



**UNIVERSIDAD AUTONOMA DEL ESTADO DE MEXICO
CENTRO UNIVERSITARIO UAEM ATLACOMULCO**

PROYECTO DE INVESTIGACION

**“CONTROL INTELIGENTE PARA UNA FUENTE DE CORRIENTE
IMPLEMENTADO EN UN SISTEMA EMBEBIDO”**

QUE PARA OBTENER EL GRADO DE MAESTRO EN CIENCIA DE LA
COMPUTACION

PRESENTA:

LUIS ANTONIO FLORES LUNA

DIRECTOR:

DR. ALLAN ANTONIO FLORES FUENTES

CODIRECTORES:

DR. ERICK ROJAS RAMÍREZ

DR. EVERADRO EFRÉN GRANDA GUTIÉRREZ



SEPTIEMBRE DE 2025



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Productos académicos

Finite State Machine based-control for a two-quadrant converter in current mode. Publicado en: Revista Aristas, Ciencia Básica y Aplicada Vol. 12 Núm. 20 (2025). Disponible en: [Finite State Machine based-control for a two-quadrant converter in current mode. | Revista Aristas](#)

Retribuciones académicas

Taller de MATLAB Simscape en Simulink básico. Impartido en: Centro Universitario UAEM Atlacomulco.

Taller de MATLAB Simulink Básico Impartido en: Universidad Tecnología del Valle de Toluca.



Índice

Resumen.....	7
1. Introducción	9
1.1 Antecedentes	9
1.2 Planteamiento del problema	10
1.3 Definición del problema	11
1.4 Objetivo de la investigación	11
1.4.1 Objetivo general	11
1.4.2 Objetivos específicos.....	11
1.5 Preguntas de investigación	12
1.6 Justificación	12
1.7 Impacto.....	12
1.8 Hipótesis	13
1.9 Cronograma de actividades	13
2. Estado del Arte.....	15
2.2 Convertidores estáticos de energía	15
3. Metodología.....	44
3.1 Características del convertidor	44
3.2 Simulación del convertidor chopper en lazo abierto	49
3.3 Diseño y simulación de la máquina de estados finitos en lazo cerrado	53
3.4 Optimización en la máquina de estados con el controlador PI	56
3.4.1 Configuración del control PI	58
4. Desarrollo y análisis	60
4.1 Introducción.....	60
4.2 Vista general del sistema.....	60
4.3. Diagrama a bloques del sistema	61
4.3.1. Placa de desarrollo Curiosity Nano®.	64
4.3.2. Acoplamiento óptico	65
4.3.3. <i>Drivers</i> o manejadores para MOSFET	67
4.3.4. Etapa de potencia: Convertidor en modo corriente	72
4.3.5. Sección de instrumentación.....	74
4.4 Desarrollo e implementación del código de la MEF y el controlador PI.....	78
4.5 Desarrollo e implementación de la interfaz para monitoreo de parámetros.	79
5. Resultados	82
5.1 Resultados de simulación del sistema	82
5.2.1 Resultados de simulación del controlador MEF	84



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

5.2.2 Resultados de simulación del controlador MEF-PI	85
5.3 Resultados del sistema físico	89
5.3.1 Señales de conmutación del microcontrolador	90
5.3.2 Señales del convertidor a lazo abierto	94
5.3.3 Señales del convertidor a lazo cerrado	96
Conclusión	102
Trabajo a futuro	104
Referencias	105

Índice de figuras

<i>Figura 2. 1 Conversión de energía eléctrica realizada por diferentes topologías de CEE.....</i>	<i>15</i>
<i>Figura 2. 2 Tipos de convertidores de CD-CD</i>	<i>16</i>
<i>Figura 2. 3 Clasificación de dispositivos electrónicos de potencia a partir de sus parámetros eléctricos y aplicaciones.....</i>	<i>17</i>
<i>Figura 2. 4 Diagrama de bloques de controlador PID</i>	<i>19</i>
<i>Figura 2. 5 Representación gráfica del estado estable y transitorio.....</i>	<i>22</i>
<i>Figura 2. 6 Técnicas control utilizadas en sistemas eléctricos.....</i>	<i>24</i>
<i>Figura 2. 7 Ejemplo de un grafo de una máquina de estados finitos y sus transiciones.</i>	<i>26</i>
<i>Figura 2. 8 Clasificación de sistemas embebidos.</i>	<i>27</i>
<i>Figura 3. 1 Cuadrantes de operación del convertidor chopper en modo corriente</i>	<i>45</i>
<i>Figura 3. 2 (a) Cuadrante de operación IV, (b) Cuadrante de operación I</i>	<i>45</i>
<i>Figura 3. 3 Topología y estados de operación del convertidor.....</i>	<i>47</i>
<i>Figura 3. 4 Asignación de tiempo y definición de tiempo de simulación</i>	<i>50</i>
<i>Figura 3. 5 (a) Primer hoja de datos del componente [51]. (b) Interfaz de asignación de valores del componente MOSFET en MATLAB/Simscape [52]</i>	<i>51</i>
<i>Figura 3. 6 Visualización y simulación del convertidor chopper desarrollada en Simscape de MATLAB/Simulink</i>	<i>52</i>
<i>Figura 3. 7 Bloque PWM_Generator.....</i>	<i>52</i>
<i>Figura 3. 8 a) Límites superior e inferior de la corriente de salida. b) Operaciones matemáticas para la obtención de límite superior (UpL) e inferior (LowL).</i>	<i>54</i>
<i>Figura 3. 9 a) Grafo de la máquina de estados basado en el diseño de Mealy. b) Máquina de estados adaptada al software de simulación Stateflow de MATLAB</i>	<i>55</i>
<i>Figura 3. 10 Diagrama general de del sistema con la MEF</i>	<i>56</i>
<i>Figura 3. 11 Máquina de estados finitos con controlador PI en estado estacionario.....</i>	<i>57</i>
<i>Figura 3. 12 Configuración del controlador PI en el estado estacionario</i>	<i>57</i>



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Figura 3. 13 Ventana para fijar los parámetros del controlador PI	58
Figura 3. 14 Esquema general de la MEF y el controlador PI	59
Figura 4. 1 Vista general del prototipo PCB diseñado para su manufactura	61
Figura 4. 2 Diagrama de bloques de los subsistemas	62
Figura 4. 3 Diagrama de conexiones eléctricas del sistema	63
Figura 4. 4 Diagrama a bloques del microcontrolador PIC18F57Q10	64
Figura 4. 5 Diagrama de conexión del microcontrolador	65
Figura 4. 6 (a) Optoacoplador para la señal PWM1. (b) Optoacoplador para la señal PWM2.....	66
Figura 4. 7 Topología del aislamiento galvánico utilizando un transformador de pulsos.	68
Figura 4. 8 Diagrama de flujo del script desarrollado	70
Figura 4. 9 (a) Transformador de pulsos para el MOSFET M1. (b) Transformador de pulsos para el MOSFET M2.....	71
Figura 4. 10 Convertidor chopper con red snubber implementada.	73
Figura 4. 11 (a) Sensor de corriente. (b) Sensor de voltaje.	76
Figura 4. 12 Diagrama de flujo del filtro RMS utilizado.....	77
Figura 4. 13 Diagrama de flujo del algoritmo de control en el microcontrolador.....	78
Figura 4. 14 Diagrama UML de aplicación	79
Figura 4. 15 Interfaz gráfica de monitoreo del sistema	80
Figura 4. 16 (a) Diagrama de flujo para la conexión del microcontrolador por comunicación serial (b) Diagrama de flujo para la desconexión del microcontrolador.....	81
Figura 5. 1 Señales de conmutación PWM1 y PWM2.....	82
Figura 5. 2 a) Voltaje de salida. b) Corriente de salida	83
Figura 5. 3 (a) Corriente de salida del convertidor con Set point a 12 A con FSM (b) Corriente de salida del convertidor con Set point a 20 A con FSM.....	84
Figura 5. 4 (a) Corriente de salida del convertidor con FSM y PI con Set point a 12A (b) Corriente de salida del convertidor con FSM y PI con Set point a 20A.....	85
Figura 5. 5 (a) Voltaje de salida con MEF. (b) Voltaje de salida con MEF + PI	86
Figura 5. 6 (a) Error del controlador MEF. (b) Error controlador MEF + PI.....	87
Figura 5. 7 (a) Valor del ciclo de trabajo del controlador MEF. (b) Valor del ciclo de trabajo del controlador MEF+PI	88
Figura 5. 8 Sistema físico conectado	89
Figura 5. 9 (a) Señal de conmutación del microcontrolador PWM1. (b) Señal de conmutación del microcontrolador PWM2	90



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

<i>Figura 5. 10 Señales de conmutación a la salida del controlador</i>	<i>91</i>
<i>Figura 5. 11 Señal de conmutación PWM1 y PWM2 a la salida del aislamiento galvánico</i>	<i>91</i>
<i>Figura 5. 12 Señal de conmutación en los pines Gate-Source de los MOSFET M1 y M2</i>	<i>92</i>
<i>Figura 5. 13 Señal de conmutación en los pines Drain-Source de los MOSFETS M1 y M2</i>	<i>93</i>
<i>Figura 5. 14 (a) Pulsos de conmutación en los pines Drenador-Fuente de los MOSFETS (b) Corriente de salida del convertidor.....</i>	<i>94</i>
<i>Figura 5. 15 (a) Eficiencias del convertidor a lazo abierto reales. (b) Eficiencias del convertidor a lazo abierto simuladas</i>	<i>95</i>
<i>Figura 5. 16 Circuito eléctrico para el cambio de carga en el convertidor.....</i>	<i>97</i>
<i>Figura 5. 17 Voltaje y corriente de salida del convertidor con cambio de carga.....</i>	<i>98</i>
<i>Figura 5. 18 Parámetros de salida del convertidor con MEF únicamente variando el valor del Set Point</i>	<i>99</i>
<i>Figura 5. 19 Parámetros de salida del convertidor con MEF y PI</i>	<i>100</i>



Resumen

Los convertidores estáticos de energía eléctrica son ampliamente utilizados en distintas áreas desde la industrial hasta el hogar y actualmente son un pilar fundamental en el proceso de conversión y adaptación de la energía. Existen diferentes tipos de convertidores cada uno con sus propiedades y características eléctricas que permiten determinar el convertidor adecuado para la aplicación. El convertidor utilizado para el desarrollo del presente trabajo de investigación es el nombrado troceador (*chopper*) el cual se caracteriza por tener una entrada de energía de CD y convertirla en otra de CD con características de amplitud de voltaje-corriente diferentes. Este tipo de convertidores presentan desafíos tales como el ruido electrónico, las pérdidas eléctricas por conmutación, el asilamiento correcto de cada uno de los componentes. El objetivo principal de este trabajo de investigación es diseñar, simular e implementar un sistema de control híbrido basado en una máquina de estados finito y un controlador proporcional-integral (PI). El controlador se simuló mediante el software con licencia educativa MATLAB/Simulink, y se desarrolló en el entorno de programación MPLAB utilizando una placa de desarrollo de nombre Curiosity Nano® de la marca Microchip® para las pruebas experimentales. El controlador propuesto redujo en un 25% el ruido eléctrico por conmutación, y alcanzó una eficiencia del convertidor de hasta el 90% en comparación con un controlador basado únicamente en una máquina de estado finito. En conclusión, el controlador híbrido es una propuesta viable en este tipo de convertidores.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Abstract

Static electrical energy converters are widely used in different areas, from industry to the home, and are currently a fundamental pillar in the process of energy conversion and adaptation. There are different types of converters, each with its own electrical properties and characteristics that allow the appropriate converter for the application to be determined. The converter used for the development of this research project is called a chopper, which is characterized by having a DC energy input and converting it into another DC input with different voltage-current amplitude characteristics. These types of converters present challenges such as electronic noise, electrical losses due to switching, and the correct isolation of each of the components. The main objective of this research work is to design, simulate, and implement a hybrid control system based on a finite state machine and a proportional-integral (PI) controller. The controller was simulated using MATLAB/Simulink educational license software and developed in the MPLAB programming environment using a Microchip® Curiosity Nano® development board for experimental testing. The proposed controller reduced electrical switching noise by 25% and achieved converter efficiency of up to 90% compared to a controller based solely on a finite state machine. In conclusion, the hybrid controller is a viable proposal for this type of converter.



1. Introducción

1.1 Antecedentes

El desarrollo de sistemas de conversión de energía requiere fundamentalmente el ajuste de parámetros para la carga eléctrica o la aplicación requerida, sin embargo, no se ha identificado en la literatura el desarrollo de un controlador, basado en una máquina de estados finitos aplicado en un convertidor troceador conocido como *chopper* clase D, de dos cuadrantes, en modo corriente. Este tipo de convertidor de corriente directa a corriente directa (C.D – C.D, por sus siglas) está conformado por dos transistores de conmutación y dos diodos auxiliares, tal como lo describe en [1]. El termino *chopper* se atribuye a que la tensión de salida es una versión cuasi rectangular, “Troceada” o “Cortada”, de la tensión continua de entrada, por lo cual, el convertidor *chopper* es capaz de cambiar la tensión fija de entrada a una tensión de salida ajustable. Existen algunos trabajos que se han enfocado en uso de diferentes técnicas de control de convertidores tipo chopper, con la finalidad de mantener el estado estable del sistema, en términos de la potencia de salida. En [2] se identificaron las variables de control y se llevó a cabo el desarrollo de una interfaz gráfica para regular la corriente de salida mediante la variación del ciclo de trabajo de los pulsos de conmutación aplicado a un sistema de descargas por barrera dieléctrica, en [3] se muestra una técnica de modulación de conmutación suave, para convertidores tipo boost de puente completo, donde la lógica de control se ha modelado con una máquina de estados finitos, en [4] se desarrolla una máquina de estados finitos aplicada a un microcontrolador con el objetivo de disminuir los costes de desarrollo de un convertidor matricial, en [5] se expone un modelo de máquina de estados finitos aplicado a un convertidor de doble puente activo para determinar el punto óptimo de operación, en [6] se implementa una máquina de estados finitos en un motor de imán permanente aplicado a un HVAC para mejorar la eficiencia del sistema, en [7] se describe la implementación de una máquina de estados utilizando un sistema embebido basado en FPGA aplicado a un dron el cual a su vez realiza las acciones correspondientes en los parámetros de control como la velocidad del motor y el sentido de giro. En [8], se utiliza



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

una máquina de estados aplicado a un convertidor estático de energía mediante un FPGA. No obstante, existe en la literatura la incorporación de un microcontrolador a un convertidor de energía utilizando una técnica de control proporcional. Esto indica que el microcontrolador es una solución ideal a sistemas de control, donde el procesamiento computacional no sea demandante. En [9] se muestra una aplicación de los microcontroladores para la ejecución de máquinas de estado, así como también la lógica difusa en un CEE. En conclusión, los microcontroladores son dispositivos ideales en aplicaciones de bajo costo, permitiendo el uso de algunas técnicas de control, como la utilización de un controlador proporcional. Sin embargo, se hará énfasis en la ejecución de una máquina de estados finitos aplicada a un convertidor chopper de dos cuadrantes clase D en modo corriente, de ser necesario se adicionará un controlador PID en alguna de sus variantes para mejorar la respuesta.

1.2 Planteamiento del problema

El comportamiento de un convertidor tipo Chopper clase D en modo corriente de 2 cuadrantes puede describirse por la corriente de salida I_o que se denota en la forma:

$$I_o = \frac{V_s(2\delta - 1)}{R}$$

Donde V_s denota el voltaje de entrada, δ el ciclo de trabajo de una señal pulsada y, R la resistencia de la carga.

Considerando I_{sp} la corriente de salida deseada, el problema de la regulación de corriente consiste en diseñar una ley de control u , tal que, el error en estado estacionario:

$$e = I_o - I_{sp}$$

Sea minimizado a una región de tolerancia Ω_e . Para lograr dicho objetivo se hacen las suposiciones al convertidor.

1. La carga R se considera constante, e invariante con el tiempo.
2. La modulación por anchos de pulso de la variable δ se hace de forma independiente para cada interruptor, y está limitada a valores de ciclo de trabajo en el intervalo descrito por $[DC1_{min}, DC1_{max}]$ y $[DC2_{min}, DC2_{max}]$.



1.3 Definición del problema

El análisis del estado del arte relacionado con los convertidores estáticos de energía, específicamente, los convertidores tipo chopper de dos cuadrantes en modo corriente, demuestra la necesidad de regular la corriente de salida. En este trabajo se propone el uso de máquinas de estado finito en combinación con controladores clásicos, para garantizar la estabilidad de la corriente de salida en términos de la modulación de las señales de disparo de los dispositivos de conmutación de estado sólido.

1.4 Objetivo de la investigación

1.4.1 Objetivo general

Diseñar un convertidor chopper clase D de dos cuadrantes en modo corriente con un controlador a lazo cerrado, tal que, partiendo de cualquier nivel de corriente de salida I_o sea posible alcanzar un valor de corriente I_{sp} en el menor tiempo posible, manteniendo el error de estado estable dentro de un valor de tolerancia en el intervalo Ω_e .

1.4.2 Objetivos específicos

1. Diseñar un convertidor chopper de dos cuadrantes en modo corriente
2. Simular el convertidor chopper de dos cuadrantes en modo corriente mediante MATLAB/Simulink
3. Diseñar un controlador para la regulación de corriente de salida en el estado transitorio basado en máquina de estados finito.
4. Diseñar un controlador digital clase PI para garantizar la estabilidad de la corriente de salida en el estado estacionario ($e < \Omega_e \|e\|$)
5. Pruebas de integración y estimación de la eficiencia del convertidor.



1.5 Preguntas de investigación

1. ¿El control basado en máquinas de estados finitos aumentará la eficiencia del convertidor estático de energía en modo corriente?
2. ¿El controlador automático puede desarrollarse e implementarse en un sistema embebido?
3. ¿Se mejorarán los tiempos de respuesta transitoria del convertidor con el controlador basado en máquinas de estado finito?

1.6 Justificación

La implementación de un convertidor estático de energía en modo corriente del tipo *chopper* requiere de un controlador de dos etapas empleando una máquina de estados finitos para mantener la corriente constante y estable. De esta manera, se obtendrá la forma natural de la señal cuasi rectangular de corriente de salida, un parámetro de comportamiento propio del convertidor. Cabe mencionar que cada estado representa una condición del sistema, a partir del cual se pueden establecer acciones de control que modifiquen la respuesta de la corriente de salida del convertidor. Adicionalmente, en el estado próximo al error de tolerancia se podrá integrar un controlador digital Proporcional-Integral, el cual garantice la estabilidad de la corriente de salida, lo que permitirá que los controladores incrementen la eficiencia del convertidor.

1.7 Impacto

- **Tecnológico**

Desarrollo de un sistema de control utilizando una máquina de estado finito implementado en un sistema embebido, para establecer las condiciones de funcionamiento y el punto de control para regular el consumo de energía de un convertidor estático de energía en modo corriente.

- **Investigación**

Divulgación del conocimiento a través de la documentación e implementación de una máquina de estados en un convertidor estático de energía utilizando un sistema embebido.



1.8 Hipótesis

Mediante la implementación de un controlador de dos etapas, basado en una máquina de estados finitos y el controlador discreto tipo PI ajustarán los ciclos de trabajo de los semiconductores de conmutación del convertidor, para seguir el punto de operación de corriente de salida. Con esta premisa se lograrán una eficiencia eléctrica del convertidor arriba del 85%.

1.9 Cronograma de actividades

A continuación, se muestra el cronograma de actividades para el desarrollo de la investigación el cual se encuentra distribuido por bimestre, cada uno representado por los números arábigos, durante dos años que dura el proyecto de tesis.

Actividades	2023			2024						2025		
	1	2	3	1	2	3	4	5	6	1	2	3
Definir los antecedentes y formar el estado del arte.												
Planteamiento y definición del problema												
Planteamiento y definición de los objetivos												
Determinar las preguntas de investigación												
Definición de la justificación												
Diseño y desarrollo del prototipo de PCB para el convertidor chopper												
Verificación y puesta a punto del convertidor												
Generación de las señales de control para el convertidor estático de energía en el sistema embebido												
Identificación de las variables de control												



2. Estado del Arte

En esta sección se describen los trabajos realizados en el campo de los convertidores estáticos de energía, se pone énfasis en los convertidores tipo chopper de dos cuadrantes en modo corriente. Con este estudio se podrán establecer mejoras en este tipo convertidor, enfocándose principalmente en el control de la corriente en lazo cerrado.

2.2 Convertidores estáticos de energía

Los convertidores estáticos de energía (CEE, por sus siglas) cumplen un rol importante en diferentes áreas de aplicación, pues son utilizados en los sistemas de energía renovables, vehículos eléctricos y la generación de energía pulsada [10]. Su diversidad y versatilidad para convertir energía es tan amplia que en la actualidad existen diferentes clasificaciones de convertidores y topologías emergentes, las cuales permiten transformar la energía en dos formas diferentes como Corriente Alterna (CA) o Corriente Directa (CD), comenzando por: CA-CA, CA-CD, CD-CA o CD-CD tal y como se muestra en la figura 2.1.

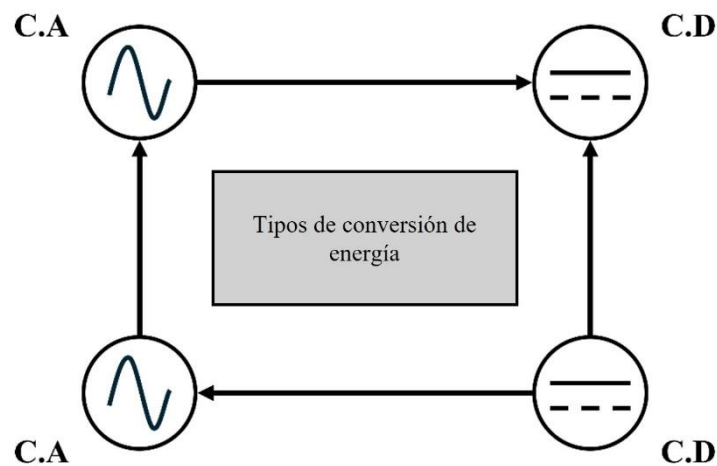


Figura 2. 1 Conversión de energía eléctrica realizada por diferentes topologías de CEE. Fuente: Elaboración propia



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

La conversión de energía eléctrica mediante diferentes topologías de CEE se hace controlando la energía a través de un sistema de potencia conformado por: a) dispositivos activos o semiconductores; b) circuitos pasivos como resistencias (R), capacitores (C) e inductores (L); c) circuito de mando; d) instrumentación electrónica e implementación del control automático. Los CEE del tipo CD-CD presentan tres modos de operación: a) modo lineal, b) modo de conmutación auxiliar y c) modo de conmutación forzada [11], este último, es el más utilizado en las diferentes topologías convertidores. En este modo de operación los CEE pueden operar con y sin aislamiento galvánico, todo dependerá del origen asociado de la energía, debido a que por lo general topologías basadas en medio puente y puente completo (*Half bridge* y *Full bridge*, por su definición en inglés) usualmente trabajan directamente con tensión CA rectificada mientras que otras topologías como *boost*, *buck-boost* suelen ir detrás de rectificadores eléctricos que ya aislaron y adaptaron la tensión. Específicamente, este documento está enfocado en el modo de conmutación forzada con aislamiento galvánico, en el cual las topologías están categorizadas con base a la cantidad de interruptores que lo conforman, por mencionar algunos ejemplos, los convertidores *buck*, *boost* y *buck-boost* pertenecientes a la categoría de un interruptor y los convertidores de tipo *push-pull*, *full bridge* y *half bridge* pertenecientes a la categoría de múltiples interruptores. La figura 2.2 muestra una categorización general de diferentes topologías de CEE, basada en M. Salem *et al* [11].

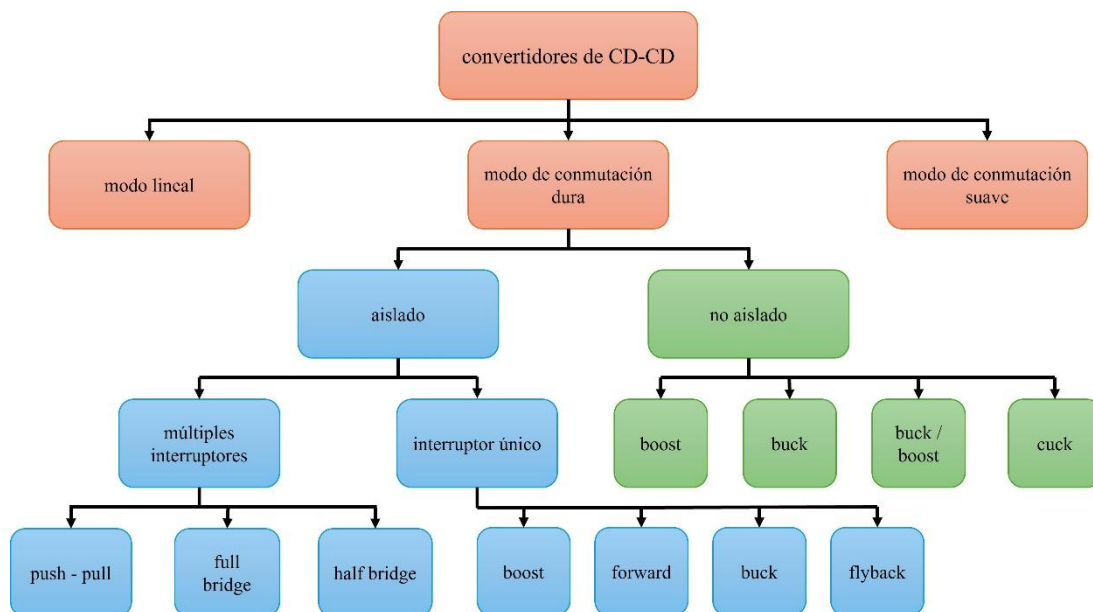


Figura 2. 2 Tipos de convertidores de CD-CD. Fuente: Elaboración propia a partir de M. Salem [3]



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

El convertidor utilizado en este trabajo es el *chopper* (troceador, por su traducción al español), clase D, de dos cuadrantes que pertenece a los de tipo aislados de múltiples interruptores, debido a que está conformado por dos dispositivos semiconductores de conmutación y dos diodos auxiliares [12]. Las características principales de este convertidor son la capacidad de mantener la corriente y el voltaje estable y puede operar en los cuadrantes I y IV, es decir, voltaje bipolar y corriente unipolar, o bien, solo en el cuadrante I o IV, tanto voltaje como corriente positivos o voltaje negativo y corriente positiva. En ambos cuadrantes; son solución sólida en aplicaciones para el manejo de la energía en celdas solares [13], el manejo de motores donde el consumo de corriente sobrepasa los límites recomendados por el fabricante al momento de arrancar el motor [14], amplificadores de sonido [15], o como sistema de carga para baterías de hidrógeno [16], entre otros. Los dispositivos de conmutación o semiconductores son el componente más importante del convertidor, ya que el manejo de corriente, voltaje y potencia dependerá de sus parámetros eléctricos de operación. Actualmente, los dispositivos de conmutación de estado sólido más utilizados son: a) MOSFET (*Metal-Oxide-Semiconductor Field-Effect Transistor*, por sus siglas en inglés), b) IGBT (*Isolated Gate Bipolar Transistor*, por sus siglas en inglés), c) BJT (*Bipolar Junction Transistor*), entre otros. La figura 2.3 muestra una clasificación de estos dispositivos y sus aplicaciones, a partir de los otros parámetros eléctricos de potencia y frecuencia.

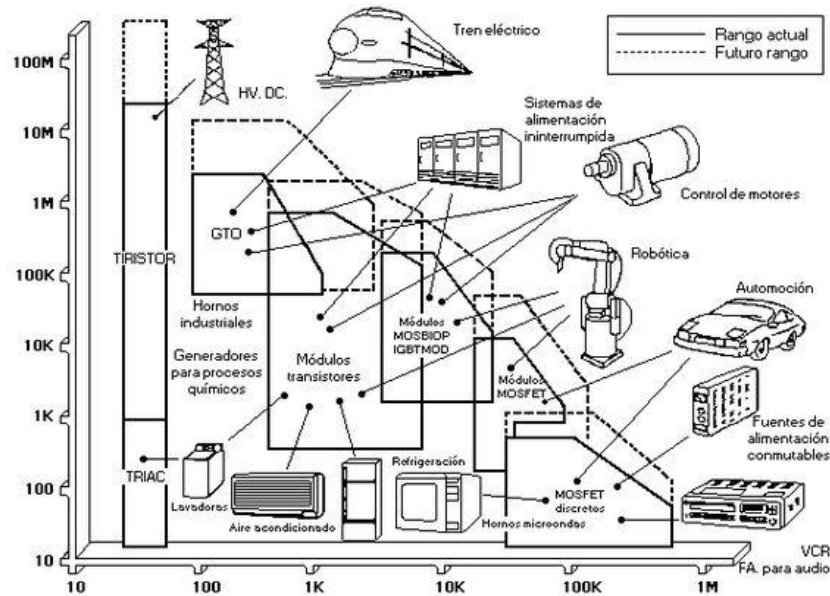


Figura 2. 3 Clasificación de dispositivos electrónicos de potencia a partir de sus parámetros eléctricos y aplicaciones. Fuente: M. H. Rashid [17]



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Los dispositivos de conmutación de estado sólido han tenido un gran desarrollo tecnológico, que ha venido evolucionando con los años, y actualmente su uso en la electrónica de potencia se está enfocando en convertidores estáticos de energía con dispositivos de conmutación de tercera generación como el de Nitruro de Galio o *Gallium Nitride* (GaN, por sus siglas en inglés) y Carburo de Silicio o *Silicon Carbide* (SiC, por sus siglas en inglés) [18].

Ambas tecnologías se utilizan en aplicaciones donde se requiere un sistema con una alta eficiencia eléctrica cercana al 93% a diferencia de los dispositivos convencionales de potencia basados solo en silicio (Si) los cuales se utilizan habitualmente en aplicaciones industriales donde se tiene una pérdida de energía y pérdida de potencia total significativamente menor cuando se manejan a altas frecuencias [19]. La implementación de este tipo de transistores con alguna de estas tecnologías representan un incremento mayor en los costos de desarrollo e implementación con respecto a los dispositivos convencionales basados en Si, debido a que es necesario el diseño específico de montajes profesionales basados en normas de construcción para la manufactura de la placa de circuito impreso (PCB, por sus siglas en inglés), además de componentes pasivos de última generación que se adecuen al prototipo, en este sentido son potencialmente una opción viable en el manejo y bajo consumo de potencia y energía debido a que se reducen las pérdidas eléctricas por conmutación derivado de que estos dispositivos ofrecen una conductividad térmica superior y una mayor eficiencia a altas frecuencias de conmutación donde las telecomunicaciones y las energías renovables son algunas de las aplicaciones donde se utilizan estos dispositivos [20].

Otro de los aspectos fundamentales en el desarrollo de un CEE es la disminución de ruido eléctrico en circuitos de conmutación, así como también la reducción del EMI (*Electromagnetic Interference*, por traducción al idioma inglés). En el caso de los CEE se debe principalmente al alto voltaje o corriente inducida a través de vías de conexión, y a la alta frecuencia de conmutación a la que operan los dispositivos semiconductores. Por otra parte, como los dispositivos semiconductores tienen propiedades no lineales, debido a su estructura intrínseca, este ruido intenso se expande a otros componentes a través de las vías de alimentación principales (potencia o tierra), produciendo corrientes que generan armónicos eléctricos derivados de la frecuencia de conmutación [21]. Una manera de realizar



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

la reducción del ruido EMI es mediante técnicas de conmutación aleatoria la cual es una técnica en la que los cambios de estado de los dispositivos de conmutación (encendido y apagado) no sigue un patrón periódico fijo, sino que varía de forma aleatoria dentro de límites establecidos previamente [22]. Otra de las técnicas de reducción de EMI por conmutación es por medio de redes de ayuda a la conmutación nombradas Redes Snubber, las cuales se pueden diseñar tanto para corriente di/dt como para voltaje dv/dt [23]. Estas redes están conformadas por circuitos RC, RL o RCD y son las más utilizadas para mejorar la conmutación de semiconductores que conforman los CEE, atenuando el ruido generado a partir de la activación del dispositivo semiconductor en el lado superior e inferior. Sin embargo, durante el proceso de desactivación, el diodo consume una corriente de recuperación inversa máxima y se desconecta abruptamente [24]. Lo anterior resulta en pérdidas por conmutación, requiriendo otro tipo de ayuda para la conmutación de tipo activa, por ejemplo, mediante un diseño asistido por el control de modulación por ancho de pulso (PWM) y tiempos de seguridad.

Como parte de un sistema de potencia, un controlador automático es indispensable; el controlador típico encontrado y que ha sido fundamental en la técnica de control es el Proporcional-Integral-Derivativo, o PID, por su acrónimo en español [25]. Este controlador es ampliamente utilizado en el sector industrial en aplicaciones como el control de velocidad, o en el contexto de los convertidores estáticos de energía controlar el modo de conmutación entre otras aplicaciones [26]. La mayoría de los sistemas basados en retroalimentación se controlan utilizando este controlador o bien sus diferentes variantes [27]. La figura 2.4 muestra un diagrama de bloques de la implementación del controlador PID a un dispositivo de salida el cual se denomina como “Planta” (sistema a controlar).

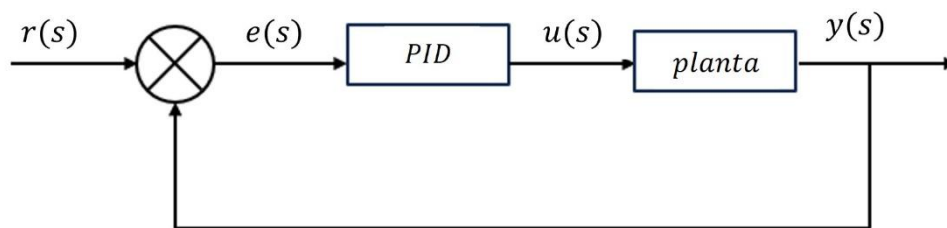


Figura 2. 4 Diagrama de bloques de controlador PID. Fuente: Ogata [28]



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Considerando la expresión matemática del control realimentado representado en el dominio de Laplace $u(s)$ queda [28]:

$$u(s) = e(s)\left(K_p + \frac{K_i}{s} + K_d s\right) \quad (2.1)$$

Donde:

$K_p =$ Constante Proporcional

$K_i =$ Constante Integral

$K_d =$ Constante Derivativa

$s =$ variable compleja

$u =$ salida del sistema

$e =$ error

Los sistemas de control implementados a convertidores estáticos de energía requieren el cálculo de las tres constantes inmersas en la ecuación (1), proporcional (K_p), integral (K_i) y derivativa (K_d). Existen diferentes representaciones de este controlador en distintos dominios para conocer y evaluar su comportamiento, por ejemplo, en el tiempo continuo $f(t)$, tiempo discreto $f(z)$ y frecuencia $f(\omega)$, entre los más estudiados en la literatura. Cabe mencionar que para el diseño del controlador PID, es posible utilizar los dominios de variables antes descritos. En el dominio del tiempo se representa mediante una ecuación diferencial e integral que describe la dinámica de la señal de control a partir del error entre la señal de referencia y la salida del sistema. Esta forma, conocida como representación continua o diferencial, se expresa como:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (2.2)$$

Donde:

$u(t) =$ Señal de control

$e(t) =$ Error

$e(\tau) =$ Error en un instante pasado

$K_p =$ Constante proporcional

$K_i =$ Constante integral

$K_d =$ Constante derivativa



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Donde el valor del error es calculado a partir de la diferencia entre la referencia $r(t)$ y la salida del sistema $y(t)$, dicha diferencia se ve reflejada mediante la siguiente ecuación:

$$e(t) = r(t) - y(t) \quad (2.3)$$

y el valor del error en un instante pasado considera el intervalo entre 0 y t, de la siguiente manera:

$$e(\tau) \text{ donde } \tau \in [0, t] \quad (2.4)$$

De forma que esta expresión matemática refleja directamente la acción de control en función del tiempo y permite una interpretación clara, donde la acción proporcional reacciona al valor instantáneo del error, por otra parte, la acción integral responde a su acumulación en el tiempo y la acción derivativa anticipa la evolución futura del error a partir de su velocidad de cambio [29].

Por otra parte, es necesario mencionar la existencia de dos estados presentes en la representación y análisis de sistemas de control el estado transitorio y el estado estacionario, donde el estado transitorio es la fase inicial de la respuesta del sistema ante un cambio en el valor de referencia (perturbación), en el controlador PID, las ganancias K_p , K_i y K_d tienen un efecto directo en la forma de la señal de control y la reducción de la oscilaciones tal como se mencionó anteriormente por lo cual, el estimar cada una de estas variables correctamente permite alcanzar una respuesta rápida y estable [30], en el caso del estado estacionario es la fase donde el sistema alcanzó un valor sostenido, sin variaciones notorias en el tiempo. Cada uno de los estados se representan en la figura 2.5.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

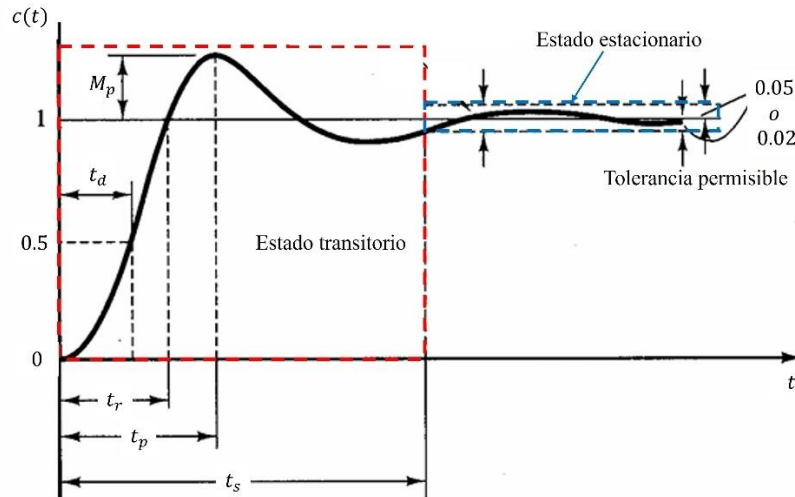


Figura 2. 5 Representación gráfica del estado estable y transitorio. Fuente: Elaboración propia a partir de Ogata [28]

Tal como se muestra en la figura 2.5 las definiciones para cada una de estas variables que conforman la respuesta transitoria del sistema fueron establecidas por Ogata [28] donde:

1. *Tiempo de retardo*, t_d , es el tiempo requerido para que la respuesta alcance por primera vez la mitad del valor final.
2. *Tiempo de levantamiento*, t_r , es el tiempo requerido para que la respuesta pase del 10 al 90% o del 0 al 100% del valor final.
3. *Tiempo pico*, t_p , es el tiempo necesario para que la respuesta alcance el primer pico del sobre impulso.
4. *Sobreimpulso máximo*, M_p , es el valor máximo de la curva de respuesta, medido a partir de la unidad. El porcentaje de sobreimpulso máximo, está definido por la siguiente ecuación:

$$\text{Porcentaje de sobreimpulso} = \frac{c(t_p) - c(\infty)}{c(\infty)} * 100 \quad (2.5)$$

Finalmente, el *tiempo de asentamiento*, t_s , es el tiempo que se requiere para que la señal alcance un rango de porcentaje alrededor del valor final (de 2 a 5%).

Otra representación del controlador PID es la forma discreta la cual se deriva del controlador continuo mediante técnicas de discretización, tal como menciona Bayat [31] esta representación es ampliamente utilizada en sistemas embebidos para el desarrollo del control digital debido a su simplicidad y efectividad. Esto permite calcular la señal de control a



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

intervalos regulares de tiempo, definidos por el tiempo de muestreo T_s . La expresión del controlador se muestra a continuación:

$$u[k] = u[k - 1] + a_0 e[k] + a_1 e[k - 1] + a_2 e[k - 2] \quad (2.6)$$

Donde:

$u[k]$ = salida del controlador en el instante de muestreo k

$e[k]$ = error entre la señal de referencia y la salida del sistema

a_0, a_1, a_2 = coeficientes de las ganancias

k = instante actual

$k - 1$ = instante anterior

$k - n$ = donde n es el numero de instantes anteriores

Cada uno de los valores para los coeficientes a_0, a_1, a_2 dependen de las ganancias K_p, K_i, K_d así como también del periodo de muestreo T_s . Es necesario mencionar que cada uno de los coeficientes no corresponden de forma particular a una sola ganancia, es decir, son el resultado de la combinación entre cada una ellas de acuerdo con el método de discretización empleado. El cual, en este caso se utiliza una técnica de discretización bilineal mediante método de Tustin de forma que se integran cada uno de los efectos proporcional, integral y derivativo [31].

Los procedimientos aplicados para el cálculo de cada una de las constantes del controlador se conocen como métodos de sintonización [28]. Uno de los métodos más utilizados es el método propuesto por Ziegler-Nichols, el cual está basado en la respuesta al escalón del sistema [28]. Otro método de tipo empírico es el control basado en lógica difusa que tienen una base de conocimiento útil compuesta por reglas que se obtienen a partir de la experiencia de operadores, observaciones experimentales (Jaen-Cuellar, 2013) [32]. Algunos estudios demuestran que el método propuesto por Ziegler-Nichols tiene la ventaja de fácil aplicación con reglas matemáticas simples [28] obteniendo así respuestas transitorias y estacionarias adecuadas así como también buenos márgenes de estabilidad [33], por lo que no provee resultados exactos con impulsos demasiados altos y un tiempo de estabilización alto [28].

Es importante mencionar que los sistemas de control basados en la respuesta al escalón del sistema aplicados en convertidores estáticos de energía se utilizan con la finalidad de reducir



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

el consumo de energía y mejorar su estabilidad reduciendo en gran medida el tiempo de establecimiento y reducir el sobre impulso en la señal de control [28]. No obstante, la mayoría de estas técnicas de sintonización requieren un puntual conocimiento de los sistemas de control, por lo tanto, la necesidad de encontrar un sistema de control alternativo a los métodos convencionales toma relevancia para disminuir los tiempos de desarrollo e implementación [33]. Una de las propuestas es emplear algoritmos de optimización inteligente o la implementación de métodos heurísticos tales como, lógica difusa, búsqueda tabú, optimización por enjambre u colmena, así como también algoritmos genéticos [34]. Estos métodos son una técnica que se fundamenta en la búsqueda de las mejores soluciones, lo anterior a costa de un procesamiento computacional razonable convirtiéndolos en una técnica viable y optima e incluso en muchos casos para establecer una solución factible concreta [35]. Algunas de las técnicas de control, así como también las metodologías de sintonización típicamente utilizadas se muestran en la figura 2.6.

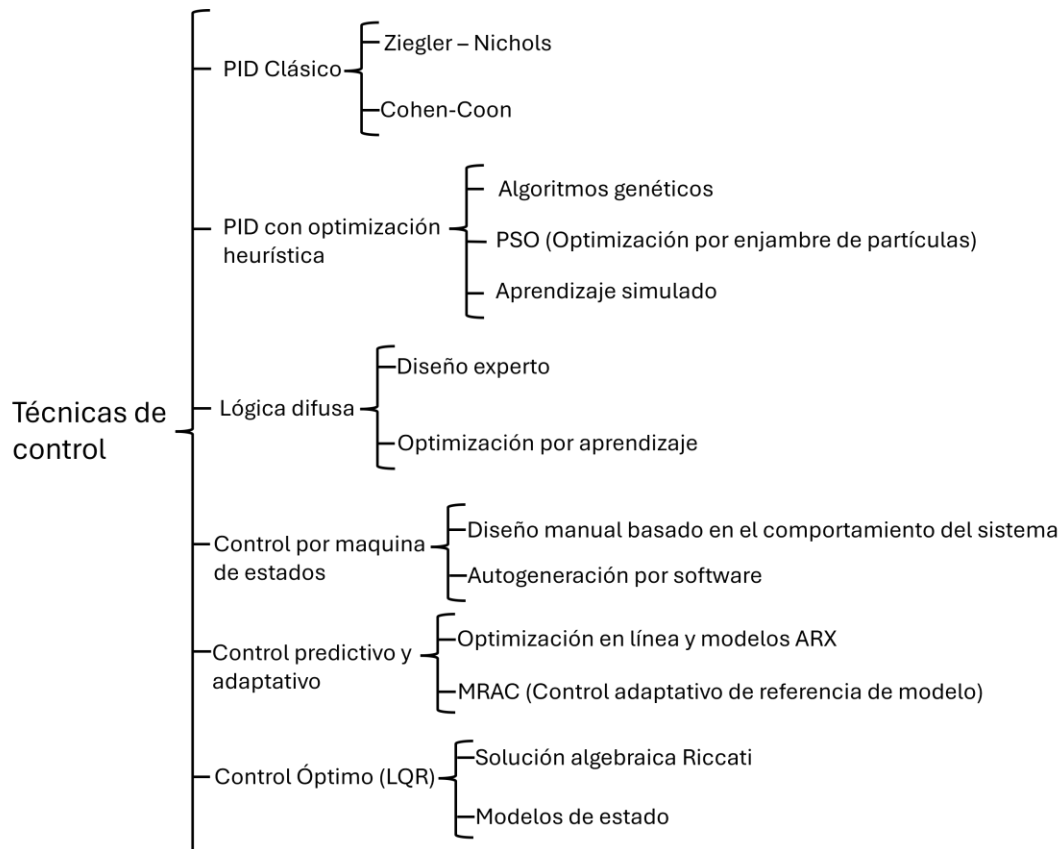


Figura 2. 6 Técnicas control utilizadas en sistemas eléctricos. Fuente: Elaboración propia a partir de Behrooz [36]



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Con referencia en la figura 2.6, se observa que la metodología para el desarrollo del controlador del convertidor *chopper* utilizando las máquinas de estado finito FSM (*Finite State Machine*, por su traducción al inglés) ha sido poco explorada, no obstante, ha sido implementada en la literatura para hacer controladores robustos los cuales mantienen un desempeño del sistema a pesar de las incertidumbres en los parámetros del modelo y el modelo matemático en caso de ser incompleto o inexacto, sin embargo, ofrecen ventajas como la estabilidad y un desempeño aceptable permitiendo un margen de ajuste en las ganancias, aunque también algunas desventajas como su difícil implementación [37]. Las FSM son una herramienta útil en el desarrollo de modelos de comportamiento de un número finito de estados mostrando la transición entre estos estados y acciones [37]. Así, las máquinas de estado finito se dividen en dos categorías principales Mealy y Moore. En primer lugar, en la Máquina de estado finito de Mealy, la salida está basada en el estado y entrada actual, en el caso del desarrollo de máquinas de estado de Moore la salida depende del estado actual únicamente por otra parte el cambio de valor en la salida en el diagrama de Mealy pueda cambiar inmediatamente con la señal de entrada en cambio con el diagrama de Moore este solo cambia con el estado actual de manera que el diagrama de Mealy es el adecuado para el diseño de máquinas de estado finitos dado que requiere menos estados en comparación con el diagrama de Moore [38]. Sin embargo, es necesario mencionar que una máquina de estados correctamente diseñada debe cumplir con los puntos que se muestran a continuación:

1. La suma de las condiciones de salida de cada estado debe ser igual a uno.
2. Ninguna combinación de las señales de entrada se puede repetir en dos o más condiciones de salida de un estado [38].

A partir de las reglas propuestas por Ballesteros [38]. Se muestra el ejemplo de modelo de máquina de estados finitos (Ver figura 2.7).



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

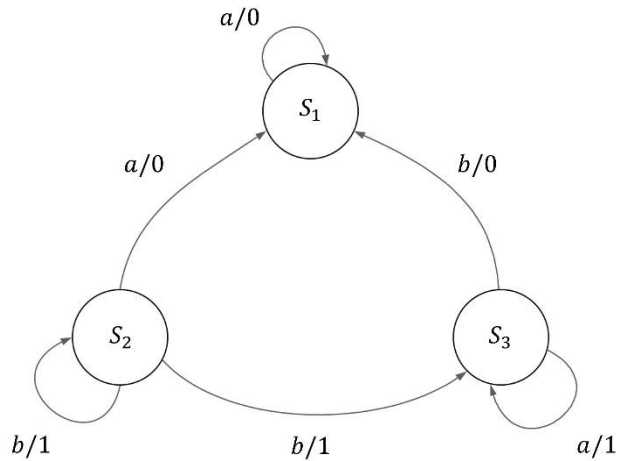


Figura 2. 7 Ejemplo de un grafo de una máquina de estados finitos y sus transiciones. Fuente: Elaboración propia.

Considerando la figura 2.7 la máquina de estados basada en el diagrama de Mealy está determinada por el estado actual y la entrada, permitiendo respuestas inmediatas a los cambios en la entrada. En la figura 2.7 estos estados (S_1, S_2, S_3) y transiciones están etiquetadas como entrada y salida ($a/0, b/0$). En la figura el modelo es efectivo para el desarrollo de circuitos síncronos debido a su menor latencia en comparación con las máquinas de tipo Moore [39].

Para la implementación de la máquina de estados el controlador debe cumplir con ciertas características:

1. Contar con módulos PWM y una frecuencia de operación mayor o igual a 1Mhz para tener un mayor margen de frecuencia en la configuración de los pulsos.
2. Contar con la capacidad de realizar *Multitasking* (Multitarea, por su traducción al idioma inglés), con la finalidad de realizar los cambios en tiempo real y evaluar el valor de las variables de control.

Algunos de los sistemas que cumplen con estas características se muestran en la figura 2.8.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

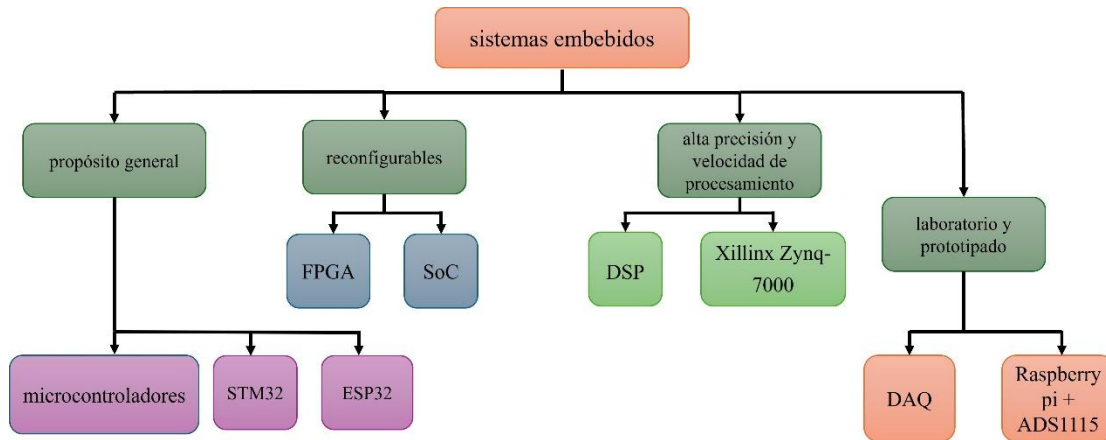


Figura 2. 8 Clasificación de sistemas embebidos. Fuente: Elaboración propia a partir de Zhou [40].

Para ejecutar los recursos requeridos por el controlador, actualmente se emplean diferentes arquitecturas de procesamiento, tales como FPGAs, microcontroladores, sistemas tipo SoC (*System on Chip*, por su traducción al inglés) o microcomputadoras presentes en marcas comerciales como Raspberry pi o Arduino Portenta. En el caso de los microcontroladores, son dispositivos que abaratan los costos de desarrollo e implementación debido a su resistencia física en su estructura y fiabilidad operativa, así como también el enfoque al que están orientados el cual es controlar un solo proceso además de contar con diferentes resoluciones del convertidor analógico digital (ADC) y disminuir el consumo de energía [41]. Autores como Rafi Al Tahtawi [42] afirma que incluso el uso de microcontroladores tales como el ATmega328 tienen una alta compatibilidad con sensores y otros *drivers* para el manejo de motores de CD o motores a paso entre otros. El diseño e implementación del control automático depende de los requerimientos del sistema donde se va a implementar, tal como menciona Peter McGrath [43] donde se utilizó un FPGA para un convertidor multicelular debido a la metodología de control está basada en métodos heurísticos. La razón de la implementación de este controlador se debe a la rapidez de procesamiento de datos y a la flexibilidad de programación, permitiendo realizar operaciones paralelas para modular las señales del tipo PWM.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Una alternativa para el desarrollo e implementación de la máquina de estado para realizar la acción de control sobre el convertidor chopper es un microcontrolador ESP32 debido a su facilidad de configuración y programación además de los costos de desarrollo, siendo un microcontrolador totalmente asequible. El autor Kumar en [44] menciona una de las ventajas de utilizar este microcontrolador es la posibilidad de agregar funcionalidades para el internet de las cosas (*Internet of Things*, por su definición original en inglés). Lo anterior propone al microcontrolador como una solución sólida para aplicaciones como la implementación de un sistema de monitoreo para un motor de inducción a través de una aplicación móvil, donde mediante sensores y el algoritmo adecuado el microcontrolador es capaz de enviar los datos en tiempo real del consumo y estado del motor.

Sin embargo, la alternativa propuesta pierde relevancia debido a las limitaciones del hardware tales como los voltajes de alimentación y de operación del ESP32 (a saber 3.3 V) considerando que los sensores tienen un voltaje nominal de 5 V surge la necesidad de implementación de otros sistemas como divisores de voltaje para la adaptación de la señal como cambiadores de voltaje, entre otros, lo que implica la generación de ruido y el desfase de la señal. Estas son algunas de las desventajas del sistema ESP32 que en ocasiones lo vuelve impráctico y no como una solución viable para el desarrollo e implementación del control.

Otros dispositivos como el microcontrolador de la serie 18 (PIC18F57Q10), de la marca Microchip®, los cuales son una propuesta sólida debido a que cubre la mayoría de los requerimientos en términos de versatilidad y rapidez de procesamiento enfocado hacia aplicaciones específicas, además de contar con la implementación de una nueva tecnología desarrollada por Microchip® nombrada *Core Independent Peripherals* (Periféricos Independientes del núcleo, por su traducción al idioma inglés) [45]. Esta tecnología permite que los periféricos del microcontrolador trabajen en paralelo sin afectar el programa principal por lo que en consecuencia el microcontrolador es más rápido en la ejecución de instrucciones de manera que representa la matriz principal en sistemas avanzados de control, esta tecnología surgió en respuesta a dar una solución efectiva al incremento constante de la complejidad de nuevos circuitos electrónicos ya que los microcontroladores cuentan con las



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

herramientas necesarias para ser utilizadas como un sistema integral [46]. Sin embargo, este tipo de tecnología es exclusiva de la familia de microcontroladores PIC18 de la marca Microchip®, el motivo se debe a que este tipo de sistemas son una herramienta flexible y adecuada para la investigación y prueba de circuitos por su eficiencia en sistemas relacionados a la electrónica de potencia [47].

Dentro de las aplicaciones de este tipo de microcontroladores con la tecnología mencionada anteriormente se encuentra la implementación desarrollada por Hranov [48] donde implemento este microcontrolador en un convertidor del tipo *buck* con una señal DC-DC para controlar y monitorear el voltaje de salida del sistema. Utilizando la ejecución en paralelo de los periféricos como ADC, PWM y el módulo de interrupciones, donde el objetivo principal es realizar las mediciones de voltaje y corriente de este tipo de convertidor por lo cual el ADC y el procesamiento de los valores de entrada son el principal objetivo y los periféricos anteriormente mencionados (PWM y el módulo de interrupciones) se ejecutan de forma paralela. Por otra parte, existen otras implementaciones en donde no está involucrada la electrónica de potencia, una de ellas fue desarrollada por Yong-Bin [49] el cual consiste en un sistema de comunicación entre una computadora y micro fuente de alimentación pulsada utilizando la comunicación asíncrona mediante el periférico UART (*Universal Asynchronous Receiver/Transmitter*, por su definición en inglés) utilizando herramientas integradas como la detección de errores utilizando interrupciones del microcontrolador.



Matriz de referencias

Referencia	Título	Metodología	Objetivos	Resultados
[10]	Descripción general de las topologías de inversores multinivel.	Análisis de características y técnicas de conmutación en inversores multinivel.	Describir las nuevas topologías de inversores y sus técnicas de conmutación basadas en la modulación por ancho de pulso (PWM).	Los resultados muestran que los inversores multinivel se mantienen en constante optimización y son ampliamente utilizados en aplicaciones de potencia media y alta.
[11]	Revisión del estado actual de los convertidores estáticos de energía para el desarrollo de un sistema eficiente.	Análisis de los convertidores multinivel a partir de su construcción, topologías y aplicaciones.	Mostrar las técnicas adecuadas para el desarrollo de un convertidor eficiente y económico que cumpla con los parámetros requeridos para la aplicación.	Los convertidores de tipo CD-CD resonantes bidireccionales y los inversores multinivel tienen una gran demanda y son adecuados en diversas aplicaciones.
[12]	Desarrollo de una interfaz electrónica de potencia basada en un convertidor chopper clase D.	Se utiliza un convertidor chopper con un controlador basado en <i>“Perturb and Observe”</i> implementado a un sistema fotovoltaico en	Mostrar el funcionamiento de una técnica de control para aprovechar mejor la potencia máxima que se	Los resultados de simulación demuestran que la eficiencia de la interfaz electrónica de potencia propuesta es elevada, sin embargo, se revela un deterioro de la



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

		un software de simulación.	puede obtener de un sistema fotovoltaico.	calidad de la forma de onda de la corriente extraída.
[13]	Estudio y análisis de un convertidor chopper de alta ganancia no aislado implementado en un sistema fotovoltaico.	Se realizan simulaciones del sistema utilizando el software MATLAB/Simulink utilizando la librería PSIM.	Analizar la eficiencia de un convertidor chopper en combinación con un algoritmo de seguimiento del punto de máxima potencia (MPPT).	Los resultados de la simulación muestran que el convertidor y el controlador propuesto realiza mejoras significativas en la eficiencia en comparación con otras topologías de última generación.
[14]	Diseño y simulación de un control de velocidad para un motor CC.	El sistema es simulado en MATLAB/Simulink utilizando un controlador PI y un convertidor chopper como regulador de voltaje.	Modelar y simular un sistema de control de velocidad eficiente aplicado a un motor de corriente continua de excitación separada.	Los resultados muestran que el control propuesto regula eficazmente la velocidad del motor validando así la viabilidad de la implementación del convertidor chopper y el controlador PI.
[15]	Desarrollo de un amplificador clase D con entrada digital (CDA)	Se analiza la intermodulación entre el corte, la modulación por	Diseñar un amplificador de clase D con entrada digital que logre un alto	Se demostró que la arquitectura desarrollada



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

		ancho de pulso (PWM) y la frecuencia de muestreo del DAC.	rango dinámico y una distorsión baja.	elimina eficazmente el ruido del módulo DAC.
[16]	Desarrollo de un sistema de acondicionamiento de energía.	Utilizando un diseño basado en 12 unidades <i>chopper</i> que funcionan en paralelo, considerando los factores clave como la eficiencia energética, la fiabilidad y la extensibilidad.	Implementar un sistema de acondicionamiento de energía para aplicaciones relacionadas en la carga de baterías de hidrogeno.	Las pruebas realizadas han demostrado un excelente rendimiento cercano al 97% tal como estaba previsto durante el proceso de diseño.
[17]	Libro basado en análisis de circuitos de potencia y sus aplicaciones.	Análisis matemático de cada uno de los circuitos eléctricos de potencia y componentes electrónicos.	Revisar cada una de las topologías eléctricas de potencia y sus aplicaciones.	Cada componente electrónico tiene un área específica de aplicación, demostrando sus debilidades y fortalezas.
[18]	Revisión del estado actual de los semiconductores.	Análisis de estructura de los componentes electrónicos semiconductores.	Desarrollar y ofrecer información útil sobre tecnología en semiconductores para la mejora en la formación de ingenieros en aplicaciones de electrónica de potencia.	Creación y adaptación y difusión de materiales educativos actualizados.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

[19]	Comparación de diferentes tecnologías de semiconductores.	Utilizando la topología de un convertidor Buck con diferentes tecnologías de semiconductores para el análisis de comportamiento y evaluación de las pérdidas de energía por conmutación.	Comparar el rendimiento de un convertidor Buck no asilado con diferentes dispositivos de potencia para identificar el dispositivo con mayor eficiencia y menor pérdida de energía.	Los resultados muestran que la combinación de dispositivos tipo JFET de SiC son la mejor opción para obtener un alto rendimiento en condiciones de funcionamiento adversas.
[20]	Revisión de la fiabilidad del rendimiento de dispositivos basados en SiC y GaN.	Análisis de literaturas relacionadas con dispositivos SiC y GaN que presentan problemas de fiabilidad como degradación térmica.	Comparar modelos predictivos de comportamiento y proponer estrategias que mejoren su viabilidad para aplicaciones de electrónica de potencia.	Los resultados muestran los retos y áreas de oportunidad para el desarrollo de tecnologías más fiables para aplicaciones recientes.
[21]	Revisión de los métodos para la reducción del ruido de conmutación en convertidores de corriente directa.	Utilizando un convertidor estático de energía de CD – CD de bajo voltaje, con técnicas de conmutación por PWM.	Maximizar la eficiencia eléctrica y reducir el ruido por conmutación del convertidor LDC.	El resultado obtenido indica una disminución efectiva del ruido en el sistema LDC, por lo cual reducir el ruido por conmutación es una tarea importante al momento de trabajar con convertidores.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

[22]	Desarrollo de un controlador para la reducción de ruido acústico en rectificadores de modo conmutado.	Utilizando una técnica de conmutación aleatoria realizando el modelado del sistema a partir de mediciones y un esquema CCPWM.	Desarrollar un controlador de bucle cerrado para la reducción eficaz de ruido acústico en un motor síncrono de imán permanente.	Las técnicas de conmutación aleatoria propuestas reducen eficazmente el ruido acústico generado por el inductor del rectificador.
[23]	Desarrollo de un amortiguador activo de conmutación suave.	Utilizando un interruptor auxiliar para descargar el condensador del amortiguador implementado en un convertidor <i>buck</i> basado en semiconductores IGBT.	Desarrollar un sistema de conmutación suave para reducir pérdidas de apagado de los IGBT debido al arrastre de corriente.	El uso del sistema propuesto permite recuperar parte de la energía almacenada en el condensador del amortiguador durante la desconexión.
[24]	Desarrollo de una red “ <i>Snubber</i> ” para la reducción de ruido en circuitos electrónicos.	Utilizando circuitos de conmutación conformados por dos semiconductores tipo FET de conmutación superior e inferior.	Reducir los picos de corriente a causa de la conmutación a altas frecuencias de los dispositivos semiconductores.	Los resultados indican que la red <i>Snubber</i> desarrollada disminuye considerablemente los picos de corriente, sin embargo, es necesario disminuir las inductancias parasitas en el circuito propuesto.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

[25]	Revisión del estado actual de controladores realimentados.	Análisis general de las características que definen cada uno de los controladores actuales.	Mostrar y ofrecer los controladores disponibles para cada aplicación en particular.	Divulgación de información de fácil acceso para el desarrollo de nuevas habilidades.
[26]	Diseño óptimo de un controlador PID fraccionario (FOPID).	Utilizando un convertidor de tipo Buck mediante un algoritmo de optimización por inteligencia de cohort.	Desarrollar y optimizar un controlador FOPID (Controlador PID fraccionario) para un convertidor <i>buck</i> . Para mejorar el rendimiento del sistema y reducir el tiempo de cálculo.	El controlador FOPID con la técnica de optimización propuesta ofrece un mejor desempeño dinámico y transitorio en comparación con un controlador PID convencional.
[27]	Desarrollo e implementación de un controlador PID optimizado por un algoritmo genético.	Utilizando un convertidor <i>boost</i> y la ecuación matemática que define el sistema.	Implementar un controlador PID optimizado que sea aplicable en cualquier otro tipo de convertidores.	Los resultados indican que el controlador PID optimizado por el algoritmo genético (QBGA) es aplicable a diferentes convertidores como el <i>buck</i> , <i>boost</i> & <i>buck</i> por mencionar algunos.
[28]	Literatura enfocada a la ingeniería de control moderna.	Análisis y desarrollo matemático de modelos de control, análisis de	Ofrecer una comprensión profunda de los fundamentos y	La literatura muestra una base sólida para el desarrollo, análisis y



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

		estabilidad y diseño de controladores.	técnicas de control automático.	modelado de sistemas de control.
[29]	Desarrollo de un controlador PID H^∞ .	Utilizando la teoría de control óptimo analizando sus propiedades y el tiempo de respuesta.	Implementar un controlador PID óptimo comparándolo con otros controladores subóptimos como el PID H2 y Maclaurin.	Se demuestra que los controladores proporcionan respuestas cuantitativas en el dominio del tiempo y la frecuencia.
[30]	Mejora en la comprensión de los comportamientos dinámicos.	Se utilizan soluciones analíticas de modelos ilustrativos para analizar estos comportamientos.	Contribuir en la comprensión intuitiva de los comportamientos en estado estacionario y transitorio.	Los resultados indican que incluso en modelos simples la dinámica de los sistemas puede ser compleja y difícil de entender de forma intuitiva, especialmente en lo que respecta al comportamiento transitorio.
[31]	Desarrollo de un controlador fraccionario FOPID en tiempo discreto.	Utilizando un controlador PID fraccionario basado en el modelo ARMAX.	Analizar el comportamiento de un controlador PID de orden fraccionario en tiempo discreto exponiendo sus	El controlador FOPID en tiempo discreto tiene la capacidad de resolver problemas de control complicados.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

			propiedades y aplicaciones.	
[32]	Desarrollo y simulación de un controlador PID sintonizado por el método del margen de ganancia fase con algoritmos genéticos.	Mediante un controlador PID sintonizado a partir de un algoritmo genético basado en el método del margen de ganancia-fase.	Desarrollar una nueva metodología de ajuste para controladores PID con el fin de mejorar el rendimiento en sistemas de control de posicionamiento.	Los resultados obtenidos indican que la optimización del controlador PID con la técnica propuesta mejora la respuesta del sistema y su estabilidad.
[33]	Diseño de un controlador PID robusto	Utilizando un convertidor <i>Buck-Boost</i> como sistema para su implementación y un algoritmo de optimización de enjambre de partículas.	Diseñar un controlador PID robusto con ganancias fijas que asegure un desempeño estable ante variaciones de sus parámetros	El PID robusto presenta mejorar en el rendimiento en comparación con los controladores PID convencionales, lo que lo convierte en una solución viable para los convertidores de potencia
[34]	Literatura control inteligente, Aplicaciones de la lógica difusa.	Desarrollo de conceptos de lógica difusa, exponiendo el fundamento teórico y casos de aplicación.	Explicar los fundamentos de la lógica difusa y sus aplicaciones.	Propone herramientas en el diseño de sistemas de control no lineales.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

[35]	implementación de un algoritmo genético para la sintonización de un controlador PID	Utilizando un controlador PID propuesto y el software de simulación y procesamiento de datos MATLAB/Simulink.	Determinar los valores óptimos de los parámetros del controlador PID, para mejorar el rendimiento del controlador.	Los algoritmos genéticos permiten encontrar los valores óptimos del controlador PID por lo que mejora su rendimiento de manera significativa.
[36]	Análisis de técnicas de control e implementación de mapas cognitivos difusos.	Utilizando un sistema de aire acondicionado y refrigeración (HVAC & R) aplicando una nueva técnica de control basada en el mapa cognitivo difuso (FCM).	Implementar y evaluar el método del mapa cognitivo difuso como método de control para los sistemas de calefacción y refrigeración.	El mapa cognitivo difuso es una solución eficaz para sistemas HVAC ya que mejora el rendimiento energético.
[37]	Revisión del estado actual de la implementación de métodos robustos en máquinas de estado finito (FSM)	Utilizando conceptos como observabilidad y controlabilidad e incertidumbre robusta en FSM	Exponer el concepto de control robusto en máquinas de estado finito (FSM), definiendo los principios de sistemas de estado continuo.	Los resultados obtenidos indican que es posible aplicar métodos de control robusto a máquinas de estado finito (FSM).
[38]	Desarrollo de una máquina de estado finito FSM para modular PWM en FPGA's.	Utilizando la herramienta CAE de Xilinx para generar de	Proponer una técnica de control basada en una máquina de estado finito para modular los ciclos	Para la aplicación en circuitos secuenciales la FSM se convierte en una solución viable para



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

		forma automática el código VHDL.	de trabajo de los pulsos PWM.	mejorar los tiempos de procesamiento.
[39]	Transformación de una máquina de estados de Mealy a Moore.	Utilizando la metodología por división de estados internos y la representación gráfica de la FSM.	Analizar y establecer un método para transformar una máquina de esta estados de Mealy a una de Moore evitando la dependencia entre entrada y salida.	La transformación utilizando el método propuesto aumenta el número de estados en un factor de 1,96 y transiciones en un factor de 2,05.
[40]	Revisión de estrategias avanzadas para el control PID.	Analizando los diferentes algoritmos de control PID y realizando una comparativa entre cada uno de estos.	Mostrar un panorama general de los controladores PID y las diferentes variantes aplicadas a procesos industriales.	Los resultados indican que los controladores PID con versiones mejoradas son utilizadas en el 90% de aplicaciones industriales.
[41]	Desarrollo de una técnica de conmutación para convertidores estáticos de energía.	Utilizando una máquina de estados (FSM) y un filtro pasa bajas aplicado a un convertidor estático de energía.	Proponer una técnica de conmutación eficaz para mejorar el rendimiento de los convertidores estáticos de energía.	La técnica propuesta facilita la implementación de microcontroladores de bajo coste y recursos a nivel de hardware mejorando así el rendimiento del convertidor.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

[42]	Desarrollo y aplicación de una FSM basada en un control PID.	Utilizando una máquina de estados finitos en conjunto con un control PID aplicado hacia un robot seguidor de línea.	Implementar una máquina de estados finitos en un robot seguidor de línea para mejorar el rendimiento del robot.	El algoritmo propuesto funciona correctamente y puede utilizarse como base para el control de robots seguidores de línea.
[43]	Desarrollo de una técnica de modulación por ancho de pulso (CSVPWM)	Utilizando estrategias convencionales de PD y modulación por ancho de pulso de vector espacial centrado (CSVPWM) aplicado a una máquina de estados finitos.	Implementar una estrategia de modulación mejorada para convertidores de condensador flotado y multicelulares.	Los resultados indican que se ha conservado las propiedades de equilibrio del convertidor logrando las ganancias armónicas esperadas.
[44]	Implementación de un sistema de monitoreo basado en IoT.	Utilizando un motor de inducción monofásico y tecnología basada en IoT.	Supervisar el comportamiento de un motor de inducción monofásico para atender de forma inmediata las averías que se presenten.	La aplicación de IoT permite una monitorización del sistema eficiente reduciendo la intervención manual.
[45]	Ficha técnica del sistema PIC18-Q10	Análisis de las características a nivel de	Mostrar los beneficios de la familia de microcontroladores	Presentar información de utilidad que permita al usuario comprender el



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

		hardware del sistema embebido PIC18.	PIC18 y áreas de oportunidad.	funcionamiento del sistema y determinar si es el adecuado para la aplicación en particular.
[46]	Literatura relacionada a los microcontroladores PIC18.	Exposición teórica con ejemplos resueltos, ejercicios, estudio de casos y material de apoyo.	Ofrecer información completa y practica sobre la programación con microcontroladores PIC18, incluyendo información de hardware y software.	Ofrece información sólida y aplicable en proyectos de electrónica, programación e interfaz de microcontroladores.
[47]	Implementación de tecnología basada en <i>Core Independent Peripherals</i> (CIP).	Utilizando un microcontrolador PIC18 con tecnología CIP mediante adaptación de código.	Implementar un sistema embebido utilizando la tecnología CIP para facilitar las técnicas de diseño y control.	La tecnología CIP es más rápida con una alta versatilidad por lo cual permitió el desarrollo de sistemas complejos como el RFID.
[48]	Desarrollo de una técnica e implementación de un microcontrolador como dispositivo de control.	Utilizando un microcontrolador, un potenciómetro digital y reductor LM2596 a través del protocolo SPI.	Presentar una técnica para controlar digitalmente el voltaje de salida de un convertidor de corriente directa utilizando un microcontrolador.	El sistema propuesto permite controlar de forma precisa el voltaje de salida del convertidor LM2596



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

[49]	Construcción de un circuito de comunicación.	Utilizando un microordenador de un solo chip C8051 y el protocolo de comunicación UART.	Desarrollar un circuito que permita la comunicación entre un ordenador y una fuente alimentación.	Se establece una comunicación estable entre el ordenador y la fuente de alimentación sin el uso excesivo de la CPU.
------	--	---	---	---



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Como conclusión de este capítulo se determina viable implementar una máquina de estados finitos como lo establecen la literatura en las referencias [39], [41-43] donde estas son implementadas en sistemas embebido basado en un microcontrolador con tecnología CIP (*Core Independent Peripherals*) para reducir el tiempo de procesamiento y optimizar la ejecución de cada una de las acciones que correspondan a los estados que conformen la máquina. Esta es el principal argumento por el que se optó implementar una MEF en lugar de un control inteligente. El sistema objetivo es un convertidor estático de energía de tipo chopper de dos cuadrantes en modo corriente utilizando dispositivos de conmutación de potencia como los MOSFET en conjunto con sistemas auxiliares como redes *Snubber* para la disminución de armónicos y la implementación de filtros digitales para mejorar la calidad de la señal de control que estará dirigida hacia el microcontrolador. Considerando las técnicas adecuadas para el diseño de circuitos eléctricos de potencia y la implementación de una aplicación para el monitoreo del sistema a partir del sistema embebido, así como también aplicando diferentes técnicas para el desarrollo de PCB basadas en las normas de diseño.



3. Metodología

En esta sección se presenta la metodología utilizada en este trabajo, toda vez que se han estudiado y analizado algunas en la literatura previa. Además, se adaptan algunos métodos retomados con el objetivo de hacer uso de sus ventajas. A continuación se describe el procedimiento detallado para el diseño e implementación del convertidor, el cual consiste en el desarrollo de: a) Características del convertidor, b) Simulación del convertidor en MATLAB/Simulink para ajustar los parámetros y dimensionar componentes, c) Simulación del sistema a lazo abierto para fijar el punto óptimo de operación del controlador mediante la máxima eficiencia del convertidor, d) Diseño de la máquina de estados, e) Pruebas de simulación del del convertidor en lazo cerrado, f) Desarrollo del circuito en físico, g) Diseño de la tarjeta PCB, h) implementación de los componentes electrónicos en la tarjeta, i) caracterización del convertidor, j) implementación de la máquina de estados finitos en el microcontrolador, k) Obtención de las señales de control.

3.1 Características del convertidor

Para realizar la simulación del convertidor *chopper* es necesario establecer los parámetros eléctricos y características para la operación adecuada del convertidor, estos pueden calcularse a partir de la literatura propuesta por Barry Williams en su documento titulado *Power Electronics* [1]. Su principal característica es la capacidad de cambiar la tensión fija a una tensión de salida variable sin almacenamiento intermedio de energía conformado por un inductor o capacitor. El nombre se atribuye a la forma de la tensión de salida similar a un tren de pulsos cuasi rectangulares cortados o troceados. Existen configuraciones diferentes en la carga para este tipo de convertidor, tales como, Resistencia-Capacitor (R-C), Inductor-Capacitor (L-C) o Resistencia-Inductor (R-L), siendo esta última configuración la implementada para el desarrollo de la simulación del convertidor.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Principalmente, el convertidor estará enfocado específicamente para trabajar en el primer cuadrante de operación, ya que su principal característica es que entregue voltaje y corriente unipolar. Los cuadrantes de operación del convertidor se ven definidos en la figura 3.1. De manera que el voltaje de salida V_o y la corriente de salida I_o son positivas y de esta forma se alimenta de manera adecuada la carga.

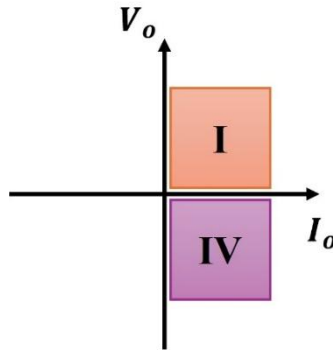


Figura 3. 1 Cuadrantes de operación del convertidor chopper en modo corriente. Fuente: Elaboración propia a partir de Williams [1]

Por otra parte, es necesario mencionar que este convertidor de corriente continua de cuadrantes (I y IV) funciona a partir de la secuencia de las señales PWM las cuales se muestran en la figura 3.2. De forma que el convertidor incorpora dos dispositivos semiconductores M1 y M2 utilizando las señales de conmutación PWM1 y PWM2 siendo esta última desfasada 180 grados, además de dos diodos complementarios D1 y D2.

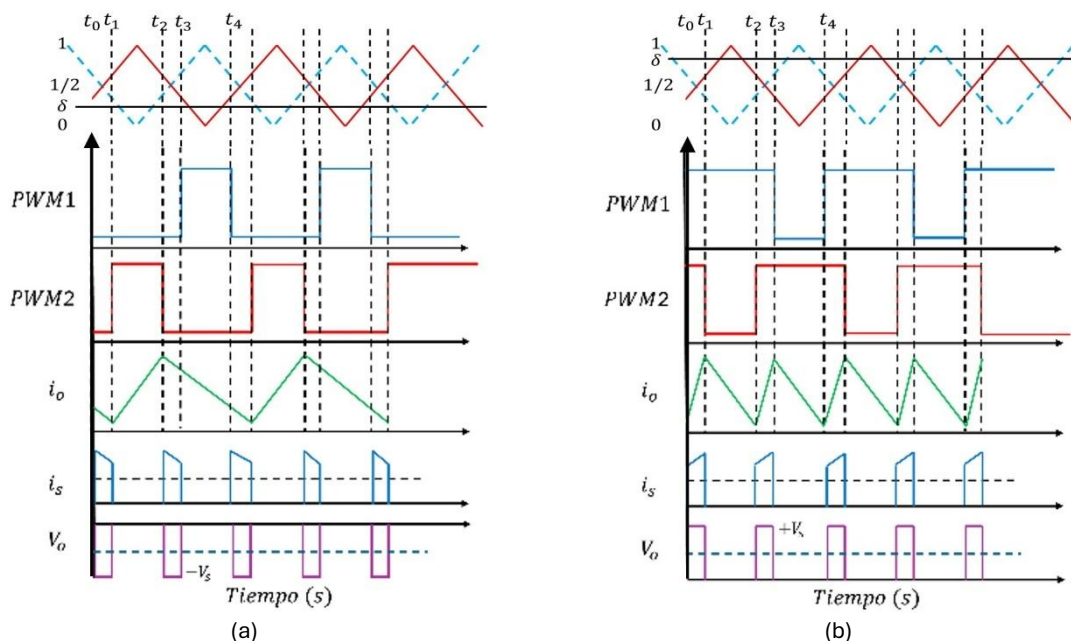
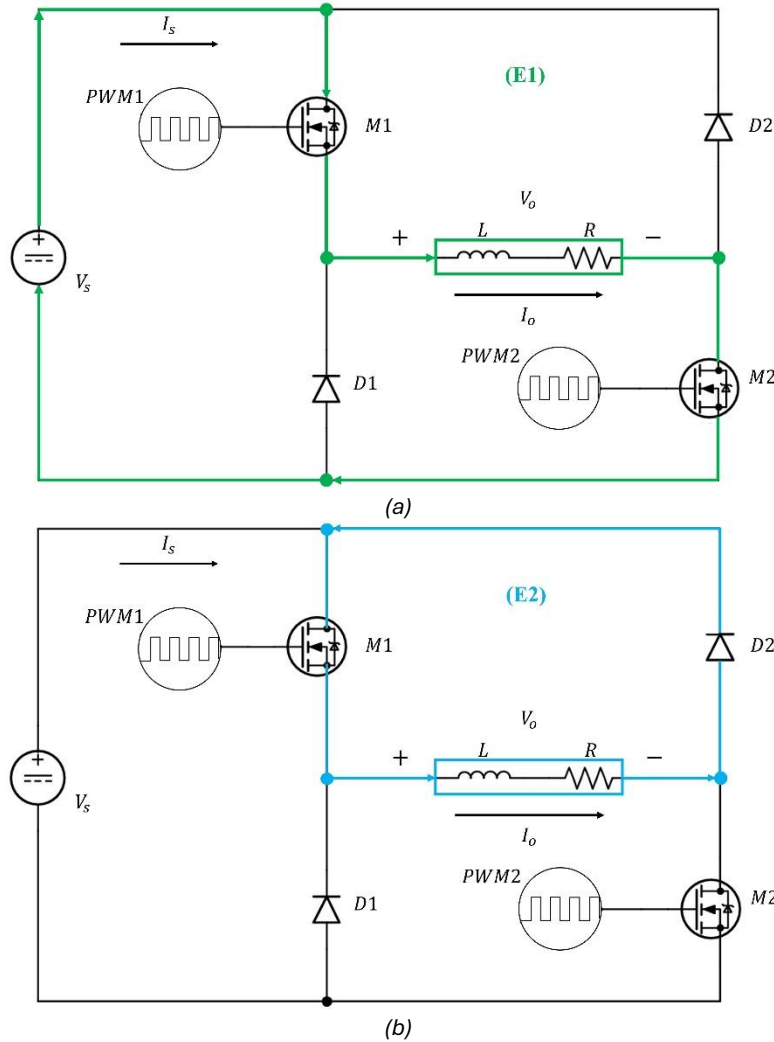


Figura 3. 2 (a) Cuadrante de operación IV, (b) Cuadrante de operación I, Fuente: Elaboración propia basado en [1]



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Las variables asociadas identificadas del convertidor son el voltaje de entrada V_s , corriente de entrada I_s , voltaje de salida V_o y corriente de salida I_o . De manera que el convertidor opera en tres modos los cuales están definidos por el comportamiento de V_o e I_o en la carga, tal como se observa en la figura 3.3. Las condiciones del ciclo de trabajo para M1 y M2, son $D > 0.5$ para el primer cuadrante y $D < 0.5$ para el cuarto cuadrante.





“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

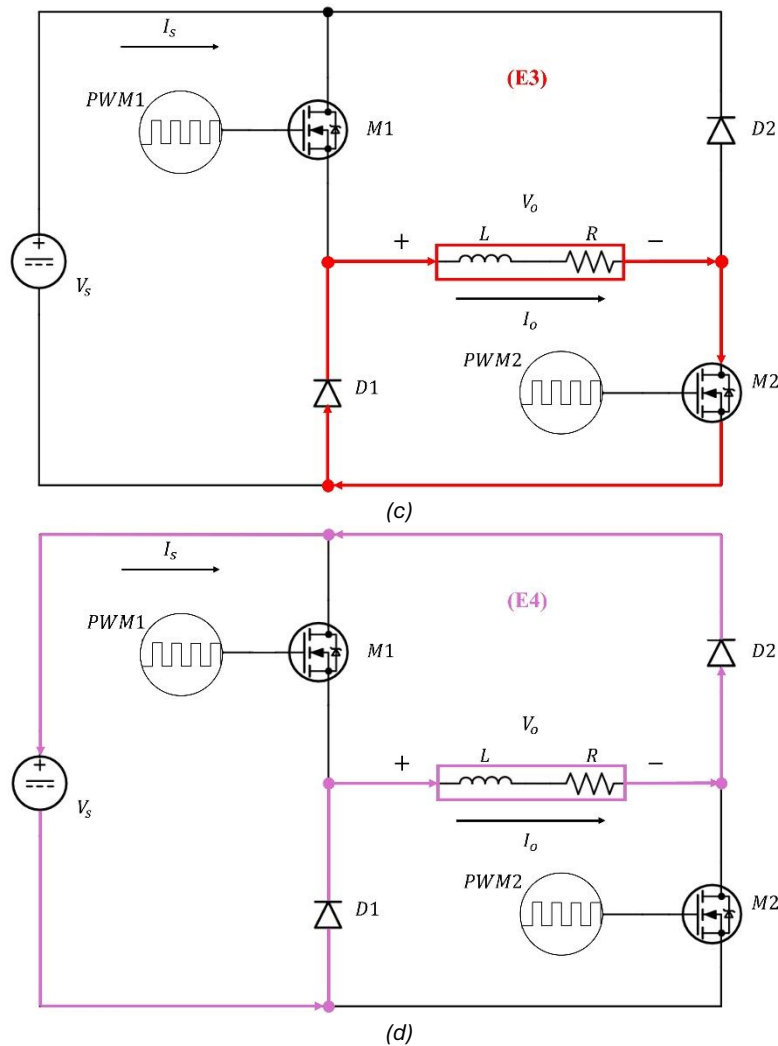


Figura 3. 3 Topología y estados de operación del convertidor. Fuente: Elaboración propia a partir de [1]

Cada uno de los incisos en la figura 3.3 están definidos por la nomenclatura (E1) perteneciente al primer estado de color verde, el segundo estado (E2) de color azul, el tercer estado (E3) de color rojo, finalmente el cuarto estado (E4) de color violeta, todos los estados ilustran el flujo de corriente a través del *chopper*, funcionando en el primer cuadrante. Donde la variable T es el periodo de conmutación y la I_o es siempre positiva y el valor de dicha variable está relacionada en la función de la carga, y el V_o en función del encendido y apagado de los MOSFET de conmutación (Ver Tabla 3.1).



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

$V_o =$	{	V_s	$t_0 < T < t_1$	$M_{1,2} \rightarrow ON$	(E1)
		0	$t_1 \leq T < t_2$	$M_1 \rightarrow ON; M_2 \rightarrow OFF$	(E2)
		V_s	$t_2 \leq T < t_3$	$M_{1,2} \rightarrow ON$	(E1)
		0	$t_3 \leq T < t_4$	$M_1 \rightarrow OFF; M_2 \rightarrow ON$	(E3)

Tabla 3. 1 Topología y estados de operación del convertidor en el cuadrante I. Fuente: Elaboración propia a partir de Williams [1]

En caso de que se requiera que el convertidor opere en el cuarto cuadrante, es decir, voltaje negativo y corriente positiva a la salida, los estados y el orden de operación de los MOSFET se muestran en la tabla 3.2.

$V_o =$	{	$-V_s$	$t_0 < T < t_1$	$M_1 \rightarrow OFF; M_2 \rightarrow OFF$	(E4)
		0	$t_1 \leq T < t_2$	$M_1 \rightarrow OFF; M_2 \rightarrow ON$	(E3)
		$-V_s$	$t_2 \leq T < t_3$	$M_1 \rightarrow OFF; M_2 \rightarrow OFF$	(E4)
		0	$t_3 \leq T < t_4$	$M_1 \rightarrow ON; M_2 \rightarrow OFF$	(E2)

Tabla 3. 2 Topología y estados de operación del convertidor en el cuadrante IV. Fuente: Elaboración propia a partir de Williams [1]

A partir de la figura 3.2.a se puede determinar que cuando el ciclo de trabajo (δ) es menor a 50% ambos interruptores nunca conducen energía de forma simultánea por lo que el voltaje de salida es de 0 o $-V_s$. Por lo que el voltaje de salida medio es independiente de la carga y para $0 \leq \delta \leq 0.5$ utilizando las formas de onda de la figura 3.2.a (PWM1 y PWM2). Las cuales vienen dadas por.

$$\bar{V}_o = \frac{1}{T} \int_{t_T}^T -V_s dt = \frac{-V_s}{T} (T - t_T) = -V_s \left(1 - \frac{t_T}{T}\right) \quad (3.1)$$

Así:

$$\text{cuando } \delta = 0 \rightarrow t_T = T \text{ y } V_o = -V_s$$

$$\text{cuando } \delta = 0.5 \rightarrow t_T = 0 \text{ y } V_o = 0$$

De manera que el portador (*Carrier*) $2T$, es el doble del periodo de conmutación T que tras sustituir t_T/T en la ecuación 3.1.

$$\bar{V}_o = -V_s \left(1 - \frac{t_T}{T}\right) \quad (3.2)$$

$$= -V_s(1 - 2\delta) = V_s(2\delta - 1) \text{ para } 0 \leq \delta \leq 0.5 \quad (3.3)$$



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Por lo cual el funcionamiento del convertidor en el cuarto cuadrante consiste en cortocircuitar primero el circuito de salida para aumentar la corriente y a continuación eliminar el corto circuito de salida, lo que obliga a la corriente volver a la fuente de alimentación de CD a partir de un diodo auxiliar. La operación del convertidor se describe en la ecuación 3.4. Tal como se muestra en la figura 3.2.b el convertidor se encuentra en el primer cuadrante, es decir al menos un interruptor está conduciendo, por lo que el voltaje de salida es $+V_s$ o 0. Para una corriente de carga continua, el voltaje de salida promedio es independiente de la carga. Por lo que viene dado por:

$$\bar{V}_o = \frac{1}{T} \int_0^{t_T} V_s dt = \frac{V_s}{T} t_T \quad (3.4)$$

Donde sustituyendo t_T/T en la ecuación 3.4 da.

$$\bar{V}_o = V_s \frac{t_T}{T} = V_s(2\delta - 1) \quad \text{para } 0.5 \leq \delta \leq 1 \quad (3.5)$$

Considerando E como una fuente voltaje constante y el voltaje de salida promedio es el mismo, la media de corriente de salida viene dada por la misma expresión.

$$\bar{I}_o = \frac{\bar{V}_o - E}{R} = \frac{V_s(2\delta - 1) - E}{R} \quad (3.6)$$

3.2 Simulación del convertidor chopper en lazo abierto

La simulación es un proceso importante para validar parámetros y ajustarlos, es parte de la metodología para encontrar el punto de operación del convertidor a partir de los componentes y parámetros. En este trabajo se llevó a cabo con MATLAB/Simulink versión v2024A con licencia de uso académico. Previo a agregar los componentes que conforman el convertidor, se realizaron ajustes previos para la ejecución de la simulación del circuito, como la asignación del solucionador Rosenbrok y el tiempo de simulación el cual es de 0.6 ms, tal como se muestra a continuación (Ver los recuadros en color rojo de la figura 3.4).



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

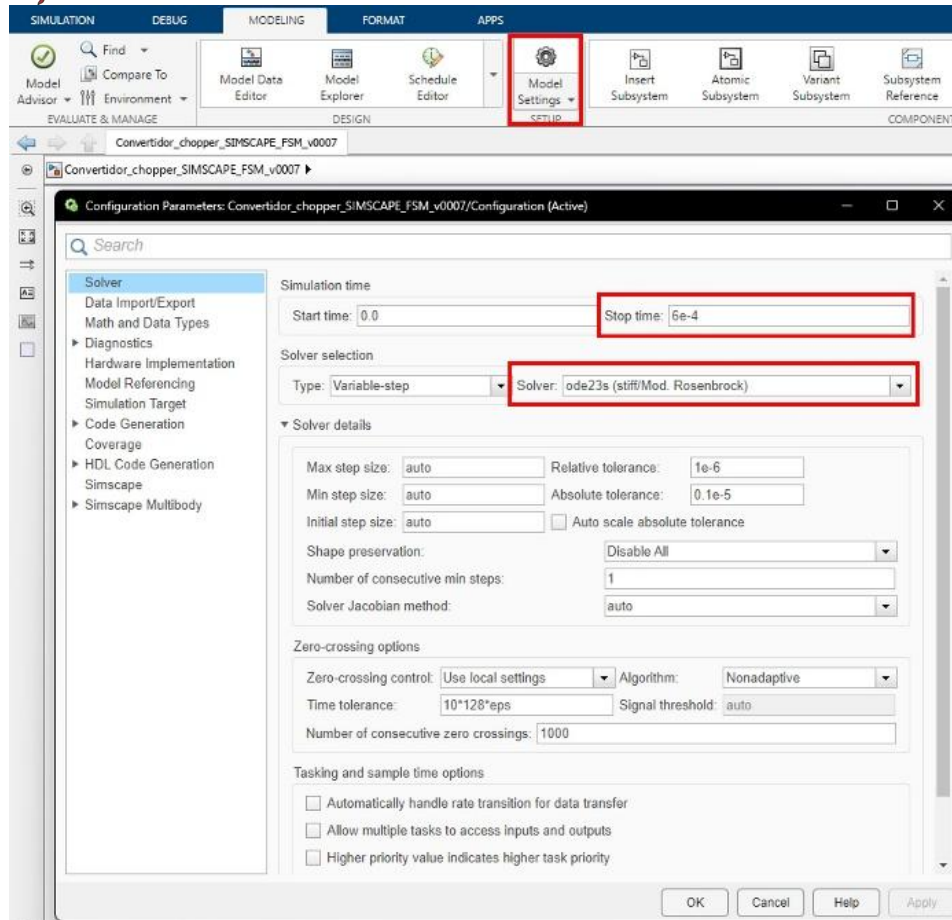


Figura 3. 4 Asignación de tiempo y definición de tiempo de simulación. Fuente: MATLAB/Simulink [50]

El motivo por el que se define el solucionador Rosenbrock es por su compatibilidad con los circuitos eléctricos simulados utilizando la librería MATLAB/ Simscape por otra parte la definición del tiempo de simulación de 0.6ms se debe a tener una mejor resolución y detalle en la captura de la señal de control [50].

Con base en lo anterior, el circuito de simulación se desarrolló con elementos de la librería Simscape en MATLAB/Simulink v2024A que tiene como característica asignar los valores físicos reales del componente en físico definidos por el fabricante [51] tal como se muestra en la figura 3.5. Los parámetros asignados a los dispositivos MOSFET son los siguientes: $R_{DS(on)} = 0.3 \Omega$, $I_{DS(on)} = 21 A$, $V_{GS}(TH) = 2.5 V$ y diodos con el parámetro $V_F = 4.5 V$. Posteriormente se utilizó una carga tipo R-L (Resistencia – Inductor) conectada en serie con valores 83.1Ω y $475 \mu H$ respectivamente y un capacitor conectado en paralelo a la resistencia



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

de $0.1\mu\text{F}$, aplicando $V_s=180\text{ V}$ para obtener una corriente de salida $I_o = 2\text{ A}$. Finalmente, la frecuencia de operación de las señales de conmutación PWM1 y PWM2 se fijó en 80 KHz [2].

POWER MOS V®

Power MOS V® is a new generation of high voltage N-Channel enhancement mode power MOSFETs. This new technology minimizes the JFET effect, increases packing density and reduces the on-resistance. Power MOS V® also achieves faster switching speeds through optimized gate layout.

- Faster Switching
- Lower Leakage
- 100% Avalanche Tested
- Popular TO-247 Package

APT6030BVR

600V 21A 0.300Ω

All Ratings: $T_c = 25^\circ\text{C}$ unless otherwise specified.

Symbol	Parameter	APT6030BVR	UNIT
V_{DS}	Drain-Source Voltage	600	Volts
I_D	Continuous Drain Current @ $T_c = 25^\circ\text{C}$	21	Amps
I_{DM}	Pulsed Drain Current ⁽¹⁾	84	Amps
V_{GS}	Gate-Source Voltage Continuous	±30	Volts
V_{GSM}	Gate-Source Voltage Transient	±40	Volts
P_D	Total Power Dissipation @ $T_c = 25^\circ\text{C}$	300	Watts
	Linear Derating Factor	2.4	W/°C
T_j, T_{STG}	Operating and Storage Junction Temperature Range	-55 to 150	°C
T_L	Lead Temperature: 0.063" from Case for 10 Sec.	300	°C
I_{AR}	Avalanche Current ⁽²⁾ (Repetitive and Non-Repetitive)	21	Amps
E_{AR}	Repetitive Avalanche Energy ⁽³⁾	30	mJ
E_{AS}	Single Pulse Avalanche Energy ⁽⁴⁾	1300	mJ

Block Parameters: M1

N-Channel MOSFET Auto Apply

NAME	VALUE
Modeling option	Threshold-based
Main	
Number of terminals	Three
Parameterization	Specify from a datasheet
Drain-source on resistance, $R_{DS(on)}$	0.300 Ohm
Drain current, I_{ds} , for $R_{DS(on)}$	21 A
Gate-source voltage, V_{gs} , for $R_{DS(on)}$	15 V
Gate-source threshold voltage, V_{th}	1.7 V
Channel modulation, L	0 1/V
Measurement temperature	25 degC
Ohmic Resistance	
Capacitance	
Body Diode	
Temperature Dependence	
Initial Targets	
Nominal Values	
Faults	

Figura 3. 5 (a) Primer hoja de datos del componente [51]. (b) Interfaz de asignación de valores del componente MOSFET en MATLAB/Simscape [52]

Tal como se muestra en la figura 3.5 se muestra la metodología de asignación de parámetros para cada uno de los valores que conforman el convertidor. Sin embargo, es necesario mencionar la implementación de diferentes subsistemas para organizar de forma correcta cada uno de estos elementos, la distribución final de los componentes que conforman el convertidor *chopper* se muestra en la figura 3.6.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

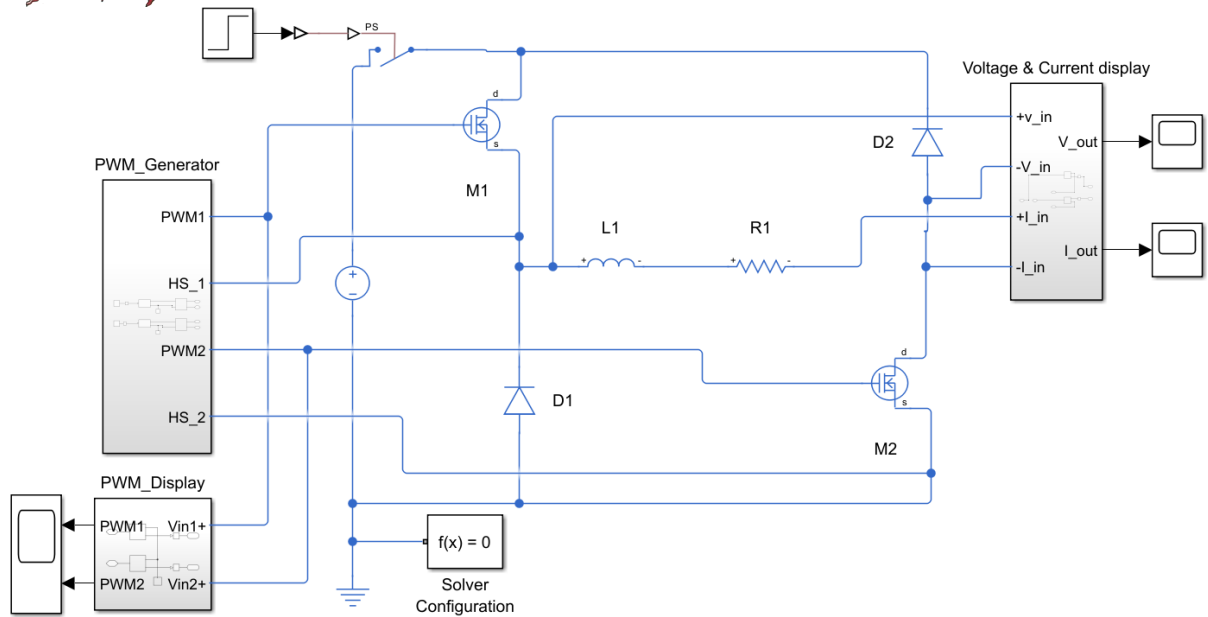


Figura 3. 6 Visualización y simulación del convertidor chopper desarrollada en Simscape de MATLAB/Simulink.
Fuente: Elaboración propia

Por otra parte, uno de los elementos fundamentales para completar el circuito de simulación son los pulsos de conmutación generados a partir del bloque PWM_Generator para producir las señales PWM1 y PWM2 desfasadas 180° una respecto a la otra y configuradas con un ciclo de trabajo ($D=60\%$). Debido a que con este valor se cae en la condición de operación en el cuadrante I. El bloque PWM_Generator está conformado por dos bloques generadores de pulsos donde se agregaron las características de la señal como la frecuencia de 80 kHz y en uno de los bloques se configuro el desfase de 180° para garantizar una correcta operación del chopper como se observa en la figura 3.1.b.

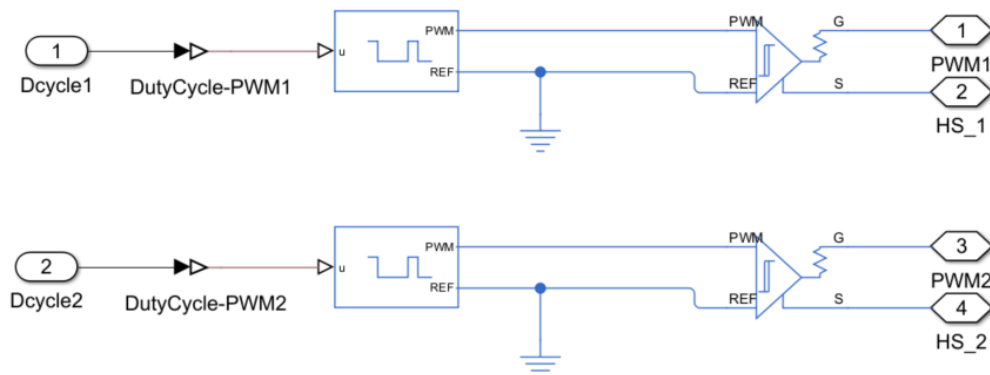


Figura 3. 7 Bloque PWM_Generator Fuente: Elaboración propia

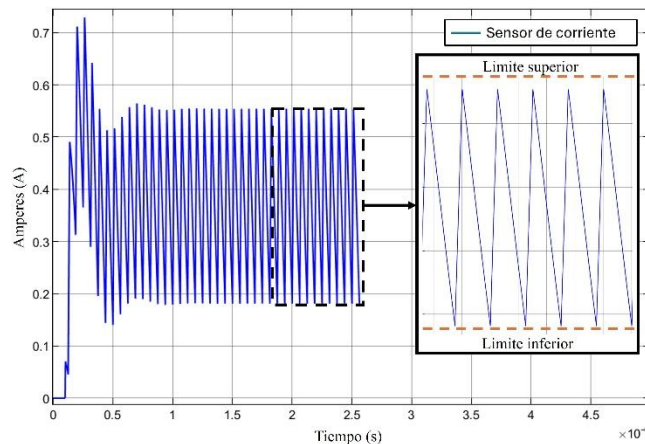


“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Finalmente se agregaron dos drivers con parámetros predeterminados con el objetivo de realizar el acoplamiento entre las señales eléctricas a las terminales de compuerta y fuente de cada MOSFET (Ver figura 3.7).

3.3 Diseño y simulación de la máquina de estados finitos en lazo cerrado

Para el desarrollo de un sistema de control es importante identificar cada una de las variables que conforman el control [28]. En este caso la variable de control son los ciclos de trabajo (*Duty Cycle*, por su traducción al inglés) de los pulsos de conmutación y la variable controlada es la corriente de salida del convertidor (I_o). Sin embargo, en la figura 4.2.b se puede apreciar que la corriente no es un valor constante sino oscilatorio, en este sentido se puede definir dos límites superior e inferior con la finalidad de que la máquina de estados no realice transiciones innecesarias entre estados en caso de que la corriente de salida se encuentre de estos límites. A este nuevo estado intermedio será el estado estacionario. Los límites están definidos por un porcentaje de $\pm 5\%$ del valor resultante del error. A continuación, se muestra la ubicación de cada uno de los límites así como también las operaciones pertinentes para la obtención de cada uno (Ver figura 3.8).



(a)



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

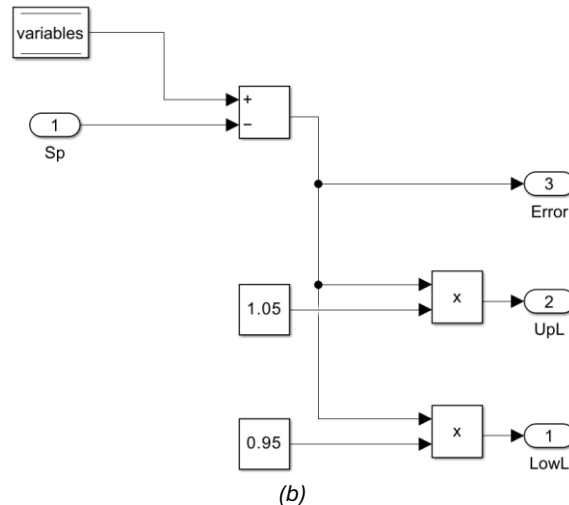
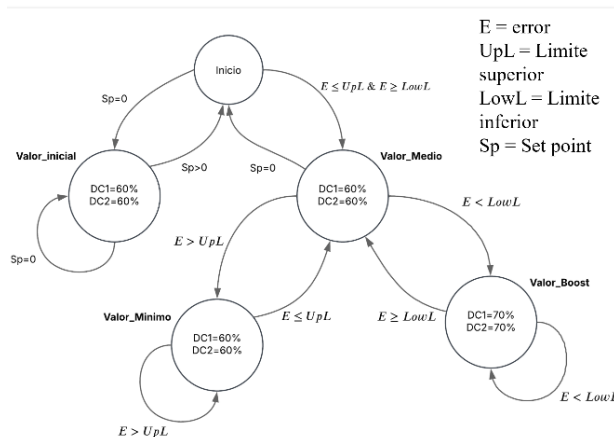


Figura 3. 8 a) Límites superior e inferior de la corriente de salida. b) Operaciones matemáticas para la obtención de límite superior (UpL) e inferior (LowL).

Por otra parte, en la figura 3. 9.a. se muestra el grafo resultante basado en el diseño de Mealy para la definición de los límites superior e inferior como condiciones de transición, así como también cada uno de los estados donde se le asignó un ciclo de trabajo determinado en un rango absoluto de 60% y 70% sin posibilidad de asignar un valor intermedio a su vez la máquina de estados adaptada al software de simulación *Stateflow* de MATLAB se muestra en la figura 3.9.b.

Tal como se puede apreciar en la figura 3.9.b, la máquina de estados implementada para la simulación del sistema es diferente al grafo realizado, esto es debido a que se realizaron optimizaciones de simulación para evitar recursividad en la simulación y crear una ventana de tiempo adecuada para que la máquina de estados realice los ajustes en los ciclos de trabajo en los pulsos de conmutación.





“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

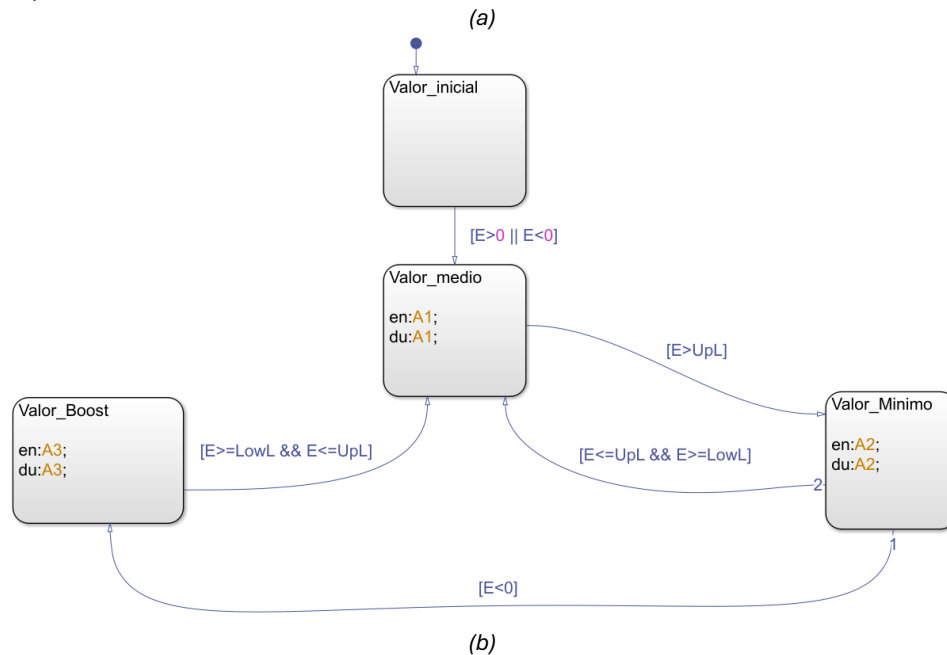


Figura 3. 9 a) Grafo de la máquina de estados basado en el diseño de Mealy. b) Máquina de estados adaptada al software de simulación Stateflow de MATLAB. Fuente: Elaboración Propia

A partir de la figura 3.10, se establece un valor objetivo con la variable “Set_point” este valor objetivo ingresa al bloque denominado como “Operaciones” donde se definen los umbrales o límites de la corriente “LowL” para el límite inferior y “UpL” para el límite superior así como también se establece el valor del error a partir de la variable “Error” del mismo modo las variables anteriormente mencionadas ingresan al bloque “Maquina de estados” donde se encuentra la máquina de estados previamente establecida, donde cada acción perteneciente a cada estado esta denominado por A como variable y reducción de la palabra acción. Por lo anterior, la función “A1()” corresponde al “Valor_medio” en la máquina de estados, la función “A2()” corresponde al “Valor_minimo” y finalmente la función “A3()” corresponde al “Valor_boost” definido en la máquina de estados, al ser llamadas estas funciones por la máquina de estados asignan un valor en el ciclo de trabajo de los MOSFET mediante el bloque “Acciones” donde solo se tienen dos variables de salida DC1 y DC2 correspondientes al ciclo de trabajo de los MOSFET M1 y M2. Finalmente, las variables medibles para la evaluación del convertidor con la máquina de estados son la corriente de salida (I_out), los pulsos de conmutación (PWM1 y PWM2), finalmente el valor objetivo (“Set_point”).



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

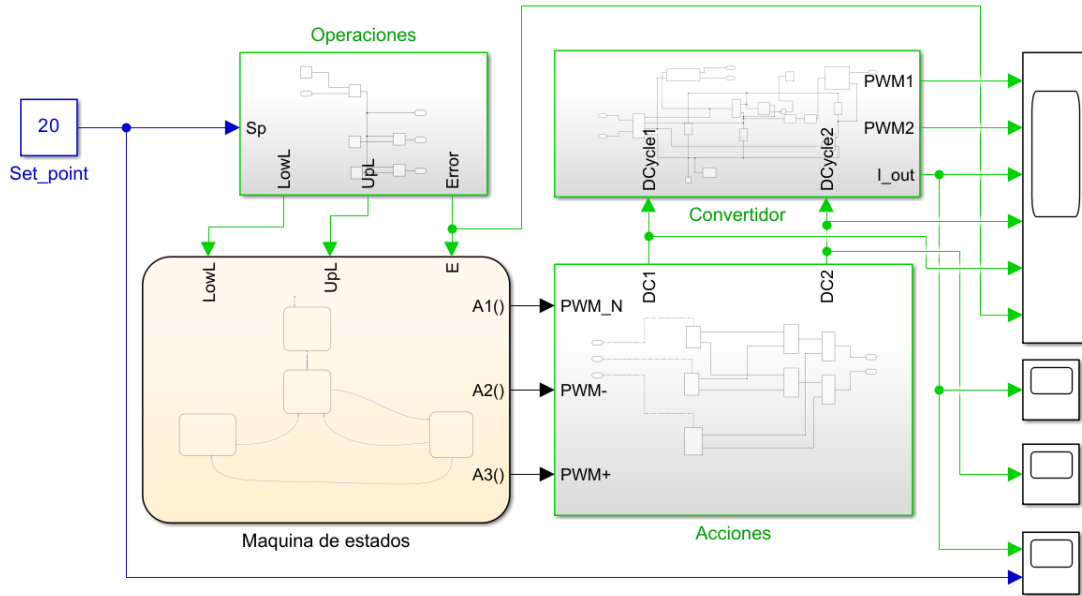


Figura 3. 10 Diagrama general de del sistema con la MEF. Fuente: Elaboración propia en MATLAB/Simulink

3.4 Optimización en la máquina de estados con el controlador PI

A partir de los resultados de simulación en lazo cerrado con la MEF en la sección 5.2.1, exactamente en la figura 5.3, se estableció optimizar el controlador mediante un control de tipo PI. Lo anterior porque era necesario asignar valores intermedios en los ciclos de trabajo como por ejemplo $D_1 = 62, D_2 = 64, D_3 = 66, \dots, D_n = \dots$. Esto implicaría agregar más estados para cada uno de los valores intermedios. Lo anterior llevaría a una disminución significativa del rendimiento del control, aumentando la recursividad de los estados agregando acciones y condiciones de transición innecesarias. Para sobreponer esta desventaja técnica se opta por implementar un controlador híbrido, es decir, agregar un controlador PID a la MEF. En este caso en particular se calcula un controlador PI (Proporcional-Integral), omitiendo la acción derivativa (di/dt) debido a que no es relevante en este momento dar rapidez al sistema [1]. De manera que el controlador PI solo realizará el ajuste de los ciclos de trabajo en el estado estacionario, es decir, cuando se encuentre dentro del umbral establecido por los límites superior e inferior en la máquina de estados tal como se muestra en la figura a continuación (Ver figura 3.11).



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

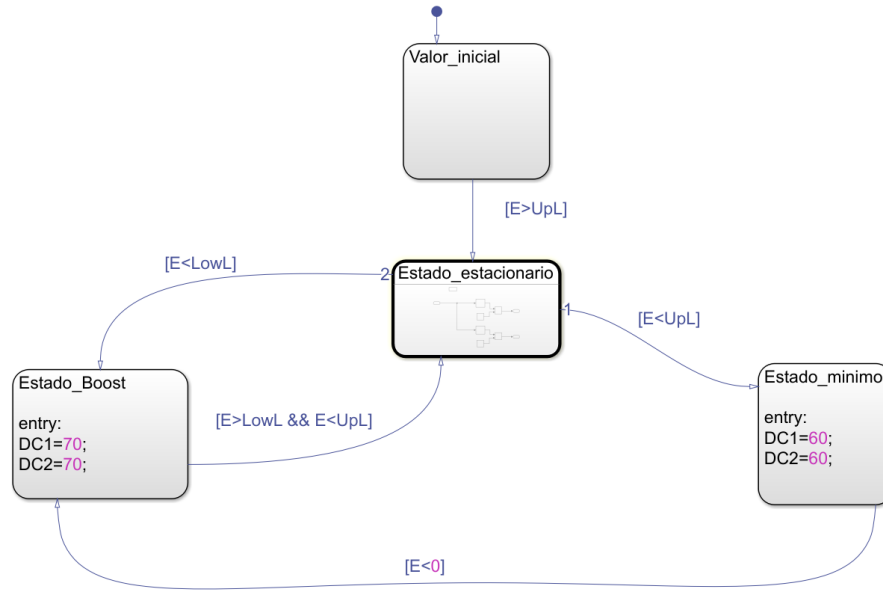


Figura 3. 11 Maquina de estados finitos con controlador PI en estado estacionario Fuente: Elaboración propia en MATLAB/Simscape

Tal como se muestra en la figura 3.11, se agregó el estado estacionario basado en Simulink a la máquina de estados esto con la finalidad de poder implementar el controlador PI, la configuración del controlador se encuentra estructurado de la siguiente manera (Ver figura 3.12).

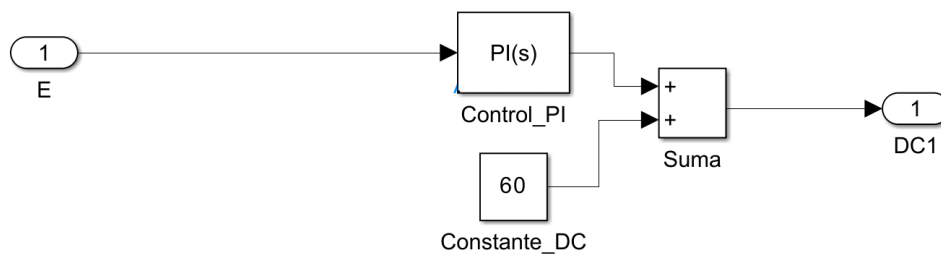


Figura 3. 12 Configuración del controlador PI en el estado estacionario, Fuente: Elaboración propia en MATLAB/Simulink

El estado estacionario está conformado por la variable de entrada denominada para la simulación como “E” la cual es el error (diferencia entre el valor objetivo “Set point” y el valor de corriente actual), dicha variable de entrada está conectada a el bloque interno de la librería Simulink de MATLAB denominado como “Control_PI”, el cual tiene diferentes configuraciones dependiendo el controlador que se desee simular. Una constante numérica “Constante_DC” con un valor fijo de 60, posteriormente se realiza una suma entre el valor calculado por el bloque “Control_PI” y el bloque “Constante_DC”, finalmente el resultado se escribe en la variable “DC1”.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

3.4.1 Configuración del control PI

Como se mencionó anteriormente, es necesario conocer la función de transferencia del sistema para la implementación de este controlador y determinar los valores óptimos de las constantes K_p y K_i . Sin embargo, para implementación y la asignación de cada una de las variables está basada en el conocimiento experto a partir de la metodología de prueba y error, utilizando la herramienta embebida integrada en MATLAB denominada como *System Toolbox* realizando varias ejecuciones de la simulación utilizando la corriente de salida como el criterio principal para determinar el valor óptimo, los parámetros asignados son los siguientes (Ver figura 3.13).

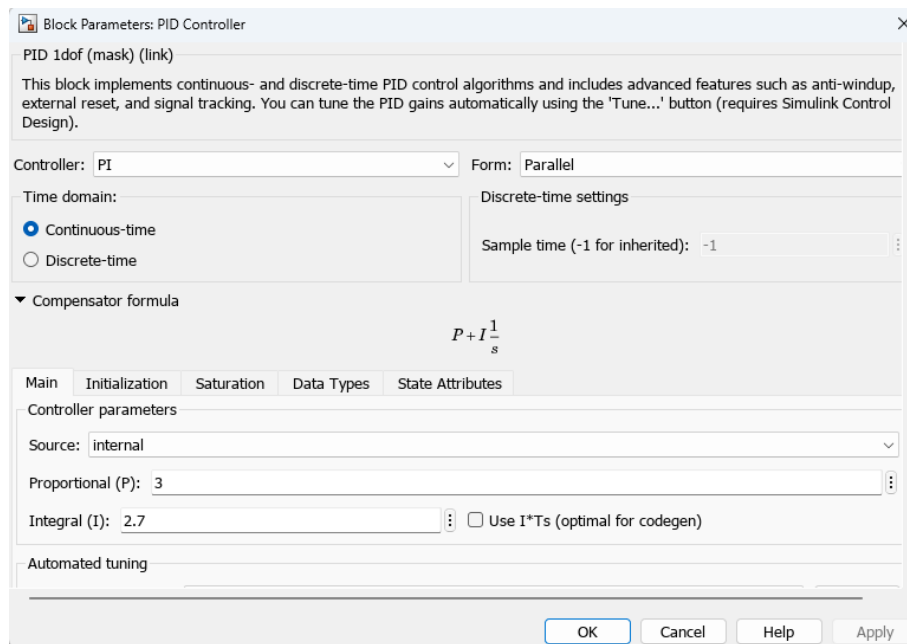


Figura 3. 13 Ventana para fijar los parámetros del controlador PI. Fuente: Elaboración propia en MATLAB/Simulink



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

De manera final, la figura 3.14 muestra un esquema general del sistema con las optimizaciones anteriormente mencionadas y la implementación del controlador PI.

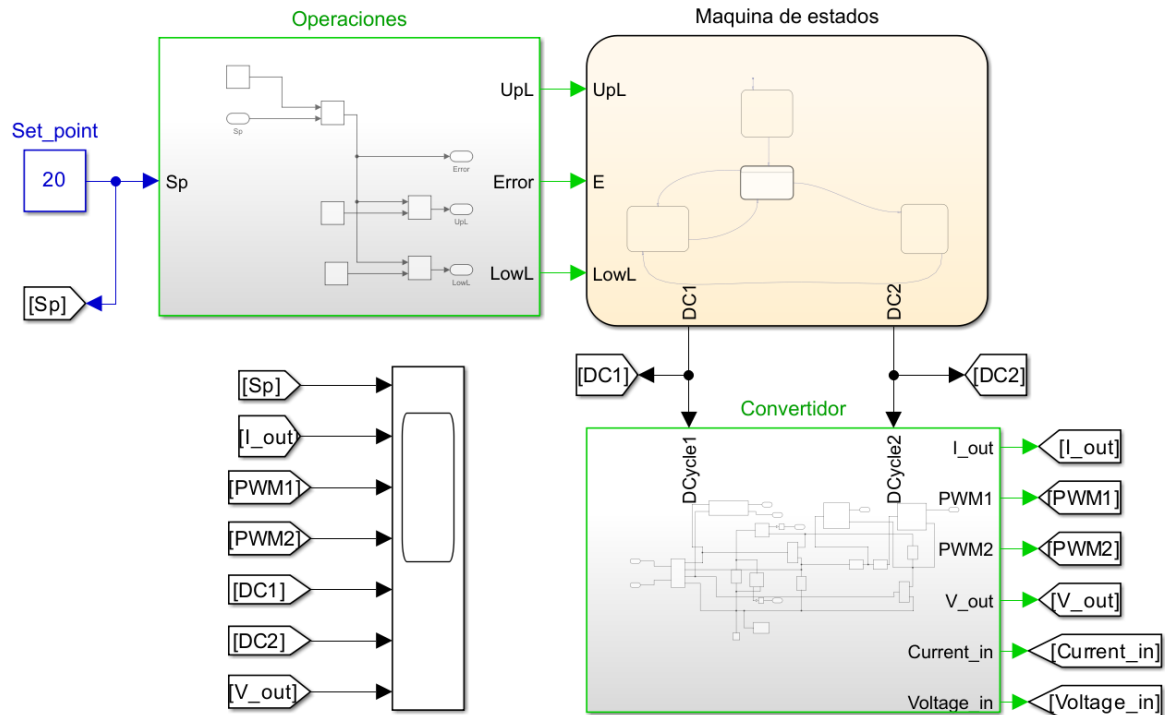


Figura 3. 14 Esquema general de la MEF y el controlador PI. Fuente: Elaboración propia a partir de MATLAB/Simulink

En la figura 3.14, el nuevo sistema modificado funciona a partir de la variable “Set_point” donde se establece el valor objetivo de la corriente de salida el cual ingresa al bloque “Operaciones” donde se establecen el límite inferior “LowL” y el límite superior “UpL” y el error obtenido a partir de la corriente actual de salida y el valor objetivo, dichas variables ingresan al bloque denominado “Maquina de estados” donde se encuentra la máquina de estados previamente establecida, donde a diferencia del diseño anterior, aquí se utilizan las variables correspondientes a los ciclos de trabajo de cada MOSFET, DC1 para M1 y DC2 para M2 respectivamente, los valores de ciclo de trabajo ingresan al bloque “Convertidor” donde se encuentra el convertidor *chopper*.

Finalmente, las variables medibles para la evaluación del convertidor con la máquina de estados y el controlador PI son la corriente de salida (I_out), voltaje de salida (“V_out”), los pulsos de conmutación (PWM1 y PWM2), finalmente el voltaje de entrada (“Voltage_in”) y la corriente de entrada (“Current_in”). Cada uno de los resultados obtenidos a partir de dicha comparación se muestran en la sección 5.2.2.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

4. Desarrollo y análisis

4.1 Introducción.

En esta sección se muestra el desarrollo matemático y el análisis de los parámetros físicos de los componentes electrónicos de potencia, con el objetivo de validar el comportamiento del controlador propuesto, además, se describen cada uno de los subsistemas que conforman el sistema propuesto, comenzando por el diagrama a bloques del sistema y la definición de cada uno de los subsistemas que lo conforman, además del desarrollo de una interfaz para la monitorización del convertidor mediante el PC y el microcontrolador por comunicación serial.

4.2 Vista general del sistema

En la figura 4.1 se muestra disposición general de cada uno de los elementos que conforman el sistema, desde los terminales de alimentación, sistemas de instrumentación para la medición de los parámetros de voltaje y corriente, los transformadores de pulsos y la fase de potencia donde se encuentra el convertidor y la red *Snubber*. El sistema completo se encuentra montado en una tarjeta PCB (*Printed Circuit Board*, por su definición al inglés) con la finalidad de disminuir el ruido electrónico además de mejorar las conexiones físicas entre cada uno de los componentes. El desarrollo del sistema se realizó mediante el software de desarrollo de circuitos electrónicos KiCad en conjunto con la compañía PCBWay para garantizar la mejor calidad en el proceso de fabricación. Los parámetros de control determinados a partir de la simulación del sistema así como también los pulsos de conmutación del sistema se muestran en la sección 5.3



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

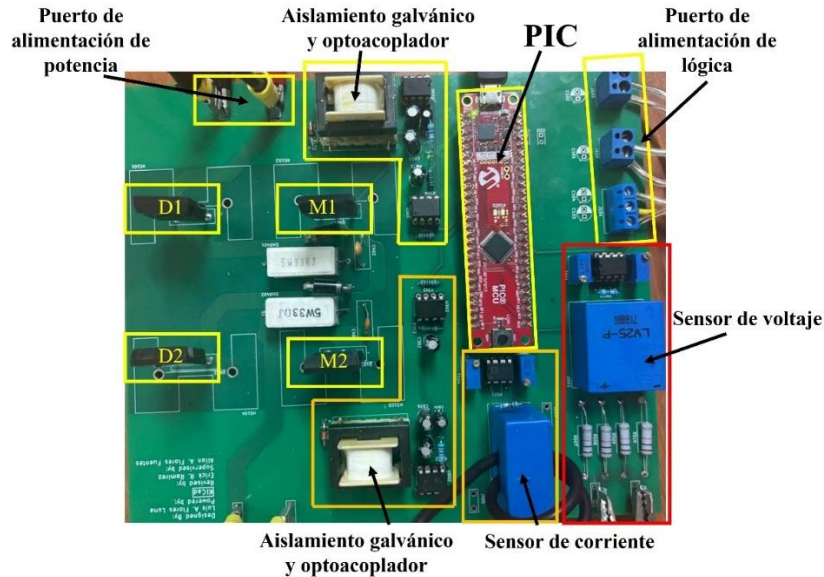


Figura 4. 1 Vista general del prototipo PCB diseñado para su manufactura. Fuente: Elaboración Propia en KiCad®

4.3. Diagrama a bloques del sistema

El sistema de pruebas está conformado por cinco bloques, comenzando por el bloque denominado “Microcontrolador” donde se encuentra alojado el dispositivo que realizara las acciones de control, posteriormente el bloque nombrado como “Optoacoplador” donde se realiza el acoplamiento óptico de la señal de control, a continuación el bloque “Drivers” o también de semiconductores se adapta la señal de control a niveles adecuados para posteriormente llegar al bloque de “Potencia” donde se encuentra el convertidor chopper de dos cuadrantes en modo corriente, finalmente se encuentra el bloque de “Instrumentación” el cual realiza la adaptación de las señales de voltaje y corriente de salida del convertidor y los ajusta a niveles adecuados para el bloque donde se encuentra el microcontrolador. Las conexiones eléctricas de cada uno de estos subsistemas se muestran en la figura 4.3. Por otra parte, cada uno de los bloques que conforman el sistema se detallan en las secciones siguientes.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

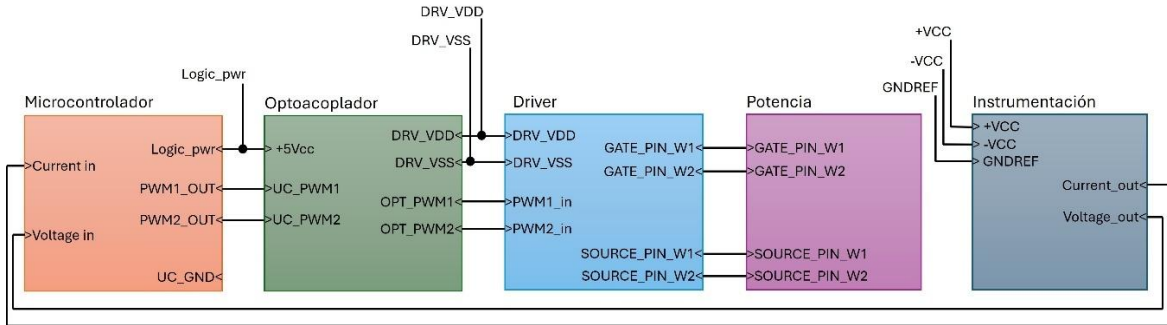


Figura 4. 2 Diagrama de bloques de los subsistemas. Fuente: Elaboración propia en KiCad®

Con el objetivo de mostrar una imagen detallada del convertidor, en la figura 4.3 se muestran los componentes que conforman cada uno de los bloques mostrados en la figura 4.2



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

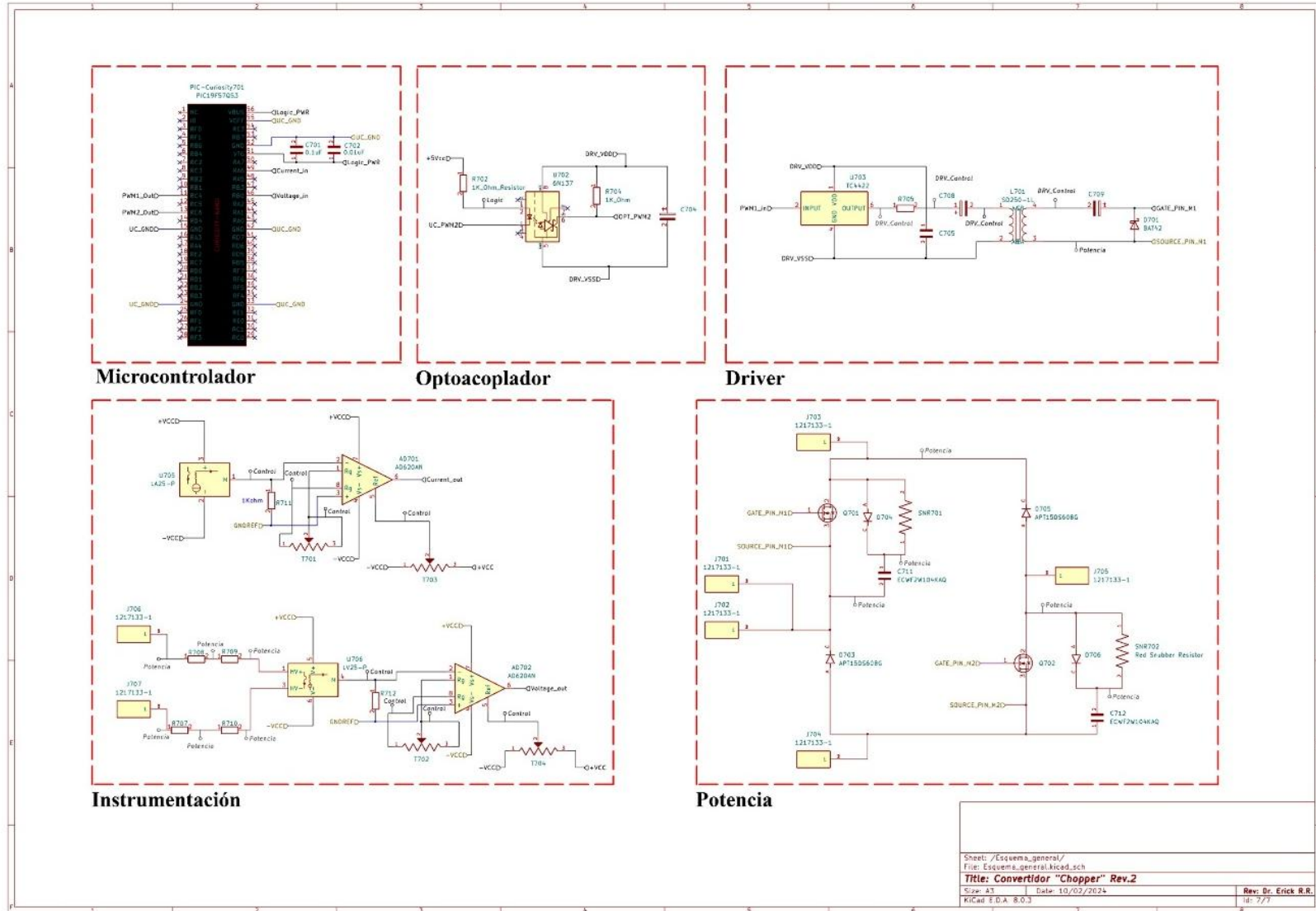


Figura 4. 3 Diagrama de conexiones eléctricas del sistema. Fuente: Elaboración propia en KiCad ®



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

4.3.1. Placa de desarrollo Curiosity Nano®.

La placa de desarrollo Curiosity Nano® de la marca Microchip® incorpora un microcontrolador PIC18F57Q10 el motivo por el cual se implementa este microcontrolador al sistema es debido a que cumple con las especificaciones requeridas para el convertidor como la generación y modulación de ancho de pulso (PWM) con la posibilidad de desfase los pulsos a un valor especificado por el usuario. Su arquitectura y características del sistema se detallan la figura 4.4.

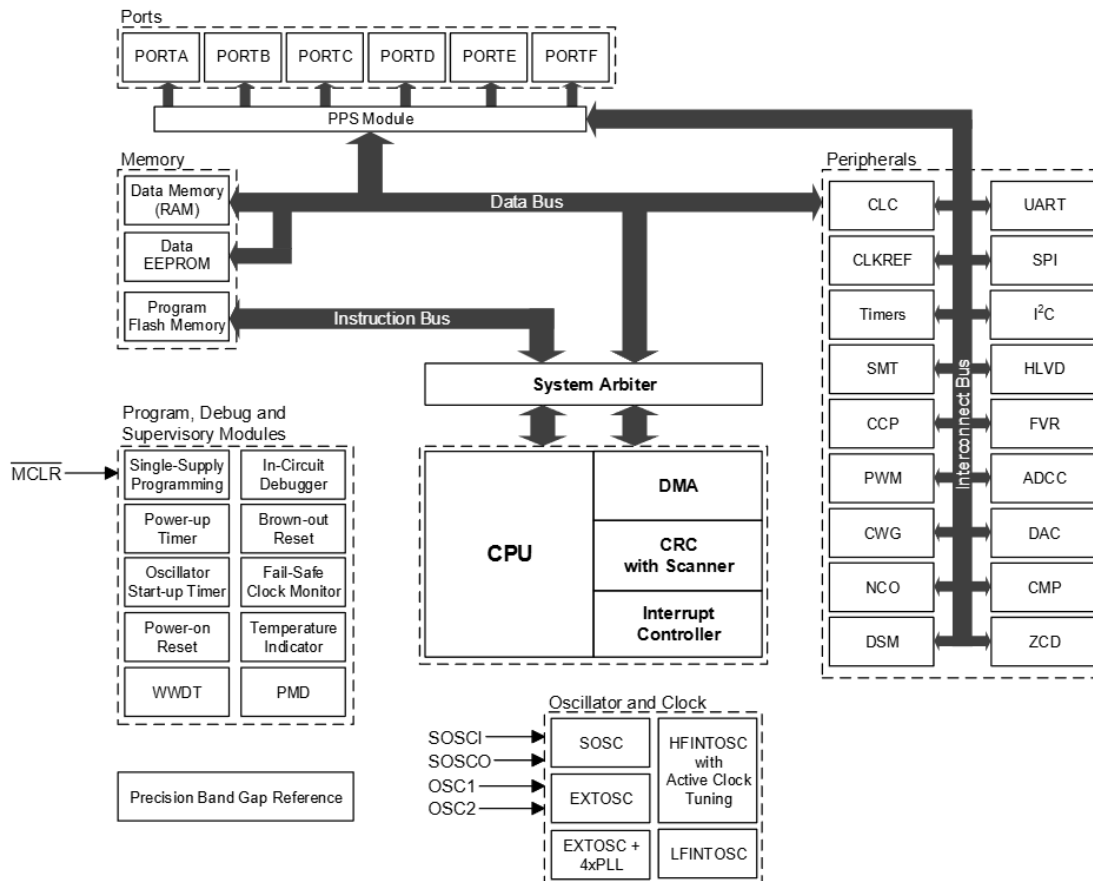


Figura 4. 4 Diagrama a bloques del microcontrolador PIC18F57Q10. Fuente: Microchip ®

En la figura 4.4 el microcontrolador cuenta con más periféricos integrados como el protocolo de comunicación UART (*Universal Asynchronous Receiver-Transmitter*, por nomenclatura en inglés) para facilitar la comunicación entre el ordenador y microcontrolador, también cuenta con un convertidor analógico-digital (ADC, por sus siglas en idioma inglés) para la lectura de pines específicos, sin embargo, una de las características más relevantes es la función integrada (CIP, *Core Independent Peripherals* por su nomenclatura en inglés) la cual se ha mencionado anteriormente a lo largo del presente documento, la cual consiste en el



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

funcionamiento de cada uno de los módulos anteriormente mencionados en segundo plano, permitiendo que el CPU realice otras funciones sin afectar el rendimiento y la ejecución de cada una de las instrucciones del código.

Por otra parte, en la figura 4.5 se muestra la conexión y los pines utilizados para la lectura y salida de los pulsos de conmutación, así como también la configuración para la alimentación del microcontrolador.

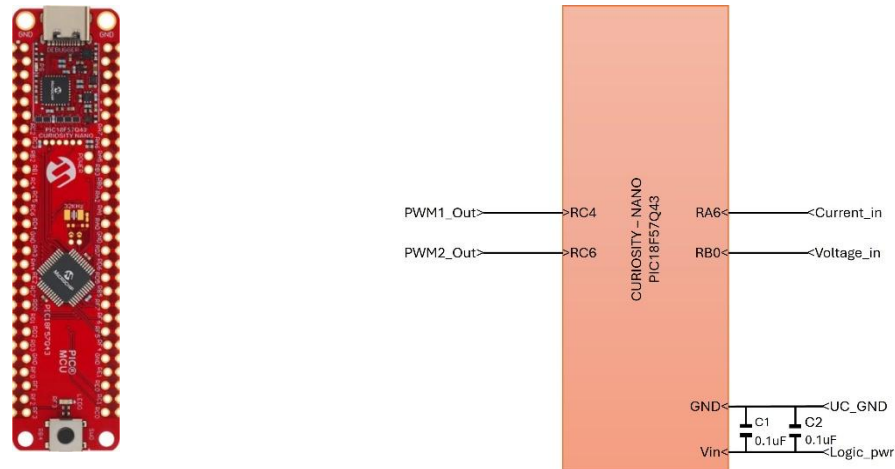


Figura 4. 5 Diagrama de conexión del microcontrolador. Fuente: Elaboración propia

En la figura 4.5 se agregaron dos capacitores (C1 y C2) de 0.1 μ F cada uno, con la finalidad de disminuir los picos de voltaje o sobre voltaje si ese fuera el caso, con el objetivo de no dañar el sistema de control.

4.3.2. Acoplamiento óptico

Se utilizaron dos optoacopladores 6N137 con el objetivo de realizar un aislamiento lógico y óptico, es decir, una protección para el sistema de control de manera que si se presentara algún cortocircuito serían los optoacopladores los primeros en dañarse en lugar del microcontrolador. La distribución de pines y las características del dispositivo se detallan en la hoja de datos del fabricante (Ver TOSHIBA Inc. [53]).

El optoacoplador 6N137 consiste en un diodo de alta frecuencia de emisión y un foto circuito integrado en un solo chip. Lo que lo convierte en un dispositivo ideal para la frecuencia a la



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

que van a operar los pulsos de conmutación de 80 KHz elegida como una característica. La conexión eléctrica de los optoacopladores se muestra en la figura 4.6.

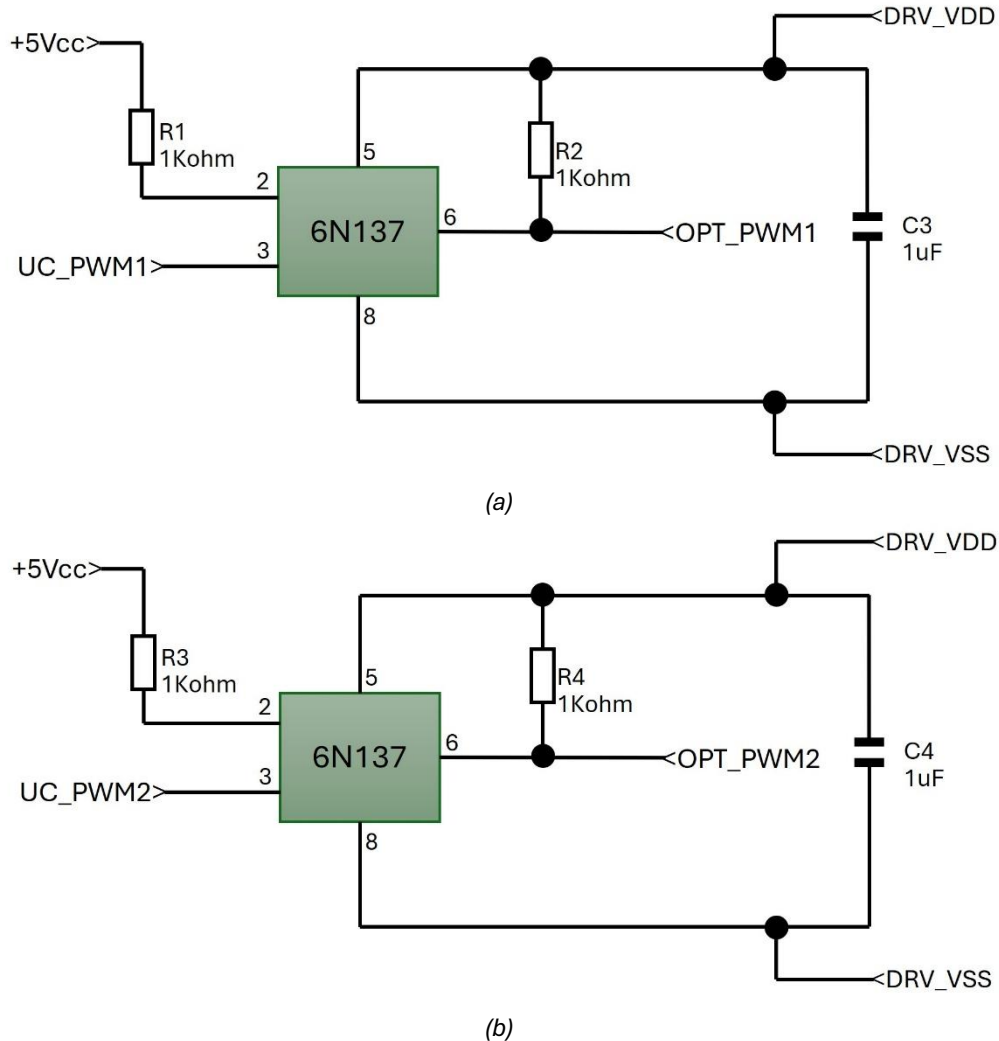


Figura 4. 6 (a) Optoacoplador para la señal PWM1. (b) Optoacoplador para la señal PWM2. Fuente: Elaboración propia

Con la configuración utilizada en la figura 4.6, el modo de conmutación del optoacoplador es en el flanco descendente, aunque puede también utilizarse el flanco ascendente, no existe un motivo específico por el cual se prefiera la forma de activación anteriormente mencionada. Por otra parte, el valor de cada uno de los componentes viene especificada en la hoja de datos del fabricante.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

4.3.3. Drivers o manejadores para MOSFET

En el bloque de los drivers se realiza un ajuste en el nivel de voltaje de los pulsos debido a que, hasta el momento, tanto el bloque del microcontrolador como el bloque de optoacopladores trabajan con un voltaje de conmutación de +5 V, sin embargo, para la operación correcta de los MOSFETS a partir de su hoja de datos el voltaje de operación debe ser de +15 V. Con base en lo anterior, se utiliza el controlador de alta velocidad TC4422 y capaz de suministrar alta corriente alrededor de 9 A en la compuerta-fuente de los dispositivos semiconductores, la disposición de pines y sus características eléctricas se encuentran en la hoja de datos publicada por el fabricante Microchip ® [54]

Tomando como referencia la hoja de datos del fabricante mostrada en Microchip ® [54], el controlador TC4422 tiene un voltaje de salida de +18 V como máximo y requiere una señal de conmutación a la entrada con un voltaje de +4.5 V. Tal como menciona el fabricante, estos dispositivos son inmunes a perturbaciones, excepto sobretensión eléctrica. Incluso pueden soportar un voltaje a la entrada de +5 V, además las terminales están protegidas ante descargas electrostáticas de hasta 4 kV. Es importante señalar que cuando se está trabajando con este tipo de convertidores de potencia se generan capacitancias e inductancias parasitas, así como también ruido electrónico resultado de la conmutación a alta frecuencia de los MOSFETS este evento (encender y apagar los transistores) genera transiciones abruptas de voltaje y corriente por lo que se crean picos de alta frecuencia (armónicos) en las señales de control y de conmutación.

Entonces, es necesario implementar una técnica de aislamiento con la finalidad de mantener la afinidad del pulso de conmutación, una técnica de aislamiento ampliamente utilizada es el aislamiento galvánico, la cual, consiste en la implementación de un transformador de pulsos y componentes pasivos (capacitores, resistencias, diodos) conectados como se muestra en la figura 4.7.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

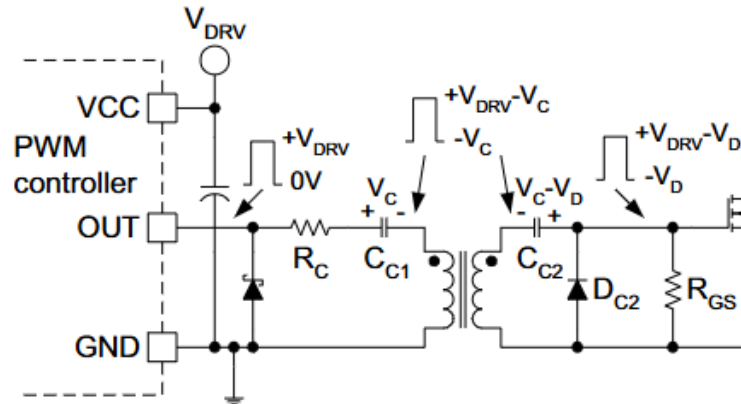


Figura 4. 7 Topología del aislamiento galvánico utilizando un transformador de pulsos. Fuente: Texas Instruments® [55]

Para el diseño del aislamiento galvánico basado en la topología mostrada en la figura 4.8 es fundamental que el transformador de pulsos mantenga una relación de 1:1 para obtener una señal de conmutación con los mismos valores generados por el *driver* TC4422. Por lo cual, el transformador SD2501L cumple con los requerimientos previamente mencionados a partir de la hoja de datos del fabricante Coilcraft (r) [56]

Así, para el diseño y los cálculos necesarios para la obtención de los valores de cada uno de los componentes que conforman el circuito se detalla a continuación. Para realizar esta operación se realizó un *script* de código para realizar cada una de las ecuaciones en base a las ecuaciones que se muestran a continuación.

$$C_{C2} = \frac{Q_G}{V_{C2}} + \frac{(V_{DRC} - V_{DC}) * D_{MAX}}{V_{C2} * R_{GS} * f_{DRV}} \quad (4.1)$$

Donde:

C_{C2} = Capacitancia del capacitor 2

Q_G = Coeficiente galvanico

V_{C2} = Voltaje del capacitor 2

V_{DRC} = Voltaje del driver

V_{DC} = Voltaje del ciclo de trabajo

D_{MAX} = Ciclo de trabajo maximo

R_{GS} = Resistencia Compuerta fuente

f_{DRV} = Frecuencia del driver



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

$$C_{C1} = \frac{Q_G}{V_{C1}} + \frac{(V_{DRC} - V_{DC2}) * D}{V_{C1} * R_{GS} * f_{DRV}} + \frac{V_{DRV} * (D^2 - D^3)}{V_{C1} * 4 * L_M * f_{DRV}^2} \quad (4.2)$$

Donde:

Q_G = Coeficiente galvanico

C_{C1} = Capacitancia del capacitor 1

V_{DRC} = Voltaje del driver

R_{GS} = Resistencia compuerta fuente

V_{DC2} = Voltaje del capacitor 2

V_{DRV} = Voltaje del driver

f_{DRV} = Frecuencia de conmutacion del driver

D = Ciclo de trabajo

L_M = Coeficiente del inductor del transformador

Ambas ecuaciones se requieren para obtener el valor del capacitor C1 y C2. Por otra parte, existen otras ecuaciones definidas por el fabricante *Texas Instruments* [55], las cuales son para definir los valores de las variables que conforman las ecuaciones anteriores, las cuales se muestran a continuación.

$$V_C = D * V_{DRV} \quad (4.3)$$

Donde:

V_C = Voltaje del capacitor

D = Ciclo de trabajo

V_{DRV} = Voltaje del driver

$$R_C \geq 2 * \sqrt{\frac{L_M}{C_C}} \quad (4.4)$$

Donde:

R_C = Resistencia del capacitor

L_M = Coeficiente del inductor del transformador

C_C = Valor del capacitor

Para hacer una tarea sencilla el cálculo de los componentes se implementó un *script* de código para realizar el cálculo con la finalidad de obtener el valor de cada uno de los componentes a partir de las ecuaciones anteriormente mostradas con la finalidad de variar los valores del



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

transformador de pulsos y obtener así un valor ideal para la conmutación del sistema y disminuir el ruido electrónico por conmutación y evitar que se filtren los armónicos por conmutación del convertidor.

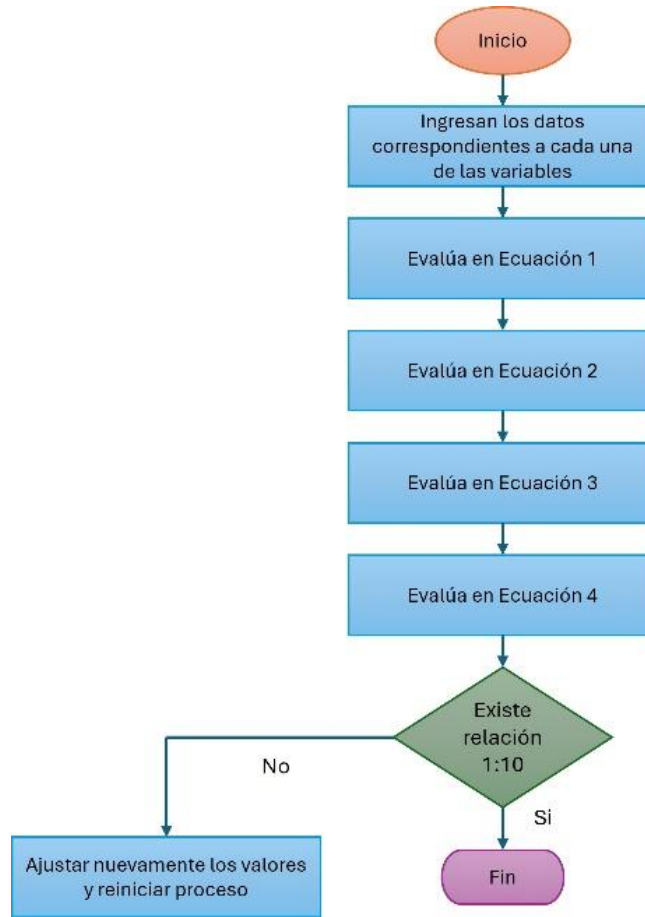


Figura 4. 8 Diagrama de flujo del script desarrollado. Fuente: Elaboración propia

En la figura 4.8 se muestra el funcionamiento del script a partir del diagrama de flujo, donde se evalúan los valores de cada una de las ecuaciones, donde el criterio de paro para el script es tal como menciona el fabricante tener una relación 1:10 entre la entrada (pulso de salida del *driver* o manejador de pulsos) y la salida (señal de salida del transformador). Para más detalles relacionados al script de código véase Anexo A.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

A continuación, se muestra el diagrama eléctrico del transformador de pulsos, como el transformador cuenta con dos dispositivos de conmutación se implementa el mismo circuito para cada uno de los MOSFETS, tal como se muestra en la figura 4.9.

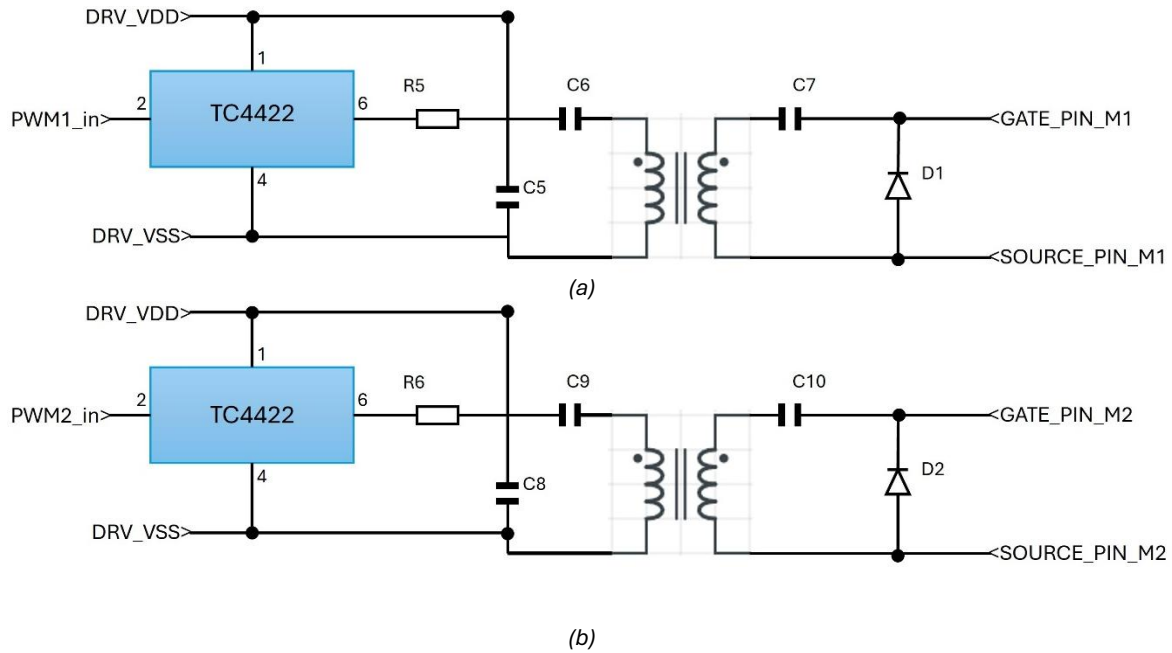


Figura 4. 9 (a) Transformador de pulsos para el MOSFET M1. (b) Transformador de pulsos para el MOSFET M2.
Fuente: Elaboración propia

Para el funcionamiento del circuito se cuenta con los pines “DRV_VDD” el cual va conecta a 15V y “DRV_VSS” el cual está conectado a GND, el pin por donde ingresa los pines esta nombrado por “PWM1_in”. Finalmente, a la salida solo hay dos pines el “GATE_PIN_M1” que como su nombre lo indica va conectado al pin de la compuerta del MOSFET y el pin “SOURCE_PIN_M1” del mismo modo como su nombre lo indica se conectará al pin fuente del MOSFET. Del mismo modo para el circuito de la figura 4.11.b solo que se conectara al MOSFET inferior.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

4.3.4. Etapa de potencia: Convertidor en modo corriente

El bloque de Potencia se encuentra el convertidor chopper de dos cuadrantes en modo corriente, siguiendo la topología propuesta por B. Williams [1] la cual consiste por dos dispositivos de conmutación y dos diodos auxiliares. Por lo cual, comenzando por el dispositivo semiconductor utilizado para la conmutación que tiene como matrícula APT6030BVFR, la distribución de pines y las características eléctricas del MOSFET se encuentran en la hoja de datos del fabricante [51].

A partir de la referencia [51] el MOSFET puede operar hasta un máximo de 30V en el pin “Gate” (compuerta, por su traducción al español) en este caso se está utilizando un voltaje de 15 V y puede tolerar hasta un máximo de 600 V en los pines *Drain-Source* (Drenador-Fuente, por su traducción en español), lo que lo convierte en un dispositivo más que ideal para su implementación en el convertidor debido a que el voltaje máximo de operación del convertidor es de 180 V.

Por otra parte, los diodos auxiliares implementados en el sistema son APT15DS60BG, la distribución de pines y las características eléctricas de los diodos auxiliares se muestran en la hoja de datos [57]

Partiendo de las características del MOSFET utilizado donde el voltaje máximo de operación entre los pines *Drain-Source* de 600 V, es necesario utilizar dos diodos auxiliares que sean capaces de soportar este voltaje como punto máximo, de manera que el diodo APT15DS60BG soporta un voltaje máximo de 600 V, esto se puede concluir a partir de la hoja de datos [57]. Por lo que es el dispositivo ideal para el convertidor.

Finalmente, es necesario mencionar que con solo implementar la topología clásica del convertidor chopper propuesta, se generan armónicos por la conmutación a altas frecuencias de operación de los MOSFETS, por lo cual, es necesario implementar dos redes *Snubber* para garantizar el proceso de carga y descarga del MOSFET. De forma que en la figura 4.10 se muestra el convertidor *chopper* clase D de dos cuadrantes en modo corriente con la red *Snubber* implementada.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

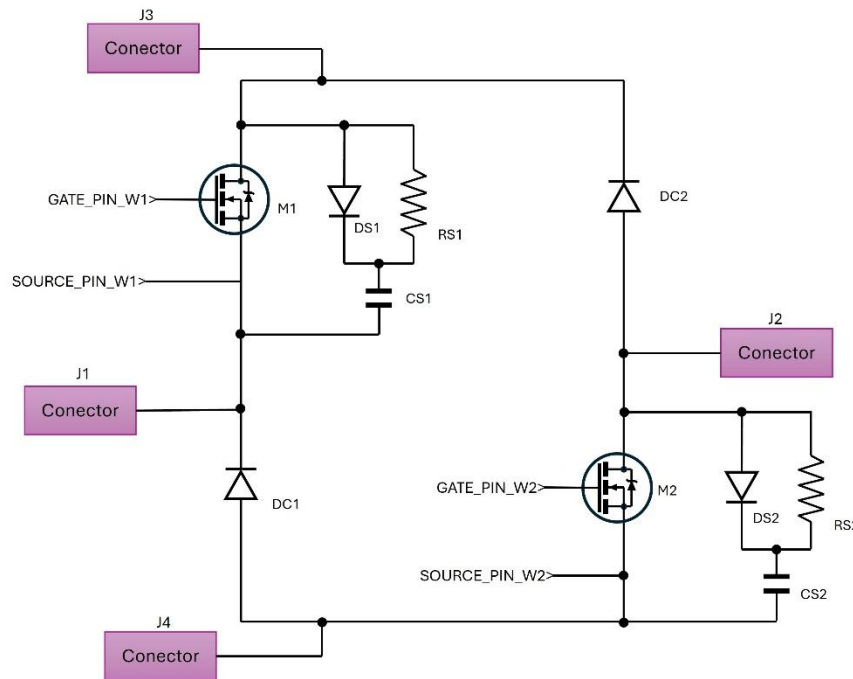


Figura 4. 10 Convertidor chopper con red snubber implementada. Fuente: Elaboración propia.

De manera que las señales de conmutación ingresan por los pines de cada uno de los MOSFETS (GATE_PIN_W1 y GATE_PIN_W2, respectivamente) M1 y M2. Además, se agrega el pin “SOURCE_PIN_W1” y “SOURCE_PIN_W2” que corresponde a el pin *source* de cada uno de los MOSFETS M1 y M2.

Para el desarrollo e implementación de la red *Snubber* se realiza a partir de la siguientes ecuaciones, comenzando por el cálculo del pico transitorio, definido por:

$$E = \frac{1}{2} L_p I^2 \quad (4.5)$$

Donde:

E = Pico transitorio

L_p = Inductancia parasita

I = Corriente que fluía justo antes del corte

Como la topología implementada es del tipo RC, la cual debe ir en paralelo con el interruptor. Esta topología disipa la energía almacenada en L_p a través de la resistencia y limita la tasa



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

máxima de cambio de voltaje mediante el capacitor. Por lo cual, para la definición del capacitor se realiza mediante:

$$C = \frac{I_{peak} * t_{rise}}{V_{max}} \quad (4.6)$$

Donde:

I_{peak} = Corriente pico [A]

t_{rise} = Tiempo de subida del voltaje

V_{max} = Voltaje máximo del MOSFET

Finalmente, la definición del valor de la resistencia se realiza por:

$$R = \sqrt{\frac{L_p}{C}} \quad (4.7)$$

Donde:

L_p = Inductancia parasita

C = Capacitor de la red Snubber

En el anexo B. se muestra de forma detallada el script en MATLAB para la estimación de los valores de los componentes que conforman la red Snubber.

4.3.5. Sección de instrumentación.

El bloque “Instrumentacion” está conformado por dos sensores para medir voltaje y corriente, ambos sensores deben tolerar un voltaje máximo de 180 V y una corriente de 10 A respectivamente. Por lo cual, se utilizó el sensor LA25-P para medir la corriente y el sensor LV25-P para medir el voltaje, las características eléctricas se detallan en la hoja de datos del fabricante [58]

El sensor LA25-P es capaz de realizar mediciones de hasta 25 A ya que es un sensor de efecto hall donde la relación es de 1:1000 es decir que a la salida se obtendrán 25 mA. Además, este sensor requiere de una fuente bipolar de ± 12 V hasta ± 15 V es necesario mencionar que los valores de las resistencias auxiliares para la conexión del sensor, está determinada por el voltaje con el que se alimenta el sensor, en este caso se está alimentando con un voltaje de



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

$\pm 12 V$ de manera que la resistencia mínima es de 10Ω . Es importante tenerlo en cuenta debido a que si se trabaja con un valor menor al especificado se puede dañar de forma inmediata y permanente el sensor de corriente.

Por otra parte, el diagrama de conexión y las características eléctricas del sensor LV25-P se muestran en la hoja de datos LEM [59].

Por lo que el sensor de voltaje es capaz de realizar mediciones a partir de los 10 V hasta los 500 V, lo que lo convierte en una solución sólida para la implementación en el sistema, esto se debe a que el sensor estructuralmente es un transductor por lo cual, de forma similar el sensor tiene una relación de conversión de 2500:1000 por lo cual a la salida y de forma similar se obtendrán 25 mA así como también, se debe considerar el valor mínimo de la resistencia que lo conforma el cual es de 30Ω .

Se omite la implementación de un divisor de voltaje, debido a la topología del convertidor, al tener un MOSFET flotado, no hay una referencia a tierra que permita realizar la medición de voltaje de forma sencilla y eficiente con el microcontrolador. Entonces, es necesario mencionar que al tener relaciones de conversión distintas ambos sensores, no es posible conectarlos de forma directa al microcontrolador, por lo tanto, se utiliza un amplificador de instrumentación ampliamente utilizado en aplicaciones de electrónica de potencia y mediciones precisas el AD620AN, la distribución de pines, diagrama de conexión y características eléctricas pueden ser consultadas a partir de Texas Instruments [60].

Para finalizar esta etapa de instrumentación, la implementación de los sensores en conjunto con el amplificador operacional es necesario mencionar que el ancho de banda definido por el fabricante es de 49.4 KHz, y la definición de la ganancia, en este caso es de 1, la definición de la ganancia deseada es importante para determinar el valor de la resistencia de la ganancia R_G la cual es un valor fijo para definir la relación entre la entrada y salida del sensor. El cálculo puede realizarse mediante la ecuación definida por el fabricante.

$$G = 1 + \frac{49.4k\Omega}{R_G} \quad (4.8)$$



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Donde:

G = Ganancia del amplificador

R_G = Resistencia de ganancia

A partir de los valores obtenidos de (4.8), en la figura 4.11 se muestra el diagrama de conexión de los sensores de corriente y de voltaje a los cuales se implementa un amplificador operacional AD620.

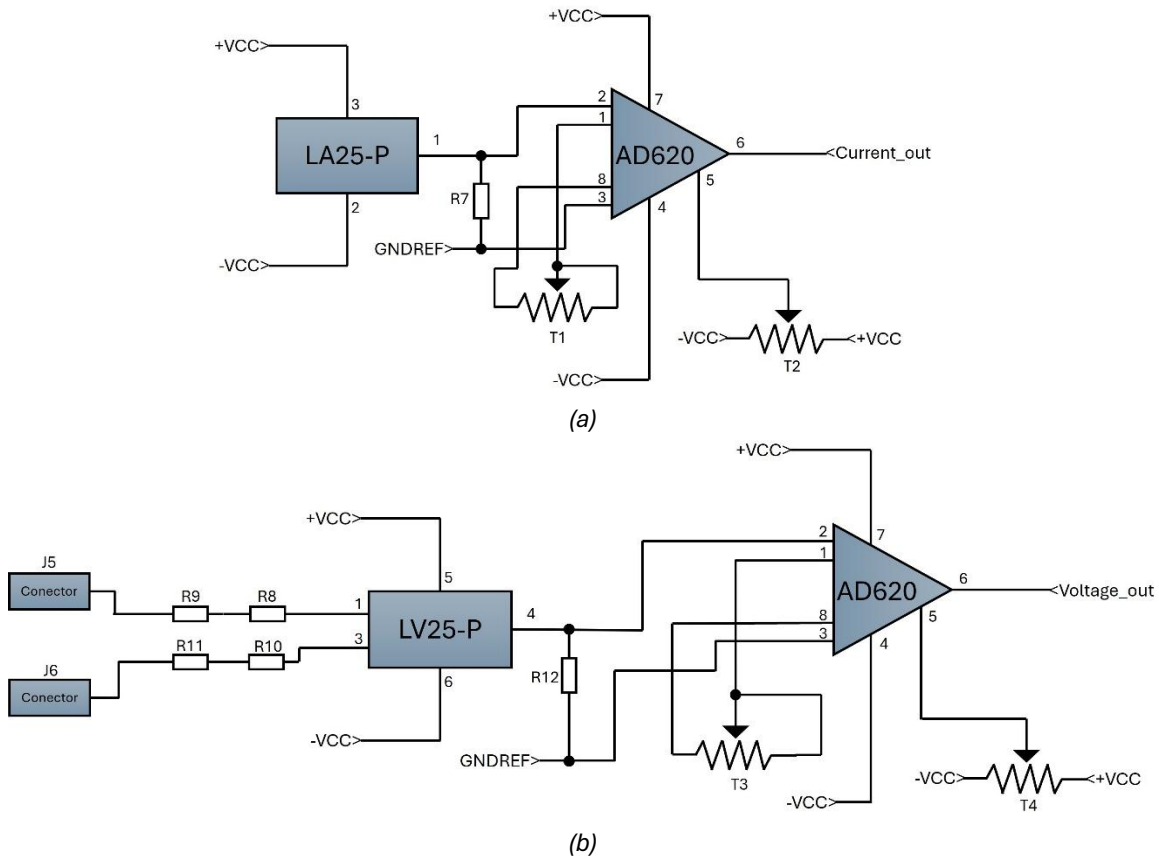


Figura 4. 11 (a) Sensor de corriente. (b) Sensor de voltaje. Fuente: Elaboración propia.

Finalmente, se agregaron dos potenciómetros para el sensor de corriente (T1 y T2) así como también el sensor de voltaje (T3 y T4) donde T1 y T3 son los potenciómetros de la ganancia, finalmente T2 y T4 son el ajuste para el *offset* del amplificador operacional.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

4.3.5.1. Filtro pasa bajo digital

Para la mejora en la señal de control (corriente de salida) se implementa el filtro media móvil RMS (*Root Mean Square*, por su abreviación en inglés) la cual es una técnica utilizada para calcular el valor eficaz de una señal variable [61]. Es especialmente útil para realizar la estimación del valor eficaz de la corriente, incluso cuando la señal de corriente presente ruido, por lo cual es una ventaja debido a que no requiere de componentes adicionales en especial para este tipo de controladores basados en microcontrolador donde sus recursos llegan a ser limitados. Para su implementación en el microcontrolador, la fórmula utilizada es la siguiente:

$$x_{RMS} = \sqrt{\frac{1}{N} \sum_{n=1}^N x^2[n]} \quad (4.9)$$

Donde:

$x[n]$ = *muestras digitales de voltaje o corriente.*

N = *Numero de muestras acumuladas.*

El funcionamiento del filtro aplicado al sistema, esta descrito por el siguiente diagrama de flujo en la figura 4.12.

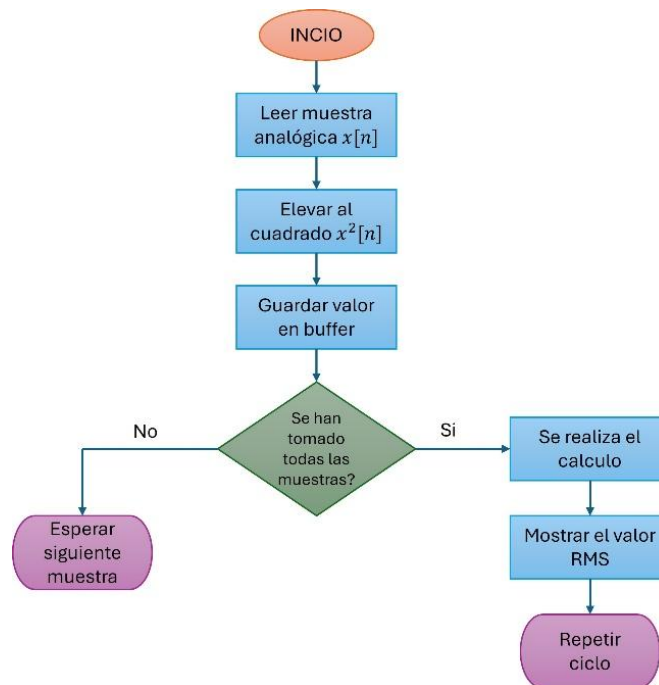


Figura 4. 12 Diagrama de flujo del filtro RMS utilizado. Fuente: Elaboración propia



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

4.4 Desarrollo e implementación del código de la MEF y el controlador PI

En base a la MEF y el controlador PI simulado en MATLAB/Simulink a continuación se muestra el diagrama de flujo en donde se detallan cada una de las etapas que conforman el código principal que ejecuta el microcontrolador (ver figura 4.13).

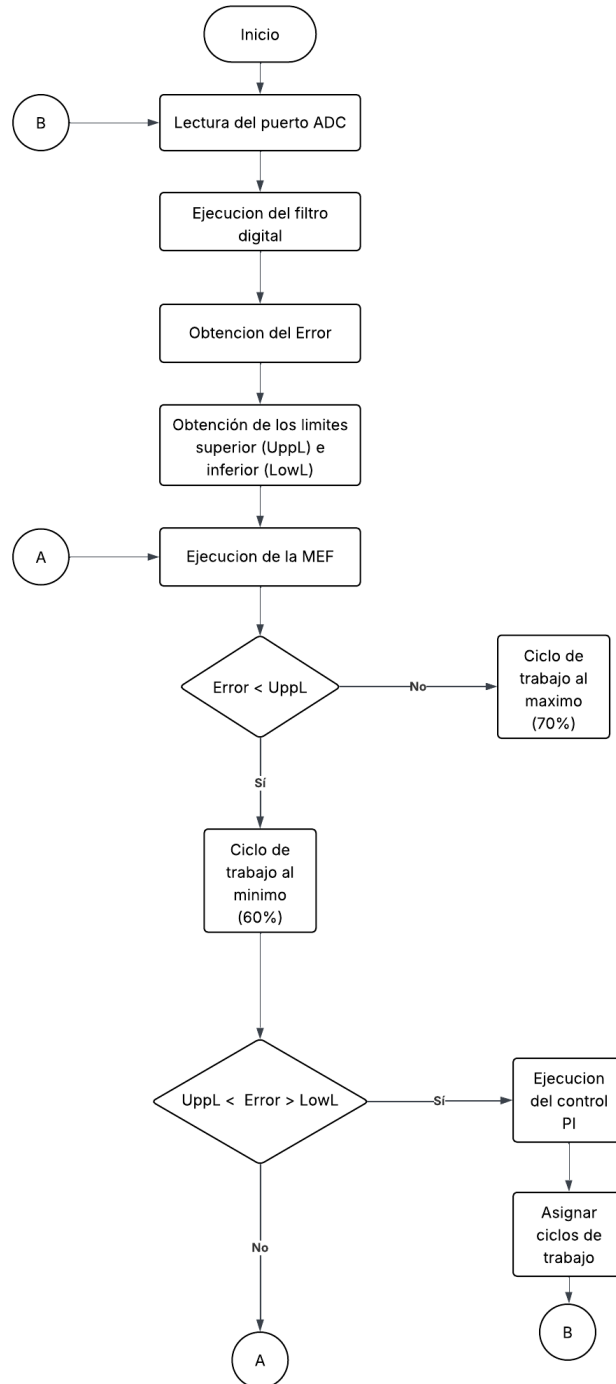


Figura 4. 13 Diagrama de flujo del algoritmo de control en el microcontrolador. Fuente: Elaboración propia



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Es necesario señalar que el diagrama de flujo solo abarca las funciones principales que conforman el control. Para conocer con más detalle el funcionamiento del código, en la sección de anexos específicamente el Anexo D se encuentra el código completo utilizado e implementado en microcontrolador.

4.5 Desarrollo e implementación de la interfaz para monitoreo de parámetros.

Como parte práctica y complementaria se realiza la implementación de una aplicación de monitoreo basada en el lenguaje de programación Python con el software *Visual Studio Code*. La figura 4.14 muestra el diagrama UML (Lenguaje de Modelado Unificado, por su traducción al español) para explicar el funcionamiento de la aplicación desarrollada.

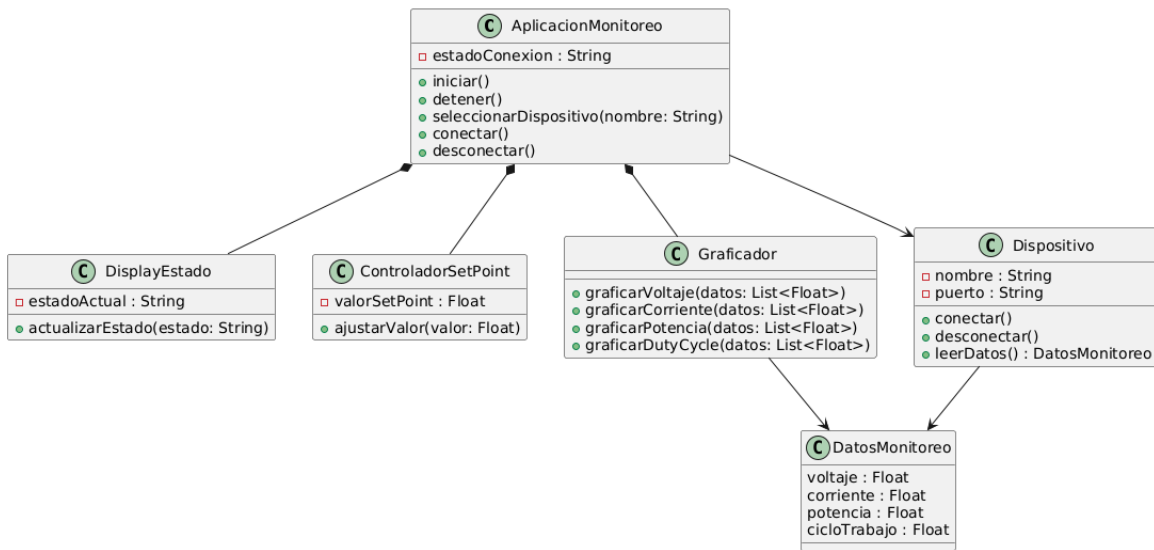


Figura 4. 14 Diagrama UML de aplicación. Fuente: Elaboración propia

En la imagen 4.14 se muestra el diagrama UML conformado por cada una de las clases que definen a la aplicación, sin embargo, no hay alguna visualización gráfica de la organización de cada uno de estos elementos que conforman esta aplicación, por lo cual en la figura 4.15 se muestra la interfaz gráfica de monitoreo para el convertidor *chopper* con cada uno de los parámetros fundamentales que se monitorean a partir de la interfaz gráfica, por otra parte se agregaron indicadores numéricos para obtener lecturas precisas además de la ayuda gráfica de cada uno de los parámetros. Con base en lo anterior se agregaron 2 *sliders* gráficos de



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

forma vertical para ajustar los ciclos de trabajo de forma manual y un *slider* grafico de forma horizontal para definir el Set_point esto con la finalidad de hacer más atractiva la interfaz y ajustar de manera precisa los valores numéricos.

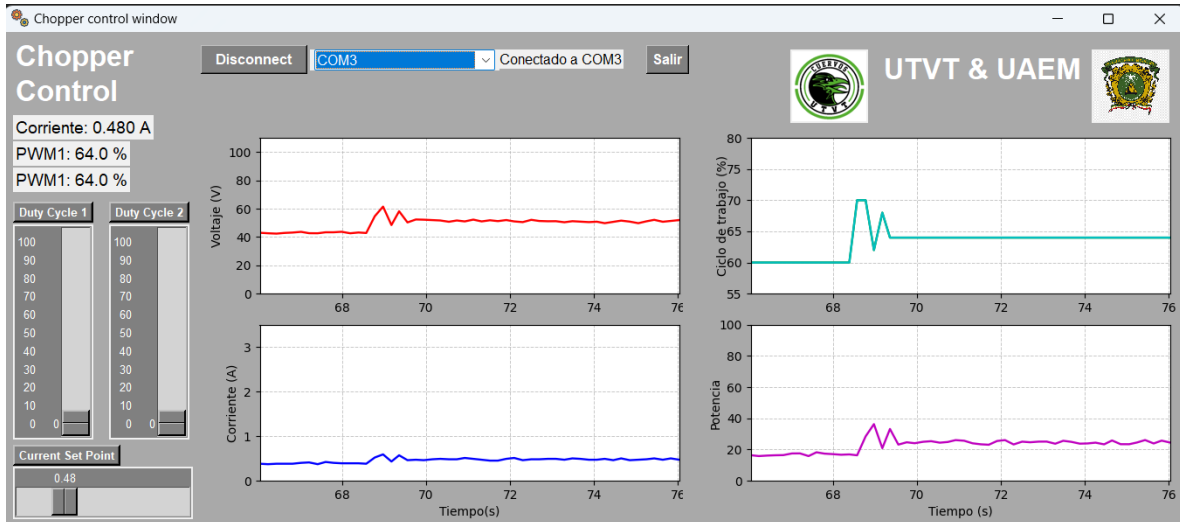
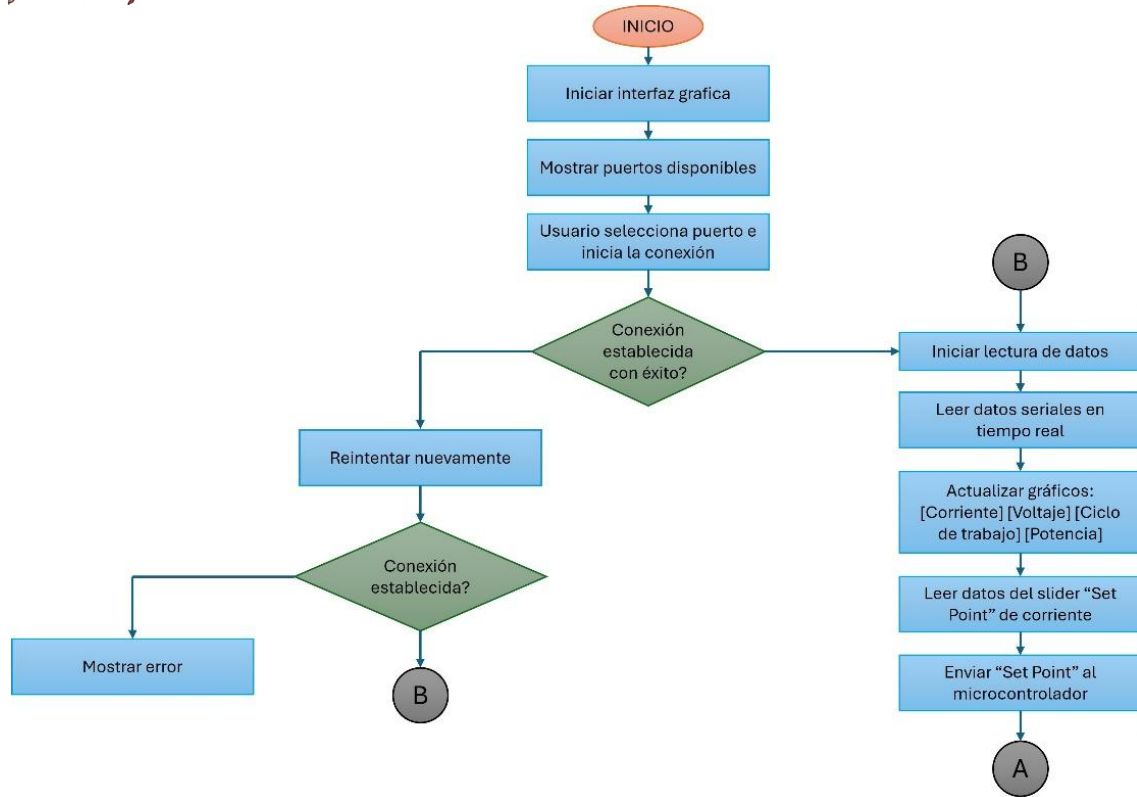


Figura 4. 15 Interfaz gráfica de monitoreo del sistema. Fuente: Elaboración propia

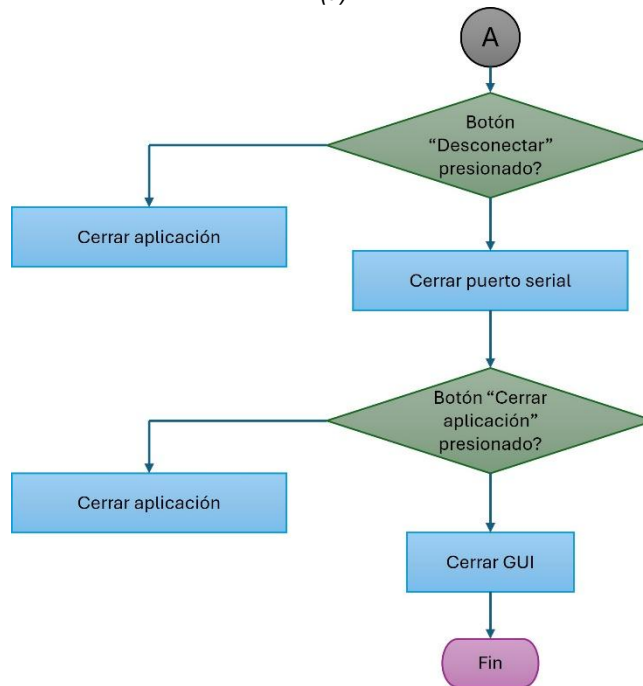
Finalmente, en la figura 4.16 se muestra el funcionamiento de la aplicación de monitoreo de forma más detallada. La interfaz realiza acciones en base a los eventos como la conexión o desconexión del microcontrolador a partir del puerto serial, así como también el cierre de la aplicación. Finalmente, la adquisición de datos se realiza mediante la conexión y la lectura continua del puerto serial y cada uno de los datos se muestran dinámicamente las gráficas mostradas al usuario. En el anexo C se muestra el script de código en Python con el objetivo de facilitar la comprensión de su funcionamiento.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”



(a)



(b)

Figura 4. 16 (a) Diagrama de flujo para la conexión del microcontrolador por comunicación serial (b) Diagrama de flujo para la desconexión del microcontrolador. Fuente: Elaboración Propia



5. Resultados

5.1 Resultados de simulación del sistema

Los pulsos de conmutación obtenidos a partir de la figura 3.7 es posible visualizarlos mediante el bloque PWM_Display de la figura 3.6 por lo que los pulsos de conmutación se observan en la figura. 5.1.

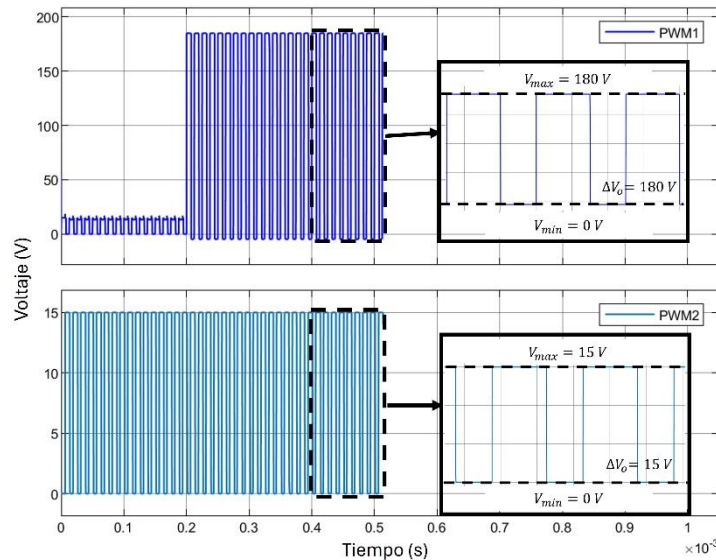


Figura 5. 1 Señales de conmutación PWM1 y PWM2. Fuente: Elaboración propia

El motivo por el cual, uno de los pulsos de conmutación llega a tener valores de conmutación máximos de 180 V se debe al tipo de conmutación que está realizando, esto se determina a partir de la posición en la que se encuentra el MOSFET (M1) [1], es decir, en el convertidor chopper están presentes dos tipos de conmutación, inferior y superior. Siendo esta última el motivo por el cual muestra valores de voltaje de conmutación equivalentes a 180 V puesto que no está referenciado a tierra, a este tipo de conexión se denomina dispositivo flotado, en comparación con el MOSFET (M2) el cual si mantiene niveles de voltaje de conmutación con un máximo de 15 V ya que este si está referenciado a tierra.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Finalmente, se muestran los resultados de la simulación en lazo abierto del convertidor mostrado en la figura 3.6 mediante el bloque “Voltage & Current display”. Considerando la corriente de salida (I_o) y el voltaje de salida (V_o) como parámetros clave para el desarrollo del controlador y un voltaje de entrada (V_s) de 180 V tal como se muestra en la figura 5.2.

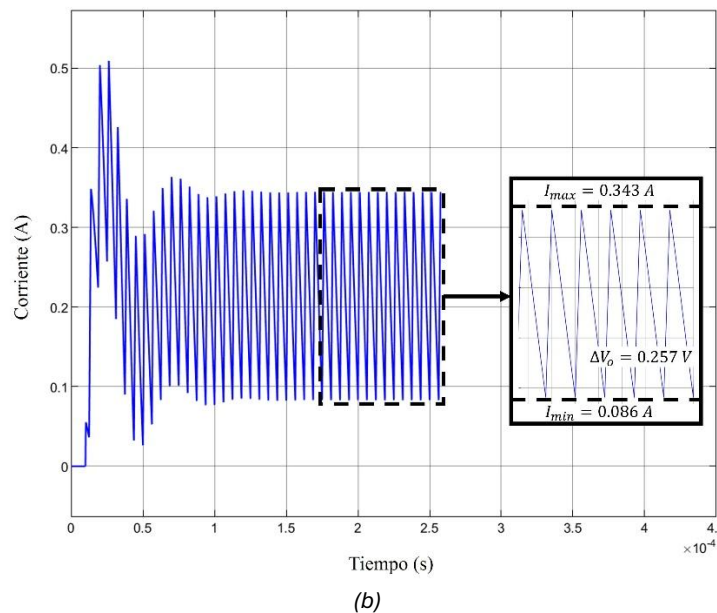
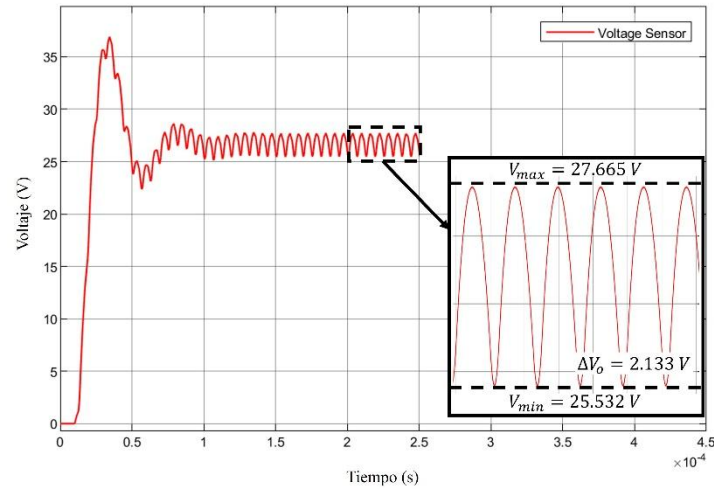


Figura 5.2 a) Voltaje de salida. b) Corriente de salida. Fuente: Elaboración propia

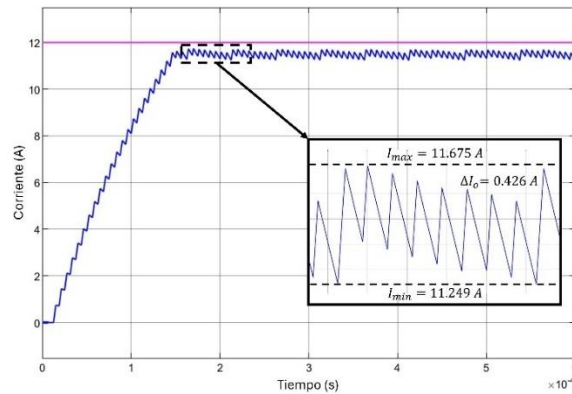
El motivo por el que se muestra un tiempo de simulación más corto en las figuras anteriormente mostradas es con el objetivo de mostrar de forma más detallada las formas de onda de cada uno de los parámetros.



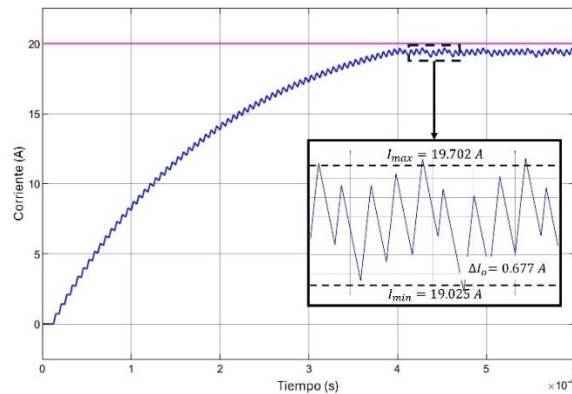
“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

5.2.1 Resultados de simulación del controlador MEF

En la figura 5.3 se muestra la corriente de salida I_o (Señal de color azul) y el valor “Set point” S_p (Señal de color violeta) en diferentes escenarios cambiando el valor del “Set point” de 12 A (ver figura 5.3.a) y 20 A (ver figura 5.3.b) utilizando la MEF, descrita en la figura 3.9, por lo que a partir de la simulación del sistema mostrado en la figura 3.10 en la figura 5.3.a la corriente de salida I_o intenta llegar al valor objetivo “Set point” S_p (señal de color violeta) sin embargo esto no es posible debido a los valores en los ciclos de trabajo establecidos del 60% al 70% sin valores intermedios lo cual indica que para alcanzar el valor “Set point” S_p debe establecerse el valor intermedio para igualar la corriente de salida con el valor objetivo “Set point” S_p . Por otra parte, en la figura 5.3.b se incrementa el tiempo de establecimiento debido a la limitación en el valor máximo del ciclo de trabajo. Como la máquina de estados solo tiene un valor de ciclo de trabajo máximo del 70% esto ocasiona que el tiempo de establecimiento incremente, para obtener un resultado mejor se tendría que incrementar el ciclo de trabajo.



(a)



(b)

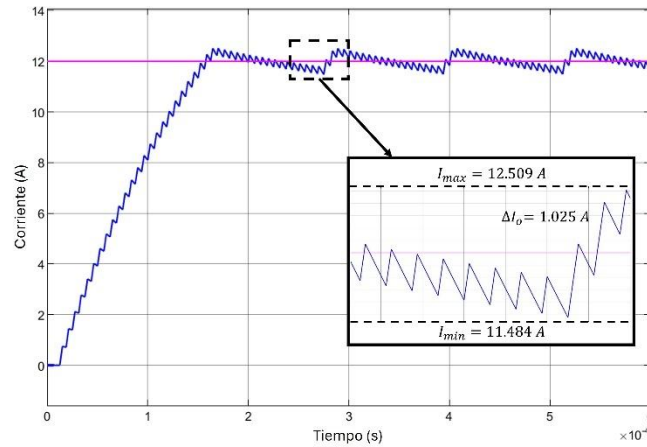
Figura 5. 3 (a) Corriente de salida del convertidor con Set point a 12 A con FSM (b) Corriente de salida del convertidor con Set point a 20 A con FSM Fuente: Elaboración propia en MATLAB / Simulink



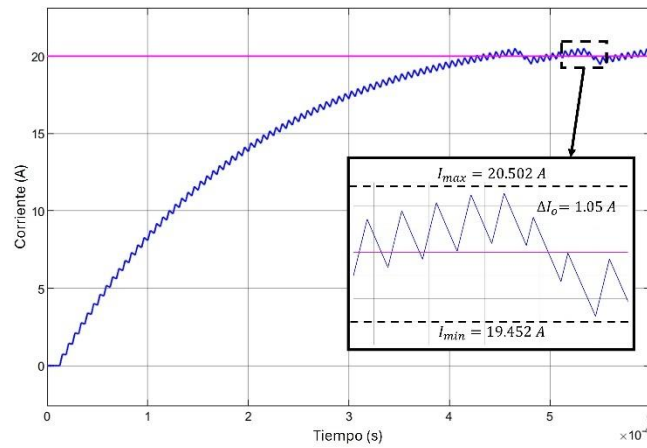
“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

5.2.2 Resultados de simulación del controlador MEF-PI

La señal de corriente de salida en los mismos escenarios anteriormente mencionados con un “Set point” de 12 y 20 A. Fueron realizados con el convertidor en conjunto con el controlador MEF-PI utilizando el esquema general mostrado en la figura 3.14, por lo que la corriente y voltaje de salida se muestran en la figura 5.4.



(a)



(b)

Figura 5. 4 (a) Corriente de salida del convertidor con FSM y PI con Set point a 12A (b) Corriente de salida del convertidor con FSM y PI con Set point a 20A. Fuente: Elaboración propia en MATLAB/Simulink

En la figura 5.4.a, se muestra que la corriente de salida del convertidor (Señal de color Azul) alcanza el valor establecido “Set point” (Señal de color rosa). Por otra parte, el tiempo de se vuelve más notoria la oscilación entre un límite superior e inferior previamente definidos por lo que se puede concluir que la máquina de estados en conjunto con el controlador PI, mantienen el valor de la corriente de salida en el valor establecido. En la figura 5.4.b donde el valor Set_point es de 20 A el tiempo de establecimiento incrementa, esto es porque el valor



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

del ciclo de trabajo definido en el estado “boost” es del 70% fijo e invariable, por lo que el sistema le toma más tiempo en alcanzar el valor de 12 A.

Con la finalidad de enfatizar el análisis entre ambos tipos de simulaciones (FSM y FSM + PI) se utilizarán los resultados obtenidos con valor del “Set point” de 12 A, por lo que los parámetros como el voltaje de salida V_o , el valor del error e , los ciclos de trabajo y la eficiencia eléctrica son parámetros importantes para dar seguimiento al análisis y evaluación del controlador (FSM + PI) por lo que a continuación se muestra una comparativa del comportamiento de cada uno de los parámetros anteriormente mencionados (ver figura 5.5).

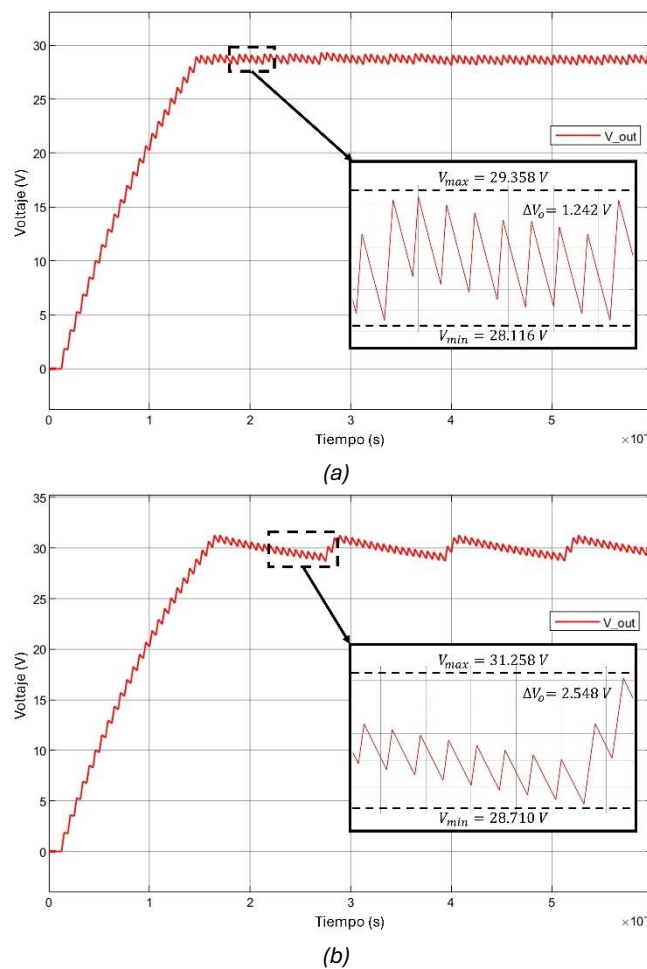


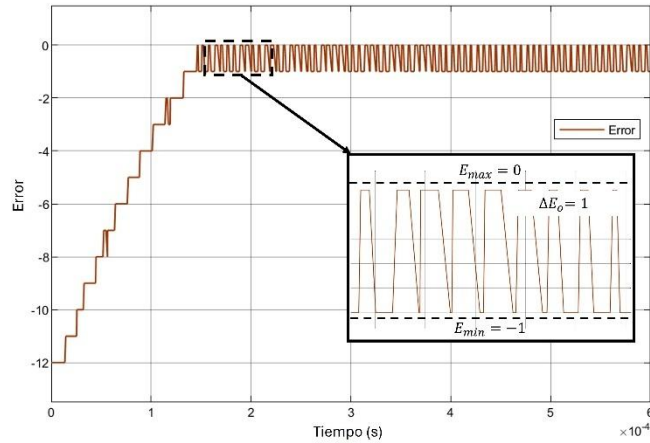
Figura 5. 5 (a) Voltaje de salida con MEF. (b) Voltaje de salida con MEF + PI

En la figura 5.5.a, existe una similitud en la forma de onda del voltaje de salida V_o y la corriente de salida I_o , de forma que se puede denotar la diferencia entre el valor objetivo y el valor al que se llega, esta diferencia se puede apreciar claramente en la figura 5.5.b, donde el controlador MEF en conjunto con el control PI si alcanza valores intermedios entre 30 V.

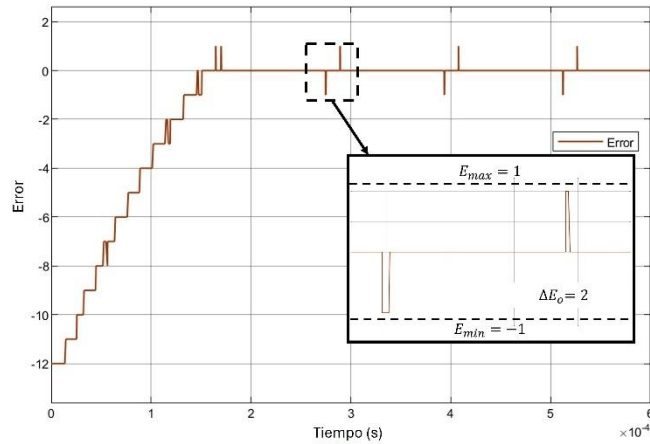


“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Del mismo modo en la figura 5.6 se muestra el comportamiento del error perteneciente a cada una de las versiones del controlador.



(a)



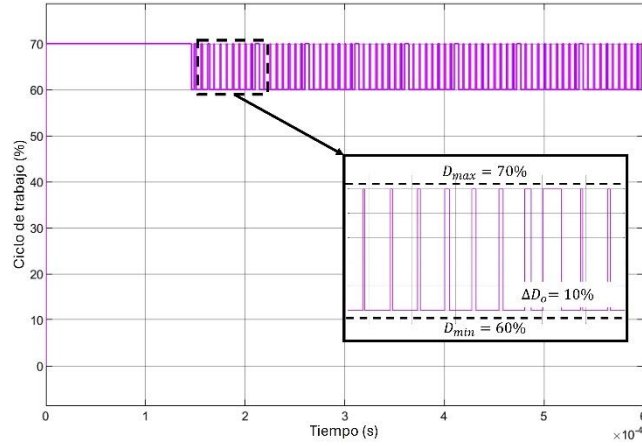
(b)

Figura 5. 6 (a) Error del controlador MEF. (b) Error controlador MEF + PI

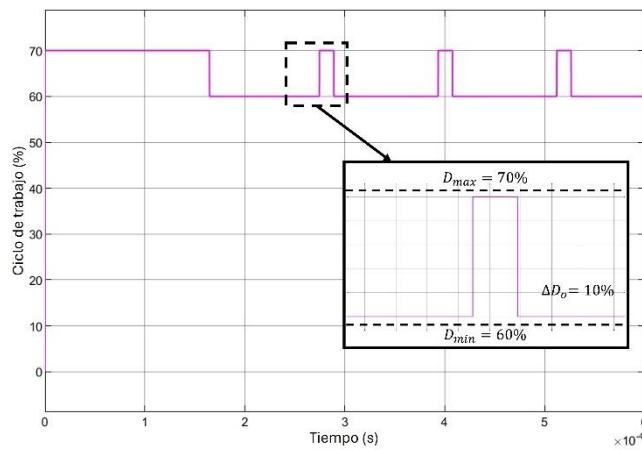
Tal como se muestra en la figura 5.6.a, donde el controlador utilizado es la MEF únicamente, el valor del error siempre es negativo y siempre se mantiene variante en el tiempo, esto se debe a la implementación de solo 3 estados en la MEF, considerando que en cada estado se hace un incremento en los ciclos de trabajo demasiado amplios, omitiendo valores intermedios. Sin embargo, en la figura 5.6.b el controlador implementado es la MEF con el controlador PI, donde en este caso si alcanza valores positivos y disminuye en mayor medida su comportamiento oscilatorio, esto se debe a que el controlador PI permite asignar valores intermedios en los ciclos de trabajo de los pulsos de conmutación.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”



(a)



(b)

Figura 5. 7 (a) Valor del ciclo de trabajo del controlador MEF. (b) Valor del ciclo de trabajo del controlador MEF+PI

El comportamiento de la señal en la figura 5.7 está relacionada con la figura 4.6, por lo cual se muestra el comportamiento del ciclo de trabajo de los pulsos de conmutación del controlador MEF en la figura 5.7.a donde se muestra un cambio demasiado rápido en los valores lo que lo convierte en un control inestable, por otra parte, el controlador MEF en conjunto con el control PI de la figura 5.7.b el valor de los ciclos de trabajo no muestra dicha oscilación, lo que lo convierte en un controlador más estable en comparación con el anterior.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

5.3 Resultados del sistema físico

Finalmente, se describe y se muestra la puesta en operación del sistema físico conectado se presenta en esta sección por medio de la figura 5.8. En esta se muestra: A) La interfaz gráfica para el monitoreo de cada uno de los parámetros que conforman el convertidor, B) el convertidor chopper con todos los subsistemas que lo conforman, C) Carga del convertidor D) fuentes de alimentación, E) dispositivo de medición con puntas para la medición del voltaje y corriente, F) fuente lineal de CD, G) Variac de energía. Por otra parte, el presente capítulo está conformado por el análisis de las señales del convertidor en lazo abierto, comenzando por las señales de conmutación generadas por el microcontrolador, la señales de salida de los optoacopladores, las señales de salida de los drivers o manejadores de los MOSFET, la señal de salida posterior al aislamiento galvánico, los pines compuerta – fuente y drenador – fuente de cada uno de los MOSFET respectivamente. Finalmente, se muestra la corriente y voltaje de salida del convertidor en lazo cerrado. Con la finalidad evaluar el comportamiento de la señales de forma real y aplicada partiendo como base las simulaciones realizadas previamente del convertidor.

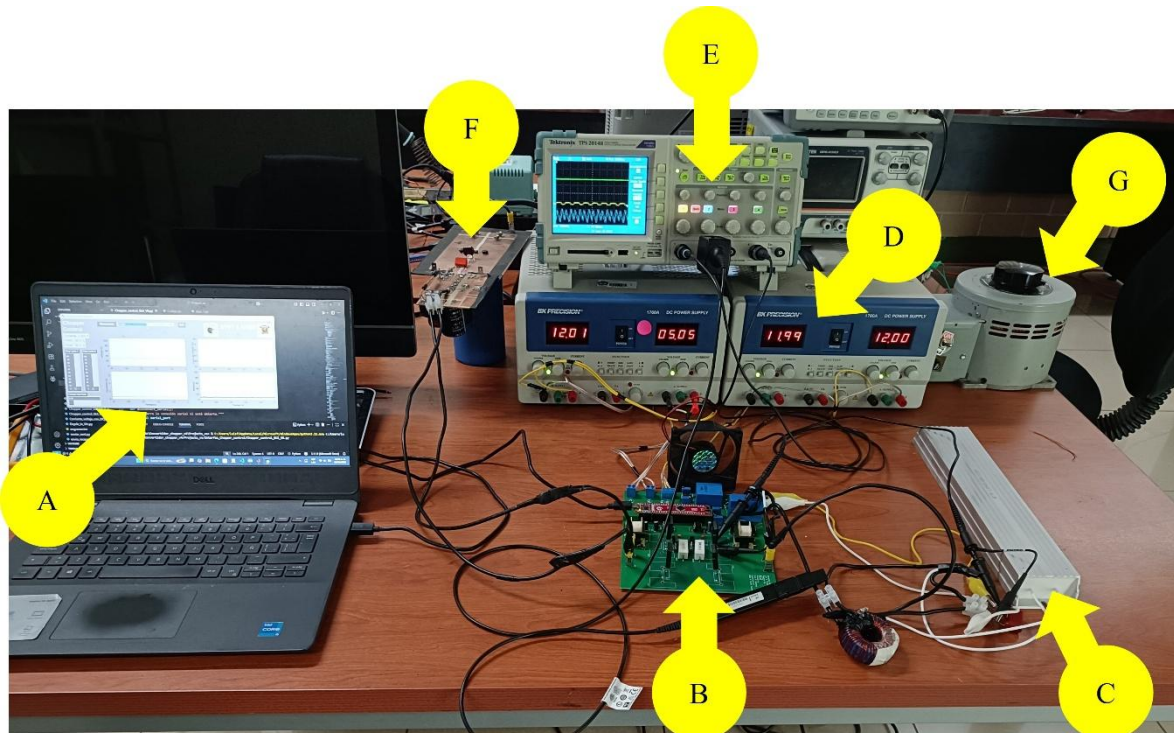


Figura 5. 8 Sistema físico conectado



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

5.3.1 Señales de conmutación del microcontrolador

Las señales que a continuación se muestran fueron obtenidas a partir de un osciloscopio mediante la implementación de las puntas para la medición del voltaje y la corriente para finalmente procesarlas y ajustarlas utilizando MATLAB. En la figura 5.9 se muestra las señales de conmutación generadas por el microcontrolador a la salida del microcontrolador de cada uno de los pines de salida, cumpliendo con las características principales como el desfase a 180° entre cada uno de los pulsos y un ciclo de trabajo mínimo del 60% para cada pulso de conmutación (PWM1 y PWM2).

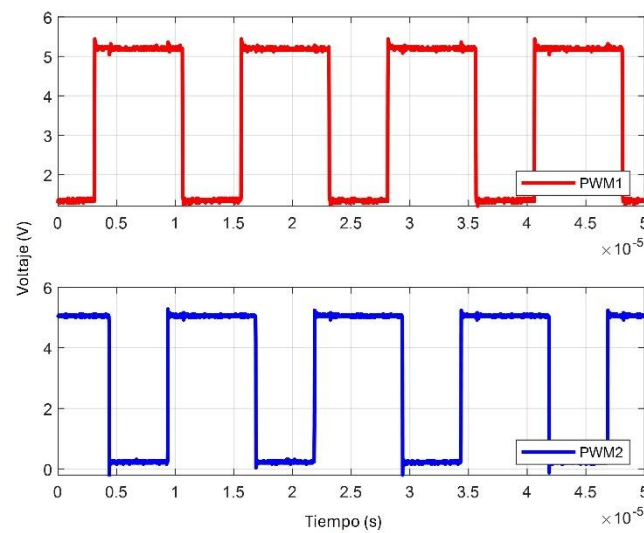


Figura 5. 9 (a) Señal de conmutación del microcontrolador PWM1. (b) Señal de conmutación del microcontrolador PWM2. Fuente: Elaboración propia

La figura 5.9 se puede apreciar que las señales de conmutación generadas por el microcontrolador son idénticas a las de la figura 3.1.b mostrada previamente, lo que garantiza que el convertidor opera en el primer cuadrante por lo cual, la corriente y el voltaje de salida será positivo. Por otra parte, es necesario mencionar que el valor del ciclo de trabajo actual para ambas señales de conmutación es del 60%.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

En la figura 5.10 se muestran las señales de conmutación obtenidas a la salida del controlador o *driver* de dispositivos MOSFET.

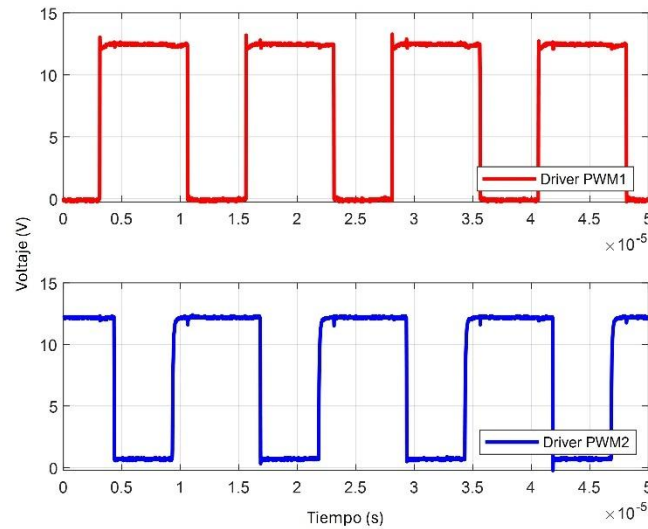


Figura 5. 10 Señales de conmutación a la salida del controlador. Fuente: Elaboración propia

La figura 5.10 se aprecia un incremento del voltaje 12 V esto indica que el subsistema está trabajando correctamente para adecuar la señal al valor indicado por el fabricante del MOSFET sin realizar cambios significativos con respecto a la señal de conmutación original

Posteriormente en la figura 5.11 se muestran las señales de conmutación a la salida del aislamiento galvánico.

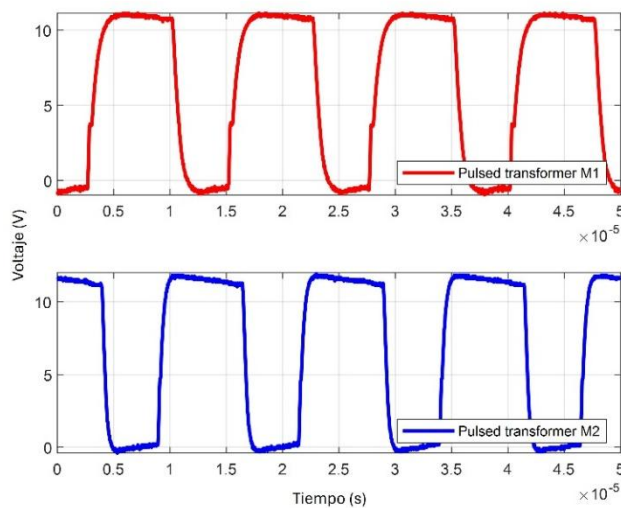


Figura 5. 11 Señal de conmutación PWM1 y PWM2 a la salida del aislamiento galvánico. Fuente: Elaboración propia.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Tal como se muestra en la figura 5.11, los pulsos de salida del transformador de pulsos son ligeramente distintos respecto a los pulsos de salida del controlador o *driver* del MOSFET, esto se debe principalmente a la presencia de los componentes pasivos que conforman el sistema del transformador de pulsos como resistencias y capacitores, en caso de que se requiera mejorar la fidelidad de la señal se requiere hacer un cálculo nuevamente modificando los parámetros de las fórmulas anteriormente mencionadas, sin embargo, es necesario mencionar que el valor obtenido debe coincidir con un valor comercial, es por ello que en este caso es un limitante para la mejora de la señal.

En la figura 5.12 se muestran los pulsos de conmutación pertenecientes a los MOSFETS inferior (M2) y superior (M1) en los pines compuerta-fuente respectivamente.

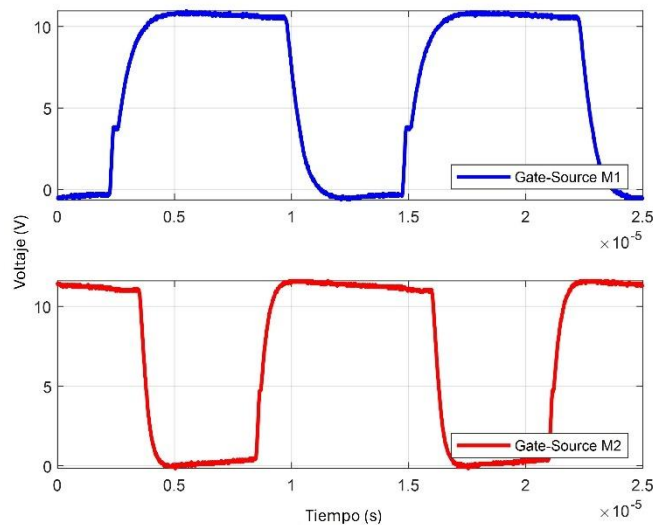


Figura 5. 12 Señal de conmutación en los pines Gate-Source de los MOSFET M1 y M2

Tal como se tenía contemplado previamente la señal de conmutación en los pines *Gate-Source* del MOSFET es idéntica a la señal de salida del transformador de pulsos, manteniendo el valor del tiempo de encendido como el de apagado en la señal de conmutación.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

En la figura 5.13 se muestran las señales de conmutación en los pines *Drain-Source* del MOSFET M1 y M2 respectivamente.

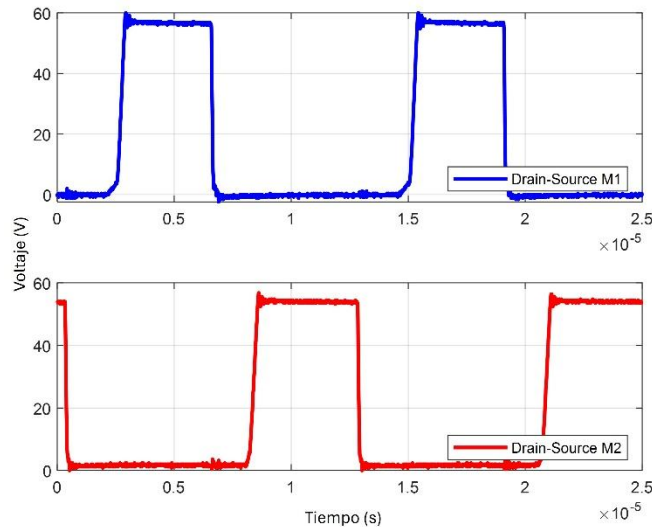


Figura 5. 13 Señal de conmutación en los pines *Drain-Source* de los MOSFETS M1 y M2

Tal como se muestra en la figura 5.13, la señal de los pulsos de conmutación está invertida si se realiza la comparativa con la figura 5.12, esto se debe a que la señal *Gate-Source* (V_{gs}) representa el pulso de control, el cual se muestra de forma inversa con respecto a la tensión obtenida respecto a los *Drain-Source* (V_{ds}) porque cuando el transistor está encendido (V_{gs} alto), la tensión en (V_{ds}) cae. Las señales mostradas en la figura 5.13

Partiendo del antecedente donde la señal de conmutación más importante de los MOSFETS que conforman el convertidor se encuentra en los pines *Drain-Source*, a continuación, se muestra la señal de conmutación de cada uno de los MOSFETS en un margen de tiempo más amplio en los pines anteriormente mencionados, así como también la corriente de salida del convertidor (ver figura 5.14).



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

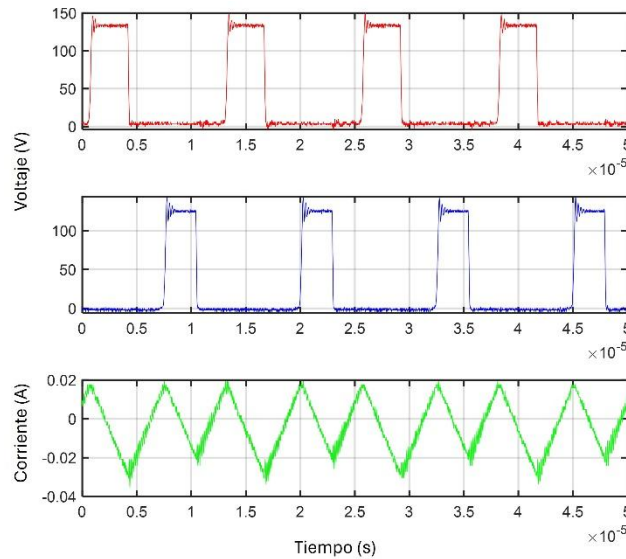


Figura 5. 14 (a) Pulsos de conmutación en los pines Drenador-Fuente de los MOSFETS (b) Corriente de salida del convertidor. Fuente: Elaboración propia, a partir de MATLAB

En la figura 514.se muestran en un inicio los pulsos de conmutación de los MOSFETS los cuales muestran un sobre impulso tolerable producto de la conexión de la carga al sistema. Por otra parte, se muestra la corriente de salida del convertidor, la cual tiene una forma de onda exactamente igual a la corriente de salida obtenida de la simulación.

5.3.2 Señales del convertidor a lazo abierto

Finalmente, como se ha mostrado en la literatura, se dice que un sistema es altamente eficiente cuando el valor de la eficiencia del convertidor de energía es del 100 %, es decir, la potencia de entrada es igual a la potencia de salida, Sin embargo, esto es en un escenario ideal donde no se consideran variables como pérdidas por conmutación o capacitancias e inductancias parasitas. Un ejemplo de escenario ideal son las simulaciones del convertidor realizadas previamente. Por lo cual, se muestran los resultados reales obtenidos del convertidor de energía y los resultados de eficiencia obtenidos a partir de la simulación ambos a lazo abierto (ver figura 5.15).



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

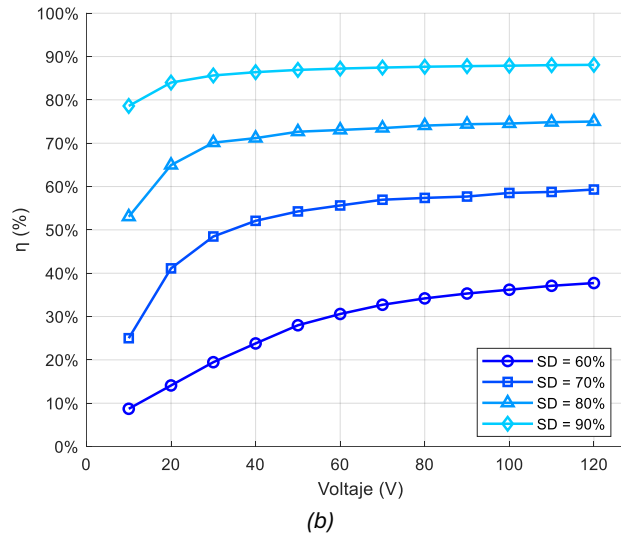
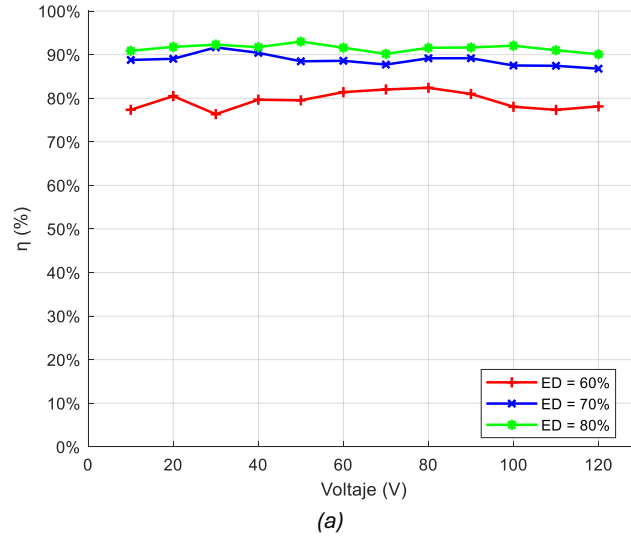


Figura 5. 15 (a) Eficiencias del convertidor a lazo reales. (b) Eficiencias del convertidor a lazo abierto simuladas. Fuente: Elaboración propia, a partir de MATLAB/Simulink®

En la figura 5.15.a se muestra el valor de la eficiencia experimental del convertidor, donde cada cambio en la eficiencia se ve afectado por el cambio del ciclo de trabajo, comenzando por un valor mínimo del 60 % e incrementando al 70 % y finalmente un máximo de 80 %, el motivo por el que no se realizaron pruebas experimentales con un ciclo de trabajo del 90 % es por motivos de seguridad y para la protección de cada uno de los componentes pasivos que conforman el convertidor. De manera que con un ciclo de trabajo del 80% (señal de color verde) se obtienen mejores valores de eficiencia confirmando que el diseño del prototipo implementado en una tarjeta PCB mejora significativamente las conexiones físicas entre cada uno de los componentes que conforman el convertidor, disminuyendo las capacitancias e inductancias parasitas significativamente.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Por otra parte, en la figura 5.15.b se muestran los valores de eficiencia obtenidos a partir de la simulación del convertidor *chopper* donde se aprecia que a partir de un ciclo de trabajo del 60 % se obtienen unos valores de eficiencia con un crecimiento estable y constante a medida que va incrementando el voltaje, en comparación con la figura 5.8.a donde se llegan a apreciar ciertas variaciones mínimas, el motivo se debe a que al ser una simulación se considera como un entorno ideal por lo cual no se consideran el ruido electrónico por conmutación o las capacitancias parasitas que puedan llegar a presentarse.

5.3.3 Señales del convertidor a lazo cerrado

Finalmente, para la evaluación del comportamiento del convertidor, se realizó un cambio de carga en la resistencia comenzando con un valor de resistencia inicial de 68Ω y posteriormente una de 10Ω . Sin embargo, para no realizar el cambio de forma manual e interferir en el comportamiento de la corriente de salida del convertidor se implementa un circuito de conmutación externo, conformado por un BJT (Transistor de Unión Bipolar, por su traducción al español) que polariza un relevador de 12 V. A partir de la hoja del fabricante [62].

Otro componente utilizado necesario para el circuito de cambio de carga es el relevador, con una capacidad de 12 V a 1 A donde se detalla la distribución de pines y las características eléctricas del mismo, a partir de [63]. La figura 5.16 se muestra el circuito eléctrico para realizar el cambio de resistencia en la carga del convertidor, esto con la finalidad de visualizar de forma más sencilla el comportamiento de la corriente de salida (I_o)



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

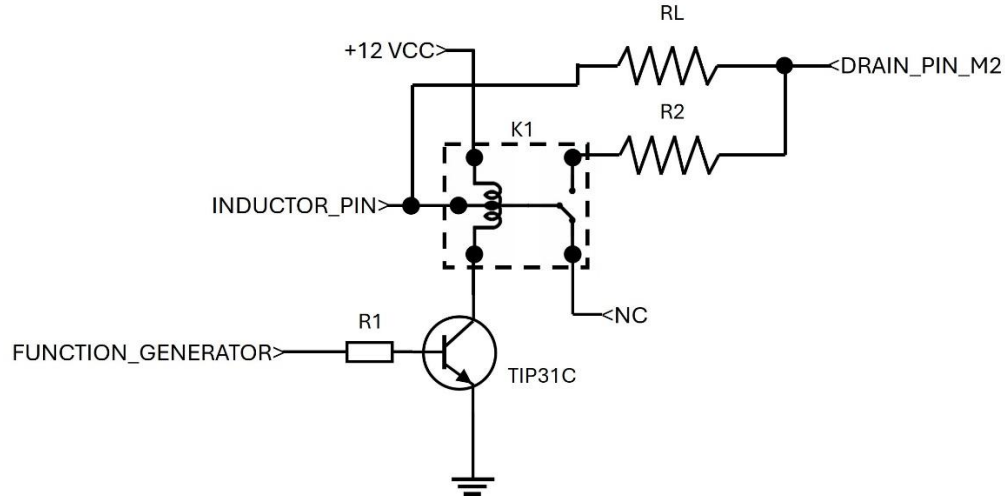
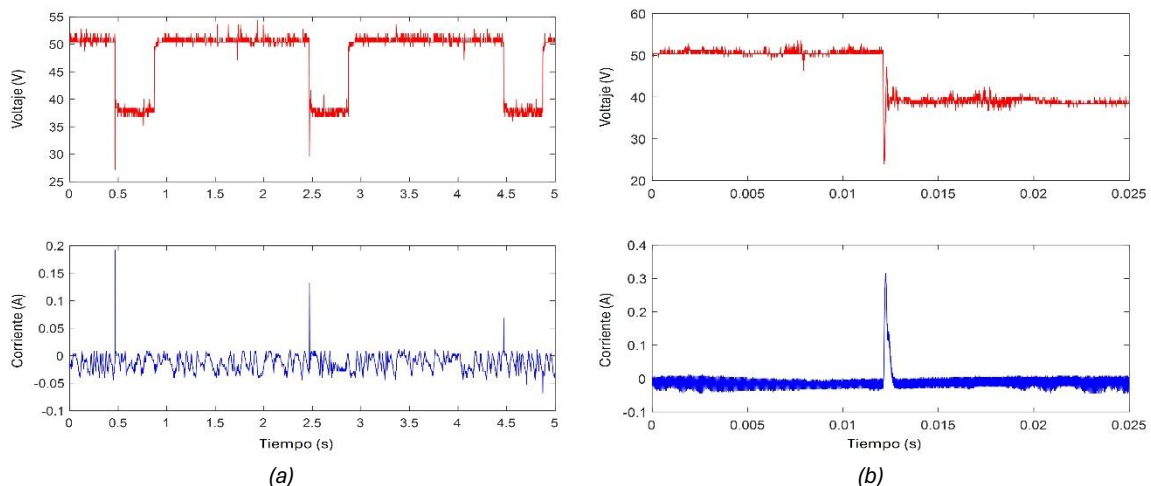


Figura 5. 16 Circuito eléctrico para el cambio de carga en el convertidor. Fuente: Elaboración propia

El circuito eléctrico mostrado en la figura 5.18 está conformado por una resistencia $R1$ de $1\text{ K}\Omega$ en el pin Base del transistor TIP31C esto con la finalidad de utilizarlo como un interruptor donde el pulso encargado de cerrar el circuito es generado mediante un generador de funciones para realizar ajustes precisos en los tiempos de activación y desactivación de 0.4 segundos, en un periodo total de 0.5 Hz, aplicado a la bobina que conforma el relevador K1. El cual, se conecta la resistencia de potencia $R2=10\ \Omega$ a 50 W, donde la conexión de la resistencia de potencia está conectada en paralelo con la resistencia original de $68\ \Omega$ a 200 W, al paralelo de R

Finalmente, en la figura 5.17 se muestra el voltaje y la corriente de salida del convertidor cuando la carga $RL=68\ \Omega$ a 200 W, se conecta en paralelo con $R2$, calculando una $RL//R2=8.71\ \Omega$.





“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

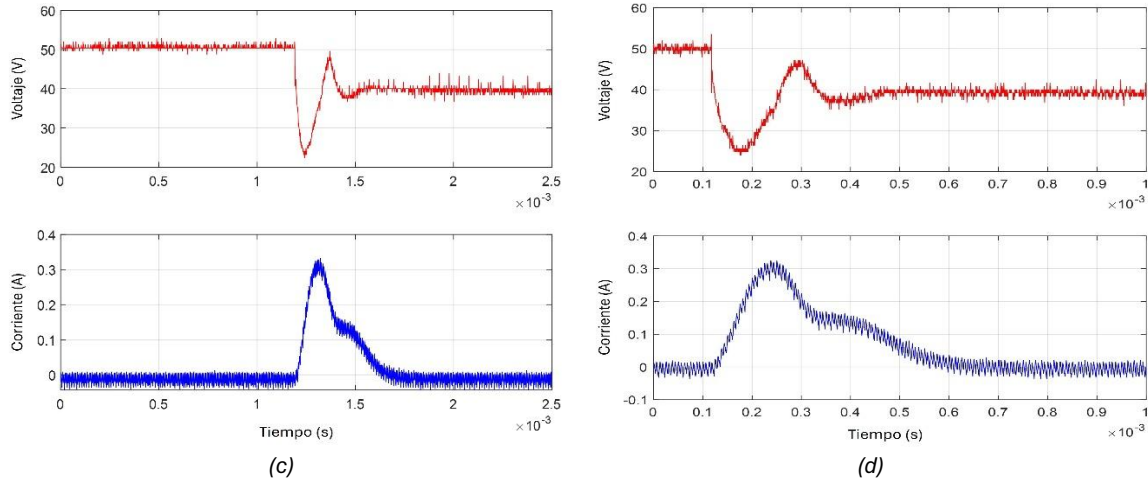


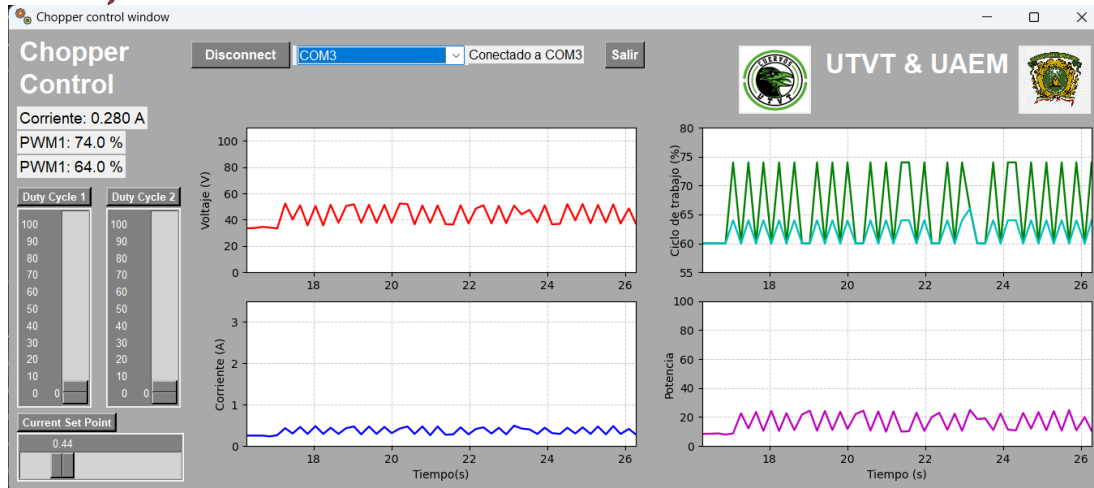
Figura 5. 17 Voltaje y corriente de salida del convertidor con cambio de carga. Fuente: Elaboración propia

En la figura 5.17.a se muestra el voltaje de salida del convertidor (señal de color rojo) y la corriente de salida (señal de color azul). Sin embargo, el voltaje de salida indica de forma visual la configuración del pulso ajustado en el generador de funciones el cual cuenta con un ciclo de trabajo del 20 % (0.4 segundos), esto es debido a que mientras que la corriente de salida aumenta el voltaje disminuye lo cual se observa de forma más sencilla en la figura 5.17.b sin embargo, este valor solo se visualiza como un sobre impulso, la forma de onda específica de la corriente de salida con el sobre impulso se puede apreciar en las figuras 5.17.c y 5.17.d. donde el valor pico máximo de corriente fue de 0.375 mA.

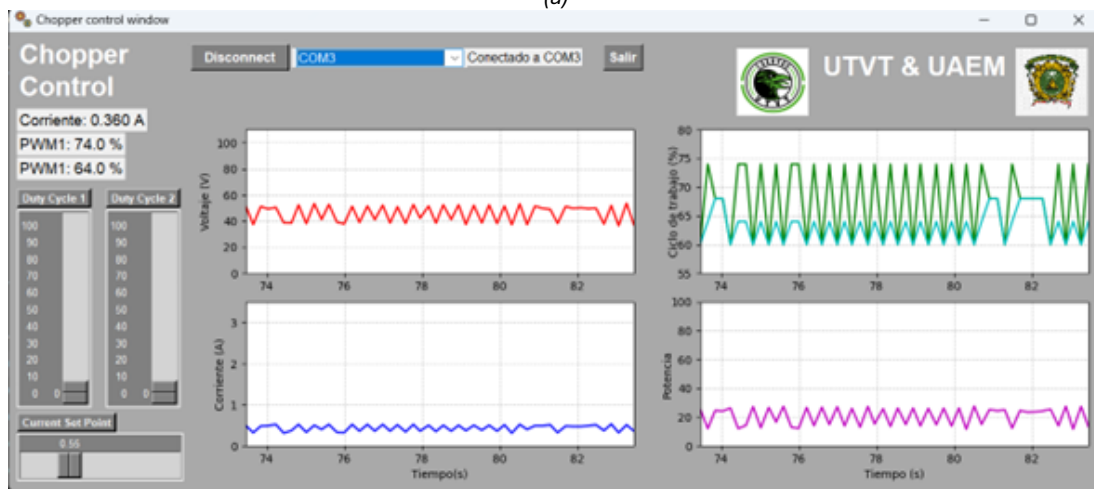
Finalmente, se realizó la comparativa entre la MEF y la MEF+PI para evaluar el comportamiento de este último. Ambas pruebas al igual que en la fase de simulación se realizaron variando el valor del Set_point en la interfaz gráfica de monitorización de parámetros. El comportamiento de la MEF simple con diferentes valores en el Set point se muestra en la figura 5.18.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”



(a)



(b)

Figura 5. 18 Parámetros de salida del convertidor con MEF únicamente variando el valor del Set Point. Fuente: Elaboración propia

La figura 5.18.a, tiene un valor Set point de 440 mA y la figura 5.18.b un valor Set point de 550 mA, comenzando por la figura 5.18.a se observa que las transiciones en los valores de los ciclos de trabajo de los pulsos de conmutación son muy abruptas y se mantienen oscilando, lo que en consecuencia termina afectando en la señal de control, es decir, la corriente de salida lo que dificulta que la máquina de estados pueda alcanzar el valor Set point establecido, por otra parte en la figura 5.18.b pese a que se realizó un incremento en el valor Set point de 0.100 mA el comportamiento de los pulsos de conmutación y de la MEF simple se mantiene oscilando debido a que solo existen dos estados definidos lo que justifica la ausencia de valores intermedios e incluso en cierto punto la placa de desarrollo termina asignando un valor no definido por los cambios recursivos en las condiciones de transición, por lo cual se presenta una señal de corriente oscilante en ambas situaciones, tal como se



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

mostró en la simulación del convertidor, el comportamiento de la corriente de salida es muy similar a la corriente obtenida a partir de la interfaz de parámetros. Por lo cual es necesario realizar una mejora en el sistema de pruebas actual.

En la figura 5.19 se muestra el comportamiento de la MEF con el controlador PI cambiando el valor Set point de 0.480 mA y de 0.550 mA con el objetivo de analizar el comportamiento de los parámetros en la interfaz gráfica de monitoreo.

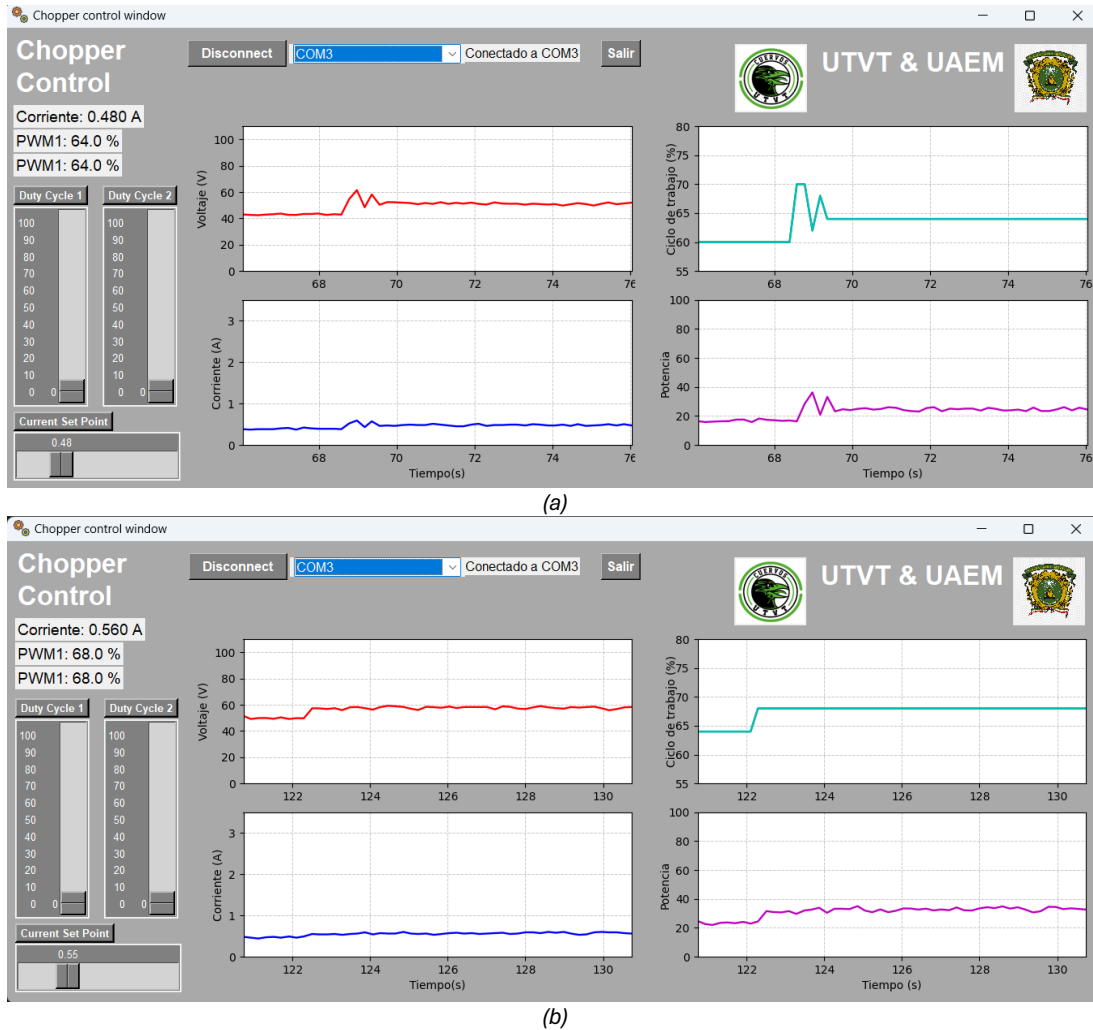


Figura 5. 19 Parámetros de salida del convertidor con MEF y PI. Fuente: Elaboración propia

La figura 5.19.a, tiene un valor objetivo de 480 mA y la figura 5.19.b un valor objetivo de 550 mA, donde en la figura 5.19.a se muestra un sobre impulso y un ajuste inicial elevado en el ciclo de trabajo de los pulsos de conmutación, esto se debe principalmente a la acción del estado boost de la máquina de estados, donde solo entrara en acción cuando el valor del error sea demasiado grande. Sin embargo, dicho sobre impulso no es demasiado prologando ya



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

que la máquina de estados finitos cuenta con un valor mínimo en los ciclos de trabajo lo que facilita que la corriente de salida del convertidor disminuya significativamente, sin embargo, para evitar que los ciclos de trabajo oscilen demasiado con respecto a la figura 5.18.a se implementa un controlador pi para estabilizar y determinar un valor intermedio en los pulso de conmutación.

En el caso de la figura 5.19.b el cambio del ciclo de trabajo es menos brusco y al contrario que la figura 5.18 no presenta oscilaciones, sino al contrario el cambio lo realiza de forma constante y estable tanto en los ciclos de trabajo como en la corriente de salida por lo que la corriente de salida alcanza el valor objetivo y lo mantiene dentro de los límites superior e inferior establecidos en la programación, de forma que se puede apreciar que la MEF en conjunto con el controlador PI realiza el ajuste apropiado en los ciclos de trabajo de los pulsos de conmutación.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Conclusión

Se determinó viable implementar una máquina de estados finitos en un sistema embebido basado en un microcontrolador que cuente con la tecnología CIP (*Core Independent Peripherals*) para reducir el tiempo de procesamiento y optimizar la ejecución de cada una de las acciones que corresponden a los estados que conformen la máquina. El convertidor estático de energía de tipo *chopper* de dos cuadrantes en modo corriente opera de manera correcta con los MOSFETs seleccionados debido a sus características eléctricas, incluyendo el diseño de las redes auxiliares de conmutación *Snubber* para la reducción del dv/dt . La implementación de un filtro digital de tipo pasa bajo calculado a frecuencia de corte $f_c = 90 \text{ khz}$ para filtrar las señales de alta frecuencia, resultó satisfactorio para que la señal de control que ingresa al microcontrolador fuera estable.

Como parte de la metodología de este trabajo se desarrolló de una simulación utilizando la biblioteca Simscape de MATLAB para el análisis de circuitos eléctricos en electrónica de potencia. Esto debido a que permite configurar los dispositivos con parámetros de sus hojas de datos, en comparación de la librería Simulink que está orientada en el análisis gráfico y matemático de los sistemas control. Por otra parte, se utilizó la librería Stateflow debido a que tiene un enfoque para el diseño y desarrollo de máquinas de estado aplicadas a sistemas de todo tipo de complejidad, ya que cuenta con las herramientas suficientes para cubrir cada una de las áreas que conforman MATLAB. Cabe señalar que estas librerías interactuaron en una sola simulación para el desarrollo de este trabajo de tesis.

Se diseñó un prototipo en una sola placa impresa o PCB desarrollada con normas establecidas como la NOM-001-SCFI-2018. Lo anterior utilizando el software de uso libre KiCad para el diseño de circuitos eléctricos y electrónico, el cual cuenta con la documentación necesaria para facilitar el aprendizaje y desarrollar circuito. Así, se logró un diseño para aislar cada una de las etapas que conforman el sistema, utilizar diferentes capas de cobre para disminuir los armónicos y las capacitancias parasitas mediante principios eléctricos.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Estos resultados se observaron en el aislamiento galvánico utilizando transformadores de pulsos, por lo tanto, estos son una solución sólida para la conmutación de MOSFETS a alta frecuencia (80 kHz) realizado el cálculo correspondiente de los valores para el circuito. De igual manera el uso de la placa de desarrollo Curiosity® de Microchip® resultó una solución asequible y de fácil implementación para sistemas de control como el desarrollado en este trabajo. Fue así como la implementación de la máquina de estados finitos implementada en este hardware proporcionó una solución cuando la dinámica del sistema es no lineal como es el caso de este tipo de convertidor de energía.

Finalmente, se desarrolló una interfaz gráfica de usuario en lenguaje de programación Python resultando ser una opción sólida debido a que cuenta con las herramientas necesarias y su fácil entendimiento y manejo por el usuario. La amplia documentación y librerías de Python fue fundamental para el desarrollo de una interfaz de usuario que le permite fijar variables de punto de operación de corriente, ciclos de trabajo, y frecuencia. Además de observar en tiempos aceptables el comportamiento de voltaje, corriente, potencia y variaciones del ciclo de trabajo de los interruptores MOSFET.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Trabajo a futuro

Como trabajo a futuro se propone un análisis extenso comparando diferentes controladores tales como, PID, lógica difusa o de tipo predictivo, inclusive los nombrados controladores inteligentes basados en el uso de la inteligencia artificial (IA). Por otra parte, se recomienda utilizar la NOM-001-SCFI-2018 para mejorar el diseño de la PCB y reducir en un porcentaje más alto el ruido eléctrico provocado por sistemas de alta frecuencia, como es el caso del convertidor desarrollado. A través de tecnología de última generación como la basada en GaN y SiC, es posible incrementar la energía y la eficiencia eléctrica hasta el 95 %, además de reducir las dimensiones del convertidor.

Finalmente, en parte computacional, se recomienda hacer una interfaz que pueda estar en línea con el usuario de manera remota, permitiendo evaluar el comportamiento del convertidor en tiempo real.



Referencias

- [1] B. W. Williams, *Power Electronics: Devices, Drivers and Applications*. 1987.
- [2] V. Rueda, A. Wiesner, R. Diez, and H. Piquet, “Power estimation of a current supplied DBD considering the transformer parasitic elements,” *IEEE Transactions on Industry Applications*, vol. 55, no. 6, pp. 6567–6575, Aug. 2019, doi: 10.1109/tia.2019.2933519.
- [3] «Soft-switching modulation for boost full bridge converter with active clamp circuit», *IEEE Conference Publication | IEEE Xplore*, 1 de Junio de 2020. <https://ieeexplore.ieee.org/document/9161891> (Accedido 24 de septiembre de 2025).
- [4] «Finite State Machine-Based Realization of Sparse Matrix Converter», *IEEE Journals & Magazine | IEEE Xplore*, 1 de abril de 2021. <https://ieeexplore.ieee.org/document/9324822?denied=> (accedido 24 de septiembre de 2025).
- [5] «Finite-state-machine model of boundary control for dual-active-bridge converter», *IEEE Conference Publication | IEEE Xplore*, 1 de agosto de 2017. <https://ieeexplore.ieee.org/document/8056297> (accedido 24 de septiembre de 2025).
- [6] L. N. Tutelea, S. I. Deaconu, y G. N. Popa, «FEM analysis of an single stator dual PM rotors axial synchronous machine», *IOP Conference Series Materials Science And Engineering*, vol. 163, p. 012006, ene. 2017, doi: 10.1088/1757-899x/163/1/012006.
- [7] N. Monterrosa, J. Montoya, F. Jarquin, and C. Bran, “Design, development and implementation of a UAV flight controller based on a state machine approach using a FPGA embedded system,” *2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC)*, pp. 1–8, Sep. 2016, doi: 10.1109/dasc.2016.7778069.
- [8] A. Tsoupos, V. Khadkikar, and P. Marpu, “Finite State Machine-Based Realization of sparse matrix Converter,” *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, vol. 2, no. 2, pp. 196–204, Jan. 2021, doi: 10.1109/jestie.2021.3051575.
- [9] N. S. Kumar, V. Sadasivam, and K. Prema, “Design and simulation of fuzzy controller for closed loop control of chopper fed embedded DC drives,” *2021 International Conference on Power System Technology (POWERCON)*, vol. 1, pp. 613–617, Jul. 2005, doi: 10.1109/icpst.2004.1460067.
- [10] M.-K. Nguyen, “Power Converters in Power Electronics: Current Research Trends,” *Electronics*, vol. 9, no. 4, p. 654, Apr. 2020, doi: 10.3390/electronics9040654.
- [11] S. Alatai et al., “A review on State-of-the-Art Power Converters: bidirectional, resonant, multilevel converters and their derivatives,” *Applied Sciences*, vol. 11, no. 21, p. 10172, Oct. 2021, doi: 10.3390/app112110172.
- [12] M. Taleb, N. Mansour, and K. Zehar, “Performance of a grid-connected solar PV system using class D chopper,” *IET Conference Proceedings.*, vol. 2022, no. 26, pp. 180–186, Feb. 2023, doi: 10.1049/icp.2023.0396.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

- [13] K. Akter, S. M. A. Motakabber, A. H. M. Z. Alam, S. H. Yousoff, S. Ahmed, and T. Annur, “Performance inspection of high gain chopper designed to extract optimum output of photovoltaic source,” *International Journal of Power Electronics and Drive Systems/International Journal of Electrical and Computer Engineering*, vol. 14, no. 4, p. 2204, Sep. 2023, doi: 10.11591/ijpeds.v14.i4.pp2204-2216.
- [14] R. A. Ahmed and E. D. Hassan, “Modeling, Design and Control of Speed DC motor using chopper,” *ijeer.forexjournal.co.in*, Dec. 2023, doi: 10.37391/IJEER.110437.
- [15] H. Zhang, M. Berkhout, K. a. A. Makinwa, and Q. Fan, “A 120.9-dB DR Digital-Input capacitively coupled Chopper Class-D audio amplifier,” *IEEE Journal of Solid-State Circuits*, vol. 58, no. 12, pp. 3470–3480, Oct. 2023, doi: 10.1109/jssc.2023.3318731.
- [16] H. Li and K. Morisaki, “Development of 500kW class PCS for hydrogen-powered fuel cell applications,” *2022 International Power Electronics Conference (IPEC-Himeji 2022- ECCE Asia)*, pp. 770–774, May 2022, doi: 10.23919/ipechimeji2022-ecce53331.2022.9807081.
- [17] M. H. Rashid, *Power Electronics Handbook*. 1990.
- [18] M. H. C. Panayiotou, “WIDE-BANDGAP SEMICONDUCTORS Gallium Nitride, GAN - Silicon Carbide, SIC,” Jan. 01, 2022. <https://par.nsf.gov/biblio/10343252> (accedido Jul. 16, 2025).
- [19] A. M. S. Al-Bayati, S. S. Alharbi, S. S. Alharbi, and M. Matin, “A comparative design and performance study of a non-isolated DC-DC buck converter based on Si-MOSFET/Si-Diode, SiC-JFET/SiC-schottky diode, and GaN-transistor/SiC-Schottky diode power devices,” *2021 North American Power Symposium (NAPS)*, pp. 1–6, Sep. 2017, doi: 10.1109/naps.2017.8107192.
- [20] T. Avraham, M. Dhyani, and J. B. Bernstein, “Reliability challenges, models, and physics of silicon carbide and gallium nitride power devices,” *Energies*, vol. 18, no. 5, p. 1046, Feb. 2025, doi: 10.3390/en18051046.
- [21] H. Chun et al., “Converter switching noise reduction for enhancing EMC performance in HEV and EV,” *PCIM Europe 2016; International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management; Proceedings Of*, pp. 1–8, May 2016, [Online]. Disponible en: <https://ieeexplore.ieee.org/document/7499499/>
- [22] N. J.-Y. Chai, N. Y.-H. Ho, N. Y.-C. Chang, and N. C.-M. Liaw, “On Acoustic-Noise-Reduction control using random switching technique for Switch-Mode rectifiers in PMSM Drive,” *IEEE Transactions on Industrial Electronics*, vol. 55, no. 3, pp. 1295–1309, Mar. 2008, doi: 10.1109/tie.2007.909759.
- [23] A. Elasser and D. A. Torrey, “Soft switching active snubbers for DC/DC converters,” *IEEE Transactions on Power Electronics*, vol. 11, no. 5, pp. 710–722, Sep. 1996, doi: 10.1109/63.535403.
- [24] S. Havanur, “SNUBBER DESIGN FOR NOISE REDUCTION IN SWITCHING CIRCUITS,” Apr. 2007. [Online]. Disponible en: <https://api.eestar.com/article/share/2021/01/46267202101311316245516.pdf>
- [25] D. Westenskow, “Fundamentals of feedback control: PID, fuzzy logic, and neural networks,” *Journal of Clinical Anesthesia*, vol. 9, no. 6, pp. 33S-35S, Sep. 1997, doi: 10.1016/s0952-8180(97)00123-2.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

- [26] P. Warriar and P. Shah, “Optimal Fractional PID Controller for Buck Converter using cohort intelligent algorithm,” *Applied System Innovation*, vol. 4, no. 3, p. 50, Aug. 2021, doi: 10.3390/asi4030050.
- [27] S. V. Devaraj et al., “Robust Queen Bee Assisted Genetic Algorithm (QBGA) optimized Fractional Order PID (FOPID) controller for not necessarily minimum phase power converters,” *IEEE Access*, vol. 9, pp. 93331–93337, Jan. 2021, doi: 10.1109/access.2021.3092215.
- [28] K. Ogata, *Modern Control Engineering*. 1990.
- [29] W. Zhang, Y. Xi, G. Yang, and X. Xu, “Design PID controllers for desired time-domain or frequency-domain response,” *ISA Transactions*, vol. 41, no. 4, pp. 511–520, Oct. 2002, doi: 10.1016/s0019-0578(07)60106-2.
- [30] E. Moxnes and P. I. Davidsen, “Intuitive understanding of steady-state and transient behaviors,” *System Dynamics Review*, vol. 32, no. 2, pp. 130–155, Apr. 2016, doi: 10.1002/sdr.1561.
- [31] F. Merrikh-Bayat, N. Mirebrahimi, and M. R. Khalili, “Discrete-time fractional-order PID controller: Definition, tuning, digital realization and some applications,” *International Journal of Control Automation and Systems*, vol. 13, no. 1, pp. 81–90, Dec. 2014, doi: 10.1007/s12555-013-0335-y.
- [32] A. Y. Jaen-Cuellar, R. De J Romero-Troncoso, L. Morales-Velazquez, and R. A. Osornio-Rios, “PID-Controller Tuning Optimization with Genetic Algorithms in Servo Systems,” *International Journal of Advanced Robotic Systems*, vol. 10, no. 9, Sep. 2013, doi: 10.5772/56697.
- [33] L. C. Borin, E. Mattos, C. R. D. Osorio, G. G. Koch, and V. F. Montagner, “Robust PID Controllers Optimized by PSO Algorithm for Power Converters,” *IEEE*, pp. 1–6, Dec. 2019, doi: 10.1109/cobep/spec44138.2019.9065642.
- [34] C. W. De Silva, *Intelligent Control*. 2018. doi: 10.1201/9780203750513.
- [35] T. Ahmmed, I. Akhter, S. M. R. Karim, and F. a. S. Ahamed, “Genetic Algorithm based PID Parameter Optimization,” *American Journal of Intelligent Systems*, vol. 10, no. 1, pp. 8–13, Dec. 2020, doi: 10.5923/j.ajis.20201001.02.
- [36] F. Behrooz, N. Mariun, M. Marhaban, M. M. Radzi, and A. Ramli, “Review of Control Techniques for HVAC Systems—Nonlinearity Approaches based on Fuzzy Cognitive Maps,” *Energies*, vol. 11, no. 3, p. 495, Feb. 2018, doi: 10.3390/en11030495.
- [37] Y. Yan, P. Xu, J. Yue, and Z. Chen, “Robust control: from Continuous-State systems to finite state machines,” *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 2, pp. 2156–2163, Feb. 2024, doi: 10.1109/tase.2024.3362975.
- [38] D. M. Ballesteros, “MODULACIÓN PWM EN FPGA BASADO EN MÁQUINAS DE ESTADO FINITO,” *Redalyc.org*, 2008. <https://www.redalyc.org/articulo.oa?id=84903874> (accedido en Jul. 23, 2025).
- [39] A. S. Klimovich and V. V. Solov’ev, “Transformation of a mealy finite-state machine into a moore finite-state machine by splitting internal states,” *Journal of Computer and Systems Sciences International*, vol. 49, no. 6, pp. 900–908, Dec. 2010, doi: 10.1134/s1064230710060080.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

- [40] Y. Zhou, “A summary of PID control algorithms based on AI-Enabled embedded systems,” *Security and Communication Networks*, vol. 2022, pp. 1–7, Apr. 2022, doi: 10.1155/2022/7156713.
- [41] C. Contreras, J. C. Quirós, I. Casaucao, A. Triviño, E. Villagrasa, and J. A. Aguado, “STATCOM switching technique based on a Finite-State machine,” *Electronics*, vol. 12, no. 6, p. 1481, Mar. 2023, doi: 10.3390/electronics12061481.
- [42] A. R. A. Tahtawi, Y. Somantri, and E. Haritman, “Design and implementation of PID control-based FSM algorithm on line following robot,” *JTERA (Jurnal Teknologi Rekayasa)*, vol. 1, no. 1, p. 23, Jan. 2017, doi: 10.31544/jtera.v1.i1.2016.23-30.
- [43] B. P. McGrath, T. Meynard, G. Gateau, and D. G. Holmes, “Optimal modulation of flying capacitor and stacked multicell converters using a state machine decoder,” *IEEE Transactions on Power Electronics*, vol. 22, no. 2, pp. 508–516, Mar. 2007, doi: 10.1109/tpel.2006.889932.
- [44] V. K. Awaar et al., “IoT enabled speed control of Single-Phase induction motor,” *E3S Web of Conferences*, vol. 391, p. 01175, Jan. 2023, doi: 10.1051/e3sconf/202339101175.
- [45] “PIC18-Q10 Microcontrollers (MCUs),” *Microchip Technology*. <https://www.microchip.com/en-us/products/microcontrollers/8-bit-mcus/pic-mcus/pic18-q10> (accessed Jul. 23, 2025).
- [46] S. Katzen, *The essential PIC18® microcontroller*. 2010. doi: 10.1007/978-1-84996-229-2.
- [47] I. Mezzah, H. Chemali, S. Mezzah, O. Kermia, and O. Abdelmalek, “MCIP: High configurable 8-bit microcontroller IP-core,” *IEEE*, pp. 1387–1390, Jul. 2015, doi: 10.1109/sai.2015.7237324.
- [48] M. O. Alsumady, Y. K. Alturk, A. Dagamseh, and M. Tantawi, “Controlling of DC-DC Buck converters using microcontrollers,” *International Journal of Circuits Systems and Signal Processing*, vol. 15, pp. 197–202, Mar. 2021, doi: 10.46300/9106.2021.15.22.
- [49] N. Z. Yong-Bin, N. J. Fang, N. L. Guang-Min, and N. Z. Lian-Xin, “Communication between industrial computer and micro-energy pulse power supply with nanosecond pulse width,” *IEEE*, pp. 413–417, Jan. 2010, doi: 10.1049/cp.2010.1334.
- [50] “Rosenbrock Function,” *Rosenbrock Function - File Exchange - MATLAB CentralFile Exchange - MATLAB Central*, May 29, 2012. <https://www.mathworks.com/matlabcentral/fileexchange/36883-rosenbrock-function> (accessed Jul. 23, 2025).
- [51] *alldatasheet.com*, “APT6030BVR PDF.” <https://www.alldatasheet.com/datasheet-pdf/view/49542/ADPOW/APT6030BVR.html> (accessed Jul. 23, 2025).
- [52] “Example List - MATLAB & Simulink.” https://la.mathworks.com/help/sps/examples.html?s_tid=CRUX_topnav (accessed Jul. 23, 2025).
- [53] *alldatasheet.com*, «6N137 PDF». <https://www.alldatasheet.com/datasheet-pdf/view/30862/TOSHIBA/6N137.html> (accedido 12 de octubre de 2024).
- [54] *Microchip Technology Inc.*, «TC4421/TC4422», 2013. [En línea]. Disponible en: <https://ww1.microchip.com/downloads/en/DeviceDoc/20001420F.pdf>



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

- [55] L. Balogh y Texas Instruments Incorporated, «Fundamentals of MOSFET and IGBT Gate Driver Circuits», 2018. [En línea]. Disponible en: <https://www.ti.com/lit/ml/slua679b/slua679b.pdf>
- [56] Coilcraft Inc., «Base/Gate Driver transformers», 2023. [En línea]. Disponible en: <https://www.coilcraft.com/getmedia/0ed4623a>
- [57] APT, «TO-247», technical report, feb. 2007. [En línea]. Disponible en: [https://www.microchip.com/documents/PSDS/ProductDocuments/DataSheets/AP T15DS60B_S\(G\)_C.pdf](https://www.microchip.com/documents/PSDS/ProductDocuments/DataSheets/AP T15DS60B_S(G)_C.pdf)
- [58] LEM International SA, «Current transducer LA 25-P», report, 2023. [En línea]. Disponible en: https://www.lem.com/products_datasheets/la_25-p_v6.pdf
- [59] LEM International SA, «Voltage Transducer LV 25-P», 2022. [En línea]. Disponible en: https://www.lem.com/products_datasheets/lv_25-p_v21.pdf
- [60] alldatasheet.com, «AD620AN PDF». <https://www.alldatasheet.com/datasheet-pdf/view/48091/AD/AD620AN.html> (accedido 22 de septiembre de 2024).
- [61] M. Ayaz, M. W. Ayub, y I. A. Qureshi, Arduino Based Fatigue Level Measurement in Muscular Activity using RMS Technique, 2021 International Conference On e-Health and Bioengineering (EHB), pp. 1-4, oct. 2020, doi: 10.1109/ehb50910.2020.9280184.
- [62] alldatasheet.com, «TIP31C PDF». <https://www.alldatasheet.com/datasheet-pdf/view/2779/MOSPEC/TIP31C.html> (accedido 17 de septiembre de 2024).
- [63] alldatasheet.com, «SRD-12VDC-SL-C PDF». <https://www.alldatasheet.com/datasheet/view/1131793/SONGLERELAY/SRD-12VDC-SL-C.html> (accedido 24 de septiembre de 2025).



ANEXOS

Anexo A. Script de código de MATLAB.

Estimación de valores en el transformador de pulsos.

```
1
2 clear;
3 clc;
4
5 Vdrv = 12;
6 incVc1 = 0.1*Vdrv;
7 incVc2 = incVc1;
8 D = 60; %Ciclo de trabajo
9 Dmax = 0.8;
10 Qg = 150e-9; %Total gate charge APT6030BVRG
11 Rgs = 100;
12 Vdc2 = 0.7; %voltaje de caída del diodo
13 K = 0.9987;
14 Lm = 2.6e-3; %inductancia mutua
15 Fdrv = 80e3; %frecuencia de conmutacion
16
17 A = Qg/incVc1;
18 B = ((Vdrv-Vdc2)*D)/(incVc1*Rgs*Fdrv);
19 C = (Vdrv*(D^2-D^3))/(incVc1*4*Lm*Fdrv^2);
20
21 Capc1 = A+B+C;
22
23 A2 = Qg/incVc2;
24 B2 = (Vdrv-Vdc2)*Dmax/(incVc2*Rgs*Fdrv);
25
26 Capc2 = A2+B2;
27
28 ResRc = 2*sqrt(Lm/Capc1);
```



Anexo B. Script de código de MATLAB.

Estimación de valores de la Red Snubber.

```
1
2 f0 ; %frecuencia de resonancia natural
3 f1 ; %frecuencia de resonancia
4 C0 ; %capacitancia paracita
5 C1 ; %capacitor de prueba agregado
6 L ; %INDUCTANCIA PARACITA
7 Csnubber; %capacitor propuesto
8 Rsnubber; %resistencia propuesta
9
10
11
12 % desplazamiento de frecuencia
13 m = f0 - f1;
14
15 %capaciad parasita
16 C0 = C1 / (m^2-1);
17
18 %inductancia parasita
19 L = (m^2-1)/((2*pi * f0)*C1);
20
21 %valor de condensador de la red snubber
22 Csnubber = 3 * C0;
23
24 %valor de la resistencia de la red snubber
25 Rsnubber = sqrt(L / C0);
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

Anexo C. Script de código en Python.

Interfaz de monitoreo.



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```
1 from datetime import datetime
2 import threading
3 import serial
4 import serial.tools.list_ports
5 import matplotlib.pyplot as plt
6 from matplotlib.animation import FuncAnimation
7 from time import time
8 import tkinter as Tk
9 import tkinter as tk
10 from tkinter import ttk
11 from tkinter import *
12 from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
13 from matplotlib.figure import Figure
14 from tkinter import Canvas
15 from PIL import Image, ImageTk
16 import csv
17 import os
18
19 # Variables globales
20 serial_port = None
21 voltages = []
22 currents = []
23 time_values = []
24 duty_cycle_1 = []
25 duty_cycle_2 = []
26 potencia = []
27 start_time = None
28 grosor = 1.7
29
30 def get_available_ports(): #lista de puertos disponibles
31     ports = serial.tools.list_ports.comports()
32     return [port.device for port in ports]
33
34 def start_serial_communication(port):
35
36     global serial_port, start_time
37     try:
38         baud_rate = 115200
39         serial_port = serial.Serial(port, baud_rate, timeout=1)
40         start_time = time()
41         status_label.config(text=f"Conectado a {port}")
42
43         # Iniciar el hilo de lectura de datos
44         thread = threading.Thread(target=read_serial, daemon=
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```

    True)
45     thread.start()
46     except serial.SerialException:
47         status_label.config(text="Error al conectar")
48
49 def read_serial():
50     """Lee datos del puerto serial en un hilo separado."""
51     global serial_port, voltages, currents, time_values,
52         duty_cycle_1, duty_cycle_2, start_time
53     while True:
54         if serial_port and serial_port.in_waiting > 0:
55             try:
56                 raw_data = serial_port.readline().decode('utf-8
57                     ').strip()
58                 if ',' in raw_data:
59                     voltage, current, duty_1, duty_2 = map(
60                         float, raw_data.split(','))
61                     current_time = time() - start_time
62                     voltages.append(voltage)
63                     currents.append(current)
64                     time_values.append(current_time)
65                     duty_cycle_1.append(duty_1)
66                     duty_cycle_2.append(duty_2)
67
68                     potencia.append(voltage * current)
69
70                     if len(time_values) > 1000:
71                         voltages.pop(0)
72                         currents.pop(0)
73                         duty_cycle_1.pop(0)
74                         duty_cycle_2.pop(0)
75                         time_values.pop(0)
76                         potencia.pop(0)
77
78             except ValueError:
79                 pass
80
81 def update_display():
82     if currents: # Solo actualizar si hay datos
83         last_current = currents[-1]
84         current_label.config(text=f"Corriente: {last_current:.3
85             f} A")
86
87     Chopper_control.after(10, update_display)
88
89 def voltage_display():
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```
85     if voltages:
86         last_voltage = voltages[-1]
87         voltage_label.config(text=f"Voltaje: {last_voltage:.1f}
88                                 V")
89
90     Chopper_control.after(10, voltage_display)
91
92 def update_dcycle():
93     if duty_cycle_1:
94         last_dcycle1 = duty_cycle_1[-1]
95         dcycle1_label.config(text=f"PWM1: {last_dcycle1} %")
96
97     Chopper_control.after(10, update_dcycle)
98
99 def update_dcycle2():
100     if duty_cycle_2:
101         last_dcycle2 = duty_cycle_2[-1]
102         dcycle2_label.config(text=f"PWM1: {last_dcycle2} %")
103
104     Chopper_control.after(10, update_dcycle2)
105
106 def update(frame):
107
108     global last_time
109
110     if time_values:
111
112         line1.set_data(time_values, voltages)
113         line2.set_data(time_values, currents)
114         line3.set_data(time_values, duty_cycle_1)
115         line4.set_data(time_values, duty_cycle_2)
116         line5.set_data(time_values, potencia)
117
118         visible_window = 10
119
120     return line1, line2, line3, line4, line5
121
122 def on_port_selected(event):
123
124     selected_port = port_combobox.get()
125     if selected_port:
126         start_serial_communication(selected_port)
127
128 def close_app():
129     Chopper_control.destroy()
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```
129
130 Chopper_control = Tk()
131
132 Chopper_control.geometry("1360x700")
133 Chopper_control.title("Chopper control window")
134 Chopper_control.config(background="#A9A9A9")
135
136 Main_title = Label(Chopper_control,
137                    text="Chopper",
138                    font=('Arial',25,'bold'),
139                    bg="#A9A9A9",
140                    fg='#FFFFFF')
141 Main_title.place(x=10,y=5)
142
143 Main_title = Label(Chopper_control,
144                    text="Control",
145                    font=('Arial',25,'bold'),
146                    bg="#A9A9A9",
147                    fg='#FFFFFF')
148 Main_title.place(x=10,y=45)
149
150 fig, (ax1, ax2) = plt.subplots(2,1,figsize=(6.3,6.5),dpi=100)
151 fig.patch.set_facecolor("#A9A9A9")
152 ax1.set_ylabel("Voltaje (V)")
153 ax2.set_ylabel("Corriente (A)")
154 ax2.set_xlabel("Tiempo(s)")
155 line1, = ax1.plot([], [], 'r-',linewidth=grosor)
156 line2, = ax2.plot([], [], 'b-',linewidth=grosor)
157 ax1.set_ylim(0, 200)
158 ax2.set_ylim(0, 2)
159 ax1.grid(True, which='both', linestyle='--', linewidth=0.7,
160         alpha=0.7)
161 ax2.grid(True, which='both', linestyle='--', linewidth=0.7,
162         alpha=0.7)
163
164 pot2 = Scale(Chopper_control,
165             from_=100,
166             to=0,
167             font=('Arial',10),
168             length=240,
169             width=32,
170             tickinterval=10,
171             troughcolor='#D3D3D3',
172             fg='#FFFFFF',
173             bg='#808080')
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```
172 pot2.place(x=120,y=320)
173
174 pot2.bind("<ButtonRelease-1>", pot_value1)
175
176 C_Sp = Label(Chopper_control,
177             text="Current Set Point",
178             font=('Arial',10,'bold'),
179             fg='#FFFFFF',
180             bg='#808080',
181             relief=RAISED,
182             bd=3,
183             padx=4)
184 C_Sp.place(x=10,y=572)
185
186 Current_Sp = Scale(Chopper_control,
187                   from_=0.0,
188                   to=2.0,
189                   resolution=0.01,
190                   font=('Arial',10),
191                   length=200,
192                   width=32,
193                   troughcolor='#D3D3D3',
194                   fg='#FFFFFF',
195                   bg='#808080',
196                   orient=HORIZONTAL)
197 Current_Sp.place(x=10,y=600)  #(x=10,y=526) 10 x 476
198
199 Current_Sp.bind("<ButtonRelease-1>", pot_value1)
200
201 update_display()
202 voltage_display()
203 update_dcycle()
204 update_dcycle2()
205
206 ani = FuncAnimation(fig, update, interval=10, cache_frame_data=
207                    False)
208 ani2 = FuncAnimation(fig2, update, interval=10,
209                    cache_frame_data=False)
208
209 Chopper_control.mainloop()
```



Anexo D. Script de código en C.

Código de microcontrolador aplicado al convertidor

```
1
2 #include "mcc_generated_files/system/system.h"
3 #include <math.h>
4 #define RMS_SAMPLES 200
5 #define VOL_RMS_SAMPLES 200
6
7 int hit;
8 char buffer[90];
9 int temp[6];
10 int a,b;
11 int DCycle1;
12 int DCycle2;
13 int DCycleG;
14 int M1;
15 int M2;
16 int DC_MAX = 80;
17 int DC_MIN = 60;
18
19 float Ic,cI,Current;
20 float Vc,cV,dV,Voltage;
21 float Sp1;
22 float v_rms,rms,IN_Current,In_Current;
23
24 //Variables PI
25 float Kp = 0.5f;
26 float Ti = 0.01f;
27 float Ts = 0.001f;
28
29 float q0,q1,q2;
30 float u_past = 0.0f;
31 float input_past = 0.0f;
32 float input_past2 = 0.0f;
33
34 void medicion(float voltage,float current,int Sw1,int Sw2);
35 void MEF_Converter(float current_in, float Sp);
36 void PI_init();
37 float PI_Control(float error);
38 float adc_read_rms(adc_channel_t channel);
39 float adc_voltage_rms (adc_channel_t channel2);
40 double sensor1(int a);
41 double sensor2(int b);
42
43 int main(void)
44 {
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```
45 SYSTEM_Initialize();
46 PI_init();
47 // If using interrupts in PIC18 High/Low Priority Mode you
   // need to enable the Global High and Low Interrupts
48 // If using interrupts in PIC Mid-Range Compatibility Mode
   // you need to enable the Global Interrupts
49 // Use the following macros to:
50
51 // Enable the Global Interrupts
52 INTERRUPT_GlobalInterruptEnable();
53
54 // Disable the Global Interrupts
55 //INTERRUPT_GlobalInterruptDisable();
56
57 PIR4bits.PWM1IF = 1;
58 while(1)
59 {
60     if(hit==1){
61         PIR4bits.PWM1IF = 0;
62         recepcionpwm(buffer,temp);
63         hit=0;
64     }
65         rms = adc_read_rms(ADC_CHANNEL_ANA6);
66         v_rms = adc_voltage_rms(ADC_CHANNEL_ANBO);
67         Csp = temp[2];
68         Current = sensor1(rms);
69         Sp1 = Csp/1000.0;
70         MEF_Converter(Current,Sp1);
71
72         __delay_ms(5);
73         PWM1CONbits.LD = 1;
74         PWM1_16BIT_SetSlice1Output1DutyCycleRegister(
           DCycle1);
75         PWM1_16BIT_SetSlice1Output2DutyCycleRegister(
           DCycle2);
76         __delay_ms(5);
77         PWM1CONbits.LD = 0;
78
79         Voltage = sensor2(v_rms);
80
81         M1 = DCycle1;
82         M2 = 100-DCycle2;
83
84         medicion(Voltage,Current,M1,M2);
85     }
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```
86 }
87
88 float adc_read_rms(adc_channel_t channel) {
89     uint32_t sum_squares = 0;
90
91     for (uint8_t i = 0; i < RMS_SAMPLES; i++) {
92         uint16_t sample = ADC_ChannelSelectAndConvert(channel);
93         sum_squares += (uint32_t)sample * sample;
94         __delay_us(5);
95     }
96
97     float mean_square = (float)sum_squares / RMS_SAMPLES;
98     float rms_value = sqrtf(mean_square);
99
100    return rms_value;
101 }
102 float adc_voltage_rms (adc_channel_t channel2){
103     uint32_t sum_raiz = 0;
104
105     for (uint8_t u = 0; u < VOL_RMS_SAMPLES; u++) {
106         uint16_t muestra = ADC_ChannelSelectAndConvert(channel2
107             );
108         sum_raiz += (uint32_t)muestra * muestra;
109         __delay_us(5);
110     }
111
112     float prom_raiz = (float)sum_raiz / VOL_RMS_SAMPLES;
113     float rms_vol_value = sqrtf(prom_raiz);
114
115     return rms_vol_value;
116 }
117
118 void medicion(float voltage,float current,int Sw1, int Sw2){
119     printf("%.2f,%.2f,%d,%d\n\r",voltage,current,Sw1,Sw2);
120     __delay_ms(10);
121 }
122
123 double sensor1(a){
124
125     Vc = (a*5.0);
126     cV = Vc / 4096;
127
128     return cV;
129 }
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```
130 double sensor2(b){
131     Ic = (b*5.0);
132     cI = Ic / 4096;
133
134     return cI;
135 }
136
137
138 void MEF_Converter(float current_in, float Sp){
139     float E = current_in - Sp;
140     float upL      = 1.05 * Sp;
141     float lowL     = 0.95 * Sp;
142
143     if (E > error_alto){
144         DCycle1 = 68;
145         DCycle2 = 32;
146         lastState = HIGH1;
147     }
148     else if (E < error_bajo){
149         DCycle1 = 70;
150         DCycle2 = 30;
151         lastState = LOW1;
152     }
153     else if (E > lowL && E < upL){
154         DCycle1 = PI_Control(current_in);
155         DCycle2 = 100 - PI_Control(current_in);
156     }
157 }
158
159 void PI_init(){
160
161     q0 = Kp+(Kp/Ti)+(Kp/Ts);
162     q1 = -Kp*(1+2*Ts);
163     q2 = 0;
164     u_past = 0;
165     input_past = 0;
166     input_past2 = 0;
167 }
168
169 float PI_Control(float current_in){
170     float u;
171     float input = current_in;
172
173     u = u_past + q0 * input + q1 * input_past + q2 *
        input_past2;
```



“Control Inteligente para una fuente pulsada en modo corriente basado en un sistema embebido”

```
174
175     if (u > 80){
176         u == DC_MAX;
177     }
178     if (u < 60){
179         u == DC_MIN;
180     }
181
182     input_past2 = input_past;
183     input_past = input;
184
185     return u;
186
187     __delay_ms(10);
188 }
```