

**DESARROLLO DE UN PROTOTIPO BCI INTEGRABLE A UNA RED DE
INTERNET DE LAS COSAS PARA INTERACTUAR CON UN ENTORNO
INTELIGENTE**

ALEXANDER FARADID PIÑANGO MARTÍNEZ

REALIZADO CON LA ASESORÍA DE:

ÓSCAR MAURICIO ARIAS BALLÉN

UNIVERSIDAD EL BOSQUE
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
SEPTIEMBRE, 2022

UNIVERSIDAD EL BOSQUE
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA

ÁREA DE ÉNFASIS: EQUIPOS MÉDICOS

**DESARROLLO DE UN PROTOTIPO BCI INTEGRABLE A UNA RED DE
INTERNET DE LAS COSAS PARA INTERACTUAR CON UN ENTORNO
INTELIGENTE**

ALEXANDER FARADID PIÑANGO MARTÍNEZ

REALIZADO CON LA ASESORÍA DE:

ÓSCAR MAURICIO ARIAS BALLÉN

Página de Aprobación. Inclusión de Acta de grado.

NOTA DE SALVEDAD

Según el artículo 37 del 14 de diciembre de 1989 del acuerdo 017, “La Universidad El Bosque, no se hace responsable de los conceptos emitidos por los investigadores en su trabajo, solo velará por el rigor científico, metodológico y ético del mismo en aras de la búsqueda de la verdad y la justicia”.

DEDICATORIA

“El mundo está en manos de aquellos que tienen el coraje de soñar y de correr el riesgo de perseguir sus sueños”.

Paulo Coelho - El Alquimista

Como persona y como ser humano:

Le dedico mi proyecto de grado primeramente a mi mamá, papá y hermana, mi núcleo familiar, por ser los seres detrás del día a día para poder lograr cada reto que me propongo. Han sido, son y seguirán siendo el eje para esforzarme en todo lo hago, buscando siempre hacer lo mejor posible.

También, a mi familia, que a pesar de la distancia siempre tengo presente. Me han enseñado y me seguirán enseñando el valor de hacer las cosas por el camino que creas correcto, sin hacerle daño a los demás, y procurando disfrutar cada pequeño detalle.

A cada persona con la que he compartido momentos en la vida, le agradezco porque sin ellos no hubiera llegado a ser lo que soy.

A todos aquellos que luchan por sus sueños y creen en la misteriosa y maravillosa ciencia, los motivo para que nunca se rindan sin importar los retos que la vida ponga en el camino. Todos los esfuerzos, eventualmente, darán sus frutos, y a través de la ciencia poco a poco podemos entender un pedacito más del Universo.

Serendipity!

AGRADECIMIENTOS

A mi madre, Olivia, por haber sido y seguir siendo el pilar fundamental de los valores, la educación y la formación en mi vida.

A mi padre, Alexander, por sembrar en mí la curiosidad y motivarme cada día a ser la mejor versión de la persona que quiero y puedo ser.

A mi hermana, Almahir, por absolutamente cada momento y cada segundo que compartimos. Eres mi bendición más grande, y la guía que el Universo me dio para ser feliz.

Todo lo que soy y lo que he logrado es gracias a ustedes.

RESUMEN

El presente proyecto se enfocó dentro del área de los sistemas BCI orientados a la asistencia para personas con patologías del sistema nervioso que ocasionan dificultades motoras severas. El foco principal fue desarrollar un sistema que permitiera identificar comandos que una persona selecciona mentalmente mediante una pantalla de estímulos visuales, y posteriormente ejecutar el comando identificado mediante la comunicación con una red IoT que contenga dispositivos conectados en un entorno de prueba (entorno local).

El sistema global consistió en la integración de un dispositivo que realizara la adquisición de señales EEG basado en el ADS1299 junto con algoritmos de análisis y procesamiento de datos, de manera que mediante la identificación de ERPs que estuvieran asociados a estímulos visuales, se pudieran detectar componentes P300 que representaran los comandos específicos seleccionados previamente por la persona a través de una matriz de estímulos visuales.

Se utilizó Python para el desarrollo y construcción de todos los algoritmos de análisis de datos, interfaces gráficas y enlaces de comunicación con los dispositivos de la red IoT. Con ello, la identificación de los ERP asociados a los comandos alcanzó un nivel de acierto mayor al 80% bajo ciertas condiciones de operación, lo cual permitió transmitir correctamente los comandos hacia una red con dispositivos de prueba para realizar acciones como enviar mensajes por la aplicación de WhatsApp hacia un teléfono registrado, o ejecutar las funciones básicas de un SmartTV (cambiar de canal, subir el volumen, lanzar una app específica, entre otras).

Palabras Clave: BCI, ERP, P300, IoT.

ABSTRACT

The present project was focused on the area of the BCI systems oriented to assist people with nervous system's pathologies that cause severe motor disabilities. The focus was to develop a system capable of identify commands mentally selected by a person through a visual stimulus screen, and then and then execute the command identified by an IoT network that contains connected devices inside a test environment (local environment).

The overall system consisted of the integration of a device that performed the acquisition of EEG signals based on the ADS1299 along with data analysis and processing algorithms, so that by identifying ERPs that were associated with visual stimuli, P300 components representing the specific commands previously selected by the person through a matrix of visual stimuli could be detected.

Python was used for the development and construction of all the data analysis algorithms, graphic interfaces and communication links with the IoT network devices. With this, the identification of the ERPs associated with the commands reached a level of accuracy greater than 80% under certain operating conditions, which allowed to correctly transmit the commands to a network with test devices to perform actions such as sending messages through the WhatsApp application to a registered phone, or execute the basic functions of a Smart TV (change channel, turn up the volume, run a specific app, among others).

Keywords: BCI, ERP, P300, IoT.

GLOSARIO DE TÉRMINOS

- Latencia: en el campo de la Ingeniería, es el período de tiempo existente entre la causa y el efecto de algún cambio físico en un sistema que está siendo observado.
- Paradigma experimental: en el campo de la psicología, representa un modelo o patrón de procedimientos estándar que dirigen el desarrollo de una investigación científica experimental.
- Epoch: representa un fragmento determinado de un EEG, normalmente delimitado por marcadores que se colocan en las mediciones.

LISTA DE ABREVIATURAS

- UI: User Interface.
- GUI: Graphic User Interface.
- BCI: Brain-Computer Interface.
- IoT: Internet of Things.
- ERP: Event-Related Potential.
- FFT: Fast Fourier Transform.
- ADC: Analog-to-Digital Converter.
- PGA: Programmable Gain Amplifier.
- SPS: Samples per Second.
- LPF: Low-pass Filter.
- HPF: High-pass Filter.
- ECG: Electrocardiography.
- EOG: Electrooculography.
- EMG: Electromyography.
- API: application programming interface.

TABLA DE CONTENIDOS

Pág.

DEDICATORIA	v
AGRADECIMIENTOS	vi
RESUMEN	vii
ABSTRACT.....	viii
GLOSARIO DE TÉRMINOS	ix
LISTA DE ABREVIATURAS	x
LISTA DE TABLAS	xv
LISTA DE FIGURAS.....	xvi
INTRODUCCIÓN	21
1. PLANTEAMIENTO DEL PROBLEMA	22
1.1 Antecedentes y Estado del Arte	22
1.1.1 Nivel Internacional	23
1.1.2 Nivel Nacional	27
1.2 Descripción y Formulación del Problema	30
1.3 Justificación	32
1.4 Objetivos	34
1.4.1 Objetivo General.....	34
1.4.2 Objetivos Específicos	34
1.5 Requerimientos	35
1.5.1 Requerimientos del usuario	35
1.5.2 Requerimientos funcionales.....	35
1.5.3 Requerimientos de calidad.....	35

1.5.4	Requerimientos restrictivos	36
1.6	Alcance y Limitaciones del Proyecto	36
2.	MARCO DE REFERENCIA.....	39
2.1	Marco Teórico.....	39
2.2	Estructura y Funciones Generales del Sistema Nervioso	39
2.3	Sistema Nervioso Central (SNC).....	41
2.3.1.1	El Cerebro	42
2.4	Sistema Nervioso Periférico (SNP)	45
2.5	Neuronas	46
2.6	Glía o neuroglia	49
2.7	Sinapsis y Transmisión Sináptica	49
2.8	Potencial de Acción	51
2.9	Ondas Cerebrales y el EEG	53
2.10	Brain-Computer Interface (BCI).....	55
2.11	Event-Related Potential (ERP)	57
2.12	El Paradigma Oddball.....	59
2.13	P300	60
2.14	P300-Speller y sistemas BCI basados en P300.....	64
2.15	PCA (Principal Component Analysis).....	66
2.16	LDA (Linear Discriminant Analysis)	68
2.17	Artefactos.....	70
2.18	Marco Legal o Normativo.....	71
	3METODOLOGÍA	72
3.1	Diseño general del prototipo.....	73
3.2	Diseño funcional	73
3.2.1	Interfaz de estímulos visuales	74
3.2.2	Adquisición de señales EEG.....	75
3.2.3	Acondicionamiento	75
3.2.4	Amplificación	75
3.2.5	Conversor Análogo-Digital (ADC)	76
3.2.6	Comunicación con PC	76

3.2.7UI (<i>User Interface</i> o Interfaz de Usuario)	76
3.2.8Procesamiento de datos.....	76
3.2.9Preprocesamiento.....	76
3.2.10Filtrado de banda EEG.....	77
3.2.11Filtros para artefactos.....	77
3.2.12Clasificación	77
3.2.13Detección de comandos	77
3.2.14Red IoT (<i>Internet of Things</i>).....	78
3.2.15Menú de opciones disponibles.....	78
3.2.16Validación de comando identificado	78
3.2.17Transmisión de comando a dispositivo de prueba	78
3.3Diseño detallado	79
3.3.1Elección de características	79
3.3.2Adquisición de señales EEG.....	81
3.3.2.1Selección de bases de datos	81
3.3.2.2Selección de equipos (Hardware)	82
3.3.2.2.1 ADS1299	84
3.3.2.2.2 Tarjeta de Adquisición: ADS1299EEG-FE.....	87
3.3.2.3Selección y ubicación de electrodos	89
3.3.2.3.1 Sistema Internacional 10/20.....	91
3.3.2.4Transmisión de datos a Python (Windows)	92
3.3.2.4.1 Comunicación con el ADS1299 por SPI	93
3.3.2.4.2 Lectura de mediciones desde Python.....	96
3.3.2.5Pruebas de funcionalidad de equipos y mediciones preliminares.....	99
3.3.2.5.1 Pruebas con bases de datos	99
3.3.2.5.2 Pruebas con señales EEG adquiridas	102
3.3.3Interfaz de estímulos visuales en OpenSesame	105
3.3.4UI principal en KivyMD/Python	109
3.3.5Procesamiento de señales.....	112
3.3.5.1Preprocesamiento y filtrado	112
3.3.5.2Cálculo del ERP.....	117

3.3.5.3Clasificación e identificación de comandos.....	120
3.3.6Red IoT	124
4RESULTADOS Y ANÁLISIS DE RESULTADOS	128
5CONCLUSIONES	132
6RECOMENDACIONES	133
7REFERENCIAS BIBLIOGRÁFICAS.....	134
8ANEXOS.....	141
8.1Anexo A: Tabla comparativa de los dispositivos de adquisición de señales EEG.....	141
8.2Anexo B: Manual de Usuario	142
8.3Anexo C: Web del Proyecto	167

LISTA DE TABLAS

Tabla	Pág.
Tabla 1. Bandas frecuenciales más comunes en un EEG de un paciente sano [21]-[31].	..54
Tabla 2. Tasa de aciertos para las sesiones realizadas con las interfaz de las banderas (Flag identification).....	129
Tabla 3. Tasa de aciertos para las sesiones realizadas con las interfaz de comandos (Command identification).....	129

LISTA DE FIGURAS

Figura	Pág.
Figura 1. Diagrama ilustrativo de las tres funciones básicas del Sistema Nervioso: a) adquirir información sensorial; b) analizar la información obtenida; c) determinar una respuesta motora. Extraído de [26].40	40
Figura 2. Diagrama de bloques de un computador genérico, con sus componentes básicos. La analogía con el sistema nervioso busca resaltar las funciones básicas de adquisición de datos, procesamiento de información y toma de decisiones. Imagen extraída de [27].....40	40
Figura 3. Representación ilustrativa de las principales partes del encéfalo. Imagen extraída de [25] 41	41
Figura 4. Principales áreas funcionales de la corteza cerebral, distinguiendo las regiones asociadas a cada función conocida. Imagen extraída de [25]43	43
Figura 5. Gráfico típico de una neurona, con la sinapsis que recibe. Imagen extraída de [28] 46	46
Figura 6. Potenciales postsinápticos excitatorios. Se muestran los registros de potenciales postsinápticos adquiridos con un estímulo de intensidad creciente. Imagen extraída de [30] 51	51
Figura 7. Potencial de acción típico que puede ser observado a través de distintos mecanismos de medición electroencefalográfica. Se muestran además las fases sucesivas del potencial de acción: el reposo (resting), la despolarización y la repolarización. Imagen extraída de [27].52	52
Figura 8. Ejemplo ilustrativo del comportamiento de mediciones EEG mostrando la disminución y aumento de voltaje de la señal al abrir y cerrar los ojos, respectivamente. Imagen extraída de [27]55	55

Figura 9. Diagrama general de una BCI, mostrando los módulos y partes principales. Imagen basada en el diagrama mostrado en [32].	56
Figura 10. Formas de onda ilustrativas que ejemplifican algunos componentes de ERP comúnmente encontrados en estudios de neurociencia. La amplitud de las señales es de referencia (puede variar), pero se puede observar los distintos picos que se pueden asociar a un estímulo aplicado. Imagen extraída de [38].	59
Figura 11. Ilustraciones de referencia para ejemplificar un experimento simple de P300- Odball. La figura 11-A representa la pantalla con los estímulos visuales presentados, la cual se conecta al equipo de procesamiento de datos, filtrado y adquisición (EEG). La figura 11- B representa las mediciones EEG. La figura 11-C(izquierda) se refiere a ciertos Epochs del EEG, y a la derecha están los ERP asociados a los dos estímulos posibles. Imagen extraída de [32].	62
Figura 12. Tres protocolos conocidos para generar respuestas P300. Imagen extraída de [39].	63
Figura 13. Ejemplo de una matriz 6x6 utilizada como fuente de los estímulos visuales para un Speller basado en P300. Imagen extraída de [39].	65
Figura 14. Diagrama de la metodología general con las etapas principales para el desarrollo del proyecto.	72
Figura 15. Diagrama de la función de transferencia general (caja negra) del prototipo propuesto.	73
Figura 16. Diseño funcional del proyecto.	74
Figura 17. Diseño detallado del prototipo propuesto.	79
Figura 18. Diagrama de bloques funcional del ADS1299. Imagen extraída del datasheet del fabricante [56].	84
Figura 19. Diagrama de ejemplo de un conversor $\Delta\Sigma$. Imagen extraída de [52].	85

Figura 20. Comandos preconfigurados en el ADS1299 para realizar las operaciones principales por SPI. Imagen extraída de [54].	86
Figura 21. Kit de desarrollo y evaluación ADS1299EEG-FE. Imagen extraída de [55]...	87
Figura 22. Diagrama de bloques del ADS1299EEG-FE. Imagen extraída de [55].	88
Figura 23. Gold cup electrodes de OpenBCI. Imagen extraída de [57].....	89
Figura 24. Pasta Ten20 para adquisición de señales EEG. Imagen extraída de [58].....	90
Figura 25. Posiciones de los electrodos para medición EEG, junto con los lóbulos asociados del cerebro. Imagen extraída de [59].	91
Figura 26. Localizaciones escogidas para los electrodos del prototipo: Cz y Pz, según el sistema 10/20. Imagen base extraída de [60].	92
Figura 27. SPI Bus Data Output. Imagen extraída de [54].	94
Figura 28. (Izquierda) Imagen tomada del ADS1299EEG-FE conectado por SPI al Arduino. (Derecha) Diagrama ilustrativo del flujo de los datos y conexiones para la transmisión de las señales hacia Python. Fuente: el autor.....	95
Figura 29. Configuraciones aceptadas por el ADS1299EEG-FE. (a) Entrada diferencial. (b) Entrada unipolar (Single-ended). Imagen extraída de [55].....	97
Figura 30. (a) Sección de código que muestra la importación de librerías y definición de las cartas parte del experimento. (b) Visualización del tipo y tamaño de las variables en la base de datos, importadas en Spyder.	99
Figura 31. (a) (Imagen superior) Gráfica de las mediciones raw. (b) (Imagen inferior) Gráfica de las mediciones distribuidas en sus respectivos canales, como un EEG.	100
Figura 32. (a) ERP para cada carta, organizados por canal. (b) Zoom de los ERP para los canales Cz y Pz. El ERP para cada carta corresponde a un color diferente (9 colores diferentes).	101

Figura 33. Puntuaciones para cada carta, que se asocian a la probabilidad de que dicha carta haya sido la escogida por la persona.....	102
Figura 34. Interfaz del software de evaluación de TI, al conectar la tarjeta. Fuente: el autor. 102	
Figura 35. (a) Respuesta del Input-short test. (b) Mapa de los registros para realizar la prueba. Fuente: el autor.	103
Figura 36. (a) Gráfica en el dominio del tiempo del CH1. (b) FFT de la señal del CH1. Fuente: el autor.	104
Figura 37. (a) Gráfica en el dominio del tiempo del CH1. (b) FFT de la señal del CH1. Fuente: el autor.	105
Figura 38. Página principal al abrir OpenSesame. Fuente: el autor.	106
Figura 39. Interfaz de estímulos visuales para las pruebas de Flag identification. Fuente: el autor. 107	
Figura 40. Interfaz de estímulos visuales para las pruebas de Command identification. Fuente: el autor.	107
Figura 41. Secuencia de las instrucciones para la interfaz, luego de las configuraciones. Fuente: el autor.	108
Figura 42. Opciones disponibles en la interfaz, divididas en pestañas (tabs). La pestaña actualmente seleccionada (activa) se resalta con el nombre de color negro y una línea amarilla debajo del nombre. Fuente: el autor.	110
Figura 43. Ejemplos de gráficas realizadas en la interfaz con datos EEG adquiridos. (a) Gráficas en el dominio del tiempo. (b) Gráficas en el dominio frecuencial. Fuente: el autor. 111	
Figura 44. Gráfica en el tiempo de una prueba de Ojos abiertos-ojos cerrados, vista desde Python mediante la configuración del modo en tiempo real. Fuente: el autor.	116

Figura 45. Gráfica en el tiempo de un canal EEG, junto con los marcadores (líneas rojas) que indican el momento en que fue mostrado cada estímulo. Fuente: el autor.	118
Figura 46. Gráfica de los ERPs para un experimento realizado con la interfaz de las banderas. Cada ERP está representado con un color diferente, identificado en la parte superior derecha. Fuente: el autor.	120
Figura 47. Diagrama funcional del clasificador construido. El orden de lectura y ejecución del algoritmo es de izquierda a derecha. Fuente: el autor.	121
Figura 48. Opciones enumeradas de la interfaz de estímulos visuales. Fuente: el autor.	125
Figura 49. Diagrama ilustrativo con algunas de las opciones disponibles desde Twilio. Imagen extraída de [65].	126
Figura 50. Diagrama de la red IoT construida para el prototipo. Fuente: el autor.	127

INTRODUCCIÓN

La existencia de patologías asociadas al sistema nervioso que afectan severamente la movilidad de personas en el mundo es diversa. Desde la Esclerosis Lateral Amiotrófica (ELS), que representa una de las discapacidades motoras-neuronales más comunes en adultos mayores de 60 años, pasando por la Distrofia Muscular de Duchenne, e incluso pacientes de Accidentes Cerebro Vasculares (ACV), la cantidad de personas con problemas de movilidad causados por daños moto-neuronales severos es un factor altamente considerable, y más teniendo en cuenta que, según las expectativas de la UN y los Consejos de Economía Mundial, la proyección de población vulnerable a éstos padecimientos se incrementará por lo menos en un 43% para el 2025. De hecho, se estima que solamente los casos de ELS aumentarán en un 69% para el 2040.

Lo cierto es que, la esperanza de vida y tratamiento de los pacientes y la población vulnerable a estas patologías pone en evidencia la necesidad de implementar herramientas para mejorar la calidad de vida de las personas involucradas en ellas, debido a que los procesos degenerativos de los padecimientos generan efectos negativos muy significativos en el desenvolvimiento de actividades del día a día. En efecto, los casos de estudio muestran que, más allá de la inexistencia de una cura definitiva, las personas que padezcan de estas condiciones necesitan no sólo de tratos especiales muy delicados, sino de seguimientos de su salud periódicos que los obligan a quedarse en entornos controlados, como las habitaciones en hospitales, clínicas o del hogar propio (o de algún familiar).

Teniendo en cuenta las ideas anteriores, mejorar la calidad de vida de las personas dentro de su entorno sería un factor profundamente beneficioso para mejorar aspectos fisiológicos, emocionales y sociales. Si los pacientes pudieran interactuar con su entorno de forma más independiente y libre, logrando cierto grado de autonomía, su salud mental y calidad de vida en general mejorarían considerablemente. En ese ámbito, el presente proyecto se enfoca dentro del campo de los sistemas BCI como alternativas asistenciales para estos casos, donde además se propone la integración con una red IoT que permita interactuar y controlar en cierto grado los dispositivos inteligentes en el entorno del paciente, buscando facilitar la ejecución de tareas simples.

1. PLANTEAMIENTO DEL PROBLEMA

1.1 Antecedentes y Estado del Arte

La investigación en el mundo realizada con dispositivos BCI que analicen registros EEG es bastante amplia, debido a los múltiples beneficios y usos que se han encontrado [1]. Desde instituciones educativas hasta centros médicos, en los últimos años la búsqueda e implementación de tecnologías que integren estos sistemas ha venido en aumento, con lo cual distintas organizaciones se han enfocado en la fabricación de equipos de alta fidelidad para la lectura de las señales EEG, pero en lo referente a las aplicaciones como tal, la integración de estos dispositivos todavía es materia de investigación.

Cabe destacar (para fines explicativos) que el foco principal del proyecto desarrollado no fue explícitamente la comunicación o transmisión directa de mensajes por parte del paciente o sujeto de prueba, sino la posibilidad de la interacción con el entorno del mismo. Sin embargo, dentro del abanico de potenciales aplicaciones siempre están los mensajes simples y similares herramientas que permitan eliminar las barreras entre el paciente y su autonomía. Así, teniendo en cuenta la gravedad de varias de las patologías[2], el desarrollo de técnicas y herramientas para ayudar en estos casos muchas veces se orienta exclusivamente a la comunicación, siendo parte fundamental para el desarrollo de la vida de cualquier persona, pero se dejan de lado todas las implicaciones respecto a la calidad de vida que deberían tener los pacientes, lo cual ha llevado a entender grandes efectos que impactan en la salud y en la vida misma de la persona. Por lo anterior, se han venido desarrollando alrededor del mundo cada vez más investigaciones que permitan minimizar dichos efectos, mediante diversas herramientas que se adaptan a las necesidades de los pacientes.

Teniendo en cuenta las ideas anteriores, distintos tipos de dispositivos han sido desarrollados dentro del campo de las BCI y la neurociencia, abordando desde interfaces que buscan diagnosticar o probar conexiones simples, hasta soluciones mucho más complejas que implican un gran procesamiento de señales e información. Así pues, a continuación se mencionarán algunas de las investigaciones y trabajos más influyentes para este proyecto:

1.1.1 Nivel Internacional

La gran mayoría de los avances y desarrollos similares para la neurociencia provienen de la región europea, Norteamérica y Asia. Específicamente para P300, existen documentos, tutoriales y guías interactivas como las brindadas por Marijin van Vliet, ingeniero de Software y Bioingeniería, doctor en neurociencia e investigador de distintos tipos de procesos y señales asociadas con el cerebro humano. Buena parte de sus investigaciones [3][4], sirvieron como base de referencia para entender la fenomenología y aspectos relacionados sobre todo con el tratamiento de la señal, filtrado e interpretación de los datos, lo cual fue clave para analizar no sólo los aspectos fisiológicos, sino cómo manejar los fenómenos electroencefalográficos y asociarlos con algoritmos computacionales, de manera que se puedan modelar los experimentos y entender los resultados, extrayendo la información deseada. Los documentos además sirven como diagramas para plantear los tratamientos de señales cerebrales como P300 y N400, pero además permiten entender la ciencia detrás de los datos y buenas prácticas para poder lograr los mejores resultados posibles.

Respecto a Interfaces Cerebro-Computador (en adelante BCI) integradas o aplicadas a un entorno “inteligente”, los trabajos de Lin C, Lin B, [...] Chang C [5] comprenden un prototipo integrando un protocolo BCI a un sistema de Control utilizando el Universal Plug and Play (*UPnP*), tomando señales electroencefalográficas (en adelante EEG) para monitorear en tiempo real el estado mental de la persona, modificando ciertos elementos del entorno de acuerdo con las emociones detectadas por el sistema. Los autores, todos de la región de China, probaron el sistema en una habitación prototipo, con resultados satisfactorios en aplicaciones de prueba para *Smart-home* (Hogar inteligente).

Peng H, Hu B, [...] Ratcliffe M[6] realizaron un proyecto presentado en una conferencia en Dublín (Irlanda), donde el equipo de trabajo se enfocó en desarrollar un sistema para mejorar y monitorear la salud mental de pacientes en sus casas, buscando realizar registros desde sus casas que puedan detectar rasgos de trastornos

mentales o depresión, basándose en dispositivos de registro ocular (*Ocular Artifacts*) junto con señales EEG.

Duru D, Deniz Duru A, [...] Ozkan M en sus investigaciones principalmente de carácter médico [7], analizan el comportamiento mental del paciente durante una nefrectomía, procedimiento quirúrgico que se realiza para remover total o parcialmente un riñón, ya sea por un mal funcionamiento o presencia de tumores benignos. Dichos procedimientos son operaciones de alto estrés que requieren mucha concentración y experiencia, y debido a eso plantearon el proyecto, para poder detectar en tiempo real el nivel de estrés en la sala de cirugía, de manera que con las señales EEG se pueda monitorear en tiempo real las emociones durante el procedimiento, y en base a ello, generar alarmas y que ciertas condiciones de la sala de operación se adapten para disminuir los niveles estrés.

Navarro A, Ceccaroni L, [...] Faller J por su lado[8], presentan un proyecto que se enfoca en la creación de una aplicación que es “consciente del entorno” al introducir el concepto de *Ambient intelligent (AmI)*, donde se presentan varias posibilidades de aplicación: reconocimiento de actividades del día a día para prevenir posibles emergencias, atención especializada para ayudar a personas con movilidad reducida a mejorar su autonomía en las tareas que realiza, e incluso se plantea la posibilidad de una arquitectura multiplataforma (que pueda incluir instituciones asistenciales de salud, pacientes y sus familiares dentro de una misma red), de forma que se le da cierta inteligencia al ambiente, ofreciendo por ejemplo una personalización dinámica de las opciones disponibles en la interfaz con el usuario, dependiendo del contexto actual del mismo (o sus emociones). Dicho proyecto fue uno de los pioneros en la creación de *BrainAble*, un proyecto fundado por la Comisión Europea (FP7 mediante el acuerdo n° 247447) que busca mejorar la calidad de vida de las personas enfocándose en dos de los mayores retos que sufren: la exclusión de las actividades del hogar y de las actividades sociales, y todo basándose en la utilización de una BCI integrada a diversos dispositivos del entorno del paciente. Asimismo, un entorno de realidad virtual (VR) de prueba se presenta como módulo de ejemplo para mostrar la

arquitectura de todo el sistema, mostrando algunos controladores de comunicación que puedan conectarse a sensores de clima, interruptores de luz, servicios audiovisuales, entre otros.

Dentro del campo de la utilización de BCI para pacientes que ya tienen ciertas patologías presentes, una investigación bastante interesante es la de Wolpaw J, Bedlack R, [...] Ruff R [9], la cual centra su atención en personas con Esclerosis Lateral Amiotrófica (ELA por sus siglas en español o ALS por sus siglas en inglés) que aceptaron ser usuarios de aplicaciones BCI desde sus casas, de manera que ellos pudieran probar diversas aplicaciones durante varios meses de uso y luego recopilar los resultados. Inicialmente, los pacientes fueron entrenados (además de los algoritmos asociados) con sus respectivas interfaces BCI en sus casas, mediante visitas programadas para realizar sesiones de práctica. Luego, dichos pacientes entraron en la fase de 12 o 18 meses de utilización de los aplicativos, como parte del estudio en cuestión, período durante el cual todos los datos de uso de la BCI (incluyendo actividad EEG) eran automáticamente guardados diariamente mediante una conexión a internet segura, además de que ocasionalmente se hacían visitas para mostrar ciertas actualizaciones, o periódicamente (cada 3 meses) personal del estudio realizaba sesiones con los pacientes para atender posibles preguntas o simplemente tener algún feedback de la BCI y su utilización. La interfaz, a grandes rasgos, contaba con una matriz de caracteres seleccionables para que el paciente pudiera escribir algún mensaje, aplicativo que integrado a una BCI se conoce normalmente dentro del campo de la neurociencia como un *Speller*. Lo interesante de todo este estudio es que básicamente se pregunta “¿Será una BCI útil y fiable en aplicaciones *at-home* para pacientes con ELA en etapa avanzada?”, y la respuesta es bastante compleja: inicialmente fueron 37 los pacientes del estudio para uso de la BCI desde casa, pero debido a diversos factores, al final del período mencionado sólo 8 pacientes eran los que estaban activos en el uso del aplicativo. Y es importante tener estos factores presentes a la hora del diseño de todos estos prototipos cuyo objetivo al final es ayudar a dichas personas, porque escenarios como episodios de hospitalización, progreso de la enfermedad en general, adaptabilidad al paciente y demás aspectos podrían ayudar

o impedir que las condiciones óptimas se cumplan. A pesar de todo lo anterior, y de la alta agresividad biológica que representa la ELA, el comportamiento en general de la BCI fue bastante satisfactorio, y 7 de los 8 pacientes que estaban al final del estudio se quedaron con el dispositivo para usos futuros, demostrando un buen interés al experimentar los resultados luego de los entrenamientos y demás esfuerzos por parte del personal involucrado.

Siguiendo con la idea anterior, Carelli L, Solca F, [...] Poletti en su investigación [10], realizan una recopilación de diversos proyectos BCI y mencionan el estado del arte de diferentes sistemas que utilizan BCI, desde algunos muy simples en cuanto a las interfaces y algoritmos, hasta los más avanzados que incluyen la combinación de diversas técnicas de neurociencia. En primera instancia, describen algunos casos de patologías neurológicas en donde se desarrollan aplicaciones simples de control de motores, pero a medida que se avanza en el artículo, presentan los casos donde ya se requiere un entrenamiento cognitivo para la BCI, entrando en el campo de las terapias de neurofeedback para desórdenes del espectro autista o desórdenes de déficit de atención/hiperactividad. Cabe destacar que, dentro de las diversas aplicaciones y métodos utilizados, mencionan el neurofeedback, los potenciales visuales de estado estacionario (SSVEP por sus siglas en inglés) y el P300, los cuales se profundizarán más adelante en este trabajo (enfocándose en el P300) y que forman parte del vasto mundo de la neurociencia. Así, principalmente dividen los trabajos e investigaciones recopiladas en dos grupos de acción: aplicaciones BCI para pruebas cognitivas y aplicaciones BCI para rehabilitación cognitiva, dependiendo del estado del paciente y la finalidad del sistema en desarrollo. Se discuten con ello, las ventajas y posibilidades de las BCI para mejorar las funciones cognitivas de los pacientes con discapacidades motrices severas, buscando también el mejoramiento de su estado mental y su motivación, factores clave para la neuroplasticidad. Con ello, muestran en los párrafos finales todos esos retos y potenciales campos de aplicación que tienen las BCI como parte del desarrollo actual y futuro.

Además de los estudios ya mencionados, en los trabajos de Jagadish B. y Rajalakshmi [11] se menciona directamente la integración entre una red IoT para la utilización de una BCI, buscando la interpretación de comandos simples (como el mencionado control de un ventilador en una habitación) a través de la detección del parpadeo voluntario del paciente. El sistema utiliza dos (2) electrodos para la obtención de señales, ubicados en Fp1 y Fp2 (región frontal del cráneo), y busca ser una alternativa de bajo consumo energético y baja complejidad de procesamiento.

Finalmente, se menciona el trabajo de Laport F, Dapena A, [...] Iglesia D [12], en el cual se desarrolla un sistema open-source tanto en hardware como en software, buscando detectar los ojos abiertos-cerrados e integrar el sistema usando un protocolo para Internet de las cosas (IoT por sus siglas en inglés). El proyecto tenía como objetivo ser de bajo costo, y para eso utilizaron para la adquisición de señales un amplificador de instrumentación de referencia AD8221 junto con los electrodos correspondientes, seguido de una red de filtrado que diseñan según las frecuencias deseadas, y una tarjeta ESP32 para interpretar los datos y tener comunicación inalámbrica (tiene incluido un módulo WiFi), la cual permite el desarrollo de distintas aplicaciones IoT. La arquitectura del sistema es sencilla, porque sólo requiere la detección de las ondas EEG alfa (como se explicará más adelante), y para la parte IoT utiliza el protocolo *Message Queue Telemetry Transport* (MQTT). Cabe destacar de que a pesar de que el proceso cognitivo neurológico asociado no es tan complejo, el trabajo abarca bastante bien los algoritmos utilizados y explican al detalle los procesos de análisis de señales y clasificación de datos.

1.1.2 Nivel Nacional

Como se explicará a lo largo de este documento a través de los distintos retos que se tuvieron en el desarrollo del proyecto, los sistemas de neurociencia demandan un alto nivel de complejidad no sólo a nivel conceptual debido a todos los delicados aspectos que rodean al sistema nervioso del cuerpo humano, sino también por los dispositivos de adquisición de señales disponibles en el mercado, los cuales en

promedio tienen un alto costo y/o son de restringida adquisición, razón por la cual es posible que, a la fecha de elaboración de este proyecto, no se evidencien proyectos de investigación en neurociencia a nivel nacional enfocados, en éste caso de interés particular, en la integración de BCI utilizando P300 con dispositivos para controlar elementos del entorno del paciente. A pesar de lo anterior, la investigación de electroencefalografía, y especialmente utilizando P300, sí ha tenido un crecimiento a lo largo de los años en el territorio nacional.

Una de las primeras referencias respecto al campo de aplicación mencionado es la de la Facultad de Ingeniería de la Universidad de Antioquia[13], un estudio piloto en el que se presentan dos sujetos (ambos adolescentes, de 15 años) como los participantes de la prueba: el primero tiene cuadriplejía, siendo incapaz de moverse por sí mismo, pero no presenta afectaciones en su aparato fonador, de forma que su comunicación de tipo verbal no se ve afectada; el segundo, posee una enfermedad neurodegenerativa que le causa dificultades para el movimiento y la comunicación tanto verbal como escrita. A lo largo del estudio, los pacientes fueron entrenados para aprender a utilizar una BCI que es controlada mediante los potenciales P300, la cual está orientada a la comunicación alternativa. La interfaz que utilizaban los pacientes se formaba por una matriz 4x3 que contenía imágenes de jugadores de fútbol junto con letras del alfabeto español, de forma que el sujeto de prueba podía seleccionar la imagen de su preferencia y la misma representaba una letra que iba formando la palabra deseada. Hay varios aspectos a resaltar dentro de este estudio, ya que incluye imágenes y no sólo letras como pasa en muchos sistemas de “Spelling” tradicionales, sino que además, utilizan distintos tipos de algoritmos base para el entrenamiento y deletreo, entre ellos el *Linear Discriminant Analysis* (LDA), el cual también fue el seleccionado como principal para el presente proyecto. Es decir, el estudio presenta diversos aspectos que sirven como referencia para el presente proyecto, entre ellos la particularidad que mencionan de que la utilización de imágenes facilita el proceso de entrenamiento a la hora de utilizar la BCI.

Otra de las investigaciones encontradas es una más reciente por parte de la Universidad Antonio Nariño [14], en cual se busca principalmente comparar diversos aspectos para diferentes algoritmos de entrenamiento y clasificación de señales P300, buscando mostrar diferencias en la velocidad de procesamiento, precisión, complejidad y otros rasgos. Utiliza bases de datos disponibles para poder comparar los distintos algoritmos, todos basados en un mismo sistema BCI que está orientado al deletreo o *Spelling* como forma de comunicación para el paciente.

Siguiendo con la idea anterior, se menciona el proyecto presentado por la Universidad de Ibagué desde la Facultad de Ingeniería [15], debido a que se busca desarrollar un sistema de adquisición de señales utilizando dispositivos electrónicos muy similares a los que se utilizaron en el presente proyecto, buscando determinar el nivel de concentración de las personas a través de lo que denominan un *Analog Front-End* (AFE). Analizan diversos datos en el software de MATLAB, e implementan varios métodos de clasificación para validar los resultados, orientados principalmente a las afecciones neurocognitivas relacionadas con la atención, como el trastorno de déficit de atención/hiperactividad (TDAH).

Se encontró el caso de un estudio realizado a pacientes diagnosticados con Epilepsia utilizando potenciales P300 [16], investigación en conjunto entre la Universidad de los Andes (Bogotá), el Hospital de Kennedy (Bogotá) y el Laboratorio de Imágenes Biomédicas de París (Francia). En el estudio, trece pacientes con epilepsia fueron evaluados mediante estímulos auditivos y visuales para generar potenciales P300, con la particularidad de que se utilizaron solamente dos (2) electrodos para la medición, ubicados en Cz y Pz. La ubicación y utilización de los electrodos se explicará en detalle más adelante en este proyecto, pero es importante destacar que los potenciales P300 se presentan principalmente en la región centroparietal de la corteza cerebral, por lo que se observa que incluso con sólo dos electrodos de medición es posible obtener resultados favorables, utilizados por los autores en sesiones de entrenamiento y terapia para los pacientes. Dentro de los distintos resultados, buscaron observar posibles diferencias entre el “grupo de

control” y los pacientes que utilizaban medicamentos antiepilépticos (AED por sus siglas en inglés), además de otros aspectos en las señales P300 detectadas que variaban, por ejemplo, según el tipo de estímulo (auditivo o visual) en la latencia y amplitud.

Como se ha observado a lo largo de los estudios, existe en general una gran presencia en el uso de P300 como los potenciales evocados responsables del funcionamiento base de las BCI, a pesar de que claramente no son el único método posible, pero si uno de los más estudiados en la mayoría de las investigaciones de electroencefalografía. Otro tipo de señales, como los mencionados SSVEP o el neurofeedback, forman parte también del abanico de opciones que permiten analizar los diversos aspectos de la neurociencia. De hecho, en los trabajos de la Universidad El Bosque [17], [18] se muestran las posibilidades que se pueden lograr al implementar diversas técnicas de análisis y procesamiento de las señales electroencefalográficas, a partir del mismo dispositivo de adquisición.

1.2 Descripción y Formulación del Problema

Los pacientes de trastornos o patologías que afectan el sistema nervioso son propensos a padecer complejos cuadros clínicos que afectan el desarrollo y tratamiento de la enfermedad. Como el sistema nervioso es responsable de controlar la mayoría de las funciones básicas del cuerpo, el deterioro de dichas funciones lleva a estos pacientes a necesitar asistencia para realizar prácticamente la totalidad de actividades que impliquen algún movimiento del cuerpo [19]. Por ello, llegan a necesitar atención la mayor parte del día.

Teniendo en cuenta el escenario anterior, es importante añadir que, en la mayoría de los casos, el deterioro muscular llega a tal punto de que se presenten dificultades severas en la comunicación verbal. Es decir, al realizar el prontuario inicial de efectos, las personas con padecimientos neuronales no tienen una manera de poder interactuar con el ambiente donde se encuentran, ni mucho menos controlar elementos de su entorno, estando limitados a una

dependencia total de las personas externas. La imposibilidad de realizar tareas tan sencillas que van desde expresar sus propios pensamientos a sus seres queridos hasta cambiar de canal el televisor de la habitación donde estén se hace presente, afectando con ello no sólo su salud mental, sino drásticamente su calidad la vida [2]. Las condiciones comprometen aspectos físicos, emocionales y sociales del paciente (además de a los cuidadores asociados) [19], ligados directamente al tratamiento de la enfermedad.

Adicionalmente a los planteamientos anteriores, la prevalencia de enfermedades asociadas a discapacidades motoras severas en las comunidades latinas se ha encontrado que es mayor que en otros países [20], condición que se debe a diversos factores como el sistema de salud tratante, aspectos genéticos y muchos otros asociados. De hecho, como pasa con los pacientes de la Distrofia Muscular de Duchenne [21], la fatiga muscular aumenta progresivamente y se hace necesario realizar procedimientos quirúrgicos normalmente asociados a fallas respiratorias (en la zona del tracto vocal, como las traqueotomías), los cuales dificultan todavía más la comunicación del paciente y que, en el ambiente latinoamericano, generan altos costos e importantes barreras administrativas debido al sistema de salud, degradando la calidad y esperanza de vida de los pacientes.

Por otro lado, si bien es cierto que la Esclerosis Múltiple es la enfermedad huérfana con más casos en Colombia [20], el deterioro progresivo de los músculos y las células nerviosas de los pacientes es una condición común entre varias patologías, por lo que la población objetivo no se limita a los pacientes con Esclerosis Lateral Amiotrófica (ELA), sino que potencialmente varios tipos de padecimientos neuronales comparten condiciones similares. En esencia entonces, y como seres humanos que somos, nuestra interacción con otras personas y con el entorno en general es fundamental para el desarrollo de la vida misma, razón por la que, desde el punto de vista de los pacientes con discapacidades motoras severas debidas a patologías neurodegenerativas, la drástica reducción de autonomía, comunicación e interacción con el entorno (incluyendo las tareas simples del día a día) representa el problema principal y el foco de atención para el cual fue pensado y desarrollado el presente proyecto.

1.3 Justificación

Las causas asociadas a la enfermedad per se del paciente son aspectos complejos desde el punto de vista clínico, pero intervenir en las actividades que podría realizar la persona para aumentar la interacción con su entorno y con ello la calidad de vida, además de poder monitorear en tiempo real el estado anímico y las señales EEG, son aspectos que, desde la Ingeniería Electrónica y diversas herramientas disponibles, es posible manejar para lograr reducir en la mayor medida posible esas barreras existentes entre el paciente y las tareas simples del día a día. Con ello, se mejoraría en gran medida no solamente la salud mental y emocional del paciente, sino que existiría una disminución drástica en los trastornos y padecimientos asociados al deterioro de las funciones comunicativas y motoras.

Según las evidencias de los estudios [2][22][23][24], existen altos índices de depresión en los pacientes, además de un grave deterioro en la salud mental asociada, causando con ello trastornos de sueño, problemas cognitivos y padecimiento de enfermedades adicionales como consecuencia de los malos hábitos y la calidad de vida en decadencia. Además, según los mismos pacientes, hay un deseo en común de querer mantener la identidad, que se refiere a mantener en lo posible una normalidad y roles ya establecidos, pero la fuerte sensación de pérdida de independencia los afecta considerablemente. Es decir, queda en evidencia que, si bien es cierto que los efectos fisiológicos de estas patologías sobre el cuerpo del paciente son devastadores, las consecuencias en la salud mental, emocional y la calidad de vida asociada son igualmente o incluso más graves que aquellas que se ven a simple vista, dejando secuelas y padecimientos que muchas veces empeoran el cuadro clínico general.

Para ayudar entonces a mejorar las condiciones mencionadas anteriormente, es importante mencionar que la mayoría de los pacientes se encuentra en uno de los dos escenarios posibles: una habitación en un entorno clínico, o en la residencia propia (o de algún familiar/allegado). En dichos sitios, y debido a que se requiere de la presencia de algún personal de atención la mayor parte del tiempo, se colocan diversos dispositivos para monitoreo y asistencia del paciente, como lo pueden ser equipos que estén midiendo la

oxigenación, respiración, señales cardíacas (ECG), ventiladores para climatización, etc. Sin embargo, también están todos los otros dispositivos que permiten que el paciente pueda sentirse lo más cómodo posible, como lo son televisores, interruptores para la iluminación, alarmas para llamar a enfermeras/asistentes, controles para la camilla y demás, los cuales representan el foco de atención que permitiría recuperar cierto nivel de independencia: mediante una interfaz BCI basada en el reconocimiento de potenciales P300, es posible que el paciente pueda controlar en cierta medida varios de los dispositivos que están en su entorno, llegando a formar una red de comunicación que interpreta los comandos simples del paciente y posteriormente los ejecuta en el dispositivo objetivo. Es decir, cambiar de canal el TV y sus diferentes opciones (volumen, uso de apps como Netflix, YouTube y similares), poder activar un reproductor inteligente de música (como el Amazon Alexa o un Google Home), enviar mensajes al doctor, enfermera más cercana o algún familiar y varias otras opciones, son parte de lo que el sistema basado en una BCI puede lograr, al interpretar los comandos y ejecutarlos dentro de una red que integre en lo posible *Smart Devices* cuyo funcionamiento sea inalámbrico (bajo una conexión a Internet), formando una arquitectura que en conjunto funciona como un sistema IoT controlado por los comandos del paciente.

Adicionalmente a los planteamientos anteriores, existen varias ventajas en el uso de estos sistemas frente a otras “tecnologías”. Está el caso, por ejemplo, de los pacientes con ELA, enfermedad neurodegenerativa que causa parálisis progresiva y cuyos pacientes han sido por experiencia uno de los principales en participar en estudios para sistemas BCI [21], pero en general, la fatiga muscular y eventual parálisis es un factor que afecta distintos pacientes de enfermedades neuromusculares, como la Distrofia Muscular de Duchenne (DMD), llegando a necesitar asistencia médica completa por los altos riesgos a los que se exponen. En dichos casos, al evaluar a los pacientes bajo diferentes cuadros clínicos, se ha encontrado que usualmente no presentan un deterioro en sus funciones cognitivas, lo cual ha hecho que sean objetivo de herramientas de comunicación alternativa, como los conocidos dispositivos de “comunicación alternativa y aumentativa” (AAC por sus siglas en inglés). Sin embargo, la progresiva debilidad muscular y fatiga asociada hace que no puedan utilizar esos dispositivos continuamente, resaltando la necesidad de que los sistemas de comunicación no dependan de la actividad muscular, y es por esto que la arquitectura propuesta basada en una BCI

controlada por potenciales P300 es una alternativa preferible, al requerir solamente de los procesos cognitivos del paciente como respuesta a ciertos estímulos.

Además de lo mencionado existen ventajas adicionales al utilizar la solución planteada, como la posibilidad de hacer análisis de datos para monitoreo EEG, lo cual permitiría realizar sesiones de telemedicina o incluso terapias dirigidas remotamente (opcionales por parte del personal involucrado); también, el escalamiento en complejidad de los sistemas BCI se hace mucho más sencillo al compararlo con cualquier otro tipo de sistemas no-invasivos (para análisis de señales bioeléctricas o biopotenciales), es decir, si durante los estudios o después de cierto tiempo es necesario modificar algunas características o se deciden agregar ciertas funciones al sistema, se puede hacer sin comprometer el desempeño inicial.

1.4 Objetivos

1.4.1 Objetivo General

Desarrollar un prototipo de un sistema BCI que identifique comandos deseados a través del paradigma ERP (*Event-Related Potential*), de modo que se puedan interpretar por un sistema IoT asociado a la interacción domótica con el entorno del usuario.

1.4.2 Objetivos Específicos

1. Diseñar un dispositivo electrónico de adquisición capaz de obtener señales EEG.
2. Diseñar e implementar un algoritmo basado en el paradigma ERP bajo una interfaz BCI, que sea capaz de identificar y clasificar comandos orientados a la interacción con el entorno.
3. Transmitir y ejecutar los comandos identificados a una red IoT que se conecte con dispositivos inteligentes en un entorno de prueba.
4. Elaborar los manuales del prototipo del equipo, que evidencien pruebas de un funcionamiento correcto

1.5 Requerimientos

1.5.1 Requerimientos del usuario

- El sistema deberá identificar alguna acción escogida por el usuario y transmitirla hacia el correspondiente dispositivo de prueba para su ejecución.
- Interfaz de interacción amigable.
- Integración en la red IoT con los dispositivos más próximos al usuario.
- Ejecución de las acciones deseadas en el menor tiempo posible.
- Interpretación que sea acorde a las intenciones del usuario.
- Interacción con el entorno que tenga el menor tiempo de respuesta posible.

1.5.2 Requerimientos funcionales

- El sistema de adquisición de señales del prototipo estará diseñado para trabajar con frecuencias de máximo 200 Hz.
- El prototipo entregará señales filtradas según los canales establecidos en el diseño.
- El sistema recibirá señales de electroencefalografía que serán procesadas según algoritmos de decisión y discretización de comandos.
- El software del sistema permitirá enviar las señales de las acciones identificadas a una red IoT.
- El sistema interpretará las acciones deseadas y las ejecutará en el entorno del usuario.

1.5.3 Requerimientos de calidad

- Adquisición de las señales EEG asociadas a potenciales de eventos relacionados (ERP-P300) en el mínimo tiempo posible para realizar una interacción con el entorno.

- El prototipo filtrará e identificará las acciones del usuario en un tiempo no mayor a 3(seg).
- El sistema deberá enviar la acción identificada a la red IoT, en un tiempo no mayor a 3 (seg).
- El prototipo deberá identificar las acciones con un error que no exceda el 20%.
- El sistema no debe producir ningún tipo de dolor, picazón, ardor ni molestia al usuario.
- El prototipo no debe contener ningún dispositivo invasivo para el usuario.
- El sistema debe contar con protecciones mínimas para una utilización adecuada y sin riesgos.

1.5.4 Requerimientos restrictivos

- El prototipo será activado por los procesos de pensamiento del sujeto de estudio.
- Las neuronas motoras del sujeto no intervendrán en los procesos del prototipo.
- Los procesos de pensamiento deben realizarse en el sujeto de estudio de manera correcta (ser humano adulto sano).
- Las condiciones adversas que interfieran en el proceso de pensamiento original deben ser mínimas, asegurando la integridad de las señales EEG.
- Las interacciones con el entorno serán con máximo tres (3) dispositivos.

1.6 Alcance y Limitaciones del Proyecto

Dentro de todas las posibilidades y herramientas disponibles en el campo de la neurociencia, el objetivo inicial del proyecto se mantuvo durante todo el desarrollo y las distintas etapas, pero diversos aspectos son importantes mencionar en el alcance de este ya que forman parte del aprendizaje y los resultados que poco a poco se fueron obteniendo:

- Inicialmente estaba planteada la transmisión y ejecución de los comandos identificados sobre diversos dispositivos asociados a una red IoT, pero la gran variedad de marcas, sistemas operativos disponibles, protocolos de comunicación y accesibilidad hicieron que se limitaran a sólo algunos sistemas disponibles, para mayor facilidad. Entre esos, existen por ejemplo muchos tipos de Sistemas Operativos (SO) para los *Smart TV*, por lo que el proyecto sólo se enfocará en *webOS*, el sistema preferido en la mayoría de los TV del fabricante LG, debido a que era al que se tenía acceso.
- El manejo de información de forma “online” (bajo una conexión a Internet) a la hora de interactuar con los dispositivos sólo se toma en cuenta para ciertos comandos. Es decir, algunos comandos particulares que se programaron en la BCI sí necesitaban conectarse a un servidor en la nube para poder ejecutarse (el caso del envío de mensajes por la aplicación de WhatsApp), lo cual implica claramente el requisito de una conexión a Internet. Ahora bien, el funcionamiento del sistema en su totalidad implica toma de datos EEG, procesamiento, identificación de comandos y posterior ejecución, con lo cual se quiere dejar claro que la conexión a Internet depende del comando identificado y el dispositivo asociado, pero no representó una restricción para el funcionamiento global del sistema. De hecho, varias pruebas y conexiones realizadas fueron posibles mediante una red local, de manera que las posibilidades para la comunicación con dispositivos del entorno no dependieran exclusivamente del envío de datos por medio de una conexión a Internet.
- Una de las mayores limitaciones se debe al hardware, el cual constaba principalmente de las tarjetas de adquisición y preprocesamiento (el ADS1299EEGFE-PDK y el ATmega328P respectivamente). En conjunto, representaron una barrera en cuanto a las opciones disponibles para poder muestrear las señales cerebrales, sobre todo por el hecho de que el manejo del *Analog-to-Digital Converter* (ADC) estaba restringido al uso de librerías

específicas que limitaban el proceso de adquisición de datos. Adicionalmente, la necesidad de utilización de la tarjeta MMB0, cuya función principal es servir como interface de alimentación y conexión al computador, restringe principalmente el modo de reconocimiento del ADC por parte del computador cuando se utiliza el software brindado por TI (Texas Instruments, el fabricante).

2. MARCO DE REFERENCIA

2.1 Marco Teórico

Antes de poder abordar y comprender las distintas herramientas disponibles que permitieron el desarrollo del proyecto, es necesario sentar las bases conceptuales relacionadas no sólo con los dispositivos electrónicos requeridos para adquirir los datos, sino con aquellos fenómenos biológicos que dan lugar a las señales generadas en los procesos cognitivos del cerebro, los cuales dieron lugar a las respuestas eléctricas detectables y de interés para este proyecto.

En primer lugar, es necesario abordar el funcionamiento básico y partes principales del sistema responsable de la mayoría de las acciones y respuestas tratadas en el campo de la neurociencia: el sistema nervioso.

2.2 Estructura y Funciones Generales del Sistema Nervioso

Las tareas que puede realizar el cuerpo humano son numerosas y variadas, pero para un correcto funcionamiento debe existir algún “organismo central” que sea responsable de organizar y dirigir a los demás subsistemas, estableciendo pautas para regular los distintos procesos de nuestro organismo. En este sentido, el responsable y principal actor de controlar, establecer rutas de comunicación y dirigir todos esos procesos es el sistema nervioso, que se puede definir de manera global como una compleja red de estructuras especializadas (el encéfalo, la médula espinal y los nervios) que se encargan de regular y mantener bajo supervisión constante los distintos órganos y sistemas del cuerpo, coordinando los componentes internos e incluso interviniendo en la relación del organismo con el mundo exterior. Es un sistema multifuncional, siendo capaz de detectar variaciones tanto en el medio interno como el externo, evaluar la información recibida y responder ante dichos datos mediante cambios en los músculos o glándulas [25].

La estructura del sistema nervioso se divide principalmente en dos grandes subsistemas: el sistema nervioso central (SNC), integrado la médula espinal y el encéfalo, y el sistema nervioso periférico (SNP), el cual comprende la vasta red de neuronas y fibras nerviosas extendidas desde el SNC hacia todo el organismo [25], [26]. Asimismo, las vías de comunicación periférica se dividen en nervios espinales (provenientes de la médula espinal) y nervios craneales (que vienen del cráneo).

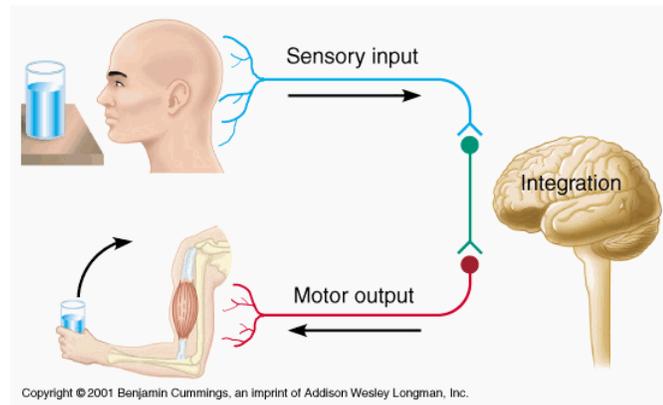


Figura 1. Diagrama ilustrativo de las tres funciones básicas del Sistema Nervioso: a) adquirir información sensorial; b) analizar la información obtenida; c) determinar una respuesta motora. Extraído de [26].

Como se observa según el diagrama anterior, el funcionamiento global del sistema nervioso se divide en “módulos” que comparten la información y realizan acciones diferentes en base a ella. En conjunto, su operación muchas veces se compara con la de un computador:

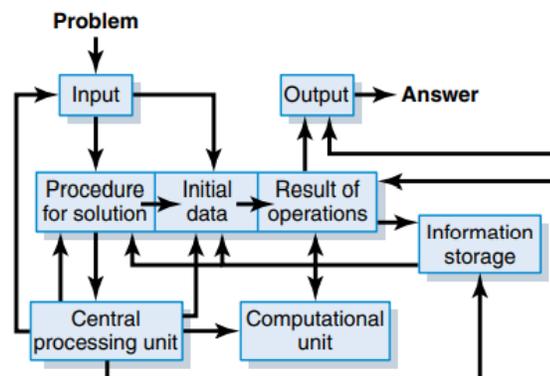


Figura 2. Diagrama de bloques de un computador genérico, con sus componentes básicos. La analogía con el sistema nervioso busca resaltar las funciones básicas de adquisición de datos, procesamiento de información y toma de decisiones. Imagen extraída de [27]

2.3 Sistema Nervioso Central (SNC)

En el SNC se concentran las funciones de monitoreo y toma de decisiones para la ejecución de actividades conscientes del organismo. Se compone del encéfalo en su totalidad y la médula espinal [26]. A grandes rasgos, interviene en prácticamente todas las funciones vitales del cuerpo humano, desde aquellas que garantizan nuestra supervivencia hasta las tareas que requieren complejos procesamientos.

El encéfalo representa toda la región cubierta por el cráneo, formando por cuatro partes principales: el tronco del encéfalo, el cerebelo, el diencefalo y el cerebro.

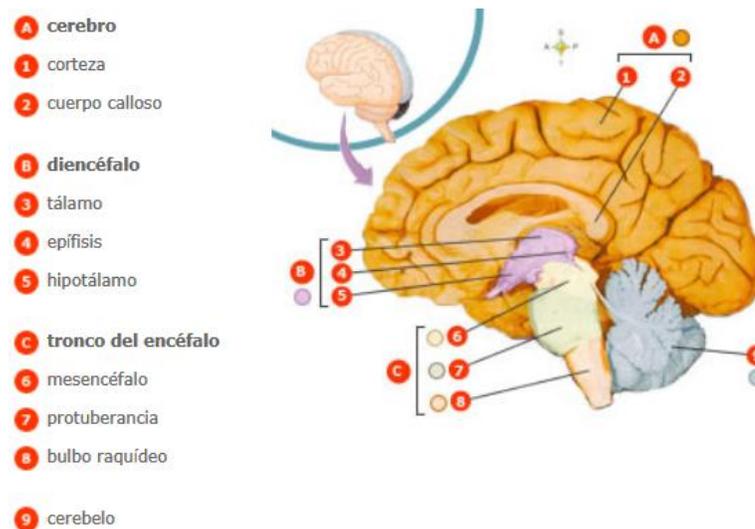


Figura 3. Representación ilustrativa de las principales partes del encéfalo. Imagen extraída de [25]

Dentro de la estructura del tronco encefálico se ubica la formación reticular, la cual es responsable de regular acciones como el despertar (estado de vigilia), la atención y el sueño. Además, se comunica preferentemente con el *hipotálamo* y la *hipófisis* para controlar los ciclos biológicos de sueño y vigilia (conocidos como *ritmos circadianos*) e incluso la temperatura corporal.

Por arriba del tronco encefálico se ubica el *tálamo*, que a modo de analogía funcional, se puede decir que opera como una central de recepción y distribución de mensajes (impulsos nerviosos), debido a que recibe y envía información neuronal de la corteza cerebral a las

partes inferiores del cuerpo, por medio del SNP y viceversa. Dicha analogía funcional aplica similarmente a una central ferroviaria, porque organiza e indica las vías y direcciones para cada ferrocarril, que vienen siendo los impulsos nerviosos, llegando a comunicar a los órganos de los sentidos con la corteza cerebral [26]. Adicionalmente, detrás del tronco encefálico se ubica el *cerebelo*, el cual es el principal actor que regula el equilibrio corporal, además de que controla los movimientos voluntarios que vienen como órdenes en forma de señales desde el cerebro (también automatiza dichos movimientos).

Las diferentes partes del encéfalo desempeñan funciones para la regulación de los movimientos, adquisición de conocimientos, emociones, memoria, control de acciones y demás actividades del organismo. Especialmente, y para efectos de interés del presente proyecto, es el *cerebro* el punto central dentro de este sistema, siendo el responsable de generar las señales que representan la información clave para ser analizada por dispositivos externos, originadas como respuesta a ciertos estímulos.

2.3.1.1 El Cerebro

De manera similar a la *Unidad Central de Procesamiento* (CPU) en un computador, el cerebro es el núcleo fundamental en la recopilación de datos, análisis de información y ejecución de acciones para prácticamente todos los procesos del cuerpo, desde los básicos hasta los más complejos.

El cerebro consta estructuralmente de la *corteza cerebral* (que es una capa superficial de sustancia gris), la sustancia blanca y los núcleos estriados. En la parte baja se ubica el tronco encefálico, formado por el bulbo o médula oblongada, el puente y el mesencéfalo, que en conjunto funcionan como conexión a la médula espinal. Asimismo, la médula espinal conjuntamente con el bulbo y el puente controlan varias funciones primordiales para la supervivencia, como la respiración, la circulación sanguínea y el latido cardíaco, entre varias otras [26].

Mediante los diversos procesos regulados por el cerebro, los seres humanos pueden escribir, leer, hablar, calcular mentalmente operaciones numéricas, componer música, recordar eventos pasados, planear el futuro e incluso imaginar abstracciones e ideas para crear artefactos que jamás han existido: es la base de lo que conocemos como “inteligencia” o, más estrictamente, *funciones intelectuales*. A lo largo de diversos estudios [27], se ha encontrado que dichas funciones se originan principalmente en la *corteza cerebral*, la parte más alta del cerebro, que es la superficie llena de pliegues llamados *circunvoluciones*.

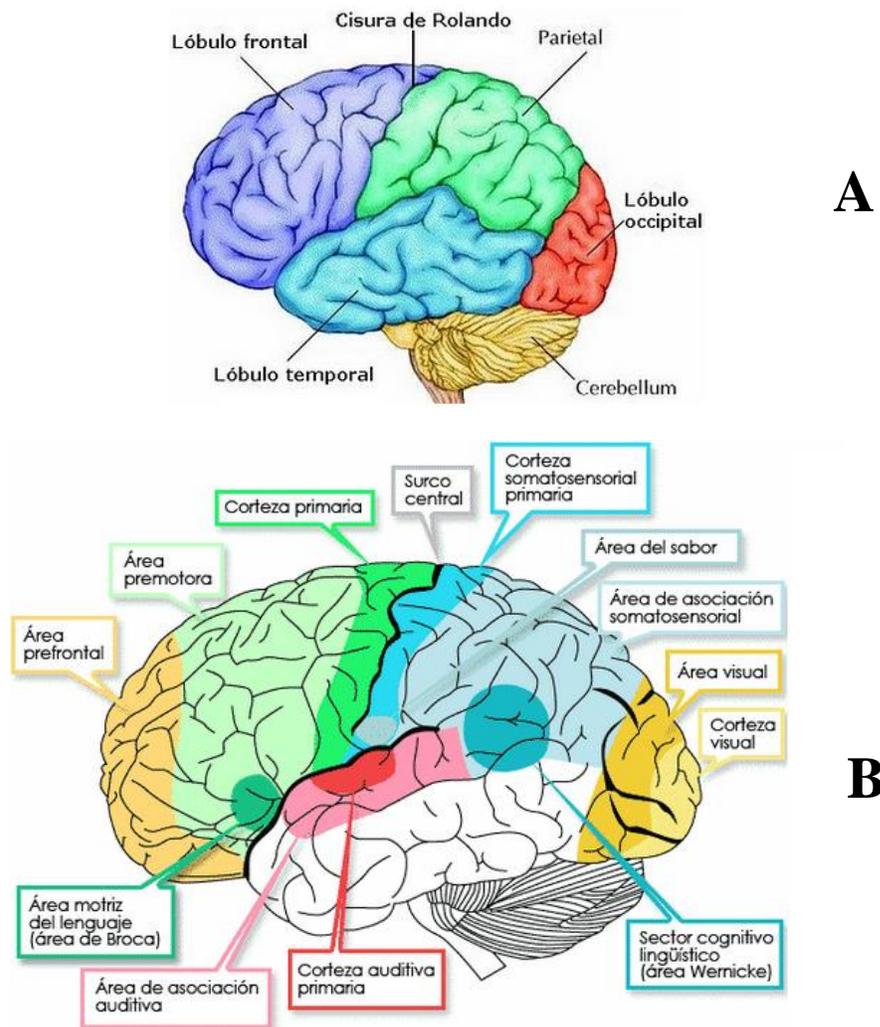


Figura 4. Principales áreas funcionales de la corteza cerebral, distinguiendo las regiones asociadas a cada función conocida. Imagen extraída de [25]

El pliegue más profundo y prominente de la corteza cerebral es el que precisamente divide el cerebro en los dos *hemisferios cerebrales*, derecho e izquierdo. Cada uno de ellos a su vez se subdivide en cuatro *lóbulos*, los cuales se llaman conforme a los huesos que los envuelven: *frontal*, *temporal*, *parietal* y *occipital*, distinguidos como se observa en la Figura 4-A. Estos lóbulos, como se verá un poco más adelante, definen en gran medida la ubicación de los electrodos de medición para las señales cerebrales de interés del proyecto, siendo la base de la localización para los protocolos internacionales aceptados a la hora de ubicar electrodos en las mediciones de actividad cerebral.

Funcionalmente, no se podría delimitar la corteza cerebral sólo teniendo en cuenta la distribución por los lóbulos, debido a que la gran cantidad de acciones posibles produce actividad eléctrica cerebral en sectores diferentes. De hecho, los diversos estudios e investigaciones han mostrado que las diferentes áreas de la corteza cerebral poseen funciones específicas, determinadas por estimulación eléctrica de la corteza durante procedimientos de neurocirugías en diversos pacientes. De esta manera, al recopilar las distintas pruebas y fuentes de información, se determinó que existe una región específica encargada de las acciones motoras (ejecutar movimientos), otra particular para la visión, y de manera similar ocurre para las demás funciones. Así surgen las denominadas *áreas funcionales*.

En general, la corteza cerebral se divide en tres grandes tipos de áreas funcionales:

- *Áreas sensoriales*: se encargan de recibir e interpretar todas las señales que vienen de los órganos de los sentidos (receptores sensoriales periféricos), situándose principalmente en la parte posterior del córtex. Incluye la recepción de señales del tacto, dolor, temperatura, formas y colores (estímulos visuales), sonidos, sabores y olores.
- *Áreas motoras*: controlan las contracciones musculares voluntarias, incluyendo las zonas que requieren un control muscular de alta precisión, como los dedos, lengua y cuerdas vocales (para la articulación correcta de los sonidos).

- *Áreas de asociación*: integran áreas motoras, sensoriales y de distintos lóbulos, porque representan las acciones o tareas de integración más complejas. Las denominadas algunas veces como “funciones intelectuales” representan tareas de alto procesamiento, como determinar la forma y textura de un objeto sin verlo (sólo imaginando), distinguir sonidos provenientes de diferentes personas/instrumentos musicales, interpretar emociones, pronosticar las acciones como al pensar las consecuencias de los movimientos antes de ejecutarlos, resolver problemas matemáticos o filosóficos complejos (cuestionamientos abstractos), controlar nuestro comportamiento según la ética y la moral, entre varios otros ejemplos.

2.4 Sistema Nervioso Periférico (SNP)

El SNP funciona como la conexión entre el encéfalo y todos los órganos internos, siendo el que adquiere y envía señales al cerebro para responder y ordenar acciones ante la información analizada, lo cual ocurre en situaciones como cuando el cuerpo detecta frío y los receptores de la piel registran la temperatura, para que posteriormente el cerebro ordene realizar alguna acción (acercarse a una fuente de calor, buscar prendas de ropa adicionales o incluso realizar ejercicio aeróbico) para contrarrestar las bajas temperaturas detectadas. Adicionalmente, el SNP es el encargado de controlar importantes actividades de regulación autónoma, como sucede durante las actividades de ejercicio muscular intenso, haciendo que se aumente el latido cardíaco para que se estimule y aumente la transmisión de oxígeno a la sangre y las fibras musculares.

Al SNP lo forman todos los nervios que se ramifican del SNC y que se subdividen hacia todas las regiones del cuerpo. Algunos, como los de los ojos y oídos, se conectan directamente con el cerebro y son los llamados *nervios craneales*. Los nervios que se ramifican de la médula espinal al resto del cuerpo se conocen como *nervios espinales* [26]. Asimismo, la gestión de la información se hace mediante las llamadas vías *aférentes* o *eferentes*: la entrada de información (normalmente sensitiva) se conoce como una aferencia,

y la respuesta ante dicha información es una eferencia (va desde un sistema central hacia las ramificaciones periféricas).

Para efectos del presente proyecto, el subsistema de especial interés es el SNC, y más específicamente el cerebro, siendo el principal responsable de la actividad eléctrica asociada a estímulos sensoriales, lo cual da origen a numerosos estudios en el campo de la neurociencia. Ahora bien, el sistema nervioso opera esencialmente gracias a las *neuronas*, quienes son los principales agentes transmisores y ejecutores de las acciones que el cuerpo realiza, pero anatómicamente, el tejido nervioso consta de dos tipos de células: las *neuronas* y la *glía o neuroglia*, siendo ambas fundamentales para que todo el sistema funcione correctamente.

2.5 Neuronas

Representan la base del funcionamiento del sistema nervioso, cumpliendo la función principal de generar señales eléctricas (y en algunos casos, señales humorales), responsables de transmitir información pertinente a prácticamente todas las zonas del cuerpo. Poseen unos terminales denominados *terminales axónicos o sinápticos*, y es a través de ellos como la neurona contacta y envía información a otra neurona o célula efectora [27], por lo que son la base fundamental de la transmisión de información.

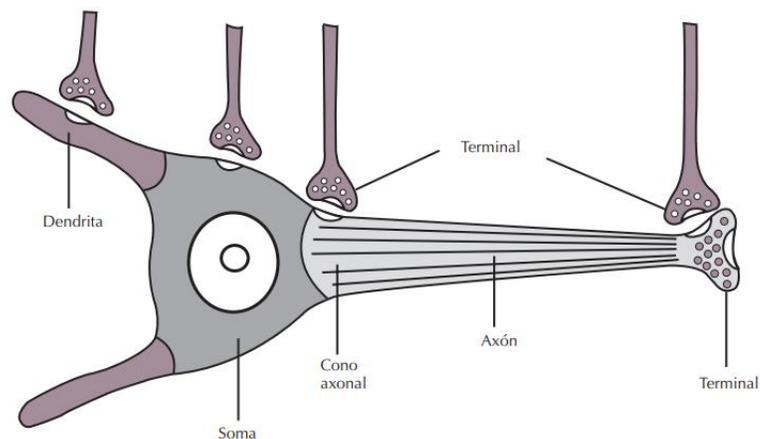


Figura 5. Gráfico típico de una neurona, con la sinapsis que recibe. Imagen extraída de [28]

Las neuronas pueden tener variadas formas y tamaños, pero como lo indica el gráfico anterior, su estructura básica se compone de tres elementos principales: dendritas, axones y el cuerpo neuronal [25].

- *Cuerpo o soma neuronal*: es donde se ubica el núcleo celular junto con el respectivo citoplasma. Además, rodeados por una membrana plasmática, se encuentran los orgánulos intracelulares.
- *Dendritas*: representan prolongaciones cortas que se ramifican, normalmente de manera múltiple, y mediante ellas es que las neuronas son capaces de adquirir las señales que vienen de neuronas vecinas. El contacto con las células neuronales vecinas mediante las prolongaciones resulta en la denominada *sinapsis*.
- *Axón*: se define normalmente como una prolongación que es de longitud variable y única, a través de la cual se transmite la información (impulsos nerviosos) proveniente del núcleo celular hacia las neuronas vecinas u órganos efectores. En el extremo, el axón se divide en muchas ramificaciones separadas con el fin de comunicarse, debido a que la “señal de salida” viaja por el axón [27].

Adicionalmente, la información que circula dentro del sistema nervioso es manejada por tres tipos de neuronas:

- *Neuronas sensoriales*, quienes se encargan de enviar la información proveniente de los órganos sensoriales y tejidos asociados hacia la médula espinal.
- *Interneuronas*, cuyo papel aparece cuando el cerebro procesa la información sensorial, formando el sistema de comunicación interna.
- *Motoneuronas*, que aparecen principalmente luego de que se procesa la información sensorial, de forma que el sistema nervioso central manda

instrucciones hacia los tejidos del cuerpo para ejecutar acciones específicas, normalmente relacionadas con el movimiento.

Los tres tipos de neuronas indican precisamente las funciones principales del sistema nervioso: en primer lugar, la adquisición de información relacionada al interior del cuerpo, lo cual implica saber en qué estado se encuentran los órganos (pulmones, corazón, etc.) o alguna parte específica del organismo. Asimismo, necesitamos saber también cómo está nuestro entorno y cuál es nuestra relación con él, lo cual da pie a la obtención de datos relacionados con la temperatura externa, luminosidad, obstáculos e incluso depredadores o peligros que pueden estar cerca; lo anterior, forma parte de todo un conjunto de receptores que captan esa información y la llevan al cerebro para analizarla y posteriormente tomar una decisión que implica alguna acción, la cual se ve reflejada internamente (en el mismo organismo) o externamente, en la conducta y reacciones asociadas (correr para escapar, adoptar posturas de defensa, relajar o contraer los músculos y demás posibilidades dependiendo de la situación). Teniendo en cuenta estas ideas, el sistema nervioso puede ser analizado a través de tres estados funcionales:

- Mantener la recopilación de información (estado de “alerta”).
- Procesar y analizar la información obtenida.
- Generar respuestas adecuadas ante la información recibida y analizada mediante la activación de zonas o funciones específicas del cuerpo.

Lo anterior resume a grandes rasgos las funciones de dicho sistema, y es también la base del funcionamiento de distintas características del presente proyecto, que como se verá a lo largo de las descripciones, muchas veces resulta ser una analogía del comportamiento del sistema nervioso llevada a dispositivos externos.

2.6 Glía o neuroglia

Dentro del tejido nervioso, las neuronas están sostenidas por un grupo de células no excitables que son las que, en conjunto, se conocen como neuroglia. También se les llama células gliales, y son en general más pequeñas que las neuronas. Algunas de las células de la neuroglia son: oligodendrocitos, astrocitos, células ependimarias, células de Schwann y células satélite. Ahora, si bien es cierto que la glía sirve como soporte para las neuronas, durante mucho tiempo se pensó que no cumplía ninguna función importante, pero hoy en día se sabe que dichas células participan en la formación, operación y modulación de los circuitos sinápticos [28]. De hecho, como todas las células nerviosas que no producen potenciales de acción se agrupan en lo que conoce como glía, las neuronas vienen siendo sólo una fracción de las células del cerebro humano, estimándose que las células gliales son aproximadamente diez veces la cantidad de neuronas (cerca del 90% de las células nerviosas) [29].

En esencia, se puede decir entonces que la glía cumple funciones diversas en la fisiología del cerebro: se ha encontrado que son clave para entender trastornos de bipolaridad y la esquizofrenia, e incluso existe un tipo de glía, llamado microglía, que se vuelve activa en enfermedades neurodegenerativas (las que afectan a los pacientes parte del público objetivo de este proyecto), aunque los efectos específicos en dichos casos son todavía materia de investigación [28], [29].

2.7 Sinapsis y Transmisión Sináptica

A lo largo de todo el sistema nervioso, la transmisión de información es la clave para poder ejecutar las distintas acciones, y dicha información (señales eléctricas conocidas como *potenciales de acción* o *impulsos nerviosos*) necesita por ende pasar por distintas neuronas durante su recorrido, las cuales no están en contacto directo [25]. Ese complejo proceso de comunicación y transmisión de la información entre neuronas es lo que se conoce como *sinapsis*, la cual ocurre en el lugar de contacto entre dos neuronas o entre una neurona y un órgano efector.

Principalmente existen dos tipos de sinapsis: la *sinapsis química* y la *sinapsis eléctrica*, cuya diferenciación se origina sobre todo porque en la química ocurre la liberación de las sustancias químicas (biomoléculas) conocidas como *neurotransmisores*, mientras que en la eléctrica esto no pasa [25]. Cabe resaltar que, casi todas las sinapsis relacionadas a la transmisión de información en el sistema nervioso central de los seres humanos son químicas, lo cual resalta la gran importancia de las distintas sustancias neurotransmisoras (acetilcolina, adrenalina, serotonina y varias otras).

A grandes rasgos, en el proceso de la transmisión sináptica se definen dos células cercanas, la neurona presináptica y la neurona postsináptica, las cuales establecen una comunicación mediante varias etapas: la célula presináptica forma la terminal presináptica al ensanchar su respectivo axón, formando sacos membranosos que contienen neurotransmisores químicos. Ahora bien, a cierta distancia de la terminal presináptica está la correspondiente terminal postsináptica, de modo que están separadas por un espacio llamado *hendidura sináptica*. Así, posteriormente el neurotransmisor se prepara para ser liberado hacia la célula postsináptica [25], donde teniendo en cuenta las terminales mencionadas, recibe la sustancia que actúa en la membrana plasmática y con ello ocasiona la activación de los correspondientes *canales iónicos*. Esto permite en cierta manera controlar el flujo de iones específicos mediante dichos canales que actúan como “puertas semipermeables”, admitiendo o bloqueando el paso de ciertos iones, lo cual puede producir la *hiperpolarización* (se detiene la transmisión del impulso nervioso) o la *despolarización* (se transmite el impulso nervioso).

Una característica especial de la mayoría de las sinapsis (las químicas) es que la señal normalmente viaja “hacia adelante”, desde el axón de la neurona precedente hacia las dendritas en la membrana celular de la neurona subsecuente. Esto obliga a que la información a transmitir viaje en direcciones particulares para poder realizar ciertas funciones nerviosas [27]. Más específicamente, en la sinapsis química la información viaja de forma unidireccional, mientras que en la sinapsis eléctrica puede hacerlo bidireccionalmente.

2.8 Potencial de Acción

Para explicar este concepto clave dentro del campo de la neurociencia, cabe mencionar primeramente que, dentro del proceso de la sinapsis entre neuronas (y en prácticamente todas las células del cuerpo) aparecen potenciales eléctricos en las membranas celulares, asociados a los cambios de concentración de iones y sus cargas eléctricas [27].

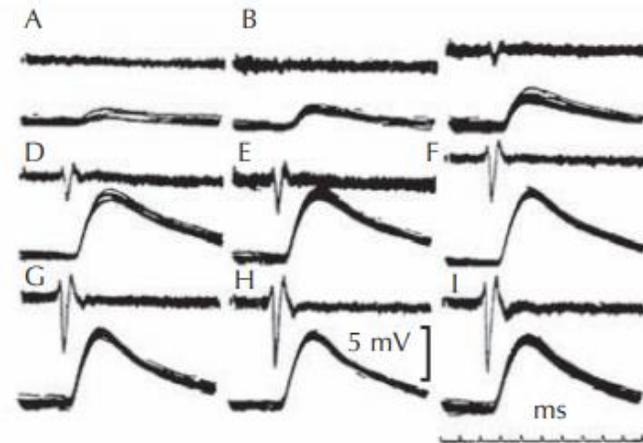


Figura 6. Potenciales postsinápticos excitatorios. Se muestran los registros de potenciales postsinápticos adquiridos con un estímulo de intensidad creciente. Imagen extraída de [30]

En el caso de ciertas células, como las células musculares y nerviosas, no solamente existen diferencias de potencial eléctrico en las membranas, sino que dichas células son capaces de generar impulsos electroquímicos que cambian rápidamente en sus membranas, y son esos impulsos los responsables de transmitir señales a través de las membranas nerviosas y musculares [27]. De hecho, las motoneuronas de la región lumbar del cuerpo son unas de las zonas más estudiadas para poder comprender la función sináptica en el sistema nervioso de los mamíferos, mencionando que, a diferencia de la unión neuromuscular, cientos de terminales sinápticos pueden ser recibidos en cada región de una neurona.

Ahora bien, las membranas celulares son selectivamente permeables, lo que hace que dejen pasar ciertos iones en algunos casos y a la vez se impide el paso de los demás. Esto origina que, por ejemplo, concentraciones de iones de potasio o sodio dentro de la fibra nerviosa causan diferencias de potencial entre el interior y el exterior de la membrana, por

las cargas eléctricas concentradas, lo cual da origen al llamado *potencial de difusión*. Adicionalmente, cuando una fibra nerviosa no transmite señales nerviosas, existe un *potencial de membrana en reposo*, siendo el caso general el denominado *potencial de membrana*, que es la diferencia de potencial entre el interior de la fibra nerviosa y el exterior.

Teniendo en cuenta las ideas anteriores, ya se puede comprender el concepto de interés principal: los impulsos nerviosos se transmiten a través de *potenciales de acción*, los cuales se manifiestan como cambios transitorios y rápidos del potencial de membrana, llegando a propagarse por todo lo largo de la fibra nerviosa [27]. Dicho evento ocurre debido a que el potencial de acción se mueve hasta llegar al final de la fibra, ocasionando que en la fase inicial las cargas positivas se desplacen hacia el interior de la fibra y en la fase final las cargas positivas vuelven al exterior. Así, a través del potencial de acción se generan los cambios asociados que conllevan a la transmisión de información/acciones hacia partes específicas del organismo.

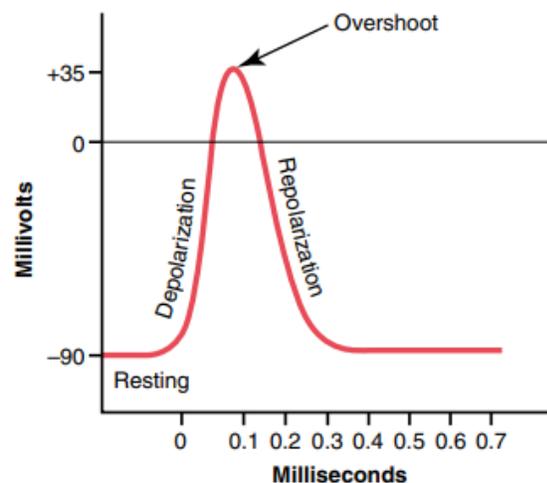


Figura 7. Potencial de acción típico que puede ser observado a través de distintos mecanismos de medición electroencefalográfica. Se muestran además las fases sucesivas del potencial de acción: el reposo (resting), la despolarización y la repolarización. Imagen extraída de [27].

2.9 Ondas Cerebrales y el EEG

Las mediciones eléctricas realizadas para captar señales fisiológicas del cuerpo se basan en el funcionamiento de dispositivos electrónicos que, a través de acoplamientos correctos de impedancias y sensores especiales con amplificadores de bajo ruido, son capaces de captar dichas señales del cuerpo. En el caso particular del cerebro, se ha demostrado que existe una continua actividad eléctrica que se refleja en la aparición de ondas que varían en intensidad, frecuencia y patrones dependiendo de distintos factores. Las señales resultantes de esos registros se conocen como *Ondas Cerebrales*, y el registro completo es el denominado EEG (*electroencephalogram*).

La actividad eléctrica cerebral varía en intensidad y patrones determinados por varios factores relacionados al nivel de excitación de diferentes regiones del cerebro, lo cual hace que las señales tengan componentes sensibles a características como el nivel de sueño, la vigilia o enfermedades cerebrales como la epilepsia o incluso la psicosis [27]. De hecho, en este punto ya se hace evidente un factor clave para la neurociencia: las diferentes enfermedades cerebrales ocasionan trastornos en el funcionamiento normal del sistema nervioso, lo cual ha hecho que con los años una de las herramientas más útiles para el diagnóstico y tratamiento de diversas patologías sea precisamente el EEG, puesto que el registro de la actividad eléctrica cerebral permite analizar diversos componentes asociados a los efectos en los pacientes.

En esencia, las características del EEG y de cada onda cerebral particular se asocian al grado de actividad en cierta región de la corteza cerebral. Muchas veces, las señales de los registros son muy irregulares y no existe algún patrón especial o característica que se pueda extraer del EEG. En otras ocasiones, pueden aparecer patrones que son característicos y específicos para ciertos trastornos, como es el caso de la epilepsia [27], o bajo ciertos estímulos, se puede lograr observar señales particulares que reflejan acciones específicas por parte del paciente.

La intensidad de las señales cerebrales registradas desde la superficie del cuero cabelludo de la cabeza varía normalmente entre 0 y 200 microvoltios [μV], y las frecuencias aparecen en el rango de hasta 50 [Hz], aunque bajo condiciones particulares, pueden llegar a verse señales alrededor de los 100 [Hz] [30]. Cabe resaltar además que, en las personas sanas, la mayoría de las señales del EEG pueden clasificarse en *Alpha*, *Beta*, *Theta*, *Delta* y *Gamma*, siendo las bandas frecuenciales ampliamente conocidas que normalmente aparecen en el EEG, como se resume en la siguiente tabla:

Tabla 1. Bandas frecuenciales más comunes en un EEG de un paciente sano [21]-[31].

Nombre	Frecuencia [Hz]	Descripción
Delta	0.5 - 4	Son las señales de menor frecuencia, comunes en los adultos al dormir/soñar. También aparecen normalmente en los bebés.
Theta	4 - 8	Aparecen durante estados de mucha relajación o meditación, aunque también se han observado en niños pequeños y durante estados de somnolencia en adolescentes y adultos.
Alpha	8 - 14	Aparecen cuando se cierran los ojos y en estados de relajación moderada.
Beta	14 - 30	Se asocian con el movimiento (a través de la corteza motora) y el pensamiento activo. También aparecen en estados de concentración, alerta y ansiedad.
Gamma	> 30	Se asocian a la corteza somatosensorial, siendo frecuentes durante una alta concentración y al resolver problemas (tareas de procesamiento cognitivo).

Diversas actividades y acciones han sido probadas para estudiar las variaciones y características que se pueden presentar en las bandas frecuenciales mencionadas durante ciertas acciones o como consecuencia de ciertos estímulos. Particularmente, es importante destacar que las Ondas Alpha son ampliamente utilizadas para probar los dispositivos de adquisición de señales EEG: estas ondas ocurren más intensamente en la región occipital, que viene siendo también la zona de la corteza visual (aunque también se pueden obtener de las regiones parietales o frontales), y precisamente su utilidad para probar fácilmente los dispositivos EEG es la sensibilidad que tiene cuando el paciente cierra y abre los ojos.

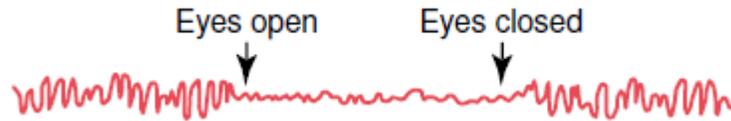


Figura 8. Ejemplo ilustrativo del comportamiento de mediciones EEG mostrando la disminución y aumento de voltaje de la señal al abrir y cerrar los ojos, respectivamente. Imagen extraída de [27]

En efecto, el *parpadeo* se registra en el EEG como un cambio súbito de voltaje durante la transición entre los ojos abiertos y los ojos cerrados, que viene siendo el cambio entre un ritmo Alpha y un ritmo Beta (menor voltaje).

2.10 Brain-Computer Interface (BCI)

Una interfaz cerebro-computador (BCI, por sus siglas en inglés) representa la denominación que se le da a cualquier sistema que toma mediciones del SNC y las convierte en señales orientadas a reemplazar, restaurar, mejorar, suplementar, potenciar o asistir las interacciones naturales del SNC, de manera que se efectúen cambios entre las respuestas del SNC y el entorno objetivo (interno o externo).

Comúnmente las BCI se diseñan en base al registro de la actividad eléctrica cerebral, buscando la comunicación y transmisión de información por medio de los pensamientos de la persona, sin necesidad de movimientos físicos (respuestas de la corteza motora). Esto, principalmente porque los usuarios o público objetivo son en general pacientes de enfermedades o trastornos del sistema nervioso que afectan la movilidad natural del cuerpo.

El término BCI ha tomado distintos enfoques, debido a los distintos procedimientos tanto *invasivos* (requieren cirugías o penetración física del cuerpo) como *no-invasivos*, pero en general el electroencefalograma (EEG) es la técnica principal utilizada para controlar los distintos dispositivos BCI, basando su funcionamiento en la interpretación de cierta información extraída del EEG para poder ejecutar alguna acción.

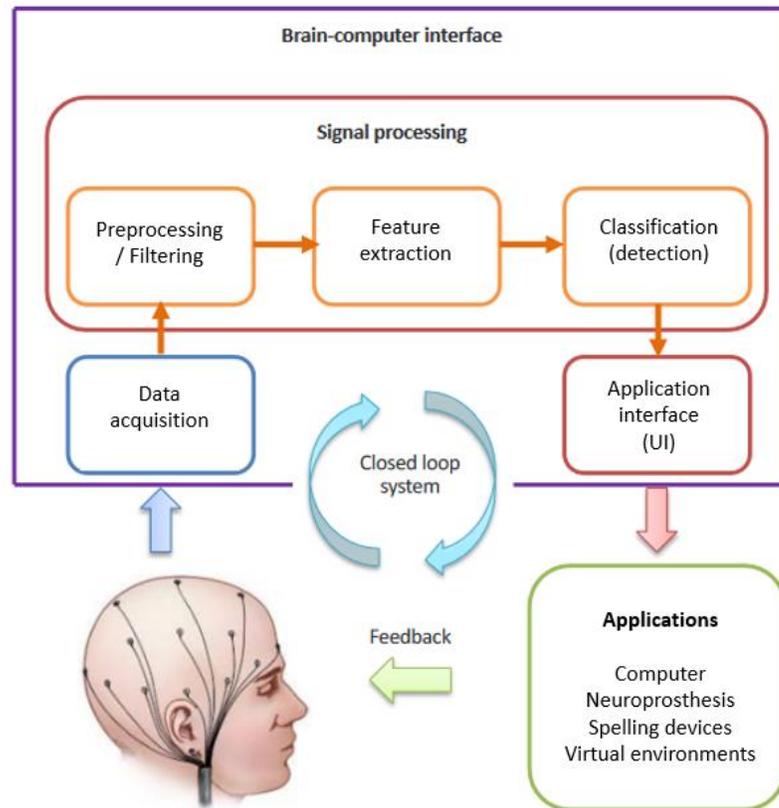


Figura 9. Diagrama general de una BCI, mostrando los módulos y partes principales. Imagen basada en el diagrama mostrado en [32].

Como se observa en la figura anterior, los sistemas BCI en general se componen de tres partes clave: el sector encargado de adquirir los datos correspondientes a las señales EEG del paciente, el posterior conjunto de elementos que realizan el procesamiento de los datos adquiridos y la parte de la interfaz que ve/interactúa con el usuario que está utilizando la BCI; al juntar las distintas operaciones en conjunto, se puede tener un mejor entendimiento del funcionamiento global de una BCI, la cual sin tener en cuenta el fin específico, requiere de los elementos mencionados para poder funcionar de manera correcta. A manera aclaratoria, el sector que muestra una “aplicación” (recuadro verde del diagrama) se refiere a el dispositivo que presenta la interfaz que interactúa con el usuario, el cual puede ser un computador, entorno virtual, prótesis o cualquier otro elemento que depende de la aplicación deseada.

2.11 Event-Related Potential (ERP)

Los potenciales relacionados a eventos (conocidos como ERP por sus siglas en inglés) representan una técnica de análisis de señales EEG basada en los potenciales eléctricos originados por el cerebro que están asociados a “eventos” específicos (estímulos, respuestas, decisiones, entre otros), los cuales pueden ser de naturaleza interna o externa. Se derivan de las mediciones EEG, por lo que son información que puede ser tomada de forma no invasiva y de prácticamente cualquier grupo demográfico, siendo una fuente de datos muy importante para numerosos estudios que se han realizado en el campo de la neurociencia [31].

Los ERP representan una técnica ampliamente utilizada por investigadores, centros de diagnóstico y tratamientos, que según las numerosas investigaciones [31]–[35], provee de una gama amplia de información relacionada con características particulares de la actividad eléctrica cerebral del paciente. Lo anterior se origina, principalmente, porque normalmente la visualización directa de efectos o características especiales en un EEG es una tarea muy compleja, derivada de la numerosa cantidad de factores asociados que pueden alterar las mediciones.

Internamente, los ERP se originan como *Potenciales Postsinápticos (PSPs)* (PSPs por sus siglas en inglés), generados como resultado de la adición de miles de potenciales derivados de neuronas organizadas a lo largo de la corteza cerebral. Así, las diversas neuronas responden a un *evento* particular y generan una respuesta cerebral consistente asociada al evento que las activó, el cual puede evidenciarse bajo distintos experimentos. Es decir, para poder realizar mediciones de los ERP, miles de neuronas de ciertas partes de la corteza cerebral deben coordinarse para sumar las respuestas nerviosas individuales, y las principales células que responden de esta manera son las llamadas *células piramidales*, conocidas como las principales células de entrada-salida en la corteza cerebral [31], por lo que la mayoría de las mediciones de los ERP reflejan neurotransmisiones que ocurren en estas células piramidales.

Ahora bien, existe una gran variedad de experimentos y eventos de los que se derivan respuestas ERP específicas, utilizadas en estudios para distintas patologías, pero para

entender las señales de interés de este proyecto (los P300), se hace necesario mencionar ciertas características importantes de los ERP:

- La clasificación de los distintos ERP que se pueden generar se hace en los llamados *Componentes ERP*, que representan la actividad eléctrica cerebral medida desde el cuero cabelludo y que es generada como una respuesta neuroanatómica cuando una operación computacional específica se realiza en las mediciones EEG [32]. Es decir, un componente ERP puede ocurrir en diferentes momentos y bajo diferentes condiciones, siempre que se mantenga el evento asociado y que se represente la misma función cognitiva de interés.
- Gran parte de los términos y efectos de los ERP provienen de manera experimental, como resultados de las investigaciones realizadas. En ese aspecto, las convenciones de los nombres de los Componentes ERP más aceptadas son las siguientes: se comienza con una P o una N, lo cual indica si el componente tiene una forma de onda positiva o negativa (respectivamente), y se coloca seguido de un número que representa normalmente la latencia pico de la señal, o alguna posición de un máximo local dentro de la forma de onda. Así, algunos ejemplos de componentes ERP son P100, N170, P300, N400, entre varios otros.
- La mayoría de los experimentos y pruebas para ERP se basan en la adición y el *promedio* de los denominados *Epochs*, término que representa un “fragmento” determinado de un EEG, y al realizar sucesivas pruebas, se promedian epochs de sectores similares de un EEG (como respuesta a ciertos estímulos) para lograr visualizar el ERP deseado. Este proceso se explicará en detalle en el capítulo de desarrollo del proyecto (ver Capítulo 3).

Teniendo en cuenta las ideas anteriores, algunos componentes ERP comúnmente encontrados en estudios ERP son los que aparecen en la siguiente figura:

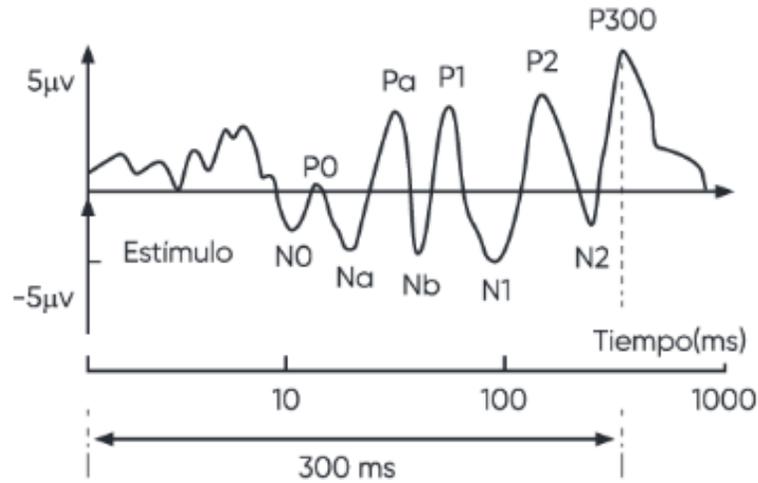


Figura 10. Formas de onda ilustrativas que ejemplifican algunos componentes de ERP comúnmente encontrados en estudios de neurociencia. La amplitud de las señales es de referencia (puede variar), pero se puede observar los distintos picos que se pueden asociar a un estímulo aplicado. Imagen extraída de [38].

2.12 El Paradigma Oddball

Muchas de las técnicas utilizadas para poder visualizar un ERP dependen del tipo de estímulo utilizado para activar los procesos sinápticos, y dichos estímulos normalmente son auditivos o visuales por la facilidad para que las BCI y los experimentos de prueba se puedan realizar (aunque en la literatura también existen estímulos somatosensoriales) [32].

En el caso de la “generación” de los componentes ERP de tipo P300, uno de los métodos clásicos y más efectivos para poder visualizarlos es basarse precisamente en el Paradigma Oddball para poder visualizar este fenómeno: dentro de la psicología, el paradigma Oddball es un paradigma experimental que consiste en presentarle al sujeto de prueba (o paciente) una serie de estímulos auditivos o visuales en serie, con la particularidad de que ocasionalmente aparezca un estímulo diferente a los anteriores, ante lo cual la persona debe estar atenta y, dependiendo del experimento, realizar algún proceso cognitivo particular.

El paradigma Oddball fue utilizado por primera vez en una investigación de ERP realizada en la Universidad de California (San Diego) en el año 1975, y en 1988 se convirtió

en la base del proyecto realizado por Farwell y Donchin como el primer sistema BCI basado en P300, el cual propusieron para implementar un deletreador [36]. Desde ahí, ha sido la clave para numerosos sistemas y experimentos, sobre todo aquellos relacionados con el componente P300.

2.13 P300

El componente ERP de interés para este proyecto, y en el que se desea basar el funcionamiento de la BCI, es el P300, también conocido como P3. Representa uno de los potenciales relacionados a eventos más utilizados en diversos experimentos [16], [31]–[34], siendo normalmente asociado al paradigma Oddball. Su nombre, proviene del hecho de que la forma de onda está caracterizada como una deflexión positiva (P), y “teóricamente” la señal presenta su máxima amplitud 300 milisegundos [ms] después del estímulo presentado, aunque según los variados experimentos está comprobado que los P3 pueden aparecer entre 250 y hasta 650 [ms] aproximadamente después de presenciado el estímulo [3], [4], [37].

La onda P300 se investiga comúnmente junto con el paradigma Oddball porque, precisamente, el desarrollo de experimentos que muestran estímulos frecuentes y *estímulos infrecuentes* hacen que se pueda evidenciar este tipo de señales; de hecho, muchas veces los experimentos de P300-Oddball son llamados las *pruebas del estímulo infrecuente*, debido a que es el efecto de presenciar dichos estímulos lo que ocasiona la organización de los procesos sinápticos en ciertas regiones de la corteza cerebral, siendo ese estímulo infrecuente el *target* u objetivo que se desea identificar.

Como se verá más adelante en el desarrollo y pruebas realizadas, el P300 no depende únicamente de funciones cognitivas, sino que aspectos como la motivación, frustración, fatiga mental y ansiedad derivada de errores impactan directamente en este tipo de experimentos. De hecho, la amplitud de un P3 es mayor cuando el paciente de la prueba se esfuerza más mentalmente en una tarea particular al presenciar un estímulo particular (*target*),

y de manera similar la amplitud es menor si el sujeto o paciente de prueba no sabe si cierto estímulo era o no el target [32], [36].

Para integrar las ideas anteriores, se tomará como referencia la Figura 11 (siguiente página), para poder explicar un caso de ejemplo de un experimento simple de P300-Oddball. En él, a un sujeto de prueba se le colocan los electrodos de medición para tomar señales EEG (Figura 11-A), y el sujeto se ubica frente a una pantalla donde solamente aparecerán dos estímulos posibles: una “X” o una “O”. Es decir, el sujeto (o varios voluntarios, de ser posible) observarán secuencias de Xs y Os de manera aleatoria, teniendo en cuenta que 80% de los estímulos presentados en pantalla serán Xs y el 20% restante serán Os, por lo que el *estímulo infrecuente* es la O. Conectado a la pantalla, hay un dispositivo para que la persona presione un botón cuando ve las Xs y otro cuando aparezcan las Os, y dichas letras (estímulos visuales) se muestran en pantalla cada 100 [ms] seguidas por una pantalla blanca que dura unos 1400 [ms] (la pantalla blanca representa un intervalo entre estímulos). Mientras el voluntario realiza estas tareas, se toman mediciones EEG y se envían a un computador para filtrar y procesar los datos obtenidos. Adicionalmente, cada vez que se presenta un estímulo (X u O) el sistema coloca una marca que se graba junto con las mediciones EEG.

Siguiendo con el ejemplo, en la Figura 11-B se reflejan las mediciones EEG con sus respectivas marcas, tomadas con un único canal de medición correspondiente a un electrodo ubicado en *Pz*, que es la zona central del lóbulo parietal. Los registros EEG se marcan y por ende se dividen en los correspondientes *Epochs* que, como se había mencionado previamente,

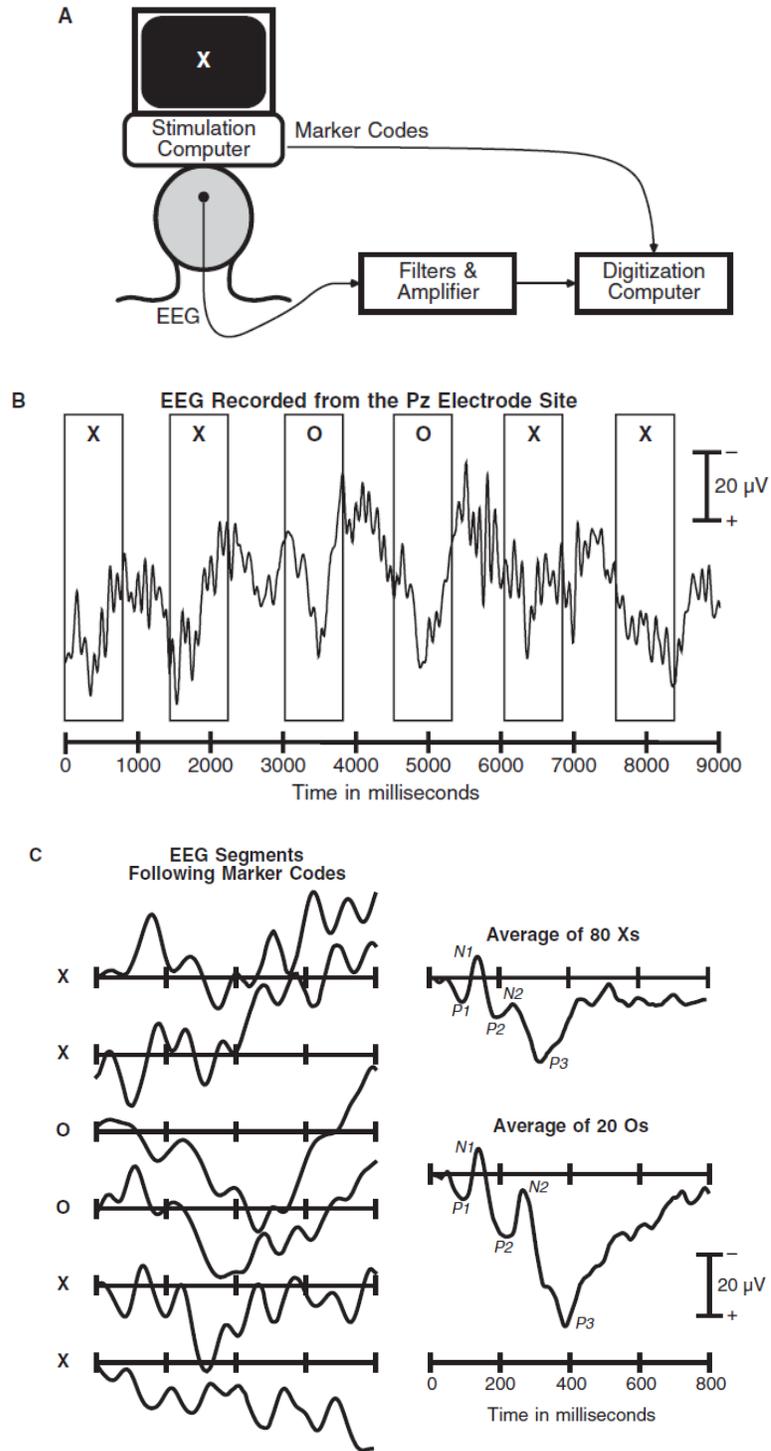


Figura 11. Ilustraciones de referencia para ejemplificar un experimento simple de P300-Odball. La figura 11-A representa la pantalla con los estímulos visuales presentados, la cual se conecta al equipo de procesamiento de datos, filtrado y adquisición (EEG). La figura 11-B representa las mediciones EEG. La figura 11-C (izquierda) se refiere a ciertos Epochs del EEG, y a la derecha están los ERP asociados a los dos estímulos posibles. Imagen extraída de [32].

son fragmentos del registro EEG delimitados por marcadores o eventos de interés. En este sentido, los datos de las mediciones muestran las respuestas grabadas ante cada estímulo, pero a simple vista, no resulta sencillo identificar cómo son realmente las amplitudes y formas de las mediciones, a pesar de que se puede apreciar cierta consistencia en las respuestas. Para solucionar eso, al final de cada sesión se realizan operaciones simples sobre las señales de respuesta, promediando los Epochs correspondientes a las Xs y Os de cada estímulo, según el EEG (Figura 11-C, parte izquierda), y dicho promedio de las señales genera lo que aparece en la Figura 11-C (derecha), es decir, el ERP correspondiente a los estímulos. A grandes rasgos, de esta manera se obtiene el ERP y, consecuentemente, el P300 asociado a los estímulos presentados, con lo cual se evidencia que el P3 de los estímulos infrecuentes (las Os, que son los targets) tiene una amplitud mucho mayor que el P3 asociado a los estímulos frecuentes (las Xs), siendo coherente con el paradigma Oddball.

Dando cuenta del ejemplo anterior, así funcionan de manera global muchos de los experimentos basados en P300, bien sea para controlar una BCI o para investigaciones sobre ciertos pacientes. Ahora bien, el paradigma Oddball no es la única forma para producir una respuesta P300, sino que existen diversas técnicas para lograr esto:

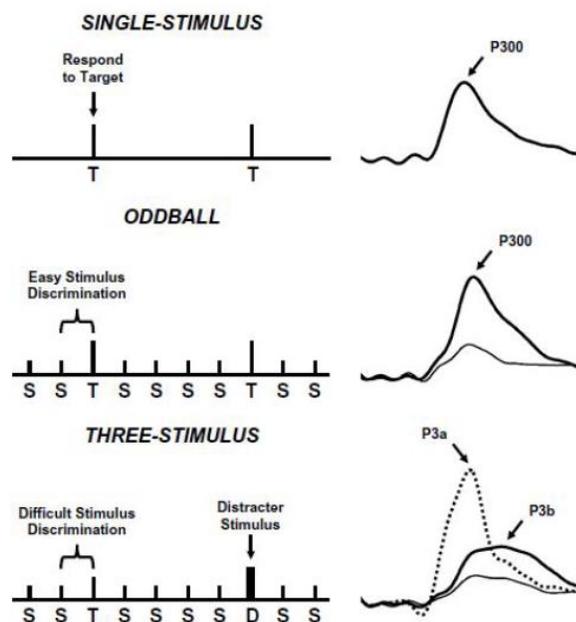


Figura 12. Tres protocolos conocidos para generar respuestas P300. Imagen extraída de [39].

Como se indica en la Figura 12, el primer protocolo mencionado (parte superior de la figura) consiste en presentarle al sujeto de prueba el estímulo deseado (target) cada cierto tiempo, esperando que al ser observado se pueda suscitar una respuesta P300. El segundo protocolo (parte central de la figura) es el ya mencionado paradigma Oddball; es importante mencionar que, usando este método, la BCI realmente no “sabe” si cierto estímulo (por ejemplo, un carácter en una pantalla) es o no el target, lo cual hace que sea un sistema adecuado para los experimentos de deletreo (*Spellers*). El tercer protocolo (parte inferior de la Figura 12) es el más complejo, porque la secuencia de estímulos que se le presentan a la persona incluye ahora un “distractor”: cuando la persona observa el estímulo distractor, se genera la respuesta conocida como *P3a* (en la Figura 12, es la curva punteada en la parte inferior), y al observar el target se origina la curva *P3b* (al lado de la *P3a* previamente indicada) [37]. Ahora bien, en la literatura de las investigaciones realizadas, no siempre *P3a* y *P3b* se asocian a los mismos componentes o variaciones de una respuesta P300 “estándar”, sino que pueden aparecer cuando el sujeto de prueba realiza ciertas tareas mentales durante la observación de los estímulos, variando de autor en autor [32]. Por eso, y para propósitos de este proyecto, todas las referencias hacia las respuestas P300 serán basadas en el paradigma Oddball.

2.14 P300-Speller y sistemas BCI basados en P300

Una de las características clave que se probó en diversas investigaciones con P300 es que al sujeto de prueba se le pueden mostrar una gran cantidad de estímulos, y no necesariamente restringir los experimentos a decisiones binarias (True/False o de sólo dos posibilidades), observándose igualmente los efectos del paradigma Oddball y, como resultado, lograr identificar un *target* particular por sobre los demás estímulos. Es decir, es posible tener un abanico de estímulos que representen diversas opciones, de modo que, si la persona selecciona una de esas como el target, el sistema podría identificar la opción escogida según las respuestas a los estímulos (bajo ciertas condiciones de operación).

Las ideas anteriores son las que dieron pie a los conocidos *Spellers*, que son sistemas BCI basados en P300 con el objetivo de deletrear palabras, mediante diversos estímulos asociados a caracteres alfanuméricos.



Figura 13. Ejemplo de una matriz 6x6 utilizada como fuente de los estímulos visuales para un *Speller* basado en P300. Imagen extraída de [39].

Como se observa en la Figura 13, una pantalla muestra la matriz con caracteres, y el sistema consiste en que cada cierto tiempo se configura que la fila o columna resaltada cambie, es decir, todos los caracteres de una fila o columna se muestran de color blanco y el resto de la matriz aparece en un gris más oscuro. Dicha matriz, manejada por un software especial, se diseña para que la persona pueda seleccionar cualquier carácter y de esta manera deletrear alguna palabra, mediante la identificación de las respuestas P300.

Basados en este principio, se han realizado pruebas experimentales cambiando los caracteres por símbolos o imágenes en algunos casos [3], [4], [13], y se ha encontrado que, de manera similar, es posible identificar el target seleccionado por la persona según las respuestas P300, teniendo en cuenta que se sigue cumpliendo el paradigma Oddball.

2.15 PCA (Principal Component Analysis)

Para la manipulación de los datos obtenidos de las mediciones EEG existen diversas técnicas, procedimientos y algoritmos disponibles. Entre esos, un grupo de los más utilizados son los *algoritmos de filtrado espacial* [37], los cuales ayudan en el proceso de promediar los valores obtenidos por los electrodos de medición, y lograr filtrar ciertos datos que no son de interés para el experimento, pero, durante este proceso surgen diversos problemas que dificultan el procesamiento de los datos, lo cual hace que sea necesario usar técnicas adecuadas para mejorar el tratamiento de la información, y una de las más utilizadas en experimentos P300 es la del PCA.

El PCA es una técnica estadística que principalmente se enfoca en la reducción de dimensionalidad de los datos adquiridos [38], permitiendo simplificar espacios muestrales de alta complejidad y al mismo tiempo conservar la información clave. En muchos casos, se escoge como herramienta cuando los datos tienen demasiadas variables que dificultan la selección de las características deseadas.

A modo global, PCA permite que se pueda encontrar un número de “factores subyacentes”, que se denominarán z como ejemplo, los cuales explican la información clave dentro de un conjunto de datos, sin necesitar todos los datos originales. Es decir, suponiendo una muestra n con p variables (X_1, X_2, \dots, X_p) , en dicho espacio muestral de p dimensiones ahora se necesitan z valores para caracterizar alguna prueba o experimento, y no todos los p valores originales ($z < p$). Así, cada una de las z nuevas variables se denomina una *componente principal*.

Cada una de las componentes principales (Z_i) se origina al combinar linealmente las variables originales. Entonces, por ejemplo, la primera componente de un grupo de variables (X_1, X_2, \dots, X_p) resulta de la siguiente operación:

$$Z_1 = \phi_{11}X_1 + \phi_{21}X_2 + \dots + \phi_{p1}X_p$$

La operación anterior es la combinación lineal normalizada de las variables. Además, es importante resaltar que en dicha operación se cumple que:

$$\sum_{j=1}^p \phi_{j1}^2 = 1$$

Donde los términos $\phi_{11}, \dots, \phi_{1p}$ reciben el nombre de *loadings* y son los que definen a la componente principal, dándoles de cierta manera el peso/importancia a la hora de determinar la información que recoge cada variable. Así, dado un set de datos X con n observaciones y p variables, el procedimiento general que normalmente se sigue para lograr calcular una componente principal es el siguiente:

- **Centralización de las variables:** para lograr que todas las variables tengan una media de cero, se busca restar a cada uno de los valores la media de la correspondiente variable asociada.
- **Encontrar los *loadings* para maximizar la varianza:** en este paso, el objetivo es encontrar el valor de los loadings para lograr maximizar la varianza, lo cual normalmente está ligado a un problema de optimización. Para resolverlo, una de las formas utilizadas implica el cálculo de los *eigenvector-eigenvalue* (también conocidos como vector y valor propio) de la matriz de covarianzas.

Posteriormente de haber calculado la primera componente (Z_1) se procede a calcular la segunda (Z_2) siguiendo igualmente el procedimiento indicado, pero se agrega la restricción de la primera componente dentro de la combinación lineal, es decir, que no puede existir correlación con la primera componente. Así, el procedimiento se realiza de forma iterativa hasta lograr calcular todas las posibles componentes, o alternativamente hasta que se decida detener la iteración.

2.16 LDA (Linear Discriminant Analysis)

El otro algoritmo ampliamente utilizado para los experimentos P300 es el LDA [21], [36], [37], [39], el cual aparece para realizar la etapa de *clasificación* de los datos. Es, en esencia, un algoritmo de clasificación basado en *machine learning*, y representa a su vez a toda una familia de algoritmos que se derivan de éste y que buscan ayudar en los procesos de identificación de los P300.

El funcionamiento del LDA se basa en buscar una combinación lineal de características que separan dos o más clases de objetos o eventos de interés, y dicha separación se realiza al computar las direcciones (llamadas “*linear discriminants*”) que representan ejes que maximizan la separación entre múltiples clases. Al ser un algoritmo de clasificación lineal, las clases son separadas por líneas o hiperplanos, aunque variaciones del algoritmo se pueden usar para tener otras formas que separen las clases, como ocurre con el QDA (*Quadratic Discriminant Analysis*) que permite manejar límites de decisión con formas curvas [40].

El algoritmo es orientado para la clasificación, por lo que realiza predicciones al estimar la probabilidad de que una nueva muestra pertenezca a cada clase según los valores de entrenamiento. Por ello, gran parte del funcionamiento del LDA se basa en esencia en la aplicación del Teorema de Bayes [40]:

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

Adicionalmente, se puede suponer que cada clase C tiene una media μ_i y la misma covarianza Σ . Entonces, la variabilidad de la dispersión entre clases se puede definir por la covarianza muestral de las medias de las clases:

$$\Sigma_b = \frac{1}{C} \sum_{i=1}^C (\mu_i - \mu) (\mu_i - \mu)^T$$

Donde μ es la media de las medias de las clases. Así, la separación entre clases en una dirección ω está dada por:

$$S = \frac{\vec{\omega}^T \Sigma_b \vec{\omega}}{\vec{\omega}^T \Sigma \vec{\omega}}$$

La expresión anterior la utilizan en la literatura para para indicar que, cuando ω es un *eigenvector* (vector propio) de $\Sigma^{-1} \Sigma_b$, la separación entre clases será igual al correspondiente *eigenvalue* (valor propio) [40].

Ahora bien, la amplia utilización del LDA para P300 también se debe a que el algoritmo tiene una serie de *suposiciones* por defecto respecto a los datos que se van a manejar:

- **Cada característica (variable, dimensión o atributo) en la base de datos sigue una distribución normal/gaussiana**, es decir, se asume que los datos en sí están distribuidos de manera gaussiana. De hecho, el potencial del LDA viene en gran parte por esta propiedad, ya que los P300 son por naturaleza similares a una gaussiana, y si el algoritmo se basa en esta morfología para clasificarlos y detectarlos, se convierte en una herramienta muy apropiada para este tipo de datos.
- **Cada característica tiene la misma varianza**, lo cual indica que el valor de cada característica de interés varía alrededor de la media con la misma cantidad en promedio.
- **Cada característica se asume que fue muestreada aleatoriamente.**
- **Existe una falta de multicolinealidad en las características independientes.** Esto es porque, el potencial de predicción disminuye al aumentar la correlación entre las variables de predicción.

2.17 Artefactos

Todas las señales que son detectadas en un EEG pero que no representan actividad eléctrica cerebral o cuyo origen no es explícitamente de la corteza cerebral son denominadas *artefactos*. En las mediciones EEG y en todos los experimentos, tratamientos o pruebas que impliquen este tipo de señales, se presentan artefactos que contaminan la señal EEG pura, principalmente debido a una de las mayores desventajas del EEG: tiene una relación Señal a Ruido (*SNR*) muy baja, lo cual hace que diversos tipos de señales no deseadas aparezcan en los registros EEG muy frecuentemente [4], [21], [36], [37].

La facilidad para que diversos artefactos aparezcan en el EEG es lo que da origen principalmente a las diversas técnicas de filtrado y procesamiento de los datos, ya que la corrección de los datos adquiridos garantiza en buena parte la calidad de los resultados y la extracción de la información de interés. En este sentido, buena parte de los artefactos en los EEG son de origen biológico (señales del cuerpo que se mezclan en los registros), entre los que se pueden mencionar los siguientes:

- EOG (*Electrooculography*), lo cual representa las señales eléctricas debidas a los movimientos oculares. Además, se incluyen los parpadeos y la actividad muscular extraocular.
- ECG (*Electrocardiography*), que representa las señales cardíacas como producto de la actividad normal del corazón.
- EMG (*Electromyography*), originados por la actividad muscular voluntaria o involuntaria.

2.18 Marco Legal o Normativo

Para este apartado se tuvo en cuenta el artículo 11 de la resolución 8430 de 1993 del ministerio de salud [41] de donde se plantea que, según el apartado relacionado a las investigaciones en seres humanos, el presente prototipo implementado dentro del campo de las BCI está en la categoría de “sin riesgo”. En ese sentido, como se especifica en la resolución, el proyecto involucra distintas técnicas y procedimientos de investigación experimental que bajo ningún concepto involucran total o parcialmente intervenciones o modificaciones de tipo fisiológico, además de que no se alteran tampoco variables psicológicas o biológicas de las personas voluntarias que hacen parte de las pruebas de la investigación. Adicionalmente, todos los procedimientos utilizados son del tipo “no invasivos”, por lo que no representan ningún riesgo potencial para la salud.

3 METODOLOGÍA

La metodología utilizada para el desarrollo del proyecto abarca las etapas desde el diseño básico y selección de componentes principales, hasta el diseño final de la interfaz de usuario que ejecuta los algoritmos de decisión y se comunica con dispositivos de prueba para verificar las acciones identificadas. Todos estos elementos se integraron para que operen como el prototipo de una BCI orientada a la identificación de comandos.

A continuación, se presenta un esquema general de las etapas involucradas en el desarrollo del proyecto:

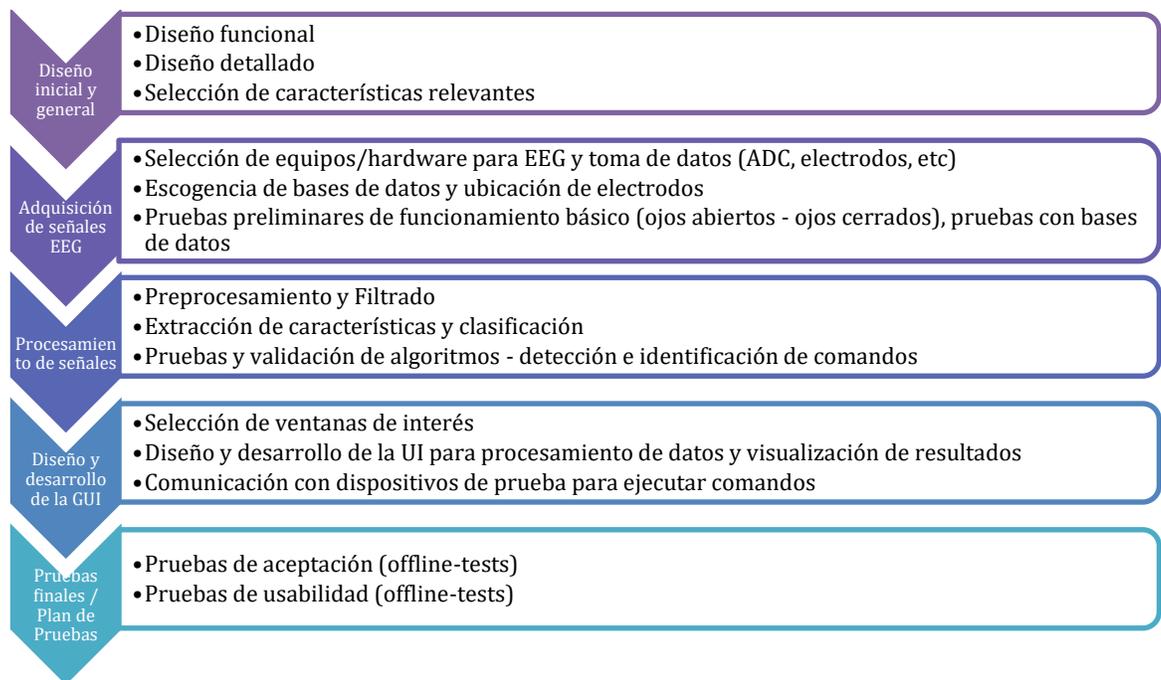


Figura 14. Diagrama de la metodología general con las etapas principales para el desarrollo del proyecto.

3.1 Diseño general del prototipo

Para la estructura global del prototipo se tuvieron en cuenta dos factores principales: la operación y puesta en prueba del sistema depende de la interacción entre las señales de adquisición y la interfaz que evalúa los datos adquiridos; y adicionalmente, la comunicación con los dispositivos de prueba para ejecutar las acciones debe originarse de la interfaz que identifique los comandos. Teniendo en cuenta lo anterior, la función de transferencia general que representa el prototipo global es la siguiente:

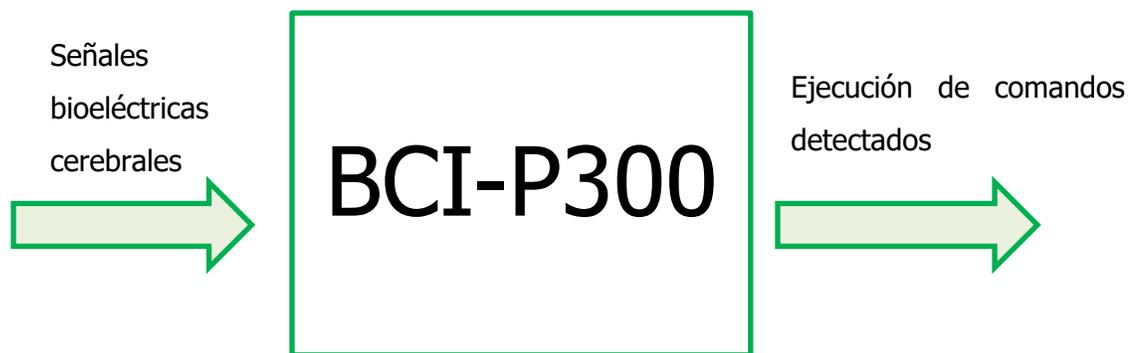


Figura 15. Diagrama de la función de transferencia general (caja negra) del prototipo propuesto.

3.2 Diseño funcional

La estructura general del prototipo se originó de la Figura 15 mostrada previamente, la cual integra a su vez los subsistemas básicos de una BCI como se pudieron observar en la Figura 9 mostrada anteriormente en el capítulo 2 (sección descriptiva de las BCI dentro del marco conceptual). Siguiendo los lineamientos anteriores, se construyó un diagrama detallado con cada uno de los subsistemas del prototipo, compuesto además por las interacciones entre las señales de la actividad eléctrica cerebral y la detección de los P300 como base para poder identificar las acciones que se transmiten a los dispositivos de prueba.

El diagrama mencionado se muestra a continuación:

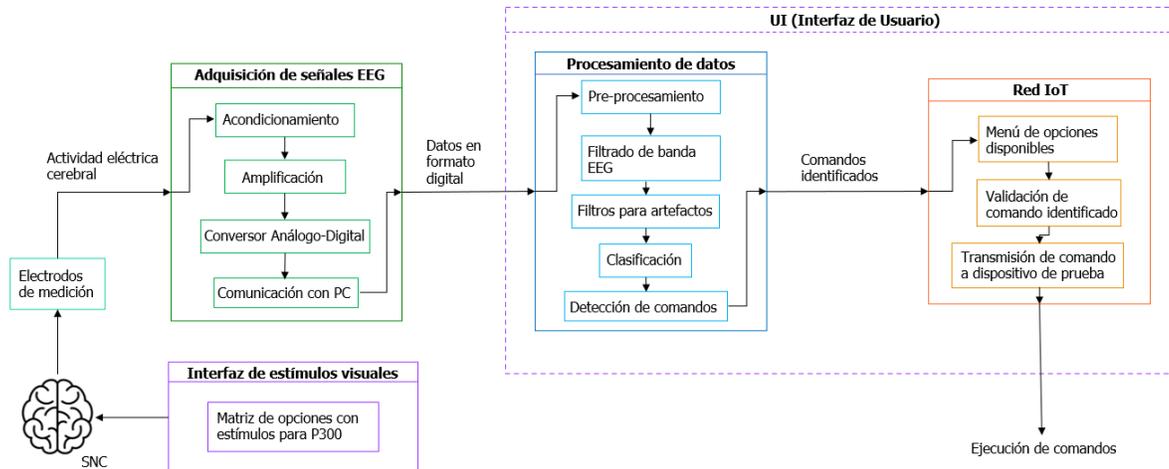


Figura 16. Diseño funcional del proyecto.

El prototipo de la BCI en desarrollo en su totalidad basándose en dos interfaces funcionalmente diferentes que separan las etapas de adquisición de señales de los demás subsistemas: la primera interfaz presenta una matriz con las opciones posibles (comandos) que el usuario puede escoger, y en dicha interfaz se presentan los estímulos visuales de manera periódica para que se origine la actividad eléctrica cerebral que da lugar a los P300; la otra interfaz gráfica se utiliza para poder realizar operaciones de procesamiento y visualización de los datos adquiridos, además de que realiza la comunicación con los dispositivos de prueba para poder ejecutar los comandos identificados.

A continuación, teniendo en cuenta la estructura descrita anteriormente y la Figura 16 como referencia, se describen cada uno de los subsistemas del prototipo, siguiendo el orden funcional estándar para el funcionamiento de todo el sistema:

3.2.1 Interfaz de estímulos visuales

Es la parte inicial porque principalmente es la fuente de los estímulos que va a observar la persona, lo cual origina los procesos sinápticos que dan lugar a los P300. En esta interfaz se presenta esencialmente la matriz con los comandos posibles que la persona puede

seleccionar, por lo que su finalidad principal es estimular el SNC para poder tomar las mediciones necesarias.

3.2.2 Adquisición de señales EEG

Previo al experimento práctico, la colocación de los electrodos sobre la cabeza de la persona es clave para poder tomar las mediciones de la mejor manera posible, y en esencia, eso es lo que se describe antes de este subsistema: el SNC es estimulado por la interfaz, y los electrodos situados sobre el cuero cabelludo captan la actividad eléctrica cerebral. Posteriormente, las señales medidas por los electrodos son enviadas a un dispositivo que empieza el proceso de tratamiento de las señales bioeléctricas.

3.2.3 Acondicionamiento

Una vez que las señales de los electrodos se distribuyen en canales que son recibidos por un equipo EEG, lo que se recibe en este subsistema son pequeñas señales analógicas producto de las mediciones. Las señales analógicas son distribuidas y referenciadas eléctricamente según las configuraciones del experimento, y posteriormente son enviadas a una etapa de amplificación.

3.2.4 Amplificación

Las señales analógicas son todavía muy pequeñas para poder ser analizadas (del orden de los microvoltios [μV]), lo cual hace que en esta etapa se le adicione una referencia eléctrica positiva o negativa (según el caso) para poder amplificar la señal hasta un nivel más adecuado.

3.2.5 Conversor Análogo-Digital (ADC)

Una de las etapas clave es ésta, porque hasta este punto las señales en el sistema siguen siendo analógicas, pero en esta etapa se convierten a señales digitales que ya pueden ser transmitidas a un equipo para su posterior análisis/procesamiento.

3.2.6 Comunicación con PC

Las señales ahora digitalizadas son enviadas a un computador mediante protocolos de comunicación, para poder analizarlas en algún software específico.

3.2.7 UI (*User Interface* o Interfaz de Usuario)

Los datos ya en formato digital son recibidos en el computador y leídos en un software para poder procesar la información de las mediciones. Ahora bien, la interfaz implementada maneja tanto la parte de procesamiento como la de la red IoT, permitiendo realizar tareas de filtrado y a la vez identificar los comandos para ejecutar.

3.2.8 Procesamiento de datos

Una vez que se obtienen los datos en formato digital, hay que realizares un tratamiento para poder visualizar la información de interés. Sabiendo de antemano la baja SNR y potenciales artefactos, hay que agregar etapas de filtrado adicionales para poder visualizar la información de manera correcta, lo cual representa el propósito principal de este subsistema.

3.2.9 Preprocesamiento

Los datos de los canales de las mediciones contienen, además de los voltajes medidos por los electrodos, datos adicionales de los registros internos de las tarjetas de adquisición y algunos otros bytes de información que no forman parte de los registros EEG, por lo que deben ser removidos de las señales adquiridas.

3.2.10 Filtrado de banda EEG

El espectro frecuencial de las señales EEG va normalmente entre 0.2 – 50 [Hz], por lo que en esta parte se sitúa el filtro pasabanda que permite delimitar los datos adquiridos de las mediciones, para poder visualizar correctamente las señales.

3.2.11 Filtros para artefactos

Las señales EOG, ECG y EMG principalmente tienen componentes que se mezclan con los registros EEG, por lo que en este apartado se colocan filtros adicionales para eliminar todos los posibles artefactos que distorsionan y afectan las mediciones EEG de interés.

3.2.12 Clasificación

Una vez que las señales de las mediciones ya están filtradas y estructuralmente correctas, se procesa a la sección en donde algoritmos computacionales buscan analizar los datos obtenidos para, en primer lugar, representar las mediciones como ERPs, y en segundo lugar, lograr detectar los componentes P300 y con eso asociarlos a los comandos disponibles. Así, mediante los procesos de *extracción de características*, se busca extraer la información de interés de las señales digitalizadas.

3.2.13 Detección de comandos

Las señales procesadas se representan de manera que se busca identificar el comando o target según las características definidas. Mediante diversas técnicas, se implementan algoritmos para identificar el comando que se deriva de las señales analizadas, y una vez detectado el comando (con un margen de posibilidad), es posible transmitirlo hacia el dispositivo que lo ejecuta.

3.2.14 Red IoT (*Internet of Things*)

El objetivo de los comandos (o acciones) es ser transmitidos a un dispositivo para poder ser ejecutados. En este sentido, el canal de comunicación y la orden a realizar cambian dependiendo del comando identificado, pero en esencia, los comandos se direccionan hacia el dispositivo que le corresponde para poder ejecutarlo.

3.2.15 Menú de opciones disponibles

Una vez identificado el comando, se envía a un algoritmo que le asigna una identificación según el menú de opciones disponibles (las mismas que previamente visualizó la persona en la interfaz de estímulos).

3.2.16 Validación de comando identificado

Al evaluar el comando identificado dentro de las opciones preexistentes, se direcciona la opción identificada hacia el dispositivo correspondiente.

3.2.17 Transmisión de comando a dispositivo de prueba

La opción identificada se envía a el dispositivo que está previamente configurado para poder ejecutar los comandos deseados, mediante los protocolos de comunicación necesarios, y finalmente se ejecuta el comando.

3.3 Diseño detallado

Todos los subsistemas fueron implementados por medio de los componentes de hardware y software descritos en la Figura 17. Cada uno de los componentes se describen a lo largo del capítulo de Metodología.

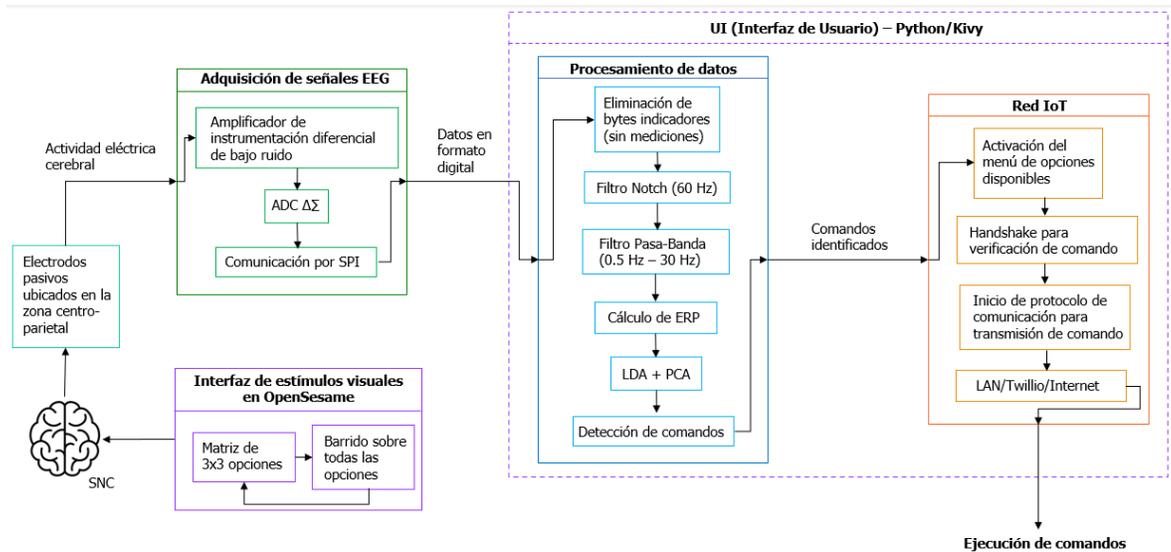


Figura 17. Diseño detallado del prototipo propuesto.

Ya conociendo la estructura detallada de todo el sistema, a continuación se irán describiendo cada una de las etapas y factores que fueron necesarios para la implementación del prototipo. Primeramente, la selección de características es la base sobre la que se desarrolla la interfaz para poder estimular al SNC y con esto poder suscitar las respuestas P300, por lo que representa el paso previo necesario para organizar los demás elementos.

3.3.1 Elección de características

El manejo de las BCI por P300 implica conocer aquellos factores que permiten identificar correctamente los ERP para poder tomar criterios de decisión. En este sentido, en el campo de la ciencia de datos y *Machine Learning* (los cuales posee una estrecha relación con la neurociencia) las *características* son precisamente todos aquellos criterios que permiten identificar y caracterizar el fenómeno o elemento de interés [3], [4], [37].

En el caso de los P300 como componentes dentro de los ERP, las diversas investigaciones realizadas[37], [39], [42]–[45] muestran que las señales P300 tienen en esencia sólo dos características fundamentales: la amplitud y la latencia. Es decir, la identificación de los componentes P300 dentro de un ERP se puede realizar simplemente según los parámetros de amplitud y latencia, sin embargo, es importante tener presente ciertos aspectos que impactan directamente en estos criterios:

- La amplitud se define como la diferencia de voltaje existente entre la referencia establecida como línea base y el mayor pico positivo que se presente en una ventana de tiempo predefinida. Por lo anterior, establecer la ventana correcta y el momento específico en el que se presenta un estímulo son factores clave que afectan la identificación de las características.
- Una señal P300 alcanza su máximo teóricamente 300 [ms] después de presentado el estímulo, pero como ya se mencionó en el Capítulo 2, en realidad ese valor puede variar entre 250 – 700 [ms] aproximadamente. De hecho, la latencia del P300 puede variar no sólo entre individuos, sino en un mismo individuo [46]: las pruebas en diferentes estudios muestran que las amplitudes y latencias de respuestas P300 varían entre poblaciones “clínicas” y “normales” e incluso, bajo las mismas condiciones sobre un mismo individuo.
- Las respuestas P300 se ven influenciadas por una amplia cantidad de factores. Por ejemplo, un estudio reciente realizado por investigadores de varios países [47], mostró que reducir el campo visual al presentar los estímulos ayuda a reducir considerablemente las amplitudes de los estímulos que no son el *target*, resaltando que normalmente los elementos del entorno funcionan como “distractores” que afectan las respuestas de las personas y, por ende, las mediciones de las que se derivan los P300.

3.3.2 Adquisición de señales EEG

A continuación se detallarán los distintos subsistemas que formaron parte de esta etapa, necesaria y clave como punto de partida para poder iniciar el manejo de las señales EEG.

3.3.2.1 Selección de bases de datos

Antes de poder manejar señales reales, lo más recomendable por los diferentes expertos en neurociencia es practicar, entrenar y estudiar los fenómenos, formas de onda y comportamientos asociados a las señales EEG mediante bases de datos preexistentes, las cuales contienen mediciones reales realizadas durante experimentos específicos y que sirven como referencia y guía para entender el comportamiento de los paradigmas de interés, algoritmos y demás factores asociados.

Para este proyecto, se utilizaron principalmente dos (2) referencias de bases de datos:

- Datasets de la *BCI Competition*, que son unas bases de datos de libre acceso brindadas por el *Wadsworth Center - NYS Department of Health*, que es el laboratorio de salud pública de investigación en Nueva York. Las mediciones provienen de pruebas en un Speller que usa P300 como paradigma de funcionamiento, mediante una matriz que intensifica las columnas-filas al sujeto que visualiza la pantalla. Además, las configuraciones y datos principales del experimento referente son: 36 clases, 64 canales EEG (0.1-60Hz), frecuencia de muestreo de 240Hz, 85 sesiones de entrenamiento y 100 sesiones de prueba, grabadas mediante el sistema *BCI2000*. Las bases de datos están disponibles en la web, mediante los links de referencia [48], [49].
- Datasets que provee *Marijin van Vliet*, investigador y experto en el campo de la neurociencia, el cual mediante su perfil en la plataforma web GitHub, brinda las bases de datos de libre acceso para poder realizar diversos tutoriales y pruebas que sirven como guías en la manipulación y procesamiento de señales EEG.

Ahora bien, las bases de datos mencionadas provienen de experimentos basados en P300: el primero de ellos, llamado “The Magic Trick” (el truco de magia), es una prueba que consiste en presentarle nueve (9) cartas (del mazo de cartas inglés) a un sujeto de manera sucesiva durante un período de tiempo definido, e intentar adivinar cuál era la carta objetivo del sujeto mediante el cálculo del ERP y los componentes P300 al utilizar técnicas especializadas de procesamiento de señales EEG; el segundo experimento, consiste en que al sujeto se le presentan seis (6) íconos en una pantalla de los cuales debe seleccionar mentalmente uno, y cada uno de los íconos era resaltado diez (10) veces, buscando que la persona se concentrara en el ícono objetivo (target), para posteriormente procesar los datos e intentar identificar el target mediante la identificación de los componentes P300 (utiliza un clasificador basado en el algoritmo LDA). Las bases de datos están disponibles desde los links de referencia [50].

El manejo de las bases de datos mencionadas fue un importante paso para poder identificar las formas de onda características en experimentos P300, procedimientos seguidos para calcular los ERP, probar ejemplos de algoritmos de filtrado y clasificación, identificar estructuras de datos principales y varios otros elementos. Algunos ejemplos de estas pruebas preliminares con las bases de datos, las cuales sirvieron como líneas base y guías, son mostrados en la sección 3.3.2.5.1 – *Pruebas con bases de datos* del presente Capítulo.

3.3.2.2 Selección de equipos (Hardware)

Los equipos necesarios para poder adquirir las señales EEG se seleccionaron después de realizar exhaustivas investigaciones en el mercado, al comparar criterios de accesibilidad, calidad, precio y comentarios previos de usuarios. Teniendo en cuenta dichos criterios, además de los requerimientos previamente expuestos (Capítulo 1), se destacan principalmente los siguientes parámetros necesarios:

- Tasa de muestreo de mínimo 200 SPS (*Samples per Second*).

- Al menos tres (3) canales de adquisición de manera simultánea.
- ADC con resolución superior a 8 bits.

Para el prototipo propuesto, y teniendo en cuenta los diagramas del diseño funcional y el diseño detallado, es importante explicar el papel fundamental que cumple el ADC (*Analog-to-Digital Converter*) para el presente proyecto: todo el sistema en conjunto busca, mediante el procesamiento de señales EEG, identificar comandos que mentalmente se escogen de un menú de opciones disponibles, lo cual es posible mediante la medición y adquisición de la actividad eléctrica cerebral; el proceso asociado de convertir dichas señales analógicas derivadas del SNC en señales digitales que puedan ser analizadas computacionalmente es el papel clave del ADC, y es desde ese punto que el resto de la BCI puede funcionar. De allí surge la importancia de enfocar la selección de componentes en las especificaciones del ADC.

La tabla comparativa de los dispositivos candidatos se encuentra en el Anexo A del presente documento. Todos los equipos que ahí se listan superan los requisitos mínimos establecidos, por lo que la selección se basó en varios criterios que optimizaran el cumplimiento de los requerimientos del prototipo propuesto.

Al realizar distintas comparaciones y luego de consultas en investigaciones similares, se determinó que la tarjeta ADS1299EEG-FE de Texas Instruments cumplía con todas las características necesarias para poder implementar el proyecto basado en P300, contando con especificaciones que se adaptaban a las pruebas que se proyectaba realizar. En particular, el ADC de la tarjeta de adquisición resultó ser uno de los más referenciados en distintos proyectos a nivel mundial [15], [17], [18], [51]–[53], presentando una buena confiabilidad. Además, era el producto más económico y por lo tanto accesible para poder realizar el prototipo, por lo que se adquirió el respectivo kit de desarrollo ADS1299EEG-FE de Texas Instruments como base para la adquisición de señales EEG.

3.3.2.2.1 ADS1299

El dispositivo de Texas Instruments es un ADC (*Analog-to-Digital Converter*) de ocho canales, bajo ruido, 24-bits de resolución, muestreo simultáneo delta-sigma ($\Delta\Sigma$), que internamente posee varios *amplificadores de ganancia programable* (PGA por las siglas inglesas), referencia interna y señales de prueba.

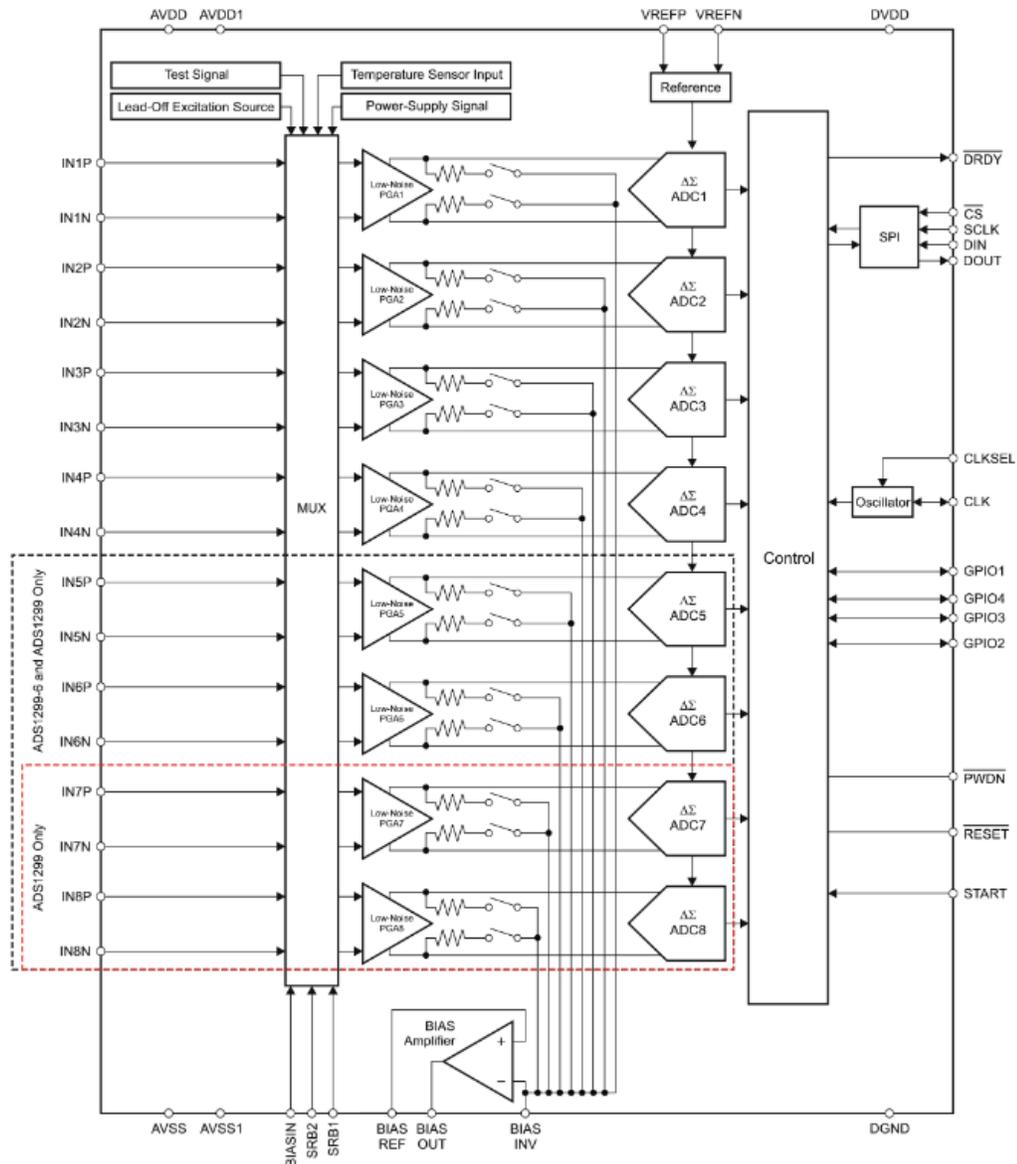


Figura 18. Diagrama de bloques funcional del ADS1299. Imagen extraída del datasheet del fabricante [56].

Como se muestra en la Figura 18, el circuito puede lidiar con hasta ocho (8) señales de entrada diferencial (referencia positiva y negativa para cada una); dichas entradas van a un multiplexor que controla las señales para habilitar o deshabilitar diferentes tipos de análisis como temperatura, alimentación, cortocircuito o referencias eléctricas externas. A la salida de los MUX, cada señal se captura en un amplificador de ganancia programable (PGA), lo cual permite multiplicar el valor de la señal de entrada por valores de 1, 2, 4, 6, 8, 12 o 24.

Posteriormente a los PGA's, las señales EEG amplificadas son muestreadas por un ADC $\Delta\Sigma$: estos representan moduladores que sobremuestran las señales, seguidos por filtros digitales/diezado que juntos producen una salida de datos en alta resolución [52]. Un posible diagrama de bloques de un convertidor $\Delta\Sigma$ se muestra en la Figura 19:

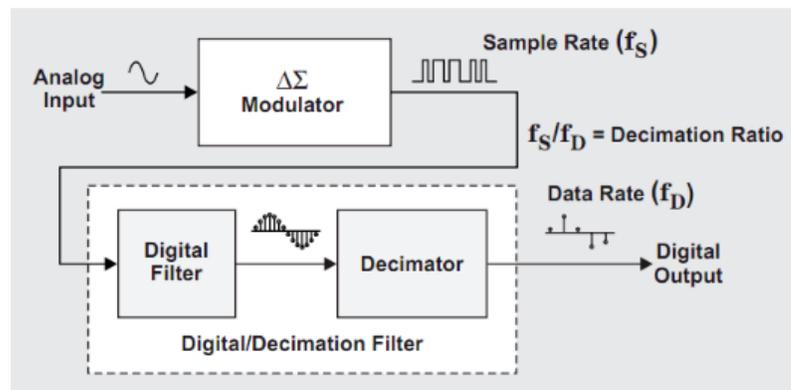


Figura 19. Diagrama de ejemplo de un conversor $\Delta\Sigma$. Imagen extraída de [52].

Una de las características de este ADC es aprovechar la técnica del sobremuestreo: permite mejorar la relación señal-a-ruido (SNR) incrementando la frecuencia de muestreo respecto a la frecuencia de Nyquist, lo cual posibilita utilizar filtros anti-alias simples debido a que las réplicas están muy lejos en el espectro frecuencial. Los ADCs $\Delta\Sigma$ son además circuitos rápidos cuyo costo global se reduce al trasladar las operaciones más complejas (como la integración) al dominio digital, donde los circuitos son más rápidos y económicos.

Las características principales del ADS1299, como se indica en el datasheet del fabricante[54], son las siguientes:

- Ocho canales de alta resolución y muestreo simultáneo.
- Ocho PGA's de bajo ruido, con ganancia de hasta 24.
- CMRR: -110 dB
- Tasa de muestreo: 250 SPS hasta 16 kSPS.
- Alimentación unipolar o bipolar:
 - Analógica: 4.75 V - 5.25 V
 - Digital: 1.8V - 3.6V
- Oscilador integrado (genera un reloj de 2.048 MHz).
- Manejo de referencia interna o externa.
- Interfaz de comunicación por SPI.

El ADS1299 cuenta en total con 24 registros de 8 bits, de los cuales la mayoría son de lectura/escritura y permiten configurar una gran cantidad de opciones en el dispositivo. Asimismo, mediante la comunicación por SPI, se pueden establecer una serie de comandos que controlan el funcionamiento de las operaciones principales. Los comandos establecidos se listan en la siguiente figura:

COMMAND	DESCRIPTION	FIRST BYTE	SECOND BYTE
System Commands			
WAKEUP	Wake-up from standby mode	0000 0010 (02h)	
STANDBY	Enter standby mode	0000 0100 (04h)	
RESET	Reset the device	0000 0110 (06h)	
START	Start and restart (synchronize) conversions	0000 1000 (08h)	
STOP	Stop conversion	0000 1010 (0Ah)	
Data Read Commands			
RDATAC	Enable Read Data Continuous mode. This mode is the default mode at power-up. ⁽¹⁾	0001 0000 (10h)	
SDATAC	Stop Read Data Continuously mode	0001 0001 (11h)	
RDATA	Read data by command; supports multiple read back.	0001 0010 (12h)	
Register Read Commands			
RREG	Read <i>n nnnn</i> registers starting at address <i>r rrrr</i>	001 <i>r rrrr</i> (2xh) ⁽²⁾	000 <i>n nnnn</i> ⁽²⁾
WREG	Write <i>n nnnn</i> registers starting at address <i>r rrrr</i>	010 <i>r rrrr</i> (4xh) ⁽²⁾	000 <i>n nnnn</i> ⁽²⁾

(1) When in RDATA mode, the RREG command is ignored.

(2) *n nnnn* = number of registers to be read or written - 1. For example, to read or write three registers, set *n nnnn* = 0 (0010). *r rrrr* = starting register address for read or write commands.

Figura 20. Comandos preconfigurados en el ADS1299 para realizar las operaciones principales por SPI. Imagen extraída de [54].

3.3.2.2 Tarjeta de Adquisición: ADS1299EEG-FE

El kit de evaluación completo para el ADS1299 es el ADS1299EEG-FE, el cual es un hardware que consta de un sistema embebido con el ADS1299 dentro de una PCB conectada físicamente a una EVM (*Evaluation Module*).



Figura 21. Kit de desarrollo y evaluación ADS1299EEG-FE. Imagen extraída de [55].

Como se observa en la Figura 21, el sistema cuenta con dos tarjetas principales: la ubicada en la parte superior, es la que contiene el ADS1299 junto con diversos pines y conexiones para los electrodos, señales de pruebas, jumpers de configuración y periféricos; la tarjeta de la parte inferior es la MMB0, diseñada alrededor del TMS320VC5507/9 DSP[56], la cual no está destinada como tarjeta para procesamiento digital de señales (DSP) (y no se adquiere de manera separada por el fabricante), sino que cumple la función principal de permitir la conexión del ADS1299EEG-FE a un computador mediante un puerto USB, además de proveer de la alimentación eléctrica necesaria.

La PCB del ADS1299EEG-FE cuenta con diversos jumpers de configuración que permiten manipular varias opciones del hardware para poder cambiar funciones o conexiones

internas específicas, como la alimentación unipolar o bipolar, activar/desactivar puertos de lectura para los electrodos, activar/desactivar señales de pruebas y varias otras opciones. Esto, se complementa con algunos LEDs e indicadores en la MMB0 que permiten saber si la tarjeta está alimentada eléctricamente, además de algunas indicaciones adicionales.

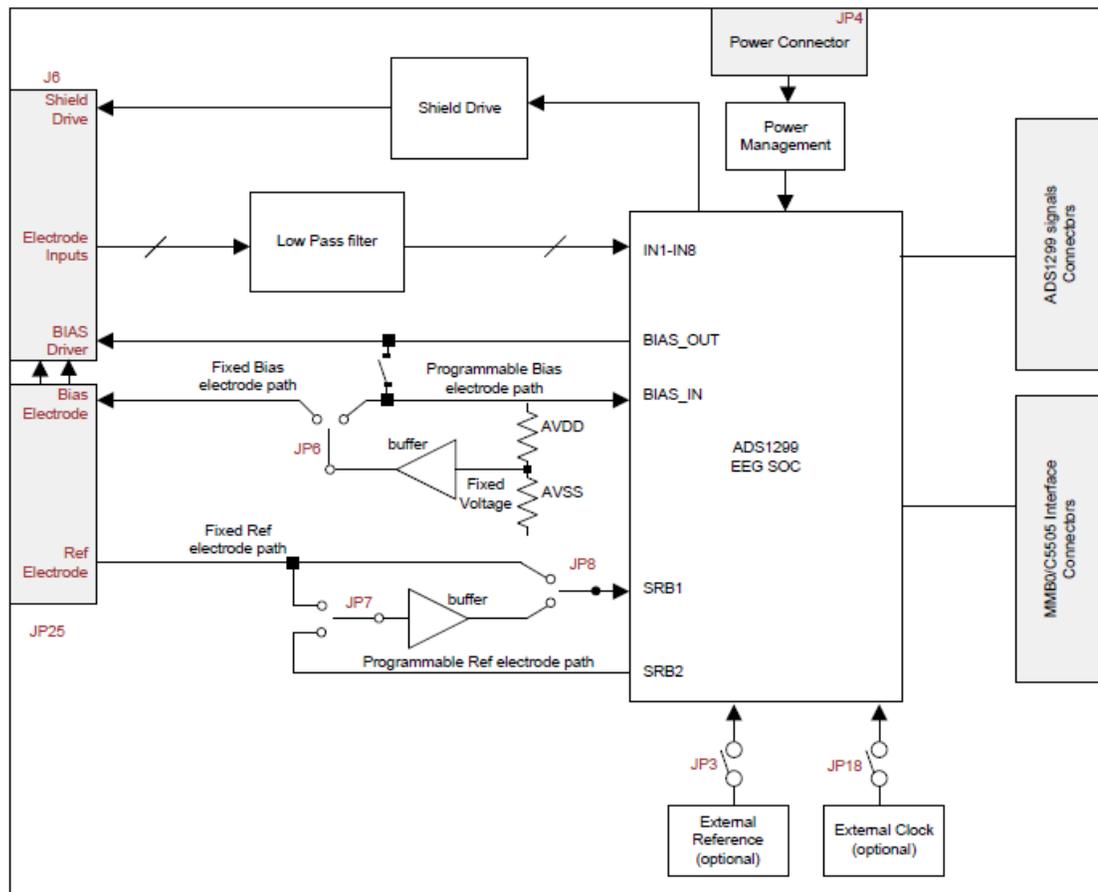


Figura 22. Diagrama de bloques del ADS1299EEG-FE. Imagen extraída de [55].

Como se observa en la Figura 22, el ADS1299EEG-FE cuenta con varios conectores que permiten manejar las configuraciones para la adquisición de señales. Particularmente, los jumpers J6 y JP25 (parte izquierda del diagrama) son los puntos de conexión de los electrodos de medición, y son los que manejan la mayoría de las configuraciones para el tipo de adquisición de datos.

Es importante destacar que el kit viene con un software disponible para su uso, sin ningún costo adicional asociado. Como se explicará en detalle en la sección de pruebas preliminares, dicho software fue utilizando inicialmente para poder validar el funcionamiento de la tarjeta completa, además de permitir validar distintas configuraciones de los registros y así encontrar la configuración más apropiada para adquirir señales EEG. El software no es de código abierto y no permite realizar procesamientos complejos en las señales, pero viene con una GUI (interfaz gráfica) muy simple que ayuda a realizar pruebas de funcionamiento.

3.3.2.3 Selección y ubicación de electrodos

Dentro de la adquisición de señales EEG, los electrodos cumplen una función primordial: son los encargados de captar las actividad eléctrica cerebral y con esto originar las señales iniciales que posteriormente se pueden analizar y convertir en datos digitales. Para lograr esto, existen diversos tipos de electrodos disponibles en el mercado, pero principalmente se dividen en cuatro grandes grupos: *pasivos/activos*, teniendo en cuenta el tipo de acondicionamiento con el que manejan las señales, o *secos/húmedos*, según los protocolos específicos para poder adquirir las señales de interés.



Figura 23. Gold cup electrodes de OpenBCI. Imagen extraída de [57].

Para el ADS1299, se siguieron recomendaciones dadas por investigaciones similares [15], [17], [18], [52], [53] y por la empresa *OpenBCI*, la cual se especializa en productos para diversos proyectos y aplicaciones en neurociencia. Así, se decidió adquirir los llamados “Gold Cup Electrodes” que se indican en la Figura 23, unos electrodos pasivos con conexión tipo jumper y recubrimiento en oro, ampliamente utilizados para adquisición de señales EEG.

Una vez adquiridos los electrodos, el proceso de ubicación y colocación depende del tipo de dispositivo escogido: la diferencia primordial entre electrodos pasivos y activos es una preamplificación y acondicionamiento adicional existente en los electrodos activos, que en algunos casos mejora la calidad de las señales adquiridas, a la vez que aumenta el costo del producto; por otro lado, el término de electrodos secos o húmedos viene del hecho de que los electrodos húmedos requieren utilizar un gel o pasta adicional que se debe aplicar en la zona de medición.



Figura 24. Pasta Ten20 para adquisición de señales EEG. Imagen extraída de [58].

Los electrodos seleccionados son originalmente secos, pero se recomienda ampliamente usarlos con una pasta conductora para que se mejore la calidad de las señales adquiridas. Para este proyecto, la pasta seleccionada fue la Ten20, que es la mostrada en la Figura 24: es una sustancia altamente conductora que se debe colocar en contacto con el electrodo de medición y sobre la zona de interés del cuero cabelludo.

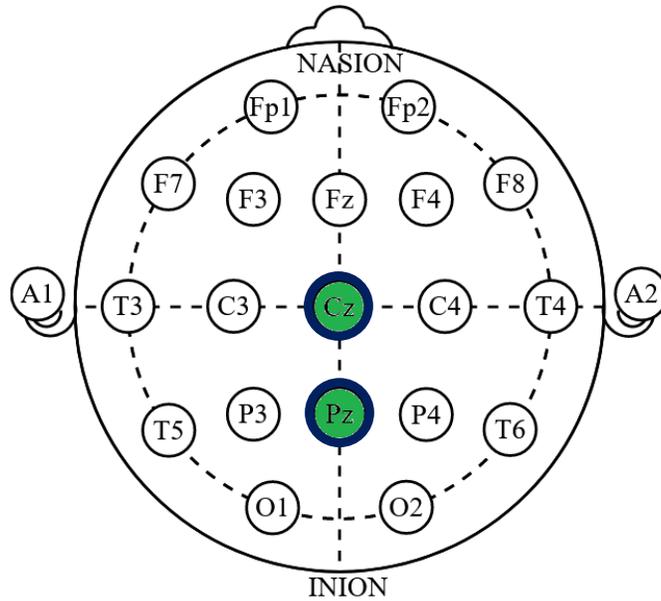


Figura 26. Localizaciones escogidas para los electrodos del prototipo: Cz y Pz, según el sistema 10/20. Imagen base extraída de [60].

3.3.2.4 Transmisión de datos a Python (Windows)

Una vez ubicados y colocados los electrodos, la siguiente etapa fue transmitir las mediciones EEG hacia el computador, para poder analizar las señales en Python, el lenguaje de programación escogido para el procesamiento y manipulación de los datos. El sistema operativo (OS) del computador al cual se transmitían los datos es un equipo basado en *Windows 10*, con las siguientes características principales:

- Motherboard: ASUS PRIME H510M-E
- SO: Windows 10 Pro (64-bits), versión 21H2
- Procesador: 12th Gen Intel(R) Core(TM) i5-12400
- RAM: 16 GB
- Tarjeta gráfica: NVIDIA GTX 1650
- Disco Duro: 1TB (SSD)

3.3.2.4.1 Comunicación con el ADS1299 por SPI

Para poder realizar la transmisión de datos, bien sea para almacenamiento o visualización en tiempo real, se utilizó la comunicación por SPI desde el ADS1299. SPI (*Serial Peripheral Interface*) es un protocolo de comunicación síncrono que generalmente se utiliza para conectar circuitos integrados a una misma tarjeta o para enviar datos entre microcontroladores y periféricos pequeños como lo pueden ser sensores, registros de desplazamiento y tarjetas SD [52]. Se basa en una arquitectura de maestro-esclavo donde el maestro es único y el número de circuitos esclavos no está limitado en principio, aunque normalmente no excede los cuatro (4) o cinco (5) dispositivos. Las principales características del protocolo SPI son:

- *Transmisión full dúplex*: existe un cable de comunicación para los datos que van desde el maestro hacia los esclavos, y otro cable para los datos que van desde los esclavos hacia el maestro. Es decir, el maestro y los esclavos siempre están operando como transmisores y receptores.

- *Comunicación de 4 hilos*:
 - *SPICLK*: la señal de reloj, siempre generada por el maestro.
 - *MOSI: Master Out Slave In*, que representa el puerto de salida de datos para el maestro, y la línea de entrada de datos para los esclavos.
 - *MISO: Master In Slave Out*, que representa el puerto de entrada de datos para el maestro, y la línea de salida de datos para los esclavos.
 - *SPISS, SS o CS*: es el puerto de *Chip Select* o *Slave Select*, que es el que habilita el circuito integrado hacia el que se envían los datos.

- *Transmisión de datos*: un número predefinido de bits son los que se envían en cada transmisión. Un bit es enviado por cada ciclo de reloj utilizando la codificación NRZ (*Non Return to Zero*) [52]. El SPICLK se inactiva cuando no se realiza transmisión, pero su nivel depende de la polaridad del reloj:
 - *CPOL = 0*: el reloj es *low* (0 lógico) cuando está inactivo.

- $CPOL = 1$: el reloj es *high* (1 lógico) cuando está inactivo.
- Maestro y esclavo deben estar de acuerdo en:
 - Número de bits en la transmisión
 - Frecuencia del reloj: reloj generado por el maestro, recibido por los esclavos. Los esclavos deben soportar dicha frecuencia.
 - Polaridad del reloj (CPOL) y fase del reloj (CPHA).

La interfaz compatible con SPI del ADS1299 consiste en cuatro señales \overline{CS} , $SCLK$, DIN y $DOUT$. Mediante las señales mencionadas, se pueden leer los datos medidos, leer/escribir registros y controlar el funcionamiento del ADS1299:

1. \overline{CS} (*Chip Select*): es la señal necesaria para activar la transmisión. Debe ir en estado low (0 lógico) para efectuarse el envío de datos.
2. $SCLK$: es el reloj para la comunicación serial.
3. $DOUT$ y DIN : son las señales $MISO$ y $MOSI$ respectivamente.

Una señal adicional, llamada \overline{DRDY} (*data-ready*), se utiliza como señal de status para indicar cuándo están listos los datos (\overline{DRDY} es un 0 lógico cuando nuevos datos están disponibles). A modo de referencia, la Figura 27 muestra la estructura completa del bus de datos para la transmisión por SPI:

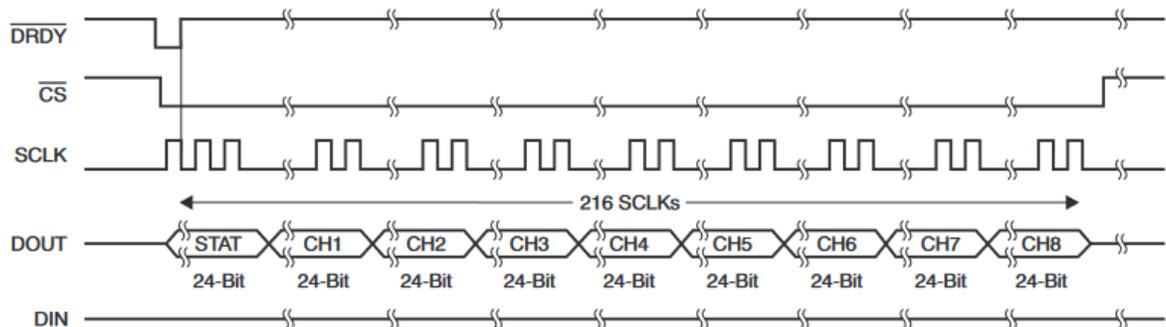


Figura 27. SPI Bus Data Output. Imagen extraída de [54].

Ahora bien, es muy importante explicar un aspecto encontrado: a pesar de que físicamente el ADS1299EEG-FE cuenta con una interfaz para SPI, directamente no es capaz de transmitir los datos hacia el computador. En primer lugar, esto se evidenció al comprobar que la EVM adquirida directamente con el fabricante sólo viene con un software que no es de código abierto y no permite guardar las mediciones; adicionalmente, la comunicación con el computador requiere originalmente el uso de la MMB0 (mediante el puerto USB) como “puente” para poder leer y visualizar los datos en el software del fabricante (TI).

Por lo anterior, fue necesario no sólo diseñar e implementar una interfaz completamente separada del software original del fabricante para poder adquirir y visualizar las señales EEG, sino que para la transmisión de los datos por SPI directamente desde el ADS1299 se requirió del uso de una tarjeta auxiliar: un Arduino basado en el microcontrolador ATmega328p. Con ello, fue posible extraer directamente desde el ADC las respectivas mediciones, siguiendo el mismo protocolo SPI mencionado previamente. El sistema en cuestión se muestra en la siguiente figura:

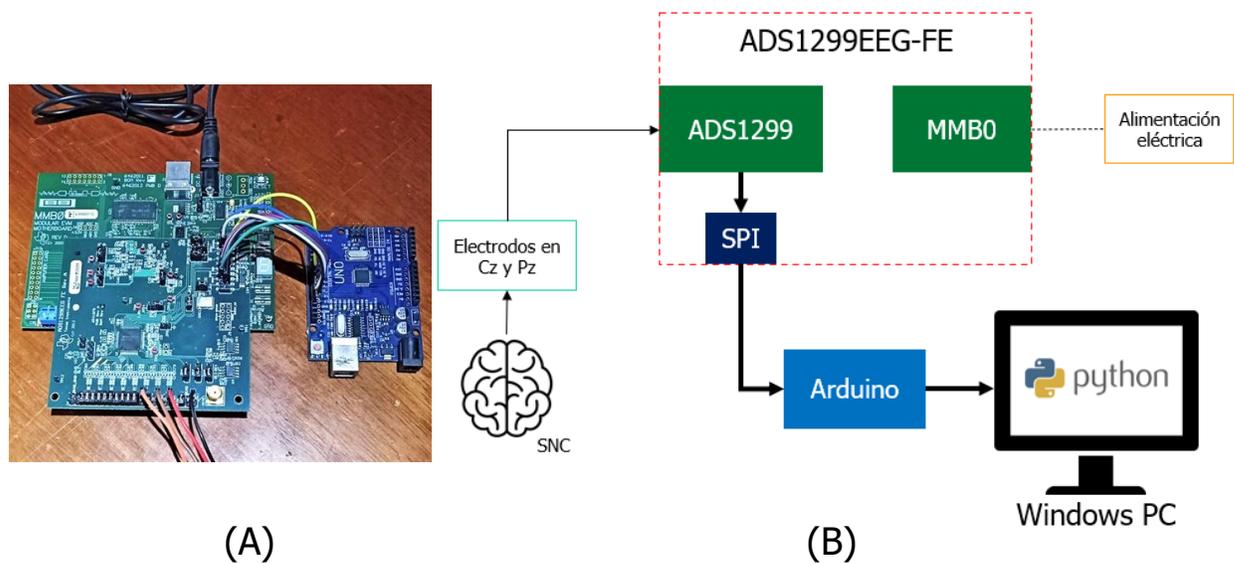


Figura 28. (Izquierda) Imagen tomada del ADS1299EEG-FE conectado por SPI al Arduino. (Derecha) Diagrama ilustrativo del flujo de los datos y conexiones para la transmisión de las señales hacia Python. Fuente: el autor.

Como se observa en la Figura 28-A (izquierda), el ADS1299 se conecta directamente con el Arduino por SPI, y posteriormente como se indica en la Figura 28-B (derecha), desde el Arduino es posible transmitir los datos al computador, para ser leídos en Python. Para lograr esto, se utilizó como base una librería de comandos e instrucciones diseñada para comunicar el ADS1299 con tarjetas Arduino, creada por Conor Russomanno, que es un investigador de neurociencia además de cofundador y actual CEO de *OpenBCI* (la empresa de dispositivos para EEG mencionada anteriormente). La librería, que es de código abierto, está disponible desde la página web del GitHub del creador [61].

Según la librería, los comandos y configuraciones no fueron específicamente diseñados para una versión particular de Arduino (UNO, Mega, DUE o adicionales), sino que teóricamente deberían funcionar para la mayoría de las variaciones si se configura correctamente la interfaz SPI. Así, para el presente proyecto, y por motivos de simplicidad y accesibilidad, se utilizó el Arduino UNO como tarjeta auxiliar para la transmisión de datos, recordando que es un microcontrolador basado en el ATmega328p, con las siguientes características principales: 14 pines I/O, 6 entradas analógicas, reloj de 16 MHz, memoria de 2KB SRAM y 32KB FLASH, voltaje I/O de 5V, voltaje de entrada (nominal) entre 7-12V, comunicación por UART/I2C/SPI y conector USB del tipo USB-B [61].

3.3.2.4.2 Lectura de mediciones desde Python

Todo el procesamiento, interpretación de mediciones e interfaces gráficas fue realizado en Python, un lenguaje de programación administrado por la *Python Software Foundation* que posee una licencia de código abierto. Es un lenguaje bastante referenciado y con una amplia variedad de librerías disponibles para todo tipo de aplicaciones, permitiendo el desarrollo de proyectos para diversas plataformas (iOS, Android, Windows, web y varias adicionales).

Para poder adquirir y leer las señales en Python, la transmisión de datos por SPI se direcciona al computador mediante un puerto USB y se reciben los datos, normalmente en

formato binario. Es decir, las mediciones de los electrodos (pequeñas señales analógicas) se transforman en voltajes numéricos que pueden ser interpretados en un lenguaje de programación (señales digitales). En este punto es menester mencionar la importancia de seleccionar el tipo de entrada analógica, que puede ser de dos tipos:

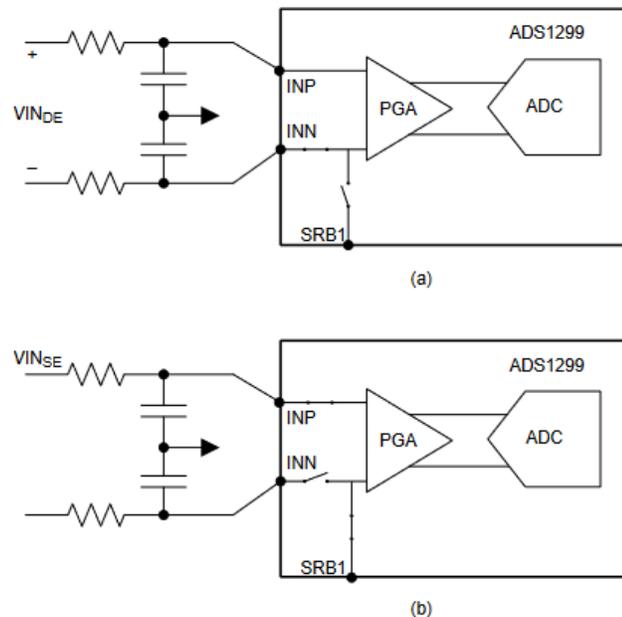


Figura 29. Configuraciones aceptadas por el ADS1299EEG-FE. (a) Entrada diferencial. (b) Entrada unipolar (Single-ended). Imagen extraída de [55].

Siguiendo lo indicado en la Figura 29, escoger entre entradas diferenciales o Single-ended afecta la cantidad de electrodos para colocar, siendo que cada terminal INP e INN (*Input Positive* e *Input Negative*) estaría ligado a una señal particular. Por ello, para el prototipo se utilizó la configuración *Single-ended* (Figura 29-b).

Una vez establecido lo anterior, es necesario manipular tres (3) señales adicionales que controlan la secuencia de encendido para poder realizar las adquisiciones de señales EEG:

1. *START*: es la instrucción que comienza la conversión de los datos.
2. *RESET* : instrucción que resetea y restaura todos los registros a sus valores predeterminados.

3. \overline{PWDN} (*Power-Down*): al colocarse esta señal en *low* (0 lógico), toda la tarjeta se apaga. Para encender, debe colocarse en el nivel *high* (1 lógico).

Las 3 señales mencionadas previamente inician el dispositivo y posteriormente ya es posible enviar los comandos para adquirir las mediciones, los cuales se especifican previamente en la Figura 20. Así, desde el IDE de Arduino, se elaboró un script donde se inicializan las variables que corresponden a los pines físicos de la interfaz SPI, para posteriormente inicializar los registros del ADS1299 y con ello poder muestrear los datos de interés.

Cuando la comunicación entre el ADS1299 y el Arduino se establece correctamente, desde el computador es posible leer los datos que recibe el Arduino mediante un puerto serial libre, normalmente denominado “COM#”, donde el numeral corresponde a un número preasignado que identifica el puerto. En este punto, existen varias posibilidades para leer los datos desde Python:

- Internamente, desde el IDE (*Integrated Development Environment*) de preferencia, la librería de Python *serial* permite leer datos directamente desde un puerto serial especificado, y posteriormente manipular la información para obtener valores numéricos. Esta opción fue la utilizada para ciertas funciones del sistema, como la visualización en tiempo real de algunos canales EEG.
- Utilizando un software de cliente/emulador de terminal/consola serial como el *PuTTY*, es posible leer los datos recibidos por un puerto serial especificado y posteriormente guardar esa información como un archivo en formato de texto, el cual se puede importar en Python para extraer y manipular la información (de manera similar a como se almacenan e importan las bases de datos para EEG). Esta fue la opción utilizada para varias funciones de la BCI, principalmente durante las pruebas de identificación de comandos (P300).

3.3.2.5 Pruebas de funcionalidad de equipos y mediciones preliminares

A continuación se mostrarán algunas pruebas realizadas tanto con los algoritmos y bases de datos como con las señales adquiridas con el ADS1299, para dar constancia de configuraciones y respuestas básicas que evidencien un funcionamiento correcto.

3.3.2.5.1 Pruebas con bases de datos

Las bases de datos que principalmente se utilizaron fueron las disponibles por Marijin van Vliet [50], debido a que todas se utilizaban en ejemplos y algoritmos basados en Python. Primeramente, la base de datos es la originada de un experimento denominado “The Magic Trick”, el cual buscaba adivinar una carta particular que una persona pensó de nueve (9) posibles. Para esto, se construyó el algoritmo en el IDE de *Spyder*, un entorno de desarrollo para Python, y lo primero que se constató fue el formato y las dimensiones de las variables, que representaban la duración del experimento y los valores de las mediciones.

```
#Librerías
import scipy.io
import matplotlib
from matplotlib.collections import LineCollection
import numpy as np
from scipy.stats import pearsonr

#Se importa la Base de Datos
m = scipy.io.loadmat('tutorial1-01.mat')
print(m.keys())

#Definición de las Cartas
cards = [
    'Ace of spades',
    'Jack of clubs',
    'Queen of hearts',
    'King of diamonds',
    '10 of spaces',
    '3 of clubs',
    '10 of hearts',
    '3 of diamonds',
    'King of spades',
]
```

(a)

	0	1	2	3	4	5
0	6.11959	6.29338	6.45916	6.61769	6.76976	6.91617
1	7.4541	7.66621	7.87249	8.07348	8.26975	8.46192
2	8.19815	8.80678	9.40251	9.98396	10.5499	11.0991
3	8.01045	8.4906	8.96449	9.43139	9.89064	10.3417
4	7.48147	7.55068	7.61378	7.67208	7.72689	7.77961
5	8.18366	8.29307	8.39133	8.47925	8.55776	8.62783
6	6.02809	6.13015	6.22235	6.30527	6.37964	6.44627

(b)

Figura 30. (a) Sección de código que muestra la importación de librerías y definición de las cartas parte del experimento. (b) Visualización del tipo y tamaño de las variables en la base de datos, importadas en *Spyder*.

La definición de los nueve posibles *eventos* (cartas) representa las opciones disponibles, y como se indica en la Figura 30-(b), la base de datos contiene “encapsuladas” las mediciones EEG, que ya están en formato numérico (float64), además de otros elementos como los marcadores que indican en qué momento fue mostrada una carta particular y datos adicionales del experimento. Con esto, fue posible graficar los datos sabiendo de la base de datos que las mediciones fueron tomadas con 7 canales EEG muestreados a 2048 Hz.

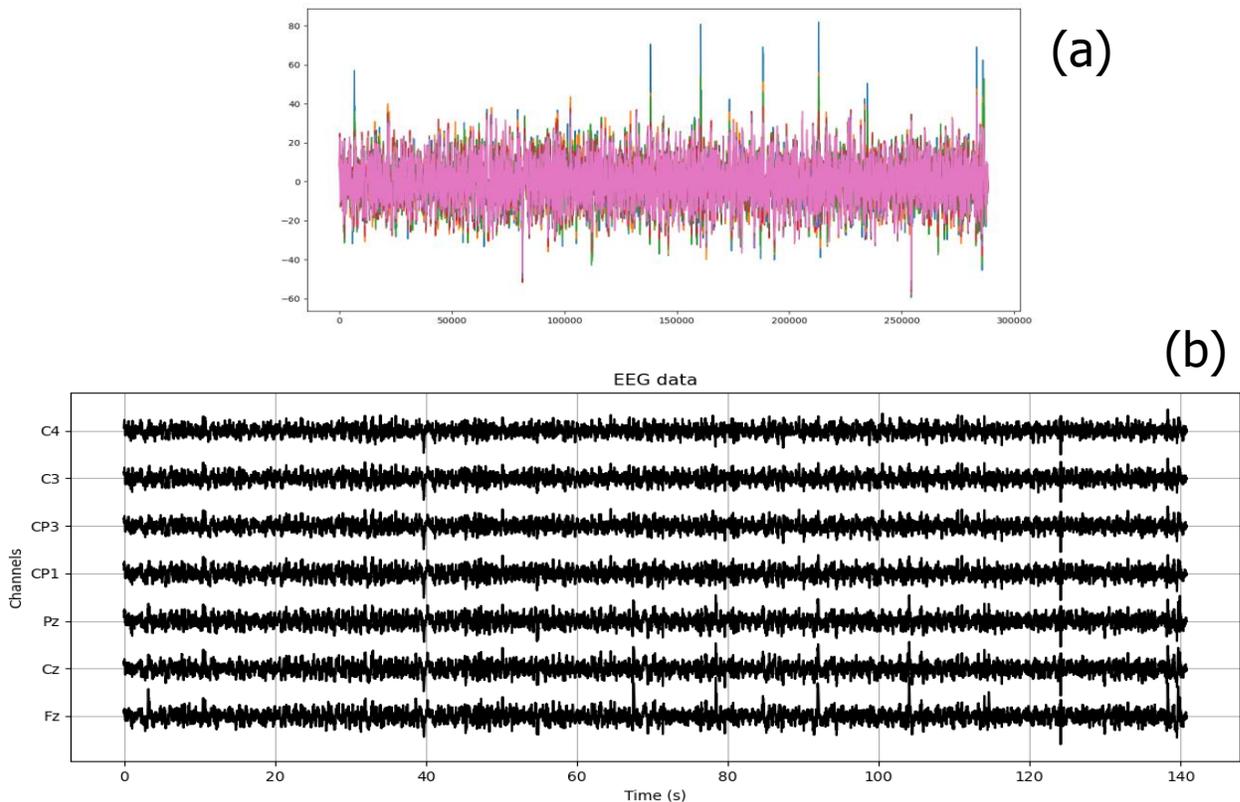


Figura 31. (a) (Imagen superior) Gráfica de las mediciones raw. (b) (Imagen inferior) Gráfica de las mediciones distribuidas en sus respectivos canales, como un EEG.

Originalmente las señales vienen como se muestra en la Figura 31-(a), donde no hay manera de identificar información de interés. Aquí se empieza a evidenciar la importancia del procesamiento y análisis de datos, porque luego de aplicar los filtros necesarios, distribuir las mediciones en sus respectivos canales (según el sistema 10/20) y ajustar las escalas, se obtiene lo mostrado en la Figura 31-(b), claramente más similar a un EEG. Posteriormente, sobre los datos en el dominio del tiempo se realizó la marcación de los *Epochs*, para separar los fragmentos donde se mostraba cada carta, y al promediar todos los epochs asociados a

una carta es posible obtener el ERP para esa carta. De esta manera, se calcularon todos los ERP para cada carta al promediar cada epoch, y se graficaron los ERP para cada canal, definiendo una ventana de un segundo (1s) de duración desde que se muestra una carta particular.

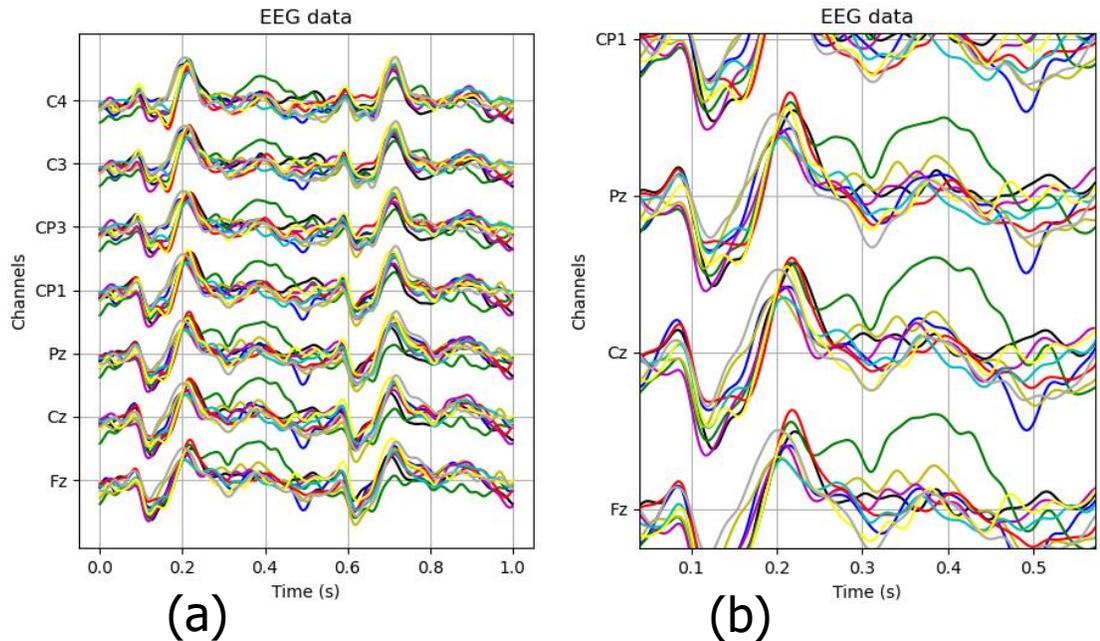


Figura 32. (a) ERP para cada carta, organizados por canal. (b) Zoom de los ERP para los canales Cz y Pz. El ERP para cada carta corresponde a un color diferente (9 colores diferentes).

Al ERP de cada carta se le asignó un color diferente, y el resultado al organizar las respuestas por canales es lo que se observa en la Figura 32-(a). Como es de interés identificar los P300, se añadió la Figura 32-(b) con el zoom sobre los canales Cz y Pz, y se puede observar que cada color, como respuesta a los estímulos de cada carta, tiene un comportamiento diferente (aunque ligeramente similar). Sin embargo, la respuesta de color verde es particularmente mayor que las demás durante el intervalo de tiempo entre 0.3 y 0.4[s], y es esta característica la que precisamente permite identificar la carta escogida por la persona. De hecho, en este caso no es necesaria la implementación de un clasificador o algún algoritmo de *machine learning*, ya que los ERP muestran una buena diferenciación. De esta manera, se asignaron puntuaciones a cada ERP y así se pudo estimar la carta escogida por la persona. En este caso, como se indica en la Figura 33, la carta con la mayor puntuación era

la Reina de Corazones (*Queen of Hearts*), que corresponde al ERP de color verde anteriormente observado.

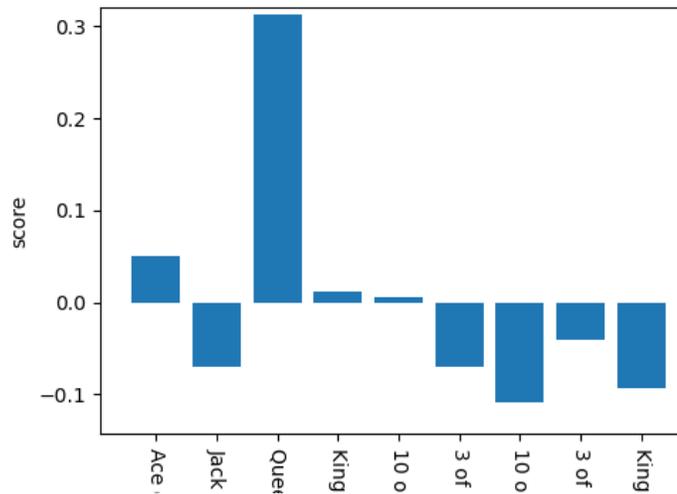


Figura 33. Puntuaciones para cada carta, que se asocian a la probabilidad de que dicha carta haya sido la escogida por la persona.

3.3.2.5.2 Pruebas con señales EEG adquiridas

Las primeras pruebas fueron realizadas con el software que provee Texas Instruments [55]. Mediante su uso, se puede comprobar la comunicación con el ADS1299EEG-FE:

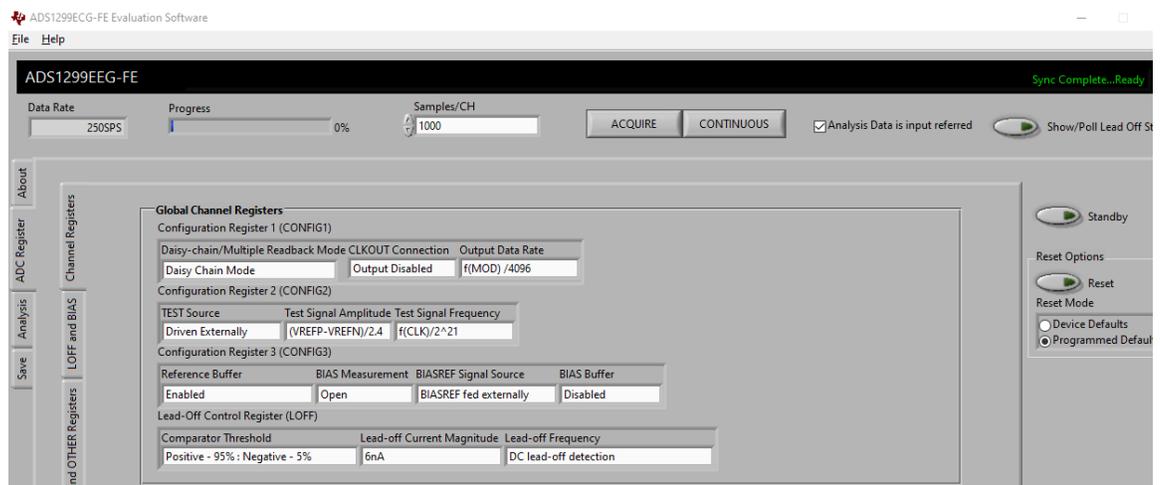


Figura 34. Interfaz del software de evaluación de TI, al conectar la tarjeta. Fuente: el autor.

El reconocimiento de la tarjeta desde el software de TI funcionó sin problemas, y como se evidencia según la Figura 34, al identificar el hardware conectado se habilitan todas las opciones de la interfaz, que permite configurar diversas opciones: tipo de referencia eléctrica, modo de operación del puerto de los electrodos, señales de prueba, ganancia de los PGAs y varias otras opciones. Entre ellas, hay una pestaña llamada “Register Map” que muestra todos los registros internos del ADS1299 y su configuración (en bits), cambiando según los ajustes que se realicen en la interfaz. Asimismo, el software provee de una sección (pestaña “Analysis”) que permite adquirir señales durante un tiempo definido y posteriormente visualizar los datos en el dominio del tiempo o en el dominio frecuencial.

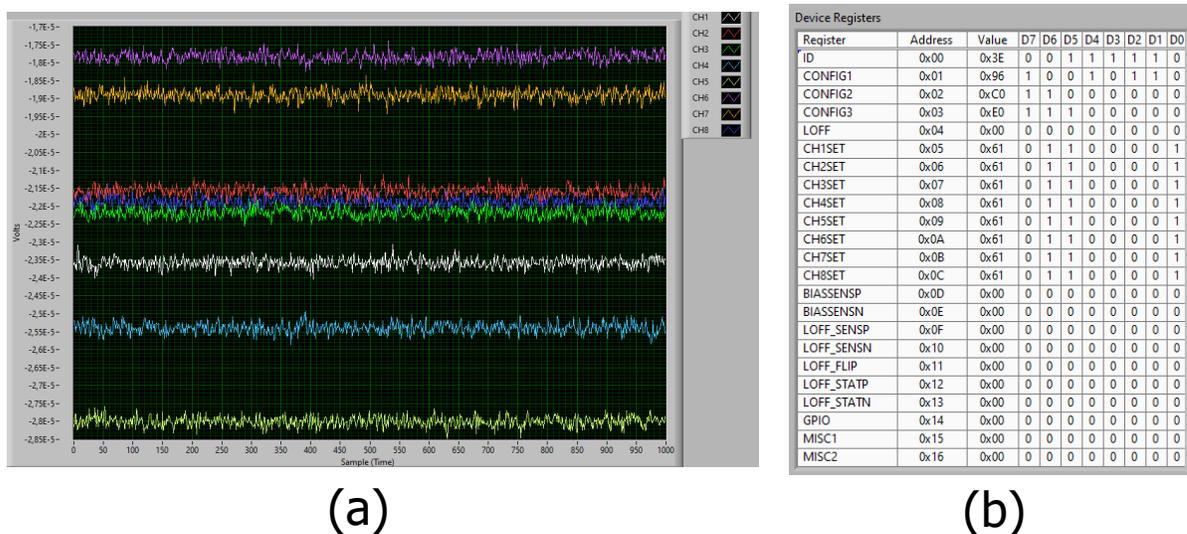


Figura 35. (a) Respuesta del Input-short test. (b) Mapa de los registros para realizar la prueba. Fuente: el autor.

La primera prueba realizada fue el “Input-short test” que, como se indica en el datasheet [55], consiste en desactivar todos los puertos de los electrodos y referenciarlos a una señal de prueba interna de la tarjeta, para verificar el funcionamiento correcto del sistema. Como se observa en la Figura 35-(a), la respuesta de los ocho canales se asemeja a un EEG, lo cual es efectivamente lo deseado, comprobándose que todos los canales de adquisición e internamente los componentes electrónicos de la tarjeta funcionan correctamente. En la Figura 35-(b) se muestra la configuración necesaria para realizar este test.

Las pruebas subsecuentes fueron basadas en el criterio de los “Ojos abiertos-Ojos cerrados”, explicado previamente en el Capítulo 2 – Sección 2.9 del presente documento, buscando captar los cambios súbitos de voltaje al parpadear. Para ello, se configuró la tarjeta para tomar mediciones desde el puerto de los electrodos (J6), y se habilitó únicamente el primer canal (CH1) para adquirir datos. El electrodo de medición se colocó, teniendo en cuenta el sistema 10/20, en la posición Fp2.

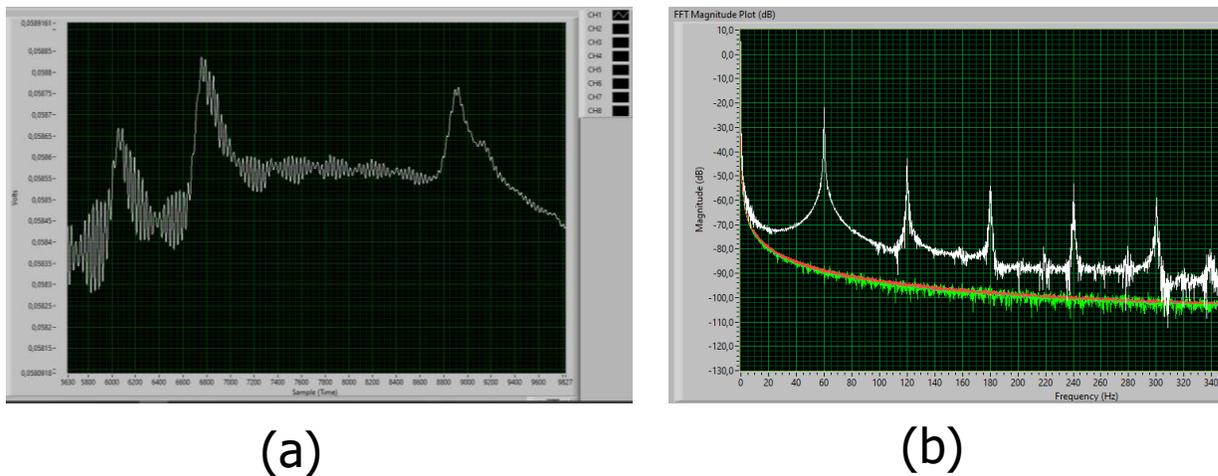


Figura 36. (a) Gráfica en el dominio del tiempo del CH1. (b) FFT de la señal del CH1. Fuente: el autor.

El resultado luego de adquirir los datos es lo mostrado en la Figura 36-(a), donde se alcanzan a apreciar ciertos picos de voltaje, pero con un alto nivel de distorsión. Para verificarlo, dentro del software está la opción de ver en el dominio frecuencial los datos adquiridos, mediante la *Transformada rápida de Fourier (Fast Fourier Transform, de ahora en adelante FFT)*, cuyo resultado es lo que se muestra en la Figura 36-(b). Al detallar el espectro, se aprecian los armónicos del componente fundamental de 60 [Hz] presentes en todo el dominio (continúan después de los 300 [Hz] en donde se recortó la imagen), lo cual justifica la distorsión que afecta las señales adquiridas.

La causa de la distorsión es principalmente otro de los componentes de cuidado para los sistemas EEG: la alimentación eléctrica. En este caso, se adquirieron las señales energizando la tarjeta con un adaptador AC/DC, por lo que los armónicos que se observan son debidos a la red eléctrica. Hay que recordar que como los datos EEG tienen una SNR bastante baja, son

muy susceptibles a las señales externas, y para corregir esa situación se cambió la fuente de alimentación a una completamente DC, basada en iones de litio (batería *Li-ion*).

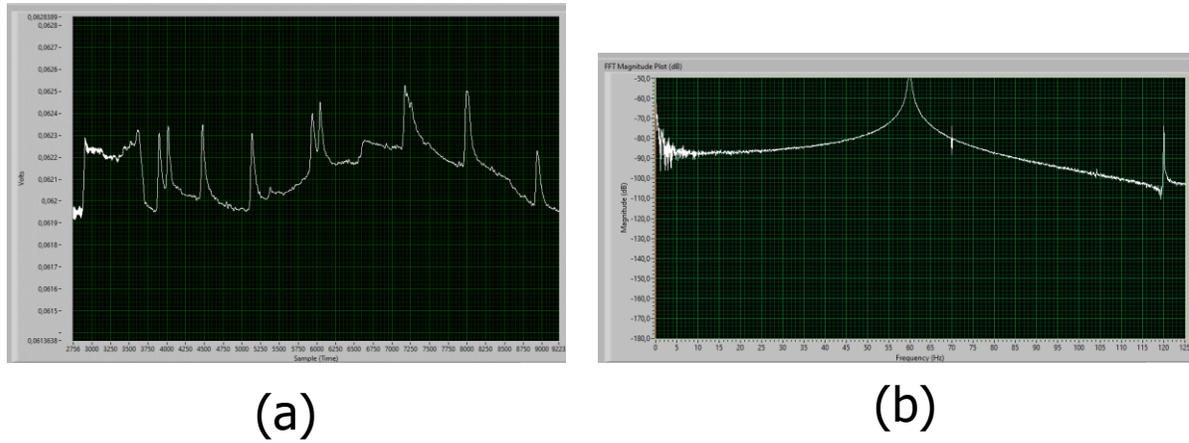


Figura 37. (a) Gráfica en el dominio del tiempo del CHI. (b) FFT de la señal del CHI. Fuente: el autor.

Como se observa en la Figura 37-(a), el resultado mejora considerablemente al punto de que ya son claramente identificables los cambios súbitos de voltaje, que al corroborar con la figura del espectro frecuencial (Figura 37-(b)) se evidencia una reducción de los armónicos que aparecían anteriormente, lo cual se traduce en una señal más limpia. Así, fue posible comprobar la detección de los parpadeos y por ende un correcto funcionamiento en la adquisición de señales EEG.

3.3.3 Interfaz de estímulos visuales en OpenSesame

OpenSesame es un software de código abierto creado totalmente en Python, y funciona como un aplicativo que permite crear experimentos para neurociencia, psicología y economía experimental [62]. El programa fue creado principalmente por Sebastiaan Mathôt, un profesor del departamento de psicología experimental de la Universidad de Groningen, en Holanda (Países Bajos).

Desde su sitio web [62] se puede descargar el programa junto con diversos plugins, manuales y tutoriales para empezar a construir los experimentos. En esencia, el software

funciona como una interfaz interactiva de donde es posible configurar elementos gráficos, duración de eventos, múltiples pantallas y transiciones entre ellas, entre varias otras opciones.

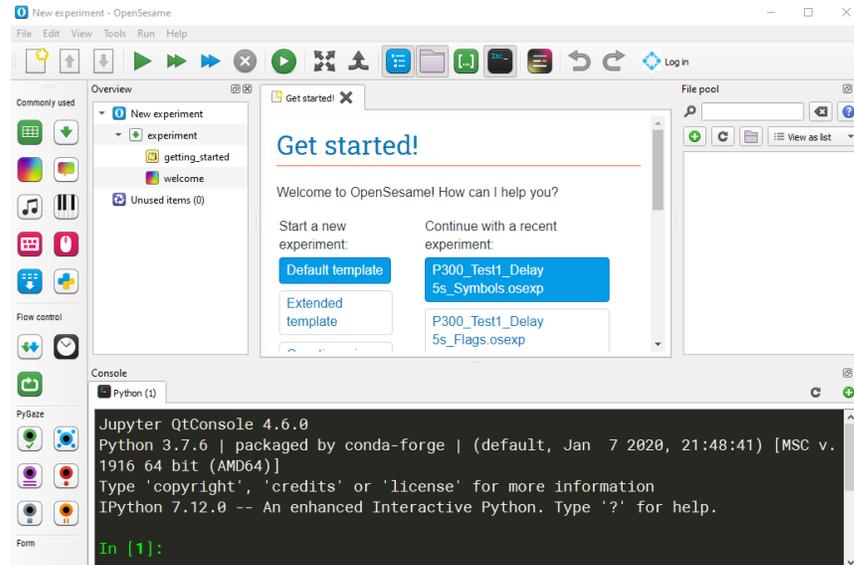


Figura 38. Página principal al abrir OpenSesame. Fuente: el autor.

La interfaz principal es la mostrada en la Figura 38. La barra de herramientas de la izquierda, que contiene varios íconos, representa las funciones y opciones estándar disponibles para la creación de los experimentos orientados a la psicología experimental. Con esto, el objetivo fue construir una interfaz gráfica que se desempeñe de manera similar a la matriz de letras para los *Spellers*, la cual se mostró previamente en la Figura 13 (Capítulo 2 – Sección 2.14), de forma que la interfaz funcione como la fuente generadora de los estímulos visuales que ve la persona.

OpenSesame también acepta dentro de sus opciones secciones de código escrito en Python, lo cual permite configurar el comportamiento de los elementos gráficos. Así, se construyó una interfaz para el prototipo del presente proyecto, la cual consta de dos (2) variaciones utilizadas respectivamente en los dos tipos de pruebas implementadas para el funcionamiento de la BCI: *Flag identification* y *Command identification*.

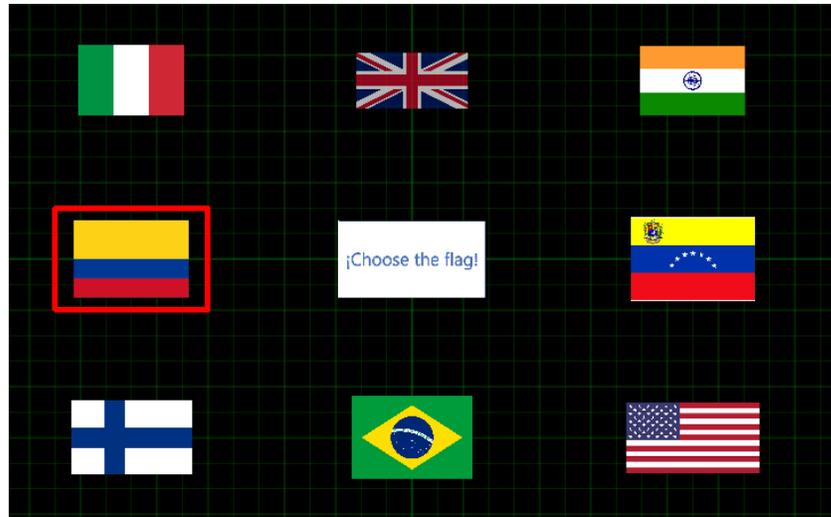


Figura 39. Interfaz de estímulos visuales para las pruebas de Flag identification. Fuente: el autor.

- *Flag Identification*: el objetivo de esta prueba es mostrar en la pantalla una matriz 3x3 con ocho (8) banderas de países distintos, y que la persona seleccione mentalmente alguna de las banderas para posteriormente intentar identificar la bandera escogida. El experimento se configura para durar un tiempo específico, y durante ese tiempo cada bandera se resalta aleatoriamente mediante un cuadro rojo que la rodea durante ciertos instantes, para posteriormente pasar a resaltar otra bandera, y así sucesivamente continúa hasta finalizar el tiempo establecido.

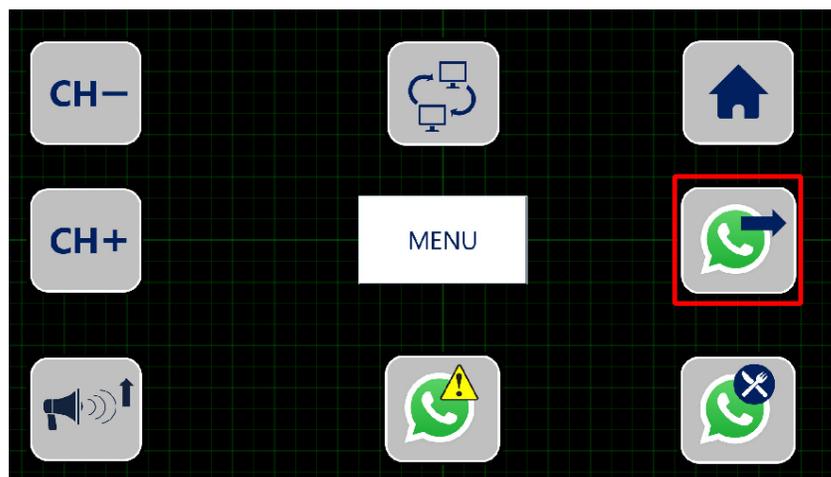


Figura 40. Interfaz de estímulos visuales para las pruebas de Command identification. Fuente: el autor.

- *Command Identification*: de manera similar a la identificación de las banderas, el objetivo de esta prueba es mostrar en la pantalla una matriz 3x3 con ocho (8) símbolos diferentes, que representan un comando específico, y que la persona seleccione mentalmente alguno de los símbolos para posteriormente intentar identificar el comando escogido. Previamente se configura la duración del experimento, y durante éste se resalta aleatoriamente cada símbolo mediante un cuadro rojo (que sólo se mantiene cierto tiempo), lo cual ocurre hasta finalizar el tiempo establecido.

En las Figuras 39 y 40 se muestran ambas interfaces con los estímulos visuales respectivos para cada prueba. En esencia, teniendo en cuenta las descripciones anteriores, ambas interfaces son funcionalmente iguales, ya que la parte de las instrucciones básicas, programación y configuración de los elementos gráficos es la misma, y se basan en el mismo paradigma experimental. La diferencia radica en el tipo de respuesta que, a nivel cerebral, es posible que cambie según el estímulo visual, lo cual se detalla un poco más adelante en el capítulo de resultados y análisis de resultados (Capítulo 4).

Mediante opciones gráficas se configuraron parte de las opciones en OpenSesame: ajustar el tiempo de duración de cada pantalla, el tiempo de duración de cada estímulo (duración del cuadro rojo alrededor de la bandera o el símbolo) y otras opciones, que cuando se completan siguen una secuencia que el programa ejecuta, y se crea un “experimento” como el mostrado en la Figura 41.

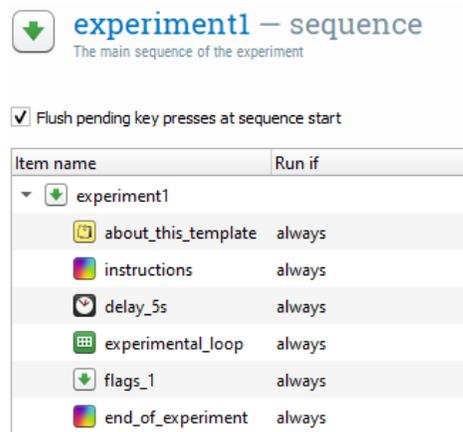


Figura 41. Secuencia de las instrucciones para la interfaz, luego de las configuraciones. Fuente: el autor.

3.3.4 UI principal en KivyMD/Python

La interfaz que maneja funcionalmente las operaciones de procesamiento y comunicación con los dispositivos de prueba es una GUI (*Graphical User Interface*) construida enteramente en Python mediante la librería de Kivy/KivyMD. A grandes rasgos, Kivy en conjunto con su versión alterna KivyMD (MD es el acrónimo de *Material Design*) representan un *framework* de código abierto desarrollado para Python, y especializado en el diseño y elaboración de interfaces gráficas.

Kivy/KivyMD son un paquete instalable en Python y funcionan como una librería que integra diversas opciones personalizables para el diseño y construcción de interfaces. En particular, KivyMD es la versión más actualizada de Kivy, contando con widgets, diseños predefinidos, manejadores de datos y una gran cantidad de herramientas que no sólo facilitan la elaboración de interfaces, sino que optimizan el código necesario para poder construirla. Las ventajas de KivyMD frente a Kivy son principalmente dos:

- Reduce el código necesario para gran parte de las opciones disponibles, logrando el mismo resultado sin comprometer calidad o funcionalidad.
- Ofrece un diseño visual más actualizado, personalizable y atractivo, lo cual mejora sustancialmente el resultado final.

La utilización de Kivy como una librería multiplataforma es además una gran ventaja porque, como valor agregado, es posible migrar fácilmente entre Windows, Linux, macOS, iOS y Android, ofreciendo la posibilidad de abarcar diversas plataformas. Ahora bien, la interfaz desarrollada fue realizada en Windows 10, y fue pensada para que funcionara como una aplicación (*app*) de escritorio. Las opciones disponibles, que aparecen dentro de la ventana principal, se muestran en la siguiente figura:

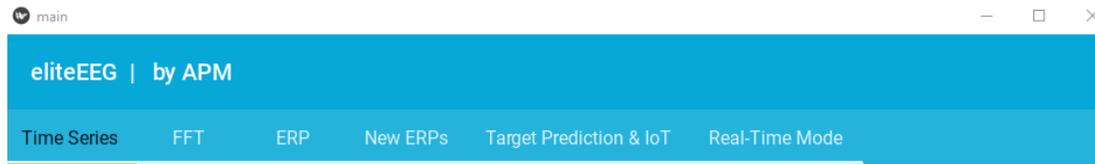


Figura 42. Opciones disponibles en la interfaz, divididas en pestañas (tabs). La pestaña actualmente seleccionada (activa) se resalta con el nombre de color negro y una línea amarilla debajo del nombre. Fuente: el autor.

Como se observa en la Figura 42, la interfaz consta principalmente de un *toolbar* (barra superior) con el nombre de la app, seguido del creador, y en parte inferior se ubican las opciones divididas en pestañas. La interfaz fue hecha completamente en inglés, y todas las opciones son navegables, por lo que el usuario puede simplemente hacer click sobre la pestaña de interés y aparecerán en la pantalla las distintas operaciones que se pueden hacer para cada opción particular. Cada pestaña permite realizar distintas operaciones, que se explican a continuación:

- *Time Series*: se enfoca en el análisis sobre el dominio del tiempo. Permite graficar respecto al tiempo ambos canales de medición (Cz y Pz), y además permite graficar un tercer canal auxiliar, para verificar por ejemplo las pruebas de Ojos abiertos-ojos cerrados, que usan normalmente el electrodo en Fp2. Las gráficas pueden ser simultáneas o separadas, y se les pueden aplicar dos filtros preestablecidos.
- *FFT*: se enfoca en el análisis sobre el dominio frecuencial. Muy similar a la pestaña de Time Series, permite graficar respecto a la frecuencia ambos canales de medición (Cz y Pz) y también el tercer canal auxiliar, de manera individual o simultánea.
- *ERP*: permite calcular y graficar los ERP para los canales de medición Cz y Pz, permitiendo aplicar dos tipos de filtros preestablecidos. Una vez graficados los datos, aparecen dos sliders que permiten navegar sobre la gráfica, para acercarse o alejarse de secciones particulares de las mediciones.

- *New ERPs*: funcionalmente opera igual que la pestaña ERP anterior, pero está asociada a varios algoritmos y filtros diferentes, como parte de las etapas de análisis y procesamiento de datos.
- *Target Prediction & IoT*: calcula y estima el objetivo/target/comando seleccionado por la persona, y permite con ello ejecutar la acción asociada al comando identificado, mediante el envío de la señal de activación hacia el dispositivo correspondiente de la red IoT.
- *Real-Time Mode*: permite conectarse con el ADS1299 y visualizar en tiempo real las señales EEG de un canal específico. Mediante un botón se establece la comunicación por el puerto serial configurado, y otro botón detiene la comunicación cuando el usuario lo requiera.

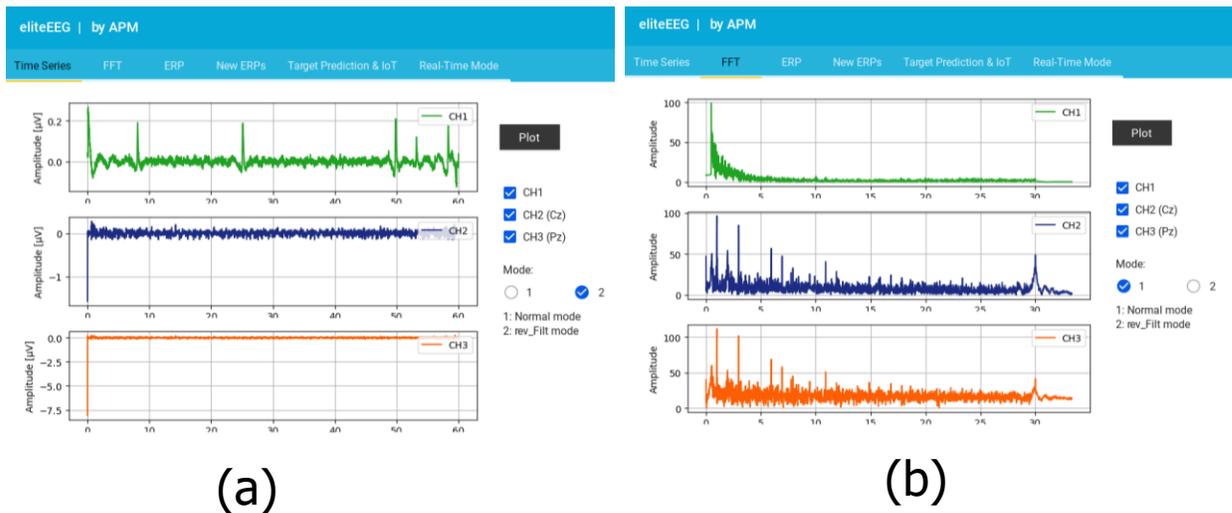


Figura 43. Ejemplos de gráficas realizadas en la interfaz con datos EEG adquiridos. (a) Gráficas en el dominio del tiempo. (b) Gráficas en el dominio frecuencial. Fuente: el autor.

3.3.5 Procesamiento de señales

Una vez adquiridas las señales EEG, fue necesario realizar una serie de etapas que permitieron extraer la información de interés. Para ello, es menester clarificar los dos posibles modos de adquisición de señales que se configuraron para el prototipo:

- **Modo en Tiempo-real (*Real-time Mode*):** en esta configuración, el objetivo es visualizar en pantalla un canal del ADS1299 en tiempo real, en el dominio del tiempo. Se usó principalmente para hacer las pruebas de Ojos abiertos-ojos cerrados, por lo que la prioridad en este caso es la velocidad en el procesamiento de los datos. Para este modo, la lectura de los datos se hace directamente desde Python, y se construyó un Script en el IDE de Arduino que funciona específicamente para esta configuración.
- **Identificación de comandos/banderas (*Command/Flag identification*):** para este modo, el objetivo es adquirir y registrar las señales EEG del usuario como respuesta a la interfaz de estímulos, que puede ser la de las banderas o los comandos. Para ello, se construyó un Script de Arduino diferente del usado para el modo en tiempo real, y en este caso los datos del ADS1299 se registran en un archivo de texto que se importa en Python, para realizar las operaciones de análisis y procesamiento.

Los pasos y procedimientos utilizados son descritos en las secciones siguientes.

3.3.5.1 Preprocesamiento y filtrado

Los datos que se obtuvieron de los electrodos se enviaron por SPI al Arduino, y posteriormente mediante el puerto serial asignado se recibieron las mediciones en el computador, para ser procesadas en Python. En ese punto, los datos estaban en formato binario, y fue necesario convertirlos a un valor numérico que representara el voltaje medido como diferencia de potencial eléctrico entre el punto de medición del electrodo y punto

configurado como referencia eléctrica. Para eso, se utilizó como base la ecuación que sugiere el datasheet del ADS1299 [54], y con algunos ajustes, fue posible convertir los datos a valores de voltajes:

$$voltage [V] = \left(\frac{2 * V_{REF}}{G} \right) * \frac{1}{2^{24} - 1} \quad (1)$$

Donde:

V_{REF} : Voltaje de referencia

G : Ganancia del PGA

El voltaje de referencia utilizado es de 5 [V], correspondiente al voltaje nominal de alimentación, y la ganancia configurada para los PGAs fue de 24. Con eso, cada uno de los datos binarios se convirtieron a un valor de voltaje que represente la actividad eléctrica cerebral. Posteriormente fue necesario organizar y estructurar las mediciones, ya que originalmente vienen como una secuencia de datos *raw*, es decir, sin ningún formato particular.

La idea fue organizar los datos siguiendo una estructura matricial de referencia:

$$Datos EEG raw [float] = \begin{matrix} Canal 1 \\ Canal 2 \\ \vdots \\ Canal 8 \end{matrix} \begin{bmatrix} dato_{11} & dato_{12} & \dots & dato_{1n} \\ dato_{21} & dato_{22} & \dots & dato_{2n} \\ \vdots & \vdots & \dots & \vdots \\ dato_{81} & dato_{82} & \dots & dato_{8n} \end{bmatrix} \quad (2)$$

La ecuación (2) muestra la estructura de referencia según la cual se organizaron las mediciones en Python. Dado que el ADS1299 cuenta con 8 canales de medición de manera simultánea, los datos que se transmiten y se reciben por el puerto serial siguen cierto orden, ya que cada vez que se muestrea el ADS1299 adquiere 8 valores en paralelo, es decir, a pesar de que se pueden configurar los registros internos para que electrónicamente sólo estén activos 3 canales para la medición (el caso presente), el ADC siempre adquiere 8 valores de manera simultánea en cada ciclo de muestreo. Por lo anterior, a la hora de recibir y organizar los datos fue necesario seguir una estructura como la presentada en la ecuación (2).

Como los canales de medición utilizados sólo fueron Cz y Pz, además de un canal auxiliar para ciertos escenarios, la estructura de los datos de interés pasó a ser la siguiente:

$$\text{Datos EEG [float]} = \begin{matrix} \text{Canal 1} \\ \text{Canal 2} \\ \text{Canal 3} \end{matrix} \begin{bmatrix} \text{dato}_{11} & \text{dato}_{12} & \dots & \text{dato}_{1n} \\ \text{dato}_{21} & \text{dato}_{22} & \dots & \text{dato}_{2n} \\ \text{dato}_{31} & \text{dato}_{32} & \dots & \text{dato}_{3n} \end{bmatrix} \quad (3)$$

Una vez ya organizadas las mediciones EEG, fue posible empezar a realizar las operaciones de filtrado. Para eso, se explicará cómo se escogió la tasa de muestreo: según las referencias e investigaciones de neurociencia, las señales EEG para P300 están en una banda frecuencial que no supera los 50 [Hz], y teniendo en cuenta el Teorema de Nyquist-Shannon:

$$F_s > 2 * F_{max} \quad (4)$$

Donde:

F_s : Frecuencia de muestreo

F_{max} : Frecuencia máxima de la señal muestreada

Entonces, según la ecuación (4), la frecuencia de muestreo debe ser mínimo del doble de la frecuencia máxima de la señal de interés para que la información de la señal analógica muestreada se pueda recuperar totalmente. Pero además, lo recomendable para obtener mejores resultados es muestrear con una frecuencia de al menos cuatro (4) veces la frecuencia máxima de la señal de interés. Así, la tasa de muestreo escogida fue de:

$$F_s = 250 \text{ [SPS]} \quad (5)$$

La primera etapa de filtrado fue un filtro pasa-bajos (llamado de ahora en adelante LPF) que limitara las frecuencias para las señales EEG. Como se mencionó, la banda frecuencial de interés no supera los 50 [Hz], pero en los trabajos sobre P300 se recomienda limitar los filtros hasta 30 [Hz], para que los componentes de la banda Gamma no interfieran en las mediciones.

En Python, la librería *scipy.signal* fue la utilizada para implementar filtros digitales, ya que permite definir filtros Butterworth del orden que se requiera, y de manera sencilla aplicarlos sobre los datos. Con eso, se definió el filtro LPF, con los siguientes parámetros:

- F_s : 250 SPS.
- Frecuencia de corte del filtro (llamada *cutoff*): 30 [Hz]
- Orden del filtro: 20.

Una vez aplicado el LPF, posteriormente se aplicó un filtro pasa-altos (llamado de ahora en adelante HPF) con frecuencia de corte en 0.5 [Hz], porque las frecuencias menores no aportan información de interés y muchas veces sólo incluyen señales de ruido. De esta manera, frecuentemente ya se tenían las mediciones en el espectro EEG.

Algo que cabe aclarar es que, al utilizar alimentación eléctrica DC, no es tan necesario aplicar el filtro Notch (rechaza-banda) en 60 [Hz], además de que la aplicación de los filtros HPF y LPF anteriores cumple indirectamente esta función. Asimismo, es preferible aplicar los filtros HPF y LPF en secuencia, y no definir un filtro pasa-banda, para optimizar tiempo de procesamiento computacional.

Ahora bien, una vez que se ubicaron los datos en la banda EEG, ya fue posible visualizarlos en el dominio del tiempo. Como la estructura de los datos es matricial, las gráficas necesitaban crear un vector de tiempo con las mismas dimensiones que las mediciones para un canal, y de esa manera se graficaron los datos. El resultado es como el mostrado en la Figura 44, que fue tomada durante una prueba de Ojos abiertos-ojos cerrados. Cada una de las formas pseudo-triangulares o gaussianas que se observan corresponden a un parpadeo, y si se detallan las amplitudes de voltaje, se puede apreciar que la diferencia de voltaje entre el pico máximo y la base de la señal es de unos 200 [μ V] aproximadamente, lo cual concuerda con los valores encontrados en pruebas de este tipo según las referencias consultadas.

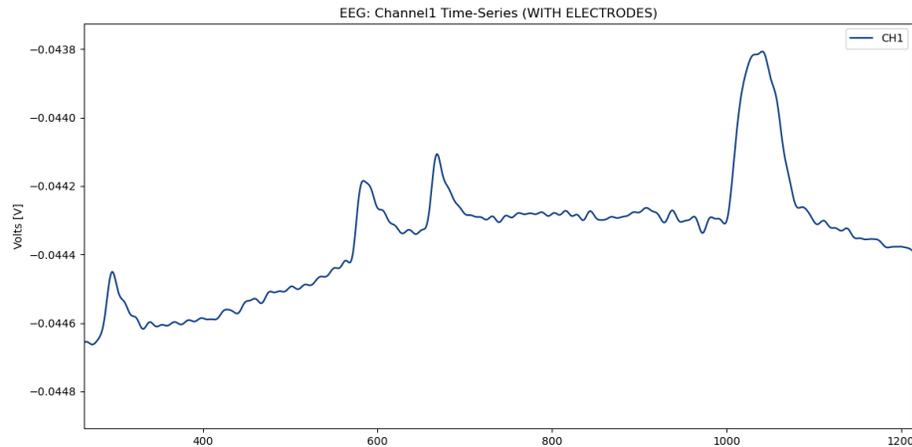


Figura 44. Gráfica en el tiempo de una prueba de Ojos abiertos-ojos cerrados, vista desde Python mediante la configuración del modo en tiempo real. Fuente: el autor.

El procesamiento de los datos hasta este punto es el mismo para ambos modos de operación (*Real-time* y *Command/Flag identification*). De hecho, para el modo *Real-time* fue prácticamente todo lo necesario para poder ver los datos, aunque posteriormente se incluyen algunos algoritmos adicionales que buscan controlar el cambio dinámico de los datos en pantalla, para mejorar la fluidez de las mediciones vistas desde la interfaz gráfica. Entonces, a partir de este punto, todas las operaciones se enfocaron en el modo *Command/Flag identification*, para poder precisamente identificar los comandos escogidos por la persona.

Con los datos adquiridos ya en el dominio temporal, se agregaron algunos filtros adicionales que, según la literatura [50], [63], permiten eliminar varios artefactos que aparecen en los registros EEG. Específicamente, comportamientos similares a un ECG, y componentes (picos de voltajes) EOG y EMG empiezan a visualizarse debido a movimientos involuntarios, ansiedad, hambre y distractores adicionales. Para ayudar a corregir eso, se introdujeron varios métodos:

- La atenuación de señales cardíacas (ECG) se realizó mediante un filtrado según la forma de onda, ya que frecuentemente parte del espectro EEG se mezcla con el ECG (0-120 [Hz], aproximadamente). Es decir, los filtros implementados consisten en secciones de código iterativo que tratan de identificar formas de

onda similares a un PQRST (la morfología típica en señales ECG), para remover o atenuar esas partes, siendo que los filtros frecuenciales no ayudan tanto en estos casos.

- Los artefactos debidos a movimientos oculares (EOG) o señales musculares (EMG) son más complicados, y su eliminación sigue siendo materia de investigación [63]. Algunos estudios sugieren, por ejemplo, aplicar un filtro Notch en aproximadamente 12 [Hz] para atenuar componentes de parpadeos y respuestas musculares involuntarias, lo cual fue uno de los filtros adicionales implementados.

3.3.5.2 Cálculo del ERP

Una vez que se obtienen los datos estructuralmente ordenados y filtrados, se procedió a calcular los ERP asociados a los eventos de la interfaz visual. En ese sentido, teniendo en cuenta lo explicado previamente en la sección 3.3.3 (interfaz de los estímulos visuales), en OpenSesame se construyó el experimento con las siguientes propiedades principales:

- Duración del experimento: 60 [s].
- Duración de cada estímulo (cuadro rojo alrededor de una opción): 1 [s].
- Cantidad de opciones totales: 8.
- Número de estímulos visuales por sesión: 36 (orden aleatorio).

Como se recuerda, la interfaz de estímulos visuales es una matriz 3x3 con 8 opciones para escoger (Figuras 38 y 39). Durante la ejecución del experimento desde OpenSesame, al usuario se le muestra una pantalla inicial sin opciones que dura 5 [s] e indica que se prepare para escoger una opción/bandera. Luego de ese tiempo, se muestra la matriz con las banderas/comandos y aleatoriamente un cuadro rojo va resaltando cada opción durante 1 [s] hasta finalizar el tiempo configurado, y finalmente se muestra una pantalla que indica que la sesión del experimento terminó.

Adicional a lo anterior, previamente se configura el ADS1299 y el Arduino para adquisición de señales en el modo *Command/Flag Identification*, de manera que mientras el usuario ve en la pantalla los estímulos visuales, se están adquiriendo y registrando las señales EEG. Una vez finalizado el experimento desde OpenSesame, los registros se guardan en un archivo de texto (como una base de datos) para posteriormente importarlo en Python.

Ahora bien, para poder calcular los ERP, además de los datos EEG importados, son necesarios también los marcadores temporales que indiquen en qué momento se estaba mostrando/resaltando cada opción. Para ello, dentro del experimento de OpenSesame se incluyó un código en Python que al finalizar cada experimento exporta un archivo con los tiempos en los cuales se resaltó cada opción.

Una vez adquiridos los datos EEG junto con los correspondientes tiempos para los estímulos, desde Python se importaron ambos archivos y se realizó un proceso de “sincronización”, que consiste en ajustar los momentos en que fueron mostrados cada estímulo según los datos EEG, mediante las librerías *os* y *datetime*. Así, fue posible graficar sobre el dominio del tiempo los momentos en que se mostraron cada estímulo:

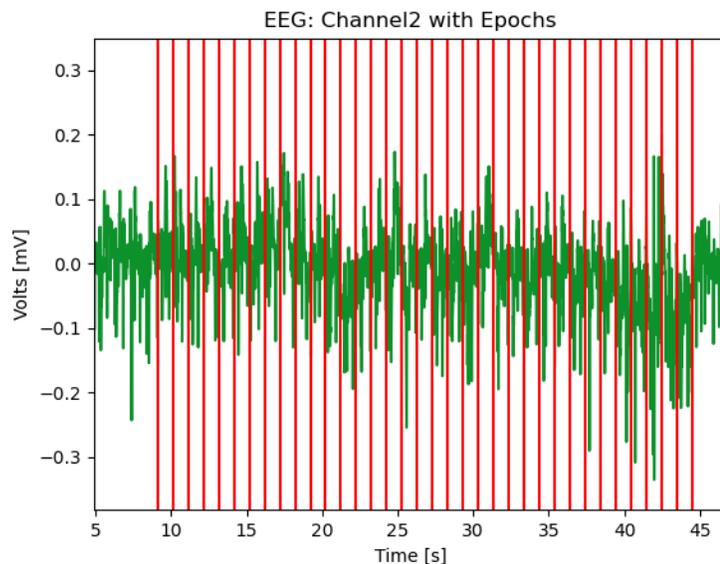


Figura 45. Gráfica en el tiempo de un canal EEG, junto con los marcadores (líneas rojas) que indican el momento en que fue mostrado cada estímulo. Fuente: el autor.

En este punto, como se indica en la Figura 45, los datos EEG son delimitados por los momentos en que se mostró cada estímulo, y cada “fragmento” de datos entre las líneas rojas indicadas es lo que corresponde al llamado *Epoch*. Es decir, un epoch contiene los datos desde que cierto estímulo fue mostrado a la persona, y continúa hasta que el siguiente estímulo se presenta, lo cual genera un epoch diferente al anterior. Así, cada epoch se “asocia” al estímulo mostrado que, según la interfaz de OpenSesame, representa la bandera/comando resaltado en cierto momento.

Con las ideas anteriores queda claro entonces que cada epoch está asociado a un estímulo presentado, y según la configuración del experimento, se indica que cada epoch es un fragmento de datos de 1 segundo (tiempo entre cada estímulo). Entonces, el ERP se calcula promediando todos los epochs asociados a un mismo estímulo:

$$ERP_n = \frac{\sum_{i=1}^m epoch_i}{m} \quad (6)$$

Donde:

n: *identificador del estímulo (de los 8 posibles)*

m: *número de epochs asociados a un estímulo*

Como cada epoch contiene la actividad cerebral que ocurre luego de que se presenta cierto estímulo visual, el promedio de todos los epochs asociados a un mismo estímulo origina el ERP asociado a ese mismo estímulo. Por ende, como existen 8 posibles estímulos visuales, de cada sesión de un experimento se pueden calcular ocho (8) ERPs, relacionados a cada una de las opciones posibles.

Desde Python, el hecho de saber en qué momento específico fue mostrado cada estímulo permitió separar y organizar los datos según el estímulo asociado, para posteriormente realizar el promedio de cada uno de los epochs, y así se calcularon los ERP para cada estímulo. El resultado, es como el presentado en la siguiente figura:

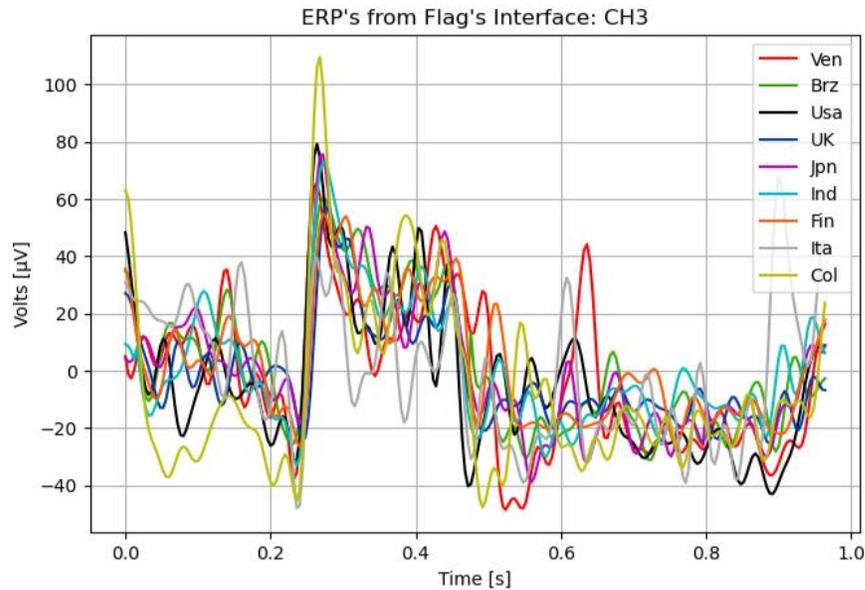


Figura 46. Gráfica de los ERPs para un experimento realizado con la interfaz de las banderas. Cada ERP está representado con un color diferente, identificado en la parte superior derecha. Fuente: el autor.

Como se observa en la Figura 46, cada ERP se identificó con un color único, y todas las señales se extienden durante 1 [s]. A modo de relacionar experimentos, se invita al lector a volver a revisar la Figura 32, de la sección 3.3.2.5.1 – *Pruebas con bases de datos*, para evidenciar la similitud entre los ERPs calculados y los ERPs de la base de datos tomada como referencia.

3.3.5.3 Clasificación e identificación de comandos

A partir de la obtención de los ERP, se entró en la fase final del procesamiento, cuyo objetivo fue la identificación de los comandos/target seleccionados por la persona. Dicha tarea implica normalmente analizar los ERP y extraer las características que sean acordes al componente P300, mediante el uso de diversas técnicas posibles. No obstante, es importante tener en cuenta dos factores clave:

- Si bien es cierto que un experimento que sigue el paradigma Oddball genera una señal P300 asociada a un estímulo infrecuente, no hay una garantía total de que

el ERP con la amplitud máxima que se registre según la actividad EEG sea estrictamente el asociado al objetivo escogido por la persona. Es decir, puede que el ERP que contenga el componente P300 de mayor amplitud no sea el asociado al target, y dicha situación puede deberse a múltiples factores: distracciones, falta de entrenamiento en la persona, mejoramiento de los algoritmos de identificación, entre otros.

- Los procedimientos y técnicas para la detección de un P300 no son estándar o restrictivos, sino que por el contrario, al ser fenómenos de origen experimental, la gran mayoría de autores e investigadores prueba con algoritmos y técnicas propias para verificar las posibilidades en la detección.

Con las ideas anteriores, desde Python se procedió a realizar la construcción de un algoritmo clasificador, que contó con las siguientes etapas:

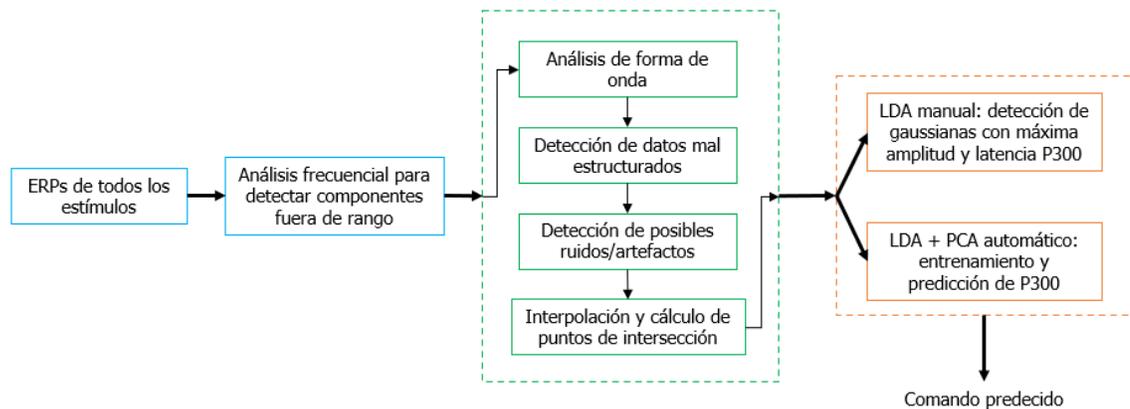


Figura 47. Diagrama funcional del clasificador construido. El orden de lectura y ejecución del algoritmo es de izquierda a derecha.

Fuente: el autor.

Cada una de las partes construidas se explican a continuación, de manera secuencial:

1. Los datos de los ERP para cada estímulo se organizan como una estructura matricial, de manera similar a la ecuación (2), para ser analizados de forma iterativa.
2. Se aplica la FFT a cada ERP, para detectar componentes frecuenciales con amplitudes mucho mayores que la media.
3. Se realiza una análisis de forma de onda para cada ERP. El objetivo es identificar picos de voltaje fuera de rango, oscilaciones y posibles ruidos que distorsionen el ERP.
4. Dependiendo de la forma de onda, el algoritmo intenta corregir la estructura del ERP y detectar errores. Para ciertos casos, fue necesario hacer una interpolación digital (agregar más muestras de las originales) para aumentar la exactitud del análisis. En otros escenarios, se implementó una rutina que identifica puntos de corte entre ERPs cuyos datos parecen afectados por artefactos o señales no identificadas previamente. Los ERPs corregidos y reestructurados se renombran como “new ERPs”.
5. Desde la interfaz gráfica, el usuario puede escoger entre dos alternativas de detección del objetivo, que se basan en algoritmos diferentes: la primera opción intenta aplicar el LDA de manera progresiva, al buscar identificar una señal de tipo gaussiana con latencia promedio y amplitud mayor que el resto de los ERPs, lo cual identifica como un P300; la segunda opción, se basa en el entrenamiento y posterior aplicación de un algoritmo que asocia los objetivos con cada ERP e intenta predecir el target mediante la búsqueda automática de una señal con características de un P300.
6. Para cualquiera de los algoritmos escogidos, el sistema asigna una probabilidad a cada opción posible, y el ERP con el componente P300 se define como la

opción más probable. La opción asociada al ERP más probable se define como el comando identificado, y se exporta como el resultado del clasificador.

En esencia, una parte del clasificador se enfoca principalmente en intentar corregir posibles errores, ruidos, artefactos y demás comportamientos no deseados porque, a la hora de analizar datos EEG reales, muchas veces no era posible aplicar criterios para identificar el objetivo o directamente alguna técnica de clasificación por la información errática presente.

Ahora bien, en el caso particular del algoritmo de aprendizaje automático o *machine learning*, es importante diferenciar dos conceptos clave:

- *Supervised Learning* (aprendizaje supervisado): aplica para los algoritmos que utilizan datos previamente marcados/etiquetados, los cuales se diseñan para entrenar el algoritmo y poder predecir resultados de manera precisa. Es un caso típico en sistemas de clasificación para asignar datos a categorías específicas [64].
- *Unsupervised Learning* (aprendizaje sin supervisión): los algoritmos de este tipo analizan y agrupa datos sin estructuras o etiquetas particulares. Es decir, descubren patrones escondidos en los datos sin la necesidad de intervención humana [64].

El algoritmo utilizado del LDA es entonces del tipo *Supervised Learning*, ya que requiere de una serie de datos para entrenarlo y posteriormente aplicarlo. Funciona de la siguiente manera: se escogen una serie de datos EEG previamente estructurados que corresponden a mediciones, y se asocian a una variable X . Posteriormente, cada uno de los datos de X se asocia a una “clase”, que en este caso representa el símbolo/bandera objetivo, y se crea un vector de clases Y . En conjunto, X e Y son entonces los datos que permiten “entrenar” el algoritmo, es decir, indicarle que cuando se presenta una entrada “similar” a los valores de X , la respuesta o salida del sistema es el valor asignado en Y .

Desde Python, mediante la librería *sklearn.discriminant_analysis* se puede importar el objeto *LinearDiscriminantAnalysis*, que es precisamente el modelo base para clasificar datos. Con eso, fue posible definir las siguientes instrucciones:

$$clf = LinearDiscriminantAnalysis() \quad (a)$$
$$clf.fit(X,Y) \quad (b)$$

Mediante las instrucciones (a) y (b) se creó un objeto clasificador que se entrena con los datos previamente estructurados y escogidos. Posteriormente, el sistema ya puede aceptar nuevos datos, y utilizar métodos adicionales:

$$clf.predict([new_sample]) \quad (c)$$

Usando la instrucción (c), el algoritmo predice la clase a la que pertenecen los nuevos datos, que en este caso, es el símbolo/bandera objetivo según los datos de entrada. El clasificador se basa en el Teorema de Bayes para hacer la predicción del target, además de que asume que los datos se distribuyen de manera gaussiana.

3.3.6 Red IoT

La salida del clasificador es el comando identificado, y para su ejecución, se toma como referencia la Figura 47 para explicar cada uno de los posibles comandos:

1. Se envía un mensaje preconfigurado (texto libre) por WhatsApp al número guardado en el directorio.
2. Mensaje preconfigurado que se envía por WhatsApp. Este mensaje está destinado para solicitar bebidas o alimentos.

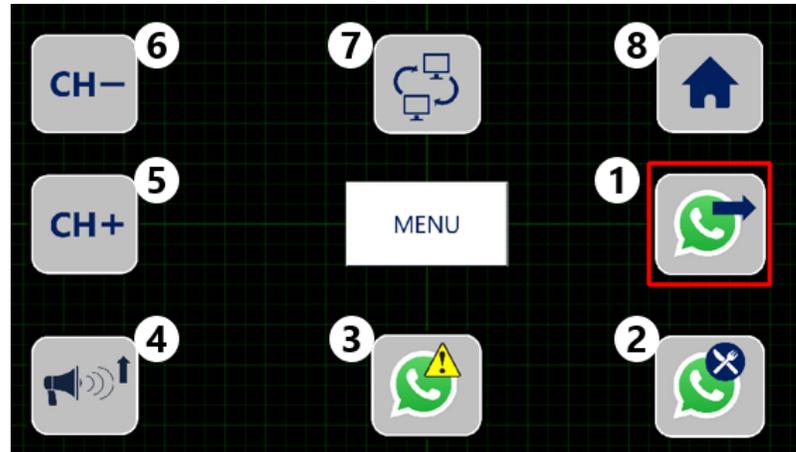


Figura 48. Opciones enumeradas de la interfaz de estímulos visuales. Fuente: el autor.

3. Mensaje preconfigurado que se envía por WhatsApp. Este mensaje está destinado para solicitar asistencia médica o notificar algún evento particular.
4. Sube el volumen del Smart TV de la habitación.
5. En el modo de TV por cable (o antena), se mueve al siguiente canal.
6. En el modo de TV por cable (o antena), se mueve al canal anterior.
7. Lanza la app configurada (Netflix por defecto).
8. Botón de “Home”, que devuelve el Smart TV al modo de canales por cable (o antena).

Los dispositivos conectados para poder ejecutar los comandos son dos (2): un smartphone basado en Android, y un SmartTV con *webOS* como sistema operativo (marca LG). Para el caso del SmartTV, la comunicación se realizó mediante la librería *pywebos* de Python, la cual permite crear un objeto de tipo cliente y asociarlo al SmartTV, además de que mediante una “*client_key*” el sistema se asegura que la información llega a un dispositivo previamente registrado y catalogado como seguro. Para la conexión, desde Python se registró

la dirección IP local del SmartTV, y posteriormente se enviaron los comandos para su ejecución.

Para el caso de los mensajes por WhatsApp, se utilizó una plataforma web llamada *Twilio*, que funciona como una API (*application programming interface*) que ofrece servicios web principalmente orientados a la comunicación, analítica de datos y operaciones en la nube. Mediante Twilio, es posible conectarse con un servidor web que permite establecer un enlace de comunicación con un teléfono registrado en la plataforma, y con ello enviar mensajes completamente personalizados. De hecho, la plataforma web posee una opción para que funcione como un *Sandbox*, es decir, un entorno de desarrollo controlado para hacer pruebas de comunicación (mensajería) de forma sencilla.



Figura 49. Diagrama ilustrativo con algunas de las opciones disponibles desde Twilio. Imagen extraída de [65].

Desde Python, se utilizó la librería *twilio.rest*, la cual permite crear un objeto de tipo “Cliente” para conectarse con los servicios web. Para hacer esto, previamente hay que registrarse en la página web, y mediante la generación de un código de autenticación que es único para cada usuario, desde Python se establece una comunicación segura.

Finalmente, el algoritmo en Python funciona como un “Menú de Opciones”, en el cual se registraron todos los comandos posibles y cada una de las operaciones. Al recibir el comando identificado, en la rutina se asigna un identificador para cada opción y se valida que el comando está dentro de las opciones. Entonces, se direcciona el comando con el

dispositivo necesario, y se establece la comunicación bien sea con el cliente de Twilio o hacia la red en donde está conectado el SmartTV, y se ejecuta el comando.

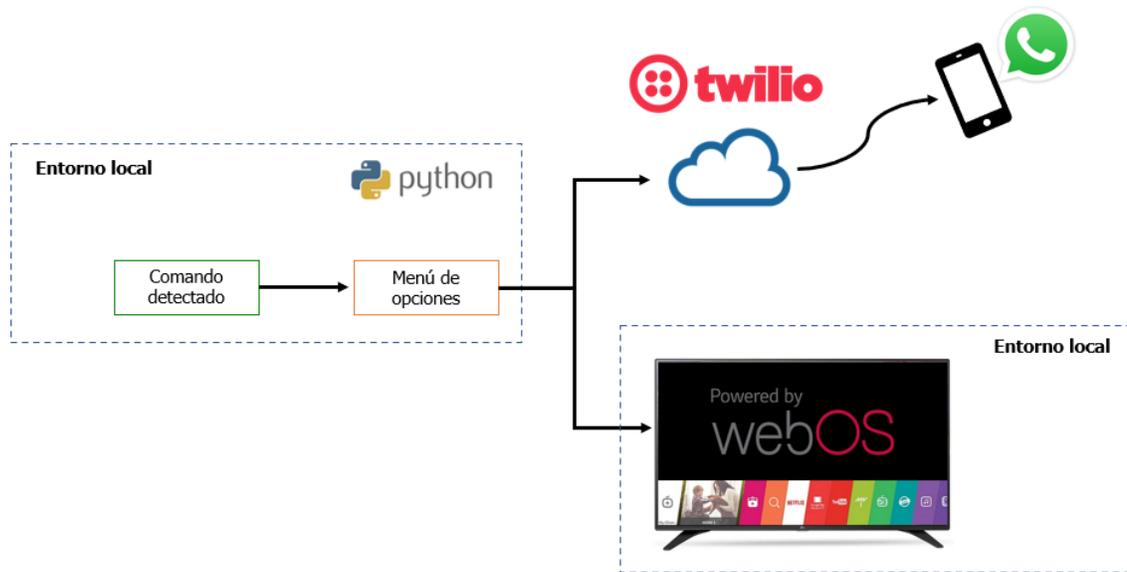


Figura 50. Diagrama de la red IoT construida para el prototipo. Fuente: el autor.

4 RESULTADOS Y ANÁLISIS DE RESULTADOS

Las pruebas realizadas con el sistema se basaron principalmente en los datos EEG adquiridos mediante la interfaz de estímulos visuales (ver Figuras 39 y 40). Como se explicó, funcionalmente los experimentos tanto para la identificación de banderas como para la identificación de los símbolos son iguales, pero particularmente la interfaz de las banderas sirvió en gran medida para entrenar cerebralmente, de manera que el cuerpo se adaptara a las sesiones de experimentos Oddball. Adicionalmente, se utilizaron ambas interfaces para probar las respuestas en la identificación de los comandos y las variaciones al cambiar el tipo de estímulo visual.

Las pruebas realizadas se estructuraron de la siguiente manera:

- Se ejecutaron pruebas en 3 sujetos diferentes.
- A cada sujeto se le mostró previamente, antes del experimento, la matriz con las banderas/opciones disponibles, y se le pidió que escogiera mentalmente una de las opciones.
- Para cada sujeto, se realizaron sesiones de 5, 10 o 12 pruebas, es decir, 5, 10 o 12 banderas/símbolos para escoger libremente de las opciones, de manera sucesiva.

Teniendo en cuenta que el algoritmo utilizado ordena los ERP según la probabilidad de que haya sido el target, se creó un indicador que representa el *porcentaje de acierto*, llamado *Accuracy [%]*, el cual se calculó para cada sesión según el siguiente sistema de ponderación:

- Si la bandera/comando identificada/o corresponde con la opción más probable, el programa asigna un 100% de acierto para esa prueba.
- Si la bandera/comando identificada/o corresponde con la segunda opción más probable, el programa asigna un 70% de acierto para esa prueba.
- Si la bandera/comando identificada/o corresponde con la tercera opción más probable, el programa asigna un 50% de acierto para esa prueba.

- Para cualquier otro resultado, no se asigna ningún porcentaje de acierto.
- El porcentaje de acierto de una sesión de varias pruebas se calculó como el promedio de los aciertos de cada prueba individual.

A continuación se muestran los resultados obtenidos en las pruebas realizadas:

Tabla 2. Tasa de aciertos para las sesiones realizadas con las interfaz de las banderas (Flag identification).

Sujeto	Cantidad de sesiones totales	Pruebas por sesión	Accuracy [%]
Sujeto 1	7	12	81.0
Sujeto 2	5	10	66.43
Sujeto 3	3	5	57.22

Tabla 3. Tasa de aciertos para las sesiones realizadas con las interfaz de comandos (Command identification).

Sujeto	Cantidad de sesiones totales	Pruebas por sesión	Accuracy [%]
Sujeto 1	3	12	72.0
Sujeto 2	3	10	62.3076
Sujeto 3	1	5	54.0

Como se aprecia en los expuesto tanto en la Tabla 2 como en la Tabla 3, los aciertos variaron considerablemente entre los sujetos. Analizando los datos, es altamente probable que dicho comportamiento esté ligado al hecho de que el Sujeto 1 fue el que realizó la mayor cantidad de pruebas, lo cual cerebralmente representa el sujeto con la mayor cantidad de tiempo de entrenamiento en los procesos sinápticos. Asimismo, detallando la cantidad de pruebas individuales, estadísticamente también se obtuvieron mejores resultados al aumentar la cantidad de banderas/comandos para identificar. Pero como clara respuesta general, fue posible identificar varios objetivos seleccionados e incluso tener sesiones con un muy buen desempeño, lo cual más allá de todos los factores asociados, demuestra que el funcionamiento

de la BCI de manera global posibilita el cumplimiento de los objetivos inicialmente planteados.

Ahora bien, las diversas pruebas realizadas mostraron comportamientos erráticos al principio, como caso general. Pero a medida que cada uno de los sujetos entrenaba y realizaba más pruebas, los resultados al analizar las señales EEG se hacían más consistentes. En ese sentido, es importante recordar que cuando una persona sin ningún tipo de entrenamiento o acercamiento previo a experimentos Oddball comienza sesiones de adquisición EEG, es biológicamente posible que ciertos componentes ERP e incluso componentes P300 aparezcan al analizar los datos, pero es incierto el hecho de que los potenciales correspondan al estímulo escogido. Por eso, los resultados obtenidos se empezaron a registrar después de algunas sesiones de entrenamiento.

Un aspecto para resaltar fue una curiosa respuesta de todos los sujetos en general: los estímulos visuales de la interfaz de las banderas parecían generar potenciales asociados más fácilmente identificables que los generados por la interfaz de comandos. Este aspecto podría deberse al hecho de que los estímulos de las banderas también eran utilizados en sesiones de entrenamiento, mientras que la interfaz de comandos solamente se utilizó para pruebas específicas.

Adicionalmente, siempre existieron diversos factores que afectaron directamente varias sesiones de toma de señales EEG: el sueño, cansancio, desconcentración, parpadeos involuntarios, señales musculares y otros, los cuales en algunas ocasiones causaron registros EEG difícilmente interpretables. Pero desde el punto de vista de los algoritmos construidos, fue posible identificar y remover en cierta medida los artefactos, para que el proceso de identificación del objetivo seleccionado tuviera la menor cantidad de errores posibles.

Finalmente, la operación de la BCI como resultado de los distintos módulos que componen el prototipo propuesto permitió no sólo analizar los datos EEG, sino extraer información que permitiera identificar comportamientos asociados a los estímulos. Con eso, la transmisión de los comandos a los dispositivos conectados se realizó sin mayores

inconvenientes, ya que desde el algoritmo de Python el menú de opciones disponibles manejaba las distintas comunicaciones, por lo que el control del SmartTV y el envío de mensajes por WhatsApp era efectivo. De hecho, queda claro que las posibilidades del prototipo son muy amplias, debido a que interactuar con algunos elementos del entorno no sólo fue un hecho, sino que potencialmente se pueden escalar los periféricos de la red y el análisis de los datos, para lograr sistemas inteligentes que ayuden a las personas en sus labores del día a día.

5 CONCLUSIONES

- La adquisición de señales EEG mediante el dispositivo electrónico escogido fue no sólo posible, sino verificable mediante diversas pruebas realizadas a los datos obtenidos. Queda en evidencia que la utilización del ADS1299EEG-FE, basado en el ADS1299, permite obtener señales de la actividad eléctrica cerebral.
- El análisis y procesamiento de las señales EEG adquiridas con los dispositivos escogidos permite, mediante distintos algoritmos, calcular y visualizar gráficamente los ERP asociados a los estímulos visuales presentados mediante una interfaz gráfica. Con ello, la clasificación e identificación de los comandos relacionados a cada ERP se logró realizar bajo ciertas condiciones de operación y funcionamiento.
- Mediante la utilización de una interfaz gráfica basada en Python que funcione de acuerdo con los protocolos y librerías adecuadas, es posible establecer un enlace de comunicación para transmitir los comandos identificados a los dispositivos inteligentes configurados, de manera que posteriormente se ejecuten las instrucciones en el entorno local.
- A través del manual de usuario, quedan en evidencia las distintas configuraciones y procesos necesarios que hicieron posible no sólo la ejecución de los comandos en el entorno local, sino el funcionamiento global del prototipo.

6 RECOMENDACIONES

- Desarrollar y fomentar el entrenamiento cerebral como un hábito para investigaciones con señales EEG. Con esto, el objetivo es dejar claro que mientras mayor entrenamiento cerebral tenga una persona, bien sea con el experimento específico que se esté desarrollando o similares dentro del campo de la neurociencia, las posibilidades de obtener resultados más consistentes y adecuados aumenta considerablemente.
- Investigar y documentar en la mayor medida posible los estudios y pruebas previas realizadas con algún equipo que se desee adquirir/utilizar, así como las librerías, programas compatibles, comentarios de usuarios previos y básicamente toda información que ayude a escoger correctamente cada dispositivo.
- Fomentar la investigación y el desarrollo de prototipos y equipos en el área de la neurociencia, resaltando las ventajas que estos sistemas brindan no sólo a las personas con patologías cerebrales específicas, sino a la comunidad científica y por ende a la población en general. Las posibilidades de escalamiento, actualización y mejora para los sistemas BCI son muy amplias, ya que hay muchos factores asociados al comportamiento del sistema nervioso que son complejos a la hora de identificar y controlar, además de que todavía existen muchos fenómenos asociados que son de origen desconocido y/o materia de investigación.

7 REFERENCIAS BIBLIOGRÁFICAS

- [1] S. N. Abdulkader, A. Atia, y M. S. M. Mostafa, «Brain computer interfacing: Applications and challenges», *Egyptian Informatics Journal*. 2015. doi: 10.1016/j.eij.2015.06.002.
- [2] T. Prell, N. Gaur, B. Stubendorff, A. Rödiger, O. W. Witte, y J. Grosskreutz, «Disease progression impacts health-related quality of life in amyotrophic lateral sclerosis», *J Neurol Sci*, vol. 397, pp. 92-95, feb. 2019, doi: 10.1016/j.jns.2018.12.035.
- [3] M. van Vliet, «Guidelines for data analysis scripts», *ArXiv*, vol. abs/1904.06163, 2019.
- [4] M. van Vliet, «Seven quick tips for analysis scripts in neuroimaging», *PLoS Comput Biol*, vol. 16, n.º 3, pp. e1007358-, mar. 2020, [En línea]. Available: <https://doi.org/10.1371/journal.pcbi.1007358>
- [5] C. Lin, B. Lin, F. Lin, y C. Chang, «Brain Computer Interface-Based Smart Living Environmental Auto-Adjustment Control System in UPnP Home Networking», *IEEE Syst J*, vol. 8, n.º 2, pp. 363-370, jun. 2014, doi: 10.1109/JSYST.2012.2192756.
- [6] H. Peng, B. Hu, Y. Qi, Q. Zhao, y M. Ratcliffe, «An improved EEG de-noising approach in electroencephalogram (EEG) for home care», en *2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops*, may 2011, pp. 469-474. doi: 10.4108/icst.pervasivehealth.2011.246021.
- [7] D. G. Duru, A. Deniz Duru, D. E. Barkana, O. Sanli, y M. Ozkan, «Assessment of surgeon's stress level and alertness using EEG during laparoscopic simple nephrectomy», en *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*, nov. 2013, pp. 452-455. doi: 10.1109/NER.2013.6695969.
- [8] A. Navarro *et al.*, «Context-Awareness as an Enhancement of Brain-Computer Interfaces», 2011, vol. 6693, pp. 216-223. doi: 10.1007/978-3-642-21303-8_30.
- [9] J. R. Wolpaw *et al.*, «Independent home use of a brain-computer interface by people with amyotrophic lateral sclerosis», *Neurology*, vol. 91, n.º 3, pp. e258-e267, jul. 2018, doi: 10.1212/WNL.00000000000005812.

- [10] L. Carelli *et al.*, «Brain-Computer Interface for Clinical Purposes: Cognitive Assessment and Rehabilitation», *BioMed Research International*, vol. 2017. Hindawi Limited, 2017. doi: 10.1155/2017/1695290.
- [11] B. Jagadish, M. P. R. S. Kiran, y P. Rajalakshmi, «A novel system architecture for brain controlled IoT enabled environments», en *2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom)*, oct. 2017, pp. 1-5. doi: 10.1109/HealthCom.2017.8210814.
- [12] F. Laport, A. Dapena, P. M. Castro, F. J. Vazquez-Araujo, y D. Iglesia, «A Prototype of EEG System for IoT», *Int J Neural Syst*, vol. 30, n.º 07, p. 2050018, feb. 2020, doi: 10.1142/S0129065720500185.
- [13] E. García, C. Fernandez, y M. Gaviria, «Interfaz cerebro computador basada en P300 para la comunicación alternativa: estudio de caso en dos adolescentes en situación de discapacidad motora », *Revista Facultad de Ingeniería Universidad de Antioquia*, pp. 0-7, 2011.
- [14] C. Blanco, «Estudio Comparativo de Métodos para el Reconocimiento de Potenciales Relacionados a Eventos P300 para una Interfaz Cerebro-Computador», Universidad Antonio Nariño, Bogotá, 2020.
- [15] W. Anaya y A. Serrano, «Diseño de un sistema de adquisición de señales encefalográficas a partir de AFE», Universidad de Ibagué, Ibagué, 2019. Accedido: oct. 05, 2022. [En línea]. Available: <https://hdl.handle.net/20.500.12313/1151>
- [16] L. Barbosa, M. Valderrama, y M. le Van Quyen, «Cognitive Function Using Event-Related Potentials (P300) On Epileptic Patients: A Colombian Study», pp. 0-11, 2019.
- [17] G. Izquierdo, «Prototipo de equipo médico de Neurofeedback para videojuego controlado con potenciales evocados», Universidad El Bosque, Bogotá, 2022.
- [18] W. Orjuela y A. Paipa, «Diseño e implementación de un prototipo cerebro-computador, basado en potenciales visuales evocados de estado estacionario, para comunicar mensajes simples», Universidad El Bosque, Bogotá, 2022.
- [19] C. Jiménez, L. Zarco, C. Castañeda, M. Otálora, y A. Martínez, «Estado actual de la esclerosis múltiple en Colombia», *Acta Neurológica Colombiana*, p. 6, 2015.
- [20] A. C. de Neurología, «Esclerosis Múltiple: la enfermedad huérfana con más casos en Colombia», Bogotá, p. 1, 2020. [En línea]. Available:

- <https://gestarsalud.com/2020/05/30/esclerosis-multiple-es-la-enfermedad-huerfana-con-mas-casos-en-colombia/>
- [21] K. Utsumi, K. Takano, Y. Okahara, T. Komori, O. Onodera, y K. Kansaku, «Operation of a P300-based brain-computer interface in patients with Duchenne muscular dystrophy», *Sci Rep*, vol. 8, n.º 1, p. 1753, 2018, doi: 10.1038/s41598-018-20125-6.
- [22] M. Irdenia, M. Rodneys, y N. Jiménez, «Calidad de vida en la esclerosis múltiple: su relación con la depresión, fatiga y calidad del sueño», *Gaceta Médica Espirituana*, vol. 15, p. 9, 2013.
- [23] G. Foley, P. O'Mahony, y O. Hardiman, «Perceptions of quality of life in people with ALS: Effects of coping and health care», *Amyotrophic Lateral Sclerosis*, vol. 8, pp. 164-169, jun. 2007, doi: 10.1080/17482960601164532.
- [24] A. Viviescas, «Disease Progression Has Significant Effect on Quality of Life of ALS Patients, Study Says», *ALS News Today*, p. 2, 2019, [En línea]. Available: <https://alsnewstoday.com/news-posts/2019/01/11/disease-progression-affects-quality-of-life/>
- [25] G. Thibodeau y Patton K, *Anatomía y Fisiología*, 6.ª ed. Madrid: Editorial Elsevier España, 2007.
- [26] M. Corona Tinoco, «Sistema nervioso central y sistema nervioso periférico», *Coordinación de Universidad Abierta, Innovación Educativa y Educación a Distancia, UNAM*, 2019.
- [27] J. Hall y A. Guyton, *Guyton and Hall Textbook of Medical Physiology*, 12.ª ed. Philadelphia: ELSEVIER, 2011.
- [28] D. Reyes-Haro, L. Bulavina, y T. Pivneva, «La glía, el pegamento de las ideas», *Revista Ciencia*, 2014.
- [29] E. Quintana y E. García, «Qué son las glías, las células “guardianas” claves para entender la bipolaridad y la esquizofrenia», *BBC News Mundo*, 2021.
- [30] B. McDermott *et al.*, «Gamma Band Neural Stimulation in Humans and the Promise of a New Modality to Prevent and Treat Alzheimer's Disease.», *J Alzheimers Dis*, vol. 65, n.º 2, pp. 363-392, 2018, doi: 10.3233/JAD-180391.
- [31] S. Luck, «Event-Related Potentials», *Handbook of Research Methods in Psychology*, pp. 0-18, 2012.

- [32] S. Luck, *An Introduction to the Event-Related Potential Technique*, 2nd ed. Iowa: MIT Press, 2014.
- [33] M. Congedo y F. da Silva, «Event-related potentials: General aspects of methodology and quantification», en *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*, 7th ed., Oxford University Press, 2018.
- [34] G. F. Woodman, «A brief introduction to the use of event-related potentials in studies of perception and attention», *Atten Percept Psychophys*, vol. 72, n.º 8, pp. 2031-2046, nov. 2010, doi: 10.3758/APP.72.8.2031.
- [35] E. S. Kappenman y S. J. Luck, *ERP Components: The Ups and Downs of Brainwave Recordings*. Oxford University Press, 2011. doi: 10.1093/oxfordhb/9780195374148.013.0014.
- [36] F. Barreto, L. C. Carrere, y C. B. Tabernig, «Cognitask: BCI system based on P300 for cognitive therapies», *IEEE Latin America Transactions*, vol. 20, n.º 6, pp. 884-890, feb. 2022, [En línea]. Available: <https://latam.ieeer9.org/index.php/transactions/article/view/6030>
- [37] E. Franz, «Development of a fast and efficient algorithm for P300 event related potential detection», 2014.
- [38] J. Amat, «Análisis de Componentes Principales (Principal Component Analysis, PCA) y t-SNE», https://www.cienciadedatos.net/documentos/35_principal_component_analysis, 2017.
- [39] N. v Manyakov, N. Chumerin, A. Combaz, y M. M. van Hulle, «Comparison of classification methods for P300 brain-computer interface on disabled subjects.», *Comput Intell Neurosci*, vol. 2011, p. 519868, 2011, doi: 10.1155/2011/519868.
- [40] J. Brownlee, «Linear Discriminant Analysis or LDA in Python», <https://machinelearningmastery.com/linear-discriminant-analysis-with-python/>, 2020.
- [41] Ministerio de Salud, *RESOLUCION NUMERO 8430*. 1993.
- [42] R. van Dinteren, M. Arns, M. L. A. Jongsma, y R. P. C. Kessels, «P300 development across the lifespan: a systematic review and meta-analysis.», *PLoS One*, vol. 9, n.º 2, p. e87347, 2014, doi: 10.1371/journal.pone.0087347.

- [43] J. Polich, «Updating P300: an integrative theory of P3a and P3b.», *Clin Neurophysiol*, vol. 118, n.º 10, pp. 2128-48, oct. 2007, doi: 10.1016/j.clinph.2007.04.019.
- [44] F. Aloise *et al.*, «P300-based brain-computer interface for environmental control: An asynchronous approach», *J Neural Eng*, 2011, doi: 10.1088/1741-2560/8/2/025025.
- [45] F. Nijboer *et al.*, «A P300-based brain-computer interface for people with amyotrophic lateral sclerosis.», *Clin Neurophysiol*, vol. 119, n.º 8, pp. 1909-1916, ago. 2008, doi: 10.1016/j.clinph.2008.03.034.
- [46] N. Subramaniyam, «LAB TALK: Brain Computer Interface(BCI) with the P300», <https://sapienlabs.org/lab-talk/brain-computer-interfacebci-with-the-p300/>, 2020.
- [47] L. Kirasirova, V. Bulanov, A. Ossadtchi, A. Kolsanov, V. Pyatin, y M. Lebedev, «A P300 Brain-Computer Interface With a Reduced Visual Field», *Front Neurosci*, vol. 14, dic. 2020, doi: 10.3389/fnins.2020.604629.
- [48] J. Wolpaw, «BCI Competition III databases», <https://www.bbci.de/competition/iii/>, 2004.
- [49] J. Wolpaw, «BCI Competition II databases», <https://www.bbci.de/competition/ii/>, 2002.
- [50] M. van Vliet, «GitHub: Marijin van Vliet», https://github.com/wmvanvliet/neuroscience_tutorials, 2021.
- [51] R. Arif, S. K. Wijaya, Prawito, y H. S. Gani, «Design of EEG data acquisition system based on Raspberry Pi 3 for acute ischemic stroke identification», en *2018 International Conference on Signals and Systems (ICSigSys)*, may 2018, pp. 271-275. doi: 10.1109/ICSIGSYS.2018.8372771.
- [52] M. Borzacchi, «Implementation of a Neonatal EEG Monitoring through Sonification on an Embedded Platform», University College Cork, Ireland, Turin, 2020.
- [53] D. Acharya, A. Rani, y S. Agarwal, «EEG data acquisition circuit system Based on ADS1299EEG FE», en *2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions)*, sep. 2015, pp. 1-5. doi: 10.1109/ICRITO.2015.7359346.
- [54] Texas Instruments, «ADS1299-x Low-Noise, 4-, 6-, 8-Channel, 24-Bit, Analog-to-Digital Converter for EEG and Biopotential Measurements»,

- https://www.ti.com/lit/ds/symlink/ads1299.pdf?ts=1611790836946&ref_url=https%253A%252F%252Fwww.google.com%252F, 2017.
- [55] Texas Instruments, «EEG Front-End Performance Demonstration Kit - User's Guide», https://www.ti.com/lit/ug/slau443b/slau443b.pdf?ts=1611760996472&ref_url=https%253A%252F%252Fwww.ti.com%252Ftool%252FADS1299EEGFE-PDK, 2016.
- [56] Texas Instruments, «ADCPro™ Hardware and Software Installation Manual», https://www.ti.com/lit/ug/slau372a/slau372a.pdf?ts=1630875983841&ref_url=https%253A%252F%252Fwww.google.com%252F, 2012.
- [57] OpenBCI, «OpenBCI Gold Cup Electrodes - Shop», <https://shop.openbci.com/products/openbci-gold-cup-electrodes>, 2021.
- [58] OpenBCI, «Ten20 Conductive Paste 8oz Jar», <https://shop.openbci.com/products/ten20-conductive-paste-8oz-jar>, 2021.
- [59] L. Krol, «EEG electrode positions in the 10-20 system», [https://en.wikipedia.org/wiki/10%E2%80%9320_system_\(EEG\)#/media/File:EEG_10-10_system_with_additional_information.svg](https://en.wikipedia.org/wiki/10%E2%80%9320_system_(EEG)#/media/File:EEG_10-10_system_with_additional_information.svg), 2020.
- [60] Asanagi, «Electrode locations of International 10-20 system for EEG (electroencephalography) recording», <https://commons.wikimedia.org/w/index.php?curid=10489987>, 2010.
- [61] C. Russomanno, «Texas Instruments ADS1299: Arduino Library», <https://github.com/conorrussomanno/ADS1299>, 2013.
- [62] S. Mathôt, «OpenSesame web page», <https://osdoc.cogsci.nl/>, 2022.
- [63] X. Jiang, G.-B. Bian, y Z. Tian, «Removal of Artifacts from EEG Signals: A Review.», *Sensors (Basel)*, vol. 19, n.º 5, feb. 2019, doi: 10.3390/s19050987.
- [64] J. Delua y IBM Analytics, «Supervised vs. Unsupervised Learning: What's the Difference?», <https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning>, 2021.
- [65] Twilio, «Twilio Homepage», <https://www.twilio.com/>, 2022.

8 ANEXOS

8.1 Anexo A: Tabla comparativa de los dispositivos de adquisición de señales EEG

Fabricante y nombre comercial del dispositivo	Texas Instruments ADS1299 EEG-FE	Open BCI Cyton	ANALOG DEVICES EVAL-SDP-CH1Z - EVAL-AD7768FMCZ	ANALOG DEVICES EVAL-SDP-CH1Z - EVAL-AD7779FMCZ	MAXIMINTEGRATED MAX11040K Evaluation Kit/Daughterboard	Peso equivalente (%)
Referencia del ADC	ADS1299	ADS1299	AD7768	AD7771	MAX11040K	NA
Tasa de muestreo	16 kSPS	16 kSPS	256 kSPS	128 kSPS	16 kSPS	14
Muestreo continuo	Si	Si	Si	Si	Si	10
Canales	8	8	8	8	4	9
Resolución del ADC (bits)	24	24	24	24	24	10
CMRR	-110 dB	-110 dB	-120 dB	-109 dB	-109 dB	15
Acople RC de entrada	Sí	Sí	Si	Sí	Si	5
Protecciones adicionales	No	Sí	Sí	Sí	No	5
Software	Básico	Avanzado	Básico	Básico	Básico	1
Interfaz de comunicación	SPI a USB	SPI a USB	SPI a USB	SPI a USB	USB	5
Inalámbrico	No	Sí	No	No	No	1
Multipataforma	Si	Si	Si	Si	Si	9
Precio (USD)(*)	199	499	337	298	328	16

(*) Nota: El precio de los dispositivos en USD es el mostrado en los respectivos catálogos a la fecha de consulta, que fue entre el 15/5/2021 y el 15/09/2021.

8.2 Anexo B: Manual de Usuario

Sistema BCI integrable a una red IoT para interacción con dispositivos inteligentes del entorno

Versión del prototipo: 1.0

Manual de usuario

1. Introducción

Este manual va dirigido para todas aquellas personas interesadas en probar y realizar experimentos con el prototipo basado en el ADS1299EEG-FE. A lo largo del documento serán mostrados los pasos necesarios para mostrar el funcionamiento básico del sistema, además de las soluciones a problemas comunes encontrados durante la operación.

2. Requerimientos de operación

El sistema total requiere la utilización de componentes de hardware y software, que se especifican a continuación.

2.1. Componentes de Hardware

- ADS1299EEG-FE
- Arduino UNO
- Cable USB-B
- Jumpers macho-macho (x7)
- Cable para jack de alimentación del ADS1299EEG-FE:



- Electroodos gold-cap (x4)
- Pasta/crema Ten20
- Cinta hipoalergénica especial para adherencia a la piel (opcional pero recomendable).
- Batería/Fuente de alimentación DC (evitar conversores AC/DC).

2.2. Componentes de Software:

- Python (v3.8+) o Anaconda
- IDE Python, recomendado: Spyder, Pycharm
- Arduino IDE (v1.8.16+)
- OpenSesame
- PuTTY

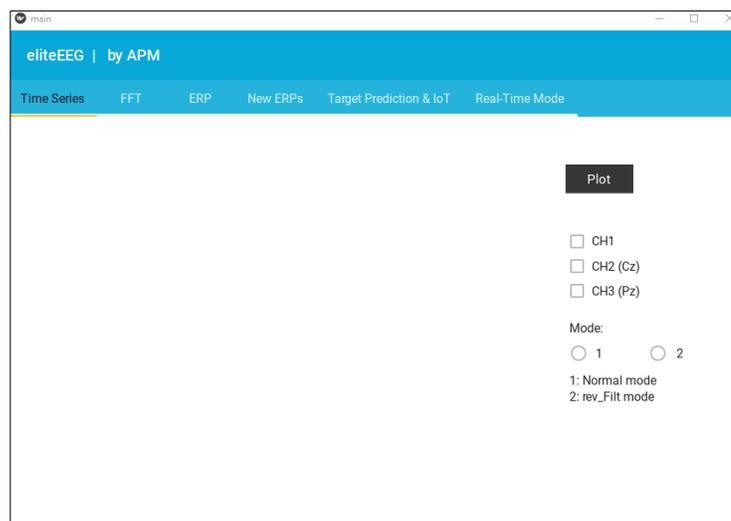
3. Funcionamiento del sistema

Primeramente, se recomienda descargar los archivos del siguiente repositorio:

<https://github.com/alexfpm/BCI-P300-IoT>

Encontrará 6 carpetas. Exceptuando la llamada “video_and_pictures”, el resto de los directorios tienen los archivos necesarios para todos los módulos del sistema. Se recomienda organizarlos en carpetas de nombre similar, para tener organizados los códigos.

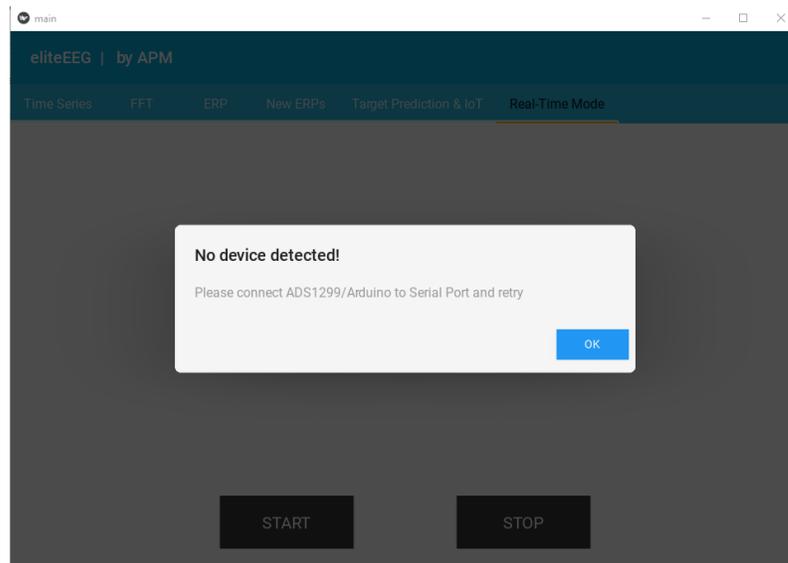
Una vez descargados todos los archivos, ejecute el IDE de preferencia e importe el archivo “mainApp.py”. Al ejecutar, verá la pantalla principal de la interfaz gráfica:



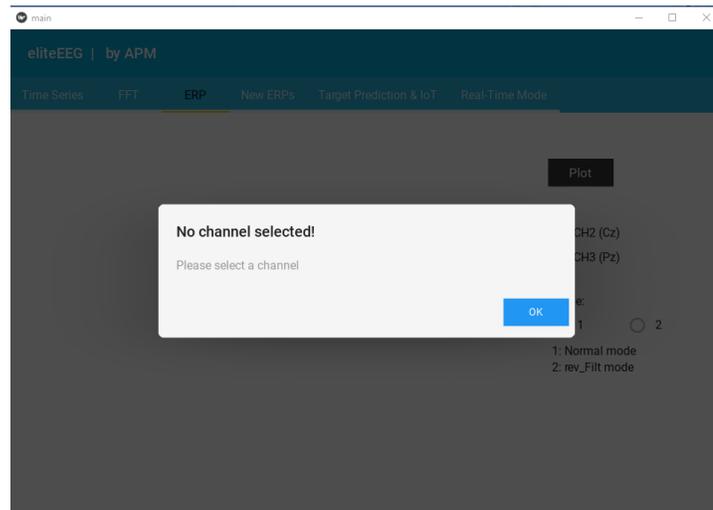
Pantalla principal al ejecutar la app.

Puede navegar por las pestañas (tabs), para familiarizarse con las opciones. Esta es la interfaz para el procesamiento de los datos EEG adquiridos. Antes de realizar algunas acciones, tenga en cuenta que el sistema globalmente tiene dos modos de funcionamiento principal:

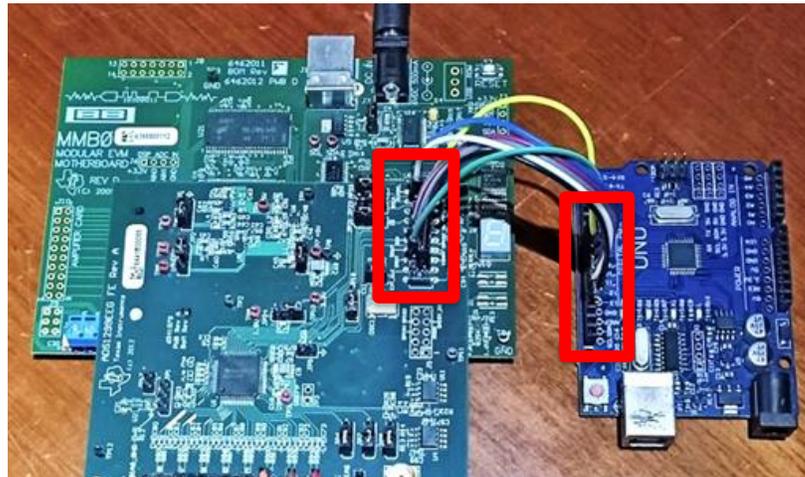
- **Real-time Mode:** Desde la última pestaña de la interfaz (con este nombre), se pueden adquirir y ver en tiempo real datos EEG de un canal de medición. Para eso, debe configurar previamente el ADS1299EEG-FE junto con el Arduino. De no hacerlo, verá un mensaje que le indica que el programa no detectó los dispositivos conectados:



- **Data-Analysis Mode:** El resto de las pestañas de la interfaz permiten hacer operaciones de procesamiento y análisis de datos EEG previamente adquiridos y registrados. Para eso, debe colocar los archivos necesarios en el mismo directorio de la app, para ser reconocidos. Si no existen archivos para analizar, en cualquiera de las opciones verá un mensaje que le indicará que debe seleccionar primero algún canal de medición.



Para poder utilizar cualquiera de las funciones y adquirir datos EEG, debe configurar primero la comunicación entre el ADS1299EEG-FE y el Arduino. Mediante jumpers, conecte la interfaz SPI del ADS1299EEG-FE con los respectivos puertos del Arduino, como se indica:



Las interfaces son las resaltadas en rojo.

Los puertos específicos del Arduino para SPI son:

Arduino Uno - Pin Assignments

MISO [DOUT] = 12

MOSI [DIN] = 11

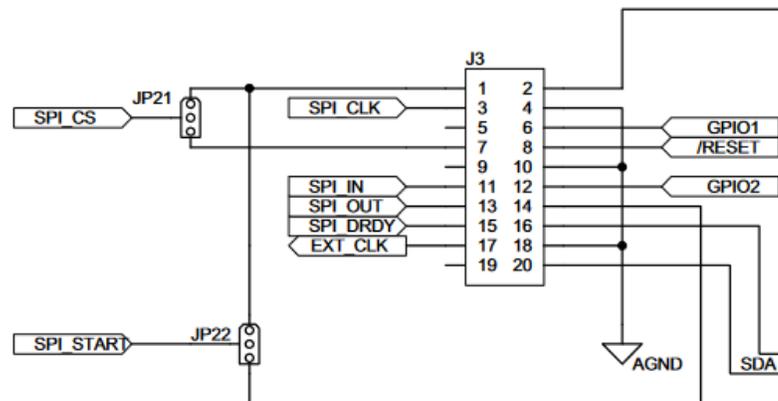
SCK = 13

DRDY = 8;

CS = 10;

RESET = 9;

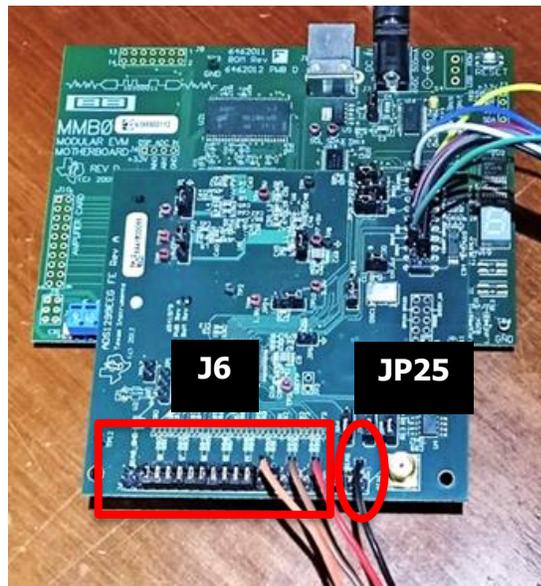
Para el ADS1299EEG-FE, tenga en cuenta la pines que indica el datasheet:



Una vez conectados correctamente, se explicarán las configuraciones para cada modo de funcionamiento en las secciones siguientes.

3.1. Real-Time Mode

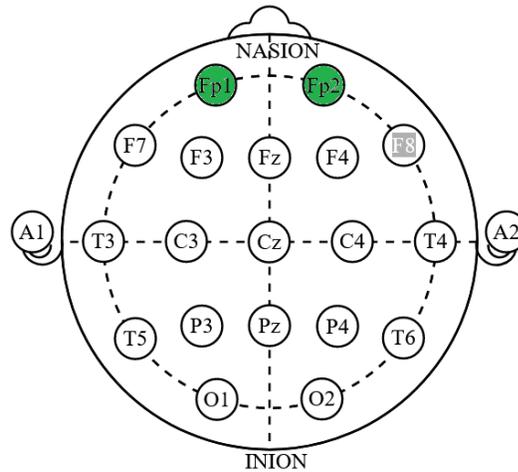
Para este modo, necesitará mínimo dos electrodos de medición, pero se recomienda conectar los 4 electrodos para posterior uso y configuración. Retire los protectores plásticos del puerto de electrodos del ADS1299EEG-FE, y conecte los electrodos a los pines físicos como se indica:



Tenga en cuenta que los pines del puerto J6 indicado es donde se ubican los electrodos de medición, y en los pines de JP25 se ubica el electrodo de referencia eléctrica (consulte el datasheet de TI para más información). Una vez conectados, coloque el terminal de medición del electrodo de referencia eléctrica en el lóbulo de la oreja derecha o izquierda (el que prefiera para su comodidad):

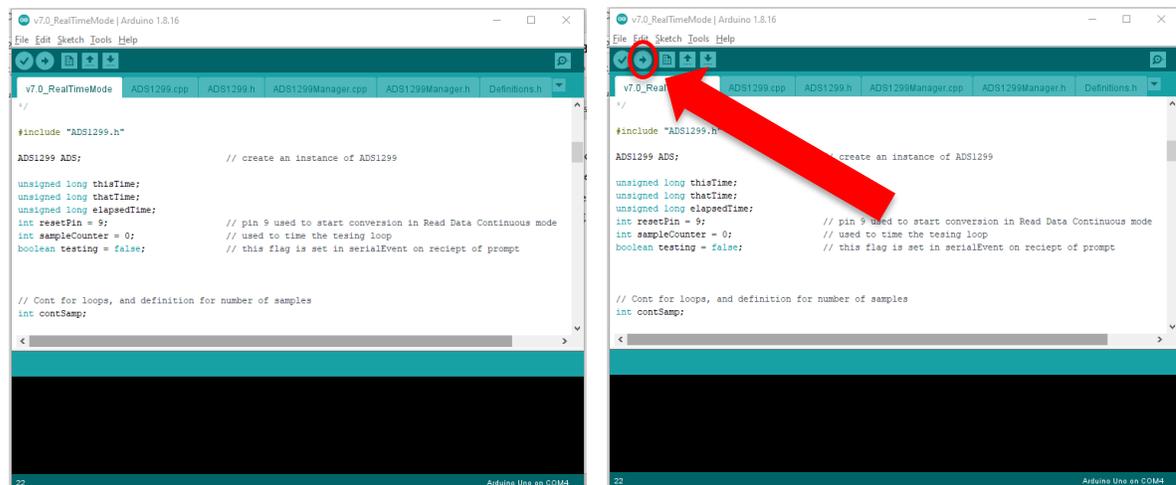


Para fijar el electrodo, aplique una pequeña cantidad de la pasta Ten20 sobre el terminal del electrodo (puede añadir la cinta hipoalergénica para mejorar la fijación). Posteriormente, ubique un electrodo de medición en la posición Fp2 o Fp1:



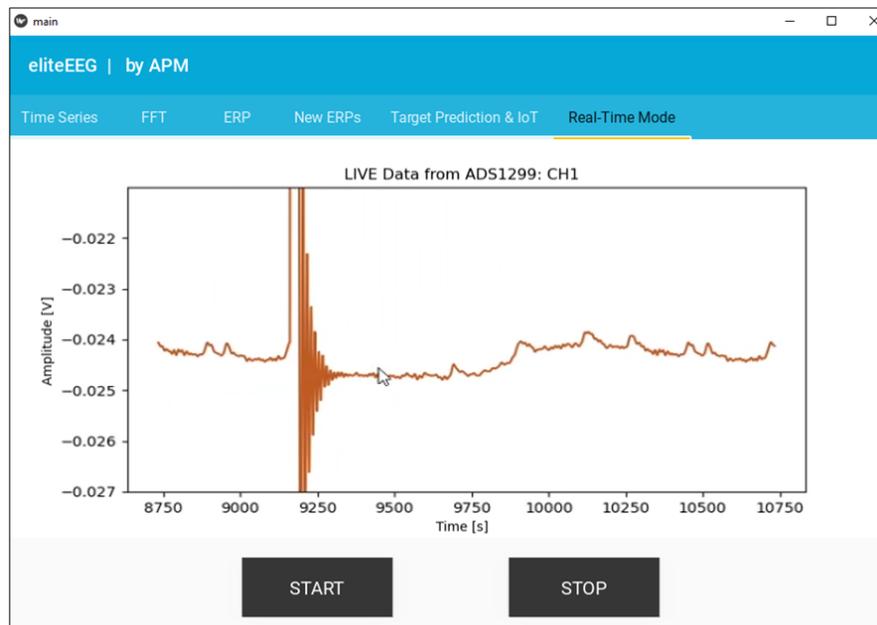
Para fijarlo, se recomienda aplicar nuevamente Ten20 sobre la superficie del electrodo o sobre el cuero cabelludo.

Una vez ubicados los electrodos, desde el computador ubique el archivo “v7.0_RealTimeMode.ino” y ejecútelo. Se abrirá el IDE de Arduino (imagen de la izquierda):



Conecte el Arduino al computador y cargue el archivo abierto a la tarjeta dando click en el botón indicado en rojo (imagen de la derecha). Una vez programado el Arduino, conecte la alimentación DC al ADS1299EEG-FE.

Posteriormente, ubique en el computador nuevamente el archivo “mainApp.py” y ejecútelo para abrir la interfaz gráfica. Ahora, diríjase a la pestaña “Real-Time Mode” y haga click en el botón “START”. El programa establecerá la comunicación con los dispositivos y comenzará a graficar la señal EEG, similar a como se indica:



Intente parpadear y ver los cambios de voltaje en la pantalla. Deben aparecer cambios súbitos de voltaje casi al mismo tiempo que parpadea, lo cual puede verificar si realiza parpadeos rápidos sucesivos. Estas pruebas son las conocidas como “Ojos abiertos-ojos cerrados”, y permiten evidenciar que todas las conexiones previas y configuraciones están correctas, además de verificar que los equipos están funcionando adecuadamente.

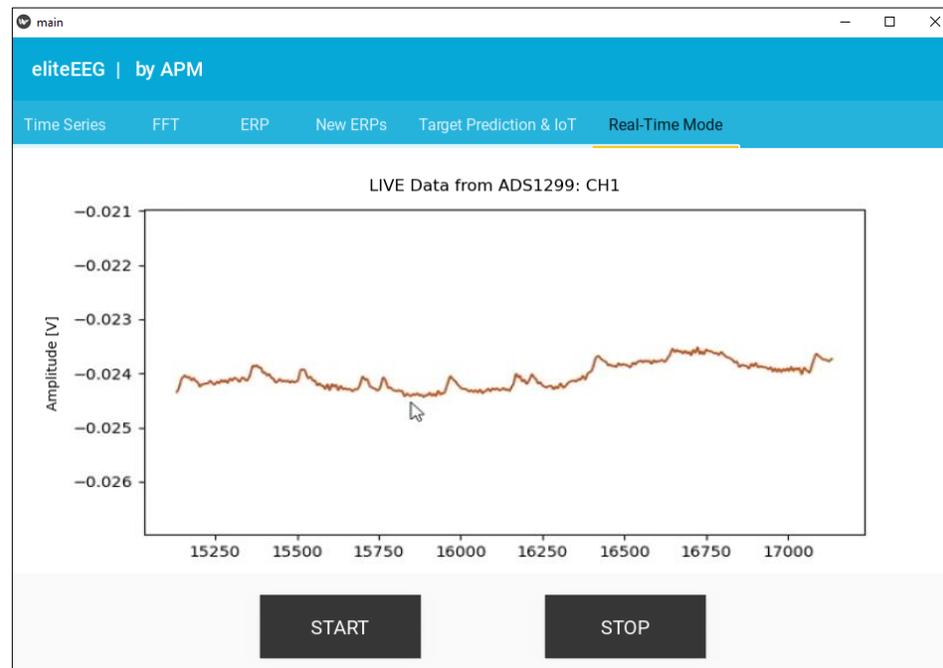


Figura: Ejemplos de parpadeos detectados en la interfaz.

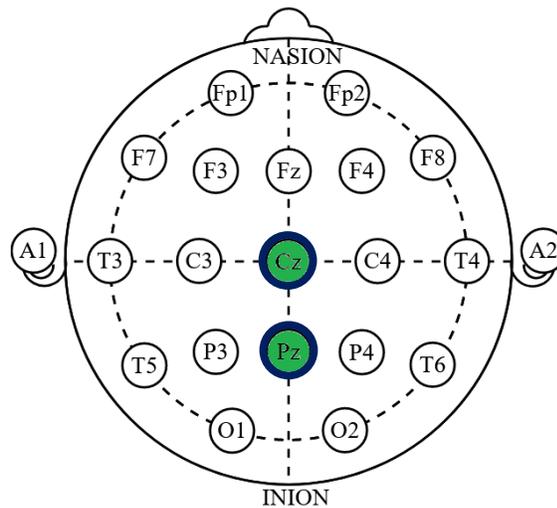
Si todas las conexiones y configuraciones están correctas, una vez iniciado el programa y pasados unos 5 segundos, debería ver una señal parecida a la mostrada en la figura superior. Cada uno de los picos que se aprecian son parpadeos detectados.

Para detener el muestreo y la visualización, pulse el botón “STOP”.

3.2. Data-Analysis Mode

En este modo, el objetivo será identificar un símbolo que verá en pantalla, mediante el análisis de los datos EEG. Para ello, previamente deberá tener conectados el ADS1299EEG-FE con el Arduino, mediante los pines para SPI (diríjase a la sección Real-Time Mode para más detalles del proceso.)

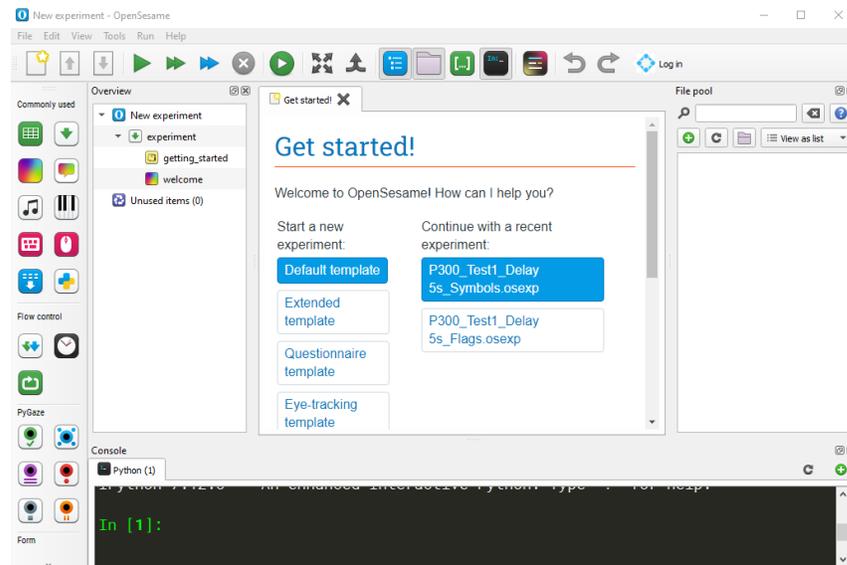
Una vez conectados los dispositivos, ubique el electrodo de referencia eléctrica en el lóbulo de la oreja derecha o izquierda (la más conveniente para su posición), y posteriormente ubique dos (2) electrodos de medición en las posiciones Cz y Pz, como se indica en el diagrama:



Recuerde colocar la pasta Ten20 sobre la zona de contacto con el electrodo en el cuero cabelludo, para asegurar una buena medición.

Una vez fijados los electrodos, conecte la alimentación DC al puerto del ADS1299EEG-FE, y posteriormente conecte el Arduino al computador. Ubique el archivo llamado “v7.0_DataAcquisition-ImageMode.ino” y ejecútelo para que se abra la correspondiente IDE de Arduino. Desde ahí, cargue el programa a la tarjeta Arduino, siguiendo como referencia el mismo proceso del Real-Time Mode.

Una vez cargado el programa, desde el computador abra el software OpenSesame:

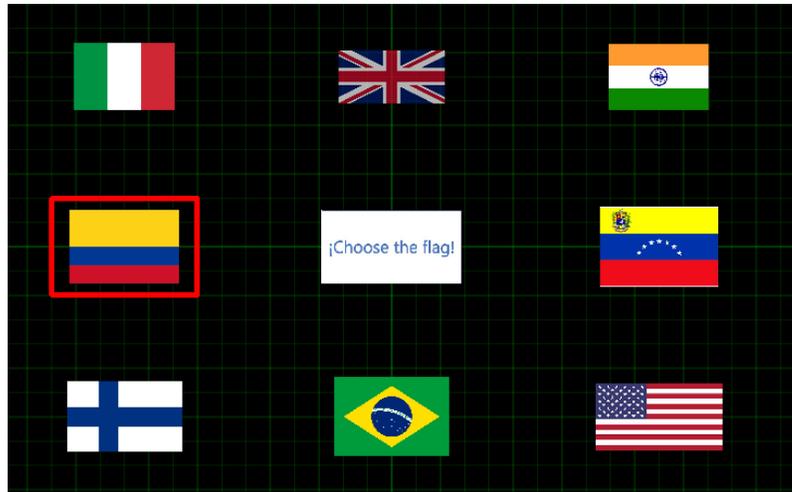


Ventana al abrir OpenSesame

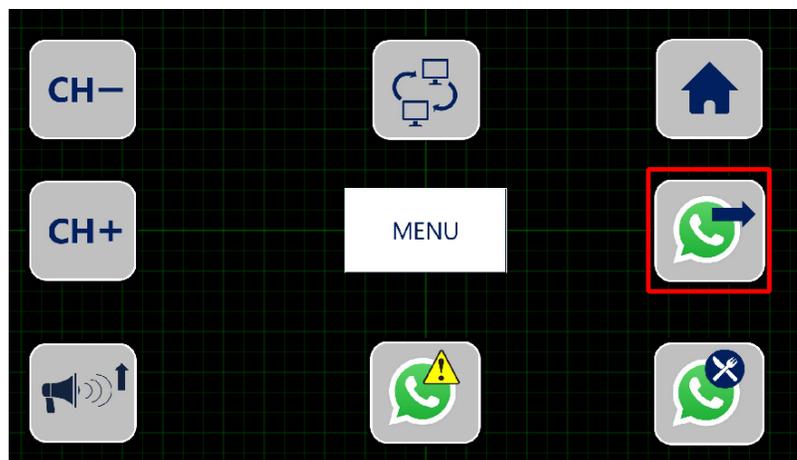
Desde la pestaña superior, haga click en File > Open y ubique alguno de los archivos llamado “P300_Test1_Delay 5s_Flags.osexp” o “P300_Test1_Delay 5s_Symbols.osexp”. Haga click en “Open” para abrir el archivo escogido. Cualquiera de los dos funciona para realizar el experimento.

Una vez abierto alguno de los dos archivos, OpenSesame estará listo para realizar un experimento Oddball. Este proceso es necesario para adquirir los datos EEG mientras usted ve la pantalla con banderas o símbolos (dependiendo del archivo que haya escogido), y adquirir los datos que se analizarán posteriormente. También puede optar por analizar datos previamente adquiridos, para lo cual se puede saltar este paso y dirigirse a la sección “3.2.1 – Análisis de datos desde la Interfaz Gráfica” de este manual.

Ahora bien, dependiendo del archivo escogido, OpenSesame mostrará dos tipos de imágenes para el experimento: una matriz 3x3 con 8 banderas posibles, o una matriz 3x3 con 8 símbolos posibles. A continuación se muestran:



Pantalla de la matriz con las banderas.



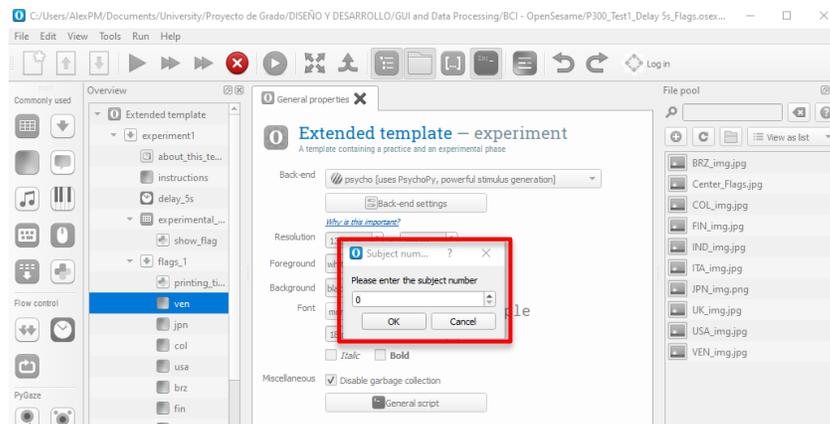
Pantalla de la matriz con los símbolos.

Se recomienda familiarizarse con cada una. El experimento que verá en pantalla es el siguiente:

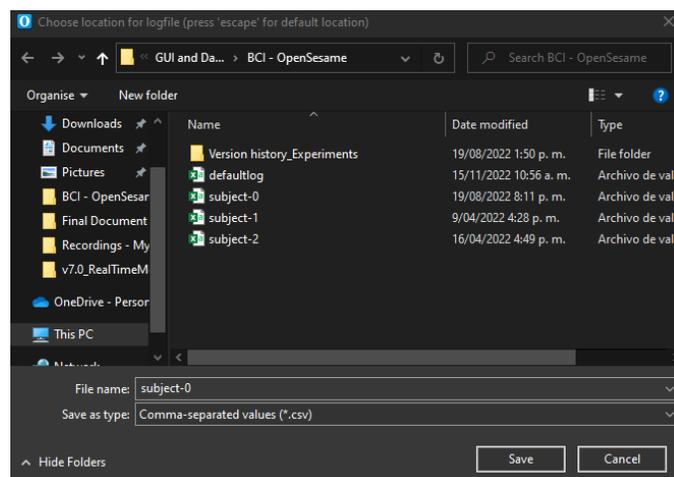
- Usted debe escoger mentalmente, de cualquiera de las pantallas, una opción (bandera/símbolo).
- Al ejecutar el archivo escogido, OpenSesame iniciará el experimento y durante 60 segundos, aleatoriamente irá resaltando cada opción con un cuadro rojo.

- Usted debe simplemente quedarse quieto y concentrarse en mirar la opción escogida en pantalla.
- Evite moverse todo lo posible durante los 60 segundos, relajando los músculos del cuerpo. Puede parpadear, pero intente que sean parpadeos suaves.
- Luego de los 60 segundos, el experimento finalizará.

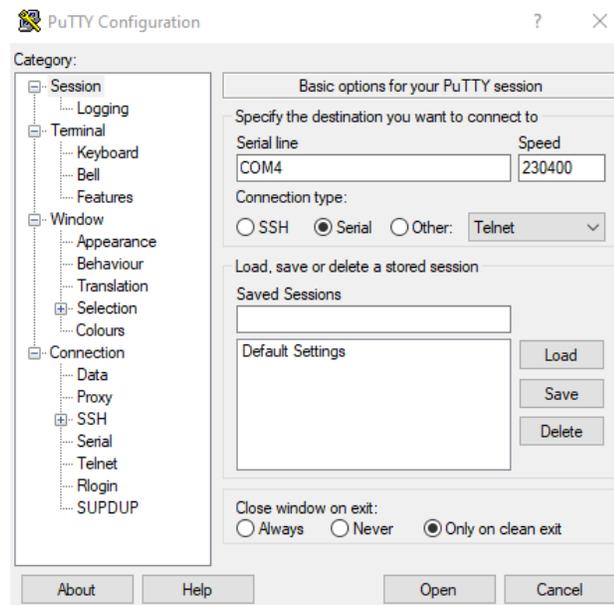
Para iniciar el experimento, de click en el botón de “Run” (triángulo verde de la barra superior) o en la opción de Run>Run fullscreen (menú superior). Aparecerá una ventana emergente:



Introduzca en la ventana el número de preferencia, y de click en “OK”. Luego, escoja un directorio en su computador para guardar el archivo de OpenSesame con extensión .csv.



Antes de dar click en “Guardar”, abra el software PuTTY en su computador:



Configure una sesión con conexión de tipo “Serial” y velocidad de 230400 baudios. Debe especificar el puerto COM al que esté conectado el Arduino/ADS1299EEG-FE. De click en “Open”. Aparecerá una ventana como la siguiente:

The image shows a PuTTY terminal window titled 'COM4 - PuTTY'. The terminal displays the following text:

```

-----
Register 0x02 modified.
CONFIG2, 0x02, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
-----

Registers 0x05 to 0x0C modified
CH1SET, 0x05, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
CH2SET, 0x06, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
CH3SET, 0x07, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
CH4SET, 0x08, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
CH5SET, 0x09, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
CH6SET, 0x0A, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
CH7SET, 0x0B, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
CH8SET, 0x0C, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
-----

Register 0x15 modified.
MISC1, 0x15, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
Register 0x14 modified.
GPIO, 0x14, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
Register 0x01 modified.
CONFIG1, 0x01, 0x00, 0, 0, 0, 0, 0, 0, 0, 0, 0
-----

Press 'x' to begin test

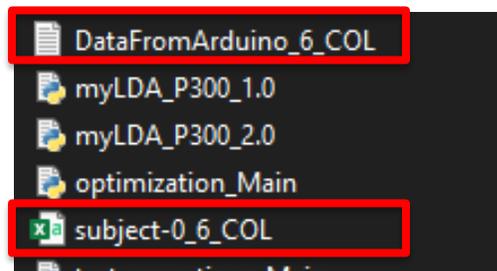
```

Si ha seguido todos los pasos hasta este punto, está listo para tomar las mediciones EEG durante el experimento. Desde OpenSesame, de click en “Guardar” sobre la ventana que estaba activa, y posteriormente, presione la tecla “X” de su teclado en la ventana del PuTTY. Comenzará la toma de datos, y segundos después en su pantalla comenzará el experimento de OpenSesame.

Pasados los 60 segundos, el experimento finalizará, y volverá a la ventana de OpenSesame, que ya puede cerrar. Espere que en la ventana del PuTTY se dejen de mostrar los datos que está adquiriendo el ADS1299, y posteriormente ya puede guardar los datos y cerrar la sesión. Desde el PuTTY, los datos de la sesión abierta quedarán registrados en un archivo de texto que debe guardar en su computador.

3.2.1. Análisis de datos desde la Interfaz Gráfica

Ubique el archivo “mainApp.py” y ejecútelo, para abrir la interfaz gráfica. Tenga presente que para utilizar datos previamente adquiridos, debe ubicar tanto el archivo de texto guardado desde el PuTTY (.txt) como el generado por OpenSesame (.csv) en el mismo directorio donde tenga guardado el script que ejecuta la interfaz gráfica (.py). Un ejemplo de ambos archivos se muestra en la siguiente imagen:



El par de archivos .txt y .csv forman una “base de datos” que funciona para analizar desde la interfaz gráfica. Pares de estos archivos con mediciones previas puede encontrar en el repositorio de GitHub indicado al comienzo del manual, en la carpeta “databases_tests”.

Ahora bien, una vez abierta la interfaz gráfica, puede empezar a visualizar los datos. Diríjase a la pestaña “Time-Series”, seleccione todos los canales y en “Mode” seleccione el segundo modo. De click en el botón de “Plot” para graficar:



Como se observa, la interfaz se divide en 3 grandes secciones:

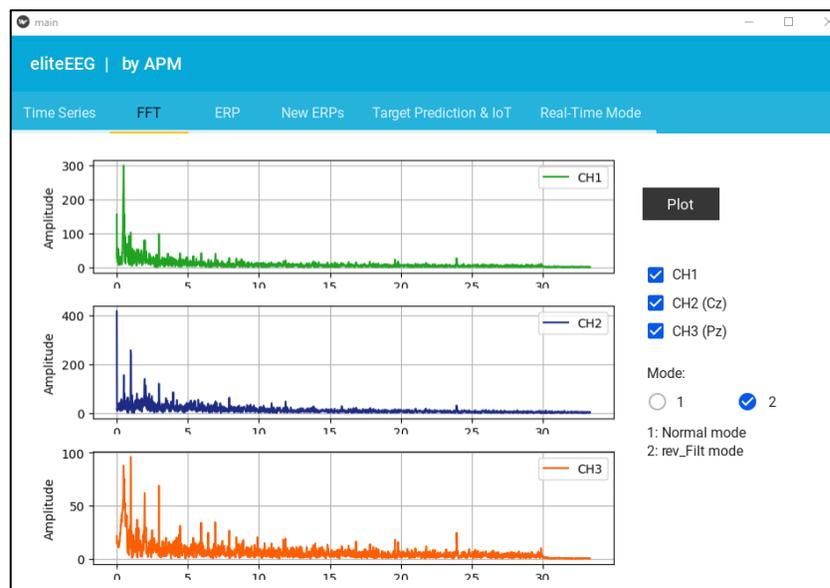
- (1) Toolbar superior: contiene las pestañas para navegación, y el nombre con el creador de la app. Utilícela para desplazarse por las diferentes pestañas y opciones disponibles.
- (2) Área de gráficas: En este espacio aparecerán las gráficas de cada una de las opciones y pestañas de la app. Cuando no existan datos, el área permanecerá blanca.

- (3) Menú de opciones de análisis/procesamiento: contiene los canales para visualizar, opciones para filtrado, sliders para navegar en los datos y demás opciones disponibles (cambian según la pestaña seleccionada).

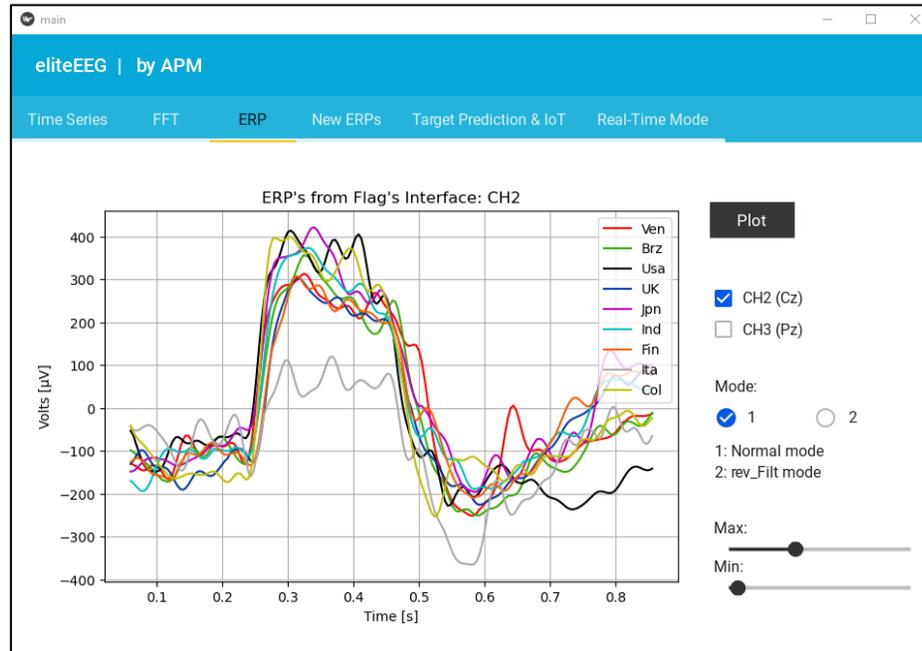
Cada pestaña le permitirá hacer operaciones particulares:

1. **Time-Series:** Análisis en el dominio temporal.
2. **FFT:** Análisis en el dominio frecuencial.
3. **ERP:** Cálculo y visualización del ERP para los datos adquiridos.
4. **New ERPs:** Filtros y algoritmos adicionales para el cálculo y visualización de los ERPs.
5. **Target Prediction & IoT:** Predicción del objetivo seleccionado y conexión con red IoT.
6. **Real-Time Mode:** Visualización en tiempo real de un canal del ADS1299.

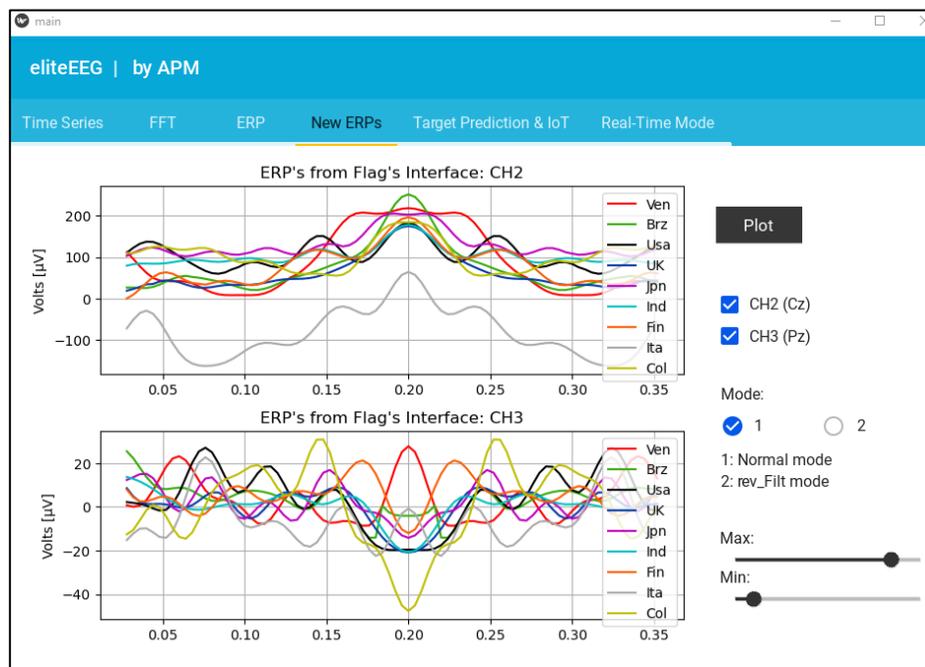
En todas las pestañas, hacer click sobre el botón de “Plot” mostrará la gráfica respectiva. Algunos ejemplos de lo que puede visualizar se muestran a continuación:



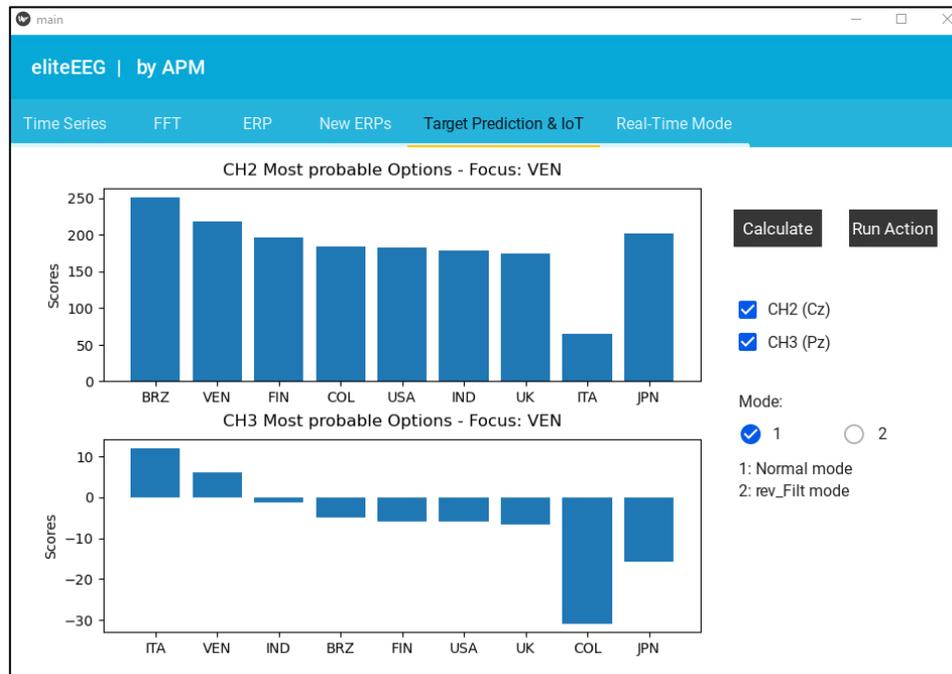
FFT de datos adquiridos para cada canal.



ERPs de datos para un experimento. Cada ERP se diferencia por un color asignado.



ERPs recalculados de los datos para el mismo experimento anterior. Los sliders que aparecen en el menú de opciones permiten navegar en las gráficas (zoom horizontal).



La imagen superior muestra las puntuaciones que calculó el programa para cada una de las opciones posibles, en este caso, de la interfaz de las banderas. Cada puntuación representa la probabilidad de que esa opción haya sido la escogida por la persona, y el sistema ordena de forma descendente cada una de las opciones según dicha puntuación.

- **Cálculo de la opción escogida**

De la pestaña “Target Prediction & IoT”, el sistema BCI identifica la opción más probable al hacer click en el botón “Calculate”. La primera opción representa la opción más probable, que es la que el algoritmo calculó como la escogida por la persona. Se recomienda que haga varias pruebas de adquisición de datos, con la interfaz de preferencia, para probar la variación en la detección de las distintas opciones posibles.

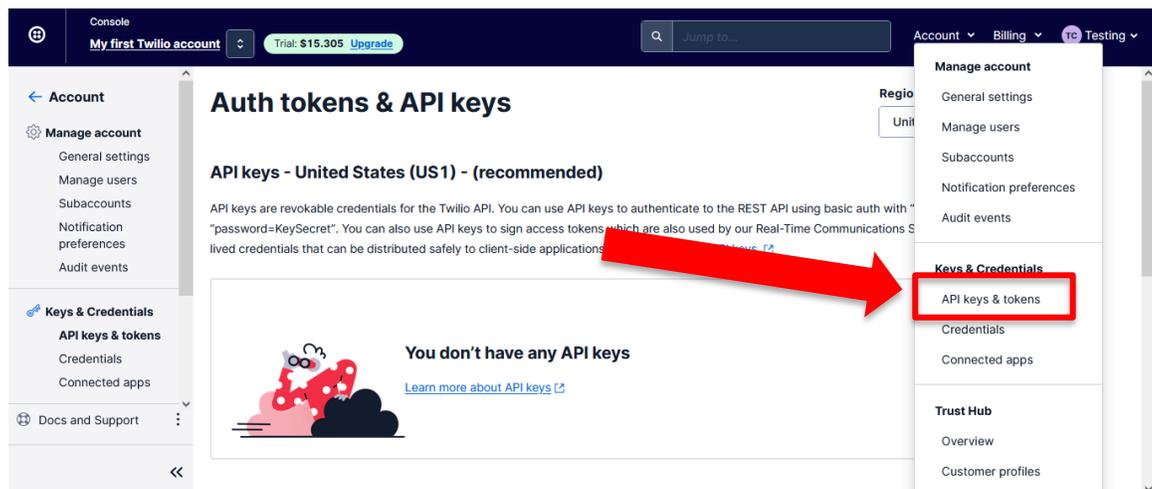
3.3. Red IoT

Este sistema BCI está construido para que pueda interactuar con algunos dispositivos inteligentes del entorno del usuario. Para este prototipo, los dispositivos compatibles son dos (2):

- Smartphone con sistema operativo Android (8.0+)
- SmartTV basado en webOS.

3.3.1. Conexión con smartphone Android

Para la comunicación con teléfonos Android, descargue la aplicación de mensajería *WhatsApp* desde la PlayStore. Posteriormente, debe registrarse en la web de Twilio (<https://www.twilio.com/>) para crear un usuario y una contraseña. Una vez registrado, dentro de la plataforma diríjase al apartado Account > API keys & tokens:



En la página, desplácese hacia abajo hasta encontrar los siguientes datos:

Auth Tokens - United States (US1)

Auth tokens can be used to authenticate while making API requests. You will need password=AuthToken. Auth tokens are specific to your account and can be used and rotate them periodically. [Learn more](#)

Live credentials

Account SID - used to exercise the REST API

Auth token [Request a secondary token](#)

⚠ Sensitive information. Store your token securely to protect your account. [Learn more](#)

[🔗](#)

Keep this somewhere safe and secure

Tome nota del “Account SID” y el “Auth token”. Deberá utilizar esos datos para comunicarse con el teléfono. Posteriormente, ubique el archivo “IoT_Actions.py” y reemplace los campos indicados en rojo con los anotados previamente:

```
def options_menu(option):
    # Logging in to Twilio Sandbox for WhatsApp
    sid = ' '
    authToken = ' '
    twilioClient = Client(sid, authToken)
```

Ahora, debe modificar los campos con el número de teléfono al cual recibirá los mensajes por WhatsApp, ubicados en el campo “to”:

```
elif option == "BRZ" or option == "SYM2":
    message = twilioClient.messages.create(to=' ',
                                           from=' ',
                                           body='OPTION 2')
```

Para el teléfono del campo “from”, debe dirigirse a su cuenta en Twilio en el apartado Develop > Messaging > Try it out > Send a WhatsApp Message. Ahí, encontrará el número que debe ubicar en el campo “from”, y también, encontrará las instrucciones para configurar correctamente un Sandbox que le permita recibir mensajes por WhatsApp.

Una vez realizado, guarde el archivo “IoT_Actions.py” con los cambios en el mismo directorio de la app. Ahora, en la interfaz gráfica, desde la pestaña “Target Prediction & IoT” podrá enviar mensajes al teléfono registrado, dando click en el botón “Run Action” de la pestaña. Tenga en cuenta que para eso, la opción identificada por la interfaz debe corresponder con algún comando de WhatsApp del menú de opciones.

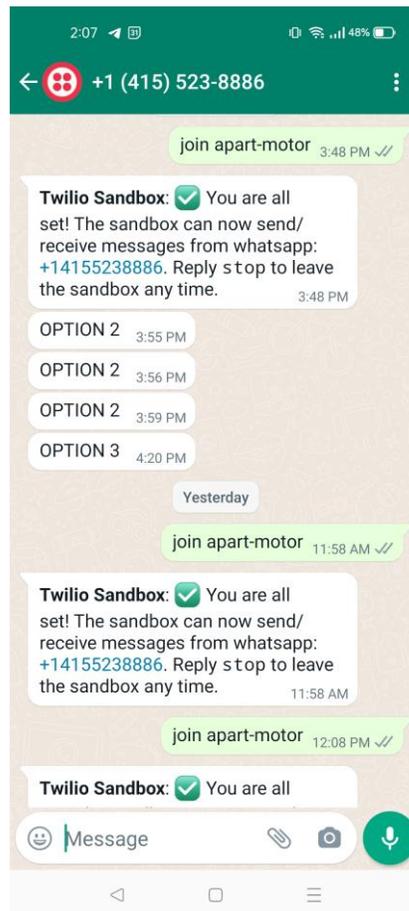


Figura: Imagen de referencia que evidencia los mensajes recibidos por WhatsApp mediante el Sandbox de Twilio.

3.3.2. Conexión con SmartTV

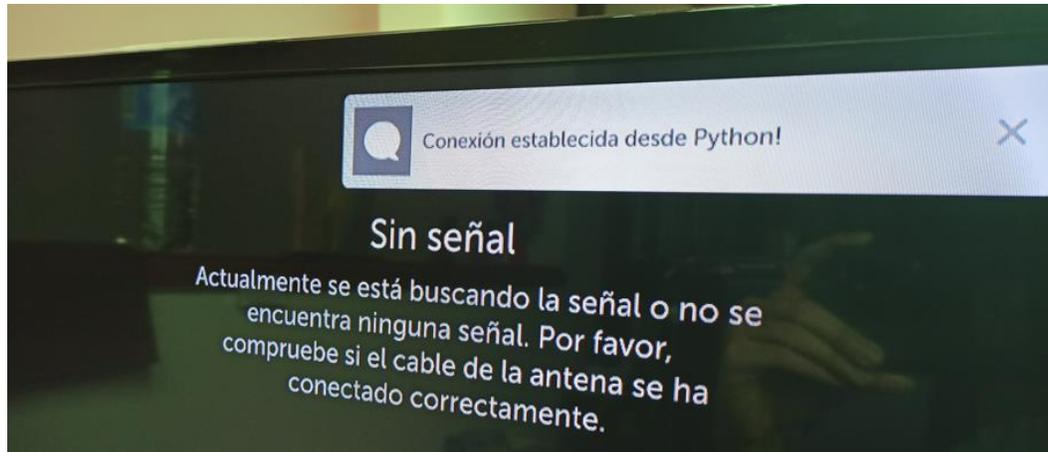
Para la comunicación con el televisor, tenga en cuenta que éste debe ser un SmartTV basado en el sistema operativo webOS. Como ejemplo de referencia, se utilizó uno de marca LG.

El control del SmartTV requiere conocer la dirección IPv4 del dispositivo. Para ello, dependiendo del modelo, normalmente encuentra este dato al dirigirse desde el menú de su TV al apartado Configuración > Red > Configuración Avanzada. Tome nota de la dirección IPv4 y, posteriormente, ubique el archivo “IoT_Actions.py” en su computador. Reemplace el campo indicado con la dirección IP de su TV:

```
# Establishing connection with TV
client = WebOSClient(" ")
client.connect()
```

Guarde el archivo con los cambios en el mismo directorio de la app, y ejecute el archivo de la interfaz gráfica. Ahí, desde la pestaña “Target Prediction & IoT” podrá realizar acciones y controlar el SmartTV, dando click en el botón “Run Action” de la pestaña. Tenga en cuenta que para eso, la opción identificada por la interfaz debe corresponder con algún comando orientado al TV, del menú de opciones.

Para más referencias sobre la librería de Python usada para webOS, puede consultar comandos, configuraciones y datos adicionales en el repositorio en GitHub: <https://github.com/supersaiyanmode/PyWebOSTV>.



La imagen superior fue tomada a la pantalla del SmartTV cuando se le envió desde Python una notificación para que la mostrara. Así, queda como evidencia de que la comunicación desde la interfaz gráfica era efectiva.

8.3 Anexo C: Web del Proyecto

Para más información sobre los códigos utilizados, archivos auxiliares, bases de datos de mediciones tomadas, videos y fotos del proyecto, puede dirigirse al siguiente repositorio en GitHub:

<https://github.com/alexfpm/BCI-P300-IoT>

Las subcarpetas y su contenido resumido son las siguientes:

- *GUI-kivy_files*: Archivos para la interfaz gráfica principal.
- *data-analysis_mode*: Archivos para leer datos durante el experimento de OpenSesame.
- *databases_tests*: Mediciones EEG tomadas con el prototipo.
- *openSesame_files*: Archivos de los experimentos en OpenSesame.
- *real-time_mode*: Archivos para leer datos en tiempo real.
- *video_and_pictures*: Galería de registros en video e imágenes del proyecto.