

E.T.S. de Ingeniería Industrial,
Informática y de Telecomunicación

DISEÑO E IMPLEMENTACIÓN DE LOS LAZOS DE CORRIENTE DE UN INVERSOR TRIFÁSICO



Máster Universitario en Energías Renovables:
Generación Eléctrica

Trabajo Fin de Máster

ADRIÁN GONZÁLEZ GONZÁLEZ

DIRECTORES: ANDONI URTASUN

JAVIER SAMANES

Pamplona, 22 de septiembre de 2021

upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

AGRADECIMIENTOS

Tras la realización de este trabajo de fin de máster, y en particular de cada trabajo realizado, es importante echar la vista atrás y darse cuenta de quien de una forma u otra ha contribuido a su realización.

En primer lugar, agradecer mis directores Andoni Urtasun y Javier Samanes por proveerme de los medios y la oportunidad para poder desarrollar este trabajo, también agradecerles las explicaciones y la paciencia tenida.

A los compañeros de laboratorio Antonio, Iñaki, Julen, Mikel, Asier, Fermín e Ibai y esos momentos de desconexión necesarios cuando las cosas no terminan de salir. Echando la vista a lo largo de este último año no hay que olvidar también a todos los compañeros y profesores del máster y en general a la Universidad Pública de Navarra por promover el desarrollo de mi formación profesional.

A mi familia, especialmente a mis padres y mis abuelos, que siempre están dispuestos a ayudarme e intentar comprenderme.

Quisiera agradecer de manera particular a Jennifer, sin ella, sus consejos y su paciencia conmigo este máster hubiera resultado más dificultoso.

Para finalizar, y sin que suene pretencioso, me quiero dar las gracias a mí mismo por demostrarme cada día que todo con dedicación y esfuerzo sale adelante.

El cerebro no es algo que llenar, sino una chispa que encender.

Plutarco

Porque ahora es mi momento y mientras haya sandunga seguiré hasta la tumba con lo propuesto.

Canserbero

RESUMEN

En el presente trabajo de fin de máster (TFM) se describe la implementación de un lazo control de corriente en un inversor real.

Cronológicamente, el TFM ha constado de 5 fases claramente diferenciadas. En primer lugar, se ha modelado el sistema para poder analizar la estabilidad y tras esto, diseñar los lazos de control de corriente. Posteriormente, se realiza un análisis del código de partida, puesto que se parte de un proyecto previo de un sistema funcionando en lazo abierto. En cuarto lugar, se procede a la implementación y programación en el equipo de real de los lazos de control, diseñados previamente. Por último, se valida el diseño y la implementación comparando los resultados obtenidos por simulación con los obtenidos de forma experimental con el equipo real.

PALABRAS CLAVE: DSP, Inversor trifásico, filtro LCL, lazos de corriente , PSIM.

ABSTRACT

This Master's thesis describes the implementation of a current control loop in a real inverter.

Chronologically, the TFM consisted of 5 clearly differentiated phases. Firstly, the system was modelled to analyse its stability and then the current control loops were designed. Subsequently, an analysis of the initial code was carried out, since it was based on a previous project of a system operating in an open loop. Fourthly, the previously designed control loops are implemented and programmed in the real equipment. Finally, the design and implementation are validated by comparing the results obtained by simulation with those obtained experimentally with the real equipment.

KEY WORDS: DSP, three-phase inverter, LCL filter, current loops, PSIM.

ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN Y OBJETIVOS	7
1.1 INTRODUCCION	7
1.2 OBJETIVOS	9
2. DESCRIPCIÓN DEL SISTEMA	11
2.1 CONVERTIDOR PCO-10T800	11
2.2 PLACA DE CONTROL.....	13
2.3 FILTRO LCL	15
2.4 CODE COMPOSER STUDIO	16
2.5 FUENTE DC.....	17
2.6 OSCILOSCOPIO.....	17
2.7 ORDENADOR	17
2.8 DIAGRAMA DE CONEXIÓN DE LOS ELEMENTOS	18
3. ESTADO INICIAL DEL PROYECTO	19
4. MODELADO Y CONTROL DEL SISTEMA	21
4.1 MODELADO DEL SISTEMA.....	21
4.1.1 Transformada de Clarke.....	22
4.1.2 Transformada de Park.....	23
4.1.3 Modelado en dq.....	24
4.2 CONTROL DEL CONVERTIDOR.....	27
4.2.1 Descripción del lazo de control.....	28
4.2.2 Filtro Digital	29
4.2.3 Modelo de la digitalización y PWM.....	30
4.2.4 Cálculo de K_p y T_n del lazo de corriente con carga resistiva	31
4.2.5 Comprobación del ajuste del controlador PI	32
4.3 CONEXIÓN A RED	34
4.3.1 PLL.....	34
4.3.2 Comprobación del ajuste del controlador PI de la PLL.....	36
4.3.3 Problemática del filtro LCL en la conexión a red	36
4.3.4 Cálculo de la K_p y T_n del lazo de corriente para la conexión a red	37
4.3.5 Análisis del sistema con distintas resistencias de amortiguamiento (R_d).....	39
4.4 RESUMEN DE PARAMETROS.....	42
5. IMPLEMENTACIÓN DEL LAZO DE CONTROL EN EL PROTOTIPO	43
5.1 DESCRIPCIÓN DE LIBRERÍAS Y FUNCIONES NECESARIAS PARA EL CONTROL DEL PROTOTIPO	43
5.2 PROGRAMACIÓN DEL CONTROL DE CORRIENTE	48
5.2.1 Diagrama de flujo	48
5.2.2 Descripción del código de control	49
5.3 ESQUEMA GENERAL DEL CÓDIGO	57
5.3.1 Mapa de estados.....	58
6. VALIDACIÓN EXPERIMENTAL.....	59
6.1 LIMITACIONES DEL PROTOTIPO.....	59
6.2 RESULTADOS DE LA SIMULACIÓN	59
6.3 RESULTADOS EXPERIMENTALES	61
7. CONCLUSIONES Y LINEAS FUTURAS	64

7.1	CONCLUSIONES	64
7.2	LINEAS FUTURAS.....	64
8.	BIBLIOGRAFIA	65

ÍNDICE DE FIGURAS

Figura 1.	Consumo final total de energía por fuente, 1973 [2].....	7
Figura 2.	Consumo final total de energía por fuente, 2018 [2].....	8
Figura 3.	Reducción de costos en la energía renovable desde 2010 [3].....	8
Figura 4.	Convertidor como nexo obligatorio entre la fuente renovable y la red [4].....	9
Figura 5.	Convertidor PCO-10T800	11
Figura 6.	Esquema de la conexión del convertidor.....	11
Figura 7.	Diagrama de bloques del convertidor PCO-10T800.....	12
Figura 8.	Placa de control CDA01-CU3.....	13
Figura 9.	Diagrama de bloques de la placa de control.....	14
Figura 10.	Diagrama de bloques sobre placa de control real	14
Figura 11.	Inversor y placa de control ensamblados.....	15
Figura 12.	Componentes del filtro LCL.....	15
Figura 13.	Interfaz gráfica Code Composer.....	16
Figura 14.	Fuente DC 300V-20A Xantrex XDC.....	17
Figura 15.	Osciloscopio Tektronik TDS 2004C.....	17
Figura 16.	Montaje completo de los elementos en el laboratorio	18
Figura 17.	Esquema de un sistema formado por un inversor, un filtro LCL y una carga resistiva	21
Figura 18.	Esquema de un sistema formado por un inversor, un filtro LCL y la red	21
Figura 19.	Sistema de referencia trifásico y $\alpha\beta$	22
Figura 20.	sistema de referencia trifásico y dq.....	23
Figura 21.	Modelos equivalentes monofásicos: izquierda conexión a red, derecha conexión a carga resistiva.....	25
Figura 22.	Equivalente monofásico en impedancias.....	25
Figura 23.	Acoplamiento de los ejes en el modelo dq	27
Figura 24.	Sistema de control del inversor conectado a una carga resistiva	27
Figura 25.	Sistema de control del inversor conectado a red	28
Figura 26.	Lazo de corriente Id	28
Figura 27.	Lazo de corriente Iq	29
Figura 28.	Diagrama de Bode del filtro analógico y del filtro digital.....	30
Figura 29.	Modelización de los retrasos debidos a la digitalización	30
Figura 30.	Diagrama de Bode de la función de transferencia en lazo abierto.....	32
Figura 31.	Diagrama de Bode de la función de transferencia en lazo cerrado	33
Figura 32.	Respuesta ante un escalón del sistema	33
Figura 33.	Respuesta del sistema ante un escalón PSIM	34
Figura 34.	Diagrama de bloques de un PLL.....	35
Figura 35.	Diagrama de Bode de la función de transferencia de la PLL en lazo abierto.....	36
Figura 36.	Diagrama de Bode de la planta del sistema con diferentes valores de la resistencia de amortiguamiento	39
Figura 37.	Diagrama de Bode de la función de transferencia del sistema en lazo abierto con distintos amortiguamientos	40
Figura 38.	Diagrama de Bode de la función de transferencia del sistema en lazo cerrado con distintos amortiguamientos	40
Figura 39.	Respuesta ante un escalón del sistema con distintos amortiguamientos	41
Figura 40.	Respuesta del sistema ante un escalón de 0 a 1 kW en la potencia activa PSIM	41
Figura 41.	Archivos del código ".c" y librerías ".h".....	43
Figura 42.	Fragmento del código del archivo main.c estado init	44
Figura 43.	Fragmento del código del archivo CONMUTE_3n1.h	45

Figura 44. Fragmento del código del archivo extra_functions.h	45
Figura 45. Fragmento del código del archivo CONMUTE_3n1_structures.h	45
Figura 46. Fragmento del código del archivo extra_functions.h función alarmas.....	46
Figura 47. Fragmento del código del archivo CONMUTE_3n1_defines.h.....	46
Figura 48. Fragmento del código del archivo F28M35x_EPwm.c	47
Figura 49. Comparativa corriente por la resistencia con distinta frecuencia de conmutación: izquierda 5 kHz y derecha 10 kHz.....	47
Figura 50. Diagrama de Flujo de la Interrupción principal.....	48
Figura 51. Fragmento del código del archivo Interrup_ADC.c, declaración de variables	49
Figura 52. Fragmento del código del archivo Interrup_ADC.c, asignación de variables.....	50
Figura 53. Fragmento del código del archivo Interrup_ADC.c, lectura de medidas	50
Figura 54. Fragmento del código del archivo Interrup_ADC.c, factores de escala	51
Figura 55. Fragmento del código del archivo Interrup_ADC.c, rutina de alarmas.....	52
Figura 56. Fragmento del código del archivo Interrup_ADC.c, filtro digital	53
Figura 57. Fragmento del código del archivo Interrup_ADC.c, ángulo para transformada dq.....	54
Figura 58. Fragmento del código del archivo Interrup_ADC.c, ángulo para transformada dq PLL.....	54
Figura 59. Fragmento del código del archivo Interrup_ADC.c, tensiones y corrientes dq	55
Figura 60. Fragmento del código del archivo Interrup_ADC.c, modulantes dq.....	56
Figura 61. Fragmento del código del archivo Interrup_ADC.c, modulantes ma, mb y mc	56
Figura 62. Fragmento del código del archivo Interrup_ADC.c, módulo PWM.....	56
Figura 63. Fragmento del código del archivo Interrup_ADC.c, reset de la interrupción	57
Figura 64. Diagrama de flujo de la estructura principal del código [5].....	57
Figura 65. Esquema completo de simulación PSIM	59
Figura 66. Corriente por la resistencia con una corriente de referencia de Iq=1.5 A PSIM.....	60
Figura 67. Tensión por la resistencia con una corriente de referencia de Iq=1.5 A PSIM	60
Figura 68. Tensión y corriente por la resistencia con una corriente de referencia de Iq=1.5 A PSIM	61
Figura 69. Inversor + Filtro LCL + Carga resistiva montado en el laboratorio	62
Figura 70. Corriente por la resistencia de la fase R con una corriente de referencia de Iq=1.5 A Prototipo	62
Figura 71. Tensión por la resistencia de la fase R con una corriente de referencia de Iq=1.5 A Prototipo.....	63
Figura 72. Tensión y corriente por la resistencia con una corriente de referencia de Iq=1.5 A Prototipo.....	63

ÍNDICE DE TABLAS

Tabla 1. Parámetros filtro LCL.....	16
Tabla 2. Checklist primera toma de contacto sistema placa+inversor [5]	19
Tabla 3. Parámetros generales del sistema	42
Tabla 4. Parámetros del control con carga resistiva.....	42
Tabla 5. Parámetros del control de la PLL.....	42
Tabla 6. Parámetros del control de corriente con conexión a red	42
Tabla 7. Factores de escala	51
Tabla 8. Parámetros filtro digital	52

1. INTRODUCCIÓN Y OBJETIVOS

1.1 INTRODUCCION

El modelo energético actual está basado principalmente en las fuentes de energías no renovable, denominados combustibles fósiles. Para la obtención de energía a partir de los combustibles fósiles es necesario la emisión de CO2 y otros gases de efecto invernadero a la atmosfera, siendo estos gases nocivos para el medio ambiente y por ese motivo perjudicial para el ser humano. Descarbonizar el sistema energético no es otra cosa que dejar de emitir CO2 y el resto de los gases de efecto invernadero, lo que implica el tener que abandonar los combustibles fósiles para la obtención de energía [1].

Esta preocupación por el medio ambiente y el querer lograr una independencia energética, que de lo contrario produce grandes inestabilidades tanto políticas como bélicas, fuerzan a la sociedad a buscar un modelo energético más sostenible que el actual. Es aquí donde se entiende un aumento de las energías renovables a lo largo de los últimos años, algo que puede observarse en las Figuras 1 y 2.

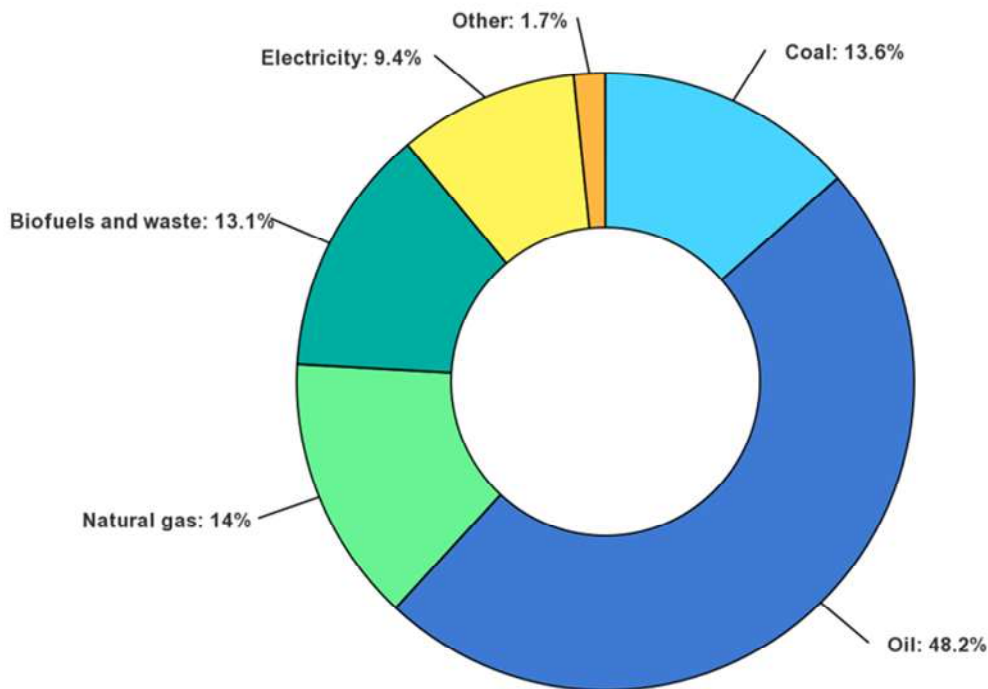


Figura 1. Consumo final total de energía por fuente, 1973 [2]

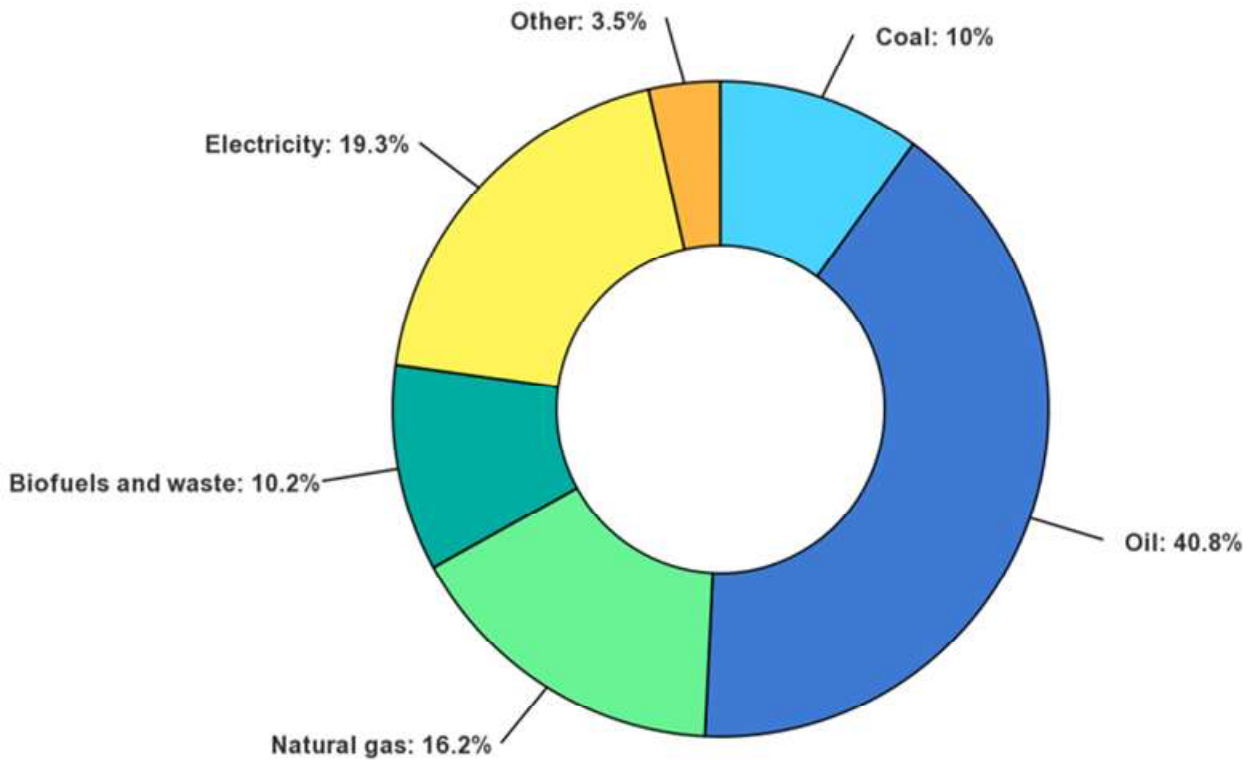


Figura 2. Consumo final total de energía por fuente, 2018 [2]

*Mtoe: Millones de toneladas equivalentes de petróleo

La principal apuesta por un modelo energético sostenible y diferente al actual, basado en energías renovables, hace que se produzcan inversiones en este aspecto, lo que a su vez produce una reducción del precio en dichas tecnologías, favoreciendo así su implantación. Las tecnologías renovables por las que se hace una apuesta más fuerte son la eólica y la fotovoltaica, en la Figura 3 se puede observar la reducción de los costos en dichas tecnologías desde el año 2010.

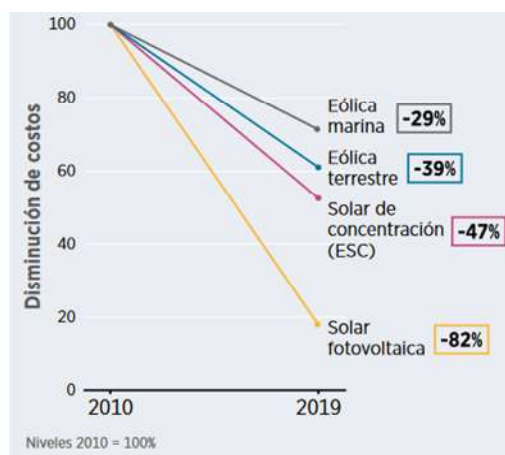


Figura 3. Reducción de costos en la energía renovable desde 2010 [3]

El aumento de las EERR no solo supone un mayor porcentaje de utilización de los recursos renovables frente a los no renovables, sino que también plantea nuevos retos relacionados con el sistema eléctrico. Estas tecnologías tienen que ser capaces de superar ciertos retos:

- Garantizar la estabilidad de la red
- Mejorar la calidad de la energía

Por otro lado, la potencia eléctrica obtenida en los sistemas fotovoltaicos es de naturaleza continua, mientras que la inyección de potencia a la red eléctrica se hace principalmente a través de corriente alterna a la frecuencia de la red, por lo tanto, es necesario transformarla. Algo parecido ocurre con la energía eléctrica obtenida a través de sistemas eólicos en este caso es de naturaleza alterna, pero a distinta frecuencia que la red. El dispositivo encargado de adaptar estas magnitudes es el convertidor electrónico de potencia.

Este dispositivo y su control se convierten en un elemento esencial a la hora de cumplir con los retos planteados anteriormente e inyectar la potencia renovable a red como puede verse en la Figura 4.

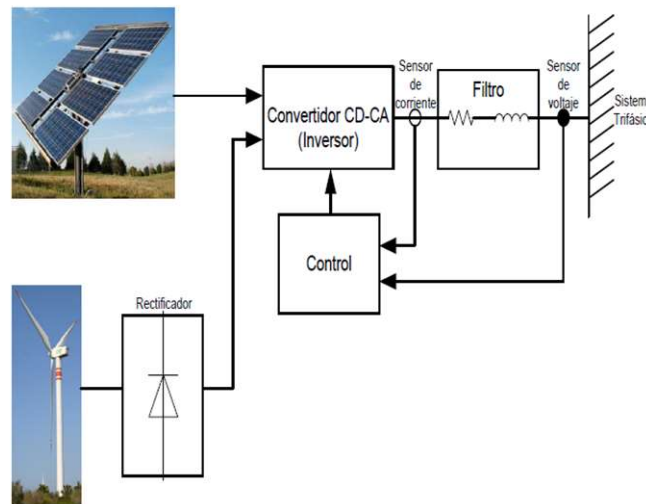


Figura 4. Convertidor como nexo obligatorio entre la fuente renovable y la red [4]

1.2 OBJETIVOS

El objetivo principal del proyecto consiste en implementar, en un equipo real del laboratorio de electrónica de potencia de la UPNa, el control de corriente de un inversor trifásico conectado a un filtro LCL para que pueda ser utilizado en posteriores trabajos e investigaciones.

Para el alcance de este objetivo se han llevado a cabo una serie de acciones:

- Análisis del sistema y las partes que lo forman
- Análisis del software utilizado para la programación del control digital, en este caso la herramienta Code Composer ofrecida por Texas Instruments
- Modelado del sistema en Matlab y PSIM
- Diseño e implementación del control digital
- Simulación con PSIM
- Implementación del control digital en el DSP y puesta en marcha del sistema real
- Comparativa y validación de los resultados

Es importante destacar que las herramientas utilizadas para las simulaciones son los programas de Matlab y PSIM y que se ha invertido gran parte del tiempo tanto en el aprendizaje del software Code Composer, como en el diseño e implementación del control.

2. DESCRIPCIÓN DEL SISTEMA

2.1 CONVERTIDOR PCO-10T800

Se trata de un convertidor trifásico diseñado para baja tensión con una potencia de 10 kVA y que dispone de todos los elementos de hardware necesarios para implementar un sistema conectado a red.

Este convertidor es el tecknoCEA PCO-10T800 que debido a su versatilidad permite al usuario implementar diferentes tipologías de convertidor digital en un entorno digital. Algunas de estas tipologías son: convertidor AC-DC Reversible (Inversor FV, Filtro activo, Aplicaciones V2G, microrredes...), convertidor DC-DC (Medio Puente, Puente Completo...), controlador de un motor AC. En la Figura 5 se puede observar una imagen real del convertidor.

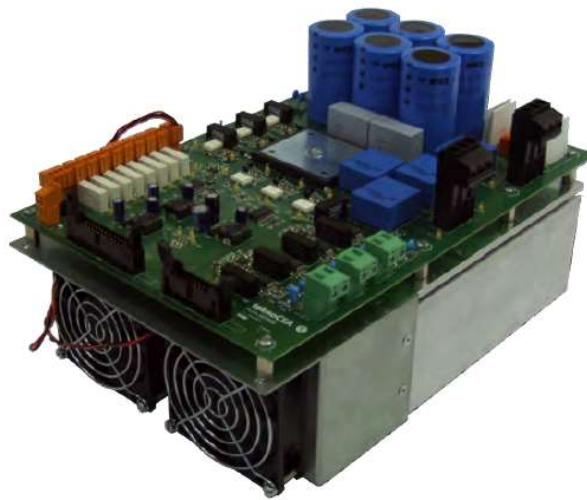


Figura 5. Convertidor PCO-10T800

Se pretende conectar en el lado DC del convertidor una fuente de tensión DC y en el lado AC un filtro LCL como puede verse en la Figura 6, a la salida de este filtro se conectará una carga resistiva.

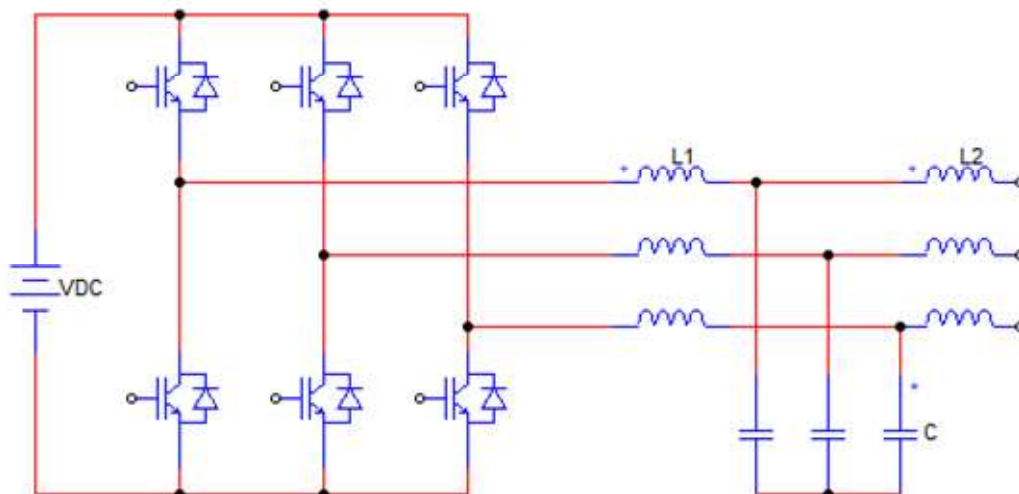


Figura 6. Esquema de la conexión del convertidor

Algunas de las características principales del convertidor son las siguientes y un diagrama de bloques más completo de este puede verse en la Figura 7.

- Voltaje de salida nominal: 400 Vrms
- Corriente de salida nominal 15 Arms
- Tensión de bus máxima 800 V DC
- Frecuencia de conmutación máxima 20 KHz
- 6 drivers aislados ópticamente que incluyen protección contra sobretensión, sobrecorriente y desconexión suave en caso de disparo por sobrecorriente. La protección contra conducción cruzada y la inserción de tiempo muerto han de ser implementadas en el sistema de control de parte del usuario
- 4 sensores de tensión. 1 para el BUS DC y los otros adicionales configurables por el usuario, se utilizarán para medir las tensiones de la red.
- 4 sensores de corriente. 1 para la corriente de BUS y los para la corriente de las 3 fases.
- Módulo de temperatura, es importante destacar que el convertidor no incluye ninguna protección contra la temperatura de los semiconductores, dicha protección deberá ser implementada por el usuario en el sistema de control
- BUS DC con una capacidad total 1,02 mF formado por 6 condensadores.
- 8 relés con un voltaje nominal de 230 Vrms, una corriente máxima de 5 A y una corriente de conmutación máxima de 3 A.
- Sistema de refrigeración forzada formado por 2 ventiladores de 24 V y un consumo de 150 mA cada uno.
- Fuente de alimentación externa necesaria de 24 V y 3 A

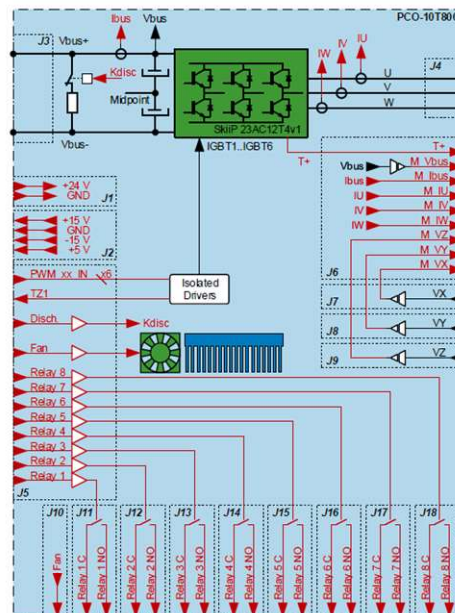


Figura 7. Diagrama de bloques del convertidor PCO-10T800

2.2 PLACA DE CONTROL

La CDA01-CU3 es una tarjeta de control independiente basada en el microcontrolador Concerto F28M35H52C de Texas Instruments. La tarjeta de control incluye todos los periféricos y circuitos necesarios para un proyecto completo de electrónica de potencia que incluya capacidades de comunicación.

La Figura 8 muestra una imagen de la placa.

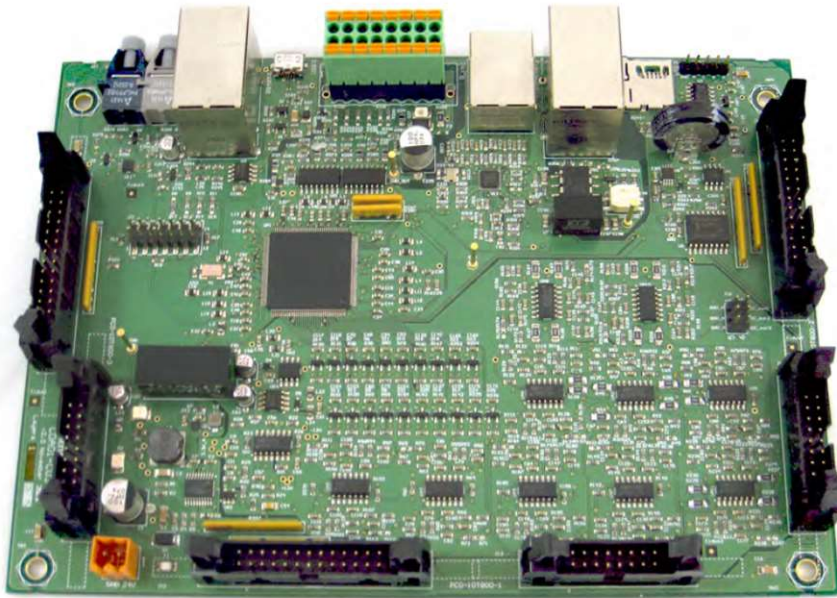


Figura 8. Placa de control CDA01-CU3

Algunas de las características destacables de la placa de control son las siguientes:

- F28M35H52C Concerto:
 - Subsistema principal - ARM Cortex – M3
 - Subsistema de control TMS320C28x 32 bit CPU
 - Reloj de entrada de 20 MHz
- Convertidor analógico-digital (ADC) de 12 bits con 20 canales de entrada
- 18 salidas digitales
- 14 señales PWM
- Conectores de fibra óptica del transmisor y del receptor para la sincronización PWM
- Interfaz aislada CAN 2.0 con controlador de línea y conector RJ45 duplicado para conexión en cadena
- RS-485 integrado con controlador de línea y conector RJ45 duplicado para conexión en cadena
- Conector Ethernet
- Conector de tarjeta microSD
- 6 entradas configurables aisladas o no aisladas en el conector de expansión
- Múltiples salidas digitales en conectores de expansión
- Convertidor digital-analógico (DAC) con 4 canales de salida
- Memoria EEPROM externa de 256 kB integrada con interfaz I2C
- Reloj en tiempo real (RTC) integrado con interfaz I2C
- Conector de emulación JTAG IEEE 1149.1 integrado
- Entrada única de alimentación de 24 V DC

En la Figura 9 puede observarse el diagrama de bloques básico de la placa de control CDA01-CU3 con el microcontrolador y todos los periféricos conectados a él.

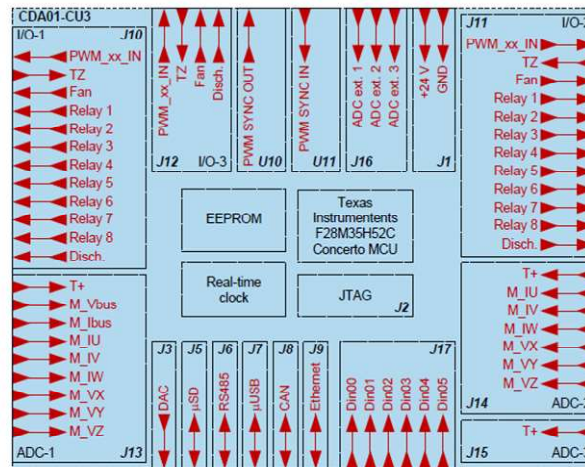


Figura 9. Diagrama de bloques de la placa de control

Por otro lado, en la Figura 10 se puede observar la distribución real de los periféricos de la placa de control CDA01-CU3.

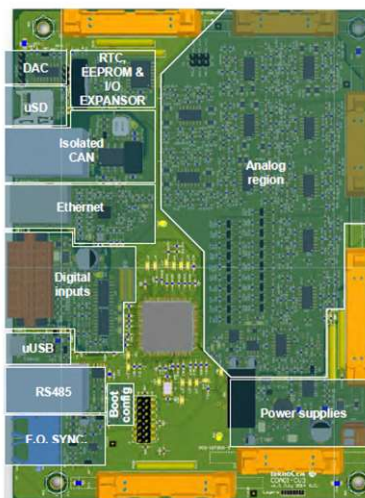


Figura 10. Diagrama de bloques sobre placa de control real

El CDA01-CU3 ha sido diseñado específicamente como tarjeta de control del convertidor PCO-10T800. Con ella es posible controlar hasta 3 convertidores PCO-10T800. En la Figura 11 se puede observar el acoplamiento del convertidor con la tarjeta de control.

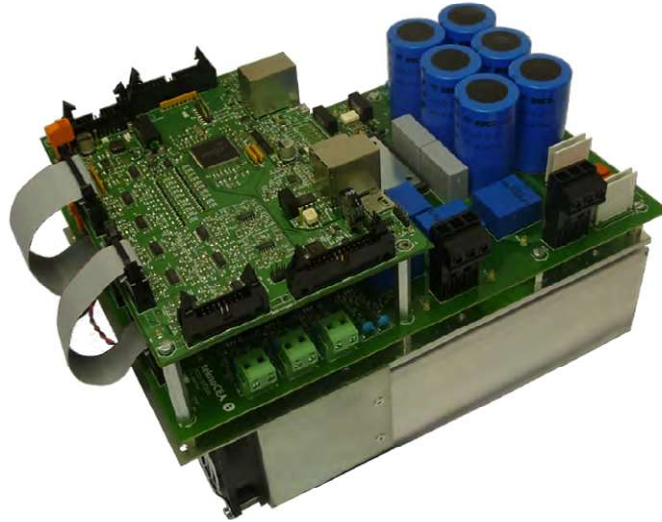


Figura 11. Inversor y placa de control ensamblados

Para más información se puede consultar la hoja de características tanto de la placa de control como del convertidor.

2.3 FILTRO LCL

Los componentes del filtro LCL utilizado se pueden ver en la Figura 12.

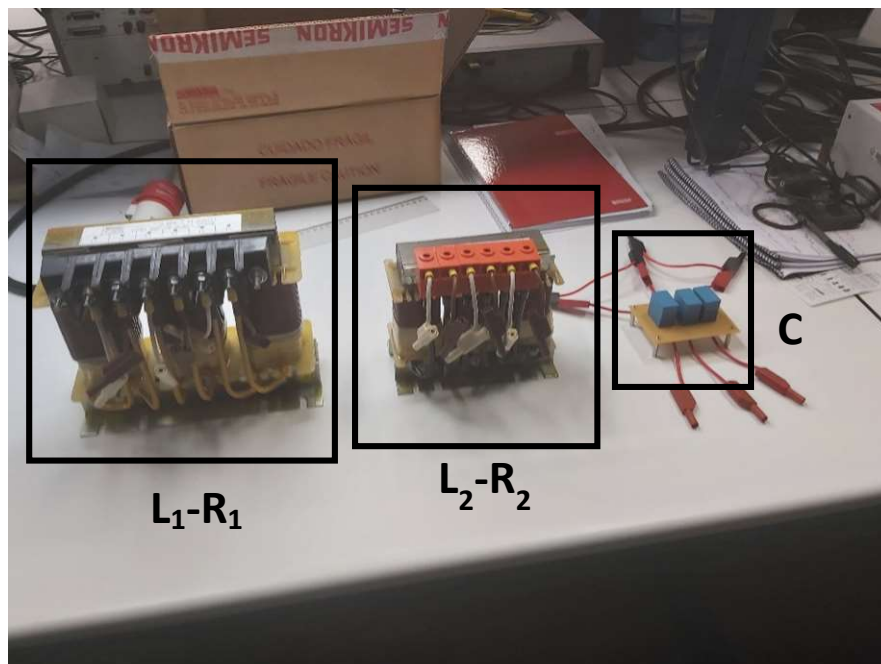


Figura 12. Componentes del filtro LCL

El valor de los parámetros del filtro se puede observar en la Tabla 1.

PARÁMETROS GENERALES	
L ₁	2.2 mH
L ₂	1.098 mH
R ₁	0.1 Ω
R ₂	0.1 Ω
C	4.7 uF

Tabla 1. Parámetros filtro LCL

2.4 CODE COMPOSER STUDIO

El software Code Composer Studio es la interfaz que permite la comunicación del usuario con la placa de control y el convertidor. Entre las funciones de las que dispone se encuentran un compilador de C/C++, un editor de código fuente, un entorno de creación de proyectos y otras funciones. En la Figura 13 se puede ver la interfaz gráfica del software.

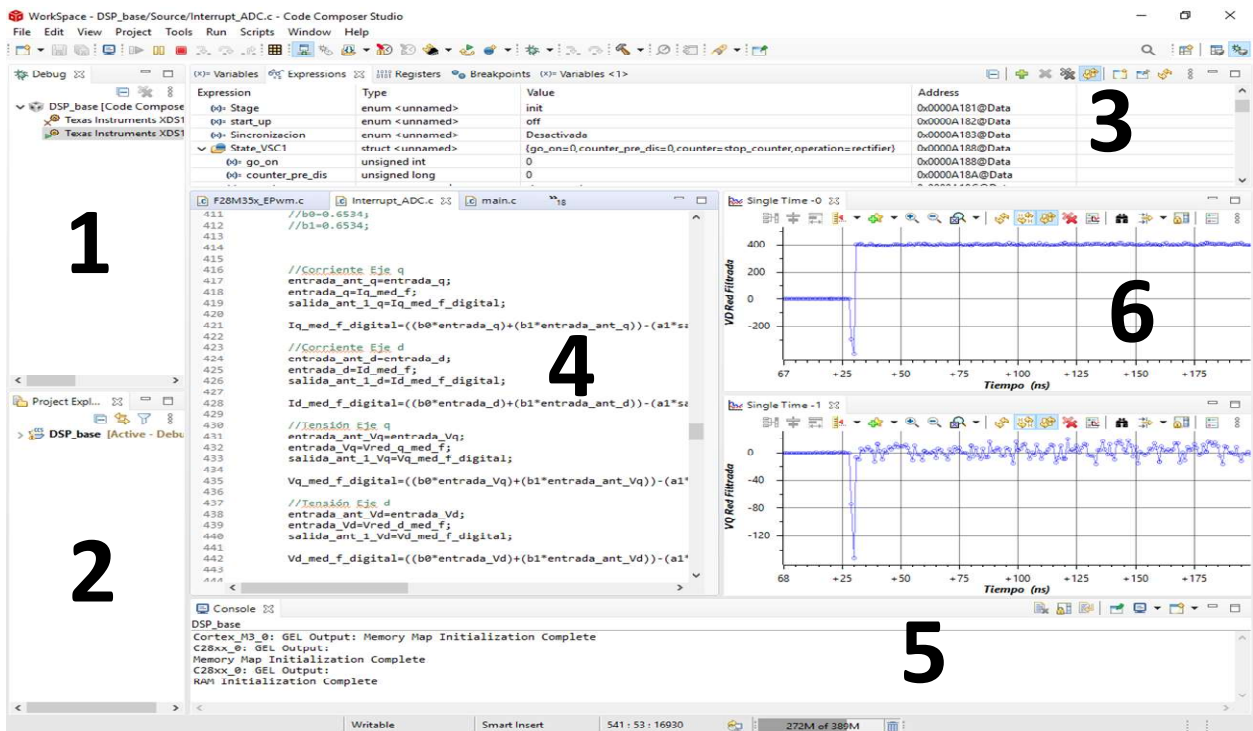


Figura 13. Interfaz gráfica Code Composer

1. Ventana de depuración
2. Ventana del proyecto
3. Ventana de las variables
4. Ventana del código del programa
5. Ventana de errores y advertencias de compilación
6. Ventana de visualización gráfica

La conexión del ordenador con el DSP se realiza a través de un conector JTAG.

2.5 FUENTE DC

La fuente de tensión a la entrada del inversor será la Xantrex XDC 300-20 que se puede ver en la Figura 14.



Figura 14. Fuente DC 300V-20A Xantrex XDC

Esta fuente permite una tensión de hasta 300 V y una corriente de 20 A.

2.6 OSCILOSCOPIO

El osciloscopio utilizado es el Tektronik TDS 2004C, se puede ver en la Figura 15.



Figura 15. Osciloscopio Tektronik TDS 2004C

2.7 ORDENADOR

Es indispensable el uso de un ordenador para poder trabajar con la interfaz de Texas Instruments Code Composer Studio.

2.8 DIAGRAMA DE CONEXIÓN DE LOS ELEMENTOS

En la Figura 16 se puede ver el montaje completo de los elementos comentados anteriormente.

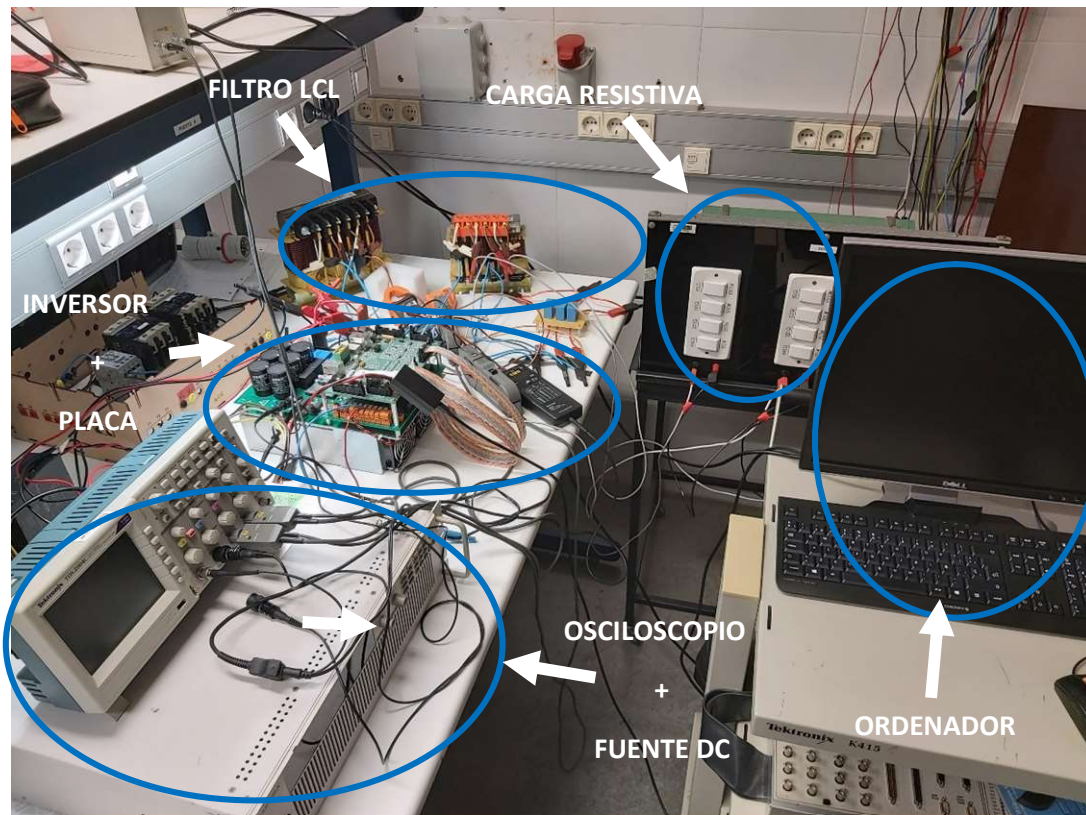


Figura 16. Montaje completo de los elementos en el laboratorio

3. ESTADO INICIAL DEL PROYECTO

El presente proyecto tiene como punto de partida un proyecto previo [5], que a su vez parte del código inicial suministrado por la empresa. Este código inicial carecía de aspectos como los estados del sistema y un sistema de detección de alarmas, entre otros.

El proyecto previo se encargó de solventar algunos de estos problemas, y proporcionó una tabla con los pasos a seguir para la primera toma de contacto con el sistema, estos pasos se encuentran en la Tabla 2. El primero de los pasos a seguir es la verificación del hardware del sistema, comprobando la correcta alimentación de la placa y del convertidor y su interconexión. Tras esto se procede a un volcado del código y se comprueba la ejecución de las interrupciones del sistema. A continuación, con tensión en el lado DC y en vacío en el lado AC, se comprueba la correcta conmutación de los IGBT's. Cuando todo esto ha sido verificado se añade un filtro L en el lado AC y se comprueban los límites de corriente.

PUESTA EN MARCHA DEL PCO-10T800 + F28M35 CDA01-CU3		
VERIFICACIÓN	PASOS	ACCIÓN ESPERADA
VERIFICACIÓN DEL HARDWARE DEL SISTEMA		
Alimentación placa de control	1. Verificar cableado de alimentación	Los pines de salida de la fuente de alimentación dan continuidad con su correspondiente polaridad
	2. Alimentar la placa de control sin conectar al convertidor	Los led verdes de la placa de control se iluminan, verificando su correcta alimentación
Alimentación del inversor desde fuente externa	1. Alimentar inversor a 24 V en conector J1	Los led verdes del inversor se iluminan, verificando su correcta alimentación
Interconexión placa de control - convertidor	1. Comprobar correcto pineado de las pistas de conexión entre placas	Todos los extremos de los cables de las pistas dan continuidad con su extremo correspondiente
	2. Conectar la placa de control con el inversor y alimentar únicamente la placa de control	Los leds del inversor se iluminan, indicando su correcta alimentación desde la placa de control por el pin correspondiente
VERIFICACIÓN DEL CÓDIGO BASE DE PROGRAMACIÓN		
Ejecución del código base	1. Sin volcar a la placa, ejecutar el debug	No existen errores de compilación
	2. Volcar código base en la placa sin tensión en bus DC	El código se ejecuta, queda en estado "init"
	3. Comprobar ejecución de la interrupción sin tensión en bus DC	El contador de los registros PWM se actualizan en la ventana Expressions y se activa el flag de la interrupción
VERIFICACIÓN DEL CONTROL EN VACÍO		
Encendido de ventiladores	1. Crear estado de encendido de ventiladores 2. Hacer que el código entre en el estado	Los ventiladores deberán comenzar a funcionar
Límites de tensión del bus DC	1. Modificar el valor de los límites a tensiones seguras, crear el estado de validación y ejecutar el código	-
	2. Alimentar el bus DC a una tensión dentro de los límites	El valor de VBUS_VSC1 deberá cambiar en la ventana "Expressions"
	3. Entrar en el estado de validación y modificar la tensión del bus fuera de los límites	El inversor entra en el estado de alarma al superarse los límites
Comutación de los transistores	1. Ejecutar el código con tensión de bus 0 V 2. Alimentar el bus DC a una tensión dentro de los límites 3. Entrar en el estado run	A la salida del inversor se observan las tensiones esperadas de la conmutación
VERIFICACIÓN DEL LÍMITE DE CORRIENTE CON FILTRO L A LA SALIDA		
Límite de corriente a la salida	1. Diseño de onda senoidal en la referencia	Las conmutaciones de la simulación y del inversor en abierto se corresponden
	2. Simulación del modelo en SIMULINK	
	3. Conexión de filtro L a la salida	El inversor entra en el estado de alarma al sobrepasar la corriente límite
	4. Ejecución del código con índice de modulación obtenido de la simulación	
	5. Aumentar la tensión del bus DC para aumentar amplitud de la corriente	

Tabla 2. Checklist primera toma de contacto sistema placa+inversor [5]

Con dicho proyecto se logró obtener un sistema que funcionara en lazo abierto, en el presente proyecto se pretende ir un paso más allá y conseguir un sistema que funcione en lazo cerrado.

4. MODELADO Y CONTROL DEL SISTEMA

En este apartado se explicarán los modelos utilizados, así como el diseño del lazo de control de corriente. Hay que destacar que los valores de los parámetros utilizados serán los obtenidos de los elementos reales presentes en el laboratorio.

4.1 MODELADO DEL SISTEMA

Es necesario modelar el sistema para poder estudiarlo y diseñar sus lazos de control. En primer lugar, se modela en la herramienta Matlab mediante la función de transferencia (FT) para posteriormente adaptar la respuesta del convertidor con el lazo diseñado gracias al modelo de FT en PSIM. El modelo realizado con Matlab es un modelo más sencillo que servirá para el posterior análisis del modelo en PSIM, siendo este un modelo más completo y que tiene en cuenta las posibles no linealidades del sistema.

En este proyecto se tiene un convertidor que será conectado de dos formas distintas, de forma aislada o a la red. Cuando la conexión es en aislada a la salida del filtro LCL se tiene una carga resistiva, como puede verse en la Figura 17. Por otro lado, la otra forma de conexión es a la red a través del filtro LCL, como puede verse en la Figura 18.

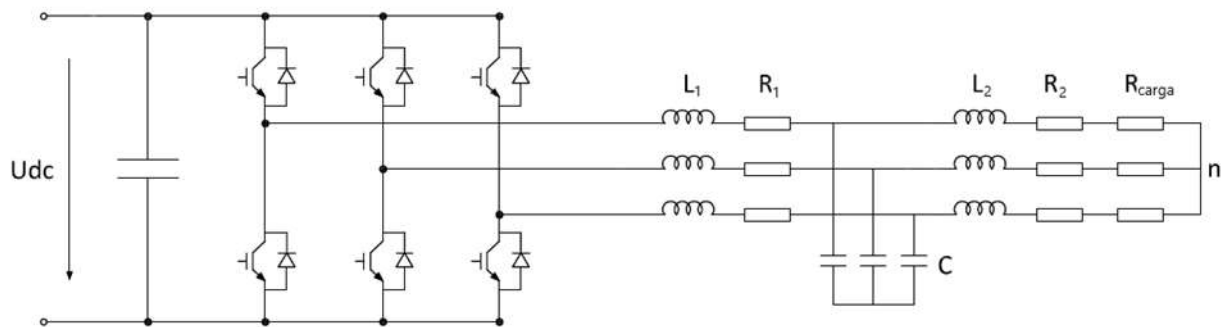


Figura 17. Esquema de un sistema formado por un inversor, un filtro LCL y una carga resistiva

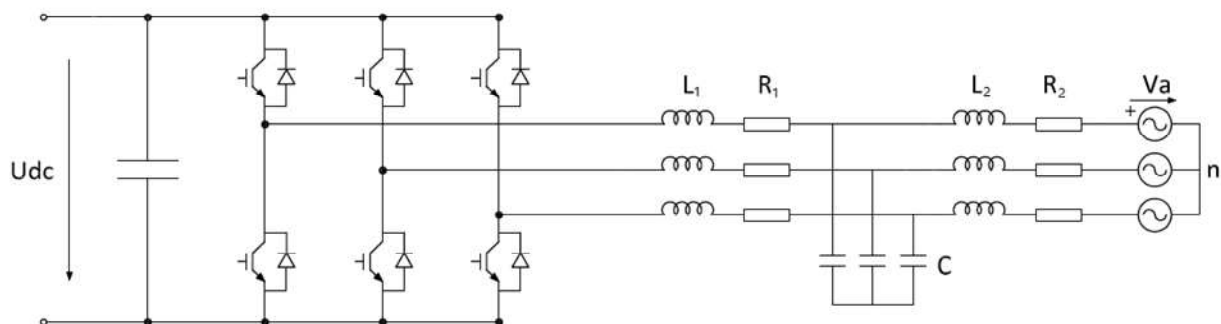


Figura 18. Esquema de un sistema formado por un inversor, un filtro LCL y la red

En un sistema trifásico se tiene tres componentes, en el que la suma de estas es nula. Esto hace que estas tres componentes no se puedan controlar de forma independiente, sino que se controlan dos componentes y la tercera se calcula. Para evitar esto se puede recurrir al uso de las transformadas (Clarke o Park), en las que se tienen dos fases ($\alpha\beta$ o dq) que pueden ser controladas de manera independiente y una tercera, que es la secuencia homopolar y permite ver si existen corrientes por tierra, en este proyecto al no tener corrientes por tierra se evitará el uso de esta tercera componente.

A continuación, se explicará la manera de obtener estas transformadas y cual se utilizará para el desarrollo del proyecto.

4.1.1 Transformada de Clarke

Como se comentaba anteriormente esta transformada permite representar los sistemas trifásicos formados por tres componentes sinusoidales que se encuentran dentro de un plano, con ejes a, b y c no ortogonales en otro sistema que sí que lo sea. Se consigue transformar el vector formado por tres componentes abc, en otro vector formado por dos componentes, α y β , que tienen un valor distinto de cero y que sí pertenecen a un sistema ortogonal [6] y otra componente que corresponde a la homopolar.

Para realizar la transformación de Clarke de manera matemática basta con multiplicar el vector abc por la matriz de la ecuación (1), denominada matriz de Clarke:

$$\begin{bmatrix} x_\alpha \\ x_\beta \\ x_0 \end{bmatrix} = \sqrt{\frac{2}{3}} \cdot \begin{bmatrix} 1 & -\frac{1}{\sqrt{2}} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \cdot \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} \quad (1)$$

En la Figura 19 se muestra la relación de los ejes α y β con un sistema trifásico.

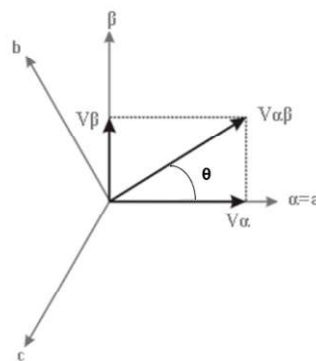


Figura 19. Sistema de referencia trifásico y $\alpha\beta$

Los nuevos ejes son estáticos, por lo tanto, el sistema depende aún del ángulo (θ). Para realizar un control en estos ejes se requiere un controlador proporcional-integral-resonante (PIR) capaz de conseguir un seguimiento a una determinada frecuencia (por ejemplo 50 Hz) con error nulo, sin embargo, el ajuste de este tipo de controladores no es tan sencillo.

Para evitar el uso de este tipo de controladores y emplear un controlador PI más sencillo se puede recurrir a la transformada de Park, en la que se emplean ejes giratorios. Estos ejes giran a la frecuencia del fasor y con

el mismo ángulo, de esta manera se obtiene una variación de forma constante y con utilizar un controlador PI se consigue tener un error de seguimiento nulo.

A continuación, se explicará la manera de obtener esta transformada, la cual se utilizará para el desarrollo del proyecto.

4.1.2 Transformada de Park

La transformada de Park, también conocida como la transformación dq o síncrona, no es más que la transformada de Clarke combinada con la matriz de giro. Esta transformada aparte de añadir la reducción de variables, que ya se tenía con Clarke, permite que estas puedan verse como constantes siempre que la frecuencia de dichas señales coincida con la frecuencia de giro del nuevo sistema de referencia, dq. Esta transformación facilita trabajar con sistemas trifásicos, aunque también tiene sus inconvenientes como se verá posteriormente. En la Figura 20 se muestra la relación de los ejes dq.

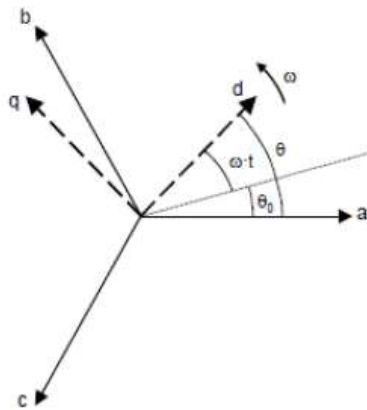


Figura 20. sistema de referencia trifásico y dq

Existen dos maneras de llegar a esta transformada, o bien de manera directa desde el vector abc, o bien desde los ejes $\alpha\beta 0$ siendo ángulo $\theta = \omega t$

Si se parte desde el vector abc utilizando la matriz de la ecuación (2).

$$\begin{bmatrix} x_d \\ x_q \\ x_0 \end{bmatrix} = \sqrt{\frac{2}{3}} \cdot \begin{bmatrix} \cos \omega t & \cos\left(\omega t - \frac{2\pi}{3}\right) & \cos\left(\omega t + \frac{2\pi}{3}\right) \\ \sin \omega t & \sin\left(\omega t - \frac{2\pi}{3}\right) & \sin\left(\omega t + \frac{2\pi}{3}\right) \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \cdot \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} \quad (2)$$

Si se parte desde los ejes $\alpha\beta 0$ utilizando la matriz de la ecuación (3).

$$\begin{bmatrix} x_d \\ x_q \\ x_0 \end{bmatrix} = \begin{bmatrix} \cos \omega t & \sin \omega t & 0 \\ -\sin \omega t & \cos \omega t & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_\alpha \\ x_\beta \\ x_0 \end{bmatrix} \quad (3)$$

El término "0" de ambos ejes como se ha comentado corresponde a la componente homopolar y adquiere relevancia en sistemas desequilibrados o cuando se desea analizar el comportamiento de las tensiones y corrientes a tierra, al no ser este análisis objeto del proyecto no se tendrá en cuenta esta componente.

La ganancia que multiplica a la matriz puede tener valores diversos en este caso se ha elegido $\sqrt{\frac{2}{3}}$ porque con este valor se consigue que la transformación sea invariante en potencia, es decir, la potencia será la misma independientemente de si el dominio es abc o dq [6].

4.1.3 Modelado en dq

Para obtener el modelo en ejes dq se parte de las ecuaciones de la bobina y el condensador (4) y (5).

$$v_c = \frac{1}{C} \cdot \int i_c \cdot dt \quad (4)$$

$$v_L = L \cdot \frac{di_L}{dt} \quad (5)$$

Y se le aplica la transformada de Park, (P), según la ecuación (6) .

$$v_{Ldq} = v_L \cdot P \quad (6)$$

Se obtienen las tensiones en d y q de la bobina, como puede verse en las ecuaciones (7) y (8).

$$v_{Ld} = L \cdot \frac{di_{Ld}}{dt} - L\omega i_q \quad (7)$$

$$v_{Lq} = L \cdot \frac{di_{Lq}}{dt} + L\omega i_d \quad (8)$$

Para obtener las tensiones d y q en el condensador se realiza de manera análoga a la bobina obteniendo las ecuaciones (9) y (10).

$$v_{cd} = \frac{1}{C} \int (i_{cd} \cdot dt) - \frac{1}{C\omega} i_q \quad (9)$$

$$v_{cq} = \frac{1}{C} \int (i_{cq} \cdot dt) + \frac{1}{C\omega} i_d \quad (10)$$

Si se desprecian los términos cruzados, tanto en las ecuaciones referidas a la bobina como al condensador, y se trasladan al dominio de Laplace las ecuaciones anteriores se obtienen las funciones de transferencia de las ecuaciones (11) y (12).

$$FT_{dq \text{ condensador}} = Cs \quad (11)$$

$$FT_{dq \text{ bobina}} = \frac{1}{Ls} \quad (12)$$

Como consecuencia de la simplificación de los términos cruzados, se tiene un filtro LCL como planta del sistema. En dicho filtro se tendrán en cuenta las diferentes resistencias asociadas a este, quedando el equivalente monofásico como en la Figura 21.

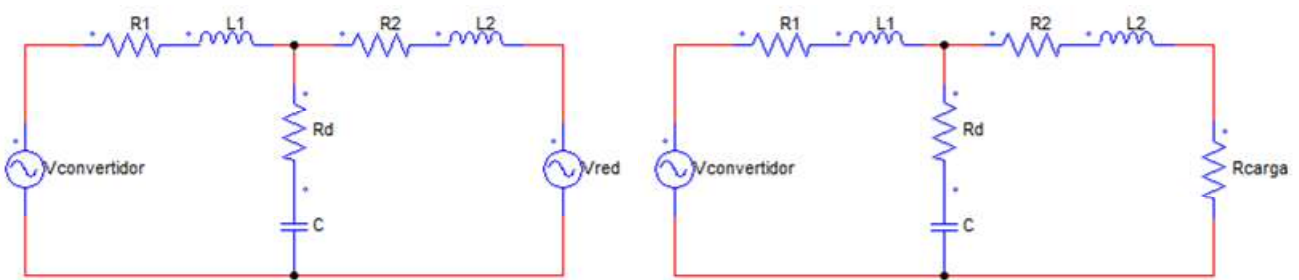


Figura 21. Modelos equivalentes monofásicos: izquierda conexión a red, derecha conexión a carga resistiva

Para obtener de una manera más simplificada la función de transferencia del sistema se utilizará el modelo equivalente monofásico por impedancias de la Figura 24.

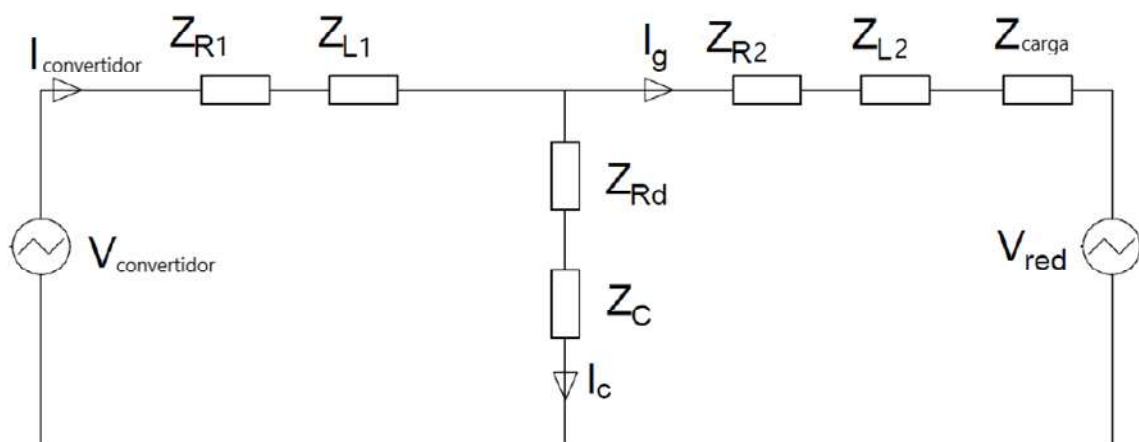


Figura 22. Equivalente monofásico en impedancias

La función de transferencia de ambos sistemas es similar, si se quiere obtener el de conexión a red basta con hacer nula la impedancia de carga.

Al pasar las impedancias al dominio de Laplace el resultado es el de las ecuaciones (13), (14), (15), (16), (17), (18) y (19).

$$Z_{R1} = R_1 \quad (13)$$

$$Z_{L1} = L_1 \cdot s \quad (14)$$

$$Z_{Rd} = R_d \quad (15)$$

$$Z_C = \frac{1}{C \cdot s} \quad (16)$$

$$Z_{R2} = R_2 \quad (17)$$

$$Z_{L2} = L_2 \cdot s \quad (18)$$

$$Z_{Rcarga} = R_{carga} \quad (19)$$

Siendo la función de transferencia resultante la ecuación (20).

$$Planta(s) = \frac{I_{conv}}{V_{conv}} = \frac{Z_{Rd} + Z_C + Z_{R2} + Z_{L2}}{(Z_{R2} + Z_{L2}) \cdot (Z_{Rd} + Z_C) + (Z_{R1} + Z_{L1}) \cdot (Z_{R2} + Z_{L2}) + (Z_{R1} + Z_{L1}) \cdot (Z_{Rd} + Z_C)} \quad (20)$$

Para obtener las tensiones dq que aplica el inversor, se puede asumir que el comportamiento de un filtro LCL es similar al del filtro L con respecto al control, siempre y cuando la corriente proporcionada por el condensador sea conocida y prácticamente constante [7].

Las ecuaciones (21), (22) y (23) describen la conexión del convertidor a la red a través de un filtro L y una carga resistiva.

$$v_{inva} = L \cdot \frac{di_a}{dt} + Ri_a + v_{reda} \quad (21)$$

$$v_{invb} = L \cdot \frac{di_b}{dt} + Ri_b + v_{redb} \quad (22)$$

$$v_{invc} = L \cdot \frac{di_c}{dt} + Ri_c + v_{redc} \quad (23)$$

Y se le aplica la transformada de Park, (P), según la ecuación (24).

$$v_{invdq} = P \cdot v_{abc} \quad (24)$$

Obteniendo las tensiones en d y q del inversor conectado a red, según las ecuaciones (25) y (26).

$$v_{invd} = L \cdot \frac{di_d}{dt} - L\omega i_q + v_{redd} \quad (25)$$

$$v_{invq} = L \cdot \frac{di_q}{dt} + L\omega i_d + v_{redq} \quad (26)$$

Si el inversor está conectado a una carga resistiva las ecuaciones que describen las tensiones en d y q son (27) y (28).

$$v_{invd} = L \cdot \frac{di_d}{dt} - L\omega i_q \quad (27)$$

$$v_{invq} = L \cdot \frac{di_q}{dt} + L\omega i_d + Ri_q + v_{redq} \quad (28)$$

Como puede observarse en la Figura 25 los ejes quedan acoplados esto adquiere relevancia cuando el sistema está conectado a red.

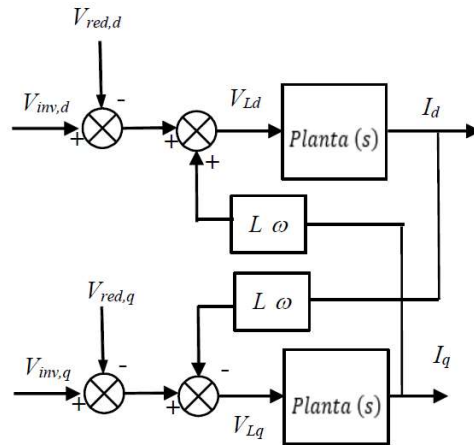


Figura 23. Acoplamiento de los ejes en el modelo dq

Las corrientes i_d e i_q serán controladas con las tensiones v_{invd} e v_{invq} , en caso de que se realice una compensación de los términos cruzados de forma ideal, se pueden considerar los ejes d y q como totalmente independientes para el modelo empleado. Debido a que una compensación ideal de los términos cruzados no es probable se debe conseguir, a través del control, que el acoplamiento de los ejes no afecte de manera significativa a la respuesta del sistema.

4.2 CONTROL DEL CONVERTIDOR

Una vez modelado el sistema en este capítulo se explicará el diseño del lazo de control de corriente del convertidor.

Los esquemas de control de ambas formas de conexión se pueden observar en las Figura (24) y (25). Con respecto a la conexión a una carga resistiva, uno de los aspectos importantes es que las corrientes de referencia deben ser impuestas por el usuario. Por otra parte, se necesitará generar un ángulo para poder aplicar la transformada de Park y obtener las corrientes en dq, medidas y filtradas, este ángulo también será necesario para aplicar la antitransformada de Park y obtener así las modulantes.

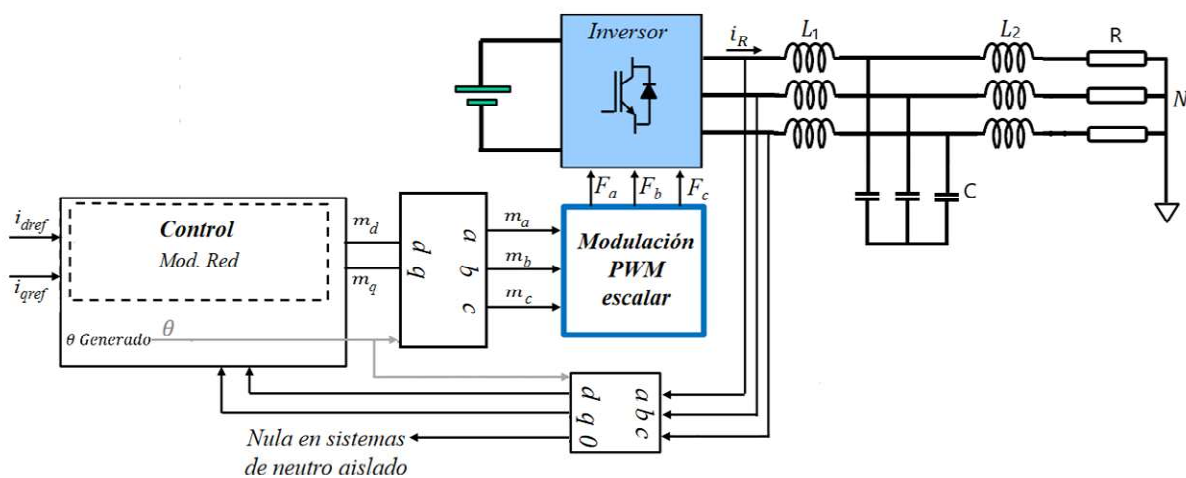


Figura 24. Sistema de control del inversor conectado a una carga resistiva

Para la conexión a red será necesario medir, además de las corrientes por las fases, las tensiones de la red. Con dichas tensiones y a través de un sistema PLL se obtendrá el ángulo de esta, que será el utilizado para poder aplicar la transformada y antitransformada de Park en este caso. Por otro lado, las corrientes de referencia serán calculadas a partir de las potencias activa y reactiva de referencia.

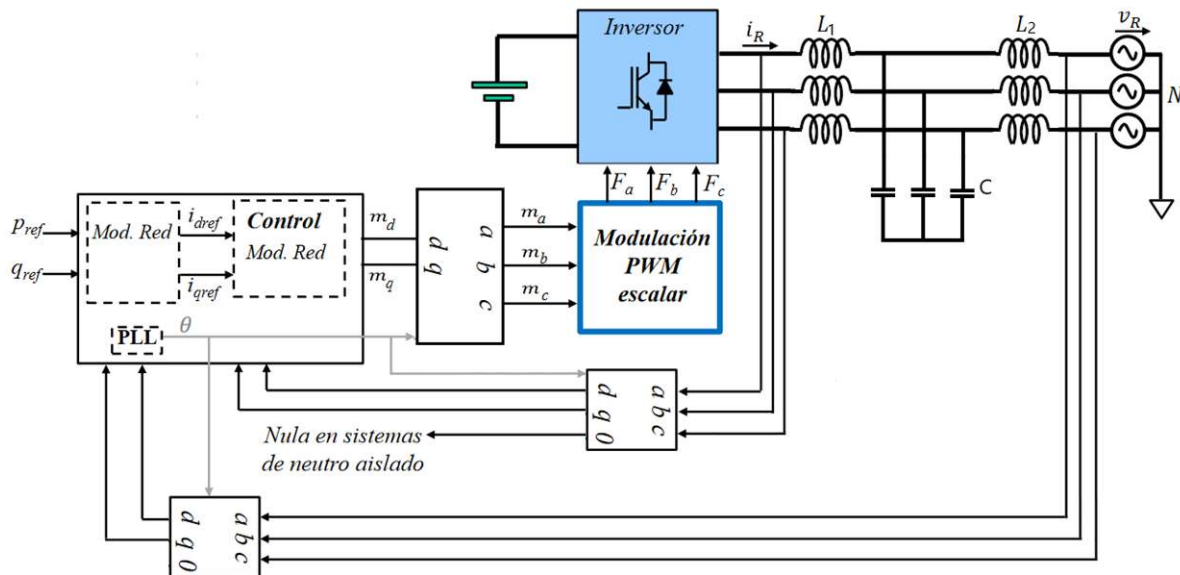


Figura 25. Sistema de control del inversor conectado a red

4.2.1 Descripción del lazo de control

Al trabajar en el dominio dq se requieren dos lazos que controlen la corriente en ambos ejes, que son los que se muestran en las Figuras 26 y 27.

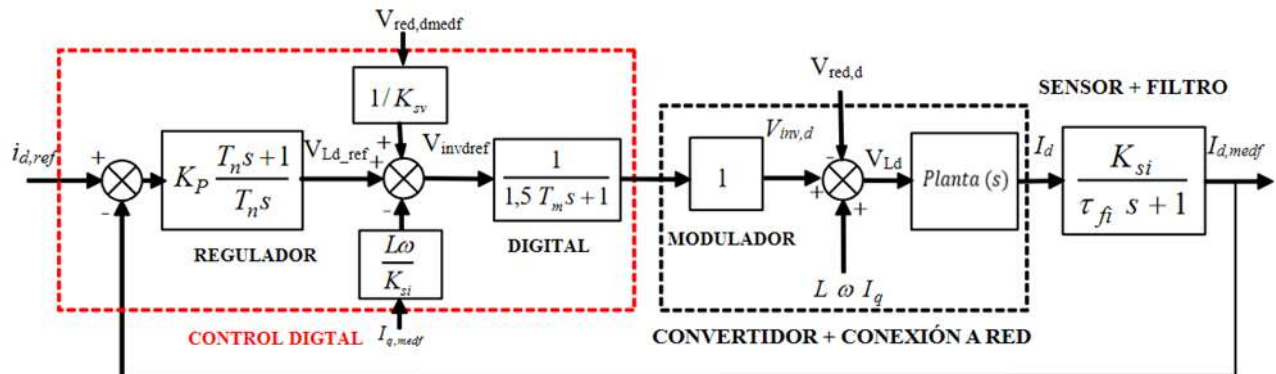


Figura 26. Lazo de corriente Id

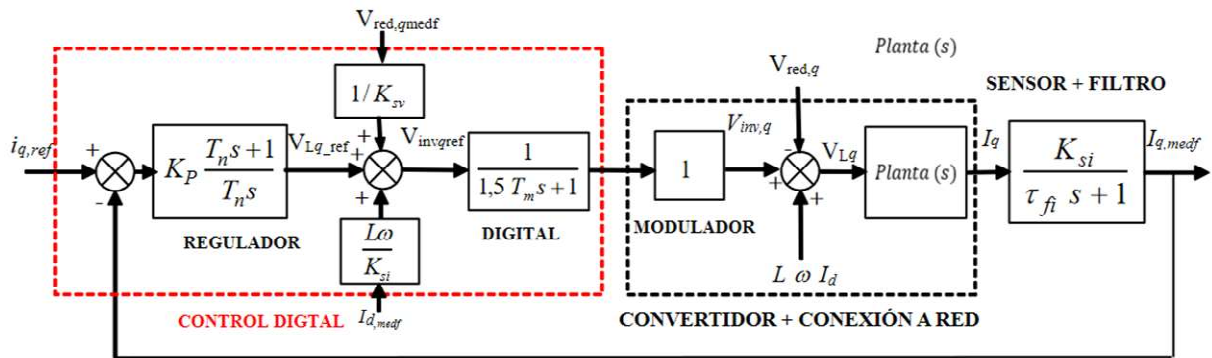


Figura 27. Lazo de corriente I_q

Se observa que ambos lazos son un bucle realimentado negativamente que compara la señal de referencia con la medida y filtrada, obteniendo el error y haciéndolo pasar por el regulador PI. Dicho regulador PI está dentro del control digital del sistema que, tras realizar las acciones oportunas, calcula las modulantes que serán comparadas con la señal triangular del PWM para controlar la conmutación de los IGBT's del inversor. De la tensión aplicada a la planta del sistema resultará una corriente a la salida. Por último, esta corriente será filtrada y llevada nuevamente a ser comparada con la referencia.

En el interior del lazo se tiene en cuenta el acoplamiento de los ejes dq. Se hace necesario compensar a través del control la influencia de la tensión de red y de los términos cruzados, para que esta no afecte a la estabilidad y rapidez del sistema. Esta compensación es llevada a cabo a la salida de los controladores siendo la tensión resultante la utilizada para realizar la modulación PWM que se encargará de enviar las señales correspondientes para las conmutaciones de los IGBT's.

Los lazos de control están formados por un controlador PI que será programado en un microcontrolador, por lo que será un controlador discreto, sin embargo, para el análisis teórico se tratará como si se implementara de manera analógica. En el sistema real el control será implementado en un DSP.

Después del controlador se tiene un bloque que emula los retrasos del convertidor y de la modulación PWM, y tras este bloque viene la planta comentada anteriormente.

Por último, se tiene un filtro que será implementado de manera digital.

4.2.2 Filtro Digital

Debido a que el sistema real no cuenta con un filtro analógico es necesario implementar un filtro digital.

La función de transferencia en el dominio de Laplace es la que se muestra en la ecuación (29).

$$FT_{Filtro} = \frac{K_{si}}{\tau_{fi} \cdot s + 1} \quad (29)$$

Siendo la constante de tiempo del filtro la ecuación (30).

$$\tau_{fi} = \frac{1}{2 \cdot \pi \cdot f_{fi}} \quad (30)$$

La frecuencia de corte de este filtro se establece en 2 kHz, por lo que la función de transferencia del filtro se puede ver en la ecuación (31).

$$FT_{\text{Filtro Analógico}} = \frac{1}{79.58 \cdot 10^{-6} \cdot s + 1} \quad (31)$$

Para su implementación digital, se discretiza, dando lugar a la ecuación (32).

$$FT_{\text{Filtro Digital}} = \frac{0.3859 \cdot z + 0.3859}{z - 0.2283} \quad (32)$$

En la Figura 28 se puede observar el diagrama de Bode tanto del filtro analógico como del filtro digital.

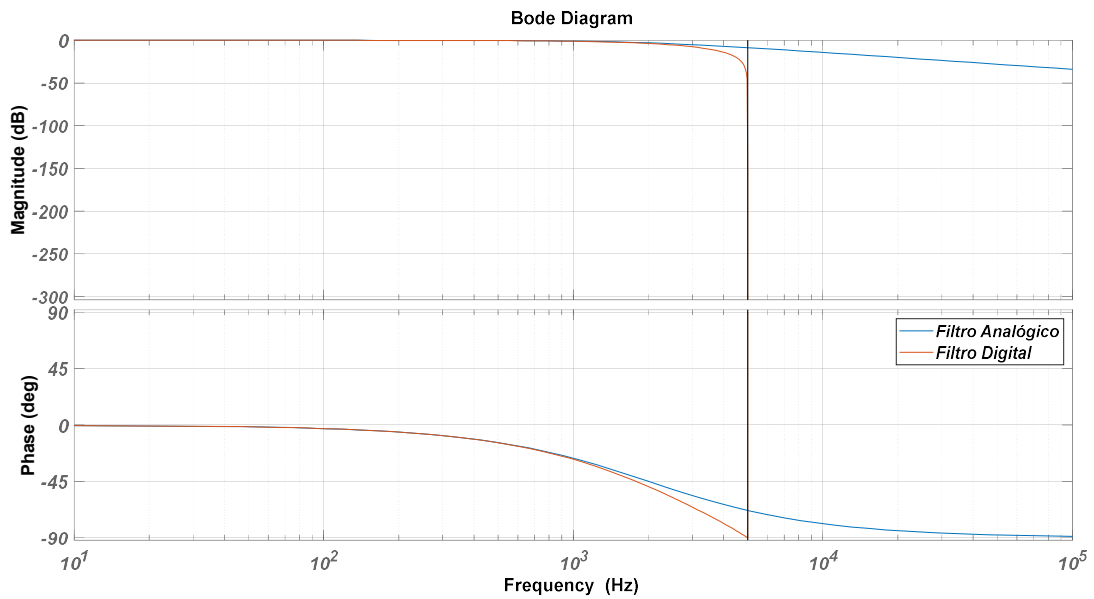


Figura 28. Diagrama de Bode del filtro analógico y del filtro digital

4.2.3 Modelo de la digitalización y PWM

Al trabajar con controladores digitales como consecuencia del tiempo de cálculo del procesador se produce un retraso de un periodo de muestreo, además se tiene el retraso debido al retenedor de orden cero que modela al modulador PWM, la modelización de estos retrasos se puede observar en la Figura 29.

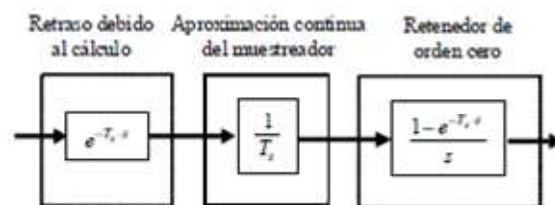


Figura 29. Modelización de los retrasos debidos a la digitalización

La función de transferencia utilizada para este modelado es la presente en la ecuación (33). Modela el retraso total como 1.5 veces el periodo de muestreo y es válida únicamente cuando se trabaja con frecuencias inferiores a la frecuencia de muestreo entre 20. En este proyecto la frecuencia de muestreo empleada será de 10 kHz, por ello la frecuencia de corte de los lazos de control deberá ser menor de 500 Hz.

$$e^{-T_m \cdot s} \cdot \frac{1}{T_m} \cdot \frac{1 - e^{-T_m \cdot s}}{s} \approx \frac{1}{1,5 \cdot T_m \cdot s + 1} \quad (33)$$

4.2.4 Cálculo de Kp y Tn del lazo de corriente con carga resistiva

Para el cálculo de los parámetros del controlador se deberán imponer unos requisitos de respuesta al diagrama de Bode del lazo abierto del sistema, que se traducen en un margen de fase y una frecuencia de corte. Cuando dichos requisitos estén impuestos, los parámetros del controlador, la variable de integración (Tn) y la ganancia (Kp), serán tales que se cumplan con los requisitos impuestos al lazo. Los requisitos del lazo son los siguientes:

- $MF = 60^\circ$
- *frecuencia de corte*, $\omega_{ci} = 350 \text{ Hz}$

La función de transferencia del controlador la ecuación (34).

$$FT_{PI} = \frac{K_p \cdot (T_n \cdot s + 1)}{T_n \cdot s} \quad (34)$$

La frecuencia de muestreo (f_m) y de conmutación (f_{conm}) será de 10 kHz por lo que se muestreará una vez por periodo de conmutación, otras características del lazo son:

- *Tiempo de muestreo*, $T_m = \frac{1}{f_m}$
- $L_1 = 2.2 \text{ mH}$
- $L_2 = 1.098 \text{ mH}$
- $R_1 = 0.1 \Omega$
- $R_2 = 0.1 \Omega$
- $R_{carga} = 30 \Omega$
- $K_{si} = 1$
- $f_{fi} = 2 \text{ kHz}$
- $\omega_{fi} = 2 \cdot \pi \cdot f_{fi} = 2 \cdot \pi \cdot 2 \cdot 10^3 \text{ rad/s}$

La función de transferencia en lazo abierto del sistema se puede ver en la ecuación (35).

$$FT_{LA} = \frac{K_p \cdot (T_n \cdot s + 1)}{T_n \cdot s} \cdot \frac{1}{1,5 \cdot T_m \cdot s + 1} \cdot \text{Planta}(s) \cdot \frac{K_{si}}{\tau_{fi} \cdot s + 1} \quad (35)$$

Para el cálculo de T_n se fija el margen de fase y se sustituye la ecuación (36) en la ecuación (37).

$$s = j \cdot \omega_{ci} \quad (36)$$

$$MF = 180^\circ + \varphi_{\text{Sistema}}(s) = 180^\circ + \varphi_{PI}(s) + \varphi_{\text{Planta}}(s) + \varphi_{\text{Digitalización}}(s) + \varphi_{\text{Filtro}}(s) \quad (37)$$

Para el cálculo de K_p se iguala el módulo de la función de transferencia en lazo abierto del sistema a la unidad y sustituye la ecuación (36) en la ecuación (38).

$$1 = \left| \frac{K_p \cdot (T_n \cdot s + 1)}{T_n \cdot s} \right| \cdot \left| \frac{1}{1,5 \cdot T_m \cdot s + 1} \right| \cdot |\text{Planta}(s)| \cdot \left| \frac{K_{si}}{\tau_{fi} \cdot s + 1} \right| \quad (38)$$

Los parámetros del controlador obtenidos son los siguientes:

$$K_p = 0.76$$

$$T_n = 1.11 \cdot 10^{-5} s$$

4.2.5 Comprobación del ajuste del controlador PI

Para que un sistema sea estable tanto el margen de ganancia como el margen de fase han de ser positivos.

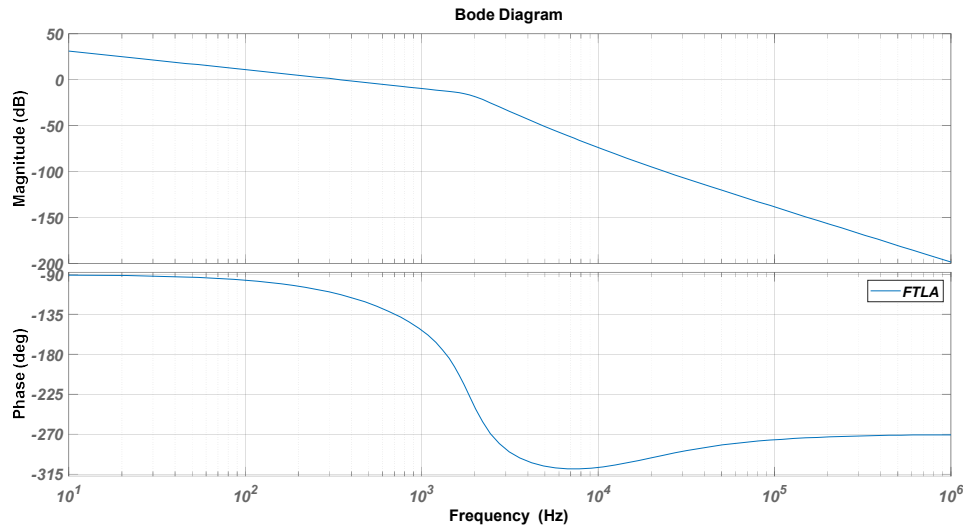


Figura 30. Diagrama de Bode de la función de transferencia en lazo abierto

En la Figura 30 puede observarse como los valores de MF y frecuencia de corte de la función de transferencia en lazo abierto en el diagrama de Bode son coincidentes con los valores fijados, esto quiere decir que la sintonización del controlador PI es correcta.

En la Figura 31 se tiene la función de transferencia en lazo cerrado del sistema.

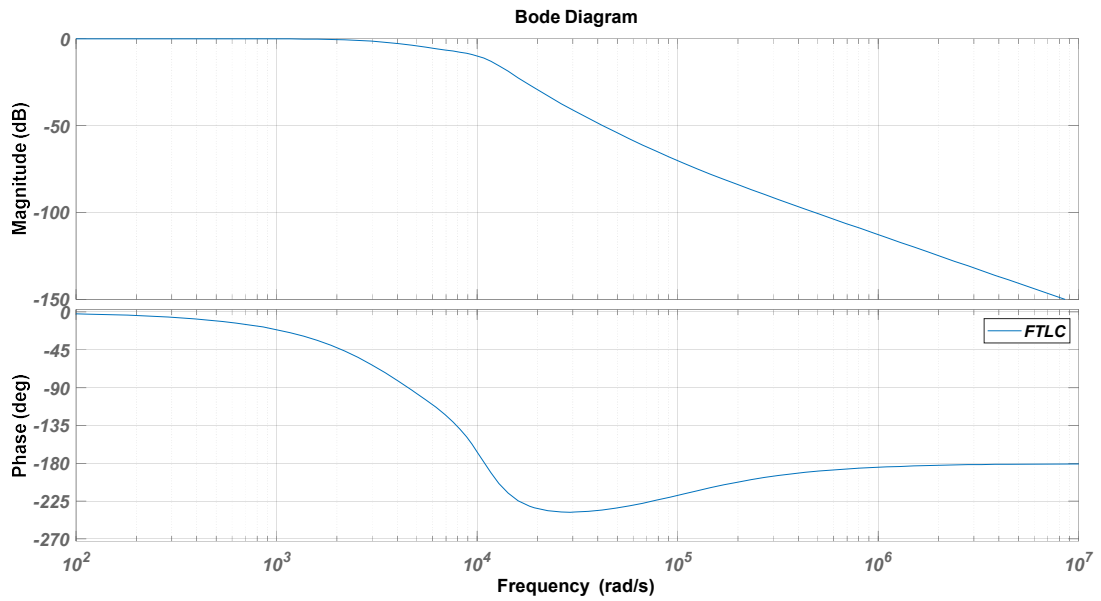


Figura 31. Diagrama de Bode de la función de transferencia en lazo cerrado

Tras ver que el resultado tanto en lazo abierto como el lazo cerrado del sistema es el esperado, se analiza la respuesta del sistema ante un escalón, mostrada en la Figura 32, y se compara con la obtenida por simulación, Figura 33.

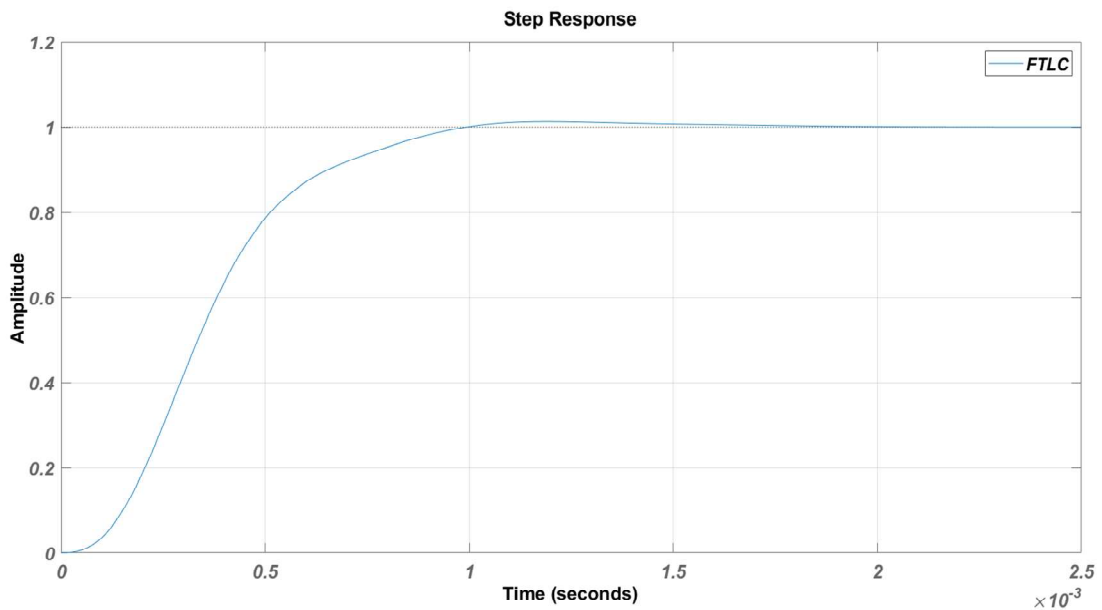


Figura 32. Respuesta ante un escalón del sistema

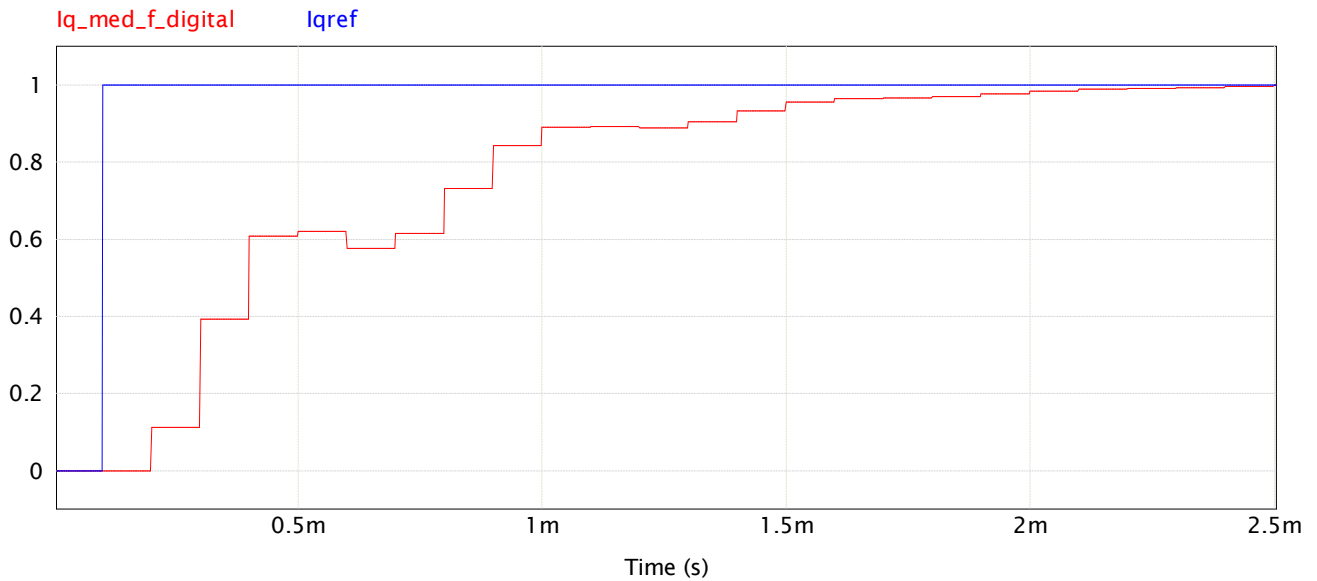


Figura 33. Respuesta del sistema ante un escalón PSIM

El tiempo de estabilización de la respuesta a escalón obtenido en Matlab es de $9 \cdot 10^{-4}s$, mientras que el tiempo de estabilización obtenido en PSIM es de $1.9 \cdot 10^{-3}s$. Esta diferencia se debe a que en el modelo de Matlab al ser un modelo más sencillo no se han considerado los términos cruzados, sin embargo, en el modelo de PSIM sí que se han tenido en cuenta estos términos, y al no existir un desacoplamiento total de los lazos, la respuesta del sistema se ve afectada, con un ligero retraso.

4.3 CONEXIÓN A RED

Uno de los requisitos para poder conectar el inversor a la red es que antes de la conexión la tensión generada por el inversor y la de la red se encuentren con la misma amplitud y estén en fase, de lo contrario se pueden producir grandes picos de corriente que puedan dañar los equipos. Como consecuencia de esto, es crítico que ambas magnitudes se puedan determinar de manera precisa.

Por un lado, que la tensión de la red y el inversor tengan la misma amplitud es relativamente sencillo basta con realizar la transformada de Park a la tensión de la red obteniendo V_d y V_q y hacer que el inversor a través de las modulantes consiga generar una tensión de la misma amplitud.

Por otro lado, ponerlas con la misma fase se torna más complicado. Uno de los métodos ampliamente utilizados para detectar la fase de la red, y el que se utilizará en este proyecto, es una PLL (Phase Locked Loop).

4.3.1 PLL

El sistema PLL está formado por un controlador PI y cuenta con una realimentación del ángulo. Partiendo de la tensión de la red y con la ayuda de la transformada de Park, se obtienen las componentes V_d y V_q . Se utiliza la componente V_q como entrada al lazo y tras pasar por el PI se obtiene la frecuencia de la red, al integrarla se consigue obtener el ángulo de esta. Este ángulo se limita entre 0 y 2π y es el que realimenta al lazo. Cuando el sistema PLL consigue hacer que la componente V_q sea 0 quiere decir que la tensión de la red esta alineada con V_d , asegurando así que el ángulo que se obtiene es el correcto.

El lazo del sistema PLL, que se puede observar en de la Figura 34, es un lazo lento y si se quiere conseguir una respuesta más rápida hay que inicializar el sistema en la frecuencia de la red, este término es representado por ω_n en Figura 35, con esta inicialización se consigue obtener un transitorio más rápido.

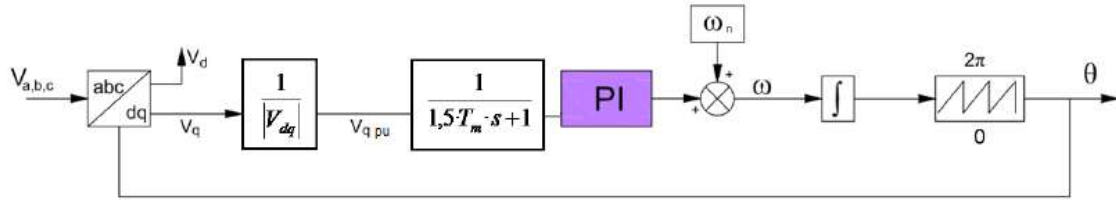


Figura 34. Diagrama de bloques de un PLL

○ Cálculo de K_p y T_n del sistema PLL

Los requisitos de respuesta del lazo son los siguientes:

- $MF = 60^\circ$
- frecuencia de corte, $\omega_{ci} = 10 \text{ Hz}$

La función de transferencia del controlador es la ecuación (39).

$$FT_{PI\ PLL} = \frac{K_{p\ PLL} \cdot (T_{n\ PLL} \cdot s + 1)}{T_{n\ PLL} \cdot s} \quad (39)$$

Otras características del lazo son:

- frecuencia de muestreo, $f_m = 10 \text{ kHz}$
- Tiempo de muestreo, $T_m = \frac{1}{f_m}$

La función de transferencia en lazo abierto del sistema PLL se puede ver en la ecuación (40).

$$FT_{LA\ PLL} = \frac{K_{p\ PLL} \cdot (T_{n\ PLL} \cdot s + 1)}{T_{n\ PLL} \cdot s} \cdot \frac{1}{1,5 \cdot T_m \cdot s + 1} \cdot \frac{1}{s} \quad (40)$$

Para el cálculo de $T_{n\ PLL}$ se fija el margen de fase y se sustituye la ecuación (41) en la ecuación (42).

$$s = j \cdot \omega_{ci} \quad (41)$$

$$MF = 180^\circ + \varphi_{Sistema}(s) = 180^\circ + \varphi_{PI}(s) + \varphi_{Digitalización}(s) + \varphi_{Integrador}(s) \quad (42)$$

Para el cálculo de $K_{p\ PLL}$ se iguala el módulo de la función de transferencia en lazo abierto del sistema a la unidad y sustituye la ecuación (41) en la ecuación (43).

$$1 = \left| \frac{K_{p\ PLL} \cdot (T_{n\ PLL} \cdot s + 1)}{T_{n\ PLL} \cdot s} \right| \cdot \left| \frac{1}{1,5 \cdot T_m \cdot s + 1} \right| \cdot \left| \frac{1}{s} \right| \quad (43)$$

Los parámetros del controlador obtenidos son los siguientes:

$$K_p = 54.71$$

$$T_n = 2.82 \cdot 10^{-2} s$$

4.3.2 Comprobación del ajuste del controlador PI de la PLL

En la Figura 35 puede observarse como los valores de MF y frecuencia de corte de la función de transferencia en lazo abierto en el diagrama de Bode son coincidentes con los valores fijados, esto quiere decir que la sintonización del controlador PI es correcta.

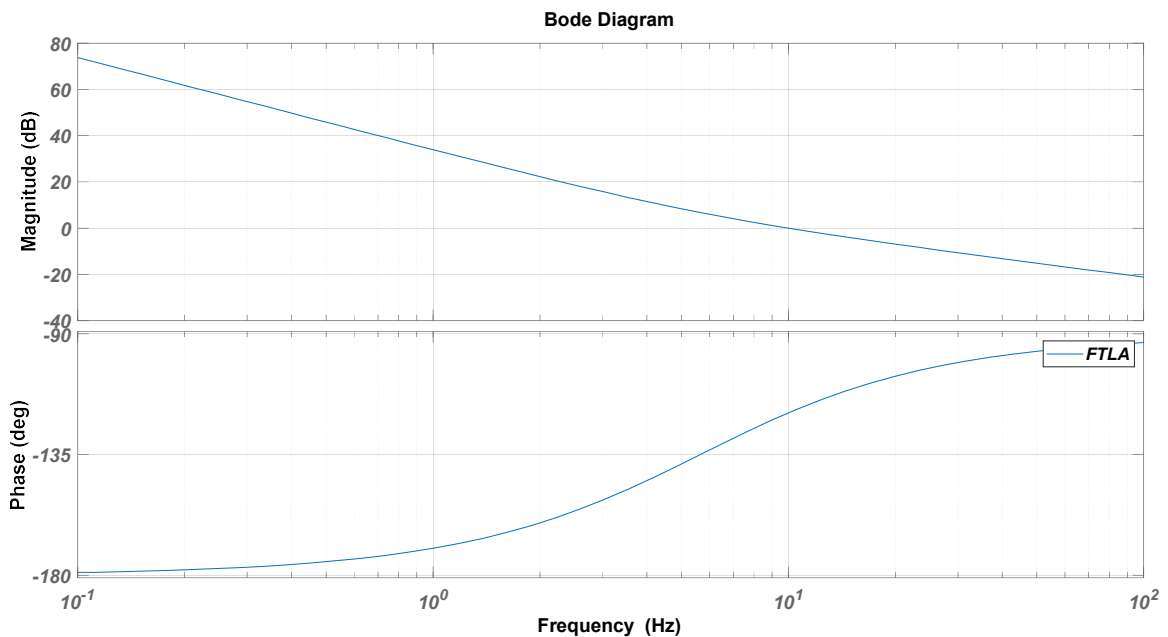


Figura 35. Diagrama de Bode de la función de transferencia de la PLL en lazo abierto

4.3.3 Problemática del filtro LCL en la conexión a red

En el apartado anterior no se ha tenido en cuenta la frecuencia de resonancia introducida por el filtro LCL debido a que al estar conectado a una carga resistiva quedaba lo suficientemente amortiguado, pero ahora ya no se dispone de ese amortiguamiento por lo que deberá ser analizada.

El fenómeno de resonancia es un aspecto que presentan los filtros de segundo y tercer orden y que hay que tener en cuenta. La resonancia debido al filtro LCL, hace que, ante los cambios en escalón de la tensión aplicada por el convertidor, la corriente de red presente oscilaciones a la frecuencia de resonancia del filtro.

La frecuencia de resonancia del filtro ω_n viene dada por la ecuación (44):

$$\omega_n = \frac{1}{\sqrt{L_{eq} \cdot C}} = \frac{1}{\sqrt{\frac{L_1 \cdot L_2 \cdot C}{L_1 + L_2}}} = \frac{1}{\sqrt{\frac{2.2 \cdot 10^{-3} \cdot 1.1 \cdot 10^{-3} \cdot 4.7 \cdot 10^{-6}}{(2.2 \cdot 10^{-3} + 1.1 \cdot 10^{-3})}}} = 17033 \frac{rads}{s} = 2710 \text{ Hz} \quad (44)$$

Las oscilaciones en la corriente de red debido a la resonancia generan dos problemas:

- Introducción de armónicos de baja frecuencia en la red
- Cuando se implementa un control digital, dependiendo del retraso de la digitalización, la resonancia puede inestabilizar el sistema.

Estos problemas llevan a la necesidad de tener que amortiguar la resonancia. Este amortiguamiento puede ser de dos maneras activo o pasivo. Por un lado, el amortiguamiento pasivo que consiste la colocación de una resistencia en el filtro LCL con objeto de disipar la energía involucrada en la oscilación de la resonancia. Sin embargo, la disipación de energía en la resistencia no solo será a la frecuencia de resonancia, sino también se dará a la frecuencia de red y a la de los armónicos de conmutación. Esta disipación de energía se traducirá en pérdidas, lo que reducirá el rendimiento de la conversión [8].

Por otro lado, el amortiguamiento activo permite amortiguar el sistema mediante algoritmos de control, prescindiendo del uso de resistencias, lo que provoca la ausencia de pérdidas. La desventaja es que al depender del algoritmo de control su rango está limitado [8].

En este proyecto, por sencillez, se utilizará un amortiguamiento pasivo añadiendo una resistencia en serie con el condensador.

Para asegurar un lazo lo suficientemente estable a la hora de introducir el amortiguamiento pasivo se deben tener en cuenta los siguientes aspectos:

- Se debe introducir una resistencia que reduzca la ganancia del sistema cuando se produce la resonancia serie equivalente, este aspecto está relacionado con la frecuencia de resonancia
- Se debe realizar de forma que la potencia disipada en la resistencia sea lo menor posible, minimizando las pérdidas, este aspecto está relacionado con la frecuencia de red.
- Se debe conseguir un filtrado suficiente a las frecuencias de las emisiones conducidas, aspecto relacionado con la frecuencia de conmutación [8].

Para sistemas típicos en el control de convertidores de potencia, el análisis de la estabilidad a partir del diagrama de Bode de la función de transferencia en lazo abierto suele ser suficiente, sin embargo, cada vez son más los convertidores en los que el sistema a controlar es avanzado. En estos casos se hace necesario conocer cuando se puede aplicar el criterio de Bode con garantías. Para que el criterio de Bode asegure el correcto análisis de la estabilidad del sistema, este tiene que cumplir en lazo abierto:

- Tener una única frecuencia de corte y un único margen de ganancia
- No tener polos inestables en lazo abierto
- No tener más de un polo en el origen
- Tener ganancia positiva a frecuencia nula.

Si no se cumplen estos requisitos el criterio de Bode no asegura la estabilidad del sistema, y en estos casos se debe recurrir a otro criterio válido (Criterio de Bode Generalizado [9], criterio de Nyquist, comprobación de los polos del sistema en lazo cerrado, etc.)

En este proyecto al no ser un sistema de control avanzado se considera suficiente el análisis de la función de transferencia en lazo abierto del sistema a través del diagrama de Bode.

4.3.4 Cálculo de la K_p y T_n del lazo de corriente para la conexión a red

Los requisitos del lazo son los siguientes:

- $MF = 55^\circ$
- frecuencia de corte, $\omega_{ci} = 350 \text{ Hz}$

La función de transferencia del controlador la ecuación (45).

$$FT_{PI} = \frac{K_p \cdot (T_n \cdot s + 1)}{T_n \cdot s} \quad (45)$$

Otras características del lazo son:

- *Tiempo de muestreo*, $T_m = \frac{1}{f_m}$
- $L_1 = 2.2 \text{ mH}$
- $L_2 = 1.098 \text{ mH}$
- $R_1 = 0.1 \Omega$
- $R_2 = 0.1 \Omega$
- $R_d = 0/1/5/10 \Omega$
- $K_{Si} = 1$
- $f_{fi} = 2 \text{ kHz}$
- $\omega_{fi} = 2 \cdot \pi \cdot f_{fi} = 2 \cdot \pi \cdot 2 \cdot 10^3 \text{ rad/s}$

La función de transferencia en lazo abierto del sistema se puede ver en la ecuación (46).

$$FT_{LA} = \frac{K_p \cdot (T_n \cdot s + 1)}{T_n \cdot s} \cdot \frac{1}{1,5 \cdot T_m \cdot s + 1} \cdot \text{Planta}(s) \cdot \frac{K_{Si}}{\tau_{fi} \cdot s + 1} \quad (46)$$

Para el cálculo de T_n se fija el margen de fase y se sustituye la ecuación (47) en la ecuación (48).

$$s = j \cdot \omega_{ci} \quad (47)$$

$$\begin{aligned} MF &= 180^\circ + \varphi_{\text{Sistema}}(s) \\ &= 180^\circ + \varphi_{PI}(s) + \varphi_{\text{Planta}}(s) + \varphi_{\text{Digitalización}}(s) + \varphi_{\text{Filtro}}(s) \end{aligned} \quad (48)$$

Para el cálculo de K_p se iguala el módulo de la función de transferencia en lazo abierto del sistema a la unidad y sustituye la ecuación (47) en la ecuación (49).

$$1 = \left| \frac{K_p \cdot (T_n \cdot s + 1)}{T_n \cdot s} \right| \cdot \left| \frac{1}{1,5 \cdot T_m \cdot s + 1} \right| \cdot |\text{Planta}(s)| \cdot \left| \frac{K_{Si}}{\tau_{fi} \cdot s + 1} \right| \quad (49)$$

Los parámetros del controlador obtenidos son los siguientes:

$$K_p = 8.06$$

$$T_n = 5 \cdot 10^{-3} \text{ s}$$

4.3.5 Análisis del sistema con distintas resistencias de amortiguamiento (R_d)

En la Figura 36 se puede observar un diagrama de Bode con la planta del sistema y diferentes resistencias de amortiguación.

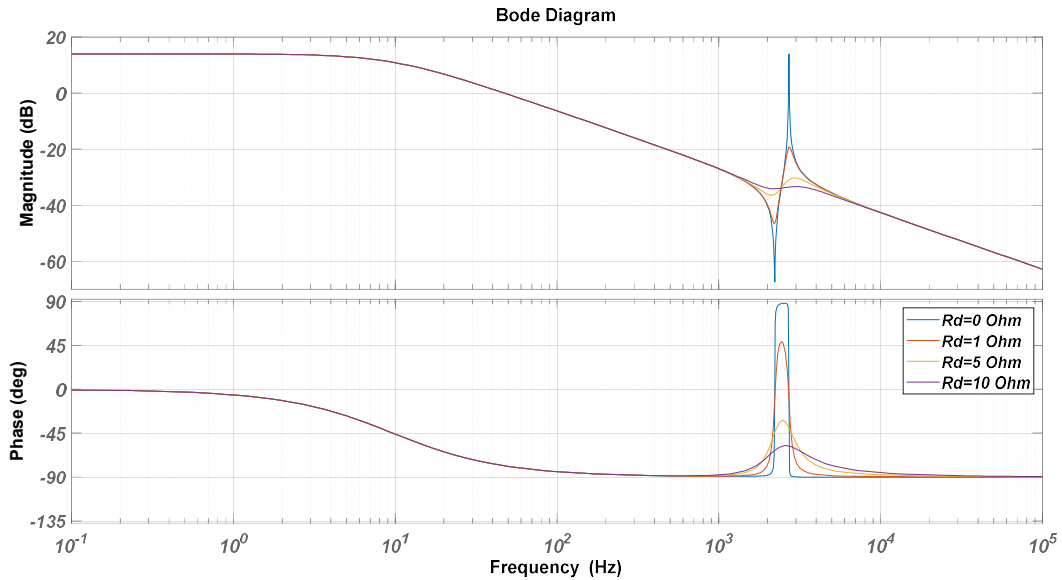


Figura 36. Diagrama de Bode de la planta del sistema con diferentes valores de la resistencia de amortiguamiento

La planta sin resistencia de amortiguamiento presenta un gran pico de resonancia entorno a la frecuencia de 2710 Hz. A medida que se aumenta la resistencia de amortiguamiento este pico se va reduciendo y quedando amortiguado.

Si se mantienen los parámetros del control, se realiza la función de transferencia en lazo abierto del sistema, y se representa el diagrama de Bode el resultado es el de la Figura 37. No se ha representado la función de transferencia en lazo abierto con una resistencia de amortiguamiento nula debido a que en este caso el sistema es inestable.

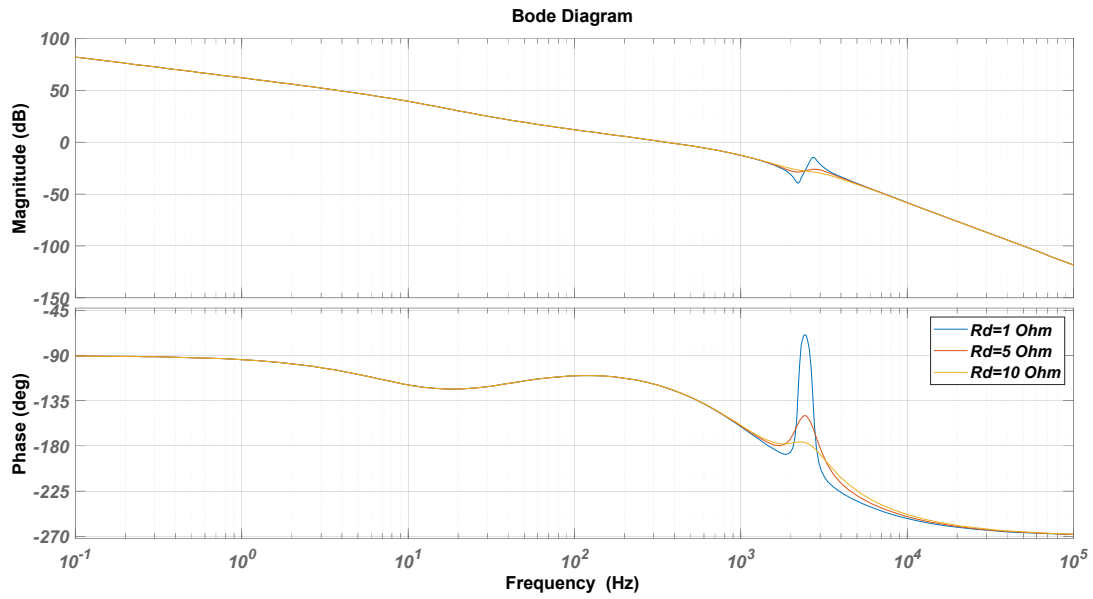


Figura 37. Diagrama de Bode de la función de transferencia del sistema en lazo abierto con distintos amortiguamientos

Si se mantienen los parámetros del control, se realiza la función de transferencia en lazo cerrado del sistema, y se representa el diagrama de Bode el resultado es el de la Figura 38.

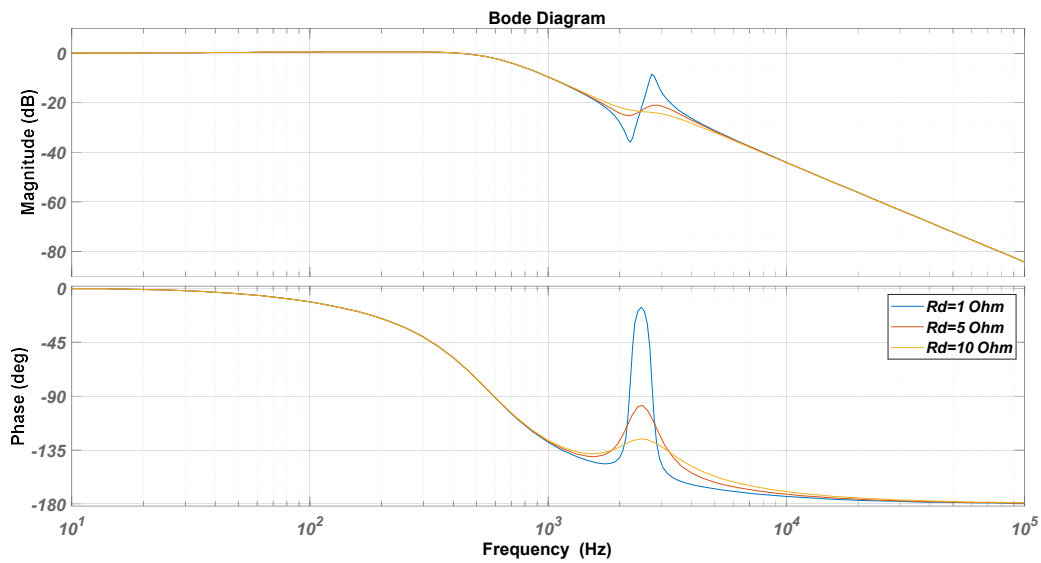


Figura 38. Diagrama de Bode de la función de transferencia del sistema en lazo cerrado con distintos amortiguamientos

Por último, se comprueba la respuesta ante un escalón del sistema con distintos amortiguamientos como puede verse en la Figura 39.

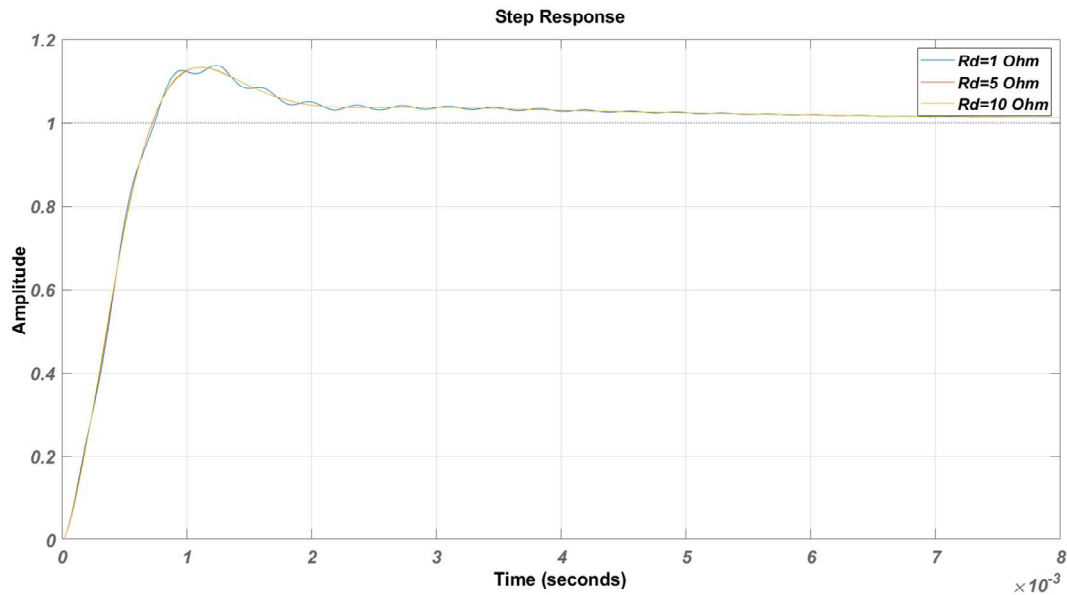


Figura 39. Respuesta ante un escalón del sistema con distintos amortiguamientos

Como puede observarse cuando se introduce una resistencia de amortiguamiento de $1 \text{ } \Omega$, la respuesta presenta oscilaciones. Sin embargo, cuando se aumenta esta resistencia hasta $5 \text{ } \Omega$ estas oscilaciones desaparecen. La respuesta con una resistencia de $5 \text{ } \Omega$ y de $10 \text{ } \Omega$ es coincidente por lo que en este caso se utilizará la resistencia de $5 \text{ } \Omega$ por generar unas menores pérdidas.

En la Figura 40 se puede observar el comportamiento de la potencia activa del sistema ante un escalón pasando de 0 a 1 kW, en el modelo de PSIM.

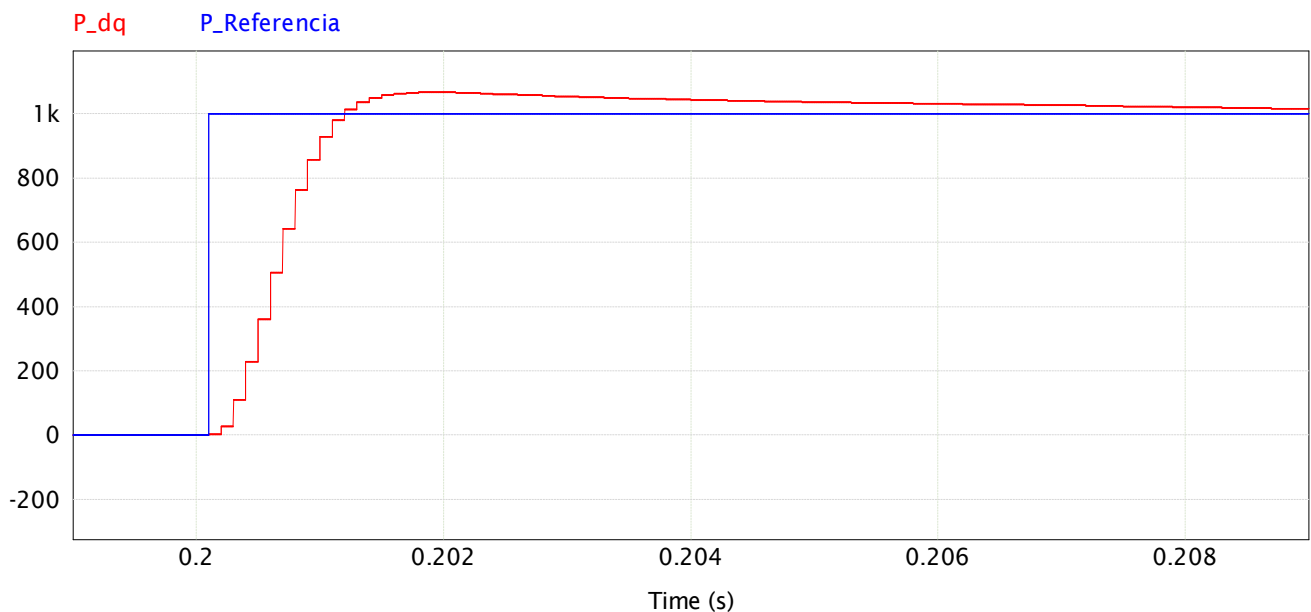


Figura 40. Respuesta del sistema ante un escalón de 0 a 1 kW en la potencia activa PSIM

El tiempo de estabilización de la respuesta a escalón obtenido en Matlab es de $6 \cdot 10^{-3} s$, mientras que el tiempo de estabilización obtenido en PSIM es de $7 \cdot 10^{-3} s$. Esta diferencia, al igual que en la conexión con una carga, se debe a que en el modelo de Matlab es más simple que el modelo de PSIM y no tiene en cuenta el acoplamiento de los lazos.

Por otro lado, se puede observar una sobre oscilación del 13% en el modelo de Matlab del mientras que en PSIM está sobre oscilación es del 7%. La sobre oscilación habitualmente está relacionada con la rapidez de la respuesta del sistema, cuanto mayor sea esta más sobre oscilación habrá, en este caso al tener el modelo de PSIM una dinámica más lenta, esto favorece a que no se tenga un sobre oscilación tan alta.

4.4 RESUMEN DE PARAMETROS

En las Tablas 3, 4, 5 y 6 se recoge un resumen de los parámetros del sistema y de los diferentes controles.

PARÁMETROS GENERALES	
L_1	2.2 mH
L_2	1.098 mH
R_1	0.1 Ω
R_2	0.1 Ω
C	4.7 μF
Frecuencia de muestreo	10 kHz
Periodo de muestreo	$1 \cdot 10^{-4} s$
K_{si}	1
Frecuencia corte filtro f_{fi}	2 kHz
Constante tiempo del filtro τ_{fi}	$79.57 \cdot 10^{-6} s$

Tabla 3. Parámetros generales del sistema

CONTROL CARGA RESISTIVA	
MF	60°
ω_{CI}	350 Hz
K_P	0.52
T_N	$7.82 \cdot 10^{-6} s$
R_{carga}	30 Ω

Tabla 4. Parámetros del control con carga resistiva

CONTROL PLL	
MF	60°
ω_{CI}	10 Hz
K_{PPLL}	54.71
T_{NPLL}	$2.82 \cdot 10^{-2} s$

Tabla 5. Parámetros del control de la PLL

CONTROL CONEXIÓN RED	
MF	55°
ω_{CI}	350 Hz
K_P	8.06
T_N	$5 \cdot 10^{-3} s$
R_d	5 Ω

Tabla 6. Parámetros del control de corriente con conexión a red

5. IMPLEMENTACIÓN DEL LAZO DE CONTROL EN EL PROTOTIPO

5.1 DESCRIPCIÓN DE LIBRERÍAS Y FUNCIONES NECESARIAS PARA EL CONTROL DEL PROTOTIPO

El software de partida necesita de ciertas librerías de extensión .h y de archivos de código de extensión .c para funcionar. Siendo un total de 16 archivos de código, estos archivos forman los diferentes módulos que serán ejecutados en función de la orden que se desea dar en cada momento [5]. En la Figura 41 se muestran estos archivos.

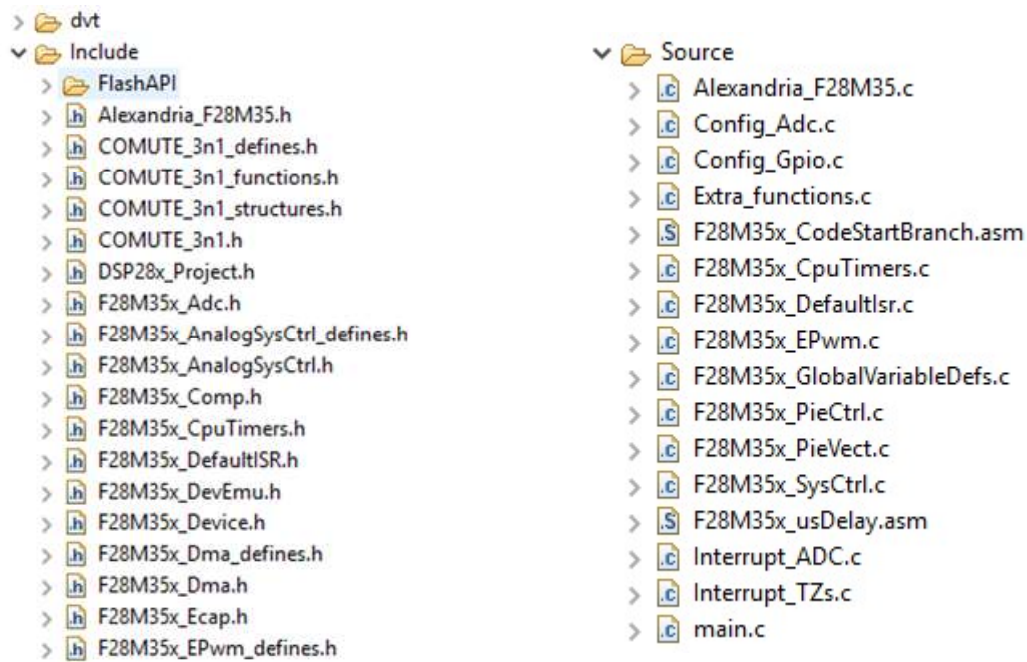


Figura 41. Archivos del código ".c" y librerías ".h"

Los archivos de código de extensión .c son archivos fuente que se compilarán. Los archivos .h se utilizan para exponer la API de un programa a cualquier otra parte de ese programa.

Debido a que se partía de un proyecto previo, como se ha mencionado anteriormente, únicamente se mencionará los archivos y librerías que se han modificado y/o se consideran relevantes para un uso futuro del inversor.

Los archivos de código ".c" y librerías ".h" más relevantes son:

- main.c
- Interrupt_ADC.c
- Interrupt_TZs.c
- CONMUTE_3n_1_structures.h
- CONMUTE_3n_1_defines.h
- CONMUTE_3n_1.h
- Extra_functions.c
- F28M35x_EPwm.c

main.c

Este archivo es el primero en iniciarse y a partir de él se ejecutan los demás archivos y funciones necesarias para el arranque del sistema, como pueden ser la inicialización de las memorias de la placa de control, las configuraciones iniciales de los periféricos y contadores del sistema, y las interrupciones [5].

En la Figura 42 se muestra un fragmento del código de este archivo.

```

58 // INFINITE LOOP
59 //=====
60 for(;;) {
61 // GRAFCET
62 switch(Stage){
63 case(init):
64     if (start_up == off) {
65         PwMtoIO_VSC1();
66
67         RELAY_1_OFF; // MAIN RELAY OPENED
68         RELAY_2_OFF; // PRECHARGE RELAY OPENED
69         RELAY_DISCHARGE_OFF; // DISCHARGE RELAY CLOSED
70         RELAY_FAN_OFF; // STOP FANs
71
72         State_VSC1.counter=stop_counter;
73         State_VSC1.counter_pre_dis=0;
74     } else if (start_up == on) {
75         State_VSC1.counter=run_counter;
76
77         if (State_VSC1.counter_pre_dis>=2000){
78             Stage=precharge1;
79             State_VSC1.go_on=0;
80             State_VSC1.counter_pre_dis=0;
81             State_VSC1.counter=run_counter;
82         }
83     }
84     break;//init
85

```

Figura 42. Fragmento del código del archivo main.c estado init

CONMUTE_3n1_structures.h, CONMUTE_3n1.h y extra_funtions

El sistema cuenta con una serie de estados que, en caso de querer añadir, eliminar o modificar estos estados se deberá acudir a las librerías "CONMUTE_3n1_structures.h", "CONMUTE_3n1.h" y "Extra_funtions.c". Un fragmento de estas librerías puede verse en las Figuras 43, 44, 45 y 46 .

```

21 // GRAFCET
22 //=====
23     modes_etapes      Stage           = init;
24     estat             State_VSC1     = {0, 0, stop_counter, inverter};
25     reset             Reset          = {0, 0};
26     Compare3F_struct COMPARE         = COMPARE3F_DEFAULTS;
27     start_up_mode     start_up       = off;
28     Estado            Sincronizacion = Desactivada;
29     modes_operation   operation      = inverter;
30     modo_PLL          PLL            = Encendida;
31 //=====

```

Figura 43. Fragmento del código del archivo CONMUTE_3n1.h

```

29 // Grafcet
30 //=====
31     extern modes_etapes      Stage;
32     extern estat             State_VSC1;
33     extern reset             Reset;
34     extern Compare3F_struct COMPARE;
35     extern start_up_mode     start_up;
36     extern Estado            Sincronizacion;
37     extern modo_PLL          PLL;
38 //=====

```

Figura 44. Fragmento del código del archivo extra_funtions.h

```

1 #include "COMUTE_3n1_defines.h"
2
3 // Enum variables
4 //=====
5 typedef enum {init, precharge1, precharge2, run, discharge, alarm} modes_etapes;
6 typedef enum {stop_counter, run_counter} modes_counter;
7 typedef enum {inverter, rectifier, interleaved, open_loop} modes_operation;
8 typedef enum {off, on} start_up_mode;
9 typedef enum {Activada, Desactivada} Estado;
10 typedef enum {Encendida, Apagada} modo_PLL;
11 //=====

```

Figura 45. Fragmento del código del archivo CONMUTE_3n1_structures.h

Dentro de la librería `extra_funtions` también se encuentra la función para detectar algún tipo de alarmas.

```

96 #if FLASH
97 #pragma CODE_SECTION(Calc_Alarms_VSC1, "ramfuncs") // This function is located in RAM
98 #endif
99
100 void Calc_Alarms_VSC1(void) {
101     if (_IQabs(Iu_VSC1.Valor) > _IQ(I_MAX_FASE) || _IQabs(Iv_VSC1.Valor) > _IQ(I_MAX_FASE) || _IQabs(Iw_VSC1.Valor) > _IQ(I_MAX_FASE)){
102         alarm_VSC1.bit.Iac_max=1;
103     }
104
105     if (_IQabs(Vu_VSC1.Valor) > _IQ(U_MAX_FASE) || _IQabs(Vv_VSC1.Valor) > _IQ(U_MAX_FASE) || _IQabs(Vw_VSC1.Valor) > _IQ(U_MAX_FASE)){
106         alarm_VSC1.bit.Ux_max=1;
107     }
108
109     if (_IQabs(Ibus_VSC1.Valor) > _IQ(I_MAX_DC)) alarm_VSC1.bit.Idc_max = 1;
110
111     if (Vbus_VSC1.Valor > _IQ(U_BUS_OV)) alarm_VSC1.bit.Udc_OV=1;
112     if (Vbus_VSC1.Valor < _IQ(U_BUS_UV)) alarm_VSC1.bit.Udc_UV=1;
113
114     if (T_VSC1.Valor > _IQ(T_MAX)) alarm_VSC1.bit.Temp=1;
115
116     if (!GpioDataRegs.GPADAT.bit.GPIO30) alarm_VSC1.bit.TripDrivers = 1;
117 }
118 }

```

Figura 46. Fragmento del código del archivo `extra_funtions.h` función alarmas

CONMUTE_3n_1_defines.h

En esta librería se definen los límites de protección del sistema, también se definen los relés del convertidor. Un fragmento de esta librería puede verse en la Figura 47.

```

1 // Contactors
2 //=====
3 #define RELAY_1_OFF          GpioDataRegs.GPBCLR.bit.GPIO50   = 1 //D_out.VSC1.Rel1
4 #define RELAY_1_ON          GpioDataRegs.GPBSET.bit.GPIO50   = 1 //D_out.VSC1.Rel1
5 #define RELAY_2_OFF          GpioDataRegs.GPBCLR.bit.GPIO53   = 1 //D_out.VSC1.Rel2
6 #define RELAY_2_ON          GpioDataRegs.GPBSET.bit.GPIO53   = 1 //D_out.VSC1.Rel2
7 #define RELAY_DISCHARGE_OFF GpioDataRegs.GPCCLR.bit.GPIO71   = 1 //D_out.VSC1.Disch
8 #define RELAY_DISCHARGE_ON  GpioDataRegs.GPCSET.bit.GPIO71   = 1 //D_out.VSC1.Disch
9 #define RELAY_FAN_OFF       GpioG2DataRegs.GPECLEAR.bit.GPIO131 = 1 //D_out.VSC1.Fan
10 #define RELAY_FAN_ON        GpioG2DataRegs.GPESET.bit.GPIO131 = 1 //D_out.VSC1.Fan
11 //=====
12 #define RELAY_3_ON          GpioDataRegs.GPBSET.bit.GPIO61   = 1 //D_out.VSC1.Rel3 PRUEBA
13 #define RELAY_3_OFF        GpioDataRegs.GPBCLR.bit.GPIO61   = 1 //D_out.VSC1.Rel3 PRUEBA
14
15 #define RELAY_4_ON          GpioDataRegs.GPBSET.bit.GPIO62   = 1 //D_out.VSC1.Rel4 PRUEBA
16 #define RELAY_4_OFF        GpioDataRegs.GPBCLR.bit.GPIO62   = 1 //D_out.VSC1.Rel4 PRUEBA
17
18
19 // Alarms
20 //=====
21 #define U_BUS_OV            300.0
22 #define U_BUS_UV            80.0 //Lo he modificado antes ponía 100 V
23 #define U_BUS_PRE           190.0 //He modificado este valor el anterior era 520 V
24 #define I_MAX_FASE          5.0
25 #define I_MAX_DC            25.0
26 #define U_MAX_FASE          625.0
27 #define T_MAX               90.0
28 //=====

```

Figura 47. Fragmento del código del archivo `CONMUTE_3n1_defines.h`

F28M35x_EPwm.c

Este archivo es importante debido a que en él se cambiará la frecuencia de la interrupción principal del programa, donde irá el control del convertidor y conjuntamente también se cambiará la frecuencia de conmutación del sistema.

Modificando el registro EPwm"X".Regs.TBPRD, siendo "X" el número de cada módulo PWM. En la Figura 48 puede observarse un fragmento de este archivo.

```

25 // Configure the pins as peripheral function is the first step
26 InitEPwm1Gpio();
27 InitEPwm2Gpio();
28 InitEPwm3Gpio();
29
30 // Activate the peripheral clock is the second step
31 EALLOW;
32 SysCtrlRegs.PCLKCR1.bit.EPwm1ENCLK = 1; // ePWM1
33 SysCtrlRegs.PCLKCR1.bit.EPwm2ENCLK = 1; // ePWM2
34 SysCtrlRegs.PCLKCR1.bit.EPwm3ENCLK = 1; // ePWM3
35 EDIS;
36
37 // EPwm1 peripheral configuration
38 EPwm1Regs.TBPRD = 7500; // Timer HR period is 0
39 EPwm1Regs.TBPRDHR = 0; // Time difference is 0
40 EPwm1Regs.TBPHS_half.TBPHS = 0x0000; // Counter reset
41 EPwm1Regs.TBCTR = 0;
42
43 EPwm1Regs.TBCTL.bit.FREE_SOFT=2;
44 EPwm1Regs.TBCTL.bit.PHSDIR = 0;
45 EPwm1Regs.TBCTL.bit.CLKDIV = 0; // TBCLK = 150 MHz --> Divided by 1
46 EPwm1Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1; // Divided by 1
47 EPwm1Regs.TBCTL.bit.SYNCSEL = TB_CTR_ZERO; // Synchronisation when CTR = 0
48 EPwm1Regs.TBCTL.bit.PRDL = 0;
49 EPwm1Regs.TBCTL.bit.PHSEN = TB_DISABLE; // EPwm1 is the master to synchronise
50 EPwm1Regs.TBCTL.bit.CTRMODE = TB_COUNT_UPDOWN;

```

Figura 48. Fragmento del código del archivo F28M35x_EPwm.c

El valor que se pondrá viene determinado por la frecuencia de conmutación que se desee de acuerdo con la ecuación (50).

$$EPwmXRegs.TBPRD = \frac{\text{Frecuencia reloj}}{2 \cdot \text{Frecuencia conmutación}} = \frac{150 \text{ MHz}}{2 \cdot 10 \text{ kHz}} = 7500 \quad (50)$$

Esto generará en los módulos PWM una triangular con un valor pico de 7500 y con una frecuencia de 10 kHz, a su vez también hará que la interrupción principal se ejecute con una frecuencia de 10 kHz.

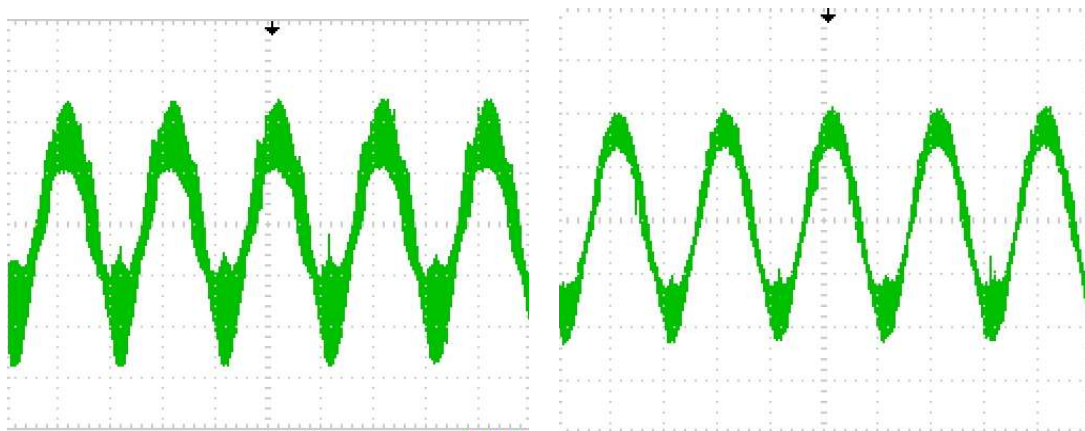


Figura 49. Comparativa corriente por la resistencia con distinta frecuencia de conmutación: izquierda 5 kHz y derecha 10 kHz

En la Figura 49 se puede ver una comparativa ante la misma corriente con diferentes frecuencias de conmutación y como es de esperar el rizado de la corriente con una frecuencia de conmutación de 10 kHz es menor que con una frecuencia de conmutación de 5 kHz.

Interrupt_TZs.c

Esta interrupción saltará en caso de que fallen los drivers de los IGBT's, llevando al sistema al estado de alarma.

Interrupt_ADC

Esta será la interrupción principal del sistema y en ella se implementará el control del convertidor. A continuación, se explicará con un mayor detalle.

5.2 PROGRAMACIÓN DEL CONTROL DE CORRIENTE

En este apartado se describirá el código de control programado en la interrupción principal de la placa de control, esta interrupción ha sido configurada e inicializada en el archivo *main.c* explicado anteriormente.

5.2.1 Diagrama de flujo

En la Figura 50 se muestra el diagrama de flujo de la interrupción principal del sistema donde se programará el control del convertidor, y que se ejecutará cada 100 us.

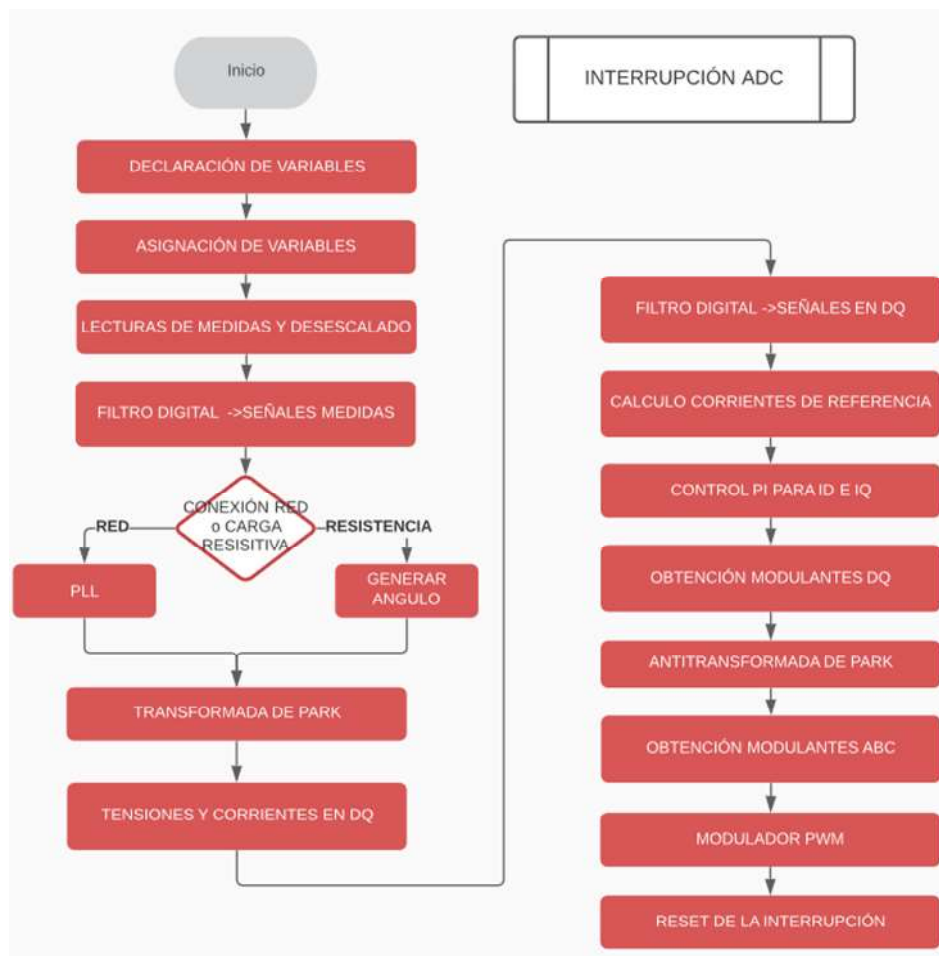


Figura 50. Diagrama de Flujo de la Interrupción principal

5.2.2 Descripción del código de control

Declaración de variables

En este apartado se añaden las librerías que podrán ser utilizadas en la interrupción además del tipo de variables y los nombres de estas. En particular la mayoría de las variables utilizadas son del tipo “double” que son variables de 8 Bytes. En la Figura 51 se puede observar la declaración de variables en la interrupción ADC.

```

53 //VARIABLES TENSIONES UVM Y DQ
54 double Vu_med, Vv_med, Vw_med, Vbus_med, Valfa_med_f, Vbeta_med_f, Vd_med_f, angulo_Vred, Vq_med_f;
55
56 //VARIABLES CORRIENTES UVM Y DQ
57 double Iu_med, Iv_med, Iw_med, Ibus_med, Ialfa_med_f, Ibeta_med_f, angulo_Ired, Id_med_f, Iq_med_f;
58
59 //VARIABLES CONTROL DE CORRIENTE
60 double Iqref, Idref, Idqref, I_Ref_A_RMS;
61 double Kp, Tn, Ki, Ksv, Ksv_DC, Ksi, Ksi_total;
62 double error_Iq, error_Id, Ierror_Iq=0, Ierror_Id=0, Ierror_Iq_ant=0, Ierror_Id_ant=0;
63 double VLref_q, VLref_d, Vinvref_q, Vinvref_d, Idqmax, Vinvref_dq;
64 double Vcontrol, Vptriangular, indicemodulacion, indicemodulacion_real;
65
66 //PARÁMETROS DEL SISTEMA
67 double L=0.0022, w=314.1593;
68
69 //VARIABLES MODULANTES
70 double md, mq, mdq, ma, mb, mc;
71
72 //VARIABLES GENERAR ANGULO Qdq PARA REALIZAR LA TRANSFORMADA DE DQ
73 double fdq=50.0, w_0, I_Odq, Odq, I_Odq_ant;
74
75
76 //VARIABLES CONSTANTES FILTRO DIGITAL
77 double a0, a1, b0, b1;
78
79
80 //VARIABLES CORRIENTES FILTRO DIGITAL
81 double entrada_q, entrada_ant_q, salida_ant_1_q, salida_ant_2_q, Iq_med_f_digital, Iq_med_f_digital_real; //Corriente Eje q
82 double entrada_d, entrada_ant_d, salida_ant_1_d, salida_ant_2_d, Id_med_f_digital, Id_med_f_digital_real; //Corriente Eje d
  
```

Figura 51. Fragmento del código del archivo Interrup_ADC.c, declaración de variables

Asignación de variables

Debido a que existen multitud de parámetros constantes, como por ejemplo los factores de escala, los parámetros de los controladores y los parámetros del filtro digital estos deben ir asociados con su variable correspondiente. En la Figura 52 se puede observar la asignación de estos parámetros con sus variables en la interrupción ADC.


```

172 //Asignacion variables Lazo de corriente
173 Ksi=1.0;
174 Ksi_total=1.0;
175 Ksv=1.0;
176
177 Kp=0.52;
178 Tn=0.0000078;
179 Ki=Kp/Tn;
180
181 //Para una frecuencia de conmutacion de 10KHz Vptriangular=7500
182 Vptriangular=7500.0;
183
184 //Asignación variables filtro digital para Fmuestreo=10KHz
185 a0=1.0;
186 a1=-0.2283;
187 b0=0.3859;
188 b1=0.3859;
189
190 //Asignación Variables PLL Fmuestreo=10KHz
191 Kp_PLL=54.71;
192 Tn_PLL=0.0282;
193 Ki_PLL=Kp_PLL/Tn_PLL;
194 wn=2*pi*50.0;
195
196
197 //Factores de escala
198 FC_V=0.0000023983;
199 FC_I=0.0000014;
200 FC_Vbus=0.00000194;
201 FC_Ibus=0.000001816;
  
```

Figura 52. Fragmento del código del archivo Interrup_ADC.c, asignación de variables

Lectura de medidas y desescalado

Una vez se han leído las medidas provenientes de los sensores y se lanza la conversión AD, dichas medidas son desescaladas, a través de los factores de escala, para trabajar con su valor real. En la Figura 53 se puede observar la lectura de medidas en la interrupción ADC.

```

140 // ADCs CHANNELS READ =====//
141 // VSC1
142 Vbus_VSC1.calc(&Vbus_VSC1); // DC bus voltage
143 Ibus_VSC1.calc(&Ibus_VSC1); // DC bus current
144 Vu_VSC1.calc(&Vu_VSC1); // Voltage U
145 Vv_VSC1.calc(&Vv_VSC1); // Voltage V
146 Vw_VSC1.calc(&Vw_VSC1); // Voltage W
147 Iu_VSC1.calc(&Iu_VSC1); // Current U
148 Iv_VSC1.calc(&Iv_VSC1); // Current V
149 Iw_VSC1.calc(&Iw_VSC1); // Current W
150 T_VSC1.calc(&T_VSC1); // Semiconductors temperature
151
  
```

Figura 53. Fragmento del código del archivo Interrup_ADC.c, lectura de medidas

Como consecuencia de las diferentes configuraciones internas de los sensores se tienen distintos fondos de escala. Para un cálculo correcto de los fondos de escala se necesita saber el rango que puede medir el convertidor, el rango correspondiente en la placa, la ganancia del sensor y el número de bits del sistema para su representación. El rango de medidas de la tensión de bus es de 0-930.81 V, el rango correspondiente en la placa de control es de 0-10 V, la ganancia $1.07433 \cdot 10^{-2}$ V/V. El fondo de escala se obtiene según la ecuación (51).

$$FE_{V_{BUS DC}} = \frac{\text{Rango Convertidor}}{\text{Rango Placa} \cdot \text{Ganancia} \cdot 2^{\text{bits representación}}} \quad (51)$$

$$= \frac{930.81}{10 \cdot 2^{32} \cdot 1.074330 \cdot 10^{-2}} = 2.017 \cdot 10^{-6}$$

Si el rango de medidas abarca medidas negativas el número de bits de representación será de 31 bits. Para obtener los demás factores de escala se sigue un procedimiento análogo. En la Tabla 7 se muestran los factores de escala utilizados.

VARIABLE	FONDO ESCALA
Tensión fase U/V/W	$3.22 \cdot 10^{-6}$
Corriente fase U/V/W	$1.41 \cdot 10^{-6}$
Tensión de BUS DC	$2.02 \cdot 10^{-6}$
Corriente BUS DC	$1.82 \cdot 10^{-6}$

Tabla 7. Factores de escala

En la Figura 54 se puede observar los factores de escala y su aplicación en la interrupción ADC.

```

217 //Factores de escala
218 FC_V=0.0000023983;
219 FC_I=0.0000014;
220 FC_Vbus=0.00000194;
221 FC_Ibus=0.000001816;
222
223
224 //LECTURA DE LAS VARIABLES
225 //Tensiones
226 Vu_med=Vu_VSC1.Valor*FC_V;
227 Vv_med=Vv_VSC1.Valor*FC_V;
228 Vw_med=Vw_VSC1.Valor*FC_V;
229
230 Vbus_med=Vbus_VSC1.Valor*FC_Vbus;
231 Vred_RMS=230.0;
232 Vred_PICO=Vred_RMS*sqrt(2.0);
233
234 //Corrientes
235 Iu_med=Iu_VSC1.Valor*FC_I;
236 Iv_med=Iv_VSC1.Valor*FC_I;
237 Iw_med=Iw_VSC1.Valor*FC_I;
238
239 Ibus_med=Ibus_VSC1.Valor*FC_Ibus;
240
  
```

Figura 54. Fragmento del código del archivo Interrup_ADC.c, factores de escala

Alarma

Tras obtener las medidas y desescalarlas se lanza la rutina de alarma que comparará los valores medidos con los límites de protección definidos en la librería *CONMUTE_3n1_defines*, Figura 55, y en caso de detectar que

alguno no se encuentra dentro de los límites establecidos mandará al inversor al estado de alarma. En la Figura 56 se puede observar la comprobación de alarmas en la interrupción ADC.

```

241 // ALARM MANAGEMENT =====//
242 if (!GpioDataRegs.GPADAT.bit.GPIO19) alarm_VSC1.bit.ParadaEmergencia = 1;
243 if (Stage == run) {
244     Calc_Alarms_VSC1();
245 }
246
247 // Any hardware alarm sends the grafcet to alarm state
248 if (alarm_VSC1.all != 0 && Stage != alarm){
249     Stage = alarm;
250     PWMtoIO_VSC1();
251     State_VSC1.counter = stop_counter;
252     State_VSC1.counter_pre_dis = 0;
253     State_VSC1.counter = run_counter;
254 } else {
255     // Nothing else
256 }
257

```

Figura 55. Fragmento del código del archivo Interrup_ADC.c, rutina de alarmas

Filtro Digital

El sistema no dispone de filtros en los sensores de medida por ello ha sido necesario implementar un filtro digital en el DSP los parámetros del filtro, diseñado con una frecuencia de cortes de 2 kHz, son las mostrados en la Tabla 8.

PARÁMETRO	VALOR
a ₀	1
a ₁	-0.2283
b ₀	0.3859
b ₁	0.3859

Tabla 8. Parámetros filtro digital

Resultando la implementación en el DSP conforme la ecuación (52).

$$y(n) = (b_0 \cdot e(n) + b_1 \cdot e(n - 1)) - a_1 \cdot y(n - 1) \quad (52)$$

Siendo e la entrada y n el número de muestra.

En la Figura 56 se puede observar la implementación del filtro digital en la interrupción ADC.

```

275 //FILTRO DIGITAL CORRIENTES MEDIDAS
276
277 //Corriente Fase U
278 entrada_ant_Iu=entrada_Iu;
279 entrada_Iu=Iu_med;
280 salida_ant_1_Iu=Iu_med_f;
281
282 Iu_med_f=((b0*entrada_Iu)+(b1*entrada_ant_Iu))-(a1*salida_ant_1_Iu);
283
284 //Corriente Fase V
285 entrada_ant_Iv=entrada_Iv;
286 entrada_Iv=Iv_med;
287 salida_ant_1_Iv=Iv_med_f;
288
289 Iv_med_f=((b0*entrada_Iv)+(b1*entrada_ant_Iv))-(a1*salida_ant_1_Iv);
290
291 //Corriente Fase W
292 entrada_ant_Iw=entrada_Iw;
293 entrada_Iw=Iw_med;
294 salida_ant_1_Iw=Iw_med_f;
295
296 Iw_med_f=((b0*entrada_Iw)+(b1*entrada_ant_Iw))-(a1*salida_ant_1_Iw);
297

```

Figura 56. Fragmento del código del archivo Interrup_ADC.c, filtro digital

Obtención del ángulo para generar las transformadas dq

Para la realización de la transformada de Park o dq se necesita un ángulo que determinará la frecuencia a la que se obtendrán las modulantes, y con ellas la tensiones y corrientes de salida del inversor.

Esta parte está implementada de dos maneras distintas. Por un lado, dicho ángulo se genera con la propia interrupción, la otra manera es a través de la PLL.

En la implementación real del inversor conectado a una carga resistiva se necesitará obtener el ángulo de la primera manera mencionada, mientras que la conexión futura del convertidor a la red se necesitará obtener este ángulo a través de la segunda manera.

o Generar ángulo

Aquí se fija una determinada frecuencia, que será la que se quiera obtener a la salida, tras integrarla y limitarla entre $-\frac{\pi}{2}$ y $\frac{\pi}{2}$ se obtiene el ángulo. Este será el ángulo utilizado para la transformada dq y en la antitransformada, para obtener las modulantes abc. En la Figura 57 se puede observar cómo se genera el ángulo para la transformada dq en la interrupción ADC.

```

335 //HALLAR ANGULO PARA REALIZAR LA TRANSFORMADA DE DQ
336
337     w_0=2.0*pi*fdq;
338     I_Odq=I_Odq_ant+(w_0*Tm);
339
340     //Actualizar integral
341     I_Odq_ant=I_Odq;
342
343     //Angulo
344     Odq=I_Odq;
345
346     //Limitar el angulo entre pi y -pi
347     contador_angulo++;
348     if(Odq>=pi)
349     {
350         Odq=Odq-(2.0*pi);
351         I_Odq_ant=Odq;
352     }
353
354     if(Odq<=-2*pi)
355     {
356         Odq=Odq;
357         I_Odq_ant=Odq;
358     }
359     contador_Odq++;
360

```

Figura 57. Fragmento del código del archivo Interrup_ADC.c, ángulo para transformada dq

La integral se hará considerando el método “Forward Euler”, que consiste en obtener la integral a partir de la muestra anterior de dicha integral y sumarle el periodo de muestreo por la muestra actual de lo que se desea integrar como puede observarse en la ecuación (53).

$$Int(k) = Int(k - 1) + T_m \varepsilon(k) \quad (53)$$

Siendo e la entrada y n el número de muestra.

- PLL

Para este procedimiento se requiere conocer las tensiones de red y el procedimiento es el explicado en el apartado de la PLL visto anteriormente. En la Figura 58 se puede observar cómo se implementa la PLL en la interrupción ADC.

```

362 //ENCONTRAR ANGULO A TRAVES DE LA PLL
363
364     Vdq_PLL=sqrt((Vq_med_f*Vq_med_f)+(Vd_med_f*Vd_med_f));
365     Vq_PLL_pu=Vq_med_f/Vdq_PLL;
366
367     //PI de la PLL
368
369     I_Vq_PLL=I_Vq_PLL_ant+(Vq_PLL_pu*Tm); //Integral del error metodo Forward Euler
370
371     //Actualizar el valor de la integral
372     I_Vq_PLL_ant=I_Vq_PLL;
373
374     w_PI=(Kp_PLL*Vq_PLL_pu)+(Ki_PLL*I_Vq_PLL);
375
376     w_PLL=w_PI+wn;
377
378     I_w_PLL=I_w_PLL_ant+(w_PLL*Tm);
379
380     //Actualizar el valor de la integral
381     I_w_PLL_ant=I_w_PLL;
382
383     //Angulo
384     O_PLL=I_w_PLL;
385
386     //Limitar el angulo entre pi y -pi
387     if(O_PLL>=pi)
388     {
389         O_PLL=O_PLL-(2.0*pi);
390         I_w_PLL_ant=O_PLL;
391     }
392     if(O_PLL<=-pi)
393     {
394         O_PLL=O_PLL;
395         I_w_PLL_ant=O_PLL;
396     }
397

```

Figura 58. Fragmento del código del archivo Interrup_ADC.c, ángulo para transformada dq PLL

Tensiones y Corrientes dq

Con el ángulo obtenido anteriormente, aplicando la transformada de Park al sistema trifásico de tensiones y corrientes, se consiguen las tensiones en dq medidas y filtradas.

Una vez se tienen estas tensiones y corrientes en los ejes dq se hacen pasar por un nuevo filtro digital con la misma frecuencia de corte que el anterior, aunque se ha implementado así este paso no es necesario. En la Figura 59 se puede observar cómo se obtienen las tensiones y corrientes en dq en la interrupción ADC.

```

419 //TENSIONES DQ PARA LA TENSIÓN DE RED
420
421 //Angulo para hacer la transformada dq
422 angulo_Vred=0_PLL;
423
424 //Calculos Convenio Europeo hallar las componentes D y Q
425
426 Vd_med_f=sqrt(2.0/3.0)*(Vu_med_f*cos(angulo_Vred)+Vv_med_f*cos(angulo_Vred-(2.0*pi/3.0))+Vw_med_f*cos(angulo_Vred+(2.0*pi/3.0)));
427 Vq_med_f=sqrt(2.0/3.0)*(-Vu_med_f*sin(angulo_Vred)-Vv_med_f*sin(angulo_Vred-(2.0*pi/3.0))-Vw_med_f*sin(angulo_Vred+(2.0*pi/3.0)));
428
429 //HALLAR CORRIENTES DQ
430
431 //Angulo para hacer la transformada dq
432 angulo_Ired=0dq;
433
434 //Calculos Convenio Europeo hallar las componentes D y Q
435
436 Id_med_f=sqrt(2.0/3.0)*(Iu_med_f*cos(angulo_Ired)+Iv_med_f*cos(angulo_Ired-(2.0*pi/3.0))+Iw_med_f*cos(angulo_Ired+(2.0*pi/3.0)));
437 Iq_med_f=sqrt(2.0/3.0)*(-Iu_med_f*sin(angulo_Ired)-Iv_med_f*sin(angulo_Ired-(2.0*pi/3.0))-Iw_med_f*sin(angulo_Ired+(2.0*pi/3.0)));
438
439 //FILTRO DIGITAL EJE DQ
440
441 //Corriente Eje q
442 entrada_ant_q=entrada_q;
443 entrada_q=Iq_med_f;
444 salida_ant_1_q=Iq_med_f_digital;
445
446 Iq_med_f_digital=((b0*entrada_q)+(b1*entrada_ant_q)-(a1*salida_ant_1_q);
447
448
  
```

Figura 59. Fragmento del código del archivo Interrup_ADC.c, tensiones y corrientes dq

Lazo de corriente

o Modulantes dq

Una vez se tienen las corrientes de referencia el sistema entra propiamente en el lazo de control calculando primeramente el error, después la integral de este para después aplicar el control PI y obtener la tensión de referencia que debe aplicar el inversor en cada uno de los ejes.

Con esta tensión se procede al cálculo de las modulantes en los ejes dq, es importante que en este cálculo que se tenga en cuenta el valor pico de la triangular.

A través de las ecuaciones (54) y (55) se calculan las modulantes:

$$m_d = \frac{\text{Tensión referencia inversor } d}{\frac{V_{BUS}}{2 \cdot V_{pico \text{ triangular}}}} \quad (54)$$

$$m_q = \frac{\text{Tensión referencia inversor } q}{\frac{V_{BUS}}{2 \cdot V_{pico \text{ triangular}}}} \quad (55)$$

En la Figura 60 se puede observar cómo se obtienen las modulantes en dq en la interrupción ADC.

```
//LAZO DE CONTROL DE CORRIENTE

//Eje q
error_Iq=Iqref-Iq_med_f_digital; //Error
Ierror_Iq=(Ierror_Iq_ant+(error_Iq*Tm)); //Integral del error metodo Forward Euler
Vlref_q=((Kp*error_Iq)+(Ki*Ierror_Iq)); //Tras aplicar el PI se halla la tensión referencia de la bobina en el eje q
Vinvref_q=(Vlref_q-(Id_med_f_digital*L*w/Ksi_total)); // Cálculo de la tensión de referencia a aplicar por el inversor en el eje
mq=Vinvref_q/(Vbus_med_f/(2.0*Vptriangular)); // Señal de modulación en q

//Actualizar el valor de la integral
Ierror_Iq_ant=Ierror_Iq;

//Eje d
error_Id=Idref-Id_med_f_digital;
Ierror_Id=(Ierror_Id_ant+(error_Id*Tm));
Vlref_d=(Kp*error_Id)+(Ki*Ierror_Id);
Vinvref_d=(Vlref_d+(Iq_med_f_digital*L*w/Ksi_total));
md=Vinvref_d/(Vbus_med_f/(2.0*Vptriangular));

//Actualizar el valor de la integral
Ierror_Id_ant=Ierror_Id;
```

Figura 60. Fragmento del código del archivo Interrup_ADC.c, modulantes dq

o Modulantes abc

Una vez se tienen las modulantes en dq es necesario realizar la antitrasformada para obtener las modulantes senoidales en abc que después se mandarán al módulo PWM para que sean comparadas con la triangular y den las ordenes de conmutación en los IGBT's. En la Figura 61 se puede observar cómo se obtienen las modulantes abc en la interrupción ADC.

```
607 //MODULANTES
608
609 //Antitrasformada para obtener las tensiones de control senoidales
610 indicemodulacion_real=indicemodulacion/Vptriangular;
611
612 ma=sqrt(2.0/3.0)*(cos(Odq)*md-sin(Odq)*mq);
613 mb=sqrt(2.0/3.0)*(cos(Odq-2.0*pi/3.0)*md-sin(Odq-2.0*pi/3.0)*mq);
614 mc=sqrt(2.0/3.0)*(cos(Odq+2.0*pi/3.0)*md-sin(Odq+2.0*pi/3.0)*mq);
615
```

Figura 61. Fragmento del código del archivo Interrup_ADC.c, modulantes ma, mb y mc

o Módulo PWM

En la parte final de la interrupción se envían las modulantes al módulo PWM a través del registro "EPwm1Regs.CMPA.HALF.CMPA" de cada módulo, como se mencionaba anteriormente el módulo genera una señal triangular no centrada en cero y que para una frecuencia de conmutación de 10 kHz tiene un valor pico de 7500. Como las modulantes obtenidas se tienen centradas en cero, antes de enviarlas al módulo PWM, se deben centrar en el valor medio de la triangular. En la Figura 62 se puede observar cómo se mandan modulantes abc a los módulos PWM en la interrupción ADC.

```
//MODULO PWM DEL MICRO
if(Stage == run){
    EPwm1Regs.CMPA.half.CMPA= ma+(Vptriangular/2.0); //Se centra la modulante en el valor medio de la triangular
    EPwm2Regs.CMPA.half.CMPA= mb+(Vptriangular/2.0);
    EPwm3Regs.CMPA.half.CMPA= mc+(Vptriangular/2.0);
}
```

Figura 62. Fragmento del código del archivo Interrup_ADC.c, módulo PWM

○ *Reset de la interrupción*

Para finalizar la propia interrupción esta debe ser reseteada a fin de que pueda volver a activarse. En la Figura 63 se puede observar cómo se resetea la interrupción ADC.

```
// RESET INTERRUPCIÓN =====//
Adc1Regs.ADCINTFLGCLR.bit.ADCINT1 = 1; // Clear ADCINT1 flag reinitialize for next SOC
Adc2Regs.ADCINTFLGCLR.bit.ADCINT1 = 1; // Clear ADCINT1 flag reinitialize for next SOC
PieCtrlRegs.PIEACK.all = PIEACK_GROUP1; // Acknowledge interrupt to PIE
return;
}
```

Figura 63. Fragmento del código del archivo Interrup_ADC.c, reset de la interrupción

5.3 ESQUEMA GENERAL DEL CÓDIGO

En el diagrama de flujo de la Figura 64 se representa la inicialización del programa a través del módulo principal *main.c*, una vez este sea ejecutado se irán ejecutando diferentes funciones para la inicialización y control del inversor [5].

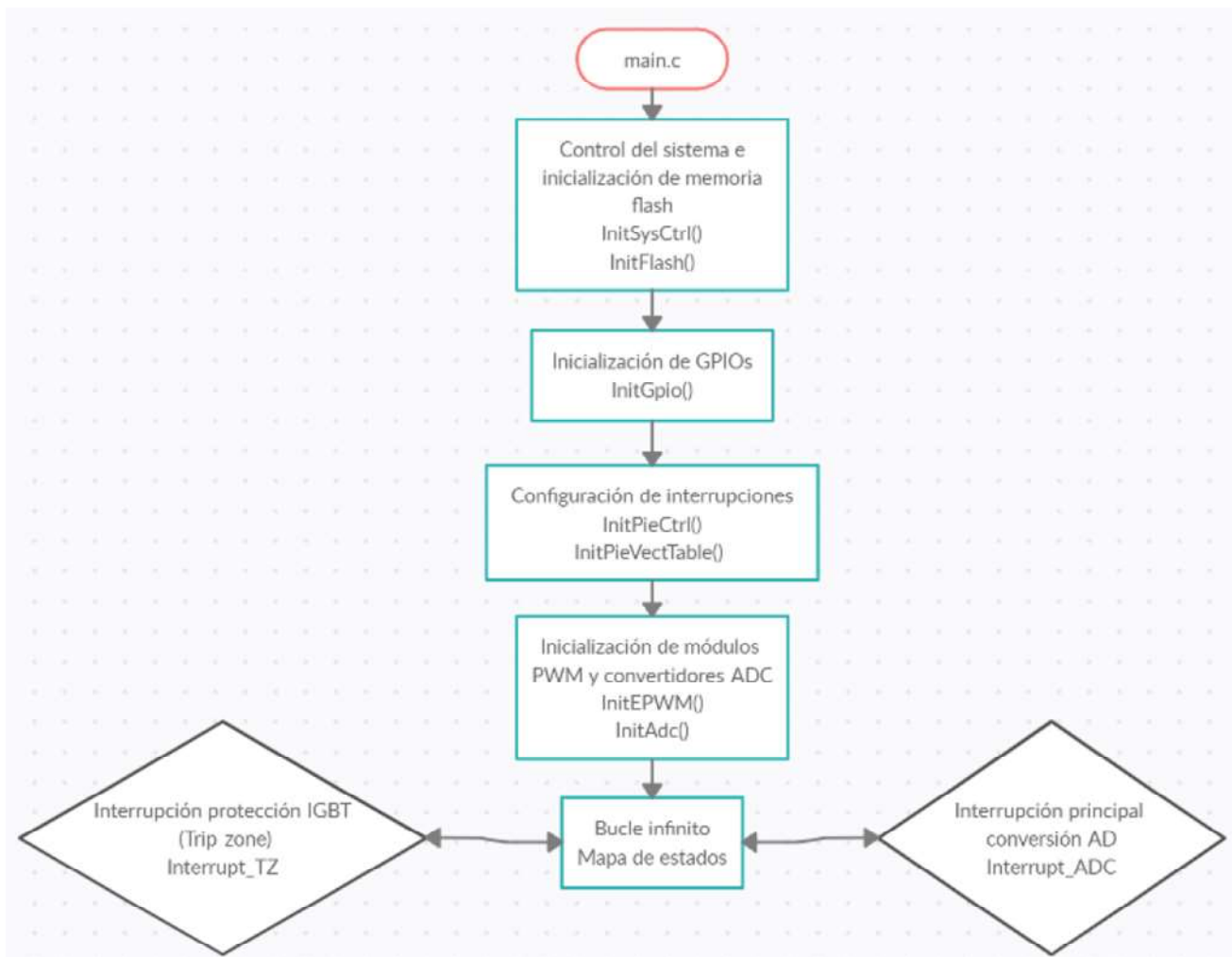


Figura 64. Diagrama de flujo de la estructura principal del código [5]

5.3.1 Mapa de estados

Los estados en los que se puede encontrar el inversor son los siguientes:

- **Init:** Estado inicial del inversor justo después de ejecutar el código. En el él inversor tendrá todos los interruptores abiertos hasta que la variable de puesta en marcha “start_up” pase al estado “on”
- **Precarga 1 y 2:** En estos estados el inversor realiza la precarga del bus DC, conectando en el estado de Precarga 1 la fuente DC al bus DC a través de una resistencia, una vez el bus DC haya alcanzado un porcentaje significativo de su tensión final se pasará al estado de Precarga 2, en el que se conectará la fuente DC directamente al bus DC a través de otro contactor.

En este proyecto se han obviado los estados de precarga 1 y 2 al contar con una fuente DC que puede limitarse en corriente, si en el futuro se desea utilizar una fuente no limitada o trabajar con mayores tensiones se deberán modificar los tiempos de ambos estados de precarga para poder equilibrar las tensiones y evitar cortocircuitos

- **Run:** Estado de funcionamiento del convertidor. En este estado se da la conmutación de los IGBT's.
- **Alarm:** Estado en el que se detiene la conmutación de los IGBT's, el inversor entrará en este estado si se detecta alguna alarma.

6. VALIDACIÓN EXPERIMENTAL

Los modelos anteriores han sido capaces de permitir analizar la respuesta dinámica del sistema, además tras diseñar los lazos de control, han permitido que estos puedan ser programados tanto en PSIM como en el prototipo real disponible en el laboratorio. En el siguiente apartado se analizarán los resultados obtenidos en la simulación de PSIM para después comparar dichos resultados con los obtenidos con el prototipo.

La validación experimental únicamente se realizará de la conexión del inversor a la carga resistiva a través del filtro LCL.

6.1 LIMITACIONES DEL PROTOTIPO

Con respecto al prototipo, en el que el código es programado a través del software Code Composer en el DSP, presenta ciertas limitaciones. Por un lado, a pesar de que la interfaz gráfica Code Composer permite visualizar las variables utilizadas en el código, la velocidad de adquisición de los datos es lenta, debido al conector JTAG empleado para la comunicación entre el ordenador y el DSP. Esto hace que no se puedan visualizar correctamente señales que varíen con una frecuencia superior a 0.01 Hz. Si a esta limitación se le añade que no se ha podido hacer uso de las salidas analógicas del DSP el rango de señales que se pueden medir en el osciloscopio queda delimitado. Este conjunto de limitaciones hace que sea muy complicado comparar, por ejemplo, las referencias de corriente y las corrientes del sistema imposibilitando el análisis de los transitorios, es por ello por lo que solo se analizará el estado estacionario de este.

6.2 RESULTADOS DE LA SIMULACIÓN

En la Figura 65 se puede observar el esquema completo del sistema utilizado en PSIM.

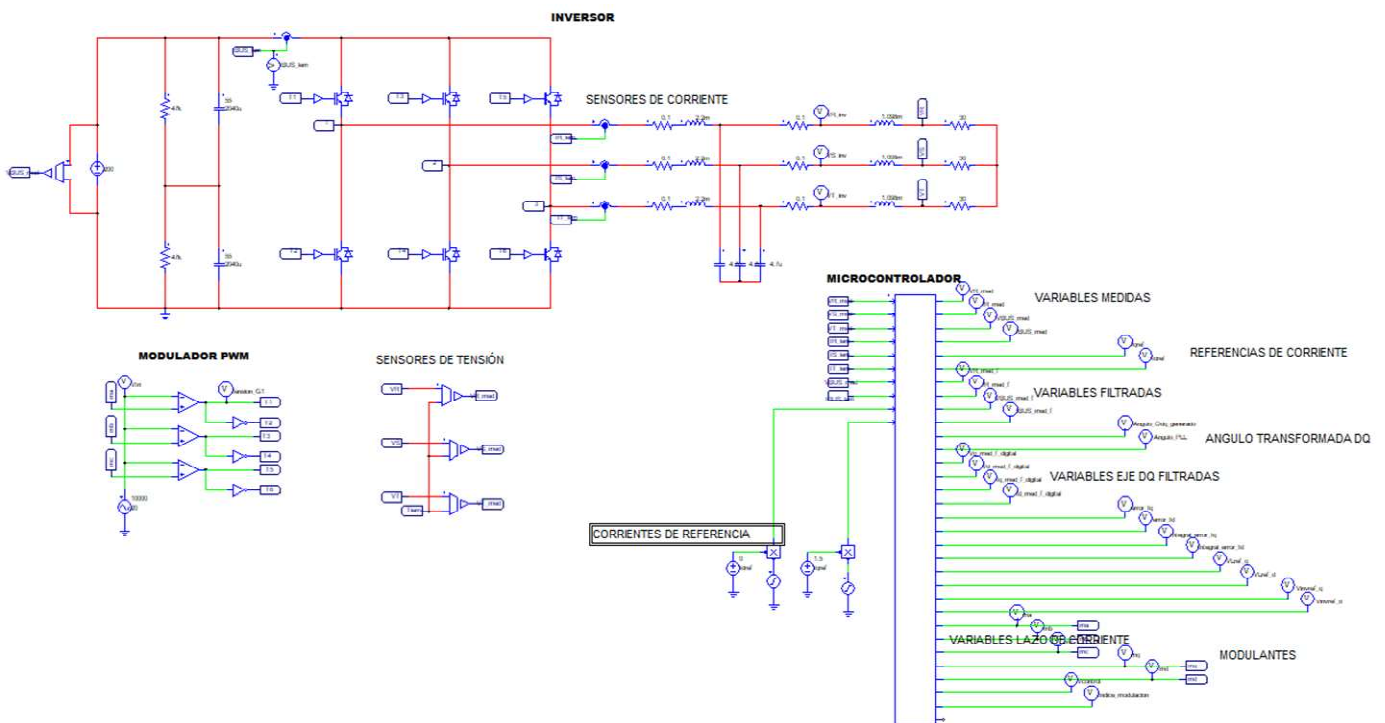


Figura 65. Esquema completo de simulación PSIM

La corriente por la resistencia en estado estacionario cuando la referencia es $I_q=1.5$ A se puede observar en la Figura 66.

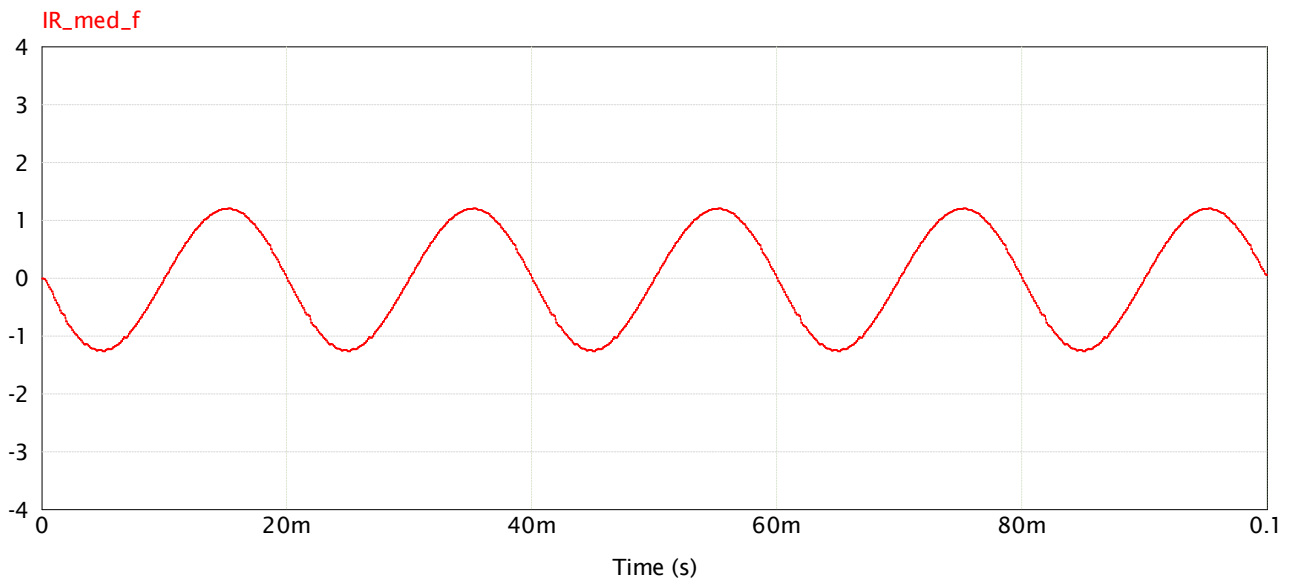


Figura 66. Corriente por la resistencia con una corriente de referencia de $I_q=1.5$ A PSIM

El valor pico de la corriente por la resistencia presenta un valor de 1.22 A este valor es razonable debido a que el valor de la corriente en los ejes dq tiene un valor absoluto de 1.5 A y debido a la transformada de Park utilizada la relación entre estos valores se muestra en la ecuación (56):

$$I_{dq} = I_{abc} \cdot \frac{\sqrt{3}}{2} \rightarrow I_{abc} = \frac{I_{dq}}{\frac{\sqrt{3}}{2}} = \frac{1.5}{\frac{\sqrt{3}}{2}} = 1.22 \text{ A} \quad (56)$$

La tensión en la resistencia se presenta en la Figura 67.

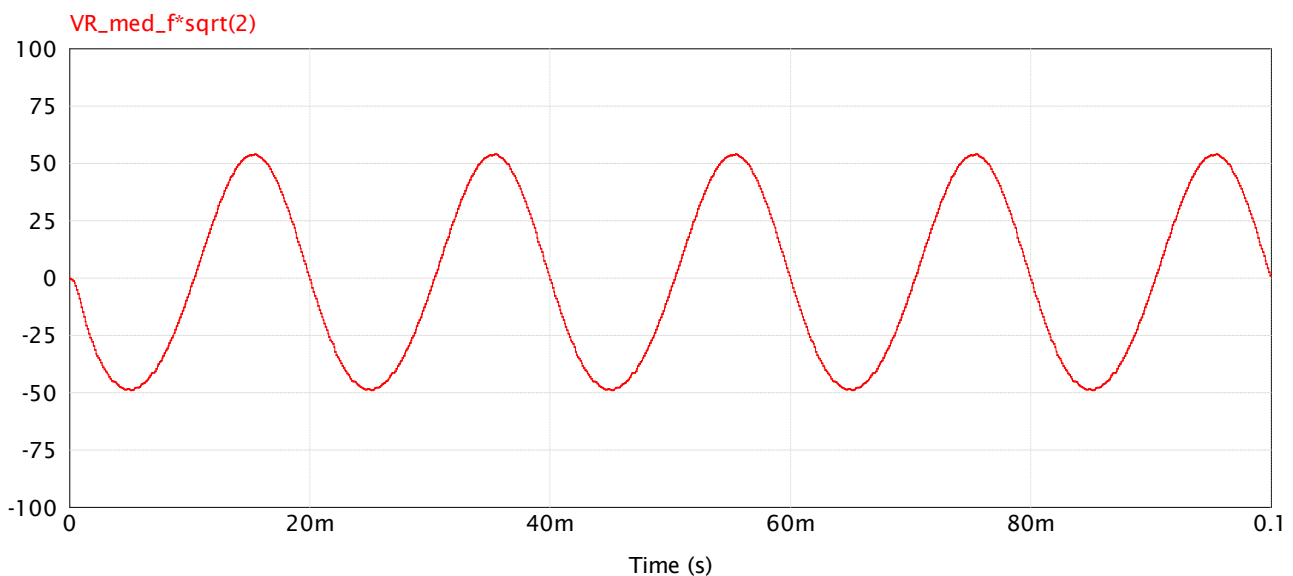


Figura 67. Tensión por la resistencia con una corriente de referencia de $I_q=1.5$ A PSIM

En la Figura 68 se puede observar la tensión y la corriente en una misma resistencia.

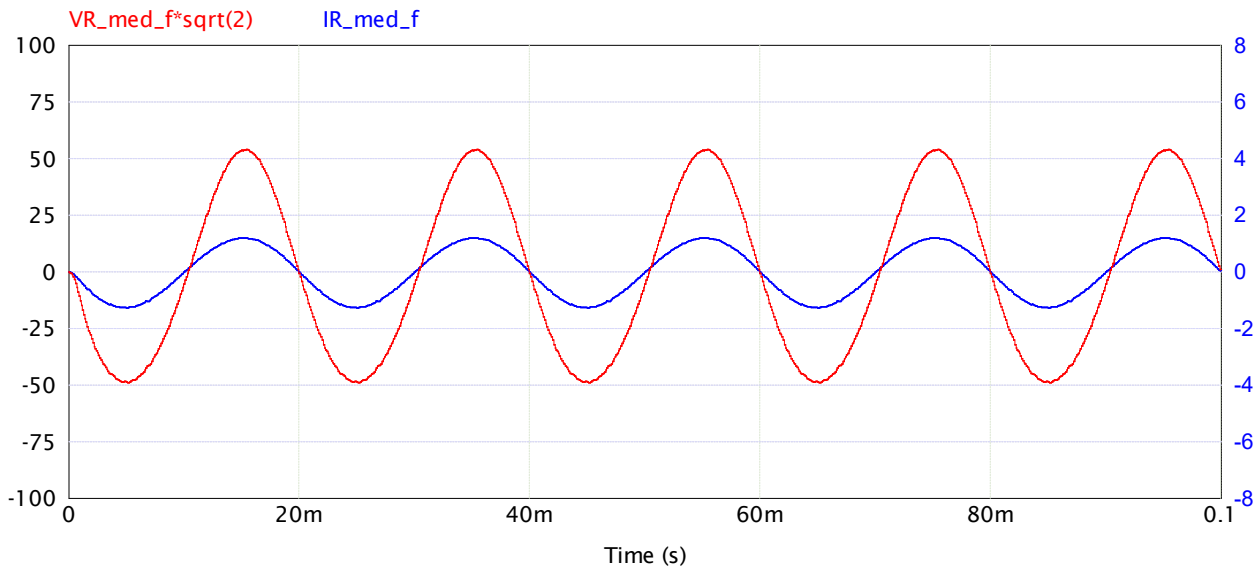


Figura 68. Tensión y corriente por la resistencia con una corriente de referencia de $I_q=1.5$ A PSIM

Aunque solo se haya mencionado una de las fases del sistema del sistema, las otras dos fases presentan valores similares pero desfasados con respecto a esta.

6.3 RESULTADOS EXPERIMENTALES

Una vez que las simulaciones han servido para la validación del modelo obtenido se procede al montaje del sistema en el laboratorio, para poder realizar comparaciones entre el sistema real y las simulaciones obtenidas.

En la Figura 69 se puede observar el montaje real del inversor con la placa de control DSP, la fuente DC, el filtro LCL y el banco de resistencias que actuará como carga resistiva. Por otro lado, también se puede observar las distintas sondas utilizadas para las medidas.

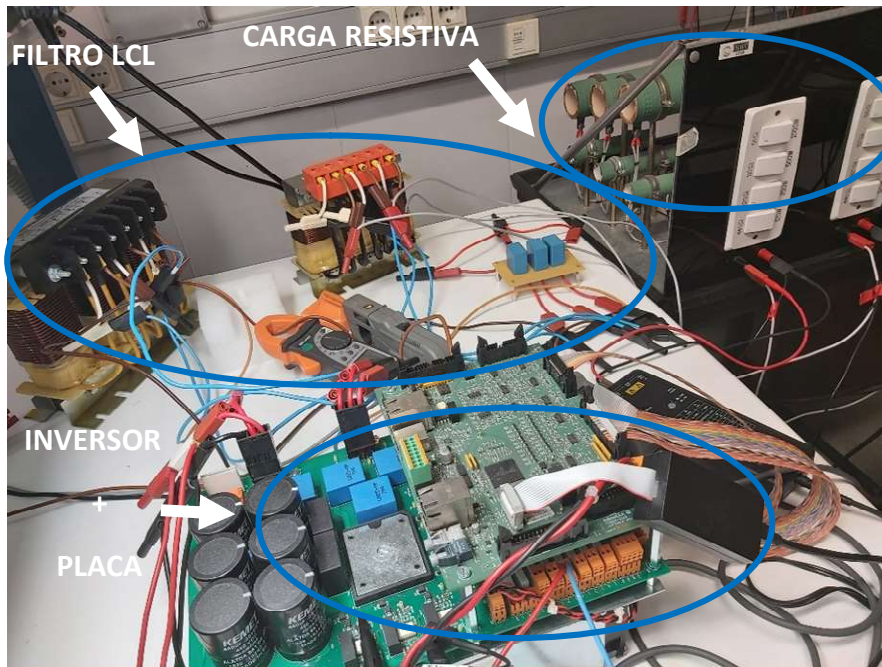


Figura 69. Inversor + Filtro LCL + Carga resistiva montado en el laboratorio

En la Figura 70 se puede observar la corriente por la resistencia cuando se tiene una corriente de referencia de $I_q=1.5$ A.

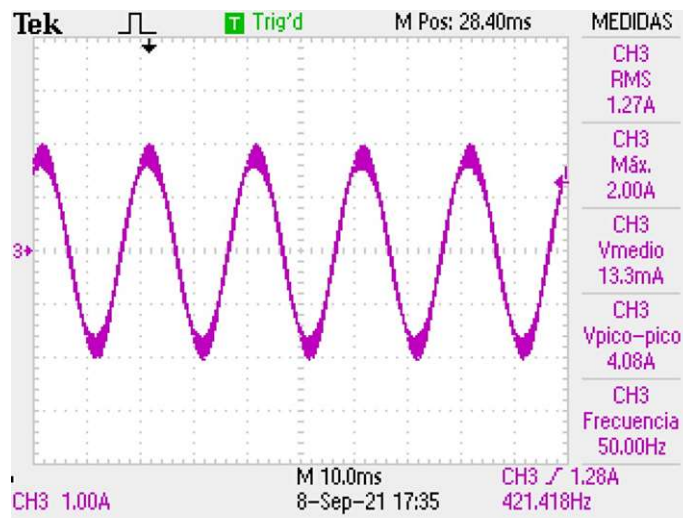


Figura 70. Corriente por la resistencia de la fase R con una corriente de referencia de $I_q=1.5$ A Prototipo

En la Figura 71 se puede observar la corriente por la resistencia cuando se tiene una corriente de referencia de $I_q=1.5$ A.

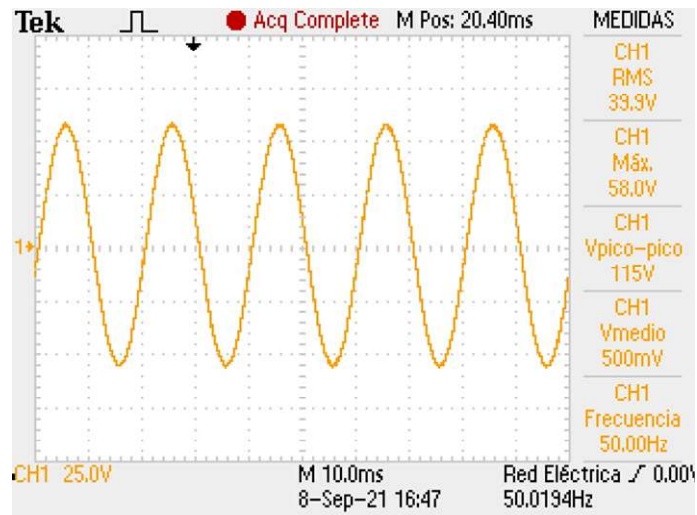


Figura 71. Tensión por la resistencia de la fase R con una corriente de referencia de $I_q=1.5$ A Prototipo

Por último, en la Figura 72 se puede observar la tensión y la corriente en una misma resistencia.

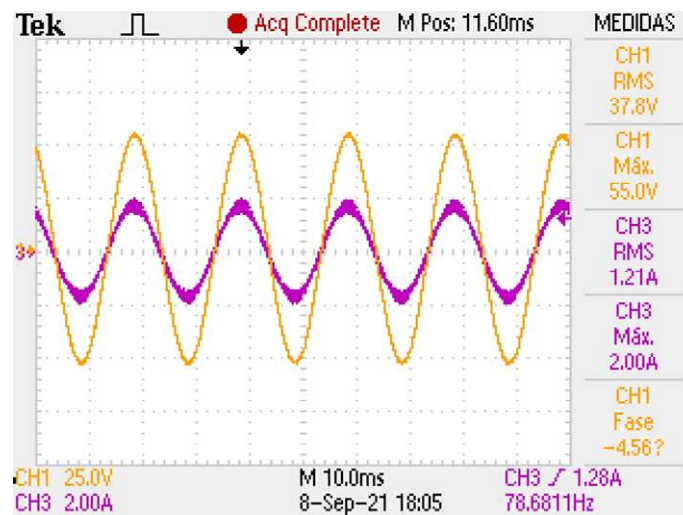


Figura 72. Tensión y corriente por la resistencia con una corriente de referencia de $I_q=1.5$ A Prototipo

Los valores obtenidos experimentalmente coinciden, en forma de onda y en magnitud, con los valores obtenidos por simulación, de lo que se puede deducir que el modelo en estado estacionario obtenido en PSIM queda ratificado por los resultados obtenidos de manera experimental con el prototipo.

A pesar de que únicamente se haya mencionado una única fase las demás fases tienen un comportamiento similar pero desfasadas con respecto a la fase analizada.

7. CONCLUSIONES Y LINEAS FUTURAS

7.1 CONCLUSIONES

El objetivo principal de este proyecto era el de implementar un lazo de control de corriente para un sistema real formado por el inversor PCO-10T800, la placa de control CDA01-CU3, un filtro LCL y conectado a una carga resistiva.

Para completar este objetivo se han llevado a cabo una serie de acciones entre las que destacan:

- La realización de un análisis de la interfaz Code Composer que a pesar de ser una interfaz cada vez más utilizada tiene una serie de limitaciones que hacen que su análisis no sea del todo intuitivo, es por ello por lo que dicho análisis de esta interfaz ha abarcado gran parte del tiempo de este proyecto.
- La obtención del modelo matemático, el diseño del control y la validación por simulación del sistema ha constituido una parte importante de este proyecto.
- Por último, la implementación y validación del control en el sistema real en el laboratorio.

Como conclusión se puede decir que el análisis de la interfaz, que permite la comunicación entre el usuario y el inversor, ha sido lo suficientemente exhaustivo como para poder implementar y validar lo obtenido en las simulaciones. Por otro lado, la validación del modelo no ha podido ser completa debido a las limitaciones que acompañan al DSP y a su interfaz gráfica, Code Composer.

Por otro lado, el modelado matemático y la elección de la estrategia de control al utilizar un sistema síncrono dq y un controlador PI y con ello trabajar con variables continuas ha sido correcta y ha facilitado la detección y la posterior resolución de errores.

7.2 LINEAS FUTURAS

Una de las ventajas de este proyecto tras su finalización es que deja un campo de futuras líneas de actuación alrededor de él.

El siguiente objetivo podría ser el de intentar conectar el sistema real a la red, como punto de partida se pueden utilizar tanto el modelado como el diseño y la implementación de la conexión a red hechas en este proyecto. Como complemento de esta misma futura línea de actuación se requerirá un análisis de los componentes armónicos del sistema.

Otro de los posibles puntos de actuación podría ser el del diseño de un lazo de tensión para el sistema.

Por último, en este proyecto se utiliza una fuente de tensión DC fija, sin embargo, en un sistema fotovoltaico esta tensión a la entrada del inversor puede variar por ello otra de las futuras líneas de actuación podría ser la de implementar un algoritmo del seguimiento del punto de máxima potencia (MPPT).

8. BIBLIOGRAFIA

- [1] Greenpeace, «¿Por qué hay que cambiar la energía para salvar el clima?,» 3 7 2019. [En línea]. Available: <https://es.greenpeace.org/es/en-profundidad/cambia-la-energia-no-el-clima/por-que-hay-que-cambiar-la-energia-para-salvar-el-clima/>. [Último acceso: 16 09 2021].
- [2] International Energy Agency (IEA), «Key world energy statics 2020».
- [3] International Renewable Energy Agency (IRENA), «Costo de generación de la energía renovable en 2019».
- [4] E. O. Ramírez Ortega, «Implementación de un convertidor PWM trifásico conectado a la red con un factor de potencia unitario,» tesis doctoral, 2014.
- [5] A. Baines García, «Puesta en marcha y Manual de Usuario del convertidor PCO-10T800 con placa de control F28M35 CDA01-CU3,» complemento al trabajo de fin de máster, 2020.
- [6] B. Brogeras García, «diseño y validación del control digital de un inversor de potencia en ejes de referencia síncronos conectado a red,» trabajo de fin de grado, 2011.
- [7] A. Villa Manrique, «Estudio del filtro LCL aplicado a inversores fotovoltaicos,» trabajo de fin de grado, 2011.
- [8] J. L. Agorreta Malumbres, «Modelado y control de inversores fotovoltaicos conectados en paralelo a la red,» tesis doctoral, 2011.
- [9] J. Samanes, A. Urtasun, E. L. Barrios, D. Lumbreras, J. López, E. Gubía y P. Sanchis, «Control Design and Stability Analysis of Power Converters: The MIMO Generalized Bode Criterion,» *IEEE Journal of Emerging and Selected Topics in Power Electronics (Volume: 8, Issue: 2, June 2020)*, vol. 8, nº 2, pp. 1880-1893.