

Universidad de Alcalá Escuela Politécnica Superior

Grado en Ingeniería en Tecnologías de Telecomunicación



Trabajo Fin de Grado

Diseño de una interfaz BCI utilizando el amplificador EEG
portátil Unicorn Hybrid Black.

ESCUELA POLITECNICA
Autor: Álvaro Martín Fernández
Tutor: José Luis Martín Sánchez

Curso 2021/2022

UNIVERSIDAD DE ALCALÁ
ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería en Tecnologías de Telecomunicación

Trabajo Fin de Grado

**Diseño de una interfaz BCI utilizando el amplificador EEG
portátil Unicorn Hybrid Black**

Autor: Álvaro Martín Fernández

Tutor: José Luis Martín Sánchez

Tribunal:

Presidente: D. Ernesto Martín Gorostiza.

Vocal 1: D. José Antonio Jiménez Calvo.

Vocal 2: D. José Luis Martín Sánchez.

Calificación:

Fecha:

Agradecimientos

A mi familia, cuya educación y valores han hecho que me convierta en la persona que soy a día de hoy, apoyándome en los buenos y malos momentos. Todos mis logros se los deberé siempre a ellos.

A mi mujer Andrea, que siempre me ha escuchado, me ha comprendido y apoyado, siendo la pared sobre la que me sostengo.

A mi hijo Alexander, el cual ha sido mi mayor motivación para poder retomar mis estudios, querer seguir mejorando y adquiriendo conocimientos día a día para poder transmitírselos en un futuro.

A los compañeros, proveedores y clientes que durante mi vida laboral me han enseñado, ayudado y transmitido un sentimiento de ambición de querer ser el mejor en mi trabajo.

Y por último a todos los profesores, cuya formación ha hecho posible adquirir los conocimientos y aptitudes necesarias para realizar el proyecto, en especial a Jose Luis, mi tutor, que ha sabido guiarme en la elaboración y planteamiento del proyecto de un modo espectacular.

Gracias a todos.

Resumen

Este proyecto tiene como principal objetivo diseñar una interfaz cerebro-computador mediante un amplificador EEG portátil Unicorn Hybrid Black conectado a una CPU mediante Bluetooth para procesar e interpretar una serie de tareas simples mediante Matlab *δ Simulink*.

Para realizar dicho diseño tendremos que realizar previamente un estudio de muestras para identificar una serie de tareas sencillas mediante Matlab.

Después realizaremos una extracción de características y haremos uso del módulo de aprendizaje máquina de Matlab para crear un modelo que nos sirva para interpretar nuevas muestras.

Para finalizar crearemos mediante la herramienta Simulink dicha interfaz utilizando el módulo creado previamente para la toma de decisiones.

Palabras clave

- EEG.
- Machine Learning.
- Features.
- Matlab.
- Simulink.

Abstract

The main objective of this project is to design a brain-computer interface using a Unicorn Hybrid Black portable EEG amplifier connected to a CPU with Bluetooth to process and interpret a series of simple tasks using Matlab δ *Simulink*.

To carry out this design we will have to previously carry out a study of samples to identify a series of simple tasks using Matlab.

Then we will perform feature extraction and use the Matlab machine learning module to create a module that will help us interpret new samples.

To finish, we will create this interface using the Simulink tool using the previously created module for decision making.

Key words

- EEG.
- Machine Learning.
- Features.
- Bluetooth.
- Simulink.

Índice

Resumen	ix
Abstract	xi
Índice	xiii
Lista de figuras	xv
Glosario	xvii
1 Introducción	1
1.1 Presentación	1
1.2 Motivación	2
1.3 Objetivos	2
1.4 Aplicaciones	2
1.5 Estructuración del proyecto	3
1.6 Estructuración de la memoria	4
2 Interfaces BCI	5
2.1 Introducción	5
2.2 El Cerebro	5
2.3 Arquitectura de la Interfaz Cerebro-Computadora	9
2.3.1 Bloque de adquisición de datos	10
2.3.2 Bloque de procesamiento de datos	12
2.3.3 Bloque clasificador	20
2.3.4 Bloque de comunicación puerto serie	23
2.3.5 Realimentación	24

3 Etapas de desarrollo de nuestra BCI	25
4 Herramientas de trabajo para el diseño de nuestra BCI	27
4.1 Hardware	27
4.2 Software	28
5 Diseño e implementación BCI	29
5.1 Adquisición de datos	29
5.2 Preprocesamiento y extracción de características	30
5.3 Clasificación de señales	33
5.4 Comunicación serie	37
5.5 Realimentación	38
6 Resultados	39
7 Presupuesto	41
8 Conclusiones y trabajos futuros	43
Bibliografía	45
Anexos	47
A.1 Código Procesamiento de ordenes Bluetooth	47
B.1 Código Libreria comunicación serie	49
C.1 Código preprocesamiento y extraccion características Etapa de Uso	51
D.1 Código preprocesamiento y extraccion características Etapa de Diseño	52

Listado de figuras

1.1	EEG <i>Unicorn Hybrid Black</i>	1
2.1	Cerebro. <i>Partes del cerebro</i>	5
2.2	Ritmos principales de señales EEG	8
2.3	Evento de desincronización bandas Mu y Beta	9
2.4	Arquitectura Interfaz Cerebro-Máquina	9
2.5	Amplificador EEG <i>Unicorn Hybrid Black</i>	10
2.6	Conexión de electrodos en la malla <i>Unicorn Hybrid Black</i>	10
2.7	Conexión de electrodos <i>Unicorn Hybrid Black</i>	11
2.8	FFT <i>Transformada Rápida de Fourier</i>	13
2.9	FFT. <i>Imaginación movimiento manos</i>	13
2.10	FFT. <i>Imaginación movimiento pies</i>	14
2.11	Densidad Espectral de Potencia. <i>Imaginación movimiento mano izquierda, ritmos Alfa y Beta</i>	15
2.12	Densidad Espectral de Potencia. <i>Imaginación movimiento mano derecha, ritmos Alfa y Beta</i>	15
2.13	Densidad Espectral de Potencia. <i>Imaginación movimiento pies, ritmos Alfa y Beta</i>	15
2.14	Wavalet <i>Transformada Wavalet</i>	16
2.15	Wavalet <i>Wavalet Packet</i>	16
2.16	Wavalet <i>Descomposición de la señal mediante Wavalet</i>	17
2.17	Transformada de fourier en tiempo corto. <i>Imaginación movimiento mano izquierda</i>	18
2.18	Actividad cerebral.	19
2.19	Inteligencia Artificial.	20
2.20	Machine Learning.	22
2.21	Deep Learning.	22
2.22	Etapas Machine Learning	23

2.23 Bluetooth.	23
3.1 Arquitectura Etapa de calibración y entrenamiento.	25
3.2 Arquitectura Etapa de diseño.	26
3.3 Arquitectura Etapa de uso.	26
4.1 Piezas Unicorn Hybrid Black.	27
4.2 Coche a tracción.	28
5.1 Diagrama de bloques BCI.	29
5.2 Configuración etapa de uso Simulink.	30
5.3 Intención motora mano derecha $t=0$	30
5.4 Intención motora mano derecha $t=1$	31
5.5 Intención motora mano derecha $t=2$	31
5.6 Intención motora mano derecha $t=3$	32
5.7 Bloque de preprocesado y extracción de características.	33
5.8 SubBloque Canal C3.	33
5.9 SubBloque Banda Alfa.	33
5.10 Modelo de Clasificación. Gráfica PPV.	35
5.11 Modelo de Clasificación. Gráfica TRP.	36
5.12 Modelos de Clasificación. Gráfica ROC.	36
5.13 Bloque comunicación serie.	37

Glosario

EEG	Electroencefalograma
BCI	Brain-Computer Interface
PSD	Densidad Espectral de Potencia
SVM	Support Vector Machines
t-SNE	Distributed Stochastic Neighbor Embedding
PCA	Principal Component Analysis
SVD	Singular Value Decomposition
FFT	Fast Transform Fourier
STFT	Short-Time Fourier Transform

Capítulo 1

Introducción

1.1 Presentación

En la actualidad todas las personas hacemos uso de interfaces hombre-máquina para comunicarnos con cualquier dispositivo electrónico. Estas interfaces han ido evolucionando con el tiempo gracias a los avances tecnológicos pasando desde los antiguos paneles de control hasta los paneles táctiles como los que tienen nuestros dispositivos móviles de hoy en día.

Dichos avances tecnológicos en las interfaces han traído consigo unas facilidades para los usuarios, haciendo que sean más sencillas para las personas poder utilizar los dispositivos electrónicos.

Todavía las interfaces hombre-máquina tienen un largo recorrido de estudio e implementación, cuya finalidad es poder comunicarnos con nuestros dispositivos a través ondas cerebrales producidas a través de nuestros pensamientos.

Hoy en día esto está mucho más cerca gracias a los avances tecnológicos de dispositivos amplificadores EEG más pequeños y que toman muestras de mayor calidad que hace unos años. 1.1



Figura 1.1: EEG *Unicorn Hybrid Black*

1.2 Motivación

Este proyecto viene motivado por la necesidad de poder facilitar a los usuarios que tengan dificultades debido a alguna discapacidad, el poder comunicarse con dispositivos electrónicos, los cuales solo tienen como finalidad hacer nuestra vida más cómoda. Además, este proyecto también podrá ser utilizado para proporcionar una interfaz más cómoda de comunicación con los dispositivos electrónicos a las personas que no tengan ningún tipo de discapacidad.

Otra motivación es el impacto social y la proyección que tienen dichos proyectos. En la actualidad estamos nada más que en el principio del desarrollo de dichas interfaces, esto se debe a la complejidad de nuestro cerebro y la variabilidad de dichas muestras entre personas o incluso en la misma persona dependiendo del momento en el que tomemos dichas muestras. Muchas multinacionales como Meta están invirtiendo grandes cantidades en desarrollar estas tecnologías para poder desarrollar metaversos y juegos de realidad virtual lo cual supondría un importante impacto económico y social en nuestro mundo.

Una última importante motivación es la de ser capaz de desarrollar una interfaz Cerebro-Computadora, lo cual considero como una adquisición de conocimientos que podrían ser relevantes para un desarrollo profesional en el futuro debido al impacto que van a tener dichas tecnologías.

1.3 Objetivos

Como hemos indicado, la finalidad del estudio de las interfaces hombre-maquina es poder comunicarnos con nuestros dispositivos electrónicos a través de las ondas cerebrales producidas por nuestros pensamientos.

Por lo tanto, el primer objetivo de nuestro proyecto es poder diseñar una interfaz BCI capaz de procesar una serie de tareas sencillas de imaginación motora, las cuales serían mover la mano izquierda, mover la mano derecha y mover los pies. Dichas tareas sencillas serán utilizadas para poder mover un aparato motorizado hacia la derecha, hacia la izquierda y hacia delante.

Como segundo objetivo de nuestro proyecto es que nuestro diseño se pueda utilizar para poder implementar nuevas tareas sencillas y tareas más complejas en un futuro.

1.4 Aplicaciones

Como ya hemos indicado los campos de aplicaciones de dicha tecnología son muy amplios como la utilización de nuestra interfaz por personas discapacitadas que no pueden comunicarse con los dispositivos electrónicos debido al tipo de interacción que conllevan las interfaces actuales. Otro campo de aplicación es la utilización de nuestro diseño por empresas que están desarrollando metaversos o juegos de realidad virtual, cuya funcionalidad sería realizar el movimiento de un avatar.

1.5 Estructuración del proyecto

En este apartado presentaremos el flujo de trabajo que se ha seguido para el desarrollo de proyecto, debido a la realización del flujo del trabajo, tendremos una mejor organización y una mayor productividad, realizando un trabajo más eficiente. Las etapas serán las siguientes:

- **Etapa 1 - Preparación:** En la primera etapa de nuestro proyecto realizaremos la instalación del software de nuestro amplificador EEG *Unicorn Hybrid Black*, de Matlab y Simulink. En esta etapa también comprobaremos la conexión exitosa entre nuestra CPU y el amplificador EEG *Unicorn Hybrid Black*.
- **Etapa 2 - Estudios y formación previos a la realización del nuestra interfaz:** Se realizará un estudio sobre el funcionamiento, la obtención de muestras y patrones de ondas cerebrales mediante electroencefalogramas, y su extracción de características que deben contener tres tareas simples. Se realizará un estudio sobre el funcionamiento del *Unicorn Hybrid Black* para poder realizar un trabajo óptimo.

Se realizará un estudio sobre los fundamentos del *Machine Learning y Deep Learning* para poder utilizarlos en el desarrollo de nuestra BCI mediante las API's correspondientes en Matlab.

Se realizará estudio sobre la integración entre el amplificador EEG y la CPU para poder utilizar nuestro diseño.

- **Etapa 3 - Toma de muestras y estudio de banco de datos:** Realizaremos una toma de muestras mediante nuestro amplificador EEG *Unicorn Hybrid Black*. La toma de muestras corresponderá con las siguientes tareas sencillas :

- Mover mano Izquierda.
- Mover mano Derecha.
- Mover pies.

Con dichas tareas interpretaremos las siguientes acciones, girar izquierda, girar derecha y movernos adelante.

También analizaremos las muestras de nuestros 2 bancos de datos restantes que corresponderán con la base de datos de la UAH[10] y la base de datos de Physionet [23].

- **Etapa 4 - Procesamiento de las muestras y diseño de Interfaz:** Se realizará el procesamiento de las muestras de las tareas sencillas de nuestros bancos de datos mediante Matlab.

Se procederá a Diseñar la interfaz BCI a través de módulos mediante bloques en Simulink.

- **Etapa 5 - Redacción de la memoria:** En la última etapa redactaremos la memoria con todos los contenidos desarrollados, conclusiones y las líneas futuras a seguir en proyectos de temática similar.

1.6 Estructuración de la memoria

En este apartado hacemos una explicación de cómo hemos estructurado la memoria, para poder ofrecer una mayor compresión del proyecto y entender los fundamentos teóricos en la que nos basamos para realizar el diseño de dicha interfaz Cerebro-Computadora.

- **Bloque de teoría**

En dicho bloque abordaremos el funcionamiento del cerebro, las características que podemos observar a través de señales EEG, para identificar ciertos patrones que se pueden asociar a un número de tareas sencillas. Posteriormente abordaremos las etapas que conforman el diseño de una BCI y cuál será la arquitectura que conforma dicha BCI. También abordaremos el funcionamiento del machine learning que utilizaremos para poder diseñar nuestra BCI y como extraer las características adecuadas de las muestras para obtener un modelo más preciso a la hora de clasificar las señales de las tareas que está realizando nuestro cerebro.

- **Bloque práctico**

En este bloque se explicará como hemos realizado nuestro diseño , como lo hemos implementado, los resultados obtenidos analizando nuestros bancos de muestras y como hemos tratado dichas muestras, finalizando con un debate de dichos resultados.

- **Conclusión**

Para finalizar la memoria realizaremos una conclusión sobre el proyecto y vistas a futuro, y el impacto dichas tecnologías pueden tener en la sociedad.

Capítulo 2

Interfaces BCI

2.1 Introducción

Durante el desarrollo de este capítulo hablaremos sobre las características de las señales que emite el cerebro sobre el cuero cabelludo que son recolectadas mediante un amplificador EEG. También explicaremos como podremos utilizar dichas características para poder interpretar ciertas tareas sencillas y desarrollar una BCI[1]. Por último, explicaremos que arquitectura compone una BCI y hablaremos detalladamente sobre la parte de clasificación que utilizaremos a partir del uso del machine learning.

2.2 El Cerebro

Para poder entender el funcionamiento de la interfaz, tenemos que inicialmente entender cuál es el medio por el que nos vamos a comunicar con la máquina, que son las ondas y patrones cerebrales, por lo tanto, tenemos que entender el funcionamiento del cerebro.

El cerebro 2.1 es el órgano más complejo de nuestro sistema ya que es el que conforma nuestra conciencia. El cerebro está conformado por dos hemisferios separados por un surco central, y a su vez cada hemisferio está dividido en cuatro lóbulos: frontal, parietal temporal y occipital.

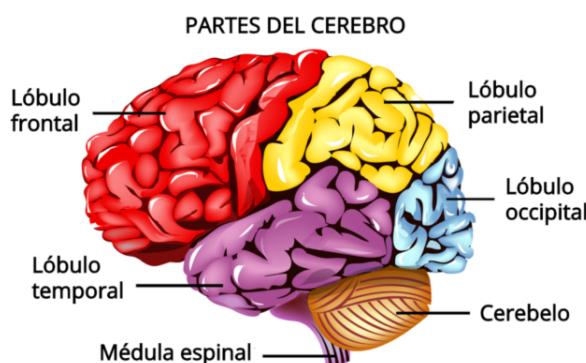


Figura 2.1: Cerebro. *Partes del cerebro*

El cerebro este compuesto por varios billones de células de las cuales 100000 millones son neuras y casi 100 trillones de conexiones entre neuronas, lo cual conforma la estructura para su funcionamiento.

La neurona es una unidad anatómica y funcional independiente, formada por un cuerpo celular del que salen numerosas ramificaciones denominadas dendritas, capaces de recibir información de otras células nerviosas, y de una prolongación principal, o axón, que conduce la información hacia otras neuronas.

Durante los últimos 50 años se han producido unos importantes avances científicos, donde podemos llegar a una serie de conclusiones:[10]

- Se han delimitado las distintas áreas de la corteza cerebral especializadas en recibir y procesar las informaciones sensoriales y controlar las reacciones musculares. Se sabe que el lóbulo occipital está especializado en el tratamiento de la información visual, la región temporal se encarga de ciertas sensaciones visuales y auditivas, los lóbulos frontales están íntimamente relacionados con la actividad motora y los lóbulos parietales con los sentidos del tacto y del gusto.

Sin embargo, estas áreas especializadas representan apenas una cuarta parte de la corteza cerebral; el resto serán denominadas áreas de asociación, no cumplen ninguna función específica y parecen estar encargadas de interpretar, integrar y coordinar las informaciones recogidas y procesadas por las áreas sensoriales y motoras. Las áreas de asociación serían, por tanto, las responsables de las funciones mentales superiores como el lenguaje, pensamiento, razonamiento, memoria, creatividad, etc.

En nuestra BCI nosotros queremos poder interpretar tareas sencillas de imaginación motora, por lo tanto, tal y como hemos indicado será el lóbulo frontal el encargado de realizar dichas tareas. Más adelante estudiaremos y explicaremos que área o áreas del lóbulo frontal trabajarán en función de la tarea que se realice. Esta información nosotros la recolectamos mediante los electrodos de un amplificador EEG situado en las zonas de trabajo, por lo que los electrodos a estudiar dependerán del lóbulo y la zona de dicho lóbulo que realice la tarea pertinente.

- Cada uno de los hemisferios controla y ejecuta funciones diferentes o aspectos diferentes de una misma función. En términos generales, parece que, en la mayor parte de las personas, el hemisferio izquierdo controla la habilidad lingüística, numérica y de pensamiento analítico, mientras que el hemisferio derecho controla las habilidades especiales complejas, como la percepción de patrones y aspectos de la ejecución artística y musical.

Sin embargo, las actividades complejas requieren la interrelación de los dos hemisferios. Así, por ejemplo, cuando leemos un relato, el hemisferio izquierdo entiende el significado de las palabras, pero es el derecho el que capta el contenido emotivo y las imágenes utilizadas.

En nuestra BCI nosotros podremos observar en las muestras adquiridas mediante nuestro EEG que dependiendo de la tarea trabajara una zona u otra.

- Por otra parte, hay muchas funciones, principalmente de las áreas motoras y sensoriales, que parecen idénticas en ambos hemisferios. En definitiva, hay una especialización funcional, pero la actividad conjunta de ambos hemisferios es necesaria para el funcionamiento integral del cerebro.
- En base a un número determinado de estudios se ha llegado a la conclusión que los registros eléctricos observados mediante un EEG demuestran una actividad continua del cerebro por lo que

podemos deducir que la actividad eléctrica no se produce solamente mediante un estímulo, sino que es permanente, y que dichos estímulos solo producen que tanto la intensidad como los patrones varíen.

Para poder interpretar las diferentes tareas nosotros estudiaremos dichos patrones, estos patrones a su vez serán los que serán clasificados por nuestra BCI para reconocer la tarea.

La actividad del cerebro podremos representarlas mediante unas señales denominadas ritmos 2.2 con unas características en la amplitud y en la frecuencia distintas en función de la actividad mental. Los ritmos más importantes serán los siguientes:

- Ritmo Alfa: Es una actividad eléctrica con una frecuencia situada entre 8 y 13 Hz presente en estado de vigilia en las regiones posteriores de la corteza cerebral relacionadas, entre otros, con el sentido de la vista. Este ritmo aparece en el EEG normal cuando los sujetos están en reposo, relajados y con los ojos cerrados. Con un grado de reactividad variable, esta oscilación se atenúa, o incluso llega a desaparecer completamente dependiendo de los sujetos, con la actividad mental o la apertura de los párpados.

En nuestro interfaz BCI observaremos esta banda para ver las características del estado de relajación.

- Ritmo Mu: Aparece en sujetos normales con una frecuencia situada en la banda alfa pero independiente del ritmo alfa por su topografía, menos sinusoidal que dicho ritmo, y su reactividad, que se observa vinculada a los sistemas sensorial y motor.^[7] Se ha demostrado en algunos estudios que veremos más adelante, que algunos de estos ritmos mu están estrechamente ligados con algunos ritmos beta como se explicará posteriormente.

En nuestro interfaz BCI nosotros analizaremos este ritmo para interpretar las tareas de imaginación motora.

- Ritmo Beta: oscilación de frecuencia comprendida entre 14 y 32 Hz que se asocia con momentos de actividad mental. Su amplitud es considerablemente más pequeña que los ritmos anteriores (5-10 μ V respecto a los 20-60 μ V de los ritmos alfa y mu) y aparece generalmente como actividad dominante en las regiones anteriores y centrales.

En nuestro interfaz BCI nosotros analizaremos este ritmo en las frecuencias bajas comprendidas entre los 14-18 Hz para poder interpretar las tareas de imaginación motora.

- Ritmo Gamma: oscilación de frecuencia comprendida entre 30 y 90 Hz que, los estudios más recientes, relacionan con el proceso de percepción consciente, con estímulos sensoriales y algunos procesos internos tales como memoria y la atención.
- Ritmo Theta: con una frecuencia situada entre 4 y 7 Hz es característico en niños o adultos en estado de estrés, frustración o desencanto. Aparece normalmente en la región parietal o temporal.
- Ritmo Delta: de frecuencia menor a 3,5 Hz, está presente en el EEG de lactantes o en adultos durante estados de sueño profundo.

También existen otros ritmos como Lambda e HyperGamma, se sabe menos acerca de estos estados debido a la rareza de la observación de ellos. Se han asociado a estados de “animación suspendida” donde los latidos del corazón, la respiración o el pulso dejan de ser perceptibles.

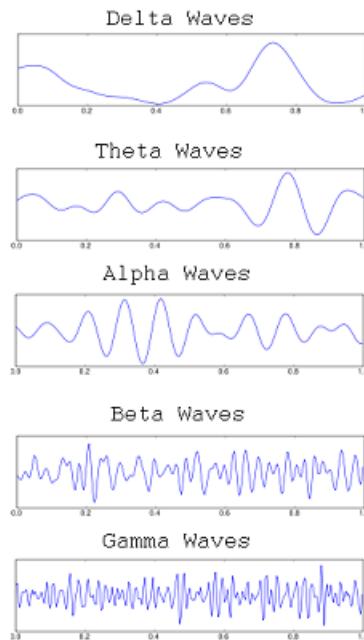


Figura 2.2: Ritmos principales de señales EEG

Los primeros estudios EEG empezaron en 1912 con Vladimir Vladimirovich Pravdich-Neminsky publicando el primer estudio sobre EEG y potenciales evocados de perros. Hans Berger (1873-1941) comenzó sus estudios sobre electroencefalografía en humanos en 1920.

Hoy en día a pesar de todos los avances y de todos los estudios e investigaciones no estamos nada más que en los primeros pasos de entender el funcionamiento del cerebro.

Como hemos indicado previamente en nuestra BCI nosotros queremos poder estudiar y captar las ondas y patrones referentes a la intención motora tanto de las manos como de los pies. Dichas tareas serán representadas a través del lóbulo frontal. Dependiendo de la acción que queramos realizar, se producirá una cierta desincronización en algunas partes del lóbulo frontal que serán recogidas por nuestra BCI, la cual se encargara de detectar las siguientes tareas:

- La primera tarea consiste en poder realizar la imaginación motora de la mano derecha cuyo movimiento podremos observarlo porque se produce una desincronización de las ondas cerebrales en el lóbulo frontal izquierdo.
- La segunda tarea consiste en poder realizar la imaginación motora de la mano izquierda cuyo movimiento podremos observarlo porque se produce una desincronización de las ondas cerebrales en el lóbulo frontal derecho.
- La tercera tarea consiste en poder realizar la imaginación motora de los pies cuyo movimiento podremos observarlo porque se produce una desincronización de las ondas cerebrales en el lóbulo frontal en su parte central.

Dichas desincronizaciones 2.3 se producirá en los ritmos mu y beta con los cuales podremos identificar la intención motora y serán recolectadas mediante un amplificador EEG. Mas adelante detallaremos como podremos observar dichas desincronizaciones mediante el procesamiento de nuestras muestras recolectadas.

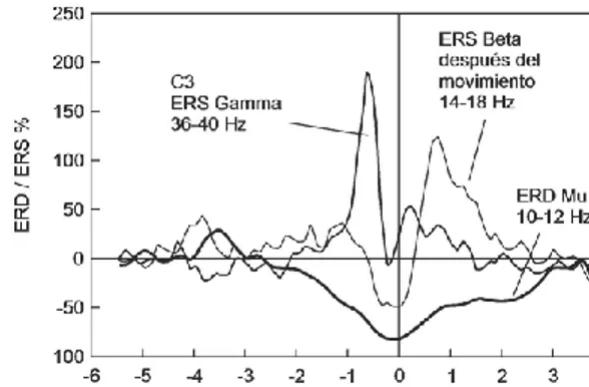


Figura 2.3: Evento de desincronización bandas Mu y Beta

2.3 Arquitectura de la Interfaz Cerebro-Computadora

En nuestro proyecto nosotros haremos uso de diseño BCI donde el método de adquisición será un método no invasivo, realizando dicha adquisición a través de electrodos colocados sobre nuestro cuero cabelludo. La arquitectura de nuestra BCI 2.4 estará compuesta por una serie bloques y subbloques que serán explicados en este capítulo detalladamente.

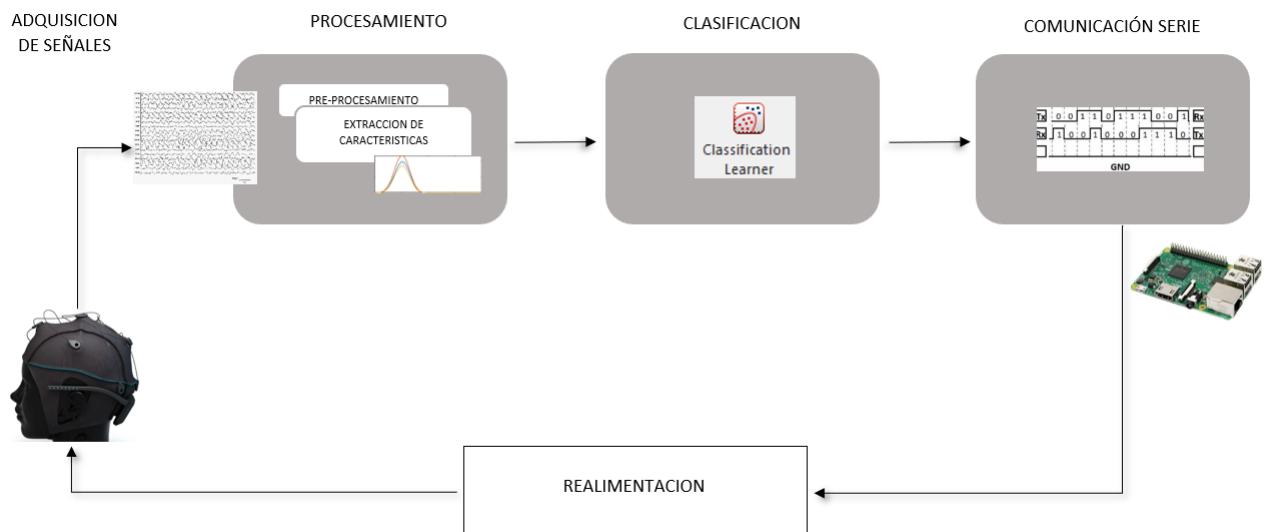


Figura 2.4: Arquitectura Interfaz Cerebro-Máquina

2.3.1 Bloque de adquisición de datos

Nuestra BCI está compuesta inicialmente por un bloque de adquisición, el cual será el encargado de recolectar las muestras para después poder analizarlas y procesarlas. Como hemos indicado previamente para la toma de muestras nosotros haremos uso de un método no invasivo a través de electrodos, dicha adquisición de datos se realizará con un amplificador EEG [2] en nuestro caso utilizaremos el amplificador Unicorn Hybrid Black. 2.5. Dicho amplificador esta compuesto por ocho canales de adquisición de registros EEG's, los cuales se corresponden con el mismo número de electrodos, tres canales de acelerómetro y tres canales de giróscopo. Este sistema se ha de montar en una malla específica de acuerdo a la metodología indicada por el fabricante [9] y que se explicará a continuación.



Figura 2.5: Amplificador EEG *Unicorn Hybrid Black*.

Durante este apartado se especifican los siguientes pasos que se han de seguir para montar el dispositivo Unicorn Hybrid Black:

- Montaje de los electrodos en la malla según la figura 2.6.
- Colocación de la malla en la cabeza del sujeto.
- Colocación de los electrodos R y L en los lóbulos de las orejas.
- Introducción del gel por los orificios de los electrodos.
- Presionar botón de encendido trasero.

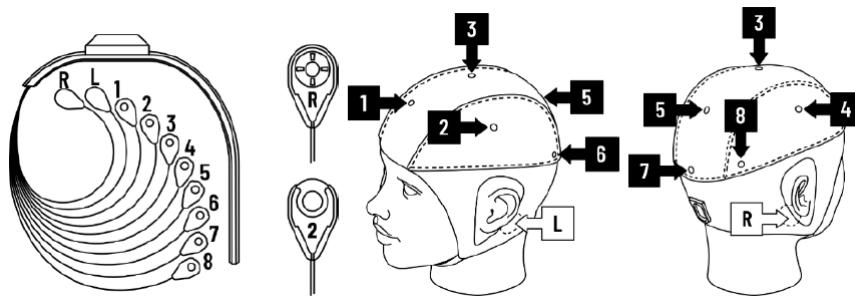


Figura 2.6: Conexión de electrodos en la malla *Unicorn Hybrid Black*.

Para nuestra BCI solo será necesario el estudio de los electrodos 2,3 y 4 que determinaremos a partir de ahora como C3, Cz y C4, dichos canales son los correspondientes a los electrodos colocados en el lóbulo frontal que sería el encargado de realizar la imaginación motora como hemos indicado previamente.

Para que la adquisición sea adecuada es importante que el usuario siempre tenga una sesión de entrenamiento previo y que se encuentre en estado de relajación. Este entrenamiento producirá que las muestras correspondientes a las tareas a realizar sean correctas. También es imprescindible que el sensor tenga la cantidad adecuada de gel para reducir la impedancia y que la adquisición sea correcta. Por último el software de Unicorn contiene una API que nos indicara si los electrodos están recolectando muestras de manera correcta, poniéndose de color amarillo en caso negativo como podemos ver en la figura 2.7.



Figura 2.7: Conexión de electrodos *Unicorn Hybrid Black*.

En nuestro diseño será imprescindible que los electrodos 2, 3 y 4 estén siempre bien conectados, dichos electrodos serán los que nos proporcionaran la información sobre las tareas que queremos realizar.

También cabe destacar que aunque se haya realizado un entrenamiento previo, es posible que las muestras recolectadas de las tareas en la adquisición no sean todas validas ya que, el cerebro humano cuando aprende a realizar una actividad necesita una realimentación para que el aprendizaje sea correcto y durante la adquisición inicial al desarrollo de la BCI no veremos esta realimentación producirse.

2.3.2 Bloque de procesamiento de datos

Este bloque será el encargado de procesar las muestras y de extraer las características necesarias para que el bloque clasificador pueda elegir a que acción corresponde dichos datos. Este bloque estará compuesto por dos subbloques.

Preprocesamiento

Este subbloque será el encargado de realizar una serie de operaciones de tratado de señales que han sido adquiridas previamente mediante el amplificador EEG *Unicorn Hybrid Black* correspondiente con la figura 2.5.

En primer lugar, en la etapa de Preprocesamiento de todo sistema BCI, se realizará un filtro Notch o de muesca a 50 Hz o 60Hz para eliminar el ruido eléctrico, la frecuencia de dicho filtro dependerá del país en el que nos encontremos, actualmente existen amplificadores EEG que ya contienen este filtro en la etapa de adquisición.

En segundo lugar, realizaremos un filtro paso banda para observar las señales en las frecuencias que deseamos, las bandas de frecuencias más importantes serán las que corresponden con los ritmos Alfa,Mu,Beta,Gamman,Theta y Delta. En nuestro sistema BCI solo observaremos las bandas de frecuencia correspondientes a los ritmos Alfa,Mu y Beta, ya que en dichos ritmos podemos visualizar las características de la imaginación motora.[6]

Después de realizar dichos filtrados es complicado poder visualizar características en el dominio temporal por lo que, en base a una serie de estudios, se presentarán a continuación varias propuestas distintas para el preprocesado, desde técnicas más sencillas a otras más complejas y se explorarán sus fortalezas y debilidades para el tratamiento de señales EEG, finalmente mezclaremos diferentes propuestas para realizar la extracción de características más óptima.

Como hemos indicado previamente la imaginación de un movimiento provoca la aparición o desaparición de patrones frecuenciales conocidos en las bandas correspondientes con los ritmos Mu y Beta, por lo que se propone la utilización de la transformada de Fourier como herramienta inicial de extracción de características. Concretamente haremos uso del cálculo de la densidad espectral de potencia (PSD) mediante el método Welch's para poder visualizar de manera más sencilla la información.

Debido a las limitaciones de esta transformada, se estudian otras técnicas que permiten extraer la información frecuencial y temporal de la señal. Más concretamente, se analizará en profundidad la transformada wavelet y su posible aplicación combinada con la transformada de Fourier para potenciar las componentes frecuenciales más relevantes y reducir el número de coeficientes empleados.

Por último, haremos uso de la transformada de Fourier de corta duración para poder observar la respuesta en frecuencia y en el tiempo de manera tridimensional.

Para poder visualizar dichas características y la actividad en el cerebro también hacemos uso de la herramienta EEGLAB de Matlab la cual se encarga de realizar una combinación de los estudios anteriores y representarlo en gráficas.

A continuación, explicaremos los métodos que utilizaremos en nuestro diseño:

- **Transformada Rápida de Fourier:** La *Transformada Rápida de Fourier* 2.8 se utiliza para visualizar la distribución de la amplitud de espectro de las señales EEG y extraer los picos del espectro para reflejar las diferentes tareas del cerebro (Zhang, Yang, Huang, 2008) [11]. Mediante el uso de la FFT, la señal de EEG puede mapearse desde el dominio del tiempo al dominio de la frecuencia. El espectro de frecuencia de la señal es reconocido por descomponer la señal en su correspondiente sinusoidal de diferentes frecuencias. Según el estudio de (S. Huang Wu, 2010) [12] la transformada de Fourier puede ser utilizada para la extracción de características en la clasificación de tareas motoras.

$$X(k) = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} nk} \quad \text{con } k = 0, 1, \dots, N-1$$

Figura 2.8: FFT *Transformada Rápida de Fourier*.

La FFT es fácil y rápida de calcular. Sin embargo, la FFT tiene sus desventajas:

- Las frecuencias utilizadas para descomponer una señal son una función de la frecuencia de muestreo de la señal y del número de frecuencias deseadas. Sin modificar estos dos parámetros, estas frecuencias no son seleccionables.
- Una onda sinusoidal simple cuya frecuencia no cae en una de las frecuencias de la transformada producirá un espectro con dispersión de energía para muchas frecuencias.

Como hemos dicho según los estudios podríamos ver una serie de patrones y características visualizando las amplitudes y los picos. En nuestro estudio para ver la imaginación motora observaremos los Ritmos Mu y Beta.

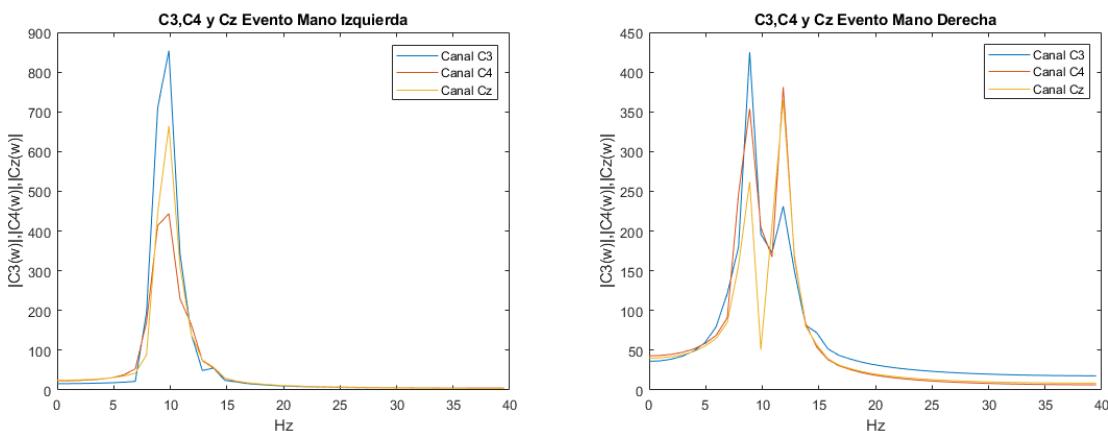


Figura 2.9: FFT. *Imaginación movimiento manos.*

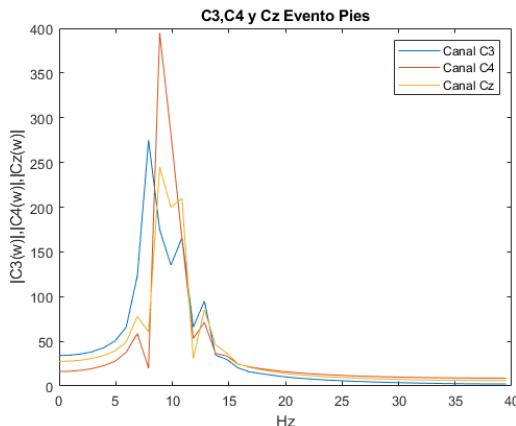


Figura 2.10: FFT. *Imaginación movimiento pies.*

Como podemos observar en las figuras 2.9 y 2.10 cuando imaginamos un movimiento motor de manos o pies se produce una desincronización en la zona de trabajo, lo cual produce un pico de menor amplitud que en las zonas no desincronizadas, estas amplitudes se podrán observar mejor si visualizamos mediante la PSD. Para extraer más características debemos hacer un análisis un poco más exhaustivo, extrayendo mejores características mediante un estudio simultáneo en tiempo y frecuencia. Por lo que tambiénaremos uso de la transformada wavelet.

- **Densidad Espectral de Potencia:** La *Densidad Espectral de Potencia* informa sobre la distribución de la potencia de una señal entre sus diferentes frecuencias. Las características de la PSD pueden ser calculadas, por ejemplo, elevando al cuadrado la transformada de Fourier de una señal, o mediante el cálculo de la transformada de Fourier de la función de autocorrelación de la señal. Las características de la PSD son probablemente las más utilizadas para el diseño de BCI, y han demostrado ser eficaces para el reconocimiento de un gran número de señales neurofisiológicas.

Aplicaciones de PSD en sistemas BCI la encontramos en (Huaiyu Xu, Jian Lou, Ruidan Su, Erpeng Zhang, 2009) [13], donde combinan PSD con los coeficientes wavelet de la señal EEG para caracterizar una señal EEG, encontrando promedios de precisión hasta del 82 % en su clasificación, y otro estudio (Palaniappan, Nae-Jen) [14] que calcula la PSD usando la energía del filtro FIR (Finite Impulso response) elíptico para obtener precisiones de clasificación hasta el 97 %, usando un clasificador de red neuronal. Por lo tanto según varios estudios cada vez que tenemos una intención motora se produce una desincronización 2.11, 2.12 y 2.13 en el ritmo alfa (8-13 Hz) la cual produce que la zona de trabajo del cerebro tenga una menor amplitud que el resto de zonas, por otro lado esta desincronización provocara en la zona baja de la banda Beta(14-18Hz) un incremento de amplitud respecto a la ritmo alfa y mayor que en las otras zonas del ritmo Beta. Nosotros haremos uso del método de Welch's para poder visualizar la PSD y sus características.

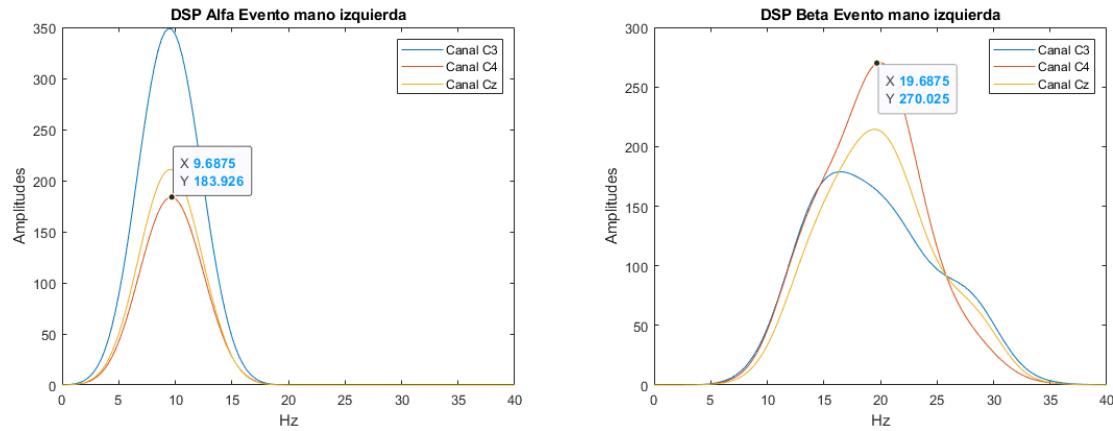


Figura 2.11: Densidad Espectral de Potencia. *Imaginación movimiento mano izquierda, ritmos Alfa y Beta.*

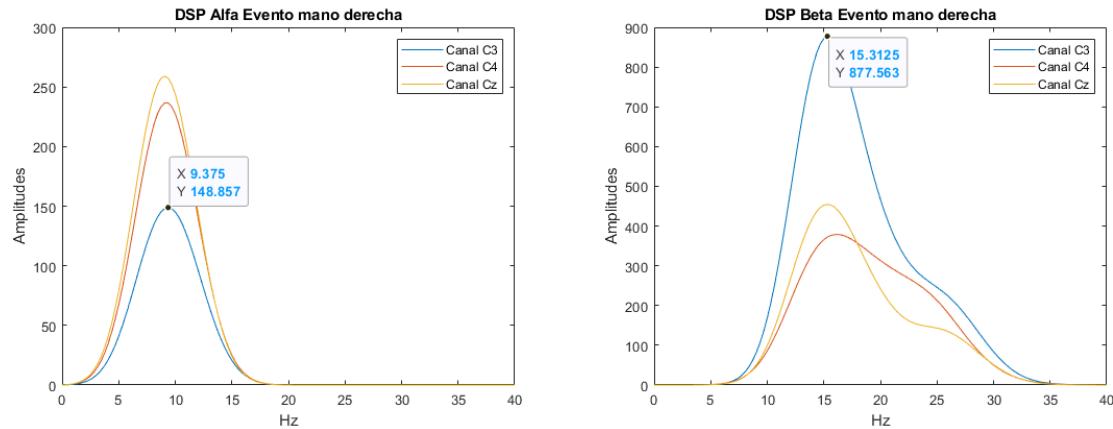


Figura 2.12: Densidad Espectral de Potencia. *Imaginación movimiento mano derecha, ritmos Alfa y Beta.*

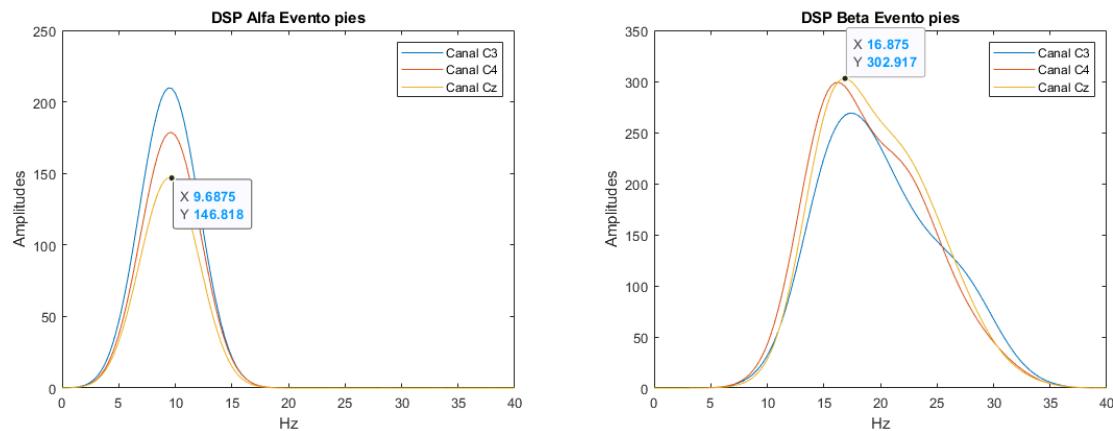


Figura 2.13: Densidad Espectral de Potencia. *Imaginación movimiento pies, ritmos Alfa y Beta.*

- **Transformada de Wavalet:** Múltiples investigadores han aplicado análisis wavelet para caracterizar señales EEG para la imaginación motora en sistemas BCI (Ming-Ai, Rui, Dong-Mei, Jin-Fu, 2009) [15]; (Zou, Wang, Shi, Ma, 2010) [16], cuyos resultados de clasificación, eligiendo una wavelet madre específica y un nivel de descomposición adecuado, mejoran respecto a otras técnicas como AAR, descomposición en subespacios espaciales comunes (CSSD) y entropía aproximada (ApEn). Investigaciones han señalado que la wavelet madre Daubechies es una más adecuada para el análisis de datos de EEG respecto a otras wavelet madre para clasificar características orientadas a la imaginación motora (Adeli, Zhou, Dadmehr, 2003) [17]. Una generalización de la transformada wavelet es la Wavelet Packet, cuya diferencia radica en la descomposición de la señal: en el primer nivel, la señal se descompone en coeficientes de aproximación y de detalle igual que en wavelets, pero la descomposición de los niveles adicionales se hace tanto en los componentes de aproximación como en los de detalle [18]. Resultados en EEG (Qiao, Wang, Li, Lifeng Tian, 2010) [19] han demostrado que el método propuesto con precisión podría extraer características sustanciales EEG y proporcionar un medio eficaz para clasificar las tareas mentales motoras. La Transformada Wavalet puede proporcionar una descomposición de las señales en tiempo-frecuencia multinivel. Esto permite el uso simultáneo de intervalos largos de tiempo para la información de baja frecuencia e intervalos cortos de tiempo para información de alta frecuencia (Ting, Guo-zheng, Banghua, Hong, 2008) [20]. Uno de los inconvenientes de La transformada wavelet es el incremento de los requisitos de memoria debido a su algoritmo basado en la convolución de filtros. La adecuada selección de una wavelet madre y el número de niveles de descomposición son muy importantes en el análisis de señales de EEG para encontrar niveles de clasificación promisorios (Hu, Li, Chen, 2011) [21].

$$W_f(s, \tau) = \int f(t) \Psi_{s,\tau}(t) dt; \quad \Psi_{s,\tau}(t) = \frac{1}{\sqrt{s}} \Psi\left(\frac{t-\tau}{s}\right)$$

Figura 2.14: Wavalet *Transformada Wavalet*.

Por lo tanto, podremos decir que la transformada Wavalet es un conjunto de filtros Paso Alto, Paso Bajo y Downsamples, y sera utilizado para facilitar la extracción de características de señales EEG.

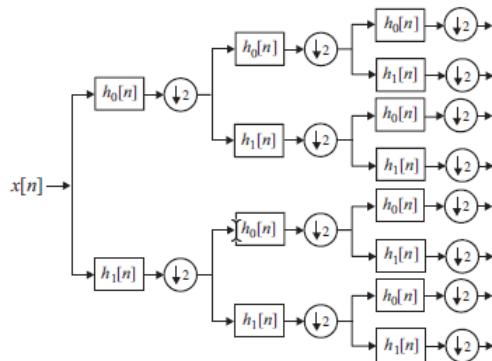


Figura 2.15: Wavalet *Wavelet Packet*.

Nosotrosaremos uso de la descomposición wavalet para poder observar mejor las señales en el dominio temporal en las frecuencias que deseamos y también para reducir el n° de coeficientes a la hora de realizar la extracción de características.

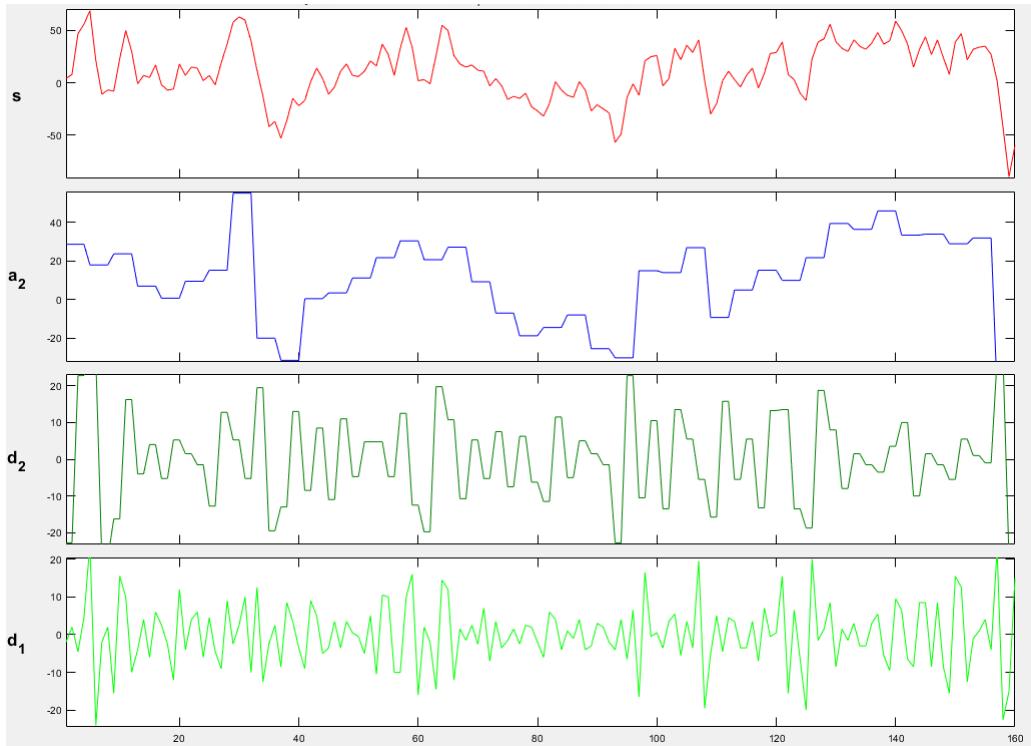


Figura 2.16: Wavalet Descomposición de la señal mediante Wavalet.

- **Transformada de Fourier en tiempo corto:** Las funciones de Fourier no son adecuadas para representar señales no estacionarias cuyo contenido espectral varía con el tiempo. Para tener una representación de la señal no estacionaria, la STFT segmenta la señal en intervalos definidos a través del tiempo, de tal forma que se pueda asumir que la señal es estacionaria para cada segmento, y por cada porción de la señal aplicar la transformada de Fourier. La división de la señal se realiza mediante una función ventana de tiempo $w(t)$ cuyo ancho representa la longitud del segmento.

Obsérvese que los segmentos más largos pueden violar la suposición de cuasi estacionariedad requerida por la transformada de Fourier, por lo tanto, se deben tener en cuenta las resoluciones de tiempo y frecuencia además de la estacionariedad de la señal antes del análisis (Motamedi-Fakhr, Moshrefi-Torbat, Hill, Hill, White, 2014) [22]. La STFT ha sido ampliamente utilizada en el análisis del EEG del sueño debido principalmente a su simplicidad y facilidad de implementación.

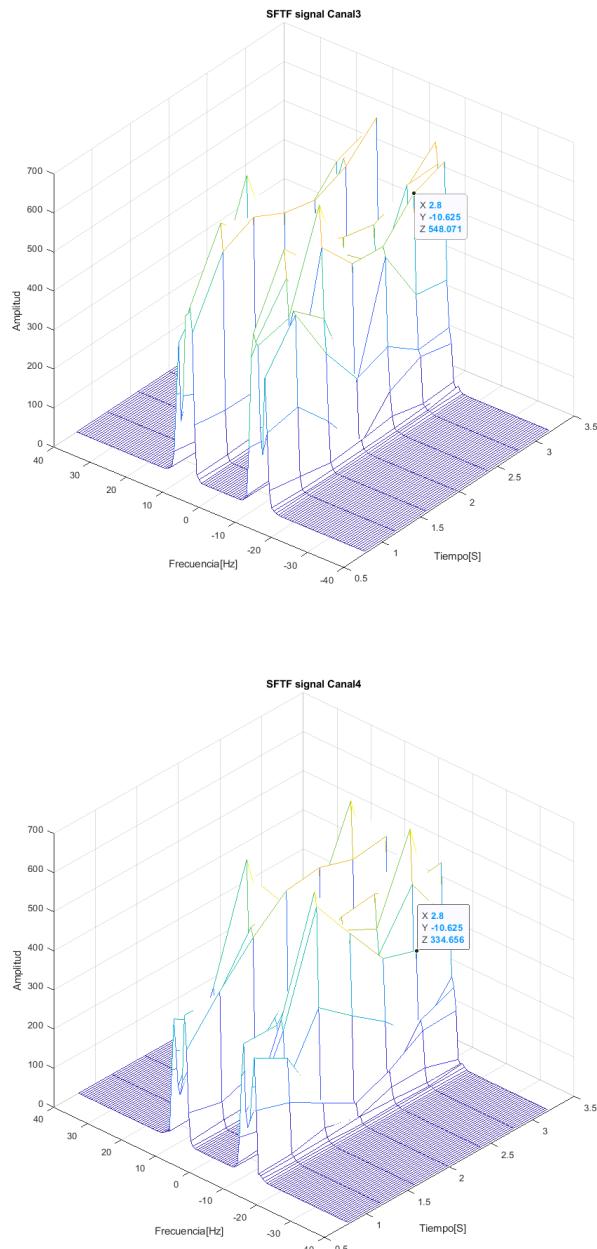


Figura 2.17: Transformada de fourier en tiempo corto. *Imaginación movimiento mano izquierda.*

Como podemos observar en la figura 2.17 podemos ver con más facilidad en qué momento del evento se produce la tarea es decir la desincronización.

Con el estudio de la SFTF llegamos a la conclusión que, aunque queramos realizar una tarea de intención motora de una de las manos no se produce durante todo el tiempo sino que durante un instante o durante unos instantes, por este motivo para realizar la fase de procesamiento de datos decidimos observar segundo a segundo y captar el momento en el que se produce dicho evento.

- **EEGLAB** Con esta herramienta de matlab podemos visualizar y ver la actividad del cerebro durante los eventos.

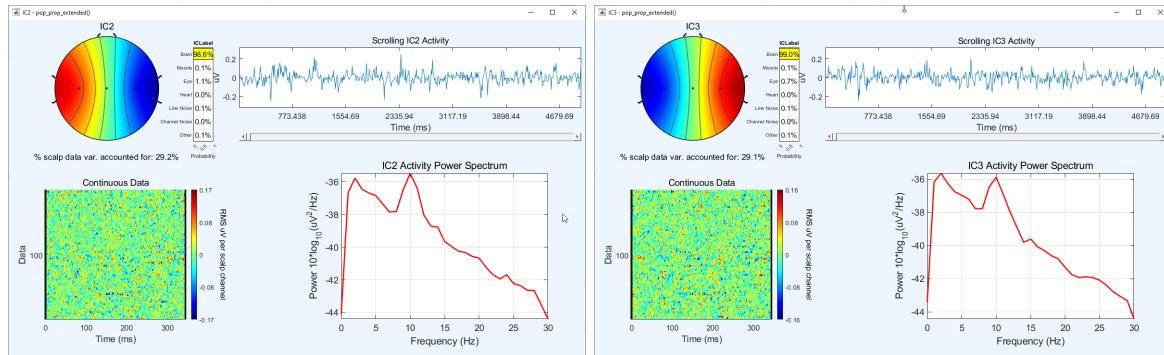


Figura 2.18: Actividad cerebral.

Como hemos podido ver las señales de EEG se pueden utilizar eficazmente para estudiar la actividad cerebral en personas y detectar estados mentales. Teniendo en cuenta que las señales de EEG son de naturaleza altamente no lineal y sus interpretaciones pueden ser propensas a variaciones entre observaciones, hemos recurrido a distintas técnicas para su análisis mediante las técnicas más representativas para extraer características de interés en las EEG, que permitan el reconocimiento de la actividad neuronal relacionada a la intención del sujeto en la imaginación de movimientos, especialmente de la mano derecha e izquierda y de los pies.

Extracción de características

En este subbloque realizaremos la extracción de características de las señales ya preprocesadas en el subbloque anterior. En nuestro estudio hemos podido observar que cuando tenemos una intención motora se produce una desincronización que en la frecuencia produce una onda con menos amplitud en el área que trabaja en dicha acción, también como hemos visto después de esa tarea se produce un incremento en la amplitud en la zona baja de la banda Beta comprendida entre los 14 y los 18 Hz.

Por lo tanto, basándonos en esta característica extraeremos el valor máximo de la PSD de las bandas correspondientes a los ritmos Mu y Beta(zona baja comprendida entre 14 -18 Hz) para identificar la tarea en la cual nos encontramos. Basándonos en otros estudios también extraemos y utilizamos otras características como son la media, la varianza, la covarianza, la desviación estándar y los valores máximo y mínimo en dominio temporal y en el dominio de la frecuencia para poder dar una mayor cantidad de características que pueda analizar nuestro bloque clasificador.

2.3.3 Bloque clasificador

Este bloque será el encargado de realizar la clasificación de la tarea que se está realizando en función de las muestras que el bloque va recibiendo del bloque de procesado.

Dicho bloque clasificador está compuesto por un modelo creado en el ámbito de la inteligencia artificial 2.19 y que puede ser del tipo Machine Learning o Deep Learning. .

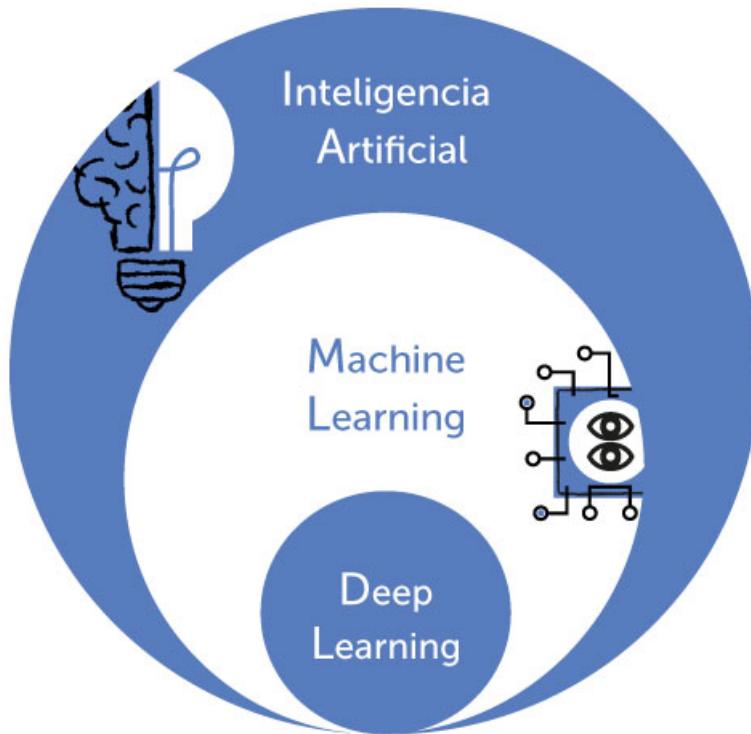


Figura 2.19: Inteligencia Artificial.

- **Machine Learning:** Es una disciplina del ámbito de la Inteligencia Artificial que crea sistemas que aprenden automáticamente pasando una serie de características. El aprendizaje es básicamente la capacidad de poder identificar patrones complejos en una gran cantidad de datos. La máquina es un algoritmo que revisa los datos y es capaz de predecir comportamientos futuros. En este contexto, implica que estos sistemas se pueden mejorar de forma autónoma con el tiempo, sin intervención humana. [3]

Dentro del machine Learning podemos identificar entre varios tipos:

- **Aprendizaje Supervisado:** Como bien indica su nombre, el aprendizaje es supervisado. Nosotros sabremos qué tipo de resultados deseamos obtener y se encuentran relacionados con los datos que vamos a usar.

El principal objetivo es construir un modelo que sea capaz de realizar predicciones con datos nuevos o no conocidos a partir de un conjunto de datos cuyos resultados ya conocemos.

Estos machine learning pueden resolver mediante 2 metodologías:

- * **Classification Learning:** El esquema de aprendizaje se presenta con un conjunto de ejemplos clasificados a partir de los cuales se espera aprender una forma de clasificar

ejemplos no vistos. En el aprendizaje por asociación, se busca cualquier asociación entre características, no solo aquellas que predicen un valor de clase particular.

- * **Regression Learning:** Cuyo objetivo es establecer un método para la relación entre un cierto número de características y una variable objetivo continua. En este sentido, el ejemplo más común es la regresión lineal.

Dentro de las dos metodologías encontraremos una serie de algoritmos de resolución que serán los siguientes.

- * Linear Regression.
- * Logistical Regression.
- * Random Forest.
- * Gradient Boosted Trees.
- * Support Vector Machines (SVM).
- * Neural Networks.
- * Decision Trees.
- * Naive Bayes.
- * K-Nearest Neighbor.

En nuestro diseño nosotros utilizaremos la metodología de Clasification Learning usando el algoritmo de redes Neuronales que según el estudio de *Palaniappan y Nae-Jen* [14] sería el algoritmo más óptimo para detectar la intención motora cosa que comprobaremos después en el capítulo 5.

- **Aprendizaje No Supervisado:** A diferencia del anterior, los resultados no los conocemos y quedan por definir. El conjunto de datos tiene información sin ninguna relación con los resultados. El principal objetivo es explorar la estructura de la información para extraer información relevante, detectar patrones recurrentes y categorizar. Estos machine learning pueden resolver mediante 2 metodologías:

- * **Clustering:** La idea es poder segmentar el set de datos en diferentes grupos basado en patrones detectados sin necesidad de tener conocimiento previo.
- * **Reducción de Dimensionalidad:** Se utiliza en el caso que queramos reducir las dimensiones de nuestro set de datos. En general, a medida que aumentamos la información del set de datos también consumimos más recursos por lo que puede llegar al caso en el que esto sea imposible de procesar. Una solución es reducir las dimensiones de los datos generando un set más compacto y rápido de procesar, pero sin perder la relevancia de la información.

Dentro de las dos metodologías encontraremos una serie de algoritmos de resolución que serán los siguientes.

- * K-means Clustering.
- * t-SNE.
- * PCA.
- * SVD.

– **Aprendizaje por Refuerzo:** Este tipo de aprendizaje consiste en entrenar al algoritmo para generar un modelo de tal forma que sea capaz de intentar producir como salida el mejor resultado o el más correcto en cierto contexto a partir de un sistema de recompensas.

El modelo se entrena de tal forma que cada acción que realice recibe recompensas o penalizaciones. A partir de prueba y error encuentra el resultado que maximiza las recompensas.

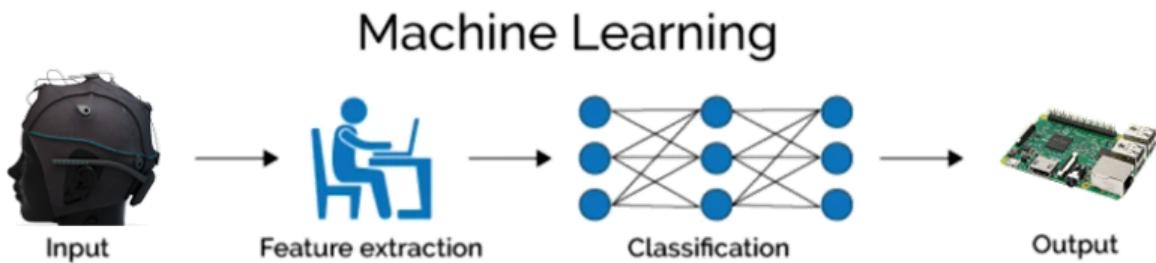


Figura 2.20: Machine Learning.

- **Deep Learning** o aprendizaje profundo es otra disciplina del ámbito de la Inteligencia Artificial posiblemente el futuro de la inteligencia artificial, el deep learning a diferencia del machine learning no se le tienen que pasar las características de las señales para que pueda identificar patrones y crear un algoritmo para diferenciarlos. Es ella misma la que se encarga de extraer las características y identificar dichos patrones.[4]

En la actualidad en el campo de la interpretación de las señales EEG el deep learning no ha conseguido profundizar lo suficiente esto se debe principalmente a que dichas señales EEG varían mucho entre personas o incluso en la misma persona varía dependiendo del momento en el que se tomen dichas muestras, por lo tanto, para tener un método efectivo de deep learning necesitariamos tener un gran nivel de computación y una gran cantidad de muestra. Aun así para una cierto límite de tareas sencillas sí que se podría realizar mediante deep learning.

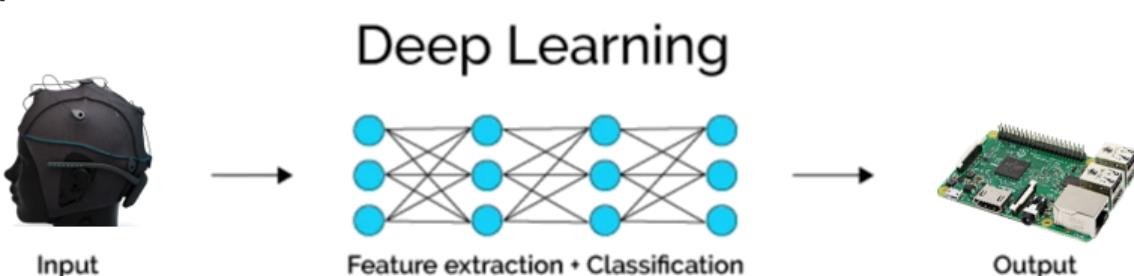


Figura 2.21: Deep Learning.

En nuestro diseño haremos uso de machine learning ya que obtendremos más conocimientos al realizar la etapa de extracción de características.

El machine Learning contara con un flujo de diseño y desarrollo compuesto por una etapa de entrenamiento y después una etapa de predicción explicado en el siguiente diagrama de la figura

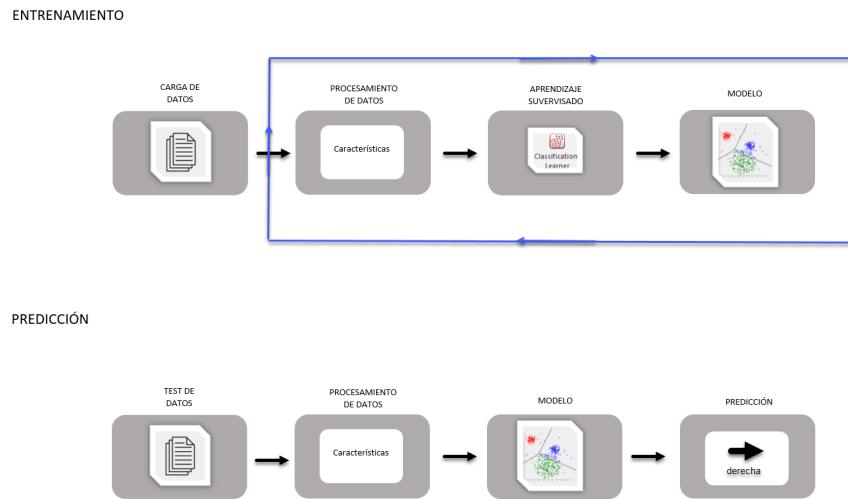


Figura 2.22: Etapas Machine Learning

2.3.4 Bloque de comunicación puerto serie

Este bloque será el encargado de realizar la comunicación con entre nuestra BCI y los dispositivos electrónicos que realizaran las tareas indicadas por el clasificador. Esta comunicación nosotros la realizaremos vía Bluetooth. 2.23

Bluetooth se trata de una especificación para redes inalámbricas de área personal (WPAN) creado por Bluetooth Special Interest Group Inc en los años 90, pero no fue hasta el año 2002 cuando se ratificó como estándar IEEE 802.15.1-2002 en donde se solucionaron una gran cantidad de errores de su versión inicial. Consiste en un estándar inalámbrico que permite la transmisión de datos entre dispositivos a corto alcance, facilitando las comunicaciones entre ambos, eliminando la presencia de cables o conectores y permitiendo una interacción sencilla y rápida entre los dispositivos por medio de un radioenlace en la banda ISM de los 2.4GHz.



Figura 2.23: Bluetooth.

2.3.5 Realimentación

La realimentación facilita información a los sujetos sobre la clasificación que la BCI está realizando en cada momento a partir de las señales electroencefalográficas del individuo. Un correcto aprendizaje de las funciones cerebrales más básicas como hablar o moverse, y de otras más complejas como pueden ser cantar o bailar requiere desde el inicio de su aprendizaje una realimentación a través de los sentidos.

El uso de una BCI no difiere en este sentido de estas habilidades adquiridas desde nuestro nacimiento. El objetivo final de una BCI es sustituir el canal habitual que, mediante el uso de los músculos, comunica el cerebro y el entorno. Por tanto, el éxito de un sistema que utilice una BCI dependerá también de la realimentación y de la adaptación cerebral del usuario al tipo de tarea mental que se requiera en función de dicha realimentación. Como ha sido ampliamente demostrado, el uso de algún tipo de realimentación mejora considerablemente tanto la fase de aprendizaje como la propia utilización de una BCI.

En caso de utilizar una realimentación discreta, ésta puede darse en forma de un número, una letra, un ícono o incluso un sonido. Si se elige una realimentación unidimensional continua, suele realizarse comúnmente mediante la representación de una barra en la pantalla de un ordenador. Para representaciones más complejas que aporten mayor información al usuario, pueden emplearse sistemas continuos de dos dimensiones como juegos del estilo de un punto desplazándose hacia una canasta que puede cambiar de lugar, o sistemas de realidad virtual en tres dimensiones, también de manera más compleja la realimentación también podría ser una silla o coche con tracción motora que se vaya moviendo en función de las tareas que interpreta el BCI.[10]

En el caso de nuestra BCI la realimentación se producirá cuando vemos nuestro coche de tracción moverse o bien si utilizaremos una silla de ruedas motorizada cuando se moviera dicha silla se produciría la realimentación, en caso de utilizar nuestro BCI para el uso de metaversos o juegos de realidad virtual la realimentación se produciría a través de nuestras gafas de realidad virtual cuando se mueva el avatar. La realimentación se considera bastante importante ya que implica un aprendizaje por parte del sujeto que producirá que con la práctica realice dichas tareas mentales con más fluidez y facilidad.

Capítulo 3

Etapas de desarrollo de nuestra BCI

Para que podamos desarrollar una BCI que cumpla con unos estándares de calidad adecuados debemos seguir un flujo de trabajo que hemos dividido en varias etapas.

- **Etapa calibración y entrenamiento:** Como hemos indicado previamente una de las mayores dificultades que tenemos en trabajar con señales EEG es la gran complejidad que tienen dichas señales, además, dichas señales son diferentes en función de la persona y del momento en el que se realice la adquisición de muestras. Para poder desarrollar una BCI eficiente debemos realizar una calibración, la cual se realiza mediante un entrenamiento de los sujetos para poder realizar una toma de datos correctas y crear un clasificador adecuado.

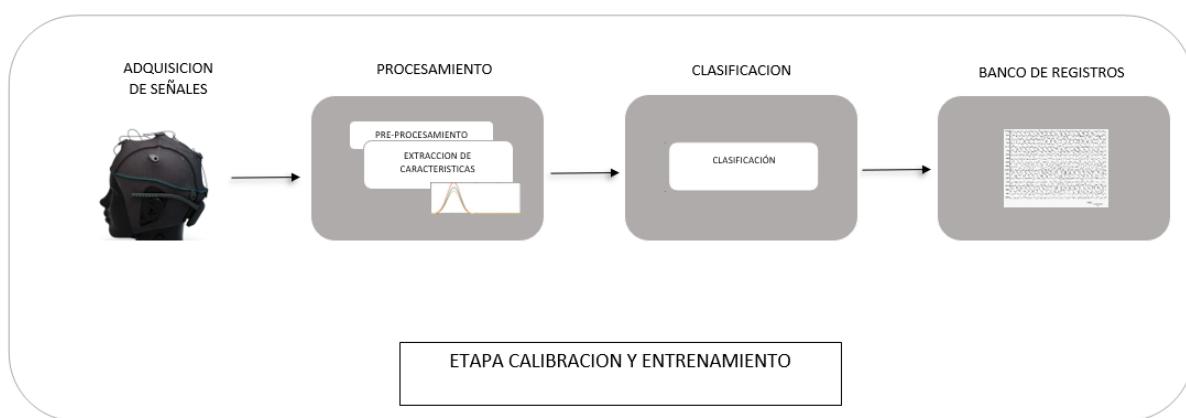


Figura 3.1: Arquitectura Etapa de calibración y entrenamiento.

Como hemos indicado en el capítulo 2 es posible que tengamos una gran cantidad de muestras que no correspondan con las tareas que queremos realizar ya que se necesita un entrenamiento previo y aun así no todas las muestras corresponderán con la tarea. Por dicho motivo es importante cuando el diseño este operativo funcional que el sujeto tenga una realimentación adecuada.

- **Etapa de diseño:** En esta etapa recolectaremos las muestras adquiridas mediante la fase de calibración y entrenamiento, y haremos un filtrado inicial de dichas muestras ya que no todas las muestras obtenidas van a corresponder con la tarea que queremos realizar, este filtrado de muestras es importante ya que si no lo realizamos le estamos indicando a nuestra machine learning que ciertas actividades corresponden con una tarea que no es la suya lo cual haría que nuestro clasificador no sea eficiente.

Por lo tanto como hemos indicado con anterioridad nosotros haremos el filtrado segundo a segundo para obtener un modelo más óptimo y después crearemos el modelo de clasificación del diseño con las muestras ya filtradas y bien clasificadas.

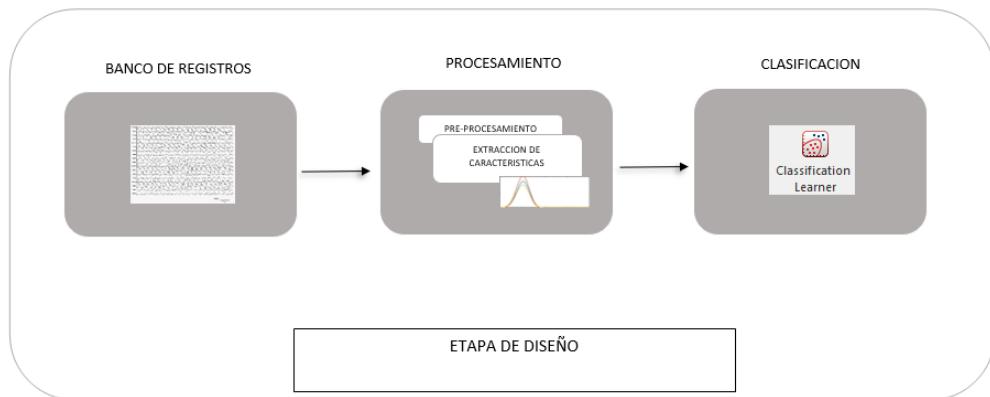


Figura 3.2: Arquitectura Etapa de diseño.

- **Etapa de uso:** En esta etapa se integrara nuestra BCI con cualquier sistema electrónico, que realizará las ordenes trasmítidas por nuestra interfaz y dicho aparato electrónico producirá una retroalimentación al sujeto, esto producirá una etapa de aprendizaje en los sujetos que usen la BCI y que con el uso el sistema sea mas óptimo.

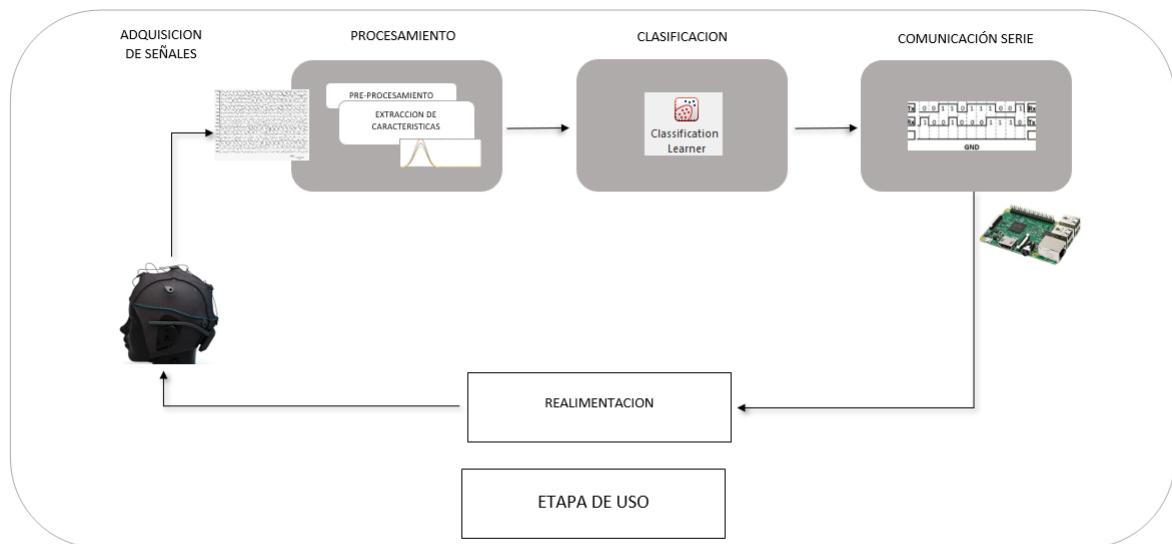


Figura 3.3: Arquitectura Etapa de uso.

Capítulo 4

Herramientas de trabajo para el diseño de nuestra BCI

En este capítulo se va a hacer una descripción breve del Software y Hardware utilizados para el desarrollo de nuestra BCI.

4.1 Hardware

- **EEG Unicorn Hybrid Black:** Es un kit amplificador de señales electro cerebrales. Permite a los desarrolladores, integrar señales del cuerpo humano en sus proyectos, desde la simple visualización de las señales hasta el diseño y control de dispositivos conectados e interacción con instalaciones, juguetes, programas informáticos o aplicaciones, y más. El amplificador Unicorn Hybrid Black adquiere el EEG de ocho electrodos de EEG híbridos Unicorn. El amplificador Unicorn consta de Unicorn Brain Interface Hybrid Black, malla ajustable para los colocación de electrodos, electrodos EEG híbridos Unicorn, cable de carga USB Unicorn y un dongle Bluetooth Unicorn para adquirir datos en una computadora.



Figura 4.1: Piezas Unicorn Hybrid Black.

- **Computadora** Para realizar la interfaz hacemos uso de un portatil HP ENVY 17 Notebook PC con las siguientes especificaciones:
 - Intel(R) Core(TM) i7-4700MQ CPU @ 2.40GHz
 - RAM:16GB
 - Disco Duro :1,5TB
- **Coche a tracción:** Para realizar una posible implementación de nuestra BCI proponemos poder hacer uso del coche diseñado en la práctica de Sistemas Electrónicos Digitales Avanzados.

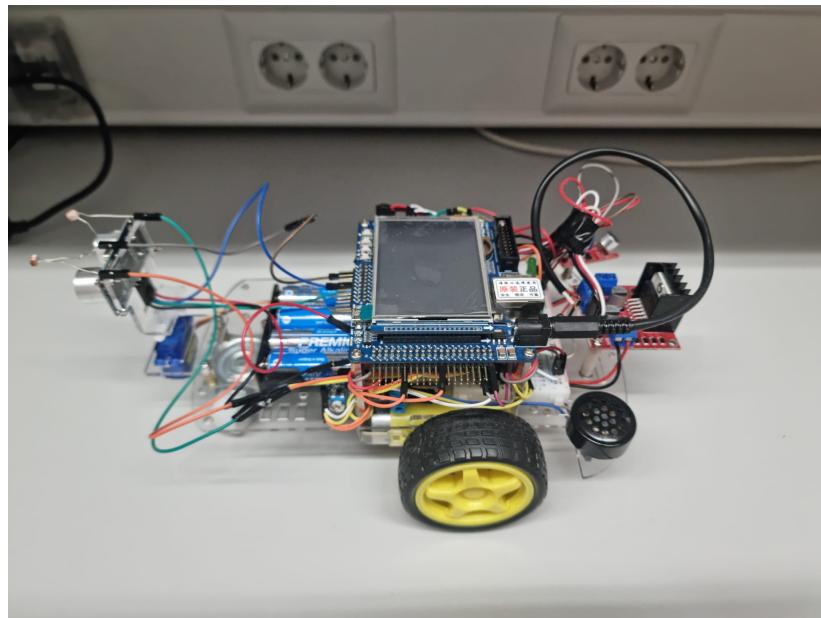


Figura 4.2: Coche a tracción.

4.2 Software

- **Unicorn Suite Hybrid Black:** Es el entorno de software de nuestro amplificador EEG Unicorn Hybrid Black,y el cual consta de aplicaciones independientes y API's para interactuar con Unicorn Brain Interface, adquirir y procesar datos y realizar paradigmas BCI. Nosotros haremos uso de la API Simulink para realizar la etapa de uso y así poder implementar nuestra BCI. También usaremos la API recorder para tomar muestras y poder realizar un estudio de señales EEG.
- **Matlab:** Hacemos uso del entorno Matlab, para analizar, estudiar, procesar y extraer las características de las muestras. También hacemos uso de Matlab para crear el clasificador a través de la API Classification Learning, para crear nuestro modelo de clasificación.[8]
- **Simulink:** Hacemos uso del entorno Simulink, para implementar mediante bloques, las etapas de adquisición, preprocesado, extracción de características, clasificación y comunicación serie.

Capítulo 5

Diseño e implementación BCI

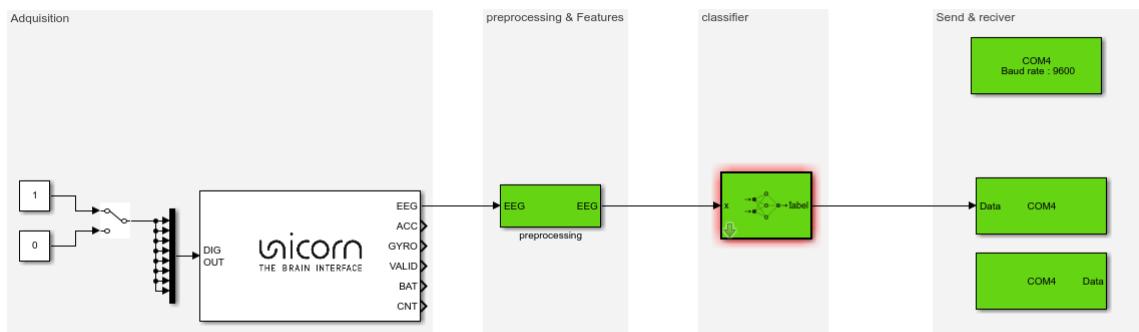


Figura 5.1: Diagrama de bloques BCI.

5.1 Adquisición de datos

En el módulo de adquisición de nuestra BCI nosotrosaremos uso de un amplificador EEG Unicorn Hybrid Black, para el diseño de nuestra BCI solo utilizaremos las muestras adquiridas de los canales 2,3 y 4 a los cuales llamaremos C3,Cz y C4. Estas muestras serán recolectadas en ficheros .mat en la fase de calibración y entrenamiento, dichos ficheros serán procesados en la fase de diseño. La adquisición en la fase de uso será realizada con una frecuencia de 250Hz y será implementada mediante un bloque en Simulink donde solo tendremos que configurar el nº serie de nuestro amplificador EEG tal y como podemos ver en la figura 5.2.

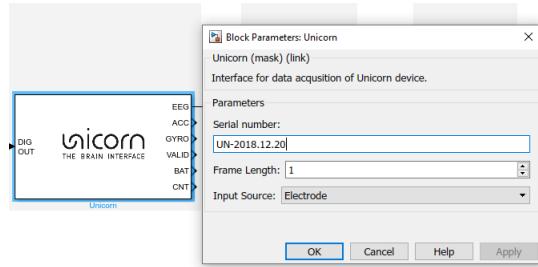


Figura 5.2: Configuración etapa de uso Simulink.

5.2 Preprocesamiento y extracción de características

En este módulo basándonos en los estudios del capítulo 2 hemos decidido hacer un preprocesamiento segundo a segundo, la finalidad de realizarlo así, es la de trasmisir a tiempo real y en todo momento una orden a la BCI, evitar que las muestras se contaminen, con esto queremos decir que si analizamos segundo a segundo siempre detectaremos los eventos sin embargo si analizamos durante toda la franja es posible que no podamos detectar las tareas que se está realizando aunque se produzca dicho evento.

Aunque se produzca dicho entrenamiento es posible que en los eventos de nuestra base de datos no se produzca dicho evento, esto se debe a que los usuarios necesitan un entrenamiento y tener un aprendizaje con realimentación adecuada para que su cerebro se acostumbre a realizar dichas tareas.

Es ciertos que hay estudios que indican que normalmente el evento se suele producir durante los dos segundos después del inicio del intento de la intención motora[10], pero no siempre es así como hemos podido observar en las muestras, por lo que decidimos que la manera más eficiente es analizar segundo a segundo.

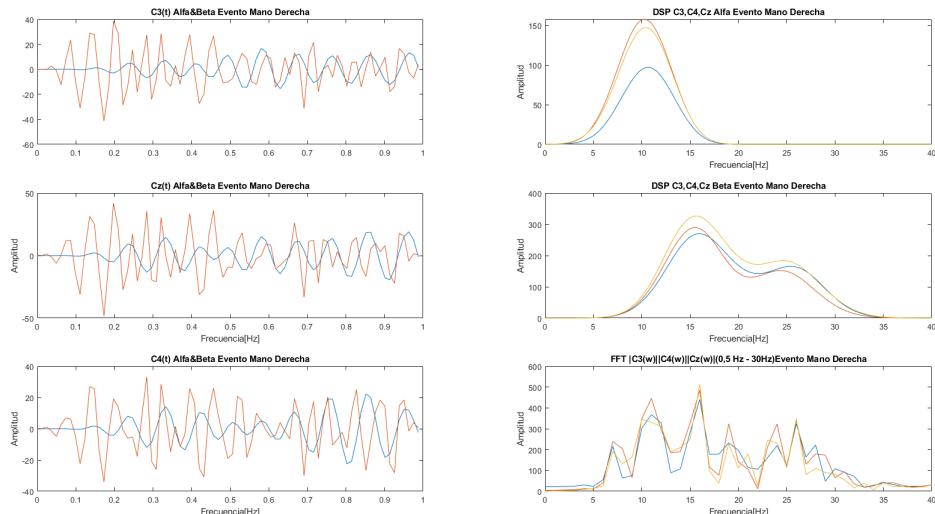


Figura 5.3: Intención motora mano derecha $t=0$.

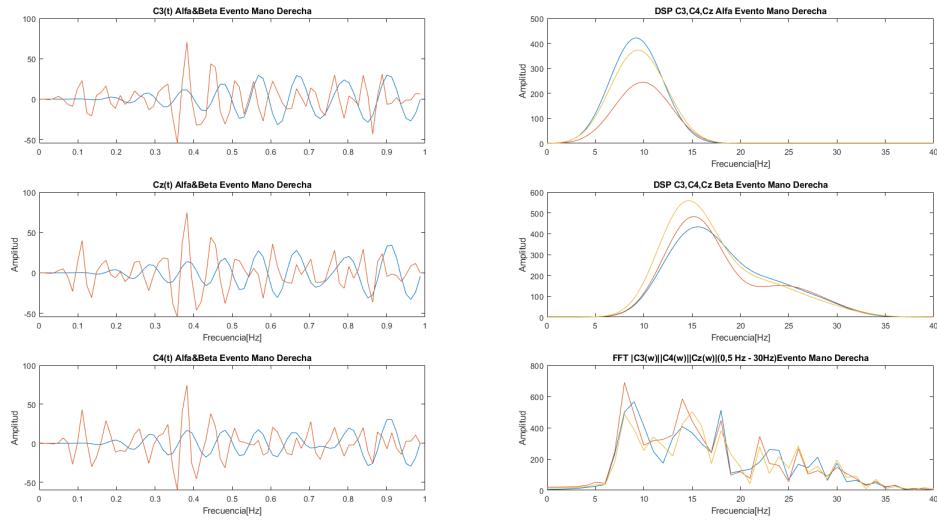


Figura 5.4: Intención motora mano derecha $t=1$.

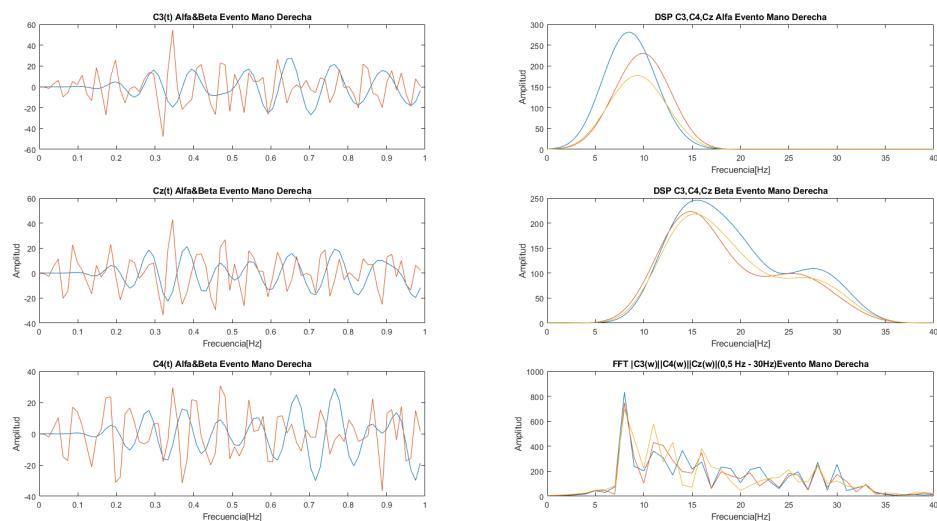


Figura 5.5: Intención motora mano derecha $t=2$.

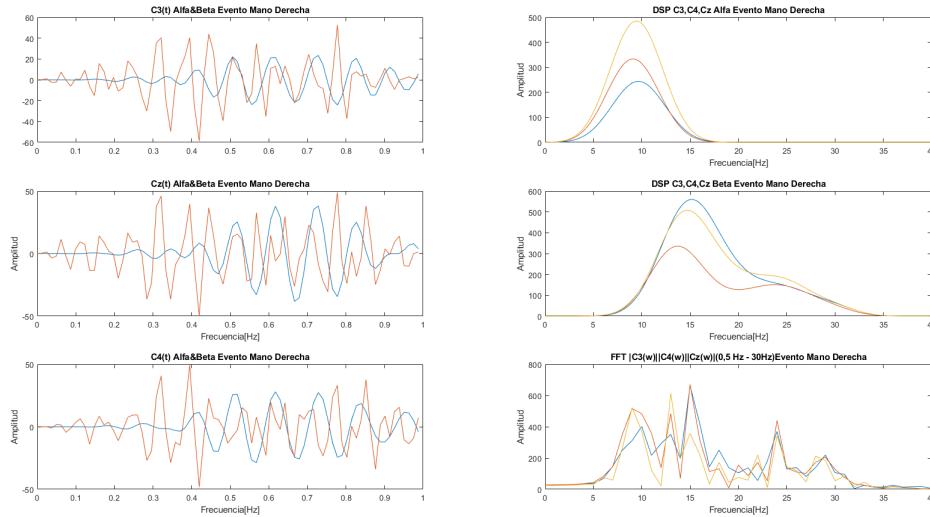


Figura 5.6: Intención motora mano derecha $t=3$.

Como podremos observar la desincronización que produce la intención motora de la mano no se produce hasta el tramo de 3-4 segundos correspondiente con la figura 5.6 desde el inicio de la tarea, tendremos otras muestras en las cuales se producen en otras franjas de tiempo o incluso durante varias franjas o incluso en ninguna franja, lo cual valida que el sistema más óptimo y que producirá mayor precisión, será observar segundo a segundo.

Nuestra etapa de preprocesamiento se basa en el filtrado de la señal en la banda correspondiente a los ritmos mu y beta que son los que queremos observar, después de realizar dicho filtrado procederemos a aplicar la transformada Wavelet para reducir el número de muestras, después realizaremos el cálculo de la densidad espectral mediante el método de Welch's, por último realizaremos la transformada de Fourier.

Después pasaremos a la parte de extracción de características en los ritmos Mu y Beta ,que serán el valor máximo de amplitud de PSD, la media, la varianza, la covarianza, la desviación estándar y los valores máximo y mínimo en el dominio del tiempo y de la frecuencia.

En la fase de diseño de nuestra BCI nosotros hemos decidido usar el banco de datos proporcionado por Physionet [23] ya que tiene un mayor número de muestras y sujetos que nuestros otros bancos de muestras. Dicha base de datos será preprocesada mediante Matlab cuyo código de procesamiento y extracción de características esta adjunto en el anexo.

Esta base de datos sera utilizada para crear y para comprobar el modelo de clasificación.

Por otro lado en la fase de uso nosotros utilizaremos bloques de Simulink para dicha extracción de características tal y como podemos ver en las figuras 5.7, 5.8y 5.9.

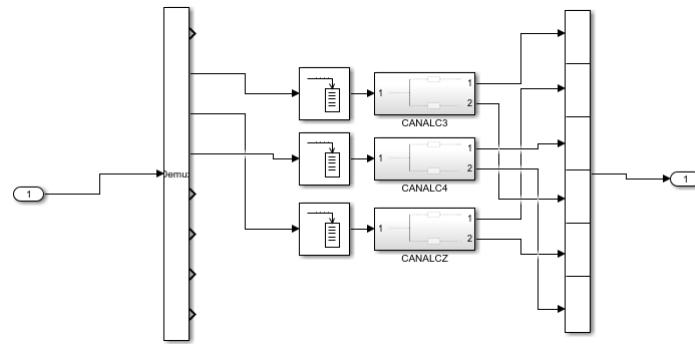


Figura 5.7: Bloque de preprocesado y extracción de características.

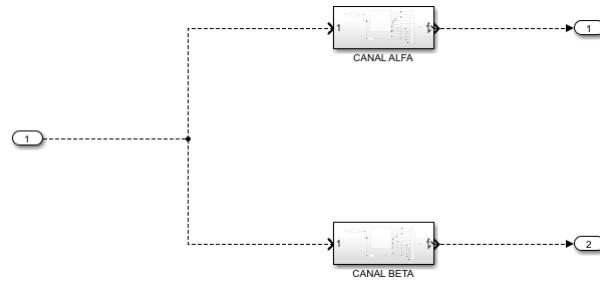


Figura 5.8: SubBloque Canal C3.

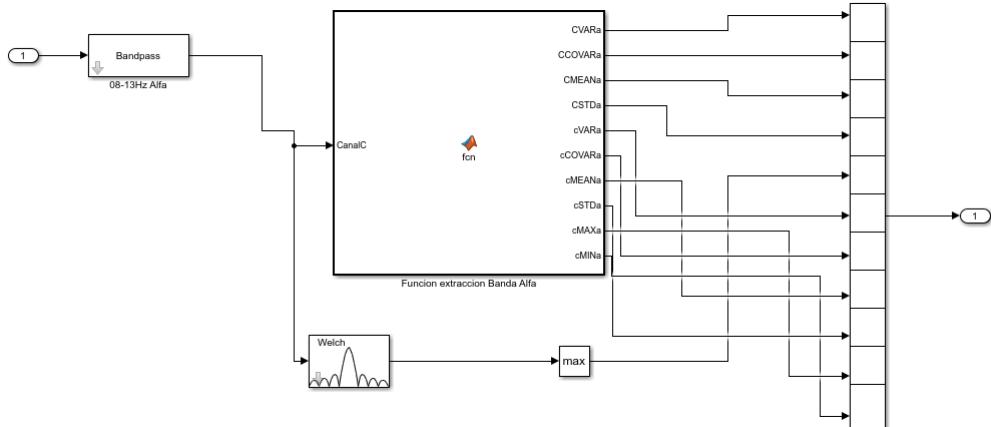


Figura 5.9: SubBloque Banda Alfa.

5.3 Clasificación de señales

Como indicamos previamente en el capítulo 2 nosotros hemos elegido realizar mediante un modelo Classification Learning. Para realizar dicho modelo nosotros hemos utilizado una base de datos suministrada por Physionet [23], hemos decidido hacer uso de esta base de datos ya que cuenta con una gran cantidad de muestras.

La base de datos utilizada es un estudio de 109 sujetos. Cada sujeto realizó 14 carreras experimentales: dos carreras de referencia de un minuto (una con los ojos abiertos, otra con los ojos cerrados) y tres carreras de dos minutos de cada una de las siguientes cuatro tareas:

- Aparece un objetivo en el lado izquierdo o derecho de la pantalla. El sujeto abre y cierra el puño correspondiente hasta que el objetivo desaparece. Entonces el sujeto se relaja.
- Aparece un objetivo en el lado izquierdo o derecho de la pantalla. El sujeto se imagina abriendo y cerrando el puño correspondiente hasta que el objetivo desaparece. Entonces el sujeto se relaja.
- Aparece un objetivo en la parte superior o inferior de la pantalla. El sujeto abre y cierra ambos puños (si el objetivo está arriba) o ambos pies (si el objetivo está abajo) hasta que el objetivo desaparece. Entonces el sujeto se relaja.
- Aparece un objetivo en la parte superior o inferior de la pantalla. El sujeto se imagina abriendo y cerrando ambos puños (si el objetivo está arriba) o ambos pies (si el objetivo está abajo) hasta que el objetivo desaparece. Entonces el sujeto se relaja.

Nuestra base de datos contara con una adquisición de muestras a 160Hz y con 64 sensores de los cuales nosotros como hemos indicado previamente solo haremos uso de los sensores C3,Cz y C4 que serian los sensores que contienen la información de la intención motora. Los muestras contendrán anotaciones que incluye uno de tres códigos (T0, T1 o T2):

- T0 corresponde al descanso
- T1 corresponde al inicio del movimiento (real o imaginario) del puño izquierdo (en las carreras 3, 4, 7, 8, 11 y 12) o ambos manos (en las carreras 5, 6, 9, 10, 13 y 14)
- T2 corresponde al inicio del movimiento (real o imaginario) del puño derecho (en las carreras 3, 4, 7, 8, 11 y 12) o ambos pies (en las carreras 5, 6, 9, 10, 13 y 14)

En nuestra diseño solo haremos uso de los eventos de relajación e intención motora ya que la finalidad es que nuestra BCI pueda ser utilizada por gente con discapacitada o bien gente que quiera realizar alguna acción solo imaginándolo. Las muestras restantes serán descartadas. Como hemos indicado previamente en nuestro estudio iremos procesando segundo a segundo por lo que identificamos realmente cual es el momento o momentos en las anotaciones en los que realmente se produce la intención motora.

Por lo tanto, nuestras muestras después de realizarlas un preprocesamiento y la extracción de sus características serán suministradas al Classification Learning para su aprendizaje y estarán identificadas por las siguientes etiquetas:

- 0: Etiqueta correspondiente con el estado de relajación y con el resto de muestras que no corresponden con la imaginación motora.
- 1: Etiqueta correspondiente con la imaginación motora de la mano derecha.
- 2: Etiqueta correspondiente con la imaginación motora de la mano izquierda.

- 3: Etiqueta correspondiente con la imaginación motora de los pies.

Para la creación del modelo de clasificación hemos usado los datos de 100 sujetos, los datos de los otros 9 sujetos serán utilizados más adelante para ver los resultados.

Debido a que la mayoría de las muestras serán etiquetados con la etiqueta 0 que corresponde con cualquier muestra que no sea la imaginación motora, debemos desechar muestras hasta aproximarnos al número de muestras que tenemos de las otras etiquetas.

Esta eliminación de muestras para crear el modelo de machine learning lo hacemos para evitar un problema de Overfitting. [5]

Como podemos observar en las figuras 5.10, 5.11, 5.12 y como habíamos indicado previamente en el capítulo 2 de los diferentes modelos de clasificación el que tendrá mayor de acierto serán las redes neuronales.

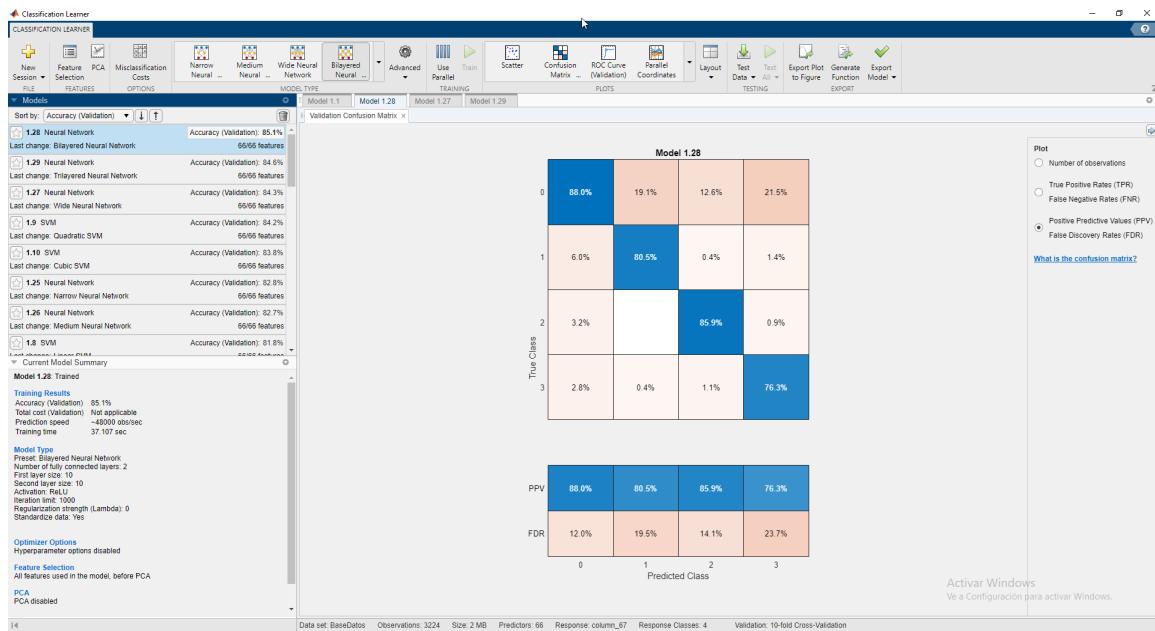


Figura 5.10: Modelo de Clasificación. Gráfica PPV.

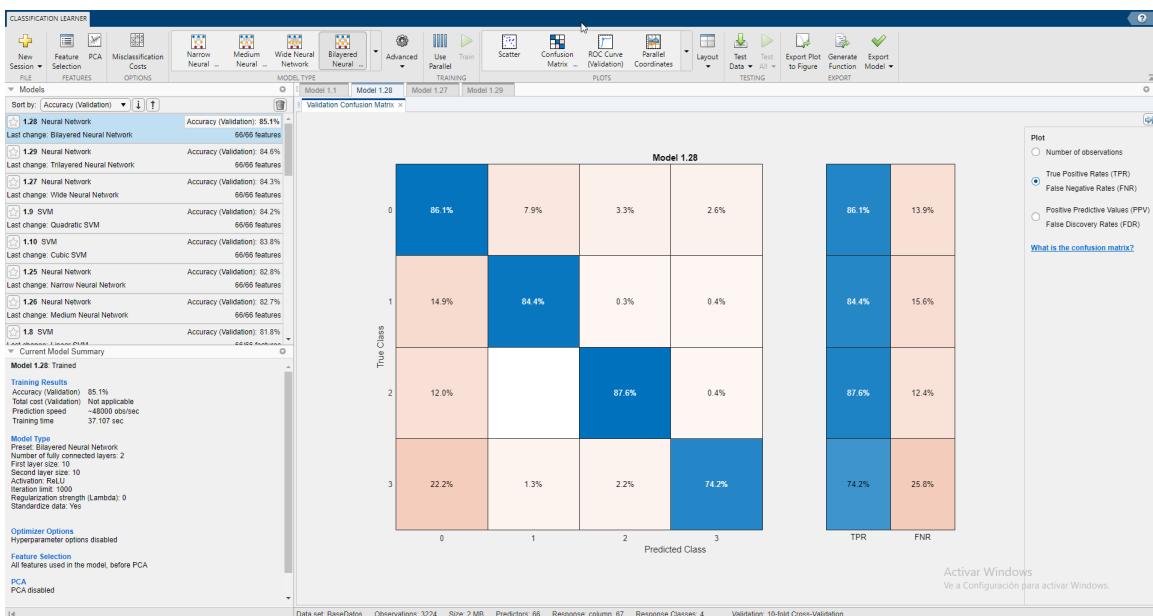


Figura 5.11: Modelo de Clasificación. Gráfica TRP.

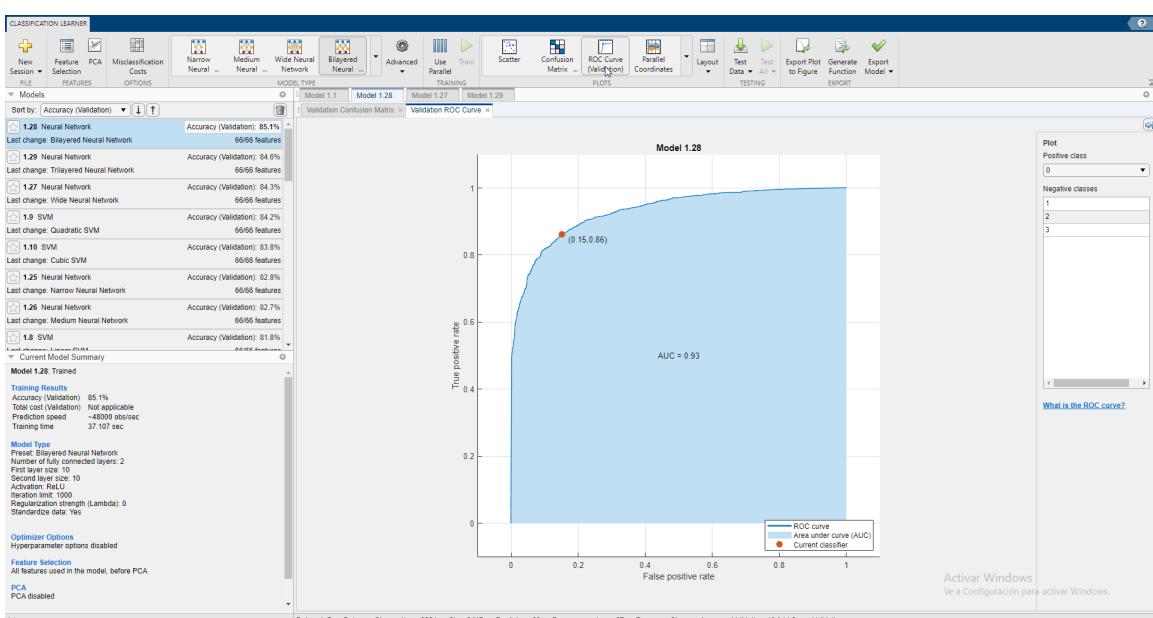


Figura 5.12: Modelos de Clasificación. Gráfica ROC.

5.4 Comunicación serie

Para el diseño del módulo de comunicación serie hacemos uso de Bluetooth y lo realizaremos a través de módulos de Simulink correspondientes. Para la recepción por parte de dispositivos electrónicos vislumbraremos un ejemplo de las órdenes bluetooth utilizado un módulo HC06 integrado en un coche de tracción diseñado en la asignatura de Sistemas Electrónicos Digitales Avanzados, dicha comunicación también se podría realizar con una silla de ruedas motora si contara con un receptor bluetooth para dar asistencia a un discapacitado, o bien se podría utilizar para mover un avatar en un juego de realidad virtual o en el metaverso.

El código utilizado para nuestro ejemplo de modulo bluetooth será una modificación del realizado en el diseño en la práctica de Sistemas Electrónicos digitales y esta adjunto en el anexo.

Como podremos ver indicamos las configuraciones de los bloques comunicación serie en las siguientes figura 5.13.

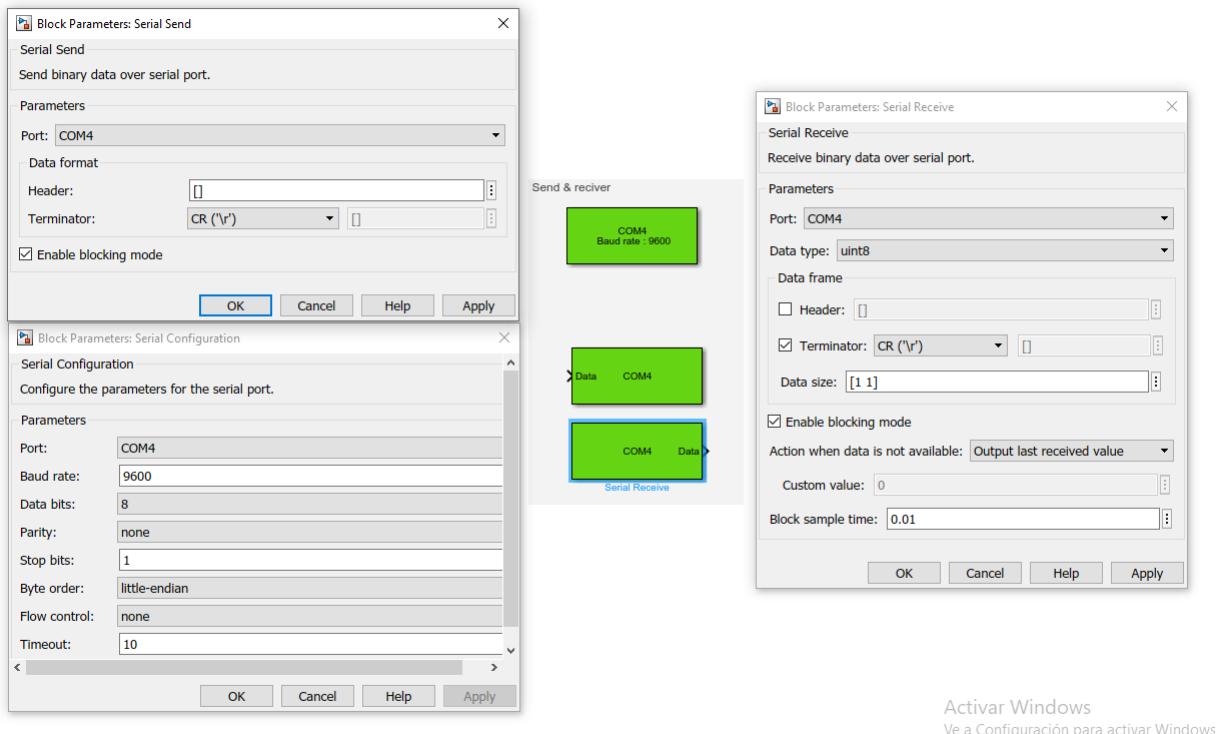


Figura 5.13: Bloque comunicación serie.

5.5 Realimentación

La realimentación en nuestro diseño se producirá cuando un dispositivo conectado mediante bluetooth se mueva hacia adelante o gire hacia a la izquierda o la derecha, o bien si se integra la BCI en un juego de realidad virtual se produciría cuando se mueva el avatar, como mencionamos en el capítulo 2 la realimentación es una parte fundamental para el funcionamiento correcto de la BCI por parte del sujeto, ya que le producirá un aprendizaje con su utilización y un método de uso más óptimo debido al aprendizaje producido por la utilización de la BCI.

Capítulo 6

Resultados

Para comprobar la eficiencia de nuestro diseño, hemos realizado una comprobación del funcionamiento de la etapa de clasificación siendo la etapa crítica de nuestro diseño y comprobaremos que la tasa de acierto corresponde con la misma o es similar a la que nos proporciona el modelo a la hora de crearlo.

Para esta fase de comprobación hemos utilizado los 4 sujetos restantes del estudio de physionet.org obteniendo los resultados adjuntos en la tabla.

Tarea	Nº de realizaciones	Nº acierto	Nº fallos	Porcentaje de acierto
Estado reposo (0)	4.697	2.843	1.854	60,53 %
Mover mano derecha(1)	48	45	3	93,75 %
Mover Mano Izquierda(2)	24	18	6	75,00 %
Mover Pies(3)	16	9	7	56,25 %

Tabla 6.1: Resultados modelo Bilayered Neural Network

Como podemos ver la tasa de acierto el diseño del modulo de clasificacion seria del 71,38 % por lo que podemos visualizar seria inferior a la obtenida en la fase del diseño la cual era del 85,10 %.

Tarea	Nº de realizaciones	Nº acierto	Nº fallos	Porcentaje de acierto)
Estado reposo (0)	4.697	3.116	1.581	66,34 %
Mover mano derecha(1)	48	34	14	70,83 %
Mover Mano Izquierda(2)	24	20	4	83,33 %
Mover Pies(3)	16	11	5	68,75 %

Tabla 6.2: Resultados modelo Wide Neural Network

Como podemos ver la tasa de acierto el diseño del modulo de clasificacion seria del 72,31 % por lo que podemos visualizar seria inferior a la obtenida en la fase del diseño la cual era del 84,30 %.

Tarea	Nº de realizaciones	Nº acierto	Nº fallos)	Porcentaje de acierto
Estado reposo (0)	4.697	3.110	1.587	66,21 %
Mover mano derecha(1)	48	38	10	79,16 %
Mover Mano Izquierda(2)	24	18	6	75,00 %
Mover Pies(3)	16	12	4	75,00 %

Tabla 6.3: Resultados modelo trilayered Neural Network

Como podemos ver la tasa de acierto el diseño del módulo de clasificación seria del 73,84 % por lo que podemos visualizar seria inferior a la obtenida en la fase del diseño la cual era del 84,60 %.

Tarea	Nº de realizaciones	Nº acierto	Nº fallos)	Porcentaje de acierto
Estado reposo (0)	4.697	3.000	1.697	63,87 %
Mover mano derecha(1)	48	33	15	68,75 %
Mover Mano Izquierda(2)	24	19	5	79,16 %
Mover Pies(3)	16	8	8	50,00 %

Tabla 6.4: Resultados modelo svm

Como podemos ver la tasa de acierto el diseño del módulo de clasificación seria del 64,65 % por lo que podemos visualizar seria inferior a la obtenida en la fase del diseño la cual era del 84,20 %.

Como observar ver todos los resultados son inferiores en la etapa de uso, como hemos podido observar tambien el diseño del capítulo 5 de mayor porcentaje de acierto no corresponde con el de la fase de uso. Sin embargo seguimos teniendo una tasa de acierto elevada.

Capítulo 7

Presupuesto

Para la realización del presupuesto del proyecto se han tenido en cuenta varios aspectos. En primer lugar coste de materiales y softwares que componen el trabajo realizado, es decir el amplificador EEG Unicorn Hybrid Black, coche de tracción y también el coste de licencias Matlab y Keil uVision. En segundo lugar, se ha tenido en cuenta la mano de obra es decir el coste por la realización de dicho diseño.

De este modo el presupuesto se muestra a continuación:

Material	Nº unidades	Coste(€)
Kit Unicorn Hybrid Black	1 uds.	13.168,00€
Coche tracción	1 ud.	149,00€
Licencia Matlab	1 ud.	2.350,00€
Licencia Keil uVision	1 ud.	9.910,00 €
Mano de obra(h)	Coste(€/h)	
300h totales	15,38€/h	
Total	30.191,00€	

Tabla 7.1: Presupuesto

Capítulo 8

Conclusiones y trabajos futuros

Con este proyecto hemos pretendido abordar una serie de objetivos que hemos ido cumpliendo.

Gracias a nuestra interfaz podemos controlar un aparato electrónico mediante una interfaz BCI.

Hemos llegado a la conclusión, según hemos visto durante la realización del diseño de la interfaz que la tasa de acierto del sistema va decrementando en función que se incrementa el número de tareas. Inicialmente solo hemos analizado las tareas de imaginación motora solamente llegando a conseguir un 96 % de tasa de acierto, sin embargo, cuando añadimos el cuarto estado de reposo el cual incluye las muestras de reposo como las muestras adquiridas durante la intención motora que no corresponden con dicha tarea la tasa de acierto se bajado considerablemente. Esto se debe esas muestras no tienen ningún patrón que la determine como tenemos con el resto de tareas, ya que pueden ser por estar en reposo o por cualquier otro evento que no corresponda con la imaginación motora.

Por lo tanto, durante el desarrollo del proyecto llegamos a la conclusión que en la extracción de características es la parte más importante para tener un BCI eficiente.

La Realimentación es una parte fundamental en la optimización del uso de nuestra BCI, es importante que el usuario aprenda a utilizar la BCI de manera adecuada y eso lo conseguirá mediante la realimentación

Debido a la gran cantidad de tareas que realiza el cerebro los trabajos futuros en dicho ámbito deberían basarse en el aprendizaje profundo o deep learning, ya que dicha tecnología se encargara también de la extracción de características, por lo que no necesita a ningún humano que supervise e identifique las características fundamentales que tienen dichas tareas cerebrales. Como punto de partida es correcto desarrollar una BCI mediante Machine Learning para entender el funcionamiento de nuestro cerebro y observar los patrones de ciertas tareas sencillas, pero dicho procedimiento de diseño no es óptimo si queremos una BCI más compleja que pueda interpretar muchas tareas.

A día de hoy no se ha conseguido mediante el deep learning llegar a la profundidad suficiente como para descifrar todos los patrones cerebrales en parte por la falta computacional, pero con los ordenadores cuánticos podríamos encontrar un considerable avance en el diseño de BCI mediante deep learning.

Bibliografía

- [1] Interfaces Cerebro- Computadora: experiencias de investigación aplicada a Neuro-rehabilitación. [En línea]. Disponible en: [HTTPS://YOUTU.BE/UL8Q2kR1BLY](https://youtu.be/UL8Q2kR1BLY)
- [2] Electroencefalografía. [En línea]. Disponible en: [HTTPS://ES.WIKIPEDIA.ORG/WIKI/ELECTROENCEFALOGRAFÍA](https://es.wikipedia.org/wiki/ELECTROENCEFALOGRAFÍA)
- [3] ¿Qué es Machine Learning? [En línea]. Disponible en: [HTTPS://WWW.IBM.COM/MX-ES/ANALYTICS/MACHINE-LEARNING](https://www.ibm.com/mx-es/analytics/machine-learning)
- [4] Aprendizaje profundo. [En línea]. Disponible en: [HTTPS://ES.WIKIPEDIA.ORG/WIKI/APRENDIZAJE_PROFUNDO](https://es.wikipedia.org/wiki/APRENDIZAJE_PROFUNDO)
- [5] Qué es overfitting y underfitting y cómo solucionarlo.
[En línea]. Disponible en: [HTTPS://WWW.APRENDEMACHINELEARNING.COM/QUE-ES-OVERFITTING-Y-UNDERFITTING-Y-COMO-SOLUCIONARLO/](https://www.aprendemachinelearning.com/QUE-ES-OVERFITTING-Y-UNDERFITTING-Y-COMO-SOLUCIONARLO/)
- [6] Clasificación de imaginación de movimiento mediante su similitud con el movimiento real en el electroencefalograma. [En línea]. Disponible en: [HTTPS://REPOSITORIO.LASALLE.MX/BITSTREAM/HANDLE/LASALLE/1838/CLASIFICACI%C3%B3N%20DE%20IMAGINACI%C3%B3N%20DE%20MOVIMIENTO%20MEDIANTE%20SU%20SIMILITUD%20CON%20EL%20MOVIMIENTO%20REAL%20EN%20EL%20ELECTROENCEFALOGRAMA.PDF?SEQUENCE=1](https://repositorio.lasalle.mx/bitstream/handle/lasalle/1838/CLASIFICACI%C3%B3N%20DE%20IMAGINACI%C3%B3N%20DE%20MOVIMIENTO%20MEDIANTE%20SU%20SIMILITUD%20CON%20EL%20MOVIMIENTO%20REAL%20EN%20EL%20ELECTROENCEFALOGRAMA.PDF?SEQUENCE=1)
- [7] Ritmo mu. [En línea]. Disponible en: [HTTPS://ES.WIKIPEDIA.ORG/WIKI/RITMO_MU](https://es.wikipedia.org/wiki/RITMO_MU)
- [8] Matlab help.[En línea]. Disponible en: [HTTPS://ES.MATHWORKS.COM/HELP/MATLAB/](https://es.mathworks.com/help/matlab/)
- [9] Unicorn Hybrid Black. [En línea]. Disponible en: [HTTPS://WWW.UNICORN-BI.COM/ES/](https://www.unicorn-bi.com/es/)
- [10] Aportaciones al diseño de interfaces persona-máquina utilizando señales EEG
- [11] Feature Extraction of EEG Signals Using Power Spectral Entropy. [En línea]. Disponible en: [HTTPS://IEEEXPLORE.IEEE.ORG/ABSTRACT/DOCUMENT/4549210](https://ieeexplore.ieee.org/abstract/document/4549210)
- [12] Feature extraction and classification of EEG for imagery movement based on mu/beta rhythms. [En línea]. Disponible en: [HTTPS://IEEEXPLORE.IEEE.ORG/ABSTRACT/DOCUMENT/5639888](https://ieeexplore.ieee.org/abstract/document/5639888)
- [13] Feature extraction and classification of EEG for imaging left-right hands movement. [En línea]. Disponible en: [HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/5234611](https://ieeexplore.ieee.org/document/5234611)

- [14] Classification of mental tasks using fixed and adaptive autoregressive models of EEG signals. [En línea]. Disponible en: [HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/1403205](https://ieeexplore.ieee.org/document/1403205)
- [15] Feature Extraction and Classification of Mental EEG for Motor Imagery. [En línea]. Disponible en: [HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/5362813](https://ieeexplore.ieee.org/document/5362813)
- [16] EEG Feature Extraction and Pattern Classification Based on Motor Imagery in Brain-Computer Interface. [En línea]. Disponible en: [HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/5599682](https://ieeexplore.ieee.org/document/5599682)
- [17] Analysis of EEG records in an epileptic patient using wavelet transform. [En línea]. Disponible en: [HTTPS://WWW.SCIRP.ORG/\(S\(VTJ3FA45QM1EAN45VVFFCZ55\)\)/REFERENCE/REFERENCESPAPERS.ASPX?REFERENCEID=365319](https://www.scirp.org/(S(vtj3fa45qm1ean45vvffc55))/REFERENCE/REFERENCESPAPERS.ASPX?REFERENCEID=365319)
- [18] Compression and Comparison of ECG Signals using DWT and DWPT. [En línea]. Disponible en: [HTTPS://SCIRESQL.S3.US-EAST-2.AMAZONAWS.COM/IJST/ARTICLES/2015/ISSUE-24/ARTICLE76.PDF](https://sciresol.s3.us-east-2.amazonaws.com/IJST/ARTICLES/2015/ISSUE-24/ARTICLE76.PDF)
- [19] Feature extraction and classifier evaluation of EEG for imaginary hand movements. [En línea]. Disponible en: [HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/5582453](https://ieeexplore.ieee.org/document/5582453)
- [20] EEG feature extraction based on wavelet packet decomposition for brain computer interface. [En línea]. Disponible en: [HTTPS://WWW.INFONA.PL/RESOURCE/BWMETA1.ELEMENT.ELSEVIER-930FAAEC-0649-382F-B6CA-EEDDEC854A13](https://www.infona.pl/resource/bwmeta1.element.elsevier-930faaec-0649-382f-b6ca-eeddec854a13)
- [21] Feature extraction of motor imagery EEG signals based on wavelet packet decomposition. [En línea]. Disponible en: [HTTPS://WWW.SEMANTICSCHOLAR.ORG/PAPER/FEATURE-EXTRACTION-OF-MOTOR-IMAGERY-EEG-SIGNALS-ON-HU-LI/5C847F940235E675BB0F63E6500466112DDF882A](https://www.semanticscholar.org/paper/feature-extraction-of-motor-imagery-eeg-signals-on-hu-li/5c847f940235e675bb0f63e6500466112ddf882a)
- [22] Signal processing techniques applied to human sleep EEG signals. [En línea]. Disponible en: [HTTPS://WWW.SCIENCEDIRECT.COM/SCIENCE/ARTICLE/ABS/PII/S174680941300178X](https://www.sciencedirect.com/science/article/abs/pii/S174680941300178X)
- [23] Banco de muestras Physionet. [En línea]. Disponible en: [HTTPS://PHYSIONET.ORG/CONTENT/EEGMMIDB/1.0.0/](https://physionet.org/content/eegmmidb/1.0.0/)

Anexos

A.1 Código Procesamiento de ordenes Bluetooth

```
1 #include <LPC17xx.H>
2 #include "uart.h"
3 #include <stdio.h>
4 #include "string.h"
5 #include "stdbool.h"
6 #include "stdlib.h"
7
8
9
10
11
12
13 char buffer[30]; // Buffer de recepción de 30 caracteres --> Utilizada en la ISR de recepcion de la uart
14 char buffer_tx[30]; // Buffer de recepción de 30 caracteres
15 char *ptr_rx; // puntero de recepción
16 char *ptr; // puntero de recepción
17 char rx_completa; // Flag de recepción de cadena que se activa a "1" al recibir la tecla return CR(ASCII=13)
18 char *ptr_tx; // puntero de transmisión
19 char tx_completa; // Flag de transmisión de cadena que se activa al transmitir el carácter null (fin de cadena)
20 char fine=0;
21 long auxmuestras = 0;
22 int auxmuestras = 0;
23 int estado_uart =1;
24 uint8_t orden_preparada = 0;
25
26 //variables externas
27 extern uint8_t Estado;
28 extern int8_t VelDer;
29 extern int8_t VelIzq;
30 extern int8_t Posserv;
31 extern bool Giro90;
32 extern bool Giromenos90;
33 extern uint32_t SLuz1,SLuz2,SPromedio;
34 extern int distancia;
35 extern int bluetooth;
36 extern int tiempo;
37 int milisegundos= 0;
38 bool modobluetooth= false;
39
40 /*Funcion encargada de ejecutar la maquina de estados para el envio de mensajes*/
41 void MaquinaEstadosBluetooth(){
42     switch(estado_uart){
43         case 1: //envio del mensaje inicial
44             ptr_rx=buffer;
45             tx_cadena_UART("Bienvenido al modo Bluetooth/USB indique su nombre\n\r");
46             estado_uart = 2; //una vez que se ha indicado la trama a transmitir cambiamos de estado para que no la vuelva a repetir
47             break;
48
49         case 2:
50             if(tx_completa == 1) //TRANSMISION DE LAS CADENAS(ESTADO SOLO PARA LA PRINCIPAL)
51                 estado_uart = 3; //en caso de que se haya transmitido toda empieza la recepcion
52
53             break;
54
55         case 3: //RECEPCION DE LAS CADENAS(ESTADO SOLO PARA LA PRINCIPAL)
56             if(rx_completa == 0) //hasta que no se reciba todo nos mantenemos en este estado
57                 estado_uart = 3;
58
59
60         if(rx_completa == 1){ //una vez que se recibe todo procedemos al estado general transmite-envia con ordenes en donde actuamos sobre el HW
61             rx_completa = 0; //borramos flag
62             estado_uart =4; //estado general
63         }
64
65         break;
66
67         case 4: //ELIGE MOVIMIENTO DEL COCHE
68             tx_cadena_UART("Elige modo:\n\r ADELANTE \n\r QUIETO \n\r DERECHA \n\r IZQUIERDA \n\r VOLVER AL MENU PREVIO\n\r ");
69             Posserv=0;
70             estado_uart = 5;
71             break;
72
73         case 5://PROCESAMOS ORDEN DE MOVIMIENTO
74             Posserv=0;
75             if(rx_completa == 0){
76                 estado_uart = 8; //se mantiene hasta que se completa
77             }
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
617
618
619
619
620
621
622
623
624
625
625
626
627
627
628
629
629
630
631
632
633
633
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496

```

```
78 }
79     else if (rx_completa ==1){
80         estado_uart = 7; //volvemos a preparar los mensajes para luego transmitirlos
81         orden_preparada = 0;
82         rx_completa = 0; //borramos flag
83
84         //Segun lo recibido se actua sobre el HW
85         if (strcmp (buffer, "3\r") == 0)
86         {
87             VelDer=75;
88             VelIzq=75;
89             estado_uart=7; //Realizamos ordenes
90         }
91         else if (strcmp (buffer, "0\r") == 0)
92         {
93             VelDer=0;
94             VelIzq=0;
95             estado_uart=7; //Realizamos ordenes
96         }
97         else if (strcmp (buffer, "1\r") == 0){
98             estado_uart = 7;
99             Giromenos90 =true;
100        }
101        else if (strcmp (buffer, "2\r") == 0){
102            estado_uart = 7;
103            Giro90 =true;
104        }
105        else if (strcmp (buffer, "VOLVER AL MENU PREVIO\r") == 0){
106            estado_uart = 4;
107            modobluetooth=false;
108        }
109        else{ //en caso de ser un comando no reconocido
110            tx_cadenaUART0("COMANDO ERRONEO\n\r");
111        }
112    }
113    break;
114 }
115 }
116 }
```

B.1 Código Libreria comunicación serie

```

1 /* uart.c
2  * contiene las funciones:
3
4 1  UART0_IRQHandler(void)
5 2  tx_cadena_UART0(char *ptr)
6 3  uart0_set_baudrate(unsigned int baudrate)
7 4  uart0_init(int baudrate)
8
9 */
10 #include <LPC17xx.h>
11 #include "uart.h"
12
13
14 /*
15  * UART0 interrupt handler
16  */
17 void UART0_IRQHandler(void) {
18
19     switch(LPC_UART0->IIR&0x0E) {
20
21     case 0x04:          /* RBR, Receiver Buffer Ready */
22         *ptr_rx=LPC_UART0->RBR;           /* lee el dato recibido y lo almacena */
23         if (*ptr_rx++ ==13)               // Caracter return --> Cadena completa
24         {
25             *ptr_rx=0;        /* Añadimos el carácter null para tratar los datos recibidos como una cadena*/
26             rx_completa = 1; /* rx completa */
27             ptr_rx=buffer;   /* puntero al inicio del buffer para nueva recepción */
28         }
29     break;
30
31
32     case 0x02:          /* THRE, Transmit Holding Register empty */
33     if (*ptr_tx!=0){
34         LPC_UART0->THR=*ptr_tx++; /* carga un nuevo dato para ser transmitido */
35         tx_completa = 0;
36     else tx_completa=1;
37     break;
38
39     }
40 }
41
42 // Función para enviar una cadena de texto
43 // El argumento de entrada es la dirección de la cadena, o
44 // directamente la cadena de texto entre comillas
45 void tx_cadena_UART0(char *cadena)
46 {
47     ptr_tx=cadena;
48     tx_completa=0;
49     LPC_UART0->THR=*ptr_tx++; // IMPORTANTE: Introducir un carácter al comienzo para iniciar TX o
50 }                         // activar flag interrupción por registro transmisor vacío
51
52
53 static int uart0_set_baudrate(unsigned int baudrate) {
54     int errorStatus = -1; //< Failure
55
56     // UART clock (FCCO / PCLK_UART0)
57     // unsigned int uClk = SystemCoreClock / 4;
58     unsigned int uClk =SystemCoreClock/4;
59     unsigned int calcBaudrate = 0;
60     unsigned int temp = 0;
61
62     unsigned int mulFracDiv, dividerAddFracDiv;
63     unsigned int divider = 0;
64     unsigned int mulFracDivOptimal = 1;
65     unsigned int dividerAddOptimal = 0;
66     unsigned int dividerOptimal = 0;
67
68     unsigned int relativeError = 0;
69     unsigned int relativeOptimalError = 100000;
70
71     uClk = uClk >> 4; /* div by 16 */
72
73     /*
74      * The formula is :
75      * BaudRate= uClk * (mulFracDiv/(mulFracDiv+dividerAddFracDiv) / (16 * DLL)
76      *
77      * The value of mulFracDiv and dividerAddFracDiv should comply to the following expressions:
78      * 0 < mulFracDiv <= 15, 0 <= dividerAddFracDiv <= 15

```

```
79 */
80 for (mulFracDiv = 1; mulFracDiv <= 15; mulFracDiv++) {
81     for (dividerAddFracDiv = 0; dividerAddFracDiv <= 15; dividerAddFracDiv++) {
82         temp = (mulFracDiv * uClk) / (mulFracDiv + dividerAddFracDiv);
83
84         divider = temp / baudrate;
85         if ((temp % baudrate) > (baudrate / 2))
86             divider++;
87
88         if (divider > 2 && divider < 65536) {
89             calcBaudrate = temp / divider;
90
91             if (calcBaudrate <= baudrate) {
92                 relativeError = baudrate - calcBaudrate;
93             } else {
94                 relativeError = calcBaudrate - baudrate;
95             }
96
97             if (relativeError < relativeOptimalError) {
98                 mulFracDivOptimal = mulFracDiv;
99                 dividerAddOptimal = dividerAddFracDiv;
100                dividerOptimal = divider;
101                relativeOptimalError = relativeError;
102                if (relativeError == 0)
103                    break;
104            }
105        }
106    }
107
108    if (relativeError == 0)
109        break;
110}
111
112 if (relativeOptimalError < ((baudrate * UART_ACCEPTED_BAUDRATE_ERROR) / 100)) {
113
114     LPC_UART0->LCR |= DLAB_ENABLE; // importante poner a 1
115     LPC_UART0->DLM = (unsigned char) ((dividerOptimal >> 8) & 0xFF);
116     LPC_UART0->DLL = (unsigned char) dividerOptimal;
117     LPC_UART0->LCR &= ~DLAB_ENABLE; // importante poner a 0
118
119     LPC_UART0->FDR = ((mulFracDivOptimal << 4) & 0xFO) | (dividerAddOptimal & 0x0F);
120
121     errorStatus = 0; //< Success
122 }
123
124 return errorStatus;
125
126
127 void uart0_init(int baudrate) {
128
129     LPC_PINCON->PINSEL0=(1<<4)|(1<<6); // Change P0.2 and P0.3 mode to TXD0 and RXD0
130
131     LPC_UART0->LCR &= ~STOP_1_BIT & ~PARITY_NONE; // Set 8N1 mode (8 bits/dato, sin paridad, y 1 bit de stop)
132     LPC_UART0->LCR |= CHAR_8_BIT;
133
134     uart0_set_baudrate(baudrate); // Set the baud rate
135
136
137     LPC_UART0->IER = THRE_IRQ_ENABLE|RBR_IRQ_ENABLE; // Enable UART TX and RX interrupt (for LPC17xx UART)
138     NVIC_EnableIRQ(UART0_IRQn); // Enable the UART interrupt (for Cortex-CM3 NVIC)
139
140 }
141
142 }
```

C.1 Código preprocesamiento y extraccion características Etapa de Uso

```
1 function [cVARa,CCOVArA, CMEANa, CSTDa, cVARa, cCOVArA, cMEANa, cSTDa, cMAXa, cMINa] = fcn(CanalC)
2 calfat=CanalC;
3
4 %Funcion Wavalet para descomponer la señas y tener reducir el nº de muestras
5 [cc ,lc]= wavedec(calfat,2,'db2');
6 calfat = appcoef(cc,lc,'db2');
7
8 %Calculamos la Densidad espectral de Potencia mediante el metodo Welch
9 Cwelchalfa= pwelch(calfat);
10 PSD=transpose(Cwelchalfa);
11 MAXPSD=max(PSD);
12 %Calculamos la FFT de la señal.
13 Calfat = abs(fft(calfat));
14
15 %%Extraccion de Caracteristicas
16 % Alfa
17 cVARa=var(Calfat);
18 CCOVArA=cov(Calfat);
19 CMEANa=mean(Calfat);
20 CSTDa=std(Calfat);
21 CMAXPSDa=MAXPSD;
22 cVARa=var(calfat);
23 cCOVArA=cov(calfat);
24 cMEANa=mean(calfat);
25 cSTDa=std(calfat);
26 cMAXa=max(calfat);
27 cMINa=min(calfat);
28
```

D.1 Código preprocesamiento y extraccion características Etapa de Diseño

```

1 %%MUESTRAS TOMADAS DE PHYSIONET https://physionet.org/content/eegmmfdb/1.0.0
2 %%Realizamos descomposicion de Wavalet Tree para tomar muestras
3 %%comprendidas entre 0-30 Hz
4 %cargamos ejemplo de toma de datos
5 dir='C:\Users\alvaro\Desktop\UNIVERSIDAD\TFG\MuestrasPhisyonet\files\';
6 c='\' ;
7 d='.edf';
8 load('PassBand.mat');
9 load('Alfa.mat');
10 load('Beta.mat');
11 load('Theta.mat');
12 load('Delta.mat');
13 %Funciones para cargar todos los sujetos y las sesiones en un fichero
14 for sujetos=1:100
15
16 for sesionaux=2:7
17
18 if sujetos<10
19     a='S00';
20 elseif (9<sujetos)&&(sujetos<100)
21     a='S0';
22 else
23     a='S';
24 end
25 sesion= sesionaux*2;
26 if sesion<10
27     b='R0';
28 else
29     b='R';
30 end
31 suj=string(sujetos);
32 ses = string(sesion);
33 filename= strcat(dim,a,suj,c,a,suj,b,ses,d);
34
35 [data, anotaciones]=edfread(filename);
36
37 %%Extraemos los canales que nos interesan par ala imaginacion motora C3,Cz y
38 %%C4
39 CanalCz= data.Cz_{1,1};
40 CanalCz= transpose(CanalCz);
41 CanalC3=data.C3_{1,1};
42 CanalC3=transpose(CanalC3);
43 CanalC4=data.C4_{1,1};
44 CanalC4=transpose(CanalC4);
45 long= length(data.Cz__);
46 for x=2:long
47 auxC3= data.C3_{x,1};
48 auxCz= data.Cz_{x,1};
49 auxC4= data.C4_{x,1};
50 auxC3= transpose(auxC3);
51 auxCz= transpose(auxCz);
52 auxC4= transpose(auxC4);
53 CanalC4= [CanalC4 auxC4];
54 CanalC3= [CanalC3 auxC3];
55 CanalCz= [CanalCz auxCz];
56 eventos;time= anotaciones.Onset;

```

```

57 eventoslabel= anotaciones.Annotations;
58 end
59
60 %Preprocesamos los 30 eventos por sesion segundo a segundo para extraer sus
61 %caracteristicas y clasificarlos
62
63 for x=1:30
64 event=seconds(eventostime(x));
65 tipoevento=eventoslabel(x);
66 for e=0:3
67 c3aux= CanalC3(1,160*event+1+(e*160):160*(event+1)+(e*160));
68 c4aux= CanalC4(1,160*event+1+(e*160):160*(event+1)+(e*160));
69 czaux= CanalCz(1,160*event+1+(e*160):160*(event+1)+(e*160));
70
71 %%Filtramos la señas para observar la componente Alfa comprendida entre
72 %8-13Hz
73 c3alfat=filter(Alfa,c3aux);
74 czalfat=filter(Alfa,czaux);
75 c4alfat=filter(Alfa,c4aux);
76 %Funcion Wavalet para descomponer la señas y tener reducir el nº de muestras
77 [cc3 ,lc3]= wavedec(c3alfat,1,'db2');
78 [ccz ,lcz]= wavedec(czalfat,1,'db2');
79 [cc4 ,lc4]= wavedec(c4alfat,1,'db2');
80 c3alfat = appcoef(cc3,lc3,'db2');
81 czalfat = appcoef(ccz,lcz,'db2');
82 c4alfat = appcoef(cc4,lc4,'db2');
83 ft=((0:length(c4alfat)-1)/length(c4alfat));
84
85 %Calculamos la Densidad espectral de Potencia mediante el metodo Welch
86 [C3welchalfa,f] = pwelch(c3alfat);
87 [Czwelchalfa,f] = pwelch(czalfat);
88 [C4welchalfa,f] = pwelch(c4alfat);
89 %Calculamos la FFT de la señal.
90 C3alfat = abs(fft(c3alfat));
91 Czalfat = abs(fft(czalfat));
92 C4alfat = abs(fft(c4alfat));
93
94 %%Filtramos la señas para observar la componente Beta comprendida entre
95 %14-30Hz
96 c3betat=filter(Beta,c3aux);
97 czbetat=filter(Beta,czaux);
98 c4betat=filter(Beta,c4aux);
99 %Funcion Wavalet para descomponer la señas y tener reducir el nº de muestras
100 [cc3 ,lc3]= wavedec(c3betat,1,'db2');
101 [ccz ,lcz]= wavedec(czbetat,1,'db2');
102 [cc4 ,lc4]= wavedec(c4betat,1,'db2');
103 c3betat = appcoef(cc3,lc3,'db2');
104 czbetat = appcoef(ccz,lcz,'db2');
105 c4betat = appcoef(cc4,lc4,'db2');
106 %Calculamos la Densidad espectral de Potencia mediante el metodo Welch
107 [C3welchbeta,f] = pwelch(c3betat);
108 [Czwelchbeta,f] = pwelch(czbetat);
109 [C4welchbeta,f] = pwelch(c4betat);
110 %Calculamos la FFT de la señal.
111 C3betat = abs(fft(c3betat));
112 Czbetat = abs(fft(czbetat));
113 C4betat = abs(fft(c4betat));
114
115 %%Filtramos la señas para observar todo el espectro entre
116 %0.5-30Hz
117 c3band=filter(PassBand,c3aux);
118 czband=filter(PassBand,czaux);
119 c4band=filter(PassBand,c4aux);
120 %Funcion Wavalet para descomponer la señas y tener reducir el nº de muestras
121 [cc3 ,lc3]= wavedec(c3band,1,'db2');
122 [ccz ,lcz]= wavedec(czband,1,'db2');
123 [cc4 ,lc4]= wavedec(c4band,1,'db2');
124 C3band = appcoef(cc3,lc3,'db2');
125 Czband = appcoef(ccz,lcz,'db2');
126 C4band = appcoef(cc4,lc4,'db2');
127 %Calculamos la FFT de la señal.
128 C3band = abs(fft(C3band));
129 Czband = abs(fft(Czband));
130 C4band = abs(fft(C4band));
131 %Cogemos la mitad del espectro
132 C3band=C3band(1:41);
133 Czband=Czband(1:41);
134 C4band=C4band(1:41);
135 fb=((0:40)/40)*40;
136 %Longitud temporal
137 ft=((0:length(c4alfat)-1)/length(c4alfat));
138 %Longitud densidad espectral
139 f=(f/(2*pi))*80;
140 % %%Representacion Grafica

```

```

141 figure(e+1);
142 subplot(3,2,1);
143 plot(ft,c3alfat,ft,c3betat);
144 title('C3(t) Alfa&Beta Evento',tipoevento);
145
146 subplot(3,2,3);
147 plot(ft,czalfat,ft,czbetat);
148 title('Cz(t) Alfa&Beta Evento',tipoevento);
149 xlabel('Frecuencia[Hz]');
150 ylabel('Amplitud');
151
152 subplot(3,2,5);
153 plot(ft,c4alfat,ft,c4betat)
154 title('C4(t) Alfa&Beta Evento',tipoevento);
155 xlabel('Frecuencia[Hz]');
156 ylabel('Amplitud');
157
158 subplot(3,2,2);
159 plot(f,C3welchalpha,f,C4welchalpha,f,Czwelchalpha)
160 title('DSP C3,C4,Cz Alfa Evento',tipoevento);
161 xlabel('Frecuencia[Hz]');
162 ylabel('Amplitud');
163
164 subplot(3,2,4);
165 plot(f,C3welchbeta,f,C4welchbeta,f,Czwelchbeta)
166 title('DSP C3,C4,Cz Beta Evento',tipoevento);
167 xlabel('Frecuencia[Hz]');
168 ylabel('Amplitud');
169
170 subplot(3,2,6);
171 plot(fb,C3band,fb,Czband,fb,C4band)
172 title('FFT |C3(w)||C4(w)||Cz(w)|(0,5 Hz ~ 30Hz)Evento',tipoevento);
173 xlabel('Frecuencia[Hz]');
174 ylabel('Amplitud');
175 %miramos las frecuencias de 14 a 18Hz
176 C3welchbeta = C3welchbeta(1:58);
177 Czwelchbeta = Czwelchbeta(1:58);
178 C4welchbeta = C4welchbeta(1:58);
179 %%Extraccion de Caracteristicas
180 % Alfa
181 C3VARa=var(C3alfat);
182 CzVARa=var(Czalfat);
183 C4VARa=var(C4alfat);
184 C3COVARa=cov(C3alfat);
185 CzCOVARa=cov(Czalfat);
186 C4COVARa=cov(C4alfat);
187 C3MEANa=mean(C3alfat);
188 CzMEANa=mean(Czalfat);
189 C4MEANa=mean(C4alfat);
190 C3STDa=std(C3alfat);
191 CzSTDa=std(Czalfat);
192 C4STDa=std(C4alfat);
193 C3MAXPSDa=max(C3welchalpha);
194 CzMAXPSDa=max(Czwelchalpha);
195 C4MAXPSDa=max(C4welchalpha);
196 c3VARa=var(c3alfat);
197 czVARa=var(czalfat);
198 c4VARa=var(c4alfat);
199 c3COVARa=cov(c3alfat);
200 czCOVARa=cov(czalfat);
201 c4COVARa=cov(c4alfat);
202 c3MEANa=cov(c3alfat);
203 czMEANa=cov(czalfat);
204 c4MEANa=cov(c4alfat);
205 c3STDa=std(c3alfat);
206 czSTDa=std(czalfat);
207 c4STDa=std(c4alfat);
208 c3MAXa=max(c3alfat);
209 czMAXa=max(czalfat);
210 c4MAXa=max(c4alfat);
211 c3MINa=min(c3alfat);
212 czMINa=min(czalfat);
213 c4MINa=min(c4alfat);
214
215 %Beta
216 C3VARb=var(C3betat);
217 CzVARb=var(Czbetat);
218 C4VARb=var(C4betat);
219 C3COVARb=cov(C3betat);
220 CzCOVARb=cov(Czbetat);
221 C4COVARb=cov(C4betat);
222 C3MEANb=mean(C3betat);
223 CzMEANb=mean(Czbetat);
224 C4MEANb=mean(C4betat);

```

```

225     C3STDb=std(C3betat);
226     CzSTDb=std(Czbetat);
227     C4STDb=std(C4betat);
228     C3MAXPSDb=max(C3welchbeta);
229     CzMAXPSDb=max(Czwelchbeta);
230     C4MAXPSDb=max(C4welchbeta);
231     c3VARb=var(c3betat);
232     czVARb=var(czbetat);
233     c4VARb=var(c4betat);
234     c3COVARb=cov(c3betat);
235     czCOVARb=cov(czbetat);
236     c4COVARb=cov(c4betat);
237     c3MEANb=cov(c3betat);
238     czMEANb=cov(czbetat);
239     c4MEANb=cov(c4betat);
240     c3STDb=std(c3betat);
241     czSTDb=std(czbetat);
242     c4STDb=std(c4betat);
243     c3MAXb=max(c3betat);
244     czMAXb=max(czbetat);
245     c4MAXb=max(c4betat);
246     c3MINb=min(c3betat);
247     czMINb=min(czbetat);
248     c4MINb=min(c4betat);
249
250 %ETIQUETAMOS EL EVENTO
251
252 % Cambiamos la etiqueta a los segundo que no corresponden con el evento
253 % El evento lo identificamos por que se produce una desincronizacion en la zona de actividad produciendo una baja respuesta en alfa y en beta
254 if tipoevento == "T2" %Imaginacion motora mano derecha
255     if ((sesion==4)|| (sesion==8)|| (sesion==12))
256         if((C3MAXPSDa< C4MAXPSDa)&&(C3MAXPSDa< CzMAXPSDa)&&(C3MAXPSDb> C4MAXPSDb)&&(C3MAXPSDb> CzMAXPSDb))
257             ETIQUETA=1;%Evento imaginacion mano derecha
258         else
259             ETIQUETA=0;%No corresponde con imaginacion mano derecha
260         end
261     else
262         ETIQUETA=4;%Muestras Descartadas
263     end
264 elseif tipoevento == "T0"
265     ETIQUETA =0;%relajacion y otras muestras que no corresponden con el movimiento de manos y pies
266 else
267     if (sesion==4)|| (sesion==8)|| (sesion==12)
268         if((C4MAXPSDa< C3MAXPSDa)&&(C4MAXPSDa< CzMAXPSDa)&&(C4MAXPSDb> C3MAXPSDb)&&(C4MAXPSDb> CzMAXPSDb))
269             ETIQUETA=2;%Evento imaginacion mano izquierda
270         else
271             ETIQUETA=0;%No corresponde con imaginacion mano izquierda
272         end
273     else
274         if((CzMAXPSDa< C3MAXPSDa)&&(CzMAXPSDa< C4MAXPSDa)&&(CzMAXPSDb> C3MAXPSDb)&&(CzMAXPSDb> C4MAXPSDb))
275             ETIQUETA=3;%Evento imaginacion pies
276         else
277             ETIQUETA=0;%No corresponde con imaginacion pies izquierda
278         end
279     end
280 end
281 %Vector de caracteristicas para el MACHINE LEARNING
282 DAT = [ C3VARa,C3COVARa,C3MEANa,C3STDa,C3MAXPSDa,c3VARa,c3COVARa,c3MEANa,c3STDa,c3MAXa,c3MINa,CzVARa,CzCOVARa,CzMEANa,CzMAXPSDa,CzSTDa,
283     czVARa,czCOVARa,czMEANa,czSTDa,czMAXa,czMINa,C4VARa,C4COVARa,C4MEANa,C4MAXPSDa,C4STDa,c4VARa,c4COVARa,c4MEANa,c4STDa,c4MAXa,c4MINa,
284     C3VARb,C3COVARb,C3MEANb,C3MAXPSDb,C3STDb,c3VARb,c3COVARb,c3MEANb,c3STDb,c3MAXb,c3MINb,CzVARb,CzCOVARb,CzMEANb,
285     CzMAXPSDb,CzSTDb,czVARb,czCOVARb,czMEANb,czSTDb,czMAXb,czMINb,C4VARb,C4COVARb,C4MEANb,C4MAXPSDb,C4STDb,c4VARb,
286     c4COVARb,c4MEANb,c4STDb,c4MAXb,c4MINb,ETIQUETA];
287 if >0
288     DATO =[DATO;DAT];
289     else
290         DATO= DAT;
291     end
292 end
293 if x>1
294     DATOS =[DATOS;DATO];
295     else
296         DATOS= DATO;
297     end
298 end
299 if sesion>4
300     SUJETO =[SUJETO;DATOS];
301     else
302         SUJETO= DATOS;
303     end
304 end
305 if sujetos>1
306     BaseDatos =[BaseDatos;SUJETO];
307     else
308         BaseDatos= SUJETO;
309     end
310 end

```


Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR

