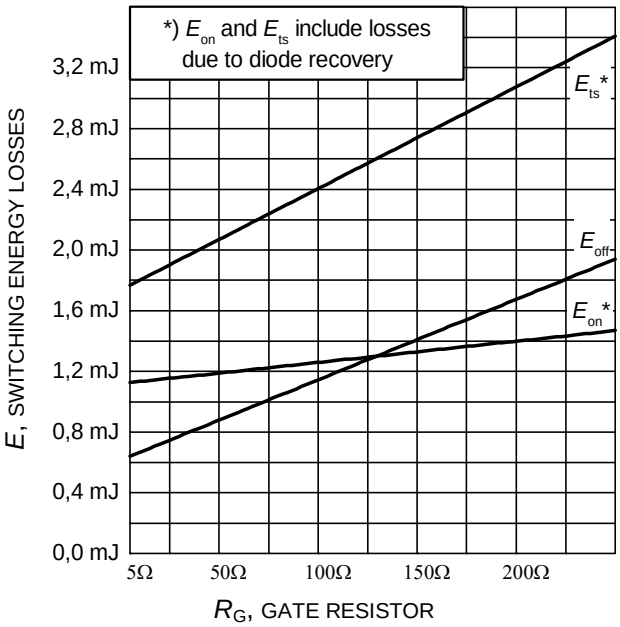


Figure 14. Typical switching energy losses as a function of gate resistor
(inductive load, T_J=150°C, V_{CE}=600V, V_{GE}=0/15V, I_C=8A, Dynamic test circuit in Figure E)

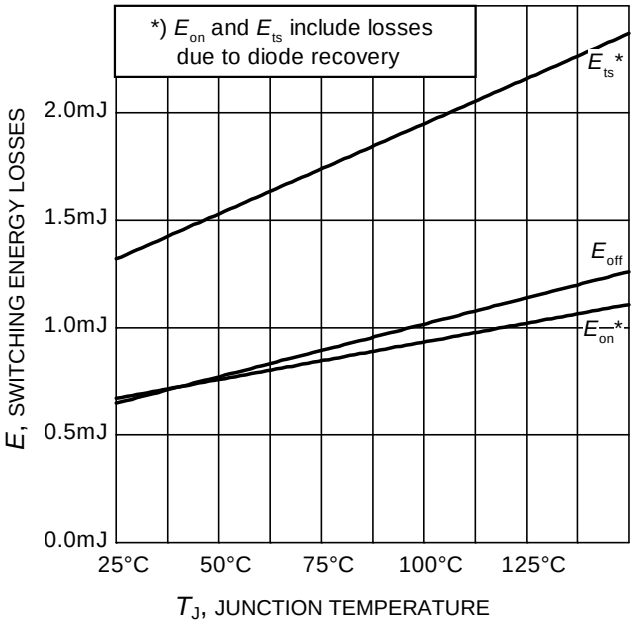


Figure 15. Typical switching energy losses as a function of junction temperature
(inductive load, V_{CE}=600V, V_{GE}=0/15V, I_C=8A, R_G=81Ω, Dynamic test circuit in Figure E)

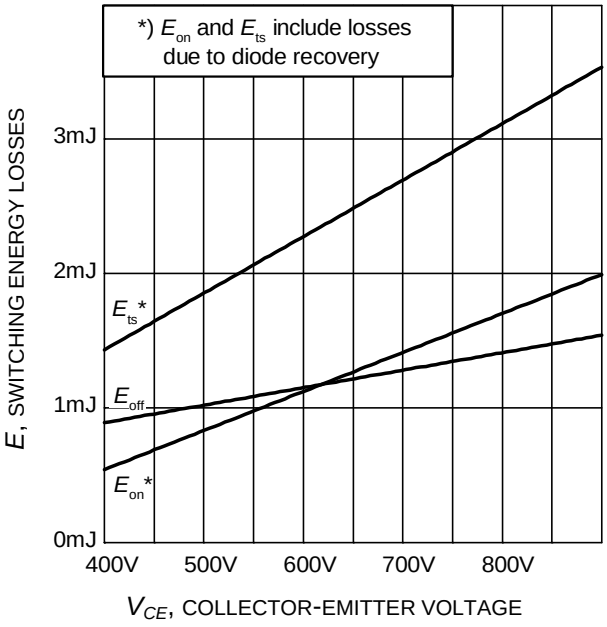


Figure 16. Typical switching energy losses as a function of collector emitter voltage
(inductive load, T_J=150°C, V_{GE}=0/15V, I_C=8A, R_G=81Ω, Dynamic test circuit in Figure E)

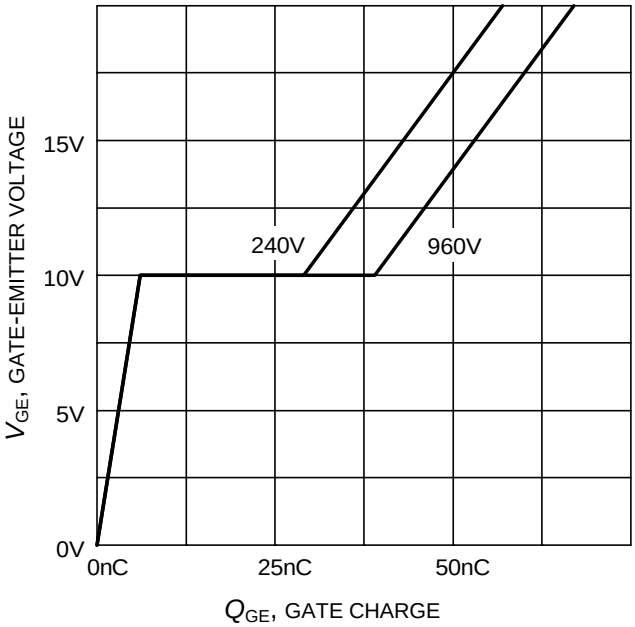


Figure 17. Typical gate charge
(I_C=8 A)

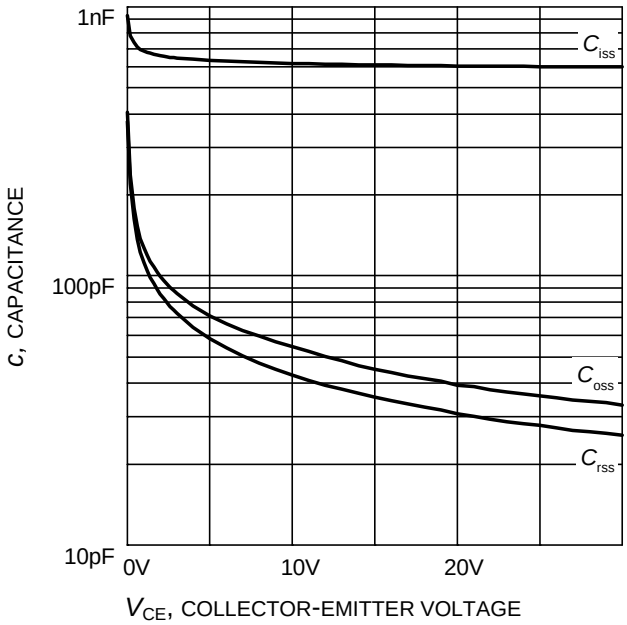


Figure 18. Typical capacitance as a function of collector-emitter voltage
(V_{GE}=0V, f = 1 MHz)

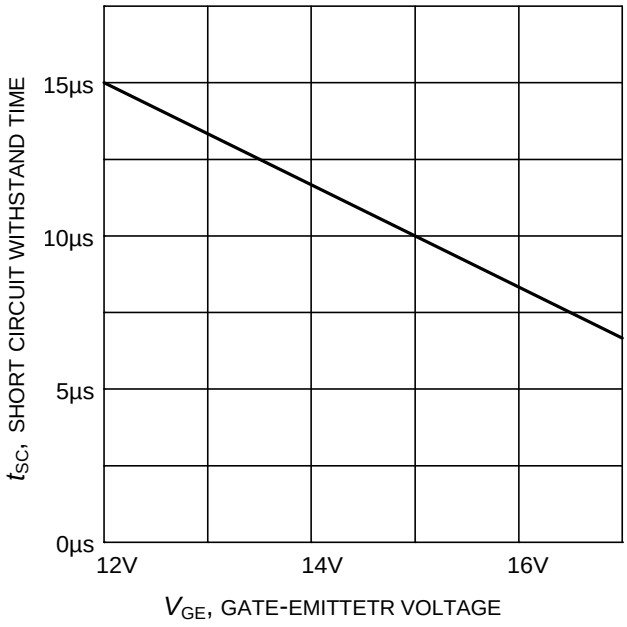


Figure 19. Short circuit withstand time as a function of gate-emitter voltage
(V_{CE}=600V, start at T_J=25°C)

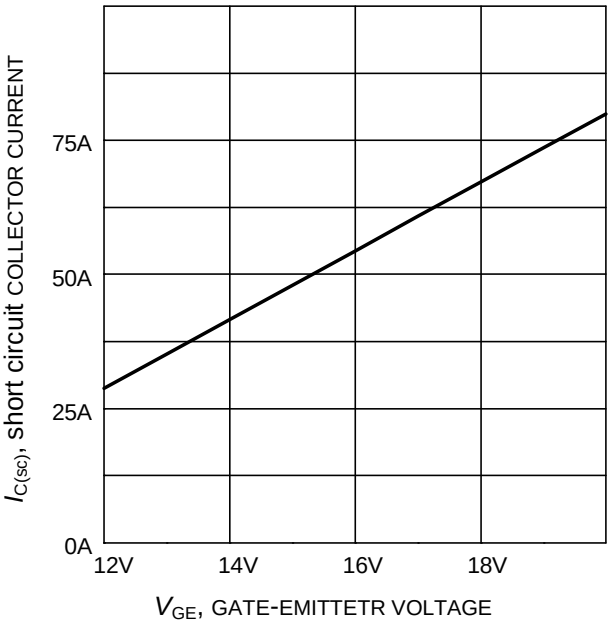


Figure 20. Typical short circuit collector current as a function of gate-emitter voltage
($V_{CE} \le$ 600V, $T_j \le$ 150°C)

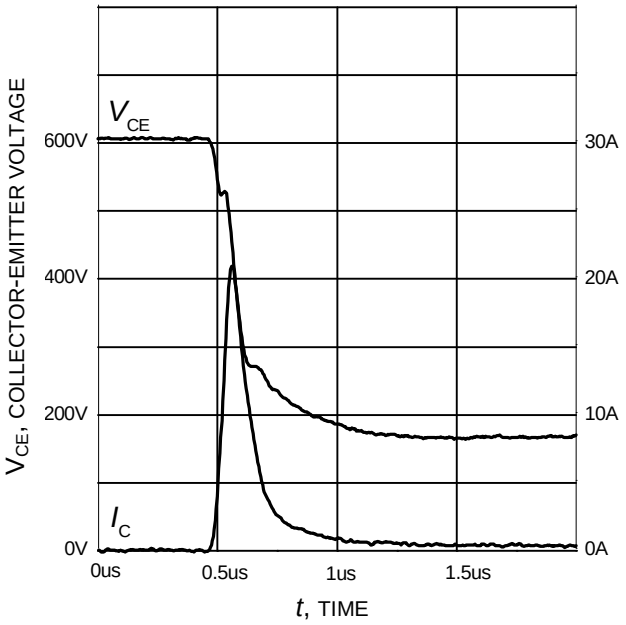


Figure 21. Typical turn on behavior
(V_{GE}=0/15V, R_G=81Ω, T_j = 150°C,
Dynamic test circuit in Figure E)

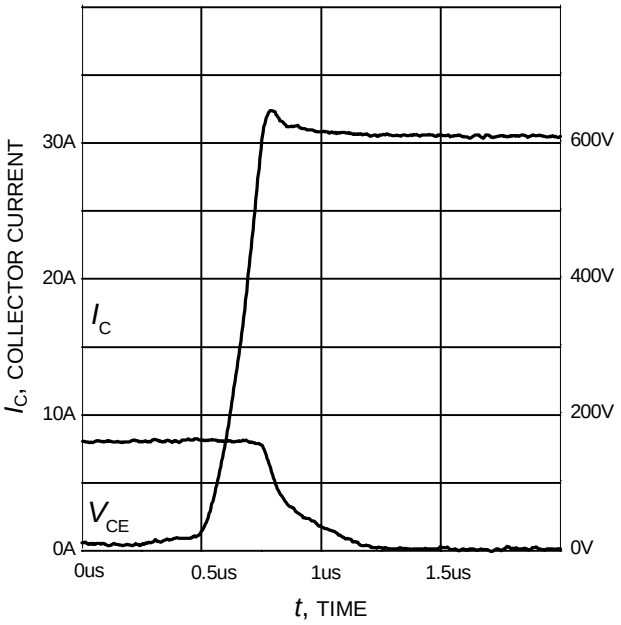


Figure 22. Typical turn off behavior
(V_{GE}=15/0V, R_G=81Ω, T_j = 150°C,
Dynamic test circuit in Figure E)

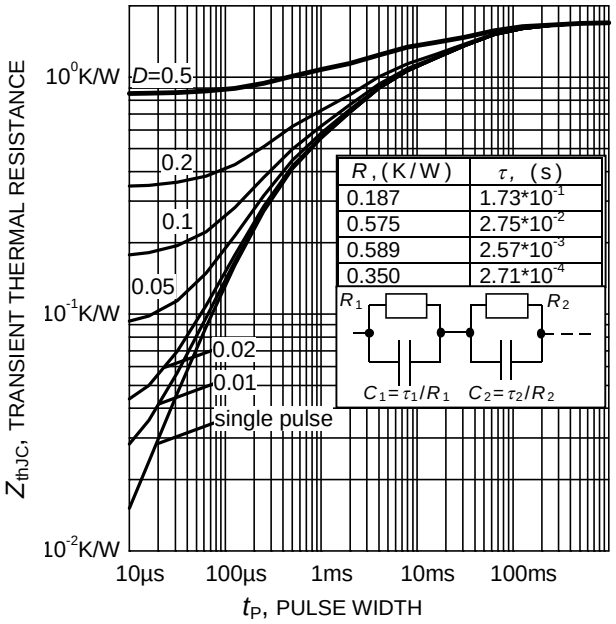


Figure 23. IGBT transient thermal resistance
($D = t_p / T$)

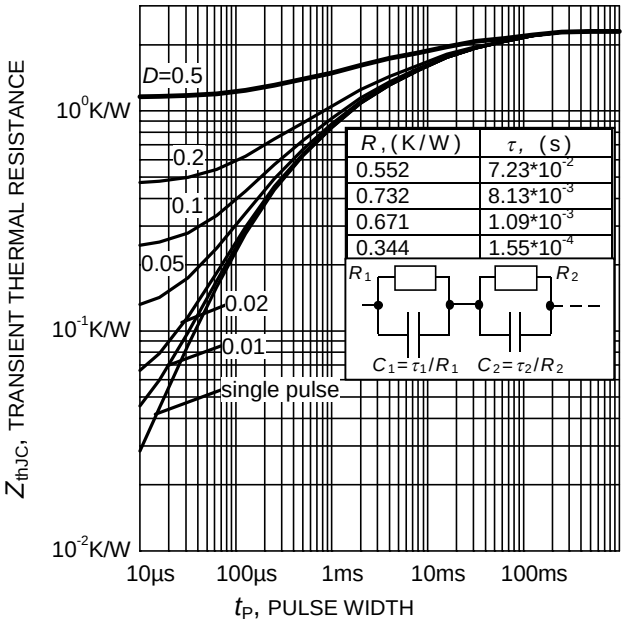


Figure 24. Diode transient thermal impedance as a function of pulse width
($D=t_P/T$)

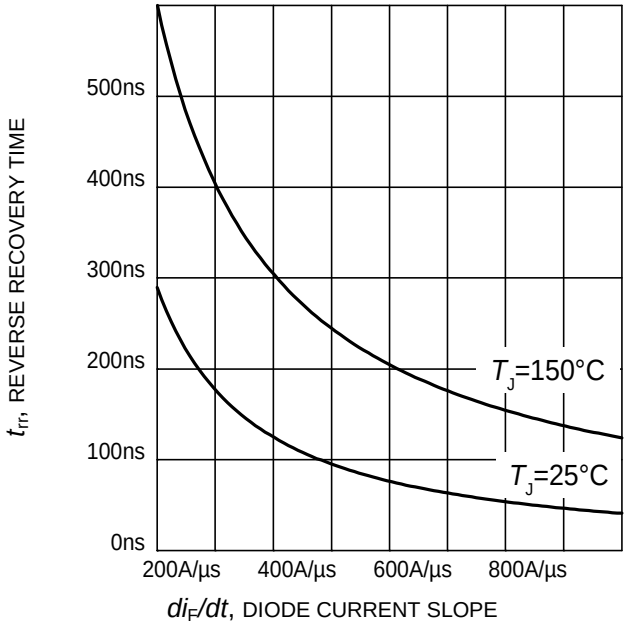


Figure 23. Typical reverse recovery time as a function of diode current slope
(V_R=600V, I_F=8A,
Dynamic test circuit in Figure E)

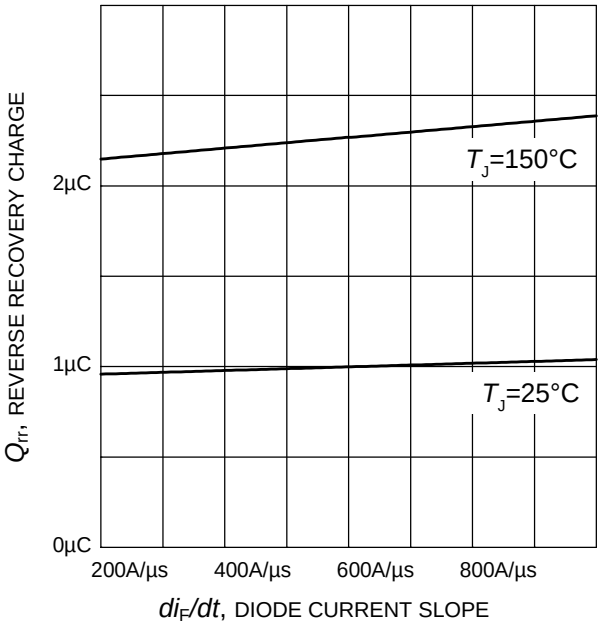


Figure 24. Typical reverse recovery charge as a function of diode current slope
(V_R=600V, I_F=8A,
Dynamic test circuit in Figure E)

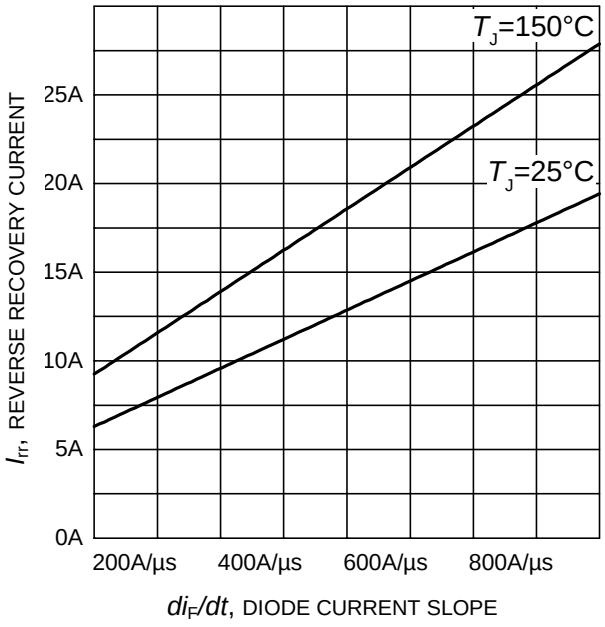


Figure 25. Typical reverse recovery current as a function of diode current slope
(V_R=600V, I_F=8A,
Dynamic test circuit in Figure E)

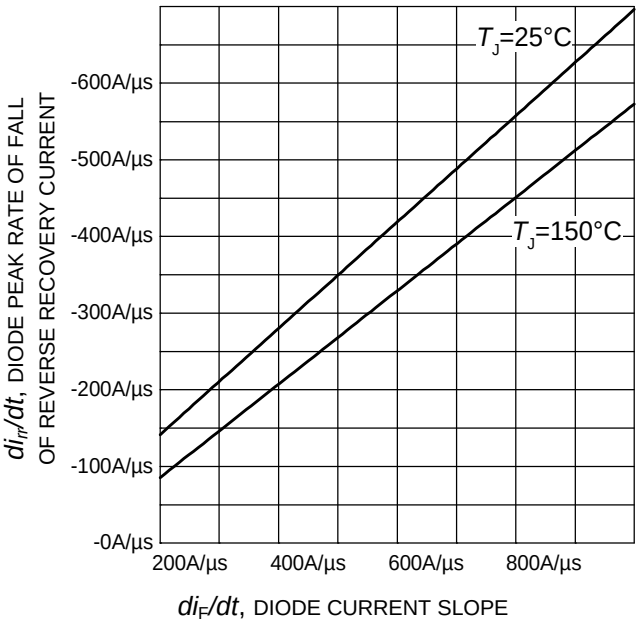


Figure 26. Typical diode peak rate of fall of reverse recovery current as a function of diode current slope
(V_R=600V, I_F=8A,
Dynamic test circuit in Figure E)

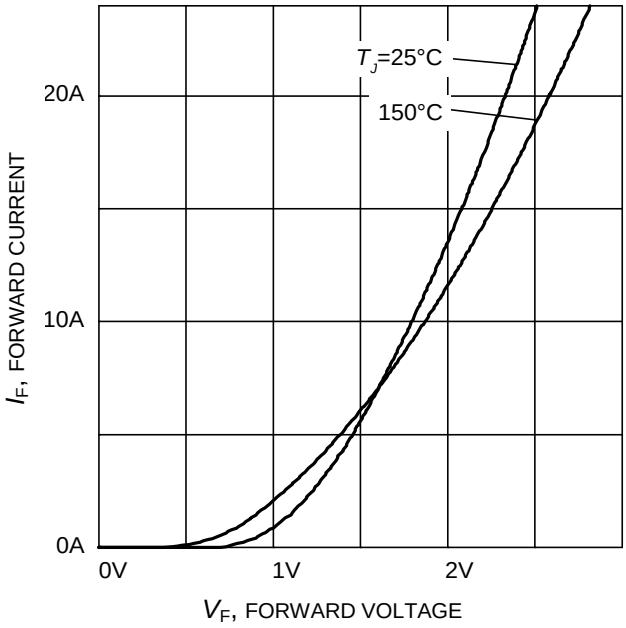


Figure 27. Typical diode forward current as a function of forward voltage

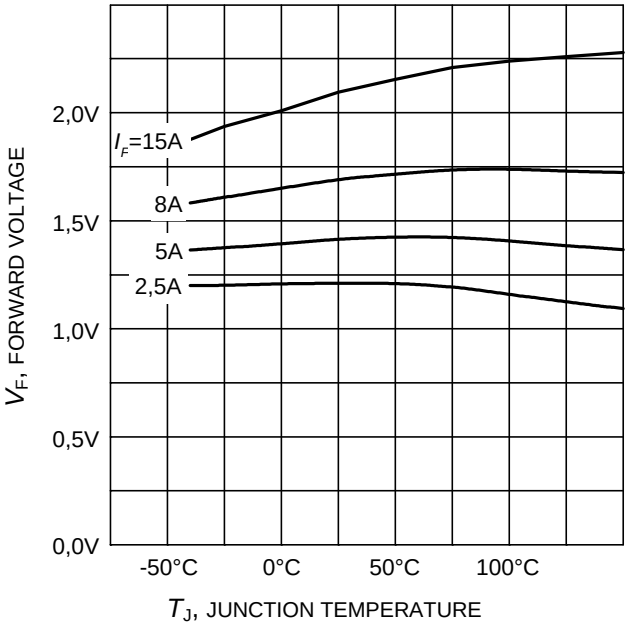
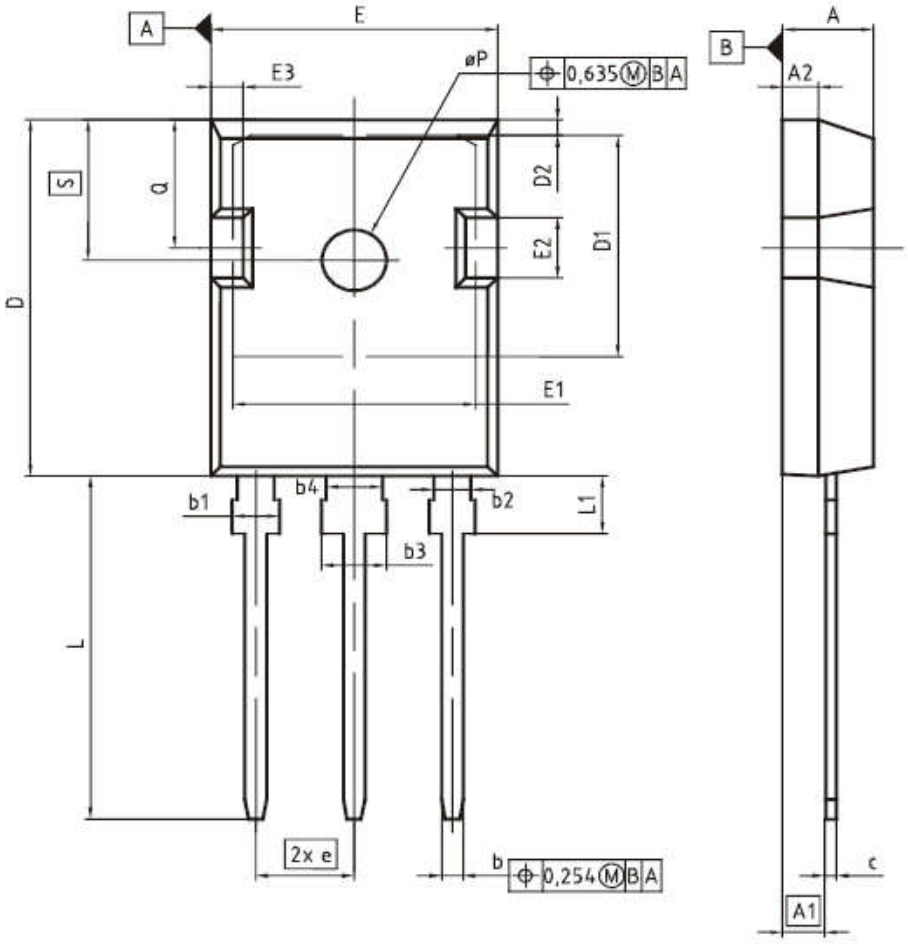


Figure 28. Typical diode forward voltage as a function of junction temperature

PG-TO247-3



DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	4,83	5,21	0,190	0,205
A1	2,27	2,54	0,089	0,100
A2	1,85	2,16	0,073	0,085
b	1,07	1,33	0,042	0,052
b1	1,90	2,41	0,075	0,095
b2	1,90	2,16	0,075	0,085
b3	2,87	3,38	0,113	0,133
b4	2,87	3,13	0,113	0,123
c	0,55	0,68	0,022	0,027
D	20,80	21,10	0,819	0,831
D1	16,25	17,65	0,640	0,695
D2	0,95	1,35	0,037	0,053
E	15,70	16,13	0,618	0,635
E1	13,10	14,15	0,516	0,557
E2	3,68	5,10	0,145	0,201
E3	1,00	2,60	0,039	0,102
e	5,44 (BSC)		0,214 (BSC)	
N	3		3	
L	19,80	20,32	0,780	0,800
L1	4,10	4,47	0,161	0,176
øP	3,50	3,70	0,138	0,146
Q	5,49	6,00	0,216	0,236
S	6,04	6,30	0,238	0,248

DOCUMENT NO.	Z8B00003327
SCALE	0 5 5 7,5mm
EUROPEAN PROJECTION	
ISSUE DATE	09-07-2010
REVISION	05

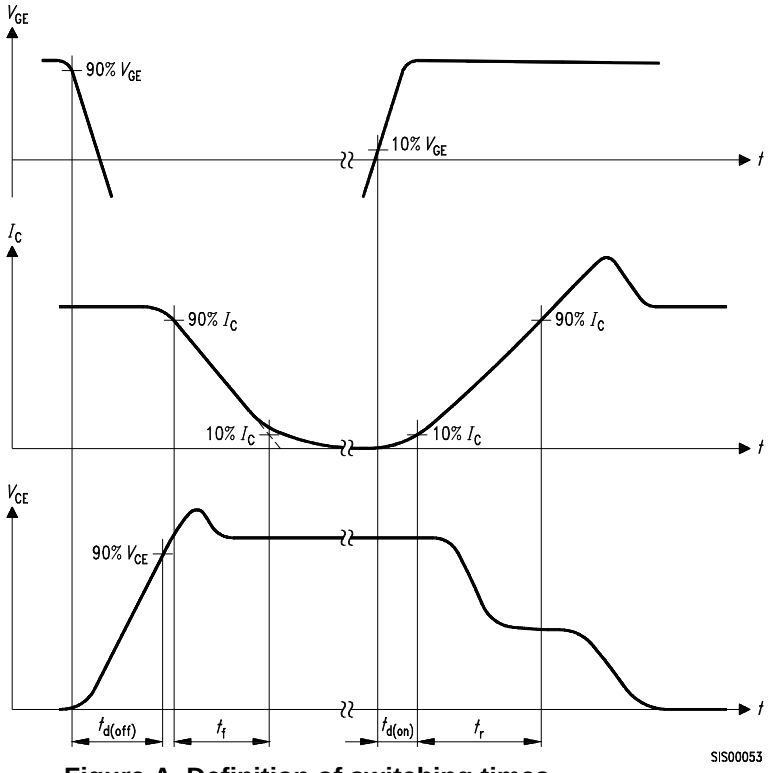


Figure A. Definition of switching times

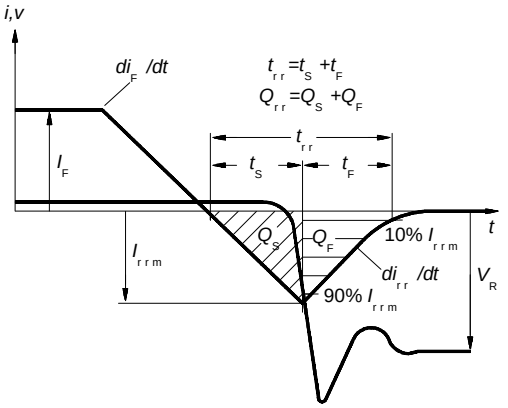


Figure C. Definition of diodes switching characteristics

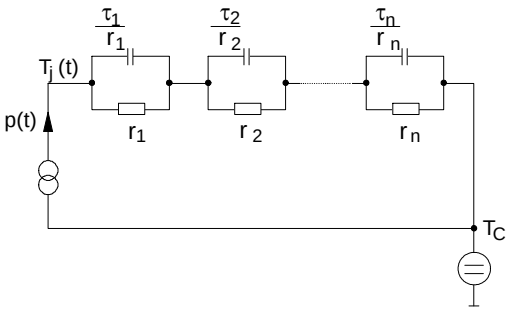


Figure D. Thermal equivalent circuit

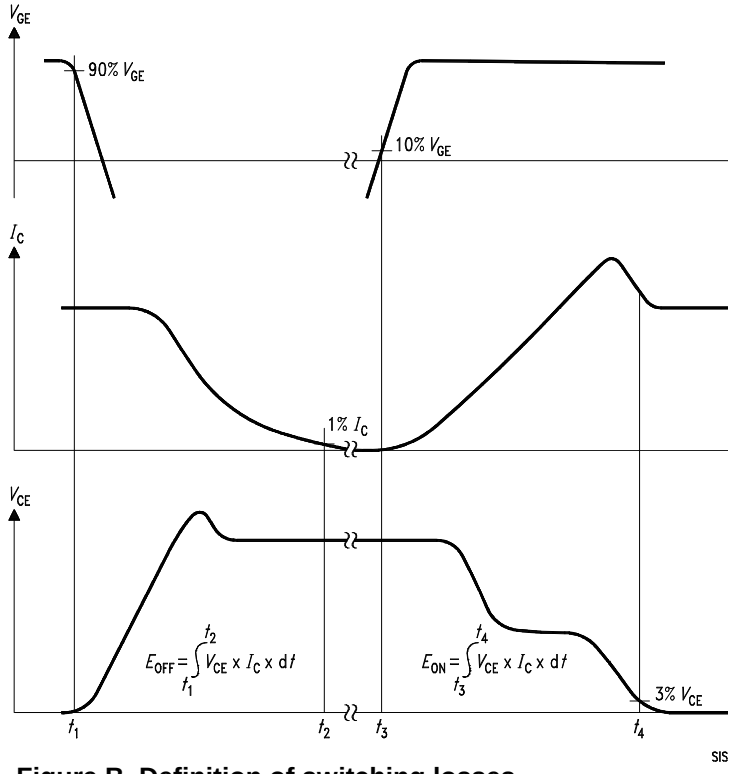


Figure B. Definition of switching losses

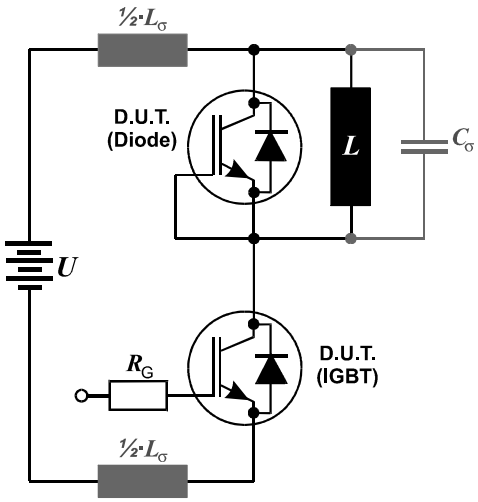


Figure E. Dynamic test circuit
Leakage inductance L_σ =180nH
and Stray capacity C_σ =39pF.

Published by
Infineon Technologies AG
81726 Munich, Germany
© 2013 Infineon Technologies AG
All Rights Reserved.

Legal Disclaimer

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation, warranties of non-infringement of intellectual property rights of any third party.

Information

For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office (**www.infineon.com**)**.**

Warnings

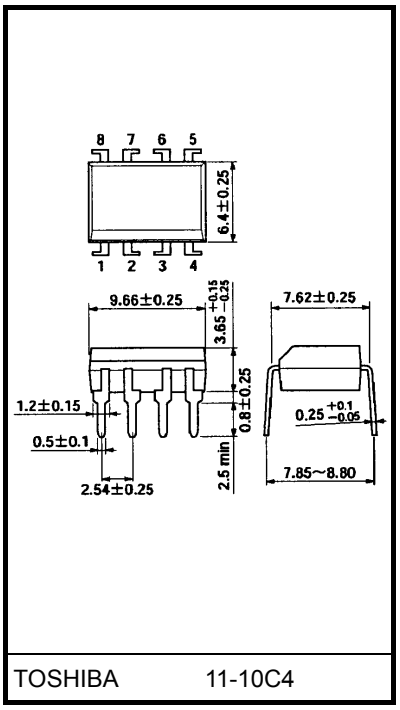
Due to technical requirements, components may contain dangerous substances. For information on the types in question, please contact the nearest Infineon Technologies Office.
The Infineon Technologies component described in this Data Sheet may be used in life-support devices or systems and/or automotive, aviation and aerospace applications or systems only with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support, automotive, aviation and aerospace device or system or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

TOSHIBA Photocoupler GaAℓAs IRED + Photo IC

TLP350

Industrial Inverter
Inverter for Air Conditioner
IGBT/Power MOSFET Gate Drive
IH (Induction Heating)

Unit: mm



TOSHIBA 11-10C4

Weight: 0.54 g (typ.)

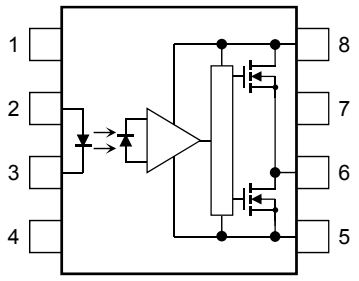
The TOSHIBA TLP350 consists of a GaAℓAs light-emitting diode and an integrated photodetector.
This unit is an 8-lead DIP package.
The TLP350 is suitable for gate driving IGBTs or power MOSFETs.

- Peak output current : I_O = ±2.5A (max)
- Guaranteed performance over temperature : -40 to 100°C
- Supply current : I_{CC} = 2 mA (max)
- Power supply voltage: V_{CC} = 15 to 30 V
- Threshold input current : I_{FLH} = 5 mA (max)
- Switching time (t_{pLH}/t_{pHL}) : 500 ns (max)
- Common mode transient immunity : 15 kV/μs
- Isolation voltage : 3750 Vrms
- UL Recognized : UL1577, File No.E67349
- Option(D4)
  VDE Approved : DIN EN 60747-5-2
  Maximum Operating Insulation Voltage : 890V_{PK}
  Highest Permissible Over Voltage : 6000V_{PK}
  (Note):When a EN 60747-5-2 approved type is needed,
  Please designate "Option(D4)"

Truth Table

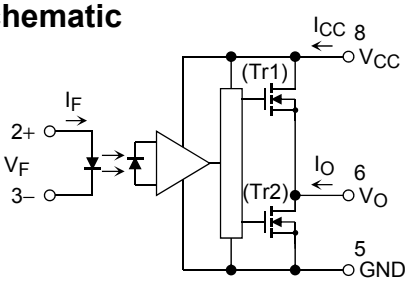
Input	LED	Tr1	Tr2	Output
H	ON	ON	OFF	H
L	OFF	OFF	ON	L

Pin Configuration (top view)



1: NC
2: Anode
3: Cathode
4: NC
5: GND
6: V_O (output)
7: NC
8: V_{CC}

Schematic



A 0.1 μF bypass capacitor must be connected between pins 8 and 5. (See Note 6)

Start of commercial production
2005/05

Absolute Maximum Ratings (Ta = 25°C)

Characteristic			Symbol	Rating	Unit
LED	Forward current		I_F	20	mA
	Forward current derating (Ta ≥ 85°C)		$\Delta I_F/\Delta Ta$	−0.54	mA/°C
	Peak transient forward current (Note 1)		I_{FPT}	1	A
	Reverse voltage		V_R	5	V
	Junction temperature		T_j	125	°C
Detector	"H" peak output current	Ta = −40 to 100°C (Note 2)	I_{OPH}	−2.5	A
	"L" peak output current		I_{OPL}	2.5	A
	Supply voltage	Ta < 95 °C	V_{CC}	35	V
	Supply voltage Derating	Ta ≥ 95 °C	$\Delta V_{CC} / \Delta Ta$	-1.0	V /°C
	Junction temperature		T_j	125	°C
Operating frequency (Note 3)			f	50	kHz
Storage temperature range			T_{stg}	−55 to 125	°C
Operating temperature range			T_{opr}	−40 to 100	°C
Lead soldering temperature (10 s) (Note 4)			T_{sol}	260	°C
Isolation voltage (AC, 1 minute, R.H. ≤ 60%) (Note 5)			BV_S	3750	Vrms

Note: Using continuously under heavy loads (e.g. the application of high temperature/current/voltage and the significant change in temperature, etc.) may cause this product to decrease in the reliability significantly even if the operating conditions (i.e. operating temperature/current/voltage, etc.) are within the absolute maximum ratings.
Please design the appropriate reliability upon reviewing the Toshiba Semiconductor Reliability Handbook ("Handling Precautions"/"Derating Concept and Methods") and individual reliability data (i.e. reliability test report and estimated failure rate, etc).

Note 1: Pulse width $P_W \leq 1$ μs, 300 pps

Note 2: Exponential waveform pulse width $P_W \leq 0.3$μs, f ≤ 15 kHz

Note 3: Exponential waveform $I_{OPH} \geq -2.0$A (≤ 0.3μs), $I_{OPL} \leq 2.0$A (≤ 0.3μs)

Note 4: At 2 mm or more from the lead root.

Note 5: This device is regarded as a two terminal device: pins 1, 2, 3 and 4 are shorted together, as are pins 5, 6, 7 and 8.

Note 6: A ceramic capacitor (0.1 μF) should be connected from pin 8 to pin 5 to stabilize the operation of the high gain linear amplifier. Failure to provide the bypass may impair the switching property.
The total lead length between capacitor and coupler should not exceed 1 cm.

Recommended Operating Conditions

Characteristic	Symbol	Min	Typ.	Max	Unit
Input current, ON (Note 7)	$I_{F\ (ON)}$	7.5	—	10	mA
Input voltage, OFF	$V_{F\ (OFF)}$	0	—	0.8	V
Supply voltage	V_{CC}	15	—	30	V
Peak output current	I_{OPH}/I_{OPL}	—	—	±2.0	A
Operating temperature	T_{opr}	−40	—	100	°C

Note: Recommended operating conditions are given as a design guideline to obtain expected performance of the device. Additionally, each item is an independent guideline respectively. In developing designs using this product, please confirm specified characteristics shown in this document.

Note 7: Input signal rise time (fall time) < 0.5 μs.

Note 8: If the rising slope of the supply voltage (V_{CC}) for the detector is steep, stable operation of the internal circuits cannot be guaranteed.
Be sure to set 3.0V/μs or less for a rising slope of the V_{CC}.

Electrical Characteristics (Ta = −40 to 100°C, unless otherwise specified)

Characteristic		Symbol	Test Circuit	Test Conditions		Min	Typ.*	Max	Unit
Forward voltage		V_F	—	I_F = 10 mA, Ta = 25°C		—	1.6	1.8	V
Temperature coefficient of forward voltage		$\Delta V_F/\Delta Ta$	—	I_F = 10 mA		—	−2.0	—	mV/°C
Input reverse current		I_R	—	V_R = 5 V, Ta = 25°C		—	—	10	μA
Input capacitance		C_T	—	V = 0 , f = 1 MHz, Ta = 25°C		—	45	250	pF
Output current (Note 9)	"H" Level	I_{OPH}	1	V_{CC} = 30 V, I_F = 5 mA V_{8-6} = 3.5 V		—	−1.6	-1.0	A
				V_{CC} = 15 V, I_F = 5 mA V_{8-6} = 7.0 V		—	—	-2.0	
	"L" Level	I_{OPL}	2	V_{CC} = 30 V, I_F = 0 mA V_{6-5} = 2.5V		1.0	1.6	—	
				V_{CC} = 15 V, I_F = 0 mA V_{6-5} = 7.0V		2.0	—	—	
Output voltage	"H" Level	V_{OH}	3	$V_{CC\ 1}$= +15 V $V_{EE\ 1}$= +15 V R_L = 200 Ω	I_F = 5 mA	11	13.7	—	V
	"L" Level	V_{OL}	4		V_F = 0.8 V	—	-14.9	-12.5	
Supply current	"H" Level	I_{CCH}	5	V_{CC} = 30 V V_O open	I_F = 10 mA	—	1.3	2.0	mA
	"L" Level	I_{CCL}	6		I_F = 0 mA	—	1.3	2.0	
Threshold input current	L → H	I_{FLH}	—	V_{CC} = 15V , V_O > 1V , I_O = 0mA		—	1.8	5	mA
Threshold input voltage	H → L	V_{FHL}	—	V_{CC} = 15V , V_O < 1V , I_O = 0mA		0.8	—	—	V
Supply voltage		V_{CC}	—	—		15	—	30	V
UVLO threshold		V_{UVLO+}	—	V_O > 2.5 V , I_F = 5 mA		11.0	12.5	13.5	V
		V_{UVLO-}	—			9.5	11.0	12.0	V
UVLO hysteresis		$UVLO_{HYS}$	—	—		—	1.5	—	V

*: All typical values are at Ta = 25°C

Note 9: Duration of I_O: ≤ 50 μs (1 PULSE)

Note 10: This product is more sensitive to static electricity (ESD) than the conventional product because of its minimal power consumption design.
General static electricity precautions are necessary for handling this component.

Isolation Characteristics (Ta = 25°C)

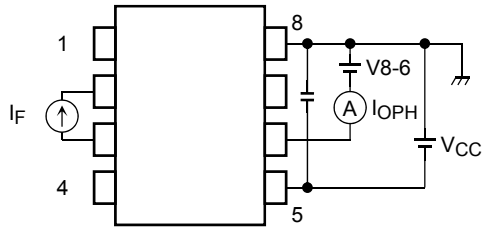
Characteristic	Symbol	Test Conditions		Min	Typ.	Max	Unit
Capacitance input to output	C_S	V = 0,f = 1MHz	(Note5)	—	1.0	—	pF
Isolation resistance	R_S	V_S = 500 V, R.H. ≤ 60%	(Note5)	1×10^{12}	10^{14}	—	Ω
Isolation voltage	BV_S	AC,1 minute		3750	—	—	V_{rms}
		AC,1 second, in oil		—	10000	—	
		DC,1 minute, in oil		—	10000	—	Vdc

Switching Characteristics (Ta = –40 to 100°C, unless otherwise specified)

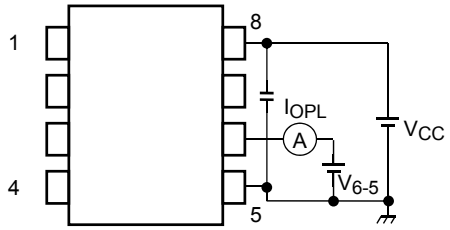
Characteristic		Symbol	Test Circuit	Test Conditions		Min	Typ.*	Max	Unit
Propagation delay time	L → H	t_{pLH}	7	V_{CC} = 30 V R_g = 20 Ω C_g = 10 nF	I_F = 0 → 5 mA	50	260	500	ns
	H → L	t_{pHL}			I_F = 5 → 0 mA	50	260	500	
Switching Time Dispersion between ON and OFF		$\lvert t_{pHL}-t_{pLH}\rvert$		V_{CC} = 30 V R_g = 20 Ω C_g = 10 nF		—	—	350	
Output rise time (10-90%)		t_r		V_{CC} = 30 V R_g = 20 Ω C_g = 10 nF	I_F = 0 → 5 mA	—	15	—	
Output fall time (90-10%)		t_f			I_F = 5 → 0 mA	—	8	—	
Common mode transient immunity at high level output		CM_H	8	V_{CM} = 1000 Vp-p Ta = 25°C V_{CC} = 30 V	I_F = 5 mA $V_{O\ (min)}$ =26V	–15000	—	—	V/μs
Common mode transient immunity at low level output		CM_L			I_F = 0 mA $V_{O\ (max)}$ =1V	15000	—	—	

*: All typical values are at Ta = 25°C

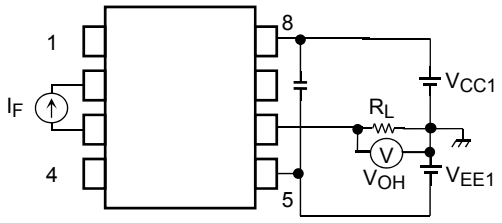
Test Circuit 1: I_{OPH}



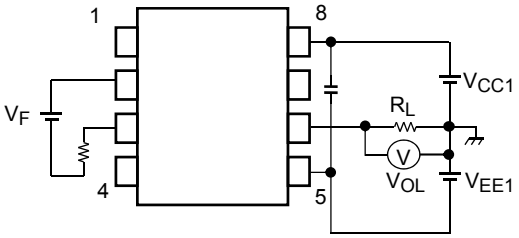
Test Circuit 2: I_{OPL}



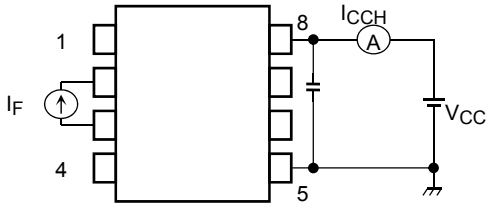
Test Circuit 3: V_{OH}



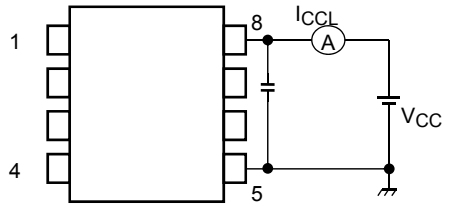
Test Circuit 4: V_{OL}



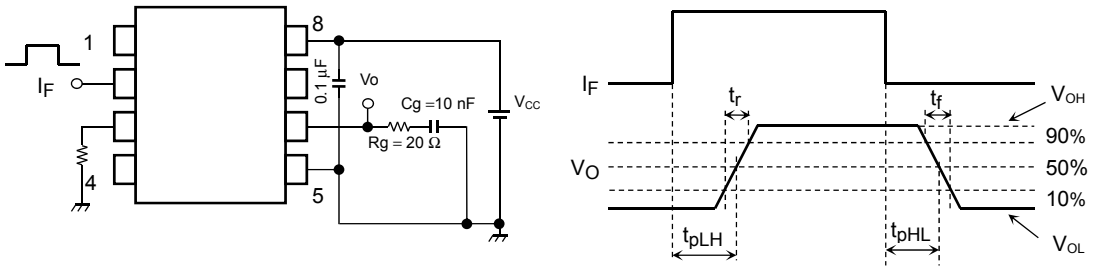
Test Circuit 5: I_{CCH}



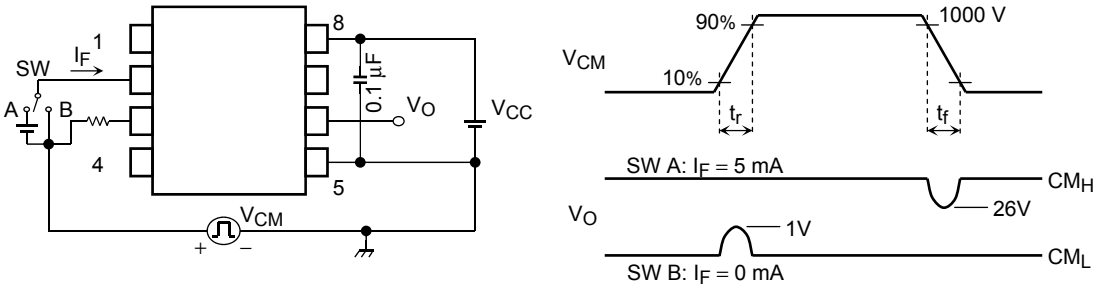
Test Circuit 6: I_{CCL}



Test Circuit 7: t_{pLH}, t_{pHL}, t_r, t_f



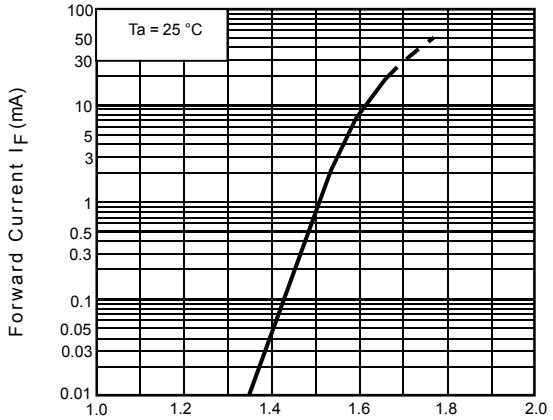
Test Circuit 8: CM_H, CM_L



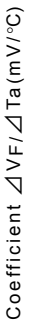
$$CM_L = \frac{800(V)}{t_r(\mu s)} \qquad CM_H = \frac{800(V)}{t_f(\mu s)}$$

CM_L (CM_H) is the maximum rate of rise (fall) of the common mode voltage that can be sustained with the output voltage in the low (high) state.

$I_F - V_F$

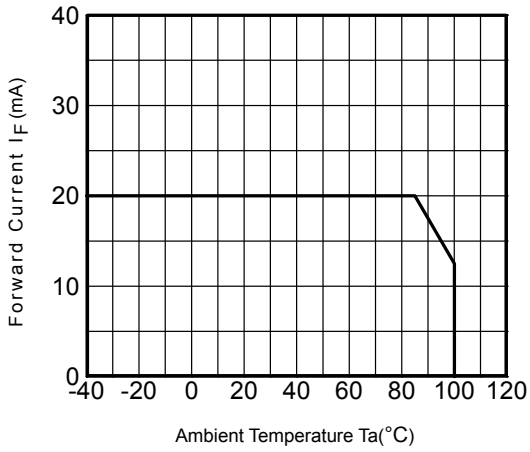


$\Delta V_F / \Delta Ta - I_F$

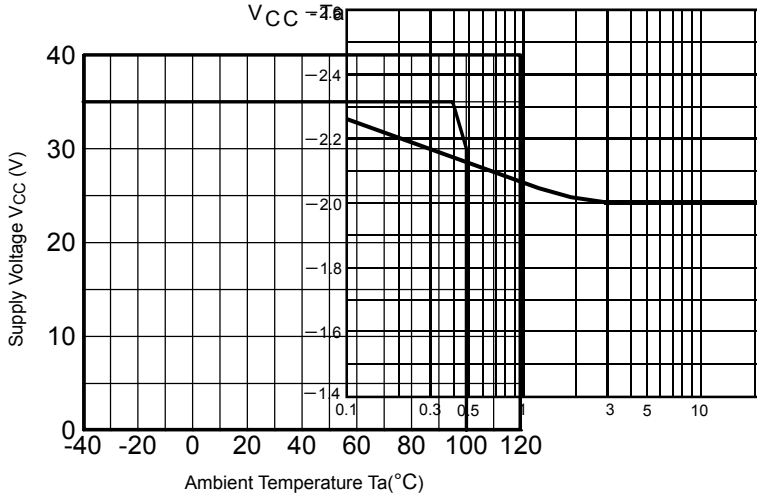


Forward Current I_F (mA)

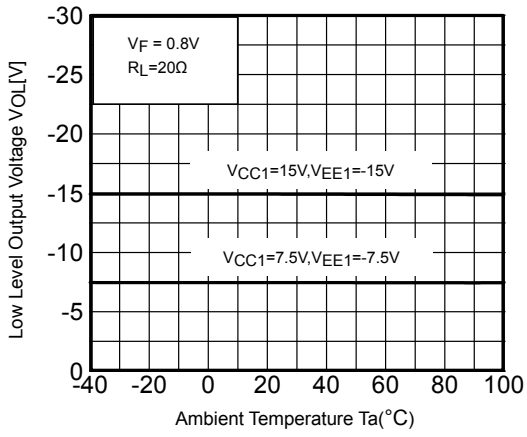
I_F -Ta



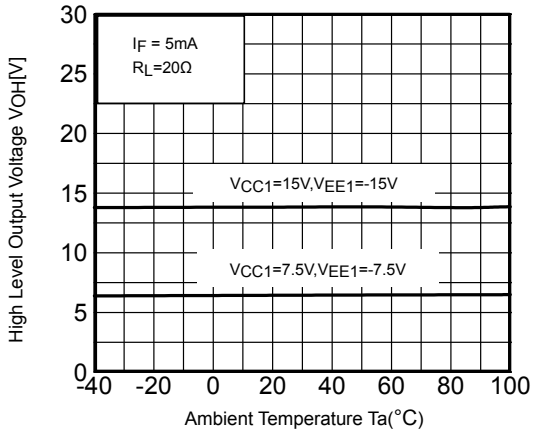
V_{CC} -Ta



V_{OL} -Ta

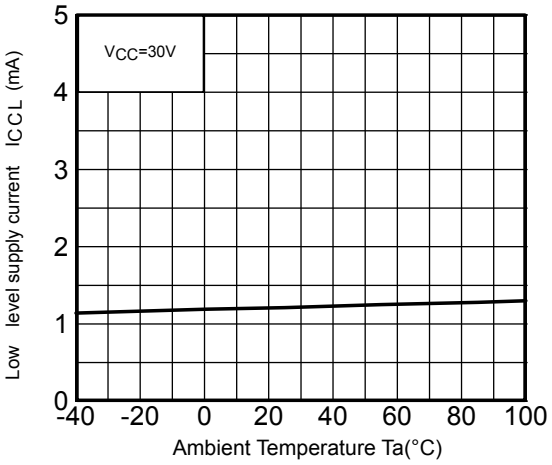


V_{OH} -Ta

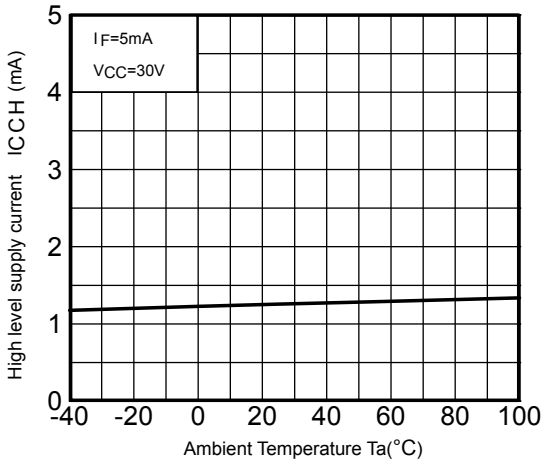


*: The above graphs show typical characteristics.

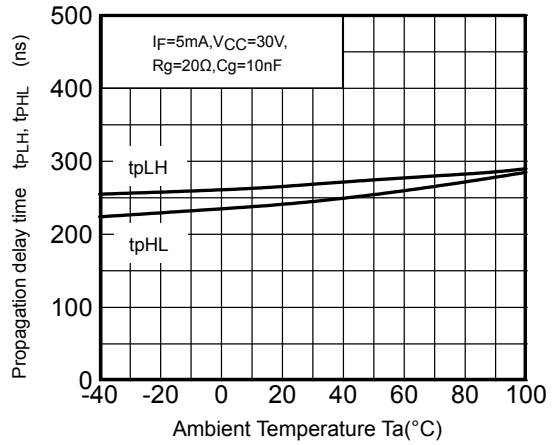
ICCL-Ta



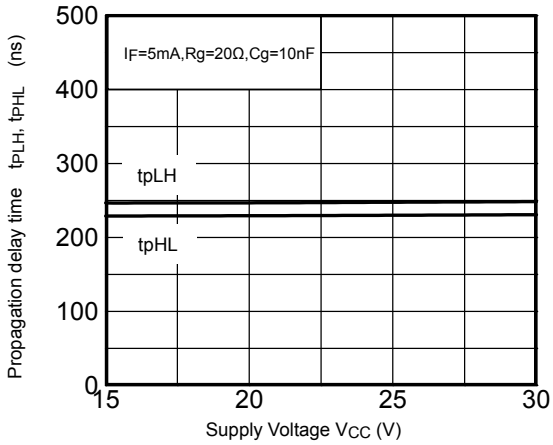
ICCH-Ta



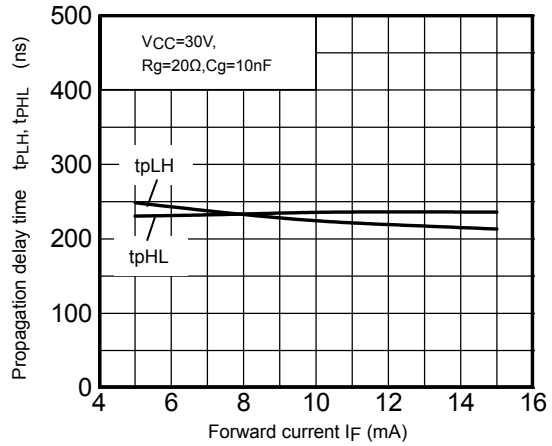
tpHL,tpLH-Ta



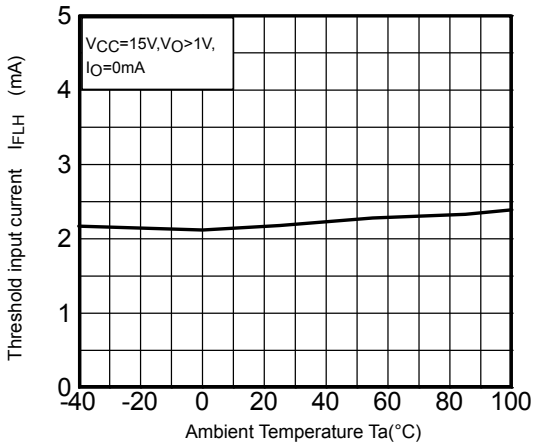
tpHL,tpLH-VCC



tpHL,tpLH-IF

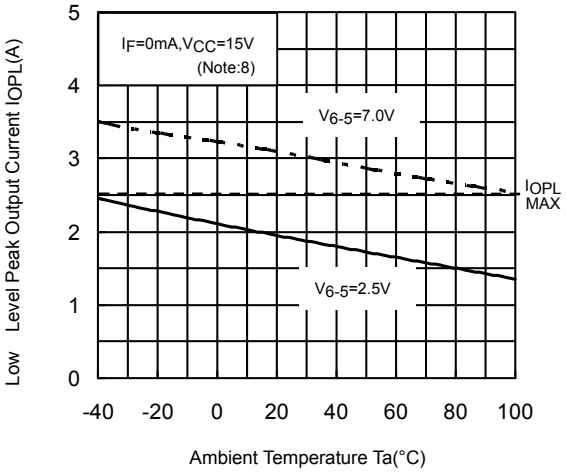


IFLH-Ta

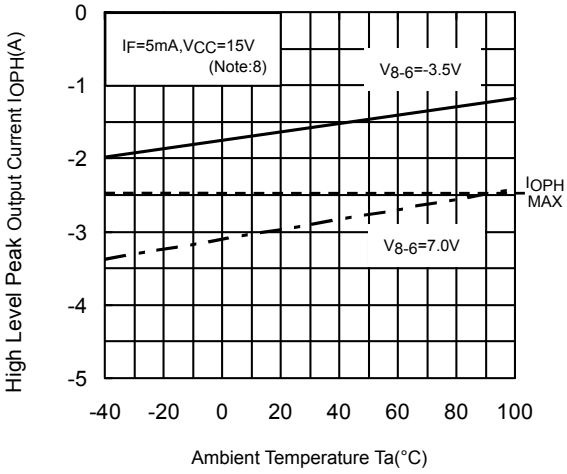


*: The above graphs show typical characteristics.

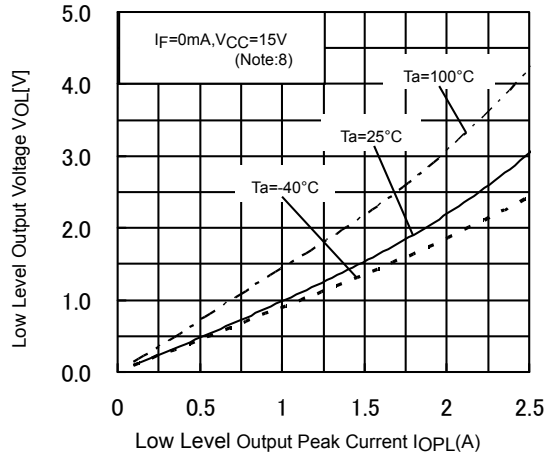
I_{OPL}-Ta



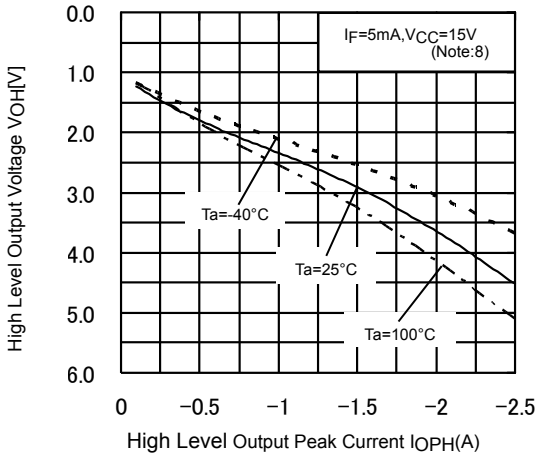
I_{OPH}-Ta



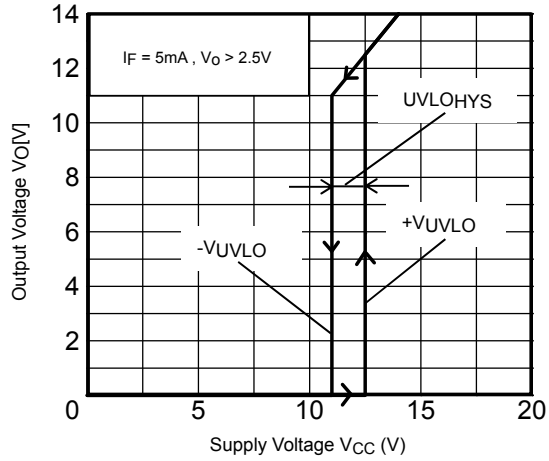
I_{OPL}-V_{OL}



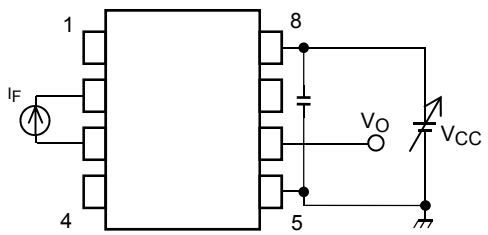
I_{OPH}-V_{OH}



V_{CC}-V_O(V_{UVLO})**



**Test Circuit : V_{CC}-V_O(V_{UVLO})



*: The above graphs show typical characteristics.

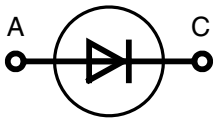
RESTRICTIONS ON PRODUCT USE

- Toshiba Corporation, and its subsidiaries and affiliates (collectively "TOSHIBA"), reserve the right to make changes to the information in this document, and related hardware, software and systems (collectively "Product") without notice.
- This document and any information herein may not be reproduced without prior written permission from TOSHIBA. Even with TOSHIBA's written permission, reproduction is permissible only if reproduction is without alteration/omission.
- Though TOSHIBA works continually to improve Product's quality and reliability, Product can malfunction or fail. Customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of Product could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Before customers use the Product, create designs including the Product, or incorporate the Product into their own applications, customers must also refer to and comply with (a) the latest versions of all relevant TOSHIBA information, including without limitation, this document, the specifications, the data sheets and application notes for Product and the precautions and conditions set forth in the "TOSHIBA Semiconductor Reliability Handbook" and (b) the instructions for the application with which the Product will be used with or for. Customers are solely responsible for all aspects of their own product design or applications, including but not limited to (a) determining the appropriateness of the use of this Product in such design or applications; (b) evaluating and determining the applicability of any information contained in this document, or in charts, diagrams, programs, algorithms, sample application circuits, or any other referenced documents; and (c) validating all operating parameters for such designs and applications. **TOSHIBA ASSUMES NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.**
- **PRODUCT IS NEITHER INTENDED NOR WARRANTED FOR USE IN EQUIPMENTS OR SYSTEMS THAT REQUIRE EXTRAORDINARILY HIGH LEVELS OF QUALITY AND/OR RELIABILITY, AND/OR A MALFUNCTION OR FAILURE OF WHICH MAY CAUSE LOSS OF HUMAN LIFE, BODILY INJURY, SERIOUS PROPERTY DAMAGE AND/OR SERIOUS PUBLIC IMPACT ("UNINTENDED USE").** Except for specific applications as expressly stated in this document, Unintended Use includes, without limitation, equipment used in nuclear facilities, equipment used in the aerospace industry, medical equipment, equipment used for automobiles, trains, ships and other transportation, traffic signaling equipment, equipment used to control combustions or explosions, safety devices, elevators and escalators, devices related to electric power, and equipment used in finance-related fields. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative.
- Do not disassemble, analyze, reverse-engineer, alter, modify, translate or copy Product, whether in whole or in part.
- Product shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
- The information contained herein is presented only as guidance for Product use. No responsibility is assumed by TOSHIBA for any infringement of patents or any other intellectual property rights of third parties that may result from the use of Product. No license to any intellectual property right is granted by this document, whether express or implied, by estoppel or otherwise.
- **ABSENT A WRITTEN SIGNED AGREEMENT, EXCEPT AS PROVIDED IN THE RELEVANT TERMS AND CONDITIONS OF SALE FOR PRODUCT, AND TO THE MAXIMUM EXTENT ALLOWABLE BY LAW, TOSHIBA (1) ASSUMES NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (2) DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO SALE, USE OF PRODUCT, OR INFORMATION, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.**
- GaAs (Gallium Arsenide) is used in Product. GaAs is harmful to humans if consumed or absorbed, whether in the form of dust or vapor. Handle with care and do not break, cut, crush, grind, dissolve chemically or otherwise expose GaAs in Product.
- Do not use or otherwise make available Product or related software or technology for any military purposes, including without limitation, for the design, development, use, stockpiling or manufacturing of nuclear, chemical, or biological weapons or missile technology products (mass destruction weapons). Product and related software and technology may be controlled under the applicable export laws and regulations including, without limitation, the Japanese Foreign Exchange and Foreign Trade Law and the U.S. Export Administration Regulations. Export and re-export of Product or related software or technology are strictly prohibited except in compliance with all applicable export laws and regulations.
- Please contact your TOSHIBA sales representative for details as to environmental matters such as the RoHS compatibility of Product. Please use Product in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. **TOSHIBA ASSUMES NO LIABILITY FOR DAMAGES OR LOSSES OCCURRING AS A RESULT OF NONCOMPLIANCE WITH APPLICABLE LAWS AND REGULATIONS.**

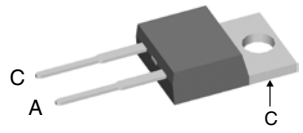
Fast Recovery Epitaxial Diode (FRED)

I_{FAV} = 12 A
V_{RRM} = 1000 V
t_{rr} = 50 ns

V_{RSM} V	V_{RRM} V	Type
1000	1000	DSEI 12-10A



TO-220 AC



A = Anode, C = Cathode

Symbol	Conditions		Maximum Ratings	
I_{FRMS}	$T_{VJ} = T_{VJM}$		25	A
I_{FAVM} ①	T_C = 100°C; rectangular, d = 0.5		12	A
I_{FRM}	t_p < 10 µs; rep. rating, pulse width limited by T_{VJM}		150	A
I_{FSM}	T_{VJ} = 45°C;	t = 10 ms (50 Hz), sine	75	A
		t = 8.3 ms (60 Hz), sine	80	
	T_{VJ} = 150°C;	t = 10 ms (50 Hz), sine	65	A
		t = 8.3 ms (60 Hz), sine	70	
I^2t	T_{VJ} = 45°C;	t = 10 ms (50 Hz), sine	28	A^2s
		t = 8.3 ms (60 Hz), sine	27	
	T_{VJ} = 150°C;	t = 10 ms (50 Hz), sine	21	A^2s
		t = 8.3 ms (60 Hz), sine	20	
T_{VJ}			-40...+150	°C
T_{VJM}			150	°C
T_{stg}			-40...+150	°C
P_{tot}	T_C = 25°C		78	W
M_d	mounting torque		0.4...0.6	Nm
Weight	typical		2	g

Symbol	Conditions		Characteristic Values typ.	max.	
I_R	$V_R = V_{RRM}$	T_{VJ} = 25°C		250	µA
	$V_R = 0.8 \cdot V_{RRM}$	T_{VJ} = 25°C		150	µA
	$V_R = 0.8 \cdot V_{RRM}$	T_{VJ} = 125°C		4	mA
V_F	I_F = 12 A	T_{VJ} = 150°C		2.1	V
		T_{VJ} = 25°C		2.7	V
V_{T0}	For power-loss calculations only			1.67	V
r_T	$T_{VJ} = T_{VJM}$			33.6	mΩ
R_{thJC}				1.6	K/W
R_{thCH}			0.5		K/W
R_{thJA}				60	K/W
t_{rr}	I_F = 1 A; -di/dt = 50 A/µs; V_R = 30 V; T_{VJ} = 25°C		50	60	ns
I_{RM}	V_R = 540 V; I_F = 12 A; $-di_F/dt$ = 100 A/µs L ≤ 0.05 µH; T_{VJ} = 100°C		6.5	7.2	A

① I_{FAVM} rating includes reverse blocking losses at T_{VJM}. $V_R = 0.8 \cdot V_{RRM}$, duty cycle d = 0.5
Data according to IEC 60747

Features

- International standard package JEDEC TO-220 AC
- Planar passivated chips
- Very short recovery time
- Extremely low switching losses
- Low I_{RM}-values
- Soft recovery behaviour
- Epoxy meets UL 94V-0

Applications

- Antiparallel diode for high frequency switching devices
- Anti saturation diode
- Snubber diode
- Free wheeling diode in converters and motor control circuits
- Rectifiers in switch mode power supplies (SMPS)
- Inductive heating and melting
- Uninterruptible power supplies (UPS)
- Ultrasonic cleaners and welders

Advantages

- High reliability circuit operation
- Low voltage peaks for reduced protection circuits
- Low noise switching
- Low losses
- Operating at lower temperature or space saving by reduced cooling

IXYS reserves the right to change limits, test conditions and dimensions.

20071004

© 2004 IXYS All rights reserved

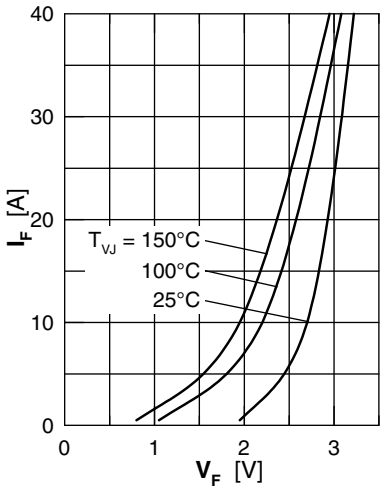


Fig. 1 Forward current versus voltage drop

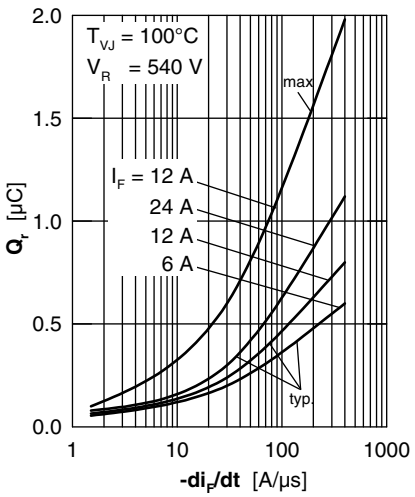


Fig. 2 Recovery charge versus $-di_F/dt$

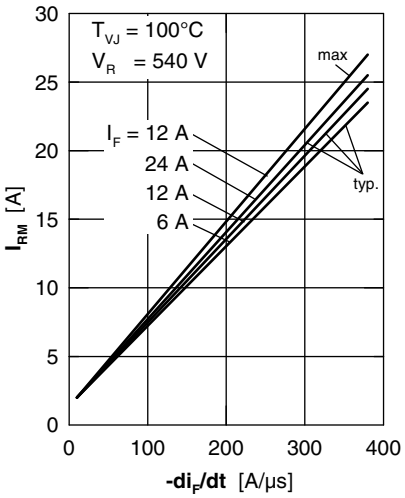


Fig. 3 Peak reverse current versus $-di_F/dt$

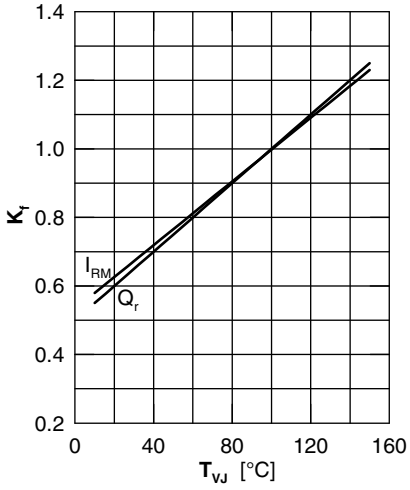


Fig. 4 Dynamic parameters versus junction temperature

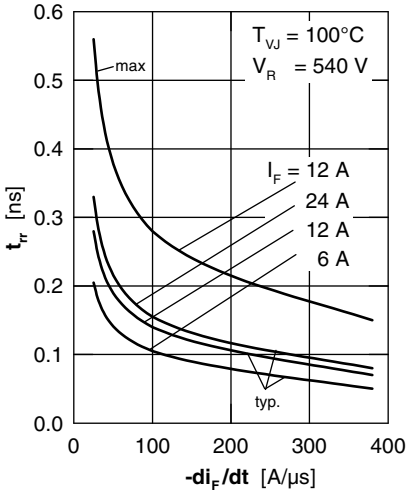


Fig. 5 Recovery time versus $-di_F/dt$

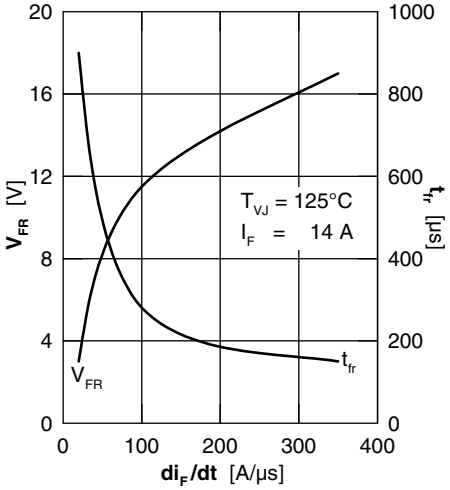


Fig. 6 Peak forward voltage versus di_F/dt

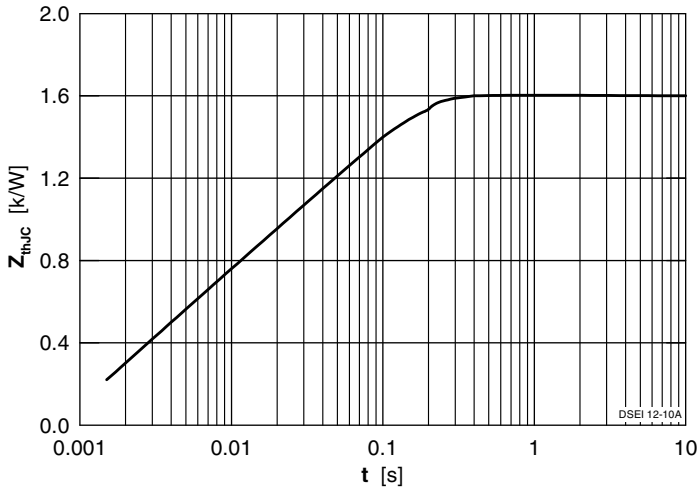


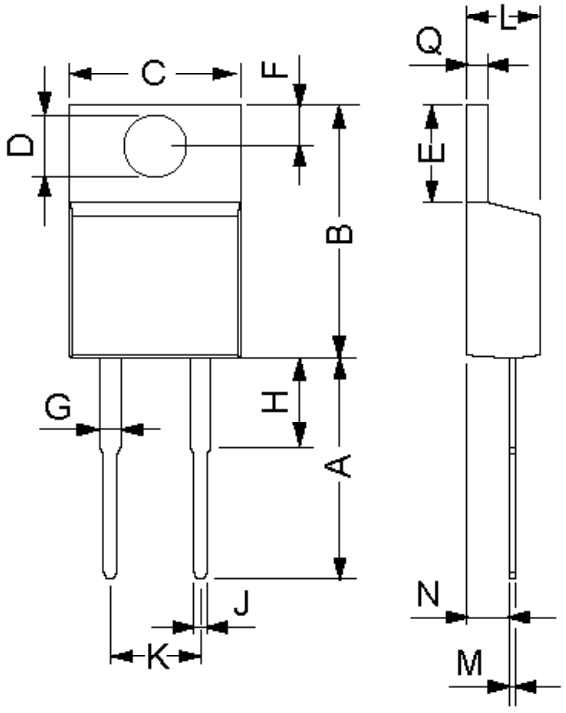
Fig. 7 Transient thermal resistance junction to case

20071004

IXYS reserves the right to change limits, test conditions and dimensions.

© 2004 IXYS All rights reserved

Dimensions TO-220 AC



Dim.	Millimeter		Inches	
	Min.	Max.	Min.	Max.
A	12.7	14.73	0.5	0.58
B	14.23	16.51	0.56	0.65
C	9.66	10.66	0.38	0.42
D	3.54	4.08	0.139	0.161
E	5.85	6.85	2.3	0.42
F	2.54	3.42	0.1	0.135
G	1.15	1.77	0.045	0.07
H	-	6.35	-	0.25
J	0.64	0.89	0.025	0.035
K	4.83	5.33	0.19	0.21
L	3.56	4.82	0.14	0.19
M	0.51	0.76	0.02	0.03
N	2.04	2.49	0.08	0.115
Q	0.64	1.39	0.025	0.055

20071004

IXYS reserves the right to change limits, test conditions and dimensions.

© 2004 IXYS All rights reserved

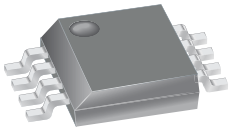
Fully Integrated, Hall Effect-Based Linear Current Sensor with 2.1 kVRMS Voltage Isolation and a Low-Resistance Current Conductor

Features and Benefits

- Low-noise analog signal path
- Device bandwidth is set via the new FILTER pin
- 5 µs output rise time in response to step input current
- 80 kHz bandwidth
- Total output error 1.5% at T_A = 25°C
- Small footprint, low-profile SOIC8 package
- 1.2 mΩ internal conductor resistance
- 2.1 kV_{RMS} minimum isolation voltage from pins 1-4 to pins 5-8
- 5.0 V, single supply operation
- 66 to 185 mV/A output sensitivity
- Output voltage proportional to AC or DC currents
- Factory-trimmed for accuracy
- Extremely stable output offset voltage
- Nearly zero magnetic hysteresis
- Ratiometric output from supply voltage

TÜV America
Certificate Number:
U8V 06 05 54214 010

Package: 8 Lead SOIC (suffix LC)



Approximate Scale 1:1

Description

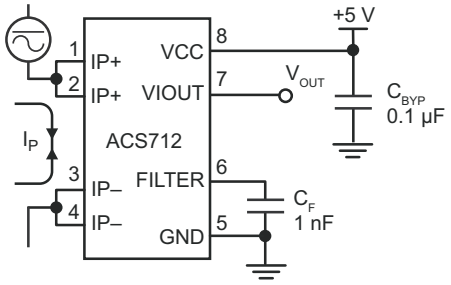
The Allegro® ACS712 provides economical and precise solutions for AC or DC current sensing in industrial, commercial, and communications systems. The device package allows for easy implementation by the customer. Typical applications include motor control, load detection and management, switched-mode power supplies, and overcurrent fault protection.

The device consists of a precise, low-offset, linear Hall sensor circuit with a copper conduction path located near the surface of the die. Applied current flowing through this copper conduction path generates a magnetic field which is sensed by the integrated Hall IC and converted into a proportional voltage. Device accuracy is optimized through the close proximity of the magnetic signal to the Hall transducer. A precise, proportional voltage is provided by the low-offset, chopper-stabilized BiCMOS Hall IC, which is programmed for accuracy after packaging.

The output of the device has a positive slope (>$V_{IOUT(Q)}$) when an increasing current flows through the primary copper conduction path (from pins 1 and 2, to pins 3 and 4), which is the path used for current sensing. The internal resistance of this conductive path is 1.2 mΩ typical, providing low power

Continued on the next page…

Typical Application



Application 1. The ACS712 outputs an analog signal, V_{OUT}. that varies linearly with the uni- or bi-directional AC or DC primary sensed current, I_P, within the range specified. C_F is recommended for noise management, with values that depend on the application.

Description (continued)

loss. The thickness of the copper conductor allows survival of the device at up to 5× overcurrent conditions. The terminals of the conductive path are electrically isolated from the sensor leads (pins 5 through 8). This allows the ACS712 current sensor to be used in applications requiring electrical isolation without the use of opto-isolators or other costly isolation techniques.

The ACS712 is provided in a small, surface mount SOIC8 package. The leadframe is plated with 100% matte tin, which is compatible with standard lead (Pb) free printed circuit board assembly processes. Internally, the device is Pb-free, except for flip-chip high-temperature Pb-based solder balls, currently exempt from RoHS. The device is fully calibrated prior to shipment from the factory.

Selection Guide

Part Number	Packing*	T_A (°C)	Optimized Range, I_P (A)	Sensitivity, Sens (Typ) (mV/A)
ACS712ELCTR-05B-T	Tape and reel, 3000 pieces/reel	–40 to 85	±5	185
ACS712ELCTR-20A-T	Tape and reel, 3000 pieces/reel	–40 to 85	±20	100
ACS712ELCTR-30A-T	Tape and reel, 3000 pieces/reel	–40 to 85	±30	66

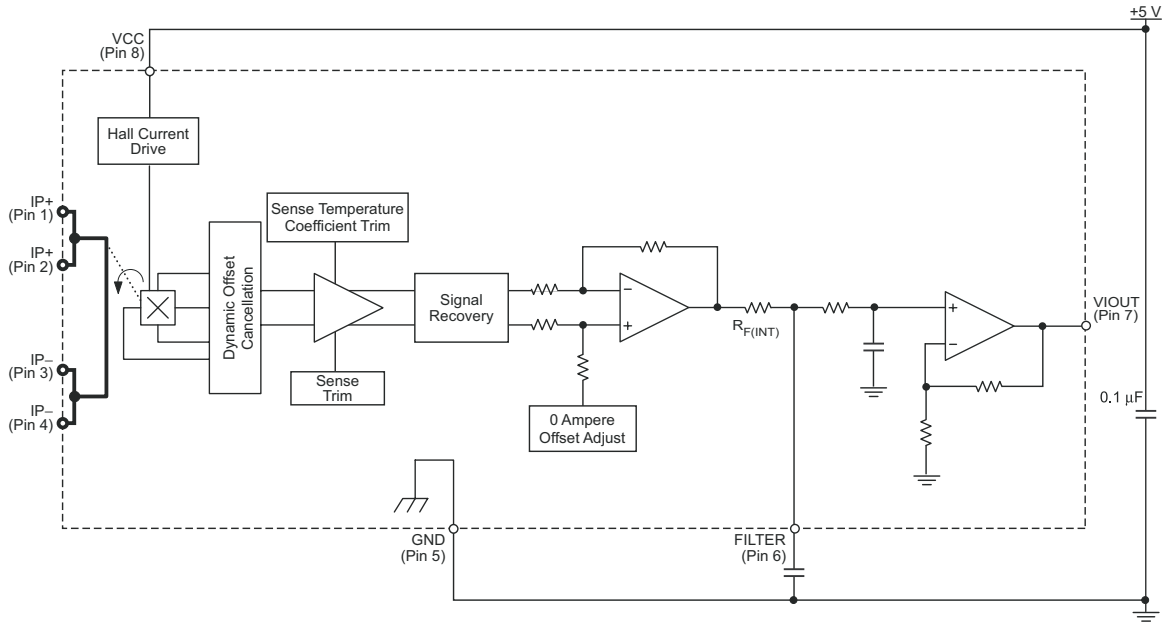
*Contact Allegro for additional packing options.

Absolute Maximum Ratings

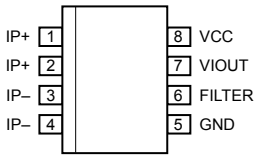
Characteristic	Symbol	Notes	Rating	Units
Supply Voltage	V_{CC}		8	V
Reverse Supply Voltage	V_{RCC}		–0.1	V
Output Voltage	V_{IOUT}		8	V
Reverse Output Voltage	V_{RIOUT}		–0.1	V
Reinforced Isolation Voltage	V_{ISO}	Pins 1-4 and 5-8; 60 Hz, 1 minute, T_A=25°C	2100	V
		Voltage applied to leadframe (Ip+ pins), based on IEC 60950	184	V_{peak}
Basic Isolation Voltage	$V_{ISO(bsc)}$	Pins 1-4 and 5-8; 60 Hz, 1 minute, T_A=25°C	1500	V
		Voltage applied to leadframe (Ip+ pins), based on IEC 60950	354	V_{peak}
Output Current Source	$I_{IOUT(Source)}$		3	mA
Output Current Sink	$I_{IOUT(Sink)}$		10	mA
Overcurrent Transient Tolerance	I_P	1 pulse, 100 ms	100	A
Nominal Operating Ambient Temperature	T_A	Range E	–40 to 85	ºC
Maximum Junction Temperature	T_J(max)		165	ºC
Storage Temperature	T_{stg}		–65 to 170	ºC

Parameter	Specification
Fire and Electric Shock	CAN/CSA-C22.2 No. 60950-1-03 UL 60950-1:2003 EN 60950-1:2001

Functional Block Diagram



Pin-out Diagram



Terminal List Table

Number	Name	Description
1 and 2	IP+	Terminals for current being sensed; fused internally
3 and 4	IP–	Terminals for current being sensed; fused internally
5	GND	Signal ground terminal
6	FILTER	Terminal for external capacitor that sets bandwidth
7	VIOUT	Analog output signal
8	VCC	Device power supply terminal

COMMON OPERATING CHARACTERISTICS[1] over full range of T_A, C_F = 1 nF, and V_{CC} = 5 V, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
ELECTRICAL CHARACTERISTICS						
Supply Voltage	V_{CC}		4.5	5.0	5.5	V
Supply Current	I_{CC}	V_{CC} = 5.0 V, output open	–	10	13	mA
Output Capacitance Load	C_{LOAD}	VIOUT to GND	–	–	10	nF
Output Resistive Load	R_{LOAD}	VIOUT to GND	4.7	–	–	kΩ
Primary Conductor Resistance	$R_{PRIMARY}$	T_A = 25°C	–	1.2	–	mΩ
Rise Time	t_r	I_P = I_P(max), T_A = 25°C, C_{OUT} = open	–	5	–	µs
Frequency Bandwidth	f	–3 dB, T_A = 25°C; I_P is 10 A peak-to-peak	–	80	–	kHz
Nonlinearity	E_{LIN}	Over full range of I_P	–	1.5	–	%
Symmetry	E_{SYM}	Over full range of I_P	98	100	102	%
Zero Current Output Voltage	$V_{IOUT(Q)}$	Bidirectional; I_P = 0 A, T_A = 25°C	–	V_{CC} × 0.5	–	V
Power-On Time	t_{PO}	Output reaches 90% of steady-state level, T_J = 25°C, 20 A present on leadframe	–	35	–	µs
Magnetic Coupling[2]			–	12	–	G/A
Internal Filter Resistance[3]	$R_{F(INT)}$			1.7		kΩ

[1]Device may be operated at higher primary current levels, I_P, and ambient, T_A, and internal leadframe temperatures, T_A, provided that the Maximum Junction Temperature, T_J(max), is not exceeded.

[2]1G = 0.1 mT.

[3]$R_{F(INT)}$ forms an RC circuit via the FILTER pin.

COMMON THERMAL CHARACTERISTICS[1]

			Min.	Typ.	Max.	Units
Operating Internal Leadframe Temperature	T_A	E range	–40	–	85	°C
					Value	Units
Junction-to-Lead Thermal Resistance[2]	$R_{\theta JL}$	Mounted on the Allegro ASEK 712 evaluation board			5	°C/W
Junction-to-Ambient Thermal Resistance	$R_{\theta JA}$	Mounted on the Allegro 85-0322 evaluation board, includes the power consumed by the board			23	°C/W

[1]Additional thermal information is available on the Allegro website.

[2]The Allegro evaluation board has 1500 mm^2 of 2 oz. copper on each side, connected to pins 1 and 2, and to pins 3 and 4, with thermal vias connecting the layers. Performance values include the power consumed by the PCB. Further details on the board are available from the Frequently Asked Questions document on our website. Further information about board design and thermal performance also can be found in the Applications Information section of this datasheet.

x05B PERFORMANCE CHARACTERISTICS T_A = –40°C to 85°C[1], C_F = 1 nF, and V_{CC} = 5 V, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
Optimized Accuracy Range	I_P		–5	–	5	A
Sensitivity	Sens	Over full range of I_P, T_A = 25°C	180	185	190	mV/A
Noise	$V_{NOISE(PP)}$	Peak-to-peak, T_A = 25°C, 185 mV/A programmed Sensitivity, C_F = 47 nF, C_{OUT} = open, 2 kHz bandwidth	–	21	–	mV
Zero Current Output Slope	$\Delta I_{OUT(Q)}$	T_A = –40°C to 25°C	–	–0.26	–	mV/°C
		T_A = 25°C to 150°C	–	–0.08	–	mV/°C
Sensitivity Slope	ΔSens	T_A = –40°C to 25°C	–	0.054	–	mV/A/°C
		T_A = 25°C to 150°C	–	–0.008	–	mV/A/°C
Total Output Error[2]	E_{TOT}	I_P =±5 A, T_A = 25°C	–	±1.5	–	%

[1]Device may be operated at higher primary current levels, I_P, and ambient temperatures, T_A, provided that the Maximum Junction Temperature, $T_{J(max)}$, is not exceeded.

[2]Percentage of I_P, with I_P = 5 A. Output filtered.

x20A PERFORMANCE CHARACTERISTICS T_A = –40°C to 85°C[1], C_F = 1 nF, and V_{CC} = 5 V, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
Optimized Accuracy Range	I_P		–20	–	20	A
Sensitivity	Sens	Over full range of I_P, T_A = 25°C	96	100	104	mV/A
Noise	$V_{NOISE(PP)}$	Peak-to-peak, T_A = 25°C, 100 mV/A programmed Sensitivity, C_F = 47 nF, C_{OUT} = open, 2 kHz bandwidth	–	11	–	mV
Zero Current Output Slope	$\Delta I_{OUT(Q)}$	T_A = –40°C to 25°C	–	–0.34	–	mV/°C
		T_A = 25°C to 150°C	–	–0.07	–	mV/°C
Sensitivity Slope	ΔSens	T_A = –40°C to 25°C	–	0.017	–	mV/A/°C
		T_A = 25°C to 150°C	–	–0.004	–	mV/A/°C
Total Output Error[2]	E_{TOT}	I_P =±20 A, T_A = 25°C	–	±1.5	–	%

[1]Device may be operated at higher primary current levels, I_P, and ambient temperatures, T_A, provided that the Maximum Junction Temperature, T_J(max), is not exceeded.

[2]Percentage of I_P, with I_P = 20 A. Output filtered.

x30A PERFORMANCE CHARACTERISTICS T_A = –40°C to 85°C[1], C_F = 1 nF, and V_{CC} = 5 V, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
Optimized Accuracy Range	I_P		–30	–	30	A
Sensitivity	Sens	Over full range of I_P, T_A = 25°C	64	66	68	mV/A
Noise	$V_{NOISE(PP)}$	Peak-to-peak, T_A = 25°C, 66 mV/A programmed Sensitivity, C_F = 47 nF, C_{OUT} = open, 2 kHz bandwidth	–	7	–	mV
Zero Current Output Slope	$\Delta I_{OUT(Q)}$	T_A = –40°C to 25°C	–	–0.35	–	mV/°C
		T_A = 25°C to 150°C	–	–0.08	–	mV/°C
Sensitivity Slope	ΔSens	T_A = –40°C to 25°C	–	0.007	–	mV/A/°C
		T_A = 25°C to 150°C	–	–0.002	–	mV/A/°C
Total Output Error[2]	E_{TOT}	I_P = ±30 A, T_A = 25°C	–	±1.5	–	%

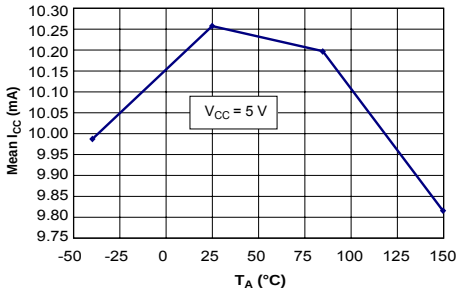
[1]Device may be operated at higher primary current levels, I_P, and ambient temperatures, T_A, provided that the Maximum Junction Temperature, T_J(max), is not exceeded.

[2]Percentage of I_P, with I_P = 30 A. Output filtered.

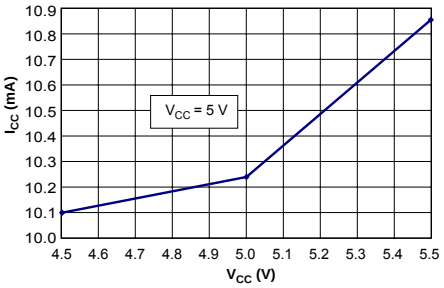
Characteristic Performance

I_P = 5 A, unless otherwise specified

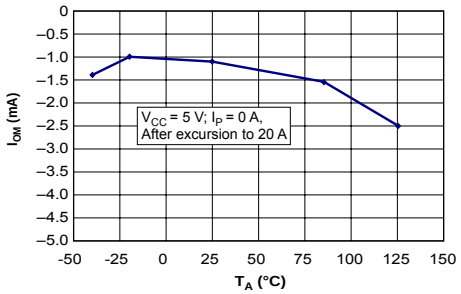
Mean Supply Current versus Ambient Temperature



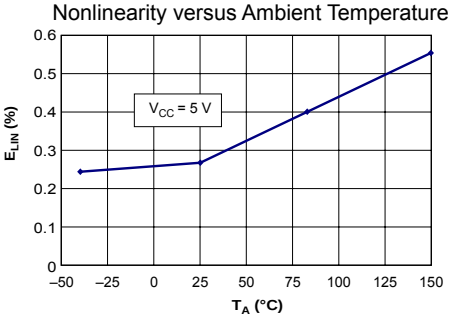
Supply Current versus Supply Voltage



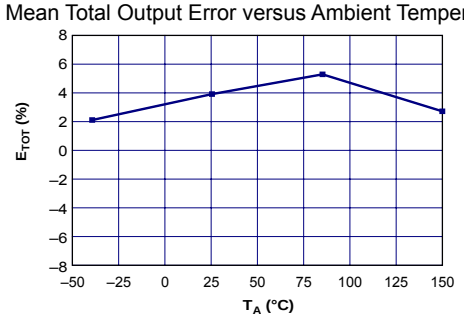
Magnetic Offset versus Ambient Temperature



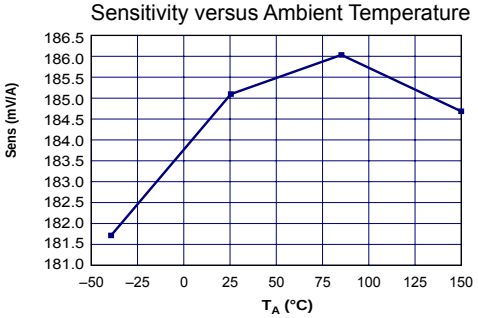
Nonlinearity versus Ambient Temperature



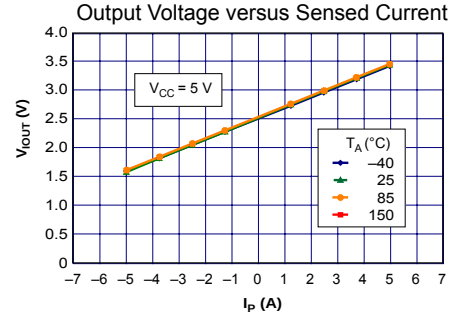
Mean Total Output Error versus Ambient Temperature



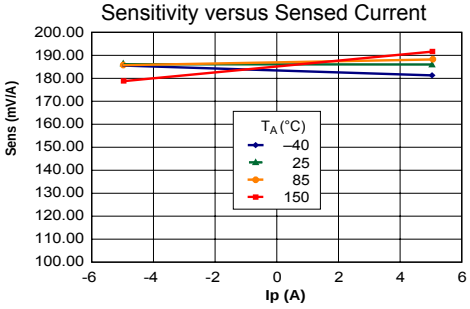
Sensitivity versus Ambient Temperature



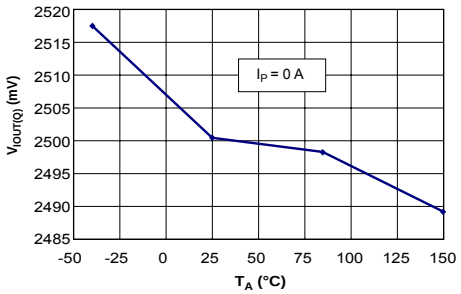
Output Voltage versus Sensed Current



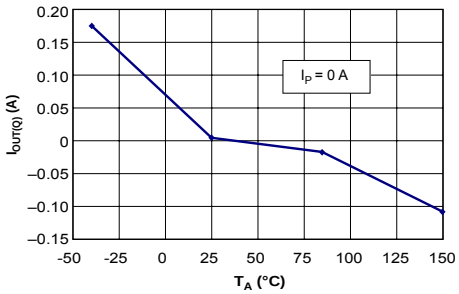
Sensitivity versus Sensed Current



0 A Output Voltage versus Ambient Temperature



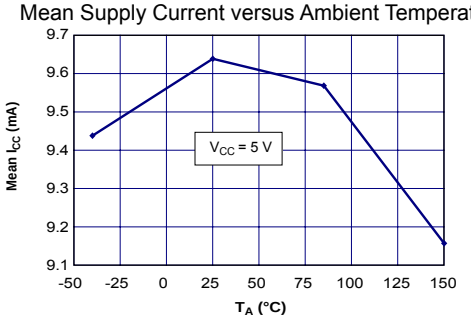
0 A Output Voltage Current versus Ambient Temperature



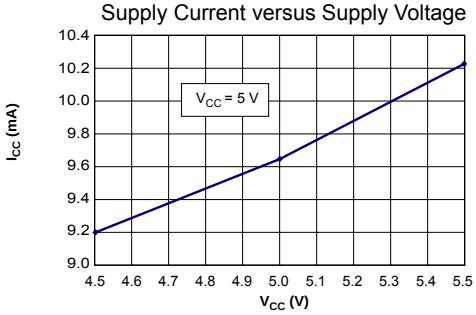
Characteristic Performance

I_P = 20 A, unless otherwise specified

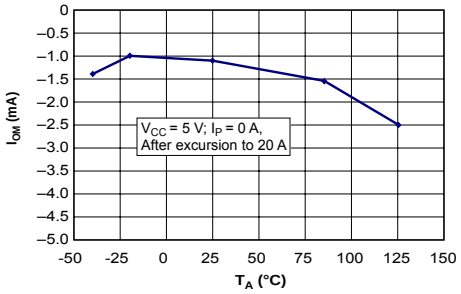
Mean Supply Current versus Ambient Temperature



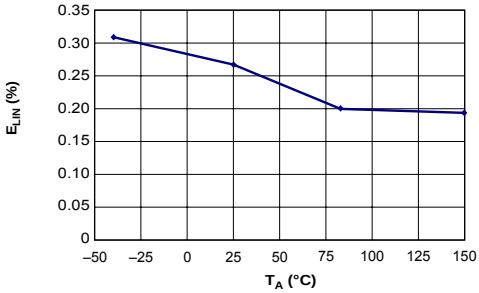
Supply Current versus Supply Voltage



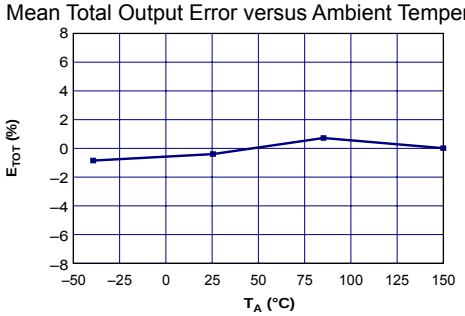
Magnetic Offset versus Ambient Temperature



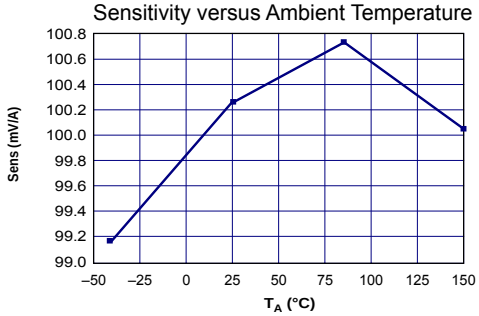
Nonlinearity versus Ambient Temperature



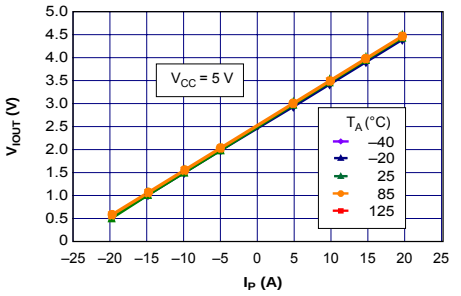
Mean Total Output Error versus Ambient Temperature



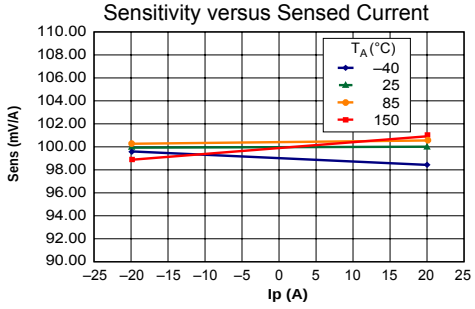
Sensitivity versus Ambient Temperature



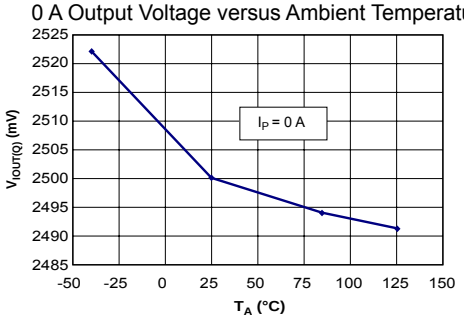
Output Voltage versus Sensed Current



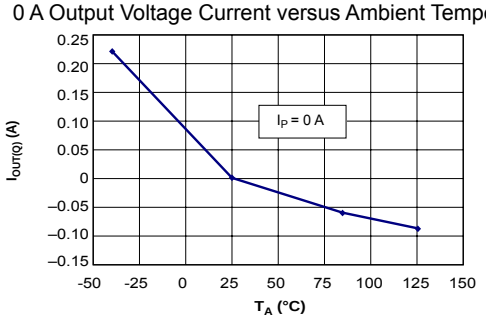
Sensitivity versus Sensed Current



0 A Output Voltage versus Ambient Temperature



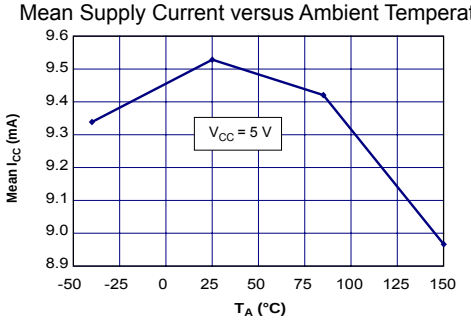
0 A Output Voltage Current versus Ambient Temperature



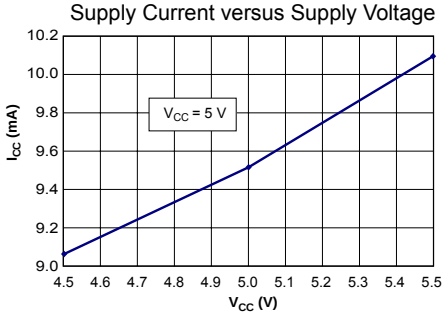
Characteristic Performance

I_P = 30 A, unless otherwise specified

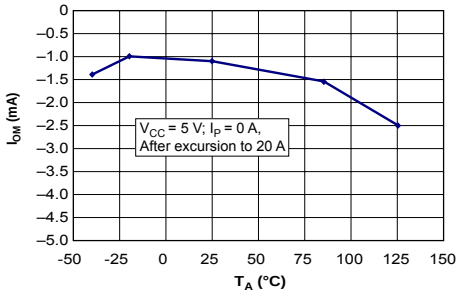
Mean Supply Current versus Ambient Temperature



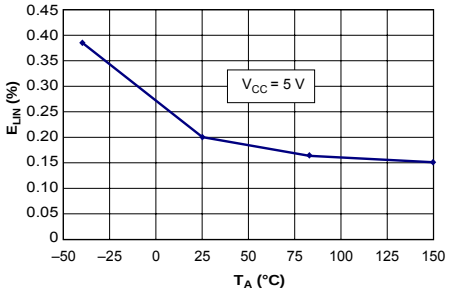
Supply Current versus Supply Voltage



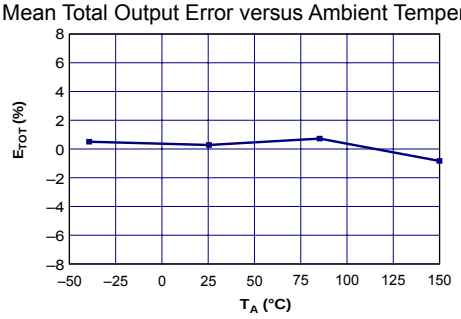
Magnetic Offset versus Ambient Temperature



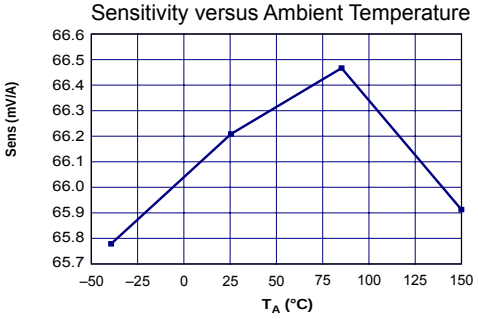
Nonlinearity versus Ambient Temperature



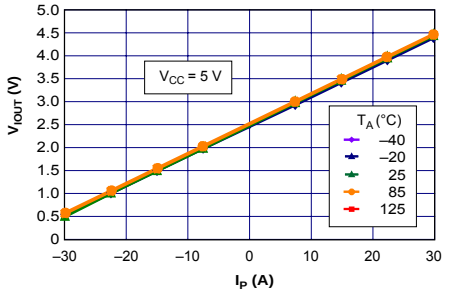
Mean Total Output Error versus Ambient Temperature



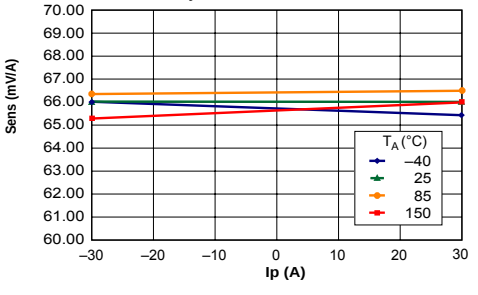
Sensitivity versus Ambient Temperature



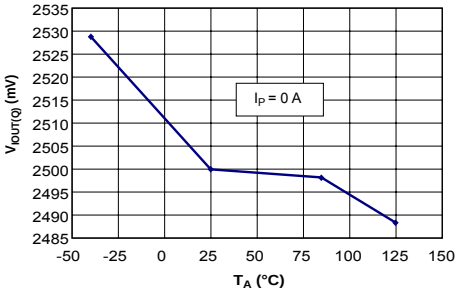
Output Voltage versus Sensed Current



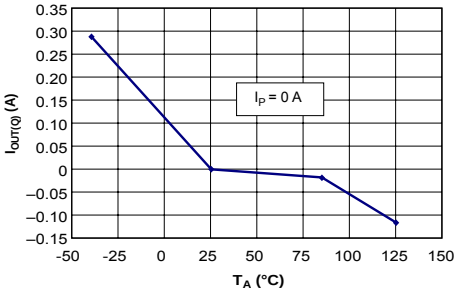
Sensitivity versus Sensed Current



0 A Output Voltage versus Ambient Temperature



0 A Output Voltage Current versus Ambient Temperature



Definitions of Accuracy Characteristics

Sensitivity (Sens). The change in sensor output in response to a 1 A change through the primary conductor. The sensitivity is the product of the magnetic circuit sensitivity (G/A) and the linear IC amplifier gain (mV/G). The linear IC amplifier gain is programmed at the factory to optimize the sensitivity (mV/A) for the full-scale current of the device.

Noise (V_{NOISE}). The product of the linear IC amplifier gain (mV/G) and the noise floor for the Allegro Hall effect linear IC (≈1 G). The noise floor is derived from the thermal and shot noise observed in Hall elements. Dividing the noise (mV) by the sensitivity (mV/A) provides the smallest current that the device is able to resolve.

Linearity (E_{LIN}). The degree to which the voltage output from the sensor varies in direct proportion to the primary current through its full-scale amplitude. Nonlinearity in the output can be attributed to the saturation of the flux concentrator approaching the full-scale current. The following equation is used to derive the linearity:

$$100\left\{1-\left[\frac{\Delta\text{ gain}\times\%\text{ sat}\,(V_{\text{IOUT_full-scale amperes}}-V_{\text{IOUT(Q)}})}{2\,(V_{\text{IOUT_half-scale amperes}}-V_{\text{IOUT(Q)}})}\right]\right\}$$

where $V_{\text{IOUT_full-scale amperes}}$ = the output voltage (V) when the sensed current approximates full-scale $\pm I_P$.

Symmetry (E_{SYM}). The degree to which the absolute voltage output from the sensor varies in proportion to either a positive or negative full-scale primary current. The following formula is used to derive symmetry:

$$100\left(\frac{V_{\text{IOUT_+ full-scale amperes}}-V_{\text{IOUT(Q)}}}{V_{\text{IOUT(Q)}}-V_{\text{IOUT_–full-scale amperes}}}\right)$$

Quiescent output voltage ($V_{IOUT(Q)}$). The output of the sensor when the primary current is zero. For a unipolar supply voltage, it nominally remains at $V_{CC}/2$. Thus, V_{CC} = 5 V translates into $V_{IOUT(Q)}$ = 2.5 V. Variation in $V_{IOUT(Q)}$ can be attributed to the resolution of the Allegro linear IC quiescent voltage trim and thermal drift.

Electrical offset voltage (V_{OE}). The deviation of the device output from its ideal quiescent value of $V_{CC}/2$ due to nonmagnetic causes. To convert this voltage to amperes, divide by the device sensitivity, Sens.

Accuracy (E_{TOT}). The accuracy represents the maximum deviation of the actual output from its ideal value. This is also known as the total ouput error. The accuracy is illustrated graphically in the output voltage versus current chart at right.

Accuracy is divided into four areas:

- **0 A at 25°C.** Accuracy of sensing zero current flow at 25°C, without the effects of temperature.
- **0 A over Δ temperature.** Accuracy of sensing zero current flow including temperature effects.
- **Full-scale current at 25°C.** Accuracy of sensing the full-scale current at 25°C, without the effects of temperature.
- **Full-scale current over Δ temperature.** Accuracy of sensing full-scale current flow including temperature effects.

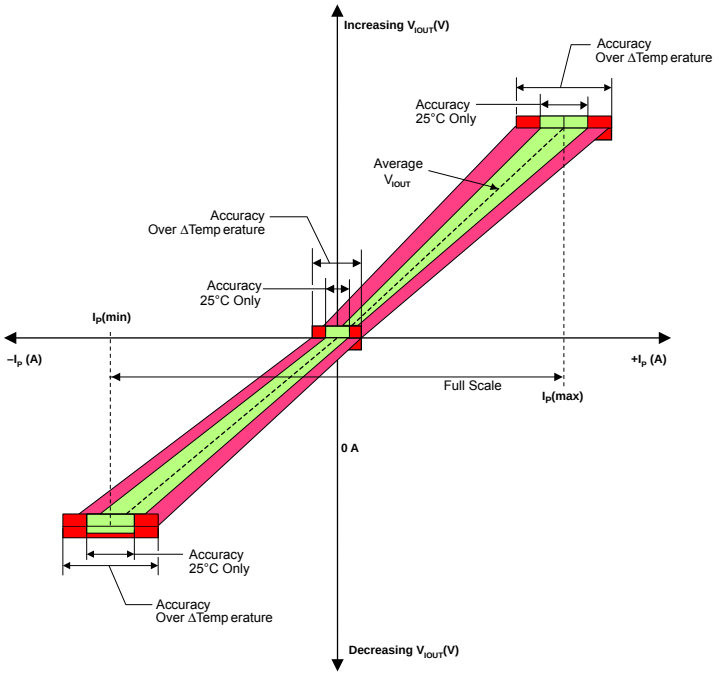
Ratiometry. The ratiometric feature means that its 0 A output, $V_{IOUT(Q)}$, (nominally equal to $V_{CC}/2$) and sensitivity, Sens, are proportional to its supply voltage, V_{CC}. The following formula is used to derive the ratiometric change in 0 A output voltage, $\Delta V_{IOUT(Q)RAT}$ (%).

$$100\left(\frac{V_{\text{IOUT(Q)VCC}}/V_{\text{IOUT(Q)5V}}}{V_{CC}/5\text{ V}}\right)$$

The ratiometric change in sensitivity, $\Delta Sens_{RAT}$ (%), is defined as:

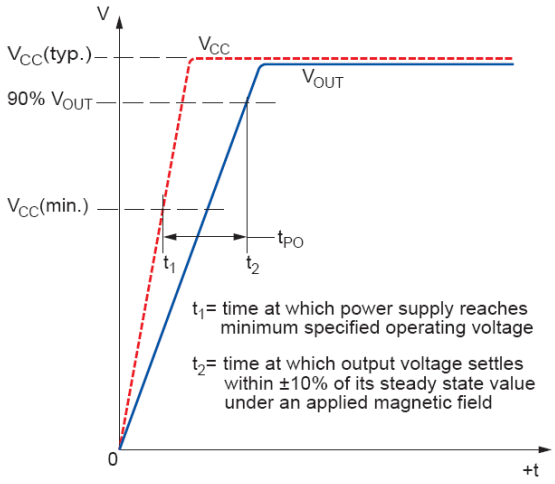
$$100\left(\frac{Sens_{VCC}/Sens_{5V}}{V_{CC}/5\text{ V}}\right)$$

Output Voltage versus Sensed Current
Accuracy at 0 A and at Full-Scale Current

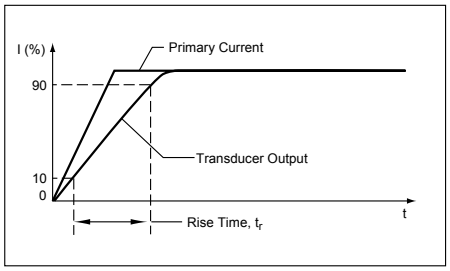


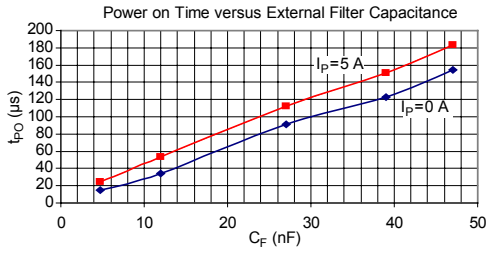
Definitions of Dynamic Response Characteristics

Power-On Time (t_{PO}). When the supply is ramped to its operating voltage, the device requires a finite time to power its internal components before responding to an input magnetic field. Power-On Time, t_{PO} , is defined as the time it takes for the output voltage to settle within ±10% of its steady state value under an applied magnetic field, after the power supply has reached its minimum specified operating voltage, V_{CC}(min), as shown in the chart at right.

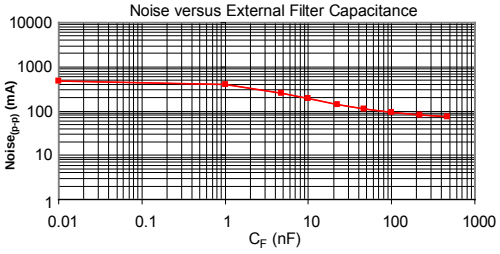


Rise time (t_r). The time interval between a) when the sensor reaches 10% of its full scale value, and b) when it reaches 90% of its full scale value. The rise time to a step response is used to derive the bandwidth of the current sensor, in which $f(-3\text{ dB}) = 0.35/t_r$. Both t_r and $t_{RESPONSE}$ are detrimentally affected by eddy current losses observed in the conductive IC ground plane.



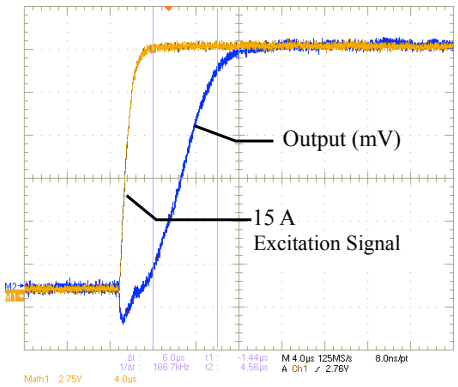


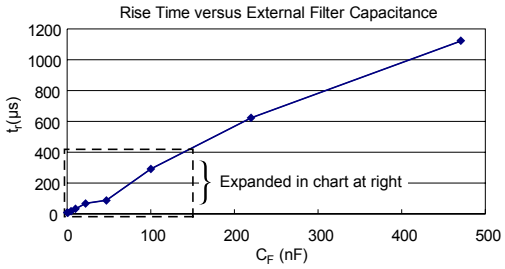
Noise vs. Filter Cap



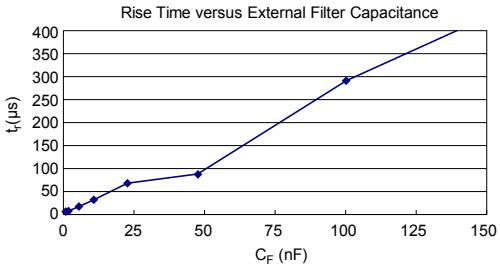
Step Response

T_A=25°C





C_F (nF)	t_r (µs)
0	6.6
1	7.7
4.7	17.4
10	32.1
22	68.2
47	88.2
100	291.3
220	623.0
470	1120.0

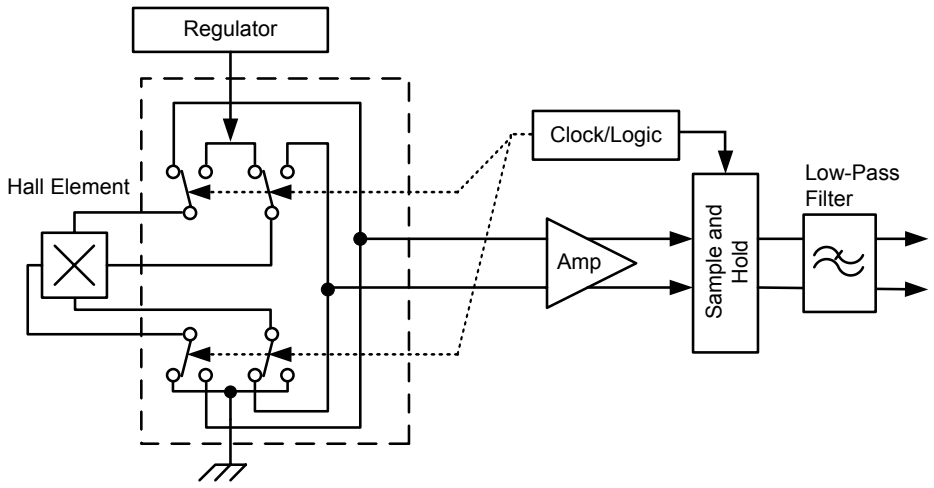


Chopper Stabilization Technique

Chopper Stabilization is an innovative circuit technique that is used to minimize the offset voltage of a Hall element and an associated on-chip amplifier. Allegro patented a Chopper Stabilization technique that nearly eliminates Hall IC output drift induced by temperature or package stress effects. This offset reduction technique is based on a signal modulation-demodulation process. Modulation is used to separate the undesired dc offset signal from the magnetically induced signal in the frequency domain. Then, using a low-pass filter, the modulated dc offset is suppressed while the magnetically induced signal passes through the filter.

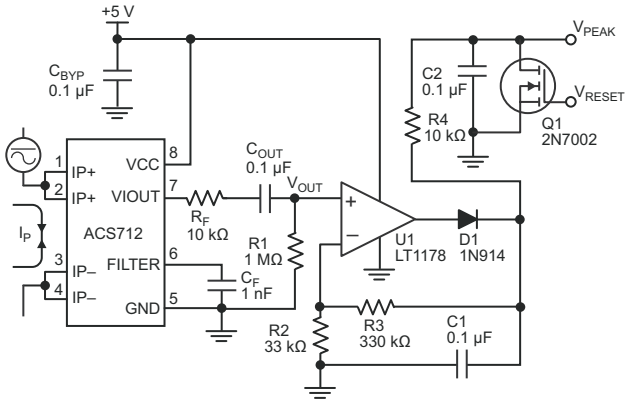
As a result of this chopper stabilization approach, the output voltage from the Hall IC is desensitized to the effects of temperature and mechanical stress. This technique produces devices that have an extremely stable Electrical Offset Voltage, are immune to thermal stress, and have precise recoverability after temperature cycling.

This technique is made possible through the use of a BiCMOS process that allows the use of low-offset and low-noise amplifiers in combination with high-density logic integration and sample and hold circuits.

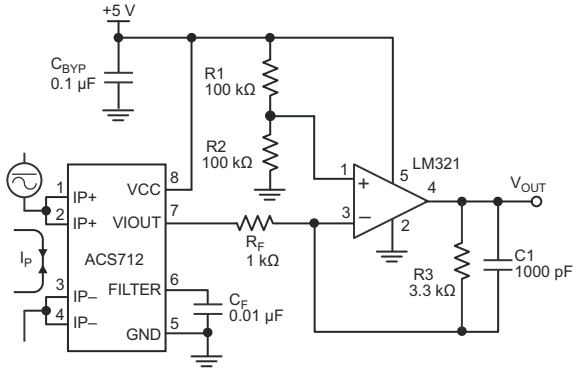


Concept of Chopper Stabilization Technique

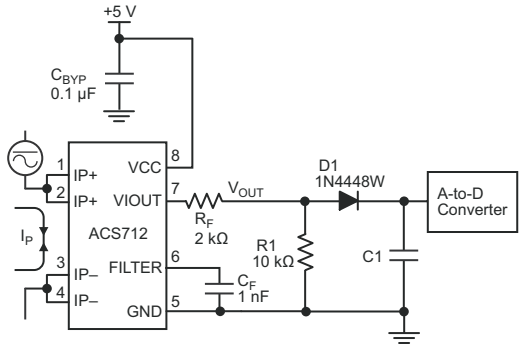
Typical Applications



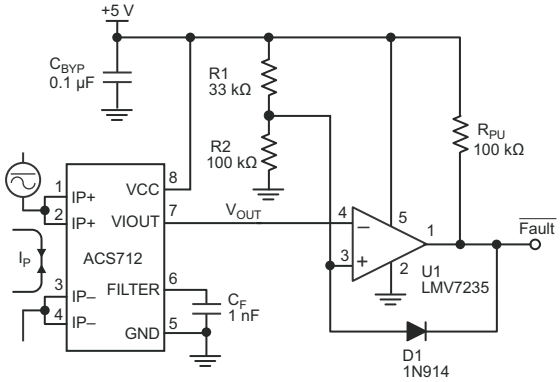
Application 2. Peak Detecting Circuit



Application 3. This configuration increases gain to 610 mV/A (tested using the ACS712ELC-05A).



Application 4. Rectified Output. 3.3 V scaling and rectification application for A-to-D converters. Replaces current transformer solutions with simpler ACS circuit. C1 is a function of the load resistance and filtering desired. R1 can be omitted if the full range is desired.



Application 5. 10 A Overcurrent Fault Latch. Fault threshold set by R1 and R2. This circuit latches an overcurrent fault and holds it until the 5 V rail is powered down.

Improving Sensing System Accuracy Using the FILTER Pin

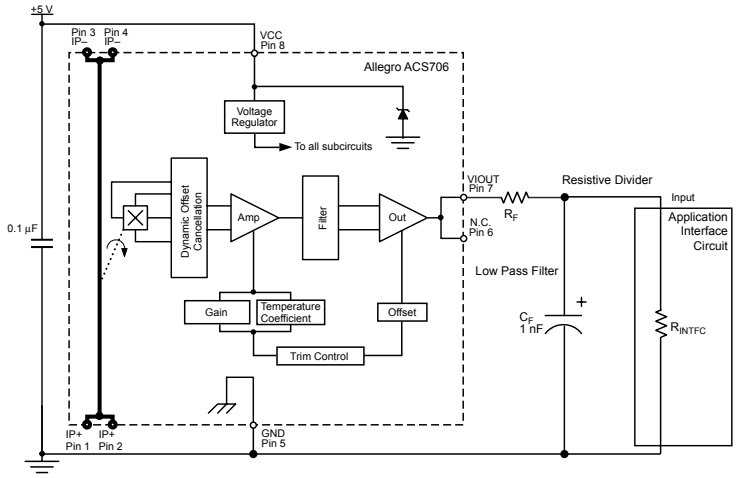
In low-frequency sensing applications, it is often advantageous to add a simple RC filter to the output of the sensor. Such a low-pass filter improves the signal-to-noise ratio, and therefore the resolution, of the sensor output signal. However, the addition of an RC filter to the output of a sensor IC can result in undesirable sensor output attenuation — even for dc signals.

Signal attenuation, ΔV_{ATT}, is a result of the resistive divider effect between the resistance of the external filter, R_F (see Application 6), and the input impedance and resistance of the customer interface circuit, R_{INTFC}. The transfer function of this resistive divider is given by:

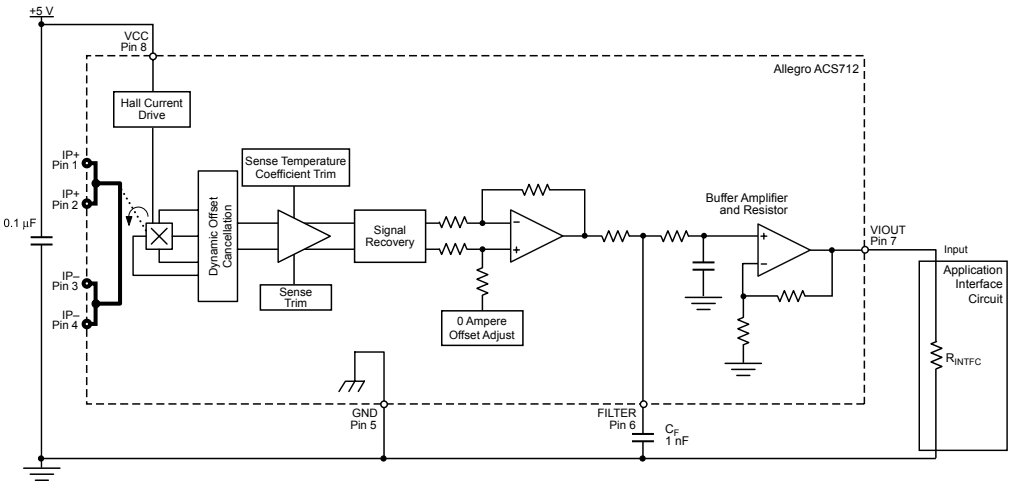
$$\Delta V_{ATT} = V_{IOUT}\left(\frac{R_{INTFC}}{R_F + R_{INTFC}}\right) .$$

Even if R_F and R_{INTFC} are designed to match, the two individual resistance values will most likely drift by different amounts over temperature. Therefore, signal attenuation will vary as a function of temperature. Note that, in many cases, the input impedance, R_{INTFC}, of a typical analog-to-digital converter (ADC) can be as low as 10 kΩ.

The ACS712 contains an internal resistor, a FILTER pin connection to the printed circuit board, and an internal buffer amplifier. With this circuit architecture, users can implement a simple RC filter via the addition of a capacitor, C_F (see Application 7) from the FILTER pin to ground. The buffer amplifier inside of the ACS712 (located after the internal resistor and FILTER pin connection) eliminates the attenuation caused by the resistive divider effect described in the equation for ΔV_{ATT}. Therefore, the ACS712 device is ideal for use in high-accuracy applications that cannot afford the signal attenuation associated with the use of an external RC low-pass filter.

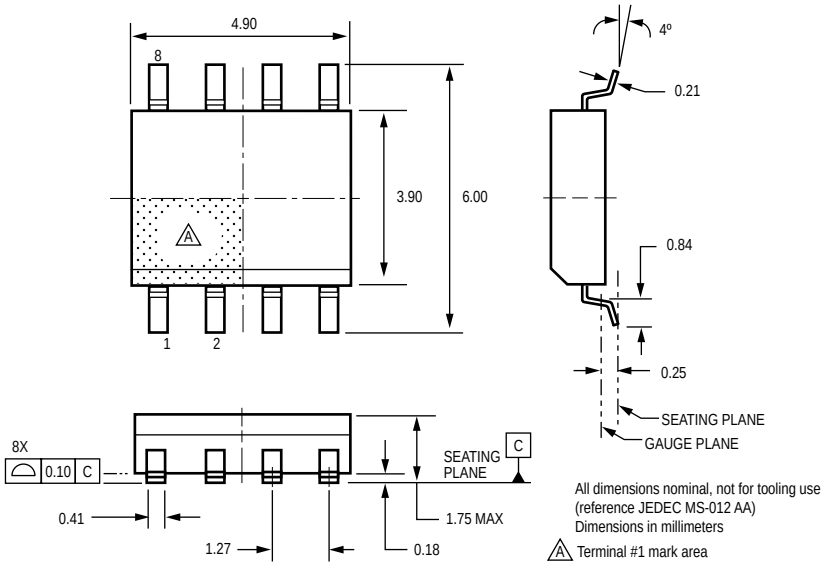


Application 6. When a low pass filter is constructed externally to a standard Hall effect device, a resistive divider may exist between the filter resistor, R_F, and the resistance of the customer interface circuit, R_{INTFC}. This resistive divider will cause excessive attenuation, as given by the transfer function for ΔV_{ATT}.



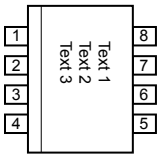
Application 7. Using the FILTER pin provided on the ACS712 eliminates the attenuation effects of the resistor divider between R_F and R_{INTFC}, shown in Application 6.

Package LC, 8-pin SOIC



Package Branding

Two alternative patterns are used



ACS712T *RLCPPP* *YYWWA*	ACS	Allegro Current Sensor
	712	Device family number
	T	Indicator of 100% matte tin leadframe plating
	R	Operating ambient temperature range code
	LC	Package type designator
	PPP	Primary sensed current
	YY	Date code: Calendar year (last two digits)
	WW	Date code: Calendar week
	A	Date code: Shift code

ACS712T *RLCPPP* *L...L* *YYWW*	ACS	Allegro Current Sensor
	712	Device family number
	T	Indicator of 100% matte tin leadframe plating
	R	Operating ambient temperature range code
	LC	Package type designator
	PPP	Primary sensed current
	L...L	Lot code
	YY	Date code: Calendar year (last two digits)
	WW	Date code: Calendar week

Copyright ©2006, 2007, Allegro MicroSystems, Inc.

The products described herein are manufactured under one or more of the following U.S. patents: 5,045,920; 5,264,783; 5,442,283; 5,389,889; 5,581,179; 5,517,112; 5,619,137; 5,621,319; 5,650,719; 5,686,894; 5,694,038; 5,729,130; 5,917,320; and other patents pending.

Allegro MicroSystems, Inc. reserves the right to make, from time to time, such departures from the detail specifications as may be required to permit improvements in the performance, reliability, or manufacturability of its products. Before placing an order, the user is cautioned to verify that the information being relied upon is current.

Allegro's products are not to be used in life support devices or systems, if a failure of an Allegro product can reasonably be expected to cause the failure of that life support device or system, or to affect the safety or effectiveness of that device or system.

The information included herein is believed to be accurate and reliable. However, Allegro MicroSystems, Inc. assumes no responsibility for its use; nor for any infringement of patents or other rights of third parties which may result from its use.

For the latest version of this document, visit our website:

www.allegromicro.com