

# Proyecto Final

Estrategias de Control para Sistemas Híbridos de Almacenamiento de Energía orientados a aplicaciones móviles



**Graselli Valentín Mateo**

**Torres Alberto José Francisco**

Instituto de Investigaciones en Electrónica, Control y Procesamiento de  
Señales (LEICI)  
Universidad de La Plata

*Director: Moré Jerónimo José*

*Co-Director: Puleston Paul*

Marzo 2024

## **Agradecimientos**

En primer lugar a nuestras familias por su constante apoyo, esfuerzo y sacrificio durante todo este proceso. Su amor incondicional y aliento han sido fundamental para mantenernos motivados y enfocados en alcanzar nuestras metas.

También queremos extender nuestra gratitud a nuestros directores, *Paul Puleston*, *Jerónimo Moré* y *Claus Nahuel Mancini*, por su experta experiencia, orientación, dedicación e invaluable consejos.

Además, agradecer a nuestros compañeros de la facultad por su apoyo incondicional a lo largo de estos años que han hecho de este proceso una experiencia memorable y enriquecedora.

Queremos expresar nuestro profundo agradecimiento a la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC) por las becas otorgadas, las cuales fueron fundamentales para llevar a cabo este proyecto. Su apoyo financiero y académico ha sido invaluable para el desarrollo de nuestra investigación y formación profesional.

A todos ellos, nuestro sincero agradecimiento por ser parte fundamental de este camino y por su contribución a este proyecto.

# Abstract

This project delves into the current global energy landscape, marked by the imperative to reduce greenhouse gas emissions and combat climate change amidst projections of increased energy demand. In Argentina, where a significant portion of energy comes from fossil fuels, specially in the transport industry, the need for sustainable solutions is urgent. For instance, electric vehicles have emerged as a promising alternative, featuring high energy efficiency compared to traditional combustion vehicles. However, energy requirements from these vehicles raise a challenge from the differing demands of sustained power consumption and sudden, short-term spikes. This performance requirements poses a significant obstacle in traditional energy storage systems, which often excel in addressing one type of demand while falling short in meeting the other. To tackle this issue effectively, Hybrid Energy Storage Systems (HESS) offer a promising approach, combining the strengths of different storage technologies to meet diverse energy demands efficiently. This project focuses on a hybrid system incorporating Lithium-Ion Batteries (BL) and supercapacitors (SC). By leveraging the high power density of SCs and the high energy density of BLs, the system aims to address both abrupt power requirements and sustained energy delivery, contributing to enhanced efficiency and performance in various applications, particularly in the transportation sector. This system is engineered to facilitate power exchange not only between the supercapacitor banks and lithium-ion batteries but also with the load, such as a DC motor, achieving high efficiency scenarios like, for example, regenerative braking. This is accomplished through bidirectional power electronic converters.

The objective is to devise an alternative control strategy and compare it with the conventional linear Proportional-Integral (PI) control technique in terms of performance, stability, dynamic response and robustness to variations in the system and operating conditions. This control strategy, incorporating gain scheduling, considers the inherent non-linearities of the converters. The system is digitally controlled by a micro-computing system of Field Programmable Gate Array (FPGA) type, enabling the implementation of the control systems and real-time data acquisition on a computer. Furthermore, a user interface is developed in Python that allows interacting with the FPGA and monitoring the system's variables of interest in real-time, as well as storing

data in CSV files. This interface permits the visualization, analysis and evaluation of the implemented control strategies both online and offline. Finally, the system is experimentally validated with a programmable electronic load, that simulates the operating conditions of a strong variability in power requirements like with an electric vehicle.

# Resumen

Este proyecto se encuentra dentro del panorama energético global actual, marcado por la necesidad de reducir las emisiones de gases de efecto invernadero y combatir el cambio climático considerando las proyecciones de aumento de la demanda energética. En Argentina, donde una parte importante de la energía proviene de combustibles fósiles, especialmente en la industria del transporte, es urgente la necesidad de soluciones sustentables. Por ejemplo, los vehículos eléctricos se han convertido en una alternativa prometedora, que presenta una alta eficiencia energética en comparación con los vehículos de combustión tradicionales. Sin embargo, los requisitos de energía de estos vehículos plantean un desafío debido a las diferentes demandas del consumo de energía sostenido y los picos repentinos. Estos requisitos de rendimiento plantean un obstáculo importante en los sistemas tradicionales de almacenamiento de energía, que a menudo se destacan en suplir un tipo de demanda pero no logran satisfacer el otro. Para abordar este problema de manera efectiva, los sistemas híbridos de almacenamiento de energía (HESS) ofrecen un enfoque prometedor, que combina las fortalezas de diferentes tecnologías de almacenamiento para satisfacer diversas demandas de energía de manera eficiente. Este proyecto se centra en un sistema híbrido que incorpora baterías de iones de litio (BL) y supercapacitores (SC). Al aprovechar la alta densidad de potencia de los SC y la alta densidad de energía de los BL, el sistema tiene como objetivo abordar tanto los requisitos de energía abruptos como la entrega sostenida de energía, contribuyendo a una mayor eficiencia y rendimiento en diversas aplicaciones, particularmente en el sector del transporte. Este sistema está diseñado para facilitar el intercambio de energía no sólo entre los bancos de SC y las baterías de iones de litio sino también con la carga, como un motor DC, logrando escenarios de alta eficiencia como, por ejemplo, el frenado regenerativo. Esto se logra mediante convertidores electrónicos de potencia bidireccionales.

El objetivo es proponer una estrategia de control alternativa y compararla con la técnica de control lineal Proporcional-Integral (PI) convencional en términos de rendimiento, estabilidad, respuesta dinámica y robustez ante variaciones en el sistema y condiciones de operación. Esta estrategia de control, que incorpora programación de ganancia, considera las no linealidades inherentes de los convertidores. El sistema es

controlado digitalmente por un sistema de micro-computo del tipo Field Programmable Gate Array (FPGA), que permite la implementación de los sistemas de control y adquisición de datos en tiempo real en una computadora. Además, se desarrolla una interfaz de usuario en Python que permite interactuar con la FPGA y monitorear las variables de interés del sistema en tiempo real, así como almacenar datos en archivos CSV. Esta interfaz permite la visualización, análisis y evaluación de las estrategias de control implementadas tanto online como offline. Finalmente, el sistema se valida experimentalmente con una carga electrónica programable, que simula las condiciones de funcionamiento de una fuerte variabilidad en los requisitos de energía como ocurre con un vehículo eléctrico.

# Índice general

<b>Índice de figuras</b>	<b>x</b>
<b>Nomenclatura</b>	<b>XIV</b>
<b>1. Introducción</b>	<b>1</b>
<b>2. Sistemas de Almacenamiento</b>	<b>4</b>
2.1. Introducción . . . . .	4
2.2. Sistemas de almacenamiento . . . . .	4
2.3. Supercapacitores . . . . .	6
2.3.1. Clasificación y principio básico de funcionamiento . . . . .	6
2.3.2. Aplicaciones . . . . .	7
2.3.3. Banco de Supercapacitores . . . . .	8
2.4. Baterías Ion-Litio . . . . .	9
2.4.1. Principio básico de funcionamiento . . . . .	9
2.4.2. Baterías de Litio-Ferrofosfato . . . . .	10
2.4.3. Proceso de Carga y Descarga . . . . .	11
2.4.4. Aplicaciones . . . . .	12
2.5. Sistema híbrido de almacenamiento de energía . . . . .	13
2.6. Resumen . . . . .	14
<b>3. Convertidores CC-CC</b>	<b>15</b>
3.1. Introducción . . . . .	15
3.2. Convertidor Elevador . . . . .	17
3.2.1. Modo conducción continua . . . . .	18
3.2.2. Convertidor elevador bidireccional en corriente . . . . .	20
3.3. Resumen . . . . .	23
<b>4. Field-Programmable Gate Arrays</b>	<b>24</b>
4.1. Introducción . . . . .	24
4.2. Arquitectura . . . . .	24

4.2.1. Bloques Lógicos configurable . . . . .	25
4.3. Lenguaje de descripción de hardware . . . . .	26
4.3.1. Verilog . . . . .	27
4.3.2. Maquina de Mealy . . . . .	27
4.4. Intel Quartus Prime . . . . .	28
4.4.1. IP cores . . . . .	29
4.5. FPGA DE10-Lite . . . . .	29
4.6. Aplicaciones . . . . .	31
4.7. Flujo de diseño . . . . .	31
4.8. Resumen . . . . .	33
<b>5. Diseño de la etapa de control</b> . . . . .	<b>34</b>
5.1. Introducción . . . . .	34
5.2. Modelo del convertidor . . . . .	34
5.2.1. Dinámica del convertidor . . . . .	35
5.2.2. Parametrizado mediante ciclo de trabajo . . . . .	36
5.2.3. Definición de la acción de control . . . . .	36
5.2.4. PWM . . . . .	36
5.2.5. Promediado del sistema de ecuaciones . . . . .	37
5.2.6. Linealización del sistema de ecuaciones . . . . .	38
5.3. Diseño del sistema de control . . . . .	40
5.3.1. Diseño del lazo interno de corriente . . . . .	41
5.3.2. Diseño del lazo de tensión externo . . . . .	53
5.3.3. Planificación de ganancias ( <i>Gain Schedule</i> ) . . . . .	54
5.3.4. Análisis de sensibilidad de los polos respecto $i_L$ . . . . .	55
5.3.5. División de regiones para las corrientes de inductor . . . . .	57
5.3.6. Sintonización de PI en las regiones definidas . . . . .	57
5.3.7. Mejoras de la planificación de ganancias en comparación al PI . . . . .	66
5.4. Modulo de almacenamiento híbrido: SC y BL . . . . .	68
5.4.1. Etapa de control del sistema completo . . . . .	70
5.5. Resumen . . . . .	74
<b>6. Diseño digital</b> . . . . .	<b>75</b>
6.1. Introducción . . . . .	75
6.2. Diseño de módulos de control . . . . .	76
6.2.1. Conversor Analógico-Digital . . . . .	76
6.2.2. Filtro digital . . . . .	80
6.2.3. Controlador PWM . . . . .	81
6.2.4. Controlador Proporcional-Integral . . . . .	82

---

6.2.5. Controlador Proporcional-Integral de tensión GS . . . . .	84
6.2.6. Feedforward . . . . .	86
6.2.7. Construcción del ciclo de trabajo . . . . .	87
6.3. Diseño de sistema de comunicación . . . . .	87
6.3.1. Codificación KLV . . . . .	88
6.3.2. Comunicación UART . . . . .	89
6.3.3. Decodificador . . . . .	92
6.3.4. Armado de paquetes KLV . . . . .	94
6.3.5. Envío de paquetes KLV . . . . .	96
6.4. Módulos auxiliares . . . . .	97
6.5. Análisis de Resultados de Síntesis y Timing en Quartus Prime . . . . .	99
6.6. Resumen . . . . .	100
<b>7. Interfaz de usuario</b>	<b>101</b>
7.1. Introducción . . . . .	101
7.2. PyQt6 . . . . .	102
7.2.1. Decodificación KLV . . . . .	104
7.3. Almacenamiento de referencias por interfaz . . . . .	106
7.4. PlotJuggler . . . . .	106
7.4.1. Lectura en tiempo real . . . . .	107
7.4.2. Lectura CSV . . . . .	108
7.4.3. Optimización en envío de datos . . . . .	109
7.5. Resumen . . . . .	109
<b>8. Ensayos y resultados experimentales</b>	<b>111</b>
8.1. Introducción . . . . .	111
8.2. Rack de pruebas . . . . .	112
8.3. Ensayos . . . . .	113
8.3.1. Fase 1: Ensayo de lazo abierto . . . . .	113
8.3.2. Fase 2: Lazo de control de corriente . . . . .	114
8.3.3. Fase 3: Lazo de control de tensión . . . . .	115
8.4. Resumen . . . . .	119
<b>9. Conclusiones</b>	<b>120</b>
<b>Bibliografía</b>	<b>122</b>

# Índice de figuras

2.1. Clasificación de los sistemas de almacenamiento [1]. . . . .	5
2.2. Tiempos de descarga en función de potencia. . . . .	5
2.3. Clasificación supercapacitores. . . . .	6
2.4. Arreglo de SC implementado. . . . .	8
2.5. Banco de SC utilizado. . . . .	9
2.6. Esquema batería Ion-Litio. . . . .	10
2.7. Batería Ion-Litio disponible en el LEICI. . . . .	11
2.8. Densidades de energía en tipos de baterías: gravimétrica y volumétrica. . . . .	12
2.9. Diagrama de Ragone. . . . .	13
3.1. Convertidor conmutado CC-CC básico y su equivalente de conmutación. . . . .	16
3.2. Tensión de salida del circuito de la figura 3.1. . . . .	16
3.3. Convertidor elevador. . . . .	17
3.4. Formas de onda del convertidor elevador. . . . .	18
3.5. Convertidor elevador bidireccional en corriente. . . . .	20
3.6. Sistema eléctrico para un VE basado en baterías. . . . .	21
3.7. Convertidores utilizados en el proyecto. . . . .	22
4.1. Arquitectura básica de un FPGA. . . . .	25
4.2. Bloque lógico de un FPGA. . . . .	26
4.3. Máquina de Mealy . . . . .	28
4.4. FPGA MAXDE10-Lite. . . . .	30
4.5. Relación entre impacto del diseño y complejidad. . . . .	31
5.1. Convertidor elevador bidireccional en corriente. . . . .	35
5.2. Modulación por ancho de pulso (PWM). . . . .	37
5.3. Ilustración del lazo de tensión propuesto. . . . .	40
5.4. Ilustración del lazo de control propuesto. . . . .	41
5.5. Ilustración del lazo de corriente propuesto. . . . .	42
5.6. Lugar de raíces de la transferencia encontrada. . . . .	43

5.7. DEP de la señal filtrada. . . . .	45
5.8. Comparación ambos filtrados. . . . .	46
5.9. Lazo de control de corriente con filtro pasa bajos. . . . .	46
5.10 Sistema de lazo cerrado con control PID. . . . .	47
5.11 Lazo de control de corriente con filtro pasa bajos. . . . .	50
5.12 Sintonización mediante <i>PID tuner</i> del lazo de corriente interno. . . . .	50
5.13 Simulación real del lazo de corriente . . . . .	51
5.14 Formas de onda obtenidas en la simulación del lazo de control de corriente. . . . .	52
5.15 Corriente del inductor en el entorno del escalón unitario de corriente. . . . .	52
5.16 Lazo de control de tensión . . . . .	53
5.17 Sensibilidad de polos respecto a la corriente del inductor . . . . .	56
5.18 Modelo linealizado para lazo externo de tensión. . . . .	57
5.19 Tensión en la salida para todas las regiones . . . . .	58
5.20 Feedforward implementado en la simulación . . . . .	60
5.21 Bloque utilizado para simular variación en la carga . . . . .	60
5.22 Simulación real del lazo de control de tensión. . . . .	61
5.23 Formas de onda obtenidas ante una variación en la resistencia de carga. . . . .	62
5.24 Formas de onda obtenidas ante una variación en la resistencia de carga. . . . .	63
5.25 Comparación de interpolación vs puntos obtenidos . . . . .	64
5.26 Implementación de la técnica GS . . . . .	65
5.27 Estabilidad de los polos a lazo cerrado. . . . .	65
5.28 Comparación entre las formas de onda utilizando la técnica GS y un PI. . . . .	66
5.29 Comparación entre las formas de onda utilizando la técnica GS y un PI. . . . .	67
5.30 Esquema del sistema completo. . . . .	68
5.31 Diseño de la etapa de control para la batería. . . . .	71
5.32 Diseño de la etapa de control para el banco de SC. . . . .	71
5.33 Planta real del sistema completo . . . . .	72
5.34 Formas de onda obtenidas mediante simulación del sistema completo. . . . .	73
6.1. Sistema de módulos de control completo . . . . .	76
6.2. Interpolación de tensión de bus . . . . .	78
6.3. Interpolación de tensión de SC y BL . . . . .	78
6.4. Interpolación de corriente de inductor de SC y BL . . . . .	79
6.5. Diagrama de puertos de módulo de conversión de tensión de bus . . . . .	79
6.6. RTL de filtro IIR en forma directa I . . . . .	80
6.7. Diagrama de puertos de filtro . . . . .	80
6.8. Testbench de filtro . . . . .	81
6.9. Diagrama de puertos de módulo PWM . . . . .	81
6.10. Testbench de módulo PWM . . . . .	82

6.11	Zoom flancos de PWM . . . . .	82
6.12	Diagrama de puertos de módulo PI de corriente. . . . .	82
6.13	Diagrama de puertos de módulo PI de tensión. . . . .	83
6.14	RTL de PI de corriente . . . . .	83
6.15	Testbench PI de corriente . . . . .	84
6.16	Diagrama de puertos de módulo GS . . . . .	84
6.17	RTL de PI de tensión con GS . . . . .	85
6.18	Testbench PI de tensión con GS . . . . .	85
6.19	Testbench módulos Kp y Ki . . . . .	86
6.20	Diagrama de puertos de módulo feedforward . . . . .	86
6.21	Testbench módulo de feedforward . . . . .	87
6.22	Diagrama de puertos de módulo constructor del ciclo de trabajo . . . . .	87
6.23	Diagrama de sistema de comunicación . . . . .	88
6.24	Diagrama de puertos de módulo receptor UART . . . . .	90
6.25	Maquina de estado de UART Rx . . . . .	91
6.26	Diagrama de puertos de módulo transmisor UART . . . . .	91
6.27	Maquina de estado de UART Tx . . . . .	92
6.28	Testbench de módulos de transmisión y recepción de UART . . . . .	92
6.29	Diagrama de puertos de módulo decodificador . . . . .	93
6.30	Maquina de estados de módulo decodificador . . . . .	93
6.31	Testbench de modulo decodificador . . . . .	94
6.32	Testbench carga de referencia de tensión de bus . . . . .	94
6.33	Diagrama de puertos de módulo de armado de paquetes . . . . .	95
6.34	Maquina de estados de armado de paquetes . . . . .	95
6.35	Testbench de modulo de armado de paquetes KLV . . . . .	96
6.36	Diagrama de puertos del módulo de envío de paquetes. . . . .	96
6.37	Maquina de estados de módulo de envío de paquetes . . . . .	97
6.38	Testbench de modulo de envío de paquetes . . . . .	97
6.39	Módulos auxiliares . . . . .	98
6.40	Reporte de timing . . . . .	99
6.41	Reporte de recursos utilizados . . . . .	100
7.1.	Interfaz diseñada para este proyecto. . . . .	103
7.2.	Decodificación de datos KLV. . . . .	105
7.3.	Análisis decodificación de datos KLV . . . . .	105
7.4.	Comparación formato <i>JSON</i> con <i>MessagePack</i> . . . . .	107
7.5.	Ejemplo del Trigger implementado para un flanco de subida. . . . .	109
8.1.	Esquemático de conexiones de rack de prueba. . . . .	112

---

8.2. Rack de ensayos. . . . .	113
8.3. Ensayos del lazo de corriente. . . . .	114
8.4. Tensión en la carga y corriente por el inductor. . . . .	115
8.5. Comparación de las formas de onda de tensión en la carga y corriente por el inductor frente a una variación en la carga de $14,54\Omega$ a $74\Omega$ . . . . .	116
8.6. Comparación de las formas de onda de tensión en la carga y corriente por el inductor frente a una variación en la carga de $74\Omega$ a $14,54\Omega$ . . . . .	116
8.7. Comparación de las formas de onda de tensión en la carga y corriente por el inductor frente a diferentes variaciones en la carga. . . . .	117
8.8. Variación en la carga de $14,54\Omega$ a $10,8\Omega$ . . . . .	118

# Nomenclatura

## **Acrónimos / Abreviaturas**

ADC Analog Digital Converter

ALU Unidad Aritmético Lógica

BL Baterías de Ión-Litio

BMS Battery Management System

CLB Bloque Lógico Configurable

DEP Densidad Espectral de Potencia

DUT Design Under Test

EV Electric Vehicle

FPGA Field-Programmable Gate Arrays

GS Gain Schedule

HDL Hardware Description Lenguaje

HESS Hybrid Energy Storage System

IIR Infite Impulse Response

KLV Key Length Value

LUT LookUp Table

MANC Módulos de Almacenamiento de energía No Convencionales

MCC Modo de Conducción Continua

PFM Pulse Frequency Modulation

PID controlador Proporcional-Integral-Derivativo

PI	controlador Proporcional-Integral
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
RTL	Register-Transfer Level
SC	SuperCapacitores
TLV	Type Length Value
UART	Universal Asynchronous Receiver-Transmitter
VE	Vehículos Eléctricos

# Capítulo 1

## Introducción

El contexto energético global enfrenta una transformación crítica, impulsada por la creciente conciencia de reducir las emisiones de gases de efecto invernadero y abordar el cambio climático. Este desafío se ve agravado dado que se estima un aumento significativo en la población mundial para el 2050, lo que incrementará aún más la demanda de energía. En Argentina, aproximadamente el 65 % de la matriz energética proviene de combustibles fósiles, siendo el sector del transporte el principal consumidor, aportando el 99,67 % de la energía necesaria.

El transporte, en particular, es responsable de casi el 30 % de las emisiones globales de CO<sub>2</sub>, lo que destaca la urgente necesidad de transitar hacia soluciones más sostenibles y eficientes [2] [3]. Los Vehículos Eléctricos (VE) han surgido como una alternativa prometedora, no solo por su contribución a la reducción de emisiones sino también por su alto rendimiento energético. Los VE han demostrado un rendimiento superior al 75 %, en comparación con el bajo 15 % de los vehículos de combustión tradicionales, marcando un hito en la eficiencia energética [4].

Sin embargo no existe una única solución ideal para el almacenamiento de energía en todas las aplicaciones, ya que los diseños disponibles en el mercado suelen dividirse en dos categorías: aquellos diseñados para aplicaciones de potencia, formados para entregar grandes cantidades de energía en cortos períodos de tiempo, y aquellos destinados a aplicaciones con alta densidad energética, que proporcionan una entrega constante de energía durante extensos periodos de tiempo.

Los Sistemas de Almacenamiento de Energía Híbridos (*HESS*, por sus siglas en inglés) combinan estas soluciones para satisfacer aplicaciones que requieren tanto respuestas rápidas de energía como así también entregas constantes. Si bien existen diversas topologías, las mas aceptadas son aquellas que emplean baterías de litio, solas o en combinación con algún otro módulo de almacenamiento [4] [5]. Cabe destacar que, estos sistemas híbridos pueden compartir la misma electrónica de potencia y hardware de conexión a la carga, reduciendo así los costos iniciales y de mantenimiento.

---

En este contexto, el trabajo se centra en un sistema híbrido, compuesto por baterías de Ión-Litio (BL) y supercapacitores (SC). Esta combinación estratégica busca aprovechar las fortalezas de ambas tecnologías, superando las limitaciones individuales. Los SC, con su alta densidad de potencia, y las BL, con su alta densidad de energía, se complementan para proporcionar una solución integral que puede abordar tanto las demandas de potencia producto de una aceleración/desaceleración del vehículo, como así también contribuir al aumento de la eficiencia global del sistema mediante la implementación de estrategias de frenado regenerativo u otras acciones [6] [7].

La flexibilidad de este sistema híbrido permite su uso en sistemas de pequeña a gran escala, y poseen un excelente performance en términos de eficiencia, vida útil, y densidad de potencia. Debido a esto, se convirtieron en elementos de sumo interés para la comunidad científico-tecnológica en el desarrollo de vehículos eléctricos y otros sistemas híbridos de energía renovable. Al utilizar estos arreglos, se implementan sistemas de balanceo, regulación, medición y protección que se encarguen de mantener las tensiones, corrientes y temperaturas de los módulos dentro de un rango seguro de operación, con el objetivo de conservar su vida útil y asegurar su funcionamiento dentro de los parámetros nominales.

En el marco de este proyecto, se utilizó un banco de SC íntegramente diseñado en el instituto LEICI de la FI, UNLP-CONICET. Estos SC son de doble capa electrostática, con capacidad nominal de  $80F$  y una tensión nominal de funcionamiento de  $27V$ . Asimismo, se consideró el uso de dos BL de  $24V$  disponibles en el Instituto. Cada una de ellas construidas con ocho celdas de batería de tipo  $LiFePO_4$ , en serie, de  $50Ah$ . Estas celdas se encuentran reguladas por sus respectivos BMS (*Battery Management System*), en este caso industriales *JBD 8S* con protección por sobrecarga. Debido que la tensión suministrada por estas fuentes es menor que la requerida por la carga, resulta la necesidad de emplear convertidores elevadores de CC-CC, los cuales se encargan de ajustar la tensión del sistema con la máxima eficiencia posible.

Por otra parte, el sistema de control será el responsable de comandar las variables eléctricas del sistema a los efectos de garantizar el desempeño eficiente y seguro del sistema híbrido. Es por ello que, a lo largo de este informe se desarrollaran distintos diseños del sistema de control, comenzando con la linealización de la planta en un punto de trabajo específico, sobre la cuál, se aplicarán técnicas tanto convencionales como avanzadas teniendo en cuenta las no linealidades de la planta.

Estas estrategias de control serán luego implementadas en una *FPGA* realizando cada bloque por separado y verificando su funcionamiento a través de herramientas de simulación digitales. Para así finalmente, realizar el desarrollo de una interfaz para el monitoreo de las variables relevantes del sistema en tiempo real desde la *PC*. Involucrando el desarrollo de un protocolo de comunicación con la *FPGA* acorde a los requerimientos de muestreo y velocidad de comunicación.

Particularmente en este informe, desde Capítulo 2 hasta el Capítulo 4, se analizan con detalle los sistemas de almacenamiento más convenientes para el proyecto, así como el funcionamiento básico de los convertidores encargados de adaptar los niveles de cada módulo al bus de continua, y los entornos de desarrollo digitales necesarios para la implementación. A partir del Capítulo 5, se obtiene el modelo linealizado de la planta, seguido de las estrategias de control utilizadas y sus respectivas simulaciones, análisis de estabilidad y performance para comparar los resultados obtenidos con cada estrategia. Posteriormente, en el Capítulo 6, se implementarán estas técnicas de control en una *FPGA* asegurando su correcto funcionamiento por medio de entornos de desarrollo digitales. Finalmente, en el Capítulo 7, se aborda el diseño de una interfaz gráfica capaz de monitorear las variables de mayor interés del sistema en tiempo real desde la *PC* y permitir la interacción de la misma con la *FPGA*.

## Capítulo 2

# Sistemas de Almacenamiento

### 2.1. Introducción

En el presente capítulo se detalla una visión general de los sistemas de almacenamiento con un enfoque específico en aquellos utilizados para el proyecto: las baterías de Ión-Litio (BL) y los bancos de supercapacitores (SC). Se abordarán aspectos fundamentales como su clasificación, sus principios básicos de funcionamiento, características constructivas y aplicaciones. Además se analizarán ambas tecnologías en base al almacenamiento de energía y la rapidez con que cada una la puede entregar.

Las BL son dispositivos empleados para el almacenamiento de energía eléctrica, en el cual los iones de litio se desplazan del ánodo de la batería hacia el cátodo durante el proceso de descarga, a través de un electrolito que posee, de igual manera, iones de litio en su composición permitiendo el flujo de corriente. Clasificadas como baterías secundarias, estas son recargables y se caracterizan por su estabilidad debido a su resistencia a la oxidación espontánea. Este tipo de baterías son aplicadas en diversas áreas, desde comunicaciones inalámbricas hasta dispositivos portátiles y vehículos eléctricos.

Por otra parte, los bancos de SC permiten almacenar energía con un elevado rendimiento a comparación de las baterías y son capaces de suministrar elevados picos de potencia, lo que hace factible su implementación en sistemas híbridos de generación de energía, incrementando la versatilidad y eficiencia de todo el conjunto. Suelen agruparse en serie para elevar la tensión de trabajo de todo el sistema para aplicaciones de potencia y en paralelo para incrementar la energía almacenable en el banco.

### 2.2. Sistemas de almacenamiento

Se define como sistema de almacenamiento de energía a cualquier dispositivo tecnológico capaz de gestionar el traslado temporal de energía para suministrarla según

las necesidades. Estos sistemas pueden clasificarse de diferentes maneras, una de ellas es según su tecnología, como se ilustra en la siguiente figura:

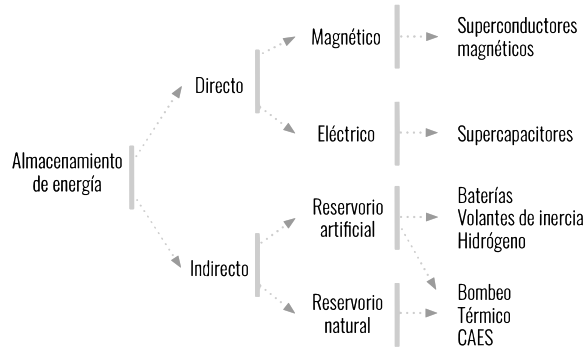


Figura 2.1 Clasificación de los sistemas de almacenamiento [1].

Otra forma de clasificarlos es según el tiempo de descarga que estos presentan, lo que diferencia la respuesta que proporcionan en función de las necesidades del sistema. A continuación se presenta las variaciones de los tiempos de descarga en función de la potencia que pueden almacenar los diversos tipos de sistemas:

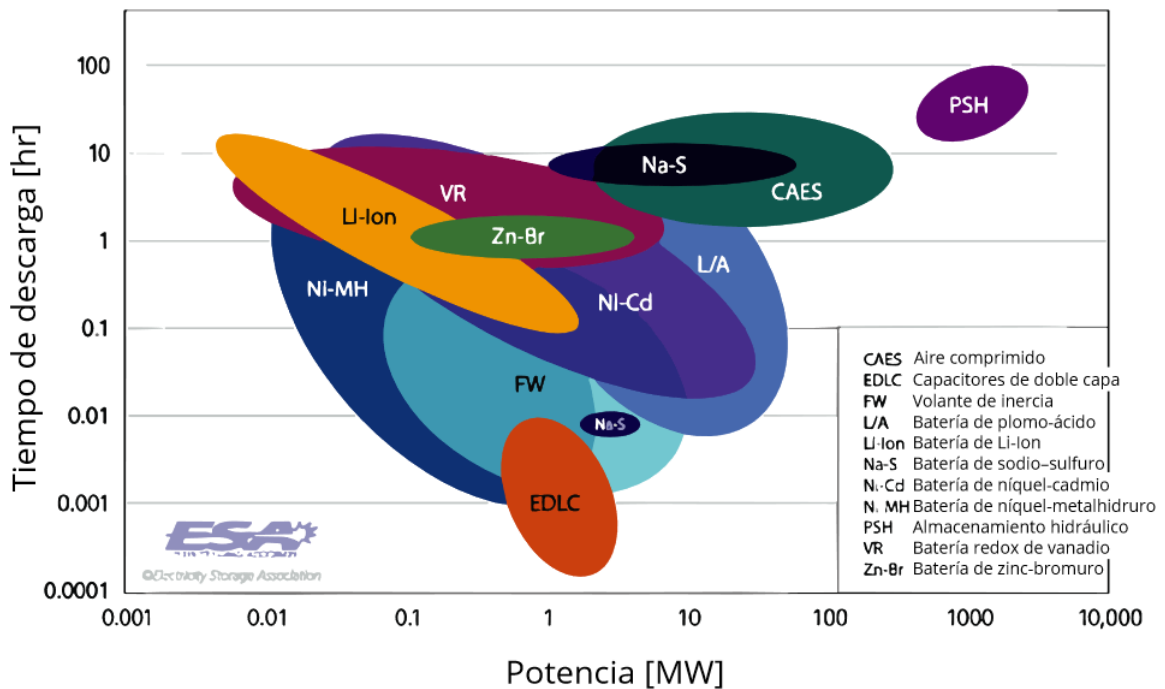


Figura 2.2 Tiempos de descarga en función de potencia.

En este proyecto se optó por la combinación de supercapacitores y baterías de litio para conformar el sistema de almacenamiento de energía. Esta decisión se fundamenta en la capacidad de los SC para proporcionar una gran cantidad de energía en intervalos cortos de tiempo, mientras que las BL son ideales para entregar potencia de manera constante durante periodos más extensos (ver figura 2.2). Es por ello que, a continuación, se llevará a cabo un análisis más detallado de ambos sistemas.

## 2.3. Supercapacitores

En los últimos años, los SC han surgido como una alternativa o complemento importante para otros dispositivos de producción o almacenamiento de energía eléctrica, como las pilas de combustible o las baterías. Sin embargo, a diferencia de estas últimas, los SC son una fuente limpia de almacenamiento, y pueden entregar/almacenar energía de forma prácticamente instantánea, con una vida útil mucho mayor que la de las baterías y celdas de combustible. Esto último los convierte en elementos que permitirían elevar no sólo la eficiencia del conjunto, sino también la vida útil y la versatilidad del mismo [8].

### 2.3.1. Clasificación y principio básico de funcionamiento

Existen diferentes tipos de supercapacitores, generalmente se suelen dividir según la composición del material dieléctrico o del conductor usado para su constitución, entre los principales tipos existen:

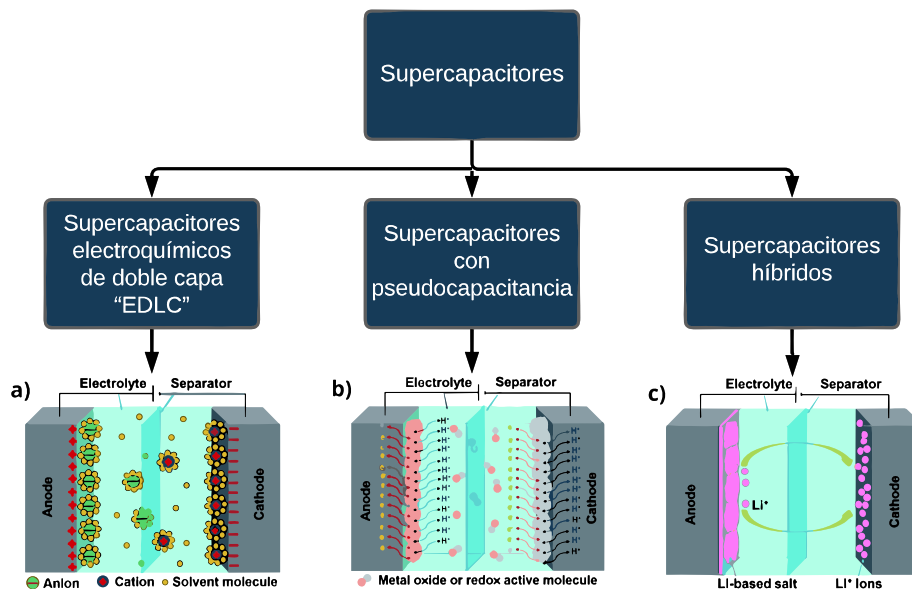


Figura 2.3 Clasificación supercapacitores.

- **a) Supercapacitores electroquímicos de doble capa “EDLC”:** El supercapacitor, también conocido como capacitor electroquímico de doble capa o ultracapacitor, emplea dos capas de carga cuando se aplica un voltaje a un electrodo sumergido en una sustancia electrolítica. Sus electrodos, construidos en carbono altamente poroso, proporcionan una superficie con una amplia relación de volumen. A diferencia de un capacitor electrolítico, que utiliza láminas metálicas y óxido como dieléctrico, el EDLC presenta una capacitancia volumétrica de  $40 - 60 F/cm^3$  debido a la pequeña separación entre capas electrolíticas y una gran área de superficie.
- **b) Supercapacitores con pseudocapacitancia:** A diferencia del EDLC, que almacenan cargas electrostáticamente, los pseudocapacitores o supercapacitores con pseudocapacitancia almacenan cargas de manera faradaica mediante la transferencia de carga entre un electrodo y un electrolito. Los pseudocapacitores se subclasifican en dos grupos según los materiales de los electrodos: aquellos compuestos por materiales polímeros conductores y aquellos compuestos por metal-óxidos.
- **c) Supercapacitores híbridos:** Estos sistemas combinan procesos no farádicos y farádicos para almacenar carga, un EDLC híbrido puede lograr capacidades de almacenamiento de energía significativamente superiores a las de un EDLC convencional. Esto se logra sin comprometer la estabilidad y el ciclo de accesibilidad, que son limitaciones clave de los pseudocapacitores.

### 2.3.2. Aplicaciones

Los SC ofrecen diversas ventajas, como su capacidad para manejar altas corrientes y la posibilidad de cargarse y descargarse en ciclos tanto cortos como relativamente largos. Aunque individualmente pueden tener limitaciones en cuanto a la tensión y la temperatura, su integración en sistemas de almacenamiento de energía presenta un amplio abanico de aplicaciones, que incluyen:

- Desarrollo de convertidores integrando supercapacitores aplicando los principios de la electrónica de potencia.
- Desarrollo de sistemas que permitan disponer de un back-up de energía integrando el uso de supercapacitores.
- Desarrollo de un sistema que permita mejorar la autonomía de carga con la integración de supercapacitores para mejorar los tiempos de carga del sistema.

### 2.3.3. Banco de Supercapacitores

Dado que los sistemas de potencia manejan tensiones más elevadas que los valores nominales de un solo SC, es necesario recurrir a los bancos de SC. Un banco de SC se define como un conjunto de supercapacitores agrupados, ya sea en serie para aumentar la tensión o en paralelo para aumentar la capacidad, permitiendo su integración en sistemas de mayor potencia y satisfaciendo requisitos que un solo supercapacitor no podría cumplir por sí solo.

Para llevar a cabo el proyecto, se optó por utilizar un banco de SC previamente desarrollado por el Instituto de Investigaciones en Electrónica, Control y Procesamiento de Señales (LEICI), debido a el buen rendimiento que este ofrece y su disponibilidad. Este banco de SC, está compuesto por un conjunto de 20 capacitores con el objetivo de alcanzar la tensión nominal de operación del banco, establecida en  $27V$ . Esta configuración se logró mediante la conexión en serie de 10 supercapacitores del modelo "Powerstore XV series", cada uno con una capacidad nominal de  $400F$  y una tensión nominal de  $2,7V$ . Es importante resaltar que este modelo de SC es de doble capa de carbono con un electrolito de aerogel a base de agua. Además de la disposición en serie, el banco de SC posee capacitores en paralelo con el propósito de aumentar la capacidad de almacenamiento de energía en el banco. La disposición de los SC en el banco desarrollado por el LEICI se muestra en la figura 2.4:

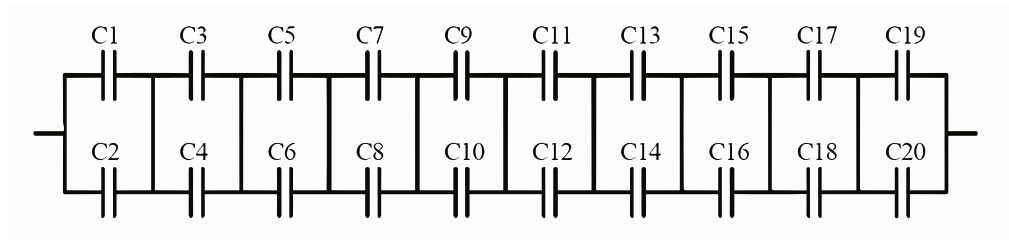


Figura 2.4 Arreglo de SC implementado.

Como consecuencia de las tolerancias que aparecen durante el proceso de fabricación, es que surgen desbalances entre los SC, algunos de los cuales pudieran ser destructivos para los mismos. Es por ello que, el prototipo utilizado contiene un método de balanceo para las celdas del banco, con el objetivo de mantener estable la tensión de trabajo de los SC y resguardar tanto la salud de los mismos así como la integridad del usuario. Para una explicación más detallada sobre la construcción del banco de SC implementado, se recomienda consultar la siguiente bibliografía [9]. A continuación, se presenta el banco SC desarrollado por el LEICI, el cual está equipado con un sistema de seguridad integrado:

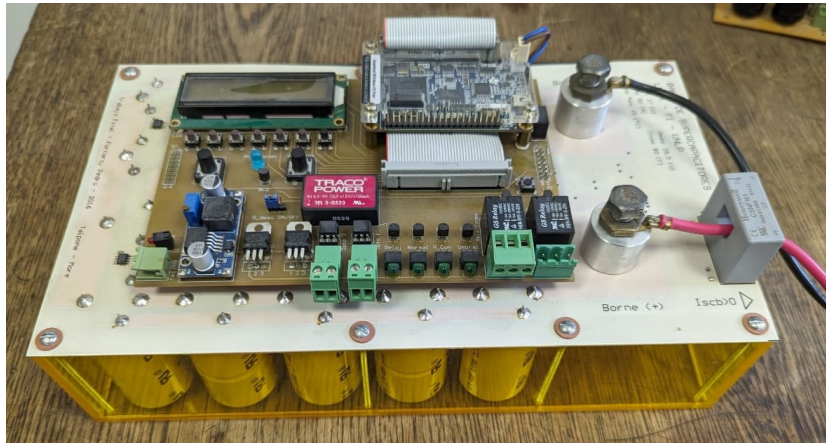


Figura 2.5 Banco de SC utilizado.

## 2.4. Baterías Ion-Litio

El contexto energético global se encuentra en una etapa de transformación impulsada por la creciente conciencia sobre la necesidad de reducir las emisiones de gases de efecto invernadero y abordar el cambio climático.

En este contexto, las baterías de litio están emergiendo como reemplazo a las baterías de níquel-cadmio debido a la alta toxicidad y el bajo rendimiento de estas últimas, mientras que las baterías de plomo comienzan a desaparecer paulatinamente. Argentina, en particular, se encuentra en una posición estratégica en este contexto global debido a que su geografía única incluye salares de alta concentración de litio, un recurso esencial en la fabricación de baterías de Ion-Litio [10].

Estas baterías se destacan por su bajo efecto de auto-descarga, conservando su carga por largos periodos de tiempo, y presentan un buen funcionamiento a alta capacidad, así como también en condiciones de bajas temperaturas.

### 2.4.1. Principio básico de funcionamiento

El funcionamiento de la batería de Ion-Litio se sustenta en procesos denominados "inserción-desinserción" de iones de litio. Estas reacciones ocurren en estado sólido entre dos compuestos de inserción que actúan como electrodos. Es importante destacar que estas baterías son estables en presencia de aire húmedo, lo que significa que no requieren condiciones de ensamblaje rigurosas como aquellas que involucran litio metálico como electrodo.

Durante el proceso de carga, un potencial eléctrico oxida los centros metálicos del cátodo y en consecuencia, el material catódico empleado libera una porción de sus iones de litio, los cuales se desplazan a través del electrolito hasta intercalarse en el ánodo, donde permanecen hasta que se inicia el proceso de descarga (ver figura 2.6).

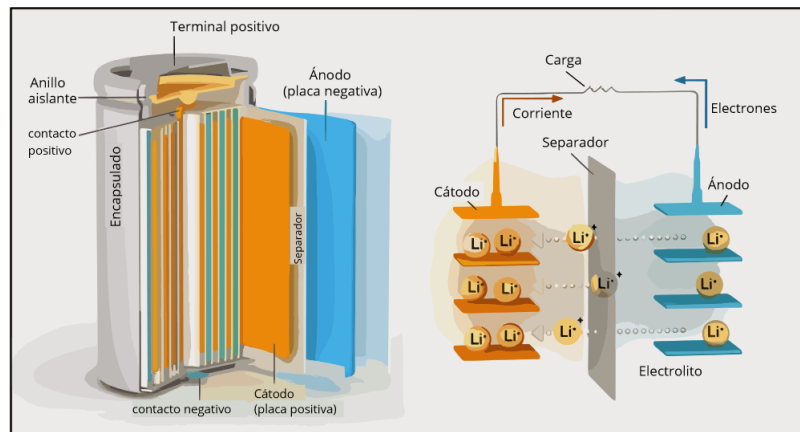


Figura 2.6 Esquema batería Ion-Litio.

En la descarga, el ánodo libera iones de litio (cargados positivamente) y electrones hacia el circuito externo, experimentando una oxidación en el proceso. Por otro lado, el cátodo acepta los iones y electrones provenientes del circuito externo, generando una reducción en el proceso. El movimiento de electrones a través del circuito externo, en los procesos de oxidación y reducción del cátodo corresponden a la corriente eléctrica. La nomenclatura de las baterías suele hacer referencia al material utilizado en el cátodo.

Cabe destacar que, el electrolito no participa en la reacción química, sino que actúa como un aislante para las cargas eléctricas y como un conductor para los iones. Su función principal es separar los electrodos y facilitar la transferencia de carga entre el ánodo y el cátodo dentro de la celda.

#### 2.4.2. Baterías de Litio-Ferrofosfato

La batería de Litio-ferrofosfato ( $\text{LiFePO}_4$  o LFP) es una variante de la batería de litio convencional donde este material se sustituye en su mayoría por laminas de fosfatos de hierro. Estas laminas se concentran en el cátodo mientras que el ánodo esta compuesto por pequeñas partículas de litio.

Las principales ventajas de esta tecnología incluyen su robustez, que preserva la vida útil independientemente de la forma de recarga, y su eficiencia de ciclo completo superior, alcanzando aproximadamente el 92 %. Asimismo, ofrece ahorros significativos en espacio y peso, hasta un 70 % en comparación con las baterías de plomo ácido. Su flexibilidad para la carga, amplio rango de temperatura, excelente rendimiento cíclico y baja resistencia interna la convierten en una solución eficaz y versátil. Se destacan por su sostenibilidad, ya que los materiales constitutivos no son tóxicos y su procesamiento al final de la vida útil presenta riesgos mínimos. Con una larga vida útil y la capacidad de mantener el voltaje de salida hasta la descarga completa, la  $\text{LiFePO}_4$  surge como

una elección integral y de alto rendimiento para aplicaciones de tracción eléctrica, eliminando la necesidad de mantenimiento continuo.

Por estas características y teniendo en cuenta la ventaja adicional de que en el instituto se dispone de dos BL de 24V, cada una de ellas construidas con ocho celdas de batería de tipo  $LiFePO_4$  (figura 2.7), en serie de 50Ah, fue de las principales razones por las que se ha optado incorporarlas en este proyecto.

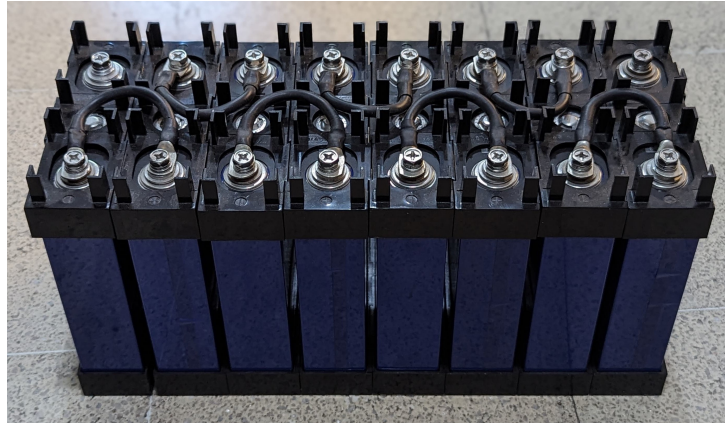


Figura 2.7 Batería Ion-Litio disponible en el LEICI.

### 2.4.3. Proceso de Carga y Descarga

El proceso de carga para una batería de Li-ion, compuesta por un conjunto de celdas en serie, se divide en tres fases:

- **Etapa de Corriente Constante (CC):** Durante esta fase, el cargador suministra corriente continua a la batería con una tensión que aumenta de manera constante hasta alcanzar el límite de tensión por celda.
- **Etapa de Balanceo:** En esta etapa, el cargador reduce la corriente de carga mientras el *Sistema de Gestión de Batería* (BMS) iguala el estado de carga de las celdas individuales. Las técnicas de balanceo pueden variar entre diferentes cargadores.
- **Fase de Tensión Constante:** Finalmente, el cargador aplica un voltaje igual a la tensión nominal de las celdas multiplicado por la cantidad de celdas en serie. La corriente disminuye gradualmente hacia cero hasta alcanzar un umbral especificado por el cargador.

Durante la descarga, los electrones fluyen desde el electrodo negativo (ánodo) hacia el electrodo positivo (cátodo) a través del circuito externo. La reacción de descarga reduce el potencial químico de la celda, transfiriendo así energía de la celda al lugar donde se disipa (principalmente en el circuito externo).

Ambos electrodos permiten que los iones de litio ingresen y salgan de sus estructuras mediante un proceso conocido como inserción o extracción, respectivamente.

Para mayor información sobre la construcción de un sistema de carga para celdas de litio-ferrofosfato se puede consultar la siguiente bibliografía [11].

#### 2.4.4. Aplicaciones

Debido a su compacto tamaño, su ligero peso, alta densidad de energía y prolongada vida útil, las baterías de iones de litio son una de las opciones principales para dispositivos de atención médica. Dispositivos como marcapasos, transmisores de radio implantables, estimuladores gástricos, biosensores portátiles, herramientas quirúrgicas, desfibriladores y monitores son solo algunos ejemplos de aplicaciones que se benefician de estas baterías [12].

Sin embargo, estas baterías también desempeñan un papel importante en vehículos eléctricos, dado que, dicha industria ha experimentado un crecimiento significativo en los últimos años, impulsado por políticas gubernamentales que buscan reducir las emisiones de gases contaminantes.

La selección del tipo de batería depende exclusivamente de los requerimientos de la aplicación donde son utilizadas. Es por ello que, la figura 2.8 ilustra la versatilidad de las baterías de litio en diversas aplicaciones, destacando su menor peso (densidad de energía gravimétrica) y mayor capacidad de energía por unidad de volumen (densidad de energía volumétrica) en comparación con otros tipos de baterías.

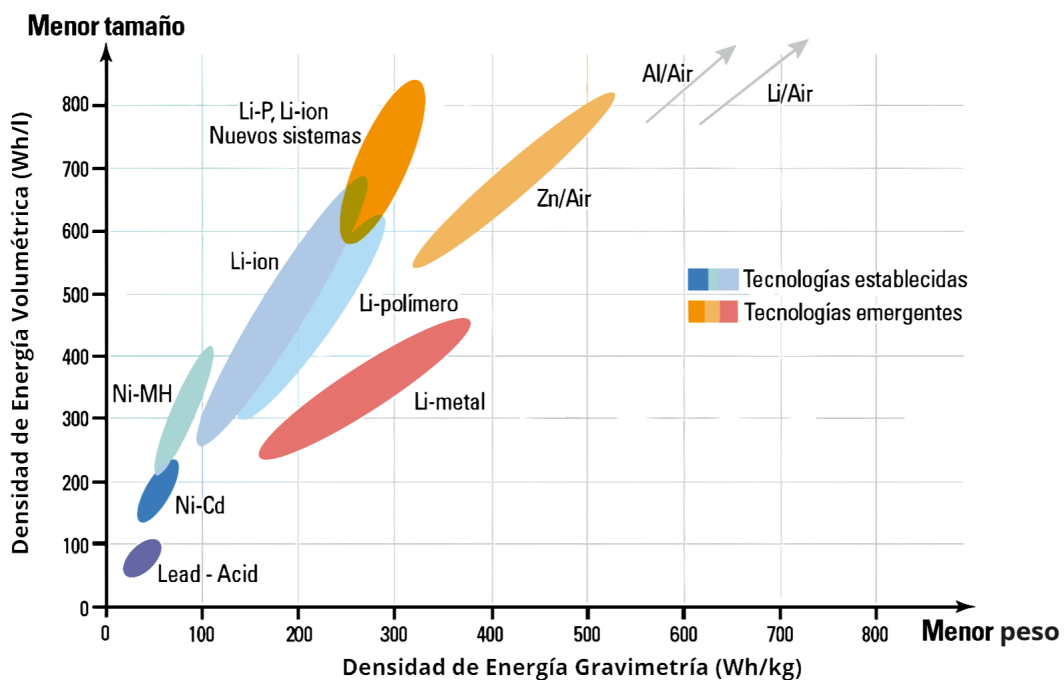


Figura 2.8 Densidades de energía en tipos de baterías: gravimétrica y volumétrica.

Siendo las baterías de Li-ion las empleadas con mayor frecuencia en aplicaciones para el cuidado de la salud, vehículos eléctricos y aeronaves por las características que estas presentan [13].

## 2.5. Sistema híbrido de almacenamiento de energía

Debido a las características explicadas en las secciones anteriores, la combinación del banco de supercapacitores y las baterías de ion-litio proporciona una solución integral para el sistema de almacenamiento de energía en el proyecto. El siguiente diagrama de Ragone, utilizado para realizar comparaciones entre fuentes de almacenamiento, muestra las diferencias entre ambos sistemas:

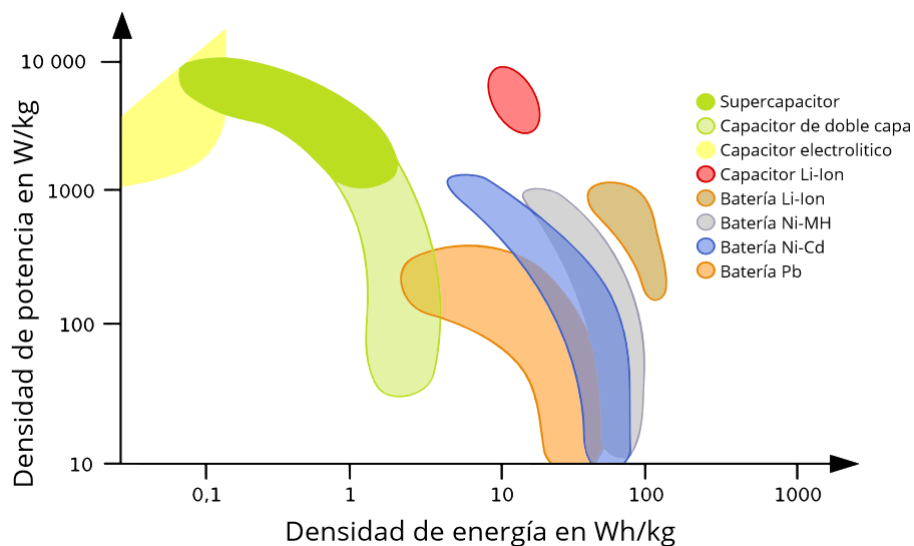


Figura 2.9 Diagrama de Ragone.

El eje horizontal denota la densidad de energía específica [J/kg], reflejando la capacidad de almacenar energía, mientras que el eje vertical indica la densidad de potencia específica [W/kg], ilustrando la velocidad con la cual esa energía puede ser entregada.

De esta manera, en el contexto de un sistema híbrido, la BL se encargará de satisfacer la demanda de corriente asociada a la potencia media requerida por el bus de continua. Mientras que por el contrario, los SC serán los encargados de lidiar con las variaciones abruptas de corriente demandadas por la carga. Esto por consecuencia, brindará el incremento de la vida útil de la BL dado que los picos de corrientes serán entregados por el SC disminuyendo las exigencias de la batería.

## 2.6. Resumen

En este capítulo, se ha llevado a cabo un análisis de los diversos sistemas de almacenamiento de energía, resaltando las virtudes de los elementos disponibles para la realización del proyecto: las BL y los SC. Se profundizó en el estudio de sus características, modo de funcionamiento y sus principales aplicaciones, así como en las diferencias en los tiempos de entrega de energía y la potencia que estos pueden almacenar. Sin embargo, la formación de este sistema híbrido implica la integración de convertidores electrónicos de potencia. Estos dispositivos son esenciales para ajustar los niveles de tensión de ambas tecnologías y facilitar su funcionamiento en conjunto. Por ende, en el próximo capítulo se detallará el análisis de dichos convertidores.

## Capítulo 3

# Convertidores CC-CC

### 3.1. Introducción

En el presente capítulo se desarrollará la importancia de los convertidores (CC-CC) en sistemas híbridos de almacenamiento con aplicaciones en vehículos eléctricos (VE). También se analizará la construcción, y características que brinda la topología utilizada.

Los convertidores actúan como puentes de transferencia de energía entre fuentes y cargas, ambas de corriente continua, que no son compatibles en términos de tensión y corriente. Estos pueden dividirse en lineales y conmutados. Los convertidores conmutados son una alternativa eficiente respecto de los reguladores lineales. Estos dispositivos minimizan las pérdidas de energía al modificar de manera controlada el flujo de energía, reduciendo la disipación de calor y mejorando la eficiencia global del sistema. Además, los convertidores conmutados son más versátiles, permiten un control más preciso de la tensión y la corriente, y son ideales para aplicaciones donde la eficiencia energética es crucial. En contraste, los reguladores lineales generan más calor y son menos eficientes, limitando su utilidad en sistemas que requieren una gestión eficiente de la energía.

En un circuito de un convertidor conmutado, los transistores operan como llaves, estando completamente prendidos o completamente apagados. Asumiendo que la llave es ideal en la Figura 3.1, la salida es igual que la entrada cuando la llave está cerrada, y nula cuando la llave está abierta.

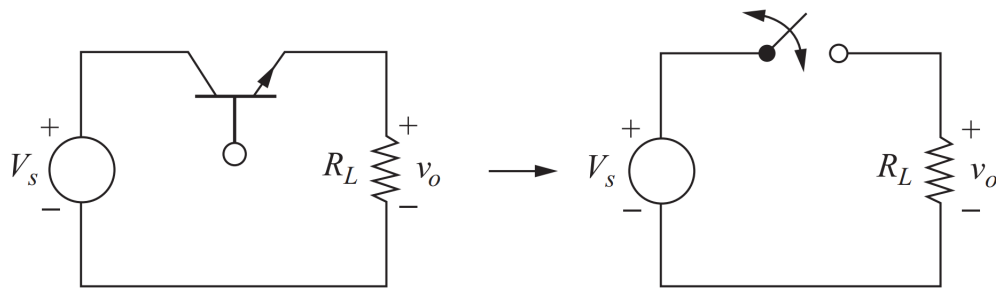


Figura 3.1 Convertidor conmutado CC-CC básico y su equivalente de conmutación.

La conmutación periódica de la llave resulta en la salida en pulsos formada en la Figura 3.2.

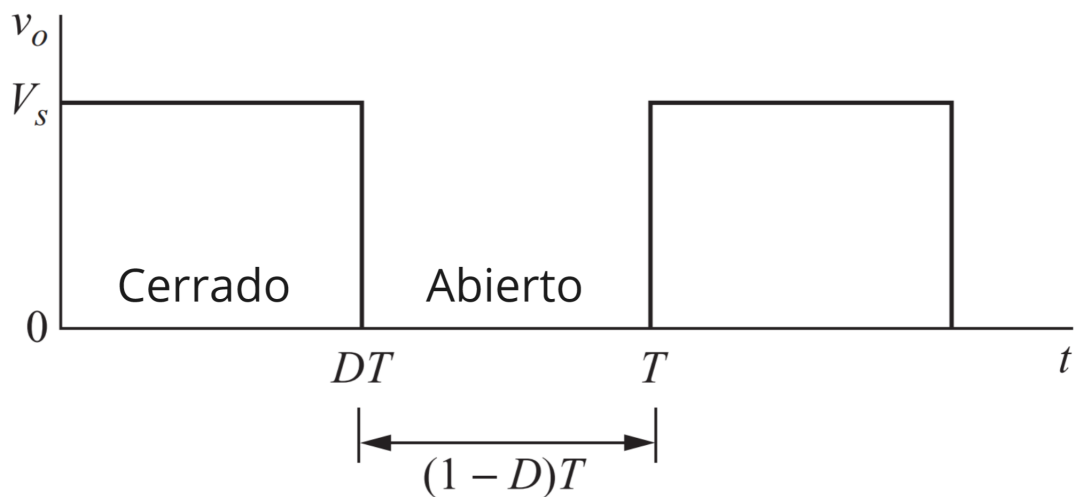


Figura 3.2 Tensión de salida del circuito de la figura 3.1.

El valor medio de la salida es:

$$V_o = \frac{1}{T} \int_0^T v_o(t) dt = \frac{1}{T} \int_0^{DT} V_s dt = V_s D \quad (3.1)$$

La componente de continua de la tensión de salida es controlada a través del ajuste del tiempo que la llave se encuentra respecto del periodo de conmutación total. Esta fracción de tiempo es llamada ciclo de trabajo y se representa con la letra  $D$ .

$$D = \frac{t_{on}}{t_{on} + t_{off}} = \frac{t_{on}}{T} = t_{on} f \quad (3.2)$$

En donde  $f$  es la frecuencia de conmutación. En este circuito elemental, la componente de continua de la tensión de salida siempre será menor o igual que la tensión de entrada del circuito. Idealmente, la potencia absorbida por la llave es cero. Cuando la

llave está abierta, no circula corriente por ella; cuando está cerrada, no cae tensión. Por lo tanto, toda la potencia es absorbida por la carga, y la eficiencia es del cien por ciento. Con una llave real, las pérdidas son inevitables, ya que ocurre una caída de tensión cuando está cerrada, y debe pasar por la región lineal cuando realiza la conmutación de un estado a otro.

En este capítulo se presenta el convertidor elevador, el cual es utilizado en este trabajo. Para mas detalles sobre los análisis realizados se puede consultar [14].

### 3.2. Convertidor Elevador

El convertidor elevador (*boost converter* en ingles) es mostrado en la Figura 3.3. Es llamado de tal manera ya que la tensión de salida es mayor o igual que la de entrada. Este tipo de convertidor es el utilizado para el proyecto, ya que es necesario elevar la tensión del banco de supercapacitores y las baterías de litio.

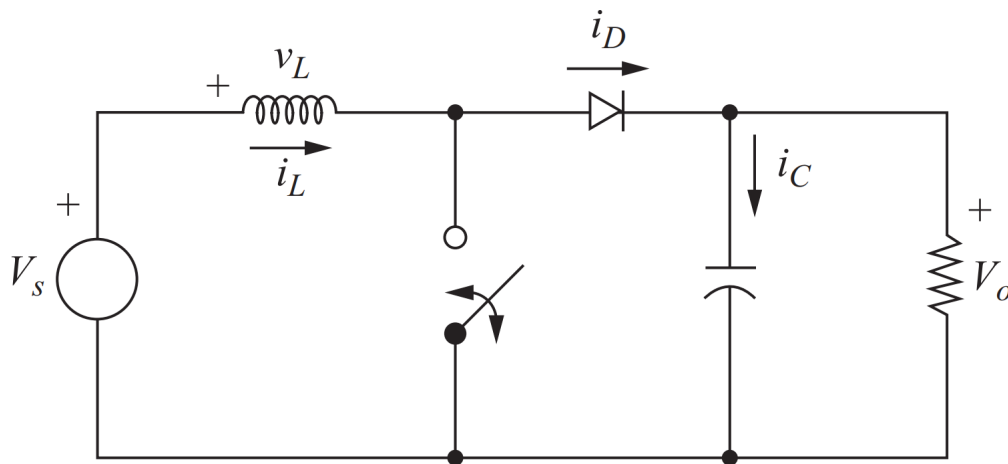


Figura 3.3 Convertidor elevador.

Antes de proceder con el análisis del modo de conducción continua (MCC) del convertidor elevador, es necesario realizar las siguientes suposiciones:

1. Existen condiciones de estado estacionario.
2. El periodo de conmutación es  $T$ , y la llave esta cerrada por un tiempo  $DT$  y abierta por un tiempo  $(1-D)T$ .
3. La corriente por el inductor no se anula y es siempre positiva.
4. El capacitor posee una capacitancia muy grande, y la tensión de salida es mantenida constante a un valor  $V_o$ .
5. Los componentes son ideales.

### 3.2.1. Modo conducción continua

Este análisis es hecho a través de la examinación de la tensión y corriente del inductor cuando la llave está cerrada, y nuevamente cuando la llave se encuentra abierta.

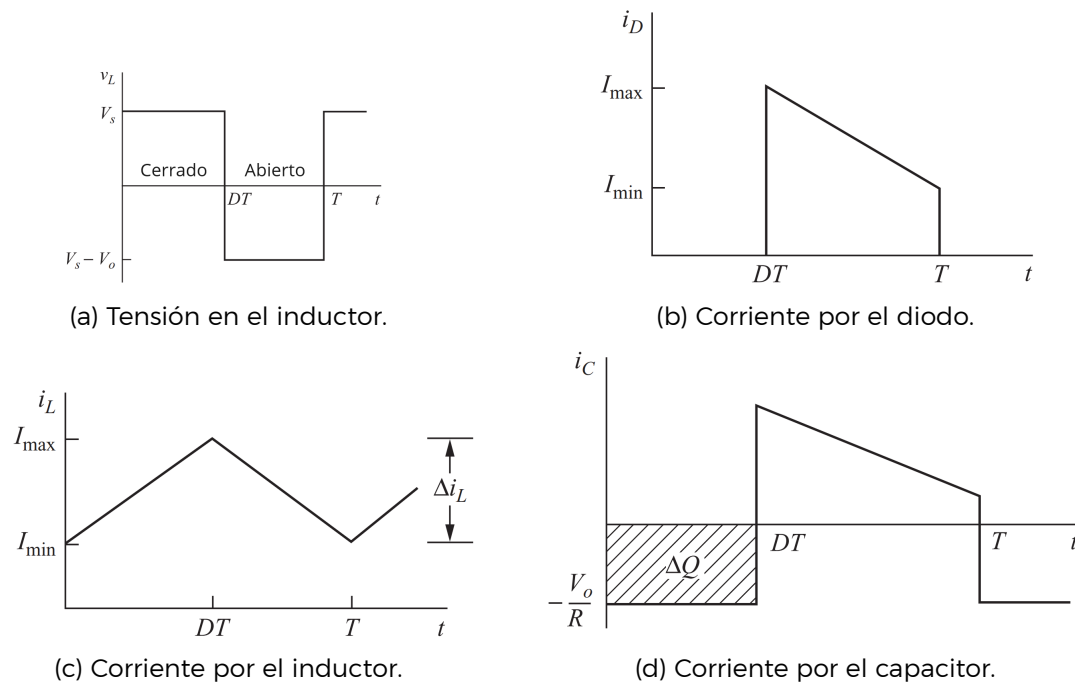


Figura 3.4 Formas de onda del convertidor elevador.

**Análisis con la llave cerrada** Cuando la llave se encuentra cerrada, el diodo se encuentra polarizado inversamente. La ley de Kirchhoff sobre el camino que contiene la fuente, el inductor, y la llave es:

$$v_L = V_s = L \frac{di_L}{dt} \quad \frac{di_L}{dt} = \frac{V_s}{L}$$

La derivada de la corriente es una constante, lo que significa que la corriente incrementa linealmente cuando la llave está cerrada, como se muestra en la Figura 3.4c. Por lo tanto, resolviendo para la tasa de cambio de la corriente del inductor  $\Delta i_L$  resulta en:

$$\boxed{(\Delta i_L)_{\text{cerrada}} = \frac{V_s DT}{L}} \quad (3.3)$$

**Análisis con la llave abierta** Cuando la llave se abre, la corriente del inductor no puede cambiar instantáneamente, así que el diodo se polariza directamente para proveer un camino a  $i_L$ . Asumiendo que la tensión de salida  $V_o$  es constante, la tensión que cae en el inductor es:

$$v_L = V_s - V_o = L \frac{di_L}{dt} \quad \text{o} \quad \frac{di_L}{dt} = \frac{V_s - V_o}{L}$$

La derivada de  $i_L$  nuevamente es una constante, entonces la corriente debe cambiar linealmente mientras la llave se encuentra abierta. La tasa de cambio en el inductor mientras la llave se encuentre en este estado es:

$$\frac{\Delta i_L}{\Delta t} = \frac{\Delta i_L}{(1-D)T} = \frac{V_s - V_o}{L}$$

Resolviendo para  $\Delta i_L$ :

$$\boxed{(\Delta i_L)_{abierto} = \frac{(V_s - V_o)(1-D)T}{L}} \quad (3.4)$$

En estado estacionario, la corriente media en el inductor debe ser cero. Utilizando las Ecuaciones (3.3) y (3.4),

$$(\Delta i_L)_{cerrada} - (\Delta i_L)_{abierto} = 0$$

$$\frac{V_s DT}{L} - \frac{(V_s - V_o)(1-D)T}{L} = 0$$

Y resolviendo para  $V_o$ ,

$$V_s(D + 1 - D) - V_o(1 - D) = 0$$

$$\boxed{V_o = \frac{V_s}{(1-D)}} \quad (3.5)$$

La Ecuación (3.5) demuestra que si la llave siempre está abierta y  $D$  es nula, la tensión de salida será la misma que la de la entrada. Si el ciclo de trabajo se va incrementando, el denominador se va haciendo más chico, resultando en una tensión de salida cada vez más grande. Por lo tanto, se deduce que el convertidor elevador produce una tensión de salida siempre mayor o igual que la tensión de entrada.

A su vez, según esta ecuación, si el ciclo de trabajo se aproxima a 1, la tensión de salida se hace infinita. Es necesario tener en cuenta que la deducción de la Ec. (3.5) fue realizada a partir de la suposición de componentes ideales. Los componentes reales, al poseer pérdidas, previenen tal evento.

La corriente media por el inductor es determinada al reconocer que la potencia media entregada por la fuente debe ser la misma que la potencia media absorbida por la carga. La potencia de salida es:

$$P_o = \frac{V_o^2}{R} = V_o I_o$$

Y la potencia de entrada es  $V_s I_s = V_s I_L$ . Igualando las potencias de salida y entrada y utilizando la Ecuación (3.5):

$$V_s I_L = \frac{V_o^2}{R} = \frac{[V_s/(1-D)]^2}{R} = \frac{V_s^2}{(1-D)^2} R$$

Realizando algunas sustituciones y resolviendo para la corriente media del inductor,  $I_L$  puede ser expresada como:

$$I_L = \frac{V_s}{(1-D)^2 R} = \frac{V_o^2}{V_s R} = \frac{V_o I_o}{V_s} \quad (3.6)$$

### 3.2.2. Convertidor elevador bidireccional en corriente

Este tipo de convertidor es más versátil que la topología elevadora convencional, dado que tiene la capacidad de transferir energía en ambos sentidos, de la entrada a la salida y viceversa, efectuando únicamente un cambio en el sentido de la corriente.

Un convertidor elevador bidireccional en corriente se construye sustituyendo el diodo que se encuentra en la topología elevadora convencional por un transistor controlado que permita el flujo de corriente en ambas direcciones. En particular, durante el proceso de fabricación de un transistor MOSFET se forma una juntura p-n entre los terminales de drain y source, permitiendo que la corriente pueda establecerse en la dirección opuesta source-drain. De esta forma se obtiene la bidireccionalidad de corriente a través de la llave. La topología resultante se muestra en la Figura 3.5.

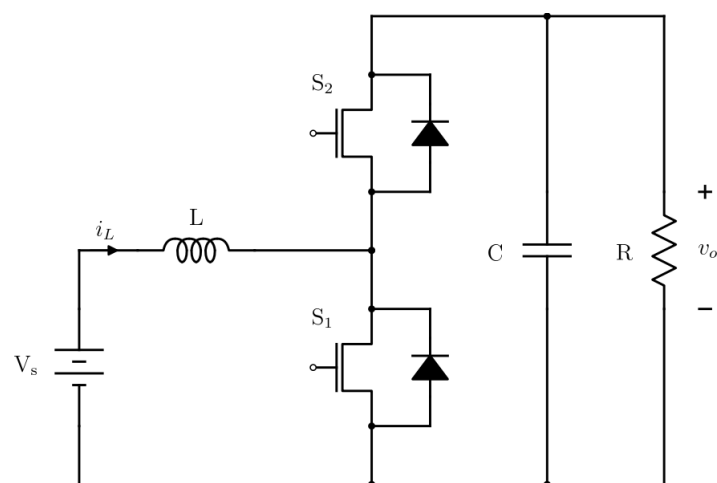


Figura 3.5 Convertidor elevador bidireccional en corriente.

Un ejemplo práctico de la aplicación de convertidores bidireccionales que facilita la comprensión de su utilidad se encuentra en el sistema de distribución de un vehículo híbrido. El vehículo tiene esencialmente dos modos de operación:

- Modo marcha: Para ponerse en marcha, el vehículo toma energía del banco de supercapacitores durante la aceleración y de las baterías de litio una vez estabilizada su velocidad. Durante este proceso, el convertidor transfiere energía desde los módulos de almacenamiento hacia el vehículo.
- Modo regenerativo: Una vez que el vehículo está en marcha, existe una regeneración de energía que es devuelta a los módulos de almacenamiento por parte del sistema electromecánico cuando se empieza el frenado. Durante este proceso, el convertidor transfiere energía desde el vehículo hacia los módulos de almacenamiento.

El ejemplo anterior pone de manifiesto la gran versatilidad que se obtiene al modificar la topología elevadora clásica para lograr bidireccionalidad en el flujo de energía.

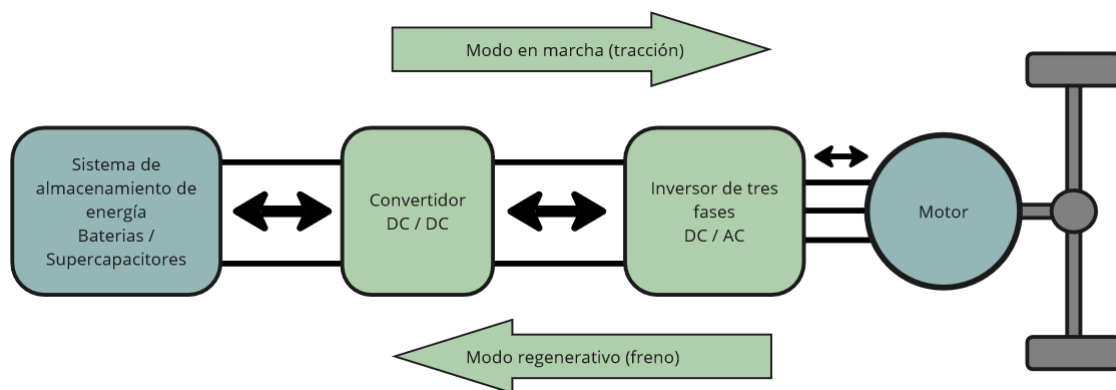
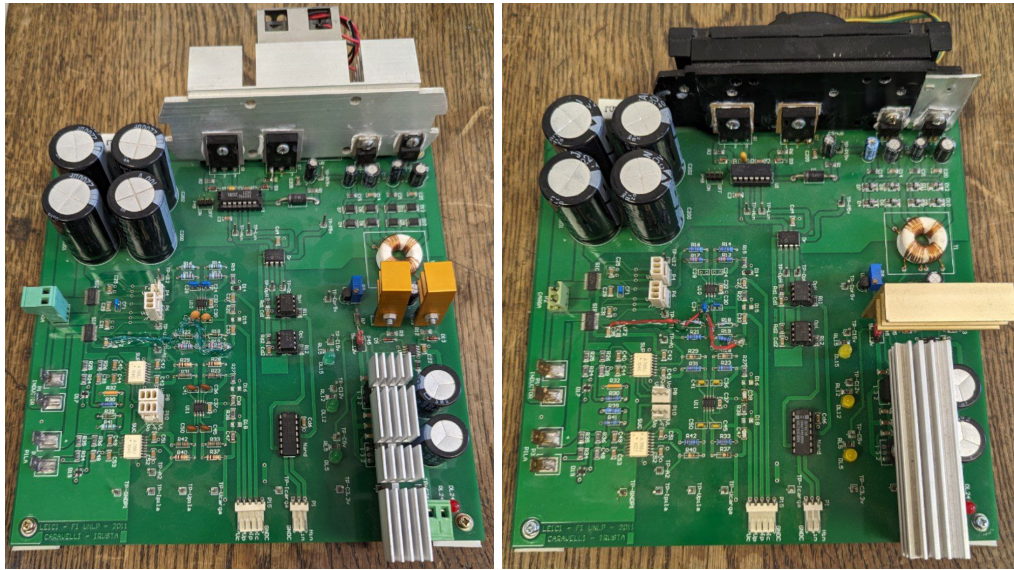


Figura 3.6 Sistema eléctrico para un VE basado en baterías.

En este trabajo se utilizan dos convertidores CC-CC bidireccionales en corriente construidos íntegramente en el instituto LEICI, para mayor información se puede consultar en la bibliografía [15]. Estos cuentan con una etapa de instrumentación que permite la medición de las corrientes y tensiones entrantes y salientes. En particular para este proyecto, se rediseño y optimizó la etapa de acondicionamiento de señal buscando aprovechar el máximo rango de lectura del ADC (Convertor Analógico Digital), y se ajustó el ancho de banda de los filtros analógicos a una frecuencia de corte de 10 kHz. Por otra parte, se rediseño el circuito para el sensado de pila y carga, adaptándolo a tensiones máximas de 30V y 80V, respectivamente. En las Tablas 3.1 y 3.2 se muestran las especificaciones técnicas de los convertidores mencionados.



(a) Convertidor 1.

(b) Convertidor 2.

Figura 3.7 Convertidores utilizados en el proyecto.

Especificación	Valor	Unidad
Potencia máxima	300	W
Tensión continua en la salida	80	V
Tensión nominal en la entrada	30	V
Corriente nominal en la entrada	15	A
Rizado pico a pico de la corriente	3.33	A
Frecuencia de conmutación	20	kHz
Inductancia	200	H
Corriente máxima por el inductor	16.67	A
Capacitancia	2200	$\mu F$

Tabla 3.1 Parámetros eléctricos de los convertidores CC-CC

Componente	Denominación	Comentarios
Núcleo del inductor	ETD-5922 CF-138	Permeabilidad relativa $\mu_r$ de 2500
Capacitor	TREC	Tensión máxima de trabajo de 100V
Interruptores	IRFP-250	MOSFET con corriente máxima de drenador de 30A
Drivers	IR2110	Tiempo de encendido y apagado de 120ns y 94ns respectivamente
Diodo de bootstrap	MUR 460	Capacitancia de $1\mu F$ con resistencia de $2,2\Omega$

Tabla 3.2 Componentes eléctricos de los convertidores CC-CC

Cabe destacar una de las ventajas al momento de analizar el sistema planteado en este proyecto. Dado que se trata de dos convertidores bidireccionales conectados a un bus de continua y que los objetivos de control de estos son complementarios, es decir que las BL entreguen la potencia media de la carga y los SC entreguen los picos de potencia (transitorios). La mayor parte del tiempo la corriente media entregada por los SC sera 0A, pero al tener convertidores bidireccionales la corriente por el inductor nunca se queda en 0A, es decir que se puede tratar este convertidor con actuación en MCC en todo momento.

Al momento de analizar las dinámicas se debe tener en cuenta que al utilizarse ambos convertidores en paralelo los capacitores de salida se suman, por lo tanto supondría una dinámica mas lenta que en la actuación independiente de estos.

### 3.3. Resumen

En este capítulo se han presentado conceptos básicos de los convertidores conmutados y analizado específicamente el funcionamiento de la topología de convertidor elevador CC-CC bajo la condición de conducción continua. También realizando una leve modificación en su estructura se introdujo el convertidor elevador CC-CC bidireccional en corriente. Finalmente, se presentan específicamente los convertidores utilizados en este proyecto y como se modifica la forma de realizar el análisis de estos al interactuar simultáneamente en el sistema planteado.

## Capítulo 4

# Field-Programmable Gate Arrays

### 4.1. Introducción

Los dispositivos *Field Programmable Gate Arrays* (FPGA), conocidos como arreglos de compuertas programables en el campo, son dispositivos semiconductores que ofrecen una flexibilidad única en el diseño de circuitos digitales. Como indica su nombre, consisten en una matriz de bloques lógicos configurables (CLBs) y conexiones interconectadas, lo que permite a los diseñadores implementar una variedad de funciones lógicas y circuitos complejos. Estos componentes lógicos pueden emular puertas lógicas básicas como AND, OR, XOR, NOT, así como funciones combinacionales más sofisticadas como decodificadores y operaciones matemáticas simples. Además, algunos FPGA también incluyen elementos de memoria, como flip-flops y bloques de memoria, que amplían sus capacidades de procesamiento y almacenamiento.

Aunque las FPGA suelen ser más lentas que los ASIC (circuitos integrados de aplicación específica), tienen ventajas significativas, como una rápida salida al mercado de productos, la capacidad de reprogramación para corregir errores después del lanzamiento y la reducción de costos de ingeniería. Los vendedores ofrecen diferentes versiones de FPGA para adaptarse a diversas necesidades, desde opciones económicas y menos flexibles hasta modelos más avanzados y personalizables.

En este capítulo, se profundiza en las características de los FPGA, incluyendo su arquitectura y los elementos básicos que la componen. Se describen los lenguajes de programación utilizados para un FPGA y los paradigmas asociados proporcionando al lector una comprensión integral de este componente en el diseño de sistemas digitales.

### 4.2. Arquitectura

Generalmente, la arquitectura de un FPGA varía de un fabricante a otro generando una constante discusión sobre cuál ofrece la mejor. Sin embargo, se han llevado a ca-

bo numerosos estudios comparativos (*benchmarks*) para demostrar la calidad de cada diseño. A pesar de estas diferencias, existen muchos componentes en común entre los FPGA de distintos fabricantes, tales como los CLBs (*Configurable Logic Block*), bloques para ejecución de MACs (*Multiplicador-Acumulador o bloques DSPs*), bloques de control de reloj (generación de frecuencias a partir de una frecuencia base, corrimiento de fase), bloques de entrada/salida, entre otros. Por estas razones, ésta sección se centrará en explicar lo principal de las mismas comenzando por un esquema ilustrativo de la arquitectura básica de un FPGA:

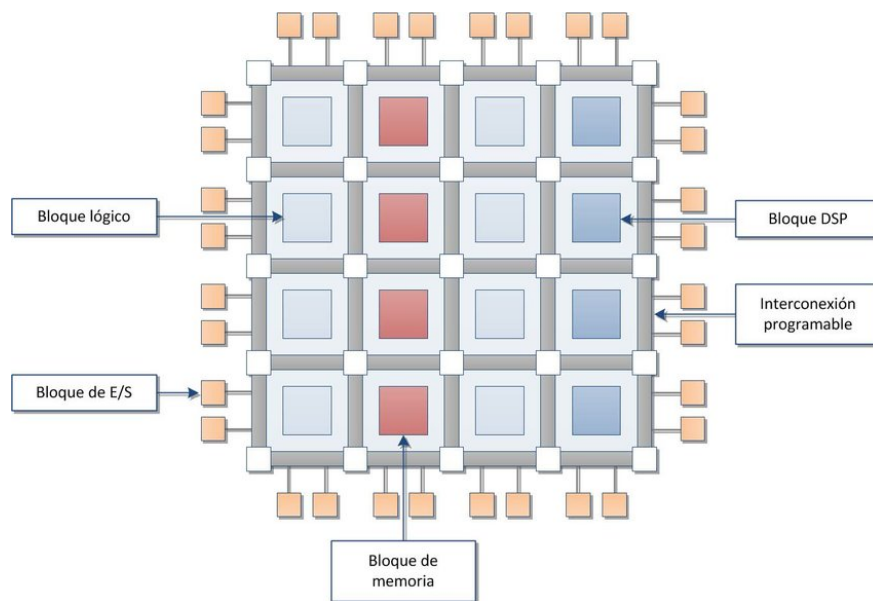


Figura 4.1 Arquitectura básica de un FPGA.

La cuál consiste en un arreglo de CLBs y canales de comunicación, donde los conectores de entrada/salida se distribuyen a lo largo de las filas y columnas del dispositivo.

#### 4.2.1. Bloques Lógicos configurable

Un CLB típico consta de cuatro entradas que alimentan una tabla de funciones lógicas (LookUp Table), un sumador completo (Full Adder) y un flip-flop tipo D (FFD) utilizado para almacenar el resultado. En la Figura 4.2, se observa la estructura de un CLB compuesto por una LookUp Table de 3 entradas, las cuales mediante un multiplexor, pueden combinarse para formar una LUT de 4 entradas. Además, la salida de estas celdas lógicas puede ser sincrónica o asincrónica, dependiendo del multiplexor restante.

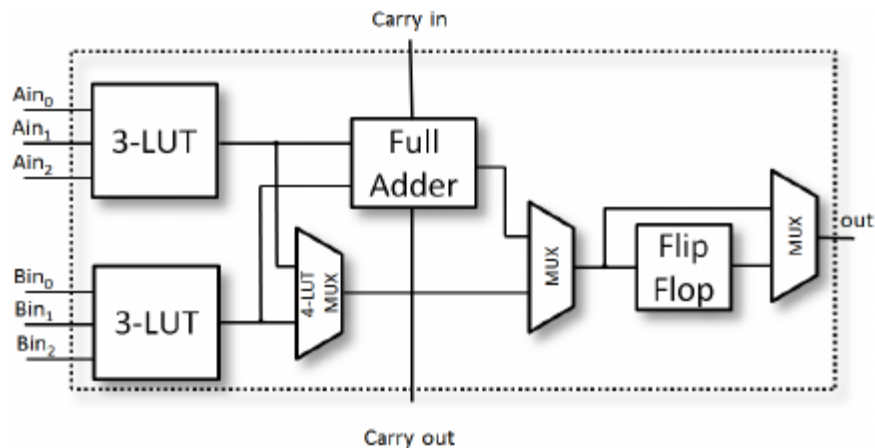


Figura 4.2 Bloque lógico de un FPGA.

### 4.3. Lenguaje de descripción de hardware

Los lenguajes de descripción de hardware (HDL) surgieron en la década de 1980 como una respuesta a la necesidad de describir, simular y sintetizar el hardware en FPGA. Hasta entonces, predominaban los lenguajes de dominio específico y las herramientas de captura esquemática para el desarrollo de sistemas digitales. VHDL y Verilog se establecieron como los principales HDL para FPGA, con cada uno teniendo su propia trayectoria de desarrollo.

VHDL, fue creado por el Departamento de Defensa de EE.UU. en 1981. Este lenguaje tuvo un enfoque inicialmente orientado a la descripción y documentación de la estructura y el comportamiento de sistemas. Con el tiempo, se convirtió en un estándar IEEE en 1987 y se utilizó para simulaciones y síntesis de circuitos digitales, experimentando numerosas modificaciones desde entonces.

Por otro lado, Verilog, inicialmente desarrollado por Gateway Design Automation en 1984, surgió como un lenguaje propietario para la simulación de funcionalidades lógicas. Con la adquisición de Gateway por parte de Cadence y su transformación en un estándar abierto bajo el control de Open Verilog International, Verilog evolucionó hacia lo que hoy se conoce como SystemVerilog (IEEE STD 1800-2009).

Aunque Verilog y VHDL comparten principios similares, difieren en su sintaxis, siendo VHDL más detallado y propenso a errores en comparación con Verilog. Sin embargo, independientemente del HDL elegido, la descripción de un sistema digital sigue una serie de elementos comunes:

- Especificación de la interfaz del sistema, que incluye los puertos de entrada y salida para la conexión con otros dispositivos.
- Definición del comportamiento del sistema.

- Especificación de las señales de tiempo involucradas en el sistema.
- Simulación del diseño configurado, que implica estimular el sistema con señales de entrada y analizar el comportamiento de las señales de salida.

En la siguiente Sección se presentará el lenguaje de Verilog, el cual es utilizado para este proyecto debido a su facilidad de programación y al poseer una sintaxis inspirada en el lenguaje de programación C, lo que permite facilitar la tarea de programación y acelerar el proceso de desarrollo de sistemas digitales en FPGA.

#### 4.3.1. Verilog

El objetivo fundamental de Verilog es proporcionar un medio para modelar hardware digital. A diferencia de los lenguajes de programación de software como C, Verilog se enfoca en la descripción y especificación de circuitos digitales. Cada línea de código en Verilog representa uno o más componentes de hardware en el diseño, ofreciendo diversos niveles de abstracción que permiten a los diseñadores modelar, simular y finalmente implementar sistemas digitales. Sin embargo, es importante destacar que no todo el código Verilog es sintetizable, es decir, no todas las construcciones del lenguaje pueden ser traducidas directamente en hardware por las herramientas de síntesis.

Una de las características más importantes de Verilog es su capacidad para trabajar en el nivel de transferencia de registros (RTL), donde se especifica la colocación de registros en el diseño y cómo fluye la información entre ellos. La síntesis de Verilog se basa en un subconjunto del lenguaje llamado RTL, comprendido por las herramientas de síntesis para generar el diseño a nivel de compuertas.

Los diseños en Verilog se organizan en módulos jerárquicamente interconectados, donde cada módulo puede contener declaraciones e instancias de otros módulos de nivel inferior. Esta estructura modular permite un diseño ordenado y escalable, facilitando la síntesis incremental de diseños más grandes.

Los módulos en Verilog se declaran e instancian de manera similar a las clases en C++, pero sin la posibilidad de anidamiento. Cada instancia de un módulo en Verilog implica la creación de una copia física del hardware correspondiente, con sus propias compuertas lógicas, registros y conexiones. Estos se interconectan a través de puertos y pueden representar desde componentes básicos hasta sistemas completos.

#### 4.3.2. Máquina de Mealy

Una máquina de estado Mealy es un tipo de máquina de estado finito en la que las salidas se determinan tanto por el estado actual como por las entradas actuales. En este modelo, las transiciones de estado están vinculadas a las entradas, y las salidas se generan en función del estado actual y las entradas presentes. Esto implica que las salidas se actualizan inmediatamente después de una transición de estado,

lo que permite una adaptación más dinámica en comparación con una máquina de estado Moore, donde las salidas dependen únicamente del estado actual. La máquina de estado Mealy es especialmente útil para sistemas que requieren respuestas más sensibles a las entradas y donde las salidas pueden cambiar con mayor dinamismo.

En la Figura 4.3 se muestra un modelo simplificado de una Máquina de Mealy donde la actualización de las salidas se produce durante la transición de estados. A partir del estado 0, cuando se produzca la entrada  $x$  se producirá la salida  $y$  alcanzándose el estado 1.

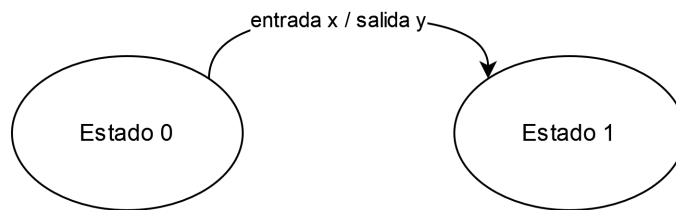


Figura 4.3 Máquina de Mealy

#### 4.4. Intel Quartus Prime

Intel Quartus Prime<sup>®</sup> representa una plataforma integral que fusiona diversas herramientas de desarrollo fundamentales para el procesamiento eficiente y accesible de diseños. Esta plataforma no solo facilita la gestión de proyectos individuales, sino que también permite abordar proyectos jerárquicos, siendo especialmente útil para diseños de alta complejidad.

Uno de los aspectos más destacados de Intel Quartus Prime<sup>®</sup> radica en su conjunto de métodos avanzados, que abarcan desde la síntesis lógica hasta la compilación, partición y simulación. Estas herramientas ofrecen a los diseñadores la capacidad de trabajar en distintas fases del proceso de diseño, desde la concepción inicial hasta la implementación definitiva en el dispositivo seleccionado.

La plataforma brinda opciones sólidas para la simulación funcional, temporal y enlazada con múltiples dispositivos. Esto otorga a los diseñadores la posibilidad de validar y verificar sus diseños antes de llegar a la fase de implementación física, lo que resulta importante para detectar y corregir posibles errores de diseño en etapas tempranas.

Para utilizar Intel Quartus Prime<sup>®</sup> de manera efectiva, se sigue un procedimiento básico que implica la introducción del diseño, su síntesis y finalmente la configuración y grabación en el dispositivo correspondiente. No obstante, es recomendable llevar a cabo simulaciones y análisis temporales del diseño antes de proceder con la implementación física permitiendo así, una mayor fiabilidad y precisión en el diseño final.

#### 4.4.1. IP cores

Los IP Cores (*Intellectual Property Cores*) constituyen una metodología clave para la expresión de la reusabilidad de códigos en HDL. Estos núcleos, diseñados por terceros, representan bloques de lógica o datos que pueden integrarse como componentes en diseños más complejos utilizando FPGA o ASIC. Organizados y disponibles en paquetes, los IP Cores pueden ser instanciados para su incorporación en otros diseños, lo que ofrece una amplia gama de funcionalidades listas para su uso. Entre los ejemplos de estos bloques se encuentran unidades de procesamiento central (CPU), unidades de comunicación como UART, I2C, SPI y Ethernet, controladores, ALU (Unidad Aritmético Lógica), interfaces, PLL (*Phase Locked Loop*) y más.

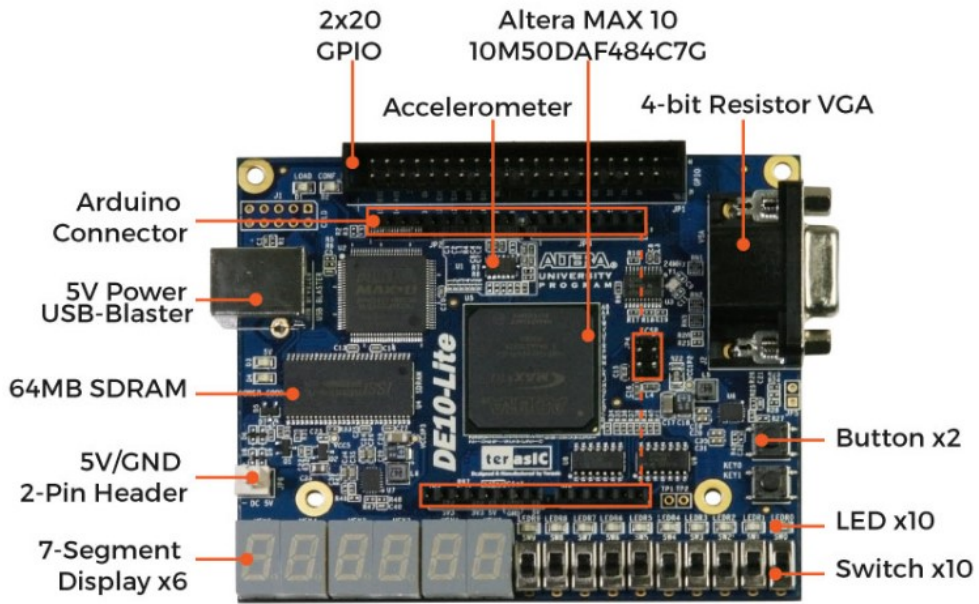
Estos IP Cores resultan necesarios para la creación de diseños más elaborados, ya que representan la descripción completa y la funcionalidad de un dispositivo mediante una declaración especial que permite su integración en diseños jerárquicos más complejos. Así, dispositivos simples como sumadores, multiplicadores, contadores y multiplexores pueden formar parte de diseños más sofisticados, donde estos componentes representan solo una parte de la arquitectura total.

### 4.5. FPGA DE10-Lite

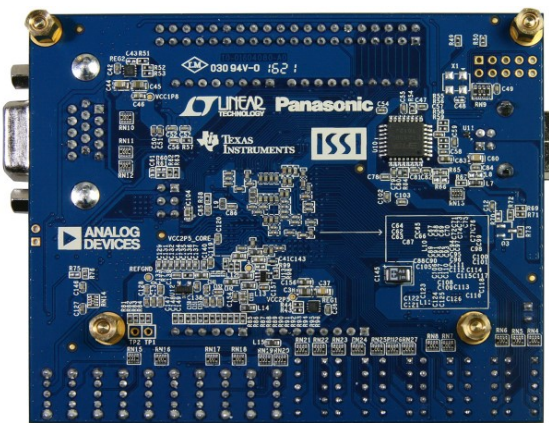
En este proyecto en particular, se optó por emplear la placa de desarrollo Terasic DE10-Lite, la cuál se encuentra disponible en el Instituto de Investigaciones en Electrónica, Control y Procesamiento de Señales (LEICI).

Terasic DE10-Lite es una placa FPGA basada en la tecnología Altera MAX 10 que cuenta con aproximadamente 50K elementos lógicos (*LEs*) y un convertidor analógico a digital (*ADC*) integrado. En términos de conectividad, la placa posee un conector GPIO 2x20 (niveles de voltaje: 3,3V) y un conector compatible con Arduino Uno R3, que incluye seis canales ADC. Además, dispone de una resistencia VGA de cuatro bits y varios componentes de entrada/salida, como LED, interruptores, botones y una pantalla de siete segmentos.

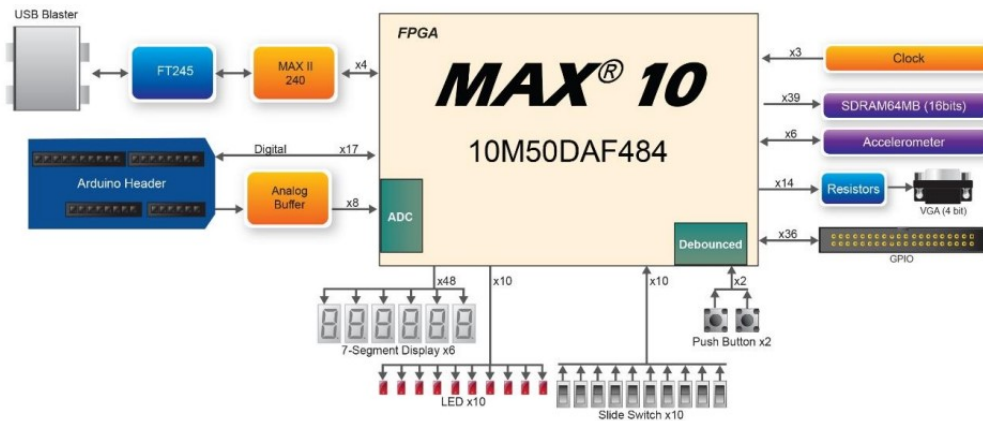
La elección de la placa de desarrollo de Altera, se considera adecuada por varias razones. Una de ellas es que posee una cantidad considerable de interruptores que permiten activar o desactivar los controladores desarrollados según sea necesario. Otra, es que cuenta con un display de 7 segmentos que posibilita la visualización de las tensiones y corrientes medidas en los convertidores de ambos sistemas de almacenamiento. Además, la presencia de LEDs en esta placa proporciona una confirmación visual inmediata sobre el funcionamiento de los controles implementados, lo que asegura un seguimiento efectivo del ensayo experimental. A continuación se muestra la placa de desarrollo seleccionada, con su respectivo diagrama de bloques:



(a) FPGA MAX DE10-Lite



(b) FPGA MAX DE10-Lite



(c) Diagrama de bloques de FPGA MAXDE10-Lite.

Figura 4.4 FPGA MAXDE10-Lite.

## 4.6. Aplicaciones

Los FPGA poseen una amplia gama de aplicaciones que abarcan diversos campos tecnológicos. Se utilizan en DSP (Procesamiento Digital de Señales), radio definido por software, sistemas aeroespaciales y de defensa, prototipos de ASIC, sistemas de imágenes médicas, sistemas de visión artificial, reconocimiento de voz, bioinformática y emulación de hardware de computadoras, entre otros. Su capacidad de paralelismo las convierte en una opción atractiva para una variedad de aplicaciones, permitiendo el desarrollo de soluciones eficientes y flexibles en diferentes áreas tecnológicas.

## 4.7. Flujo de diseño

En proyectos que involucran diseño digital, usualmente se sigue un esquema general de trabajo que abarca desde la etapa conceptual hasta la implementación física. La Figura 4.5, representa las diferentes etapas que conlleva este proceso y cómo la complejidad y el impacto en el diseño evolucionan a lo largo del mismo:



Figura 4.5 Relación entre impacto del diseño y complejidad.

Esta figura resalta la importancia de las decisiones tomadas en la etapa conceptual, ya que establecen los objetivos y la estrategia a seguir, lo que influye significativamente en el diseño final. A medida que avanza el ciclo de diseño, es posible realizar optimizaciones en cada etapa, pero esto conlleva una mayor complejidad y un menor impacto total, debido a las restricciones impuestas por las etapas anteriores. A continuación, se detallarán las etapas que típicamente atraviesa cualquier proyecto que involucre diseño digital.

### Nivel conceptual

Las decisiones iniciales sientan las bases del proyecto. En esta etapa, se definen los objetivos del proyecto, se identifican los requisitos del sistema y se exploran las

posibles soluciones para abordarlos. Se procede al desarrollo de algoritmos de control basados en estas especificaciones, y se evalúan los distintos comportamientos del sistema. Generalmente, se emplea un software como MATLAB® (MATrix LABoratory), donde se utiliza el criterio de operaciones en punto flotante para asegurar la precisión y la eficiencia del diseño.

### **Alto nivel**

Una etapa más adelante, se trabaja en la adaptación del control diseñado desde un formato de punto flotante a un formato de punto fijo como parte del proceso de prueba y verificación del sistema. Esto implica la conversión del controlador, inicialmente diseñado en punto flotante, a formatos de datos más adecuados para la implementación en hardware.

Durante esta etapa, se suelen utilizar herramientas de software específicas para la compilación del código adaptado a punto fijo, considerando las características y limitaciones del hardware. Además, se emplean simulaciones del entorno de trabajo para verificar el correcto funcionamiento del control general en condiciones similares a las reales.

### **Descripción RTL (Nivel de transferencia de registros)**

En la etapa de descripción RTL, se lleva a cabo la conversión del sistema a representación punto fijo con una mayor rigurosidad en el mapeo de hardware, ya que una elección incorrecta de la resolución puede resultar en un incremento directo en el tamaño del hardware o en el peor de los casos errores en el control implementado.

Se realiza una verificación de desempeño para evaluar la degradación causada por la cuantización de las variables. Para esto, se emplea en un simulador la estrategia de control en punto fijo diseñado específicamente para modelar el comportamiento del hardware final. Este simulador incluye referencias de relojes, registros y el modelado del comportamiento en paralelo de la arquitectura que se implementará.

La codificación se realiza en lenguaje de descripción de hardware (HDL), como Verilog o VHDL. En el diseño de nivel superior, se destaca la partición del sistema en sus diversos componentes, los cuales se definen adicionalmente en el nivel de transferencia de registro (RTL). En este nivel de abstracción, el diseñador digital especifica todos los registros y elabora cómo los datos fluirán a través de estos registros.

Para verificar el comportamiento del diseño RTL se emplean técnicas de verificación funcional conocidos como testbench, donde se estimula la unidad bajo prueba (Design Under Test - DUT) con entradas de prueba y se comparan las salidas con los resultados esperados. También, se utilizan herramienta como Modelsim para simular estos testbench que se encargan de generar las entradas y verificar las salidas del diseño.

### **Nivel de compuertas**

En esta etapa del diseño, el enfoque se centra en la traducción del diseño a un nivel de abstracción más bajo, donde se representan las conexiones a nivel de puerta (netlist). Este proceso de síntesis se inicia tomando el diseño RTL previamente definido y traduciéndolo en una descripción detallada de las interconexiones entre las celdas lógicas del dispositivo.

### **Nivel de transistores**

El nivel de transistores es la última etapa del flujo de diseño digital, donde el diseño se traduce en su representación física a nivel de transistores. Esta etapa implica la generación del layout, que es la representación del circuito integrado en términos de formas geométricas planas que corresponden a los patrones de capas de metal, óxido o semiconductor que integran los componentes del circuito.

En esta etapa, se tiene en cuenta la disposición final de los componentes en el chip, considerando aspectos como el enrutamiento de las conexiones, el diseño de las capas de metal y la optimización del área ocupada por el diseño.

En aplicaciones que utilizan FPGA, la herramienta de síntesis también se encarga de generar la disposición final de los componentes, preparando el diseño para su implementación en el dispositivo FPGA.

## **4.8. Resumen**

En este capítulo, se exploró la arquitectura de los FPGA y se presentaron los lenguajes de programación Verilog y VHDL, fundamentales para su programación. Además, se analizó la herramienta de desarrollo como Quartus Prime de Intel<sup>®</sup>, que permite sintetizar y compilar el código para los FPGA y se explicaron conceptos como la Máquina de Mealy y los IP cores. En los siguientes capítulos, se revelará cómo se utilizan estas herramientas para implementar el sistema de control.

## Capítulo 5

# Diseño de la etapa de control

### 5.1. Introducción

Como se observo en el Capitulo 3, el convertidor CC-CC se encuentra directamente influenciado por la frecuencia y el ciclo de trabajo de las señales de excitación aplicadas a sus interruptores. Sin embargo, para garantizar un rendimiento óptimo, es necesario implementar una estrategia de control que supervise y ajuste dinámicamente el ciclo de trabajo. El principal objetivo de este control es corregir de forma eficiente las desviaciones en los valores de salida ante perturbaciones en la fuente de energía o la carga, como así también, mitigar los riesgos asociados con transitorios tanto en la fase de encendido como apagado del sistema, asegurando una operación estable y segura en todas las condiciones operativas.

Es por ello que, en el presente Capitulo, se profundizará en el análisis y diseño del convertidor, abarcando desde la formulación del modelo matemático hasta la implementación de estrategias de control. En la Sección 5.2, se detallará el modelo dinámico del convertidor, su parametrización mediante el ciclo de trabajo y la posterior linealización del sistema. La Sección 5.3 se centrará en el diseño del sistema de control, dividiéndose en la planificación de los lazos internos de corriente y externo de tensión. Además, se llevará a cabo un análisis de estabilidad de polos en las distintas regiones empleadas para el control por Gain Schedule. Se comparará la efectividad de esta estrategia con la implementación basada en un controlador proporcional-integral (PI) frente a diversas perturbaciones, utilizando herramientas de software MATLAB® y Simulink®. Finalmente, en la Sección 5.5 se realizará un resumen conciso de los resultados obtenidos en las simulaciones previas.

### 5.2. Modelo del convertidor

En primer lugar, se debe encontrar un sistema de ecuaciones lineales que represente el convertidor elevador discutido en el capítulo anterior. Esto permitirá no solo

la aplicación de métodos conocidos para evaluar la estabilidad, sino también la implementación de técnicas de control convencionales. Para lograr este objetivo, se procederá a realizar un análisis sobre el funcionamiento del convertidor CC-CC bidireccional en corriente, identificando sus distintos modos de operación y en base a ello, obtener una expresión en términos del ciclo de trabajo.

### 5.2.1. Dinámica del convertidor

Retomando el esquema del convertidor CC-CC bidireccional en corriente presentado en el Capítulo 3:

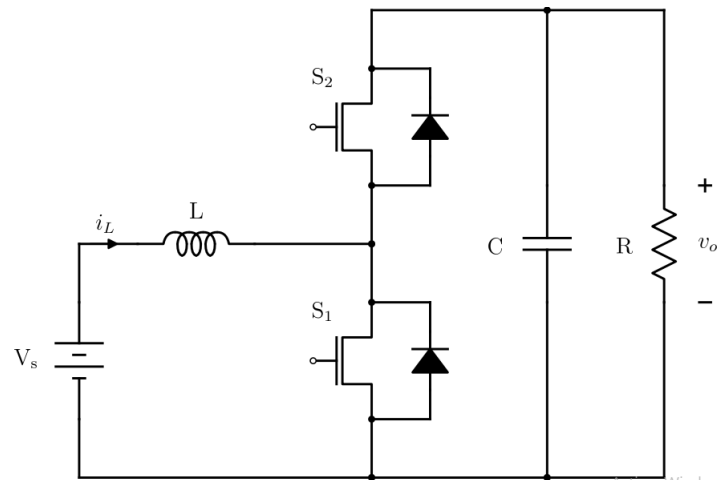


Figura 5.1 Convertidor elevador bidireccional en corriente.

Las llaves S1 y S2 trabajan en forma complementaria: cuando S1 conduce, S2 no conduce, y viceversa. Bajo esta consideración y asumiendo que el convertidor se encuentra trabajando en modo de conducción continua (MCC), se deducen dos posibles estados en los que el mismo puede encontrarse:

#### S1 conduce y S2 no conduce

$$\begin{cases} L \frac{di_L(t)}{dt} = v_s(t) \\ C \frac{dv_o(t)}{dt} = -\frac{v_o(t)}{R} \end{cases} \quad (5.1)$$

#### S1 no conduce y S2 conduce

$$\begin{cases} L \frac{di_L(t)}{dt} = v_s(t) - v_o(t) \\ C \frac{dv_o(t)}{dt} = i_L(t) - \frac{v_o(t)}{R} \end{cases} \quad (5.2)$$

### 5.2.2. Parametrizado mediante ciclo de trabajo

Con el objetivo de unificar las Ecuaciones (5.1) y (5.2) halladas para el convertidor elevador de un solo modelo, se definen los posibles estados de la acción de control:

$$d(t) = \begin{cases} 1 & S_1 \text{ conduce y } S_2 \text{ no conduce.} \\ 0 & S_1 \text{ no conduce y } S_2 \text{ conduce.} \end{cases} \quad (5.3)$$

Donde, combinando las ecuaciones (5.1) y (5.2), utilizando la variable  $d(t)$  para representar el ciclo de trabajo, se obtienen las siguientes ecuaciones resultantes:

$$\begin{cases} L \frac{di_L(t)}{dt} = v_s(t) - [1 - d(t)]v_o(t) \\ C \frac{dv_o(t)}{dt} = [1 - d(t)]i_L(t) - \frac{v_o(t)}{R} \end{cases} \quad (5.4)$$

### 5.2.3. Definición de la acción de control

El primer paso en el diseño del sistema de control implica la definición del tipo de señal que se empleará como excitación en el gate de los transistores del convertidor. En este contexto, se presentan dos alternativas posibles:

- **Sistemas de control a frecuencia variable:** Estos mantienen constante el tiempo durante la cuál el transistor se excita con un nivel alto, pero permiten que la frecuencia de conmutación varíe. Un ejemplo es la modulación de la frecuencia del pulso o PFM (del ingles *pulse frequency modulation*).
- **Sistemas de control a frecuencia fija:** Poseen un ciclo de trabajo variable, pero una frecuencia de conmutación constante. Un ejemplo es la modulación por ancho de pulso o PWM (del ingles *pulse width modulation*).

La técnica PFM se distingue por generar menos interferencia que la técnica PWM, gracias a cambios más suaves en la señal. Además, ofrece mayor precisión en la regulación de la alimentación, especialmente beneficiosa para dispositivos sensibles. Sin embargo, la variabilidad de frecuencia en esta técnica puede resultar en amplitudes significativas del ripple de conmutación, especialmente a bajas frecuencias. En este contexto, la PWM al mantener una frecuencia constante, limita el ripple simplificando su implementación en la acción de control. Además, esta última exhibe una eficiencia energética superior y una implementación más sencilla. Por estos motivos, se ha optado por utilizar la técnica PWM en este proyecto.

### 5.2.4. PWM

Como se ha mencionado anteriormente, la PWM es una técnica que modifica el ciclo de trabajo de una señal periódica (senoidal o cuadrada, por ejemplo), ya sea pa-

ra transmitir información a través de un canal de comunicaciones o para controlar la cantidad de energía que se envía a la carga. El ciclo de trabajo (*duty cycle*) de una señal periódica es el ancho de su parte positiva, en relación con el período, tal como se observa a continuación:

$$d(t) = \frac{t}{T}$$

Donde:

- d: Ciclo de trabajo.
- t: Tiempo en estado alto.
- T: Periodo, tiempo total.

Básicamente, la PWM consiste en activar una salida digital durante un intervalo específico y mantenerla inactiva el resto del tiempo, generando pulsos positivos que se repiten de manera constante (ver Figura 5.2). En este método, la frecuencia permanece constante (es decir, el tiempo entre los disparos de los pulsos), pero se ajusta la amplitud del pulso, denominada ciclo de trabajo.

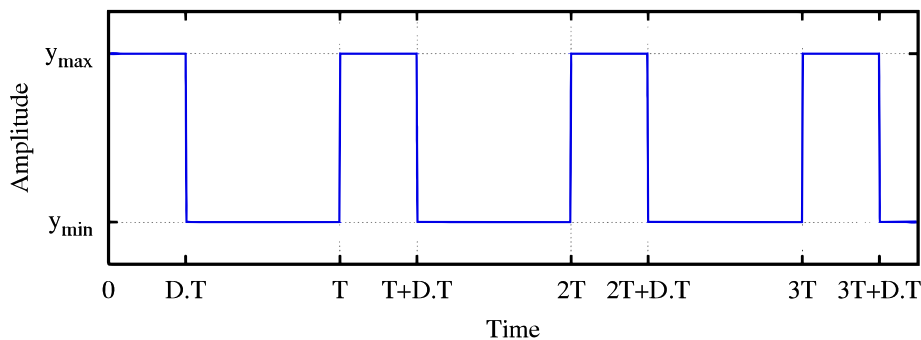


Figura 5.2 Modulación por ancho de pulso (PWM).

### 5.2.5. Promediado del sistema de ecuaciones

En los convertidores de voltaje conmutados, como el convertidor "Boost", se presenta comúnmente el fenómeno conocido como rizado de conmutación" en las señales. Este rizado puede introducir variaciones no deseadas en las señales, lo que puede complicar el análisis y diseño del sistema. Para abordar esta complejidad y simplificar las ecuaciones que describen su comportamiento, se recurre a técnicas de modelado que permiten capturar la esencia dominante de la señal.

El modelo representado por las Ecuaciones (5.4) describe un sistema de estructura variable no lineal. En este contexto, el convertidor "Boost" se comporta como diferentes

sistemas no lineales continuos en diversas regiones de su espacio de estados, influenciado por una acción de control discontinua. Esta complejidad en el análisis y control del sistema puede ser mitigada mediante técnicas de modelado.

Una técnica habitualmente utilizada es el empleo de *medias móviles*, donde cada señal se sustituye por su promedio durante un período de conmutación  $T_s$  fijo. Este enfoque simplifica las ecuaciones del sistema, al obtener una representación más suave y continua del comportamiento del sistema. Las *medias móviles* tienen la ventaja adicional de capturar la parte dominante de las señales, eliminando las perturbaciones como el ripple de conmutación, que no son relevantes para el diseño del sistema de control del convertidor.

Utilizando el método de promediado presentado [16] y [14], para simplificar las ecuaciones que rigen la dinámica de los sistemas, se procede de la siguiente manera:

$$\bar{x}(t) = \frac{1}{T_s} \int_t^{t+T_s} x(\tau) d\tau \quad (5.5)$$

Encontrando los valores promedio de  $V_o(t)$  y  $I_L(t)$ :

$$\bar{V}_o(t) = \frac{1}{T_s} \int_t^{t+T_s} V_o(\tau) d\tau \quad (5.6)$$

$$\bar{i}_L(t) = \frac{1}{T_s} \int_t^{t+T_s} i_L(\tau) d\tau \quad (5.7)$$

Puede demostrarse que aplicando las Ecuaciones (5.6) y (5.7) a las Ecuaciones (5.4), se obtiene:

$$\begin{cases} L \frac{d\bar{i}_L(t)}{dt} = \bar{v}_s(t) - [1 - \bar{d}(t)]\bar{v}_o(t) \\ C \frac{d\bar{v}_o(t)}{dt} = [1 - \bar{d}(t)]\bar{i}_L(t) - \frac{\bar{v}_o(t)}{R} \end{cases} \quad (5.8)$$

Es importante señalar que es viable obtener un modelo promediado del convertidor operando en modo de conducción discontinua (MCD) [17]. No obstante, dicho desarrollo no se presenta en este informe, ya que escapa de los objetivos establecidos para este trabajo.

### 5.2.6. Linealización del sistema de ecuaciones

Si observamos las Ecuaciones (5.8), el modelo obtenido no es lineal ya que aparecen productos entre los estados del sistema y la señal de control promediada  $\bar{d}(t)$ . Por consecuencia, es necesario linealizar el sistema para aplicar las técnicas tradicionales que conocemos.

Con el objetivo de lograr una adecuada linealización del modelo en pequeña señal, se determinan los valores promedio de las variables de interés que serán nombrados de la siguiente manera:

- $I_{L_{ee}}$  : Corriente en la inductancia en estado estacionario.
- $V_{s_{ee}}$  : Tensión de entrada en estado estacionario.
- $V_{o_{ee}}$  : Tensión de salida en estado estacionario.
- $D_{ee}$  : Ciclo de trabajo en estado estacionario.

A partir de estos valores, se construye un *modelo de pequeña señal* alrededor del punto de trabajo. Para ello, se asume que cada variable de interés promediada es igual a un valor de estado estacionario más una perturbación, como se puede observar a continuación:

$$\begin{cases} \bar{d}(t) = D_{ee} + \hat{d}(t) \\ \bar{v}_o(t) = V_{o_{ee}} + \hat{v}_o(t) \\ \bar{i}_L(t) = I_{L_{ee}} + \hat{i}_L(t) \end{cases} \quad (5.9)$$

Considerando que las perturbaciones  $\hat{d}(t)$ ,  $\hat{v}_o(t)$  y  $\hat{i}_L(t)$  son considerablemente menores que los valores de estado estacionario, se deduce que los valores promedios  $\bar{d}(t)$ ,  $\bar{v}_o(t)$  y  $\bar{i}_L(t)$  permanecerán cercanos a los valores del punto de operación. De este modo, sustituyendo los valores promedios definidos por la Ecuación (5.9) en el sistema dado por la Ecuación (5.8), se obtiene:

$$\begin{cases} L \frac{d[I_{L_{ee}} + \hat{i}_L(t)]}{dt} = V_s - [1 - D_{ee} - \hat{d}(t)][V_{o_{ee}} + \hat{v}_o(t)] \\ C \frac{d[V_{o_{ee}} + \hat{v}_o(t)]}{dt} = [1 - D_{ee} - \hat{d}(t)][I_{L_{ee}} + \hat{i}_L(t)] - \left[ \frac{V_{o_{ee}}}{R} + \frac{\hat{v}_o(t)}{R} \right] \end{cases} \quad (5.10)$$

Operando:

$$\begin{cases} L \frac{d\hat{i}_L(t)}{dt} = V_s - (1 - D_{ee})V_{o_{ee}} - (1 - D_{ee})\hat{v}_o + \hat{d}(t)V_{o_{ee}} + \hat{d}(t)\hat{v}_o(t) \\ C \frac{d\hat{v}_o(t)}{dt} = (1 - D_{ee})I_{L_{ee}} + (1 - D_{ee})\hat{i}_L(t) - \hat{d}(t)I_{L_{ee}} - \hat{d}(t)\hat{i}_L(t) - \frac{V_{o_{ee}}}{R} - \frac{\hat{v}_o(t)}{R} \end{cases} \quad (5.11)$$

Y teniendo en cuenta las siguientes consideraciones:

- $V_s = (1 - D_{ee})V_{o_{ee}}$
- $\frac{V_{o_{ee}}}{R} = I_{o_{ee}} = (1 - D_{ee})I_{L_{ee}}$
- Se descartan los términos no lineales de segundo orden, ya que se supone que las perturbaciones son muy pequeñas, y por ende, su producto aún más.

Finalmente, se deduce el modelo lineal del convertidor:

$$\begin{cases} L \frac{d\hat{i}_L(t)}{dt} = -(1 - D_{ee})\hat{v}_o + \hat{d}(t)V_{o_{ee}} \\ C \frac{d\hat{v}_o(t)}{dt} = (1 - D_{ee})\hat{i}_L(t) - \hat{d}(t)I_{L_{ee}} - \frac{\hat{v}_o(t)}{R} \end{cases} \quad (5.12)$$

### 5.3. Diseño del sistema de control

Una vez obtenidas las ecuaciones del modelo lineal del convertidor, el siguiente paso implica el diseño de la estrategia de control, la cuál estará influenciada por las especificaciones disponibles en cuanto a los modos de operación y las características de la respuesta dinámica requerida. En este proyecto en particular, teniendo en cuenta que la función principal de los convertidores CC-CC es mantener constante el valor medio de la tensión de salida frente a variaciones en la carga y en la tensión de alimentación, la acción de control será la encargada de asegurar una tensión constante en la carga. Sin embargo, al considerar un lazo de realimentación, como se ilustra en la Figura 5.3, se puede observar rápidamente que esta estrategia no es viable para un convertidor elevador.

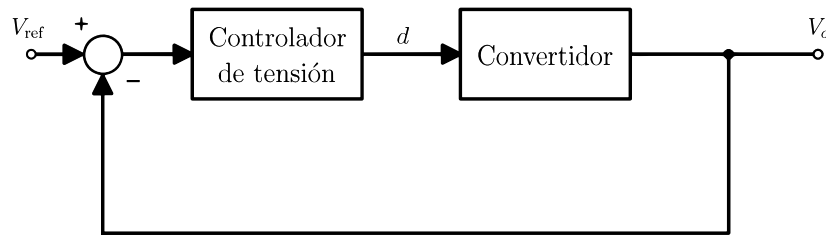


Figura 5.3 Ilustración del lazo de tensión propuesto.

Esto se debe a que al implementar únicamente un lazo cerrado de tensión en un convertidor CC-CC, surge lo que se conoce como *dinámica escondida* o *dinámica cero*, la cuál se manifiesta como una inestabilidad en la respuesta del sistema. En otras palabras, la corriente de salida puede experimentar oscilaciones indeseadas o incluso divergir, resultando en un comportamiento inestable del convertidor.

Este fenómeno se puede observar asumiendo inicialmente que la tensión de salida alcanza un valor estacionario  $V_{o_{ee}}$  y considerando una tensión de entrada constante  $V_s$ . Por lo tanto, teniendo en cuenta estas consideraciones en la Ecuación (5.8):

$$\begin{cases} L \frac{d\bar{i}_L(t)}{dt} = V_s - [1 - \bar{d}(t)]V_{o_{ee}} \\ 0 = [1 - \bar{d}(t)]\bar{i}_L(t) - \frac{V_{o_{ee}}}{R} \end{cases} \quad (5.13)$$

Reemplazando el ciclo de trabajo promediado, se obtiene:

$$L \frac{d\bar{i}_L(t)}{dt} = V_s - \frac{V_{o_{ee}}^2}{R\bar{i}_L(t)} \quad (5.14)$$

Si se expresa en términos de potencia:

$$L \frac{d\bar{i}_L(t)}{dt} = \frac{P_{s_{ee}}}{\bar{i}_L(t)} - \frac{P_{o_{ee}}}{\bar{i}_L(t)} \quad \forall t \quad (5.15)$$

Considerando que en estos convertidores la potencia de salida es inferior a la potencia de entrada debido a las pérdidas existentes en el mismo, resulta evidente que la derivada de la corriente será siempre positiva, y por consecuencia, la corriente a través del inductor diverge. Para abordar eficazmente este problema, se introduce un lazo interno de corriente. Este lazo interno, se encargará de regular la corriente a través del inductor, garantizando una respuesta rápida y estable ante cambios en la carga. Al combinar el lazo interno de corriente con el lazo externo de tensión, se logra un control que permite tanto la estabilidad como la precisión del convertidor CC-CC.

De esta manera, la estrategia de control estará compuesta por un **lazo interno de corriente** anidado dentro del **lazo externo de tensión** como se observa en la figura 5.4. En este esquema, el lazo de tensión será el encargado de calcular y suministrar la referencia al lazo de corriente, para que este último genere el ciclo de trabajo requerido para mantener la tensión en la carga constante.

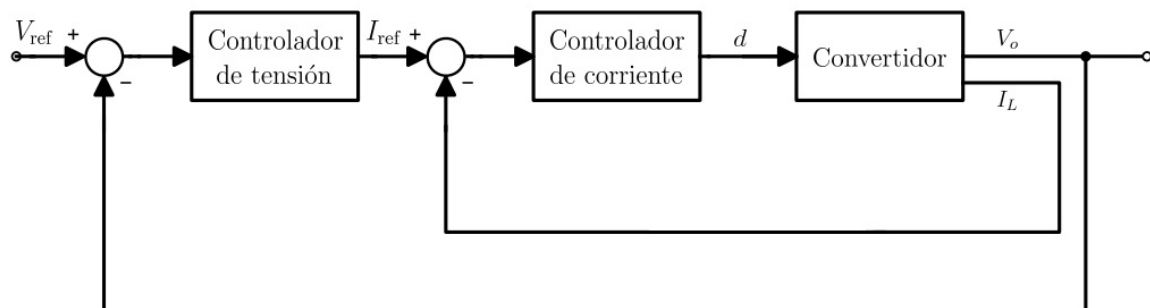


Figura 5.4 Ilustración del lazo de control propuesto.

### 5.3.1. Diseño del lazo interno de corriente

Para el diseño del lazo interno de corriente, se propone la siguiente configuración de control:

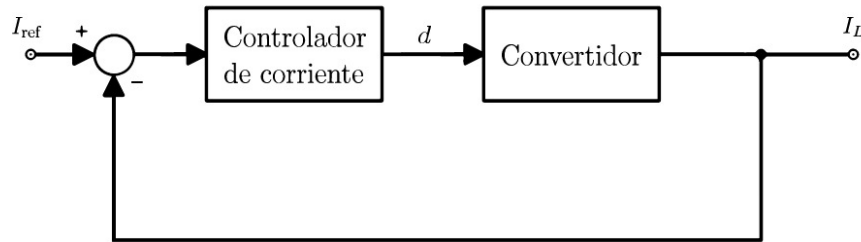


Figura 5.5 Ilustración del lazo de corriente propuesto.

En este diseño, se compara la corriente en el inductor con la referencia deseada proporcionada por el lazo externo de tensión. La señal de error resultante ingresa al controlador de corriente, que determinará la acción de control necesaria para el convertidor, como se muestra en la Figura 5.5.

En base al esquema planteado, es necesario determinar la función de transferencia del convertidor. Para lograrlo, se emplea el modelo linealizado del convertidor elevador, expresado en la Ecuación (5.12). Aplicando la transformada de Laplace sobre el modelo mencionado y asumiendo que las condiciones iniciales son nulas, se obtiene:

$$\begin{cases} sLI_L(s) = -(1 - D_{ee})V_o(s) + D(s)V_{o_{ee}} \\ sCV_o(s) = (1 - D_{ee})I_L(s) - D(s)I_{L_{ee}} - \frac{V_o(s)}{R} \end{cases} \quad (5.16)$$

Por lo tanto, si de ambos términos se despeja  $V_o(s)$  y se igualan, se obtiene la función de transferencia buscada:

$$G_I(s) = \frac{I_L(s)}{D(s)} = \frac{\frac{V_{o_{ee}}}{L}s + \left[ \frac{V_{o_{ee}}}{LCR} + \frac{I_{L_{ee}}(1-D_{ee})}{LC} \right]}{s^2 + \frac{1}{RC}s + \frac{(1-D_{ee})^2}{LC}} \quad (5.17)$$

Ahora, deben determinarse los valores de tensión de entrada  $V_s$ , ciclo de trabajo  $D_{ee}$ , resistencia de carga  $R$ , inductancia  $L$  y capacitancia  $C$ . Para ello, se considera una tensión ligeramente inferior a la nominal del banco de supercapacitores y baterías de litio. De esta manera, conocidas ambas tensiones, es posible calcular el ciclo de trabajo:

$$D_{ee} = 1 - \frac{V_s}{V_{o_{ee}}} \quad (5.18)$$

En cuanto a la inductancia y la capacitancia se obtuvieron directamente del diseño del convertidor elevador mencionado en los capítulos anteriores. Dado que el rango de corriente del convertidor varía de 0A a una corriente máxima de aproximadamente 15A, se determinó que un punto de operación adecuado sería en la mitad de este rango, es decir para corrientes cercanas a 7,5A. Por lo tanto, conociendo este valor es

posible calcular la resistencia de carga en base a la siguiente Ecuación:

$$R = \frac{V_{o_{ee}}}{(1 - D_{ee})I_{L_{ee}}} \quad (5.19)$$

De esta manera, se obtiene la siguiente tabla de asignación de valores:

$V_s$	24V
$V_{o_{ee}}$	48V
$D_{ee}$	0,5
$C$	4400 $\mu$ F
$L$	200 $\mu$ H
$R$	12,8 $\Omega$

Tabla 5.1 Punto de trabajo alrededor del cuál se linealiza el convertidor

En base a los valores planteados en la tabla 5.1 y reemplazando en la Ecuación (5.17), la función de transferencia queda determinada de la siguiente manera:

$$G_I(s) = \frac{I_L(s)}{D(s)} = \frac{240000(s + 35,51)}{s^2 + 17,76s + 284100} \quad (5.20)$$

La cuál posee un cero en  $s = -35,51$  y dos polos complejos conjugados que se encuentran en  $s_{1,2} = -8,88 \pm j532,936$  como se ilustra en el lugar de raíces de a continuación:

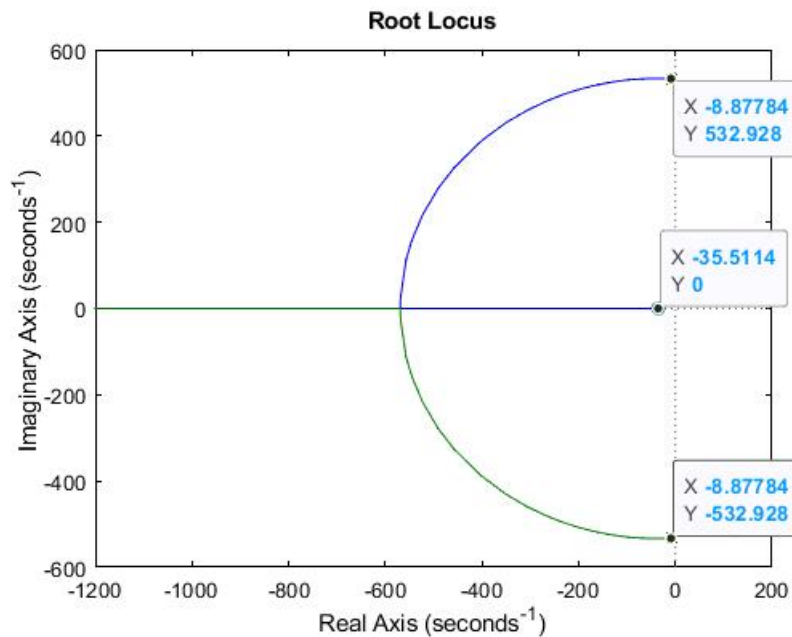


Figura 5.6 Lugar de raíces de la transferencia encontrada.

Como se discutió en el Capítulo 4, el hardware de control esta constituido por un FPGA que opera mediante muestras adquiridas a través de un conversor. Por esta razón, el diseño del controlador se lleva a cabo en el dominio discreto. Para la discretización de la planta se utilizó el método de *Euler en avance*:

$$s = \frac{z - 1}{T_s} \quad (5.21)$$

En este contexto,  $T_s$  representa el tiempo de muestreo de los conversores analógicos-digitales, el cuál debe ser significativamente menor que la menor constante de tiempo del sistema. Con el objetivo de cumplir este requisito, se configura el controlador de los ADCs para que posea un periodo de muestreo de  $5\mu s$ , garantizando así que esté notablemente por debajo del período de conmutación de las llaves, que es de  $50\mu s$ . De esta forma, al sustituir el tiempo de muestreo mencionado anteriormente, se expresa numéricamente la función de transferencia:

$$G_I(z) = \frac{I_L(z)}{D(z)} = \frac{1,20005z - 1,19983}{z^2 - 1,99990z + 0,99991} \quad (5.22)$$

Para el lazo cerrado de control, es necesario implementar un filtro de corriente ya que la acción de control se calculará a partir de la corriente media por el inductor y no su valor instantáneo. En este sentido, se utiliza un filtro pasa-bajos que elimine las componentes del *ripple* y, al mismo tiempo, sea lo suficientemente rápido para poder seguir la dinámica de la planta modelada. A continuación, se realizará un análisis del filtro pasa bajos de primer orden implementado, comparándolo con uno de segundo orden, con énfasis en aspectos como la efectividad del filtrado y el desfase que genera.

### Filtro de corriente

Dado que el *ripple* se genera debido a la conmutación de las llaves con una frecuencia fundamental de  $20kHz$  y que los polos de la planta poseen un valor absoluto de 535, se decidió implementar un filtro pasa bajos de primer orden con una frecuencia de corte en  $1,5kHz$ , de tal forma que asegure atenuar eficazmente el rizado que se encuentra a frecuencias mayores. Por lo tanto, si consideramos un filtro pasa-bajos de primer orden, su función de transferencia se expresa de la siguiente manera:

$$F_{I_1^{\circ}orden}(s) = \frac{W_c}{s + W_c} = \frac{2\pi 1,5 \times 10^3}{s + 2\pi 1,5 \times 10^3} \quad (5.23)$$

Transformándola al dominio discreto mediante el método de *Euler en avance*:

$$F_{I_1^{\circ}orden}(z) = \frac{W_c T_s}{z + (W_c T_s - 1)} = \frac{0,04712}{z - 0,9528} \quad (5.24)$$

Con el propósito de determinar si el filtro actual es adecuado para atenuar el ripple generado por la conmutación de la llave, o si se requiere uno de segundo orden, se llevó a cabo un análisis comparativo utilizando el software MATLAB®. Donde se graficaron las respuestas del filtro pasa bajos de primer orden y un filtro Butterworth de segundo orden, ambos con la misma frecuencia de corte, en presencia de dos sinusoides: la primera con una frecuencia de  $550\text{Hz}$ , representa la frecuencia del rango de interés para el diseño del controlador utilizando el modelo promediado, mientras que la segunda, con una frecuencia de  $20\text{kHz}$  y una amplitud de  $3/16$  respecto a la primera, simula la frecuencia de la PWM que se quiere filtrar. En la Figura 5.7 se muestra la DEP (Densidad Espectral de Potencia) de la señal sin filtrar, la señal filtrada con el filtro de primer orden, y la señal filtrada con el filtro de segundo orden, respectivamente:

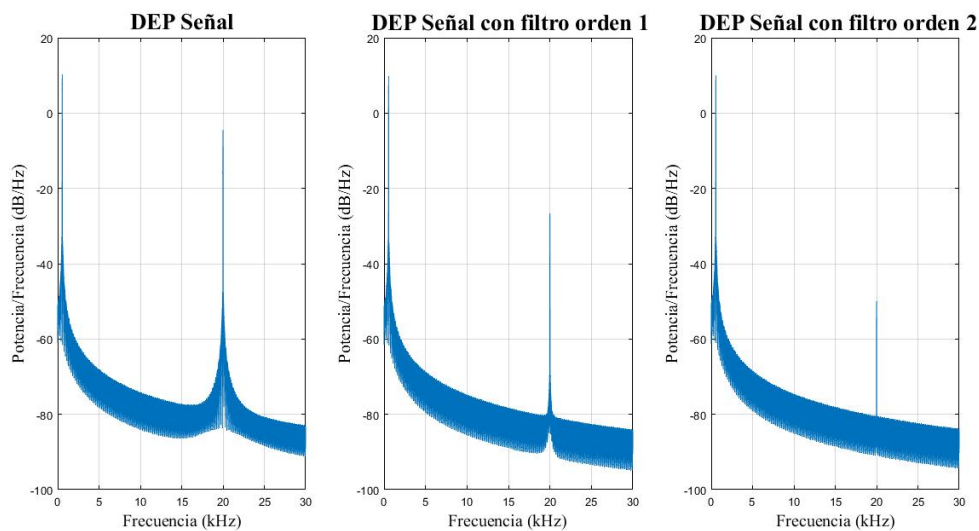


Figura 5.7 DEP de la señal filtrada.

De las imágenes anteriores se puede apreciar la efectividad del filtro de segundo orden en la supresión de la amplitud de la PWM. Este mejor desempeño es coherente con lo esperado, ya que los filtros de segundo orden suelen ofrecer una mayor capacidad para atenuar frecuencias no deseadas pero también presentan un mayor desfase respecto de la señal. No obstante, es importante señalar que el filtro de primer orden demuestra una notable capacidad para filtrar la mayoría de la amplitud de la PWM, y presenta un menor desfase en comparación con el filtrado de segundo orden. Esta diferencia se aprecia de mejor manera en el siguiente gráfico en función del tiempo:

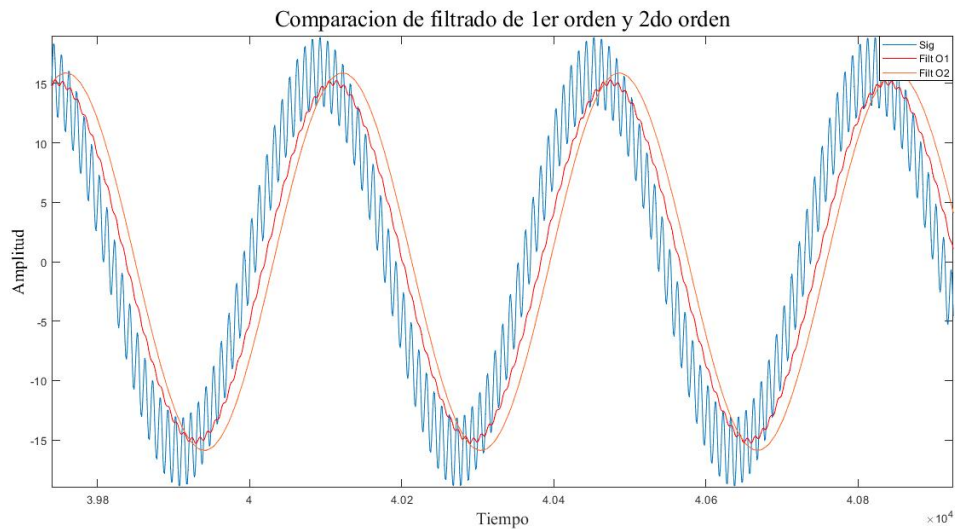


Figura 5.8 Comparación ambos filtrados.

En el contexto de este proyecto, se optó por utilizar un filtro de primer orden, fundamentado en su efectiva capacidad para suprimir el ruido generado por la PWM, así como en su mayor facilidad de implementación. De este modo, el lazo interno de corriente queda conformado de la siguiente manera:

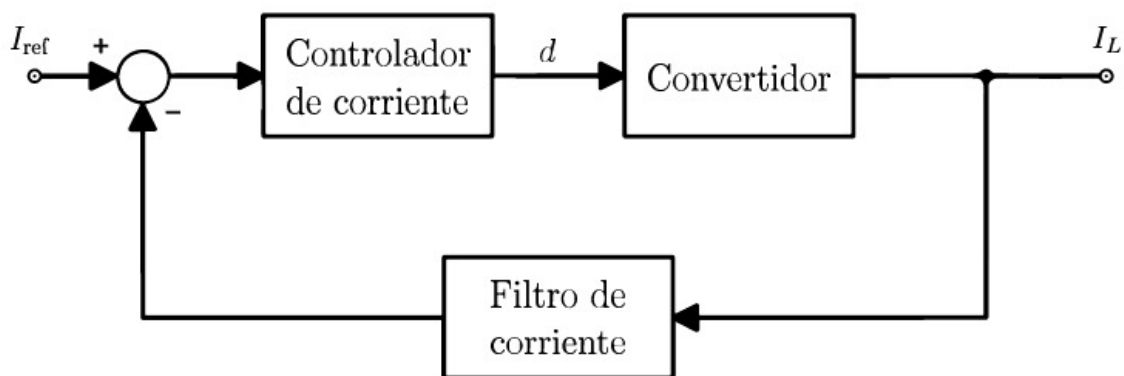


Figura 5.9 Lazo de control de corriente con filtro pasa bajos.

Seleccionado el filtro a utilizar, el siguiente paso implica establecer las especificaciones respecto a la respuesta dinámica de la corriente del inductor, con el propósito de seleccionar el controlador adecuado. En este sentido, las especificaciones a considerar son las siguientes:

- Error de estado estacionario nulo.

- Tiempo de establecimiento del orden del milisegundo o menor.
- La amplitud de los sobrepicos no ponga en riesgo a los componentes del sistema, ni a la carga.

En este proyecto se optó por utilizar un controlador de la familia de los *controladores PID* (*proporcional-integral-derivativo*), decisión fundamentada en razones que se explicarán más adelante.

### Controladores PID

Las estrategias de control *PID* se introdujeron en el ámbito industrial en el primer cuarto del siglo pasado mediante ajustes completamente empíricos. Aunque la teoría de control ha avanzado en este último tiempo, el controlador *PID* sigue siendo utilizado a nivel mundial en el entorno industrial.

La integración de los *PID* en la industria ha alcanzado un nivel tal que revela un fenómeno aparentemente contradictorio. Por un lado, los usuarios se resisten a reemplazar el controlador *PID* por controladores con estructura diferente, incluso si son más simples y robustos. Por otro lado, están dispuestos a adoptar controladores adaptables y sofisticados siempre que se presente bajo la etiqueta de *PID*. Es por ello que, en el marco de este proyecto, se buscará aplicar nuevas técnicas de control utilizando estos controladores.

Los miembros de la familia de controladores *PID*, incluyen tres acciones: *proporcional* (P), *integral* (I), *derivativo* (D). Estos controladores son los denominados P, I, PI, PD y PID. A continuación, se muestra un esquema típico de un lazo cerrado de control que emplea este tipo de controlador:

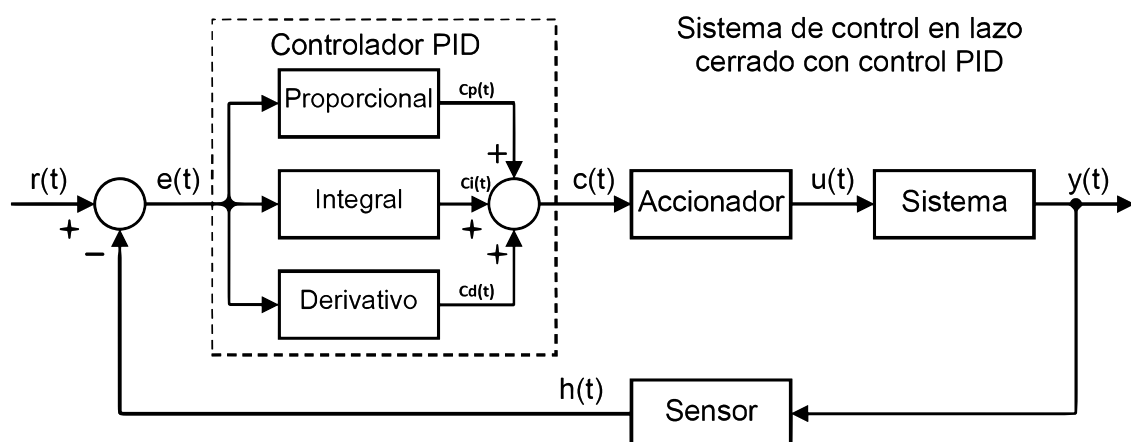


Figura 5.10 Sistema de lazo cerrado con control PID.

- **P: acción proporcional**, da una salida al controlador que es proporcional al error.

$$C_p(t) = K_p e(t) = K_p (r(t) - y(t)) \quad (5.25)$$

En la ecuación anterior,  $K_p$  representa la ganancia proporcional ajustable, la cuál determina la cantidad de acción proporcional que tendrá el controlador. Esta acción de control intenta minimizar el error del sistema, cuando el error es grande, la acción de control es grande y tiende a minimizar este error. Aumentar la acción proporcional  $K_p$  tiene los siguientes efectos.

- Aumenta la velocidad de respuesta del sistema.
- Disminuye el error del sistema en estado estacionario.
- Disminuye la estabilidad del sistema.

Los dos primeros efectos son positivos y deseables. El último efecto es negativo y se debe intentar reducir. Pese a este último efecto indeseado, la principal desventaja de utilizar un controlador proporcional puro es que produce un error de estado estacionario.

- **I: acción de control integral**, da una salida del controlador que es proporcional al error acumulado, lo que implica que es un modo de controlar lento.

$$C_i(t) = K_i \int_0^t e(\tau) d\tau \quad (5.26)$$

Y su función de transferencia es:

$$C_i(s) = \frac{K_i}{s} \quad (5.27)$$

Esta acción de control calcula la integral de la señal de error  $e(t)$  y la multiplica por la constante  $K_i$ , donde dicha integral representa la acumulación de pequeños errores a lo largo del tiempo. La presencia del polo en el origen permite la reducción a cero del error de estado estacionario ante señales escalón en la referencia o perturbaciones en la carga. Sin embargo, esta acción integral introduce una cierta inercia al sistema, lo que puede aumentar su inestabilidad. Aumentar la acción integral  $K_i$  tiene los siguientes efectos:

- Elimina el error en estado estacionario ante señales escalón en la referencia.
- Aumenta la inestabilidad del sistema.

Cabe agregar que, cuando se implementa un controlador con acción integral, puede surgir el llamado efecto *windup* en los casos en que el actuador satura,

dato que el integrador continua operando. Es así que, en determinadas circunstancias, tales como un cambio abrupto en la referencia, es posible que el valor acumulado por el integrador durante la saturación sea elevado, requiriendo de un considerable tiempo para descargarse. Esto da origen a sobrepasos indeseados en la salida. Para mitigar este problema, se recurre a métodos *anti-windup*.

- **D: acción de control derivativa**, permite "predecir" valores futuros de la señal del error.

$$C_d(t) = K_d \frac{de(t)}{dt} \quad (5.28)$$

Y su función de transferencia es:

$$C_d(s) = K_d s \quad (5.29)$$

Esta acción de control es proporcional a la derivada de la señal del error  $e(t)$  multiplicada por la constante  $K_d$ . Esta acción de control tiene un gran potencial para mejorar el rendimiento del sistema, ya que puede prever tendencias incorrectas en la señal de error y contrarrestarlas, lo que contribuye a una acción de control más rápida. No obstante, presenta la desventaja significativa de amplificar las señales de ruido, lo que podría provocar saturación en el actuador.

Específicamente en el marco de este proyecto, considerando la conmutación de las llaves y la generación de rizado en las variables de estado controladas, la inclusión del término derivativo en el control amplificaría este rizado de manera no deseada, lo que podría resultar en oscilaciones indeseables en el sistema de control. Por esta razón, se decidió implementar un controlador *proporcional-integral (PI)*.

### Controladores PI

Es la configuración más comúnmente utilizada para un controlador, y su función de transferencia puede expresarse como:

$$PI_I(s) = K_p + \frac{K_i}{s} \quad (5.30)$$

Al discretizarla, se obtiene:

$$PI_I(z) = K_p + \frac{K_i T_s}{z - 1} \quad (5.31)$$

Donde  $K_p$  y  $K_i$  son ajustados mediante la sintonización del controlador *PI* con el fin de lograr la respuesta deseada en la configuración a lazo cerrado.

### Simulación del lazo interno de corriente

Se procede a ajustar los parámetros del PI implementado para el lazo de corriente. Esto se realiza utilizando la transferencia lineal obtenida a partir de la ecuación (5.22). Mediante la herramienta Simulink®, se configura el lazo de corriente, siguiendo el esquema presentado en la figura 5.9:

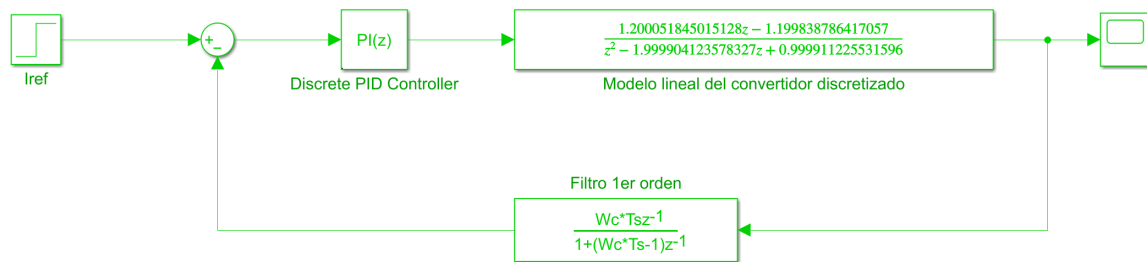
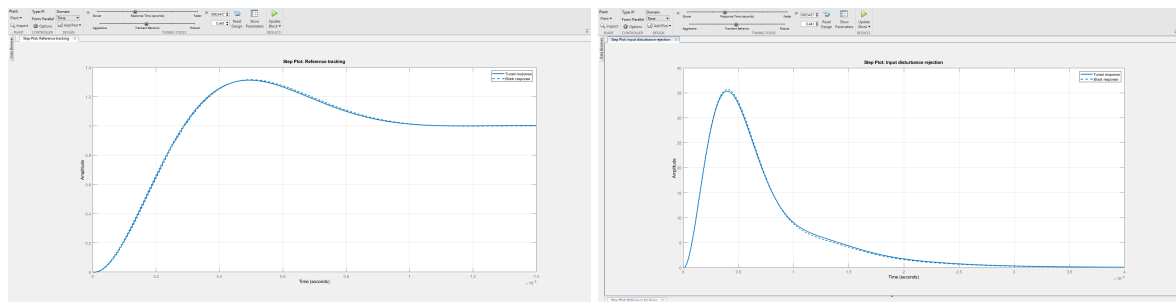


Figura 5.11 Lazo de control de corriente con filtro pasa bajos.

Por otra parte, MATLAB® ofrece una herramienta llamada *PID Tuner* que posibilita la mejora interactiva del rendimiento del controlador. Esto permite ajustar el ancho de banda del lazo y el margen de fase, ya sea para favorecer el seguimiento de la referencia o para el rechazo a perturbaciones. En consecuencia, a través de esta herramienta, se ajustaron los parámetros del controlador PI para lograr una respuesta satisfactoria y un buen rechazo a perturbaciones, como se observa a continuación.



(a) Seguimiento a la referencia.

(b) Rechazo a perturbaciones.

Figura 5.12 Sintonización mediante *PID tuner* del lazo de corriente interno.

Luego de evaluar el desempeño del controlador PI en la planta lineal, se llevan a cabo pruebas de simulación utilizando el modelo real del convertidor. Donde efectuando nuevos ajustes, se logró la siguiente configuración del controlador:

$$PI_I(z) = 0,027 + \frac{37 \times T_s}{z - 1} \quad (5.32)$$

En la cuál, se deduce que la constante proporcional  $K_p$  es 0,027 y la constante integral  $K_i$  es 37. En base al esquema de la Figura 5.13 se obtuvieron los siguientes resultados, en los cuales se observa que la corriente por el inductor filtrada (Figura 5.14c) atenúa satisfactoriamente al rizado de la forma de onda, obteniendo un valor de sobrepico de 4,2 % respecto del valor de estado estacionario de 8,5A, y un tiempo de establecimiento con criterio del 2 % de aproximadamente 0,8ms:

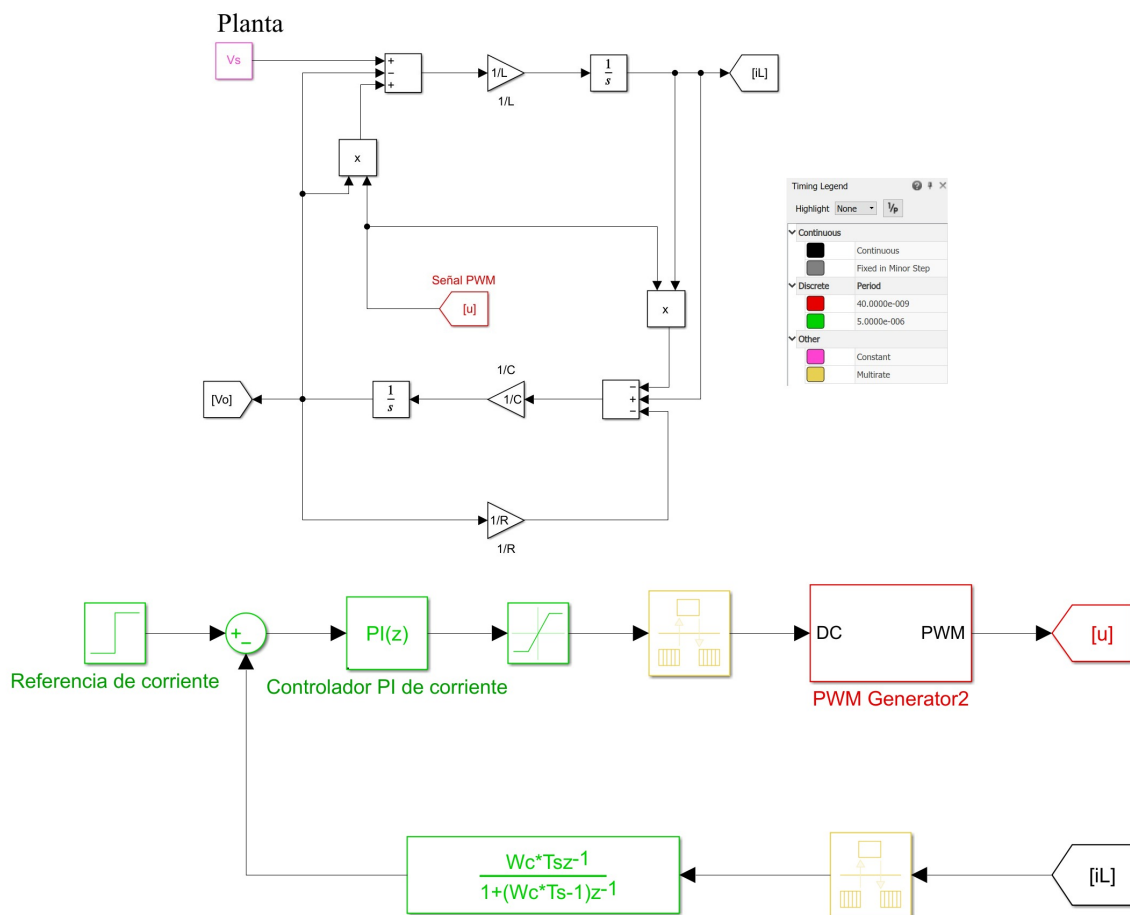


Figura 5.13 Simulación real del lazo de corriente

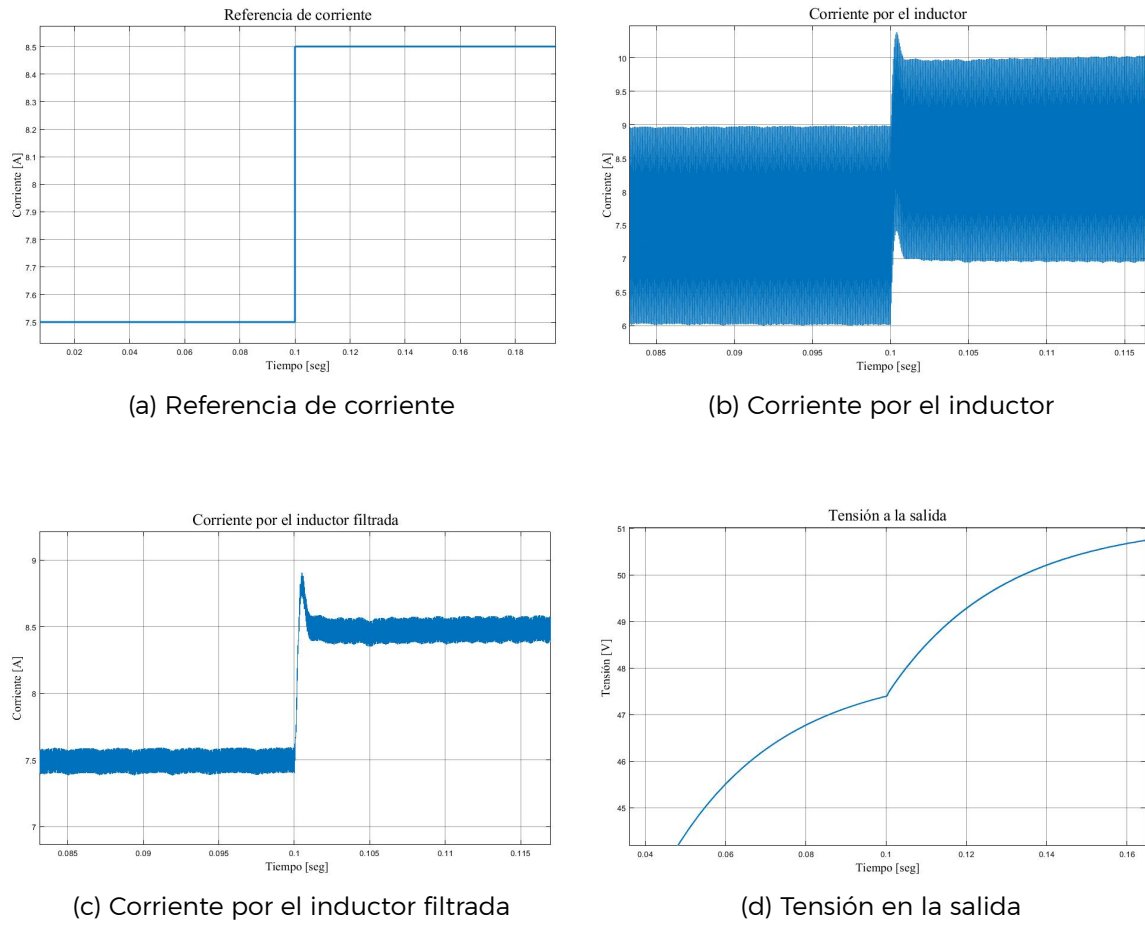


Figura 5.14 Formas de onda obtenidas en la simulación del lazo de control de corriente.

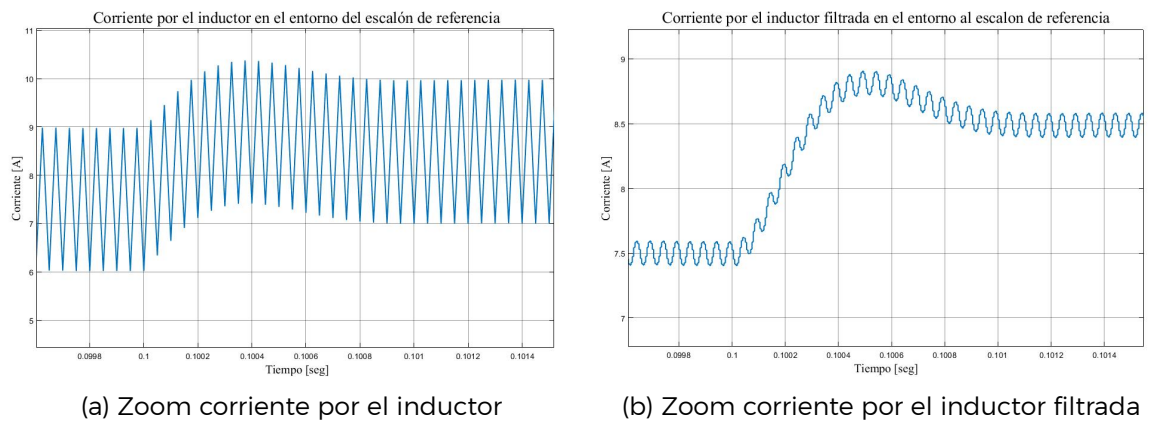


Figura 5.15 Corriente del inductor en el entorno del escalón unitario de corriente.

### 5.3.2. Diseño del lazo de tensión externo

Para lograr controlar la tensión de salida del convertidor, se implementa la siguiente estrategia de control:

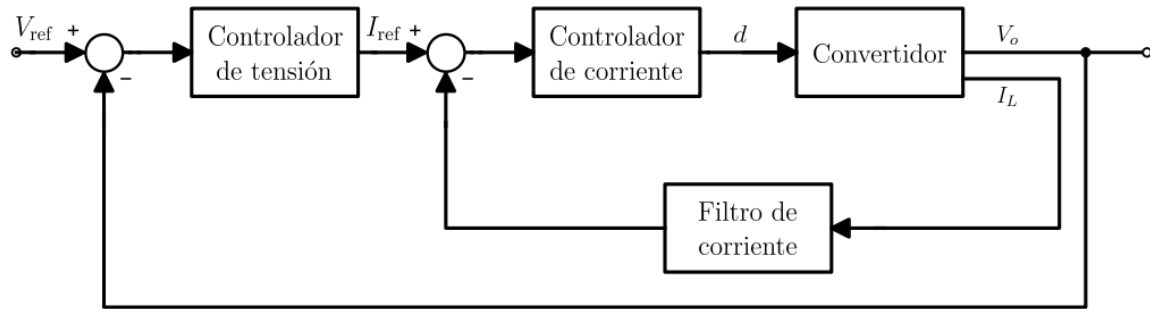


Figura 5.16 Lazo de control de tensión

En este diseño, se observa que el lazo de corriente está anidado dentro del control de tensión. Es por ello que surge la necesidad de expandir el modelo del sistema para incluir el lazo de control de corriente con su respectivo filtro. Es importante destacar que, para el control del lazo de tensión externo, no es necesario transformarlo al dominio discreto. Esto se debe a que el tiempo de muestreo utilizado es considerablemente menor que la dinámica dominante del lazo mencionado. En consecuencia, desde un punto de vista teórico, la respuesta en continuo de los controladores será muy similar a la respuesta en el dominio discreto. Por lo tanto, se opta por sintonizar los controladores en el dominio continuo, lo cual simplifica significativamente el diseño y la implementación del sistema de control. Por lo tanto, al desarrollar los modelos de estado para el controlador PI de corriente y su filtro, se obtienen las siguientes ecuaciones:

$$\begin{cases} \dot{x}_1(t) = I_{ref} - x_2(t) \\ \dot{x}_2(t) = W_c I_L - W_c x_2(t) \end{cases} \quad (5.33)$$

Considerando que:

$$d(t) = K_{p_I} \dot{x}_1(t) + K_{i_I} x_1(t) \quad (5.34)$$

Reemplazando esta última expresión en las Ecuaciones del modelo lineal del convertidor (5.12):

$$\begin{cases} \dot{i}_L(t) = -(1 - D_{ee}) \frac{v_o(t)}{L} + \frac{V_{oee}}{L} K_{p_I} I_{ref} - \frac{V_{oee}}{L} K_{p_I} x_2(t) + \frac{V_{oee}}{L} K_{i_I} x_1(t) \\ \dot{v}_o(t) = (1 - D_{ee}) \frac{i_L(t)}{C} - \frac{v_o(t)}{CR} - \frac{I_{Lee}}{C} K_{p_I} I_{ref} + \frac{I_{Lee}}{C} K_{p_I} x_2(t) - \frac{I_{Lee}}{C} K_{i_I} x_1(t) \end{cases} \quad (5.35)$$

Expandiendo el sistema con las Ecuaciones (5.33), se obtiene:

$$\begin{bmatrix} \dot{i}_L(t) \\ \dot{v}_o(t) \\ \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{L}(1 - D_{ee}) & \frac{V_{oee}}{L} K_{i_I} & -\frac{V_{oee}}{L} K_{p_I} \\ \frac{1}{C}(1 - D_{ee}) & -\frac{1}{CR} & -\frac{I_{Lee}}{C} K_{i_I} & \frac{I_{Lee}}{C} K_{p_I} \\ 0 & 0 & 0 & -1 \\ W_c & 0 & 0 & -W_c \end{bmatrix} \begin{bmatrix} i_L(t) \\ v_o(t) \\ x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} \frac{V_{oee}}{L} K_{p_I} \\ -\frac{I_{Lee}}{C} K_{p_I} \\ 1 \\ 0 \end{bmatrix} I_{ref}$$

Este sistema de ecuaciones representa la planta con el lazo cerrado de corriente. Como se mencionó anteriormente, es importante que la dinámica del lazo de tensión sea de al menos un orden de magnitud más lenta que la del lazo de corriente. Esto garantiza un seguimiento preciso de la referencia sin inducir oscilaciones no deseadas. Para abordar el control de tensión externo, se implementó una estrategia *Gain Schedule*, la cual será explicada en la siguiente sección.

### 5.3.3. Planificación de ganancias (*Gain Schedule*)

Las técnicas no lineales desempeñan un papel crucial cuando se trata de sistemas dinámicos complejos. A diferencia de los métodos lineales convencionales, los enfoques no lineales tienen la capacidad de modelar y controlar sistemas más complejos y variables. Esta flexibilidad es esencial cuando los sistemas presentan comportamientos no lineales inherentes o cuando se requiere una precisión más alta.

Una estrategia importante dentro de las técnicas de control no lineal es el *Gain Schedule* (GS) o "Plan de Ganancias". Este enfoque utiliza una o más variables observables, conocidas como variables de planificación, para identificar la región operativa actual del sistema. Estas variables proporcionan información sobre las condiciones del sistema y permiten seleccionar el controlador lineal más adecuado para esa región específica. Concretamente en este proyecto, estas variables podrían ser tensión y/o corriente.

El GS puede proporcionar mejoras significativas en el rendimiento del sistema en comparación con los métodos lineales tradicionales. Sin embargo, también presenta desafíos en términos de diseño y sintonización del controlador. La implementación del GS puede llevarse a cabo de diferentes formas, siendo dos de las más comunes el enfoque por umbral y el enfoque de interpolación.

#### Enfoque por Umbral

El método por umbrales predefinidos se basa en variables observables para determinar la región operativa actual del sistema. Cada conjunto de ganancias se asigna a una región específica del espacio de estados, activándose el conjunto correspondiente

cuando el sistema atraviesa uno de estos umbrales. Este enfoque ofrece una implementación simple y directa, con una menor complejidad computacional en comparación con otros métodos. Sin embargo, una de las mayores desventajas son las posibles discontinuidades en el comportamiento del controlador en los límites de los umbrales y por lo tanto, mayor complejidad en la demostración de la estabilidad.

### Interpolación de Ganancias

En este enfoque, las ganancias se interpolan entre conjuntos definidos utilizando una función de interpolación. Este proceso ajusta las ganancias de manera continua según la posición actual del sistema en el espacio de estados, lo que permite una transición suave entre los conjuntos de ganancias y una adaptación dinámica a las condiciones cambiantes del sistema. Aunque este método ofrece una mejora significativa en la adaptabilidad y suavidad de la transición entre conjuntos de ganancias, su implementación conlleva una mayor complejidad computacional. Esto se debe al cálculo de las ganancias interpoladas y a la necesidad de utilizar una función de interpolación adecuada, lo que puede aumentar los requisitos de procesamiento y recursos computacionales.

Explicados ambos enfoques, y considerando que para este proyecto la potencia de cómputo resulta suficiente por las características que presenta la FPGA comentadas en Capítulos anteriores. Es entonces, que se decide utilizar la interpolación de ganancias debido a su capacidad de proporcionar una transición suave y continua entre conjuntos de ganancias, lo que garantiza una adaptación precisa y continua a las condiciones cambiantes del sistema.

#### 5.3.4. Análisis de sensibilidad de los polos respecto $i_L$

Para implementar la técnica de control GS, se utiliza la corriente del inductor ( $i_L$ ) como variable elegida para linealizar en distintos puntos de la planta. Por esta razón, resulta necesario llevar a cabo un análisis de la sensibilidad de los polos de la planta frente a diferentes niveles de corriente. Este análisis permitirá determinar las regiones más sensibles en las que deben operar los controladores para lograr una respuesta más eficiente. Observando la matriz de lazo abierto obtenida del sistema de ecuaciones que representa la planta con el lazo cerrado de corriente:

$$\begin{bmatrix} 0 & -\frac{1}{L}(1 - D_{ee}) & \frac{V_{oee}}{L} K_{i_I} & -\frac{V_{oee}}{L} K_{p_I} \\ \frac{1}{C}(1 - D_{ee}) & -\frac{1}{CR} & -\frac{I_{Lee}}{C} K_{i_I} & \frac{I_{Lee}}{C} K_{p_I} \\ 0 & 0 & 0 & -1 \\ W_c & 0 & 0 & -W_c \end{bmatrix}$$

Estos polos se determinan calculando el determinante de la matriz resultante, donde  $I$  representa la matriz identidad y posee las mismas dimensiones que  $A$ :

$$\det(A - I\lambda) = 0$$

Donde las raíces de la ecuación obtenida representan los polos del sistema en los valores evaluados. El sistema cuenta con cuatro polos, dos debidos a la planta, uno por el filtro y otro por el PI de corriente. Por lo tanto, al iterar  $i_L$  desde 0,5A hasta 16A, se obtiene la evolución de estos polos. Luego, se aproximaron los valores de cada polo (tanto la parte real como la imaginaria) con una función dependiente de la corriente, con el fin de obtener una expresión derivable y de esta forma poder analizar la sensibilidad de los polos. Es importante destacar que para el primer polo real se utilizó una función de aproximación exponencial debido a que se ajustaba mejor a la curva, mientras que para los demás polos se empleó una función polinómica. Finalmente, se derivaron cada una de las funciones y se ponderó cada una de las variaciones reales e imaginarias de forma cuadrática:

$$S_T(i_L) = \sqrt{\left(\sum \frac{dRe(polos)}{di_L}\right)^2 + \left(\sum \frac{dIm(polos)}{di_L}\right)^2}$$

Obteniendo la sensibilidad total de los polos respecto a la variación de la corriente sobre el inductor, como se puede observar a continuación (Figura 5.17):

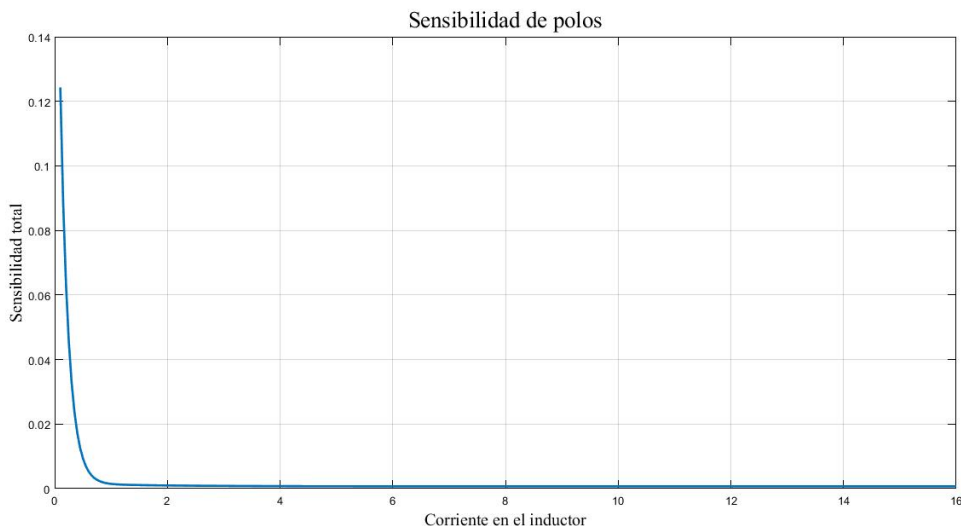


Figura 5.17 Sensibilidad de polos respecto a la corriente del inductor

Observando la Figura 5.17, se puede notar que en para corrientes pequeñas los polos poseen gran sensibilidad. Esta sensibilidad se adjudica a los polos mas rápidos

por lo que, a pesar de las variaciones significativas en esa región, la dinámica total permanece relativamente inalterada debido a que está dominada por los polos más lentos.

### 5.3.5. División de regiones para las corrientes de inductor

La mayor complejidad con respecto al GS, así como en la mayoría de las técnicas no lineales, es la demostración de la estabilidad del mismo. Existen distintas formas de implementación ya que se debe decidir como transicionar entre zonas. La mas conocida es utilizando un umbral en el cual actúa un PI hasta que se de la condición de cambio de zona y se cambia por otro PI. Siguiendo esta línea de razonamiento y considerando que la sensibilidad se mantiene casi constante en la mayoría de los valores de corriente, se decide dividir la planta en cinco regiones de igual tamaño para su linealización. Estas regiones serán de  $3A$ , donde se linealizará la planta con el valor de corriente en el centro de cada una de las regiones para la sintonización del PI correspondiente. Estas regiones se encontrarán distribuidas de la siguiente manera:

Región	Corriente	Resistencia media
1	1,5A	64Ω
2	4,5A	21,33Ω
3	7,5A	12,8Ω
4	10,5A	9,14Ω
5	13,5A	7,11Ω

Tabla 5.2 División de regiones para aplicar la técnica GS

### 5.3.6. Sintonización de PI en las regiones definidas

Una vez identificadas las zonas que requieren un controlador, se procede a ajustar cada PI de manera independiente utilizando la herramienta de ajuste *PID Tuner*. Este proceso se realiza mediante una aproximación lineal y ajustes progresivos utilizando el esquema linealizado que se muestra a continuación:

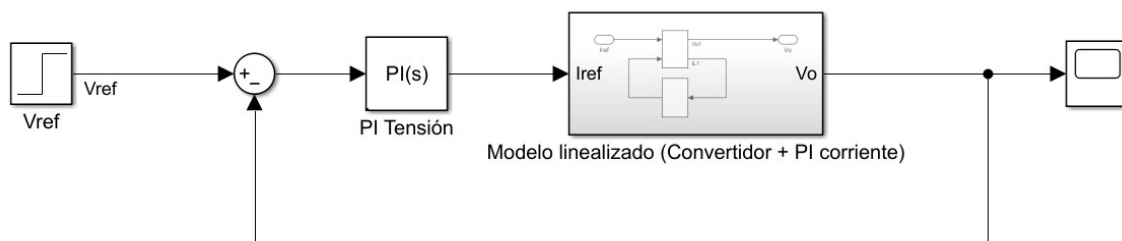


Figura 5.18 Modelo linealizado para lazo externo de tensión.

En base a este esquema, se ajustan los parámetros de los controladores para cada una de las regiones utilizando nuevamente la herramienta *PID Tuner*, la cuál fue presentada anteriormente. Durante este proceso, se mantuvo el mismo rango de valores tanto para el tiempo de establecimiento como para el de subida. Por lo tanto, recordando la expresión del controlador *PI* de tensión:

$$PI_v = K_{p_v} + \frac{K_{i_v} T_s}{z - 1}$$

Se elabora una tabla que muestra los parámetros de cada controlador sintonizado, junto con sus respectivos tiempos de establecimiento y de subida:

Región	$K_{p_v}$	$K_{i_v}$	Nombre	Tiempo establecimiento [ms]	Tiempo de subida [ms]
I	2,56	187	$V_{o_1}$	37.3	5.3
II	2,26	190	$V_{o_2}$	35.8	5.92
III	2,17	226	$V_{o_3}$	32	5.97
IV	2,1	278	$V_{o_4}$	28.8	5.85
V	1,99	313	$V_{o_5}$	27.1	5.91

Tabla 5.3 Sintonización de los PI en cada región

Luego de ajustar los controladores de cada región, se llevaron a cabo simulaciones individuales de cada controlador con el fin de verificar que todas las respuestas fueran similares. La siguiente Figura ilustra el comportamiento de cada controlador frente a un escalón unitario de referencia:

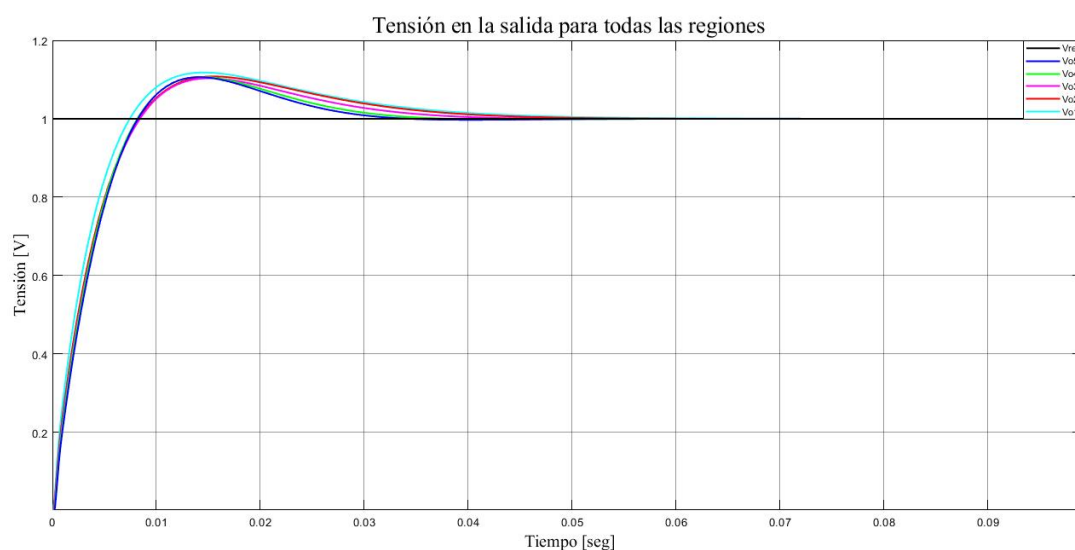


Figura 5.19 Tensión en la salida para todas las regiones

En esta imagen, se puede apreciar que todos los controladores exhiben respuestas similares y valores acordes a los especificados en la Tabla 5.3.

Es importante destacar que en este proyecto se busca mantener constante la tensión de salida del convertidor frente variaciones en la carga. Por esta razón, se diseñaron los controladores de tensión con una respuesta lo suficientemente rápida para rechazar eficazmente las perturbaciones. Esto por consecuencia, implica picos elevados de corriente por el inductor, los cuáles se tuvieron en cuenta a la hora de diseñar los controladores. No obstante, cuando se lleva el sistema desde una tensión inicial de  $24V$  hasta la tensión de operación de  $48V$ , se genera un pico extremadamente alto de corriente. Esto se debe a que los controladores son muy rápidos y necesitan llevar rápidamente el sistema al punto de operación deseado. Para abordar este problema, se implementa una limitación de variación (*Rate Limiter*) en la tensión de referencia, cuya función es limitar la corriente por el inductor hasta que el sistema alcance la tensión de operación de  $48V$ . Una vez que el sistema alcanza este punto, los controladores actúan exclusivamente para rechazar las perturbaciones.

Además, en el diseño del control, se implementa una acción de alimentación anticipada (*feedforward*), la cual será explicada detalladamente en la próxima subsección.

### **Feedforward**

El *feedforward*, o avance, es una estrategia de control utilizada para anticipar y compensar los efectos de las perturbaciones conocidas en un sistema dinámico. A diferencia del *feedback*, que responde a las señales de salida del sistema, el *feedforward* actúa directamente sobre las entradas del sistema con el objetivo de contrarrestar los efectos indeseados antes de que afecten a la salida.

El *feedforward* ofrece varias ventajas, como una respuesta más rápida a las perturbaciones y una mayor estabilidad del sistema. Es importante tener en cuenta que el *feedforward* no sustituye al *feedback*, sino que complementa su acción. Ambas estrategias de control se utilizan en conjunto para optimizar el rendimiento del sistema y garantizar una respuesta robusta frente a perturbaciones tanto conocidas como desconocidas.

Teniendo en cuenta lo mencionado anteriormente, se propone realizar un *feedforward* para llevar la planta a un punto de trabajo deseado. Aprovechando que se trabaja con una planta conocida, es decir el convertidor desarrollado en el Capítulo 3, es posible calcular la acción de control para lograr la tensión de salida deseada dado que se conoce la tensión de entrada. Recordando la siguiente expresión:

$$V_o = \frac{V_s}{1 - D} \quad (5.36)$$

Si consideramos los dos convertidores, se requiere la medición de la tensión de los SC y las BL para realizar sus respectivas acciones de control que, sumadas a la de *feedback*, conforman un control robusto y estable. A continuación se presenta el bloque *feedforward* implementado en la simulación:

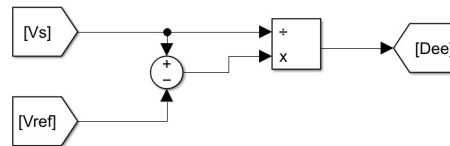


Figura 5.20 Feedforward implementado en la simulación

### Variación en la carga

Para observar el comportamiento de los controladores ante cambios en la carga, se realiza una simulación utilizando el software de Simulink®. En esta simulación, se modifica la carga conectada al convertidor elevador en lazo cerrado de tensión, lo que permite obtener una representación más precisa de situaciones reales en las que la carga experimenta variaciones durante su funcionamiento.

Para inducir modificaciones en la carga, se consideran los escenarios más desfavorables que podrían encontrarse en condiciones reales. Inicialmente, se varía la resistencia de carga desde un valor  $64\Omega$  (Región I) hasta un valor  $7,11\Omega$  (Región V) en  $t = 0,3$  segundos, reflejando así la perturbación más significativa que podría afectar al sistema. Posteriormente, en  $t = 0,4$  segundos, la resistencia se ajusta a un valor de  $21,33\Omega$  (Región II), luego a  $9,14\Omega$  (Región IV) en  $t = 0,5$  segundos y finalmente a  $12,8\Omega$  (Región III) en  $t = 0,6$  segundos. De esta manera, se representan algunas de las posibles perturbaciones, atravesando todas las regiones y observando el comportamiento de cada controlador PI sintonizado tanto en los peores escenarios como en los más favorables. El siguiente bloque diseñado en Simulink® es el encargado de realizar dicha tarea:

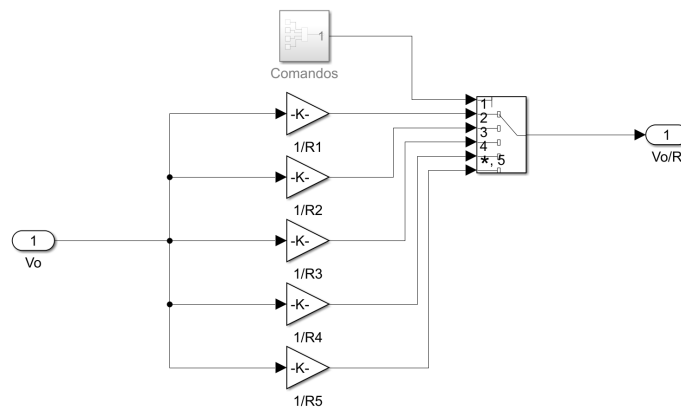


Figura 5.21 Bloque utilizado para simular variación en la carga

Empleando el bloque de variación en la carga y simulando un periodo de  $t = 0,7$  segundos, junto con el esquema propuesto de la Figura 5.22, se puede observar la respuesta de cada controlador PI sintonizado frente a este tipo de perturbaciones. Las cuales no presentan sobrepicos y son lo suficientemente rápidas para volver a la tensión establecida a partir de la referencia seleccionada.

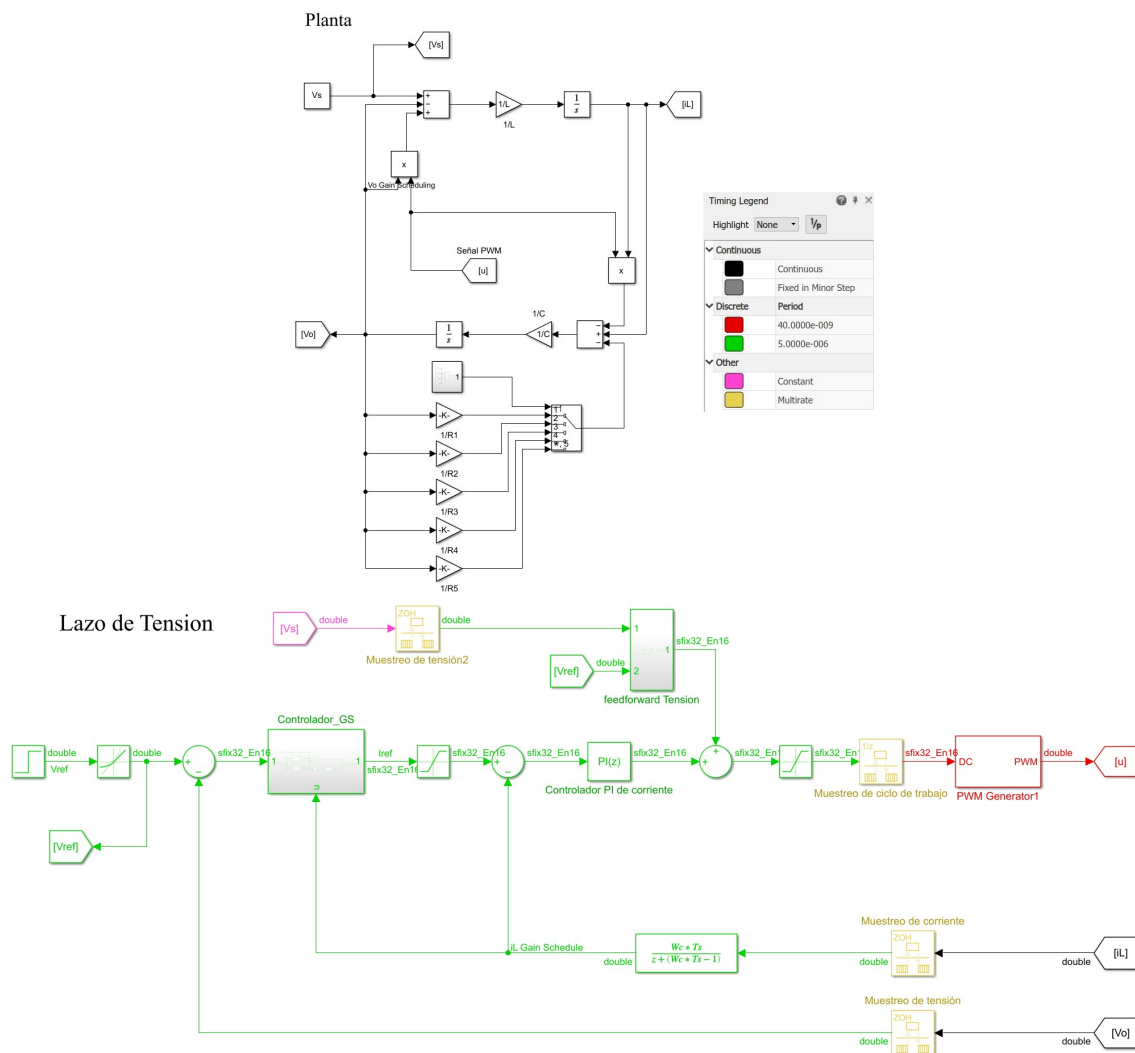
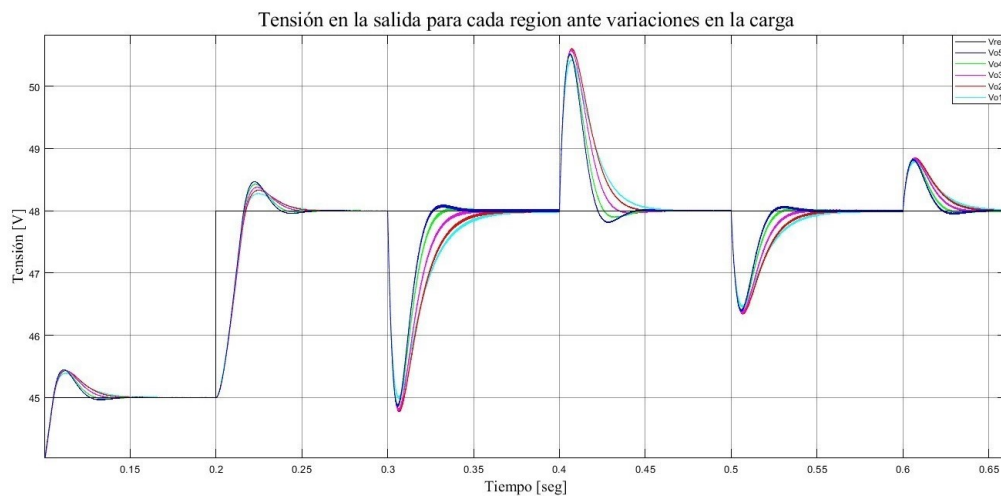


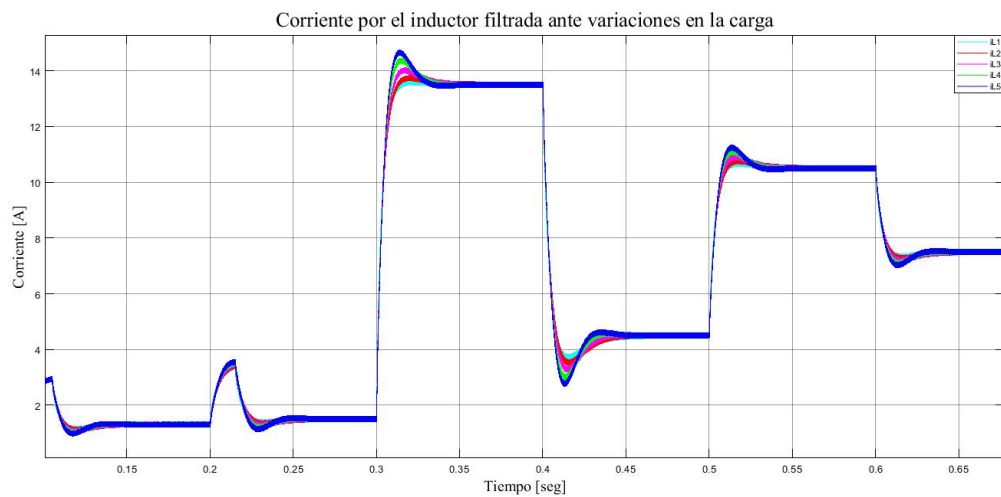
Figura 5.22 Simulación real del lazo de control de tensión.

Como se puede apreciar en la Figura 5.23a, cada controlador proporcional-integral muestra un rendimiento óptimo cuando opera dentro de su región específica. Por ejemplo, en  $t = 0,2$  segundos, el PI correspondiente a la región I exhibe un menor sobrepico y un tiempo de establecimiento similar respecto de las demás respuestas. De igual manera, en  $t = 0,3$  segundos, el PI asociado a la región V presenta la respuesta más rápida debido a la variación de la resistencia a  $7,11\Omega$ , y así sucesivamente para

las demás regiones. Sin embargo, aunque este último PI mencionado demuestra una mayor velocidad en algunas perturbaciones, en otras presenta una respuesta subamortiguada y un pico de corriente más elevado, lo cuál es indeseable. Por consiguiente, se hace necesario que exista una interacción entre los PI sintonizados de forma que cada controlador actúe en la región correspondiente, garantizando un comportamiento óptimo del sistema en cada situación.

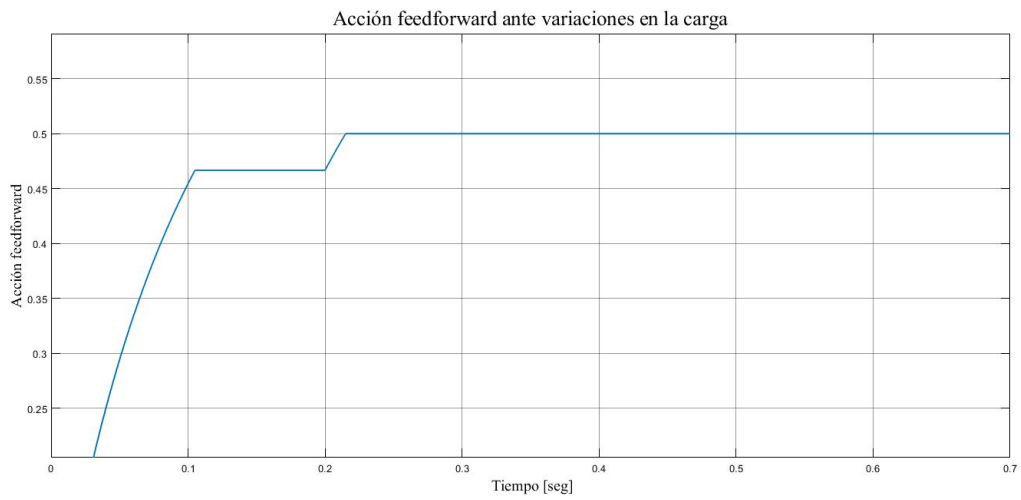


(a) Tensión en la salida del convertidor.

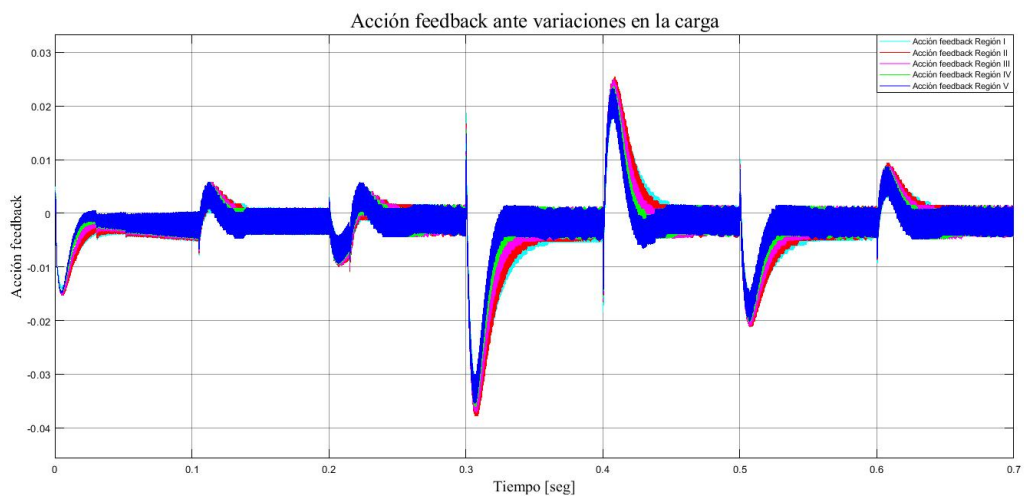


(b) Corriente por el inductor filtrada.

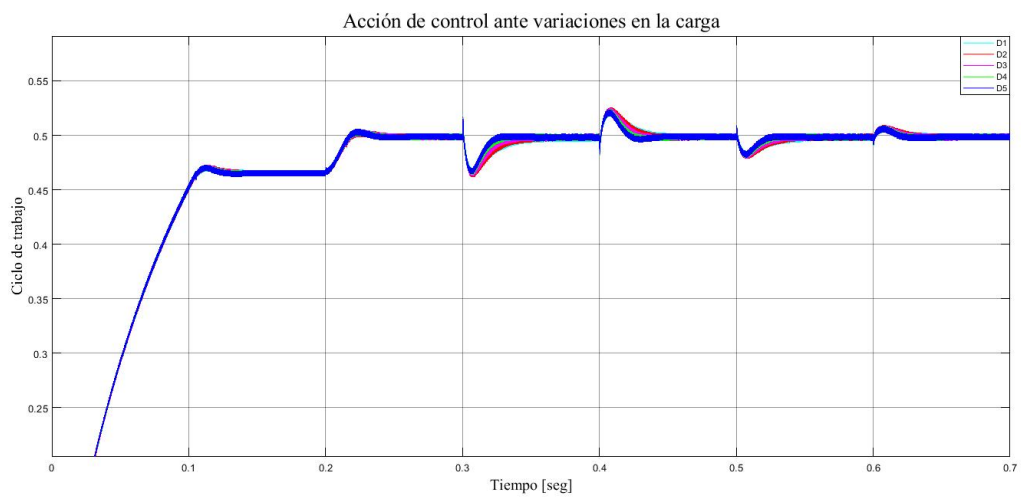
Figura 5.23 Formas de onda obtenidas ante una variación en la resistencia de carga.



(a) Acción feedforward.



(b) Acción feedback.



(c) Acción de control.

Figura 5.24 Formas de onda obtenidas ante una variación en la resistencia de carga.

### Funciones $K_p$ y $K_i$ para el control de tensión

Para garantizar que cada PI actúe en su respectiva región y avalar así un comportamiento óptimo del sistema, se adquieren los valores de cada constante proporcional e integral y se aproximan con funciones dependientes de la corriente tal como se muestra en la Figura 5.25. Esta aproximación asegura la ausencia de discontinuidades entre los cambios de controlador, lo que a su vez respalda la estabilidad del sistema. Para lograrlo, se lleva a cabo una interpolación utilizando un polinomio de segundo grado tanto para la constante proporcional como para la integral. Es importante observar que se utiliza el modulo de la corriente para el calculo ya que se tiene en cuenta la bidireccionalidad de los convertidores y se espera utilizar la misma acción de control sin importar el sentido, lo que conduce a las siguientes funciones:

$$\begin{cases} K_P(i_L) = 3,1746 \times 10^{-3} |i_L|^2 - 90,9524 \times 10^{-3} |i_L| + 2,6624 \\ K_I(i_L) = 634,9206 \times 10^{-3} |i_L|^2 + 1,8095 |i_L| + 178,0857 \end{cases} \quad (5.37)$$

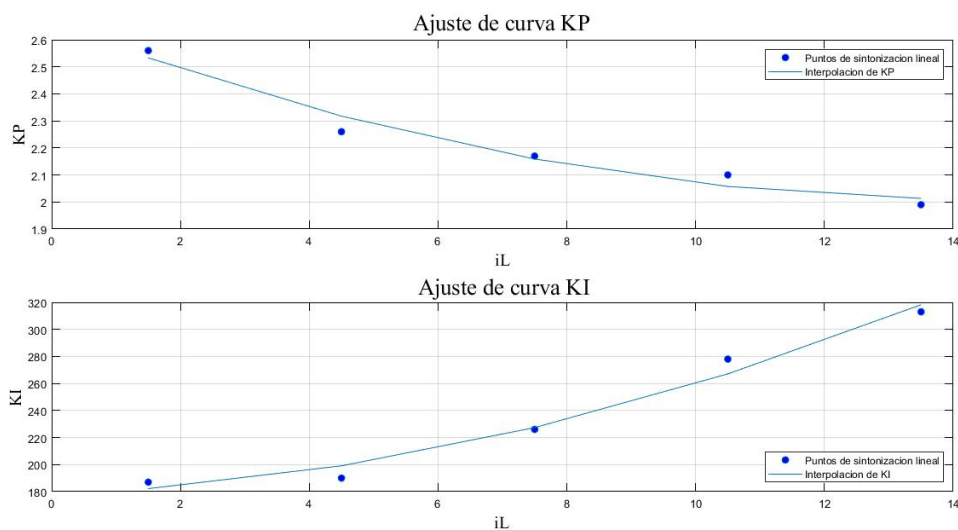


Figura 5.25 Comparación de interpolación vs puntos obtenidos

Dado que la constante proporcional disminuye a medida que se incrementan las regiones y la constante integral viceversa, se simplifica el ajuste del polinomio de segundo grado a la curva, presentando errores mínimos con los valores sintonizados. De esta manera, es posible asumir que la función propuesta exhibirá una respuesta muy similar a las simulaciones observadas anteriormente. Obtenidas ambas funciones, se procede a implementarlas mediante el siguiente bloque en Simulink®:

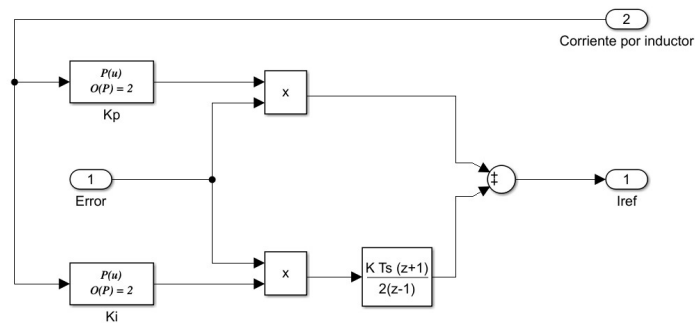


Figura 5.26 Implementación de la técnica GS

En este bloque, se introducen los coeficientes del polinomio obtenido, los cuales son multiplicados por el error, generando así los términos integral y proporcional.

### Análisis de estabilidad

En la presente sección se lleva a cabo un análisis de estabilidad de los polos en función de la corriente por el inductor teniendo en cuenta el control de tensión previamente diseñado, planteando así el lazo cerrado completo del sistema. Este análisis se lleva a cabo con el objetivo de asegurar que el control no presente inestabilidades al emplear la técnica de *Gain Schedule*. En la Figura 5.27 se muestra como se desplazan los polos del sistema en función de la corriente por el inductor. En conclusión, se observa que los polos varían dentro de una zona estable (polos en el SPI), lo que indica que el sistema mantendrá su estabilidad al modificar de PI utilizando esta técnica. Esto confirma la viabilidad y robustez del enfoque propuesto, asegurando que el sistema no se vuelva inestable al modificar los parámetros del controlador.

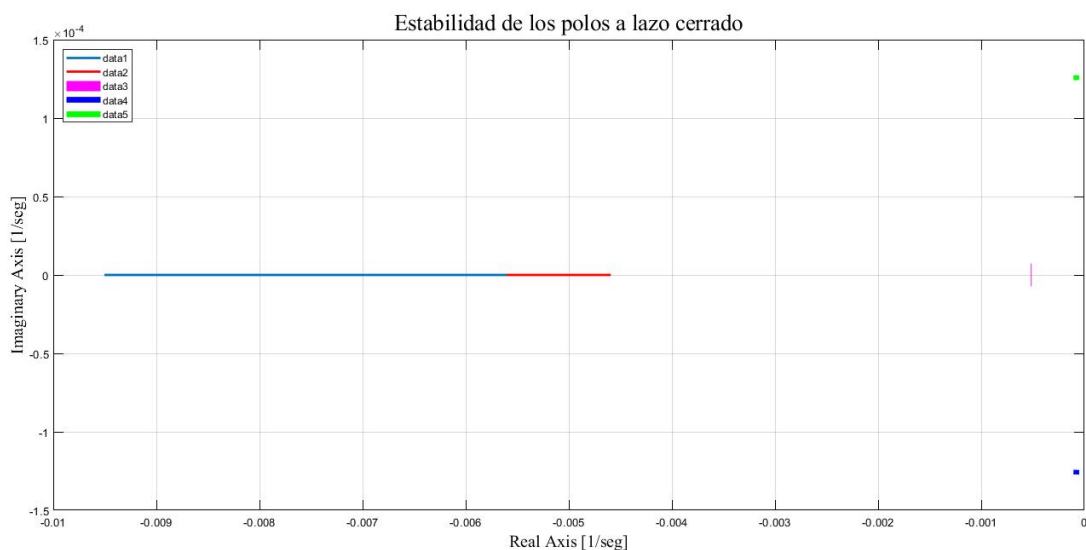
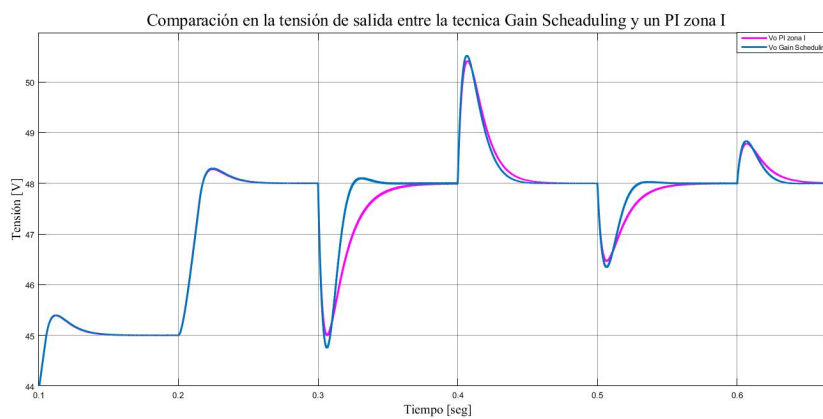


Figura 5.27 Estabilidad de los polos a lazo cerrado.

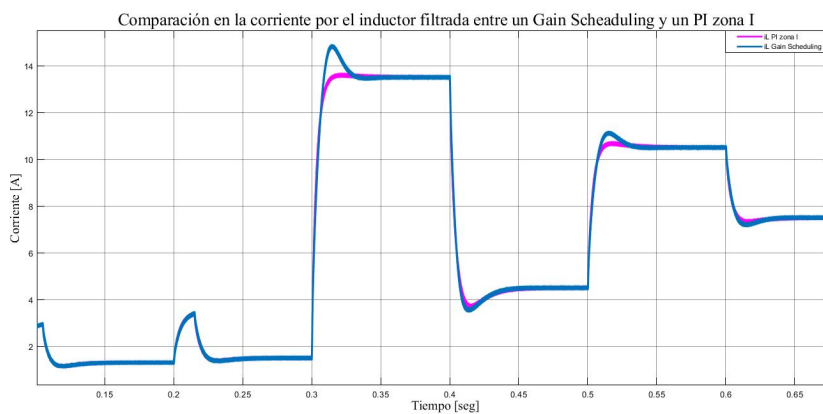
### 5.3.7. Mejoras de la planificación de ganancias en comparación al PI

En este proyecto se busca destacar el performance del GS respecto a un único PI. Debido a que la linealización de la planta en diferentes zonas implica un mayor tiempo de sintonización y desarrollo, esto permite realizar diferentes comparaciones en el performance. Para ello, se emplean los PIs sintonizados en la zona I y la zona V respectivamente, con el fin de observar las diferencias con el GS completo.

En la Figura 5.28a se observa la comparación con la *zona I*, es decir para la corriente del inductor de 1,5A. Se aprecia la misma dinámica en dicha zona ya que esta actuando el mismo PI en el GS. A medida que se realizan variaciones en la carga, se evidencia que el GS presenta una respuesta más rápida en comparación con el PI convencional, debido a que este último está sintonizado en una región de menor exigencia, lo que conlleva a una mayor presencia de sobrepicos de corriente. No obstante, incluso ante una perturbación de cinco regiones, la corriente se mantiene dentro de los límites manejados por el convertidor.



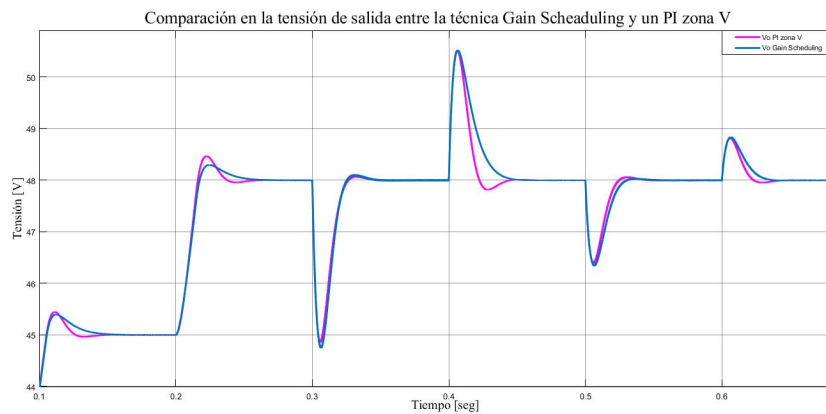
(a) Tensión en la salida del convertidor.



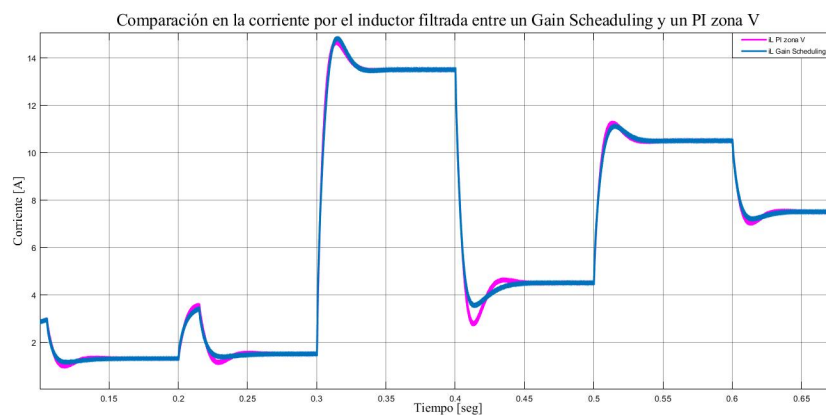
(b) Corriente por el inductor filtrada.

Figura 5.28 Comparación entre las formas de onda utilizando la técnica GS y un PI.

En la Figura 5.29a se observa la comparación con la zona V, es decir para la corriente del inductor de 13,5A. Así como en la anterior, se aprecia la misma dinámica en dicha zona ya que esta actuando el mismo PI en el GS. En las zonas de corrientes menos exigentes se distingue una dinámica mas rápida en la respuesta a las perturbaciones que el GS. Esto se debe a que se encuentra sintonizado en una zona de mayor exigencia, implicando acciones de control mas bruscas e induciendo una respuesta subamortiguada.



(a) Tensión en la salida del convertidor.



(b) Corriente por el inductor filtrada.

Figura 5.29 Comparación entre las formas de onda utilizando la técnica GS y un PI.

En conclusión, la implementación del GS representa una mejora significativa en comparación con la sintonización de un solo PI en términos de robustez y adaptabilidad del sistema de control. El GS ofrece una respuesta óptima en diversas situaciones sin la necesidad de reajustes manuales. Esto se evidencia en la capacidad del GS para mantener una respuesta constante y estable frente a perturbaciones en el sistema, adaptándose eficientemente a cambios de carga y condiciones operativas variables. Además, el GS ofrece una mayor flexibilidad al operar en múltiples zonas de operación,

garantizando un rendimiento óptimo en cada región sin comprometer la estabilidad del sistema.

#### 5.4. Modulo de almacenamiento híbrido: SC y BL

Finalmente, se quiere lograr la implementación de un modelo híbrido integrando ambos módulos en un mismo sistema junto con los respectivos convertidores y la carga asociada. Este enfoque permite plantear objetivos complementarios para cada uno:

- El Módulo de SC se encarga de suministrar los picos de potencia, regulando la tensión del bus y rechazando las posibles perturbaciones del sistema.
- El Módulo de BL tiene como función proporcionar la potencia media requerida por la carga, al mismo tiempo que se encarga de cargar los SC hasta un nivel de tensión deseado.

La Figura 5.30 ilustra la configuración del sistema completo, donde se pueden apreciar ambos módulos integrados junto con los convertidores y la carga.

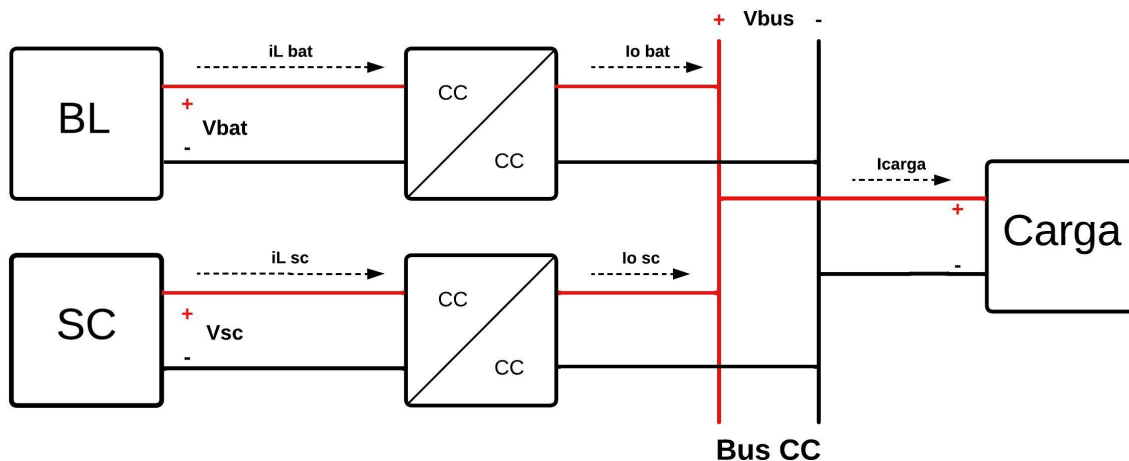


Figura 5.30 Esquema del sistema completo.

En este contexto, la inclusión de dos convertidores en un único bus conlleva la definición de tres nuevas variables de estados. Entre ellas, se encuentra la corriente a través del inductor de los SC ( $i_{L_{sc}}$ ) y la corriente a través del inductor de las BL ( $i_{L_{bat}}$ ). Para determinar dichas variables, se parte de la ecuación que describe la tensión en el inductor de un convertidor elevador genérico:

$$\begin{cases} V_L = V_s - V_o(1 - D) \\ i_L = \frac{V_L}{L} \end{cases} \quad (5.38)$$

Donde  $V_L$  representa la tensión del inductor del convertidor elevador,  $V_s$  la tensión de entrada del mismo y  $V_o$  la tensión de salida. Teniendo en cuenta las ecuaciones anteriores, se obtienen las siguientes expresiones para las derivadas de las corrientes de los inductores de cada convertidor, tal como se observa a continuación:

$$\begin{cases} \dot{i}_{L_{sc}}(t) = \frac{V_{sc}(t)}{L} - \frac{V_{bus}(t)}{L}(1 - D_{sc}) \\ \dot{i}_{L_{bat}}(t) = \frac{V_{bat}(t)}{L} - \frac{V_{bus}(t)}{L}(1 - D_{bat}) \end{cases} \quad (5.39)$$

En esta última ecuación es importante resaltar que  $L_{bat} = L_{sc} = L$ . Por otro lado, conociendo que la suma de las corrientes del inductor de cada convertidor es igual a la de carga y a su vez, teniendo en cuenta que dicha corriente se encuentra directamente relacionada con la tensión del bus:

$$\begin{cases} I_{o_{bat}} = I_{L_{bat}}(1 - D_{bat}) \\ I_{o_{sc}} = I_{L_{sc}}(1 - D_{sc}) \\ I_{carga} = I_{o_{sc}} + I_{o_{bat}} \\ \dot{V}_{bus} = \frac{I_{bus}}{C_{bus}} \end{cases} \quad (5.40)$$

Reemplazando las ecuaciones anteriores y considerando que el SC agrega otro estado más ( $V_{sc}$ ), se definen las variables de estado restantes:

$$\begin{cases} \dot{V}_{bus}(t) = -\frac{V_{bus}(t)}{RC_{bus}} + \frac{i_{L_{bat}}(t)}{C_{bus}}(1 - D_{bat}) + \frac{i_{L_{sc}}(t)}{C_{bus}}(1 - D_{sc}) \\ \dot{V}_{sc}(t) = -\frac{i_{L_{sc}}(t)}{C_{sc}} \end{cases} \quad (5.41)$$

Por lo tanto, unificando las Ecuaciones (5.39) y (5.41) se obtiene finalmente el sistema de ecuaciones que representa el modelo de la planta conformada por ambos sistemas de almacenamiento:

$$\begin{cases} \dot{i}_{L_{sc}}(t) = \frac{V_{sc}(t)}{L} - \frac{V_{bus}(t)}{L}(1 - D_{sc}) \\ \dot{i}_{L_{bat}}(t) = \frac{V_{bat}(t)}{L} - \frac{V_{bus}(t)}{L}(1 - D_{bat}) \\ \dot{V}_{bus}(t) = -\frac{V_{bus}(t)}{RC_{bus}} + \frac{i_{L_{bat}}(t)}{C_{bus}}(1 - D_{bat}) + \frac{i_{L_{sc}}(t)}{C_{bus}}(1 - D_{sc}) \\ \dot{V}_{sc}(t) = -\frac{i_{L_{sc}}(t)}{C_{sc}} \end{cases} \quad (5.42)$$

Con la planta real definida en su totalidad, se procede al diseño de la etapa de control requerida para el sistema completo. Para esta etapa, se consideró que la regulación de tensión del modulo de SC se logra con la planificación de ganancias junto con el controlador interno de corriente desarrollados previamente.

Por otro lado, para el diseño de la etapa de control de las BL se debe tener en cuenta la bidireccionalidad en todos los módulos incluyendo la carga. De modo que, cuando la carga requiera una potencia media  $P_x$ , los SC sean los encargados del transitorio descargándose hasta satisfacer dicha potencia, conllevando a la reducción de su nivel de tensión. Posteriormente, con una dinámica más lenta, las BL son las que suministran esa potencia media en estado estacionario. Si las BL suministran más potencia ( $P_{BL} > P_x$ ) al bus de continua, la potencia excedente que la carga no puede absorber es entregada a los SC aprovechando su bidireccionalidad. La velocidad de carga de los SC varía según la magnitud de la potencia sobrante. Siguiendo esta línea de razonamiento, en la siguiente Sección se procede a comentar la etapa de control de ambos módulos.

#### 5.4.1. Etapa de control del sistema completo

Para el control de las BL se propuso un enfoque que utiliza un PI de corriente similar al empleado en los SC, con el objetivo regular la corriente (y, por ende, la potencia) entregada por las BL al bus, que en estado estacionario será igual a lo que demanda la carga. Además, se introdujo un PI externo para regular la tensión de los SC con una dinámica más lenta, del orden de los minutos. Esta elección se justifica debido a la naturaleza gradual de la carga y descarga de las baterías de litio, que requieren un control más suave y menos reactivo en comparación con los SC, que pueden cargar y descargar rápidamente debido a su baja resistencia interna y alta capacidad de almacenamiento de energía. La implementación de una dinámica más lenta en el PI de tensión de las BL contribuye a una gestión más eficiente de la energía en el sistema, minimizando el riesgo de sobrecargas o sobredescargas de las mismas y alargando su vida útil.

Una vez que los SC se descargan al compensar el transitorio, el control de tensión de las BL genera la referencia necesaria para que la corriente de la batería sea ligeramente mayor que la de la carga. Esto permite que las BL suministren la potencia media requerida por la carga y, al mismo tiempo, recarguen los SC. En la Figura 5.31 se puede observar el diseño de la etapa de control para la batería implementado.

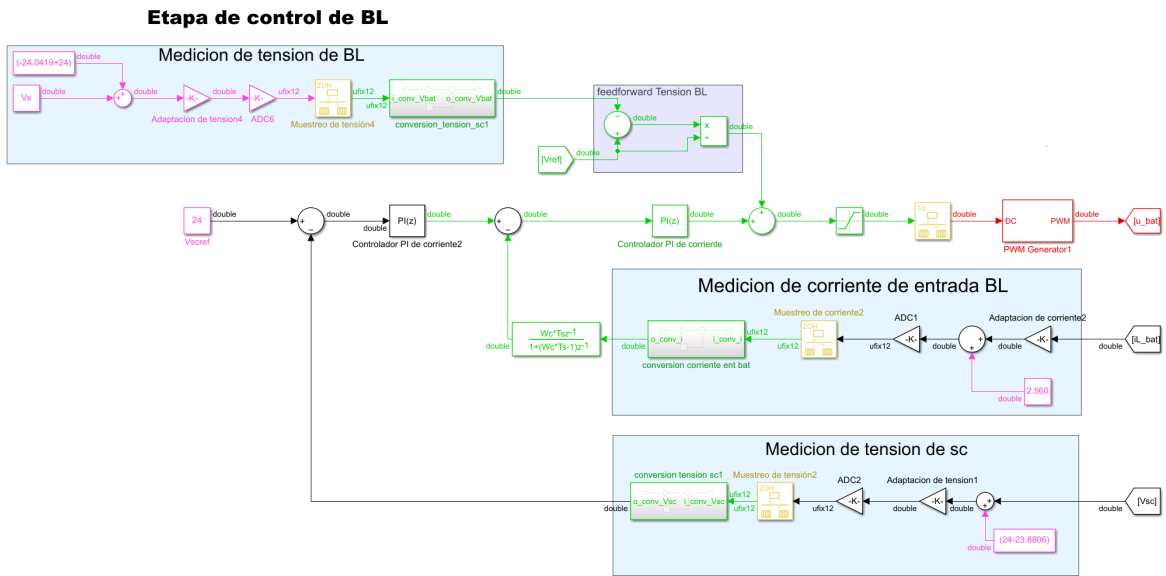


Figura 5.31 Diseño de la etapa de control para la batería.

Por otra parte, el lazo de control del banco de SC queda conformado como se ha discutido a lo largo de este capítulo:

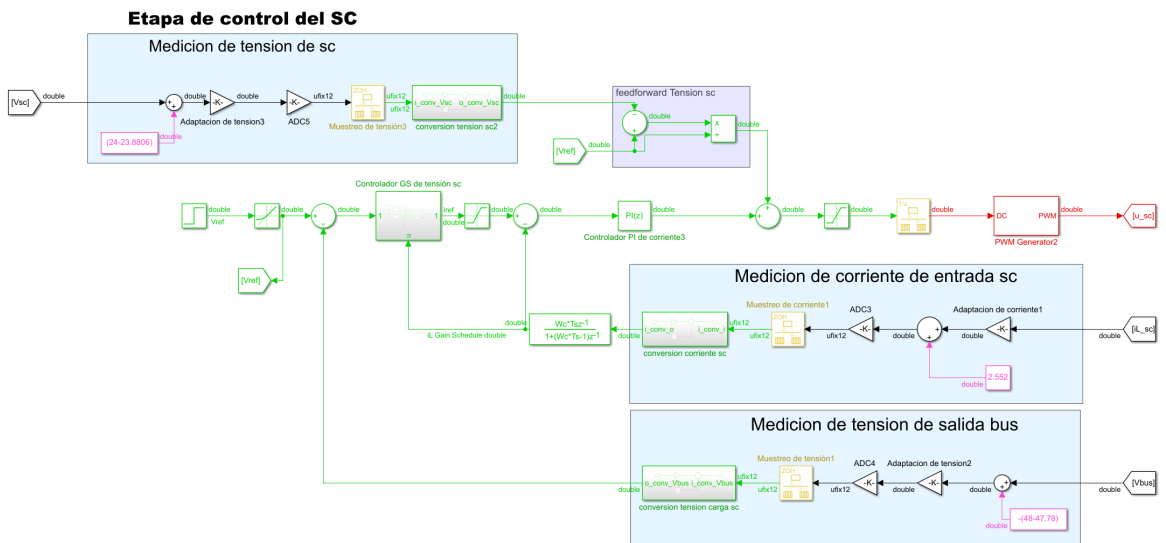


Figura 5.32 Diseño de la etapa de control para el banco de SC.

Planteando ambos sistemas de control, se procede a realizar una simulación de 70 segundos, aplicando una variación de carga con el objetivo de visualizar si el control se comporta conforme a lo esperado. Si bien se estima una dinámica del orden de los minutos para la carga de los SC, y dado que la complejidad computacional implica mucho tiempo de simulación, se utilizó un valor de  $20F$  como valor de SC (menor al utilizado en el LEICI de  $80F$ ). Para el PI de tensión de las BL se utilizaron 5 y 0,3 para la

constante proporcional e integral respectivamente. Esta simulación permite capturar las dinámicas del sistema bajo diferentes condiciones de carga, así como verificar la estabilidad y robustez del control implementado a lo largo del tiempo. Además, se analizará el desempeño del sistema en respuesta a una variación de carga, evaluando tanto la respuesta transitoria como el comportamiento en estado estacionario.

A continuación se observa el modelo de la planta implementado en Simulink® con sus respectivas formas de ondas obtenidas: la corrientes por el inductor del SC, las BL y la carga, además se gráfica la tensión del bus y del SC.

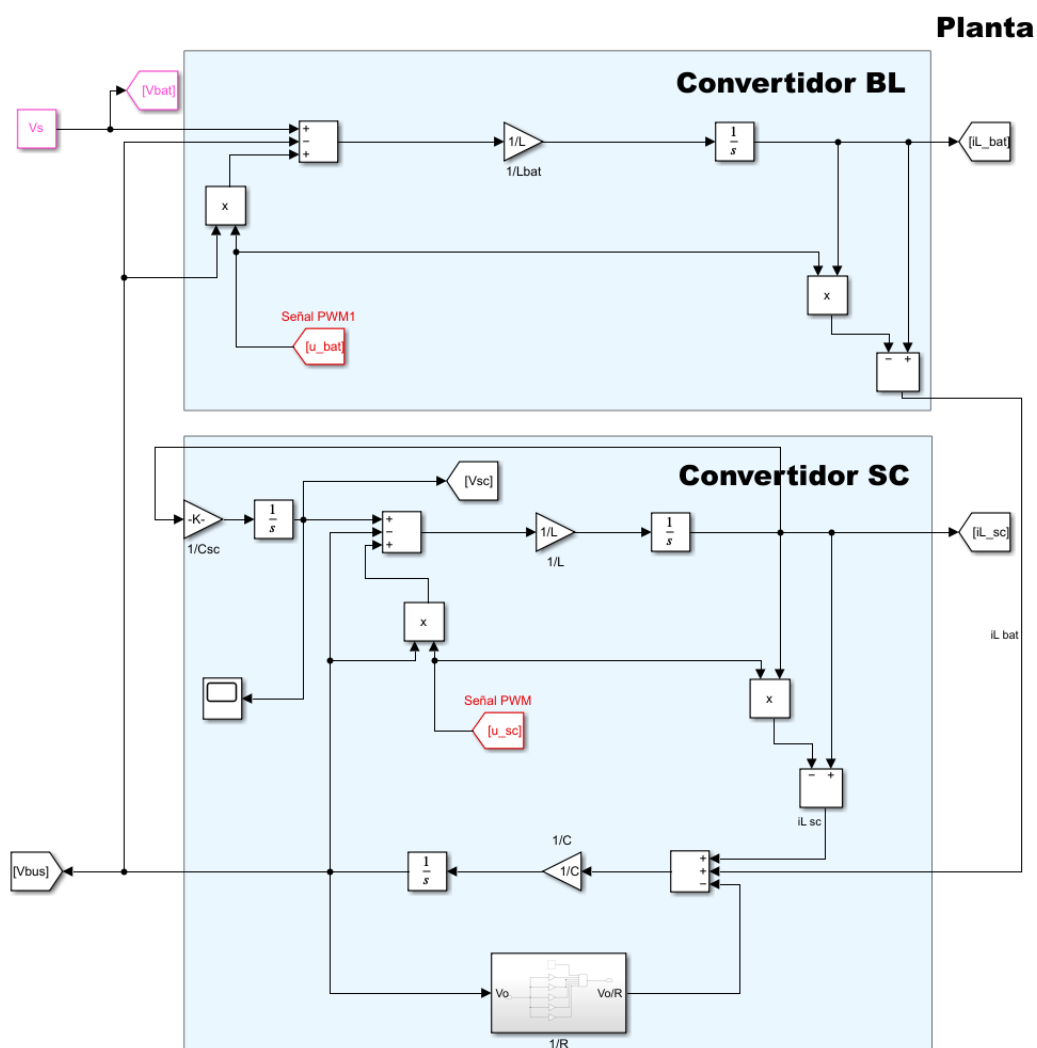
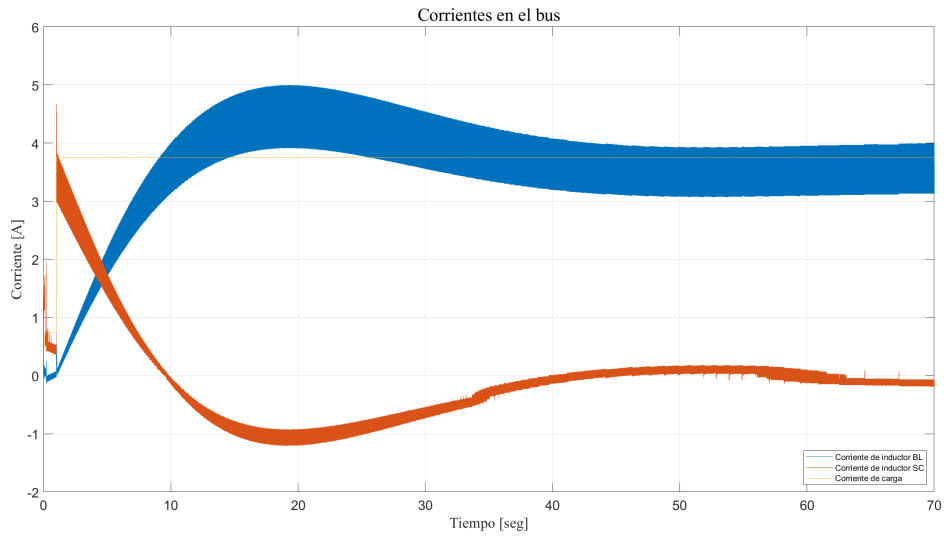
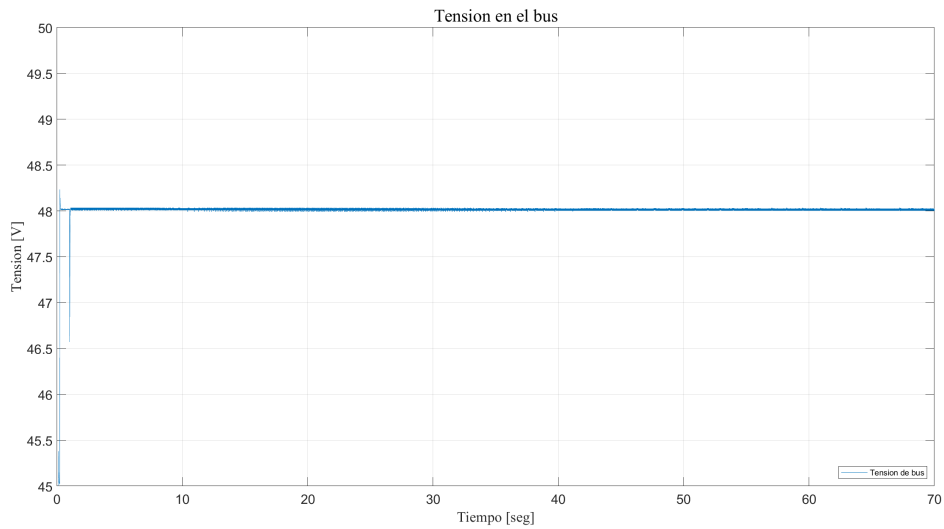


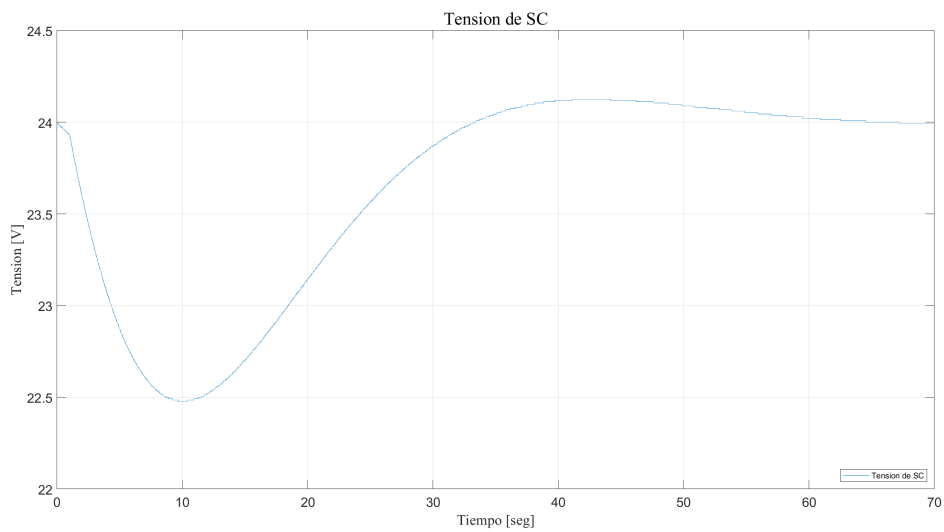
Figura 5.33 Planta real del sistema completo



(a) Corriente del sistema completo.



(b) Tensión de bus del sistema completo.



(c) Tensión del SC.

Figura 5.34 Formas de onda obtenidas mediante simulación del sistema completo.

La simulación inicia con una corriente de carga de  $0,6A$  y en el segundo  $0,2$  se realiza una variación en la carga. Se aprecia en la Figura 5.34a las respuestas de las corrientes a dicha perturbación. La corriente del SC responde rápidamente para compensar la caída de tensión del bus de continua (Figura 5.34b), resultando en la descarga de los SC. Luego, con una dinámica mas lenta, la corriente de las BL incrementa hasta lograr que la tensión de los SC vuelva a su referencia de  $24V$  (Figura 5.34c). Una vez que los SC alcanzan los  $24V$ , las BL se encarga de entregar la potencia demandada por la carga.

## 5.5. Resumen

En este Capítulo, se ha abordado el proceso completo de diseño del sistema de control, comenzando por el desarrollo de las ecuaciones matemáticas que describen sus componentes (filtro, controladores, etc). Se trabajó en la formulación del modelo linealizado del sistema, permitiendo así la aplicación de técnicas convencionales de análisis y control. Además, se llevó a cabo un estudio del filtro implementado y un estudio de la sensibilidad de los polos respecto a la corriente del inductor, asegurando la estabilidad del sistema ante las estrategias de control propuestas. Se complementó este análisis con una evaluación de la estabilidad del control diseñado a lazo cerrado considerando el modulo de almacenamiento híbrido.

Por otro parte, se realizaron simulaciones en cada etapa del proceso para verificar el funcionamiento adecuado de la acción de control proporcionada por la etapa de control propuesta. Donde, inicialmente se evaluó el diseño de control en un solo convertidor, comparando los resultados de la sintonización del PI de tensión con la técnica del GS, concluyendo en una mejora significativa con el segundo enfoque. Posteriormente, se evaluó el diseño de control considerando la planta en su totalidad. Finalmente, se llevaron a cabo simulaciones con variaciones en la carga, reproduciendo posibles eventos del sistema real y confirmando el comportamiento satisfactorio del control a lazo cerrado, así como su estabilidad.

El siguiente capítulo detallará la implementación de este sistema en el circuito de desarrollo utilizado, una FPGA, mediante el lenguaje Verilog. Se realizarán nuevas simulaciones en este entorno para confirmar la correcta implementación del sistema diseñado.

# Capítulo 6

## Diseño digital

### 6.1. Introducción

Este Capítulo se centra en la etapa de diseño digital del proyecto, con especial énfasis en la implementación del control y la construcción del sistema de comunicación.

Se dividirá en dos partes, por un lado en la Sección 6.2 se aborda el diseño a un alto nivel desde los sistemas de control diseñados en el Capítulo 5 en Simulink, siguiendo una metodología modular. Cada módulo se desarrolla de forma independiente a nivel de RTL, facilitando la comprensión, el desarrollo y la depuración del sistema. Esta aproximación modular también contribuye a prevenir problemas de integración, permitiendo la identificación temprana y la resolución eficiente de posibles inconvenientes. Con el propósito de asegurar el desempeño preciso de cada módulo, se emplearán testbenchs simulados mediante ModelSim. Estos testbenchs se configuran meticulosamente para verificar la funcionalidad individual de cada módulo, garantizando su conformidad con las especificaciones y requisitos de diseño.

Luego se procede con el sistema de comunicación en la Sección 6.3. Se utilizarán máquinas de estado para integrar el protocolo de comunicación diseñado. Estas desempeñarán un papel fundamental en la gestión eficiente de la comunicación entre los distintos componentes del sistema, proporcionando una estructura lógica y coherente para la transmisión de información con la PC.

También se implementarán lógicas con llaves físicas en la placa para realizar de forma progresiva la etapa de pruebas experimentales verificando el funcionamiento de los controladores.

En síntesis, este capítulo explora el diseño digital, haciendo hincapié en la validación modular a través de testbenchs para asegurar la funcionalidad precisa de cada componente antes de la integración final.

## 6.2. Diseño de módulos de control

En esta Sección se desarrollaran cada uno de los módulos del sistema de control diseñado en el capítulo anterior que se observan en la Figura 6.1. Iniciando por la lecturas realizadas por el ADC, luego el filtrado de las muestras y por ultimo los módulos de los controladores.

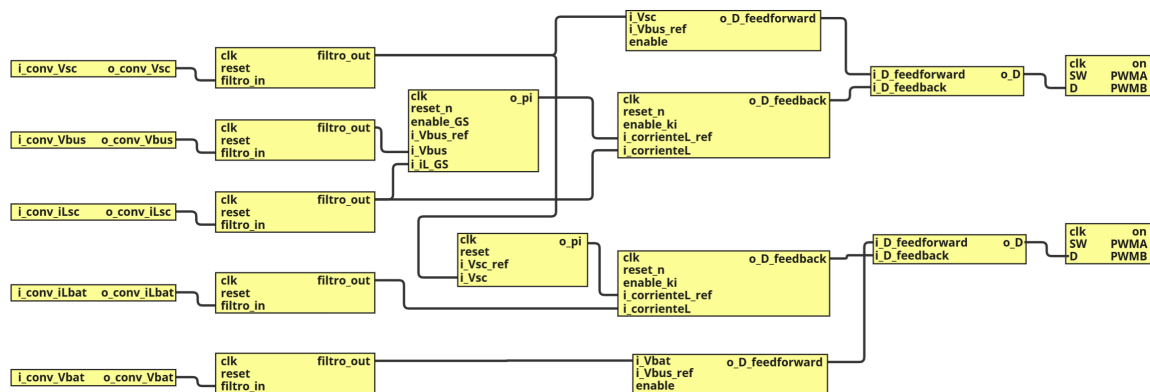


Figura 6.1 Sistema de módulos de control completo

La primera aproximación al diseño digital consiste en representar todas las variables del sistema de control desarrollado en el Capítulo 5 a punto fijo. Este paso tiene mucho impacto en el desempeño total del sistema ya que una mala representación puede causar distintos problemas. Por un lado se debe contar con una buena resolución para compensar los pequeños errores, y por otro una mala representación puede acarrear errores de overflow en las variables.

### 6.2.1. Conversor Analógico-Digital

Repasando algunos aspectos de la placa a utilizar DE10-Lite vistos en el Capítulo 4, el FPGA MAX 10 cuenta con un ADC (del inglés *Analog-Digital Converter*) interno de 12bits que trabaja con una frecuencia máxima de  $1\text{MHz}$ . Éste tipo de módulos con una aplicación específica es denominado IP-core. Este término se refiere a un diseño de propiedad intelectual predefinido y reutilizable, que se integra en el diseño digital para proporcionar funcionalidades específicas sin la necesidad de diseñarlas desde cero. La inclusión de un ADC como un IP-core en la placa MAX 10 simplifica el proceso de diseño y reduce el tiempo de desarrollo, al tiempo que garantiza un rendimiento óptimo y consistente del sistema.

Se utiliza una tensión de referencia de  $5\text{V}$  y la máxima frecuencia de muestreo  $1\text{MHz}$ . Como se requiere realizar cinco mediciones para efectuar el control, se configura un canal para cada una resultando en una frecuencia de muestreo de  $1\text{MHz}/5 = 200\text{kHz}$ .

La implementación de este bloque dentro del sistema es un bucle continuo que a cada clock del sistema ( $50MHz$ ) verifica la finalización de la lectura del ADC. De ser así, se cambia el canal hasta completar los cinco.

Una vez obtenidas las lecturas de las variables, estas ingresan a los módulos convertidores que se encargan de obtener la representación de los datos en punto fijo a partir de su medida de 12 bits.

Para obtener la función que transforma los valores del ADC a los valores reales se realizaron mediciones sobre los convertidores. Las mediciones de tensión son realizadas con un amplificador aislado y luego adaptadas con circuitos utilizando amplificadores operacionales. Las mediciones de corriente son realizadas con sensores de efecto hall y también son adaptadas con amplificadores operacionales. Cada una de estas tienen un rango de valores de medida:

- $V_{bus}$ : 0V a 80V
- $V_{SC}$ : 0V a 30V
- $V_{BL}$ : 0v a 30V
- $I_{LSC}$ : -15A a 15A
- $I_{LBL}$ : -15A a 15A

Para las mediciones se utilizó una fuente controlada externa, capaz de regular tensión o corriente, y se midió la tensión aplicada al ADC. Para el caso de las mediciones de tensión, tensión de entrada y salida del convertidor de los SC y tensión de entrada del de BL, se conectó la fuente a la salida o entrada y se aumentó la tensión de a 5V en su respectivo rango y midiendo la tensión del ADC se relevan los puntos de la curva, siendo estos el valor de salida del ADC en 12 bits y el valor de tensión aplicado. Para las mediciones de corriente se modificó la fuente a regulación de corriente y se aumentó de a 0,5A en todo el rango, cambiando los bornes de la fuente cuando la corriente cambia de signo. Luego, se utiliza una aproximación lineal para obtener la función que relaciona el valor de salida ADC ( $N_x$ ) y el valor de la medición real.

$$V_{bus} = 0,021025890388122 \times N_x + 0,336560100240133 \quad (6.1)$$

$$V_{SC} = 0,007570692161031 \times N_x - 0,056423765392081 \quad (6.2)$$

$$V_{BL} = 0,007465131795654 \times N_x - 0,254899031191860 \quad (6.3)$$

$$I_{LSC} = -0,008971907544623 \times N_x + 18,733570241497013 \quad (6.4)$$

$$V_{LBL} = -0,009107293454962 \times N_x + 18,983079680355484 \quad (6.5)$$

En las siguientes figuras se observan como se ajustan las curvas interpoladas a los puntos relevados para cada una de las mediciones:

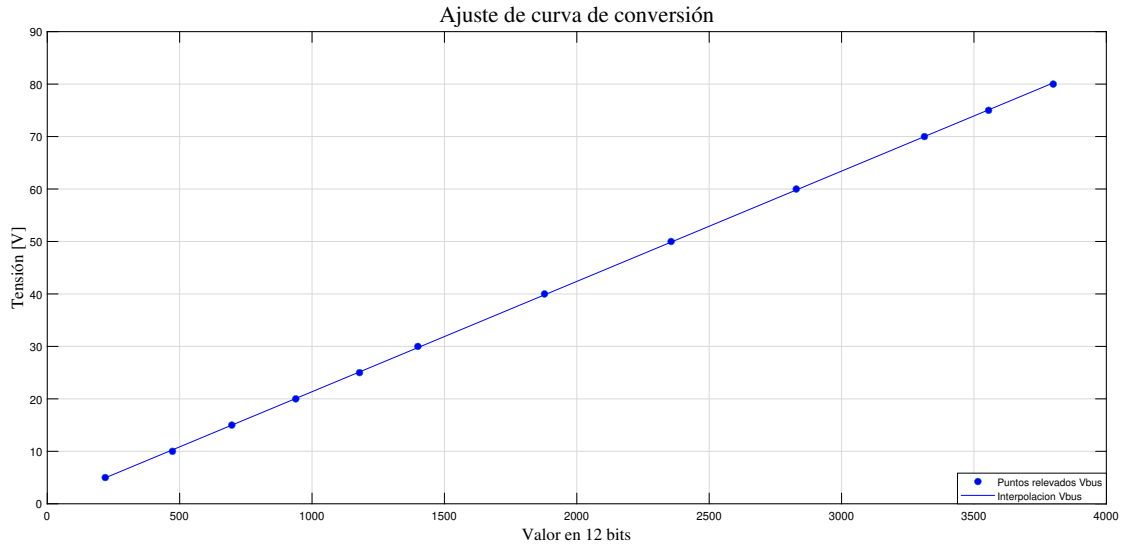


Figura 6.2 Interpolación de tensión de bus

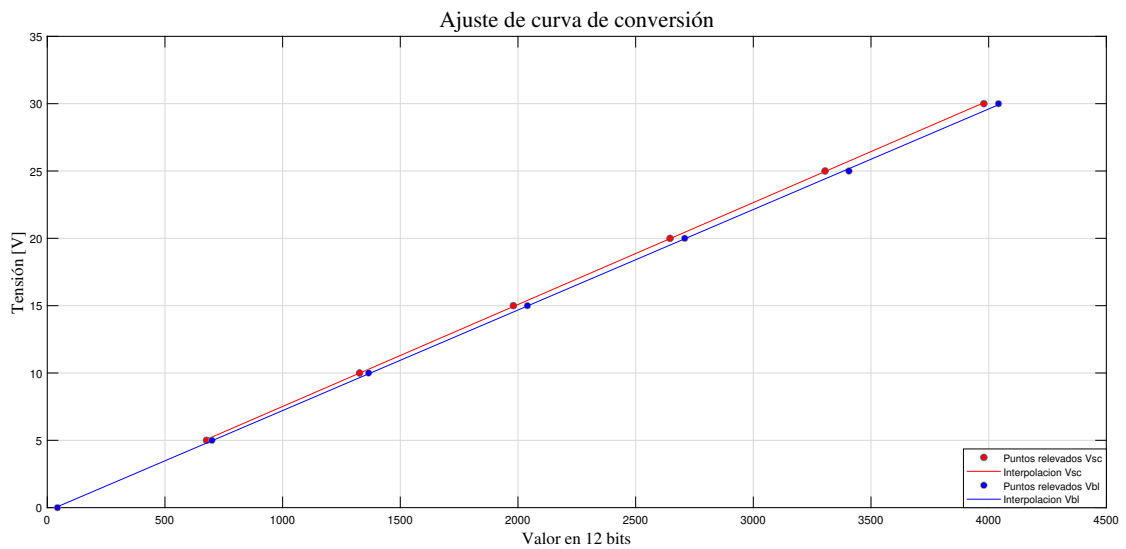


Figura 6.3 Interpolación de tensión de SC y BL

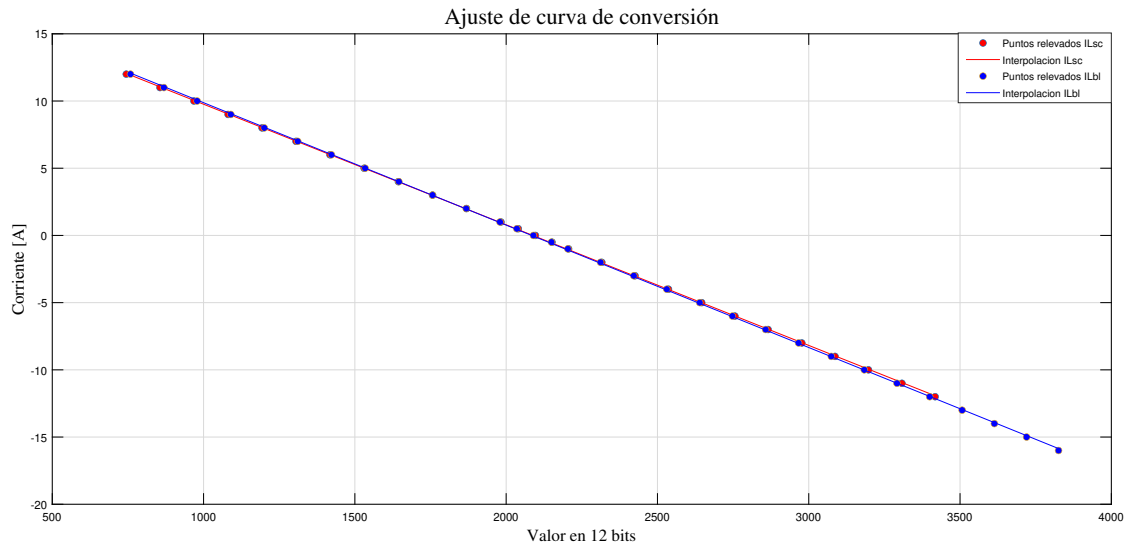


Figura 6.4 Interpolación de corriente de inductor de SC y BL

Una vez se ingresa con el valor del ADC se realiza una multiplicación y luego se suma una constante para lograr el valor real. Para el diseño se tuvo en cuenta la primera multiplicación ya que depende de la resolución que se utilice para la cuantización del coeficiente la representación de la variable en el sistema. La máxima resolución posible viene dada por el ADC, es decir un bit del ADC representa la menor tensión que se puede observar. Con una tensión de referencia de  $5V$ , un bit representa:

$$\frac{5V}{2^{12}} = 0,001220703125V \quad (6.6)$$

Luego para obtener la cantidad de bits capaces de representar el número luego de multiplicar por el coeficiente  $0,02102565765$  se hace el logaritmo en base 2:

$$\log_2(0,001220703125 \times 0,02102565765) = -15,2498 \quad (6.7)$$

Con esto se determina que es necesario mínimamente 16 bits de fracción para no perder información y se decide utilizar una representación con 20 bits fraccionales de 32.

En la Figura 6.5 se observa el puerto de entrada de 12 bits y el puerto de salida de 32 bits de uno de los módulos.

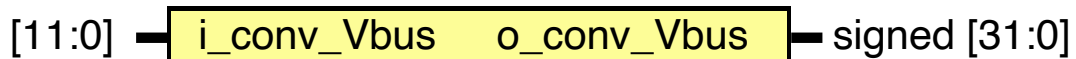


Figura 6.5 Diagrama de puertos de módulo de conversión de tensión de bus

### 6.2.2. Filtro digital

El filtro digital propuesto en la Sección 5.3.1 se trata de un filtro IIR (del inglés *Infinite Impulse Response*). La Ecuación (5.24) representa a la transferencia discreta del filtro a implementar. Desarrollando y expresando en potencias negativas de Z se obtiene:

$$Y(z) - 0,954Z^{-1}Y(z) = 0,046Z^{-1}X(z) \quad (6.8)$$

Antitransformando y reordenando:

$$y[n] = 0,046x[n - 1] + 0,954y[n - 1] \quad (6.9)$$

Para la implementación se utilizó la forma directa I representada en el RTL (register-transfer level) de la siguiente Figura:

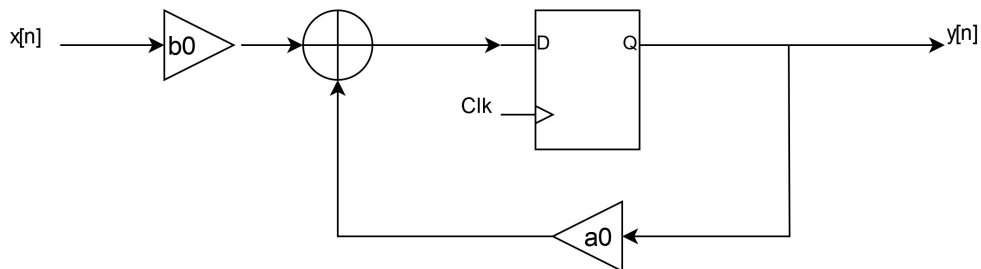


Figura 6.6 RTL de filtro IIR en forma directa I

En la Figura 6.7 se observa un diagrama donde se indican los puertos de entrada y salida del módulo.



Figura 6.7 Diagrama de puertos de filtro

Para el testbench de la Figura 6.8 se genera una señal de entrada conformada por dos sinusoidales, una de  $550\text{Hz}$  y otra de  $20\text{kHz}$  para simular el ruido. Se puede observar un filtrado y un retraso de la señal acorde con la Figura 5.8. Luego se verifica el funcionamiento del reset sincrónico, llevándolo de 0 a 1 y se observa una salida nula.

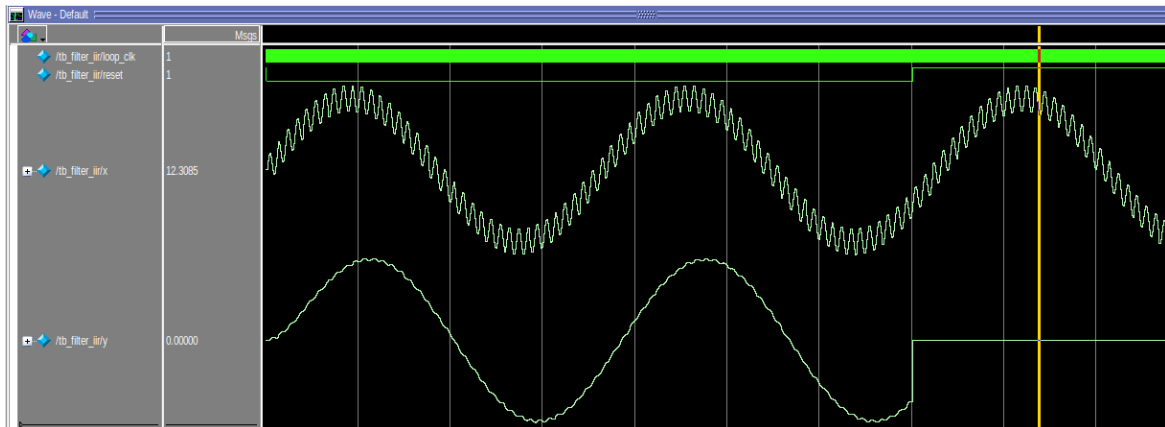


Figura 6.8 Testbench de filtro

### 6.2.3. Controlador PWM

El módulo PWM se encarga de excitar las llaves de los convertidores según el ciclo de trabajo ingresado. Estas señales de salida deben ser complementarias, es decir cuando la de la columna superior se encuentra abierta, la otra cerrada y viceversa. Se utiliza la salida "on" como verificación del estado activo de las salidas en las pruebas. En el siguiente diagrama se observan los puertos:

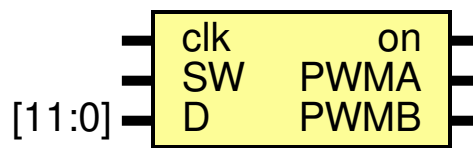


Figura 6.9 Diagrama de puertos de módulo PWM

El funcionamiento consta de un contador utilizando el clock de entrada y manteniendo la señal de salida PWMA en alto hasta que este alcance el valor del ciclo de trabajo y luego bajarla. La señal PWMB al ser complementaria esta en bajo el mismo tiempo que PWMA en alto y espera un clock más para ponerse en alto. Esto es para evitar que las llaves de potencia se encuentren cerradas al mismo tiempo.

Se realiza un testbench para verificar el funcionamiento, donde inicialmente se tiene SW en alto obteniendo las salidas y luego al pasar a cero se observa como las salidas se anulan.

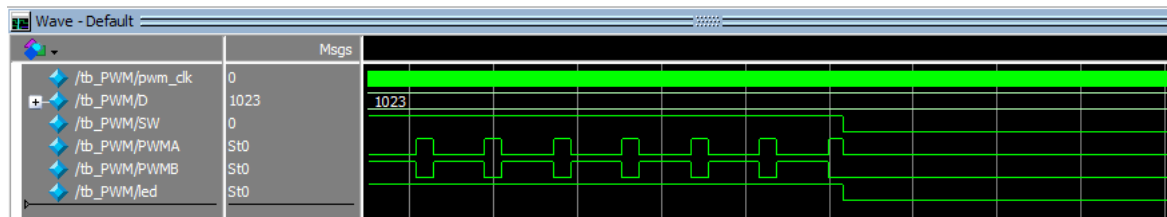


Figura 6.10 Testbench de módulo PWM

Luego para apreciar el retardo de un clock entre las señales complementarias, se hace zoom a un flanco que se ilustra en la siguiente Figura:

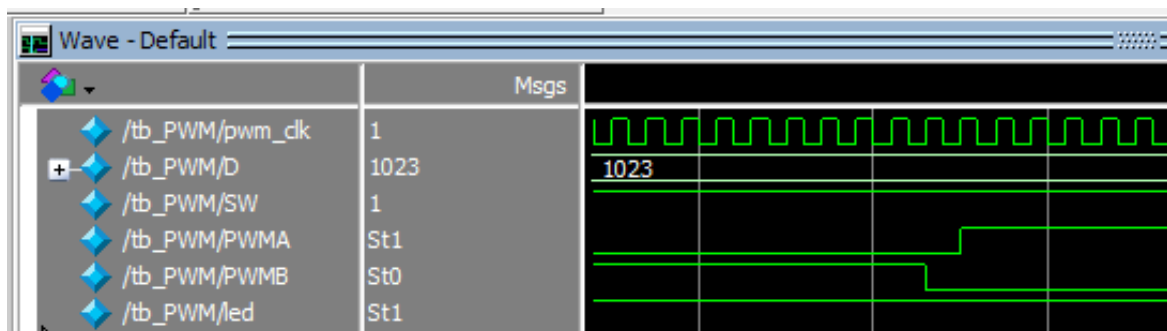


Figura 6.11 Zoom flancos de PWM

#### 6.2.4. Controlador Proporcional-Integral

El sistema completo cuenta con tres controladores Proporcional-Integral (PI). Dos son utilizados en el lazo interno de corriente de cada convertidor y luego se utiliza un tercero, con distintas constantes, en el lazo externo de tensión del convertidor de las BL.

Éstos módulos cuentan con cuatro entradas y una salida. En la Figura 6.12 se observan los puertos del PI de corriente al cual ingresa el clock, el reset negativo, el habilitador del integrador, la corriente del inductor, su referencia y tiene como salida el ciclo de trabajo del lazo realimentado. En la Figura 6.13 se aprecian los puertos del PI de tensión de las BL al cual ingresa la tensión de los SC, su referencia y como salida tiene la corriente del inductor de las BL.

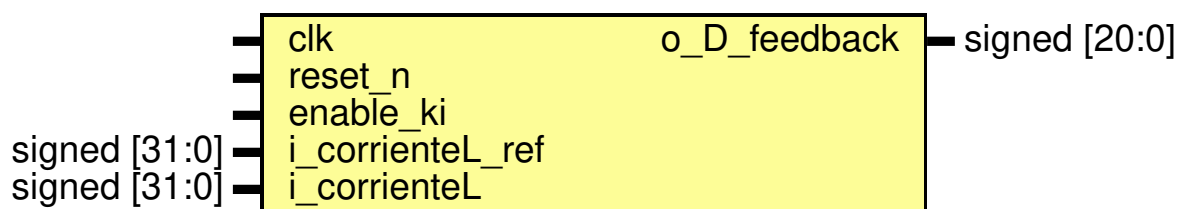


Figura 6.12 Diagrama de puertos de módulo PI de corriente.



cuando vuelve a 1 se puede observar que el integrador reseteo su valor al empezar a integrar desde 0. Esto se puede observar en la simulación de la Figura 6.15.

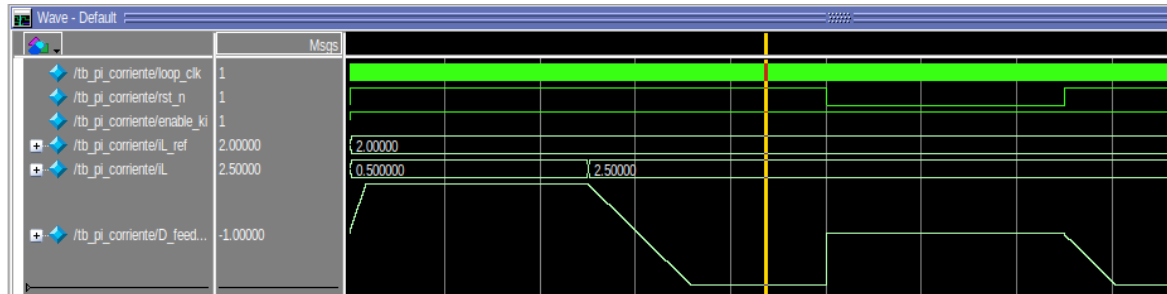


Figura 6.15 Testbench PI de corriente

### 6.2.5. Controlador Proporcional-Integral de tensión GS

Este módulo aplica el mismo principio de funcionamiento que el PI de tensión pero agregando la corriente del inductor del convertidor de los SC y un habilitador de la planificación de ganancias como entradas adicionales como se observa en el diagrama de puertos del módulo de la Figura 6.16. La salida del PI es la corriente de referencia del inductor y el error es construido con la tensión de bus con la referencia. Cuando la planificación de ganancias se encuentra deshabilitada, se utilizan las ganancias fijas sintonizadas en la zona I.

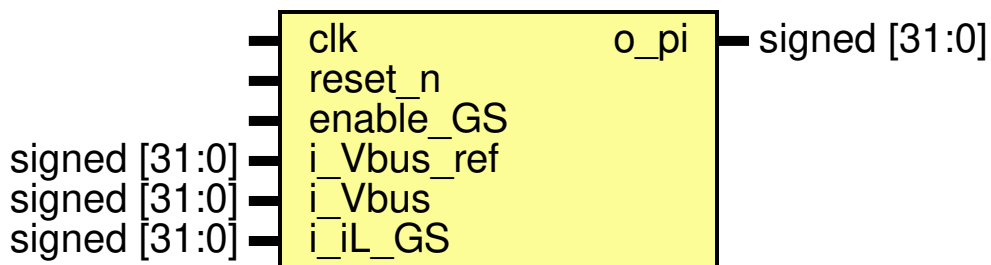


Figura 6.16 Diagrama de puertos de módulo GS

En la Figura 6.17 se observa el RTL del módulo de GS.

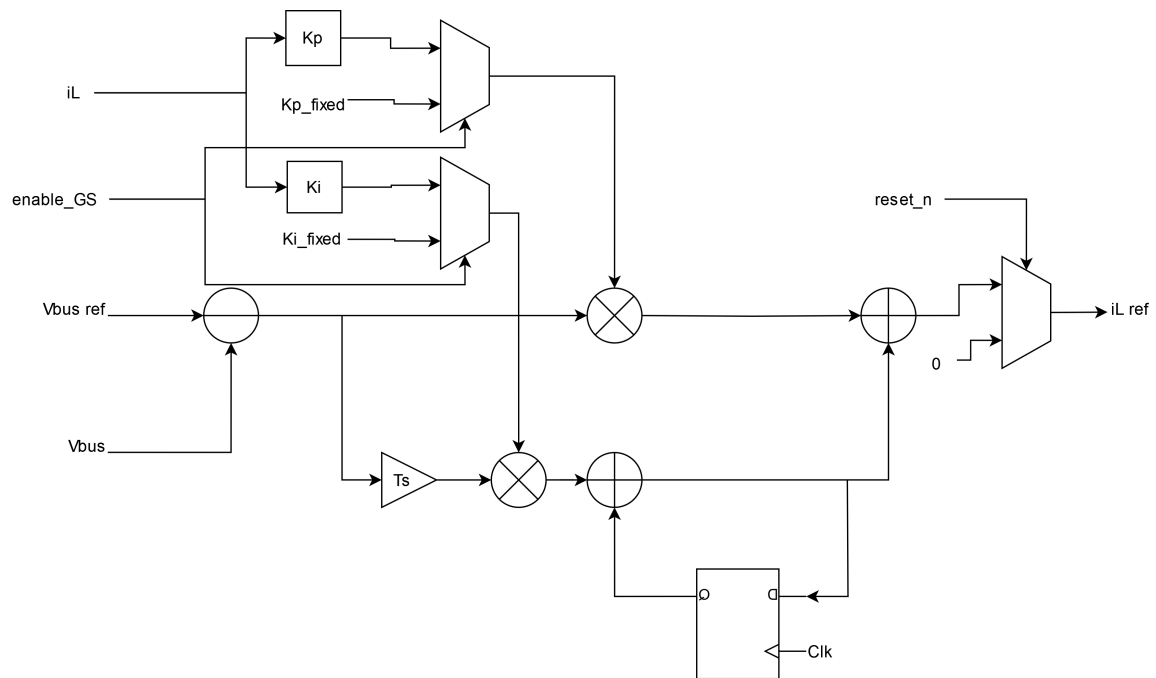


Figura 6.17 RTL de PI de tensión con GS

El testbench se construye para verificar las mismas funciones del PI anterior: el anti-windup, el reset para deshabilitar el módulo y resetear el integrador. Estas se realizan a distintos valores de corriente para observar un cambio en la respuesta que evidencie la actuación del GS, es decir, al tratarse de errores constantes, las acciones de control deben ser rampas y en estas se evidencia un cambio de pendiente. En la Figura 6.18 se observa la simulación del testbench.

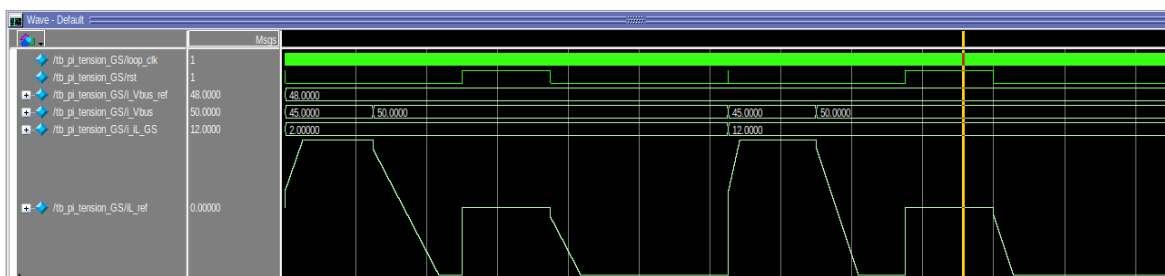


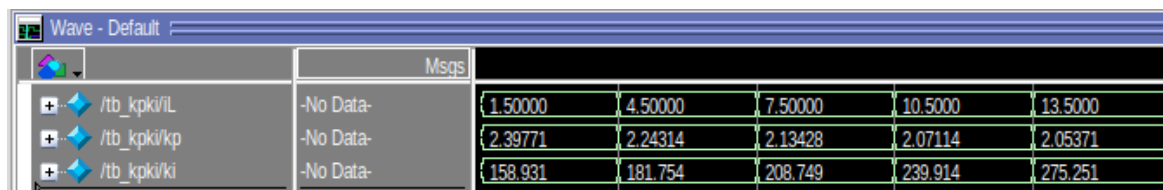
Figura 6.18 Testbench PI de tensión con GS

Dentro del módulo se utilizan dos módulos adicionales  $K_p$  y  $K_i$  los cuales únicamente tienen como entrada la corriente del inductor y de salida el valor de la constante, es decir que se trata de módulos concurrentes al no utilizar el clock. Estos módulos implementan las funciones logradas en el Capítulo 5. Es importante resaltar que al tratarse de convertidores bidireccionales, las corrientes pueden ser positivas y negati-

vas pero la variación de la dinámica del convertidor se debe a la magnitud de ésta, es por esto que se utiliza el valor absoluto de corriente.

### Kp y Ki

Estos módulos cuentan con una entrada y una salida, la corriente del inductor y Kp o Ki respectivamente. Se trata de módulos concurrentes al no contar con clock de entrada. En la Figura 6.19 se observa la simulación del testbench en la que se ingresan los distintos valores de corriente y se comprueban los valores de Kp y Ki obtenidos en la Figura 5.25 verificando los funcionamientos de cada uno.



Msgs						
/tb_kpk/iL	-No Data-	1.50000	4.50000	7.50000	10.5000	13.5000
/tb_kpk/kp	-No Data-	2.39771	2.24314	2.13428	2.07114	2.05371
/tb_kpk/ki	-No Data-	158.931	181.754	208.749	239.914	275.251

Figura 6.19 Testbench módulos Kp y Ki

### 6.2.6. Feedforward

El módulo Feedforward utiliza la tensión de entrada y conociendo la dinámica de la planta (en este caso el convertidor) desarrollada en el Capítulo 5, se calcula la acción de control necesaria para llegar a la referencia deseada. Esta acción de control se realiza como polarización para alcanzar el punto de trabajo deseado y de esta forma el ciclo de trabajo debido al Feedback se encarga de las variaciones mas finas, es por esto que se decide utilizar 12 bits para el ciclo de trabajo del Feedforward. Se trata de un bloque concurrente al no contar con clock de entrada. En la Figura 6.20 se observa un diagrama de puertos del módulo implementado.

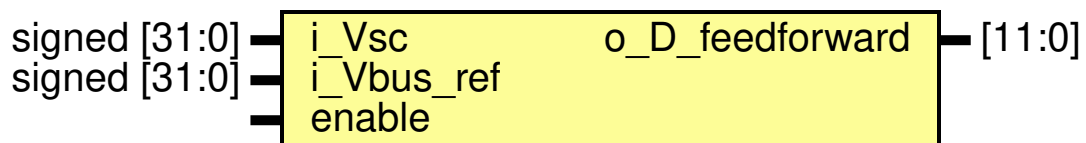



Figura 6.20 Diagrama de puertos de módulo feedforward

Para el testbench se instancio dos veces el módulo, uno para las BL y otro para los SC. Iniciando con los módulos deshabilitados se observa que las señales enable funcionan al ponerse en alto. Luego se observa que para distintos niveles de tensión respecto a la tensión de bus, se obtienen los ciclos de trabajo correspondientes. Esto se verifica utilizando la ecuación que relaciona el ciclo de trabajo y las tensiones de entrada y salida.



Signal	Value	Time	Value	Time	Value
/tb_feedforward/enable1	1				
/tb_feedforward/enable2	1				
/tb_feedforward/V_bus_ref	48.0000		48.0000		
/tb_feedforward/V_sc	24.0000		24.0000		
/tb_feedforward/V_bat	30.0000		30.0000		
/tb_feedforward/w_D_feedforward1	0.500000		0.000000		0.500000
/tb_feedforward/w_D_feedforward2	0.375000		0.000000		0.375000

Figura 6.21 Testbench módulo de feedforward

### 6.2.7. Construcción del ciclo de trabajo

Por último se realiza un módulo que se encarga de realizar la suma de las acciones de control, de Feedback y Feedforward. Este módulo es concurrente al no necesitar del clock para realizar la cuenta. En la Figura 6.22 se observan los puertos del módulo teniendo como salida el ciclo de trabajo a aplicar.

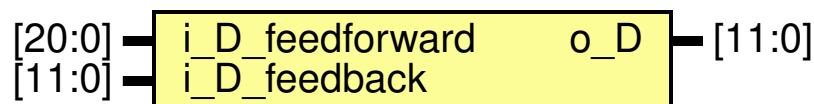


Figura 6.22 Diagrama de puertos de módulo constructor del ciclo de trabajo

## 6.3. Diseño de sistema de comunicación

El objetivo de la comunicación con la PC es realizar mediciones en tiempo real y realizar cambios de referencia. Para esto es necesario implementar un flujo continuo de información desde la FPGA con las mediciones, siendo posible, desde la PC identificarlas en el flujo de datos. Por otro lado la PC utiliza comandos para pedir las distintas mediciones, cambiar referencias y verificar las referencias cargadas anteriormente.

Para la comunicación se utilizó el protocolo UART (del inglés *Universal Asynchronous Receiver-Transmitter*) con una codificación KLV. En esta sección se desarrollan a través de maquinas de estados los módulos encargados de recibir, decodificar, codificar y transmitir la información. En la Figura 6.23 se observa el diagrama de conexión entre los módulos involucrados para realizar el sistema de comunicación completo.

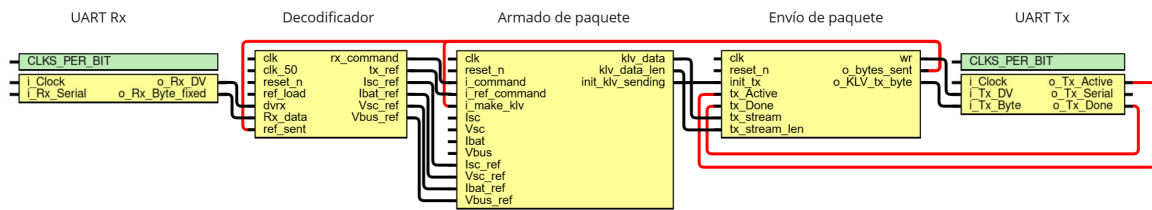


Figura 6.23 Diagrama de sistema de comunicación

### 6.3.1. Codificación KLV

El estándar de codificación de datos KLV (*Key-Length-Value*) es un método comúnmente utilizado para codificar información en flujos de vídeo. Este estándar se basa en un esquema de codificación tipo-longitud-valor (TLV). Los elementos se codifican en tripletes Key-Length-Value, donde la clave identifica los datos, la longitud especifica la longitud de los datos y el valor es la propia información.

La codificación KLV facilita el análisis de datos al proporcionar un formato estructurado y estandarizado para la transmisión de información. Esta capacidad de etiquetado y transmisión de datos mejora la interoperabilidad entre sistemas de medición y análisis, lo que resulta fundamental en aplicaciones que requieren una monitorización precisa y eficiente de parámetros eléctricos en tiempo real, como en este caso la interfaz diseñada. En este proyecto en particular, los elementos desempeñan los siguientes roles:

- **Clave:** Identifica el tipo de dato asociado, como tensión o corriente. Se ha asignado un tamaño de medio byte para la clave, considerado suficiente para identificar todas las tensiones y corrientes requeridas.
- **Longitud:** Indica la cantidad de bytes que ocupa el valor asociado. Proporciona la información necesaria para que el receptor del dato KLV pueda determinar el tamaño del valor y extraerlo correctamente. Se asigna un tamaño de medio byte, considerando que los valores asociados serán de dos, cuatro u ocho bytes de longitud.
- **Valor:** Contiene la información real asociada con la clave correspondiente. Si bien las variables se representan con 32 bits internamente, se propone utilizar las mediciones del ADC que son de 12 bits y luego realizar la transformación en la PC. Dependiendo de la clave identificada, estos valores serán de dos o cuatro bytes de longitud.

Para la identificación del inicio del paquete KLV se agrega una cabecera de un byte con valor en hexadecimal "FF". Una vez recibido el comando desde la PC y se interpreta que se deben mandar las mediciones, se arman y se envían los paquetes

constantemente con nueva información. El envío del valor de referencia se realiza una única vez.

En la siguiente tabla, representando los nibbles en hexadecimal, se muestran las codificaciones diseñadas para el proyecto:

Medición o Referencia	Encabezado	Clave	Longitud	Valor
Corriente SC	FF	1	2	0XXX
Corriente BL	FF	2	2	0XXX
Tensión SC	FF	3	2	0XXX
Tensión Bus	FF	4	2	0XXX
Corriente SC y Tensión Bus	FF	5	4	0XXX0XXX
Corriente BL y Tensión SC	FF	6	4	0XXX0XXX
Todas las mediciones	FF	7	8	0XXX0XXX0XXX0XXX
Corriente SC referencia	FF	C	4	XXXX
Corriente BL referencia	FF	D	4	XXXX
Tensión SC referencia	FF	E	4	XXXX
Tensión Bus referencia	FF	F	4	XXXX

Tabla 6.1 Codificación de paquetes KLV

### 6.3.2. Comunicación UART

La comunicación UART se trata de una comunicación serial que utiliza un cable para la transmisión y otro para la recepción. Toma bytes de datos y transmite los bits individuales de forma secuencial. En el destino, un segundo UART reensambla los bits en bytes completos.

Para que esta comunicación sea efectiva, ambos (receptor y transmisor) deben utilizar la misma velocidad de lectura de bits así como la misma cantidad de bits de datos.

El protocolo UART utiliza un bit de start para iniciar la comunicación, ocho bits de datos y un bit de stop. Esto es lo que permite el asincronismo.

Se utilizaron maquinas de estados de Mealy para realizar lo dos módulos de la comunicación UART, de recepción y transmisión.

#### Módulo de recepción UART

En la Figura 6.24 se presenta un diagrama con los puertos de entrada y salida con la cantidad de bits.

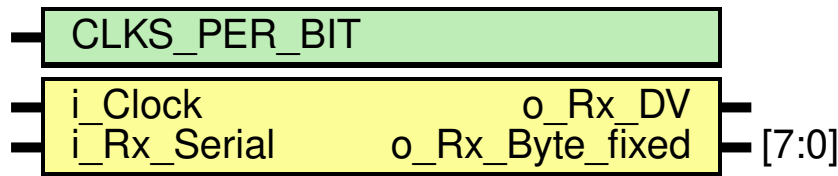


Figura 6.24 Diagrama de puertos de módulo receptor UART

Este módulo cuenta con el clock de  $50MHz$  del sistema. El parámetro clock por bit, como el nombre lo indica, es la cantidad de clocks de entrada que deja pasar para obtener un bit. Con esto es posible configurar la velocidad de transmisión en baudios (bps). Se determina utilizar  $921,6kbps$  siendo la velocidad máxima posible sin perder información desde la PC. Aplicando la siguiente formula, utilizando los baudios deseados, se puede calcular éste parámetro.

$$CLKS\_PER\_BIT = \frac{50MHz}{921,6kbps} = 54$$

Por último en la maquina de estados de la Figura 6.25 se observa el comportamiento del módulo. Una vez que se ingresa el bit de start, se levanta una flag de inicio de lectura y se verifica el estado a la mitad de los clock por bit para volver al estado de espera o iniciar la lectura de los bits de dato. Una vez en el estado de bits de datos se esperan los clock por bit para ir aumentando el índice de cantidad de bits leídos hasta alcanzar los ocho. Por ultimo se procede a leer el bit de fin de la comunicación, bajar la flag de lectura y sacar el byte completo leído.

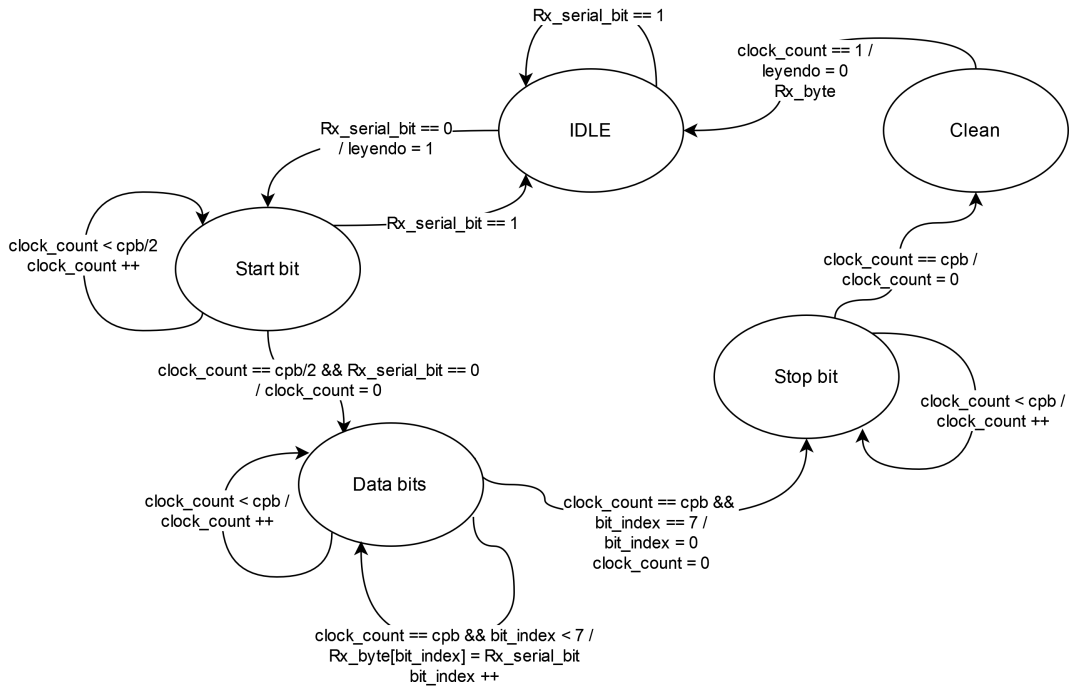


Figura 6.25 Maquina de estado de UART Rx

**Módulo de transmisión UART**

En la Figura 6.26 se presenta un diagrama con los puertos de entrada y salida con la cantidad de bits.

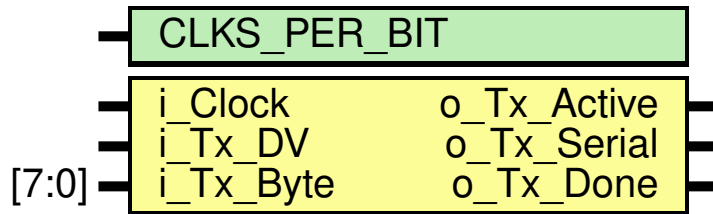


Figura 6.26 Diagrama de puertos de módulo transmisor UART

De forma análoga al módulo receptor, se ingresa el mismo parámetro de clock por bit y el clock del sistema. Cuenta con el byte que se desea transmitir a la entrada y una flag para iniciar la transmisión, así como dos flags, una para indicar que se esta transmitiendo un byte y otra para indicar que se terminó la transmisión, siendo útiles para el sistema de comunicación total.

Una vez se ingresa el bit de inicio de comunicación se procede con el envío del bit de inicio y se levanta la flag de transmisión activa. Luego se pasa al estado de transmisión de bits del byte, incrementando el índice cada vez que se alcanzan los clocks por bit. Por ultimo se manda el bit de stop y se baja la bandera de activo.

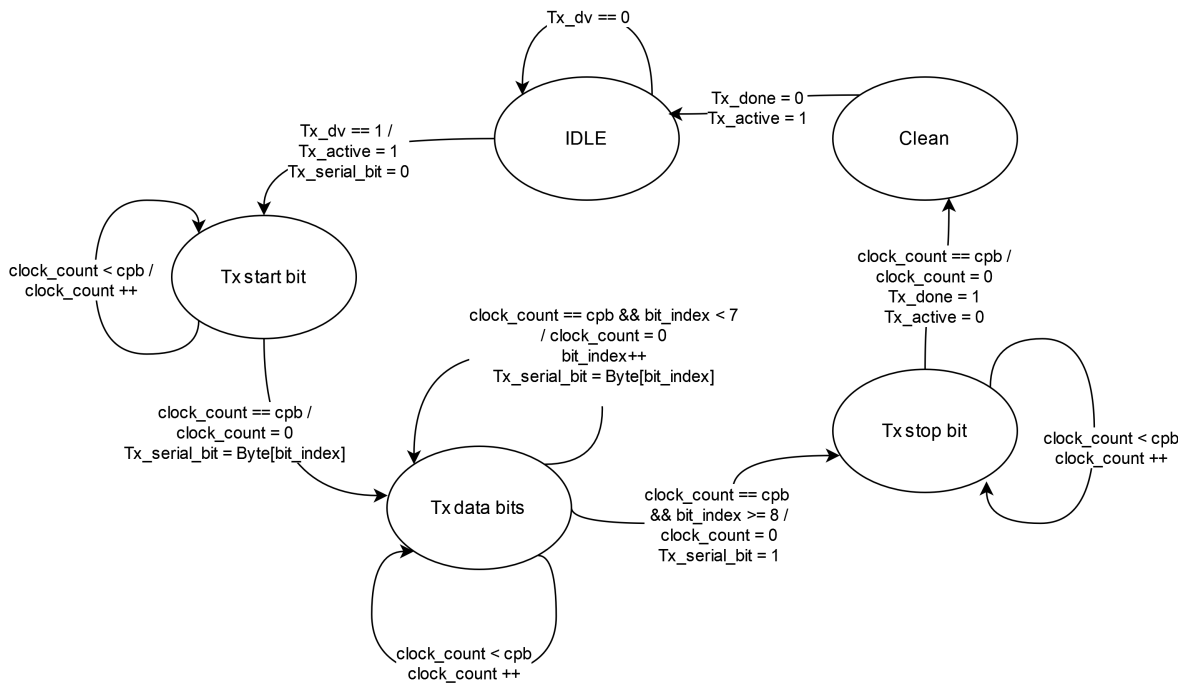


Figura 6.27 Máquina de estado de UART Tx

Por último se realizó un testbench que verifique el funcionamiento de ambos módulos. Se instanció un módulo UART Tx para enviar los bits a una instancia del módulo UART Rx y verificar la recepción del byte. En la Figura 6.28 se observa el inicio de la comunicación cuando el bit serial de entrada envía el bit *start*. Luego procede desde los bits menos significativos (*LSB*) y termina con el bit *stop* en alto. Una vez terminada la conversión, la señal *dvr<sub>x</sub>* se pone en bajo.

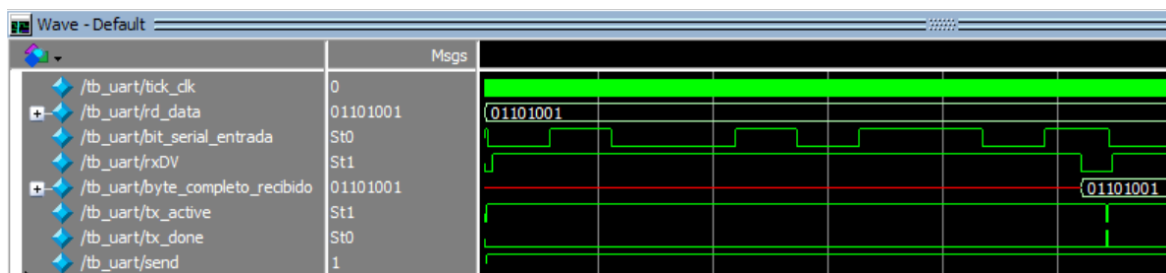


Figura 6.28 Testbench de módulos de transmisión y recepción de UART

### 6.3.3. Decodificador

Una vez recibidos los bytes, estos ingresan al módulo decodificador para decidir si se trata de un cambio de referencia, envío de la referencia en la FPGA o un pedido de lectura de alguna de las variables de tensión o corriente. Es por esto que en el diagrama de la Figura 6.29 de este módulo salen todas las referencias de la comunicación y el

comando identificado. Adicionalmente ingresan dos clocks, uno a la misma frecuencia UART para identificar los bytes cuando se reciben y otro a la frecuencia del control. La señal *ref\_load* se utiliza para aplicar el cambio de referencia una vez cargado el valor y utilizando el clock de control se implementa un limitador de pendiente en la tensión de bus de referencia para no realizar un cambio brusco.

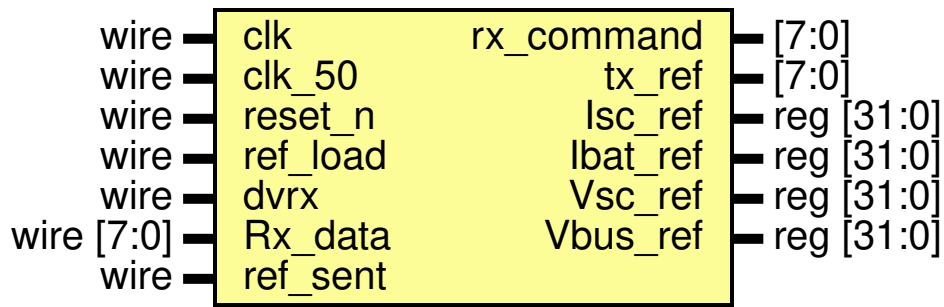


Figura 6.29 Diagrama de puertos de módulo decodificador

En la Figura 6.30 se observa la maquina de estados implementada utilizando el flanco de bajada de la flag del módulo UART receptor para identificar cuando se tiene un nuevo dato y se verifica que tipo de comando se envió desde la PC. Siendo que de 1 a 7 se envían paquetes KLV constantemente, del 8 al 11 se cargan referencias en la FPGA y del 12 al 15 se leen las referencias previamente cargadas.

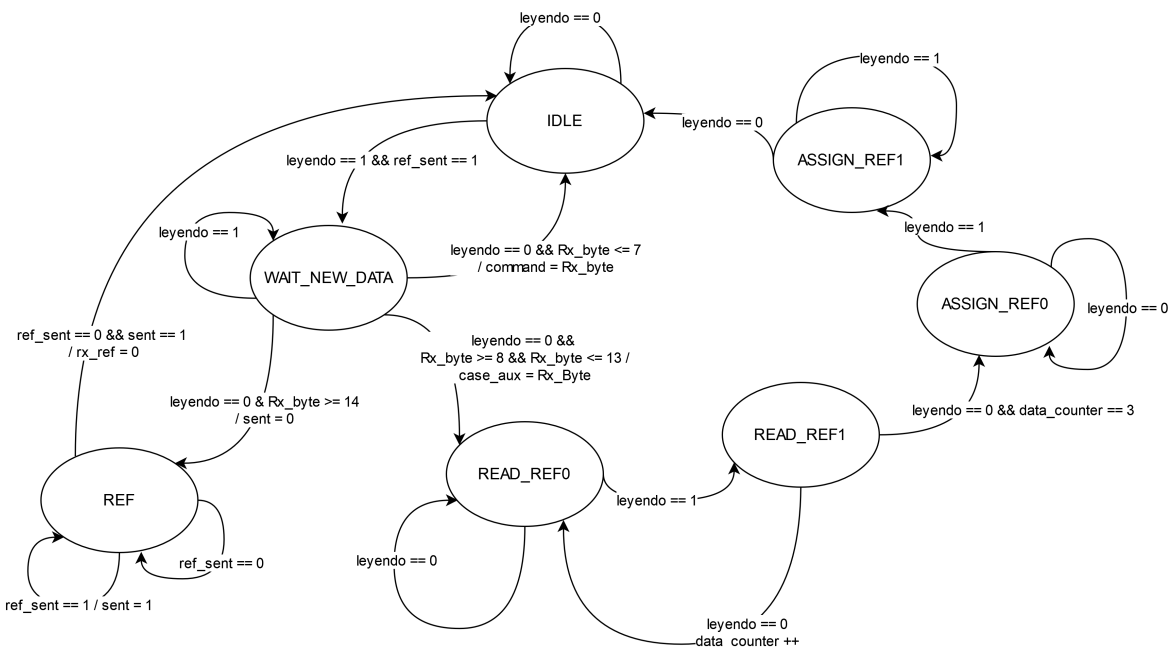


Figura 6.30 Maquina de estados de módulo decodificador

En la Figura 6.31 se realiza un testbench para verificar el funcionamiento. Se instancian los módulos UART para simular el envío de comandos y la recepción de los mismos para ser evaluados por el modulo decodificador. Se inicia con el envío del comando 1 para enviar datos obteniendo a la salida un 1 en *Rx\_command* y 0 en *tx\_ref* que serán utilizados por el modulo de armado de paquetes KLV. Luego se envia el comando 8 para cargar con los siguientes bytes el valor de la corriente de referencia de los supercapacitores. Cuando se realiza el el flanco con la señal SW se aplica el cambio de referencia.

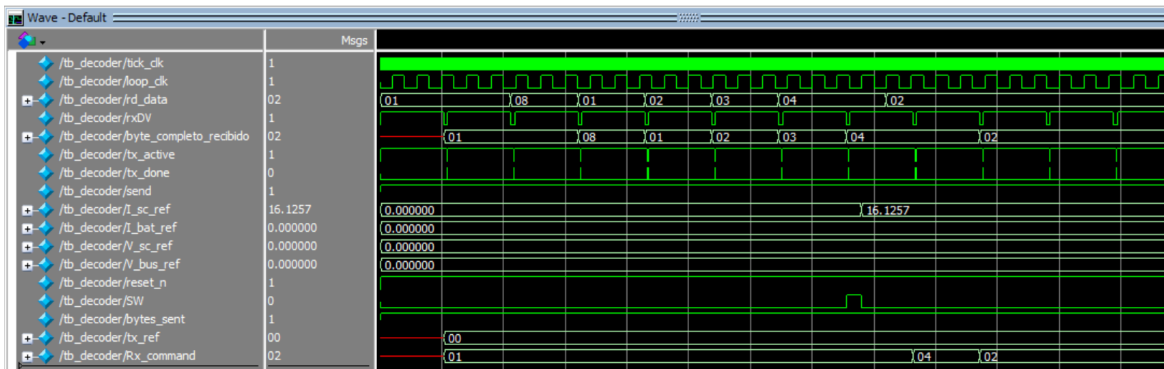


Figura 6.31 Testbench de modulo decodificador

Por ultimo se verifica la aplicación de la limitación de pendiente en el cambio de referencia de la tensión de bus en la Figura 6.32. Se puede observar que una vez se hace el flanco para aplicar el cambio de referencia, se esperan 50 ciclos del clock de control para realizar un paso de tensión hasta el valor cargado.

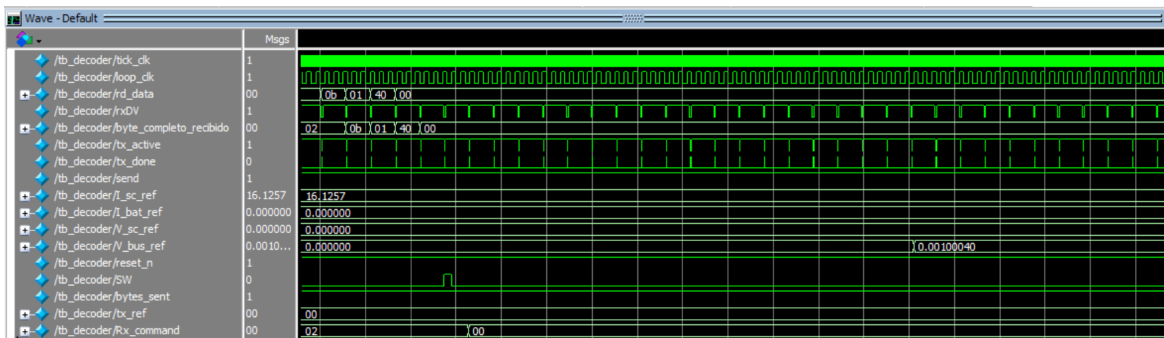


Figura 6.32 Testbench carga de referencia de tensión de bus

#### 6.3.4. Armado de paquetes KLV

Este módulo se encarga de recibir el comando del decodificador que pida lecturas de las variables o de referencias cargadas y codificarlas en paquetes KLV con cabecera, tipo de dato, largo y el dato en si para la transmisión. Es por estos que ingresan todas

las variables y referencias. En la Figura 6.33 se indican los puertos de entrada y salida, así como el tamaño en bits de los mismos.

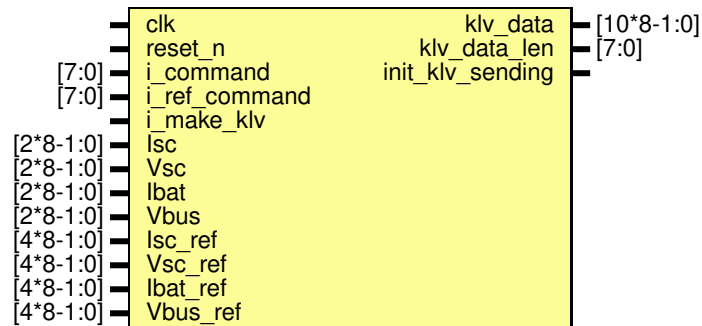


Figura 6.33 Diagrama de puertos de módulo de armado de paquetes

Una vez recibido el comando de envío del decodificador, de datos o de referencia, se espera la flag de armado del paquete para armar el paquete correspondiente. De salida se tiene el stream de datos con el largo máximo posible, una byte aparte indica el largo actual del paquete y se levanta una flag una vez armado el paquete para iniciar el envío del mismo. En la Figura 6.34 se observa la maquina de estados del módulo implementado.

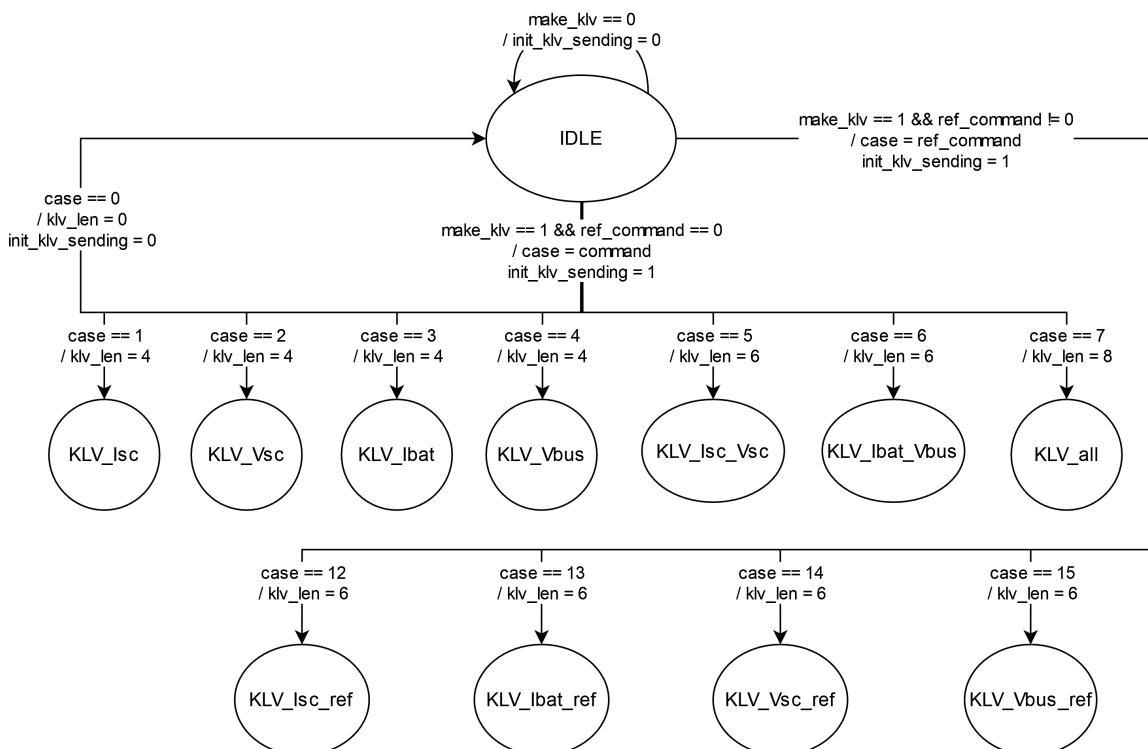


Figura 6.34 Maquina de estados de armado de paquetes

Se realiza un testbench para verificar el funcionamiento del modulo dependiendo del comando ingresado. En la Figura 6.35 se observa que el armado del paquete se realiza una vez la señal *bytes\_sent* esta en alto y el armado de los paquetes de referencia es coherente con la tabla 6.1. Por ultimo se envía el comando 1 para verificar el armado de paquete KVL de una medición.

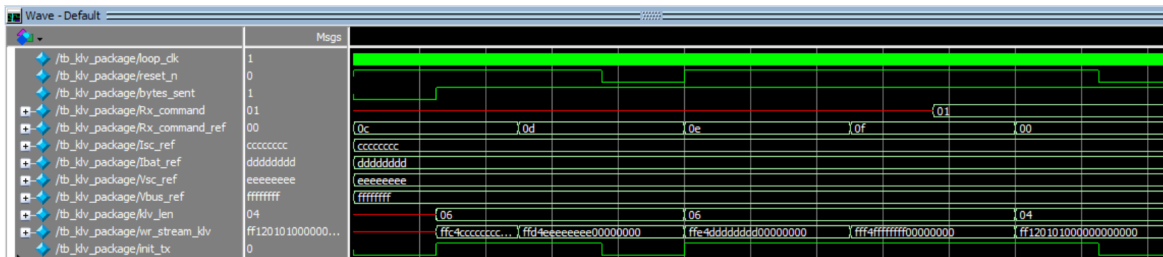


Figura 6.35 Testbench de modulo de armado de paquetes KLV

### 6.3.5. Envío de paquetes KLV

Ya que el módulo de transmisión UART puede mandar la información de a un byte, es necesario un módulo que reciba los paquetes KLV armados y pase byte a byte al paquete para la transmisión. Es por esto que se necesita una entrada con el largo maximo del paquete que se pueda enviar, una con el largo del paquete actual y las flags del transmisor UART para identificar cuando es necesario pasar otro byte del paquete. También cuenta con una flag de salida que permite saber cuando se termina de enviar el paquete completo.

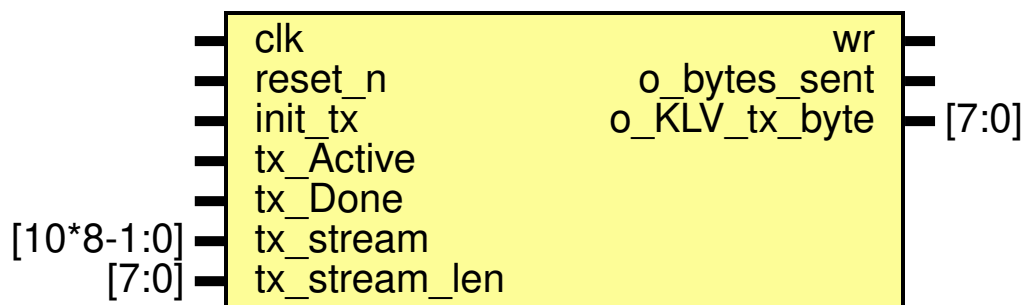


Figura 6.36 Diagrama de puertos del módulo de envío de paquetes

La maquina de estados de la Figura 6.37 demuestra el funcionamiento de manera simple. Una vez recibida la señal de envío de un nuevo paquete, la flag de bytes enviados se pone en cero. Luego se pasa el byte que corresponde y se espera que el módulo de transmisión termine de enviarlo para aumentar el índice del byte a enviar hasta alcanzar el tamaño en bytes del paquete ingresado. Luego se termina subiendo la flag de bytes enviados.

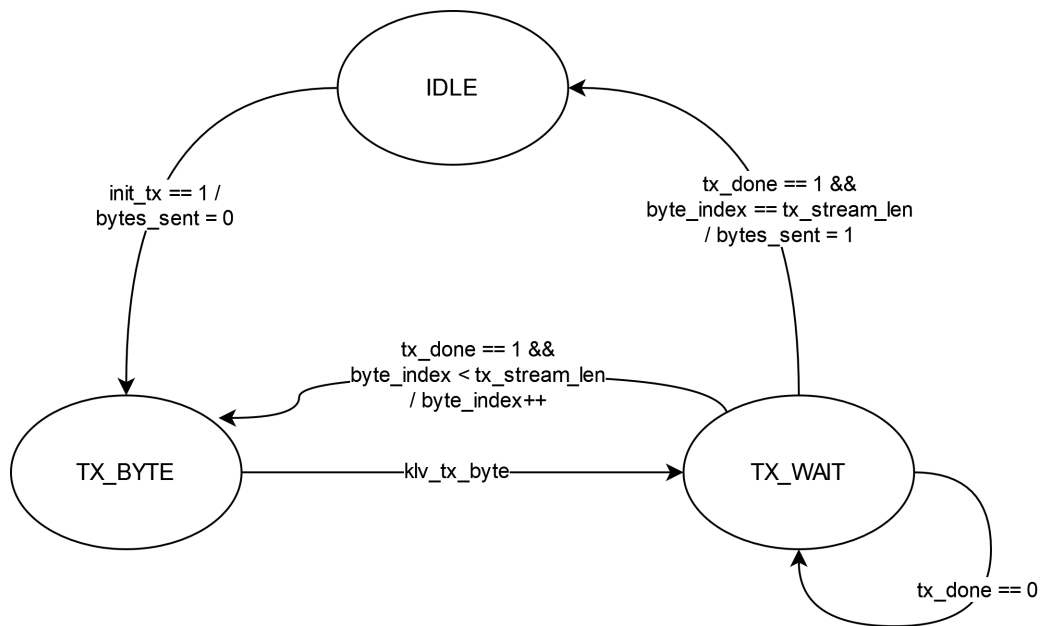


Figura 6.37 Máquina de estados de módulo de envío de paquetes

Se actualiza el testbench de armado de paquetes agregando una instancia del módulo de envío de paquetes y la transmisión de UART. Se verifica el armado de paquetes KLV de datos constante y el envío byte a byte del mismo a través del UART. Se simula el pedido de envío de referencia y se observa que se interrumpe el envío de KLV de datos para mandar un paquete KLV de referencia y volver al envío constante de datos.

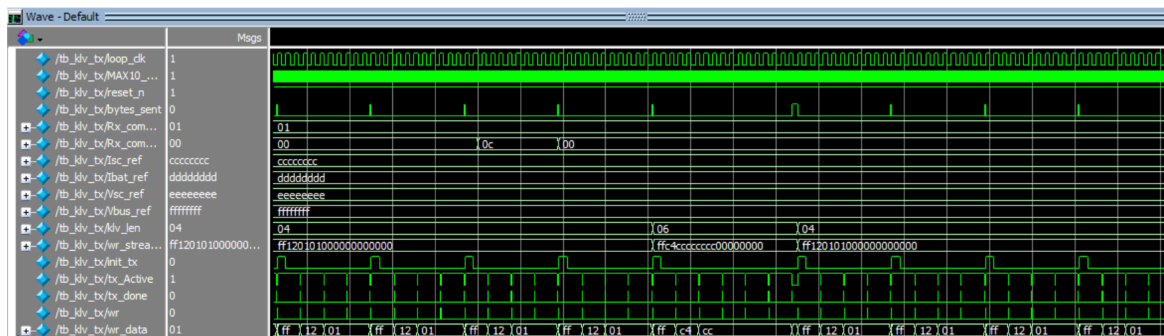


Figura 6.38 Testbench de modulo de envío de paquetes

## 6.4. Módulos auxiliares

Con fines prácticos, se definen distintos módulos para aprovechar los componentes de la FPGA y obtener una forma de verificación del funcionamiento. Se utilizan los display de siete segmentos, las llaves y los leds disponibles.

## Llaves

Las llaves se utilizan a modo de control. Su función será la de habilitar y deshabilitar los PI de tensión, feedforward, pwm1 y pwm2 y seleccionar las distintas variables para mostrar en los display de siete segmentos. En la siguiente Tabla se detallan las funciones de cada llave:

Numero de llave	Descripción
Llave 0	Selección entre mediciones y referencias
Llave 1	Selección entre corriente y tensión
Llave 2	Selección del convertidor de la variable
Llave 3	Habilitación PI de tensión bus
Llave 4	Habilitación de PI de regulación de tensión de SC
Llave 5	Habilitación de feedforwards
Llave 6	Habilitación de salidas PWM1
Llave 7	Habilitación de salidas PWM2
Llave 8	Habilitación de la comunicación
Llave 9	Habilitación de GS

Tabla 6.2 Asignación de funcionalidades a las llaves

## Display de siete segmentos

Se cuenta con seis display de siete segmentos, lo que permite utilizarlos para ver las mediciones y referencias en la FPGA. Se utiliza uno para el signo, dos de entero, dos de fracción y uno para la unidad de medición.

En la figura 6.39 se observa el sistema que permite seleccionar la variable a visualizar con el módulo selector, luego utiliza el siguiente módulo para obtener el módulo y signo de la variable y por último se instancia seis veces el módulo para transformar el dígito en un número con el siete segmentos.

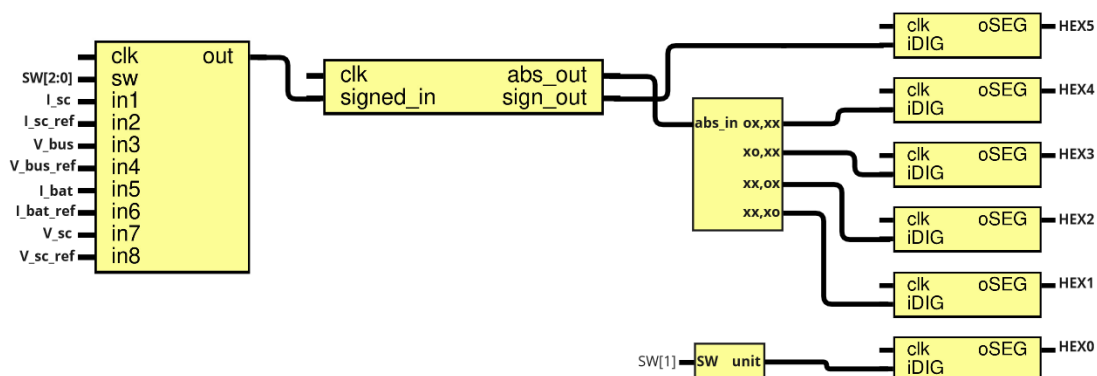


Figura 6.39 Módulos auxiliares

## 6.5. Análisis de Resultados de Síntesis y Timing en Quartus Prime

Por último se analizan los resultados del análisis y la síntesis de la implementación final realizadas por el software Quartus Prime.

Una parte crucial del proceso de compilación en Quartus Prime es el análisis de timing, que evalúa la temporización del diseño y garantiza que todas las restricciones temporales sean cumplidas. En este sentido, se examina el reporte de timing generado por la herramienta, prestando especial atención al peor slack, que indica la distancia entre la ruta de peor temporización y la restricción de timing asociada. Un slack positivo indica que la restricción se cumple, mientras que un slack negativo indica que la restricción no se cumple y se requiere una optimización adicional. En la Figura 6.40 se observa el reporte de timing con el peor slack siendo positivo, lo que sugiere que el diseño cumple con todas las restricciones temporales establecidas.

Multicorner Timing Analysis Summary						
<<Filter>>						
	Clock	Setup	Hold	Recovery	Removal	Minimum Pulse Width
1	▼ Worst-case Slack	5.032	0.148	8.920	0.477	5.809
1	ADC_CLK_10	N/A	N/A	N/A	N/A	96.000
2	MAX10_CLK1_50	12.024	0.148	N/A	N/A	9.327
3	MAX10_CLK2_50	N/A	N/A	N/A	N/A	16.000
4	PWM_CLK_inst altpll_comp...to_generated pll1 clk[0]	8.989	0.312	N/A	N/A	5.809
5	PWM_CLK_inst altpll_comp...to_generated pll1 clk[1]	5.032	0.214	8.920	4.491	1999.593
6	u0 altpll_sys sd1 pll7 clk[0]	33.865	0.206	37.859	0.477	19.709
7	u0 altpll_sys sd1 pll7 clk[1]	N/A	N/A	N/A	N/A	44.575
2	▼ Design-wide TNS	0.0	0.0	0.0	0.0	0.0
1	ADC_CLK_10	N/A	N/A	N/A	N/A	0.000
2	MAX10_CLK1_50	0.000	0.000	N/A	N/A	0.000
3	MAX10_CLK2_50	N/A	N/A	N/A	N/A	0.000
4	PWM_CLK_inst altpll_comp...to_generated pll1 clk[0]	0.000	0.000	N/A	N/A	0.000
5	PWM_CLK_inst altpll_comp...to_generated pll1 clk[1]	0.000	0.000	0.000	0.000	0.000
6	u0 altpll_sys sd1 pll7 clk[0]	0.000	0.000	0.000	0.000	0.000
7	u0 altpll_sys sd1 pll7 clk[1]	N/A	N/A	N/A	N/A	0.000

Figura 6.40 Reporte de timing

Además del análisis de timing, se revisan los recursos de la FPGA utilizados por el diseño. Esto incluye el número y tipo de elementos lógicos (LUTs), registros, bloques de memoria, y otros recursos específicos de la FPGA. Entender cómo se distribuyen los recursos en la FPGA es esencial para optimizar el diseño y maximizar el rendimiento del sistema. En la Figura 6.41 se observa el reporte de los recursos de la FPGA utilizados, donde se concluye que el diseño hace un uso eficiente de los recursos disponibles ya que se utilizan 18 % de los elementos lógicos, obteniendo la posibilidad de profundizar en la complejidad del diseño en futuros desarrollos.

Flow Summary	
<input type="text" value="Filter"/>	
Flow Status	Successful - Sun Mar 17 21:18:27 2024
Quartus Prime Version	23.1std.0 Build 991 11/28/2023 SC Lite Edition
Revision Name	hss_vehicular
Top-level Entity Name	hss_vehicular
Family	MAX 10
Device	10M50DAF484C7G
Timing Models	Final
Total logic elements	8,825 / 49,760 ( 18 % )
Total registers	1100
Total pins	90 / 360 ( 25 % )
Total virtual pins	0
Total memory bits	0 / 1,677,312 ( 0 % )
Embedded Multiplier 9-bit elements	186 / 288 ( 65 % )
Total PLLs	2 / 4 ( 50 % )
UFM blocks	0 / 1 ( 0 % )
ADC blocks	1 / 2 ( 50 % )

Figura 6.41 Reporte de recursos utilizados

## 6.6. Resumen

Se realizó el desarrollo del diseño digital a partir de un nivel de abstracción alto previamente trabajado en el Capítulo 5 en Simulink. Avanzando de forma modular se explicaron individualmente cada uno de los módulos que conforman el sistema de control con la validez de la simulación de los testbenchs a través de Modelsim.

Se definió el protocolo de comunicación a utilizar con la PC y se aproximó al problema dividiendo las funcionalidades en distintos módulos. Cada uno de estos fueron resueltos con maquinas de estado individualmente para luego integrarlos en un único sistema.

Se aprovecharon distintos recursos disponibles, tales como las llaves para habilitar distintas partes del control de forma progresiva en la etapa experimental. También se utilizaron los display de siete segmentos para ver las tensiones y corrientes de los convertidores.

Por último se presenta un análisis de los resultados de síntesis y timing obtenidos mediante la herramienta de compilación Quartus Prime. Se destaca el cumplimiento de las restricciones temporales, evidenciado por el peor slack positivo en el reporte de timing. Además, se concluye que el diseño hace un uso eficiente de los recursos de la FPGA, aprovechando solo el 18% de los elementos lógicos disponibles. Estos hallazgos sugieren la posibilidad de profundizar en la complejidad del diseño para futuras mejoras y optimizaciones.

## Capítulo 7

# Interfaz de usuario

### 7.1. Introducción

En este Capítulo se comentará en detalle el diseño y la implementación de la interfaz GUI (*Graphical User Interface*) desarrollada para el proyecto. Esta interfaz permite la visualización de datos en tiempo real como así también, la adquisición de los mismos a partir de archivos en formato CSV. Además, cuenta con la posibilidad de ajustar los valores de referencia de las corrientes y tensiones de ambos convertidores. Esta interfaz se desarrolló con el objetivo de generar una plataforma intuitiva y amigable para interactuar con el usuario, permitiendo a este último controlar y monitorear el funcionamiento de manera sencilla.

En este contexto, Python y PyQt6 han sido seleccionados como las herramientas principales para el desarrollo de la interfaz gráfica. Python es un lenguaje de programación, multiparadigma y multinivel, con soporte en programación orientada a objetos, imperativa y funcional. Se emplea esta plataforma ya que es un lenguaje de programación de código abierto, de propósitos generales y gratuito. Cuenta con una vasta comunidad de desarrolladores y una abundante cantidad de recursos de documentación, lo que simplifica la resolución de problemas y el aprendizaje del uso de PyQt6 y Qt Designer. Por otra parte, PyQt6 es una biblioteca que proporciona enlaces Python para la biblioteca Qt, permitiendo el desarrollo de aplicaciones GUI multiplataforma de manera sencilla.

Es por ello que, a lo largo de este capítulo se presentarán los conceptos básicos de PyQt6 y elementos básicos como widgets, layouts, señales y ranuras, como así también, las características que presenta dicha interfaz incluyendo la obtención de datos y la visualización mediante un software dedicado. Además, se abordarán aspectos importantes como la optimización en el guardado de archivos de datos y en el envío de los mismos.

## 7.2. PyQt6

PyQt6 es una biblioteca de Python que proporciona enlaces para la biblioteca Qt, permitiendo el desarrollo de aplicaciones gráficas multiplataforma. En esta Sección, se explorarán los conceptos básicos de PyQt6, incluidos widgets, layouts y señales, así como ejemplos prácticos para crear y manejar elementos de interfaz gráfica comunes.

### Widgets y Layouts

El módulo Qt Widgets proporciona un conjunto de elementos de interfaz de usuario (UI) para crear una interfaz clásica de estilo escritorio. Estos Widgets constituyen los componentes fundamentales para la construcción de interfaces en Qt, capaces de mostrar datos, información de estado, recibir entrada del usuario y servir como contenedor para otros Widgets que deben agruparse juntos. Estos representan componentes visuales como botones, etiquetas, cuadros de texto, entre otros. Esta biblioteca ofrece una amplia variedad de Widgets predefinidos que pueden ser utilizados para construir interfaces gráficas complejas.

Los Layouts son estructuras que definen la disposición y organización de los Widgets dentro de una ventana o un contenedor. PyQt6 proporciona varios tipos de Layouts, como QVBoxLayout, QHBoxLayout y GridLayout, que permiten organizar los Widgets de una manera más fácil.

### Señales y ranuras

En la programación de interfaces gráficas de usuario (GUI), es común la necesidad de que un Widget notifique a otro cuando ocurre un cambio. En este contexto, el uso de Callbacks ha sido una técnica popular para lograr esta comunicación. Sin embargo, los Callbacks pueden ser difíciles de manejar y pueden presentar problemas de tipado.

En PyQt6, se cuenta con una alternativa más intuitiva y segura: señales y ranuras. Una señal se emite cuando ocurre un evento específico, y una ranura es una función que se llama en respuesta a esa señal. Esto proporciona una forma más estructurada y segura de comunicar eventos entre objetos en una GUI. La combinación de señales y ranuras en PyQt6 ofrece varios beneficios:

- **Seguridad de Tipos:** Las señales y las ranuras en PyQt6 están diseñadas para garantizar la compatibilidad de tipos, lo que ayuda a evitar errores de tipado durante la compilación.
- **Acoplamiento Flexible:** Los objetos que emiten señales no necesitan saber quién o qué está recibiendo esas señales, lo que permite un acoplamiento flexible y un mejor encapsulado de la información.

- Escalabilidad: Es posible conectar múltiples señales a una única ranura, así como conectar una señal a múltiples ranuras, lo que proporciona una gran flexibilidad y escalabilidad en el diseño de la GUI.

Las señales se emiten por un objeto cuando su estado interno ha cambiado de alguna manera que podría ser interesante para el cliente u propietario del objeto. Cuando se emite una señal, los slots conectados a ella generalmente se ejecutan de inmediato, al igual que una llamada de función normal. En este momento, el mecanismo de señales y ranuras es totalmente independiente de cualquier ciclo de eventos de la GUI.

### Qt Designer

Qt Designer es una aplicación visual que forma parte del conjunto de herramientas proporcionadas por Qt para el desarrollo de interfaces gráficas de usuario (GUI) de manera rápida y sencilla. Esta herramienta permite diseñar interfaces gráficas de forma visual, arrastrando y soltando Widgets en una ventana de diseño, lo que facilita enormemente el proceso de creación de interfaces complejas.

Se integra perfectamente con PyQt6, lo que permite diseñar interfaces gráficas utilizando Qt Designer y luego integrarlas en aplicaciones PyQt6. Esto simplifica el proceso de desarrollo y permite una mayor separación entre el diseño de la interfaz y la lógica de la aplicación.

Específicamente para este proyecto, combinando Widgets y Layouts con una programación de señales mediante la aplicación de Qt Designer, se realizó la siguiente interfaz gráfica:



Figura 7.1 Interfaz diseñada para este proyecto.

Como se observa en la Figura 7.1, la interfaz en cuestión habilita al usuario para llevar a cabo diversas configuraciones relacionadas con la comunicación y adquisición de datos. En primer lugar, brinda la posibilidad de seleccionar el puerto de comunicación y los baudios necesarios para establecer una conexión eficiente con la FPGA. Esta capacidad de elección asegura la adaptabilidad del software a diferentes configuraciones de hardware.

Adicionalmente, la interfaz ofrece la opción de decidir entre la obtención de datos en tiempo real o la importación de datos desde archivos en formato CSV. La elección de esta última alternativa permite al usuario establecer parámetros específicos para la activación del trigger. Esta configuración abarca la determinación del valor umbral que desencadenará el trigger, la elección del flanco (ya sea de subida o bajada), la selección de la señal a medir y la posibilidad de especificar el tiempo requerido para la operación. Estas opciones proporcionan un alto grado de control sobre el proceso de adquisición de datos y aseguran que se ajuste a las necesidades y requisitos del usuario de manera precisa y eficiente. A lo largo de este Capítulo se detallará como fueron implementadas estas opciones que proporciona la interfaz.

### 7.2.1. Decodificación KLV

Como se presentó en el Capítulo 6, los datos son proporcionados por el FPGA en formato de codificación KLV (*Key-Length-Value*). La interfaz diseñada en Python, utilizando PyQt6, se encargará de recibir y decodificar estos datos extrayendo la información de interés, como la tensión o la corriente, según indique la clave proporcionada. Por este motivo, es fundamental contar con un proceso eficiente de recepción y decodificación para extraer y utilizar la información de manera efectiva.

La decodificación KLV se realiza en el programa de Python. Se heredan las propiedades y métodos de la clase que permite la comunicación serial "PySerial" y se implementa el nuevo método para decodificar los datos KLV. Su función principal es identificar el encabezado del dato para extraer la clave, la longitud y el valor. Considerando que los datos se transmiten en bytes, se analizan los valores en su representación hexadecimal. La lectura del dato se realiza leyendo ventanas de a dos bytes como se ilustra en la Figura 7.2, en este proceso verifica la presencia de un dato válido cuando este sea **FF00**<Dato<**FFFF**. Esto es realizable gracias al encabezado agregado en el envío de datos que permite distinguir la parte inicial del dato y facilitar su correcta recepción.

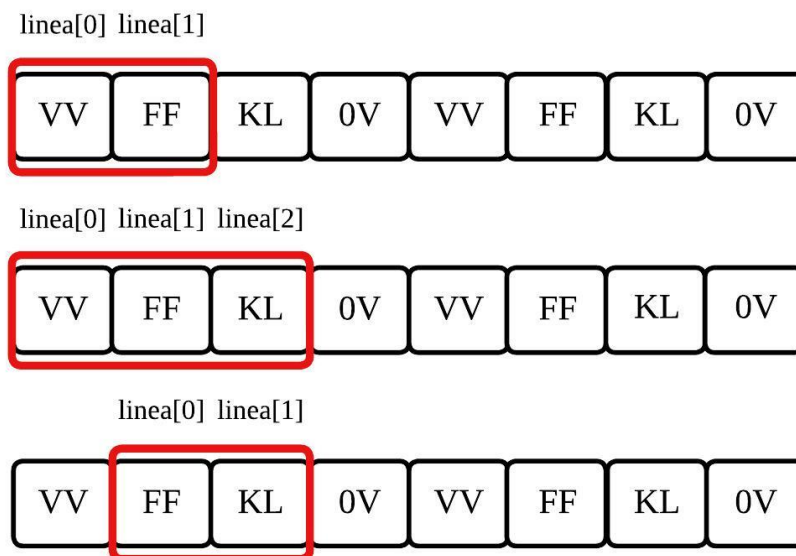


Figura 7.2 Decodificación de datos KLV.

De esta manera, el modulo va desplazando la ventana de a un byte y descartando el byte anterior como se observa en la Figura anterior hasta encontrar el encabezado del dato. Para dar fiabilidad de que con este método no se perderán datos, se realiza un análisis donde se ilustran los posibles eventos que pueden ocurrir durante la recepción y decodificación de los datos KLV. Cabe destacar que el valor de una medición, ya sea de corriente o tensión, puede ser representado con 24 bits, es por ello que el primer medio byte siempre será 0. Con lo cuál, para un ejemplo donde el valor es de dos bytes, se tendrá la siguiente representación:

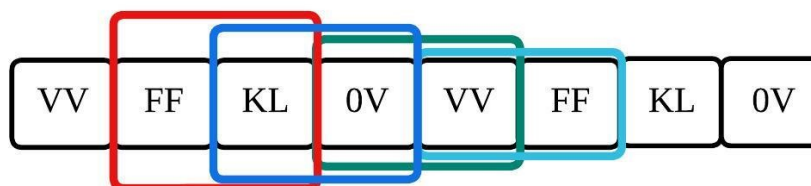


Figura 7.3 Análisis decodificación de datos KLV

De igual modo, este análisis es aplicable cuando el valor posee un tamaño de cuatro bytes.

Mediante la siguiente tabla se estudian los posibles valores que podrían adoptar la *key*, *length*, *value* y las condiciones que no cumplen para la detección del encabezado:

Cuadro	Key	Length	Value	Condición
◦	<= F	< F	-	✓
◦	<= F	< F	-	< FF00
◦	-	-	-	< FF00
◦	-	-	= FF	>= FFFF
◦	-	-	< FF	< FF00

Tabla 7.1 Representación de los posibles eventos que pueden ocurrir

De esta última tabla se puede concluir que el método implementado resulta efectivo para la decodificación KLV asegurando la obtención de todos los datos medidos y transmitidos por el FPGA.

### 7.3. Almacenamiento de referencias por interfaz

La interfaz permite mostrar que valores de referencia se encuentran actualmente almacenados. Cuando se selecciona el valor de referencia que se quiere conocer, la interfaz establece comunicación a través de UART con el número correspondiente (Ver tabla 7.2), recibiendo la respuesta por parte de la FPGA y mostrando el valor en la interfaz.

En caso de que se desee modificar el valor, la interfaz envía primero el valor de almacenamiento representativo de la variable a referenciar (Ver tabla 7.2) a la FPGA, seguido del valor establecido en la interfaz.

Referencia	Almacenar	Lectura
Corriente SC	8	C
Corriente BL	9	D
Tensión SC	A	E
Tensión Bus	B	F

Tabla 7.2 Key de Almacenamiento/Lectura especificadas para las referencias.

### 7.4. PlotJuggler

PlotJuggler es un programa que ofrece una interfaz intuitiva y fácil de usar que permite a los usuarios visualizar datos de manera dinámica y en tiempo real. Esta herramienta es especialmente útil para la representación gráfica de series temporales, lo

que la hace ideal para aplicaciones que requieren monitoreo continuo de datos, como sistemas de control, adquisición de datos y análisis de señales. La principal utilidad de PlotJuggler radica en su capacidad tanto para trazar gráficas en tiempo real como para graficar datos a partir de archivos de registro.

#### 7.4.1. Lectura en tiempo real

Para emplear este programa en tiempo real se utiliza, como fuente de datos, el Protocolo de Datagramas de Usuario (UDP). Es un protocolo de nivel de transporte que funciona entre la capa de red y la capa de aplicación en el modelo OSI. UDP permite el envío rápido de datagramas en redes IP sin la necesidad de establecer una conexión previa, ya que la información necesaria sobre el destinatario se incorpora directamente en la cabecera del datagrama. A diferencia de TCP, UDP carece de confirmación de entrega o recepción. Sin embargo, esto no representa un problema significativo debido a la codificación de los datos, lo que asegura que el encabezado siempre sea encontrado priorizando así la velocidad de visualización de los datos.

Los datos se encapsulan en formato *MessagePack*. Es un formato eficiente de serialización binaria que permite el intercambio de datos entre múltiples lenguajes, similar a *JSON*. Sin embargo, es más rápido y compacto. Los enteros pequeños se codifican en un solo byte, y las cadenas cortas típicas solo requieren un byte adicional además de las propias cadenas. En la siguiente figura, se muestra un ejemplo particular del formato *MessagePack* en comparación al formato *JSON*, considerando que se envía un dato de la tensión de salida del convertidor de un valor de 48V, en un tiempo específico de 1s:

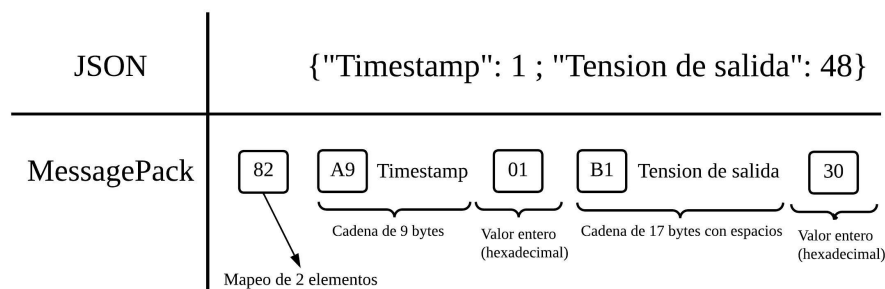


Figura 7.4 Comparación formato *JSON* con *MessagePack*

Una vez iniciado el proceso de lectura del dato, encapsulado en *MessagePack* y envío para plotear, no existe control de parte del usuario para terminarlo. Por esto surge la necesidad de realizar dicho proceso a través de un hilo (del inglés *thread*), de la clase *QThread*, brindando la posibilidad al proceso principal de enviar una señal para terminar la comunicación con PlotJuggler.

### 7.4.2. Lectura CSV

Las siglas CSV vienen del inglés "*Comma Separated Values*" y significan valores separados por comas. Dicho esto, un archivo CSV es cualquier archivo de texto en el cual los caracteres están separados por comas, haciendo una especie de tabla en filas y columnas.

Como se mencionó anteriormente, una de las funciones proporcionadas por la interfaz es la capacidad de generar un archivo CSV con el nombre de la fecha actual, incluyendo el tiempo y el valor del "trigger" (que se explicará en la próxima Sección), en el cuál rellena los nombres de las columnas con el tiempo y el valor de la variable dependiendo de la *key* del dato decodificado.

Adicionalmente, se utiliza una función que permite editar el CSV recién creado. Esto posibilita cambiar los nombres de las curvas según la variable medida que se requiera, facilitando así la apertura de Plotjuggler con las curvas graficadas correctamente etiquetadas.

#### Trigger

La interfaz además incluye una sección dedicada al *trigger* que permite configurar el umbral a partir del cuál se comenzarán a almacenar datos para el archivo CSV. Esta función se agregó con el objetivo de que el usuario pueda almacenar datos, por ejemplo, a partir del momento en el que el sistema alcanza su punto de trabajo deseado, descartando así el período transitorio o datos irrelevantes que no se deseen visualizar.

En la implementación del trigger, los valores se almacenan constantemente, y dependiendo si se elige el flanco de subida o bajada, el trigger se "armará" cuando la medición seleccionada (tensión y/o corriente) es menor o mayor, respectivamente, al valor umbral establecido.

Para garantizar una lectura eficiente de las mediciones, se ha implementado un delta de cambio de 0,05 tanto para la tensión como para la corriente. Esto permite actualizar los valores solo cuando hay una modificación significativa. Es decir, el valor del trigger almacenado se actualiza únicamente si la diferencia entre el valor actual y el anterior es mayor al delta de cambio. De esta manera, se evita la actualización de los valores si no han cambiado considerablemente.

Una vez que el trigger se encuentra armado, los datos se guardarán en el archivo CSV, siempre y cuando se cumplan ciertas condiciones. Por ejemplo, en el caso del flanco de subida, se verificará que el delta de cambio sea siempre positivo y que el último dato almacenado en el trigger sea mayor que el umbral establecido en la interfaz; y viceversa para el caso del flanco de bajada.

Para una comprensión más clara, se presenta la siguiente figura ilustrativa que muestra un ejemplo cuando se selecciona el flanco de subida en la interfaz:

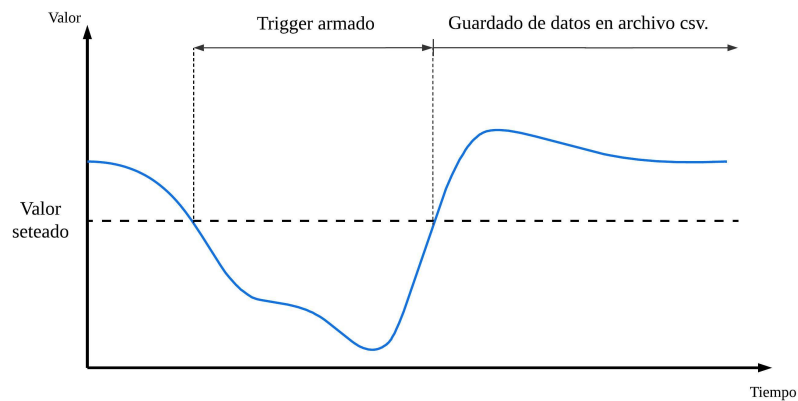


Figura 7.5 Ejemplo del Trigger implementado para un flanco de subida.

### 7.4.3. Optimización en envío de datos

En un principio se optó por utilizar la representación decimal, donde cada dato obtenido del ADC consta de 12 bits, lo que permite representar un rango entre 0 y 4095. Bajo esta representación decimal, el valor máximo (4095) se compone de cuatro caracteres (4 bytes).

Al implementar una representación hexadecimal, el valor máximo del rango puede representarse con **FFF**, lo que resulta en un ahorro de un byte en el envío de datos. Expresando en términos de porcentaje, esto representa una mejora de 25 %.

## 7.5. Resumen

En este Capítulo se presentó el diseño de la interfaz gráfica de usuario (GUI) utilizada para el proyecto, creada para permitir la decodificación y visualización de datos en tiempo real, como así también en archivos CSV. Además se explicaron las opciones que brinda la misma, desde aspectos como el trigger hasta la carga de valores de referencia. En donde, Python y PyQt6 se destacan como herramientas principales para el desarrollo de esta interfaz.

En la sección de PyQt6 se exploran los conceptos básicos de esta biblioteca, incluyendo widgets, layouts, señales y ranuras. Por otra parte, Qt Designer también se introduce como una herramienta importante para el diseño visual de interfaces gráficas, permitiendo una creación rápida y sencilla de la interfaz.

Se aborda la decodificación KLV para la recepción y procesamiento de datos provenientes de una FPGA, así como la carga de referencias mediante la interfaz y la optimización en el envío de datos mediante la representación hexadecimal.

La sección sobre PlotJuggler explora su utilidad para la visualización dinámica de datos en tiempo real y la lectura de archivos CSV, explicando cómo integrarlo con la interfaz desarrollada.

En el próximo Capítulo se comentarán los ensayos experimentales llevados a cabo para comprobar el correcto funcionamiento de la etapa de diseño digital.

## Capítulo 8

# Ensayos y resultados experimentales

### 8.1. Introducción

Para asegurar la eficacia y fiabilidad de la etapa de diseño digital, se ha realizado una serie de ensayos experimentales progresivos. En una primera fase, se sometió el convertidor a una prueba a lazo abierto, centrándose específicamente en la verificación del correcto funcionamiento del convertidor elevador.

Como segunda fase, se procedió a evaluar exclusivamente el lazo de corriente el cuál fue implementado con un controlador PI previamente sintonizado, como se detalló en secciones anteriores. Este análisis permitió verificar el desempeño y la estabilidad del sistema en relación con el control de corriente.

Como tercera y última fase, se realizó la evaluación del lazo de tensión externo, donde inicialmente se utilizó el controlador PI sintonizado en la región I, lo que permitió obtener una base de referencia sólida para así finalmente aplicar la técnica de *Gain Schedule* posibilitando mejorar la eficiencia del control de tensión respecto del PI sintonizado.

Es importante destacar que, aunque el enfoque del proyecto se centra en el uso de baterías de litio, todos los ensayos se realizaron utilizando una fuente de potencia que emula el comportamiento de las baterías. Esta decisión se tomó por consideraciones de seguridad tanto para la electrónica utilizada como para los usuarios. Dado que la estación de ensayo experimental aún no cuenta con los elementos de seguridad necesarios para trabajar con baterías, la fuente de potencia ya incorpora algunas medidas de seguridad, como la limitación de corriente.

Por otra parte, la disponibilidad de una interfaz que permite cargar tanto las referencias de corriente como de tensión, y posibilita la visualización de las mediciones requeridas, simplifica enormemente la realización de estos ensayos.

## 8.2. Rack de pruebas

Para realizar las pruebas experimentales en un entorno controlado y seguro, se ha construido un rack. Su función principal es proporcionar una plataforma organizada y centralizada para conectar los distintos módulos, contribuyendo así, la realización de pruebas bajo condiciones controladas y reproducibles.

En la Figura 8.1 se muestra un esquemático de la disposición y las conexiones realizadas en el rack. En la parte superior del rack se encuentra ubicada la carga electrónica, seguida de los convertidores utilizados y, por último, los módulos de almacenamiento.

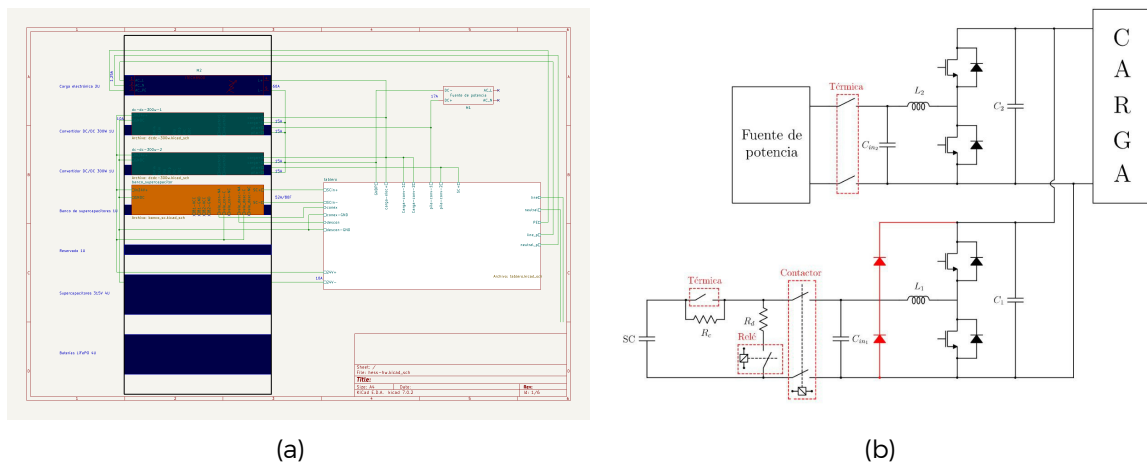


Figura 8.1 Esquemático de conexiones de rack de prueba.

Se han incorporado elementos de seguridad para llevar a cabo de manera controlada el proceso de carga y descarga de los supercapacitores. En la Figura 8.2 se presenta la disposición del sistema eléctrico híbrido ensayado en el instituto. Los elementos presentes en este ensayo son:

1. Carga electrónica.
2. Convertidor CC-CC conectado a fuente de potencia.
3. Convertidor CC-CC conectado a banco de supercapacitores.
4. Modulo de seguridad.
5. Banco de supercapacitores.
6. Baterías de litio.
7. Fuente de potencia
8. Alimentación del rack y bornes de conexión.

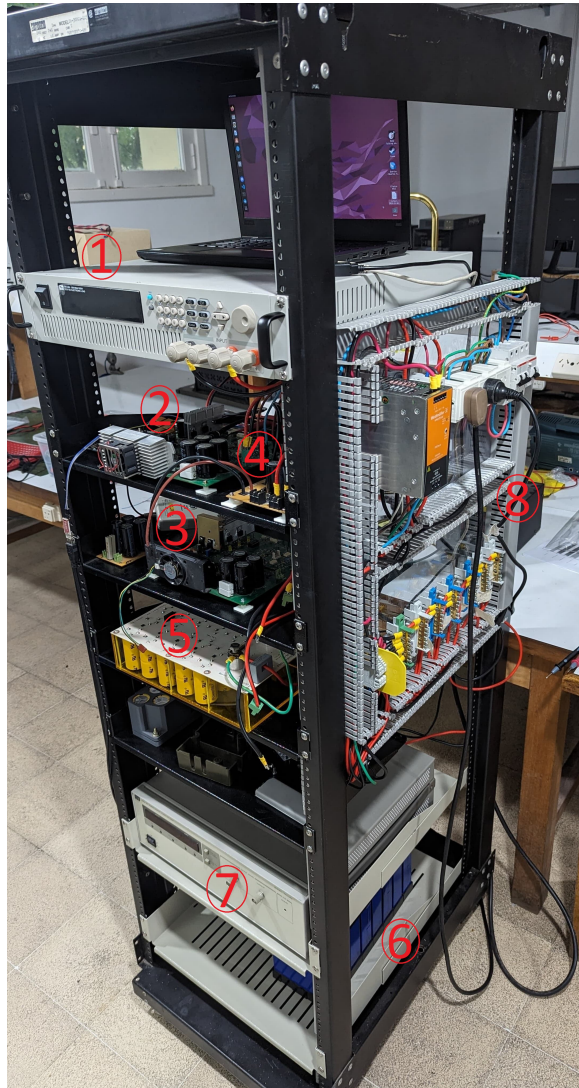


Figura 8.2 Rack de ensayos.

## 8.3. Ensayos

### 8.3.1. Fase 1: Ensayo de lazo abierto

Para comenzar, se llevó a cabo una prueba del convertidor en lazo abierto, donde se configuró un ciclo de trabajo fijo de 50 % y una tensión de entrada de 10V. Durante estos ensayos, se utilizó una fuente de potencia para probar el algoritmo implementado en la FPGA en un entorno controlado, en caso de algún error tanto en el diseño como en la programación. En esta prueba se registró una tensión de salida de aproximadamente 20V, lo que confirma el correcto funcionamiento del convertidor CC-CC.

### 8.3.2. Fase 2: Lazo de control de corriente

El siguiente ensayo consistió en probar el control de lazo cerrado de corriente por el inductor del convertidor CC-CC. Para ello, se incorporó la referencia de corriente y el controlador PI que calcula la acción de control necesaria para seguir la referencia configurada. Como medida de seguridad, se implementaron llaves que permiten habilitar o deshabilitar la acción integral del controlador. Además, se agregó una saturación a la salida para evitar la obtención de un ciclo de trabajo del 100%; y el método de anti-windup explicado anteriormente.

En primera instancia, para las pruebas del algoritmo de control de corriente se considero solo la parte proporcional del controlador. Se ingresaron escalones de 3A en la corriente de referencia a través de la interfaz, lo que permitió verificar que el sistema de control reaccionara adecuadamente a estos cambios, aunque se observó un error estacionario significativo, como era de esperarse. Posteriormente, al habilitar el integrador, se pudo observar cómo la corriente del inductor seguía la referencia impuesta por la FPGA con un error estacionario nulo. En la Figura 8.3 se muestra la respuesta del control de corriente ante un escalón positivo de 1A seguido de un escalón negativo de 3A. Es importante destacar que estas formas de onda se obtuvieron directamente desde la opción de exportación a CSV proporcionada por la interfaz, y posteriormente graficadas mediante PlotJuggler.

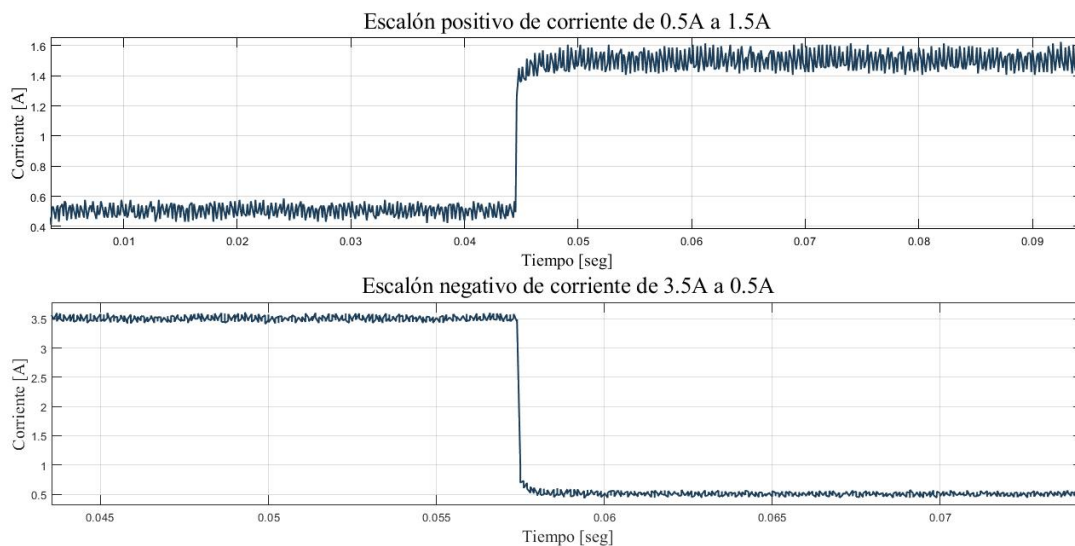


Figura 8.3 Ensayos del lazo de corriente.

### 8.3.3. Fase 3: Lazo de control de tensión

Se procede habilitando el PI de tensión con una ganancia fija. Utilizando la interfaz, se carga una referencia de 48V y se observa en la Figura 8.4 la actuación del *rate limiter* variando la referencia suavemente para evitar picos de corriente que puedan dañar el sistema. Una vez en estado estacionario, se aprecia un error de estado estacionario nulo debido a la acción integral.

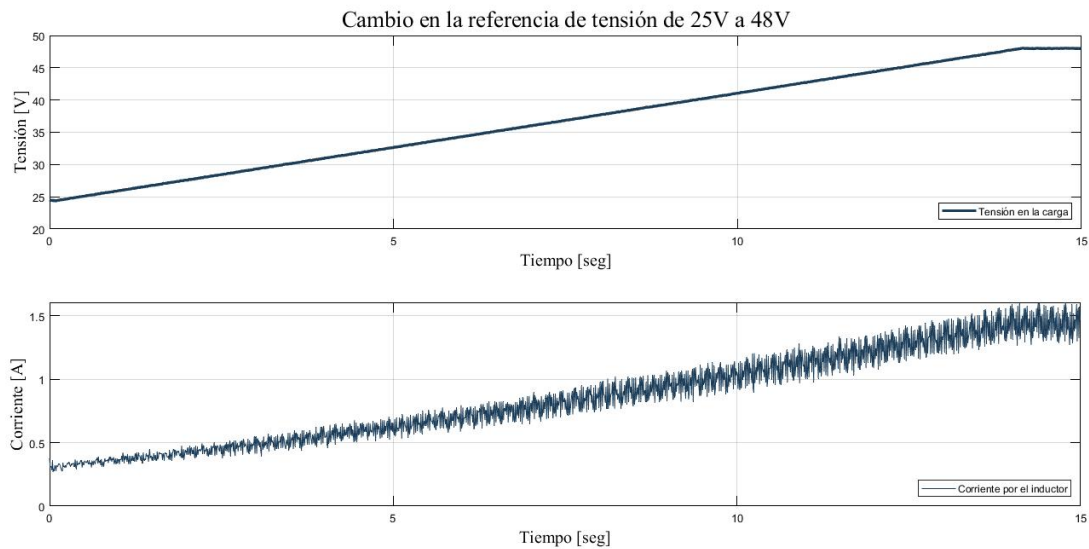


Figura 8.4 Tensión en la carga y corriente por el inductor.

Luego, utilizando la carga electrónica, se realizan variaciones de carga para observar el rechazo a perturbaciones del PI con ganancias fijas respecto al PI con planificación de ganancias.

Inicialmente se realiza un escalón de carga de  $14,54\Omega$  a  $74\Omega$  que implica un escalón de corriente en la entrada del convertidor de  $7,5A$  a  $1,5A$ . Si bien, en teoría estaría actuando el mismo PI en la primera región, el hecho de aplicar la interpolación de ganancia implica que los parámetros del controlador no serán exactamente los mismos, esto se debe a que al implementar la curva de interpolación de segundo orden, aparece un error mínimo con los parámetros sintonizados modificando la respuesta del control. De esta manera, se espera un comportamiento ligeramente diferente pero favorable (más rápido), el cuál se ve reflejado en la Figura 8.5.

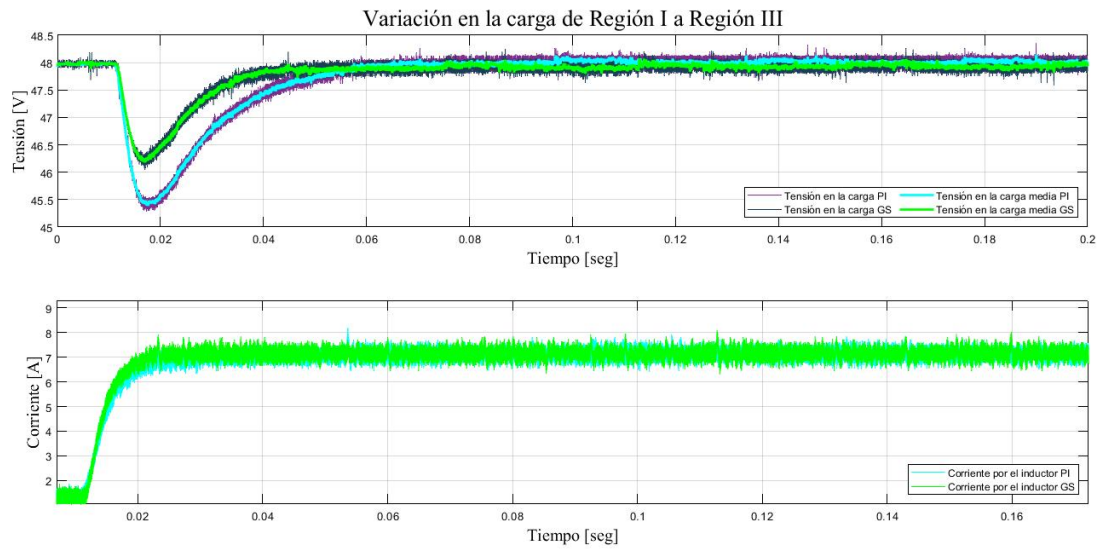


Figura 8.5 Comparación de las formas de onda de tensión en la carga y corriente por el inductor frente a una variación en la carga de  $14,54\Omega$  a  $74\Omega$ .

Posteriormente, se invierte el escalón de carga de  $74\Omega$  (Región I) a  $14,54\Omega$  (Región III). Como se está actuando en una zona distinta a la sintonizada, se espera una mejor respuesta de parte de la planificación de ganancias. En la Figura 8.6 se grafican ambos rechazos y se aprecia la mejora esperada validando el desarrollo realizado en el Capítulo 5.

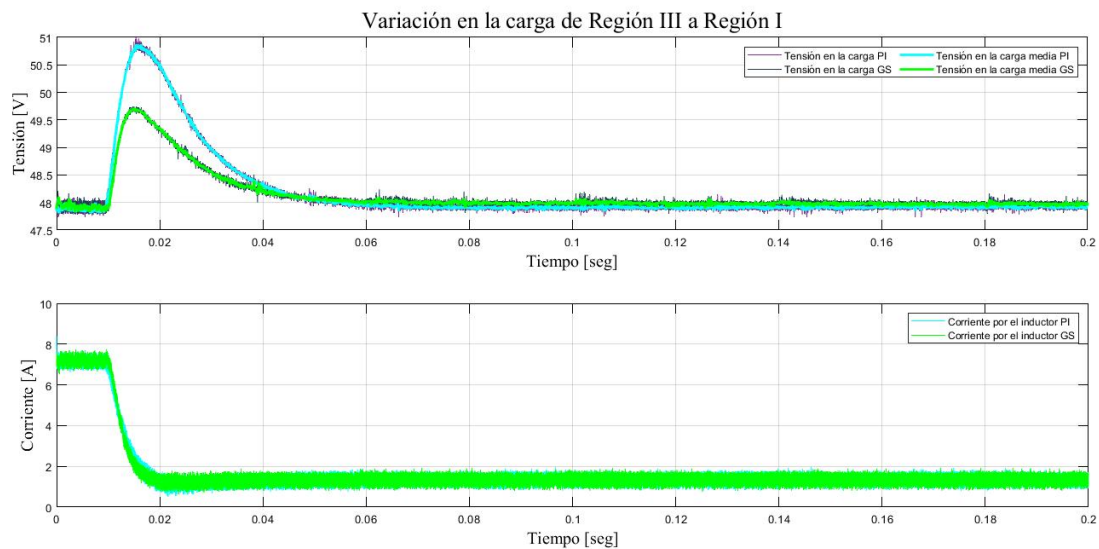
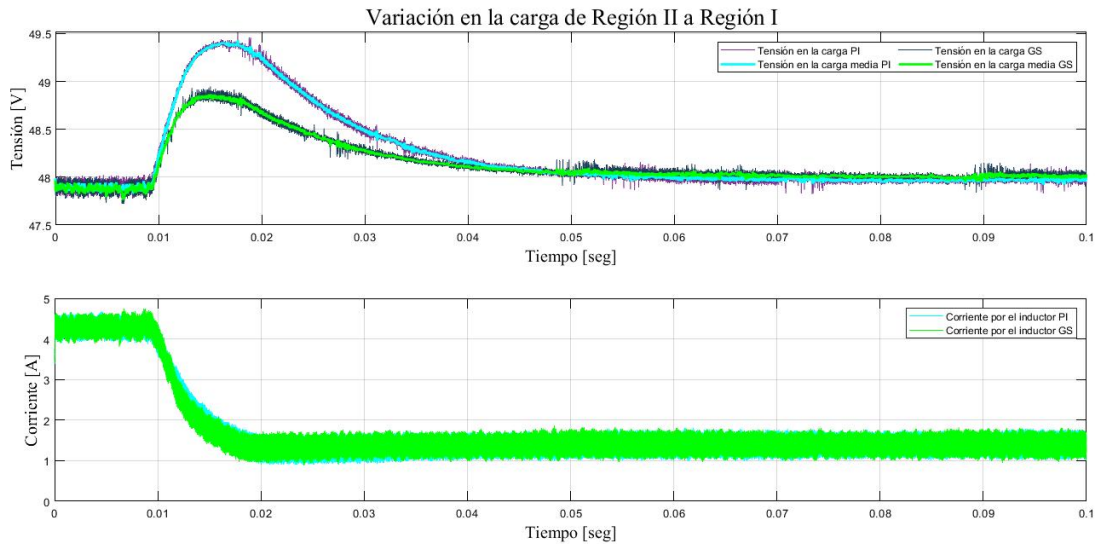
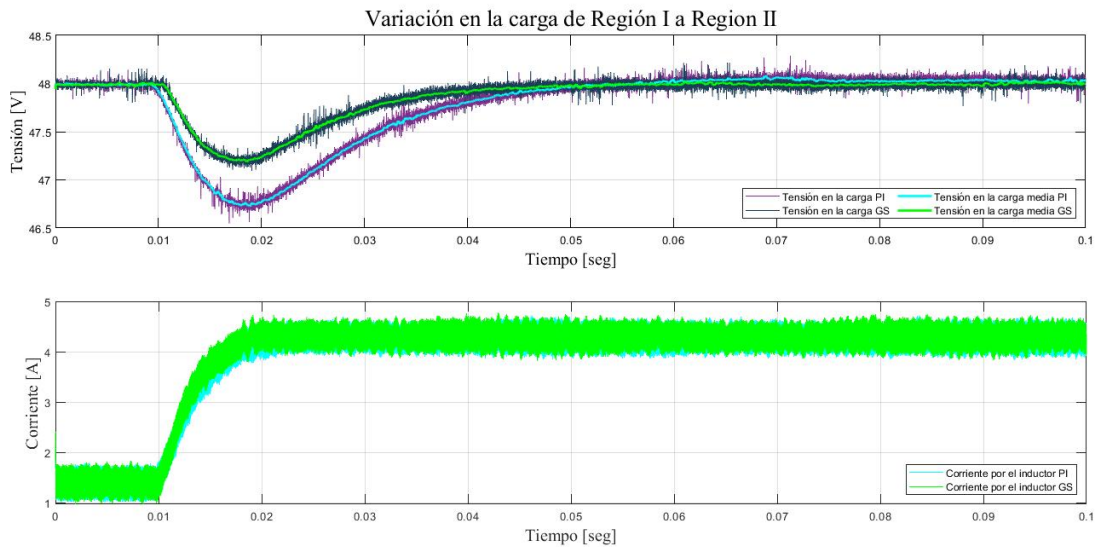


Figura 8.6 Comparación de las formas de onda de tensión en la carga y corriente por el inductor frente a una variación en la carga de  $74\Omega$  a  $14,54\Omega$ .

A continuación se muestran una serie de ensayos realizados de distintas perturbaciones en la carga. Con el objetivo de obtener una mejor noción del desempeño de la planificación de ganancias respecto al PI en distintas condiciones operativas.



(a) Variación en la carga de  $74\Omega$  a  $24\Omega$ .



(b) Variación en la carga de  $24\Omega$  a  $74\Omega$ .

Figura 8.7 Comparación de las formas de onda de tensión en la carga y corriente por el inductor frente a diferentes variaciones en la carga.

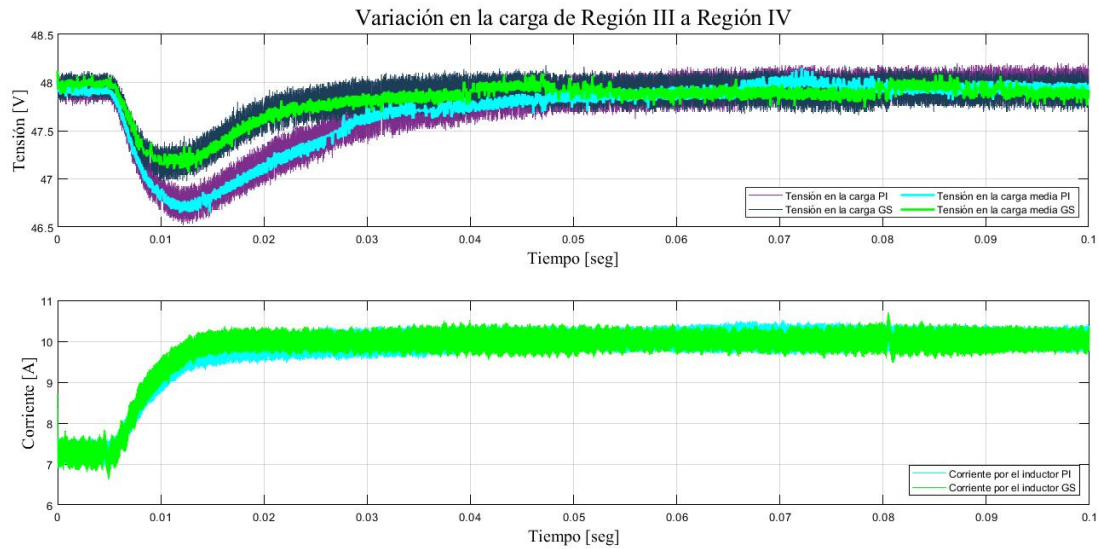


Figura 8.8 Variación en la carga de  $14,54\Omega$  a  $10,8\Omega$ .

Luego de llevar a cabo estos ensayos, se concluye que tanto el controlador PI de corriente como el de tensión exhiben respuestas satisfactorias. La técnica GS permite mantener un tiempo de establecimiento constante, aproximadamente de  $35ms$ , tal como se observó en la simulación presentada en el Capítulo 5, ante diversas variaciones en la carga.

Además, se destaca que esta última es más rápida en comparación con la sintonización de un único controlador PI, demostrando una capacidad notable para rechazar perturbaciones, exhibiendo una caída de tan solo  $1,5V$  para variaciones en la carga de una región y menos de  $2V$  para perturbaciones en dos regiones.

Por estas razones, se concluye que el control avanzado no lineal, en particular la técnica de GS, es una buena opción para este proyecto.

## **8.4. Resumen**

En este Capítulo, se abordaron una serie de ensayos experimentales y los resultados de los mismos en el contexto del desarrollo del sistema de control.

Primeramente, se describió de manera concisa el rack de pruebas desarrollado en el marco de este proyecto, el cual demostró ser una herramienta clave para la validación y evaluación de sistemas, proporcionando un entorno controlado y seguro para llevar a cabo experimentos, mediciones y ensayos.

A partir de los resultados obtenidos en los ensayos experimentales, se validó el desarrollo del sistema de control con planificación de ganancias, haciendo especial énfasis en la consistencia del desempeño del mismo ante diversas condiciones operativas.

## Capítulo 9

# Conclusiones

En el presente trabajo se realizó un estudio de las estrategias de control para un sistema híbrido de almacenamiento de energía basado en baterías de litio y banco de supercapacitores orientados a aplicaciones móviles. En este contexto, se trabajó en el modelado, simulación e implementación de controladores para convertidores electrónicos de potencia con el desarrollo de una interfaz gráfica capaz de interactuar con la FPGA y monitorear en tiempo real o mediante la obtención de datos en archivos CSV las variables de interés del sistema.

En resumen, este desarrollo ha demostrado que la técnica GS posee la capacidad de mantener un tiempo de establecimiento constante frente a diversos modos de operación, destacando su eficacia para rechazar distintas perturbaciones en comparación con un controlador PI convencional. Estos resultados subrayan la importancia de desarrollar controladores que tienen en cuenta las no linealidades del sistema, especialmente en el contexto de sistemas híbridos con aplicaciones móviles.

Sin embargo, este proyecto no solo confirma la eficiencia de la técnica GS y correcto funcionamiento de la interfaz, sino que considerando un uso de solo el 18 % de los elementos lógicos disponibles del FPGA, se abren nuevas puertas para la exploración de técnicas de control aún más sofisticadas, como el *sliding mode control*, que prometen ofrecer un rendimiento aún más robusto y adaptable ante diversas condiciones y perturbaciones. Asimismo, se plantea la posibilidad de llevar a cabo experimentos que integren ambos sistemas de almacenamiento con las correspondientes medidas de seguridad, así como la implementación de mejoras en la interfaz para ofrecer una experiencia más enriquecedora al usuario.

En definitiva, este estudio sienta las bases para futuras investigaciones en el campo del control de sistemas híbridos, ofreciendo no solo una comprensión más profunda de las técnicas de control, sino también la mejora continua en la eficiencia y la fiabilidad de estos sistemas en aplicaciones móviles y más allá.

Este proyecto abarca distintas áreas de la ingeniería electrónica, como el modelado de distintos sistemas, implementación digital, codificación de datos para comunicación y la interfaz de usuario para la interacción con el sistema embebido. Durante dicho desarrollo se profundizaron conocimientos introducidos en la carrera y se adquirieron diversas habilidades indispensables al momento de trabajar en un proyecto de gran magnitud. Algunas de estas fueron la herramienta de versionamiento git para mantener una estructura ordenada, la programación orientada a objetos con el lenguaje python, la programación de hardware a través del lenguaje Verilog y el análisis de sistemas no lineales.

# Bibliografía

- [1] B. d. D. Muñoz Zurita, "Evaluación técnica y económica de distintos tipos de tecnologías de almacenamiento de energía y posibles aplicaciones en el sistema eléctrico nacional," 2020.
- [2] "World energy outlook," 2022.
- [3] "Global coal demand expected to decline in coming years," 2023.
- [4] "Global ev outlook 2019," 2019.
- [5] A. M. M.A. Hannan, F.A. Azidin, "Hybrid electric vehicles and their challenges," 2013.
- [6] B. Long, S. T. Lim, Z. F. Bai, J. H. Ryu, and K. T. Chong, "Energy management and control of electric vehicles, using hybrid power source in regenerative braking operation," 2014.
- [7] A. H. M.Y. Ayad, M. Becherif, "Vehicle hybridization with fuel cell, supercapacitors and batteries," 2010.
- [8] E. F. MÉNDEZ, E. V. ARROBO, and A. F. MOROCHO, "Supercapacitors as a contribution to electrical energy development, comparative analysis using computer simulation tools applied," 2020.
- [9] P. F. y J. I. Talpone y J. Moré y D.Riva, "Diseño y construcción de un banco de supercapacitores," 2016.
- [10] M. y Alejandro, "Explotación del litio, producción y comercialización de baterías de litio en argentina," 2014.
- [11] M. V. I. A. y Martínez Prado Miguel A, "Sistema de carga para celdas de litio-ferrofosfato," 2021.
- [12] R. Byczek, "Transporting lithium batteries used in medical devices," 2019.
- [13] I. MAHAMMAD A. HANNAN 1 (Senior Member, I. MD. MURSHADUL HOQUE2 (Student Member, Y. Y. AINI HUSSAIN3, and I. PIN JERN KER1, (Member, "State-of-the-art and energy management system of lithium-ion batteries in electric vehicle applications: Issues and recommendations," 2018.
- [14] D. W. Hart, *Power Electronics*. McGraw-Hill, 2010.
- [15] E. C. y Gonzalo M. Irusta, "Diseño y control de una etapa de potencia para una pila de combustible tipo pem," 2011.

- [16] D. M. Robert W. Erickson, "Fundamentals of power electronics, supercapacitors and batteries third edition," 2020.
- [17] J. S. y col., "Averaged modeling of pwm converters operating in discontinuous conduction mode," 2001.