

TRABAJO FIN DE GRADO

BRAIN COMPUTER INTERFACES: DESARROLLO DE UN SISTEMA DE IDENTIFICACIÓN DE ESTADOS MENTALES ALFA.

FEBRERO 2016

Pablo Iglesias López

DIRECTORES DEL TRABAJO FIN DE GRADO:

Eduardo de la Torre

Javier Mora de Sambricio

TRABAJO FIN DE GRADO PARA
LA OBTENCIÓN DEL TÍTULO DE
GRADUADO EN INGENIERÍA EN
TECNOLOGÍAS INDUSTRIALES

AGRADECIMIENTOS

A mi familia por su apoyo a lo largo de estos cuatro años fuera de casa , principalmente a mis padres, sin los cuales no habría podido conseguir estar donde estoy hoy.

A Eduardo de la Torre, profesor de la Escuela Técnica Superior de Ingenieros Industriales de la UPM , y a Javier Mora, investigador en formación en el Centro de Electrónica Industrial(CEI). Por su disposición, entrega y dedicación en mi trabajo. También a todo el personal del CEI por su gran espíritu de colaboración.

A mis compañeros Ramón y Carlos, por su preocupación y entrega durante los primeros compases del proyecto.

A mis compañeros y amigos sin los cuales esta etapa de cuatro años en nuestras vidas habría sido diferente.

RESUMEN

En el mundo actual las aplicaciones basadas en sistemas biométricos, es decir, aquellas que miden las señales eléctricas de nuestro organismo, están creciendo a un gran ritmo. Todos estos sistemas incorporan sensores biomédicos, que ayudan a los usuarios a controlar mejor diferentes aspectos de la rutina diaria, como podría ser llevar un seguimiento detallado de una rutina deportiva, o de la calidad de los alimentos que ingerimos. Entre estos sistemas biométricos, los que se basan en la interpretación de las señales cerebrales, mediante ensayos de electroencefalografía o EEG están cogiendo cada vez más fuerza para el futuro, aunque están todavía en una situación bastante incipiente, debido a la elevada complejidad del cerebro humano, muy desconocido para los científicos hasta el siglo XXI.

Por estas razones, los dispositivos que utilizan la interfaz cerebro-máquina, también conocida como BCI (Brain Computer Interface), están cogiendo cada vez más popularidad. El funcionamiento de un sistema BCI consiste en la captación de las ondas cerebrales de un sujeto para después procesarlas e intentar obtener una representación de una acción o de un pensamiento del individuo. Estos pensamientos, correctamente interpretados, son posteriormente usados para llevar a cabo una acción. Ejemplos de aplicación de sistemas BCI podrían ser mover el motor de una silla de ruedas eléctrica cuando el sujeto realice, por ejemplo, la acción de cerrar un puño, o abrir la cerradura de tu propia casa usando un patrón cerebral propio.

Los sistemas de procesamiento de datos están evolucionando muy rápido con el paso del tiempo. Los principales motivos son la alta velocidad de procesamiento y el bajo consumo energético de las FPGAs (Field Programmable Gate Array). Además, las FPGAs cuentan con una arquitectura reconfigurable, lo que las hace más versátiles y potentes que otras unidades de procesamiento como las CPUs o las GPUs.

En el CEI (Centro de Electrónica Industrial), donde se lleva a cabo este TFG, se dispone de experiencia en el diseño de sistemas reconfigurables en FPGAs. Este TFG es el segundo de una línea de proyectos en la cual se busca obtener un sistema capaz de procesar correctamente señales cerebrales, para llegar a un patrón común que nos permita actuar en consecuencia. Más concretamente, se busca detectar cuando una persona está quedándose dormida a través de la captación de unas ondas cerebrales, conocidas como ondas alfa, cuya frecuencia está acotada entre los 8 y los 13 Hz. Estas ondas, que aparecen cuando cerramos los ojos y dejamos la mente en blanco, representan un estado de relajación mental.

Por tanto, este proyecto comienza como inicio de un sistema global de BCI, el cual servirá como primera toma de contacto con el procesamiento de las ondas cerebrales, para el posterior uso de hardware reconfigurable sobre el cual se implementarán los algoritmos evolutivos. Por ello se vuelve necesario desarrollar un sistema de procesamiento de datos en una FPGA. Estos datos se procesan siguiendo la metodología de procesamiento digital de señales, y en este caso se realiza un análisis de la frecuencia utilizando la transformada rápida de Fourier, o FFT.

Una vez desarrollado el sistema de procesamiento de los datos, se integra con otro sistema que se encarga de captar los datos recogidos por un ADC (Analog to Digital Converter), conocido como ADS1299. Este ADC está especialmente diseñado para captar potenciales del cerebro humano. De esta forma, el sistema final capta los datos mediante el ADS1299, y

los envía a la FPGA que se encarga de procesarlos. La interpretación es realizada por los usuarios que analizan posteriormente los datos procesados.

Para el desarrollo del sistema de procesamiento de los datos, se dispone primariamente de dos plataformas de estudio, a partir de las cuales se captarán los datos para después realizar el procesamiento:

1. La primera consiste en una herramienta comercial desarrollada y distribuida por OpenBCI, proyecto que se dedica a la venta de hardware para la realización de EEG, así como otros ensayos. Esta herramienta está formada por un microprocesador, un módulo de memoria SD para el almacenamiento de datos, y un módulo de comunicación inalámbrica que transmite los datos por Bluetooth. Además cuenta con el mencionado ADC ADS1299. Esta plataforma ofrece una interfaz gráfica que sirve para realizar la investigación previa al diseño del sistema de procesamiento, al permitir tener una primera toma de contacto con el sistema.
2. La segunda plataforma consiste en un kit de evaluación para el ADS1299, desde la cual se pueden acceder a los diferentes puertos de control a través de los pines de comunicación del ADC. Esta plataforma se conectará con la FPGA en el sistema integrado.

Para entender cómo funcionan las ondas más simples del cerebro, así como saber cuáles son los requisitos mínimos en el análisis de ondas EEG se realizaron diferentes consultas con el Dr Ceferino Maestu, neurofisiólogo del Centro de Tecnología Biomédica (CTB) de la UPM. Él se encargó de introducirnos en los distintos procedimientos en el análisis de ondas en electroencefalogramas, así como la forma en que se deben de colocar los electrodos en el cráneo.

Para terminar con la investigación previa, se realiza en MATLAB un primer modelo de procesamiento de los datos. Una característica muy importante de las ondas cerebrales es la aleatoriedad de las mismas, de forma que el análisis en el dominio del tiempo se vuelve muy complejo. Por ello, el paso más importante en el procesamiento de los datos es el paso del dominio temporal al dominio de la frecuencia, mediante la aplicación de la transformada rápida de Fourier o FFT (Fast Fourier Transform), donde se pueden analizar con mayor precisión los datos recogidos.

El modelo desarrollado en MATLAB se utiliza para obtener los primeros resultados del sistema de procesamiento, el cual sigue los siguientes pasos.

1. Se captan los datos desde los electrodos y se escriben en una tabla de datos.
2. Se leen los datos de la tabla.
3. Se elige el tamaño temporal de la muestra a procesar.
4. Se aplica una ventana para evitar las discontinuidades al principio y al final del bloque analizado.
5. Se completa la muestra a convertir con *zero-padding* en el dominio del tiempo.
6. Se aplica la FFT al bloque analizado con ventana y *zero-padding*.
7. Los resultados se llevan a una gráfica para ser analizados.

Llegados a este punto, se observa que la captación de ondas alfas resulta muy viable. Aunque es cierto que se presentan ciertos problemas a la hora de interpretar los datos debido a la baja resolución temporal de la plataforma de OpenBCI, este es un problema que se soluciona en el modelo desarrollado, al permitir el kit de evaluación (sistema de captación de datos) actuar sobre la velocidad de captación de los datos, es decir la frecuencia de muestreo, lo que afectará directamente a esta precisión.

Una vez llevado a cabo el primer procesamiento y su posterior análisis de los resultados obtenidos, se procede a realizar un modelo en Hardware que siga los mismos pasos que el

desarrollado en MATLAB, en la medida que esto sea útil y viable. Para ello se utiliza el programa XPS (Xilinx Platform Studio) contenido en la herramienta EDK (Embedded Development Kit), que nos permite diseñar un sistema embebido. El sistema de procesamiento de datos desarrollado en este TFG se muestra en la ilustración I.1.

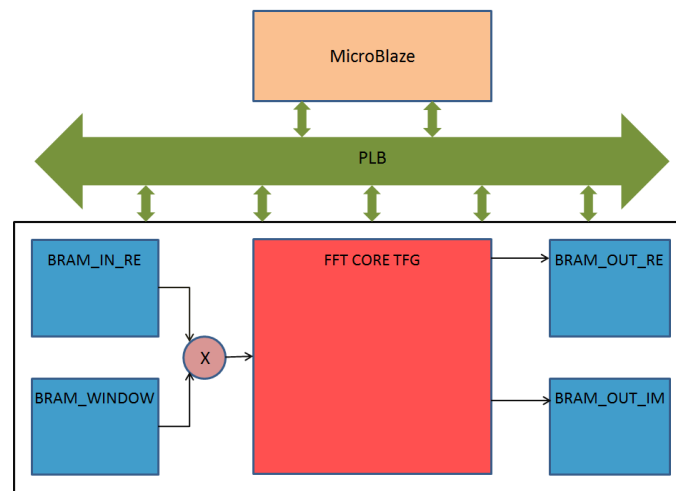


Ilustración I.1 Sistema de procesamiento de datos diseñado en Hardware.

Este sistema cuenta con:

- Un microprocesador de tipo soft-core llamado MicroBlaze, que se encarga de gestionar y controlar todo el sistema.
- Un bloque FFT que se encarga de realizar la transformada rápida Fourier.
- Cuatro bloques de memoria BRAM, donde se almacenan los datos de entrada y salida del bloque FFT y un multiplicador para aplicar la ventana a los datos de entrada al bloque FFT.
- Un bus PLB, que consiste en un bus de control que se encarga de comunicar el MicroBlaze con los diferentes elementos del sistema.

Tras el diseño Hardware se procede al diseño Software utilizando la herramienta SDK (Software Development Kit). También en esta etapa se integra el sistema de captación de datos, el cual se controla mayoritariamente desde el MicroBlaze. Por tanto, desde este entorno se programa el MicroBlaze para gestionar el Hardware que se ha generado. A través del Software se gestiona la comunicación entre ambos sistemas, el de captación y el de procesamiento de los datos. También se realiza la carga de los datos de la ventana a aplicar en la memoria correspondiente.

En las primeras etapas de desarrollo del sistema, se comienza con el testeo del bloque FFT, para poder comprobar el funcionamiento del mismo en Hardware. Para este primer ensayo, se carga en la BRAM los datos de entrada al bloque FFT y en otra BRAM los datos de la ventana aplicada. Los datos procesados saldrán a dos BRAM, una para almacenar los valores reales de la transformada y otra para los imaginarios. Los resultados obtenidos son satisfactorios como se puede ver en la ilustración I.2.

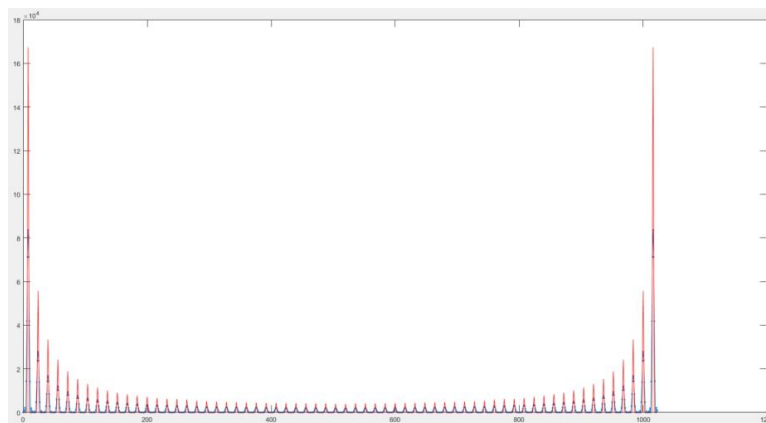


Ilustración I.2 FFT de una onda cuadrada. En el eje de abscisas se representa la frecuencia en Hz y en el eje de ordenadas las unidades de tensión empleadas por el ADS1299 (1 unidad= 0.54 uV). También se puede ver que

Tras comprobar el correcto funcionamiento del bloque FFT, se integra junto al sistema de adquisición de datos. Posteriormente se procede a realizar un ensayo de EEG real, para captar ondas alfa. Tras realizar el experimento, se llega a los resultados que se pueden ver en la ilustración I.3.

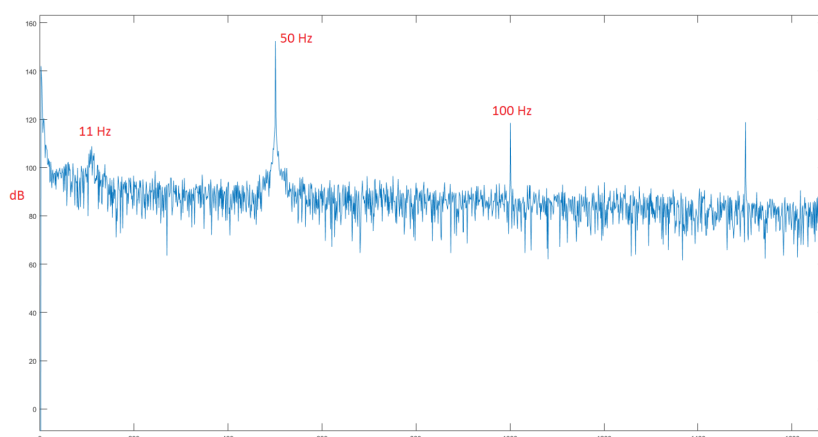


Ilustración I.3 Aplicación del sistema desarrollado sobre un EEG real. Se puede ver la frecuencia de ondas alfa a 11 Hz.

Como se puede observar en la figura, aparece un pico en la frecuencia de los 11 Hz, por lo que el ensayo se ha realizado con éxito y se consigue el objetivo principal del TFG.

Por otro lado, y para validar el uso de las FPGAs como unidades ideales de procesamiento, se realiza una medición del tiempo que tarda el bloque FFT en realizar la transformada. Este tiempo se compara con el tiempo que tarda MATLAB en realizar la misma transformada a los mismos datos. Se comparan los resultados y se llega a lo mostrado en la tabla I.4.

BLOQUE FFT	MATLAB
<pre>*** START *** Procesando Hecho. 99 us (12399 ciclos)</pre>	<pre>>> in_re = eeg(:,3); >> tic; out = fft(in_re); toc Elapsed time is 0.002698 seconds.</pre>

Tabla I.1 A la izquierda el tiempo de ejecución del procesamiento en el bloque FFT. A la derecha, en MATLAB.

$$Ganancia = \frac{Tiempo\ Software}{Tiempo\ Hardware} = \frac{0.002698\ s}{99 * 10^{-6}\ s} = 27.25$$

Esto significa que el sistema desarrollado en Hardware realiza la transformada rápida de Fourier 27 veces más rápido que lo que tarda MATLAB, por lo que se puede ver aquí la gran ventaja competitiva del Hardware en lo que a tiempos de ejecución se refiere.

En lo que al aspecto didáctico se refiere, este TFG engloba diferentes campos. En el campo de la electrónica:

- Se han mejorado los conocimientos en MATLAB, así como diferentes herramientas que ofrece como FDATool (Filter Design Analysis Tool).
- Se han adquirido conocimientos de técnicas de procesamiento de señal, y en particular, de análisis espectral.
- Se han mejorado los conocimientos en VHDL, así como su uso en el entorno ISE de Xilinx.
- Se han reforzado los conocimientos en C mediante la programación del MicroBlaze para el control del sistema.
- Se ha aprendido a crear sistemas embebidos usando el entorno de desarrollo de Xilinx usando la herramienta EDK (Embedded Development Kit).

En el campo de la neurología, se ha aprendido a realizar ensayos EEG, así como a analizar e interpretar los resultados mostrados en el mismo.

En cuanto al impacto social, los sistemas BCI afectan a muchos sectores, donde destaca el volumen de personas con discapacidades físicas, para los cuales, este sistema implica una oportunidad de aumentar su autonomía en el día a día. También otro sector importante es el sector de la investigación médica, donde los sistemas BCIs son aplicables en muchas aplicaciones como, por ejemplo, la detección y estudio de enfermedades cognitivas

INDICE

Contenido

AGRADECIMIENTOS	1
RESUMEN	3
INDICE	9
1. INTRODUCCIÓN	¡Error! Marcador no definido.
1.1 MOTIVACIÓN Y OBJETIVOS	13
1.2 PLANTEAMIENTO INICIAL Y ALCANCE	14
1.3 ESTRUCTURA DEL TFG	16
2. ESTADO DEL ARTE Y MARCO TEÓRICO.	19
2.1 ESTADO DEL ARTE	19
2.2 ONDAS CEREBRALES	20
2.2.1 TIPOS DE ONDAS CEREBRALES	20
2.3 ANALISIS DE PROCESAMIENTO DE DATOS	21
2.3.1 VENTANAS	21
2.3.2 TRANSFORMADA RAPIDA DE FOURIER	23
2.3.3 ZERO PADDING	24
2.3.4 FILTRADO DIGITAL	25
2.3.5 TEOREMA DE NYQUIST	26
2.3 APORTACIÓN AL TFG	26
3. INVESTIGACIÓN PREVIA	31
3.1 CONFIGURACIÓN Y DISEÑO DEL EXPERIMENTO	32
3.2 RECEPCIÓN Y ALMACENAMIENTO DE DATOS	33
3.3 ANALISIS DE SEÑALES EN MATLAB	34
3.3.1 ANALISIS EN FRECUENCIA	34
3.3.2 ANALISIS EN TIEMPO	35
3.3.3 ANALISIS EN ESPACIO	36
3.4 CONCLUSIÓN FINAL A LA INVESTIGACIÓN PREVIA.	36
4. CALCULO, DISEÑO Y DESARROLLO DEL SISTEMA	41
4.1 ELEMENTOS PARA EL DESARROLLO	41
4.2 DISEÑO DEL BLOQUE FFT EN HARDWARE	42
4.2.1 POSIBILIDADES DEL BLOQUE FFT	42

4.2.2 CONFIGURACION DEL BLOQUE FFT	44
4.2.3 TIMING DEL BLOQUE FFT	46
4.3 DISEÑO HARDWARE DEL SISTEMA	47
4.3.1 SISTEMA MICROPROCESADOR.....	48
4.3.1 MEMORIAS BRAM.....	48
4.3.2 VENTANA	50
4.3.3 INTEGRACION DEL SISTEMA DE ADQUISICION DE DATOS	50
4.4 DISEÑO SOFTWARE DEL SISTEMA.....	51
4.5 ASPECTOS IMPORTANTES DEL SISTEMA	51
5. OBTENCIÓN Y ANÁLISIS DE RESULTADOS	55
5.1 CAPTACIÓN DE SEÑAL	55
5.2 ANÁLISIS DE UNA ONDA CUADRADA	57
5.3 PROCESAMIENTO DE EEG REAL.....	58
5.4 VELOCIDAD DE PROCESAMIENTO	59
6. CONCLUSIONES Y LINEAS FUTURAS.....	63
6.1 CONCLUSIONES	63
6.2 LINEAS FUTURAS	65
7. IMPACTOS Y ASPECTOS DE RESPONSABILIDAD LEGAL, ETICA Y PROFESIONAL	69
7.1 RESPONSABILIDAD ETICA Y PROFESIONAL	69
7.2 IMPACTOS PROFESIONALES.....	69
7.3 IMPACTOS ÉTICOS	70
8. PLANIFICACION TEMPORAL Y PRESUPUESTO.....	73
8.1 PLANIFICACIÓN TEMPORAL.....	73
8.2 DIAGRAMA DE GANTT.....	74
8.3 PRESUPUESTO	75
8.3.1 PRESUPUESTO EN HARDWARE.....	75
8.3.2 PRESUPUESTO EN SOFTWARE	75
8.3.3 SALARIOS.....	76
9. REFERENCIAS	77
10. INDICE DE ILUSTRACIONES.....	79
11. ABREVIATURAS Y ACRÓNIMOS	81
ANEXO A CÓDIGO IMPLEMENTADO EN C EN EL MICROBLAZE	82
ANEXO B: ENSAYOS EN MATLAB EN LA INVESTIGACIÓN PREVIA.	84
ANEXO C MODELO DEFINITIVO XPS.	85

Capítulo 1

Introducción

1. INTRODUCCIÓN

1.1 MOTIVACIÓN Y OBJETIVOS

En el desarrollo de una interfaz cerebro-máquina, también conocida como BCI (Brain Computer Interface), se vuelve esencial el procesamiento que se le da a las señales bioeléctricas captadas. La alta velocidad de procesamiento de las FPGAs (Field Programmable Gate Array) y su bajo consumo energético hacen de las mismas el dispositivo ideal para desarrollar un sistema de procesamiento de datos para realizar ensayos de tipo EEG (Electroencefalograma).

En el CEI (Centro de Electrónica Industrial), donde se lleva a cabo este TFG, se dispone de experiencia en el diseño de sistemas reconfigurables en FPGAs. Este TFG es el segundo de una línea de proyectos en la cual se busca obtener un sistema capaz de adquirir y procesar correctamente señales cerebrales, para llegar a un patrón común que nos permita, por ejemplo, detectar cuando una persona está quedándose dormida, de forma que se pueda actuar en consecuencia despertándola si fuese necesario.

Este TFG se centra en el diseño y desarrollo de un sistema de procesamiento de datos. Este sistema recibe los datos transmitidos por SPI entre una FPGA y un sensor de señales bioeléctricas ADS1299. Este sensor se adapta a diferentes potenciales: potenciales del cerebro humano (EEG) y potenciales cardiovasculares (ECG), por lo que se utiliza en instrumentación médica.

En este TFG se trabaja con potenciales del cerebro que se captan por electroencefalografía, esto es, mediante la colocación de una serie de electrodos de una forma determinada en la cabeza, cercanos a la corteza cerebral.

En los últimos años se ha experimentado un gran crecimiento en aplicaciones que están relacionadas con la medida de señales biométricas, más en concreto con aquellos sistemas biológicos en los cuales, mediante el análisis de dichas señales biométricas, permiten detectar ciertos patrones que están directamente relacionados con eventos del cuerpo humano o de la mente humana. De entre todas estas señales, las señales eléctricas necesitan de un procesamiento más meticuloso. Esto se debe a al alto número de problemas que se presenta en el análisis de dichas señales. Uno de los problemas que se presentan es el ruido electromagnético que proviene de la red a 50 Hz, al cual estamos expuestos constantemente y el cual afectará directamente a los experimentos.

En la actualidad existe una alta demanda en sistemas relacionados con el cerebro humano mediante la interfaz cerebro-máquina (*Brain Computer Interface*), por lo que el crecimiento y desarrollo de estos sistemas ha aumentado de forma significativa [1]

Por otro lado, el uso de hardware evolutivo mediante reconfiguración por hardware demuestra una alta capacidad a la hora de filtrar y procesar señales, debido a una mayor velocidad de procesamiento, un bajo consumo y una mayor adaptabilidad. Esto hace que resulte lógico relacionar y combinar un sistema BCI con un sistema de reconfiguración basado en hardware evolutivo.

Esta tarea se simplifica gracias al circuito integrado ADS1299, desarrollado por Texas Instruments, especialmente diseñado para captar señales bioeléctricas con una precisión del orden de microvoltios, gracias a subsistemas de filtrado y eliminación del ruido. Este kit deja accesible un puerto SPI, a partir del cual se realizan las comunicaciones con una FPGA Virtex5 que realizará el procesamiento. A su vez este kit cuenta con un software que nos permite entre otras cosas, ver el valor de los diferentes registros, así como manipular a nuestro antojo y a más bajo nivel el ADS1299. De esta forma se puede emular la secuencia de comunicación para controlar en el posterior diseño de forma más fiable los datos a enviar, así como comprender más fácilmente el porqué de ciertos valores en ciertos registros que nos serán útiles en el procesamiento de los datos.

El objetivo final de este proyecto es la implementación de un sistema que sea capaz de realizar el procesamiento de los datos captados por el ADS1299 en una FPGA Virtex5, y como aplicación de esto detectar un tipo de ondas cerebrales que permitirán saber a priori si una persona se está quedando dormida o no.

Desde el punto de vista del software, se utiliza la herramienta SDK (Software Development Kit) que facilita la empresa Xilinx. Este software permite controlar un microprocesador interno que tiene la FPGA, desde el que se controlan las diferentes memorias o los diferentes bloques que se utilizan en el sistema. A este microprocesador interno se le llama MicroBlaze.

Desde el punto de vista del hardware, el objetivo es que este se encargue de aplicar un algoritmo conocido como transformada rápida de Fourier (FFT) a los datos captados mediante el ADS1299 y almacenados en diferentes memorias internas BRAM de la FPGA. Este algoritmo se utiliza en el campo de procesamiento digital de señales para pasar del dominio del tiempo al dominio de la frecuencia.

La validación del cumplimiento de los objetivos se realizará mediante experimentación, esto es, captando datos reales y procesándolos en la FPGA. Posteriormente se comparará entre el procesamiento de los datos realizados en la FPGA y el procesamiento de los mismos datos realizado mediante MATLAB. En ambos procedimientos se debe de llegar al mismo resultado.

1.2 PLANTEAMIENTO INICIAL Y ALCANCE

El planteamiento que se sigue tiene en cuenta que dentro del sector del BCI existen distintos niveles de complejidad a la hora de estudiar las ondas cerebrales. Es por ello necesario analizar en las condiciones del experimento (esto es, detectar un tipo de onda cerebral), cuáles son las especificaciones más exigentes en los análisis de electroencefalografía (EEG).

La placa comercial de la que se dispone en el CEI es una placa desarrollada por OpenBCI [2]. OpenBCI es un proyecto, basado en crowdfunding que desarrolla y vende diferentes placas para ser usadas en BCIs. Esta placa cuenta con una unidad de comunicación inalámbrica por Bluetooth. La velocidad de captación del módulo Bluetooth impide captar datos en tiempo real a una frecuencia superior a 250 Hz, la frecuencia de muestreo se

vuelve limitante en nuestro sistema, de forma que no permite captar datos a más frecuencia lo que repercute directamente en una peor precisión que si por ejemplo, se captaran los datos a 500 Hz o 1 KHz. Por ello y para solucionar este problema, se procede a trabajar con el kit de evaluación de Texas Instruments, en el cual si se permite modificar el registro que controla la frecuencia de muestreo. Esta parte fue desarrollada en otro TFG

A raíz del aumento de la frecuencia de muestreo, se diseña un bloque para realizar la FFT a un tamaño de muestra igual o superior a dicha frecuencia, para disponer de esta forma de mayor margen de análisis en el dominio del tiempo

El desarrollo del bloque diseñado se realizará en las siguientes etapas:

1. Investigación de la placa de OpenBCI. Se realizará un modelo en MATLAB que permita procesar los datos almacenados en una hoja de cálculo.
2. Análisis de los resultados obtenidos.
3. Estudio del funcionamiento y las características de la FFT, así como entornos de aplicación.
4. Diseño de un bloque para la realización de la FFT en Hardware, utilizando la herramienta CoreGenerator que facilita Xilinx. CoreGenerator es una herramienta que permite crear un bloque en Hardware a partir de unos parámetros que definen dicho bloque.
5. Estudio del funcionamiento y las características de los diferentes tipos de ventanas, así como sus entornos de aplicación.[Referencia al apartado de las ventanas]
6. Diseño y desarrollo de todo el sistema hardware mediante XPS (Xilinx Platform Studio), incluido en el EDK (Embedded Development Kit) de Xilinx. XPS es un programa de Xilinx que permite estudiar y manipular las diferentes partes de un sistema en Hardware, es decir, sus buses, puertos de E/S, memorias, etc
7. Integración del sistema de captación de datos con el bloque FFT y cuatro memorias de tipo BRAM, las cuales se encargaran de almacenar datos, pre y post procesamiento.
8. Simulación del modelo hardware desarrollado en el entorno ISIM de Xilinx.
9. Programación software del sistema para su control. Para ello se utiliza el entorno de programación SDK (Software Development Kit) también incluido en el EDK.

Una vez implementado el sistema se realiza una comparación entre los resultados obtenidos mediante el procesamiento en Hardware y el obtenido mediante MATLAB.

El alcance del proyecto se define en los siguientes puntos:

- Análisis de los estudios basados en la electroencefalografía (EEG) y adecuación de las mejores condiciones al sistema desarrollado.
- Diseño de un sistema embebido mediante el uso de Embedded Development Kit (EDK) de Xilinx.
- Estudio del funcionamiento de la Transformada Rápida de Fourier (FFT) y sus aplicaciones en sistemas BCI.
- Obtención de resultados experimentales.

- Análisis de la eficiencia y rendimiento del sistema desarrollado tomando como referencia el mismo experimento en MATLAB.
- Detección de ondas alfa en Hardware mediante el sistema embebido desarrollado como aplicación.

1.3 ESTRUCTURA DEL TFG

En el resto del documento se encuentran los siguientes capítulos:

- Marco teórico y estado del arte. En este apartado se introduce la teoría básica de ondas cerebrales, así como la teoría básica para realizar procesamiento digital de señales, donde destaca la transformada rápida de Fourier o FFT (Fast Fourier Transform) y el uso de ventanas temporales.
- Investigación previa. En este apartado se realiza una investigación previa para analizar las ventajas y desventajas de la plataforma OpenBCI. Se realizan una serie de ensayos para analizar la viabilidad del modelo de comunicación inalámbrico entre el ADS1299 y la FPGA. También se comentan los diferentes tipos de análisis que se tienen en cuenta para afrontar ensayos de EEG. Por último se realiza un ensayo real de EEG captando ondas cerebrales alfa.
- Calculo, diseño y desarrollo del sistema. En este capítulo se detalla toda la secuencia de procesos seguidos para calcular, diseñar y desarrollar el sistema de procesamiento de datos en la FPGA. Además también se detalla la integración con el sistema de adquisición de datos desarrollado en otro TFG, así como aspectos fundamentales del sistema.
- Obtención y análisis de resultados. Se muestran una serie de resultados para validar el correcto funcionamiento del sistema diseñado. A su vez se muestran los resultados de un ensayo real EEG de captación de ondas cerebrales alfa con el sistema desarrollado.
- Conclusiones y líneas futuras. Después del análisis de los resultados, se realiza una reflexión sobre la funcionalidad del sistema. También se mencionan una serie de proyectos a corto y largo plazo .
- Impactos y aspectos de responsabilidad legal, ética y profesional. Se analizan los diferentes impactos que repercuten en este TFG.
- Planificación y presupuesto. Se detalla la planificación temporal y el presupuesto del TFG.

Capítulo 2

Estado del Arte y Marco
Teórico

2. ESTADO DEL ARTE Y MARCO TEÓRICO.

2.1 ESTADO DEL ARTE

Con el paso de los años, se han ido desarrollando cada vez más sistemas que son capaces de captar, procesar y analizar diferentes señales que han sido generadas por el cuerpo humano. Dentro de estas señales, destacan las basadas en el cerebro humano, esto es, las que se captan por electroencefalografía. Este tipo de señales cerebrales, se pueden clasificar atendiendo a la diferente funcionalidad que presentan. Por tanto puede tratarse de procesos cognitivos, ondas motoras, emocionales o de percepción.

En el proceso evolutivo relacionado con los BCI, son muchas las aplicaciones y dispositivos que han surgido. Entre ellas destacan las aplicaciones que permiten controlar otro dispositivo, formando un sistema embebido. Ejemplos de dispositivos comerciales embebidos de tipo BCI son: Emotiv EPOC [3], Kokoon EEG [4] y hasta incluso BCI para mascotas, como es NoMoreWoof [5].

Separándose de la rama comercial, también existen diferentes plataformas dedicadas al ámbito de la investigación. Entre ellas destacan las aplicaciones con fines médicos, como por ejemplo las que permiten controlar sillas de ruedas a parálíticos [6]

También hay aplicaciones en el mundo de los videojuegos y en el de la realidad virtual [7].

La estructura de funcionamiento de un sistema tipo BCI se muestra en la siguiente ilustración:

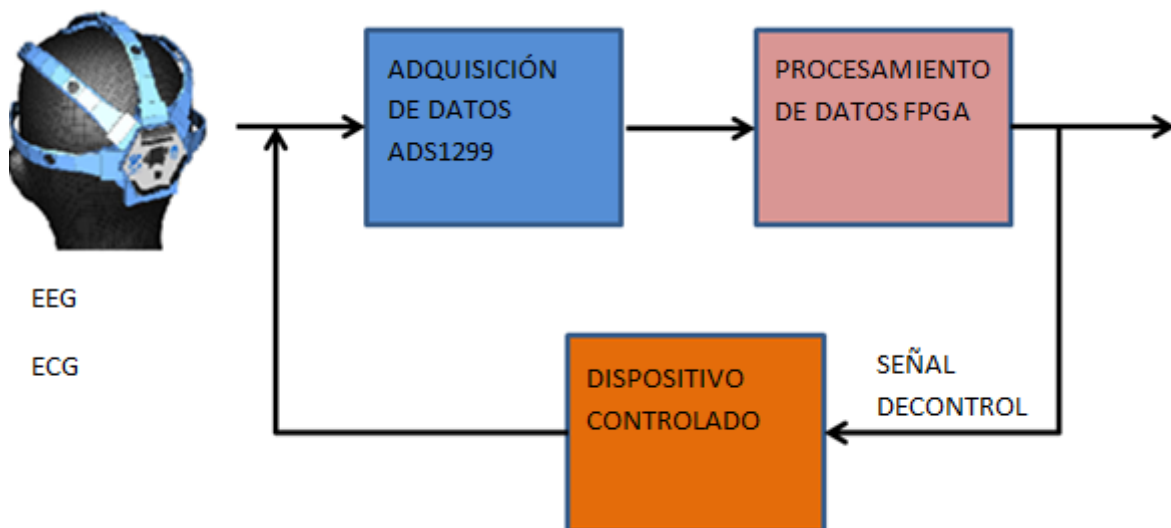


Ilustración 2.0.1 Funcionamiento sistema BCI . En el caso de este TFG, los datos son captados por el ADS1299 y procesados por una FPGA Virtex 5.

Ilustración 2.1

Inicialmente se captura la señal mediante la colocación de los electrodos en su lugar correspondiente.). Estas señales son adquiridas por el ADS1299. Es importante recalcar el papel que realiza el conversor analógico digital en esta fase, ya que este ADC (Analogic-Digital Converter) está especialmente diseñado para medir biopotenciales. Esto se realiza registrando y digitalizando la tensión de entrada. Estas señales son posteriormente procesadas por una FPGA Virtex5, para después relacionar cierto patrón con una señal de control. Finalmente se retroalimenta la señal para asegurar la veracidad de los resultados conseguidos.

Por otro lado, el uso de Hardware evolutivo para técnicas de clasificación presenta muchas ventajas. Más concretamente, se proveen a los clasificadores más destacados y al EHW de las mismas características extraídas de ondas obtenidas por electromiograma (EMG). Los resultados experimentales demuestran que el EHW es capaz de competir con el estado del arte de los tres métodos clasificadores más destacados. [8].

Respecto al procesamiento en Hardware, en la universidad tecnológica de Tallinn ,Estonia ,se está implementando en FPGAs un prototipo para analizar EEG. En este caso, se analizan los EEG en tiempo real para diagnosticar enfermedades de desorden mental. A diferencia de en este proyecto en el que se realiza la FFT, el procesamiento es realizado siguiendo un algoritmo conocido como SASI [9]. En la universidad de Belfast, sin embargo, si utilizan la FFT para realizar procesamiento de imágenes en FPGAs [10].

2.2 ONDAS CEREBRALES

En el cerebro humano se producen impulsos eléctricos que viajan a través de las neuronas produciendo ritmos que se conocen como ondas cerebrales. Estos impulsos son información que viaja a través de las neuronas para ejecutar una función concreta. Esta actividad se puede registrar mediante electroencefalografía (EEG).

Estas ondas están relacionadas con diferentes estados de la conciencia, como son la concentración intensa, estados de sueño profundo, alerta, relajación, hipnosis, etc...

2.2.1 TIPOS DE ONDAS CEREBRALES

Existen cuatro tipos de ondas cerebrales, que se clasifican dependiendo de la frecuencia que presenten. Estas ondas, ordenadas de mayor a menor actividad, son:

- Beta: las ondas beta se producen cuando el cerebro se encuentra despierto por lo que implican una actividad mental intensa. Son las que presentan mayor actividad de los cuatro tipos, y su frecuencia oscila entre valores de 14 Hz y 30 Hz.
- Alfa: las ondas alfa son más lentas y presentan mayor amplitud que las ondas beta. Su frecuencia oscila entre valores de 8 Hz y 13 Hz. Su aparición conlleva un estado de relajación y una menor actividad cerebral.
- Theta: las ondas teta son a su vez más lentas y presentan mayor amplitud que las ondas alfa. Su frecuencia oscila entre 4 Hz y 7 Hz. Representan un estado de calma profundo y de inspiración de ideas y soluciones creativas.

- Delta: las ondas delta son las ondas de mayor amplitud y menor actividad. Su frecuencia oscila entre 1.5 Hz y 3.5 Hz. El cerebro siempre genera una cierta cantidad de ondas delta, de lo contrario significaría la muerte cerebral. Son las ondas que se generan en un estado de sueño profundo.

En la siguiente ilustración se puede ver como las ondas van aumentando su amplitud según va disminuyendo su frecuencia.

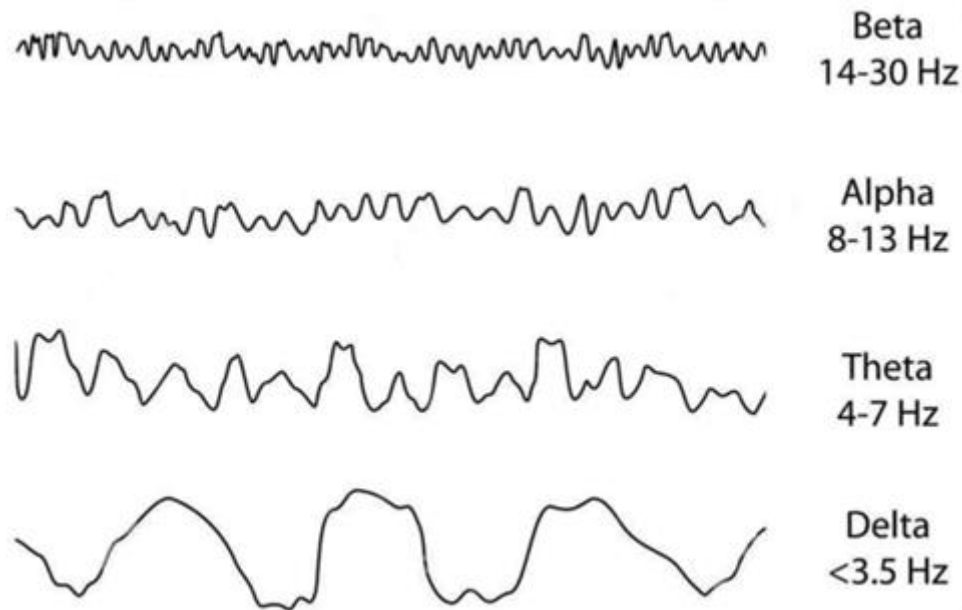


Ilustración 2.2 Tipos de ondas cerebrales e intervalo de frecuencias de las mismas.

En este TFG, se trabajará con ondas cerebrales alfa, que por su mayor amplitud y su acotado rango de frecuencias que las ondas beta, se vuelven relativamente fáciles de detectar. Como ya se ha mencionado anteriormente, estas ondas nos indicarían a priori cuando una persona ha entrado en un estado de relajación mental. Por otro lado, captar ondas Theta e incluso Delta se vuelve muy difícil en el entorno de trabajo en el que se desarrolla este TFG, lo que hace de las ondas alfa las ideales para empezar a introducirse en esta especialidad.

2.3 ANALISIS DE PROCESAMIENTO DE DATOS

En este apartado se cuenta lo esencial en el procesamiento digital de señales seguido en este proyecto. La señal, adquirida por el ADS1299, llega a la FPGA Virtex5 donde se aplica inicialmente una ventana (Referencia al apartado 2.2.1). A las muestras ventaneadas se le aplica la transformada rápida de Fourier (Ver apartado 2.2.2) para obtener la señal en el dominio de la frecuencia (Ver apartado 3, donde hablo de dominio frecuencia amplitud), donde será más fácil de analizar y encontrar un patrón que nos dé información útil.

2.3.1 VENTANAS

Las ventanas son funciones matemáticas usadas en el análisis y procesamiento de señales para evitar las discontinuidades al principio y al final de los bloques analizados. Cuando se usa la FFT para medir la componente en frecuencia de una señal, se basan los análisis en que la señal está formada por un número entero de periodos, es decir, se repite un número entero de veces por lo que el punto inicial y final de la muestra estarían unidos.

Sin embargo, cuando el número de periodos no es entero, o cuando, simplemente, la señal no es periódica, el principio de la muestra y el final no son el mismo, por lo que al pasar los datos por la FFT, en el dominio de la frecuencia aparecerán componentes a alta frecuencia.

En procesamiento de señales, una ventana se utiliza cuando ocurre esto, y a efectos de reducir esas componentes a alta frecuencia. Para ello se atenúa la amplitud en las discontinuidades de la muestra. El efecto de aplicación de una ventana se muestra en la ilustración.[11]

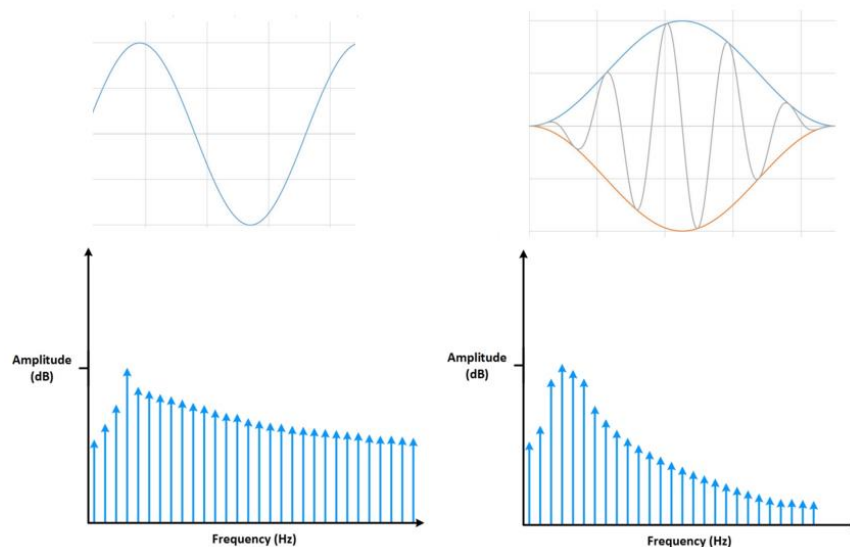


Ilustración 2.3 Efecto de la ventana. A la izquierda, muestra con número de periodos no entero sin aplicación de ventana. A la derecha, misma muestra con ventana. Se puede ver la atenuación de las altas frecuencias debidas a la discontinuidad.

2.3.1.1 TIPOS DE VENTANAS

Alguno de los tipos de ventana empleados habitualmente son(ver ilustración :

- Rectangular: es equivalente a no aplicar ventana, por lo que no consigue atenuar las frecuencias debido a los bordes.
- Triangular: esta ventana suaviza el efecto de los bordes pero no lo suficiente, por lo que no consigue atenuar correctamente estas frecuencias.
- Hann: esta ventana tiene forma senoidal y se emplea en análisis de frecuencia de señales
- Flat top: se caracteriza por tener un ancho de banda superior a la Hann y se usa en sistemas de calibración. Se puede observar que presenta valores negativos.

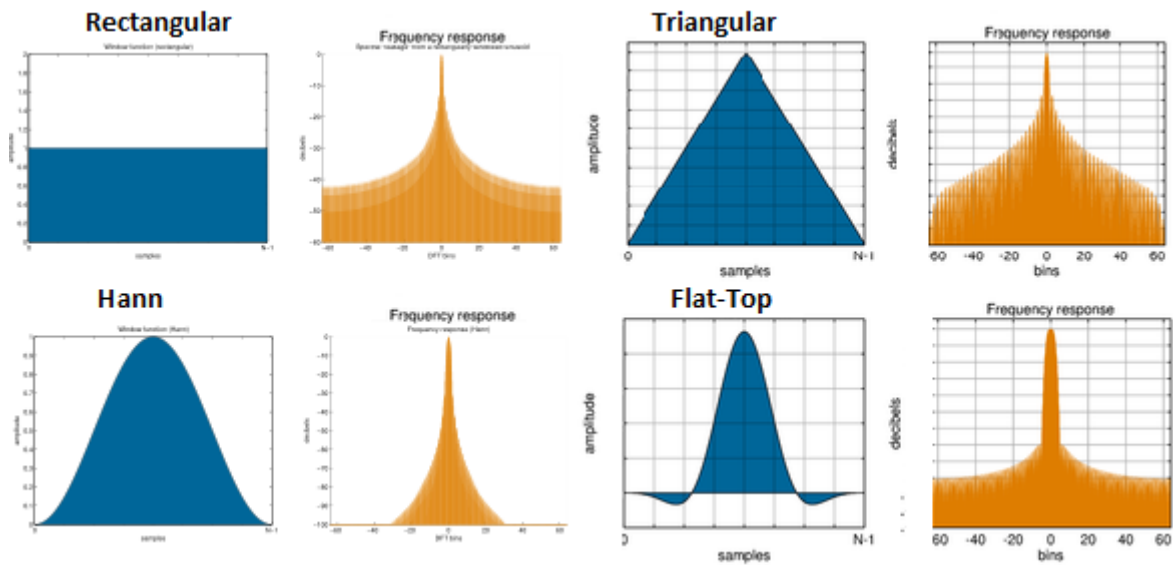


Ilustración 2.4 Tipos de ventana y su respuesta en frecuencia.

En este TFG, la ventana que se utilizará es la Hann. Esto se debe a las ventajas que presenta la misma en el procesamiento digital de señales. Principalmente consigue atenuar muy bien las altas frecuencias aunque reduzca ligeramente la resolución.

2.3.2 TRANSFORMADA RÁPIDA DE FOURIER

La FFT (Fast Fourier Transform) es una herramienta fundamental que se utiliza, entre otros muchos campos, en el procesamiento digital de señales. Se trata de un algoritmo que realiza los cálculos necesarios para pasar una señal del dominio del tiempo al dominio de la frecuencia.

Las ondas que se desarrollan en el cerebro se distinguen, entre otras cosas, por su frecuencia [Ver apartado 2.2.1]. Por ello, obtener la señal en el dominio de la frecuencia es sumamente importante en sistemas BCI, al facilitar la interpretación de ciertas acciones realizadas por el sujeto portador del BCI. Esto es así ya que el cerebro produce simultáneamente ondas en todo el espectro de la frecuencia, por lo que en el dominio del tiempo será sumamente difícil distinguir una acción cuya componente, por ejemplo, sea de 20 Hz. Sin embargo, en el dominio de la frecuencia, se apreciara cierto pico en dicha frecuencia, con lo que se puede distinguir claramente que se trata de una señal con una componente en frecuencia más intensa en esa frecuencia.

Aunque a efectos prácticos de este TFG, no ha resultado necesario conocer los detalles de funcionamiento del algoritmo que realiza la FFT, la base matemática viene resumidamente explicada en el apartado 2.3.2.1 TEORÍA DE OPERACIÓN DE LA FFT.

2.3.2.1 TEORÍA DE OPERACIÓN DE LA FFT

La FFT es un algoritmo computacionalmente eficiente para calcular una transformada discreta de Fourier de una muestra cuyo tamaño es una potencia positiva de 2. Por un lado, la transformada discreta de Fourier se define como:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-\frac{jnk2\pi}{N}} \quad k=0, \dots, N-1$$

donde N es el tamaño de la muestra y $j = \sqrt{-1}$. La transformada inversa de Fourier, IFDT (Inverse Fourier Discrete Transform) es dada por:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{\frac{jnk2\pi}{N}} \quad n=0, \dots, N-1$$

Es importante destacar que la FFT no es una transformada diferente sino que se trata de un algoritmo eficaz para el cálculo de la DFT (Discrete Fourier Transform). La importancia de la FFT radica en que esta elimina parte de los cálculos repetitivos que la DFT realizaba, por lo que se logra un cálculo de la serie más rápido. Por otro lado, la FFT consigue una mayor precisión que la DFT al conseguir disminuir los errores de redondeo.

2.3.3 ZERO PADDING

El zero padding es una técnica que se utiliza en procesamiento de señales con el objetivo de aumentar la resolución de la señal captada. Dependiendo de dónde se añadan los ceros, se hablará de padding en el tiempo, o padding en la frecuencia.

El padding en el tiempo consiste en la adición de ceros al final de la muestra captada, de forma que cuando dicha muestra se pase al diagrama de la frecuencia, ésta aparece definida por más puntos en el dominio de la frecuencia, ya que la transformada discreta de Fourier de una señal de N puntos en el dominio del tiempo se transforma en una señal de N/2 puntos en la frecuencia(en realidad , N/2 senos y N/2 cosenos).

Análogamente, si los ceros se añaden en el diagrama de la frecuencia, al pasar al diagrama del tiempo la señal tendrá más resolución en el dominio del tiempo, por lo que se puede realizar una interpolación sin modificar el comportamiento frecuencial de la señal interpolada.

Hay que recalcar que esta técnica es útil para un determinado número de ceros. Si añadimos pocos ceros, es posible que no se consiga aumentar en nada la resolución, siendo la técnica completamente inútil .Por otro lado, si se excede el número de ceros, se podría llegar a distorsionar la precisión real de la señal captada al realizarse la interpolación con excesivos ceros. Es por ello que hay que buscar añadir la cantidad óptima de ceros.

No obstante, y en lo que respecta a los experimentos desarrollados en este TFG, una frecuencia de muestreo superior a 250 Hz (tras eliminar la limitación impuesta por el Bluetooth) dará suficiente precisión como para no tener que realizar zero padding.

2.3.4 FILTRADO DIGITAL

En el procesamiento de señales puede resultar muy interesante eliminar o atenuar una gama de frecuencias o una frecuencia concreta que solo añade o suponga ruido a la muestra. Esta frecuencia se elimina con el uso de un filtro que puede modificar tanto amplitud como fase.

2.3.4.1 TIPOS DE FILTROS

Existe una multitud de filtros que se pueden utilizar dependiendo de la frecuencia que se quiera atenuar. Los principales son:

- Notch: se utiliza para atenuar una frecuencia concreta. La frecuencia es atenuada por un valor definido en el filtro, normalmente 60 dB.
- Pasa banda: este filtro deja sin atenuar una gama de frecuencias que define el usuario, atenuando el resto de las frecuencias del espectro normalmente un valor de 60 dB.
- Paso bajo: este filtro atenúa las altas frecuencias, normalmente 60 dB, dejando pasar las frecuencias más bajas.
- Paso alto: exactamente igual que el paso bajo, pero dejando pasar las frecuencias más altas y atenuando las bajas.

En la siguiente ilustración se muestran ejemplos de los tipos de filtros anteriormente descritos:

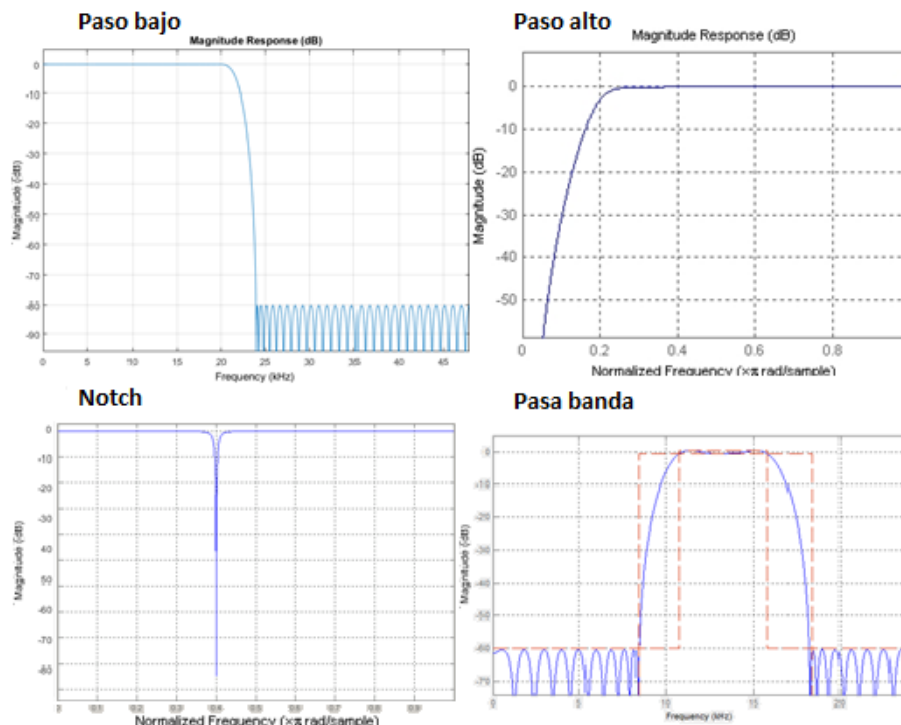


Ilustración 2.5 Tipos de filtros esenciales en el procesamiento digital de señales.

2.3.5 TEOREMA DE NYQUIST

El teorema de Nyquist, también conocido como el teorema de muestreo, es un principio que se usa en la discretización de señales analógicas. En la conversión analógico-digital (ADC), para dar como resultado una reproducción fiel de la señal, las muestras de la forma de onda analógica se deben tomar con una frecuencia determinada. Esta frecuencia se conoce como tasa de muestreo o frecuencia de muestreo.

Una señal analógica arbitraria, salvo que sea senoidal pura, consta siempre de componentes en distintas frecuencias. En el caso de la onda sinusoidal, toda la energía de la señal se concentra en una sola frecuencia. En la práctica, las señales analógicas suelen tener formas de onda complejas, con componentes en muchas frecuencias. La componente de frecuencia más alta de una señal analógica determina el ancho de banda de esa señal. Cuanto mayor sea la frecuencia, mayor es el ancho de banda, si todos los demás factores se mantienen constantes.

Supongamos que el componente de frecuencia más alta, en Hz, para una señal analógica dada es F_{max} . De acuerdo con el teorema de Nyquist, la frecuencia de muestreo debe ser al menos $2F_{max}$, o lo que es lo mismo, dos veces la mayor componente de frecuencia. Si la tasa de muestreo es menor que $2F_{max}$, algunos de los componentes de alta frecuencia en la señal de entrada analógica no serán representados correctamente en la salida digitalizada. Cuando dicha señal digital es convertida de nuevo en forma analógica, esto es mediante un convertidor de digital a analógico, aparecen componentes falsos de frecuencia que no estaban en la señal analógica original. A este fenómeno indeseable se le conoce como aliasing.

A efectos de este TFG, como las ondas que se quieren detectar, ondas cerebrales alfa, se presentan en valores de frecuencia comprendidos entre los 8 Hz y los 13 Hz, bastaría con una frecuencia de muestreo de 26 Hz para satisfacer el teorema de Nyquist.

No obstante, a una frecuencia de muestreo tan baja, no se sabría con precisión suficiente cuándo aparece la frecuencia de las ondas alfa, por lo que es necesaria una frecuencia de muestreo superior para aumentar la precisión. Además, la realización de cualquier filtro que opere a frecuencias próximas a la mitad de la de muestreo son más complejos que si esa frecuencia se halla más alejada de los puntos singulares del filtro.

2.3 APORTACIÓN AL TFG

El desarrollo de este TFG comprende : análisis del funcionamiento de la FFT en procesamiento de datos captados por encefalografía(EEG), diseño Hardware del sistema de procesamiento de dichos datos y programación del Software de control.

Al realizar este proyecto se permite el análisis de diferentes señales biológicas, de tipo, EEG y ECG, por lo que no solo aumentaría el número de aplicaciones médicas, domésticas y comerciales, sino que también se conseguiría una mayor precisión al analizar los resultados

pues se contaría con tres fuentes diferentes para establecer correlaciones. La plataforma de procesamiento de los datos ofrece la posibilidad de realizar los análisis en tiempo real, con lo que el abanico de posibilidades es muy amplio. Si además añadimos la reconfiguración evolutiva no solo se conseguirían encontrar nuevas aplicaciones sino que se ayudaría en el campo de la investigación estudiando el comportamiento de las distintas ondas cerebrales.

Capítulo 3

Investigación previa

3. INVESTIGACIÓN PREVIA

Inicialmente, y antes del diseño y desarrollo del sistema de procesamiento de datos en Hardware, se realiza una investigación previa. Esta investigación es realizada por un equipo de tres personas entre las que se incluye el autor de este proyecto.

El trabajo previo consiste en el estudio de una plataforma que use el mismo ADC que usaremos posteriormente en nuestro sistema en Hardware. Para ello nos apoyamos en la plataforma *OpenBCI*, mostrada en la figura 3-1.

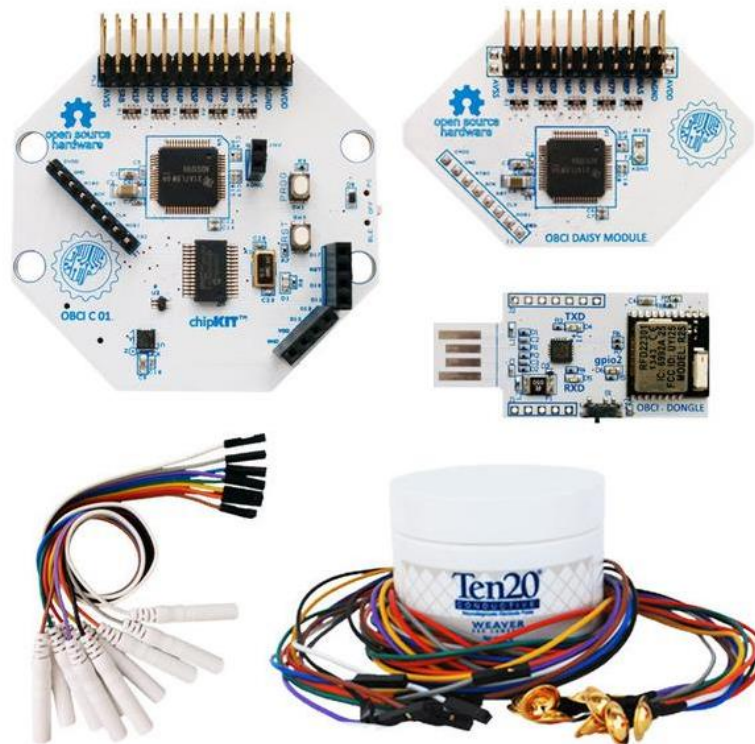


Ilustración 0.1

Ilustración 3.1: Kit OpenBCI

En la primera fase de esta investigación, se procede a realizar análisis de señales EEG en esta plataforma para, primero, tener una primera toma de contacto con el entorno de trabajo, y segundo, acotar el alcance de este ADC en nuestro modelo. El procedimiento que se ha seguido es el siguiente:

1. Diseño del experimento.
2. Captar datos y almacenarlos.
3. Analizar los datos en MATLAB.
4. Evaluar los resultados obtenidos.
5. Rediseñar el experimento con los resultados obtenidos.

3.1 CONFIGURACIÓN Y DISEÑO DEL EXPERIMENTO

En la primera etapa, configuración y diseño del experimento, se cuenta con la ayuda del Dr. Ceferino Maestú, investigador y profesor en el centro de tecnología biomédica (CTB) de la UPM. Entre los experimentos más básicos que nos propone, destacan dos:

El primero consiste en la captación de ondas alfa, donde el sujeto mantiene los ojos cerrados y se relaja durante un periodo corto de tiempo. Esta relajación se traduce en la aparición de una señal a una frecuencia comprendida entre los 9 y los 12 Hz. El segundo consiste en cerrar el puño de una mano e intentar captar e interpretar dicha información en la señal EEG recogida.

Para la correcta colocación de los electrodos, se utiliza el sistema internacional 10-20, denominado así porque los electrodos están espaciados entre el 10% y el 20% de la distancia total entre puntos reconocibles del cráneo. El esquema se muestra en la siguiente figura

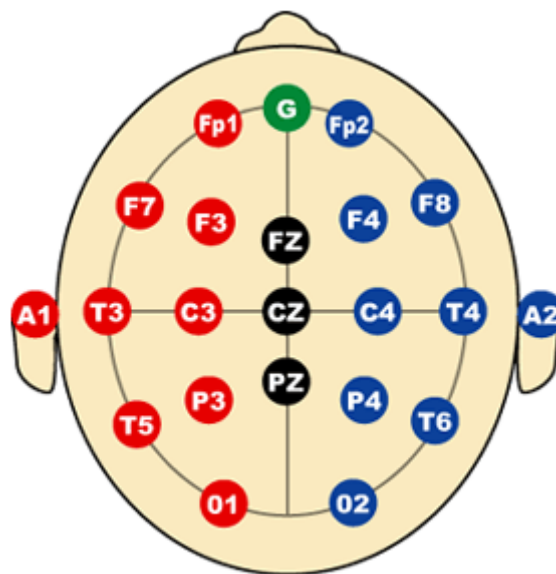


Ilustración 3.2: Sistema internacional 10-20 para la colocación de los electrodos en el cráneo.

En el primer experimento, el de captación de las ondas alfa, se conectan 3 electrodos principales, dos de ellos en posiciones cercanas a O1 y O2, esto es en el lóbulo occipital donde la generación de las ondas alfa se produce con más intensidad. El otro electrodo se coloca en la oreja, actuando como referencia común. El resto de electrodos no se colocan pues no interesa lo que capten para el experimento inicial.

En la plataforma del experimento, la frecuencia de muestreo no se puede variar, por lo que viene predeterminada por el programa, y es de 250 muestras/segundo. Esto es así para asegurar una comunicación inalámbrica fiable entre el emisor, *OpenBCI* y el receptor Bluetooth.

En el segundo experimento, el de captación del cierre de un puño, se conectan de nuevo 3 electrodos, dos de ellos en posiciones cercanas a C3 y C4, esto es el lóbulo parietal, donde se realizan las ordenes motoras del cerebro. El otro electrodo se coloca en la oreja para actuar de referencia común. Igual que en el experimento anterior, el resto de electrodos no se colocan. Hay que destacar que en el desarrollo de este experimento, no se consigue captar información relevante que nos permita distinguir cuando se ha cerrado el puño. Se llega a la conclusión de que esto se debe principalmente a la baja calidad de los electrodos de los que se dispone, con lo que se rechaza este experimento.

Las gráficas mostradas a continuación son referidas en su totalidad al experimento de captación de las ondas alfa.

3.2 RECEPCIÓN Y ALMACENAMIENTO DE DATOS

La empresa desarrolladora del sistema proporciona una interfaz gráfica en tiempo real que nos permite, observar las señales captadas por los electrodos y ver, de esta forma, si los electrodos se han colocado correctamente o no para realizar el experimento. La interfaz gráfica se muestra en la siguiente figura:

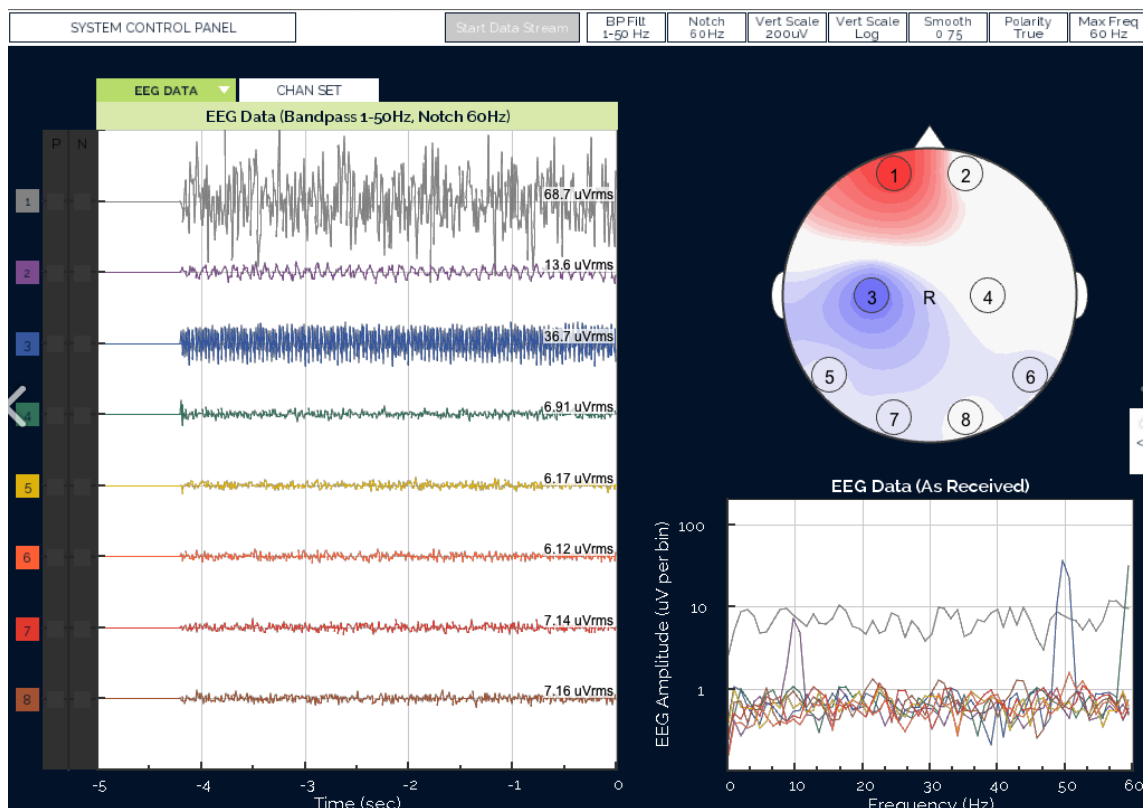


Ilustración 3.3 Interfaz gráfica facilitada por OpenBCI. A la izquierda se muestran las señales en tiempo real en el dominio del tiempo. A la derecha arriba se muestran las actividades cerebrales captadas por cada electrodo, y abajo a la derecha se muestran las señales el dominio de la frecuencia.

Además, también se facilita un programa de control basado en Python. Este programa no dispone de una interfaz gráfica, pero nos permite controlar el comportamiento del sistema. Una vez se ha comprobado que tenemos los electrodos correctamente colocados, los datos se almacenan de forma inalámbrica en un archivo de extensión CSV.

Para la interpretación y procesamiento de los datos captados, la empresa desarrolladora también facilita un programa basado en Processing. Sin embargo, se usará MATLAB como software de procesamiento ya que ofrece funciones matemáticas de alto nivel que nos facilitarán la interpretación de los datos.

3.3 ANALISIS DE SEÑALES EN MATLAB

Para el análisis de los datos almacenados se determina que los principales análisis a realizar, según Mike X Cohen, en su libro “Analyzing Neural Times Series Data, Theory and Practice” son tres: frecuencia, en tiempo y espacio (ver ilustración 3.3). Al tratarse de un equipo de tres miembros durante la fase de investigación previa, cada miembro del grupo se encarga de un análisis en concreto. El autor de este proyecto se centra en el análisis en el dominio de la frecuencia.

3.3.1 ANALISIS EN FRECUENCIA

Los datos recogidos para desarrollar los experimentos correspondientes se procesan en MATLAB de acuerdo a las siguientes pautas:

1. Se leen los datos de la hoja de datos (archivo CSV).
2. Se elige el tamaño de la muestra a analizar.
3. Se filtra la señal para eliminar el ruido de 50 Hz.
4. Se aplica una ventana para evitar las discontinuidades al principio y al final de la muestra analizada.
5. Se completa con *zero padding* en el tiempo.
6. Se aplica la FFT al bloque analizado con ventana y *zero-padding*.
7. Los resultados se llevan a una gráfica para ser analizados.

En las gráficas que se muestra a continuación, el sujeto del experimento se mantuvo primero 10 segundos con los ojos abiertos y después 10 segundos con los ojos cerrados.

El procesamiento de los datos del experimento ha sido el siguiente:

- El tamaño de la muestra que se elige corresponde con la zona de los 10 segundos en la que se cierran los ojos para poder captar las ondas cerebrales alfa.
- Los resultados que se muestran en la imagen han pasado por un filtro Notch a 50 Hz. Posteriormente se ha pasado la muestra filtrada por un filtro paso bajo con la frecuencia de corte en 49 Hz.
- El número de ceros que se ha añadido para el Padding es de 10 veces el tamaño inicial de la muestra.
- La ventana aplicada es de tipo Hann.

Los resultados obtenidos son los siguientes:

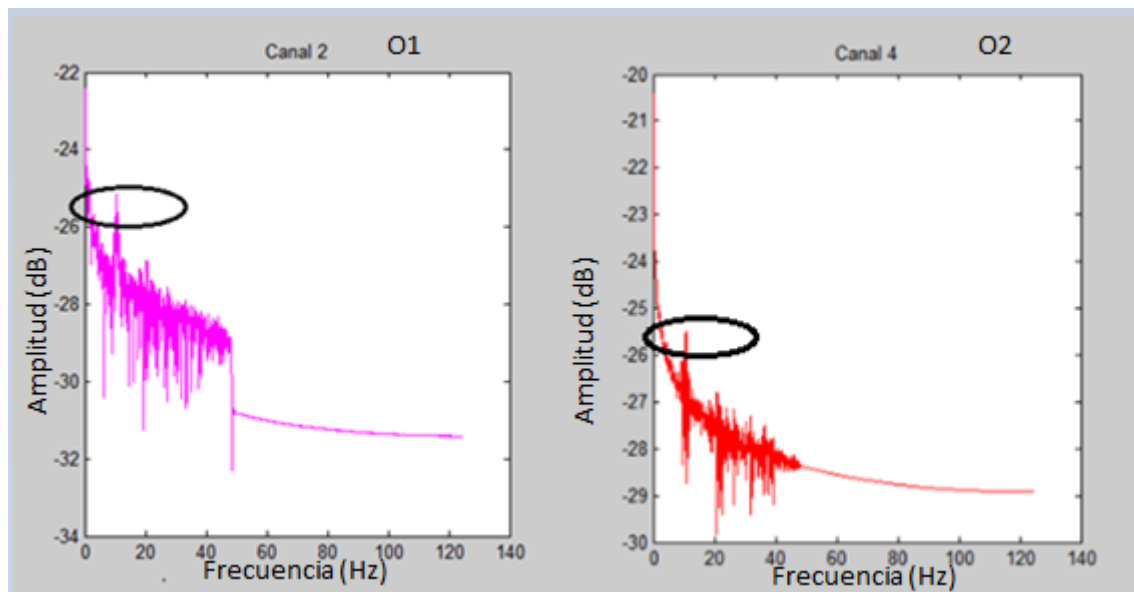


Ilustración 3.4 Ensayo de captación de ondas alfa y procesadas en MATLAB. La imagen de la izquierda corresponde a los datos captados por el electrodo situado en O1. A la derecha los captados por O2. En ambas se puede apreciar claramente un pico en la frecuencia cercana a los 10 Hz.

Se puede observar la aparición de un pico destacado en la frecuencia de 10 Hz, como corresponde con la frecuencia de las ondas cerebrales alfa, lo que nos indica que el experimento se ha desarrollado con éxito. Por otro lado, se puede ver como el filtrado que se ha realizado atenúa eficazmente las frecuencias superiores a 49 Hz, las cuales solo añadirían ruido al experimento.

Otro aspecto a considerar es la diferencia de decibelios entre los electrodos colocados en O1 y O2. Esto se debe a la diferente impedancia que tienen los electrodos al haberse colocado en la cabeza del sujeto, dada la dificultad en la colocación de la pasta conductiva en los mismos.

Además, se puede observar que la frecuencia máxima es de 125 Hz. Esto es consecuencia directa del teorema de Nyquist (Referencia a 2.2.5) con una frecuencia de muestreo de 250 muestras/segundo.

3.3.2 ANALISIS EN TIEMPO

Este tipo de análisis gana fuerza en caso de realizar experimentos basados en potenciales evocados. Como se explicó anteriormente (referencia a apartado potenciales evocados), la señal aparece 300 ms después de percibir el sujeto del experimento el estímulo, lo que hace este análisis valioso en detrimento del análisis en frecuencia, en cuyo caso no se apreciaría la respuesta temporal.

Otra forma interesante de analizar potenciales evocados, o incluso de realizar otros experimentos, consiste en la representación en el dominio del tiempo de los distintos

valores de la frecuencia (ver figura, sacada de libro “Analyzing Neural Times Series Data, Theory and Practice”,)

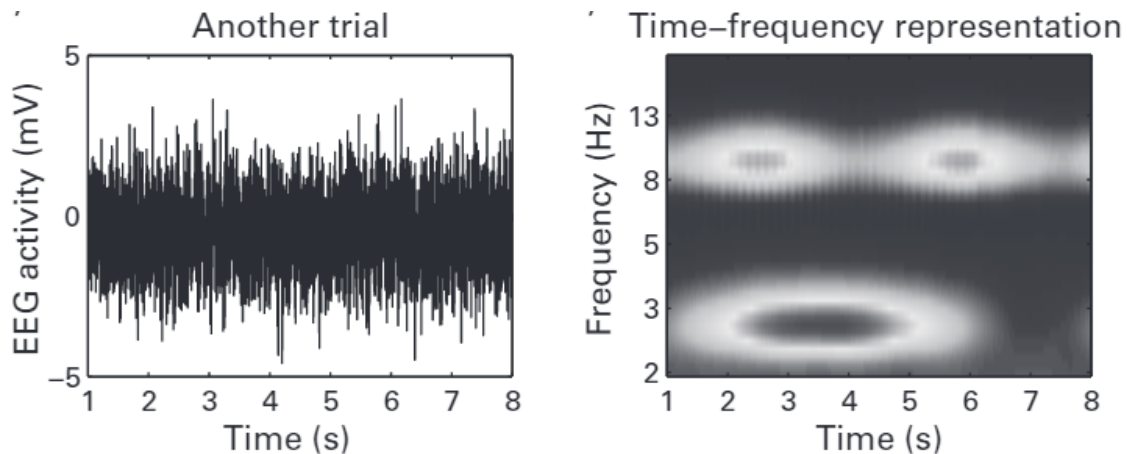


Ilustración 3.5 Aplicación del análisis temporal. Imagen obtenida del libro *Analyzing Neural Times Series Data, Theory and Practice*.

3.3.3 ANALISIS EN ESPACIO

En el análisis en espacio, se aplica un filtro Laplaciano. Este filtro se adapta a 8 electrodos activos y no presenta una gran precisión, debido a que el número mínimo de electrodos necesarios es 64, y como la plataforma OpenBCI de la que disponemos solo acepta 8 electrodos, el uso del filtro Laplaciano no es posible.

Por estos motivos, se descarta el análisis espacial para los ensayos realizados en este TFG.

3.4 CONCLUSIÓN FINAL A LA INVESTIGACIÓN PREVIA.

Tras la evaluación del procedimiento en el análisis de los datos y los resultados obtenidos, se llega a la conclusión de que existe un factor limitante en el proceso. Este factor es la frecuencia de muestreo, la cual afecta directamente a la precisión de la muestra captada. Al ser esta de 250 Hz y no poderse actuar sobre ella en la plataforma OpenBCI de la que se dispone, se aprueba crear un sistema de adquisición de datos [12], el cual usará el mismo sensor ADS1299 que mandará la información a una FPGA Virtex5 que realizará el procesamiento de los datos, esta vez, a mayor frecuencia de muestreo.

Analizando los resultados se llega también a la conclusión de que la técnica del padding en el tiempo influye en la resolución y no en la precisión. Al tratarse de una frecuencia de 250 Hz, la precisión es de 250 puntos en una muestra de un segundo, por lo que un padding excesivo añade puntos en el dominio de la frecuencia que falsean la interpretación real de los datos.

Se define así el objetivo principal de este TFG, el cual consiste en el desarrollo del sistema de procesamiento de los datos, el cual siga los mismos pasos que en el análisis de los datos mediante MATLAB.

En el Anexo B se muestran diferentes ensayos realizados en MATLAB durante la investigación previa.

Capítulo 4

Cálculo, diseño y desarrollo
del sistema.

4. CALCULO, DISEÑO Y DESARROLLO DEL SISTEMA

4.1 ELEMENTOS PARA EL DESARROLLO

En el desarrollo del sistema de procesamiento de datos se utilizan los siguientes elementos.

En lo que respecta al Hardware:

- FPGA: XUPV5-LX110T
- Kit de evaluación del ADS1299EEG-FE
- Tarjeta de memoria Flash

En lo que respecta al Software:

- ISE Design Suite de Xilinx v13.2
- MATLAB versión 2015b.

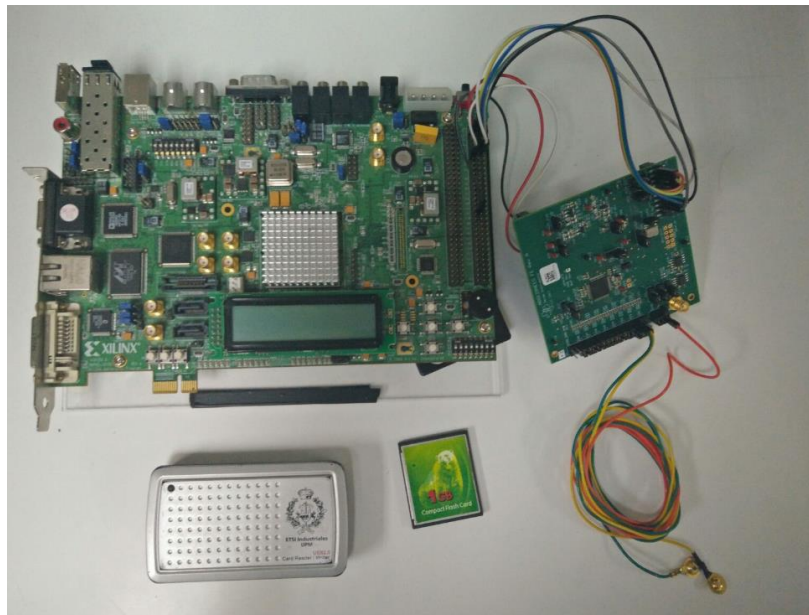


Ilustración 4.1 Elementos usados en el sistema de procesamiento de datos

4.2 DISEÑO DEL BLOQUE FFT EN HARDWARE

Para diseñar un sistema que siga con los pasos llevados a cabo en la investigación previa, se vuelve necesario realizar un bloque que realice la FFT. Debido al alto número de cálculos que se deben realizar en la ejecución de la transformada rápida de Fourier, se elige realizar este bloque en Hardware para aprovechar así la alta velocidad de procesamiento y el bajo consumo energético que presenta el Hardware en detrimento del Software.

Xilinx, la empresa facilitadora del software ISE Design Suite v13.2, ofrece una herramienta llamada CoreGenerator en la cual se dispone de distintos bloques prediseñados, cada uno para realizar una determinada tarea. Las características de estos bloques son parametrizables, con lo que se puede ajustar su funcionamiento a las condiciones de trabajo del experimento que se desee realizar.

4.2.1 POSIBILIDADES DEL BLOQUE FFT

Se escoge el bloque prediseñado por Xilinx “LogiCORE IP Fast Fourier Transform v7.1”, perteneciente a la sección de DSP (Digital Signal Processing), que se encargará de realizar la FFT a los datos enviados desde el kit de evaluación del ADS1299. Todas las posibles señales que de las que dispone este bloque son:

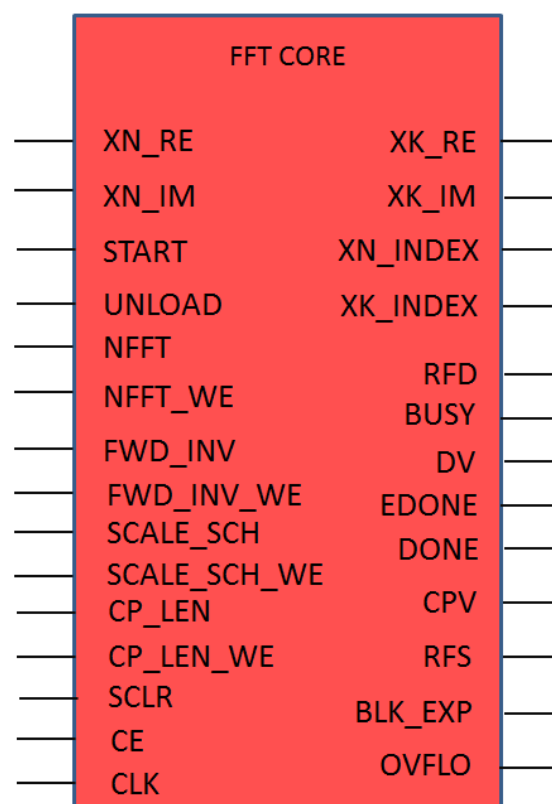


Ilustración 4.2 Señales E/S del bloque FFT a generar usando la herramienta facilitada por Xilinx, CoreGenerator.

Las diferentes señales y sus funcionalidades son:

Nombre del Puerto	Anchura del puerto	Dirección	Descripción.
XN_RE	Entre 8 y 34 bits	Entrada	Datos de entrada reales a la FFT.
XN_IM	Entre 8 y 34 bits	Entrada	Datos de entrada imaginarios a la FFT.
START	1	Entrada	Para empezar a cargar datos en el bloque.
UNLOAD	1	Entrada	Para empezar a descargar datos del bloque.
NFFT	5	Entrada	Indica el tamaño de la transformada. Solo se usa en configuración en tiempo de ejecución.
NFFT_WE	1	Entrada	Para todos los procesos e inicializa el bloque al nuevo tamaño configurado. Solo se usa en configuración en tiempo de ejecución.
FWD_INV	1	Entrada	Indica si se trata de transformada directa o inversa de Fourier. 1 indica inversa, 0 directa.
FWD_INV_WE	1	Entrada	Para cambiar FWD_INV.
SCALE_SCH		Entrada	Representa el tamaño despreciado para realizar el escalado correctamente.
SCALE_SCH_WE	1	Entrada	Solo en arquitecturas con escala.
CP_LEN	log2(tamaño máximo)	Entrada	Longitud del prefijo cíclico. Representa el número de muestras que son inicialmente salida como prefijo cíclico, antes de que toda la transformada sea salida.
CP_LEN_WE	1	Entrada	Para cambiar CP_LEN.
SCLR	1	Entrada	Reset síncrono. Puerto opcional
CE	1	Entrada	Para activar el reloj.
CLK	1	Entrada	Reloj.
XK_RE	Dependerá de la arquitectura y de si es con escala o sin escala.	Salida	Datos de salida reales de la FFT.
XK_IM	Dependerá de la arquitectura y de si es con escala o sin escala	Salida	Datos de salida imaginarios de la FFT.
XN_INDEX	log2(tamaño máximo)	Salida	Índice de los datos de entrada.
XK_INDEX	log2(tamaño máximo)	Salida	Índice de los datos de salida.
RFD	1	Salida	Listo para el dato. Está a 1

			durante el proceso de carga.
BUSY	1	Salida	Ocupado. Está a 1 durante el procesamiento de los datos.
DV	1	Salida	Dato válido. Se pone a 1 cuando hay datos procesados para sacar de la FFT.
EDONE	1	Salida	Se pone a 1 un ciclo antes de DONE.
DONE	1	Salida	Fin. Se pone a 1 cuando acaba el procesamiento de los datos en la FFT.
BLK_EXP	5	Salida	Exponente de escala. Solo en bloques configurados para trabajar con punto flotante.
OVFLO	1	Salida	“Overflow”. Se pone a 1 durante la descarga si algún valor sobrepasa el tamaño.
CPV	1	Salida	Se pone a 1 cuando se presenta a la salida datos validos con prefijo cíclico.
RFS	1	Salida	Se pone a 1 cuando el bloque está preparado para que se introduzca START. Solo válido para prefijo cíclico y en “streaming”.

4.2.2 CONFIGURACION DEL BLOQUE FFT

Las características de este bloque se eligen de la siguiente ventana:

Ilustración 4.3 Ventanas en la configuración del bloque FFT diseñado para este TFG.

Por tanto el bloque definitivo presenta las siguientes características:

- Dispone de un canal de entrada de datos, esto es una señal de entrada real y otra imaginaria. Se podría configurar para tener hasta 8 canales.
- La longitud de la transformada es de 1024 datos. Se podría configurar hasta de 65536.
- Se elige la arquitectura Radix-2 Lite. Corresponde a una mariposa de radix 2, la más pequeña, que ofrece el menor tamaño, aunque menor velocidad. Sin embargo, esto no es un problema debido al número de muestras y su velocidad.
- El formato de los datos es de punto fijo (Fixed Point).
- El ancho de de los datos de entrada es de 24 bits máximo. Se podría configurar hasta de 32.
- El ancho del factor de fase es de 16 bits máximo. Se podría configurar hasta de 32.
- Se elige la opción de no aplicar ningún factor de escala a los datos de salida.
- Se añaden los pines opcionales CE y SCLR.
- Se elige la opción “3 clock cycle offset”.
- Se elige la opción “truncation” como modo de aproximación.
- Los datos a la salida se ordenan de forma natural, esto es desde el dato 0 hasta el 1023. Se podría configurar para que salieran ordenados del 1023 al 0.
- Como opciones de memoria, los datos se guardan en memorias BRAM.
- Se elige lógica CLB y optimización de los recursos.

Por tanto, el bloque utilizado en el desarrollo de este TFG cuenta con las siguientes señales:

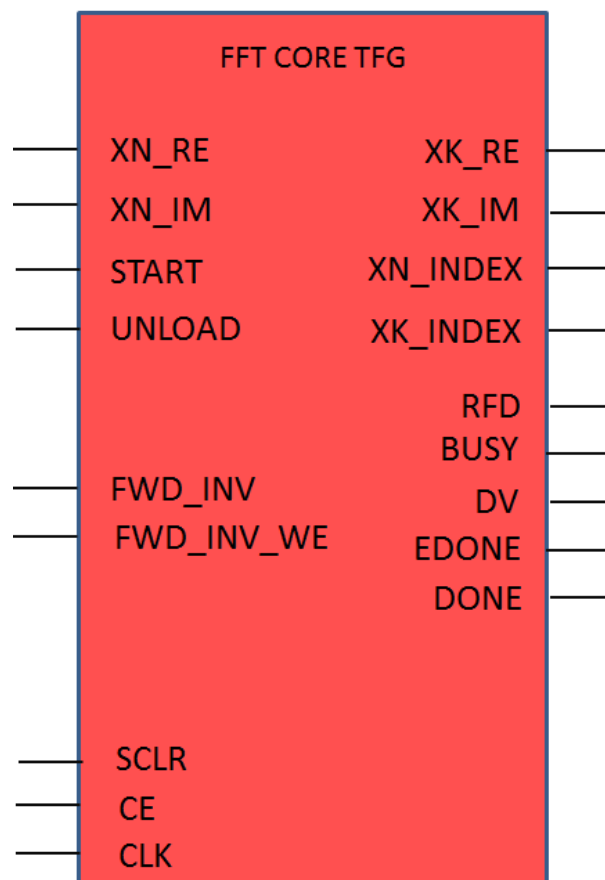


Ilustración 4.4 Señales E/S del bloque FFT diseñado para este TFG.

4.2.3 TIMING DEL BLOQUE FFT

El bloque FFT puede diseñarse atendiendo a diferentes arquitecturas. Estas arquitecturas se diferencian por la forma en se procesan los datos, lo que hará de algunas más eficientes que otras en determinados contextos. En el caso de este TFG, se ha elegido la arquitectura Radix-2 Lite y su funcionamiento es el siguiente:

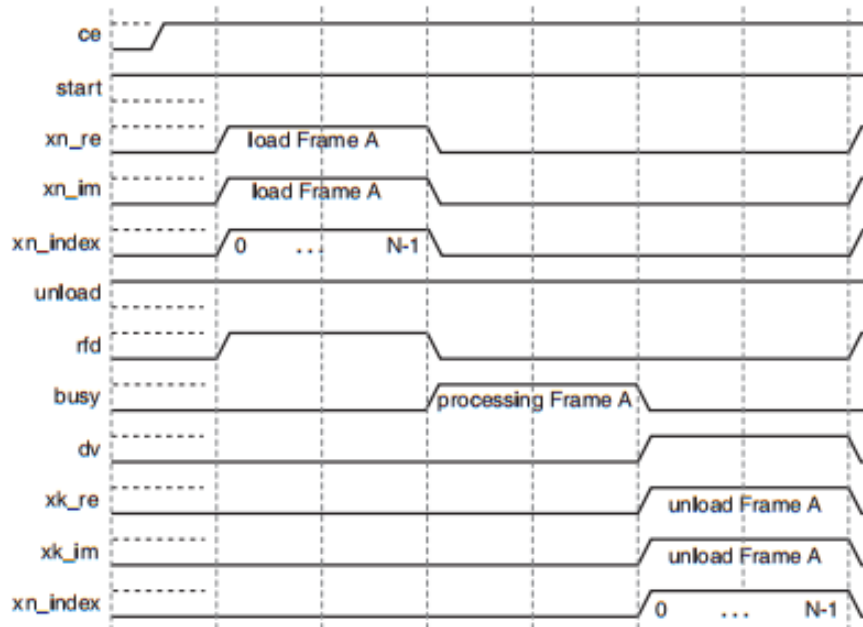


Ilustración 4.5 Timing sin detalles del bloque FFT.

Inicialmente, la señal de CE (Clock Enable) se pone a 1, con lo que el CLK (Clock) en este caso se pondrá en marcha. Posteriormente, START se pone a 1 (en la imagen se mantiene a 1 desde el principio), de forma que el proceso está listo para empezar a cargar datos. Una vez los datos están listos para cargar, RFD se pone a 1, y empieza el proceso de carga (Load Frame). Seguidamente, el sistema empieza a procesar los datos (Processing Frame) y BUSY se pone a 1 hasta que se acabe el procesamiento. En ese momento, DV se pone a 1, y el sistema empieza a descargar los datos (Unload Frame) hasta que se llega al final de la muestra donde se acaba el bucle. Por último, se puede observar que el sistema no presenta ninguna señal que indique exactamente el momento en el que acaba la descarga.

Por otro lado, hay que destacar que el diagrama muestra los distintos pasos en el funcionamiento de todo el bloque, pero las longitudes de las distintas partes del mismo no están a escala real. El tiempo de procesamiento de los datos es bastante más grande que el de carga y descarga de los datos.

Una vez realizado el modelo, se procede a su simulación para asegurar el correcto funcionamiento de la FFT. Para poder realizar la simulación, se utilizan dos memorias accesibles de la FPGA, donde se carga unos datos inventados que harán de datos de entrada al bloque. Los datos procesados se almacenarán en otras dos memorias. Se utilizan dos memorias a la entrada y dos a la salida para guardar en una la parte real de los datos y en la otra la parte imaginaria. El sistema simulado es el siguiente.

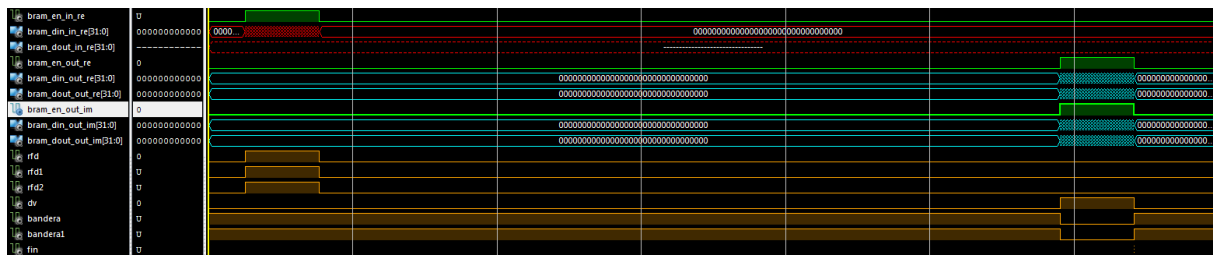


Ilustración 4.6 Simulación del bloque FFT. Se puede apreciar como los tiempos de carga y descarga son diferentes del tiempo de procesamiento, el cual es bastante más largo.

En el modelo simulado no se muestra la señal BUSY, que indica cuando los datos están siendo procesados. Esto se debe a que no se va a usar esa información para ninguna acción en concreto, con lo que se decide no implementar dicha señal en el modelo de prueba.

Es importante destacar, y aunque no se puede ver en la imagen, la carga de los datos se realiza tres ciclos de reloj inmediatamente después de que RFD se ponga a 1, tal y como corresponde con la configuración de “3 clock cycle offset”. Esto se mostrará con más detalle en el capítulo 4.3.1 MEMORIAS BRAM.

4.3 DISEÑO HARDWARE DEL SISTEMA

Una vez diseñado y comprobado el correcto funcionamiento del bloque encargado de realizar la transformada rápida de Fourier, se pasa al diseño completo del modelo. Este diseño se realizará utilizando el programa XPS, incluido en el EDK de Xilinx. Para ello se van a utilizar varias memorias BRAM de las que se dispone en la FPGA XUPV5-LX110T, así como el bus PLB (Processor Local Bus). El modelo en XPS se muestra en el Anexo C.

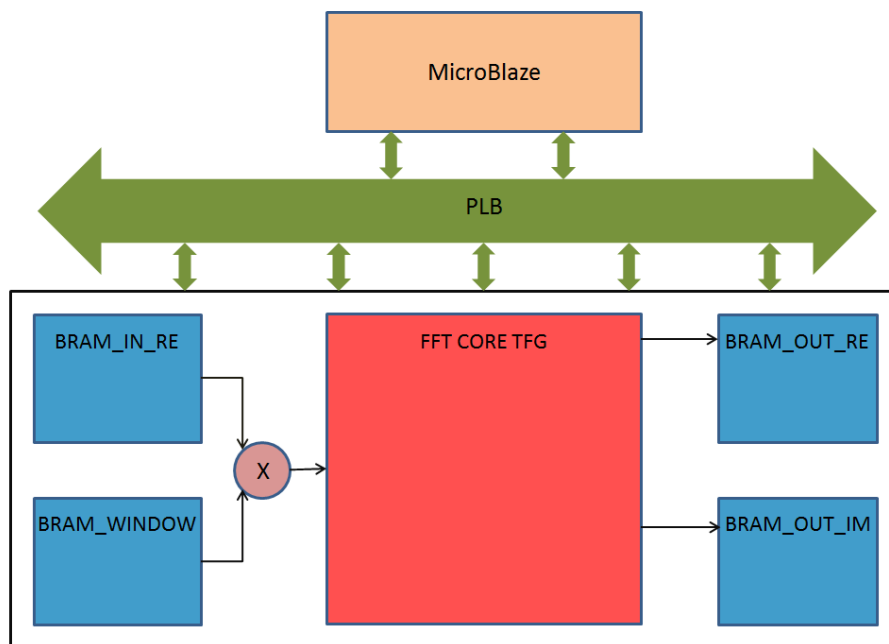


Ilustración 4.7 Diagrama funcional del diseño Hardware del sistema desarrollado.

4.3.1 SISTEMA MICROPROCESADOR

El sistema microprocesador es el sistema que se encargará de gestionar las diferentes órdenes que se le den al sistema entre los distintos elementos que lo forman. También informará de los posibles errores que se produzcan.

En este TFG, el sistema está compuesto por:

- MicroBlaze: consiste en un microprocesador interno que se puede instanciar en todas las FPGAs de Xilinx. Este microprocesador se encarga de ejecutar todas las órdenes que se le den desde el programa SDK.
- PLB: consiste en un bus de control que se encarga de comunicar el MicroBlaze con los distintos elementos del sistema.

Por un lado, el MicroBlaze, salvo en contadas excepciones, realiza las diferentes acciones que se le ordene a ciclo a ciclo de reloj. Esto permite mantener el flujo de información a ciclo de reloj con lo que no se pierde eficiencia en el sistema. Por otro lado, el MicroBlaze dispone de un sistema de interconexión muy versátil lo que permitiría su adaptación a una gran variedad de sistemas embebidos.

Por su parte, el PLB, como bus de control, al que se le añade un timer, rutas de direcciones, escritura y lectura, y un registro de control del dispositivo, para controlar los posibles errores que aparezcan.

4.3.1 MEMORIAS BRAM

Las memorias BRAM que se utilizan son memorias estándar que forman parte de la arquitectura interna de la FPGA. Son de 32 bits de ancho de palabra, mientras que la capacidad es de 4 KB más que suficiente para almacenar los 1024 datos de 24 bits de los captados por el sistema de adquisición de datos.

. Hay que destacar que, al tratarse de datos captados por electroencefalografía, estos no presentan parte imaginaria, por lo que no es necesario implementar una BRAM para almacenar los datos de entrada imaginarios:

En el diseño final se añaden cuatro memorias BRAM, que son:

- BRAM_IN_RE: en esta memoria se almacenan los datos captados por el ADS1299.
- BRAM_IN_WINDOW: en esta memoria se cargará mediante el microprocesador interno (MicroBlaze) de la FPGA una ventana de tipo Hann. .En caso de querer realizar padding, también se cargará aquí.

Los datos procesados se almacenarán en otras dos memorias, una para la parte real de la FFT, y la otra para la parte imaginaria.

- BRAM_OUT_RE: en la memoria se almacenará la parte real de los datos procesados por el bloque FFT.
- BRAM_OUT_IM: en esta memoria se almacenará la parte imaginaria de los datos procesados por el bloque FFT.

Las conexiones entre las BRAMs y el bloque FFT son las siguientes:

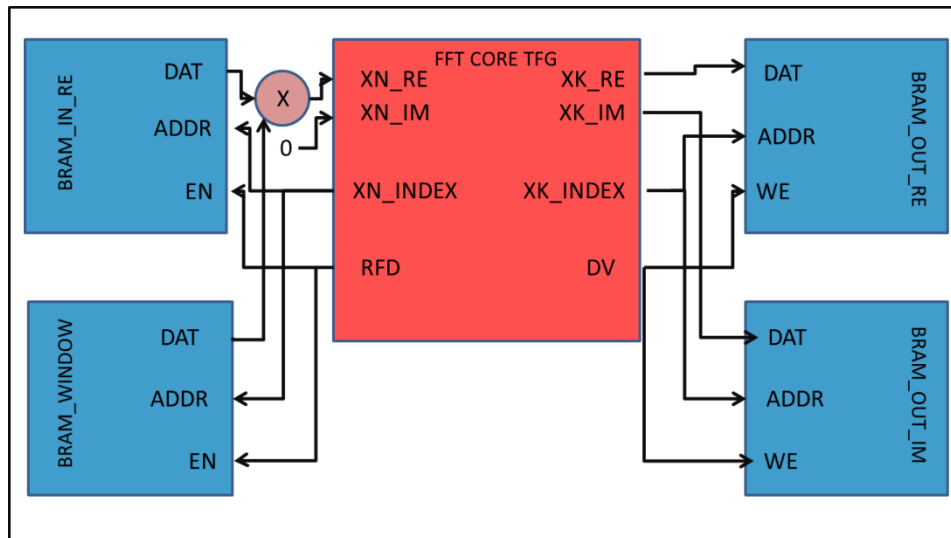


Ilustración 4.8 Conexiones entre las memorias BRAM y el bloque FFT.

Como ya se mencionó anteriormente, la carga y descarga de los datos se produce 3 ciclos de reloj después de ponerse a 1 RFD. Esto es importante ya que RFD activa el Enable de las BRAMs de entrada, por lo que para que se produzca la carga coherentemente, es decir los índices del dato 0 correspondan con la primera dirección de memoria de la BRAM, se debe compensar este adelanto.

En lo que corresponde a la carga de los datos, haciendo zoom se tiene:

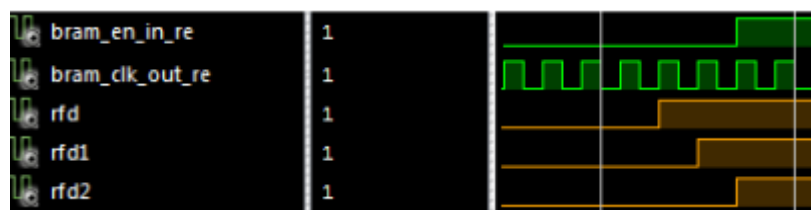


Ilustración 4.9. Detalle de la simulación en el inicio de la carga de los datos, correspondiente al modo del bloque “3 clock cycle offset”.

Se observa cómo RFD se produce 3 ciclos de reloj antes, por lo que desplazándolo obtenemos RFD2, que ya activará los Enable coherentemente.

En la parte correspondiente a la descarga, esta empieza a producirse en el momento en que DV se pone a 1. Para disponer de un pulso que indique el momento en que acaba la descarga, se realizan las siguientes modificaciones y se añade la señal “FIN”.



Ilustración 4.10 Detalle de la simulación en el fin de la descarga de los datos, donde se puede ver que se ajusta el sistema para producir la señal fin, de un ciclo de reloj.

Vemos que el pulso FIN se produce dos ciclos de reloj posterior al final de la descarga.

Hay que destacar que aunque el bloque FFT ya presenta una señal, DONE, que indica el fin de la descarga, esta se pone a 1 permanentemente, con lo que no es un pulso, el cual presenta ciertas ventajas.

4.3.2 VENTANA

Una vez implementadas todas las BRAM del sistema, se vuelve necesario aplicar una ventana a los datos captados por el ADS1299, para que de esta forma los datos entren al bloque FFT con la ventana ya aplicada.

Para ello, se dispone de una memoria BRAM, llamada BRAM_IN_WINDOW, donde estarán cargados los datos de la ventana que queramos aplicar. Estos datos se cargarán por Software, tal y como se explicará en el apartado 4.4 DISEÑO SOFTWARE DEL SISTEMA.

Una vez cargados los datos, estos saldrán de la memoria de forma que la dirección 0x00 de la BRAM_IN_WINDOW se multipliquen con la dirección 0x00 de la BRAM_IN_RE, entrando así al bloque FFT cada dato multiplicado por su “simétrico” correspondiente de la ventana.

De esta forma, aunque el MicroBlaze se encargará de cargar los datos de la ventana correspondiente en la BRAM_IN_WINDOW, la acción de la misma se realiza por Hardware.

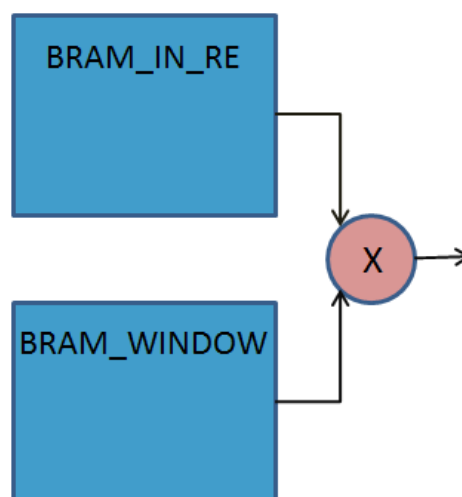


Ilustración 4.11. Aplicación de la ventana en Hardware.

4.3.3 INTEGRACION DEL SISTEMA DE ADQUISICION DE DATOS

Por último, y para terminar de desarrollar todo el Hardware del sistema, es necesario integrar el sistema de captación de datos [12]. Para ello, se integran al diseño en XPS del sistema los siguientes elementos:

- Timer de 32 bits: se trata de un timer que es utilizado en el sistema de adquisición de datos para llevar tiempos de espera. Además, se usa también para medir tiempos de ejecución en los experimentos que se realizan.
- Bloque Compact Flash: permite el uso de una memoria compact flash que será útil para exportar datos y manipularlos desde MATLAB.
- Bloque SPI: para la adquisición de datos siguiendo el protocolo SPI.
- Bloque GPIO: el cual tiene 3 salidas-entradas de propósito general.

4.4 DISEÑO SOFTWARE DEL SISTEMA

El sistema será controlado por un MicroBlaze que se encargara de toda la gestión y comunicación del sistema. Este MicroBlaze se programa en lenguaje C, utilizando el programa SDK que viene junto con la plataforma EDK.

Principalmente, y como parte funcional del sistema de procesamiento de datos, el MicroBlaze se encargará de las siguientes tareas. En lo que respecta al sistema de adquisición de datos:

- Se encarga de toda la comunicación entre la FPGA y el ADS1299 a partir de la cual se captan los datos. Esto se realiza siguiendo el protocolo de comunicación SPI.
- El MicroBlaze se encarga de cargar los datos captados en la BRAM_IN_RE. Además, este se encarga también de cargar los datos de la ventana tipo Hann en la BRAM_WINDOW.
- Permite la configuración de los diferentes registros del ADS1299, lo que permite realizar diferentes experimentos a diferente frecuencia de muestreo, así como activar el modo "Test Signal" para comprobar el correcto funcionamiento en caso de producirse algún error en la comunicación.
- Mediante el Timer se mide el tiempo que tarda en realizarse la FFT.

Para más información detallada, el código implementado en el MicroBlaze se encuentra en el Anexo A.

4.5 ASPECTOS IMPORTANTES DEL SISTEMA

En este apartado se mencionan brevemente los principales aspectos del sistema.

- La aplicación de la transformada de Fourier se realiza sin aplicación de ninguna escala, y el tamaño de la palabra de salida es de 34 bits. De esta forma se consigue una precisión exacta en la salida. En los experimentos que se realizan, la salida nunca va a alcanzar un 1 en el bit 33, por lo que se cogen los bits que van del 0 al 31, es decir, se quitan los 2 bits más significativos ya que siempre van a ser 0. De esta forma no se pierde información relevante, mientras que escalando se perderán los bits menos significativos, los cuales si presentan algo de información útil.
- El sistema está implementado para realizar un ciclo de FFT, es decir, una carga, un procesamiento y una descarga de los datos. En caso de querer realizarlo de otra forma, se debería primero determinar el funcionamiento que se quiere.

Posteriormente se realizarían los cambios necesarios en el código VHDL de la arquitectura del sistema.

- Al tratarse de EEG, la señal captada no presenta parte imaginaria, por lo que el sistema desarrollado no está implementado para adaptarse a señales que si presenten parte imaginaria como entrada al bloque FFT.
- Aunque la implementación de la ventana se produce en Hardware, la ventana se carga desde el Software, lo que permite elegir el tipo de ventana que se desee en el experimento a realizar.
- El bloque FFT usa 6912 *flip-flops* , lo que supone un 10% del total de los flip flops. Además, usa 5692 LUTs, lo que supone un 8% del total de las LUTs. Las memorias BRAM usan un 22% de la memoria BRAM de la FPGA. y usa 8 bloques de DSP, lo que supone un 12% del total.
- La frecuencia de muestreo se puede configurar desde el Software y está comprendida entre valores de 250 a 16000 Hz.

Capítulo 5

Obtención y Análisis de
Resultados.

5. OBTENCIÓN Y ANÁLISIS DE RESULTADOS

En este capítulo se muestran los resultados a los que se ha llegado durante la realización de este TFG.

Primero se muestra como se comprobó la correcta integración del sistema. Posteriormente y viendo que el sistema ya recibía los datos captados por el ADS1299, se procedió a realizar un ensayo de aplicación del sistema integrado, esto es, aplicando el sistema a una onda cuadrada y viendo que su salida en el dominio de la frecuencia corresponde con la de una onda cuadrada. Por último, se realiza un experimento de captación y procesamiento de un EEG real, mediante la captación de ondas alfa reales.

5.1 CAPTACIÓN DE SEÑAL

Como se mencionó anteriormente, el primer resultado al que se llega es a la correcta integración del sistema. Para ello se utiliza la configuración “Test Signal” del ADS1299. En esta configuración el ADS1299 crea una señal cuadrada de testeo, la cual nos permite ajustar el sistema para comprobar si envía datos correctamente. La “Test Signal” se muestra en la ilustración 5.1.

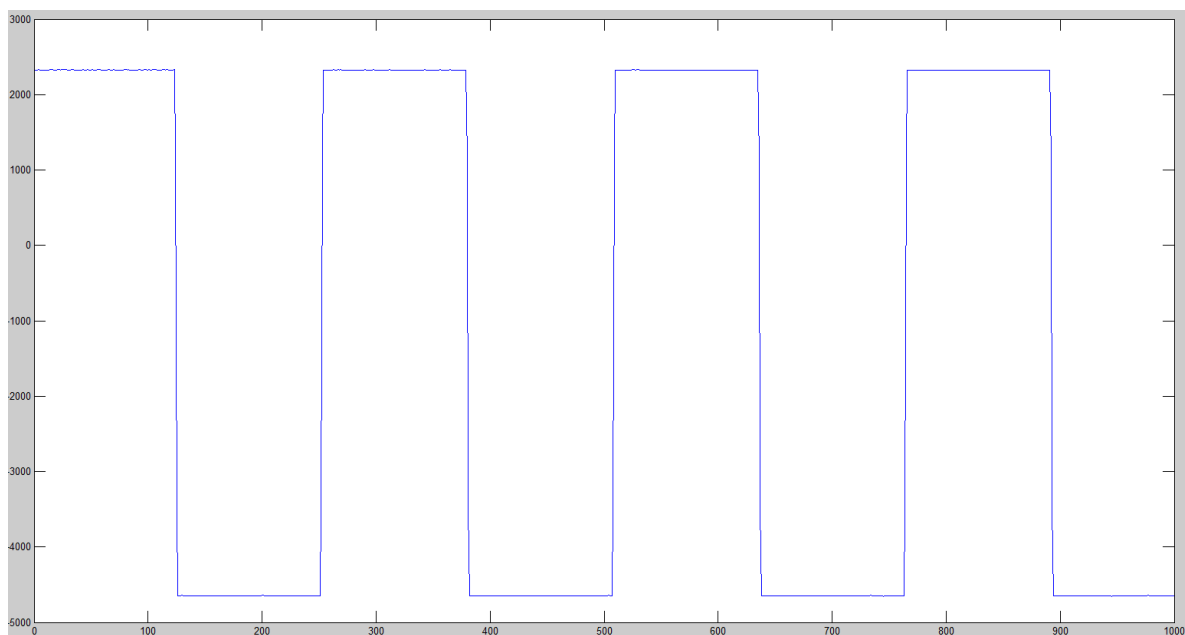


Ilustración 5.1 Test Signal del ADS1299

A continuación, se procede a comprobar que la adquisición de señales externas por el ADS1299 se está llevando a cabo correctamente. En experimentos anteriores, realizados empleando el software de control provisto por Texas Instruments, se observó una fuerte presencia de un pico de señal a 50 Hz, síntoma de que se estaba captando algo de ruido

procedente del exterior, posiblemente debido a interferencias electromagnéticas emitidas por aparatos eléctricos conectados a la red, tales como los tubos fluorescentes, que producen emisiones electromagnéticas a la frecuencia de la red, 50 Hz. Para comprobar que se están adquiriendo datos correctamente se ha procedido a conectar uno de los canales del ADS1299 en cortocircuito empleando un cable, como puede verse en la ilustración 5.2. Este montaje debería producir una medida constante de 0 V en condiciones ideales, sin embargo actuará de “antena” captando posibles interferencias electromagnéticas.

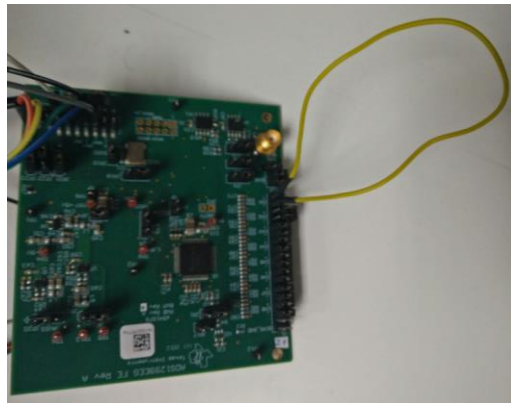


Ilustración 5.2 Cable que hace de antena para captar ruido.

Una vez realizado el experimento, y tras realizar la FFT de los datos obtenidos usando Matlab, se llega al resultado mostrado en la ilustración 5.3, en la que puede observarse un claro pico a la frecuencia de 50 Hz, validando el funcionamiento del sistema.

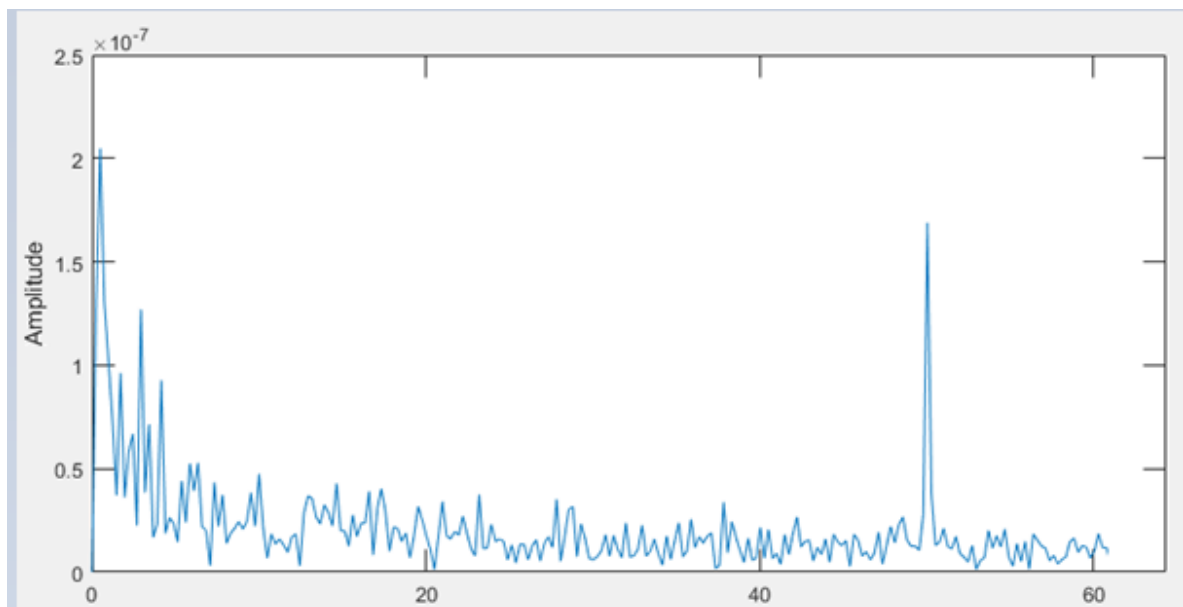


Ilustración 5.3 FFT a la señal captada por la antena. Se puede apreciar un pico en la frecuencia de 50 Hz, correspondiente al ruido eléctrico de la red. En el eje de ordenadas se muestra la amplitud y en el de abscisas la frecuencia en Hz

Por último, para comprobar que los valores obtenidos son correctos, se procede a conectar dos electrodos a los terminales de una fuente de tensión para medir su diferencia de potencial empleando el ADS1299. La fuente genera 1 V, y se muestra en la ilustración

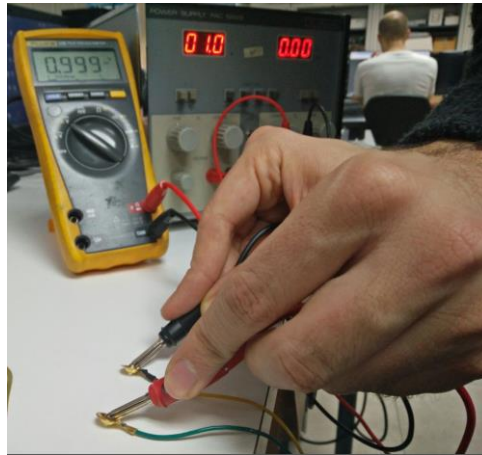


Ilustración 5.4 Fuente de tensión para el ensayo. Se puede observar que la fuente genera 1 V y que el voltímetro verifica que es 1 V.

El valor obtenido por el ADS1299 se muestra en la Ilustración 5.4

```
*** START ***
Iniciando ADS1299
Enviando reset
Configurando
+0.999455 V
+0.967654 V
+1.000584 V
+1.001009 V
+1.001825 V
+1.001766 V
+1.003346 V
```

Ilustración 5.5 Valores de la fuente de tensión captados por el ADS1299.

5.2 ANÁLISIS DE UNA ONDA CUADRADA

Posteriormente, y viendo que el sistema estaba correctamente integrado, se procedió a evaluar el bloque FFT Hardware empleando como señal de entrada una onda cuadrada generada desde el MicroBlaze. Para ello se cargan los valores correspondientes para simular una onda cuadrada en la BRAM_IN_RE, es decir, en la memoria de entrada al sistema. Se aplica una ventana tipo Hann y la salida se almacena en las BRAM correspondientes. El resultado se muestra en la ilustración 5.4.

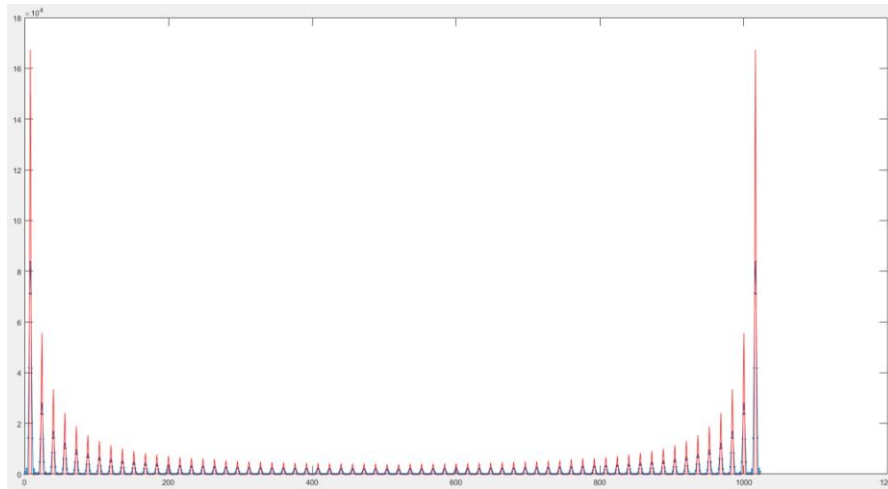


Ilustración 5.6 FFT de onda cuadrada. En el eje de las ordenadas se muestra el log de la amplitud en dBs y en el eje de ordenadas el número de muestras.

Como se puede observar, la salida que se muestra corresponde con la salida correspondiente a la transformada rápida de Fourier a una señal cuadrada como la introducida en el bloque, por lo que se valida experimentalmente el correcto funcionamiento del sistema de procesamiento de datos. En azul se muestra la FFT a la onda cuadrada habiendo tomado muestras a una frecuencia de muestreo 500 Hz, mientras que en rojo se han tomado las muestras a una frecuencia de muestreo 1000 Hz. Se puede apreciar que a 1000 Hz de muestreo, se dispone del doble de muestras en 1 segundo que a 500 Hz de muestreo, por lo que la amplitud se dobla.

5.3 PROCESAMIENTO DE EEG REAL

Una vez validado, por un lado, el sistema de adquisición de datos mediante la colocación de la antena que capta el ruido a 50 Hz, y por otro lado, el sistema de procesamiento de los datos mediante la aplicación de la FFT a la onda cuadrada, se procede a realizar un ensayo completo de ambos sistemas integrados. Para ello se realiza un experimento que consiste en la captación, almacenamiento y procesamiento de los datos de un EEG real que captará ondas alfa.

En el experimento, se conectan 3 electrodos, dos de ellos en posiciones cercanas a O1 y O2, esto es en el lóbulo occipital donde la generación de las ondas alfa se produce con más intensidad. El otro electrodo se coloca en la oreja, actuando como referencia común. El resto de electrodos no se colocan pues no nos interesa lo que capten para nuestro experimento. El tiempo del experimento es de 2 segundos, a una frecuencia de muestreo de 1000 Hz.

El resultado se muestra en la ilustración 5.5.

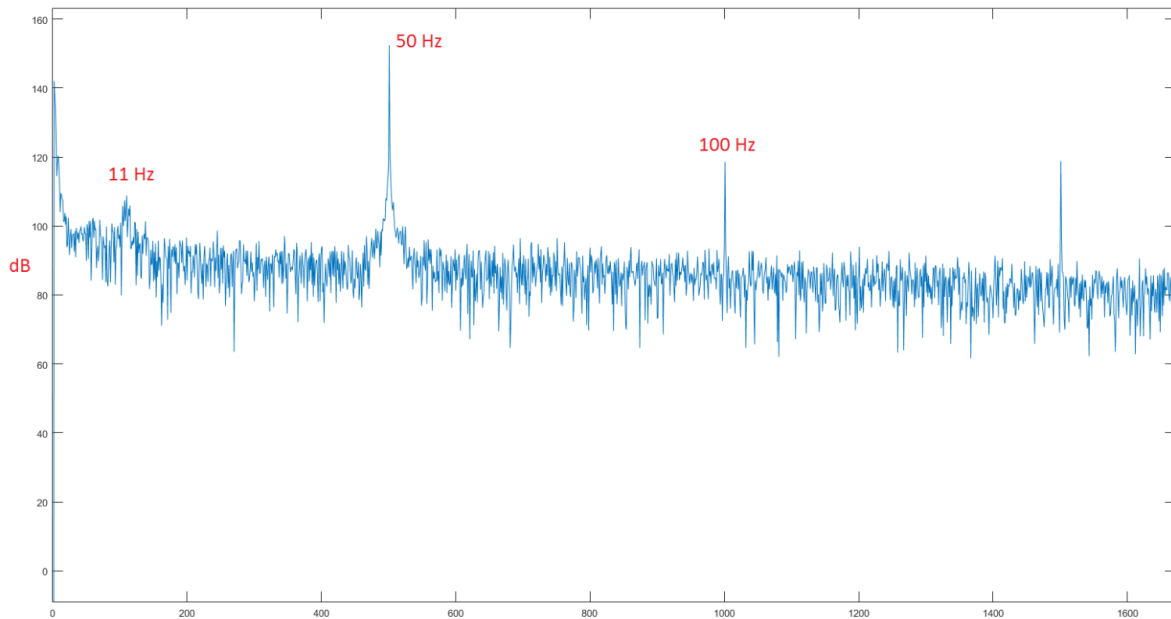


Ilustración 5.7 Aplicación del sistema desarrollado sobre un EEG real. Se puede ver la frecuencia de ondas alfa a 11 Hz. En el eje de ordenadas se representa la amplitud, en dB, y en el de abscisas el número de muestras.

Se puede observar la presencia de 2 picos fundamentales. Por un lado, el pico a 50 Hz, junto con armónicos a 100 y 150 Hz indicador del ruido eléctrico captado en el experimento. Para corregir este pico, se aplicaría un filtro tipo Notch a la frecuencia de 50 Hz. Por otro lado, se puede apreciar un claro pico a la frecuencia de 11 Hz, lo que corresponde con la aparición de ondas alfa. A efectos de reducir el pico de los 50 Hz, se justificaría la aplicación de un filtro tipo Notch, para atenuar esta frecuencia.

Por otro lado, y para detectar dicho pico en un sistema de procesamiento en tiempo real, se realizaría un detector, a base de comparar los umbrales del máximo en esa zona frente al resto de la señal filtrada. De esta forma, si se supera la amplitud umbral en un intervalo de frecuencia de ondas alfa, esto significaría la generación de dichas ondas por el usuario analizado.

5.4 VELOCIDAD DE PROCESAMIENTO

Una vez realizados los experimentos, se mide el tiempo de ejecución de la transformada rápida de Fourier. Por un lado, se mide el tiempo que tarda el bloque FFT en realizar los algoritmos de transformación, llegando al siguiente resultado mostrado en la ilustración 5.6.

```

*** START ***
Iniciando ADS1299
Enviando reset
Configurando
Recibiendo 1024 muestras
Muestras recibidas! Escribiendo
Procesando
Hecho. 99 us (12399 ciclos)
    
```

Ilustración 5.8 Tiempo de ejecución de la FFT en Hardware.

También se mide el tiempo de la simulación, obteniéndose el resultado mostrado en la ilustración 5.7.

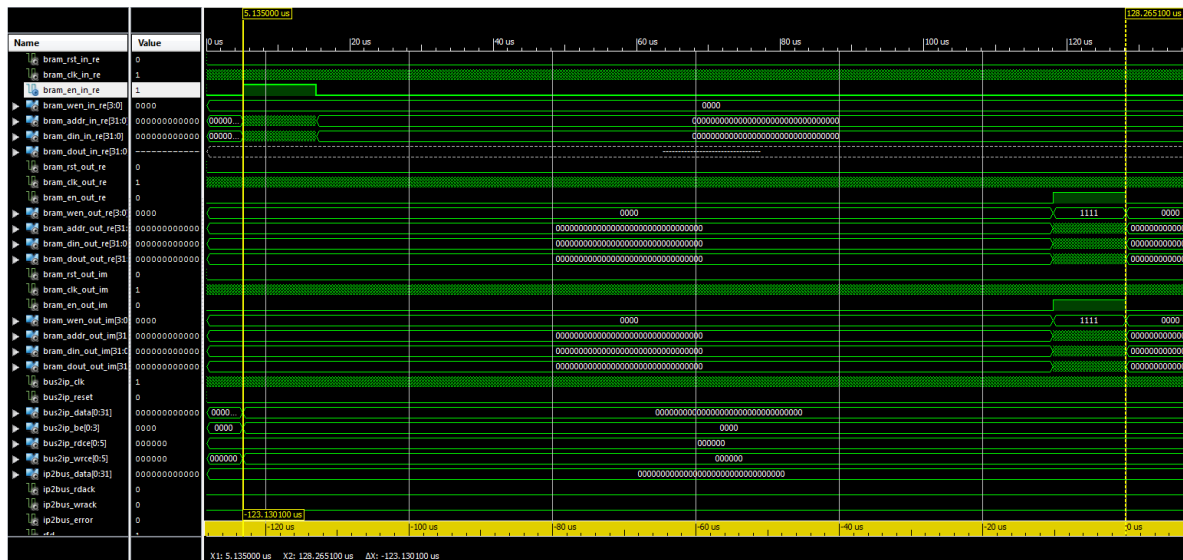


Ilustración 5.9 Tiempo de simulación de la FFT en Hardware.

Por un lado, en el experimento se ha tardado 12399 ciclos de reloj , mientras que en la simulación se ha tardado 123.13 microsegundos. Como el ciclo de reloj en la simulación es de 10 nanosegundos, se obtiene que la simulación se ha realizado en 12313 ciclos de reloj, lo cual se asemeja bastante al tiempo real de ejecución, pudiéndose justificar la diferencia por el overhead introducido por el bus de comunicaciones.

Para validar y comprobar si la velocidad de ejecución del procesamiento en Hardware es superior al procesamiento en Software, se utiliza MATLAB como herramienta de procesamiento en Software. La función “Tic...Toc” de MATLAB nos permite saber el tiempo de ejecución de la función que se quiera analizar. Aplicando esto a la función FFT de MATLAB, se obtiene el tiempo de ejecución que se muestra en la ilustración 5.8.

```
>> in_re = eeg(:,3);
>> tic; out = fft(in_re); toc
Elapsed time is 0.002698 seconds.
```

Ilustración 5.10 Tiempo de ejecución de la FFT en Software, usando MATLAB

Se ve que en el caso del Software, la ejecución es del orden de 2 milisegundos, siendo unas 27 veces más lenta que en el Hardware.

Capítulo 6

Conclusiones y Líneas Futuras

6. CONCLUSIONES Y LINEAS FUTURAS

6.1 CONCLUSIONES

En primer lugar, se han conseguido los objetivos principales de este TFG: diseñar y desarrollar un sistema de procesamiento en una FPGA de datos captados por el chip ADS1299, y realizar un ensayo de un EEG real de ondas cerebrales alfa utilizando dicho sistema.

El aumento del número de muestras que se pueden procesar, al captarse más muestras por segundo (mayor frecuencia de muestreo) que con la plataforma de OpenBCI tiene como inmediata consecuencia que los ensayos serán más precisos al disponer de más información el sistema de procesamiento.

Por otro lado, analizando los resultados, se puede llegar a las siguientes conclusiones. Primero, se ha conseguido integrar el sistema de captación de datos con el sistema de procesamiento de datos, lo que facilita el desarrollo de nuevos ensayos más complejos y difíciles de abordar sin el sistema integrado.

Además, se ha generado e integrado un bloque que se encarga de realizar la transformada rápida de Fourier en Hardware. En el procesamiento digital de señales, la FFT es una herramienta esencial, por lo que el desarrollo del control de este bloque ofrece una gran gama de usos, al poderse usar en cualquier proyecto que fuese requerido. En la ilustración 6.1 se representan en la misma gráfica la FFT en Hardware (color rojo) y la FFT en MATLAB (color azul), y como se puede observar, no se aprecia diferencia ninguna. Haciendo zoom se aprecia una pequeña diferencia, mostrada en la ilustración 6.2.

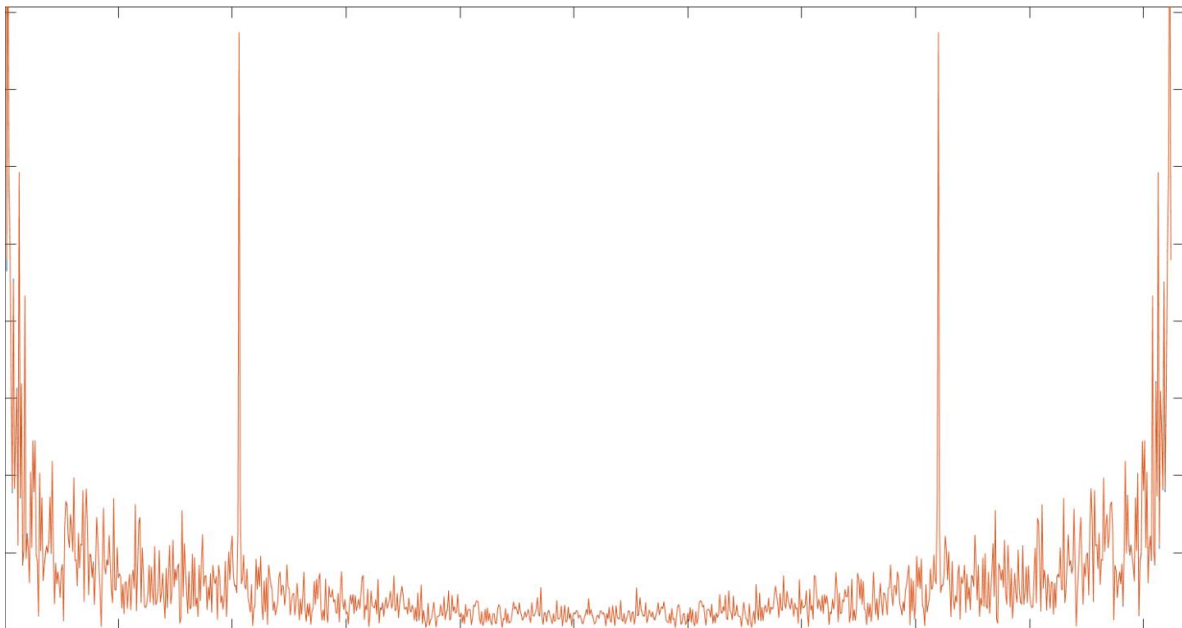


Ilustración 6.1 FFT realizada en Hardware y FFT realizada en MATLAB, ambas representadas en la misma gráfica.

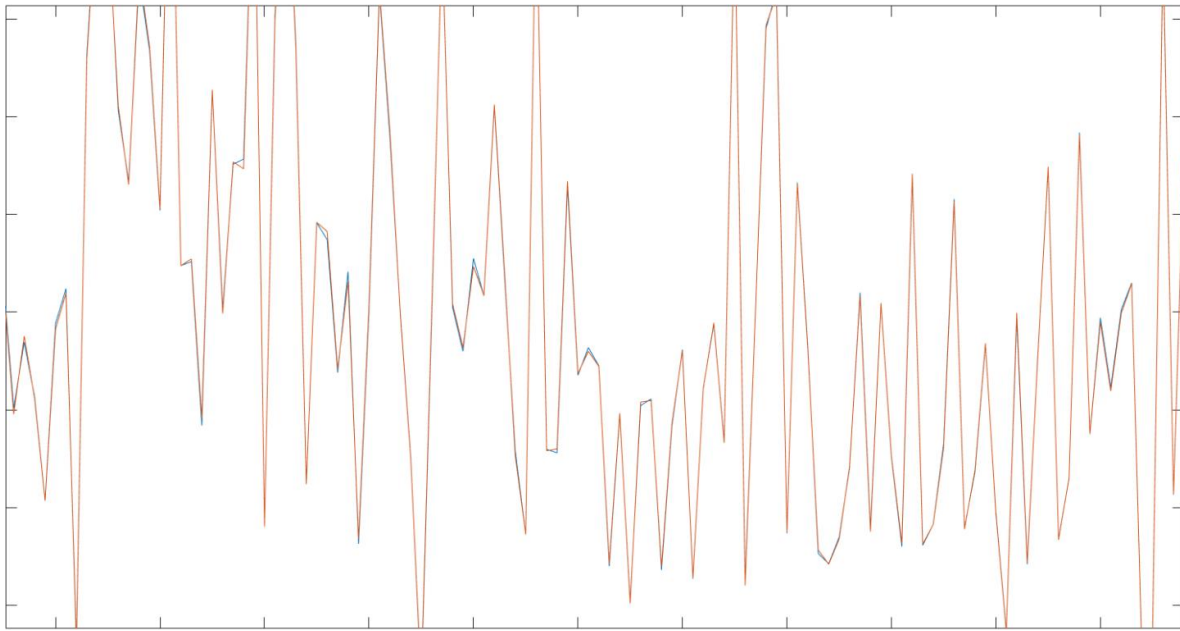


Ilustración 6.2 Zoom de la ilustración 6.1. En rojo, la FFT realizada en Hardware, y en azul la FFT realizada en MATLAB.

Las ventajas de realizar la FFT en Hardware son numerosas, pero la ventaja principal radica en el tiempo de ejecución. Apoyándose en los resultados obtenidos en cuanto a tiempos de ejecución, se llega a la conclusión de que la ganancia en tiempo de ejecución, definida como el tiempo de ejecución en Software entre el tiempo de ejecución en Hardware es de:

$$Ganancia = \frac{Tiempo\ Software}{Tiempo\ Hardware} = \frac{0.002698\ s}{99 * 10^{-6}\ s} = 27.25$$

Esto significa que el sistema desarrollado en Hardware realiza la transformada rápida de Fourier 27 veces más rápido que lo que tarda MATLAB. Hay que destacar que MATLAB es un software bastante depurado en cuanto a tiempos de ejecución, por lo que se puede ver aquí la gran ventaja competitiva del Hardware en lo que a tiempos de ejecución se refiere.

En lo que respecta el ensayo de ondas cerebrales alfa, se llegan a las siguientes conclusiones.

En primer lugar, se ha conseguido realizar un ensayo de EEG real, utilizando el kit de evaluación del ADS1299 y una FPGA. Comparando este ensayo con el realizado en la investigación previa, se puede observar la similitud en el pico captado a la frecuencia de los 11 Hz, por lo que se valida el ensayo realizado. Además, al disponerse de más frecuencia de muestreo, y con ello más precisión, el uso de la técnica de zero-padding se vuelve innecesario para este tipo de ensayos al no necesitarse aumentar la resolución.

En segundo lugar, hay que destacar que al poder realizar ensayos de EEG reales mediante el sistema desarrollado, se presentan numerosas ventajas respecto a la plataforma de

OpenBCI. La primera es la alta velocidad de procesamiento de la FPGA, y la segunda es la alta frecuencia de muestreo que ofrece el ADS1299. Si a esto se le añade la capacidad del bloque FFT de recibir datos de ocho canales distintos, a una frecuencia de 16000 Hz, el flujo de información procesada es mucho mayor que el que permite la plataforma de OpenBCI, lo que ofrece una amplia gama de posibilidades en el desarrollo de futuros proyectos.

6.2 LINEAS FUTURAS

Una línea de proyecto a corto plazo será la integración de los elementos utilizados en el sistema desarrollado en una placa propia. Estos elementos son una FPGA y el chip ADS1299. De esta forma, esta placa dispondría de la unidad de adquisición de datos, así como de la unidad de procesamiento en el mismo elemento. Además, también se dispondrá de ocho canales para realizar ensayos de electroencefalografía.

Una vez integrada la placa, una línea de proyecto sería la revisión de experimentos utilizando el ADS1299 que estén documentados. Algunos de estos experimentos son: potenciales evocados visuales o táctiles, P300, o incluso procesos motores. Una vez documentados, aplicar el sistema desarrollado a los diferentes experimentos.

Por otro lado, una línea de proyecto a largo plazo es el uso de algoritmos evolutivos para la reconfiguración de la FPGA, lo que se conoce como Hardware evolutivo o EHW (Evolvable Hardware). Estos sistemas son aquellos en los que la arquitectura puede variar dinámicamente y de manera autónoma, de forma que se propone un medio, y el sistema hace frente a este medio adaptándose. Esta adaptación se traduce en una variación de la circuitería interna de la FPGA, y está controlada por un algoritmo evolutivo que propone unas series de circuitos hasta llegar a una solución óptima.

Bajo esta perspectiva de buscar obtener la solución óptima, se pretende buscar una correlación entre los flujos de información captados a diferentes personas mediante ensayos EEG, por lo que se tendría una raíz común en los electroencefalogramas de todos los sujetos que permitiría una amplia variedad de aplicaciones.

Capítulo 7

Impactos y Aspectos de
Responsabilidad Legal, Ética y
Profesional.

7. IMPACTOS Y ASPECTOS DE RESPONSABILIDAD LEGAL, ETICA Y PROFESIONAL

7.1 RESPONSABILIDAD ETICA Y PROFESIONAL

Durante el desarrollo de este TFG, se han llevado a cabo diferentes experimentos en los cuales se han medido ondas cerebrales de diferentes sujetos mediante la colocación de un determinado número de electrodos en el cuero cabelludo. Para realizar estas mediciones, se han realizado las siguientes proposiciones al usuario del experimento.

En primer lugar se pide consentimiento al paciente para realizar el experimento. En segundo lugar se explica el objetivo del mismo y los distintos inconvenientes, como es el uso de una pasta conductiva en el cuero cabelludo para mejorar la conductividad eléctrica entre cerebro y electrodo. Esta pasta deja una textura pegajosa en el cuero cabelludo que desaparecerá con agua caliente. También se utiliza alcohol en la limpieza de dichos electrodos antes de colocarlos en el cuero cabelludo, con lo que podrían quedar ciertos restos del mismo. En tercer y último lugar, se mantendrá si así se desea el anonimato de forma pública de los datos captados y analizados.

Por otro lado, se toman las siguientes pautas profesionales para facilitar el desarrollo del sistema de procesamiento y la durabilidad de los elementos utilizados en el mismo.

- Se respeta el espacio de trabajo del resto de compañeros en el laboratorio, así como se cuida del equipo de forma responsable al tratarse de elementos muy susceptibles a sobretensiones que provoquen que se rompa.
- Se pregunta, en caso de duda, al director o codirector del proyecto sobre la utilización del equipo o ante algún problema en la etapa de diseño y desarrollo del sistema.

7.2 IMPACTOS PROFESIONALES

En lo que respecta al impacto profesional de este TFG, este trabajo sirve como referencia a la hora de diseñar y desarrollar un sistema complejo de procesamiento de datos sobre una FPGA. También sirve como referencia para proyectos que vayan a usar el ADS1299 para realizar EEG o EMG. Por otro lado, son numerosas las posibles líneas de investigación futuras lo que repercute directamente en el número de trabajos fin de grado o máster que se podrían ofertar.

7.3 IMPACTOS ÉTICOS

Los sistemas biométricos EEG captan potenciales del cerebro humano, lo cual permite relacionar ciertos patrones o comportamientos con una acción. Esto puede ayudar a muchas personas que presenten cierta discapacidad, ya sea motriz, de comunicación... con el simple hecho de pensar una acción. En cierta medida, estos sistemas surgieron con fines médicos, con lo que llevan en su esencia el factor ético.

Capítulo 8

Planificación Temporal y
Presupuesto

8. PLANIFICACION TEMPORAL Y PRESUPUESTO

8.1 PLANIFICACIÓN TEMPORAL

En la siguiente tabla se muestran las diferentes tareas realizadas durante este TFG, así como la fecha de inicio de las mismas y el tiempo dedicado.

Fecha de Inicio	Horas	Tarea
03/11/2014	1	Primera reunión con tutor sobre el proyecto en general.
17/11/2014	4	Reunión con tutor: Introducción de Hardware evolutivo y FPGAs.
10/12/2014	4	Reunión con tutor: Continuación anterior reunión.
20/01/2015	4	Reunión con Dr. Ceferino (CTB).
02/02/2015	30	Estudio de análisis por EEG
05/03/2015	3	Reunión con el equipo, puesta en común e introducción al procesamiento
10/03/2015	30	Desarrollo sistema procesamiento de datos en Matlab
16/03/2015	5	Preparación plataforma experimental para el Annual Meeting CEI
14/04/2015	10	Interfaz gráfica procesamiento datos
16/04/2015	5	Presentación plataforma experimental Annual Meeting
30/04/2015	4	Reunión: Evaluación de los resultados y definición objetivo individual
05/10/2015	4	Reunión con equipo para redefinir objetivos
06/10/2015	25	Conversión a VHDL de parte del código en Matlab usando HDL Coder de Matlab
19/10/2015	25	Diseño bloque FFT
02/11/2015	10	Reunión: Creación de sistemas embebidos en FPGA usando el programa ISE de Xilinx
04/11/2015	40	Implementación bloque FFT + 4 BRAM en XPS
18/11/2015	10	Simulación y ajuste del modelo implementado en Hardware
23/11/2015	30	Integración con el sistema de adquisición de datos y ajustes en el Software
14/12/2015	30	Captación ondas alfa con el sistema desarrollado en Hardware
26/12/2015	120	Redacción del TFG, una vez terminados los objetivos del mismo.

Sumando las horas, se obtiene un **TOTAL** de **394 horas** empleadas en la realización de las diferentes partes de este TFG.

8.2 DIAGRAMA DE GANTT

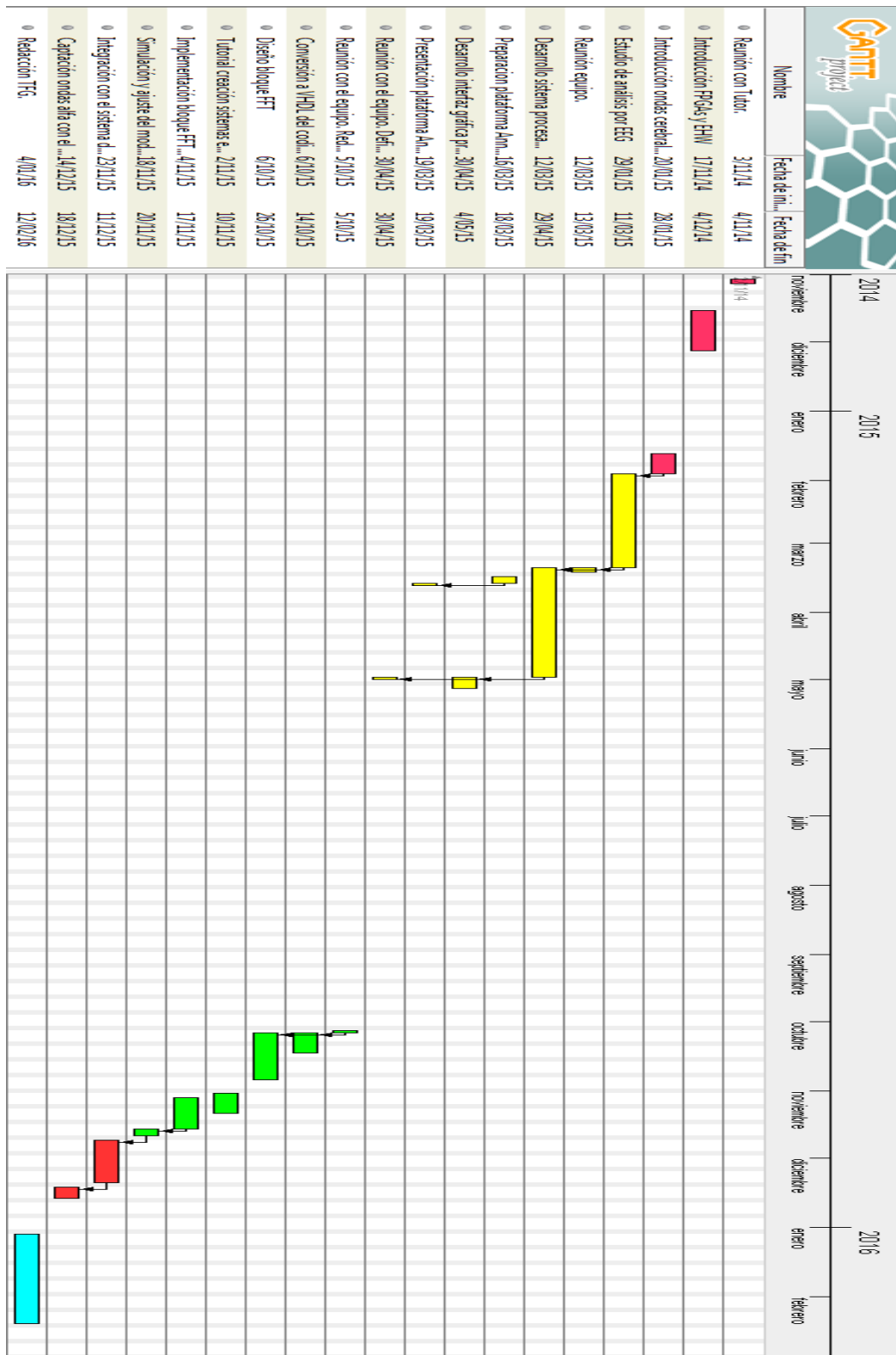


Ilustración 8.1 Diagrama de Gantt.

8.3 PRESUPUESTO

Durante el desarrollo del TFG, se han utilizado una serie de equipos para realizar los experimentos, así como diferentes programas informáticos. Los precios de estos equipos, así como su amortización en el tiempo utilizado vienen mostrados en los siguientes apartados. El salario del ingeniero que ha realizado este TFG se muestra en el apartado de salarios.

8.3.1 PRESUPUESTO EN HARDWARE

Aquí se incluyen los distintos equipos utilizados de forma directa en el proyecto. Estos son:

Dispositivo	Vida útil(años)	Precio(€)	Uso(años)	Amortización(€)
OpenBCI 32-bit Board Kit + OpenBCI Electrode Kit	5	829.77	0.5	82.97
ADS1299EEG-FE Evaluation Kit	5	183.69	0.5	18.36
FPGA Virtex5 XUPV5-LX110T	5	687.7	0.5	68.77
TOTAL(€)				170.1

8.3.2 PRESUPUESTO EN SOFTWARE

Aquí se incluyen las diferentes herramientas informáticas utilizadas en el desarrollo del TFG:

Programa	Vida útil(años)	Precio(€)	Uso(años)	Amortización(€)
Xilinx ISE	4	2700	0.5	337.5
Matlab 2015	1	500	0.5	250
TOTAL(€)				587.5

8.3.3 SALARIOS

Aquí se incluye el salario del Ingeniero que ha realizado el TFG.

Profesión	Salario(€/hora)	Tiempo proyecto(hora)	Coste
Ingeniero	12	394	4728

Sumando los costes en Hardware, Software y Salarios, se llega a un total de:

$$\text{Coste Total} = \text{Coste Hardware} + \text{Coste Software} + \text{Salarios} = 5485.6(€)$$

9. REFERENCIAS

- [1] Jonathan R. Wolpaw (Guest Editor), Niels Birbaumer, William J. Heetderks, Dennis J. McFarland, P. Hunter Peckham, Gerwin Schalk, Emanuel Donchin, Louis A. Quatrano, Charles J. Robinson, and Theresa M. Vaughan (Guest Editor) ,” Brain–Computer Interface Technology: A Review of the First International Meeting “, IEEE TRANSACTIONS ON REHABILITATION ENGINEERING, VOL. 8, NO. 2, JUNE 2000,
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.331.2636&rep=rep1&type=pdf>
- [2] OpenBCI.
<http://www.openbci.com/>.
- [3] EMOTIV
<https://emotiv.com/>
- [4] Kokoon
<https://kokoon.io/>
- [5] NoMoreWoof
<http://www.nomorewoof.com/>
- [6] Niels Birbaumer^{a,b}, Ander Ramos Murguialdaya^c and Leonardo Cohen, “Brain–computer interface in paralysis”, 2008.
<https://web.njit.edu/~alvarez/NEURAL%20ENGINEERING/JP/J14%20Birbaumer%20BCI%20paralysis.pdf>
- [7] Anatole Lécuyer and Fabien Lotte, INRIA Richard B. Reilly, Trinity College Robert Leeb, Graz University of Technology Michitaka Hirose, University of Tokyo Mel Slater , Technical University of Catalunya , “Brain-Computer Interfaces, Virtual Reality, and Videogames”. NOVEMBER 2008.
http://people.rennes.inria.fr/Anatole.Lecuyer/Lecuyer_computer_draft.pdf
- [8] Kyrre Glette, Jim Torresen University of Oslo, Norway ,Thiemo Gruber, Bernhard Sick University of Passau, Paul Kaufmann, Marco Platzner University of Paderborn, Germany, “Comparing Evolvable Hardware to Conventional Classifiers for Electromyographic Prosthetic Hand Control” Conference: Adaptive Hardware and Systems, DECEMBER 2008
<http://hjem.ifi.uio.no/kyrrehg/pf/papers/glette-ahs08.pdf>

- [9] Maksim Gorev, Vadim Pesonen, Dmitri Mihhailov, Maksim Jenihhin, Peeter Ellervee, Department of Computer Engineering, Tallinn University of Technology, Estonia ,
“FPGA-Based Implementation of EEG Analyze”

<http://research.microsoft.com/en-us/um/cambridge/events/date2011/ellervee.pdf>
- [10] Uzun.I.S Sch. Of Comput. Sci., Queen´s Univ. of Belfast, “FPGA implementations of fast Fourier transforms for real-time signal and image processing”, Vision, Image and Signal Processing, IEE Proceedings , 3 JUNE 2005
- [11] National Instruments. “Understanding FFTs and Windowing”

<http://www.ni.com/white-paper/4844/en/>
- [12] Ramón Conejo Laguna, “Brain Computer Interfaces: Desarrollo de sistema de adquisición de datos”.

10. INDICE DE ILUSTRACIONES

ILUSTRACIÓN 2-1 FUNCIONAMIENTO SISTEMA BCI. EN EL CASO DE ESTE TFG, LOS DATOS CAPTADOS POR EL ADS1299 Y PROCESADOS POR UNA FPGA VIRTEX 5.....	19
ILUSTRACIÓN 2-2 TIPOS DE ONDAS CEREBRALES E INTERVALO DE FRECUENCIAS DE LAS MISMAS.....	21
ILUSTRACIÓN 2-3 EFECTOS DE LA VENTANA. A LA IZQUIERDA, MUESTRA CON NÚMERO DE PERIODOS NO ENTERO SIN APLICACIÓN DE VENTANA. A LA DERECHA, MISMA MUESTRA CON VENTANA. SE PUEDE VER LA ATENUACION DE LAS ALTAS FRECUENCIAS DEBIDAS A LA DISCONTINUIDAD.....	22
ILUSTRACIÓN 2-4 TIPOS VENTANA Y SU RESPUESTA EN FRECUENCIA.....	23
ILUSTRACIÓN 2-5 TIPOS DE FILTROS ESENCIALES EN EL PROCESAMIENTO DIGITAL DE SEÑALES.....	25
ILUSTRACIÓN 3-1 KIT DE OPENBCI.....	31
ILUSTRACIÓN 3-2 SISTEMA INTERNACIONAL 10-20 PARA LA COLOCACION DE LOS ELECTRODOS EN EL CRANEO.....	32
ILUSTRACIÓN 3-3 INTERFAZ GRAFICA FACILITADA POR OPENBCI. A LA IZQUIERDA SE MUESTRAN LAS SEÑALES EN TIEMPO REAL EN EL DOMINIO DEL TIEMPO. A LA DERECHA ARRIBA SE MUESTRAN LAS ACTIVIDADES CEREBRALES CAPTADAS POR CADA ELECTRODO, Y ABAJO A LA DERECHA SE MUESTRAN LAS SEÑALES EN EL DOMINIO DE LA FRECUENCIA.....	33
ILUSTRACIÓN 3-4 ENSAYO DE CAPTACION DE ONDAS ALFA Y PROCESADAS EN MATLAB. LA IMAGEN DE LA IZQUIERDA CORRESPONDE A LOS DATOS CAPTADOS POR EL ELECTRODO SITUADO EN 01. A LA DERECHA LOS CAPTADOS POR 02. EN AMBAS SE PUEDE APRECIAR CLARAMENTE UN PICO EN LA FRECUENCIA CERCANA A LOS 10 HZ.....	35
ILUSTRACIÓN 3-5 APLICACIÓN DEL ANALISIS TEMPORA. IMAGEN OBTENIDA DEL LIBRO ANALYZING NEURAL TIME SERIES DATA, THEORY AND PRACTICE.....	36
ILUSTRACIÓN 4-1 ELEMENTOS USADOS EN EL SISTEMA DE PROCESAMIENTO DE DATOS.....	41
ILUSTRACIÓN 4-2 SEÑALES DE E/S DEL BLOQUE FFT A GENERAR USANDO LA HERRAMIENTA FACILITADA POR XILINX COREGENERATOR.....	42
ILUSTRACIÓN 4-3 VENTANAS EN LA CONFIGURACIÓN DEL BLOQUE FFT DISEÑADO PARA ESTE TFG.....	44
ILUSTRACIÓN 4-4 SEÑALES DE E/S DEL BLOQUE FFT DISEÑADO PARA ESTE TFG.....	45
ILUSTRACIÓN 4-5 TIMING SIN DETALLES DEL BLOQUE FFT.....	46
ILUSTRACIÓN 4-6 SIMULACION DEL BLOQUE FFT. SE PUEDE APRECIAR COMO LOS TIEMPOS DE CARGA Y DESCARGA SON DIFERENTES DEL TIEMPO DE PROCESAMIENTO , EL CUAL ES BASTANTE MAS LARGO.....	47
ILUSTRACIÓN 4-7 DIAGRAMA FUNCIONAL DEL DISEÑO HARDWARE DEL SISTEMA DESARROLLADO.....	49
ILUSTRACIÓN 4-8-CONEXIONES ENTRE LAS MEMORIAS BRAM Y EL BLOQUE FFT.....	49

Referencias, Ilustraciones, Abreviaturas, Acrónimos y Anexos

ILUSTRACIÓN 4-9-DETALLE DE LA SIMULACION EN EL INICIO DE LA CARGA DE LOS DATOS, CORRESPONDIENTE AL MODO DEL BLOQUE "3 CLOCK CYCLE OFFSET.....	49
ILUSTRACIÓN 4-10 DETALLE DE LA SIMULACION EN EL FIN DE LA DESCARGA DE LOS DATOS, DONDE SE PUEDE VER QUE SE AJUSTA EL SISTEMA PARA PRODUCIR LA SEÑAL FIN, DE UN CICLO DE RELOJ.	50
ILUSTRACIÓN 4-11 APLICACIÓN DE LA VENTANA EN HARDWARE.....	50
ILUSTRACIÓN 5-1 TEST SIGNAL DEL ADS1299.....	55
ILUSTRACIÓN 5-2 CABLE QUE HACE DE ANTENA PARA CAPTAR RUIDO.....	56
ILUSTRACIÓN 5-3 FFT A LA SEÑAL CAPTADA POR LA ANTENA. SE PUEDE APRECIAR UN PICO EN LA FRECUENCIA DE 50 HZ, CORRESPONDIENTE AL RUIDO ELECTRICO DE LA RED. EN EL EJE DE ORDENADAS SE MUESTRA EL LOG DE LA AMPLITUD Y EN EL DE ABSCISAS LA FRECUENCIA EN HZ.....	56
ILUSTRACIÓN 5-4 FUENTE DE TENSIÓN PARA EL ENSAYO. SE PUEDE OBSERVAR QUE LA FUENTE GENERA 1 V Y QUE EL VOLTIMETRO VERIFICA QUE ES 1 V	57
ILUSTRACIÓN 5-5 VALORES DE LA FUENTE DE TENSIÓN CAPTADOS POR EL ADS1299.....	57
ILUSTRACIÓN 5-6 FFT DE ONDA CUADRADA. EN EL EJE DE LAS ORDENADAS SE MUESTRA EL LOG DE LA AMPLITUD EN DBS Y EN EL EJE DE ORDENADAS EL NUMERO DE MUESTRAS	58
ILUSTRACIÓN 5-7 APLICACIÓN DEL SISTEMA DESARROLLADO SOBRE UN EEG REAL. SE PUEDE VER LA FRECUENCIA DE ONDAS ALFA A 11 HZ. EN EL EJE DE ORDENADAS SE REPRESENTA LA AMPLITUD, EN DBS, Y EN EL DE ABSCISAS EL NUMERO DE MUESTRAS	59
ILUSTRACIÓN 5-8 TIEMPO DE EJECUCION DE LA FFT EN HARDWARE	59
ILUSTRACIÓN 5-9 TIEMPO DE SIMULACION DE LA FFT EN HARDWARE	60
ILUSTRACIÓN 5-10 TIEMPO DE SIMULACION DE LA FFT EN SOFTWARE, USANDO MATLAB	60
ILUSTRACIÓN 6-1 FFT REALIZADA EN HARDWARE Y FFT REALIZADA EN MATLAB, AMBAS REPRESENTADAS EN LA MISMA GRAFICA	63
ILUSTRACIÓN 6-2 ZOOM DE LA ILUSTRACIÓN 6.1. EN ROJO, LA FFT REALIZADA EN HARDWARE, Y EN AZUL LA FFT REALIZADA EN MATLAB.....	64
ILUSTRACIÓN 8-1 DIAGRAMA DE GANTT	74

11. ABREVIATURAS Y ACRÓNIMOS

ADC	Analogic to Digital Converter
BCI	Brain Computer Interface
BRAM	Block Random Access Memory
CEI	Centro de Electrónica Industrial
CLB	Configurable Logic Block
CLK	Clock
CPU	Central Processing Unit
CTB	Centro de Tecnología Biomédica
DSP	Digital Signal Processing
DV	Data Valid
EDK	Embedded Development Kit
EEG	Electroencephalogram
EHW	Evolvable Hardware
EMG	Electromyogram
FFT	Fast Fourier Transform
FPGAs	Field Programmable Gate Array
GPUs	Graphics Processor Unit
PLB	Processor Local Bus
RFD	Ready For Data
SASI	Spectral Asymmetry Index
SPI	Serial Peripheral Interface
UPM	Universidad Politécnica de Madrid
VHDL	Very Hardware Description Language
XPS	Xilinx Platform Studio

ANEXO A CÓDIGO IMPLEMENTADO EN C EN EL MICROBLAZE

```
#include <stdio.h>
#include <stdint.h> // int32_t
#include <xparameters.h> // XPAR_*_BASEADDR
#include <math.h> // sin()
#include <xil_cache.h> // Xil_DCacheDisable()

// Librerías Sistema Adquisición de Datos
#include <sysace_stdio.h>
#include "ads1299.h" // librerías ramon
#include "funciones_ads1299.h" // librerías ramon

#define SAMPLES 1024
static int muestras[SAMPLES][9] __attribute__((section(".sbss")));

// Voltmeter
static void print_volts(int channel) {
    const float Vref = 4.5f;
    const int gain = 1;
    const int maxval = (1<<23) - 1;

    int sample[9];
    muestraa(&sample, 1);
    float volts = (Vref * sample[channel]) / (maxval * gain);

    //xil_printf("%f V\n", volts); // xil_printf doesn't support %f
    int uV = volts * 1000000;
    if (uV < 0) xil_printf("-%d.%06d V\n", (-uV)/1000000, (-uV)%1000000);
    else xil_printf("+%d.%06d V\n", (+uV)/1000000, (+uV)%1000000);
}

#define T (1024/8)

volatile int32_t *BRAM_IN_RE = (int32_t *) XPAR_XPS_BRAM_IF_CNTL_IN_RE_BASEADDR;
volatile int32_t *BRAM_OUT_RE = (int32_t *) XPAR_XPS_BRAM_IF_CNTL_OUT_RE_BASEADDR;
volatile int32_t *BRAM_OUT_IM = (int32_t *) XPAR_XPS_BRAM_IF_CNTL_OUT_IM_BASEADDR;
volatile int32_t *BRAM_WINDOW = (int32_t *) XPAR_XPS_BRAM_IF_CNTL_WINDOW_BASEADDR;
volatile int32_t *FFT = (int32_t *) XPAR_PRUEBA_FFT_0_BASEADDR;

int main(void)
{
    Xil_DCacheDisable(); // just in case

    float tiempo;

    xil_printf("\n\n*** START ***\n");
    xil_printf("Iniciando ADS1299\n");
    initADS();

    xil_printf("Enviando reset\n");
    pinReset();
    xil_printf("Configurando\n");
    Config1();

    // Voltmeter
    //while (1) print_volts(2);

    xil_printf("Recibiendo %d muestras\n", SAMPLES);
    muestraa(muestras, SAMPLES);

    xil_printf("Muestras recibidas! Escribiendo\n");
    finishADS();

    /*SYSACE_FILE *file;
    file = sysace_fopen("EEGtime.BIN", "w");
    sysace_fwrite(muestras, 1, sizeof muestras, file);
    sysace_fclose(file);
    xil_printf("Escritura OK.\n");*/

    //if 0
    xil_printf("Procesando \n");

    int i;

    for (i=0; i<1024; i++) {
        BRAM_WINDOW[i] = (0.5f - 0.5f * cosf(i*2*(float)M_PI/ 1024)) * (1<<16); // Hann window, pre-multiplied by 2^16
        //BRAM_WINDOW[i] = 1<<16; // constant 1, pre-multiplied by 2^16
    }
}
```

```

//#if 0
xil_printf("Procesando \n");

int i;

for (i=0; i<1024; i++) {
    BRAM_WINDOW[i] = (0.5f - 0.5f * cosf(i*2*(float)M_PI/ 1024)) * (1<<16); // Hann window, pre-multiplied by 2^16
    //BRAM_WINDOW[i] = 1<<16; // constant 1, pre-multiplied by 2^16
}

for (i=0; i<SAMPLES; i++) {
    BRAM_IN_RE[i] = muestras[i][2]; // Datos del canal 2
}

startTimer();
resetTimer();

*FFT = 1;
while (*FFT != 0) {

    unsigned int t = getTimer();
    stopTimer();

    xil_printf("Hecho. %d us (%d ciclos)\n", t / (XPAR_TMRCTR_0_CLOCK_FREQ_HZ/1000000), t);
    //xil_printf("ventanadoble = [\n");
    //for (i=0; i<1024; i++) xil_printf("%d\n", BRAM_WINDOW[i]);
    //xil_printf("];\n");

    //xil_printf("entrada = [\n");
    //for (i=0; i<1024; i++) xil_printf("%d\n", BRAM_IN_RE[i]);
    //for (i=0; i<1024; i++) xil_printf("%d\n", muestras[i][1]);
    //xil_printf("];\n");

    /*!!! file = sysace_fopen("FFTre.BIN", "w");
    sysace_fwrite(BRAM_OUT_RE, 1, SAMPLES * sizeof *BRAM_OUT_RE, file);
    sysace_fclose(file);
    xil_printf("Escritura FFT_re OK.\n");

    file = sysace_fopen("FFTim.BIN", "w");
    sysace_fwrite(BRAM_OUT_IM, 1, SAMPLES * sizeof *BRAM_OUT_IM, file);
    sysace_fclose(file);
    xil_printf("Escritura FFT_im OK.\n");*/

    unsigned int t = getTimer();
    stopTimer();

    xil_printf("Hecho. %d us (%d ciclos)\n", t / (XPAR_TMRCTR_0_CLOCK_FREQ_HZ/1000000), t);
    //xil_printf("ventanadoble = [\n");
    //for (i=0; i<1024; i++) xil_printf("%d\n", BRAM_WINDOW[i]);
    //xil_printf("];\n");

    //xil_printf("entrada = [\n");
    //for (i=0; i<1024; i++) xil_printf("%d\n", BRAM_IN_RE[i]);
    //for (i=0; i<1024; i++) xil_printf("%d\n", muestras[i][1]);
    //xil_printf("];\n");

    /*!!! file = sysace_fopen("FFTre.BIN", "w");
    sysace_fwrite(BRAM_OUT_RE, 1, SAMPLES * sizeof *BRAM_OUT_RE, file);
    sysace_fclose(file);
    xil_printf("Escritura FFT_re OK.\n");

    file = sysace_fopen("FFTim.BIN", "w");
    sysace_fwrite(BRAM_OUT_IM, 1, SAMPLES * sizeof *BRAM_OUT_IM, file);
    sysace_fclose(file);
    xil_printf("Escritura FFT_im OK.\n");*/

    //xil_printf("salida = [\n");
    //for (i=0; i<1024; i++) xil_printf("%d+%di\n", BRAM_OUT_RE[i], BRAM_OUT_IM[i]);
    //xil_printf("];\n");

    // plot(0:1023,abs(salida),'.-'); plot(0:1023,ventana);

    // Bug fix
    while (1) { }

}

//#endif // 0

xil_printf("FIN\n");
return 0;
}

```

ANEXO B: ENSAYOS EN MATLAB EN LA INVESTIGACIÓN PREVIA.

Las siguientes ilustraciones fueran tomadas en la fase de investigación previa, y aunque no fueron determinantes en el devenir del desarrollo del sistema, es interesante la información que se puede obtener de ellas.

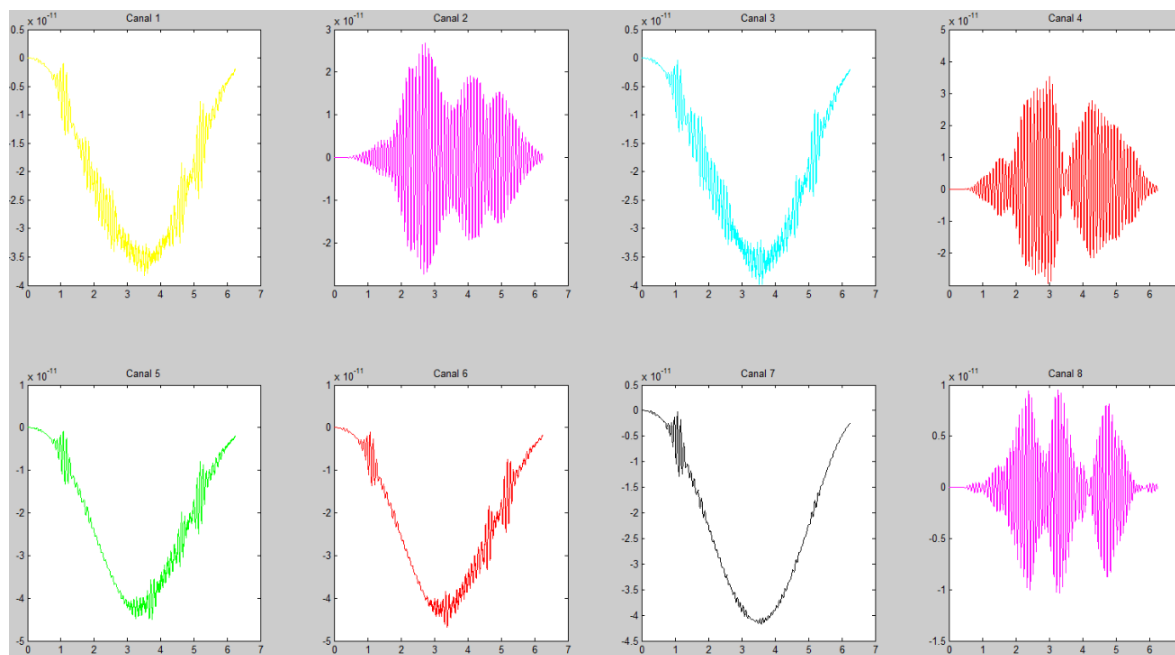


Ilustración 12.0.1 Señal temporal con filtrado Notch y pasabajo. La ventana usada es de tipo Hann. En los canales 2, 4 y 8 los electrodos están colocados en la corteza craneal. Los demás están al aire.

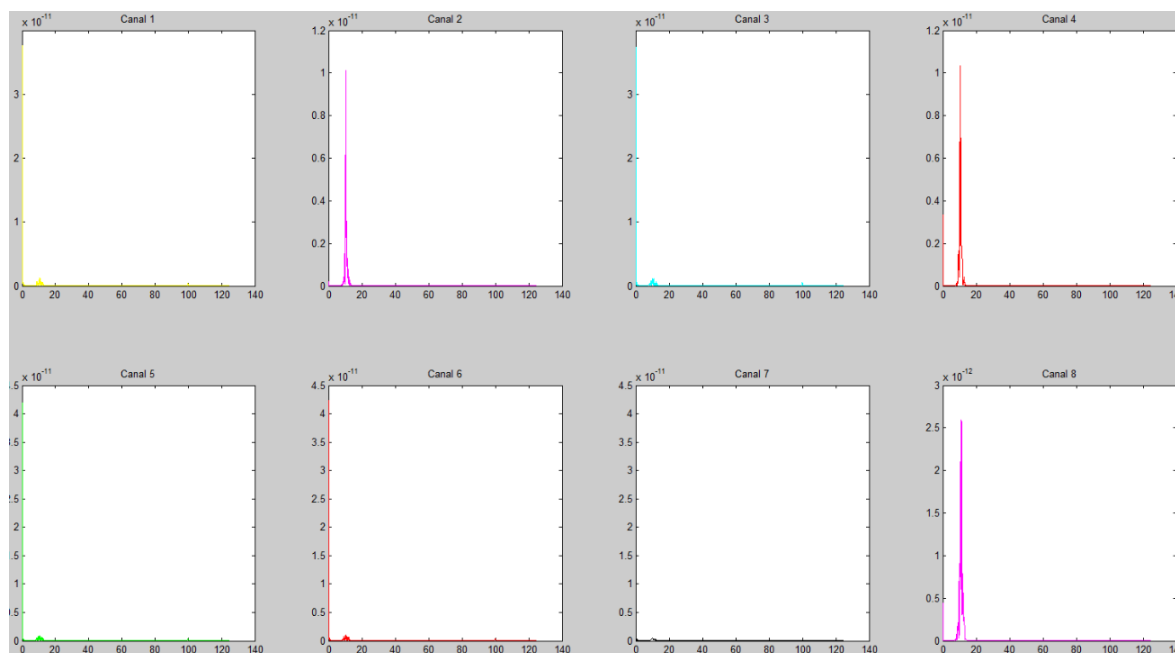


Ilustración 12.0.2 FFT a la señal filtrada con Notch y Pasa banda. Se puede observar el efecto del filtro pasa banda en la creación de un sistema de detección de picos, ya que deja pasar la frecuencia del rango entre 7 y 13 Hz.

ANEXO C MODELO DEFINITIVO XPS.

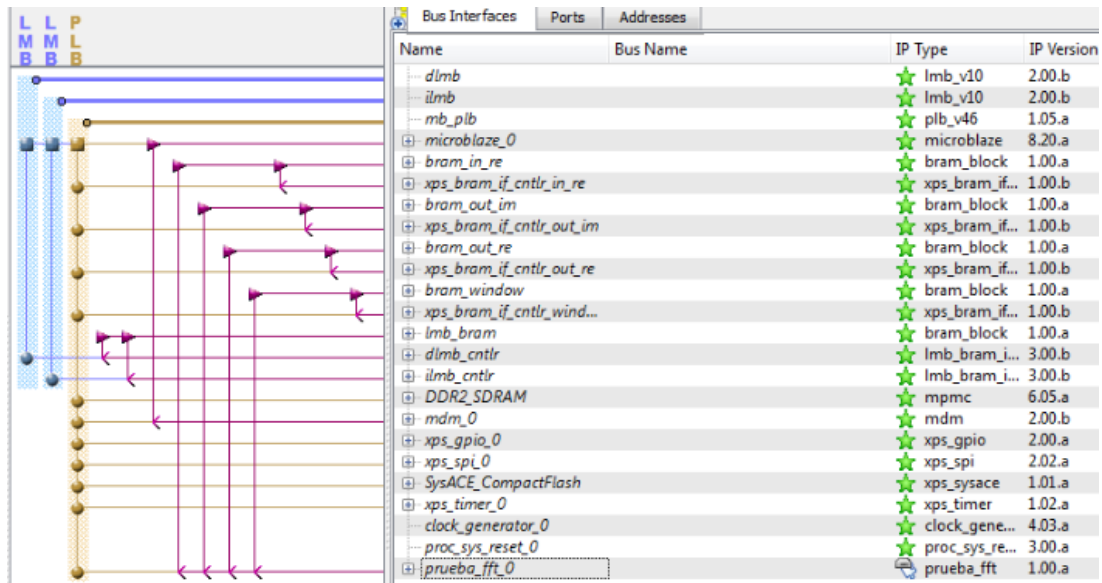


Ilustración 12.0.1 Modelo definitivo XPS.