

UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



**Grado en Ingeniería de Tecnologías y Servicios de
Telecomunicación**

TRABAJO FIN DE GRADO

**DISEÑO DE INTERFACES CEREBRO-MÁQUINA Y
HOMBRE-MÁQUINA CONTROLADOS POR
SEÑALIZACIÓN BIOLÓGICA.**

**Ana Sotomayor Romillo
Tutor: Pablo Varona Martínez**

JULIO 2014

**DISEÑO DE INTERFACES CEREBRO-MÁQUINA Y
HOMBRE-MÁQUINA CONTROLADOS POR
SEÑALIZACIÓN BIOLÓGICA.**

**AUTOR: Ana Sotomayor Romillo
TUTOR: Pablo Varona Martínez**

**Grupo de Neurocomputación Biológica
Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Julio de 2014**

Agradecimientos

Lo primero, quiero darte las gracias a ti que estás leyendo esto, porque de una forma u otra has contribuido a que hoy este trabajo esté acabado. Gracias.

Gracias a Pablo, mi tutor, por haberme dado la oportunidad de realizar este proyecto, por su tiempo, confianza y libertad a la hora de trabajar. Gracias también al equipo del GNB, por hacer del laboratorio un sitio en el que da gusto estar. Gracias a Víctor por su música, sus peces y su ayuda en todo momento, y gracias a Carlos por su impresora 3D y sus consejos. Gracias también a Paco, Álvaro, Manuel, Alejandro, Ángel y a Irene y sus cangrejos. Seguimos bailando.

Gracias al "Cottolengo" por su formidable labor día a día, por permitirme ir y enseñarme tanto, en especial a D. Javier, Sonia y a la Madre Claudi. Gracias a tres personas muy bonitas que desde el primer momento me ofrecieron su paciencia, su sonrisa y su "Sí".

Quiero agradecer también a los 23 voluntarios que han dedicado su tiempo a dejar que les "midiera las cabezas", por haber aguantado las pruebas y ayudarme a realizar este trabajo.

Gracias a todos mis compañeros de clase, que han hecho de esta carrera un camino que merece la pena recorrer. Gracias por cada comida, clase, tarde de laboratorio y fin de semana de biblioteca. Por cada "ir a tomar algo" al final del día, cada partido y cada (aunque escasa) fiesta. Gracias a Sara, Patri y Marta, por ser mi equipo de remo estos 4 años. A Paula, por haber sido un apoyo fundamental durante este curso, y a Nacho y Pekas porque no es posible estar con ellos y no sonreír.

Gracias a mis amigas, las de siempre, por darle significado a la palabra "incondicional". Por demostrar a lo largo de los años que en lo bueno son geniales, y en lo malo, las mejores.

Nunca podré agradecer lo suficiente a la familia Val Cano por abrirme su casa para estudiar, comer, charlar o lo que haya hecho falta. Gracias a Jota, por haber estado y por seguir estando.

Por supuesto, gracias a mis padres, a Jaime y a Suri por ser los que han sufrido mi peor parte de estos cuatro años y aun así no han dejado ni un momento de darme su apoyo. Gracias por darme una educación, unos valores y una referencia que seguir en la vida. Gracias a los Romillo, a mis 15 primos-hermanos por ser más hermanos que primos y a mis tíos por sus fiestas, consejos, palabras de ánimo y sus "¿Te pregunto la lección?". Gracias a mi abuela por su ejemplo, su alegría, sus canciones y su fuerza inagotable.

Pero sobre todo y en especial, quiero darle las gracias a mi abuelo. Por ser el que nunca dudó de mí. Por estar ahí, desde sacándole punta a mis lápices de colores hasta para explicarme las sucesiones geométricas. Gracias por la cara con la que me mirabas cuando decías a tus amigos que estudiaba Teleco, y por tantas tardes de hospital estudiando juntos Redes II. Gracias por haber sido el ingeniero que todos deberíamos ser. El abuelo, padre, esposo, amigo y persona que todos deberíamos ser.

*Ana Sotomayor Romillo
Julio, 2014.*

RESUMEN

En este trabajo se aborda el diseño y construcción de un dispositivo portátil e inalámbrico de estimulación visual que funcione en una interfaz cerebro-máquina, con el objetivo de ofrecer un medio de comunicación a personas con discapacidades físicas y motoras que les impidan el movimiento o incluso el habla. Este dispositivo deberá ser capaz de adaptarse a cada usuario y modificar los valores de sus parámetros para conseguir los mejores resultados.

Para ello, se comenzará realizando un estudio sobre la situación actual de las interfaces cerebro-máquina explicando qué son, cuál es su utilidad, su funcionamiento y los distintos sistemas de adquisición y de estimulación que pueden ser utilizados.

Una vez entendida la necesidad de este dispositivo, se contemplarán las distintas alternativas a la hora de implementarlo, explicando cada opción, sus ventajas y sus inconvenientes. Se optará finalmente por un diseño híbrido hardware-software basado en la plataforma Arduino-Yún, y se explicarán sus características, desarrollando cada una de las partes y explicando los procedimientos que se han seguido para llegar al diseño final. El resultado será un dispositivo de estimulación visual de bajo coste, capaz no solamente de encender cada LED a una frecuencia distinta, sino de comunicarse con la interfaz para ir cambiando parámetros de forma que se ajusten a cada sujeto, además de permitir almacenar los datos de frecuencias de cada usuario y minimizar las molestias visuales a la hora de ser utilizado.

A lo largo del trabajo se valida la funcionalidad de este dispositivo, verificando la precisión temporal del estímulo, su capacidad de comunicación inalámbrica, la eficacia de la configuración y el consumo reducido. Con el dispositivo desarrollado, se analizarán los resultados obtenidos en un total de 23 sujetos voluntarios, primero con el objetivo de comprobar si su funcionamiento es correcto y después para fijar los parámetros de diseño que maximizan dichos resultados, como pueden ser las frecuencias escogidas, la distancia entre focos de luz, el brillo, color, etc.

Se ha querido evaluar el funcionamiento de este dispositivo más allá de las pruebas con sujetos sanos, para ver si verdaderamente es posible utilizarlo como medio de comunicación en personas discapacitadas. Por esta razón, se incluye un análisis de las pruebas que se han realizado en el centro "Cottolengo", una residencia de mujeres discapacitadas situado en Algete.

Por último, se evaluarán los resultados obtenidos, comentando las conclusiones a las que se han llegado a partir de ellos, y se sugerirán unas líneas de estudio futuro para seguir avanzando en el diseño de este tipo de sistemas basados en la adaptación a cada sujeto.

ABSTRACT

The aim of this project is to develop a portable, wireless visual stimulating device that works under a brain-computer interface, with the goal of providing a way of communication to people with physical or motor disabilities that prevent them from moving or even talking. This device must be able to adapt itself to every user, and modify its parameters to ensure the best results.

In order to achieve that, the first step is to carry out an investigation about the current situation of brain-computer interfaces (BCI), explaining what they are, their serviceability and the different acquisition and stimulation systems that can be used along with the BCI.

Once the need for this device has been justified, different ways of implementation will be considered, explaining every option with its advantages and drawbacks. The Arduino-based software-hardware hybrid design will be the final choice, and its hardware and software features will be explained, developing each part and explaining the procedures that lead to the final design. The final result will be a low cost visual stimulating device able not only of configuring each LED to a different frequency, but to communicate with the interface in order to switch parameters to offer each subject his best frequencies, moreover the device is able to load and save data to a card, and it is designed to minimize the annoyance in the users.

During this work the functionality of this device will be validated, checking the temporal resolution of the stimuli, its wireless features, the effectiveness of the configuration and its reduced consumption. With the final device developed, the results will be analyzed in a total of 23 volunteers, to test if it works as expected and also to fix the design parameters that optimize the results, for example the frequencies, distance between LEDs, brightness, color, etc.

In addition to this evaluation on healthy subjects, there is a chapter in this project in which the device is tested on handicapped people, to check if the goal of this work is achieved. Those tests were done in "Cottolengo", a center for disabled women in the town of Algete.

Finally, the results obtained will be used to draw the conclusions about the performance of the device; moreover some guidelines about further work on these interfaces will be included.

LISTA DE PALABRAS CLAVE

Interfaz cerebro-máquina (BCI), electroencefalograma (EEG), Steady-State Visual Evoked Potentials (SSVEP), Arduino, LED, estimulación específica del sujeto.

KEY WORDS

Brain Computer Interface (BCI), electroencephalogram (EEG), Steady-State Visual Evoked Potentials (SSVEP), Arduino, LED, subject's specific stimulation.

GLOSARIO

Interfaz cerebro-máquina (BCI)

Un BCI es un sistema de comunicaciones hardware y software que permite controlar un ordenador o dispositivo externo únicamente mediante la actividad cerebral [1].

Electroencefalograma (EEG)

El electroencefalograma mide la actividad eléctrica en el cerebro causada por el flujo de corrientes eléctricas durante las excitaciones sinápticas de las dendritas en las neuronas [2].

Steady-State Visually Evoked Potentials (SSVEP)

Los "Visual Evoked Potentials" (VEP) son modulaciones en la actividad del cerebro que suceden en el córtex visual después de recibir un estímulo visual. Estas modulaciones son relativamente fáciles de detectar, ya que la amplitud de los VEPs se incrementa considerablemente cuando el estímulo se acerca al centro del campo de visión [3]. Los "Steady-State Visually Evoked Potentials" (SSVEP) son VEPS en respuesta a estimulación en frecuencias específicas que generan actividad a la misma frecuencia que tiene el estímulo [4].

Arduino

Arduino es una plataforma de hardware libre, de bajo coste y altas prestaciones, basada en una placa con un microcontrolador y un entorno de desarrollo sencillo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios [5].

LED

De las siglas Light-Emitting diode, 'diodo emisor de luz', se trata de un dispositivo de señalización utilizado en electrónica, ordenadores, paneles numéricos, etc. En este proyecto se utilizan LEDs para evocar SSVEPs y poder reconocer la intención de una persona en cuanto al manejo de un BCI [6].

Artefacto

Esta palabra, utilizada en el ámbito del trazado de un aparato registrador, se refiere a toda variación no originada por el órgano cuya actividad se desea registrar [7].

BCI Illiteracy

Fenómeno que denomina el hecho de que una interfaz cerebro –máquina no funcione para un porcentaje de sujetos no despreciable, de alrededor de entre un 15% y un 30% [8].

Emotiv EPOC

Emotiv EPOC es un casco comercial de registro de EEG que actúa como una interfaz personal para que las personas interactúen con ordenadores. Realiza un registro de electroencefalografía de alta resolución multicanal, utilizando sensores que transforman la actividad cerebral en señales eléctricas.

Impresión 3D

Una impresión 3D es un grupo de tecnologías de fabricación por adición donde un objeto tridimensional es creado mediante la superposición de capas sucesivas de material [9].

Tetraplejia

La tetraplejia es la ausencia de movimiento y de sensibilidad en las cuatro extremidades (brazos y piernas). Es causada por algún daño a la médula espinal, y suele producirse como resultado de una lesión en las vértebras torácicas o por encima [10].

Afasia

La afasia es un trastorno causado por lesiones en las partes del cerebro que controlan el lenguaje. Puede dificultar la lectura, la escritura y la capacidad de expresar lo que se desea decir [11].

Ictus cerebral

Son un conjunto de trastornos transitorios o permanentes que afectan al cerebro y son producidos por una alteración de la circulación cerebral [10].

Parálisis Cerebral

Es un grupo de trastornos que pueden comprometer las funciones del cerebro y del sistema nervioso como el movimiento, el aprendizaje, la audición, la visión y el pensamiento [11].

ÍNDICE

1. INTRODUCCIÓN	1
1.1 MOTIVACIÓN	1
1.2 OBJETIVOS.....	2
1.3 ESTRUCTURA DE LA MEMORIA.....	3
2. ESTADO DEL ARTE.....	5
2.1 BCI: INTERFAZ CEREBRO – MÁQUINA.....	5
2.1.1 <i>Etapas de un BCI</i>	6
2.1.2 <i>Aplicaciones</i>	10
2.2 EEG: OBTENCIÓN DE LA SEÑAL.....	11
2.2.1 <i>Clasificación de los electrodos de un EEG</i>	12
2.2.2 <i>Artefactos</i>	12
2.3 SSVEP: STEADY-STATE VISUALLY EVOKED POTENTIALS.....	13
3. DISEÑO Y DESARROLLO.....	15
3.1 ELECCIÓN DEL DISPOSITIVO A DESARROLLAR.....	15
3.1.1 <i>Implementación enteramente software</i>	15
3.1.2 <i>Implementación enteramente hardware</i>	16
3.1.3 <i>Implementación híbrida</i>	16
3.2 HARDWARE.....	17
3.2.1 <i>Arduino Yún</i>	17
3.2.2 <i>LEDs</i>	21
3.2.3 <i>Power Bank</i>	22
3.2.4 <i>Placa Matriz</i>	22
3.2.5 <i>Diseño de la caja</i>	24
3.2.6 <i>Características del diseño final</i>	26
3.2.7 <i>Casco de registro de EEG Emotiv EPOC</i>	27
3.3 SOFTWARE.....	29
3.3.1 <i>Configuración de los LEDs</i>	29
3.3.2 <i>Comunicación con la interfaz cerebro-máquina</i>	31
3.3.3 <i>Almacenamiento de datos</i>	34
3.3.4 <i>LEDs verdes de indicación</i>	36
3.3.5 <i>Código final</i>	38
3.3.6 <i>Procesado de la señal</i>	39
4. EXPERIMENTOS Y RESULTADOS.....	41
4.1 VALIDACIÓN DE LA PRECISIÓN TEMPORAL DEL ESTÍMULO SSVEP.....	42
4.2 IMPLEMENTACIÓN DE LA COMUNICACIÓN VÍA WI-FI.....	43
4.3 EXPERIMENTOS DE VALIDACIÓN DEL ESTÍMULO CON EMOTIV.....	44
4.3.1 <i>Primera fase: Barrido en frecuencia</i>	46
4.3.2 <i>Segunda fase: Fijación de características</i>	49
4.4 PRUEBAS EN PERSONAS CON DISCAPACIDAD.....	52
5. CONCLUSIONES Y ESTUDIO FUTURO.....	57
6. REFERENCIAS.....	59
ANEXOS.....	I
ANEXO I: CÓDIGO DE ENCENDIDO DE LOS LEDs.....	I
A) <i>Encendido de un LED utilizando retardos</i>	I
B) <i>Encendido de un LED sin retardos</i>	I
ANEXO II: CÓDIGO DEL SISTEMA CLIENTE - SERVIDOR.....	III
ANEXO 3: HABILITACIÓN DE LA TARJETA MICROSD.....	V
A) <i>Función “guardar”</i>	V
B) <i>Función “cargar”</i>	VI

ÍNDICE DE TABLAS

TABLA 1: ANÁLISIS DE LOS DIFERENTES MÉTODOS DE ADQUISICIÓN DE SEÑAL EN UN BCI (ADAPTADA DE [15])	7
TABLA 2: FACTORES PARA LA ELECCIÓN DE LOS DIFERENTES PROCEDIMIENTOS DE ESTIMULACIÓN (ADAPTADO DE [15]).	9
TABLA 3: CARACTERÍSTICAS DEL MICROCONTROLADOR AVR DE ARDUINO YÚN.	19
TABLA 4: CARACTERÍSTICAS DEL MICROPROCESADOR LINUX DE ARDUINO YÚN.	19
TABLA 5: CARACTERÍSTICAS FINALES DE DISEÑO DEL DISPOSITIVO ESTIMULADOR PARA BCIS BASADOS EN SSVEPs.	26
TABLA 6: COMPARATIVA DE DIFERENTES CÓDIGOS DE PROGRAMA MEDIDOS CON EL OSCILOSCOPIO.	43

ÍNDICE DE FIGURAS

FIGURA 1: REPRESENTACIÓN ESQUEMÁTICA DE LAS ETAPAS DE PROCESAMIENTO EN UN BCI	6
FIGURA 2: ALGORITMO DE REGRESIÓN LINEAL VS ALGORITMO DE CLASIFICACIÓN.	10
FIGURA 3: ESTÍMULO SSVEP.	13
FIGURA 4: FUNCIONAMIENTO DEL MICROPROCESADOR EN ARDUINO YÚN.	18
FIGURA 5: VISTA FRONTAL DE LA PLACA ARDUINO YÚN.	20
FIGURA 6: VISTA LATERAL DE LA PLACA ARDUINO YÚN.	22
FIGURA 7: VISTA FRONTAL DE LA PLACA MATRIZ DONDE SE COLOCARAN LOS LEDs PARA CONECTARLOS AL ARDUINO YÚN. .	23
FIGURA 8: DISEÑO DE LA CAJA EN CST PARA SU IMPRESIÓN 3D.	24
FIGURA 9: DISEÑO DE LA CAJA EN CST SIN ELEMENTOS.	25
FIGURA 10 A, B Y C: ASPECTO FRONTAL DE LA CAJA EN TRES NIVELES.	26
FIGURA 11: VISTA LATERAL DE LA CAJA: RANURA PARA CARGAR, BOTÓN DE ENCENDIDO Y SLOT PARA LA TARJETA MICROSD	26
FIGURA 12: DISPOSICIÓN DE LOS ELECTRODOS SEGÚN EL SISTEMA INTERNACIONAL 10-20.	27
FIGURA 13: CAPTURA DE PANTALLA DEL TESTBENCH DE EMOTIV.	27
FIGURA 14: CAPTURA DEL OSCILOSCOPIO EN LA PRUEBA DE PRECISIÓN TEMPORAL DE UNA FRECUENCIA DE 6 HZ.	42
FIGURA 15: ETAPAS DEL SISTEMA BCI CON EL SISTEMA DE ESTIMULACIÓN VISUAL DESARROLLADO EN ESTE TRABAJO.	44
FIGURA 16: PRIMERA ETAPA DEL BCI: ADQUISICIÓN DE LA SEÑAL	45
FIGURA 17: RESULTADOS DEL ANÁLISIS DE EEG EN RESPUESTA A LA ESTIMULACIÓN CON EL DISPOSITIVO DISEÑADO EN EL RANGO DE 6 A 13 HZ.	47
FIGURA 18: DETALLE DE LA RESPUESTA A 11 HZ EN EL EJEMPLO MOSTRADO EN LA FIGURA 17.	47
FIGURA 19: RESULTADOS PARA UN SUJETO QUE NO RESPONDE ADECUADAMENTE AL ESTÍMULO.	48
FIGURA 20: EJEMPLO DE PRUEBA PARA LA SELECCIÓN DEL DISIPADOR.	49
FIGURA 21: RESULTADOS DE LA ESTIMULACIÓN PARA UNA SEPARACIÓN ENTRE LEDs DE 8 CM.	50
FIGURA 22: COMPARACIÓN DE RESULTADOS CON DIFERENTE INTENSIDAD DE BRILLO	51
FIGURA 23: BARRIDO DE FRECUENCIAS EN LA PACIENTE A	53
FIGURA 24: ANÁLISIS DE PREGUNTAS DE “SÍ” Y “NO” EN LA PACIENTE A.	54
FIGURA 25: PRUEBA CON CUATRO LUCES EN LA PACIENTE A.	54
FIGURA 26: BARRIDO DE FRECUENCIAS EN LA PACIENTE C.	55
FIGURA 27: ANÁLISIS DE PREGUNTAS DE “SÍ” Y “NO” EN LA PACIENTE C.	56
FIGURA 28: ANÁLISIS CON CUATRO OPCIONES.	56

1. Introducción

1.1 Motivación

Los primeros ordenadores comenzaron a construirse a principios del siglo XX, pero no fue hasta 1939 cuando se creó el primer ordenador electrónico en el Iowa State College (EEUU) [12]. Más tarde, en 1968, aparecieron las dos interfaces clásicas que nos han permitido transmitir nuestras intenciones y órdenes al ordenador: el ratón y el teclado electrónico [13]. Su simplicidad y facilidad a la hora de utilizarlos, además de su bajo coste, han hecho que estas dos interfaces sean las más comunes para la comunicación persona-ordenador desde hace más de 40 años. Sin embargo, estos dispositivos tienen sus limitaciones. A día de hoy seguimos teniendo que pulsar cada una de las letras de cada palabra, con el tiempo que eso conlleva, y el movimiento del ratón no es una forma natural para comunicarse.

El avance en la tecnología ha permitido solucionar, o al menos buscar una alternativa a estos problemas. Durante estos últimos años se ha avanzado considerablemente en las técnicas de reconocimiento de voz y los sistemas de pantallas táctiles, más intuitivas y sencillas de manejar, y mejor acondicionadas a la naturaleza del ser humano. Estas nuevas interfaces, sin embargo, resultan inútiles para personas discapacitadas o con alguna minusvalía que limite su desarrollo motor. Sin forma de comunicarse verbalmente, y sin capacidad de realizar movimientos físicos voluntariamente, este sector de personas queda excluido de la utilización de dichas interfaces y, consecuentemente, de hacer uso de la tecnología de una forma fácil y accesible.

En esta línea de investigación, se han desarrollado nuevos dispositivos que actúen como interfaces entre el ser humano y la tecnología, permitiendo recibir y procesar nuevos parámetros de entrada además de los que se han mencionado anteriormente. Un dispositivo determinado puede utilizar el electroencefalograma (EEG) para actuar como una interfaz hombre-máquina (también denominado BCI por sus siglas en inglés, *Brain-Computer Interface*). Registrando la actividad cerebral de un sujeto, se pueden traducir esas señales en instrucciones y ser capaces de controlar un ordenador o un dispositivo como una silla de ruedas, un brazo protésico, etc.

No obstante, dependiendo de cada persona y de cada set de instrucciones, el sistema BCI a utilizar será efectivo en mayor o menor medida. Un porcentaje cercano al 20% de la población no es capaz de aprender a manejar un BCI [8] [14].

El propósito de este trabajo es desarrollar un dispositivo de estimulación visual que permita generar en el cerebro determinadas ondas que puedan ser detectadas por el electroencefalograma, interpretadas y traducidas en instrucciones, de manera que se pueda establecer un medio de comunicación con personas que de lo contrario no tendrían forma de hacerse entender.

1.2 Objetivos

El objetivo de este trabajo es el diseño y desarrollo de un dispositivo de estimulación visual específica para cada sujeto que lo utilice, buscando un resultado compacto, portátil y de bajo coste que se pueda integrar en una interfaz cerebro-máquina y suponga un medio de comunicación para personas con discapacidades físicas y motoras.

Para ello, se han ido siguiendo una serie de hitos progresivos que han llevado al resultado final y que se explican a continuación:

- En primer lugar se ha realizado una revisión del estado del arte, estudiando la tecnología existente hasta el momento y evaluando resultados de análisis anteriores, con el objetivo de conocer la situación actual y actuar en consecuencia.
- Una vez se ha entendido cuál es esta situación, el siguiente objetivo es decidir el tipo de dispositivo que se plantea desarrollar, evaluando las ventajas y desventajas de cada modo de hacerlo para seleccionar el más adecuado.
- El siguiente paso es realizar un prototipo inicial no configurable, que verifique que la estimulación con distintas frecuencias se establece correctamente, y que resulta eficaz para que distintos sujetos lo utilicen en un BCI basado en SSVEPs; para ello se utilizará la plataforma Arduino Yún.
- A partir de este prototipo, se ha desarrollado el diseño para ser capaz de adaptarse a cada usuario, seleccionando las mejores frecuencias para cada sujeto y estableciéndolas en los LEDs mediante una comunicación inalámbrica entre el ordenador y el dispositivo, que se realizará vía Wi-Fi. Esta estructura formada por la placa Arduino y los LEDs se integrará dentro de una caja diseñada con este propósito para un diseño pequeño y compacto.
- Llegados a este punto, se realizan experimentos y se evalúan los resultados utilizando un casco de Emotiv que permite registrar con comodidad el electroencefalograma, determinando la efectividad del dispositivo y la validez de su utilización. A través del casco Emotiv se establecen también los parámetros de diseño que mejor se adaptan a los sujetos y que permanecerán fijos, como la distancia entre LEDs, color, brillo...
- El último objetivo consiste en validar la utilidad del dispositivo como medio de comunicación, y por ello se realizan algunas pruebas en el centro "Cottolengo" para personas discapacitadas.

1.3 Estructura de la Memoria

La estructura de esta memoria está organizada en siete capítulos, según se describe a continuación:

- En el *Capítulo 2* se realizará un estudio del estado del arte, revisando brevemente el trabajo que hasta ahora se ha realizado en este campo. Se definirán y explicarán los conceptos básicos de una interfaz cerebro-máquina, así como un electroencefalograma y los potenciales visuales evocados.
- A lo largo del *Capítulo 3* se contempla el diseño y desarrollo del dispositivo de estimulación SSVEP, comenzando por la elección del dispositivo a desarrollar y siguiendo con el desarrollo tanto hardware como software, explicando cada componente y el código desarrollado.
- A continuación, en el *Capítulo 4* se observan los resultados obtenidos, comenzando por pruebas que verifican que el dispositivo funciona correctamente y con la precisión temporal esperada; pasando a continuación a evaluar el dispositivo en un grupo de voluntarios. Se realizan numerosas pruebas en un total de 23 sujetos, y se interpretan estos resultados. También se realizan pruebas en un centro de personas discapacitadas para verificar la funcionalidad del dispositivo como sistema de comunicación.
- Por último, en el *Capítulo 5* se comentan las conclusiones extraídas de este proyecto, y se sugieren algunas líneas de trabajo para continuar este estudio en un futuro.
- A continuación se listan las referencias que se han utilizado durante el desarrollo de este trabajo.
- Finalmente, en los *Anexos* se comentan los fragmentos más importantes del código que se ha utilizado, para un mejor entendimiento de este.

2. Estado del arte

Para la realización de este estado del arte, se comenzará hablando sobre las interfaces cerebro-máquina, describiendo qué son, en qué etapas se dividen y cuáles son sus aplicaciones.

Una vez entendida la funcionalidad básica de los BCIs, se explicará el electroencefalograma como sistema de adquisición de señales para determinar la intención de una persona en relación al control de un dispositivo, explicando qué es, qué tipos de electrodos existen y los artefactos que pueden aparecer.

Por último, se hablará de los sistemas de estimulación SSVEP, que será el método a desarrollar durante este trabajo, explicando cómo funcionan y qué tipos existen.

2.1 BCI: Interfaz Cerebro – Máquina.

Una interfaz cerebro-máquina (BCI) es un sistema de comunicaciones hardware y software que permite controlar un ordenador o dispositivo externo únicamente mediante la actividad cerebral. El objetivo de la investigación en estas interfaces es poder proporcionar capacidades de comunicación a personas con serias discapacidades provocadas por una parálisis total o un bloqueo causado por un trastorno neuromuscular y neurológico, tales como la esclerosis lateral amiotrófica o una lesión de la médula espinal [15].

Una interfaz cerebro-máquina permite al usuario interactuar con su entorno sin tener que hacer uso de nervios periféricos y músculos, sino únicamente a partir de las señales de control generadas por la actividad encefalográfica. De esta forma se crea un nuevo canal, en el que no intervienen los músculos, para transmitir las intenciones de una persona a dispositivos externos como ordenadores, sintetizadores de voz, dispositivos de asistencia y prótesis neurales. Estas características permitirán a aquellas personas que sufren discapacidades motoras mejorar su calidad de vida, y al mismo tiempo reducir el coste de los cuidados intensivos [16].

La tecnología BCI en su origen fue rechazada por la investigación científica seria debido a que la idea de descifrar pensamientos e intenciones por medio de la actividad cerebral se tenía como remota e irrealizable [17]. La investigación en el campo de la actividad cerebral se había limitado al análisis de desórdenes neurológicos o exploración de las funciones cerebrales. Sin embargo, este contexto ha cambiado radicalmente en los últimos 20 años, considerándose hoy en día un campo prometedor con más de 100 grupos de investigación a lo largo del mundo.

2.1.1 Etapas de un BCI

Una interfaz cerebro-máquina es un sistema capaz de reconocer unos determinados patrones en las señales obtenidas del cerebro, siguiendo cinco etapas consecutivas: primero se adquiere la señal a través de distintos sistemas de medición. Después, la señal obtenida se preprocesa para acondicionarla y extraer sus características, que se interpretan y clasifican para determinar la orden que el usuario desea ejecutar, y por último se envía dicha orden a la interfaz de control [18].

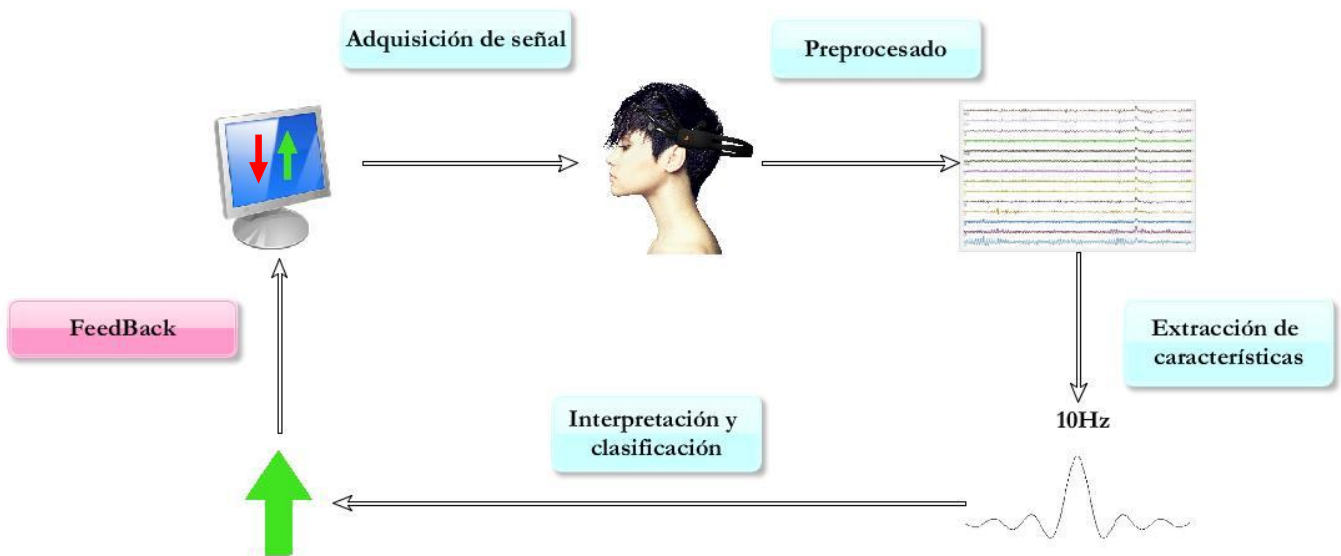


Figura 1: Representación esquemática de las etapas de procesamiento en un BCI

La primera etapa es la de adquisición de señal. Existen dos tipos de actividad cerebral que pueden ser monitorizadas por distintos sistemas de obtención de señal:

La primera es la actividad electrofisiológica, que se genera cuando dos neuronas intercambian información mediante transmisores electroquímicos. Estas neuronas generan corrientes iónicas que fluyen dentro y a través del resto de neuronas, y estas corrientes pueden ser registradas mediante sistemas como un electroencefalograma (EEG), electrocorticograma, magnetoencefalograma o un sistema de adquisición de señales eléctricas en neuronas individuales [19].

El segundo tipo de actividad cerebral que se puede monitorizar es la respuesta hemodinámica [20]. La respuesta hemodinámica es el proceso mediante el cual la sangre libera más glucosa a las neuronas activas que a las inactivas. La glucosa y oxígeno que se entrega a través del flujo de sangre en las zonas activas del cerebro se pueden medir con métodos de neuroimagen, como por ejemplo una resonancia magnética o una espectroscopia infrarroja.

Aunque también existen BCIs que funcionan mediante resonancias [20] [21], se suele elegir la actividad electrofisiológica como señal a procesar, ya que se considera un sistema directamente relacionado con la actividad neuronal, al contrario que la respuesta hemodinámica que se considera indirecta. Así pues, el siguiente paso es elegir qué sistema se va a utilizar para monitorizar dicha actividad neuronal.

La mayoría de los BCI utilizan el electroencefalograma, que tiene una alta resolución temporal, bajo coste, portabilidad alta y poco riesgo para el usuario. Existen otros métodos para obtener la señal, como el magnetoencefalograma, electrocorticograma o sistemas de adquisición de señales eléctricas en neuronas individuales [15], [22], [23].

El magnetoencefalograma registra la actividad magnética en lugar de la eléctrica a través de una inducción magnética. Tiene menos distorsiones provocadas por el cráneo y el cuero cabelludo, pero no existen sistemas portátiles y son muy costosos, resultando imposibles para el uso diario [21].

El electrocorticograma es un sistema invasivo en el que los electrodos se colocan directamente en la superficie del cerebro, pero tiene la ventaja de tener una alta resolución tanto espacial como temporal, así como mayores amplitudes y menor vulnerabilidad a artefactos [24]. Los registros invasivos se utilizan en pacientes con patologías severas.

Por último, el sistema de adquisición de señales eléctricas en neuronas individuales registra la actividad cerebral dentro de la materia gris del cerebro. Se trata también de un sistema invasivo, pero tiene una resolución tanto temporal como espacial alta, y las señales obtenidas son más fáciles de procesar que las de un EEG. Sin embargo, la calidad de la señal se puede ver afectada por la reacción del tejido cerebral, que puede con el tiempo dañar el microelectrodo [25].

A continuación se expone una tabla resumiendo las propiedades de cada sistema de adquisición de señal, con sus ventajas e inconvenientes:

Método	Actividad registrada	Resolución temporal	Resolución espacial	Riesgo	Portabilidad
EEG	Eléctrica	~0.05s	~10 mm	No invasivo	Portátil
MEG	Magnética	~0.05s	~5 mm	No invasivo	No Portátil
ECoG	Eléctrica	~0.003s	~1 mm	Invasivo	Portátil
Adquisición en neuronas individuales	Eléctrica	~0.003s	~0.5mm(LFP) ~0.1mm(MUA) ~0.05mm(SUA)	Invasivo	Portátil

Tabla 1: Análisis de los diferentes métodos de adquisición de señal en un BCI (adaptada de [15]).

Con el electroencefalograma realizamos la primera etapa (adquisición de señal), pero para las siguientes etapas de preprocesamiento y extracción de características es necesario determinar qué señal se va a utilizar para determinar las intenciones e instrucciones del usuario. Las señales que emite el cerebro incluyen toda la actividad neuronal en cada momento, la mayoría de ellas aún desconocidas e incomprensibles para el ser humano. Sin embargo, el origen fisiológico de algunas señales sí que ha sido decodificado de forma que las personas puedan aprender a modularlas según su voluntad, permitiendo al sistema BCI interpretar las órdenes que el usuario desee enviar, y son estas señales las que son objeto de estudio y susceptibles de ser utilizadas como posibles señales de control. Según su origen, las señales se pueden clasificar como señales endógenas o exógenas.

Son señales exógenas aquellas en las lo que genera la instrucción es un estímulo externo, como por ejemplo una luz parpadeante a una frecuencia específica. Estos sistemas tienen la ventaja de que no requieren de mucho entrenamiento y resultan fáciles y rápidos de utilizar, aunque necesitan que se preste una atención permanente al estímulo externo, lo que provoca cansancio y molestias en algunos sujetos. Los sistemas que generan estímulos exógenos son los VEP y los P300 (llamados así porque generan potenciales con una latencia de unos 300mS). Los VEPs son modulaciones que ocurren en el córtex al recibir un estímulo visual [26], como puede ser un LED oscilando a una determinada frecuencia o una señal a través del monitor. Estas modulaciones se pueden detectar con facilidad y permiten un elevado número de elecciones a la hora de asignar estímulos a instrucciones. Por otro lado, el P300 es también un estímulo exógeno en el que se muestran al usuario varias opciones que se van resaltando una cada vez. Cuando la instrucción que el usuario quiere escoger es la que está resaltada, aparece una deflexión positiva en el electroencefalograma, y el sistema detecta que ese es el estímulo escogido. Al igual que el SSVEP, el P300 tiene un alto número de elecciones posibles [27].

Las señales endógenas surgen del propio sujeto, que mediante un entrenamiento y tiempo de práctica aprende a generar patrones cerebrales específicos que se pueden registrar en el BCI. Este método tiene la ventaja de que una vez aprendido es independiente de cualquier estimulación, se puede ejecutar con libertad y es muy útil para personas con órganos sensoriales afectados, pero el inconveniente de que se puede tardar meses o años en aprender a manejarlo, y no todos los usuarios pueden aprender a controlarlo. Además, se necesitan EEG multicanales para un buen rendimiento, y la tasa transferencia información en bits es más baja que en los endógenos. Los sistemas que utilizan señales endógenas son los SCPs y los ritmos sensoriomotores.

Los SCPs (Slow Cortical Potentials) son movimientos lentos en el EEG que duran entre un segundo y varios segundos. Son señales que proceden de la actividad cortical, y pueden ser negativas o positivas, siendo esa la diferencia entre las órdenes binarias que un SCP puede detectar [28].

Los ritmos sensoriomotores son modulaciones sincronizadas con actividades motoras, es decir, para generar una orden basta con que el sujeto piense en realizar un movimiento. Aunque dicho movimiento no llegue a realizarse, al

concentrarse en ello genera actividad que el EEG puede registrar y traducir en la orden [25].

El sistema que la mayoría de BCIs utiliza para generar la señal que posteriormente se extraerá con el electroencefalograma es el SSVEP debido al elevado número de instrucciones que se pueden implementar, su facilidad a la hora de aprender y su elevada tasa de transmisión de información (ver tabla 2).

Señal	Tipo	Número de elecciones	Entrenamiento	Tasa de transmisión de información
VEP	Exógena	Altas	No	60-100 bits/min
P300	Exógena	Bajas (2 o 4)	No	20-25 bits/min
SCP	Endógena	Altas	Sí	5-12 bits/min
S.R	Endógena	Bajas (2-5)	Sí	3-35 bits/min

Tabla 2: Factores para la elección de los diferentes procedimientos de estimulación (adaptado de [15]).

Después de determinar qué características son las que se están buscando, los últimos pasos son extraer dichas características y clasificar la señal que será enviada a la interfaz de control. Debido a que en el cerebro a cada instante se realizan una variedad de tareas que incluyen pensamientos, órdenes motoras voluntarias e involuntarias, coordinación de las actividades del resto del organismo, etc. se hace complicado distinguir una característica determinada en una señal, ya que está solapada con el resto de tareas mencionadas anteriormente. Por esta razón, a la hora de extraer las características es necesario algo más que un método simple como podría ser un filtro paso banda. La información se recibe a través de varios canales, no todos ellos de interés, así que se utilizan técnicas de reducción como el análisis de componentes principales o análisis de los componentes independientes, que reducen las dimensiones de los datos originales eliminando y reduciendo la información redundante [29].

Para clasificar la información ya analizada y reducida, se pueden utilizar tanto algoritmos de regresión como de clasificación, siendo los algoritmos de clasificación los más utilizados [30]. Los algoritmos de regresión se basan en las características extraídas para predecir la intención del sujeto, mientras que los de clasificación separan esas características, definiendo una frontera entre las instrucciones.

En la figura 2, se puede observar un ejemplo de un algoritmo de regresión en azul, siendo la recta que une, en la medida de lo posible, los puntos obtenidos. El algoritmo de clasificación, sin embargo, agrupa dichos puntos en bloques, en este caso en tres: un cluster azul, uno verde y uno rojo.

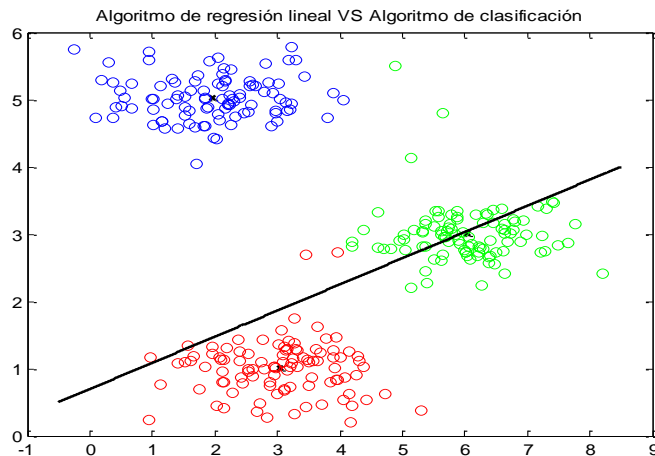


Figura 2: Algoritmo de regresión lineal VS Algoritmo de Clasificación.

Para un caso en el que hay que determinar una instrucción entre dos posibles, tanto clasificación como regresión precisan de una sola función. Sin embargo, para cuatro instrucciones posibles, el algoritmo de clasificación necesitaría tres funciones, una por cada frontera, mientras que para la regresión seguiría bastando con una. Por este motivo suele utilizarse regresión para casos en los que hay varias instrucciones, y clasificación para instrucciones de dos casos. Además, la aproximación por regresión resulta útil en casos en los que haya un feedback continuo, como el control de un cursor [18].

2.1.2 Aplicaciones

Una vez concluidas las etapas que conforman el funcionamiento de un BCI, cabe mencionar cuál es su función y en qué aplicaciones resulta beneficioso. Dependiendo del sujeto que lo va a utilizar, se pueden distinguir dos tipos de BCIs: el grupo principal está enfocado hacia aquellos que serán utilizados por personas discapacitadas hasta un grado de parálisis casi total, pero con los suficientes recursos para controlar el sistema. Existe también una línea de investigación hacia BCIs más rápidos que se utilizarían para gente sana. En este caso las aplicaciones BCI son limitadas, ya que en estos casos son más prácticos y fáciles de utilizar los tradicionales ratón y teclado, así que estos BCIs se utilizan en su mayoría con fines de entretenimiento, por ejemplo en videojuegos.

Las principales aplicaciones de los BCIs para las personas discapacitadas son aquellas relacionadas con la comunicación, ya que es una actividad esencial de los seres humanos. En este ámbito, se han creado teclados virtuales en los que el usuario puede seleccionar una letra del alfabeto a través del BCI, por ejemplo a través de un P300 [31], [32]. También se pueden controlar cursores mediante SCPs que se muevan sobre el teclado y seleccionen la letra[33].

Otro campo de aplicaciones BCI es el de la restauración de la actividad motora. Se pueden crear dispositivos que, mediante instrucciones ordenadas por el usuario, puedan por ejemplo agarrar y transportar objetos, así como manejar sillas de ruedas [34]. También se puede controlar el medio en el que viven, permitiéndole manejar mandos de televisión, encender y apagar luces de la casa o la temperatura [15].

2.2 EEG: Obtención de la señal.

La actividad cerebral se registra para determinar las intenciones del usuario. Para esto, el BCI necesita de un sistema que monitorice esta actividad cerebral, para posteriormente traducir esta información a señales eléctricas manejables. Se han mencionado diferentes métodos para conseguir estas señales, y el más utilizado de ellos es el electroencefalograma (EEG).

Un electroencefalograma mide la actividad eléctrica en el cerebro causada por el flujo de corrientes eléctricas durante las excitaciones sinápticas de las dendritas en las neuronas, y es extremadamente sensible a los efectos de corrientes secundarias [35]. Las señales de un EEG se registran fácilmente de forma no invasiva a través de unos electrodos colocados en el cuero cabelludo. El hecho de que la señal tenga que cruzar el cuero cabelludo, el cráneo y demás capas hace que la señal tenga una calidad baja. También afecta a la calidad de la señal el ruido de fondo generado tanto en el interior del cerebro como exteriormente en el cuero cabelludo.

Las neuronas están polarizadas eléctricamente por su membrana de transporte, que se encarga de bombear iones. Estas células están constantemente intercambiando iones con el medio extracelular y produciendo corrientes eléctricas. Mediante un casco de EEG se puede medir la diferencia de potencial a lo largo del tiempo entre el electrodo activo y el electrodo de referencia. Esta técnica de medida se conoce como registro bipolar [35].

También existe un tercer electrodo, conocido como electrodo de tierra, que se utiliza para medir la diferencia de potencial entre el punto activo y el de referencia. Estos tres electrodos componen la configuración mínima para registrar una señal EEG correctamente, y se conocen como registro unipolar [15]. En aplicaciones clínicas, se utilizan típicamente 19 electrodos colocados alrededor del cráneo para realizar las mediciones, aunque si es necesario aumentar la resolución espacial en un área particular del cerebro se puede aumentar el número de electrodos, llegando hasta 256 electrodos colocados de forma más o menos uniforme en el cuero cabelludo.

Un sistema EEG está compuesto de electrodos, amplificadores, conversores A/D y un dispositivo de grabación. Los electrodos adquieren la señal del cuero cabelludo, y los amplificadores procesan la señal analógica para aumentar la amplitud de la señal EEG para que el convertor A/D pueda digitalizar la señal con mayor precisión. Finalmente, el dispositivo de grabación, que puede ser un ordenador personal, almacena y procesa estas señales.

2.2.1 Clasificación de los electrodos de un EEG

Existen diferentes tipos de electrodos que pueden conformar un EEG, que se clasifican en función del material con el que están hechos. Los más comunes son los de plata y oro, existiendo también electrodos hechos de estaño, más baratos pero menos usados hoy en día.

Los electrodos también pueden clasificarse en activos o pasivos. Los electrodos activos tienen en su interior un amplificador, que mejora la calidad de la señal y evita tener que poner geles conductores en el cuero cabelludo. Los pasivos son más simples y tienen un menor coste, pero requiere que cada electrodo sea acondicionado manualmente para captar la actividad cerebral, y precisa de geles conductores.

2.2.2 Artefactos

Al analizar las señales registradas por el EEG hay que tener en cuenta los distintos artefactos que pueden surgir en las señales. Se denomina artefacto a toda variación no originada por el órgano cuya actividad se desea registrar. Los datos registrados en un EEG casi siempre están contaminados por estos elementos, y su amplitud puede llegar a ser relativamente amplia en comparación con las señales que resultan de interés. Estos artefactos se pueden clasificar en dos categorías: artefactos fisiológicos y artefactos técnicos [15].

Los artefactos técnicos se atribuyen en su mayoría a cambios en la impedancia de los electrodos y a ruidos en la potencia de la línea, que normalmente se pueden evitar mediante un correcto filtrado y protección. Debido a que tienen una solución más fácil, los estudios se suelen centrar en los artefactos fisiológicos, que suponen un reto mayor al distorsionar más la señal obtenida.

Los artefactos fisiológicos se deben a actividades musculares, oculares o cardíacas, como por ejemplo el movimiento de los ojos, parpadeo, movimientos musculares involuntarios extraoculares, el latido del corazón, contracción de la mandíbula... Cuando el movimiento es voluntario, estos artefactos se pueden corregir simplemente pidiendo al sujeto que evite dicho movimiento, como por ejemplo parpadear lo menos posible. Sin embargo, artefactos como el pulso cardíaco son involuntarios y no se pueden corregir de esta manera. En este caso, se pueden monitorizar las señales de los artefactos de forma separada, y sustraer esta señal de la señal registrada por el EEG.

2.3 SSVEP: Steady-State Visually Evoked Potentials.

Para poder registrar correctamente la orden que se registra en el EEG, a menudo se utilizan técnicas exógenas (de las que se ha hablado en el apartado 2.1) que provocan en el cerebro la respuesta buscada. Para generar este estímulo, una técnica utilizada son los "evoked potentials", en el que se registra la actividad cerebral mientras se presenta un estímulo, ya sea visual, auditivo o somatosensorial. Como se ha definido anteriormente, los "Visual Evoked Potentials" (VEP) son modulaciones en la actividad del cerebro que suceden en el córtex visual después de recibir un estímulo visual [26]. Estas modulaciones son relativamente fáciles de detectar, ya que la amplitud de los VEPs se incrementa considerablemente cuando el estímulo se acerca al centro del campo de visión.



Figura 3: Estímulo SSVEP.

El sujeto mira una luz parpadeante a una frecuencia específica y en su corteza visual se produce una oscilación a esa misma frecuencia, la cual se puede detectar en un registro de EEG. Este fenómeno se puede utilizar para determinar la intención del sujeto cuando mira a una fuente de luz específica que codifica un comando.

Los VEP se pueden clasificar en VEP transitorios (TVEPs) y de estado estacionario, "steady-state" (SSVEPs) [36]. Se considera TVEP cuando la frecuencia de estimulación visual es menor de 6 Hz, mientras que para un rango mayor de frecuencias produce un SSVEP. Los SSVEPs corresponden a una actividad neuronal oscilatoria que tiene la misma frecuencia del estímulo. Los sistemas BCI basados en SSVEP permiten al usuario seleccionar un objetivo utilizando la mirada. La estimulación visual consiste en varias fuentes de luz parpadeantes a distintas frecuencias, cada una de las cuales representa un comando. Sólo con fijar la mirada en el objetivo el BCI puede identificar mediante un análisis de características espectrales del SSVEP, y esto lo convierte en una de las modalidades más prometedoras para un sistema BCI no invasivo.

A la hora de seleccionar las frecuencias de estimulación en un BCI basado en SSVEPs, hay que tener en cuenta que la amplitud será mayor cuanto menor sea

la frecuencia [4]. Por tanto en la zona de bajas frecuencias del SSVEP, que comprende las frecuencias entre 4 y 12 Hz, será más fácil detectar la instrucción. Este rango de frecuencias, por otra parte, tiene el inconveniente de que puede causar molestias y fatiga en el sujeto. Frecuencias más altas reducen la sensación de fatiga pero son más difíciles de identificar.

A la hora de diseñar este tipo de dispositivos, es importante tener en cuenta que la respuesta de cada sujeto al mismo estímulo es diferente y difícil de predecir [37][38]. Aunque en la mayor parte de los BCIs basados en SSVEP se utilizan frecuencias fijas, el Grupo de Neurocomputación Biológica de la EPS-UAM ha demostrado que puede ser mucho más eficaz seleccionar frecuencias específicas para cada sujeto [39]. Es por esto que resultaría útil diseñar un dispositivo de estimulación que sea capaz de admitir una configuración de frecuencias variables y adaptables a cada individuo que utiliza el BCI.

En este trabajo se va a implementar un sistema SSVEP con cuatro instrucciones asociadas a cuatro frecuencias distintas que pueden ser configurables para cada usuario. En el siguiente apartado se explicarán las diferentes maneras de obtener una respuesta SSVEP mediante diferentes métodos, con el objetivo de lograr diseñar un sistema que no sólo sea capaz de configurar las frecuencias deseadas, sino que además se comunique con el sistema BCI y responda en consecuencia.

3. Diseño y Desarrollo

En este capítulo se explicará cómo se han implementado las diferentes partes que componen este trabajo. Se comenzará mostrando los tipos de dispositivos que se podrían haber escogido, y las razones por las que se ha seleccionado la implementación híbrida hardware-software. Después se hablará de las características hardware, y por último del software que permite el correcto funcionamiento y la configuración flexible del sistema de estimulación SSVEP.

3.1 Elección del dispositivo a desarrollar.

Quizá una buena elección del sistema inicial sobre el que se va a realizar todo el trabajo sea la decisión más importante y en la que se va a basar todo el estudio. Así pues, el primer paso en el diseño de este dispositivo es hacer un estudio de las opciones disponibles en este ámbito, evaluando las ventajas e inconvenientes de cada procedimiento y, finalmente, escogiendo la opción que desemboca en los siguientes apartados de este desarrollo.

Partiendo de la base de que lo que se va a diseñar es un dispositivo de estimulación visual, cabe esperar que su implementación sea o bien hardware o bien software. La tercera opción es un desarrollo híbrido que contemple una combinación tanto de hardware como de software. A continuación se explica cada una de ellas.

3.1.1 Implementación enteramente software

En esta primera opción, se trataría de implementar un sistema de estimulación visual haciendo uso de la pantalla de un ordenador. Sin necesidad alguna de hardware adicional, mediante programación se conseguiría un sistema por pantalla que mostrase, por ejemplo, cuatro rectángulos blancos sobre fondo negro que parpadearan cada uno a una frecuencia distinta.

La gran ventaja de que este sistema no se implementase en hardware es que no estaría limitado por hardware. Mediante parámetros se podría configurar la distancia que los separa, el brillo, el número de estímulos, los colores, etc... y adaptar dichos parámetros a cada sujeto.

El problema de los SSVEP cuando se muestran por pantalla es que la frecuencia de oscilación está limitada por la tasa de refresco de la pantalla LCD o del tubo de rayos catódicos del monitor [40][41]. Otro problema a tener en cuenta es que sólo un pequeño número de estímulos a una determinada frecuencia provocan una señal SSVEP intensa, lo que hace que el rango de frecuencias disponibles en SSVEP sea limitado, detectándose respuestas entre los 4Hz y los 80 o 90Hz [4].

Una pantalla tradicional suele tener una tasa de refresco de 75 Hz, lo que deja 75 frecuencias de estimulación disponibles, una por cada submúltiplo de 75 (entero y decimal). De estas frecuencias, sólo 18 son mayores que 4Hz, y por debajo de 4 Hz no se detectan los estímulos. Si además se tiene en cuenta que a estas frecuencias se les tiene que poder aplicar una FFT para su detección, las frecuencias disponibles se reducen a 7. Si se añade que hay frecuencias que son armónicos y subarmónicos de otras frecuencias, hay que descartar esas frecuencias por no saber distinguir cuál es la frecuencia y cuál el armónico. Esto deja con 4 frecuencias restantes, que son las disponibles para generar los estímulos que el BCI basado en SSVEP ha de detectar [39].

Esto se puede solucionar en parte con el protocolo MFSC (*Multiple Frequencies Sequential Coding*), que consigue codificar un objetivo utilizando múltiples frecuencias en secuencia durante cada ciclo [40]. Así se consigue implementar muchos más objetivos con las limitadas frecuencias disponibles, en comparación con los códigos tradicionales de codificación de frecuencias.

3.1.2 Implementación enteramente hardware

Otra solución es sustituir la pantalla del monitor por un circuito compuesto por cuatro LEDs más los componentes sean necesarios para configurar las cuatro frecuencias utilizadas. Tiene el inconveniente de que el sujeto tiene que apartar la mirada de la pantalla para centrarla en un dispositivo adicional, pero elimina el problema de la tasa de refresco del monitor y permite implementar cualquier frecuencia de oscilación en un LED, limitadas únicamente por aquellas frecuencias que el ser humano es capaz de reproducir fielmente en la zona occipital de su cerebro.

No obstante, la distancia entre LEDs, brillo, color e incluso las frecuencias están prefijadas, y no pueden adaptarse a cada usuario. Una vez diseñado el circuito, no admite cambios ni modificación de parámetro alguno.

3.1.3 Implementación híbrida

Por último, una implementación híbrida eliminaría las desventajas que tienen ambos casos anteriores: añadiendo un hardware externo a la pantalla del ordenador se elimina el problema de la tasa de refresco del monitor, y mediante software se puede controlar algunos parámetros de diseños y ser adaptados a cada usuario. Esta opción sigue teniendo la desventaja de que una vez establecida la distancia entre LEDs no se podrá cambiar, por eso habrá que tener especial cuidado a la hora de establecer dicha distancia; el modelo de LED utilizado también será fijo, junto con su color. Sí que se podrá controlar el brillo, la frecuencia de cada LED e incluso modificar estos parámetros durante el transcurso de la sesión BCI.

Por las características que se acaban de enumerar, parece razonable que sea este el modelo escogido para implementar este diseño. Se dividirá entonces el desarrollo en dos partes: hardware y software, que se relacionarán a través de un microcontrolador.

3.2 Hardware

Una vez determinado el enfoque que se le va a dar a este dispositivo, es necesario explicar qué componentes se va a utilizar y por qué. Se ha escogido una placa Arduino Yún para controlar el encendido de los LEDs, alimentada por un Power Bank y todo ello dentro de una caja que se ha imprimido con una impresora 3D.

3.2.1 Arduino Yún

Antes de explicar el porqué de la elección del Arduino Yún es necesario conocer qué es Arduino y cuáles son sus ventajas.

Arduino es una plataforma de desarrollo de computación física de código abierto que está basado en una placa con un microcontrolador y un entorno de desarrollo para crear software para la placa [5].

En función de las aplicaciones que se le vaya a dar, resultará más adecuado escoger un modelo u otro de Arduino, pero en general todos tienen las siguientes características que los hacen destacar sobre otras placas [5]:

- Asequibles: Si se comparan con otras plataformas de microcontroladores, las placas Arduino resultan asequibles tanto si se realiza un montaje manual como si se adquiere ya montada.
- Multi-plataforma: El software de Arduino funciona tanto en Windows como en Macintosh OSX y Linux, mientras que la mayoría de plataformas tienen el sistema operativo limitado a Windows. Esto resulta muy conveniente ya que el BCI con el que se harán las pruebas adquiere las señales en Linux, y será este el sistema operativo en el que se implementará el diseño.
- Entorno de programación simple y directo: Arduino dispone de un entorno de programación cuyo lenguaje sencillo y flexible es similar en algunos aspectos a C++. Dotado de ejemplos ilustrativos y tratándose de una licencia libre preparada para ser ampliada, no resulta complicado familiarizarse con este entorno.

Una vez vistas las cualidades y ventajas de Arduino, cabe mencionar los requisitos que buscamos de cara a su empleo en el dispositivo estimulador:

- Bajo consumo: Dado que el dispositivo SSVEP está pensado para funcionar durante largas sesiones, sería conveniente que el consumo de la placa y del circuito final no fuera excesivamente alto y permitiera una autonomía de varias horas.

- Garantizar la conectividad con los LEDs: Sería contraproducente seleccionar una placa que no fuera compatible con los LEDs que servirán como estímulos. Convendría además que la placa dispusiera de pines de PWM para una buena regulación y configuración de los LEDs.
- Garantizar un dispositivo compacto: Ya que uno de los objetivos de este diseño es que sea manejable, portátil, y sea sostenido por el sujeto, es conveniente que el circuito final sea lo más pequeño posible y que el peso sea mínimo.
- Bajo coste para favorecer la diseminación de esta tecnología, especialmente en el contexto de las personas discapacitadas.
- Comunicación inalámbrica: Aunque no es un requisito imprescindible para la implementación, sí que es recomendable que la conexión se pudiera hacer vía bluetooth, wifi o xbee para un diseño final wireless.

En base a estos requisitos, se realizó un estudio de los diferentes modelos de Arduino. Tanto la placa Arduino Uno como la Leonardo cumplían los requisitos de conectividad, con 6 y 7 pines de PWM respectivamente, medidas y peso razonables para su propósito y un voltaje de entrada en torno a 7-12 Voltios. Ambas placas cumplían con los requisitos mínimos para la correcta implementación del dispositivo SSVEP, sin embargo, la única placa de Arduino que actualmente cuenta con conexión wifi es la Arduino Yún.

La principal ventaja de Arduino Yún con respecto a las demás placas es que además del microprocesador ATmega32u4, que se puede encontrar en Leonardo, dispone de un sistema Linux embebido, ofreciendo una combinación entre un ordenador en red y la facilidad de programación de Arduino.

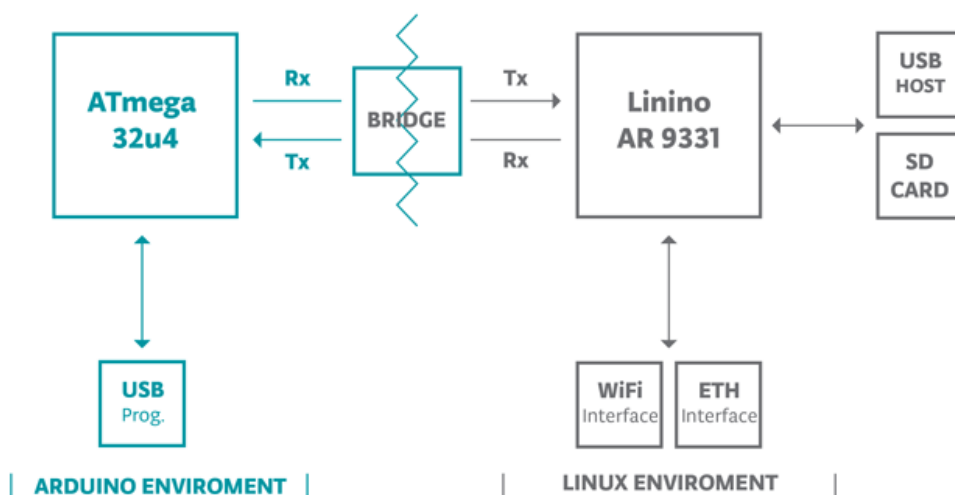


Figura 4: Funcionamiento del microprocesador en Arduino Yún.

Arduino Yún dispone de un procesador Atheros AR9331 con una distribución Linux basada en OpenWRT llamada Linino, con un procesador MIPS que funciona a 400MHz. Este entorno es el que maneja el host USB, la interfaz Wi-Fi y Ethernet, así como una tarjeta microSD que se utilizará en este trabajo para almacenar datos y parámetros de configuración. Dispone también de una memoria RAM de 64MB y memoria Flash de 16MB.

Por su parte, el microprocesador ATmega dispone de comunicación construida directamente con un puerto microUSB, lo que elimina la necesidad que tienen otras placas de tener un procesador secundario. Dispone de una memoria Flash de 32KB, SRAM de 2.5KB y EEPROM de 1KB, con un número máximo de escrituras y borrados. El reloj funciona a 16MHz, y opera a un voltaje de 5V.

Ambos procesadores se comunican entre sí a través de la librería Bridge, permitiendo a los sketches de Arduino que se vuelcan en el ATmega controlar el Linino.

En las siguientes tablas se muestran las principales características de cada uno de los dos procesadores:

Microcontrolador AVR	
Microcontrolador	ATmega32u4
Voltaje de operación	5V
Voltaje de entrada	5V
Pines digitales E/S	7
Canales de PWM	12
Corriente DC máxima por el pin E/S	40mA
Corriente DC para los pines a 3.3V	50mA
Memoria Flash	32KB
Memoria SRAM	2.5KB
Memoria EEPROM	1KB
Velocidad de reloj	16MHz

Tabla 3: Características del Microcontrolador AVR de Arduino Yún.

Microprocesador Linux	
Procesador	Atheros AR9331
Arquitectura	MIPS a 400MHz
Voltaje de operación	3.3V
Ethernet	IEEE 802.3 10/100Mbit/s
Wi-Fi	IEEE 802.11b/g/n
USB Tipo A	2.0 Host/Device
Lector de tarjetas	microSD
Memoria Flash	16MB

Tabla 4: Características del Microprocesador Linux de Arduino Yún

La tarjeta microSD que se va a utilizar en este dispositivo tiene una capacidad de 2GB de memoria.

En cuanto a características físicas, esta placa tiene unas dimensiones de 68 mm x 53 mm x 15 mm, y un peso de 0.0406 Kg. Dispone de cuatro perforaciones para permitir la unión entre la placa y otra superficie o estuche.

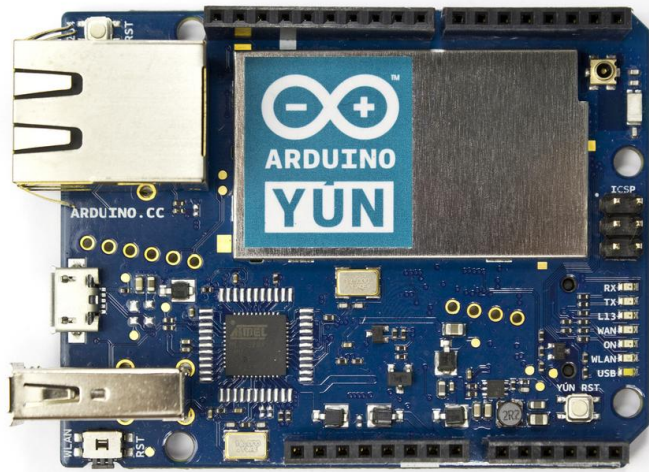


Figura 5: Vista frontal de la placa Arduino Yún.

3.2.2 LEDs

Aunque se ha mencionado anteriormente que con este tipo de diseño se consigue un número elevado de órdenes para controlar el BCI, se va a limitar este estudio a cuatro LEDs oscilando a diferentes frecuencias, que estarán asociados a las instrucciones "arriba", "abajo", "derecha" e "izquierda". Esta decisión se debe a que con un número más elevado de parámetros a tener en cuenta el análisis de resultados resultaría más complejo y susceptible a un mayor número de errores. Se comprobará si el sistema funciona para diferentes frecuencias configuradas en un mismo LED, lo cual indicará si en un futuro se puede implementar este sistema con más LEDs, resultando en un número más elevado de instrucciones para el BCI.

Para este diseño se van a utilizar un total de nueve LEDs. Cuatro de ellos serán LEDs de estimulación oscilando cada uno a una frecuencia distinta, de colores verde, blanco, rojo y amarillo. Otros cuatro LEDs más pequeños, de color verde, se colocarán 2mm por debajo de ellos, y serán indicadores de posición para guiar al sujeto hacia el LED que ha de mirar durante los experimentos de validación. Durante el uso de una aplicación concreta se utilizarán para que cuando un sujeto fije su atención en una instrucción (por ejemplo, el LED de "arriba") y el BCI reciba e interprete correctamente esa orden, ilumine el LED verde confirmando que se ha detectado la instrucción.

Por último, se encenderá un LED rojo cuando el programa se esté inicializando y al cambiar la configuración de un LED, que indicará que el sistema está haciendo operaciones y aún no es el momento de fijar la atención visual.

Para los primeros LEDs, se ha utilizado un modelo de alto brillo adquirido en la tienda de electrónica Merchán con un recubrimiento cilíndrico de 5mm de diámetro y 5 mm de altura. Tienen una intensidad de brillo máxima de 700 mcd (excepto el LED rojo que tiene un mcd mayor, factor que se soluciona regulando el brillo a través de su PWM) y un voltaje y corriente de funcionamiento de 3.2V y 20mA, respectivamente. Su ángulo de exposición es de 100°, factor que permitirá una mayor flexibilidad a la hora de establecer el ángulo con el que el sujeto mira la luz. Se tratan de componentes de inserción, con dos patas que irán soldadas a pines PWM de la placa Arduino para que sea posible regular la intensidad del brillo.

Para los LEDs restantes, tanto los verdes como el rojo utilizarán el modelo TLHR4400 (rojo) y TLHR4401 (verde), cuya referencia es 708-2753 en www.rs-online.com. Se trata de LEDs de 3 mm de diámetro y una intensidad luminosa de 3mcd, bastante más baja que la de los LEDs del primer tipo. Esto se debe a que en este caso se trata de LEDs indicativos y no deben destacar con respecto a los que se encargan de reproducir la frecuencia de estimulación para evocar SSVEPs. Su ángulo de exposición es de 60°, y funcionan a 3V y 30mA.

Los LEDs se colocarán en los vértices de un cuadrado con una distancia en diagonal de 9 cm. Se ha elegido este valor después de numerosas pruebas (ver capítulo 4) por ser el que proporciona un equilibrio distancia-eficiencia.

3.2.3 Power Bank

Por último, es necesario hablar de la forma de alimentación de la placa Arduino. Se podría haber utilizado una pila, pero este método es más laborioso a la hora de recargar el dispositivo. Dado que el Arduino tiene un voltaje de entrada bastante común y estable, se ha encontrado un Power Bank que proporciona un voltaje de salida de 5V, el indicado para alimentar el Arduino, y tiene una capacidad de almacenamiento de 2200mAh, lo cual ofrece una autonomía de 6 horas, una vez conectado al Arduino. También permite su uso al mismo tiempo que se está cargando, siendo el tiempo de carga de alrededor de 6:30 horas.

Se trata de un modelo de 94mm x 23mm x 23mm, pesando 0.076 Kg con un puerto de entrada microUSB y un puerto de salida USB. El puerto USB irá conectado al puerto microUSB de la placa Arduino Yun como alimentador del circuito, y a su vez el puerto microUSB irá conectado a un puerto USB hembra por cuestiones de diseño. El Power Bank consta también de un botón de encendido/apagado, que resultará muy útil ya que controla el encendido y apagado completo del circuito.

3.2.4 Placa Matriz

Para fijar los LEDs a la caja y conectarlos al Arduino, es necesario un soporte que proporcione una gran estabilidad y protección tanto para los LEDs como para el resto de elementos de la caja. Para ello, se ha utilizado una placa Matriz de 12.5 cm x 11.5 cm, con agujeros separados entre sí 2.54 mm y de un diámetro de 1 mm, que se introducirá en la caja sobre los muros de protección del Power Bank y el Arduino. Es sobre esta placa donde se soldarán los LEDs, y mediante cables se realizarán las conexiones a los pines digitales de Arduino.



Figura 6: Vista lateral de la placa Arduino Yún.

Los LEDs en los que se configuran las frecuencias se conectarán a los pines PWM número 5, 9, 3 y 11, en las posiciones arriba, abajo, derecha e izquierda respectivamente. Por su parte, los pines correspondientes a los LEDs de posición serán los número 4, 10, 2 y 12. Finalmente, el LED rojo indicador de funcionamiento se conectará al pin 13.

Esta placa se conectará al Arduino mediante unos pines PCB macho-macho. Como puede verse en la figura 6, entre el pin 7 y el 8 la distancia, en lugar de ser de 2.54 mm, es de 4.06 mm, pero esto no causa problemas a la hora de ser insertado en la placa matriz.

El aspecto de la placa matriz una vez soldados todos los componentes es el observado en la figura 7. El rectángulo azul indica la zona conectada al Arduino y a los pines.

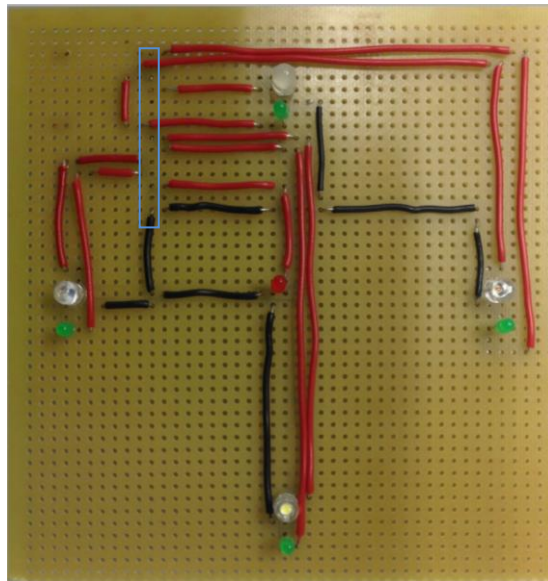


Figura 7: Vista frontal de la Placa Matriz donde se colocaran los LEDs para conectarlos al Arduino Yún.

3.2.5 Diseño de la caja

Una vez elegidos los componentes, se procede al diseño de la caja que los contendrá, de manera que sea lo más compacta posible para que el dispositivo final sea pequeño y fácil de transportar, de manera que un usuario que vaya a someterse una sesión larga pueda sostenerlo o acomodarlo sin esfuerzo. En la figura 8 se ve el diseño de esta caja a través de la herramienta CST [42], que posteriormente se imprimirá en la impresora 3D del laboratorio del GNB en plástico PLA verde, un material biodegradable que puede inyectarse y termoformarse.

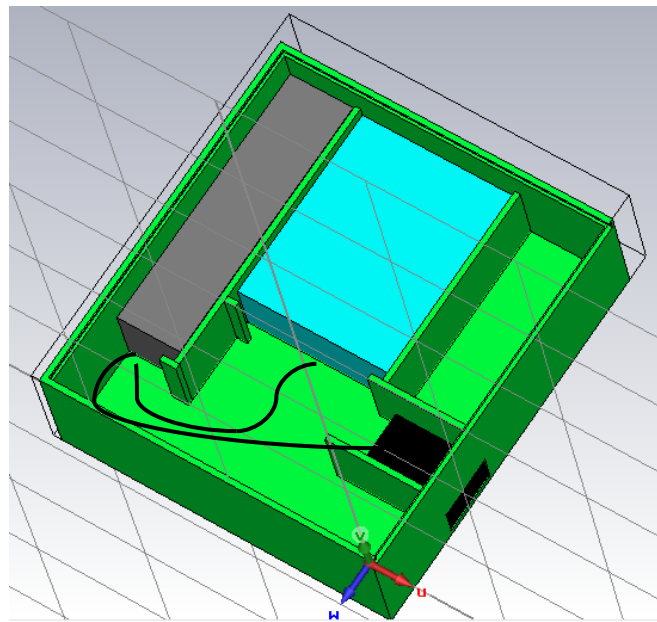


Figura 8: Diseño de la caja en CST para su impresión 3D.

La caja tiene unas dimensiones exteriores de 13 x 12 cm, que vienen limitadas por los componentes. El Power Bank de 94 cm tiene conectado un cable cuyo cabezal le añade 3 cm, y se han dejado 0.6 cm de margen para poder girar el cable, lo que hacen los 13 cm de ancho. Por su parte, la distancia entre diámetros de los LEDs es de 9cm, dejando un margen de 1.5 y 2 cm a cada lado.

Se han añadido paredes internas para una correcta fijación de los elementos a la caja, evitando que se muevan al ser transportados. El prisma gris representa el Power Bank, el azul a la placa Arduino y el negro el conector USB hembra para recargar el Power Bank. Las paredes tienen un grosor de 0.2 cm y una altura de 0.25 cm, siendo superior a la altura de cualquiera de los elementos ya que estas paredes servirán como soporte para sujetar la placa matriz sobre las que se colocarán los LEDs.

Además, en la parte superior de la caja se encuentra un raíl que recorre tres de las cuatro caras externas para que se deslice la tapa superior blanca que actuará

como disipador de los LEDs. El raíl tiene una altura de 2mm, y una profundidad de 1mm.

En la figura 9 se encuentra la caja sin elementos. Se han hecho tres agujeros en las paredes exteriores con distintos fines. El agujero inferior servirá para poder conectar un cable que cargue el Power Bank, la ranura la de pared derecha sirve para introducir o extraer la tarjeta microSD y poder examinar los resultados o añadir nuevos datos sin tener que abrir toda la caja, y el último agujero será un botón que se utilizará para encender y apagar todo el circuito.

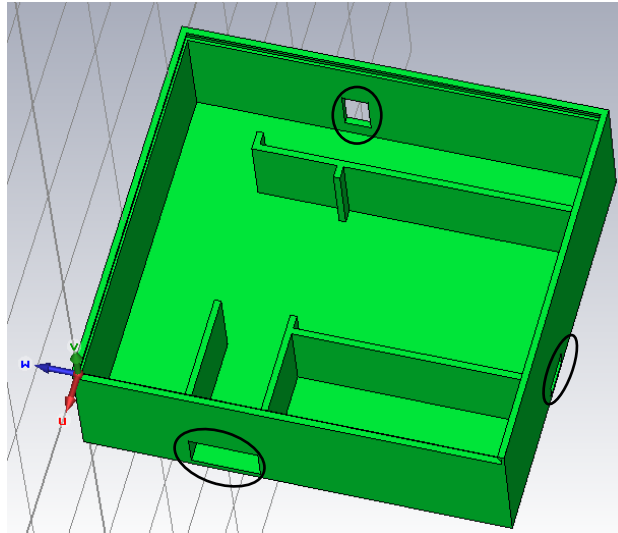


Figura 9: Diseño de la caja en CST sin elementos.

Sobre esta caja, a través del sistema de raíles, se colocará una tapa de plástico de color blanco de un grosor de 1.5mm que servirá como disipador para controlar que el brillo de los LEDs no resulte incómodo, además de servir como protección al circuito.

3.2.6 Características del diseño final.

Una vez montado el circuito e integrado en la caja, el dispositivo tiene las siguientes prestaciones:

Tamaño	13 cm x 12 cm x 4 cm
Peso	310.82 g
Voltaje de entrada	5 V
Autonomía	6 horas
Tiempo de carga	6:30 h
Coste Arduino	52 €
Coste Power Bank	17.00 €
Coste Placa Matriz	6.31 €
Coste LEDs	1.48 €
Coste Total	78.11 €

Tabla 5: Características finales de diseño del dispositivo estimulador para BCIs basados en SSVEPs.

En la figura 10 se puede observar el aspecto final de la caja, en tres niveles. A la izquierda se muestra el interior de la caja, con el Power Bank y la placa Arduino. En la imagen central, se incorpora la placa matriz con los LEDs. Por último, la imagen de la izquierda muestra la placa en funcionamiento, con los 4 LEDs iluminándose a distintas frecuencias.

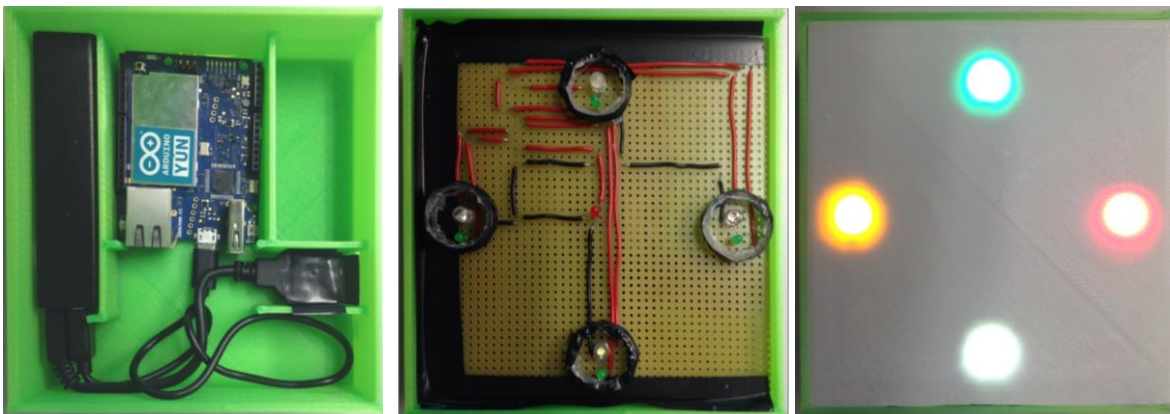


Figura 10 a, b y c: Aspecto frontal de la caja en tres niveles.



Figura 11: Vista lateral de la caja: Ranura para cargar, botón de Encendido y slot para la tarjeta microSD

3.2.7 Casco de registro de EEG Emotiv EPOC.

El sistema de registro de EEG que se va a utilizar en este BCI es el diseñado por EMOTIV, un dispositivo multicanal comercial de relativo bajo coste (unos 700 euros con la SDK de investigación). Ofrece una alta resolución, y se conecta de forma inalámbrica a un ordenador que puede funcionar tanto en Windows como MAC o Linux.

Se trata de un sistema unipolar que utiliza un set de 14 sensores y dos referencias, distribuidos según el sistema Internacional 10-20, estandarizado por la AES (*American Electroencephalographic Society*). Este sistema estipula que a partir de dos puntos de referencia los electrodos se dispongan en planos que se abran a intervalos de 10% y 20% desde el plano que une las dos referencias. Esta distribución se puede observar en la figura 12.

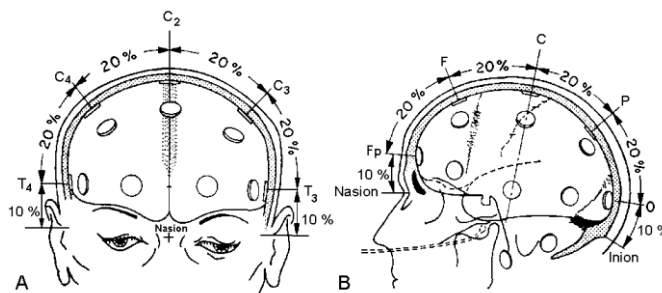


Figura 12: Disposición de los electrodos según el sistema Internacional 10-20.

Los electrodos de EMOTIV EPOC se colocan en las posiciones AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4, donde F significa Frontal, A indica el lóbulo de la oreja, C la región central, P el parietal, T el temporal y O el occipital.

Cada uno de estos electrodos es un canal que registra la actividad cerebral en ese punto y la muestra por pantalla permitiendo guardar los datos que registra, que serán los analizados posteriormente para extraer las características específicas que traducen la actividad cerebral en intenciones que emite el sujeto. En la figura 13 se observa un ejemplo de uso de la aplicación TestBench de Emotiv que muestra el status de los electrodos, indicándose con una luz verde si la calidad del contacto del electrodo es óptima y con una luz amarilla, naranja, rojo o negro si no lo es con esa gradación, así como las diferentes señales que registra cada electrodo. Esta herramienta se utiliza para registrar el EEG durante las pruebas de validación del dispositivo diseñado.

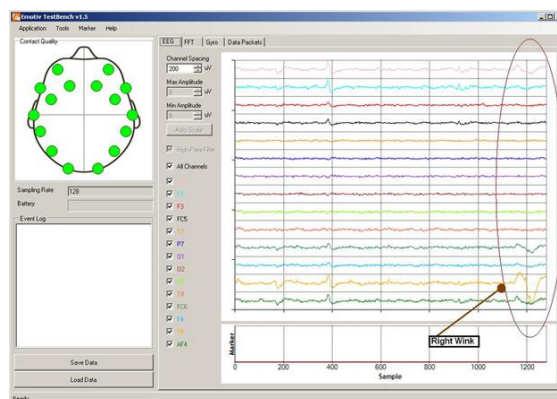


Figura 13: Captura de pantalla del TestBench de Emotiv.

3.3 Software

Aparte de la arquitectura del dispositivo, el Arduino se maneja a través de un programa que la web de Arduino facilita. Se utilizará la versión Arduino 1.5.6-r2 BETA, que tiene soporte para la placa Arduino Yún.

Todo sketch implementado en Arduino deberá constar de una función inicial de `setup`, que se ejecutará una sola vez e inicializará el proceso, y otra función que se ejecutará en bucle indefinidamente.

Esta sección está dividida en varias partes en función de las diferentes incorporaciones que se han hecho al código. Se comenzará con la implementación del software que hace que los cuatro LEDs se iluminen cada uno a una frecuencia determinada. Posteriormente, se añadirá el sistema Cliente-Servidor que permite la comunicación vía Wi-Fi entre dispositivo y ordenador. Después se introducirá la utilización de la tarjeta microSD, y para finalizar se configurará el control de los LEDs verdes de indicación.

3.3.1 Configuración de los LEDs

Como punto de partida, se crea un código inicial cuya función es encender y apagar un LED en función de un periodo de tiempo que se declara como constante. Utilizando los ejemplos de código que proporciona Arduino para entender el lenguaje de programación [5], se consigue iluminar un LED a la frecuencia deseada mediante el uso de la función `delay()`, introduciendo como parámetro el periodo en milisegundos y cambiando el estado del LED de forma digital (ver Anexo I A). Como se ha mencionado anteriormente, el código que se implemente funcionará en bucle indefinidamente, por lo que con un solo `delay` se consigue la frecuencia deseada.

Sin embargo, este método deja de ser útil en el momento en el que se introduce otro LED. Para que dos LEDs oscilen simultáneamente a dos frecuencias distintas harían falta dos `delay()`, cada uno esperando al periodo que le corresponde, pero el tiempo que el programa consume para el primer retardo afecta a todo el código, y lo mismo pasa con el retardo del segundo LED, haciendo que se desconfiguren las frecuencias y no oscilen según el periodo que se había establecido.

La solución óptima a este problema sería la introducción de hilos en el programa, utilizando un hilo para cada LED, pero esta opción no es viable debido a que Arduino no tiene soporte nativo de multithreading.

La opción que se ha escogido es la de implementar un código que no utilice retardos que paralicen el programa (ver Anexo I B). En su lugar, se hará uso de la función `micros()` para medir el tiempo transcurrido desde la última vez que el LED cambió de estado. Para conseguir esto, se almacena en una variable el tiempo (en microsegundos) en el que el LED pasa de HIGH a LOW o viceversa. A partir de ese momento, y para cada pasada del bucle, se obtiene el tiempo en microsegundos y se le resta el tiempo en el que el LED cambió de estado.

Ese valor se compara con la mitad del periodo (porque dentro de un periodo la mitad del tiempo está encendido y la mitad apagado), y en el momento en el que dicho valor iguala o supera al periodo se vuelve a cambiar de estado y se repite el proceso.

Con este método, se puede expandir el código y repetirse para los cuatro LEDs, trabajando simultáneamente y siendo el tiempo que tarda en ejecutar cada línea de código despreciable frente al intervalo que ha sido fijado.

El volcado del programa en la placa se puede hacer tanto por cable como vía Wi-Fi. Configurando previamente el Arduino, un ordenador con Wi-Fi mostrará en su lista de redes la opción "Arduino-90A2DAF01C6F", y con conectarse a ella y elegir la opción de subida por Wi-Fi bastará para una correcta configuración de Arduino sin necesidad de sacarlo del dispositivo de estimulación SSVEP para volcar el programa.

Además de todo esto, se ha añadido una variable "fade" que controla el brillo de los LEDs, de forma que se puede configurar para una intensidad mayor o menor. Los LEDs elegidos para codificar los comandos "arriba", "abajo", "izquierda" y "derecha" tienen una intensidad de brillo diferente para cada uno debido al color, siendo el brillo máximo una luz que puede causar molestias en el usuario, y cuando están a la vez los cuatro LEDs encendidos pueden interferir unos con otros, aumentando aún más la molestia. En el capítulo 4 se realizarán pruebas con varios sujetos para observar a qué intensidad de brillo se encuentran más cómodos y se obtienen los mejores resultados.

Para controlar este brillo se aprovecha el hecho de que los pines de salida de los LEDs que oscilan son PWM. A la hora de escribir de forma digital el valor de encendido (`digitalWrite(led1Pin, HIGH)`), se escribirá de forma analógica, dándole un valor entre 0, apagado, y 255, máxima potencia (`analogWrite(ledPin, led1State)`), donde led1State tendrá valores LOW o fade). La variable fade que antes se mencionó tendrá entonces un valor entre 0 y 255, que se configurará manualmente dentro del código. De esta forma, quedan configuradas tanto las frecuencias como el brillo de los LEDs de estimulación.

En el capítulo 4 se comprobará si las frecuencias fijas del estímulo lumínico son las esperadas realizando mediciones con un osciloscopio.

3.3.2 Comunicación con la interfaz cerebro-máquina

El objetivo de este trabajo es el desarrollo de un dispositivo de estimulación SSVEP que sea capaz de comunicarse con la interfaz cerebro-máquina. Esto es necesario ya que el BCI es capaz de calcular las mejores frecuencias para cada sujeto, y una vez obtenidas estas frecuencias hay que enviárselas al dispositivo SSVEP, que las configurará en los LEDs. También habrá que saber qué frecuencia corresponde a cada instrucción, y mantener una comunicación que permita al BCI comunicar al dispositivo cualquier cambio en las frecuencias.

Bajo estas circunstancias, ya no es una opción viable declarar cuatro frecuencias al comienzo del código y volcar el programa a la placa, ya que esto significaría modificar líneas de código para cada sujeto y para cada circunstancia, añadiendo un tiempo adicional de subida y conocimientos de programación de los que el sujeto puede carecer.

Las placas Arduino tienen una forma de comunicarse con el usuario a la vez que siguen ejecutando el programa, y es a través del puerto Serie. Una vez subido el programa a la placa, y desde la interfaz de Arduino, es posible acceder al puerto Serie que permitirá al usuario introducir cambios y órdenes a la placa. Estas órdenes serán procesadas por el Arduino y ejecutadas, y comunicará los cambios al usuario también a través del puerto Serie.

La opción del puerto Serie tiene la desventaja de que de nuevo el usuario tiene que entrar en el programa de Arduino, subir el programa a la placa y una vez subido abrir dicho puerto. Cada vez que se quisiera utilizar el dispositivo habría que volcar el programa aunque fuera una sola vez, ya que el puerto Serie va asociado a un código de programa y sin él no es posible abrirlo.

Pero, como se ha explicado anteriormente, Arduino Yún tiene la opción de una comunicación vía Wi-Fi. Es esta característica la que se va a utilizar para realizar la comunicación con la placa.

El objetivo de esta implementación será poder enviar instrucciones a la placa Arduino mediante comandos a través de cualquier navegador de internet y cualquier sistema operativo, ya que serán peticiones HTTP. Mientras que no todo sujeto tiene por qué tener conocimientos sobre programación, el uso de navegadores de internet y de URLs está ampliamente extendido y a día de hoy resulta intuitivo y cómodo de manejar. Una vez esté implementado en la placa, bastará con conectarse a la red Wi-Fi de Arduino, abrir un navegador, bien desde el ordenador o bien desde una tablet o un Smartphone, e introducir una URL como la siguiente:

<http://arduinoTFG.local/arduino/arriba/11>

Para todos los casos, "arduinoTFG" será el nombre de la placa, y ".local/arduino" es una condición que viene dada por la configuración del sistema cliente-servidor de Arduino.

El comando "arriba" indica que se va a configurar el LED de arriba, y "11" son 11Hz. Cuando se pulsa la tecla a enviar, se mostrará en el navegador el mensaje *"El LED de arriba se ha configurado a una frecuencia de 11.00Hz."* También es posible enviar este String mediante la terminal de comandos en Linux mediante el comando "curl http://arduinoTFG.local/arduino/arriba/11".

En cualquier momento se puede modificar la frecuencia de cualquier LED de forma casi instantánea repitiendo estos mismos pasos, y sería posible configurar el BCI para que construyera un string con esta estructura y lo mandara a través del puerto 80, obteniendo los mismos resultados.

Ahora se explicará cómo se ha realizado la configuración en el Arduino para conseguir el objetivo que se acaba de explicar. El código explícito puede verse en el Anexo II.

Como se describió en las características del Arduino Yún, la conexión Wi-Fi se realiza desde el procesador Linino, al que se accede mediante la librería `Bridge`. Además de esta librería, se añadirán las librerías `YunServer` y `YunClient`, para crear una comunicación cliente-servidor mediante peticiones HTTP. En el setup se inicializa el Bridge, y se configura un servidor "server" de tipo `YunServer` para que escuche en el puerto 80.

Una vez dentro del bucle, para cada pasada se comprobará si se ha recibido un cliente "client" de tipo `YunClient` a través del servidor. Si es así, se procesará el cliente, y si no se seguirá ejecutando el resto del bucle.

En el caso de que se reciba una petición de cliente, "client" será una cadena de caracteres que, para el ejemplo que se ha descrito anteriormente, sería "arriba/11" (y no http://arduinoTFG.local/arduino/arriba/11 como se podría pensar). Recibido esto, bastará con identificar el comando "arriba" y la frecuencia de 11Hz, y modificar el periodo del primer LED. También se informará al usuario por pantalla de que se han realizado los cambios. El resto del programa se ejecutará como se explicó en el apartado anterior, con la diferencia de que ahora el periodo del primer LED ha sido modificado.

Esta técnica tiene un inconveniente importante con respecto al funcionamiento del programa. Comprobar si se ha recibido o no un cliente consume un tiempo variable, que es distinto para cada pasada del bucle. Además, todos los cambios a realizar para que el LED cambie su frecuencia también generan retardo. Este tiempo ya no es despreciable a la hora de calcular el periodo de cada LED y cambiar de estado. Cuando se comprueba con el osciloscopio si las frecuencias son correctas (Capítulo 4), se observa que las frecuencias se han desconfigurado, oscilan alrededor de la frecuencia seleccionada pero en ningún momento se mantienen estables.

Para solucionar este problema, de nuevo hay que descartar la opción de utilizar hilos porque Arduino no lo permite de forma nativa. Trasladar la recepción a la función "setup" tampoco solucionaría el problema porque al salir del setup y comenzar el bucle indefinido ya no podría volver a recibir clientes y no sería posible cambiar el valor de las frecuencias.

Así pues, sabiendo que la escucha en el puerto tiene que realizarse dentro del bucle, pero que ese tiempo que consume desequilibra las frecuencias, una opción posible para solucionar este problema es revisar si ha llegado un cliente cada cierto tiempo, y durante el resto del tiempo dejar que los LEDs sigan funcionando normalmente. Seleccionando un tiempo lo suficientemente grande como para que dicho desequilibrio resulte despreciable, pero lo suficientemente pequeño como para no perder clientes ni estar demasiado tiempo a la espera del cambio, este problema queda solucionado.

Tras realizar numerosas pruebas, se encuentra un equilibrio de ambos factores en aproximadamente 5 segundos, que se traducen en 62500 pasadas de bucle. Un contador permite acceder a la comprobación de recepción de nuevos clientes pasado este número determinado. Si el BCI o un usuario quieren cambiar el valor de una frecuencia, tardará como máximo 5 segundos en realizarse dicha petición, que es una espera razonable. Cada 62500 pasadas de bucle, habrá 1 cuyo periodo se desvíe ligeramente, lo cual no afecta a la percepción humana. De todas maneras, se implementa un LED rojo adicional que se ilumina cuando al llenarse el contador se encuentra un cliente, e indica que mientras se realizan los cambios oportunos no se obtendrán estimulaciones fiables. Al apagarse dicho LED, las frecuencias vuelven a ser estables y se puede continuar con la ejecución.

Para agilizar este proceso durante la configuración inicial, se crea un bucle condicional dentro de la función "setup" que recibe de forma instantánea los valores de cada frecuencia sin necesidad del contador. La condición del bucle es que todos los LEDs tengan una frecuencia configurada. Cuando se acaba de configurar las cuatro, comienzan a funcionar y es entonces cuando se salta a la función "loop" y se pone en marcha el contador.

3.3.3 Almacenamiento de datos

Una vez implementada la comunicación a través del Bridge, el dispositivo ya cuenta con su funcionalidad básica. Es capaz de recibir en cualquier momento información desde el BCI y traducirla en instrucciones, e ilumina cada LED a la frecuencia correcta. La implementación de un sistema de almacenamiento de datos a través de la tarjeta microSD no es estrictamente necesaria, pero va a utilizarse por comodidad y para agilizar el proceso de configuración de los LEDs.

A diferencia del resto de placas Arduino, la tarjeta microSD es controlada desde el procesador Linino, y por tanto vuelve a ser necesario invocar al Bridge con su librería correspondiente, para acceder a los recursos de esta parte de la placa. También es necesario añadir la librería "FileIO", que es donde se encuentran las funciones que manejan la tarjeta.

Para realizar las configuraciones necesarias y poder acceder a la tarjeta microSD, en la función setup habrá no solo que inicializar el puente, sino también inicializar la función FileSystem (`FileSystem.begin()`), para comenzar a utilizar las funciones que acceden a la tarjeta microSD.

Si se quiere utilizar una microSD en una placa de Arduino, es necesario formatearla previamente en un formato FAT o FAT32, y añadir un directorio con nombre "arduino", siendo el directorio que contenga esta carpeta donde se almacenará el contenido que resulte de interés.

Como cabe esperar, la funcionalidad de una tarjeta microSD es o bien leer de la tarjeta o bien escribir en ella. Leer de la tarjeta servirá para implementar la opción de "cargar", que leerá de un fichero las frecuencias correspondientes a un usuario y las volcará en la placa. De la misma forma, escribir en la tarjeta ofrecerá la opción "guardar", con la que se almacenan en la tarjeta las frecuencias de oscilación que se estén utilizando en ese momento.

I. Cargar.

Cargar un set de frecuencias directamente a la placa soluciona el inconveniente de tener que introducir de una en una las frecuencias para cada usuario. Por ejemplo, el set de frecuencias base con las que se empieza para observar la respuesta del sujeto son de 6, 7, 8 y 9Hz. Sin la opción de cargar, habría que introducir esta serie de URLs:

<http://arduinoTFG.local/arduino/arriba/9>
<http://arduinoTFG.local/arduino/abajo/10>
<http://arduinoTFG.local/arduino/derecha/11>
<http://arduinoTFG.local/arduino/izquierda/12>

Sin embargo, implementada la funcionalidad de cargar, bastaría con incluir en la tarjeta microSD un fichero "*FreqBase.txt*" que almacenase cuatro valores numéricos, de manera que el primero se corresponda con la instrucción "arriba", el segundo "abajo", el tercero corresponde a "derecha" y el último a "izquierda". De esta forma, la URL a introducir se reduciría a:

<http://arduinoTFG.local/arduino/cargar/FreqBase>

Otro ejemplo en el que este comando resultaría útil es el caso en que dos personas utilizan con asiduidad el dispositivo SSVEP para un BCI. Sabiendo cuáles son sus mejores frecuencias, se almacenaría un fichero "*Sujeto1.txt*" con las frecuencias de ese usuario, y así para cada sujeto. Cuando un usuario fuera a utilizar la interfaz, bastaría con cargar sus frecuencias en lugar de volver a calcularlas y cargarlas una a una.

Para cargar un set de frecuencias de la tarjeta microSD a la placa, el comando a introducir en el navegador habrá de tener el siguiente formato:

<http://arduinoTFG.local/arduino/cargar/Sujeto1>

Como se ha visto en el apartado anterior, el cliente recibido será un string cuyo contenido será "cargar/Sujeto4". En caso de que exista un fichero .txt en la tarjeta (en el directorio /mnt/sd, que FileSystem asocia al directorio /arduino), la orden se procesará, y en caso contrario se informará por pantalla de que no existe tal usuario.

Para procesar la orden, se abre el fichero y se extraen los valores numéricos que éste contiene. Una vez hecho esto, basta con asociar cada valor al periodo de oscilación de cada LED, cerrar el fichero y continuar con el programa con normalidad. En el Anexo III A se muestra el código fuente de esta operación.

II. Guardar.

La opción de guardar los datos es la complementaria de cargarlos. Se utiliza para guardar las frecuencias que en ese momento estén configuradas para los LEDs de estimulación, y evitar tener que configurarlas de nuevo en un futuro.

Una vez se han detectado las mejores frecuencias de un usuario y estén en ese momento ejecutándose en los LEDs, basta con introducir el comando:

<http://arduinoTFG.local/arduino/guardar/Sujeto1>

Si se escribe esto durante el periodo de setup e inicialización de los LEDs, el programa informará de que no se puede guardar aún ya que no hay frecuencias cargadas en los LEDs. En caso de estar dentro del bucle, guardará en un fichero "*Sujeto1.txt*" los valores de las frecuencias. Si ese fichero ya existe, se informará por pantalla y se sobrescribirá, ya que las mejores frecuencias pueden cambiar con el tiempo y se almacenará el cálculo más reciente. En el Anexo III B se encuentra el código de esta implementación.

3.3.4 LEDs verdes de indicación

Por último, se han añadido cuatro LEDs verdes de indicación situados 2mm por debajo de cada LED de frecuencia y con un brillo mucho menor que el de los LEDs blancos principales para no alterar los resultados. Existen dos propósitos diferenciados para esta implementación, el primero es mostrar al usuario en qué LED tiene que concentrarse, y el segundo es confirmar al usuario que se ha detectado su intención y se ha realizado la orden.

I. Mostrar el LED en el que fijar la atención.

Es una implementación sencilla en la que se hace uso de la tarjeta microSD para establecer una secuencia de LEDs a seguir. Se crea un fichero "*Secuencia.txt*" que contiene en cada línea una instrucción y un tiempo (por ejemplo, *ARRIBA 10* indica que se ilumine el LED de arriba durante 10 segundos). Se lee la instrucción y el tiempo, y se ilumina el LED verde correspondiente a la instrucción.

De la misma manera que se comprobaba en los LEDs de estimulación el periodo, se comprueba en los verdes de indicación si ha transcurrido el tiempo especificado, con la excepción de que se añade una constante "*start*", que se declara al final de la función *setup*, y almacena un valor entero correspondiente al tiempo en microsegundos transcurrido desde el comienzo del programa hasta que se comienza a leer de la tarjeta. Esta constante es necesaria porque a la hora de comprobar el tiempo que ha pasado, la función *micros* devuelve el tiempo desde que comenzó a ejecutarse el programa. En el encendido de los LEDs de frecuencia el intervalo de tiempo es tan pequeño que no influye que la primera oscilación del LED sea más corta debido al tiempo que ha consumido el *setup*, pero en este caso si la inicialización (creación del *Bridge* y declaración de unos valores de frecuencias iniciales en los LEDs) tarda, por ejemplo, un minuto, cuando se haga la primera comprobación al extraer del fichero de la tarjeta un tiempo de 10 segundos en el LED de arriba, ese tiempo ya habrá transcurrido y se saltará al siguiente, pasando por alto la primera línea del fichero. Sin embargo, si al valor de "*CurrentMicros*" se le resta el tiempo de *start*, este problema queda solucionado y no se pierde el valor de la primera línea.

Una vez acabado ese tiempo, se lee la siguiente línea en el fichero, se extraen los siguientes dos parámetros y se repite el proceso. Esto se implementa en el bucle principal concurrentemente con el código de los LEDs de estimulación, y se ha comprobado que este proceso no influye en la estabilidad de la frecuencia de dichos LEDs.

Un caso práctico en el que utilizar esto será en la detección de las frecuencias mejores de un usuario. Se sabe que en un dispositivo SSVEP las frecuencias se encuentran típicamente en un rango entre 6 y 40 Hz, aunque por motivos prácticos en el análisis de resultados se utilizará el rango de 6Hz a 15Hz para realizar las medidas.

Así pues, se ha creado un programa que inicializa las frecuencias de los cuatro LEDs de estimulación a 6, 7, 8 y 9Hz. Cuando el usuario comienza la secuencia, se activa el LED verde que le indica que ha de mirar arriba, donde la frecuencia es de 6 Hz. Pasados 20 segundos, cambia el LED de posición verde abajo, a 7Hz. Mientras el usuario mira al LED de abajo, el de arriba cambia su frecuencia a 10Hz. Pasados otros 20 segundos, cambia la posición a la derecha, y el LED de abajo cambia de 7 a 11Hz. Para cuando el usuario vuelva a la posición de arriba, las frecuencias serán de 10, 11, 12 y 13Hz. En dos rondas y media, se tienen todas las frecuencias registradas y tras un análisis de datos se puede observar qué frecuencias son mejores para cada usuario, guardarlas en la tarjeta microSD para un futuro y comenzar la sesión BCI.

II. Indicador de confirmación.

Este método se utilizaría en tiempo real para indicar al usuario que se ha recibido la instrucción. Una vez establecidas las frecuencias que hacen que se detecte la instrucción para un sujeto concreto, comenzaría la sesión BCI.

El análisis del electroencefalograma tarda entre 2 y 3 segundos en detectar una instrucción. Con una ventana de ese tamaño, realizaría un análisis en frecuencia e interpretaría la instrucción correspondiente. Por ejemplo, un usuario cuyas frecuencias mejores fueran 6, 8, 10 y 12 Hz, se concentraría en el LED de arriba hasta que el EEG realizara el análisis en frecuencia y observara un pico diferenciado en 6Hz. Entonces, mandaría al dispositivo SSVEP la orden de que encendiera el LED verde de posición correspondiente a la instrucción "arriba", y esto indicaría al usuario que puede dejar de concentrarse ya que la instrucción se ha recibido. Esta orden se mandaría de nuevo a través del puerto en el que el Arduino escucha, por lo que habría que contar también con los 5 segundos adicionales que puede llegar a introducir el contador.

Siendo un programa relativamente sencillo, sus aplicaciones pueden resultar de gran utilidad. Para personas discapacitadas, que es el objetivo de esta investigación, se podrían convertir las instrucciones "arriba", "abajo", "derecha" e "izquierda" en "Sí", "No", "Tal vez" y "No lo sé". Si se comprueba que este dispositivo funciona correctamente, se podría ampliar el número de instrucciones hasta, por ejemplo, un teclado, permitiendo una comunicación completa con personas que de otra forma no tendrían un medio para comunicarse.

Esto no se ha implementado porque no entra dentro de los objetivos del TFG y actualmente el análisis de las señales de EEG desde el casco EMOTIV EPOC no tiene soporte en tiempo real, pero el dispositivo SSVEP estaría preparado para ser utilizado de esta manera.

3.3.5 Código final

Debido a que no se trata de un sistema en tiempo real, será necesario utilizar dos códigos distintos que se subirán a la placa Arduino vía Wi-Fi (evitando tener que conectar el dispositivo al ordenador cada vez que vaya a cambiarse el código).

El primer código no se ejecutará hasta que se reciba un cliente cualquiera a través del navegador. Cuando se reciba, se iluminará el LED rojo durante 10 segundos, para que dé tiempo a comenzar a registrar la señal de EEG cuando empiece la primera frecuencia. Pasados los 10 segundos, se apagará dicho LED y se realizará un barrido en frecuencias que irá desde los 6 hasta los 15 Hz, tomando una muestra de 22 segundos para cada frecuencia. Será un código que hará uso de la tarjeta microSD para configurar los LEDs verdes de posición e indicar al usuario sobre qué LED fijar su atención y durante cuánto tiempo (como se ha dicho, este tiempo será de 22 segundos para obtener una muestra fiable). Estos resultados se analizarán en Matlab para saber qué frecuencias son más adecuadas para un usuario concreto, y se guardarán en la tarjeta microSD las cuatro mejores frecuencias.

Una vez determinadas las cuatro mejores frecuencias, se subirá a la placa el segundo código, que recibirá a través del navegador las frecuencias que se quieren configurar, manualmente o a través de un fichero cargado en la SD. Se cargarán y se reproducirán en los LEDs, permitiendo al sujeto expresar sus intenciones fijando la vista en alguno de ellos. Se podrán realizar varias pruebas, pero hasta que no se analicen los datos grabados a través de Matlab no se podrá saber qué intención quería expresar el sujeto.

Estos dos códigos se podrían simplificar en uno solo si se tratase de un sistema en tiempo real, evitando tener que cargar cada vez un código distinto, pero al no tener forma de determinar las cuatro mejores frecuencias en el momento, ese único código resulta poco práctico.

3.3.6 *Procesado de la señal*

Una vez generado el programa, y una vez adquiridos los datos a través del casco de EEG, es necesario procesar esa información para validar el efecto de la estimulación. Para eso, se ha utilizado un código de Matlab proporcionado por el Grupo de Neurocomputación Biológica que permite primero transformar la información que se graba en formato .edf a un formato .txt legible, a través de la herramienta *EEGLAB*, y posteriormente realizar una transformada de Fourier para un análisis en frecuencia. A estos códigos se le han realizado las modificaciones necesarias para acondicionarlo a este programa.

Para el paso a .txt, se extrae la matriz EEG.data, que contiene la información de 14 canales, y cada canal almacena los datos obtenidos a una tasa de 128 muestras por segundo.

Se hace una distinción entre si la señal a procesar es un baseline de 60 segundos, una muestra simple de 20 segundos o el barrido en frecuencias, que al ser de 6 a 15 Hz, tomará 8 frecuencias de 22 segundos cada una. En este caso, habrá que separar dichos datos en 8 ficheros .txt de 20 segundos (se elimina 1 segundo al principio y 1 al final por ser el cambio de una frecuencia a otra), para posteriormente realizarles la transformación. Los ficheros resultantes tendrán que tener un número de muestras exactas (tiempo de cada caso multiplicado por el número de muestras por segundo), por lo que si el tiempo es mayor se eliminarán muestras del principio y del final. Si la duración es menor que el mínimo, será necesario repetir la prueba.

Una vez se han pasado todos los datos al formato correcto, se utilizan los datos del baseline para calcular el ruido, y los datos de la medición como señal, y en función de ello se calcula la SNR en dB que se representa para observar el comportamiento en frecuencia. Si la medición es correcta, deberá aparecer un pico de amplitud considerable a la frecuencia en la que el sujeto estaba fijando su atención.

Para ello se escogen ventanas de tiempo de 2 segundos, es decir, 256 muestras, con un desplazamiento entre ventanas de 20 muestras entre ventana y ventana. Con estos valores, el solapamiento entre ventanas es muy elevado, pero se hace así para poder ver con claridad el estímulo SSVEP. Se utiliza una ventana de Hanning porque en el análisis se consideran lóbulos secundarios. A esta ventana se le aplica una FFT, al igual que al baseline, y se transforman a dB. Una vez están ambos en dB, se restan las señales obteniendo finalmente la SNR.

4. Experimentos y Resultados

En este capítulo se realizarán los experimentos de validación del dispositivo de estimulación SSVEP diseñado y se interpretarán los resultados de esta implementación, realizando las modificaciones consecuentes a los fallos que se puedan descubrir.

Se comenzará comprobando con un osciloscopio que las frecuencias que se establecen en el código del programa se corresponden con la precisión temporal de la oscilación de los LEDs, en particular que el control simultáneo de todos ellos no produce retardos. Posteriormente se observarán los resultados de la implementación de la comunicación vía Wi-Fi tanto sin contador como con él, y una vez establecido este aspecto se procederá a analizar los resultados de la estimulación que se obtienen a través del electroencefalograma con el casco de Emotiv EPOC para sacar conclusiones sobre cuáles son los mejores parámetros que hay que determinar sobre el dispositivo SSVEP para su adaptación a cada sujeto.

Por último, en el apartado 4.4 se verificará la utilización del dispositivo de estimulación como sistema de comunicación en personas discapacitadas, realizando pequeñas pruebas en tres pacientes del centro "Cottolengo" de Algete.

4.1 Validación de la precisión temporal del estímulo SSVEP.

La primera prueba a realizar antes de comprobar el efecto de la estimulación en el EEG a través del casco Emotiv es comprobar si cuando se seleccionan las frecuencias de los LEDs, estas se configuran correctamente y con la precisión temporal esperada. Para ello, se utiliza un osciloscopio que comprueba la salida en los pines del Arduino.

En el osciloscopio se puede observar que la frecuencia que se configura es exacta con un margen de error de hasta $\pm 0.01\text{Hz}$, y permanece constante durante todo el periodo de funcionamiento. En la segunda columna de la tabla 6 se puede observar la frecuencia que se consigue en los pines para diferentes frecuencias pasadas como parámetros. Posteriormente se podrá observar que ese error de precisión no afecta a la hora de medir los resultados en los sujetos, ya que al realizar la transformada de Fourier de la señal obtenida en los canales, aparecerá un pico en la frecuencia deseada (en el caso de que esa frecuencia sea detectable en el sujeto).

A continuación, en la figura 14, se muestra un ejemplo del resultado del osciloscopio para una frecuencia de 6Hz.

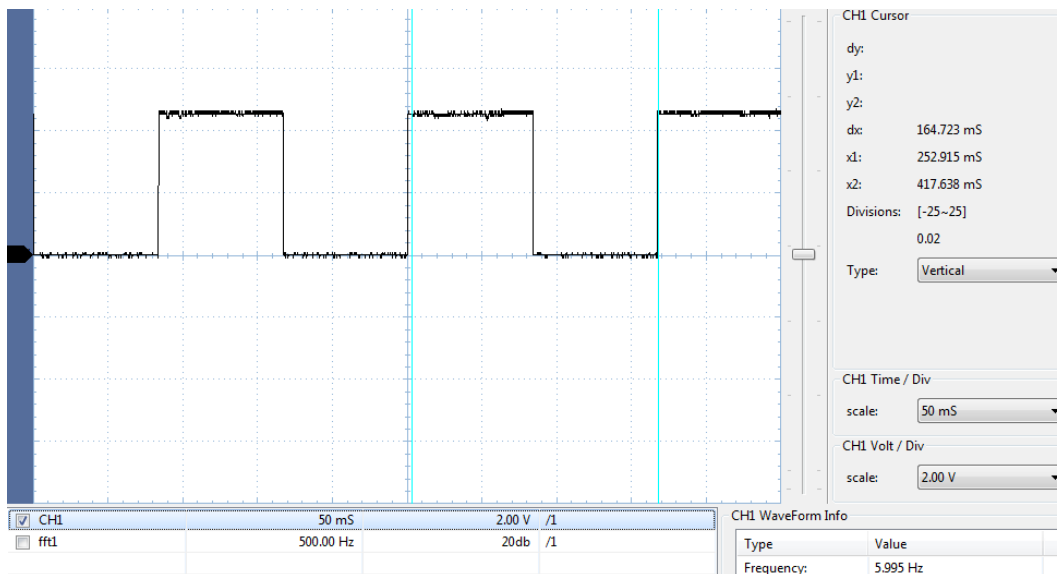


Figura 14: Captura del osciloscopio en la prueba de precisión temporal de una frecuencia de 6 Hz.

4.2 Implementación de la comunicación vía Wi-Fi.

Cuando se implementó la primera versión de este sistema cliente-servidor, a simple vista parecía que funcionaba correctamente, pero al realizarse la prueba con el osciloscopio se pudo comprobar que las frecuencias que se establecen en los pines son variables, y con estos resultados no se puede realizar un análisis fiable con el casco para validar la eficacia de la estimulación visual. En tercera columna de la tabla 14 se puede observar una media de los valores que se obtienen en el osciloscopio en un intervalo de tiempo de 10 segundos. No obstante, para cada instante de tiempo se obtienen valores de frecuencia muy dispares, llegando a desviarse varios hertzios del valor fijado en algunas frecuencias.

Sin embargo, al introducir el contador este problema queda solucionado, obteniéndose prácticamente los mismos valores que antes de incluir las modificaciones. Se obtienen los mismos valores de frecuencias para el código final con la implementación de la tarjeta microSD y los LEDs verdes de posición.

Frecuencias	Sin sistema Cliente- Servidor	Con sistema Cliente- Servidor (sin contador)	Con sistema Cliente- Servidor (con contador)
6 Hz	5.995 Hz	5.886 Hz	5.999 Hz
7 Hz	6.996 Hz	6.809 Hz	6.996 Hz
8 Hz	7.996 Hz	7.673 Hz	7.996 Hz
9 Hz	8.993 Hz	8.624 Hz	8.993 Hz
10 Hz	9.990 Hz	9.551 Hz	9.995 Hz
11 Hz	10.994 Hz	10.368 Hz	10.994 Hz
12 Hz	11.995 Hz	11.226 Hz	11.990 Hz
13Hz	12.997 Hz	12.346 Hz	12.993 Hz
14Hz	13.996 Hz	11.123 Hz	13.992 Hz
15Hz	15Hz	14.875 Hz	14.999 Hz

Tabla 6: Comparativa de diferentes códigos de programa medidos con el osciloscopio.

4.3 Experimentos de validación del estímulo con Emotiv.

Sabiendo que las frecuencias que se van a emitir son correctas, el siguiente paso es medir los resultados en diferentes sujetos para examinar si el dispositivo se puede utilizar para producir la estimulación exógena en un BCI basado en SSVEPs.

Antes de comenzar a describir las pruebas, es conveniente volver a las etapas de un BCI que se mencionaron en el estado del arte, e identificarlas en este trabajo. La primera etapa de adquisición de señal se realizará mediante el dispositivo de estimulación visual que se ha implementado, junto con el casco Emotiv que registra la actividad cerebral que evoca el estímulo. Para el preprocesado del EEG se utilizará Matlab junto con el toolbox EEGLAB [43], [44]. La extracción de características se realizará mediante una FFT, de nuevo a través de Matlab. La validación de que se produce el efecto deseado mediante la estimulación con el dispositivo se realizará observando en el análisis de FFT el valor del pico de mayor amplitud. La última etapa de FeedBack, que supondría un análisis en tiempo real, no se ha implementado en este proyecto.

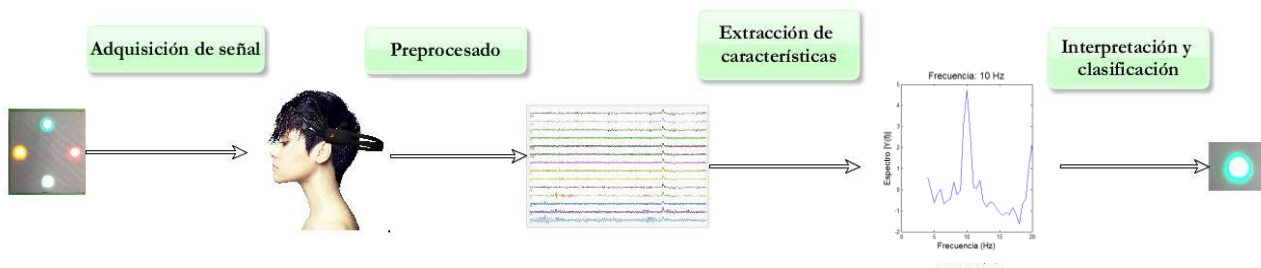


Figura 15: Etapas del sistema BCI con el sistema de estimulación visual desarrollado en este trabajo.

Se realizarán las pruebas en un total de 23 sujetos voluntarios, tanto hombres como mujeres y de distintas edades, con la única restricción de no tener problemas de ataques epilépticos, ya que la exposición a la luz podría provocarlos en personas que sufren de esta enfermedad. Se les pedirá que se concentren en una luz determinada situada a una distancia de aproximadamente 20 cm, primero habiendo únicamente una luz encendida y, posteriormente, con cuatro luces a oscilando simultáneamente a distintas frecuencias. Todas las pruebas se realizarán en el mismo lugar, con las mismas condiciones de luz y ruido de fondo, y todas con los mismos dispositivos.

Hay que tener en cuenta que los resultados variarán de un sujeto a otro, ya que la respuesta registrada en el EEG a la estimulación lumínica depende de cada individuo [39]. Además, al tratarse solamente de una muestra relativamente pequeña, los resultados obtenidos pueden no reflejar la variabilidad existente en la totalidad de la población.

Antes de realizar las pruebas habrá que grabar en cada sujeto su "baseline" de EEG, lo que se consigue mirando durante 60 segundos a una superficie lisa y blanca sin la estimulación de los LEDs. Esta señal determinará el ruido que produce el entorno y el propio cerebro, y se restará a la señal (en dBs) obtenida durante la estimulación para determinar la relación señal a ruido.

Las pruebas consistirán en dos fases. En la primera, se realizará el barrido en frecuencias que determina las mejores frecuencias para cada usuario.

En la segunda fase, se iluminarán los LEDs acordes a las mejores frecuencias con distintas variaciones para encontrar las características fijas en las que se obtienen mejores resultados, como pueden ser el brillo, la distancia entre LEDs, el color, etc. Habrá usuarios en los que no se obtengan resultados en la primera fase, por lo que carecerá de sentido continuar con la segunda.

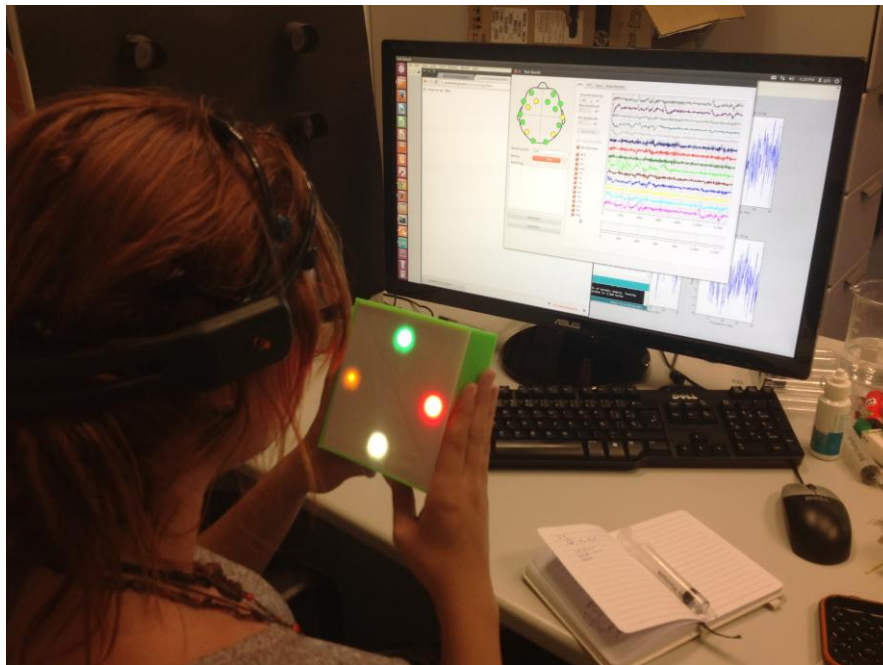


Figura 16: Primera etapa del BCI: Adquisición de la señal

4.3.1 Primera fase: Barrido en frecuencia.

En esta fase se determinará si el usuario es válido para la utilización de este dispositivo de estimulación SSVEP, ya que no todo el mundo es apto y se dan casos en los que no se registran picos de frecuencias diferenciados en ninguna frecuencia [8][16][39]. Para este caso, las frecuencias se mostrarán una a una, eliminando las interferencias que producen los cuatro LEDs encendidos a la vez para fijar correctamente los parámetros básicos, como por ejemplo el grosor del dissipador, y observar si el sujeto responde bien a una frecuencia.

La primera prueba que se realiza es la exposición a un LED de luz blanca oscilando a diferentes frecuencias en un rango comprendido entre 6 y 15 Hz. Inicialmente se hicieron pruebas en un rango de frecuencias mayor, entre 1 y 30 Hz, y se pudo observar que por debajo de 6 Hz no se obtiene ningún efecto en respuesta a la estimulación (esto es coherente, ya que por debajo de 6Hz no se trata de señales SSVEP, sino TVEP), y por encima de 15 Hz sí se obtienen respuestas hasta los 30 Hz, pero por ser demasiado largo el barrido en el análisis se realizan en este intervalo (6-15Hz). En caso de no encontrar frecuencias válidas, se podría ampliar el barrido a otras frecuencias.

Se toman muestras de una duración de 22 segundos, y el resultado que se muestra es una media de los valores obtenidos en los canales occipitales y frontales en ese tiempo, por ser las regiones del cerebro en las que se observa con claridad la frecuencia de estimulación.

En la figura 17 se muestra un ejemplo ilustrativo de los valores que se espera obtener en un usuario para esta medida, en un rango de frecuencias entre 6 y 13 Hz. Para este caso, se puede observar que la persona responde adecuadamente a la estimulación, y se generan picos diferenciados para todas las frecuencias, exactamente en la frecuencia configurada en el LED. Este sería un usuario válido con el que se podría proceder a la siguiente prueba.

Para esta, y para todas las figuras siguientes, se realiza el análisis sobre la señal $(O2+O1)-(F3+F4)$, donde O se refiere a occipital y F a Frontal en la nomenclatura de los electrodos, y O1, O2, F3 y F4 son las señales obtenidas de los electrodos colocados en dichas posiciones (ver figura 12). Se elige esta señal por ser la que mejores resultados produce, al ser la región occipital en la que se reproducen las frecuencias. En la figura 17 y sucesivas el eje x indica las frecuencias, mientras que el eje y muestra la amplitud de la señal en dB.

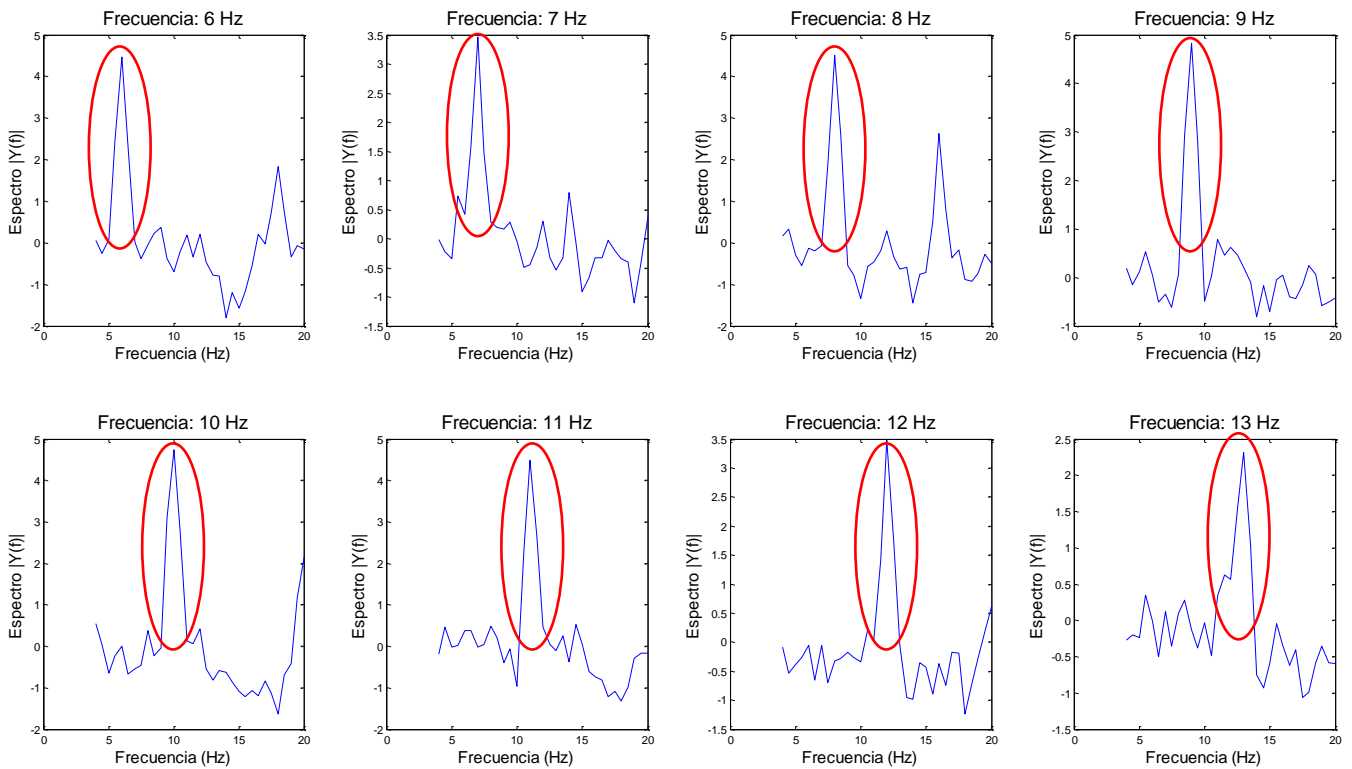


Figura 17: Resultados del análisis de EEG en respuesta a la estimulación con el dispositivo diseñado en el rango de 6 a 13 Hz.

Ampliando esta imagen para observar con detalle una de las gráficas, se observa en la figura 18 que el pico se produce en la frecuencia correcta, siendo su amplitud con respecto al siguiente máximo de 4 dB.

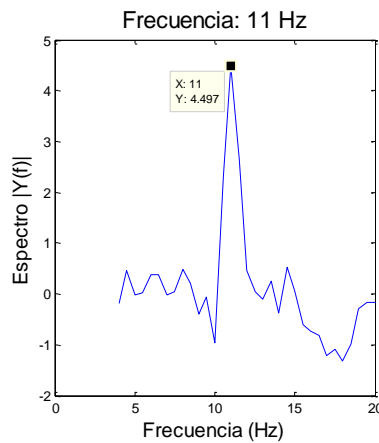


Figura 18: Detalle de la respuesta a 11 Hz en el ejemplo mostrado en la figura 17.

Sin embargo, en la figura 19 se muestra el caso contrario, en el que no se observa una respuesta lo suficientemente diferenciada en frecuencias y la amplitud de la señal es demasiado baja como para que el resultado sea determinante. Esto se debe a que no todo el mundo reacciona igual ante los

mismos estímulos, y en determinadas personas un dispositivo SSVEP puede no funcionar. Este fenómeno se conoce en inglés como “BCI Illiteracy” que se puede presentar con distintos grados [14]. Este tipo de sujetos se descartarán para las pruebas siguientes.

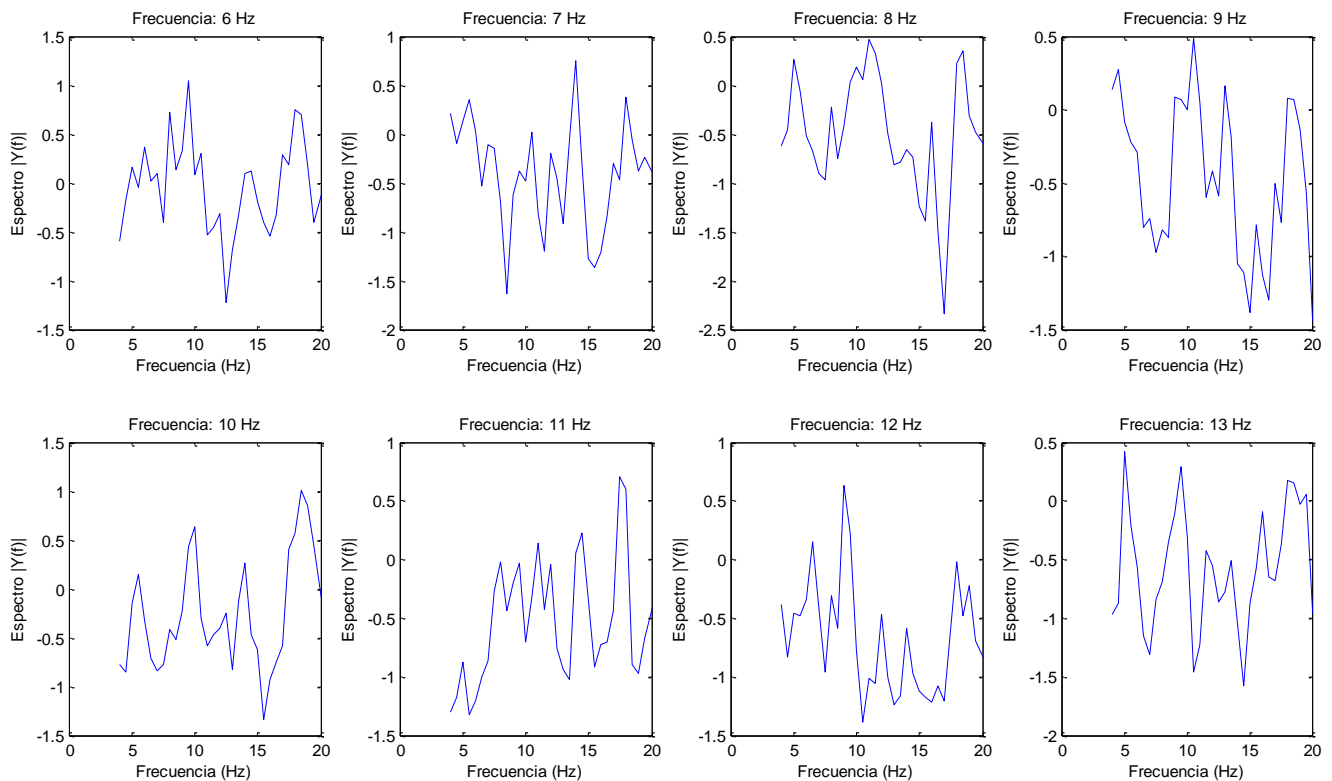


Figura 19: Resultados para un sujeto que no responde adecuadamente al estímulo.

De los 23 sujetos seleccionados, se observó en 3 de ellos que las frecuencias no tenían una amplitud suficiente como para considerarse un pico en la FFT, o ni siquiera se obtenía un pico en dicha frecuencia sino en otra distinta. En la mayoría de sujetos restantes, para el rango de 6 a 15 Hz algunas frecuencias resultaban incorrectas, pero se han obtenido al menos 4 frecuencias válidas.

Se realizaron pruebas con diferentes tipos de LEDs y disipadores de distinto grosor, así como variando el brillo de la luz emitida. Se pudo comprobar que cuanto mayor era el brillo, mejores eran los resultados, pero aumentaba la molestia y disminuía la concentración en los sujetos. De la misma forma, con brillos más bajos los resultados eran bastante pobres. Así pues, tras realizar las pruebas pertinentes se fijó un disipador de 1.5 mm de grosor, y se decidió dejar la decisión del valor del brillo a la siguiente fase, debido a que cuanto mayor sea el brillo más interferirá con la luz de los LEDs de su alrededor, y por tanto en esta decisión influye también la distancia a la que están colocados los LEDs.

Como ejemplo de análisis de la elección del disipador, en la figura 20 se puede observar en la gráfica de la izquierda el resultado del análisis de un sujeto en respuesta a una de sus mejores frecuencias, en este caso 14 Hz, con una tapa de plástico fabricada con la impresora 3D, actuando como un disipador de 1mm.

En la gráfica central se observa el efecto de la estimulación a esa misma frecuencia para una tapa de un grosor de 1.5 mm, y en la de la izquierda se utiliza un dissipador de un grosor de 1 mm que, en este caso, no se trata de una tapa translúcida sino de un componente circular con un diámetro de 4 cm. En este caso, se consiguen picos a la frecuencia esperada en los dos dissipadores en tapa, siendo el segundo el que ofrece una amplitud mayor, mientras que el tercer dissipador devuelve un resultado incorrecto. En este sujeto se aprecian amplitudes menores, pero el pico sigue siendo considerable con respecto al máximo secundario.

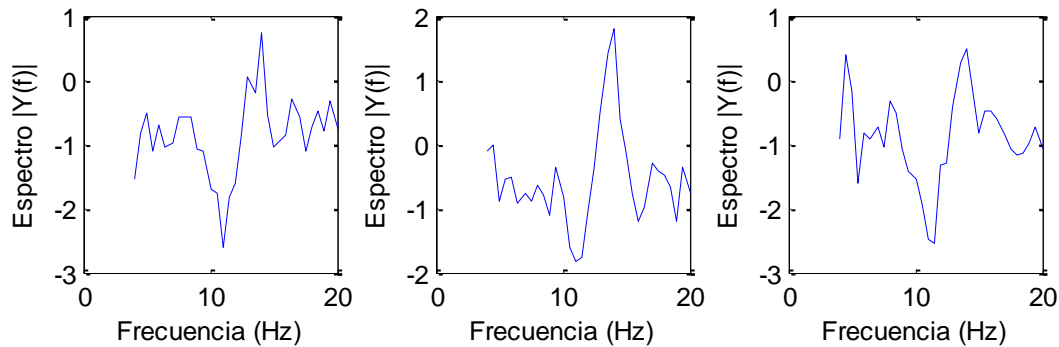


Figura 20: Ejemplo de prueba para la selección del dissipador.

Se realizaron estas pruebas en 6 sujetos, y aunque en general los resultados fueron similares para los tres casos, se observó que el dissipador en tapa de 1.5 mm era el que mejor resultados producía frente al de 1mm, mientras que el dissipador circular ofrecía peores resultados. Por este motivo, se ha elegido este dissipador frente a los otros dos, que además tiene una integración más sencilla en la caja al actuar como tapa del dispositivo.

4.3.2 Segunda fase: Fijación de características.

Si un usuario es capaz de responder a una frecuencia del estimulador, se espera que sea capaz de responder aun cuando existan factores interferentes como van a ser otras tres frecuencias oscilando simultáneamente en el campo de visión. Esto será posible siempre y cuando se seleccionen unos parámetros que permitan maximizar la obtención de resultados válidos. Los parámetros que habrá que determinar en el análisis serán la distancia a la cual colocar los LEDs, el brillo de los mismos y su color. Para explorar un parámetro, es necesario fijar el resto para no realizar pruebas en la que influyan dos variables, ya que si no resulta difícil identificar el factor que mejora o empeora los resultados.

El primer parámetro que se va a determinar es la distancia entre los LEDs. Puede deducirse intuitivamente que cuanto más separados estén los LEDs, menos interferirán unos con otros y los resultados obtenidos serán de una calidad mayor. Así pues, se comenzará con una distancia pequeña y se irá aumentando hasta conseguir unos resultados aceptables para un porcentaje alto de sujetos.

Se realizó esta prueba en 5 voluntarios, y para una separación entre diagonales de 8 cm comenzaron a encontrarse resultados adecuados en tres de ellos (ver Figura 21). Sin embargo, se decidió fijar la distancia en 9 cm ya que el diseño de la caja venía limitado por sus componentes en 12cm x 13cm, dejando así un margen de 1.5 cm y 2 cm entre los LEDs y los extremos de la caja.

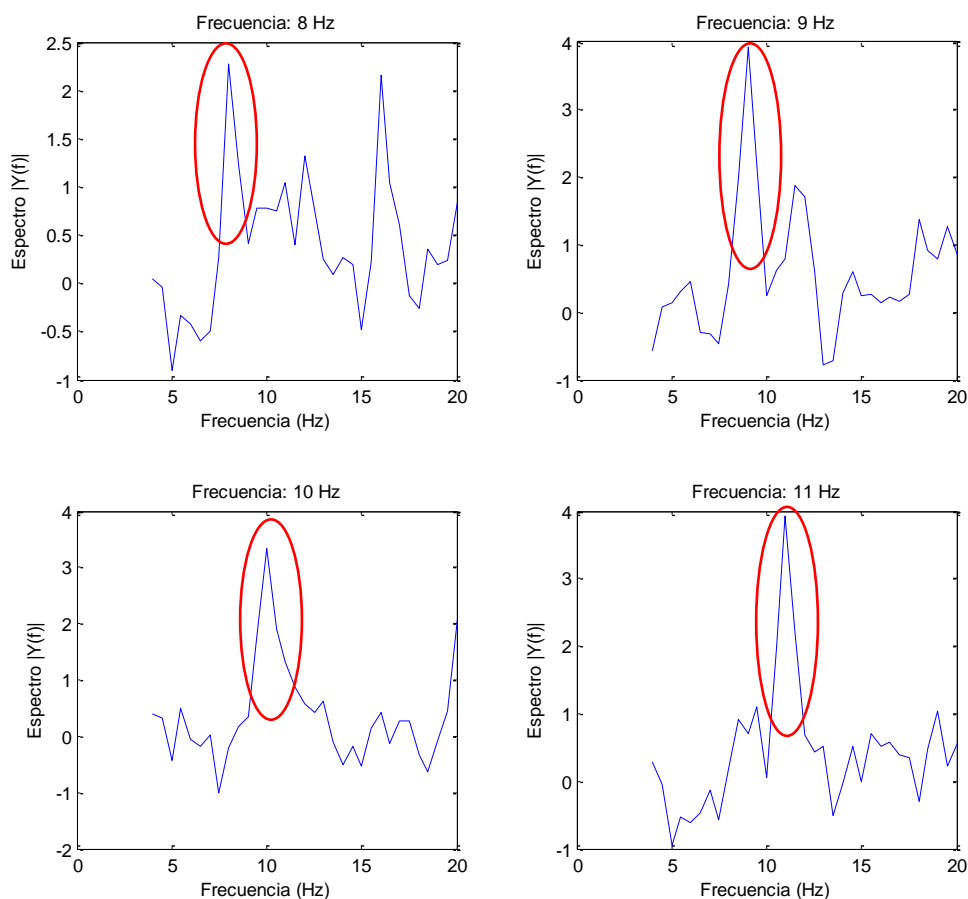


Figura 21: Resultados de la estimulación para una separación entre LEDs de 8 cm

En la figura 21 puede observarse en 8Hz un segundo pico, pero se trata de un armónico a 16 Hz y no se tiene en cuenta.

Una vez fijada la distancia entre LEDs, va a examinarse el brillo de los mismos. Inicialmente se pensó en incluir en el código de barrido de LEDs un barrido de brillo, ofreciendo así la opción de seleccionar el brillo más adecuado a cada usuario, pero para ver resultados es necesario determinar una frecuencia que funcione para el sujeto, y en esa frecuencia aplicar el barrido de brillo. Debido a que no es un sistema en tiempo real no se pueden analizar los resultados hasta que no acaba la prueba, así que o se implementaba un barrido que para cada frecuencia hiciese otro barrido de brillo o era necesario realizar tres pruebas en cada sujeto: una para el barrido en frecuencia, otra para el barrido en brillo y otra para la detección de parámetros. Teniendo en cuenta que cada muestra son grabaciones de 20 segundos y que los sujetos son voluntarios, se decidió descartar esta posibilidad y establecer un brillo fijo que funcione bien para un número alto de sujetos. No obstante, si las circunstancias lo requirieran se podría cambiar el brillo en el código del programa.

Al tratarse de pines de PWM, se pueden configurar de forma analógica a un valor de entre 0 y 255, siendo el 0 apagado y el 255 brillo máximo. Como los LEDs a utilizar son de alto brillo, 255 es un valor demasiado alto y provoca incomodidad y alta interferencia entre los LEDs. A menos de 180 el brillo es demasiado bajo como para obtener buenos resultados. Así pues, se han realizado barridos en frecuencia para cada sujeto en alguna de sus mejores frecuencias, para valores de 250, 225 y 200. En la figura 22 se muestra en la gráfica de la izquierda un análisis con los LEDs configurados a 250, en la central a 225 y en la de la derecha a 200.

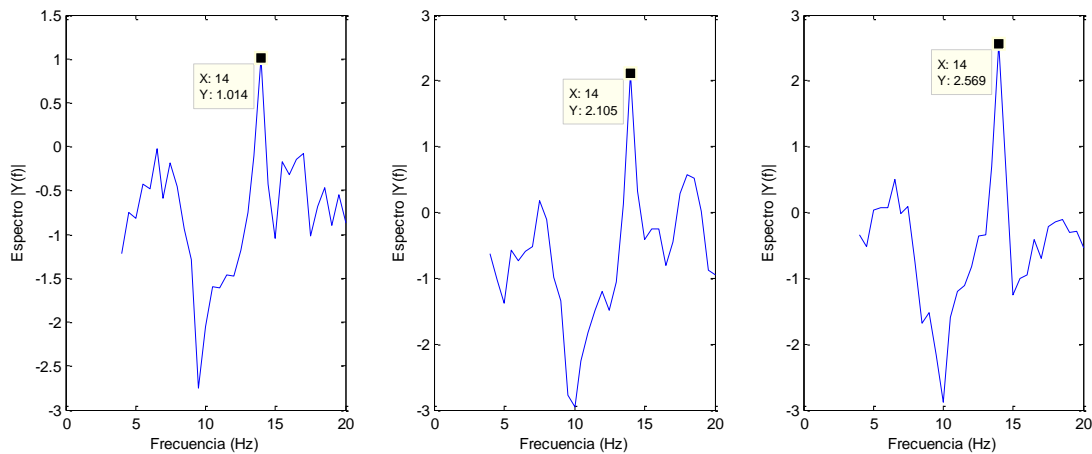


Figura 22: Comparación de resultados con diferente intensidad de brillo

Los resultados obtenidos en los 4 sujetos en los que se ha realizado esta prueba son similares, y para los tres valores de brillo la única diferencia que hay es en amplitud. Para el valor de 200 es donde se obtiene el máximo de respuesta, y a partir de ese valor comienza a decrecer hasta desaparecer el pico en 180. Así pues, el valor que se utilizará en esta caja para la configuración del PWM es de 200.

Por último, se ha querido comprobar si el color de los LEDs influye de cara a los resultados. Para una distancia de 9 cm entre diagonales y un brillo de 200, se ha realizado un análisis de resultados para cada una de las cuatro mejores frecuencias, primero con todos los LEDs de estimulación de color blanco y posteriormente variando los colores, de manera que a cada LED le correspondiera un color. Se han elegido los colores verde, blanco, rojo y ámbar. Se ha observado en los sujetos que el color de los LEDs reduce la incomodidad al fijar la atención en un solo LED durante un periodo de tiempo, y ayuda a mantener la concentración. De cara a los resultados, se comprueba que para los 5 experimentos realizados inicialmente no se aprecia diferencia, pero a lo largo del tiempo, cuando se utilizan LEDs de colores los resultados siguen siendo buenos mientras que con todos los LEDs blancos comienzan a degradarse las señales, debido posiblemente a que los sujetos comienzan a desenfocar la vista y los otros LEDs comienzan a interferir.

4.4 Pruebas en personas con discapacidad.

Siendo el objetivo de este trabajo la creación de un dispositivo que permita a personas discapacitadas una vía de comunicación, se ha querido probar en individuos con algún tipo de discapacidad física, no con la intención de comprobar si el dispositivo SSVEP funciona según lo esperado, que es el objetivo del apartado anterior, sino con el de verificar si ofrece una alternativa como medio de comunicación.

Con este objetivo, se contactó con el centro "Cottolengo", una residencia para mujeres con diferentes tipos de minusvalía, que autorizó la realización de una serie de pruebas en tres casos distintos: la primera paciente sufre de una tetraplejia, que limita seriamente el movimiento de brazos y piernas y tiene también dificultades de habla. En el segundo caso el problema que limita su comunicación es una afasia consecuencia de un ictus cerebral, con problemas para producir o comprender el lenguaje. Por último, se ha tratado un caso de parálisis cerebral que afecta a su psicomotricidad. Las pruebas se realizaron no solamente con el consentimiento médico del centro sino también con el de las propias pacientes.

Para los tres casos, se han realizado las mismas pruebas en cuatro fases diferenciadas:

- Primero, se realizó una grabación de 60 segundos observando la caja en estado de reposo, obteniendo así el baseline.
- Seguidamente, se realizó el barrido en frecuencia de 6 a 14 Hz, prueba que duró 3 minutos y 40 segundos en los que cada 22 segundos se cambiaba el LED en el que fijar la atención. Cuando se obtuvieron ambos datos, se procesaron con Matlab para encontrar las cuatro mejores frecuencias y se guardaron en la caja. En este punto, se cargó en la caja el segundo programa, que configuró las frecuencias en los LEDs.
- Por hacer el programa más intuitivo, se encendieron sólo los LEDs rojo y verde oscilando a las dos mejores frecuencias obtenidas, asignando al LED verde la instrucción "Sí" y al rojo el valor "No". La tercera fase consiste pues en la realización de tres preguntas, en las que la paciente ha de fijar su atención en el LED correcto durante 20 segundos. Esta prueba no sólo evalúa la capacidad del paciente de fijar su atención en un LED, sino también su capacidad de comprensión de la pregunta que se le hace.
- Finalmente, se cargaron las cuatro frecuencias en los LEDs. En este caso, resultó complicado hacer entender a las pacientes las instrucciones asociadas a cada luz, e incluso una vez entendidas encontrar la respuesta a la pregunta también causaba problemas. Por ello, se decidió finalmente que escogieran su color favorito, y dos preguntas sencillas bajo los valores "Mucho", "Poco", "Siempre", "Nunca".

A continuación, se describen los resultados para cada una de las tres pacientes:

I. Paciente A.

Como se ha mencionado anteriormente, en este caso la paciente sufre de una tetraplejía a causa de una enfermedad degenerativa que la inmoviliza casi por completo. Tiene control de una mano, que utiliza para manejar una silla de ruedas, y también sobre los gestos de la cara a través de los cuales se comunica, pero poco a poco va perdiendo movilidad.

Para el barrido en frecuencia no se obtienen resultados, no tanto por el fenómeno de BCI Illiteracy sino debido a que la paciente no fijaba su atención y cerraba los ojos. Estos factores también han de ser tenidos en cuenta a la hora de elaborar este tipo de dispositivos, disminuyendo el tiempo para cada frecuencia o reduciendo el número de frecuencias del barrido. El único valor en el que se obtienen resultados es a 9 Hz, que será uno de los valores a utilizar en la siguiente fase. Se utilizó también el valor de 8 Hz, pues aunque no se produzca un pico en ese punto, sí que existe uno en su primer armónico, 16 Hz.

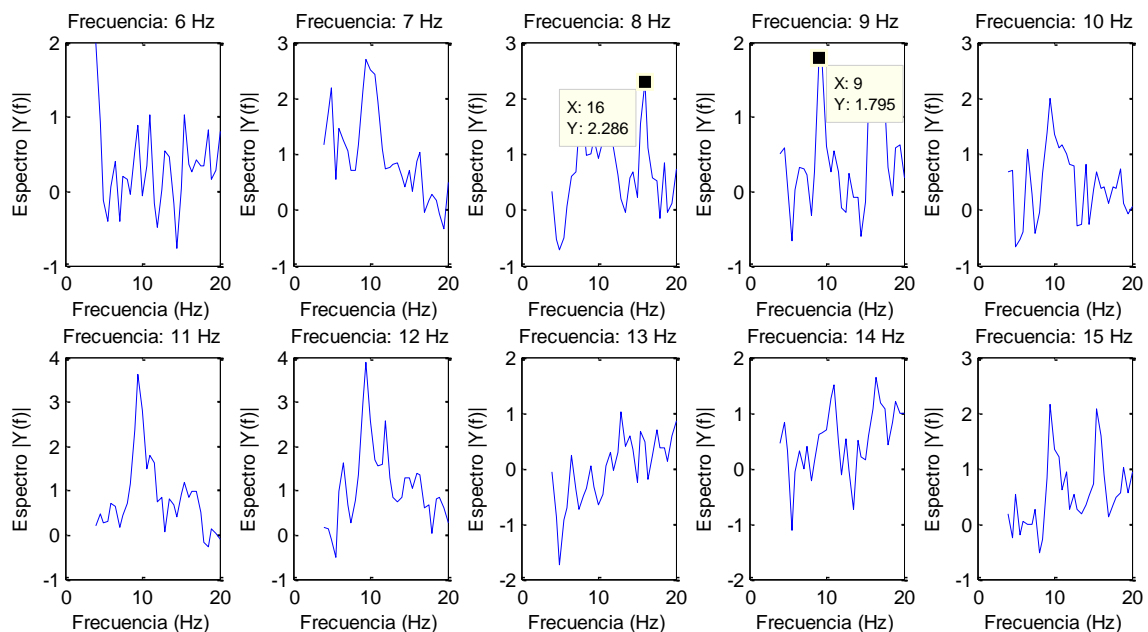


Figura 23: Barrido de frecuencias en la paciente A

Para la segunda prueba, se realizaron tres preguntas sencillas, de respuesta "Sí" (9Hz) o "No" (8Hz). La primera pregunta fue "¿Te llamas [Nombre del Sujeto]?", a lo que la paciente respondió que "Sí", aunque en el resultado no es claro en el análisis en frecuencia, que se observa en la primera gráfica de la Figura 24.

La segunda pregunta realizada fue que si tenía hambre, a lo que la paciente respondió que "Sí", pero el resultado tampoco se aprecia con claridad (segunda gráfica de la Figura 24). Por último, se le preguntó si había comido puré esa tarde, de nuevo siendo la respuesta "Sí", y esta vez fue registrada con éxito como puede verse en la tercera gráfica.

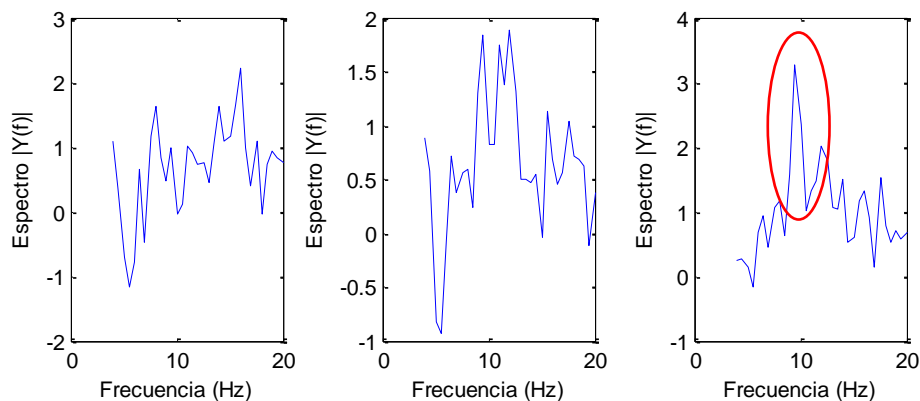


Figura 24: Análisis de preguntas de "Sí" y "No" en la paciente A.

En esta paciente no fue posible realizar la cuarta fase porque las instrucciones le resultaron confusas, sólo se hizo una prueba con las cuatro luces encendidas a la vez a 8Hz (verde), 9Hz (blanco), 12 Hz (rojo) y 15 Hz (ámbar) pidiéndole que se concentrara en su luz favorita, y este fue el resultado es un pico a 8 Hz (Figura 25), que se corresponde con el LED de color verde que es en el que se concentró. En esta figura se observa otro pico a 16 Hz, se trata del primer armónico.

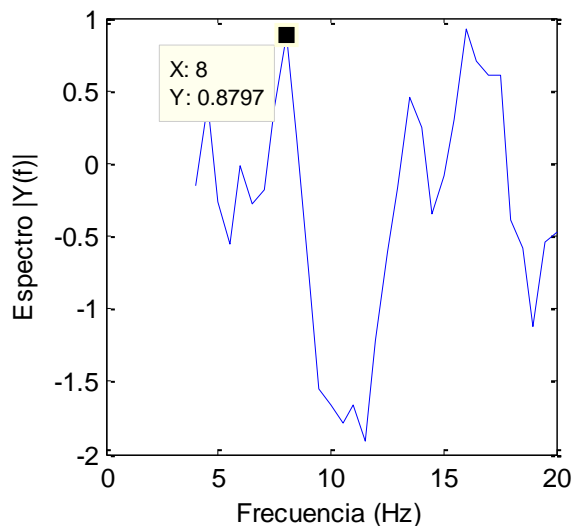


Figura 25: Prueba con cuatro luces en la paciente A.

II. Paciente B.

En este caso, la paciente que se sometió a la prueba sufría de afasia, un trastorno del lenguaje debido a una lesión cerebral, en este caso un ictus.

A pesar de este trastorno, tiene una alta capacidad de concentración, y sí que realizó bien la fase del barrido. Sin embargo, debido al BCI Illiteracy, en este caso no se obtuvo ningún valor de frecuencia utilizable, por lo que no se pudo continuar con las siguientes pruebas.

III. Paciente C.

Por último, se hicieron las pruebas en una paciente con parálisis cerebral. A diferencia de la primera paciente este trastorno no es degenerativo, y a día de hoy es capaz de comunicarse a través de una tablet mediante iconos e incluso escribiendo, controlando el dispositivo con pequeños gestos de la mano. No tiene dificultades a la hora de entender y comprender lo que se le dice, y mantuvo la concentración a lo largo de la fase de barrido. Observando el barrido en la Figura 26, sus mejores frecuencias se encuentran en 11, 12 y 13 Hz, existiendo también en 7 Hz un pico en su armónico.

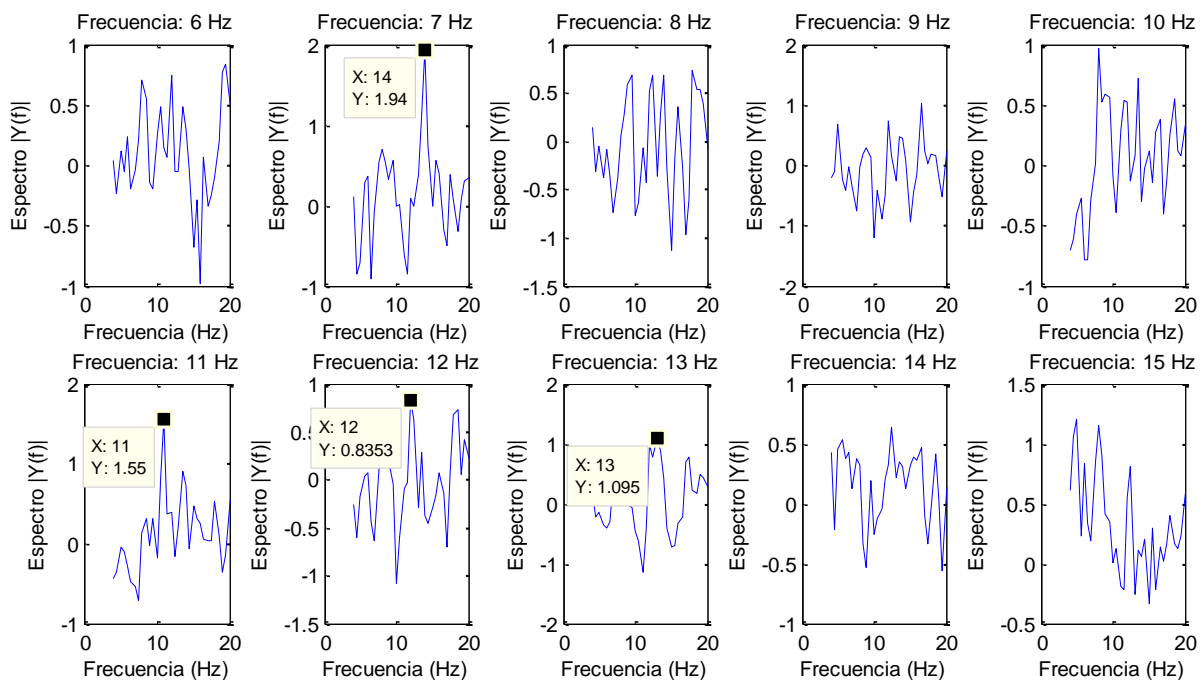


Figura 26: Barrido de frecuencias en la paciente C.

En la segunda fase se utilizaron dichas frecuencias en las respuestas de "Sí" (12Hz) y "No" (13Hz). La primera pregunta que se le hizo fue, de nuevo, "¿Te llamas [Nombre del Sujeto]?", y se puede observar en la primera gráfica de la Figura 27 que la respuesta fue "Sí", ya que existe un pico a 12 Hz.

Seguidamente, se le preguntó si estaba lloviendo, a lo que contestó con "No", hecho que se vió reflejado en el análisis con un pico a la frecuencia de 13 Hz (segunda gráfica de la Figura 27). Con esta pregunta se comprueba su capacidad de comprensión de la pregunta y del entorno, puesto que se trataba de un día soleado y la respuesta es correcta. Por último, se le preguntó si en esa residencia había peces, a lo que empezó contestando con "No" y a mitad del análisis rectificó al recordar que sí había peces. Al realizarse el registro la mitad del tiempo mirando la luz a 12 Hz y la otra mitad a 13 Hz, el resultado ha sido un pico en 12 Hz y uno de amplitud mayor en 13 Hz.

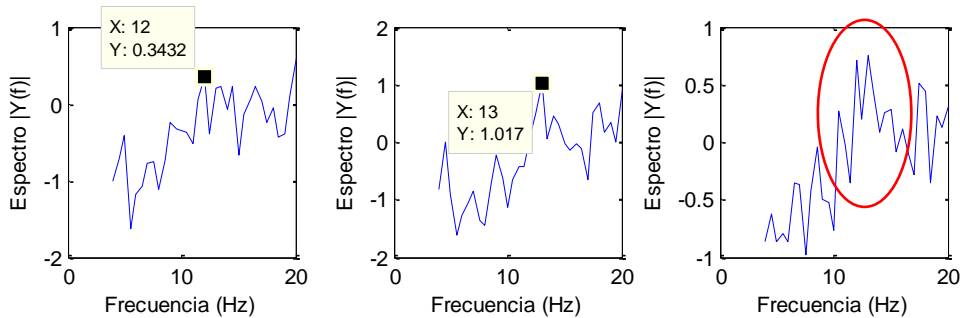


Figura 27: Análisis de preguntas de "Sí" y "No" en la paciente C.

En este caso, dada la elevada capacidad de comprensión de la paciente, sí que se pudo realizar la cuarta fase. Se configuraron las frecuencias añadiendo los valores de 7 y 8 Hz, quedando 12Hz arriba, 7Hz abajo, 13 Hz a la derecha y 11 Hz a la izquierda. Se pidió que fijase su atención en su color preferido y escogió la luz ámbar, a 8Hz, observándose un pico a 8.5 Hz en el análisis. Para la segunda prueba no se detectó resultado, pero en la última se le preguntó cuántas veces había montado en avión, ofreciendo las opciones de "mucho" (arriba), "poco" (abajo), "siempre" (izquierda) y "nunca" (derecha), a lo que la paciente respondió mirando hacia "nunca", y se observa el pico en 13.5 Hz. Aunque no son los valores exactos en frecuencia, al estar limitado el rango de opciones a 7, 8, 12 y 13 Hz, se aproximan al resultado esperado.

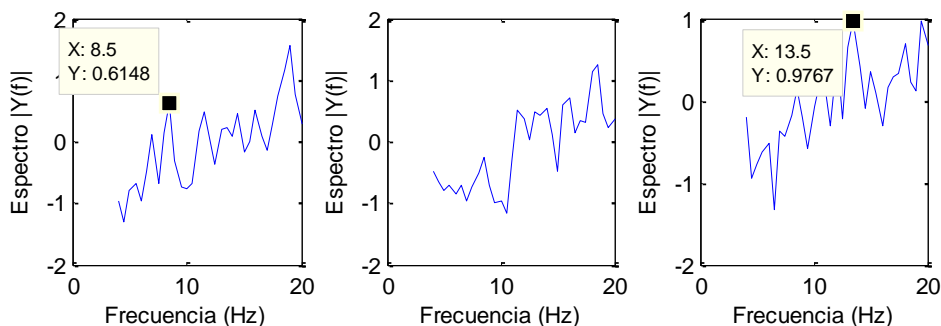


Figura 28: Análisis con cuatro opciones.

5. Conclusiones y estudio futuro

En este trabajo se ha diseñado, implementado y validado un dispositivo portátil, inalámbrico y de bajo coste para la estimulación exógena en BCIs basados en Steady State Visually Evoked Potentials. El diseño basado en el control con Arduino Yún y en un soporte fabricado con impresión 3D permite la configuración de los parámetros de estimulación para cada usuario, lo que puede contribuir a maximizar la eficacia en este tipo de interfaces cerebro-máquina.

Después de todas las pruebas analizadas, se puede concluir que se trata de un dispositivo de estimulación SSVEP válido, que genera las frecuencias de estimulación especificadas con precisión y que es capaz de modificarlas en función de la respuesta que se registre en el electroencefalograma de forma casi instantánea. Aunque no esté implementado por no realizarse el análisis en tiempo real, a lo largo de una sesión BCI la interfaz podría mandar valores de frecuencias nuevos al dispositivo, y se configurarían en el momento, resultando en una adaptación inmediata al usuario.

A pesar de que la muestra de sujetos con la que se ha validado el dispositivo es relativamente baja, se ha comprobado su funcionamiento como dispositivo de comunicación siempre que la persona que lo utilice sea capaz de comprender lo que se le dice y pueda mantener fija su atención visual en el dispositivo.

También se ha comprobado el efecto del "BCI Illiteracy", demostrando que cada persona presenta particularidades específicas en su actividad cerebral que pueden ser utilizadas para mejorar el rendimiento, aunque no en todos los casos un dispositivo SSVEP configurable es la solución.

Como líneas de trabajo futuro, sería conveniente implementar un sistema en tiempo real que fuera capaz de transformar los datos para ser analizados en frecuencia en ventanas de tiempo cortas, de forma que tan pronto como se registrara un pico en frecuencia se transfiriera ese dato al dispositivo, permitiendo agilizar el análisis y obtener resultados visibles haciendo uso de los LEDs de confirmación. El sistema de análisis en tiempo real que se desarrollará en el futuro permitirá establecer un ciclo cerrado para encontrar de forma interactiva y online las mejores frecuencias para cada sujeto.

Convendría también, una vez implementado el tiempo real y agilizando así el proceso de barrido de frecuencias, sustituir los cuatro LEDs por un array de LEDs, de tal forma que fuese posible determinar la distancia entre LEDs óptima para cada sujeto, así como ajustar el brillo de los LEDs para cada usuario.

También sería una línea a seguir el aumentar de cuatro a un número mayor de frecuencias, permitiendo un número más alto de instrucciones. Además, sería conveniente asociar cada instrucción a algo intuitivo, como puede ser una imagen, para facilitar la comprensión del uso del dispositivo por el sujeto.

Por otra parte, se podría abrir otra línea de investigación en la que el dispositivo de estimulación fuera auditivo en lugar de visual, para ofrecer una alternativa a aquellos sujetos que debido al BCI Illiteracy no responden a estímulos visuales, o para su uso simultáneo con la estimulación visual.

6. Referencias

- [1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clin. Neurophysiol.*, vol. 113, no. 6, pp. 767–791, 2002.
- [2] N. E. and da S. F. L. (2004), "Electroencephalography: Basic Principles, Clinical Applications and Related Fields." .
- [3] G. R. Müller-Putz, R. Scherer, C. Brauneis, and G. Pfurtscheller, "Steady-state visual evoked potential (SSVEP)-based communication: impact of harmonic frequency components.," *J. Neural Eng.*, vol. 2, no. 4, pp. 123–130, 2005.
- [4] C. S. Herrmann, "Human EEG responses to 1-100 Hz flicker: resonance phenomena in visual cortex and their potential correlation to cognitive phenomena," *Exp. Brain Res.*, vol. 137, no. 3–4, pp. 346–353, 2001.
- [5] "<http://www.arduino.cc>." .
- [6] M. Byczuk, P. Poryzala, and a. Materka, "On diversity within operators' EEG responses to LED-produced alternate stimulus in SSVEP BCI," *Bull. Polish Acad. Sci. Tech. Sci.*, vol. 60, no. 3, pp. 447–453, Jan. 2012.
- [7] B. Blankertz, C. Sannelli, S. Halder, E. M. Hammer, A. Kübler, K.-R. Müller, G. Curio, and T. Dickhaus, "Neurophysiological predictor of SMR-based BCI performance.," *Neuroimage*, vol. 51, no. 4, pp. 1303–1309, 2010.
- [8] C. Vidaurre and B. Blankertz, "Towards a cure for BCI illiteracy.," *Brain Topogr.*, vol. 23, no. 2, pp. 194–198, 2010.
- [9] [Http://www.createitreal.com/](http://www.createitreal.com/), "Process." .
- [10] [Http://www.iriscom.org](http://www.iriscom.org), "Iriscom." .
- [11] [Http://www.nlm.nih.gov/medlineplus/](http://www.nlm.nih.gov/medlineplus/), "MedlinePlus." .
- [12] "Historia de los ordenadores,
<http://www.uclm.es/Bloque%201/Historia.htm>." .
- [13] "A 45 años de la primera aparición pública del mouse,
<http://www.lanacion.com.ar/1646153-a-45-anos-de-la-primera-aparicion-publica-del-mouse>." .
- [14] C. Guger, B. Z. Allison, B. Großwindhager, R. Prückl, C. Hintermüller, C. Kapeller, M. Bruckner, G. Krausz, and G. Edlinger, "How many people could use an SSVEP BCI?," *Front. Neuroprosthetics*, vol. 6, p. 169, 2012.

- [15] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain Computer Interfaces, a Review," *Sensors*, vol. 12, no. 2, pp. 1211–1279, 2012.
- [16] P. F. Diez, S. M. Torres Müller, V. A. Mut, E. Laciari, E. Avila, T. F. Bastos-Filho, and M. Sarcinelli-Filho, "Commanding a robotic wheelchair with a high-frequency steady-state visual evoked potential based brain-computer interface.," *Med. Eng. Phys.*, vol. 35, no. 8, pp. 1155–1164, 2013.
- [17] N. Birbaumer, "Breaking the silence: brain-computer interfaces (BCI) for communication and motor control," *Psychophysiology*, vol. 43, no. 6, pp. 517–532, 2006.
- [18] M. van Gerven, J. Farquhar, R. Schaefer, R. Vlek, J. Geuze, A. Nijholt, N. Ramsey, P. Haselager, L. Vuurpijl, S. Gielen, and P. Desain, "The brain-computer interface cycle," *J. Neural Eng.*, vol. 6, no. 4, p. 41001, 2009.
- [19] O. Friman, I. Volosyak, and A. Graser, "Multiple Channel Detection of Steady-State Visual Evoked Potentials for Brain-Computer Interfaces," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 4, pp. 742–750, 2007.
- [20] S. Ruiz, K. Buyukturkoglu, M. Rana, N. Birbaumer, and R. Sitaram, "Real-time fMRI brain computer interfaces: self-regulation of single brain regions to networks.," *Biol. Psychol.*, vol. 95, pp. 4–20, 2014.
- [21] J. Sulzer, S. Haller, F. Scharnowski, N. Weiskopf, N. Birbaumer, M. L. Blefari, A. B. Bruehl, L. G. Cohen, R. C. DeCharms, R. Gassert, R. Goebel, U. Herwig, S. LaConte, D. Linden, A. Luft, E. Seifritz, and R. Sitaram, "Real-time fMRI neurofeedback: progress and challenges.," *Neuroimage*, vol. 76, pp. 386–399, 2013.
- [22] O. Jensen, A. Bahramisharif, R. Oostenveld, S. Klanke, A. Hadjipapas, Y. O. Okazaki, and M. A. J. van Gerven, "Using brain-computer interfaces and brain-state dependent stimulation as tools in cognitive neuroscience.," *Front. Psychol.*, vol. 2, p. 100, Jan. 2011.
- [23] G. Schalk and E. C. Leuthardt, "Brain-computer interfaces using electrocorticographic signals," *IEEE Rev. Biomed. Eng.*, vol. 4, pp. 140–154, 2011.
- [24] F. Cincotti, D. Mattia, F. Aloise, S. Bufalari, G. Schalk, G. Oriolo, A. Cherubini, M. G. Marciani, and F. Babiloni, "Non-invasive brain-computer interface system: towards its application as assistive technology.," *Brain Res. Bull.*, vol. 75, no. 6, pp. 796–803, 2008.
- [25] S. Halder, D. Agorastos, R. Veit, E. M. Hammer, S. Lee, B. Varkuti, M. Bogdan, W. Rosenstiel, N. Birbaumer, and A. Kübler, "Neural mechanisms of brain-computer interface control.," *Neuroimage*, vol. 55, no. 4, pp. 1779–1790, 2011.

- [26] C. Jia, X. Gao, B. Hong, and S. Gao, "Frequency and phase mixed coding in SSVEP-based brain-computer interface.," *IEEE Trans Biomed Eng.*, vol. 58, no. 1, pp. 200–206, 2011.
- [27] R. Fazel-Rezai, B. Z. Allison, C. Guger, E. W. Sellers, S. C. Kleih, and A. Kübler, "P300 brain computer interface: current challenges and emerging trends.," *Front. Neuroeng.*, vol. 5, no. July, p. 14, 2012.
- [28] C. Brunner, B. Z. Allison, C. Altstätter, and C. Neuper, "A comparison of three brain-computer interfaces based on event-related desynchronization, steady state visual evoked potentials, or a hybrid approach using both signals.," *J. Neural Eng.*, vol. 8, no. 2, p. 25010, 2011.
- [29] M. Kamrunnahar, N. S. Dias, S. J. Schiff, and B. J. Gluckman, "Model-based responses and features in Brain Computer Interfaces.," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2008, pp. 4482–4485, 2008.
- [30] J.-H. Lim, H.-J. Hwang, C.-H. Han, K.-Y. Jung, and C.-H. Im, "Classification of binary intentions for individuals with impaired oculomotor function: 'eyes-closed' SSVEP-based brain-computer interface (BCI).," *J. Neural Eng.*, vol. 10, no. 2, p. 26021, 2013.
- [31] E. Yin, Z. Zhou, J. Jiang, F. Chen, Y. Liu, and D. Hu, "A novel hybrid BCI speller based on the incorporation of SSVEP into the P300 paradigm.," *J. Neural Eng.*, vol. 10, no. 2, p. 26012, 2013.
- [32] C. Guger, S. Daban, E. Sellers, C. Holzner, G. Krausz, R. Carabalona, F. Gramatica, and G. Edlinger, "How many people are able to control a P300-based brain-computer interface (BCI)?," *Neurosci Lett*, vol. 462, no. 1, pp. 94–98, Oct. 2009.
- [33] N. Birbaumer, N. Ghanayim, T. Hinterberger, I. Iversen, B. Kotchoubey, A. Kübler, J. Perelmouter, E. Taub, and H. Flor, "A spelling device for the paralysed.," *Nature*, vol. 398, no. 6725, pp. 297–298, 1999.
- [34] G. R. Müller-Putz and G. Pfurtscheller, "Control of an electrical prosthesis with an SSVEP-based BCI.," *IEEE Trans. Biomed. Eng.*, vol. 55, no. 1, pp. 361–364, 2008.
- [35] N. E. and da Silva, "Electroencephalography: Basic Principles, Clinical Applications and Related Fields." .
- [36] F. Di Russo, S. Pitzalis, T. Aprile, G. Spitoni, F. Patria, A. Stella, D. Spinelli, and S. A. Hillyard, "Spatiotemporal analysis of the cortical sources of the steady-state visual evoked potential.," *Hum. Brain Mapp.*, vol. 28, no. 4, pp. 323–334, 2007.
- [37] I. Volosyak, H. Cecotti, and A. Gräser, "Impact of Frequency Selection on LCD Screens for SSVEP Based Brain-Computer Interfaces," in *Proc. 4th Int. IEEE/EMBS Conference on Neural Engineering NER 09*, 2009, pp. 706–713.

- [38] I. Volosyak, "SSVEP-based Bremen-BCI interface—boosting information transfer rates," *J. Neural Eng.*, vol. 8, no. 3, p. 36020, 2011.
- [39] J. Fernandez-Vargas, H. U. Pfaff, F. B. Rodriguez, and P. Varona, "Assisted closed-loop optimization of SSVEP-BCI efficiency," *Front. Neural Circuits*, vol. 7, p. Article 27, 2013.
- [40] Y. Zhang, P. Xu, T. Liu, J. Hu, R. Zhang, and D. Yao, "Multiple Frequencies Sequential Coding for SSVEP-Based Brain-Computer Interface.," *PLoS One*, vol. 7, no. 3, p. e29519, 2012.
- [41] Z. Wu, Y. Lai, Y. Xia, D. Wu, and D. Yao, "Stimulator selection in SSVEP-based BCI.," *Med. Eng. Phys.*, vol. 30, no. 8, pp. 1079–1088, 2008.
- [42] CST, "CST - Computer Simulation Technology." .
- [43] A. Delorme, T. Mullen, C. Kothe, Z. Akalin Acar, N. Bigdely-Shamlo, A. Vankov, and S. Makeig, "EEGLAB, SIFT, NIFT, BCILAB, and ERICA: New tools for advanced EEG processing," *Comput. Intell. Neurosci.*, vol. 2011, p. 130714, 2011.
- [44] A. Delorme and S. Makeig, "EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis," *J. Neurosci. Methods*, vol. 134, no. 1, pp. 9–21, 2004.

Anexos

Anexo I: Código de encendido de los LEDs.

A) Encendido de un LED utilizando retardos:

```
const int led1Pin = 9; //Pin de la placa Arduino.
const float freq = 11;
const float interval1 = (1000000/(2*freq));
int state = HIGH;

void setup() {
  pinMode(led, OUTPUT); } // Se declara el pin 9 como salida.

void loop()
{
  digitalWrite(led1Pin, state);
  if(state=HIGH) //Cambia el estado del LED.
    state=LOW;
  else
    state=HIGH;

  delay(interval1); //Espera medio periodo.
}
```

B) Encendido de un LED sin retardos:

Este es el código que se repetirá para los cuatro LEDs:

```
unsigned long previousMicros1=0;

void setup() {
  pinMode(led, OUTPUT); }

void loop()
{
  unsigned long currentMicros1 = micros();

  if(currentMicros1 - previousMicros1 >= interval1)
  {
    previousMicros1 = currentMicros1;

    if (state == LOW)
      state = HIGH;
    else
      state = LOW;

    digitalWrite(led1Pin, led1State);}
```


Anexo II: Código del Sistema Cliente - Servidor.

Por simplificar, se muestra sólo la parte del código que corresponde a la implementación del sistema.

```
void setup()
{...
    // Se crea el Bridge
    Bridge.begin();
    // Le dice al servidor que escuche solo del LocalHost (en
    // el puerto 80)

    server.listenOnLocalhost();
    server.begin();
...}

void loop()
{...
// Lo primero que hara este bucle es procesar los cambios en los
// LEDs, modificarlos y una vez hecho esto pondra las
// frecuencias a oscilar correctamente.
// El led 13 avisara de cuando se pueden mirar las freqs.
// Se comprobará si existe un cliente nuevo cada intervalo de
// tiempo definido por "cont"

    if (cont == 62500)
    {
        YunClient client = server.accept();

        //Si existe un cliente...
        if (client)
        {
            digitalWrite(13, HIGH); // PIN 13 indica que se
            // estan haciendo cambios y aun no hay que mirar
            String command = client.readStringUntil('/');

            if (command == "arriba")
            {
                ledArriba = client.parseFloat();

                // Se escribe por pantalla para asegurar el cambio
                client.print(F
                ("El LED de arriba se ha configurado a una frecuencia de "));
                client.print(ledArriba);
                client.println(F("Hz. "));

                //Pasamos de frecuencia a periodo
                interval1 = (1000000/(2*ledArriba));
            }
        }
    }
...}
```

Este código se repetiría para el caso de que "command" fuese igual a "abajo", "derecha" e "izquierda". Posteriormente, se saldría de la función if y continuaría con el código de iluminación de los LEDs.

Anexo 3: Habilitación de la tarjeta microSD.

Se muestra sólo la parte del código en la que se guardan o cargan los datos. Cuando se recibe un cliente, se procesa el comando para ver si la función recibida es "guardar", "cargar", "arriba", "abajo", "derecha" o "izquierda". Se muestra aquí el código desde esa comprobación.

A) Función "guardar"

```
...
if (command == "guardar")
{
    // Se lee el usuario que se pasa por el navegador
    String user = client.readStringUntil('\r');
    user.toCharArray(usuario, buff);

    client.print(F("El usuario a guardar es: " ));
    client.println(usuario);

    // Se crea el String que incluye el directorio y nombre del
    // archivo
    strcpy(NombreFichero, "/mnt/sd/");
    strcat(NombreFichero, usuario);
    strcat(NombreFichero, ".txt");

    // Se comprueba si el usuario existía previamente
    if(FileSystem.exists(NombreFichero))
        client.println("Este fichero ya existe, se va a sobrescribir.");
    else
        client.println("Nuevo usuario.");

    File dataFile =FileSystem.open(NombreFichero, FILE_WRITE);

    // Se guardan en el fichero el valor de las frecuencias
    dataFile.println(ledArriba);
    dataFile.println(ledAbajo);
    dataFile.println(ledDerecha);
    dataFile.println(ledIzquierda);

    client.println("Frecuencias guardadas.");

    // Se cierra el fichero para continuar con el programa
    dataFile.close();
}
...
```

B) Función "cargar"

```
...
else if (command == "cargar")
{
    // Se lee el fichero que se quiere cargar en los LEDs
    String user = client.readStringUntil('\r');
    user.toCharArray(usuario, buff);

    client.print(F("El usuario es: " ));
    client.print(usuario);

    strcpy(NombreFichero, "/mnt/sd/");
    strcat(NombreFichero, usuario);
    strcat(NombreFichero, ".txt");

    if(FileSystem.exists(NombreFichero))
    {
        File dataFileread =FileSystem.open(NombreFichero,
        FILE_READ);

        // Se lee el valor de las frecuencias
        ledArriba= dataFileread.parseFloat();
        ledAbajo= dataFileread.parseFloat();
        ledDerecha= dataFileread.parseFloat();
        ledIzquierda= dataFileread.parseFloat();

        // Se cargan estas frecuencias en los LEDs
        interval1 = (1000000/(2*ledArriba));
        interval2 = (1000000/(2*ledAbajo));
        interval3 = (1000000/(2*ledDerecha));
        interval4 = (1000000/(2*ledIzquierda));

        Serial.println("Frecuencias cargadas.");
        dataFileread.close();

    }
    else
        client.print(F("Ese usuario no esta registrado; no se
        pueden cargar los datos." ));
}
...

```