

UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**CONTROL DIGITAL DE
CONVERTIDOR CONMUTADO
MEDIANTE FPGA MIXTA
ANALÓGICA/DIGITAL**

Ingeniería de Telecomunicación

Patricia Cuesta Marcos
Julio 2015

CONTROL DIGITAL DE CONVERTIDOR CONMUTADO MEDIANTE FPGA MIXTA ANALÓGICA/DIGITAL

AUTOR: Patricia Cuesta Marcos
TUTOR: Ángel de Castro Martín

Human Computer Technology Laboratory (HCTLab)
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Julio 2015

Resumen

El presente proyecto tiene como objetivo la regulación de la tensión de salida de un convertidor Boost Elevador mediante controles en lazo cerrado.

La regulación se llevará a cabo mediante la placa de desarrollo “Actel Fusion Kit” de Microsemi. Destacar que la FPGA incluida en la placa de desarrollo utilizada, además de contar con un bloque digital, cuenta con un bloque analógico a través del cual será posible llevar a cabo las conversiones analógica a digital.

En primer lugar, se llevará a cabo la regulación mediante un control por medio de un único lazo de tensión y por último, se llevará a cabo la regulación por medio de un control de doble lazo. Este control utilizará un lazo rápido de corriente y un lazo lento de tensión.

Abstract

This project aims to regulate the output voltage of a Boost converter by different closed loop controls.

The regulation will take place with the development board provided in Microsemi’s “ Actel Fusion Kit”. The FPGA included in the development board incorporates an analog block in addition to the digital one. This analog block will be responsible of carrying out the analog-to-digital conversions.

First of all, a single voltage loop will be implemented. Secondly, a double loop control will be implemented to control the output voltage of the Boost Converter. The double control loop consists of a fast current loop and a slow voltage loop interaction.

Agradecimientos

Agradecer en primer lugar a mi tutor Ángel la labor de dirección y supervisión realizada, sin la cual este proyecto no hubiera sido posible. También agradecer al HCTLab haberme acogido.

A todos esos amigos con los que he compartido el tiempo a lo largo de estos años.

A mis compañeros de trabajo, por dejarme escapar cuando lo he necesitado.

A mi familia, por su apoyo incondicional, y en especial a Miguel, por toda su paciencia.

A Paco, por permanecer a mi lado durante todos estos años, tanto en los buenos, como en los malos momentos.

A mis padres, Marcelino y Mercedes, que sin ellos nada de esto hubiera sido posible.

Y en último lugar, pero no menos importante, a mi hermano, Alberto, a quien dedico este proyecto porque ¿Qué haría yo sin él?.

Muchas gracias a todos.

Índice general

Agradecimientos	v
Índice de figuras	x
1. Introducción y Estado del Arte	1
1.1. Antecedentes Históricos	1
1.2. Estado del Arte	3
1.2.1. Fuentes de alimentación	3
1.2.2. Sistemas de Control	5
1.2.3. Factor de Potencia	10
1.3. Objetivos del proyecto	14
2. Tecnología utilizada	17
2.1. “ <i>Fusion Starter Kit</i> ”	17
2.1.1. Bloque Digital	19
2.1.2. Bloque Analógico de la FPGA AFS600	19
2.2. Herramientas Software Utilizadas	23
2.2.1. Matlab	24
2.2.2. ModelSim	25
2.2.3. Libero IDE	25
3. Convertidor Boost Elevador	31
3.1. Convertidor Boost Elevador	31
3.1.1. MCC - Modo de Conducción Continuo	32
3.1.2. MCD - Modo de Conducción Discontinua	34
3.2. Boost Utilizado	36
4. Estrategia de control	39
4.1. Controladores Digitales	39
4.1.1. Fundamentos Matemáticos de los Sistemas de Control	41
4.1.2. Sistemas de Control en Lazo Cerrado	42

4.1.3. Acciones de Control	43
4.1.4. Estructuras de Control	45
4.2. Diseño del Lazo de Tensión	49
4.2.1. Diseño del Controlador	51
4.3. Diseño del Doble Lazo	52
4.3.1. Diseño de Controladores	54
4.4. Implementación de los Lazos	57
4.4.1. Formato QX.Y	58
4.4.2. Implementación de Funciones de Transferencia	59
5. Simulaciones	61
5.1. Modelos VHDL	61
5.2. Controladores	63
5.2.1. Control de Lazo de Tensión	63
5.2.2. Control de Doble Lazo	65
6. Pruebas Experimentales	69
6.1. Resultados Experimentales	69
6.1.1. Control de Lazo de Tensión	70
6.1.2. Control de Doble Lazo	76
6.1.3. Resultados en Lazo Cerrado	77
7. Conclusiones y Trabajo Futuro	79
7.1. Conclusiones	79
7.2. Trabajo Futuro	80
Glosario de acrónimos	81
Bibliografía	83
Anexo: FPGA	I
Anexo: Modelo VHDL Convertidor Boost Elevador	IX
Anexo: Modelo VHDL ADC	XI
Anexo: Código VHDL Regulador Lazo de Tensión Real	XIII
Anexo: Código VHDL Regulador Lazo de Tensión	XV
Anexo: Código VHDL Regulador Doble Lazo	XIX

Anexo: Código VHDL Selec	XXV
Presupuesto	XXVII
Pliego de condiciones	XXIX

Índice de figuras

1.1. Cronograma	2
1.2. Diagrama de Bloques de la Fuente Lineal	4
1.3. Diagrama de Bloques de la Fuente Conmutada	4
1.4. Topologías de Convertidores Conmutados	5
1.5. Modulación PWM: Onda Cuadrada	7
1.6. Lazo Abierto	8
1.7. Lazo Cerrado	8
1.8. Control PWM por muestra de tensión	9
1.9. Control PWM por muestra de corriente	9
1.10. Control de Doble Lazo	10
1.11. Triángulo de Potencias	11
1.12. Diagrama Rectificador	12
1.13. Circuitos Pasivos	13
1.14. Convertidor AC/DC de Doble Etapa	13
1.15. Boost Elevador	14
2.1. Sistema Típico	17
2.2. Fusion Starter Kit	18
2.3. Reloj	20
2.4. Diagrama Bloques ADC	20
2.5. Diagrama SAR del ADC	21
2.6. Diagrama de bloques del multiplexor de entrada del ADC	22
2.7. Sisotool	24
2.8. ModelSim	25
2.9. ProjectFlowWindow	26
2.10. Analog System Builder	27
2.11. Analog system Buider Advanced Option Dialog Box	28
2.12. Interfaz Gráfica FlashPro	29
3.1. Boost Elevador	32

3.2. Formas de onda en MCC	33
3.3. Formas de onda en MCD	35
3.4. Rizado en Tensión de Salida	36
3.5. Boost Elevador	36
3.6. Divisores de Tensión de Entrada y Salida	37
3.7. Divisor de Tensión para la Medición de la Corriente de Entrada	37
4.1. Lazo Cerrado	39
4.2. Función de Transferencia	41
4.3. Control Realimentado	43
4.4. Acciones de Control P, PI y PD	44
4.5. Ciclos de trabajo distintos para mismo periodo	45
4.6. Lazo de Tensión	46
4.7. Doble Lazo	46
4.8. Formas de onda para V_i, I_L, P_i, P_o, V_o	47
4.9. Tensión Rectificada	47
4.10. Detección de paso por cero	48
4.11. Frecuencia de la Red normal vs frecuencia rectificada	48
4.12. Regulador de lazo de tensión	50
4.13. Respuesta Controlador Lazo de Tensión	51
4.14. Respuesta Controlador Lazo de Tensión - CT1	52
4.15. Reguladores de lazo de tensión y corriente	54
4.16. Respuesta Controlador Lazo de Tensión	55
4.17. Respuesta Controlador Lazo de Corriente	55
4.18. Respuesta Controlador Lazo de Corriente - C1	56
4.19. Respuesta Controlador Lazo de Corriente - C2	57
4.20. Respuesta Controlador Lazo de Corriente - C3	57
4.21. Formato QX.Y	58
4.22. Suma Formato QX.Y	58
4.23. Producto Formato QX.Y	59
4.24. Generación de la señal PWM	60
5.1. Sincronización del canal y resultado del ADC	63
5.2. Resultados Sisotool vs ModelSim	64
5.3. Comparativa Regulador real VS std_logic_vector	64
5.4. Simulación controladores utilizados D= 50 %	66
5.5. Simulación Lazo Cerrado - Tiempo de adquisición y conversión ADC = 3,33 μ s	66

5.6. ASB - Secuencia de muestreo	67
5.7. Simulación Lazo Cerrado - Tiempo de adquisición y conversión ADC = 36,15 μ s .	67
5.8. Simulación Lazo Cerrado - Tiempo de adquisición y conversión ADC = 1000/100 ns	68
6.1. SmartTime	70
6.2. Esquema Conexionado: Control de Lazo de Tensión	70
6.3. Control Lazo de Tensión: Lazo Abierto	71
6.4. Control Lazo de Tensión: Lazo Abierto	72
6.5. Control Lazo de Tensión: Lazo Cerrado	72
6.6. Simulación Lazo Cerrado - Vref=40 V; Vin=15 V; R=200 Ω ;	73
6.7. Simulación Lazo Cerrado - Vref=40 V; Vin=19;21;27 V; R=200 Ω ;	73
6.8. Simulación Lazo Cerrado - Vref=40 V; Vin=15 V; R=500 Ω ;	74
6.9. Simulación Lazo Cerrado - Vref=40 V; Vin=19;21;27 V; R=500 Ω ;	74
6.10. Simulación Lazo Cerrado - Vref=60 V; Vin=17 V; R=200 Ω ;	75
6.11. Simulación Lazo Cerrado - Vref=60 V; Vin=23;31;37 V; R=200 Ω ;	75
6.12. Simulación Lazo Cerrado - Vref=60 V; Vin=17;23;37 V; R=500 Ω ;	76
6.13. Esquema Conexionado Control de Doble Lazo	76
6.14. Control Doble Lazo: Lazo Abierto	77
6.15. Control Doble Lazo: Lazo Cerrado	78

1

Introducción y Estado del Arte

En este capítulo serán expuestos los antecedentes históricos que han contribuido al desarrollo de lo que hoy conocemos como electrónica junto al estado del arte actual de los controladores digitales. Como el propio nombre del proyecto indica, “*Control digital de convertidor conmutado mediante FPGA mixta analógica/digital*”, el control digital de convertidores conmutados será el tema central del presente proyecto fin de carrera.

Dentro del estado del arte actual se incluye el estado de las fuentes de alimentación, lineales y conmutadas, para después introducir la modulación por ancho de pulso, técnica utilizada para la regulación de la tensión de salida del convertidor conmutado, y los tipos de lazos de control, abierto o cerrado. Finalmente se introduce el factor de potencia, indicador del correcto aprovechamiento de la energía eléctrica y las soluciones para la mejora de este factor.

1.1. Antecedentes Históricos

Desde principios del siglo XVII, la electricidad ha sido materia de interés científico. Desde ese momento, los experimentos llevados a cabo por diferentes científicos en torno a los fenómenos eléctricos y electromagnéticos, fueron asentando las bases para lo que poco tiempo después sería una nueva especialidad, primero la física y seguidamente la ingeniería. Sin embargo, las investigaciones dentro de la ingeniería eléctrica no empezaron a intensificarse hasta el siglo XIX.

El invento de la pila voltaica en 1800 hizo posible el desarrollo de los primeros circuitos eléctricos, es decir, conjunto de elementos que unidos entre sí permiten una circulación de electrones o lo que es lo mismo una corriente eléctrica. La primera aplicación de estos circuitos fue la iluminación eléctrica.

A continuación, en 1884 Thomas Edison detectó el fenómeno termoiónico por el cual los electrones que se encuentran en la superficie de los metales pueden llegar a desprenderse de la misma como consecuencia de un aumento en su velocidad producida por un aumento de temperatura.

Thomas Edison no pudo darse cuenta de la importancia de su descubrimiento y fue John Ambrose Fleming quien en 1905 descubrió que colocando dentro de la bombilla incandescente otro electrodo independiente, y algo alejado del filamento, se establecía una circulación de corriente eléctrica desde el filamento (cátodo) hacia el nuevo electrodo (ánodo). Fue así como Fleming

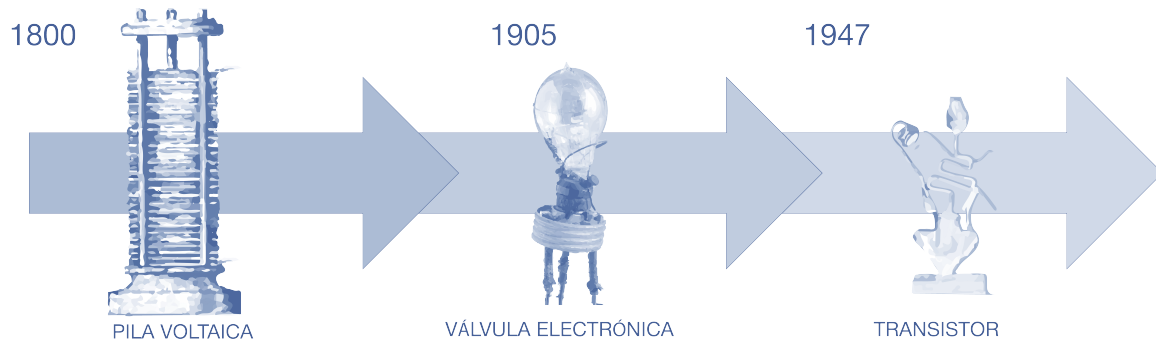


Figura 1.1: Cronograma

inventó la primera válvula electrónica, lo que dio lugar al nacimiento de la ingeniería electrónica y abrió el camino a los posteriores avances de la electrónica moderna. Es importante destacar que todos estos descubrimientos originaron el comienzo del control del paso de corriente.

La válvula presentaba inconvenientes entre los cuales cabe destacar el elevado consumo energético y la escasa fiabilidad debida a sus elevadas temperaturas de funcionamiento, lo que con el paso del tiempo ocasionó que terminaran siendo sustituidos por los transistores [1] tal y como se observa en la figura 1.1. Fue en 1947 cuando William Shockley, John Bardeen y Walter Brattain construyeron el primer transistor respondiendo a la necesidad técnica de hacer llamadas telefónicas a larga distancia. Es por ello que el transistor es considerado uno de los mayores desarrollos tecnológicos del siglo XX.

Los transistores permitían cumplir con las mismas funciones que las válvulas a la vez que utilizaban menores cantidades de energía y se trataba de dispositivos mucho más versátiles, económicos y pequeños.

En los años posteriores a la invención del primer transistor se siguieron produciendo adelantos en este campo: se inventaron distintos tipos de transistores, se emplearon distintos materiales y se logró la integración de un gran número de transistores en un oblea de silicio. Este último adelanto dio lugar a la creación del circuito integrado [2] en 1958 por Kilby. El circuito integrado disminuía la complejidad en el diseño de circuitos electrónicos al aumentar el número de componentes como consecuencia de las malas conexiones, además de propiciar una reducción de las dimensiones, el peso y el coste en relación al número de elementos integrados.

A partir de la comercialización de los circuitos integrados en 1961, la densidad de transistores en un mismo espacio se ha duplicado cada 24 meses. Esta tendencia se ha mantenido hasta la actualidad y es conocida como ley de Moore [3].

Hoy en día, la industria de los semiconductores está sacando al mercado circuitos integrados con transistores de 22 nanómetros [4] [5]. Esto permite que podamos encontrar 4,31 billones de transistores en un único procesador como sucede en el caso del procesador *E7 – 8857* de Intel. Debido a que seguir decrementando el tamaño de los transistores podría suponer un coste de fabricación superior al de venta, los fabricantes están comenzando a plantarse en los tamaños actuales y buscar alternativas como por ejemplo en los materiales [6] [7].

Actualmente existen distintas tecnologías para la implementación de circuitos integrados digitales. Aunque con los ASICs [8] [9], circuitos integrados de aplicación específica, se obtienen velocidades superiores al poder optimizar al máximo el diseño y un menor consumo [10] [11], no resultan beneficiosos para etapas de prototipado. Por el contrario, las FPGAs [12] [13] son dispositivos semiconductores que contienen bloques de lógica cuya interconexión y funcionalidad puede ser reprogramada mediante un lenguaje de descripción especializado. La lógica programable puede reproducir desde funciones tan sencillas como las llevadas a cabo por una puerta

lógica o un sistema combinacional hasta complejos sistemas en un chip.

Dada la versatilidad, flexibilidad y por ser fácilmente programable, el presente proyecto se ha decantado por la utilización de una plataforma hardware de desarrollo que contiene señales mixtas analógica/digital de Microsemi [14] que será detallada en el capítulo 2. Gracias a las capacidades de la FPGA incluida en la plataforma a utilizar, es posible llevar a cabo la captura de señales analógicas mediante un ADC integrado y realizar el procesado con la lógica programable, lo cual permitirá la regulación de la tensión de salida de un convertidor boost elevador.

A continuación, se presenta un estudio del estado del arte de las fuentes de alimentación al encontrarse el convertidor boost elevador dentro de la categoría de las fuentes de alimentación conmutadas. Posteriormente, nos centraremos en los controladores, sistemas por medio de los cuales es posible gobernar una señal de salida a través de una señal de entrada. Por último, se tratará el factor de potencia, indicador que permite describir la cantidad de energía transformada en potencia en las fuentes de alimentación. El factor de potencia se mueve entre valores de 0 y 1, un mayor factor de potencia permite la reducción del consumo energía y por tanto la reducción de la factura eléctrica [15].

1.2. Estado del Arte

Cualquiera de los dispositivos o circuitos electrónicos mencionados anteriormente y los existentes en la actualidad, tienen una cosa en común: la dependencia de la energía para poder funcionar. La energía, la cual puede obtenerse de diversas fuentes como por ejemplo a través de una pila, una batería o de la red eléctrica, tiene un inconveniente, no cuenta con un formato unificado, es decir, podemos encontrarla como corriente continua o corriente alterna.

Actualmente, en España, la red eléctrica viene dada por corriente alterna de 230 V. Esto, en muchos casos, no satisface las necesidades de los diferentes dispositivos conectados a la red eléctrica lo que hace necesario una serie de elementos que permitan transformar la energía de entrada a una de salida con los valores adecuados. Estos elementos, denominados fuentes de alimentación, serán detallados a continuación.

1.2.1. Fuentes de alimentación

En la actualidad las fuentes de alimentación [16] se pueden clasificar en lineales y conmutadas. Cada una de estas fuentes cuenta con sus propias características, ventajas y inconvenientes. Por ello, la utilización de una tipología en concreto dependerá de su uso final y los requerimientos de estabilidad y rendimiento de la carga a alimentar.

Fuentes Lineales

Las fuentes de alimentación lineales, las primeras en desarrollarse, utilizan un transformador para aumentar o disminuir el voltaje de la red eléctrica y poder adecuarlo así a las necesidades del circuito. Estas fuentes, a grandes rasgos quedan definidas por los bloques incluidos en la figura 1.2.

El transformador de entrada modifica los niveles de tensión alterna a los requeridos por el circuito a alimentar aumentando o reduciendo la tensión de red a otra tensión más adecuada para ser tratada.

A continuación, el rectificador es el encargado de convertir la tensión alterna que sale del transformador en tensión continua. Para realizar esta conversión se utilizan diodos, los cuales

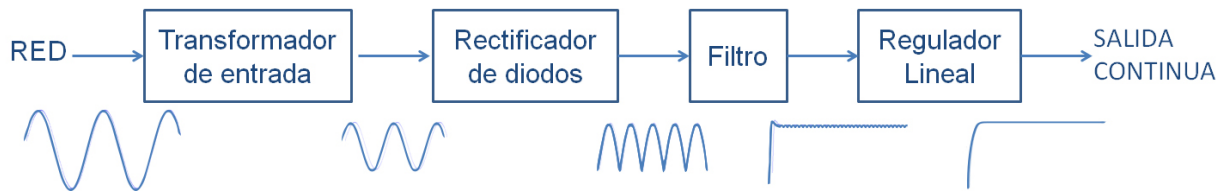


Figura 1.2: Diagrama de Bloques de la Fuente Lineal

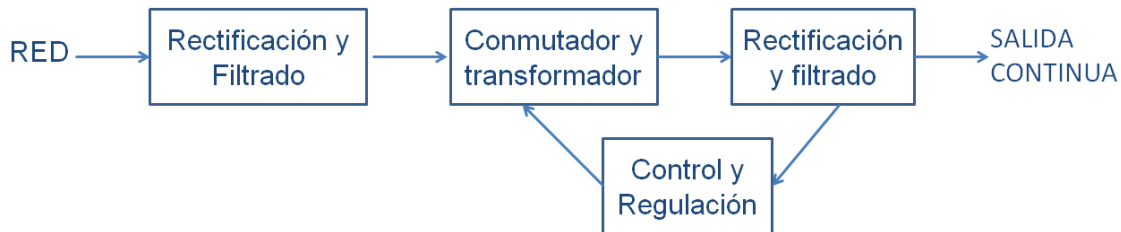


Figura 1.3: Diagrama de Bloques de la Fuente Conmutada

conducen cuando la tensión de su ánodo es mayor que la de su cátodo. La tensión de salida de un rectificador tiene forma de pulsos, es decir, en un ciclo de salida completo la tensión aumenta de cero a un valor de pico, para caer después de nuevo a cero.

Debido a que la mayor parte de circuitos electrónicos precisan de una tensión constante, similar a la que produce una batería, después del rectificador es necesario el empleo de filtro para la obtención de una tensión rectificada. El tipo más común de filtro es el condensador a la entrada del circuito ya que se trata de un elemento que se opone a las variaciones bruscas de la tensión que se le aplica. Sin embargo, en algunos casos puede no ser suficiente y es necesaria la utilización de algunos componentes adicionales.

Por último, el regulador lineal o estabilizador es el circuito encargado de reducir el rizado que puede llevar asociada la tensión proporcionando así una tensión de salida adecuada.

Uno de los principales inconvenientes de las fuentes lineales es su gran tamaño, además de disipar gran parte de la energía en forma de calor. Esto, unido a algunas de las necesidades de la carrera aeroespacial como la reducción de peso y consumo, desencadenó el primer desarrollo de las fuentes de alimentación conmutadas en los años 60, las cuales producen una mayor eficiencia energética y un menor calentamiento además de la reducción en tamaño, peso y coste.

Fuentes Conmutadas o Convertidores

A diferencia de las fuentes lineales, las fuentes conmutadas son convertidores continua/continua, por lo que la red debe ser previamente rectificada y filtrada con una amplitud de rizado aceptable. Estas fuentes se componen de circuitos relativamente complejos, aunque siempre es posible diferenciar cuatro bloques básicos identificados en la figura 1.3.

El primer bloque rectifica y filtra la tensión alterna de entrada convirtiéndola en una continua pulsante. A continuación, el segundo bloque se encarga de convertir esa continua en una onda cuadrada de alta frecuencia.

El tercer bloque rectifica y filtra la salida de alta frecuencia del bloque anterior, entregando así una corriente continua pura.

Por último, el cuarto bloque se encarga de regular la oscilación del segundo bloque. Este bloque consiste en un oscilador de frecuencia fija, una tensión de referencia, un comparador de tensión y un modulador de ancho de pulso, Pulse Width Modulation (PWM). El modulador

recibe el pulso del oscilador y modifica su ciclo de trabajo según la señal del comparador, el cual coteja la tensión continua de salida del tercer bloque con la tensión de referencia.

En sus principios las fuentes conmutadas eran una tecnología muy cara y desconocida, pero el abaratamiento de la electrónica y la miniaturización ha conseguido el desarrollo de las mismas con una mayor estabilidad, seguridad, eficiencia y un precio similar a las fuentes lineales.

Hoy en día se siguen utilizando las fuentes lineales ya que generan menor ruido y rizado, tienen un bajo coste y por su simplicidad, son más inmunes a las radiaciones electromagnéticas.

Topologías de Convertidores Conmutados

Como ya se ha mencionado anteriormente, los convertidores conmutados transforman una tensión de entrada continua en una de salida también continua de diferente nivel. Aunque existen diferentes topologías de convertidores, a continuación se detallan los más frecuentes:

- Elevador o Boost: La tensión de salida es mayor que la de entrada.
- Reductor o Buck: La tensión de salida es menor que la de entrada.
- Elevador-Reductor o Boost-Buck: La tensión de salida puede ser mayor o menor pero invertida.

Todos ellos cuentan con un funcionamiento básico consistente en el almacenamiento temporal de energía y la cesión de esta tras un periodo de tiempo que determinará la cantidad de energía cedida a la carga.

A continuación, en la figura 1.4 se pueden observar los esquemas de cada una de las tipologías mencionadas. Aunque todos los esquemas cuentan con los mismos elementos, la disposición de estos varía en función de la topología.

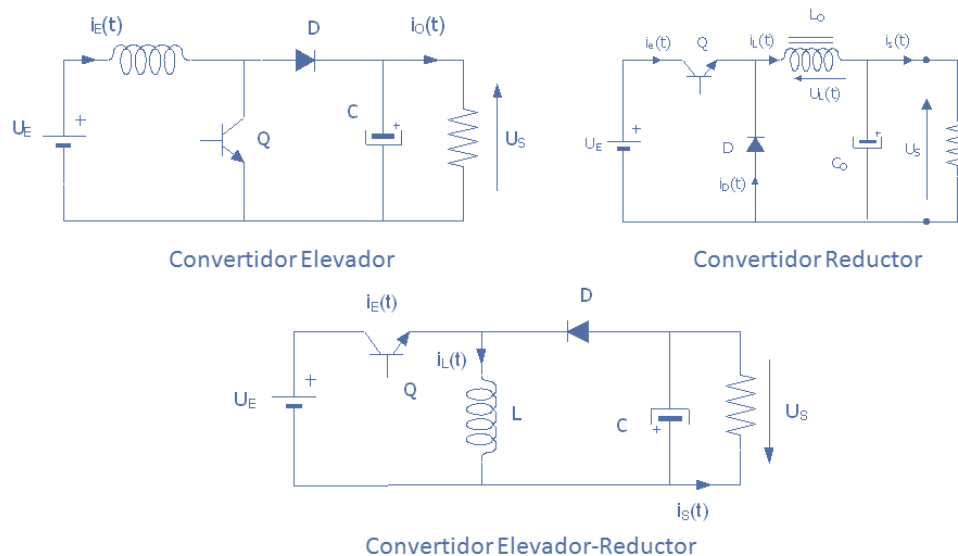


Figura 1.4: Topologías de Convertidores Conmutados

1.2.2. Sistemas de Control

Los métodos de análisis de sistemas y diseño de los controladores [17] [18] [19] han evolucionado del mismo modo que la mayoría de las herramientas utilizadas en ingeniería. Es por ello que en los últimos años estos progresos han permitido que las investigaciones y aplicaciones en la teoría de control hayan pasado de utilizar una implementación analógica a una implementación digital.

En concreto, un sistema de control puede definirse como un sistema que trata de gobernar un cierto proceso físico mediante la generación de una señal de salida a partir de una señal de entrada. La finalidad de un sistema de control es conseguir, mediante la manipulación de las variables de control, un dominio sobre las variables de salida, de modo que estas alcancen unos valores prefijados.

Mientras que en un esquema analógico todas las variables son función de un tiempo continuo, las variables de los esquemas digitales dependen de instantes determinados, es decir, de un tiempo discreto.

La ventaja de un control analógico es el gran ancho de banda con el que trabaja, consecuencia del uso de señales en tiempo real, lo que es equivalente a frecuencia de muestreo infinita de modo que el efecto del controlador está presente en todo momento. Sin embargo, el incremento en la utilización de los sistemas digitales frente a los analógicos con el paso de los años se debe a las ventajas que estos proporcionan.

Para poder entender el aumento del uso de los controles digitales, a continuación se pueden encontrar las ventajas más destacadas de estos sistemas [17]:

- Menor susceptibilidad al envejecimiento y a las variaciones de las condiciones ambientales.
- Menor sensibilidad al ruido y a las perturbaciones.
- Mayor fiabilidad ya que poseen una mejor sensibilidad frente a la variación de parámetros.
- Posibilidad de generar algoritmos de control altamente complejos.
- Facilidad de modificación de los algoritmos de control ya que un cambio en el control no requiere cambios en el hardware.
- Posibilidad de uso del ordenador para tareas adicionales como reconocimiento de imágenes, conversión de coordenadas, etc.
- Reprogramación.
- Menor peso y tamaño.

Aunque tampoco se debe olvidar que los sistemas de control digital también presentan desventajas frente a los sistemas analógicos como por ejemplo:

- Necesidad de la utilización de conversores analógico/digital que permitan discretizar las señales y conversores digital/analógico para devolver las señales a un tiempo continuo.
- Limitaciones en la velocidad de cálculo y en la resolución de la señal debido a la longitud de la palabra finita del procesador digital en contraste con los controladores analógicos que trabajan en tiempo real y tienen una resolución teóricamente infinita.
- Limitación en la velocidad provocada por retrasos en el lazo de control, los cuales pueden provocar inestabilidad en el sistema.
- Precio.

En lo referente a los sistemas de control digital, existen múltiples criterios de clasificación, pero en este caso nos centraremos en la estrategia de control que siguen, es decir, lazo abierto o lazo cerrado, ya que el presente proyecto fin de carrera tiene como objetivo la utilización de un control digital de lazo cerrado para la regulación de la tensión de salida de un convertidor alterna/continua.

Modulación por Ancho de Pulso (PWM)

La modulación por ancho de pulso o PWM [20] de una señal o fuente de energía es una técnica en la que se modifica el ciclo de trabajo de una señal periódica, ya sea para codificar la información de una transmisión o para controlar la energía entregada a una carga o dispositivo eléctrico.

En la actualidad es posible encontrar modulación por ancho de pulso incluida en circuitos funcionales que tienen como objetivo el control de fuentes conmutadas, motores, elementos termoeléctricos, etc.

Se denomina ciclo de trabajo de la señal periódica a la proporción de tiempo en alto a intervalos regulares o periódicos de tiempo. El ciclo de trabajo queda expresado entonces por:

$$D = \frac{t_{on}}{T} \quad (1.1)$$

donde:

- D es el ciclo de trabajo
- t_{on} es el tiempo en alto o ancho de pulso
- T es el periodo de la señal

El ciclo de trabajo se expresa en tanto por ciento. Por ello, $D = 100\%$ corresponde con un periodo completo en estado alto.

Para el caso concreto bajo estudio, controladores digitales, el buen funcionamiento de la modulación por ancho de pulso se debe a su naturaleza de encendido/apagado que permite establecer de manera sencilla el ciclo de trabajo deseado.

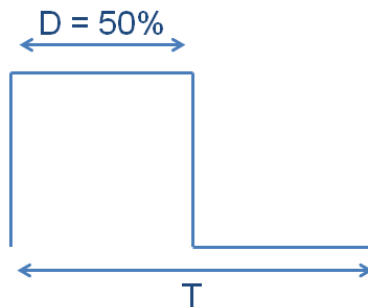


Figura 1.5: Modulación PWM: Onda Cuadrada

Lazo Abierto

Los sistemas de control en lazo abierto (ver figura 1.6) son aquellos en los que la variable de salida no tiene efecto sobre la acción o variable de control. Este sistema suele aparecer en dispositivos con control secuencial, en el que no hay una regulación de variables sino que se realizan una serie de operaciones de una manera determinada.

La estrategia de control lazo abierto puede compensar los retrasos inherentes del sistema anticipándose a las necesidades del usuario, sin embargo, esto generalmente es insuficiente debido a los errores del modelo y a los errores en la estimación de las perturbaciones.

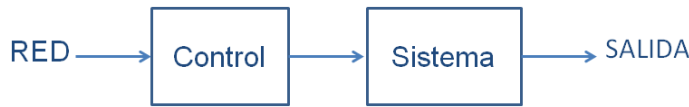


Figura 1.6: Lazo Abierto

Lazo Cerrado

Los sistemas de control en lazo cerrado (ver figura 1.7) son aquellos en los que la señal de salida del sistema tiene efecto directo sobre la acción o variable de control.

Para aquellos casos en los que se utiliza una estrategia de control de lazo cerrado, las perturbaciones, aunque sean desconocidas, son consideradas indirectamente mediante sus efectos sobre las variables de salida. Este tipo de estrategia de control puede aplicarse sea cual sea la variable controlada. Por ello, la gran mayoría de los sistemas de control que se desarrollan en la actualidad son en lazo cerrado.

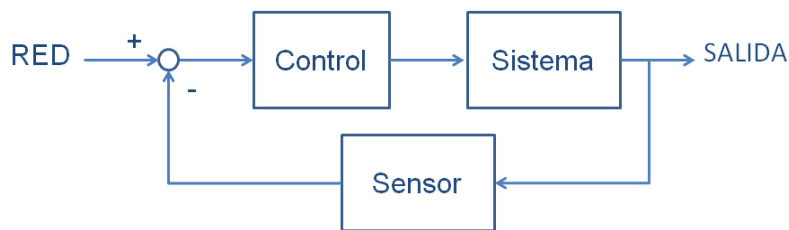


Figura 1.7: Lazo Cerrado

En el caso particular del control digital, independientemente del tipo de convertidor utilizado, la señal de salida tiene una dependencia directa con la señal de ciclo de trabajo. Dependiendo de la frecuencia de conmutación de esta señal es posible encontrar:

- Sistemas de control a frecuencia fija.
- Sistemas de control a frecuencia variable.

Los sistemas de control a frecuencia fija son aquellos sistemas donde la frecuencia de conmutación se mantiene constante, variando únicamente el tiempo de conducción de la señal de ciclo de trabajo. Por el contrario, los sistemas de control a frecuencia variable pueden realizarse a su vez de dos formas, según el tiempo de conducción sea fijo o variable.

El sistema de control a frecuencia fija, el utilizado en el presente proyecto fin de carrera, consiste en cambiar la señal de error obtenida de la comparación de la señal de salida y la señal de referencia, con otra señal de frecuencia fija procedente de un oscilador que determina la frecuencia de funcionamiento del sistema. El resultado de esta combinación es otra señal, cuya frecuencia queda fijada por el oscilador y cuyo ciclo de trabajo es función de la señal de salida. Se dice entonces que la señal obtenida está modulada mediante Modulación por Ancho de Pulso (PWM).

A su vez, los sistemas de control en lazo cerrado pueden categorizarse en:

- Sistema de control PWM por muestra de tensión.
- Sistemas de control PWM por muestra de corriente.
- Sistema de control PWM por muestra combinada tensión-corriente.

Sistema de Control PWM por muestra de tensión

El funcionamiento de este sistema, también conocido por Control de Tensión, se basa en la obtención de una muestra directamente de la salida del circuito para ser comparada posteriormente con una tensión de referencia que contiene la tensión de salida deseada como se puede observar en la figura 1.8. De esta comparación se obtiene una señal de error capaz de generar la señal de PWM que actúa sobre el conmutador obligando a este a variar su ciclo de trabajo.

En estos sistemas, la regulación de la tensión de salida no es inmediata y depende de la magnitud de la variación, por tanto, es posible que el circuito necesite varios ciclos de conmutación para llevar a la señal de salida al nivel deseado.

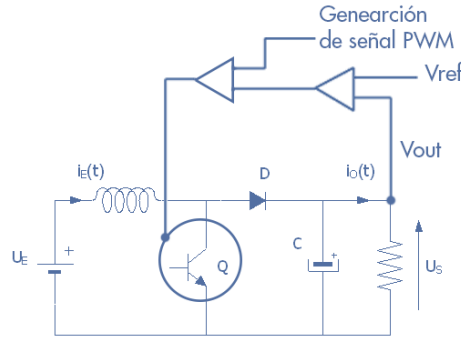


Figura 1.8: Control PWM por muestra de tensión

Sistemas de control PWM por muestra de corriente

Los sistemas de control PWM por muestra de corriente, también conocidos como Control de Corriente, toman normalmente la muestra de corriente directamente sobre el inductor ya que esta es la zona donde se producen las máximas variaciones de la corriente por la carga. Este sistema estabiliza al circuito frente a las posibles variaciones de la señal de entrada.

Del mismo modo que sucede en el Control de Tensión, la muestra de corriente es comparada con una corriente de referencia dando lugar a una señal de error encargada de la generación de la señal de conmutación que actúa sobre el conmutador (ver figura 1.9).

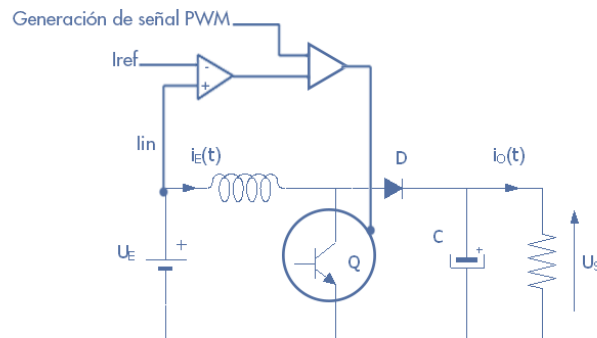


Figura 1.9: Control PWM por muestra de corriente

Sistema de control PWM por muestra combinada tensión-corriente

Este sistema lleva a cabo el control por medio de una etapa que contiene un doble lazo, mientras un lazo regula la corriente de entrada, el otro regula la tensión de salida.

El lazo de tensión, compara la tensión de salida con una tensión de referencia generando así

la señal de error. Esta señal de error será utilizada como referencia en el lazo de corriente, por lo que será comparada con la corriente de entrada de modo que la salida tras la comparación actuará sobre el ciclo de trabajo utilizado como conmutador.

El lazo de corriente se encarga de mantener la corriente de entrada en el valor deseado. La corriente de entrada deberá ser proporcional a la tensión de entrada y conseguir así la corrección del factor de potencia.

A continuación, en la figura 1.10 se puede observar el esquema de doble lazo analizado.

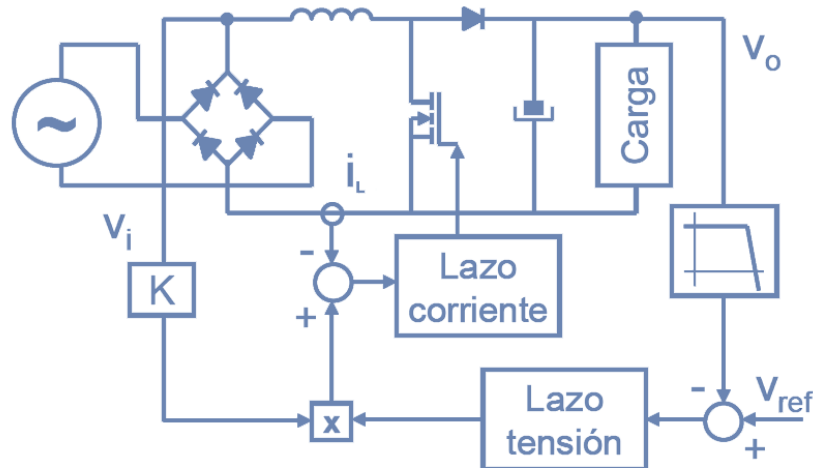


Figura 1.10: Control de Doble Lazo

Este sistema de control se encuentra dentro de las soluciones activas para el corrector de factor de potencia descritas en el apartado 1.2.3.

1.2.3. Factor de Potencia

El Factor de Potencia [20] [21] es un indicador cualitativo y cuantitativo del correcto aprovechamiento de la energía eléctrica aunque también puede describirse como el término utilizado para describir la cantidad de energía eléctrica convertida en trabajo.

La medición de potencia en corriente alterna es más complicada que la de corriente continua debido al efecto de las bobinas y condensadores, por ello en cualquier circuito de corriente alterna aparecen estos tres parámetros: inductancia, capacitancia y resistencia en una variedad de combinaciones.

En circuitos puramente resistivos la tensión está en fase con la corriente. Algunos ejemplos de estos artefactos son lámparas incandescentes, planchas, estufas eléctricas, etc. en los cuales la totalidad de la energía se transforma en energía lumínica o calorífica, algo que no sucede en los circuitos inductivos o capacitivos donde la tensión y la corriente están desfasadas 90° una respecto a la otra.

En un circuito puramente inductivo la corriente está atrasada 90° respecto de la tensión mientras que en un circuito puramente capacitivo la corriente va adelantada 90° respecto de la tensión. Estos circuitos además del consumo de la potencia activa que suministra la fuente consumen también una potencia reactiva la cual no proporciona ningún tipo de trabajo útil, sino corriente que circula por la red (con sus pérdidas) sin entregar potencia a la carga.

La potencia activa es aquella que consumen los aparatos conectados a la red eléctrica, la cual se transforma en forma de calor o trabajo y se corresponde con la energía útil o potencia

activa o simplemente potencia. Sin embargo, la potencia reactiva es aquella que aparece en motores, transformadores y en general todos los dispositivos eléctricos que hacen uso del efecto de un campo electromagnético, donde además de requerir una potencia activa para efectuar el trabajo útil es necesaria una potencia reactiva para la generación del campo magnético, la cual no produce ningún trabajo. La potencia reactiva está 90° desfasada de la potencia activa.

Por último, la potencia aparente, es aquella que resulta de considerar la tensión aplicada al consumo de la corriente que éste demanda, o lo que es lo mismo la suma de los vectores de la potencia activa y la potencia reactiva (ver figura 1.11).

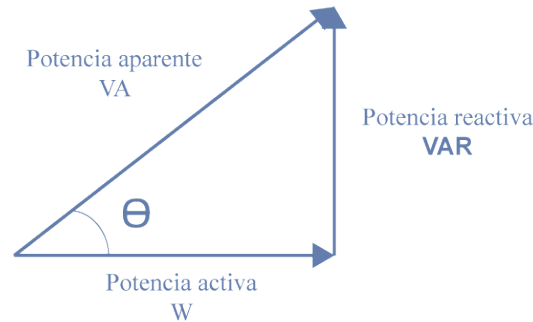


Figura 1.11: Triángulo de Potencias

En concreto, el Factor de potencia es la relación entre las potencias activa y aparente si las corrientes y es son señales sinusoidales. Si estas son señales perfectamente sinusoidales el factor de potencia será igual al $\cos \theta$, o bien, al coseno del ángulo que forman los fasores de la corriente y la tensión, designándose en este caso como $\cos \theta$ el valor de dicho ángulo.

Hasta los años 70, las cargas conectadas a la red eléctrica eran mayoritariamente lineales, es decir, demandaban corriente en forma de onda sinusoidal. Esto producía que el factor de potencia pudiese ser definido por la siguiente ecuación:

$$PF = \frac{\text{PotenciaActiva}}{\text{PotenciaAparente}} = \frac{V_{ef} I_{ef} \cos \theta}{V_{ef} I_{ef}} = \cos \theta \quad (1.2)$$

A partir de ese momento, el número de cargas no lineales conectadas a la red ha ido aumentando constantemente. Debido a que las cargas no lineales demandan corriente no sinusoidales se produce la introducción de armónicos en la red eléctrica, y por lo tanto, será necesario la obtención de una nueva ecuación de factor de potencia:

$$PF = \frac{\text{PotenciaActiva}}{\text{PotenciaAparente}} = \frac{V_{ef} I_{ef1} \cos \theta}{V_{ef} I_{ef}} = \frac{I_{ef1} \cos \theta}{I_{ef}} \quad (1.3)$$

donde I_{ef1} se refiere a la corriente eficaz del primer armónico.

La proliferación del número de cargas no lineales conectadas actualmente a la red eléctrica pueden provocar un alto contenido armónico. Si no se hiciese nada para corregir este comportamiento, se podrían llegar a dañar los equipos conectados a la red. Para tratar de solucionar este problema en la medida de lo posible, distintas organizaciones internacionales han elaborado una normativa destinada a reducir el nivel de armónicos introducidos en la red (Norma EN61000-3-2) [22]. Es por ello que las técnicas de Corrección del Factor de Potencia han experimentado un notable desarrollo.

La norma EN 61000-3-2 abarca todos aquellos equipos electrónicos y eléctricos que se conecten a la red pública de baja tensión y cuya corriente es menor o igual que 16 A. El objetivo es

limitar el contenido armónico de la corriente absorbida de la red para lo cual clasifica los equipos en cuatro clases y aplica límites diferentes en función de la clase a la que pertenezca el aparato e incluso de las características de potencia del equipo.

Soluciones para la Corrección de Factor de Potencia

Tradicionalmente, para realizar la conversión alterna/continua se empleaba un circuito compuesto por un puente de diodos con un condensador de filtro ya que se trataba de un circuito sencillo, robusto y barato (ver figura 1.12). Debido a que la forma de onda de la corriente absorbida de la red en este tipo de rectificadores se compone de picos de corta duración y elevado valor, tanto positivos como negativos, y que el condensador presenta una tensión continua de valor muy próximo a la tensión de pico de red, sólo se producirá circulación de corriente cuando el valor instantáneo de la tensión de entrada sea mayor que la tensión en el condensador.

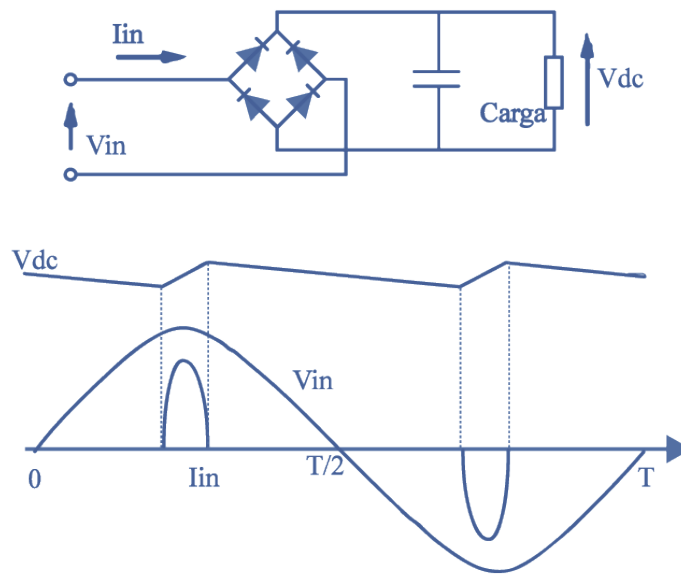


Figura 1.12: Diagrama Rectificador

Idealmente, para evitar la circulación de armónicos por la red, la corriente absorbida por los equipos debería ser sinusoidal y para evitar componentes reactivas, la tensión y la corriente debería estar en fase. En el caso del puente de diodos con condensador de filtro la corriente absorbida de la red dista mucho de ser sinusoidal como se muestra en la figura 1.12, por tanto, presenta un contenido armónico muy elevado.

Para cumplir con la Norma EN61000-3-2 [22] no siempre es necesario contar con una corriente de entrada sinusoidal, sino que es posible la utilización de un gran número de circuitos que podemos clasificar en pasivos y activos.

Inicialmente la corrección del factor de potencia utilizó circuitos pasivos los cuales tenían como objetivo principal la corrección del desplazamiento existente entre la tensión y corriente en la red eléctrica. Con el paso del tiempo, además del filtrado armónico producido por la corrección de desplazamiento surgen necesidades adicionales como el control de la tensión y potencia de salida los cuales requieren de la utilización de circuitos activos.

Soluciones Pasivas

Las soluciones pasivas (ver figura 1.13) están basadas en la utilización de elementos pasivos los cuales suavizan la corriente de entrada. Estas soluciones añaden elementos reactivos, bobinas

y condensadores, al rectificador para que de esta manera el contenido armónico quede dentro de los límites establecidos por la norma.

Las soluciones pasivas se caracterizan por ser soluciones simples, robustas y baratas, pero presentan el problema de realizarse para unas condiciones de carga determinadas lo que provoca que fuera de estas condiciones la corrección ya no será tan efectiva.

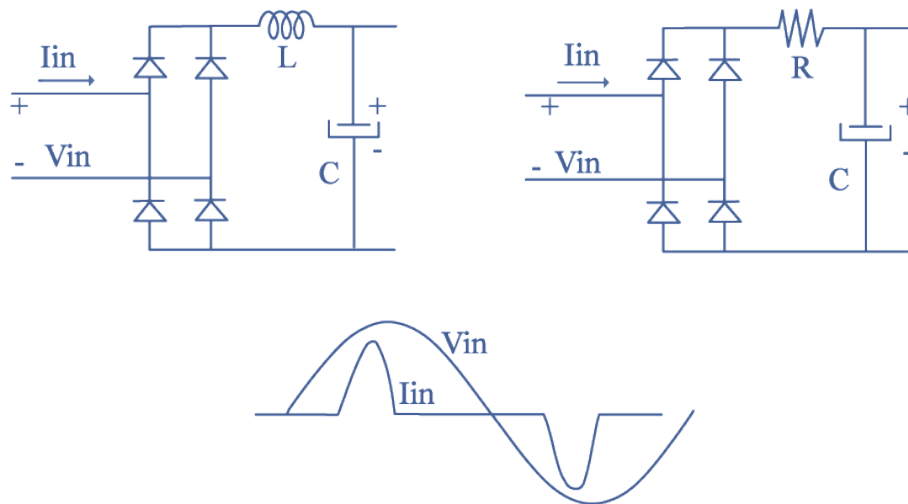


Figura 1.13: Circuitos Pasivos

Soluciones Activas

Las soluciones activas son circuitos basados en convertidores DC/DC que consiguen obtener de la red una corriente sinusoidal o cuasi-sinusoidal lo que provoca que nos encontremos ante soluciones más costosas y menos robustas que las pasivas.

Dentro de las soluciones activas es posible encontrar diferentes categorías aunque en esta ocasión serán detallados los convertidores AC/DC de dos etapas.

Como se puede observar en la figura 1.14, en los convertidores de dos etapas existe un etapa dedicada a la corrección de factor de potencia y una etapa de conversión DC/DC. En la etapa de corrección de factor de potencia conviven dos lazos, mientras uno obliga a la corriente de entrada a adoptar una forma idéntica a la de la tensión de entrada, así como a permanecer en fase con ella, el segundo lazo regula la tensión en el condensador de almacenamiento. Por último, la segunda etapa, cuenta con un único lazo encargado de la regulación de la tensión de salida.

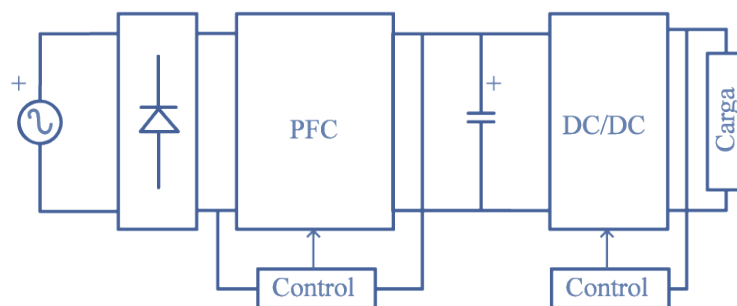


Figura 1.14: Convertidor AC/DC de Doble Etapa

A pesar de las buenas características que presenta el convertidor de dos etapas su principal

inconveniente es la complejidad puesto que es necesario llevar a cabo el diseño de dos convertidores completos.

Ventajas del Control del Factor de Potencia

La importancia de contar con un buen corrector de factor de potencia, es decir, un factor de potencia elevado, se puede entender al observar las siguientes características:

- Reducción de las pérdidas de energía que derivan en una reducción de la factura eléctrica.
- Reducción de la potencia aparente.
- Aumento en el nivel de voltaje en la red de distribución.
- Aumento de la vida útil de las instalaciones.

Aunque también existen desventajas, tales como un mayor coste y complejidad del convertidor y una menor fiabilidad del sistema.

1.3. Objetivos del proyecto

La motivación del presente proyecto es la regulación de la tensión de salida de un convertidor conmutado a través de sistemas de control digital en lazo cerrado.

La regulación se realizará mediante la placa de desarrollo proporcionada en el “Actel Fusion Starter Kit” donde se incluye una FPGA que además de contar con módulo digital cuenta con un módulo analógico. Destacar que la placa de desarrollo contiene un único ADC de varios canales, por lo que para una correcta regulación dentro de los límites del convertidor utilizado, será necesario prestar especial importancia a las frecuencias de muestreo cuando se mida más de una señal.

En el capítulo 2 es posible encontrar una descripción a alto nivel de la placa de desarrollo incluida en “Actel Fusion Starter Kit”, una descripción detallada de los módulos utilizados, y por último, una descripción de las herramientas software empleadas en el presente proyecto.

Al igual que se describen las especificaciones de la placa de desarrollo y las herramientas software utilizadas, en el capítulo 3 se incluye la descripción del convertidor conmutado sobre el que se realiza la regulación de la tensión de salida. En concreto, el presente proyecto ha empleado un convertidor boost elevador por medio del cual es posible obtener una tensión de salida superior a la de entrada mediante la utilización de los diferentes estados del pulsador (on/off) que aparece en la figura 1.15.

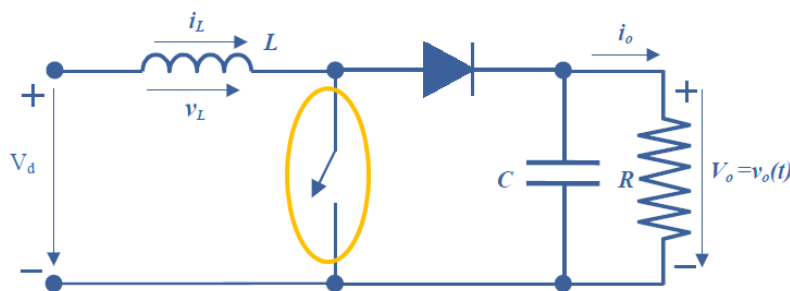


Figura 1.15: Boost Elevador

A continuación, en el capítulo 4, se detallan las acciones teóricas llevadas a cabo para la regulación de la tensión de salida sobre el convertidor, es decir, la obtención de las funciones de transferencia mediante la herramienta Matlab y su posterior implementación en lenguaje VHDL.

Mientras que el capítulo 5 se incluyen las simulaciones llevadas a cabo por medio de la herramienta ModelSim para asegurar la correcta implementación del sistema de control, es en el capítulo 6 donde se comprueba su funcionamiento a través de pruebas experimentales.

Finalmente, en el capítulo 7 se incluyen las conclusiones del presente proyecto fin de carrera.

2

Tecnología utilizada

Aunque en la actualidad existen múltiples estudios relacionados con el control digital de convertidores conmutados, el presente proyecto tiene como particularidad la realización de este tipo de control utilizando una placa de desarrollo hardware capaz de trabajar con señales mixtas analógico/digital.

La placa de desarrollo hardware utilizada para el desarrollo de este proyecto fin de carrera es la incluida en uno de los kits de FPGAs de uno de los principales fabricantes, “*Fusion Starter Kit*” de Microsemi. Este kit de evaluación cuenta con las ventajas de las FPGAs con señales mixtas, es decir, un único dispositivo que combina lógica programable, RAM, flash y un bloque analógico.

En este capítulo, en primer lugar, se llevará a cabo la descripción de la placa de desarrollo hardware y sus principales elementos. Finalmente se describirán las herramientas software utilizadas en el presente proyecto: Matlab, ModelSim y el conjunto de herramientas incluidas en Libero IDE.

2.1. “*Fusion Starter Kit*”

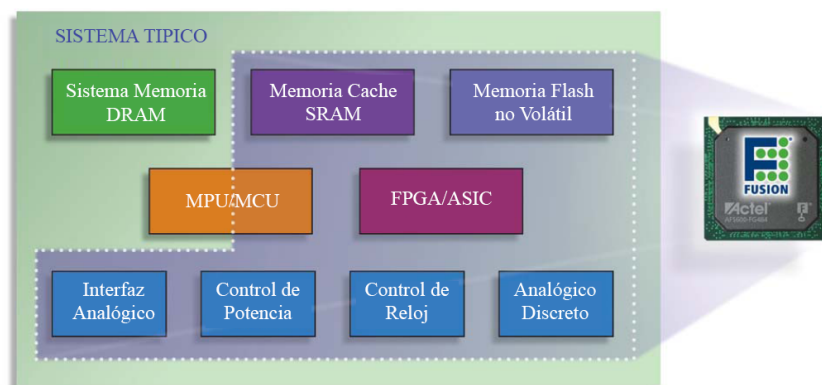


Figura 2.1: Sistema Típico

Como se puede observar en la figura 2.1, la tecnología *Fusion* utilizada en el presente proyecto es capaz de proporcionar señales mixtas, robustas y flexibles, con un alto rendimiento. En concreto, la FPGA incluida en “*Fusion Starter Kit*” [23] integra memoria flash, funcionalidades analógicas, SRAM no volátil y lógica programable. Todo ello permite el diseño de sistemas totalmente personalizables y de sencillo uso.

Fusion incorpora la estructura de eficacia probada de las FPGAs de Microsemi. Concretamente, esta tecnología posee una circuitería de señal mixta con miles de compuertas que permiten implementar funciones como el manejo de potencia, generadores de reloj, controladores de motores y cargadores de batería inteligentes, además de las aplicaciones típicas de las FPGAs.

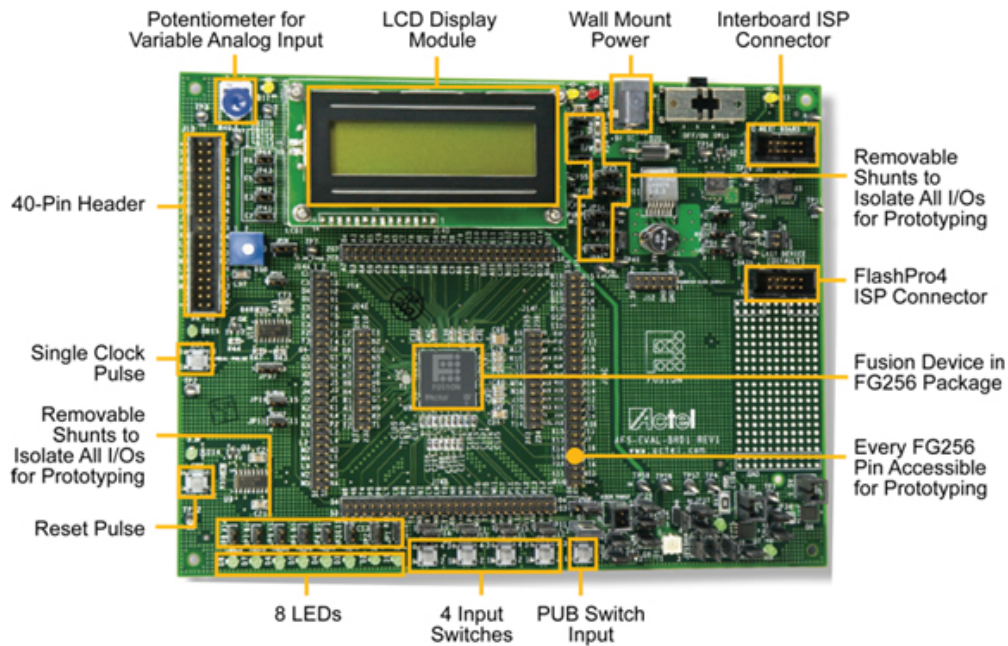


Figura 2.2: Fusion Starter Kit

Tal y como se observa en la figura 2.2, dentro del “*Fusion Starter Kit*” utilizado para el presente proyecto fin de carrera es posible encontrar:

- FPGA AFS600 con encapsulado FG256
 - 600.000 puertas, 512 kB de memoria flash y 108 kB de SRAM
 - Periféricos entre los que se encuentran temporizadores, Conversor Analógico a Digital, Conversor Digital a Analógico y recursos analógicos adicionales.
- Regulador de tensión.
- Pines para la conexión de tarjetas externas que permiten la utilización de sistemas desarrolladas por Microsemi, clientes o terceras partes.
 - Cabezales de programación In System Programming (ISP) y Joint Test Action Group (JTAG).
 - Oscilador RC, circuito capaz de producir una señal periódica a la salida sin tener ninguna entrada periódica.
 - Oscilador de Cristal, oscilador armónico (salida senoidal) cuya frecuencia está determinada por un cristal de cuarzo.
 - Lazos de seguimiento de fase capaces de distribuir, manipular y generar la señal de reloj.

- Pantalla de cristal líquido.
- LEDs multicolor.

Para mayor información sobre la placa de desarrollo utilizada acudir al anexo 7.2.

2.1.1. Bloque Digital

Dentro de los elementos incluidos en el bloque digital de la placa de desarrollo utilizada [24], el presente proyecto ha utilizado:

- Un oscilador RC con una frecuencia de trabajo de 100 MHz.
- Divisor de frecuencia de Reloj para obtener frecuencia de inicialización del ADC (10 MHz).
- Pines de entrada y salida (I/O) que han permitido la validación de los resultados con la ayuda de herramientas de visualización.
- Pulsadores para la validación de resultados.
- Memoria flash para la carga de los archivos de configuración del bloque analógico.
- Módulo JTAG (IEEE 1532) para la programación de la placa de desarrollo.

2.1.2. Bloque Analógico de la FPGA AFS600

Aunque el bloque analógico incluido dentro de la FPGA AFS600 [25] [26] incluye los elementos listados a continuación, el presente proyecto únicamente ha hecho uso del ADC (ver apartado 2.1.2) utilizando una resolución de 12 bits:

- Analog to Digital Converter (ADC) con frecuencia de muestreo de hasta 600 KHz.
- Digital to Analog Converter (DAC) con frecuencia de muestreo de hasta 600 KHz.
- Resoluciones de 8,10 ó 12 bits.
- Prescaler automático que permite entradas de $-11,5$ V hasta $+14,4$ V.
- Monitor de temperatura.
- Potenciómetro.
- Entrada de referencia de tensión externa.

El bloque analógico introduce una estructura analógica Quad I/O en la cual cada Quad analógico consta de tres entradas analógicas y un gate driver. La configuración del quad analógico, la resolución del ADC, la secuencia de muestreo de canales y la tasa de muestreo pueden ser programadas e implementadas a través del software integrado en Libero IDE [27], herramienta de Microsemi para la gestión de todo el flujo del diseño, síntesis, simulación y análisis, que será descrita posteriormente. El software utilizado para la configuración del bloque analógico se denomina Analog System Builder (ASB), y al igual que Libero IDE será descrito en el apartado herramientas del presente capítulo.

Es necesario tener en cuenta que la carga de los archivos de configuración del bloque analógico a la FPGA se realiza mediante una memoria flash, en este caso en concreto denominada **Flash Memory System Builder (FMSB)** [8]. La recomendación de Microsemi para la utilización conjunta del ASB y FMSB es que ambos bloques deben contar con la misma frecuencia de inicialización, la cual en ningún caso deberá sobrepasar los 10 MHz.

Como se ha indicado en el apartado 2.1.1, el presente proyecto utilizará una frecuencia superior a 10 MHz. Por este motivo, se ha incluido un divisor de frecuencia capaz de obtener la frecuencia máxima de inicialización a partir de la frecuencia del oscilador.

Como se muestra en la figura 2.3, a través de un multiplexor será posible la selección de la frecuencia de reloj de inicialización (SLOWCLK) o la frecuencia de reloj de funcionamiento (FASTCLK).

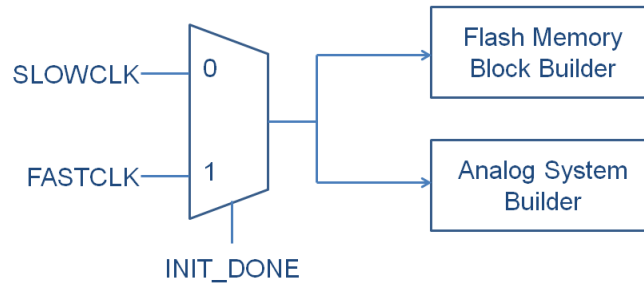


Figura 2.3: Reloj

Para que no se produzcan fallos, también denominados glitches [29], al llevar a cabo la conmutación de frecuencias una vez finalizada la etapa de configuración, se ha incluido el multiplexor de Microsemi diseñado para tal fin, No Glitch-MUX. Este componente será el encargado de la conmutación de frecuencia a partir de la señal de control la cual será activada una vez haya finalizado la fase de configuración.

ADC - Conversor Analógico Digital

El presente proyecto ha utilizado el ADC incluido en el bloque analógico para la captura de muestras de tensión sobre un convertidor. La función del ADC será la captura de muestras discretas de un voltaje analógico y continuo obteniendo así una representación binaria y discreta de una determinada señal.

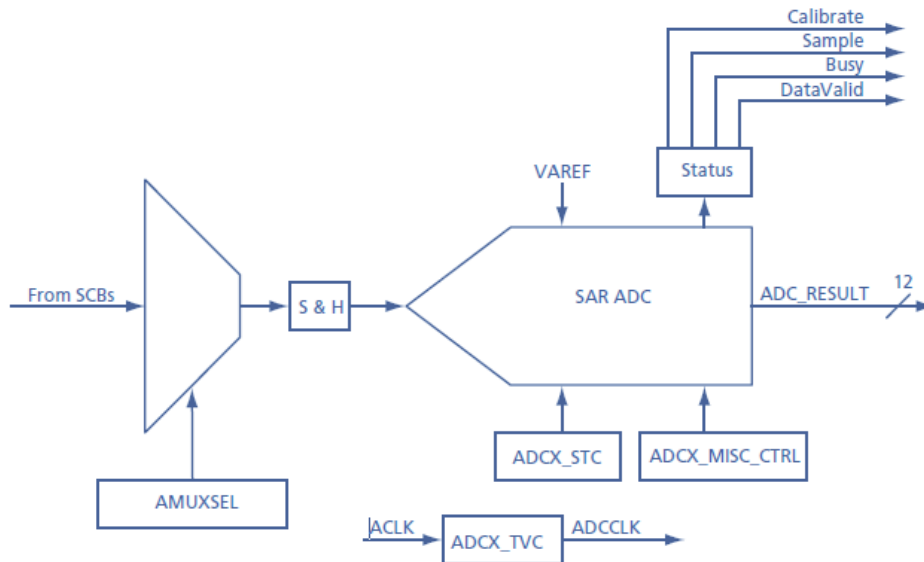


Figura 2.4: Diagrama Bloques ADC

De manera simplificada el ADC queda definido por lo mostrado en la figura 2.4. Como se puede observar, para la utilización del ADC se necesitará la fijación de una tensión de referencia

(VAREF) que podrá ser directamente la incluida por defecto (2,56 V) o una referencia externa. También será necesario la definición de la resolución de la salida (ADCRESULTS).

Entre las salidas del ADC se encuentran unas señales de estado que proporcionan información sobre el estado del ADC: Calibrate, Sample, Busy y DataValid.

Como también se aprecia en la figura 2.4, el convertidor utilizado por este ADC se encuentra dentro del grupo de los denominados convertidores de aproximaciones sucesivas, Successive Approximation Register (SAR), los cuales se basan en el uso de un algoritmo de búsqueda por aproximaciones sucesivas aplicado sobre un convertidor digital analógico, DAC, y un comparador, para encontrar, tras varias iteraciones, el código digital de N bits que mejor representa la señal analógica de entrada.

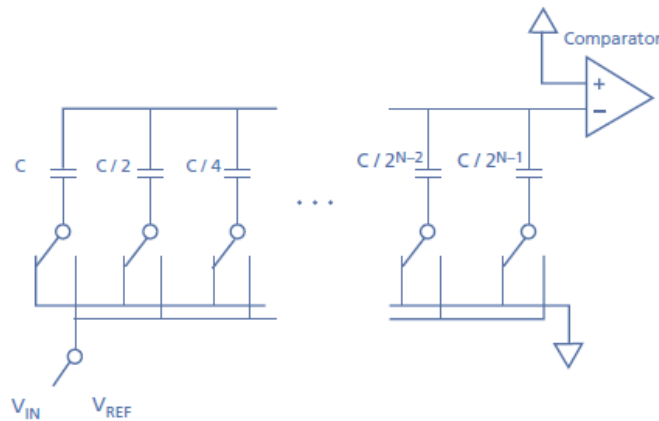


Figura 2.5: Diagrama SAR del ADC

En concreto, el ADC integrado en la FPGA incluida en *Fusion Starter Kit* está formado por condensadores en paralelo activados a través de interruptores. Antes de comenzar la conversión analógico a digital, todos los condensadores son descargados para posteriormente aplicar sobre estos la tensión analógica de entrada durante un tiempo de adquisición denominado t_{sample} . Una vez finalizado el tiempo de adquisición, la tensión de entrada es aplicada al comparador donde permanecerá almacenada.

Para determinar el primer bit más significativo, Most Significant Bit (MSB), al primer condensador, C , se le aplica la tensión de referencia, lo que produce que a la entrada del comparador nos encontremos con el siguiente voltaje:

$$\text{Voltaje a la entrada del comparador} = -V_{in} + \frac{V_{ref}}{2} \quad (2.1)$$

en caso de que V_{in} sea superior a $\frac{V_{ref}}{2}$, el MSB, es decir, la salida del comparador, será igual a 1, en caso contrario, será 0.

Si la salida del comparador fue 0, C permanecerá conectado a V_{ref} , en caso contrario será conectado a tierra. Independientemente, a continuación $\frac{C}{2}$ será conectado a la tensión de referencia para determinar así el siguiente bit más significativo. Esta operación se repetirá hasta obtener el array de 8, 10 ó 12 bits con el binario equivalente a la tensión de entrada.

Como se puede observar en la figura 2.6 la entrada analógica del ADC cuenta con un multiplexor 32:1. Mientras que la primera y última entrada a este multiplexor se corresponden con una tensión continua de 1,5 V denominada V_{cc} y un monitor de temperatura interna, las restantes entradas provienen de 10 quads. Cada uno de estos quads contiene tres señales de salida que serán canales de tensión, corriente y temperatura de entrada al multiplexor.

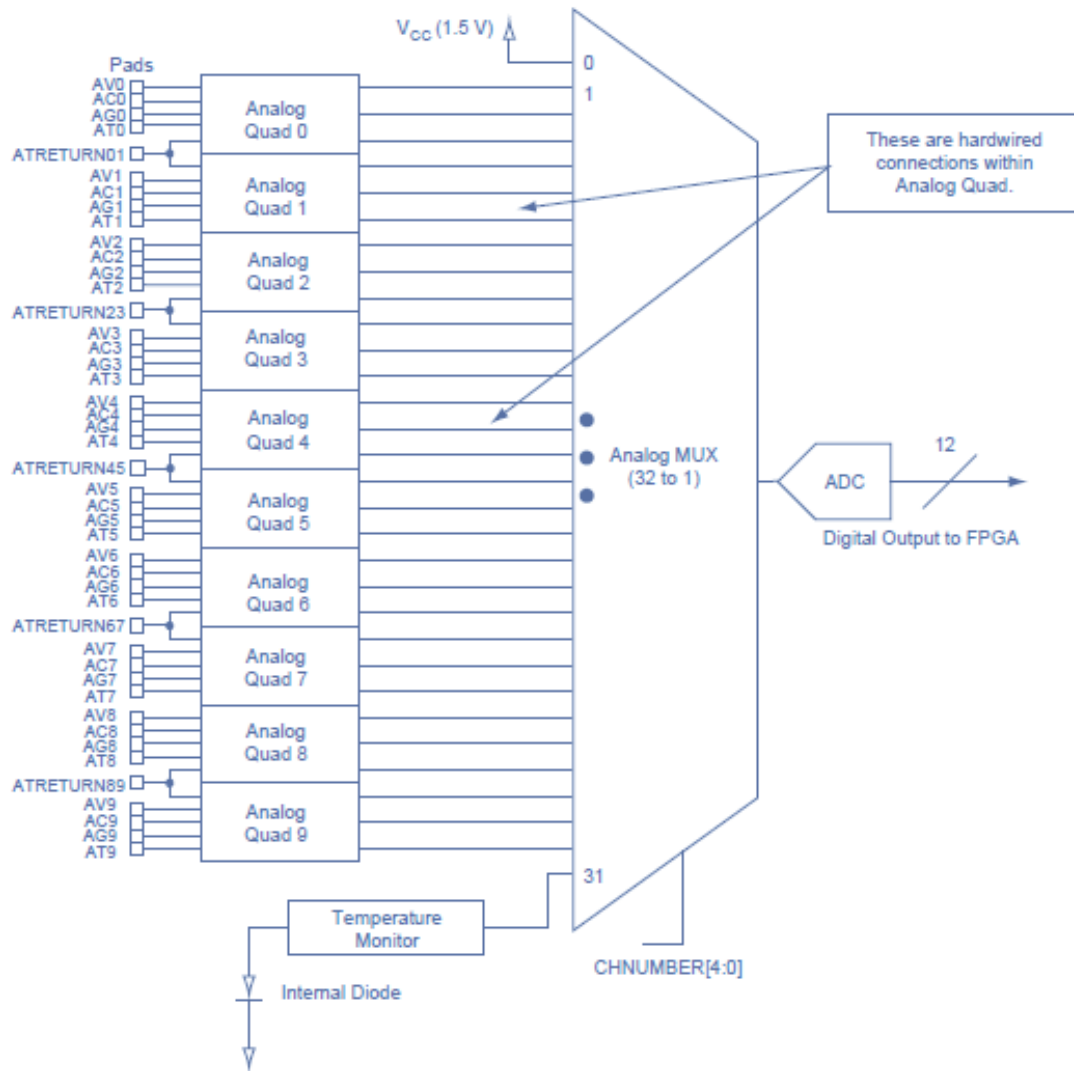


Figura 2.6: Diagrama de bloques del multiplexor de entrada del ADC

Entre las características más importantes del ADC se encuentra el ancho de banda, es decir, el número máximo de conversiones por segundo que es capaz de realizar. Para poder determinar el ancho de banda es necesario conocer los tiempos de adquisición y conversión en cada diseño.

Antes de mostrar las ecuaciones a partir de las cuales es posible obtener los tiempos de adquisición y conversión es necesario conocer el periodo de reloj del DAC (adc clock period), el cual se encuentra definido por:

$$adc\ clock\ period = \frac{1}{ADCCLK} \quad (2.2)$$

donde:

$$ADCCLK = \frac{SYSCLK}{4(1 + TVC)} \quad (2.3)$$

$SYSCLK$ es la frecuencia del sistema y TVC es el contenido del valor de registro $ADCx_TVC$ utilizado para asegurar que la frecuencia de reloj ADC no exceda 10 MHz ni caiga por debajo de 0,5 MHz.

A continuación, se procede a la definición de los tiempos de adquisición y conversión mencionados con anterioridad.

Tiempo de adquisición

El tiempo de adquisición especifica el tiempo de carga del condensador por la señal analógica de entrada y se encuentra definido por la siguiente ecuación:

$$t_sample = (2 + STC)(adc\ clock\ period) \quad (2.4)$$

donde STC es el control del tiempo de adquisición y puede tomar valores entre 0 y 255.

Tiempo de Conversión

Durante el proceso de conversión se experimentan tres fases: muestreo, distribución y post-calibración. El tiempo de cada una de estas fases debe tenerse en cuenta para el cálculo del tiempo de conversión, definido, como el tiempo total necesario para convertir una entrada analógica en una salida digital.

$$t_conv = t_sync_read + t_sample + t_distrib + t_post_cal + t_sync_write \quad (2.5)$$

Durante la fase de distribución el ADC obtiene el valor digital equivalente al que se encuentra en el condensador. Este tiempo de distribución depende de la resolución, es decir, el número de bits de la señal de salida, y su función viene dada por:

$$t_distrib = (Resolution)(adc\ clock\ period) \quad (2.6)$$

La fase de post-calibración es opcional. Actel recomienda activar esta fase para compensar las pérdidas y los efectos de la temperatura. EL tiempo de post-calibración es igual a:

$$t_post_cal = 2(adc\ clock\ period) \quad (2.7)$$

Por último, t_sync_read y t_sync_write son el tiempo transcurrido hasta obtener la señal de entrada o la de salida, esta última, una vez realizada la conversión. Ambos tiempos vienen dados por:

$$t_sync_read = t_sync_write = sys_clk_period \quad (2.8)$$

donde sys_clk_period es el periodo de la interfaz del reloj y oscila entre 10 ns y 250 ns.

Se debe tener en cuenta que, si la señal cambia más rápidamente que la frecuencia de muestreo, la información se pierde durante la conversión. Basándose en el teorema de muestreo de Nyquist-Shannon, la tasa mínima de muestreo debe ser al menos el doble de la frecuencia de la componente de frecuencia más alta de la señal objetivo (Frecuencia de Nyquist), proporcionando así suficientes muestras para representar adecuadamente la señal de entrada.

2.2. Herramientas Software Utilizadas

A continuación se describen cada una de las herramientas software utilizadas para la realización del presente proyecto.

2.2.1. Matlab

La primera de las herramientas utilizadas ha sido *MatLab* [30]. Se trata de un entorno informático de análisis numérico y representación gráfica de fácil manejo que permite crear funciones propias y programas especiales en código *MatLab*.

MatLab permite el desarrollo de aplicaciones para la resolución de problemas de múltiples áreas. Entre sus utilidades se encuentran:

- Cálculo matricial y álgebra lineal.
- Polinomios e interpolación.
- Regresión y ajuste de funciones.
- Ecuaciones diferenciales ordinarias.
- Integración.
- Funciones y gráficos en dos y tres dimensiones.

Además, *Matlab* incluye una herramienta gráfica denominada *Sisotool* que facilita en gran medida el diseño de controladores [31]. Es por ello, que el presente proyecto ha utilizado esta herramienta para la obtención de las funciones de transferencia utilizadas para cada uno de los lazos.

Sisotool

Sisotool es una herramienta que trabaja de forma gráfica usando el método del lugar geométrico de las raíces, lo que permite un rápido análisis de las mismas al detectar la situación de los polos y los ceros de un sistema [32].

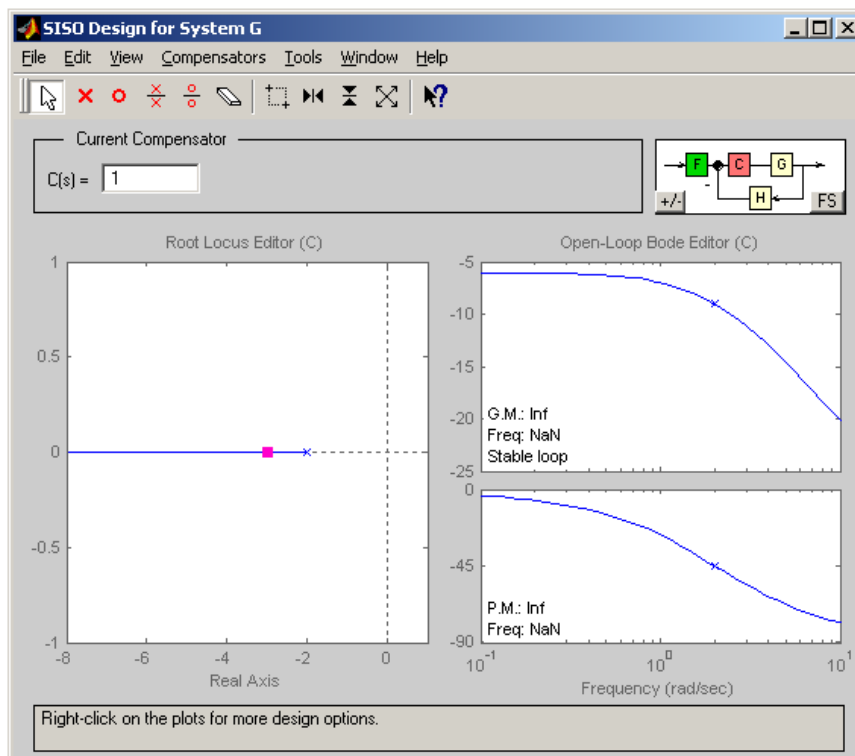


Figura 2.7: Sisotool

En la figura 2.7 se muestra la ventana de *Sisotool*. A la izquierda, el lugar de las raíces del sistema a analizar cuando realimentamos el sistema bajo estudio. A la derecha, el diagrama de Bode en lazo abierto, tanto de amplitud como de fase.

Mientras los polos y ceros del sistema en lazo abierto se muestran como \times y \circ respectivamente, los polos en lazo cerrado se muestran como \square .

2.2.2. ModelSim

Modelsim [33] es un software que administra un entorno que permite editar, compilar, simular y depurar diseños de sistemas digitales descritos en VHDL [34] y Verilog [35] entre otros. Es por ello, que tras la implementación en VHDL de las funciones de transferencia obtenidas mediante la herramienta *Matlab*, se ha utilizado *Modelsim* para la depuración y simulación de los sistemas de control implementados.

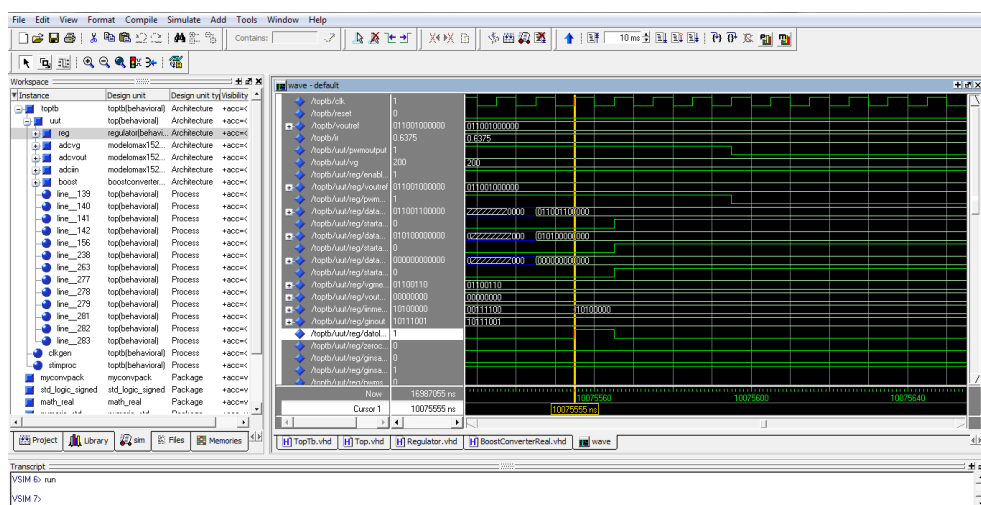


Figura 2.8: ModelSim

Como se muestra en la figura 2.8, la ventana principal de esta herramienta se encuentra dividida en tres áreas:

- Componentes: en este cuadro se muestran todos los componentes que forman parte del diseño así como las librerías utilizadas.
- Simulación o waveform: se muestra el valor de las señales añadidas durante el tiempo que se ha corrido la simulación.
- Consola: mediante línea de comando nos permite ejecutar cualquier operación que *Modelsim* sea capaz de realizar, como añadir y eliminar señales de la simulación, ejecutar, compilar

2.2.3. Libero IDE

Para apoyar a la tecnología *Fusion*, Microsemi cuenta con una serie de herramientas que ayudan a maximizar la productividad del diseño. El entorno de diseño integrado, Integrated Design Environment (IDE), permite a los diseñadores crear instancias y configurar periféricos fácilmente, establecer vínculos entre los periféricos, crear o importar bloques de construcción o diseños de referencia y realizar la verificación de hardware. Además, esta suite de herramientas también añade características de depuración de hardware/software.

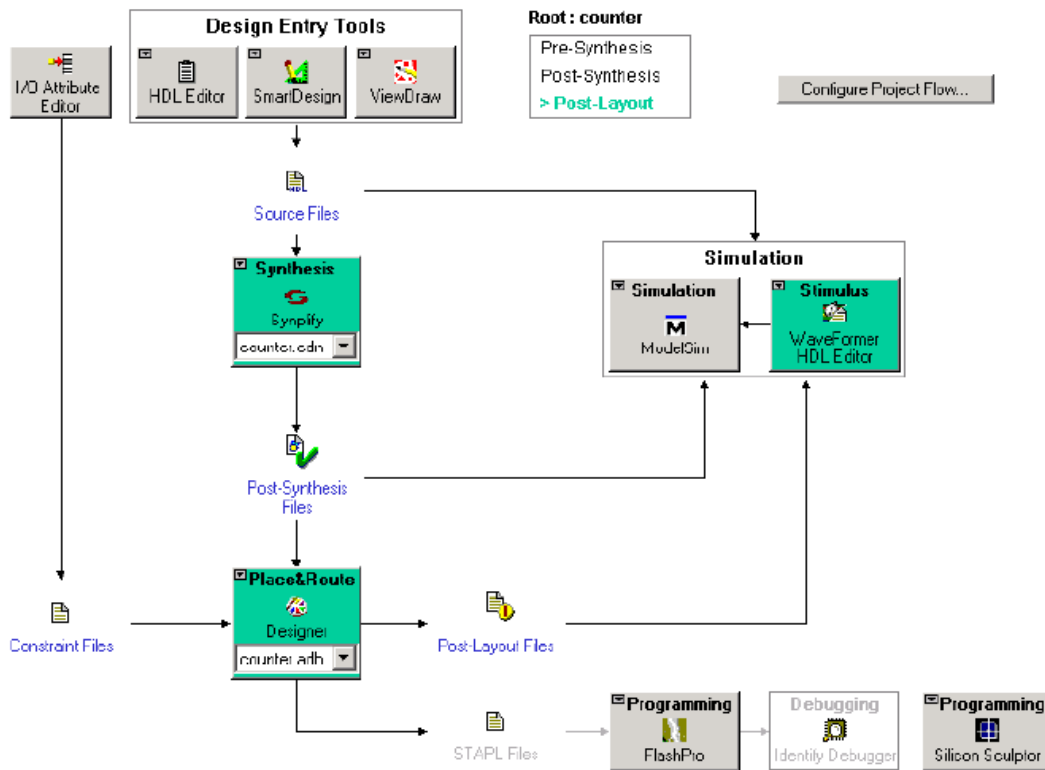


Figura 2.9: ProjectFlowWindow

En concreto, Libero IDE [27] es el software de Microsemi utilizado para la gestión de todo el flujo del diseño, síntesis, simulación y análisis. Todas las herramientas que intervienen en el proceso de diseño pueden observarse desde la *Project Flow Window* que aparece en la figura 2.9.

Dentro de *Project Flow Window* se encuentra *Smartdesign*, una herramienta que actúa de pizarra para bloques funcionales. Además de permitir el montaje e interconexión de bloques funcionales, también permite agregar componentes desde el catálogo y componentes creados por el diseñador sobre diferentes lenguajes HDL: VHDL o Verilog.

Una vez finalizado el diseño, es necesario ejecutar la herramienta de síntesis *Synplify* [36] [37], la cual lleva a cabo una síntesis de alto nivel, es decir, crea una lista de conexiones del diseño o *Netlist*.

A continuación, la herramienta *Designer* llevará a cabo el *Place&Route*. Este proceso es el encargado de asignar lugares y recursos físicos de la FPGA a la *Netlist* creada en el paso anterior. Una vez realizado el *Place&Route* se creará un archivo binario con toda la información, listo para ser descargado en la FPGA.

La programación de la FPGA “*Fusion Starter Kit*” se realiza mediante una combinación del software FlashPro [33] [34] y el hardware de programación. Para llevar a cabo la programación únicamente será necesario la conexión de la memoria flash a un PC y a la placa de destino.

Por último, *Libero IDE* también permite llevar a cabo la verificación del diseño mediante una simulación funcional través del simulador integrado, *ModelSim* de Mentor Graphics.

ASB - Analog System Builder

Como se observa en la figura 2.10 ASB [30] es una herramienta a través de la cual se lleva a cabo la configuración del bloque analógico completo de la FPGA incluida en *Fusion Starter Kit*.

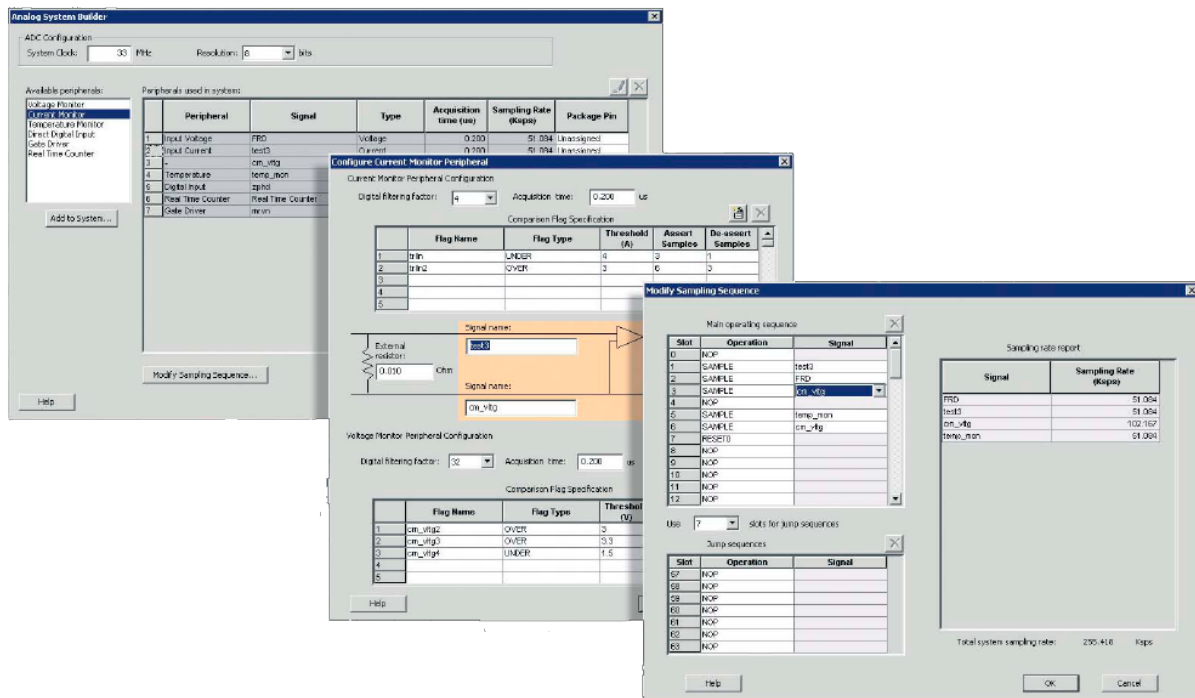


Figura 2.10: Analog System Builder

Algunas de las posibles acciones que se pueden llevar a cabo se listan a continuación:

- Elegir el número de canales de entrada analógica que se desea monitorear.
- Elegir el tipo de cada canal de entrada.
- Elegir el número de canales de salida analógica.
- Especificar la ubicación de cada canal.
- Definir el orden de muestreo requerido.

Las opciones avanzadas de ASB además de ajustar la tensión de referencia externa permite generar configuraciones personalizadas (observar figura 2.11).

En concreto, el uso de un voltaje externo de referencia, permite al ADC realizar conversiones más precisas. En caso de no introducir un voltaje de referencia, por defecto se utiliza Voltaje de Referencia (V_{ref}) igual a 2,56V.

Por otro lado, dentro de las configuraciones personalizadas del ASB se encuentran tres modalidades:

- *IP Cores for ADC data processing and sequence control.*
- *IP Cores for ADC Sequence control.*
- *ADC only.*

IP Cores for ADC data processing and sequence control permite la secuenciación, generación de banderas, promedio de datos y, en general, la gestión del ADC sin casi intervención del usuario.

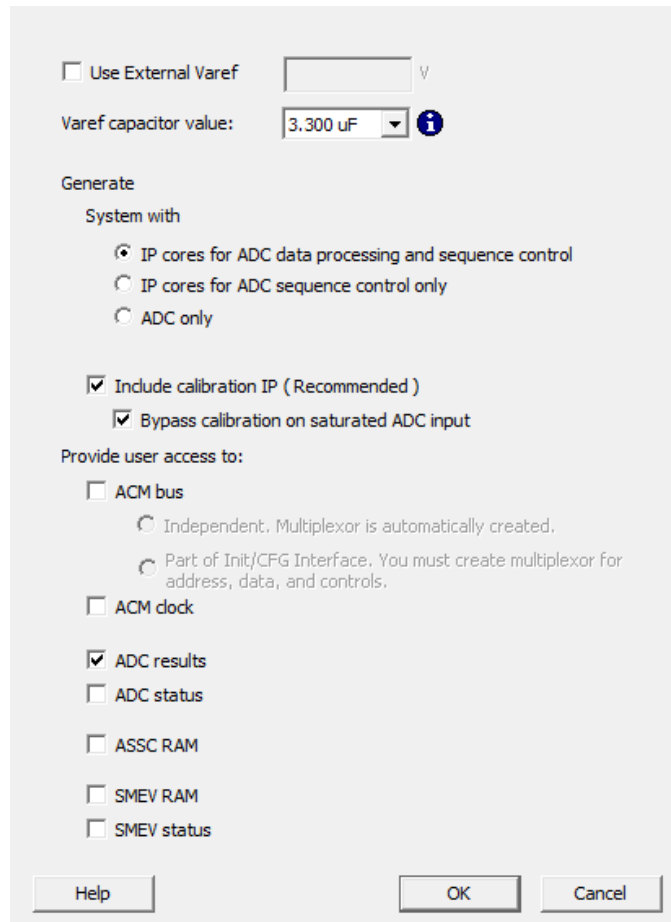


Figura 2.11: Analog system Builder Advanced Option Dialog Box

IP Cores for ADC Sequence control instancia únicamente el bloque analógico y la memoria RAM

Por último, **ADC only** omite el procesamiento de datos, la secuenciación y las funciones de gestión del ADC de manera que es el propio usuario quien debe manejarle por completo. De este modo, el usuario deberá declarar las señales de entrada listadas a continuación.

- ADCSTART. Indica el comienzo de la conversión.
- ADCRESET. Restablece y desactiva el bloque analógico.
- PWRDWN. Cuando está activo, el bloque analógico se apagará mientras la parte digital sigue en funcionamiento.
- MODE[3:0]. Este registro, además de controlar la resolución de ADC Result, controla la calibración y el apagado después de cada conversión.
- STC[7:0]. Control para la división del reloj.
- TVC[7:0]. Control del tiempo de muestreo.
- CHNUMBER. Selecciona el canal de trabajo.

Las salidas obtenidas del bloque analógico serán por tanto:

- DATAVALID. Indica la finalización de la conversión.
- CALIBRATE. Indicador de estado de la calibración.
- BUSY. Indicador de estado de la ocupación.
- ADCRESULT. Proporciona el resultado binario obtenido de la medición de la señal analógica en 8, 10 ó 12 bits.

FlashPro

Para llevar a cabo la programación de la FPGA es necesaria la utilización de la interfaz gráfica de FlashPro [38] [39]. Como se puede observar en la figura 2.12 la interfaz gráfica de usuario está compuesta por una ventana de configuración de dispositivos y una ventana de registros que muestra la información de programación, mensajes de error y mensajes de advertencia.

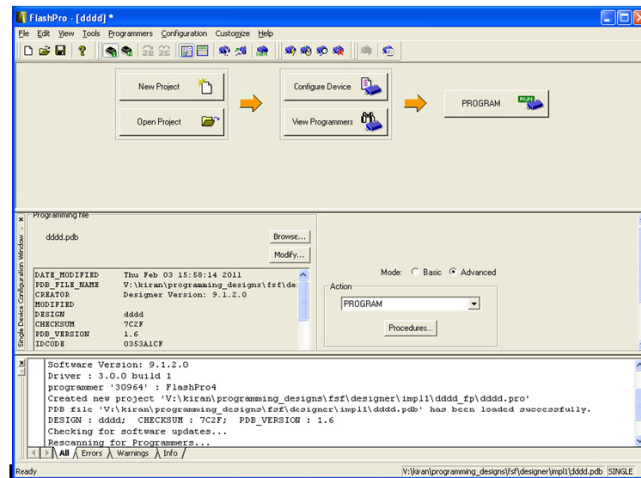


Figura 2.12: Interfaz Gráfica FlashPro

3

Convertidor Boost Elevador

El presente proyecto llevará a cabo la regulación de la tensión de salida de una fuente conmutada, en concreto, de un convertidor boost elevador. Por ello, este capítulo se centra en descripción de los estados de operación del convertidor utilizado: modo de conducción continuo y modo de conducción discontinuo, los cuales nos permitirán comprender su funcionamiento.

En último lugar, se describirán las características del boost utilizado.

3.1. Convertidor Boost Elevador

La verdadera eficacia de las fuentes conmutadas radica en el apagado y encendido del interruptor o conmutador de manera que las pérdidas obtenidas sean minimizadas en la medida de lo posible. Por ello, no es de extrañar que el principio básico de estas fuentes consista en la posibilidad de obtener estados distintos dependiendo del estado del conmutador (ver figura 3.1).

Para el caso concreto del Boost Elevador [40], donde el voltaje promedio de la salida es mayor que el voltaje promedio de la entrada, existen diferentes estados de operación:

- **Modo de Conducción Continuo (MCC).** La corriente que atraviesa la bobina varía entre unos valores máximo y mínimo, pero nunca llega a anularse.
- **Modo de Conducción Discontinuo (MCD).** La corriente que atraviesa la bobina se hace nula en un momento determinado a lo largo de un intervalo de tiempo durante el cual el interruptor está abierto.

De manera simplificada, el funcionamiento del Boost Elevador es el siguiente [41]. La energía precedente de la entrada es conducida por el elemento de conmutación para ser almacenada en la bobina. Este almacenamiento de energía únicamente se produce durante el periodo de conducción del interruptor, no existiendo transferencia de energía a la carga.

Por el contrario, la apertura del interruptor ocasiona que la energía acumulada en la bobina se transfiera a la carga. Para que la corriente disminuya la tensión en la bobina tiene que ser negativa, y por ello, la tensión de salida es mayor que la de entrada.

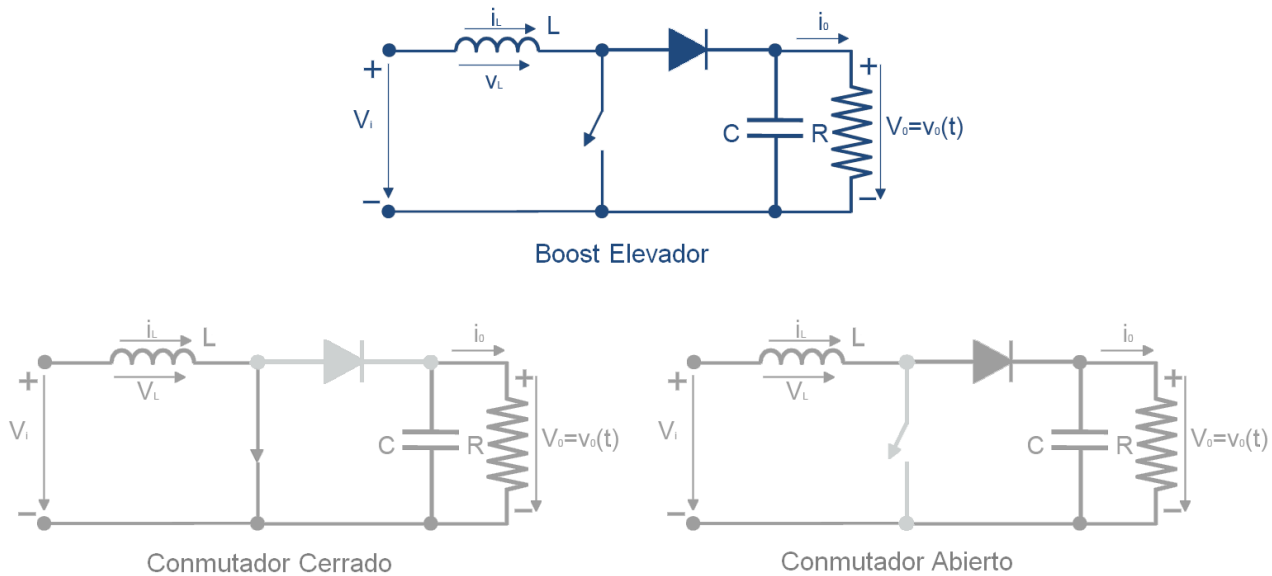


Figura 3.1: Boost Elevador

3.1.1. MCC - Modo de Conducción Continuo

El Modo de Conducción Continuo [42] se caracteriza porque la corriente que atraviesa la bobina (i_L) nunca se anula. Esto quiere decir que cuando el conmutador se encuentra cerrado, la bobina se encuentra en un estado de carga gracias a esta corriente. Cuando el conmutador permanece abierto, la corriente de la bobina se descarga a través de la carga sin llegar a vaciarse la energía acumulada en la bobina.

En la figura 3.2 se pueden observar las típicas formas de onda del Boost Elevador operando en MCC.

En la parte superior se muestra la forma de onda del conmutador. t_{on} es el periodo de tiempo que el conmutador se encuentra cerrado, por el contrario, t_{off} es el periodo de tiempo que este se encuentra abierto.

También es posible encontrar las formas de onda de los voltajes V_i , V_o , y V_L que se corresponden con los voltajes a la entrada, a la salida y en la bobina. V_s es la diferencia de potencial en el interruptor.

En último lugar se observa la forma de onda de la corriente i_L de la bobina. Esta, muestra sus dos etapas: la etapa de carga y la etapa de descarga. El rizado Δi_L es la diferencia de la corriente máxima y mínima. El valor medio de la forma de onda de i_L es i_{avg} .

Cuando el conmutador permanece cerrado, la tensión de la bobina V_L corresponde con la tensión de entrada V_i , y en consecuencia, la corriente en la bobina i_L sufre un incremento lineal positivo.

$$\frac{\Delta i_L}{\Delta t} = \frac{V_i}{L} \quad (3.1)$$

Al final de la etapa de carga en la bobina, el incremento de la corriente a través del inductor está dada por la siguiente ecuación, donde D es igual al

$$\Delta i_{Lon} = \int_0^{DT} \frac{V_i}{L} dt = \frac{V_i \cdot D \cdot t_{on}}{L} \quad (3.2)$$

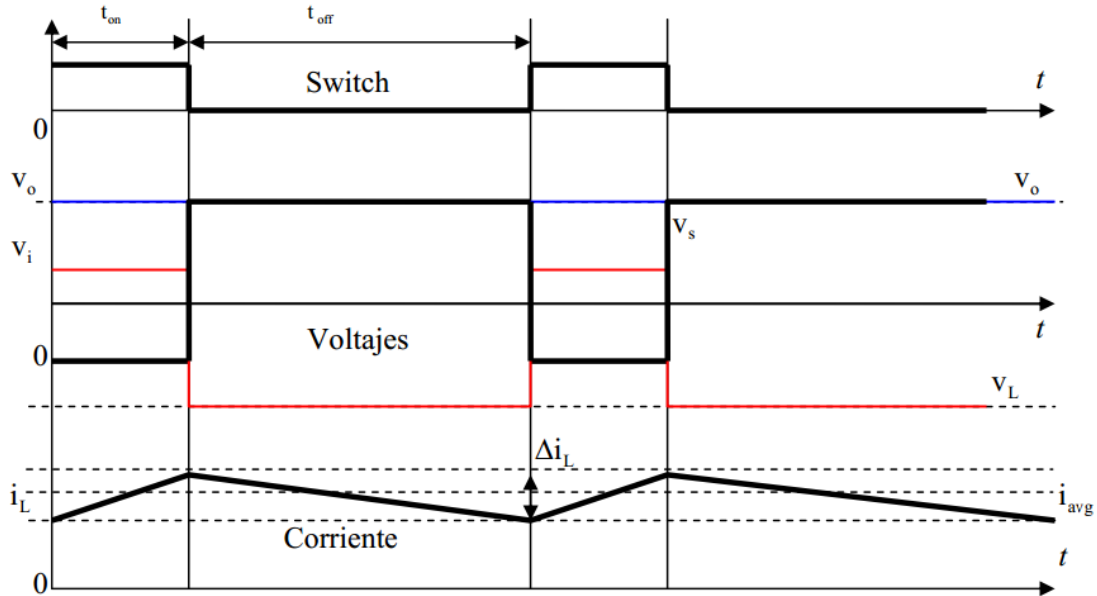


Figura 3.2: Formas de onda en MCC

Cuando el conmutador permanece abierto, la tensión en este (V_s) pasa a ser positiva e igual a la tensión de salida, y el voltaje en la bobina (V_L) pasa a ser negativo, lo que se traduce en un cambio en la polaridad de la bobina.

La corriente en la bobina (i_L) sufre un decremento lineal producido por la descarga de energía que sufre debido a la carga.

Si consideramos situaciones ideales, la no existencia de caída de tensión en el diodo y que el condensador es lo suficiente grande para mantener el voltaje de salida constante, la evolución de la corriente en la bobina puede definirse mediante:

$$V_i - V_o = L \frac{di_L}{dt} \quad (3.3)$$

$$\Delta i_{L_{off}} = \int_0^{(1-D)T} \frac{(V_i - V_o) dt}{L} = \frac{(V_i - V_o)(1-D)t_{off}}{L} \quad (3.4)$$

La cantidad de energía almacenada por cada uno de los componentes es la misma al principio y la final de cada ciclo completo de conmutación, cuando consideramos que el Boost opera en condiciones estacionarias.

Entonces la energía almacenada en la bobina es:

$$E = \frac{1}{2} L \cdot i_L^2 \quad (3.5)$$

La suma de la corriente en la bobina en la etapa de carga y descarga es entonces igual a cero:

$$\Delta i_{L_{on}} + \Delta i_{L_{off}} = 0 \quad (3.6)$$

Sustituyendo las ecuaciones 3.2 y 3.4 en 3.6 se obtiene:

$$\Delta i_{L_{on}} + \Delta i_{L_{off}} = \frac{V_i \cdot D \cdot t_{on}}{L} + \frac{(V_i - V_o)(1-D)t_{off}}{L} \quad (3.7)$$

Quedando:

$$\frac{V_o}{V_i} = \frac{1}{1 - D} \quad (3.8)$$

Finalmente:

$$D = 1 - \frac{V_i}{V_o} \quad (3.9)$$

Lo que confirma que el voltaje de salida es siempre superior al de entrada ya que D corresponde a un valor variable entre 0 y 1.

3.1.2. MCD - Modo de Conducción Discontinua

Si, por ejemplo, en régimen de funcionamiento estacionario aumentamos la resistencia de carga, disminuirá la corriente de salida del convertidor permaneciendo constante la tensión de salida. Debido a esto, disminuirá la corriente en la bobina hasta llegar el momento en el que $i_L = 0$, a este funcionamiento se le conoce como Modo de Conducción Discontinuo [42].

En la figura 3.3 se distinguen la formas de onda básicas cuando el Boost Elevador opera en modo MCD.

Las formas de ondas son las mismas que aparecen en el modo MCC a excepción del parámetro δ , el cual se corresponde con el tiempo que la bobina se encuentra descargándose hasta llegar a 0.

En esta figura se puede observar cómo el conmutador permanece cerrado un tiempo t_{on} y es δt_{off} el tiempo durante el cual la bobina se encuentra en estado de descarga.

En la figura 3.3 también podemos apreciar que la tensión de salida (V_o) se mantiene constante. El voltaje en el conmutador (V_s) se mantiene nulo en el periodo t_{on} , y cuando cambia el periodo a δt_{off} , la tensión V_s pasa a ser igual que la tensión de salida V_o . La tensión de la bobina (V_L) se mantiene positiva cuando está en fase de carga pero luego cambia su polaridad cuando para a la fase de descarga. El tiempo en que $i_L = 0$, la tensión en la bobina V_L se anula y el voltaje en el conmutador disminuye y se equipara a la tensión de entrada (V_i).

Por último, se observa cómo la onda de corriente de la bobina durante la fase de carga almacena energía en forma de corriente. Cuando el conmutador se abre, la corriente almacenada en la bobina se descarga a través de la carga hasta llegar a cero, para permanecer así hasta el final del ciclo.

Durante MCD, la corriente es máxima en la bobina ($i_{L_{Max}}$) cuando el conmutador se encuentra cerrado y queda definida por:

$$i_{L_{Max}} = \frac{V_i \cdot D \cdot t_{on}}{L} \quad (3.10)$$

Sin embargo, cuando el conmutador se encuentra abierto:

$$i_{L_{Max}} + \frac{(V_i - V_o) \cdot \delta D \cdot t_{on}}{L} = 0 \quad (3.11)$$

Podemos apreciar que después de δT la corriente en la bobina se vuelve 0. Si unamos las ecuaciones 3.10 y 3.11 y simplificamos se obtiene:

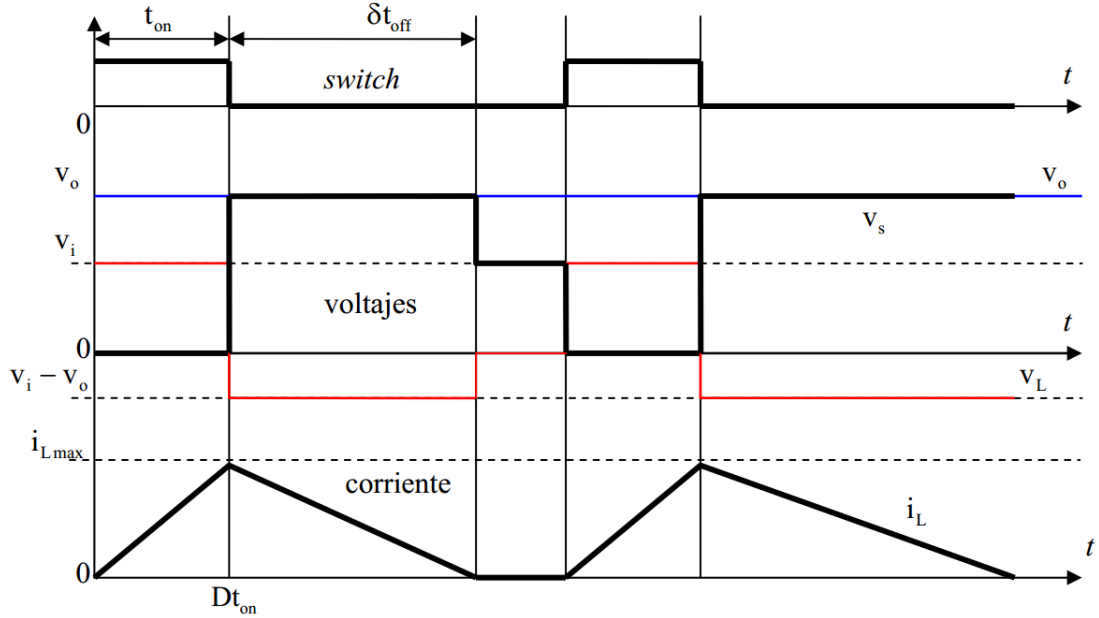


Figura 3.3: Formas de onda en MCD

$$\delta = \frac{V_i \cdot D}{V_o - V_i} \quad (3.12)$$

La corriente de salida (i_o) es igual a la corriente que pasa por el diodo, y esto es igual, al área del triángulo rectángulo formado entre i_{LMax} y δ en su fase de descarga. Por tanto, la corriente de salida tiene la siguiente expresión:

$$i_o = \frac{i_{LMax} \delta}{2} \quad (3.13)$$

Si ahora se sustituye 3.10 y 3.12 en 3.13 se obtiene:

$$i_o = \frac{V_i \cdot D \cdot t_{on}}{2L} \frac{V_i \cdot D}{V_o - V_i} = \frac{V_i^2 \cdot D^2 \cdot t_{on}}{2L(V_o - V_i)} \quad (3.14)$$

De modo que:

$$\frac{V_i}{V_o} = 1 + \frac{V_i \cdot D^2 \cdot t_{on}}{2L \cdot i_o} \quad (3.15)$$

Se observa entonces que la expresión de la tensión de salida es más compleja en MCD que en MCC. En MCD, la ganancia de tensión depende de t_{on} (D), de la corriente de salida (i_o), de la capacidad de la bobina (L), y del voltaje de entrada (V_i).

En cualquiera de los casos, es importante destacar que como consecuencia de la utilización de dispositivos reales no se darán situaciones ideales. Para el caso en concreto del condensador se debe tener en cuenta que este no contará con una capacidad infinita, por ello producirá un pequeño rizado en el voltaje de salida.

[44] En t_{on} la corriente por el condensador será la misma pero de signo contrario que la de salida, y en t_{off} , la corriente decrece dado que la tensión aumenta. Esto se debe a que en t_{off} ,

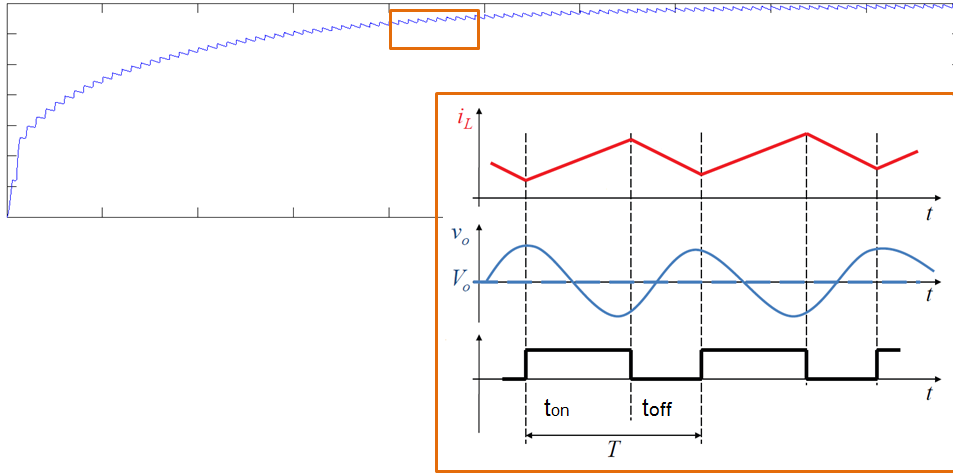


Figura 3.4: Rizado en Tensión de Salida

el condensador se carga y por tanto aumenta su tensión, y en t_{on} , el condensador se descarga produciendo así un rizado en la tensión de salida (ver figura 3.4).

El comportamiento de la tensión en la bobina es parecido al anterior. En esta se obtienen valores de tensión diferentes dependiendo del estado de t_{on} y t_{off} , lo que provoca una suave variación de la corriente, es decir, rizado durante estos intervalos.

3.2. Boost Utilizado

En la figura 3.5 se muestra el Boost Elevador utilizado en el presente proyecto [44]. Este Boost cuenta con las siguientes características:

- Tensión de entrada máxima de hasta 230 V eficaces a una frecuencia de 50 Hz.
- Tensión nominal máxima de salida de hasta 400 V.
- Potencia nominal máxima de hasta 300 W.
- Frecuencia de conmutación del MOSFET 100 kHz.
- Condensador de 100 μF el cual permite un rizado máximo en la tensión de salida 3 %.
- Bobina de 1,1 mH la cual permite un rizado máximo de corriente del 30 %.

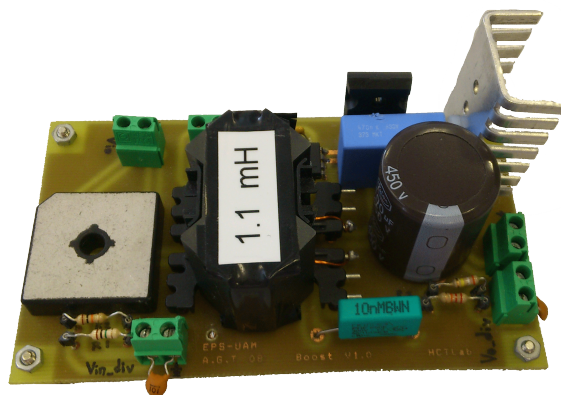


Figura 3.5: Boost Elevador

El MOSFET, además de contar una frecuencia de conmutación suficientemente grande para satisfacer las necesidades del presente proyecto, también deberá soportar las tensiones máximas con las que trabajaremos. En este sentido, el MOSFET utilizado (IRFP450) soporta una tensión máxima de 500 V, tensión que supera a la tensión máxima de salida del boost.

Además de la bobina, el condensador y el MOSFET indicados anteriormente, el Boost Elevador utilizado está compuesto por otros elementos que también deben soportar las tensiones y corrientes máximas que circularán por el convertidor [43].

Por ello, el diodo de potencia utilizado (DYV29500) también cumple con las especificaciones de tensión y corriente del sistema y tiene un tiempo de conmutación menor de 60 ns.

El puente de diodos, utilizado para rectificar la tensión de entrada (FBL2504L), también soporta 400 V de tensión de pico y una corriente de 25 A.

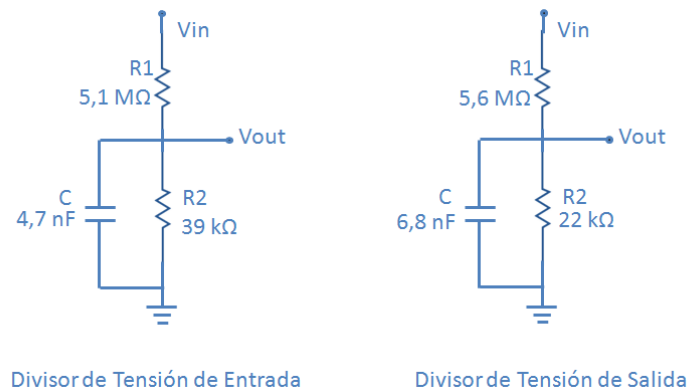


Figura 3.6: Divisores de Tensión de Entrada y Salida

Adicionalmente, el Boost utilizado tiene integrados dos divisores de tensión que serán utilizados para la medición de la tensión de entrada y la tensión de salida por la FPGA. Estos divisores se presentan necesarios como consecuencia de la limitación presente en el bloque analógico de la FPGA el cual únicamente es capaz de soportar tensión de hasta 15 V [25].

Para eliminar el ruido de alta frecuencia, a los divisores de tensión se les ha añadido un condensador a la salida que actuará como filtro. En la figura 3.6 se pueden encontrar los diseños de los divisores de tensión utilizados para las mediciones de la tensión de entrada y la tensión de salida.

Como se puede observar en la figura 3.7, también ha sido necesario un divisor adicional, no integrado en el boost, para llevar a cabo la medición de la corriente de entrada, es decir, la corriente de la bobina.

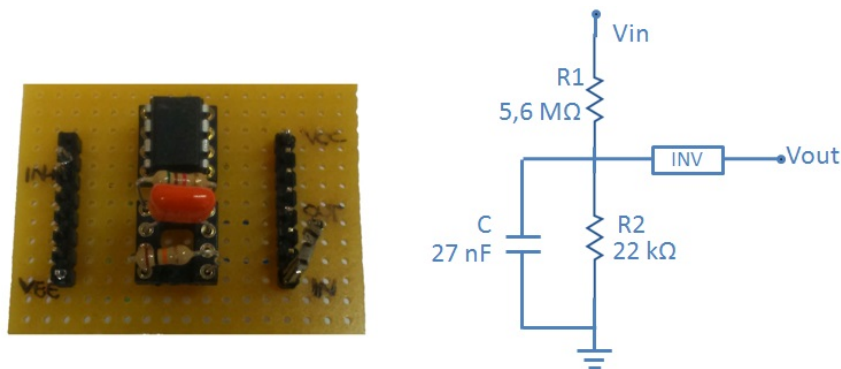


Figura 3.7: Divisor de Tensión para la Medición de la Corriente de Entrada

En este caso en concreto también se ha utilizado un divisor de tensión, para ello, se ha incluido una resistencia de 1Ω por medio de la cual se llevará a cabo la medición. Por último, como consecuencia del signo negativo de la corriente en la bobina, ha sido necesaria la utilización de un inversor [45].

Finalmente, destacar que para llevar a cabo las simulaciones necesarias para el desarrollo del presente proyecto fin de carrera ha sido necesaria la utilización de un modelo VHDL del Boost Elevador el cual ha sido incluido dentro del anexo 7.2.

4

Estrategia de control

Después de la descripción de la placa de desarrollo hardware y las herramientas software en el capítulo 2 y la descripción de los modos de conducción y las características del boost utilizado en el capítulo 3, el presente capítulo describirá la estrategia de control seguida en el presente proyecto fin de carrera.

En primer lugar se describirán los controladores digitales, haciendo hincapié en sus fundamentos matemáticos, los sistemas de control en lazo cerrado, las acciones de control, y por último, las estructuras de control utilizadas en el presente proyecto: lazo de tensión y el doble lazo (tensión y corriente).

Una vez conocidos los fundamentos matemáticos de los lazos a utilizar se describirá como se ha llevado a cabo el diseño de las controladores que posteriormente nos permitirán llevar a cabo la regulación sobre el boost elevador.

Por último, se describirá como se ha llevado a cabo el proceso de implementación de los controladores en lenguaje VHDL.

4.1. Controladores Digitales

El elemento más importante de los sistemas digitales de regulación realimentados [46] es sin lugar a dudas el controlador o también denominado regulador. El controlador digital produce una magnitud de control discreta en función de la señal de error digitalizada que depende de la discrepancia entre la señal que se desea imponer y la señal que realmente se obtiene.

Si tomamos como ejemplo el lazo cerrado típico de la figura 4.1 [47]:

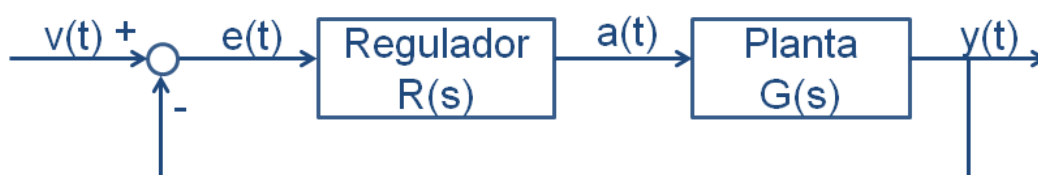


Figura 4.1: Lazo Cerrado

$$E(s) = X(s) - Y(s); \quad (4.1)$$

$$Y(s) = G(s)A(s); \quad (4.2)$$

$$A(s) = R(s)E(s); \quad (4.3)$$

de donde se obtiene:

$$Y(s) = G(s)R(s)E(s); \quad (4.4)$$

$$E(s) = \frac{Y(s)}{G(s)R(s)}; \quad (4.5)$$

por lo tanto

$$X(s) - Y(s) = \frac{Y(s)}{G(s)R(s)}; \quad (4.6)$$

Por último, despejando se obtiene la función de transferencia del sistema:

$$\frac{Y(s)}{X(s)} = \frac{G(s)R(s)}{1 + G(s)R(s)}; \quad (4.7)$$

Para compensar el error, los controladores deben contar, en la medida de lo posible, con las siguientes características:

- Estabilidad: La respuesta a una señal debe alcanzar y mantener un valor útil durante un período razonable. Un sistema de control inestable produciría oscilaciones en la señal.
- Exactitud: Un sistema de control debe ser exacto dentro de ciertos límites especificados, reduciendo cualquier error a un límite aceptable. Se debe tener en cuenta que el coste de un sistema de control aumentará conforme lo haga su exactitud.
- Rapidez: Un sistema de control debe completar su respuesta a una señal de entrada en un tiempo aceptable. Aunque un sistema sea estable y tenga la exactitud requerida no tiene ningún valor si el tiempo de respuesta a una entrada es mucho mayor que el tiempo entre las señales.

El controlador deberá ser diseñado de manera tal que cumpla en la medida de lo posible con estos tres condicionantes. Esto no siempre será sencillo, ya que las condiciones tienden a ser incompatibles y debe establecerse una solución de compromiso entre ellas. El sistema de control ideal es estable, de una exactitud absoluta y responderá instantáneamente a cualquier cambio en las variables de referencia algo que a día de hoy es imposible obtener.

4.1.1. Fundamentos Matemáticos de los Sistemas de Control

Para llevar a cabo el diseño de los sistemas de control digital es necesario conocer la representación matemática de los mismos [48].

En primer lugar es necesario introducir la definición de integral de convolución [49]. Para el caso de sistemas lineales e invariantes en el tiempo [49], la integral de convolución permite determinar la respuesta del sistema ante cualquier entrada a partir del conocimiento de la respuesta del sistema ante una única entrada particular, el impulso. Si la respuesta del sistema ante un impulso se denota como $g(t)$, la salida del sistema excitado con una entrada cualquiera $x(t)$ está dada por la expresión:

$$y(t) = \int_{-\infty}^{\infty} x(t - \tau)g(\tau)d\tau = x(t) * g(t) \quad (4.8)$$

La integral de convolución no es una operación sencilla, por este motivo se utilizan transformaciones sobre las señales del sistema. En concreto, la transformada de Laplace [50] de un sistema de tiempo continuo está definida por:

$$X(s) = L\{x(t)\} = \int_0^{\infty} x(t)e^{-st} dt \quad (4.9)$$

donde s es una variable compleja. Una de las ventajas de este dominio transformado es que la operación de convolución queda definida por una simple multiplicación.

$$Y(s) = X(s)G(s) \quad (4.10)$$

La transformada de Laplace en el dominio discreto se conoce como transformada Z y está definido por:

$$X(z) = \sum_{k=-\infty}^{\infty} x_k z^{-k} \quad (4.11)$$

debido a su naturaleza discreta, esta transformación es la utilizada para el diseño del controlador digital del presente proyecto (ver figura 4.2).



Figura 4.2: Función de Transferencia

Como se puede observar en la figura 4.2 $Y(z)$ queda definidor por:

$$Y(z) = X(z)G(z) \quad (4.12)$$

Despejando directamente de la ecuación anterior se obtiene:

$$G(z) = \frac{Y(z)}{X(z)} \quad (4.13)$$

conocida como función de transferencia y a partir de la cual es posible obtener una sencilla relación entre la entrada y la salida del sistema.

El denominador de la función de transferencia se conoce como función característica, pues determina, a través de los valores de sus coeficientes, las características físicas de los elementos que componen el sistema. Mientras que las raíces de la ecuación característica se denominan polos del sistema, las raíces del numerador reciben el nombre de ceros del sistema.

Por otro lado, utilizando las ecuaciones 4.13 y 4.11 es posible obtener:

$$G(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + \dots + b_mz^{-m}}{a_0 + a_1z^{-1} + a_2z^{-2} + \dots + a_mz^{-m}} \quad (4.14)$$

reagrupando términos se obtiene:

$$(a_0 + a_1z^{-1} + a_2z^{-2} + \dots + a_mz^{-m})Y(z) = (b_0 + b_1z^{-1} + b_2z^{-2} + \dots + b_mz^{-m})X(z) \quad (4.15)$$

De la ecuación 4.15 es posible obtener una relación entre los elementos de la señal de salida y la señal de entrada conocida por ecuación en diferencias.

$$a_0y_k + a_1y^{k-1} + a_2y^{k-2} + \dots + a_ny^{k-n} = b_0x_k + b_1x^{k-1} + b_2x^{k-2} + \dots + b_nx^{k-n})X(z) \quad (4.16)$$

$$y_k = -\frac{a_1}{a_0}y^{k-1} - \dots - \frac{a_n}{a_0}y^{k-n} + \frac{b_0}{a_0}x^k - \dots - \frac{b_m}{a_0}x^{k-n} \quad (4.17)$$

Para el caso en concreto del diseño de controladores será necesario sustituir estos por un bloque que realizará en todo momento una misma función y donde la salida estará determinada por la función de entrada. No hay que olvidar, que para poder utilizar la integral de convolución definida en la ecuación 4.8, el bloque deberá contar con las características de linealidad e invarianza temporal. Mientras que la linealidad hace referencia a señales de entrada y salida proporcionales, un sistema se denomina invariante en el tiempo si su salida es siempre la misma ante una misma entrada, sin importar el instante de tiempo en el que se aplica dicha entrada.

4.1.2. Sistemas de Control en Lazo Cerrado

Como se ha visto anteriormente, un sistema puede ser modelado mediante un diagrama de bloques, es decir, una representación gráfica de las funciones realizadas por cada componente los cuales se encuentran caracterizados por su función de transferencia.

Una vez conocido el sistema completo, el proceso de control empieza con la comparación entre la lectura del sensor y el valor de referencia al que se desea que el sistema llegue. De esa comparación resulta la señal de error, que será la entrada del controlador.

No hay que olvidar que la señal de error está muy ligada a las perturbaciones, es decir, señales no deseadas que afectan a la salida del sistema. Las perturbaciones pueden ser internas, si estas se generan dentro del sistema, o externas, si se generan fuera del sistema y constituyen una entrada. Por este motivo siempre, se intentará trabajar con sistemas libres de perturbaciones en la medida de lo posible [51].

Como se puede observar en la figura 4.3, dependiendo de la señal de error obtenida, se proporcionará una entrada al controlador para que tras su actuación, la salida del sistema sea lo más parecida posible a la salida deseada. Dependiendo de la señal de error, los controladores tienen diversas formas de actuación para atenuar la desviación de la señal de salida. Algunas de las acciones más frecuentes que pueden ser llevadas a cabo por los controladores de los sistemas de lazo cerrado, son detalladas a continuación.

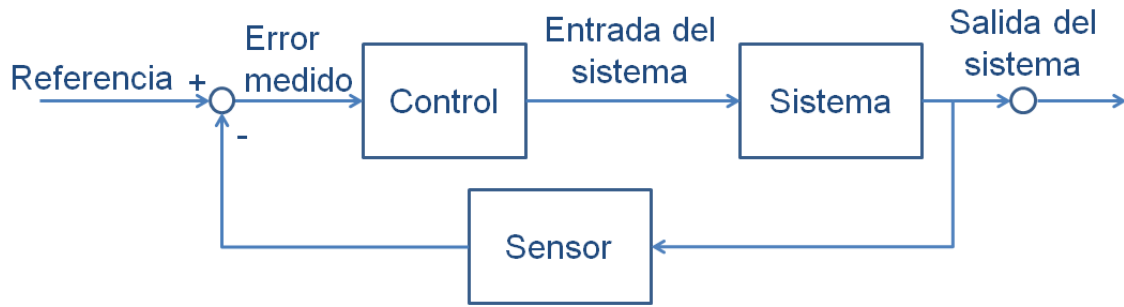


Figura 4.3: Control Realimentado

4.1.3. Acciones de Control

Antes de pasar a describir algunas de las acciones de control [52], es necesario definir el error de estado estacionario [49] [50] de un sistema de control como la medida de la exactitud para seguir una entrada dada después de desaparecer la respuesta transitoria. Por otro lado, una sistema permanecerá estable cuando ante una entrada acotada el sistema responde con salida acotada.

Acción de Control Proporcional (P)

Esta acción produce una señal de accionamiento proporcional a la señal de error del sistema. Es el más simple de todos los tipos de control y consiste simplemente en amplificar la señal de error antes de aplicarla al sistema.

La relación entre la salida del controlador y la señal de error es la siguiente:

$$u(t) = K_p e(t) \quad (4.18)$$

donde K_p es la ganancia proporcional que determinará el nivel de amplificación del elemento de control.

En estos casos, para evitar que el sistema sea sensible al ruido de medición, la ganancia de lazo no debe ser muy grande. Por tanto, el establecimiento de la ganancia debe ser considerado como un compromiso entre la necesidad de corrección del error y el incremento de ruido en el sistema.

La acción proporcional normalmente producirá error de estado estacionario.

Acción de Control Proporcional Integral (PI)

En la práctica no existen controladores que tengan únicamente acción integral, por ello, esta acción se encuentra siempre combinada con una acción proporcional. La función principal de la acción integral es asegurar que la salida del proceso concuerde con la referencia en estado estacionario.

Como consecuencia de la acción proporcional, donde normalmente existiría un error de estado estacionario, la acción integral producirá un incremento en la señal de control ante errores positivos y un decremento ante errores negativos.

La acción PI queda entonces definida por la siguiente ecuación:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) \quad (4.19)$$

en donde K_p es la ganancia proporcional y se denomina T_i al tiempo integral.

En conclusión, la acción integral elimina el error de estado estacionario producido por la acción proporcional aunque introducir una respuesta oscilatoria, lo que puede resultar inconveniente.

Acción de Control Proporcional Derivativo (PD)

La acción proporcional y derivativa puede ser interpretada como si el control proporcional fuese hecho para predecir la salida del proceso. La estructura básica de un controlador PD está dada por:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} \quad (4.20)$$

en donde K_p es la ganancia proporcional y T_d es una constante denominada tiempo derivativo.

Una ventaja del uso de la acción de control derivativa es que esta responde a la velocidad del cambio del error y produce una corrección significativa antes de que la magnitud del error se vuelva demasiado grande. Por tanto, el control derivativo prevé el error, inicia una acción oportuna y tiende a aumentar la estabilidad del sistema. Aunque el control derivativo no afecta en forma directa el error de estado estacionario, añade amortiguamiento al sistema y, por tanto, permite el uso de un valor más grande en la ganancia, lo cual provoca una mejora en la precisión en estado estable.

A continuación, en la figura 4.4, se pueden observar las acciones de control descritas anteriormente.

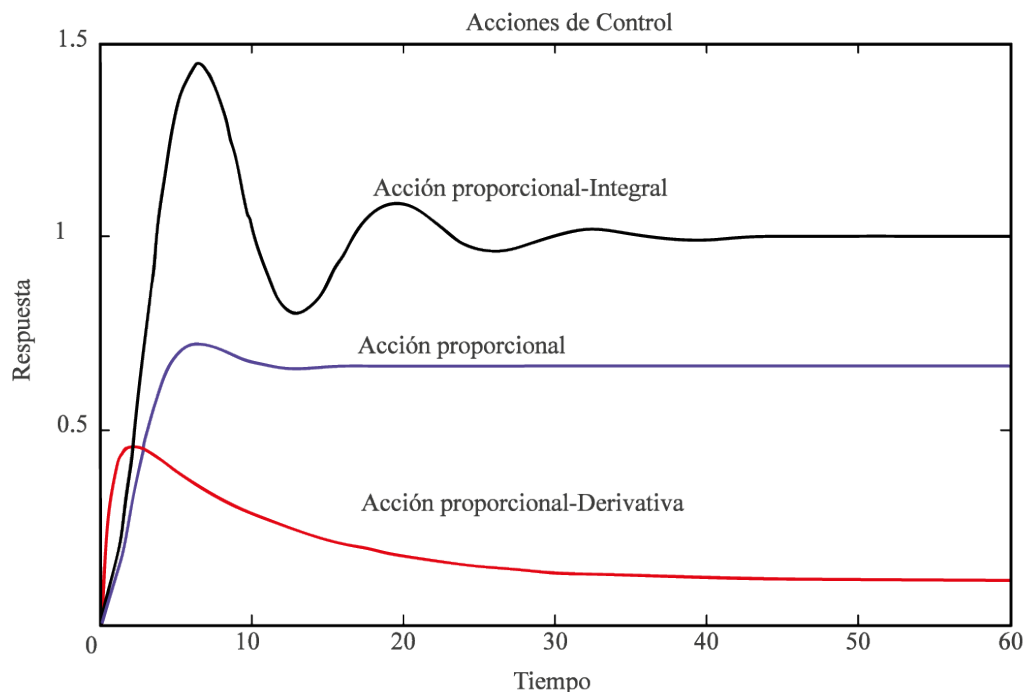


Figura 4.4: Acciones de Control P, PI y PD

Acción de Control Proporcional Integral Derivativo (PID)

La combinación de una acción de control proporcional, una acción de control integral y una acción de control derivativa se denomina acción de control Proporcional Integral Derivativo o PID. Esta acción de control combina las ventajas de cada una de las tres acciones de control individuales, por ellos se trata de la acción más extendida.

La relación entre la salida del controlador y la señal de error de la acción de control Proporcional Integral Derivativo está dada por:

$$u(t) = K_p e(t) + K_p T_d \frac{de_t}{dt} + \frac{K_p}{T_i} \int_0^t e(t) \quad (4.21)$$

en donde K_p es la ganancia proporcional, T_i es el tiempo integral y T_d es el tiempo derivativo.

4.1.4. Estructuras de Control

Como ya se ha indicado anteriormente, el control de una planta o sistema consiste en medir la magnitud que se quiere controlar y actuar sobre los parámetros modificables para conseguir que la magnitud a controlar se ajuste al valor deseado.

El presente proyecto fin de carrera tiene como objetivo la regulación de la tensión de salida de un boost elevador mediante un lazo de tensión y un doble lazo, con lazo interno de control de corriente y lazo externo de control de tensión (ver apartado 4.1.4).

En ambos casos, el control se va a llevar a cabo mediante una modulación por ancho de pulsos (también conocida por PWM) de una señal a la cual se modifica su ciclo de trabajo abriendo o cerrando así el elemento de conmutación. El conmutador permanecerá abierto cuando este reciba un estado bajo, por el contrario, permanecerá cerrado cuando reciba un estado alto.

Este proyecto en particular utiliza una señal cuadrada con periodo T y ciclo de trabajo D variable que dependerá, entre otros, de la tensión de salida.

A continuación, en la figura 4.5 se puede observar cómo el ciclo de trabajo puede variar mientras que el periodo de la señal PWM sigue constante.

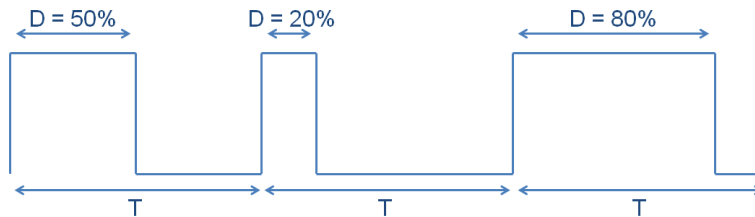


Figura 4.5: Ciclos de trabajo distintos para mismo periodo

Lazo de Tensión

Para llevar a cabo la regulación de la tensión de salida mediante un único lazo de tensión, es necesaria la medición de la tensión de salida aplicada a la carga para compararla con el valor de referencia de tensión de salida establecido. El error de esta comparación se aplica al regulador (diseñado según las especificaciones de tensión de salida de referencia) de donde se obtendrá una señal de control que proporciona el estado del interruptor (ver figura 4.6).

Como consecuencia de la utilización de un único lazo, el periodo del mismo vendrá limitado por la frecuencia de conmutación del mosfet incluido en el boost utilizado. Por ello, el presente

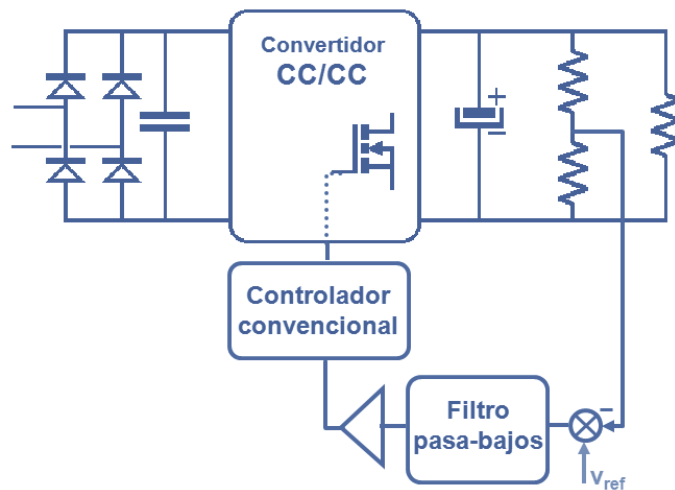


Figura 4.6: Lazo de Tensión

proyecto ha utilizado un periodo de actuación igual a $10 \mu\text{s}$. Este periodo coincide con el proporcionado por la máxima frecuencia del mosfet, es decir $f = 100\text{KHz}$. Por tanto, al ser $T = \frac{1}{f}$, el periodo de actuación mínimo proporcionado por el boost será $10 \mu\text{s}$.

Doble Lazo: Tensión y Corriente

Esta estrategia implementa un primer lazo de realimentación en el circuito de control que obliga al interruptor a conmutar de forma que la corriente de entrada siga un cierto patrón. De esta manera, es posible forzar a la corriente de entrada a una forma sinusoidal en fase con la tensión de entrada, consiguiendo una corriente de bajo contenido armónico. Este fenómeno es conocido como corrección del factor de potencia, Power Factor Correction (PFC).

Por el contrario, el segundo lazo obliga a la corriente de entrada a aportar la potencia necesaria para obtener la tensión de salida deseada.

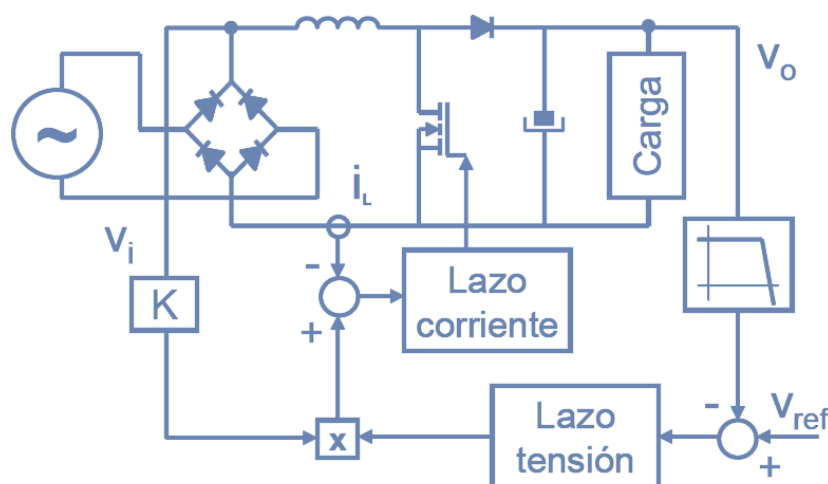


Figura 4.7: Doble Lazo

Como se muestra en la figura 4.7, la referencia de corriente es generada con el lazo externo de tensión. Por tanto, para garantizar la estabilidad del sistema se debe tener en cuenta que el controlador interno de corriente debe ser más rápido que el externo de tensión.

La ventaja de este control radica en que a la vez que se controla la tensión de salida, también se mide y controla la corriente de entrada, ofreciendo así una protección contra cortocircuitos. Además, es más modular, flexible y tolerante ante la variación de parámetros de la planta.

Actualmente, apenas existen posible mejoras sobre el lazo de corriente gracias al control digital. Es por ello que la mayoría de mejoras se centran en la mejora de la dinámica del lazo de tensión.

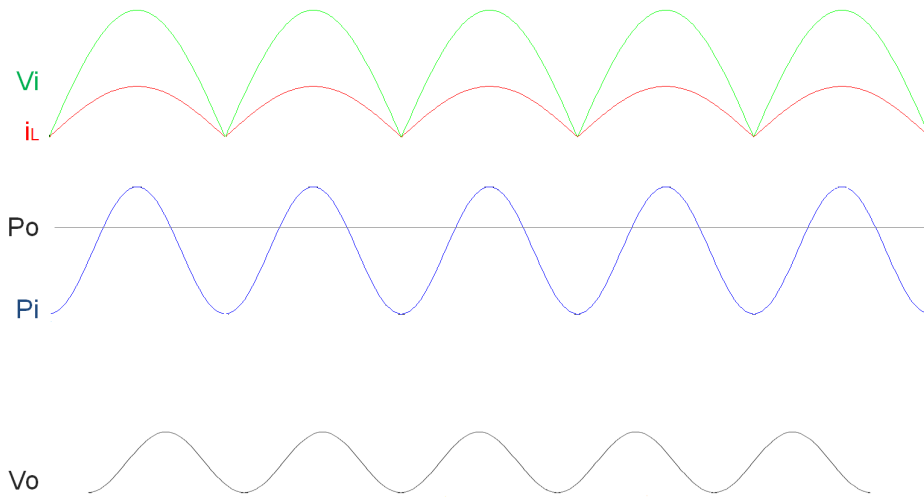


Figura 4.8: Formas de onda para V_i , I_L , P_i , P_o , V_o

En la figura 4.8 se observa que, al igual que la corriente de entrada, la tensión de salida también presenta un pequeño rizado como consecuencia del ciclo de conmutación. Mientras que la tensión presenta un rizado muy pequeño, el rizado de la corriente de entrada es comparable con su valor medio. Este hecho ha de tenerse en cuenta para determinar el periodo de actuación de los lazos. Cuanto menor sea el periodo de actuación, mayor será la precisión del lazo de control.

De mismo modo que sucede en el lazo de tensión (apartado 4.1.4), el periodo de actuación del lazo rápido, es decir, el de corriente, estará limitado por la frecuencia máxima del mosfet, es decir $10 \mu\text{s}$.

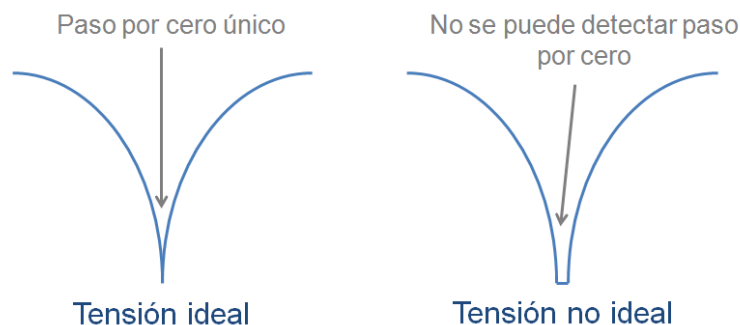


Figura 4.9: Tensión Rectificada

Como también se puede observar en la figura 4.8, el presente proyecto lleva a cabo una sincronización de los lazos basada en el paso por cero de la tensión y corriente de entrada. Como consecuencia de la rectificación de la tensión de la red normal, la señal de entrada no es ideal, y por lo tanto, realmente no presenta un paso por cero ya que los valores de tensión mínima no

se dan en un único instante tal y como se puede apreciar en la figura 4.9.

Para la medición de los pasos por cero es necesaria la detección de flancos de subida y bajada con respecto a unos determinados umbrales. Si una muestra es menor que un umbral y la siguiente es mayor que el umbral, entonces se habrá detectado un flanco de subida. En caso contrario, si una muestra es mayor que el umbral y la siguiente menor, entonces se habrá detectado una flanco de bajada.

Debido a que las muestras de la tensión rectificada de entrada presentan oscilaciones es necesario umbrales separados para identificar correctamente flancos de subida o bajada.

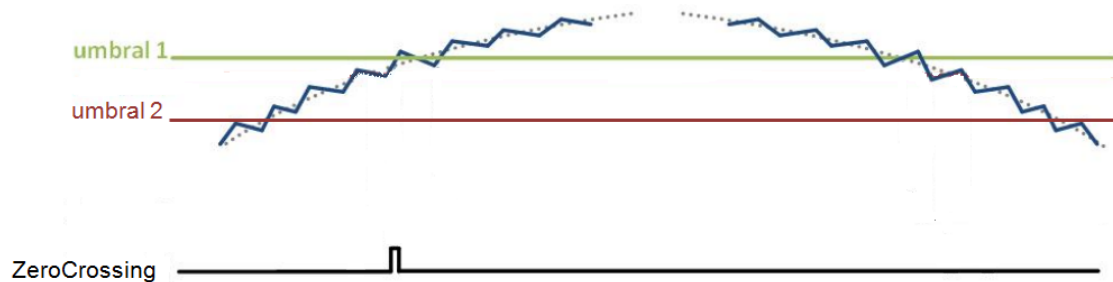


Figura 4.10: Detección de paso por cero

El presente proyecto ha utilizado un proceso denominado *ZeroCrossing* para determinar el paso por cero de la corriente de entrada. Dicho proceso utiliza umbrales de 16 V y 30 V para determinar el paso por cero de la tensión de entrada de modo que para las pequeñas variaciones en la tensión de entrada no interfieran en la identificación del paso por cero (ver figura 4.10).

Debido a la rectificación de la señal producida por la red normal, la cual está caracterizada por una frecuencia de 50 Hz, el periodo de muestreo del lazo de tensión será igual a 10 ms.

$$T = \frac{1}{2f} = 0,01s \quad (4.22)$$

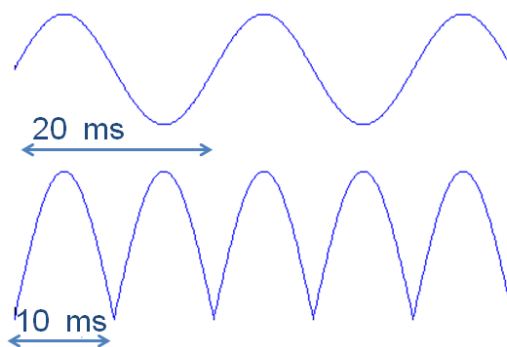


Figura 4.11: Frecuencia de la Red normal vs frecuencia rectificada

En resumen, el presente proyecto utilizará una periodo de 10 ms en el lazo de tensión y 10 μ s en el lazo de corriente. La regulación se llevará a cabo por medio del periodo de conmutación, es decir el periodo del lazo de corriente, a través del estado del conmutador. Sin embargo, será el lazo de tensión, el que actualice la tensión de error, es decir, la diferencia entre la tensión de salida deseada y la tensión real medida.

4.2. Diseño del Lazo de Tensión

Una vez vistos los fundamentos matemáticos y las diferentes estructuras de control, a continuación se describe el proceso llevado a cabo para el diseño del lazo de tensión en el presente proyecto fin de carrera.

En primer lugar, para poder llevar a cabo el diseño de un controlador es fundamental la correcta caracterización de la planta a partir de la cual será obtenida la función de transferencia del regulador. En este caso en concreto, el regulador para el lazo de tensión deberá relacionar la tensión de salida con una señal de PWM que determinará el estado del mosfet.

La obtención de la función de transferencia no se encuentra dentro del alcance del presente proyecto, por ello, ha sido obtenida directamente del libro “Fundamentals of Power Electronics” incluido en la bibliografía [42]. La función de transferencia utilizada se muestra en la ecuación 4.23.

$$\frac{Vo(s)}{d(s)} = G_{d0} \frac{1 - \frac{s}{w_z}}{1 + \frac{s}{Qw_0} + \left(\frac{s}{w_0}\right)^2} \quad (4.23)$$

donde

$$G_{d0} = \frac{Vo}{D'} \quad (4.24)$$

$$w_z = \frac{D'^2 R}{L} \quad (4.25)$$

$$Q = D' R \sqrt{\frac{C}{L}}; \quad (4.26)$$

$$Q = \frac{D'}{\sqrt{CL}}; \quad (4.27)$$

$$D' = 1 - D; \quad (4.28)$$

V_o indica tensión de salida que se desea obtener, D ciclo de trabajo de la señal PWM que determinará el estado del mosfet y R , C y L se corresponden con los valores de resistencia o carga, condensador y bobina utilizados en la planta.

Sustituyendo en la ecuación 4.23 por valores utilizados en el presente proyecto:

- $V_o = 20$ V
- $C = 100$ μ F
- $B = 1,1$ mH
- $R = 200$ Ω

se obtiene:

$$\frac{Vo(s)}{d(s)} = \frac{-0,00088s + 40}{4,4 \cdot 10^{-7}s^2 + 2,2 \cdot 10^{-5}s + 1} \quad (4.29)$$

La función de transferencia se encuentra en el dominio de tiempo continuo por lo cual es necesario realizar una transformación al dominio de tiempo discreto. Para la transformación del dominio del tiempo el presente proyecto ha utilizado la función `c2d` de Matlab.

La función `c2d` convierte el sistema continuo en el sistema discreto considerando por defecto que existe un retenedor de orden cero en la entrada y asumiendo un periodo de muestreo de T segundos. Durante el proceso de discretización, el retenedor de orden cero mantiene el valor de la salida constante hasta la llegada de un nuevo valor.

En este caso, el periodo de muestreo será el indicado en el apartado 4.1.4, es decir, $10 \mu\text{s}$.

Para la obtención de la función de transferencia de tiempo discreto se ha utilizado el siguiente código Matlab:

```
%Boost DC-DC

V=20;
Dp=0.5;
L=1.1e-3;
C=100e-6;
R=200;
Gd0=V/Dp
w0=Dp/sqrt(L*C)
Q=Dp*R*sqrt(C/L)
wz=Dp*Dp*R/L
num=Gd0*[-1/wz 1]
den=[1/(w0*w0) 1/(Q*w0) 1]
Gvd=tf(num,den)
Gvdz=c2d(Gvd,10e-6)
sisotool(Gvdz);
```

del cual se ha obtenido la siguiente función de transferencia:

$$\frac{Vo(z)}{d(z)} = \frac{-0,01545z + 0,02454}{z^2 - 1,999z + 0,9995} \quad (4.30)$$



Figura 4.12: Regulador de lazo de tensión

Una vez obtenida la función de transferencia de la planta en tiempo discreto, es hora de obtener el controlador. Para ello, se ha utilizado la herramienta Sisotool de Matlab descrita en el apartado 2.2.1

4.2.1. Diseño del Controlador

A través de Sisotool, conociendo la función de transferencia en lazo abierto, es posible dibujar el lugar geométrico de las raíces en lazo cerrado lo que permitirá definir las acciones del controlador para la obtención de la salida deseada. El análisis de un sistema empleando el lugar geométrico de las raíces se traduce en ubicar los polos y ceros de la función de transferencia en lazo abierto sobre el plano s o z y a partir de estos determinar la trayectoria que seguirían los polos en lazo cerrado cuando uno o más parámetros varían.

Controlador Utilizado

El controlador utilizado en el presente proyecto para el lazo de tensión ha sido el indicado en la ecuación 4.31 y en la figura 4.13,

$$\frac{d(z)}{V_{error}(z)} = 6,67572 \cdot 10^{-6} \frac{z - 0,125}{z - 1} \quad (4.31)$$

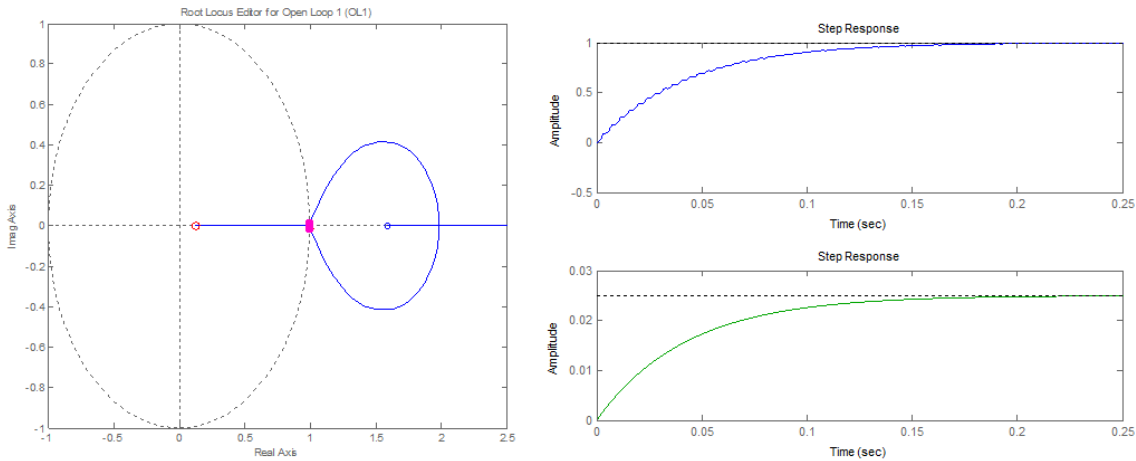


Figura 4.13: Respuesta Controlador Lazo de Tensión

Además, en la figura 4.13 se pueden observar las diferentes respuestas del controlador del lazo de tensión utilizado. Mientras que la gráfica de la izquierda representa el lugar de las raíces, las gráficas de la derecha representan la salida del sistema y la respuesta del regulador ante un escalón.

Como veíamos en el apartado 4.1, todo controlador debe cumplir con unas restricciones de diseño: estabilidad, exactitud y rapidez.

Estabilidad quiere decir que ante una entrada acotada se obtenga una salida acotada. Si el sistema es inestable no existe régimen permanente aunque numéricamente se puedan encontrar los valores de los límites en el dominio de Laplace.

Un sistema es estable cuando todos los polos del sistemas se encuentran dentro del círculo unidad. Por otro lado, incluir un inversor (polo en 1) ayudará a eliminar el error en régimen permanente.

Los polos del sistema son las raíces de la ecuación que resulta de igualar a cero el denominador de la función de transferencia del sistema. Esa ecuación se conoce con el nombre de ecuación característica del sistema. Por tanto, las raíces de la ecuación característica ofrecen información no solo del transitorio del sistema, sino también de su estabilidad.

Como ya he visto en el capítulo 4.1.2 existen diversas maneras de hacer estable un controlador aunque en este caso en concreto para el diseño del controlador del lazo de tensión en primer lugar ha sido necesario un ajuste de ganancia. Al no ser esta suficiente para obtener un error nulo en régimen permanente en la respuesta en bucle cerrado del sistema, ha sido añadido un control integral (polo en 1 en el lugar de la raíces). Este integrador puede producir que el sistema se vuelva inestable por lo tanto deberemos añadir un cero dentro del círculo unidad, a ser posible cerca de 1, para atenuar el efecto de un polo.

Concluir que, como se observa en la figura 4.13, el controlador utilizado se encuentra prácticamente estabilizado tras los primeros 200 ms.

Otros Controlador Estudiados

Antes de seleccionar el controlador utilizado, también se han estudiado otros controladores obtenidos a partir de la de la función de transferencia definida por la ecuación 4.30. Uno de estos controladores es el indicado por la ecuación 4.32 (CT1) y la figura 4.14:

$$\frac{d(z)}{Verror(z)} = 8,344652 \cdot 10^{-7} \frac{z - 0,125}{z - 1} \quad (4.32)$$

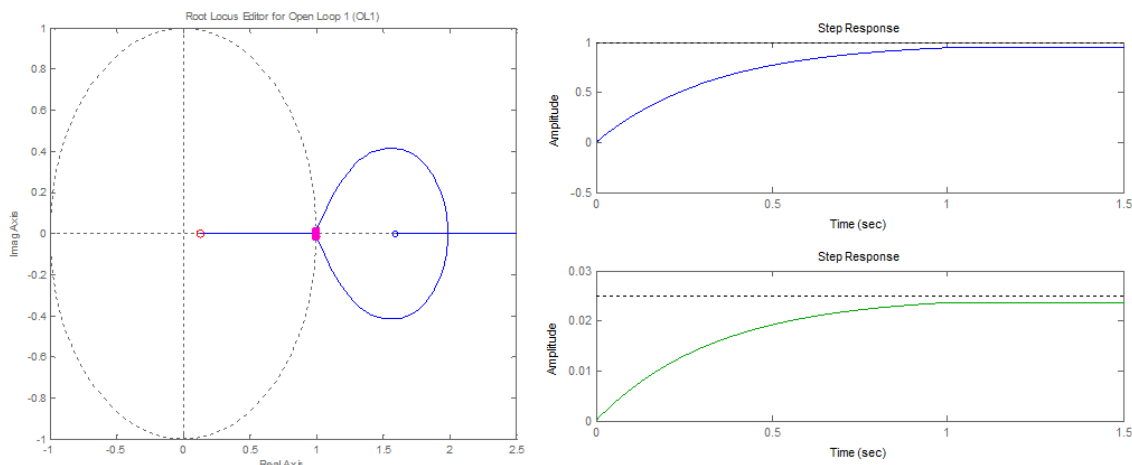


Figura 4.14: Respuesta Controlador Lazo de Tensión - CT1

Como se distingue en la figura 4.14, este controlador se estabiliza pasado el primer segundo, por lo tanto, tiene una respuesta más lenta que el utilizado.

4.3. Diseño del Doble Lazo

El diseño del doble lazo se caracteriza por la utilización de dos controladores. Mientras el lazo lento, el lazo de tensión, tendrá como entrada la tensión de salida del boost y como salida una ganancia que determinara la corriente de referencia, el lazo rápido, lazo de corriente, tendrá como entrada la corriente de la bobina y como salida la señal de PWM que determinará el estado del mosfet.

Del mismo modo que sucede en el diseño del lazo de tensión, las funciones de transferencia que caracterizan la planta de ambos lazos han sido obtenidas del libro “Fundamentals of Power

Electronics” [42]. Las funciones de transferencia utilizadas pueden observarse en las ecuaciones 4.33 y 4.34.

$$\frac{V_o(s)}{Gin(s)} = \frac{\frac{V_i^2 R}{2V_o}}{\frac{RCs}{2} + 1} \quad (4.33)$$

$$\frac{i_L(s)}{d(s)} = \frac{V_o}{Ls} \quad (4.34)$$

donde V_i indica la tensión de entrada, V_o la tensión de salida que se desea obtener, Gin la ganancia que determinará la corriente de referencia, del ciclo de trabajo de la señal PWM que determinará el estado del mosfet, y R , C y L se corresponden con los valores de resistencia, condensador y bobina utilizados en la planta.

Sustituyendo en las ecuaciones 4.33 y 4.34 por valores utilizados en el presente proyecto:

- $V_i = 230 \text{ V}$
- $V_o = 400 \text{ V}$
- $C = 100 \mu\text{F}$
- $B = 1,1 \text{ mH}$
- $R = 533,33 \Omega$

se obtiene:

$$\frac{V_o(s)}{Gin(s)} = \frac{3,527 \cdot 10^4}{0,02667s + 1} \quad (4.35)$$

$$\frac{i_L(s)}{d(s)} = \frac{400}{0,0011s} \quad (4.36)$$

De igual modo que se ha realizado para el lazo de tensión, al encontrarse las ecuaciones 4.35 y 4.36 en el dominio de tiempo continuo, es necesario realizar una transformación al dominio de tiempo discreto.

Para la discretización del lazo de corriente y el lazo de tensión se utilizarán los periodos definidos en el apartado 4.1.4, es decir, $10 \mu\text{s}$ y 10 ms respectivamente.

Para la obtención de las funciones de transferencia de tiempo discreto se ha utilizado el siguiente código Matlab:

```
%Planta lenta
Vi=230;
R=400*400/300;
C=100e-6;
Vo=400;
num=Vi*Vi*R/(2*Vo);
den=[R*C/2 1];
Rvo=tf(num,den)
Rvoz=c2d(Rvo,10e-3)
sisotool(Rvoz);
```

```
%Planta Rápida
```

```

L=1.1e-3;
Vo=400;
den=[L 0];
Rii=tf(Vo,den)
Riiz=c2d(Rii,10e-6)
sisotool(Riiz);

```

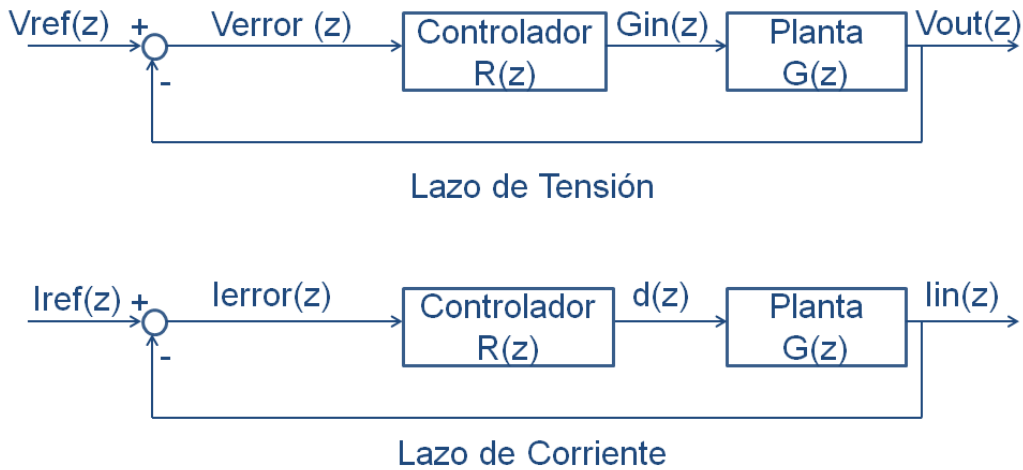


Figura 4.15: Reguladores de lazo de tensión y corriente

del cual se han obtenido las siguientes funciones de transferencia:

$$\frac{Vo(z)}{Gin(z)} = \frac{1,103 \cdot 10^4}{z - 0,6873} \quad (4.37)$$

$$\frac{i_L(z)}{d(z)} = \frac{3,636}{z - 1} \quad (4.38)$$

Al igual que sucedía con el diseño del lazo de tensión, Sisotool será la herramienta proporcionada por Matlab a partir de la cual se llevará a cabo el diseño de ambos controladores.

4.3.1. Diseño de Controladores

Además de la utilización de la herramienta Sisotool, para determinar las funciones de transferencia de los controladores del doble lazo es necesario conocer las características deseables de estos.

En el presente proyecto, el lazo de tensión es el más crítico, ya que es el encargado de la regulación de la tensión de salida, la cual se pretende que sea constante. Por este motivo, el lazo de tensión debe ser lo más estable posible evitando así la sobreoscilación.

El lazo de corriente, al no ser tan crítico, permitirá entonces un diseño sin tantas restricciones.

Controlador Utilizado

Mientras que el controlador utilizado para el lazo de tensión ha sido el indicado en la ecuación 4.39 y en la figura 4.16, el controlador del lazo de corriente ha sido el indicado en la ecuación 4.40 y en la figura 4.17.

$$\frac{Gin(z)}{Verror(z)} = 2^{-15} \frac{z - 0,5}{z - 1} \quad (4.39)$$

$$\frac{d(z)}{Ierror(z)} = 0,25 \frac{z - 0,96875}{z - 1} \quad (4.40)$$

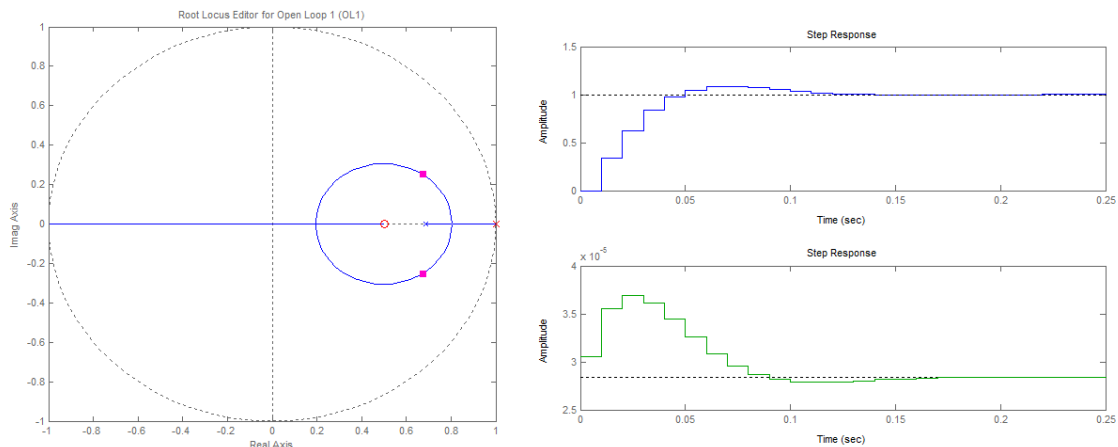


Figura 4.16: Respuesta Controlador Lazo de Tensión

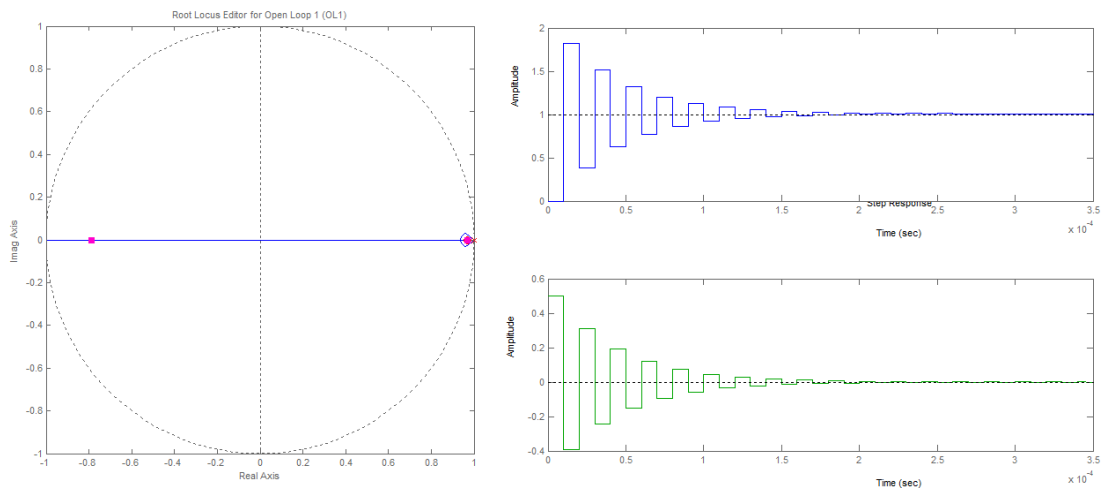


Figura 4.17: Respuesta Controlador Lazo de Corriente

Del mismo modo que se ha realizado para la obtención del regulador del control por medio de lazo de tensión, para la obtención de los reguladores (lazo de corriente y tensión) del control por medio de doble lazo, se ha optado por un ajuste de ganancia, añadir un integrador (polo en 1 en el lugar de la raíces) y finalmente añadir un cero dentro del círculo unidad.

Como se observa en la figura 4.16, el controlador de lazo de tensión utilizado se encuentra prácticamente estabilizado tras los primeros 50 ms.

Por el contrario, el controlador de lazo de corriente (ver figura 4.17) se estabiliza tras los primeros $150 \mu s$. Aunque el periodo de tiempo hasta llegar a la estabilización es muy superior al periodo del lazo de corriente, al tratarse de unidades de tiempo tan pequeñas, μs , es posible considerar al controlador válido.

Otros Controladores Estudiados

Para llegar a los controladores utilizados fueron estudiados otros controladores. Esto ha hecho posible llevar a cabo una comparación del comportamiento de los controladores eligiendo finalmente el de mejores prestaciones. A continuación, se muestran algunos de los controladores del lazo de corriente estudiados.

$$\frac{d(z)}{I_{error}(z)} = 0,0156 \frac{z - 0,999}{z - 1} \quad (4.41)$$

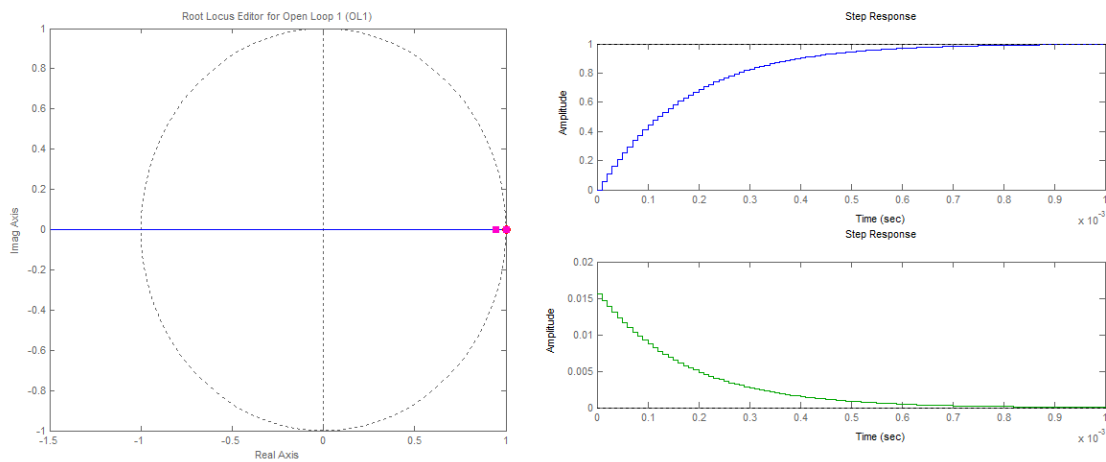


Figura 4.18: Respuesta Controlador Lazo de Corriente - C1

Como se puede observar en la figura 4.18, el controlador representado por la ecuación 4.41 (C1) no es lo suficientemente rápido como para afrontar las necesidades del presente proyecto. El controlador debería llegar a un estado de estabilización en un tiempo inferior o próximo a $10 \mu s$. En concreto, el presente controlador, se estabiliza se pasados los primeros $700 \mu s$.

$$\frac{d(z)}{I_{error}(z)} = 0,0075 \frac{z - 0,99}{z - 1} \quad (4.42)$$

Del mismo modo que sucede en el caso anterior, como se puede observar en la figura 4.19, el controlador representado por la ecuación 4.42 (C2) tampoco es lo suficientemente rápido ya que se estabiliza pasados los primeros $3 ms$.

$$\frac{d(z)}{I_{error}(z)} = 0,25 \frac{z - 0,75}{z - 1} \quad (4.43)$$

Por último, a diferencia de los controladores anteriores, el controlador representado por la figura 4.20 y la ecuación 4.43 (C3) se estabiliza pasados los primero $100 \mu s$, medida que se aproxima al periodo de conmutación del lazo de corriente en el presente proyecto.

Comparando este último controlador con el utilizado en el presente proyecto fin de carrera, se puede pensar que este último tendría mejores resultados ya que el tiempo que transcurre hasta

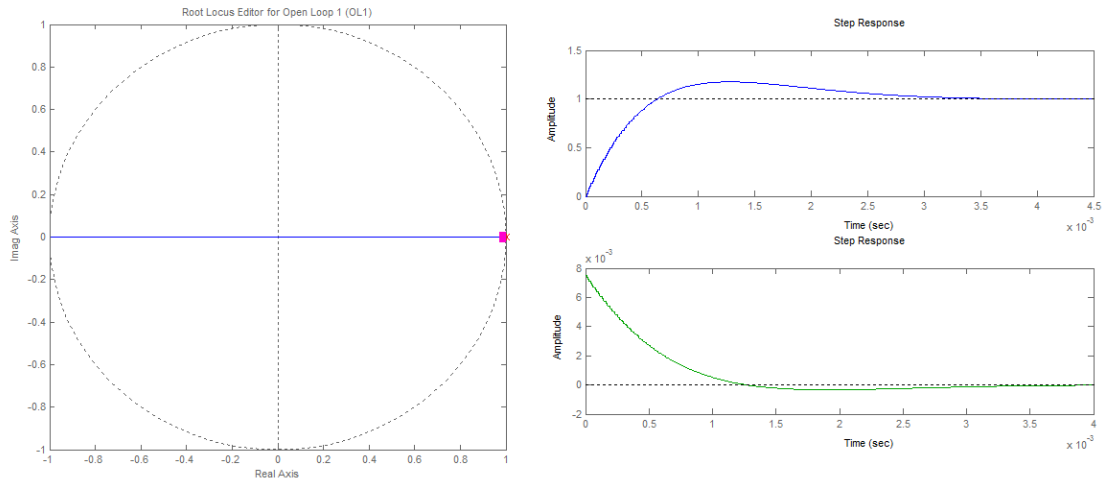


Figura 4.19: Respuesta Controlador Lazo de Corriente - C2

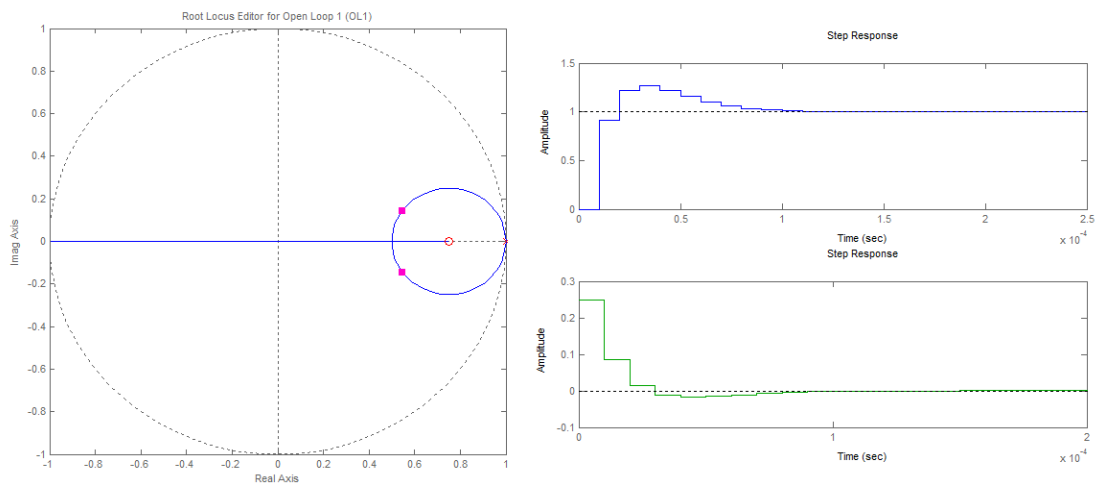


Figura 4.20: Respuesta Controlador Lazo de Corriente - C3

llegar al estado de estabilización en el controlador finalmente utilizado es $150 \mu\text{s}$, es decir, superior al presente en el controlador anterior. Sin embargo, tras llevar a cabo diversas simulaciones mediante la herramienta ModelSim se ha observado que el controlador utilizado obtiene mejor respuesta que el presentado anteriormente. Esto se debe a que las condiciones del lazo de corriente son muy variables a lo largo del periodo de red.

4.4. Implementación de los Lazos

Antes de comenzar con la descripción de la implementación de funciones de transferencia en VHDL, se hace indispensable indicar cómo es posible implementar estas funciones de manera sencilla a partir de sus ecuaciones en diferencias, de forma que únicamente será necesario realizar operaciones de suma y multiplicación.

Para ello, se definirán cada uno de los coeficientes de la ecuación en diferencias en formato binario QX.Y, de manera que se pueda llevar a cabo de manera sencilla la suma y multiplicación de coeficientes.

4.4.1. Formato QX.Y

El formato QX.Y representa un número en coma fija donde X e Y indican el número de bits utilizados para la definición de la parte entera y la parte decimal respectivamente. Es necesario indicar que este formato utiliza la numeración en complemento a 2, por medio de la cual es posible representar números binarios negativos. Por último, para la definición de un número binario en formato QX.Y serán necesarios $1 + X + Y$ bits (ver figura 4.21), donde el bit adicional se utilizará para la definición del signo.

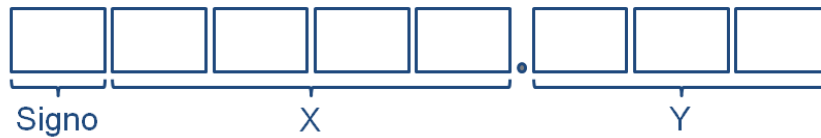


Figura 4.21: Formato QX.Y

A continuación se procede a la realización de un ejemplo que ayudará a la comprensión del formato QX.Y. Para representar 0,484375 en formato Q0.6, será necesario obtener el equivalente en complemento a 2 de 0,484375. Como el número a representar en formato Q0.6 cuenta con 6 decimales, su complemento a 2 equivale a representar el entero obtenido al realizar $0,484375 \cdot 2^6 = 31$, el cual es 011111. Como en este caso en particular no existe parte entera, únicamente será necesario tener en cuenta el bit de signo, por lo tanto 0,484375 en formato Q0.6 será equivalente a 0011111.

No hay que olvidar que en ocasiones pueden existir errores de redondeo a la hora de llevar a cabo una representación en complemento a 2, aunque estos sean pequeños, habrá que tenerlo en cuenta a la hora de la implementación, sobre todo, si se realizan operaciones sucesivas en las que intervienen estos errores.

Del mismo modo, si se quiere representar 0,485894 en formato Q0.6, es necesario conocer el complemento a 2 del entero resultante al realizar $0,485894 \cdot 2^6 = 31,097$. Al no dar como resultado un número entero exacto, aunque la representación en formato Q0.6 será igual a la utilizada para 0,484375, es decir 0011111, en este caso existirán errores de redondeo.

Además de las consideraciones anteriores, para llevar a cabo operaciones de suma o multiplicación será necesario tener en cuenta algunas otras. En la suma de dos números en formato QX.Y es imprescindible que ambos operandos tengan el mismo número de dígitos decimales, es decir, que estén alineados a la derecha. Para ello, si los sumandos no contienen los mismo decimales es necesario añadir ceros por la derecha al que tenga menos decimales. Esta operación únicamente añade decimales sin valor para realizar la suma. El resultado de la operación de suma podrá ir en el formato más grande de operando (el de mayor X), aunque este hecho podría provocar un desbordamiento. Para evitar problemas de desbordamiento se recomienda incrementar en 1 bit la X de ambos operandos antes de llevar a cabo esta operación (ver figura 4.22).

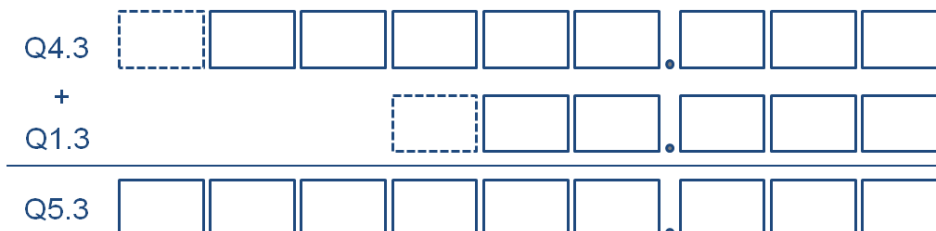


Figura 4.22: Suma Formato QX.Y

Por el contrario, para llevar a cabo una operación de multiplicación, ambos operando pueden

encontrarse en cualquier formato. Si por ejemplo, llevamos a cabo la multiplicación dos números con los siguiente formatos, QA.B y QC.D, el resultado de la operación tendrá por formato $Q(A + C + 1).(B + D)$ (ver figura 4.23).

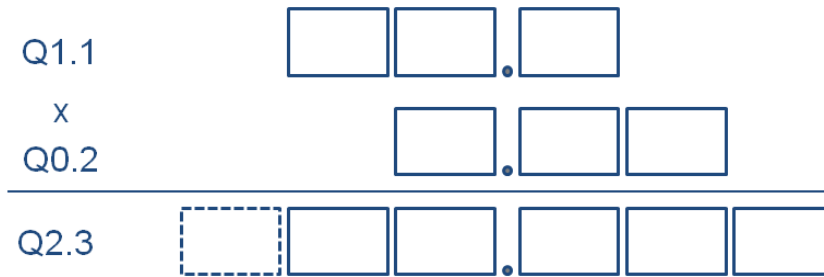


Figura 4.23: Producto Formato QX.Y

Es importante resaltar que esta notación no es interpretada por el lenguaje de programación ya que estará definida mediante vector con signo (`std_logic_vector`) de longitud $X + Y + 1$. Será labor del programador el interpretar el valor de las señales resultantes.

4.4.2. Implementación de Funciones de Transferencia

Para llevar a cabo la implementación de los lazos de control, además de la la representación de las funciones de transferencia en lenguaje VHDL, es necesario tener algunas otras consideraciones que permiten y aseguran el funcionamiento del mismo.

Una vez definidas las funciones de transferencia de los distintos reguladores es necesario llevar a cabo la representación de estas en ecuaciones en diferencias mediante la siguiente expresión:

$$Y(k) = A0Y(k - 1) + A1X(k) - A2X(k - 1) \quad (4.44)$$

$Y(k)$ representa la señal de PWM tanto en la función de transferencia indicada por la ecuación 4.31 como en 4.43. En ambas implementaciones, control de tensión de salida mediante un único lazo de tensión o mediante un doble lazo, la señal de PWM determinará el estado del conmutador y cuenta con una resolución de 500 valores. Como la frecuencia de reloj utilizada en el presente proyecto es de 50 MHz. Cada uno de los valores del contador será generado cada 20 ns, por tanto, el periodo de la señal de PWM será de 10μ , el cual coincide con el periodo de conmutación del mosfet.

Aunque inicialmente se ha utilizado un PWM fijo al 50 % para llevar a cabo las simulaciones necesarias, finalmente esta será variable pudiendo llevar a cabo la correcta regulación de la tensión de salida. En estos últimos casos, la señal de PWM es generada a través de un contador de 500 valores que van desde 0 hasta 499. Este contador es comparado con una señal de entrada, si el valor del contador es inferior a la señal de entrada la señal de PWM generada es un '1'. En caso de que el contador sea igual o superior a la señal de entrada, la señal de PWM será '0' (ver figura 4.24).

La utilización de una señal de control digital supone ciertas ventajas sobre el analógico como por ejemplo la introducción sencilla de limitación a través de líneas de código. El presente proyecto fin de carrera ha introducido limitaciones para limitar el ciclo de trabajo máximo y mínimo de la señal de control.

El ciclo de trabajo máximo en todos los casos ha sido limitado al 95 %. Esta medida también garantiza que la frecuencia de conmutación del mosfet continúe siendo 100 KHz ya que si se diese la situación donde D fuese 1, no existiría conmutación durante ese ciclo. Del mismo modo el ciclo de trabajo mínimo ha sido limitado al 5 %.

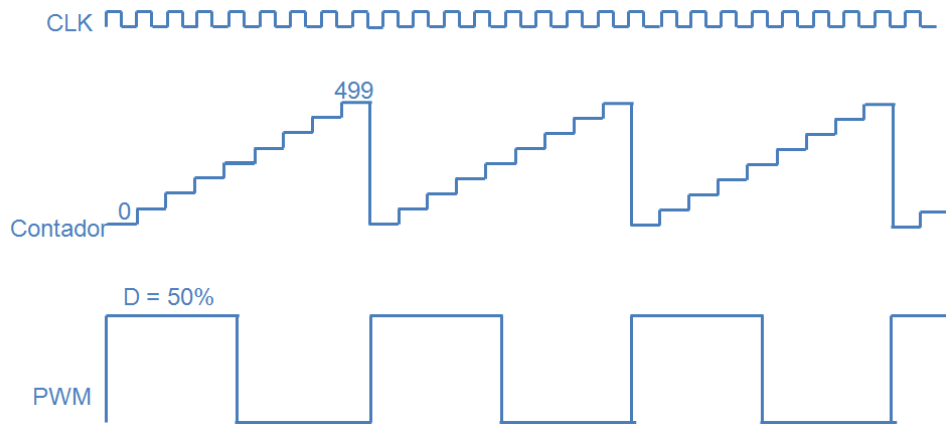


Figura 4.24: Generación de la señal PWM

Implementación del Lazo de Tensión

Para el caso del control de la tensión de salida mediante un único lazo de tensión, $Y(k)$ representa la señal de PWM que determina el estado del mosfet mientras que $X(z)$ representa la tensión error, es decir la tensión de referencia menos la tensión de salida en el instante actual.

Entonces, la función de transferencia obtenida a partir de la ecuación 4.23 queda representada por la ecuación 4.45:

$$d(z) = d(z - 1) + 6,67572 \cdot 10^{-6} \cdot Verror(z) - 6,67572 \cdot 10^{-6} \cdot 0,125 \cdot Verror(z - 1) \quad (4.45)$$

Esta función será utilizada en el código VHDL implementado que actuará de regulador. Su implementación VHDL puede encontrarse dentro de los Anexos.

Implementación del Doble Lazo

En el caso concreto del control de la tensión de salida mediante un doble lazo, en el lazo de tensión $Y(k)$ representa la salida del regulador mientras que $X(k)$ representa la tensión de error, es decir, la diferencia entre la tensión de salida deseada y la obtenida. Por el contrario, en la ecuación en diferencias del lazo de corriente, $Y(k)$ representa la señal PWM que determina el estado del mosfet y $X(k)$ la corriente error.

En concreto, las funciones de transferencia obtenidas a partir de las ecuaciones 4.39 y 4.40 quedan representadas por las ecuaciones 4.46 y 4.47 respectivamente:

$$Gin(z) = Gin(z - 1) + 2^{-15} \cdot Verror(z) - 2^{-15} \cdot 0,5 \cdot Verror(z - 1) \quad (4.46)$$

$$d(z) = d(z - 1) + 0,25 \cdot Ierror(z) - 0,25 \cdot 0,96875 \cdot Ierror(z - 1) \quad (4.47)$$

Estas funciones serán utilizadas en el código VHDL implementado que actuará de regulador. La implementación VHDL de ambos reguladores puede encontrarse dentro de los Anexos.

5

Simulaciones

Después de la descripción de cómo se ha llevado a cabo la implementación en VHDL de los diversos reguladores utilizados en el capítulo 4, en el presente capítulo se detallarán los resultados obtenidos de las simulaciones llevadas a cabo a través de la herramienta ModelSim.

En primer lugar, se detallarán los modelos VHDL utilizados: modelo Boost Elevador y modelo ADC de Microsemi. Estos modelos imitan el comportamiento de estos bloques de manera que es posible predecir su funcionamiento y verificar así la regulación de la tensión de salida sobre el Boost Elevador. En caso de fallo, la simulación permite la corrección de fallos antes de proceder con las pruebas experimentales.

A continuación, se detallarán los resultados obtenidos de las simulaciones para el caso del control por medio de un único lazo de tensión y el control por medio de doble lazo.

Por último, será en el capítulo 6 donde se describan las pruebas experimentales llevadas a cabo. Estas pruebas, permiten la visualización del verdadero comportamiento de la regulación para cada modelo de control: lazo de tensión y doble lazo.

5.1. Modelos VHDL

A continuación se describen los modelos VHDL utilizados para la etapa de simulación. Mientras el modelo de Boost Elevador describirá el funcionamiento del convertidor dependiendo de la señal de PWM que tiene como entrada, el modelo del ADC realizará una conversión analógico digital sobre las mediciones llevadas a cabo.

Destacar que para el control por medio un único lazo de tensión, únicamente se llevan a cabo mediciones sobre el valor de tensión de salida. Por el contrario, para el control por medio de doble lazo, se llevarán a cabo mediciones sobre la tensión de entrada y salida y sobre la corriente de entrada.

El código VHDL de los modelos utilizados, *Modelo Boost Elevador* y *Modelo ADC* pueden encontrarse dentro del apartado Anexos.

Modelo Boost Elevador

Este modelo es el encargado de predecir la tensión de salida y la corriente de entrada en el convertidor Boost Elevador tras la asignación de la tensión de entrada, el estado de conmutador y la fijación de una carga.

Por ello, las entradas del mismo, además de las señal de reloj y reset, que permiten la sincronización e inicialización del bloque, serán la corriente de entrada (V_i), el estado del conmutador (D) y la corriente de salida (i_o).

Como es de suponer, las salidas de este modelo serán señales reales que representan la tensión de salida (V_o) y la corriente de entrada (i_L), las cuales dependen del estado del conmutador.

Para el caso concreto del control por medio de un lazo de tensión, en primer lugar se ha fijado una tensión de entrada continua de 15 V y una resistencia de 200Ω .

Para el caso de control mediante doble lazo, mientras la tensión de entrada en alterna se ha fijado en 230 V, la carga ha sido fijada a $533,33\Omega$. Tal y como se indica en el apartado 3.2 la potencia nominal máxima de salida soportada por el Boost Elevador es de 400 W, al fijar la carga a $533,33\Omega$ se asegura que la potencia nominal máxima a la salida será 300 W, es decir, el valor máximo soportado por el convertidor.

$$R = \frac{V_o^2}{P_o} = \frac{400^2 W}{300 W} = 533,33\Omega \quad (5.1)$$

Modelo ADC de Microsemi

Una vez conocido el valor real instantáneo de tensión de salida para el caso de control por medio de un único lazo de tensión y los valores reales instantáneos de la tensión de entrada, tensión de salida y la corriente de entrada para el caso del control por medio del doble lazo, como consecuencia de la utilización de un regulador digital, es necesario llevar a cabo una conversión analógica a digital de los diferentes valores, siendo el bloque *Modelo ADC de Microsemi* es el responsable de esta conversión.

Este modelo, además de las entradas de reloj y reset, contará con tantas entradas reales como sea necesario convertir.

La conversión de analógico a digital se realiza a través de la función *conv_std_logic_vector* de la librería *IEEE.numeric_std.all*. En primer lugar, se utiliza la función *floor* para pasar las señales reales proporcionadas por el modelo del convertidor a señales enteras. Posteriormente, a través de la función *conv_std_logic_vector* se obtienen señales *std_logic_vector* teniendo como entrada señales de enteros.

Por último, las salidas de este modelo se corresponden con las salidas del ADC integrado en la FPGA AFS600 en el modo utilizado, es decir, *IP core for ADC data processing and sequence control* (ver apartado 2.2.3):

- *DATAVALID*
- *ASSC_DONE*
- *ASSC_WAIT*
- *ASSC_CHSAT*
- *ASSC_CHLATC*
- *ADC_RESULT [11:0]*
- *ADC_CHNUMBER [4:0]*

5.2. Controladores

Como ya se ha indicado anteriormente el regulador tiene como salida la señal de PWM que determinará el estado del conmutador del Boost Elevador utilizado. Como se introdujo en el capítulo 3, cuando el conmutador se encuentra cerrado la energía procedente de la fuente es almacenada en la bobina. Por el contrario, cuando el conmutador se encuentra abierto, la tensión de almacenada en el la bobina se suma a la producida por la fuente obteniendo así a la salida una tensión superior a la de entrada.

De esta manera, lo que se intenta en ambos tipos de control, control mediante un único lazo de tensión o control mediante un doble lazo, es obtener un tensión de salida constante igual a una tensión de referencia fijada para cada caso.

Las señales de entrada del *Regulador*, además de las señales de reloj y reset serán al menos algunas de las salidas del *Modelo ADC de Microsemi: ADC RESULTS, ADC CHNUMBER, DATAVALID* y *ASSC_DONE*. También podrán encontrarse otras señales de entrada que se corresponden con pulsadores o pines de entrada de la FPGA que nos han ayudado a la visualización de los resultados de los diferentes canales del ADC y al establecimiento de la tensión de referencia entre otros.

Es importante tener en cuenta que a través de un estado alto simultáneo en las señales *DATAVALID* y *ASSC_DONE* se establece cuando la salida del canal analizado por el ADC ha finalizado la conversión y por tanto el resultado es válido. El análisis de estas dos señales junto con *ADC CHNUMBER*, hace posible muestrear las señales digitales que contienen los diferentes valores de la tensión de salida para el control por medio del lazo de tensión o la tensión de entrada, tensión de salida y corriente de entrada para el caso del control mediante el doble lazo.

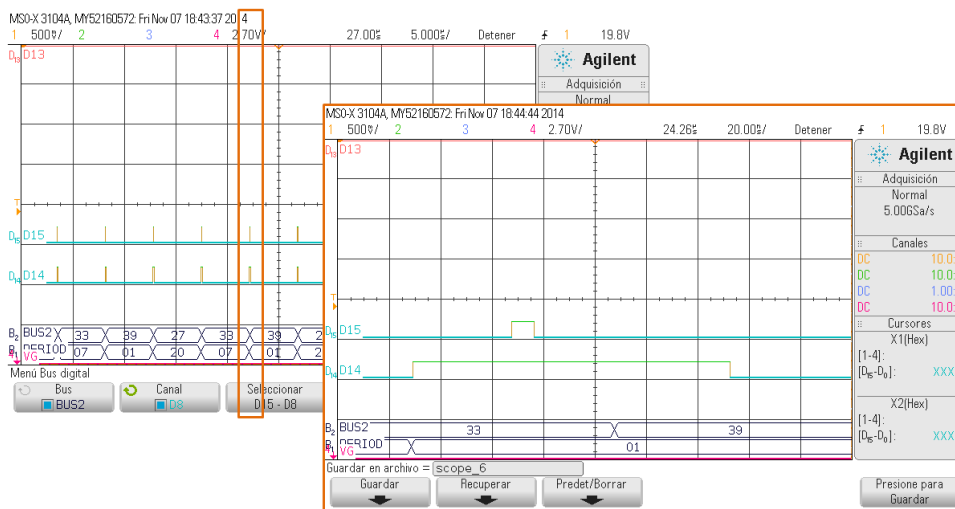


Figura 5.1: Sincronización del canal y resultado del ADC

En la figura 5.1, el BUS 1 y BUS 2 se corresponden con el resultado proporcionado por el ADC y el número del canal analizado respectivamente. Por otro lado, D15 y D14 se corresponden con *DATAVALID* y *ASSC_DONE* respectivamente.

5.2.1. Control de Lazo de Tensión

En primer lugar, para facilitar la implementación VHDL del control mediante un lazo de tensión (ver ecuación 4.31) se han utilizado señales reales, evitando de esta manera la dependencia con la señal de reloj.

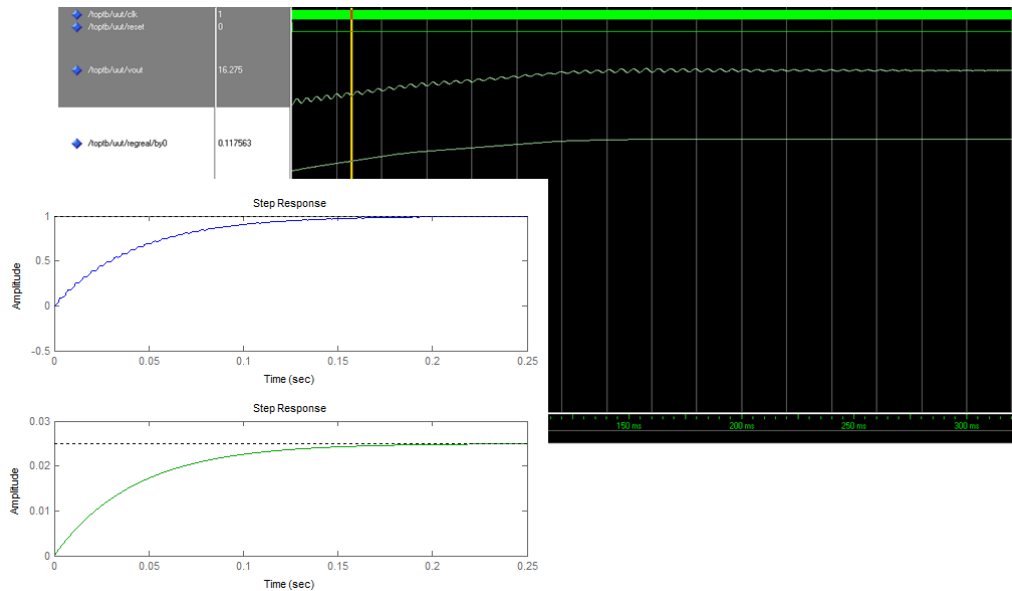


Figura 5.2: Resultados Sisotool vs ModelSim

Como se comprueba en la figura 5.2, la salida del sistema, es decir, la tensión de salida, y la respuesta del sistema ante un escalón, es decir, la señal de PWM, presentan un comportamiento similar en los resultados obtenidos con Sisotool durante la fase de diseño y los resultados obtenidos durante la fase de simulación con ModelSim. Este hecho corrobora la correcta implementación VHDL del regulador.

Los resultados de ModelSim de la figura 5.2 muestran cómo aplicando una tensión continua de 15 V a la entrada, el regulador consigue obtener una tensión constante igual a 20 V pasados los primeros 200 ms.

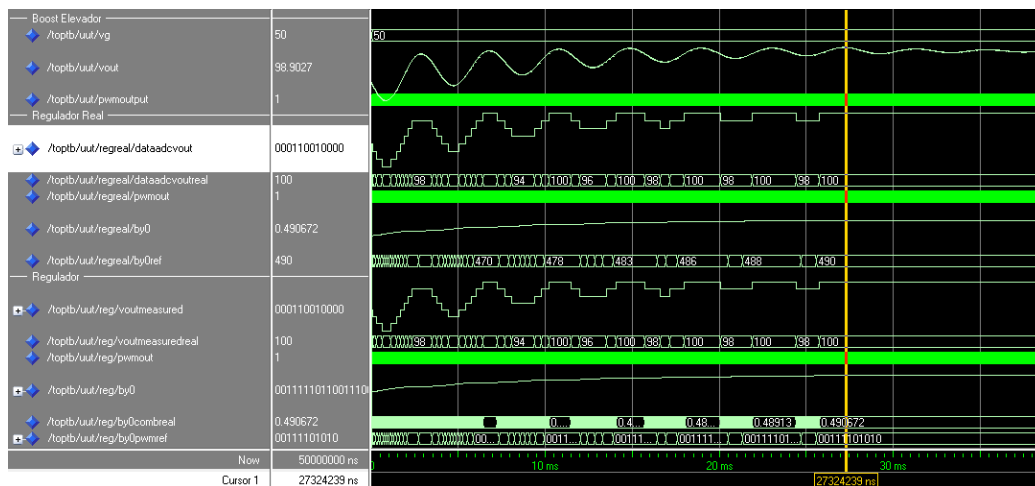


Figura 5.3: Comparativa Regulador real VS `std_logic_vector`

Una vez comprobado el correcto funcionamiento, se ha llevado a cabo la implementación del regulador con señales `std_logic_vector` teniendo en cuenta el formato QX.Y explicado anteriormente en el apartado 4.4.1.

Es posible asegurar el correcto funcionamiento del regulador basado en señales `std_logic_vector` mediante la comparación de estas señales con las señales reales obtenidas anteriormente (ver figura 5.3).

Para el control por medio del lazo de tensión se ha utilizado un tiempo de adquisición y conversión total del ADC de 10 μ s. De esta manera, cada periodo de actuación del lazo de tensión se obtiene una nueva muestra de la tensión de salida.

El código VHDL de ambos reguladores pueden encontrarse dentro del apartado Anexos.

5.2.2. Control de Doble Lazo

Es necesario tener en cuenta que en el caso concreto de control por medio del doble lazo ha sido necesaria la implementación de un bloque denominado *Selec* como consecuencia de la existencia de un único canal de ADC en la FPGA de Microsemi utilizada. En este caso en concreto, el bloque *Selec* determinará si la salida de datos del ADC (*ADC RESULT*) se corresponde con la medida de tensión de entrada, tensión de salida o corriente de entrada por medio del número de canal (*ADC CHNUMBER*).

El control de doble lazo cuenta con dos reguladores relacionados entre si (ver ecuaciones 4.40 y 4.39). La interacción de los lazos, produce la señal de salida de PWM que tiene como objetivo determinar el estado del mosfet asegurando así llegar a la tensión de salida de referencia.

Para comprobar el correcto funcionamiento del regulador, en primer lugar se han llevado a cabo pruebas con una tensión uniforme a la entrada. Al tratarse de un tensión fija, y que por tanto no tiene paso por cero, es necesario fijar un ciclo de trabajo que asegure su funcionamiento.

A continuación, se ha pasado a realizar la pruebas en lazo cerrado aplicando una tensión alterna rectificadora a la entrada.

Como ya se ha indicado, el objetivo del control por medio del doble lazo lo es conseguir una tensión de salida de 400 V a partir de una entrada de 230 V proporcionada por la red normal.

Simulación Lazo Abierto

Para las simulaciones en lazo abierto se ha utilizado una tensión de entrada uniforme igual a 200 V. Dada la imposibilidad de llevar a cabo la sincronización de los lazos tras el paso por cero, se ha fijado en ciclo de trabajo al 50 %.

Como se muestra en la figura 5.4, aplicando una tensión de entrada uniforme igual a 200 V y fijando el ciclo de trabajo al 50 % es posible obtener una tensión de salida, con un pequeño rizado, igual a 400V.

En la figura 5.4 también es posible diferenciar cómo cuando la señal de PWM tiene un estado alto (conmutador cerrado), se produce un incremento en la corriente de entrada. Por el contrario, cuando la señal de PWM tiene un estado bajo (conmutador abierto), la corriente por la bobina disminuye.

Simulación Lazo Cerrado

Una vez comprobado el correcto funcionamiento del lazo abierto es posible empezar con las simulaciones en lazo cerrado. En esta ocasión, se ha utilizado una tensión de entrada sinusoidal rectificadora siendo posible entonces la sincronización de los lazos tras el paso por cero de la tensión y corriente de entrada.

Debido a la existencia de un único ADC en la placa de desarrollo utilizada, se han llevado a cabo las conversiones de la tensión de entrada, corriente de entrada y tensión de salida sucesivamente.

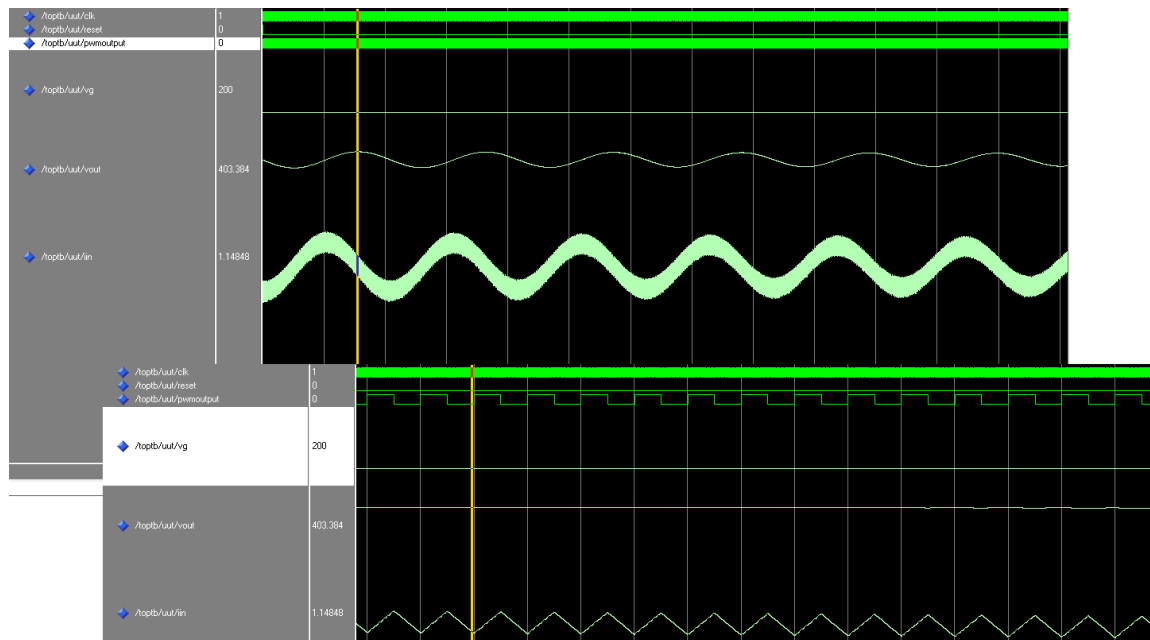


Figura 5.4: Simulación controladores utilizados $D=50\%$

Se debe tener en cuenta que el periodo de actuación del lazo de corriente es igual a $10\ \mu\text{s}$, por tanto, es obligatoria la captura y conversión de analógico a digital de la tensión y corriente de entrada una vez cada periodo de actuación. Por este motivo, para esta fase de simulación, se ha determinado que el mayor tiempo de adquisición y conversión posible del ADC para cada muestra de datos será igual a $3,33\ \mu\text{s}$.

En la figura 5.5 se puede observar la simulación obtenida tras la aplicación de unos tiempos de adquisición y conversión iguales a $3,33\ \mu\text{s}$. La simulación muestra un rizado de la tensión de salida de $\pm 12\text{V}$. Además, las señales de tensión de salida y corriente de entrada tienen un aspecto similar a las señales teóricas mostradas en la figura 4.8.

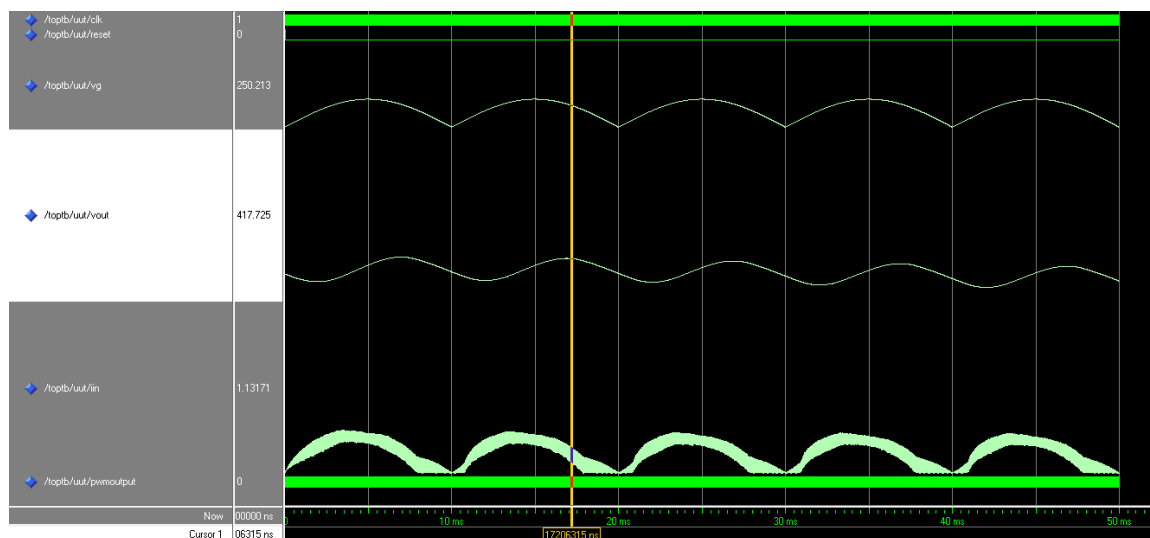


Figura 5.5: Simulación Lazo Cerrado - Tiempo de adquisición y conversión ADC = $3,33\ \mu\text{s}$

Sin embargo, tras la comprobación de los tiempos de adquisición y conversión del ADC mediante la herramienta Libero IDE, se ha observado que para una resolución de 12 bits, la mayor frecuencia de conversión posible es igual a $27,663\ \text{kps}$ por muestra (ver figura 5.6). Por

lo tanto, el ADC proporcionará una muestra cada $36,15 \mu\text{s}$, lo que se aleja mucho de los $3,3 \mu\text{s}$ deseados.

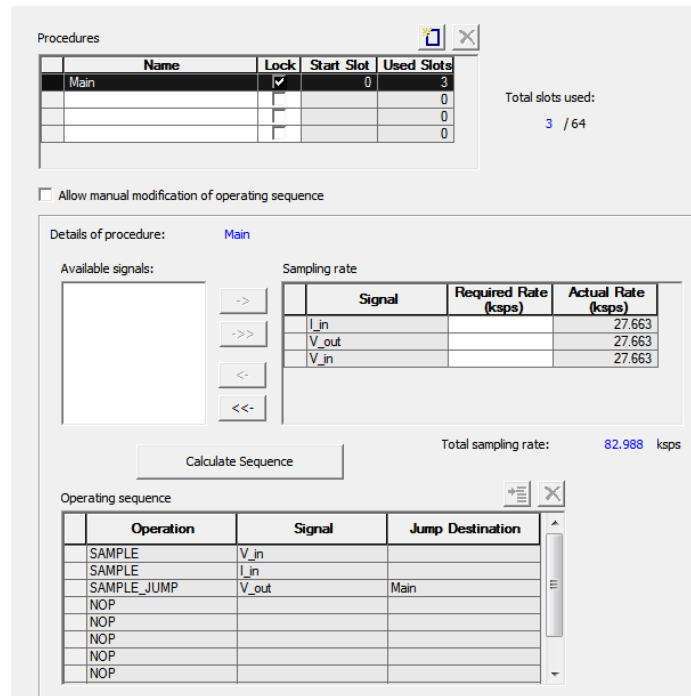


Figura 5.6: ASB - Secuencia de muestreo

Como se observa en la figura 5.7, aplicando unos tiempos de adquisición y conversión de $36,15 \mu\text{s}$ para cada canal, una vez pasado el periodo de estabilización, se obtiene una señal de corriente de entrada que además de no aproximarse a una forma sinusoidal contiene picos bastante amplios que llegan hasta los 6 A, lo que comparado con el valor esperado, 0,75 A de valor eficaz, resulta desproporcionado. Esto provoca que el rizado en la tensión de salida sea superior a los $\pm 12 \text{ V}$ esperados.

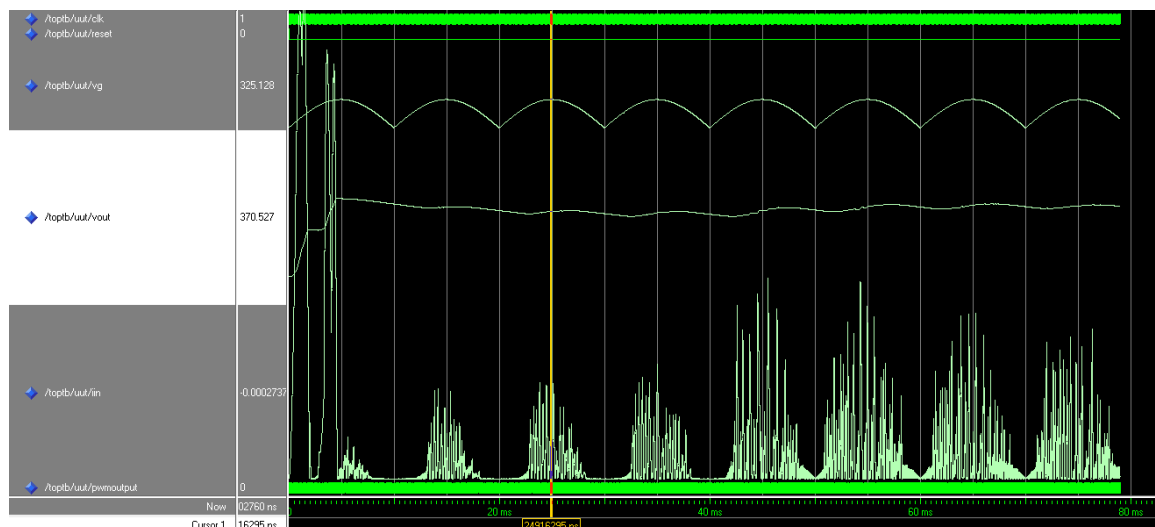


Figura 5.7: Simulación Lazo Cerrado - Tiempo de adquisición y conversión ADC = $36,15 \mu\text{s}$

Además, tal y como se indicó en el capítulo 1.2.3, para evitar la circulación de armónicos por la red y obtener así un elevado factor de potencia, la corriente de entrada debe tener una forma sinusoidal además de estar en fase con la tensión de entrada. Como se comprueba en la figura

5.7, aunque la tensión y corriente de entrada se encuentran en fase, esta última no presente una forma sinusoidal por lo que no puede llevar a cabo la corrección del factor de potencia.

Por último, como se muestra en la figura 5.8, los resultados obtenidos tras las simulaciones serán más próximos a los teóricos (ver figura 4.8) cuanto menor sea el tiempo de conversión del ADC. Se observa, que aunque en ambos casos la tensión y corriente de entrada se encuentran en fase, es en la figura con un tiempo de adquisición y conversión menor donde la corriente se aproxima más a una forma sinusoidal. Por tanto será esta última donde mayor factor de potencia se obtendría.

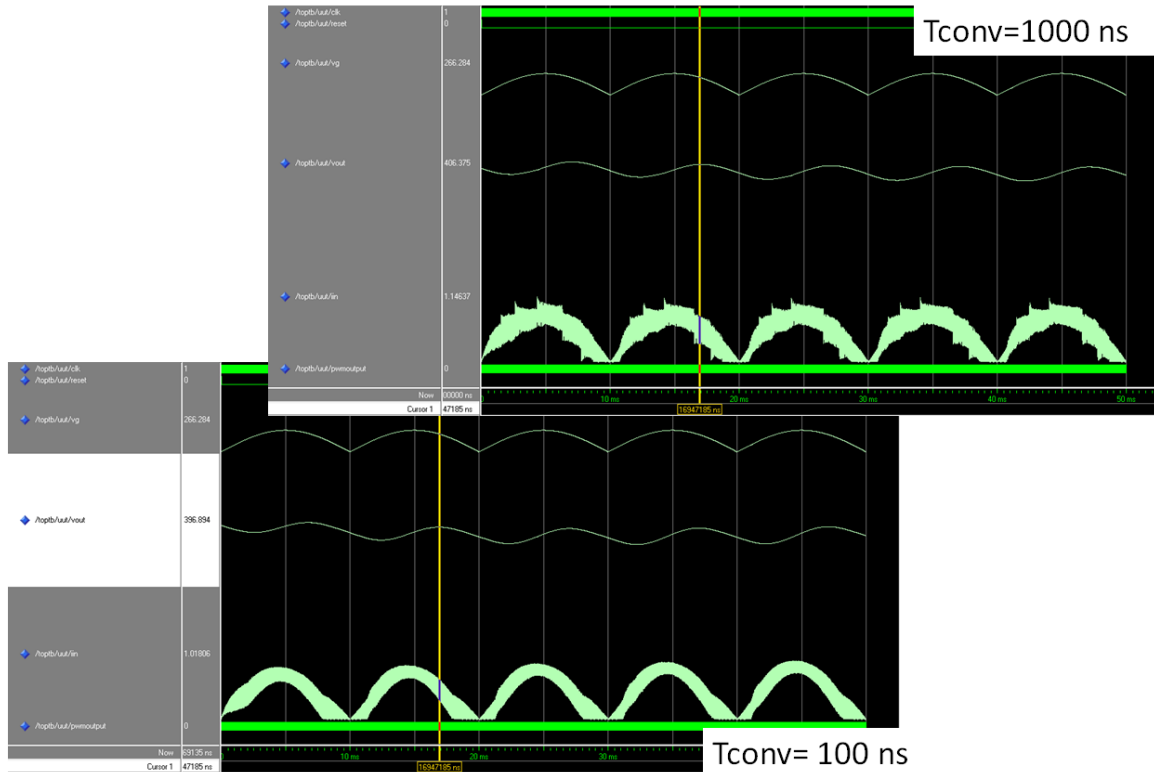


Figura 5.8: Simulación Lazo Cerrado - Tiempo de adquisición y conversión ADC = 1000/100 ns

6

Pruebas Experimentales

Después de realizar las simulaciones del control por medio de un único lazo de tensión y el control por medio del doble lazo, es momento de llevar a cabo las pruebas experimentales, las cuales serán detalladas en el presente capítulo.

En este capítulo también se describirán las limitaciones encontradas a la hora de usar la placa de desarrollo hardware utilizada.

Además de los resultados de las pruebas en lazo abierto y lazo cerrado para los diferentes tipos de control, en el presente capítulo también se especificarán las conexiones físicas realizadas para llevar a cabo las pruebas experimentales.

Por último, será en el siguiente capítulo donde se detallen las conclusiones obtenidas durante el desarrollo del presente proyecto.

6.1. Resultados Experimentales

Después de comprobar el correcto funcionamiento de los reguladores tras la fase de simulación es necesario pasar a las pruebas experimentales.

En esta ocasión no han sido necesarios los modelos VHDL creados para la fase de simulación sino que se ha utilizado el hardware específico: la FPGA de Microsemi y el convertidor Boost Elevador. Además de estos elementos, han sido necesarias fuentes de alimentación continua y alterna, una carga y un osciloscopio. Este último servirá para conocer el comportamiento del convertidor y por tanto de la regulación llevada a cabo.

Destacar, que una vez realizadas las pruebas experimentales se han encontrado limitaciones en la placa de desarrollo hardware no identificadas con anterioridad. En esta ocasión, utilizando una frecuencia de sistema igual a 100 MHz (máxima frecuencia proporcionada por el RC Oscillator), con un grado de segmentación máximo en el código VHDL, se presentan retardos superiores a los permitidos por la FPGA en alguna de las pistas. Por este motivo, la frecuencia de sistema utilizada finalmente ha sido 50MHz en lugar de 100 MHz como se tenía previsto al inicio de proyecto.

Finalmente, como se puede observar en la figura 6.1, utilizando una frecuencia de sistema igual a 50 MHz, el slack en una de las pistas es superior al permitido (-1,671 ns). El slack

obtenido por analizador de tiempos de la FPGA supone el peor de los casos, es por ello, que al tratarse de ns, al programarlo sobre la FPGA esta actúa sin tener problemas de tiempos.

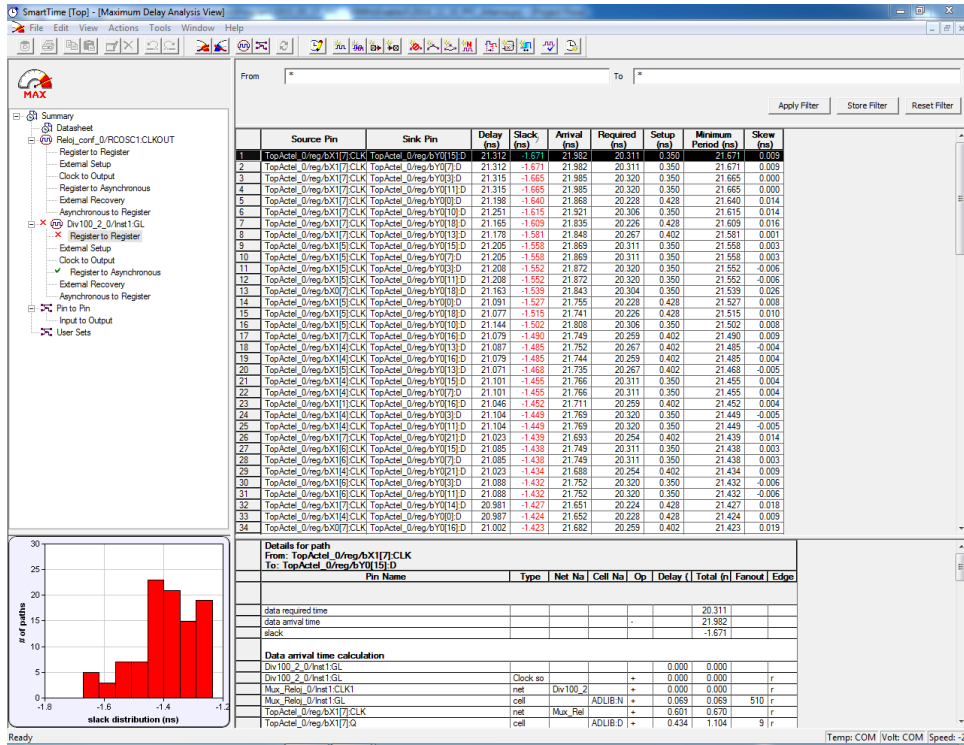


Figura 6.1: SmartTime

Anteriormente, utilizando un frecuencia de 100 MHz era posible observar grandes saltos en el valor del PWM, ya que era la pista que asigna el valor al PWM la que sufría retrasos. Sin embargo, utilizando una frecuencia de 50 MHz, no se han encontrado saltos, creciendo y decreciendo la señal de PWM de manera uniforme.

6.1.1. Control de Lazo de Tensión

Como se observa en la figura 6.13, para poder llevar a cabo la realización de la pruebas experimentales es necesaria la interconexión de los elementos hardware.

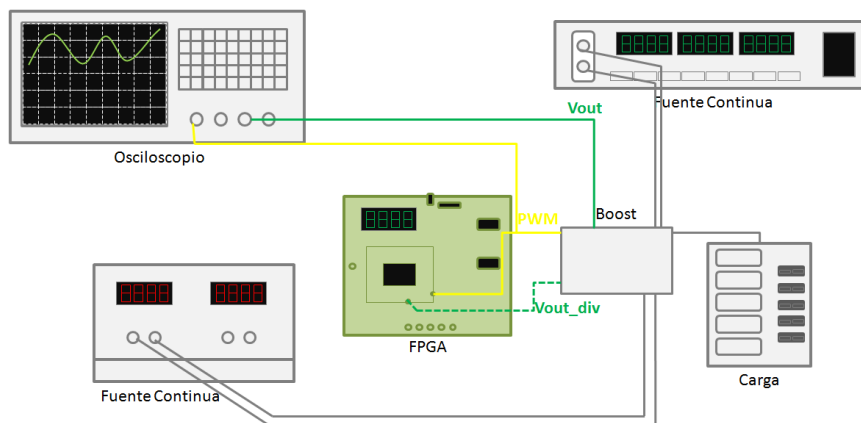


Figura 6.2: Esquema Conexionado: Control de Lazo de Tensión

En primer lugar, al convertidor es necesario conectarle la carga. Por otro lado, necesitaremos

diversas fuentes de alimentación continua. Mientras que una fuente se utilizará para alimentar al MOSFET con una tensión de 11 V, otra fuente de alimentación se conectará al convertidor para proporcionar la tensión de entrada.

A continuación, debe conectarse la tensión de salida del convertidor, tras su paso por un divisor de tensión, a una de las entradas del ADC de la FPGA utilizada. Por tanto, el ADC será el responsable de la conversión analógica digital de las muestras de tensión de salida.

De la misma manera, la señal de PWM, salida del regulador, se conectará al convertidor para que de esta manera pueda llevarse a cabo la regulación de la tensión de salida.

Por último, se utilizarán los canales del osciloscopio para observar los valores de PWM y tensión de salida, los cuales permitirán comprobar el correcto comportamiento del regulador.

Antes de la lectura del valor de tensión de salida en osciloscopio se debe tener en cuenta que las señales `std_logic_vector` han sido transformadas de manera que el formato QX.Y muestre los verdaderos valores. En este caso en concreto, la tensión de salida cuenta con un formato igual a Q10.2.

Resultados en Lazo Abierto

A continuación, en las figuras 6.3 y 6.4, se puede observar como aplicando un ciclo de trabajo del 50 % obtenemos una tensión de salida prácticamente igual al doble en la entrada.

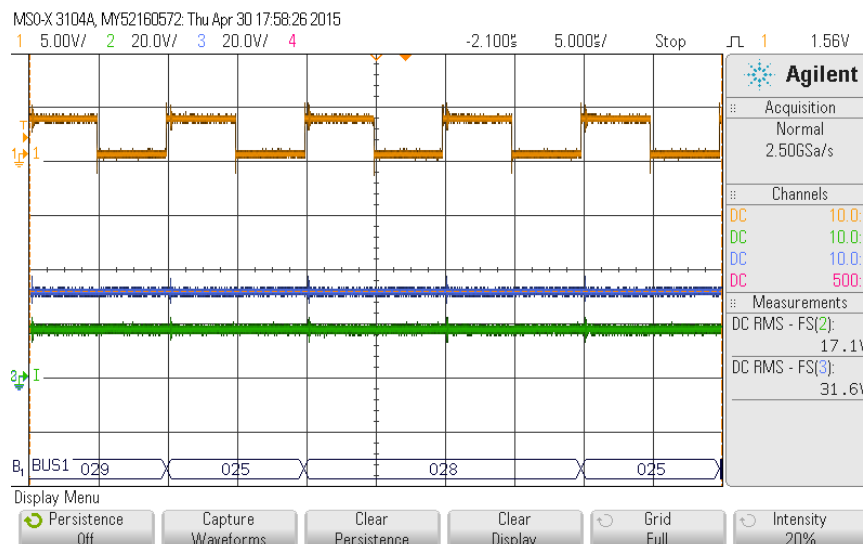


Figura 6.3: Control Lazo de Tensión: Lazo Abierto

En ambas figuras, las señales amarillas, azules y verdes del osciloscopio representan las señales de PWM, tensión de salida y tensión de entrada respectivamente. Por ello, en la figura 6.3, es posible determinar como para una tensión de entrada igual a 17,1 V se obtiene una tensión de salida de 31,6 V.

Del mismo modo, en la figura 6.4, se observa como para una tensión de entrada de 40,9 V se obtiene una salida de 79,0 V.

Resultados en Lazo Cerrado

Para el caso del lazo cerrado, es posible comprobar cómo los resultados obtenidos de la fase de simulación mediante la herramienta de ModelSim se corresponden con los resultados obtenidos

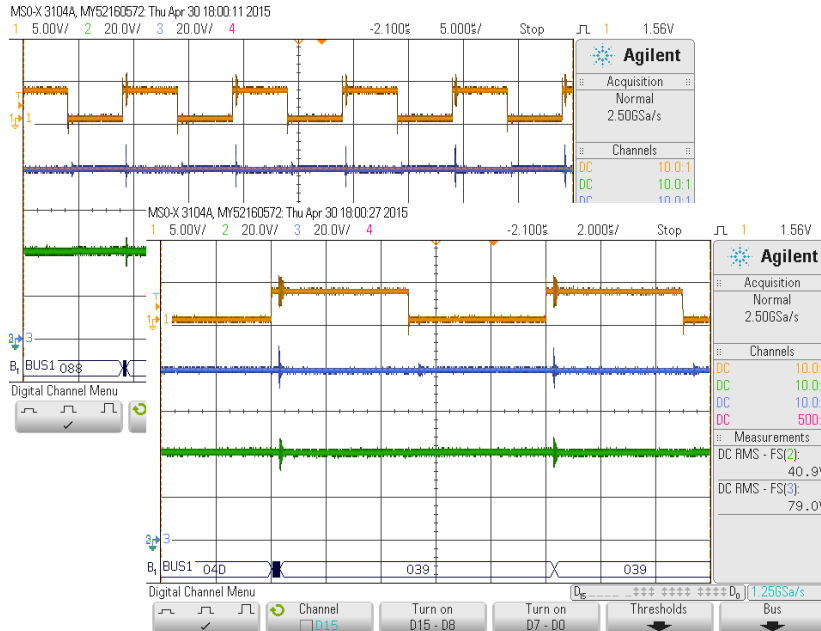


Figura 6.4: Control Lazo de Tensión: Lazo Abierto

en el osciloscopio.

En la figura 6.5 se puede apreciar cómo durante la fase de inicialización se produce un aumento progresivo de la tensión de salida hasta llegar a la tensión de referencia, en este caso en concreto 20 V, tras la aplicación de una tensión de entrada de 15 V. Los resultados del osciloscopio nos muestran cómo se llega a la regulación pasado 200 ms, resultados que coinciden con los obtenidos de mediante las herramientas Matlab y ModelSim.

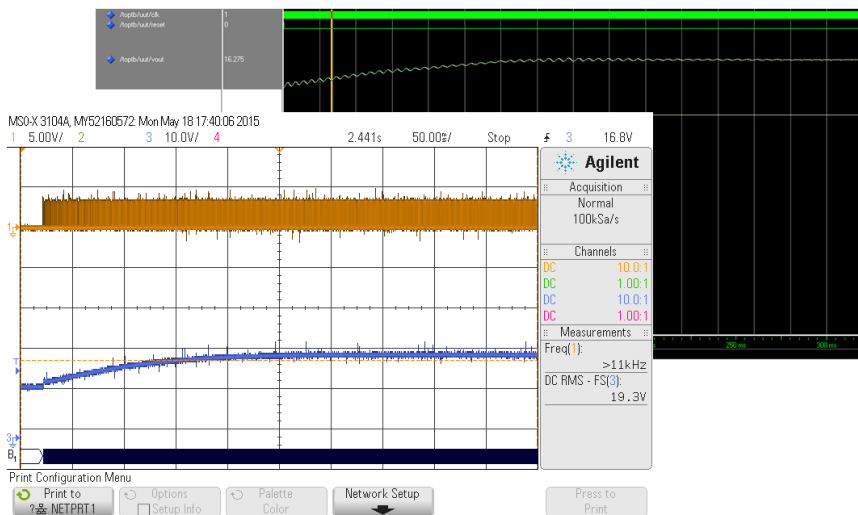


Figura 6.5: Control Lazo de Tensión: Lazo Cerrado

Por otro lado, a continuación se muestran los resultados obtenidos utilizando diferentes tensiones de referencia y valores de la carga. En las siguientes figuras, mientras la señal superior representará la señal de PWM, la señal inferior representará la tensión de salida del boost.

En la figura 6.6 se aprecia cómo aplicando una tensión de entrada igual a 15 V y fijando una tensión de referencia igual a 40 V, a la salida del boost utilizado es posible obtener 39,3 V con una señal de PWM aproximadamente al 70 %.

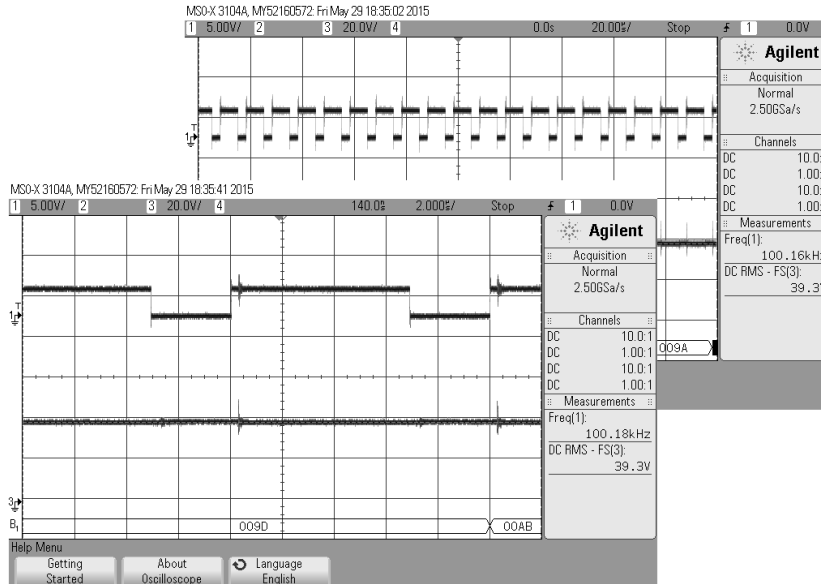


Figura 6.6: Simulación Lazo Cerrado - $V_{ref}=40\text{ V}$; $V_{in}=15\text{ V}$; $R=200\ \Omega$;

En la figura 6.7 se observa como para tensiones de entrada iguales a 19, 21 y 27 V también es posible obtener la tensión de referencia de 40 V. Las señales de PWM se corresponden con el 59 %, 52 % y 39 % respectivamente. De esta manera, se deduce que para tensiones de entrada menores que la mitad de la tensión de referencia contienen un señal de PWM por encima de 50 %. Del mismo modo, para tensiones de entrada superiores a la mitad del valor de referencia contienen señales de PWM menores al 50 %.

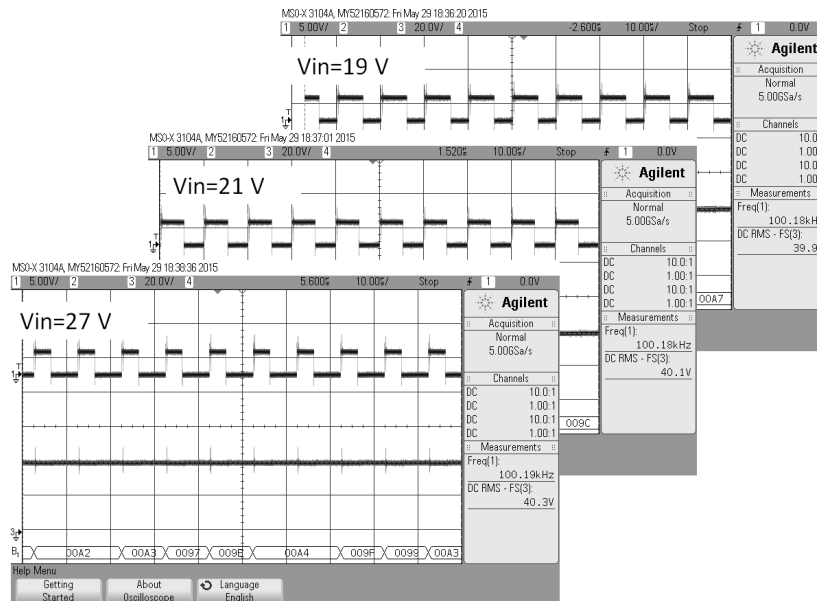


Figura 6.7: Simulación Lazo Cerrado - $V_{ref}=40\text{ V}$; $V_{in}=19;21;27\text{ V}$; $R=200\ \Omega$;

Los valores de corriente de entrada pico una vez llegado al punto de estabilización obtenidos para tensiones de entradas igual a 15, 19, 21 y 27 V se corresponden con 1,12, 0,87, 0,75 y 0,5 A respectivamente. Se observa que cuanto mayor es la tensión aplicada a la entrada del boost, menor es la corriente de entrada necesaria para llegar al valor de referencia.

Las figuras 6.8 y 6.9 muestran los resultados obtenidos con los mismos valores de tensión de

entrada y referencia, 15 V y 40 V respectivamente, pero aplicando una carga de 500Ω.

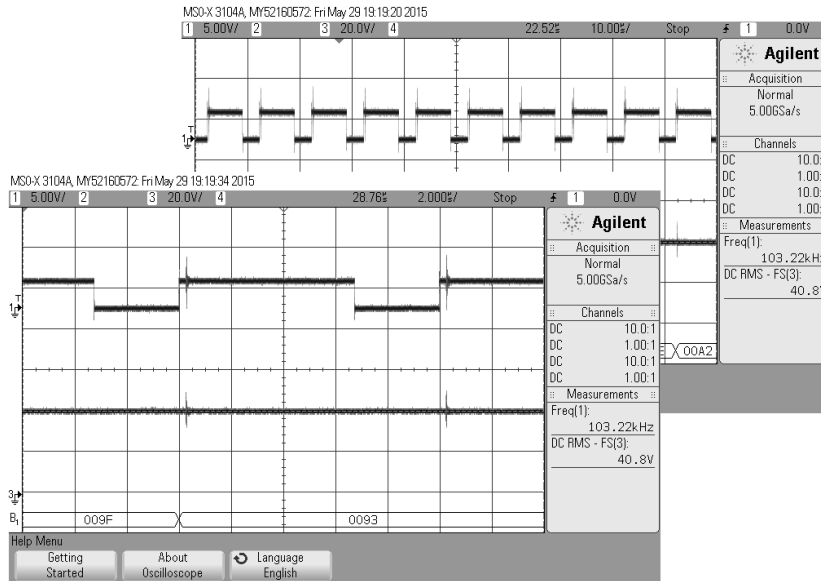


Figura 6.8: Simulación Lazo Cerrado - $V_{ref}=40$ V; $V_{in}=15$ V; $R=500$ Ω;

Como se puede observar, para tensiones de entrada iguales a 15, 19, 21 y 27 V se obtienen señales de PWM iguales al 70 %, 58 %, 52 % y 38 % respectivamente.

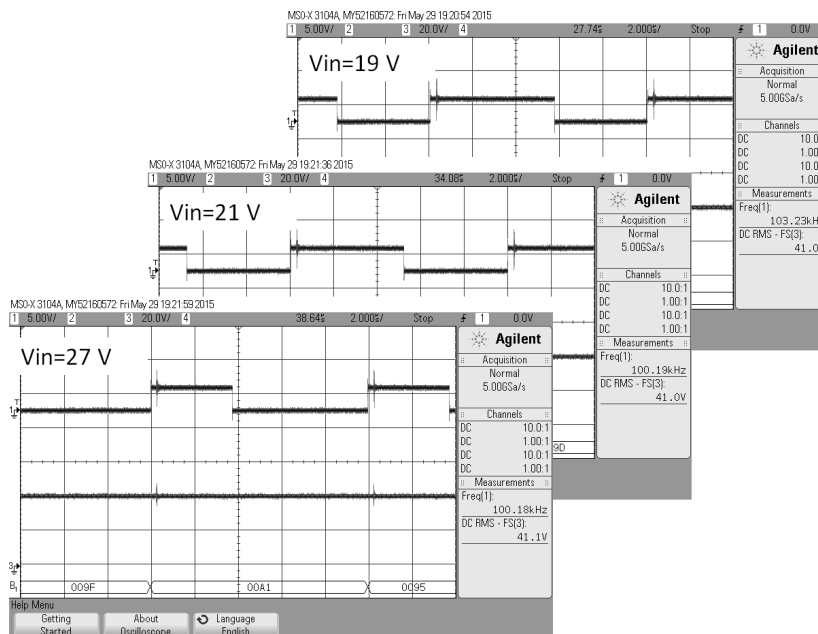


Figura 6.9: Simulación Lazo Cerrado - $V_{ref}=40$ V; $V_{in}=19;21;27$ V; $R=500$ Ω;

En esta ocasión, los valores de corriente de entrada pico una vez llegado el punto de estabilización de corresponden con 0,45, 0,35, 0,3 y 0,21 A para tensiones de entrada igual a 15, 19, 21 y 27 V respectivamente. Del mismo modo que sucedía para una carga igual a 200 Ω, para llevar al valor de tensión de salida de referencia, la corriente de entrada pico disminuye al aumentar la tensión de entrada.

Del mismo modo que se ha actuado con las pruebas llevadas a cabo para una tensión de referencia de 40 V, a continuación se muestran los resultados obtenidos con una tensión de

referencia igual a 60 V y diferentes tensiones de entrada y cargas.

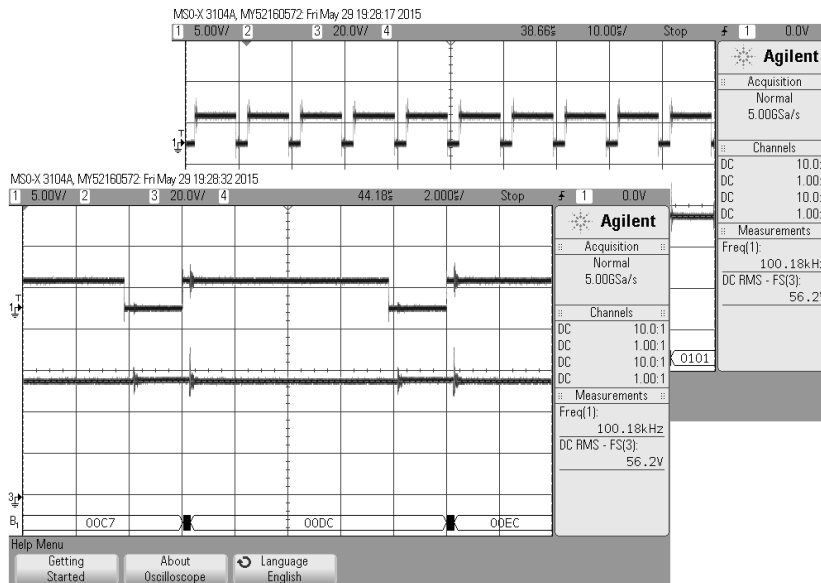


Figura 6.10: Simulación Lazo Cerrado - $V_{ref}=60$ V; $V_{in}=17$ V; $R=200$ Ω ;

La figuras 6.10 y 6.11 muestran los resultados obtenidos al aplicar una tensión de referencia igual a 60 V, una carga de 200 Ω y tensiones de entrada iguales a 17, 23, 31 y 37 V.

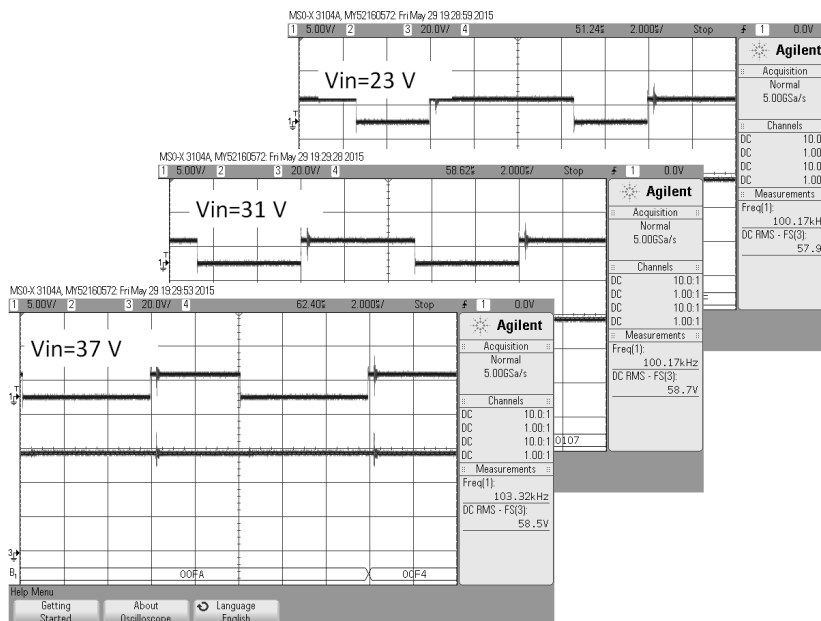


Figura 6.11: Simulación Lazo Cerrado - $V_{ref}=60$ V; $V_{in}=23;31;37$ V; $R=200$ Ω ;

Se observa que para estas tensiones de entrada: 17, 23, 31 y 37 V, se obtienen señales de PWM iguales a 78 %, 65 %, 56 % y 44 % respectivamente.

Estos resultados concuerdan con los mostrados en la figura 6.12, donde para una tensión de referencia de 60 V, una carga igual a 500 Ω y tensiones de entrada de 17, 23 y 37 V se obtienen señales de PWM iguales al 76 %, 64 % y 42 %.

Para el caso de la corriente, vuelve a suceder lo mismo, cuanto mayor sea la tensión de entrada aplicada menor será la corriente de entrada necesaria para a una tensión de salida igual

a la de referencia.

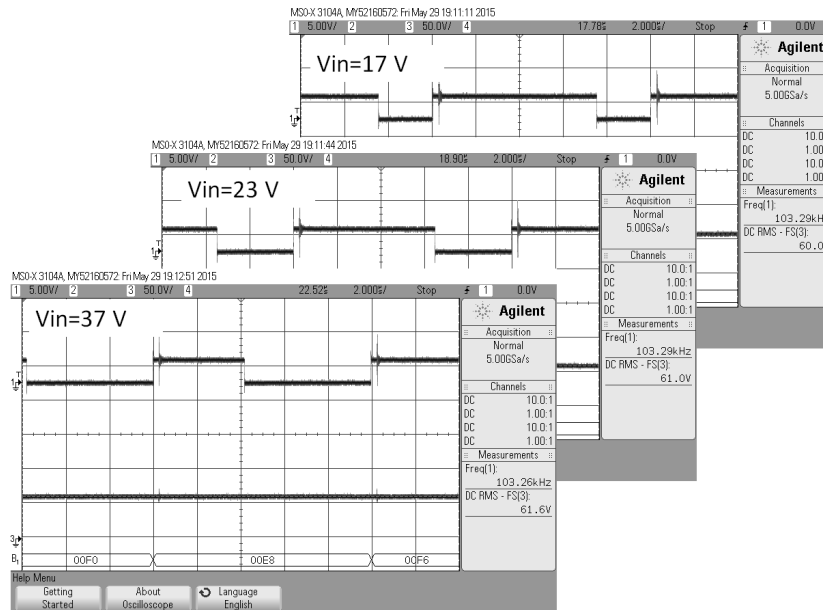


Figura 6.12: Simulación Lazo Cerrado - $V_{ref}=60\text{ V}$; $V_{in}=17;23;37\text{ V}$; $R=500\ \Omega$;

Finalmente, aunque las señales de PWM parecen tener el mismo valor independientemente de la carga, concluir que La señal de PWM tendrá un mayor rizado cuanto menor sea el valor de la carga. En consecuencia, la tensión de salida de boost presentará un mayor rizado cuanto menor sea la carga.

6.1.2. Control de Doble Lazo

Del mismo modo que sucede en el control mediante lazo de tensión, para el control por medio del doble lazo habrá que llevar a cabo las misma conexiones entre la FPGA y los elementos hardware utilizados.

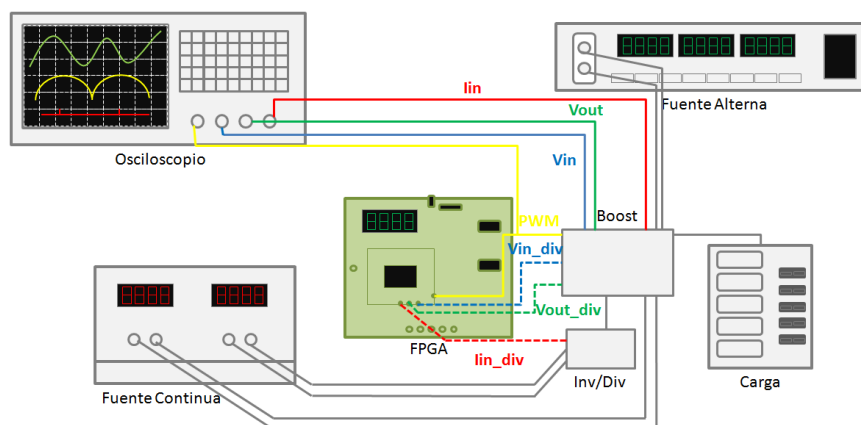


Figura 6.13: Esquema Conexionado Control de Doble Lazo

En este caso en concreto, además de la conexiones realizadas en el control por medio del lazo de tensión, serán también necesarias la alimentación del inversor utilizado en el divisor de tensión de la corriente de entrada por medio de una tensión de 3 V, la conexión de la tensión y corriente de entrada al ADC, y la conexión de la tensión y corriente de entrada al osciloscopio para su visualización.

Resultados en Lazo Abierto

Antes de llevar a cabo las pruebas experimentales del regulador en lazo cerrado se han llevado a cabo pruebas en lazo abierto para la posible detección de fallos.

A continuación, en la figura 6.14 se puede observar el resultado de la tensión de salida y corriente de entrada cuando es aplicado un $D = 50\%$ y una tensión de entrada de aproximadamente 33 V.

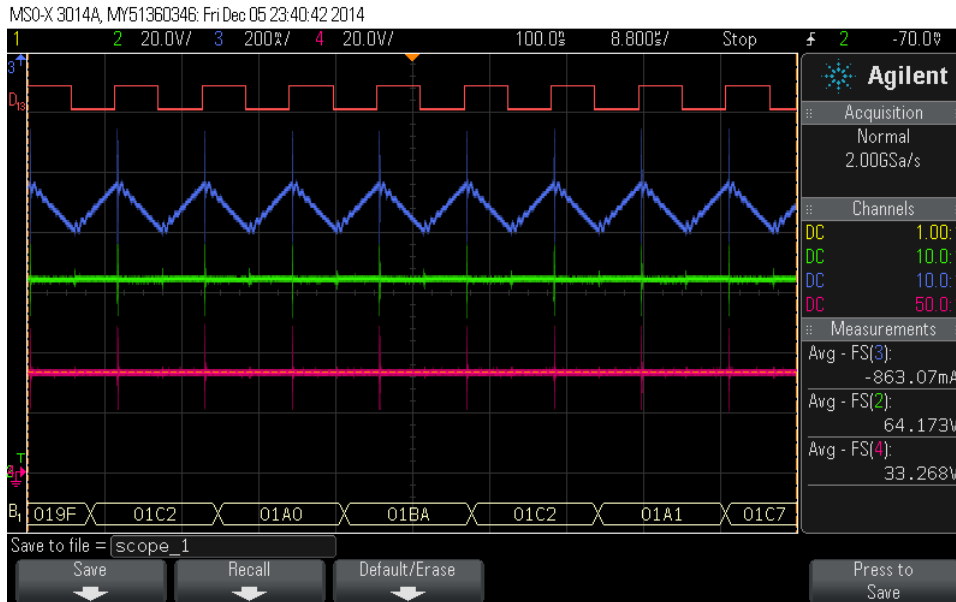


Figura 6.14: Control Doble Lazo: Lazo Abierto

Mientras que la señal roja representa la señal de PWM, las señales azul, verde y rosa representan la corriente de entrada, la tensión de salida y la tensión de entrada respectivamente.

Observando la figura 6.14, se puede comprobar el correcto funcionamiento del control realizado. Para una tensión de entrada uniforme igual a 33 V se obtiene una tensión de salida igual a 64 V. Además, del mismo modo que sucedía en la fase de simulación, la corriente de entrada aumenta cuando el pulsador se encuentra cerrado (señal de PWM en alto), y decrece cuando el pulsador se encuentra abierto (señal de PWM en bajo).

6.1.3. Resultados en Lazo Cerrado

Tras la comprobación del correcto funcionamiento del regulador en lazo abierto se procede a la realización de pruebas en lazo cerrado.

Como ya se anticipaba tras los resultados de las simulaciones del doble lazo en el apartado 5.2.2, debido a los tiempos de adquisición y conversión mínimos proporcionados por el ADC, no es posible llevar a cabo una corrección del factor de potencia.

En la figura 6.15 se verifica que tal y como se indicó en la fase de simulación, la forma de la corriente de entrada no se aproxima a una forma sinusoidal y por tanto no es posible llevar a cabo una corrección en el factor de potencia.

Aunque no es posible llevar a cabo la corrección del factor de potencia, la figura 6.15 muestra como aplicando una tensión de entrada igual a 200 V es posible obtener una tensión de salida en el rango de los 400 V.



Figura 6.15: Control Doble Lazo: Lazo Cerrado

7

Conclusiones y Trabajo Futuro

Después de la descripción de la tecnología utilizada, el boost elevador, la estrategia seguida, las simulaciones realizadas mediante ModelSim y los resultados observados a través del osciloscopio, en el presente capítulo se detallarán las conclusiones obtenidas del presente proyecto fin de carrera y los posibles trabajos futuros.

7.1. Conclusiones

El presente proyecto fin de carrera tiene como objetivo la regulación de la tensión de salida de un convertidor boost elevador a través de la placa de desarrollo hardware incluida en el “Actel Fusion Starter Kit”.

La placa de desarrollo hardware utilizada se encuentra caracterizada por incluir, además del módulo digital, un módulo analógico el cual incluye un conversor analógico a digital. Este conversor ha sido utilizado para llevar a cabo la conversión de las muestras analógicas que intervienen en el proceso de regulación.

La regulación se ha llevado a cabo por medio de sistemas de control en lazo cerrado a través de una modulación por ancho de pulso. Los sistemas de control utilizados en el presente proyecto son: control PWM por muestra de tensión y control PWM por muestra combinada tensión corriente.

El control por medio de lazo de tensión se basa en la obtención de una muestra directamente de la salida del circuito para después ser comparada con la tensión deseada o de referencia. De esta comparación se obtiene una señal de error capaz de generar la señal de PWM que actúa sobre el conmutador del boost obligando a este a variar su ciclo de trabajo.

La regulación por medio de este tipo de control no es inmediata y depende de la magnitud de la variación. Por tanto, el circuito necesita varios ciclos de conmutación para llevar a la señal de salida al nivel deseado.

En segundo lugar, en el control por medio del doble lazo (lazo de tensión y lazo de corriente), mientras el lazo de corriente regula la corriente de entrada, el lazo de tensión regula la tensión de salida. El lazo de tensión compara la tensión de salida con una tensión de referencia generando una señal de error que será utilizada como referencia en el lazo de corriente. Esta corriente se

compara a su vez con la corriente de entrada de modo que la salida de la comparación actúa sobre el ciclo de trabajo de la señal que proporciona el estado del conmutador.

Como consecuencia de los altos tiempos de conversión del ADC para cada canal en este control, $30 \mu\text{s}$, el presente proyecto no ha podido llevar a cabo la regulación de la tensión de salida sobre el boost elevador. Debido a que el periodo de actuación del lazo rápido, el lazo de corriente, se ha fijado en $10 \mu\text{s}$, para llevar a cabo la correcta regulación de la tensión de salida se debe obtener una muestra de la corriente y tensión de entrada en este periodo. En consecuencia, la señal de corriente de entrada no se aproxima a su forma sinusoidal teórica y por tanto no se puede llevar a cabo una corrección del factor de potencia.

El factor de potencia es un indicador cualitativo y cuantitativo del correcto aprovechamiento de la energía eléctrica. Para ello, se debe evitar la circulación de armónicos por la red (Norma EN61000-3-2) utilizando corrientes de entrada con forma sinusoidal en fase con la tensión de entrada.

Se puede concluir que aunque la placa de desarrollo utilizada incluye un módulo analógico y por tanto no es necesario llevar a cabo la sincronización entre esta y ADC externos, los tiempos de conversión que este proporciona no son suficientes para llevar a cabo la regulación mediante el control de doble lazo de sistemas electrónicos que cuentan con altas frecuencias de funcionamiento. Por el contrario, será posible la utilización de esta placa de desarrollo para sistemas mecánicos con frecuencias menores o eléctricos de frecuencias inferiores a 100 KHz .

Mediante el control por medio de un lazo de tensión será posible la regulación de sistemas con frecuencias de sistema elevadas. Sin embargo, será necesario que estos sistemas puedan esperar varios ciclos de conmutación hasta llegar al estado de regulación, es decir, que soporten la lenta respuesta que este tipo de control proporciona.

7.2. Trabajo Futuro

Siguiendo con esta línea de investigación surgen diferentes posibles mejoras que se pueden aplicar a este proyecto:

- Estudio de la posibilidad de reducir el tiempo de adquisición y conversión del ADC mediante la configuración *ADC only*.
- Mejora del Lazo de tensión
- Utilización de nuevas FPGAs con ADC integrado incluido de mayor velocidad, como la familia 2 de Xilinx.

Glosario de Acrónimos

- **ADC**:Analog to Digital Converter
- **CAD**:Conversor Analógico Digital
- **DAC**:Digital to Analog Converter
- **CDA**:Conversor Digital Analógico
- **ISP**:In System Programming
- **PLL**:Phase-Locked Loop
- **PLLs**:Phase-Locked Loops
- **LCD**:Liquid Crystal Display
- **JTAG**:Joint Test Action Group
- **LED**:Light Emitting Diode
- **LEDs**:Light Emitting Diodes
- **SAR**:Successive Approximation Register
- **IDE**:Integrated Design Environment
- **FPGA**:Field Programmable Gate Array
- **FPGAs**:Field Programmable Gate Arrays
- **MSB**:Most Significant Bit
- **AC**:Alternating Current
- **DC**:Direct Current
- **PWM**:Pulse Width Modulation
- **PFC**:Power Factor Correction
- **PF**:Power Factor
- **V_{in}**:Tensión de Entrada
- **V_{in1}**:Tensión de Entrada en Instante Anterior
- **I_{in}**:Corriente de Entrada
- **I_{in1}**:Corriente de Entrada en Instante Anterior
- **V_{out}**:Tensión de Salida

- **dt**:Diferencial de tiempo
- **L**:Bobina
- **C**:Condensador
- **R**:Resistencia
- **ASB**:Analog System Builder
- **Vref**:Voltaje de Referencia
- **FMSB**:Flash Memory System Builder
- **MCC**:Modo de Conducción Continuo
- **MCD**:Modo de Conducción Discontinuo
- **ASIC**:Application Specific Integrated Circuit

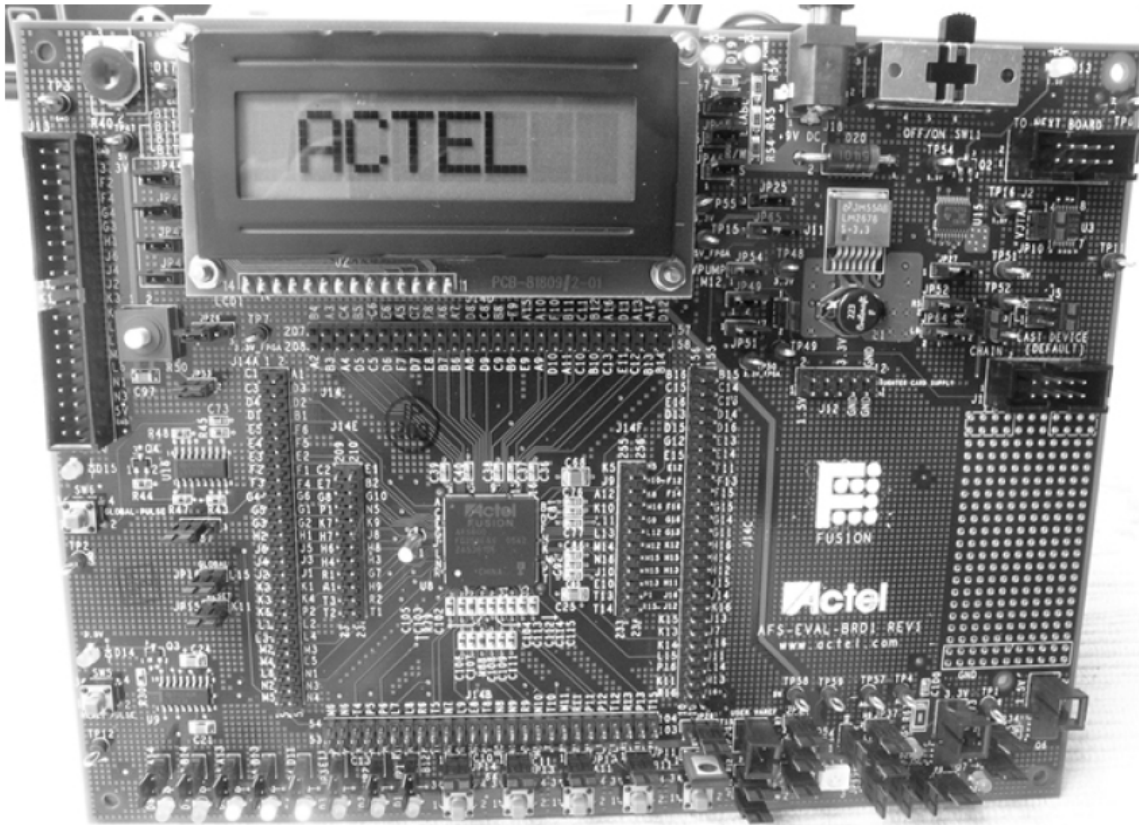
Bibliografía

- [1] Boylestad Nashelsky; *Electrónica: teoría de circuitos y dispositivos electrónicos*.
- [2] Peter J. Ashenden; *Digital design: an embedded systems approach using VHDL*. Morgan Kaufmann, 2007.
- [3] Senén Barro Ameneiro, Alberto J. Bugarin Diz; *Fronteras de la Computacion*.
- [4] Robert L. Boylestad; *Introducción al análisis de circuitos*.
- [5] http://ark.intel.com/es/products/75254/Intel-Xeon-Processor-E7-8857-v2-30M-Cache-3_00-GHz.
- [6] <http://cpuboss.com/cpu/Intel-Xeon-E7-8857-v2>.
- [7] <http://www.muycomputer.com/2015/02/05/transistores-siliceno-atomo-grosor>.
- [8] Randeep Singh Soin, Franco Maloberti; *Analogue-digital ASICs: Circuit Techniques, Design Tools and Applications*.
- [9] Norman Einspruch; *Application Specific Integrated Circuit (ASIC) Technology*.
- [10] <http://www.xilinx.com/fpga/asic.htm>.
- [11] <http://only-vlsi.blogspot.com.es/2008/05/fpga-vs-asic.html>.
- [12] Philip Simpson; *FPGA Design: Best Practices for Team-based Design*.
- [13] Steve Kilts; *Advanced FPGA Design: Architecture, Implementation, and Optimization*.
- [14] <http://www.microsemi.com/company/about-us>.
- [15] http://www.schneiderelectric.es/documents/local/productos-servicios/distribucion_electrica/guia_instalacion-mejora-factor-potencia-filtrado-armonicos.pdf.
- [16] Charles A. Schuler; *Electrónica, principios y aplicaciones*.
- [17] Benjamin C.Kuo; *Sistemas de control automático*. Séptima Edición.
- [18] Elizabeth Gpe., Lara Hdz., José Manuel Rocha Núñez; *Ingeniería de control* . Séptima Edición.
- [19] Carlos Valdivia Miranda; *Sistemas de Control Continuos y Discretos*
- [20] Muhammad H. Rashid, Muhammad H. Rasid; Virgilio González, Pozo Agustín Suárez Fernández; *Electrónica de potencia: circuitos, dispositivos y aplicaciones*
- [21] Jesús Arrillaga Garmendia, Luis Ignacio Eguiliz Morán; *Armónicos en Sistemas de Potencia*.
- [22] *IEC 61000-3-2:1995 + A.1:1997 + A.2:1998, EMC Part 3-2: Limits for harmonic current emissions*

- [23] Microsemi; *Fusion Starter Kit User's Guide*.
- [24] Microsemi; *Actel Fusion. Key Fusion Features Programmable Analog User's Guide*.
- [25] Microsemi; *Actel SmartFusion Programmable Analog User's Guide*.
- [26] Microsemi; *Fusion and Extended Temperature Fusion FPGA Fabric User's Guide*.
- [27] Microsemi; *Libero IDE User's Guide*.
- [28] Microsemi; *SmartGen, FlashROM, Analog System Builder and Flash Memory System Builder User's Guide*.
- [29] Sally Myles and Mario Di Bernardo; *Preventing multiple switchings in Power Electronics Circuits: Effects of the Latch on the nonlinear Dynamics of the DC-DC Buck Converter*.
- [30] Amos Gilat; *Matlab: una introducción con ejemplos prácticos*.
- [31] <http://es.mathworks.com/help/control/ref/controlsystemdesigner.html;jsessionid=35b4340768a873ac73b40>
- [32] Dingyu Xue, YangQuan Chen, Derek P. Atherto; *Linear Feedback Control: Analysis and Design with MATLAB*.
- [33] <http://www.mentor.com/products/fv/modelsim/>.
- [34] Peter J. Ashenden; *Digital design: an embedded systems approach using VHDL*.
- [35] Oscar GM; *Iniciación a Verilog*.
- [36] <http://www.synopsys.com/Tools/Implementation/FPGAImplementation/FPGASynthesis/Pages/SynplifyI>
- [37] <http://www.microsemi.com/products/fpga-soc/design-resources/design-software/synplify-pro-me#overview>.
- [38] Microsemi; *FlashPro for Software User's Guide*.
- [39] Microsemi; *FlashPro User's Guide*.
- [40] D.W.Hart; *Electrónica de Potencia* Prentice Hall, 2001.
- [41] Alberto Pigazo López, Víctor Manuel Moreno Sáiz; *Sistemas electrónicos de potencia en el buque*.
- [42] Robert W. Erickson, Dragan Macsimovic; *Fundamentals of Power Electronics*. Second Edition.
- [43] Javier Rollón Rodríguez; *Trabajo Fin de Máster: Convertidor DC/DC Elevador (Boost) con control de corriente y tensión*. Universidad Pública de Navarra, 2014
- [44] Alejandro García Talón; *Control Digital de Fuentes de Alimentación*. Universidad Autónoma de Madrid, 2009
- [45] <http://datasheets.maximintegrated.com/en/ds/MAX4165-MAX4169.pdf>.
- [46] <http://facultad.bayamon.inter.edu/arincon/Introduccion%20al%20Control%20Automatico.pdf>.
- [47] Ángel de Castro; *Tesis Doctoral: Aplicación del control digital basado en hardware específico para convertidores de potencia conmutados*. Universidad Politécnica de Madrid, 2003

- [48] Mario Alberto Perez, Analía Perez Hidalgo, Elisa Perez Berenguer. Universidad Nacional de San Juan, Departamento de Electrónica y Automática; *Introducción a los Sistemas de Control y Modelo Matemático para Sistemas Lineales Invariantes en el Tiempo*.
- [49] Alan V. Oppenheim, Alan S. Willsky; *Señales y Sistemas* Segunda Edición.
- [50] Alan V. Oppenheim, Ronald W. Schaffer; *Tratamiento de señales en tiempo discreto* Tercera Edición.
- [51] Jose Mari González de Durana, Euskal Herriko Unibertsitatea; *Automatización de Procesos Industriales*.
- [52] Mauricio Améstegui Moreno, Universidad Mayor de San Andrés; *Apuntes de Control PID*.
- [53] Chung-Chieh Fang; *Exact orbital stability analysis of static and dynamic ramp compensations in DC-DC Converters*.

Anexo: FPGA



Dentro del presente Anexo se incluye la descripción técnica de la FPGA de Microsemi utilizada. En ella se puede encontrar el contenido de la FPGA junto con las características más relevantes.

En ultimo lugar se ha incluido el esquema a alto nivel de la placa.

Hardware Components

This chapter describes the hardware components of the Fusion Evaluation Board.

Fusion Evaluation Board

The Fusion Evaluation Board consists of the following:

- Wall-mount power supply connector, with switch and LED indicator
- Jumpers to select either 1.5 V or 3.3 V for I/O Bank0 or Bank1
- 10-pin 0.1"-pitch programming connector compatible with Altera® connections
- 40 MHz oscillator and two independent manual clock options for global reset and pulse
- Eight LEDs (driven by outputs from the device)
- Jumpers (allow disconnection of all external circuitry from the FPGA)
- Two monostable pulse generator switches ("global" and "reset")
- Four switches (provide input to the device)
- Potentiometer for variable analog input
- Large alphanumeric LCD to facilitate detailed message outputs from the FPGA application
- Multicolor LED for illustrating PWM fan control and temperature measurement
- 1.5 V and 3.3 V MOSFET-driven fan control circuits with shrouded headers for external fans

For further information, refer to the following appendices:

- [Appendix A – "FG256 Package Connections for AFS600 Devices" on page 79](#)
- [Appendix B – "Board Schematics" on page 89](#)

Detailed Board Description and Usage

The Fusion Evaluation Board has various advanced features that are covered in later sections of this chapter. The Fusion architecture provides access to a one-chip flash FPGA solution containing both analog and digital components, including a built-in flash drive.

Note that the AFS600-FG256 Actel FPGA is soldered directly to the board. The Fusion Starter Kit board is available only in a directly soldered configuration. Socketed configurations are not available.

Full schematics are available on the Starter Kit Tutorial CD supplied with the Starter Kit or from the [Actel website at http://www.actel.com/documents/Fusion_Starter_Kit_SS.pdf](http://www.actel.com/documents/Fusion_Starter_Kit_SS.pdf). The electronic schematics can be enlarged on screen to show more detail than will be shown when this manual is printed.

Power Supplies

A 9 V power supply is provided with the kit. There are two power supply components on the board to provide 1.5 V and 3.3 V to the Fusion FPGA. An additional +5 V voltage bank is provided for use by the LCD module and in illustrating an analog voltage input to the analog aspects of the Fusion Evaluation Board.

The external +9 V positive-center power supply provided to the board via connector J18 goes to a voltage regulator chip, U11. This regulator has been protected against application of a reverse supply voltage by a reverse polarity protection diode. As soon as the external voltage is connected to the board, the red "power applied" LED (D19, the only red LED on the board) illuminates to indicate that an external supply has been connected to the board. As soon as switch SW11 is moved to the ON position (to the right, as labeled on the board "OFF/ON"), the disabling ground signal is removed from pin 7 of U11, and the regulator begins to provide power at its output.

The U11 switching voltage regulator provides a dedicated 3.3 V supply at its output. The board's 3.3 V supply feeds a separate regulator that delivers 1.5 V via U15. Although all Fusion FPGAs can also support 1.8 V and 2.5 V I/O

standards, these voltages are not provided on the Fusion Evaluation Board. The 1.5 V supply is required for the core voltage of the Fusion family, and the 3.3 V supply is required for extended I/O bank capability, such as for LVTTL. The presence of these voltages is indicated by two yellow LEDs (D13 for 1.5 V and D11 for 3.3 V) located at the top right of the board. Each LED is labeled with the voltage it represents and its component identifier. Both voltages are selectable on I/O banks 0 and 1 (the two northernmost banks on a Fusion device). Note that only the larger Fusion devices (AFS600 and AFS1500) have five I/O banks.

The 3.3 V supply can also be used to provide the VPUMP programming voltage. This VPUMP voltage may be provided to the chip during programming by connecting a FlashPro3 programmer to the J1 interface and selecting VPUMP from the FlashPro v4.1 (or later) programming software, and may also be provided directly to the chip from the board. The user simply leaves the JP54 jumper in place to apply the 3.3 V supply to the VPUMP pin (M12) of the FG256-packaged FPGA. If both FlashPro3 and the board are selected to provide VPUMP, the connection on the board will override; FlashPro3 will detect that a voltage is available, issue an information message in the programming software, and then tristate its VPUMP output pin, allowing the board to provide all the power.

The board must be powered up during programming, as the chip's core voltages must be provided and VJTAG must be detected by the FlashPro3 programmer before it can set its JTAG signal voltages to the right level. The value of VJTAG can be set to 1.5 V or 3.3 V on the board by setting the position of jumper JP27 to join pins 1 and 2 for 1.5 V and pins 2 and 3 for 3.3 V. It is recommended that VJTAG be left at 3.3 V on the Fusion Evaluation Board because an interboard buffer chip is used that has limited low-voltage capability and needs the higher setting of 3.3 V to ensure good signals at the output of the buffer chip array U3.

The LCD has its own dedicated 5 V power supply, all components of which, including the regulator U20, are mounted on the circuit board underneath the LCD module. A yellow LED (D17) representing 5 V supply availability is positioned at the top left of the board.

The external +9 V power supply is rated at 2.2 A maximum. On page 1 of the dedicated schematics and in [Figure B-4 on page 93](#), it will be noted that the 3.3 V supply is rated at 5 A maximum. The derived power supply of 1.5 V is rated at 2 A maximum, and the LCD 5 V power supply is rated at 500 mA.

The components at U11 (LM2678S-3.3) and U20 (LM2674M-5.0) are rated for an input voltage range of +8 V to +40 V, so a wide range of power supplies may be used with the board with no worry of over-voltage conditions occurring from inadvertent usage of the wrong power supply. It is expected that the voltage provided will be positive at the center pin of connector J18 and grounded on the outside. There is protection on the Fusion Evaluation Board for reverse voltages to prevent damage, but correct polarity must be provided for the board to function. Note that greater heating of the regulator chips will occur with higher voltages. Actel recommends that only the included power supply or an equivalent substitute be used with the Fusion Starter Kit, as it has been rated for the Fusion Evaluation Board and any Actel daughter cards that may be attached to the board.

Daughter Card Power Supply Connections

Limited power may be supplied by the Fusion Evaluation Board to a daughter card. The connector for the daughter card (header J12) is shown on page 5 of the dedicated schematics and in [Figure B-7 on page 96](#). All the FPGA voltages of 1.5 V and 3.3 V are provided to the daughter card via a 12-pin, 0.1"-pitch connector. The reason for 12 pins is that this is compatible with the 12 pins used on the ProASIC®3 Starter Kits and makes the daughter cards potentially compatible between the kits. The voltages are arranged with no-connection pins interspersing the voltage pins to prevent use of a jumper to inadvertently short a supply rail to ground by connecting differing supply rails together. This is not to protect the power supply regulators, as these will go high-impedance when an over-voltage condition is detected, but to protect the FGPA from a higher voltage being unintentionally applied to the 1.5 V core. Three of the 12 pins are ground pins to provide more than sufficient current return capability for future Actel daughter cards that will work with the Fusion Evaluation Board. The remaining pins are no-connection to prevent accidental shorting.

Power Supplies and Chaining Boards Together

There is a special note to be made of VPUMP connections when chaining boards together. This is detailed here. Actel recommends that the reader, unless experienced with Actel Starter Kit boards, return to this section after reading [“Programming the AFS-EVAL-BRD1 with a FlashPro3” on page 11](#).

When joining multiple Fusion or ProASIC3 Starter Kit boards together via the chain programming connection, connect J2 on one board to J1 on the next board in the chain using a standard 10-pin, 0.1"-pitch programming cable. Ideally, twisted-pair ribbon cabling should be used for this connection. The length of the cable should be kept as short as possible, as multiple boards connected to form a JTAG chain of Fusion and ProASIC3/E and/or ARM®-enabled Fusion or ProASIC3 devices will provide much greater noise pick-up and may degrade the TCK clock for devices remote from the FlashPro3 programmer. Actel recommends that VJTAG be set at 3.3 V to help with signal integrity when chaining boards together. On all boards the jumper at JP10, if connected, is used for providing VJTAG to a downstream board that needs to know what VJTAG setting is being used by your board (e.g., some arbitrary board of your own design that you wish to supply with the VJTAG voltage used by the Fusion component). In most normal cases, this shunt should be disconnected. The shunt normally in this location can be safely stored across pins 11 and 12 or 9 and 10 of the J12 daughter card power supply connector. For particularly long chains, the value of TCK used during programming should be reduced. The lowest value it can take is 1 MHz.

To date, only one revision of the Fusion Evaluation Board has been produced. This first version of the board is labeled REV1 next to the AFS-EVAL-BRD1 label on the board. If the Fusion Evaluation Board is connected in a chain to ProASIC3/E boards, the documentation relating to the ProASIC3/E board should be read in detail to determine the appropriate connection setting, depending on the version of the ProASIC3/E board being used. The following notes are only for the Fusion Evaluation Board.

Procedure for Rev1 Fusion Evaluation Boards

To determine if the board is a Rev1 board:

To identify a Rev1 board, examine the front of the board and look for the part number located just beneath the large Actel logo silkscreened on the board. The part number will be "AFS-EVAL-BRD1" followed by "REV1".

To chain Rev1 boards together:

All boards, starting from the board nearest the FlashPro3 programmer, should have the shunt moved from the default location connecting pins 3 and 4 of the J5 header to connect pins 1 and 2 of the J5 header. On the board and schematic this is labeled "CHAIN" (pins 1 and 2) and "LAST DEVICE (DEFAULT)" (pins 3 and 4). Only the very last board in the chain should have the shunt remaining across pins 3 and 4 of the J5 connector.

When connecting these boards together via a connection from J2 of one board to J1 of another board, VPUMP will be connected from one board to another. When powering up one board with a connector in place, notice that the 1.5 V and 3.3 V LEDs will light on the board to which no power has been applied, and the FPGA on that board, if programmed, will start operating. This is clearly an inappropriate situation for a large chain of boards. This is caused by having connector JP54 supplying VPUMP to other boards in the chain, as VPUMP is itself connected to the 3.3 V supply output that is used to generate the other FPGA voltages on a board. To prevent VPUMP from being used as the 3.3 V supply, you should remove the shunt that is in place on the JP54 connector to force JP54 open-circuit. To prevent loss of the shunt, it may safely be stored on the J12 header for the daughter card power supply, as it is impossible to cause a short by joining any adjacent pins.

Programming the AFS-EVAL-BRD1 with a FlashPro3

The base board used for all Fusion starter kits is the AFS-EVAL-BRD1.

In an AFS EVAL KIT the Actel part number for the board is AFS-EVAL-BRD600-SA. The part number indicates that the board is fitted with an AFS600-FG256 part, directly soldered (-SA) to the board.

Connecting the FlashPro3 Programmer to the Board

Connect the FlashPro3 programmer to your computer via the USB cable. Follow the instructions in the *FlashPro User's Guide* to install the software and connect to the FlashPro3 programmer. The amber (yellow) power LED on the FlashPro3 should be illuminated at this stage. If it is not, recheck the procedure in the FlashPro user's guide until you obtain a steady amber (yellow) power LED illumination.

Make sure the board power switch SW11 is in the OFF position and only the red board external power LED is illuminated on the board.

Connect the FlashPro3 programmer to the board via the 10-pin programming cable supplied with the programmer hardware. The connector to use on the board is labeled “FP3” and is the lower shrouded and keyed header J1. The pin 1 location on the cable indicated by the red ribbon running along the side of the cable will be on the left side as it enters into the board. After connecting the FlashPro3 programmer, using the FlashPro software, select **Analyze Chain** from the **File** menu. If all is well, the appropriate device ID for the Fusion part will appear in the software display on the PC. If you suspect a JTAG communication issue, try changing the VJTAG voltage to 3.3 V using JP27.

Programming or Re-Programming the Example Design

On the Fusion Starter Kit CD, you will find a *Designer* directory containing a STAPL file for programming the target design. Select the *TOP_AFS.STP* file from the CD and use that as the STAPL file in the FlashPro v4.1 software. Clicking **PROGRAM** will erase, program, and verify the part. The total programming time is approximately 2 minutes and 30 seconds.

Jumpers for Isolating Switches and LEDs from the FPGA

The many jumpers on the Fusion Evaluation Board are provided to allow the user to disconnect various switch combinations or LEDs from the FPGA I/O banks.

The jumpers are shown in the schematic and are labeled on the top-layer silkscreen as JP*, where * is a number. All jumpers are also labeled with the FPGA I/O pin number to which they are connected, e.g., JP54, for 3.3 V connection of VPUMP to the FPGA, is labeled “M12”, which indicates that it is connected to pin M12. Similarly, SW4 has a jumper above it called JP14 that is labeled “C10”, indicating that SW4 is connected through to pin C10 of the FPGA when this jumper is in place.

[Figure 2-1 on page 14](#) shows the board’s silkscreen layer overlaid with a grid, and [Table 2-1 on page 15](#) describes the function of each jumper and its location.

By disconnecting jumpers JP11, JP12, JP13, and JP14, momentary push button switches SW1, SW2, SW3, and SW4, respectively, can be disconnected from the FPGA such that the I/O pins B11, A11, C11, and C10, respectively, may be used for other purposes. Disconnecting jumpers JP1–JP8 will disconnect LEDs D1–D8 from FPGA I/O pins B12, C13, E11, D11, B13, A13, B14, and A14, respectively.

Momentary push button switches SW5 and SW6, for applying a reset pulse and a global pulse, are connected via jumpers JP15 and JP16 to I/Os K11 and L15, respectively. Again, all labeling is clearly shown on the silkscreen.

The LCD also has associated jumpers for its data, namely JP41, JP43, and JP44, located on the top left side of the board. These are connected to I/O pins E2, E3, F5, and F6, respectively. The LCD control signals Enable, R/W, and RS are provided from I/O pins D3, E5, and D1 via jumpers JP47, JP45, and JP46, respectively.

Analog Circuitry Provided on the Board

A variety of analog inputs and outputs are provided on the Fusion Evaluation Board to show what Fusion can do as a single-chip solution. These components are detailed below and in [Table 2-1 on page 15](#), enumerating all possible connectivity. As with all other components, the FPGA pins may be disconnected from these components to allow the board to be used for other functions.

A potentiometer (R50) is provided on the board for simple user variation of analog input. This input is fed to the AV0 pad.

A multicolor LED, U1, is provided and has three individual color components that can be controlled by a set of AG pins (AG6 – Blue, AG7 – Orange, AG8 – Green). By using the analog outputs, PWM control as well as individual switch control can be effected. The multicolor LED is enabled with a p-channel MOSFET (Q5), where pad AG3 is connected to the gate of the MOSFET.

Motor Control

The Fusion Starter Kit board provides connections to two n-channel MOSFETs, Q6 and Q7, that are used to control current fed via the 1.5 V supply and the 3.3 V supply to an externally connected fan. Fans can be connected to the shrouded headers, JP72 and JP73, mounted on the board. An RC network is connected across each of these shrouded headers to filter out the back-EMF pulse that will occur when suddenly switching on or off a motor winding. Current is being measured via the AV and AC pads in each case by monitoring the voltage drop across a 0.1 Ω resistor (R62 for 1.5 V and R66 for 3.3 V) in each case. The AV and AC pads are tolerant to ± 12 V DC; however, the voltage spike from a motor winding being switched off will be much greater than this, so an RC “snubber” network is required to dampen such a voltage pulse and prevent it from reaching the FPGA inputs. Failure to apply an RC circuit across a motor when measuring the current through the field winding via direct connection to the FPGA will almost certainly result in voltages outside the maximum tolerance range of the FPGA’s analog inputs, which may result in catastrophic failure of the FPGA.

On the Fusion Evaluation Board, the n-channel MOSFETs are connected to a positive voltage supply. Because they are driven by the AG pads, their operation is inverse to normal. When AG pads on the FPGA are active, they float down or up to 0 V. When inactive they are tristated and switch off an n-channel MOSFET that is connected to the positive supply rail when activated. Normally, n-channel MOSFETs use the AG gate-drivers when the n-channel MOSFET is connected to a negative voltage supply. In this case, activating the AG gate-driver pad to 0 V will switch on the n-channel MOSFET.

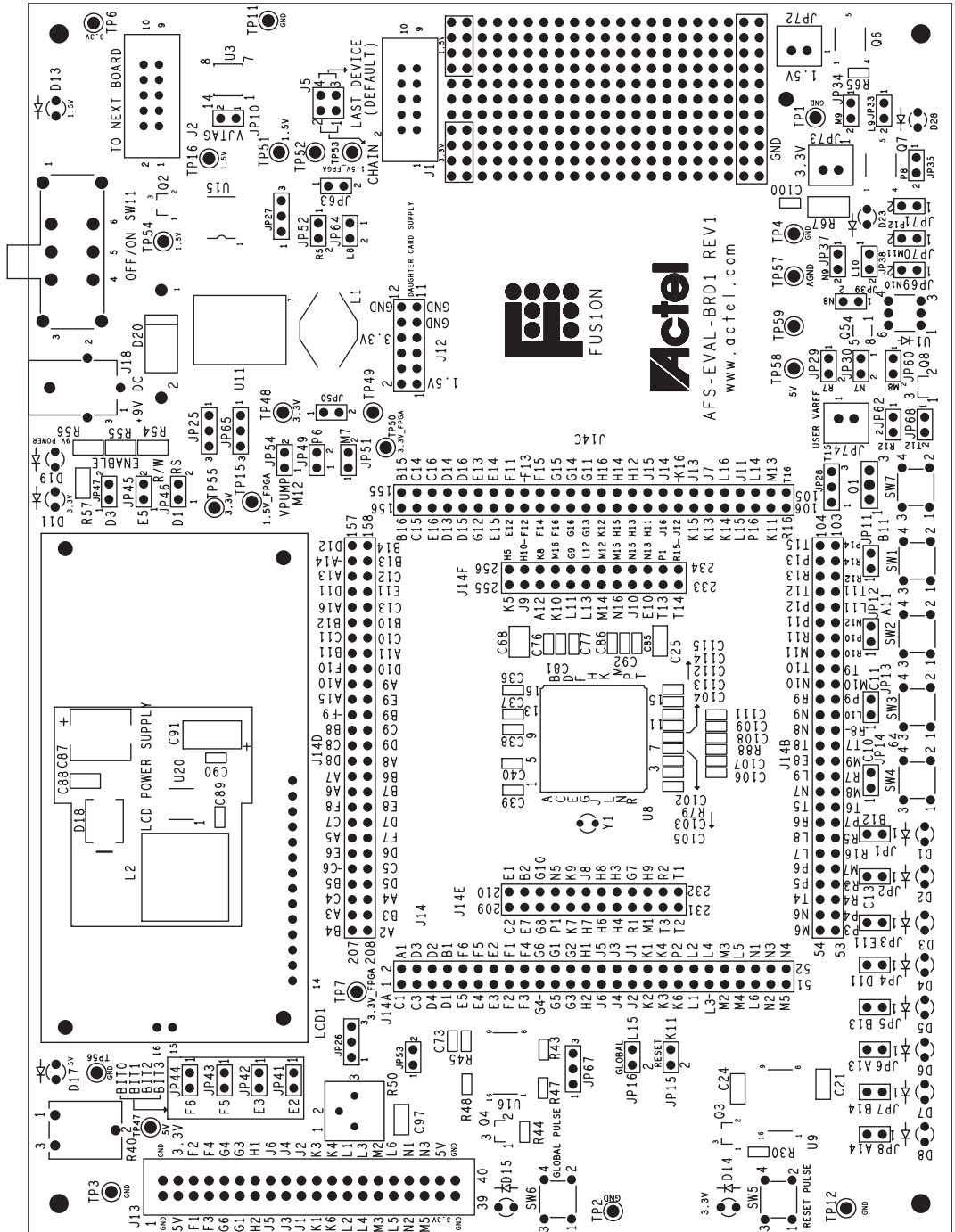


Figure B-1 · Top View of Fusion Evaluation Board

Anexo: Modelo VHDL Convertidor Boost Elevador

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;
use IEEE.std_logic_arith.all;
use IEEE.math_real.all;

entity BoostConverterReal is port(
    Clk : in std_logic;
    Reset : in std_logic;
    Mosfet : in std_logic; -- On = '1', off = '0'
    Vg : in real;
    Ir : in real;
    Iin : out real;
    Vout : out real);
end BoostConverterReal;

architecture Behavioral of BoostConverterReal is

    constant C : real := 0.0001; -- 10 uF
    constant L : real := 0.001; -- 1.1 mH
    constant dt : real := 0.00000001;
    signal iL : real := 0.0;
    signal voutAux : real := 0.0;
    signal iLAdd, voutAuxAdd : real := 0.0;

    --constant VOINIT : real := 0.0;
    constant VOINIT : real := 400.0;
    constant ILINIT : real := 0.0;

    constant dtL : real := dt/L;
    constant dtC : real := dt/C;

begin

    Iin <= iL;
    Vout <= voutAux;

    SWITCHMUX: process(Mosfet, Vg, Ir, iL, voutAux)
        -- Selection (multiplexer) of values to be added to input current and output voltage
    begin
        if Mosfet = '1' then -- Closed switch
            iLAdd <= Vg;
            voutAuxAdd <= -(Ir);
        else -- Open switch
            if iL > 0.0 then -- CCM
                iLAdd <= (Vg - voutAux);
                voutAuxAdd <= (iL - Ir);
            else -- DCM
                iLAdd <= 0.0;
                voutAuxAdd <= -(Ir);
            end if;
        end if;
    end process SWITCHMUX;

    DIFFEQ: process(Clk, Reset)
        -- Update of Vout and Iin each clock cycle
    begin
        if Reset = '0' then
            voutAux <= VOINIT;
            iL <= ILINIT;
        elsif rising_edge(Clk) then
            iL <= iL + iLAdd*dtL;
            voutAux <= voutAux + voutAuxAdd*dtC;
        end if;
    end process DIFFEQ;

end Behavioral;
```


Anexo: Modelo VHDL ADC

```
library WORK;
use WORK.MyConvPack.all; -- Funcion floor
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;
use IEEE.numeric_std.all;
use IEEE.math_real.all;
use IEEE.std_logic_arith.all;

entity ModeloAdcActel is port(
  Clk : in std_logic;
  Reset : in std_logic;
  VaRef : in real;
  VgAnalogIn : in real;
  VoutAnalogIn : in real;
  IinAnalogIn : in real;
  --ChNumber : in std_logic_vector(4 downto 0);
  --AdcStart : out std_logic;
  --AdcReset : in std_logic;
  --Purdown : in std_logic;
  --VaRefSel : in std_logic;
  --Mode : in std_logic_vector(3 downto 0);
  --STC : in std_logic_vector(7 downto 0);
  --TVC : in std_logic_vector(7 downto 0);
  Datavalid : out std_logic;
  --Calibrate : out std_logic;
  --Busy : out std_logic;
  --Sample : out std_logic;
  ASSC_DONE : out std_logic;
  AdcResults1 : out std_logic_vector(7 downto 0);
  AdcResults2 : out std_logic_vector(7 downto 0);
  AdcResults3 : out std_logic_vector(7 downto 0);
  AdcResults : out std_logic_vector(7 downto 0);
  ChNumber : out std_logic_vector(4 downto 0));
end ModeloAdcActel;

architecture Behavioral of ModeloAdcActel is

  signal VgSampleAndHold, VoutSampleAndHold, IinSampleAndHold : real;
  signal VgDataOut, IinDataOut : std_logic_vector(7 downto 0) := (others => '0');
  --signal VoutDataOut : std_logic_vector(7 downto 0); -- Empieza en Voutref para evitar transitorios de simulación
  signal VoutDataOut : std_logic_vector(7 downto 0) := "11001000"; -- Empieza en Voutref para evitar transitorios de simulación
  signal DatavalidS : std_logic;
  signal cntAdcStart : std_logic_vector(10 downto 0);
  constant VREFMAX : real := 4.00;

  signal ChNumberS : std_logic_vector(4 downto 0);
  signal AdcStartS : std_logic;

begin

  ChannelGen: process
    begin
      loop
        ChNumberS <= "00001";
        wait for 10 ns;
        AdcStartS <= '1';
        wait for 10 ns;
        AdcStartS <= '0';
        wait for 9.98 us;
        ChNumberS <= "00010";
        wait for 10 ns;
        AdcStartS <= '1';
        wait for 10 ns;
        AdcStartS <= '0';
        wait for 9.98 us;
        ChNumberS <= "00011";
        wait for 10 ns;
        AdcStartS <= '1';
        wait for 10 ns;
        AdcStartS <= '0';
        wait for 9.98 us;
        end loop;
    end process ChannelGen;

  --AdcStart <= AdcStartS;
  ChNumber <= ChNumberS;

  VgADCPR : process
```

```

begin
  if VgAnalogIn < 0.0 then
    VgSampleAndHold <= 0.0; -- Se captura dato
  elsif VgAnalogIn >= VREFMAX then
    VgSampleAndHold <= VREFMAX - 0.0001;
  else
    VgSampleAndHold <= VgAnalogIn; -- Se captura dato
  end if;

  wait for 10 us;
  VgDataOut <= conv_std_logic_vector(floor( (VgSampleAndHold/VREFMAX) * 2.0**8), 8); -- Dato válido
end process;

VoutADCPR : process
begin
  if VoutAnalogIn < 0.0 then
    VoutSampleAndHold <= 0.0; -- Se captura dato
  elsif VoutAnalogIn >= VREFMAX then
    VoutSampleAndHold <= VREFMAX - 0.0001;
  else
    VoutSampleAndHold <= VoutAnalogIn; -- Se captura dato
  end if;

  wait for 10 us;
  VoutDataOut <= conv_std_logic_vector(floor( (VoutSampleAndHold/VREFMAX) * 2.0**8), 8); -- Dato válido
end process;

IinADCPR : process
begin
  if IinAnalogIn < 0.0 then
    IinSampleAndHold <= 0.0; -- Se captura dato
  elsif IinAnalogIn >= VREFMAX then
    IinSampleAndHold <= VREFMAX - 0.0001;
  else
    IinSampleAndHold <= IinAnalogIn; -- Se captura dato
  end if;

  wait for 10 us;
  IinDataOut <= conv_std_logic_vector(floor( (IinSampleAndHold/VREFMAX) * 2.0**8), 8); -- Dato válido
end process;

Channel : process (Clk, Reset)
begin
  if Reset='0' then
    AdcResults <= (others => '0');
  elsif rising_edge(Clk) then
    if ChNumberS = "00001" then
      AdcResults <= VgDataOut;
    elsif ChNumberS = "00010" then
      AdcResults <= VoutDataOut;
    elsif ChNumberS = "00011" then
      AdcResults <= IinDataOut;
    else
      AdcResults <= (others => '0');
    end if;
  end if;
end process;

Dvalid : process (Clk, Reset, AdcStartS)
begin
  if Reset='0' then
    DatavalidS <= '0';
    cntAdcStart <= (others => '0');
  elsif rising_edge(Clk) then
    cntAdcStart <= cntAdcStart + '1';
    if AdcStartS = '1' then
      cntAdcStart <= (others => '0');
      DatavalidS <= '0';
    elsif cntAdcStart > "01110000100" then --900 ciclos de reloj
      DatavalidS <= '1';
    end if;
  end if;
end process;

Datavalid <= DatavalidS;
AdcResults1 <= VgDataOut;
AdcResults2 <= VoutDataOut;
AdcResults3 <= IinDataOut;

end Behavioral;

```

Anexo: Código VHDL Regulador Lazo de Tensión Real

```
library IEEE;
use IEEE.std_logic_1164.ALL;
use IEEE.std_logic_signed.all;
use IEEE.std_logic_arith.all;
use IEEE.NUMERIC_STD.all;
use IEEE.math_real.all;

entity RegulatorReal is port (
    Clk : in std_logic;
    Reset : in std_logic;
    --DataValid : out std_logic;
    --DataADCVg : in std_logic_vector(11 downto 0); -- Q9.3 without sign bit
    --DataADCIin : in std_logic_vector(11 downto 0); -- Q3.9 without sign bit
    DataADCVout : in std_logic_vector(11 downto 0); -- Q10.2 without sign bit
    Ind1 : in std_logic;
    Ind2 : in std_logic;
    Ind3 : in std_logic;
    Ind4 : in std_logic;
    Ind5 : in std_logic;
    Ind6 : in std_logic;
    Ind7 : in std_logic;
    Ind8 : in std_logic;
    PwmOut : out std_logic);
end RegulatorReal;

architecture Behavioral of RegulatorReal is

    Function PasarReal(din : std_logic_vector(12 downto 0); ndec: integer) return real is
        variable val : integer;
        variable result : real;
    begin
        val := conv_integer(din);
        result := real(val) / real(2**ndec);
        return result;
    end PasarReal;

    Function Ciclo(din : real; max_c : integer; min_C : integer ) return integer is
        variable result : integer;
    begin
        if din < 0.05 then
            result := min_c ;
        elsif din >= 0.95 then
            result := max_c;
        else
            --result := integer((real(max_c)/real(min_c)) * din);
            result := integer (floor (din * 999.0));
        end if;
        return result;
    end Ciclo;

    --signal DataADCVgReal : real := 0.0;
    --signal DataADCIinReal : real := 0.0;
    signal DataADCVoutReal : real := 0.0;
    signal VoutRef : std_logic_vector (11 downto 0) := (others=> '0');
    signal VoutRefReal : real := 0.0;

    constant B0 : real := 0.00000667572;
    constant B1 : real := 0.000000834465; --0.125*0.00000667572;
    signal bY0 : real := 0.0;
    signal bY1 : real := 0.444;
    signal bX0 : real := 0.0;
    signal bX1 : real := 0.0;
    signal bX0E0 : real := 0.0;
    signal bX1E1 : real := 0.0;

    signal counter : integer;
    signal bY0Ref : integer := 444;

    constant COUNTER_MAX : integer := 999;
    constant COUNTER_SUP : integer := 995;
    constant COUNTER_INF : integer := 5;

begin

    process(Clk, Reset)
    begin
```

```
        if Reset = '1' then
            counter <= 0;
        elsif rising_edge(Clk) then
            if counter < COUNTER_MAX then
                counter <= counter+1;
            else
                counter <= 0;
            end if;
        end if;
    end process;

    voutRef <= "00" & Ind1 & Ind2 & Ind3 & Ind4 & Ind5 & Ind6 & Ind7 & Ind8 & "00"; --Q10.2

    process(counter)
    begin
        if counter = 1 then
            --DataADCVgReal <= PasarReal (DataADCVg,3);
            --DataADCIinReal <= PasarReal (DataADCIin,9);
            DataADCvoutReal <= PasarReal ('0'&DataADCvout,2);
            VoutRefReal <= PasarReal ('0'&VoutRef,2);
            bY1<=bY0;      -- bY1 << bY0
            bX1<=bX0;      -- bY1 << bY0
        end if;
    end process;

    -- bI0 = VoutRef - DataADCVgReal;

    bX0 <= VoutRefReal - DataADCvoutReal;

    -- bY0 <= bY1 + B0*bI0;

    bX0B0 <= bX0 * B0;
    bX1B1 <= bX1 * B1;
    bY0 <= bY1 + bX0B0 - bX1B1;

    -- bY0Ref = fuint(bY0)

    bY0Ref <= Ciclo(bY0, COUNTER_MAX, COUNTER_INF);

    PwmOut <= '1' when counter < bY0Ref else '0';

end Behavioral;
```

Anexo: Código VHDL Regulador Lazo de Tensión

```
library IEEE;
use IEEE.std_logic_1164.ALL;
use IEEE.std_logic_signed.all;
use IEEE.std_logic_arith.all;
use IEEE.NUMERIC_STD.all;
use IEEE.math_real.all;

entity Regulator is port (
    Clk : in std_logic;
    Reset : in std_logic;
    --DataValid : in std_logic;
    DataADCVout : in std_logic_vector(11 downto 0); -- Q10.2 without sign bit
    Ind1 : in std_logic;
    Ind2 : in std_logic;
    Ind3 : in std_logic;
    Ind4 : in std_logic;
    Ind5 : in std_logic;
    Ind6 : in std_logic;
    Ind7 : in std_logic;
    Ind8 : in std_logic;
    PwmOut : out std_logic);
end Regulator;

architecture Behavioral of Regulator is

    constant COUNTER_MAX : integer := 499;

    signal in1 : std_logic := '0';
    signal in2 : std_logic := '0';
    signal in3 : std_logic := '0';
    signal in4 : std_logic := '0';
    signal in5 : std_logic := '0';
    signal in6 : std_logic := '0';
    signal in7 : std_logic := '0';
    signal in8 : std_logic := '0';

    signal voutRef : std_logic_vector(12 downto 0) := (others => '0');
    signal voutRefReal : real := 0.0;

    signal counter : std_logic_vector(10 downto 0);
    signal cnt_max : std_logic;
    signal cnt_cyc1 : std_logic;
    signal cnt_cyc2 : std_logic;
    signal cnt_cyc3 : std_logic;
    signal cnt_cyc4 : std_logic;

    signal voutMeasured : std_logic_vector (11 downto 0);
    signal voutMeasuredReal : real :=0.0;

    signal bX0cmb : std_logic_vector (12 downto 0) := (others => '0'); --Q10.2
    signal bX0 : std_logic_vector (12 downto 0) := (others => '0'); --Q10.2
    signal bX1 : std_logic_vector (12 downto 0) := (others => '0'); --Q10.2

    constant B0 : std_logic_vector (4 downto 0) := "00111"; -- Q-16.20 --> 0.0000067572;
    constant B1 : std_logic_vector (4 downto 0) := "00111"; -- Q-19.23 --> 0.125 * 0.0000067572

    signal bX0B0 : std_logic_vector (17 downto 0) := (others => '0'); --Q-5.22
    signal bX1B1 : std_logic_vector (17 downto 0) := (others => '0'); --Q-2.19

    signal bX0B0Ext : std_logic_vector (25 downto 0) := (others => '0'); --Q0.25
    signal bX1B1Ext : std_logic_vector (25 downto 0) := (others => '0'); --Q0.25

    signal bY0 : std_logic_vector (25 downto 0) := (others => '0'); --Q0.25
    signal bY0comb : std_logic_vector (25 downto 0) := (others => '0'); --Q0.25
    signal bY0combReal : real := 0.0; --Q0.25
    signal bY1 : std_logic_vector (25 downto 0) := (others => '0'); --Q0.25
    signal bY0PwmRefScl : std_logic_vector (36 downto 0) := (others => '0'); --Q0.22
    signal bY0PwmRefCmb : std_logic_vector (10 downto 0) := (others => '0'); --Q0.22

    --DPWH reference. Output of the current loop.
    constant DUTYMAX : std_logic_vector(25 downto 0) := conv_std_logic_vector(integer(0.975*(2**25)), 26); -- 0.95 in Q0.25
    constant DUTYMIN : std_logic_vector(25 downto 0) := conv_std_logic_vector(integer(0.025*(2**25)), 26); -- := conv_std_logic_vector

    signal bY0PwmRef : std_logic_vector(10 downto 0);

    constant PWM_REF_INIT : std_logic_vector(10 downto 0) := conv_std_logic_vector(25, bY0PwmRef'length); --50
    constant bY0_REF_INIT : std_logic_vector(25 downto 0) := conv_std_logic_vector(integer(0.025*(2**25)), bY0'length);

begin
```

```

cnt_max <= '1' when counter=conv_std_logic_vector(COUNTER_MAX, counter'length) else '0';
cnt_cyc1 <= '1' when counter=conv_std_logic_vector(1, counter'length) else '0';
cnt_cyc2 <= '1' when counter=conv_std_logic_vector(2, counter'length) else '0';
cnt_cyc3 <= '1' when counter=conv_std_logic_vector(3, counter'length) else '0';
cnt_cyc4 <= '1' when counter=conv_std_logic_vector(4, counter'length) else '0';

process(Reset, Clk)
begin
    if Reset = '1' then
        counter <= (others => '0');
    elsif rising_edge(Clk) then
        if cnt_max = '1' then
            counter <= (others => '0');
        else
            counter <= counter + 1;
        end if;
    end if;
end process;

--Data Read
process(Clk, Reset)
begin
    if Reset = '1' then
        voutMeasured <= (others => '0');
    elsif rising_edge(Clk) then
        if cnt_cyc1 = '1' then
            voutMeasured <= DataADCvout; -- Q10.2
            in1 <= Ind1;
            in2 <= Ind2;
            in3 <= Ind3;
            in4 <= Ind4;
            in5 <= Ind5;
            in6 <= Ind6;
            in7 <= Ind7;
            in8 <= Ind8;
        end if;
    end if;
end process;

-- Registers of the voltage loop
voutRef <= "000" & in1 & in2 & in3 & in4 & in5 & in6 & in7 & in8 & "00"; --Q10.2
voutRefReal <= real(conv_integer(voutRef)) / real(2**2);

bX0cmb <= voutRef - ('0' & voutMeasured); -- Q10.2 = Q10.2 - Q10.2
voutMeasuredReal <= real(conv_integer(voutMeasured)) / real(2**2);

process(Clk, Reset)
begin
    if Reset = '1' then
        bX0 <= (others => '0');
        bX1 <= (others => '0');
    elsif rising_edge(Clk) then
        if cnt_cyc2 = '1' then
            bX0 <= bX0cmb;
            bX1 <= bX0; --Q10.2
        end if;
    end if;
end process;

bX0B0 <= bX0 * B0; -- Q-5.22 = Q10.2 * Q-16.20
bX1B1 <= bX1 * B1; -- Q-8.25 = Q10.2 * Q-19.23

---- Sign extension so there are no overflows when adding them
bX0B0Ext <= bX0B0(17) & bX0B0(17) & bX0B0(17) & bX0B0(17) & bX0B0(17) & bX0B0 & "000"; -- Q0.25
bX1B1Ext <= bX1B1(17) & bX1B1(17) & bX1B1(17) & bX1B1(17) & bX1B1(17) & bX1B1(17) & bX1B1(17) & bX1B1(17) & bX1B1;
-- Q0.25

bY0comb <= bY0 + bX0B0Ext - bX1B1Ext; -- Q0.25 = Q0.25 + Q0.25 - Q0.25, no overflow because bX0A0 and bX1A1 were previously extended

--bY0comb <= bY0 + bX0B0Ext - bX1B1Ext; -- Q0.25 = Q0.25 + Q0.25 - Q0.25, no overflow because bX0A0 and bX1A1 were previously extended
bY0combReal <= real(conv_integer(bY0comb)) / real(2**25);

process(Clk, Reset)
begin
    if Reset = '1' then
        bY0 <= bY0_REF_INIT;
    elsif rising_edge(Clk) then -- bY0 is registered for pipeline purposes
        if cnt_cyc3 = '1' then
            if bY0comb < DUTYMIN then
                bY0 <= DUTYMIN;
            elsif bY0comb > DUTYMAX then
                bY0 <= DUTYMAX;
            else
                bY0 <= bY0comb;
            end if;
        end if;
    end if;
end process;

bY0PwmRefScl <= bY0 * conv_std_logic_vector(COUNTER_MAX, counter'length);
bY0PwmRefCmb <= bY0PwmRefScl(36-1 downto 36-1-bY0PwmRefCmb'length+1);

```

```
-- DPWM signal generation, using 'counter' and 'pwmRef'
process(Reset, Clk)
begin
    if Reset = '1' then
        --bYOPwmRef <= ( OTHERS => '0' );
        bYOPwmRef <= PWM_REF_INIT;
    elsif rising_edge(Clk) then
        if cnt_cyc4 = '1' then
            bYOPwmRef <= bYOPwmRefCmb;
        end if;
    end if;
end process;

PwmOut <= '1' when counter < bYOPwmRef else '0';

end Behavioral;
```


Anexo: Código VHDL Regulador Doble Lazo

```

library IEEE;
use IEEE.std_logic_1164.ALL;
use IEEE.std_logic_signed.all;
use IEEE.std_logic_arith.all;
use IEEE.math_real.all;

entity Regulator is port (
    Clk : in std_logic;
    Reset : in std_logic;

    Plspwm :in std_logic;
    Pls : in std_logic;
    Pls1 : in std_logic;
    Pls2 : in std_logic;
    Pls3 : in std_logic;

    DataADCVg : in std_logic_vector(11 downto 0); -- Q9.3 without sign bit
    DataADCIn : in std_logic_vector(11 downto 0); -- Q3.9 without sign bit
    DataADCVout : in std_logic_vector(11 downto 0); -- Q10.2 without sign bit

    PwmOut : out std_logic;
    MeasuredOut : out std_logic_vector(12 downto 0);
    GinOut : out std_logic_vector(13 downto 0);
    ZeroCrossing : out std_logic;
end Regulator;

architecture Behavioral of Regulator is

    signal counter : std_logic_vector(10 downto 0); -- DPWM counter. From 0 to 999
    constant VoutRef : std_logic_vector(12 downto 0) := "0011001000000" ; -- 400V in Q10.2
    --constant VoutRef : std_logic_vector(12 downto 0) := "0001100100000" ; -- 200V in Q10.2
    --constant VoutRef : std_logic_vector(12 downto 0) := "0000110010000" ; -- 100V in Q10.2

    -----
    -- Transfer function of the current loop (Loop 'A'):signal counter : std_logic_vector(10 downto 0);
    --
    --          d(z)          (z-0.96875)
    -- G(z) = ----- = 0.5 -----
    --          Error(z)         (z-1)
    --
    --          d(k) = d(k-1) + 2^(-1)Error(k) - 2^(-1) 0.96875 Error(k-1)
    --          y(k) = y(k-1) + A0·x(k) - A1·x(k-1)
    --          aY0 = aY1 + aI0A0 - aI1A1
    --          Tsample: 10 us
    -----

    signal iInRef : std_logic_vector(26 downto 0); --Q5.21. iInRef = gIn * DataADCVg
    -- DataADCIn has 12 bits in the range from 0 to 8 A, therefore comes with 9 decimals
    -- iInMeasured <= DataADCIn * 4096 / 8;

    signal iInRefSat : std_logic_vector(12 downto 0); --Q3.9

    constant AA1 : std_logic_vector(6 downto 0) := "0011111" ; -- 0.5*0.96875 in Q0.6
    constant AA0 : std_logic_vector(6 downto 0) := "0100000"; -- 0.5 in Q0.6

    --constant AA1 : std_logic_vector(6 downto 0) := "0001111" ; -- 0.25*0.96875 in Q0.6
    --constant AA0 : std_logic_vector(6 downto 0) := "0010000"; -- 0.25 in Q0.6

    signal aX0,aX1 : std_logic_vector(12 downto 0); -- Q3.9

    signal aX0A0 : std_logic_vector(19 downto 0);
    signal aX1A1 : std_logic_vector(19 downto 0);

    signal aX0A0Ext : std_logic_vector(21 downto 0); -- Q6.15
    signal aX1A1Ext : std_logic_vector(21 downto 0); -- Q6.15

    signal aY0, aY1 : std_logic_vector(21 downto 0); --Q6.15

    --DPWM reference. Output of the curennt loop.
    constant DUTYMAX : std_logic_vector(21 downto 0) := "00000000111100110011010"; -- 0.95 in Q6.15
    --constant DUTYMAX : std_logic_vector(21 downto 0) := "0000000011110101110001"; -- 0.98 in Q6.15
    constant DUTYMIN : std_logic_vector(21 downto 0) := "00000000000011001100110"; -- 0.05 in Q6.15

    signal pwmRef : std_logic_vector(10 downto 0); -- Q0.10

    -----
    -- Transfer function of voltage loop (Loop 'B'):
    --
    --          (z - 0.5)
    -- G(z) = 2^(-15) -----
    --          (z - 1)

```

```

--
--          y(k) = y(k-1) + 2^(-15)·x(k) - 2^(-15)·0,5·x(k-1)
--
--          y(k) = y(k-1) + A0·x(k) - A1·x(k-1)
--
--          bY0 = bY1 + bIOA0 - bX1A1
--
--          Tsample: 0.01 s
--.....
-- DataADCYout has 12 bits in the range from 0 to 1024 V, therefore comes with 2 decimals
-- Vout <= DataADCYout * 4096 / 1024;

--constant BA0 : std_logic_vector(2 downto 0) := "010"; -- 2^-15 in Q-14.16
--constant BA1 : std_logic_vector(2 downto 0) := "001"; -- 2^-16 = 2^-15*0,5 in Q-14.16

signal bX0, bX1 : std_logic_vector(12 downto 0); --Q10.2

signal bX0A0 : std_logic_vector(13 downto 0); -- Q-5.18. bIOA0 = bX0 * 2^-15 but in the same range of bX1A1 because they are added
signal bX1A1 : std_logic_vector(13 downto 0); -- Q-5.18. bX1A1 = bX1 * 2^-16

-- Q-3.18. They are going to be used in addition of 3 terms, so before being added, they are sign extended with 2 bits for avoid
signal bX0A0Ext, bX1A1Ext : std_logic_vector(15 downto 0);

signal bY0, bY1 : std_logic_vector(15 downto 0); --Q-3.18

-- Conductance. Output of the voltage loop
constant GINNOMINAL : std_logic_vector(15 downto 0) := "0000010111001111"; -- 0.005671077504726 in Q-3.18
--constant GINMAX : std_logic_vector(15 downto 0) := "0000011011110011"; -- 0.0068053 in Q-3.18
constant GINMAX : std_logic_vector(15 downto 0) := "0000101110010010"; -- 0.0113 in Q-3.18
constant GINMIN : std_logic_vector(15 downto 0) := (others => '0'); -- 0 in Q-3.18
signal gIn : std_logic_vector(13 downto 0); -- Q-5.18.

-- Switching cycles counter in order to trigger the voltage loop after 1000 cycles.
constant MAXSWITCHINGCYCLES : integer := 999;
signal cntSwitchingCycles : integer range 0 to MAXSWITCHINGCYCLES;

--signal voutRefR : std_logic_vector(11 downto 0); -- Registered Vout reference

signal iInMeasured, voutMeasured, vgMeasured : std_logic_vector(12 downto 0); -- Data from ADC's with positive sign

--signal counterStartADCs : std_logic_vector(8 downto 0); -- Counter to delay the reading of the ADCs

signal zeroCross, zeroCrossR, ZeroCrossingS : std_logic;

signal GinSatMin, GinSatMax, PwmSatMin, PwmSatMax, DatoLeidoADCs : std_logic;

begin

-- Process to count switchingCycles (from 0 to MAXSWITCHINGCYCLES)
process(Reset, Clk)
begin
    if Reset='0' then
        cntSwitchingCycles <= 1;
    elsif rising_edge(Clk) then
        if zeroCrossR = '0' and zeroCross = '1' then -- rising edge of StartADCYout, when Vout crosses 0 V
            cntSwitchingCycles <= 0;
        elsif (cntSwitchingCycles < MAXSWITCHINGCYCLES) and (counter = 0) then -- counts every switching cycle
            cntSwitchingCycles <= cntSwitchingCycles + 1;
        end if;
    end if;
end process;

-----
-- Voltage loop operations --
-----

-- Registers of the voltage loop
process(Clk, Reset)
begin
    if Reset = '0' then
        bX1 <= (others => '0');
        bY1 <= GINNOMINAL; -- Initial value of voltage loop so simulation starts in steady state (with Vout = 400 V)
        bX0 <= (others => '0');
    elsif rising_edge(Clk) then
        if cntSwitchingCycles = 1 and counter = 0 then -- 1 switch cycle after sampling Vout, every input voltage semicycle
            bX1 <= bX0;
            bY1 <= bY0;
            bX0 <= voutRef - voutMeasured;
        end if;
    end if;
end process;

bX0A0 <= bX0 & '0'; -- instead of bX0 * BA0
bX1A1 <= bX1(12) & bX1; -- instead of bX1 * BA1

-- Sign extension so there are no overflows when adding them
bX0A0Ext <= bX0A0(13) & bX0A0(13) & bX0A0;
bX1A1Ext <= bX1A1(13) & bX1A1(13) & bX1A1;

process(Clk, Reset)
begin
    if Reset = '0' then
        bY0 <= GINNOMINAL;
        GinSatMax <= '0';
        GinSatMin <= '0';
    elsif rising_edge(Clk) then -- bY0 is registered for pipeline purposes
        bY0 <= bY1 + bX0A0Ext - bX1A1Ext; -- Q-3.18 = Q-3.18 + Q-3.18 - Q-3.18, no overflow because bX0A0 and bX1A1 were

```

```

    if bY0 > GINMAX then
        bY0 <= GINMAX;
        GinSatMax <= '1';
        GinSatMin <= '0';
    elsif bY0 < GINMIN then
        bY0 <= GINMIN;
        GinSatMax <= '0';
        GinSatMin <= '1';
    else
        GinSatMax <= '0';
        GinSatMin <= '0';
    end if;
end if;
end process;

gIn <= bY0(13 downto 0); -- Q-5.18
-- Uncomment the following line to simulate only the current loop
--gIn <= "00010111001111", "00101110011110" after 1 ms;

-----
-- Link between both loops --
-----

-- Input current reference of the current loop
iInRef <= gIn * vgMeasured; -- Q5.21 = Q-5.18 * Q9.3

iInRefSat <= iInRef(24 downto 12); -- Q3.9 in order to be in the range of Iin to be subtracted
-- No saturation needed because gIn is in the range of 0 to 0.0113 and Vg in the range of 0 to 512
-- Therefore, the max iInRef is 5.78 A, which fits in Q3.9

-----
-- Current loop operations --
-----

-- Registers of the current loop
process(Clk, Reset)
begin
    if Reset = '0' then
        aX1 <= (others => '0');
        aY1 <= DUTYMAX; --Initial value of duty cycle (because simulation starts when Vg is 0 V)
        aX0 <= (others => '0');
    elsif rising_edge(Clk) then
        if counter = 0 then -- every switching cycle
            aX1 <= aX0;
            aY1 <= aY0;
            aX0 <= iInRefSat - iInMeasured; -- Q3.9 = Q3.9 - Q3.9
        end if;
    end if;
end process;

aX0A0 <= aX0 * AA0; -- Q4.15 = Q3.9 * Q0.6
aX1A1 <= aX1 * AA1; -- Q4.15 = Q3.9 * Q0.6
-- Sign extension so there are no overflows when adding them
aX0A0Ext <= aX0A0(19) & aX0A0(19) & aX0A0; --Q6.15
aX1A1Ext <= aX1A1(19) & aX1A1(19) & aX1A1; --Q6.15

process(Clk, Reset)
begin
    if Reset = '0' then
        aY0 <= DUTYMAX;
        PWMSatMin <= '0';
        PWMSatMax <= '0';
    elsif rising_edge(Clk) then -- aY0 is registered for pipeline purposes
        aY0 <= aY1 + aX0A0Ext - aX1A1Ext; -- Q6.15 = Q6.15 + Q6.15 - Q6.15
        if aY0 > DUTYMAX then
            aY0 <= DUTYMAX;
            PWMSatMax <= '1';
            PWMSatMin <= '0';
        elsif aY0 < DUTYMIN then
            aY0 <= DUTYMIN;
            PWMSatMin <= '1';
            PWMSatMax <= '0';
        else
            PWMSatMin <= '0';
            PWMSatMax <= '0';
        end if;
    end if;
end process;

process(Clk, Reset)
begin
    if Reset = '0' then
        pwmRef <= (others => '0');
    elsif rising_edge(Clk) then
        if Plspwm = '0' then
            pwmRef <= "00111110100"; --0.5 fixed duty cycle
        else
            --pwmRef <= aY0(15 downto 5); -- Reference of the DPWM in Q0.10
            --pwmRef <= aY0(16 downto 6); -- 1/2 Reference of the DPWM in Q0.10
            pwmRef <= aY0(17 downto 7); -- 1/4 Reference of the DPWM in Q0.10
            --pwmRef <= aY0(18 downto 8); -- 1/8 Reference of the DPWM in Q0.10
        end if;
    end if;
end process;

-----
-- DPWM generation --
-----

```

```

-----
-- Counter of the DPWH
process(Reset, Clk)
begin
    if Reset = '0' then
        counter <= (0 => '1', others => '0');
    elsif rising_edge(Clk) then
        if counter = conv_std_logic_vector(999, counter'length) then -- 1 switching cycle equals to 1000 CLK ticks
            counter <= (others => '0');
        else
            counter <= counter + 1;
        end if;
    end if;
end process;

-- DPWH signal generation, using 'counter' and 'pwmRef'
process(Reset, Clk)
begin
    if Reset = '0' then
        PwmOut <= '0';
    elsif rising_edge(Clk) then
        if pwmRef <= counter then
            PwmOut <= '0';
        else
            PwmOut <= '1';
        end if;
    end if;
end process;

-----
-- Read Data --
-----
ReadData : process(Clk, Reset)
begin
    if Reset = '0' then
        iInMeasured <= (others => '0');
        vgMeasured <= (others => '0');
        voutMeasured <= (others => '0');
    elsif rising_edge(Clk) then
        iInMeasured <= ('0' & DataADCIin); -- Q3.9
        vgMeasured <= ('0' & DataADCVg); -- Q9.3
        voutMeasured <= ('0' & DataADCVout); -- Q10.2
    end if;
end process;

-----
-- Zero Crossing --
-----
process(Clk, Reset)
begin
    if Reset = '0' then
        zeroCross <= '0';
        zeroCrossR <= '0';
    elsif rising_edge(Clk) then
        zeroCrossR <= zeroCross;
        --if vgMeasured <= "000000010000" then -- 2.0 V in Q9.3
        --if vgMeasured <= "000001000000" then -- 8.0 V in Q9.3
        if vgMeasured <= "000010000000" then -- 16.0 V in Q9.3
            zeroCross <= '1';
        --elsif vgMeasured > "000001000000" then -- 8.0 V in Q9.3
        --elsif vgMeasured > "000001111000" then -- 15.0 V in Q9.3
        elsif vgMeasured > "000011110000" then -- 30.0 V in Q9.3
            zeroCross <= '0';
        end if;
    end if;
end process;

process(Clk, Reset)
begin
    if Reset = '0' then
        ZeroCrossingS <= '0';
    elsif rising_edge(Clk) then
        if zeroCrossR = '0' and zeroCross = '1' then
            ZeroCrossingS <= '1';
        else
            ZeroCrossingS <= '0';
        end if;
    end if;
end process;

-----
-- Regulator Outputs --
-----
process(Clk, Reset)
begin
    if Reset = '0' then
        MeasuredOut <= (others => '0');
    elsif rising_edge(Clk) then
        if Pls <= '0' and Pls1 <= '0' and Pls2 <= '0' and Pls3 <= '0' then --Iin
            MeasuredOut <= iInMeasured;
        elsif Pls <= '1' and Pls1 <= '0' and Pls2 <= '0' and Pls3 <= '0' then --Vout
            MeasuredOut <= VoutMeasured;
        elsif Pls <= '0' and Pls1 <= '1' and Pls2 <= '0' and Pls3 <= '0' then --Vin
            MeasuredOut <= VgMeasured;
        elsif Pls <= '0' and Pls1 <= '0' and Pls2 <= '1' and Pls3 <= '0' then --pwm
            MeasuredOut <= aY0(15 downto 3);
        end if;
    end if;
end process;

```

```
    elsif Pls <= '0' and Pls1 <= '0' and Pls2 <= '0' and Pls3 <= '1' then --pumsatmax
        MeasuredOut <= "000000000000" & PWMSatMax;
    elsif Pls <= '1' and Pls1 <= '1' and Pls2 <= '0' and Pls3 <= '0' then --pumsatmin
        MeasuredOut <= "000000000000" & PWMSatMin;
    elsif Pls <= '0' and Pls1 <= '0' and Pls2 <= '1' and Pls3 <= '1' then --ginsatmax
        MeasuredOut <= "000000000000" & GinSatMax;
    elsif Pls <= '0' and Pls1 <= '1' and Pls2 <= '1' and Pls3 <= '0' then --ginsatmin
        MeasuredOut <= "000000000000" & GinSatMin;
    end if;
end process;

GinOut <= gIn;
ZeroCrossing <= ZeroCrossingS;

end Behavioral;
```


Anexo: Código VHDL Selec

```
library WORK;
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;
use IEEE.numeric_std.all;
use IEEE.math_real.all;
use IEEE.std_logic_arith.all;

entity Selec is port(
  Clk : in std_logic;
  Reset : in std_logic;
  DataValid : in std_logic;
  ASSC_DONE : in std_logic;
  AdcResults : in std_logic_vector(11 downto 0);
  Vin : out std_logic_vector(11 downto 0);
  Vout : out std_logic_vector(11 downto 0);
  Iin : out std_logic_vector(11 downto 0);
  ChNumber : in std_logic_vector(4 downto 0));
end Selec;

architecture Behavioral of Selec is

  signal VinOut,VoutOut, IinOut : std_logic_vector(11 downto 0);

begin

  CHANNEL : process (Clk,Reset,DataValid,AdcResults,ChNumber)
  begin
    if Reset = '0' then
      VinOut <= (others => '0');
      VoutOut <= (others => '0');
      IinOut <= (others => '0');
    elsif rising_edge(Clk) then
      if DataValid = '1' and ASSC_DONE = '1' then
        if ChNumber = "10011" then --AV6 19 R9
          VinOut <= AdcResults;
        elsif ChNumber = "11100" then --AV9 28 P13
          VoutOut <= AdcResults;
        elsif ChNumber = "11001" then --AV8 25 P11
          IinOut <= AdcResults;
        else
          VinOut <= (others => '0');
          VoutOut <= (others => '0');
          IinOut <= (others => '0');
        end if;
      end if;
    end if;
  end process;

  Vin <= VinOut;
  Vout <= VoutOut;
  Iin <= IinOut;

end Behavioral;
```


Presupuesto

1. Ejecución Material

▪ Compra de ordenador personal (Software incluido)	800 €
▪ Software necesario (ModelSim y Matlab)	1.050 €
▪ Material de oficina	200 €
▪ Compra equipo electrónico	750 €
▪ Total de ejecución material	2.800 €

2. Beneficio Industrial

▪ 6 % sobre Ejecución Material	168 €
--------------------------------	-------

3. Gastos Generales

▪ 16 % sobre Ejecución Material	448 €
---------------------------------	-------

4. Honorarios Proyecto

▪ 900 horas a 16€/hora	14.400 €
------------------------	----------

5. Material Fungible

▪ Gastos de impresión	45 €
▪ Encuadernación	20 €

6. Subtotal del Presupuesto

▪ Subtotal Presupuesto	17.881 €
------------------------	----------

7. I.V.A. Aplicable

▪ 21 % Subtotal Presupuesto	3.755 €
-----------------------------	---------

8. Total Presupuesto

▪ Total Presupuesto	21.636 €
----------------------------	-----------------

Madrid, Julio 2015

El Ingeniero Jefe de Proyecto

Fdo.: Patricia Cuesta Marcos

Ingeniero de Telecomunicación

Pliego de condiciones

Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un *Control digital de convertidor conmutado mediante FPGA mixta analógica/digital*. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales.

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.
2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.
3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.
4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.
5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.
6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.
7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las

- diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.
8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.
 9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.
 10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.
 11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.
 12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.
 13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.
 14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.
 15. La garantía definitiva será del 4
 16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.
 17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.
 18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.
20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.
21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.
22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.
23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrataz anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares.

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.
2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.
3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.
4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.
5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.
7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.
8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.
9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.
10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.
11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.
12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.