Técnicas de procesamiento coherente de señales: implementación y aplicaciones en instrumentación

Tesis de Doctorado en Ingeniería

Matías Javier Oliva

Director de tesis: Pablo Andrés García Co-director de tesis: Alejandro Luis Veiga



Facultad de Ingeniería Universidad Nacional de La Plata

Resumen

La medición de señales de amplitudes pequeñas inmersas en ruido es un problema común en instrumentación electrónica, que aparece en áreas tan diversas como las telecomunicaciones, la bioingeniería o el control automático. Las técnicas a aplicar para posibilitar estas mediciones son muy variadas y dependen de la señal que se desee detectar y la naturaleza del ruido asociado. En particular, si la señal es periódica o puede ser modulada a una frecuencia determinada las técnicas de detección o demodulación coherente se presentan como una buena solución. Existe una gran variedad de estas técnicas, que aprovechan el conocimiento de la periodicidad de la señal para diferenciarla del ruido circundante. Entre estas técnicas destacan la amplificación *Lock-in* y la promediación coherente, las cuales han recibido especial atención tanto por la relativa simplicidad de su implementación como por los buenos resultados que permiten obtener. En la actualidad existen dispositivos comerciales que efectúan este tipo de mediciones, pero tienen la desventaja de ser costosos y voluminosos, bien adaptados para su uso en entornos controlados de laboratorio, pero no para aplicaciones embebidas y portables.

El desarrollo de sistemas de detección coherente de bajo costo, portables y bien diseñados para su integración con dispositivos funcionales se ha identificado como un problema con diversas aplicaciones. En este punto se destacan tres por su relevancia en esta tesis, aunque el abanico de casos es mucho más amplio. Estas son: la Tomografía Magnética para la detección de nanopartículas ferromagnéticas en técnicas de terapias oncológicas, el desarrollo de interfaces cerebro computadora basadas en estímulos visuales periódicos y la instrumentación de sistemas de Espectroscopía Mössbauer. El primer capítulo de esta tesis se encarga de presentar estas aplicaciones y justificar la utilidad de las técnicas de detección coherente para su implementación.

Con el fin de desarrollar estas aplicaciones se han explorado en detalle los aspectos teóricos de las técnicas de *Lock-in* y promediación coherente y se ha evaluado cómo se pueden combinar para lograr dispositivos eficientes. En este sentido se ha determinado que, bajo ciertas circunstancias, combinar las técnicas de promediación coherente y *Lock-in* proporciona los mismos resultados que usar esta última individualmente. Luego, esta propiedad se ha utilizado para minimizar la cantidad de multiplicaciones involucradas en los cálculos, contribuyendo a mejorar la eficiencia de los algoritmos. La revisión detallada de los aspectos teóricos de estas técnicas y la derivación del algoritmo resultante, bautizado CALI, puede encontrarse en el segundo capítulo de esta tesis.

Luego del estudio de los aspectos teóricos de la detección coherente se aborda su implementación, para su posterior aplicación en los contextos planteados. La implementación tiene estrictos requerimientos de tiempo real, lo que es difícil de cumplir con microprocesadores tradicionales, especialmente cuando la frecuencia de la señal es elevada. Es por esto que las implementaciones exploradas incorporan el procesamiento digital de la señal en lógica programable. El desarrollo de sistemas basados en dispositivos de este tipo lleva aparejados una gran cantidad de desafíos que se han intentado simplificar a lo largo de esta investigación. En este sentido, se ha desarrollado un sistema reutilizable de procesamiento de señales de código abierto, que se ha probado en sistemas de lógica programable de distintos fabricantes. El desarrollo de este sistema reutilizable de señales comprende el tercer capítulo de esta tesis.

A partir de este sistema reutilizable se han diseñado e implementado sistemas de detección coherente de señales en distintos dispositivos de lógica programable. Utilizando lo estudiado previamente, se han diseñado estas herramientas de forma eficiente, mediante una combinación de las técnicas de *Lock-in* y promediación coherente. La descripción de estos diseños y su caracterización se puede encontrar en el cuarto capítulo de esta tesis.

En el quinto capítulo se describe cómo los dispositivos desarrollados se utilizaron en las aplicaciones propuestas. Estos logran solucionar algunos de los desafíos inherentes de cada aplicación, proveyendo equipos embebidos y portables. En el campo de la Tomografía Magnética se ha logrado predecir la posición de las partículas ferromagnéticas en elementos de prueba, reemplazando a un *Lock-in* comercial, voluminoso y difícilmente trasladable utilizado en implementaciones anteriores, acelerando además el proceso de medición. En el campo de las interfaces cerebro computadora se ha logrado detectar la atención del usuario a tres estímulos visuales distintos, sin que éste realice acciones motoras, utilizando amplificación *Lock-in*. A diferencia de esta propuesta, la mayoría de los trabajos previos recurren a la Transformada de Fourier, lo que representa una solución menos eficiente. Finalmente, en el campo de la Espectroscopía Mössbauer se han utilizado los sistemas desarrollados para optimizar el proceso de calibración de los equipos, mejorando la calidad de los espectros obtenidos y reemplazando el uso del osciloscopio en la promediación coherente, que ahora se realiza de forma más simple y automatizada. En todos los casos se destaca la flexibilidad y escalabilidad de los dispositivos implementados.

Finalmente, en el sexto capítulo se enumeran las principales conclusiones de este trabajo de investigación, resaltando los logros alcanzados y las limitaciones encontradas a lo largo de su desarrollo. Además, se explora el trabajo a futuro que surge de este estudio. Se presentan varios aspectos que quedan por profundizar, tanto en las aplicaciones prácticas como en los aspectos teóricos, por lo que esta investigación sirve como un punto de partida para diversas líneas de trabajo.

Índice general

1	\mathbf{Intr}	Introducción 7				
	1.1.	Sistemas de detección coherente				
	1.2.	Dispositivos SoC-FPGA				
	1.3.	Técnicas tomográficas para detección de nanopartículas				
		1.3.1. Hipertermia Magnética				
		1.3.2. Tomografía Magnética				
		1.3.2.1. Fundamentos \ldots 10				
		1.3.2.2. Determinación de los parámetros de operación				
		1.3.2.3. Medición de la tensión en la espira sensora $\ldots \ldots \ldots \ldots \ldots \ldots 12$				
	1.4.	Interfaces cerebro computadora				
	1.5.	Espectroscopía Mössbauer				
	1.6.	Propuesta para un sistema de detección coherente en SoC-FPGA 15				
	1.7.	Resumen de publicaciones producidas 15				
		1.7.1. Trabajos publicados en revistas				
		1.7.2. Trabajos publicados en congresos				
		1.7.3. Repositorios de código abierto 16				
	1.8.	Estructura de la tesis				
2	Téc	nicas de detección coherente 18				
-	2.1.	Introducción				
	2.2.	Lock-in 19				
		2.2.1. Lock-in de una fase				
		2.2.2. Lock-in de dos fases				
		2.2.3. Análisis en el Dominio Compleio				
		2.2.4. Efecto sobre el ruido blanco				
		2.2.5. Elección del filtro pasa-bajos				
		2.2.5.1. Filtro de media móvil				
		2.2.5.2. Filtros de respuesta impulsional infinita (IIR)				
		2.2.6. Efecto del ruido de cuantización				
		2.2.6.1. Case $f_m = K f_c$				
		2.2.6.2. Case $f_m \neq K f_c$				
		2.2.6.3 Besumen				
		2.2.7. Case $f_c = 100 \text{ kHz}$				
		2.2.8 Antecedentes 35				
	2.3	Promediación coherente 40				
	2.0.	2.3.1 Fundamentos 40				
		2.3.2. Antecedentes				

	2.4.	Combinación de promediación coherente y Lock-in								
		2.4.1. Formulación matemática								
		2.4.1.1. Conceptos generales $\ldots \ldots 43$								
		2.4.1.2. Aproximación intuitiva $\ldots \ldots 44$								
		2.4.1.3. Demostración formal $\ldots \ldots 45$								
		2.4.1.4. Número de operaciones								
		2.4.1.5. Simulaciones								
	2.5.	Conclusiones								
~	ъ									
3	Pro	cesamiento de senales en SoC-FPGA 51								
	ა.⊥. აი	Introduccion								
	3.2.	Cyclone V SoU-FPGA								
		3.2.1. Placa de expansion Highspeed AD/DA daughter cara (ADA-GPIO) 53								
	<u></u>	3.2.2. Acoptamiento en continua de la placa Highspeed AD/DA Card								
	3.3.	Sistema reutilizable de procesamiento de senales								
		3.3.1. Esquema general del diseno								
		3.3.2. Etapa de control								
		3.3.3. Etapa de fuente de senal \ldots								
		$3.3.3.1.$ Senal digital \ldots 01								
		3.3.3.2. Sintesis digital directa por truncamiento de fase								
		3.3.3.3. Generation de secuencias pseudo-aleatorias								
		$3.3.3.4.$ Controlador del ADC High-Speed $\ldots \ldots \ldots$								
		$3.3.3.5.$ Controlador del DAC High-Speed $\ldots \ldots \ldots$								
		3.3.3.6. Controlador del ADU LIU 2308								
		3.3.4. Etapa de procesamiento de senal								
	9.4	3.3.4.1. Filtros de respuesta impulsional finita								
	3.4.	Ejemplo de procesamiento de senales en SoU-FPGA								
		$3.4.1. \text{ Resultados} \dots \dots$								
		3.4.1.1. Recursos utilizados								
		3.4.1.2. Mediciones temporales								
	95	3.4.1.3. Respuesta del filtro FIR								
	3.5.	Adaptacion del sistema a dispositivos SoC-FPGA de AMD-Ailinx								
		3.5.1. Red Pitaya								
		3.5.2. Implementation								
		3.5.3. Uso de recursos y temporización								
	9.0	3.5.4. Filtro pasa-bajos								
	3.6.									
4	Det	ección coherente en SoC-FPGA 77								
	4.1.	Introducción								
	4.2.	Implementación del sistema en DE1-SoC								
		4.2.1. Hardware								
		4.2.1.1. Promediación coherente								
		4.2.1.2. Lock-in								
		4.2.1.3. Combinación de promediación coherente y Lock-in								
		4.2.2. Interfaz de usuario								
		4.2.2.1. Programas base para el sistema de detección coherente								
		4.2.2.2. Servidor web								

	4.3.	3. Resultados en De1-SoC					
		4.3.1.	Tiempo individual de operaciones	. 89			
		4.3.2.	Uso de recursos de cada algoritmo	. 90			
		Implementación del sistema de detección coherente	. 90				
		4.3.3.1. Consideraciones temporales	. 90				
		4.3.3.2. Uso de recursos del sistema completo	. 92				
		4.3.3.3. Mediciones de Ruido	. 92				
			4.3.3.4. Medición de transferencias conocidas	. 92			
	4.4.	Impler	mentación del sistema en Red Pitaya	. 96			
		4.4.1. Fuente de datos y conversor digital analógico					
		4.4.2.	Lock-in	. 98			
		4.4.3.	Control del sistema	. 99			
	4.5.	Result	ados en Red Pitaya	. 99			
		4.5.1.	Uso de recursos y temporización	. 99			
		4.5.2.	Mediciones de ruido	. 99			
		4.5.3.	Medición de transferencias conocidas	. 100			
	4.6.	Comp	aración de DE-li y RP-li	. 103			
		4.6.1.	Aspectos generales	. 103			
		4.6.2.	Mediciones en f=100 kHz \ldots	. 105			
	4.7.	Conclu	usiones	. 107			
5	Anl	icacior	105	108			
0	5 1	Tomor	orafía de permeabilidad magnética	108			
	0.1.	511	Diseño del Tomógrafo Magnético	109			
		0.1.1.	5.1.1.1 Diseño del arreglo de bobinas y del circuito magnético	111			
			5.1.1.2 Circuito controlador de campo magnético	111			
			5.1.1.3 Tiempos de medición	112			
			5.1.1.4 Adaptación del DE-li al tomógrafo magnético	113			
			5.1.1.5. Beconstrucción	. 113			
		512	Resultados	114			
		0.1.2.	5.1.2.1 Determinación del nivel de error en la medida de cada espira	114			
			5.1.2.2. Variaciones de V_0 en el tiempo	. 115			
			5.1.2.3. Caracterización de las variaciones de tensión en espiras sensoras	. 115			
			5.1.2.4. Beconstrucción de imágenes con <i>phantoms</i>	. 117			
		5.1.2.4. Acconduction de anagenes con <i>phantoms</i>					
	5.2.	Interfa	aces cerebro computadora	. 121			
		5.2.1.	Introducción	. 121			
		5.2.2. Implementación del sistema					
			5.2.3.1. Promediado de canales	. 124			
			5.2.3.2. Interferencia electromagnética	. 126			
		5.2.4.	Alimentación y aislación	. 126			
		5.2.5.	Detección	. 126			
		5.2.6.	Estimulación visual	. 128			
		5.2.7.	Resultados	. 128			
		5.2.7.1. Medidas de ruido de adquisición con ADS131E08					
			5.2.7.2. Implementación del sistema de detección de SSVEPs	. 130			
			5.2.7.3. Medidas de SSVEPs	. 131			

		5.2.8.	Conclusio	ones	. 132							
	5.3. Espectroscopía Mössbauer											
		5.3.1.	Introduce	ción	. 135							
		5.3.2.	Implement	ntación del sistema	. 137							
			5.3.2.1.	Procesamiento en la FPGA	. 138							
			5.3.2.2.	Promedio Lineal	. 138							
			5.3.2.3.	Promediación Coherente	. 139							
			5.3.2.4.	Límites de K y N_{ca}	. 140							
			5.3.2.5.	Control de Disparo	. 140							
			5.3.2.6.	Control del μ P	. 141							
5.3.3. Resultados \ldots			os	. 141								
			5.3.3.1.	Utilización de recursos	. 141							
			5.3.3.2.	Validación del sistema	. 142							
		5.3.4.	Conclusio	ones	. 142							
6	tes originales y líneas abiertas de trabajo	144										
	6.1.	Conclu	siones gei		. 144							
	6.2.	Líneas	abiertas o	de investigación	. 146							
Bibliografía												

Capítulo 1

Introducción

1.1. Sistemas de detección coherente

La medición de señales de amplitudes pequeñas inmersas en ruido es un problema común en instrumentación electrónica. Cuando se hacen mediciones de estas características, la relación entre la magnitud de la señal deseada y el nivel de ruido que la contamina suele conocerse como relación señal a ruido, y es usualmente representada por la sigla SNR, de su acrónimo en inglés.

A medida que decrece la SNR de un sistema es cada vez más difícil detectar la señal deseada. Sin embargo, si su frecuencia de oscilación es conocida o al menos estimable, o si puede modularse a una frecuencia conocida, es posible aplicar técnicas de detección (o demodulación) coherente para facilitar la tarea. Estas técnicas aprovechan el conocimiento de la frecuencia de la señal para procesarla, discriminando los datos de interés de la señal del ruido de fondo.

El interés por estas técnicas se remonta a 1941 [1], y en sus inicios se recurrió a circuitos puramente analógicos para detectar las señales de interés. A medida que la industria de electrónica digital fue avanzando, estas técnicas pasaron a aplicarse en sistemas muestreados con implementaciones digitales. Actualmente, existen dispositivos comerciales que realizan detección coherente de señales con enfoques mayormente digitales [2], [3]. Estos sistemas resultan muy útiles en aplicaciones de laboratorio, pero son instrumentos costosos y difíciles de incorporar en sistemas funcionales.

Para aplicar estas técnicas correctamente es clave mantener en todo momento el sincronismo de la señal medida con alguna señal de referencia interna, para aprovechar la periodicidad de la señal incógnita. Este requerimiento hace que la implementación de estos sistemas tenga estrictos requerimientos de tiempo real, lo que es difícil de cumplir con microprocesadores tradicionales, sobre todo cuando la frecuencia de la señal es elevada [4]. Es por esto que la mayoría de los dispositivos comerciales incorporan el procesamiento digital de la señal en lógica programable.

La implementación de estas técnicas de detección coherente en sistemas embebidos basados en lógica programable, con un enfoque de software y hardware abierto, y su aplicación a distintos escenarios es lo que ha motivado esta tesis.

1.2. Dispositivos SoC-FPGA

Un aspecto importante a considerar a la hora de implementar sistemas de detección coherente confiables es la plataforma en la que se implementarán estos sistemas, sobre todo cuando se quieren alcanzar frecuencias de demodulación altas, como en algunas de las aplicaciones que se han desarrollado durante esta tesis.

Existen variedad de plataformas que pueden servir a este propósito. La primera en que se puede pensar es en una computadora personal (PC). Las PCs son sistemas versátiles y potentes, capaces de ejecutar una amplia gama de aplicaciones y tareas. Pero su principal desventaja surge precisamente de que con ella se ejecutan muchos sistemas secuenciales en simultáneo. Un sistema de procesamiento y detección coherente de señales tiene importantes restricciones de tiempo real duro: aquellos en donde si el tiempo de respuesta está por encima de una cota se dice que ha fallado [4]. Estos requerimientos justifican la elección de sistemas embebidos (SE) para la aplicación de estas técnicas.

A diferencia de una PC convencional, un SE se caracteriza por un enfoque especializado y mayor eficiencia en recursos. Se denomina sistema embebido o empotrado a un sistema electrónico con capacidad de cómputo diseñado para ejecutar una o varias tareas bien específicas. En general los SE poseen menos capacidad de cómputo que las computadoras personales y se diseñan optimizando el tamaño y el consumo de energía [4].

Una opción interesante para el desarrollo de SEs son los dispositivos SoC-FPGA, que se caracterizan por contar en un mismo chip con un microprocesador (μ P) y un arreglo de celdas lógicas programables (FPGA de su acrónimo en inglés). La FPGA permite implementar directamente en hardware las tareas que requieren alta velocidad y determinismo, mientras que el microcontrolador le otorga flexibilidad al conjunto, permitiendo que se adapte a distintas condiciones, inclusive en tiempo de ejecución. El acrónimo SoC viene de *system on chip*: sistema en un chip.

La principal desventaja de estos sistemas tiene que ver con el tiempo de desarrollo. Un diseñador de sistemas SoC-FPGA enfrenta una curva de aprendizaje empinada y debe adquirir una variedad de habilidades para abordar con éxito los desafíos inherentes. Estos incluyen el diseño del hardware de la FPGA, en algún lenguaje de descripción de hardware como Verilog o VHDL, el desarrollo del software embebido que debe ejecutarse en el microprocesador y la comunicación exitosa de ambos elementos de procesamiento. Todo este proceso requiere de un diseñador especializado en diseño digital y en la arquitectura particular de la plataforma objetivo.

A lo largo de esta investigación se ha trabajado en simplificar este flujo de trabajo. En este sentido, se han desarrollado diseños modelo para el procesamiento de señales en SoC-FPGA, inicialmente en plataformas Intel-Altera y posteriormente adaptados con éxito a dispositivos de AMD-Xilinx. Luego, estos modelos fueron utilizados para implementar los sistemas de detección coherente, objeto de este trabajo, en dispositivos de ambos fabricantes. Todo el código generado es de acceso abierto y está disponible a través de los enlaces proporcionados en cada sección.

Si bien hay una enorme cantidad de contextos en los que se pueden aplicar las técnicas de procesamiento coherente de señales, la principal aplicación que ha impulsado este trabajo de investigación ha sido la implementación de un Tomógrafo Magnético, destinado a la detección de partículas ferromagnéticas en técnicas de terapias oncológicas. Mientras se perseguía este objetivo han surgido nuevas líneas de investigación, dando mayor valor al eje principal de este trabajo. En este sentido se destacan principalmente dos contribuciones adicionales: la implementación de sistemas de detección coherente en SoC-FPGA para impulsar interfaces cerebro computadora (comúnmente conocidas por sus siglas en inglés BCI, *brain-computer interface*) y la calibración de un equipo de Espectroscopía Mössbauer.

1.3. Técnicas tomográficas para detección de nanopartículas

1.3.1. Hipertermia Magnética

Las propiedades curativas del calor en diversas enfermedades son conocidas desde hace mucho tiempo. Estas terapias, comúnmente conocidas como hipertermias, tienen sus primeros antecedentes hace unos 150 años, cuando Busch descubrió que los tumores dejan de crecer a temperaturas superiores a 42 °C, mientras que el tejido sano no se ve afectado [5]. Estas terapias han evolucionado enormemente a través de los años y actualmente dañar el tejido tumoral mediante un aumento local de la temperatura constituye una terapia contra el cáncer aprobada y establecida [6], [7]. Para lograr alcanzar las temperaturas necesarias para el tratamiento del cáncer en tejidos biológicos se han probado diversas técnicas [8], entre las cuales se pueden citar el calentamiento óhmico mediante corrientes de alta frecuencia aplicadas a través de electrodos [9], la irradiación por microondas aplicada a través de antenas de radiofrecuencia [10], [11], la irradiación láser a través de fibras [12] o el calentamiento en baño de agua [13]. Todas estas técnicas adolecen de un problema común: el control de la extensión espacial del calentamiento del tejido sigue siendo un desafío sin resolver [14].

La hipertermia por partículas magnéticas (o simplemente hipertermia magnética) busca resolver este problema introduciendo nanopartículas magnéticas (NPs), que actúan como fuente de calor, en el tejido tumoral. Las NPs se calientan mediante el uso de un campo magnético alternante externo, logrando controlar la extensión espacial del calentamiento del tejido. Los trabajos experimentales sobre hipertermia magnética se remontan hace casi 60 años [15] y en la actualidad han demostrado su éxito en los primeros ensayos clínicos [16], [17], [18], [19], en particular en tratamientos para el cáncer de próstata y gliomas. Estos ensayos además han mostrado que las molestias y efectos secundarios resultan moderados.

Si bien los estudios recientes de esta terapia son muy prometedores, el método aún necesita importantes mejoras antes de convertirse en un procedimiento médico estándar [20]. Para ello existen principalmente dos inconvenientes: por un lado, es crucial mejorar la potencia de calentamiento específica de las NPs para alcanzar y mantener temperaturas terapéuticamente adecuadas dentro del tejido tumoral con la menor cantidad posible de material magnético entregado al tumor. Por otro, es necesario determinar un método seguro, cómodo y reproducible para posicionar las NPs en la región tumoral.

Este es un problema particularmente desafiante, ya que el entorno biológico interactúa con las NPs, remodela su estructura y cambia sus propiedades. Esta interacción puede resultar en la eliminación de las NPs del cuerpo, o simplemente en una modificación en su distribución [21], [22]. En este contexto surge la necesidad de contar con una forma de determinar la posición en tiempo real de las NPs de una forma no invasiva, no destructiva, y sin contacto, para rastrear la distribución de las NPs antes y después del procedimiento médico [23].

La implementación de un instrumento diseñado para abordar este problema, considerando su importancia en el contexto de esta terapia oncológica emergente, es el principal problema que ha buscado solucionar este trabajo de investigación. Para aquellos lectores interesados en profundizar en las técnicas específicas, los ensayos clínicos y los antecedentes en hipertermia magnética, un resumen completo se encuentra disponible en [14].

1.3.2. Tomografía Magnética

1.3.2.1. Fundamentos

Para determinar en tiempo real la ubicación de las nanopartículas magnéticas las técnicas tomográficas resultan de especial interés. La tomografía es una técnica de exploración, que permite obtener imágenes de un corte o plano concreto de un cuerpo o un objeto [24]. Existen una amplia variedad de técnicas de tomografía, pero las técnicas eléctricas, en particular, cobran especial relevancia para el objeto de esta tesis. Si bien estas técnicas no ofrecen la mejor resolución de imagen, presentan características ventajosas para esta aplicación: son económicas, seguras, no invasivas y no utilizan radiación ionizante. Las técnicas eléctricas se pueden subdividir aún más y se ha prestado considerable atención a los métodos basados en la medición de cantidades eléctricas pasivas, a saber, resistencia (impedancia), capacitancia e inductancia [25].

La tomografía de impedancia eléctrica (EIT, por sus siglas en inglés) se basa en la medición de patrones de resistencia y reactancia [26], [27], [28] y puede producir imágenes de conductividad y permitividad. La tomografía de capacitancia eléctrica (ECT, por sus siglas en inglés) implica la medición de perfiles de capacitancia [29], [30], [31] y genera imágenes de distribuciones de permitividad. La tomografía de mutua inductancia, también conocida como tomografía de inductancia magnética (MIT, por sus siglas en inglés), completa el conjunto de estas técnicas, y es la que se desarrollará durante esta tesis.

Esta técnica tomográfica permite mapear, sin contacto y en forma no invasiva, las propiedades electromagnéticas pasivas (PEP) de un objeto mediante la medición de la inductancia mutua entre pares de bobinas colocadas alrededor de un perímetro [32], [33]. La MIT es sensible a las tres propiedades PEP de un objeto: conductividad eléctrica, permitividad dieléctrica y susceptibilidad magnética (σ , ϵ y χ_m respectivamente). Esta última es una constante de proporcionalidad adimensional que indica el grado de sensibilidad a la magnetización de un material influenciada por un campo magnético externo. Se ha observado que existe una diferencia de varios ordenes de magnitud entre la χ_m de los tejidos biológicos y la χ_m de las NPs en coloides terapéuticos. En ausencia de NPs, los tejidos biológicos tienen una χ_m del orden de 10⁻⁶ [34], mientras que se espera que la susceptibilidad de una dosis terapéutica de NPs de óxido de hierro esté en el rango de 10⁻⁴ a 10⁻³ [35]. Esto implica que, si se pudiera ajustar la selectividad de MIT para la susceptibilidad magnética, podría emplearse esta técnica para detectar con éxito la presencia de NPs.

La Figura 1.1 muestra el esquema básico de un sistema de tomografía de mutua inductancia. Para su implementación es necesaria una espira excitadora y una espira sensora. Cuando una corriente eléctrica alterna circula a través de la espira excitadora se produce un campo eléctrico variable en ella. Este producirá un campo magnético en el espacio, que a su vez induce una tensión en la espira sensora. La naturaleza de esta tensión depende de las propiedades electromagnéticas del objeto ubicado entre las espiras, o de su ausencia. En consecuencia, esta medición permite inferir características relevantes del objeto.

Aumentando la cantidad de espiras se puede mejorar la resolución espacial del sistema. En un esquema de N espiras por ejemplo, cada espira puede ser utilizada alternativamente como excitadora o sensora, produciendo N – 1 mediciones por espira, para un total de N(N – 1) mediciones. Luego estas pueden utilizarse para reconstruir el medio circundante entre espiras y determinar la presencia o ausencia de NPs en él. Un buen resumen que cubre los avances de la metodología, la instrumentación y las aplicaciones de esta técnica se puede encontrar en [36].



Figura 1.1: Tomografía magnética. La figura muestra las inductancias de las distintas espiras y del objeto de interés, junto con las corrientes inyectadas o inducidas. También se representan la inductancia mutua entre la espira excitadora y el objeto bajo estudio (M), entre el objeto y la espira sensora (M_{od}) y entre espiras (M_{ed}).

1.3.2.2. Determinación de los parámetros de operación

La determinación de las variaciones de tensión esperadas en la espira sensora es un problema complejo, que parte de las ecuaciones de Maxwell con condiciones de contorno adecuadas. Una solución analítica de estas ecuaciones diferenciales solo es posible en algunos casos simples. Por ejemplo, en [37] se propone una solución analítica para el caso de un objeto cilíndrico de radio R y grosor t posicionado coaxialmente a la mitad entre dos bobinas a una distancia 2a, con t \ll 2a. Esta deducción puede extenderse para el caso que se muestra esquemáticamente en la Figura 1.2: un disco de tejido de radio R, grosor t y susceptibilidad despreciable, que incluye un disco más pequeño en su interior de radio R_p que contiene NPs con susceptibilidad magnética χ_m [23].

En estas condiciones, considerando que la contribución magnética se debe exclusivamente a la susceptibilidad magnética de las NPs y simbolizando la conductividad eléctrica, la permeabilidad relativa del tejido y la frecuencia de operación con σ , ϵ_r y f respectivamente, se pueden expresar las variaciones de tensión en las bobinas de detección como:

$$\frac{\Delta B}{B} = \frac{\Delta V}{V} = K_{p}\chi_{m} + K\epsilon_{r}\epsilon_{0}\omega^{2} + K\sigma\omega, \qquad (1.1)$$

donde B es el campo magnético, ω la frecuencia angular ($\omega = 2\pi f$) y $\epsilon_0 = 8,85 \times 10^{-12} \frac{\text{F}}{\text{m}}$ es la permitividad del vacío. K y K_p dependen de la geometría y están dadas por:

$$K_{p} = a^{3} \frac{t}{2} R_{p}^{2} \frac{8a^{2} - R_{p}^{2}}{(a^{2} + R_{p}^{2})^{4}}$$
(1.2)

$$\mathbf{K} = \mu_0 \mathbf{a}^3 \mathbf{t} \left(\frac{1}{\mathbf{a}^2} - \frac{\mathbf{a}^2 + 2\mathbf{R}^2}{(\mathbf{a}^2 + \mathbf{R}^2)^2}\right)$$
(1.3)

donde $\mu_0 = 4\pi 10^{-7} \frac{\text{H}}{\text{m}}$ representa la permeabilidad magnética del vacío.

La tensión inducida en la espira sensora para esta geometría en particular depende de tres términos: Uno proporcional a la susceptibilidad magnética, χ_m , uno proporcional a la permitividad dieléctrica, ϵ_r , y uno proporcional a la conductividad eléctrica, σ . Es importante notar que el término de interés, en el caso que nos concierne, es el que depende de χ_m , que es el único que no



Figura 1.2: Modelo simplificado para una solución analítica de la tensión inducida en la espira sensora. Disco de radio R, que incluye una zona concéntrica de radio R_p que contiene NPs. El disco está centrado en el medio de dos espiras separadas 2a. El grosor del disco t es mucho menor a a.

depende de la frecuencia. Esto significa que se puede buscar una frecuencia de operación que haga despreciables los términos que dependen de ϵ_r y σ , posibilitando la medición.

Examinando la ecuación 1.1 se ve que mientras más baja sea la frecuencia menor sera el aporte de los términos que dependen de ϵ_r y σ , es decir que en un principio una frecuencia baja sería deseable para el experimento. Sin embargo, según la ley de Faraday, la tensión inducida en las espiras es proporcional a la variación temporal del campo magnético, por lo que éste resulta más difícil de detectar a medida que disminuye la frecuencia. En [23] se estudian los aportes de cada término en la ecuación 1.1 para distintos tejidos del cuerpo humano, como la sangre, el hígado, la grasa, el riñón, el pulmón y el intestino. En este análisis se llega a la conclusión de que una frecuencia de 100 kHz constituye una buena solución de compromiso entre una frecuencia de operación lo suficientemente alta para efectuar la medición, y una lo suficientemente baja para despreciar los términos no deseados, y es la frecuencia objetivo que se tomará a lo largo de este trabajo.

1.3.2.3. Medición de la tensión en la espira sensora

Suponiendo que los términos no deseados son despreciables a la frecuencia de interés, y considerando una geometría como la descripta en la sección anterior, con 2a = 45 mm, R = 40 mm, $R_p = 7 \text{ mm}$ y t = 1 mm, las variaciones de tensión en la espira sensora que se pretender detectar pueden obtenerse a partir de las ecuaciones 1.1 y 1.2:

$$\frac{\Delta B}{B} = \frac{\Delta V}{V} \approx K_p \chi_m \approx 0.01174 \chi_m \tag{1.4}$$

Como se ha mencionado en la sección 1.3.2.1, la χ_m de las NPs está en el rango de 10^{-4} a 10^{-3} . Suponiendo el peor caso para la detección, esto implica que se quiere medir una variación de tensión relativa de aproximadamente $0,0117 \times 10^{-4} = 1,17 \times 10^{-6}$.

Una medición de estas características no es sencilla, ya que la señal esta inmersa en ruido que puede ser de algunos ordenes de magnitud mayor. Sin embargo, la señal de excitación del experimento, es decir, la señal que pasa a través de la espira excitadora, es conocida, y de hecho es generada por el mismo aparato de medición. Esto posibilita utilizar sistemas de detección coherente de señal para realizar las mediciones.

1.4. Interfaces cerebro computadora

Las interfaces cerebro computadora (o BCIs, por su acrónimo en inglés) son dispositivos que proporcionan al usuario un nuevo canal de comunicación y control, para el cuál no debe realizar acciones motoras [38]. Para lograr este objetivo existen variedad de técnicas y métodos que están actualmente en desarrollo. En particular, el diseño de BCIs a partir de potenciales evocados visuales de estado estable (o SSVEPs por su acrónimo en inglés) es una técnica que ha demostrado ser efectiva para desarrollar dispositivos intuitivos, fáciles de usar y con altas tasas de transmisión de la información [39], [40].

Estas técnicas se basan en un fenómeno que aparece en el registro electroencefalográfico (EEG) cuando una persona observa estímulos visuales periódicos con frecuencias superiores a 6 Hz. En estas condiciones, es posible detectar en el EEG una señal con la misma periodicidad que el estímulo visual. Asociando distintos comandos a distintas señales lumínicas, donde cada una parpadea a distintas frecuencias o en distintas fases, se puede generar un sistema que solo se valga del enfoque visual del usuario para controlar algún dispositivo. En la Figura 1.3 se muestra esquemáticamente un ejemplo de aplicación de esta técnica en un sistema simple con dos luces parpadeantes y dos comandos.

Esta aplicación tiene características ideales para la implementación de la técnica de detección coherente, ya que la señal evocada en el registro EEG es modulada por el mismo sistema, mediante



Figura 1.3: Ejemplo esquemático de una implementación de BCI basada en SSVEPs. El usuario puede mirar a dos luces parpadeantes, de 14 o 18 Hz. Según la periodicidad que se detecte en su registro EEG se activa un comando on/off.

estimulación lumínica en la frecuencia que se pretende detectar y, una vez medida, debe someterse a un procesamiento coherente, que detecte la presencia o ausencia de componentes en la frecuencia objetivo. En esta aplicación, la frecuencia de excitación y detección no es particularmente alta, sino en el orden de los Hz, pero es deseable que el sistema pueda realizar la detección coherente de señales en varias frecuencias simultáneamente, lo que se conoce como detección coherente multifrecuencial. Además, es deseable que la adquisición de la señal y la estimulación visual estén perfectamente sincronizadas, para facilitar la detección de los SSVEPs [41].

1.5. Espectroscopía Mössbauer

La Espectroscopía Mössbauer, también conocida como absorción resonante de rayos gamma, es una técnica que utiliza la emisión y absorción de fotones en núcleos de átomos determinados para estudiar características de los materiales. En esta técnica el núcleo de un átomo, en su estado excitado o radioactivo emite un fotón, que en ciertas circunstancias será absorbido por el núcleo de un átomo del mismo tipo, en estado absorbente o fundamental. Esta absorción, llamada absorción resonante, puede ser destruida si se cambia ligeramente la velocidad relativa entre los núcleos de los átomos. Aprovechando este fenómeno, la velocidad del núcleo excitado se varía en intervalos discretos y se analizan las mediciones del detector para cada velocidad. Al estudiar las velocidades en las que se produce la absorción resonante, es posible determinar características de los materiales.

Para que de esta operación surja un espectro de calidad adecuada es fundamental conocer la relación canal a velocidad del sistema de medida. Tradicionalmente, esta relación se obtiene a partir de un enfoque empírico, buscando que un espectro conocido tenga una relación canal a velocidad correcta [42],[43]. Sin embargo, las desviaciones de esta relación de la esperada son casi imposibles de predecir, y varían considerablemente cuando se modifican los parámetros de la velocidad o el sistema de control de lazo cerrado.

La velocidad de la fuente radiactiva suele ser controlada por un servo amplificador, del cual se distinguen dos señales de control: la señal Monitor, que es proporcional a la velocidad requerida en una bobina secundaria y la señal de Error, que es proporcional a la diferencia entre la tensión real en la bobina secundaria y la tensión requerida. Si se pueden medir estas señales en tiempo real, manteniendo el sincronismo y la resolución necesarias dadas por la aplicación, se puede determinar la relación canal a velocidad en tiempo real, mejorando los espectros obtenidos.

Las señales de Error y Monitor son periódicas, de frecuencias bajas (en el orden de 20 Hz), y para medirlas con la resolución deseada es conveniente contar con capacidades de promediación coherente. En un trabajo reciente [44], se realizó el muestreo de las formas de onda Error y Monitor con un osciloscopio digital estándar, manteniendo la sincronización y resolución necesarias para un análisis posterior. Aunque los osciloscopios tienen capacidades de promediación coherente, útiles para la adquisición de estas señales, los requisitos de este experimento exceden sus capacidades habituales, por lo que este procedimiento, que ha demostrado su utilidad, resulta una tarea laboriosa y temporalmente demandante.

Es en este punto donde encuentran utilidad los sistemas de detección coherente de señales. En esta aplicación, sin embargo, no es tan importante la velocidad a la que se muestrean las señales, sino su resolución. Además, este sistema debe poder recuperar, al menos, todo un periodo de la señal objetivo, en lugar de simplemente extraer la magnitud de esta señal del ruido circundante. Como este es un sistema con aplicaciones en laboratorios de Espectroscopía Mössbauer es deseable que el sistema sea versátil y fácilmente manipulable por investigadores no especializados en instrumentación electrónica. Si bien la elección de un sistema SoC-FPGA puede parecer sobre di-

mensionada para esta aplicación en particular, la incorporación de una herramienta de este tipo abre muchas áreas de trabajo a futuro en esta técnica.

1.6. Propuesta para un sistema de detección coherente en SoC-FPGA

Con estas aplicaciones en mente, y buscando aprovechar las características de los dispositivos SoC-FPGA, se decidió avanzar en la implementación de un sistema de detección coherente de señales reconfigurable, con capacidades para ser integrado en los distintos proyectos de interés.

Las características de las aplicaciones deseadas determinan algunas de las características del sistema a implementar. Este debe poder demodular señales de frecuencias desde unos pocos Hz hasta frecuencias en el orden de los cientos de kHz, para poder detectar la presencia de nanopartículas ferromagnéticas. También debe poder operar en múltiples frecuencias simultáneamente, para poder implementar con él un sistema de BCI de múltiples comandos, y debe poder recuperar periodos enteros de la señal con resoluciones adecuadas, para poder estudiar las señales de Error y Monitor del servoamplificador utilizado en la técnica de Espectroscopía Mössbauer. Además es deseable que el sistema sea fácil de utilizar, que sus parámetros de operación sean configurables en tiempo real y que la implementación sea flexible, de manera tal de poder adaptarla a distintos dispositivos de lógica programable y a las distintas aplicaciones.

1.7. Resumen de publicaciones producidas

Durante el desarrollo de este trabajo de tesis han surgido publicaciones en revistas y congresos, así como repositorios de hardware y software libre. Estos serán referenciados oportunamente a lo largo del texto y se presentan a continuación como resumen.

1.7.1. Trabajos publicados en revistas

- Matías J. Oliva et al. «Improving the computational efficiency of lock-in algorithms through coherent averaging». En: *Digital Signal Processing* 154 (2024), pág. 104693. ISSN: 1051-2004. DOI: https://doi.org/10.1016/j.dsp.2024.104693.
- [2] Matías Javier Oliva, Gustavo Pasquevich y Alejandro Luis Veiga. «Embedded device for digitalizing monitor and error signals in Mössbauer Spectroscopy». En: *Hyperfine Interact* 244.23 (2023). DOI: 10.1007/s10751-023-01834-5.
- [3] Matías Oliva et al. «Real-Time SSVEP Measurements Through Lock-In Detection in FPGA-Based Platform». En: Advances in Bioengineering and Clinical Engineering. Ed. por Fernando Emilio Ballina et al. Cham: Springer Nature Switzerland, 2024, págs. 161-171. ISBN: 978-3-031-61960-1. DOI: 10.1007/978-3-031-61960-1_16.
- [4] M. J. Oliva et al. «SoC-FPGA systems for the acquisition and processing of electroencephalographic signals». En: International Journal of Reconfigurable and Embedded Systems (IJRES) 10.3 (nov. de 2021), págs. 237-248. DOI: 10.11591/ijres.v10.i3.pp237-248.
- [5] M. J. Oliva, P. A. García y E. M. Spinelli. «A System on Chip based electroencephalogram acquisition system». En: *Revista Argentina de Bioingeniería* 24.3 (2020). Presentado en Congreso Argentino de Bioingeniería 2020.

1.7.2. Trabajos publicados en congresos

- [1] Matías Javier Oliva et al. «Open-source SoC-FPGA Platform for Signal Processing». En: Southern Programmable Logic Conference 2023. San Luis, Argentina, mar. de 2023.
- Federico N. Guerrero, Matías Javier Oliva y Enrique Mario Spinelli. «Real-time noise reduction through independent channel averaging for real-time biomedical signal acquisition».
 En: Southern Programmable Logic Conference 2023. San Luis, Argentina, mar. de 2023.
- [3] Matías Javier Oliva et al. «Implementation of a digital Lock-in amplifier in SoC-FPGA for Biomedical Tomography». En: Congreso Argentino de Bioingeniería 2022. San Juan, Argentina, sep. de 2022.
- [4] P. A. García et al. «Bioimpedanciómetro basado en microcontrolador de 32 bits». En: Congreso Argentino de Bioingeniería 2022. San Juan, Argentina, sep. de 2022.
- [5] P. García, E. Spinelli y M. Oliva. «System on Chip based brain-computer interface». En: Congreso Argentino de Control Automático 2020. Buenos Aires, Argentina, oct. de 2020.
- [6] M. J. Oliva et al. «Interfaz cerebro computadora basada en SoC». En: Congreso Argentino de Sistemas Embebidos 2020. Buenos Aires, Argentina, ago. de 2020.
- [7] M. Haberman et al. «A wearable low-power and low-cost electromyographic sensor for arm prosthesis». En: Congreso Argentino de Bioingeniería 2020. Piriapolis, Uruguay, mar. de 2020.
- [8] Matías Javier Oliva, Pablo Andrés García y A. L. Veiga. «Evaluación del desempeño de sistemas embebidos heterogéneos para aplicaciones de tomografía magnética». En: Jornadas de Investigación y Transferencia UNLP. La Plata, Argentina, mar. de 2019.
- [9] M. J. Oliva et al. «Holter digital electrocardiográfico». En: Simposio Argentino de Sistemas Embebidos (SASE). Córdoba, Argentina, ago. de 2018.

1.7.3. Repositorios de código abierto

- [1] Matías Oliva. *Repositorio con recursos para probar el algoritmo CALI*. https://github. com/ushikawa93/lockin_coherent_average. 2023.
- [2] Matias Oliva. Signal processing in FPGA. https://github.com/ushikawa93/signal_ processing_fpga.git. 2022.
- [3] Matías Oliva. Signal Processing Red Pitaya. https://github.com/ushikawa93/signal_ processing_red_pitaya. 2024.
- [4] Matías Oliva. Funcionalidades para controlar un lockin SR865 de Stanford en forma remota. https://github.com/ushikawa93/sr865_software. 2024.
- [5] Matías Oliva. Lock-in en DE1-SoC. https://github.com/ushikawa93/lockin_de1soc. 2024.
- [6] Matías Oliva. Lock-in en Red Pitaya. https://github.com/ushikawa93/lockin_redpitaya. 2024.
- [7] Matías Javier Oliva. SSVEP FPGA ADS1299. https://github.com/ushikawa93/ssvep_ fpga_ads1299. 2024.
- [8] M.J. Oliva. Adquisidor en Red Pitaya. https://github.com/ushikawa93/adquisidor_ red_pitaya. 2024.

1.8. Estructura de la tesis

En este capítulo han presentado las principales motivaciones y objetivos de este trabajo de investigación. En el capítulo 2 se describen en detalle los sistemas de detección coherente de señales, incluyendo resultados teóricos conocidos y originales. En el capítulo 3 se describen los dispositivos SoC-FPGA, y las plataformas de procesamiento de señales que se han desarrollado para éstos durante esta tesis, mientras que en el capítulo 4 se describe la implementación de sistemas de detección coherente en SoC-FPGA, comparando los resultados obtenidos con la bibliografía disponible, y con sistemas comerciales similares. El capítulo 5 muestra las aplicaciones de los sistemas desarrollados. Este se divide en tres partes: el desarrollo del tomógrafo magnético, la implementación de una interfaz cerebro computadora basada en potenciales evocados visuales y la incorporación de los sistemas desarrollados a la calibración de un sistema de Espectroscopía Mössbauer. Finalmente, en el capítulo 6 se discuten los resultados obtenidos, se plantean conclusiones y se delinea el trabajo que se pretende desarrollar a futuro.

Capítulo 2

Técnicas de detección coherente

2.1. Introducción

Medir señales de amplitudes pequeñas perturbadas por ruido es un desafío típico en instrumentación electrónica, que aparece en áreas tan diversas como las telecomunicaciones, la bioingeniería o el control automático. Un parámetro esencial para la descripción de este tipo de sistemas es la relación señal a ruido (SNR), que cuantifica la amplitud de la señal de interés frente al ruido en la que está inmersa. En sistemas en donde se conoce la amplitud o la potencia de la señal y del ruido esta magnitud puede calcularse mediante las fórmulas 2.1 y 2.2.

$$SNR_{dB} = 20 \log_{10} \left(\frac{A_{señal}}{A_{ruido}} \right)$$
(2.1)

$$SNR_{dB} = 10 \log_{10} \left(\frac{P_{señal}}{P_{ruido}} \right)$$
(2.2)

Cuanto menor sea esta magnitud, es decir, cuanto más ruido haya en el sistema, más complicado será detectar la señal deseada. Sin embargo, si se conoce o al menos se puede estimar la frecuencia de oscilación de la señal desconocida, o si esta se puede modular a una frecuencia conocida, pueden aplicarse técnicas de procesamiento coherente para posibilitar la tarea. Entre éstas existen aquellas que aplican algoritmos que no realimentan la señal de salida, comúnmente llamadas técnicas de lazo abierto, y aquellas que utilizan los datos de salida para realimentar y robustecer los cálculos, conocidas como técnicas de lazo cerrado [45].

Las técnicas de lazo cerrado utilizan algoritmos recursivos para minimizar el error entre la información modelada y la medida, con el objetivo de estimar una variable desconocida. Existen distintos algoritmos para optimizar este proceso, entre los cuales se pueden mencionar el filtro de Kalman [46], [47], [48], el filtro de Lyapunov[49], [50], [51] y las técnicas de diseño directo [52].

Entre las técnicas de lazo abierto se encuentran el *Lock-in* y la promediación coherente, que son de las más estudiadas en la bibliografía disponible. El estudio y desarrollo de estas técnicas es el que ha impulsado este trabajo de investigación, y en el que se centra esta tesis. En este capítulo se presentan en detalle, incluyendo antecedentes, resultados conocidos y originales.

Gran parte de los resultados que se muestran en este capítulo fueron presentados en [53] y [54].

2.2. Lock-in

El amplificador *Lock-in* es una herramienta muy utilizada en instrumentación científica. Su implementación permite obtener información de señales de banda estrecha moduladas a una frecuencia determinada, f_c . Para ello se utiliza una señal periódica de referencia, que puede ser cuadrada o sinusoidal, de la misma frecuencia, y opcionalmente una señal de referencia desfasada 90°. Este último paso divide a esta técnica en *Lock-in* de una fase (o canales) y de dos fases.

2.2.1. Lock-in de una fase

La Figura 2.1 muestra la forma más elemental de aplicar el concepto de Lock-in. Se dispone de una señal de referencia r(t), que oscila a la frecuencia f_c , y una señal desconocida s(t), que puede ser la modulación de una señal desconocida a la frecuencia f_c o la respuesta de un sistema cuando es excitado por una señal de esta frecuencia.

$$r(t) = A_r \operatorname{sen}(\omega_c t + \phi_r)$$

$$s(t) = A_s \operatorname{sen}(\omega_c t + \phi_s)$$

donde $\omega_c = 2\pi f_c$. La señal s(t) se multiplica por la referencia, obteniendo la señal $X_f(t)$, que puede expresarse como:

$$X_{f}(t) = \frac{A_{s}A_{r}}{2} \cdot (\cos(\phi_{r} - \phi_{s}) - \cos(2\omega_{c}t + \phi_{r} + \phi_{s}))$$
(2.3)

y consta de dos términos: un término de continua y uno que oscila al doble de la frecuencia f_c . Esta señal ingresa a un filtro pasa-bajos que, suponiendo que puede rechazar totalmente el término de frecuencia $2f_c$, proporciona un salida dada por:

$$X = \frac{A_s A_r}{2} \cdot \cos(\phi_r - \phi_s) \tag{2.4}$$

Este resultado depende de la diferencia entre la fase de la señal de referencia y la señal desconocida, y alcanza su máximo valor cuando ambas fases coinciden. Como la referencia es una señal accesible para el usuario se puede ajustar su fase hasta encontrar el máximo en la señal X, X_{max} , para finalmente obtener la amplitud de la señal desconocida como:

$$A_s = 2 \cdot \frac{X_{max}}{A_r} \tag{2.5}$$

Si bien este método permite recuperar correctamente la amplitud de la señal desconocida, su aplicación implica ajustar manualmente la fase de la referencia, dificultando la medición, e incorporando una fuente de error adicional. Estos problemas pueden solucionarse utilizando una señal de referencia extra, desfasada 90° de la original, como se verá en la próxima sección.

2.2.2. Lock-in de dos fases

Al agregar al sistema una señal de referencia desfasada 90° se puede eliminar la dependencia de los resultados con el desfasaje de las señales. Considerando las señales indicadas en la Figura 2.2 se pueden distinguir:

$$r_s(t) = A_r \operatorname{sen}(\omega_c t + \phi_r)$$
$$r_c(t) = A_r \cos(\omega_c t + \phi_r)$$
$$s(t) = A_s \operatorname{sen}(\omega_c t + \phi_s)$$



Figura 2.1: Lock-in de una fase

La señal desconocida es multiplicada por la señal de referencia (comúnmente llamada referencia en fase), y por la señal desfasada 90° (conocida como referencia en cuadratura), para luego aplicar sendos filtros pasa-bajos a las señales obtenidas, en un procedimiento análogo al descripto para el *Lock-in* de una fase.

$$X_f(t) = \frac{A_s A_r}{2} \cdot \left(\cos(\phi_r - \phi_s) - \cos(2\omega_c t + \phi_r + \phi_s)\right)$$
$$Y_f(t) = \frac{A_s A_r}{2} \cdot \left(\sin(\phi_r - \phi_s) - \sin(2\omega_c t + \phi_r + \phi_s)\right)$$

$$X = \frac{A_s A_r}{2} \cdot \cos(\phi_r - \phi_s) \tag{2.6}$$

$$Y = \frac{A_s A_r}{2} \cdot \operatorname{sen}(\phi_r - \phi_s) \tag{2.7}$$

De donde se pueden despejar tanto la amplitud como la fase de la señal desconocida:

$$A_s = \frac{2}{A_r} \sqrt{X^2 + Y^2}$$
(2.8)

$$\phi_s = \frac{2}{A_r} \arctan\left(\frac{Y}{X}\right) \tag{2.9}$$

Con este sencillo procedimiento es posible eliminar la necesidad de ajustar manualmente la fase de la referencia, evitando una posible causa de errores, y contribuyendo a la automatización del proceso, a cambio de duplicar los recursos, ya sea digitales o analógicos, necesarios para la implementación del *Lock-in*.



Figura 2.2: Lock-in de dos fases

2.2.3. Análisis en el Dominio Complejo

Las deducciones dadas en las secciones 2.2.1 y 2.2.2 pueden presentarse en forma más compacta considerando señales en el plano complejo. Para ello se tomará una referencia compleja dada por:

$$r(t) = A_r e^{j(\omega_c t + \phi_r)}$$

= $A_r \cos(\omega_c t + \phi_r) + j A_r \sin(\omega_c t + \phi_r)$

Por simplicidad se considerará una señal de entrada cosenoidal, que puede representarse en notación compleja como:

$$s(t) = A_s \cos(\omega_c t + \phi_s)$$
$$= \frac{A_s}{2} (e^{j(\omega_c t + \phi_s)} + e^{-j(\omega_c t + \phi_s)})$$

Ahora esta señal se multiplica por la referencia compleja, obteniendo:

$$s(t)r(t) = \frac{A_s A_r}{2} (e^{j(\omega_c t + \phi_s)} + e^{-j(\omega_c t + \phi_s)}) e^{j(\omega_c t + \phi_r)}$$
$$= \frac{A_s A_r}{2} (e^{j2\omega_c t} e^{j(\phi_r + \phi_s)} + e^{j(\phi_r - \phi_s)})$$

Y es luego sometida a un pasa-bajos, eliminando el término del doble de frecuencia, obteniendo:

$$P_B\{s(t)r(t)\} = \frac{A_s A_r}{2} e^{j(\phi_r - \phi_s)} = \frac{A_s A_r}{2} (\cos(\phi_r - \phi_s) + j \sin(\phi_r - \phi_s))$$
(2.10)

donde se ha simbolizado la operación de pasa-bajos con el operador P_B {}. Tomando la parte real y la parte imaginaria de esta última ecuación se llega a las mismas expresiones de X e Y dadas en la sección anterior:

$$Re\{P_B\{s(t)r(t)\}\} = X = \frac{A_s A_r}{2} \cdot \cos(\phi_r - \phi_s)$$
$$Im\{P_B\{s(t)r(t)\}\} = Y = \frac{A_s A_r}{2} \cdot \sin(\phi_r - \phi_s)$$

de donde pueden deducirse las expresiones 2.8 y 2.9, que corresponden al módulo y la fase de la magnitud compleja dada por la ecuación 2.10.

Es interesante notar, además, que en la condición $\phi_s = \phi_r$ la parte real de este resultado es máxima, igual a $\frac{A_s A_r}{2}$ y la parte imaginaria de el resultado es igual a 0. Este es el caso que se busca ajustando la fase de la referencia hasta que el resultado sea máximo cuando se hace *Lock-in* de una fase, de donde se desprende que el *Lock-in* de una fase es un caso particular del *Lock-in* de dos fases.

2.2.4. Efecto sobre el ruido blanco

Para abordar la pregunta de cómo este esquema de procesamiento de señales contribuye a reducir el ruido en la salida del sistema, es crucial delinear las características del ruido que lo afecta. En esta sección se considerará el efecto del ruido blanco n(t), definido como un proceso aleatorio estacionario en sentido amplio (PAESA), con distribución gaussiana, media nula y varianza σ^2 , es decir, un proceso cuyas muestras siguen una distribución $\mathcal{N}(0, \sigma^2)$. Además, se asumirá que las muestras del proceso en distintos momentos no están correlacionadas entre sí, por lo que su función de autocorrelación se puede modelar mediante la función generalizada conocida como delta de Dirac $(\delta(\tau))$ y la desviación estándar del ruido (σ_i) según:

$$R_{nn}(\tau) = \sigma_i^2 \delta(\tau) \tag{2.11}$$

La densidad espectral de potencia de este proceso puede obtenerse tomando la transformada de Fourier de esta función de autocorrelación, como establece el teorema de Wiener-Khinchin [55], [56]:

$$S_{nn}(f) = \int_{-\infty}^{\infty} R_{nn}(\tau) e^{-j2\pi f\tau} d\tau$$
$$= \int_{-\infty}^{\infty} \sigma_i^2 \delta(\tau) e^{-j2\pi f\tau} d\tau = \sigma_i^2$$

con lo que se llega a una definición alternativa del ruido blanco: un proceso aleatorio con densidad espectral de potencia (DEP) constante para todas las frecuencias:

$$S_{nn}(f) = \sigma_i^2 = \frac{N_0}{2} \qquad \forall f$$

donde los términos $\frac{N_0}{2}$ y σ_i^2 , se utilizarán en forma equivalente a lo largo de este texto.

En este punto vale una aclaración: este tipo de proceso es en realidad una abstracción matemática, ya que la potencia media de una señal como esta sería infinita, por lo que no tiene sentido físico. Esto puede mostrarse integrando esta densidad espectral sobre todas las frecuencias:

$$\langle p \rangle = \int_{-\infty}^{\infty} S_{nn}(f) df = \int_{-\infty}^{\infty} \frac{N_0}{2} df \longrightarrow \infty$$
 (2.12)

Sin embargo, esta abstracción permite introducir el concepto de ruido blanco limitado en banda, definido como un proceso cuya densidad espectral de potencia es constante e igual a $\frac{N_0}{2}$ dentro de un cierto ancho de banda B_n , y nula fuera de él:

$$S_{nn}(f) = \begin{cases} \frac{N_0}{2} & |f| < B_n \\ 0 & \text{en otro caso} \end{cases}$$
(2.13)

Ahora sí, la potencia media de este proceso es finita:

$$\langle p \rangle = \int_{-\infty}^{\infty} S_{nn}(f) \, df = \int_{-B_n}^{B_n} \frac{N_0}{2} \, df = N_0 \cdot B_n$$
 (2.14)

Se supondrá entonces que la señal original, s(t), está formada por la señal de interés $s_0(t)$ y por ruido blanco limitado en banda n(t):

$$s(t) = s_0(t) + n(t) \tag{2.15}$$

Esta es la señal que ingresa al Lock-in. En el análisis que sigue se considerará $s_0(t) = 0$, sin perder generalidad gracias al principio de superposición. Se considerará el caso de Lock-in de dos fases ya que, como se ha visto en la sección 2.2.3, el Lock-in de una fase es un caso particular de éste.

La señal s(t) es multiplicada por una exponencial compleja $(e^{j2\pi f_c t}, \text{ considerando 0 la fase de la referencia por simplicidad}) y luego sometida a un pasa-bajos, para finalmente tomar el módulo del resultado (equivalente a la ecuación 2.8) y su fase (equivalente a la ecuación 2.9). Primero se considerará la multiplicación, que provoca un desplazamiento de las componentes del ruido en la frecuencia. Para analizar esto se planteará la función de autocorrelación de la señal multiplicada <math>n_{mult}(t) = n(t)e^{j2\pi f_c t}$. Como n(t) es un proceso PAESA esta autocorrelación está dada por:

$$R_{n_{mult}n_{mult}}(\tau) = E\{n_{mult}(t)n_{mult}^{*}(t+\tau)\}$$

= $E\{n(t)e^{j2\pi f_{c}t}n^{*}(t+\tau)e^{-j2\pi f_{c}(t+\tau)}\}$
= $E\{n(t)n^{*}(t+\tau)e^{-j2\pi f_{c}\tau}\}$
= $R_{nn}(\tau)e^{-j2\pi f_{c}\tau}$ (2.16)

donde el superíndice * denota el conjugado de la magnitud compleja. Conociendo el par transformado $R_{nn} \leftrightarrow S_{nn}(f)$, y aplicando la propiedad de traslación de la transformada de Fourier:

$$f(t)e^{j2\pi\alpha t} \Longleftrightarrow F(f-\alpha) \tag{2.17}$$

puede verse que la densidad espectral de potencia de la señal multiplicada $n(t)e^{j2\pi f_c t}$ sigue siendo $\frac{N_0}{2}$, pero ahora centrada en $f + f_c$:

$$S_{mult}(f) = S_{nn}(f + f_c) = \begin{cases} \frac{N_0}{2} & -B_n < f + f_c < B_n \\ 0 & \text{En otro caso} \end{cases}$$
$$S_{mult}(f) = \begin{cases} \frac{N_0}{2} & -B_n - f_c < f < B_n - f_c \\ 0 & \text{En otro caso} \end{cases}$$
(2.18)

Es decir que el ruido que se encontraba en las inmediaciones de la frecuencia f_c termina en banda base, y será este el ruido que aparecerá en la salida del sistema luego del filtrado pasa-bajos. Con esto ya se puede ver que la acción conjunta de mezclador y filtro será equivalente a un pasa-banda centrado en la frecuencia de interés f_c .

Llegados a este punto es necesario delinear las propiedades del filtro pasa-bajos que se utilizará. Considerando a este filtro como lineal e invariante en el tiempo, con una función de transferencia H(f), la densidad espectral de potencia en su salida será:

$$S_o(f) = |H(f)|^2 S_{mult}(f)$$
 (2.19)

Si además el filtro es ideal, es decir que su función de transferencia es un cajón de una cierta ganancia K de ancho Δf :

$$H(f) = \begin{cases} K & -\frac{\Delta f}{2} < f < \frac{\Delta f}{2} \\ 0 & \text{En otro caso} \end{cases}$$
(2.20)

se puede calcular la densidad espectral de ruido a la salida del Lock-in según:

$$S_o(f) = \begin{cases} K^2 \frac{N_0}{2} & -\frac{\Delta f}{2} < f < \frac{\Delta f}{2} \\ 0 & \text{En otro caso} \end{cases}$$
(2.21)

donde se supuso que el ancho de banda donde el ruido puede considerarse blanco incluye a la banda de paso del filtro.

A partir de este resultado se puede calcular la autocorrelación de ruido a la salida haciendo uso nuevamente del teorema de Wiener-Khinchin. Luego considerando que en el ruido blanco limitado en banda no hay correlación entre los valores de señal en dos momentos distintos, la potencia media se puede obtener evaluando esta autocorrelación en $\tau = 0$.

$$R_{xx}(\tau) = \int_{-\infty}^{\infty} S_o(f) e^{-j2\pi f\tau} df \qquad (2.22)$$
$$= \int_{-\frac{\Delta f}{2}}^{\frac{\Delta f}{2}} K^2 \frac{N_0}{2} e^{-j2\pi f\tau} df = \frac{N_0}{2} K^2 \frac{e^{j2\pi \frac{\Delta f}{2}\tau} - e^{-j2\pi \frac{\Delta f}{2}\tau}}{j2\pi\tau}$$
$$= \frac{N_0}{2} K^2 \Delta f \frac{e^{j\pi \Delta f\tau} - e^{-j\pi \Delta f\tau}}{2j\pi \Delta f\tau} = \frac{N_0}{2} K^2 \Delta f \frac{f \operatorname{sen}(\pi \Delta f\tau)}{\pi \Delta f\tau}$$

Definiendo la función $\operatorname{sinc}(x) = \frac{\operatorname{sen}(x)}{x}$, que tiene la propiedad de aproximarse infinitamente a 1 cuando x = 0, se puede sintetizar este resultado y obtener la potencia media de ruido a la salida:

$$R_{xx}(\tau) = \frac{N_0}{2} K^2 \Delta f \quad \operatorname{sinc}(\Delta f \pi \tau)$$
$$R_{xx}(0) = K^2 \frac{N_0}{2} \Delta f = K^2 \frac{N_0}{2} \Delta f \qquad (2.23)$$

Lo cual lleva al resultado:

$$\sqrt{\frac{R_{xx}(0)}{\frac{N_0}{2}}} = \frac{\sigma_o}{\sigma_i} = K\sqrt{\Delta f}$$
(2.24)

Si bien este es un análisis simplificado, permite discernir algunos aspectos de la acción del Lockin. La desviación estándar del ruido en la salida disminuye proporcionalmente con la raíz cuadrada del ancho de banda del filtro pasa-bajos utilizado, con lo cual la elección de éste resulta un factor clave en su diseño. Con filtros más selectivos se obtiene una mayor reducción del ruido a la salida. Idealmente si se pudiese lograr $\Delta f = 0$ se rechazaría todo el ruido a la salida, aunque esto implicaría integrar la señal durante un tiempo infinito.

2.2.5. Elección del filtro pasa-bajos

Como muestra el análisis anterior, la combinación de multiplicación y filtro pasa-bajos se comporta como un pasa-banda centrado en ω_c , sin muchas de las dificultades asociadas con el diseño de filtros pasa banda estrechos [57]. Esto implica que un elemento clave en la amplificación *Lock-in* es la selección del filtro pasa bajos [58]. Este cumple con dos propósitos: en primer lugar, debe eliminar la componente de frecuencia doble en la señal multiplicada. En segundo lugar, se encarga de reducir el ruido que se involucra en los cálculos.

2.2.5.1. Filtro de media móvil

Una opción interesante para la técnica *Lock-in* es un filtro de media móvil simple (o MAF, del inglés *moving average filter*) de largo T. La operación de este filtro sobre una entrada continua arbitraria x(t) puede describirse matemáticamente mediante su respuesta impulsional, dada por 2.25.

$$h(t) = \begin{cases} \frac{1}{T} & -\frac{T}{2} < t < \frac{T}{2} \\ 0 & \text{En otro caso} \end{cases}$$
(2.25)

En el dominio frecuencial esta operación resulta:

$$H(\omega) = \int_{-\frac{T}{2}}^{\frac{T}{2}} \frac{1}{T} e^{-j\omega t} d\omega = \frac{1}{T} \frac{e^{-j\frac{T}{2}\omega} - e^{j\frac{T}{2}\omega}}{-j\omega} = \frac{\operatorname{sen}(\frac{\omega T}{2})}{\frac{\omega T}{2}} = \operatorname{sinc}\left(\frac{\omega T}{2}\right)$$
(2.26)

Los ceros de la función sinc se dan cuando su argumento es igual a $m\pi$, con $m \neq 0$:

$$\frac{\omega_m T}{2} = m\pi \to \omega_m = \frac{m2\pi}{T} \tag{2.27}$$

Es decir que siempre que $f = m_{\overline{T}}^1$ la función presentará un cero. Si se elige T como un múltiplo entero del periodo de la función de referencia $(T = k_{\overline{f_c}}^1)$ resulta que los ceros caen en $f = kmf_c \rightarrow$ $f = \alpha f_c$, es decir en múltiplos enteros de la frecuencia de interés f_c . Esto resulta especialmente interesante para la acción del *Lock-in*, que debe bloquear el componente del doble de frecuencia generado por el mezclador. En el caso del *Lock-in* con MAF este bloqueo será total, ya que el filtro presenta un cero en esta frecuencia.

Para este filtro, el ancho de banda es inversamente proporcional al largo T ($\Delta f = \frac{\alpha}{T}$). Esto implica que si se utiliza en la técnica *Lock-in*, el ruido de medición puede reducirse promediando un mayor número de periodos de la señal [59], [60]. Para un análisis más detallado de esto se examinará qué ocurre con el ruido luego de la acción conjunta del mezclador y el filtro MAF. Para ello se partirá de la ecuación 2.19 y se calculará la DEP de ruido a la salida para determinar la potencia de ruido:

$$S_{o}(f) = |H(f)|^{2} S_{mult}(f) = \frac{N_{0}}{2} \operatorname{sinc}^{2} \left(\frac{\omega T}{2}\right)$$
$$R_{nn}(\tau) = \frac{N_{0}}{2} F^{-1} \{\operatorname{sinc}^{2} \left(\frac{\omega T}{2}\right)\} = \frac{N_{0}}{2} \frac{2}{T} F^{-1} \{\operatorname{sinc}^{2} (\omega)\} = \frac{N_{0}}{2} \frac{2}{T} \operatorname{Tri}(\tau)$$
$$R_{nn}(0) = P_{o} = \frac{N_{0}}{2} \frac{2}{T}$$
(2.28)

Donde se ha tenido en cuenta el par transformado $\operatorname{sinc}^2(\omega) \leftrightarrow \operatorname{Tri}(t)$, con $\operatorname{Tri}(x)$ representando la función triangular, dada por:

$$\operatorname{Tri}(x) = \begin{cases} 1 - |x| & |x| \le 1\\ 0 & \text{En otro caso} \end{cases}$$

Ahora se puede relacionar el ruido en la entrada con el ruido a la salida del sistema:

$$\frac{\sigma_o^2}{\sigma_i^2} = \frac{\frac{N_0}{2}\frac{2}{T}}{\frac{N_0}{2}} \longrightarrow \sigma_o = \frac{\sigma_i}{\sqrt{\frac{T}{2}}}$$
(2.29)

Por lo que con MAF de largo T el *Lock-in* reduce la potencia de ruido en un factor $\sqrt{\frac{T}{2}}$.

Todo este análisis se ha hecho para señales analógicas. En sistemas digitales, muestreados a una frecuencia de muestreo f_m $(T_m = \frac{1}{f_m})$, el filtro MAF de un número entero de ciclos de la señal puede describirse matemáticamente según:

$$Y_{[n]} = \frac{1}{NM} \sum_{k=n}^{n+NM-1} x_{[k]}$$
(2.30)

donde N es la cantidad de ciclos de la señal que se integran y M la cantidad de muestras que entran en un ciclo de la señal periódica. Por simplicidad se llamará en adelante $N_T = NM$. Al igual que su contraparte analógica, este filtro tiene la propiedad de presentar ceros en frecuencias que son múltiplos de la frecuencia de la señal, siempre y cuando se integren un número entero de periodos [61], por lo que puede bloquear completamente la componente $2\omega_c$. Esta propiedad se puede deducir de manera similar, partiendo de su transformada de Fourier en tiempo discreto (TFTD):

$$H(e^{j\omega}) = \frac{1}{N_T} \frac{1 - e^{-j\omega N_T}}{1 - e^{-j\omega N_T}} = e^{-j\omega \frac{N_T - 1}{2}} \frac{\operatorname{sen}(\frac{\omega N_T}{2})}{N_T \operatorname{sen}(\frac{\omega}{2})}$$
(2.31)

donde ω representa en realidad la frecuencia normalizada dada por $\omega = \frac{2\pi f}{f_m}$. Esta última expresión guarda una estrecha relación con la función sinc, con un desplazamiento en la frecuencia dada por su exponencial compleja. Si a un filtro como éste ingresa una secuencia discreta $n_{[k]}$ con densidad espectral de potencia igual a $\frac{N_0}{2}$ la DEP a su salida estará dada por:

$$S_{yy}(e^{j\omega}) = \frac{N_0}{2} |H(e^{j\omega})|^2 = \frac{N_0}{2} \left| \frac{\operatorname{sen}(\frac{\omega N_T}{2})}{N_T \operatorname{sen}(\frac{\omega}{2})} \right|^2$$
(2.32)

La antitransformada de esta TFTD es la secuencia triangular. Esto resulta lógico, ya que la multiplicación de dos transformadas de una función cajón es la convolución de ellas en el dominio temporal, y la convolución de dos cajones es una función triangular. Esto enfatiza la relación entre la expresión dada en la ecuación 2.31 y la función sinc en el tiempo continuo, pero eso está más allá de las pretensiones de este texto. Antitransformando $S_{yy}(e^{j\omega})$ se obtiene la autocorrelación del ruido y puede calcularse la potencia en la salida del *Lock-in* con MAF digital:

$$R_{yy[n]} = \frac{N_0}{2} \frac{2}{N_T} tri\left(\frac{n}{N_T}\right) \longrightarrow R_{yy[0]} = \frac{N_0}{2} \frac{2}{NM}$$
(2.33)

Finalmente relacionando la potencia de ruido en la entrada y en la salida del sistema puede llegarse a la ecuación 2.34, que describe como resulta la potencia de ruido en relación al ruido de entrada para la acción combinada de mezclador y filtro MAF.

$$\sigma_o = \frac{\sigma_i}{\sqrt{\frac{NM}{2}}} \tag{2.34}$$

Las ecuaciones 2.34 y 2.29 guardan una estrecha relación, con T cumpliendo el mismo rol de tiempo de integración que NM. Una opción alternativa para estimar la relación entre el ruido a la entrada y a la salida del sistema es tomar un enfoque empírico, determinando mediante simulaciones la constante K que cumple:

$$\sigma_o = \sigma_i \frac{K}{\sqrt{NM}} \tag{2.35}$$



Figura 2.3: Estimación de parámetro k en ecuación 2.35

Para ello se parte de una señal sinusoidal de amplitud conocida, se perturba con ruido blanco, y se estima la amplitud de la señal subyacente utilizando la expresión 2.8. Este proceso se repite N veces y se toma la desviación estándar de las amplitudes calculadas, obteniendo un estimador de la potencia de ruido presente en la señal de salida del amplificador *Lock-in*, σ_o . Este proceso se repite para valores crecientes de N. Seguidamente se grafica σ_o frente a los valores de N y se hace un ajuste de mínimos cuadrados para estimar el valor de K que mejor predice los valores obtenidos con la ecuación 2.35. Los gráficos obtenidos con su ajuste correspondiente, para distintos valores de SNR pueden verse en la Figura 2.3. Observando estos resultados es claro que el factor de proporcionalidad para este filtro en particular es aproximadamente igual a $\sqrt{2}$ para los valores ensayados, llegando a la misma conclusión de la ecuación 2.34.

Este análisis es original y está publicado en [53].

2.2.5.2. Filtros de respuesta impulsional infinita (IIR)

Otros filtros que han recibido mucha atención para la implementación de la técnica de *Lock-in* son los filtros de respuesta impulsional infinita (o IIR por su acrónimo en inglés de *infinite impulse response*). Estos son filtros recursivos que tienen la ventaja de ser sencillos de implementar, y utilizar pocos recursos, aunque no tienen la propiedad de bloquear completamente la componente del doble de frecuencia, como en los de media móvil. Un filtro IIR de orden N puede ser descripto por la ecuación en diferencias:

$$Y_n = b_0 X_n + \dots + b_N X_{n-N} + a_1 Y_n - 1 + \dots + a_N Y_{n-N}$$
(2.36)

Los coeficientes del filtro pueden calcularse para generar filtros pasa-bajo de distintos anchos de banda. La relación descripta por la ecuación 2.24 se sigue cumpliendo para estos filtros. Es decir que la desviación estándar del ruido en la salida disminuye proporcionalmente con la raíz cuadrada del ancho de banda del filtro. En [62] pueden encontrase simulaciones para calcular los factores de proporcionalidad K de la ecuación 2.24 en una forma similar a lo desarrollado para el filtro de media móvil, según el orden y el ancho de banda del filtro IIR seleccionado.

2.2.6. Efecto del ruido de cuantización

Se analizarán ahora los errores introducidos en la medición Lock-in por la cuantización de las señales. Para ello se parte de una señal continua s(t) periódica de una frecuencia f_s de la cual se desea conocer su magnitud y fase. Esta señal es muestreada a una frecuencia f_m (periodo de muestreo T_m), superior a la de Nyquist, y cuantizada a B bits, resultando una secuencia en tiempo discreto $s_{[n]}$:

$$s(t) = A_s \operatorname{sen}(\omega_s t + \Phi)$$

$$s(nT_m) = s_{[n]} = A_s \operatorname{sen}(\omega_s nT_m + \Phi) + \epsilon(nT_m)$$

donde el término $\epsilon(nT_m)$ contabiliza la perturbación en amplitud producida por la cuantización de la señal. Como esta se hace en 2^B niveles esta componente estará acotada por:

$$-\delta/2A_{FS} < \epsilon(nT_m) < \delta/2A_{FS}$$

donde $\delta = 2^{-B}$ es el mínimo paso de cuantización y A_{FS} es la amplitud de fondo de escala del ADC. Luego esta señal es sometida a procesamiento *Lock-in* con filtro de media móvil. Para ello primero se la multiplica por secuencias discretas $r_{s[n]}$ y $r_{c[n]}$ (referencias en fase y cuadratura). Para este análisis se supone que estas señales están cuantizadas en B_r bits, tal que $B_r \gg B$ por lo que se despreciará el error de cuantización presente en ellas.

$$r_{s[n]} = A_r \operatorname{sen}(\omega_s nT_m + \phi_r)$$
$$r_{c[n]} = A_r \cos(\omega_s nT_m + \phi_r)$$

Sin perder generalidad y para simplificar cuentas se puede suponer $\phi_r = 0$. Las señales multiplicadas son luego sometidas a sendos filtros de media móvil de largo igual a N ciclos de la secuencia $s_{[n]}$ (MAF de MN puntos, llamando M a la cantidad de puntos por periodo de la secuencia), obteniendo las magnitudes $X \in Y$. Finalmente se calcula la amplitud y fase de la señal s[n] utilizando las conocidas expresiones 2.8 y 2.9. En esta sección se estimará cómo afecta el error de cuantización de la señal (la perturbación llamada $\epsilon(kT_m)$) a los cálculos finales de amplitud y fase de la señal incógnita.

Para ello se parte de la expresión de $s_{[n]} \cdot r_{s[n]}$.

$$s(nT_m)r_s(nT_m) = (A_s \operatorname{sen}(\omega_s nT_m + \Phi) + \epsilon(nT_m))A_r \operatorname{sen}(\omega_s nT_m)$$

= $A_s A_r \operatorname{sen}(\omega_s nT_m + \Phi)\operatorname{sen}(\omega_s nT_m) + A_r \epsilon(nT_m)\operatorname{sen}(\omega_s nT_m)$
= $\frac{A_s A_r}{2}(\cos(\Phi) - \cos(2\omega_s nT_m + \Phi)) + A_r \epsilon(nT_m)\operatorname{sen}(\omega_s nT_m)$ (2.37)

En esta expresión se pueden apreciar dos términos: uno que depende de la señal y uno que depende del ruido de cuantización. Ahora bien, esta señal pasa por un filtro pasa-bajos que se espera que rechace la componente del doble de frecuencia (en el caso del filtro de media móvil este rechazo es absoluto), por lo que luego de aplicar el pasa-bajos se tendrá:

$$P_B\{s(nT_m)r_s(nT_m)\} = \frac{A_sA_r}{2}\cos(\Phi) + P_B\{A_r\epsilon(nT_m)\operatorname{sen}(\omega_s nT_m)\}$$
(2.38)

donde se ha simbolizado la operación de filtrado pasa-bajos con P_B {}. Esta expresión muestra por un lado la magnitud X que se esta buscando medir con la acción del *Lock-in* $(\frac{A_sA_r}{2}\cos(\Phi))$ y por otro un término producido por los errores de cuantización en la señal. Se analizará este término buscando determinar cuánto puede afectar a la medición final de X.

$$D_x = P_B\{A_r \epsilon(nT_m) \operatorname{sen}(\omega_s nT_m)\}$$
(2.39)

En este punto es necesario discriminar entre dos casos distintos: cuando la frecuencia de muestreo es un múltiplo entero de la frecuencia de la señal objetivo $(f_m = Kf_s)$ y cuando no lo es. En el primer caso el error de cuantización resultará periódico, con la misma frecuencia de la señal (f_s) [58]. A modo de ejemplo se muestra en la Figura 2.4 la Transformada Discreta de Fourier del error de cuantización para distintas señales muestreadas a $f_m = 1$ kHz. Para aquellos valores de f_s que son divisores enteros de la frecuencia de muestreo (10 Hz y 25 Hz) se ve la periodicidad del ruido de cuantización, con componentes claramente marcados en f_s y sus armónicos. Para los otros casos el espectro se distribuye entre frecuencias cercanas.

Antes de hacer cuentas es posible sacar algunas conclusiones del efecto del Lock-in sobre estas señales mirando los espectros correspondientes. Para los casos en los que $f_m = Kf_s$ la componente del error de cuantización que esta en la frecuencia f_s inevitablemente aparecerá en la medición de amplitud y fase del Lock-in, y no se podrá reducir su efecto afinando el filtro pasa-bajos utilizado (aumentando N para el caso del filtro MA). Por otro lado todas las armónicas superiores se rechazarán, en forma total para el filtro MA. Para los casos en que $f_m \neq Kf_s$ la componente exactamente en fase con la señal puede ser menor, por lo que probablemente se podrá reducir el error de cuantización afinando el filtro pasa-bajos. Pero por otro lado, habrá más ruido distribuido en las inmediaciones de esta f_s por lo que es de suponer que el ruido para valores bajos de N (filtros pasa-bajos menos selectivos) será superior.

2.2.6.1. Caso $f_m = K f_s$

Se analizará primero el caso en que $f_m = K f_s$. En estas condiciones, como el ruido de cuantización es periódico, se puede representar a $\epsilon(nT_m)$ con su serie de Fourier:

$$\epsilon(nT_m) = E_{0f} + E_{1f}\cos(\omega_s nT_m - \phi_{1f}) + E_{2f}\cos(2\omega_s nT_m - \phi_{2f}) + E_{3f}\cos(3\omega_s nT_m - \phi_{3f}) + \dots$$
(2.40)

Con lo que la expresión 2.39 se convierte en:

$$D_x = P_B \{ A_r (E_{0f} + E_{1f} \cos(\omega_s nT_m - \phi_{1f}) + E_{2f} \cos(2\omega_s nT_m - \phi_{2f}) + E_{3f} \cos(3\omega_s nT_m - \phi_{3f}) + \dots) \operatorname{sen}(\omega_s nT_m) \}$$



Figura 2.4: Espectro de ruido de cuantización para señales muestreadas a $f_m = 1kHz$.

Distribuyendo y aplicando propiedades trigonométricas esta expresión resulta:

$$D_{x} = P_{B} \{ A_{r} E_{0f} \operatorname{sen}(\omega_{s} n T_{m}) + A_{r} E_{1f} \cos(\omega_{s} n T_{m} - \phi_{1f}) \operatorname{sen}(\omega_{s} n T_{m}) + A_{r} E_{2f} \cos(2\omega_{s} n T_{m} - \phi_{2f}) \operatorname{sen}(\omega_{s} n T_{m}) + A_{r} E_{3f} \cos(3\omega_{s} n T_{m} - \phi_{3f}) \operatorname{sen}(\omega_{s} n T_{m}) + \ldots \}$$

$$D_{x} = P_{B} \{ A_{r} E_{0f} \operatorname{sen}(\omega_{s} n T_{m}) + \ldots \}$$

$$(2.41)$$

$$A_{r} \frac{E_{1f}}{2} (\operatorname{sen}(\phi_{1f}) + \operatorname{sen}(2\omega_{s}nT_{m} - \phi_{1f})) + A_{r} \frac{E_{2f}}{2} (\operatorname{sen}(nT_{m} - \phi_{2f}) \operatorname{sen}(3\omega_{s}nT_{m} - \phi_{2f})) + A_{r} \frac{E_{3f}}{2} (\operatorname{sen}(2\omega_{s}nT_{m} - \phi_{3f}) \operatorname{sen}(4\omega_{s}nT_{m} - \phi_{3f})) + \dots \}$$

$$(2.42)$$

En esta expresión puede verse un término en banda base $(\frac{A_r E_{1f}}{2} \operatorname{sen}(\phi_{1f}))$ y términos en frecuencias que son múltiplos de la frecuencia de la señal (f_s) . Todos estos términos se espera que sean rechazados por el filtro pasa-bajos, y son de hecho rechazados en su totalidad en el caso del filtro MA. Con lo cual esta expresión resulta:

$$D_x = \frac{A_r E_{1f}}{2} \operatorname{sen}(\phi_{1f}) \le \frac{A_r E_{1f}}{2}$$
(2.43)

Como se supuso en el análisis cualitativo, la componente del error de cuantización que está en la misma frecuencia de la señal es la que aparece en la salida del *Lock-in*. Un análisis análogo puede hacerse con la multiplicación de la señal de interés con la referencia en cuadratura $r_{c[n]}$, resultando:

$$D_y = P_B\{s(nT_m)r_c(nT_m)\} = \frac{A_r E_{1f}}{2}\cos(\phi_{1f}) \le \frac{A_r E_{1f}}{2}$$
(2.44)

Para ver como afecta estas Dx y Dy a la medición de amplitud del *Lock-in* se puede pensar a las magnitudes Dx y Dy como errores cometidos al medir X e Y y ver como se propagan estos errores a la medición de A_s prevista con la expresión 2.8:

$$\Delta A_s \le \left| \frac{2}{A_r} \frac{2X}{2\sqrt{X^2 + Y^2}} \right| \Delta X + \left| \frac{2}{A_r} \frac{2Y}{2\sqrt{X^2 + Y^2}} \right| \Delta Y = \frac{2D(|X| + |Y|)}{A_r\sqrt{X^2 + Y^2}}$$

donde $D_x = D_y = D$. Esta última desigualdad se puede acotar utilizando la desigualdad de Cauchy-Shwartz, que establece:

$$(|X| + |Y|)^2 \le (|X|^2 + |Y|^2)(1^2 + 1^2)$$
$$\frac{(|X| + |Y|)}{\sqrt{X^2 + Y^2}} \le \sqrt{2}$$

Con lo que la cota superior para el error de A_s queda dada por:

$$\Delta A_s \le \sqrt{2} \frac{2D}{A_r} \le \sqrt{2} E_{1f} \tag{2.45}$$

Una posible cota, muy pesimista, para E_{1f} puede ser el mínimo paso de cuantización $\delta = A_{FS}2^{-B}$. Por lo cual seguro se va a cumplir para este caso donde $f_m = Kf_s$:

$$\Delta A_s \le A_{FS} \sqrt{2} \cdot 2^{-B} \tag{2.46}$$

En la Figura 2.5 se muestra el error de cuantización en la estimación de amplitud de señales de distintas frecuencias con amplificador *Lock-in* con filtro de media móvil. La señal de entrada es de $A_s = 1$ V de amplitud y es muestreada a $f_m = 10$ kHz. La cuantización de la señal se hace en B = 14 bits, con una tensión de fondo de escala $A_{FS} = 1$ V. En este gráfico solo se muestran señales que cumplen $f_m = K f_s$, con K variando desde 10 hasta 100. En todos los casos el error respeta con creces la cota calculada en la ecuación 2.46 (esta es una cota muy pesimista que supone que toda la potencia de la señal de error esta en f_s). Es interesante notar que el error de cuantización en estas circunstancias es independiente de N, de hecho puede verse que los valores obtenidos para un N de 10 y un N de 100 son exactamente iguales. Esto es lógico ya que este error es cometido por la componente que está en la frecuencia a la que se hace el *Lock-in* (E_{1f}) , y el error de cuantización solo tiene componentes en esta frecuencia y sus armónicos, como se veía en la Figura 2.4, que son totalmente rechazados por el filtro MA.

En esta Figura también puede verse el error de cuantización para el caso de un filtro IIR de orden 4, con una frecuencia de corte $\frac{f_s}{20}$. Tampoco se aprecian diferencias entre el error de cuantización utilizando este filtro y un filtro MA, ya que este error depende directamente de la componente del ruido que esta en la frecuencia f_s . Con filtros IIR de menor frecuencia de corte seguirá pasando lo mismo, ya que esta componente del error está justamente en la frecuencia que se está intentando medir con el Lock-in.

En [58] se acota E_{1f} para el caso $f_m = 16f_s$ utilizando la Transformada Discreta de Fourier del ruido de cuantización, obteniendo $E_{1f} \leq 0,89\delta$. Para otros valores de K se pueden encontrar cotas que seguramente serán más ajustadas que la calculada $(A_{FS}\sqrt{2} \cdot 2^{-B})$ pero eso excede las pretensiones de este texto.



Figura 2.5: Error de cuantización en señales que cumplen $f_m = K f_s$, con $f_m = 10$ kHz

2.2.6.2. Caso $f_m \neq K f_s$

Para este caso el error de cuantización ya no tiene la forma periódica que se puede apreciar cuando $f_m = K f_s$, sino que tiene componentes tanto en la frecuencia f_s como en las frecuencias circundantes. Esto dificulta encontrar una expresión matemática para acotar los efectos del error de cuantización en las mediciones del *Lock-in*. Sin embargo, se pueden sacar algunas conclusiones interesantes analizando simulaciones en distintos escenarios.

En la Figura 2.6 se ve el efecto del ruido de cuantización en las mediciones de amplitud del Lock-in con filtro MA con un N = 10, para $f_m = 10$ kHz, $A_s = A_{FS} = 1$ V y distintos valores de f_s cercanos a 100 Hz. Se ve claramente que en muchas de las frecuencias que cumplen $f_m \neq K f_s$ el error en la medición provocado por la cuantización excede la cota calculada en la sección 2.2.6.1. Esto es porque a estas frecuencias el error de cuantización tiene componentes en frecuencias distintas a f_s y sus múltiplos, que entran al Lock-in perturbando la medida. Como en N = 10 el filtro no resulta muy selectivo este efecto se hace muy notorio.

Promediando una mayor cantidad de muestras, es decir, afinando el filtro pasa-bajos, se puede reducir el efecto del ruido de cuantización, como se ve en la Figura 2.7. Es interesante notar que el ruido en las frecuencias que cumplen $f_m = K f_s$ (100 Hz en este caso) se ve inalterado por un incremento en la promediación. De todas formas hay un límite en la mejora posible dada por la amplitud de la componente del error de cuantización a la frecuencia f_s , que si bien no se puede determinar exactamente para todos los valores de f_s , es esperable que sea menor que en los casos donde $f_m = k f_s$, debido a que la misma potencia de señal se distribuye entre f_s y frecuencias cercanas. De esta Figura se ha omitido la cota calculada en 2.2.6.1 por claridad, ya que queda muy por encima de todos los errores calculados para estos valores de N.

En caso de utilizar otros tipos de filtros también se puede apreciar la reducción de ruido al reducir el ancho de banda del filtro PB, en frecuencias donde $f_m \neq k f_s$. En la Figura 2.8, por ejemplo, se muestra el ruido de cuantización para f_s variando desde 90 hasta 110 Hz. En ella se ve que la media de ruido baja para el filtro de menor ancho de banda, por más que en la frecuencia que sí cumple $f_m = k f_s$ ($f_s = 100$ Hz) el error sea igual.



Figura 2.6: Error de cuantización en señales, incluyendo aquellas que no cumplen $f_m = K f_s$, para N = 10 y $f_m = 10$ kHz



Figura 2.7: Error de cuantización en señales, incluyendo aquellas que no cumplen $f_m = K f_s$, para $f_m = 10$ kHz y distintos valores de N.

2.2.6.3. Resumen

En conclusión, el error de cuantización afecta los resultados del *Lock-in* de distintas formas, dependiendo la relación entre la frecuencia de operación (f_s) y la frecuencia de muestreo del sistema (f_m) .

- En casos donde f_s es divisor exacto de f_m el error de cuantización es siempre igual, independientemente del filtro pasa-bajos utilizado. Esto es porque en estas condiciones la señal error de cuantización adquiere una forma periódica, sincronizado con la señal, por lo que tiene componentes solo en f_s y sus armónicos, que son rechazados por el filtro pasa-bajos.
- En estas condiciones una cota pesimista del error de amplitud de la señal debido a la cuantización esta dada por $\Delta A_s < A_{FS}\sqrt{2} \cdot 2^{-B}$. Cotas menores pueden calcularse para casos particulares de la relación entre $f_m \ge f_s$.



Figura 2.8: Error de cuantización en señales, incluyendo aquellas que no cumplen $f_m = K f_s$, para filtros IIR, con $f_m = 10$ kHz.

- Para los casos en donde f_s no es divisor exacto de f_m el error de cuantización sí depende del filtro pasa-bajos utilizado, ya que en estas condiciones la señal error de cuantización tiene componentes tanto en f_s como en las frecuencias cercanas. La componente en f_s no se puede reducir, pero si las atribuciones de las otras componentes.
- Para filtro de media móvil, por ejemplo, para valores bajos de N el error de cuantización resulta mucho mayor en los casos donde $f_m \neq K f_s$, y para valores muy altos de N el error resulta ligeramente menor.
- En conclusión hay más espacio de mejora para los casos $f_m \neq K f_s$, pero mayor predictibilidad y mejor relación entre el ruido final y la exigencia impuesta al filtro pasa-bajos para los casos con $f_m = K f_s$.

2.2.7. Caso $f_s = 100 \text{ kHz}$

Un caso de especial interés en esta tesis es el de una frecuencia objetivo de 100 kHz. Esta se utilizará para la detección de nanopartículas ferromagnéticas mediante tomografía magnética. En esta sección, se analizan los niveles de ruido esperables al aplicar amplificación *Lock-in* con MAF a una señal de estas características.

Un análisis inicial, sin considerar el error de cuantización, indica que el nivel de ruido a la salida del *Lock-in* disminuye con la raíz de la cantidad de muestras consideradas en el promedio, como describe la ecuación 2.34. Sin embargo, el límite en la reducción del ruido está impuesto por el error de cuantización de la señal, tal como se estudió en la sección 2.2.6. Suponiendo que la señal es muestreada a un múltiplo de la frecuencia de señal, por ejemplo a 64 MHz, y tomando como válida la cota calculada en la sección 2.2.6.1, el menor nivel de ruido esperable considerando un ADC de 14 bits y 1V de tensión de fondo de escala es:

$$\Delta A_s \le A_{FS} \sqrt{2} \cdot 2^{-B} \approx 86 \mu V \tag{2.47}$$


Figura 2.9: Error de cuantización de señales de 100 kHz y distintas amplitudes, muestreada a 64 MHz y cuantizadas en 14 bits para distintos valores de la fase inicial de la señal.

No obstante, esta cota es conservadora. Para estimar el error con mayor precisión, se simuló el proceso. La simulación parte de una señal sinusoidal de 100 kHz y amplitud ±1 V, muestreada a una tasa muy alta para simular una señal continua, con una fase inicial variando entre 0 y 2π . Posteriormente, se muestreó esta señal a 64 MHz y se cuantizó a 14 bits, con un rango de entrada $A_{FS} = 1$ V. Usando *Lock-in* con MAF, se estimó la amplitud de la señal y se calculó el error cometido, considerando que la amplitud de la señal de entrada es conocida. Los resultados se muestran en la Figura 2.9.

En este caso, dado que $f_m = K f_s$, el error de cuantización es independiente del número de ciclos promediados en el *Lock-in*, y no hay otras fuentes de error en este modelo. Por ello, se consideró N = 1. Dependiendo de la fase inicial de la señal de entrada, la cuantización varía ligeramente, lo que afecta el error en la estimación del *Lock-in*. Los resultados indican que el error permanece por debajo de 6 μ V, con una media de 2,35 μ V. Este valor corresponde a una señal de entrada que cubre todo el rango dinámico del ADC ($A_s = A_{FS}$), lo que define un límite en la mejora del ruido impuesto por un ADC de 14 bits para este caso.

Si la señal no cubre todo el rango de entrada $(A_s < A_{FS})$ su potencia es menor. Esto hace que la componente del error de cuantización en la frecuencia objetivo sea menor, lo que reduce el error en la estimación del *Lock-in* asociado a la cuantización. Este efecto también se observa en la Figura 2.9, donde se ve que una señal de entrada con $A_s = 0,1$ V resulta en un error promedio de $1,5 \,\mu$ V. Aunque el error puede variar significativamente para otras amplitudes y combinaciones de frecuencias, no es el objetivo de este texto cubrir todos los casos posibles. Es importante notar sin embargo que el límite en la mejora está determinado por el error de cuantización, que depende directamente de la resolución del ADC.

2.2.8. Antecedentes

Los primeros registros de los principios que rigen el funcionamiento de los amplificadores *Lock-in* pueden encontrarse en publicaciones de la década de 1930. Tal vez la primera implementación que se pueda encontrar sea la de Cosens [63], quien desarrolló un circuito analógico basado en válvulas para un detector coherente que mejoraba la detección del punto de balance en puentes de corriente alterna. Más tarde Michels [64], [65] tomó las ideas de Cosens y las utilizó para desarrollar detectores con alta selectividad frecuencial. El fue, quizás, el primero en emplear el término *Lock-in*

como se lo conoce al día de hoy.

A mediados del siglo 20 esta técnica de medición ganó popularidad y comenzó a comercializarse. Al día de hoy los principales fabricantes de este tipo de instrumental son *Stanford Research* y *Zurich Instruments*. Estos proveen dispositivos con prestaciones varias, pero tienen en común que son costosos y robustos, bien adaptados para uso en entornos controlados de laboratorio. En el cuadro 2.1 se muestra una visión general de los dispositivos disponibles en la actualidad.

Serie	Fabricante	Costo [USD]	Consumo	Ancho de banda	Ruido $\left[\frac{nV}{\sqrt{\text{Hz}}}\right]$
SR830	Stanford	5995	40 W	1 mHz - 102.4 kHz	6
SR860	Stanford	6495	$60 \mathrm{W}$	$1~\mathrm{mHz}$ - $500~\mathrm{kkHz}$	2,5
SR865A	Stanford	9150	$60 \mathrm{W}$	1 mHz - 4 MkHz	2,5
SR844	Stanford	8450	$70 \mathrm{W}$	25 kHz - 200 MkHz	5
					2,5 @ 1 khz
MFLI	Zurich	6290	< 40W	DC a 500 kHz *	$7 @ 10 \mathrm{~Hz}$
					$40 @ 1 \mathrm{~Hz}$
HF2LI	Zurich	†	Ť	DC - 50 MHz	5
UHFLI	Zurich	Ť	Ť	DC - 600 MHz	4
CHELI	Zurich	+	+	DC 18 CH2	3.5 @ 100 kHz
GIII LI	Zurich		1	DC - 1,8 GHZ	6 @ 800 MHz
SHEI I	Zurich	+	+		3.5 @ 100 kHz
5111 11	Zurich		1	DC - 0,5 GHZ	2,5 @ 800 MHz
LI 5660	NF	11359	Ť	0,5 Hz - 11 MHz	4,5
LI 5655	NF	8099	Ť	0,5 Hz - 3 MHz	4,5
LI 5650	\mathbf{NF}	6859	Ť	$1 \mathrm{~mHz}$ a $250 \mathrm{~kHz}$	4,5
LI 5640	NF	†	$40 \mathrm{W}$	$1 \mathrm{~mHz}$ a $100 \mathrm{~kHz}$	6
LI 5610B	NF	†	†	0,5 Hz a 200 kHz	5

Cuadro 2.1: Análisis de amplificadores Lock-in disponibles comercialmente

*Posibilidad de aumentarlo a 5MHz bajo pedido.

† Valores no provistos por el fabricante.

La amplia gama de aplicaciones en donde los amplificadores *Lock-in* pueden utilizarse justifica el interés en desarrollar dispositivos portátiles y de bajo costo para aplicaciones embebidas. Un buen resumen sobre implementaciones de este tipo puede encontrarse en [66].

Para este trabajo de investigación cobran especial relevancia las implementaciones de *Lock-in* en dispositivos FPGA y SoC-FPGA. Se detallan a continuación algunas publicaciones de este tipo de sistemas. Cabe destacar que esta enumeración no pretende cubrir todos las publicaciones disponibles, sino dar una visión general del panorama actual en este sentido. Esta información se encuentra resumida en el cuadro 2.2, donde se muestra un resumen de las publicaciones enunciadas, destacando los principales aspectos de cada una.

 En [67] se desarrolla un Lock-in que logra recuperar señales con una SNR de 60 dB utilizando una combinación de un DSP (Procesador de señales digitales) y una FPGA, para aplicaciones en sistemas de Tomografía de impedancia eléctrica (EIT). En este trabajo la FPGA se usa para generar las señales de excitación y sincronizar el proceso y el DSP se utiliza para la demodulación. Las señales generadas pueden variar en el rango de 10 Hz a 1 MHz, pero no se especifican las capacidades de demodulación del DSP. El sistema utiliza un filtro de media móvil, y se prueba con señales de 10 kHz.

- En [58] se estudian las posibles fuentes de error en la determinación de la fase de un Lock-in implementado en una FPGA Xilinx Spartan3A-DSP FPGA, y luego se prueban experimentalmente los resultados, implementando un dispositivo que demodula señales hasta 1 kHz. Para el filtro pasa-bajos, se ensayan dos alternativas: un filtro FIR (Filtro de respuesta impulsional finita) equirriple de fase lineal de orden 59 y un filtro FIR modificado para balancear eficiencia y limitaciones de hardware (MFIR).
- En [68] se desarrolla un Lock-in en una FPGA Spartan 3 de AMD-Xilinx para aplicaciones en mapeo foto-acústico. El dispositivo desarrollado funciona en el rango de frecuencias de 10 Hz a 1 kHz y utiliza un filtro de promedio móvil dinámico como pasa-bajos. Se comparan resultados obtenidos de la demodulación de señales con el dispositivo desarrollado con los mismos utilizando un Lock-in comercial SR830, con resultados satisfactorios. Se destaca la reducción de espacio, peso y costo logradas en la implementación.
- En [69] se implementa un Lock-in con aplicaciones en espectroscopia funcional del infrarrojo cercano, una técnica de monitorización óptica del cerebro para obtener neuroimágenes funcionales. El sistema logrado permite demodular señales hasta 2 kHz, y se testean filtros FIR e IIR (filtro de respuesta impulsional infinita) como pasa-bajos, destacando el IIR como mejor alternativa en términos de recursos, velocidad y rechazo del ruido. Se destaca que la implementación permite una gran modularidad, posibilitando dispositivos multi-canal y se realizan pruebas de inmunidad al ruido para probar el sistema.
- En [70] se presenta el diseño de código abierto de un *Lock-in* basado en una FPGA de AMD-Xilinx Zynq 7010 SoC, montado en una placa de desarrollo Red Pitaya. El dispositivo desarrollado utiliza un filtro IIR de un único polo, permite demodular señales de hasta 50 MHz y tiene un ruido referido a la entrada de 90 $nV/\sqrt{\text{Hz}}$. Se comparan los resultados obtenidos con un *Lock-in* comercial HF2LI de *Zurich Instruments* con resultados satisfactorios.
- En [71] se aplican los conceptos del *Lock-in* para desarrollar un sensor que permite medir la permeabilidad de materiales. El dispositivo obtenido permite recuperar señales con SNRs entre 56,3 dB y 78,05 dB, dependiendo de la frecuencia utilizada. El sistema puede operar hasta 125 kHz, con una operación óptima en 31,25 kHz, aunque se afirma que este rango de operación podría extenderse hasta algunos MHz, y utiliza una FPGA, aunque no se dan detalles muy precisos sobre su implementación.
- En [72] se presenta un sistema desarrollado en la Universidad de Buenos Aires que implementa distintos instrumentos en una plataforma Red Pitaya, que contiene una FPGA Zynq 7010 de AMD-Xilinx. Entre estos dispositivos se implementan dos amplificadores *Lock-in* para cubrir diferentes tipos de aplicaciones: uno armónico estándar, con un rango de frecuencias que va desde 3 Hz hasta 49,6 kHz, y un *Lock-in* de onda cuadrada con un rango de frecuencias más amplio, desde 30 mHz hasta 31 MHz. Es destacable de este trabajo la distribución de las tareas entre el procesador y la FPGA, donde esta última se encarga del manejo de las señales y el procesador puede usar programas predefinidos o trabajar directamente con las señales en Python, Matlab, etc. El sistema utiliza un filtro pasa-bajos de orden y frecuencia de corte configurables en tiempo real, y se muestra su aplicación con éxito a distintos experimentos.
- En [73] se utiliza un arreglo de 100 amplificadores *Lock-in* de bajas frecuencias implementados en una FPGA Arria V de Intel-Altera para construir imágenes fluorescentes de células, que

pueden utilizarse para reconocer un patrón específico de odorantes. Este artículo no provee métricas específicas de desempeño, pero si pone en perspectiva como pueden emplearse las FPGAs para realizar *Lock-in* en muchos canales en forma simultanea.

- En [74] se desarrolla un *Lock-in* portable para implementar un sensor foto-acústico, destinado a dispositivos vestibles biomédicos. Para ello se utiliza un multiplicador analógico implementado con una celda de Gilbert y una FPGA de Intel-Altera, montada en una placa de desarrollo de Terasic, para realizar el filtrado pasa-bajos de la señal. El sistema desarrollado funciona en un rango de frecuencias de 0,1 MHz a 10 MHz y, si bien no se proveen métricas detalladas de desempeño sí se afirma que logra mejorar la precisión de detección frente a alternativas similares.
- En [75] se desarrolla un Lock-in con seguimiento de frecuencia automático. Para ello primero se somete la señal a una Transformada rápida de Fourier (FFT), tomando el punto de máxima amplitud como primera aproximación de la frecuencia de operación, para luego mejorarla mediante un PLL (lazo de enganche de fase). Para caracterizar el sistema se estudia la linealidad del sistema, se calcula el Q del filtro equivalente ($Q = \frac{f_0}{\Delta f}$) obteniendo Q = 82 y se miden algunas señales para caracterizar el error relativo, obteniendo errores menores al 2%. El filtro pasa-bajos utilizado fue un FIR de orden 16 y no se especifica el ancho de banda del equipo, aunque se puede inferir que esta optimizado para señales de 1 kHz.
- En [76] se implementa un *Lock-in* en FPGA para medir interferencia electromagnética en una estación generadora de alta tensión. El sistema puede operar en frecuencias entre 100 Hz y 10 kHz, utiliza una referencia cuadrada y un acumulador como filtro pasa-bajos. Los resultados de las pruebas realizadas muestran que cuando la SNR de la señal a ser probada es de -20 dB, el error de medición de su amplitud es inferior al 3%.
- En [77] se describe un método para medir la frecuencia de señales con un error menor a $10 \,\mu\text{Hz}$, basado en amplificadores *Lock-in* e implementado en una FPGA. Para ello se mide la amplitud de la señal a distintas frecuencias cercanas a la frecuencia de interés y luego se interpolan los resultados mediante un ajuste sinusoidal. El método propuesto tiene la ventaja de proveer adicionalmente la amplitud y fase de la señal a su máxima frecuencia, a diferencia de un frecuenciómetro comercial. Su eficiencia es $\log_2(N)$ veces mayor que la de una FFT.
- En [78] se describe el diseño de un sistema de detección de señales magnéticas débiles para la detección de tensiones residuales, utilizando un método basado en corrientes de Foucault. Para ello se implementa un *Lock-in* en un SoC-FPGA Zynq-7020, de AMD-Xilinx. En este sistema se utiliza la FPGA para generar y medir las señales y el procesador ARM del SoC para realizar la extracción de características de la señal, lo cual no es una práctica muy común pero pone en evidencia la flexibilidad de los sistemas SoC-FPGA. No se provee una descripción exacta de la frecuencia de trabajo, pero la etapa analógica puede operar hasta 5 MHz y el sistema se prueba con señales hasta 100 kHz.

De este breve análisis bibliográfico se pueden sacar algunas conclusiones interesantes. El desarrollo de amplificadores *Lock-in* en dispositivos SoC-FPGA es un tema de interés actual, que se emplea en una amplia gama de aplicaciones y está lejos de alcanzar una solución definitiva. Los sistemas SoC-FPGA proporcionan mucha flexibilidad a la implementación: desde la distribución de las tareas y la elección de los filtros y multiplicadores a utilizar hasta el método de interacción con el usuario son aspectos que cada diseñador resuelve de distinta manera. La elección de filtro

Ref.	Ancho de banda	Filtro pasa-bajos	Métrica de desempeño	Aplicación
[67]	10 Hz - 10 kHz *	Media móvil	SNR 60 dB	EIT
[58]	0 - 1 kHz	FIR y MFIR	-	Teórico
[68]	10 Hz - 1 kHz	Media móvil	Comparación con LI comercial	Mapeo foto acústico
[69]	0 - 2 kHz	FIR e IIR	Test de inmunidad de ruido	Espectroscopia funcional del infrarrojo cercano
[70]	0 - 50 MHz	IIR	Comparación con LI comercial Ruido: $90 \frac{nV}{\sqrt{Hz}}$	Ť
[71]	0 - 125 kHz	Ť	SNR entre 56,3 dB y 78,05 dB	Medición de permeabilidad
[72]	1 Hz - 4936 kHz 30 mHz - 31 MHz	Filtro reconfigurable	Aplicaciones experimentales	Instrumentación
[73]	Bajas frecuencias	Ť	Aplicaciones experimentales	Imágenes fluorescentes de células (Implementación multi-canal)
[74]	0,1 MHz - 10 MHz	Ť	Comparación con alternativas	Sensor foto-acústico
[75]	$\approx 1 k H z$	FIR orden 16	Buena linealidad $Q_{\text{filtro eq}} = 82$ Error $< 2\%$	Seguimiento de frecuencia automático
[76]	100 Hz - 10 kHz	Acumulador	m Error < 3%	Interferencia electromagnética
[77]	†	†	Error de frec $< 10\mu Hz$	Medición frecuencial
[78]	Hasta 100 kHz	Ť	†	Detección de señales magnéticas

Cuadro 2.2: Resumen de publicaciones de amplificadores Lock-in en FPGA

* Posiblemente pueda llegar hasta 10 MHz. † No especificado. pasa-bajos y de la etapa analógica de adquisición de señal son aspectos fundamentales que hacen al desempeño de los equipos diseñados, aunque comparar los distintos sistemas directamente resulta complicado por la variedad de métricas que se usan para medir su desempeño.

Aspectos muy destacables de la implementación de amplificadores *Lock-in* en SoC-FPGA son su bajo costo frente a alternativas comerciales, incluso alcanzando prestaciones similares, la adaptabilidad de la implementación a una aplicación específica, la reducción de tamaño y peso y la capacidad de implementar dispositivos multi-canal. En cuanto a los anchos de banda de las publicaciones estudiadas son más comunes las aplicaciones a frecuencias bajas y medias; solo en [70], [72] y [74] se llega a aplicaciones en el orden de los MHz, utilizando la plataforma Red Pitaya en [70] y [72] y un ADC de alta velocidad desarrollado por Terasic en [74]. Más adelante (ver secciones 3.2 y 3.5.1) se discutirán en detalle estas plataformas de desarrollo.

Otro aspecto relevante para esta tesis es la eficiencia computacional de los algoritmos implementados. Se han realizado sucesivos esfuerzos para optimizar los algoritmos de Lock-in, buscando implementaciones eficientes en amplificadores multicanal o en dispositivos con prestaciones moderadas. Por ejemplo, en [79], los algoritmos involucrados en un Lock-in con filtro de media móvil se implementan como una multiplicación matricial. Posteriormente, en [80], se propone una simplificación interesante al considerar una señal sinusoidal de 4 puntos por ciclo como referencia. Esta simplificación, representada como la progresión numérica [0; 1; 0; -1], elimina la necesidad de multiplicaciones y mejora la resolución mediante sobremuestreo. En [81], se reduce la cantidad de multiplicaciones en un factor de 4, siguiendo un enfoque similar al de [80]. Mientras tanto, [82] investiga el uso de ondas cuadradas para la detección Lock-in, una técnica más eficiente computacionalmente. Este trabajo también estudia la selección de filtros pasa-bajos y la frecuencia de operación para mitigar los efectos de los armónicos generados por estas referencias. Además, en [83], se optimiza un Lock-in de dos fases con filtro de media móvil para mejorar la precisión de las mediciones de fase en sistemas de transferencia de energía inalámbrica, mientras que en [84], se aplica un enfoque similar al de [80] para simplificar los algoritmos de Lock-in y adaptarlos a microcontroladores encargados de la extracción de parámetros de calidad de energía en redes de distribución.

Estos son solo algunos de los avances que se han hecho en el camino de reducir la demanda computacional de los algoritmos *Lock-in*, que se enumeran para poner en perspectiva la intención de la comunidad científica de avanzar en este sentido.

2.3. Promediación coherente

2.3.1. Fundamentos

La promediación coherente (CA, por su acrónimo del inglés *coherent averaging*) es una técnica comúnmente utilizada en el procesamiento de señales periódicas ruidosas. En este enfoque, se promedian coherentemente las muestras de una señal ruidosa con una periodicidad conocida, en una operación que puede describirse matemáticamente según:

$$Y_{ca[n]} = \frac{1}{N_{ca}} \sum_{i=0}^{N_{ca}-1} x_{[n+iM]} \qquad 0 \le n < M$$
(2.48)

en donde M representa la cantidad de muestras por ciclo de señal y N_{ca} representa la cantidad de ciclos de señal promediados. De esta manera, las muestras que ocupan la misma posición relativa con respecto al número total de muestras en un período se promedian juntas, lo que tiene el efecto

de reducir el ruido en la señal [85], como ejemplifica la Figura 2.10. En esta se parte de una señal perturbada con ruido blanco, con una SNR de -20 dB , tal que en un principio es indistinguible el ruido y la señal subyacente. A medida que se promedian coherentemente más ciclos de la señal se puede empezar a apreciar la señal original.

Para aplicar correctamente la técnica de promediación coherente, es necesario conocer con exactitud el punto de inicio de cada ciclo de la señal. Tener algún tipo de variación en esta medida podría ser problemático y conducir a grandes errores [86].

2.3.2. Antecedentes

Las técnicas de promediación coherente han sido estudiadas en detalle por Rompleman. En un primer artículo [85], el autor analiza cómo esta técnica reduce el ruido en señales periódicas y determina cómo puede modelarse su efecto mediante un filtro equivalente. En resumen, llega a tres conclusiones interesantes: primero, la reducción de ruido es más efectiva al aumentar el número de muestras, con una mejora proporcional a $\sqrt{N_{ca}}$; segundo, las componentes de DC o muy lentas se ven poco afectadas; y tercero, la función de transferencia del filtro equivalente es periódica. Esto significa que algunas señales no deseadas se reducen más que otras, según su frecuencia. En un segundo artículo [86], Rompleman analiza como las variaciones en puntos de inicio de cada ciclo (comúnmente conocidos como Trigger Jitter) pueden afectar las medidas. Estas fluctuaciones suavizan la respuesta, introduciendo un efecto de filtrado, con una transferencia que es deducida en el texto. Luego se estudian efectos particulares de una aplicación específica: las respuestas cerebrales evocadas. En particular se estudia como afecta el solapamiento de ciclos de la señal periódica a las mediciones, y como estos efectos pueden mitigarse mediante estimulación aperiódica. Un tercer artículo [87] se centra en los efectos de la digitalización de las señales, en particular de errores de cuantización y aliasing.

Estos artículos dan una visión general de la técnica de promediación coherente. Desde entonces se han hecho grandes avances y es hoy una técnica usualmente empleada. De las publicaciones



Figura 2.10: Ejemplos de aplicación de promediación coherente en señal de SNR = -20 dB, para distintos valores de N_{ca} .

actuales son de destacar aquellas que buscan mejorar la eficiencia de los algoritmos, como en [88], una implementación en FPGA que logra una reducción del 96 % en el almacenamiento de memoria, del 98 % en el número de sumadores, y del 32 % en el tiempo de procesamiento frente a alternativas similares, o en [89], en donde se busca solucionar el problema de los errores causados por el desconocimiento de la periodicidad exacta de la señal, con aplicaciones en comunicaciones satelitales. Otros artículos interesantes son [90] y [91], en donde la técnica de promediación coherente se utiliza para mejorar la SNR de las señales, con aplicaciones en mediciones fisiológicas. Estos son solo algunas de las publicaciones disponibles sobre el tema, que ponen en relieve la importancia actual de esta técnica de procesamiento.

2.4. Combinación de promediación coherente y Lock-in

La promediación coherente y la amplificación *Lock-in* con filtro de media móvil (MAF) pueden utilizarse independientemente, o pueden combinarse para conformar un sistema único, como puede verse en la Figura 2.11.

En este escenario la señal pasa por una etapa de promedio coherente de N_{ca} ciclos, luego por una etapa de multiplicación y finalmente por una etapa de media móvil de N_{ma} ciclos. Dado que la señal de modulación se genera internamente, el punto de inicio de cada periodo es conocido y el promedio coherente se puede realizar con éxito. De este esquema surge naturalmente la pregunta de cómo dividir las muestras entrantes entre el promedio coherente y el procesamiento de MAF, es decir que valores tomar para N_{ca} y N_{ma} .

En la sección 2.4.1 se demostrará que realizar realizar la detección *Lock-in* con filtro de media móvil durante N ciclos de señal es matemáticamente equivalente a realizar un promedio coherente durante N_{ca} ciclos, seguido de detección *Lock-in* con MAF durante N_{ma} ciclos, siempre y cuando $N = N_{ma}N_{ca}$. Seguidamente se analizará el número de operaciones entre diferentes métodos, llegando a la conclusión de que la forma más eficiente de realizar las operaciones es con $N_{ca} = N$ y $N_{ma} = 1$. Esta deducción es original y ha sido publicada en [54].



S[n] (señal de referencia)

Figura 2.11: Combinación de promediación coherente y Lock-in

2.4.1. Formulación matemática

2.4.1.1. Conceptos generales

Se tiene una señal periódica $s_{[n]}$ de periodo M, tal que $s_{[n]} = s_{[n+kM]}$ para cualquier entero k. Esta señal puede utilizarse para hacer *Lock-in* de una fase durante N periodos en una señal arbitraria de entrada $x_{[n]}$. Esta operación se puede describir matemáticamente con la ecuación 2.49:

$$Y_{1[n]} = \frac{1}{NM} \sum_{k=n}^{n+NM-1} x_{[k]} s_{[k]}$$
(2.49)

Ahora se supone que una señal arbitraria $x_{[n]}$ se promedia coherentemente durante N_{ca} periodos de una señal periódica $s_{[n]}$. Posteriormente, la señal promediada ingresa a un sistema clásico de detección *Lock-in* con MAF de N_{ma} periodos.

Para esta última operación, se necesitan N_{ma} periodos de la señal, donde cada uno es el promedio coherente de N_{ca} periodos. Las muestras del primer periodo de la señal promediada coherentemente $Y_{ca[n]}$ se pueden expresar según:

$$Y_{ca[n]} = \frac{1}{N_{ca}} \sum_{i=0}^{N_{ca}-1} x_{[n+iM]} \qquad 0 \le n < M$$
(2.50)

El segundo periodo de Y_{ca} se obtiene promediando los segundos N_{ca} periodos de la señal entrante. Entonces, si n está entre M y 2M, el promedio comienza desde $N_{ca}M$ y se promedian N_{ca} periodos. Para la formulación matemática, es útil considerar un índice relativo, l = n - M, que es la posición de la muestra en relación con el periodo de M puntos.

$$Y_{ca[n]} = \frac{1}{N_{ca}} \sum_{i=0}^{N_{ca}-1} x_{[N_{ca}M+l+iM]} \qquad M \le n < 2M \qquad 0 \le l < M$$
$$= \frac{1}{N_{ca}} \sum_{i=0}^{N_{ca}-1} x_{[N_{ca}M+(n-M)+iM]} \qquad M \le n < 2M \qquad (2.51)$$

El último periodo de la señal promediada coherentemente será aquel donde n está entre $(N_{ma} - 1)M$ y $N_{ma}M$. Para obtener esta señal, el promedio debe comenzar desde $(N_{ma} - 1)N_{ca}M$ y continuar durante N_{ca} periodos. Ahora, la posición relativa dentro del periodo se da por $l = n - (N_{ma} - 1)M$.

Luego, la expresión general para Y_{ca} resulta:

$$Y_{ca}[n] = \begin{cases} \frac{1}{N_{ca}} \sum_{i=0}^{N_{ca}-1} x_{[n+iM]} & 0 \le n < M \\ \frac{1}{N_{ca}} \sum_{i=0}^{N_{ca}-1} x_{[N_{ca}M+(n-M)+iM]} & M \le n < 2M \\ \frac{1}{N_{ca}} \sum_{i=0}^{N_{ca}-1} x_{[2N_{ca}M+(n-2M)+iM]} & 2M \le n < 3M \\ \vdots & \vdots \\ \frac{1}{N_{ca}} \sum_{i=0}^{N_{ca}-1} x_{[(N_{ma}-1)N_{ca}M+(n-(N_{ma}-1)M)+iM]} & (N_{ma}-1)M \le n < N_{ma}M \end{cases}$$
(2.52)

Esta ecuación se puede reescribir empleando dos sumatorias y un factor condicional, obteniendo la expresión:

$$Y_{ca[n]} = \frac{1}{N_{ca}} \sum_{h=0}^{N_{ma}-1} \sum_{i=0}^{N_{ca}-1} x_{[hN_{ca}M+(n-hM)+iM]} \cdot (hM \le n < (h+1)M)$$
(2.53)

Ahora, esta señal promediada coherentemente ingresa a la detección *Lock-in* con un filtro de media móvil (MAF) de N_{ma} periodos. Para esto, primero se multiplica por la señal periódica $s_{[n]}$ y luego se promedia durante N_{ma} periodos. La ecuación 2.54 resume la operación sobre la señal entrante $x_{[n]}$ para la combinación de promedio coherente y detección *Lock-in* con MAF.

$$Y_{2[n]} = \frac{1}{MN_{ma}} \sum_{k=n}^{n+N_{ma}M-1} s_{[k]} \frac{1}{N_{ca}} \sum_{h=0}^{N_{ma}-1} \sum_{i=0}^{N_{ca}-1} x_{[k+M(i+h(N_{ca}-1))]} \cdot (hM \le k < (h+1)M) \quad (2.54)$$

En las siguientes secciones, se mostrará que las expresiones 2.49 y 2.54 (señales $Y_{1[n]} e Y_{2[n]}$) son matemáticamente equivalentes siempre y cuando $N_{ca}N_{ma} = N$. Aunque la demostración se realiza con el *Lock-in* de una fase, puede aplicarse al *Lock-in* de dos fases sin perder generalidad, ya que los cálculos son los mismos, con una señal de referencia desplazada 90° de $s_{[N]}$.

2.4.1.2. Aproximación intuitiva

Para entender de manera intuitiva de por qué estas expresiones son equivalentes, se realizará una expansión de la ecuación 2.49.

$$Y_{[n]} = \frac{1}{MN} \sum_{k=n}^{n+NM-1} x_{[k]} s_{[k]} =$$

$$= \frac{1}{MN} (s_{[n]} x_{[n]} + \dots + s_{[n+M-1]} x_{[n+M-1]} + \dots$$

$$+ s_{[n+M]} x_{[n+M]} + \dots + s_{[n+2M-1]} x_{[n+2M-1]} + \dots$$

$$\vdots$$

$$+ s_{[n+(N-1)M]} x_{[n+(N-1)M]} + s_{[n+NM-1]} x_{[n+NM-1]}) (2.55)$$

Dado que la señal $s_{[n]}$ es periódica con periodo M, $s_{[n]} = s_{[n+M]} = s_{[n+(N-1)M]}$, la ecuación 2.55 se puede factorizar según:

$$Y_{[n]}MN = s_{[n]}(x_{[n]} + x_{[n+M]} + \dots + x_{[n+(N-1)M]}) + \dots + s_{[n+1]}(x_{[n+1]} + \dots + x_{[n+1+(N-1)M]}) + \dots$$

$$\vdots + s_{[n+M-1]}(x_{[n+M-1]} + x_{[n+M-1+M]} + x_{[n+M-1+(N-1)M]})$$
(2.56)

El factor que se multiplica por cada $s_{[n]}$ es precisamente el promedio coherente de la señal $x_{[n]}$

durante N periodos. Por lo tanto, la expresión se puede reescribir como:

$$Y_{[n]}MN = s_{[n]} \sum_{i=0}^{N-1} x_{[n+iM]} + \dots + s_{[n+1]} \sum_{i=0}^{N-1} x_{[n+1+iM]} + \dots + s_{[n+M-1]} \sum_{i=0}^{N-1} x_{[n+M-1+iM]}$$
(2.57)

Tomando una sumatoria externa, la ecuación anterior se puede expresar como:

$$Y_{[n]} = \frac{1}{MN} \sum_{k=n}^{n+M-1} s_{[k]} \sum_{i=0}^{N-1} x_{[k+iM]}$$
(2.58)

Lo cual es la ecuación 2.54 en el caso particular de que $N_{ma} = 1$ y $N_{ca} = N$. Aunque esto no demuestra que las ecuaciones 2.49 y 2.54 sean equivalentes para todos los casos, si da una idea intuitiva de por qué se cumple la igualdad.

2.4.1.3. Demostración formal

Para demostrar la equivalencia entre las ecuaciones 2.49 y 2.54, se expandirá la expresión 2.54.

$$Y_{1[n]} = \frac{1}{NM} \sum_{k=n}^{n+NM-1} x_{[k]} s_{[k]}$$
(2.49)

$$Y_{[n]} = \frac{1^{n+N_{ma}M-1}}{MN_{ma}} \sum_{k=n}^{N_{ma}M-1} \frac{1}{N_{ca}} \sum_{h=0}^{N_{ma}-1} \sum_{i=0}^{N_{ca}-1} x_{[k+M(i+h(N_{ca}-1))]} \cdot (hM \le k < (h+1)M)$$
(2.54)

$$Y_{[n]}MN_{ca}N_{ma} = \sum_{k=n}^{n+N_{ma}M-1} \sum_{h=0}^{N_{ma}-1} \left(x_{[k+Mh(N_{ca}-1)]} + x_{[k+M(1+h(N_{ca}-1))]} + \dots + x_{[k+M((N_{ca}-1)+h(N_{ca}-1))]} \right) \cdot (hM \le k < (h+1)M)$$

$$\begin{split} & {}^{n+N_{ma}M-1} = \sum_{k=n}^{n+N_{ma}M-1} s_{[k]}((x_{[k]} + x_{[k+M]} + \ldots + x_{[k+(N_{ca}-1)M]}) \cdot (0 \leq k < M) + \ldots \\ & + (x_{[k+(N_{ca}-1)M]} + x_{[k+N_{ca}M]} + \ldots + x_{[k+2(N_{ca}-1)M]}) \cdot (M \leq k < 2M) + \ldots \\ & \vdots \\ & + (x_{[k+M((N_{ma}-1)(N_{ca}-1))]} + \ldots \\ & + x_{[k+M((N_{ca}-1)+(N_{ma}-1)(N_{ca}-1))]}) \cdot ((N_{ma}-1)M \leq k < N_{ma}M)) \\ & = s_{[n]}(x_{[n]} + x_{[n+M]} + \ldots + x_{[n+(N_{ca}-1)M]}) + \ldots \\ & + s_{[n+1]}(x_{[n+1]} + x_{[n+1+M]} + \ldots + x_{[n+1+(N_{ca}-1)M]}) + \ldots \\ & + s_{[n+M]}(x_{[n+M+M(N_{ca}-1)]} + \ldots + x_{[n+M+2(N_{ca}-1)M]}) + \ldots \\ & \vdots \\ & + s_{[n+N_{ma}M-1]}(x_{[n+N_{ma}M-1+M(N_{ma}-1)(N_{ca}-1)]} + \ldots \\ & + x_{[n+N_{ma}M-1]}(x_{[n+N_{ma}M-1+M(N_{ma}-1)(N_{ca}-1)]}) + \ldots \\ \end{split}$$

Ahora se aprovechará la periodicidad de $s_{[n]}$. Dado que $s_{[n]} = s_{[n+M]}$, entonces $s_{[n]}x_{[n+M]} = s_{[n+M]}x_{[n+M]}$.

$$Y_{[n]}MN_{ca}N_{ma} = s_{[n]}x_{[n]} + s_{[n+M]}x_{[n+M]} + \dots + s_{[n+(N_{ca}-1)M]}x_{[n+(N_{ca}-1)M]}) + \\ + s_{[n+1]}x_{[n+1]} + s_{[n+1+M]}x_{[n+1+M]} + \dots + s_{[n+1+(N_{ca}-1)M]}x_{[n+1+(N_{ca}-1)M]} + \\ \vdots \\ + s_{[n+N_{ma}M-1+M(N_{ma}-1)(N_{ca}-1)]} + \dots + s_{[n+N_{ma}M-1+M(N_{ma}-1)(N_{ca}-1)]} + \dots + s_{[n+N_{ma}M-1+M((N_{ca}-1)+(N_{ma}-1)(N_{ca}-1))]} + \\ \vdots \\ x_{[n+N_{ma}M-1+M((N_{ca}-1)+(N_{ma}-1)(N_{ca}-1))]} + \dots + s_{[n+N_{ma}M-1+M((N_{ca}-1)+(N_{ma}-1)(N_{ca}-1))]} + (2.60)$$

En cada término de esta última sumatoria, el índice de $s_{[n]}$ siempre es igual al índice de $x_{[n]}$. Ninguno de los índices se repite y todos se suman, desde n hasta $k_{final} = n + N_{ma}M - 1 + M((N_{ca} - 1) + (N_{ma} - 1)(N_{ca} - 1))$. Este k_{final} se puede expresar de la siguiente manera:

$$k_{final} = n + N_{ma}M - 1 + M((N_{ca} - 1) + (N_{ma} - 1)(N_{ca} - 1)) =$$

= $n + N_{ma}M - 1 + MN_{ca} - M + MN_{ma}N_{ca} - MN_{ma} - MN_{ca} + M$
= $n - 1 + MN_{ma}N_{ca}$

por lo que la ecuación 2.60 se puede reescribir como:

$$Y(n) = \frac{1}{MN_{ca}N_{ma}} \sum_{k=n}^{n+MN_{ma}N_{ca}-1} s_{[k]}x_{[k]}$$

lo cual es la ecuación 2.49 en el caso especial donde $N = N_{ma}N_{ca}$, completando la prueba. Como se ha mencionado al principio de esta sección, esta demostración es original y está publicada en [54].

2.4.1.4. Número de operaciones

Aunque, desde un punto de vista matemático, las ecuaciones 2.49 y 2.54 son equivalentes, no involucran las mismas operaciones. El término constante $\frac{1}{N}$ y los cálculos finales de amplitud y fase se ignorarán en las siguientes deducciones, ya que no es necesario calcularlos en tiempo real, sino que pueden calcularse después de que se obtenga y procese toda la señal.

En el caso del primer método, el enfoque clásico de *Lock-in* con MAF durante $N = N_{ma}N_{ca}$, se requieren MN multiplicaciones y MN - 1 sumas.

Para el segundo método; el promedio coherente de N_{ca} periodos de la señal, seguido de *Lock-in* con MAF de N_{ma} periodos, cada muestra de la señal promediada coherentemente requiere $N_{ca} - 1$ sumas. Se necesitan MN_{ma} muestras para el *Lock-in*, por lo que obtener la señal promediada coherentemente requiere $MN_{ma}(N_{ca}-1)$ sumas. Luego, esta señal entra en el *Lock-in*, que requiere MN_{ma} multiplicaciones y $MN_{ma} - 1$ sumas.

En total, esto implica MN_{ma} multiplicaciones y $MN_{ma}(N_{ca}-1)+N_{ma}M-1=MN_{ma}N_{ca}-1=NM-1$ sumas. Esto significa que ambos métodos requieren la misma cantidad de sumas, pero el segundo reduce drásticamente la cantidad de multiplicaciones necesarias, de $MN_{ma}N_{ca}$ a MN_{ma} .

Las ventajas computacionales de utilizar el promedio coherente antes de la amplificación Lock-in son aún más claras para el Lock-in de dos fases. En este caso, la señal arbitraria se multiplica por una señal de referencia adicional cosenoidal. Si se realizaran los cálculos con un Lock-in tradicional con MAF, la ecuación 2.49 debería calcularse dos veces, una con cada señal de referencia. Esto significa 2MN multiplicaciones y 2(MN - 1) sumas.

Por otro lado, la señal promediada coherentemente es la misma para las dos referencias, por lo que solo se debe calcular una vez $(MN_{ma}(N_{ca}-1) \text{ sumas})$. Luego, esta señal promediada coherentemente entra en el *Lock-in* de dos fases, que requiere $2MN_{ma}$ multiplicaciones y $2(MN_{ma}-1)$ sumas, para un total de $MN_{ma}(N_{ca}-1)+2(MN_{ma}-1)=MN_{ma}(N_{ca}+1)-2$ sumas, casi la mitad de las sumas cuando N_{ca} es suficientemente grande. Estos resultados se resumen en el Cuadro 2.3.

De este análisis se desprende que, para mantener el número de multiplicaciones al mínimo, N_{ma}

Algoritmo	Número de sumas	Número de productos
LIA con MAF de		
$N_{ma}N_{ca}$ periodos	$MN_{ma}N_{ca} - 1$	$MN_{ma}N_{ca}$
Una fase		
CA de N_{ca} periodos +		
LIA de N_{ma} periodos	$MN_{ma}N_{ca} - 1$	MN_{ma}
Una fase		
LIA con MAF de		
$N_{ma}N_{ca}$ periodos	$2(MN_{ma}N_{ca}-1)$	$2MN_{ma}N_{ca}$
Dos fases		
CA de N_{ca} periodos +		
LIA de N_{ma} periodos	$MN_{ma}(N_{ca}+1)-2$	$2MN_{ma}$
Dos fases		

Cuadro 2.3: Número de operaciones

*Cada periodo de la señal de referencia tiene M muestras

N	$A_{\rm LI}$	LI n°	$A_{\rm CALI}$	CALI n°	σ_o	σ_o
		de productos		de productos		esperado
2	1.0183	512	1.0183	256	0.21474	0.25516
4	0.97311	1024	0.97311	256	0.16137	0.18042
8	1.0012	2048	1.0012	256	0.12147	0.12758
16	0.99393	4096	0.99393	256	0.090401	0.090211
32	1.0089	8192	1.0089	256	0.062989	0.63789
64	1.002	16384	1.002	256	0.044055	0.045105
128	0.99883	32768	0.99883	256	0.031676	0.031894
256	1.0004	65536	1.0004	256	0.021749	0.022553
512	0.99962	131072	0.99962	256	0.016181	0.015947
1024	0.99945	262144	0.99945	256	0.011478	0.011276

Cuadro 2.4: Resultados de simulación

*Para todos los casos A = 1, M = 128 y $\sigma_i = 2,8868$

 $^{**}\sigma_o$ y la amplitud fueron calculados a partir de 100 realizaciones

*** σ_o esperado es calculado con la expresión 2.34

debe ser lo más pequeño posible. El caso más conveniente es dejar todo el promedio al sistema CA, estableciendo $N_{ca} = N$ y $N_{ma} = 1$. Este es, de hecho, el caso de la deducción de la sección 2.4.1.2. Las desventajas de este método son que la señal CA necesita memoria para ser almacenada, que la convergencia de la solución es más lenta y que se requiere un ciclo adicional en los cálculos, debido a los M ciclos de reloj necesarios para realizar las multiplicaciones y el MAF una vez que se ha adquirido y promediado coherentemente toda la señal. Estas conclusiones se pueden aplicar a cualquier sistema de *Lock-in*, siempre y cuando utilice MAF como filtro pasa-bajos.

2.4.1.5. Simulaciones

Para mostrar la equivalencia entre el Lock-in clásico con MAF de N ciclos (en adelante LI) y la combinación de promediación coherente de N ciclos y Lock-in con MAF de 1 ciclo (en adelante CALI), y simultáneamente corroborar la ecuación 2.34 se realizaron algunas simulaciones. Para estas se consideró una señal sinusoidal de amplitud A = 1 y M = 128 puntos por ciclo, perturbada con ruido uniforme de desviación estándar σ_i como entrada del sistema. Para la referencia, se utilizaron ondas seno y coseno, calculadas con M puntos por ciclo y amplitud A_{ref} . Bajo estas condiciones, los algoritmos LI y CALI pueden usarse para estimar la amplitud de la señal subyacente. Dependiendo de la elección de N, esta estimación será más o menos precisa, pero será siempre la misma con ambos métodos. Para enfatizar esto, se examinarán la Figura 2.12 y el Cuadro 2.4.

La Figura 2.12 muestra la estimación de la amplitud de la señal utilizando ambos algoritmos a medida que avanza el tiempo y la señal ingresa al algoritmo, una muestra en cada pulso de reloj. El resultado final es el mismo, como se esperaba, y está muy cerca del valor esperado de A = 1, pero los cálculos intermedios son bastante diferentes. El LI converge al resultado de manera más gradual y proporciona los resultados un ciclo antes. La desventaja de este enfoque es la cantidad requerida de multiplicaciones, como se señaló en secciones anteriores. El CALI, por otro lado, toma un ciclo adicional y converge abruptamente a los resultados.

El Cuadro 2.4 muestra las salidas de los algoritmos para diferentes valores de N con una SNR fija. Para la estimación de la amplitud, se tomaron 100 realizaciones de los cálculos y se tomó el



Figura 2.12: Ejemplo de la estimación de la amplitud de una señal sinusoidal con $\sigma_i = 0,2887$ utilizando el *Lock-in* convencional (LI) y el promedio coherente seguido de *Lock-in* (CALI) con N = 4.

valor promedio de los resultados, etiquetado como $A_{\rm LI}$ y $A_{\rm CALI}$ para cada algoritmo. También se tomó la desviación estándar de estas mediciones y se etiquetó como σ_o . En todos los casos, tanto LI como CALI arrojan los mismos resultados, y los valores de σ_o se comportan como se esperaba de la ecuación 2.34.

El código para replicar estas simulaciones es abierto, parametrizable y está disponible en [62].

2.5. Conclusiones

En este capítulo se han descripto los sistemas de detección coherente de señales, poniendo énfasis en las técnicas de *Lock-in* y promediación coherente, que son las que se desarrollarán a lo largo de esta tesis. Se ha mostrado analíticamente, y luego en simulaciones, que al utilizar la técnica de *Lock-in* la desviación estándar del ruido en la salida disminuye proporcionalmente con la raíz cuadrada del ancho del filtro pasa-bajos utilizado, con lo cual la elección de éste resulta un factor clave en su diseño. En particular, se ha encontrado que si se utiliza un filtro de media móvil de N ciclos de señal como filtro pasa-bajos el *Lock-in* reduce en $\sqrt{\frac{NM}{2}}$ el ruido de la señal, donde M es la cantidad de muestras de la señal que entran en un ciclo.

Seguidamente, se ha analizado el efecto del error de cuantización cometido durante el muestreo de las señales. Estos efectos resultan predecibles, constantes y se han acotado en $\Delta A_s < \sqrt{2} \cdot 2^{-B}$ para los casos en que $f_m = kf_s$, donde f_m es la frecuencia de muestreo del sistema y f_s la frecuencia de la señal incógnita. Para los casos en que $f_m \neq kf_s$ estos efectos resultan menos predecibles, pero pueden ser reducidos afinando el filtro pasa-bajos utilizado (incrementando N para el caso de filtro MA).

Luego se ha demostrado que la combinación del promedio coherente de N_{ca} ciclos de una señal de interés y un sistema de *Lock-in* convencional con filtro de media móvil de N_{ma} ciclos produce los mismos resultados que un *Lock-in* de N ciclos de la señal, siempre y cuando $N = N_{ca}N_{ma}$. Aunque ambos métodos proporcionan los mismos resultados, los cálculos necesarios para obtener-los son bastante diferentes. Específicamente, el número de multiplicaciones necesarias se reduce a medida que N_{ca} aumenta (y N_{ma} disminuye). El caso más eficiente se logra cuando $N_{ca} = N$ y $N_{ma} = 1$, donde el número de multiplicaciones requeridas se fija en M, independientemente del N seleccionado.

Estos conceptos pueden aplicarse a cualquier sistema de *Lock-in* que utilice filtro de media móvil. En particular en esta tesis se utilizarán para desarrollar esquemas eficientes de detección coherente de señales, centrados en dispositivos FPGA y SoC-FPGA de gama media.

Capítulo 3

Procesamiento de señales en SoC-FPGA

3.1. Introducción

En los últimos años se ha evidenciado una creciente demanda de capacidad computacional. Ésta ha motivado el desarrollo de nuevas arquitecturas y sistemas computacionales, entre los cuales se destacan las soluciones basadas en arreglos de celdas lógicas programables (conocidas como FPGAs, de su nombre en inglés *field programmable gate array*). Este tipo de dispositivos se destacan por su alta capacidad de reconfigurabilidad, elevadas prestaciones y alta eficiencia energética y, en los últimos años, los chips que los integran han evolucionado para incluir distintos sub-sistemas que complementan sus capacidades. Así han surgido sistemas conocidos como SoC-FPGA, que integran una FPGA con bloques de memoria embebidos, lazos de enganche de fase (PLLs), bloques específicos de procesamiento de señales (usualmente conocidos como bloques DSP, del inglés *digital signal processing*) e incluso microprocesadores.

En particular, un sistema SoC-FPGA con un microprocesador embebido resulta especialmente apropiado para implementar sistemas de procesamiento de señal como en el que se inscribe este trabajo de investigación. En este esquema, es la FPGA quien se encarga del procesamiento en tiempo real, incluyendo el control de los sistemas de adquisición y generación de señales analógicas, mientras que el microprocesador se encarga del control de la operación y la interfaz de usuario.

Aunque las plataformas SoC-FPGA evolucionan, la programación y optimización de las aplicaciones que las utilizan sigue siendo un proceso extenso y complicado, que requiere de diseñadores muy especializados en el diseño digital y en la arquitectura particular de la plataforma objetivo.

El flujo de trabajo habitual al diseñar una aplicación en un sistema de este tipo comienza con el diseño del hardware a nivel de transferencia de registros (o RTL, del inglés *register transfer level*) en algún lenguaje de descripción de hardware (o HDL, del inglés *hardware description lenguaje*) como Verilog o VHDL, con la asistencia de alguna herramienta de simulación para su verificación. En este proceso debe tenerse especial cuidado cuando se quieren utilizar elementos como lazos de enganche de fase (PLLs) o bloques especiales de procesamiento de señal, ya que el HDL debe escribirse correctamente para que el compilador pueda inferir en forma correcta los bloques necesarios. Luego se utilizan los compiladores provistos por los distintos fabricantes (Quartus de Intel-Altera o Vivado de AMD-Xilinx, por ejemplo) para generar el *bitstream* necesario para programar la FPGA. Este debe testearse experimentalmente, para solucionar posibles problemas que no hayan aparecido en la

simulación. Adicionalmente, en los SoC-FPGA que incluyen un microprocesador, la programación de este elemento de procesamiento debe realizarse independientemente, con código en algún lenguaje de alto nivel como C, C++, Python, etc. Este código debe compilarse para el microprocesador objetivo, ya sea transcompilándolo con herramientas provistas por el fabricante o compilándolo nativamente en el microprocesador. Por último, todo el sistema debe verificarse. Todo este proceso es tedioso y resulta un territorio fértil para la proliferación de todo tipo de errores. Además, al involucrar varios sistemas interconectados, muchas veces resulta difícil localizar la fuente de estos errores.

Con el objetivo de aliviar este flujo de trabajo han surgido en los últimos años las llamadas herramientas de síntesis en alto nivel (o HLS del inglés *high level syntesis tools*). Estas permiten generar código HDL a partir de código en C/C++/OpenCL [92], [93], [94], [95]. Si bien estos lenguajes generan códigos de programación secuencial, permiten implementar algoritmos paralelizados utilizando distintas directivas de computación. Esto simplifica en gran medida el diseño de algoritmos directamente implementables en la FPGA, pero no simplifican el resto de las etapas involucradas en el diseño del sistema.

En este capítulo, se describirá la arquitectura de un sistema SoC-FPGA reutilizable, destinado en específico al procesamiento de señales. Un sistema de este tipo debe recibir la señal objetivo, con algún conversor analógico- digital (ADC) apropiado, luego procesarla mediante algún algoritmo y finalmente mostrar los resultados al usuario. El diseño presentado incluye distintas opciones para las entradas de señal, que pueden ser analógicas, provenientes de algunos ADCs comúnmente utilizados, o puramente digitales, generadas internamente en la FPGA. También incluye un modelo que deben cumplir las etapas de procesamiento para interactuar con el sistema, mostrando ejemplos concretos de aplicación, y distintos módulos que pueden utilizarse para controlar la operación y recuperar los resultados. El objetivo del diseño es que el programador solo deba concentrarse en la aplicación de sus algoritmos de procesamiento de señal escribiendo su HDL o utilizando las herramientas HLS, sin preocuparse en la integración de los distintos sub-sistemas de interés. Finalmente se mostrará la utilidad de este esquema genérico de procesamiento de señales mediante un ejemplo: un sistema que muestrea una señal, la somete a un filtro de respuesta impulsional finita (filtro FIR, por sus siglas en inglés), y luego la convierte de nuevo en una señal analógica.

El diseño fue implementado en una FPGA Cyclone V de Intel-Altera. Para la programación de HDLs se utilizó Verilog con el agregado de algunos bloques de propiedad intelectual (IP) de Intel-Altera, incluidos en sus licencias gratuitas, mientras que para las herramientas desarrolladas en el elemento de control se utilizó C/C++. El diseño es de código libre y esta disponible en [96], junto con un archivo *readme* que explica su uso. Una versión modificada de esta implementación ha sido exitosamente adaptada a una plataforma Red Pitaya de AMD-Xilinx, lo que muestra que los conceptos desarrollados son transferibles a otras arquitecturas. Esta implementación está disponible en [97] y se describirá brevemente sobre el final de este capítulo.

La mayor parte de los resultados de este capítulo fueron presentados en [98].

3.2. Cyclone V SoC-FPGA

La plataforma elegida para una primera implementación del sistema es un Cyclone V SoC 5CSEMA5F31, montado en una plataforma DE1-SoC desarrollada por Terasic Inc. [99]. Esta es una plataforma de diseño robusta fabricada alrededor del System-on-Chip (SoC) FPGA Cyclone V, que combina un procesador de doble núcleo Cortex-A9 con un arreglo de celdas lógicas programables, con ambos elementos de procesamiento interconectados por buses de gran ancho de



Figura 3.1: Vista superior de la plataforma DE1-SoC [99]

banda. Adicionalmente la placa DE1-SoC incorpora memoria DDR3 de alta velocidad, capacidades de video y audio y conversor analógico digital, entre otras características. En la Figura 3.1 se puede apreciar la vista superior de esta placa de desarrollo, con todos los elementos de interés señalados.

Este dispositivo cuenta con los recursos mostrados en el Cuadro 3.1. Las ALMS y ALUTs son los elementos lógicos que la FPGA utiliza para mapear las operaciones lógicas en hardware, mientras que los M10K son bloques de memoria especializados integrados, con 10 kbits de memoria cada uno. Además de estos recursos, la plataforma DE1-SoC cuenta con cuatro osciladores de 50 MHz que pueden utilizarse con los PLL internos del SoC para generar relojes de hasta 800 MHz.

Una característica interesante del Cyclone V y otros dispositivos FPGA son sus bloques DSP (siglas provenientes del inglés, *digital signal processing*). Estos bloques permiten implementar operaciones de procesamiento de señales digitales como multiplicaciones de manera eficiente pero, como se puede observar en el Cuadro 3.1, son un recurso limitado.

3.2.1. Placa de expansión Highspeed AD/DA daughter card (ADA-GPIO)

Si bien la plataforma DE1-SoC cuenta con un ADC LTC2308 (500 kmps y 12 bits de resolución), este no alcanza las velocidades requeridas para las aplicaciones deseadas. Además, no posee conversores digitales analógicos (DAC), que también resultan necesarios para el procesamiento coherente de señales. Para solucionar ambos problemas se incorporó al sistema una placa complementaria: *Highspeed ADC/DAC daughter card*, con interfaz GPIO (ADA-GPIO), desarrollada por Terasic

ALMs	32070
Registros Lógicos	64,140
ALUTS	64140
Bits de memoria	4065280
M10K	397
Bloques DSP	87
PLLs	6

Cuadro 3.1: Recursos totales de la plataforma Cyclone V

[100]. Este sistema integra dos canales de conversión analógica digital de 14 bits y 65 Msps de velocidad máxima, resueltos a través del chip AD9248 y dos canales de conversión digital analógica de 14 bits y 125 MHz de velocidad máxima, implementados con el chip AD9767. Adicionalmente provee una etapa de acoplamiento en alterna de la señal.

La conexión de esta placa a la plataforma DE1-SoC es directa, a través de sus conectores de propósitos generales (GPIO), como se ve en la figura 3.2.



Figura 3.2: Conexión del ADA-GPIO al DE1-SoC [100]

3.2.2. Acoplamiento en continua de la placa Highspeed AD/DA Card

El conversor AD/DA de alta velocidad de Terasic tiene un inconveniente importante para las aplicaciones abordadas en este trabajo de investigación: está acoplado en alterna, a través de un transformador que atenúa toda señal por debajo de 1 MHz. Tanto los canales de AD como DA utilizan los mismos transformadores, en una configuración que se puede ver, para el canal B, en la Figura 3.3. En esta también se puede ver que los circuitos incorporan una adaptación de señal diferencial (entrada de los conversores) a *single-ended* (conectores accesibles indicados como Jack-SMA).

Para solucionar este inconveniente se decidió acoplar en continua los conversores. Para esto se retiraron las resistencias R11 y R12 en el circuito del ADC, y el transformador T6 en el circuito del DAC. De esta manera queda disponible la salida en tensión diferencial del DAC (pines 1 y 3), y las entradas diferenciales de tensión del ADC (pines VINp_B y VINn_B).



Figura 3.3: Acoplamiento en alterna [101]



Figura 3.4: Circuito de acoplamiento en continua para el DAC AD9767

A continuación, se diseñó el circuito para manejar los conversores. Para el DAC se implementaron dos buffers utilizando un amplificador operacional doble OPA2350, como se muestra en la Figura 3.4. Para el ADC se diseño un circuito controlador basado en el amplificador operacional full- diferencial THS4521, y dos OPA350 como buffers, como se muestra en la Figura 3.5.

El circuito del ADC incorpora un filtro *anti-aliasing* a través de las resistencias R5 y R6 y el capacitor C3 para evitar que el ruido de alta frecuencia ingrese en el ADC. Con estos valores, la frecuencia de corte del filtro es de 29 MHz, seleccionados en esta etapa del trabajo para minimizar el ruido de alta frecuencia. Si se requiere alcanzar todo el ancho de banda disponible en los conversores,



Circuito ADC

Figura 3.5: Circuito de acoplamiento en continua para el ADC AD9248

esta frecuencia podría relajarse. Por otro lado, podrían incorporarse filtros adicionales al sistema mediante los capacitores C1 y C2, que en esta configuración se han dejado desconectados. Además, los amplificadores operacionales utilizados atenúan la señal a frecuencias altas debido a su producto ganancia-ancho de banda (GBP) de 38 MHz, lo que limita la medición precisa de señales de muy alta frecuencia. Para alcanzar frecuencias superiores, sería necesario reemplazar los amplificadores por otros con mayor GBP y relajar el filtro *anti-aliasing*. Sin embargo, estas optimizaciones no resultan necesarias para las aplicaciones pretendidas en este trabajo de investigación.

3.3. Sistema reutilizable de procesamiento de señales

3.3.1. Esquema general del diseño

El diseño propuesto se compone por tres etapas: etapa de control, etapa de fuente de señal y etapa de procesamiento, resumidas en la figura 3.6.

La etapa de control tiene a su cargo la configuración inicial del sistema. Esto incluye ajustar un PLL y un divisor del reloj a partir de un oscilador externo de 50 MHz, y establecer los parámetros necesarios para el procesamiento. Éstos pueden ser, por ejemplo, la cantidad de ciclos de integración de una señal o los coeficientes de un filtro y pueden modificarse en tiempo de ejecución por el usuario. Además, esta etapa inicia el proceso mediante una señal de habilitación (enable) y espera a la señal de finalización generada por la etapa de procesamiento. Una vez completada la operación, los resultados quedan disponibles en memorias tipo FIFO (*first In, first Out*), útiles tanto para algoritmos que requieren historial de señal como para aquellos que solo utilizan un resultado puntual. El diseño permite trabajar con señales de 32 o 64 bits, y puede ser comandado tanto por el microprocesador integrado en el SoC como por un procesador NIOS implementado en la FPGA, dependiendo de la plataforma disponible. Para ambos casos, se han desarrollado capas de abstracción de hardware en C/C++.



Figura 3.6: Esquema general del diseño

La etapa de fuente de señal define las características de las señales que ingresan al procesamiento. Se han desarrollado distintos tipos de fuentes de señal: una completamente digital con nivel de ruido configurable, ideal para probar diversos algoritmos de procesamiento, y otra específica para la gestión de ADCs. En este contexto, se han implementado controladores para dos conversores: el LTC2308, de 12 bits con una frecuencia máxima de muestreo de 500 kHz, utilizado en las plataformas de Terasic, y el AD9248, integrado en la plataforma *High-speed A/D and D/A Development Kit.* Como parte del módulo también se incluye el controlador del conversor digital-analógico AD9767, disponible en esta misma plataforma.

La etapa de procesamiento de señal es en donde el diseñador debe implementar sus algoritmos. Esta puede estar formada a su vez por distintas sub-etapas o sub-sistemas. Se ha estandarizado el modelo que debe cumplir cada una de ellas, donde la salida de cada modulo incluye una señal extra que funciona como habilitación para la etapa siguiente, garantizando la sincronización del proceso. Esta señal, llamada data_valid, e incluida en cada bus que transmite datos, se pone en alto en todo ciclo de reloj en que el resultado sea válido, para que se registre correctamente en la etapa siguiente. Este esquema, que se utiliza en todas las etapas del procesamiento, es una versión simplificada del conocido en la documentación de Intel-Altera como interfaz Avalon Strea-ming (Avalon-ST). Las etapas pueden parametrizarse en tiempo de ejecución, a través los distintos parámetros configurables, y una vez finalizado el procesamiento deben poner en alto la señal de finalización para al módulo de control de la disponibilidad de los resultados.

En las siguientes secciones se profundizará sobre las distintas etapas incluidas en el diseño.

3.3.2. Etapa de control

La etapa de control puede subdividirse en dos elementos bien diferenciados. Por un lado está el hardware diseñado en la FPGA, representado en la Figura 3.7. Este contiene elementos para configurar la operación de procesamiento, controlar su velocidad y flujo y recopilar los resultados disponibles. Por otro, está el elemento de control en sí, que puede ser un microprocesador embebido en el SoC, o un procesador programado en las celdas lógicas de la FPGA.

El reloj principal del sistema se genera mediante un PLL reconfigurable, disponible dentro del chip Cyclone V. Para instanciarlo correctamente se utilizan dos bloques de IP provistos por Intel-Altera: Altera PLL y Altera PLL reconfig. Estos bloques permiten generar, a partir de un reloj de 50 MHz, uno de una frecuencia personalizada, comprendida en el rango 1-65 MHz. Adicionalmente se provee un divisor del reloj, para poder implementar frecuencias menores. El usuario puede utilizar el elemento de control para gestionar el reloj por medio de funciones definidas en una capa de abstracción de hardware escrita en lenguaje C: configurarPLL(int) y setClockDivider(int). En adelante se referirá a esta con el acrónimo HAL, del inglés *Hardware abstraction layer*.

La etapa de control de parámetros esta pensada para que el usuario pueda parametrizar el procesamiento en tiempo real (parámetros de salida), y para que la etapa de procesamiento pueda comunicar algún resultado auxiliar al procesador (parámetros de entrada). Estos últimos pueden ser, por ejemplo, la cantidad de veces que encontró un determinado valor, o una bandera que avise sobre algún evento. Para que el elemento de control pueda leer y escribir correctamente los parámetros de entrada y salida de los módulos es necesario adaptar la lógica a interfaces Avalon Memory Mapped (Avalon-MM), mediante un módulo propio inspirado en templates provistos por Intel-Altera. En este texto no se profundiza en la interfaz Avalon-MM, aunque se destaca que es la que permite al elemento de control, ya sea el procesador embebido o el NIOS, leer correctamente los parámetros; para más información sobre las interfaces de Intel-Altera consultar [102]. En este diseño se incluyeron 4 interfaces de parámetros, cada una con 10 salidas y 6 entradas, lo que da un total de 24 parámetros de entrada y 40 de salida, aunque podrían agregarse fácilmente más al diseño. El usuario puede utilizar el elemento de control para escribir y leer estos parámetros fácilmente mediante las funciones definidas en la HAL setParamX(int,int) y getParamX(int).

Las señales de enable, reset y finalización sirven para controlar el flujo del proceso. La señal de enable da inicio al procesamiento, la señal de finalización le informa al módulo de control



Figura 3.7: Módulo de control



Figura 3.8: Acceso a resultados del procesamiento

que este ha finalizado y la señal de **reset** sirve para reiniciar resultados y vaciar las memorias correspondientes. El usuario puede utilizar el elemento de control para controlar el flujo de programa mediante funciones **setEnable()**, **unsetEnable()**, **Reset()**, **getFin()** y **waitForFin()** definidas en la HAL.

Los resultados del procesamiento entran al módulo en formatos de 32 o 64 bits, y se almacenan en sendas memorias FIFO. Además de los datos a almacenar, la lógica de procesamiento debe incluir la señal data_valid, que se pone en alto en todo ciclo de reloj en que el resultado sea válido. Las memorias FIFO fueron implementadas mediante bloques IP del Altera, con interfaz de entrada Avalon-ST, e interfaz de salida Avalon-MM. Esta interfaz de entrada permite adaptar directamente a las memorias con la lógica propietaria del diseñador, mientras que la elección de interfaces Avalon-MM como salida de los FIFOs permite que el elemento de control pueda leer correctamente las memorias, en forma similar a lo descripto para la etapa de control de parámetros. En total el diseño dispone de 4 memorias, dos para valores de 64 bits y 2 para valores de 32 bits, que pueden utilizarse para almacenar resultados de procesamiento, o señales crudas provenientes de los conversores, redirigiéndolas con parámetros provistos en el código. En la Figura 3.8 se muestra esquemáticamente este procedimiento.

El usuario puede utilizar el elemento de control para acceder a las memorias mediante funciones leer_fifo_32_bit(int), leer_fifo_64_bit(int), leer_resultado_32_bit(int) y leer_ resultado_64_bit(int) definidas en la HAL. En el Cuadro 3.2 se resumen las funciones desarrolladas para la capa de abstracción de hardware.

El elemento de control se incluye en el diseño mediante bloques IP de Intel-Altera. En el caso del procesador *soft* se utilizó la versión gratuita del procesador NIOS. Este es un procesador RISC (del inglés *reduced instruction set computer*, es decir computador con conjunto de instrucciones reducido) de 32 bits optimizado para ahorrar recursos. Cuenta con entradas de datos con interfaces Avalon-

Función	Entradas	Salidas	Descripción
configurarPLL(int X)	Frecuencia objetivo	-	Determina la frecuencia del PLL principal
setClockDivider(int X)	Divisor de la frecuencia	-	Determina el divisor del reloj principal
setParamX(int X,int Y)	X: Posición del parámetro Y: Valor del parámetro	-	Setea el parámetro X con el valor Y
getParamX(int X)	-	Valor del parámetro	Obtiene el valor del parámetro X
setEnable()	-	-	Inicia la operación
unsetEnable()	-	-	Finaliza la operación
Reset()	-	-	Reinicia la operación
getFin()	-	-	Verifica si se terminaron los cálculos
waitForFin()	-	-	Espera a que los cálculos se terminen
$leer_fifo_32_bit(int X)$	FIFO apuntado	Puntero a resultados	Lee resultados almacenados en FIFOs de 32 bits
$leer_fifo_64_bit(int X)$	FIFO apuntado	Puntero a resultados	Lee resultados almacenados en FIFOs de 64 bits
$leer_resultado_32_bit()$	FIFO apuntado	Resultado	Lee resultado actual de 32 bits
$leer_resultado_64_bit()$	FIFO apuntado	Resultado	Lee resultado actual de 64 bits

Cuadro 3.2: Capa de abstracción de hardware (HAL)

MM, por lo que se pueden conectar directamente a él las entradas y salidas de los distintos módulos usando la herramienta de integración del sistema (QSYS) disponible en el software Quartus.

Por otro lado, para conectar el procesador externo, un ARM-Cortex A9, al diseño es necesario utilizar uno de los tres buses integrados en el chip Cyclone V. Estos son: el *FPGA-to-HPS Bridge*, el *HPS-to-FPGA Bridge* y el *Lightweight HPS-to-FPGA Bridge* [102]. Todo el procesamiento en tiempo real del sistema se realiza en la FPGA y la comunicación con el procesador externo no es crítica para la operación, por lo que se seleccionó el *Lightweight HPS-to-FPGA Bridge*. Este puente, recomendado por el fabricante para gestionar tráficos de baja velocidad, expone una interfaz a la que el usuario puede conectar interfaces Avalon-MM, por lo que su conexión puede realizarse directamente, en forma similar al procesador NIOS.



Figura 3.9: Módulo de fuente de señal

3.3.3. Etapa de fuente de señal

La señal de origen para el procesamiento puede ser de tres fuentes distintas: digital, proveniente del ADC LTC2308 (en adelante ADC_2308) o proveniente del AD9248 (en adelante ADC_HS, por *high-speed*). Adicionalmente se diseñó el controlador para el conversor digital-analógico AD9767 (DAC_HS) y un modelo que debería seguir el módulo que controle algún conversor analógico digital distinto que quiera incorporarse al diseño. De estos conversores, el ADC_2308 es el que esta disponible en gran parte de las plataformas de desarrollo de la Cyclone V, mientras que el ADC_HS y el DAC_HS están incorporados en la placa de extensión *Highspeed AD/DA Card* [100]. En esta etapa del trabajo se diseñaron controladores tratando de cubrir la mayor cantidad de opciones, maximizando el uso que pueda tener esta plataforma reutilizable de procesamiento de señales.

El módulo que controla las señales se muestra esquemáticamente en la Figura 3.9, de la que se han omitido por simplicidad las señales que se envían a los dispositivos físicos (ADC y DAC). En forma análoga a lo que se ha comentado previamente, cada una de las señales de datos tiene su correspondiente señal data_valid que permite sincronizar etapas posteriores.

3.3.3.1. Señal digital

La señal digital es una señal sinusoidal que se genera a partir de una tabla de consulta (*look-up table*), con la opción de perturbarla con ruido uniforme. Esta señal, que resulta particularmente útil en la etapa de prueba de los sistemas, proporciona una muestra de señal en cada flanco ascendente de reloj en el cual la señal de **enable** esté en alto. Tiene dos modos de funcionamiento: tabla de consulta directa o síntesis por truncamiento de fase. En el primero de los casos se especifica la cantidad de puntos por ciclo de la sinusoide y se varía su frecuencia mediante la frecuencia del reloj,



Figura 3.10: Síntesis digital directa por truncamiento de fase

mientras que el segundo caso se verá en detalle en la sección 3.3.3.2. Cinco parámetros permiten configurar la operación: el modo de funcionamiento ($modo_func$), la cantidad de puntos por ciclo para el modo de tabla de consulta directa (M), el incremento de fase para el modo de síntesis por truncamiento de fase ($\Delta\Theta$), el método con el que se genera la secuencia pseudoaleatoria que simula el ruido uniforme ($modo_ruido$) y la amplitud de éste (bits_ruido).

3.3.3.2. Síntesis digital directa por truncamiento de fase

Un método común para generar digitalmente una sinusoide emplea un esquema de tabla de consulta. Si esta tiene 2^b muestras y se recorre a una frecuencia de reloj de f_{clk} la señal con mínima frecuencia que se puede generar, es decir la resolución frecuencial de la señal sinusoidal generada, será $\frac{f_{clk}}{2^b}$. Esto implica, por ejemplo, que para generar una señal sinusoidal de frecuencia 10 Hz utilizando un reloj de 65 MHz se necesitarían 6500000 muestras. Suponiendo que cada una ocupa 2 bytes esto representa aproximadamente 12 GB de memoria. Es cierto que no es necesario guardar toda la onda sinusoidal, dada su simetría de cuarto de onda, pero 3 GB de memoria siguen siendo restrictivos y desproporcionados.

Un primer intento de solucionar este problema puede ser modificar la f_{clk} , mediante el PLL reconfigurable, el divisor del reloj y la cantidad de puntos por ciclo, manteniendo este valor dentro de límites aceptables. De esta manera, se genera la sinusoide para el modo de funcionamiento de tabla de consulta directa. Sin embargo, este método no resulta ideal, ya que, o bien desaprovecha la velocidad de procesamiento del sistema, o bien produce dos dominios de reloj distintos (uno para el manejo de señales y uno para el procesamiento), que puede llevar a problemas como la metaestabilidad, que es un fenómeno que se produce cuando se quiere cambiar el valor de un registro muy cerca de un flanco de reloj, violando su tiempo de *setup* o de *hold* y llevando el registro a un valor indeterminado [103].

Para mejorar la resolución de frecuencia sin incrementar el tamaño de la tabla de consulta un método muy utilizado es el de truncamiento de fase [104], que se muestra esquemáticamente en la Figura 3.10. En este esquema se configura un incremento de fase cuantizado en *B* bits, que luego es truncado a un número de b bits (B > b), que es el que se utiliza para direccionar la tabla de consulta, que almacena 2^b muestras. De esta manera se pueden generar ondas sinusoidales de frecuencia arbitraria con una tabla de 2^b muestras, es decir utilizando solo $2\frac{2^b}{4}$ bytes de memoria, suponiendo que cada muestra ocupa 2 bytes. Por otro lado, la resolución frecuencial del sistema queda determinada por un incremento de 1 en el $\Delta\Theta$, pero ahora este está cuantizado en *B* bits, es decir, queda dada por: $\frac{f_{clk}}{2^B}$.

Como desventaja a este método para generar ondas sinusoidales se puede mencionar que la onda sinusoidal obtenida no es numéricamente homogénea de un ciclo a otro para todos los incrementos de fase. Esto trae problemas, por ejemplo, para aplicar directamente en la práctica los resultados de la sección 2.4 para todas las frecuencias de operación. Es por esto que se ha dejado como opción la tabla de consulta directa para generar la sinusoide.

Para generar una onda sinusoidal de frecuencia f_c con este método basta con ingresar al sistema un incremento de fase ($\Delta \Theta$) de:

$$\Delta\Theta = 2^B \frac{f_c}{f_{clk}} \tag{3.1}$$

Por ejemplo, si se eligieran B = 27 bits para el acumulador de fase y $f_{clk} = 65$ MHz la resolución frecuencial resulta aproximadamente en 0,5 Hz, y el incremento de fase para una frecuencia f_c puede calcularse según:

$$\Delta \Theta = 2,064888 f_c \tag{3.2}$$

Como es lógico, la cantidad de muestras de la tabla de consulta (2^b) es la que determina la pureza de la onda sinusoidal subyacente. Un análisis realizado en [104] determina que la amplitud de los armónicos de la onda sinusoidal siempre va a estar por debajo de $6 \cdot b$ en decibeles. En otras palabras, si se requiere una supresión de armónicos de S dB en la forma de onda de salida, referenciada al tono primario de 0 dB, la tabla de consulta debe tener al menos b = S/6 bits. Para un S = 70 dB, por ejemplo, esto implica una tabla de 2^{12} bits, es decir 4096 muestras, que puede lograrse con 2 kB de espacio de almacenamiento.

3.3.3.3. Generación de secuencias pseudo-aleatorias

Dado que generar secuencias verdaderamente aleatorias utilizando dispositivos determinísticos es extremadamente difícil, en este sistema de procesamiento de señales se han utilizado métodos para generar secuencias pseudo-aleatorias, con el propósito de agregar ruido a una señal sinusoidal. Se han ensayado dos métodos: un algoritmo llamado registro de desplazamiento con retroalimentación lineal (o LFSR, del inglés *linear feedback shift register*) [105] y un generador congruencial lineal típico, dado por la ecuación 3.3. En esta el operador % representa la operación módulo, y los parámetros utilizados son: c = 1, a = 69069 y $m = 2^{32}$, que son los parámetros utilizados en versiones antiguas de la librería de GNU C (glibc) [106].

$$N_{i+1} = (aN_i + c) \% m \tag{3.3}$$

Una vez generada la secuencia pseudo-aleatoria de 32 bits, se escala según la amplitud de ruido solicitada. De esta manera, utilizando la señal simulada se pueden generar señales sinusoidales de diversas frecuencias y con variados niveles de relación señal a ruido para probar los algoritmos.

3.3.3.4. Controlador del ADC High-Speed

El AD9248 es un conversor analógico-digital de dos canales, 14 bits y 65 Mmps de frecuencia de muestreo máxima. El diseño incluye un módulo para controlarlo, que provee una muestra en cada flanco ascendente del reloj clk_adc_hs, en el que la señal de enable esté en alto. El reloj puede conectarse por ejemplo al clk_personalizado provisto por la etapa de control. De esta manera, este reloj determina la frecuencia de muestreo a la que opera. Las señales de salida data_adc_hs_a y data_adc_hs_b incluyen su señal respectiva data_adc_hs_valid para conectar directamente los datos de entrada con la etapa de procesamiento o con las memorias FIFO.

Parámetro	Función	Opciones
Modo func.	Define como se genera la secuencia sinusoidal	DDS o LU Table
М	Puntos por ciclo de onda sinusoidal en modo LU Table	0-2048
Fase Sen	Incremento de fase en modo DDS	$\Delta \Theta = 2^B \frac{f_c}{f_{clk}}$
Modo Ruido	Establece como se genera la secuencia pseudo-aleatoria	LFSR o GCL
Amplitud Ruido	Amplitud del ruido en bits	0-24
Frec. Muestreo	Frecuencia de muestreo para ADC 2308	1 Hz - 500 kHz
Selección canal	Selecciona el canal para ADC 2308	Ch A - Ch B

Cuadro 3.3: Resumen de parámetros reconfigurables de módulo de fuente de señal

3.3.3.5. Controlador del DAC High-Speed

El conversor digital-analógico AD9767 es un DAC de 125 Mmps y 14 bits de resolución. El controlador incluido en este diseño permite generar una señal sinusoidal mediante una tabla de consulta almacenada internamente o a través de datos provistos por las entradas digital_data_in y digital_data_in_valid. La frecuencia a la que se actualizan los valores del DAC se puede modificar mediante su reloj, que puede estar conectado al clk_personalizado. De esta manera, o utilizando el método descripto en la sección 3.3.3.2, se pueden generar ondas sinusoidales de frecuencia variable. Si se quieren generar otras formas de onda basta con modificar la tabla de consulta con la que se generan los datos, o ingresar la forma de onda buscada, cuantizada en 14 bits, a través de las entradas de datos digitales.

3.3.3.6. Controlador del ADC LTC 2308

El ADC LTS2308 es un conversor de 8 canales, 12 bits y 500 kmps de frecuencia de muestreo máxima, y es el que está disponible en la serie DE de las plataformas de Terasic. Si bien la frecuencia que puede alcanzar este conversor no es suficiente para algunas de las aplicaciones en la que se inscribe esta tesis, se utilizó en una primera etapa por disponibilidad, por lo que se incluye su controlador en este esquema general de procesamiento.

Para que este controlador funcione correctamente se le debe proveer un reloj de 40 MHz, para lo que se puede usar el clk_personalizado. A diferencia de los módulos que controlan el ADC y DAC *high-speed*, la frecuencia de muestreo no queda determinada por el clk_personalizado, sino que es configurable a través del parámetro f_muestreo. Otro parámetro, ch_sel, permite elegir el canal de conversión.

Los parámetros reconfigurables de la etapa de fuente de señal se resumen en el Cuadro 3.3.

3.3.4. Etapa de procesamiento de señal

Para conectar correctamente las etapas que se encargan del procesamiento de las señales al sistema éstas deben cumplir con una interfaz de entradas y salidas como la que se muestra en la figura 3.11.

Sin importar cual de las fuentes de señal sea la seleccionada, los datos entran a la etapa de procesamiento de señal de a una muestra por cada ciclo de reloj en el que el data_valid esté en alto. Esto permite que las etapas que procesan la señal, que pueden ser tantas como quiera el diseñador, incorporen un *pipeline*, que solo avanza si la señal de data_valid esta en alto. Las N etapas pueden estar divididas en distintos módulos, donde cada uno sincroniza la señal con interfaces Avalon-ST.

Los módulos de procesamiento de señal pueden ser parametrizados desde el elemento de control y pueden proveer salidas auxiliares. Una vez finalizado el procesamiento el módulo debe encargarse de poner en alto la señal de finalización, para notificar al elemento de control. Una señal extra ready_to_calculate le informa al resto de los elementos del sistema que la etapa de procesamiento esta lista para aceptar nuevos cálculos. Esto es útil cuando el módulo tiene que actualizar parámetros, o limpiar buffers internos antes de comenzar a procesar la señal, por ejemplo. A modo de ejemplo se describirá un módulo que implementa un filtro de respuesta impulsional infinita (Filtro FIR) en la señal de entrada.

3.3.4.1. Filtros de respuesta impulsional finita

Los filtros de respuesta impulsional finita, o FIR, se caracterizan porque su salida (Y) depende de las entradas (X) actuales y pasadas, pero no de las salidas en instantes anteriores. En general la expresión de la salida en un instante n de un filtro FIR esta dada por la ecuación 3.4.

$$Y_n = b_0 X_n + b_1 X_{n-1} \dots + b_N X_{n-N}$$
(3.4)

Este tipo de filtro tiene todos los polos en el origen, por lo que son estables, pero tienen la desventaja de necesitar un orden mayor respecto a los filtros de respuesta impulsional infinita (IIR) para cumplir las mismas especificaciones, lo que se traduce en un mayor gasto computacional. Modificando sus coeficientes se pueden generar filtros pasa- altos y pasa- bajos.



Figura 3.11: Módulo de procesamiento

El módulo incluido en el diseño permite configurar un filtro FIR de 32 coeficientes (parámetros b_i), que pueden ser configurados en tiempo de compilación mediante **parameters** de Verilog, o en tiempo de ejecución utilizando los parámetros reconfigurables de la etapa de control. La ecuación 3.5 describe el comportamiento del filtro implementado.

$$Y_n = \sum_{i=n-M}^n b_i x_i \tag{3.5}$$

La operación del módulo que controla este filtro es como sigue:

- La operación empieza en estado de **reset**. En este estado se registran los coeficientes del filtro, provenientes de la etapa de control, y se almacenan en memoria. Se hace de este modo para evitar que estos cambien mientras la señal se esta filtrando.
- Una vez que se sale del estado de **reset** y se recibe la señal de **enable** el procesamiento solo avanza en los ciclos de reloj en que la señal de **data_valid** este en alto. En cada ciclo de reloj que esto se cumpla se registra la señal entrante, se almacena en una memoria con espacio para 32 muestras y se calcula la salida según la ecuación 3.4.
- Cuando ya han pasado más de 32 ciclos con data_valid en alto esto significa que ya se recibieron 32 muestras de la señal entrante. En estas condiciones se empieza a sobrescribir la muestra más antigua, y se pone en alto la señal de finalización para indicarle al módulo de control que la salida filtrada ya es válida.

3.4. Ejemplo de procesamiento de señales en SoC-FPGA

Para probar el sistema, se implementó una configuración como la que se muestra en la Figura 3.12. En ésta, la señal se genera con un *Lock-in* comercial (SR865) y se adquiere con el AD9248 a una frecuencia de muestreo configurable, de hasta 65 MHz. Luego pasa por un filtro FIR de orden 32, con coeficientes configurables.

El microprocesador ejecuta un programa diseñado en C++ que utiliza la HAL para controlar la operación. Este se ejecuta directamente en el sistema operativo del microprocesador: un Linux



Figura 3.12: Configuración de pruebas



Figura 3.13: Interfaz gráfica de usuario (GUI)

con distribución Ubuntu, y puede utilizarse desde un terminal, conectando el SoC-FPGA a una computadora personal a través de una interfaz serie, o directamente desde el SoC-FPGA. Para esta última configuración, que es la seleccionada para esta demostración, se diseñó una interfaz gráfica (GUI) en C# que se ejecuta a través de la implementación Mono del framework .Net [107]. Esta GUI, mostrada en la Figura 3.13, permite al usuario configurar fácilmente la operación e implementa *named pipes* para configurar la FPGA a través de la HAL, como se muestra en la Figura 3.14.

Usando este programa, el usuario puede configurar los coeficientes del filtro y la frecuencia de muestreo. De esta manera, el mismo sistema se puede utilizar para implementar diferentes tipos de filtros, con diferentes frecuencias de corte. Una vez procesada, la señal se alimenta al DAC, para ver las señales de entrada y salida juntas en un osciloscopio, y también se almacena en las memorias FIFO, para que el usuario pueda leerlas directamente en la pantalla de su computadora. La GUI utiliza esta información para trazar fragmentos de la señal, con el objetivo de que el usuario pueda verificar el funcionamiento sin necesidad de equipo adicional. Finalmente, el SR865 se utiliza para medir la amplitud de la señal analógica de salida, con el fin de verificar todo el sistema. Este *Lock-in* se puede controlar y programar en forma remota a través de una conexión Ethernet, para lo cual se diseñó un programa disponible en [108]. Éste puede ser reconfigurado para utilizar en distintos experimentos.

Los coeficientes para esta demostración fueron seleccionados para implementar filtros pasa bajo y pasa alto de diferentes frecuencias de corte normalizadas ($0 < \omega < 1,0$). Los coeficientes ingresan al módulo de procesamiento como números enteros de 16 bits, a través del módulo de control de parámetros. Para convertir los coeficientes proporcionados por algún *toolbox* de procesamiento de



Figura 3.14: Comunicación de los módulos



Figura 3.15: Foto de la interfaz de pruebas

señales (como NumPy de Python, por ejemplo), simplemente hay que multiplicar los coeficientes por 2¹⁶ y luego redondear el número al entero más cercano. La frecuencia de corte (f_{corte}) real del filtro depende del ω seleccionado y de la frecuencia de muestreo (f_m) , como se muestra en la ecuación 3.6.

$$f_{corte} = \omega f_m / 2; \tag{3.6}$$

En la Figura 3.15 puede verse una foto del sistema de pruebas.

3.4.1. Resultados

En este apartado se describirán los resultados de la implementación del esquema de procesamiento de señales para la plataforma Cyclone V. Se describirá la ocupación de celdas lógicas del modelo, que dependen del elemento de control seleccionado (procesador embebido o programado en las celdas lógicas de la FPGA), se mostrarán algunas mediciones temporales para dar una idea de la máxima frecuencia de operación que puede alcanzarse con cada configuración y se comparará la función de transferencia de la señal filtrada con la transferencia teórica.

3.4.1.1. Recursos utilizados

El uso de recursos del sistema propuesto depende de la elección del elemento de control. Por un lado, seleccionar el microprocesador embebido supone un ahorro considerable de bloques de memoria, a expensas de una mayor cantidad de elementos lógicos. Por otro, si se selecciona el procesador NIOS, el sistema necesita una cantidad significativa de bloques de memoria adicionales, ya que su memoria está implementada directamente en ellos. Finalmente, para la demostración del filtro FIR, el sistema necesita una gran cantidad de bloques DSP, que se utilizan para implementar las multiplicaciones de manera eficiente. Sin embargo, estos podrían ser reemplazados por multiplicadores implementados con elementos lógicos, con una importante reducción de la eficiencia temporal y un aumento en el área. Esto podría ser una buena opción para dispositivos con menos multiplicadores integrados. El resumen del uso de recursos para cada modo de operación se muestra en el Cuadro 3.4. Es importante destacar que este puede variar entre compilaciones, y depende de la versión particular de compilador que se esté utilizando.

Recurso	Con μP	Con NIOS	Sistema completo*
Elementos lógicos (en ALMS)	3875~(12%)	2212 (7%)	6992~(22~%)
Registros	5937	3170	11282
Bloques de memoria	262272~(6%)	1473536~(36%)	1342592~(33%)
Bloques de RAM	30(8%)	195~(49%)	183~(46%)
Bloques DSP	0	0	67~(77%)
Pines	157~(34%)	85~(19%)	157~(34%)
PLLs	1~(6%)	$1 \ (6 \ \%)$	$1 \ (6 \ \%)$

Cuadro 3.4: Utilización de recursos

[†] Porcentajes calculados para dispositivo Cyclone V 5CSEMA5F31C6N *Sistema completo: μP + NIOS + filtro FIR de orden 32

3.4.1.2. Mediciones temporales

Para cada configuración del sistema, se calculó la frecuencia máxima alcanzable utilizando las herramientas proporcionadas por Intel-Altera. En todos los casos se logró la frecuencia máxima requerida para esta demostración (65 MHz). En los casos donde el filtro FIR no está implementado la frecuencia máxima está limitada por los pocos IPs de Altera utilizados en el diseño, mientras que en los casos donde el filtro FIR está implementado es este el que limita la frecuencia. Si se desea alcanzar una velocidad mayor, se deberían considerar otras arquitecturas de filtros. En el Cuadro 3.5 se muestran los resultados obtenidos.

Cuadro 3.5: Máxima frecuencia de operación

NIOS	μP	Filtro FIR	Máxima frecuencia
\checkmark	-	-	$115 \mathrm{~MHz}$
\checkmark	-	\checkmark	$82,31 \mathrm{~MHz}$
-	\checkmark	-	$97,82 \mathrm{MHz}$
-	\checkmark	\checkmark	$87 \mathrm{~MHz}$
\checkmark	\checkmark	\checkmark	$85 \mathrm{~MHz}$

3.4.1.3. Respuesta del filtro FIR

Con la configuración descripta anteriormente, se implementaron y probaron un filtro pasa bajo y un filtro pasa alto. Ambos fueron diseñados con una frecuencia de corte de $\omega = 0,05$, y se seleccionó una frecuencia de muestreo de 1 MHz. Con estos parámetros, se logra una frecuencia de corte de 25 kHz, como se espera de la ecuación 3.6. Los coeficientes seleccionados para los filtros se obtuvieron del *toolbox* de procesamiento de señales NumPy, de Python.

Con esta configuración, se midió la amplitud de las funciones de transferencia |H(f)| de ambos filtros con el SR865, y luego se compararon con las transferencias teóricas. Los resultados se muestran en la Figura 3.16 y en la Figura 3.17.



Figura 3.16: Transferencia del filtro pasa-bajos



Figura 3.17: Transferencia del filtro pasa-altos


Figura 3.18: Plataforma Red Pitaya

3.5. Adaptación del sistema a dispositivos SoC-FPGA de AMD-Xilinx

Aunque la implementación desarrollada en este capítulo está dirigida principalmente a plataformas SoC-FPGA de Intel-Altera, es importante destacar que los conceptos y la lógica subyacente pueden adaptarse a otras arquitecturas. La mayor parte de los módulos RTL escritos en Verilog son independientes de la plataforma de destino, lo que facilita su portabilidad entre diferentes sistemas.

Sin embargo, las IP propietarias específicas de Intel-Altera pueden requerir ajustes al migrar a otras plataformas. En esta sección, se describirá el proceso de adaptación de este sistema reconfigurable de procesamiento de señales a un SoC-FPGA de AMD-Xilinx, destacando las características y consideraciones clave que surgieron durante la integración del sistema.

Esta adaptación, disponible en [97], permite cubrir a los dos principales fabricantes de SoC-FPGA en la industria, ampliando así la aplicabilidad y la versatilidad del sistema desarrollado.

3.5.1. Red Pitaya

Para este diseño se decidió utilizar una plataforma Red Pitaya STEM^{LAB}125-14 [109]. Este es un dispositivo con una filosofía de hardware abierto que puede ser reconvertido en diversos dispositivos útiles como osciloscopios, analizadores de frecuencia y más, incluso sin la necesidad de conocimientos específicos sobre reconfiguración de hardware. Está basado en un SoC-FPGA Zynq 7010 de AMD-Xilinx y cuenta con dos entradas (AD9680) y dos salidas (AD9144) analógicas que pueden operar hasta 125 MHz con una resolución de 14 bits. Estas tienen la ventaja de estar acopladas en continua, por lo que no se necesitó de hardware adicional para utilizar su front-end analógico. En la Figura 3.18 se puede ver esta placa de desarrollo, con todos sus elementos de interés señalados.

3.5.2. Implementación

El esquema general de la implementación es el mismo que el desarrollado para la plataforma DE1-SoC (ver Figura 3.6). Los módulos que controlan el procesamiento se adaptan directamente a



Figura 3.19: Interfaz de módulo de procesamiento en Red Pitaya



Figura 3.20: Interfaz de módulo de señal en Red Pitaya

la nueva arquitectura, ya que consisten casi exclusivamente en lógica RTL. Para probar el sistema se replicó el filtro FIR desarrollado anteriormente, ahora con coeficientes constantes, configurados a través de **parameters** de Verilog, aunque éstos podrían hacerse configurables en tiempo de ejecución con relativa facilidad. Además se ha agregado un módulo de decimación, para poder controlar la cantidad de datos que entran a las memorias FIFO del módulo de control. La interfaz de entradas y salidas del módulo de procesamiento puede verse en la Figura 3.19.

Para el módulo de fuente de señal se aprovechó el IP de Xilinx dds-compiler [104], que genera señales sinusoidales utilizando el método de truncamiento de fase (ver sección 3.3.3.2), y se adaptaron los módulos que controlan los ADC y DAC para que funcionen con el nuevo *front-end* analógico. En esta implementación, el DAC simplemente genera señales sinusoidales que pueden utilizarse para alimentar los ADC del sistema, en vez de digitalizar la señal procesada, que solo queda disponible internamente en las memorias del elemento de control. La interfaz de entradas y salidas de este módulo puede verse en la Figura 3.20.

Las entradas marcadas en esta figura como phase_sen_interno y phase_dac son los incrementos de fase de la tabla de consulta. En esta implementación se seleccionó una $f_{clk} = 125$ MHz, para aprovechar el ancho de banda de los conversores, y un acumulador de fase de B = 28 bits, para mantener la resolución frecuencial de 0,5 Hz. Los valores de $\Delta\Theta$ para una onda sinusoidal de frecuencia f_c pueden calcularse con la ecuación 3.1.



Figura 3.21: Interfaz de módulo de control en Red Pitaya

El módulo de control instancia un IP ZYNQ7 Processing System [110], que simplifica en gran medida la comunicación de la FPGA con el procesador incorporado en el SoC-FPGA, un ARM Cortex-A9 de doble núcleo. Esta IP, junto con el AXI-Interconnect [111] y varias instancias de interfaces entrada/ salida, resueltos a través del IP AXI-GPIO [112] permiten una conexión sencilla entre los módulos de señal y procesamiento y el usuario. Para parametrizar el sistema se dejaron disponibles 10 parámetros de salida y 2 parámetros de entrada. Este módulo incorpora una memoria FIFO que permite guardar la señal procesada. Para esta implementación se diseño una HAL similar a la desarrollada para la Cyclone V, pero no se incorporó al sistema un procesador *soft*, aunque de necesitarse podría utilizarse el Micro-Blaze, análogo al NIOS para sistemas de AMD-Xilinx. Tampoco se ha desarrollado la GUI, aunque podría incorporarse en forma similar a lo desarrollado para la plataforma Cyclone V, ya que ambos elementos de procesamiento portan el mismo sistema operativo Linux. La interfaz de entradas y salidas de este módulo puede verse en la Figura 3.21.

Como el ADC opera a 125 MHz se requeriría mucha capacidad de memoria para analizar las formas de onda desde la etapa de control cuando las señales generadas están en el orden de los Hz o unos pocos kHz. Es por eso que se agregó al sistema un módulo decimador que en realidad tiene dos modos de funcionamiento: en uno de ellos decima la señal de un modo tradicional tomando 1 cada K muestras. En un segundo modo promedia K muestras, aprovechando esta capacidad de sobremuestreo para mejorar la resolución de la señal adquirida [113]. El modo de funcionamiento y el factor de sobremuestreo o decimación (K) son parámetros configurables del módulo. Su diagrama en bloques puede verse en la Figura 3.22.

3.5.3. Uso de recursos y temporización

El uso de recursos para el sistema completo se muestra en el Cuadro 3.6. El sistema cumple las condiciones de temporización, y la frecuencia máxima a la que se puede operar el reloj es de 135 MHz. Para lograr esta frecuencia de reloj debieron agregarse etapas de *pipeline* adicionales al cálculo del filtro FIR, en comparación a los calculados en la sección 3.4.1.2.

Del Cuadro 3.6 también es destacable que en la tecnología AMD-Xilinx la nomenclatura de los recursos utilizado varía ligeramente. A continuación se describe brevemente a que se refieren los valores del Cuadro. Para más información consultar [114].



Figura 3.22: Módulo de decimación en RP-li

- LUT: *Look-up-table* o tabla de consulta.
- SLICE: Cada slice está compuesto por cuatro tablas de consulta (LUTs) de seis entradas y ocho elementos de almacenamiento.
- Block-RAM Tile: Memoria Block-RAM. Cada mosaico (*tile*) de la memoria tiene 4 kBytes.
 Estos son *Dual-Port*, lo que permite leerlos y escribirlos en el mismo flanco de reloj.
- Multiplexores F7: Multiplexores de la serie 7 de Xilinx. Estos son utilizados para dirigir señales entre diferentes elementos lógicos dentro del chip.
- DSPs: Similarmente a los descriptos para Intel-Altera, son bloques que permiten realizar operaciones como multiplicaciones de manera eficiente.
- Bonded IOB: Bloques de Entrada/Salida (E/S) físicamente conectados dentro del dispositivo FPGA.
- Bonded IOPADs: Pines de E/S que están conectados o unidos a otros componentes dentro del chip. Estos se utilizan para conectar la FPGA con otros dispositivos dentro del sistema, como sensores, actuadores u otros dispositivos periféricos.
- IBUFDS: Input Buffer Differential Signaling o buffer de entrada para señales diferenciales.
- ILOGIC: Bloques de entrada de lógica programable.
- OLOGIC: Bloques de salida de lógica programable.
- BUFCTRL: Buffer de clock global.
- MMCME2_ADV: Bloque de gestión avanzada de relojes y multiplicador de frecuencia.

Recurso	Cantidad	Usado	Porcentaje
Slice LUTs	17600	3137	18%
Slice Registers	35200	4143	12%
Slice	4400	1418	32%
LUT as Logic	17600	3067	18%
LUT as Memory	6000	70	$1,\!16\%$
Block RAM Tile	60	2	3%
DSPs	80	44	55%
Bonded IOB	100	73	73%
Bonded IOPADs	130	130	100%
IBUFDS	96	3	3%
ILOGIC	100	14	14%
OLOGIC	100	18	18%
BUFGCTRL	32	4	12,5%
MMCME2_ADV	2	1	50%

Cuadro 3.6: Utilización de recursos del sistema en la plataforma Red Pitaya

3.5.4. Filtro pasa-bajos

Para verificar la operación del sistema, se diseñó un filtro FIR pasa-bajos con una frecuencia de corte normalizada de $\omega = 0.05$. Con una tasa de muestreo de $f_m = 125$ MHz, esto corresponde a una frecuencia de corte real de 3,125 MHz. Utilizando esta configuración, se muestrearon señales en el rango de 0,05 a 8 MHz y se estimó su amplitud utilizando $\frac{1}{2}(V_{\text{max}} - V_{\text{min}})$.

En este caso se consideró redundante la implementación de un filtro pasa-altos, principalmente debido a que el objetivo en esta etapa era verificar la correcta operación del sistema reconfigurable de procesamiento de señales en la plataforma Red Pitaya Cabe señalar que, en caso de requerirse otro tipo de filtrado, bastaría con modificar los **parameters** del módulo RTL que define el procesamiento de la señal.

Los resultados obtenidos se presentan en la Figura 3.23, donde se observa que el sistema opera de manera correcta, con valores que coinciden en gran medida con los resultados teóricos.

3.6. Conclusiones

En este capítulo se mostró el desarrollo de un sistema SoC-FPGA destinado al procesamiento de señales, proporcionando un diseño bien estructurado que los desarrolladores pueden seguir. El resultado es un sistema de código abierto, con opciones para adaptarlo a diferentes SoC-FPGA de Intel-Altera y AMD-Xilinx, y con conceptos básicos que son transferibles a las arquitecturas de otros fabricantes.

El diseño fue probado en dispositivos SoC-FPGA de ambos fabricantes. Se utilizó una plataforma DE1-SoC para Intel-Altera y un sistema Red Pitaya para AMD-Xilinx. Ambas configuraciones se evaluaron implementando un filtro FIR de orden 32 que opera en tiempo real con señales de alta frecuencia, un sistema con requisitos estrictos de velocidad y caudal de datos que resultan limitantes para la mayoría de los microprocesadores tradicionales.

Para el caso del DE1-SoC, la característica de transferencia del filtro se midió con un *Lock-in* SR865, verificando su correspondencia con la respuesta teórica esperada. En el caso de la Red Pita-



Figura 3.23: Función de transferencia del filtro pasa-bajos implementado en Red Pitaya

ya el sistema se validó generando una señal sinusoidal, para luego digitalizarla, filtrarla y guardarla internamente en la memoria del microcontrolador. Repitiendo esta operación para señales de distintas frecuencias se obtuvo la transferencia del filtro, que concuerda con la esperada teóricamente. En ambos casos se presentan los resultados obtenidos en cuanto al uso de recursos y las posibilidades de temporización, que pueden servir como referencia para desarrolladores interesados en utilizar este diseño.

En capítulos siguientes se utilizará este sistema de procesamiento de señales para implementar esquemas de *Lock-in* y promediación coherente, necesarios para las aplicaciones pretendidas como objetivo final de este trabajo de investigación.

Capítulo 4

Detección coherente en SoC-FPGA

4.1. Introducción

En el capítulo 2 se ha profundizado sobre las técnicas de *Lock-in* y promediación coherente de señales, y se ha visto como éstas pueden combinarse para obtener sistemas eficientes de detección coherente. Luego, en el capítulo 3, se ha diseñado un sistema de procesamiento de señales reutilizable, construido alrededor de un dispositivo SoC-FPGA Cyclone V desarrollado por Intel-Altera, y se ha estudiado como éste puede modificarse para ser implementado en dispositivos de AMD-Xilinx.

En este capítulo se describirá la implementación del sistema de detección coherente de señales utilizando las herramientas desarrolladas. Las especificaciones que debe cumplir el sistema están dadas por las aplicaciones en las que se inscribe este trabajo de tesis, descriptas en el capítulo 1. Éstas incluyen el desarrollo de un tomógrafo magnético, la implementación de interfaces cerebro computadora (BCI) basadas en potenciales evocados visuales de estado estable (SSVEPs) y la instrumentación de un equipo para medir señales de interés en el contexto de la Espectroscopía Mössbauer. Se buscó que el sistema diseñado pueda utilizarse en éstas y otras aplicaciones con cambios mínimos. A continuación, se resumen algunas de las características que se plantearon como objetivos en esta etapa del trabajo:

- Generación de las señales de excitación del sistema a través del conversor Digital/Analógico de alta velocidad (AD9767).
- Datos de entrada al procesamiento a través del conversor Analógico/Digital de alta velocidad (AD9248).
- Posibilidad de trabajar internamente con datos simulados, sobre todo para la etapa de prueba de los algoritmos.
- Implementación de *Lock-in* de dos fases, es decir con referencias en fase y cuadratura, para evitar la incertidumbre en la fase de la señal, como fuera descripto en la sección 2.2.2.
- Frecuencia de operación configurable desde unos pocos Hz hasta, al menos, 500 kHz. Si bien los cálculos efectuados en el capítulo 1 establecieron un mínimo de 100 kHz para detectar la señal proveniente del tomógrafo magnético sería deseable que el sistema pueda registrar frecuencias mayores. Se ha tomado media década por encima de este valor de 100 kHz como valor de compromiso.

- Posibilidad de realizar la medida utilizando Lock-in de N_{ma} ciclos de filtro de media móvil, o promediando coherentemente N_{ca} ciclos de la señal para luego someterla a un Lock-in de N_{ma} ciclos. Si bien se ha demostrado analíticamente (sección 2.4) la equivalencia entre ambos métodos, la implementación experimental permitirá sacar nuevas conclusiones al respecto.
- Capacidad de recuperar la señal promediada coherentemente N_{ca} ciclos. Esto tendrá aplicación directa en la instrumentación de equipos de Espectroscopía Mössbauer.
- Tiempo de integración de la señal configurable. Como se analizó en la sección 2.2.4, un tiempo mayor en el filtro pasa-bajos permite rechazar más ruido, pero un tiempo de integración elevado impide medir fenómenos cortos, por lo que se decidió que este sea un parámetro ajustable.
- Interfaz de usuario intuitiva y amigable, ejecutándose en el microcontrolador del SoC-FPGA, con posibilidad de transferir los datos obtenidos a una computadora anfitriona (*host*).
- Posibilidad de implementar más de un canal de detección coherente en las medidas, para maximizar los canales de detección en aplicaciones como la BCI. Esto implica realizar una cuidadosa evaluación de los recursos utilizados en la implementación, para maximizar los canales sin necesidad de migrar a dispositivos con mayores prestaciones.
- Portabilidad de los módulos desarrollados, para poder adaptarlos a SoC-FPGAs de AMD-Xilinx u otros fabricantes.

En las próximas secciones se describirán los módulos de procesamiento desarrollados, siguiendo la estructura propuesta en el capítulo 3. Primero se describirá la implementación en dispositivos de Intel-Altera, incluyendo el hardware, el software, la interfaz de usuario, y su aplicación en las medidas propuestas. También se presentan conclusiones adicionales sobre el uso de técnicas de *Lock-in* y promediación coherente de señales que complementan los resultados teóricos obtenidos previamente. A continuación, se describe la migración del sistema a dispositivos de AMD-Xilinx y se comparan las mediciones realizadas con ambos sistemas, así como con las obtenidas mediante un *Lock-in* comercial SR865. Cabe destacar que los sistemas desarrollados son de código y hardware abierto; están disponibles en [115] y [116], y pueden fabricarse a una fracción del costo de un *Lock-in* comercial como el SR865.

La mayor parte de los resultados de este capítulo fueron presentados en [53] y [54].

4.2. Implementación del sistema en DE1-SoC

4.2.1. Hardware

Para la implementación del sistema en dispositivos del Intel-Altera se utilizó un dispositivo DE1-SoC, con una placa de extensión *Highspeed AD/DA Card*, ambos descriptos en detalle en la sección 3.2. En la Figura 4.1 se muestra un diagrama en bloques del sistema implementado, donde pueden verse los controladores del DAC y del ADC, ya descriptos en la sección 3.3.3, el módulo de promediación coherente, el módulo de *Lock-in*, que se compone a su vez de un multiplicador y un filtro pasa-bajos, el módulo que controla el PLL para generar el reloj configurable y las memorias FIFO que guardan los resultados del procesamiento.

De este diagrama en bloques se han omitido por simplicidad las señales de enable, reset y finalización, que permiten controlar el flujo del proceso desde la interfaz de usuario, y la señal



Figura 4.1: Diagrama en bloques del sistema de detección coherente

digital simulada, que puede reemplazar a la acción del ADC y el DAC para probar el sistema. También por simplicidad se ha marcado la salida del *Lock-in* de dos fases como una única señal cuando deberían ser dos, una en fase (correspondiente a la multiplicación de la señal de interés por una referencia en fase) y una en cuadratura (correspondiente a la multiplicación de la señal por una referencia desfasada 90°) y se muestra una única memoria FIFO. A continuación, se describirá cada módulo en mayor detalle.

4.2.1.1. Promediación coherente

El módulo que controla la promediación coherente de la señal tiene una interfaz de entradas y salidas como la que se muestra en la Figura 4.2. La señal de entrada ingresa al módulo con su respectiva señal de data_valid y, luego de procesada, sale de él con la misma interfaz.

El punto de inicio de cada ciclo de la señal debe ser conocido para poder realizar correctamente la promediación. Para ello se incorpora la señal **sinc**, que se genera en el módulo de fuente de señal y se pone en alto en el flanco de reloj en que el acumulador de fase pasa por el 0, señalizando un nuevo ciclo (ver sección 3.3.3.2). La cantidad de ciclos de la señal que se deben promediar (N_{ca}) también es un parámetro que el módulo debe conocer. Todo el sistema es sincronizado por una señal de clock y señales auxiliares, **enable** y **reset_n**, permiten habilitar y reiniciar la operación. Para implementar este sistema en lógica puede seguirse un diagrama en bloques como el que se muestra en la Figura 4.3.



Figura 4.2: Interfaz del módulo de promediación coherente



Figura 4.3: Diagrama en bloques del sistema de promediación coherente

La señal de interés, cuantizada en Q_x bits, ingresa al sistema de promediación coherente, donde se suma con las últimas N_{ca} muestras que ocupan la misma posición relativa en el período. En el peor caso, se necesitarán $Q_1 = Q_x + \log_2(N_{ca})$ bits para almacenar cada muestra de la señal promediada. Una vez que se promedian los N_{ca} períodos de la señal esta queda a disposición de las siguientes etapas. Para implementar este sistema en hardware se necesita:

- Lógica para seleccionar la muestra de la memoria que debe ser añadida a la muestra entrante.
- Un sumador de $Q_1 = Q_x + \log_2(N_{ca}) = 14 + \log_2(N_{ca})$ bits.
- Memoria para almacenar la señal promediada de manera coherente: M muestras de Q_1 bits, donde M es la cantidad de puntos que entran en un periodo de la señal.

Es decir que se pueden definir tres operaciones independientes a ejecutarse, cada una con un tiempo de cómputo:

- Fetch: Registrar la muestra entrante $(X_{[i]})$ y la muestra promediada correspondiente de la memoria $(Y_{ca[i]})$. El tiempo que tarda esta operación se denotará con T_{fetch} .
- Sum: Registrar la suma de la muestra entrante y la promediada $(S_i = X_{[i]} + Y_{ca[i]})$. El tiempo que tarda esta operación se denotará con T_{sum_1} .

Clock Tarea	0	1	2	3	4
Fetch (T_{fetch}) Sum (T_{sum_1})	$x_{[0]} Y_{ca[0]}$ -	$\begin{aligned} x_{[1]} \\ Y_{ca[1]} \\ S_0 &= x_{[0]} + Y_{ca[0]} \end{aligned}$		 $S_2 = x_{[2]} + Y_{ca[2]}$	
Save (T_{save})	-	-	$Y_{ca[0]} = S_0$	$Y_{ca[1]} = S_1$	$Y_{ca[2]} = S_2$

Cuadro 4.1: Pipeline de promediación coherente

• Save: Guardar la nueva muestra promediada en la memoria $(Y_{ca} = S_i)$. El tiempo que tarda esta operación se denotará con T_{save} .

Si estas operaciones se ejecutan secuencialmente para cada muestra entrante (sin los registros indicados en la Fig. 4.3), el sistema no estará listo para recibir una nueva muestra hasta que todas las operaciones finalicen. Es decir que la máxima frecuencia a la que podría implementarse la operación estará limitada por $T_{fetch} + T_{sum_1} + T_{save}$.

Por otro lado, al incorporar los registros indicados en la Figura 4.3 es posible implementar estas tres tareas en un *pipeline*, como se muestra en el Cuadro 4.1. Con esta modificación la operación se vuelve más eficiente, ya que cada tarea no debe esperar a que la anterior se termine. De esta manera, la máxima frecuencia de operación del sistema pasa a estar limitada por la más lenta de las operaciones, que es razonable asumir que es el acceso a memoria o la suma. Es decir que la máxima frecuencia de operación quedará determinada por $\frac{1}{T_{save}}$ o $\frac{1}{T_{sum_1}}$. En la sección 4.3.1 se explorará en mayor detalle cual es la más lenta de estas operaciones.

4.2.1.2. Lock-in

En la Figura 4.4 puede verse la interfaz de entradas y salidas del módulo que calcula el Lock-in. La señal de entrada ingresa con su respectiva señal data_valid, y luego de procesada exporta dos señales, una correspondiente al cálculo en fase y otra al cálculo en cuadratura. La multiplicación en punto fijo hace que la cantidad de bits aumente, por lo que para mantener una resolución adecuada las interfaces de salida son de 64 bits. El módulo debe conocer el inicio de cada ciclo de la señal, dado por la señal sinc, para poder promediar un número entero de ciclos en el promedio móvil y garantizar la eliminación del componente del doble de frecuencia. Además debe conocer la cantidad de ciclos que debe promediar (N_{ma}) . Todo el sistema es sincronizado por una señal de reloj, y señales auxiliares, enable y reset_n, permiten habilitar y reiniciar la operación. La señal de finalización permite al módulo informar a la etapa de control que han finalizado los cálculos. Un sistema de Lock-in puede implementarse en lógica siguiendo un diagrama en bloques como el que se muestra en la Figura 4.5.

La señal de interés, cuantizada en Q_x bits, ingresa al *Lock-in*, donde es multiplicada por la señal de referencia, cuantizada en Q_{ref} bits. Esto produce una salida de Q_2 bits, que en el peor de los casos será igual a $Q_x + Q_{ref}$. Luego, la señal multiplicada ingresa a la etapa de promedio móvil, donde se suma con hasta $N_{ma}M$ muestras multiplicadas, donde M es la cantidad de muestras que



Figura 4.4: Interfaz del módulo del Lock-in

entran en un ciclo de señal. En el peor de los casos, cada muestra será igual a 2^{Q_2} , por lo que la suma de $N_{ma}M$ de estas muestras producirá una salida de $Q_3 = Q_2 + \log_2(N_{ma}M)$ bits.

Considerando que la señal es digitalizada con un ADC de 14 bits y considerando una señal de referencia cuantificada en 16 bits ($Q_x = 14$ y $Q_{ref} = 16$) la implementación del sistema *Lock-in* con filtro de media móvil requiere:

- Un multiplicador con una salida de $Q_2 = Q_x + Q_{ref} = 30$ bits. El tiempo que tarda esta operación se denotará con T_{mult_1} .
- Un sumador con una salida de $Q_2 = Q_1 + \log_2(N_{ma}M) = 30 + \log_2(N_{ma}M)$ bits. El tiempo que tarda esta operación se denotará con T_{sum_2} .
- Memoria para almacenar la referencia: M muestras de $Q_{ref} = 16$ bits.

En forma similar al caso anterior, si estas operaciones se ejecutan secuencialmente para cada muestra entrante (sin los registros indicados en la Fig. 4.5), el sistema no estará listo para recibir una nueva muestra hasta que la multiplicación y la suma estén finalizadas, lo que tomará



Figura 4.5: Diagrama en bloques del Lock-in

Clock Tarea	0	1	2	3	4
Reg. muestra (T_{reg}) Reg. producto (T_{mult_1}) Reg. acumulador (T_{sum_2})	x _[0] R _[0] -	$x_{[1]} \\ R_{[1]} \\ P_0 = x_{[0]} R_{[0]} $	$x_{[2]}$ $R_{[2]}$ $P_1 = x_{[1]}R_{[1]}$ $Y = Y + P_0$	\dots $P_2 = x_{[2]}R_{[2]}$ $Y = Y + P_1$	 $Y = Y + P_2$

Cuadro 4.2: Pipeline del Lock-in

aproximadamente $T_{mult_1} + T_{sum_2}$, más el tiempo para acceder a la memoria en búsqueda de la referencia. Por otro lado, al colocar registros intermedios, se pueden definir tres tareas independientes y consecutivas:

- Registrar la muestra entrante $(X_{[i]})$ y la referencia correspondiente $(R_{[i]})$. $[T_{reg}]$.
- Registrar el producto $(P_i = X_{[i]}R_{[i]})$. $[T_{mult_1}]$
- Registrar las sumas $(Y = Y + P_i)$. $[T_{sum_2}]$

En donde se ha colocado entre corchetes el símbolo que denota el tiempo de cada operación. Con estas tareas se puede definir un *pipeline* de tres etapas, en forma análoga a lo hecho para el caso de la promediación coherente, como se muestra en el Cuadro 4.2. De esta manera, la máxima frecuencia de operación del módulo está determinada por la tarea más lenta, que es razonable asumir que es la multiplicación, estableciendo la máxima frecuencia de operación del módulo en $\frac{1}{T_{mult_1}}$. Esta hipótesis se verificará en la sección 4.3.1.

4.2.1.3. Combinación de promediación coherente y Lock-in

Como se ha visto en la sección 2.4 la combinación de estos algoritmos de procesamiento permite reducir la cantidad de multiplicaciones involucradas en los cálculos, manteniendo intactos los resultados. En esta sección se mostrará como puede implementarse este sistema combinado en lógica, y cuales son las ventajas de hacerlo así. En la Figura 4.6 se muestra un diagrama en bloques del sistema combinado.

Este diagrama en bloques se puede subdividir en dos etapas: promediación coherente y *Lock-in*. Lo interesante de esta segmentación es que ambos sistemas pueden ejecutarse con relojes diferentes. Como es el CLK_1 el que registra las muestras entrantes, la operación del CLK_2 no resulta crítica para la operación del sistema. Esto implica que la frecuencia estará limitada por la suma del sistema de promediación coherente, y no por la multiplicación del *Lock-in*.

Por lo tanto, el sistema combinado traslada el cuello de botella de la implementación desde una multiplicación con una salida de $Q_x + Q_{ref} = 30$ bits a una suma de $Q_x + \log_2(N) = 14 + \log_2(N)$ bits, logrando una mejora en la frecuencia de reloj admisible, de $\frac{1}{T_{mult_1}}$ a $\frac{1}{T_{sum_1}}$ o $\frac{1}{T_{save}}$ (dependiendo de cual de estas tareas resulte ser la más lenta). Esta mejora será cuantificada en la sección 4.3.3.1.



Figura 4.6: Diagrama en bloques del sistema combinado

En secciones siguientes se retomará la nomenclatura adoptada en la sección 2.4.1.5: LI para referir al sistema de *Lock-in* con filtro de media móvil de N ciclos, y CALI para referir al sistema de promediación coherente de N ciclos seguido de *Lock-in* con filtro de media móvil de 1 ciclo. Es interesante notar que para que está equivalencia se cumpla el número de muestras en cada ciclo de sinusoide (M) debe ser constante, por lo que las señales deben generarse por el método de tabla de consulta directa y no por el de truncamiento de fase, como fuera predicho en la sección 3.3.3.2.

4.2.2. Interfaz de usuario

Controlar la operación del sistema de detección coherente implica configurar una serie de parámetros, iniciar la operación mediante la señal de **enable**, esperar a la señal de **finalización**, leer los resultados y finalmente reiniciar al sistema para que este preparado para la próxima operación.

Todas estas operaciones pueden ser controladas a través de la etapa de abstracción de hardware (HAL) descripta en la sección 3.3.2. Los parámetros a configurar están resumidos en el Cuadro 4.3, junto con una descripción de su efecto y al número de parámetro, correspondiente al offset que requiere la función de la HAL.

Parámetro	Offset	Descripción
Fuente de procesamiento	0	0: ADC LTC2308 1: ADC HS 2: Simulación
$N_{ma_{CALI}}$	2	Ciclos de media móvil en algoritmo CALI †
$N_{ca_{CALI}}$	3	Ciclos de promediación coherente en algoritmo CALI †
Bits ruido de simulación	4	Escala el ruido en la señal simulada
Selección algoritmo	5	0: CALI 1: LI
N_{LI}	6	Ciclos de media móvil en algoritmo LI
Delta fase	7	Controla la frecuencia de operación (ver sección 3.3.3.2)
LED Test	10	Controla el estado de un LED para verificar operación

Cuadro 4.3: Parámetros modificables del sistema de detección coherente

[†] Si bien el algoritmo CALI como fue definido supone $N_{ma} = 1$ y $N_{ca} = N$ se dejó como parámetro reconfigurable para probar distintos escenarios en la práctica.

4.2.2.1. Programas base para el sistema de detección coherente

Sobre esta HAL se desarrollaron algunos programas en C++ que controlan la operación del sistema de detección coherente y pueden ejecutarse a través de la linea de comandos del sistema Linux, que a su vez puede manejarse desde la computadora *host* a través de una conexión ssh. También pueden utilizarse estos programas base desde rutinas en Python, haciendo uso de llamadas disponibles en paquetes como subprocess.

► toggle_led led_state

Este programa puede utilizarse para verificar la comunicación entre el microcontrolador y la FPGA. Su funcionamiento es sencillo y directo, permitiendo encender o apagar un LED en función del parámetro led_state. Es una herramienta útil para pruebas iniciales de conectividad y diagnóstico.

- measure_lockin sim_ruido N frecuencia fuente modo nombre_archivo [f_clk] Este programa calcula la respuesta del Lock-in en una frecuencia específica y guarda los resultados en nombre_archivo.
- adquirir2 N frecuencia ciclos2display nombre_archivo [sim_ruido f_clk fuente] Este programa genera un archivo de texto nombre_archivo con ciclos2display ciclos de las señales guardadas en los FIFOs 1 y 2.

- barrido_en_f f_clk N fuente modo f_inicial f_final f_step nombre_archivo sim_ruido Este programa calcula la respuesta del Lock-in en un rango de frecuencias especificado por los parámetros f_inicial, f_final y f_step, guardando los resultados en nombre_archivo. Esta opción es particularmente útil para medir funciones de transferencia de un circuito, por ejemplo.
- ▶ barrido_cte_tiempo frecuencia N_inicial N_final iteraciones fuente sim_ruido nombre_archivo Este programa calcula el Lock-in en la frecuencia seleccionada, iteraciones veces para cada valor de N comprendido entre N_inicial y N_final. Luego calcula la media y desviación estándar de las mediciones en cada N, guardando los datos en nombre_archivo.

Para ejecutar estos programas correctamente, es necesario proporcionar los parámetros especificados en cada uno. Los opcionales, indicados entre corchetes, pueden omitirse. A continuación, se resumen los parámetros comunes a todas las funciones desarrolladas.

- sim_ruido: Nivel de ruido en la señal simulada.
- N: Número de ciclos de promediación coherente para CALI o de promedio móvil para LI.
- frecuencia: Frecuencia de la señal generada en el DAC y de las señales de referencia.
- fuente: Origen de la señal (0 para adc_2308, 1 para adc_hs, 2 para simulación).
- modo: Modo de operación (0 para CALI con media móvil de 1 ciclo, 1 para LI).
- f_clk: Frecuencia del reloj principal.

4.2.2.2. Servidor web

Para que el usuario pueda controlar la operación del sistema de detección coherente desde una computadora *host*, se diseñó una interfaz web utilizando HTML, CSS y JavaScript. Esta permite ejecutar los programas base según los parámetros ingresados por el usuario. Para ello, se implementó un servidor Apache para recibir las consultas HTML provenientes de la computadora *host*, junto con un entorno de ejecución Node.js del lado del servidor para procesar estas consultas y controlar los programas base.

De esta manera un usuario sin conocimientos específicos de hardware reconfigurable ni sistemas embebidos puede utilizar la herramienta para efectuar mediciones y transferirlas fácilmente a la computadora *host*. En la Figura 4.8 se muestran capturas de pantalla de esta interfaz.

Fuente:	Frecuencia [Hz]:
Simulación interna 🖌	100000
Ruido [bits]:	N:
10	1
Tipo de procesamiento:	Frecuencia del reloj [MHz]:
LI 👻	64
Corregir fase?	
NO 🗸	
	R : 1.0272
Iniciar	PHI: 0.0848105

Lockin en DE1-SoC

(a) Ventana: Cálculo de Lock-in

Señales involucradas

(b) Ventana: Visualizador de señales

Barrido en frecuencia

Erocuoncia final [Hz]:	NO 🗸
Frequencia final [Hz]:	
riecuencia final [riz].	Archivo de salida:
000000	barrido.txt
	Paso [Hz]:
	10000

(c) Ventana: Barrido en frecuencia

Iniciar barrido







(b) Ventana: Medidas de ruido

Figura 4.8: Interfaz de usuario del sistema de detección coherente en DE1-SoC. Ejemplo para una señal simulada, cuantizada en 14 bits con 10 bits de amplitud de ruido.

4.3. Resultados en De1-SoC

En esta sección se resumen algunos de los resultados obtenidos al trabajar en la plataforma DE1-SoC. Primero, se evalúan los algoritmos por separado, cuantificando sus tiempos de cómputo y el uso de recursos. Luego, se implementa el sistema de detección coherente completo y se evalúa su desempeño. Para ello, se determina la máxima frecuencia de operación del sistema y la ocupación de celdas lógicas en la FPGA. Posteriormente, se realizan mediciones de ruido en distintas condiciones de operación y se caracterizan algunas funciones de transferencia conocidas, comparándolas con medidas obtenidas con un *Lock-in* comercial.

4.3.1. Tiempo individual de operaciones

Para cuantificar las suposiciones realizadas en la sección 4.2.1 respecto al tiempo de ejecución de cada operación, se implementaron y probaron algunos sistemas simples.

- Para el sistema de promediación coherente:
 - Sumador con una salida de $Q_x + \log_2(N_{ca})$ bits = $14 + \log_2(N_{ca})$ bits (T_{sum_1}) .
 - Lógica para guardar una muestra en memoria (T_{save}) .
 - Lógica para recuperar una muestra desde memoria (T_{fetch}) .
- Para el sistema de *Lock-in*:
 - Multiplicador con salida de $Q_x + Q_{ref} = 30$ bits (T_{mult_1}) .
 - Sumador con salida de $Q_x + Q_{ref} + \log_2(NM)$ bits = $30 + \log_2(NM)$ (T_{sum_2}) .

Es interesante notar que, en el caso del sistema combinado (promediación coherente seguida de *Lock-in*), el cuello de botella se encuentra en la etapa de promediación coherente, ya que es la encargada de registrar las muestras entrantes y, en definitiva, procesar la señal en tiempo real.

Los parámetros M y N varían según los requisitos específicos y las características de la aplicación. Para esta estimación, se asumirá que $N \leq 8192$ y $M \leq 128$. Los valores de Q_{ref} y Q_x utilizados son los propuestos en la Sección 4.2.1, 16 y 14 bits, respectivamente. Con estos valores, T_{sum_2} se convierte en una suma con una salida de 50 bits y T_{sum_1} en una suma con una salida de 27 bits.

Para cada operación, se calculó la frecuencia máxima utilizando las herramientas proporcionadas por el software Quartus II de Intel-Altera. Para las multiplicaciones se consideraron dos opciones: usar bloques DSP, que ofrecen mejor rendimiento pero son un recurso limitado, o implementarlas con elementos lógicos. El Cuadro 4.4 resume los resultados.

Este análisis confirma algunas de las suposiciones realizadas en la Sección 4.2.1. El cuello de botella del sistema *Lock-in* es, en efecto, la multiplicación (141,3 MHz), siempre y cuando no se utilicen bloques DSP. Sin embargo, si se utilizan, la suma de 50 bits limita la operación del sistema (215,24 MHz).

Por otro lado, el sistema de promediación coherente (y por lo mencionado anteriormente del sistema combinado) tiene su rendimiento limitado por el acceso a la memoria, en particular el tiempo para recuperar una muestra de memoria (279,25 MHz). En ambos casos (con o sin bloques DSP), este enfoque es temporalmente más eficiente que el *Lock-in* tradicional con filtro de media móvil. Los

valores temporales fueron obtenidos bajo condiciones ideales; sin embargo, en la implementación de un sistema *Lock-in* completo, los tiempos reales se aproximarán a los estimados.

4.3.2. Uso de recursos de cada algoritmo

El uso de recursos de cada algoritmo bajo diferentes circunstancias se muestra en el Cuadro 4.5. Basándose en este cuadro, es evidente que el algoritmo CALI requiere más recursos, especialmente en cuanto a la utilización de memoria, ya que necesita espacio para guardar la señal promediada coherentemente. Sin embargo, es más eficiente en términos de utilización de bloques DSP, ya que los utilizados para el sistema CALI se emplean en el *Lock-in* de un ciclo, lo que no tiene impacto en el rendimiento del sistema, por lo que podrían eliminarse de forma segura.

Como se mencionó anteriormente, estos bloques son un recurso limitado, y el Cuadro 4.5 muestra que un solo Lock-in implementado con el algoritmo tradicional consume aproximadamente el 5% de ellos en este dispositivo. Esto puede ser un problema, por ejemplo, si se desea implementar un Lock-in multifrecuencial o si otras partes del sistema los necesitan. Por otro lado, al utilizar el enfoque CALI, el límite para el número de frecuencias que se pueden detectar está dado por la memoria y los elementos lógicos, que son más un recurso menos escaso, y los bloques DSP pueden ser reservados para otras partes del sistema.

4.3.3. Implementación del sistema de detección coherente

4.3.3.1. Consideraciones temporales

Los sistemas de *Lock-in* (LI) y promediación coherente seguido de *Lock-in* (CALI) fueron implementados en el dispositivo DE1-SoC, siguiendo las pautas de diseño proporcionadas en la sección 4.2.1. En el Cuadro 4.6 se muestra, para cada algoritmo, la frecuencia máxima alcanzable con y sin el uso de bloques DSP.

Estos resultados son coherentes con las suposiciones hechas en secciones anteriores. Si bien el reloj del sistema CALI (CLK_2) es el más restrictivo, no trabaja con la señal en tiempo real; en su lugar, se utiliza para realizar el *Lock-in* de un ciclo en la señal promediada. Por lo tanto, el sistema CALI supera al sistema LI, incluso sin la necesidad de bloques DSP.

De esta manera, el algoritmo CALI aumenta la frecuencia máxima de operación desde 159 MHz (utilizando bloques DSP) o 119 MHz (sin bloques DSP) hasta 216 MHz, lo que representa una mejora considerable.

Operación	DSPs	Frecuencia Máx. [MHz]	Símbolo Relacionado
Multiplicación [30 bits]	\checkmark	310,08	T_{mult_1}
Multiplicación [30 bits]	Х	141,3	T_{mult_1}
Suma $[50 \text{ bits}]$	*	$215,\!24$	T_{sum_1}
Suma [27 bits]	*	320,92	T_{sum_2}
Guardar Muestra	*	$315,\!06$	T_{save}
Recuperar Muestra	*	$279,\!25$	T_{fetch}

Cuadro 4.4: Tiempo por operación

*Mismo resultado con o sin bloques DSP

Algo- ritmo	DSPs	M	ALMS	ALUTs	Registros	Bits de memoria	M10K	Bloques DSP
			(32070)	(64140)	(64140)	(4065280)	(397)	(87)
LI	\checkmark	32	$111,8\ (0,3\%)$	$222 \ (0,3\%)$	$256\ (0,3\%)$	0	0	$4 \\ (4,5\%)$
LI	Х	32	$357,4\ (1,1\%)$	$680 \\ (1\%)$	${360 \atop (0,5\%)}$	0	0	0
LI	\checkmark	128	$137,2 \ (0,4\ \%)$	$268 \ (0,4\%)$	$259\ (0,4\%)$	0	0	$4 \\ (4,5\%)$
LI	Х	128	$382,9\ (1,1\%)$	$726 \ (1,1\%)$	$336 \ (0,5\%)$	0	0	0
CALI	\checkmark	32	$229,7\ (0,7\%)$	$387 \ (0,6\%)$	$580\ (0,9\%)$	$1728 \\ (0,04\%)$	$2 \\ (0,5\%)$	${6} \ (6,8\%)^{\dagger}$
CALI	Х	32	$940,2 \\ (2,9\%)$	$1604 \\ (2,5\%)$	$686 \ (1\%)$	$1728 \\ (0,04\%)$	$2 \\ (0,5\%)$	0
CALI	\checkmark	128	$263,9\ (0,8\%)$	$437 \\ (0,\!6\%)$	$586\ (0,9\%)$	$\begin{array}{c} 6912 \\ (0,\!17\%) \end{array}$	$2 \\ (0,5\%)$	${6} \ (6,8\%)^{\dagger}$
CALI	X	128	$963,5\ (3\%)$	$1654 \ (2,5\%)$	$665 \ (1\%)$	$6912 \ (0,17\%)$	$2 \\ (0,5\%)$	0

Cuadro 4.5: Utilización de recursos para cada algoritmo

*Valores obtenidos con ${\cal N}=8192$
 [†]Pueden ser eliminados sin efecto en el rendimiento

Cuadro 4.6: Tiempo por algoritmo

Algoritmo	DSPs	Frecuencia Máx. [MHz]
LI	\checkmark	159,21
LI	Х	119,39
CALI $[CLK_1]$	*	$216{,}54$
CALI $[CLK_2]$	\checkmark	143,33
CALI $[CLK_2]$	Х	$90,\!32$

*Mismo resultado con o sin bloques DSP

4.3.3.2. Uso de recursos del sistema completo

El sistema de detección coherente, incluyendo la interfaz de usuario, memorias para almacenar las señales, y la posibilidad de realizar la operación con algoritmo CALI y LI ocupan los recursos mostrados en el Cuadro 4.7. Estos recursos son mucho mayores que los reportados en el Cuadro 4.5, debido a que en este se muestran los recursos utilizados solo por cada algoritmo, sin tener en cuenta al resto del sistema. Estos recursos podrían reducirse, de ser necesario, restringiendo las operaciones que el sistema realiza.

Recurso	Usado	Total disponible	Porcentaje utilizado
Elementos lógicos (en ALMS)	10,391	32,070	32%
Registros	9,106	64,140	14%
Bloques DSP	16	87	18%
Bits de memoria	$2,\!652,\!160$	4,065,280	65~%
M10K	327	397	82%
PLLs	1	6	16~%

Cuadro 4.7: Utilización de recursos

4.3.3.3. Mediciones de Ruido

Para medir los efectos del sistema sobre el ruido, se procesaron algunas señales al sistema CALI operando directamente en la FPGA. En la primera prueba, la señal fue una sinusoide puramente digital de 100 kHz, cuantizada a 14 bits, con una perturbación de ruido aditiva generada mediante el algoritmo LFSR (registro de desplazamiento con retroalimentación lineal, ver sección 3.3.3.3). En una segunda prueba, se generó una sinusoide de 100 kHz, también cuantizada a 14 bits, utilizando el DAC_HS y luego se digitalizó con el ADC_HS. En ambos casos, las pruebas se realizaron para valores crecientes de N, y para cada N, se tomó la desviación estándar de la amplitud a partir de 100 iteraciones. Luego se normalizaron los valores, dividiéndolos por la máxima desviación estándar obtenida. Estos resultados se muestran en la Figura 4.9, junto con los valores teóricos esperados de la ecuación 2.34. Para valores altos del tiempo de integración se empieza a ver una reducción de ruido menor que la predicha teóricamente, efecto que puede atribuirse al ruido de cuantización, como fuera predicho en la sección 2.2.6.

Para el tiempo de integración más largo de los ensayados en esta sección (N = 512) la desviación estándar de las mediciones de tensión para la señal generada por el DAC y muestreada con el ADC es de 7,67 μ V, frente a una señal de 918 mV de amplitud. Es decir que el sistema puede detectar variaciones de tensión mayores a $8,35 \times 10^{-6}$. Tiempos de integración mayores pueden ayudar a reducir el ruido en las medidas pero no en niveles muy significativos, debido al limite dado por la cuantización de las señales.

4.3.3.4. Medición de transferencias conocidas

Para probar el sistema en condiciones reales, se midió la función de transferencia de distintos circuitos utilizando simultáneamente el algoritmo LI y el algoritmo CALI, y luego se compararon los resultados obtenidos con un *Lock-in* comercial SR865, en una configuración como la que se muestra en la Figura 4.10. Para estas mediciones se utilizó el *Highspeed AD/DA card*. Con la placa de acoplamiento en DC descripta en la sección 3.2.2, la interfaz analógica del DE-li comienza a afectar la transferencia en frecuencias superiores a los 500 kHz, motivo por el cual se realizaron pruebas hasta dicho límite. Este rango de frecuencias resulta adecuado para las aplicaciones abordadas en



Figura 4.9: Medición de ruido con *Lock-in* en DE1-SoC a medida que aumenta el tiempo de integración

esta tesis. En caso de requerirse una extensión hacia frecuencias mayores, sería necesario rediseñar la etapa analógica, como se detalla en la sección correspondiente; sin embargo, esta mejora escapa al alcance de los objetivos planteados. En todos los casos las tolerancias de los componentes pasivos utilizados fue del 1%.

Carga pasiva diferencial RC $f_c = 7,95 \, \text{kHz}$

Primero se probó el sistema con un circuito RC diferencial con $R = 10 k\Omega \text{ y C} = 1 \text{ nF}$ como el que se muestra en la Figura 4.10. Ambos sistemas (CALI y LI) se configuraron para funcionar con N = 512 ciclos de señal. La función de transferencia obtenida para el sistema con los distintos métodos, junto con la esperada de la teoría puede verse en la Figura 4.11, sobre la que pueden sacarse algunas conclusiones:

Con el procesamiento LI y el procesamiento CALI se obtienen exactamente los mismos resultados, como se predijo en la sección 2.4.

La frecuencia de corte a -3 dB del filtro formado por este circuito RC está dada teóricamente por la ecuación 4.1.

$$fc = \frac{1}{2\pi \text{R2C}} \approx 7,95 \,\text{kHz}$$
 (4.1)

Examinando la función de transferencia, la frecuencia de corte puede determinarse como 7,956 kHz con el SR865 y 7,972 kHz con el procesamiento de FPGA, un error de aproximadamente 0.2%



Figura 4.10: Circuito de pruebas



Figura 4.11: Función de transferencia de una carga RC medida con diferentes métodos



Figura 4.12: Función de transferencia de una carga RC medida con diferentes métodos

considerando la medida por el SR865 como el valor real de esta frecuencia de corte.

En el Cuadro 4.8 se muestran los valores medios, mínimos y máximos del error cometido al estimar la magnitud y la fase del sistema con el procesamiento en DE1-SoC, considerando como valor verdadero de esta función de transferencia tanto los valores teóricos como los obtenidos con el SR865. Tanto los valores de errores de fase como de magnitud se mantienen bastante por debajo del 1 % para este rango de frecuencias.

Carga pasiva diferencial RC $f_c = 117 \, \text{kHz}$

El segundo circuito ensayado es idéntico al de la Figura 4.10 pero con $R = 680 \Omega$ y C = 1 nF. La función de transferencia obtenida para el sistema puede verse en la Figura 4.12. Los resultados con algoritmo CALI y LI son idénticos como en la figura anterior, por lo que se han omitido en esta ocasión y se resumen dentro de «Datos medidos (DE1-SoC)».

La frecuencia de corte de un filtro como éste tiene un valor teórico de 117 kHz. Con el SR865 esta frecuencia de corte se puede estimar en 106 kHz, mientras que con el procesamiento en DE1-SoC se puede estimar en 115 kHz. Esto supone un error del 8 % tomando al valor del SR como verdadero, o de 1,7 % considerando la diferencia con el valor teórico.

En el Cuadro 4.8 se muestran los valores medios, mínimos y máximos del error cometido en la estimación de la magnitud y fase de la transferencia con el procesamiento en DE1-SoC, considerando como valor verdadero tanto los valores teóricos como los obtenidos con el SR865. Para este rango de frecuencias los errores cometidos frente al procesamiento con el SR865 tienen valores un poco más elevados, cerca del 2 %, aunque los valores de errores frente a la transferencia teórica se mantienen

Experimento	Diferencias entre teoría y DE1-SoC	Diferencias entre SR865 y DE1-SoC
$f_c = 7,95 \text{ kHz}$ 0 < f < 125 kHz	Amplitud: Media: 0,0028 (0,2759%) Mínima: 0,000066 (0,0066%) Máxima: 0,0072 (0,7163%)	Amplitud: Media: 0,0009 (0,0857 %) Mínima: 0,0000 (0,0000 %) Máxima: 0,0038 (0,3841 %) 0 <th< td=""></th<>
	Fase: Media: 0, 41° (0, 1130 %) Mínima: 0,0006° (0,0002 %) Máxima: 1,0191° (0,2831 %)	Fase: Media: 0,53° (0,1481%) Mínima: 0,0002° (0,0001%) Máxima: 1,2117° (0,3366%)
$\begin{aligned} f_c &= 117 \text{ kHz} \\ 0 &< f < 500 \text{ kHz} \end{aligned}$	Amplitud: Media: 0,0026 (0,2635%) Mínima: 0,000008 (0,0008%) Máxima: 0,0071 (0,7099%)	Amplitud: Media: 0,0209 (2,0939%) Mínima: 0,000003 (0,0003%) Máxima: 0,0304 (3,0423%)
	Fase: Media: 1, 43° (0, 3976 %) Mínima: 0, 0013° (0, 0004 %) Máxima: 2, 66° (0, 7381 %)	Fase: Media: 4,87° (1,3514%) Mínima: 0,0027° (0,0008%) Máxima: 7,59° (2,1080%)

Cuadro 4.8: Resumen de diferencias de medidas entre teoría, procesamiento en dispositivo DE1-SoC (DE-li) y SR865 para experimentos con carga RC diferencial

por debajo del 1%.

Para ambos rangos de frecuencia, los errores en la determinación de la transferencia son significativamente menores al 1% en comparación con los valores teóricos esperados. En el rango de frecuencias más bajas, los errores, al contrastarlos con el *Lock-in* comercial, también se mantienen por debajo del 1%, mientras que en el rango de frecuencias más altas son algo mayores. Esto podría atribuirse a las diferencias en las etapas analógicas de los dispositivos. En las frecuencias de interés para la aplicación más exigente de esta tesis (en torno a 100 kHz), los errores se encuentran significativamente por debajo del 1%, por lo que se puede concluir que el sistema desarrollado parece ser adecuado para el propósito de esta investigación

4.4. Implementación del sistema en Red Pitaya

Aunque el principal objetivo de esta tesis es el desarrollo de las aplicaciones propuestas en la introducción, la disponibilidad de un sistema Red Pitaya y el avance logrado en la simplificación de esquemas de procesamiento de señales en este tipo de dispositivos ha generado un interés adicional en el desarrollo de sistemas de detección coherente de señales dentro de esta nueva arquitectura. Por esta razón, en esta etapa del trabajo se dedicaron esfuerzos a implementar estos sistemas en la plataforma Red Pitaya.

Para migrar el sistema de *Lock-in* a esta plataforma se partió del esquema reconfigurable de procesamiento de señales (ver sección 3.5). El diagrama general es similar al de la plataforma DE1-SoC (ver figura 4.1), y el código RTL que comanda el *Lock-in* puede reutilizarse directamente. Por simplicidad se ha implementado el sistema LI, pero si se quisiera mejorar su eficiencia temporal

se podría implementar el sistema CALI, siguiendo el mismo esquema desarrollado previamente. El código generado es abierto y esta disponible en [116]. En la Figura 4.13 se presenta un diagrama en bloques del sistema, generado por la herramienta Vivado. A continuación, se describe brevemente, destacando las diferencias entre este y el *Lock-in* implementado en la DE1-SoC. Para diferenciar ambos sistemas, se utilizarán en adelante las siglas DE-li y RP-li.

4.4.1. Fuente de datos y conversor digital analógico

Este módulo puede verse en detalle en la Figura 4.14, y engloba los controladores del ADC, la fuente de referencias y la señal simulada. Una diferencia importante entre el DE-li y el RP-li es que este último ya tiene los ADC acoplados en continua, por lo que no fue necesaria una etapa de acondicionamiento de señal como la descripta en la sección 3.2.2, y que estos conversores son *single-ended* en lugar de diferenciales. En esta configuración el ADC opera siempre en su máxima frecuencia de reloj de 125 MHz.

Para generar las señales de referencia y de las señales digitales para simulación se ha utilizado el IP dds-compiler, que genera formas de onda sinusoidales mediante el método de truncamiento de fase (ver sección 3.3.3.2). En este sistema no se ha agregado una fuente aleatoria para simular el ruido, pero podría hacerse con relativa facilidad adaptando los módulos RTL correspondientes del DE-li. Un multiplexor controlado por la etapa de control permite seleccionar a la señal proveniente del ADC o a la señal simulada como salidas de este módulo, que son a su vez las entradas del módulo *Lock-in*. En simultaneo con la señal de referencia se genera la señal de **start**, cada vez que esta pasa por 0 (equivalente a la señal **sinc** de DE-li). De esta forma se le señaliza al *Lock-in* el inicio de cada ciclo, manteniendo la sincronización.

Para generar las señales de excitación del *Lock-in* se aprovecho el DAC integrado en la plataforma Red Pitaya. Al igual que en el caso del ADC, los DAC de la plataforma Red Pitaya ya están acoplados en continua y operan a 125 MHz, por lo que no se necesito de hardware adicional para su incorporación al sistema. Como el ADC opera a 125 MHz se requeriría mucha capacidad de memoria para analizar las formas de onda desde la etapa de control cuando las señales generadas están en el orden de los Hz o unos pocos kHz. Es por eso que se agregó al sistema un módulo decimador, idéntico al descripto en la sección 3.5.2.



Figura 4.13: Diagrama en bloques del sistema Lock-in en Red Pitaya (RP-li)



Figura 4.14: Módulo de fuente de datos en RP-li



signal processing LI v1 0

Figura 4.15: Módulo de Lock-in en RP-li

4.4.2. Lock-in

El RTL del Lock-in ya desarrollado se adapta directamente a la nueva arquitectura. La Figura 4.15 muestra su interfaz de entradas y salidas, donde se destacan las entradas de señal, referencia y sincronización (start), que provienen del módulo de fuente de señal, así como las señales de reset y enable, que provienen de la etapa de control. Además, se incluye la cantidad de ciclos a promediar en el filtro MAF (N_{ma}), que figura como parameter_in_1, y la cantidad esperada de puntos por ciclo de señal (M), representada con parameter_in_0. Las salidas en fase y cuadratura de 64 bits se exportan hacia la etapa de control, junto con la señal de finalización, y la bandera que avisa que el módulo esta listo para procesar, siguiendo los lineamientos de la sección 3.5. Una señal auxiliar datos_promediados lleva la cuenta de la cantidad de muestras que se han promediado en el filtro MAF, para ajustar variaciones de la cantidad de puntos por ciclo que se generan en el módulo de fuente de señal, como consecuencia del método de generación por truncamiento de fase.

4.4.3. Control del sistema

El controlador del sistema funciona desde el microprocesador de la Red Pitaya, que porta un sistema operativo Linux. Su operación es muy similar a la descripta en secciones anteriores (ver secciones 3.5 y 4.2.2) por lo que no se profundizará aquí sobre este tema. Los parámetros que debe controlar el sistema RP-li son:

- N_{ma} : Ciclos de promediación en el filtro MA.
- frec_dac: Frecuencia de las señales de excitación.
- frec_ref: Frecuencia de las señales de referencia.
- sel: Selección de fuente de los datos (0: simulación, 1: ADC).
- decimator: Valor de decimación K.
- decimator_method: Método de decimación (0: descartar muestras, 1. promedio lineal).

Programas escritos en C++ controlan la operación en el Linux de la Red Pitaya. Estos pueden ser accedidos usando conexión ssh desde una computadora *host*, que a su vez pueden ser utilizados desde programas en Python, simplificando la operación del RP-li, en forma análoga a lo descripto para el DE-li. En etapas futuras de este desarrollo se contempla añadir al sistema una interfaz de usuario tipo Servidor Web, como la descripta para el DE-li en la sección 4.2.2.2.

4.5. Resultados en Red Pitaya

4.5.1. Uso de recursos y temporización

El sistema RP-li se implementó en la plataforma Red Pitaya, cumpliendo con las especificaciones temporales dadas por sus entradas y salidas analógicas (125 MHz). La máxima frecuencia a la que puede operar el reloj es de 157 MHz y el uso de recursos se muestra en el Cuadro 4.9, donde se ve que hay recursos de sobra para implementar más canales de detección *Lock-in* u otros elementos de interés.

4.5.2. Mediciones de ruido

Para medir los efectos del sistema sobre el ruido se replicaron las medidas efectuadas en la sección 4.3.3.3, pero ahora para el sistema RP-li. En esta ocasión se generó una señal sinusoidal de $f_s = 1$ MHz dentro de la FPGA, se exteriorizó mediante el DAC y luego se midió con el ADC. Esta señal fue sometida a procesamiento *Lock-in* con filtro MA, para valores crecientes de N, y para cada N se tomó la desviación estándar de la amplitud a partir de 100 iteraciones. En la Figura 4.16 se muestran los resultados de estas medidas, junto con los valores teóricos esperados de la ecuación 2.34. Es interesante notar que para valores bajos y medios de N las medidas se asemejan mucho a lo predicho por esta ecuación pero, para valores de N más elevados la ecuación predice una reducción de ruido mayor a la lograda en la práctica. Esto es porque los efectos del ruido de cuantización imponen un límite a la reducción de ruido, como fuera predicho en la sección 2.2.6.

Para el tiempo de integración más largo de los ensayados en esta sección (N = 4096) la desviación estándar de las mediciones de tensión para la señal generada por el DAC y muestreada con el ADC es de 11, 1µV, frente a una señal de 965 mV de amplitud. Es decir que el sistema puede detectar variaciones de tensión mayores a 11,5 × 10⁻⁶.

Recurso	Cantidad	Usado	Porcentaje
Slice LUTs	17600	3832	21,7%
Slice Registers	35200	7316	$20,\!78\%$
Slice	4400	2187	49,7%
LUT as Logic	17600	3746	21,3%
LUT as Memory	6000	86	1,4%
Block RAM Tile	60	3.5	5.8%
DSPs	80	2	2.5%
Bonded IOB	100	73	73%
Bonded IOPADs	130	130	100%
IBUFDS	96	3	3.1%
ILOGIC	100	14	14%
OLOGIC	100	18	18%
BUFGCTRL	32	4	12,5%
MMCME2_ADC	2	1	50%

Cuadro 4.9: Utilización de recursos para el RP-li

4.5.3. Medición de transferencias conocidas

En forma similar a lo desarrollado en la sección 4.3.3.4, se probó el sistema RP-li midiendo algunas funciones de transferencia conocidas. El circuito utilizado para estas mediciones es similar al mostrado en la Figura 4.10, con la salvedad de que la etapa analógica de la plataforma Red Pitaya utiliza conversores en configuración *single-ended*, por lo que tanto las transferencias medidas como la conexión al SR865 se adaptaron en consecuencia. Además, esta etapa analógica cuenta con conversores acoplados en continua capaces de operar hasta 125 MHz, lo que permitió extender el rango de prueba hasta 2 MHz, superando el límite impuesto por la interfaz del sistema basado en DE1-SoC. En todos los casos las tolerancias de los componentes pasivos utilizados fue del 1%.

Filtro pasa-bajos RC $f_c = 15,91 \, \text{kHz}$

Primero se probó el sistema con un circuito RC pasa-bajos con $R = 10 k\Omega \text{ y C} = 1 \text{ nF}$. La función de transferencia obtenida para el sistema puede verse en la Figura 4.17, en donde se muestra también la transferencia teórica esperada. Estas medidas se realizaron con N = 512 ciclos de promediación para cada frecuencia ensayada.

Los valores medios, máximos y mínimos del error cometido por el procesamiento en RP-li, tanto para la magnitud como para la fase de la transferencia, frente a la teoría y los valores obtenidos con el SR865 se muestran en el Cuadro 4.10. En este rango de frecuencias los errores se encuentran por debajo del 1%, sin importar que valor se tome como verdadero.

La frecuencia de corte de este filtro es teóricamente igual a $f_c = 15,91$ kHz. Con el procesamiento en RP-li puede medirse en 15,89 kHz, mientras que el procesamiento del SR865 resulta en una frecuencia de corte de 15,73 kHz. Esto representa un error de 0,8% tomando el valor del SR865 como valor verdadero de esta frecuencia de corte.

Filtro pasa-altos RC, con $f_c = 23, 4 \text{ kHz}$

El segundo circuito ensayado es idéntico al de la primera prueba, pero con los componentes R y C invertidos, de forma tal de formar un filtro pasa-altos. Los valores seleccionados para los



Figura 4.16: Medición de ruido con RP-li a medida que aumenta el tiempo de integración



Figura 4.17: Función de transferencia de una carga $\mathbf{R}=10\,\mathrm{k}\Omega$ y $\mathbf{C}=1\,\mathrm{nF}$ con RP-li



Figura 4.18: Función de transferencia de un filtro RC pasa-altos, con R = 680Ω y C = 10 nF, medida con RP-li

componentes fueron de R = 680Ω y C = 10 nF, de forma tal que la frecuencia de corte esté en aproximadamente 23,4 kHz. Los resultados pueden verse en la Figura 4.18

Los errores medios mínimos y máximos de las mediciones de amplitud y fase de la transferencia se presentan en el Cuadro 4.10. En este rango de frecuencias el procesamiento en RP-li resulta más parecido a la teoría que el del SR865. De todas maneras el error frente a ambos métodos de medida se mantiene en valores menores al 1%, con algunas excepciones en frecuencias particulares.

La frecuencia de corte de este filtro es teóricamente igual a $f_c = 23, 4$ kHz. Con el procesamiento en RP-li puede medirse en 22,9 kHz, mientras que el procesamiento del SR865 da una frecuencia de corte de 20,1 kHz, un error del 8 % considerando como verdadero valor al obtenido por el SR865, o del 2 % frente a los valores teóricos.

Filtro pasa-bajos RC $f_c = 234 \, \text{kHz}$

Como tercer circuito se ensayó un filtro RC igual al de la primera prueba, pero con componentes $R = 680 \Omega \text{ y } C = 1 \text{ nF}$, de forma tal que la frecuencia de corte esté en aproximadamente 234 kHz. Los resultados pueden verse en la Figura 4.19

Los errores medios, mínimos y máximos de la amplitud y fase de la transferencia se muestran en el Cuadro 4.10 donde se ve nuevamente que los valores obtenidos con el RP-li resultan más parecidos a la teoría que los obtenidos con el SR865.

La frecuencia de corte teórica de este filtro es de aproximadamente 234 kHz. Con el SR865 esta frecuencia de corte se puede estimar en 213 kHz y con el RP-li 250 kHz, un error del 17% considerando como verdadero valor al obtenido por el SR865, o del 6% frente al valor teórico.



Figura 4.19: Función de transferencia de una carga $\mathbf{R} = 680 \,\Omega$ y $C = 1 \,\mathrm{nF}$ con RP-li

4.6. Comparación de DE-li y RP-li

4.6.1. Aspectos generales

Los sistemas desarrollados en este capítulo comparten muchas características y presentan algunas diferencias importantes, aunque ambos resultan adecuados para las aplicaciones planteadas en esta investigación. El procesamiento en FPGA para cada dispositivo es bastante similar, aunque se observan diferencias de implementación debido a las variaciones entre los fabricantes. La interfaz de control de usuario y los modos de funcionamiento se encuentran en un estado más avanzado para el caso del DE-li. Por ejemplo, en esta etapa del desarrollo, el DE-li tiene la posibilidad de implementar los cálculos mediante algoritmo CALI o algoritmo LI, mientras que el RP-li realiza los cálculos directamente mediante algoritmo LI.

Las principales diferencias de ambos dispositivos, sin embargo, están dadas por sus etapas analógicas. La plataforma DE1-SoC, con su placa de adquisición de alta frecuencia *Highspeed AD-DA*, utilizada en el desarrollo del DE-li incorpora por diseño una etapa de acoplamiento AC, que recorta toda señal por debajo de 1 MHz. Para las aplicaciones apuntadas por esta investigación se debió adaptar esta etapa mediante un circuito propio de acoplamiento en CC (ver sección 3.2.2). El resultado es un ADC con entrada diferencial, con una etapa de acondicionamiento analógica propia diseñada para esta aplicación. El RP-li, por otro lado, tiene entradas single-ended, y viene acoplado en DC por diseño, por lo que no hizo falta una etapa adicional de acondicionamiento. Esto implica una mayor posibilidad de reproducibilidad para el RP-li. Ambos ADCs tienen 14 bits de resolución por lo que no se aprecian diferencias importantes en el límite de reducción de ruido mediante *Lock-in* dado por el error de cuantización de las señales.

La medición de transferencias conocidas con ambos dispositivos dio resultados similares con errores menores al 1%, comparando con las transferencias teóricas y contrastado con un *Lockin* comercial SR865, para frecuencias hasta 125 kHz. Los resultados en este intervalo dieron un poco mejor para el caso del DE-li. Para frecuencias más altas, las mediciones obtenidas con ambos Cuadro 4.10: Resumen de diferencias de medidas entre teoría, procesamiento en Red Pitaya y SR865 para experimentos con carga RC.

Experimento	Diferencias entre teoría y Red Pitaya	Diferencias entre SR865 y Red Pitaya	
$f_c = 15,91 \text{ kHz} \\ 0 < f < 125 \text{ kHz}$	Amplitud: Media: 0,0046 (0,46%) Mínima: 0,000015 (0,0015%) Máxima: 0,006211 (0,6212%)	Amplitud: Media: 0,0045 (0,45%) Mínima: 0,000011 (0,0011%) Máxima: 0,006846 (0,6846%)	
	Fase: Media: 0,9995° (0,2776%) Mínima: 0,0048° (0,0013%) Máxima: 1,1681° (0,3245%)	Fase: Media: 0,6661° (0,1850%) Mínima: 0,03698° (0,0103%) Máxima: 0,9189° (0,2553%)	
$\begin{array}{l} f_c = 23,4 \ \mathrm{kHz} \\ \mathrm{(Pasa-altos)} \\ 0 < f < 500 \ \mathrm{kHz} \end{array}$	Amplitud: Media: 0,0021 (0,2073%) Mínima: 0,0000 (0,0000%) Máxima: 0,0090 (0,9038%)	Amplitud: Media: 0,0059 (0,5925%) Mínima: 0,0000 (0,0000%) Máxima: 0,0516 (5,1578%)	
	Fase: Media: 0, 1711° (0, 0475%) Mínima: 0, 0000° (0, 0000%) Máxima: 0, 7976° (0, 2216%)	Fase: Media: 1,8422° (0,5117%) Mínima: 0,2098° (0,0583%) Máxima: 5,3585° (1,4885%)	
$\begin{aligned} f_c &= 234 \text{ kHz} \\ 0 &< f < 2 \text{ MHz} \end{aligned}$	R: Media: 0,0182 (1,8237%) Mínima: 0,0008 (0,0826%) Máxima: 0,0327 (3,2657%)	R: Media: 0,0352 (3,5187%) Mínima: 0,0005 (0,0490%) Máxima: 0,0665 (6,6540%)	
	Fase: Media: 2,2788° (0,6330%) Mínima: 0,6949° (0,1930%) Máxima: 2,6504° (0,7362%)	Fase: Media: 2,6556° (0,7377%) Mínima: 0,4291° (0,1192%) Máxima: 4,1090° (1,1414%)	

dispositivos resultan bastante similares a la teoría. Tanto las medidas con el RP-li como con el DE-li fueron más cercanas a la teoría que las obtenidas con el *Lock-in* comercial SR865. El RP-li mostró un rendimiento ligeramente superior en frecuencias altas, probablemente debido a su etapa analógica optimizada de fábrica. Además, debido a su ADC de mayor velocidad (125 MHz frente a 65 MHz) el RP-li tiene la capacidad de medir señales de frecuencias más elevadas.

La máxima frecuencia de reloj que puede alcanzarse es mayor para el caso del DE-li en su configuración CALI (216,54 MHz), y los recursos utilizados en la FPGA son un poco menores para el caso del DE-li, que se encuentra mejor optimizado e incluso incorpora más funcionalidades.

El cuadro comparativo 4.11 resume algunas de las características de los dispositivos desarrollados.

Característica	DE-li	RP-li
Acondicionamiento analógico	Diseño propio	De fábrica
Entrada ADC	Entrada diferencial	Entrada <i>single-ended</i>
Resolución ADC	14 bits	14 bits
Interfaz de usuario	Servidor Web HAL en c++/ Python	HAL en c++/ Python
Fabricante FPGA	Intel-Altera	AMD-Xilinx
Software de diseño	Quartus Prime	Vivado
Máxima frecuencia de operación	$216{,}54~\mathrm{MHz}$	$157 \mathrm{~MHz}$
Utilización de celdas lógicas	32%	49,7%
Error medio en R intervalo $0 < f < 125 \ \rm kHz$	Teoría: $0,28\%$ SR865: $0,09\%$	Teoría: $0,46\%$ SR865: 0.46%
Error medio en R intervalo $0 < f < 500 \ \rm kHz$	Teoría: $0,26\%$ SR865: $2,11\%$	Teoría: $0,21\%$ SR865: $0,6\%$
Error medio en R intervalo $0 < f < 2 \ {\rm MHz}$	Teoría: - SR865: -	Teoría: 1,8 % SR865: 3,5 %

Cuadro 4.11: Comparación de los dispositivos desarrollados

4.6.2. Mediciones en f=100 kHz

Para evaluar la capacidad de los sistemas desarrollados para la principal aplicación en la que se inscribe este trabajo de investigación (la detección de nanopartículas ferromagnéticas mediante tomografía magnética) se muestran en la tabla 4.12 los niveles de ruido obtenidos utilizando el DE-li, el RP-li y el SR865 a una frecuencia de 100 kHz, que es la apuntada por esta aplicación. Para cada dispositivo se generó una señal de la amplitud indicada en la tabla, y se midió mediante *Lock-in* utilizando distintas constantes de tiempo. Esta operación se repitió 100 veces, y se tomó la desviación estándar de las medidas como estimador del error cometido. A partir de esta medida se puede determinar cual es la menor variación de tensión detectable por el dispositivo.

La tabla 4.12 muestra que tanto el DE-li como el RP-li pueden detectar variaciones de tensión muy pequeñas, de algunos μ V en señales de aproximadamente 1V y 500 mV de amplitud, lo cual *a priori* lo hace apropiado para la aplicación de Tomografía Magnética. También se observa el límite en la precisión de los instrumentos, dados principalmente por el error de cuantización, como fuera predicho en secciones anteriores. Además, es interesante que ambos sistemas llegan a medir con valores de error comparables, e incluso mejores, a los del SR865, un *Lock-in* comercial de elevado costo.

Cuadro 4.12: Errores cometidos al estimar la amplitud de una señal de 100 kHz con los dispositivos desarrollados y con un *Lock-in* comercial. Los valores de la columna «Mínima variación detectable» están expresados en unidades $\times 10^{-6}$.

Dispositivo	Constante de	nte de Amplitud	Desviación	Mínima variación
	tiempo		estándar (σ_V) [μ V]	detectable $\frac{\sigma_V}{V}$ [×10 ⁻⁶]
DE-li	$10 \ \mu s$		284	306
	$320 \ \mu s$	$927 \mathrm{~mV}$	61	66
	$640 \ \mu s$		41	44
	1,28 ms		22	23
	$5,12 \mathrm{\ ms}$		8	8
	10,24 ms		8	8
	$500 \mathrm{ms}$		7	7
	$10 \ \mu s$		37	840
	$320 \ \mu s$		8,6	195
	$640 \ \mu s$	44 mV	$5,\!5$	125
	1,28 ms		2,9	65
	$5,12 \mathrm{\ ms}$		$1,\!2$	27
	$10,24 \mathrm{\ ms}$		0,7	15
	$500 \mathrm{ms}$		$0,\!4$	9
	$10 \ \mu s$	$509 \mathrm{~mV}$	182	357
	$320~\mu { m s}$		13	25
	$640 \ \mu s$		11	22
	1,28 ms		8	15
	$5,12 \mathrm{\ ms}$		7	13
RP-li	$10,24 \mathrm{\ ms}$		7	13
	$500 \mathrm{ms}$		6	11
	$10 \ \mu s$		10,2	221
	$320 \ \mu s$		$6,\!6$	143
	$640 \ \mu s$		$5,\!2$	113
	1,28 ms	46 mV	$3,\!5$	76
	$5,12 \mathrm{\ ms}$		2,2	47
	$10,24 \mathrm{\ ms}$		$1,\!3$	28
	$500 \mathrm{ms}$		1	21
SR865	$1 \ \mu s$		98	490
	$100 \ \mu s$		14	70
	10 ms	200 mV	11	55
	$100 \mathrm{ms}$		8	40
	1 s		7	35
4.7. Conclusiones

En este capítulo se ha mostrado cómo los sistemas de procesamiento de señales presentados en el capítulo 3 se pueden aplicar para implementar sistemas de detección coherente con diversos dispositivos SoC-FPGA. Se desarrollaron dos dispositivos, denominados DE-li y RP-li, que presentan características similares a los modelos comerciales, pero con la ventaja de poder ser fabricados utilizando elementos de software y hardware libre. El desarrollo completo de estos sistemas ha sido documentado, es de código abierto y está disponible en [115] y [116].

El sistema DE-li permitió verificar experimentalmente la equivalencia teórica entre el algoritmo LI, que utiliza un filtro de media móvil de N ciclos, y el algoritmo CALI, que promedia coherentemente N ciclos antes de aplicar un *Lock-in* de 1 ciclo a la señal promediada, como se describe en el capítulo 2. Este enfoque permitió mejorar la eficiencia temporal de los algoritmos de *Lock-in*, trasladando el cuello de botella de la implementación desde una multiplicación a una simple adición.

Ambos sistemas fueron evaluados mediante mediciones de ruido, y los resultados obtenidos muestran una excelente concordancia con los valores teóricos desarrollados en el capítulo 2. Además, los sistemas fueron probados midiendo transferencias conocidas, con errores generalmente menores al 1% en comparación tanto con los valores teóricos como con las mediciones realizadas por el *Lock-in* comercial SR865. Aunque ambos sistemas ofrecen mediciones satisfactorias en los rangos de frecuencia ensayados, el DE-li está mejor optimizado para aplicaciones hasta 125 kHz, mientras que el RP-li es más adecuado para mediciones por encima de ese rango. Es interesante destacar que no se observaron diferencias significativas en la calidad de las mediciones obtenidas con los sistemas desarrollados y el *Lock-in* comercial SR865, que es un equipo considerablemente más costoso y robusto.

En capítulos posteriores, el sistema desarrollado se empleará para el desarrollo de las aplicaciones de interés de este trabajo de investigación.

Capítulo 5

Aplicaciones

5.1. Tomografía de permeabilidad magnética

La tomografía de permeabilidad magnética (MIT, del inglés magnetic induction tomography) permite mapear, sin contacto y en forma no invasiva, las propiedades electromagnéticas pasivas (PEP) de un objeto mediante la medición de la inductancia mutua entre pares de bobinas colocadas alrededor de un perímetro [32], [33]. La MIT es sensible a las tres propiedades PEP de un objeto: conductividad eléctrica, permitividad dieléctrica y susceptibilidad magnética (σ , ϵ y χ_m respectivamente). Esta última es una constante de proporcionalidad adimensional que indica el grado de sensibilidad a la magnetización de un material influenciada por un campo magnético externo.

En el contexto de la hipertermia magnética se observa que existe una diferencia de varios ordenes de magnitud entre la χ_m de los tejidos biológicos y la χ_m de coloides terapéuticos de las nanopartículas (NPs) inyectadas en tumores. En ausencia de NPs, los tejidos biológicos tienen una χ_m del orden de 10^{-6} [34], mientras que se espera que la susceptibilidad de una dosis terapéutica de NPs de óxido de hierro esté en el rango de 10^{-4} a 10^{-3} [35]. Por lo tanto, ajustando la selectividad de MIT para la susceptibilidad magnética, podría emplearse esta técnica para detectar la presencia de NPs.

Esta técnica consiste en inyectar tensión alterna en bobinas excitadoras y medir la tensión inducida en bobinas sensoras. Una estimación de los parámetros de operación de un equipo como este ya se ha llevado a cabo en la introducción (sección 1.3.2.2). En este análisis se llegó a la conclusión de que excitar a las espiras con tensiones de 100 kHz constituye una buena solución de compromiso entre una frecuencia de operación lo suficientemente alta para efectuar la medición, y una lo suficientemente baja para despreciar los términos no deseados, provocados por σ y ϵ , y es la frecuencia objetivo que se tomará en esta implementación. Además el análisis llevado a cabo estima que se deben poder detectar variaciones de tensión del orden de 10^{-6} , que se estiman posibles con los dispositivos diseñados.

Esta técnica ha sido utilizada en una amplia gama de aplicaciones, principalmente en procesos industriales, pruebas no destructivas y obtención de imágenes biomédicas. En este último campo, la tomografía de mutua inductancia se ha desarrollado para obtener imágenes de conductividad, particularmente para mapear la conductividad de tejidos, facilitando la evaluación de accidentes cerebrovasculares hemorrágicos [117]. Por otro lado, la obtención de imágenes de permeabilidad ha sido explorada principalmente en entornos industriales, destacándose aplicaciones como la inspección de la integridad de barras de acero en estructuras de concreto y de tuberías metálicas en la industria del petróleo y gas [118], [119], [120], [121]. También se ha utilizado para reconstruir

imágenes de permeabilidad magnetostática 3D, logrando estimar la distribución de permeabilidad en objetos de prueba [122] y para reconstruir la permeabilidad magnética de materiales ferromagnéticos mediante imágenes de diferencia de frecuencia [123]. Como se ha mencionado en la introducción (sección 1.3.2.1), un buen resumen de estas técnicas está disponible en [36]. Según la literatura revisada, este trabajo, junto con el presentado en [23], constituye el primer diseño de tomografía magnética optimizado específicamente para obtener imágenes de permeabilidad en aplicaciones biomédicas.

Esta propuesta ofrece una solución más sencilla y económica para evaluar la persistencia del material magnético en comparación con tecnologías más costosas como la Tomografía Computarizada (TC) [124], la Imágenes por Resonancia Magnética (IRM) [125], la Imágenes por Partículas Magnéticas (MPI) [126], las mediciones de relaxometría [127], la tomografía magnetoacústica (MAT) [128], la tomografía por emisión de fotón único (SPECT) y la tomografía por emisión de positrones (PET) [129]. A diferencia de la TC, esta tecnología no utiliza radiación ionizante y, a diferencia de SPECT y PET, el material magnético se detecta sin necesidad de etiquetado. Además, un sistema MIT puede implementarse en dispositivos portátiles y no invasivos, una característica crucial para su combinación con tratamientos de hipertermia oncológica. El sistema diseñado tampoco requiere una preparación especial de la muestra y permite una interpretación rápida de los resultados sin necesidad de experiencia especializada.

En esta sección se describirá el diseño del tomógrafo magnético, que se implementó a partir de un modelo previo desarrollado en el grupo de trabajo y publicado en [23]. Seguidamente, se mostrará la incorporación de las herramientas desarrolladas en esta investigación al equipo, para luego verificar la operación del conjunto con medidas de un *phantom*; un objeto artificial que se utiliza para calibrar, probar y evaluar el rendimiento de los sistemas de imagen. Los detalles de la preparación del *phantom* están más allá de las pretensiones de este texto pero pueden encontrarse en [23]. Los resultados preliminares de esta sección están disponibles en [130] y [53], mientras que la publicación de los resultados finales está prevista para trabajos futuros.

5.1.1. Diseño del Tomógrafo Magnético

Una matriz típica de inducción magnética está compuesta por varias espiras inductivas, distribuidas equitativamente alrededor de una región circular de imagen, como se muestra en la Fig. 5.1. El procedimiento de medición consiste en excitar una de las bobinas con una corriente alterna y detectar la tensión inducida en las bobinas restantes. Cuando se coloca un objeto en la región de imagen, sus propiedades modifican el campo magnético detectado por las bobinas no excitadas. El procedimiento se puede repetir excitando una bobina a la vez, o rotando toda la matriz alrededor de su centro. Para obtener imágenes en 3 dimensiones se puede trasladar la matriz a lo largo del eje longitudinal perpendicular al plano de la matriz, dando como resultado una región de imagen cilíndrica.

Existen una serie de dificultades asociadas a la medición de los cambios en la tensión inducida en las espiras sensoras causadas por el objeto de interés. La primera de ellas es que los valores de variaciones $\frac{\Delta V}{V}$ para bobinas coaxiales son pequeños, en el orden de los 10^{-4} . Esta situación se agrava por el hecho de que la tensión V es mucho mayor en las bobinas de detección que están ubicadas más cerca de la bobina excitada. Por ejemplo, en una disposición circular simétrica, como se muestra en la Figura 5.1 A, la tensión inducida en la bobina 2 es más de 10 veces mayor que la inducida en la bobina 5 [123]. Este es un problema típico en la tomografía por inducción, para el cual se han propuesto varias soluciones, como el uso de gradiómetros [131] o magnetómetros tipo fluxgate [132]. Sin embargo, el alto rango dinámico del dispositivo *Lock-in* desarrollado durante



Figura 5.1: A) Matriz de ocho espiras con un objeto de interés en la región de imagen. B) Simulación estática utilizada para estimar la redistribución de la densidad de flujo (software de visualización Vizimag versión 3.185) cuando se excita la espira 1 y se coloca un escudo de alta permeabilidad en la periferia. C) Reconstrucción tomográfica a partir de las tensiones detectadas cuando se aplica tensión de corriente alterna (AC) a cada bobina en secuencia [23].

este trabajo de investigación permite realizar estas mediciones utilizando bobinas simples.

Una segunda dificultad instrumental surge del hecho de que las imágenes se reconstruyen utilizando la diferencia entre dos mediciones que no se pueden registrar simultáneamente, en lo que comúnmente se llama imagen por diferencia o técnica de diferencia temporal [133]. En este caso se miden previamente las tensiones inducidas en ausencia del objeto (V_0), y seguidamente se miden las tensiones generadas en presencia del objeto. Por lo tanto, es importante que las variaciones a largo plazo de esta tensión debido a la temperatura u otros factores externos (ΔV_0) no sean significativas en comparación con las variaciones de V (ΔV). Es decir que la medición es útil solo si $\Delta V_0 \ll \Delta V$. Por lo tanto, la minimización de la interacción térmica, mecánica y eléctrica con el entorno se vuelve relevante para aumentar la sensibilidad. La estabilidad a largo plazo del sistema determina con qué frecuencia deben medirse los valores de referencia V_0 . Así, una estabilidad deficiente ralentiza el procedimiento de imagen.

El tomógrafo diseñado consta de tres partes: una matriz de bobinas de ocho canales con blindaje magnético, un circuito controlador para la operación controlada del campo magnético, y el amplificador *Lock-in* DE-li diseñado a lo largo de esta tesis, con los circuitos que lo adaptan a esta aplicación en particular. Para esta aplicación se ha seleccionado el DE-li por encima del RP-li por su mejor rendimiento en las pruebas preliminares en el rango de frecuencias de interés para esta aplicación, como se vio en la sección 4.6. El diámetro de la zona de imagen (45 mm) se adoptó de las dimensiones de un aplicador de campo magnético AC portátil que se diseñó previamente para el tratamiento de hipertermia oncológica asistido con nanopartículas magnéticas en ratones [134], [135]. Esta elección proporciona compatibilidad entre los casos a investigar con ambos dispositivos.



Figura 5.2: A) Núcleo de ferrita (diámetro interior de 54,5 mm). B) Soporte impreso en PLA verde (diámetro interior de 45 mm) con una bobina colocada. C) Montaje final, incluyendo las 8 bobinas, antes de la instalación de los conectores LEMO y el receptáculo para las muestras [23].

5.1.1.1. Diseño del arreglo de bobinas y del circuito magnético

Las bobinas utilizadas son cuadradas y están distribuidas uniformemente alrededor del área de imagen. Su número de vueltas supone una relación de compromiso entre el espacio requerido y las necesidades de potencia del controlador, que impone el diámetro de los cables. El área de las bobinas también implica una relación de compromiso entre la magnitud de la señal (acoplamiento) y el grosor de la sección (resolución longitudinal). En el diseño final, las bobinas fueron diseñadas con 25 vueltas de alambre de cobre aislado de 0,2 mm en dos capas, con un área de 12,5 mm x 15,8 mm y una altura de 2,6 mm. La inductancia de las bobinas es de 23 μ H y su resistencia es de 0,9 Ω . Para caracterizar la repetibilidad en la fabricación, cada bobina fue medida a 100 kHz, resultando en un valor máximo de inductancia de 23, 85 μ H y un mínimo de 22, 59 μ H.

Para contener el flujo magnético fuera de la región de imagen se utilizó un núcleo de ferrita, mostrado en la Figura 5.2 A (modelo TDK B64290L0730X087-N87, diámetro exterior de 87,0 mm, diámetro interior de 54,3 mm, altura de 13,5 mm). Su alta permeabilidad relativa (2200) evita la interacción magnética con objetos cercanos, proporcionando un camino de retorno para el flujo fuera del área de imagen. Un soporte impreso en 3D, mostrado en la Figura 5.2 B, fue diseñado para las bobinas. Es fácil de reproducir y, si se imprime con ácido poliláctico (PLA) común, presenta buenas propiedades electromagnéticas [136]

El diseño con las 8 bobinas montadas se muestra en la Figura 5.2 C. La bobina excitadora está protegida con una lámina de cobre conectada a tierra para evitar la interacción capacitiva directa con las bobinas receptoras. Se utilizaron conectores autoacoplantes Lemo 00 y cables coaxiales de 50Ω para evitar interferencias entre canales.

5.1.1.2. Circuito controlador de campo magnético

Para aumentar la capacidad de conducción del DE-li la señal proveniente de su DAC fue reforzada por el circuito mostrado en la Figura 5.3. Este circuito incluye tres amplificadores operacionales OPA350 conectados en paralelo para alcanzar una corriente máxima de ± 150 mA. Un cuarto amplificador de retroalimentación asegura que la tensión en la bobina de excitación siga exactamente



Figura 5.3: Circuito de control para la bobina excitadora. Se incluyen resistencias de 3,3 Ω para igualar las salidas de los amplificadores. La red de 5,6 k $\Omega/3$, 3 pF se incluye para lograr estabilidad. Alimentado por una fuente de ±3 V, este circuito funciona correctamente a 100 kHz. Proporciona una corriente de hasta ±150 mA, suficiente para alimentar una bobina de 23 μ H con una tensión sinusoidal de 2 V de valor máximo.

la salida de referencia del DE-li. El conjunto de bobinas, el circuito de control y el DE-li fueron montados en una disposición portátil para facilitar su traslado.

En esta implementación, el circuito de control siempre está conectado a la misma bobina, inicialmente en la posición 1. Para obtener un conjunto de mediciones, el conjunto de bobinas se rota manualmente, de modo que la bobina de excitación se coloca en las posiciones de la 1 a la 8 en secuencia. Para cada una de estas posiciones, las otras 7 bobinas se utilizan como sensoras, seleccionadas manualmente mediante un conmutador mecánico de baja resistencia (menos de 2, Ω). En total se requieren 56 mediciones para cubrir todas las combinaciones posibles entre bobinas excitadoras y sensoras distintas. Estas mediciones se pueden representar como una matriz ΔV de 8 filas y 8 columnas, donde la diagonal principal es nula, ya que no se mide una espira contra sí misma.

5.1.1.3. Tiempos de medición

El tiempo que toman las mediciones está determinado principalmente por tres factores: el tiempo de conmutación manual de las bobinas, el tiempo necesario para medir los valores de referencia sin material magnético (V_0) , y el tiempo para medir las tensiones con material magnético. En la práctica, se ha observado que el primer factor afecta en gran medida al procedimiento completo, por lo que se decidió utilizar una constante de tiempo elevada para garantizar la mayor reducción posible de ruido en el *Lock-in*, considerando que de todas formas la principal limitación temporal del sistema está dada por la conmutación manual de espiras. Por este motivo, se seleccionó N = 50000 en el filtro de media móvil del *Lock-in*, lo que requiere 0,5 s para obtener cada lectura, considerando una señal de 100 kHz. Así, las 56 mediciones, sin considerar la conmutación de espiras, toman



Figura 5.4: Circuitos de adaptación del tomógrafo con el DE1-li. A) Amplificador de instrumentación AD620, utilizado para adaptar la señal diferencial proveniente del DE-li a la señal *single-ended* esperada por el tomógrafo. B) Circuito basado en el amplificador full-diferencial THP210 para adaptar la salida *single-ended* del tomógrafo con la señal de entrada diferencial del DE-li.

aproximadamente 28 s en completarse, o el doble si se incluyen las medidas de V_0 .

En un desarrollo futuro se prevee reemplazar el conmutador mecánico por un multiplexor electrónico de señales analógicas, controlado directamente por la plataforma SoC-FPGA del DE-li, y automatizar la rotación del conjunto de bobinas utilizando una configuración mecánica simple. Si se logran reducir estos tiempos también tendría sentido reducir la constante de tiempo, ya que, como se ha mostrado en la sección 4.6.2 promediar N = 512 ciclos de señal permite obtener mediciones con casi el mismo nivel de ruido. También es posible realizar mediciones más rápidas, aunque de menor calidad, mediante la lectura de un subconjunto de las bobinas, sacrificando resolución espacial.

5.1.1.4. Adaptación del DE-li al tomógrafo magnético

En esta configuración, tanto la entrada (tensión de excitación) como la salida (tensión inducida en las bobinas) son *single-ended*, mientras que las entradas y salidas del DE-li son diferenciales, por lo que fueron necesarios circuitos de adaptación de las señales. Para el caso de la señal de excitación se utilizó un amplificador de instrumentación AD620 [137], mientras que para la medición de las señales inducidas en las bobinas se diseño un circuito basado en el amplificador full-diferencial THP210 [138]. Ambos circuitos fueron desarrollados siguiendo los lineamientos de sus respectivas hojas de datos, y se muestran en la Figura 5.4.

5.1.1.5. Reconstrucción

El proceso de generación de imágenes propuesto es común a otros tipos de tomografía y consta de dos etapas: el modelado directo y el proceso de reconstrucción. El modelado directo comprende la simulación de los fenómenos electromagnéticos basada en las ecuaciones de Maxwell con condiciones de contorno apropiadas. El proceso de reconstrucción, también conocido como problema inverso, es la generación de la imagen tomográfica a partir de las mediciones y las simulaciones del modelado directo. Este es un problema mal condicionado, para el cual existen muchos algoritmos de reconstrucción diferentes que lo abordan desde distintas perspectivas. En este trabajo se utilizó el método clásico de Tikhonov, cuya descripción detallada queda fuera del alcance de este texto. Una descripción detallada de los algoritmos utilizados para la reconstrucción tomográfica a partir de las matrices ΔV medidas puede encontrarse en [23].

Espira Detectora	N	A $[mV]$	$\sigma \ [\ \mu V]$				
2	8 64 256	83,867 83,994 84,014	33,4 6,4 2,0	Espira Detectora	N	A $[mV]$	$\sigma_o \; [\; \mu \mathrm{V}]$
	$1024 \\ 50000$	84,014 84,013 84,002	$ \begin{array}{c} 2,0\\ 1,2\\ 0,8 \end{array} $	6	8 64 256	8,557 8,569 8,570	19,8 4,2 1.5
3	8 64 256	15,070 15,094 15,007	12,8 3,5 1.6	0	$1024 \\ 50000$	8,570 8,570 8,570	$ \begin{array}{c} 1,3 \\ 0,6 \\ 0,1 \end{array} $
	$1024 \\ 50000$	15,097 15,098 15,098	$ \begin{array}{c} 1,0\\ 0,8\\ 0,6 \end{array} $	7	8 64 256	14,845 14,895 14,807	16,2 3,8 1.7
4	8 64 256	8,597 8,610 8,612	13,0 3.7 1.4		$1024 \\ 50000$	14,897 14,898 15,898	1,7 1,0 0,6
	$1024 \\ 50000$	8,612 8,612 8,612	$0,6 \\ 0,1$	8	8 64 256	82,253 82,371 82,385	33,1 5,7 2.0
5	8 64 256	7,437 7,447 7,448	19,2 4,0		$1024 \\ 50000$	82,390 82,391	$1,0 \\ 0,6$
	$1024 \\ 50000$	7,448 7,448 7,448	$ \begin{array}{c c} 0,6 \\ 0,2 \end{array} $				

Cuadro 5.1: Estimación de la amplitud de tensión inducida y el error cometido por las mediciones en cada espira cuando se excita la espira 1.

5.1.2. Resultados

Con el tomógrafo construido y adaptado a su operación con el DE-li se procedió a hacer algunas medidas con un *phantom*, simulando la acción de las NP ferromagnéticas, cuya preparación se describe en detalle en [23]. Para estas mediciones el DAC del DE-li inyecta una señal de aproximadamente 2 V pico a pico y la tensión inducida en las espiras se mide utilizando *Lock-in* con filtro de media móvil de N = 50000 de ciclos, es decir que se promedian 0,5 s considerando una señal de 100 kHz.

5.1.2.1. Determinación del nivel de error en la medida de cada espira

Para estimar el nivel de error en la medición de las tensiones inducidas en cada espira de detección, se colocó la espira excitadora en la posición 1 (respecto a las posiciones indicadas en la Figura 5.1) y se tomaron 100 mediciones en cada espira detectora, variando las constantes de tiempo. Luego, se utilizó la media de estas mediciones como estimador de su amplitud y su desviación estándar como estimador del error en cada espira. Estos resultados se presentan en el cuadro 5.1. Para valores grandes de N el sistema es capaz de detectar la amplitud de la señal inducida en las espiras sensoras con muy buena sensibilidad, logrando errores por debajo del μV en todos los casos. En estas mediciones también es posible observar la variabilidad de las tensiones inducidas en cada espira, producto de la geometría del conjunto. Por ejemplo, se aprecia que la espira 1 induce una tensión mayor en las espiras 2 y 8, que son las más próximas espacialmente, en comparación con el resto de las espiras.

_

i.

1



Figura 5.5: Cambios en la tensión inducida en espiras 2 (izquierda) y 3 (derecha) como resultado de una tensión de excitación en la espira 1. Para cada figura se muestra, a la izquierda, la tensión total medida durante un intervalo completo de 15 minutos y a la derecha, las variaciones de tensión respecto al valor medio en intervalos menores de 2 minutos y medio.

5.1.2.2. Variaciones de V_0 en el tiempo

Con la espira excitadora en la posición 1 y la espira sensora en las posiciones 2 y 3 se tomaron medidas de tensión durante 15 minutos, para caracterizar los cambios en la tensión V_0 en cada caso. Estos resultados se muestran en la Figura 5.5. Si bien en periodos largos como el considerado la tensión inducida tiene variaciones debido al entorno, en periodos cortos se mantiene razonablemente estable.

5.1.2.3. Caracterización de las variaciones de tensión en espiras sensoras

Con la espira excitadora en la posición 1, se registró la tensión inducida en cada una de las espiras sensoras. Se realizaron medidas durante 1 minuto (una por segundo) sin *phantom*, luego durante 2 minutos con el *phantom* ubicado en su centro geométrico y, finalmente, 1 minuto adicional sin *phantom*. Posteriormente, se calculó el promedio de las tensiones medidas con y sin *phantom*, con lo que se determinó la variación de tensión inducida generada por su presencia (ΔV). A continuación, se repitió el procedimiento con el *phantom* ligeramente desplazado hacia arriba, más cerca de la posición 1, donde se encuentra la espira excitadora. Estos resultados se presentan en la Figura 5.6.



Figura 5.6: Detección de variaciones de tensión en las espiras provocados por *phantoms*. A la izquierda variaciones de tensión provocadas posicionando el *phantom* en el centro geométrico del área de medidas. A la derecha variaciones de tensión provocadas posicionando el *phantom* ligeramente por encima de su centro geométrico, en posición (0;0,4) respecto a un circulo de radio igual a 1.

Sobre esta se pueden extraer algunas conclusiones interesantes. En primer lugar, el cambio en la tensión inducida en la mayoría de las espiras sensoras es detectable por el sistema en ambos casos. Además, para el caso del *phantom* ubicado en el centro geométrico del área de medida, estos resultados son consistentes con los obtenidos en trabajos anteriores [23], aunque con niveles de tensión menores. La disposición geométrica y la posición central del *phantom* en este caso hacen que los cambios en la tensión inducida sean muy similares en pares de espiras ubicadas en las posiciones 2 y 8 (V2 y V8), 3 y 7 (V3 y V7), y 4 y 6 (V4 y V6). Las diferencias observadas en el ΔV medido en las espiras 4 y 6 (V4 y V6), atribuidas a posibles defectos de fabricación, así como la ausencia de cambios en las bobinas 3 y 7 (V3 y V7), son consistentes con los resultados reportados en estudios previos. Por otro lado, cuando el *phantom* se ubica más cerca de la espira excitadora, se producen mayores cambios en la tensión de la mayoría de las espiras sensoras, y se pierde la simetría observada en el caso anterior.

En el cuadro 5.2 se resumen algunas características de los ΔV a detectar en cada bobina para cada experimento. Observando estos valores, junto con los resultados de las secciones 5.1.2.2 y 5.1.2.1 se observa que, para intervalos temporales pequeños, se cumple que las variaciones de tensión debido a factores ambientales (ΔV_0) resultan significativamente menores a las variaciones de tensión debido a la presencia del *phantom* ($\Delta V_0 \ll \Delta V$), posibilitando las mediciones.

	Experimento 1			Experimento 2			
Espira [posición]	V [mV]	$\Delta V \left[\mu V \right]$	$\frac{\Delta V}{V}$ [%]	V [mV]	$\Delta V \left[\mu V \right]$	$\frac{\Delta V}{V}$ [%]	
1 [1]		[/]	V	L J	[/]	V	
2	04.00	× 0	0.000	01.00	11.0	0.01.4	
2	84.36	-5.6	-0.006	84.06	-11.8	-0.014	
3	14.94	0.0	0	15.22	5.2	0.032	
4	8.46	2.1	0.02	8.70	5.3	0.061	
5	7.80	3.3	0.04	7.41	3.2	0.043	
6	8.93	2.6	0.03	8.53	1.3	0.023	
7	15.25	0.0	0	14.88	0.0	0	
8	82.76	-5.5	-0.006	82.46	-5.5	-0.006	

Cuadro 5.2: Cambios en la tensión inducida en las bobinas sensoras cuando se introduce un *phantom* mientras se excita una espira en la posición 1. Experimento 1 corresponde a un *phantom* en el centro geométrico del área de medida, y Experimento 2 a un *phantom* ubicado ligeramente por encima de su centro geométrico, en posición (0;0,4) respecto a un círculo de radio igual a 1.

5.1.2.4. Reconstrucción de imágenes con phantoms

A partir de matrices de ΔV obtenidas con el sistema desarrollado y posicionando *phantoms* en distintos sectores del área de medida se reconstruyeron las imágenes tomográficas correspondientes. Estos resultados se muestran en las Figuras 5.7 y 5.8. Para cada medición se puede observar, arriba, la matriz correspondiente a las diferencias de tensión medidas en μV , al medio la reconstrucción obtenida, donde la escala coloreada corresponde a la función de distribución de probabilidades del centro de la partícula, y abajo una foto de la ubicación del *phantom* en cada caso. En todos los casos esta posición se ha podido predecir mediante la reconstrucción, verificando la viabilidad del método desarrollado.

5.1.3. Conclusiones

La tomografía de permeabilidad magnética es una técnica prometedora que puede utilizarse para detectar en forma no invasiva la presencia de nanopartículas ferromagnéticas durante terapias oncológicas de Hipertermia Magnética. Este procedimiento consiste en excitar alternativamente las espiras de un arreglo, sensando la tensión inducida en las restantes. Esta cambia si se posicionan nanopartículas ferromagnéticas entre espiras, por lo que se puede reconstruir la posición de las NP midiendo la tensión inducida en presencia y ausencia de ellas.

La instrumentación de un equipo de este tipo trae asociadas una serie de dificultades, algunas de las cuales ya han sido resueltas exitosamente en trabajos previos. En particular, para la excitación de las espiras y la detección coherente de los cambios inducidos en las espiras sensoras se había utilizado un *Lock-in* comercial, caro, poco portable y de difícil accionamiento y configuración. De hecho, la necesidad de cambiarlo por un diseño propio, basado en FPGAs o similar ya se menciona en [23]. En esta sección se ha mostrado cómo se pueden utilizar las herramientas desarrolladas durante esta tesis para optimizar el tomógrafo, dando como resultado un dispositivo mucho más práctico, confiable y rápido.

El amplio rango dinámico del DE-li desarrollado en el capitulo 4 permite detectar variaciones en la tensión en las espiras sensoras provocados por las NPs mediante una conexión directa, incluso cuando estas son mucho menores que las tensiones inducidas en ausencia de las NPs. Los cambios inducidos por el posicionamiento de *phantoms* en el área de medición se han cuantificado en esta



Figura 5.7: Reconstrucción mediante tomografía magnética de un *phantom* ubicado en distintas posiciones de la zona de medidas. Estas imágenes corresponden a una partícula ubicada en posición (0;0,4), a la izquierda, y (0,0), a la derecha, considerando al área de medidas como un círculo de radio 1.



Figura 5.8: Reconstrucción mediante tomografía magnética de un *phantom* ubicado en distintas posiciones de la zona de medidas. Estas imágenes corresponden a una partícula ubicada en posición (0,35;0,3), a la izquierda, y (-0,35,-0,15), a la derecha, considerando al área de medidas como un círculo de radio 1.

sección, verificando que estos pueden medirse y distinguirse de las variaciones de tensión en las espiras provocadas por factores ambientales.

Conociendo la geometría del área de medidas se pueden aplicar las leyes de Maxwell para modelar el sistema, en lo que se conoce como problema directo. Luego midiendo todas las posibles combinaciones de espiras excitadoras y sensoras se puede obtener una matriz de K x K, donde K es el número de espiras, en este caso 8. Utilizando esta información las imágenes tomográficas correspondientes pueden obtenerse en lo que se conoce como problema inverso. En este estudio se han tomado matrices con *phantoms* ubicados en distintos puntos del área de mediciones para luego reconstruir las imágenes correspondientes utilizando el método de Tikhonov. Se han ensayado cuatro posiciones distintas para los *phantoms*, verificando en cada caso que el sistema predice correctamente su posición.

La verificación de la operación del sistema muestra la utilidad de los dispositivos desarrollados durante esta investigación, con los que se pueden lograr resultados similares a dispositivos comerciales por una fracción de su precio, y con una mayor portabilidad y adaptabilidad al problema específico, y abre el camino a nuevas lineas de investigación en el tema. Pruebas *In Vivo* en pequeños animales ya han sido llevadas a cabo en [23], pero resta repetirlas utilizando los nuevos dispositivos desarrollados. Un aspecto importante a mejorar del sistema es la conmutación de espiras sensoras y excitadoras, que actualmente se realiza de forma manual. En este sentido se buscará reemplazar el conmutador mecánico por un multiplexor electrónico de señales analógicas, controlado directamente por la plataforma SoC-FPGA del DE-li, y automatizar la rotación del conjunto de bobinas utilizando una configuración mecánica simple.

5.2. Interfaces cerebro computadora

5.2.1. Introducción

Una interfaz cerebro computadora (BCI) es un dispositivo que provee al usuario un nuevo canal de comunicación y control para el cual no debe realizar acciones motoras [38]. Esto tiene la potencialidad de permitir que personas con movilidad reducida y pérdida del habla tengan la oportunidad de comandar un deletreador, una silla de ruedas o un mouse de computadora, por ejemplo [139], [140].

En general, las BCIs adquieren potenciales cerebrales del usuario, comúnmente mediante electroencefalografía (EEG), y los utilizan para comandar distintos dispositivos. Según el tipo de potenciales cerebrales que el sistema obtenga y procese, se pueden diseñar distintos tipos de BCIs [141]. Entre los sistemas más comunes se puede mencionar la desincronización/sincronización relacionada con eventos (ERD/ERS) [142], la detección de la componente P300 de potenciales relacionados con eventos (ERPs) [143], los potenciales corticales lentos (SCPs) [144] y los potenciales evocados visuales de estado estable (SSVEP, del inglés steady state visually evoked potentials).

En particular, los SSVEPs son potenciales periódicos que pueden obtenerse mediante el registro EEG en la zona occipital. Se presentan ante estímulos visuales periódicos superiores a 6 Hz y presentan la misma periodicidad que el estímulo [39]. Estos resultan de utilidad en interfaces cerebro computadora, ya que permiten relacionar el enfoque de atención de un usuario con un determinado comando. Su funcionamiento se basa en un estímulo visual E_1 que destella a una frecuencia conocida f_{E_1} , y un comando asociado a él. En caso de que el usuario esté enfocando su atención en E_1 la señal de EEG medida contendrá componentes en la frecuencia f_{E_1} y sus armónicos. De esta manera, si se detecta un componente de esa frecuencia lo suficientemente fuerte, se activará el comando. También podrían configurarse distintos comandos desfasando las señales de excitación, en vez de variar la frecuencia [145].

Este tipo de sistema tiene características ideales para la aplicación de sistemas de detección coherente en SoC-FPGA, ya que la señal evocada en el registro EEG es modulada por el mismo sistema, mediante estimulación lumínica en la frecuencia que se pretende detectar y, una vez medida, debe someterse a un procesamiento coherente, que detecte la presencia o ausencia de componentes en la frecuencia objetivo. En los sistemas tradicionales de SSVEPs [146], [147], [148], este procesamiento se realiza mediante la Transformada Discreta de Fourier (DFT), aunque existen estudios en donde se utiliza la técnica de *Lock-in* para la detección de estos componentes [149], [150], [151], [152]. Por otro lado, si bien las frecuencias de excitación no son particularmente altas, para poder detectar correctamente las señales evocadas es deseable que la adquisición y la estimulación estén perfectamente sincronizadas, algo difícil de cumplir con microprocesadores tradicionales, debido al carácter secuencial de las operaciones en una única unidad de procesamiento. Además, el proceso de adquisición de las señales también supone una carga de procesamiento elevada para el sistema, debido a la naturaleza de las señales involucradas.

El uso de SSVEPs para implementar BCIs está ampliamente estudiado. Este tiene la ventaja de lograr dispositivos simples de usar, con tasas de transmisión de información (ITR, del inglés *information transfer rate*) altas [39], [40]. En la mayoría de las publicaciones disponibles [139], [153], [154] los sistemas utilizan una etapa de adquisición de biopotenciales y una PC, lo que limita ampliamente la portabilidad del conjunto. Las implementaciones portables de estos tipos de sistemas, por otro lado, son escasas en la bibliografía disponible, y es hacia donde se pretende avanzar.



Figura 5.9: Interfaz cerebro computadora basada en SSVEPs

Por estos motivos, el sistema de detección coherente de señales desarrollado durante esta tesis tiene aplicaciones interesantes en este campo. La naturaleza paralela de la lógica programable permite ejecutar sistemas de detección coherente en varias frecuencias en forma simultánea, permitiendo sistemas de SSVEPs con múltiples comandos y manteniendo perfecta sincronización entre adquisición y generación de estímulos, sin que la adquisición se vea comprometida. Además, estos sistemas resultan portables y no dependen de una computadora personal para su funcionamiento.

Teniendo esto en cuenta se diseñó un sistema para detección de SSVEPs con codificación frecuencial en un dispositivo FPGA MAX 10 10M50DAF484C7G de Intel-Altera, disponible en una placa de desarrollo DE10 LITE de Terasic [155], que realiza en tiempo real la detección por el método de *Lock-in*, y la generación de los estímulos visuales. A diferencia de otros dispositivos estudiados en esta investigación este dispositivo no cuenta con un μ P embebido. En este capítulo, se describirá en detalle esta implementación, que es de código abierto y está disponible en [156].

Los resultados de esta sección han sido presentados en [157], [158], [159], [160], [161] y [162].

5.2.2. Implementación del sistema

El esquema general del sistema para detección de SSVEPs puede verse en la Figura 5.9. La señal es obtenida utilizando electrodos activos situados en la región occipital del cráneo del usuario y medida con un conversor analógico-digital ADS1299, gobernado a través de lógica dedicada diseñada en la FPGA. Al mismo tiempo que se adquiere la señal, la lógica genera señales de excitación en tres frecuencias distintas ($f_1 = 16,67$ Hz, $f_2 = 19,23$ Hz y $f_3 = 14,70$ Hz) que comandan pantallas LED que el usuario puede observar alternativamente, y que constituyen los estímulos visuales con los que

el usuario comanda la BCI. La señal de EEG es sometida a procesamiento coherente en la FPGA, para detectar la magnitud de la respuesta en cada una de las frecuencias de interés. Periódicamente se envía a la PC la señal de EEG y la magnitud de cada componente frecuencial obtenida por el procesamiento coherente a través de un puerto serie. Todo el sistema debe estar aislado de la red eléctrica para resguardar la integridad del usuario. A continuación, se profundizará en cada una de las etapas de este proceso.

5.2.3. Adquisición

La primera etapa relevante en un equipo de SSVEPs es la de adquisición. Cuando se miden señales biomédicas del cuerpo, los electrodos transductores introducen un desplazamiento de corriente continua con un rango de $V_{\text{off}_{\text{max}}} = \pm 300 \text{ mV}$. Este valor se toma de los estándares de electrocardiografía (ECG) para un canal diferencial [163] y, aunque es pesimista, suele ser el caso que el desplazamiento se encuentre dentro de un rango de 10 mV a 100 mV. Por otro lado, las señales de biopotenciales deben medirse con un piso de ruido del orden de 1 μV_{rms} o menos [164], [165], lo que puede traducirse en una amplitud de señal de 6 μV_{pp} considerando $\pm 3\sigma$, lo que lleva el rango dinámico a $20\log_{10}(\frac{600 \text{ mV}}{6 \mu \text{ V}}) = 100 \text{ dB}.$

La medición de señales de EEG puede incluso necesitar un rango dinámico mayor que 100 dB. Una evaluación exhaustiva del ruido de entrada del equipo de EEG se lleva a cabo en [166], donde los autores concluyen que una amplitud espectral de ruido de 10 $\frac{nV}{\sqrt{Hz}}$ es apropiada para las señales más pequeñas y solo en caso de abrasión o perforación de la piel podría ser necesaria una menor cantidad de ruido. Considerando un ancho de banda de 1 kHz, esto significa medir con pisos de ruido de 10 $\frac{nV}{\sqrt{Hz}}\sqrt{1 \text{ kHz}} = 0,316 \,\mu\text{V}$, es decir, un ruido de $0,24 \,\text{V}_{\text{pp}}$, o un rango dinámico de 112 dB considerando la señal de continua sobreimpuesta por los potenciales de electrodo.

Para lograr esta medición existen dos posibles caminos. El primero de ellos; el más tradicional, consiste en aplicar un filtrado pasa-alto a la señal, es decir, acoplarla en alterna. Al bloquear el desplazamiento de corriente continua, el rango dinámico necesario se reduce a aproximadamente 70 dB necesarios para la señal misma. En estos casos, la estrategia habitual es aplicar el filtrado en combinación con un factor de amplificación relativamente alto para permitir la adquisición con un ADC de 12 a 14 bits. Sin embargo, hay una serie de desventajas en esta metodología. La primera es la necesidad de una etapa analógica capaz de proporcionar una ganancia relativamente alta del orden de 100-1000 veces, mientras se rechazan los componentes de desplazamiento de corriente continua del electrodo [167]. La segunda está relacionada con una forma de interferencia que afecta las mediciones biomédicas llamada artefactos, que se producen cuando los electrodos o la piel son perturbados mecánicamente (por ejemplo, al tirar de los cables o por el desplazamiento de los propios músculos). Los artefactos producen amplias fluctuaciones de la señal de base que pueden saturar los filtros y las etapas de amplificación, lo que resulta en una demora considerable para restaurar un rango operativo óptimo. Además, los front-end con amplificación adicional pueden requerir circuitos de balanceo más complejos [168] en comparación con las soluciones de ganancia unitaria [169].

Así, los convertidores $\Sigma\Delta$ de alto rango dinámico y bajo ruido se han identificado como ventajosos para las mediciones de señales biomédicas [170], y esta estrategia se ha incorporado en sistemas de adquisición integrados de última generación [171], [172]. La industria de semiconductores ha incluido estos ADC en productos estándar específicos de aplicaciones comerciales (ASSPs, por sus siglas en inglés) como las líneas ADS129x y ADS13x de Texas Instruments y AD7779 de Analog Devices, como algunos ejemplos. Estos adquisidores son útiles en sistemas basados en lógica progra-

Ganancia - ODR $^{\rm 1}$	1 - 1 kHz	12 - 1 kHz	1 -16 kHz	12 - 16 kHz	1 -250 Hz	12 - 250 Hz		
ADS131E08								
$DR^2 [dB]$	117,7	108,0	102,8	94,2	-	-		
ENOB 3 [bits]	19,6	18,0	17,07	$15,\!65$	-	-		
RTI ⁴ $[\mu V_{rms}]$	$2,\!13$	$0,\!54$	12,33	2,75	-	-		
ADS1299								
$DR^2 [dB]$	123.3	118.3	103.3	101.3	129,3	124,3		
ENOB 3 [bits]	20.49	19.65	17.16	16.83	21,48	$20,\!65$		
RTI ⁴ $[\mu V_{rms}]$	2.17	0.32	21.70	2.27	1,08	0,16		

Cuadro 5.3: Especificaciones del adquisidores comerciales de interés.

¹ Tasa de datos de salida

² Rango Dinámico

³ Número efectivo de bits

⁴ Tensión de ruido referido a la entrada

mable como front-ends de señales biomédicas [173], [174]. Con estos conversores se puede lograr un nivel de ruido referido a la entrada (RTI, del inglés reffered to input) lo suficientemente bajo como para permitir la adquisición de señales biomédicas sin amplificación, preservando todo el rango dinámico del dispositivo. Sin embargo, aumentar la tasa de datos de salida (ODR, del inglés *output* data rate) resulta en una degradación de las propiedades de ruido de estos dispositivos debido a su compensación fundamental entre resolución y ancho de banda. Para ilustrar este punto, se presentan en el Cuadro 5.3 los valores obtenidos de la hoja de datos de dos de estos ADCs comerciales, el ADS131E08 [175] y el ADS1299 [176].

5.2.3.1. Promediado de canales

Para disminuir los niveles de ruido que se muestran en el Cuadro 5.3 y lograr que estos resulten compatibles con las medidas de EEG necesarias para impulsar un equipo de SSVEPs, se decidió mejorar el equipo de adquisición aprovechando una característica interesante de estos conversores: cada uno tiene 8 canales independientes de medición. Como solo se necesita medir un canal para la detección de SSVEPs, se puede adquirir simultáneamente la misma señal con todos los canales y luego promediar las muestras digitalizadas [166], como se muestra en la Figura 5.10. En este punto es importante notar que, si bien la operación de promediado de canales podría ser muy exigente para microprocesadores tradicionales, no supone una carga significativa para la FPGA, debido a la naturaleza paralelizable de sus celdas lógicas.

Para un análisis simplificado de esta operación de promediado se puede partir considerando que una señal $v_{in}(t)$ es muestreada, obteniendo:

$$x(k) = v_{in}(kT) + n_{ADC}(k) = s(k) + n_{ADC}(k)$$
(5.1)

donde $n_{ADC}(k)$ es el ruido introducido por la conversión, y s(k) es la señal de interés. Ahora,



Figura 5.10: Promediado de canales independientes.

suponiendo que se muestrean N canales independientes y se promedian los resultados, se obtiene:

$$y(k) = \frac{1}{N} \sum_{i=1}^{N} x_i(k) = \frac{1}{N} \sum_{i=1}^{N} (s_i(k) + n_{\text{ADC},i}(k))$$

= $s(k) + \frac{1}{N} \sum_{i=1}^{N} n_{\text{ADC},i}(k) = s(k) + n_T(k)$ (5.2)

La salida y(k) contiene la señal de interés s(k) sin modificar y el ruido promediado de todos los canales de adquisición, $n_T(k)$. Dado el supuesto de que el ruido es un proceso aleatorio independiente e idénticamente distribuido (iid) con distribución gaussiana, el valor rms puede obtenerse calculando su desviación estándar σ_{ADC} , que puede estimarse mediante la desviación estándar a lo largo del tiempo de muestreo, bajo el supuesto de ergodicidad.

La varianza del ruido en cada instante k estará dada por:

$$E\{n_T(k)^2\} = \sigma_T^2 = E\{\left(\frac{1}{N}\sum_{i=1}^N n_{ADC,i}\right)^2\}$$

= $\frac{1}{N^2}\sum_{i=1}^N \sigma_{ADC}^2 + \frac{1}{N^2}\sum_{i=1}^N \sum_{j=1}^{N,i\neq j} \rho_{i,j}\sigma_i\sigma_j$ (5.3)

donde se ha supuesto que el valor esperado del ruido $E\{n_{ADC}\}$ es igual a 0, y $\rho_{i,j}$ es la correlación entre pares de canales. Suponiendo que los canales son completamente independientes, esta correlación será igual a 0 para canales diferentes, por lo que la varianza del ruido estará dada por:

$$\sigma_T(k)^2 = \frac{\sigma_{ADC}^2}{N} \therefore \sigma_T(k) = \frac{\sigma_{ADC}}{\sqrt{N}}$$
(5.4)

Por lo que, promediando N canales de adquisición, se obtiene una mejora en la SNR de \sqrt{N} . Si, en cambio, todos los canales estuvieran completamente correlacionados ($\rho_{i,j} = 1$ para todo i,j) $\sigma_T(k) = \sigma_{ADC}$, es decir que el promediado de canales no aportaría nada a la medición (lo cual es lógico ya que sería medir con exactamente el mismo ADC en el mismo momento la señal).

Para el caso de 8 canales, que es el que se aborda en el caso actual, la mejora en el SNR es teóricamente igual a $\sqrt{8} \approx 2,82$, es decir, una mejora de 9 dB en el rango dinámico disponible. Volviendo al Cuadro 5.3 esto permitiría un rango dinámico superior a los 112 dB objetivos utilizando cualquiera de los conversores con tasas de datos de 1 kHz o 250 Hz.

5.2.3.2. Interferencia electromagnética

Además del problema de los potenciales de continua introducidos por la interfaz electrodo-piel, en un equipo de adquisición de biopotenciales es importante tomar medidas para rechazar la interferencia electromagnética (EMI) debido a la tensión de red. Con este objetivo se han incorporado al sistema electrodos activos implementados con amplificadores operacionales TLD2201 configurados como *buffers*, para mitigar la EMI debido a acoplamientos capacitivos a los cables, como es su función habitual [177]. En el esquema pretendido de promediado de canales, estos electrodos cumplen una función extra: los *buffers* de entrada resultan primordiales para evitar la degradación de la impedancia de entrada debido a la conexión a múltiples etapas de entrada en paralelo. Además, se ha incorporado al sistema un circuito DRL (del inglés *driven right leg*) independiente. Este último sirve para fijar la tensión de referencia del paciente, y está ampliamente estudiado en la bibliografía disponible [178], [179].

5.2.4. Alimentación y aislación

El sistema se implementó como una plataforma completa de adquisición de biopotenciales, incluyendo aislamiento de grado médico para realizar las mediciones *in-vivo*. La aislación se logró a través de un ADuM6401 [180] que permite implementar un bus SPI aislado. El sistema se alimenta desde baterías de litio reguladas del lado aislado, y desde el bus USB proveniente de la PC en el otro extremo. De esta manera, la alimentación del sistema digital se mantiene separada de la alimentación analógica del ADC, de la alimentación de los electrodos activos y del circuito de DRL.

5.2.5. Detección

Una vez adquirida la señal, esta pasa a la etapa de detección, que tiene la responsabilidad de determinar la magnitud de la componente de la señal en cada una de las frecuencias de interés. Para ello, se implementó una etapa de *Lock-in* de dos fases por cada frecuencia de estímulo, como se muestra en la Figura 5.11, para lo que se utilizaron los módulos diseñados y validados en el capítulo 4. La deducción de las ecuaciones que gobiernan el comportamiento de este tipo de sistemas ya ha sido tratada en el capítulo 2 pero se repasa brevemente aquí.

Por cada frecuencia f_e se produce una señal discreta de referencia de valor $r(n) = A\sin(2\pi f_e n)$. Esta se utiliza para los cálculos del *Lock-in*, y además se pasa por una compuerta para obtener una onda cuadrada que se exterioriza por pines digitales de salida de la FPGA, para generar las señales de excitación. Para generar estas señales se utilizó una tabla de consulta fija, de M puntos. La naturaleza discreta de esta señal y la frecuencia de muestreo del ADC determinan la frecuencia real de los estímulos visuales. Para esta implementación se decidió tomar una frecuencia de muestreo de 250 muestras por segundo. Si se quiere generar una señal cercana a 16 Hz, por ejemplo, se debe tomar el valor de M que produzca la señal con frecuencia más cercana a este valor, es decir:



Figura 5.11: Detección de SSVEPs con Lock-in y generación de estímulos

$$f_{\text{objetivo}} = 16$$

$$M = \text{floor}(\frac{f_s}{f_{\text{objetivo}}}) = \text{floor}(15, 625) = 15$$

$$f_{\text{real}} = \frac{f_s}{M} = \frac{250 \text{ Hz}}{15} = 16, 67 \text{ Hz}$$
(5.5)

Un análisis análogo con frecuencias objetivo de 14 Hz y 16 Hz arrojan frecuencias reales de 14,70 Hz y 19,23 Hz, y son estas las frecuencias que se han utilizado para la implementación del sistema.

La salida de los conversores ADC, luego de la promediación descripta en la sección anterior, es una secuencia (señal discreta) s(n), donde n es la variable discreta de tiempo. Esta señal se somete a un filtro pasa alto con una frecuencia de corte de 3 Hz para remover su valor medio, resultando $s_{ac}(n)$ y luego se envía a los módulos de detección. Cada uno multiplica la señal medida con la señal de referencia correspondiente, y una versión en cuadratura de la referencia, resultando:

$$r_c(n) = r(n)e^{j\pi/2}$$

$$x(n) = s_{ac}(n)r(n)$$

$$y(n) = s_{ac}(n)r_c(n)$$

Para la componente de la señal $s_{ac}(n)$ que tiene la misma frecuencia que la señal de referencia, esta multiplicación genera una componente en banda base y una componente en el doble de la frecuencia. Aplicando un filtro pasa-bajos luego de cada multiplicación, se puede eliminar el componente del doble de frecuencia, resultando señales X e Y, que contienen la información del módulo y fase de la señal $s_{ac}(n)$ en la frecuencia de interés, según:

$$A_{E_i}(n) = \sqrt{X^2 + Y^2}$$
 (5.6)

$$\phi_{E_i}(n) = \arctan \frac{I}{X} \tag{5.7}$$

v

Esta combinación de mezclador y filtro pasa-bajos tiene la propiedad de atenuar en gran medida las componentes de la señal de frecuencias distintas a f_e . Un elemento crucial en este sistema es el filtro pasa-bajos que se utiliza para eliminar los componentes fuera de la banda base luego de la etapa de mezclado. Mientras más selectivo sea este filtro, mejor podrá eliminar componentes de la señal que se encuentren cercanas a f_e . En esta implementación, se utilizaron filtros de media móvil, que se actualizan muestra a muestra y abarcan una ventana de tiempo configurable, correspondiente a N muestras. Como ya se ha estudiado en la sección 2.2.5 si N se escoge como un número entero de ciclos de la señal, se produce un filtrado total de la componente de doble de frecuencia f_e que se produce luego del batido. Luego de pruebas preliminares, se determinó una ventana de largo de N muestras equivalente a 2 segundos como un factor de compromiso entre la selectividad del filtro y la responsividad del sistema.

La señal de fase puede ser útil en el caso de contar con estímulos codificados en fase [145]. En esta implementación se optó por una codificación frecuencial de comandos, por lo que solo se utilizó la salida en módulo.

5.2.6. Estimulación visual

Los estímulos visuales se implementaron a partir de una matriz de 8x8 leds WS2812B disponible comercialmente, a la que se le aplicó un difusor para generar una iluminación uniforme potenciando los SSVEP [4] en un rectángulo de 6 cm \times 6 cm. Estudios anteriores han demostrado que la estimulación con paneles LED produce una mayor amplitud en la frecuencia fundamental del SSVEP, frente a otras opciones como monitores CRT o LCD [181]. El panel se controló con un microcontrolador que recibe el bus de sincronización de estímulos de la FPGA, lo que garantiza la coincidencia temporal entre la adquisición y la estimulación.

5.2.7. Resultados

5.2.7.1. Medidas de ruido de adquisición con ADS131E08

En una etapa preliminar del trabajo, se decidieron hacer medidas utilizando el ADS13E08 para estudiar experimentalmente las conclusiones teóricas de la sección 5.2.3.1. Para ello, todos los canales de este conversor fueron cortocircuitados a una referencia de tensión utilizando el multiplexor interno del dispositivo. El ruido resultante en tiempo real (RTI) fue evaluado tomando 32000 muestras a 1 kHz y 16 kHz para todos los niveles de ganancia disponibles (1, 2, 4, 8 y 12), y en los niveles de ganancia 1 y 12 para todas las tasas de datos de salida disponibles (1, 2, 4, 8 y 16 kHz). Los resultados fueron comparados con los datos extraídos de la hoja de datos del ADS131E08 [175], y se muestran en la Figura 5.12 y en la Figura 5.13.

En la Figura 5.12 se muestran dos conjuntos de valores del ruido RTI medido, unidos por una línea completa: corresponden al ruido de un canal a 1 kHz y 16 kHz de ODR para sus 5 configuraciones de ganancia. Una ganancia más alta resulta en un ruido RTI más bajo, ya que la contribución de las etapas posteriores al amplificador de ganancia programable tiene menos peso. El ruido efectivo reportado en la hoja de datos del componente se muestra en pequeños puntos y coincide hasta dentro de un 5% con todas las mediciones, excepto a 16 kHz, ganancia 12, que consistentemente desvió con un 13% menos de ruido. En la misma figura, se muestran los resultados de las mediciones promediadas con una línea punteada, verificando que de hecho se obtuvo una reducción de ruido. A una tasa de datos de 1 kHz, el ruido se redujo en un factor de 2,9 \pm 0,1, y a 16 kHz en 2,97 \pm 0,08. Las mediciones de ruido de las configuraciones de ganancia más baja y más alta (1 y 12) se muestran además en la Fig. 5.13, nuevamente unidas por una línea completa para



Figura 5.12: Ruido referido a la entrada medido para tasas de datos de 1 kHz y 16 kHz en diferentes configuraciones de ganancia. La línea completa sigue las mediciones de un solo canal y la línea punteada representa el promedio de los 8 canales. Los valores reportados en la hoja de datos se muestran en pequeños puntos negros.



Figura 5.13: Ruido RTI medido para una ganancia de 1 y 12 en diferentes configuraciones de tasas de datos. La línea completa sigue las mediciones de un solo canal y la línea punteada representa el promedio de los 8 canales. Los valores reportados en la hoja de datos se muestran en pequeños puntos negros.



Figura 5.14: Sistema implementado



Figura 5.15: Setup experimental. Se observan los electrodos de medida, el DRL en la frente, y el estimulador.

el ruido RTI del caso de un solo canal, con marcadores en las 5 configuraciones de tasa de datos y pequeños marcadores de puntos que muestran los valores de la hoja de datos del componente. Se obtuvo una reducción promedio de ruido de $2,80 \pm 0,05$ para la ganancia 1 y $3,04 \pm 0,02$ para la ganancia 12.

Los resultados de las figuras 5.12 y 5.13 muestran que para el ADS131E08 es posible reducir el ruido mediante el promediado de canales. El factor de reducción es ligeramente mayor que el esperado para el promedio de 8 canales ($\sqrt{8} \approx 2,82$) y podría ser explicado por una desviación de las suposiciones estadísticas realizadas y el necesario uso de una estimación de la desviación estándar. Sin embargo, es un indicador de que no hay correlaciones entre canales que impidan la reducción de ruido. La reducción medida más baja coincide con el valor teórico, y efectivamente extiende el rango dinámico en 9 dB.

5.2.7.2. Implementación del sistema de detección de SSVEPs

El sistema implementado se muestra en las Figuras 5.14 y 5.15. Para este sistema se utilizó el ADS1299. Si bien este no es exactamente el estudiado en la sección 5.2.7.1 es de la misma línea de conversores, más orientado hacia aplicaciones específicas de electrofisiología como EEG.

Parámetro	Valor
Ruido RTI	$0,24\mu\mathrm{V_{rms}}$
Tasa de datos	250 Hz
Ancho de banda	$64 \mathrm{~Hz}$
Ancho de banda de detección	$0,\!45~\mathrm{Hz}$
	$f_1 = 16,67\mathrm{Hz}$
Frecuencias de estímulo	$f_2 = 19,23\mathrm{Hz}$
	$f_3 = 14,70\mathrm{Hz}$

Cuadro 5.4: Parámetros de funcionamiento

Elemento	Uso (porcentaje)
Elementos lógicos	28016~(56%)
Bits de memoria	386560~(23%)
Multiplicadores embebidos (bloques DSP)	216~(75%)
PLLs	1 (4%)
Registros	8763~(18%)

Cuadro 5.5: Reporte de recursos de la FPGA MAX10



Figura 5.16: Simulación de la respuesta en frecuencia del sistema de detección

Los parámetros de funcionamiento del equipo se muestran en el Cuadro 5.4, y el reporte de compilación de la FPGA en el Cuadro 5.5. La densidad espectral de ruido referido a la entrada se muestra en la Figura 5.17 y la respuesta en frecuencia esperada del sistema de detección en la 5.16.

5.2.7.3. Medidas de SSVEPs

Para comprobar la capacidad del sistema para detectar los potenciales evocados visuales, se implementó un protocolo automatizado que le muestra al usuario una luz verde durante 5 segundos señalizando el comienzo del experimento, luego 5 segundos sin estímulo, seguido de 5 segundos con el estímulo f_1 , repitiendo este patrón para los estímulos f_2 y f_3 . Los datos generados se capturan en una PC donde se recibe la versión acoplada en alterna de la señal cruda y la salida de cada uno de los *Lock-in*.

Los electrodos fueron de tipo copa dorada aplicados con pasta de EEG colocados en posición



Figura 5.17: Densidad espectral de ruido referido a la entrada

O1, O2, y el DRL se colocó en la frente con electrodos adhesivos húmedos estándar de Ag/AgCl. Se realizaron pruebas con 3 participantes S1, S2 y S3 con grado bajo, alto y medio de entrenamiento previo con otras interfaces basadas en SSVEP respectivamente. Las salidas del sistema para dos pruebas de los 3 participantes se observan en la Figura 5.18, donde puede verificarse por inspección visual la detección del SSVEP generado por cada frecuencia de estimulación.

En esta figura se incluyen en línea punteada marcadores de referencia para el cálculo de la relación señal a ruido (SNR) obtenida. Para ello, se consideró la presencia de estímulos espaciados cada 5 segundos, pero se agregó un retardo para compensar la variabilidad natural en la generación de SSVEP, la ventana de tiempo del *Lock-in* y la indeterminación en el orden de algunas décimas de segundo cuando el experimentador iniciaba la grabación de señales ante la luz verde que indicaba el inicio del protocolo automatizado. La SNR se obtuvo según la siguiente ecuación:

$$SNR_{i} = \frac{\sum_{t_{i}}^{t_{i}+5s} x_{f_{i}}(t)}{\sum_{t_{i}-5s}^{t_{i}} x_{f_{i}}(t)}$$
(5.8)

Donde x_{f_i} es la señal de cada detector y t_i es el tiempo de inicio de la ventana de 5 s que contiene al estímulo. Se obtuvo un valor promedio de $3,2 \pm 1,7$ y, por inspección visual, se observa que en las ventanas con SNR > 2 la presencia de estímulo es fácilmente distinguible.

5.2.8. Conclusiones

Durante este capítulo se ha mostrado cómo pueden aplicarse los sistemas de detección coherente desarrollados durante esta tesis el desarrollo de interfaces cerebro computadora, incorporándolo como un elemento más en un sistema funcional y no simplemente como un aparato de medida,



Figura 5.18: Salidas de los detectores implementados. Los números indican la relación señal a ruido comparando la señal dentro de los 5 s señalados por la línea punteada con la señal 5 s inmediatamente antes. S1, S2 y S3 indican los participantes

como suele utilizarse un *Lock-in* en un laboratorio tradicional. Esto ejemplifica la potencialidad de las herramientas de detección coherente desarrolladas.

También se ha mostrado cómo la elección de una FPGA como plataforma de procesamiento permite la implementación paralela de los módulos, garantizando completa sincronización entre los estímulos, la adquisición y el procesamiento de la señal, además de permitir mejorar la calidad de las mediciones a través del promediado de canales de ADC. Como se ve en el reporte de compilación y teniendo en cuenta que los multiplicadores embebidos pueden reemplazarse con elementos lógicos en el diseño, aún se podrían implementar más módulos con esta plataforma, por lo que resulta fácilmente escalable. Además, la plataforma utilizada no fue exactamente la misma que en capítulos anteriores de este trabajo, lo cual pone en evidencia la flexibilidad y reconfigurabilidad de las herramientas desarrolladas durante esta investigación.

Desde el punto de vista específico de la aplicación, los experimentos llevados a cabo muestran la capacidad del sistema para adquirir la señal de electroencefalografía, detectar en tiempo real la presencia de SSVEPs mediante *Lock-in* con filtro de media móvil de 2 s y transmitir simultáneamente los datos a una PC. También se comprobó, analizando los registros de los sujetos S1, S2 y S3, la observación reportada previamente en [4] de cómo el entrenamiento mejora los niveles de SSVEPs. Si bien una constante de tiempo mayor podría utilizarse para lograr una mayor selectividad en los filtros, compensando potencialmente esta falta de entrenamiento, esto llevaría a un deterioro de la responsividad del sistema.

En el futuro, este sistema podría mejorarse, por ejemplo, incrementando los comandos asociados con una combinación de codificación en fase y frecuencia de los estímulos [145]. También puede considerarse la migración del sistema hacia una plataforma con microprocesador embebido, como la DE1-SoC estudiada en capítulos anteriores. Esto permitiría añadir una capa de aplicación al sistema, permitiendo cerrar el lazo de control de una interfaz cerebro computadora completamente funcional.

5.3. Espectroscopía Mössbauer

5.3.1. Introducción

El efecto Mössbauer, también conocido como absorción resonante de rayos gamma, es un fenómeno descubierto por el físico alemán Rudolf Ludwig Mössbauer a fines de la década del 50 [182]. Se trata de una técnica de resonancia nuclear que utiliza la emisión libre de retroceso de rayos gamma desde un núcleo radiactivo, seguido por la absorción selectiva de dichos fotones en otro núcleo cercano, para determinar características de los materiales involucrados. El fenómeno es extremadamente sensible a pequeñas diferencias entre la estructura del núcleo emisor y el núcleo absorbente, ya que la absorción depende fuertemente del estado electrónico, químico y magnético de los átomos que componen el material en estudio [183], [184].

Al igual que la nube de electrones del átomo, el núcleo puede existir sólo en ciertos estados de energía definidos. Una fuente radiactiva emite un fotón durante la transición desde un nivel determinado de energía a uno de energía inferior, liberando fotones de alta energía (desde KeV hasta unos pocos MeV) que se conocen como rayos gamma. En el proceso inverso, para que un fotón pueda ser absorbido por un núcleo, su energía debe ser exactamente igual a la diferencia entre dos de dichos estados. Este fenómeno se denomina absorción resonante.

Sin embargo, si un átomo esta libre, el fotón emitido pierde un poco de energía en el proceso de emisión, por lo que este nunca podrá ser absorbido por el núcleo de un átomo similar. En cambio, si el átomo emisor pertenece a una red de un sólido, la energía que pierde en el retroceso es despreciable y el fotón emitido sí puede ser absorbido por el núcleo de un átomo similar. Este fenómeno se conoce como emisión libre de retroceso.

Por otro lado, si se cambia la velocidad relativa entre el núcleo emisor y el núcleo absorbente, la energía de los fotones emitidos experimentan un corrimiento. Este fenómeno es conocido como efecto doppler.

La Espectroscopía Mössbauer se vale de estos fenómenos para caracterizar materiales. En esta técnica se utiliza el mismo tipo de núcleo en dos sólidos, uno de ellos en estado de radiación o excitado y en el otro en estado absorbente o fundamental, de manera que sea posible absorber en el segundo la radiación gamma del primero. El método es tan sensible que si un sólido se mueve con relación al otro con una velocidad tan baja como un milímetro por segundo la energía de la radiación gamma varía por efecto Doppler, y el núcleo receptor no puede absorber el fotón. La absorción resonante también desaparecerá si los estados de energía, tanto de la fuente como del absorbente, son afectados por algún parámetro externo, como por ejemplo ante la aplicación de campo magnético o la variación de la temperatura. En este caso puede variarse la velocidad relativa entre la fuente y el absorbente hasta restablecer la absorción resonante, pudiéndose así cuantificar los efectos de dicho parámetro. En la Figura 5.19 se muestra la disposición clásica de un equipo de Espectroscopía Mössbauer.

Para obtener un espectro utilizando esta técnica, la velocidad se varía en intervalos discretos. Para cada una de las velocidades ensayadas se estudian las mediciones del detector, obteniéndose lo que se denomina canal. Cada canal corresponde a una medida específica de intensidad de radiación absorbida o dispersada en función de la velocidad relativa entre la fuente y el detector. Naturalmente, la relación de canal a velocidad en la adquisición de un espectro de Mössbauer es una característica fundamental para la obtención de espectros de calidad adecuada. Esta relación se obtiene generalmente a partir de un enfoque empírico, configurando parámetros para que un espectro conocido tenga una relación canal a velocidad correcta [42], [43].



Figura 5.19: Disposición del transductor electromecánico, la fuente, el absorbente en estudio y el detector en un espectrómetro Mössbauer típico [184]

La velocidad de la fuente radiactiva suele ser controlada por un servo amplificador, del cual se distinguen dos señales de control: la señal Monitor, que es proporcional a la velocidad requerida en una bobina secundaria y la señal de Error, que es proporcional a la diferencia entre la tensión real en la bobina secundaria y la tensión requerida. La determinación de estas señales permite un control del funcionamiento correcto del espectrómetro, y puede utilizarse para mejorar la determinación de la relación canal a velocidad [44].

De esta forma, en lugar de medir únicamente un espectro de referencia como se hace típicamente [42], [43], las señales de Error y Monitor del controlador de Mössbauer pueden utilizarse para inferir la velocidad verdadera durante la adquisición de un espectro. Esto se puede utilizar para realizar el control de calidad de los espectros, validar regiones de buena linealidad, corregir las no linealidades, caracterizar la calidad de la adquisición del espectro, inferir nuevos intervalos de velocidad para cambios en el rango de velocidad espectral y también como un parámetro de entrada en la calibración de la relación canal a velocidad.

Las desviaciones de la relación canal a velocidad de la esperada son casi imposibles de predecir, y varían considerablemente cuando se modifican los parámetros de la velocidad o el sistema de control de ciclo cerrado, pero se pueden tener en cuenta midiendo la forma de onda del Error. En un trabajo reciente [44], se realizó el muestreo de las formas de onda Error y Monitor con un osciloscopio digital estándar, manteniendo la sincronización y resolución necesarias para un análisis posterior. Aunque los osciloscopios tienen capacidades de promediación coherentes, útiles para la adquisición de estas señales, los requisitos de este experimento exceden las capacidades habituales, por lo que este procedimiento, que ha demostrado su utilidad, resulta una tarea laboriosa y temporalmente demandante.

Para optimizar esta tarea se reconoció la utilidad del sistema de detección coherente en SoC-FPGA diseñado durante esta investigación. Los módulos de promediación coherente implementados resultan ideales para medir las señales de Error y Monitor y su naturaleza reutilizable permite su integración en laboratorios de Espectroscopía Mössbauer como un elemento más de control. Además, el μ P incluido en los chips SoC-FPGA permite una operación intuitiva de las herramientas desarrolladas, permitiendo que investigadores sin conocimientos específicos de lógica programable puedan utilizarlo en sus experimentos y fortaleciendo la colaboración interdisciplinaria en campos



Figura 5.20: Diagrama en bloques del sistema implementado

aparentemente dispares.

Para esta implementación se decidió trabajar con la plataforma Red Pitaya STEM^{LAB}125-14 [109], ya estudiada en capítulos anteriores. Este es un dispositivo con una filosofía de hardware abierto que puede ser reconvertido en varios dispositivos útiles como osciloscopios, analizadores de frecuencia, y más, incluso sin la necesidad de conocimientos específicos de hardware reconfigurable. Este dispositivo se basa en un Zynq-7010 FPGA de AMD-Xilinx y tiene dos entradas y dos salidas analógicas que pueden operar hasta 125 MHz con una resolución de 14 bits. Se eligió este dispositivo por su naturaleza intuitiva, por su filosofía de hardware y software abierto, por sus funcionalidades accesibles incluso sin conocimientos específicos y por su poderoso *front end* analógico.

Las señales de Error y Monitor a medir son de baja frecuencia (alrededor de 20 Hz), lo que hace que la frecuencia de muestreo de la Red Pitaya sea muy superior a la requerida para este experimento. Por consiguiente, la señal puede medirse con una tasa de sobremuestreo que luego puede utilizarse para mejorar la resolución del sistema [113]. Esto, sumado a las técnicas de promediación coherente aplicadas, mejoran sustancialmente la calidad de las medidas en comparación al enfoque de osciloscopio adoptado en [44].

Todo el código desarrollado para esta investigación es abierto, con licencia bajo los términos de la licencia MIT, y está disponible en [185]. La Red Pitaya también es de hardware abierto, con múltiples recursos disponibles en [186], por lo que este sistema puede ser replicado por otros investigadores que trabajen en Espectroscopía Mössbauer. Los resultados de esta sección han sido presentados en [187].

5.3.2. Implementación del sistema

Como se ha estudiado en secciones anteriores, la Red Pitaya tiene dos elementos de procesamiento: una FPGA y un microprocesador (μ P). En esta implementación, la FPGA es responsable de la adquisición y procesamiento en tiempo real de las señales de Monitor y Error, mientras que el μ P implementa el control del sistema e interacción con el usuario. La figura 5.20 muestra un diagrama de bloques simplificado del sistema. Las señales de Error y Monitor están disponibles en la mayoría de los controladores (*drivers*) de Espectroscopía Mössbauer, como es el caso del Wissel MR-360 utilizado en este experimento, y la mayor parte de los generadores de funciones proporcionan la señal **start**, que es útil para la sincronización del sistema. La FPGA se ocupa de la adquisición y procesamiento de la señal, y el μ P se ocupa del control del sistema, por lo que no se necesita hardware adicional para esta aplicación, aparte de una PC anfitriona (*host*) que se comunica con la Red Pitaya a través de un cable Ethernet, y se utiliza para iniciar el proceso y visualizar los resultados.

5.3.2.1. Procesamiento en la FPGA

La FPGA recibe una muestra de datos de cada uno de los dos canales ADC en cada flanco ascendente de un reloj de 125 MHz. Luego los datos entran en dos etapas de procesamiento: una etapa de promedio lineal y una etapa de promediación coherente.

Como se indicó anteriormente, las señales de Monitor y Error son señales periódicas de baja frecuencia (alrededor de 20 Hz), y la frecuencia de muestreo del ADC de la Red Pitaya esta dada por su reloj de 125 MHz. La FPGA tiene memoria de acceso aleatorio en bloques (block-RAM) disponible para almacenar aproximadamente 60000 muestras de 32 bits, por lo que si se adquiriera la señal directamente a 125 MHz, la memoria no sería suficiente para un solo período de la señal. Por otro lado, utilizar el ADC a una frecuencia más baja es un desperdicio de la capacidad de muestreo del sistema. Una solución superadora es realizar un sobremuestreo de K muestras de la señal (como se ha mostrado en la sección 3.5.2), reduciendo la frecuencia de muestreo a 125 MHz/K. Esto se puede lograr promediando K muestras consecutivas, siendo este el propósito de la etapa de promediado lineal de la cadena de procesamiento.

La segunda etapa aprovecha la periodicidad de la señal para realizar un promedio coherente de N_{ca} ciclos, reduciendo el ruido de la señal, como se mostró en la sección 2.3 [85]. Para realizar correctamente este promedio coherente, se debe conocer precisamente el punto de inicio de cada período [86]. Para esto, se utilizó la señal de salida **start** del generador de funciones utilizado en la espectroscopía.

Una vez que la señal entrante es promediada lineal y coherentemente, esta se almacena en dos memorias block-RAM, cada una con capacidad para 30000 muestras de 32 bits. Estas memorias son accesibles desde el μ P de Red Pitaya. Todo este flujo de procesamiento se puede ver en la Fig. 5.21. El sistema fue diseñado para ser reutilizable en distintos experimentos, por lo que tanto N_{ca} como K son controlados por el μ P y pueden ser configurados por el investigador para cada experimento.

5.3.2.2. Promedio Lineal

La primera etapa de procesamiento promedia K muestras consecutivas. Dado que la FPGA funciona mejor con aritmética de punto fijo, la división por K no se realiza en la FPGA, sino más tarde en el lado del μ P, por lo que el promedio es realmente una acumulación, como se muestra en la ecuación 5.9, donde s[n] representa las muestras entrantes del ADC. Con esta operación, se preserva toda la resolución en la señal, pero si K es grande, el crecimiento de bits en la salida podría representar un problema.

$$y_{[n]} = \sum_{i=0}^{K-1} s_{[i+nK]}$$
(5.9)



Figura 5.21: Procesamiento en la FPGA

Este sobremuestreo tiene la propiedad de reducir el ruido RMS en un factor aproximadamente igual a la raíz cuadrada de la relación de sobremuestreo. Por lo tanto, cada duplicación de la frecuencia de muestreo disminuye el ruido en banda por 3 dB, aumentando la resolución en medio bit [113].

Para este experimento, se calculó K para que se ajustara a un solo período de las señales de Error y Monitor en las 30000 muestras de la memoria block-RAM. Considerando una señal de 20 Hz, esto significa muestrear a una frecuencia de 600 kHz, obteniendo una relación de sobremuestreo de $K = \frac{125 \text{ MHz}}{600 \text{ kHz}} \approx 208$. Esto significa que esta etapa aumenta la resolución de la medida aproximadamente en $\frac{\log_2(K)}{2} \approx 4$ bits.

5.3.2.3. Promediación Coherente

Para reducir aún más el ruido, la señal promediada entra a una etapa de promedio coherente de N_{ca} ciclos, aprovechando su periodicidad y la certeza del punto de inicio de cada período, dado por la señal **start** de la unidad generadora de funciones digitales. La ecuación 5.10 muestra la operación sobre las muestras entrantes donde, al igual que en el caso lineal, el promedio es realmente una acumulación. En esta ecuación, M representa el número de muestras en un período de la señal, e Y_2 representa la señal proveniente del promedio lineal y una etapa de división parcial que se describirá más adelante, en la sección 5.3.2.4. La división por N_{ca} se realiza posteriormente en el μ P.

$$y_{ca[n]} = \sum_{i=0}^{N_{ca}-1} Y_2[n+iM]$$
(5.10)

Como se analizó en el capítulo 2, si una señal perturbada con ruido blanco de desviación estándar σ_i es promediada coherentemente durante N_{ca} ciclos, la desviación estándar de la amplitud en la salida (σ_o) depende del número de puntos que se están promediando y de (σ_i) a través de la ecuación 5.11.

$$\sigma_o = \frac{\sigma_i}{\sqrt{\frac{MN_{ca}}{2}}} \tag{5.11}$$

De esta ecuación, es claro que para rechazar el ruido blanco N_{ca} debe ser alto, lo que se traduce en tiempos de promediado más largos. Idealmente, si se pudiera promediar un número infinito de ciclos, todo el ruido de la señal sería rechazado por este procesamiento. Sin embargo, en tiempos de promediado largos podrían ocurrir variaciones lentas en la señal que afectarían el resultado promediado final. Debido a esto, N_{ca} se deja programable, lo que permite adaptarlo a la dinámica específica de cada experimento.

5.3.2.4. Límites de K y N_{ca}

Dado que los cálculos no realizan la división en la FPGA y las memorias block-RAM tienen capacidad para muestras de 32 bits, el crecimiento de bits puede ser un problema si K y N_{ca} son lo suficientemente grandes. Considerando los escenarios más desfavorables (muestras entrantes iguales a 2^{14}), esta condición significa que la ecuación 5.12 debe cumplirse.

$$\log_2(N_{ca}) + \log_2(K) + 14 < 32; \tag{5.12}$$

Para K = 208, esto significa que $N_{ca} \leq 1260$, un tiempo de integración de aproximadamente 63 segundos para una señal de 20 Hz. Para la mayoría de los propósitos, este tiempo de promediado sería suficiente.

Si se requieren tiempos de promediado más largos, el μ P puede establecer un parámetro adicional llamado DIV, que realiza una división entera parcial después del promedio lineal, como se indica en la ecuación 5.13. Esta división limita el crecimiento de bits del sistema sin perder resolución innecesaria. El resto de la división necesaria para el promedio (división por $\frac{\text{DIV}}{N_{ca}K}$) se realiza más tarde en el lado del μ P.

$$y_{2[n]} = \frac{y_{[n]}}{\text{DIV}} \tag{5.13}$$

Con este parámetro, las limitaciones en N_{ca} y K se vuelven las que se muestran en la ecuación 5.14.

$$\log_2(N_{ca}) + \log_2(K) + 14 - \log_2(\text{DIV}) < 32; \tag{5.14}$$

5.3.2.5. Control de Disparo

Conocer el punto de inicio de cada período de la señal es crucial para la operación de promediación coherente. Esta funcionalidad es proporcionada por la señal start del generador de funciones utilizado en el experimento, que es digitalizada utilizando un pin de entrada-salida de propósito general (GPIO) de la Red Pitaya, configurado con una resistencia de *pull-down* para evitar disparos accidentales.

El promedio coherente se realiza desde una señal start hasta la siguiente. Luego se espera a la próxima señal de start, ignorando el ciclo de señal intermedio. En esta configuración, el tiempo para realizar los cálculos es de $2N_{ca}$ veces el período de la señal (T_{signal}) debido a los ciclos intermedios que se ignoran cuando el procesamiento está esperando la señal para sincronizarse. El sistema también incluye opciones para activar el promedio después de que se reciben M muestras o cuando la señal excede un umbral (como un osciloscopio clásico). Estos tipos de disparo, configurables desde el μ P de la Red Pitaya, son problemáticos para esta aplicación pero pueden ser útiles en otros experimentos.

5.3.2.6. Control del μP

Para controlar la adquisición, el promediado y la visualización de los resultados, se desarrolló un programa simple en C que se ejecuta en el μ P de la Red Pitaya. Este programa primero configura el bitstream en el FPGA, que reconfigura el hardware como el sistema de adquisición y procesamiento descripto en las secciones anteriores. Luego configura los parámetros K, N_{ca} y DIV, establece las opciones para el disparo y luego habilita la adquisición. Cuando la adquisición está lista, después de $2N_{ca}T_{signal}$, la FPGA señaliza al μ P, el programa C lee las memorias block-RAM, realiza la división por $\frac{\text{DIV}}{N_{ca}K}$ y guarda los resultados en archivos de texto. Toda esta operación puede ser gestionada utilizando scripts en bash desde una computadora host conectada a través de Ethernet a la Red Pitaya. Esto permite al usuario incluir este sistema de adquisición y procesamiento de señales en un flujo de trabajo habitual de Python o Matlab. La Figura 5.22 ilustra este flujo de trabajo.



Figura 5.22: Sistema de control en la CPU

Los archivos de texto que deben ser transferidos entre el μ P de la Red Pitaya y la computadora *host* almacenan la información de dos canales de datos, cada uno con 30000 muestras de 32 bits. Incluyendo un bit adicional por muestra para una coma que actúa como delimitador y un encabezado de 100 bytes, esto significa que aproximadamente se deben transferir 2400 kbits (300 kBytes) de datos por adquisición. Las redes Ethernet modernas funcionan a 100 Mbits/s, e incluso las más antiguas funcionan a 10 Mbits/s, por lo que el volumen de datos de las mediciones es fácilmente manejable para esta tecnología de red.

5.3.3. Resultados

5.3.3.1. Utilización de recursos

El sistema fue implementado en la Red Pitaya. Todos los módulos cumplen con los requisitos de temporización con el reloj de 125 MHz. La utilización de recursos en la FPGA se muestra en el Cuadro 5.6. De esta tabla, se desprende que hay espacio para etapas adicionales de procesamiento en la señal. La limitación principal es la memoria block-RAM, que fue casi completamente utilizada.

LUTs de Slice	Registros de Slice	Multiplexores F7	Slice	LUT como lógica
(17600)	(35200)	(8800)	(4499)	(17600)
3368	5576	54	1556	3303
LUT como memoria	Block-RAM Tile	DSPs	Bonded IOB	Bonded IOPADs
(6000)	(60)	(80)	(100)	(130)
65	58.5	3	81	130
IBUFDS	ILOGIC	OLOGIC	BUFCTRL	MMCME2_ADV
(95)	(100)	(100)	(32)	(2)
3	28	18	6	1

Cuadro 5.6: Utilización de recursos

5.3.3.2. Validación del sistema

La adquisición fue comparada con la correspondiente obtenida utilizando el mismo procedimiento mostrado en [44]. Se utilizó un osciloscopio Tektronix TDS 3012 para esta adquisición. Como era de esperar, los ADCs de Red Pitaya tienen errores de offset y de ganancia, que, debido a la distribución natural de las características eléctricas de las entradas y salidas analógicas, difieren ligeramente entre varias placas y pueden cambiar con el tiempo. Estos errores fueron estimados durante esta comparación, obteniendo un error de ganancia de aproximadamente 1,131 y un error de offset de aproximadamente 0,015 V, muy similares a los reportados por el fabricante en [188]. La Figura 5.23 muestra la digitalización de las señales Monitor y Error utilizando la metodología del trabajo anterior y esta nueva propuesta. La nueva adquisición es mucho más rápida que la anterior y tiene una resolución comparable, aunque hay evidencia de algunas pequeñas no linealidades que deberían corregirse en el futuro.

5.3.4. Conclusiones

En este capítulo se ha mostrado cómo los sistemas de detección coherente de señales desarrollados se pueden aplicar a la instrumentación de un equipo de Espectroscopía Mossbaüer. Nuevamente se ha verificado que desarrollar sistemas en hardware programable permite la incorporación de estos sistemas a prototipos funcionales y no simplemente como un sistema de medida tradicional.

En esta implementación, la plataforma utilizada ha sido una Red Pitaya STEM^{LAB}125-14, desarrollada por AMD-Xilinx. Las características específicas de este dispositivo han abierto nuevas posibilidades de trabajo interdisciplinar, ya que permiten separar el procesamiento en tiempo real de las señales, que tiene lugar en la FPGA embebida en la Red Pitaya, del análisis y la experimentación en laboratorio, que puede ejecutarse en el microprocesador del dispositivo o en la computadora *host*. Esto permite separar las tareas de diseño digital de las tareas de laboratorio, permitiendo que diseñadores digitales e investigadores en Espectroscopía Mossbaüer (u otras ramas de la física) trabajen conjuntamente. Todo el código desarrollado durante este trabajo es de libre acceso y disponible para la comunidad científica en [185].

La digitalización de las señales Monitor y Error se llevó a cabo con éxito utilizando el hardware de la Red Pitaya. El resultado es un método eficiente que utiliza un dispositivo estándar, disponible y de bajo costo con una filosofía de hardware abierto que lo hace ideal para su incorporación en el laboratorio como un elemento de control más. En este sentido se desarrollaron herramientas de


Figura 5.23: Validación del sistema. a) Datos sin procesar obtenidos de Red Pitaya y del método anterior [44]. b) Corrección de errores de desplazamiento y ganancia. c) y d) Ampliación en partes de las señales de Error que muestran algunas evidencias de no linealidades.

hardware específicas de la aplicación y herramientas de software para controlarlas fácilmente.

La naturaleza reconfigurable y paralela del dispositivo explorado abre diversas oportunidades para la instrumentación de la Espectroscopía Mössbauer. En el futuro cercano, se pretende mejorar el sistema mediante la incorporación de etapas de procesamiento adicionales para capturar la dispersión de la señal a lo largo de ciclos coherentemente promediados. Esto permitirá cuantificar la periodicidad de la onda, eliminando así datos de registros que exhiben altas dispersiones causadas por fuentes de ruido aisladas. Además, se espera aprovechar las capacidades Ethernet de la Red Pitaya para desarrollar software de medición que pueda ejecutarse convenientemente desde navegadores web. En una perspectiva a más largo plazo, se avanzará en la adquisición en tiempo real de los eventos del detector del espectrómetro, con su correspondiente marca de tiempo y energía, de manera tal de poder discriminar entre eventos independientes sin la pérdida de información que ocurre cuando se calcula el histograma en tiempo real. Se espera que este desarrollo, posibilitado por la incorporación de sistemas SoC-FPGA en laboratorios de Espectroscopía Mössbauer, permita innovadoras soluciones en esta técnica.

Capítulo 6

Conclusiones, aportes originales y líneas abiertas de trabajo

6.1. Conclusiones generales

A lo largo de esta tesis se han estudiado en detalle la teoría y la implementación de técnicas de detección coherente de señales, abordando su integración en dispositivos SoC-FPGA y su aplicación a diversos sistemas de interés.

Los aspectos teóricos de estas técnicas fueron analizados en detalle en el capítulo 2. Estas permiten determinar la amplitud de señales de frecuencias conocidas, incluso en ambientes con muy baja relación señal a ruido. La forma más conocida en la que se aplica el concepto de detección coherente es el amplificador *Lock-in*, cuyo funcionamiento de basa en la multiplicación de la señal de interés por una señal de referencia periódica, de la frecuencia que se pretende detectar, seguida de un filtro pasa-bajos. En este esquema, el mezclador genera una señal compuesta por una componente en banda base, que contiene la información deseada, y otra componente al doble de la frecuencia de operación, que es eliminada por un filtro pasa bajos. La respuesta de este filtro es clave para el funcionamiento del *Lock-in*. A lo largo de esta investigación se ha puesto especial énfasis en el filtro de media móvil, por su simplicidad y su propiedad de rechazar completamente, en ciertas circunstancias, la componente del doble de frecuencia proveniente del mezclador. Para este filtro se ha determinado que, en presencia de ruido blanco en la señal de interés, la componente de ruido a la salida se reduce en un factor de $\sqrt{\frac{N}{2}}$, donde N es la cantidad de muestras de señal que se promedian en el filtro. También se han analizado otras fuentes de ruido, como el ruido de cuantización, llegando a la conclusión de que es, en definitiva, la componente de la señal error de cuantización que está en la frecuencia objetivo la que limita la mejora de la SNR del sistema.

Otra técnica muy utilizada para el procesamiento coherente de señales es la promediación coherente, que consiste en promediar las muestras de la señal, teniendo en cuenta la posición relativa que ocupan en cada periodo. Durante el desarrollo de la investigación se ha determinado que la combinación del promedio coherente de N_{ca} ciclos de una señal de interés y un sistema de *Lock-in* convencional con filtro de media móvil de N_{ma} ciclos (sistema CALI) produce los mismos resultados que un *Lock-in* de N ciclos de la señal (sistema LI), siempre y cuando $N = N_{ca}N_{ma}$. Aunque ambos métodos proporcionan los mismos resultados, los cálculos necesarios para obtenerlos son bastante diferentes. Específicamente, el número de multiplicaciones necesarias se reduce a medida que N_{ca} aumenta (y N_{ma} disminuye). El caso más eficiente se logra cuando $N_{ca} = N$ y $N_{ma} = 1$, donde el número de multiplicaciones requeridas se fija en la cantidad de muestras en un único ciclo de la señal (M), independientemente del N seleccionado.

La alta frecuencia de operación de las aplicaciones planteadas, las necesidades de sincronización de los algoritmos de detección coherente y las posibilidades de escalabilidad de los sistemas desarrollados dificultan su aplicación en dispositivos tradicionales. Es por ello que se decidió que la implementación de los sistemas se haga en dispositivos SoC-FPGA, que incorporan un arreglo de celdas lógicas programables (FPGA) junto con un microprocesador. Estos dispositivos resultan muy potentes, y tienen características ideales para las aplicaciones planteadas, pero presentan como desventaja una curva de aprendizaje muy lenta, que requiere de diseñadores especializados en el diseño digital y en la arquitectura específica de cada sistema. En este sentido se han hecho esfuerzos por automatizar en lo posible el flujo de trabajo, dando lugar a sistemas de procesamiento de señal reutilizables, de código abierto, que fueron probados en dispositivos de Intel-Altera y AMD-Xilinx, los dos principales fabricantes del medio, y con conceptos básicos que son transferibles a las arquitecturas de otros fabricantes. El capítulo 3 cubre estos sistemas reutilizables, incluyendo su diseño y las pruebas preliminares para verificar su funcionamiento.

Utilizando este sistema de procesamiento de señales se diseñaron dos dispositivos Lock-in: el DE-li, implementado en una plataforma DE1-SoC, desarrollada por Terasic y basada en dispositivos SoC-FPGA de Intel-Altera, y el RP-li, implementado en una plataforma Red Pitaya STEM^{LAB}125-14 basada en dispositivos de AMD-Xilinx. Estos desarrollos se presentan en el capítulo 4, que cubre el diseño de cada sistema, las pruebas que se hicieron para determinar sus características principales y una comparación del desempeño de cada uno en distintos escenarios de operación. En este capítulo, además, se retoma el tema de la equivalencia entre el algoritmo CALI y el algoritmo LI, verificando cómo la implementación del sistema CALI, con su respectiva reducción en las multiplicaciones necesarias, puede servir para incrementar la frecuencia de operación de los sistemas en la implementación en el contexto de las FPGAs. Esto es posible gracias a que el algoritmo CALI traslada el cuello de botella de la implementación en tiempo real desde una multiplicación hasta una simple adición. Se ha verificado que los sistemas diseñados tienen prestaciones similares a algunos *Lock-in* disponibles comercialmente, como el SR865, por una fracción del precio. Además, su naturaleza reconfigurable permite adaptar los sistemas a cada aplicación en particular, optimizando el diseño en cada caso.

Finalmente, en el capítulo 5 se aplican los dispositivos desarrollados a tres aplicaciones distintas: La tomografía magnética para la detección de nanopartículas ferromagnéticas en técnicas de terapias oncológicas, las interfaces cerebro computadora y la calibración de equipos de Espectroscopia Mössbauer. Como primera conclusión al respecto se destaca la flexibilidad de los sistemas desarrollados, que permite su incorporación en tres áreas aparentemente dispares.

En el ámbito de la Tomografía Magnética, se ha logrado incorporar a un modelo previamente desarrollado la detección coherente de señales de 100 kHz utilizando el DE-li, en reemplazo de un *Lock-in* comercial de mayor costo y tamaño. Esta innovación contribuye significativamente a la automatización del proceso, reduciendo drásticamente los tiempos de medida e independizándolo de un dispositivo costoso y poco configurable. Además, la especificidad de la aplicación permite configurar el DE-li para las necesidades particulares, obteniendo medidas de mayor resolución con un dispositivo de menor costo. Utilizando el sistema diseñado y algoritmos de reconstrucción diseñados oportunamente se ha logrado determinar la posición de un *phantom*, un objeto artificial que se utiliza para calibrar, probar y evaluar el rendimiento de los sistemas de imagen, en distintas posiciones dentro del área de medidas.

Por otro lado, en el área de las interfaces cerebro computadora, los sistemas desarrollados han sido reconfigurados para implementar un detector de potenciales evocados visuales de estado estable (conocidos como SSVEPs) de varios canales, fácilmente escalable y basado en un *Lock-in* multifrecuencia. En esta aplicación, destaca cómo el sistema de detección coherente se integra como un elemento funcional dentro del sistema completo, en lugar de ser utilizado únicamente como un aparato de medida, como es común con los *Lock-in* en un laboratorio tradicional. Los experimentos llevados a cabo muestran la capacidad del sistema para adquirir la señal de electroencefalografía, detectar en tiempo real la presencia de SSVEPs mediante *Lock-in* con filtro de media móvil de 2 s y transmitir los datos a una PC.

En el contexto de la Espectroscopía Mössbauer se han utilizado los dispositivos desarrollados para medir señales de Monitor y Error de un controlador de velocidad y así inferir la velocidad verdadera durante la adquisición de un espectro. Para este desarrollo se partió de la plataforma Red Pitaya, adaptando el RP-li para lograr la adquisición de las señales promediadas coherentemente, lo que permite una calibración más precisa del instrumental involucrado en esta técnica y, en consecuencia, la obtención de espectros con mayor exactitud. La integración de la Red Pitaya a un laboratorio de Espectroscopía Mössbauer abre además nuevas posibilidades de trabajo interdisciplinar, ya que este dispositivo separa la adquisición y procesamiento en tiempo real de la señal del análisis y interacción con el experimento en sí. De esta manera, un diseñador especializado en diseño digital puede desarrollar el procesamiento específico requerido por el experimento, dejándole la experimentación al investigador especializado.

6.2. Líneas abiertas de investigación

146

Si bien este trabajo de investigación ha alcanzado un nivel avanzado que permite la presentación de resultados y el uso cotidiano de los dispositivos involucrados, aún quedan numerosos aspectos teóricos por desarrollar y diversas áreas por mejorar, tanto en los instrumentos implementados como en las aplicaciones desarrolladas.

Desde punto de vista de la teoría es necesario un análisis más profundo de los efectos del ruido de cuantización en distintos escenarios. Durante esta tesis se ha determinado que en definitiva es la componente de este error que está en la misma frecuencia que la señal de interés la que limita la mejora de la SNR que se puede lograr al aplicar técnicas de detección coherente. Sin embargo, esta componente depende de múltiples factores, como la relación entre la frecuencia de muestreo y la frecuencia objetivo, la amplitud de la señal medida y la tensión de fondo de escala del ADC. Si bien se ha logrado acotar este efecto, la cota obtenida es muy conservadora y, tanto en las pruebas prácticas como en las simulaciones, las contribuciones reales de este error resultan significativamente menores. Es deseable en un futuro acotar en mayor medida esta fuente de error, determinando las condiciones para minimizarlo y mejorando así el sistema de medida.

Otro aspecto interesante para explorar en este sentido es la utilización de otros tipos de filtros pasa-bajo en el *Lock-in*, posiblemente utilizando una combinación de filtros de media móvil con filtros IIR. Esto garantizará la eliminación del componente del doble de frecuencia generada por el mezclador gracias al filtro MAF y permitirá reducir el ruido de forma, probablemente, más eficiente con el filtro IIR. Un modelo teórico que cubra todos estos aspectos sería útil para determinar los parámetros de estos filtros antes de una posible implementación.

En las implementaciones, tanto el RP-li como el DE-li tienen aspectos a mejorar. El DE-li se encuentra en un estado más avanzado de desarrollo, con una interfaz Web que permite gestionar el *Lock-in* sin conocimientos específicos del hardware ni software de base. Esta capa de abstracción falta en el RP-li, que actualmente debe utilizarse mediante rutinas de C o Python. En cuanto al hardware, podrían agregarse funcionalidades para implementar *Lock-in* con otros tipos de filtros, aparte del MAF, y en el caso del DE-li sería deseable rediseñar la etapa de acondicionamiento analógico del conversor *Highspeed AD/DA card* para cubrir todo el rango de frecuencias que ofrece (DC-65 MHz). Para ello, debería considerarse reducir la frecuencia de corte del filtro anti-aliasing y cambiar los amplificadores operacionales por otros de mayor ancho de banda, como se detalló en la sección 3.2. Además, un correcto encapsulamiento de la plataforma DE-li facilitaría su portabilidad, así como una adaptación para uso con fuente de energía inalámbrica. Otras limitaciones incluyen la imposibilidad de trabajar con una fuente externa como referencia y, en el caso del RP-li, la de operar con señales diferenciales. Estos son algunos de los aspectos que se planean explorar en futuras implementaciones.

En las líneas de aplicaciones específicas también se prevee gran trabajo a futuro. En el caso de la Tomografía Magnética el principal aspecto a mejorar es la permutación de las espiras, que actualmente debe hacerse en forma manual. En este sentido, se buscará reemplazar el conmutador mecánico por un multiplexor electrónico de señales analógicas, controlado directamente por la plataforma SoC-FPGA del DE-li y automatizar la rotación del conjunto de bobinas utilizando una configuración mecánica simple. Además, resta repetir las mediciones *In Vivo* de nanopartículas ferromagnéticas inyectadas en pequeños animales reportadas en [23], pero ahora con el nuevo dispositivo. Una vez verificada la operación el sistema ya se encontrará en un estado funcional para su aplicación en estudios de Hipertermia Magnética.

En el campo de las interfaces cerebro computadora sería deseable incrementar los comandos asociados al dispositivo, por ejemplo con una combinación de codificación en fase y frecuencia de los estímulos. La modularidad del *Lock-in* desarrollado permite agregar en un único dispositivo muchos detectores, por lo que podría implementarse la detección conjunta de tantos estímulos como sean posibles con las celdas lógicas y bloques DSP del hardware disponible. Asimismo, sería útil migrar la implementación hacia una plataforma con microprocesador embebido, como la DE1-SoC. Esto permitiría añadir una capa de aplicación al sistema, permitiendo cerrar el lazo de control de una interfaz cerebro computadora completamente funcional.

Con respecto a la Espectroscopía Mössbauer se planea agregar etapas adicionales al procesamiento para capturar la dispersión de la señal a lo largo de ciclos coherentemente promediados, para poder eliminar registros de altas dispersiones causadas por fuentes de ruido aisladas. Además se pretende aprovechar las capacidades Ethernet de la Red Pitaya para generar una interfaz de usuario más intuitiva, ejecutada a través de un servidor Web, en la misma línea del desarrollo llevado a cabo para el DE-li. En una perspectiva a más largo plazo, se avanzará en la adquisición en tiempo real de los eventos del detector del espectrómetro, con su correspondiente marca de tiempo y energía, de manera tal de poder discriminar entre eventos independientes sin la pérdida de información que ocurre cuando se calcula el histograma en tiempo real.

Finalmente, es de destacar que las funcionalidades desarrolladas para la Espectroscopía Mössbauer han encontrado nuevas aplicaciones en el campo del descongelamiento de tejidos criopreservados. Un desarrollo reciente en el Instituto de Física de La Plata [189] emplea promediación coherente mediante un osciloscopio para la detección de señales de radiofrecuencia, en un enfoque similar al implementado en [44]. En el corto plazo, se planea reemplazar esta técnica basada en osciloscopio por el dispositivo desarrollado sobre la plataforma Red Pitaya. Se prevé que, con adaptaciones mínimas, el dispositivo pueda aplicarse a este nuevo ámbito, aparentemente disímil, reafirmando la flexibilidad y la versatilidad de los sistemas desarrollados.

Bibliografía

- W C Michels y N L Curtis. «A pentode lock-in amplifier of high frequency selectivity». En: Review of Scientific Instruments 12 (1941), pág. 444.
- [2] Stanford Research Systems. Lock-In Amplifiers. 2024. URL: https://www.thinksrs.com/ products/lockin.html.
- [3] Zurich Instruments. Lock-In Amplifiers. 2024. URL: https://www.zhinst.com/americas/ en/lock-in-amplifiers.
- [4] Pablo Andrés García. «Sistemas embebidos de tiempo real con aplicaciones en bioingeniería». Tesis doct. La Plata, Buenos Aires, Argentina: Universidad Nacional de La Plata, 2019.
- [5] C. J. Busch. «Einfluss heftiger Erysipeln auf organisierte Neubildungen». En: Verhandlungen Des Naturhistorischen Vereins Der Preussischen Rheinlande und Westphalens (1866). Ed. por C. J. Andrä, págs. 28-33.
- [6] M. H. Falk y R. D. Issels. «Hyperthermia in oncology». En: International Journal of Hyperthermia 17.1 (ene. de 2001), págs. 1-18. DOI: 10.1080/02656730150201552.
- [7] P. Wust et al. «Hyperthermia in combined treatment of cancer». En: Lancet Oncol. 3 (2002), págs. 487-97. DOI: 10.1016/S1470-2045(02)00818-5.
- [8] J. W. Hand y Gail ter Haar. «Heating techniques in hyperthermia». En: The British Journal of Radiology 54.642 (1981). PMID: 7016234, págs. 443-466. DOI: 10.1259/0007-1285-54-642-443. eprint: https://doi.org/10.1259/0007-1285-54-642-443.
- [9] IA Brezovich y JH Young. «Hyperthermia with implanted electrodes». En: Med Phys 8.1 (ene. de 1981), págs. 79-84. DOI: 10.1118/1.594911.
- John W Strohbehn et al. «An invasive microwave antenna for locally-induced hyperthermia for cancer therapy». En: Journal of Microwave Power 14.4 (dic. de 1979), págs. 339-350.
 DOI: 10.1080/16070658.1979.11689169.
- [11] L.S. Taylor. «Implantable radiators for cancer therapy by microwave hyperthermia». En: Proceedings of the IEEE 68.1 (1980), págs. 142-149. DOI: 10.1109/PROC.1980.11594.
- [12] André Roggan et al. «Medizinphysik: Diagnostik und Therapie mit dem Laser». En: Physikalische Blätter 55.3 (1999), págs. 25-30. DOI: https://doi.org/10.1002/phbl. 19990550307.
- H I Robins et al. «Phase I clinical trial of melphalan and 41.8 degrees C whole-body hyperthermia in cancer patients.» En: *Journal of Clinical Oncology* 15.1 (1997). PMID: 8996137, págs. 158-164. DOI: 10.1200/JC0.1997.15.1.158.

- [14] Rudolf Hergt y Silvio Dutz. «Magnetic particle hyperthermia—biophysical limitations of a visionary tumour therapy». En: Journal of Magnetism and Magnetic Materials 311.1 (2007). Proceedings of the Sixth International Conference on the Scientific and Clinical Applications of Magnetic Carriers, págs. 187-192. ISSN: 0304-8853. DOI: https://doi.org/10.1016/j.jmmm.2006.10.1156.
- [15] RK Gilchrist et al. «Selective inductive heating of lymph nodes». En: Ann Surg 146.4 (oct. de 1957), págs. 596-606. DOI: 10.1097/00000658-195710000-00007.
- [16] Klaus Maier-Hauff et al. «Efficacy and safety of intratumoral thermotherapy using magnetic iron-oxide nanoparticles combined with external beam radiotherapy on patients with recurrent glioblastoma mutiforme». En: Journal of neuro-oncology 103 (jun. de 2011), págs. 317-24. DOI: 10.1007/s11060-010-0389-0.
- Peter Wust et al. «Magnetic nanoparticles for interstitial thermotherapy feasibility, tolerance and achieved temperatures». En: International Journal of Hyperthermia 22.8 (2006).
 PMID: 17390997, págs. 673-685. DOI: 10.1080/02656730601106037. eprint: https://doi. org/10.1080/02656730601106037.
- [18] M Johannsen et al. «Thermotherapy of prostate cancer using magnetic nanoparticles: feasibility, imaging, and three-dimensional temperature distribution». En: European Urology 52.6 (dic. de 2007), págs. 1653-1661. DOI: 10.1016/j.eururo.2006.11.023. eprint: 17125906.
- [19] M Johannsen et al. «Magnetic nanoparticle hyperthermia for prostate cancer». En: International Journal of Hyperthermia 26.8 (2010). Epub 2010 Jul 23, págs. 790-795. DOI: 10.3109/02656731003745740. eprint: 20653418.
- [20] B Thiesen y A Jordan. «Clinical applications of magnetic nanoparticles for hyperthermia». En: Int J Hyperthermia 24.6 (sep. de 2008), págs. 467-474. DOI: 10.1080 / 02656730802104757.
- [21] Jelena Kolosnjaj-Tabi et al. «Biotransformations of magnetic nanoparticles in the body». En: Nano Today 11.3 (2016), págs. 280-284. ISSN: 1748-0132. DOI: https://doi.org/10.1016/j.nantod.2015.10.001.
- [22] Qiyi Feng et al. «Uptake, distribution, clearance, and toxicity of iron oxide nanoparticles with different sizes and coatings». En: Scientific Reports 8 (feb. de 2018). DOI: 10.1038/s41598-018-19628-z.
- [23] A. L. Veiga et al. «Imaging of Magnetic Nanoparticles With Permeability Tomography». En: *IEEE Transactions on Biomedical Engineering* 70.11 (2023), págs. 3216-3226. DOI: 10. 1109/TBME.2023.3283787.
- [24] Diccionario de la Real Academia Española. Online. 2014.
- [25] A.J. Peyton et al. «An overview of electromagnetic inductance tomography: Description of three different systems». En: *Measurement Science and Technology* 7 (mar. de 1996), pág. 261. DOI: 10.1088/0957-0233/7/3/006.
- [26] A D Seagar, D C Barber y B H Brown. «Electrical impedance imaging». En: IEE Proc. A 134 (1987), págs. 201-10.
- [27] Khusnul Ain, Ricky Wibowo y Soegianto Soelistiono. «Modeling of electrical impedance tomography to detect breast cancer by finite volume methods». En: Journal of Physics: Conference Series 853 (mayo de 2017), pág. 012001. DOI: 10.1088/1742-6596/853/1/012001.

- [28] Qilong Deng et al. «A Parallel Impedance Measurement System for Electrical Impedance Tomography System with Multi - Microcontroller - Unit Architecture». En: 2018 IEEE International Conference on Manipulation, Manufacturing and Measurement on the Nanoscale (3M-NANO). 2018, págs. 397-401. DOI: 10.1109/3M-NAND.2018.8552230.
- [29] Songming Huang et al. «Design of Sensor Electronics for Electrical Capacitance Tomography». En: Circuits, Devices and Systems, IEE Proceedings G 139 (mar. de 1992), págs. 83-88. DOI: 10.1049/ip-g-2.1992.0014.
- [30] Ziqiang Cui, Zihan Xia y Huaxiang Wang. «Electrical Capacitance Tomography Sensor Using Internal Electrodes». En: *IEEE Sensors Journal* 20.6 (2020), págs. 3207-3216. DOI: 10.1109/JSEN.2019.2954736.
- [31] Ziqiang Cui et al. «Further Development in Differential Electrical Capacitance Tomography».
 En: *IEEE Sensors Journal* 18.23 (2018), págs. 9781-9791. DOI: 10.1109/JSEN.2018.
 2873399.
- [32] Hermann Scharfetter, Helmut K Lackner y Javier Rosell. «Magnetic Induction Tomography: Hardware for Multi-Frequency Measurements in Biological Tissues». En: *Physiological Measurement* 22.1 (2001). Published under licence by IOP Publishing Ltd, pág. 131. DOI: 10. 1088/0967-3334/22/1/317.
- [33] H. Griffiths, W. R. Stewart y W. Gough. «Magnetic Induction Tomography: A Measuring System for Biological Tissues». En: Ann. NY Acad. Sci. 873 (1999), págs. 335-345. DOI: 10.1111/j.1749-6632.1999.tb09440.x.
- [34] E. Mark Haacke et al. «Quantitative susceptibility mapping: current status and future directions». En: Magnetic Resonance Imaging 33.1 (2015), págs. 1-25. ISSN: 0730-725X. DOI: https://doi.org/10.1016/j.mri.2014.09.004.
- [35] Marcela Fernández van Raap et al. «Anticipating hyperthermia efficiency of magnetic colloids by a semi-empirical model: a tool to help medical decisions». En: *Phys. Chem. Chem. Phys.* 19 (feb. de 2017). DOI: 10.1039/C6CP08059F.
- [36] Lu Ma y Manuchehr Soleimani. «Magnetic Induction Tomography Methods and Applications: A Review». En: Measurement Science and Technology 28.7 (2017), pág. 072001. DOI: 10.1088/1361-6501/aa6de8.
- [37] Hermann Scharfetter, Roberto Casañas y Xavier Rosell. «"Biological tissue characterization by magnetic induction spectroscopy (MIS): Requirements and limitations" ». En: *IEEE transactions on bio-medical engineering* 50 (ago. de 2003), págs. 870-80. DOI: 10.1109/ TBME.2003.813533.
- [38] Jonathan R Wolpaw et al. «Brain-computer interfaces for communication and control». En: *Clin. Neurophysiology* 113 (2002), págs. 767-791.
- [39] X Gao et al. «A BCI-based environmental controller for the motion-disabled». En: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 11 (2003), págs. 137-140.
- [40] R Srinivasan, F A Bibi y P L Nunez. «Steady-state visual evoked potentials: distributed local sources and wave-like dynamics are sensitive to flicker frequency». En: Brain Topography 18 (2006), págs. 167-187.
- [41] Pablo García, Enrique Spinelli y Graciela Toccaceli. «An embedded system for evoked biopotential acquisition and processing». En: Int. J. of Embedded Systems 6 (ene. de 2014), págs. 86-93. DOI: 10.1504/IJES.2014.060920.

- [42] Fernande Grandjean y Gary J. Long. «Best Practices and Protocols in Mössbauer Spectroscopy». En: Chemistry of Materials 33.11 (2021), págs. 3878-3904. DOI: 10.1021/acs. chemmater.1c00326.
- [43] Alfred X. Trautwein Philipp Gütlich Eckhard Bill. Mössbauer Spectroscopy and Transition Metal Chemistry: Fundamentals and Applications. Weinheim: Springer Science y Business Media, 2010.
- [44] G.A. Pasquevich y A.L. Veiga. «Velocity waveform digitalization for quality control and enhancement of Mössbauer effect spectra acquisition». En: *Hyperfine Interactions* 242 (2021). DOI: 10.1007/s10751-021-01741-7.
- [45] David Harcombe, Michael Ruppert y Andrew Fleming. «A review of demodulation techniques for multifrequency atomic force microscopy». En: Beilstein Journal of Nanotechnology 11 (ene. de 2020), págs. 76-91. DOI: 10.3762/bjnano.11.8.
- [46] R. E. Kalman. «A New Approach to Linear Filtering and Prediction Problems». En: Journal of Basic Engineering 82.1 (mar. de 1960), págs. 35-45. ISSN: 0021-9223. DOI: 10.1115/1. 3662552.
- [47] M.S. Grewal, V.D. Henderson y R.S. Miyasako. «Application of Kalman filtering to the calibration and alignment of inertial navigation systems». En: *IEEE Transactions on Automatic Control* 36.1 (1991), págs. 3-13. DOI: 10.1109/9.62283.
- [48] S. Y. Chen. «Kalman Filter for Robot Vision: A Survey». En: IEEE Transactions on Industrial Electronics 59.11 (2012), págs. 4409-4420. DOI: 10.1109/TIE.2011.2162714.
- [49] Michael R. P. Ragazzon et al. «Lyapunov Estimator for High-Speed Demodulation in Dynamic Mode Atomic Force Microscopy». En: *IEEE Transactions on Control Systems Tech*nology 26.2 (2018), págs. 765-772. DOI: 10.1109/TCST.2017.2692721.
- [50] Michael G. Ruppert et al. «Frequency domain analysis of robust demodulators for highspeed atomic force microscopy». En: 2017 American Control Conference (ACC). 2017, págs. 1562-1567. DOI: 10.23919/ACC.2017.7963175.
- [51] David M. Harcombe et al. «Higher-harmonic AFM imaging with a high-bandwidth multifrequency Lyapunov filter». En: 2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM). 2017, págs. 725-730. DOI: 10.1109/AIM.2017.8014103.
- [52] Michael G. Ruppert et al. «Direct Design of Closed-loop Demodulators for Amplitude Modulation Atomic Force Microscopy». En: 2018 Annual American Control Conference (ACC). 2018, págs. 4336-4341. DOI: 10.23919/ACC.2018.8430896.
- [53] Matías Javier Oliva et al. «Implementation of a digital Lock-in amplifier in SoC-FPGA for Biomedical Tomography». En: Congress Argentino de Bioingeniería 2022. San Juan, Argentina, sep. de 2022.
- [54] Matías J. Oliva et al. «Improving the computational efficiency of lock-in algorithms through coherent averaging». En: *Digital Signal Processing* 154 (2024), pág. 104693. ISSN: 1051-2004. DOI: https://doi.org/10.1016/j.dsp.2024.104693.
- [55] Norbert Wiener. «Generalized harmonic analysis». En: Acta Mathematica 55.none (1930), págs. 117-258. DOI: 10.1007/BF02546511.
- [56] A. Khintchine. «Korrelationstheorie der stationären stochastischen Prozesse». En: Mathematische Annalen 109 (1934), págs. 604-615.

- [57] Stanford Research Systems. About Lock-in Amplifiers Application note #3. [Online]. Available: https://www.thinksrs.com/downloads/pdfs/applicationnotes/AboutLIAs.pdf.
- [58] Jean-Jacques Vandenbussche, Peter Lee y Joan Peuteman. «On the Accuracy of Digital Phase Sensitive Detectors Implemented in FPGA Technology». En: *IEEE Transactions on Instrumentation and Measurement* 63 (ago. de 2014), págs. 1926-1936. DOI: 10.1109/TIM. 2014.2303257.
- [59] M. A. Banuelos-Saucedo y K. B. Ozanyan. «Fast response lock-in amplifier». En: 2012 8th IEEE International Symposium on Instrumentation and Control Technology (ISICT) Proceedings. London, UK, 2012, págs. 122-125. DOI: 10.1109/ISICT.2012.6291623.
- [60] Chuanhong He et al. «A digital phase-sensitive detector for electrical impedance tomography». En: 2008 World Automation Congress. Hawaii, HI, 2008, págs. 1-4.
- [61] S. Golestan et al. «Moving Average Filter Based Phase-Locked Loops: Performance Analysis and Design Guidelines». En: *IEEE Transactions on Power Electronics* 29.6 (2014), págs. 2750-2763. DOI: 10.1109/TPEL.2013.2273461.
- [62] Matías Oliva. Repositorio con recursos para probar el algoritmo CALI. https://github. com/ushikawa93/lockin_coherent_average. 2023.
- [63] C. R. Cosens. «A balance-detector for alternatingcurrent bridges». En: Proceedings of the Physical Society 46 (1934), pág. 818.
- [64] W. C. Michels. «A Double Tube Vacuum Tube Voltmeter». En: Rev. Sci. Instrum. 9 (1938), pág. 10.
- [65] W. C. Michels y N. L. Curtis. «A Pentode LockIn Amplifier of High Frequency Selectivity». En: Rev. Sci. Instrum. 12 (1941), pág. 444.
- [66] Kaushal Kishore. «Evolution of Lock In Amplifier as Portable Sensor Interface Platform: A Review». En: *IEEE Sensors Journal* PP (mayo de 2020), págs. 1-1. DOI: 10.1109/JSEN. 2020.2993309.
- [67] Chuanhong He et al. «A digital phase-sensitive detector for electrical impedance tomography». En: (ene. de 2008).
- [68] Gonzalo Macias-Bobadilla et al. «Dual-Phase Lock-In Amplifier Based on FPGA for Low-Frequencies Experiments». En: Sensors 16 (mar. de 2016), pág. 379. DOI: 10.3390/ s16030379.
- [69] Costantino Giaconia et al. «Exploring FPGA Based Lock-in Techniques for Brain Monitoring Applications». En: (feb. de 2017). DOI: 10.20944/preprints201702.0050.v1.
- [70] Guy Stimpson et al. «An open-source high-frequency lock-in amplifier». En: Review of Scientific Instruments 90 (sep. de 2019), pág. 094701. DOI: 10.1063/1.5083797.
- [71] Yafeng Hu et al. «FPGA Implementation of a Complex Permeability Measurement Instrument». En: Tianjin Key Laboratory of Process Measurement and Control School of Electrical and Information Engineering Tianjin University Tianjin (2019).
- [72] Marcelo Luda et al. «Compact embedded device for lock-in measurements and experiment active control». En: (feb. de 2019).
- [73] Yuji Sukekawa y Takamichi Nakamoto. «Odor biosensor system based on image lock-in measurement for odorant discrimination». En: *Electronics and Communications in Japan* 102 (ene. de 2019). DOI: 10.1002/ecj.12154.

- [74] Zhongyuan Fang et al. «A Quadrature Adaptive Coherent Lock-in Chip-Based Sensor for Accurate Photoacoustic Detection». En: 2020 IEEE International Symposium on Circuits and Systems (ISCAS). 2020, págs. 1-4. DOI: 10.1109/ISCAS45731.2020.9180612.
- [75] Cheng Zhang et al. «FPGA-Based Digital Lock-in Amplifier With High-Precision Automatic Frequency Tracking». En: *IEEE Access* PP (jun. de 2020), págs. 1-1. DOI: 10.1109/ACCESS. 2020.3006070.
- [76] Jiabing Song et al. «Application of digital Lock-in amplifier in complex electromagnetic interference of substation». En: Progress In Electromagnetics Research C 106 (ene. de 2020), págs. 105-120. DOI: 10.2528/PIERC20071001.
- [77] Jun Lu. «Lock-in frequency measurement with high precision and efficiency». En: Review of Scientific Instruments 91 (jul. de 2020), pág. 075106. DOI: 10.1063/5.0002377.
- [78] Yanyu Wei et al. «Design of weak magnetic signal detection system for residual stress detection». En: IOP Conference Series: Earth and Environmental Science 632 (ene. de 2021), pág. 042064. DOI: 10.1088/1755-1315/632/4/042064.
- [79] James M Masciotti, Joseph M Lasker y Andreas H Hielscher. «Digital Lock-In Detection for Discriminating Multiple Modulation Frequencies With High Accuracy and Computational Efficiency». En: *IEEE Transactions on Instrumentation and Measurement* 57.1 (2008), págs. 170-176. DOI: 10.1109/TIM.2007.904021.
- [80] Gang Li et al. «A novel algorithm combining oversampling and digital lock-in amplifier of high speed and precision». En: *The Review of scientific instruments* 82 (sep. de 2011), pág. 095106. DOI: 10.1063/1.3633943.
- [81] J. Leis, Philip Martin y David Buttsworth. «Simplified digital lock-in amplifier algorithm».
 En: *Electronics Letters* 48 (mar. de 2012), págs. 259-261. DOI: 10.1049/el.2012.0193.
- [82] Shengzhao Zhang et al. «Optimization of a digital lock-in algorithm with a square-wave reference for frequency-divided multi-channel sensor signal detection». En: *Review of Scientific Instruments* 87 (ago. de 2016). DOI: 10.1063/1.4959978.
- [83] Aaron Troy, Francisco Paz y Martin Ordonez. «Robust Digital Algorithm for Rapid Phase Angle Tracking in Wireless Power Transfer». En: 2020 IEEE Applied Power Electronics Conference and Exposition (APEC). 2020, págs. 832-837. DOI: 10.1109/APEC39645.2020. 9124468.
- [84] Saikat Jana y S. Srinivas. «A Computationally Efficient Harmonic Extraction Algorithm for Grid Applications». En: *IEEE Transactions on Power Delivery* 37.1 (2022), págs. 146-154. DOI: 10.1109/TPWRD.2021.3054554.
- [85] O. Rompelman y H.H. Ros. «Coherent averaging technique: A tutorial review Part 1: Noise reduction and the equivalent filter». En: Journal of Biomedical Engineering 8.1 (1986), págs. 24-29. DOI: 10.1016/0141-5425(86)90026-9.
- [86] O. Rompelman y H.H. Ros. «Coherent averaging technique: A tutorial review Part 2: Trigger jitter, overlapping responses and non-periodic stimulation». En: Journal of Biomedical Engineering 8.1 (1986), págs. 30-35. DOI: 10.1016/0141-5425(86)90027-0.
- [87] O. Rompelman y H.H. Ros. «Coherent average technique a tutorial review». En: Journal of Biomedical Engineering 13.4 (1991), págs. 275-280. ISSN: 0141-5425. DOI: https://doi. org/10.1016/0141-5425(91)90108-J.

- [88] Eaglekumar Tarpara y V. Patankar. «Reconfigurable Hardware Implementation of Coherent Averaging Technique for Ultrasonic NDT Instruments». En: Ultrasonics 105 (feb. de 2020), pág. 106106. DOI: 10.1016/j.ultras.2020.106106.
- [89] Xin Zhao et al. «Fast acquisition algorithm based on improved coherent average method for high dynamic BDS signal». En: 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC). 2018, págs. 1-8. DOI: 10.1109/GNCC42960.2018.9019140.
- [90] Dennis T.T. Plachta et al. «Detection of baroreceptor activity in rat vagal nerve recording using a multi-channel cuff-electrode and real-time coherent averaging». En: 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society. 2012, págs. 3416-3419. DOI: 10.1109/EMBC.2012.6346699.
- [91] Brent W. Robinson y Mohammad Saquib. «Adaptive coherent averaging for real-time electrocardiogram enhancement». En: 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. 2014, págs. 5048-5051. DOI: 10.1109/EMBC. 2014.6944759.
- [92] Intel Corporation. Intel® High Level Synthesis Compiler: User Guide. Online. Accessed on November 2022. URL: https://www.intel.com/content/www/us/en/docs/programmable/ 683456/22-3/pro-edition-user-guide.html.
- [93] AMD- Xilinx. Vitis High-Level Synthesis User Guide (UG1399). Online. Accessed on November 2022. URL: https://docs.xilinx.com/r/en-US/ug1399-vitis-hls.
- [94] Intel Corporation. SDK Intel® FPGA para OpenCL. Online. Accessed on November 2022. URL: https://www.intel.la/content/www/xl/es/support/programmable/supportresources/design-software/opencl-support.html.
- [95] S. Lahti et al. «Are We There Yet? A Study on the State of High-Level Synthesis». En: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 38.5 (mayo de 2019), págs. 898-911. DOI: 10.1109/TCAD.2018.2834439.
- [96] Matias Oliva. Signal processing in FPGA. https://github.com/ushikawa93/signal_ processing_fpga.git. 2022.
- [97] Matías Oliva. Signal Processing Red Pitaya. https://github.com/ushikawa93/signal_ processing_red_pitaya. 2024.
- [98] Matías Javier Oliva et al. «Open-source SoC-FPGA Platform for Signal Processing». En: Southern Programmable Logic Conference 2023. San Luis, Argentina, mar. de 2023.
- [99] Terasic Technologies Inc. *DE1-SoC User Manual*. Accessed on November 2022. URL: http: //www.terasic.com.tw/.
- [100] Terasic Technologies Inc. THDB-ADA High-Speed A/D and D/A Development Kit User Manual. Accessed November 2022. URL: https://www.terasic.com.tw/.
- [101] Terasic High-speed AD/DA Daughter Board Schematic. Online. URL: http://ridl.cfd. rit.edu/products/manuals/Altera/THDB_ADA_CD_V1.3/Schematic/ada_v11.pdf.
- [102] Intel Corporation. Avalon (R) Interface Specifications. 2020. URL: https://www.intel.com/ (visitado 2022).
- [103] Intel Corporation. Understanding Metastability in FPGAs. https://www.intel.com/ content/www/us/en/content-details/650346/understanding-metastability-infpgas.html. Accessed: 2024-05-29. 2009.

- [104] Inc. Xilinx. DDS Compiler. Accessed: 2024-05-27. 2024. URL: https://www.xilinx.com/ products/intellectual-property/dds_compiler.html#documentation.
- [105] Shruti Hathwalia y Meenakshi Yadav. «Design and analysis of a 32 bit linear feedback shift register using VHDL». En: Indian Journal of Pure and Applied Physics (IJPAP) 52.3 (2015), págs. 203-209.
- [106] GNU Scientific Library: Other random number generators. Online. Accessed on November 2022. URL: https://www.gnu.org/software/gsl/doc/html/rng.html#other-randomnumber-generators.
- [107] Mono Project. Mono Project. URL: https://www.mono-project.com/ (visitado 2022).
- [108] Matías Oliva. Funcionalidades para controlar un lockin SR865 de Stanford en forma remota. https://github.com/ushikawa93/sr865_software. 2024.
- [109] RedPitaya. STEMLAB 125-14. Available at https://redpitaya.com/stemlab-125-14/. 2023.
- [110] Inc. Xilinx. Processing System7. Accessed: 2024-05-27. 2024. URL: https://www.xilinx. com/products/intellectual-property/processing_system7.html#documentation.
- [111] Inc. Xilinx. AXI Interconnect. Accessed: 2024-05-27. 2024. URL: https://www.xilinx.com/ products/intellectual-property/axi_interconnect.html#documentation.
- [112] Inc. Xilinx. AXI GPIO. Accessed: 2024-05-27. 2024. URL: https://www.xilinx.com/ products/intellectual-property/axi_gpio.html#documentation.
- [113] J. C. Candy y G. C. Temes. Oversampling Delta-Sigma Data Converters: Theory, Design, and Simulation. Wiley-IEEE Press, 1992.
- [114] Xilinx. Zynq-7000 SoC Technical Reference Manual. Accessed: 26/02/2024. Xilinx. 2023. URL: https://docs.xilinx.com/r/en-US/ug585-zynq-7000-SoC-TRM/Programmable-Logic-Description.
- [115] Matías Oliva. Lock-in en DE1-SoC. https://github.com/ushikawa93/lockin_de1soc. 2024.
- [116] Matías Oliva. Lock-in en Red Pitaya. https://github.com/ushikawa93/lockin_ redpitaya. 2024.
- [117] Y. N. Chen et al. «Imaging hemorrhagic stroke with magnetic induction tomography: realistic simulation and evaluation». En: *Physiological Measurement* 31 (2010), págs. 809-827.
- [118] X. Ma et al. «Hardware and software design for an electromagnetic induction tomography system for high contrast metal process applications». En: Measurement Science and Technology 17 (2006), págs. 111-118.
- [119] R. A. Albrechtsen, Z. Z. Yu y A. J. Peyton. «Preliminary experiments on the investigation of the inductive technique for measuring water content in multiphase flow». En: *Proceedings* of ECAPT. Bergen, 1995, págs. 205-213.
- [120] E. A. Hammer y G. Fossdal. «A new water-in-oil monitor based on high frequency magnetic field excitation». En: Proceedings of the 2nd International Symposium on Process Tomography. Wroclaw, Poland, 2002, págs. 9-16.
- [121] M. Zhang, L. Ma y M. Soleimani. «Dual modality MIT-ECT multi-phase flow imaging». En: Flow Measurement and Instrumentation 46 (2015), págs. 240-254.

- [122] M. Soleimani y W. Lionheart. «Image reconstruction in three-dimensional magnetostatic permeability tomography». En: *IEEE Transactions on Magnetics* 41 (2005), págs. 1274-1279.
- [123] L. Ma y M. Soleimani. «Induction Spectroscopy for Permeability Imaging». En: Scientific Reports 8 (2018), pág. 7025. DOI: 10.1038/s41598-018-25552-x.
- [124] E. Alphandéry. «Iron oxide nanoparticles as multimodal imaging tools». En: RSC Advances 9 (oct. de 2019), págs. 40577-40588.
- [125] J. Zhang et al. «Quantifying iron-oxide nanoparticles at high concentration based on longitudinal relaxation». En: Magnetic Resonance in Medicine 71 (jun. de 2014), págs. 1982-1988.
- [126] B. Gleich y J. Weizenecker. «Tomographic Imaging Using the Nonlinear Response of Magnetic Particles». En: Nature 435.7046 (2005), págs. 1214-1217.
- [127] E. Romanus et al. «Magnetic nanoparticle relaxation measurement as a novel tool for in vivo diagnostics». En: Journal of Magnetism and Magnetic Materials 252 (nov. de 2002), págs. 387-389.
- [128] Hu et al. «Magnetoacoustic imaging of magnetic iron oxide nanoparticles embedded in biological tissues with microsecond magnetic stimulation». En: Applied Physics Letters 100 (ene. de 2012), pág. 013704.
- [129] P. Gawne J. Pellico y R. M. de Rosales. «Radiolabelling of nanomaterials for medical imaging and therapy». En: *Chemical Society Reviews* 50 (ene. de 2021), págs. 3355-3423.
- [130] Matías Javier Oliva, Pablo Andrés García y A. L. Veiga. «Evaluación del desempeño de sistemas embebidos heterogéneos para aplicaciones de tomografía magnética». En: Jornadas de Investigación y Transferencia UNLP. La Plata, Argentina, mar. de 2019.
- [131] H. Scharfetter et al. «A new type of gradiometer for magnetic induction tomography». En: Physiological Measurement 26 (2005), S307-S318. DOI: 10.1088/0967-3334/26/5/022.
- [132] B. Ficko et al. «Development of a magnetic nanoparticle susceptibility magnitude imaging array». En: Physics in Medicine and Biology 59 (2014), págs. 1047-1071. DOI: 10.1088/ 0031-9155/59/4/1047.
- [133] R. G. Liu, M. Soleimani y W. R. B. Lionheart. «Time difference imaging of magnetic induction tomography in a three-layer physical phantom». En: *Measurement Science and Technology* 25 (2014), pág. 065402. DOI: 10.1088/0957-0233/25/6/065402.
- [134] D. F. Coral et al. «Nanoclusters of crystallographically aligned nanoparticles for magnetic thermotherapy: aqueous ferrofluid, agarose phantoms and ex vivo melanoma tumour assessment». En: Nanoscale 10 (2018), págs. 21262-21274. DOI: 10.1039/C8NR06115D.
- [135] S. A. Gonzalez et al. «Portable electromagnetic field applicator for magnetic hyperthermia experiments». En: *IEEE 8th Latin American Symposium on Circuits and Systems (LAS-CAS)*. 2017, págs. 1-4. DOI: 10.1109/LASCAS.2017.7948084.
- [136] T. Nakagawa et al. «Electrical properties of biodegradable polylactic acid film». En: IEEE Transactions on Industry Applications 40.4 (jul. de 2004), pág. 1020.
- [137] Analog Devices. Low Cost Low Power Instrumentation Amplifier AD620. Accessed: 2024-10-30. 2011. URL: https://www.analog.com/media/en/technical-documentation/datasheets/AD620.pdf.
- [138] Texas Instruments. THP210 Ultra-Low Offset, High-Voltage, Low-Noise, Precision, Fully-Differential Amplifier. Accessed: 2024-10-30. 2020. URL: https://www.ti.com/lit/ds/ symlink/thp210.pdf.

- [139] Aleksandra Chabuda, Przemyslaw Durka y Jerzy Zygierewicz. «High Frequency SSVEP-BCI With Hardware Stimuli Control and Phase-Synchronized Comb Filter». En: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26.2 (2018), págs. 344-352.
- [140] Jinyi Long et al. «A Hybrid Brain Computer Interface to Control the Direction and Speed of a Simulated or Real Wheelchair». En: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 20.5 (2012), págs. 720-729.
- [141] Dimitrios Pantazis et al. «A Review of Hybrid Brain-Computer Interface Systems». En: Advances in Human-Computer Interaction 2013 (2013), pág. 187024. ISSN: 1687-5893. DOI: 10.1155/2013/187024.
- [142] G. Pfurtscheller y F.H. Lopes da Silva. «Event-related EEG/MEG synchronization and desynchronization: basic principles». En: *Clinical Neurophysiology* 110.11 (1999), págs. 1842-1857. ISSN: 1388-2457. DOI: https://doi.org/10.1016/S1388-2457(99)00141-8.
- [143] Eric W. Sellers, Yoni Arbel y Emanuel Donchin. «12 BCIs That Use P300 Event-Related Potentials». En: Brain-Computer Interfaces: Principles and Practice. 2012, pág. 215.
- [144] N. Birbaumer et al. «A spelling device for the paralyzed». En: Nature 398.6725 (1999), págs. 297-298. ISSN: 1476-4687. DOI: 10.1038/18581.
- [145] Kuo-Kai Shyu, Po-Lei Lee, Ming-Hao Lee et al. «Development of a Low-Cost FPGA-Based SSVEP BCI Multimedia Control System». En: *IEEE Transactions on Biomedical Circuits* and Systems 4 (2010), págs. 125-132.
- [146] M. A. Lopez-Gordo, F. Pelayo y A. Prieto. «A high performance SSVEP-BCI without gazing». En: The 2010 International Joint Conference on Neural Networks (IJCNN). 2010, págs. 1-5. DOI: 10.1109/IJCNN.2010.5596325.
- [147] Guang Chen, Dandan Song y Lejian Liao. «A Multi-channel SSVEP-Based Brain-Computer Interface Using a Canonical Correlation Analysis in the Frequency Domain». En: Foundations and Practical Applications of Cognitive Systems and Information Processing. Ed. por Fuchun Sun, Dewen Hu y Huaping Liu. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, págs. 603-616.
- [148] Xiaorong Gao et al. «A BCI-Based Environmental Controller for the Motion-Disabled». En: IEEE Transactions on Neural Systems and Rehabilitation Engineering 11.2 (jun. de 2003).
- [149] Manuel Middendorf et al. «Brain-computer interfaces based on the steady-state visual-evoked response». En: *IEEE Transactions on Rehabilitation Engineering* 8 (2000), págs. 211-214.
- [150] Gernot R Müller-Putz et al. «Steady-state visual evoked potential (SSVEP)-based communication: impact of harmonic frequency components». En: Journal of Neural Engineering 2 (2005), págs. 123-130.
- [151] Gernot R Müller-Putz et al. «Comparison of DFT and lock-in amplifier features and search for optimal electrode positions in SSVEP-based BCI». En: *Journal of Neuroscience Methods* 168 (2008), págs. 174-181.
- [152] Daniel E Garcia, Kuan-Wei Zheng, Yu-Shiang Tao et al. «Capturing Pictures from Human Vision Using SSVEP and Lock-in Amplifier». En: 2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). IEEE. 2020, págs. 172-179.
- [153] P. A. García, E. M. Spinelli y G. Toccaceli. «An Embedded System for Evoked Biopotential Acquisition and Processing». En: International Journal of Embedded Systems (IJES) 6.1 (2014), págs. 86-93.

- [154] Xiaogang Chen et al. «High speed spelling with a brain-computer interface». En: *Proceedings* of the National Academy of Sciences 112.44 (2015).
- [155] Terasic Inc. DE10-Lite Development Kit User Manual. Accessed: February 22, 2024. Terasic Inc. 2020. URL: https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language= English&CategoryNo=218&No=1021&PartNo=4#contents.
- [156] Matías Javier Oliva. SSVEP FPGA ADS1299. https://github.com/ushikawa93/ssvep_ fpga_ads1299. 2024.
- [157] Matías Oliva et al. «Real-Time SSVEP Measurements Through Lock-In Detection in FPGA-Based Platform». En: Advances in Bioengineering and Clinical Engineering. Ed. por Fernando Emilio Ballina et al. Cham: Springer Nature Switzerland, 2024, págs. 161-171. ISBN: 978-3-031-61960-1. DOI: 10.1007/978-3-031-61960-1_16.
- [158] Federico N. Guerrero, Matías Javier Oliva y Enrique Mario Spinelli. «Real-time noise reduction through independent channel averaging for real-time biomedical signal acquisition». En: Southern Programmable Logic Conference 2023. San Luis, Argentina, mar. de 2023.
- [159] M. J. Oliva et al. «SoC-FPGA systems for the acquisition and processing of electroencephalographic signals». En: International Journal of Reconfigurable and Embedded Systems (IJRES) 10.3 (nov. de 2021), págs. 237-248. DOI: 10.11591/ijres.v10.i3.pp237-248.
- [160] M. J. Oliva et al. «Interfaz cerebro computadora basada en SoC». En: Congreso Argentino de Sistemas Embebidos 2020. Buenos Aires, Argentina, ago. de 2020.
- [161] P. García, E. Spinelli y M. Oliva. «System on Chip based brain-computer interface». En: Congreso Argentino de Control Automático 2020. Buenos Aires, Argentina, oct. de 2020.
- [162] M. J. Oliva, P. A. García y E. M. Spinelli. «A System on Chip based electroencephalogram acquisition system». En: *Revista Argentina de Bioingeniería* 24.3 (2020). Presentado en Congreso Argentino de Bioingeniería 2020.
- [163] AAMI. Medical Electrical Equipment Part 2-25: Particular Requirements For The Basic Safety And Essential Performance Of Electrocardiographs. Standard ANSI/AAMI/IEC 60601-2-25:2011 (R2016). Association for the Advancement of Medical Instrumentation, 2016.
- [164] J. J. Halford et al. «American Clinical Neurophysiology Society Guideline 4: Recording Clinical EEG on Digital Media». En: *The Neurodiagnostic Journal* 56.4 (oct. de 2016), págs. 261-265.
- [165] R. Merletti y G. Cerone. «Tutorial. Surface EMG detection, conditioning and pre-processing: Best practices». En: Journal of Electromyography and Kinesiology 54 (oct. de 2020), pág. 102440.
- [166] Hans-Jörg Scheer, Thomas Sander y Lutz Trahms. «The influence of amplifier, interface and biological noise on signal quality in high-resolution EEG recordings». En: *Physiological* measurement 27 (2006), págs. 109-117.
- [167] F. N. Guerrero y E. M. Spinelli. «Chapter 4: Biopotential acquisition systems». En: Medicine-Based Informatics and Engineering. Ed. por F. Simini y P. Bertemes-Filho. Cham: Springer International Publishing, 2022, págs. 51-79.
- [168] T Degen y H Jäckel. «Enhancing interference rejection of preamplified electrodes by automated gain adaption». En: *IEEE Transactions on Biomedical Engineering* 51.11 (nov. de 2004), págs. 2031-9.

- [169] F N Guerrero y E M Spinelli. «A two-wired ultra-high input impedance active electrode». En: *IEEE Transactions on Biomedical Circuits and Systems* 12.2 (2018), págs. 437-445.
- [170] D Berry, F Duignan y R Hayes. «An Investigation of the use of a High Resolution ADC as a Digital Biopotential Amplifier». En: 4th European Conference of the International Federation for Medical and Biological Engineering. 2009, págs. 1-6.
- [171] X Yang et al. «A 108 dB DR Sigma-Delta Front-End With 720 mV pp Input Range and > 300 mV Offset Removal for Multi-Parameter Biopotential Recording». En: *IEEE Transactions on Biomedical Circuits and Systems* 15.2 (2021), págs. 199-209.
- [172] Y Jung et al. «A wide-dynamic-range neural-recording ic with automatic gain-controlled afe and ct dynamic-zoom Delta-Sigma adc for saturation-free closed-loop neural interfaces». En: *IEEE Journal of Solid-State Circuits* 57.10 (2022), págs. 3071-3082.
- [173] J Chen et al. «A high precision eeg acquisition system based on the compactpci platform». En: 2014 7th International Conference on Biomedical Engineering and Informatics. IEEE. 2014, págs. 511-516.
- [174] D Liu et al. «FPGA-based real-time compressed sensing of multichannel EEG signals for wireless body area networks». En: Biomedical Signal Processing and Control 49 (2019), págs. 221-230.
- [175] Texas Instruments. ADS131E0x 4-, 6-, and 8-Channel, 24-Bit, Simultaneously-Sampling, Delta-Sigma ADC. 2017. URL: https://www.ti.com/lit/ds/symlink/ads131e08.pdf? ts=1708542484714.
- [176] Texas Instruments. ADS1299-x Low-Noise, 4-, 6-, 8-Channel, 24-Bit, Analog-to-Digital Converter for EEG and Biopotential Measurements. 2017. URL: https://www.ti.com/lit/ds/ symlink/ads1299.pdf?ts=1708524186421.
- [177] S. Nishimura, Y. Tomita y T. Horiuchi. «Clinical application of an active electrode using an operational amplifier». En: *IEEE Transactions on Biomedical Engineering* 39.10 (1992), págs. 1096-1099.
- [178] Bruce B. Winter y John G. Webster. «Driven-right-leg circuit design». En: IEEE Transactions on Biomedical Engineering BME-30 (1983), págs. 62-66.
- [179] F. N. Guerrero y E. Spinelli. «High gain driven right leg circuit for dry electrode systems». En: Medical Engineering and Physics 39 (jun. de 2017), págs. 117-122.
- [180] Analog Devices. ADUM6401 Datasheet: Dual-Channel Digital Isolators. Rev. B. Analog Devices, Inc. Norwood, MA, 2020. URL: https://www.analog.com/media/en/technicaldocumentation/data-sheets/adum6400_6401_6402_6403_6404.pdf.
- [181] Zhigang Wu, Yuting Lai, Yang Xia et al. «Stimulator selection in SSVEP-based BCI». En: Medical Engineering & Physics 30 (2008), págs. 1079-1088.
- [182] Rudolf L Mössbauer. «Kernresonanzfluoreszenz von Gammastrahlung in Ir191». En: Zeitschrift für Physik 151 (1958), pág. 124.
- [183] T. E. Cranshaw. «Mössbauer Spectroscopy». En: Journal of Physics E: Scientific Instruments 7 (1974), pág. 497.
- [184] Alejandro L. Veiga. «Avances en Espectroscopia Mössbauer: Nuevas Estrategias para Experimentos de Velocidad Programable». Tesis doctoral. La Plata, Argentina: Universidad Nacional de La Plata, 2008.

- [185] M.J. Oliva. Adquisidor en Red Pitaya. https://github.com/ushikawa93/adquisidor_ red_pitaya. 2024.
- [186] RedPitaya. Developers Guide. Available at https://redpitaya.readthedocs.io/en/ latest/developerGuide/devGuideTop.html. 2023.
- [187] Matías Javier Oliva, Gustavo Pasquevich y Alejandro Luis Veiga. «Embedded device for digitalizing monitor and error signals in Mössbauer Spectroscopy». En: *Hyperfine Interact* 244.23 (2023). DOI: 10.1007/s10751-023-01834-5.
- [188] Red Pitaya Team. System Calibration Red Pitaya Docs. https://redpitaya. readthedocs.io/en/latest/appsFeatures/systemtool/calibration.html. Último acceso: 27 de febrero de 2025. 2025.
- [189] G. A. Basso, I. J. Bruvera y P. Mendoza Zelis. «Ciclos Magnéticos de Radiofrecuencia: Caracterización de Nanopartículas para Aplicaciones Biomédicas». En: 7^o Jornadas ITEE - 2023. Facultad de Ingeniería, UNLP. La Plata, Argentina: Facultad de Ingeniería, UNLP, 2023, pág. 135.